TECHNICAL UNIVERSITY OF CRETE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Analysis of Functional Magnetic Resonance Imaging Data on the Fisher-Shannon Information Plane

by

Ioannis Karvounakis

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DIPLOMA OF

ELECTRICAL AND COMPUTER ENGINEERING

2024

THESIS COMMITTEE

Professor Dionissios Hristopulos, *Supervisor*
Professor Athanasios P. Liavas
Professor Michalis Zervakis

Περίληψη

Το Επίπεδο Πληροφορίας Fisher-Shannon (Fisher Shannon Information Plane, εν συντομία FSIP) είναι ένα αποτελεσματικό εργαλείο για την ανάλυση και διαχωρισμό δεδομένων χρονοσειρών, το οποίο βασίζεται σε δύο μόνο παραμέτρους: το μέτρο πληροφορίας κατά Fisher (Fisher Information Measure–FIM) και την εντροπική ισχύ κατά Shannon (Shannon Entropy Power–SEP). Το FSIP αποδεικνύεται ιδιαίτερα χρήσιμο στη διάκριση χρονοσειρών και την κατηγοριοποίηση τους σε διαφορετικούς τύπους σημάτων, παρέχοντας ένα ισχυρό εργαλείο για την διερεύνηση των δεδομένων από διάφορα επιστημονικά πεδία . Σε αυτή τη διπλωματική εργασία εξετάζεται η εφαρμογή του FSIP σε τυχαία δεδομένα προερχόμενα από διάφορες κατανομές πιθανότητας, όπως Gaussian, Power exponential, Student-t, Gamma, Weibull, Log-Normal και Uniform κατανομές, με έμφαση στο πώς οι παράμετροι κλίμακας (scale) και σχήματος (shape) της κατανομής επηρεάζουν την αναπαράσταση των δεδομένων στο Επίπεδο Πληροφορίας Fisher-Shannon . Η μεθοδολογία για τον υπολογισμό του FSIP περιλαμβάνει την εκτίμηση της συνάρτησης πυκνότητας πιθανότητας (probability density function–PDF) από τα δεδομένα μέσω συναρτήσεων πυρήνα και εφαρμογή της εκτίμησης πυκνότητας μέσω του πυρήνα (Kernel Density Estimator–KDE). Η εκτίμηση KDE μπορεί να λειτουργήσει αποτελεσματικά ακόμα και για χρονοσειρές με πολύπλοκη δομή. Η ανάλυση των δεδομένων λειτουργικής απεικόνισης μαγνητικού συντονισμού (Functional Magnetic Resonance Imaging –fMRI), τα οποία χαρακτηρίζονται από μεγάλους όγκους περίπλοκων σημάτων χρονοσειρών, παρουσιάζει σημαντικές προκλήσεις. Το fMRI, λόγω της περίπλοκης δυναμικής των χρονοσειρών που το διέπουν, δίνει χώρο για την εφαρμογή του FSIP και αντιμετωπίζεται ως μελέτη περίπτωσης (case study). Η διπλωματική εργασία διερευνά την απόκριση αντίθεσης εξαρτώμενης από το επίπεδο οξυγόνου στο αίμα (Blood Oxygen Level Dependent –BOLD), συνθετικών και πραγματικών δεδομένων fMRI, με εφαρμογή της μεθοδολογίας FSIP για την αναγνώριση προτύπων και την ταξινόμηση χρονοσειρών εγκεφαλικής δραστηριότητας. Τα αποτελέσματα αναδεικνύουν την ευελιξία του FSIP, και την προοπτική του ως αξιόπιστη μέθοδος για τη διάκριση πολύπλοκων σημάτων σε βιοϊατρικές μελέτες καθώς και σε άλλους τομείς.

Abstract

The Fisher Shannon Information Plane (FSIP) represents a novel and compact framework for analyzing and distinguishing time series data. FSIP is based on only two parameters: the Fisher Information Measure (FIM) and Shannon Entropy Power (SEP). FSIP proves particularly useful in differentiating time series and categorizing them into distinct signal types, providing a powerful tool for data exploration. This thesis investigates FSIP's application to data derived from a variety of probability distributions, including the Normal, Power Exponential, Student-t, Gamma, Weibull, Log-Normal, and Uniform models, focusing on how scale and shape parameters influence the FSIP representation. The methodology for FSIP estimation involves estimating the Probability Density Function (PDF) of a time series by means of Kernel Density Estimation (KDE), enabling precise computation of FIM and SEP. KDE can be effectively applied to time series with complex correlations. Functional Magnetic Resonance Imaging (fMRI) provides a compelling context for applying FSIP. However, in this thesis FSIP application to fMRI data is treated primarily as a case study. The thesis explores the analysis of Blood Oxygen Level Dependent (BOLD) response for synthetic and real fMRI data, using FSIP to identify patterns and classify signals of brain activity time series. The findings highlight FSIP's versatility and its potential as a reliable method for distinguishing intricate signal patterns in biomedical data and other fields.
Thesis Supervisor: Professor Dionissios Hristopulos

Acknowledgments

First of all, I sincerely thank my thesis supervisor, Professor Dionissios Hristopulos, for his exceptional guidance, valuable advice, and constant support throughout this project.

Also, I want to thank my friends Giorgos A., Iosif V., Manolis P., Marietta S., Roza C. for the memories we shared all these years, but also for their support on tough periods.

A special thanks to people outside of my academic life but always by my side, Kostas G., Rafail G., Nikos G. I would also like to thank Melina M. and Natalia M. for our incredible scientific conversations.

Last but not least, I would like to thank my family for their support in all aspects throughout my academic years.

Contents

Appendices

List of Abbreviations

FSIP Fisher Shannon Information Plane
FSC Fisher Shannon Complexity
FIM Fisher Information Measure
HR Haemodynamic Response
SEP Shannon Entropy Power
MRI Magnetic Resonance Imaging
MR Magnetic Resonance
sMRI Structural Magnetic Resonance Imaging
fMRI Functional Magnetic Resonance Imaging
KDE Kernel Density Estimation
PET Positron Emission Tomography
CT Computed Tomography
PDF Probability Density Function
CDF Cumulative Distribution Function
PPF Percent Point Function
HF Hazard Function
CHF Cumulative Hazard Function
SF Survival Function
ISF Inverse Survival Function
DF Degrees of Freedom
GGD Generalized Gaussian Distribution
PED Power Exponential Distribution
CSF Cerebrospinal Fluid
TR Repetition Time
RF Radiofrequency
FID Free Induction Decay
BOLD Blood Oxygen Level Dependent
T Tesla
s seconds
SNR Signal-to-Noise Ratio

FWHM Full Width at Half Maximum
dHb Deoxyhemoglobin
m minutes
min minutes
sec seconds
ms milliseconds
EEG Electroencephalography
HRF Haemodynamic Response Function
FS Fisher-Shannon
FIR Finite Impulse Response
UGHH University General Hospital of Heraklion

List of Figures

Chapter 1

Introduction

1.1 Motivation

In that section, we are going to describe the main reason that gave us the motivation for this work. Today, biomedical topics are very challenging. One of the most important such topics is functional Magnetic Resonance Imaging (MRI). fMRI is a non-invasive method of studying the human brain. From the results of this method, we can study the functionality of the brain and rule out some serious conclusions.

Since the output of the fMRI method is a huge number of measured signals that are also called time series, which in most cases are complex, we want to find a method to interpret them effectively.

Through the acquisition of those signals' measurements, we have to face many issues like artifacts, lack of accuracy in measurement, move-of-the-subject oriented problems, etc.

Trying to answer the issues that come with the fMRI method, we believe that the so-called Fisher-Shannon method could help us in the analysis of the fMRI data. From the Fisher-Shannon method and by using the Fisher Information Measure and Shannon Entropy Power we can use a tool called Fisher Fisher-Shannon Information Plane (FSIP) in the analysis of the desired data. Although the FSIP method has been used in neuroscience, biophysics, geophysics, finance, or other science fields, the main characteristic in all of them is the complex systems that govern them.

The Fisher-Shannon Information Plane (FSIP) methodology opens up new possibilities to distinguish between different types of signal by analyzing the unique statistical properties of their probability distributions.

- What can we learn by applying FSIP to various kinds of signal, and could this approach unlock new ways to classify them more accurately?

- What differences emerge when FSIP is applied to actual data compared to synthetic models, and what role do the parameters within probabilistic models play in shaping these outcomes?

- The Gaussian limit introduces an interesting aspect to consider: How might this boundary influence signal classification within FSIP?

- Is it possible that a relatively compact tool like the FSIP, with its limited set of parameters, could successfully characterize such a wide range of distributions and signal types?

By exploring these questions, this research aims to demonstrate FSIP's potential to improve classification and differentiation between signal types, offering valuable information for both practical applications and theoretical exploration.

1.2 Related Work

Functional Magnetic Resonance Imaging (fMRI) studies have become increasingly popular for clinicians and researchers as they are capable of providing unique insights into brain functionality. Since neural activity is a dynamic system with many factors that affect it, there is a need for a tool made for the interpretation of such systems. This tool can be the Fisher Shannon Information Plane, which comes from FIM and SEP. Shannon Entropy and the Fisher Information Measure of the probability distributions are becoming increasingly important tools of scientific analysis in a variety of disciplines of scientific inquiry.

Over the past 30 years, the study of the human mind has benefited significantly from innovations in magnetic resonance imaging, in particular the development of techniques to detect physiological markers of neural activity. The most widely used of these techniques exploits the changes in blood flow and oxygenation associated with neural activity (hemodynamic response) and the different magnetic properties of oxygenated and deoxygenated blood. The paramagnetic properties of de-oxyhemoglobin (Deoxyhemoglobin (dHb)) create local field inhomogeneities, leading to reduced transverse relaxation times (T2) and, as a result, a reduction in image intensity. In contrast, increased oxyhemoglobin concentrations produce increased T2 relaxation times and a relative increase in image intensity. This Blood Oxygen Level-Dependent contrast mechanism (BOLD) is the basis for functional magnetic resonance imaging. ([2], [3], [4])

Although the idea of a hemodynamic response detected spatially in areas of neural activity dates back to Roy and Sherington [5], the mechanisms by which neural activity causes changes in cerebral blood volume, flow, and oxygenation are not yet fully understood (e.g., [6], [7]), which poses a considerable constraint on the interpretation of fMRI studies of cognition ([8], [9]). At the measurement level, the extent to which changes in the BOLD signal are colocalized with neural activity depends on several imaging aspects, including the magnetic field strength and the imaging sequence used ([10], [9]). At a more fundamental physiological level, it is also not clear which aspects of neural activity are most closely related to the hemodynamic response (see [9], for an excellent review). Although activation of excitatory neurons has been shown to cause changes in local blood flow ([11]), other studies have reported that input and activity within local neuronal circuits are better indicators of the hemodynamic response than output from pyramidal cells (e.g. [6], [12]), indicating that the hemodynamic response does not simply reflect the level of peak activity. Indeed, in some cases, the hemodynamic response has been observed in the absence of radial output ([6], [13], [12]). A further complication is that both excitatory and inhibitory inputs generate metabolic demands ([14]), making it even more difficult to explain the simultaneous vascular response as a pure measure of the firing rate of the neurons ([9]; although in some cases neuronal inhibition can cause metabolic and hemodynamic decline; [15]). In sum, the hemodynamic response is believed to reflect an average response to a range of metabolic demands imposed by neural activity, including excitatory and inhibitory postsynaptic processing, neuronal spiking, and neuromodulation ([9], [16]).

Fisher-Shannon analysis is used widely by many studies in fields such as wind, magnetotelluric data, seismograms, vegetation analysis, the analysis of drought/wetness episodes, and generally in fields of geoscience and geostatistics with pretty encouraging results. This type of analysis can find application in complex time series, linear or not, from which one can obtain a complexity measure and finally be an effective and efficient data exploration tool.

1.3 Thesis Contribution

This thesis makes significant contributions to the field of functional magnetic resonance imaging (fMRI) analysis by introducing and validating the Fisher-Shannon Information Plane (FSIP) as an innovative method for interpreting complex neural data. Traditional fMRI analysis techniques, while useful, often fall short of addressing the inherent complexities of time series data generated from brain imaging. These challenges include managing artifacts, handling measurement inaccuracies, and mitigating issues related to subject movement. To overcome these obstacles, this research leverages the Fisher Information Measure (FIM) and Shannon Entropy Power (SEP) to create the FSIP, a novel tool previously underutilized in biomedical applications but widely recognized in geostatistics.

The primary contribution of this work lies in adapting and applying the FSIP framework to fMRI data, demonstrating its potential to enhance the interpretability and precision of brain activity analysis. By rigorously testing FSIP across various probability distribution models -such as Normal, Student-t, Gamma, Weibull, and Log-Normal distributions - the thesis provides a comprehensive evaluation of its effectiveness in handling both linear and nonlinear time series data. This analysis not only broadens the application of FSIP but also underscores its versatility as a data exploration tool, capable of revealing nuanced patterns in neural activity that traditional methods might overlook.

Another key contribution is the detailed investigation of the expected fMRI response through various haemodynamic response functions. This exploration provides insights into how different models, including the Poisson, Gamma-variate, and Gaussian functions, can be used to better understand the complexities of the BOLD signal. The application of FSIP to both synthetic and real fMRI data further validates its utility, showing how it can be used to differentiate between complex signal structures and improve the overall quality of fMRI data interpretation.

In general, this thesis contributes to the growing body of knowledge in neuroimaging by offering a robust and innovative approach to fMRI data analysis. It opens up new avenues for research, particularly in the development of more sophisticated tools for understanding brain functionality, and sets the stage for future work in integrating FSIP with other advanced analytical techniques in biomedical research.

1.4 Outline

In **Chap. 2**, the foundational concepts of time series and fMRI are introduced, including their definitions, components, and the fMRI data acquisition process. In **Chap. 3** , the expected fMRI response is explored, with a detailed discussion of the HRF and its mathematical modeling using functions like Poisson, gamma-variate, and Gaussian. In **Chap. 4** , the methodological framework for the estimation of FIM and SEP is established, along with the development of the FSIP and its components. In **Chap. 5** , FSIP is applied to various probability distribution models, ranging from normal and Student-t to power exponential and gamma distributions, to study their information-theoretic characteristics. In **Chap. 6** , numerical investigations are conducted using FSIP on randomly distributed numbers, synthetic fMRI data, and real fMRI datasets, leading to insights into its applicability and robustness. Finally, in **Chap. 7** , the study's findings are summarized, and future research directions are proposed to extend FSIP applications to broader contexts.

Chapter 2

Background

2.1 Times Series

2.1.1 Definition

A sequence of data points that occur in successive time points can be called a time series. Any variable that changes over time can give us a time series. There are no restrictions on the time interval and/or the time series length.

We assume that the observed magnitude is the variable X. The time series consists of a set of values

$$\{x_1, x_2, \cdots, x_N\}$$

in different times

$$\{t_1, t_2, \cdots, t_N\}$$

. These different times are arranged in proper chronological order. At any time t the value x_t of the time series can be considered as the implementation of the random variable X_t.

If records of a time series refers to only a variable, it is termed as univariate. On the other hand, if records of a time series refer to more than one variable, it is termed multivariate. Time series can also be divided on continuous, when observations are measured in every infinitesimal instance of time and discrete when observations are measured on discrete time points. For instance, concentration of a chemical process can be recorded as continuous time series but population of a city can be recorded as discrete time series. In case of discrete time series the time interval usually is the same for all measures, such us hourly, daily, monthly etc. If a variable being observed in a time series that is discrete, is assumed to be measured as a continuous time series with the proper real number scaling. [18] Also a continuous time series can be easily converted to a discrete one, using a proper time interval.

2.1.2 Components

Time series affected by four main components that can be separated from the data. These components are shown below and will be briefly described.

- *Trend*, also known as the Secular Trend, refers to the general direction in which a time series moves whether it is increasing, decreasing, or remaining steady over an extended period. This trend represents the long-term behavior of the data. For example, population growth and the number of houses in a city typically show an upward trend, while a downward trend might be seen in data related to mortality rates or epidemics.

- *Cyclical* variations describe the medium-term fluctuations that occur in cycles, usually lasting two or more years. These cycles are often observed in economic and financial data, with the business cycle being a typical example. The business cycle includes four phases: Prosperity, Decline, Depression, and Recovery.

- *Seasonal* variations refer to fluctuations within a single year that occur during specific seasons. Factors such as climate, weather, customs, and traditional habits play a significant role in these variations. Examples include increased sales of ice cream during summer and wool clothing during winter. Understanding seasonal variations is crucial for business owners, retailers, and producers as they plan for the future.

- *Irregular* or random variations are caused by unexpected events that do not follow any regular pattern. These variations arise from unpredictable occurrences such as wars, strikes, earthquakes, floods, or revolutions. Since these fluctuations are random, there is no specific statistical method to measure them.

Considering the influence of these four components, two primary types of model are typically employed in time series analysis: the multiplicative model and the additive model.

Multiplicative Model:

$$(Y(t) = T(t) \times S(t) \times C(t) \times I(t)) \tag{2.1}$$

Additive Model:

$$(Y(t) = T(t) + S(t) + C(t) + I(t)) \tag{2.2}$$

In this context, $Y(t)$ represents the observed value, while $T(t)$, $S(t)$, $C(t)$, and $I(t)$ correspond to the trend, seasonal, cyclical, and irregular variations at time t, respectively. The multiplicative model assumes that the four components of a time series are interdependent and can influence each other. In contrast, the additive model assumes that these four components are independent and do not affect each other.

