

Implementation of Geostatistical Algorithms and Applications in Geological Media Simulation

(A thesis submitted for the degree of Diploma)



Vasileios Androulakis

DEPARTMENT OF MINERAL RESOURCES ENGINEERING
Technical University of Crete

Advisory Committee:

Professor D. T. Hristopoulos (supervisor), Dept. of Mineral Resources Engineering, TUC

Professor M. G. Galetakis, Dept. of Mineral Resources Engineering, TUC

Doctor D. Kourounis, Inst. of Computational Science, Università della Svizzera italiana, Switzerland

June, 2017

Acknowledgements

I would like to heartily thank my advising committee: my supervisor Professor D. T. Hristopulos, for the opportunity he gave me to work upon an intriguing topic and for the nice cooperation we have had during this work, Doctor D. Kourounis for the advice he provided me with, along with the Marmousi Model data, and Professor M. G. Galetakis for his usefull comments.

I would also like to express my gratitude to the members of the Geostatistics Laboratory of the Mineral Resources Engineering School of Technical University of Crete Mr Manolis Petrakis, Mrs Vicky Agou and Dr Andrew Pavlides for their invaluable help during this thesis.

Abstract

A commonly encountered problem in Geosciences is that of geological media simulation. The knowledge of the geological stratigraphy and of the physical properties of the strata, e.g. porosity, permeability, velocity of seismic waves, concentration of chemical elements or minerals, is of great importance for many disciplines such as hydrogeology, mining, petroleum engineering, geophysics and environmental restoration. The increasingly sophisticated data acquisition methods provide huge amounts of information about the spatial, temporal or spatiotemporal distribution of such physical attributes. However, they can not provide an extensive and complete image of the spatial distribution of the observable properties. This is due to the fact that such information is only available at a limited number of points which are randomly or regularly distributed in space. The mapping and graphical interpretation of these properties on regular grids in one, two, or three dimensions respectively, is more desirable and helpful.

Geostatistics provides various tools for deriving such maps and graphical interpretations via statistic and stochastic methods. The most widely used geostatistical tools are: i) Kriging and ii) Simulation.

In this thesis, such tools are applied to a geological media simulation problem. More explicitly, a known gridded dataset is sampled, both regularly and randomly, and it is reconstructed with geostatistical tools. Both the regular and the random sample contain 10,16% percentage of the original data. The first sample consists of 39 columns of the grid thus mimicking data obtained by well-logging drill-holes. The second sample involves randomly selected grid points. The original dataset is the Marmousi velocity model, a synthetic two-dimensional acoustic model developed in 1988 by the Institut Francais du Petrole (IFP), and used for a workshop on practical aspects of seismic data inversion at the 1990 EAEG (European Association of Exploration Geophysicists) meeting in Copenhagen. The Marmousi model was generated using a 2-D acoustic, finite-difference modelling program, so as to resemble an overall continental drift geological setting. The geometry is based on a geological profile through the North Quenguela trough in the Cuanza basin, which is located in north-western Angola along the Atlantic Coast of West Africa. The Marmousi model is characterized by high geological complexity: numerous large normal faults and tilted blocks,

resulting from the continental drift, complicate the structure towards its centre. The model also contains many reflectors, steep dips, and strong velocity variations in both the lateral and the vertical directions.

The purpose of this thesis is the exploratory application of geostatistical tools for the simulation of the above described geological section. The methods which are used herein are: i) Ordinary Kriging and ii) Directional Gradient Curvature (DGC) simulation method. Ordinary Kriging is selected because it is the most classic and widely used geostatistical tool. DGC is a novel, non-parametric, local simulation and gap-filling method, the comparison of which with the Ordinary Kriging can provide useful information. For the implementation of Kriging, software was developed in the MATLAB programming environment, while for the implementation of DGC simulation method software developed by the Geostatistics Laboratory (Technical University of Crete) was used. Several methods of parameter inference and anisotropy estimation have been used in connection with OK. The results of each method are evaluated and compared through proper validation measures (e.g. mean squared error, maximum reconstruction error, Pearson's correlation coefficient, Spearman's correlation coefficient).

The applied methods achieve relatively high correlation between the reconstructed field and the real data, i.e., 90-95%, for both samples. In the case of Kriging, the optimal performance is obtained by including separate estimation of anisotropy and data transformation to isotropy. Finally, the DGC simulation method has shown comparable, but slightly inferior, performance measures to OK estimation. We believe that this is due to the local nature of the latter in contrast with kriging, which accounts for correlations with longer range.

Περίληψη

Ένα ευρέως διαδεδομένο πρόβλημα στις Γεωεπιστήμες αποτελεί η προσομοίωση ενός γεωλογικού μέσου. Η ακριβής γνώση της γεωλογικής στρωματογραφίας και των ιδιοτήτων των στρωμάτων (π.χ. πορώδες, διαπερατότητα, ταχύτητα σεισμικών κυμάτων, συγκέντρωση χημικών στοιχείων και ορυκτών) είναι αντικείμενο κεφαλαιώδους σημασίας με εφαρμογές σε πολλά επιστημονικά πεδία, όπως η υδρογεωλογία, η μεταλλευτική, η μηχανική πετρελαίου, η γεωφυσική και η περιβαλλοντική αποκατάσταση. Τα σύγχρονα συστήματα συλλογής δεδομένων παρέχουν σημαντική ποσότητα πληροφορίας σχετικά με την χωρική, χρονική ή χωροχρονική μεταβολή τέτοιων φυσικών ιδιοτήτων. Παρόλα αυτά, δεν μπορούν να μας δώσουν μια ολοκληρωμένη εικόνα της κατανομής των ιδιοτήτων αυτών. Αυτό οφείλεται στο γεγονός ότι η πληροφορία αυτή είναι σε γεωαναφερόμενη μορφή (*georeferenced*), δηλ. συνδέεται με συγκεκριμένα μόνο σημεία του χώρου. Η χαρτογράφηση και η γραφική απεικόνιση αυτών των ιδιοτήτων σε κανονικά πλέγματα (στις 1, 2 και 3 διαστάσεις τουλάχιστον) είναι περισσότερο χρήσιμη και επιθυμητή.

Η γεωστατιστική παρέχει διάφορα εργαλεία για τον σχεδιασμό τέτοιων χαρτών και γραφικών απεικονίσεων μέσω στατιστικών και στοχαστικών μεθόδων. Τα πλέον ευρέως χρησιμοποιούμενα γεωστατιστικά εργαλεία είναι: α) η χωρική παρεμβολή με **Kriging** και β) η Προσομοίωση.

Στην παρούσα διπλωματική εργασία, τέτοιου είδους εργαλεία εφαρμόζονται για την επίλυση ενός προβλήματος προσομοίωσης γεωλογικού μέσου. Πιο συγκεκριμένα, ανακατασκευάζεται ένα πλεγματοκό σύνολο δεδομένων (συνθετική ψηφιακή εικόνα) μέσω της γεωστατιστικής ανάλυσης δειγμάτων προερχόμενων από την αρχική εικόνα. Τα δείγματα που χρησιμοποιούνται είναι ένα κανονικό και ένα τυχαίο δείγμα. Και τα δύο αποτελούνται από ένα ποσοστό σημείων που ανέρχεται στο 10,16% του συνόλου των πλεγματοκών σημείων της εικόνας. Το πρώτο περιλαμβάνει 39 στήλες του αρχικού πλέγματος με μια γεωμετρία που μιμείται δεδομένα από διαγραφές γεωτρήσεων. Το δεύτερο δείγμα αποτελείται από τυχαία επιλεγμένα σημεία του πλέγματος. Το αρχικό σύνολο δεδομένων είναι το μοντέλο ταχύτητας **Marmousi**, ένα συνθετικό ακουστικό μοντέλο δύο διαστάσεων που κατασκευάστηκε με βάση το γεωλογικό προφίλ της τάφρου **North Quenguela**,

στην λεκάνη **Cuanza**, που βρίσκεται στην βορειοδυτική Αγκόλα και κατά μήκος της Ατλαντικής Ακτογραμμής της δυτικής Αφρικής. Το μοντέλο **Marmousi** χαρακτηρίζεται από μεγάλη γεωλογική πολυπλοκότητα: πολυάριθμα κανονικά ρήγματα και ανυψωμένα τεκτονικά κέρατα, τα οποία προκύπτουν από την μετακίνηση της γεωλογικής πλάκας, καθιστούν το κεντρικό τμήμα της δομής ιδιαίτερα περίπλοκο. Το μοντέλο περιέχει επίσης πολλές επιφάνειες ανάκλασης, απότομες βυθίσεις και μεγάλες διακυμάνσεις της ταχύτητας τόσο στην οριζόντια όσο και στην κατακόρυφη διεύθυνση.

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι η διερευνητική εφαρμογή γεωστατιστικών μεθόδων για την προσομοίωση της παραπάνω ψηφιακής εικόνας του γεωλογικού μέσου. Οι μέθοδοι που χρησιμοποιήθηκαν είναι: α) το Κανονικό **Kriging** και β) η μέθοδος προσομοίωσης Κατευθυντικής Βαθμίδας και Καμπυλότητας (**Directional Gradient Curvature (DGC)**). Το Κανονικό **Kriging** χρησιμοποιείται καθώς είναι η πιο κλασική και ευρέως χρησιμοποιούμενη μέθοδος χωρικής παρεμβολής. Η μέθοδος **DGC** είναι μια καινοτόμα, μη-παραμετρική, τοπική μέθοδος προσομοίωσης και πλήρωσης κενών, η σύγκριση της οποίας με το Κανονικό **Kriging** θα μπορούσε να δώσει χρήσιμες πληροφορίες. Για την εφαρμογή του **Kriging** αναπτύχθηκε λογισμικό στο περιβάλλον προγραμματισμού **MATLAB**, ενώ για την εφαρμογή της μεθόδου προσομοίωσης **DGC** χρησιμοποιήθηκε λογισμικό που έχει αναπτυχθεί από την ερευνητική Μονάδα Γεωστατιστικής (Πολυτεχνείο Κρήτης).

Διάφορες μέθοδοι για την εκτίμηση των παραμέτρων των εξεταζόμενων μοντέλων συνδιασποράς και την αντιμετώπιση της ανισοτροπίας υλοποιήθηκαν για την εφαρμογή του Κανονικού **Kriging**. Τα αποτελέσματα των διαφόρων μοντέλων και μεθόδων αξιολογούνται και συγκρίνονται μέσω κατάλληλων στατιστικών μέτρων (π.χ. μέσο τετραγωνικό σφάλμα, μέγιστο σφάλμα ανακατασκευής, συντελεστής συσχέτισης **Pearson**, συντελεστής συσχέτισης **Spearman**).

Οι μέθοδοι που χρησιμοποιήθηκαν επιτυγχάνουν υψηλή συσχέτιση, της τάξεως του 90-95%, μεταξύ των ανακατασκευασμένων και των πραγματικών δεδομένων και για τα δύο δείγματα. Επιπλέον, από την αξιολόγηση και τη σύγκριση αυτών προκύπτει ότι οι μέθοδοι εκτίμησης των παραμέτρων των εξεταζόμενων μοντέλων με τις καλύτερες επιδόσεις είναι εκείνες που περιλαμβάνουν τον μετασχηματισμό σε ισοτροπικές συντεταγμένες. Τέλος, η μέθοδος προσομοίωσης **DGC** επιτυγχάνει συγκρίσιμα αλλά ελαφρώς κατώτερα αποτελέσματα (της τάξεως του 89%) από αυτά του Κανονικού **Kriging**. Αυτό θεωρούμε ότι οφείλεται στον περιορισμό της τελευταίας μεθόδου να διακρίνει μόνο τοπικές συσχετίσεις σε αντίθεση με το **Kriging**, το οποίο λαμβάνει υπόψη του συσχετίσεις μεγαλύτερης ακτίνας.

Εκτεταμένη Περίληψη

Ένα ευρέως διαδεδομένο πρόβλημα στις Γεωεπιστήμες αποτελεί η προσομοίωση ενός γεωλογικού μέσου. Η ακριβής γνώση της γεωλογικής στρωματογραφίας και των ιδιοτήτων των στρωμάτων (π.χ. πορώδες, διαπερατότητα, ταχύτητα σεισμικών κυμάτων, συγκέντρωση χημικών στοιχείων και ορυκτών) είναι αντικείμενο κεφαλαιώδους σημασίας με εφαρμογές σε πολλά επιστημονικά πεδία, όπως η υδρογεωλογία, η μεταλλευτική, η μηχανική πετρελαίου, η γεωφυσική και η περιβαλλοντική αποκατάσταση. Τα σύγχρονα συστήματα συλλογής δεδομένων παρέχουν σημαντική ποσότητα πληροφορίας σχετικά με την χωρική, χρονική ή χωροχρονική μεταβολή τέτοιων φυσικών ιδιοτήτων. Παρόλα αυτά, δεν μπορούν να μας δώσουν μια ολοκληρωμένη εικόνα της κατανομής των ιδιοτήτων αυτών. Αυτό οφείλεται στο γεγονός ότι η πληροφορία αυτή είναι σε γεωαναφερόμενη μορφή (*georeferenced*), δηλ. συνδέεται με συγκεκριμένα μόνο σημεία του χώρου. Η χαρτογράφηση και η γραφική απεικόνιση αυτών των ιδιοτήτων σε κανονικά πλέγματα (στις 1, 2 και 3 διαστάσεις τουλάχιστον) είναι περισσότερο χρήσιμη και επιθυμητή.

Η γεωστατιστική παρέχει διάφορα εργαλεία για τον σχεδιασμό τέτοιων χαρτών και γραφικών απεικονίσεων μέσω στατιστικών και στοχαστικών μεθόδων. Τα πλέον ευρέως χρησιμοποιούμενα γεωστατιστικά εργαλεία είναι: α) η χωρική παρεμβολή με **Kriging** και β) η Προσομοίωση.

Στην παρούσα διπλωματική εργασία, τέτοιου είδους εργαλεία εφαρμόζονται για την επίλυση ενός προβλήματος προσομοίωσης γεωλογικού μέσου. Πιο συγκεκριμένα, ανακατασκευάζεται ένα πλεγματοκό σύνολο δεδομένων (συνθετική ψηφιακή εικόνα) μέσω της γεωστατιστικής ανάλυσης δειγμάτων προερχόμενων από την αρχική εικόνα. Τα δείγματα που χρησιμοποιούνται είναι ένα κανονικό και ένα τυχαίο δείγμα. Και τα δύο αποτελούνται από ένα ποσοστό σημείων που ανέρχεται στο 10,16% του συνόλου των πλεγματοκών σημείων της εικόνας. Το πρώτο περιλαμβάνει 39 στήλες του αρχικού πλέγματος με μια γεωμετρία που μιμείται δεδομένα από διαγραφίες γεωτρήσεων. Το δεύτερο δείγμα αποτελείται από τυχαία επιλεγμένα σημεία του πλέγματος. Το αρχικό σύνολο δεδομένων είναι το μοντέλο ταχύτητας **Marmousi**, ένα συνθετικό ακουστικό μοντέλο δύο διαστάσεων που αναπτύχθηκε το 1988 από το Ινστιτούτο Πετρελαίου της Γαλλίας (**Institut Francais**

du Petrole). Το μοντέλο αυτό χρησιμοποιήθηκε για τις εργαστηριακές ανάγκες ενός συνεδρίου της EAEG (European Association of Exploration Geophysicists) για την αντιστροφή σεισμικών δεδομένων που διεξήχθη στην Κοπεγχάγη το 1990. Το μοντέλο **Marmousi** κατασκευάστηκε με βάση το γεωλογικό προφίλ της τάφρου **North Quenguela**, στην λεκάνη **Cuanza**, που βρίσκεται στην βορειοδυτική Αγκόλα και κατά μήκος της Ατλαντικής Ακτογραμμής της δυτικής Αφρικής. Το μοντέλο **Marmousi** χαρακτηρίζεται από μεγάλη γεωλογική πολυπλοκότητα: πολυάριθμα κανονικά ρήγματα και ανυψωμένα τεκτονικά κέρατα, τα οποία προκύπτουν από την μετακίνηση της γεωλογικής πλάκας, καθιστούν το κεντρικό τμήμα της δομής ιδιαίτερα περίπλοκο. Το μοντέλο περιέχει επίσης πολλές επιφάνειες ανάκλασης, απότομες βυθίσεις και μεγάλες διακυμάνσεις της ταχύτητας τόσο στην οριζόντια όσο και στην κατακόρυφη διεύθυνση.

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι η διερευνητική εφαρμογή γεωστατιστικών μεθόδων για την προσομοίωση της παραπάνω ψηφιακής εικόνας του γεωλογικού μέσου. Οι μέθοδοι που χρησιμοποιήθηκαν είναι: α) το Κανονικό **Kriging** και β) η μέθοδος προσομοίωσης Κατευθυντικής Βαθμίδας και Καμπυλότητας (**Directional Gradient Curvature (DGC)**). Το Κανονικό **Kriging** χρησιμοποιείται καθώς είναι η πιο κλασική και ευρέως χρησιμοποιούμενη μέθοδος χωρικής παρεμβολής. Η μέθοδος **DGC** είναι μια καινοτόμα, μη-παραμετρική, τοπική μέθοδος προσομοίωσης και πλήρωσης κενών, η σύγκριση της οποίας με το Κανονικό **Kriging** θα μπορούσε να δώσει χρήσιμες πληροφορίες. Για την εφαρμογή του **Kriging** αναπτύχθηκε λογισμικό στο περιβάλλον προγραμματισμού **MATLAB**, ενώ για την εφαρμογή της μεθόδου προσομοίωσης **DGC** χρησιμοποιήθηκε λογισμικό που έχει αναπτυχθεί από την Ερευνητική Μονάδα Γεωστατιστικής (Πολυτεχνείο Κρήτης).

Στην περίπτωση του Κανονικού **Kriging**, τα βήματα της γεωστατιστικής ανάλυσης είναι τα εξής:

1. Προκαταρκτική Ανάλυση,
2. Εκτίμηση Παραμέτρων,
3. Επιλογή Βέλτιστου Μοντέλου (Διασταυρωτική Επιβεβαίωση),
4. Χωρική Εκτίμηση, και
5. Αξιολόγηση Απόδοσης

Στο βήμα της Εκτίμησης Παραμέτρων χρησιμοποιήθηκαν πέντε παραλλαγές για την εκτίμηση των παραμέτρων των εξεταζόμενων μοντέλων. Οι παραλλαγές αυτές διαφέρουν μεταξύ τους στον τρόπο εκτίμησης της ανισοτροπίας (παραμετρική ή μη παραμετρική εκτίμηση) καθώς και στην εφαρμογή ή μη μετασχηματισμού των δεδομένων σε ισοτροπικά. Τα μοντέλα συνδιασποράς που μελετώνται είναι τα: α) Γενικευμένο Εκθετικό, β) Γκαου-

σιανό, γ) Σφαιρικό, δ) **Mátern** και ε) Σπαρτιάτικο. Οι παραλλαγές για την εκτίμηση των παραμέτρων, οι οποίες έχουν ονομαστεί αυθαίρετα, είναι οι εξής:

- **DirVar0:** Δεν επιχειρείται εκτίμηση των παραμέτρων της ανισοτροπίας εκ των προτέρων. Ως εκ τούτου χρησιμοποιούνται σε όλα τα στάδια της ανάλυσης ανισοτροπικές συναρτήσεις βαριογραμμάτων, οι οποίες περιέχουν τον μέγιστο αριθμό αγνώστων παραμέτρων.
- **DirVar1:** Αρχικά, εκτιμώνται οι παράμετροι της ανισοτροπίας για καθένα απο τα εξεταζόμενα μοντέλα συνδιασποράς με την χρήση κατευθυντικών βαριογραμμάτων. Εν συνεχεία, οι παράμετροι αυτές εισάγονται στην αντίστοιχη ανισοτροπική συνάρτηση βαριογράμματος, ελαττώνοντας τον αριθμό των αγνώστων παραμέτρων κατά τρεις για κάθε μοντέλο. Οι νέες ανισοτροπικές συναρτήσεις βαριογραμμάτων (με λιγότερους βαθμούς ελευθερίας) χρησιμοποιούνται για την εκτίμηση των υπόλοιπων παραμέτρων.
- **DirVar2:** Αρχικά, εκτιμώνται οι παράμετροι της ανισοτροπίας για καθένα απο τα εξεταζόμενα μοντέλα συνδιασποράς με την χρήση κατευθυντικών βαριογραμμάτων. Στη συνέχεια, οι παράμετροι αυτές χρησιμοποιούνται για τον μετασχηματισμό του αρχικού ανισοτροπικού συστήματος συντεταγμένων σε ένα νέο ισοτροπικό σύστημα. Αυτό γίνεται για καθένα απο τα πέντε μοντέλα συνδιασποράς. Μετά τον μετασχηματισμό του συστήματος συντεταγμένων εκτιμώνται οι παράμετροι των αντίστοιχων ισοτροπικών συναρτήσεων συνδιασποράς.
- **CHI1:** Αρχικά, εκτιμώνται οι παράμετροι της ανισοτροπίας για καθένα απο τα εξεταζόμενα μοντέλα συνδιασποράς με την χρήση της Εσσιανής ταυτότητας της συνδιασποράς. Εν συνεχεία, οι παράμετροι αυτές εισάγονται στην αντίστοιχη ανισοτροπική συνάρτηση βαριογράμματος, ελαττώνοντας τον αριθμό των αγνώστων παραμέτρων κατά τρεις για κάθε μοντέλο. Οι νέες ανισοτροπικές συναρτήσεις βαριογραμμάτων (με λιγότερους βαθμούς ελευθερίας) χρησιμοποιούνται για την εκτίμηση των υπόλοιπων παραμέτρων.
- **CHI2:** Αρχικά, εκτιμώνται οι παράμετροι της ανισοτροπίας για καθένα απο τα εξεταζόμενα μοντέλα συνδιασποράς με την χρήση της Εσσιανής ταυτότητας της συνδιασποράς. Στη συνέχεια, οι παράμετροι αυτές χρησιμοποιούνται για τον μετασχηματισμό του αρχικού ανισοτροπικού συστήματος συντεταγμένων σε ένα νέο ισοτροπικό σύστημα. Αυτό γίνεται για καθένα απο τα πέντε μοντέλα συνδιασποράς. Μετά τον μετασχηματισμό του συστήματος συντεταγμένων εκτιμώνται οι παράμετροι των αντίστοιχων ισοτροπικών συναρτήσεων συνδιασποράς.

Η επιλογή του βέλτιστου μοντέλου γίνεται με την εφαρμογή διασταυρωτικής επιβεβαίωσης (LOOCV). Τα αποτελέσματα των διαφόρων μοντέλων αξιολογούνται και συγκρίνο-

νται μέσω κατάλληλων στατιστικών μέτρων (π.χ. μέσο τετραγωνικό σφάλμα, μέγιστο σφάλμα ανακατασκευής, συντελεστής συσχέτισης **Pearson**, συντελεστής συσχέτισης **Spearman**). Η τελική επιλογή του μοντέλου γίνεται βάσει ενός συντελεστή, ο οποίος υπολογίζεται για κάθε εξεταζόμενο μοντέλο και συνδυάζει τις τιμές του μέσου τετραγωνικού σφάλματος και των συντελεστών συσχέτισης **Pearson** και **Spearman**.

Τα αποτελέσματα της παρούσας διπλωματικής εργασίας υποδεικνύουν ότι τα γεωστατιστικά εργαλεία είναι ιδιαίτερα χρήσιμα σε περιπτώσεις όπου τα διαθέσιμα δεδομένα περιέχουν ικανή πληροφορία για την εκάστοτε μελετώμενη φυσική ιδιότητα. Επομένως, η Γεωστατιστική θα μπορούσε να χρησιμοποιηθεί για την υποβοήθηση των υφιστάμενων γεωφυσικών μεθόδων για την προσομοίωση γεωλογικών δομών.

Τα μέτρα αξιολόγησης, και για τους δύο τύπους δειγματοληψίας, δείχνουν ότι όλες οι μέθοδοι που εφαρμόστηκαν επιτυγχάνουν υψηλή συσχέτιση, της τάξεως του 90-95%, μεταξύ των ανακατασκευασμένων και των πραγματικών δεδομένων. Επιπλέον, από την σύγκριση των μεθόδων προκύπτει ότι το Κανονικό **Kriging** λειτουργεί με μεγαλύτερη αποτελεσματικότητα όταν το αρχικό ανισοτροπικό σύστημα συντεταγμένων μετασχηματίζεται σε ισοτροπικό. Όσον αφορά τα μοντέλα συνδιασποράς που εξετάστηκαν, από τα αποτελέσματα δεν υποδεικνύεται η υπεροχή κάποιου εξ αυτών στην ικανότητα περιγραφής των χωρικών συσχετίσεων της υπό μελέτη ιδιότητας έναντι των υπολοίπων. Τέλος, η μέθοδος προσομοίωσης **DGC** επιτυγχάνει συγκρίσιμα, αλλά ελαφρώς κατώτερα, αποτελέσματα (της τάξεως του 89%) από αυτά του Κανονικού **Kriging**. Αυτό θεωρούμε ότι οφείλεται στον περιορισμό της τελευταίας μεθόδου να διακρίνει μόνο τοπικές συσχετίσεις σε αντίθεση με το **Kriging**, το οποίο λαμβάνει υπόψη του συσχετίσεις μεγαλύτερης ακτίνας.

Contents

Abstract	v
Περίληψη	vii
Εκτεταμένη Περίληψη	ix
List of Figures	xvii
List of Tables	xxv
Nomenclature	xxx
Introduction	1
1 Spatial Interpolation and Geostatistics	3
2 Random Fields	5
2.1 Random Field	5
2.2 Probability Functions	6
2.2.1 1D Probability Functions	6
2.2.2 2D Probability Functions	6
2.3 Statistical Moments	7
2.3.1 1D Statistical Moments	8
2.3.2 2D Statistical Moments	10
2.4 Stationarity, Ergodicity and Isotropy	12
2.4.1 Stationarity	12
2.4.2 Ergodicity	12
2.4.3 Isotropy & Anisotropy	13
2.5 Permissible Covariance Functions	17
2.6 Experimental Covariance and Variogram	25

3	Kriging	27
3.1	Basic Notions of Kriging	27
3.1.1	Error Variance	28
3.2	Methods of Kriging	28
3.2.1	Simple Kriging	29
3.2.2	Ordinary Kriging	30
4	Stochastic Simulations	33
4.1	Turning-Bands Method	33
4.2	Directional Gradient-Curvature method	34
5	Data Analysis	37
5.1	Preliminary Analysis	37
5.2	Parameter Inference	40
5.2.1	Common Methods of Parameter Inference	41
5.2.2	Estimation of Anisotropy	43
5.3	Model Selection	44
5.3.1	Common Types of Cross-Validation	45
5.3.2	Measures of Cross-Validation Performance	45
5.4	Spatial Prediction	47
6	Case Study: Marmousi Model	49
6.1	Description of the Data	49
6.1.1	Geologic Profile of the Data	49
6.1.2	Basic Notions and Assumptions for the Data	51
6.2	Methodology	53
6.3	Synthetic Data Test	56
6.3.1	Preliminary Analysis	56
6.3.2	Calculation of Experimental Variogram	59
6.3.3	DirVar0	59
6.3.4	DirVar1	65
6.3.5	Synopsis	71
6.4	Regular Sample of Marmousi Model	71
6.4.1	Ordinary Kriging	71
6.4.2	DGC Simulation	114
6.4.3	Synopsis	118
6.5	Random Sample of Marmousi Model	120

6.5.1	Ordinary Kriging	120
6.5.2	DGC Simulation	161
6.5.3	Synopsis	165
6.6	Problems Encountered	167
7	Conclusions	169
7.1	Conclusions	169
7.2	Future Studies	170
	Bibliography	173
	Appendix A Figures	179
A.1	Synthetic Dataset	180
A.1.1	DirVar0	180
A.1.2	DirVar1	182
A.2	Regular Sample	184
A.2.1	DirVar0	184
A.2.2	DirVar1	186
A.2.3	CHI1	188
A.3	Random Sample	190
A.3.1	DirVar0	190
A.3.2	DirVar1	192
A.3.3	CHI1	194
	Appendix B Codes in MATLAB environment	197
B.1	Synthetic Dataset	197
B.2	Regular Sample	215
B.3	Random Sample	266

List of Figures

1.1	Graphical interpretation of Spatial Interpolation problems	4
2.1	Several sets of (x, y) points, with the Pearson correlation coefficient of x and y for each set [https://en.wikipedia.org/wiki/Correlation_and_dependence] .	11
2.2	Inverse transformation of anisotropic coordinations system	15
2.3	Permissible Covariance Models: a) Generalized Exponential; b) Gaussian; c) Spherical; d) Cardinal Sine; e) Matérn; f) Spartan (3D). The used parameters are $\sigma_z^2 = 1$, $\xi = 1$ and $\eta_0 = 10$ (h is the dimensionless distance $h = r/\xi$ where ξ the correlation length).	22
2.4	Variogram Models: a) Generalized Exponential; b) Gaussian; c) Spherical; d) Cardinal Sine; e) Matérn; f) Spartan (3D). The used parameters are $\sigma_z^2 = 1$, $\xi = 1$ and $\eta_0 = 10$ (h is the dimensionless distance $h = r/\xi$ where ξ the correlation length).	23
2.5	Summation of Permissible Model with Nugget Effect: a) Covariance; b) Variogram.	24
2.6	Space separation into cyclical sections	26
6.1	Profile through the North Quenguela trough in the Cuanza basin (Angola) after Verrier and Branco (1972)	50
6.2	Normalised Marmousi Model (Velocity Model)	52
6.3	Samples of original data	53
6.4	Synthetic RF and random sample	56
6.5	Histograms of original and sample datasets	57
6.6	Normal probability plot of random sample	58
6.7	Directional experimental variograms of the synthetic data. For both cases the angular tolerance is 20° , the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 16, and is divided into 45 distance lags.	60

6.8	Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Gen. Mátern with parameters $\sigma_z^2 = 4.945, \xi_1 = 1.694, R = 0.429, \phi = -65.2^\circ, c_0 = 0.260, \nu = 1.525$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 16. . . .	62
6.9	Original and Estimation of the field. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 4.945, \xi_1 = 1.694, R = 0.429, \phi = -65.2^\circ, c_0 = 0.260, \nu = 1.525$	63
6.10	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 4.945, \xi_1 = 1.694, R = 0.429, \phi = -65.2^\circ, c_0 = 0.260, \nu = 1.525$. . .	64
6.11	Uncertainty of the total field's estimation	65
6.12	Fitting of ellipses to the pairs (ϕ, ξ) of the examined variogram models . .	67
6.13	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Mátern with parameters $\sigma_z^2 = 4.066, \xi_1 = 1.846, R = 0.432, \phi = -42.3^\circ, c_0 = 0.418, \nu = 2.832$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	69
6.14	Original and Estimation of the stochastic component of the field. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 4.066, \xi_1 = 1.846, R = 0.432, \phi = -42.3^\circ, c_0 = 0.418, \nu = 2.832$	70
6.15	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 4.066, \xi_1 = 1.846, R = 0.432, \phi = -42.3^\circ, c_0 = 0.418, \nu = 2.832$	70
6.16	Uncertainty of the total field's estimation	71
6.17	Histograms of original and sample datasets	72
6.18	Normal probability plot of regular sample	73
6.19	Histogram and normal probability plot of the transformed regular sample. The transformation applied is the Box-Cox with $\lambda = -0.2606$	73
6.20	Normal probability plots of the fluctuations resulting from the estimated trend models.	75
6.21	Multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points. The trend equation is given by $m_z(\mathbf{s}) = -0.3044 + 5.1258 \cdot 10^{-4}x - 0.0098y$	76
6.22	Histogram of transformed and detrended sample dataset	77

6.23	Transformed and detrended Marmousi model	77
6.24	Directional experimental variograms of the field. For both cases the angular tolerance is 20° , the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 80, and is divided into 45 distance lags.	79
6.25	Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.026, \xi_1 = 7.657, R = 0.109, \phi = -84.4^\circ, c_0 = 0.008$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	82
6.26	Original and Estimation of the stochastic component of the field. The model used is the Spherical with parameters $\sigma_z^2 = 0.026, \xi_1 = 7.657, R = 0.109, \phi = -84.4^\circ, c_0 = 0.008$	83
6.27	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is the Spherical with parameters $\sigma_z^2 = 0.026, \xi_1 = 7.657, R = 0.109, \phi = -84.4^\circ, c_0 = 0.008$	83
6.28	Original and estimated total field	84
6.29	Histograms and Scatter plot of original and estimated values (total field)	84
6.30	Uncertainty of the total field's estimation	85
6.31	Fitting of ellipses to the pairs (ϕ, ξ) of the investigated variogram models	88
6.32	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.832, \xi_1 = 40.338, R = 0.307, \phi = -50.9^\circ, c_0 = 0.000, \eta_1 = 1.928$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	90
6.33	Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 0.832, \xi_1 = 40.338, R = 0.307, \phi = -50.9^\circ, c_0 = 0.000, \eta_1 = 1.928$	91
6.34	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 0.832, \xi_1 = 40.338, R = 0.307, \phi = -50.9^\circ, c_0 = 0.000, \eta_1 = 1.928$	91
6.35	Original and estimated total field	92
6.36	Histograms and Scatter plot of original and estimated values (total field)	92
6.37	Uncertainty of the total field's estimation	93
6.38	Experimental directional variograms of the new coordinations systems	96

6.39	Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.	97
6.40	Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.033, \xi = 1.031, c_0 = 0.000, \nu = 0.948$	98
6.41	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.033, \xi = 1.031, c_0 = 0.000, \nu = 0.948$	99
6.42	Original and estimated total field	99
6.43	Histograms and Scatter plot of original and estimated values (total field) . .	100
6.44	Uncertainty of the total field's estimation	100
6.45	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.033, \xi_1 = 79.760.338, R = 0.108, \phi = -88.5^\circ, c_0 = 0.000$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	104
6.46	Original and Estimation of the stochastic component of the field. The model used is a Spherical with parameters $\sigma_z^2 = 0.033, \xi_1 = 79.760.338, R = 0.108, \phi = -88.5^\circ, c_0 = 0.000$	105
6.47	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spherical with parameters $\sigma_z^2 = 0.033, \xi_1 = 79.760.338, R = 0.108, \phi = -88.5^\circ, c_0 = 0.000$	105
6.48	Original and estimated total field	106
6.49	Histograms and Scatter plot of original and estimated values (total field) . .	106
6.50	Uncertainty of the total field's estimation	107
6.51	Experimental directional variograms of the new coordinations system . . .	109
6.52	Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.	110
6.53	Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 9.687, \xi = 328.725, c_0 = 0.001, \eta_1 = 467.369$	111
6.54	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 9.687, \xi = 328.725, c_0 = 0.001, \eta_1 = 467.369$	112
6.55	Original and estimated total field	112
6.56	Histograms and Scatter plot of original and estimated values (total field) . .	113

6.57	Uncertainty of the total field's estimation	113
6.58	Results of DGC simulation with 4 classes	115
6.59	Results of DGC simulation with 16 classes	116
6.60	Results of DGC simulation with 100 classes	117
6.61	Histograms of original and sample datasets	121
6.62	Normal probability plot of random sample	121
6.63	Histogram and normal probability plot of the transformed random sample. The transformation applied is the Box-Cox with $\lambda = -0.2944$	122
6.64	Normal probability plots of the fluctuations resulting from the estimated trend models.	123
6.65	Multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points. The trend equation is given by $m_z(\mathbf{s}) =$ $-0.3023 + 5.2017 \cdot 10^{-4}x - 0.0101y$	124
6.66	Histogram of transformed and detrended sample dataset	125
6.67	Transformed and detrended Marmousi model	125
6.68	Directional experimental variograms of the field. For both cases the angular tolerance is 20° , the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 80, and is divided into 45 distance lags.	127
6.69	Fitting of the best theoretical model to the directional experimental vari- ograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.035, \xi_1 = 9.671, R = 0.370, \phi = -83.4^\circ, c_0 = 0.000, \nu = 0.734$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. . . .	130
6.70	Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.035, \xi_1 = 9.671, R =$ $0.370, \phi = -83.4^\circ, c_0 = 0.000, \nu = 0.734$	131
6.71	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.035, \xi_1 = 9.671, R = 0.370, \phi = -83.4^\circ, c_0 = 0.000, \nu = 0.734$. . .	131
6.72	Original and estimated total field	132
6.73	Histograms and Scatter plot of original and estimated values (total field) . .	132
6.74	Uncertainty of the total field's estimation	133
6.75	Fitting of ellipses to the pairs (ϕ, ξ) of the investigated variogram models .	135

6.76	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.010, \xi_1 = 52.606, R = 0.522, \phi = -90.0^\circ, c_0 = 0.008, \eta_1 = -1.999$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. . . .	138
6.77	Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 0.010, \xi_1 = 52.606, R = 0.522, \phi = -90.0^\circ, c_0 = 0.008, \eta_1 = -1.999$	139
6.78	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 0.010, \xi_1 = 52.606, R = 0.522, \phi = -90.0^\circ, c_0 = 0.008, \eta_1 = -1.999$	139
6.79	Original and estimated total field	140
6.80	Histograms and Scatter plot of original and estimated values (total field) . .	140
6.81	Uncertainty of the total field's estimation	141
6.82	Experimental directional variograms of the new coordinations systems. . . .	143
6.83	Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.	144
6.84	Original and Estimation of the stochastic component of the field. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 0.033, \xi = 1.780, c_0 = 0.001, \nu = 0.509$	145
6.85	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 0.033, \xi = 1.780, c_0 = 0.001, \nu = 0.509$	146
6.86	Original and estimated total field	146
6.87	Histograms and Scatter plot of original and estimated values (total field) . .	147
6.88	Uncertainty of the total field's estimation	147
6.89	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.038, \xi_1 = 0.021.338, R = 0.208, \phi = -89.1^\circ, c_0 = 0.000, \nu = 0.115$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	151
6.90	Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.038, \xi_1 = 0.021.338, R = 0.208, \phi = -89.1^\circ, c_0 = 0.000, \nu = 0.115$	152

6.91	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.038, \xi_1 = 0.021.338, R = 0.208, \phi = -89.1^\circ, c_0 = 0.000, \nu = 0.115$.	152
6.92	Original and estimated total field	153
6.93	Histograms and Scatter plot of original and estimated values (total field) . .	153
6.94	Uncertainty of the total field's estimation	154
6.95	Experimental directional variograms of the new coordinations system . . .	155
6.96	Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.	157
6.97	Original and Estimation of the stochastic component of the field. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 0.033, \xi = 13.192, c_0 = 0.000, \nu = 0.554$	158
6.98	Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 0.033, \xi = 13.192, c_0 = 0.000, \nu = 0.554$	159
6.99	Original and estimated total field	159
6.100	Histograms and Scatter plot of original and estimated values (total field) . .	160
6.101	Uncertainty of the total field's estimation	160
6.102	Results of DGC simulation with 4 classes	162
6.103	Results of DGC simulation with 16 classes	163
6.104	Results of DGC simulation with 100 classes	164
A.1	Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Gen. Mátern with parameters $\sigma_z^2 = 4.945, \xi_1 = 1.694, R = 0.429, \phi = -65.2^\circ, c_0 = 0.260, \nu = 1.525$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 16. . . .	181
A.2	Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Mátern with parameters $\sigma_z^2 = 4.066, \xi_1 = 1.846, R = 0.432, \phi = -42.3^\circ, c_0 = 0.418, \nu = 2.832$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.	183

- A.3 Fitting of the best theoretical model to the directional experimental (semi-) variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.026$, $\xi_1 = 7.657$, $R = 0.109$, $\phi = -84.4^\circ$, $c_0 = 0.008$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. 185
- A.4 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.832$, $\xi_1 = 40.338$, $R = 0.307$, $\phi = -50.9^\circ$, $c_0 = 0.000$, $\eta_1 = 1.928$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. 187
- A.5 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.033$, $\xi_1 = 79.760$, $R = 0.108$, $\phi = -88.5^\circ$, $c_0 = 0.000$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. . . . 189
- A.6 Fitting of the best theoretical model to the directional experimental (semi-) variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.035$, $\xi_1 = 9.671$, $R = 0.370$, $\phi = -83.4^\circ$, $c_0 = 0.000$, $\nu = 0.734$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. 191
- A.7 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.010$, $\xi_1 = 52.606$, $R = 0.522$, $\phi = -90.0^\circ$, $c_0 = 0.008$, $\eta_1 = -1.999$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. 193
- A.8 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.038$, $\xi_1 = 0.021$, $R = 0.208$, $\phi = -89.1^\circ$, $c_0 = 0.000$, $\nu = 0.115$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80. 195

List of Tables

2.1	Common used probability density functions	7
2.2	Permissible Covariance Functions. $K_v(\cdot)$ denote the modified Bessel functions of the second kind of order v , $\Gamma(\cdot)$ denote the Gamma function, h denotes the dimensionless distance, $\Delta = \eta_1^2 - 4 ^{1/2}$, $\beta_{1,2} = 2 \mp \eta_1 ^{1/2}/2$, and $\omega_{1,2} = (\eta_1 \mp \Delta /2)^{1/2}$	18
2.3	Special cases of Matérn model for different values of v	20
2.4	Permissible Variogram Functions. $K_v(\cdot)$ denote the modified Bessel functions of the second kind of order v , $\Gamma(\cdot)$ denote the Gamma function, h denotes the dimensionless distance, $\Delta = \eta_1^2 - 4 ^{1/2}$, $\beta_{1,2} = 2 \mp \eta_1 ^{1/2}/2$, and $\omega_{1,2} = (\eta_1 \mp \Delta /2)^{1/2}$	21
5.1	Common trend models	39
6.1	Description of the five variations of Parameter Inference step. DVF: Directional Variogram Fitting, CHI: Covariance Hessian Identity (P.I.: Parameter Inference).	54
6.2	Original and sample datasets statistics	57
6.3	Initial values and boundaries of the parameters for optimization step, where $\hat{g}_{max}(\equiv \text{maximum value of experimental variogram}) = 5.7026$, $d_{max} \simeq 16$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}1/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 = 1000, \xi_1 = d_{max}1/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in [0, d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$	61
6.4	Optimum Parameters of the examined variogram models	61
6.5	Leave-One-Out Cross Validation Scores	63
6.6	Ordinary Kriging Scores	64
6.7	Anisotropy parameters of the examined variogram models estimated with DVF method	66

6.8	Optimum Parameters of the variogram models with the lower degrees of freedom	66
6.9	Leave-One-Out Cross Validation Scores	68
6.10	Ordinary Kriging Scores	70
6.11	Original and sample datasets statistics	72
6.12	Transformed (Box-Cox with $\lambda = -0.2606$) dataset statistics	72
6.13	Estimated trend models (regular sample)	74
6.14	Transformed and detrended dataset statistics	76
6.15	Initial values and boundaries of the parameters for optimization step, where $\hat{g}_{max}(\equiv \text{maximum value of experimental variogram}) = 0.0384$, $d_{max} \simeq 80$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 = 1000, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$	80
6.16	Optimum Parameters of the investigated variogram models	80
6.17	Leave-One-Out Cross Validation Scores (see section 5.3.2	81
6.18	Ordinary Kriging Scores	83
6.19	Total Field Estimation Scores	85
6.20	Classification Measures	86
6.21	Anisotropy parameters of the investigated variogram models estimated with DVF method	87
6.22	Optimum Parameters of the variogram models with the lower degrees of freedom	87
6.23	Leave-One-Out Cross Validation Scores (see section 5.3.2	89
6.24	Ordinary Kriging Scores	89
6.25	Total Field Estimation Scores	93
6.26	Classification Measures	93
6.27	Anisotropy parameters of the new coordinations systems	95
6.28	Optimum Parameters of the isotropic variogram models	95
6.29	Leave-One-Out Cross Validation Scores (see section 5.3.2	95
6.30	Ordinary Kriging Scores	98
6.31	Total Field Estimation Scores	99
6.32	Classification Measures	101
6.33	Anisotropy parameters of the investigated variogram models estimated with CHI method	101

6.34 Optimum Parameters of the variogram models with the lower degrees of freedom	102
6.35 Leave-One-Out Cross Validation Scores (see section 5.3.2	102
6.36 Ordinary Kriging Scores	103
6.37 Total Field Estimation Scores	105
6.38 Classification Measures	106
6.39 Anisotropy parameters of the new coordinations systems	108
6.40 Optimum Parameters of the isotropic variogram models	108
6.41 Leave-One-Out Cross Validation Scores (see section 5.3.2	111
6.42 Ordinary Kriging Scores	112
6.43 Total Field Estimation Scores	113
6.44 Classification Measures	114
6.45 DGC Validation Measures	114
6.46 Summarized results for the regular sample	119
6.47 Original and sample datasets statistics	120
6.48 Transformed (Box-Cox with $\lambda = -0.2606$) dataset statistics	121
6.49 Estimated trend models (random sample)	122
6.50 Transformed and detrended dataset statistics	124
6.51 Initial values and boundaries of the parameters for optimization step, where $\hat{g}_{max}(\equiv \text{maximum value of experimental variogram}) = 0.0384$, $d_{max} \simeq 80$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 =$ $1000, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in$ $[0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in$ $[0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$	128
6.52 Optimum Parameters of the investigated variogram models	128
6.53 Leave-One-Out Cross Validation Scores (see section 5.3.2	129
6.54 Ordinary Kriging Scores	129
6.55 Total Field Estimation Scores	133
6.56 Classification Measures	134
6.57 Anisotropy parameters of the investigated variogram models estimated with DVF method	136
6.58 Optimum Parameters of the variogram models with the lower degrees of freedom	136
6.59 Leave-One-Out Cross Validation Scores (see section 5.3.2	136
6.60 Ordinary Kriging Scores	137
6.61 Total Field Estimation Scores	139

6.62	Classification Measures	140
6.63	Anisotropy parameters of the new coordinations systems	142
6.64	Optimum Parameters of the isotropic variogram models	142
6.65	Leave-One-Out Cross Validation Scores (see section 5.3.2	142
6.66	Ordinary Kriging Scores	145
6.67	Total Field Estimation Scores	146
6.68	Classification Measures	148
6.69	Anisotropy parameters of the investigated variogram models estimated with CHI method	148
6.70	Optimum Parameters of the variogram models with the lower degrees of freedom	149
6.71	Leave-One-Out Cross Validation Scores (see section 5.3.2	149
6.72	Ordinary Kriging Scores	150
6.73	Total Field Estimation Scores	150
6.74	Classification Measures	154
6.75	Anisotropy parameters of the new coordinations systems	155
6.76	Optimum Parameters of the isotropic variogram models	156
6.77	Leave-One-Out Cross Validation Scores (see section 5.3.2	158
6.78	Ordinary Kriging Scores	159
6.79	Total Field Estimation Scores	160
6.80	Classification Measures	161
6.81	DGC Validation Measures	161
6.82	Summarized results for the random sample	166
7.1	Summarized scores for the regular sample	170
7.2	Summarized scores for the random sample	170

Nomenclature

Roman Symbols

A Subset domain of Ω

$c_z(\mathbf{s}, \mathbf{s}')$ Covariance of the random field Z between the points $(\mathbf{s}, \mathbf{s}')$

D Subset domain of \mathbb{R}^n

$\mathbb{E}[\cdot]$ Expectation operator

$F_z(z_s; \mathbf{s})$ Cumulative probability function of the *event* z_s of the random field Z

$f_z(z_s; \mathbf{s})$ Probability density function of the *event* z_s of the random field Z

$\gamma_z(\mathbf{s}, \mathbf{s}')$ Variogram of the random field Z between the points $(\mathbf{s}, \mathbf{s}')$

$F_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}')$ Joint cumulative distribution function of the random field Z at the two points $(\mathbf{s}, \mathbf{s}')$

$f_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}')$ Joint probability density function of the random field Z at the two points $(\mathbf{s}, \mathbf{s}')$

$m_z^n(\mathbf{s})$ Raw statistical moment of order n of the random field Z at the point \mathbf{s}

$m_z^n(\mathbf{s}, \mathbf{s}')$ Raw 2D statistical moment of order n of the random field Z between the points $(\mathbf{s}, \mathbf{s}')$

$m_{z,c}^n(\mathbf{s})$ Central statistical moment of order n of the random field Z at the point \mathbf{s}

$m_{z,c}^n(\mathbf{s}, \mathbf{s}')$ Central 2D statistical moment of order n of the random field Z between the points $(\mathbf{s}, \mathbf{s}')$

P Probability function on A

$\rho_z(\mathbf{s}, \mathbf{s}')$ Correlation of the random field Z between the points $(\mathbf{s}, \mathbf{s}')$

\mathbf{s}, \mathbf{u}	Position vectors of points belonging in D
$Z(\cdot)$	Scalar spatial random field
$z(\cdot)$	Single realization of the scalar spatial random field Z
$\hat{Z}(\cdot)$	Estimate of the random variable $Z(\cdot)$
z_j	Single realization of the scalar spatial random field Z at the j -th point \mathbf{s}_j of the domain D , i.e. $z_j \equiv z(\mathbf{s}_j)$
z_s	Single realization of the scalar spatial random field Z at the point \mathbf{s} , i.e. $z_s \equiv z(\mathbf{s})$

Greek Symbols

Ω	Sample space of a probability field
ω	State of the sample space of a probability field

Introduction

Geostatistics, since its empirical development by Matheron and Krige in the early 1960s (Cressie, 1990; Krige, 1951; Matheron, 1971) and the subsequent theoretical enhancement, was widely adopted by diverse scientific and engineering fields such as mining (Galetakis, 1998; Goovaerts, 1997; Pavlides et al., 2015) and petroleum engineering (Deutsch, 2002; Kelkar and Perez, 2002), hydrogeology (Mariethoz and Renard, 2010; Papadopoulou et al., 2009; Varouchakis and Hristopulos, 2013), meteorology (Agou, 2016; Bargaoui and Chebbi, 2009; Liu et al., 2010), atmospheric science (Žukovič and Hristopulos, 2013a), remote sensing (Žukovič and Hristopulos, 2013b), environmental and earth sciences (Christakos, 1992; Varouchakis et al., 2015), and material science (Greene, 1992). The widespread utility and success of the tools provided by Geostatistics is due to the growing demand for efficient analysis of the increasing amount of data being gathered by the sophisticated acquisition methods of the last decades.

A common problem in Geosciences is that of geological media modelling, i.e. the mapping of properties of the geological stratas' (e.g. mineral or pollutant concentration, porosity, permeability, seismic waves velocity, resistivity). The graphical representation of these properties' spatial, temporal or spatiotemporal distribution can improve the understanding of the environmental parameters affecting the geological systems of interest and the decision-making for the scientists and engineers.

In this thesis we present an application of geostatistical methods in a problem of geological media modelling. Random and regular samples of the Marmousi Model (Bourgeois et al., 1990; Versteeg and Grau, 1990), a synthetic, two dimensional, acoustic model released at the workshop of 52nd EAEG meeting in 1990, are used as data. For the reconstruction of the original dataset, two methods are employed: i) Kriging estimation and ii) Directional Gradient Curvature (DGC) simulation method (Žukovič and Hristopulos, 2013a). Several methods regarding the parameter inference and anisotropy estimation of the data (i.e. parametric - non parametric anisotropy parameter inference) and the transformation of the data to isotropic or not, are investigated in the case of OK. To evaluate the performance of the

OK and the DGC simulation, proper validation measures are calculated for each method and compared.

This thesis is structured as follows: In the first chapter a brief definition of Geostatistics is given. The second chapter comprises basic notions of random fields, which constitute the primary tool of Geostatistics. The third and fourth chapters present the kriging and the simulation methods, respectively. The fifth chapter describes the commonly followed steps of data analysis. The sixth focuses on the case study of this thesis, concerning the application of geostatistical methods in the above described geological media modelling. Finally, in the seventh chapter the conclusions of the thesis are presented.

Chapter 1

Spatial Interpolation and Geostatistics

Assume that at two points of a 2D space, $\mathbf{s}_1(x_1, y_1)$ and $\mathbf{s}_2(x_2, y_2)$, the values of a physical variable, $Z(\mathbf{s}_1)$ and $Z(\mathbf{s}_2)$, respectively, are measured and known. Assume a third point in the same space, $\mathbf{u}(x_3, y_3)$, at which the value of the physical variable is unknown, $Z(\mathbf{u}) = ?$. The principle of spatial continuity (also known as the 1st Law of Geography [Tobler \(1970\)](#)) indicates that physical phenomena are not randomly distributed in space but exhibit spatial correlation. As such, the information provided by the known values can be used to produce an estimation for the missing value, i.e. $\hat{Z}(\mathbf{u}) = f(Z(\mathbf{s}_1), Z(\mathbf{s}_2))$. This simple problem, illustrated in Fig. [1.1](#), is frequently encountered in spatial interpolation. It commonly arises due to lack of practicality, accessibility, financing or time (or a combination of them) demanded for an exhaustive sampling or due to malfunctions of the sampling equipment.

In practice the unknown values are usually nodes of a regular, numerical grid which covers the area of interest. The data set which provides the necessary information for the estimation of the missing values comprises hundreds or even thousands of known values. Thus, the estimation of the missing values results in mapping the distribution of the investigated physical variable in the whole area (see Fig. [1.1](#)).

The various methods of spatial interpolation can be divided into two broad categories, the **deterministic** and the **stochastic** methods. The first category includes methods which use mathematical functions to calculate the values at the investigated locations based on the values of neighboring data. Some of these methods are Thiessen polygons (also known as Dirichlet or Voronoi diagrams), Natural Neighbors Method, Distance Weighted Methods, (Fuzzy) k-Nearest Neighbors, Moving Average Methods and Trend Surface Analysis ([Barnett, 1981](#); [Burrough et al., 2013](#)).

The stochastic methods combine mathematical and statistical tools to estimate the unknown values and also assess the uncertainty of these estimations. They can be applied in mathematical modelling of spatial, temporal and spatiotemporal correlated data.

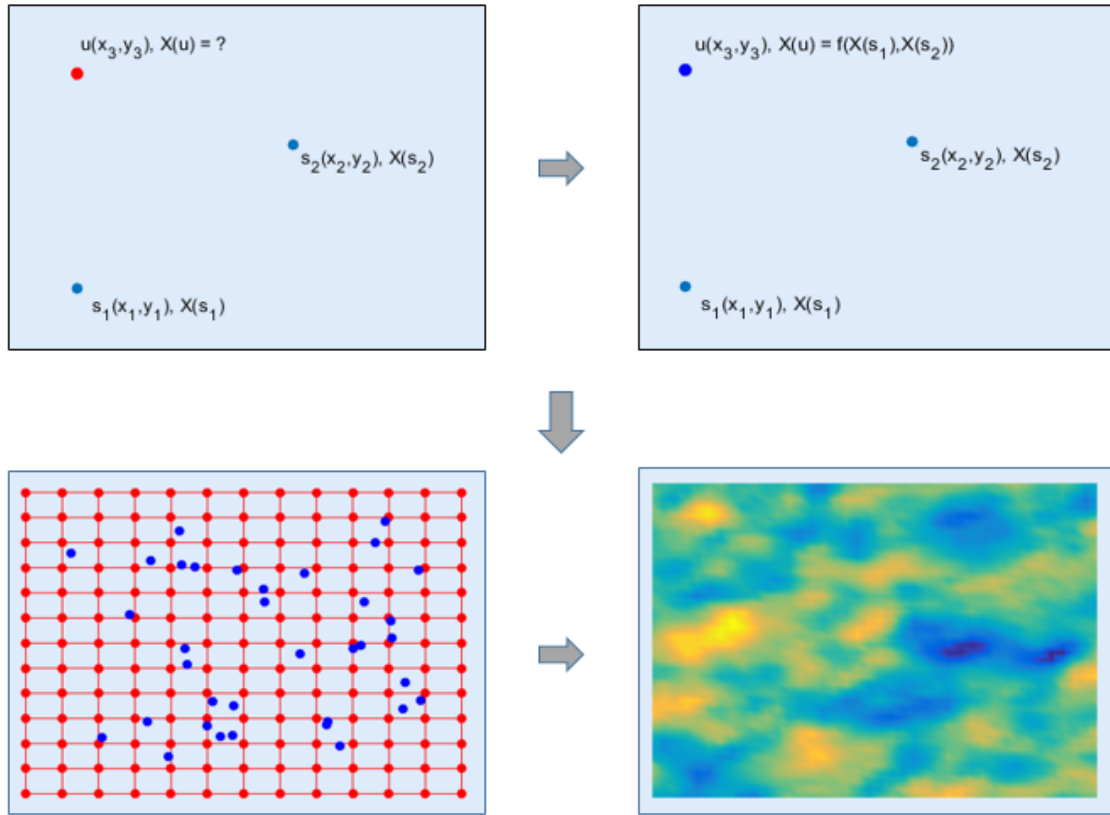


Figure 1.1 Graphical interpretation of Spatial Interpolation problems

The stochastic methods can be further divided into **Estimation processes** and **Simulation processes**. The estimation processes aim to determine the optimal value at the points of interest under one optimization criterion (e.g. minimizing the mean square error or maximizing the likelihood). Simulation processes, on the other hand, are intended to produce many of the possible states (alternative scenarios) of the field which satisfactorily agree with the existing statistical limitations resulting from the training sample (e.g. mean, standard deviation, variogram). Geostatistics belong to the stochastic methods (Lantuéjoul, 2013).

Frequently, the advantages of deterministic methods are the simplicity and ease of implementation, and the small computational cost. Disadvantages include lack of flexibility and uncertainty assessment. The stochastic methods, on the contrary, are more flexible, provide uncertainty assessment and higher accuracy. The limitations of them are related to the high computational cost and the possible big number of parameters needed to be tuned (Chilès and Delfiner, 2012; Lantuéjoul, 2013).

Chapter 2

Random Fields

2.1 Random Field

Random Fields (RFs) comprise the primary tool of Geostatistics. Generally, a random field is a set of random values that describe the distribution of a physical property in an n -dimensional space. Each of these values is attributed to a particular point in the n -dimensional space and at every point of this space the possible values follow a probability distribution function (pdf). The pdf can be the same or differ from one point to the other. The distribution of the random variable throughout the field can be described from the joint probability distribution function (j-pdf), i.e. the j-pdf describes the correlations between different points. As such, the value at one point can be written as a function of the values of other points, especially of the most adjacent ones (principle of spatial continuity). This spatial dependence (correlation) is the special feature that differentiates RFs from totally random (uncorrelated) processes.

A more strictly mathematical definition of random field is the following: "*Given a domain $D \subset \mathbb{R}^n$ (with a positive volume) and a probability space (Ω, A, P) , a random field is a function of two variables $Z(\mathbf{s}, \omega)$ such that for each $\mathbf{s} \in D$ the section $Z(\mathbf{s}, \cdot)$ is a random variable on (Ω, A, P) . Each of the functions $Z(\cdot, \omega)$ defined on D as the section of the RF at $\omega \in \Omega$ is a realization of the RF.*" Herein, for brevity and simplicity, RF is denoted by $Z(\mathbf{s})$, and with the lowercase $z(\mathbf{s})$ is denoted a single realization of the RF at the point \mathbf{s} (Chilès and Delfiner, 2012).

A probability space (also known as probability field) is a mathematical triplet (Ω, A, P) consisting of:

- the sample space (or space of elementary events) Ω , which is a non-empty set of all possible events of a model
- the σ -algebra A , which is a set of subsets of Ω , called events, and

- the probability measure $P : A \rightarrow [0, 1]$, which is a function on A , satisfying the condition $P(\Omega) = 1$ (Kolmogorov, 1960).

Depending on their properties the Random Fields can be classified into various categories. Markov random fields (MRF) (Kindermann, 1980; Rozanov, 1982), and Gaussian random fields (GaRF) (Adler and Taylor, 2007; Khoshnevisan, 2002) represent some of the major amongst these categories.

2.2 Probability Functions

2.2.1 1D Probability Functions

The one-dimensional (point) *probability density function (pdf)* at a point \mathbf{s} of a random field, denoted as $f_z(z_s; \mathbf{s})$, describes the possible situations of the field at that specific point. In other words, the pdf gives the interval probability of the *event* z_s

$$f_z(z_s; \mathbf{s}) = P(z_s) = P(z_1 \leq z_s \leq z_2). \quad (2.1)$$

Similarly can be defined the *cumulative distribution function (cdf)* at the point \mathbf{s} of the random field, $F_z(z_s; \mathbf{s})$, as

$$F_z(z_s; \mathbf{s}) = P(z_s \leq z_1). \quad (2.2)$$

From Eq. (2.1) and (2.2), if $F_z(z_s; \mathbf{s})$ is differentiable in z_s , follows that

$$f_z(z_s; \mathbf{s}) = F_z(z_2; \mathbf{s}_2) - F_z(z_1; \mathbf{s}_1) = \frac{d}{dx} F_z(z_s; \mathbf{s}). \quad (2.3)$$

Conversely, if $f_z(z_s; \mathbf{s})$ exists, it is valid that

$$F_z(z_s; \mathbf{s}) = \int_{-\infty}^{z_s} f_z(v; \mathbf{s}) dv. \quad (2.4)$$

Some commonly used pdf are shown in Table 2.1.

2.2.2 2D Probability Functions

The two-dimensional probability functions of a random field, i.e. the *joint probability density function (j-pdf)*, $f_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}')$, and the *joint cumulative distribution function (j-cdf)*, $F_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}')$, describe the correlations between the possible situations of the field at two points. Specifically, they give respectively the interval and the cumulative probabilities of

Table 2.1 Common used probability density functions

Model	Probability Density Function
Uniform	$f_z(z_s; \mathbf{s}) = \begin{cases} \frac{1}{(b-a)}, & z_s \in [a, b] \\ 0, & z_s \notin [a, b] \end{cases}$
Exponential	$f_z(z_s; \mathbf{s}) = \begin{cases} b^{-1} \exp(-b^{-1}x), & z_s \geq 0 \\ 0, & z_s < 0 \end{cases}$
Gaussian	$f_z(z_s; \mathbf{s}) = (\sqrt{2\pi}\sigma)^{-1} \exp(-\frac{(x-m)^2}{2\sigma^2})$

the appearance both of the *events* $z_s, z_{s'}$

$$f_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}') = P(z_1 \leq z_s \leq z_2) \cap P(z_3 \leq z_{s'} \leq z_4), \quad (2.5)$$

$$F_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}') = P(z_s \leq z_1) \cap P(z_{s'} \leq z_2). \quad (2.6)$$

Note that Eqs. (2.5) and (2.6) are reciprocally linked

$$f_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}') = \frac{\partial^2}{\partial z_s \partial z_{s'}} F_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}'), \quad (2.7)$$

$$F_z(z_s, z_{s'}; \mathbf{s}, \mathbf{s}') = \int_{-\infty}^{z_s} \int_{-\infty}^{z_{s'}} f_z(v, v'; \mathbf{s}, \mathbf{s}') dv dv'. \quad (2.8)$$

2.3 Statistical Moments

The probability functions describe completely a random field, but they cannot always be easily and accurately defined. Thus, it is more convenient to work with the statistical moments, which contain partial (but sufficient) information about the random field and can be more easily estimated from the data. The Statistical Moments comprise a set of deterministic quantities, which represent average values (expectations), on all the possible states of the RF, of various combinations of the field's values in one or more locations.

In practical applications, the statistical moments of interest are those of low order (up to 4th-order), since the increase in the order implies harder estimation. Low-order moments are **mean**, **variance**, **skewness** and **kurtosis** for one-dimensional pdfs, while for joint pdfs

covariance and **variogram** are the main low-order moments (Casella and Berger, 2001; Kitanidis, 1997; Papoulis and Pillai, 2002; Thijssen, 2016). An analytic description of them is presented in the following sections.

2.3.1 1D Statistical Moments

The *raw statistical moment of order n* for a continuum variable is defined according to the equation

$$m_z^n(\mathbf{s}) = \mathbb{E}[Z^n(\mathbf{s})] = \int_{-\infty}^{\infty} z_s^n f_z(z_s; \mathbf{s}) dz_s, \quad (2.9)$$

where $\mathbb{E}[\cdot]$ denotes the expectation operator.

To estimate the raw statistical moments of a **sample**, the following equation is used

$$\hat{m}_z^n \equiv \overline{z^n} = \frac{1}{M} \sum_{j=1}^M z_j^n, \quad (2.10)$$

where M the size of the sample.

Another class of statistical moments is the central statistical moments, which represent the expectation values of the field's *fluctuations* around the mean value (see 2.3.1), $Z'(\mathbf{s}) = Z(\mathbf{s}) - m_z(\mathbf{s})$, i.e. they provide a measure of *dispersion* around the mean.

The *central statistical moments of order n* for a continuum variable and a sample are given by

$$m_{z,c}^n(\mathbf{s}) = \mathbb{E}[(Z(\mathbf{s}) - m_z(\mathbf{s}))^n] = \mathbb{E}[(Z'(\mathbf{s}))^n] = \int_{-\infty}^{\infty} (z_s - m_z(\mathbf{s}))^n f_z(z_s; \mathbf{s}) dz_s, \quad (2.11)$$

and

$$\hat{m}_{z,c}^n \equiv \overline{z'^n} = \frac{1}{M} \sum_{j=1}^M (z_j - m_z(\mathbf{s}_j))^n, \quad (2.12)$$

respectively.

Similar equations can be derived for the estimation of raw and central statistical moments in the case of discrete variables by replacing the integrals with summations.

Mean Value

The mean value is the 1st order statistical moment

$$m_z(\mathbf{s}) = \mathbb{E}[Z(\mathbf{s})], \quad (2.13)$$

and represents a measure of the central tendency of the probability distribution.

The mean value may not be independent of the position \mathbf{s} due to possible spatial dependence of the one-dimensional pdf. It is common practice to model the $m_z(\mathbf{s})$ with a deterministic trend function and, subtracting it from the field values, to statistically process the fluctuations.

Variance

The variance is the 2st order central statistical moment

$$\sigma_z^2(\mathbf{s}) \equiv m_{z,c}^2(\mathbf{s}) = \mathbb{E}[(Z(\mathbf{s}) - m_z(\mathbf{s}))^2] = \mathbb{E}[(Z'(\mathbf{s}))^2], \quad (2.14)$$

and measures the dispersion from the center of the distribution. Its positive square root is the *standard deviation* σ .

Skewness

The skewness is the standardized 3rd order central statistical moment

$$s_z(\mathbf{s}) = \frac{1}{\sigma_z^3} \mathbb{E}[(Z(\mathbf{s}) - m_z(\mathbf{s}))^3] = \frac{1}{\sigma_z^3} \mathbb{E}[(Z'(\mathbf{s}))^3], \quad (2.15)$$

and measures the assymetry of the distribution. Right skewed (\equiv right tail longer) distributions have positive skewness, while the left skewed ones have negative skewness. A symmetric distribution (i.e. gaussian) has $s_z(\mathbf{s}) = 0$.

Kurtosis

The kurtosis is the standardized 4th order central statistical moment

$$k_z(\mathbf{s}) = \frac{1}{\sigma_z^4} \mathbb{E}[(Z(\mathbf{s}) - m_z(\mathbf{s}))^4] = \frac{1}{\sigma_z^4} \mathbb{E}[(Z'(\mathbf{s}))^4], \quad (2.16)$$

and measures the heaviness of the tail of the distribution, compared to the normal distribution of the same variance. Gaussian distribution has $k_z(\mathbf{s}) = 3$, distributions with narrower peaks, called *leptokurtic*, have $k_z(\mathbf{s}) > 3$, and distributions with wider peaks, called *platykurtic*, have $k_z(\mathbf{s}) < 3$.

2.3.2 2D Statistical Moments

Covariance

The covariance is a 2nd order central 2D statistical moment

$$\begin{aligned} c_z(\mathbf{s}, \mathbf{s}') &= \mathbb{E}[(Z(\mathbf{s}) - m_z(\mathbf{s}))(Z(\mathbf{s}') - m_z(\mathbf{s}'))] \\ &= \mathbb{E}[Z(\mathbf{s})Z(\mathbf{s}')] - m_z(\mathbf{s})m_z(\mathbf{s}') \\ &= \mathbb{E}[Z'(\mathbf{s})Z'(\mathbf{s}')], \end{aligned} \quad (2.17)$$

and describes the quantitative dependence of the field's fluctuations between two different points. Higher absolute values indicate a higher dependence (correlation) and vice versa. Positive covariance implies variables with similar behaviors, i.e. increasing or decreasing of the one variable's values correspond to also increasing or decreasing, respectively, of the other variable's values. In the opposite case, negative covariance implies variables with opposite behaviors, i.e. increasing of the one variable's values correspond to decreasing of the other variable's values.

The covariance is dependent on the distance $\mathbf{r} = \mathbf{s}' - \mathbf{s}$ between the two points \mathbf{s} and \mathbf{s}' . When the distance between the two points is ∞ , the covariance is zero. As the distance decreases the covariance is increasing and when distance tends to zero, i.e. $\mathbf{s} \equiv \mathbf{s}'$, the covariance takes the maximum possible value, which is equal to the variance of the random field at that point

$$c_z(\mathbf{s}, \mathbf{s}) = \sigma_z^2(\mathbf{s}). \quad (2.18)$$

Thus, the Eq. (2.17) can be rewritten as

$$c_z(\mathbf{s}, \mathbf{s} + \mathbf{r}) \equiv c_z(\mathbf{s}, \mathbf{r}) = \mathbb{E}[Z'(\mathbf{s})Z'(\mathbf{s} + \mathbf{r})]. \quad (2.19)$$

The standardization of the covariance by dividing it with the product of standard deviations at the two points (or with the variance if $\sigma_z(\mathbf{s}) = \sigma_z(\mathbf{s}')$) gives a more accurate and meritocratic measure of the spatial dependences, which is called **correlation coefficient**

$$\rho_z(\mathbf{s}, \mathbf{s}') = \frac{c_z(\mathbf{s}, \mathbf{s}')}{\sigma_z(\mathbf{s})\sigma_z(\mathbf{s}')} \in [-1, 1]. \quad (2.20)$$

or, taking into account the dependence of covariance on $\mathbf{r} = \mathbf{s}' - \mathbf{s}$,

$$\rho_z(\mathbf{s}, \mathbf{r}) = \frac{c_z(\mathbf{s}, \mathbf{r})}{\sigma_z(\mathbf{s})\sigma_z(\mathbf{s} + \mathbf{r})} \in [-1, 1]. \quad (2.21)$$

A graphical interpretation of the effect of different data distributions on the correlation coefficient can be seen in Fig. 2.1 where several sets of (x,y) points with estimated correlation coefficient are plotted. It can be noted that the correlation reflects the noisiness and direction of a linear relationship (top row), but not the slope of that relationship (middle), nor many aspects of nonlinear relationships (bottom).

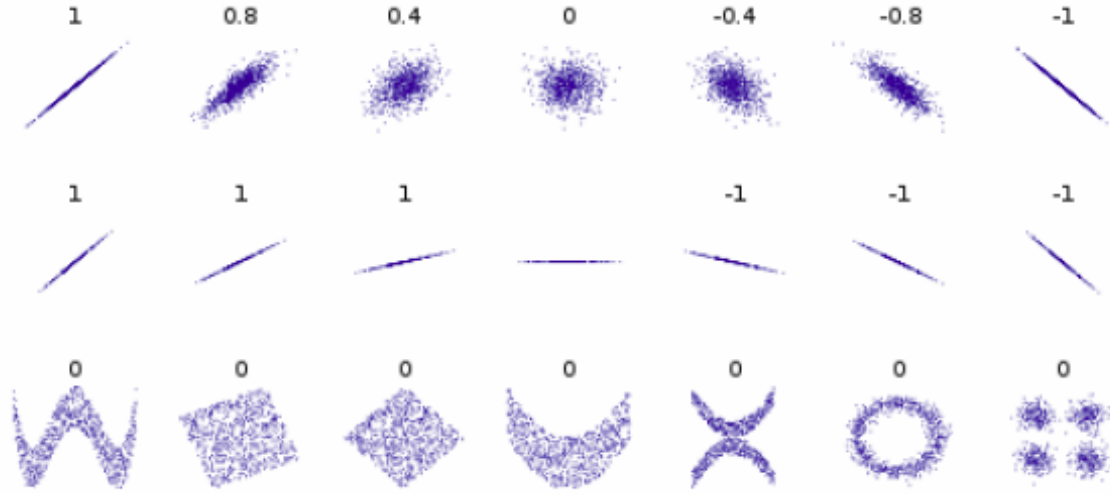


Figure 2.1 Several sets of (x, y) points, with the Pearson correlation coefficient of x and y for each set [https://en.wikipedia.org/wiki/Correlation_and_dependence]

Variogram

The variogram is also a 2nd order central 2D statistical moment

$$\begin{aligned}\gamma_z(\mathbf{s}, \mathbf{s}') &= \frac{1}{2} \mathbb{E}[(Z'(\mathbf{s}) - Z'(\mathbf{s}'))^2] \\ &= \frac{1}{2} \text{Var}[Z'(\mathbf{s}) - Z'(\mathbf{s}')],\end{aligned}\tag{2.22}$$

and like the covariance describes the spatial dependence of the field's fluctuations between two different points.

Similarly to covariance, variogram is dependent on the distance between the two points ($\mathbf{r} = \mathbf{s}' - \mathbf{s}$). When $\|\mathbf{r}\| = 0$, $\gamma_z(\mathbf{s}, \mathbf{s}') = 0$ and as $\|\mathbf{r}\|$ increases, the same does $\gamma_z(\mathbf{s}, \mathbf{s}')$. However, variogram, in contrast with the covariance, does not require the *a priori* knowledge of the mean value of the RF, since by differencing the values of the RF the stochastic trends are removed. Therefore, it holds that

$$\gamma_z(\mathbf{s}, \mathbf{s}') \equiv \gamma_z(\mathbf{r}) = \frac{1}{2} \mathbb{E}[(Z'(\mathbf{s}) - Z'(\mathbf{s} + \mathbf{r}))^2].\tag{2.23}$$

Basic Parameters of the 2D Statistical Moments

In order to define the 2D Statistical Moments (i.e. covariance, correlation and variogram) and describe a random field, the following two parameters have to be determined:

- **covariance** (σ_z^2), which measures the width of the fields variations, and
- **correlation length** (ξ), which normalizes the distance ($\|\mathbf{r}\|$) such that the covariance can be also expressed as function of a dimensionless distance (h) instead of ξ .

2.4 Stationarity, Ergodicity and Isotropy

Except from the statistical moments, there are some properties of the RFs, such as **stationarity, ergodicity and isotropy**, which are of great importance for stochastic analysis.

2.4.1 Stationarity

Stationarity (or Homogeneity) implies that the probabilistic behavior of a stochastic process is independent of the position in space or time or spacetime. In practice, because strict stationarity is very difficult to establish, a weakened version of it is used ([Chilès and Delfiner, 2012](#); [Shumway and Stoffer, 2011](#)).

Definition 2.1. A random field $Z(\mathbf{s})$ is called weakly or 2nd-order stationary, if it is a finite variance process such that

- (i) the mean value is constant and does not depend on position in space,
 $\mathbb{E}[Z(\mathbf{s})] = m_z, \forall \mathbf{s} \in D$, and
- (ii) the covariance, $c_z(\mathbf{s}, \mathbf{s}')$, defined in (2.17) depends only on $\mathbf{r} = \mathbf{s}' - \mathbf{s}$,
 $c_z(\mathbf{s}, \mathbf{s}') = c_z(\mathbf{r}), \forall \mathbf{s}, \mathbf{s}' \in D$.

One important consequence of 2nd-order stationarity of a random field is that the covariance and the variogram of it are related

$$\gamma_z(\mathbf{r}) = \sigma_z^2 - c_z(\mathbf{r}), \quad (2.24)$$

where σ_z^2 is the constant variance $\forall \mathbf{s} \in D$.

2.4.2 Ergodicity

Ergodicity attributes stochastic processes that their average behavior over time remains identical to that over space. Therefore it follows that ergodic processes are perforce stationary

(without the reverse being true). From practical aspect, ergodicity implies that a single sample is sufficient for the estimation of the statistical properties (i.e. moments) of a random process (Yaglom, 1987).

Since the analysis of random fields is restricted to the study of 2nd-order moments (as already mentioned) the ergodic behavior of a stationary RF can sufficiently be established by demonstrating that the mean and covariance functions are ergodic (Chilès and Delfiner, 2012).

2.4.3 Isotropy & Anisotropy

A stochastic process is called isotropic if its attributes are independent of spatial direction, i.e. it exhibits uniform behavior in all orientations. For a stationary random field, isotropy means that its covariance and variogram can be expressed as functions of the modulus of the vector \mathbf{r} instead of the simple vector, i.e.

$$c_z(\mathbf{r}) = c_z(\|\mathbf{r}\|), \quad (2.25)$$

and

$$\gamma_z(\mathbf{r}) = \gamma_z(\|\mathbf{r}\|). \quad (2.26)$$

Generally, there are two types of anisotropy, **zonal anisotropy** and **range or elliptical anisotropy**. In the first case, the random field exhibits directional dependent variance (σ_z^2). In the later case the directionally dependent attribute of the field is the correlation length (ξ).

Zonal anisotropy can be modelled simply as the superposition of n number of random fields, where n the dimensionality of the field. On the contrary, modelling range anisotropy is more complicated, as the model's parameters include n number of correlation lengths $\xi_1, \xi_2, \dots, \xi_n$ (one for each dimension) and $m = \frac{n}{2} = \frac{n!}{2(n-2)!}$ number of rotation angles $\phi_1, \phi_2, \dots, \phi_m$ (one for each plane containing two of the n axes). It is obvious that isotropy is a subcategory of range anisotropy, where $\xi_1 = \xi_2 = \dots = \xi_n$ and $\phi_1 = \phi_2 = \dots = \phi_m$. For a homogeneous random field modelling anisotropy can be achieved with the computation of an anisotropic norm of the distance, i.e. the covariance can be expressed as

$$c_z(\mathbf{r}) = c_z(\|\mathbf{r}\|_A), \quad (2.27)$$

where $\|\mathbf{r}\|_A$ is the anisotropic norm and is equivalent to the above mentioned dimensionless distance h .

The analytic computation of the anisotropic norm can be derived by using transformation matrices. Any random field can be considered as an anisotropic field, which has been derived

from an initially isotropic one that was rescaled and rotated to its final elliptical form. In a 2D case, this can be mathematically depicted by using the following computations:

$$\begin{aligned}
 \mathbf{r}_{\text{an}} &= \mathbf{R}\mathbf{S}\mathbf{r}_{\text{is}} \\
 \Leftrightarrow \begin{bmatrix} x' \\ y' \end{bmatrix} &= \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\
 \Leftrightarrow \begin{bmatrix} x' \\ y' \end{bmatrix} &= \begin{bmatrix} \xi_1 \cos \phi & -\xi_2 \sin \phi \\ \xi_1 \sin \phi & \xi_2 \cos \phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\
 \Leftrightarrow \mathbf{r}_{\text{an}} &= \mathbf{B}\mathbf{r}_{\text{is}},
 \end{aligned} \tag{2.28}$$

where \mathbf{r}_{is} the initial isotropic coordinations system, \mathbf{r}_{an} the tranformed anisotropic coordinations system, \mathbf{R} the rotation matrix, \mathbf{S} the rescaling matrix, $\mathbf{B} = \mathbf{R}\mathbf{S}$ the total tranformation matrix, ϕ the rotation angle measured counterclockwise between the horizontal axis of the coordinate system and the first met principal axis of the anisotropy ellipse, and ξ_1, ξ_2 the correlation lengths along each of the principal axes direction (ξ_1 refers to the direction of the rotation angle ϕ).

The anisotropic norm of the transformed (anisotropic) coordinations system, which is the point, is equivalent to the euclidean norm of the original isotropic coordinations system, i.e. $\|\mathbf{r}_{\text{an}}\|_A = \|\mathbf{r}_{\text{is}}\|$. Therefore, the inversion of the transformation, a graphical representation of which can also be seen in Fig. 2.2, and subsequently the calculation of the euclidean norm of the inverted (isotropic) coordinations system become necessary steps for the processing of an anisotropic field.

The transformation can be inverted as

$$\mathbf{r}_{\text{is}} = \mathbf{B}^{-1}\mathbf{r}_{\text{an}}, \tag{2.29}$$

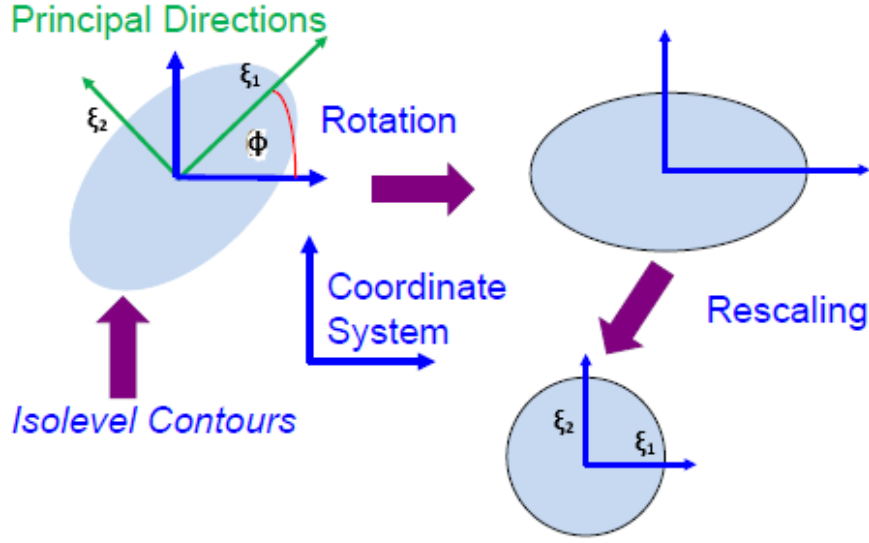


Figure 2.2 Inverse transformation of anisotropic coordinations system

where

$$\begin{aligned}
 \mathbf{B}^{-1} &= [\mathbf{RS}]^{-1} = \mathbf{S}^{-1}\mathbf{R}^{-1} \\
 &= \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix}^{-1} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}^{-1} \\
 &= \frac{1}{\xi_1 \xi_2} \begin{bmatrix} \xi_2 & 0 \\ 0 & \xi_1 \end{bmatrix} \frac{1}{\cos^2 \phi + \sin^2 \phi} \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \\
 &= \begin{bmatrix} \frac{1}{\xi_1} & 0 \\ 0 & \frac{1}{\xi_2} \end{bmatrix} \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} = \begin{bmatrix} \frac{\cos \phi}{\xi_1} & \frac{\sin \phi}{\xi_1} \\ -\frac{\sin \phi}{\xi_2} & \frac{\cos \phi}{\xi_2} \end{bmatrix}
 \end{aligned} \tag{2.30}$$

and the square of the euclidean norm of the inverted coordinations is, then, calculated as

$$\begin{aligned}
 h^2 &\equiv \|\mathbf{r}_{\text{an}}\|_A^2 \equiv \|\mathbf{r}_{\text{is}}\|^2 = \mathbf{r}_{\text{is}}^T \mathbf{r}_{\text{is}} \\
 &= (\mathbf{B}^{-1} \mathbf{r}_{\text{an}})^T \mathbf{B}^{-1} \mathbf{r}_{\text{an}} = \mathbf{r}_{\text{an}}^T (\mathbf{B}^{-1})^T \mathbf{B}^{-1} \mathbf{r}_{\text{an}}.
 \end{aligned} \tag{2.31}$$

By setting $\mathbf{A} = (\mathbf{B}^{-1})^T \mathbf{B}^{-1}$ and substituting in Eq. (2.31) follows a more concise expression for the dimensionless distance

$$h \equiv \|\mathbf{r}_{\text{an}}\|_A = \sqrt{\mathbf{r}_{\text{an}}^T \mathbf{A} \mathbf{r}_{\text{an}}}, \quad (2.32)$$

where A can be computed analytically by using the Eq. (2.30) as

$$\begin{aligned} A &= (\mathbf{B}^{-1})^T \mathbf{B}^{-1} \\ &= \begin{bmatrix} \frac{\cos \phi}{\xi_1} & \frac{\sin \phi}{\xi_1} \\ -\frac{\sin \phi}{\xi_2} & \frac{\cos \phi}{\xi_2} \end{bmatrix}^T \begin{bmatrix} \frac{\cos \phi}{\xi_1} & \frac{\sin \phi}{\xi_1} \\ -\frac{\sin \phi}{\xi_2} & \frac{\cos \phi}{\xi_2} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\cos \phi}{\xi_1} & -\frac{\sin \phi}{\xi_2} \\ \frac{\sin \phi}{\xi_1} & \frac{\cos \phi}{\xi_2} \end{bmatrix} \begin{bmatrix} \frac{\cos \phi}{\xi_1} & \frac{\sin \phi}{\xi_1} \\ -\frac{\sin \phi}{\xi_2} & \frac{\cos \phi}{\xi_2} \end{bmatrix} \\ &= \begin{bmatrix} \left(\frac{\cos \phi}{\xi_1}\right)^2 + \left(\frac{\sin \phi}{\xi_2}\right)^2 & \left(\frac{1}{\xi_1^2} - \frac{1}{\xi_2^2}\right) \cos \phi \sin \phi \\ \left(\frac{1}{\xi_1^2} - \frac{1}{\xi_2^2}\right) \cos \phi \sin \phi & \left(\frac{\sin \phi}{\xi_1}\right)^2 + \left(\frac{\cos \phi}{\xi_2}\right)^2 \end{bmatrix}. \end{aligned} \quad (2.33)$$

Setting

$$A_1 = \left(\frac{\cos \phi}{\xi_1}\right)^2 + \left(\frac{\sin \phi}{\xi_2}\right)^2 \quad (2.34a)$$

$$A_2 = \left(\frac{\sin \phi}{\xi_1}\right)^2 + \left(\frac{\cos \phi}{\xi_2}\right)^2. \quad (2.34b)$$

$$A_{12} = \left(\frac{1}{\xi_1^2} - \frac{1}{\xi_2^2}\right) \cos \phi \sin \phi \quad (2.34c)$$

in Eq. (2.33) and combining with Eq. (2.32) it follows that

$$\begin{aligned}
 h &\equiv \|\mathbf{r}_{\text{an}}\|_A = \sqrt{\mathbf{r}_{\text{an}}^T \mathbf{A} \mathbf{r}_{\text{an}}} \\
 &= \left(\begin{bmatrix} x' \\ y' \end{bmatrix}^T \begin{bmatrix} A_1 & A_{12} \\ A_{12} & A_2 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \right)^{1/2} \\
 &= \sqrt{A_1 x'^2 + 2A_{12} x' y' + A_2 y'^2}.
 \end{aligned} \tag{2.35}$$

Note that in case of isotropic initial random field, i.e. $\xi_1 = \xi_2 = \xi$ and $\phi = 0$, it follows that $A_1 = A_2 = 1/\xi^2$, $A_{12} = 0$ and the resulting dimensionless distance is

$$h = \sqrt{x'^2 + y'^2} / \xi = \|\mathbf{r}\| / \xi, \tag{2.36}$$

as it would be expected. Also, noticeable is the fact that an anisotropic field can be equivalently processed either as anisotropic by using the 2D dimensionless distance defined in Eq. (2.35) or as isotropic, using Eq. (2.36), after inverting the transformation effect.

Similar calculations can also be applied for 3-dimensional RFs, with the difference that three rotation angles (Euler angles) and three correlation lengths are needed in these cases to perform the inverse transformation. The corresponding 3D transformation matrix \mathbf{B} is

$$\mathbf{B} = \begin{bmatrix} \xi_1 & 0 & 0 \\ 0 & \xi_2 & 0 \\ 0 & 0 & \xi_3 \end{bmatrix} \begin{bmatrix} \cos \phi_1 & -\sin \phi_1 & 0 \\ \sin \phi_1 & \cos \phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_2 & -\sin \phi_2 \\ 0 & \sin \phi_2 & \cos \phi_2 \end{bmatrix} \begin{bmatrix} \cos \phi_3 & -\sin \phi_3 & 0 \\ \sin \phi_3 & \cos \phi_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.37}$$

2.5 Permissible Covariance Functions

The covariance functions can not be any function, but have to meet some conditions. The conditions that determine the permissible covariance functions are provided by the Bochner's theorem.

Theorem 2.1. *A function $c_z(\mathbf{r})$ is a permissible covariance function, if the following conditions hold:*

- (i) *the integral $\tilde{c}_z(\mathbf{k}) = \int c_z(\mathbf{r}) e^{-i\mathbf{k}\mathbf{r}} d\mathbf{r}$ exists and is symmetric $\tilde{c}_z(\mathbf{k}) = \tilde{c}_z(-\mathbf{k})$,*
- (ii) *it is non negative for all frequencies \mathbf{k} , and*
- (iii) *is bounded for all frequencies \mathbf{k} .*

The first condition implies that the Fourier transform of $c_z(\mathbf{r})$ (also called spectral density function of the random field) exists. Moreover, the symmetry of $c_z(\mathbf{r})$ ensures the positive semidefiniteness of the covariance function, which is essential for the inversion of covariance matrices (see section 3). The second condition demands that $\tilde{c}_z(\mathbf{k}) \geq 0 \quad \forall \mathbf{k} \in \mathbb{R}^n$. Finally, the third condition ensures the existence of the RF's variance (σ_z^2) (Bochner et al., 1959).

Table 2.2 Permissible Covariance Functions. $K_\nu(\cdot)$ denote the modified Bessel functions of the second kind of order ν , $\Gamma(\cdot)$ denote the Gamma function, h denotes the dimensionless distance, $\Delta = |\eta_1^2 - 4|^{1/2}$, $\beta_{1,2} = |2 \mp \eta_1|^{1/2}/2$, and $\omega_{1,2} = (|\eta_1 \mp \Delta|/2)^{1/2}$.

Model	Covariance
Nugget Effect	$c_z(h) = \begin{cases} \sigma_z^2, & h = 0 \\ 0, & h \neq 0 \end{cases}$
Generalized Exponential	$c_z(h) = \sigma_z^2 e^{-h^\nu}, \quad 0 < \nu < 2$
Gaussian	$c_z(h) = \sigma_z^2 e^{-h^2}$
Spherical	$c_z(h) = \begin{cases} \sigma_z^2 (1 - 1.5h + 0.5h^3), & 0 \leq h \leq 1 \\ 0, & h > 1 \end{cases}$
Cubic	$c_z(h) = \begin{cases} \sigma_z^2 \left(1 - 7h^2 + \frac{35}{4}h^3 - \frac{7}{2}h^5 + \frac{3}{4}h^7 \right), & 0 \leq h \leq 1 \\ 0, & h > 1 \end{cases}$
Matérn (or Bessel-K)	$c_z(h) = \sigma_z^2 \frac{h^\nu K_\nu(h)}{2^{\nu-1} \Gamma(\nu)}, \quad \nu > 0$
Generalized Cauchy	$c_z(h) = \sigma_z^2 (1 + h^\alpha)^{-\nu/\alpha}, \quad 0 < \alpha < 2, \quad \nu > 0$
Rational Quadratic	$c_z(h) = \sigma_z^2 (1 + h^2)^{-\nu}, \quad \nu > 0$
Exponential Sine	$c_z(h) = \sigma_z^2 \sin\left(\frac{\pi}{2} e^{-h}\right)$
Cardinal Sine	$c_z(h) = \sigma_z^2 \frac{\sin(h)}{h}$
Spartan (3D)	$c_z(h) = \begin{cases} \frac{\eta_0}{2\pi\Delta} e^{-h\beta_2} \left(\frac{\sin(h\beta_1)}{h} \right), & \eta_1 < 2 \quad (\sigma_z^2 = \frac{\eta_0}{4\pi\sqrt{2+\eta_1}}), \\ \frac{\eta_0}{8\pi} e^{-h}, & \eta_1 = 2 \quad (\sigma_z^2 = \frac{\eta_0}{8\pi}), \\ \frac{\eta_0}{4\pi\Delta} \left(\frac{e^{-h\omega_1} - e^{-h\omega_2}}{h} \right), & \eta_1 > 2 \quad (\sigma_z^2 = \frac{\eta_0}{4\pi\Delta} (\omega_2 - \omega_1)) \end{cases}$

The mathematical expressions of some permissible covariance functions are shown in Table 2.2, while Fig. 2.3 depicts some graphs of them.

The model of Nugget Effect describes fluctuations that are spatially uncorrelated, such those that take place in distances shorter than the resolution of the sample or those resulting from errors of the sampling method.

The Generalized Exponential model characterizes distributions with abrupt spatial changes, i.e. the spatial correlation decreases rapidly with the increase of distance. The rapidity of the reduction increases with the decreasing of the value of the smoothness parameter ν . The most widely used model of the Generalized Exponential family is the one with $\nu = 1$; the Exponential model.

In contrast, the Gaussian model characterizes smoother spatial changes. However, a strong disadvantage of it is that it can lead to numerical instabilities in the covariance matrix calculation.

Spherical and Cubic models indicate RFs with even faster decreasing. Also, unlike the other models which asymptotically decline to zero when $h \rightarrow \infty$, they are zeroed for $h > 1$ (h denotes the dimensionless distance as defined in section 2.4.3).

Exponential Sine and Cardinal Sine models describe fluctuations with wavy and oscillatory behavior. The Generalized Cauchy model and the Rational Quadratic model, which is a subcase of the first, exhibit power-law dependence with different exponents in small and large scales (Chilès and Delfiner, 2012).

The Matérn or Bessel-K model is characterized by great flexibility since different values of the parameter ν correspond to different behaviors of the model. In Table 2.3 the special mathematical expressions of the Matérn model for different values of ν are presented (Chilès and Delfiner, 2012; Guttorp and Gneiting, 2006).