2.1.3 Applications

Time series can be widely used in many domains such as economics, industry, engineering, biomedicals, science, and the study of many natural phenomena. One of the most interesting feature of time series can be found in forecasting. Given specific data of one measured magnitude, a suitable model can be fitted by computing the specific parameters estimated using the real data values. This kind of procedure, in order to find the proper model is termed as Time Series Analysis. After that, using the model on given data, future events can be predicted. However, time series are not deterministic in nature. So predictions are not characterized by certainty. Uncertainty introduces the concept of possibility. Time series is assumed to follow a certain probability model describing the joint distribution of the random variable. The probability structure of time series described by a mathematical model, called a stochastic process X_t. [18]

2.2 Functional Magnetic Resonance Imaging

There are 2 types of Magnetic Resonance Imaging MRI, Structural Magnetic Resonance Imaging (sMRI) is a technique that is used for medical purposes to display internal body organs, taking advantage of the hydrogen nuclei (protons) of different body tissues. When a large external magnetic field takes place, protons align in parallel with the field. Larmor frequency is the frequency that keeps protons rotating and is proportional to the magnetic field. In the presence of an excitation such as a radio-frequency pulse, the hydrogen nuclei absorb electromagnetic energy. The magnetic properties of nuclei of different tissues absorb different amounts of electromagnetic energy. Images can be built from those differences. In contrast, in Functional Magnetic Resonance Imaging fMRI the main issue investigated is not the difference between different tissues, but the difference between regions of the brain and the contrasts that occur in oxygen levels inside the blood. These alternations in blood oxygen levels reveal a higher or lower brain activity. When a region of the brain is activated, the metabolic needs for that region become higher. As a result, it is possible to research brain functionality and achieve brain mapping. One of the most important features of magnetic resonance is that it is not an invasive method.

2.2.1 fMRI Method

As previously stated, fMRI research focuses on examining the oxygenation and deoxygenation of blood in neuronal regions throughout the brain. When a region of the brain is activated, neurons in that region demand higher amounts of oxygen, which increases metabolism. Therefore, blood flow, which is rich in oxygen and glucose, is also increased. The oxygenated blood follows Haemodynamic Response Haemodynamic Response (HR) for that region. It is possible to measure the changes of oxygen in the blood as a result of magnetic properties, diamagnetic or paramagnetic, when the blood is oxygenated or deoxygenated, respectively. From the magnetic resonance Magnetic Resonance (MR) scanning, the BOLD signal can be obtained. After that, many signal and image processing techniques are used, as well as statistic techniques, to achieve the fMRI analysis. As a result, a brain map is made showing the brain functionality under some specific circumstances such as a stimulus. Generally, [1] fMRI is an important tool for mind scientists that help them understand the human mind. However, low spatial and temporal resolution may prove to be a serious obstacle to studying tiny timing differences between different processes, or small subsystems of the human brain.

2.2.2 MR Physics and BOLD Imaging

The MR scanner consists of superconducting electromagnets to produce a static, uniform magnetic field of high strength. In past decades, the most common magnetic field strength used in fMRI was around 1.5 Tesla Tesla (T) to 3.0 T. Today some research centers have scanners with remarkably stronger fields that reach 10 T. The procedure to produce and measure MR signal consists of 3 main steps.

The first step is to place the patient in a strong, static, uniform magnetic field, B_0. The orientation of nuclei of patients' atoms is at first randomly distributed. When the patient enters the strong, static, uniform magnetic field all the nuclei align along or opposite the magnetic field and spin. Nevertheless, their spin is out of phase and their rotation speed is proportional to the Larmor frequency, which depends on the magnetic field. The aligned with the field nuclei behave as a magnetic dipole. The average magnetic dipole moment density is termed a net magnetization vector.

The second step is to apply a second radiofrequency Radiofrequency (RF) magnetic field, B_1. This RF pulse is applied vertically to the first magnetic field, B_0. When the frequency of RF becomes equal to the Larmor frequency, resonance occurs. The existence of resonance causes the excitation of nuclei, such that all of them spin in phase and align their magnetization along with the transversal plane. The absorption of RF pulses' energy by nuclei causes a transition from higher to lower energy levels.

The third step is relaxation which is separated into two processes, T1 and T2 relaxation. After turning off the RF pulses, the nuclei tend to reach their equilibrium alignment state. The T1 relaxation takes place when the net magnetization returns to its initial maximum value, parallel to B_0, and is defined as the time needed to achieve the 63% of the maximum longitudinal magnetization. The T2 and T2* relaxation refers to the transversal plane. As a result of inhomogeneities of the remaining magnetic field (T2* decays) and spin-to-spin interactions (T2 decays), when the RF pulses are turned off, the nuclei are out-of-phase. Inhomogeneities such that, may be the result of intrinsic defects in the magnet itself or from susceptibility-induced field distortions produced by the tissue or other materials placed within the field. T2 relaxation refers to the time needed for the nuclei to dephase 37% of their original values. When the original equilibrium state of nuclei is reached and relaxation is completed, the transition from higher to lower energy levels causes an emission of a specific quantity of energy. That energy is detected by coils as a raw MR signal. The quantity of energy is proportional to the energy the nuclei absorbed in the second step. After detection, some procedures take place such as amplification, and then displayed as the Free Induction Decay Free Induction Decay (FID).

Furthermore, to create a three-dimensional display of the human brain it is necessary to use gradients, which can be computed thanks to three sets of gradient coils G_x, G_y, and G_z within the MR system. The disorders of static magnetic field can be varied as a matter of position. Therefore, it is possible to predict these disorders as a result of different positions in a three-dimensional plane and achieve the reconstruction of the brain map as an image. In addition, frequency measurements can reveal the different positions of MR signals. The software that controls all these magnetic fields is typically called 'the pulse sequence' and is responsible for spatial resolution thanks to three gradient coils G_x, G_y, and G_z. A main computer manages the functionality of the pulse sequence, but also of the scanner. In most fMRI experiments, the stimuli presented to the subject are generated by a second computer that also records the subjects' behavioral responses. Both computers have to be synchronized so that the onset of each stimulus presentation occurs at a precisely controlled moment during the image shown to the patient. As mentioned, in sMRI the measurement that matters is the difference between tissues of the body such as bone, gray matter, Cerebrospinal Fluid (CSF), and tumors. However, in the fMRI, in most cases, the measurements focus on the Blood Oxygen Level Dependent (BOLD) signal, which results from oxygen or its absence in hemoglobin through blood movement into vessels. That describes the two types of pulse sequences that are mostly used.

BOLD signal indicates whether hemoglobin in blood is oxygenated or de-oxygenated. Oxygenated hemoglobin (or oxyhemoglobin) which is a protein molecule found in red blood cells (erythrocytes) is created when blood passes through the lungs and binds oxygen molecules. Oxyhemoglobin has no unpaired electrons and is weakly diamagnetic. When oxygen is metabolized by the neurons during brain processes, hemoglobin is without the bound oxygen and becomes deoxygenated (deoxyhemoglobin). In this situation, it is strongly paramagnetic. The paramagnetic properties of deoxyhemoglobin lead to local magnetic field distortions around the activated regions of the brain such as neurons and blood vessels which feed them with oxygen and glucose. The T2 and T2* relaxation times are inversely proportional to the presence of deoxyhemoglobin. As the deoxygenated hemoglobin increases, the T2 and T2* times decrease which results in weaker signals in that regions. On the contrary, regions with a high presence of oxyhemoglobin give significantly stronger signals which are displayed by brighter shades in the final image.

To summarize, BOLD fMRI technique detects distortion in the homogeneity of the main magnetic field which comes from differences in oxygen levels in the blood. An idealized example for the process of BOLD response is shown in Fig. 2.1. When brain functionality increases in an area, metabolic demands rise and, as a result, the vascular system rushes oxyhemoglobin into the area. The rush of oxyhemoglobin into the area causes the ratio of oxygenated to deoxygenated hemoglobin (i.e., the BOLD signal) to rise quickly. Interestingly, the vascular system overcompensates, in the sense that the BOLD signal rises well above baseline to a peak at around 6 seconds (s) after the end of the neural activity that caused the above responses. Following this peak, the BOLD signal gradually decays back to baseline throughout

20–25 s.

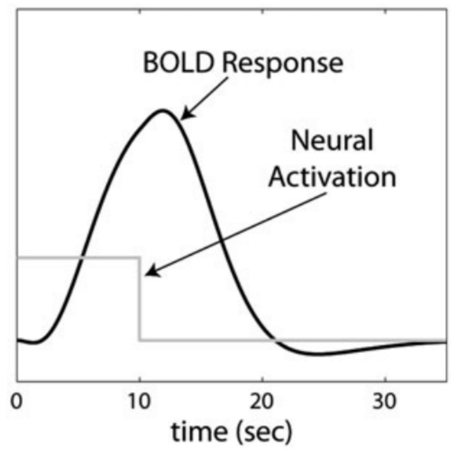


Figure 2.1: A hypothetical BOLD response (black curve) to a constant 10 s neural activation (gray curve) [1]

2.2.3 Scanning Session

Various scans are commonly included in an fMRI scanning session. In most cases [1], there are at least four types of scans acquired, which are described below.

The first, and the most critical scan in the process of an fMRI, is the localizer. This type of scan is a quick structural scan (1-2 minutes) of low spatial resolution which is used to locate the human brain in 3-dimensional space. This knowledge is needed to optimize the location of the slices that will be taken through the brain in the high-resolution structural and functional scans that follow.

The second scan, in most cases, is the high-resolution structural scan. Depending on two factors, the resolution of that scan and the exact nature of the pulse sequence that is used to control the scanner as we acquire the data, it possibly takes 8-10 minutes to complete. That second step plays a crucial role in the analysis of functional data. Speed is the higher priority during an fMRI scan, to have as higher temporal resolution as possible. However, high temporal resolution leads to low spatial resolution. To counterbalance the loss on spatial resolution, the structural scan contributes during the pre-processing steps, with the alignment of the structural with the functional data. In that way, we cover the bigger percentage of loss is made during the functional data collection.

The third step is often the collection of functional data. That step is the most time-consuming and can be made either on a long scan with duration of 20-30 minutes or can be broken down into 2-3 shorter runs. Between shorter runs, there are brief rests for the subjects. There are many parameter choices to make here, but two are especially important for the subsequent fMRI data analysis.

One choice is the repetition time Repetition Time (TR), which is the time between successive whole-brain scans. For whole brain scans, typical time for TR lasts 2-3 seconds, but in many cases and machines, especially if some brain parts are excluded from the scan, this time may even be 1 second.

A second important choice we have to make is the voxel size. Voxel size is an important factor that determines the spatial resolution of the functional data. When a subject enters the scanner, the brain occupies a certain volume. If we assign a coordinate system to the bore of the magnet, then we could identify any point in the subject's brain by a set of three coordinate values (x, y, z). By convention, the z direction runs down the length of the bore (from the feet to the head), and the x and y directions reference the plane that is created by taking a cut perpendicular to the z-axis. The brain, of course, is a continuous medium, in the sense that neurons exist at (almost) every set of coordinate values inside the brain. fMRI data, however, are discrete. The analog-to-digital conversion is performed by dividing the brain into a set of cubes (or more accurately, rectangular right prisms). These cubes are called voxels because they are three-dimensional analogs of pixels – that is, they could be considered as volume pixels.

A typical voxel size in functional imaging might be $3\ mm \times 3\ mm \times 3.5\ mm$. In this case, in a typical human brain, 33 separate slices might be acquired, each containing a 64×64 array of voxels for a total of 135,168 voxels in the brain. In each fMRI run, a BOLD response is recorded every TR seconds in each voxel. Thus, for example, in a 30-minute run with a TR of 2 s, 135,168 BOLD responses could be recorded 900 separate times (i.e., 30 times per minute $\times$ 30 minutes (min)), for a total of 121,651,200 BOLD values. This is an immense amount of data, and its sheer volume greatly contributes to the difficulties in data analysis.

Many studies stop when the functional data acquisition is complete, but some other types of scans are also common. The field map scan is a fourth type of scan that may take place during the session. We made the ideal assumption that the scanner has a completely uniform magnetic field across its entire bore. One assumption that can be easily refuted because when a human subject enters the bore, even if before has a uniform magnetic field, the field is now distorted to some extent. After the subject is inside the scanner, all inhomogeneities in the magnetic field are corrected via a process known as shimming. If shimming is successful, the magnetic field will be uniform at the start of scanning. Sometimes, however, especially in less reliable machines, distortions in the magnetic field will appear in the middle of the session. The field map, which takes only a minute or two to collect, measures the homogeneity of the magnetic field at the moment when the map is created. Thus, the field map can be used during later data analysis to correct for possible nonlinear distortions in the strength of the magnetic field that develop during the scanning session.

2.2.4 Data Acquisition

In this section, we present real-world task-related fMRI data. Specifically, we processed four datasets, recorded at the University General Hospital of Heraklion (UGHH), from a group of 25 healthy adults performing four visual tasks that were identical in all but one aspect (the precise kinematics of an observed person-directed action). First, we quote some information regarding the experiment design.

Experimental Design

The fMRI block design consists of four action observation conditions, each involving four "active" 35 s blocks alternating with four 35 s baseline blocks.

Indicative specifications are presented below. A video clip illustrating a two-movement action sequence was presented 6 times within each "active" block. The stimulus set-up was identical across blocks and conditions, consisting of a fixed red spot at the center of the display, presenting a female person sitting behind a table. A white tea cup was positioned on the table and a ceramic bowl 30 cm in diameter was located on a smaller table right next to the person's head. The data employed in the main analyses reported here were derived from four experimental conditions examining the effects of an action with the same goal but different kinematics:

- Fast to cup - Slow to person: It consists of a rapid grasping movement toward the teacup (time duration equal to 700 milliseconds (ms) and average velocity equal to 0.64 minutes (m)/seconds (sec)), followed by a much slower movement that brings the cup to the person's mouth (time duration equal to 3300 ms and average velocity equal to 0.14 m/sec).

- Slow to cup - Fast to person: It consists of a slow grasping movement toward the teacup (time duration equal to 3300 ms and average velocity equal to 0.14 m/sec), followed by a much faster movement that brings the cup to the person's mouth (time duration equal to 700 ms and average velocity equal to 0.64 m/sec). The stimulus layout is identical to the first.

- Fast to cup - Slow to bowl: It consists of a rapid grasping movement toward the teacup (time duration equal to 700 ms and average velocity equal to 0.64 m/sec), followed by a much slower movement that brings the cup over the edge of the bowl (time duration equal to 3300 ms and average velocity equal to 0.14 m/sec) which is located in the position of the person's head in first and second condition.

- Aimless action: The proximal and distal tables seen in the first three conditions constitute the visual background in a 5 sec video clip depicting an extended hand executing a two-step motion toward the center of the proximal table and then toward (but not reaching) the distal table). Peak motion velocity was the same for the first and second steps of the action, which was repeated 6 times within each “active” block.

2.2.5 Raw Data and Preprocessing

When an fMRI scan is running, we acquire a BOLD signal every TR seconds for each voxel. That signal is a value that indicates every single voxel’s oxygenation or deoxygenation for the region of the brain it belongs to. The data we acquire from that procedure is raw and not appropriate for display or analysis. For that reason, the pre-processing of data is necessary, to be suitable for display and analysis. There are some specific steps in the pre-processing stage that we follow without taking into account their experimental design. The preprocessing steps are modelled and finally solved [19], [20], [21], [22], [23], [24]. Pre-processing steps are implemented by SPM toolbox[1] and the steps are presented below. The header for each fMRI image contains an affine transformation matrix, which also includes sharing and scaling, except for translation and rotation. Scaling is used to alter the size of an image and shearing slants the shape of an image. After that, the estimation of the rigid body transformation matrix is necessary for re-slicing. The re-slicing method applies to all 2D images and then the images are mapped into the coordinate system of the reference image. Finally, B-splines interpolation takes place.

Realignment

The product of an fMRI running session is a sequence of N 3D images, where N indicates the number of time points in each voxel’s time series. Despite that, the main part of the analysis of the data is in slices corresponding to 2D

[1]The **SPM Toolbox** refers to a suite of tools and software developed for the analysis of brain imaging data, particularly functional and structural neuroimaging data. SPM stands for Statistical Parametric Mapping, and it is widely used in the field of neuroimaging for analyzing data from functional magnetic resonance imaging (fMRI), positron emission tomography (Positron Emission Tomography (PET)), electroencephalography (Electroencephalography (EEG)), and other brain imaging modalities.

images. Usually, on fMRI session the subject may move slightly their head. This motion may lead to a misregistration among the images, and finally drive us to false detection of activation for that voxel, or an entire region of the brain. Although that fact, there are techniques to eliminate the artifacts that come from motions, such as realignment. The task of realignment is to estimate the best rotation and translation parameters to properly register to a common coordinate system.

Slice-time Correction

During the analysis of fMRI data, we assume that the 3D image is measured at the same time for all points. In fact, the slices of the human brain at the time of scanning, are measured at slightly different times. The time, which is needed for a complete brain scanning, is TR. It is possible with the suitable time shifting between slices to eliminate that problem. For that purpose, we use a reference slice, which is usually the first or the middle slice. In addition, the data shifting requires further resampling (Interpolation) of the slice time courses.

Co-registration

Alignment and overlay of fMRI data of a single subject with its structural data, is termed as co-registration. The structural data can be acquired, either from an MRI session, from Positron Emission Tomography PET, or from Computed Tomography (CT). Due to their high resolution, structural data, after co-registration can be used to identify activated areas with higher accuracy. Co-registration attempts to find a rigid body transformation that maximizes a voxel matching criterion. Mutual information measure can be used as a voxel matching criterion to find the dependence of fMRI and structural data.

Segmentation

The purpose of segmentation is to divide the images given from the previous steps by their tissue class. The main tissue classes we may find on an anatomical image of the human brain are bone, grey matter, white matter, and Cerebrospinal Fluid CSF. For this purpose, a probabilistic model such as the Gaussian mixture model is necessary and can be used combined with smooth intensity variation and a tissue probability map. Tissue probability maps are constructed from a large number of human brains that are registered in a common space. MR images are usually corrupted by a smooth, spatially varying artifact that modifies the intensity of the image (bias). So, a parametric biased correction model is used. It is a fact that a voxel may contain information from different brain tissues.

Spatial Normalization

During the spatial normalization step, the images obtained from fMRI sessions from different individuals are registered into a common coordinate system defined from a reference or template image. An intensity-based method is used for that purpose. After that, a coordinate set (x, y, z) of the fMRIs of different subjects refers to the same voxel, although the fact that the brain shape of individuals may differ.