Finally, the Spartan model (SSRF) derives from the generalized Gibbs random fields theory. It differs from the other models since it is expressed through the parameters η_0 and η_1 instead of σ_z^2 . The parameter η_0 is the scaling factor (defined the magnitude) of the covariance function, while the parameter η_1 (rigidity parameter) is related to the shape. The Spartan model, similar to the Matérn, exhibits great flexibility due to the different expressions obtained depending on the value of the parameter η_1 . For $\eta_1 = 2$ the Spartan model is equivalent to the Exponential model, for $|\eta_1| < 2$ it is similar to the product of Exponential and Cardinal Sine models and exhibits oscillatory behavior, and for $\eta_1 > 2$ a new form is obtained (Hristopulos and Elogne, 2007).

Furthermore, any combination (e.g. summation, multiplication, etc.) of permissible covariance functions leads to also permissible covariance function. The most commonly used combination is the summation of the nugget effect model with any of the other permissible

models (Fig. 2.5). This combination enables the modelling of a data sample which contains errors due to the low resolution or the errors of the acquisition method or equipment.

Table 2.3 Special cases of Matérn model for different values of ν

ν	Expression	Name
$\nu = 1/3$	$c_z(h) = \sigma_z^2 2^{2/3} h^{1/3} K_{1/3}(h) / \Gamma(1/3)$	von Kármán
$\nu = 1/2$	$c_z(h) = \sigma_z^2 e^{-h}$	Exponential
$\nu = 1$	$c_z(h) = \sigma_z^2 h K_1(h)$	Whittle
$\nu = 3/2$	$c_z(h) = \sigma_z^2 (1 + h) e^{-h}$	Modified Exponential
$\nu = 2$	$c_z(h) = \sigma_z^2 \frac{h^2}{2} K_2(h)$	-
$\nu = 5/2$	$c_z(h) = \sigma_z^2 (1 + h + \frac{1}{3} h^2) e^{-h}$	3rd order autoregressive
$\nu = 3$	$c_z(h) = \sigma_z^2 \frac{1}{8} h^3 K_3(h)$	-
$\nu = 7/2$	$c_z(h) = \sigma_z^2 \frac{h^{7/2} K_{7/2}(h)}{2^{5/2} \Gamma(7/2)}$	-

Having determined the permissible covariance models, it is possible to determine the corresponding variogram models, at least for stationary random fields, as Eq. (2.24) indicates. Examples of permissible variogram models are shown in Table 2.4 and plots in Fig. 2.4.

Table 2.4 Permissible Variogram Functions. $K_\nu(\cdot)$ denote the modified Bessel functions of the second kind of order ν , $\Gamma(\cdot)$ denote the Gamma function, h denotes the dimensionless distance, $\Delta = |\eta_1^2 - 4|^{1/2}$, $\beta_{1,2} = |2 \mp \eta_1|^{1/2}/2$, and $\omega_{1,2} = (|\eta_1 \mp \Delta|/2)^{1/2}$.

Model	Variogram
Nugget Effect	$\gamma_z(h) = \begin{cases} 0, & h = 0 \\ \sigma_z^2, & h \neq 0 \end{cases}$
Generalized Exponential	$\gamma_z(h) = \sigma_z^2 \left(1 - e^{-h^\nu}\right), \quad 0 < \nu < 2$
Gaussian	$\gamma_z(h) = \sigma_z^2 \left(1 - e^{-h^2}\right)$
Spherical	$\gamma_z(h) = \begin{cases} \sigma_z^2 \left(1.5h - 0.5h^3\right), & 0 \leq h \leq 1 \\ \sigma_z^2, & h > 1 \end{cases}$
Cubic	$\gamma_z(h) = \begin{cases} \sigma_z^2 \left(7h^2 - \frac{35}{4}h^3 + \frac{7}{2}h^5 - \frac{3}{4}h^7\right), & 0 \leq h \leq 1 \\ \sigma_z^2, & h > 1 \end{cases}$
Matérn (or Bessel-K)	$\gamma_z(h) = \sigma_z^2 \left(1 - \frac{h^\nu K_\nu(h)}{2^{\nu-1} \Gamma(\nu)}\right), \quad \nu > 0$
Generalized Cauchy	$\gamma_z(h) = \sigma_z^2 \left(1 - (1 + h^\alpha)^{-\nu/\alpha}\right), \quad 0 < \alpha < 2, \quad \nu > 0$
Rational Quadratic	$\gamma_z(h) = \sigma_z^2 \left(1 - (1 + h^2)^{-\nu}\right), \quad \nu > 0$
Exponential Sine	$\gamma_z(h) = \sigma_z^2 \left(1 - \sin\left(\frac{\pi}{2}e^{-h}\right)\right)$
Cardinal Sine	$\gamma_z(h) = \sigma_z^2 \left(1 - \frac{\sin(h)}{h}\right)$
Spartan (3D)	$\gamma_z(h) = \begin{cases} \frac{\eta_0}{2\pi} \left(\frac{1}{2\sqrt{2+\eta_1}} - \frac{1}{\Delta}\right) e^{-h\beta_2} \frac{\sin(h\beta_1)}{h}, & \eta_1 < 2, \\ \frac{\eta_0}{8\pi} \left(1 - e^{-h}\right), & \eta_1 = 2, \\ \frac{\eta_0}{4\pi\Delta} (\omega_2 - \omega_1) \left(1 - \frac{e^{-h\omega_1} - e^{-h\omega_2}}{h(\omega_2 - \omega_1)}\right), & \eta_1 > 2 \end{cases}$

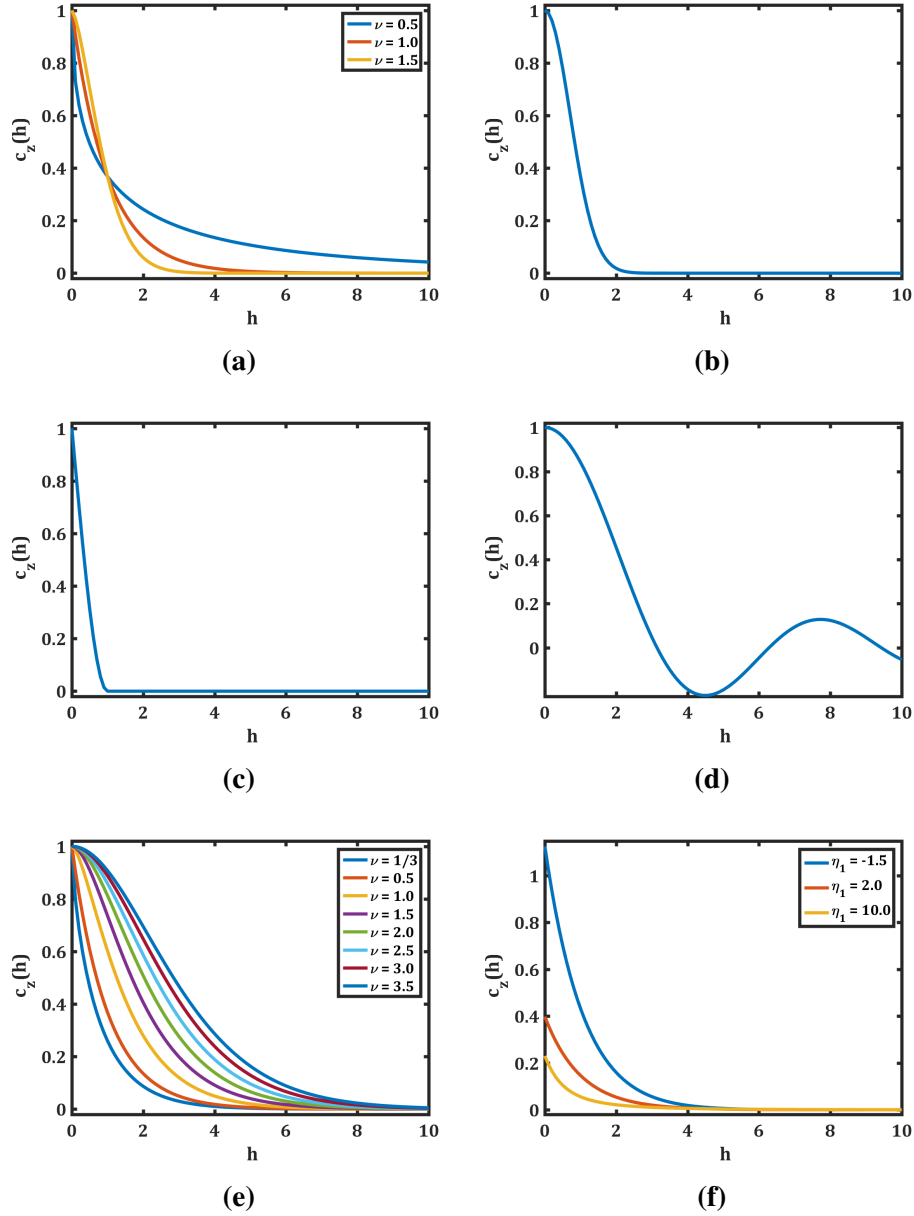


Figure 2.3 Permissible Covariance Models: a) Generalized Exponential; b) Gaussian; c) Spherical; d) Cardinal Sine; e) Matérn; f) Spartan (3D). The used parameters are $\sigma_z^2 = 1$, $\xi = 1$ and $\eta_0 = 10$ (h is the dimensionless distance $h = r/\xi$ where ξ the correlation length).

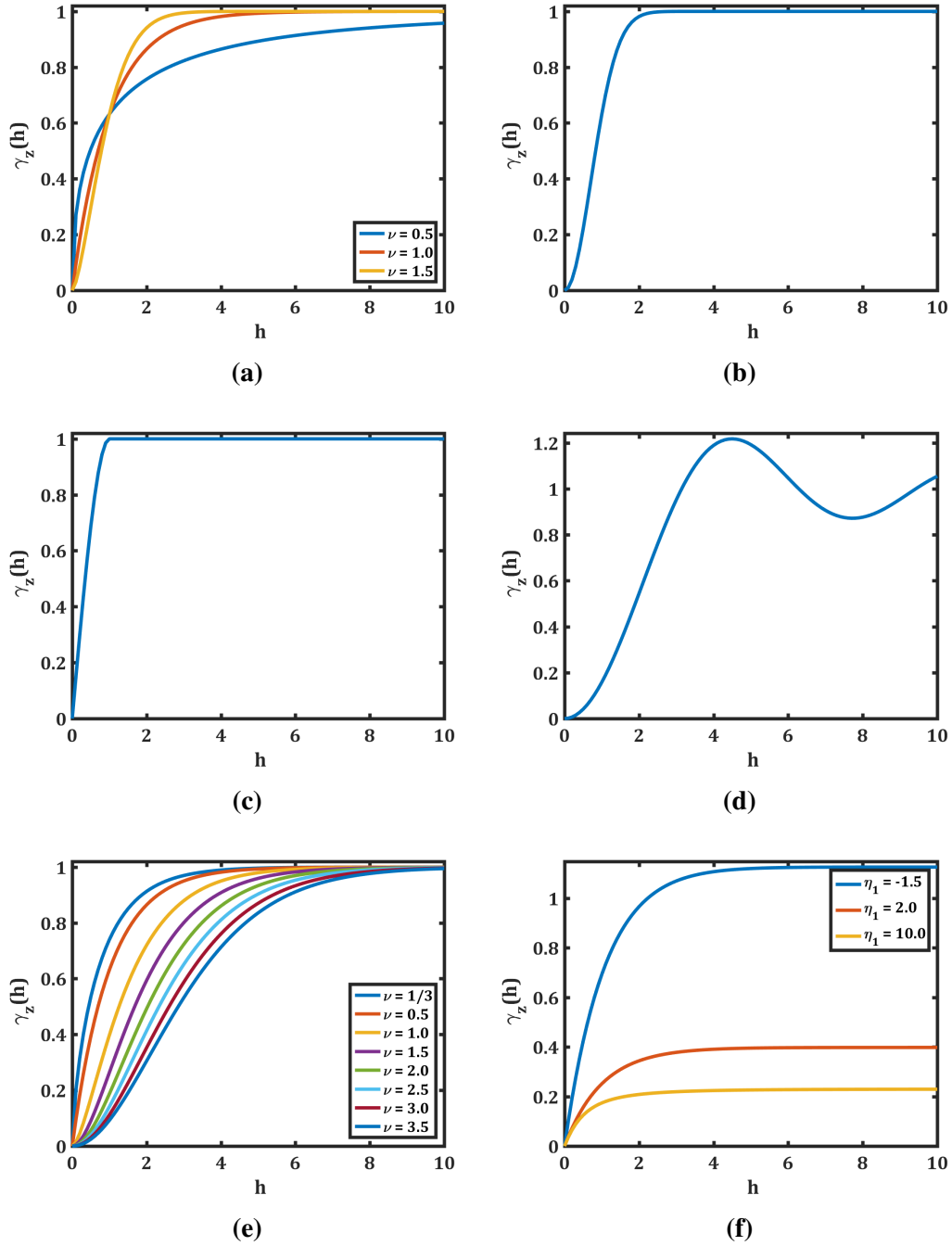


Figure 2.4 Variogram Models: a) Generalized Exponential; b) Gaussian; c) Spherical; d) Cardinal Sine; e) Matérn; f) Spartan (3D). The used parameters are $\sigma_z^2 = 1$, $\xi = 1$ and $\eta_0 = 10$ (h is the dimensionless distance $h = r/\xi$ where ξ the correlation length).

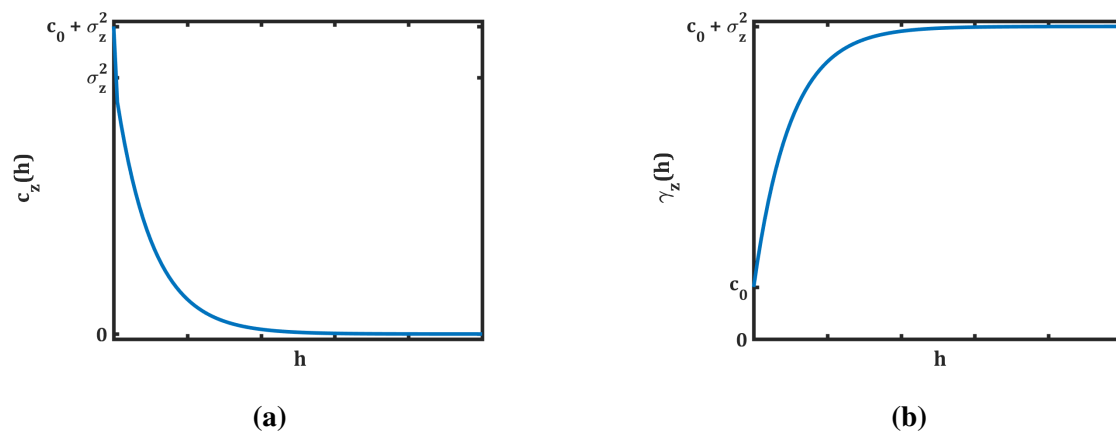


Figure 2.5 Summation of Permissible Model with Nugget Effect: a) Covariance; b) Variogram.

2.6 Experimental Covariance and Variogram

In many cases, the computation of the experimental covariance or variogram from the data is necessary. The experimental variogram is computed as:

$$\hat{\gamma}_z(\mathbf{h}) = \frac{1}{2N(\mathbf{h})} \sum_{i=1}^{N(\mathbf{h})} [z(\mathbf{s}_i) - z(\mathbf{s}_i + \mathbf{h})]^2, \quad (2.38)$$

and the experimental covariance as:

$$\hat{C}_z(\mathbf{h}) = \frac{1}{2N(\mathbf{h})} \sum_{i=1}^{N(\mathbf{h})} [z(\mathbf{s}_i)z(\mathbf{s}_i + \mathbf{h}) - \hat{m}_z(\mathbf{s}_i - \mathbf{h})\hat{m}_z(\mathbf{s}_i + \mathbf{h})], \quad (2.39)$$

where $N(\mathbf{h})$ is the number of pairs of data locations i which are separated by a vector \mathbf{h} . Hence, in a 2D anisotropic case the space is separated into cyclical sections defined from an upper and lower angular limit and an upper and lower distance limit (see Fig. 2.6). For each of these sections the corresponding experimental variogram value $\hat{\gamma}_z(\mathbf{s}_{\mathbf{c},\mathbf{k}})$ is calculated and attached to the center of it, $\mathbf{s}_{\mathbf{c},\mathbf{k}}$. For isotropic 2D cases or 1D cases, only distance limits are applied and the experimental variogram values are attached to the center of the resulting distance sections (or lags).

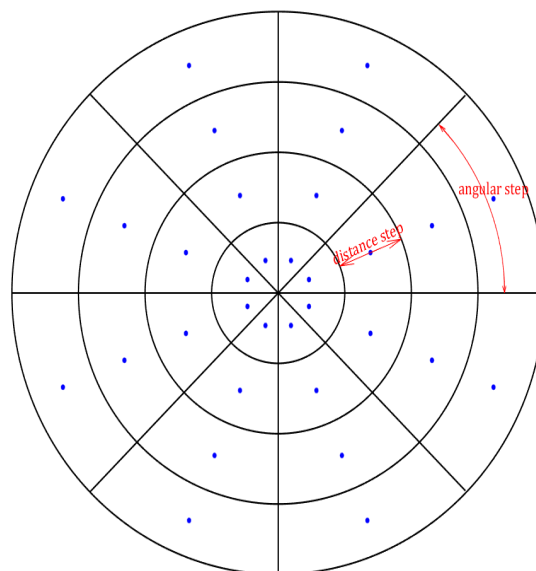


Figure 2.6 Space separation into cyclical sections

Chapter 3

Kriging

3.1 Basic Notions of Kriging

Kriging is the method developed by the South African engineer Danie G. Krige in 1951 while working on the evaluation of the Witwatersrand reef complex (South Africa) gold resources (Cressie, 1990; Krige, 1951). Kriging is the most commonly used geostatistical method of interpolation. The Kriging Estimator predicts the value of a random field $Z(\cdot)$ at a point \mathbf{s}_u , where no measurements are available, as a weighted average of n neighboring values ($Z(\mathbf{s}_1), Z(\mathbf{s}_2), \dots, Z(\mathbf{s}_n)$). The set of prediction points (S_u) comprises the nodes of a regular grid, which covers the area of interest. In this way, the estimation of the missing values leads to the mapping and the iso-level contour representation of the random field over it. The neighboring points, where measurements are available, consist the sampling set (S_s). For the predictions ideally the whole sampling set is used. However, in the cases where the computational complexity is high, it is possible to use only a part of the dataset. This can be achieved by using only the sampling points within a user-defined kriging neighborhood around the prediction point.

The generalized Kriging Estimator is mathematically expressed as:

$$\hat{Z}(\mathbf{u}) = m_z(\mathbf{u}) + \sum_{i=1}^{n(\mathbf{u})} \lambda_i [Z(\mathbf{s}_i) - m_z(\mathbf{s}_i)], \quad (3.1)$$

where λ_i are the linear weights of the sampling points and $n(\mathbf{u})$ is the number of the sampling points within the kriging neighborhood of the prediction point \mathbf{s}_u .

The weights of the Kriging Estimator are calculated by considering three restrictions:

- the estimations must respect the real data; thus, the statistical information derived from the data (covariance or variogram model) is used,

- the variance of the error of the estimation must be minimized, and
- the unbiasedness constraint must be fulfilled.

The predictions given by the Kriging Estimator, due to the minimization of the variance of the estimation's error, are the best linear unbiased estimations (BLUE) (Goovaerts, 1997). Kriging, in contrast to other methods of interpolation, makes possible the quantification of the uncertainty of the estimations at each point, giving a validation tool for the estimations' reliability.

3.1.1 Error Variance

The error variance is given from the equation:

$$\sigma_E^2(\mathbf{u}) = \text{Var} \left[Z(\mathbf{u}) - \hat{Z}(\mathbf{u}) \right]. \quad (3.2)$$

The minimization of it and therefore the obtainment of the optimal linear weights λ_i is achieved by setting to zero each of the $n(\mathbf{u})$ partial first derivatives of the error variance with respect to the weights, i.e.

$$\frac{\partial \sigma_E^2(\mathbf{u})}{\partial \lambda_i} = 0, \quad i = 1, \dots, n(\mathbf{u}). \quad (3.3)$$

The restriction imposed by the *unbiasedness constraint*, under which the optimization must be done, ensures that the mean value of the error $\varepsilon(\mathbf{u}) = Z(\mathbf{u}) - \hat{Z}(\mathbf{u})$ is zeroed, i.e.

$$\mathbb{E}[\varepsilon(\mathbf{u})] = \mathbb{E}[Z(\mathbf{u}) - \hat{Z}(\mathbf{u})] = 0. \quad (3.4)$$

If it is not fulfilled by default from the Kriging Estimator (see section 3.2.2), it is imposed as an additional restriction for the linear weights (Chilès and Delfiner, 2012; Oliver and Webster, 2015).

3.2 Methods of Kriging

There are various methods of Kriging, with the following to be the most widely used:

- **Simple Kriging (SK)**, which applies when the mean value of the random field is constant and known.
- **Ordinary Kriging (OK)**, which applies when the mean value of the field is constant within the neighborhood of the estimated point but unknown.

- **Universal Kriging (UK)**, which applies when the mean value of the field is not constant (depends on the location) and unknown.
- **Indicator Kriging (IK)**, which applies when the values of the random field can be sorted into categories by using threshold values.
- **Co-Kriging (CK)**, which applies when the values of the random field are correlated to more than one variables, i.e. in multivariate cases, where the cross-correlation functions between the variables must be taken into account.

Herein, only the first two methods of Kriging will be described in details.

3.2.1 Simple Kriging

Simple Kriging (SK) is applied when the mean value of the random field is known and constant throughout the area of interest, i.e. $\mathbb{E}[Z(\mathbf{s})] = m_z$. Replacing to Eq. (3.1), the expression for the Simple Kriging Estimator derives as:

$$\hat{Z}(\mathbf{u}) = m_z \left[1 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i \right] + \sum_{i=1}^{n(\mathbf{u})} \lambda_i Z(\mathbf{s}_i). \quad (3.5)$$

The mean of the estimation's error is:

$$\begin{aligned} \mathbb{E}[\varepsilon(\mathbf{u})] &= \mathbb{E}[Z(\mathbf{u}) - \hat{Z}(\mathbf{u})] \\ &= \mathbb{E}\left[Z(\mathbf{u}) - m_z \left[1 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i \right] - \sum_{i=1}^{n(\mathbf{u})} \lambda_i Z(\mathbf{s}_i)\right] \\ &= \mathbb{E}[Z(\mathbf{u})] - \mathbb{E}[m_z] + \sum_{i=1}^{n(\mathbf{u})} \lambda_i \mathbb{E}[m_z] - \sum_{i=1}^{n(\mathbf{u})} \lambda_i \mathbb{E}[Z(\mathbf{s}_i)] \\ &= m_z - m_z + \sum_{i=1}^{n(\mathbf{u})} \lambda_i m_z - \sum_{i=1}^{n(\mathbf{u})} \lambda_i m_z = 0. \end{aligned} \quad (3.6)$$

This means that the SK Estimator fulfills by construction the unbiasedness constraint and therefore no additional restriction has to be imposed to the error variance minimization.

The optimal linear weights λ_i are calculated by solving the linear equations system deriving from Eq. (3.3):

$$\sum_{i=1}^{n(\mathbf{u})} \lambda_j c_z(\mathbf{s}_i - \mathbf{s}_j) = c_z(\mathbf{s}_i - \mathbf{s}_u), \quad i, j = 1, \dots, n(\mathbf{u}), \quad (3.7)$$

or in formulation of matrix equation:

$$\begin{aligned}
 \mathbf{C}_{i,j}\lambda_j &= \mathbf{C}_{i,u} \\
 \Leftrightarrow \begin{bmatrix} c_z(\mathbf{s}_1 - \mathbf{s}_1) & \cdots & \cdots & c_z(\mathbf{s}_1 - \mathbf{s}_n) \\ c_z(\mathbf{s}_2 - \mathbf{s}_1) & \cdots & \cdots & c_z(\mathbf{s}_2 - \mathbf{s}_n) \\ \vdots & \vdots & \vdots & \vdots \\ c_z(\mathbf{s}_n - \mathbf{s}_1) & \cdots & \cdots & c_z(\mathbf{s}_n - \mathbf{s}_n) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} &= \begin{bmatrix} c_z(\mathbf{s}_1 - \mathbf{u}) \\ c_z(\mathbf{s}_2 - \mathbf{u}) \\ \vdots \\ c_z(\mathbf{s}_n - \mathbf{u}) \end{bmatrix} \quad (3.8) \\
 \Leftrightarrow \begin{bmatrix} \sigma_z^2 & \cdots & \cdots & c_z(\mathbf{s}_1 - \mathbf{s}_n) \\ c_z(\mathbf{s}_2 - \mathbf{s}_1) & \cdots & \cdots & c_z(\mathbf{s}_2 - \mathbf{s}_n) \\ \vdots & \vdots & \vdots & \vdots \\ c_z(\mathbf{s}_n - \mathbf{s}_1) & \cdots & \cdots & \sigma_z^2 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} &= \begin{bmatrix} c_z(\mathbf{s}_1 - \mathbf{u}) \\ c_z(\mathbf{s}_2 - \mathbf{u}) \\ \vdots \\ c_z(\mathbf{s}_n - \mathbf{u}) \end{bmatrix},
 \end{aligned}$$

where $\mathbf{C}_{i,j}$ represents the covariance matrix between the neighbors of the studied missing value, and $\mathbf{C}_{i,u}$ represents the covariance matrix between the neighbors and the missing value (Goovaerts, 1997). These covariance matrices are calculated using the variogram model, inferred from the real data, as indicated by Eq. (2.24).

The linear system has a solution if and only if the covariance function is permissible (semi-positive definite), which implies that the covariance matrix can be inverted. The solution is given by the following equation:

$$\lambda_j = \mathbf{C}_{i,j}^{-1} \mathbf{C}_{i,u}, \quad \forall j = 1, \dots, n(\mathbf{u}), \quad (3.9)$$

and the error variance, which determines the uncertainty of the estimation, is calculated as:

$$\sigma_{E,SK}^2(\mathbf{u}) = \sigma_z^2 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i c(\mathbf{s}_i - \mathbf{s}_u), \quad i = 1, \dots, n(\mathbf{u}). \quad (3.10)$$

3.2.2 Ordinary Kriging

Ordinary Kriging (OK) is applied when the mean value of the random field is constant within the neighborhood of the missing point but unknown. In this case, the Ordinary Kriging Estimator which derives from Eq. (3.1) is expressed as:

$$\hat{Z}(\mathbf{u}) = \sum_{i=1}^{n(\mathbf{u})} \lambda_i Z(\mathbf{s}_i). \quad (3.11)$$

The mean of the estimation's error is:

$$\begin{aligned}\mathbb{E}[\varepsilon(\mathbf{u})] &= \mathbb{E}[Z(\mathbf{u}) - \hat{Z}(\mathbf{u})] = \mathbb{E}[Z(\mathbf{u}) - \sum_{i=1}^{n(\mathbf{u})} \lambda_i Z(\mathbf{s}_i)] \\ &= \mathbb{E}[Z(\mathbf{u})] - \sum_{i=1}^{n(\mathbf{u})} \lambda_i \mathbb{E}[Z(\mathbf{s}_i)] = m_z - \sum_{i=1}^{n(\mathbf{u})} \lambda_i m_z.\end{aligned}\quad (3.12)$$

Considering that the mean value of the estimations has, obviously, to be equal to the mean value of the random field, i.e. $m_{\hat{z}} \equiv m_z$ and setting the mean of the estimation's error to zero, as unbiasedness constraint imposes, Eq. (3.12) leads to the following:

$$\begin{aligned}\mathbb{E}[\varepsilon(\mathbf{u})] = 0 &\Leftrightarrow m_z - \sum_{i=1}^{n(\mathbf{u})} \lambda_i m_z = 0 \Leftrightarrow m_z - \sum_{i=1}^{n(\mathbf{u})} \lambda_i m_z = 0 \Leftrightarrow m_z \left(1 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i\right) = 0 \\ &\Leftrightarrow 1 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i = 0 \Leftrightarrow \sum_{i=1}^{n(\mathbf{u})} \lambda_i = 1.\end{aligned}\quad (3.13)$$

The above result indicates that the OK Estimator, as opposed to the SK Estimator, is not unbiased by structure but the additional limitation of unit summed linear weights needs to be imposed to the error variance minimization. This is achieved by means of the *Lagrange multiplier method*, and one more partial derivative of the error variance is added to the linear equations system of Eq. (3.3), this with respect to the Lagrange multiplier, μ , i.e.

$$\frac{\partial \sigma_E^2(\mathbf{u})}{\partial \mu} = 0. \quad (3.14)$$

Therefore, the optimal linear weights λ_i of Ordinary Kriging are calculated by solving the following linear equations system:

$$\begin{aligned}\sum_{i=1}^{n(\mathbf{u})} \lambda_j c_z(\mathbf{s}_i - \mathbf{s}_j) + \mu &= c_z(\mathbf{s}_i - \mathbf{s}_{\mathbf{u}}), \quad i, j = 1, \dots, n(\mathbf{u}) \\ \sum_{i=1}^{n(\mathbf{u})} \lambda_i &= 1,\end{aligned}\quad (3.15)$$

or in formulation of matrix equation:

$$\begin{bmatrix} \sigma_z^2 & c_z(\mathbf{s}_1 - \mathbf{s}_2) & \cdots & c_z(\mathbf{s}_1 - \mathbf{s}_n) & 1 \\ c_z(\mathbf{s}_2 - \mathbf{s}_1) & \sigma_z^2 & \cdots & c_z(\mathbf{s}_2 - \mathbf{s}_n) & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_z(\mathbf{s}_n - \mathbf{s}_1) & c_z(\mathbf{s}_n - \mathbf{s}_2) & \cdots & \sigma_z^2 & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ \mu \end{bmatrix} = \begin{bmatrix} c_z(\mathbf{s}_1 - \mathbf{u}) \\ c_z(\mathbf{s}_2 - \mathbf{u}) \\ \vdots \\ c_z(\mathbf{s}_n - \mathbf{u}) \\ 1 \end{bmatrix}. \quad (3.16)$$

The error variance of the Ordinary Kriging estimation is calculated as:

$$\sigma_{E,OK}^2(\mathbf{u}) = \sigma_z^2 - \sum_{i=1}^{n(\mathbf{u})} \lambda_i c(\mathbf{s}_i - \mathbf{s}_u) - \mu, \quad i = 1, \dots, n(\mathbf{u}). \quad (3.17)$$

Comparing it with the error variance of the SK it follows that

$$\sigma_{E,OK}^2(\mathbf{u}) = \sigma_{E,SK}^2(\mathbf{u}) - \mu, \quad (3.18)$$

which, taking into account the always negative value of parameter μ , indicates that the uncertainty of OK is greater than this of SK. This is due to the assumption of unknown mean value that OK method makes, which leads to greater uncertainty.

Chapter 4

Stochastic Simulations

Stochastic Simulations is a widely used tool for estimating missing values of a random field in cases when the existing information is deemed inadequate for straightforward interpolation, for estimating the cdf of a variable from a dataset or for estimating error margins. In general, stochastic simulations generate a large number of realizations of the studied field, which emulate the characteristics of the observed realization (real data). By processing statistically these realizations the best estimation of the field, as well as the associated uncertainty can be derived.

The main difference between stochastic simulation and kriging is that the former aims to maintain the general characteristics of the data over local accuracy, while the later gives the estimation with the minimum error variance without deference to the global statistics.

There are several theoretically established and practically implemented stochastic simulation methods. They can be, broadly, categorized into *parametric* and *non-parametric* depending on the necessity or not of a priori knowledge of an explicit mathematical (parametric) model, as well as into *conditional* and *unconditional* depending on the placement or not of constraints on the values generated at sampling points, so as the simulation returns the known values (Webster and Oliver, 2007).

Herein, two methods will be briefly presented: i) Turning-Bands (TB) method, and ii) Directional Gradient-Curvature (DGC) method.

4.1 Turning-Bands Method

The Turning-Bands Method (Chentsov, 1957; Mantoglou and Wilson, 1982) is a parametric unconditional simulation method, which applies on second-order stationary, isotropic and Gaussian random fields.

The general procedure of the TB method is described by the following steps:

1. Performance, if needed, of suitable transformations of the data values, aiming to the fulfilment of stationarity, isotropy and normality preconditions. Data values are assumed to transform from the z-space to the y-space.
2. Computation and modelling of the covariance $c_s(h)$ of the y-space data.
3. Determination of a regular square or cubic grid, in two or three dimensions, respectively, covering the study area.
4. Generation of L number of random lines (bands) around an arbitrary origin of the grid (usually a centroid located at the grid center) with the corresponding direction vectors \mathbf{e} uniformly distributed on the unit circle or sphere. The number of the bands is controlled by the modeler.
5. Generation, for each line, of a second-order stationary unidimensional discrete process with zero mean and covariance function $c_s(l_i)$, where l_i is the coordinate on the corresponding line. Due to the discrete line process, a set of bands is defined from the limits of each discretized segment, which turn, as the lines turn; hence the name 'turning bands method'.
6. Orthogonal projection of the regular grid's nodes \mathbf{s}_g to each line and assignment to them the corresponding values of the one dimensional discrete process. As a result derives the generation of L independent unidimensional realizations (one for each line) with covariance function $c_s(l_i)$, i.e. at every node of the regular grid there are L values $z_i(l_i)$ assigned from the L unidimensional realizations (simulations).
7. Assignment to each node the value $z(\mathbf{s}_g)$ given by

$$z(\mathbf{s}_g) = \frac{1}{\sqrt{L}} \sum_{i=1}^L z_i(l_i), \quad i = 1, \dots, L, \quad (4.1)$$

as the realization of the two- or three- dimensional random field.

8. Estimation of the uncertainty of each node's value as

$$\sigma^2(\mathbf{s}_g) = \text{Var}[z_i(l_i)z_j(l_j)], \quad i, j = 1, \dots, L. \quad (4.2)$$

9. Back-transform the simulated y-space values to the simulated z-space values.

4.2 Directional Gradient-Curvature method

The Directional Gradient-Curvature (DGC) method, established by [Žukovič and Hristopulos \(2013a,b\)](#), is a novel tool for non-parametric conditional simulations designed for reconstructing missing grid data. It is based on the matching of the normalised squared gradient and

curvature of the sample and entire domain data using conditional Monte Carlo simulations. Direction-dependent information (e.g. non-stationarity, anisotropy) is captured by using the local information of the immediate neighborhood centered at each missing point. The DGC method can be used to model complex structured random fields without requiring a parametric formulation of it. In addition, the necessary user input is minimal.

In order to reduce the computational cost of the method, the sample's continuously valued field $Z(\mathbf{s}_s)$ is transformed to an integer valued indicator field $I_s \equiv I(\mathbf{s}_s)$ by means of threshold levels. Thus, the reconstruction of missing data is reduced to a spatial classification problem. The number of classes N_c is defined according to the desirable resolution and accuracy of the application. Generally, increasing N_c leads to more accurate estimations. In low resolution applications a small number of classes (i.e. $N_c = 8$) is sufficient. At the limit $N_c \rightarrow \infty$ the DGC method approximates continuous interpolation. Consequently, the reconstruction of the missing data is rendered equivalent to assigning a class label $I_g \equiv I(\mathbf{s}_g)$ at each point \mathbf{s}_g of the mapping grid.

The local square gradient and curvature terms in an arbitrary direction \mathbf{e}_n with corresponding lattice step α_n are given by

$$G_n(I; \mathbf{s}_i) = \frac{[I(\mathbf{s}_i + \alpha_n \mathbf{e}_n) - I(\mathbf{s}_i)]^2}{\alpha_n^2}, \quad n = 1, \dots, d, \quad (4.3)$$

$$C_n(I; \mathbf{s}_i) = \frac{[I(\mathbf{s}_i + \alpha_n \mathbf{e}_n) + I(\mathbf{s}_i - \alpha_n \mathbf{e}_n) - 2I(\mathbf{s}_i)]^2}{\alpha_n^4}, \quad n = 1, \dots, d. \quad (4.4)$$

The average of these terms, for each direction e_n , is taken over the grid, so as to normalise them.

The estimated indicator index of the grid nodes \hat{I}_g are derived through the minimization of the following objective function:

$$U(I_g | I_s) = \sum_{n=1}^d \left[w_1 \phi(\bar{G}_n(I_g), \bar{G}_n(I_s)) + w_2 \phi(\bar{C}_n(I_g), \bar{C}_n(I_s)) \right], \quad (4.5)$$

$$\phi(x, x') = \begin{cases} (1 - x/x')^2, & x' \neq 0 \\ x^2, & x' = 0 \end{cases}, \quad (4.6)$$

where $\bar{G}_n(I_g)$ and $\bar{G}_n(I_s)$ represent the normalized squared gradient and curvature, respectively, terms in one direction, defined as averages over the grid, w_1, w_2 are weights of the gradient and the curvature ($w_1, w_2 \geq 0$, $w_1 + w_2 = 1$), and d is the number of directions used.

The main steps of the DGC method can be summarized as follows:

1. Definition of the number of simulations M , the number of classes N_c , the size of the neighborhood, the residual cost function tolerance tol and the maximum number of Monte Carlo steps i_{max} (optional).
2. Discretization of $Z(s_s)$ to obtain the sample class identity field I_s .
3. Calculation of the sample directional normalised squared gradient and curvature $\bar{G}_n(I_g), \bar{G}_n(I_s), n = 1, \dots, d$.
4. For each simulation:
 - (a) Assignment of initial values $\hat{I}_g^{(0)}$ to the prediction points \mathbf{s}_g , by majority rule, based on the prevailing value of each point's sample neighbors. If no majority is reached up to the defined neighborhood, the initial value is assigned (i) randomly from the range of the labels with tie votes or (ii) from the entire range of labels $1, \dots, N_c$, if majority is not reached due to absence of sampling points within the neighborhood.
 - (b) Calculation of the initial energy values $\bar{G}_n(I_g^{(0)}), \bar{G}_n(I_s(0)), n = 1, \dots, d$, and the objective function $U^{(0)} = U(\hat{I}_g^{(0)} | I_s)$.
 - (c) Definition for each grid node of the state (amongst the possible states) that minimizes the objective function. An iterative process is followed starting from the initial state and generating a new state by randomly adding ± 1 to the previous. The updating of class identity states uses the "greedy" Monte Carlo (MC) method (Papadimitriou and Steiglitz, 1982). If the objective function of the new state is less than the one of the initial state, it is accepted else the initial state is kept and the next grid node is investigated.
5. Evaluation of the statistics from the simulations.

The Directional Gradient-Curvature (DGC) model is inspired by Spartan spatial random fields (SSRF) (Hristopulos and Elogne, 2007), which are based on short-range interactions between the field values. The DGC method do not strictly belong neither to the interpolation methods nor to the conditional simulation methods. Interpolation methods provide a single optimal estimation of the missing values. Conditional simulation sample the entire data and try to reconstruct the joint conditional probability density function of the missing data. On the other hand, DGC takes account only the data that corresponds to local minima of the objective function. The stochastic nature of the DGC derives from the multiple realizations that it returns (Žukovič and Hristopulos, 2013a).

Chapter 5

Data Analysis

Real data processing in Geostatistics is a demanding problem and consists of many successive steps. There is not a standard procedure applying to every data set, but each researcher can follow different way depending on the data and the final purpose of the analysis.

In Geostatistics every data set is considered as a single realization of a random field, summed with a trend model and an independent gaussian fluctuation, i.e.

$$Z(\mathbf{s}) = m_z(\mathbf{s}) + Z'(\mathbf{s}) + \varepsilon(\mathbf{s}), \quad (5.1)$$

where $Z(\mathbf{s})$ the total random field which coincides with the real data at the sample's points, $m_z(\mathbf{s})$ the global trend model, $Z'(\mathbf{s})$ a random field with zero mean value corresponding to the fluctuations (also called residuals) of the field $Z(\mathbf{s})$ around the trend and $\varepsilon(\mathbf{s})$ the error.

It is common practice in geostatistical analysis to initially evaluate and remove the trend model and then process the residuals. More specific, a general and widely accepted procedure for data analysis includes the following indicative steps (Goovaerts, 1997):

1. Preliminary Analysis
2. Parameter Inference
3. Model Selection
4. Spatial Prediction
5. Performance Assessment

In the following sections a brief description of each step is presented.

5.1 Preliminary Analysis

Preliminary analysis of data aims to examine some basic statistical properties before any other analysis tool is applied.

Normality

To assess whether a data set comes from a normal (Gaussian) distribution, often the normal probability plot of it is examined. This plot displays the theoretical quantiles of a normal distribution (x axis) versus the experimental quantiles of the studied data (y axis). Therefore, the normality of the data set can be assessed visually; if the resulting plot coincides with a theoretical straight line (corresponding with the normal distribution) or deviates to a small extend from it, the data is deemed Gaussian. Normal probability plot can also be used to identify and remove outliers.

In the case of non-gaussian data set suitable transformations have to be implemented to it, so as to transform the data to a new data set with approximately normal distribution. The most widely used transformation is the Box-Cox Power transformation (Box and Cox, 1964). The one parametric Box-Cox transformation for a variable x is defined as:

$$x^{(\lambda)} = \begin{cases} \frac{x^\lambda - 1}{\lambda}, & \lambda \neq 0 \\ \ln(x), & \lambda = 0 \end{cases}, \quad (5.2)$$

and the two parametric Box-Cox transformation, which can also handle negative values of x , is defined as:

$$x^{(\lambda_1, \lambda_2)} = \begin{cases} \frac{(x + \lambda_2)^{\lambda_1} - 1}{\lambda_1}, & \lambda_1 \neq 0 \\ \ln(x + \lambda_2), & \lambda_1 = 0 \end{cases}. \quad (5.3)$$

The optimum parameters λ_1, λ_2 are usually calculated by maximizing the Log-Likelihood Function (LLF), or minimizing the negative LLF (see also section 5.2.1. Note that Box-Cox transformation cannot restore the normality for each data set. In such cases, other more complex non-linear transformations may have to be applied. A common practice is to assume gaussianity of the transformed dataset after applying the Box-Cox transformation in order to avoid such complex transformations.

Trend Removal

The trend function $m_z(\mathbf{s})$, which represents the deterministic part of the random field $Z(\mathbf{s})$, is usually modelled by low-order polynomials of the coordinations of the sample's points. Alternatively, if the data exhibits periodicity a combination of the linear with the periodic model (sinusoidal) can be used. In Table 5.1 are shown some common trend functions for 2D cases.

Table 5.1 Common trend models

Model	Trend Function(2D)
Mean	$m_z(\mathbf{s}) = a_0$
Linear	$m_z(\mathbf{s}) = a_0 + a_1x + a_2y$
Quadratic	$m_z(\mathbf{s}) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2$
Cubic	$m_z(\mathbf{s}) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 + a_6x^2y + a_7xy^2 + a_7x^3 + a_8y^3$
Quartic	$m_z(\mathbf{s}) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 + a_6x^2y + a_7xy^2 + a_8x^3 + a_9y^3$ $+ a_{10}x^2y^2 + a_{11}x^3y + a_{12}xy^3 + a_{13}x^4 + a_{14}y^4$
Linear+Periodic	$m_z(\mathbf{s}) = a_0 + a_1x + a_2y + \sum_{j=1}^n \{a_{4(j-1)+3} \cos 2\pi f_{x,j}x + a_{4(j-1)+4} \sin 2\pi f_{x,j}x$ $+ a_{4(j-1)+5} \cos 2\pi f_{y,j}y + a_{4(j-1)+6} \sin 2\pi f_{y,j}y\}$

The linear coefficients a_i are usually estimated by means of *Multiple Linear Regression (MLR)*. The MLR method estimates the coefficients a_i of the independent variables, also called regressors, of a linear model of order p (in this case the coordinations x_i and y_i) that give the least square error between the investigated model and the dependent variable or regressand ($Z(\mathbf{s}_i)$) by solving the linear equations system:

$$\mathbf{Z}(\mathbf{s}_i) = \mathbf{C}a, \quad (5.4)$$

$$\text{where } \mathbf{Z}(\mathbf{s}_i) = \begin{bmatrix} Z(\mathbf{s}_1) \\ Z(\mathbf{s}_2) \\ \vdots \\ Z(\mathbf{s}_n) \end{bmatrix}, a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}, \text{ and } \mathbf{C} = \begin{bmatrix} x_1^{k=0}y_1^{l=0} & x_1^{k=1}y_1^{l=0} & x_1^{k=0}y_1^{l=1} & \dots & x_1^k y_1^l \\ x_2^{k=0}y_2^{l=0} & x_2^{k=1}y_2^{l=0} & x_2^{k=0}y_2^{l=1} & \dots & x_2^k y_2^l \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_n^{k=0}y_n^{l=0} & x_n^{k=1}y_n^{l=0} & x_n^{k=0}y_n^{l=1} & \dots & x_n^k y_n^l \end{bmatrix}, \quad (k+l \leq p).$$

The dominant frequencies of the data, $f_{x,i}, f_{y,i}$, in x and y directions, respectively, for the periodic model are estimated by means of *Fast Fourier Transformation (FFT)*. The squaring of the FFT of the data results to the magnitudes of the frequencies in the spectral domain. Consequently, these magnitudes and the corresponding frequencies are sorted in descending order of the magnitude, for both x and y directions. The first elements of the resulting frequencies lists (the number of which is defined by the modeler) are the desirable quantities.

The selection of the best trend model is done by means of Information Criteria, e.g. Akaike Information Criterion (AIC), corrected Akaike Information Criterion (AICc), Bayesian Information Criterion (BIC) (Akaike, 1974; Schwarz, 1978), or Least Squared Error (LSE).

5.2 Parameter Inference

In the next step of the data analysis, a number of theoretical models for examination is chosen and the optimum parameters of them $\boldsymbol{\theta}_i = [\theta_1, \theta_2, \dots, \theta_n]_i$, which give the best fitting to the data, are estimated. The estimation of these parameters is achieved by minimizing an objective functional, which typically matches an experimentally calculated term from the available data with the corresponding theoretical term of the investigated model. The most widely used methods for the parameter inference are: i) the Covariance-Variogram Fitting (CVF), ii) the Maximum Likelihood Estimation (MLE) and iii) the Method of Moments (MoM).

It is also possible to estimate some of the model's parameters with different methods and then use one of the above methods to estimate the remaining parameters. This is usually

applied in the case of anisotropic data, where the parameters of anisotropy are firstly estimated. In this way, the computational cost of the optimization steps is reduced significantly and the reliability of the results is improved. The classic Directional CVF (DCVF) method and the more modern Covariance Hessian Identity (CHI) method are some methods for estimating the parameters of anisotropy from a data set.

5.2.1 Common Methods of Parameter Inference

Covariance-Variogram Fitting (CVF)

Covariance-Variogram Fitting (CVF) estimates the optimum parameters by minimizing one error function between the experimentally calculated covariance or variogram and the corresponding theoretical quantity, which derive from the investigated model. The error functions can have various expressions. For example it can be the net sum of the squared error for the variogram fitting:

$$f_{er}(\theta) = \sum_{k=1}^{N_c} [\hat{\gamma}_z(\mathbf{s}_{c,k}) - \gamma_z(\mathbf{s}_{c,k}; \theta)]^2, \quad (5.5)$$

the sum of the squared error weighted with the corresponding theoretical variogram value:

$$f_{er}(\theta) = \sum_{k=1}^{N_c} \frac{[\hat{\gamma}_z(\mathbf{s}_{c,k}) - \gamma_z(\mathbf{s}_{c,k}; \theta)]^2}{\gamma_z^2(\mathbf{s}_{c,k}; \theta)}, \quad (5.6)$$

the the sum of the squared error divided by the corresponding number of pairs:

$$f_{er}(\theta) = \sum_{k=1}^{N_c} \frac{[\hat{\gamma}_z(\mathbf{s}_{c,k}) - \gamma_z(\mathbf{s}_{c,k}; \theta)]^2}{n(\mathbf{s}_{c,k})}, \quad (5.7)$$

or the combination of the last two errors:

$$f_{er}(\theta) = \sum_{k=1}^{N_c} \frac{1}{n(\mathbf{s}_{c,k})} \frac{[\hat{\gamma}_z(\mathbf{s}_{c,k}) - \gamma_z(\mathbf{s}_{c,k}; \theta)]^2}{\gamma_z^2(\mathbf{s}_{c,k}; \theta)}, \quad (5.8)$$

where $\hat{\gamma}_z(\mathbf{s}_{c,k})$ the experimental variogram values at the centers of the cyclical sections (see section 2.6), $\gamma_z(\mathbf{s}_{c,k}; \theta)$ the theoretical variogram values at the centers of the cyclical sections given by the investigated model with parameters θ , N_c the number of cyclical sections, and $n(\mathbf{s}_{c,k})$ the number of pairs of data included into the k th cyclical section. Similar equations can be derived for the case of covariance fitting.

In practice, the angular sections are allocated between the semicycle and the separation vectors \mathbf{h} of the pairs corresponding to diametrically opposite directions, i.e. to angles ϕ and $\phi + \pi$, are considered as identical. This follows from the ellipsoid representation of anisotropy. Also, note that for the minimization of the error the rotation angle ϕ is restricted between $-\pi/2$ and $\pi/2$ in order to avoid equivalent solutions (Olea, 2006).

Maximum Likelihood Estimation (MLE)

The Maximum Likelihood Estimation, proposed by Fisher (1997) estimates optimum parameters by maximizing the likelihood the known realization (real data) can be produced by a given parameter set. In practice, this is achieved by minimizing the negative log-likelihood function (NLLF). The basic concepts of the MLE method as described by Pardo-Igúzquiza (1998) are briefly represented below.

The joint pdf of n experimental multivariate Gaussian data with zero mean can be expressed as:

$$f_z(\mathbf{z}; \boldsymbol{\theta}) = (2\pi)^{-n/2} |\mathbf{C}|^{-n/2} \exp\left\{-\frac{1}{2} \mathbf{z} \mathbf{C}^{-1} \mathbf{z}\right\}, \quad (5.9)$$

where n the number of experimental data, \mathbf{C} the $n \times n$ covariance matrix of the data, $|\cdot|$ denotes the determinant, and $\boldsymbol{\theta}$ the $m \times 1$ vector of parameters that defines the covariance matrix.

The NLLF of the data given the parameters $\boldsymbol{\theta}$, then, is expressed as:

$$L(\sigma_z^2, \boldsymbol{\theta}'; \mathbf{z}) = \frac{n}{2} \ln(2\pi) + n \ln(\sigma) + \frac{1}{2} \ln(|A|) + \frac{1}{2\sigma_z^2} \mathbf{z} \mathbf{A}^{-1} \mathbf{z}, \quad (5.10)$$

where $\boldsymbol{\theta}'$ the covariance parameters without the variance, and $\mathbf{A} = \mathbf{C}/\sigma_z^2$ the correlation matrix.

Replacing in Eq. (5.10) the ML estimate of the variance:

$$\hat{\sigma}_z^2 = \frac{1}{n} \mathbf{z} \mathbf{A}^{-1} \mathbf{z}, \quad (5.11)$$

the NLLF becomes:

$$L(\hat{\sigma}_z^2, \boldsymbol{\theta}'; \mathbf{z}) = \frac{n}{2} (\ln(2\pi) + 1 - \ln(n)) + \frac{1}{2} \ln(|A|) + \frac{n}{2} \mathbf{z} \mathbf{A}^{-1} \mathbf{z}. \quad (5.12)$$

The covariance parameters $\boldsymbol{\theta}'$ are estimated as the values that minimize the Eq. (5.12), while the variance $\hat{\sigma}_z^2$ is subsequently estimated according to Eq. (5.11).

Method of Moments (MoM)

The Method of Moments (MoM) estimates the parameters of a model by relating the sample moments to the parameters of interest and solving the resulting system (Bowman and Shenton, 2004). More specifically the k unknown parameters $\theta_1, \theta_2, \dots, \theta_k$ defining the pdf $f_z(z; \theta)$ of a random variable Z are estimated as the solution of the system consisting of the k first moments of the variable, which can be expressed as functions of θ , i.e.

$$\begin{aligned}\hat{\mu}_1 &\equiv \mathbb{E}[z(\mathbf{s}_i)] = \frac{1}{n} \sum_{i=1}^n z(\mathbf{s}_i) = g_1(\theta) \\ \hat{\mu}_2 &\equiv \mathbb{E}[z^2(\mathbf{s}_i)] = \frac{1}{n} \sum_{i=1}^n z^2(\mathbf{s}_i) = g_2(\theta) \\ &\vdots \\ \hat{\mu}_k &\equiv \mathbb{E}[z^k(\mathbf{s}_i)] = \frac{1}{n} \sum_{i=1}^n z^k(\mathbf{s}_i) = g_k(\theta).\end{aligned}\tag{5.13}$$

5.2.2 Estimation of Anisotropy

Directional Covariance-Variogram Fitting (DCVF)

Directional CVF is a method for estimating the parameters of anisotropy (ϕ, ξ_1, ξ_2) of a field. In this method, like in the CVF, the space is divided into cyclical sections defined from angular and distance limits, and the experimental variogram corresponding to these sections are computed from the data, referred to the centers $(\phi_{c,i}, \xi_{c,i})$ of them. However, in this case, the theoretical model, in its isotropic form, is fitted separately to each of the j number of angular sections. Thus, j number of parameter vectors θ , corresponding to the j angular sections, and consequently same number of (ξ, ϕ) pairs are obtained. The estimation of the anisotropy parameters follows from the fitting of an ellipse to these pairs. The lengths of the ellipsis major axes correspond to the (ξ_1, ξ_2) parameters, while the angle between the first met, while rotating counter-clockwise, major axis of the theoretical ellipsis and the horizontal plane corresponds to the rotation angle.

Covariance Hessian Identity (CHI)

The CHI offers a fast, non-parametric and non-iterative method for estimating the anisotropic parameters of a 2D random field (Chorti and Hristopulos, 2008; Hristopulos, 2002; Petrakis, 2012; Petrakis and Hristopulos, 2012). It is based on linking the second-order derivatives of the covariance function with the ensemble average of the *Gradient Kronecker Product*

(GKP) of the data. More specifically, according to [Swerling \(1962\)](#), the CHI of the covariance function and the sample-estimated expectation of GKP are related via the following expression:

$$Q_{ij} \equiv \mathbb{E} \left[\frac{\partial Z(\mathbf{s})}{\partial s_i} \frac{\partial Z(\mathbf{s})}{\partial s_j} \right] = \frac{\partial^2 c_z(\mathbf{h})}{\partial h_i \partial h_j} \Big|_{h=(0,0)}, \quad (5.14)$$

where Q_{ij} represents a tensor. Obviously, this is valid only for differentiable random fields. For non-differentiable randoms fields numerical approximations of the derivatives can be estimated ([Chorti and Hristopulos, 2008](#)).

In the case of a 2D, differentiable and homogeneous random field, the explicit equations for the elements of the CHI derive by expressing the covariance function in terms of the ratio of the correlation lengths $R = \xi_1/\xi_2$ and the rotation angle ϕ :

$$Q_{11} = \frac{\sigma_z^2 \zeta^2}{\xi_1^2} \left(\cos^2 \phi + R^2 \sin^2 \phi \right), \quad (5.15)$$

$$Q_{22} = \frac{\sigma_z^2 \zeta^2}{\xi_1^2} \left(R^2 \cos^2 \phi + \sin^2 \phi \right), \quad (5.16)$$

$$Q_{12} = Q_{21} = \frac{\sigma_z^2 \zeta^2}{\xi_1^2} \left[\sin \phi \cos \phi (1 - R^2) \right]. \quad (5.17)$$

The anisotropic parameters R and ϕ are then estimated by minimizing the following objective function:

$$F(\phi, R) = \left[\frac{Q_{22}}{Q_{11}} - \frac{R^2 + \tan^2 \phi}{1 + R^2 \tan^2 \phi} \right]^2 + \left[\frac{Q_{12}}{Q_{11}} - \frac{\tan^2 \phi (1 - R^2)}{1 + R^2 \tan^2 \phi} \right]^2. \quad (5.18)$$

The constants ζ and ξ_1 remain undetermined from the optimization of the objective function. The first one is eliminated by the division, thus its value doesn't need to be determined. The second is usually of interest and can be determined from the experimental variogram.

5.3 Model Selection

After estimating the optimum parameters of a number of covariance models, an assessment of their predictive accuracy is required in order to select one model, this with the best performance, for the following steps of data analysis. This is usually achieved by means of *Cross-Validation (CV)*.

Cross-Validation is a technique for providing information about how an estimated model will perform in a real problem. In general, CV is performed by partitioning a sample dataset into two subsets, called *training subset* and *validation subset*. The first subset is used as input for the estimation of the missing values of the second one. The estimator in this step is the same as the one which is intended to be applied in the next step of Spatial Prediction.

5.3.1 Common Types of Cross-Validation

The most commonly used types of CV is the exhaustive *Leave-p-out Cross-Validation (LpOCV)* and the non-exhaustive *k-fold Cross-Validation (k-fold CV)*. Exhaustive cross-validation methods compute all possible ways to divide the original sample into a training and a validation set (Geisser, 1975, 1993; Seni and Elder, 2010).

Leave-p-out Cross-Validation (LpOCV)

In LpOCV the validation set consists of p sample points of the sample dataset, while the remaining realizations are used as training set. The cross-validation process is repeated for every possible combination of p number of realizations and the results are then averaged to provide a single estimation. This method may lead to high computational cost. For this reason it is preferred in cases where the sample dataset is relatively small.

A particular case of LpOCV with $p = 1$, also called *Leave One Out Cross-Validation (LOOCV)* is preferred, due to its simplicity and its lower computational cost.

k-fold Cross-Validation (k-fold CV)

In k-fold CV the sample dataset is randomly divided into k equal sized subsets. It is apparent that this method requires a larger dataset than LOOCV. Then the CV process is repeated k times (the folds) using each time one of the k subsets as validation set and the remaining $k - 1$ subsets as training set. The results, like to LpOCV, are also averaged to give the final estimation. The simplest variation of k-fold CV is the one with $k = 2$ (2-fold CV), while in the case of $k = n$ (the size of the sample dataset) the k-fold CV is exactly the LOOCV.

5.3.2 Measures of Cross-Validation Performance

Common validation measures

In order to assess the performance of the cross-validation, it is useful to compute and compare validation measures for each model. Such measures are the Mean Absolute Error

(*MAE*), the Maximum Absolute Error (*MxAE*), the Mean Squared Error (*MSE*), the Root Mean Squared Error (*RMSE*), the Pearson's correlation coefficient (ρ) and the Spearman's correlation coefficient (r_S). The above mentioned errors measure the accuracy and precision of the predictions, while the Pearson's and Spearman's correlation coefficients provide a measure of the linear and non-linear, respectively the relationship between the real and the predicted data. These measures are calculated by the following equations:

$$MnAE = \frac{1}{N} \sum_{i=1}^N |\hat{z}(\mathbf{s}_i) - z(\mathbf{s}_i)|, \quad (5.19)$$

$$MxAE = \max(|\hat{z}(\mathbf{s}_i) - z(\mathbf{s}_i)|), \quad (5.20)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N [\hat{z}(\mathbf{s}_i) - z(\mathbf{s}_i)]^2, \quad (5.21)$$

$$RMSE = \frac{1}{N} \sqrt{\sum_{i=1}^N [\hat{z}(\mathbf{s}_i) - z(\mathbf{s}_i)]^2}, \quad (5.22)$$

$$\bar{\rho}_{Z,\hat{Z}} = \frac{\sum_{i=1}^N [z(\mathbf{s}_i) - \overline{z(\mathbf{s})}] [\hat{z}(\mathbf{s}_i) - \overline{\hat{z}(\mathbf{s})}]}{\sqrt{\sum_{i=1}^N [z(\mathbf{s}_i) - \overline{z(\mathbf{s})}]^2} \sqrt{\sum_{i=1}^N [\hat{z}(\mathbf{s}_i) - \overline{\hat{z}(\mathbf{s})}]^2}}, \quad (5.23)$$

$$r_S = 1 - \frac{\sum_{i=1}^N (R_{z_i} - R_{\hat{z}_i})^2}{N(N^2 - 1)}, \quad (5.24)$$

where R_{z_i} denotes the rank of $z(\mathbf{s}_i)$ among all $z(\mathbf{s})$ values. The rank is computed by sorting the z values in ascending order; the rank of a given value is equal to its order of appearance in the sorted list.

These measures can be compared individually or to be combined to one kind of a coefficient, at the rate of which will the final choice be made. In this way, a more meritocratic measure that also enables a more automatic selection of a model derives. For instance, this coefficient could be of the form:

$$r_F = \frac{1}{relMSE_i^2} \rho_i r_{S,i} \in [0, 1], \quad (5.25)$$

where i corresponds to the investigated models and $relMSE_i = MSE_i / \min(MSE_i) > 1$ is the model's relative MSE. The $relMSE_i$ in the denominator is squared so as to equalize the product of the two correlation coefficients in the numerator. As the performance of the

investigated model improves, the correlation coefficients increase and the MSE decreases. As the correlation coefficients increase so does the numerator of the r_F , converging to 1. As the MSE decreases, the $relMSE$ and consequently the denominator of the r_F also decreases converging to 1. Thus, as the accuracy of the investigated model improves, the final coefficient increases converging to 1.

Classification measures

Except from the above validation measures it is often useful to calculate some classification measures. This requires converting the continuum valued field into a discrete one by separating the continuum values into n equal bins and assigning n integer indicators to each one.

Discretizing both the original and the estimated values of the field, the performance of the classification of the unknown values can be measured by calculating:

- the Pearson's and Spearman's correlation coefficients of the original and estimated indicators of the missing values, and
- the *misclassification rate* (MCR), i.e. the percent of the estimations that are not equal to the corresponding original indicators.

These measures can be estimated for both classification methods (e.g. DGC simulation method) and interpolation methods (e.g. Ordinary Kriging).

5.4 Spatial Prediction

In this step the desirable estimation method is applied in order to predict the missing values and to produce a map of the study area. Widely used estimation methods are the Kriging and the Stochastic Simulations, which have been described in Chapters 3 and 4, respectively.

The parametric estimation methods (e.g. Ordinary Kriging and Turning Bands Simulation) require as input the best model defined in the steps of Parameter inference and Model Selection. On the contrary, the non-parametric method of DGC simulation, does not need a pre-defined model, i.e. the steps 2 and 3 (Parameter inference and Model Selection) are skipped.

The uncertainty of the estimations is also evaluated in this step, usually by means of confidence intervals at some confidence level. The confidence interval at 95% confidence level of the estimations in the case of Ordinary Kriging is calculated as:

$$(\hat{z}(\mathbf{s}_i) - 1.96\sigma_{E,OK}(\mathbf{s}_i), \hat{z}(\mathbf{s}_i) + 1.96\sigma_{E,OK}(\mathbf{s}_i)). \quad (5.26)$$

Chapter 6

Case Study: Marmousi Model

6.1 Description of the Data

6.1.1 Geologic Profile of the Data

The purpose of this thesis is the exploratory implementation of geostatistical tools for the simulation of geological media. The tools which have been used are: i) Ordinary Kriging and ii) DGC simulation method. For the implementation of Kriging software was developed in MATLAB programming environment, while for the implementation of Directional Gradient Curvature (DGC) simulation method software developed by the Geostatistics Laboratory (Technical University of Crete) is used. Several variations for parameter inference (parametric-non parametric) and anisotropy tackling (transform data to isotropic or not) have been used in OK.

More explicitly, a known gridded dataset is sampled, both randomly and regularly, and the reconstruction of it is subsequently attempted with the above mentioned methods. The original dataset is the Marmousi Model, a synthetic 2D acoustic model created in 1988 by the Institute Francais du Petrole (IFP), and used for the workshop on practical aspects of seismic data inversion at the 1990 EAEG meeting in Copenhagen. It was generated using a 2-D acoustic finite-difference modeling program, so as to resemble an overall continental drift geological setting. The geometry of the Marmousi is based on a profile through the North Quenguela trough in the Cuanza basin. The Cuanza basin is in northwestern Angola on the Atlantic Coast of West Africa and is about 300 km long north-south and 170 km wide east-west (Fig. 6.1).

The geological model of the basin consists, from top to bottom, of:

- Surface and subsurface sediments of Lower and Upper Cretaceous, Paleocene, Eocene, and Miocene strata.

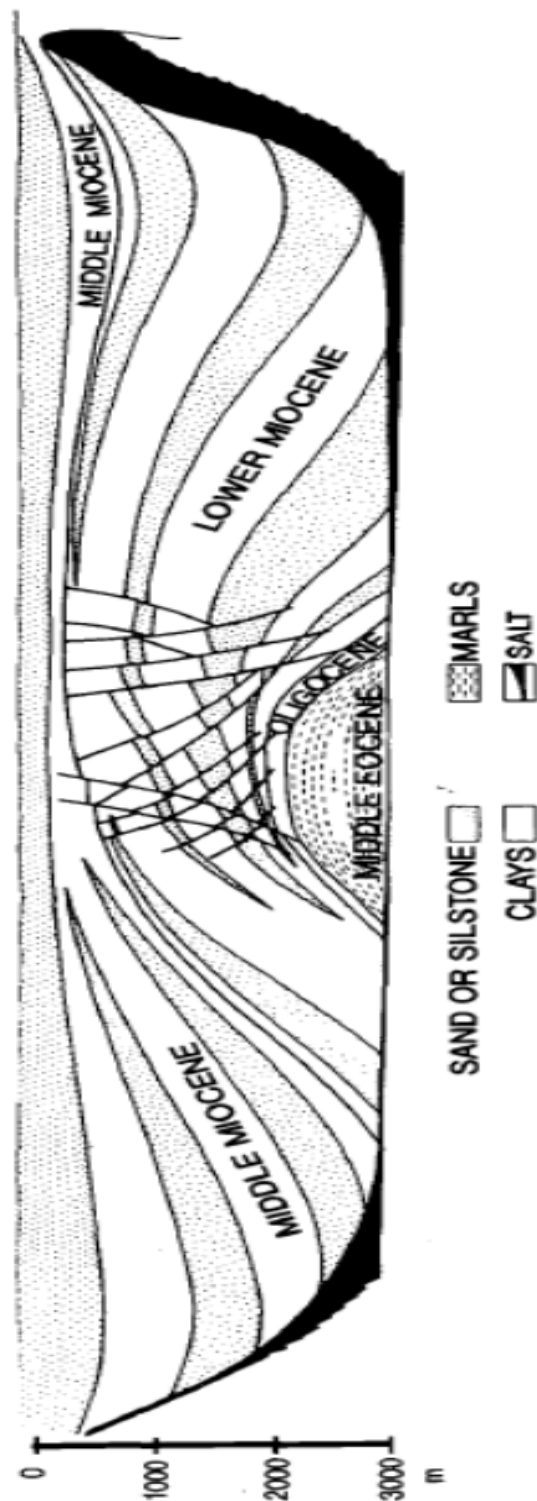


Figure 6.1 Profile through the North Quenguela trough in the Cuanza basin (Angola) after Verrier and Branco (1972)

- Early Cretaceous carbonate-evaporite (mainly salt) sequence and a Late Cretaceous and Tertiary argillaceous-arenaceous sequence.
- Precambrian crystalline basement, which is partly covered by extrusive rocks and granite-wash type sediments.

The post-Aptian (Early Cretaceous) tectonic process of the West Africa Salt Basin, has affected significantly the salt movements into the basin. As a result of the basinward tilt of the margin, the carbonate stratas of the Early Cretaceous and the overlying Upper Cretaceous sequence slid on the salt layer and broke up into smaller blocks. This process caused a mobilization of the salt and the formation of salt rollers, salt diapirs and a whole suite of raft tectonics structures such as turtlebacks, carbonate rafts and severe folding at the toe of the salt basin.

Occurrences of oil and gas have been reported in almost all of the stratigraphic units in the Cuanza basin, and there is major production from the Cretaceous rocks. These hydrocarbon occurrences are expected to be entrapped into the folded carbonate and salt structures of the basin, due to the nature of the basement and the salt tectonics (Brognon and Verrier, 1966; Spathopoulos, 1996).

Based on this profile a geometric model was created using the MIMIC™ module of the SIERRA package. Then this model was transformed into a 2-D velocity/density grid (Fig. 6.2). The Marmousi model contains 158 horizontally layered horizons. Numerous large normal faults and tilted blocks, resulting from the continental drift, complicates the model towards its center. The model sits under approximately 32 m of water and is 9.2 km in length and 3 km in depth. The target zone is a reservoir located at a depth of about 2.5 km. The model contains many reflectors, steep dips, and strong velocity variations in both the lateral and the vertical direction (with a minimum velocity of 1500 m/s and a maximum of 5500 m/s) (Bourgeois et al., 1990; Irons, 2008; Versteeg and Grau, 1990).

6.1.2 Basic Notions and Assumptions for the Data

The data consists a 122x384 grid, i.e. 46848 values in total. This means that each cell of the grid corresponds to a 24,60x23,96 m (depth x length) space. The information mentioned in section 6.1.1 as well as the visualisation of the data (Fig. 6.2), indicate that the studied dataset is highly complexed and possibly non-stationary, as at least the anisotropy characteristics change significantly over the space. The geological section can be separated into three vertical bands. The left and right bands exhibit anisotropy which can be described by an ellipsis with an almost horizontal major axis, while in the central band the major axis turns to about 40°. For simplicity, in this thesis the field is arbitrarily assumed stationary and

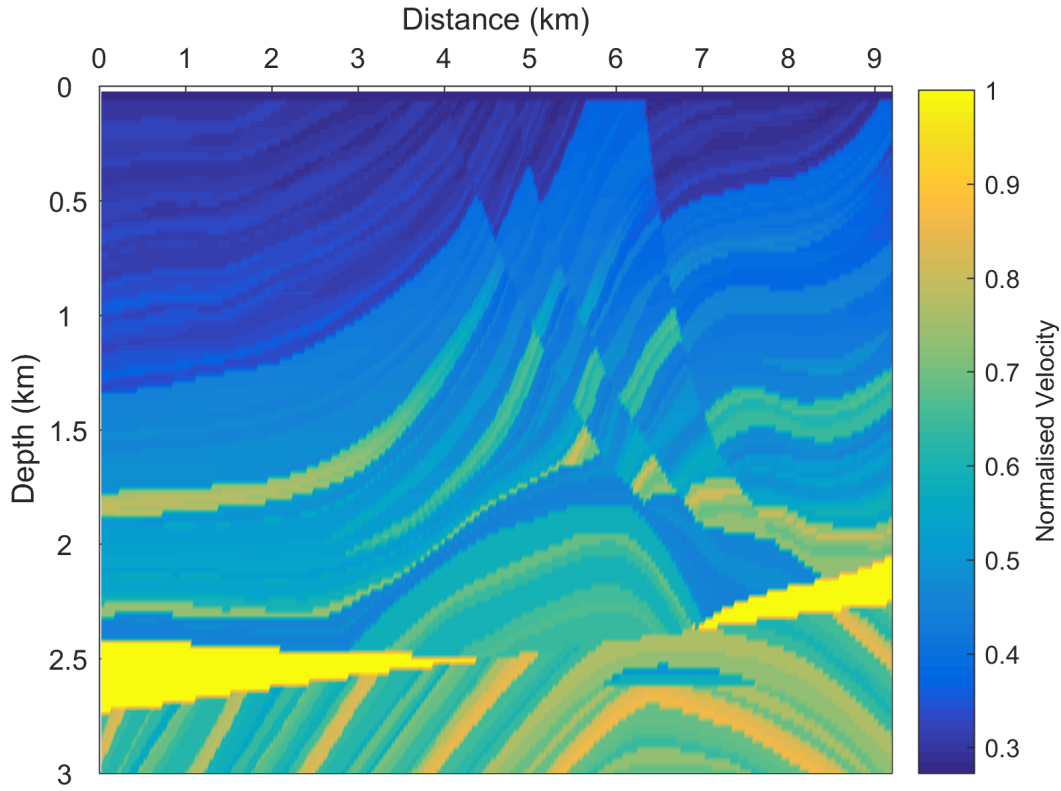
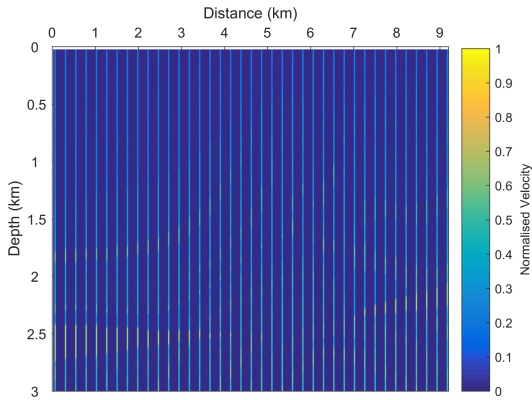


Figure 6.2 Normalised Marmousi Model (Velocity Model)

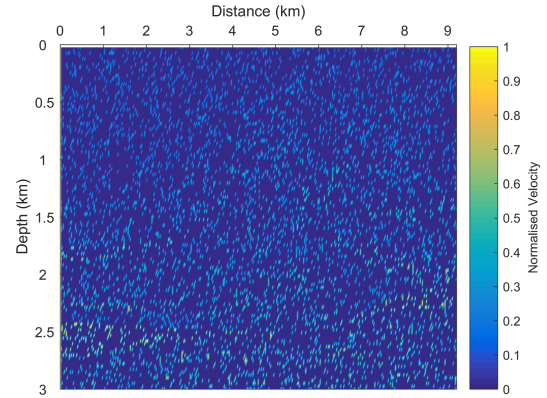
is modelled as such. However, the reliability and the performance of the final models are evaluated in the light of this complexity.

As mentioned above, two samples of the Marmousi Model are used, shown in Fig. 6.3. Both of the samples contain 4758 points, which correspond a 10,16% of the original data. The first sample consists of 39 columns of the grid, starting from the 3rd and selecting one per 10 columns until the end of the grid. The geometry of this sample resembles, in some way, to data obtained by well-logging drill-holes (which in this case is impractical as the distance of the sample's drill-holes corresponds only to 240 m). Hereafter, this sample is called regular. The second sample is sampled randomly in the 2D space.

In order to decrease the computational cost we normalise the original values dividing with the maximum of them (5500 m/s). In this way, the resulting values belong to $[0,1]$. For the same reasons, the coordinations of the data points used are not the real coordinations but the (i, j) indices of the 122×384 grid's nodes. The i index correspond to the depth of the point, while j index correspond to the distance alongside the geological section. This means that the i 's take values from 1 to 122, with the 1 corresponding to the horizontal edge of the grid closer to the surface, and the j 's take values from 1 to 384, with the 1 corresponding to the eastern vertical edge of the grid.



(a) Regular Sample: 4758 points (10,16% of the original data) containing 39 columns of the grid (starting from the 3rd and selecting one per 10 columns until the end of the grid.)



(b) Random Sample: 4758 points (10,16% of the original data) selected randomly from the original. data

Figure 6.3 Samples of original data

6.2 Methodology

Ordinary Kriging

For the implementation of Ordinary Kriging to the samples of Marmousi Model, the general procedure described in Chapter 5 is followed. The indicative steps of this procedure are the following:

1. Preliminary Analysis
2. Parameter Inference
3. Model Selection (LOOCV)
4. Spatial Prediction
5. Performance Assessment

For the analysis procedure the following variogram models have been used: 1) Generalized Exponential, 2) Gaussian, 3) Spherical, 4) Generalized Matern, 5) Spartan(3D).

In the step of Parameter Inference, five variations are used. These methods differed regarding anisotropy estimation (i.e. parametric versus non parametric) and data transformation (transform data to isotropic or not). A concise description of each variation, which are arbitrarily named, follows (see also Table 6.1):

- **DirVar0:** A priori anisotropy parameter inference is not attempted. Thus, the anisotropic variogram functions, which contain the maximum number of unknown parameters for each model, are used in the optimization procedure of the 2nd step as well as in the following steps.

Table 6.1 Description of the five variations of Parameter Inference step. DVF: Directional Variogram Fitting, CHI: Covariance Hessian Identity (P.I.: Parameter Inference).

Variation	Anisotropy P.I. Method	Anisotropic Inversion
DirVar0	—	NO
DirVar1	DVF	NO
DirVar2	DVF	YES
CHI1	CHI	NO
CHI2	CHI	YES

- **DirVar1:** Anisotropy parameter inference is initially applied for each model by means of Directional Variogram Fitting (DVF) (see section 5.2.2). Subsequently, the estimated parameters are inserted to the corresponding anisotropic variogram functions, reducing by three the number of unknown parameters for each model. The new anisotropic variogram functions (with lower degrees of freedom) are used in the optimization procedure of the 2nd step as well as in the following steps.
- **DirVar2:** Anisotropy parameter inference is initially applied for each model by means of DVF. Subsequently, the estimated parameters are used in order to inverse anisotropic effect (see section 2.4.3) for each model separately. The step of *Parameter Inference* is then repeated using this time the isotropic variogram functions for each model. The isotropic variogram functions are used in the 2nd optimization procedure as well as in the following steps.
- **CHI1:** Anisotropy parameter inference is initially applied for each model by means of CHI (see section 5.2.2). Subsequently, the estimated parameters are inserted to the corresponding anisotropic variogram functions, reducing by three the number of unknown parameters for each model. The new anisotropic variogram functions (with lower degrees of freedom) are used in the optimization procedure of the 2nd step as well as in the following steps.
- **CHI2:** Anisotropy parameter inference is initially applied for each model by means of CHI. Subsequently, the estimated parameters are used in order to inverse anisotropic effect for each model separately. The step of *Parameter Inference* is then repeated using this time the isotropic variogram functions for each model. The isotropic variogram functions are used in the 2nd optimization procedure as well as in the following steps.

Also, the type of error used for the optimization in the step of *Parameter Inference* is the error function defined in Eq. (5.8).

Finally, in the Model Selection step where Leave-One-Out-Cross-Validation is applied the immediate neighbors are not taken into account in order to avoid overestimating the models. The risk of overestimation holds as OK generally assigns higher weights to the nearest neighbors and lower weights to the more distant ones. In the case of the regular sample, this means that for each estimation the data taken into account is all the columns except from the one containing the studied missing point.

Software for Ordinary Kriging

The above variations have been implemented to the investigated data by using software developed in the MATLAB programming environment. Specifically, 6 main functions were developed for:

- i) detrending 2D data,
- ii) calculating experimental anisotropic or isotropic variogram,
- iii) fitting theoretical variogram model to the experimental variogram (parameter inference),
- iv) estimating the anisotropic parameters of a random field with the directional variograms fitting method (DVF),
- v) performing cross validation, and
- vi) estimating the missing values with ordinary kriging, respectively.

Some auxiliary functions, for the computation of the covariance and variogram function and the construction of random fields, were also developed. These functions, along with a complete example of their application, are available in <https://github.com/billisandr/geostats2D>. For software testing purposes the above described analysis is implemented to a random sample of a synthetic dataset presented in section 6.3.

Software developed by D. T. Hristopulos for the estimation of 2D anisotropy parameters with the CHI method, which is available in <http://www.geostatistics.tuc.gr/index.php?id=5677>.

DGC Simulation method

The DGC Simulation method (section 4.2) is implemented to the two samples of Marmousi Model in an one-step procedure (the step of *Spatial Prediction*). For this method, software developed by Milan Žukovič and D. T. Hristopulos (which is currently unavailable online) was used.

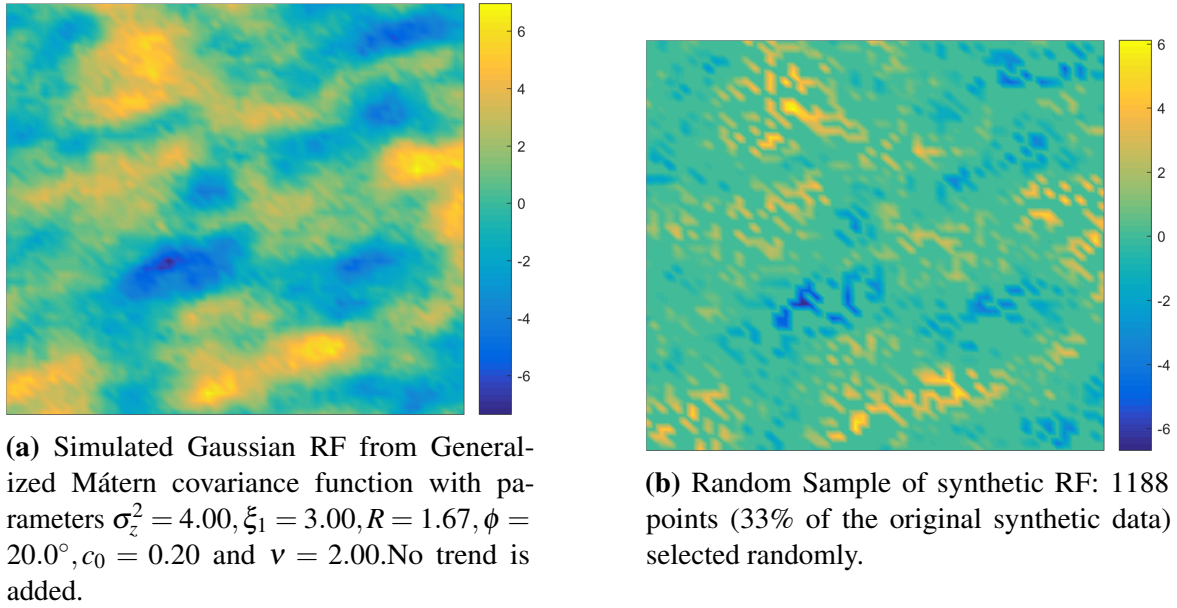


Figure 6.4 Synthetic RF and random sample

6.3 Synthetic Data Test

In this section, is presented the implementation of the developed functions to synthetic data so as to test the performance of them. Two variations of the analysis procedure, *DirVar0* and *DirVar1* (see Section 6.2), are used for the geostatistical analysis of synthetic data sampled randomly from a known constructed random field. The data are simulated from a Gaussian RF using a Generalized Matérn covariance function with parameters $\sigma_z^2 = 4.00$, $\xi_1 = 3.00$, $R = 1.67$, $\phi = 20.0^\circ$, $c_0 = 0.20$ and $\nu = 2.00$ (without adding any trend). The size of the simulated random field is 60x60 and the sample consists of the 33% of the it (i.e. 1188 points of 3600). The simulated RF and the sample are shown in Fig. 6.4.

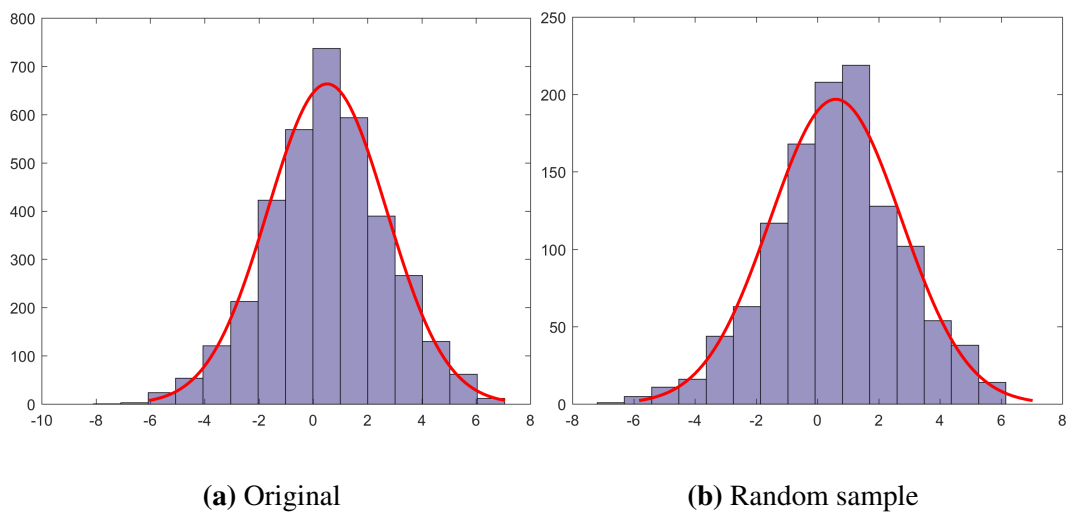
6.3.1 Preliminary Analysis

The statistics (Table 6.2) and the histograms (Fig. 6.5) of the sample and the original dataset show that the sample is representative of the original data, as well as that both of the datasets come from a gaussian distribution.

The normal probability plot of the sample (Fig. 6.6) also indicates that the data does come from a gaussian distribution. Moreover, the examined data are from structure gaussian and unbiased so there is no need for any transformations or trend removal.

Table 6.2 Original and sample datasets statistics

Dataset	Min	Max	Mean	Median	Variance	Skewness	Kurtosis
Original	-7.348	6.958	0.514	0.515	4.777	-0.074	3.020
Sample	-6.677	6.114	0.602	0.645	4.588	-0.164	3.121

**Figure 6.5** Histograms of original and sample datasets

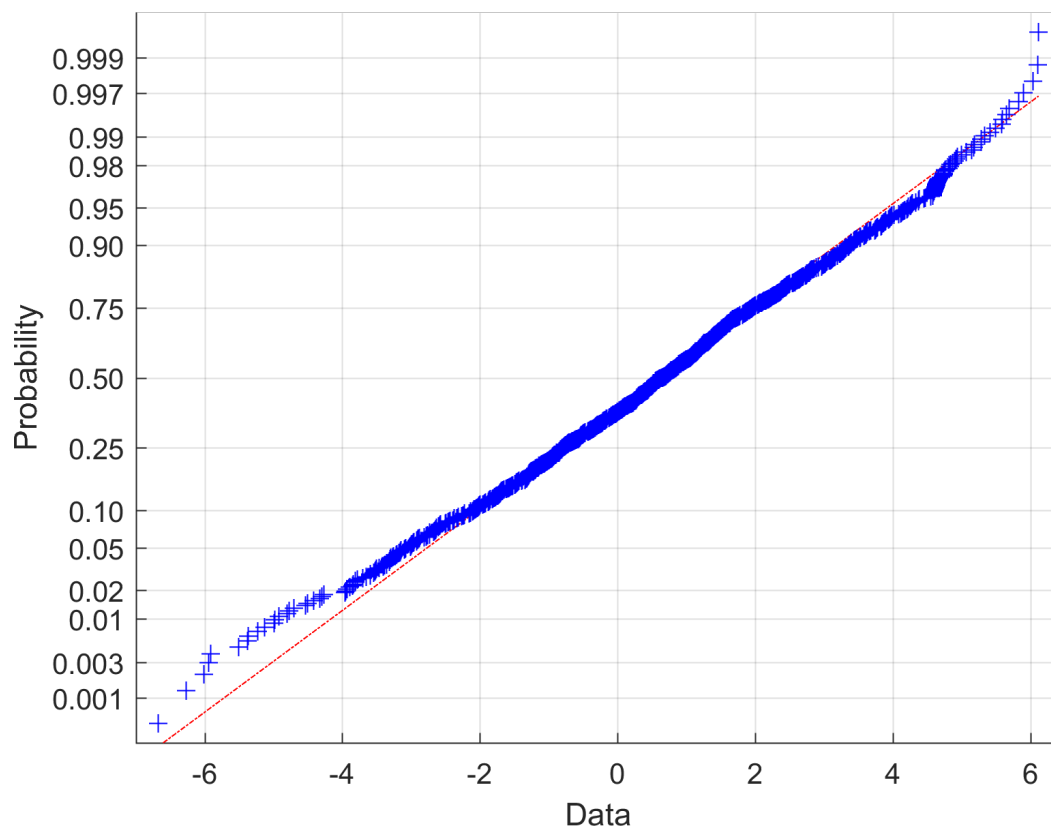


Figure 6.6 Normal probability plot of random sample

6.3.2 Calculation of Experimental Variogram

The experimental directional variograms with angular step 4° and 15° are shown in Fig. 6.7. These figures display clearly the anisotropic characteristics of the field. More specifically, it can be easily seen that the spatial correlation differs significantly from direction to direction; the variogram is stabilized around a maximum value (sill) in longer distances along the horizontal and almost horizontal directions than the other directions.

6.3.3 DirVar0

In the *DirVar0* method the investigated random field (transformed and detrended Marmousi model) is modeled by anisotropic variogram equations without estimating the anisotropy parameters in advance. Hence, the anisotropic variogram equations contain the maximum number of unknown parameters for each model. The parameter inference of these models is achieved through the minimization of the error function described by Eq. (5.8) between the experimental directional variograms and the chosen anisotropic theoretical models.

The initial values and the boundaries of the parameters for each model used in the optimization/parameter inference step are presented in Table 6.3, and the resulting optimum parameters are presented in Table 6.4. The estimated σ_z^2 , ξ_1 and ϕ for all the models except from Spartan are very close to the real, while the nugget effect (c_0) is estimated with good accuracy only from Generalized Exponential and Generalized Matérn models. From the estimated anisotropy ratios only the one deriving from the Generalized Matérn is closer to the real, while the other models give highly underestimated ratios. From the above derives that the most close to the reality model is the Generalized Matérn, which is the one used for the simulation of the synthetic data. The only erroneous estimation made from this model is the estimation of the smoothness parameter ν , which is underestimated.

The scores of the LOOCV, presented in Table 6.5, confirm the above observations as they clarify that the Generalized Matérn model has the best performance, followed by Spartan and Generalized Exponential.

A visualisation of the the selected theoretical model's fitting to the experimental variogram is interpreted in Fig. 6.8, in which the directional experimental variograms of the field on the directions $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° are plotted against the theoretical model (the rest directional variograms in directions $15^\circ, 45^\circ, 75^\circ, 105^\circ, 135^\circ$, and 165° can be found in Appendix A). It can be seen that the best anisotropic model fits well to the directional experimental variograms.

The estimation of the field values derived from OK with the best model is shown in Fig. 6.9. A scatter diagram of original and estimated values of the field, as well as their

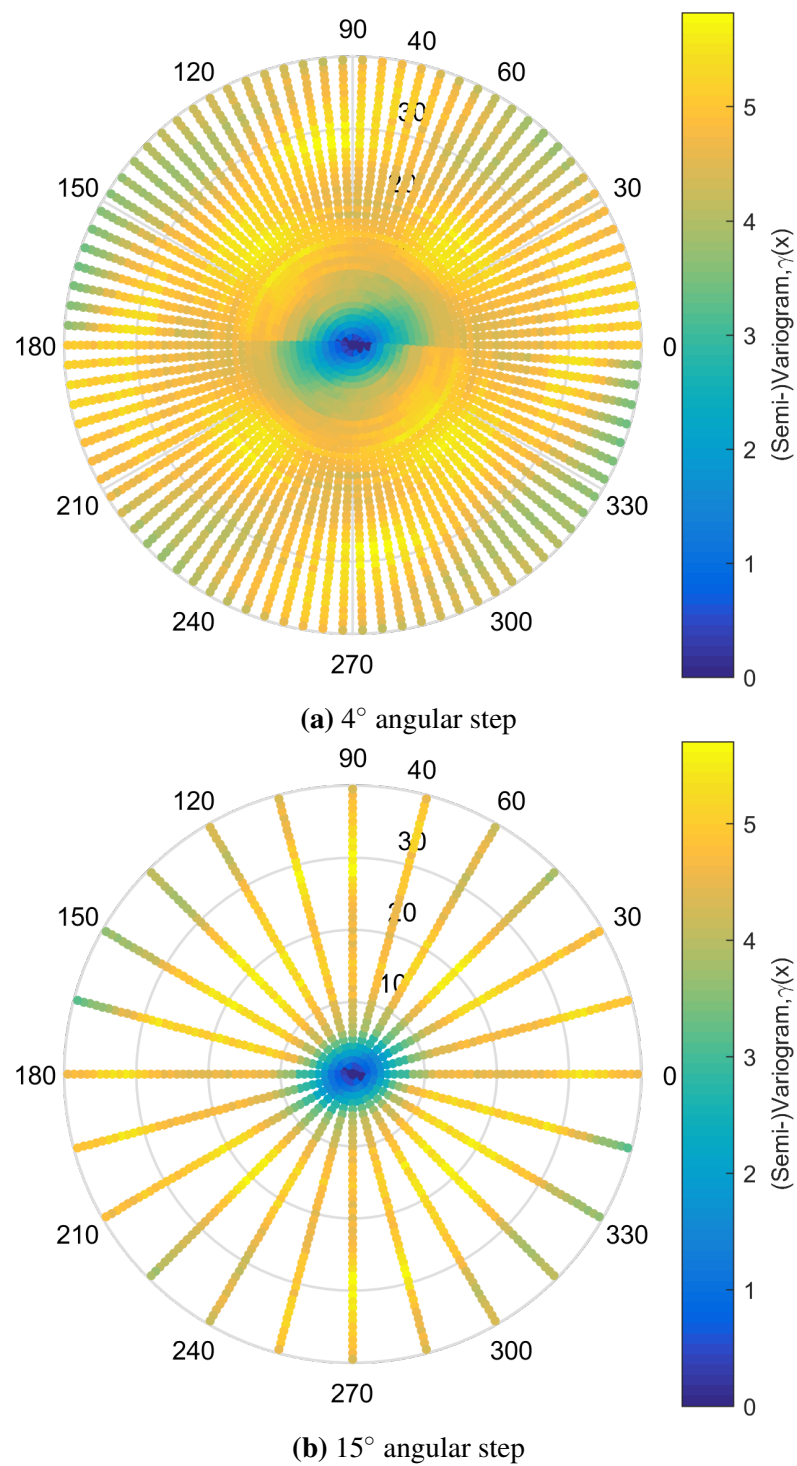


Figure 6.7 Directional experimental variograms of the synthetic data. For both cases the angular tolerance is 20°, the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 16, and is divided into 45 distance lags.

Table 6.3 Initial values and boundaries of the parameters for optimization step, where \hat{g}_{max} (\equiv maximum value of experimental variogram) = 5.7026, $d_{max} \simeq 16$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}1/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 = 1000, \xi_1 = d_{max}1/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in [0, d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$.