Spatial Smoothing

Spatial smoothing, which is the final step of pre-processing, averages each voxel with a weighted sum of its neighbours. The weights are defined by a Gaussian kernel and the width of that kernel is expressed by Full Width at Half Maximum Full Width at Half Maximum (FWHM). The goal of that final step is to improve the signal-to-noise ratio Signal-to-Noise Ratio (SNR) but also to deal with the differences between individuals that may occur. The improvement of SNR may help us detect activated areas more efficiently. As in any other case, so in this one, there are some trade-offs. The fact that spatial smoothing decreases spatial resolution and the size of the kernel window is bigger than it has to be, may lead to undetected small areas that are activated.

Chapter 3

The Expected fMRI Response

The expected response is the ideal theoretical response that is believed to be taken from an fMRI session if we completely remove the noise acquired during the fMRI session. The expected response is the result of convolution of the HRF with the stimulus function. Ideally, if there was not any kind of noise, the time series of each activated voxel would be identical to the expected response.

3.1 Haemodynamic Response Function

First of all, we must define the HRF which express the HR. Many studies [25], [26], [27] in fMRI data analysis have shown that the brain response to a stimulus is approximately equal to the convolution of the input paradigm with the HRF. The HRF is the theoretical (impulse) response of the BOLD signal to a very short unit intensity stimulus. To better understand the dynamics of the human brain, knowledge of HRF is one of the most significant factors. There are two main categories in which the HRF computation can be classified, parametric and non-parametric.

Parametric category methods presume that the HRF is a non-linear function of certain parameters, and these parameters are often given physiological meaning. A number of functions have been proposed to model the HRF: the Poisson function, the gamma-variate function, the Gaussian function, spine-like function, and a difference of two gamma-variate functions.

Non-parametric methods make no prior hypothesis about the shape of the response function. Methods include averaging over regions, selective averaging, smooth Finite Impulse Response (FIR) filters, de-convolution, and the Bayesian method.

Below we present basic type functions that can model the HRF and we make a discussion for each method and which of them we are most likely to be chosen.

3.1.1 Poisson Function

According to [27] the Poisson function for modeling the HRF is a questionable choice. This comes from the fact that variance and mean for the Poisson function is equal and defined by the same parameter λ. We must not forget that τ is a positive integer and represents the seconds.

$$h(\tau) = \frac{\lambda^{\tau} e^{-\lambda}}{\tau\,!} \tag{3.1}$$

If $\tau \in \mathbf{R}$ then Eq. 3.1 can be replaced by a Gamma-distribution with appropriate moments.

3.1.2 Gamma-variate Function

An experiment took place by [28], when they apply a visual flash stimuli of many durations. After that, they averaged the fMRI signal time course in visual cortex to repeated stimulation in order to assess the sensitivity of fMRI recordings to very short duration stimuli. They use flash stimuli of three durations, 34ms, 100ms and 1000ms. Then they conclude on the following equation for HRF.

$$h(\tau) = k\tau^{a} e^{-\frac{\tau}{b}} \tag{3.2}$$

where k is adjusted to give unit amplitude at equilibrium, $a = 8.60$, $b = 0.547$ as measured in [29] after fitting of the data from the averaged responses to 1-s stimuli to a three-parameter gamma variate function using the Levenberg-Marquardt algorithm. That experiment can be illustrated as shown in Fig. 3.1. From that figure, we can conclude that the three parameter Gamma-function is very closely related with the HRF than the Poisson function. Although that, it is not the perfect fit.

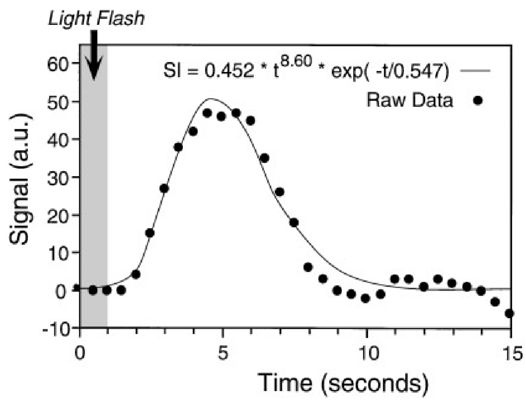


Figure 3.1: Illustration of HRF function

3.1.3 Gaussian Function

According to [30] the Gaussian function oriented HRF is given by

$$h(\tau) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\tau-\mu)^2}{2\sigma^2}}, \tag{3.3}$$

After that, we must show the necessity that the integral of HRF must be unity

$$\int_{-\infty}^{\infty} h(t)\mathrm{dt} = 1 \tag{3.4}$$

This condition is mandatory for the probability density functions and, in fact, all above parametric models, Poisson, Gamma, and Gaussian, are widely used density functions in the theory of probability.

Now we must introduce two parameters, lag and dispersion of HRF $h(t)$

$$\mathrm{lag} \triangleq \int_{-\infty}^{\infty} th(t)\mathrm{dt} \tag{3.5}$$

$$\mathrm{dispersion} \triangleq \int_{-\infty}^{\infty} (t - \mathrm{lag})^2 h(t)\mathrm{dt} \tag{3.6}$$

A very interesting fact is that the two parameters of mean and variance of the Gaussian function are equal to the lag and dispersion of the HRF. Ironically, the lag and dispersion values provided by Poisson and Gamma models are related to each other. As mentioned in [30], the Gaussian oriented HRF presents a reasonable and flexible model to represent the delay and temporal correlations seen in fMRI time-series, and seems valid because the hemodynamic parameters determined in that experiment are closer to those previously reported. Both Gamma and Poisson models cannot independently account for the lag and dispersion of fMRI time-series because these parameters are linearly related.

3.1.4 Difference of two Gamma-variate Functions

According to [31], a difference of two gamma functions to model the slight intensity drop after the response has fallen back to zero [32] and is believed to be as accurate as it can be in relation to actual HRF. An example is the HRF available in SPM'96,

$$h(\tau) = \left(\frac{t}{d_1}\right)^{a_1} e^{-\frac{t-d_1}{b_1}} - c\left(\frac{t}{d_2}\right)^{a_2} e^{-\frac{t-d_2}{b_2}}, \tag{3.7}$$

where t is time in seconds, $d_j = a_j b_j$ is the time to the peak, and $a_1 = 6$, $a_2 = 12$, $b_1, b_2 = 0.9$ seconds, and $c = 0.35$ [33].

3.2 Computation of Expected Response

It is common to assume a linear relationship between neuronal activity and expected response, where linearity implies that the magnitude and shape of the evoked HRF do not depend on any of the preceding stimuli. The ability to assume linearity is important, as it allows the relationship between stimuli and the BOLD response to be modeled using a linear time invariant system, where the stimulus acts as the input and the HRF acts as the impulse response function. As mentioned above, in a linear system framework the signal at time t, $x(t)$, is modeled as the convolution of a stimulus function $v(t)$ and the HRF h, that is [34],

$$E(t) = (v \circledast h)(t). \tag{3.8}$$

One of the most reliable methods to model HRF is by use the difference of two Gamma functions [33]. Varying stimulus patterns will give rise to responses with radically different features as in Fig. 3.2, where we see that phenomena [34].

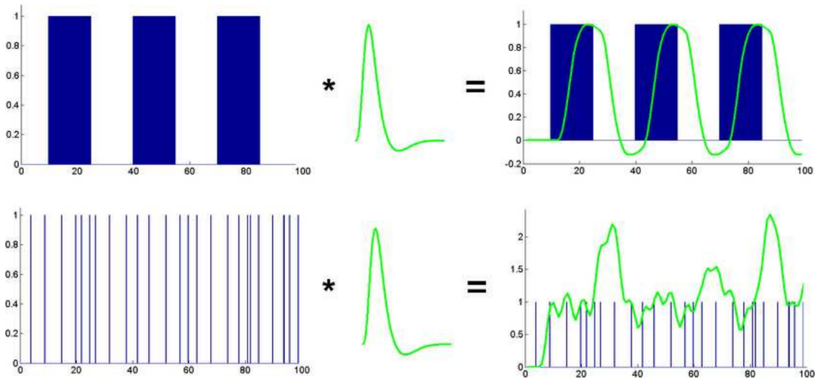


Figure 3.2: Expected response as convolution of HRF and stimuli

The form of expected response that is believed to be close to the clear signal (without noise) we gain after an fMRI session from the brain is shown in Fig. 3.3. In that figure, we see the convolution of HRF with a block shaped design of stimuli.

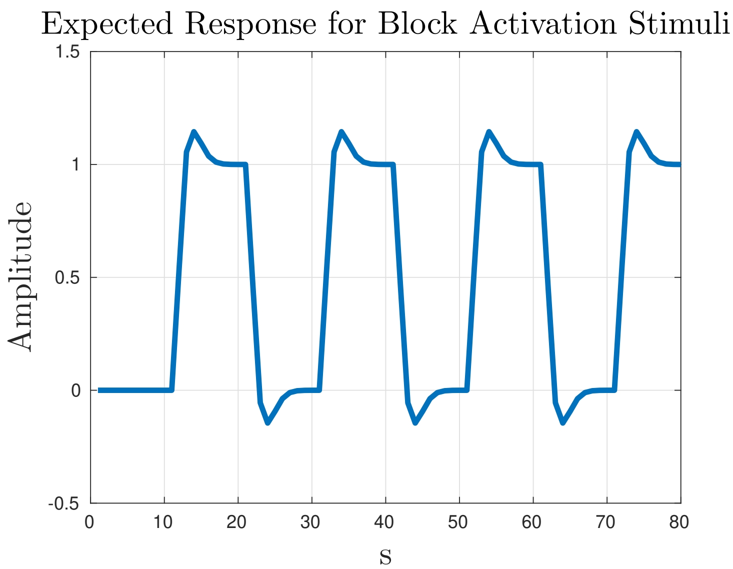


Figure 3.3: Expected response for blocked activation stimuli

Although we have a significant consideration about responses of the brain, we must take also into consideration that every signal we measure has a serious contribution of noise in it. Also we are not sure at all about the total amplitude of the expected response. When we have a high-amplitude noise, contaminated in a low-amplitude signal, every try for interpretation of data may lead to a failure. Also, many changes in the amplitude may take place depending on the region of the brain. So, in some regions we may have strong signal and on another we may take weak signal of brain activity.

3.3 Linear Model for Expected Response

After the computation of the clear signal of expected response, a question arises. Are we capable of modeling the actual fMRI data and in what way could that took place? As we see in the previous subsection, we want a scalar size in order to express the variation between the amplitudes of the signal in different brain regions. We also want additive noise because during the fMRI session a portion of the noise is taken close to the signal. Therefore we can suggest the above in order to express a synthetic data model

$$D(t) = \alpha \times E(t) + n(t) \tag{3.9}$$

In the eq. 4.1, $D(t)$ is the response BOLD response using the eq. 4.1, where each term has a specific purpose to help us interpret neural activity.

- **Observed Signal** $D(t)$: The signal $D(t)$ constitutes the BOLD response as captured by the fMRI scanner, which reflects changes in blood oxygen linked to neural activity. By analyzing y over time, we can understand how the brain responds to a stimulus.

- **Expected Response** $E(t)$: The expected response, $E(t)$, is often created by convolving the stimulus with the HRF. The HRF models how blood flow responds to neural activity, with a typical peak around 4-6 seconds after the stimulus starts, then slowly returning to baseline.

- **Scaling Parameter** a: The scalar a adjusts the amplitude of $E(t)$, capturing the strength of the BOLD response related to the stimulus.

Differences in between brain regions or experimental conditions can reflect differences in neural activity or blood flow response.

- **Error Term** n: The term n captures noise in the observed signal that the model does not explain. It includes things like physiological fluctuations, scanner noise, and any unrelated signals that add to the variability in $D(t)$. For purposes of this work, $n(t)$ can be one of the following distributions with adjusted parameters for each case in order to observe the FSIP behavior. These distributions are:
  - Gaussian Distribution
  - Student-t Distribution
  - Power Exponential Distribution
  - Gamma Distribution
  - Weibull Distribution
  - Log-Normal Distribution
  - Uniform Distribution

  More information about these distributions and their implementation with the ideal expected fMRI response will be given in the following chapters.

Chapter 4

Methods

4.1 Introduction

In this chapter, we describe the methodology that is appropriate to be applied, in order to find the activated regions of human brain. For the purposes of that analysis we use the already preprocessed fMRI data, which are acquired from the pre-processing steps. The main tool for that purpose is the Fisher-Shannon analysis, in which two significant quantities are calculated. These quantities is FIM and SEP. By using these quantities we can create the FSIP and study the behaviour of the fMRI data.

4.2 FIM & SEP Estimation

In this section, will be described the tools that are used for estimation of FIM and SEP. That estimation will lead to FSIP in order to continue with analysis. The first way is directed using formulas calculated by [35] and [36]. The other is by using a Kernel-based approach with KDE where particular formulas are used to compute the PDF of the series.

4.2.1 Theoretical Calculations

For the purposes of this work, we use six different probability distribution models. For the Gaussian, Weibull, Gamma and Log-normal distributions we use analytical formulas from [36]. For the Student-t and power exponential distributions, we use analytical formulas from [35]. For the computation of FIM and SEP we take into account the shape and / or scale parameters of the distributions. These formulas are presented in Sect. 5.

4.2.2 Kernel Density Estimation

Taking into account that computation of FIM and SEP requires to know the PDF of the series by definition, it is necessary to find a way to compute PDF with small computational complexity. In that need, KDE can gives us the solution. As it described in [37], for a given time series x_i of length L, we have

$$\hat{f}_{b,L} = \frac{1}{bL}\sum_{i=1}^{L} K\left(\frac{x - x_i}{b}\right), \tag{4.1}$$

where b is the bandwidth and $K(x)$ is the kernel function, which is assumed to be continuous, non-negative, symmetric around zero, and satisfying the following constraint

$$\int_{-\infty}^{+\infty} K(u)du = 1. \tag{4.2}$$

In our case, we use a gaussian kernel with zero mean and unit variance. In that case eq. 4.1 is transformed in

$$\hat{f}_{b,L} = \frac{1}{bL\sqrt{2\pi}}\sum_{i=1}^{L} e^{-\frac{1}{2}\left(\frac{x-x_i}{b}\right)^2}. \tag{4.3}$$

This method optimizes the bandwidth of the kernel density estimator to estimate the PDF $f(x)$. The optimal value of the bandwidth [38] is given by

$$b^* = n^{-\frac{1}{5}}\left[J\left(f\right)\right]^{-\frac{1}{5}}\left[M\left(K\right)\right]^{\frac{1}{5}} \tag{4.4}$$

where

$$J(f) = \int_{-\infty}^{+\infty}\left(f^{''}(x)\right)^2 dx \tag{4.5}$$

with $f^{''}(x)$ is the 2^{nd} derivative of f and

$$M(K) = \int_{-\infty}^{+\infty} K^2(u)du \tag{4.6}$$

The bandwidth b is approximated by means of an iterative approximation of $J(f)$. Thus, a sequence of positive numbers $b(k)$ is constructed through the iterations, where k indicates the number of iterations.

4.3 Fisher-Shannon Information Plane

The Fisher-Shannon method leads to FSIP and it is widely used to analyze many complex dynamical processes in section of geophysics. These processes are met in tsunamigenic or non-tsunamigenic earthquakes, and generally in seismology and volcanology. They are also met on analyzing environmental processes such as climatic regimes on rainfall time series, on hydrological regime discrimination, and study of sea surface temperature and wind speed data. Research on the Fisher-Shannon method is rather diffuse and comes from various fields, e.g., information theory, physics, dynamical systems, machine learning, and statistics.

In this section, we are going to give insights on FIM, SEP, Fisher-Shannon Complexity and the Gaussian Limit.

- Fisher Information Measure, a tool that quantifies the sensitivity of a probability distribution to changes in its underlying parameters. FIM serves as a gauge of local variability of a distribution, highlighting the precision with which certain parameters can be estimated.

- Shannon Entropy Power, a measure that reflects the "spread" or uncertainty of a distribution. SEP offers a way to interpret entropy in terms of the effective number of configurations, providing insight into the disorder or randomness inherent in the system.

- Fisher-Shannon Complexity, an innovative measure derived from combining FIM and SEP. This measure captures the trade-off between organization and randomness within a distribution, allowing us to analyze the complexity of a system in a manner that encompasses both local and global information.

- Gaussian Limit, a critical aspect in the analysis of information measures. We discuss how, in many contexts, probability distributions converge to a Gaussian form, representing a state of minimum Fisher-Shannon complexity. This limit offers a benchmark for understanding how distributions deviate from or approximate Gaussian behavior, illuminating patterns in both structured and random systems.

4.3.1 Shannon Entropy Power

Consider a univariate random variable X with its PDF $f(x)$, which is sufficiently regular. Its differential entropy is defined as

$$H_{\mathrm{X}} = \mathbb{E}\left[-\log f(X)\right] = -\int f(x)\log f(x)dx \tag{4.7}$$

However, a more convenient quantity to work with, is the above

$$N_{\mathrm{X}} = \frac{1}{2\pi e}e^{2H_{\mathrm{X}}} \tag{4.8}$$

which is termed as Shannon Entropy Power (SEP) and is a strictly monotonically transformation of H_X. In the most cases, entropies H_X and N_X are interpreted as global measures of disorder/uncertainty/spread of $f(x)$. That means, the higher the entropy, the higher the disorder.

4.3.2 Fisher Information Measure

The FIM [35], which is also termed as the Fisher information of X with respect to a scalar translation parameter [39] is defined as

$$I_{\mathrm{X}} = \mathbb{E}\left[\left(\frac{\partial}{\partial x}\log f(X)\right)^2\right] = \int \frac{\left[\frac{\partial}{\partial x}f(x)\right]^2}{f(x)}dx. \tag{4.9}$$

The FIM is equivalent to the Fisher information of a location parameter of a parametric distribution. [40] The quantity of eq. 4.9 should not be confused with the Fisher information of a distribution parameter. As described in [36], the derivative of the log density is relative to x and not to some parameter. The form of the above equation , under mild regularity conditions is the following

$$I_{\mathrm{X}} = \mathbb{E}\left[-\frac{\partial^2}{\partial x^2}\log f(X)\right] \tag{4.10}$$

The quantity I_{X} is sometimes interpreted as a measure of order/organization/narrowness of X.

4.3.3 Fisher Shannon Complexity

The Fisher Shannon Complexity (FSC) is a statistical complexity measure and is defined as $C_X = I_X N_X$. FSC is constant if we apply scalar multiplication or addition, and standardization or normalization of X has no effect on FSC. Also, the isoperimetric inequality for entropies states that $C_X \geq 1$, with equality if and only if X is Gaussian. The equality for the FSC can be observed in FSIP and is the well-known Gaussian limit.

The FSIP is shown as an example in Fig. 4.1. It is often used the standard linear scale on analysis with FSIP. However, many times a log-log scale in plot is more sufficient in practice. In the FSIP, the only reachable values are in the set $\mathcal{D} = \{(N_X, I_X) \in \mathbb{R}^2 |\ N_X > 0, I_X > 0\ ,\ N_X I_X \geq 1\}$ * In the Gaussian case, the standard deviation σ (which plays the role of the scaling parameter) is equivalent to the multiplicative factor α. Therefore, while a point in the FSIP is described by (N_X, I_X), one can also describe it by (α, C_X). In the light of this, one can also think of FSC as a scale-independent measure of non-Gaussianity of X. The line $I_X N_X = 1$ separates the FSIP into two parts: one allowed ($I_X N_X > 1$) and one not allowed ($I_X N_X < 1$), and the distance of a signal point from the 'isocomplexity line' $I_X N_X = 1$ can measure the degree of complexity of the signal.

FIM, SEP and their differential entropy, closely interact. Let Z be a random variable independent of X with finite variance σ_Z^2. According to [40], using the *de Bruijn* identity leads to

$$\frac{d}{dy} H_X + \sqrt{t} Z \bigg|_{t=0} = \frac{1}{2} \sigma_Z^2 I_X, \tag{4.11}$$

which means that the variation of the differential entropy of a perturbed X

*$\mathbb{R}^2$ (read as 'R two' or 'R squared'), it means that each element in the set is a pair of real numbers. The notation represents the Cartesian plane, which is the two-dimensional Euclidean space.

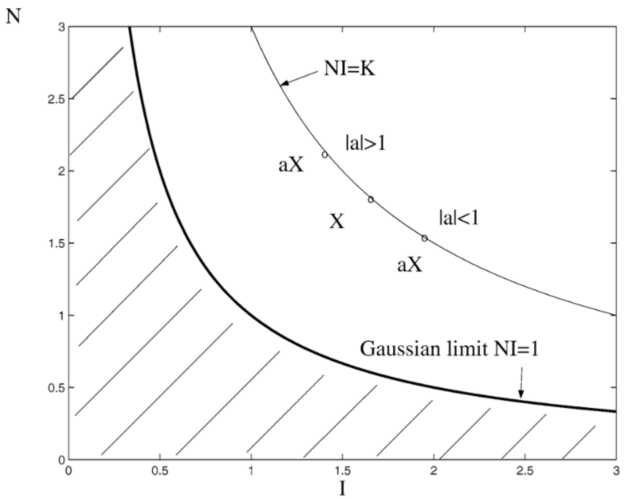


Figure 4.1: FSIP relation to FSC

is proportional to I_x. Also it easily be shown that we obtain

$$\frac{d}{dy}N_X + \sqrt{t}Z\bigg|_{t=0} = 2N_X\frac{d}{dy}H_X + \sqrt{t}Z\bigg|_{t=0} = \\ \sigma_Z^2 N_X I_X = \sigma_Z^2 C_X. \tag{4.12}$$

Hence, the FSC can be interpreted as a sensitivity measure of N_X to a small independent additive perturbation.

4.3.4 Gaussian Limit

The uncertainty property denotes the following.

$$I_\chi N_\chi \geq 1 \tag{4.13}$$

where I_χ is the FIM and N_χ is the SEP. The equality exists if and only if the random variable is used, follows Gaussian distribution. For every distribution other than Gaussian, the multiplication between I_χ and N_χ gives a number greater than 1.

However, there are cases that the calculation of FSIP gives values below 1. When a signal is not close to any standard distribution and the product of FIM and SEP is below 1, we are observing a unique situation that suggests a signal with very low structural information and/or low entropy. This can indicate several potential characteristics:

- High Randomness with Low Information Density: The low FIM implies that the signal lacks sharp transitions or distinct structural features, often seen in highly irregular or noisy signals. If SEP is also low, the signal's entropy (or randomness level) does not compensate with enough complexity to elevate its classification above the Gaussian limit.

- Sparse or Anomalous Data: Signals with sparse or minimal information content can fall below the Gaussian limit because they lack the statistical richness or well-defined characteristics of typical distributions. These signals might exhibit randomness without structure, such as sparse sensor data with noise interference, low-signal-to-noise ratio measurements, or degraded signals.

- Uninformative or Noise-like Signals: When FIM and SEP both contribute minimally, the signal might be largely uninformative or represent background noise with no dominant patterns or characteristics.

- Possible Measurement Limitations: In practical applications, low FIM and SEP values might indicate limits in measurement quality or signal resolution, where the data lacks sufficient granularity or is heavily degraded.

Chapter 5

Probability Functions and FSIP for Different Distribution Models

5.1 Normal Distribution

The general formula for the PDF of the normal distribution is

$$f(x) = \frac{e^{-(x-\mu)^2/(2\sigma^2)}}{\sigma\sqrt{2\pi}}, \tag{5.1}$$

where μ is the location parameter and σ is the scale parameter. The location and scale parameters of the normal distribution can be estimated with the sample mean and sample standard deviation, respectively. In Fig. 5.1a is shown the plot of the PDF for four values of σ.

The case where $\mu = 0$ and $\sigma = 1$ is called the standard normal distribution. The equation for the standard normal distribution is

$$f(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}. \tag{5.2}$$

Since the general form of probability functions can be expressed in terms of the standard distribution, all subsequent formulas in this section are given for the standard form of the function.

The formula for the Cumulative Distribution Function (CDF) of the standard normal distribution is

$$F(x) = \int_{-\infty}^{x} \frac{e^{-x^2/2}}{\sqrt{2\pi}} \tag{5.3}$$

and this integral does not exist in a simple closed formula, but it is computed numerically. In Fig. 5.1b is shown the plot of the CDF for four values of σ.
The formula for the Percent Point Function (PPF) of the normal distribution does not exist in a simple closed formula. It is computed numerically. In Fig. 5.1c is shown the plot of the PPF for four values of σ.
The formula for the Hazard Function (HF) of the normal distribution is

$$h(x) = \frac{\phi(x)}{\Phi(x)} \tag{5.4}$$

where Φ is the CDF of the standard normal distribution and ϕ is the PDF of the standard normal distribution. In Fig. 5.1d is shown the plot of the Hazard Function for four values of σ.
The Cumulative Hazard Function (CHF) can be computed from the CDF of normal distribution. In Fig. 5.2a is shown the plot of the CHF for four values of σ.
The normal Survival Function (SF) can be computed from the normal CDF. In Fig. 5.2b is shown the plot of the SF for four values of σ.
The normal Inverse Survival Function (ISF) can be computed from the normal percent point function. In Fig. 5.2c is shown the plot of the ISF for four values of σ.
Some of the most significant measures that are commonly used in statistics are presented on Table 5.1.

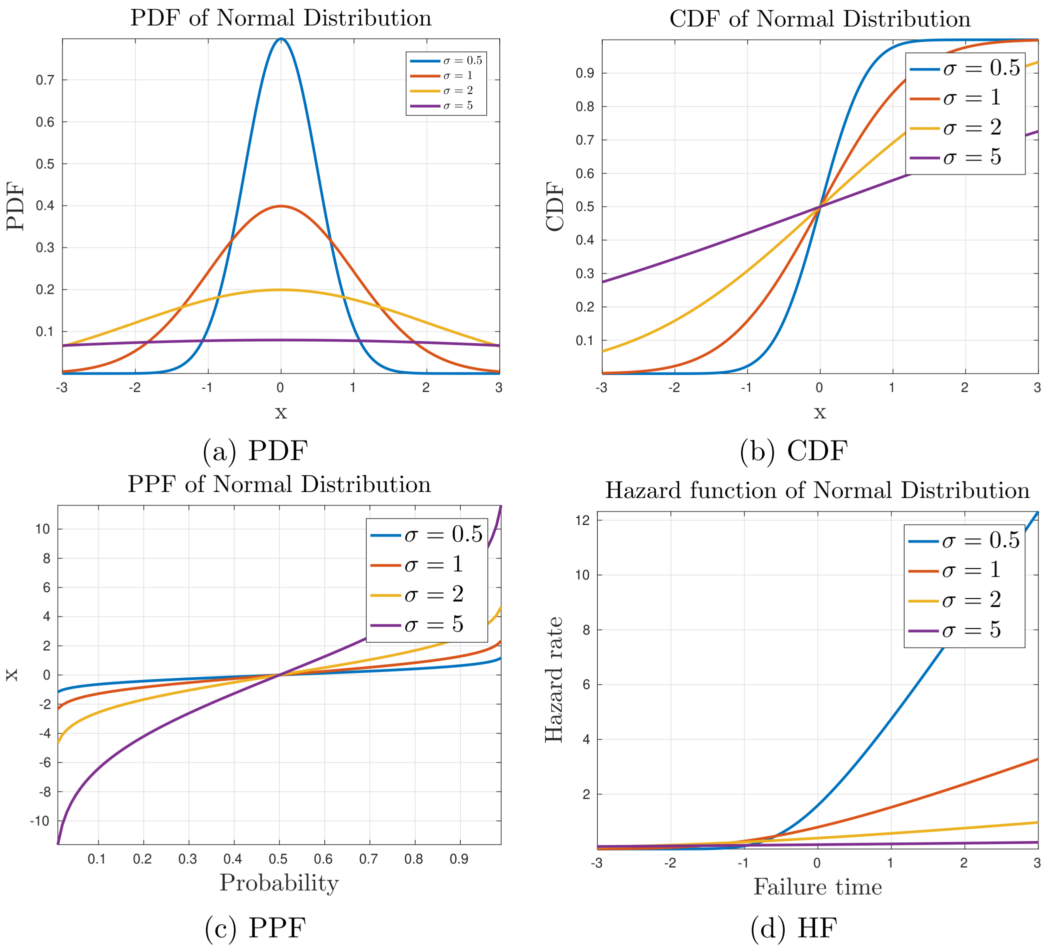
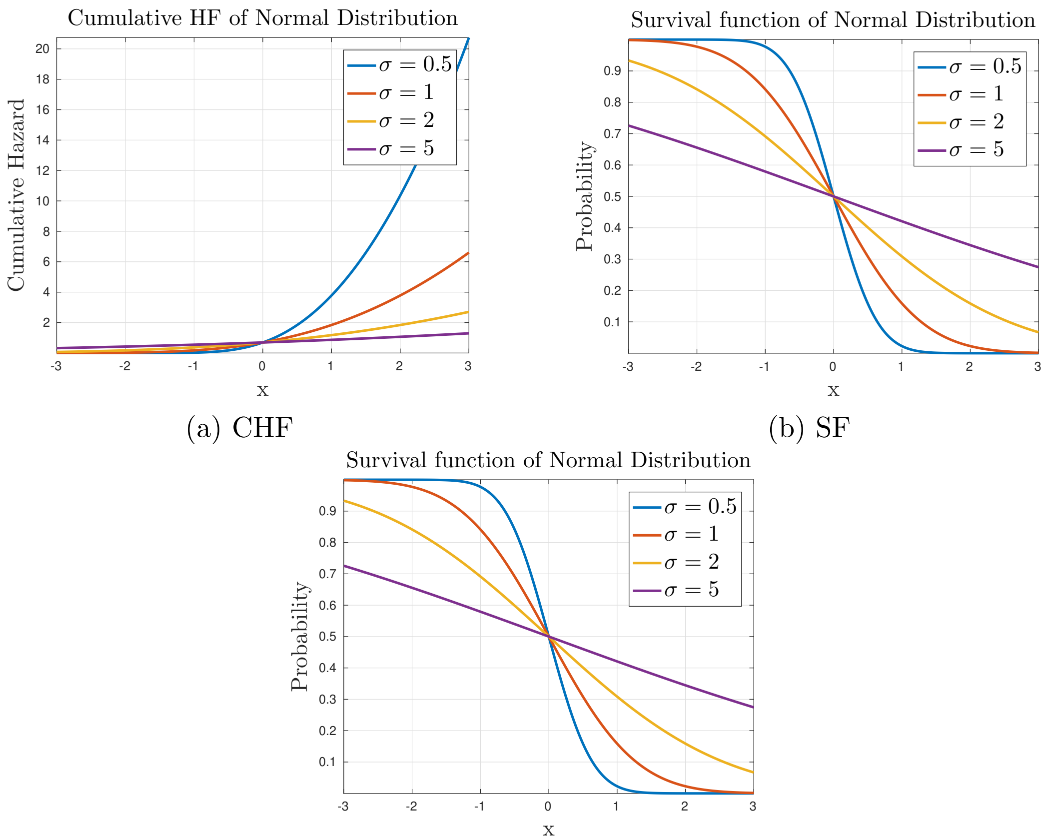


Figure 5.1: Probability functions of Gaussian Distribution (a)



(a) CHF

(b) SF

(c) ISF

Figure 5.2: Probability functions of Gaussian Distribution (b)

Table 5.1: Common Statistics for Normal Distribution

Common Statistics	
Statistic	**Value**
Mean	The location parameter μ
Median	The location parameter μ
Mode	The location parameter μ
Range	$(-\infty, +\infty)$
Standard Deviation	The scale parameter σ
Coefficient of Variation	$\frac{\sigma}{\mu}$
Skewness	0
Kurtosis coefficient	3

The SEP and the FIM of the Normal distribution with parameters μ and σ are

$$N_S(\mu, \sigma) = \sigma^2, \tag{5.5}$$

$$I_S(\mu, \sigma) = \frac{1}{\sigma^2}, \tag{5.6}$$

and the Fisher Shannon Information Plane is shown in Fig 5.3. In that figure we use a $\sigma \in [0.41, 2.45]$. For values of σ less than 0.41 we have a maximization on y-axis (maximization on FIM). For values greater than 2.45 we have a maximization on x-axis (maximization on SEP).

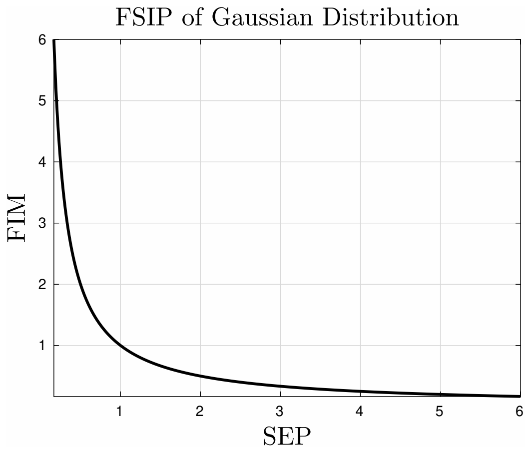


Figure 5.3: Normal Distribution. FSIP for different σ's.

5.2 Student-t Distribution

The formula for the PDF of the Student-t distribution is

$$f(x) = \frac{\left(1+\frac{x^2}{m}\right)^{\frac{-(m+1)}{2}}}{B\left(\frac{1}{2},\frac{1}{2}m\right)\sqrt{m}}, \tag{5.7}$$

where B is the beta function and m is a positive integer shape parameter. The formula for the beta function is

$$B(\alpha,\beta) = \int_0^1 t^{\alpha-1}\,(1-t)^{\beta-1}\,dt. \tag{5.8}$$

Shape parameter $m > 0$ also termed as Degrees of Freedom (DF) for that distribution. In Fig. 5.4a is shown the plot of the PDF for four values of DF. The formula for the CDF of the t distribution is

$$F_m(x) = \frac{1}{2} + \frac{x\Gamma\left(\frac{1}{2}(m+1)\right)\ {}_2F_1\left(\frac{1}{2},\frac{1}{2}(m+1);\frac{3}{2};-\frac{x^2}{m}\right)}{\sqrt{\pi m}\,\Gamma\left(\frac{1}{2}m\right)} \tag{5.9}$$

where m is the number of degrees of freedom, $x \in [-\infty,+\infty]$, ${}_2F_1(a,b;c;z)$ is a hypergeometric function[1], and Γ is the gamma function which has the formula

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}dt, \tag{5.10}$$

and in Fig. 5.4b is shown the plot of the CDF for four values of DF.
The formula for the PPF of the Student-t distribution does not exist in a simple closed form. It is computed numerically and in Fig. 5.4c is shown the plot of the PPF for four values of DF.
The HF of the Student-t distribution can be computed from PDF an CDF of that distribution and follows to Fig. 5.4d.
The CHF can be computed from the CDF of Student-t distribution. In Fig. 5.5a is shown the plot of the CHF for four values of DF.
The Student-t SF can be computed from the Student-t cumulative distribu-

[1] *Hypergeometric function* ${}_2F_1$ *is:* ${}_2F_1\,(a,b;c;z) = \Sigma_{n\geq 0}\frac{(a)_n(b)_n}{(c)_n}\frac{z^n}{n!}$ with $(a)_n := a(a+1)\cdots(a+n-1)$.