Model	Initial Values	Boundaries
Gen.		
Exponential	$[b, \nu = 1.5]$	$[b_b, \nu \in (0, 2)]$
Gaussian	$[b]$	$[b_b]$
Spherical	$[b]$	$[b_b]$
Gen. Matérn	$[b]$	$[b_b]$
Spartan	$[b_{sp}, \eta_1 = 1]$	$[b_{sp,b}, \eta_1 \in (-2, \infty)]$

Table 6.4 Optimum Parameters of the examined variogram models

Model	σ_z^2	ξ_1	R	ϕ	c_0	ν
Gen.						
Exponential	4.965	1.694	0.203	-65.2°	0.167	1.494
Gaussian	4.427	1.703	0.213	-66.0°	0.456	—
Spherical	4.840	1.519	0.095	-64.4°	0.000	—
Gen. Matérn	4.945	1.694	0.429	-65.2°	0.260	1.525
	η_0	ξ_1	R	ϕ	c_0	η_1
Spartan	165.129	0.625	0.081	-27.1°	0.000	2.000

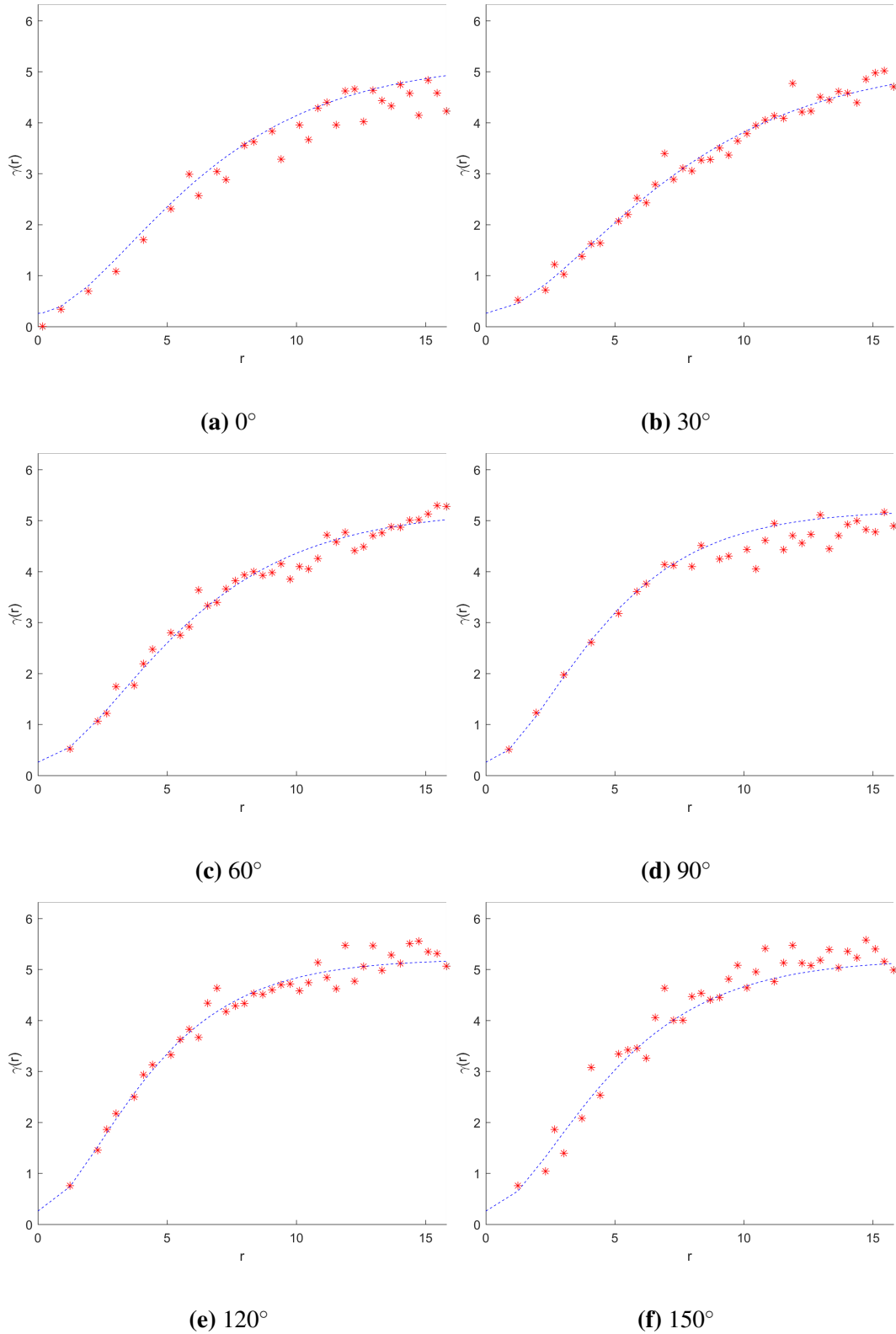
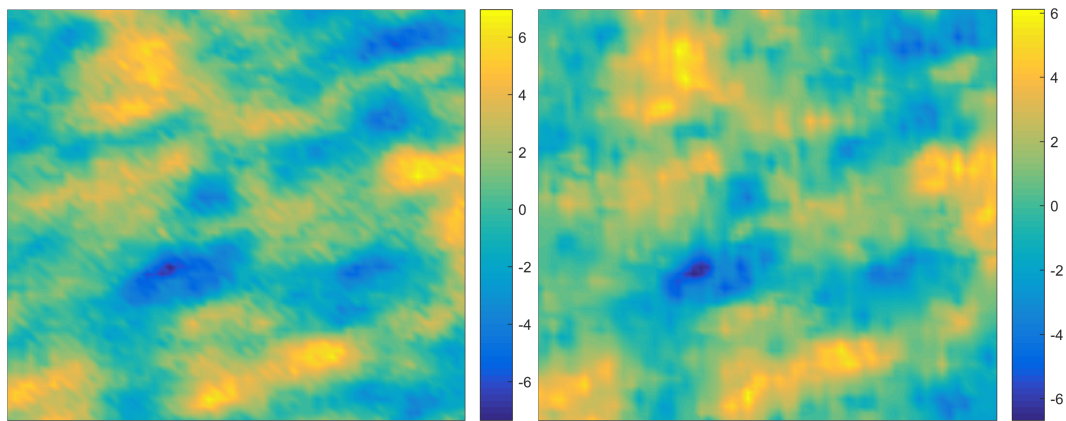


Figure 6.8 Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Gen. Matérn with parameters $\sigma_z^2 = 4.945$, $\xi_1 = 1.694$, $R = 0.429$, $\phi = -65.2^\circ$, $c_0 = 0.260$, $\nu = 1.525$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 16.

Table 6.5 Leave-One-Out Cross Validation Scores

Model	MeanAE	MaxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	1.171	4.790	2.217	1.489	0.736	0.721	0.156
Gaussian	1.171	4.780	2.216	1.489	0.736	0.721	0.156
Spherical	1.171	4.773	2.216	1.489	0.736	0.721	0.156
Gen. Matérn	0.855	3.566	1.201	1.096	0.885	0.866	0.766
Spartan	1.142	4.488	2.109	1.452	0.754	0.738	0.181

**(a)** Original Stochastic Component**(b)** Estimated Stochastic Component**Figure 6.9** Original and Estimation of the field. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 4.945$, $\xi_1 = 1.694$, $R = 0.429$, $\phi = -65.2^\circ$, $c_0 = 0.260$, $\nu = 1.525$.

histograms can be seen in Fig. 6.10. In general, the estimations follow the original values achieving a good proximity to the original distribution. In Table 6.6 the measures of the estimation performance for the three best models are presented. From these measures it can be observed that the first model's performance is much better than others two, which is almost identical. In addition, Fig. 6.11 maps the confidence interval width of the estimations, i.e. it depicts the spatial distribution of the uncertainty. The uncertainty is zero at the known points, while it increases up to 4.7 gradually with the distance of the missing points from the known locations.

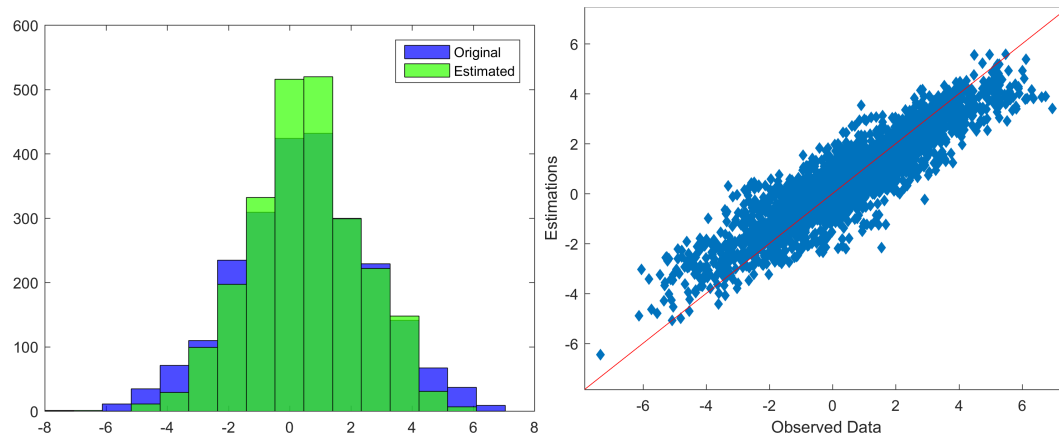


Figure 6.10 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Mátern with parameters $\sigma_z^2 = 4.945$, $\xi_1 = 1.694$, $R = 0.429$, $\phi = -65.2^\circ$, $c_0 = 0.260$, $\nu = 1.525$.

Table 6.6 Ordinary Kriging Scores

Model	MeanAE	MaxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Mátern	0.737	3.713	0.904	0.951	0.907	0.897	0.813
Spartan	1.194	5.198	2.239	1.496	0.752	0.743	0.091
Gen. Exponential	1.197	5.037	2.254	1.501	0.750	0.742	0.089

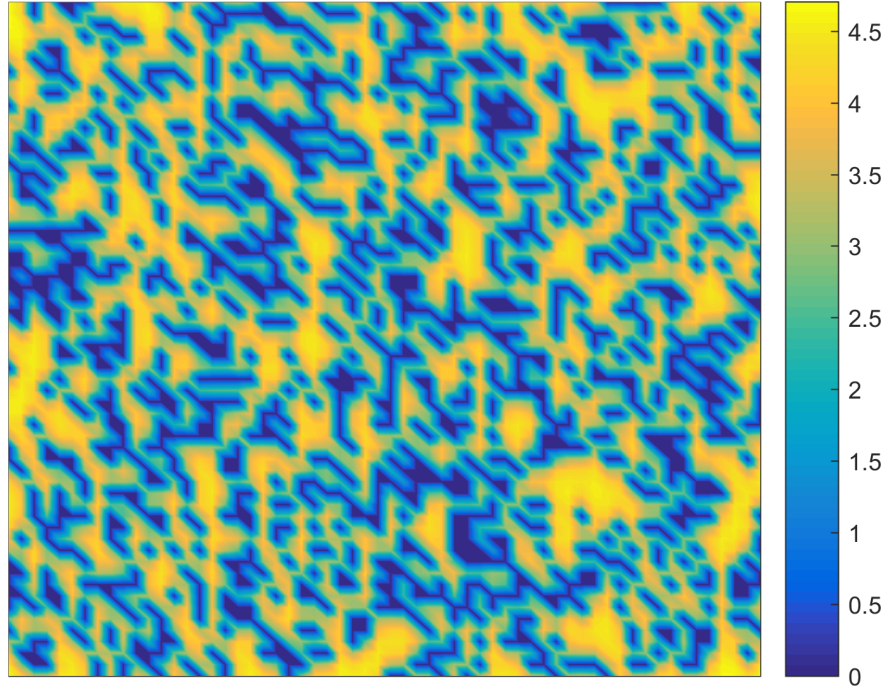


Figure 6.11 Uncertainty of the total field's estimation

6.3.4 DirVar1

In the *DirVar1* method the investigated random field is modeled by anisotropic variogram equations after estimating the anisotropy parameters of them. The anisotropy parameters are estimated by means of DVF (see section 5.2.2). Subsequently, the estimated parameters are replaced to the corresponding anisotropic variogram functions, reducing by three the number of unknown parameters for each model. The parameter inference of the models with the lower degrees of freedom is achieved through the minimization of the error function between the experimental directional variograms and the chosen anisotropic theoretical models.

The initial values and the boundaries of the parameters for each isotropic model used in the anisotropy parameter inference step with DVF method are as specified in Table 6.3 without considering the anisotropy parameters R and ϕ . The anisotropy parameters for each model are estimated by fitting an ellipse to the resulting pairs of (ϕ, ξ) of each model. The fitting of the ellipses to the pairs of (ϕ, ξ) of each model are depicted in Fig. 6.12, while the estimated anisotropy parameters for all models are given in Table 6.7.

All the investigated models, except from the Spartan, indicate that the major axis of the anisotropy ellipsis lies on between 0° and 47° . These estimations are relatively close to the used anisotropy angle of 20° for the construction of the synthetic random field. The Spartan model diverges significantly from the other models and gives the direction of the major

Table 6.7 Anisotropy parameters of the examined variogram models estimated with DVF method

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	4.867	0.563	-58.5°
Gaussian	4.874	0.653	-65.5°
Spherical	11.557	0.743	-90.0°
Gen. Matérn	1.846	0.342	-42.3°
Spartan	3.858	0.536	15.3°

Table 6.8 Optimum Parameters of the variogram models with the lower degrees of freedom

Model	σ_z^2	c_0	ν
Gen. Exponential	4.248	0.201	1.533
Gaussian	4.232	0.196	—
Spherical	4.285	0.162	—
Gen. Matérn	4.066	0.418	2.832
	η_0	c_0	η_1
Spartan	108.431	0.000	1.759

axis on 105° . This is probably due to minimization miscalculations caused by inappropriate objective function (very smooth or possible local minima), inappropriate initial values, etc. As for the estimated anisotropy ratios the Generalized Matérn converges to a lower value from the real, while the rest models converge to closer values. Finally, the minor axis of the ellipsis anisotropy is best estimated by the Generalized Matérn while the other models give much higher values and especially the Spherical. Anyway, more reliable conclusions can be drawn after the cross validation procedure.

By replacing the estimated anisotropy parameters to the anisotropic variogram models and minimizing the error function (Eq. (5.8)) of the "new" models and the experimental directional variograms the rest parameters are estimated, as presented in Table 6.8.

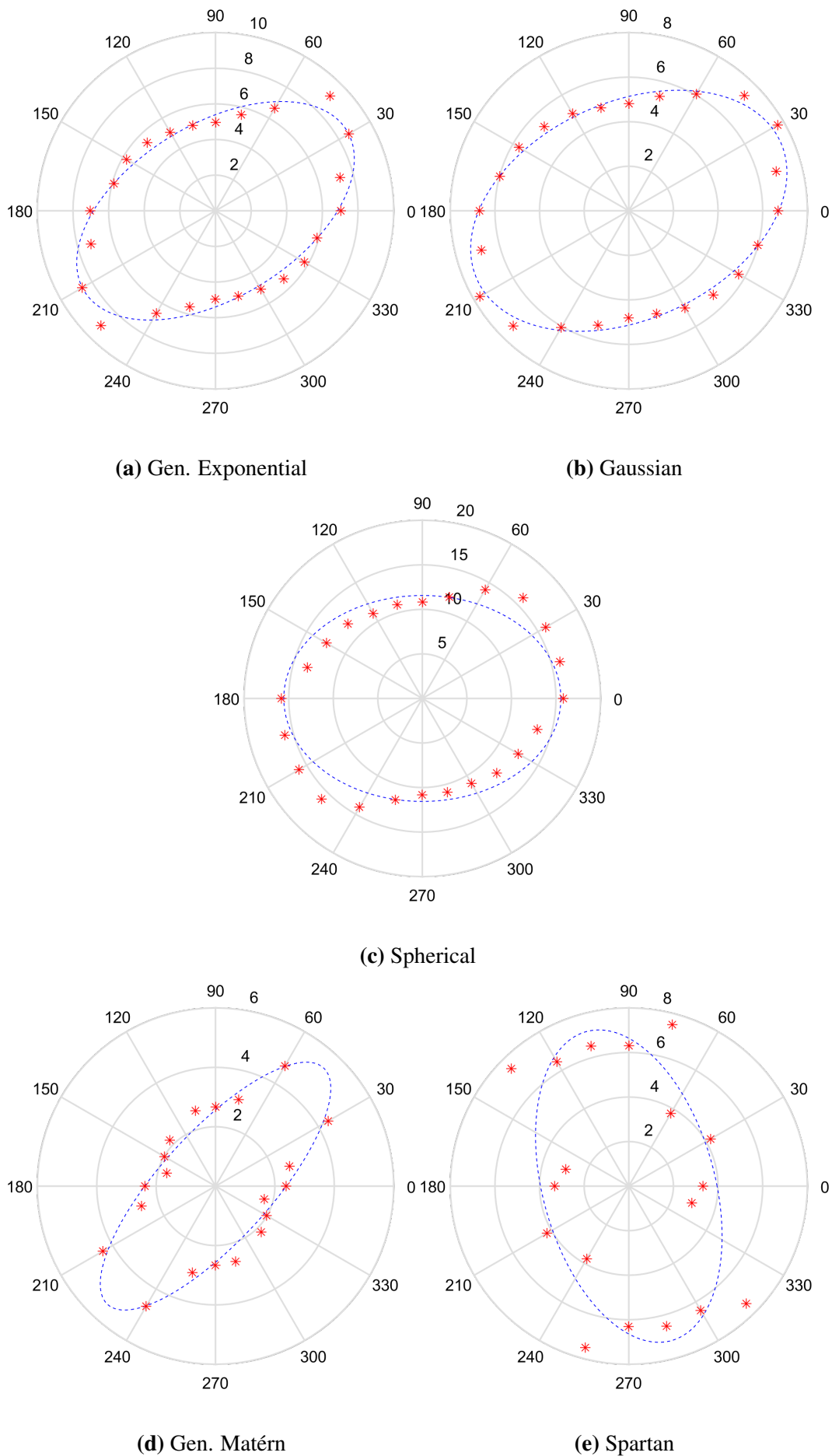


Figure 6.12 Fitting of ellipses to the pairs (ϕ, ξ) of the examined variogram models

Table 6.9 Leave-One-Out Cross Validation Scores

Model	MeanAE	MaxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	1.079	4.319	1.888	1.374	0.794	0.773	0.228
Gaussian	1.098	4.316	1.948	1.396	0.785	0.764	0.209
Spherical	1.275	5.642	2.626	1.621	0.664	0.654	0.083
Gen. Matérn	0.812	7.452	1.149	1.072	0.873	0.868	0.758
Spartan	13.851	4177.341	16826.381	129.717	0.034	0.296	0.000

After the parameter inference Leave-One-Out Cross Validation (LOOCV) is applied in order to define the best models. The scores of the LOOCV, presented in Table 6.9, give as best model the Generalized Matérn, followed by the Generalized Exponential and the Gaussian.

The fitting of the best model to the experimental variogram along the directions of $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° is interpreted in Fig. 6.13 (the rest directional variograms in directions $15^\circ, 45^\circ, 75^\circ, 105^\circ, 135^\circ$, and 165° can be found in Appendix A). As it can be seen the best theoretical variogram model fits well to the experimental directional variograms.

Implementing OK with the determined best model, the resulting estimation of the field is as shown in Fig. 6.14. The scatter diagram and histograms of the original and the estimated values are illustrated in Fig. 6.15, and the measures of the estimation performance for the three best models are presented in Table 6.10. In general, the estimations reproduce the original field quite well as it be seen from the histograms. As for the models' performance the Gen. Matérn achieved significantly higher measures than the other two which have similar measures. Finally, Fig. 6.16 shows the spatial distribution of the ordinary kriging estimations uncertainty. The uncertainty is zero at the known points, while it increases up to 4.7 gradually with the distance of the missing points from the known points.

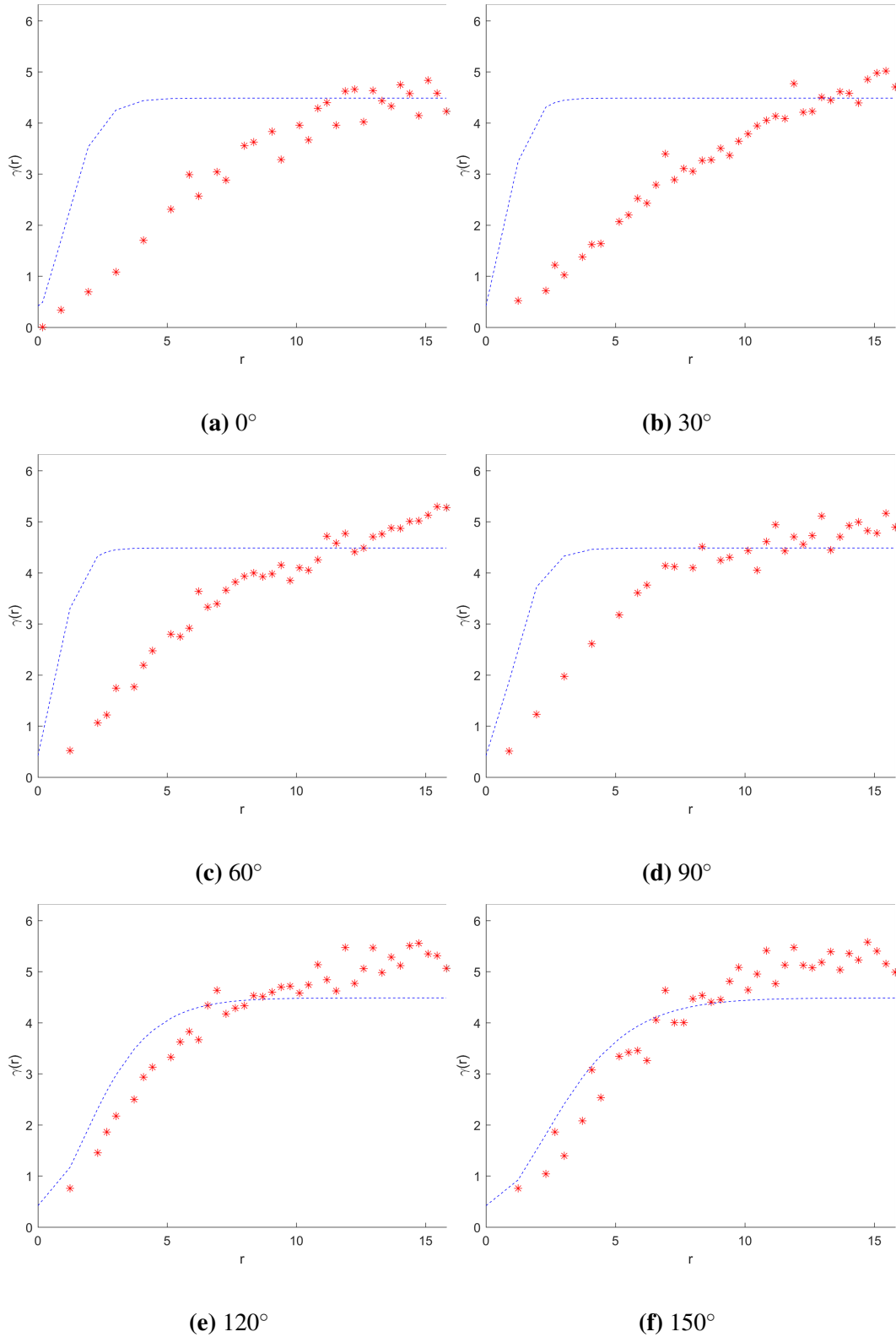


Figure 6.13 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Matérn with parameters $\sigma_z^2 = 4.066$, $\xi_1 = 1.846$, $R = 0.432$, $\phi = -42.3^\circ$, $c_0 = 0.418$, $\nu = 2.832$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

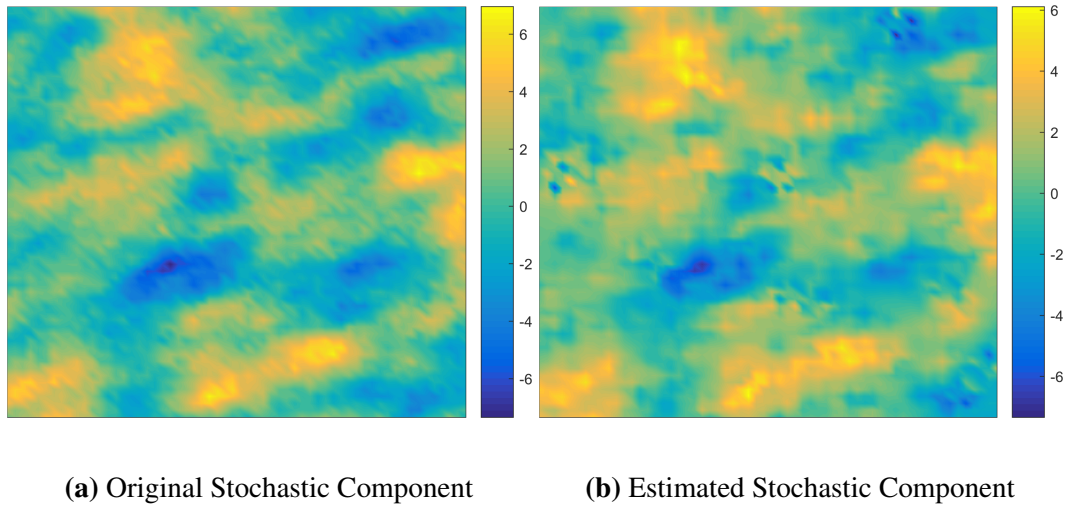


Figure 6.14 Original and Estimation of the stochastic component of the field. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 4.066$, $\xi_1 = 1.846$, $R = 0.432$, $\phi = -42.3^\circ$, $c_0 = 0.418$, $\nu = 2.832$.

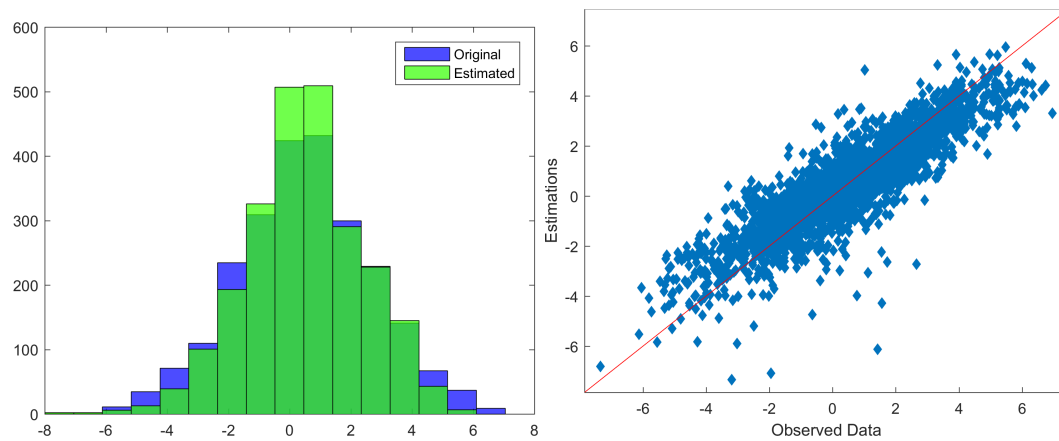


Figure 6.15 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 4.066$, $\xi_1 = 1.846$, $R = 0.432$, $\phi = -42.3^\circ$, $c_0 = 0.418$, $\nu = 2.832$.

Table 6.10 Ordinary Kriging Scores

Model	MeanAE	MaxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Matérn	0.798	7.558	1.115	1.056	0.878	0.878	0.771
Gen. Exponential	0.952	4.324	1.489	1.220	0.852	0.841	0.402
Gaussian	0.903	4.331	1.357	1.165	0.857	0.849	0.491

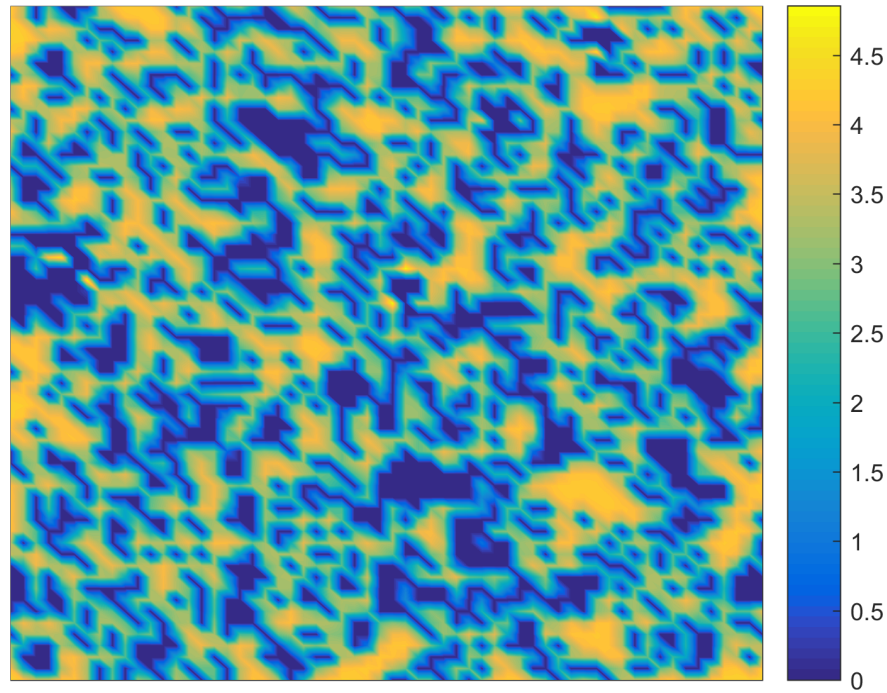


Figure 6.16 Uncertainty of the total field's estimation

6.3.5 Synopsis

The above described analysis of the simple synthetic dataset indicates that the developed codes can handle successively such datasets and provide relatively reliable and accurate results.

In the next sections these codes are applied to a more complex dataset.

6.4 Regular Sample of Marmousi Model

6.4.1 Ordinary Kriging

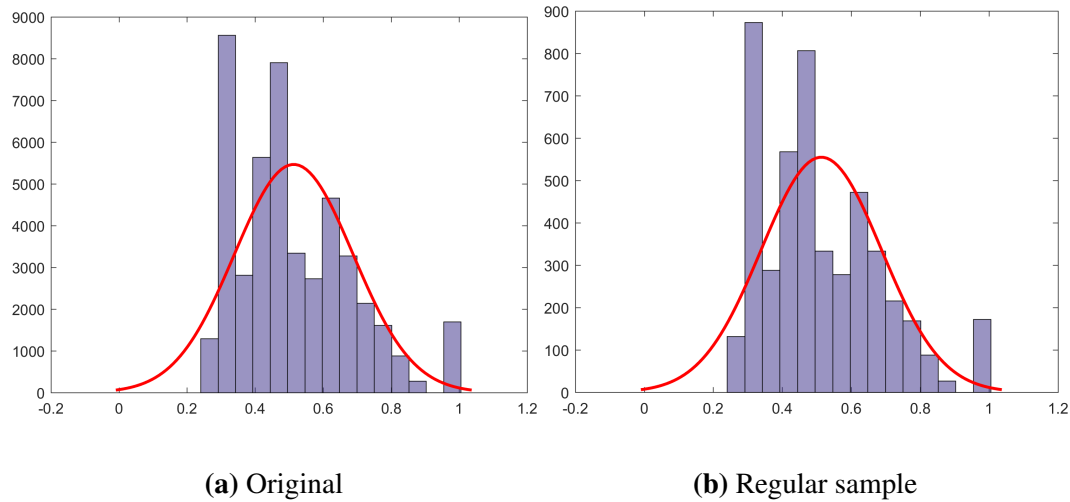
Preliminary Analysis

It is essential to examine whether the sample is representative of original data. From the statistics of both the sample and the original datasets (Table 6.11) as well as the histograms of them (Fig. 6.17), it is clear that the sample is totally representative of the original dataset.

The normal probability plot of the sample (Fig. 6.18), however, indicates that the data does not come from a gaussian distribution. Thus, Box-Cox transformation is necessary to be applied to the sample dataset. The best Box-Cox transformation is attained when $\lambda = -0.2606$. The statistics of the transformed values can be seen in Table 6.12, while the histogram and the normal probability plot of them are presented in Fig. 6.19). It is

Table 6.11 Original and sample datasets statistics

Dataset	Min	Max	Mean	Median	Variance	Skewness	Kurtosis
Original	0.273	1.000	0.514	0.475	0.030	0.824	3.277
Sample	0.273	1.000	0.514	0.473	0.030	0.823	3.270

**Figure 6.17** Histograms of original and sample datasets

clear that the Box-Cox transformation does not provides a truly gaussian dataset, but the transformed dataset is closer to the gaussian distribution than the original sample. Thus, no further actions are taken and the analysis proceeds under the assumption that the transformed dataset resembles sufficiently a dataset obtained from a gaussian distribution.

After transforming the regular sample's values, the next step is the removal of any possible trend. The trend models described in Section 5.1 (see Table 5.1) have been tested. The parameters of these models are inferred by means of multilinear regression of the transformed normalised velocities field on the (i, j) indices of the sample points. The complete expressions of the resulting trend functions are given in Table 6.13, while the normal probability plots of the fluctuations of each model are shown in Fig. 6.20. The best model is choosen as the one resulting in fluctuations which are closer to the gaussian distribution. This model, as can be seen from Fig. 6.20, is the linear trend model.

Table 6.12 Transformed (Box-Cox with $\lambda = -0.2606$) dataset statistics

Min	Max	Mean	Median	Variance	Skewness	Kurtosis
-1.546	0.000	-0.810	-0.826	0.157	0.025	2.127

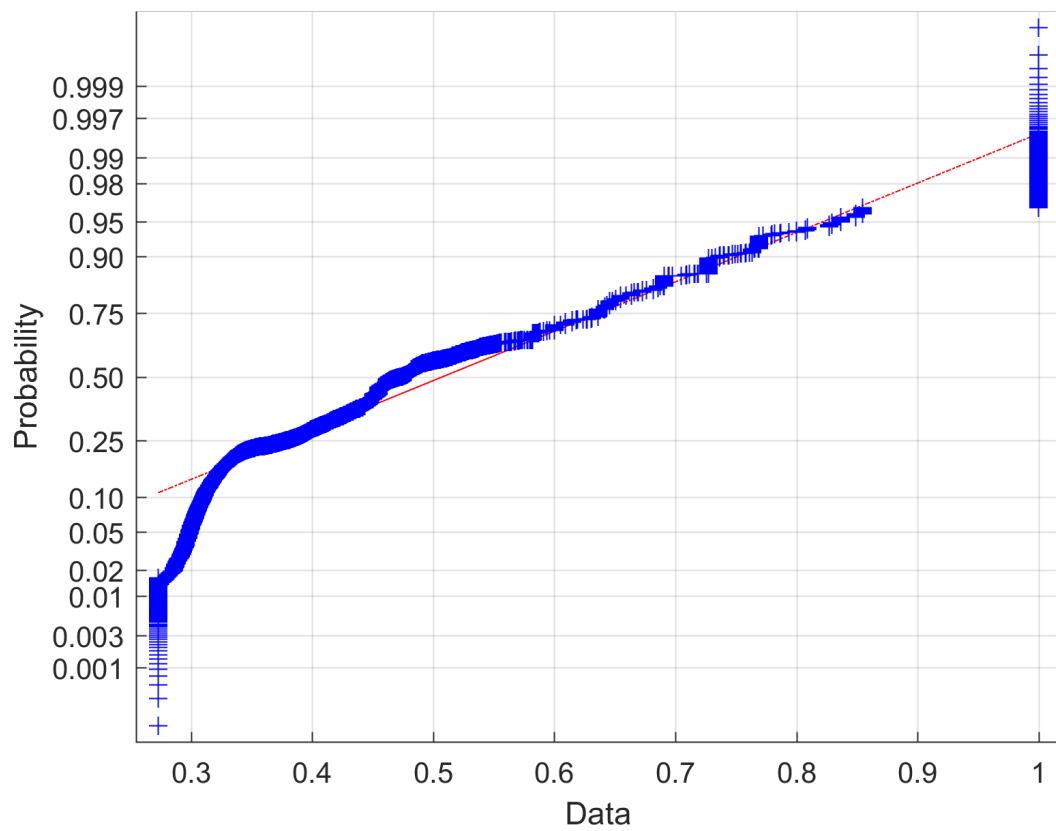
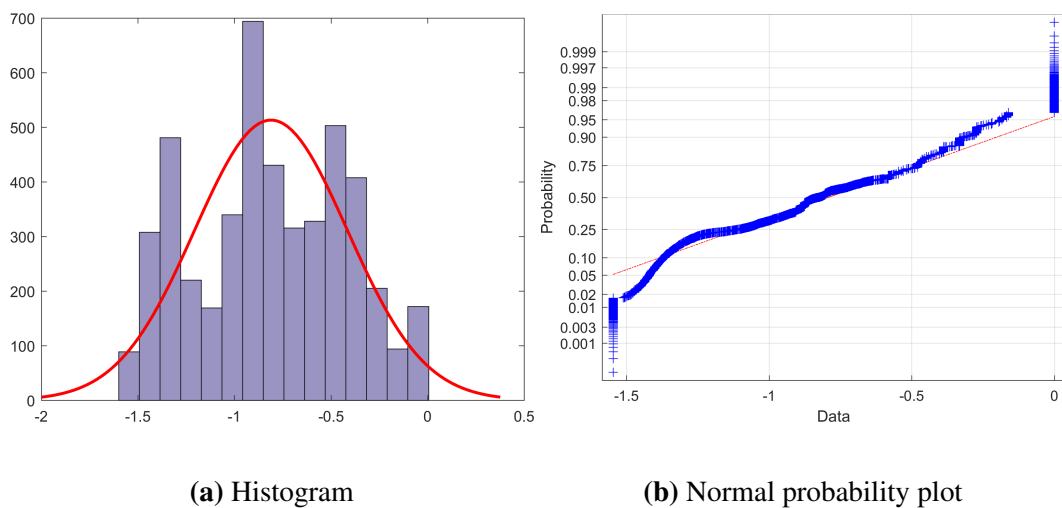


Figure 6.18 Normal probability plot of regular sample



(a) Histogram

(b) Normal probability plot

Figure 6.19 Histogram and normal probability plot of the transformed regular sample. The transformation applied is the Box-Cox with $\lambda = -0.2606$.

Table 6.13 Estimated trend models (regular sample)

Model	Estimated Trend Function
Mean	$m_z(\mathbf{s}) = -0.8105$
Linear	$m_z(\mathbf{s}) = -0.3044 + 5.1258 \cdot 10^{-4}x - 0.0098y$
Quadratic	$m_z(\mathbf{s}) = -0.3316 + 1.6886 \cdot 10^{-4}x - 0.0065y + 5.1848 \cdot 10^{-6}xy$ $+ 6.4395 \cdot 10^{-8}x^2 - 3.5533 \cdot 10^{-5}y^2$
Cubic	$m_z(\mathbf{s}) = -0.2066 - 0.0022x - 0.0066y + 4.3039 \cdot 10^{-5}xy + 8.3445 \cdot 10^{-6}x^2$ $- 1.0253 \cdot 10^{-4}y^2 - 4.8771 \cdot 10^{-8}x^2y - 1.5470 \cdot 10^{-7}xy^2$ $- 9.1203 \cdot 10^{-9}x^3 + 5.2494 \cdot 10^{-7}y^3$
Quartic	$m_z(\mathbf{s}) = -0.2760 - 0.0043x + 0.0074y + 1.4800 \cdot 10^{-5}xy + 3.5298 \cdot 10^{-5}x^2$ $- 4.8234 \cdot 10^{-4}y^2 - 1.0227 \cdot 10^{-7}x^2y + 7.4577 \cdot 10^{-7}xy^2$ $- 1.1655 \cdot 10^{-7}x^3 + 3.8748 \cdot 10^{-6}y^3 + 8.4378 \cdot 10^{-10}x^2y^2$ $- 8.6842 \cdot 10^{-11}x^3y - 6.6459 \cdot 10^{-9}xy^3 + 1.4608 \cdot 10^{-10}x^4$ $- 8.4033 \cdot 10^{-9}y^4$
Linear+Periodic	$m_z(\mathbf{s}) = -0.3661 + 5.1253 \cdot 10^{-4}x - 0.0088y + 0.0033 \cos 2\pi 0.0256x$ $- 0.0029 \sin 2\pi 0.0256x - 0.0524 \cos 2\pi 0.0082y + 0.0520 \sin 2\pi 0.0082y$ $- 4.2847 \cdot 10^{-4} \cos 2\pi 0.0513x + 0.0032 \sin 2\pi 0.0513x$ $- 0.0025 \cos 2\pi 0.0164y + 0.0211 \sin 2\pi 0.0164y$

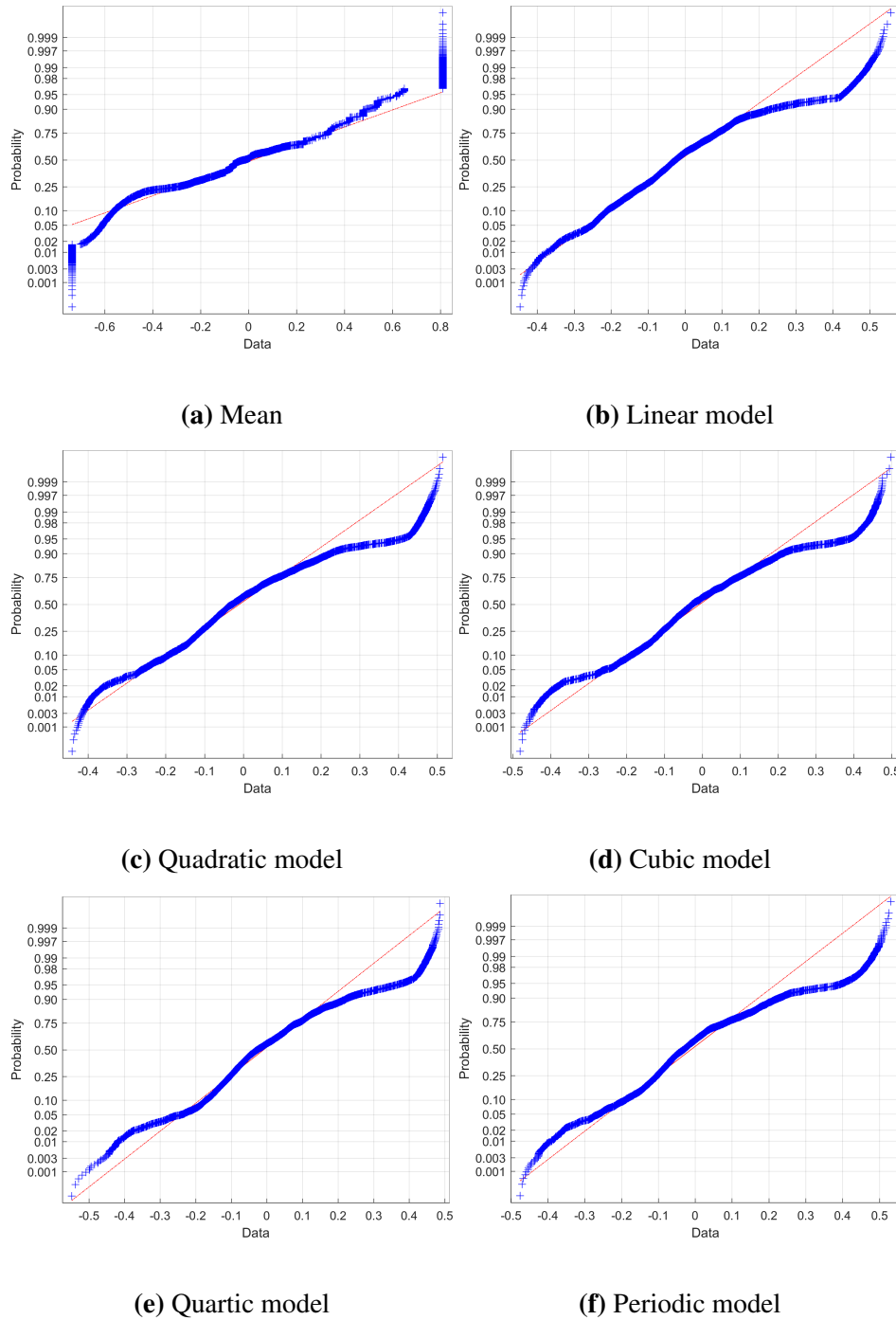


Figure 6.20 Normal probability plots of the fluctuations resulting from the estimated trend models.

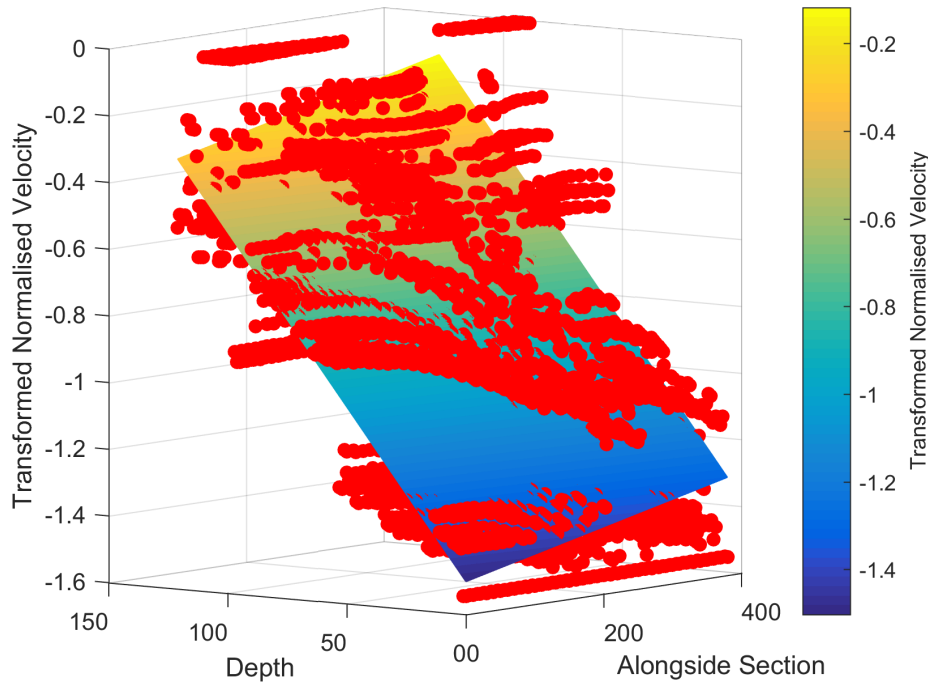


Figure 6.21 Multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points. The trend equation is given by $m_z(\mathbf{s}) = -0.3044 + 5.1258 \cdot 10^{-4}x + 0.0098y$.

Table 6.14 Transformed and detrended dataset statistics

Min	Max	Mean	Median	Variance	Skewness	Kurtosis
-0.446	0.557	0.000	-0.025	0.033	0.744	3.745

In Fig. 6.21 the multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points is depicted, while in Fig. 6.22 is plotted the histogram of the transformed and detrended data (i.e. the fluctuations), and in Table 6.14 are evaluated the statistics of them. Finally, in Fig. 6.23 is presented the transformed and detrended Marmousi model, in order to obtain an inspection of the random field, which we intend to regenerate hereinafter with the five variations of the analysis procedure, described in Section 6.2.

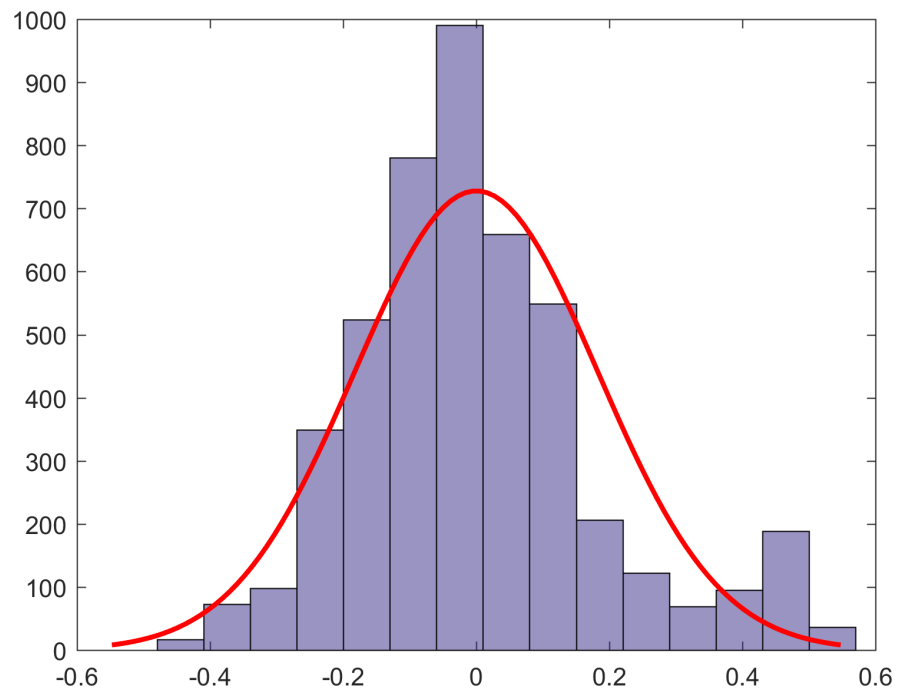


Figure 6.22 Histogram of transformed and detrended sample dataset

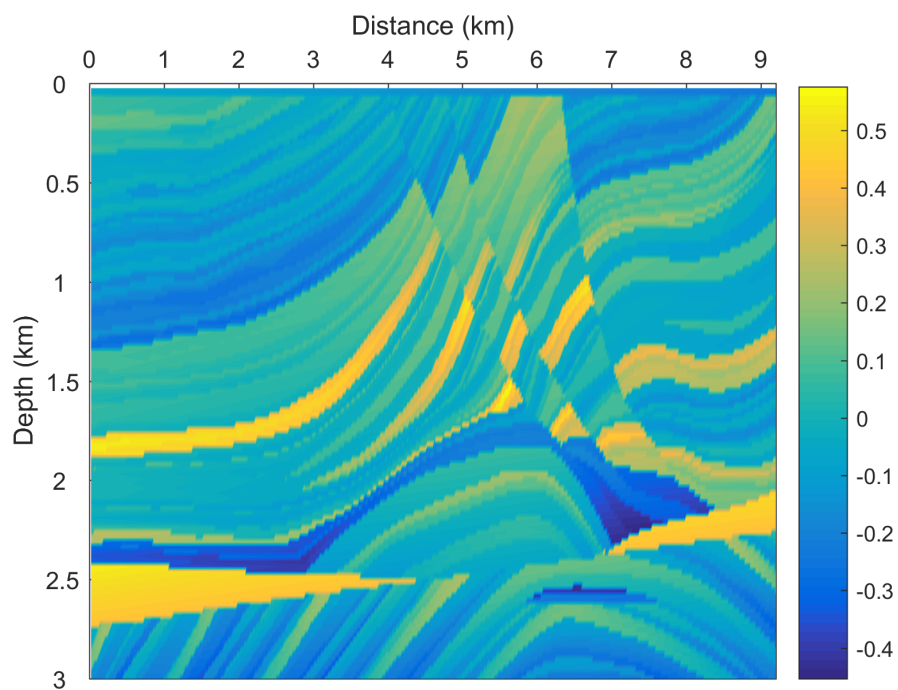


Figure 6.23 Transformed and detrended Marmousi model

Calculation of Experimental Variogram

The experimental directional variograms with angular step 4° and 15° are shown in Fig. 6.24. These figures display clearly the significant anisotropic characteristics of the field (as it is highly expected for such geological data). More specifically, it can be easily seen that the spatial correlation differs significantly from direction to direction; the variogram is stabilized around a maximum value (sill) in much more longer distances along the horizontal and almost horizontal directions than the other directions.

The experimental variograms with the smaller angular step (4°) have been calculated only in order to provide a clearer visualization of the anisotropy of the field, while the directional experimental variograms with the higher angular step (15°) are used in practice in the following steps. We avoid to use the experimental variograms with the small angular step due to the higher computational cost that they will lead the further analysis.

DirVar0

In the *DirVar0* method the investigated random field (transformed and detrended Marmousi model) is modeled by anisotropic variogram equations without estimating the anisotropy parameters in advance. Hence, the anisotropic variogram equations contain the maximum number of unknown parameters for each model. The parameter inference of these models is achieved through the minimization of the error function described by Eq. (5.8) between the experimental directional variograms and the chosen anisotropic theoretical models.

The initial values and the boundaries of the parameters for each model used in the optimization step are presented in Table 6.15, and the resulting optimum parameters are presented in Table 6.16. The investigated models' parameters generally agree, with only exception the anisotropy ratios which exhibit relatively high discrepancies. However, the correctness of them cannot be evaluated before the cross validation procedure.

After the parameter inference, the best model is selected by means of Leave-One-Out Cross Validation (LOOCV). In order to obtain the estimations of the original field (normalised velocities) on the dataset locations and to compare them with the original values, the kriged values are back-transformed in two steps:

- a)) the corresponding trend values at the known points are added to the estimations of the stochastic component (detrended and transformed normalised velocities), and
- b)) the Box-Cox transformation is inverted.

The performance of the models is evaluated by calculating the validation measures described in Section 5.3.2, and the best model is chosen based on the coefficient expressed by Eq.

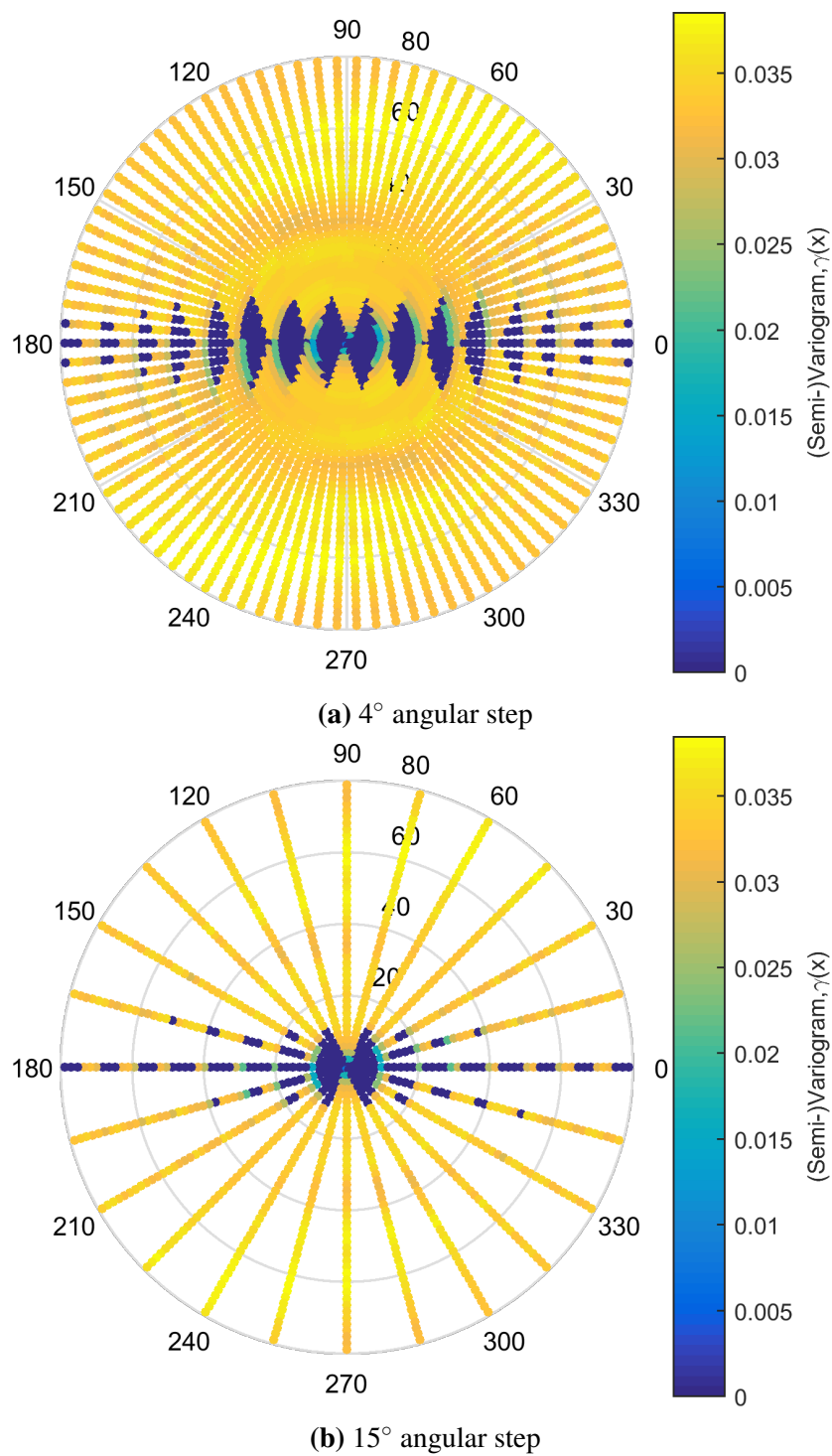


Figure 6.24 Directional experimental variograms of the field. For both cases the angular tolerance is 20°, the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 80, and is divided into 45 distance lags.

Table 6.15 Initial values and boundaries of the parameters for optimization step, where $\hat{g}_{max}(\equiv \text{maximum value of experimental variogram}) = 0.0384$, $d_{max} \simeq 80$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 = 1000, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$.

Model	Initial Values	Boundaries
Gen.		
Exponential	$[b, \nu = 1.5]$	$[b_b, \nu \in (0, 2)]$
Gaussian	$[b]$	$[b_b]$
Spherical	$[b]$	$[b_b]$
Gen. Matérn	$[b]$	$[b_b]$
Spartan	$[b_{sp}, \eta_1 = 1]$	$[b_{sp,b}, \eta_1 \in (-2, \infty)]$

Table 6.16 Optimum Parameters of the investigated variogram models

Model	σ_z^2	ξ_1	R	ϕ	c_0	ν
Gen.						
Exponential	0.035	8.745	0.340	-83.8°	0.000	0.756
Gaussian	0.026	5.879	0.253	-85.3°	0.008	—
Spherical	0.026	7.657	0.109	-84.4°	0.008	—
Gen. Matérn	0.034	8.847	0.210	-83.8°	0.000	0.300
	η_0	ξ_1	R	ϕ	c_0	η_1
Spartan	4.257	8.306	0.085	-84.0°	0.000	88.809

Table 6.17 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.051	0.395	0.006	0.079	0.894	0.927	0.617
Gaussian	0.051	0.393	0.006	0.078	0.896	0.928	0.640
Spherical	0.048	0.372	0.005	0.073	0.910	0.934	0.850
Gen. Matérn	0.050	0.384	0.006	0.076	0.902	0.931	0.723
Spartan	0.049	0.375	0.005	0.073	0.909	0.934	0.829

(5.25). As can be seen from the validation measures of the LOOCV presented in Table 6.17 the best model is the Spherical, followed by the Spartan and the Generalized Matérn models.

A visualisation of the the selected theoretical model's fitting to the experimental variogram is interpreted in Fig. 6.25, in which the directional experimental variograms of the field on the directions $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° are plotted against the theoretical model (while the rest directional variograms in directions $15^\circ, 45^\circ, 75^\circ, 105^\circ, 135^\circ$, and 165° can be found in Appendix A). The best anisotropic model fits fairly well to the directional experimental variograms.

In the next step, Ordinary Kriging is applied to the sample stochastic component field, using the three best models and an arbitrarily defined rectangular neighborhood of size 22×4 . This neighborhood is selected so as to ensure that neighbors from at least two drill-holes away on both sides will be included. Only the results of the first model is presented in detail, while the other two are used only for comparison.

The estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) given by the best model is shown in Fig. 6.26. A scatter diagram of original and estimated values of the stochastic component, as well as their histograms can be seen in Fig. 6.27. Finally, in Table 6.18 the measures of the stochastic component estimation performance for the three best models are presented.

By adding the trend to the estimations of the stochastic component of the field and inverting the Box-Cox transformation, the estimation of the total field, which is shown in Fig. 6.28, is derived. As for the stochastic component, a scatter diagram of the original and the estimated values of the total field, as well as their histograms can be seen in Fig. 6.29. In general, the estimations follow the original values without achieving satisfying proximity of the total distribution. Both the tails of the distribution as the middle of it exhibit significant discrepancies.

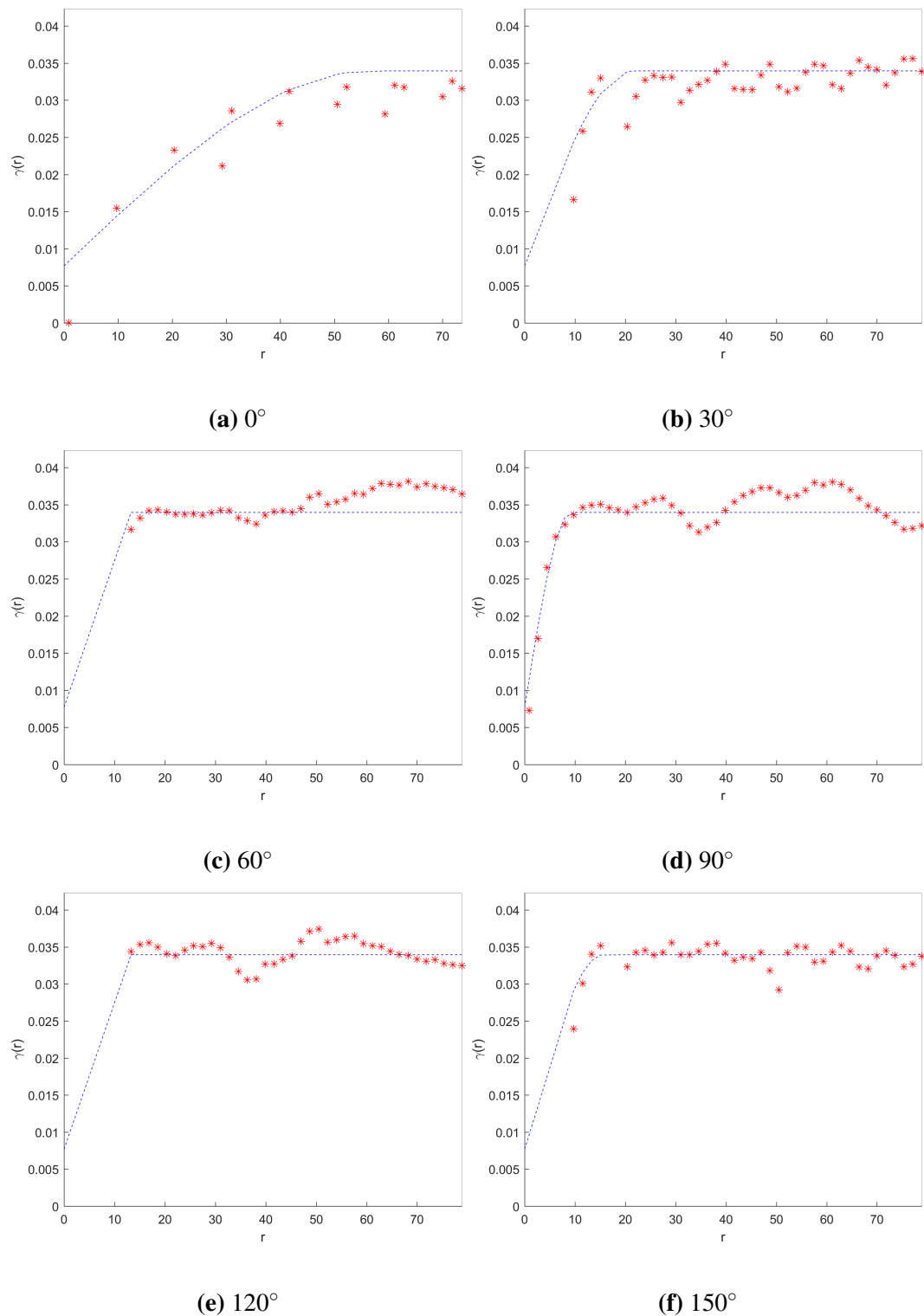


Figure 6.25 Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.026, \xi_1 = 7.657, R = 0.109, \phi = -84.4^\circ, c_0 = 0.008$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

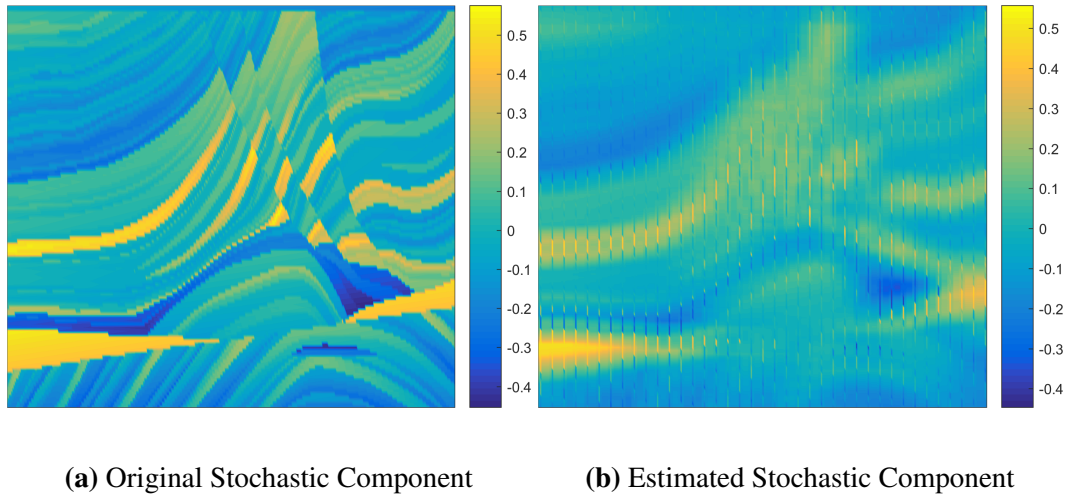


Figure 6.26 Original and Estimation of the stochastic component of the field. The model used is the Spherical with parameters $\sigma_z^2 = 0.026$, $\xi_1 = 7.657$, $R = 0.109$, $\phi = -84.4^\circ$, $c_0 = 0.008$.

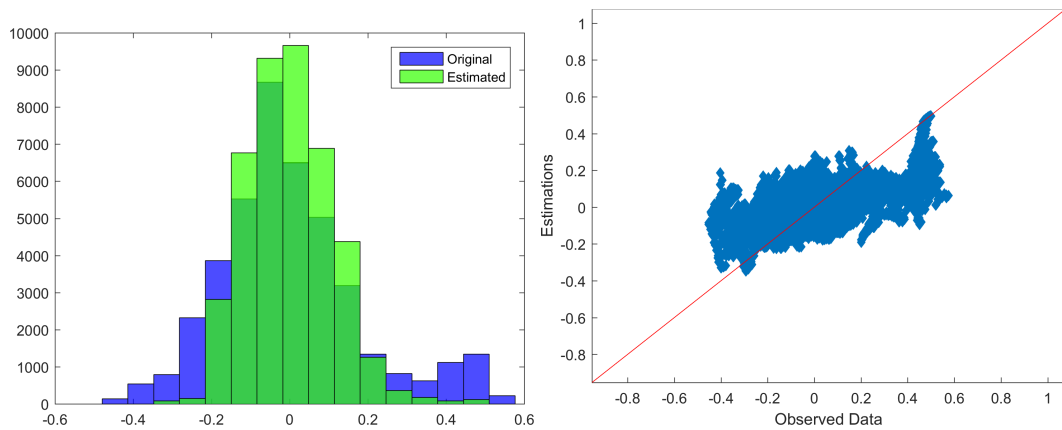


Figure 6.27 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is the Spherical with parameters $\sigma_z^2 = 0.026$, $\xi_1 = 7.657$, $R = 0.109$, $\phi = -84.4^\circ$, $c_0 = 0.008$.

Table 6.18 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spherical	0.094	0.579	0.016	0.127	0.732	0.723	0.530
Spartan	0.094	0.580	0.016	0.127	0.728	0.720	0.512
Gen. Matérn	0.097	0.589	0.017	0.132	0.700	0.697	0.413

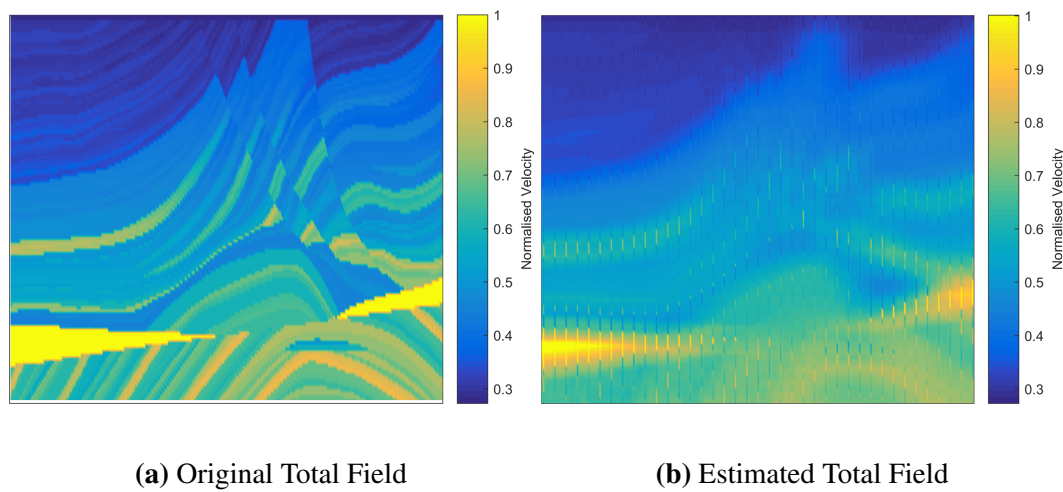


Figure 6.28 Original and estimated total field

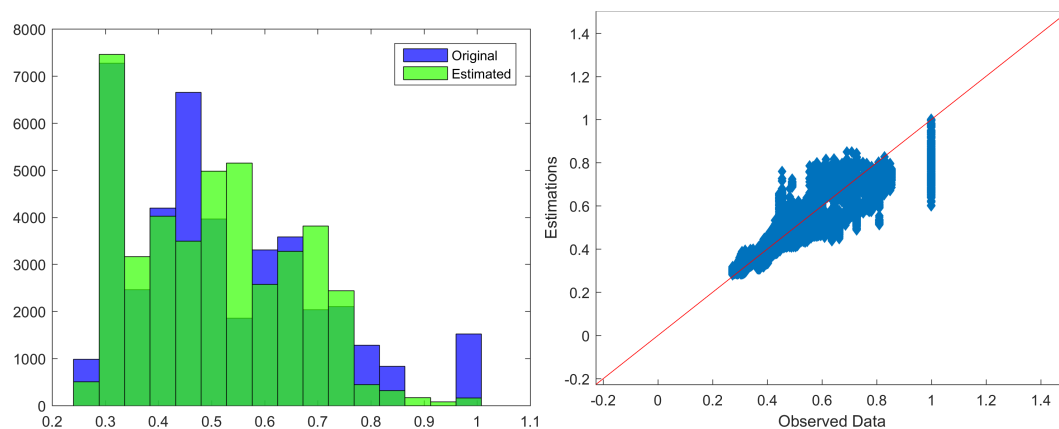


Figure 6.29 Histograms and Scatter plot of original and estimated values (total field)

In Table 6.19 the measures of the total field (i.e. normalised velocities) estimation performance for the three best models are presented. From these measures it can be observed that the two first models' performance is almost identical, while the third one follows closely. In addition, Fig. 6.30 maps the confidence interval width of the estimations, i.e. it depicts the spatial distribution of the uncertainty. The uncertainty is zero at the known locations, while at the investigated points it is equal to 0.37 with slightly smaller values at the edges. This can be explained by the fact that the number of the neighbors taken into account by the ordinary kriging estimations is the same for all the missing points, except for the edges of the field where it decreases, due to the regularity of the sample.

In Table 6.20 are presented the classification measures (see section 5.3.2) for the cases of 4 and 16 classes. By increasing the number of classes, the correlation coefficients increase,

Table 6.19 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spherical	1.317	1.867	1.803	1.343	0.935	0.943	0.881
Spartan	1.317	1.866	1.803	1.343	0.934	0.942	0.880
Gen. Matérn	1.317	1.865	1.803	1.343	0.930	0.938	0.872

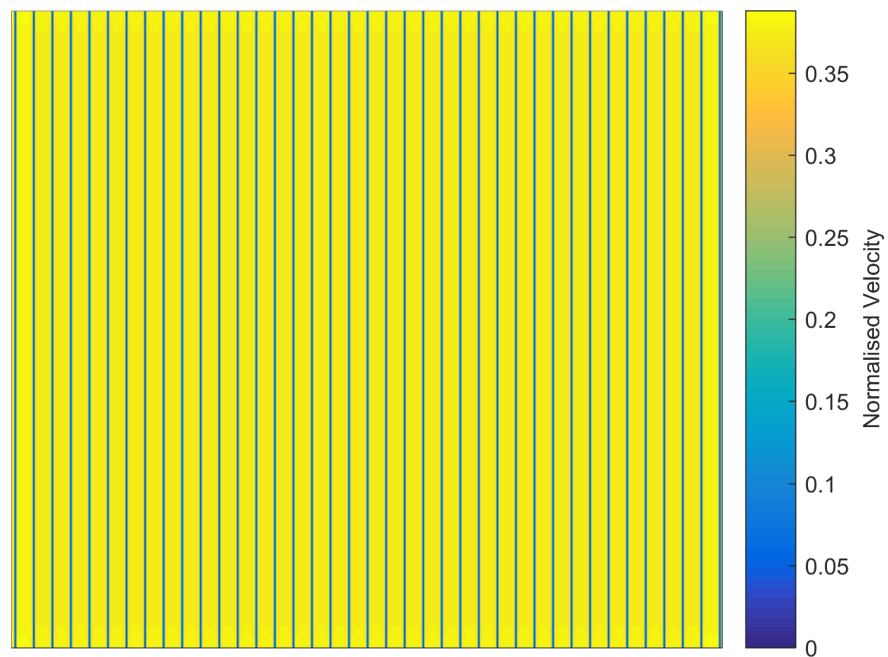
**Figure 6.30** Uncertainty of the total field's estimation

Table 6.20 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Spherical	0.860	0.873	0.220	0.916	0.936	0.584
Spartan	0.859	0.873	0.222	0.914	0.936	0.584
Gen. Matérn	0.852	0.867	0.227	0.906	0.931	0.588

while the MCR decreases. The increasing of the N_c leads to a more complex model, which closer to the reality. That is the reason of the correlation coefficients' improvement. On the contrary, the misclassification rate's deterioration can be attributed to the thinning of the binning, which incommodes the interpolation.

DirVar1

In the *DirVar1* method the investigated random field is modeled by anisotropic variogram equations after estimating the anisotropy parameters of them. The anisotropy parameters are estimated by means of DVF (see section 5.2.2). Subsequently, the estimated parameters are replaced to the corresponding anisotropic variogram functions, reducing by three the number of unknown parameters for each model. The parameter inference of the models with the lower degrees of freedom is achieved through the minimization of the error function between the experimental directional variograms and the chosen anisotropic theoretical models.

The initial values and the boundaries of the parameters for each isotropic model used in the anisotropy parameter inference step with DVF method are as specified in Table 6.15 without considering the anisotropy parameters R and ϕ . The anisotropy parameters for each model are estimated by fitting an ellipse to the resulting pairs of (ϕ, ξ) of each model. The fitting of the ellipses to these pairs for the 5 models are depicted in Fig. 6.31, while the estimated anisotropy parameters for all models are given in Table 6.21. In general, the investigated models' fitting show that the major axis of the anisotropy ellipsis is either on the almost horizontal direction (0°) or on the direction of about $30 - 40^\circ$. This results agree with the intuitive estimations of the anisotropy angle mentioned at section 6.1.2. The non-stationarity of the field is also indicated by the strong dependence of the correlation length values' on the direction as it is illustrated in Fig. 6.31. For all the models the correlation lengths on the vertical or almost vertical directions are significantly smaller than these on the other directions. These rapid changes may also affect negatively the optimization procedure.

Table 6.21 Anisotropy parameters of the investigated variogram models estimated with DVF method

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	3.981	0.264	-90.0°
Gaussian	4.827	0.298	-90.0°
Spherical	11.550	0.397	-90.0°
Gen. Matérn	2.987	0.145	-90.0°
Spartan	40.338	0.307	-50.9°

Table 6.22 Optimum Parameters of the variogram models with the lower degrees of freedom

Model	σ_z^2	c_0	ν
Gen. Exponential	0.033	0.000	1.012
Gaussian	0.026	0.008	—
Spherical	0.026	0.007	—
Gen. Matérn	0.031	0.002	0.700
	η_0	c_0	η_1
Spartan	0.832	0.000	1.928

As regards the variations on the estimated anisotropy ratios, no reliable conclusions can be drawn before the cross validation procedure.

By replacing the estimated anisotropy parameters to the anisotropic variogram models and minimizing the error function of the new models and the experimental directional variograms the rest parameters are estimated, as presented in Table 6.22. The common parameters generally agree for all the investigated models.

After the parameter inference Leave-One-Out Cross Validation (LOOCV) is applied in order to define the best models. The validation measures of the LOOCV, presented in Table 6.23, give as best model the Spartan, followed by the Generalized Matérn and the Generalized Exponential models.

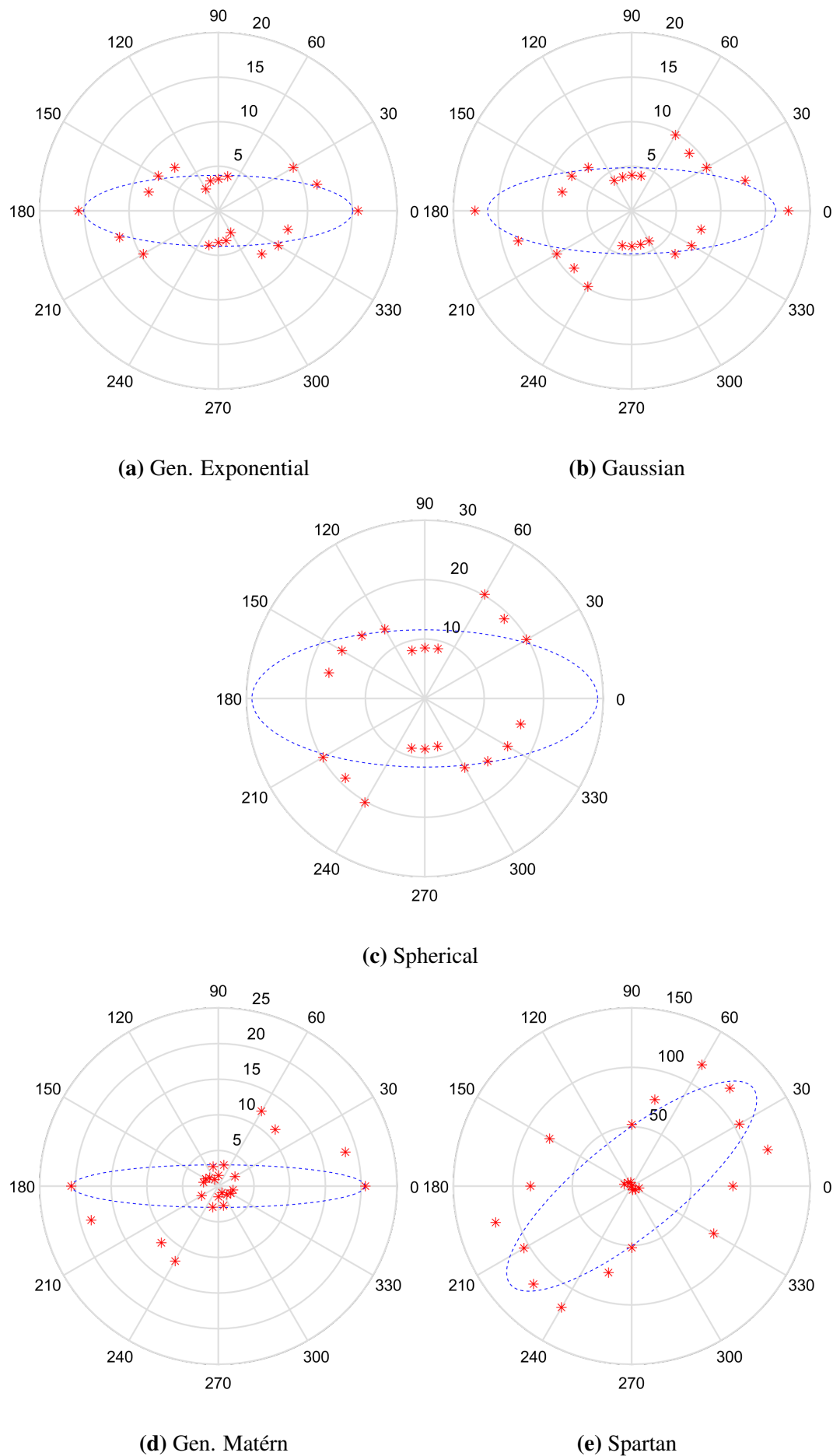


Figure 6.31 Fitting of ellipses to the pairs (ϕ, ξ) of the investigated variogram models

Table 6.23 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.053	0.401	0.007	0.081	0.887	0.923	0.562
Gaussian	0.055	0.420	0.007	0.086	0.872	0.914	0.437
Spherical	0.054	0.411	0.007	0.083	0.880	0.919	0.502
Gen. Matérn	0.053	0.401	0.007	0.081	0.887	0.923	0.562
Spartan	0.049	0.372	0.005	0.074	0.908	0.933	0.848

Table 6.24 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	0.094	0.676	0.017	0.129	0.716	0.709	0.507
Gen. Matérn	0.104	0.605	0.020	0.142	0.631	0.641	0.273
Gen. Exponential	0.104	0.605	0.020	0.142	0.632	0.642	0.276

The fitting of the best model to the experimental variogram along selected directions is interpreted in Fig. 6.32 (the rest directional variograms can be found in Appendix A). As it can be seen the best theoretical variogram model diverges significantly from the experimental directional variograms. This can be attributed to minimization miscalculations, i.e. inappropriate objective function (very smooth or possible local minima), inappropriate initial values or to the fields complexity (as mentioned above). Thus, the analysis is continued without taking any further action.

Implementing OK with the determined best model, the resulting estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) is as shown in Fig. 6.33. The scatter diagram and histograms of the original and the estimated values of the stochastic component are illustrated in Fig. 6.34, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.24.

The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.35, and the corresponding scatter diagram and histograms are shown in Fig. 6.36. In general, the estimations follow the original values

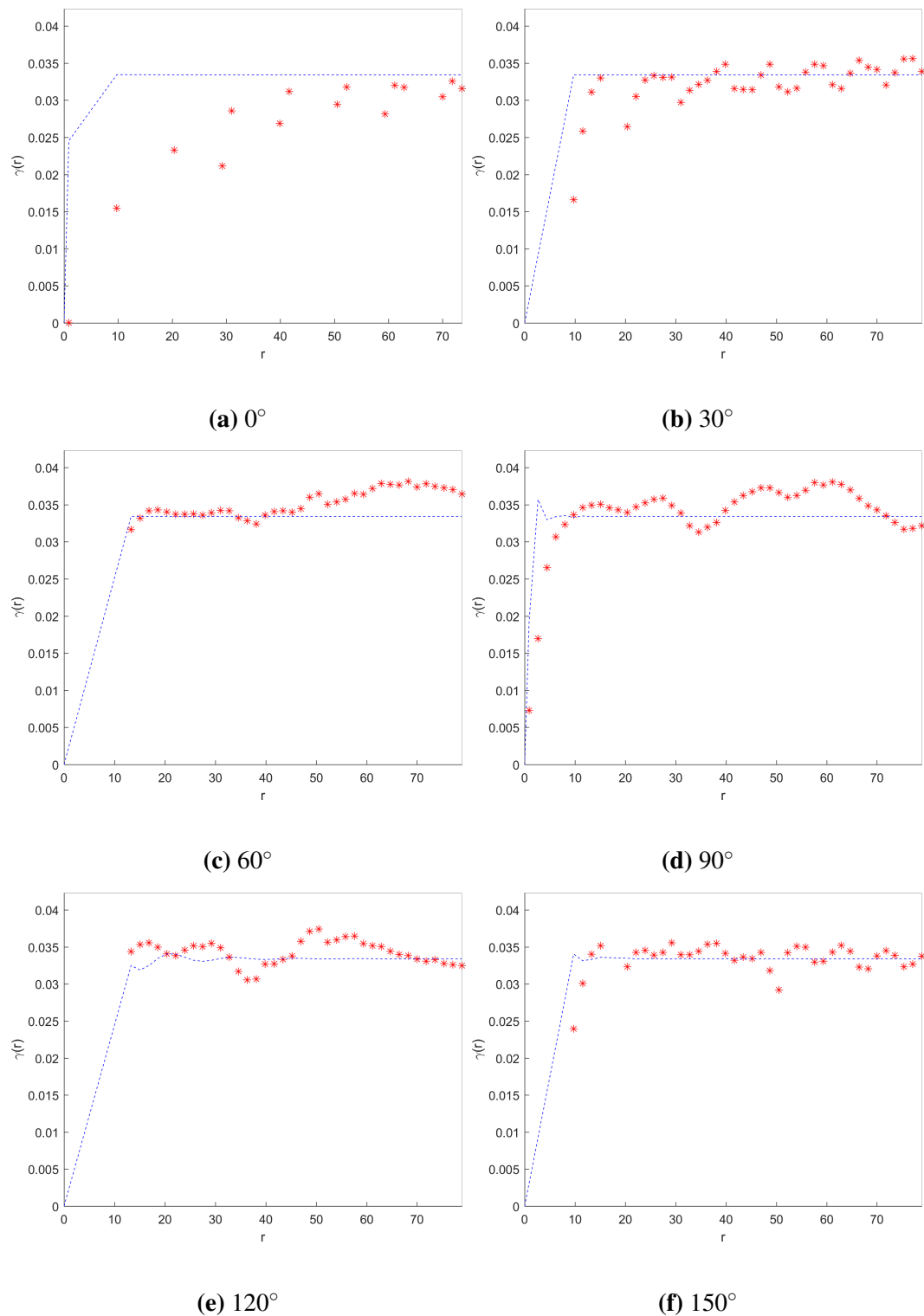
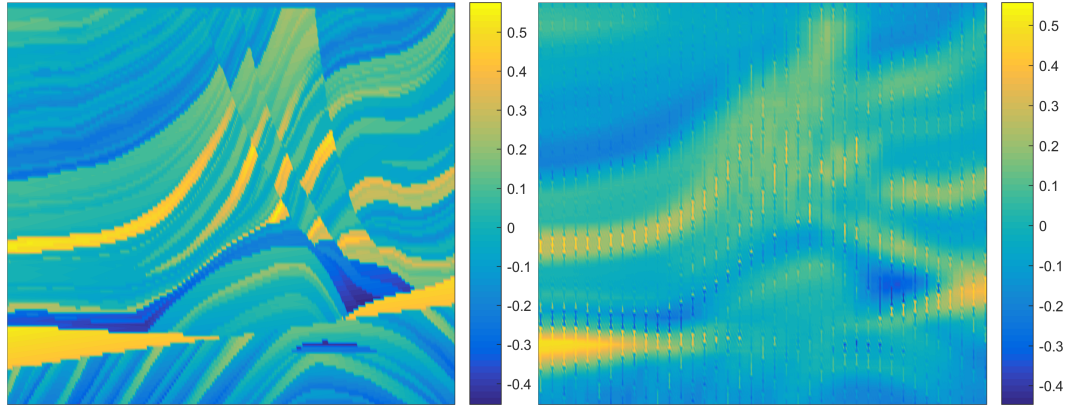


Figure 6.32 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.832$, $\xi_1 = 40.338$, $R = 0.307$, $\phi = -50.9^\circ$, $c_0 = 0.000$, $\eta_1 = 1.928$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.



(a) Original Stochastic Component

(b) Estimated Stochastic Component

Figure 6.33 Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 0.832$, $\xi_1 = 40.338$, $R = 0.307$, $\phi = -50.9^\circ$, $c_0 = 0.000$, $\eta_1 = 1.928$.

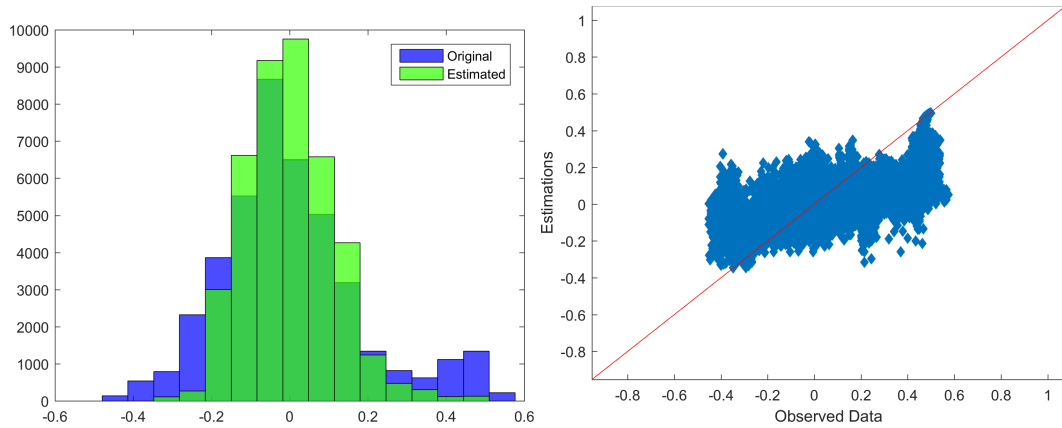


Figure 6.34 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 0.832$, $\xi_1 = 40.338$, $R = 0.307$, $\phi = -50.9^\circ$, $c_0 = 0.000$, $\eta_1 = 1.928$.

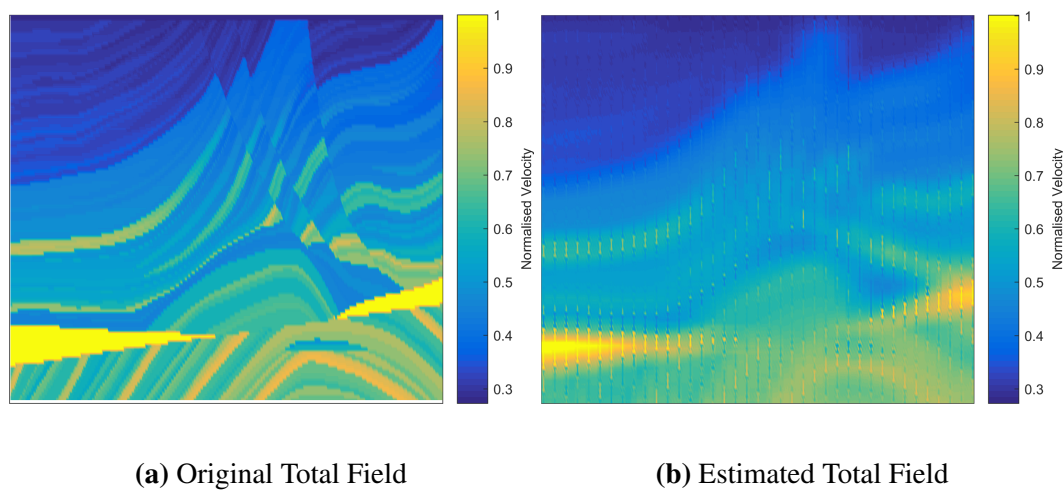


Figure 6.35 Original and estimated total field

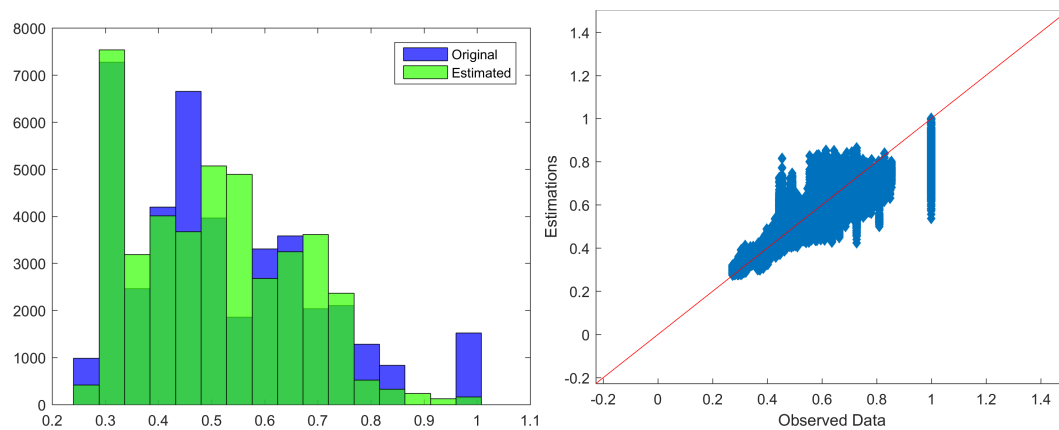


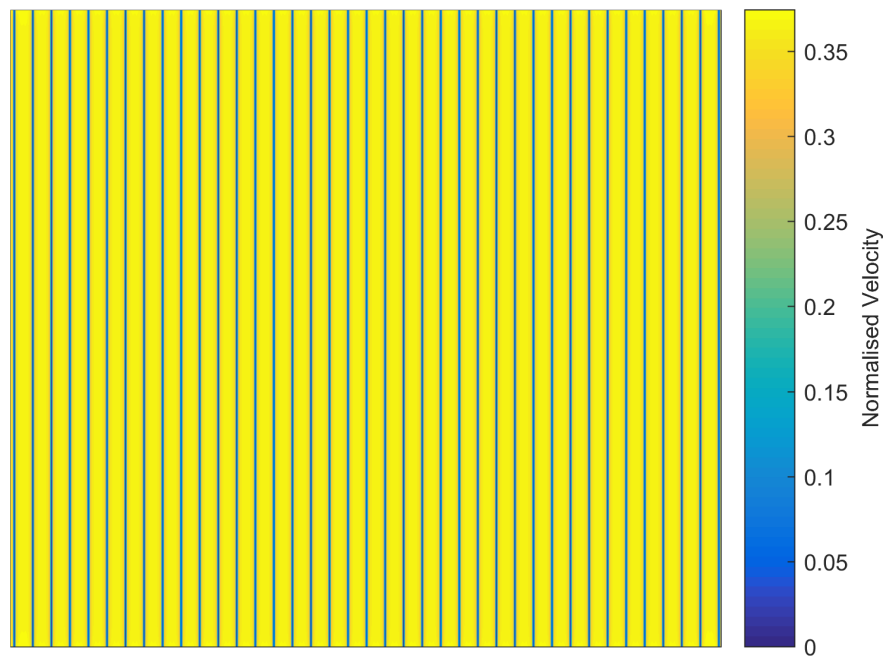
Figure 6.36 Histograms and Scatter plot of original and estimated values (total field)

without, however, achieving satisfying proximity of the total distribution; both the tails of the distribution as the middle of it exhibit significant discrepancies. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.25, show that the first model has significantly better performance than the other two which have almost the same validation measures. Finally, Fig. 6.37 shows the spatial distribution of the ordinary kriging estimations uncertainty. The uncertainty is zero at the drill-holes (known points), while increases from $\simeq 0.30$ near the drill-holes to 0.37 at great distance between the drill-holes.