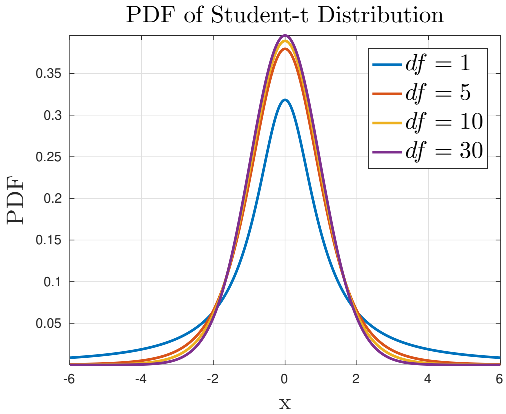
Table 5.2: Common Statistics for Student-t Distribution

Common Statistics	
Statistic	**Value**
Mean	0 (Undefined for $m = 1$)
Median	0
Mode	0
Range	$(-\infty, +\infty)$
Standard Deviation	$\sqrt{\frac{m}{(m-2)}}$, $m \geq 3$
Coefficient of Variation	Undefined
Skewness	0. Undefined for $m \leq 3$.
Kurtosis coefficient	$\frac{3(m-2)}{(m-4)}$.Undefined for $m \leq 4$

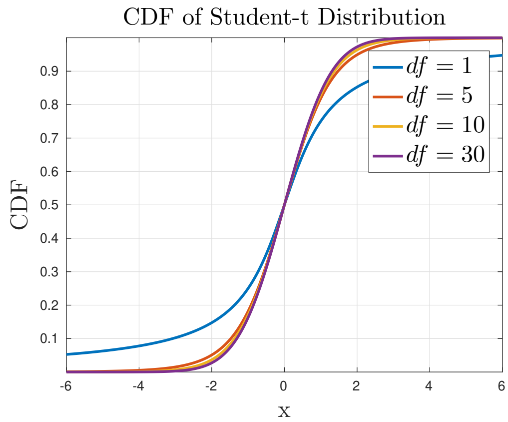
tion function. In Fig. 5.5b is shown the plot of the SF for four values of DF.

The normal ISF can be computed from the Student-t PPF. In Fig. 5.5c is shown the plot of the ISF for four values of DF.

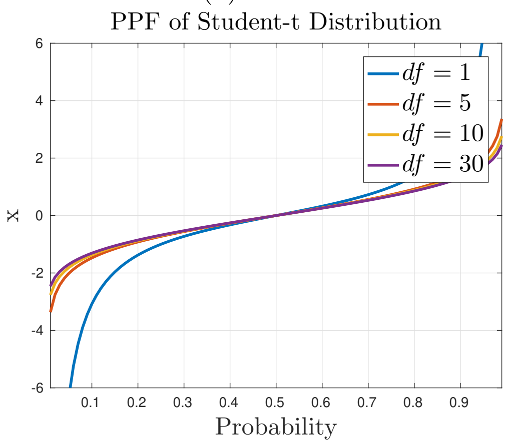
Some of the most significant measures that are commonly used in statistics are presented on Table 5.2.



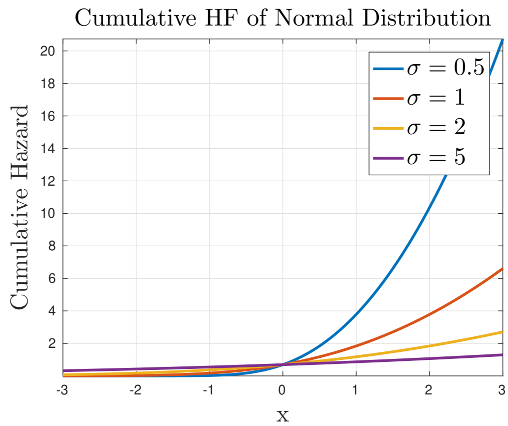
(a) PDF



(b) CDF

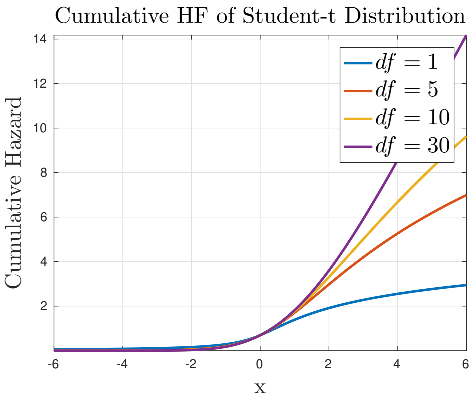


(c) PPF

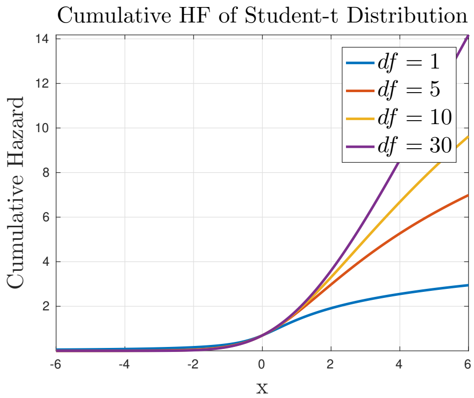


(d) HF

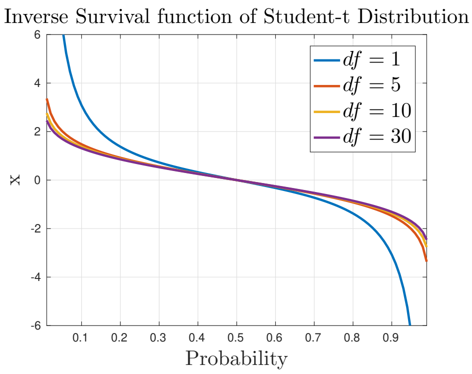
Figure 5.4: Probability functions of Student-t Distribution (a)



(a) CHF



(b) SF



(c) ISF

Figure 5.5: Probability functions of Student-t Distribution (b)

The SEP and the FIM of the Student-t distribution with parameters m and σ are

$$N_S(m,\sigma) = \frac{1}{2\pi e}\left(\frac{\sigma\sqrt{m-2}\Gamma(\frac{m}{2})\Gamma(\frac{1}{2})}{\Gamma(\frac{m+1}{2})}\right)^2 e^{(m+1)(\psi(\frac{m+1}{2})-\psi(\frac{m}{2}))} \tag{5.11}$$

$$I_S(m,\sigma) = \frac{1}{\sigma^2}\frac{m(m+1)}{(m-2)(m+3)}\,. \tag{5.12}$$

where $m = \frac{1+q}{1-q}$ and q is the extensivity parameter, with $m > 2$ and $1 > q > 0$, σ is a scale parameter and ψ is the digamma function. The digamma function is

$$\psi(x) = \frac{d}{dx}\ln\Gamma(x) \tag{5.13}$$

where Γ is the Gamma function, or

$$\psi(x) = -\gamma + \sum_{k=0}^{\infty}\left(\frac{1}{k+1} - \frac{1}{x+k}\right). \tag{5.14}$$

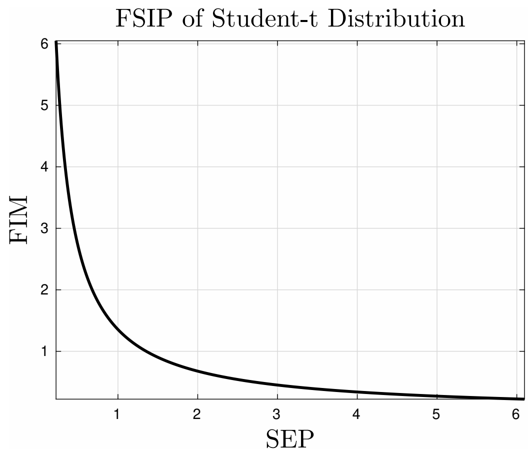


Figure 5.6: Student-t Distribution. FSIP for different σ's.

The FSIP is shown in Fig 5.6. In that figure we use scale parameter $\sigma \in [0.575, 3]$. For values of σ less than 0.575 we have a maximization on y-axis (maximization on FIM). For values greater than 3 we have a maximization on x-axis (maximization on SEP). The degrees of freedom is $m = 3$ in Fig 5.6.

5.3 Normal & Student-t Distribution Comparison

As we saw from the above PDFs of Normal and Student-t distributions, they are both stationary, bell-shaped, symmetrical distributions about the mean, the median and the mode. The most significant difference between those two distributions is that in the Student-t case PDF has heavier tails than the normal distribution. That means that in tail ends there are more values of the distribution than in Normal case. So, tails decline more slowly. At this point, it should be mentioned that as the number of degrees of freedom

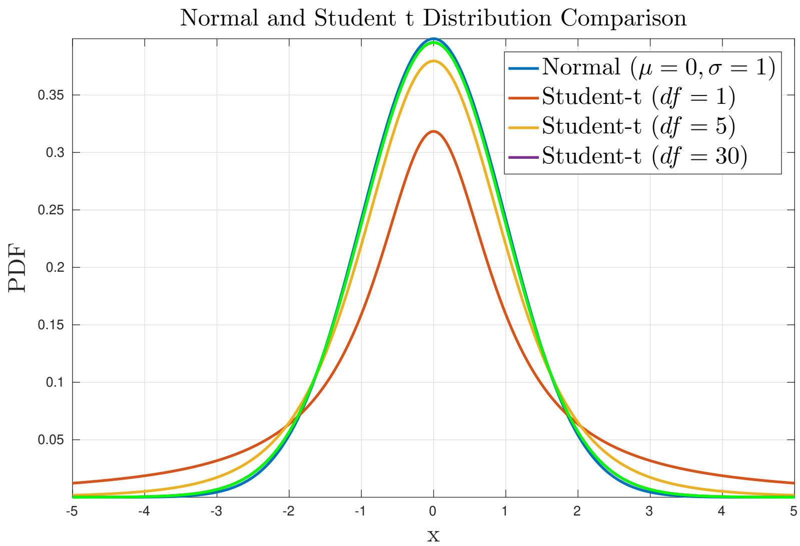


Figure 5.7: Normal & Student-t Distribution PDF

increases, in such a way the Student-t distribution approaches the normal. In order to illustrate this, we must consider the following Fig. 5.7 that shows the shape of the Student-t distribution with varied degrees of freedom. The shape of the t distribution depends on the degrees of freedom. The curves with more degrees of freedom are taller and have thinner tails. All Student-t distributions have “heavier tails” than the Standard Normal distribution. For values of degrees of freedom higher than 30 we can assume that the Student-t distribution is almost identical to the Normal one.

In statistical jargon, the metric called kurtosis coefficient is used to define how heavy-tailed a distribution is. Therefore, kurtosis coefficient is lower on the normal case than the Student-t one. In practice, we usually use the Student-t distribution to perform various hypothesis tests or to construct confidence intervals.

5.4 Power Exponential Distribution

The general formula for the probability density function (PDF) of the Power Exponential Distribution (PED) is

$$f(x) = \frac{\gamma\lambda^{\frac{1}{\gamma}}}{2\Gamma\left(\frac{1}{\gamma}\right)} e^{-\lambda|x-\mu|^{\gamma}}, \text{ with } x, \mu \in \mathbb{R}; \lambda > 0, \gamma > 1, \tag{5.15}$$

where μ is the location parameter, $\lambda > 0$ is a scale parameter and $\gamma > 1$ is a shape parameter.
PED also termed as Generalized Gaussian Distribution (GGD). The case where $\mu = 0$ and $\lambda = 1$ is called the standard GGD. The equation for the standard GGD reduces to

$$f(x) = \frac{\gamma}{2\Gamma\left(\frac{1}{\gamma}\right)} e^{-|x|^{\gamma}}, \text{ with } x \in \mathbb{R}; \gamma > 1, \tag{5.16}$$

Since the general form of probability functions can be expressed in terms of the standard distribution, all subsequent formulas in this section are given for the standard form of the function.
The formula for the cumulative distribution function CDF of the GGD is

$$F(x) = 1 - \frac{\Gamma\left(\frac{1}{\gamma}, x^{\gamma}\right)}{2\Gamma\left(\frac{1}{\gamma}\right)} \tag{5.17}$$

The formula for the percent point function of the gamma distribution does not exist in a simple closed form. It is computed numerically.
The formula for the hazard function of the GGD is

$$h(x) = \begin{cases} \frac{\gamma e^{-|x|^{\gamma}}}{2\Gamma\left(\frac{1}{\gamma}\right) - \Gamma\left(\frac{1}{\gamma}, |x|^{\gamma}\right)} & x \leq 0 \\ \frac{\gamma e^{-x^{\gamma}}}{\Gamma\left(\frac{1}{\gamma}, x^{\gamma}\right)} & x > 0 \end{cases} \tag{5.18}$$

The formula for the cumulative hazard function of the gamma distribution is

Table 5.3: Common Statistics for GGD

Common Statistics	
Statistic	**Value**
Mean	0 (else μ)
Median	0 (else μ)
Mode	0 (else μ)
Range	$(-\infty, +\infty)$
Standard Deviation	$\frac{\lambda^2\Gamma\left(\frac{3}{\gamma}\right)}{\Gamma\left(\frac{1}{\gamma}\right)}$
Coefficient of Variation	Undefined
Skewness	0
Kurtosis coefficient	$\frac{\Gamma\left(\frac{5}{\gamma}\right)\Gamma\left(\frac{1}{\gamma}\right)}{\Gamma\left(\frac{3}{\gamma}\right)^2} - 3$

$$H(x) = -\log\left\{1 - \frac{\Gamma\left(\gamma, \gamma|x|^{\frac{1}{\gamma}}\right)}{2\Gamma(\gamma)}\right\} \tag{5.19}$$

The formula for the survival function of the GGD distribution is

$$S(x) = \frac{\Gamma\left(\frac{1}{\gamma}, x^{\gamma}\right)}{2\Gamma\left(\frac{1}{\gamma}\right)} \tag{5.20}$$

The PED inverse survival function does not exist in simple closed form. It is computed numerically.

$$Z(\alpha) = G(1-\alpha) \tag{5.21}$$

Some of the most significant measures that are commonly used in statistics ($\mu = 0$, $\lambda = 1$ considered) are presented on Table 5.3.

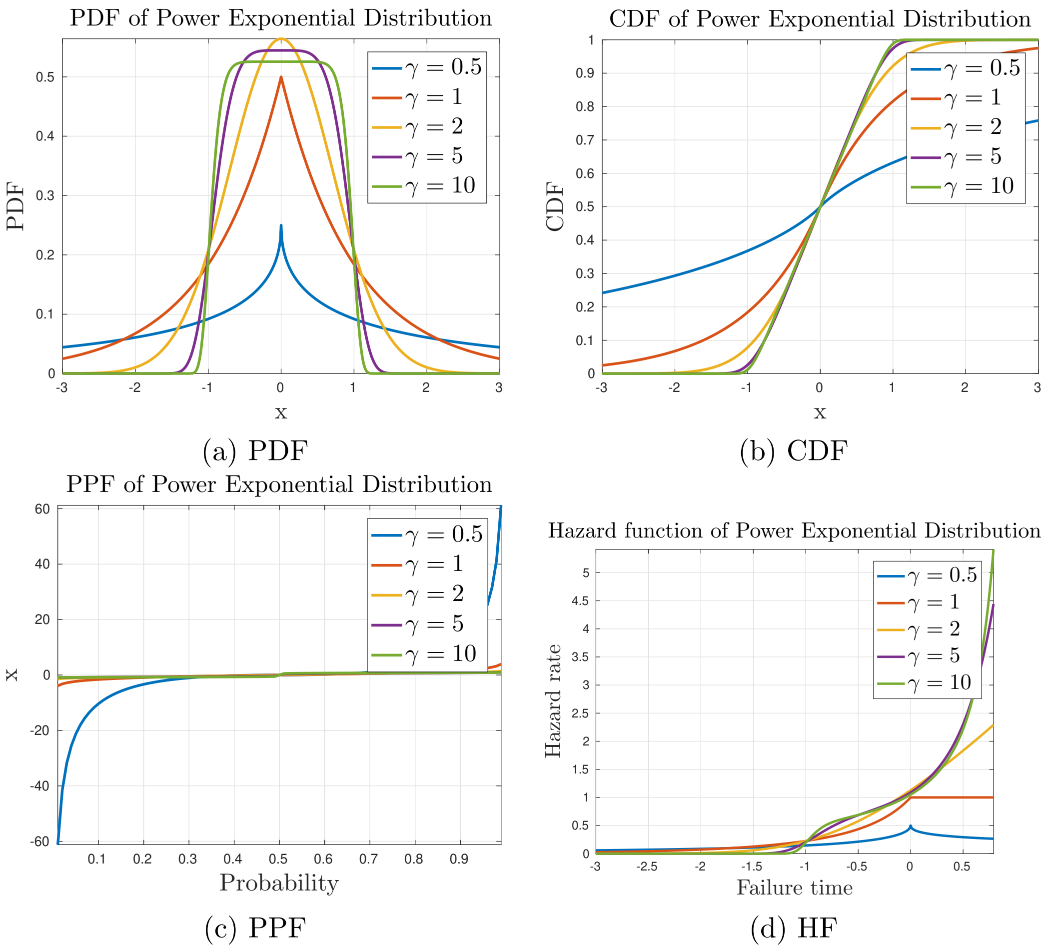
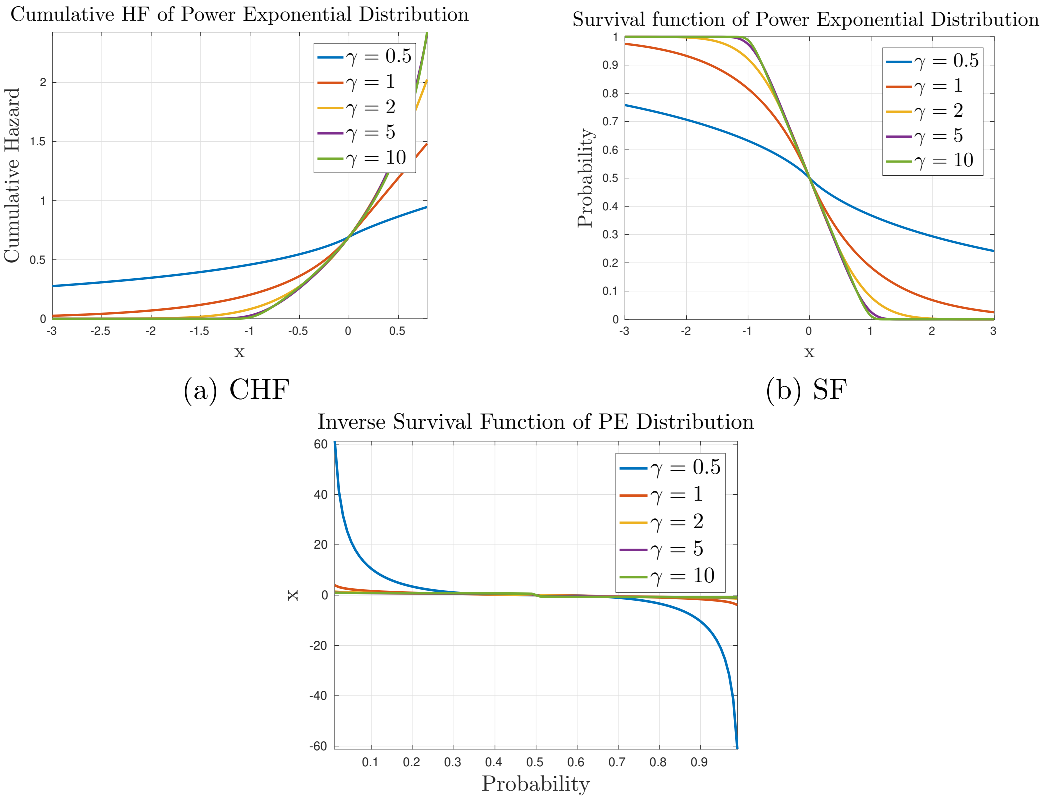


Figure 5.8: Probability functions of Power Exponential Distribution (a)



(a) CHF

(b) SF

(c) ISF

Figure 5.9: Probability functions of Power Exponential Distribution (b)

The SEP and the FIM of the power exponential law with parameters λ and γ are

$$N_{PE}(\lambda,\gamma) = \frac{2}{2\pi e}\left(\frac{1}{\gamma}\Gamma\left(\frac{1}{\gamma}\right)\left(\frac{e}{\lambda}\right)^{\frac{1}{\gamma}}\right)^2 , \tag{5.22}$$

$$I_{PE}(\lambda,\gamma) = \frac{\Gamma\left(1-\frac{1}{\gamma}\right)}{\Gamma\left(\frac{1}{\gamma}\right)}\gamma\left(\gamma-1\right)\lambda^{\frac{2}{\gamma}} . \tag{5.23}$$

and the FSIP is shown in Fig 5.10. In that figure we use a $\lambda \in [0.005, 2.8]$. For values of λ less than 0.005 we have a maximization on y-axis (maximization on FIM). For values greater than 2.8 we have a maximization on x-axis (maximization on SEP).