Also, in Table 6.26 are presented the classification measures for the cases of 4 and 16 classes. The same observations, as in *DirVar0*, apply.

Table 6.25 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	1.317	1.869	1.803	1.343	0.932	0.940	0.876
Gen. Matérn	1.317	1.863	1.804	1.343	0.918	0.927	0.851
Gen. Exponential	1.317	1.863	1.804	1.343	0.918	0.928	0.851

**Figure 6.37** Uncertainty of the total field's estimation**Table 6.26** Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Spartan	0.856	0.871	0.219	0.912	0.934	0.578
Gen. Matérn	0.840	0.859	0.235	0.889	0.921	0.600
Gen. Exponential	0.840	0.859	0.235	0.889	0.921	0.600

DirVar2

In the *DirVar2* method the anisotropy parameters are also estimated by means of DVF (see section 5.2.2) in advance, as in the *DirVar1* variation, but in this case these parameters are used for rescaling and rotating the original anisotropic coordinations system to a new isotropic coordinations system. Subsequently, the field is modeled by isotropic variogram equations following a similar to the previous variations procedure (i.e. parameter inference by minimizing the error function, LOOCV and OK).

Therefore, the 5 sets of anisotropy parameters estimated in the *DirVar1* method (see Table 6.21) are used for the inversion of the anisotropic effect, giving equal number of new coordination systems. Each system corresponds to a variogram model with the isotropic form of which it is tested in the following steps. Before this, however, it is necessary to examine whether the new systems are actually isotropic or have retained anisotropic characteristics. This is dependent on the reliability of the anisotropy parameters estimation. For this purpose, the experimental directional variograms for each coordinations system are calculated, and subsequently the new anisotropy parameters are estimated by means of DVF. The results are shown in Table 6.27, while in Fig. 6.38 are displayed the experimental directional variograms of the new coordination systems. The results show that despite small improvement (indicated by the proximity of the new correlation length to 1 and the new anisotropy angle to 0) the anisotropic effect remains at relatively high levels as indicated by the very small increase of the anisotropy ratios. Nevertheless, analysis procedure continues by assuming the new coordinations systems as isotropic.

By minimizing the error function of the isotropic variogram functions of the models and the corresponding experimental omnidirectional (isotropic) variograms of the new coordinations systems (see Fig. 6.39) the parameters of the isotropic models are estimated, as presented in Table 6.28. The common parameters are generally close for all the investigated models.

The Leave-One-Out Cross Validation (LOOCV) for the estimated models gives the validation measures presented in Table 6.29. As best model derives the Generalized Exponential, followed by Generalized Mátern and Spartan. As it can be seen from Fig. 6.39, though, all the models except Spartan (probably due to optimization miscalculations) fit fairly well to their corresponding experimental omnidirectional variograms.

The estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) resulting from OK is as shown in Fig. 6.40. The scatter diagram and histograms of the original and the estimated values of the stochastic component are, also, shown in Fig. 6.41, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.30. The resulting estimation of the total

Table 6.27 Anisotropy parameters of the new coordinations systems

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	0.833	0.432	5.2°
Gaussian	0.822	0.468	0.0°
Spherical	0.832	0.324	6.1°
Gen. Matérn	1.598	0.235	11.3°
Spartan	0.826	0.445	0.0°

Table 6.28 Optimum Parameters of the isotropic variogram models

Model	σ_z^2	ξ	c_0	ν
Gen. Exponential	0.033	1.031	0.000	0.948
Gaussian	0.025	1.002	0.007	—
Spherical	0.025	1.083	0.007	—
Gen. Matérn	0.033	1.425	0.000	0.405
	η_0	ξ	c_0	η_1
Spartan	0.045	0.081	0.000	-1.989

Table 6.29 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.025	0.350	0.002	0.048	0.962	0.974	0.937
Gaussian	0.032	0.364	0.003	0.053	0.953	0.967	0.609
Spherical	0.031	0.362	0.003	0.052	0.955	0.969	0.647
Gen. Matérn	0.028	0.370	0.002	0.048	0.961	0.972	0.881
Spartan	0.025	0.361	0.002	0.048	0.961	0.974	0.874

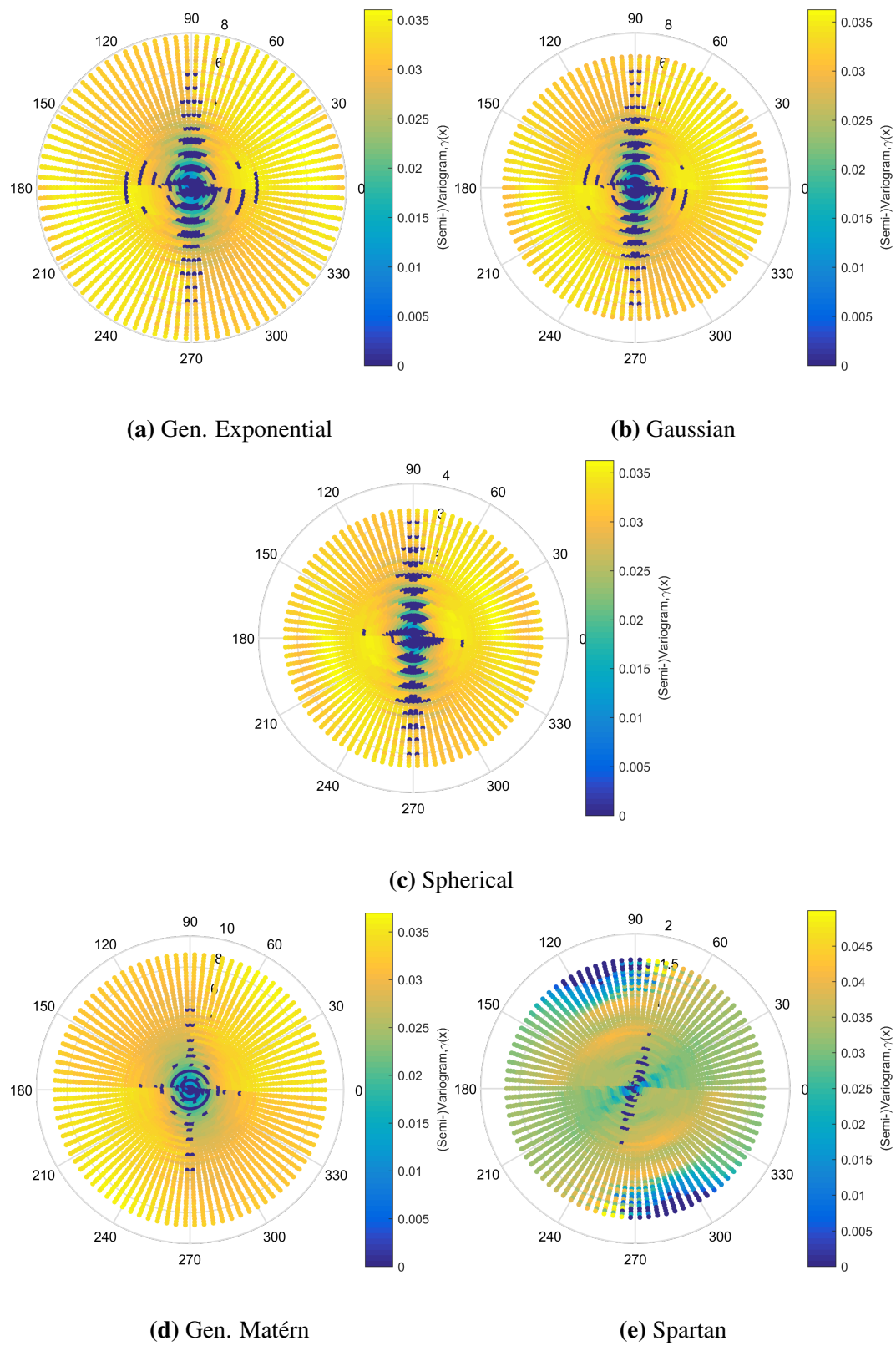
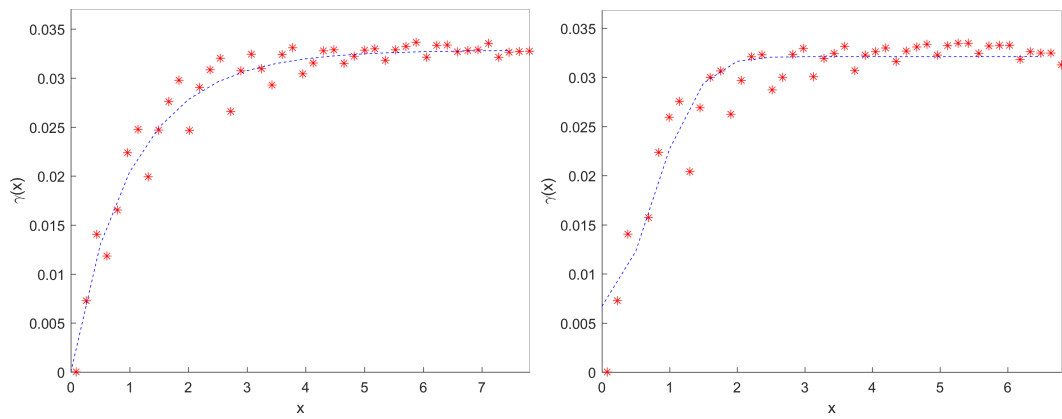
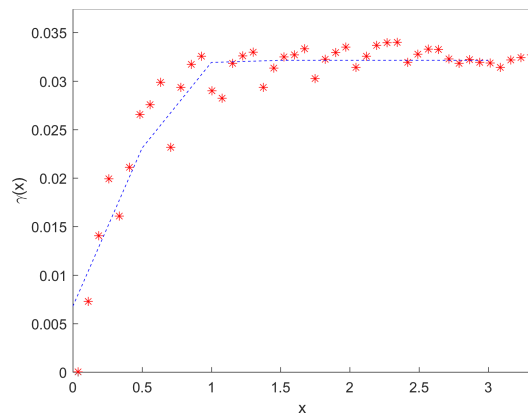


Figure 6.38 Experimental directional variograms of the new coordinations systems

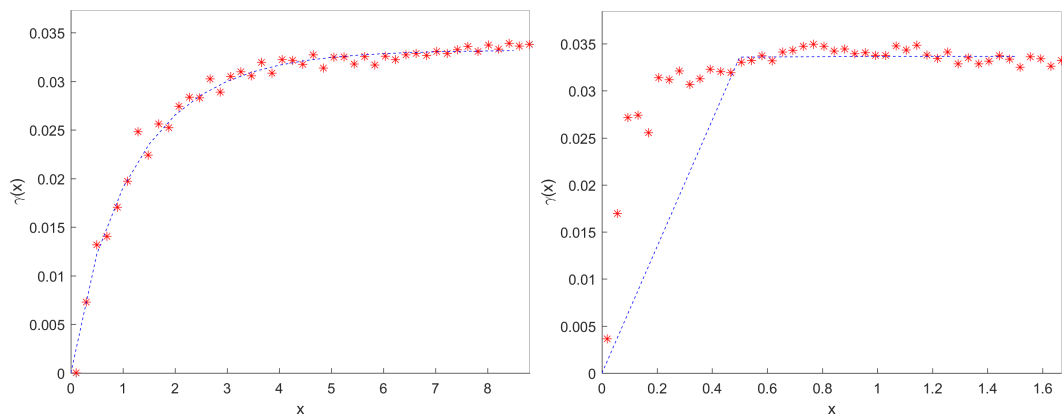


(a) Gen. Exponential

(b) Gaussian



(c) Spherical



(d) Gen. Matérn

(e) Spartan

Figure 6.39 Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.

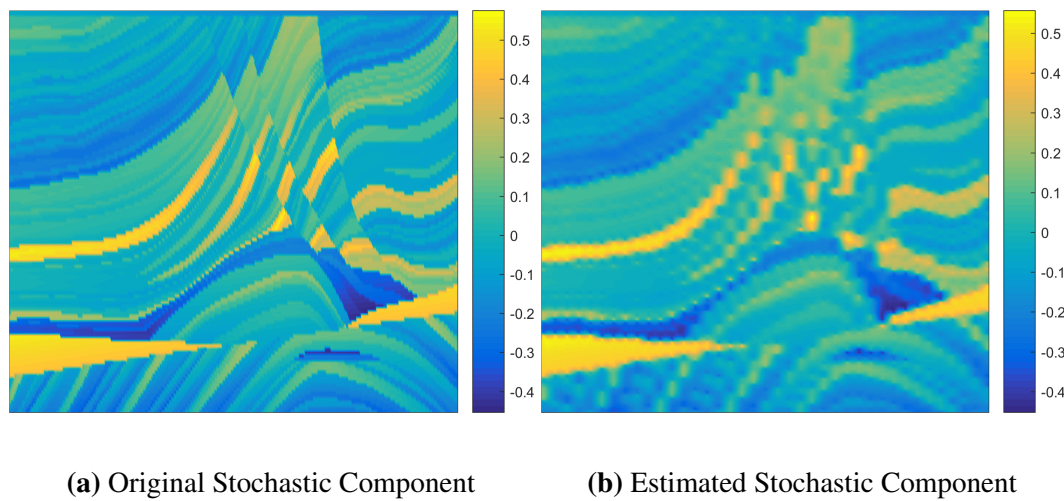


Figure 6.40 Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.033$, $\xi = 1.031$, $c_0 = 0.000$, $\nu = 0.948$.

Table 6.30 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.044	0.681	0.005	0.073	0.916	0.907	0.831
Gen. Matérn	0.046	0.661	0.006	0.074	0.914	0.905	0.783
Spartan	0.066	0.658	0.009	0.094	0.875	0.867	0.286

field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.42, and the corresponding scatter diagram and histograms are shown in Fig. 6.43. In general, the estimations follow the original values achieving a very good proximity of the total distribution. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.31, show that their performance is almost identical. Also, the spatial distribution of the ordinary kriging estimations uncertainty is shown in Fig. 6.44. The uncertainty (as expected from theory) increases gradually from $\simeq 0.00$ near the drill-holes to 0.20 at great distance between the drill-holes, and exhibits the same pattern all over the examined space due to the regularity of the sample.

Finally, in Table 6.32 are presented the classification measures for the cases of 4 and 16 classes. The results show that for all models the measures increase as the number of classes increase, while the best model has the higher performance rates for any number of classes.

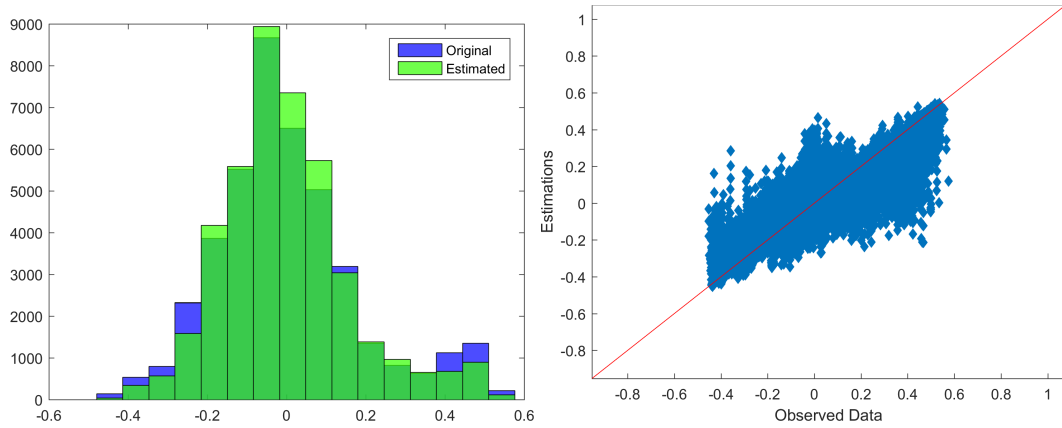


Figure 6.41 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.033$, $\xi = 1.031$, $c_0 = 0.000$, $\nu = 0.948$.

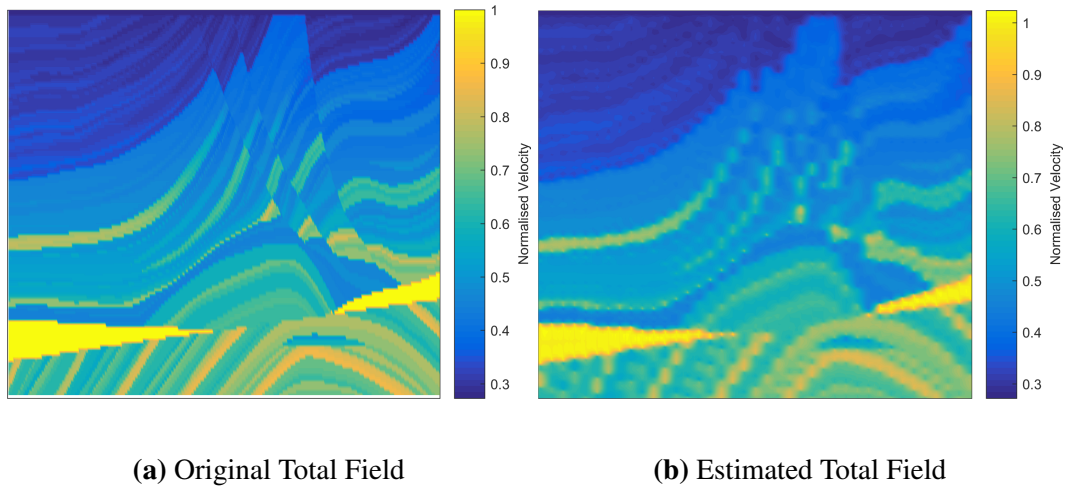


Figure 6.42 Original and estimated total field

Table 6.31 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	1.321	1.843	1.803	1.343	0.960	0.979	0.938
Gen. Matérn	1.321	1.843	1.803	1.343	0.960	0.979	0.938
Spartan	1.318	1.867	1.801	1.342	0.961	0.970	0.932

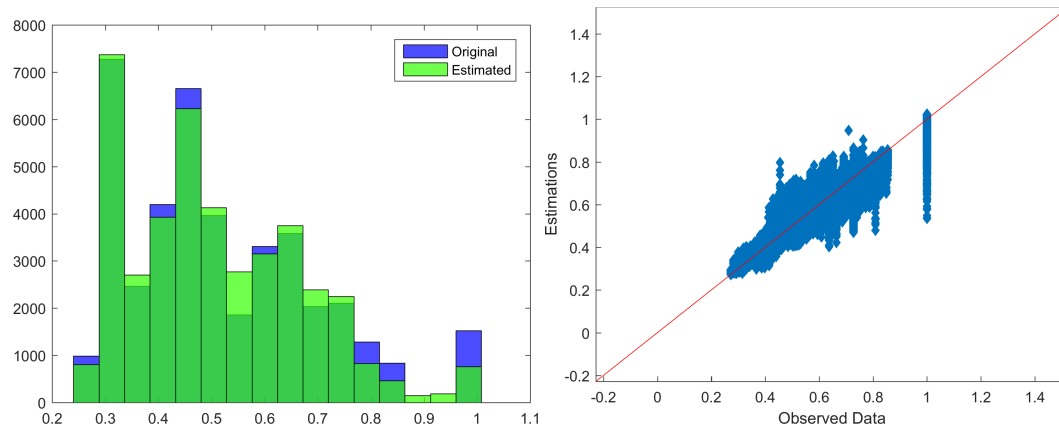


Figure 6.43 Histograms and Scatter plot of original and estimated values (total field)

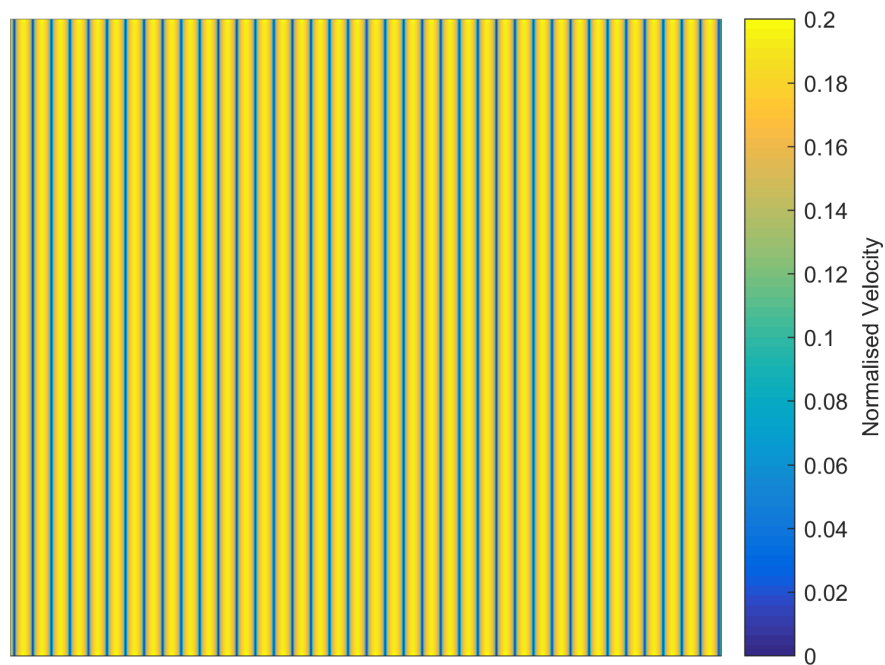


Figure 6.44 Uncertainty of the total field's estimation

Table 6.32 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Gen. Exponential	0.929	0.930	0.116	0.972	0.975	0.317
Gen. Matérn	0.927	0.929	0.119	0.971	0.975	0.330
Spartan	0.891	0.899	0.173	0.953	0.963	0.464

CHI1

In the *CHI1* method the investigated random field is modeled by anisotropic variogram equations after estimating the anisotropy parameters. The anisotropy parameters in this case are estimated by means of CHI (see section 5.2.2) and are the same for all the models. Subsequently, the estimated parameters are replaced to the corresponding anisotropic variogram functions, reducing by two the number of unknown parameters for each model (in the previous variations the parameters were reduced by three) as the CHI method estimates only R and ϕ . The parameter inference of the models with the lower degrees of freedom is achieved through the minimization of the error function between the experimental directional variograms and the chosen anisotropic theoretical models.

The estimated anisotropy parameters are given in Table 6.33. The parameters captured by the CHI estimator indicates that the major axis of the anisotropy ellipsis is almost horizontal and is also significantly greater ($\simeq 10$ times) than the minor axis.

By replacing the estimated anisotropy parameters to the anisotropic variogram models and minimizing the error function of the new models and the experimental directional variograms the rest parameters for each model are estimated, as presented in Table 6.34. The estimated values of the variance and the nugget effect are very close but the correlation lengths differ depending on the model. The correlation length of the Spherical model is significantly higher from the other models.

The Leave-One-Out Cross Validation (LOOCV) gives as best model the Spherical, followed by Spartan and Generalized Exponential.

Table 6.33 Anisotropy parameters of the investigated variogram models estimated with CHI method

R	ϕ
0.108	-88.5°

Table 6.34 Optimum Parameters of the variogram models with the lower degrees of freedom

Model	σ_z^2	ξ	c_0	ν
Gen. Exponential	0.033	4.994	0.000	1.098
Gaussian	0.028	8.471	0.006	—
Spherical	0.033	79.760	0.000	—
Gen. Matérn	0.029	1.074	0.004	3.500
	η_0	ξ	c_0	η_1
Spartan	0.841	5.032	0.000	2.000

Table 6.35 Leave-One-Out Cross Validation Scores (see section [5.3.2](#))

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.052	0.398	0.006	0.080	0.889	0.924	0.678
Gaussian	0.053	0.407	0.007	0.082	0.884	0.921	0.622
Spherical	0.050	0.386	0.006	0.077	0.900	0.930	0.837
Gen. Matérn	0.053	0.404	0.007	0.082	0.885	0.922	0.635
Spartan	0.052	0.398	0.006	0.080	0.890	0.925	0.691

Table 6.36 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spherical	0.099	0.588	0.018	0.133	0.692	0.689	0.476
Spartan	0.103	0.602	0.020	0.140	0.645	0.652	0.344
Gen.							
Exponential	0.103	0.602	0.020	0.141	0.640	0.648	0.333

The fitting of the best model to the experimental variogram on selected directions is interpreted in Fig. 6.45 (the rest directional variograms can be found in Appendix A). As it can be seen the best theoretical variogram model diverges significantly from the experimental directional variograms. This can be attributed to minimization miscalculations, i.e. inappropriate objective function (very smooth or possible local minima), inappropriate initial values or to the fields complexity. Thus, the analysis proceeds without taking any further action.

Implementing OK with the determined best model, the resulting estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) is as shown in Fig. 6.46. The scatter diagram and histograms of the original and the estimated values of the stochastic component are illustrated in Fig. 6.47, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.36.

The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.48, and the correspondng scatter diagram and histograms are shown in Fig. 6.49. In general, the estimations follow the original values without, however, achieving satisfying proximity of the total distribution; both the tails of the distribution as the middle of it exhibit significant discrepancies. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.37, show that the first model has significantly better performance than the other two which have almost the same validation measures. Finally, Fig. 6.50 shows the spatial distribution of the ordinary kriging estimations uncertainty. The uncertainty is zero at the known locations, while at the investigated points it is equal to 0.37 with slightly smaller values at the edges.

Also, in Table 6.38 are presented the classification measures for the cases of 4 and 16 classes. The same comments, as in *DirVar2*, apply.

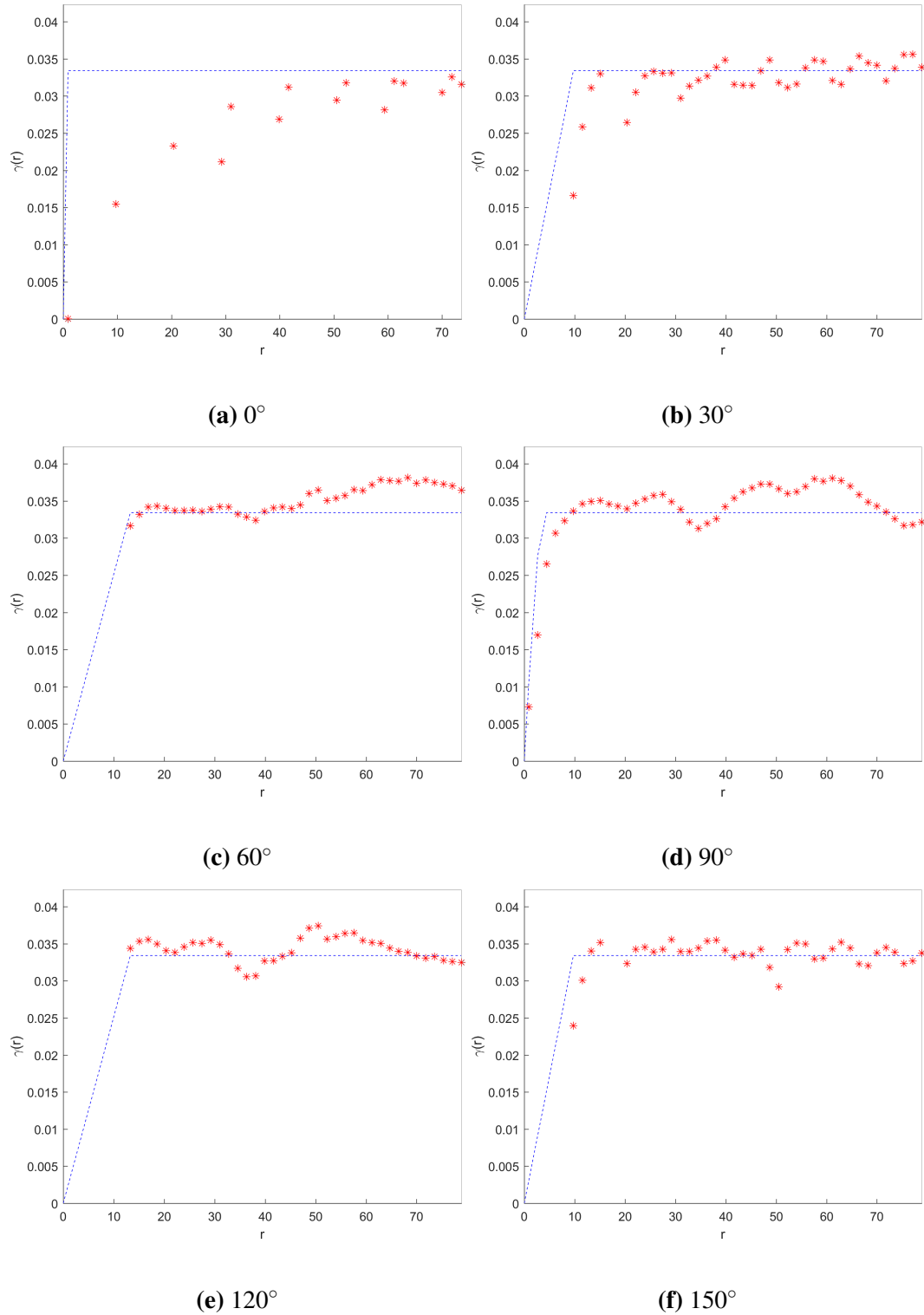


Figure 6.45 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.033$, $\xi_1 = 79.760.338$, $R = 0.108$, $\phi = -88.5^\circ$, $c_0 = 0.000$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

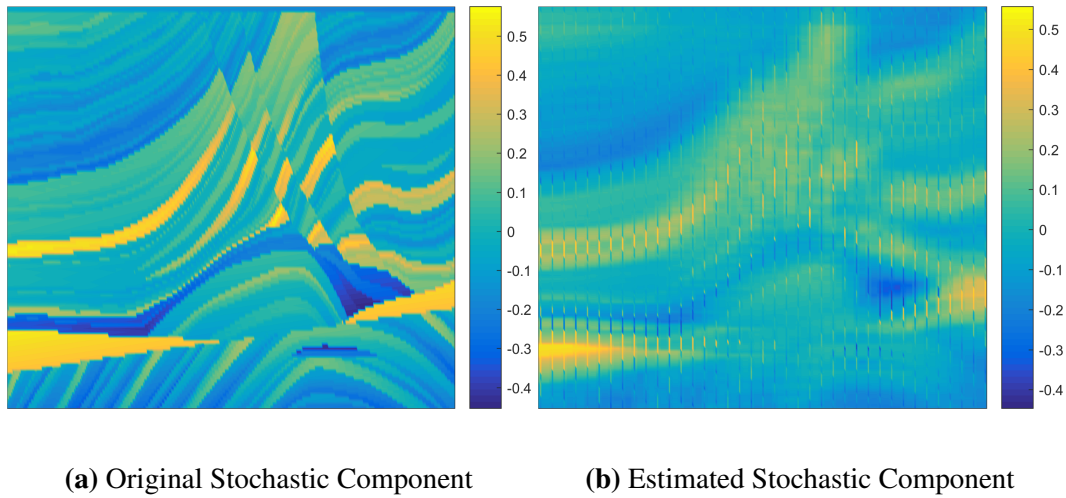


Figure 6.46 Original and Estimation of the stochastic component of the field. The model used is a Spherical with parameters $\sigma_z^2 = 0.033$, $\xi_1 = 79.760.338$, $R = 0.108$, $\phi = -88.5^\circ$, $c_0 = 0.000$.

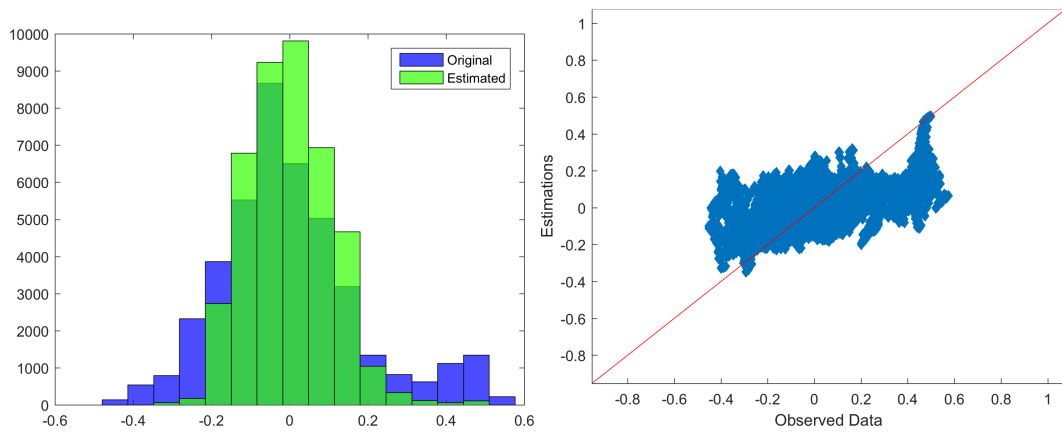


Figure 6.47 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spherical with parameters $\sigma_z^2 = 0.033$, $\xi_1 = 79.760.338$, $R = 0.108$, $\phi = -88.5^\circ$, $c_0 = 0.000$.

Table 6.37 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spherical	1.317	1.865	1.803	1.343	0.929	0.937	0.870
Spartan	1.317	1.864	1.804	1.343	0.920	0.930	0.855
Gen. Exponential	1.317	1.864	1.804	1.343	0.920	0.929	0.854

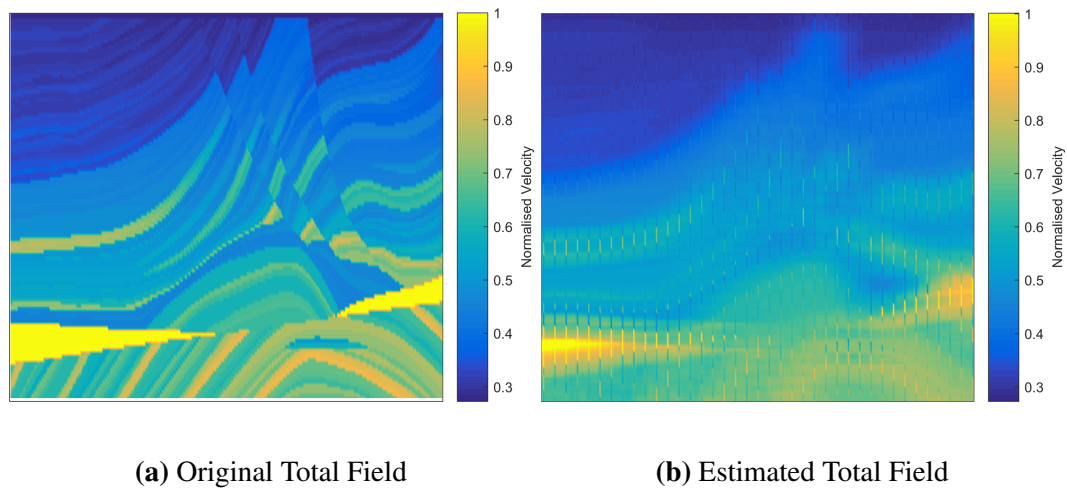


Figure 6.48 Original and estimated total field

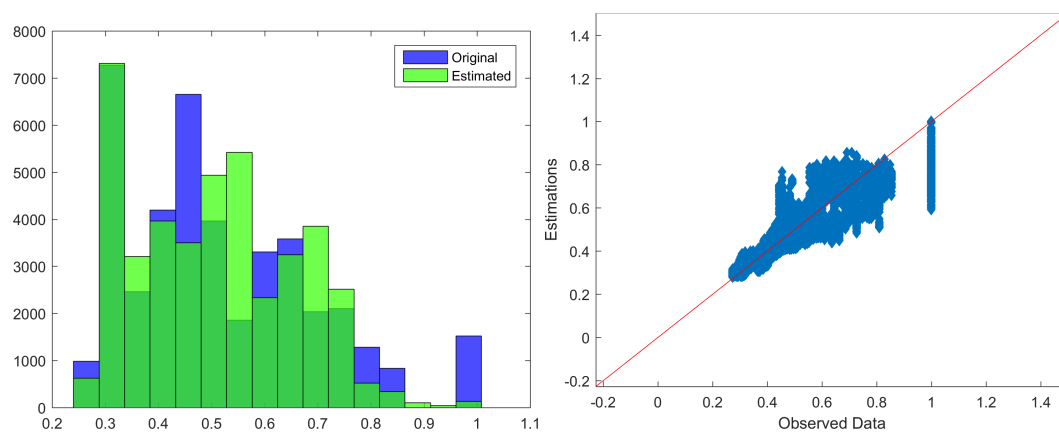


Figure 6.49 Histograms and Scatter plot of original and estimated values (total field)

Table 6.38 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Spherical	0.848	0.864	0.228	0.904	0.929	0.591
Spartan	0.842	0.860	0.234	0.892	0.923	0.598
Gen. Exponential	0.841	0.860	0.235	0.891	0.922	0.599

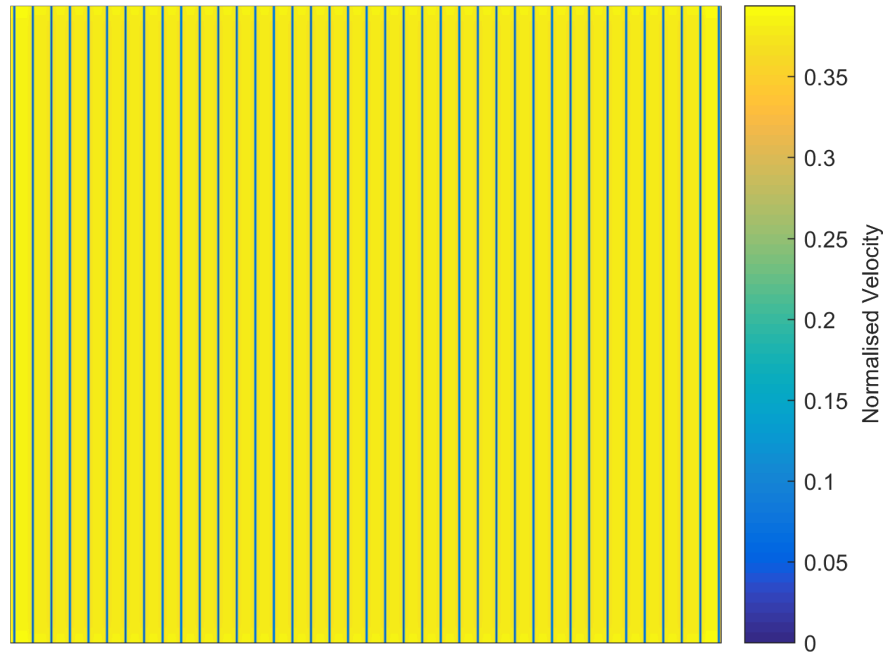


Figure 6.50 Uncertainty of the total field's estimation

CHI2

In the *CHI2* method the anisotropy parameters are also estimated by means of CHI in advance, as in the *CHI1* variation, but in this case these parameters are used for rescaling and rotating the original anisotropic coordinations system to a new isoropic coordinations system. Subsequently, the field is modeled by isotropic variogram equations following a similar to the previous variations procedure (i.e. parameter inference by minimizing the error function, LOOCV and OK).

Therefore, the 1 set of anisotropy parameters estimated in the *CHI1* variation (see Table 6.33) is used for the inversion of the anisotropic effect, giving a new coordination systems. In order to invert the original coordination system it is assumed that $\xi_1 = R$, as long as the CHI method estimates only R and ϕ . Subsequently, the new system is modeled using the 5 isotropic variogram models. Before this, however, it is necessary to examine whether the new system is actually isotropic or have retained anisotropic characteristics. This is obviously dependent on the reliability of the anisotropy parameters estimation. For this purpose, the experimental directional variograms of the new coordinations system is calculated, and then the new anisotropy parameters are estimated by means of DVF (see section 5.2.2) using the 5 variogram models. The results are shown in Table 6.39, while in Fig. 6.51 is displayed the experimental directional variograms of the new coordinations system. All the models give increased anisotropy ratio relatively to the original one, but only

Table 6.39 Anisotropy parameters of the new coordinations systems

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen.			
Exponential	35.297	0.394	6.1°
Gaussian	33.661	0.738	16.6°
Spherical	73.440	0.899	12.4°
Gen. Matérn	42.206	0.465	19.8°
Spartan	47.751	0.535	22.0°

Table 6.40 Optimum Parameters of the isotropic variogram models

Model	σ_z^2	ξ	c_0	ν
Gen.				
Exponential	0.033	32.950	0.000	0.776
Gaussian	0.026	36.473	0.007	—
Spherical	0.026	91.788	0.007	—
Gen. Matérn	0.034	48.7994	0.000	0.322
<hr/>				
	η_0	ξ	c_0	η_1
Spartan	9.687	328.725	0.001	467.369

the two of them recognize it as almost isotropic (i.e. Gaussian and Spherical). Nevertheless, analysis procedure continues by assuming the new coordinations system as isotropic.

By minimizing the error function of the isotropic variogram functions of the models and the corresponding experimental omnidirectional (isotropic) variogram of the new coordinations system (see Fig. 6.52) the parameters of the isotropic models are estimated, as presented in Table 6.40. All the models fit well to the experimental omnidirectional variogram. Also, the estimated parameters are very close, except from the correlation lengths which differ significantly depending on the model; the Spartan model is attributed by the extremely large value of about 329, the Spherical model by the lower value of about 92 while the rest models' correlation length range from 32 to 49.

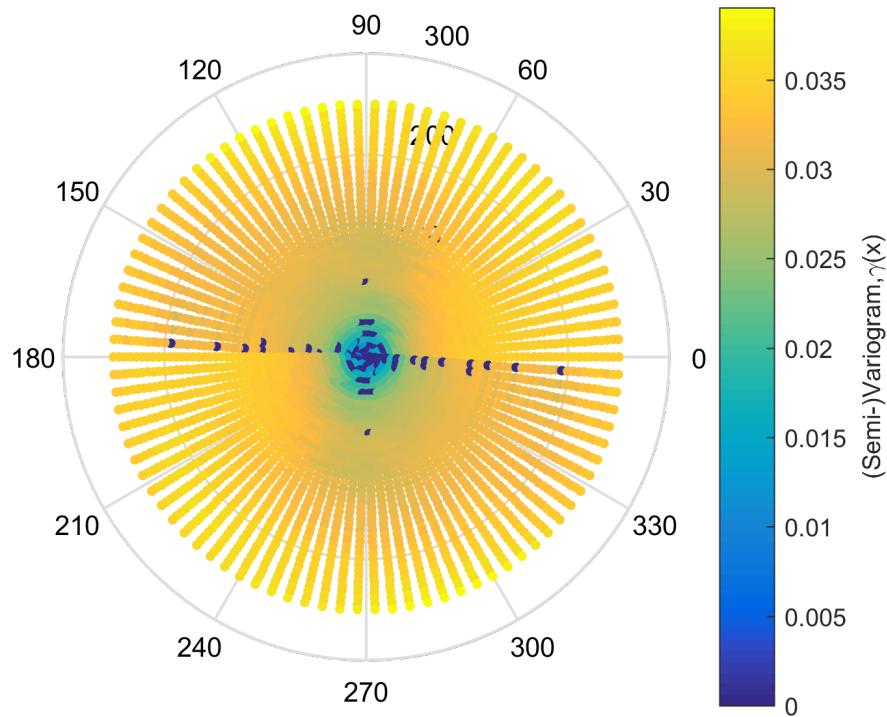
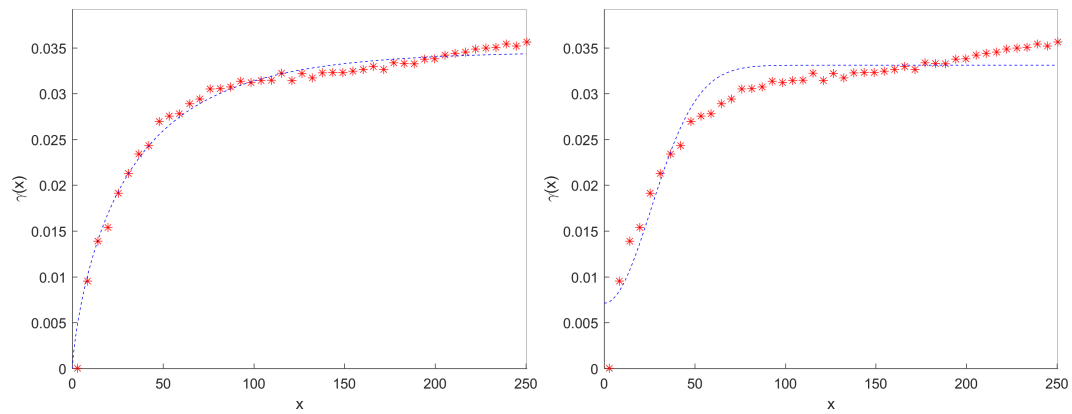


Figure 6.51 Experimental directional variograms of the new coordinations system

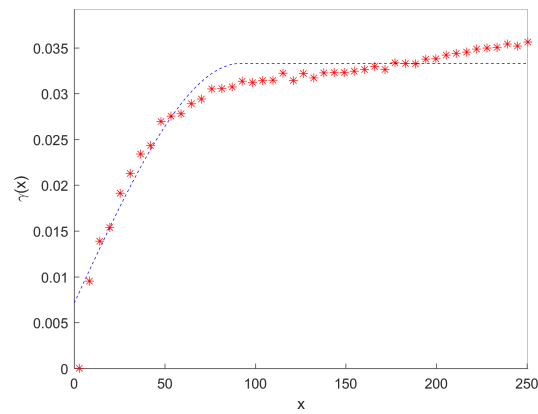
The Leave-One-Out Cross Validation (LOOCV) for the estimated models gives the validation measures presented in Table 6.41. As best model derives the Spartan, followed by Generalized Exponential and Generalized Matérn. As it can be seen from Fig. 6.52), the above mentioned models are those that fit better to the experimental omnidirectional variogram.

The estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) resulting from OK is as shown in Fig. 6.53. The scatter diagram and histograms of the original and the estimated values of the stochastic component are shown in Fig. 6.54, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.42. The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.55, and the corresponding scatter diagram and histograms are shown in Fig. 6.56. In general, the estimations follow the original values achieving a very good proximity of the total distribution. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.43, show that their performance is almost identical. Also, the spatial distribution of the ordinary kriging estimations uncertainty is shown in Fig. 6.57. The uncertainty (as expected from theory) increases quickly from $\simeq 0.00$ near the

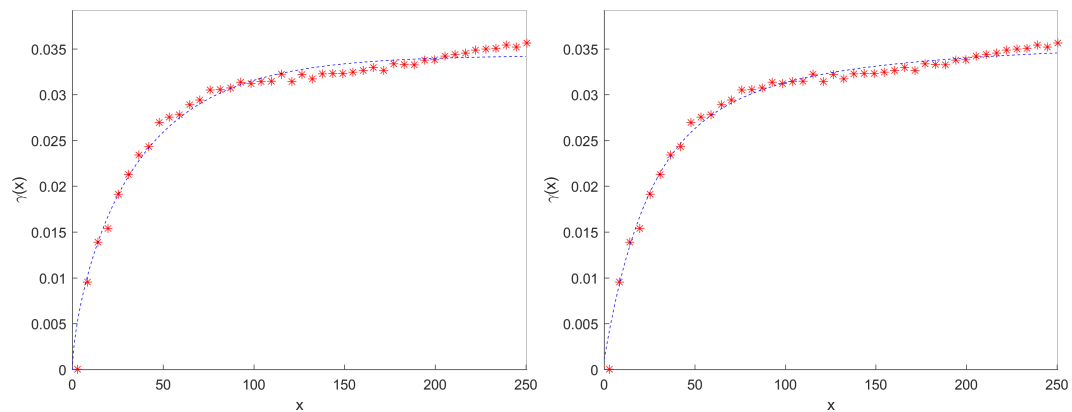


(a) Gen. Exponential

(b) Gaussian



(c) Spherical



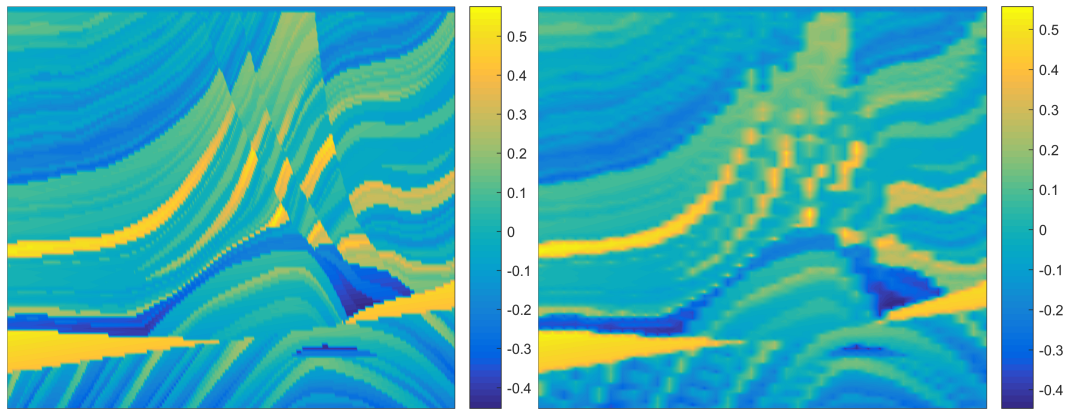
(d) Gen. Matérn

(e) Spartan

Figure 6.52 Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.

Table 6.41 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.030	0.346	0.002	0.049	0.960	0.968	0.897
Gaussian	0.034	0.346	0.003	0.054	0.951	0.960	0.604
Spherical	0.032	0.344	0.003	0.052	0.955	0.964	0.724
Gen. Matérn	0.030	0.346	0.002	0.050	0.959	0.967	0.876
Spartan	0.030	0.347	0.002	0.049	0.960	0.968	0.930

**(a)** Original Stochastic Component**(b)** Estimated Stochastic Component**Figure 6.53** Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 9.687$, $\xi = 328.725$, $c_0 = 0.001$, $\eta_1 = 467.369$.

drill-holes to $\simeq 0.20$ at great distance between the drill-holes, and exhibits the same pattern all over the exained space due to the regularity of the sample.

Finally, in Table 6.44 are presented the classification measures for the cases of 4 and 16 classes. The results show that for all models have almost identical performance, while the measures increase as the number of classes increase.

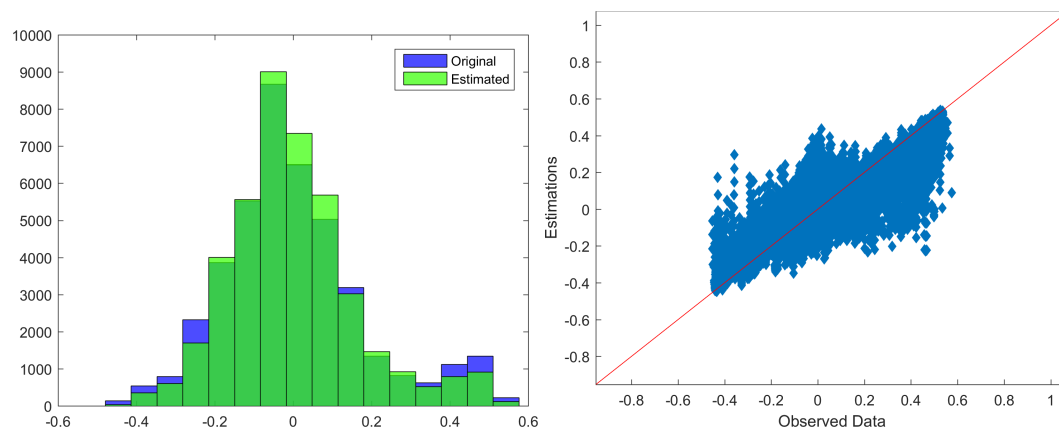


Figure 6.54 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 9.687$, $\xi = 328.725$, $c_0 = 0.001$, $\eta_1 = 467.369$.

Table 6.42 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	0.040	0.695	0.005	0.072	0.920	0.908	0.835
Gen. Exponential	0.041	0.703	0.005	0.072	0.920	0.908	0.831
Gen. Matérn	0.042	0.691	0.005	0.072	0.919	0.908	0.819

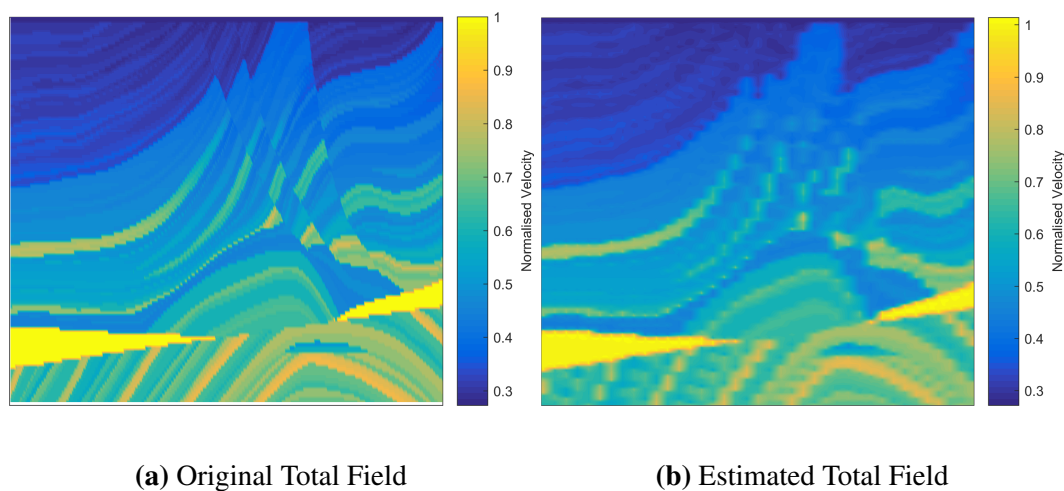


Figure 6.55 Original and estimated total field

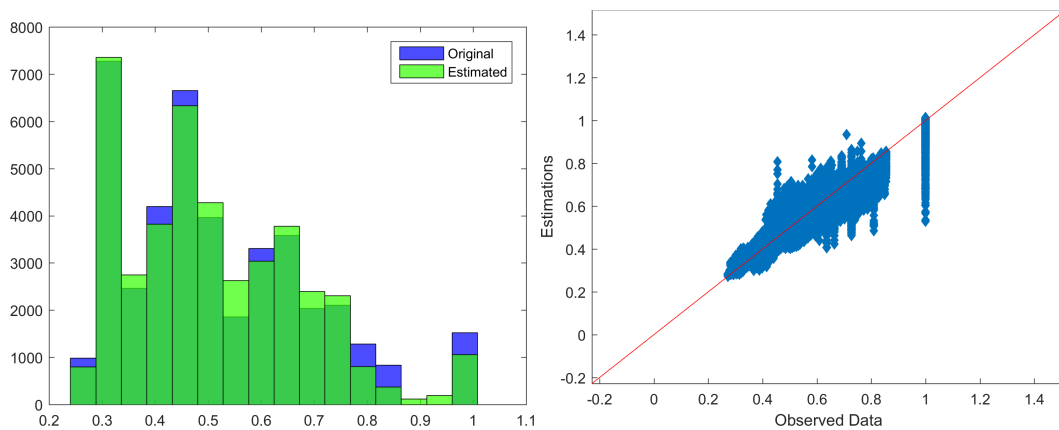


Figure 6.56 Histograms and Scatter plot of original and estimated values (total field)

Table 6.43 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	1.321	1.839	1.803	1.343	0.959	0.980	0.940
Gen. Exponential	1.321	1.839	1.803	1.343	0.960	0.980	0.940
Gen. Matérn	1.321	1.839	1.803	1.343	0.960	0.980	0.940

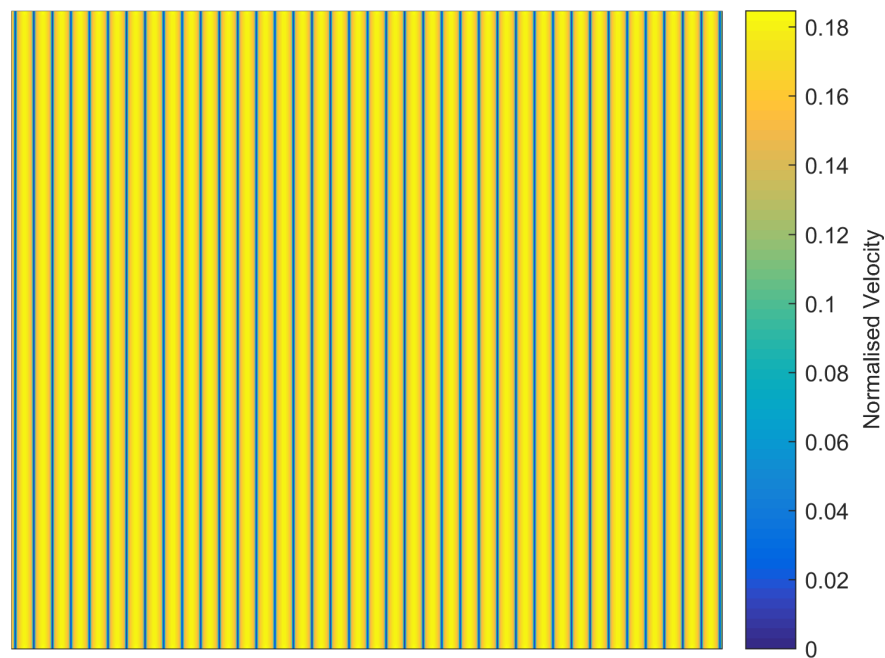


Figure 6.57 Uncertainty of the total field's estimation

Table 6.44 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Spartan	0.930	0.931	0.114	0.974	0.976	0.295
Gen. Exponential	0.930	0.932	0.114	0.974	0.976	0.295
Gen. Matérn	0.929	0.931	0.115	0.974	0.976	0.302

6.4.2 DGC Simulation

The DGC simulation method is applied to the original dataset (i.e normalised velocities) in one step. The simulation is calculated using 4, 16 and 100 classes; in the first two case the result is a classification while in the third the procedure approximates an interpolation. The number of simulations calculated for each case is 10 and the mean of them is the final results, which are interpreted in Figs. 6.58, 6.59, and 6.60 along with the histograms of the original and the estimated data and the distribution of the uncertainty of the estimations. The neighborhood used for the DGC simulation procedure is the same as the one used in OK, i.e 22x4. Finally, some measures of the DGC simulation performance are summarized in Table 6.45. The classification results show that the increasing of the number of classes from 4 to 16 leads to lower errors and higher correlation coefficient. This can be explained by the fact that the field is described better by a more complex model. The misclassification rate, on the contrary, increases with the increasing of the N_c due to the thinning of the bins which affects negatively the accuracy of the estimations. On the other hand, the results obtained by increasing the number of classes to 100 show that the interpolation performance is relatively close to the classification with $N_c = 16$, with slightly higher errors and smaller correlation but significantly higher MCR than the later.

Table 6.45 DGC Validation Measures

Classes	MnAE	RMSE	R_p	MCR
4	0.071	0.090	0.858	0.193
16	0.043	0.072	0.915	0.465
100	0.045	0.079	0.894	0.915

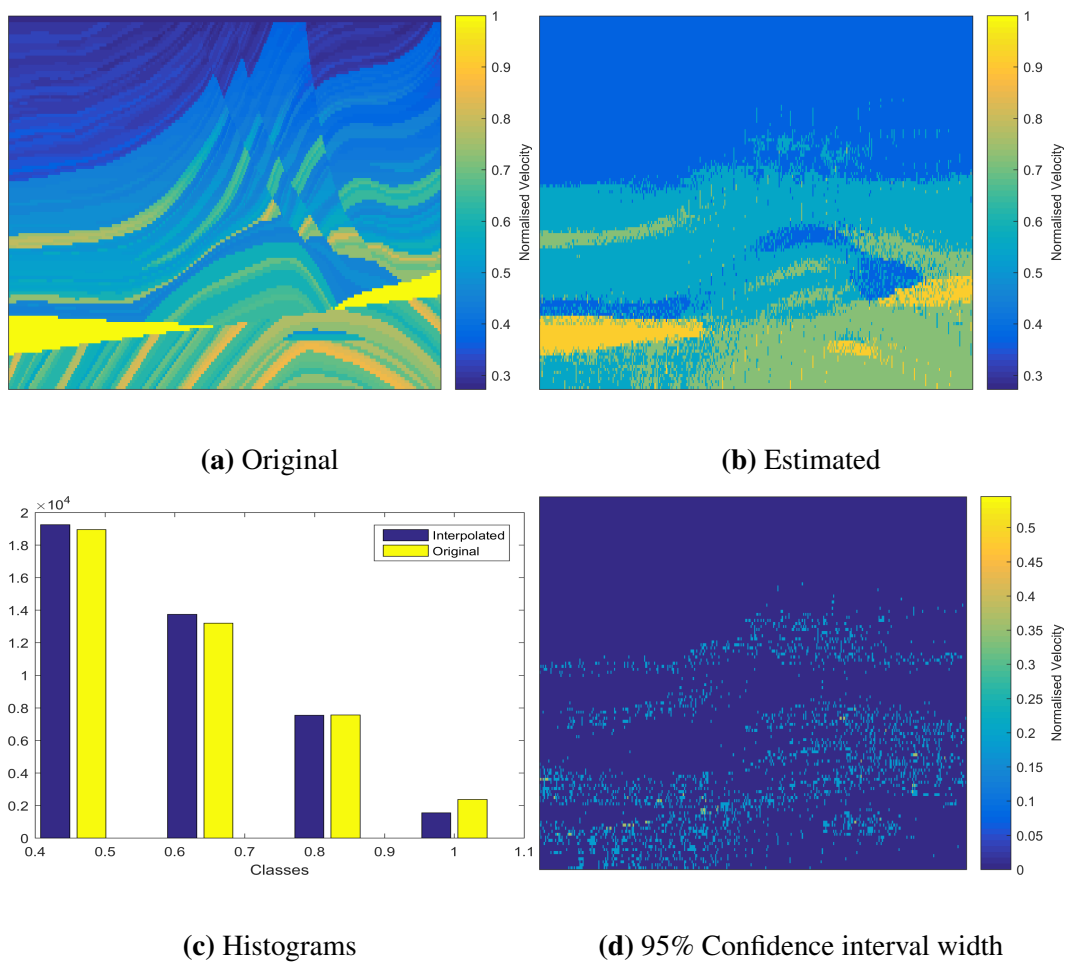


Figure 6.58 Results of DGC simulation with 4 classes

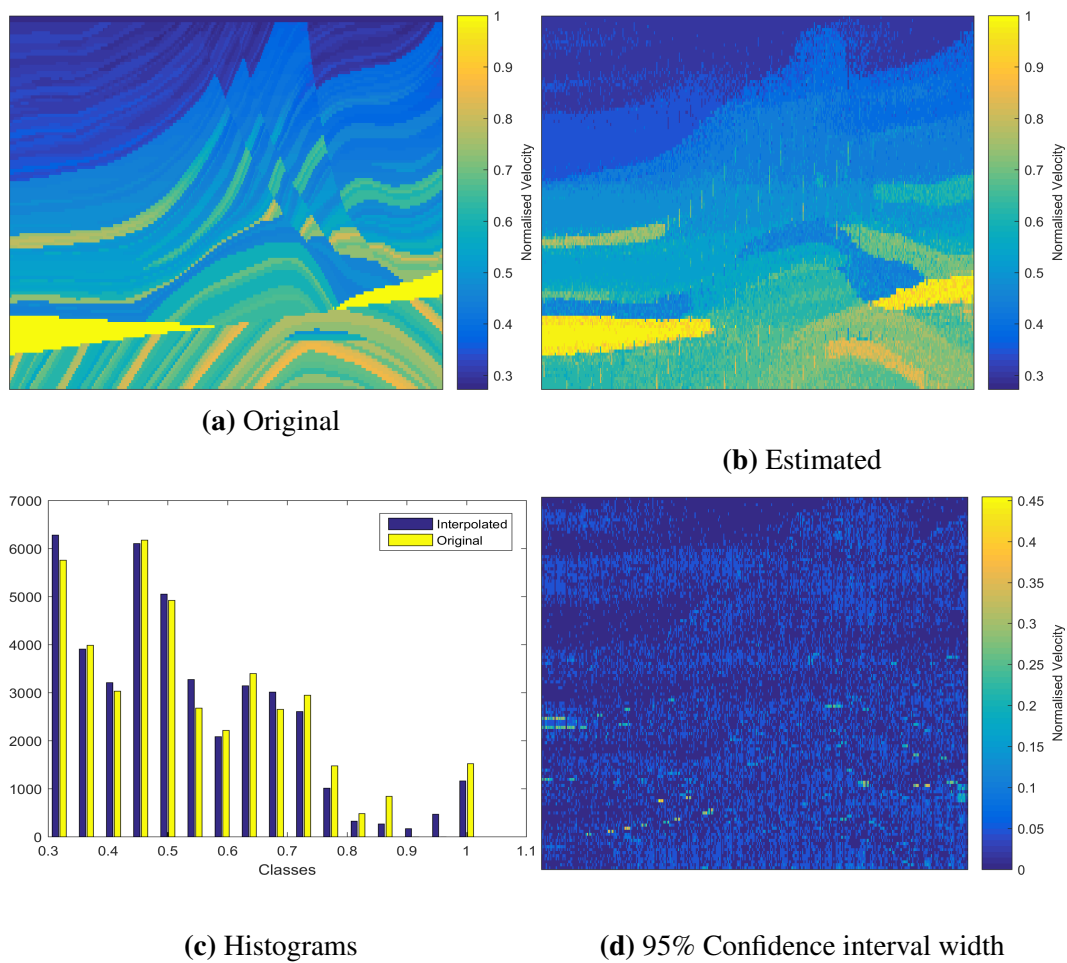


Figure 6.59 Results of DGC simulation with 16 classes

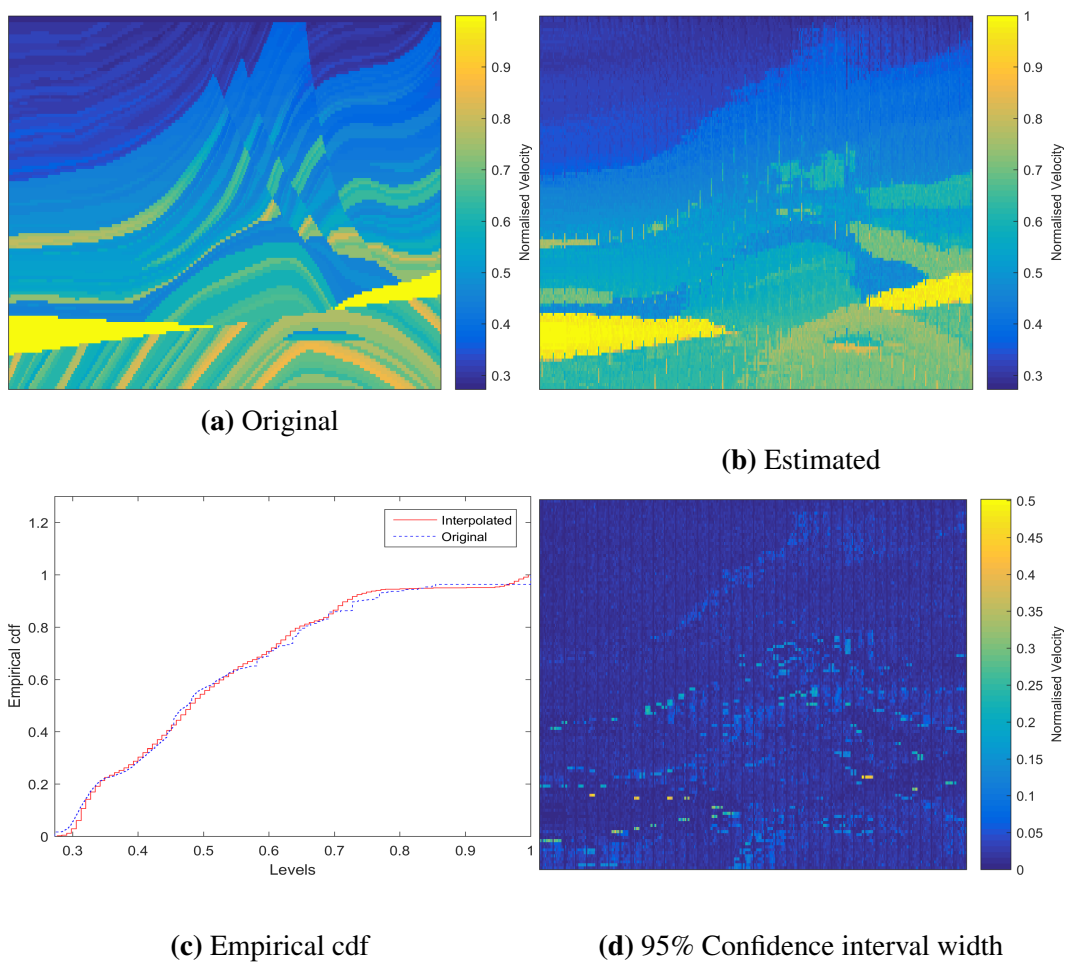


Figure 6.60 Results of DGC simulation with 100 classes

6.4.3 Synopsis

The results of the methods applied to the analysis of the investigated field's regular sample are summarized in Table 6.46. These results show that the best methods for the reconstruction of the Marmousi model are the variations *DirVar2* and *CHI2*, which exhibit equal performance. As regards the investigated models, no one shows significantly better performance. However, the Spartan model is always amongst the three best models. The DGC simulation method has shown comparative performance rates to those of OK estimation, but slightly lower from the later. Notable is also the fact that in all methods the increase of the number of classes results to the improvement of the estimations as indicated by the increase of the correlation coefficients. However, in the DGC method the increase of the classes leads also to the worsening of the misclassification rate, as the complexity of the procedure increases.

Table 6.46 Summarized results for the regular sample

Method	Model	Interpolation			4 Classes		16 Classes	
		MnAE	RMSE	R_p	R_p	MCR	R_p	MCR
DirVar0	Spherical	1.317	1.343	0.935	0.860	0.220	0.916	0.584
	Spartan	1.317	1.343	0.934	0.859	0.222	0.914	0.584
	Gen. Matérn	1.317	1.343	0.930	0.852	0.227	0.906	0.588
DirVar1	Spartan	1.317	1.343	0.932	0.856	0.219	0.912	0.578
	Gen. Matérn	1.317	1.343	0.918	0.840	0.235	0.889	0.600
	Gen. Expon.	1.317	1.343	0.918	0.840	0.235	0.889	0.600
DirVar2	Gen. Expon.	1.321	1.343	0.960	0.929	0.116	0.972	0.317
	Gen. Matérn	1.321	1.343	0.960	0.927	0.119	0.971	0.330
	Spartan	1.318	1.342	0.961	0.891	0.173	0.953	0.464
CHI1	Spherical	1.317	1.343	0.929	0.848	0.228	0.904	0.591
	Spartan	1.317	1.343	0.920	0.842	0.234	0.892	0.598
	Gen. Expon.	1.317	1.343	0.920	0.841	0.235	0.891	0.599
CHI2	Spartan	1.321	1.343	0.959	0.930	0.114	0.974	0.295
	Gen. Expon.	1.321	1.343	0.960	0.930	0.114	0.974	0.295
	Gen. Matérn	1.321	1.343	0.960	0.929	0.115	0.974	0.302
DGC	—	0.045	0.079	0.894	0.858	0.193	0.915	0.465

6.5 Random Sample of Marmousi Model

The analysis of the random sample follows the same procedure as for the regular sample.

6.5.1 Ordinary Kriging

Preliminary Analysis

The statistics (Table 6.47) and the histograms (Fig. 6.61) of the sample and the original dataset show that the sample is totally representative of the original data.

The normal probability plot of the sample (Fig. 6.62), however, indicates that the data does not come from a gaussian distribution. Thus, Box-Cox transformation is necessary to be applied to the sample dataset. The best Box-Cox transformation is attained when $\lambda = -0.2944$. The statistics of the transformed values can be seen in Table 6.48, while the histogram and the normal probability plot of them are presented in Fig. 6.63. The Box-Cox transformation does not provides a truly gaussian dataset, but the transformed dataset is closer to the gaussian distribution than the original sample. However, no further actions are taken and the analysis procedure is continued under the assumption that the transformed dataset comes from a gaussian distribution.

After transforming the random sample's values, the next step is the removal of any possible trend. The trend models described in Section 5.1 (see Table 5.1) have been tested. The parameters of these models are inferred by means of multilinear regression of the transformed normalised velocities field on the (i, j) indices of the sample points. The complete expressions of the resulting trend functions are given in Table 6.49, while the normal probability plots of the fluctuations of each model are shown in Fig. 6.64. The best model is choosen as the one resulting in fluctuations which are closer to the gaussian distribution. This model, as can be seen from Fig. 6.64, is the linear trend model.

In Fig. 6.65 the multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points is depicted, while in Fig. 6.66 is plotted the histogram of the transformed and detrended data (i.e. the fluctuations), and in Table 6.50 are evaluated the statistics of them. Finally, in Fig. 6.67 is presented the transformed and detrended

Table 6.47 Original and sample datasets statistics

Dataset	Min	Max	Mean	Median	Variance	Skewness	Kurtosis
Original	0.273	1.000	0.514	0.475	0.030	0.824	3.277
Sample	0.273	1.000	0.512	0.471	0.030	0.851	3.326

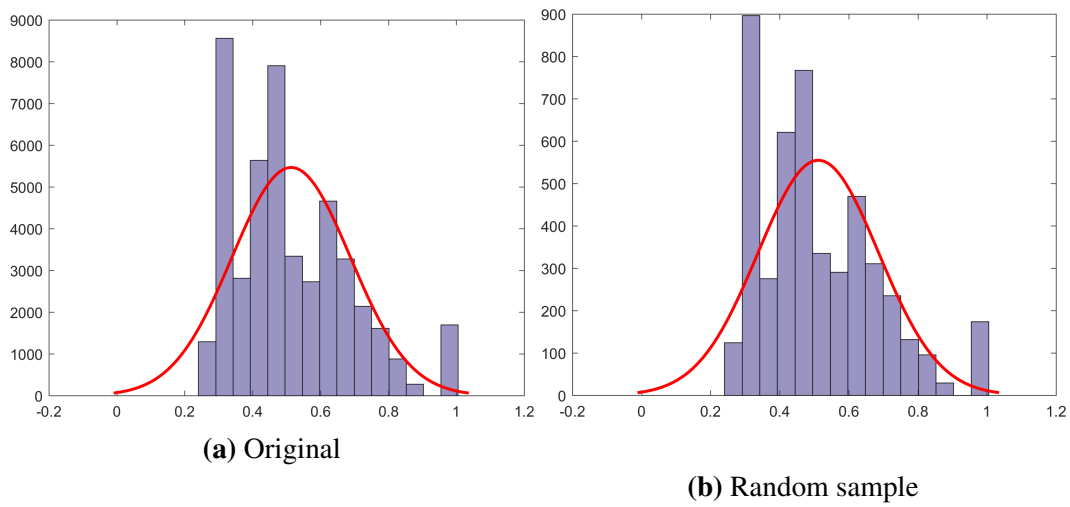


Figure 6.61 Histograms of original and sample datasets

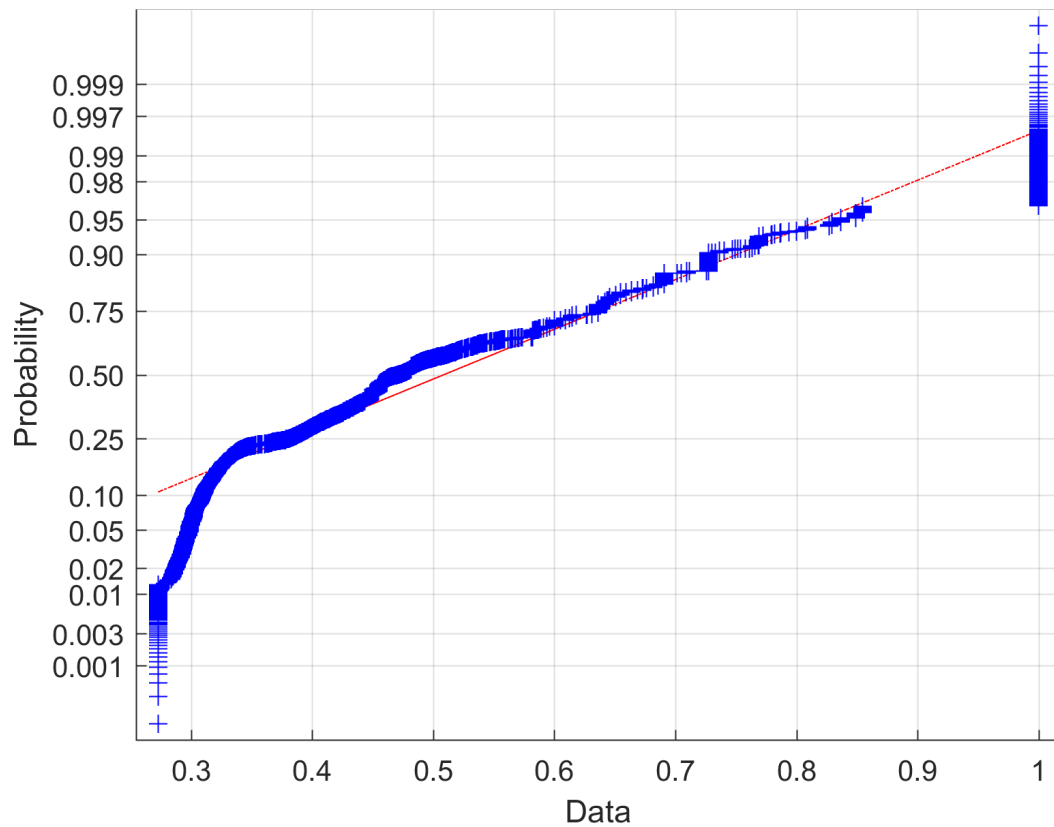


Figure 6.62 Normal probability plot of random sample

Table 6.48 Transformed (Box-Cox with $\lambda = -0.2606$) dataset statistics

Min	Max	Mean	Median	Variance	Skewness	Kurtosis
-1.583	0.000	-0.827	-0.844	0.164	0.028	2.126

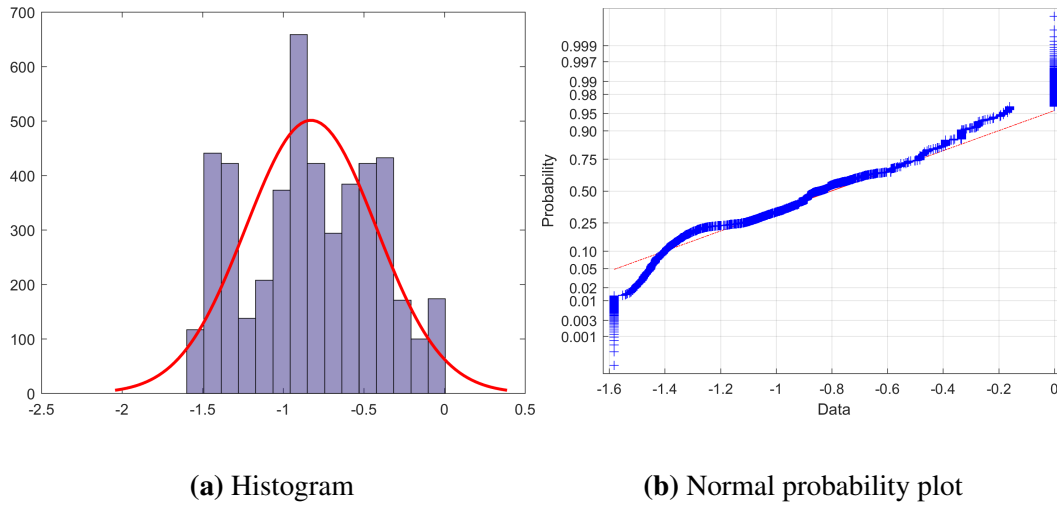


Figure 6.63 Histogram and normal probability plot of the transformed random sample. The transformation applied is the Box-Cox with $\lambda = -0.2944$.

Table 6.49 Estimated trend models (random sample)

Model	Estimated Trend Function
Mean	$m_z(\mathbf{s}) = -0.8271$
Linear	$m_z(\mathbf{s}) = -0.3023 + 5.2017 \cdot 10^{-4}x - 0.0101y$
Quadratic	$m_z(\mathbf{s}) = -0.3143 - 2.5325 \cdot 10^{-5}x - 0.0067y + 6.0531 \cdot 10^{-6}xy$ $+ 4.4887 \cdot 10^{-7}x^2 - 3.7038 \cdot 10^{-5}y^2$
Cubic	$m_z(\mathbf{s}) = -0.1126 - 0.0037x - 0.0085y + 5.4460 \cdot 10^{-5}xy + 1.5396 \cdot 10^{-5}x^2$ $- 9.0123 \cdot 10^{-5}y^2 - 6.7206 \cdot 10^{-8}x^2y$ $- 1.8463 \cdot 10^{-7}xy^2 - 1.8824 \cdot 10^{-8}x^3 + 4.8737 \cdot 10^{-7}y^3$
Quartic	$m_z(\mathbf{s}) = -0.1699 - 0.0068x + 0.0063y + 2.3499 \cdot 10^{-5}xy + 5.3995 \cdot 10^{-5}x^2$ $- 4.9044 \cdot 10^{-4}y^2 - 1.4725 \cdot 10^{-7}x^2y + 8.3151 \cdot 10^{-7}xy^2$ $- 1.6992 \cdot 10^{-7}x^3 + 3.9641 \cdot 10^{-6}y^3 + 7.9058 \cdot 10^{-10}x^2y^2$ $- 2.3328 \cdot 10^{-11}x^3y - 7.2008 \cdot 10^{-9}xy^3 + 1.9912 \cdot 10^{-10}x^4$ $- 8.4700 \cdot 10^{-9}y^4$
Linear+Periodic	$m_z(\mathbf{s}) = -0.3038 + 5.1872 \cdot 10^{-4}x - 0.0101y - 0.0036 \cos 2\pi 0.0256x$ $- 0.0046 \sin 2\pi 0.0256x - 0.0016 \cos 2\pi 0.0656y + 0.0217 \sin 2\pi 0.0656y$ $- 0.0041 \cos 2\pi 0.3846x + 0.0026 \sin 2\pi 0.3846x$ $- 0.0184 \cos 2\pi 0.0738y + 0.0034 \sin 2\pi 0.0738y$

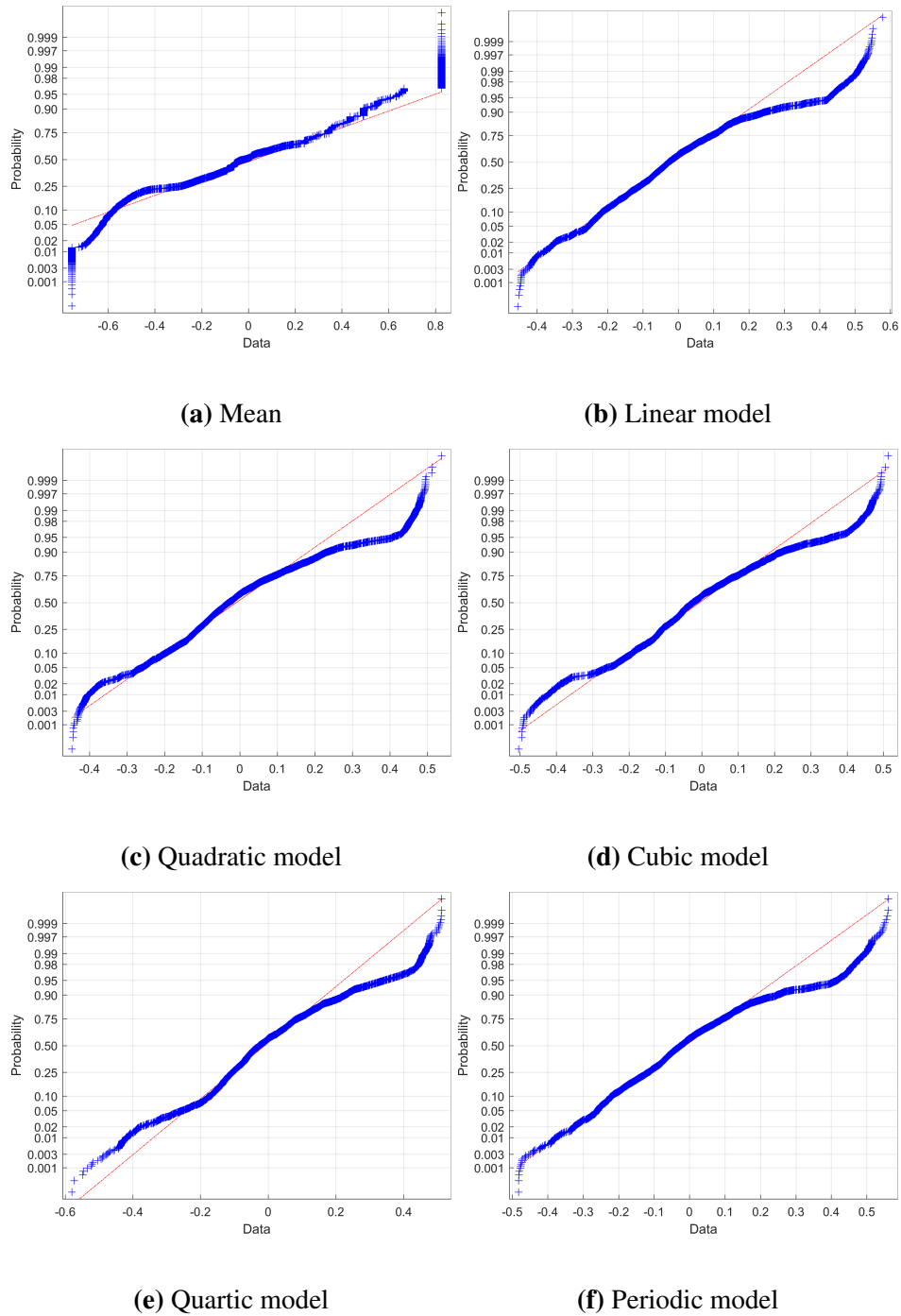


Figure 6.64 Normal probability plots of the fluctuations resulting from the estimated trend models.

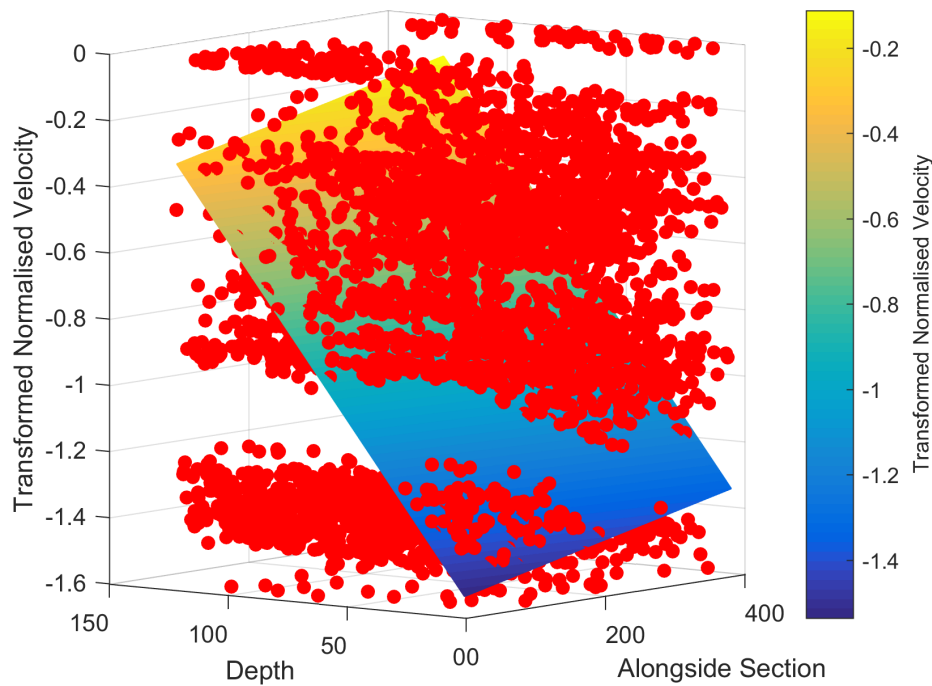


Figure 6.65 Multilinear regression of the transformed normalised velocities field on the indices (i, j) of the sample points. The trend equation is given by $m_z(\mathbf{s}) = -0.3023 + 5.2017 \cdot 10^{-4}x - 0.0101y$.

Table 6.50 Transformed and detrended dataset statistics

Min	Max	Mean	Median	Variance	Skewness	Kurtosis
-0.454	0.579	0.000	-0.027	0.035	0.714	3.660

Marmousi model, in order to obtain an inspection of the random field, which we intend to regenerate hereinafter with the five variations of the analysis procedure, described in Section 6.2.

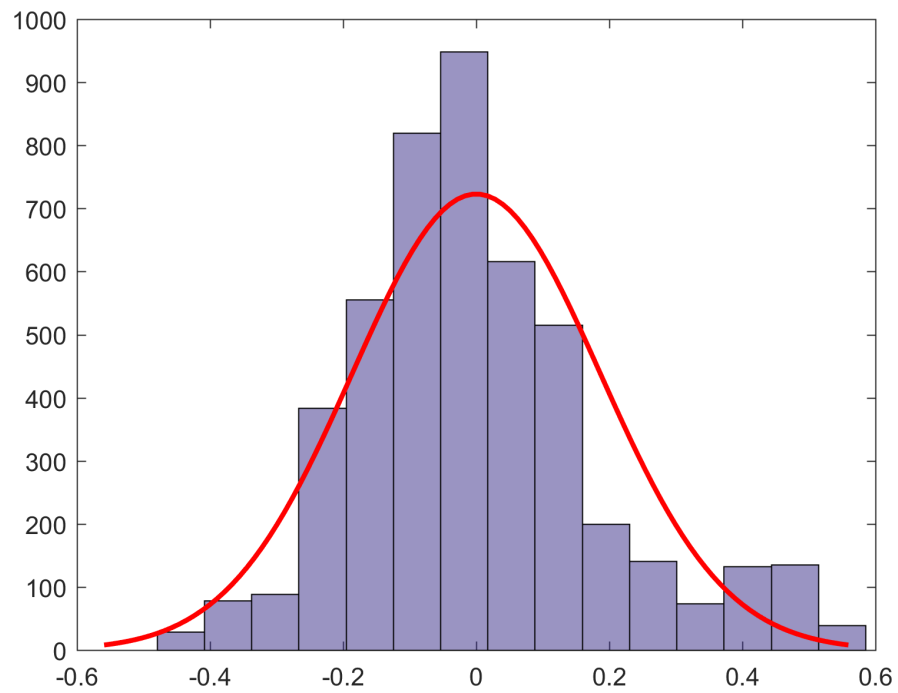


Figure 6.66 Histogram of transformed and detrended sample dataset

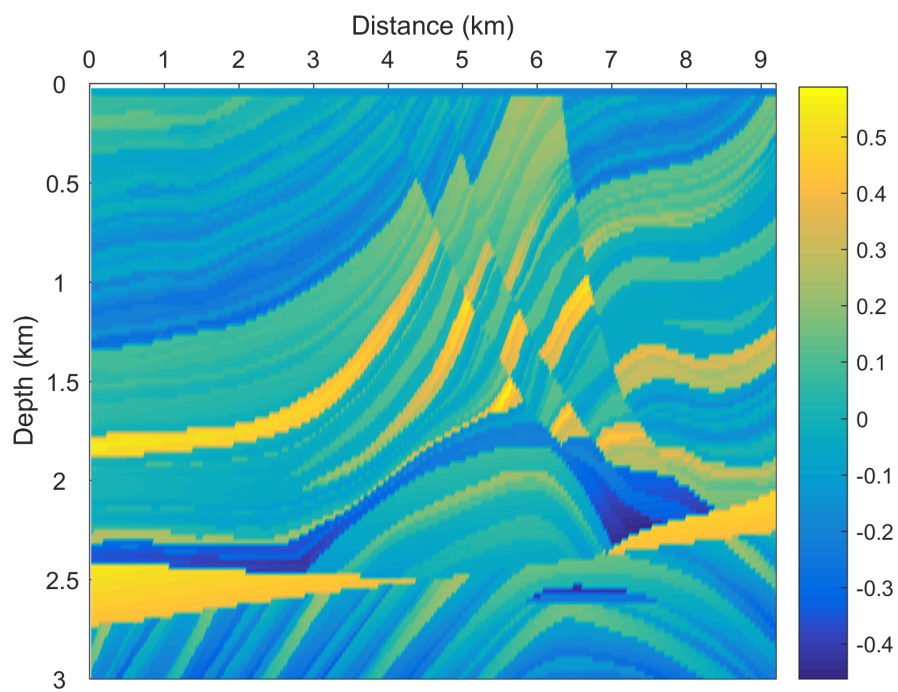


Figure 6.67 Transformed and detrended Marmousi model

Calculation of Experimental Variogram

The experimental directional variograms with angular step 4° and 15° are shown in Fig. 6.68. These figures display clearly the significant anisotropic characteristics of the field (as it is highly expected for such geological data). More specifically, it can be easily seen that the spatial correlation differs significantly from direction to direction; the variogram is stabilized around a maximum value (sill) in longer distances along the horizontal and almost horizontal directions than the other directions.