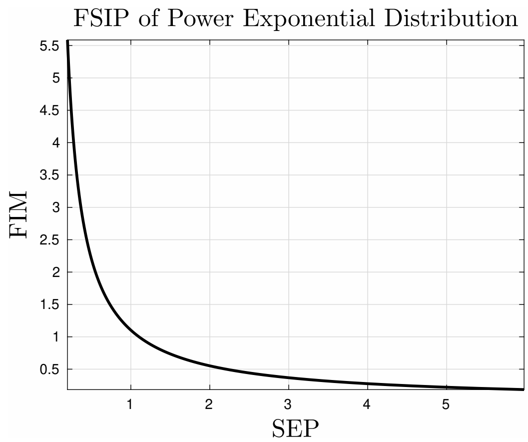


Figure 5.10: Power Exponential Distribution. FSIP for different λ's.

The FSIP calculated for different scale parameter λ's and shape parameter $\gamma = 3$.

5.5 Gamma Distribution

The general formula for the probability density function PDF of the gamma distribution is

$$f(x) = \frac{\left(\frac{x-\mu}{\theta}\right)^{\kappa-1} e^{-\frac{x-\mu}{\theta}}}{\theta\Gamma(\kappa)}, \text{ with } x \geq \mu; \kappa, \theta > 0 \tag{5.24}$$

where κ is the shape parameter, μ is the location parameter, θ is the scale parameter and Γ is the Gamma function.
The case where $\mu = 0$ and $\theta = 1$ is called the standard gamma distribution. The equation for the standard gamma distribution reduces to

$$f(x) = \frac{x^{\kappa-1} e^{-x}}{\Gamma(\kappa)}, \text{ with } x \geq 0; \kappa > 0 \tag{5.25}$$

Since the general form of probability functions can be expressed in terms of the standard distribution, all subsequent formulas in this section are given for the standard form of the function.
The formula for the CDF of the gamma distribution is

$$F(x) = \frac{\Gamma_x(\kappa)}{\Gamma(\kappa)}, \text{ with } x \geq 0; \kappa > 0 \tag{5.26}$$

where Γ is the gamma function defined above and $\Gamma_x(\alpha)$ is the incomplete gamma function. The incomplete gamma function has the formula

$$\Gamma_x(\alpha) = \int_0^x t^{\alpha-1} e^{-t} dt \tag{5.27}$$

The formula for the percent point function of the gamma distribution does not exist in a simple closed form. It is computed numerically.
The formula for the hazard function of the gamma distribution is

$$h(x) = \frac{x^{\kappa-1} e^{-x}}{\Gamma(\kappa) - \Gamma_x(\kappa)}, \text{ with } x \geq 0; \kappa > 0 \tag{5.28}$$

The formula for the cumulative hazard function of the gamma distribution is

Table 5.4: Common Statistics for Gamma Distribution

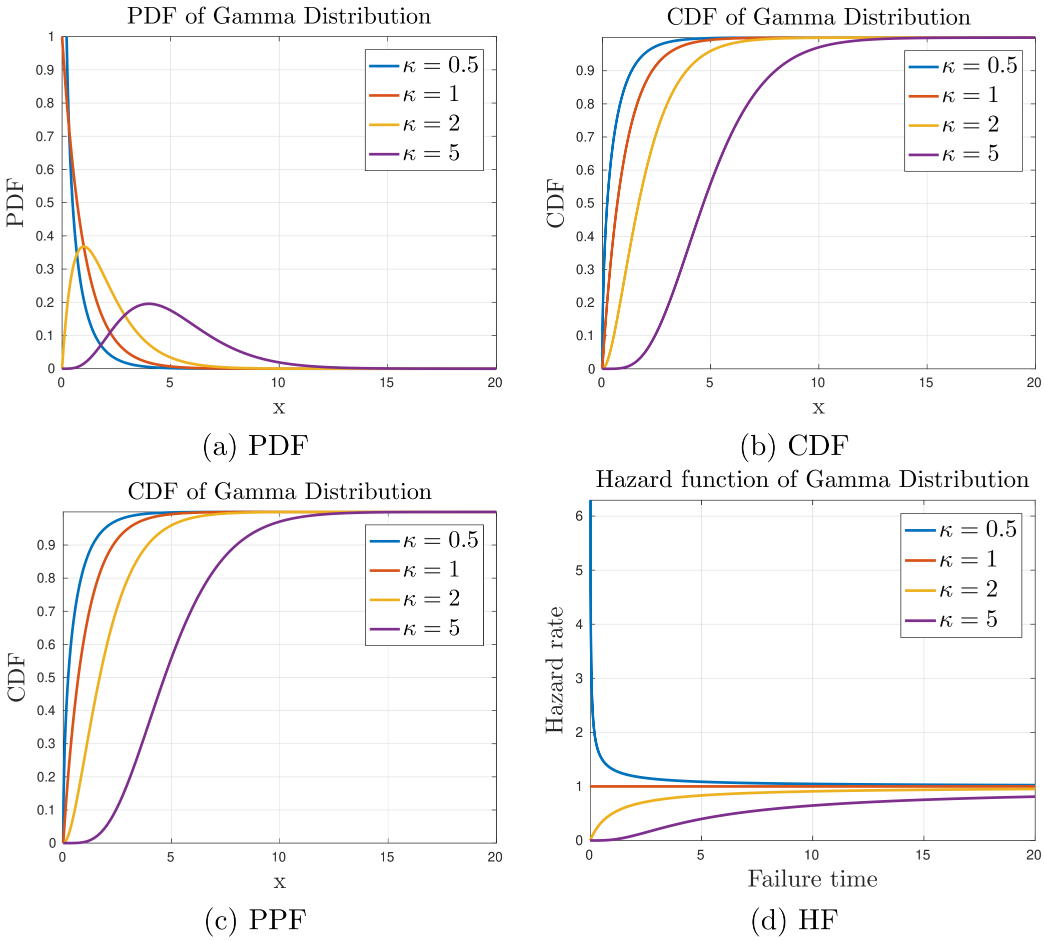
Common Statistics	
Statistic	**Value**
Mean	κ
Mode	$\kappa - 1,\ \kappa \geq 1$
Range	$[0, +\infty)$
Standard Deviation	$\sqrt{\kappa}$
Coefficient of Variation	$\frac{1}{\sqrt{\kappa}}$
Skewness	$\frac{2}{\sqrt{\kappa}}$
Kurtosis coefficient	$3 + \frac{6}{\kappa}$

$$H(x) = -\log\left(1 - \frac{\Gamma_x(\kappa)}{\Gamma(\kappa)}\right), \text{ with } x \geq 0; \kappa > 0 \tag{5.29}$$

where Γ is the gamma function defined above and $\Gamma_x(\alpha)$ is the incomplete gamma function. The incomplete gamma function defined above.
The formula for the survival function of the gamma distribution is

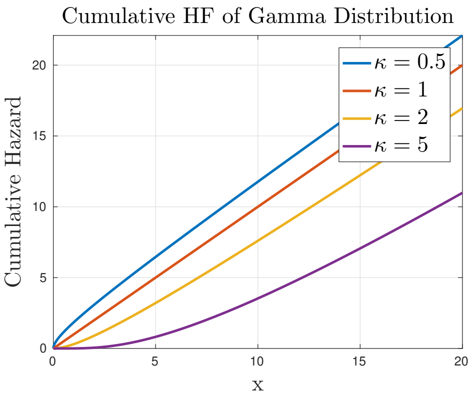
$$S(x) = 1 - \frac{\Gamma_x(\kappa)}{\Gamma(\kappa)}, \text{ with } x \geq 0; \kappa > 0 \tag{5.30}$$

where Γ is the gamma function defined above and $\Gamma_x(\alpha)$ is the incomplete gamma function. The incomplete gamma function defined above.
The gamma inverse survival function does not exist in simple closed form. It is computed numerically.
Some of the most significant measures commonly used in statistics ($\mu = 0$, $\theta = 1$ considered) are presented in Table 5.4.

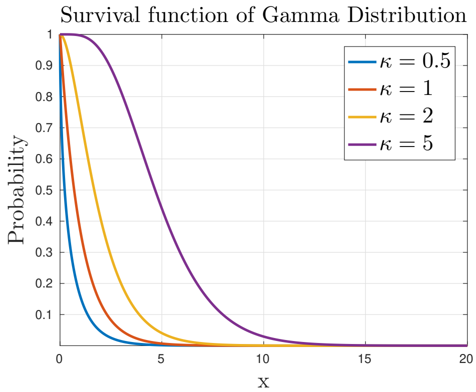


(a) PDF (b) CDF (c) PPF (d) HF

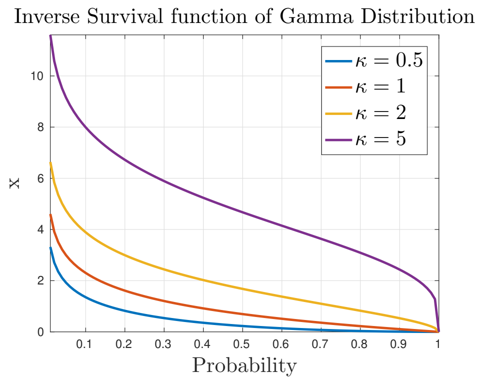
Figure 5.11: Probability functions of Gamma Distribution (a)



(a) CHF



(b) SF



(c) ISF

Figure 5.12: Probability functions of Gamma Distribution (b)

The SEP of the Gamma distribution with scale $\theta > 0$ and shape $\kappa > 0$ is

$$N_X(\theta, \kappa) = \frac{\theta^2 \Gamma^2(\kappa)}{2\pi e} e^{2[(1-\kappa)\psi(\kappa)+\kappa]} \tag{5.31}$$

The FIM of the Gamma distribution with scale $\theta > 0$ and shape $\kappa > 2$ is

$$I_X(\theta, \kappa) = \frac{1}{(\kappa - 2)\,\theta^2} \tag{5.32}$$

and the FSIP is shown in Fig 5.13. In that figure we use a $\theta \in [0.23, 1.2]$ and $\kappa = 5$.

For values of λ less than 1.32 we have a maximization on y-axis (maximization on FIM). For values greater than 1000 we have a maximization on x-axis (maximization on SEP).

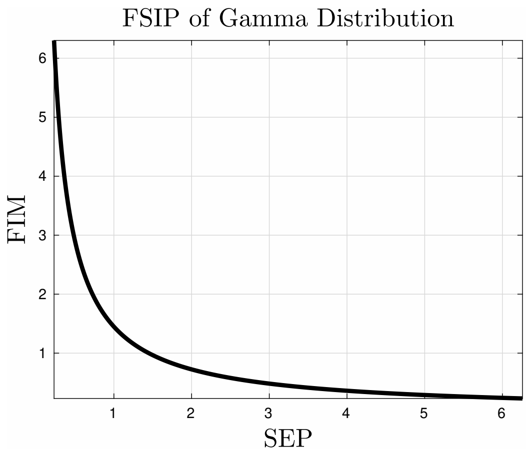


Figure 5.13: Gamma Distribution. FSIP for different θ's.

5.6 Weibull Distribution

The formula for the probability density function of the general Weibull distribution is

$$f(x) = \frac{\kappa}{\lambda}\left(\frac{x-\mu}{\lambda}\right)^{\kappa-1} e^{-\left(\frac{x-\mu}{\lambda}\right)^{\kappa}}, \text{ with } x \geq \mu; \kappa, \lambda > 0 \tag{5.33}$$

where κ is the shape parameter, μ is the location parameter and λ is the scale parameter. The case where $\mu = 0$ and $\lambda = 1$ is called the standard Weibull distribution. The case where $\mu = 0$ is called the 2-parameter Weibull distribution. The equation for the standard Weibull distribution reduces to

$$f(x) = \kappa x^{\kappa-1} e^{-x^{\kappa}}, \text{ with } x \geq 0; \kappa > 0 \tag{5.34}$$

Since the general form of probability functions can be expressed in terms of the standard distribution[1], all subsequent formulas in this section are given for the standard form of the function.
The formula for the cumulative distribution function of the Weibull distribution is

$$F(x) = 1 - e^{-x^{k}}, \text{ with } x \geq 0; \kappa > 0 \tag{5.35}$$

The formula for the percent point function of the Weibull distribution is

$$G(p) = (-\ln(1-p))^{\frac{1}{\kappa}}, \text{ with } 0 \leq p < 1; \kappa > 0 \tag{5.36}$$

The formula for the hazard function of the Weibull distribution is

$$h(x) = \kappa x^{\kappa-1}, \text{ with } x \geq 0; \kappa > 0 \tag{5.37}$$

The formula for the cumulative hazard function of the Weibull distribution is

$$H(x) = x^{\kappa}, \text{ with } x \geq 0; \kappa > 0 \tag{5.38}$$

[1]The standard form of any distribution is the form that has location parameter zero and scale parameter one.

Table 5.5: Common Statistics for Weibull Distribution

Common Statistics	
Statistic	**Value**
Mean	$\Gamma\left(\frac{\kappa+1}{\kappa}\right)$, $\Gamma()$ is the gamma function
Median	$\ln(2)^{\frac{1}{\kappa}}$
Mode	$\left(1-\frac{1}{\kappa}\right)^{\frac{1}{\kappa}}$, if $\kappa > 1$
	0 , if $\kappa \leq 1$
Range	$[0, +\infty)$
Standard Deviation	$\sqrt{\Gamma\left(\frac{\kappa+2}{\kappa}\right) - \left(\Gamma\left(\frac{\kappa+1}{\kappa}\right)\right)^2}$
Coefficient of Variation	$\sqrt{\frac{\Gamma\left(\frac{\kappa+2}{\kappa}\right)}{\left(\Gamma\left(\frac{\kappa+1}{\kappa}\right)\right)^2} - 1}$
Skewness	There is not a closed formula
Kurtosis coefficient	There is not a closed formula

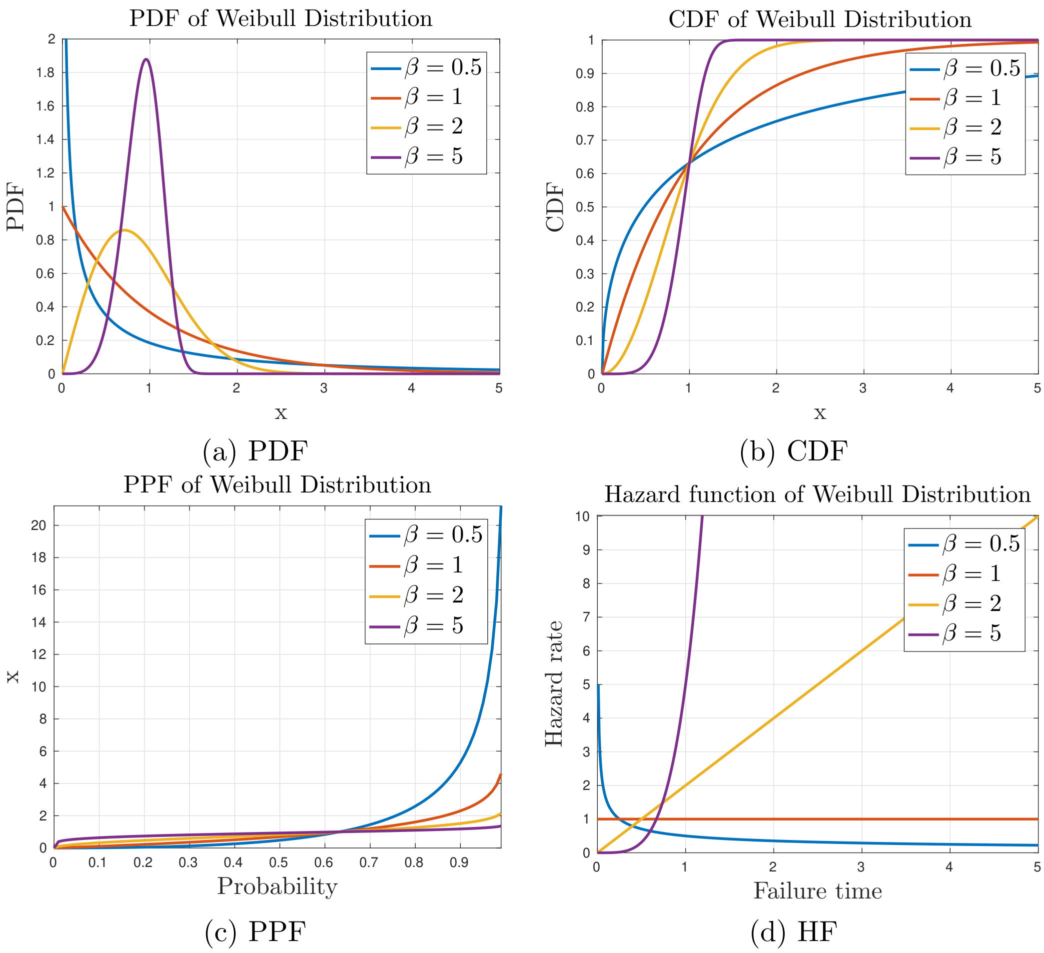
The formula for the survival function of the Weibull distribution is

$$S(x) = e^{-x^{\kappa}}, \text{ with } x \geq 0; \kappa > 0 \tag{5.39}$$

The formula for the inverse survival function of the Weibull distribution is

$$Z(p) = (-\ln(p))^{\frac{1}{\kappa}}, \text{ with } 0 \leq p < 1; \kappa > 0 \tag{5.40}$$

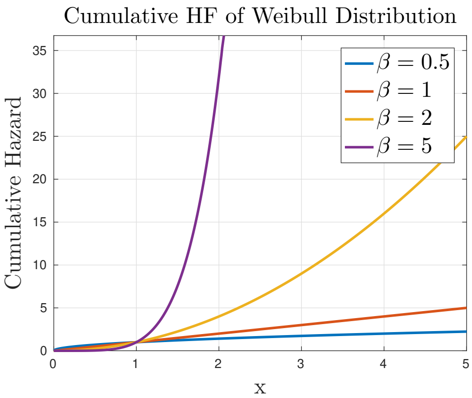
Some of the most significant measures commonly used in statistics for the standard Weibull distribution are presented in Table 5.5.