Again the experimental variograms with the smaller angular step (4°) have been calculated only in order to provide a clearer visualization of the anisotropy of the field, while the directional experimental variograms with the higher angular step (15°) are used in practice in the following steps. We avoid to use the experimental variograms with the small angular step due to the higher computational cost that they will lead the further analysis.

DirVar0

The initial values and the boundaries of the parameters for each model used in the optimization step are presented in Table 6.51, and the resulting optimum parameters are presented in Table 6.52.

As can be seen from the validation measures of the LOOCV presented in Table 6.53 the best model is the Generalized Exponential, followed by Spartan and Spherical.

A visualisation of the the selected theoretical model's fitting to the experimental variogram is interpreted in Fig. 6.69, in which the directional experimental variograms of the field on the directions $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° are plotted against the theoretical model (the rest directional variograms can be found in A). The best anisotropic model fits fairly well to the directional experimental variograms.

The estimation derived from OK of the stochastic component of the field (i.e the transformed and detrended normalised velocities) given by the best model is shown in Fig. 6.70. A scatter diagram of original and estimated values of the stochastic component, as well as their histograms can be seen in Fig. 6.71. Finally, in Table 6.54 the measures of the stochastic component estimation performance for the three best models are presented.

By adding the trend to the estimations of the stochastic component of the field and inverting the Box-Cox transformation, the estimation of the total field, shown in Fig. 6.72, is derived. For the stochastic component, a scatter diagram of the original and the estimated values of the total field, as well as their histograms can be seen in Fig. 6.73. In general, the estimations follow the original values without achieving satisfying proximity of the

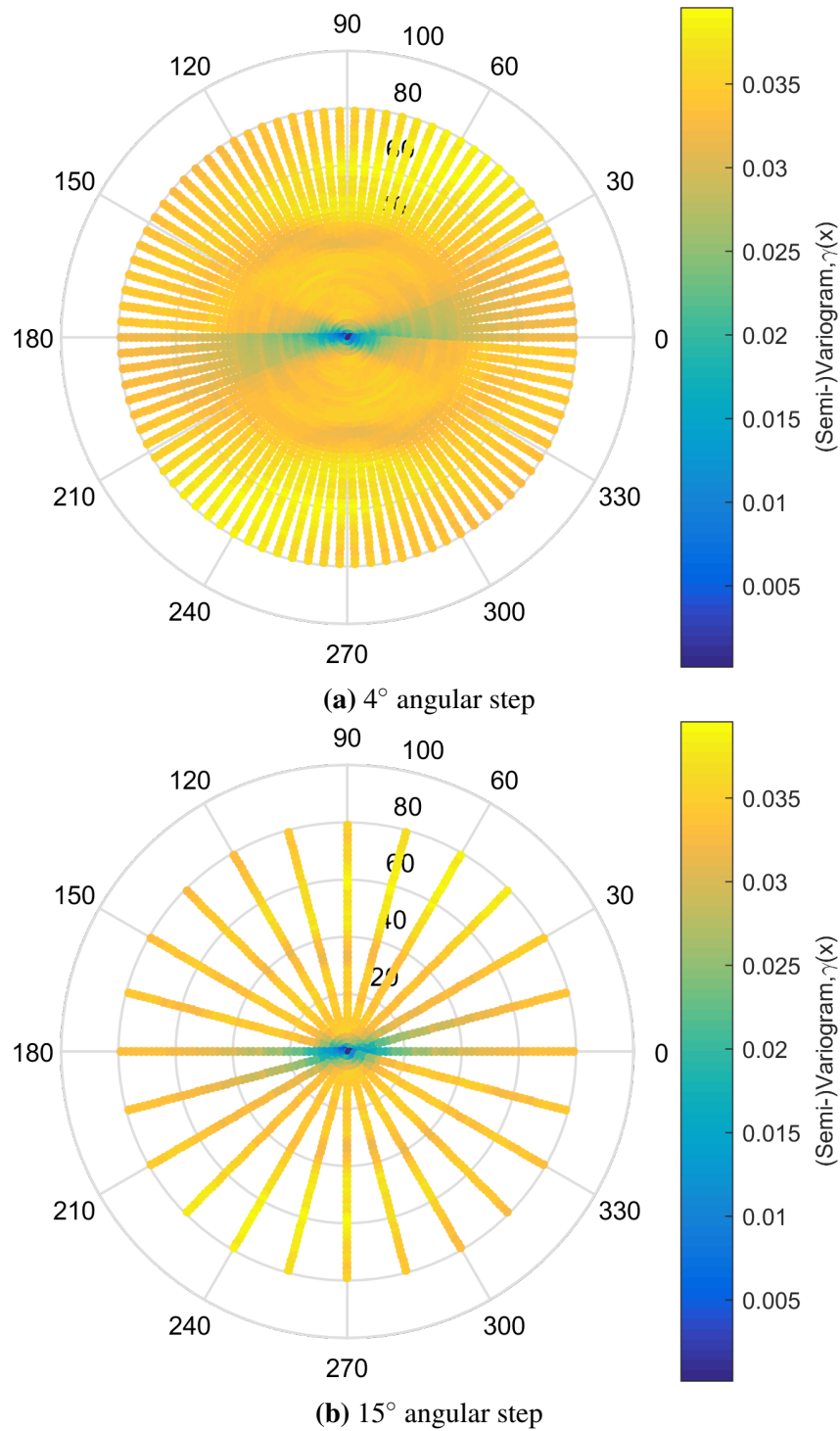


Figure 6.68 Directional experimental variograms of the field. For both cases the angular tolerance is 20°, the maximum distance taken into account is the 20% of the grid's diagonal which is equivalent to about 80, and is divided into 45 distance lags.

Table 6.51 Initial values and boundaries of the parameters for optimization step, where \hat{g}_{max} (\equiv maximum value of experimental variogram) = 0.0384, $d_{max} \simeq 80$, $b = [\sigma_z^2 = \hat{g}_{max}, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_{sp} = [\eta_0 = 1000, \xi_1 = d_{max}2/3, R = 0.5, \phi = 10^\circ, c_0 = \hat{g}_{max}/100]$, $b_b = [\sigma_z^2 \in [0, 1.5\hat{g}_{max}], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$, $b_{sp,b} = [\eta_0 \in [0, \infty], \xi_1 \in [0, 1.5d_{max}], R \in [0, 30], \phi \in [-90^\circ, 90^\circ], c_0 \in [0, \hat{g}_{max}/5]]$.

Model	Initial Values	Boundaries
Gen.		
Exponential	$[b, \nu = 1.5]$	$[b_b, \nu \in (0, 2)]$
Gaussian	$[b]$	$[b_b]$
Spherical	$[b]$	$[b_b]$
Gen. Matérn	$[b]$	$[b_b]$
Spartan	$[b_{sp}, \eta_1 = 1]$	$[b_{sp,b}, \eta_1 \in (-2, \infty)]$

Table 6.52 Optimum Parameters of the investigated variogram models

Model	σ_z^2	ξ_1	R	ϕ	c_0	ν
Gen.						
Exponential	0.035	9.671	0.370	-83.4°	0.000	0.734
Gaussian	0.034	108.181	0.901	-83.0°	0.008	—
Spherical	0.027	8.663	0.116	-83.9°	0.008	—
Gen. Matérn	0.035	9.848	0.227	-83.4°	0.000	0.300
	η_0	ξ_1	R	ϕ	c_0	η_1
Spartan	5.094	9.852	0.082	-83.5°	0.000	123.852

Table 6.53 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.045	0.405	0.005	0.070	0.917	0.945	0.866
Gaussian	0.052	0.392	0.007	0.076	0.895	0.928	0.640
Spherical	0.046	0.401	0.005	0.072	0.913	0.941	0.782
Gen. Matérn	0.046	0.391	0.005	0.072	0.912	0.941	0.781
Spartan	0.046	0.400	0.005	0.072	0.913	0.941	0.784

Table 6.54 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.090	0.772	0.017	0.130	0.721	0.719	0.519
Spartan	0.096	0.582	0.017	0.131	0.719	0.711	0.488
Spherical	0.097	0.572	0.017	0.132	0.715	0.707	0.472

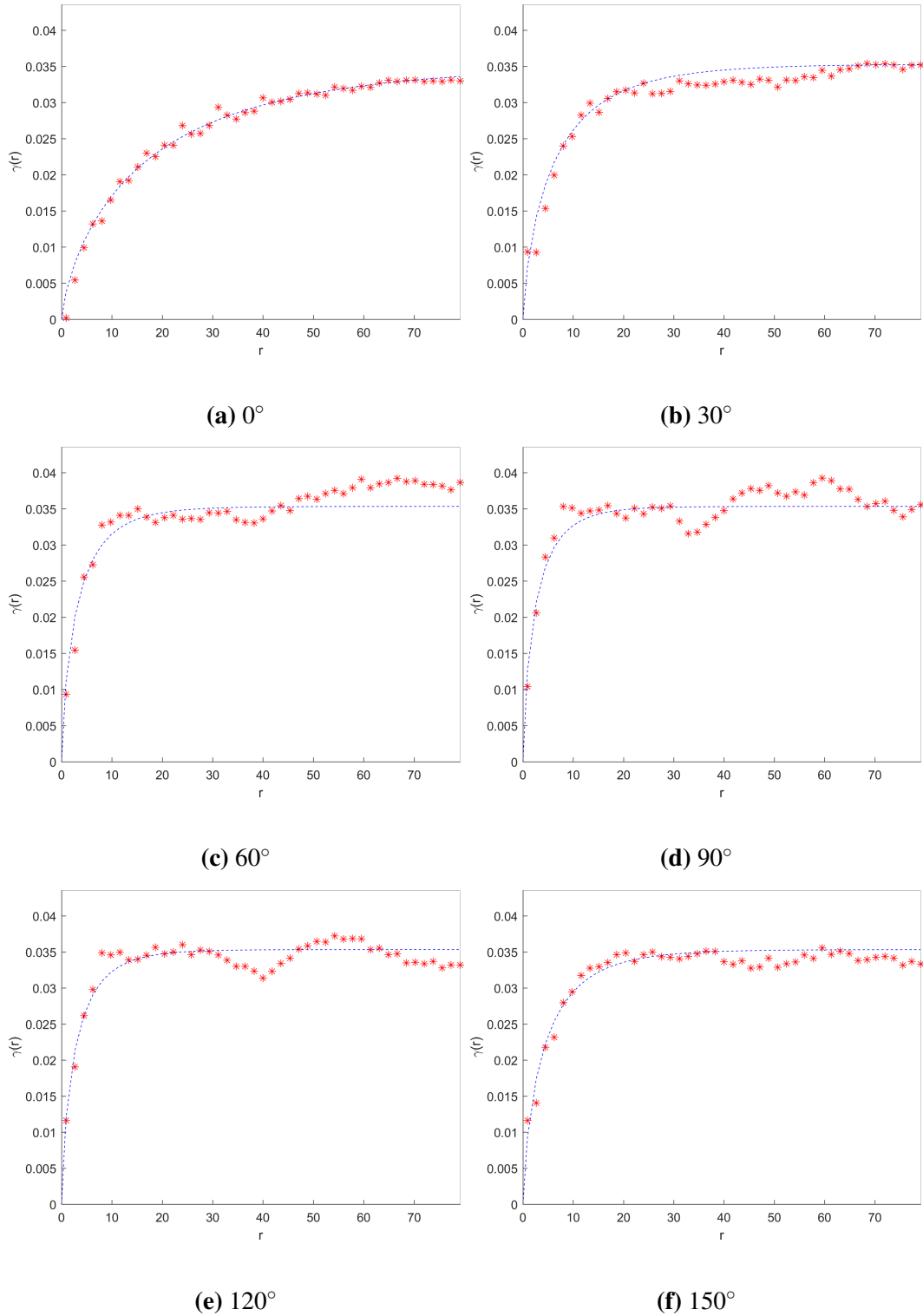
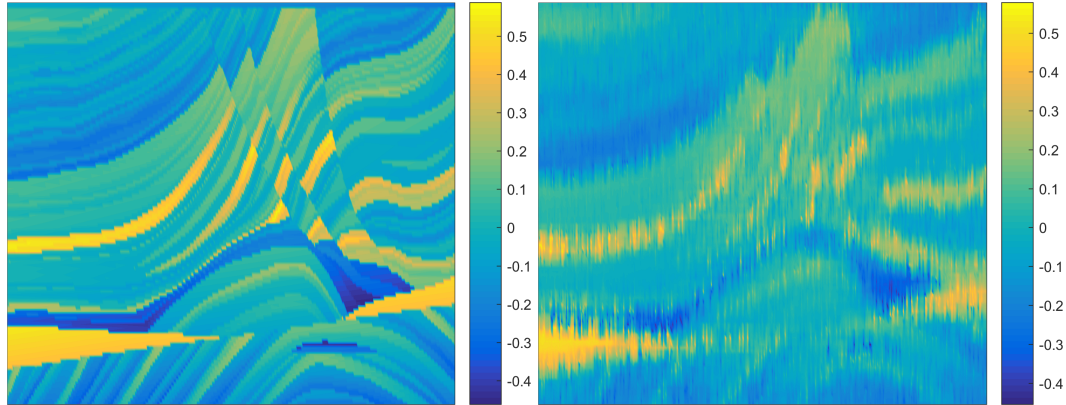


Figure 6.69 Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.035$, $\xi_1 = 9.671$, $R = 0.370$, $\phi = -83.4^\circ$, $c_0 = 0.000$, $\nu = 0.734$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.



(a) Original Stochastic Component

(b) Estimated Stochastic Component

Figure 6.70 Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.035$, $\xi_1 = 9.671$, $R = 0.370$, $\phi = -83.4^\circ$, $c_0 = 0.000$, $\nu = 0.734$.

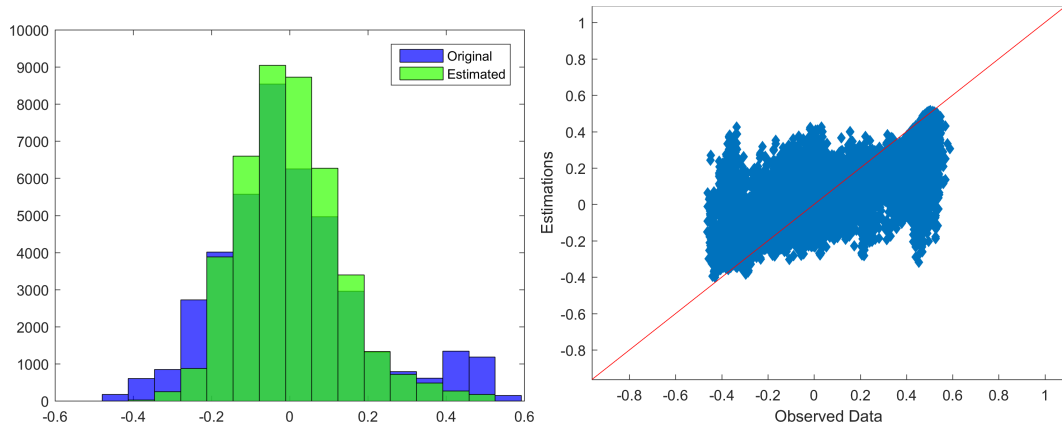


Figure 6.71 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.035$, $\xi_1 = 9.671$, $R = 0.370$, $\phi = -83.4^\circ$, $c_0 = 0.000$, $\nu = 0.734$.

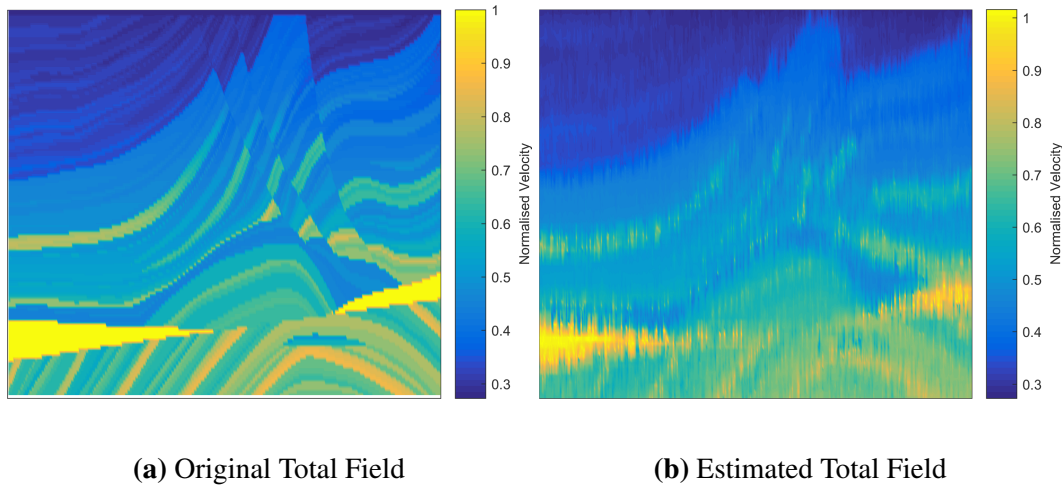


Figure 6.72 Original and estimated total field

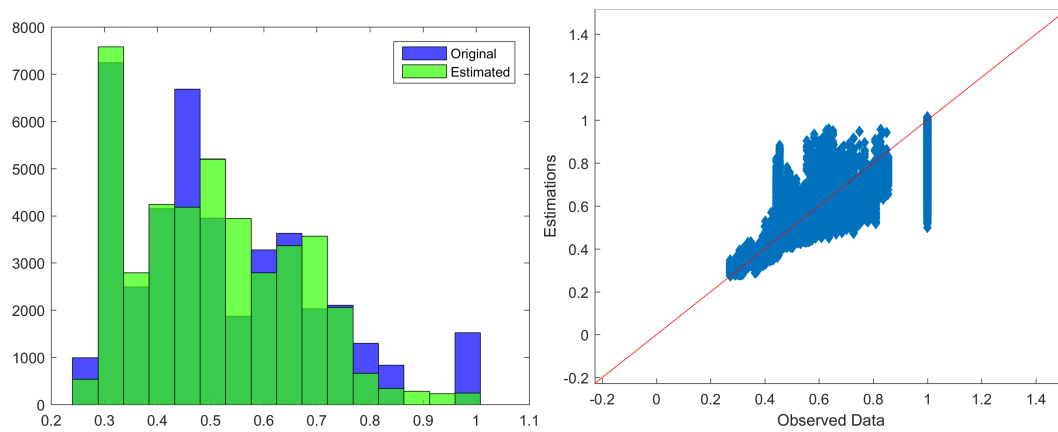


Figure 6.73 Histograms and Scatter plot of original and estimated values (total field)

total distribution. Both the tails of the distribution as the middle of it exhibit significant discrepancies.

In Table 6.55 the measures of the total field (i.e. normalised velocities) estimation performance for the three best models are presented. From these measures it can be observed that the three model's performance is almost identical with the last two performing slightly better than the first. In addition, Fig. 6.74 maps the confidence interval width of the estimations, i.e. it depicts the spatial distribution of the uncertainty. The uncertainty is zero at the known points, while it increases up to 0.37 relatively quickly with the distance of the missing locations from the known points. Also, at the edges of the field of the uncertainty is increased due to the lack of available neighbors.

Table 6.55 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen.							
Exponential	1.331	1.932	1.842	1.357	0.927	0.943	0.872
Spartan	1.329	1.905	1.840	1.357	0.931	0.940	0.875
Spherical	1.329	1.904	1.840	1.357	0.930	0.939	0.874

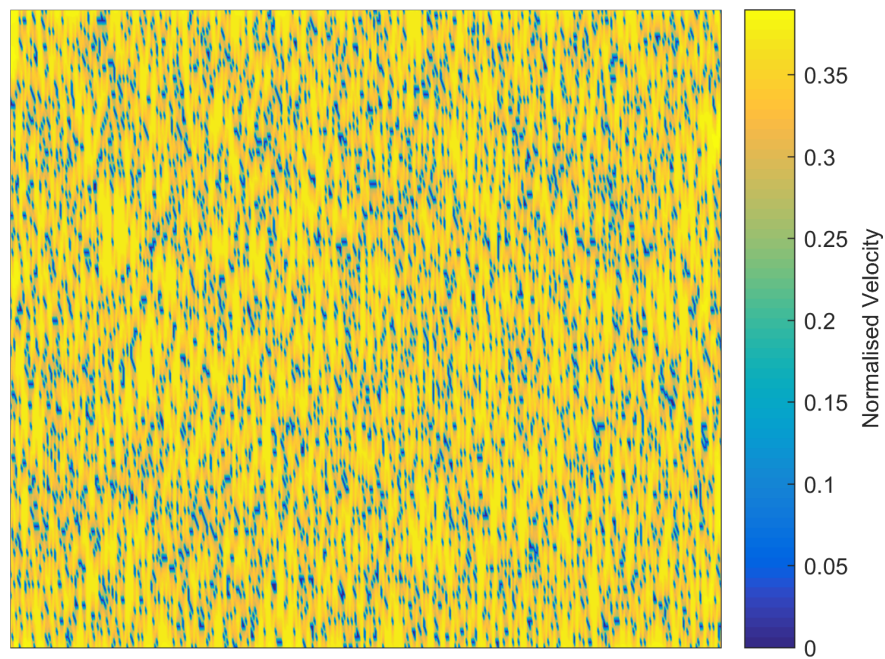
**Figure 6.74** Uncertainty of the total field's estimation

Table 6.56 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Gen.						
Exponential	0.857	0.876	0.208	0.912	0.938	0.537
Spartan	0.855	0.870	0.219	0.911	0.934	0.579
Spherical	0.855	0.870	0.219	0.911	0.933	0.580

In Table 6.56 are presented the classification measures for the cases of 4 and 16 classes. The increasing of the number of classes leads to the correlation coefficients increasing and to the MCR decreasing. The correlation coefficients' improvement can be attributed to the more realistic model which derives by increasing the N_c . On the contrary, the misclassification rate increases due to the thinning of the binning, which incommodes the interpolation. Also, the first model exhibits the best performance.

DirVar1

The fitting of the ellipses to the pairs of (ϕ, ξ) of each model are depicted in Fig. 6.75, while the estimated anisotropy parameters for all models by means of DVF (see section 5.2.2) are given in Table 6.57. In general, the investigated models' fitting show that the major axis of the anisotropy ellipsis is either on the almost horizontal direction (0°) or on the direction of about $30 - 40^\circ$. This results agree with the intuitive estimations of the anisotropy angle mentioned at section 6.1.2. The rapid changes of the correlation lengths depending on the direction (due to non-stationarity) may also affect negatively the optimization procedure. On the other hand, the estimated anisotropy ratios cannot be evaluated before the cross validation procedure.

By replacing the estimated anisotropy parameters to the anisotropic variogram models and minimizing the error function of the new models and the experimental directional variograms the rest parameters are estimated, as presented in Table 6.58. The common parameters generally agree for all the investigated models.

After the parameter inference Leave-One-Out Cross Validation (LOOCV) is applied in order to define the best models. The validation measures of the LOOCV, presented in Table 6.59, give as best model the Spartan, followed by the Generalized Matérn and the Generalized Exponential models.

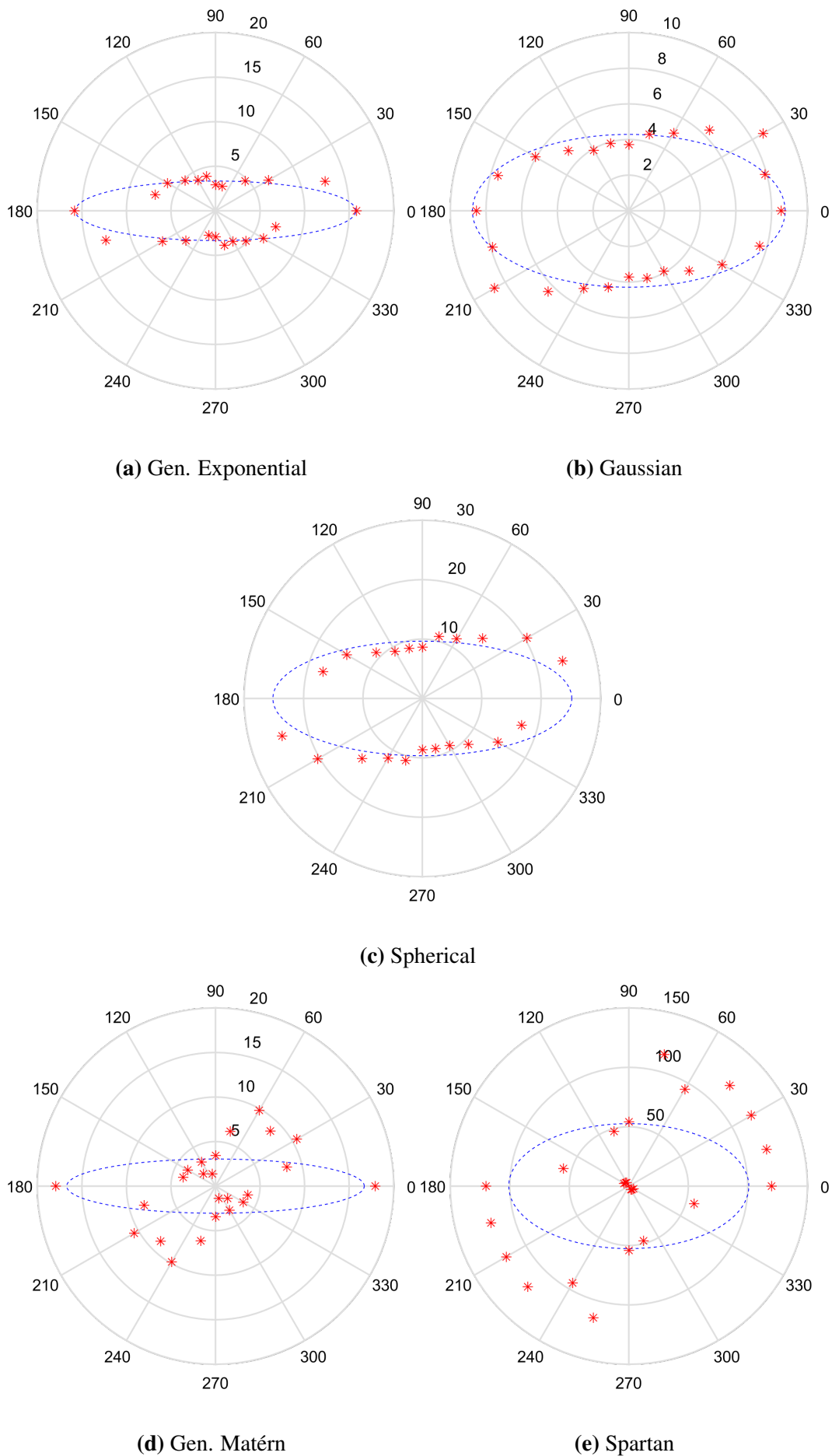


Figure 6.75 Fitting of ellipses to the pairs (ϕ, ξ) of the investigated variogram models

Table 6.57 Anisotropy parameters of the investigated variogram models estimated with DVF method

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	3.338	0.212	-90.0°
Gaussian	4.288	0.489	-90.0°
Spherical	9.645	0.383	-90.0°
Gen. Matérn	3.043	0.182	-90.0°
Spartan	52.606	0.522	-50.9°

Table 6.58 Optimum Parameters of the variogram models with the lower degrees of freedom

Model	σ_z^2	c_0	ν
Gen. Exponential	0.030	0.004	1.095
Gaussian	0.026	0.008	—
Spherical	0.026	0.008	—
Gen. Matérn	0.032	0.002	0.499
	η_0	c_0	η_1
Spartan	0.010	0.008	-1.999

Table 6.59 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.046	0.385	0.005	0.072	0.912	0.941	0.816
Gaussian	0.046	0.383	0.005	0.073	0.911	0.940	0.810
Spherical	0.046	0.383	0.005	0.073	0.911	0.940	0.809
Gen. Matérn	0.046	0.385	0.005	0.072	0.912	0.941	0.817
Spartan	0.045	0.403	0.005	0.072	0.914	0.943	0.862

Table 6.60 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	0.097	0.829	0.021	0.145	0.639	0.655	0.295
Gen. Matérn	0.092	0.792	0.018	0.133	0.701	0.700	0.491
Gen. Exponential	0.093	0.794	0.018	0.134	0.699	0.699	0.484

The fitting of the best model to the experimental variogram along the directions of $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° is interpreted in Fig. 6.76 (the rest directional variograms can be found in Appendix A). As it can be seen the best theoretical variogram model diverges significantly from the experimental directional variograms. This can be attributed to minimization miscalculations, i.e. inappropriate objective function (very smooth or possible local minima), inappropriate initial values or to the fields complexity. Thus, the analysis is continued without taking any further action.

Implementing OK with the determined best model, the resulting estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) is as shown in Fig. 6.77. The scatter diagram and histograms of the original and the estimated values of the stochastic component are illustrated in Fig. 6.78, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.60.

The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.79, and the correspondng scatter diagram and histograms are shown in Fig. 6.80. In general, the estimations follow the original values without achieving satisfying proximity of the total distribution. Both the tails of the distribution as the middle of it exhibit significant discrepancies. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.61, show that the three model's performance is almost identical with the last two performing slightly better than the first. Finally, Fig. 6.81 shows the spatial distribution of the ordinary kriging estimations uncertainty. The uncertainty is zero at the known points and increases up to 0.37 gradually with the distance of the investigated points from the data.

Also, in Table 6.62 are presented the classification measures for the cases of 4 and 16 classes. For both $N_c = 4$ and $N_c = 16$ the performance of the two last models is almost identical and slightly better than the first one. Also, in the case of $N_c = 16$ the classification performs better than in the case of $N_c = 4$.

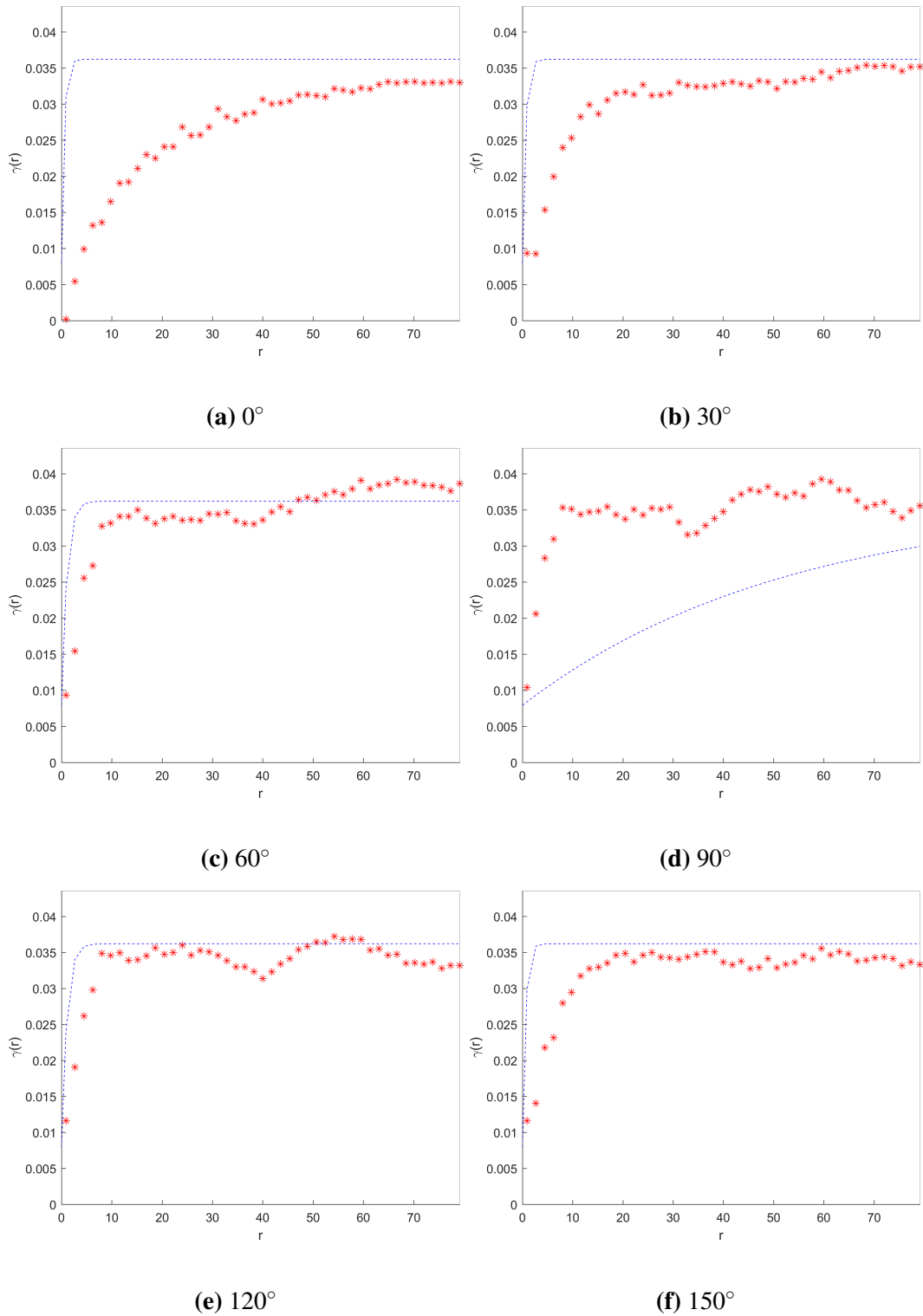


Figure 6.76 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.010$, $\xi_1 = 52.606$, $R = 0.522$, $\phi = -90.0^\circ$, $c_0 = 0.008$, $\eta_1 = -1.999$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

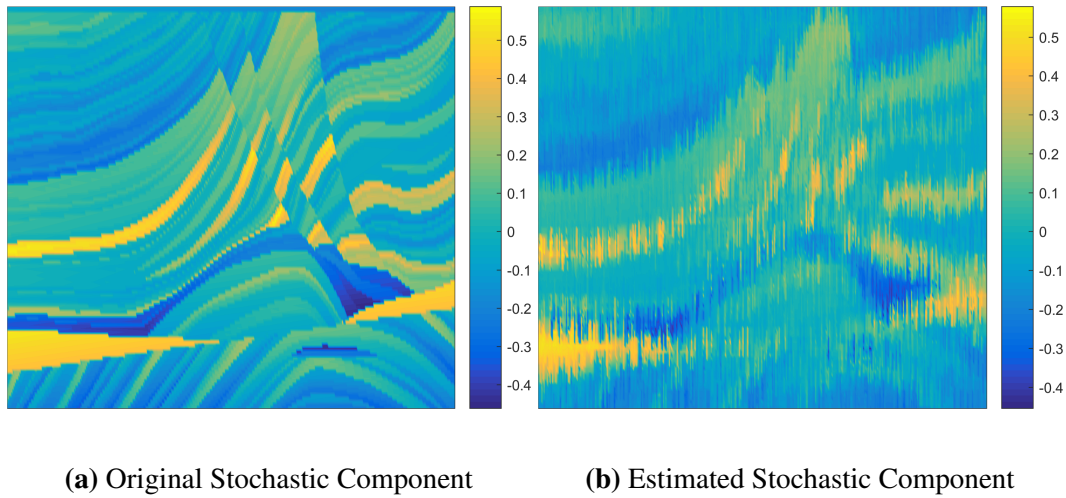


Figure 6.77 Original and Estimation of the stochastic component of the field. The model used is a Spartan with parameters $\eta_0 = 0.010$, $\xi_1 = 52.606$, $R = 0.522$, $\phi = -90.0^\circ$, $c_0 = 0.008$, $\eta_1 = -1.999$.

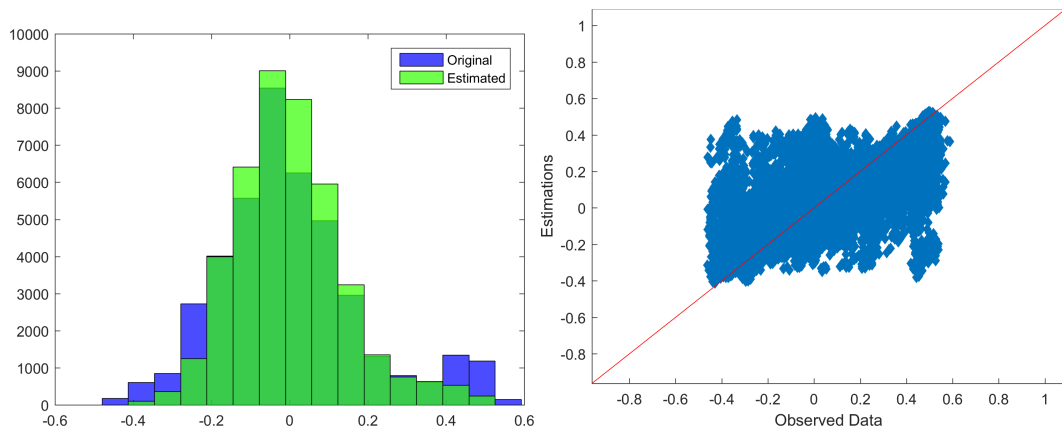


Figure 6.78 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Spartan with parameters $\eta_0 = 0.010$, $\xi_1 = 52.606$, $R = 0.522$, $\phi = -90.0^\circ$, $c_0 = 0.008$, $\eta_1 = -1.999$.

Table 6.61 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Spartan	1.331	1.945	1.846	1.359	0.904	0.930	0.838
Gen. Matérn	1.331	1.932	1.843	1.357	0.923	0.939	0.867
Gen. Exponential	1.331	1.932	1.843	1.357	0.923	0.939	0.866

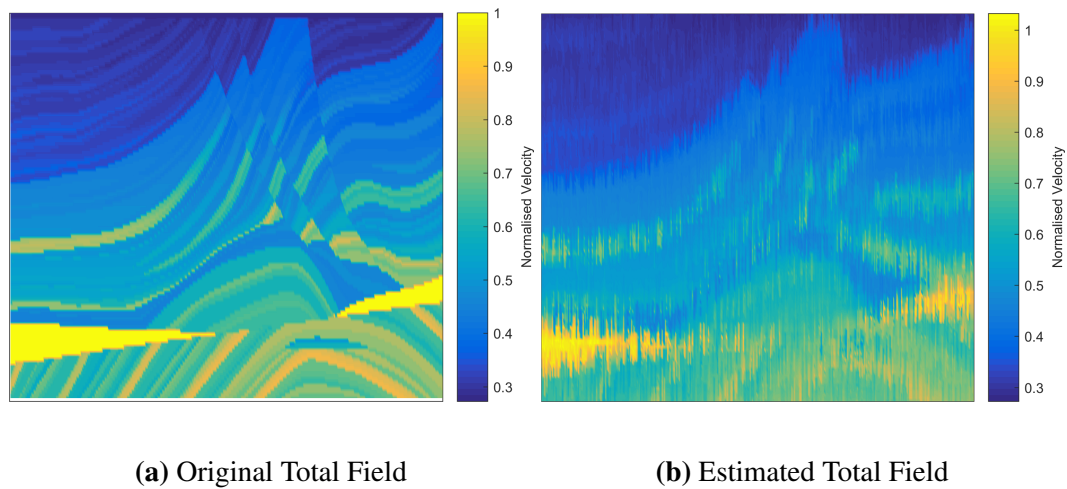


Figure 6.79 Original and estimated total field

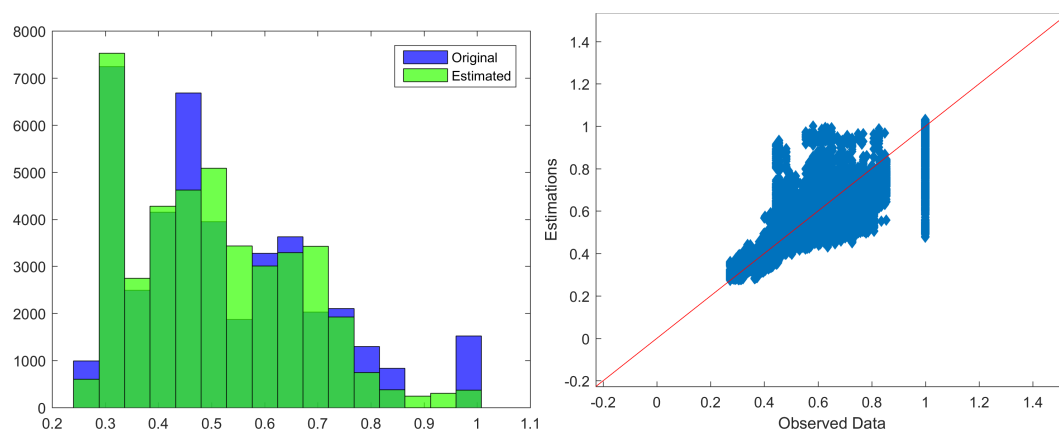


Figure 6.80 Histograms and Scatter plot of original and estimated values (total field)

Table 6.62 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Spartan	0.838	0.864	0.221	0.888	0.925	0.537
Gen. Matérn	0.854	0.873	0.211	0.907	0.934	0.543
Gen. Exponential	0.853	0.873	0.211	0.907	0.934	0.543

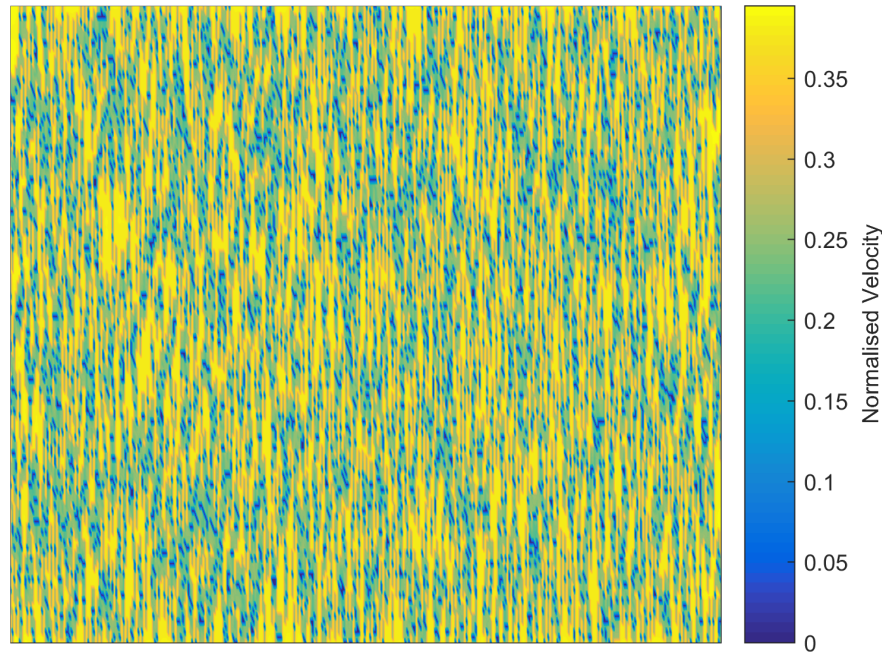


Figure 6.81 Uncertainty of the total field's estimation

DirVar2

The anisotropy parameters of the new coordinations system, estimated by means of DVF, are shown in Table 6.63, while in Fig. 6.82 are displayed the experimental directional variograms of them. The results show that despite small improvement (indicated by the proximity of the new correlation length to 1 and the new anisotropy angle to 0) the anisotropic effect remains at relatively high levels as indicated by the very small increase of the anisotropy ratios. Nevertheless, analysis procedure continues by assuming the new coordinations systems as isotropic.

By minimizing the error function of the isotropic variogram functions of the models and the corresponding experimental omnidirectional (isotropic) variograms of the new coordinations systems (see Fig. 6.83) the parameters of the isotropic models are estimated, as presented in Table 6.64. The parameters of the investigated models generally agree, except from the correlation length given by the Spartan which is significantly smaller than the other models.

The Leave-One-Out Cross Validation (LOOCV) for the estimated models gives the validation measures presented in Table 6.65. As best model derives the Generalized Matérn, followed by Generalized Exponential and Spherical. As it can be seen from Fig. 6.83), though, all the models except Spartan and Gaussian fit fairly well to their corresponding experimental omnidirectional variograms.

Table 6.63 Anisotropy parameters of the new coordinations systems

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	1.055	0.564	-0.5°
Gaussian	0.906	0.305	4.3°
Spherical	0.940	0.325	3.4°
Gen. Matérn	1.007	0.293	5.8°
Spartan	1.320	0.269	6.1°

Table 6.64 Optimum Parameters of the isotropic variogram models

Model	σ_z^2	ξ	c_0	ν
Gen. Exponential	0.034	1.128	0.000	1.013
Gaussian	0.033	0.797	0.000	—
Spherical	0.026	1.219	0.006	—
Gen. Matérn	0.033	1.780	0.001	0.509
	η_0	ξ	c_0	η_1
Spartan	0.044	0.089	0.000	-1.989

Table 6.65 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.032	0.430	0.003	0.057	0.945	0.964	0.883
Gaussian	0.037	0.364	0.003	0.053	0.955	0.980	0.607
Spherical	0.036	0.420	0.004	0.061	0.937	0.960	0.661
Gen. Matérn	0.032	0.371	0.003	0.057	0.946	0.964	0.912
Spartan	0.034	0.437	0.004	0.061	0.937	0.961	0.656

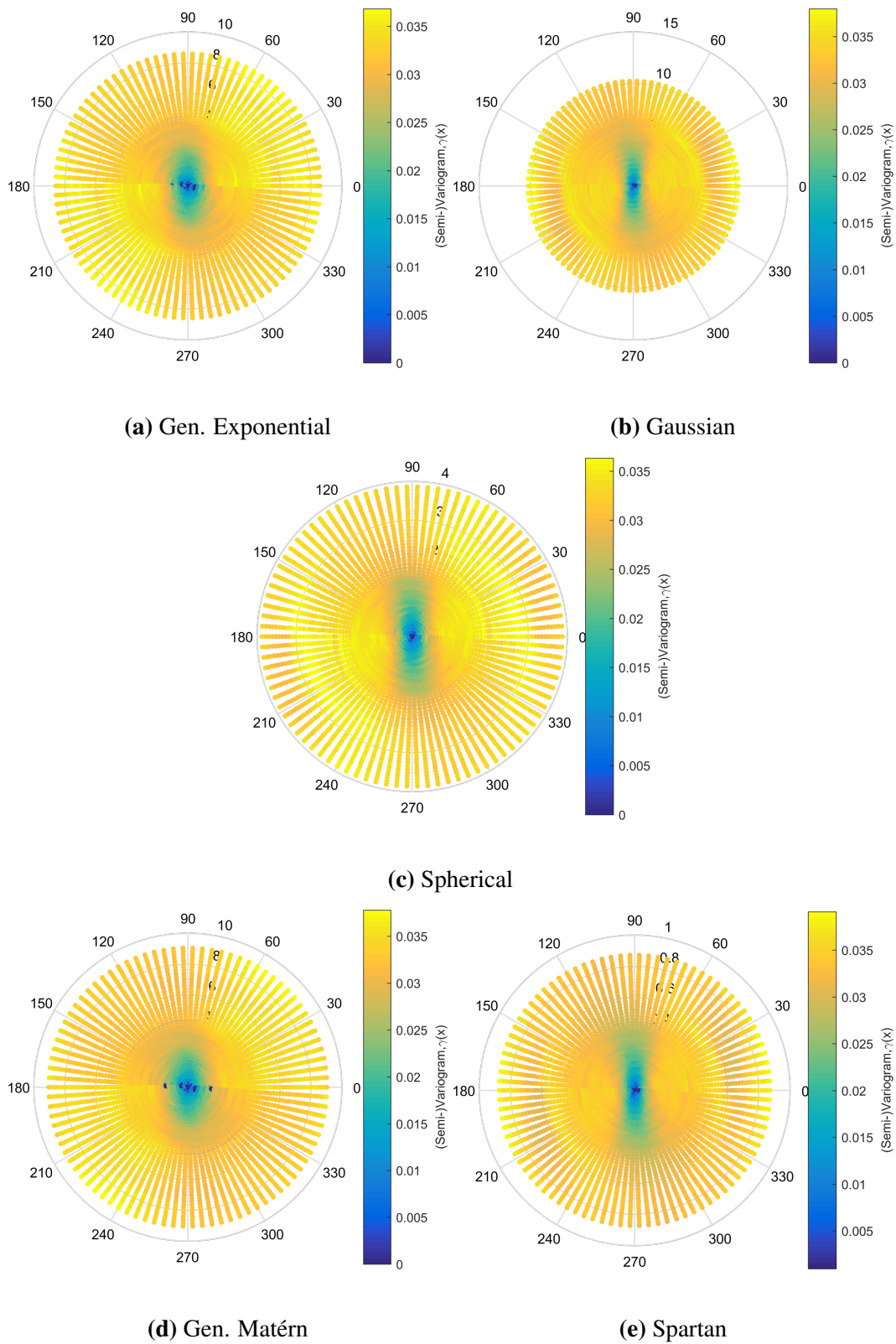


Figure 6.82 Experimental directional variograms of the new coordinations systems.

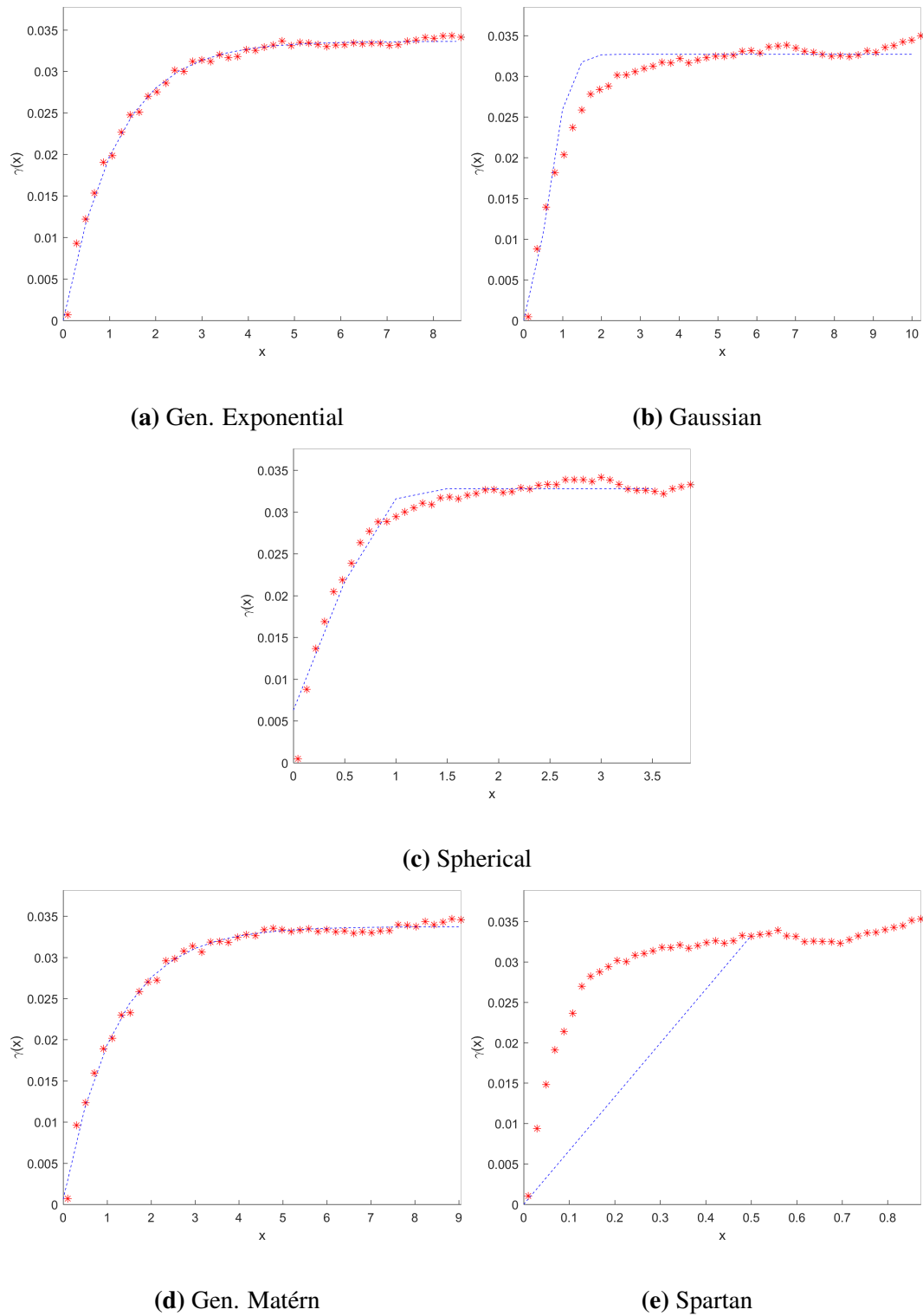


Figure 6.83 Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.

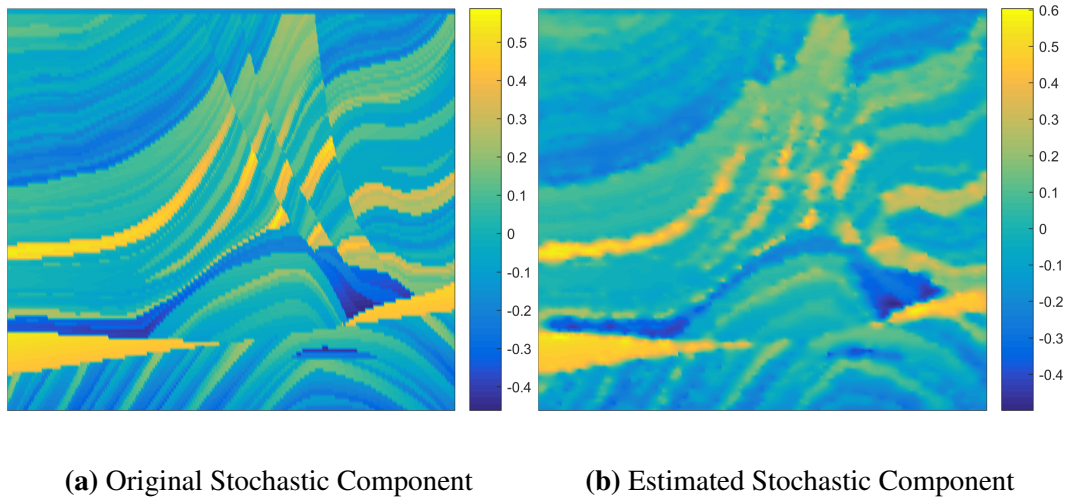


Figure 6.84 Original and Estimation of the stochastic component of the field. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 0.033$, $\xi = 1.780$, $c_0 = 0.001$, $\nu = 0.509$.

Table 6.66 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Matérn	0.049	0.698	0.006	0.080	0.904	0.891	0.805
Gen. Exponential	0.054	0.763	0.008	0.087	0.885	0.876	0.554
Spherical	0.061	0.727	0.008	0.090	0.877	0.865	0.468

The estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) resulting from OK is as shown in Fig. 6.84. The scatter diagram and histograms of the original and the estimated values of the stochastic component are, also, shown in Fig. 6.85, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.66. The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.86, and the corresponding scatter diagram and histograms are shown in Fig. 6.87. The estimations follow the original values achieving a very good proximity of the total distribution. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.67, show that their performance is similar with the first model to be slightly better. Also, the spatial distribution of the ordinary kriging estimations uncertainty is shown in Fig. 6.88. The uncertainty (as expected from theory) increases gradually from $\simeq 0.00$ near the known locations to 0.30 at great distances from them.

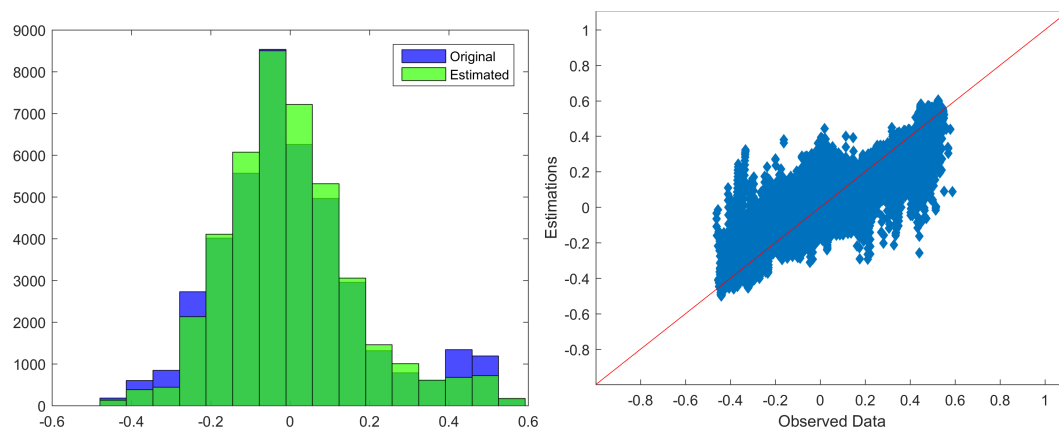


Figure 6.85 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 0.033$, $\xi = 1.780$, $c_0 = 0.001$, $\nu = 0.509$.

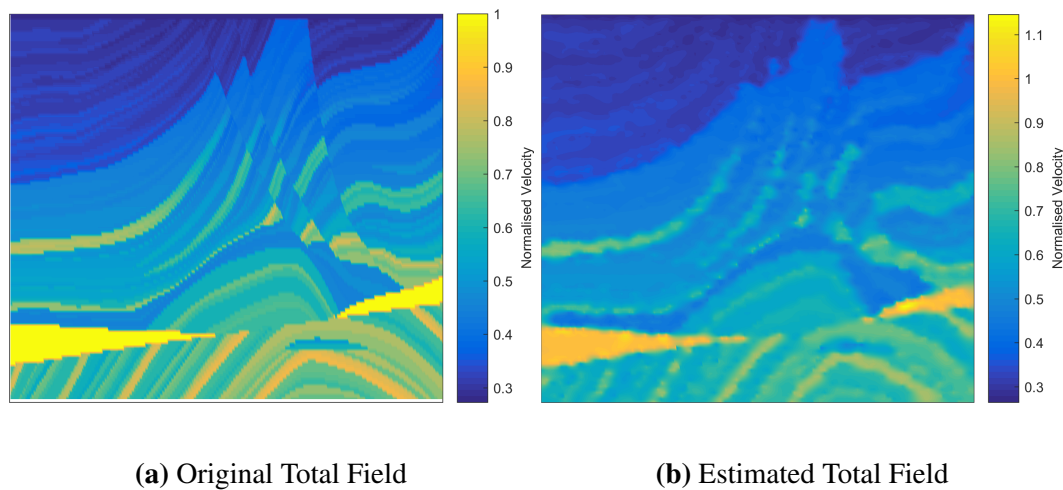


Figure 6.86 Original and estimated total field

Table 6.67 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Matérn	1.334	1.888	1.842	1.357	0.955	0.977	0.932
Gen. Exponential	1.334	1.909	1.842	1.357	0.952	0.974	0.926
Spherical	1.333	1.899	1.842	1.357	0.953	0.971	0.926

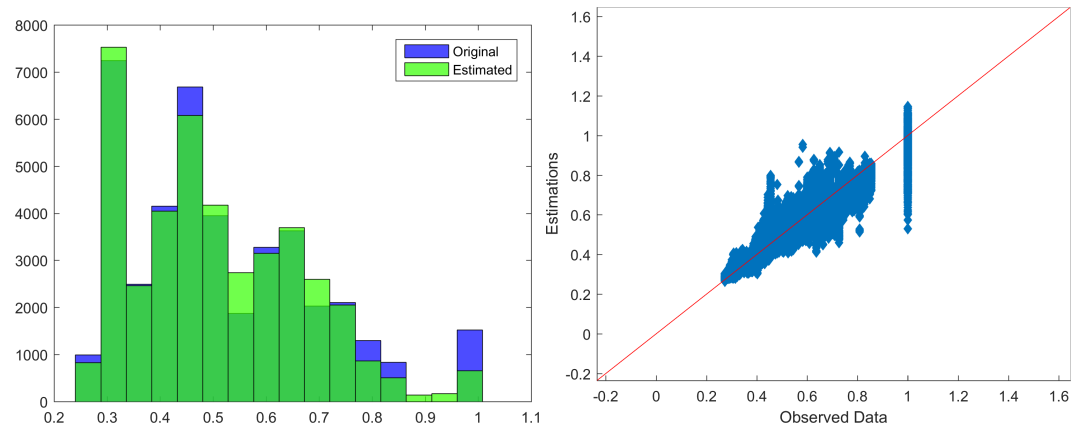


Figure 6.87 Histograms and Scatter plot of original and estimated values (total field)

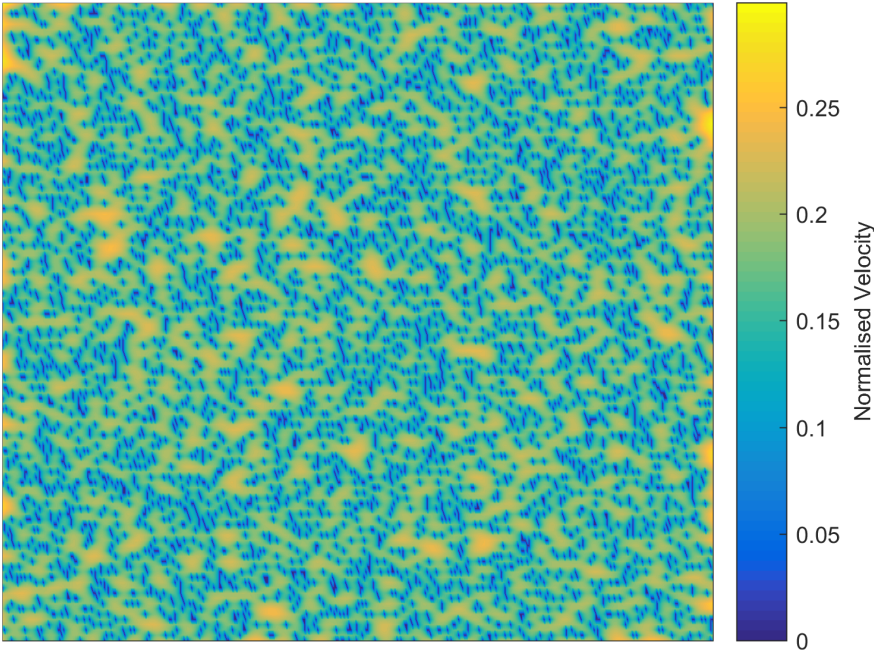


Figure 6.88 Uncertainty of the total field's estimation

Table 6.68 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Gen. Matérn	0.921	0.923	0.127	0.969	0.972	0.347
Gen. Exponential	0.914	0.919	0.135	0.961	0.969	0.370
Spherical	0.904	0.911	0.153	0.958	0.966	0.420

Finally, in Table 6.68 are presented the classification measures for the cases of 4 and 16 classes. The results show that for all models the measures increase as the number of classes increase, while the first model has the best performance for any number of classes.

CHI1

The estimated anisotropy parameters are given in Table 6.69. The parameters captured by the CHI estimator indicates that the major axis of the anisotropy ellipsis is almost horizontal and is also significantly greater ($\simeq 5$ times) than the minor axis.

By replacing the estimated anisotropy parameters to the anisotropic variogram models and minimizing the error function of the new models and the experimental directional variograms the rest parameters for each model are estimated, as presented in Table 6.70. The estimated values of the variance and the nugget effect are very close but the correlation lengths differ depending on the model. The Spherical model is the one that exhibits a relatively bigger correlation length from the other models, the correlation length of which are close.

The Leave-One-Out Cross Validation (LOOCV) gives as best model the Generalized Exponential, followed by Generalized Matérn and Spartan.

The fitting of the best model to the experimental variogram along the directions of $0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, and 150° is interpreted in Fig. 6.89 (the rest directional variograms can be found in Appendix A). As it can be seen the best theoretical variogram model diverges significantly from the experimental directional variograms. This can be attributed to minimization miscalculations, i.e. inappropriate objective function (very smooth or possible

Table 6.69 Anisotropy parameters of the investigated variogram models estimated with CHI method

R	ϕ
0.280	-89.1°

Table 6.70 Optimum Parameters of the variogram models with the lower degrees of freedom

Model	σ_z^2	ξ	c_0	ν
Gen.				
Exponential	0.038	0.021	0.000	0.115
Gaussian	0.026	3.533	0.008	—
Spherical	0.028	8.532	0.006	—
Gen. Matérn	0.028	1.057	0.006	3.500
	η_0	ξ	c_0	η_1
Spartan	0.857	2.618	0.000	2.000

Table 6.71 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen.							
Exponential	0.040	0.391	0.004	0.062	0.937	0.955	0.895
Gaussian	0.047	0.384	0.005	0.073	0.911	0.940	0.448
Spherical	0.047	0.383	0.005	0.073	0.911	0.940	0.449
Gen. Matérn	0.043	0.438	0.005	0.069	0.919	0.947	0.546
Spartan	0.046	0.390	0.005	0.072	0.912	0.941	0.464

Table 6.72 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.083	0.562	0.013	0.114	0.804	0.788	0.634
Gen. Mátern	0.086	0.788	0.016	0.127	0.736	0.734	0.355
Spartan	0.092	0.784	0.017	0.132	0.710	0.708	0.280

Table 6.73 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	1.330	1.888	1.839	1.356	0.945	0.954	0.902
Gen. Mátern	1.331	1.939	1.843	1.357	0.928	0.946	0.874
Spartan	1.331	1.933	1.842	1.357	0.925	0.941	0.867

local minima), inappropriate initial values or to the fields complexity. Thus, the analysis is continued without taking any further action.

Implementing OK with the determined best model, the resulting estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) is as shown in Fig. 6.90. The scatter diagram and histograms of the original and the estimated values of the stochastic component are illustrated in Fig. 6.91, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.72.

The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.92, and the correspondding scatter diagram and histograms are shown in Fig. 6.93. In general, the estimations follow the original values without achieving satisfying proximity of the total distribution. Both the tails of the distribution as the middle of it exhibit significant discrepancies. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.73, show that the first model has significantly better performance than the other two which have almost the same scores. Finally, Fig. 6.94 shows the spatial distribution of the ordinary kriging estimations uncertainty. The uncertainty is zero at the known points, while at the missing points it is equal to 0.37 with slightly smaller values at the edges.

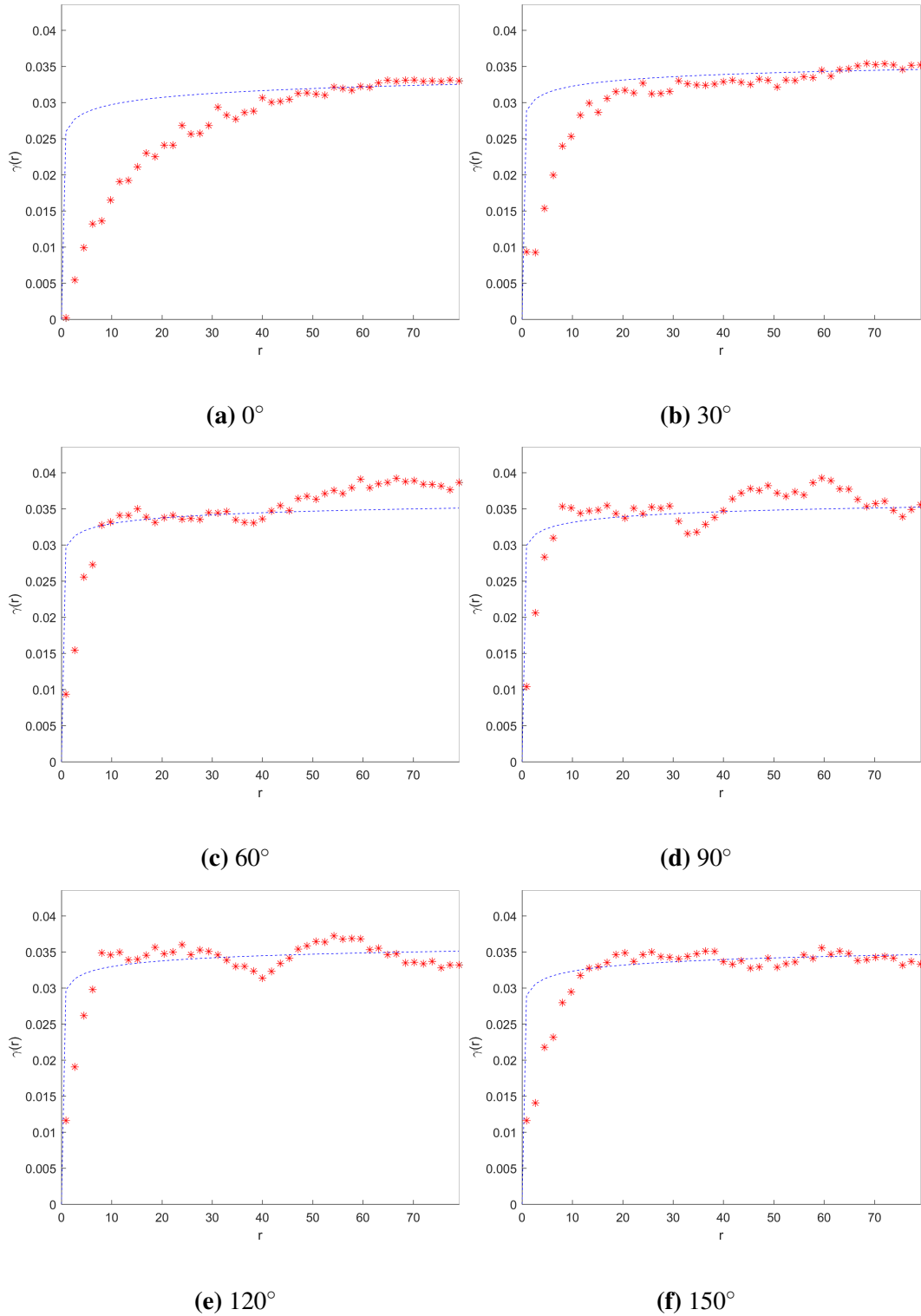


Figure 6.89 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.038$, $\xi_1 = 0.021.338$, $R = 0.208$, $\phi = -89.1^\circ$, $c_0 = 0.000$, $\nu = 0.115$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

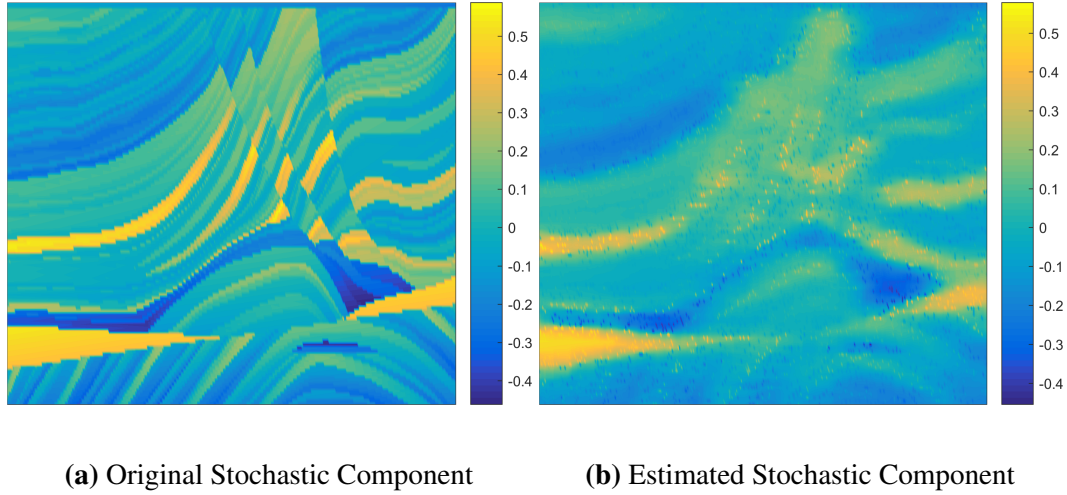


Figure 6.90 Original and Estimation of the stochastic component of the field. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.038$, $\xi_1 = 0.021.338$, $R = 0.208$, $\phi = -89.1^\circ$, $c_0 = 0.000$, $\nu = 0.115$.

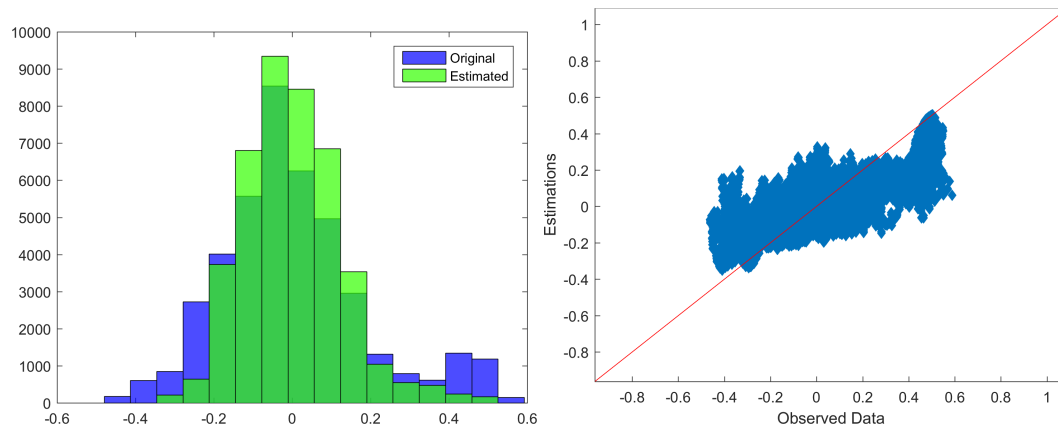


Figure 6.91 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Exponential with parameters $\sigma_z^2 = 0.038$, $\xi_1 = 0.021.338$, $R = 0.208$, $\phi = -89.1^\circ$, $c_0 = 0.000$, $\nu = 0.115$.

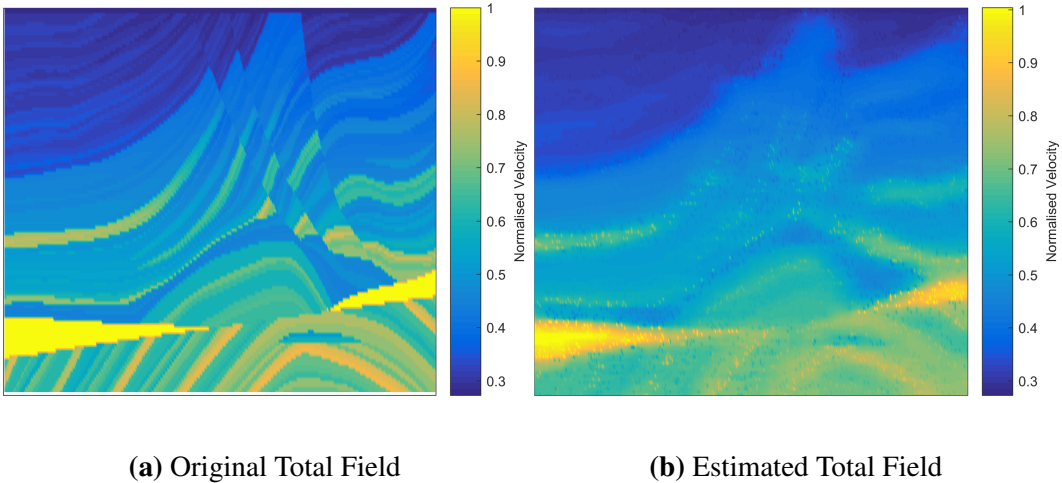


Figure 6.92 Original and estimated total field

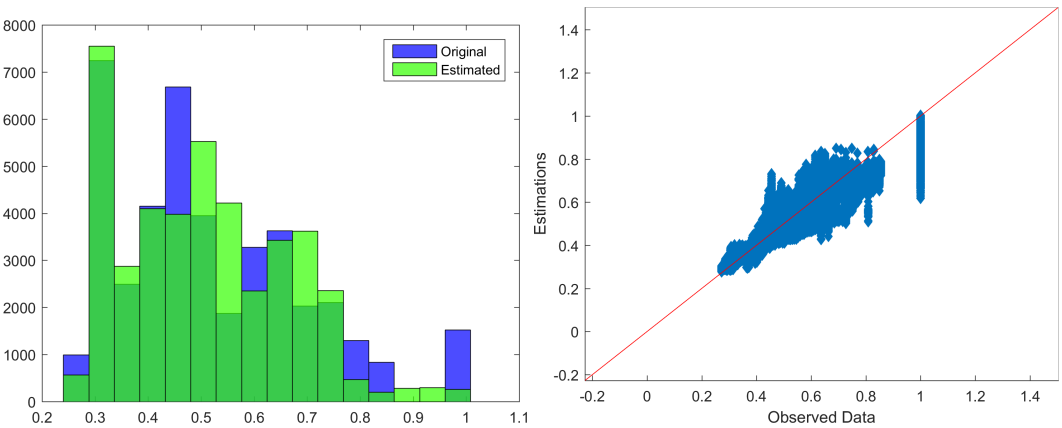


Figure 6.93 Histograms and Scatter plot of original and estimated values (total field)

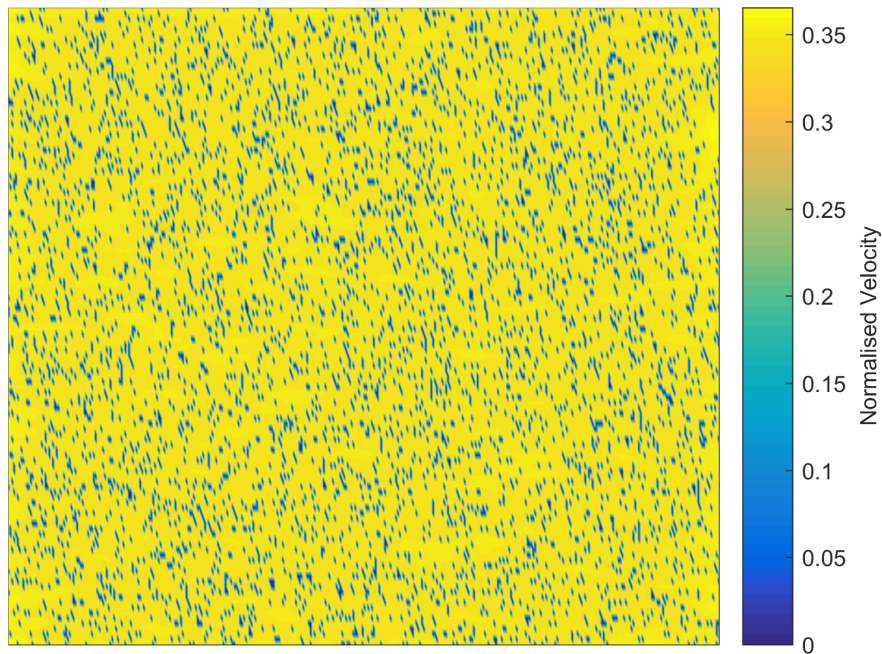


Figure 6.94 Uncertainty of the total field's estimation

Table 6.74 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Gen. Exponential	0.876	0.886	0.197	0.936	0.948	0.538
Gen. Mátern	0.862	0.880	0.201	0.916	0.941	0.514
Spartan	0.856	0.874	0.209	0.909	0.935	0.541

Also, in Table 6.74 are presented the classification measures for the cases of 4 and 16 classes.

CHI2

The new anisotropy parameters, estimated by means of DVF using the 5 variogram models, are shown in Table 6.75, while in Fig. 6.95 are interpreted the experimental directional variograms of the new coordinations system. All the models give increased anisotropy ratio relatively to the original one, but they still indicate strong anisotropic characteristics. Nevertheless, analysis procedure continues by assuming the new coordinations system as isotropic.

Table 6.75 Anisotropy parameters of the new coordinations systems

Model	Anisotropy Parameters		
	ξ_1	R	ϕ
Gen. Exponential	12.432	0.340	5.6°
Gaussian	13.228	0.386	3.8°
Spherical	32.732	0.381	4.9°
Gen. Matérn	16.851	0.290	20.7°
Spartan	37.306	0.404	41.6°

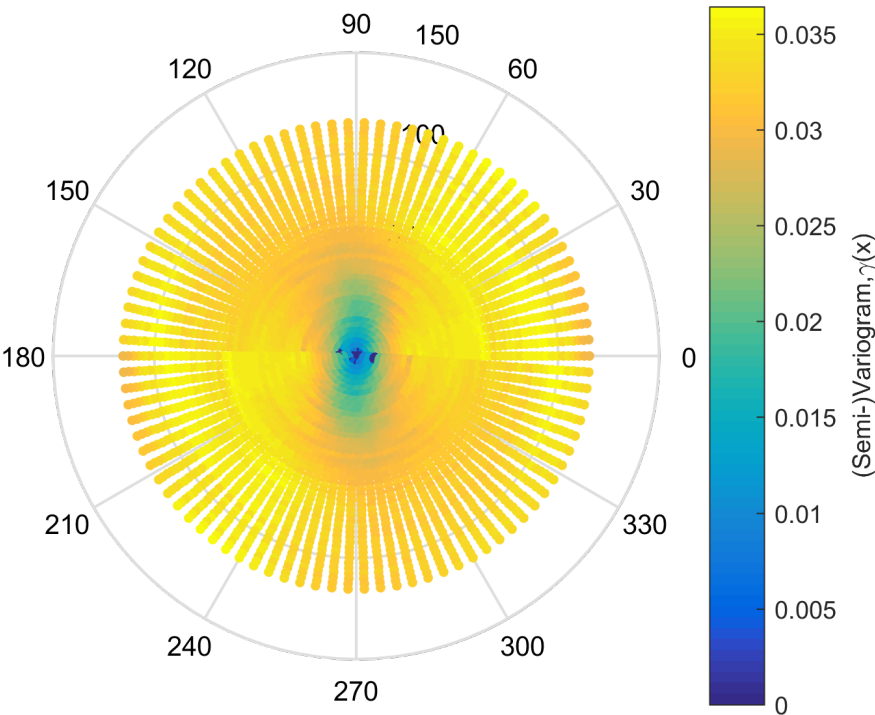


Figure 6.95 Experimental directional variograms of the new coordinations system

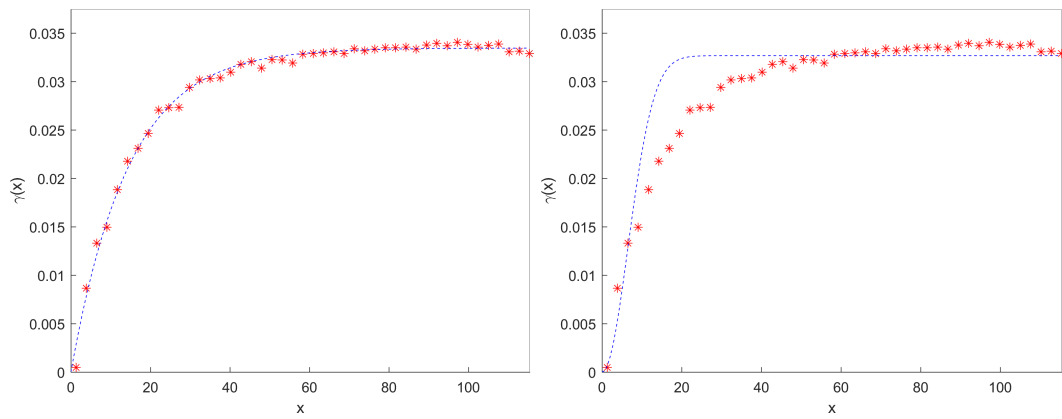
Table 6.76 Optimum Parameters of the isotropic variogram models

Model	σ_z^2	ξ	c_0	ν
Gen. Exponential	0.033	14.426	0.000	1.006
Gaussian	0.033	9.201	0.000	—
Spherical	0.026	41.402	0.007	—
Gen. Matérn	0.033	13.192	0.000	0.554
<hr/>				
	η_0	ξ	c_0	η_1
Spartan	0.945	15.553	0.000	3.005

By minimizing the error function of the isotropic variogram functions of the models and the corresponding experimental omnidirectional (isotropic) variogram of the new co-ordinations system (see Fig. 6.96) the parameters of the isotropic models are estimated, as presented in Table 6.76. All the models fit well to the experimental omnidirectional variogram. Also, the estimated parameters are very close, except from the correlation lengths which differ significantly depending on the model; the Spherical model is attributed by the large value of about 42, while the rest models' correlation length range from 9 to 15.

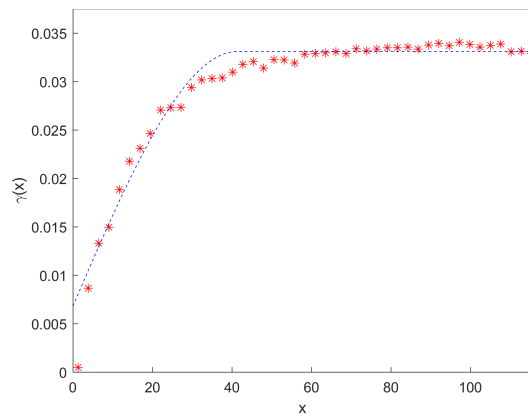
The Leave-One-Out Cross Validation (LOOCV) for the estimated models gives the validation measures presented in Table 6.77. As best model derives the Generalized Matérn, followed by Generalized Exponential and Spartan. As it can be seen from Fig. 6.96), the above mentioned models are those that fit better to the experimental omnidirectional variogram.

The estimation of the stochastic component of the field (i.e the transformed and detrended normalised velocities) resulting from OK is as shown in Fig. 6.97. The scatter diagram and histograms of the original and the estimated values of the stochastic component are, also, shown in Fig. 6.98, and the measures of the stochastic component estimation performance for the three best models are presented in Table 6.78. The resulting estimation of the total field, after the trend addition and Box-Cox transformation inversion, is as shows Fig. 6.99, and the corresponding scatter diagram and histograms are shown in Fig. 6.100. In general, the estimations follow the original values achieving a very good proximity of the total distribution. The measures of the total field (i.e. normalised velocities) estimation performance for the three best models, presented in Table 6.79, show that their performance is identical. Also, the spatial distribution of the ordinary kriging estimations uncertainty is

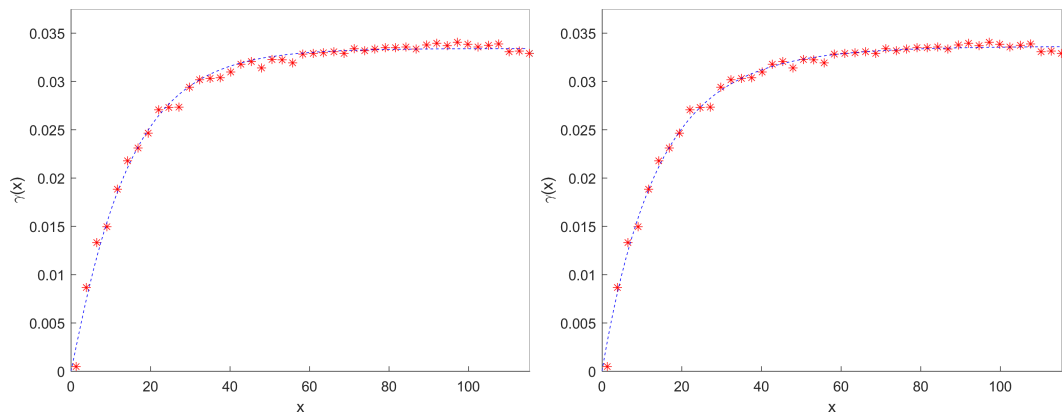


(a) Gen. Exponential

(b) Gaussian



(c) Spherical



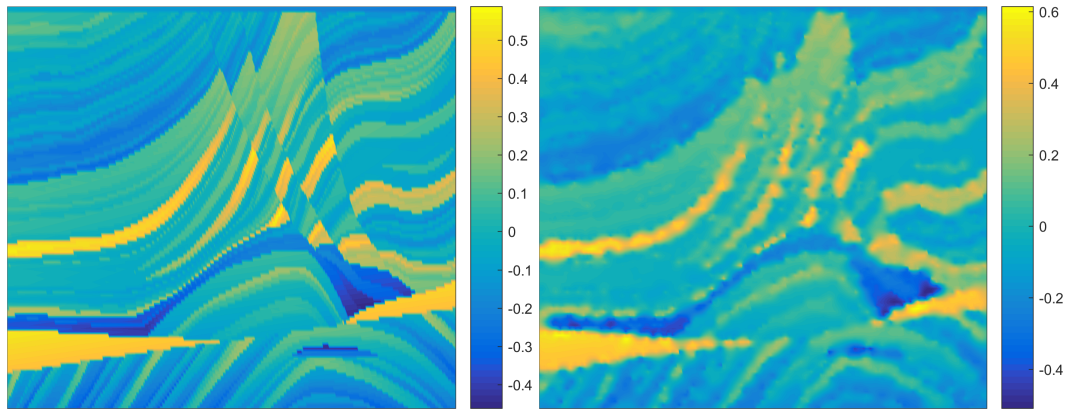
(d) Gen. Matérn

(e) Spartan

Figure 6.96 Fitting of the theoretical models to the corresponding experimental omnidirectional variograms of the new coordinations systems.

Table 6.77 Leave-One-Out Cross Validation Scores (see section 5.3.2)

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Exponential	0.023	0.398	0.002	0.044	0.967	0.976	0.932
Gaussian	12.844	28071.759	254337.941	504.319	0.002	0.813	0.000
Spherical	0.029	0.365	0.002	0.049	0.960	0.972	0.602
Gen. Matérn	0.023	0.403	0.002	0.044	0.967	0.976	0.945
Spartan	0.023	0.397	0.002	0.045	0.967	0.976	0.927

**(a)** Original Stochastic Component**(b)** Estimated Stochastic Component**Figure 6.97** Original and Estimation of the stochastic component of the field. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 0.033$, $\xi = 13.192$, $c_0 = 0.000$, $\nu = 0.554$.

shown in Fig. 6.88. The uncertainty (as expected from theory) increases gradually from $\simeq 0.00$ near the known locations to 0.30 at great distance from them.

Finally, in Table 6.80 are presented the classification measures for the cases of 4 and 16 classes. The results show that for all models have almost identical performance, while the measures increase as the number of classes increase.

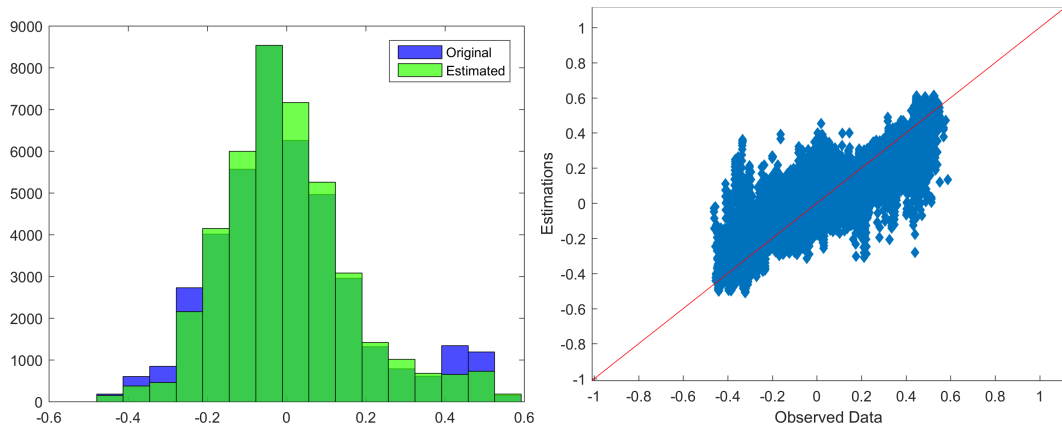


Figure 6.98 Histograms and Scatter plot of original and estimated values of the field's stochastic component. The model used is a Gen. Matérn with parameters $\sigma_z^2 = 0.033$, $\xi = 13.192$, $c_0 = 0.000$, $\nu = 0.554$.