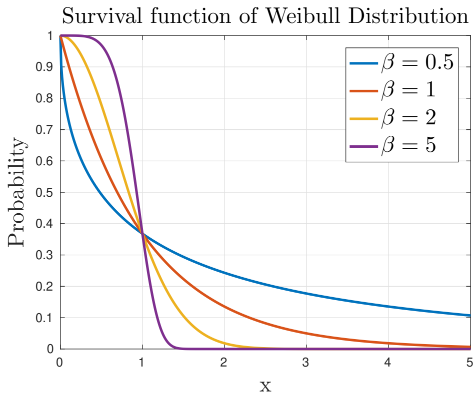
(a) PDF (b) CDF

(c) PPF (d) HF

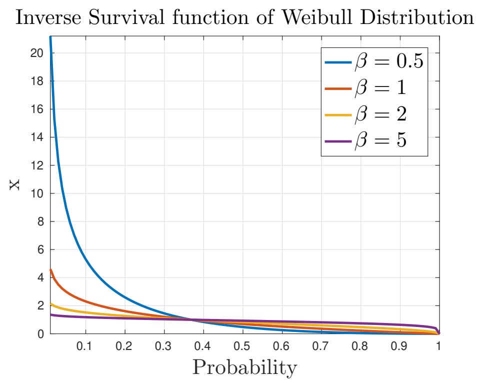
Figure 5.14: Probability functions of Weibull Distribution (a)



(a) CHF



(b) SF



(c) ISF

Figure 5.15: Probability functions of Weibull Distribution (b)

The SEP of the Weibull distribution with location μ , scale $\lambda > 0$ and shape $\kappa > 0$ is

$$N_X(\lambda,\kappa) = \frac{(1-\alpha)^2 \lambda^2 e}{2\pi} e^{2\alpha\gamma}, \tag{5.41}$$

where $\alpha = \frac{k-1}{k}$ and γ is the Euler-Mascheroni constant.

The FIM of the Weibull distribution with location μ , scale $\lambda > 0$ and shape $\kappa > 2$ is

$$I_X(\lambda,\kappa) = \frac{\alpha^2}{(1-\alpha)^2 \lambda^2} \Gamma(2\alpha - 1) \tag{5.42}$$

and the Fisher Shannon Information Plane is shown in Fig 5.16. In that figure we use a $\gamma = 0.5772156649$ which is the gamma Euler-mascheroni constant, the scale parameter $\lambda \in [1.32, 7.7]$, shape parameter $\kappa = 2.5$ and $\alpha = \frac{\kappa-1}{\kappa}$. For values of λ less than 1.32 we have a maximization on y-axis (maximization on FIM). For values greater than 7.7 we have a maximization on x-axis (maximization on SEP).

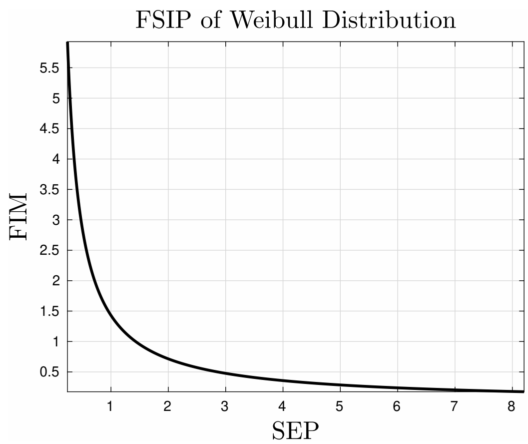


Figure 5.16: Weibull Distribution.FSIP for different λ's.

5.7 Log-Normal Distribution

A variable X is log-normally distributed if $Y = ln(X)$ is normally distributed with ln denoting the natural logarithm. The general formula for the probability density function of the lognormal distribution is

$$f(x) = \frac{e^{-\left(\frac{\ln\left(\frac{(x-\theta)}{m}\right)^2}{2\sigma^2}\right)}}{(x-\theta)\sigma\sqrt{2\pi}}, \text{ with } x > \theta; m, \sigma > 0 \tag{5.43}$$

where σ is the shape parameter (and is the standard deviation of the log of the distribution), θ is the location parameter and m is the scale parameter (and is also the median of the distribution).
If $x = \theta$, then $f(x) = 0$. The case where $\theta = 0$ and $m = 1$ is called the standard lognormal distribution. The case where θ equals zero is called the 2-parameter lognormal distribution, and the formula for the standard lognormal distribution is

$$f(x) = \frac{e^{-\left(\frac{\ln(x)^2}{2\sigma^2}\right)}}{x\sigma\sqrt{2\pi}}, \text{ with } x > 0; \sigma > 0 \tag{5.44}$$

Since the general form of probability functions can be expressed in terms of the standard distribution, all subsequent formulas in this section are given for the standard form of the function.
The lognormal distribution is commonly parameterized with

$$\mu = \log(m) \tag{5.45}$$

The parameter μ is the mean of the logarithmic distribution. If the μ parameterization is used, the lognormal pdf is

$$f(x) = \frac{e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}}}{x\sigma\sqrt{2\pi}}, \text{ with } x > 0; \sigma > 0 \tag{5.46}$$

Generally to use the m parameterization since m is an explicit scale parameter.

The formula for the cumulative distribution function of the lognormal distribution is

$$F(x) = \Phi\left(\frac{\ln(x)}{\sigma}\right), \text{ with } x > 0; \sigma > 0 \tag{5.47}$$

where Φ is the cumulative distribution function of the normal distribution.
The formula for the percent point function of the lognormal distribution is

$$G(p) = e^{\sigma\Phi^{-1}(p)}, \text{ with } 0 \leq p < 1; \sigma > 0 \tag{5.48}$$

where Φ^{-1} is the percent point function of the normal distribution.
The formula for the hazard function of the lognormal distribution is

$$h(x) = \frac{\left(\frac{1}{x\sigma}\right)\phi\left(\frac{\ln x}{\sigma}\right)}{\Phi\left(\frac{-\ln x}{\sigma}\right)}, \text{ with } x > 0; \sigma > 0 \tag{5.49}$$

where ϕ is the probability density function of the normal distribution and Φ is the cumulative distribution function of the normal distribution.
The formula for the cumulative hazard function of the log-normal distribution is

$$H(x) = -\ln\left(1 - \Phi\left(\frac{\ln x}{\sigma}\right)\right), \text{ with } x > 0; \sigma > 0 \tag{5.50}$$

where Φ is the cumulative distribution function of the log-normal distribution.
The formula for the survival function of the log-normal distribution is

$$S(x) = 1 - \Phi\left(\frac{\ln x}{\sigma}\right), \text{ with } x > 0; \sigma > 0 \tag{5.51}$$

where Φ is the cumulative distribution function of the log-normal distribution.
The formula for the inverse survival function of the lognormal distribution is

$$Z(p) = e^{\sigma\Phi^{-1}(1-p)}, \text{ with } 0 \leq p < 1; \sigma > 0 \tag{5.52}$$

Table 5.6: Common Statistics for Log-Normal Distribution

Common Statistics	
Statistic	**Value**
Mean	$e^{\frac{\sigma^2}{2}}$
Median	Scale parameter m
Mode	$\frac{1}{\sigma^2}$
Range	$[0, +\infty)$
Standard Deviation	$\sqrt{e^{\sigma^2}\left(e^{\sigma^2}-1\right)}$
Coefficient of Variation	$\sqrt{e^{\sigma^2}-1}$
Skewness	$\left(e^{\sigma^2}+2\right)\sqrt{e^{\sigma^2}-1}$
Kurtosis coefficient	$\psi^4+2\psi^3+3\psi^2-3$
	where $\psi = e^{\sigma^2}$

where Φ^{-1} is the percent point function of the normal distribution.
Some of the most significant measures commonly used in statistics for the standard log-normal distribution are presented in Table 5.6.

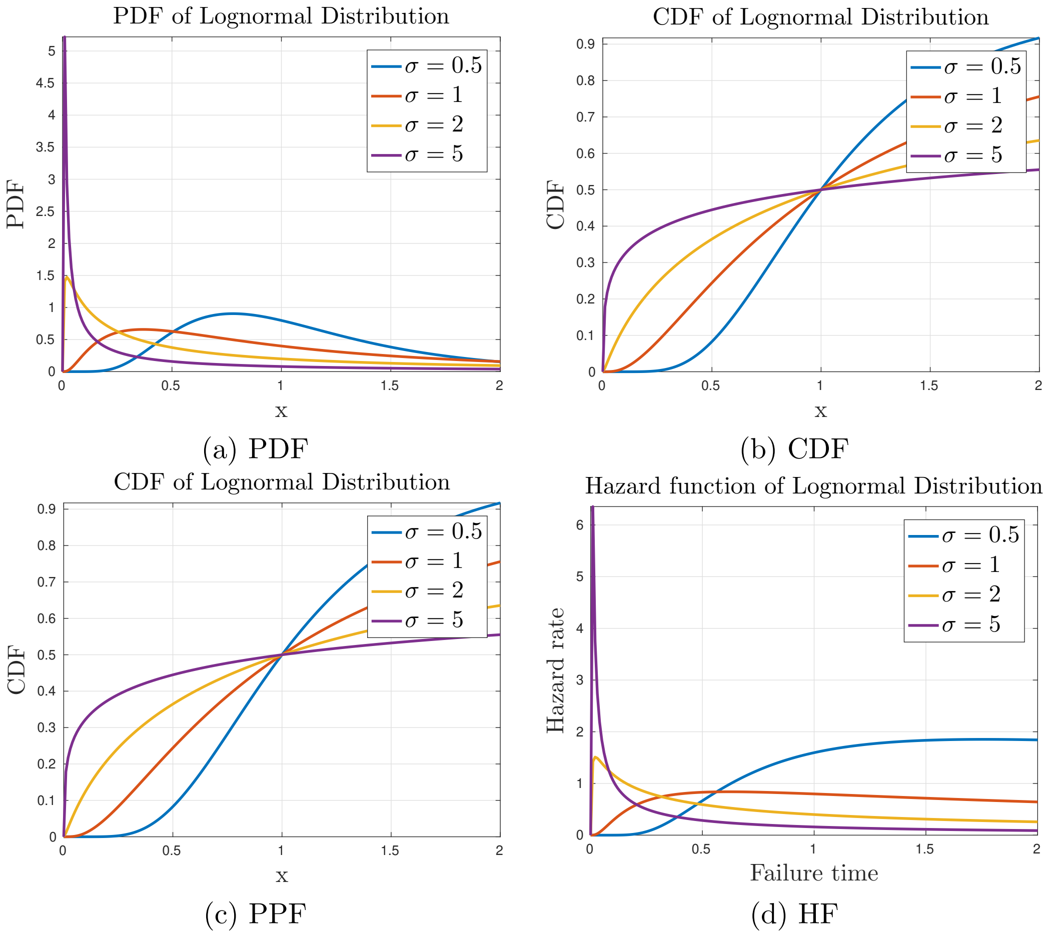
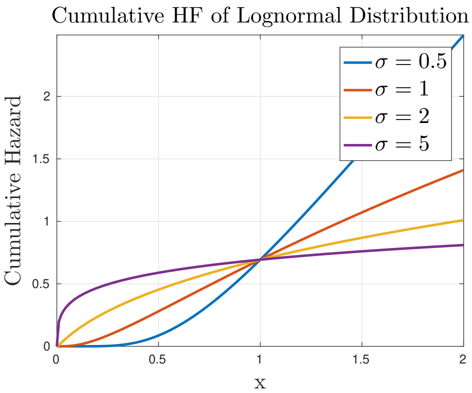
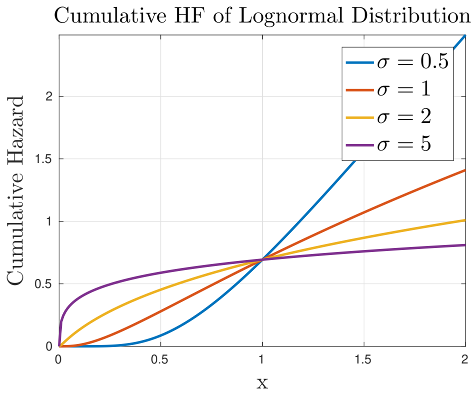


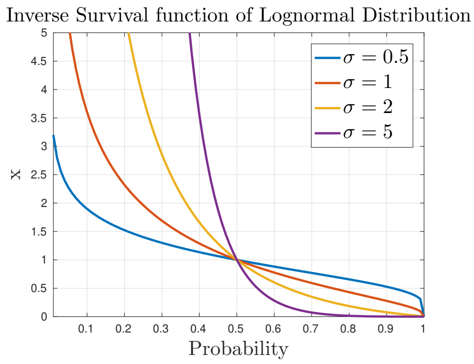
Figure 5.17: Probability functions of Log-Normal Distribution (a)



(a) CHF



(b) SF



(c) ISF

Figure 5.18: Probability functions of Log-Normal Distribution (b)

The SEP and the FIM of the log-normal distribution with μ and $\sigma > 0$ are given by

$$N_X(\mu, \sigma) = \sigma^2 e^{2\mu} \tag{5.53}$$

$$I_X(\mu, \sigma) = \left(1 + \frac{1}{\sigma^2}\right) e^{2\left(\sigma^2 - \mu\right)} \tag{5.54}$$

and the FSIP is shown in Fig 5.19. In that figure we use a standard deviation $\sigma \in [0.2, 1.2]$. For values of σ less than 0.2 we have a maximization on y-axis (maximization on FIM). For values greater than 1.2 we have a maximization on x-axis (maximization on SEP) till SEP = 400 and simultaneously maximization on y-axis (FIM). The location parameter is $\mu = 0.7$ in this plot.

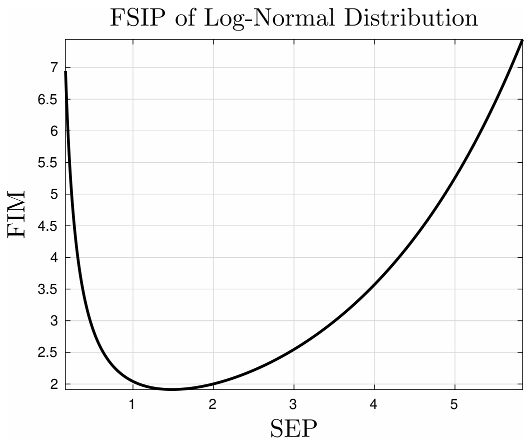


Figure 5.19: Log-Normal Distribution. FSIP for different σ's.