Table 6.78 Ordinary Kriging Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Matérn	0.048	0.721	0.006	0.080	0.903	0.892	0.806
Gen. Exponential	0.049	0.715	0.006	0.080	0.903	0.892	0.801
Spartan	0.049	0.714	0.006	0.080	0.903	0.892	0.799

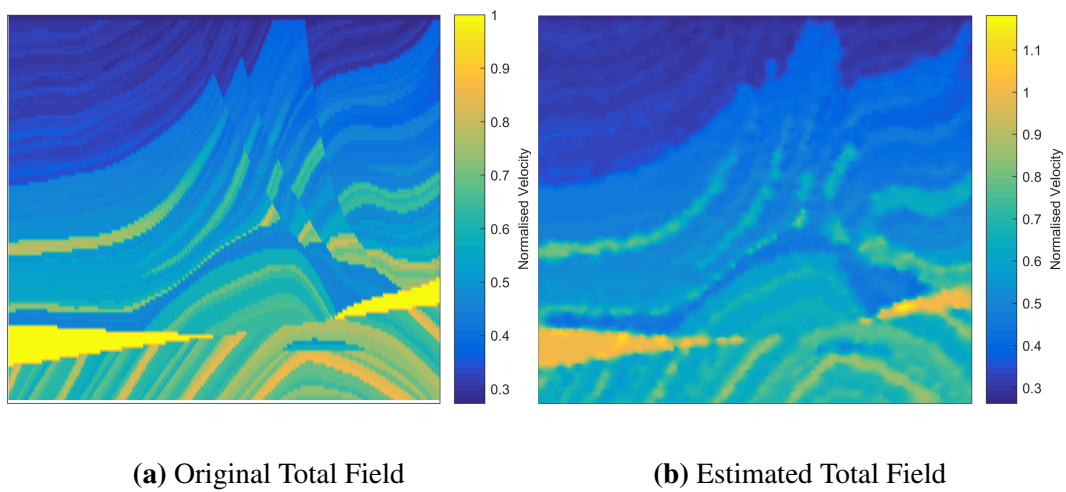


Figure 6.99 Original and estimated total field

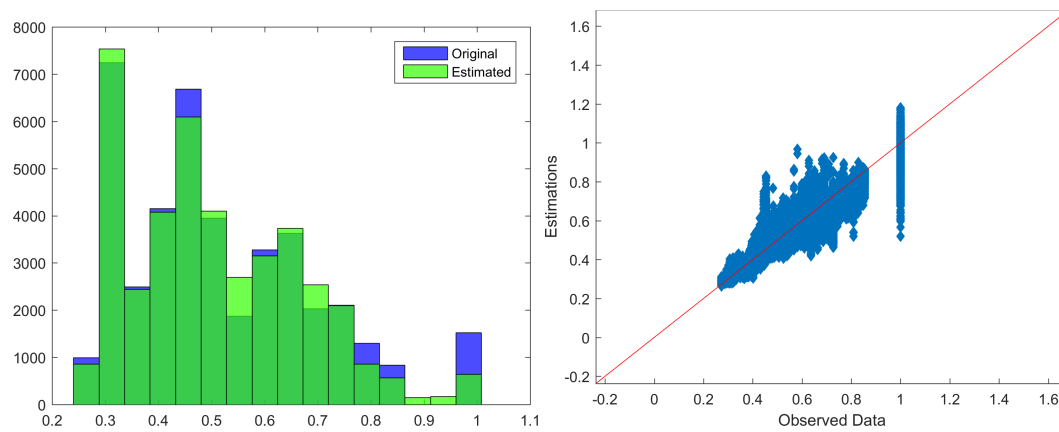


Figure 6.100 Histograms and Scatter plot of original and estimated values (total field)

Table 6.79 Total Field Estimation Scores

Model	MnAE	MxAE	MSE	RMSE	R_p	R_{sp}	r_F
Gen. Matérn	1.334	1.895	1.842	1.357	0.954	0.977	0.932
Gen. Exponential	1.334	1.895	1.842	1.357	0.955	0.977	0.933
Spartan	1.334	1.894	1.842	1.357	0.955	0.977	0.933

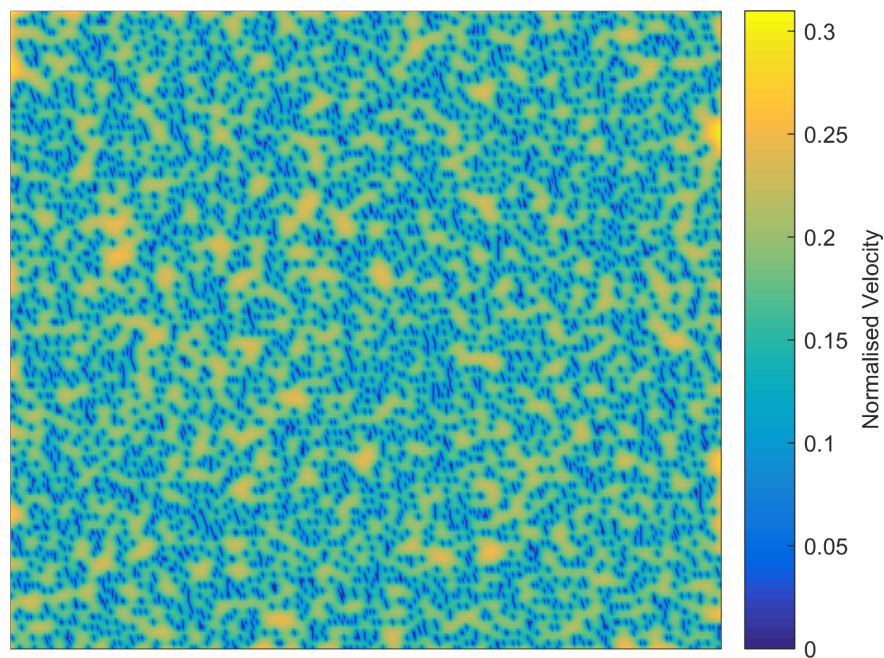


Figure 6.101 Uncertainty of the total field's estimation

Table 6.80 Classification Measures

Model	4 classes			16 classes		
	R_p	R_{sp}	MCR	R_p	R_{sp}	MCR
Gen. Matérn	0.923	0.925	0.124	0.968	0.973	0.342
Gen. Exponential	0.923	0.925	0.125	0.968	0.973	0.345
Spartan	0.922	0.925	0.125	0.968	0.972	0.346

6.5.2 DGC Simulation

The DGC simulation method is applied to the original dataset (i.e normalised velocities) in one step. The simulation is calculated using 4, 16 and 100 classes. The final results are interpreted in Figs. 6.102, 6.103, and 6.104 along with the histograms of the original and the estimated data and the distribution of the uncertainty of the estimations. The neighborhood used for the DGC simulation procedure is the same as the one used in OK, i.e 22x4. Finally, some measures of the DGC simulation performance are summarized in Table 6.81. The classification results show that the increasing of the number of classes from 4 to 16 leads to lower errors and higher correlation coefficient. This can be explained by the fact that the field is described better by a more complex model. The misclassification rate, on the contrary, increases with the increasing of the N_c due to the thinning of the bins which affects negatively the accuracy of the estimations. On the other hand, the results obtained by increasing the number of classes to 100 show that the interpolation performance is relatively close to the classification with $N_c = 16$, with slightly higher errors and smaller correlation but significantly higher MCR than the later.

Table 6.81 DGC Validation Measures

Classes	MnAE	RMSE	R_p	MCR
4	0.070	0.080	0.860	0.178
16	0.042	0.077	0.901	0.364
100	0.048	0.085	0.880	0.912

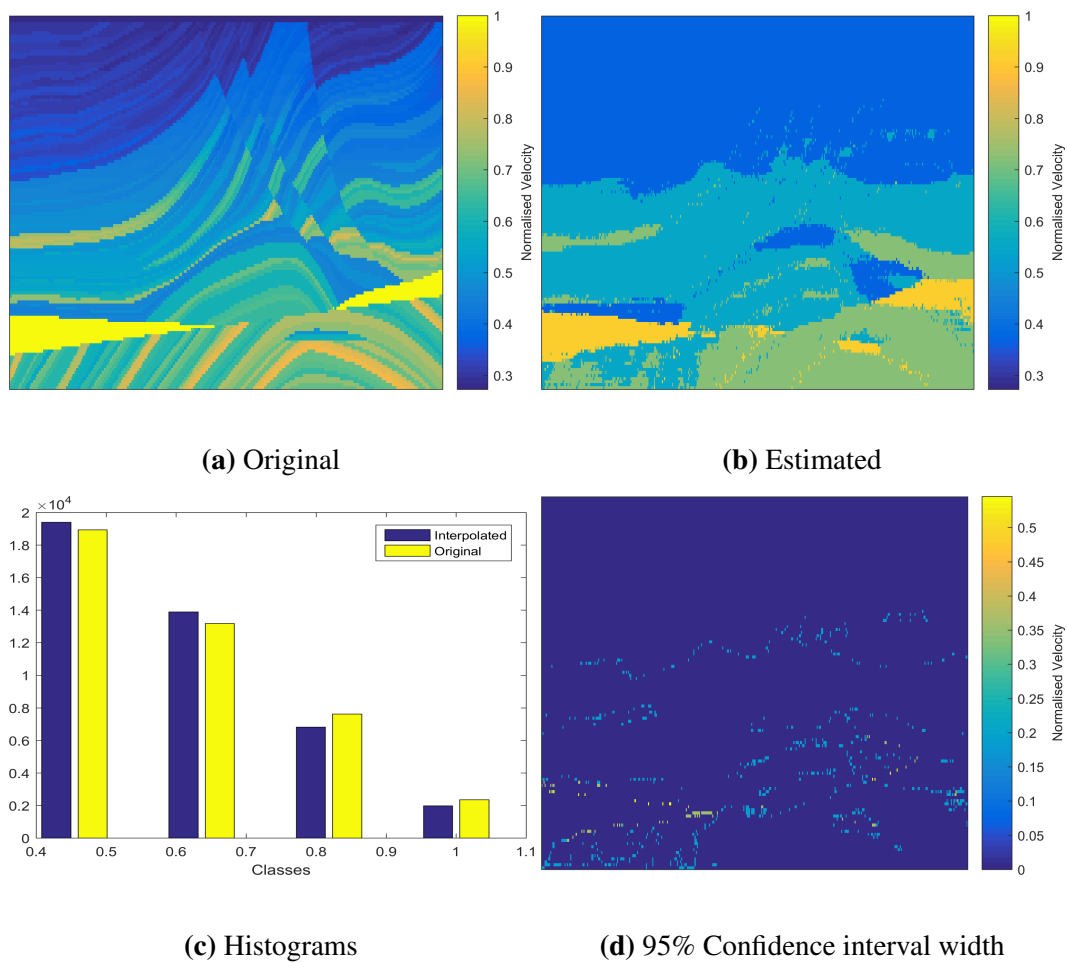


Figure 6.102 Results of DGC simulation with 4 classes

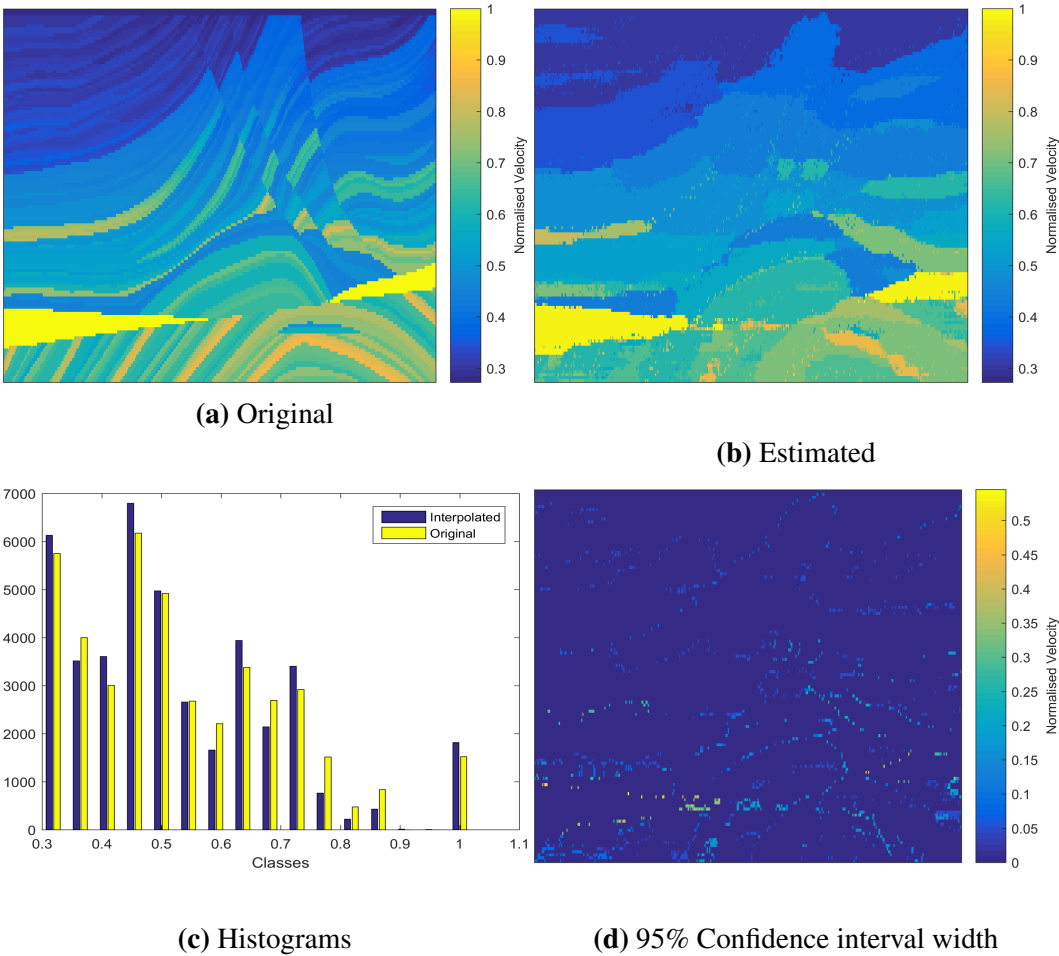


Figure 6.103 Results of DGC simulation with 16 classes

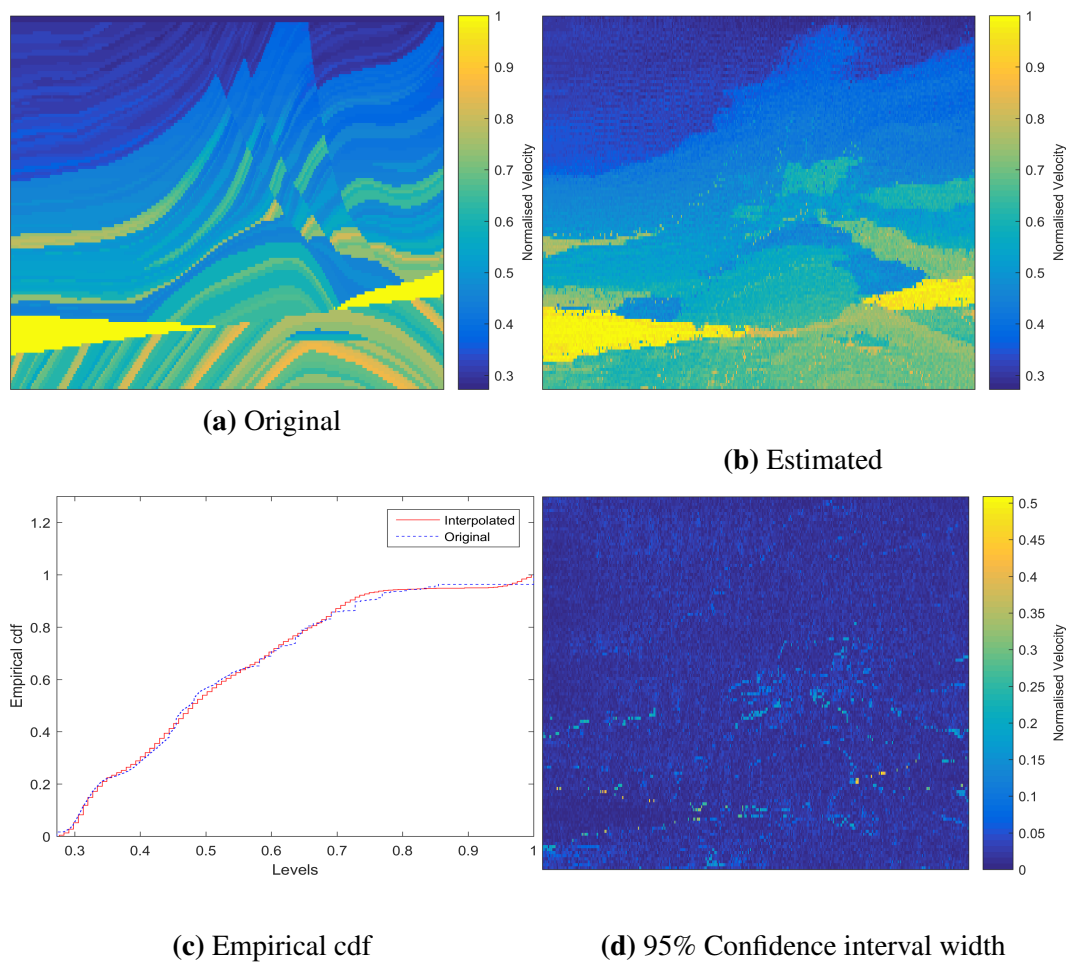


Figure 6.104 Results of DGC simulation with 100 classes

6.5.3 Synopsis

The results of the methods applied to the analysis of the investigated field's random sample are summarized in Table 6.82. These results show that the best methods for the reconstruction of the Marmousi model are the variations *DirVar2* and *CHI2*, which exhibit equal performance. As regards the investigated models, no one shows significantly better performance. However, the Generalized Exponential model is always amongst the three best models. The DGC simulation method has shown comparative performance rates to those of OK estimation, but slightly lower from the later. Notable is also the fact that in all methods the increase of the number of classes results to the improvement of the estimations as indicated by the increase of the correlation coefficients. However, in the DGC method the increase of the classes leads also to the worsening of the misclassification rate, as the complexity of the procedure increases.

Table 6.82 Summarized results for the random sample

Method	Model	Interpolation			4 Classes		16 Classes	
		MnAE	RMSE	R_p	R_p	MCR	R_p	MCR
DirVar0	Gen. Expon.	1.331	1.357	0.927	0.857	0.208	0.912	0.537
	Spartan	1.329	1.357	0.931	0.855	0.219	0.911	0.579
	Spherical	1.329	1.357	0.930	0.855	0.219	0.911	0.580
DirVar1	Spartan	1.331	1.359	0.904	0.838	0.221	0.888	0.537
	Gen. Matérn	1.331	1.357	0.923	0.854	0.211	0.907	0.543
	Gen. Expon.	1.331	1.357	0.923	0.853	0.211	0.907	0.543
DirVar2	Gen. Matérn	1.334	1.357	0.955	0.921	0.127	0.969	0.347
	Gen. Expon.	1.334	1.357	0.952	0.914	0.135	0.961	0.370
	Spherical	1.333	1.357	0.953	0.904	0.153	0.958	0.420
CHI1	Gen. Expon.	1.330	1.356	0.945	0.876	0.197	0.936	0.538
	Gen. Mátern	1.331	1.357	0.928	0.862	0.201	0.916	0.514
	Spartan	1.331	1.357	0.925	0.856	0.209	0.909	0.541
CHI2	Gen. Matérn	1.334	1.357	0.954	0.923	0.124	0.968	0.342
	Gen. Expon.	1.334	1.357	0.955	0.923	0.125	0.968	0.345
	Spartan	1.334	1.357	0.955	0.922	0.125	0.968	0.346
DGC	—	0.048	0.085	0.880	0.860	0.178	0.901	0.364

6.6 Problems Encountered

During the above work two major problems were encountered: a) the common entrapment of optimization algorithms in local minima, and b) the non positive-definiteness of some covariance matrices used in the application of OK estimations.

The first problem is crucial for the accuracy and the reliability of the inferred parameters and thus influences significantly the steps following. This problem is strongly related to user defined parameters, such as the maximum distance taken into account in the calculation of the experimental variograms and the number of bins of this distance, to the selection of objective function that is minimized, and to the algorithm employed for the minimization. This issue, for simplicity was addressed by assuming that the selected user defined parameters, objective function (Eq. (5.8)) and optimization algorithm give reliable results. This assumption is not necessarily adequate in all cases. As described in chapter 6, in a few cases the inferred variogram model does not fit well to the corresponding experimental variograms.

Regarding the non positive definiteness of the calculated covariance matrices (despite mathematical proof (see *Bochner's theorem* in section 2.5) that every permissible covariance function as positive definite) it can be attributed to the fast convergence of the majority of permissible covariance function to zero in long distances. This leads to sparse matrices (matrices with many zero elements) that do not have positive eigenvalues. The solution adopted to this problem was the restriction of the neighborhood size used for OK, which was arbitrarily chosen as described in the previous sections.

Chapter 7

Conclusions

7.1 Conclusions

The main goal of this thesis was to investigate whether geostatistical tools can be used for the simulation geophysical properties based on partial information (i.e., data sets with missing data), and more specifically for the simulation of a synthetic geological media stratigraphic data set. For this purpose two geostatistical methods were applied, Ordinary Kriging and Directional Gradient-Curvature simulation, to reconstruct the Marmousi dataset, a synthetic 2D acoustic model (workshop of 52nd EAEG meeting in 1990), using as data a random and a regular (drill-holes) sample of the entire model. In the case of OK, five methods of parameter inference were used. These methods differed regarding anisotropy estimation (i.e. parametric versus non parametric) and data transformation (transform data to isotropic or not). The results of these methods are summarized in Tables 7.1 and 7.2 for the regular and the random sampling, respectively.

As regards the adequacy of geostatistical tools in geological media simulation, this thesis showed that they can be very helpful if the available data can provide enough information about the investigated property. Then, the geostatistical tools can be used to supplement existing geophysical methods.

Performance measures for both types of sampling (see Tables 7.1 and 7.2) show that all the methods can give relatively good estimates of the missing values, as the achieved correlations between the reconstructed dataset and the real data are high. Moreover, the evaluation and comparison of the methods shows that the OK methods with the best performance are those including separate estimation of anisotropy and data transformation to isotropy, i.e *DirVar2* and *CHI2*. Finally, the DGC simulation method has shown comparable but slightly inferior performance measures to OK estimation. We believe that this is due to the local nature of the latter in contrast with kriging, which accounts for correlations with longer range.

Table 7.1 Summarized scores for the regular sample

Method	Model	Interpolation			4 Classes		16 Classes	
		MnAE	RMSE	R_p	R_p	MCR	R_p	MCR
DirVar0	Spherical	1.317	1.343	0.935	0.860	0.220	0.916	0.584
DirVar1	Spartan	1.317	1.343	0.932	0.856	0.219	0.912	0.578
DirVar2	Gen. Expon.	1.321	1.343	0.960	0.929	0.116	0.972	0.317
CHI1	Spherical	1.317	1.343	0.929	0.848	0.228	0.904	0.591
CHI2	Spartan	1.321	1.343	0.959	0.930	0.114	0.974	0.295
DGC	—	0.045	0.079	0.894	0.858	0.193	0.915	0.465

Table 7.2 Summarized scores for the random sample

Method	Model	Interpolation			4 Classes		16 Classes	
		MnAE	RMSE	R_p	R_p	MCR	R_p	MCR
DirVar0	Gen. Expon.	1.331	1.357	0.927	0.857	0.208	0.912	0.537
DirVar1	Spartan	1.331	1.359	0.904	0.838	0.221	0.888	0.537
DirVar2	Gen. Matérn	1.334	1.357	0.955	0.921	0.127	0.969	0.347
CHI1	Gen. Expon.	1.330	1.356	0.945	0.876	0.197	0.936	0.538
CHI2	Gen. Matérn	1.334	1.357	0.954	0.923	0.124	0.968	0.342
DGC	—	0.048	0.085	0.880	0.860	0.178	0.901	0.364

Notable is also the fact that in all methods the increase of the number of classes improves the correlation coefficients, but leads to higher misclassification rates.

7.2 Future Studies

In future research, the improvement of the developed geostatistical codes should be the first goal. Improving the optimization step could be essential. A sensitivity analysis that explores the effect of the user defined parameters (i.e. maximum distance taken into account in the calculation of the experimental variograms, number of bins of this distance, objective function that is minimized, algorithm employed for the minimization) could provide useful information. Moreover, different parameter inference methods, such as method of moments

and maximum likelihood estimation, could be used. Enhancement of the currently used methods by calculating the derivatives of objective function can also be investigated.

Another point of interest could be the tackling of the non positive-definite matrices that appear in the ordinary kriging step. Also, a comparison of these codes with other existing algorithms could be crucial for the assessment of their performance.

Finally, the implementation of different estimation and simulation methods, such as indicator kriging or plurigaussian simulation method, should be explored. A challenging but promising alternative could also be the analysis of the data with non stationary statistics, in order to tackle the high complexity of the observed patterns.

Bibliography

- Adler, R. J. and Taylor, J. E. (2007). *Random Fields and Geometry*. Springer Monographs in Mathematics. Springer. [6](#)
- Agou, V. (2016). Geostatistical analysis of precipitation on the island of crete. Master's thesis, Technical University of Crete. [1](#)
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723. [40](#)
- Bargaoui, Z. K. and Chebbi, A. (2009). Comparison of two kriging interpolation methods applied to spatiotemporal rainfall. *Journal of Hydrology*, 365(1):56–73. [1](#)
- Barnett, V. (1981). *Interpreting Multivariate Data*. Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley. [3](#)
- Bochner, S., Tenenbaum, M., and Pollard, H. (1959). *Lectures on Fourier Integrals*. *Annals of Mathematics Studies*. Princeton University Press. [18](#)
- Bourgeois, A., Bourget, M., Lailly, P., Poulet, M., Ricarte, P., and Versteeg, R. (1990). Marmousi, model and data. In *EAEG Workshop-Practical Aspects of Seismic Data Inversion*. [1](#), [51](#)
- Bowman, K. O. and Shenton, L. R. (2004). Estimation: Method of moments. *Encyclopedia of statistical sciences*. [43](#)
- Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B*, 26(2):211–252. [38](#)
- Brognon, G. P. and Verrier, G. R. (1966). Oil and geology in cuanza basin of angola. *AAPG Bulletin*, 50(1):108–158. [51](#)
- Burrough, P., McDonnell, R., and Lloyd, C. D. (2013). *Principles of Geographical Information Systems:3rd Revised*. Oxford University Press, 1st edition. [3](#)
- Casella, G. and Berger, R. L. (2001). *Statistical Inference*. Duxbury Press, 2nd edition. [8](#)
- Chentsov, N. N. (1957). Lévy brownian motion for several parameters and generalized white noise. *Theory of Probability and Its Applications*, 2(2):265–266. [33](#)
- Chilès, J.-P. and Delfiner, P. (2012). *Geostatistics: Modeling Spatial Uncertainty*. Wiley Series in Probability and Statistics. Wiley, 2nd edition. [4](#), [5](#), [12](#), [13](#), [19](#), [28](#)

- Chorti, A. and Hristopulos, D. T. (2008). Nonparametric identification of anisotropic (elliptic) correlations in spatially distributed data sets. *IEEE Transactions on signal processing*, 56(10):4738–4751. 43, 44
- Christakos, G. (1992). *Random Field Models in Earth Sciences*. Academic Press, 2nd edition edition. 1
- Cressie, N. (1990). The origins of kriging. *Mathematical geology*, 22(3):239–252. 1, 27
- Deutsch, C. V. (2002). *Geostatistical Reservoir Modeling*. Oxford university press, 1st edition. 1
- Fisher, R. A. (1997). On an absolute criterion for fitting frequency curves. *Statistical Science*, 12(1):39–41. 42
- Galetakis, M. (1998). Evaluation of multiple seam lignite deposits of ptolemaes area, using statistical methods. *Mineral Wealth*, 107:55–64. 1
- Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American Statistical Association*, 70(350):320–328. 45
- Geisser, S. (1993). *Predictive Inference*, volume 55. CRC press. 45
- Goovaerts, P. (1997). *Geostatistics for Natural Resources Evaluation*. Oxford University Press. 1, 28, 30, 37
- Greene, H. J. (1992). *Evaluation of corrosion protection methods for aluminum metal matrix composites*. PhD thesis, University of Southern California. 1
- Guttorp, P. and Gneiting, T. (2006). Studies in the history of probability and statistics xlix on the matérn correlation family. *Biometrika*, 93(4):989–995. 19
- Hristopulos, D. T. (2002). New anisotropic covariance models and estimation of anisotropic parameters based on the covariance tensor identity. *Stochastic Environmental Research and Risk Assessment*, 16(1):43–62. 43
- Hristopulos, D. T. and Elogne, S. N. (2007). Analytic properties and covariance functions for a new class of generalized gibbs random fields. *IEEE Transactions on Information Theory*, 53(12):4667–4679. 19, 36
- Irons, T. (2008). Marmousi model. <http://reproducibility.org/RSF/book/data/marmousi/paper.pdf>, Accesed on 10 March 2017. 51
- Kelkar, M. and Perez, G. (2002). *Applied Geostatistics for Reservoir Characterization*. Richardson, Tex.: Society of Petroleum Engineers. 1
- Khoshnevisan, D. (2002). *Multiparameter Processes: An Introduction to Random Fields*. Springer Monographs in Mathematics. Springer. 6
- Kindermann, R. (1980). *Markov Random Fields And Their Applications*. Contemporary mathematics 1. American Mathematical Society. 6

- Kitanidis, P. K. (1997). *Introduction to Geostatistics: Applications in Hydrogeology*. Cambridge University Press. 8
- Kolmogorov, A. N. (1960). *Foundations of the Theory of Probability*. Chelsea Pub Co, 2nd edition. 6
- Krige, D. G. (1951). A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy*, 52(6):119–139. 1, 27
- Lantuéjoul, C. (2013). *Geostatistical Simulation: Models and Algorithms*. Springer Science & Business Media. 4
- Liu, H., Shi, J., and Erdem, E. (2010). Prediction of wind speed time series using modified taylor kriging method. *Energy*, 35(12):4870–4879. 1
- Mantoglou, A. and Wilson, J. L. (1982). The turning bands method for simulation of random fields using line generation by a spectral method. *Water Resources Research*, 18(5):1379–1394. 33
- Mariethoz, G. and Renard, P. (2010). Reconstruction of incomplete data sets or images using direct sampling. *Mathematical Geosciences*, 42(3):245–268. 1
- Matheron, G. (1971). *The Theory of Regionalized Variables and Its Applications*. Centre de Morphologie Mathématique Fontainebleau: Les cahiers du Centre de Morphologie Mathématique de Fontainebleau. École nationale supérieure des mines. 1
- Olea, R. A. (2006). A six-step practical approach to semivariogram modeling. *Stochastic Environmental Research and Risk Assessment*, 20(5):307–318. 42
- Oliver, M. A. and Webster, R. (2015). *Basic Steps in Geostatistics: The Variogram and Kriging*. SpringerBriefs in Agriculture. Springer. 28
- Papadimitriou, C. H. and Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall. 36
- Papadopoulou, M., Varouchakis, E., and Karatzas, G. (2009). Simulation of complex aquifer behavior using numerical and geostatistical methodologies. *Desalination*, 237(1-3):42–53. 1
- Papoulis, A. and Pillai, S. U. (2002). *Probability, Random Variables and Stochastic Processes*. McGraw-Hill Europe, 4th edition. 8
- Pardo-Igúzquiza, E. (1998). Maximum likelihood estimation of spatial covariance parameters. *Mathematical Geology*, 30(1):95–108. 42
- Pavlidis, A., Hristopulos, D. T., Roumpos, C., and Agioutantis, Z. (2015). Spatial modeling of lignite energy reserves for exploitation planning and quality control. *Energy*, 93(P2):1906–1917. 1
- Petrakis, M. P. (2012). Elliptical anisotropy statistics on spatial data and geostatistical applications. Master's thesis, Technical University of Crete. 43

- Petrakis, M. P. and Hristopulos, D. T. (2012). Elliptical anisotropy statistics of two-dimensional differentiable gaussian random fields: Joint probability density function and confidence regions. *arXiv preprint arXiv:1203.5010*. 43
- Rozanov, Y. A. (1982). *Markov Random Fields*. Springer-Verlag New York. 6
- Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464. 40
- Seni, G. and Elder, J. F. (2010). Ensemble methods in data mining: Improving accuracy through combining predictions. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1):1–126. 45
- Shumway, R. H. and Stoffer, D. S. (2011). *Time Series Analysis and Its Applications: With R Examples*. Springer Texts in Statistics. Springer-Verlag New York, 3rd edition. 12
- Spathopoulos, F. (1996). An insight on salt tectonics in the angola basin, south atlantic. *Geological Society, London, Special Publications*, 100(1):153–174. 51
- Swerling, P. (1962). Statistical properties of the contours of random surfaces. *IRE Transactions on Information Theory*, 8(4):315–321. 44
- Thijssen, J. (2016). *A Concise Introduction to Statistical Inference*. Chapman and Hall/CRC. 8
- Tobler, W. R. (1970). A computer movie simulating urban growth in the detroit region. *Economic geography*, 46(sup1):234–240. 3
- Varouchakis, E., Giannakis, G., Lilli, M., Ioannidou, E., Nikolaidis, N., and Karatzas, G. (2015). Development of a statistical tool for the estimation of riverbank erosion probability. *SOIL Discussions*, 2:647–674. 1
- Varouchakis, E. and Hristopulos, D. (2013). Comparison of stochastic and deterministic methods for mapping groundwater level spatial variability in sparsely monitored basins. *Environmental monitoring and assessment*, 185(1):1–19. 1
- Verrier, G. and Branco, F. C. (1972). Quenguela-north tertiary trough and oilfield. In *Revue de L Institut Francais du Petrole et Annales des Combustibles Liquides*, volume 27, page 51. xvii, 50
- Versteeg, R. and Grau, G. (1990). Practical aspects of seismic data inversion, the marmousi experience eaeg workshop. In *52nd EAEG Meeting, Eur. Assoc. Expl. Geophys., Proceedings of 1990 EAEG Workshop, 52nd EAEG Meeting*, pages 1–194. 1, 51
- Webster, R. and Oliver, M. A. (2007). *Geostatistics for Environmental Scientists*. Statistics in Practice. Wiley, 2nd edition. 33
- Yaglom, A. M. (1987). *Correlation Theory of Stationary and Related Random Functions: Volume I: Basic Results*. Springer, 1st edition. 13
- Žukovič, M. and Hristopulos, D. T. (2013a). A directional gradient-curvature method for gap filling of gridded environmental spatial data with potentially anisotropic correlations. *Atmospheric Environment*, 77:901–909. 1, 34, 36

- Žukovič, M. and Hristopulos, D. T. (2013b). Reconstruction of missing data in remote sensing images using conditional stochastic optimization with global geometric constraints. *Stochastic Environmental Research and Risk Assessment*, 27(4):785–806. [1](#), [34](#)

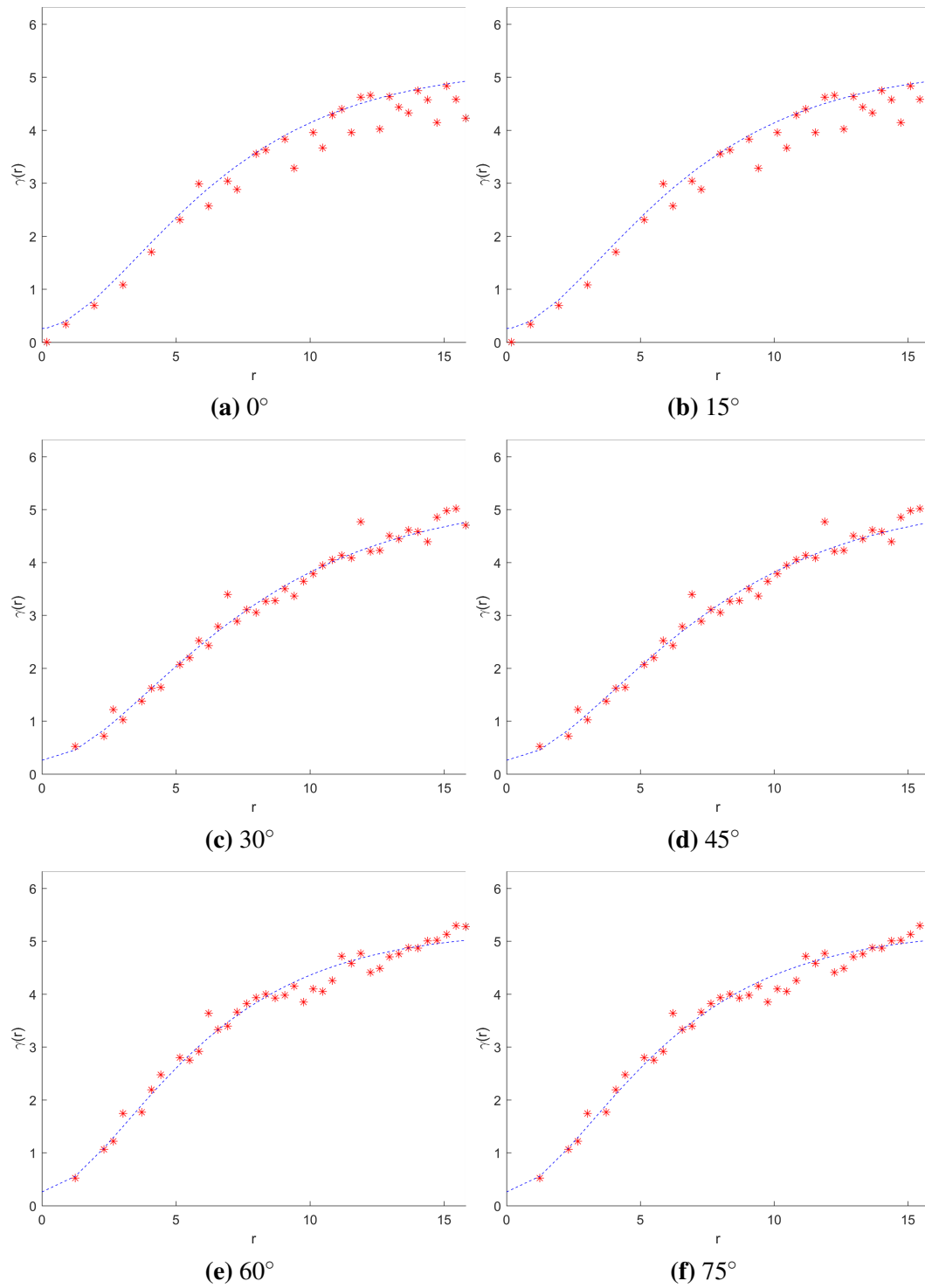
Appendix A

Figures

In this Appendix are presented the figures illustrating the fitting of the best estimated anisotropic model to the experimental directional variograms implemented to the *DirVar0*, *DirVar1* and *CH11* methods for the synthetic dataset and the regular and random sample of Marmousi model.

A.1 Synthetic Dataset

A.1.1 DirVar0



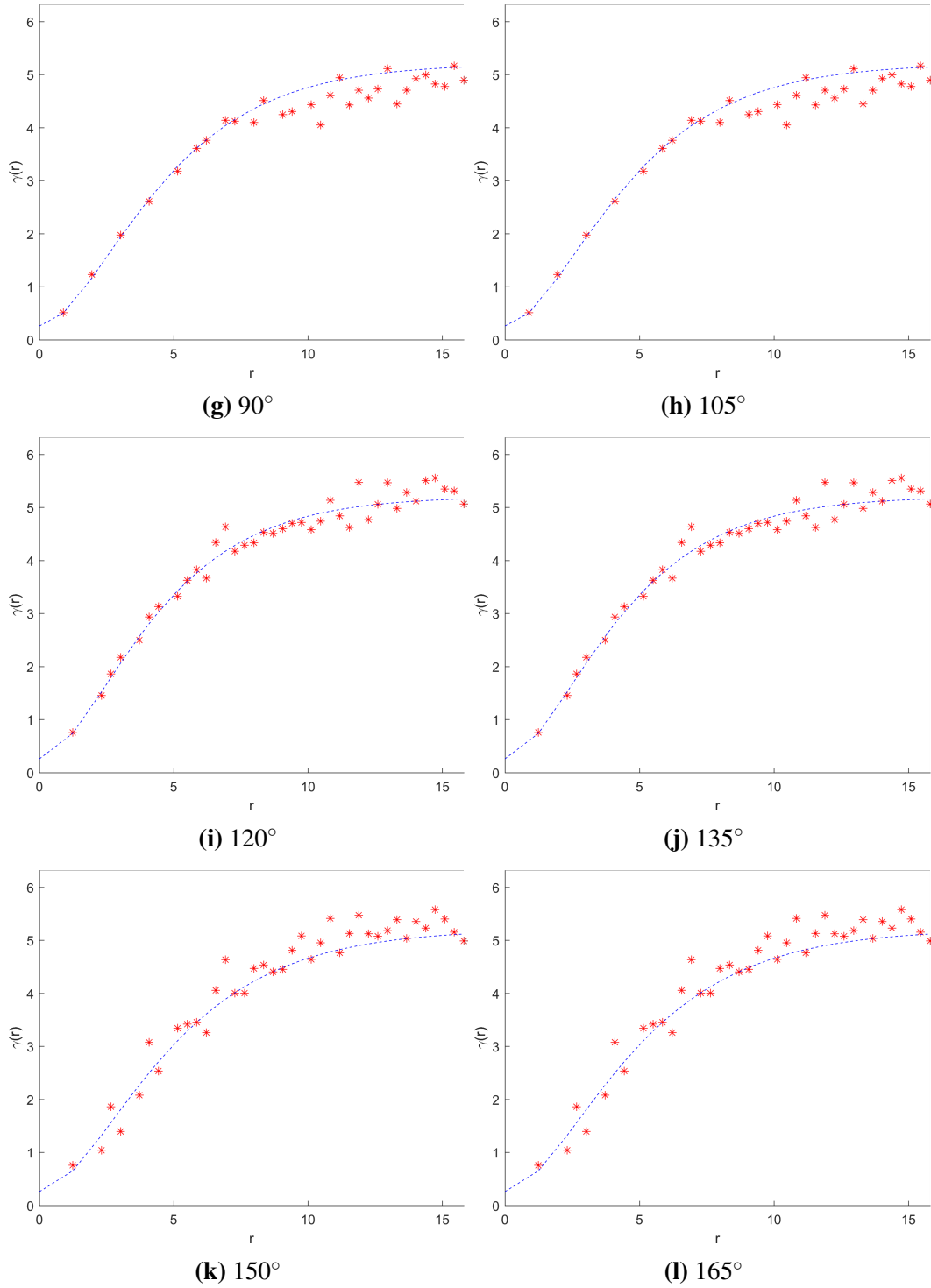
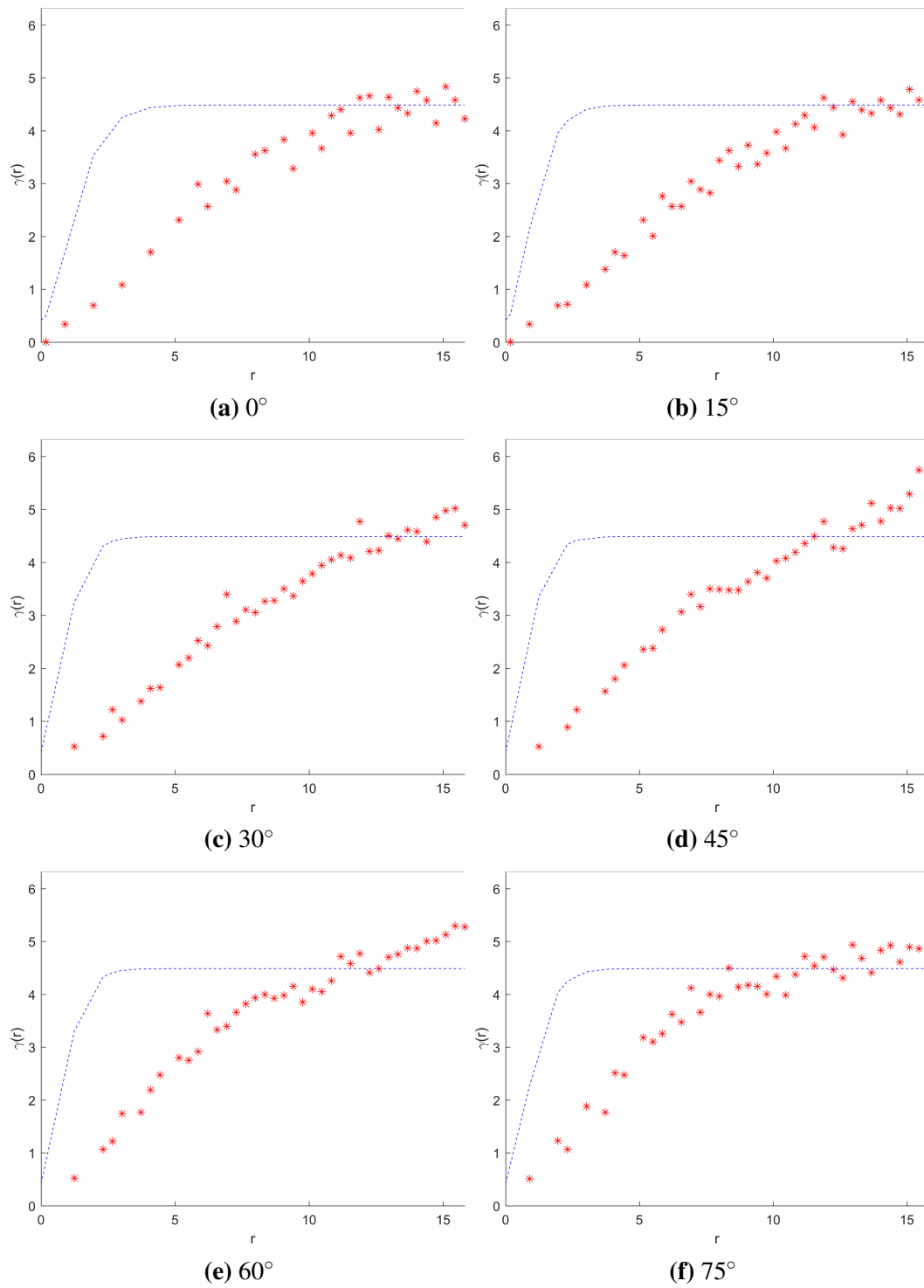


Figure A.1 Fitting of the best theoretical model to the directional experimental variograms of the field. The best model is a Gen. Mátern with parameters $\sigma_z^2 = 4.945$, $\xi_1 = 1.694$, $R = 0.429$, $\phi = -65.2^\circ$, $c_0 = 0.260$, $\nu = 1.525$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 16.

A.1.2 DirVar1



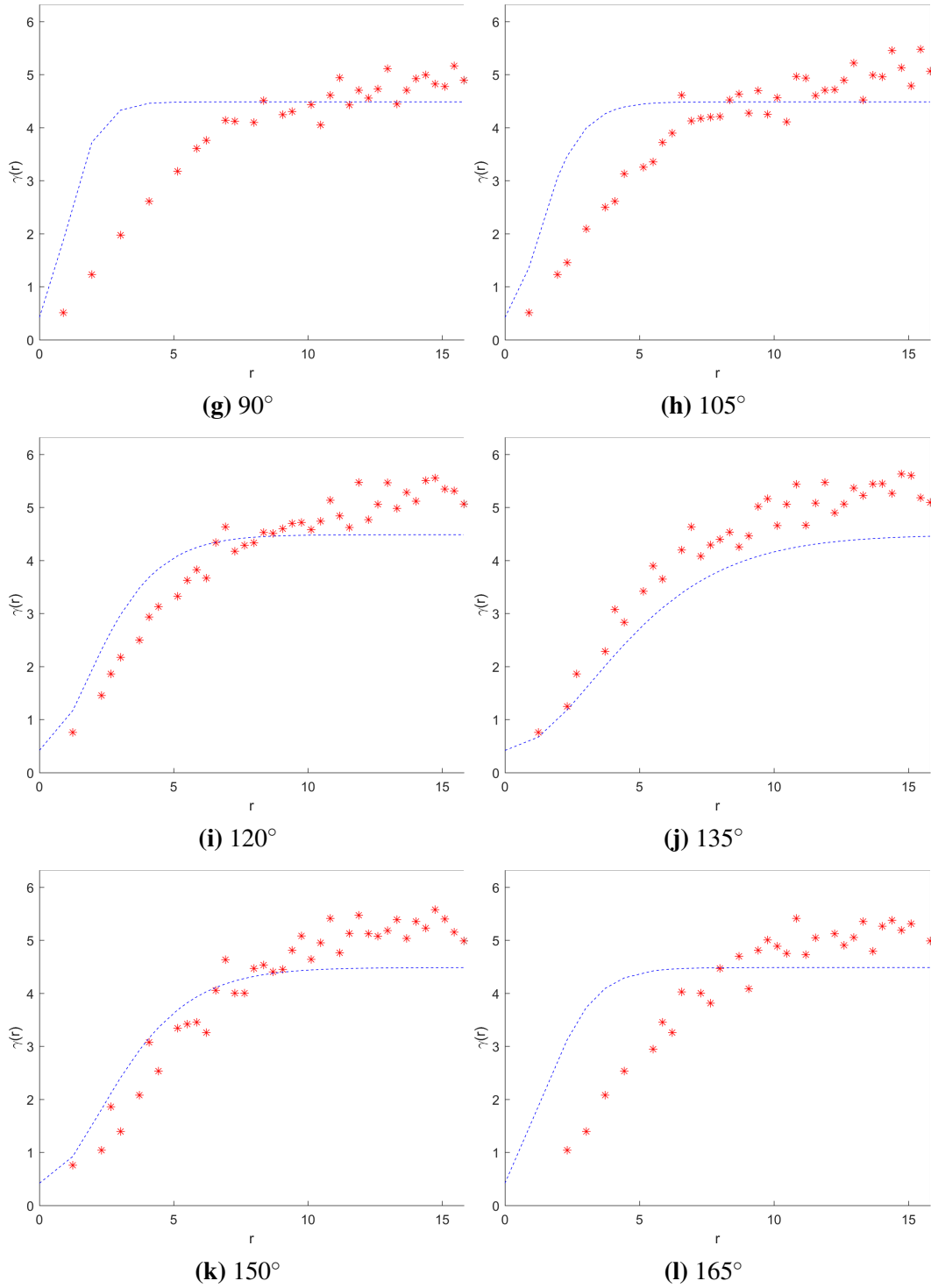
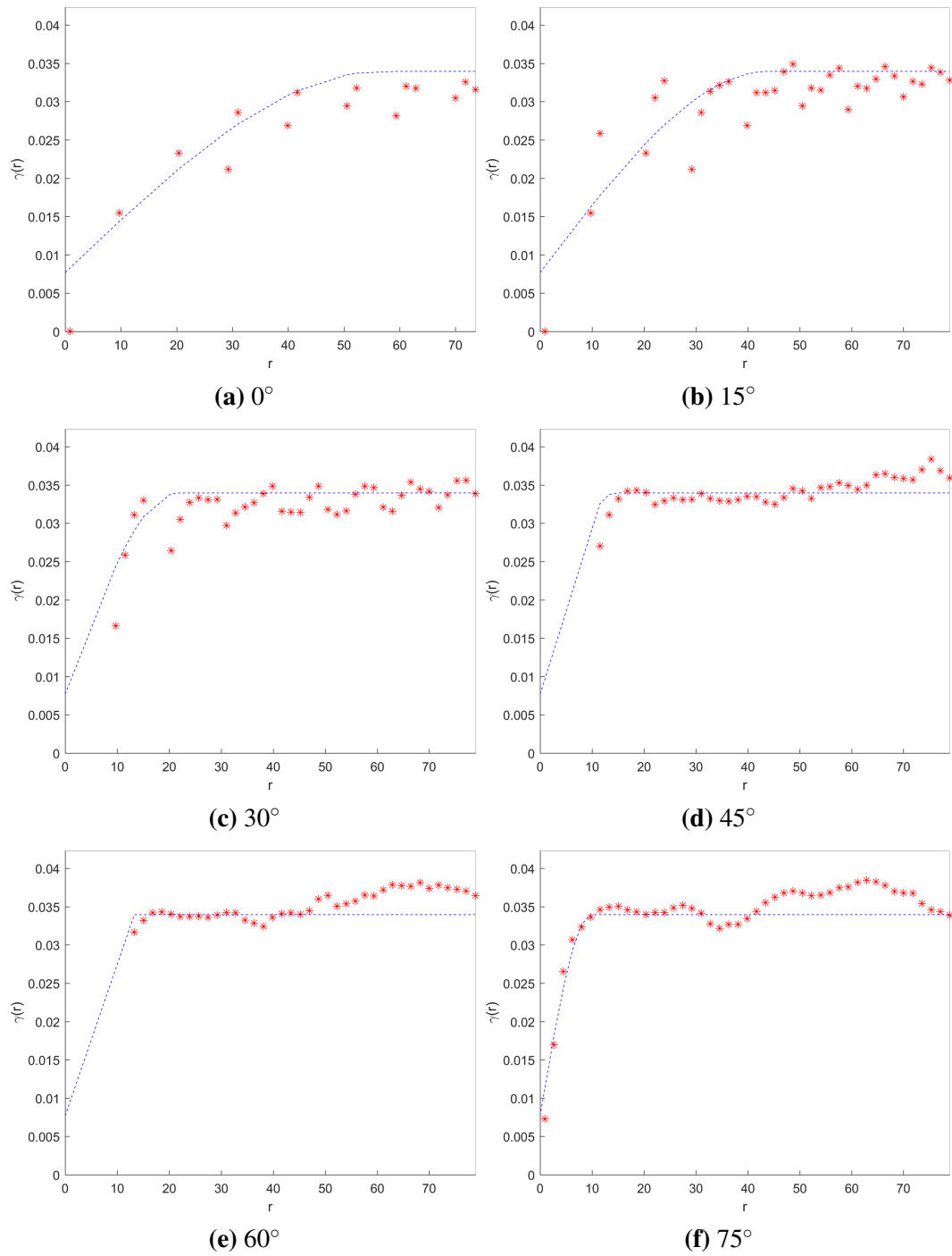


Figure A.2 Fitting of the best theoretical model to the experimental directional variograms of the field. The best model is a Gen. Matérn with parameters $\sigma_z^2 = 4.066$, $\xi_1 = 1.846$, $R = 0.432$, $\phi = -42.3^\circ$, $c_0 = 0.418$, $\nu = 2.832$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.2 Regular Sample

A.2.1 DirVar0



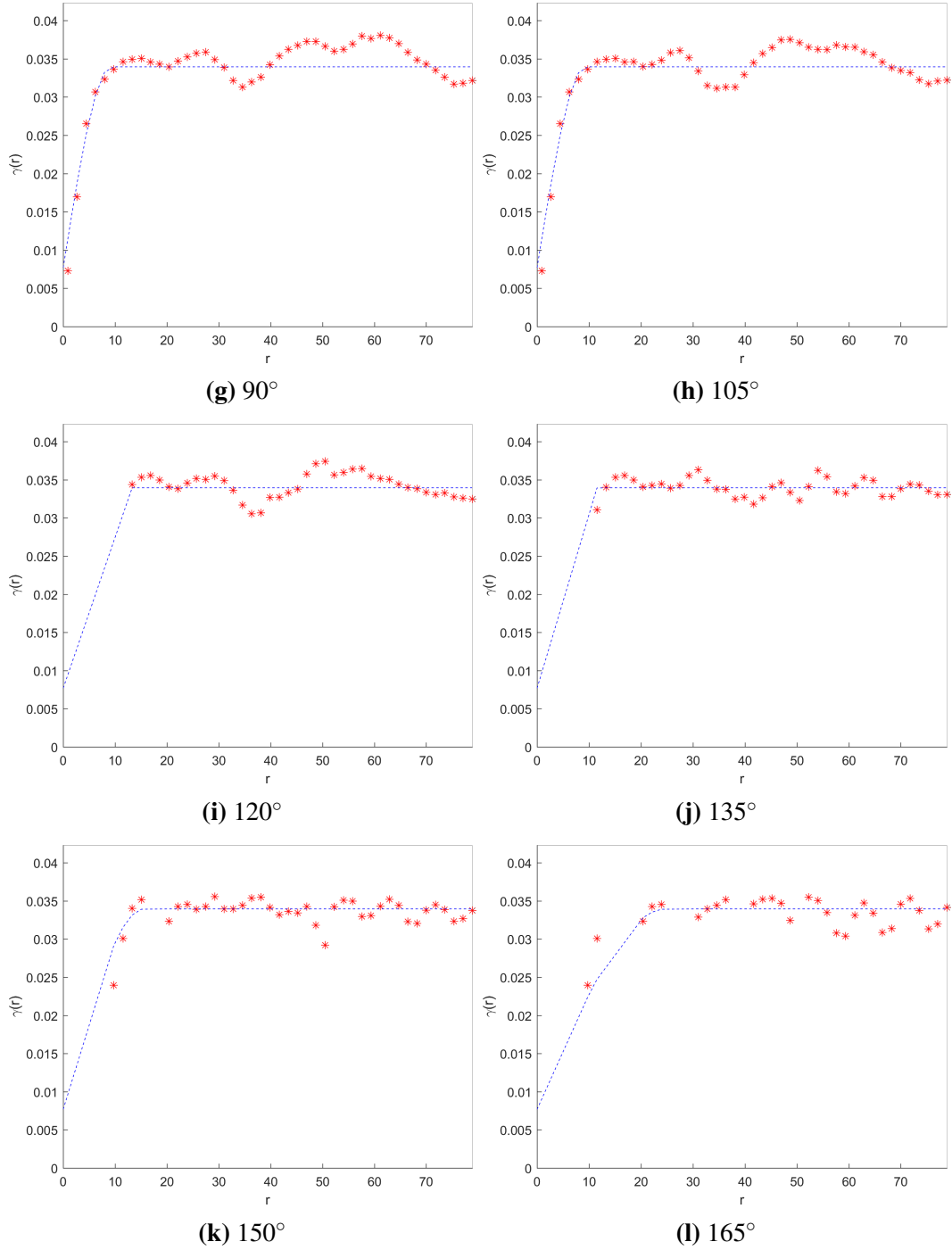
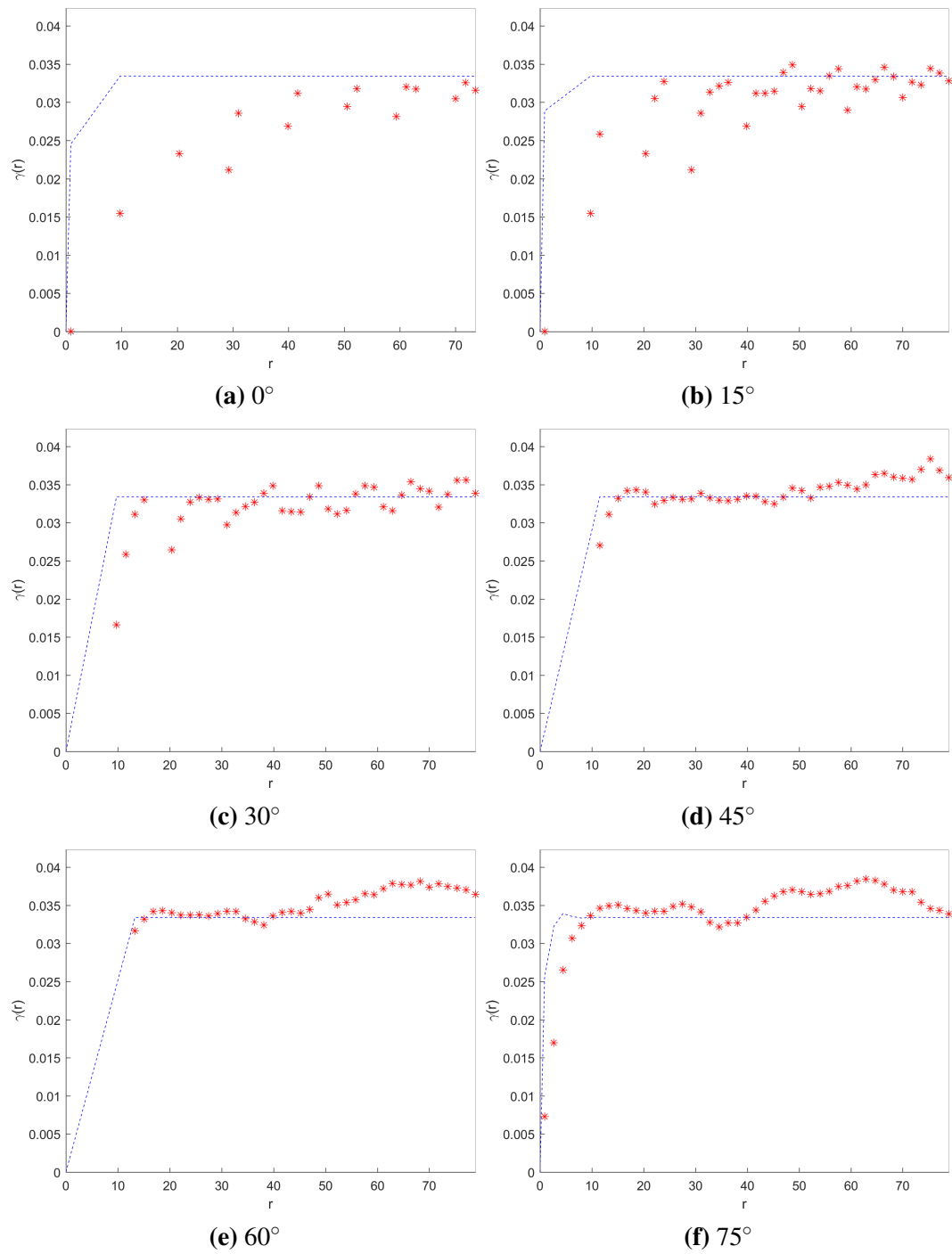


Figure A.3 Fitting of the best theoretical model to the directional experimental (semi-) variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.026$, $\xi_1 = 7.657$, $R = 0.109$, $\phi = -84.4^\circ$, $c_0 = 0.008$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.2.2 DirVar1



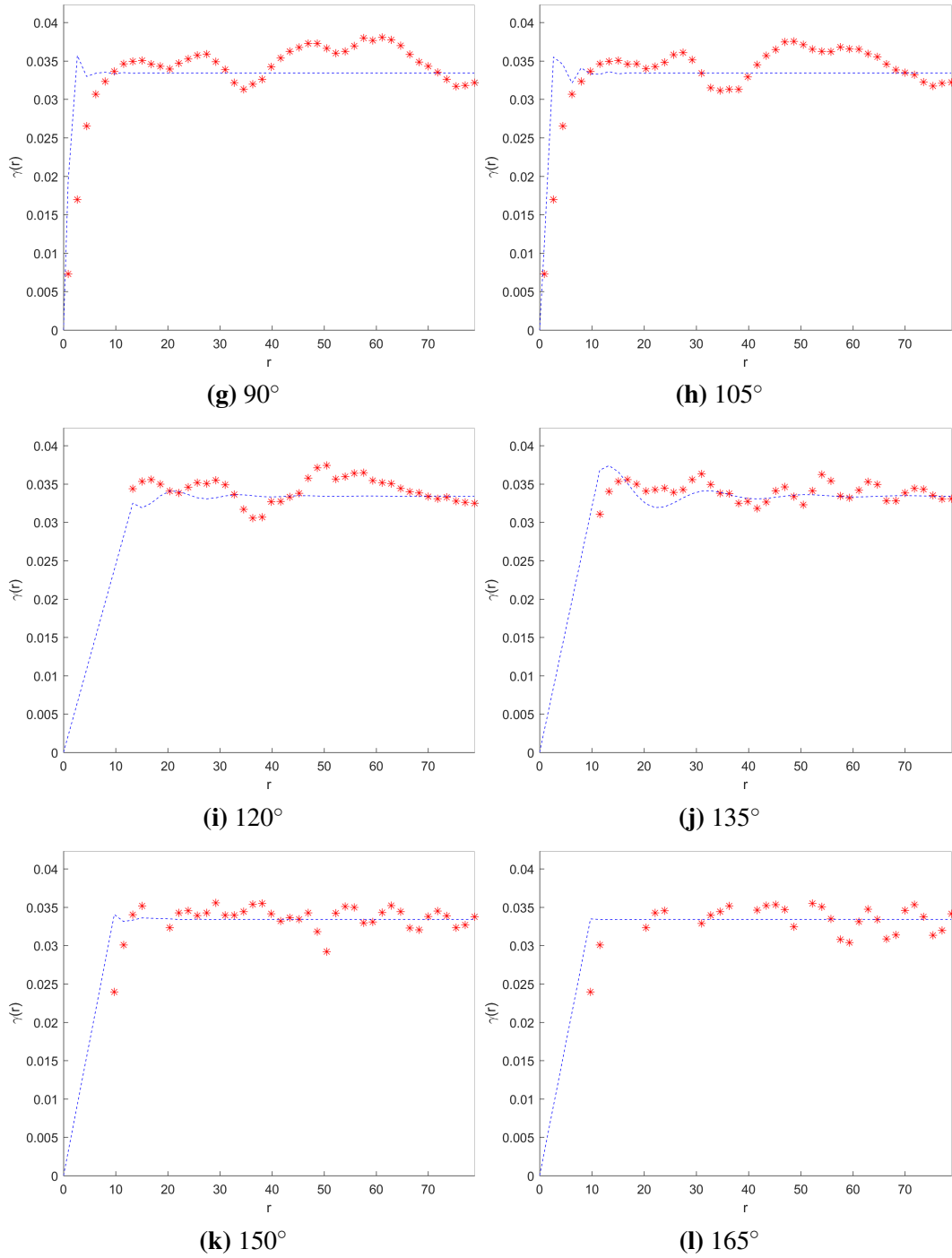
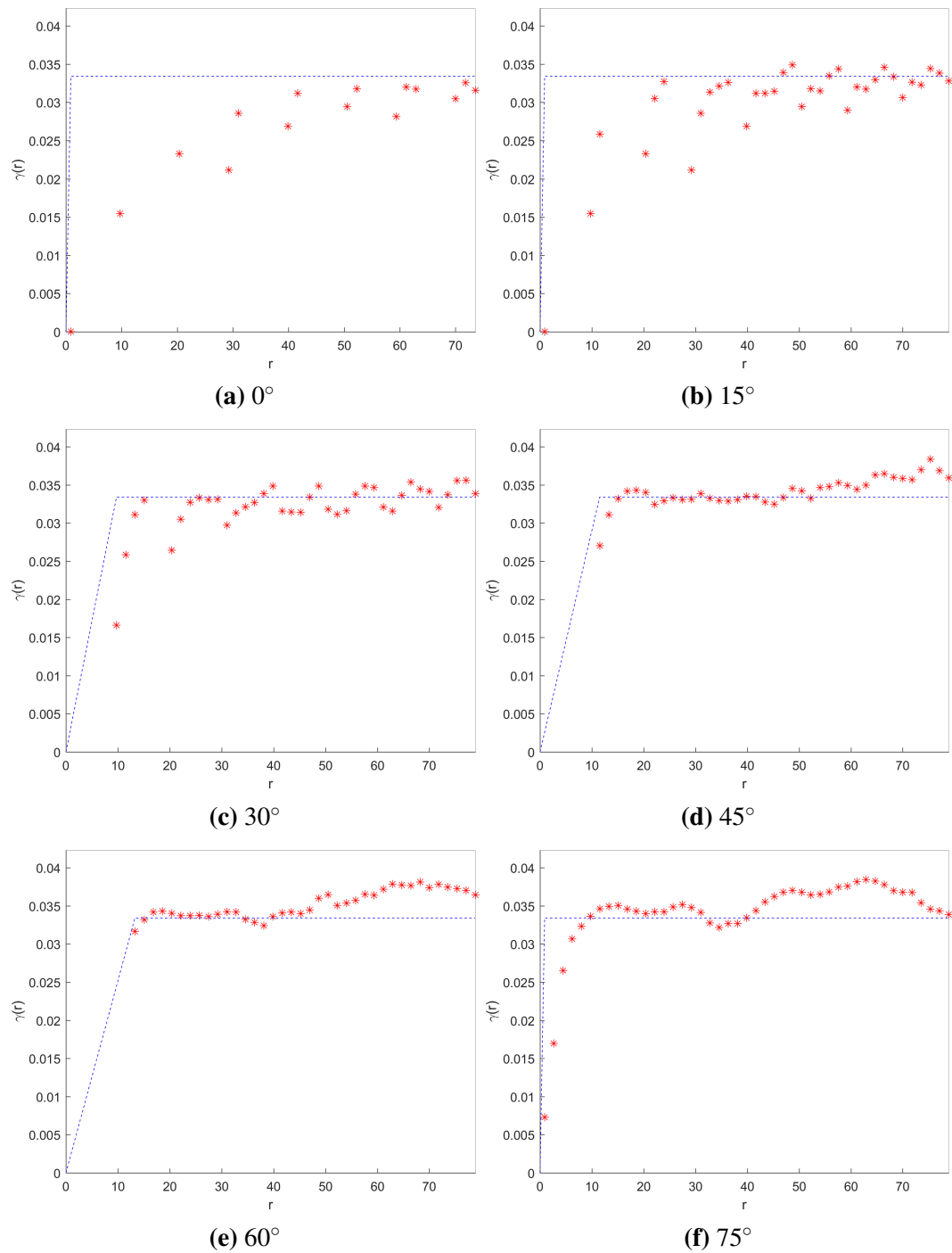


Figure A.4 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.832$, $\xi_1 = 40.338$, $R = 0.307$, $\phi = -50.9^\circ$, $c_0 = 0.000$, $\eta_1 = 1.928$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.2.3 CHI1



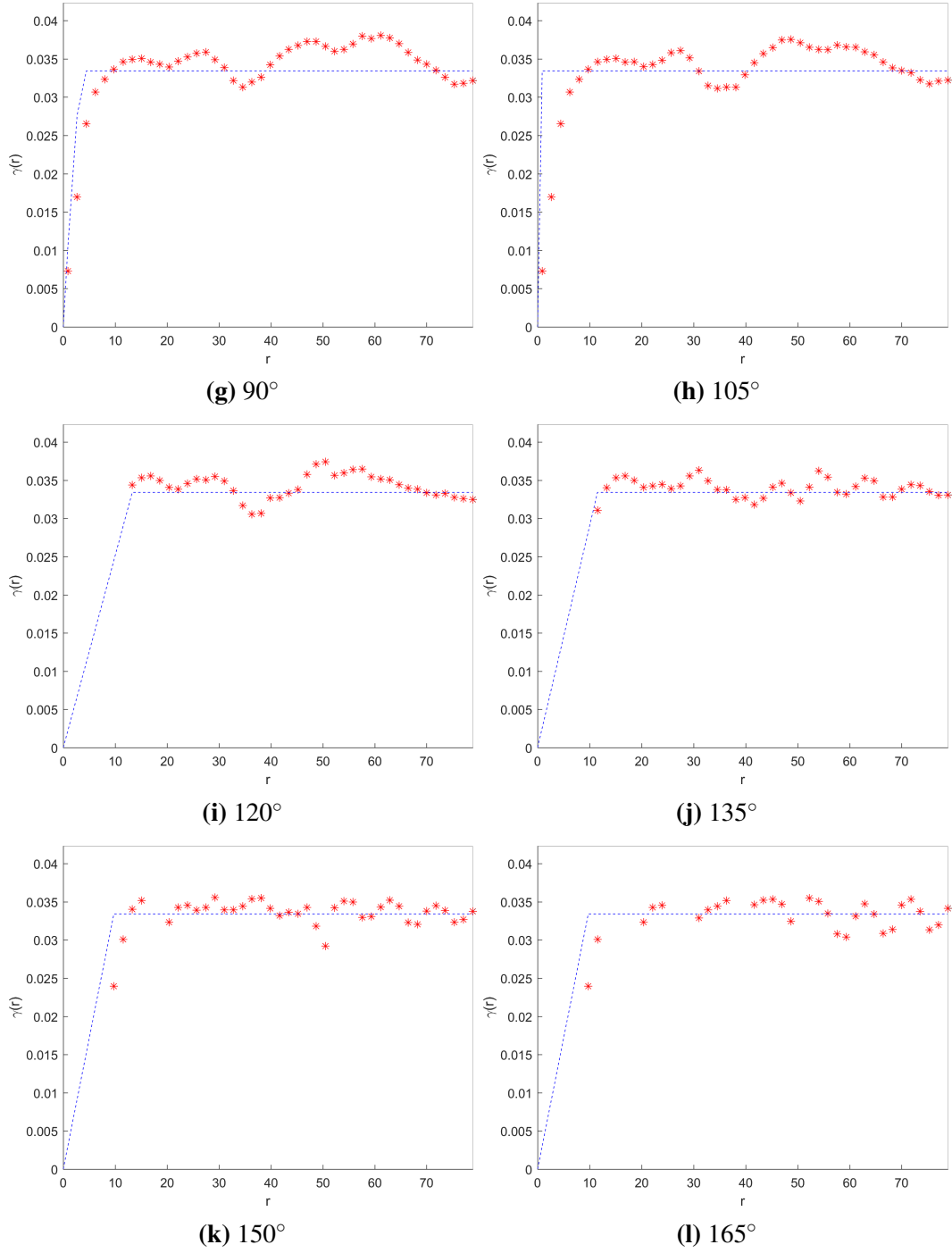
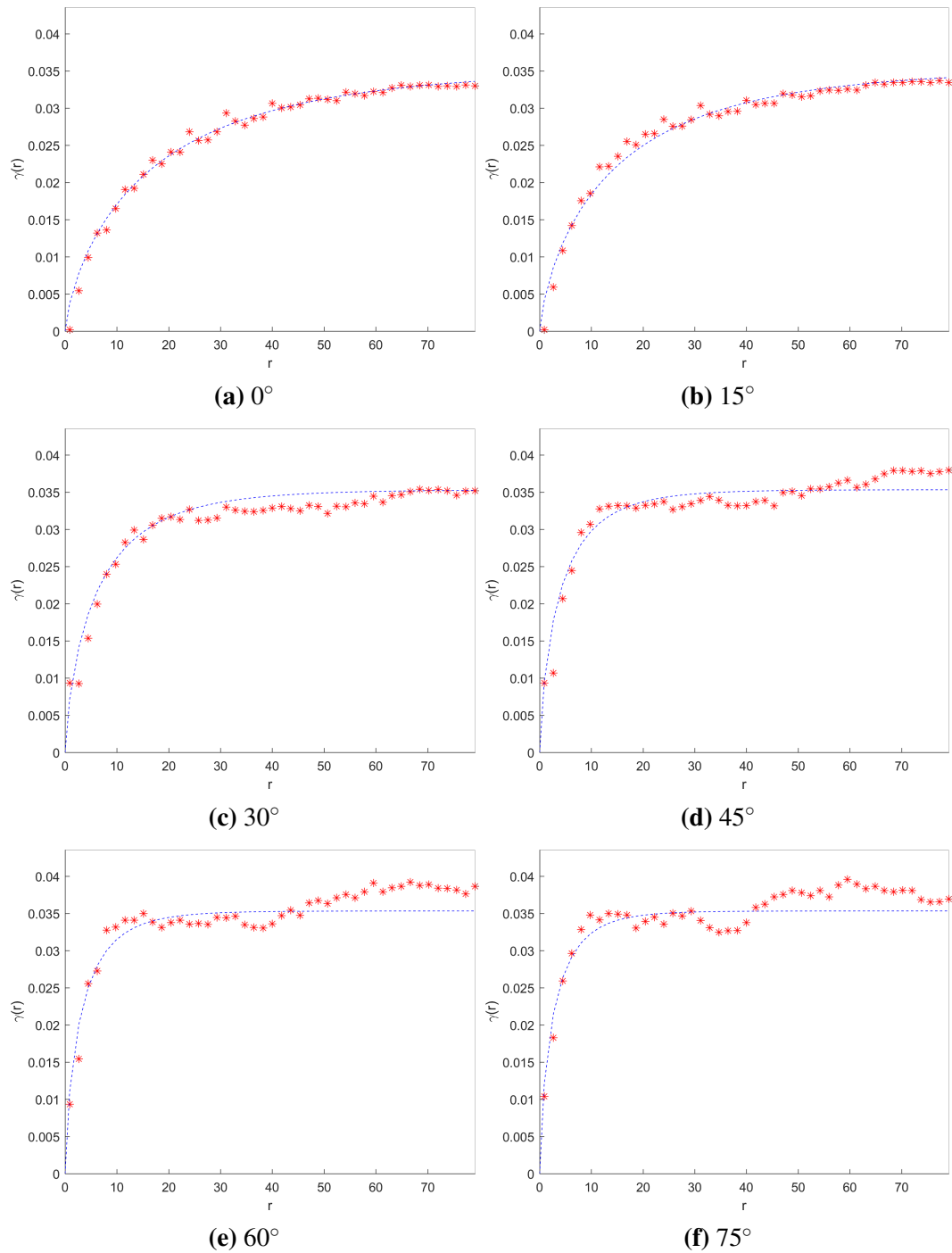


Figure A.5 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spherical with parameters $\sigma_z^2 = 0.033$, $\xi_1 = 79.760.338$, $R = 0.108$, $\phi = -88.5^\circ$, $c_0 = 0.000$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.3 Random Sample

A.3.1 DirVar0



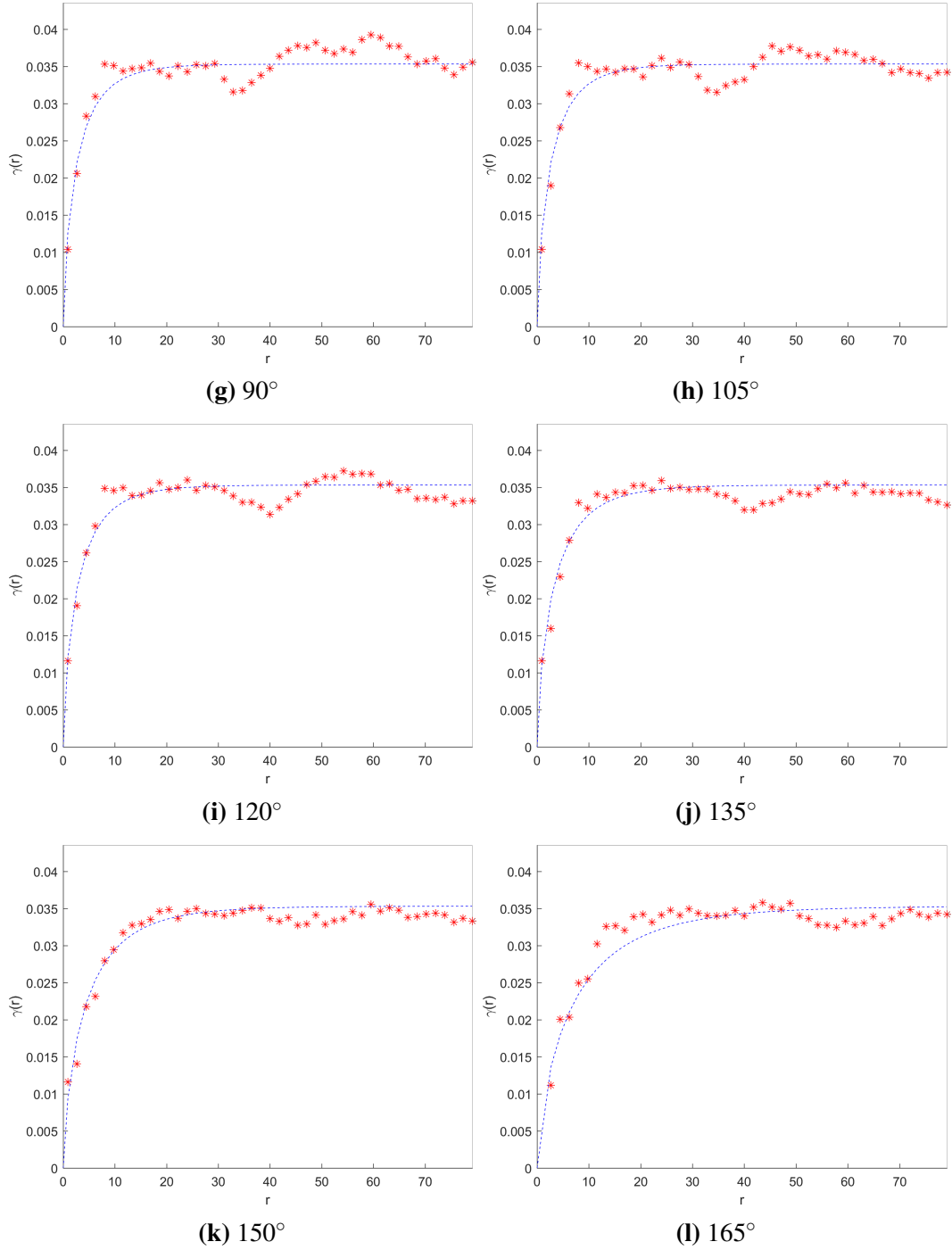
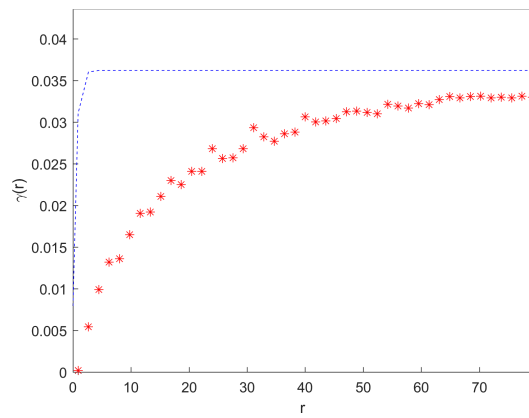
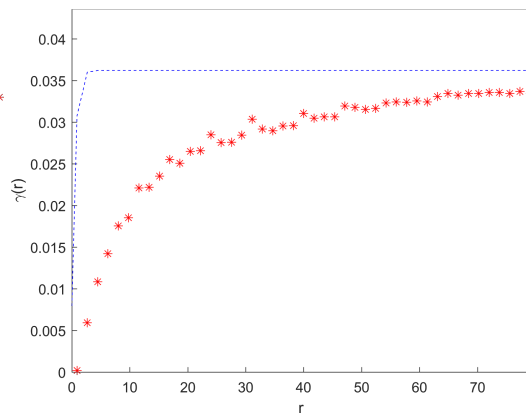
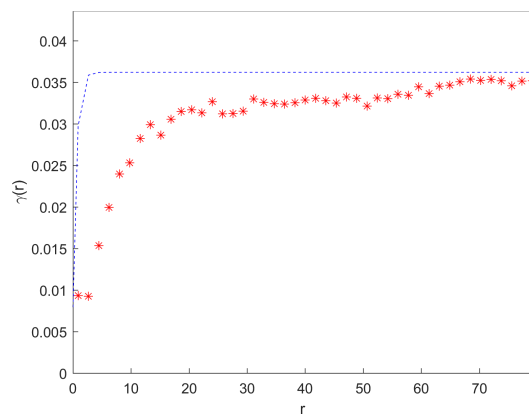
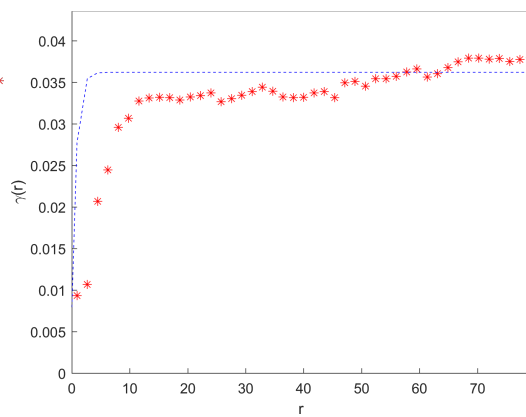
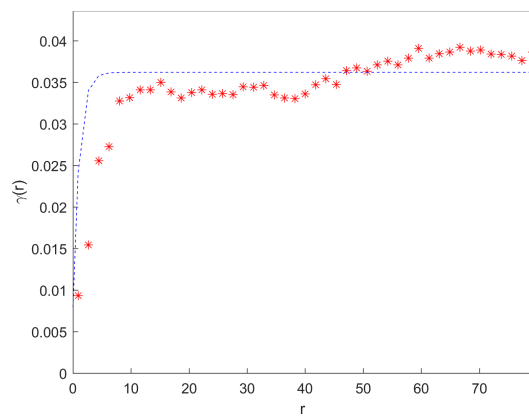
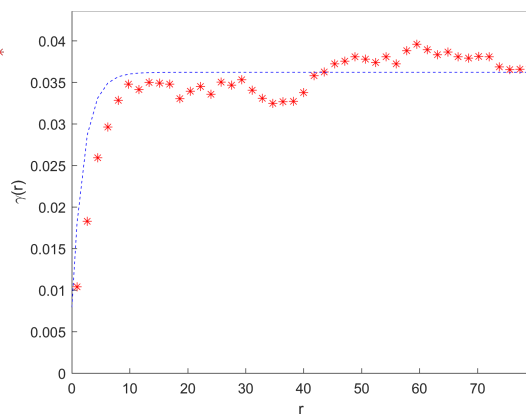


Figure A.6 Fitting of the best theoretical model to the directional experimental (semi-) variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.035$, $\xi_1 = 9.671$, $R = 0.370$, $\phi = -83.4^\circ$, $c_0 = 0.000$, $\nu = 0.734$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.3.2 DirVar1

(a) 0° (b) 15° (c) 30° (d) 45° (e) 60° (f) 75°

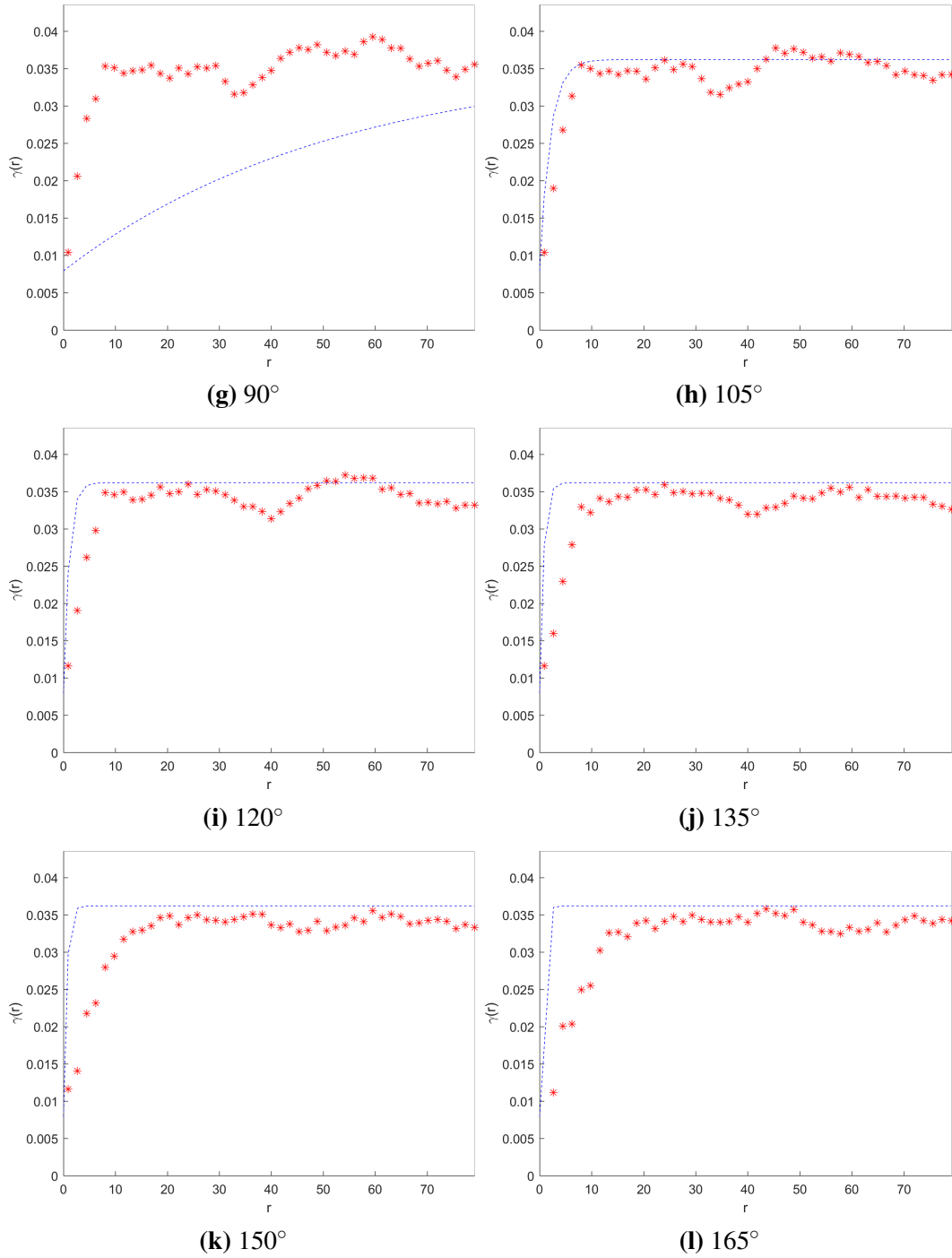
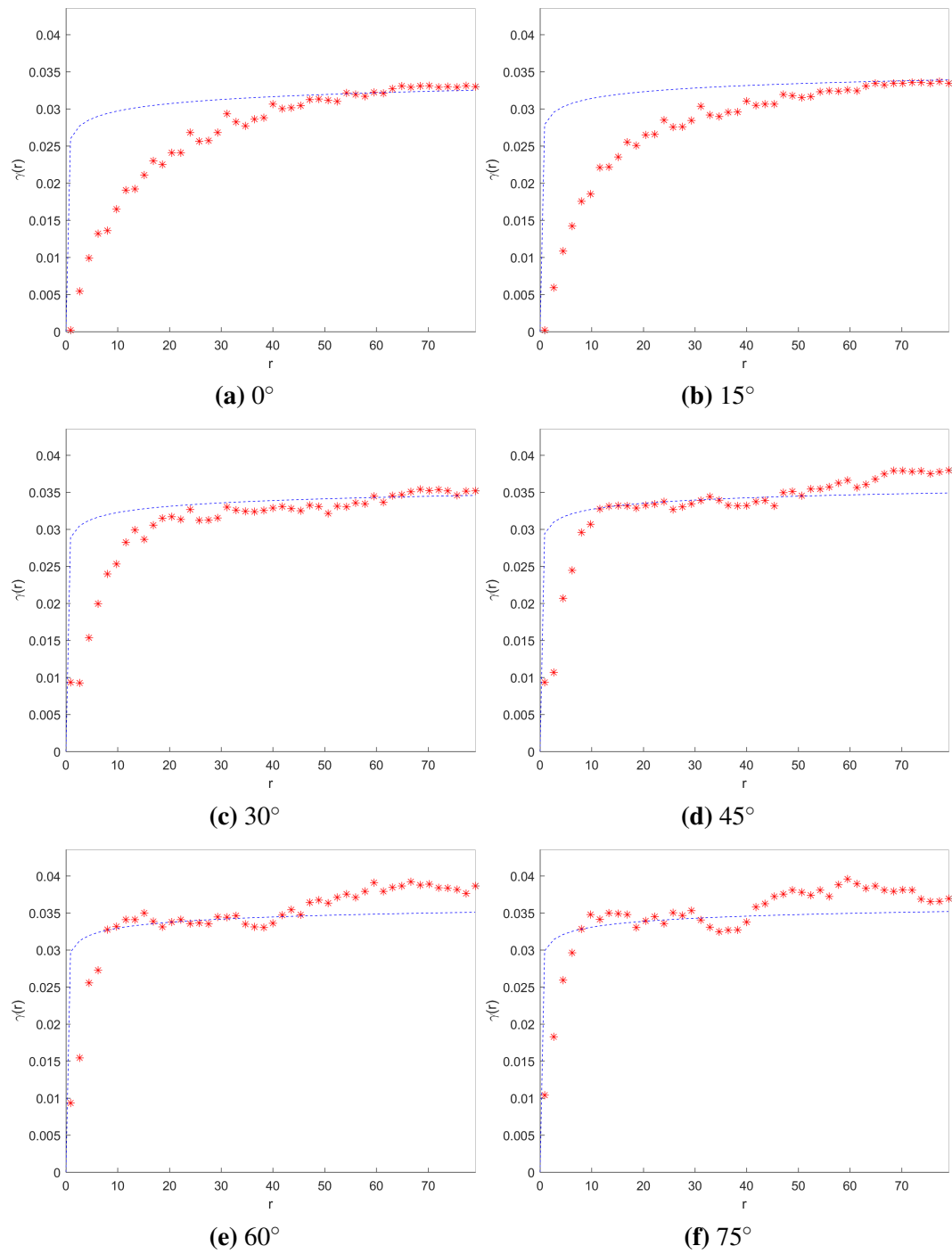


Figure A.7 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Spartan with parameters $\eta_0 = 0.010$, $\xi_1 = 52.606$, $R = 0.522$, $\phi = -90.0^\circ$, $c_0 = 0.008$, $\eta_1 = -1.999$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

A.3.3 CHI1



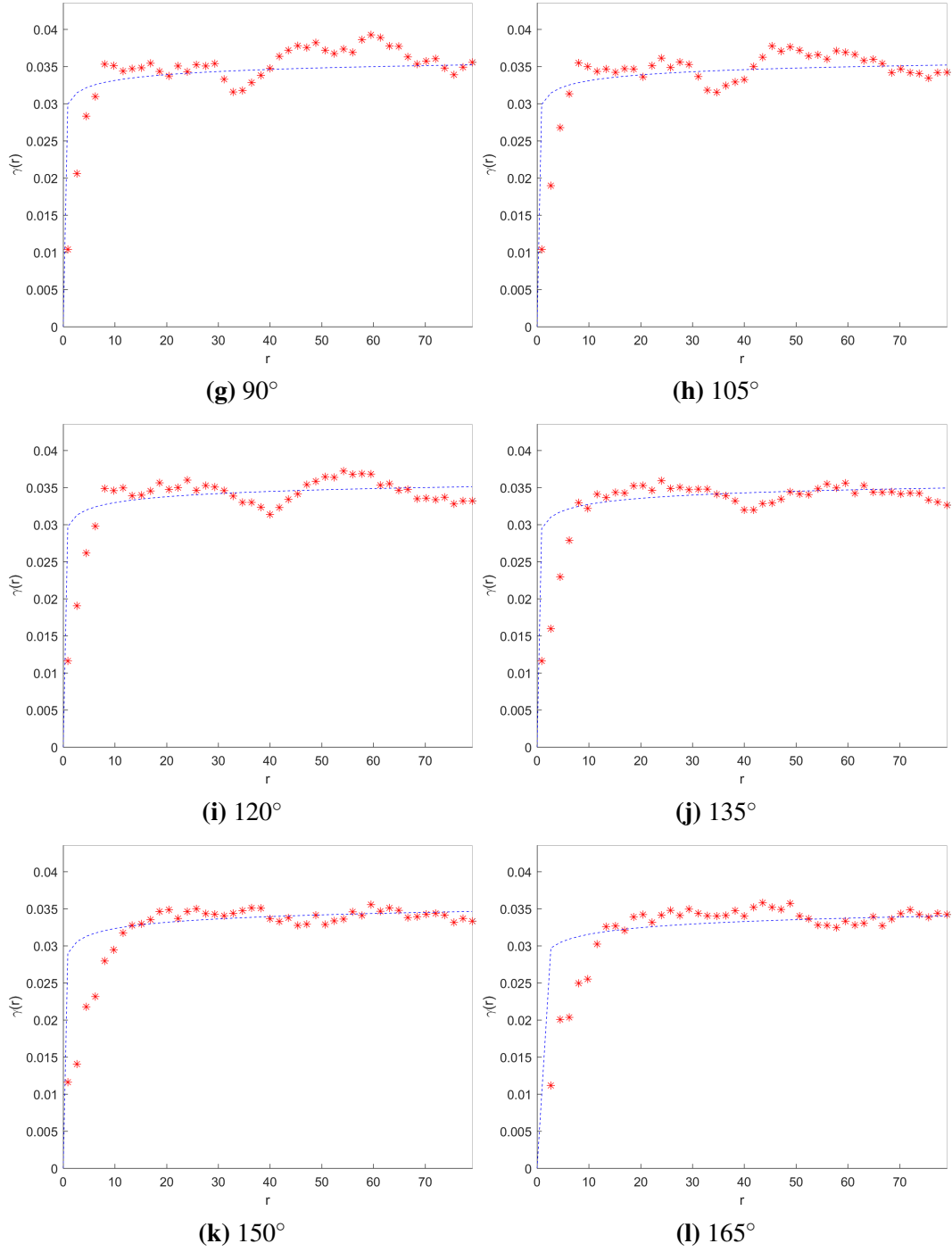


Figure A.8 Fitting of the best theoretical model to the experimental directional (semi-) variograms of the field. The best model is a Gen. Exponential with parameters $\sigma_z^2 = 0.038$, $\xi_1 = 0.021.338$, $R = 0.208$, $\phi = -89.1^\circ$, $c_0 = 0.000$, $\nu = 0.115$. The experimental variogram is calculated with angular tolerance 20° , 45 distance lags and taking into account maximum distance equivalent to about 80.

Appendix B

Codes in MATLAB environment

B.1 Synthetic Dataset

```
1                                     % Synthetic Data – Test
3  clc;clear variables;close all;
5  ##### Preliminary Analysis #####
6  %=====
7  % Construct Synthetic Data
8  [x,y,rf] = randomfield('Mate',[4,3,1.8,20,0.2,2],60,60,0); %random field
9  %Mx = 1.2 + 0.1*x + 0.01*y; %trend
10 data = rf; %synthetic data
11 nx = 60; ny = 60;
13 [row_all, col_all, v_all1] = find(data);
15 % >>>Original Data Plots<<<
17 figure; %Simple plot plus colorbar
18 pcolor(data); shading interp %flat
19 %title('Synthetic data')
20 colorbar
21 set(gca,'XTick',[],'YTick',[])
23 data_tot_vec = data(:); %total image data in vector form
25
27 % Random Sampling (33% of original data)
28 spoints = 0.33*numel(data); %number of sample points
```

```

[~,idx] = datasample(data_tot_vec , spoints , 'Replace' , false); %sampling
29 qpoints = data_tot_vec; qpoints(idx) = 0;
qpoints = reshape(qpoints ,ny,nx); %"matrix form" of unknown points
31 sample = data - qpoints; %"matrix form" of known data
[qrow ,qcol ,qv1] = find(qpoints); %unknown points coordinations and
values - VALIDATION SET
33 [row ,col ,v1] = find(sample); %known points coordinations and values -
TRAINING SET
N = length(v1); %number of known points
35
% Plot Sample
37 figure; %Simple plot plus colorbar
pcolor(sample); shading interp %flat
39 %title('Random Sample')
colorbar
41 set(gca , 'XTick' ,[] , 'YTick' ,[])

43 %>>>Data Histograms & Statistical Moments<<<

45 numbins = 15;
% Total image histogram
47 figure;
%subplot(1,2,1)
49 histfit(data_tot_vec ,numbins)
alpha(0.5)
51 %title('Total image histogram')
% Sample data histogram
53 figure;%subplot(1,2,2)
histfit(v1,numbins)
55 alpha(0.5)
%title('Sample data histogram')
57
% Total image data moments
59 totim_stats = [min(data(:)) max(data(:)) mean(data(:)) median(data(:))
var(data(:)) skewness(data(:)) kurtosis(data(:))];
% "Drill-holes" data moments
61 sample_stats = [min(v1) max(v1) mean(v1) median(v1) var(v1) skewness(v1)
kurtosis(v1)];
63
65 %>>>Normality of data checking and Transformation<<<
67
% Histogram and Normal Probability Plot
figure; %same as the previous figure plus NPP

```

```

%subplot(1,2,1)
69 histfit(v1,numbins)
alpha(0.5)
71 %title('Sample data histogram')
figure;%subplot(1,2,2)
73 nnp = normplot(v1);
h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String',''); % remove normplot title
75 [h_orig,kst_p_orig,ksstat_orig,cv_orig] = kstest(v1(:));

77 v = v1;
qv = qv1;
79 v_all = v_all1;

81 fluc = v(:); Mx = zeros(size(fluc)); qMx = zeros(size(qcol));
qfluc = qv-qMx;
83 fluc_all = v_all-zeros(size(v_all));

85
%>>> Statistical Analysis of Residuals <<<
87
% Residuals/Fluctuations moments
89 fluc_stats = [min(fluc) max(fluc) mean(fluc) median(fluc) var(fluc)
skewness(fluc) kurtosis(fluc)];

91
%Maximum distance
93 c = [col,row];
[k,l] = find(triu(true(N)));
95 d = hypot(c(k,l)-c(1,1),c(k,2)-c(1,2));
% d = triu(pdist2(c,c));
97 ncpc = 0.2;
maxdist = max(max(d))*ncpc;
99 clear k l d
%=====
101
103
%% ##### Experimental Variogram
#####
105 %=====

107 % >>> Initialize Basic Parameters <<<
load('statanal.mat')

```

```

109 c = {col,row}; %known points' coordinations
qc = {qcol,qrow}; %"unknown" points' coordinations
111 %ncpc = 0.2;
N = size(col,1);
113 Nu = size(qcol,1);
%maxdist = hypot(col(1,1)-col(N,1),row(1,1)-row(N,1))*ncpc;
115 nrbins = 45;
phistep = 15;
117 phitol = 20;
N_kr_mod = 3;
119
% >>> Experimental (Semi-)Variogram (anisotropic) <<<
121 x = col; y = row; rf = fluc; iso = 0; flag = 1;
[~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
    high analysis
123 [gexp, nr_pairs, c_centers] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,
    phitol,flag);
gexpmax = max(max(gexp));
125 %=====
127
%% ##### DirVar0 #####
129 %=====

131 % >>> Fitting I and Parameters of Anisotropic Correlation Estimation (s2
    ,xi1,xi2,phi,c0,v or etal) <<<

133 % Desired Models
models = {'Gexp','Gaus','Sphe','Mate','Spar'};
135 n_models = length(models);

137 % Initial Values and Limits for optimization
b = [gexpmax,maxdist*1/3,0.5,10,gexpmax/100]; % s2,xi1,R,phi & c0
139 b_lb = [eps,eps,eps,-90,eps]; b_ub = [gexpmax*1.5,maxdist,30,90,gexpmax
    /5]; %lower and upper limits
bsp = [1000,maxdist*1/3,0.5,10,gexpmax/100]; % eta0, xi1,R,phi & c0
141 bsp_lb = [eps,eps,eps,-90,eps]; bsp_ub = [inf,maxdist,30,90,gexpmax/5];
    %lower and upper limits

143 % Summary cells
model_par0 = {[b,1.5];b;b;[b,1.5];[bsp,1]}; %initial parameters values
145 model_par_lb = {[b_lb,eps];b_lb;b_lb;[b_lb,0.3];[bsp_lb,-2+eps]}; %lower
    bounds

```

```

model_par_ub = {[b_ub,2-eps];b_ub;b_ub;[b_ub,3.5];[bsp_ub,inf]}; %upper
bounds
147
clear b b_lb b_ub bsp bsp_lb bsp_ub
149
% Estimation of Parameters (s2,xi1,xi2,phi,c0,v or etal)
151 bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
iso = 00;objmod = 'NWEr_m';flag = 1;
153 for i=1:n_models
    model.function = models{i,1};
155    model.params0 = model_par0{i,1};
    model.paramslb = model_par_lb{i,1};
157    model.paramsub = model_par_ub{i,1};

159    [bmodel{i,1},fval(i,1),tit{i,1}] =...
        variogramfit(gexp,nr_pairs,c_centers,iso,model,objmod,flag);
161 end

163 % Anisotropy Estimation (#Not Necessary#)
R0(n_models,1) = 0;phi0(n_models,1) = 0;xi10(n_models,1) = 0;xi20(
    n_models,1) = 0;
165 for i=1:n_models
    R0(i,1) = bmodel{i,1}(1,3);
167    phi0(i,1) = bmodel{i,1}(1,4);
    xi10(i,1) = bmodel{i,1}(1,2);
169    xi20(i,1) = bmodel{i,1}(1,2)/R0(i,1);
    if R0(i,1)>1
171        R0(i,1) = 1/R0(i,1);bmodel{i,1}(1,3) = R0(i,1);
        xi101 = xi10(i,1);
173        xi10(i,1) = xi20(i,1); bmodel{i,1}(1,2) = xi10(i,1);
        xi20(i,1) = xi101;
175        if phi0(i,1)>0
            phi0(i,1) = phi0(i,1)-90;
177        else
            phi0(i,1) = phi0(i,1)+90;
179        end
        bmodel{i,1}(1,4) = phi0(i,1);
181    end
end
183
R = mean(R0); phi = mean(phi0);xi1 = mean(xi10);xi2 = mean(xi20);
185

187 % >>> Cross Validation <<<

```

```

189 % Matrices and Cells preallocation
    cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
191 cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;

193 % Inputs definition
    iso = 00; d_col = 1;

195 % Cross Validation
197 for i=1:n_models
    model.function = models{i,1};
199    model.params = bmodel{i,1};
    model.r_ok = [6,6];

201    [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
203        crossval(x,y,rf,iso,d_col,model);
    cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
205 end

207 % Cross Validation Scores
    table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
        'finalscore'};
209 % % table_r = {'Gexp','Gaus','Sphe','Mate1/3','Mate1.0','Mate1.5','Mate2
        .0','Mate2.5';
    % % 'Mate3.0','Mate3.5','Spar'};
211 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

213 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
    FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
215 cv_Ss = [cv_Ss, FinalScore];
    cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
217 clear relMSE FinalScore

219 %Trend addition and Boxcox Inversion
    cv_St(n_models,6)= 0;
221 for i=1:n_models

223     cv_matr{i,1}.Z_tr = cv_matr{i,1}.Z;

225     cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;

227     cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
    cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>
229

```

```

    cv_St(i,:) = correstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores
231 end
233 % Total Cross Validation Scores
235 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
    FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
237 cv_St = [cv_St, FinalScore];
    cv_Stf = array2table(cv_St, 'VariableNames',table_h, 'RowNames',table_r);
239 clear relMSE FinalScore

241 %Plots
    for i=1:n_models
243
        % Stochastic Component's figures
245         cc1 = [col,row,rf(:);qcol,qrow,zeros(Nu,1)];
        cc2 = [col,row,cv_matr{i,1}.Z(:);qcol,qrow,zeros(Nu,1)];
247         Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
        Z2=Z1;
249         for j = 1:size(cc1,1)
            Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
251             Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
        end

253
        figure;pcolor(Z1);%title('Sample Stochastic Component');
255         view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
        figure;pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s',tit{i,1}));
257         view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
        figure;scatter(rf(:),cv_matr{i,1}.Z(:),'filled','d');hold on;
259         dvec1 = [rf(:);cv_matr{i,1}.Z(:)];
        plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
261         axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
        %title('Scatter Plot')
263         xlabel('Observed Data'); ylabel('Estimations');
        figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
265         hold on
        histogram(cv_matr{i,1}.Z(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1
0],'FaceAlpha',0.7)
267         %title('Histograms of Sample Stochastic Component')
        legend({'Original','Estimated'});
269         clear h

```

```

271 % Total Data figures
273 cc3 = [ col ,row ,v1 (:); qcol ,qrow , zeros (Nu,1) ];
275 cc4 = [ col ,row ,cv_matr{i ,1} . Z_ibt (:); qcol ,qrow , zeros (Nu,1) ];
277 Z3(max(cc3 (:,2)),max(cc3 (:,1)))=0; %#ok<SAGROW>
279 Z4=Z3;
281 for j = 1:size(cc3,1)
    Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
    Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
end
283 % figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
% colorbar; set(gca,'XTick',[],'YTick',[]);
285 figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s',tit{i,1}));
colorbar; set(gca,'XTick',[],'YTick',[]);
287 figure; scatter(v1(:),cv_matr{i,1}.Z_ibt(:),'filled','d'); hold on;
dvec2 = [v1(:); cv_matr{i,1}.Z_ibt(:)];
289 plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
291 %title('Scatter Plot')
xlabel('Observed Data')
ylabel('Estimations')
figure; h = histogram(v1(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
293 hold on
histogram(cv_matr{i,1}.Z_ibt(:),'BinEdges',h.BinEdges,'EdgeColor'
,[0.2 1 0],'FaceAlpha',0.7)
295 %title('Histograms of Sample Data')
legend({'Original','Estimated'});
297 clear h

299 clear Z1 Z2 Z3 Z4 dvec1 dvec2

301 end

303 % >>> Ordinary Kriging <<<

305 % Sort models based on cross validation scores
307 [~,ind] = sort(table2array(cv_Stf(:,7)),'descend');

309 % Matrices and Cells preallocation
Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];

```



```

311 kr_matr{N_kr_mod,1}=[];
kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
313 CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];

315 % Inputs definition
xu = qcol; yu = qrow; iso = 00;

317 % Ordinary Kriging
319 for i=1:N_kr_mod
    model.function = models{ind(i),1};
321    model.params = bmodel{ind(i),1};
    model.r_ok = [6,6];

323    [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
325        ordkrig(x,y,rf,xu,yu,iso,model);
    kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
327    table_r2{i,1} = table_r{ind(i),1};

329    %Confidence Intervals (95%)
    CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
331    CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
    CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
333    UNC{i,1} = real(CI1{i,1}.uncer);
    realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
335 end

337 % Kriging Scores
relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
339 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
341 kr_Ssf = array2table(kr_Ss,'VariableNames',table_h,'RowNames',table_r2);
clear relMSE FinalScore

343 %Trend addition and Boxcox Inversion
345 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0;Z_ibt{N_kr_mod,1} = 0;
347 for i=1:N_kr_mod

349     Z_tr{i,1} = Z{i,1};

351     Z_ibt1{i,1} = Z_tr{i,1};

353     Z_ibt{i,1} = real(Z_ibt1{i,1});
    kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

```

```

355     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
357
359 end
359
361 % Total Kriging Scores
361 relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
361 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
363 kr_St = [kr_St, FinalScore];
363 kr_Stf = array2table(kr_St, 'VariableNames',table_h, 'RowNames',table_r2);
365 clear relMSE FinalScore
367
367 %Plots
367 for i=1:N_kr_mod
369
369     % Stochastic Component's figures
371     cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
371     cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
373     Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %ok<SAGROW>
373     Z2=Z1;
375     for j = 1:size(cc1,1)
375         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
377         Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
377     end
379     figure;pcolor(Z1);%title('Original Stochastic Component');
379     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
381     figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
381     Component \n%s',tit{ind(i),1}));
381     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
383     figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
383     dvec1 = [qfluc(:);Z{i,1}(:)];
385     plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
385     +0.5],'r');
385     axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
387     %title('Scatter Plot')
387     xlabel('Observed Data')
389     ylabel('Estimations')
389     figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha'
389     ,0.7);
391     hold on
391     histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
391     'FaceAlpha',0.7)
393     %title('Histograms of Stochastic Component')
393     legend({'Original', 'Estimated'});

```

```

395     clear h

397     % Total Data figures
    cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
399    cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
    Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
401    Z4=Z3;
    for j = 1:size(cc3,1)
403        Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
        Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
405    end
    figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
    Estimation of Original Data \n%s',tit{ind(i),1}));
407    colorbar;set(gca,'XTick',[],'YTick',[]);
    figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
409    dvec2 = [qv1(:);Z_ibt{i,1}(:)];
    plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
    +0.5],'r');
411    axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
    %title('Scatter Plot')
413    xlabel('Observed Data')
    ylabel('Estimations')
415    figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
    hold on
417    histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
    'FaceAlpha',0.7)
    %title('Histograms of Data')
419    legend({'Original','Estimated'});
    figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
    Interval');
421    colorbar;set(gca,'XTick',[],'YTick',[]);
    clear h

423
425    clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

427
429    %% ##### DirVar1 #####
    %=====
431
433    % >>> Fitting I and Anisotropy Estimation <<<

    % Desired Models

```

```

435 models = { 'Gexp', 'Gaus', 'Sphe', 'Mate', 'Spar' };
n_models = length(models);
437
% Initial Values and Limits for optimization
439 b = [gexpmax, maxdist*1/3, gexpmax/100]; % s2, xi & c0
b_lb = [eps, eps, eps]; b_ub = [gexpmax*1.5, maxdist, gexpmax/5]; %lower and
    upper limits
441 bsp = [1000, maxdist*1/3, gexpmax/100]; % eta0, xi, c0
bsp_lb = [eps, eps, eps]; bsp_ub = [inf, maxdist, gexpmax/5]; %lower and
    upper limits
443
% Summary cells
445 model_par0 = {[b, 1.5]; b; b; [b, 1.5]; [bsp, 1]}; %initial parameters values
model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
    bounds
447 model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
    bounds
449 clear b b_lb b_ub bsp bsp_lb bsp_ub
451
% Estimation of Anisotropy
R(n_models, 1)=0; phi(n_models, 1)=0; xi1(n_models, 1)=0; xi2(n_models, 1)=0;%
    matrices preallocation
453 er1(n_models, 1)=0; er2(n_models, 1)=0; %matrices preallocation
objmod = 'NWEr_m'; flag = 1;
455 % nrsampl = 10; samplpc = 50;
for i=1:n_models
457     model.function = models{i, 1};
    model.params0 = model_par0{i, 1};
459     model.paramslb = model_par_lb{i, 1};
    model.paramsub = model_par_ub{i, 1};
461
    [R(i, 1), phi(i, 1), xi1(i, 1), xi2(i, 1), er1(i, 1), er2(i, 1)] = ...
463         aniso_dvf(x, y, rf, model, objmod, ncpc, nrbins, phistep, phitol, flag);
%     [R(i, 1), phi(i, 1), xi1(i, 1), xi2(i, 1), er1(i, 1), er2(i, 1)] = ...
465 %         dirvar_ccv_s(x, y, rf, model, objmod, N_ms, ncpc, nrsampl, samplpc,
    nrbins, phistep, phitol, flag);
467 end
469 %% >>> Fitting II and Parameters of Correlation Estimation (s2, c0, v or
    etal) <<<
471
% Initial Values and Limits for optimization

```

```

b{n_models,1}=[]; b_lb{n_models,1}=[]; b_ub{n_models,1}=[];
473 for i = 1:n_models
    b{i,1} = [gexpmax, xil(i,1), R(i,1), phi(i,1), gexpmax/100];
475    b_lb{i,1} = [eps, xil(i,1), R(i,1), phi(i,1), eps];
    b_ub{i,1} = [inf, xil(i,1), R(i,1), phi(i,1), gexpmax/5];
477 end

479 % Summary cells
model_par0 = {[b{1,1}, 1.1]; b{2,1}; b{3,1}; [b{4,1}, 1.5]; [b{5,1}, 1]}; %
    initial parameters values
481 model_par_lb = {[b_lb{1,1}, eps]; b_lb{2,1}; b_lb{3,1}; [b_lb{4,1}, 0.3]; [
    b_lb{5,1}, -2+eps]}; %lower bounds
model_par_ub = {[b_ub{1,1}, 2]; b_ub{2,1}; b_ub{3,1}; [b_ub{4,1}, 3.5]; [b_ub
    {5,1}, inf]}; %upper bounds
483
% Estimation of Parameters (s2, c0, v or etal)
485 bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
iso = 00; objmod = 'NWEr_m'; flag = 1;
487 for i=1:n_models
    model.function = models{i,1};
489    model.params0 = model_par0{i,1};
    model.paramslb = model_par_lb{i,1};
491    model.paramsub = model_par_ub{i,1};

    [bmodel{i,1}, fval(i,1), tit{i,1}] = ...
        variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
495 end

497
% >>> Cross Validation <<<
499
% Matrices and Cells preallocation
501 cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
    cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;
503
% Inputs definition
505 iso = 00; d_col = 1;

507 % Cross Validation
for i=1:n_models
509    model.function = models{i,1};
    model.params = bmodel{i,1};
511    model.r_ok = [6,6];

```

```

513     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
        crossval(x,y,rf,iso,d_col,model);
515     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
end
517
% Cross Validation Scores
519 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
    'finalscore'};
table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};
521
relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
523 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
cv_Ss = [cv_Ss, FinalScore];
525 cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
clear relMSE FinalScore
527
%Trend addition and Boxcox Inversion
529 cv_St(n_models,6)= 0;
for i=1:n_models
531
    cv_matr{i,1}.Z_tr = cv_matr{i,1}.Z;
533
    cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
535
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
537    cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1);
539
    cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores
541 end
543
% Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
545 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
547 cv_Stf = array2table(cv_St,'VariableNames',table_h,'RowNames',table_r);
clear relMSE FinalScore
549
%Plots
551 for i=1:n_models
553
    % Stochastic Component's figures
    cc1 = [col,row,rf(:);qcol,qrow,zeros(Nu,1)];
555    cc2 = [col,row,cv_matr{i,1}.Z(:);qcol,qrow,zeros(Nu,1)];

```

```

557 Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
Z2=Z1;
559 for j = 1:size(cc1,1)
    Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
    Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
561 end

563 figure;pcolor(Z1);%title('Sample Stochastic Component');
view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
565 figure;pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s',tit{i,1}));
view(2);shading interp;colorbar;set(gca,'XTick',[],'YTick',[]);
567 figure;scatter(rf(:),cv_matr{i,1}.Z(:),'filled','d');hold on;
dvec1 = [rf(:);cv_matr{i,1}.Z(:)];
569 plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
571 %title('Scatter Plot')
xlabel('Observed Data'); ylabel('Estimations');
573 figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
575 histogram(cv_matr{i,1}.Z(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1
0],'FaceAlpha',0.7)
%title('Histograms of Sample Stochastic Component')
577 legend({'Original','Estimated'});
clear h
579

581 % Total Data figures
cc3 = [col,row,v1(:);qcol,qrow,zeros(Nu,1)];
583 cc4 = [col,row,cv_matr{i,1}.Z_ibt(:);qcol,qrow,zeros(Nu,1)];
Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
585 Z4=Z3;
for j = 1:size(cc3,1)
587     Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
    Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
589 end
% figure;pcolor(Z3);view(2);shading interp;%title('Original Sample
');
591 % colorbar;set(gca,'XTick',[],'YTick',[]);
figure;pcolor(Z4);view(2);shading interp;%title(sprintf('Estimation
of Original Sample \n%s',tit{i,1}));
593 colorbar;set(gca,'XTick',[],'YTick',[]);
figure;scatter(v1(:),cv_matr{i,1}.Z_ibt(:),'filled','d');hold on;

```

```

595     dvec2 = [v1(:);cv_matr{i,1}.Z_ibt(:)];
        plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5], 'r');
597     axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
        %title('Scatter Plot')
599     xlabel('Observed Data')
        ylabel('Estimations')
601     figure;h = histogram(v1(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
        hold on
603     histogram(cv_matr{i,1}.Z_ibt(:),'BinEdges',h.BinEdges,'EdgeColor'
,[0.2 1 0],'FaceAlpha',0.7)
        %title('Histograms of Sample Data')
605     legend({'Original','Estimated'});
        clear h
607
        clear Z1 Z2 Z3 Z4 dvec1 dvec2
609
end
611

613 % >>> Ordinary Kriging <<<

615 % Sort models based on cross validation scores
[~,ind] = sort(table2array(cv_Stf(:,7)),'descend');
617
% Matrices and Cells preallocation
619 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
kr_matr{N_kr_mod,1}=[];
621 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];
623
% Inputs definition
625 xu = qcol; yu = qrow; iso = 00;

627 % Ordinary Kriging
for i=1:N_kr_mod
629     model.function = models{ind(i),1};
        model.params = bmodel{ind(i),1};
631     model.r_ok = [6,6];

        [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
            ordkrig(x,y,rf,xu,yu,iso,model);
633
        kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
635     table_r2{i,1} = table_r{ind(i),1};

```



```

637     %Confidence Intervals (95%)
639     CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
        CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
641     CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
        UNC{i,1} = real(CI1{i,1}.uncer);
643     realCI(i,1) = isreal(CI1{i,1}.uncer);
end
645
% Kriging Scores
647 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
649 kr_Ss = [kr_Ss, FinalScore];
kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
651 clear relMSE FinalScore

653 %Trend addition and Boxcox Inversion
kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
655 Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
for i=1:N_kr_mod
657
        Z_tr{i,1} = Z{i,1};
659
        Z_ibt1{i,1} = Z_tr{i,1};
661
        Z_ibt{i,1} = real(Z_ibt1{i,1});
663     kr_realZ(i,1) = isreal(Z_ibt1{i,1});

665     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
667 end

669 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
671 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
673 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore

675 %Plots
677 for i=1:N_kr_mod

679     % Stochastic Component's figures
        cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];

```

```

681 cc2 = [ col ,row , rf (:);qcol ,qrow ,Z{i ,1 }];
Z1(max(cc1 (: ,2)) ,max(cc1 (: ,1)))=0; %#ok<SAGROW>
683 Z2=Z1;
for j = 1:size(cc1 ,1)
685     Z1(cc1(j ,2) ,cc1(j ,1)) = cc1(j ,3);
        Z2(cc2(j ,2) ,cc2(j ,1)) = cc2(j ,3);
687 end
figure;pcolor(Z1);%title(' Original Stochastic Component ');
689 view(2);shading interp; colorbar;set(gca , 'XTick' ,[], 'YTick' ,[]);
figure;pcolor(Z2); %title(sprintf(' Estimation of Stochastic
Component \n%s' , tit{ind(i) ,1}));
691 view(2);shading interp; colorbar;set(gca , 'XTick' ,[], 'YTick' ,[]);
figure;scatter(qfluc (:) ,Z{i ,1 }(:) , 'filled' , 'd');hold on;
693 dvec1 = [qfluc (:);Z{i ,1 }(:)];
plot([min(dvec1) -0.5,max(dvec1) +0.5] ,[min(dvec1) -0.5,max(dvec1)
+0.5] , 'r');
695 axis([min(dvec1) -0.5,max(dvec1) +0.5 ,min(dvec1) -0.5,max(dvec1) +0.5])
%title(' Scatter Plot ')
697 xlabel ( 'Observed Data' )
ylabel ( 'Estimations' )
699 figure;h = histogram(qfluc (:) ,16 , 'FaceColor' , [0 0 1] , 'FaceAlpha'
,0.7);
hold on
701 histogram(Z{i ,1 }(:) , 'BinEdges' ,h.BinEdges , 'FaceColor' , [0.2 1 0] , '
FaceAlpha' ,0.7)
%title(' Histograms of Stochastic Component ')
703 legend({ 'Original' , 'Estimated' });
clear h
705
% Total Data figures
707 cc3 = [ col ,row ,v1 (:);qcol ,qrow ,Z_ibt{i ,1 }];
cc4 = [ col ,row ,zeros (size (col ,1) ,1); qcol ,qrow ,UNC{i ,1 }];
709 Z3(max(cc3 (: ,2)) ,max(cc3 (: ,1)))=0; %#ok<SAGROW>
Z4=Z3;
711 for j = 1:size(cc3 ,1)
        Z3(cc3(j ,2) ,cc3(j ,1)) = cc3(j ,3);
713     Z4(cc4(j ,2) ,cc4(j ,1)) = cc4(j ,3);
end
715 figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s' , tit{ind(i) ,1}));
colorbar;set(gca , 'XTick' ,[], 'YTick' ,[]);
717 figure;scatter(qv1 (:) ,Z_ibt{i ,1 }(:) , 'filled' , 'd');hold on;
dvec2 = [qv1 (:);Z_ibt{i ,1 }(:)];

```

```

719     plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5], 'r');
721     axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
722     %title('Scatter Plot')
723     xlabel('Observed Data')
724     ylabel('Estimations')
725     figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
726     hold on
727     histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
728     %title('Histograms of Data')
729     legend({'Original','Estimated'});
730     figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
731     colorbar;set(gca,'XTick',[],'YTick',[]);
732     clear h
733
734     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2
735 end
%
```

C:/Users/Vasilis/Desktop/AB/Dissertation/DiplomaThesis/MyThesis/Appendix2/Test4Appendix.m

B.2 Regular Sample

```

2                                     % Dissertation
3                                     % Regular Sample
4
5
6     clc;clear variables;close all;
7
8     %##### Preliminary Analysis #####
9     %=====
10
11     % Basic parameters
12     n_data = 2; %if 2 normalised velocities are used,else if 1 are used
13         %velocities in km/s,else if 0 the original velocities are
14         used
15     n_dr = 10; %distance between drills
```

```

14 % Load Data
16 load marm.in
ny = 122; nx = 384; %>>a priori known size of the image<<
18 marm_rf = reshape(marm,ny,nx);
marm_rf = flipud(marm_rf);
20
idata0 = marm_rf; %original velocities (m/s)
22 idata1 = marm_rf/1000;%velocities in km/s
idata2 = marm_rf/max(max(marm_rf)); %normalised velocities
24
color = 'white'; fsize = 18.0; font = 'Cambria'; fweig = 'bold';
26
if n_data==0
28     data = idata0;
    colbtitle = 'Velocity (m/s)';
30 elseif n_data==1
    data = idata1;
32     colbtitle = 'Velocity (km/s)';
    elseif n_data==2
34     data = idata2;
    colbtitle = 'Normalised Velocity';
36 else
    error('Error: Not valid value for n_data input. It must be 0,1 or 2.
    ')
38 end

40 [row_all , col_all , v_all1] = find(data);

42 % >>>Original Data Plots<<<

44 figure;%Simple plot
pcolor(data); shading interp %flat
46 %title('Geological Section (Velocity Model)')
set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',[])
48
figure; %Simple plot plus colorbar
50 pcolor(data); shading interp %flat
%title('Geological Section (Velocity Model)')
52 c = colorbar;
c.Label.String = colbtitle;
54 set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',[])

56 figure; %Plot with actual distances and colorbar

```

```

pcolor(flipud(data)); shading interp %flat
58 %title('Geological Section (Velocity Model)')
c = colorbar;
60 c.Label.String = colbtitle;
xlabel('Distance (km)');
62 ylabel('Depth (km)');
ax = gca;
64 ax.XAxisLocation = 'top';
ax.XLim = [0 nx]; ax.YLim = [0 ny];
66 ax.XTick = (0:9.2)*nx/9.2;
ax.XTickLabel = (ax.XTick)*9.2/nx;
68 ax.YTick = (0:0.5:3)*ny/3;
ax.YTickLabel = (ax.YTick)*3/ny;
70 ax.YDir = 'reverse';

72 data_tot_vec = data(:); %total image data in vector form

74 % >>>Define "drill-holes"<<<
data2 = data;
76 drl = floor((nx - floor(nx/n_dr)*n_dr)/2)+1; %location of first "
    drilling" on x axis
data2(:,drl:n_dr:nx)= 0;
78 [qrow,qcol,qv1] = find(data2); %unknown points coordinations and values
drills = data - data2; %square form of "drill-holes"
80 [row,col,v1] = find(drills); %known points coordinations and values -
    TRAINING SET
N = length(v1); %number of known points

82 % >>>Sample ("drill-holes") Plot<<<

84 figure; %Simple plot
pcolor(drills); shading interp %flat
86 %title('"Drill-holes": part of geological section taken as data')
88 set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',[])

90 figure; %Simple plot plus colorbar
pcolor(drills); shading interp %flat
92 %title('"Drill-holes": part of geological section taken as data')
c1 = colorbar;
94 c1.Label.String = colbtitle;
set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',[])
96 figure; %Plot with actual distances and colorbar
98 pcolor(flipud(drills)); shading interp %flat

```

```

%title('Geological Section (Velocity Model)')
100 c1 = colorbar;
    c1.Label.String = colbtitle;
102 xlabel('Distance (km)');
    ylabel('Depth (km)');
104 ax = gca;
    ax.XAxisLocation = 'top';
106 ax.XLim = [0 nx]; ax.YLim = [0 ny];
    ax.XTick = (0:9.2)*nx/9.2;
108 ax.XTickLabel = (ax.XTick)*9.2/nx;
    ax.YTick = (0:0.5:3)*ny/3;
110 ax.YTickLabel = (ax.YTick)*3/ny;
    ax.YDir = 'reverse';
112
%>>>Data Histograms & Statistical Moments<<<
114
    numbins = 15;
116 % Total image histogram
    figure;
118 %subplot(1,2,1)
    histfit(data_tot_vec,numbins)
120 alpha(0.5)
    %title('Total image histogram')
122 % "Drills-holes" data histogram
    figure;%subplot(1,2,2)
124 histfit(v1,numbins)
    alpha(0.5)
126 %title('Drill-holes data histogram')

128 % Total image data moments
    totim_stats = [min(data_tot_vec) max(data_tot_vec) mean(data_tot_vec)
        median(data_tot_vec) var(data_tot_vec) skewness(data_tot_vec)
        kurtosis(data_tot_vec)];
130 % "Drill-holes" data moments
    drills_stats = [min(v1) max(v1) mean(v1) median(v1) var(v1) skewness(v1)
        kurtosis(v1)];
132

134 %>>>Normality of data checking and Transformation<<<

136 % Histogram and Normal Probability Plot
    figure;%same as the previous figure plus NPP
138 %subplot(1,2,1)
    histfit(v1,numbins)

```

```

140 alpha(0.5)
    %title('Drill-holes data histogram')
142 figure; %subplot(1,2,2)
    nnp = normplot(v1);
144 h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
    '); % remove normplot title
    [h_orig,kst_p_orig,ksstat_orig,cv_orig] = kstest(v1(:));
146
    % BoxCox Transformation
148 [v, lambda] = boxcox(v1); %known points boxcox transformation
    qv = (qv1.^lambda -1)/lambda; %unknown points boxcox transformation
150 v_all = (v_all1.^lambda -1)/lambda; %all points boxcox transformation
    figure;
152 %subplot(1,2,1)
    histfit(v,numbins)
154 alpha(0.5)
    %title('Transformed data histogram')
156 figure; %subplot(1,2,2)
    normplot(v)
158 h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
    '); % remove normplot title
    [h_bxtr,p_bxtr,ksstat_bxtr,cv_bxtr] = kstest(v(:));
160
    %>>> Data Trend Estimation <<<
162 nfr = 2;
    trmodel = 'linear';
164 v_trend = reshape(v,ny,N/ny);
    [fluc,Mx,Mx_func,a,trend_scores,dfreq,a_trends] = detrendv(col,row,
        v_trend,nfr,trmodel,0);
166 qMx = Mx_func(qcol,qrow,a); %trend on the unknown points
    qfluc = qv-qMx; %fluctuations/residuals on the unknown points
168
    %Plot data & trend
170 X = col(:); Y = row(:); rf1 = v_trend(:);
    figure;
172 scatter3(X,flipud(Y),rf1,'r','filled')
    hold on
174 Xfit = min(X):1:max(X); nxfit = length(Xfit);
    Yfit = min(Y):1:max(Y); nyfit = length(Yfit);
176 [XFIT,YFIT] = meshgrid(Xfit,Yfit);
    VFIT = Mx_func(XFIT(:),YFIT(:),a);
178 VFIT = flipud(reshape(VFIT,nyfit,nxfit));
    mesh(XFIT,YFIT,VFIT)
180 colorbar

```

```

%xlabel('x')
182 %ylabel('y')
    xlabel('Alongside Section')
184 ylabel('Depth')
    zlabel('Transformed Normalised Velocity')
186 %set(gca,'YTick',flipud(get(gca,'YTick')));
    %set(gca,'YDir','reverse');
188 %title('Data & Trend Model')
    view(-52,6)
190 shading interp
    c = colorbar; c.Label.String = 'Transformed Normalised Velocity';
192
%Detrend whole dataset
194 Mx_all = Mx_func(col_all,row_all,a); %trend on the unknown points
    fluc_all = v_all-Mx_all; %fluctuations/residuals on the unknown points
196 fluc_all = reshape(fluc_all,ny,nx);

198 figure; %Plot with actual distances and colorbar
    pcolor(flipud(fluc_all)); shading interp %flat
200 %title('Detrended Geological Section')
    c1 = colorbar;
202 % c1.Label.String = colbtitle;
    xlabel('Distance (km)');
204 ylabel('Depth (km)');
    ax = gca;
206 ax.XAxisLocation = 'top';
    ax.XLim = [0 nx]; ax.YLim = [0 ny];
208 ax.XTick = (0:9.2)*nx/9.2;
    ax.XTickLabel = (ax.XTick)*9.2/nx;
210 ax.YTick = (0:0.5:3)*ny/3;
    ax.YTickLabel = (ax.YTick)*3/ny;
212 ax.YDir = 'reverse';

214 %>>> Statistical Analysis of Residuals <<<

216 % Residuals/Fluctuations moments
    fluc_stats = [min(fluc) max(fluc) mean(fluc) median(fluc) var(fluc)
        skewness(fluc) kurtosis(fluc)];
218
% Histogram and Normal Probablity Plot
220 figure;
    %subplot(1,2,1)
222 histfit(fluc,numbins)
    alpha(0.5)

```



```

224 %title('Histogram of detrended data')
figure; %subplot(1,2,2)
226 normplot(fluc)
h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
'); % remove normplot title
228 [h_fluc,p_fluc,ksstat_fluc,cv_fluc] = kstest(fluc(:));
%=====
230
232
%% ##### Experimental Variogram
#####
234 %=====

236 % >>> Initialize Basic Parameters <<<
c = {col,row}; %known points' coordinations
238 qc = {qcol,qrow}; %"unknown" points' coordinations
ncpc = 0.2;
240 N = size(col,1);
Nu = size(qcol,1);
242 maxdist = hypot(col(1,1)-col(N,1),row(1,1)-row(N,1))*ncpc;
nrbins = 45;
244 phistep = 15;
phitol = 20;
246 N_kr_mod = 3;

248 % >>> Experimental (Semi-)Variogram (anisotropic) <<<
x = col; y = row; rf = fluc; iso = 0; flag = 1;
250 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
high analysis
[gexp,nr_pairs,c_centers] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,
phitol,flag);
252 gexpmax = max(max(gexp));
%=====
254

256 %% ##### DirVar0 #####
%=====

258 % >>> Fitting I and Parameters of Anisotropic Correlation Estimation (s2
,xi1,xi2,phi,c0,v or eta1) <<<
260
% Desired Models
262 models = {'Gexp','Gaus','Sphe','Mate','Spar'};

```

```

n_models = length(models);
264
% Initial Values and Limits for optimization
266 b = [gexpmax, maxdist*2/3, 0.5, 10, gexpmax/100]; % s2, xi1, R, phi & c0
b_lb = [eps, eps, eps, -90, eps]; b_ub = [gexpmax*1.5, maxdist*1.5, 30, 90,
    gexpmax/5]; %lower and upper limits
268 bsp = [1000, maxdist*2/3, 0.5, 10, gexpmax/100]; % eta0, xi1, R, phi & c0
bsp_lb = [eps, eps, eps, -90, eps]; bsp_ub = [inf, maxdist*1.5, 30, 90, gexpmax
    /5]; %lower and upper limits
270
% Summary cells
272 model_par0 = {[b, 1.5]; b; b; [b, 1.5]; [bsp, 1]}; %initial parameters values
model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
    bounds
274 model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
    bounds
276 clear b b_lb b_ub bsp bsp_lb bsp_ub
278 % Estimation of Parameters (s2, xi1, xi2, phi, c0, v or eta1)
bmodel{n_models, 1}=[]; fval(n_models, 1)=0; tit{n_models, 1}=[];
280 iso = 00; objmod = 'NWEr_m'; flag = 1;
for i=1:n_models
282     model.function = models{i, 1};
    model.params0 = model_par0{i, 1};
284     model.paramslb = model_par_lb{i, 1};
    model.paramsub = model_par_ub{i, 1};
286
    [bmodel{i, 1}, fval(i, 1), tit{i, 1}] = ...
288     variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
end
290
% Anisotropy Estimation (#Not Necessary#)
292 R0(n_models, 1) = 0; phi0(n_models, 1) = 0; xi10(n_models, 1) = 0; xi20(
    n_models, 1) = 0;
for i=1:n_models
294     R0(i, 1) = bmodel{i, 1}(1, 3);
    phi0(i, 1) = bmodel{i, 1}(1, 4);
296     xi10(i, 1) = bmodel{i, 1}(1, 2);
    xi20(i, 1) = bmodel{i, 1}(1, 2)/R0(i, 1);
298     if R0(i, 1)>1
        R0(i, 1) = 1/R0(i, 1); bmodel{i, 1}(1, 3) = R0(i, 1);
300         xi101 = xi10(i, 1);
        xi10(i, 1) = xi20(i, 1); bmodel{i, 1}(1, 2) = xi20(i, 1);

```

```

302     xi20(i,1) = xi101;
      if phi0(i,1)>0
304         phi0(i,1) = phi0(i,1)-90;
      else
306         phi0(i,1) = phi0(i,1)+90;
      end
308     bmodel{i,1}(1,4) = phi0(i,1);
  end
310 end

312 R = mean(R0); phi = mean(phi0); xi1 = mean(xi10); xi2 = mean(xi20);

314
316 % >>> Cross Validation <<<

318 % Matrices and Cells preallocation
cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;

320
322 % Inputs definition
x = reshape(x,ny,N/ny); y = reshape(y,ny,N/ny);
rf = reshape(rf,ny,N/ny); iso = 00; d_col = 1;

324
326 % Cross Validation
for i=1:n_models
    model.function = models{i,1};
328     model.params = bmodel{i,1};
    model.r_ok = [22,4];

330
    [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] =...
332         crossval(x,y,rf,iso,d_col,model);
    cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
334 end

336 % Cross Validation Scores
table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
           'finalscore'};
338 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

340 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
342 cv_Ss = [cv_Ss, FinalScore];
cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
344 clear relMSE FinalScore

```

```

346 %Trend addition and Boxcox Inversion
cv_St(n_models,6)= 0;
348 for i=1:n_models

350     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

352     if lambda==0
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
354     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
356     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
    ;
358     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
360     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

362     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

364 end

366 % Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
368 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
370 cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore

372 %Plots
374 for i=1:n_models

376     % Stochastic Component's figures
    Z1 = reshape(rf,ny,N/ny);
378     Z2 = reshape(cv_matr{i,1}.Z,ny,N/ny);

380     figure;pcolor(Z1);%title('Sample Stochastic Component');
    view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
382     figure;pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
    view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
384     figure;scatter(Z1(:),Z2(:),'filled','d');hold on;
    dvec1 = [Z1(:);Z2(:)];

```

```

386     plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
388     axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
%title('Scatter Plot')
390     xlabel('Observed Data'); ylabel('Estimations');
figure; h = histogram(Z1(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
392     histogram(Z2(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1 0],'
FaceAlpha',0.7)
%title('Histograms of Sample Stochastic Component')
394     legend({'Original','Estimated'});
clear h

396

398 % Total Data figures
Z3 = reshape(v1,ny,N/ny);
400 Z4 = reshape(cv_matr{i,1}.Z_ibt,ny,N/ny);
% figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
402 % c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[1],'YTick',[1]);
figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s',tit{i,1}));
404 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca,'XTick'
,[1],'YTick',[1]);
figure; scatter(Z3(:),Z4(:),'filled','d'); hold on;
406 dvec2 = [Z3(:);Z4(:)];
plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5], 'r');
408 axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
%title('Scatter Plot')
410 xlabel('Observed Data')
ylabel('Estimations')
412 figure; h = histogram(Z3(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
414 histogram(Z4(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1 0],'
FaceAlpha',0.7)
%title('Histograms of Sample Data')
416 legend({'Original','Estimated'});
clear h

418

420 clear Z1 Z2 Z3 Z4 dvec1 dvec2

end

```

```

422
424 % >>> Ordinary Kriging <<<
426 % Sort models based on cross validation scores
[~,ind] = sort(table2array(cv_Stf(:,7)), 'descend');
428
% Matrices and Cells preallocation
430 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
kr_matr{N_kr_mod,1}=[];
432 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];
434
% Inputs definition
436 x = reshape(x,ny,N/ny); y = reshape(y,ny,N/ny); rf = reshape(rf,ny,N/ny)
;
xu = reshape(qcol,ny,Nu/ny); yu = reshape(qrow,ny,Nu/ny); iso = 00;
438
% Ordinary Kriging
440 for i=1:N_kr_mod
    model.function = models{ind(i),1};
442    model.params = bmodel{ind(i),1};
    model.r_ok = [22,4];
444
    [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
ordkrig(x,y,rf,xu,yu,iso,model);
446    kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
    table_r2{i,1} = table_r{ind(i),1};
448
    %Confidence Intervals (95%)
    CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
450    CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
    CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
452    UNC{i,1} = real(CI1{i,1}.uncer);
    realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
454
456 end
458
% Kriging Scores
relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
460 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
462 kr_Ssf = array2table(kr_Ss, 'VariableNames',table_h, 'RowNames',table_r2);
clear relMSE FinalScore
464

```

```

%Trend addition and Boxcox Inversion
466 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
468 for i=1:N_kr_mod

470     Z_tr{i,1} = qMx + Z{i,1};

472     if lambda==0
        Z_ibt1{i,1} = Z_tr{i,1};
474     elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
476     else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
478     end
    Z_ibt{i,1} = real(Z_ibt1{i,1});
480    kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

482    kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores

484 end

486 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
488 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
490 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore

492 %Plots
494 for i=1:N_kr_mod

496     % Stochastic Component's figures
    cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
498    cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
500    Z2=Z1;
    for j = 1:size(cc1,1)
502        Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
        Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
504    end
    figure; pcolor(Z1);%title('Original Stochastic Component');
506    view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    figure; pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s', tit{ind(i),1}));

```

```

508 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; scatter(qfluc(:), Z{i,1}(:), 'filled', 'd'); hold on;
510 dvec1 = [qfluc(:); Z{i,1}(:)];
plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)+0.5], 'r');
512 axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
%title('Scatter Plot')
514 xlabel('Observed Data')
ylabel('Estimations')
516 figure; h = histogram(qfluc(:), 16, 'FaceColor', [0 0 1], 'FaceAlpha', 0.7);
hold on
518 histogram(Z{i,1}(:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0], 'FaceAlpha', 0.7)
%title('Histograms of Stochastic Component')
520 legend({'Original', 'Estimated'});
clear h
522
% Total Data figures
524 cc3 = [col, row, v1(:); qcol, qrow, Z_ibt{i,1}];
cc4 = [col, row, zeros(size(col,1),1); qcol, qrow, UNC{i,1}];
526 Z3(max(cc3(:,2)), max(cc3(:,1)))=0; %#ok<SAGROW>
Z4=Z3;
528 for j = 1:size(cc3,1)
    Z3(cc3(j,2), cc3(j,1)) = cc3(j,3);
530 Z4(cc4(j,2), cc4(j,1)) = cc4(j,3);
end
532 figure; pcolor(Z3); view(2); shading interp; %title(sprintf('
Estimation of Original Data \n%s', tit{ind(i),1}));
c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick', [], 'YTick', []);
534 figure; scatter(qv1(:), Z_ibt{i,1}, 'filled', 'd'); hold on;
dvec2 = [qv1(:); Z_ibt{i,1}(:)];
536 plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)+0.5], 'r');
axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
538 %title('Scatter Plot')
xlabel('Observed Data')
ylabel('Estimations')
540 figure; h = histogram(qv1(:), 16, 'FaceColor', [0 0 1], 'FaceAlpha', 0.7);
hold on
542 histogram(Z_ibt{i,1}(:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0], 'FaceAlpha', 0.7)
%title('Histograms of Data')
544

```



```

546     legend({'Original', 'Estimated'});
figure; pcolor(Z4); view(2); shading interp; %title('95% Confidence
Interval');
c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
548     clear h

550     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

552 end

554 % >>>Correlation Coefficient of Indicators<<<
edges4 = (1500:4000/4:5500)/5500;
556 [~,~,ind_data4] = histcounts(qv1,edges4);
ind4_est{N_kr_mod,1}=[]; Rpearson4(N_kr_mod,1)=0; Rspearman4(N_kr_mod,1)
=0;
558 MCR4(N_kr_mod,1)=0;

560 edges16 = (1500:4000/16:5500)/5500;
[~,~,ind_data16] = histcounts(qv1,edges16);
562 ind16_est{N_kr_mod,1}=[]; Rpearson16(N_kr_mod,1)=0; Rspearman16(N_kr_mod
,1)=0;
MCR16(N_kr_mod,1)=0;

564 for i=1:N_kr_mod
566     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
568     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
570     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4, 'type', 'Spearman');
MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;

572     [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
574     ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
576     Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16, 'type', 'Spearman')
;
578     MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;

580 end

582 % Kriging Indicators Scores Table

```

```

table_h = { 'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'Rpearson', 'Rspearman'
};
584 table_h2 = { 'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'Rpearson', 'Rspearman',
, 'Rpearson4', 'Rspearman4', 'MCR4', 'Rpearson16', 'Rspearman16', 'MCR16' };
kr_iSsf = array2table(kr_Ss(:,1:end-1), 'VariableNames', table_h, 'RowNames',
, table_r2);
586 kr_iStf = array2table([kr_St(:,1:end-1), Rpearson4, Rspearman4, MCR4,
Rpearson16, Rspearman16, MCR16], 'VariableNames', table_h2, 'RowNames',
, table_r2);

%=====
588
590 %% ##### DirVar1 #####
%=====
592
% >>> Fitting I and Anisotropy Estimation <<<
594
% Desired Models
596 models = { 'Gexp', 'Gaus', 'Sphe', 'Mate', 'Spar' };
n_models = length(models);
598
% Initial Values and Limits for optimization
600 b = [gexpmax, maxdist*2/3, gexpmax/100]; % s2, xi & c0
b_lb = [eps, eps, eps]; b_ub = [gexpmax*1.5, maxdist*1.5, gexpmax/5]; %lower
and upper limits
602 bsp = [1000, maxdist*2/3, gexpmax/100]; % eta0, xi, c0
bsp_lb = [eps, eps, eps]; bsp_ub = [inf, maxdist*1.5, gexpmax/5]; %lower and
upper limits
604
% Summary cells
606 model_par0 = {[b,1.5]; b; b; [b,1.5]; [bsp,1]}; %initial parameters values
model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb,0.3]; [bsp_lb,-2+eps]}; %lower
bounds
608 model_par_ub = {[b_ub,2-eps]; b_ub; b_ub; [b_ub,3.5]; [bsp_ub,inf]}; %upper
bounds
610 clear b b_lb b_ub bsp bsp_lb bsp_ub

612 % Estimation of Anisotropy
R(n_models,1)=0; phi(n_models,1)=0; xi1(n_models,1)=0; xi2(n_models,1)=0;%
matrices preallocation
614 er1(n_models,1)=0; er2(n_models,1)=0; %matrices preallocation
objmod = 'NWEr_m'; flag = 1;
616 % nrsampl = 10; samplpc = 50;

```

```

for i=1:n_models
618     model.function = models{i,1};
        model.params0 = model_par0{i,1};
620     model.paramslb = model_par_lb{i,1};
        model.paramsub = model_par_ub{i,1};
622
        [R(i,1),phi(i,1),xi1(i,1),xi2(i,1),er1(i,1),er2(i,1)] =...
624         aniso_dvf(x,y,rf,model,objmod,ncpc,nrbins,phistep,phitol,flag);
%       [R(i,1),phi(i,1),xi1(i,1),xi2(i,1),er1(i,1),er2(i,1)] =...
626 %       dirvar_ccv_s(x,y,rf,model,objmod,N_ms,ncpc,nrsampl,samplpc,
        nrbins,phistep,phitol,flag);

628 end

630 save('dirvarl_iso.mat')

632 % >>> Fitting II and Parameters of Correlation Estimation (s2,c0,v or
        etal) <<<

634 % Initial Values and Limits for optimization
        b{n_models,1}=[]; b_lb{n_models,1}=[]; b_ub{n_models,1}=[];
636 for i = 1:n_models
        b{i,1} = [gexpmax,xi1(i,1),R(i,1),phi(i,1),gexpmax/100];
638     b_lb{i,1} = [eps,xi1(i,1),R(i,1),phi(i,1),eps];
        b_ub{i,1} = [inf,xi1(i,1),R(i,1),phi(i,1),gexpmax/5];
640 end

642 % Summary cells
        model_par0 = {[b{1,1},1.1];b{2,1};b{3,1};[b{4,1},1.5];[b{5,1},1]}; %
        initial parameters values
644     model_par_lb = {[b_lb{1,1},eps];b_lb{2,1};b_lb{3,1};[b_lb{4,1},0.3];[
        b_lb{5,1},-2+eps]};%lower bounds
        model_par_ub = {[b_ub{1,1},2];b_ub{2,1};b_ub{3,1};[b_ub{4,1},3.5];[b_ub
        {5,1},inf]}; %upper bounds
646
        % Estimation of Parameters (s2,c0,v or etal)
648     bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
        iso = 00;objmod = 'NWEr_m';flag = 1;
650     for i=1:n_models
        model.function = models{i,1};
652     model.params0 = model_par0{i,1};
        model.paramslb = model_par_lb{i,1};
654     model.paramsub = model_par_ub{i,1};

```

```

656     [bmodel{i,1}, fval(i,1), tit{i,1}] = ...
        variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
658 end

660

662 % >>> Cross Validation <<<

664 % Matrices and Cells preallocation
664 cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
664 cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;
666

668 % Inputs definition
668 x = reshape(x,ny,N/ny); y = reshape(y,ny,N/ny);
668 rf = reshape(rf,ny,N/ny); iso = 00; d_col = 1;
670

672 % Cross Validation
672 for i=1:n_models
672     model.function = models{i,1};
674     model.params = bmodel{i,1};
674     model.r_ok = [22,4];
676
676     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
678         crossval(x,y,rf,iso,d_col,model);
678     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
680 end

682 % Cross Validation Scores
682 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
        'finalscore'};
684 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

686 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
686 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
688 cv_Ss = [cv_Ss, FinalScore];
688 cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
690 clear relMSE FinalScore

692 %Trend addition and Boxcox Inversion
692 cv_St(n_models,6)= 0;
694 for i=1:n_models

696     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

698     if lambda==00

```

```

    cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
700 elseif lambda==0
    cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
702 else
    cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
;
704 end
cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
706 cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

708 cv_St(i,:) = correstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

710 end

712 % Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
714 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
716 cv_Stf = array2table(cv_St, 'VariableNames',table_h, 'RowNames',table_r);
clear relMSE FinalScore

718 %Plots
720 for i=1:n_models

722 % Stochastic Component's figures
Z1 = reshape(rf,ny,N/ny);
724 Z2 = reshape(cv_matr{i,1}.Z,ny,N/ny);

726 figure;pcolor(Z1);%title('Sample Stochastic Component');
view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
728 figure;pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s',tit{i,1}));
view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
730 figure;scatter(Z1(:),Z2(:),'filled','d');hold on;
dvec1 = [Z1(:);Z2(:)];
732 plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
734 %title('Scatter Plot')
xlabel('Observed Data'); ylabel('Estimations');
736 figure; h = histogram(Z1(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
738 histogram(Z2(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1 0]','
FaceAlpha',0.7)

```

```

%title('Histograms of Sample Stochastic Component')
740 legend({'Original', 'Estimated'});
clear h

742

744 % Total Data figures
Z3 = reshape(v1,ny,N/ny);
746 Z4 = reshape(cv_matr{i,1}.Z_ibt,ny,N/ny);
% figure;pcolor(Z3);view(2);shading interp; %title('Original Sample
');
748 % c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'
XTick',[ ],'YTick',[ ]);
figure;pcolor(Z4);view(2);shading interp; %title(sprintf('Estimation
of Original Sample \n%s',tit{i,1}));
750 c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick'
,[ ],'YTick',[ ]);
figure;scatter(Z3(:),Z4(:),'filled','d');hold on;
752 dvec2 = [Z3(:);Z4(:)];
plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
754 axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
%title('Scatter Plot')
756 xlabel('Observed Data')
ylabel('Estimations')
758 figure;h = histogram(Z3(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
760 histogram(Z4(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1 0],'
FaceAlpha',0.7)
%title('Histograms of Sample Data')
762 legend({'Original', 'Estimated'});
clear h

764

clear Z1 Z2 Z3 Z4 dvec1 dvec2

766
end

768

770
% >>> Ordinary Kriging <<<
772
% Sort models based on cross validation scores
774 [~,ind] = sort(table2array(cv_Stf(:,7)),'descend');
776
% Matrices and Cells preallocation

```

```

Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
778 kr_matr{N_kr_mod,1}=[];
kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
780 CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];

782 % Inputs definition
x = reshape(x,ny,N/ny); y = reshape(y,ny,N/ny); rf = reshape(rf,ny,N/ny)
;
784 xu = reshape(qcol,ny,Nu/ny); yu = reshape(qrow,ny,Nu/ny); iso = 00;

786 % Ordinary Kriging
for i=1:N_kr_mod
788     model.function = models{ind(i),1};
model.params = bmodel{ind(i),1};
790     model.r_ok = [22,4];

792     [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
ordkrig(x,y,rf,xu,yu,iso,model);
794     kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
table_r2{i,1} = table_r{ind(i),1};
796
%Confidence Intervals (95%)
798     CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
800     CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
UNC{i,1} = real(CI1{i,1}.uncer);
802     realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
end
804
% Kriging Scores
806 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
808 kr_Ss = [kr_Ss, FinalScore];
kr_Ssf = array2table(kr_Ss, 'VariableNames',table_h, 'RowNames', table_r2);
810 clear relMSE FinalScore

812 %Trend addition and Boxcox Inversion
kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
814 Z_ibt1{N_kr_mod,1} = 0;Z_ibt{N_kr_mod,1} = 0;
for i=1:N_kr_mod
816
Z_tr{i,1} = qMx + Z{i,1};

818
if lambda==00

```

```

820     Z_ibt1{i,1} = Z_tr{i,1};
elseif lambda==0
822     Z_ibt1{i,1} = exp(Z_tr{i,1});
else
824     Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
end
826 Z_ibt{i,1} = real(Z_ibt1{i,1});
kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>
828
kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
830
end
832
% Total Kriging Scores
834 relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
836 kr_St = [kr_St, FinalScore];
kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
838 clear relMSE FinalScore
840
%Plots
for i=1:N_kr_mod
842
    % Stochastic Component's figures
844     cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
     cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
846     Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
     Z2=Z1;
848     for j = 1:size(cc1,1)
         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
850         Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
     end
852     figure;pcolor(Z1);%title('Original Stochastic Component');
     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
854     figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s', tit{ind(i),1}));
     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
856     figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
     dvec1 = [qfluc(:);Z{i,1}(:)];
858     plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
     axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
860     %title('Scatter Plot')
     xlabel('Observed Data')

```



```

862     ylabel ('Estimations')
figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha',
,0.7);
864     hold on
histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
866     %title('Histograms of Stochastic Component')
legend({'Original','Estimated'});
868     clear h

870     % Total Data figures
cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
872     cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
874     Z4=Z3;
for j = 1:size(cc3,1)
876         Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
878     end
figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
880     c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick',
,[],'YTick',[]);
figure;scatter(qv1(:),Z_ibt{i,1},'filled','d');hold on;
882     dvec2 = [qv1(:);Z_ibt{i,1}(:)];
plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
884     axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
%title('Scatter Plot')
886     xlabel ('Observed Data')
ylabel ('Estimations')
888     figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
hold on
890     histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
%title('Histograms of Data')
892     legend({'Original','Estimated'});
figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
894     c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[],'YTick',[]);
clear h
896

clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

```

```

898 end
900
901 %% >>>Correlation Coefficient of Indicators<<<
902 edges4 = (1500:4000/4:5500)/5500;
[~,~,ind_data4] = histcounts(qv1,edges4);
904 ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
    =0;
MCR4(N_kr_mod,1)=0;
906
edges16 = (1500:4000/16:5500)/5500;
908 [~,~,ind_data16] = histcounts(qv1,edges16);
ind16_est{N_kr_mod,1}=[];Rpearson16(N_kr_mod,1)=0;Rspearman16(N_kr_mod
    ,1)=0;
910 MCR16(N_kr_mod,1)=0;

912 for i=1:N_kr_mod
    [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
914 ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
916 Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
918 MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;

    [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
922 ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
924 Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
    ;
MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
926 end

928 % Kriging Indicators Scores Table
table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    };
930 table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    , 'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
    ,};
kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
    ,table_r2);
932 kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
    Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
    table_r2);

```

```

934 %=====
936 %% ##### DirVar2 #####
938 %=====
940 load('dirvar1_iso.mat')
942 % >>> Rescaling and Rotation of the random field
944 % (Inverse) Transformation Matrix
A{n_models,1} = []; c_tr{n_models,1} = []; x_tr{n_models,1} = []; y_tr{
    n_models,1} = [];
    qc_tr{n_models,1} = []; qx_tr{n_models,1} = []; qy_tr{n_models,1} = [];
946 R_tr(n_models,1)=0; phi_tr(n_models,1)=0; xi1_tr(n_models,1)=0; xi2_tr(
    n_models,1)=0;
948 gexp_tr{n_models,1}=[]; nr_pairs_tr{n_models,1}=[]; c_centers_tr{
    n_models,1}=[];
950 for i=1:n_models
A{i,1} = [cos(phi(i,1)*pi/180)/xi1(i,1), sin(phi(i,1)*pi/180)/xi1(i,1);
952 -sin(phi(i,1)*pi/180)/xi2(i,1), cos(phi(i,1)*pi/180)/xi2(i,1)];
954 % Rescale and Rotate Coordinations
c_tr{i,1} = A{i,1}*[c{1,1}';c{1,2}']; %transformed coordinations of
    known points
956 x_tr{i,1} = reshape(c_tr{i,1}(1,:),ny,N/ny);
y_tr{i,1} = reshape(c_tr{i,1}(2,:),ny,N/ny);
958 qc_tr{i,1} = A{i,1}*[qc{1,1}';qc{1,2}']; %transformed coordinations of
    unknown points
qx_tr{i,1} = reshape(qc_tr{i,1}(1,:),ny,Nu/ny);
960 qy_tr{i,1} = reshape(qc_tr{i,1}(2,:),ny,Nu/ny);
962 % >>> Check isotropy <<<
964 % Experimental Variogram (anisotropic)
x = x_tr{i,1}; y = y_tr{i,1}; rf = fluc; iso = 0;
966 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
    high analysis
[~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,phitol,2);
968 % Estimation of New Anisotropy

```

```

970 model.function = models{i,1};
model.params0 = model_par0{i,1};
972 model.paramslb = model_par_lb{i,1};
model.paramsub = model_par_ub{i,1};
974 [R_tr(i,1), phi_tr(i,1), xi1_tr(i,1), xi2_tr(i,1), ~, ~] = ...
    aniso_dvf(c_tr{i,1}(1,:), c_tr{i,1}(2,:), rf, model, 'NWEr_m', 1.5*ncpc,
    nrbins, phistep, phitol, 0);
976
end
978

% >>> Fitting II and Parameters of Correlation Estimation (s2, c0, v or
    etal) <<<

982 % Experimental (Semi-)Variogram (isotropic)
gexp2{n_models,1}=[]; nr_pairs2{n_models,1}=[]; c_centers2{n_models
    ,1}=[];
984 gexpmax2(n_models,1)=0; maxdist2(n_models,1)=0;
for i=1:n_models
986 x = x_tr{i,1}; y = y_tr{i,1}; rf = fluc; iso = 1; flag = 1;
[~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
    high analysis
988 [gexp2{i,1}, nr_pairs2{i,1}, c_centers2{i,1}] = expvar(x,y,rf,iso,ncpc,
    nrbins, phistep, phitol, flag);
gexpmax2(i,1) = max(max(gexp2{i,1}));
990 maxdist2(i,1) = hypot(c_tr{i,1}(1,1)-c_tr{i,1}(1,N), c_tr{i,1}(2,1)-c_tr{
    i,1}(2,N))*ncpc;
end
992

% Initial Values and Limits for optimization
994 b{n_models,1}=[]; b_lb{n_models,1}=[]; b_ub{n_models,1}=[];
for i = 1:n_models
996     b{i,1} = [gexpmax2(i,1), maxdist2(i,1)*2/3, gexpmax2(i,1)/100];
    b_lb{i,1} = [eps, eps, eps];
998     b_ub{i,1} = [inf, maxdist2(i,1)*1.5, gexpmax2(i,1)/5];
end
1000

% Summary cells
1002 model_par0 = {[b{1,1}, 1.1]; b{2,1}; b{3,1}; [b{4,1}, 1.5]; [b{5,1}, 1]}; %
    initial parameters values
model_par_lb = {[b_lb{1,1}, eps]; b_lb{2,1}; b_lb{3,1}; [b_lb{4,1}, 0.3]; [
    b_lb{5,1}, -2+eps]}; %lower bounds
1004 model_par_ub = {[b_ub{1,1}, 2]; b_ub{2,1}; b_ub{3,1}; [b_ub{4,1}, 3.5]; [b_ub
    {5,1}, inf]}; %upper bounds

```

```

1006 % Estimation of Parameters (s2,c0,v or etal)
      bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
1008 iso = 1;objmod = 'NWEr_m';flag = 1;
      for i=1:n_models
1010         model.function = models{i,1};
            model.params0 = model_par0{i,1};
1012         model.paramslb = model_par_lb{i,1};
            model.paramsub = model_par_ub{i,1};
1014
            [bmodel{i,1},fval(i,1),tit{i,1}] =...
1016             variogramfit(gexp2{i,1},nr_pairs2{i,1},c_centers2{i,1},iso,model,
                objmod,flag);
1018     end

1020 % >>> Cross Validation <<<

1022 % Matrices and Cells preallocation
      cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
1024 cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;

1026 % Inputs definition
      x = reshape(x,ny,N/ny); y = reshape(y,ny,N/ny);
1028 rf = reshape(rf,ny,N/ny); iso = 1; d_col = 1;

1030 % Cross Validation
      for i=1:n_models
1032         x = reshape(x_tr{i,1},ny,N/ny); y = reshape(y_tr{i,1},ny,N/ny);
            model.function = models{i,1};
1034         model.params = bmodel{i,1};
            r_ok1 = pdist2([x(1,1),y(1,1)],[x(1,2),y(1,2)])*2.0;
1036         r_ok2 = pdist2([x(1,1),y(1,1)],[x(2,1),y(2,1)])*4.0;
            model.r_ok = [r_ok1, r_ok2];
1038
            [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] =...
1040             crossval(x,y,rf,iso,d_col,model);
            cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
1042     end

1044 % Cross Validation Scores
      table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
                'finalscore'};
1046 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

```

```

1048 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
1050 cv_Ss = [cv_Ss, FinalScore];
cv_Ssf = array2table(cv_Ss, 'VariableNames', table_h, 'RowNames', table_r);
1052 clear relMSE FinalScore

1054 %Trend addition and Boxcox Inversion
cv_St(n_models,6)= 0;
1056 for i=1:n_models

1058     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1060     if lambda==0
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1062     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1064     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
        ;
1066     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
1068     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

1070     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1072 end

1074 % Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
1076 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
1078 cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore

1080 %Plots
1082 for i=1:n_models

1084     % Stochastic Component's figures
    Z1 = reshape(rf,ny,N/ny);
1086     Z2 = reshape(cv_matr{i,1}.Z,ny,N/ny);

1088     figure;pcolor(Z1);%title('Sample Stochastic Component');
    view(2);shading interp; colorbar;set(gca, 'XTick',[], 'YTick',[]);

```

```

1090     figure; pcolor(Z2); %title(sprintf(' Estimation of Sample Stochastic
Component \n%s ', tit{i,1}));
1092     view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; scatter(Z1(:), Z2(:), 'filled', 'd'); hold on;
dvec1 = [Z1(:); Z2(:)];
1094     plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)
+0.5], 'r');
axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
1096     %title(' Scatter Plot ')
xlabel('Observed Data'); ylabel('Estimations');
1098     figure; h = histogram(Z1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
hold on
1100     histogram(Z2(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], '
FaceAlpha', 0.7)
%title('Histograms of Sample Stochastic Component')
1102     legend({'Original', 'Estimated'});
clear h
1104
1106     % Total Data figures
Z3 = reshape(v1, ny, N/ny);
1108     Z4 = reshape(cv_matr{i,1}.Z_ibt, ny, N/ny);
%     figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
1110 %     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
figure; pcolor(Z4); view(2); shading interp; %title(sprintf(' Estimation
of Original Sample \n%s ', tit{i,1}));
1112     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
, [], 'YTick', []);
figure; scatter(Z3(:), Z4(:), 'filled', 'd'); hold on;
1114     dvec2 = [Z3(:); Z4(:)];
plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
1116     axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
%title(' Scatter Plot ')
xlabel('Observed Data')
1118     ylabel('Estimations')
figure; h = histogram(Z3(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
1120     hold on
histogram(Z4(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], '
FaceAlpha', 0.7)
%title('Histograms of Sample Data')
1122     legend({'Original', 'Estimated'});
1124

```

```

clear h
1126
clear Z1 Z2 Z3 Z4 dvec1 dvec2
1128
end
1130
1132 % >>> Ordinary Kriging <<<
1134 % Sort models based on cross validation scores
[~,ind] = sort(table2array(cv_Stf(:,7)), 'descend');
1136
% Matrices and Cells preallocation
1138 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
kr_matr{N_kr_mod,1}=[];
1140 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];
1142
% Inputs definition
1144 rf = reshape(rf,ny,N/ny); iso = 1;
1146 % Ordinary Kriging
for i=1:N_kr_mod
1148 x = reshape(x_tr{i,1},ny,N/ny); y = reshape(y_tr{i,1},ny,N/ny);
xu = qx_tr{i,1}; yu = qy_tr{i,1};
1150 model.function = models{ind(i),1};
model.params = bmodel{ind(i),1};
1152 r_ok1 = pdist2([x(1,1),y(1,1)],[x(1,2),y(1,2)])*2.0;
r_ok2 = pdist2([x(1,1),y(1,1)],[x(2,1),y(2,1)])*4.0;
1154 model.r_ok = [r_ok1, r_ok2];

[Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
ordkrig(x,y,rf,xu,yu,iso,model);
1158 kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
table_r2{i,1} = table_r{ind(i),1};
1160

%Confidence Intervals (95%)
1162 CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
1164 CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
UNC{i,1} = real(CI1{i,1}.uncer);
1166 realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
end
1168

```



```

% Kriging Scores
1170 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
    FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
1172 kr_Ss = [kr_Ss, FinalScore];
    kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
1174 clear relMSE FinalScore

%Trend addition and Boxcox Inversion
1176 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
1178 Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
    for i=1:N_kr_mod
1180
        Z_tr{i,1} = qMx + Z{i,1};
1182
        if lambda==0
1184            Z_ibt1{i,1} = Z_tr{i,1};
        elseif lambda==0
1186            Z_ibt1{i,1} = exp(Z_tr{i,1});
        else
1188            Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
        end
1190 Z_ibt{i,1} = real(Z_ibt1{i,1});
        kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>
1192
        kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
1194
    end
1196
% Total Kriging Scores
1198 relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
    FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
1200 kr_St = [kr_St, FinalScore];
    kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
1202 clear relMSE FinalScore

%Plots
1204 for i=1:N_kr_mod
1206
    % Stochastic Component's figures
1208 cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
        cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
1210 Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
        Z2=Z1;
1212 for j = 1:size(cc1,1)

```

```

1214     Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
1215     Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1216 end
1217 figure; pcolor(Z1); %title('Original Stochastic Component');
1218 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
1219 figure; pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s', tit{ind(i),1}));
1220 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
1221 figure; scatter(qfluc(:), Z{i,1}(:), 'filled', 'd'); hold on;
1222 dvec1 = [qfluc(:); Z{i,1}(:)];
1223 plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)
+0.5], 'r');
1224 axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
1225 %title('Scatter Plot')
1226 xlabel('Observed Data')
1227 ylabel('Estimations')
1228 figure; h = histogram(qfluc(:), 16, 'FaceColor', [0 0 1], 'FaceAlpha'
, 0.7);
1229 hold on
1230 histogram(Z{i,1}(:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0], '
FaceAlpha', 0.7)
1231 %title('Histograms of Stochastic Component')
1232 legend({'Original', 'Estimated'});
1233 clear h

1234 % Total Data figures
1235 cc3 = [col, row, v1(:); qcol, qrow, Z_ibt{i,1}];
1236 cc4 = [col, row, zeros(size(col,1),1); qcol, qrow, UNC{i,1}];
1237 Z3(max(cc3(:,2)), max(cc3(:,1)))=0; %#ok<SAGROW>
1238 Z4=Z3;
1239 for j = 1:size(cc3,1)
1240     Z3(cc3(j,2), cc3(j,1)) = cc3(j,3);
1241     Z4(cc4(j,2), cc4(j,1)) = cc4(j,3);
1242 end
1243 figure; pcolor(Z3); view(2); shading interp; %title(sprintf('
Estimation of Original Data \n%s', tit{ind(i),1}));
1244 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
, [], 'YTick', []);
1245 figure; scatter(qv1(:), Z_ibt{i,1}, 'filled', 'd'); hold on;
1246 dvec2 = [qv1(:); Z_ibt{i,1}(:)];
1247 plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
1248 axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
1249 %title('Scatter Plot')

```

```

1250     xlabel ( 'Observed Data' )
1251     ylabel ( 'Estimations' )
1252     figure ; h = histogram ( qv1 (:), 16, 'FaceColor', [0 0 1], 'FaceAlpha', 0.7 );
1253     hold on
1254     histogram ( Z_ibt { i, 1 } (:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0],
1255     'FaceAlpha', 0.7)
1256     %title ( 'Histograms of Data' )
1257     legend ( { 'Original', 'Estimated' } );
1258     figure ; pcolor ( Z4 ); view ( 2 ); shading interp ; %title ( '95% Confidence
1259     Interval' );
1260     c1 = colorbar ; c1.Label.String = 'Normalised Velocity' ; set ( gca, '
1261     XTick', [], 'YTick', [] );
1262     clear h
1263
1264     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2
1265
1266 end
1267
1268 % >>> Correlation Coefficient of Indicators <<<
1269 edges4 = (1500:4000/4:5500)/5500;
1270 [~,~,ind_data4] = histcounts ( qv1, edges4 );
1271 ind4_est { N_kr_mod, 1 } = []; Rpearson4 ( N_kr_mod, 1 ) = 0; Rspearman4 ( N_kr_mod, 1 )
1272     = 0;
1273 MCR4 ( N_kr_mod, 1 ) = 0;
1274
1275 edges16 = (1500:4000/16:5500)/5500;
1276 [~,~,ind_data16] = histcounts ( qv1, edges16 );
1277 ind16_est { N_kr_mod, 1 } = []; Rpearson16 ( N_kr_mod, 1 ) = 0; Rspearman16 ( N_kr_mod
1278     , 1 ) = 0;
1279 MCR16 ( N_kr_mod, 1 ) = 0;
1280
1281 for i = 1 : N_kr_mod
1282     [~,~,ind4_est { i, 1 } ] = histcounts ( Z_ibt { i, 1 }, edges4 );
1283     ind4_est { i, 1 } ( Z_ibt { i, 1 } <= edges4 ( 1 ) ) = 1;
1284     ind4_est { i, 1 } ( Z_ibt { i, 1 } >= edges4 ( end ) ) = length ( edges4 ) - 1;
1285     Rpearson4 ( i, 1 ) = corr ( ind4_est { i, 1 }, ind_data4 );
1286     Rspearman4 ( i, 1 ) = corr ( ind4_est { i, 1 }, ind_data4, 'type', 'Spearman' );
1287     MCR4 ( i, 1 ) = sum ( ind4_est { i, 1 } ~= ind_data4 ) / Nu;
1288
1289     [~,~,ind16_est { i, 1 } ] = histcounts ( Z_ibt { i, 1 }, edges16 );
1290     ind16_est { i, 1 } ( Z_ibt { i, 1 } <= edges16 ( 1 ) ) = 1;
1291     ind16_est { i, 1 } ( Z_ibt { i, 1 } >= edges16 ( end ) ) = length ( edges16 ) - 1;
1292     Rpearson16 ( i, 1 ) = corr ( ind16_est { i, 1 }, ind_data16 );

```

```

    Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
;
1290    MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;

1292 end

1294 % Kriging Indicators Scores Table
table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
};
1296 table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
,'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
,};
kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
,'table_r2');
1298 kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
table_r2);

%=====

1300

1302 %% ##### CHI1 #####
%=====

1304

% >>> Anisotropy estimation with CHI <<<
1306
x1 = reshape(x,ny,N/ny);y1 = reshape(y,ny,N/ny);rf1 = reshape(rf,ny,N/ny
);
1308 [R,phi] = aniso_cc_grid(x1,y1,rf1);
if R>1
1310     R = 1/R;
    if phi>0
1312         phi = phi-90;
    elseif phi<0
1314         phi = phi+90;
    end
1316 end

1318 save('chi1_iso.mat')

1320

% >>> Fitting II and Parameters of Correlation Estimation (s2,c0,v or
    etal) <<<
1322

% Desired Models

```

```

1324 models = { 'Gexp', 'Gaus', 'Sphe', 'Mate', 'Spar' };
      n_models = length(models);
1326
      % Initial Values and Limits for optimization
1328 b = [gexpmax, maxdist*1/3, R, phi, gexpmax/100]; % s2, xi & c0
      b_lb = [eps, eps, R, phi, eps]; b_ub = [gexpmax*1.5, maxdist, R, phi, gexpmax
        /5]; %lower and upper limits
1330 bsp = [1000, maxdist*1/3, R, phi, gexpmax/100]; % eta0, xi, c0
      bsp_lb = [eps, eps, R, phi, eps]; bsp_ub = [inf, maxdist, R, phi, gexpmax/5]; %
        lower and upper limits
1332
      % Summary cells
1334 model_par0 = {[b, 1.5]; b; b; [b, 1.5]; [bsp, 1]}; %initial parameters values
      model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
        bounds
1336 model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
        bounds
1338 clear b b_lb b_ub bsp bsp_lb bsp_ub
1340
      % Estimation of Parameters (s2, xi1, c0, v or etal)
      bmodel{n_models, 1}=[]; fval(n_models, 1)=0; tit{n_models, 1}=[];
1342 iso = 00; objmod = 'NWEr_m'; flag = 1;
      for i=1:n_models
1344         model.function = models{i, 1};
            model.params0 = model_par0{i, 1};
1346         model.paramslb = model_par_lb{i, 1};
            model.paramsub = model_par_ub{i, 1};
1348
            [bmodel{i, 1}, fval(i, 1), tit{i, 1}] = ...
1350             variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
      end
1352
1354 % >>> Cross Validation <<<
1356
      % Matrices and Cells preallocation
      cv_scores{n_models, 1}=[]; cv_checks{n_models, 1}=[];
1358 cv_matr{n_models, 1}=[]; cv_Ss(n_models, 6)= 0;
1360
      % Inputs definition
      x = reshape(x, ny, N/ny); y = reshape(y, ny, N/ny);
1362 rf = reshape(rf, ny, N/ny); iso = 00; d_col = 1;

```

```

1364 % Cross Validation
1365 for i=1:n_models
1366     model.function = models{i,1};
1367     model.params = bmodel{i,1};
1368     model.r_ok = [22,4];

1370     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
        crossval(x,y,rf,iso,d_col,model);
1372     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
end

1374 % Cross Validation Scores
1376 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
    'finalscore'};
table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

1378 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
1380 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
cv_Ss = [cv_Ss, FinalScore];
1382 cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
clear relMSE FinalScore

1384 %Trend addition and Boxcox Inversion
1386 cv_St(n_models,6)= 0;
1387 for i=1:n_models
1388
1389     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1390
1391     if lambda==00
1392         cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1393     elseif lambda==0
1394         cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1395     else
1396         cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
1397     ;
1398     end
1399     cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
1400     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

1402     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1404 end

% Total Cross Validation Scores

```

```

1406 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
1408 cv_St = [cv_St, FinalScore];
cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
1410 clear relMSE FinalScore

1412 %Plots
for i=1:n_models
1414
    % Stochastic Component's figures
1416 Z1 = reshape(rf,ny,N/ny);
Z2 = reshape(cv_matr{i,1}.Z,ny,N/ny);
1418
    figure; pcolor(Z1); %title('Sample Stochastic Component');
1420 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; pcolor(Z2); %title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
1422 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; scatter(Z1(:), Z2(:), 'filled', 'd'); hold on;
1424 dvec1 = [Z1(:); Z2(:)];
plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)
+0.5], 'r');
1426 axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
%title('Scatter Plot')
1428 xlabel('Observed Data'); ylabel('Estimations');
figure; h = histogram(Z1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
1430 hold on
histogram(Z2(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], 'FaceAlpha', 0.7)
1432 %title('Histograms of Sample Stochastic Component')
legend({'Original', 'Estimated'});
1434 clear h

1436
    % Total Data figures
1438 Z3 = reshape(v1,ny,N/ny);
Z4 = reshape(cv_matr{i,1}.Z_ibt,ny,N/ny);
1440 % figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
% c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
1442 figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s', tit{i,1}));

```

```

c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick',
[], 'YTick', []);
1444 figure; scatter(Z3(:), Z4(:), 'filled', 'd'); hold on;
dvec2 = [Z3(:); Z4(:)];
1446 plot([min(dvec2) - 0.5, max(dvec2) + 0.5], [min(dvec2) - 0.5, max(dvec2)
+ 0.5], 'r');
axis([min(dvec2) - 0.5, max(dvec2) + 0.5, min(dvec2) - 0.5, max(dvec2) + 0.5])
1448 %title('Scatter Plot')
xlabel('Observed Data')
1450 ylabel('Estimations')
figure; h = histogram(Z3(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
1452 hold on
histogram(Z4(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], '
FaceAlpha', 0.7)
1454 %title('Histograms of Sample Data')
legend({'Original', 'Estimated'});
1456 clear h

clear Z1 Z2 Z3 Z4 dvec1 dvec2

1460 end

1462

1464 % >>> Ordinary Kriging <<<

1466 % Sort models based on cross validation scores
[~, ind] = sort(table2array(cv_Stf(:, 7)), 'descend');
1468

% Matrices and Cells preallocation
1470 Z{N_kr_mod, 1} = []; Z_error{N_kr_mod, 1} = []; kr_checks{N_kr_mod, 1} = [];
kr_matr{N_kr_mod, 1} = [];
1472 kr_Ss(N_kr_mod, 6) = 0; table_r2{N_kr_mod, 1} = [];
CI1{N_kr_mod, 1} = []; UNC{N_kr_mod, 1} = [];
1474

% Inputs definition
1476 x = reshape(x, ny, N/ny); y = reshape(y, ny, N/ny); rf = reshape(rf, ny, N/ny)
;
xu = reshape(qcol, ny, Nu/ny); yu = reshape(qrow, ny, Nu/ny); iso = 00;
1478

% Ordinary Kriging
1480 for i = 1:N_kr_mod
model.function = models{ind(i), 1};
1482 model.params = bmodel{ind(i), 1};

```



```

1484     model.r_ok = [22,4];

1486     [Z{i,1}, Z_error{i,1}, kr_checks{i,1}, kr_matr{i,1}] = ...
1488     ordkrig(x,y,rf,xu,yu,iso,model);
kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
1490     table_r2{i,1} = table_r{ind(i),1};

1492     %Confidence Intervals (95%)
CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
1494     CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
1496     UNC{i,1} = real(CI1{i,1}.uncer);
realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
1498 end

1498 % Kriging Scores
relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
1500 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
1502 kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore

1504 %Trend addition and Boxcox Inversion
1506 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
1508 for i=1:N_kr_mod

1510     Z_tr{i,1} = qMx + Z{i,1};

1512     if lambda==0
        Z_ibt1{i,1} = Z_tr{i,1};
1514     elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
1516     else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
1518     end
    Z_ibt{i,1} = real(Z_ibt1{i,1});
1520     kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

1522     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores

1524 end

1526 % Total Kriging Scores

```

```

relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
1528 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
1530 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore
1532
%Plots
1534 for i=1:N_kr_mod

% Stochastic Component's figures
cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
1538 cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
1540 Z2=Z1;
for j = 1:size(cc1,1)
1542     Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
    Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1544 end
figure;pcolor(Z1);%title('Original Stochastic Component');
1546 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s',tit{ind(i),1}));
1548 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
1550 dvec1 = [qfluc(:);Z{i,1}(:)];
plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
1552 axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
%title('Scatter Plot')
1554 xlabel('Observed Data')
ylabel('Estimations')
1556 figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha'
,0.7);
hold on
1558 histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
%title('Histograms of Stochastic Component')
1560 legend({'Original','Estimated'});
clear h
1562
% Total Data figures
1564 cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
1566 Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>

```

```

Z4=Z3;
1568 for j = 1:size(cc3,1)
        Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
1570        Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
        end
1572        figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
        c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick'
,[], 'YTick',[]);
1574        figure;scatter(qv1(:),Z_ibt{i,1},'filled','d');hold on;
        dvec2 = [qv1(:);Z_ibt{i,1}(:)];
1576        plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
        axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
1578        %title('Scatter Plot')
        xlabel('Observed Data')
1580        ylabel('Estimations')
        figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
1582        hold on
        histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
1584        %title('Histograms of Data')
        legend({'Original','Estimated'});
1586        figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
        c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[], 'YTick',[]);
1588        clear h

        clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

1590
1592 end

1594 % >>>Correlation Coefficient of Indicators<<<
        edges4 = (1500:4000/4:5500)/5500;
1596 [~,~,ind_data4] = histcounts(qv1,edges4);
        ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
=0;
1598 MCR4(N_kr_mod,1)=0;

1600 edges16 = (1500:4000/16:5500)/5500;
        [~,~,ind_data16] = histcounts(qv1,edges16);
1602 ind16_est{N_kr_mod,1}=[];Rpearson16(N_kr_mod,1)=0;Rspearman16(N_kr_mod
,1)=0;

```

```

MCR16(N_kr_mod,1)=0;
1604
for i=1:N_kr_mod
1606     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
    ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
1608     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
    Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
1610     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
    MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;
1612
    [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
    ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
1614     ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
    Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
1616     Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman');
    ;
    MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
1620
end
1622
% Kriging Indicators Scores Table
1624 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    };
    table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman',
        'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
        ,};
1626 kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
    ',table_r2);
    kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
        Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
        table_r2);
1628 %=====
1630
1632 %% ##### CHI2 #####
    %=====
1634 load('chi1_iso.mat')
1636 % >>> Fitting II and Estimation of xi1 & xi2 <<<
1638 xi1 = R;xi2 = 1;

```

```

1640 % Desired Models
models = { 'Gexp'; 'Gaus'; 'Sphe'; 'Mate'; 'Spar' };
1642 n_models = length(models);

1644 % >>> Rescaling and Rotation of the random field

1646 % (Inverse) Transformation Matrix
A = [cos(phi*pi/180)/xi1, sin(phi*pi/180)/xi1;
1648 -sin(phi*pi/180)/xi2, cos(phi*pi/180)/xi2];

1650 % Rescale and Rotate Coordinations
c_tr = A*[c{1,1}';c{1,2}']; %transformed coordinations
1652 x_tr = reshape(c_tr(1,:),ny,N/ny);
y_tr = reshape(c_tr(2,:),ny,N/ny);
1654 qc_tr = A*[qc{1,1}';qc{1,2}']; %transformed coordinations of unknown
points
qx_tr = reshape(qc_tr(1,:),ny,Nu/ny);
1656 qy_tr = reshape(qc_tr(2,:),ny,Nu/ny);

1658 % >>> Check isotropy <<<

1660 % Experimental Variogram (anisotropic)
x = x_tr; y = y_tr; rf = fluc; iso = 0;
1662 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
high analysis
%[gexp_tr, nr_pairs_tr, c_centers_tr] = expvar(x,y,rf,iso,ncpc,nrbins,
phistep,phitol,2);

1664 %Initial Values and Limits for optimization
1666 b = [gexpmax,maxdist*1/3,gexpmax/100]; % s2,xi & c0
b_lb = [eps,eps,eps]; b_ub = [gexpmax*1.5,maxdist,gexpmax/5]; %lower and
upper limits
1668 bsp = [1000,maxdist*1/3,gexpmax/100]; % eta0, xi, c0
bsp_lb = [eps,eps,eps]; bsp_ub = [inf,maxdist,gexpmax/5]; %lower and
upper limits

1670 %Summary cells
1672 model_par0 = {[b,1.5];b;b;[b,1.5];[bsp,1]}; %initial parameters values
model_par_lb = {[b_lb,eps];b_lb;b_lb;[b_lb,0.3];[bsp_lb,-2+eps]}; %lower
bounds
1674 model_par_ub = {[b_ub,2-eps];b_ub;b_ub;[b_ub,3.5];[bsp_ub,inf]}; %upper
bounds

1676 clear b b_lb b_ub bsp bsp_lb bsp_ub

```

```

1678 % Estimation of New Anisotropy
R_tr(n_models,1)=0; phi_tr(n_models,1)=0; xi1_tr(n_models,1)=0; xi2_tr(
    n_models,1)=0;
1680 for i=1:n_models
    model.function = models{i,1};
1682 model.params0 = model_par0{i,1};
    model.paramslb = model_par_lb{i,1};
1684 model.paramsub = model_par_ub{i,1};
    [R_tr(i,1), phi_tr(i,1), xi1_tr(i,1), xi2_tr(i,1), ~, ~] = ...
1686 aniso_dvf(c_tr(1,:), c_tr(2,:), rf, model, 'NWEr_m', 20, 0.4, nrbins,
    phistep, phitol, 0);
end
1688
1690 % >>> Fitting III and Parameters of Correlation Estimation (s2, c0, v or
    etal) <<<
1692
1694 % Experimental (Semi-)Variogram (isotropic)
x = x_tr; y = y_tr; rf = fluc; iso = 1; flag = 1;
[~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,flag); %exper. variogr.
    of high analysis
1696 [gexp2, nr_pairs2, c_centers2] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,
    phitol,flag);
gexpmax2 = max(max(gexp2));
1698
1699 % Initial Values and Limits for optimization
1700 maxdist2 = hypot(c_tr(1,1)-c_tr(1,N), c_tr(2,1)-c_tr(2,N))*ncpc;
1702 b = [gexpmax2, maxdist2*2/3, gexpmax2/100]; % s2, xi & c0
    b_lb = [eps, eps, eps]; b_ub = [gexpmax2*1.5, maxdist2*1.5, gexpmax2/5]; %
    lower and upper limits
1704 bsp = [1000, maxdist2*2/3, gexpmax2/100]; % eta0, xi, c0
    bsp_lb = [eps, eps, eps]; bsp_ub = [inf, maxdist2*1.5, gexpmax2/5]; %lower
    and upper limits
1706
1707 % Summary cells
1708 model_par0 = {[b,1.5]; b; b; [b,1.5]; [bsp,1]}; %initial parameters values
    model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
    bounds
1710 model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
    bounds

```

```

1712 clear b b_lb b_ub bsp bsp_lb bsp_ub

1714 % Estimation of Parameters (s2,c0,v or etal)
bmodel2{n_models,1}=[]; fval2(n_models,1)=0; tit2{n_models,1}=[];
1716 iso = 1;objmod = 'NWEr_m';flag = 1;
for i=1:n_models
1718     model.function = models{i,1};
    model.params0 = model_par0{i,1};
1720     model.paramslb = model_par_lb{i,1};
    model.paramsub = model_par_ub{i,1};
1722
    [bmodel2{i,1},fval2(i,1),tit2{i,1}] =...
1724         variogramfit(gexp2,nr_pairs2,c_centers2,iso,model,objmod,flag);
end
1726

1728 % >>> Cross Validation <<<
1730
1732 % Matrices and Cells preallocation
cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;
1734

1736 % Inputs definition
x = reshape(x_tr,ny,N/ny); y = reshape(y_tr,ny,N/ny);
rf = reshape(rf,ny,N/ny); iso = 1;d_col = 1;
1738

1740 % Cross Validation
for i=1:n_models
    model.function = models{i,1};
1742     model.params = bmodel2{i,1};
    r_ok1 = pdist2([x(1,1),y(1,1)],[x(1,2),y(1,2)])*2.0;
1744     r_ok2 = pdist2([x(1,1),y(1,1)],[x(2,1),y(2,1)])*4.0;
    model.r_ok = [r_ok1, r_ok2];
1746
    [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] =...
1748         crossval(x,y,rf,iso,d_col,model);
    cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
1750 end

1752 % Cross Validation Scores
table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
    'finalscore'};
1754 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

```

```

1756 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
1758 cv_Ss = [cv_Ss, FinalScore];
cv_Ssf = array2table(cv_Ss, 'VariableNames', table_h, 'RowNames', table_r);
1760 clear relMSE FinalScore

1762 %Trend addition and Boxcox Inversion
cv_St(n_models,6)= 0;
1764 for i=1:n_models

1766     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1768     if lambda==0
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1770     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1772     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
        ;
1774     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
1776     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

1778     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1780 end

1782 % Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
1784 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
1786 cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore

1788 %Plots
1790 for i=1:n_models

1792     % Stochastic Component's figures
    Z1 = reshape(rf,ny,N/ny);
1794     Z2 = reshape(cv_matr{i,1}.Z,ny,N/ny);

1796     figure;pcolor(Z1);%title('Sample Stochastic Component');
    view(2);shading interp; colorbar;set(gca, 'XTick',[], 'YTick',[]);

```



```

1798     figure; pcolor(Z2); %title(sprintf(' Estimation of Sample Stochastic
Component \n%s ', tit{i,1}));
1800     view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
1802     figure; scatter(Z1(:), Z2(:), 'filled', 'd'); hold on;
dvec1 = [Z1(:); Z2(:)];
1804     plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)
+0.5], 'r');
axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
1806     %title(' Scatter Plot ')
xlabel('Observed Data'); ylabel('Estimations');
1808     figure; h = histogram(Z1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
hold on
1810     histogram(Z2(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], '
FaceAlpha', 0.7)
%title('Histograms of Sample Stochastic Component')
1812     legend({'Original', 'Estimated'});
clear h

1814 % Total Data figures
Z3 = reshape(v1, ny, N/ny);
1816 Z4 = reshape(cv_matr{i,1}.Z_ibt, ny, N/ny);
%     figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
1818 %     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
figure; pcolor(Z4); view(2); shading interp; %title(sprintf(' Estimation
of Original Sample \n%s ', tit{i,1}));
1820 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
, [], 'YTick', []);
figure; scatter(Z3(:), Z4(:), 'filled', 'd'); hold on;
1822 dvec2 = [Z3(:); Z4(:)];
plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
1824 axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
%title(' Scatter Plot ')
1826 xlabel('Observed Data')
ylabel('Estimations')
1828 figure; h = histogram(Z3(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
hold on
1830 histogram(Z4(:), 'BinEdges', h.BinEdges, 'EdgeColor', [0.2 1 0], '
FaceAlpha', 0.7)
%title('Histograms of Sample Data')
1832 legend({'Original', 'Estimated'});

```

```

clear h
1834
clear Z1 Z2 Z3 Z4 dvec1 dvec2
1836
end
1838
1840 % >>> Ordinary Kriging <<<
1842 % Sort models based on cross validation scores
[~,ind] = sort(table2array(cv_Stf(:,7)), 'descend');
1844
% Matrices and Cells preallocation
1846 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
kr_matr{N_kr_mod,1}=[];
1848 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];
1850
% Inputs definition
1852 x = reshape(x_tr,ny,N/ny); y = reshape(y_tr,ny,N/ny);
rf = reshape(rf,ny,N/ny); xu = qx_tr; yu = qy_tr; iso=1;
1854
% Ordinary Kriging
1856 for i=1:N_kr_mod
    model.function = models{ind(i),1};
1858    model.params = bmodel2{ind(i),1};
    r_ok1 = pdist2([x(1,1),y(1,1)],[x(1,2),y(1,2)])*2.0;
1860    r_ok2 = pdist2([x(1,1),y(1,1)],[x(2,1),y(2,1)])*4.0;
    model.r_ok = [r_ok1, r_ok2];
1862
    [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
1864        ordkrig(x,y,rf,xu,yu,iso,model);
    kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
1866    table_r2{i,1} = table_r{ind(i),1};

    %Confidence Intervals (95%)
    CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
1870    CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
    CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
1872    UNC{i,1} = real(CI1{i,1}.uncer);
    realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
1874 end
1876 % Kriging Scores

```

```

relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
1878 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
1880 kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore
1882
%Trend addition and Boxcox Inversion
1884 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
1886 for i=1:N_kr_mod

1888     Z_tr{i,1} = qMx + Z{i,1};

1890     if lambda==0
        Z_ibt1{i,1} = Z_tr{i,1};
1892     elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
1894     else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
1896     end
    Z_ibt{i,1} = real(Z_ibt1{i,1});
1898     kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

1900     kr_St(i,:) = correstats(qv,Z_ibt{i,1});%total kriging scores

1902 end

1904 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
1906 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
1908 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore
1910
%Plots
1912 for i=1:N_kr_mod

1914     % Stochastic Component's figures
    cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
1916     cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
1918     Z2=Z1;
    for j = 1:size(cc1,1)
1920         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);

```

```

        Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1922 end
        figure; pcolor(Z1); %title('Original Stochastic Component');
1924 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
        figure; pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s', tit{ind(i),1}));
1926 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
        figure; scatter(qfluc(:), Z{i,1}(:), 'filled', 'd'); hold on;
1928 dvec1 = [qfluc(:); Z{i,1}(:)];
        plot([min(dvec1)-0.5, max(dvec1)+0.5], [min(dvec1)-0.5, max(dvec1)
+0.5], 'r');
1930 axis([min(dvec1)-0.5, max(dvec1)+0.5, min(dvec1)-0.5, max(dvec1)+0.5])
        %title('Scatter Plot')
1932 xlabel('Observed Data')
        ylabel('Estimations')
1934 figure; h = histogram(qfluc(:), 16, 'FaceColor', [0 0 1], 'FaceAlpha'
, 0.7);
        hold on
1936 histogram(Z{i,1}(:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0], '
FaceAlpha', 0.7)
        %title('Histograms of Stochastic Component')
1938 legend({'Original', 'Estimated'});
        clear h
1940
        % Total Data figures
1942 cc3 = [col, row, v1(:); qcol, qrow, Z_ibt{i,1}];
        cc4 = [col, row, zeros(size(col,1),1); qcol, qrow, UNC{i,1}];
1944 Z3(max(cc3(:,2)), max(cc3(:,1)))=0; %#ok<SAGROW>
        Z4=Z3;
1946 for j = 1:size(cc3,1)
            Z3(cc3(j,2), cc3(j,1)) = cc3(j,3);
1948            Z4(cc4(j,2), cc4(j,1)) = cc4(j,3);
        end
1950 figure; pcolor(Z3); view(2); shading interp; %title(sprintf('
Estimation of Original Data \n%s', tit{ind(i),1}));
        c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
, [], 'YTick', []);
1952 figure; scatter(qv1(:), Z_ibt{i,1}, 'filled', 'd'); hold on;
        dvec2 = [qv1(:); Z_ibt{i,1}(:)];
1954 plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
        axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
1956 %title('Scatter Plot')
        xlabel('Observed Data')

```

```

1958     ylabel ( 'Estimations' )
1959     figure ; h = histogram ( qv1 ( : ) , 16 , 'FaceColor' , [ 0 0 1 ] , 'FaceAlpha' , 0.7 ) ;
1960     hold on
1961     histogram ( Z_ibt { i , 1 } ( : ) , 'BinEdges' , h . BinEdges , 'FaceColor' , [ 0.2 1 0 ] ,
1962     'FaceAlpha' , 0.7 )
1963     %title ( 'Histograms of Data' )
1964     legend ( { 'Original' , 'Estimated' } ) ;
1965     figure ; pcolor ( Z4 ) ; view ( 2 ) ; shading interp ; %title ( '95% Confidence
1966     Interval' ) ;
1967     cl = colorbar ; cl . Label . String = 'Normalised Velocity' ; set ( gca , '
1968     XTick' , [ ] , 'YTick' , [ ] ) ;
1969     clear h
1970
1971     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2
1972
1973 end
1974
1975 % >>>Correlation Coefficient of Indicators<<<
1976 edges4 = ( 1500 : 4000 / 4 : 5500 ) / 5500 ;
1977 [ ~ , ~ , ind_data4 ] = histcounts ( qv1 , edges4 ) ;
1978 ind4_est { N_kr_mod , 1 } = [ ] ; Rpearson4 ( N_kr_mod , 1 ) = 0 ; Rspearman4 ( N_kr_mod , 1 )
1979     = 0 ;
1980 MCR4 ( N_kr_mod , 1 ) = 0 ;
1981
1982 edges16 = ( 1500 : 4000 / 16 : 5500 ) / 5500 ;
1983 [ ~ , ~ , ind_data16 ] = histcounts ( qv1 , edges16 ) ;
1984 ind16_est { N_kr_mod , 1 } = [ ] ; Rpearson16 ( N_kr_mod , 1 ) = 0 ; Rspearman16 ( N_kr_mod
1985     , 1 ) = 0 ;
1986 MCR16 ( N_kr_mod , 1 ) = 0 ;
1987
1988 for i = 1 : N_kr_mod
1989     [ ~ , ~ , ind4_est { i , 1 } ] = histcounts ( Z_ibt { i , 1 } , edges4 ) ;
1990     ind4_est { i , 1 } ( Z_ibt { i , 1 } <= edges4 ( 1 ) ) = 1 ;
1991     ind4_est { i , 1 } ( Z_ibt { i , 1 } >= edges4 ( end ) ) = length ( edges4 ) - 1 ;
1992     Rpearson4 ( i , 1 ) = corr ( ind4_est { i , 1 } , ind_data4 ) ;
1993     Rspearman4 ( i , 1 ) = corr ( ind4_est { i , 1 } , ind_data4 , 'type' , 'Spearman' ) ;
1994     MCR4 ( i , 1 ) = sum ( ind4_est { i , 1 } ~ ind_data4 ) / Nu ;
1995
1996     [ ~ , ~ , ind16_est { i , 1 } ] = histcounts ( Z_ibt { i , 1 } , edges16 ) ;
1997     ind16_est { i , 1 } ( Z_ibt { i , 1 } <= edges16 ( 1 ) ) = 1 ;
1998     ind16_est { i , 1 } ( Z_ibt { i , 1 } >= edges16 ( end ) ) = length ( edges16 ) - 1 ;
1999     Rpearson16 ( i , 1 ) = corr ( ind16_est { i , 1 } , ind_data16 ) ;
2000     Rspearman16 ( i , 1 ) = corr ( ind16_est { i , 1 } , ind_data16 , 'type' , 'Spearman' )
2001     ;

```

```

1996     MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
1998
2000 % Kriging Indicators Scores Table
    table_h = { 'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'Rpearson', 'Rspearman'
    };
2002 table_h2 = { 'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'Rpearson', 'Rspearman'
    , 'Rpearson4', 'Rspearman4', 'MCR4', 'Rpearson16', 'Rspearman16', 'MCR16'
    , };
    kr_iSsf = array2table(kr_Ss(:,1:end-1), 'VariableNames', table_h, 'RowNames'
    , table_r2);
2004 kr_iStf = array2table([ kr_St(:,1:end-1), Rpearson4, Rspearman4, MCR4,
    Rpearson16, Rspearman16, MCR16], 'VariableNames', table_h2, 'RowNames',
    table_r2);
    %
    =====

```

C:/Users/Vasilis/Desktop/AB/Dissertation/DiplomaThesis/MyThesis/Appendix2/AllRG4Appendix.m

B.3 Random Sample

```

1         % Dissertation
         % Random Sample
3
4
5     clc;clear variables;close all;
6
7     %##### Preliminary Analysis #####
8     %=====
9
10    % Basic parameters
    n_data = 2; %if 2 normalised velocities are used (values from 0 to 1),
        else
11        %if 1 are used velocities in km/s, else if 0 the original
        %velocities are used (m/s)
12
13
14    % Load Data
15    load marm.in
    ny = 122; nx = 384; %>>a priori known size of the image<<
17    marm_rf = reshape(marm,ny,nx);
    marm_rf = flipud(marm_rf);

```

```

19 idata0 = marm_rf; %original velocities (m/s)
21 idata1 = marm_rf/1000;%velocities in km/s
    idata2 = marm_rf/max(max(marm_rf)); %normalised velocities
23
    color = 'white'; fsize = 18.0; font = 'Cambria'; fweig = 'bold'; %figure
        appearance parameters
25
    if n_data==0
27        data = idata0;
        colbtitle = 'Velocity (m/s)';
29    elseif n_data==1
        data = idata1;
31        colbtitle = 'Velocity (km/s)';
    elseif n_data==2
33        data = idata2;
        colbtitle = 'Normalised Velocity';
35    else
        error('Error: Not valid value for n_data input. It must be 0,1 or 2.
            ');
37    end

39 [row_all , col_all , v_all1] = find(data);

41 % >>>Original Data Plots<<<

43 figure;%Simple plot
    pcolor(data); shading interp %flat
45 %title('Geological Section (Velocity Model)')
    set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',['])
47
    figure; %Simple plot plus colorbar
49 pcolor(data); shading interp %flat
    %title('Geological Section (Velocity Model)')
51 c = colorbar;
    c.Label.String = colbtitle;
53 set(gca,'XLim',[0 nx],'YLim',[0 ny],'XTick',[],'YTick',['])

55 figure; %Plot with actual distances and colorbar
    pcolor(flipud(data)); shading interp %flat
57 %title('Geological Section (Velocity Model)')
    c = colorbar;
59 c.Label.String = colbtitle;
    xlabel('Distance (km)');

```

```

61 ylabel('Depth (km)');
   ax = gca;
63 ax.XAxisLocation = 'top';
   ax.XLim = [0 nx]; ax.YLim = [0 ny];
65 ax.XTick = (0:9.2)*nx/9.2;
   ax.XTickLabel = (ax.XTick)*9.2/nx;
67 ax.YTick = (0:0.5:3)*ny/3;
   ax.YTickLabel = (ax.YTick)*3/ny;
69 ax.YDir = 'reverse';

71 data_tot_vec = data(:); %total image data in vector form

73
   % Random Sampling (122*39=4758 points)
75 spoints = 122*39; %number of sample points
   [~,idx] = datasample(data_tot_vec, spoints, 'Replace', false); %sampling
77 qpoints = data_tot_vec; qpoints(idx) = 0;
   qpoints = reshape(qpoints, ny, nx); %"matrix form" of unknown points
79 sample = data - qpoints; %"matrix form" of known data
   [qrow, qcol, qv1] = find(qpoints); %unknown points coordinations and
       values - VALIDATION SET
81 [row, col, v1] = find(sample); %known points coordinations and values -
       TRAINING SET
   N = length(v1); %number of known points

83
   % Plot Sample
85 figure; %Simple plot
   pcolor(sample); shading interp %flat
87 %title('Random Sample')
   set(gca, 'XLim', [0 nx], 'YLim', [0 ny], 'XTick', [], 'YTick', [])

89
   figure; %Simple plot plus colorbar
91 pcolor(sample); shading interp %flat
   %title('Random Sample')
93 c1 = colorbar;
   c1.Label.String = colbtitle;
95 set(gca, 'XLim', [0 nx], 'YLim', [0 ny], 'XTick', [], 'YTick', [])

97 figure; %Plot with actual distances and colorbar
   pcolor(flipud(sample)); shading interp %flat
99 %title('Geological Section (Velocity Model)')
   c1 = colorbar;
101 c1.Label.String = colbtitle;
   xlabel('Distance (km)');

```



```

103 ylabel('Depth (km)');
    ax = gca;
105 ax.XAxisLocation = 'top';
    ax.XLim = [0 nx]; ax.YLim = [0 ny];
107 ax.XTick = (0:9.2)*nx/9.2;
    ax.XTickLabel = (ax.XTick)*9.2/nx;
109 ax.YTick = (0:0.5:3)*ny/3;
    ax.YTickLabel = (ax.YTick)*3/ny;
111 ax.YDir = 'reverse';

113 %>>>Data Histograms & Statistical Moments<<<

115 numbins = 15;
    % Total image histogram
117 figure;
    %subplot(1,2,1)
119 histfit(data_tot_vec,numbins)
    alpha(0.5)
121 %title('Total image histogram')
    % Sample data histogram
123 figure;%subplot(1,2,2)
    histfit(v1,numbins)
125 alpha(0.5)
    %title('Sample data histogram')

127 % Total image data moments
129 totim_stats = [min(data(:)) max(data(:)) mean(data(:)) median(data(:))
    var(data(:)) skewness(data(:)) kurtosis(data(:))];
    % "Drill-holes" data moments
131 sample_stats = [min(v1) max(v1) mean(v1) median(v1) var(v1) skewness(v1)
    kurtosis(v1)];

133 %>>>Normality of data checking and Transformation<<<

135 % Histogram and Normal Probability Plot
137 figure; %same as the previous figure plus NPP
    %subplot(1,2,1)
139 histfit(v1,numbins)
    alpha(0.5)
141 %title('Sample data histogram')
    figure;%subplot(1,2,2)
143 nnp = normplot(v1);

```

```

h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
'); % remove normplot title
145 [h_orig,kst_p_orig,ksstat_orig,cv_orig] = kstest(v1(:));

147 % BoxCox Transformation
[v, lambda] = boxcox(v1);%known points boxcox transformation
149 qv = (qv1.^lambda -1)/lambda; %unknown points boxcox transformation
v_all = (v_all1.^lambda -1)/lambda; %all points boxcox transformation
151 figure;
%subplot(1,2,1)
153 histfit(v,numbins)
alpha(0.5)
155 %title('Transformed data histogram')
figure;%subplot(1,2,2)
157 normplot(v)
h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
'); % remove normplot title
159 [h_bxtr,p_bxtr,ksstat_bxtr,cv_bxtr] = kstest(v(:));

161 %>>> Data Trend Estimation <<<

163 nfr = 2;
trmodel = 'linear';
165 v_trend = reshape(v,ny,N/ny);
[fluc,Mx,Mx_func,a,trend_scores,dfreq,a_trends] = detrendv(col,row,
v_trend,nfr,trmodel,0);
167 qMx = Mx_func(qcol,qrow,a); %trend on the unknown points
qfluc = qv-qMx; %fluctuations/residuals on the unknown points
169

%Plot data & trend
171 X = col(:); Y = row(:); rf1 = v_trend(:);
figure;
173 scatter3(X,flipud(Y),rf1,'r','filled')
hold on
175 Xfit = min(X):1:max(X); nxfit = length(Xfit);
Yfit = min(Y):1:max(Y); nyfit = length(Yfit);
177 [XFIT,YFIT] = meshgrid(Xfit,Yfit);
VFIT = Mx_func(XFIT(:),YFIT(:),a);
179 VFIT = flipud(reshape(VFIT,nyfit,nxfit));
mesh(XFIT,YFIT,VFIT)
181 colorbar
%xlabel('x')
183 %ylabel('y')
xlabel('Alongside Section')

```

```

185 ylabel('Depth')
    xlabel('Transformed Normalised Velocity')
187 %set(gca,'YTick',flipud(get(gca,'YTick')));
    %set(gca,'YDir','reverse');
189 %title('Data & Trend Model')
    view(-52,6)
191 shading interp
    c = colorbar; c.Label.String = 'Transformed Normalised Velocity';
193
    %Detrend whole dataset
195 Mx_all = Mx_func(col_all,row_all,a); %trend on the unknown points
    fluc_all = v_all-Mx_all; %fluctuations/residuals on the unknown points
197 fluc_all = reshape(fluc_all,ny,nx);

199 figure; %Plot with actual distances and colorbar
    pcolor(flipud(fluc_all)); shading interp %flat
201 %title('Detrended Geological Section')
    cl = colorbar;
203 % cl.Label.String = colbtitle;
    xlabel('Distance (km)');
205 ylabel('Depth (km)');
    ax = gca;
207 ax.XAxisLocation = 'top';
    ax.XLim = [0 nx]; ax.YLim = [0 ny];
209 ax.XTick = (0:9.2)*nx/9.2;
    ax.XTickLabel = (ax.XTick)*9.2/nx;
211 ax.YTick = (0:0.5:3)*ny/3;
    ax.YTickLabel = (ax.YTick)*3/ny;
213 ax.YDir = 'reverse';

215 %>>> Statistical Analysis of Residuals <<<

217 % Residuals/Fluctuations moments
    fluc_stats = [min(fluc) max(fluc) mean(fluc) median(fluc) var(fluc)
        skewness(fluc) kurtosis(fluc)];
219

221 % Histogram and Normal Probablity Plot
    figure;
223 %subplot(1,2,1)
    histfit(fluc,numbins)
225 alpha(0.5)
    %title('Histogram of detrended data')
227 figure;%subplot(1,2,2)

```

```

normplot(fluc)
229 h_ch=get(gcf,'Children');h_str=get(h_ch(1),'Title');set(h_str,'String','
    '); % remove normplot title
[h_fluc,p_fluc,ksstat_fluc,cv_fluc] = kstest(fluc(:));
231
233 %Maximum distance
c = [col,row];
235 [k,l] = find(triu(true(N)));
d = hypot(c(k,1)-c(l,1),c(k,2)-c(l,2));
237 % d = triu(pdist2(c,c));
ncpc = 0.2;
239 maxdist = max(max(d))*ncpc;
clear k l d
241 %=====
243
245 %% ##### Experimental Variogram
    #####
    %=====
247
% >>> Initialize Basic Parameters <<<
249 c = {col,row}; %known points' coordinations
qc = {qcol,qrow}; %"unknown" points' coordinations
251 %ncpc = 0.2;
N = size(col,1);
253 Nu = size(qcol,1);
%maxdist = hypot(col(1,1)-col(N,1),row(1,1)-row(N,1))*ncpc;
255 nrbins = 45;
phistep = 15;
257 phitol = 20;
N_kr_mod = 3;
259
261 % >>> Experimental (Semi-)Variogram (anisotropic) <<<
x = col; y = row; rf = fluc; iso = 0; flag = 1;
263 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
    high analysis
[gexp, nr_pairs, c_centers] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,
    phitol,flag);
265 gexpmax = max(max(gexp));
%=====
267

```

```

269 %% ##### DirVar0 #####
270 %=====
271
272 % >>> Fitting I and Parameters of Anisotropic Correlation Estimation (s2
    ,xi1,xi2,phi,c0,v or etal) <<<
273
274 % Desired Models
275 models = {'Gexp','Gaus','Sphe','Mate','Spar'};
    n_models = length(models);
276
277 % Initial Values and Limits for optimization
278 b = [gexpmax,maxdist*2/3,0.5,10,gexpmax/100]; % s2,xi1,R,phi & c0
    b_lb = [eps,eps,eps,-90,eps]; b_ub = [gexpmax*1.5,maxdist*1.5,30,90,
        gexpmax/5]; %lower and upper limits
281 bsp = [1000,maxdist*2/3,0.5,10,gexpmax/100]; % eta0, xi1,R,phi & c0
    bsp_lb = [eps,eps,eps,-90,eps]; bsp_ub = [inf,maxdist*1.5,30,90,gexpmax
        /5]; %lower and upper limits
282
283 % Summary cells
284 model_par0 = {[b,1.5];b;b;[b,1.5];[bsp,1]}; %initial parameters values
    model_par_lb = {[b_lb,eps];b_lb;b_lb;[b_lb,0.3];[bsp_lb,-2+eps]}; %lower
        bounds
287 model_par_ub = {[b_ub,2-eps];b_ub;b_ub;[b_ub,3.5];[bsp_ub,inf]}; %upper
        bounds
288
289 clear b b_lb b_ub bsp bsp_lb bsp_ub
290
291 % Estimation of Parameters (s2,xi1,xi2,phi,c0,v or etal)
    bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
293 iso = 00;objmod = 'NWEr_m';flag = 1;
    for i=1:n_models
295         model.function = models{i,1};
            model.params0 = model_par0{i,1};
297         model.paramslb = model_par_lb{i,1};
            model.paramsub = model_par_ub{i,1};
298
299         [bmodel{i,1},fval(i,1),tit{i,1}] =...
301             variogramfit(gexp,nr_pairs,c_centers,iso,model,objmod,flag);
    end
302
303 % Anisotropy Estimation (#Not Necessary#)
304 R0(n_models,1) = 0;phi0(n_models,1) = 0;xi10(n_models,1) = 0;xi20(
    n_models,1) = 0;

```

```

for i=1:n_models
307   R0(i,1) = bmodel{i,1}(1,3);
      phi0(i,1) = bmodel{i,1}(1,4);
309   xi10(i,1) = bmodel{i,1}(1,2);
      xi20(i,1) = bmodel{i,1}(1,2)/R0(i,1);
311   if R0(i,1)>1
      R0(i,1) = 1/R0(i,1); bmodel{i,1}(1,3) = R0(i,1);
313   xi101 = xi10(i,1);
      xi10(i,1) = xi20(i,1); bmodel{i,1}(1,2) = xi10(i,1);
315   xi20(i,1) = xi101;
      if phi0(i,1)>0
317       phi0(i,1) = phi0(i,1)-90;
      else
319       phi0(i,1) = phi0(i,1)+90;
      end
321   bmodel{i,1}(1,4) = phi0(i,1);
      end
323 end

325 R = mean(R0); phi = mean(phi0); xi1 = mean(xi10); xi2 = mean(xi20);

327 % >>> Cross Validation <<<
329 % Matrices and Cells preallocation
331 cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
      cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;
333 % Inputs definition
335 iso = 00; d_col = 1;

337 % Cross Validation
for i=1:n_models
339   model.function = models{i,1};
      model.params = bmodel{i,1};
341   model.r_ok = [22,4];

343   [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
      crossval(x,y,rf,iso,d_col,model);
345   cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
end
347 % Cross Validation Scores

```

```

349 table_h = { 'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'rpearson', 'rspearman',
              'finalscore' };
table_r = { 'Gexp', 'Gaus', 'Sphe', 'Mate', 'Spar' };
351
relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
353 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
cv_Ss = [cv_Ss, FinalScore];
355 cv_Ssf = array2table(cv_Ss, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore
357
%Trend addition and Boxcox Inversion
359 cv_St(n_models,6)= 0;
for i=1:n_models
361
    cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;
363
    if lambda==0
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
    elseif lambda==0
365         cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
    else
367         cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
369 ;
    end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
    cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>
373
    cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores
375
end
377
% Total Cross Validation Scores
379 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
381 cv_St = [cv_St, FinalScore];
cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
383 clear relMSE FinalScore

385 %Plots
for i=1:n_models
387
    % Stochastic Component's figures
389 cc1 = [col,row, rf(:); qcol,qrow, zeros(Nu,1)];
    cc2 = [col,row, cv_matr{i,1}.Z(:); qcol,qrow, zeros(Nu,1)];

```

```

391 Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
Z2=Z1;
393 for j = 1:size(cc1,1)
    Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
395 Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
end
397
figure; pcolor(Z1);%title('Sample Stochastic Component');
399 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
401 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
figure; scatter(rf(:),cv_matr{i,1}.Z(:), 'filled', 'd'); hold on;
403 dvec1 = [rf(:);cv_matr{i,1}.Z(:)];
plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
405 axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
%title('Scatter Plot')
407 xlabel('Observed Data'); ylabel('Estimations');
figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
409 hold on
histogram(cv_matr{i,1}.Z(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1
0],'FaceAlpha',0.7)
411 %title('Histograms of Sample Stochastic Component')
legend({'Original', 'Estimated'});
413 clear h
415
% Total Data figures
417 cc3 = [col,row,v1(:);qcol,qrow,zeros(Nu,1)];
cc4 = [col,row,cv_matr{i,1}.Z_ibt(:);qcol,qrow,zeros(Nu,1)];
419 Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
Z4=Z3;
421 for j = 1:size(cc3,1)
    Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
423 Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
end
425 % figure;pcolor(Z3);view(2);shading interp; %title('Original Sample
');
% c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
427 figure;pcolor(Z4);view(2);shading interp; %title(sprintf('Estimation
of Original Sample \n%s', tit{i,1}));

```



```

c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick',
[], 'YTick', []);
429 figure; scatter(v1(:), cv_matr{i,1}.Z_ibt(:), 'filled', 'd'); hold on;
dvec2 = [v1(:); cv_matr{i,1}.Z_ibt(:)];
431 plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
433 %title('Scatter Plot')
xlabel('Observed Data')
435 ylabel('Estimations')
figure; h = histogram(v1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
437 hold on
histogram(cv_matr{i,1}.Z_ibt(:), 'BinEdges', h.BinEdges, 'EdgeColor',
,[0.2 1 0], 'FaceAlpha', 0.7)
439 %title('Histograms of Sample Data')
legend({'Original', 'Estimated'});
441 clear h

443 clear Z1 Z2 Z3 Z4 dvec1 dvec2

445 end

447 T_4 = toc(t1)/60-T_1-T_2-T_3

449

451 % >>> Ordinary Kriging <<<

453 % Sort models based on cross validation scores
[~, ind] = sort(table2array(cv_Stf(:,7)), 'descend');

455 % Matrices and Cells preallocation
Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
457 kr_matr{N_kr_mod,1}=[];
kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
459 CI1{N_kr_mod,1}=[]; UNC{N_kr_mod,1}=[];

461 % Inputs definition
xu = qcol; yu = qrow; iso = 00;

463 % Ordinary Kriging
465 for i=1:N_kr_mod
    model.function = models{ind(i),1};
467    model.params = bmodel{ind(i),1};
    model.r_ok = [22,4];

```

```

469 [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
471     ordkrig(x,y,rf,xu,yu,iso,model);
kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
473 table_r2{i,1} = table_r{ind(i),1};

475 %Confidence Intervals (95%)
CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
477 CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
479 UNC{i,1} = real(CI1{i,1}.uncer);
realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
481 end

483 % Kriging Scores
relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
485 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
487 kr_Ssf = array2table(kr_Ss, 'VariableNames',table_h, 'RowNames',table_r2);
clear relMSE FinalScore

489 %Trend addition and Boxcox Inversion
491 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0;Z_ibt{N_kr_mod,1} = 0;
493 for i=1:N_kr_mod

495     Z_tr{i,1} = qMx + Z{i,1};

497     if lambda==0
        Z_ibt1{i,1} = Z_tr{i,1};
499     elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
501     else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
503     end
    Z_ibt{i,1} = real(Z_ibt1{i,1});
505     kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

507     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores

509 end

511 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE

```

```

513 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
515 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore
517
%Plots
519 for i=1:N_kr_mod

    % Stochastic Component's figures
    cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
523 cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
525 Z2=Z1;
    for j = 1:size(cc1,1)
527         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
        Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
529    end
    figure; pcolor(Z1);%title('Original Stochastic Component');
531    view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    figure; pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s', tit{ind(i),1}));
533    view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    figure; scatter(qfluc(:),Z{i,1}(:), 'filled', 'd'); hold on;
535    dvec1 = [qfluc(:);Z{i,1}(:)];
    plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
537    axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
    %title('Scatter Plot')
539    xlabel('Observed Data')
    ylabel('Estimations')
541    figure; h = histogram(qfluc(:),16, 'FaceColor', [0 0 1], 'FaceAlpha'
,0.7);
    hold on
543    histogram(Z{i,1}(:), 'BinEdges', h.BinEdges, 'FaceColor', [0.2 1 0], '
FaceAlpha', 0.7)
    %title('Histograms of Stochastic Component')
545    legend({'Original', 'Estimated'});
    clear h
547

% Total Data figures
549 cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
    cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
551 Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
    Z4=Z3;

```

```

553     for j = 1:size(cc3,1)
        Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
555     Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
    end
557     figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
    c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick'
, [], 'YTick', []);
559     figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
    dvec2 = [qv1(:);Z_ibt{i,1}(:)];
561     plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
    axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
563     %title('Scatter Plot')
    xlabel('Observed Data')
565     ylabel('Estimations')
    figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
567     hold on
    histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
569     %title('Histograms of Data')
    legend({'Original','Estimated'});
571     figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
    c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[],'YTick',[]);
573     clear h

575     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

577 end

579 % >>>Correlation Coefficient of Indicators<<<
    edges4 = (1500:4000/4:5500)/5500;
581 [~,~,ind_data4] = histcounts(qv1,edges4);
    ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
=0;
583 MCR4(N_kr_mod,1)=0;

585 edges16 = (1500:4000/16:5500)/5500;
    [~,~,ind_data16] = histcounts(qv1,edges16);
587 ind16_est{N_kr_mod,1}=[];Rpearson16(N_kr_mod,1)=0;Rspearman16(N_kr_mod
,1)=0;
    MCR16(N_kr_mod,1)=0;

```

```

589 for i=1:N_kr_mod
591     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
        ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
593     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
        Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
595     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
        MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;
597
        [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
599     ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
        ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
601     Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
        Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman');
        ;
603     MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;

605 end

607 % Kriging Indicators Scores Table
        table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
                };
609 table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
                , 'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
                ,};
        kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
                ,table_r2);
611 kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
                Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
                table_r2);

        %=====
613

615 %% ##### DirVar1 #####
        %=====
617

        % >>> Fitting I and Anisotropy Estimation <<<
619

        % Desired Models
621 models = {'Gexp','Gaus','Sphe','Mate','Spar'};
        n_models = length(models);
623

        % Initial Values and Limits for optimization
625 b = [gexpmax,maxdist*2/3,gexpmax/100]; % s2,xi & c0

```

```

b_lb = [eps,eps,eps]; b_ub = [gexpmax*1.5,maxdist*1.5,gexpmax/5]; %lower
    and upper limits
627 bsp = [1000,maxdist*2/3,gexpmax/100]; % eta0 , xi , c0
    bsp_lb = [eps,eps,eps]; bsp_ub = [inf,maxdist*1.5,gexpmax/5]; %lower and
    upper limits
629
% Summary cells
631 model_par0 = {[b,1.5];b;b:[b,1.5];[bsp,1]}; %initial parameters values
    model_par_lb = {[b_lb,eps];b_lb;b_lb:[b_lb,0.3];[bsp_lb,-2+eps]}; %lower
    bounds
633 model_par_ub = {[b_ub,2-eps];b_ub;b_ub:[b_ub,3.5];[bsp_ub,inf]}; %upper
    bounds
635 clear b b_lb b_ub bsp bsp_lb bsp_ub

637 % Estimation of Anisotropy
    R(n_models,1)=0;phi(n_models,1)=0;xi1(n_models,1)=0;xi2(n_models,1)=0;%
    matrices preallocation
639 er1(n_models,1)=0;er2(n_models,1)=0; %matrices preallocation
    objmod = 'NWEr_m';flag = 1;
641 % nrsampl = 10; samplpc = 50;
    for i=1:n_models
643         model.function = models{i,1};
        model.params0 = model_par0{i,1};
645         model.paramslb = model_par_lb{i,1};
        model.paramsub = model_par_ub{i,1};
647
        [R(i,1),phi(i,1),xi1(i,1),xi2(i,1),er1(i,1),er2(i,1)] =...
649         aniso_dvf(x,y,rf,model,objmod,ncpc,nrbins,phistep,phitol,flag);
        % [R(i,1),phi(i,1),xi1(i,1),xi2(i,1),er1(i,1),er2(i,1)] =...
651 %         dirvar_ccv_s(x,y,rf,model,objmod,N_ms,ncpc,nrsampl,samplpc,
            nrbins,phistep,phitol,flag);

653 end

655 save('dirvar1_iso.mat')

657
% >>> Fitting II and Parameters of Correlation Estimation (s2,c0,v or
    etal) <<<
659
% Initial Values and Limits for optimization
661 b{n_models,1}=[];b_lb{n_models,1}=[];b_ub{n_models,1}=[];
    for i = 1:n_models

```

```

663     b{i,1} = [gexpmax, xi1(i,1), R(i,1), phi(i,1), gexpmax/100];
        b_lb{i,1} = [eps, xi1(i,1), R(i,1), phi(i,1), eps];
665     b_ub{i,1} = [inf, xi1(i,1), R(i,1), phi(i,1), gexpmax/5];
end
667
% Summary cells
669 model_par0 = {[b{1,1}, 1.1]; b{2,1}; b{3,1}; [b{4,1}, 1.5]; [b{5,1}, 1]}; %
        initial parameters values
        model_par_lb = {[b_lb{1,1}, eps]; b_lb{2,1}; b_lb{3,1}; [b_lb{4,1}, 0.3]; [
        b_lb{5,1}, -2+eps]}; %lower bounds
671 model_par_ub = {[b_ub{1,1}, 2]; b_ub{2,1}; b_ub{3,1}; [b_ub{4,1}, 3.5]; [b_ub
        {5,1}, inf]}; %upper bounds

673 % Estimation of Parameters (s2, c0, v or etal)
        bmodel{n_models, 1}=[]; fval(n_models, 1)=0; tit{n_models, 1}=[];
675 iso = 00; objmod = 'NWEr_m'; flag = 1;
        for i=1:n_models
677             model.function = models{i, 1};
                model.params0 = model_par0{i, 1};
679             model.paramslb = model_par_lb{i, 1};
                model.paramsub = model_par_ub{i, 1};

681
                [bmodel{i, 1}, fval(i, 1), tit{i, 1}] = ...
683                     variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
        end
685

687 % >>> Cross Validation <<<

689 % Matrices and Cells preallocation
        cv_scores{n_models, 1}=[]; cv_checks{n_models, 1}=[];
691 cv_matr{n_models, 1}=[]; cv_Ss(n_models, 6)= 0;

693 % Inputs definition
        iso = 00; d_col = 1;
695

% Cross Validation
697 for i=1:n_models
        model.function = models{i, 1};
699         model.params = bmodel{i, 1};
                model.r_ok = [22, 4];

701
                [cv_scores{i, 1}, cv_checks{i, 1}, cv_matr{i, 1}] = ...
703                     crossval(x, y, rf, iso, d_col, model);

```

```

    cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
705 end

% Cross Validation Scores
707 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
    'finalscore'};
709 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

711 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
    FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
713 cv_Ss = [cv_Ss, FinalScore];
    cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
715 clear relMSE FinalScore

%Trend addition and Boxcox Inversion
717 cv_St(n_models,6)= 0;
719 for i=1:n_models

721     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

723     if lambda==00
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
725     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
727     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
    ;
729     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
731     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

733     cv_St(i,:) = correstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

735 end

% Total Cross Validation Scores
737 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
739 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
    cv_St = [cv_St, FinalScore];
741 cv_Stf = array2table(cv_St,'VariableNames',table_h,'RowNames',table_r);
    clear relMSE FinalScore
743

%Plots
745 for i=1:n_models

```



```

747 % Stochastic Component's figures
cc1 = [ col ,row , rf ( : ) ; qcol , qrow , zeros (Nu,1) ];
749 cc2 = [ col ,row , cv_matr{ i ,1 }.Z ( : ) ; qcol , qrow , zeros (Nu,1) ];
Z1(max(cc1 (: ,2) ),max(cc1 (: ,1) ))=0; %#ok<SAGROW>
751 Z2=Z1;
for j = 1:size(cc1,1)
753     Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
    Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
755 end

figure; pcolor(Z1);%title('Sample Stochastic Component');
view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
759 figure; pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
761 figure; scatter(rf (:), cv_matr{ i ,1 }.Z (:), 'filled', 'd'); hold on;
dvec1 = [ rf (:); cv_matr{ i ,1 }.Z (:) ];
763 plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
765 %title('Scatter Plot')
xlabel('Observed Data'); ylabel('Estimations');
767 figure; h = histogram(rf (:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
769 histogram(cv_matr{ i ,1 }.Z (:), 'BinEdges', h.BinEdges, 'EdgeColor',[0.2 1
0], 'FaceAlpha',0.7)
%title('Histograms of Sample Stochastic Component')
771 legend({'Original', 'Estimated'});
clear h
773

775 % Total Data figures
cc3 = [ col ,row , v1 ( : ) ; qcol , qrow , zeros (Nu,1) ];
777 cc4 = [ col ,row , cv_matr{ i ,1 }.Z_ibt ( : ) ; qcol , qrow , zeros (Nu,1) ];
Z3(max(cc3 (: ,2) ),max(cc3 (: ,1) ))=0; %#ok<SAGROW>
779 Z4=Z3;
for j = 1:size(cc3,1)
781     Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
    Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
783 end
% figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
```

```

785 %      c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
      XTick', [], 'YTick', []);
      figure; pcolor(Z4); view(2); shading interp; %title(sprintf(' Estimation
      of Original Sample \n%s', tit{i,1}));
787 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
      , [], 'YTick', []);
      figure; scatter(v1(:), cv_matr{i,1}.Z_ibt(:), 'filled', 'd'); hold on;
789 dvec2 = [v1(:); cv_matr{i,1}.Z_ibt(:)];
      plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
      +0.5], 'r');
791 axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
      %title('Scatter Plot')
793 xlabel('Observed Data')
      ylabel('Estimations')
795 figure; h = histogram(v1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
      hold on
797 histogram(cv_matr{i,1}.Z_ibt(:), 'BinEdges', h.BinEdges, 'EdgeColor'
      , [0.2 1 0], 'FaceAlpha', 0.7)
      %title('Histograms of Sample Data')
799 legend({'Original', 'Estimated'});
      clear h
801
      clear Z1 Z2 Z3 Z4 dvec1 dvec2
803
805 end
807
809 % >>> Ordinary Kriging <<<
811 % Sort models based on cross validation scores
      [~, ind] = sort(table2array(cv_Stf(:, 7)), 'descend');
813 % Matrices and Cells preallocation
      Z{N_kr_mod, 1} = []; Z_error{N_kr_mod, 1} = []; kr_checks{N_kr_mod, 1} = [];
815 kr_matr{N_kr_mod, 1} = [];
      kr_Ss(N_kr_mod, 6) = 0; table_r2{N_kr_mod, 1} = [];
817 CI1{N_kr_mod, 1} = []; UNC{N_kr_mod, 1} = [];
819 % Inputs definition
      xu = qcol; yu = qrow; iso = 00;
821
823 % Ordinary Kriging
      for i = 1:N_kr_mod

```

```

model.function = models{ind(i),1};
825 model.params = bmodel{ind(i),1};
model.r_ok = [22,4];

827
[Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
829   ordkrig(x,y,rf,xu,yu,iso,model);
kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
831 table_r2{i,1} = table_r{ind(i),1};

%Confidence Intervals (95%)
833 CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
835 CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
837 UNC{i,1} = real(CI1{i,1}.uncer);
realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
839 end

% Kriging Scores
841 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
843 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
845 kr_Ssf = array2table(kr_Ss,'VariableNames',table_h,'RowNames',table_r2);
clear relMSE FinalScore

847 %Trend addition and Boxcox Inversion
849 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0;Z_ibt{N_kr_mod,1} = 0;
851 for i=1:N_kr_mod

853   Z_tr{i,1} = qMx + Z{i,1};

855   if lambda==00
       Z_ibt1{i,1} = Z_tr{i,1};
857   elseif lambda==0
       Z_ibt1{i,1} = exp(Z_tr{i,1});
859   else
       Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
861   end
   Z_ibt{i,1} = real(Z_ibt1{i,1});
863   kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

865   kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
867 end

```

```

869 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
871 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
873 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore
875
%Plots
877 for i=1:N_kr_mod

879     % Stochastic Component's figures
cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
881 cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
883 Z2=Z1;
for j = 1:size(cc1,1)
885     Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
887 end
figure;pcolor(Z1);%title('Original Stochastic Component');
889 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s',tit{ind(i),1}));
891 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
893 dvec1 = [qfluc(:);Z{i,1}(:)];
plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
895 axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
%title('Scatter Plot')
897 xlabel('Observed Data')
ylabel('Estimations')
899 figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha'
,0.7);
hold on
901 histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
%title('Histograms of Stochastic Component')
903 legend({'Original','Estimated'});
clear h
905
% Total Data figures
907 cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];

```

```

cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
909 Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
Z4=Z3;
911 for j = 1:size(cc3,1)
    Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
913 Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
end
915 figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick'
,[],'YTick',[]);
917 figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
dvec2 = [qv1(:);Z_ibt{i,1}(:)];
919 plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
921 %title('Scatter Plot')
xlabel('Observed Data')
923 ylabel('Estimations')
figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
925 hold on
histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
927 %title('Histograms of Data')
legend({'Original','Estimated'});
929 figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[],'YTick',[]);
931 clear h

933 clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

935 end

937 % >>>Correlation Coefficient of Indicators<<<
edges4 = (1500:4000/4:5500)/5500;
939 [~,~,ind_data4] = histcounts(qv1,edges4);
ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
=0;
941 MCR4(N_kr_mod,1)=0;

943 edges16 = (1500:4000/16:5500)/5500;
[~,~,ind_data16] = histcounts(qv1,edges16);

```

```

945 ind16_est{N_kr_mod,1}=[]; Rpearson16(N_kr_mod,1)=0; Rspearman16(N_kr_mod
    ,1)=0;
MCR16(N_kr_mod,1)=0;
947
for i=1:N_kr_mod
949     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
    ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
951     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
    Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
953     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
    MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;
955
    [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
957     ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
    ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
959     Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
    Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
    ;
961     MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
end
963
% Kriging Indicators Scores Table
965 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    };
    table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
        , 'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
        ,};
967 kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
    ',table_r2);
    kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
        Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
        table_r2);
969 %=====
971
%% ##### DirVar2 #####
973 %=====
975 load('dirvar1_iso.mat')
977 % >>> Rescaling and Rotation of the random field
979 % (Inverse) Transformation Matrix

```

```

A{n_models,1} = []; c_tr{n_models,1} = []; x_tr{n_models,1} = []; y_tr{
    n_models,1} = [];
981 qc_tr{n_models,1} = []; qx_tr{n_models,1} = []; qy_tr{n_models,1} = [];
R_tr(n_models,1)=0; phi_tr(n_models,1)=0; xi1_tr(n_models,1)=0; xi2_tr(
    n_models,1)=0;
983
gexp_tr{n_models,1}=[]; nr_pairs_tr{n_models,1}=[]; c_centers_tr{
    n_models,1}=[];
985
for i=1:n_models
987 A{i,1} = [cos(phi(i,1)*pi/180)/xi1(i,1), sin(phi(i,1)*pi/180)/xi1(i,1);
    -sin(phi(i,1)*pi/180)/xi2(i,1), cos(phi(i,1)*pi/180)/xi2(i,1)];
989
% Rescale and Rotate Coordinations
991 c_tr{i,1} = A{i,1}*[c{1,1}';c{1,2}']; %transformed coordinations of
    known points
x_tr{i,1} = c_tr{i,1}(1,:);
993 y_tr{i,1} = c_tr{i,1}(2,:);
qc_tr{i,1} = A{i,1}*[qc{1,1}';qc{1,2}']; %transformed coordinations of
    unknown points
995 qx_tr{i,1} = qc_tr{i,1}(1,:);
qy_tr{i,1} = qc_tr{i,1}(2,:);
997
% >>> Check isotropy <<<
999
% Experimental Variogram (anisotropic)
1001 x = x_tr{i,1}; y = y_tr{i,1}; rf = fluc; iso = 0;
[~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
    high analysis
1003 %[gexp_tr{i,1}, nr_pairs_tr{i,1}, c_centers_tr{i,1}] = expvar(x,y,rf,iso
    ,ncpc,nrbins,phistep,phitol,2);

1005 % Estimation of New Anisotropy
model.function = models{i,1};
1007 model.params0 = model_par0{i,1};
model.paramslb = model_par_lb{i,1};
1009 model.paramsub = model_par_ub{i,1};
[R_tr(i,1), phi_tr(i,1), xi1_tr(i,1), xi2_tr(i,1),~,~] = ...
1011 aniso_dvf(c_tr{i,1}(1,:),c_tr{i,1}(2,:),rf,model,'NWEr_m',ncpc,
    nrbins,phistep,phitol,0);

1013 end

```

```

1015 % >>> Fitting II and Parameters of Correlation Estimation (s2,c0,v or
      etal) <<<

1017 % Experimental (Semi-)Variogram (isotropic)
      gexp2{n_models,1}=[]; nr_pairs2{n_models,1}=[]; c_centers2{n_models
      ,1}=[];
1019 gexpmax2(n_models,1)=0; maxdist2(n_models,1)=0;
      [k,1] = find(triu(true(N)));
1021 for i=1:n_models
      x = x_tr{i,1}; y = y_tr{i,1}; rf = fluc; iso = 1; flag = 1;
1023 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
      high analysis
      [gexp2{i,1}, nr_pairs2{i,1}, c_centers2{i,1}] = expvar(x,y,rf,iso,ncpc,
      nrbins,phistep,phitol,flag);
1025 gexpmax2(i,1) = max(max(gexp2{i,1}));
      d = hypot(c_tr{i,1}(1,k)-c_tr{i,1}(1,1),c_tr{i,1}(2,k)-c_tr{i,1}(2,1));
1027 maxdist2(i,1) = ncpc*max(max(d));
      clear d
1029 end

1031 % Initial Values and Limits for optimization
      b{n_models,1}=[]; b_lb{n_models,1}=[]; b_ub{n_models,1}=[];
1033 for i = 1:n_models
      b{i,1} = [gexpmax2(i,1),maxdist2(i,1)*2/3,gexpmax2(i,1)/100];
1035 b_lb{i,1} = [eps,eps,eps];
      b_ub{i,1} = [inf,maxdist2(i,1)*1.5,gexpmax2(i,1)/5];
1037 end

1039 % Summary cells
      model_par0 = {[b{1,1},1.1];b{2,1};b{3,1};[b{4,1},1.5];[b{5,1},1]}; %
      initial parameters values
1041 model_par_lb = {[b_lb{1,1},eps];b_lb{2,1};b_lb{3,1};[b_lb{4,1},0.3];[
      b_lb{5,1},-2+eps]};%lower bounds
      model_par_ub = {[b_ub{1,1},2];b_ub{2,1};b_ub{3,1};[b_ub{4,1},3.5];[b_ub
      {5,1},inf]}; %upper bounds
1043

      % Estimation of Parameters (s2,c0,v or etal)
1045 bmodel{n_models,1}=[]; fval(n_models,1)=0; tit{n_models,1}=[];
      iso = 1; objmod = 'NWEr_m'; flag = 1;
1047 for i=1:n_models
      model.function = models{i,1};
1049 model.params0 = model_par0{i,1};
      model.paramslb = model_par_lb{i,1};
1051 model.paramsub = model_par_ub{i,1};

```



```

1053     [bmodel{i,1}, fval(i,1), tit{i,1}] =...
        variogramfit(gexp2{i,1}, nr_pairs2{i,1}, c_centers2{i,1}, iso, model,
        objmod, flag);
1055 end

1057
1059 % >>> Cross Validation <<<

1061 % Matrices and Cells preallocation
cv_scores{n_models,1}=[]; cv_checks{n_models,1}=[];
cv_matr{n_models,1}=[]; cv_Ss(n_models,6)= 0;

1063 % Inputs definition
1065 rf = rf(:); iso = 1; d_col = 1;

1067 % Cross Validation
for i=1:n_models
1069     x = x_tr{i,1}(:); y = y_tr{i,1}(:);
        model.function = models{i,1};
1071     model.params = bmodel{i,1};
        rx_tr = pdist2(x_tr{i,1}(:), x_tr{i,1}(:));
1073     ry_tr = pdist2(y_tr{i,1}(:), y_tr{i,1}(:));
        model.r_ok = [mean(rx_tr(:))/5, mean(ry_tr(:))/5];

1075     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] =...
        crossval(x,y,rf,iso,d_col,model);
1077     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
1079 end

1081 % Cross Validation Scores
table_h = {'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'rpearson', 'rspearman',
        'finalscore'};
1083 table_r = {'Gexp', 'Gaus', 'Sphe', 'Mate', 'Spar'};

1085 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
1087 cv_Ss = [cv_Ss, FinalScore];
cv_Ssf = array2table(cv_Ss, 'VariableNames', table_h, 'RowNames', table_r);
1089 clear relMSE FinalScore

1091 %Trend addition and Boxcox Inversion
cv_St(n_models,6)= 0;
1093 for i=1:n_models

```

```

1095     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1097     if lambda==00
        cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1099     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1101     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
;
1103     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
1105     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

1107     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1109 end

1111 % Total Cross Validation Scores
relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
1113 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
cv_St = [cv_St, FinalScore];
1115 cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore

1117 %Plots
1119 for i=1:n_models

1121     % Stochastic Component's figures
    cc1 = [col,row,rf(:);qcol,qrow,zeros(Nu,1)];
1123     cc2 = [col,row,cv_matr{i,1}.Z(:);qcol,qrow,zeros(Nu,1)];
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
1125     Z2=Z1;
    for j = 1:size(cc1,1)
1127         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
        Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1129     end

1131     figure;pcolor(Z1);%title('Sample Stochastic Component');
    view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
1133     figure;pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
    view(2);shading interp;colorbar;set(gca,'XTick',[],'YTick',[]);
1135     figure;scatter(rf(:),cv_matr{i,1}.Z(:),'filled','d');hold on;

```

```

dvec1 = [ rf(:);cv_matr{i,1}.Z(:) ];
1137 plot([ min(dvec1)-0.5,max(dvec1)+0.5],[ min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
axis([ min(dvec1)-0.5,max(dvec1)+0.5, min(dvec1)-0.5,max(dvec1)+0.5])
1139 %title(' Scatter Plot ')
xlabel('Observed Data'); ylabel('Estimations');
1141 figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
1143 histogram(cv_matr{i,1}.Z(:),'BinEdges',h.BinEdges,'EdgeColor',[0.2 1
0],'FaceAlpha',0.7)
%title('Histograms of Sample Stochastic Component')
1145 legend({'Original','Estimated'});
clear h
1147

% Total Data figures
1149 cc3 = [ col,row,v1(:);qcol,qrow,zeros(Nu,1) ];
1151 cc4 = [ col,row,cv_matr{i,1}.Z_ibt(:);qcol,qrow,zeros(Nu,1) ];
Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
1153 Z4=Z3;
for j = 1:size(cc3,1)
1155     Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
    Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
1157 end
% figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
');
1159 % c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[ ],'YTick',[ ]);
figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s',tit{i,1}));
1161 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca,'XTick'
,[ ],'YTick',[ ]);
figure; scatter(v1(:),cv_matr{i,1}.Z_ibt(:),'filled','d'); hold on;
1163 dvec2 = [ v1(:);cv_matr{i,1}.Z_ibt(:) ];
plot([ min(dvec2)-0.5,max(dvec2)+0.5],[ min(dvec2)-0.5,max(dvec2)
+0.5], 'r');
1165 axis([ min(dvec2)-0.5,max(dvec2)+0.5, min(dvec2)-0.5,max(dvec2)+0.5])
%title(' Scatter Plot ')
1167 xlabel('Observed Data')
ylabel('Estimations')
1169 figure; h = histogram(v1(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
hold on
1171 histogram(cv_matr{i,1}.Z_ibt(:),'BinEdges',h.BinEdges,'EdgeColor'
,[0.2 1 0],'FaceAlpha',0.7)

```

```

%title('Histograms of Sample Data')
1173 legend({'Original', 'Estimated'});
clear h

1175
clear Z1 Z2 Z3 Z4 dvec1 dvec2

1177
end
1179

1181 % >>> Ordinary Kriging <<<

1183 % Sort models based on cross validation scores
[~,ind] = sort(table2array(cv_Stf(:,7)), 'descend');
1185
% Matrices and Cells preallocation
1187 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
kr_matr{N_kr_mod,1}=[];
1189 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
CI1{N_kr_mod,1}=[];UNC{N_kr_mod,1}=[];
1191
% Inputs definition
1193 rf = rf(:); iso = 1;

1195 % Ordinary Kriging
for i=1:N_kr_mod
1197 x = x_tr{i,1}(:); y = y_tr{i,1}(:);
xu = qx_tr{i,1}(:); yu = qy_tr{i,1}(:);
1199 model.function = models{ind(i),1};
model.params = bmodel{ind(i),1};
1201 rx_tr = pdist2(x_tr{i,1}(:), x_tr{i,1}(:));
ry_tr = pdist2(y_tr{i,1}(:), y_tr{i,1}(:));
1203 model.r_ok = [mean(rx_tr(:))/5, mean(ry_tr(:))/5];

1205 [Z{i,1}, Z_error{i,1}, kr_checks{i,1}, kr_matr{i,1}] = ...
ordkrig(x,y,rf,xu,yu,iso,model);
1207 kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
table_r2{i,1} = table_r{ind(i),1};

1209
%Confidence Intervals (95%)
1211 CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
1213 CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
UNC{i,1} = real(CI1{i,1}.uncer);
1215 realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>

```

```

end
1217
% Kriging Scores
1219 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
1221 kr_Ss = [kr_Ss, FinalScore];
kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
1223 clear relMSE FinalScore

%Trend addition and Boxcox Inversion
1225 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
1227 Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
for i=1:N_kr_mod
1229
    Z_tr{i,1} = qMx + Z{i,1};
1231
    if lambda==0
        Z_ibt1{i,1} = Z_tr{i,1};
1233
    elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
1235
    else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
1237
    end
    Z_ibt{i,1} = real(Z_ibt1{i,1});
1239
    kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>
1241
    kr_St(i,:) = correstats(qv,Z_ibt{i,1});%total kriging scores
1243
end
1245
% Total Kriging Scores
1247 relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
1249 kr_St = [kr_St, FinalScore];
kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
1251 clear relMSE FinalScore

%Plots
for i=1:N_kr_mod
1253
    % Stochastic Component's figures
1255
    cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
1257
    cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
1259
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>

```

```

Z2=Z1;
1261 for j = 1:size(cc1,1)
        Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
1263     Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
    end
1265 figure;pcolor(Z1);%title('Original Stochastic Component');
    view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
1267 figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s',tit{ind(i),1}));
    view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
1269 figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
    dvec1 = [qfluc(:);Z{i,1}(:)];
1271 plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
    axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
1273 %title('Scatter Plot')
    xlabel('Observed Data')
1275 ylabel('Estimations')
    figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha',
,0.7);
1277 hold on
    histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
1279 %title('Histograms of Stochastic Component')
    legend({'Original','Estimated'});
1281 clear h

% Total Data figures
cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
1285 cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
    Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
1287 Z4=Z3;
    for j = 1:size(cc3,1)
1289         Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
        Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
1291    end
    figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
1293 c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick',
,[],'YTick',[]);
    figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
1295 dvec2 = [qv1(:);Z_ibt{i,1}(:)];
    plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');

```

```

1297     axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
1298     %title('Scatter Plot')
1299     xlabel('Observed Data')
1300     ylabel('Estimations')
1301     figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
1302     hold on
1303     histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
1304     'FaceAlpha',0.7)
1305     %title('Histograms of Data')
1306     legend({'Original','Estimated'});
1307     figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
1308     Interval');
1309     c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
1310     XTick',[1],'YTick',[1]);
1311     clear h
1312
1313     clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2
1314
1315 end
1316
1317 % >>>Correlation Coefficient of Indicators<<<
1318 edges4 = (1500:4000/4:5500)/5500;
1319 [~,~,ind_data4] = histcounts(qv1,edges4);
1320 ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
1321 =0;
1322 MCR4(N_kr_mod,1)=0;
1323
1324 edges16 = (1500:4000/16:5500)/5500;
1325 [~,~,ind_data16] = histcounts(qv1,edges16);
1326 ind16_est{N_kr_mod,1}=[];Rpearson16(N_kr_mod,1)=0;Rspearman16(N_kr_mod
1327 ,1)=0;
1328 MCR16(N_kr_mod,1)=0;
1329
1330 for i=1:N_kr_mod
1331     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
1332     ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
1333     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
1334     Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
1335     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
1336     MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;
1337
1338     [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
1339     ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
1340     ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;

```

```

1337     Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
1338     Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
1339     ;
1340     MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
1341
1342 end
1343
1344 % Kriging Indicators Scores Table
1345 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
1346     };
1347 table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
1348     ,'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
1349     ,};
1350 kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
1351     ',table_r2');
1352 kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
1353     Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
1354     table_r2);
1355
1356 %=====
1357
1358 %% ##### CHI1 #####
1359
1360 %=====
1361
1362 % >>> Anisotropy estimation with CHI <<<
1363
1364 [R,phi] = aniso_interp_scatter(x,y,rf,nx,'v4');
1365 if R>1
1366     R = 1/R;
1367     if phi>0
1368         phi = phi-90;
1369     elseif phi<0
1370         phi = phi+90;
1371     end
1372 end
1373
1374 save('chi1_iso.mat')
1375
1376 % >>> Fitting II and Parameters of Correlation Estimation (s2,c0,v or
1377     etal) <<<
1378
1379 % Desired Models
1380 models = {'Gexp','Gaus','Sphe','Mate','Spar'};
1381 n_models = length(models);

```



```

1373 % Initial Values and Limits for optimization
b = [gexpmax, maxdist*1/3, R, phi, gexpmax/100]; % s2, xi & c0
1375 b_lb = [eps, eps, R, phi, eps]; b_ub = [gexpmax*1.5, maxdist, R, phi, gexpmax
/5]; %lower and upper limits
bsp = [1000, maxdist*1/3, R, phi, gexpmax/100]; % eta0, xi, c0
1377 bsp_lb = [eps, eps, R, phi, eps]; bsp_ub = [inf, maxdist, R, phi, gexpmax/5]; %
lower and upper limits

1379 % Summary cells
model_par0 = {[b, 1.5]; b; b; [b, 1.5]; [bsp, 1]}; %initial parameters values
1381 model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
bounds
model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
bounds

1383 clear b b_lb b_ub bsp bsp_lb bsp_ub
1385
% Estimation of Parameters (s2, xi1, c0, v or etal)
1387 bmodel{n_models, 1}=[]; fval(n_models, 1)=0; tit{n_models, 1}=[];
iso = 00; objmod = 'NWEr_m'; flag = 1;
1389 for i=1:n_models
    model.function = models{i, 1};
1391    model.params0 = model_par0{i, 1};
    model.paramslb = model_par_lb{i, 1};
1393    model.paramsub = model_par_ub{i, 1};

1395    [bmodel{i, 1}, fval(i, 1), tit{i, 1}] = ...
        variogramfit(gexp, nr_pairs, c_centers, iso, model, objmod, flag);
1397 end

1399
% >>> Cross Validation <<<
1401
% Matrices and Cells preallocation
1403 cv_scores{n_models, 1}=[]; cv_checks{n_models, 1}=[];
cv_matr{n_models, 1}=[]; cv_Ss(n_models, 6)= 0;
1405
% Inputs definition
1407 x = x(:); y = y(:); rf = rf(:);
iso = 00; d_col = 1;
1409
% Cross Validation
1411 for i=1:n_models

```

```

1413     model.function = models{i,1};
1414     model.params = bmodel{i,1};
1415     model.r_ok = [22,4];

1416     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
1417         crossval(x,y,rf,iso,d_col,model);
1418     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
1419 end

1420 % Cross Validation Scores
1421 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','rpearson','rspearman',
1422     'finalscore'};
1423 table_r = {'Gexp','Gaus','Sphe','Mate','Spar'};

1424 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
1425 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
1426 cv_Ss = [cv_Ss, FinalScore];
1427 cv_Ssf = array2table(cv_Ss,'VariableNames',table_h,'RowNames',table_r);
1428 clear relMSE FinalScore

1429 %Trend addition and Boxcox Inversion
1430 cv_St(n_models,6)= 0;
1431 for i=1:n_models

1432     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1433     if lambda==00
1434         cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1435     elseif lambda==0
1436         cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1437     else
1438         cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
1439     ;
1440     end
1441     cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
1442     cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %#ok<SAGROW>

1443     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1444 end

1445 % Total Cross Validation Scores
1446 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
1447 FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore

```

```

cv_St = [cv_St, FinalScore];
1455 cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore
1457
%Plots
1459 for i=1:n_models

    % Stochastic Component's figures
    cc1 = [col,row, rf(:); qcol,qrow, zeros(Nu,1)];
    1463 cc2 = [col,row, cv_matr{i,1}.Z(:); qcol,qrow, zeros(Nu,1)];
    Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
    1465 Z2=Z1;
    for j = 1:size(cc1,1)
        1467 Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
        Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
    1469 end

    figure; pcolor(Z1);%title('Sample Stochastic Component');
    view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    1473 figure; pcolor(Z2);%title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
    view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    1475 figure; scatter(rf(:), cv_matr{i,1}.Z(:), 'filled', 'd'); hold on;
    dvec1 = [rf(:); cv_matr{i,1}.Z(:)];
    1477 plot([min(dvec1)-0.5,max(dvec1)+0.5], [min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
    axis([min(dvec1)-0.5,max(dvec1)+0.5, min(dvec1)-0.5,max(dvec1)+0.5])
    1479 %title('Scatter Plot')
    xlabel('Observed Data'); ylabel('Estimations');
    1481 figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
    hold on
    1483 histogram(cv_matr{i,1}.Z(:), 'BinEdges', h.BinEdges, 'EdgeColor',[0.2 1
0], 'FaceAlpha',0.7)
    %title('Histograms of Sample Stochastic Component')
    1485 legend({'Original', 'Estimated'});
    clear h
    1487

    % Total Data figures
    cc3 = [col,row, v1(:); qcol,qrow, zeros(Nu,1)];
    1491 cc4 = [col,row, cv_matr{i,1}.Z_ibt(:); qcol,qrow, zeros(Nu,1)];
    Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
    1493 Z4=Z3;
    for j = 1:size(cc3,1)

```

```

1495     Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
1496     Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
1497 end
1498 %     figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
1499 %     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick',[], 'YTick',[]);
1500     figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s', tit{i,1}));
1501 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
,[], 'YTick',[]);
1502     figure; scatter(v1(:), cv_matr{i,1}.Z_ibt(:), 'filled', 'd'); hold on;
1503 dvec2 = [v1(:); cv_matr{i,1}.Z_ibt(:)];
1504     plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
1505     axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
1506     %title('Scatter Plot')
1507     xlabel('Observed Data')
1508     ylabel('Estimations')
1509     figure; h = histogram(v1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
1510     hold on
1511     histogram(cv_matr{i,1}.Z_ibt(:), 'BinEdges', h.BinEdges, 'EdgeColor'
,[0.2 1 0], 'FaceAlpha', 0.7)
1512     %title('Histograms of Sample Data')
1513     legend({'Original', 'Estimated'});
1514     clear h
1515
1516     clear Z1 Z2 Z3 Z4 dvec1 dvec2
1517 end
1518
1519
1520
1521 % >>> Ordinary Kriging <<<
1522
1523 % Sort models based on cross validation scores
1524 [~,ind] = sort(table2array(cv_Stf(:,7)), 'descend');
1525
1526 % Matrices and Cells preallocation
1527 Z{N_kr_mod,1}=[]; Z_error{N_kr_mod,1}=[]; kr_checks{N_kr_mod,1}=[];
1528 kr_matr{N_kr_mod,1}=[];
1529 kr_Ss(N_kr_mod,6)= 0; table_r2{N_kr_mod,1} = [];
1530 CI1{N_kr_mod,1}=[]; UNC{N_kr_mod,1}=[];

```

```

1533 % Inputs definition
xu = qcol; yu = qrow; iso = 00;

1535 % Ordinary Kriging
1537 for i=1:N_kr_mod
    model.function = models{ind(i),1};
1539    model.params = bmodel{ind(i),1};
    model.r_ok = [22,4];

1541    [Z{i,1}, Z_error{i,1}, kr_checks{i,1}, kr_matr{i,1}] = ...
1543        ordkrig(x,y,rf,xu,yu,iso,model);
    kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
1545    table_r2{i,1} = table_r{ind(i),1};

    %Confidence Intervals (95%)
    CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
1549    CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
    CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
1551    UNC{i,1} = real(CI1{i,1}.uncer);
    realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>

1553 end

1555 % Kriging Scores
relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
1557 FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
kr_Ss = [kr_Ss, FinalScore];
1559 kr_Ssf = array2table(kr_Ss, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore

1561 %Trend addition and Boxcox Inversion
1563 kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
Z_ibt1{N_kr_mod,1} = 0; Z_ibt{N_kr_mod,1} = 0;
1565 for i=1:N_kr_mod

1567     Z_tr{i,1} = qMx + Z{i,1};

1569     if lambda==00
        Z_ibt1{i,1} = Z_tr{i,1};
1571     elseif lambda==0
        Z_ibt1{i,1} = exp(Z_tr{i,1});
1573     else
        Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
1575     end
    Z_ibt{i,1} = real(Z_ibt1{i,1});

```

```

1577     kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

1579     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores

1581 end

1583 % Total Kriging Scores
relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
1585 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
kr_St = [kr_St, FinalScore];
1587 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
clear relMSE FinalScore

1589 %Plots
1591 for i=1:N_kr_mod

1593     % Stochastic Component's figures
cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
1595 cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
1597 Z2=Z1;
for j = 1:size(cc1,1)
1599     Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
    Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1601 end
figure;pcolor(Z1);%title('Original Stochastic Component');
1603 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
Component \n%s',tit{ind(i),1}));
1605 view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
1607 dvec1 = [qfluc(:);Z{i,1}(:)];
plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5],'r');
1609 axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
%title('Scatter Plot')
1611 xlabel('Observed Data')
ylabel('Estimations')
1613 figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha',
,0.7);
hold on
1615 histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],',
FaceAlpha',0.7)
%title('Histograms of Stochastic Component')

```

```

1617     legend({'Original', 'Estimated'});
1618     clear h
1619
1620     % Total Data figures
1621     cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
1622     cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
1623     Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
1624     Z4=Z3;
1625     for j = 1:size(cc3,1)
1626         Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
1627         Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
1628     end
1629     figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
1630 Estimation of Original Data \n%s',tit{ind(i),1}));
1631 c = colorbar; c.Label.String = 'Normalised Velocity';set(gca,'XTick'
1632 ,[],'YTick',[]);
1633 figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
1634 dvec2 = [qv1(:);Z_ibt{i,1}(:)];
1635 plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
1636 +0.5],'r');
1637 axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
1638 %title('Scatter Plot')
1639 xlabel('Observed Data')
1640 ylabel('Estimations')
1641 figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
1642 hold on
1643 histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
1644 'FaceAlpha',0.7)
1645 %title('Histograms of Data')
1646 legend({'Original', 'Estimated'});
1647 figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
1648 Interval');
1649 c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
1650 XTick',[],'YTick',[]);
1651 clear h
1652
1653 clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2
1654
1655 end
1656
1657 % >>>Correlation Coefficient of Indicators<<<
1658 edges4 = (1500:4000/4:5500)/5500;
1659 [~,~,ind_data4] = histcounts(qv1,edges4);

```

```

ind4_est{N_kr_mod,1}=[]; Rpearson4(N_kr_mod,1)=0; Rspearman4(N_kr_mod,1)
    =0;
1655 MCR4(N_kr_mod,1)=0;

1657 edges16 = (1500:4000/16:5500)/5500;
[~,~,ind_data16] = histcounts(qv1,edges16);
1659 ind16_est{N_kr_mod,1}=[]; Rpearson16(N_kr_mod,1)=0; Rspearman16(N_kr_mod
    ,1)=0;
MCR16(N_kr_mod,1)=0;
1661
for i=1:N_kr_mod
1663     [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
    ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
1665     ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
    Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
1667     Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
    MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;
1669

1671     [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
    ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
1673     ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
    Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
1675     Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
    ;
    MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
1677
end
1679
% Kriging Indicators Scores Table
1681 table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    };
table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
    , 'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
    ,};
1683 kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
    ,table_r2);
kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
    Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
    table_r2);
1685 %=====
1687
%% ##### CHI2 #####

```



```

1689 %=====
1691 load('chil_iso.mat')
1693 % >>> Fitting II and Estimation of xi1 & xi2 <<<
1695 xi1 = R; xi2 = 1;
1697 % Desired Models
1698 models = {'Gexp','Gaus','Sphe','Mate','Spar'};
1699 n_models = length(models);
1701 % >>> Rescaling and Rotation of the random field
1703 % (Inverse) Transformation Matrix
1704 A = [cos(phi*pi/180)/xi1, sin(phi*pi/180)/xi1;
1705      -sin(phi*pi/180)/xi2, cos(phi*pi/180)/xi2];
1707
1708 % Rescale and Rotate Coordinations
1709 c_tr = A*[c{1,1}';c{1,2}']; %transformed coordinations of known points
1710 x_tr = c_tr(1,:);
1711 y_tr = c_tr(2,:);
1712 qc_tr = A*[qc{1,1}';qc{1,2}']; %transformed coordinations of unknown
1713           points
1714 qx_tr = qc_tr(1,:);
1715 qy_tr = qc_tr(2,:);
1717
1718 % >>> Check isotropy <<<
1719
1720 % Experimental Variogram (anisotropic)
1721 x = x_tr; y = y_tr; rf = fluc; iso = 0;
1722 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
1723           high analysis
1724 %[gexp_tr, nr_pairs_tr, c_centers_tr] = expvar(x,y,rf,iso,ncpc,nrbins,
1725           phistep,phitol,2);
1727
1728 %Initial Values and Limits for optimization
1729 b = [gexpmax,maxdist*1/3,gexpmax/100]; % s2, xi & c0
1730 b_lb = [eps,eps,eps]; b_ub = [gexpmax*1.5,maxdist,gexpmax/5]; %lower and
1731           upper limits
1732 bsp = [1000,maxdist*1/3,gexpmax/100]; % eta0, xi, c0
1733 bsp_lb = [eps,eps,eps]; bsp_ub = [inf,maxdist,gexpmax/5]; %lower and
1734           upper limits

```

```

1729 %Summary cells
model_par0 = {[b,1.5];b;b;[b,1.5];[bsp,1]}; %initial parameters values
1731 model_par_lb = {[b_lb,eps];b_lb;b_lb;[b_lb,0.3];[bsp_lb,-2+eps]}; %lower
        bounds
model_par_ub = {[b_ub,2-eps];b_ub;b_ub;[b_ub,3.5];[bsp_ub,inf]}; %upper
        bounds
1733
clear b b_lb b_ub bsp bsp_lb bsp_ub
1735
% Estimation of New Anisotropy
1737 R_tr(n_models,1)=0;phi_tr(n_models,1)=0;xi1_tr(n_models,1)=0;xi2_tr(
        n_models,1)=0;
for i=1:n_models
1739     model.function = models{i,1};
        model.params0 = model_par0{i,1};
1741     model.paramslb = model_par_lb{i,1};
        model.paramsub = model_par_ub{i,1};
1743     [R_tr(i,1),phi_tr(i,1),xi1_tr(i,1),xi2_tr(i,1),~,~] = ...
        aniso_dvf(c_tr(1,:),c_tr(2,:),rf,model,'NWEr_m',20,0.4,nrbins,
        phistep,phitol,0);
1745 end
1747
1749 % >>> Fitting III and Parameters of Correlation Estimation (s2,c0,v or
        etal) <<<
1751 % Experimental (Semi-)Variogram (isotropic)
x = x_tr; y = y_tr; rf = fluc; iso = 1; flag = 1;
1753 [~,~,~] = expvar(x,y,rf,iso,ncpc,nrbins,4,phitol,2); %exper. variogr. of
        high analysis
[gexp2, nr_pairs2, c_centers2] = expvar(x,y,rf,iso,ncpc,nrbins,phistep,
        phitol,flag);
1755 gexpmax2 = max(max(gexp2));
1757 % Initial Values and Limits for optimization
[k,1] = find(triu(true(N)));
1759 d = hypot(c_tr(1,k)-c_tr(1,1),c_tr(2,k)-c_tr(2,1));
maxdist2(i,1) = ncpc*max(max(d));
1761 % maxdist2 = hypot(c_tr(1,1)-c_tr(1,N),c_tr(2,1)-c_tr(2,N))*ncpc;
1763 b = [gexpmax2,maxdist2*2/3,gexpmax2/100]; % s2,xi & c0

```

```

b_lb = [eps, eps, eps]; b_ub = [gexpmax2*1.5, maxdist2*1.5, gexpmax2/5]; %
    lower and upper limits
1765 bsp = [1000, maxdist2*2/3, gexpmax2/100]; % eta0, xi, c0
    bsp_lb = [eps, eps, eps]; bsp_ub = [inf, maxdist2*1.5, gexpmax2/5]; %lower
        and upper limits
1767
% Summary cells
1769 model_par0 = {[b, 1.5]; b; b; [b, 1.5]; [bsp, 1]}; %initial parameters values
    model_par_lb = {[b_lb, eps]; b_lb; b_lb; [b_lb, 0.3]; [bsp_lb, -2+eps]}; %lower
        bounds
1771 model_par_ub = {[b_ub, 2-eps]; b_ub; b_ub; [b_ub, 3.5]; [bsp_ub, inf]}; %upper
        bounds

1773 clear b b_lb b_ub bsp bsp_lb bsp_ub

1775 % Estimation of Parameters (s2, c0, v or etal)
    bmodel2{n_models, 1}=[]; fval2(n_models, 1)=0; tit2{n_models, 1}=[];
1777 iso = 1; objmod = 'NWEr_m'; flag = 1;
    for i=1:n_models
1779         model.function = models{i, 1};
            model.params0 = model_par0{i, 1};
1781         model.paramslb = model_par_lb{i, 1};
            model.paramsub = model_par_ub{i, 1};
1783
            [bmodel2{i, 1}, fval2(i, 1), tit2{i, 1}] = ...
1785                 variogramfit(gexp2, nr_pairs2, c_centers2, iso, model, objmod, flag);
    end
1787

1789 % >>> Cross Validation <<<

1791 % Matrices and Cells preallocation
    cv_scores{n_models, 1}=[]; cv_checks{n_models, 1}=[];
1793 cv_matr{n_models, 1}=[]; cv_Ss(n_models, 6)= 0;

1795 % Inputs definition
    x = x_tr; y = y_tr; iso = 1; d_col = 1;
1797
    rx_tr = pdist2(x_tr(:), x_tr(:));
1799 ry_tr = pdist2(y_tr(:), y_tr(:));

1801 % Cross Validation
    for i=1:n_models
1803         model.function = models{i, 1};

```

```

1805     model.params = bmodel2{i,1};
1806     model.r_ok = [mean(rx_tr(:))/5, mean(ry_tr(:))/5];

1807     [cv_scores{i,1}, cv_checks{i,1}, cv_matr{i,1}] = ...
        crossval(x,y,rf,iso,d_col,model);
1809     cv_Ss(i,:) = table2array(cv_scores{i,1}(:,2:end));
end

1811 % Cross Validation Scores
1812 table_h = {'MeanAbsErr', 'MaxAbsErr', 'MSE', 'RMSE', 'rpearson', 'rspearman',
        'finalscore'};
1813 table_r = {'Gexp'; 'Gaus'; 'Sphe'; 'Mate'; 'Spar'};

1815 relMSE = cv_Ss(:,3)/min(cv_Ss(:,3)); %relative MSE
1817 FinalScore = ((1./relMSE).^2).*cv_Ss(:,5).*cv_Ss(:,6); %finalscore
cv_Ss = [cv_Ss, FinalScore];
1819 cv_Ssf = array2table(cv_Ss, 'VariableNames', table_h, 'RowNames', table_r);
clear relMSE FinalScore

1821 %Trend addition and Boxcox Inversion
1823 cv_St(n_models,6)= 0;
for i=1:n_models

1825     cv_matr{i,1}.Z_tr = Mx + cv_matr{i,1}.Z;

1827     if lambda==00
1829         cv_matr{i,1}.Z_ibt1 = cv_matr{i,1}.Z_tr;
1831     elseif lambda==0
        cv_matr{i,1}.Z_ibt1 = exp(cv_matr{i,1}.Z_tr);
1833     else
        cv_matr{i,1}.Z_ibt1 = (lambda*cv_matr{i,1}.Z_tr + 1).^(1/lambda)
;
1835     end
    cv_matr{i,1}.Z_ibt = real(cv_matr{i,1}.Z_ibt1);
    cv_realZ(i,1) = isreal(cv_matr{i,1}.Z_ibt1); %ok<SAGROW>

1837     cv_St(i,:) = correlstats(v1,cv_matr{i,1}.Z_ibt);%total cv scores

1839 end

1841 % Total Cross Validation Scores
1843 relMSE = cv_St(:,3)/min(cv_St(:,3)); %relative MSE
FinalScore = ((1./relMSE).^2).*cv_St(:,5).*cv_St(:,6); %finalscore
1845 cv_St = [cv_St, FinalScore];

```

```

cv_Stf = array2table(cv_St, 'VariableNames', table_h, 'RowNames', table_r);
1847 clear relMSE FinalScore

1849 %Plots
for i=1:n_models
1851
    % Stochastic Component's figures
1853 cc1 = [col,row, rf(:); qcol,qrow, zeros(Nu,1)];
    cc2 = [col,row, cv_matr{i,1}.Z(:); qcol,qrow, zeros(Nu,1)];
1855 Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %#ok<SAGROW>
    Z2=Z1;
1857 for j = 1:size(cc1,1)
        Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
1859 Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
    end

1861
    figure; pcolor(Z1); %title('Sample Stochastic Component');
1863 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    figure; pcolor(Z2); %title(sprintf('Estimation of Sample Stochastic
Component \n%s', tit{i,1}));
1865 view(2); shading interp; colorbar; set(gca, 'XTick', [], 'YTick', []);
    figure; scatter(rf(:), cv_matr{i,1}.Z(:), 'filled', 'd'); hold on;
1867 dvec1 = [rf(:); cv_matr{i,1}.Z(:)];
    plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
+0.5], 'r');
1869 axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
    %title('Scatter Plot')
1871 xlabel('Observed Data'); ylabel('Estimations');
    figure; h = histogram(rf(:),16,'EdgeColor',[0 0 1],'FaceAlpha',0.7);
1873 hold on
    histogram(cv_matr{i,1}.Z(:), 'BinEdges', h.BinEdges, 'EdgeColor',[0.2 1
0], 'FaceAlpha',0.7)
1875 %title('Histograms of Sample Stochastic Component')
    legend({'Original', 'Estimated'});
1877 clear h

1879
    % Total Data figures
1881 cc3 = [col,row, v1(:); qcol,qrow, zeros(Nu,1)];
    cc4 = [col,row, cv_matr{i,1}.Z_ibt(:); qcol,qrow, zeros(Nu,1)];
1883 Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
    Z4=Z3;
1885 for j = 1:size(cc3,1)
        Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);

```

```

1887     Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
1888     end
1889 %     figure; pcolor(Z3); view(2); shading interp; %title('Original Sample
1890 %         ');
1891 %     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, '
XTick', [], 'YTick', []);
1891     figure; pcolor(Z4); view(2); shading interp; %title(sprintf('Estimation
of Original Sample \n%s', tit{i,1}));
1892     c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca, 'XTick'
, [], 'YTick', []);
1893     figure; scatter(v1(:), cv_matr{i,1}.Z_ibt(:), 'filled', 'd'); hold on;
1894     dvec2 = [v1(:); cv_matr{i,1}.Z_ibt(:)];
1895     plot([min(dvec2)-0.5, max(dvec2)+0.5], [min(dvec2)-0.5, max(dvec2)
+0.5], 'r');
1896     axis([min(dvec2)-0.5, max(dvec2)+0.5, min(dvec2)-0.5, max(dvec2)+0.5])
1897     %title('Scatter Plot')
1898     xlabel('Observed Data')
1899     ylabel('Estimations')
1900     figure; h = histogram(v1(:), 16, 'EdgeColor', [0 0 1], 'FaceAlpha', 0.7);
1901     hold on
1902     histogram(cv_matr{i,1}.Z_ibt(:), 'BinEdges', h.BinEdges, 'EdgeColor'
, [0.2 1 0], 'FaceAlpha', 0.7)
1903     %title('Histograms of Sample Data')
1904     legend({'Original', 'Estimated'});
1905     clear h
1906
1907     clear Z1 Z2 Z3 Z4 dvec1 dvec2
1908
1909 end
1910
1911
1912 % >>> Ordinary Kriging <<<
1913
1914 % Sort models based on cross validation scores
1915 [~, ind] = sort(table2array(cv_Stf(:,7)), 'descend');
1916
1917 % Matrices and Cells preallocation
1918 Z{N_kr_mod,1} = []; Z_error{N_kr_mod,1} = []; kr_checks{N_kr_mod,1} = [];
1919 kr_matr{N_kr_mod,1} = [];
1920 kr_Ss(N_kr_mod,6) = 0; table_r2{N_kr_mod,1} = [];
1921 CI1{N_kr_mod,1} = []; UNC{N_kr_mod,1} = [];
1922
1923 % Inputs definition

```

```

1925 x = x_tr; y = y_tr; xu = qx_tr; yu = qy_tr; iso=1;

1927 % Ordinary Kriging
for i=1:N_kr_mod
1929     model.function = models{ind(i),1};
        model.params = bmodel2{ind(i),1};
1931     model.r_ok = [mean(rx_tr(:))/5,mean(ry_tr(:))/5];

        [Z{i,1},Z_error{i,1},kr_checks{i,1}, kr_matr{i,1}] =...
            ordkrig(x,y,rf,xu,yu,iso,model);
1935     kr_Ss(i,:) = correlstats(Z{i,1},qfluc);
        table_r2{i,1} = table_r{ind(i),1};

1937     %Confidence Intervals (95%)
1939     CI1{i,1}.low = Z{i,1} - 1.96*sqrt(Z_error{i,1});
        CI1{i,1}.up = Z{i,1} + 1.96*sqrt(Z_error{i,1});
1941     CI1{i,1}.uncer = 1.96*sqrt(Z_error{i,1});
        UNC{i,1} = real(CI1{i,1}.uncer);
1943     realCI(i,1) = isreal(CI1{i,1}.uncer); %#ok<SAGROW>
end

1945 % Kriging Scores
1947 relMSE = kr_Ss(:,3)/min(kr_Ss(:,3)); %relative MSE
        FinalScore = ((1./relMSE).^2).*kr_Ss(:,5).*kr_Ss(:,6); %finalscore
1949 kr_Ss = [kr_Ss, FinalScore];
        kr_Ssf = array2table(kr_Ss,'VariableNames',table_h,'RowNames',table_r2);
1951 clear relMSE FinalScore

1953 %Trend addition and Boxcox Inversion
        kr_St(N_kr_mod,6)= 0; Z_tr{N_kr_mod,1} = 0;
1955 Z_ibt1{N_kr_mod,1} = 0;Z_ibt{N_kr_mod,1} = 0;
        for i=1:N_kr_mod

1957             Z_tr{i,1} = qMx + Z{i,1};

1959             if lambda==00
                    Z_ibt1{i,1} = Z_tr{i,1};
1961             elseif lambda==0
                    Z_ibt1{i,1} = exp(Z_tr{i,1});
1963             else
                    Z_ibt1{i,1} = (lambda*Z_tr{i,1} + 1).^(1/lambda);
1965             end
1967             Z_ibt{i,1} = real(Z_ibt1{i,1});
            kr_realZ(i,1) = isreal(Z_ibt1{i,1}); %#ok<SAGROW>

```

```

1969     kr_St(i,:) = correlstats(qv,Z_ibt{i,1});%total kriging scores
1971
1973 end
1975 % Total Kriging Scores
1975 relMSE = kr_St(:,3)/min(kr_St(:,3)); %relative MSE
1975 FinalScore = ((1./relMSE).^2).*kr_St(:,5).*kr_St(:,6); %finalscore
1977 kr_St = [kr_St, FinalScore];
1977 kr_Stf = array2table(kr_St, 'VariableNames', table_h, 'RowNames', table_r2);
1979 clear relMSE FinalScore
1981 %Plots
1981 for i=1:N_kr_mod
1983
1983     % Stochastic Component's figures
1985     cc1 = [col,row,rf(:);qcol,qrow,qfluc(:)];
1985     cc2 = [col,row,rf(:);qcol,qrow,Z{i,1}];
1987     Z1(max(cc1(:,2)),max(cc1(:,1)))=0; %ok<SAGROW>
1987     Z2=Z1;
1989     for j = 1:size(cc1,1)
1989         Z1(cc1(j,2),cc1(j,1)) = cc1(j,3);
1991         Z2(cc2(j,2),cc2(j,1)) = cc2(j,3);
1991     end
1993     figure;pcolor(Z1);%title('Original Stochastic Component');
1993     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
1995     figure;pcolor(Z2); %title(sprintf('Estimation of Stochastic
1995     Component \n%s',tit{ind(i),1}));
1997     view(2);shading interp; colorbar;set(gca,'XTick',[],'YTick',[]);
1997     figure;scatter(qfluc(:),Z{i,1}(:),'filled','d');hold on;
1999     dvec1 = [qfluc(:);Z{i,1}(:)];
1999     plot([min(dvec1)-0.5,max(dvec1)+0.5],[min(dvec1)-0.5,max(dvec1)
1999     +0.5],'r');
2001     axis([min(dvec1)-0.5,max(dvec1)+0.5,min(dvec1)-0.5,max(dvec1)+0.5])
2001     %title('Scatter Plot')
2001     xlabel('Observed Data')
2003     ylabel('Estimations')
2003     figure;h = histogram(qfluc(:),16,'FaceColor',[0 0 1],'FaceAlpha'
2003     ,0.7);
2005     hold on
2005     histogram(Z{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
2005     'FaceAlpha',0.7)
2007     %title('Histograms of Stochastic Component')
2007     legend({'Original', 'Estimated'});

```



```

2009     clear h

2011 % Total Data figures
cc3 = [col,row,v1(:);qcol,qrow,Z_ibt{i,1}];
2013 cc4 = [col,row,zeros(size(col,1),1); qcol,qrow,UNC{i,1}];
Z3(max(cc3(:,2)),max(cc3(:,1)))=0; %#ok<SAGROW>
2015 Z4=Z3;
for j = 1:size(cc3,1)
2017     Z3(cc3(j,2),cc3(j,1)) = cc3(j,3);
    Z4(cc4(j,2),cc4(j,1)) = cc4(j,3);
2019 end
figure;pcolor(Z3); view(2);shading interp; %title(sprintf('
Estimation of Original Data \n%s',tit{ind(i),1}));
2021 c = colorbar; c.Label.String = 'Normalised Velocity'; set(gca,'XTick'
,[],'YTick',[]);
figure;scatter(qv1(:),Z_ibt{i,1}(:),'filled','d');hold on;
2023 dvec2 = [qv1(:);Z_ibt{i,1}(:)];
plot([min(dvec2)-0.5,max(dvec2)+0.5],[min(dvec2)-0.5,max(dvec2)
+0.5],'r');
2025 axis([min(dvec2)-0.5,max(dvec2)+0.5,min(dvec2)-0.5,max(dvec2)+0.5])
%title('Scatter Plot')
2027 xlabel('Observed Data')
ylabel('Estimations')
2029 figure;h = histogram(qv1(:),16,'FaceColor',[0 0 1],'FaceAlpha',0.7);
hold on
2031 histogram(Z_ibt{i,1}(:),'BinEdges',h.BinEdges,'FaceColor',[0.2 1 0],
'FaceAlpha',0.7)
%title('Histograms of Data')
2033 legend({'Original','Estimated'});
figure;pcolor(Z4); view(2);shading interp; %title('95% Confidence
Interval');
2035 c1 = colorbar; c1.Label.String = 'Normalised Velocity'; set(gca,'
XTick',[],'YTick',[]);
clear h

2037

2039 clear cc1 cc2 cc3 cc4 Z1 Z2 Z3 Z4 dvec1 dvec2

2041 end

2043 % >>>Correlation Coefficient of Indicators<<<
edges4 = (1500:4000/4:5500)/5500;
[~,~,ind_data4] = histcounts(qv1,edges4);
2045 ind4_est{N_kr_mod,1}=[];Rpearson4(N_kr_mod,1)=0;Rspearman4(N_kr_mod,1)
=0;

```

```

MCR4(N_kr_mod,1)=0;
2047
edges16 = (1500:4000/16:5500)/5500;
2049 [~,~,ind_data16] = histcounts(qv1,edges16);
ind16_est{N_kr_mod,1}=[];Rpearson16(N_kr_mod,1)=0;Rspearman16(N_kr_mod
,1)=0;
2051 MCR16(N_kr_mod,1)=0;

2053 for i=1:N_kr_mod
    [~,~,ind4_est{i,1}] = histcounts(Z_ibt{i,1},edges4);
2055 ind4_est{i,1}(Z_ibt{i,1}<=edges4(1)) = 1;
ind4_est{i,1}(Z_ibt{i,1}>=edges4(end)) = length(edges4)-1;
2057 Rpearson4(i,1) = corr(ind4_est{i,1},ind_data4);
Rspearman4(i,1) = corr(ind4_est{i,1},ind_data4,'type','Spearman');
2059 MCR4(i,1) = sum(ind4_est{i,1}~=ind_data4)/Nu;

2061 [~,~,ind16_est{i,1}] = histcounts(Z_ibt{i,1},edges16);
ind16_est{i,1}(Z_ibt{i,1}<=edges16(1)) = 1;
2063 ind16_est{i,1}(Z_ibt{i,1}>=edges16(end)) = length(edges16)-1;
Rpearson16(i,1) = corr(ind16_est{i,1},ind_data16);
2065 Rspearman16(i,1) = corr(ind16_est{i,1},ind_data16,'type','Spearman')
;
MCR16(i,1) = sum(ind16_est{i,1}~=ind_data16)/Nu;
2067 end

2069 % Kriging Indicators Scores Table
table_h = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
};
2071 table_h2 = {'MeanAbsErr','MaxAbsErr','MSE','RMSE','Rpearson','Rspearman'
,'Rpearson4','Rspearman4','MCR4','Rpearson16','Rspearman16','MCR16'
,};
kr_iSsf = array2table(kr_Ss(:,1:end-1),'VariableNames',table_h,'RowNames'
',table_r2);
2073 kr_iStf = array2table([kr_St(:,1:end-1),Rpearson4,Rspearman4,MCR4,
Rpearson16,Rspearman16,MCR16],'VariableNames',table_h2,'RowNames',
table_r2);

%
=====

```