5.8 Uniform Distribution

The general formula for the probability density function PDF of the uniform distribution is the following.

$$f(x;a,b) = \begin{cases} \frac{1}{b-a}, & \text{for } a \le x \le b \\ 0, & \text{otherwise} \end{cases} \tag{5.55}$$

where a is the lower bound of the interval (minimum value), b is the upper bound of the interval (maximum value), where $b > a$.
Since the general form of probability functions can be expressed in terms of the standard distribution, all subsequent formulas in this section are given for the standard form of the function.
The formula for the CDF of the uniform distribution is

$$F(x;a,b) = \begin{cases} 0, & \text{for } x < a \\ \frac{x-a}{b-a}, & \text{for } a \le x \le b \\ 1, & \text{for } x > b \end{cases} \tag{5.56}$$

where a is the lower bound of the interval (minimum value), b is the upper bound of the interval (maximum value), where $b > a$.
The formula for the percent point function of the uniform distribution does not exist in a simple closed form. It is computed numerically.

$$F^{-1}(p;a,b) = a + p(b-a) \quad \text{for } p \in [0,1] \tag{5.57}$$

The formula for the hazard function of the uniform distribution is

$$h(x;a,b) = \frac{f(x;a,b)}{S(x;a,b)} = \begin{cases} \frac{1}{b-x}, & \text{for } a \le x < b \\ 0, & \text{otherwise} \end{cases} \tag{5.58}$$

The formula for the cumulative hazard function of the uniform distribution is

$$h^{-1}(z;a,b) = b - \frac{1}{z} \tag{5.59}$$

Table 5.7: Common Statistics for Uniform Distribution

Common Statistics	
Statistic	**Value**
Mean	$\mu = \frac{a+b}{2}$
Mode	None
Range	$b - a$
Standard Deviation	$\frac{b-a}{\sqrt{12}}$
Coefficient of Variation	$\frac{\sigma}{\mu} = \frac{2(b-a)}{\sqrt{12}(a+b)}$
Skewness	0
Kurtosis (Excess)	$-\frac{6}{5}$
Total Kurtosis	$\frac{9}{5}$

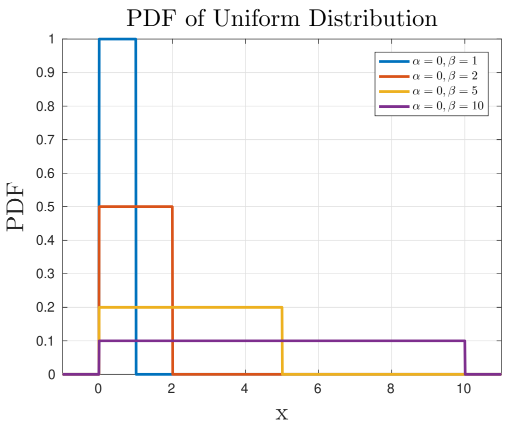
where b is the upper bound of the interval (maximum value) and z is the CDF of uniform distribution.
The formula for the survival function of the uniform distribution is as follows.

$$S(x; a, b) = 1 - F(x; a, b) = \begin{cases} 1, & \text{for } x < a \\ \frac{b-x}{b-a}, & \text{for } a \le x \le b \\ 0, & \text{for } x > b \end{cases} \tag{5.60}$$

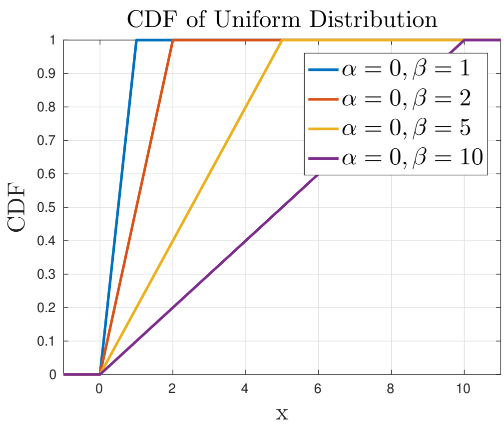
where a is the lower bound of the interval (minimum value), b is the upper bound of the interval (maximum value), where $b > a$.
The uniform inverse survival function does not exist in a simple closed form. It is computed numerically.

$$S^{-1}(q; a, b) = a + (1 - q)(b - a) \quad \text{for } q \in [0, 1] \tag{5.61}$$

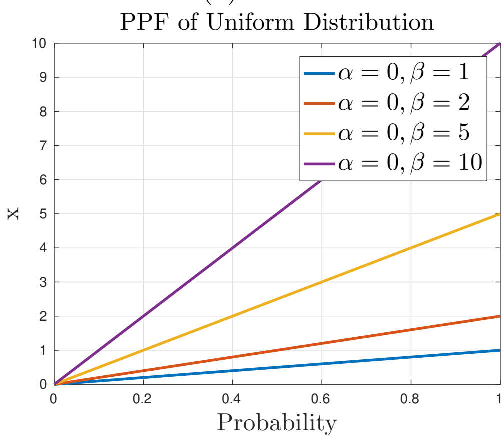
Some of the most significant measures commonly used in statistics for a standard uniform distribution are presented in Table 5.7.



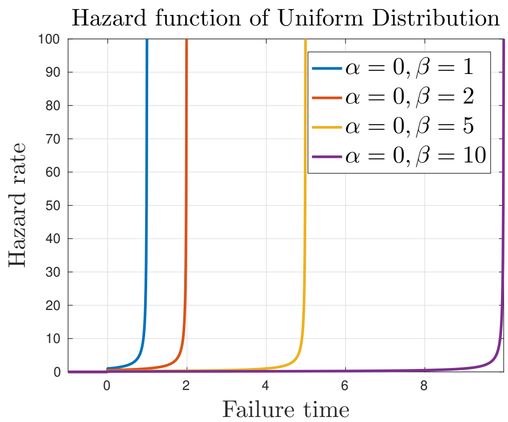
(a) PDF



(b) CDF

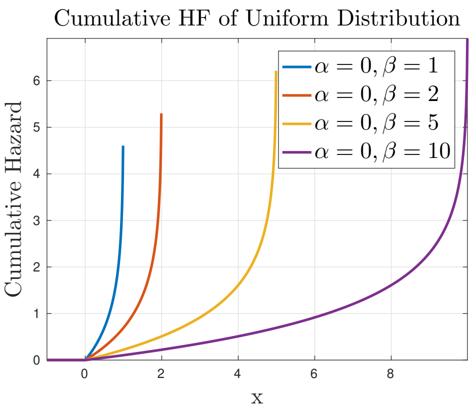


(c) PPF

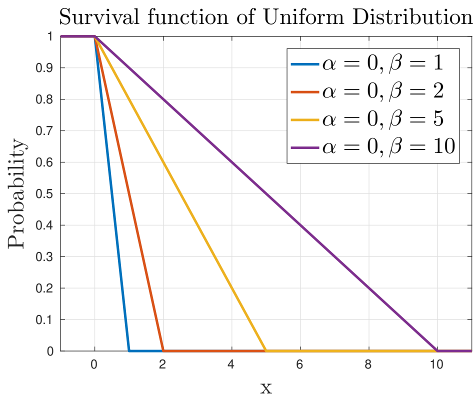


(d) HF

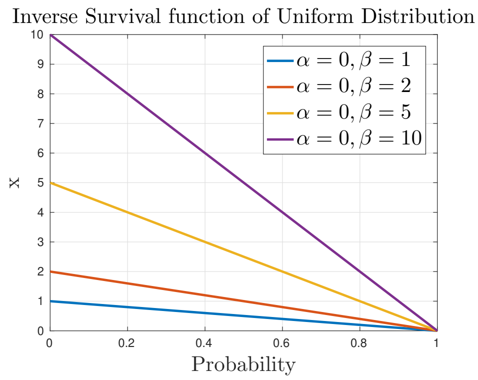
Figure 5.20: Probability functions of Uniform Distribution (a)



(a) CHF



(b) SF



(c) ISF

Figure 5.21: Probability functions of Uniform Distribution (b)

The SEP of the Uniform distribution has a unique characteristic. There is no formula for calculation of SEP and FIM. Its form make the process for computing the FSIP through the Kernel Density Estimation a challenging task.

Chapter 6

Numerical Investigations of FSIP

6.1 Introduction

First of all, it is useful to determine the behavior of FSIP in case we have only the expected response with a varied amplitude. After that we have to figure out the FSIP behavior for different probability distribution models. For that step, we generate random numbers with the desired distribution and then we compute Fisher-Shannon (FS) values in order to understand the respective behavior in FSIP for certain parameters of every distribution. After that we generate synthetic data with the linear model described above 3.9 for a certain amplitude α with additive noise of every distribution that we analyze in Chap. 5.

6.2 FSIP for Expected fMRI Response

In order to see the FSIP for the expected response, we generate instances of the function $S(t) = \alpha\, E(t)$. The scalar value α is in the range of $[15 \times 10^{-2}, 10]$. As we see in Fig. 6.1, the computation of FIM and SEP is below the Gaussian limit.

For the expected response, varying the values of α results in distinct combinations of the values of FIM and SEP. For small α values, the FSIP shows high FIM and low SEP values, indicative of distributions with high precision and low entropy, often found in deterministic or structured systems with localized information.

As α increases and reaches a value of 1.5, the FSIP indicates a big decrease on FIM. At that point, the SEP value starts to increase.

For $\alpha > 2$, there is a significant decrease in FIM (values near zero) and an increase in SEP, characterizing broad and uncertain distributions with low precision, typical of systems with significant randomness or uncertainty.

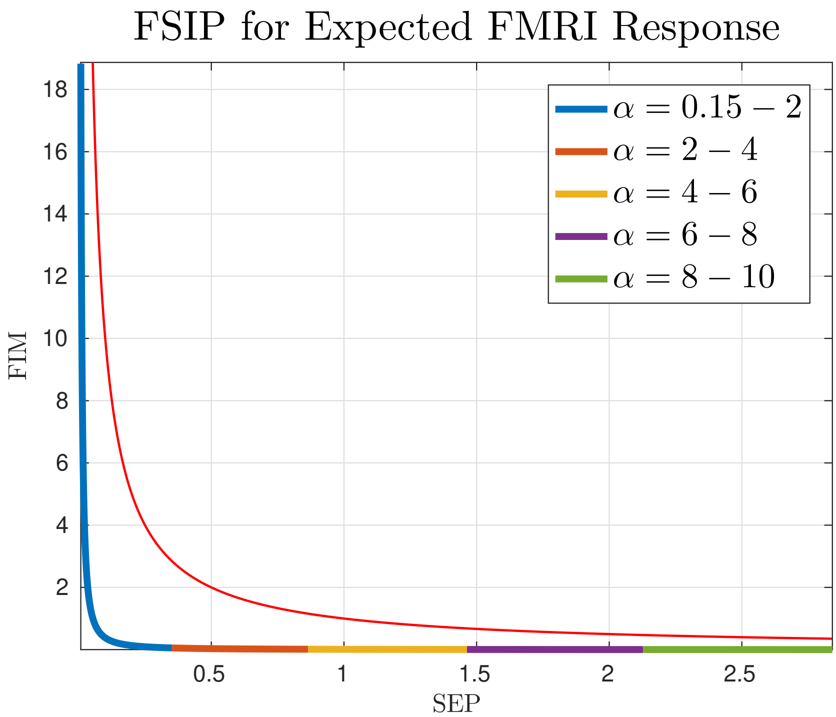


Figure 6.1: FSIP for expected FMRI response

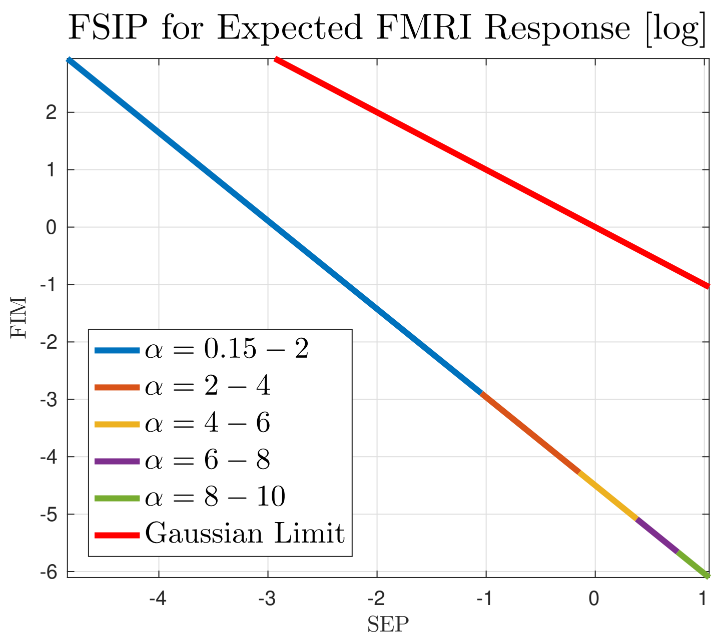


Figure 6.2: FSIP for expected fMRI response (Logarithmic values)

6.3 FSIP for Randomly Distributed Numbers

In that section, we compute the FS values come from randomly distributed numbers and compare them with the theoretical FSIP from Chap. 5. We determine whether various parameters that describe these distributions affects the FSIP behavior. At first we generate random numbers that follow a certain distribution with built-in function of matlab for Gaussian, Student-t, Log-Normal, Gamma, and Weibull. For the case of Power Exponential distribution for which there is no built-in function inside Matlab, we use Inverse Transform Sampling [1]. More information about that method is presented in the Appendix of that thesis.

[1]Inverse transform sampling, also known as the inverse transform method or inverse CDF sampling, is a technique used to generate random samples from a probability distribution given its cumulative distribution function (CDF).

6.3.1 Normal Distribution

We remind the reader that the formula of normal distribution is

$$f(x) = \frac{e^{-(x-\mu)^2/(2\sigma^2)}}{\sigma\sqrt{2\pi}}, \tag{6.1}$$

where μ is the location parameter(mean) and σ is the scale parameter.

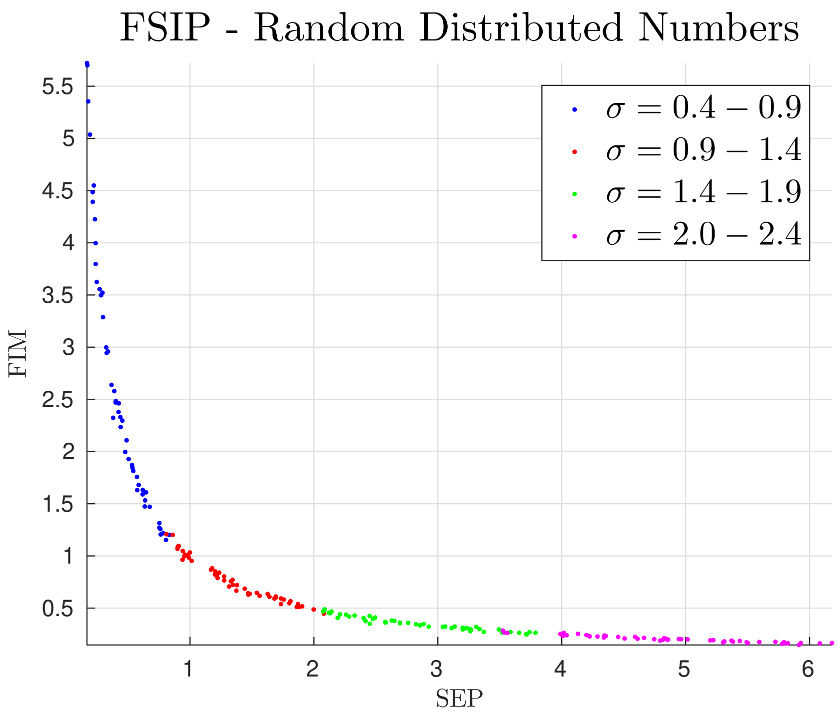


Figure 6.3: FSIP for Normal Distributed Numbers

For the normal distribution, we can see that for various standard deviation σ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for small σ values, the FSIP gives high FIM values and small SEP values. That combination in FSIP, represents distributions that are concentrated with high precision but low entropy, often seen in deterministic or highly structured systems where information is localized.

As σ increases for $\sigma \approx 1.5$, we observe that the plane gives us values around the Gaussian limit, the lower bound that any random variable could reach.

For σ values greater than 2 we observe that there is a minimization on FIM and a maximization on SEP. That situation suggests broad and highly uncertain distributions with low precision, which may indicate systems with significant randomness or uncertainty.

6.3.2 Student-t Distribution

We remind the reader that the formula of Student-t distribution is

$$f(x) = \frac{\left(1 + \frac{x^2}{m}\right)^{\frac{-(m+1)}{2}}}{B\left(\frac{1}{2}, \frac{1}{2}m\right)\sqrt{m}}, \tag{6.2}$$

where B is the beta function and m is a positive integer shape parameter(degrees of freedom).

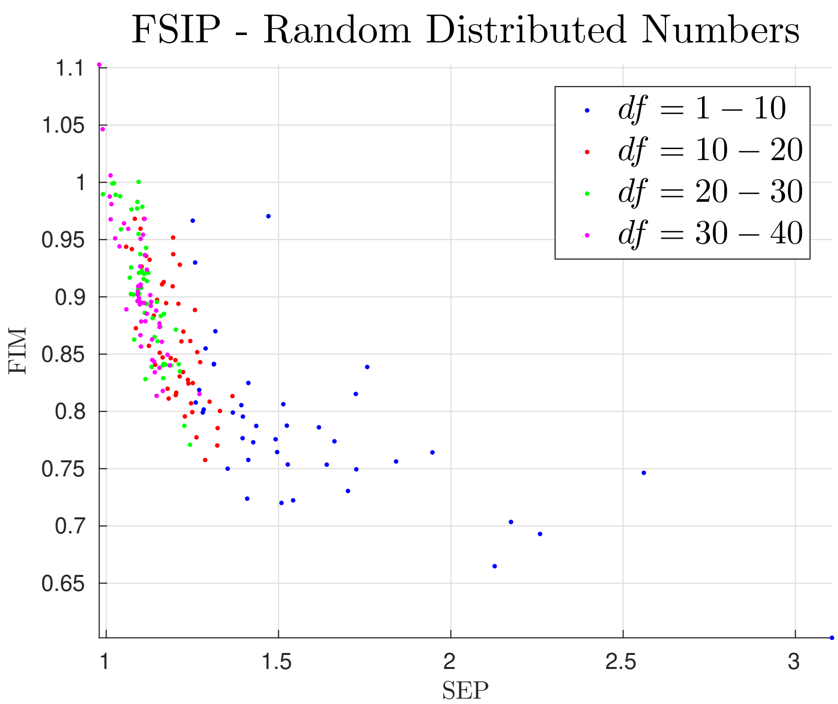


Figure 6.4: FSIP for Student-t Distributed Numbers

For the Student-t distribution, we can see that for various degrees of freedom $df = m$ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for small m values FSIP leads to a situation in which FIM and SEP values are far from the Gaussian limit and show high values concurrently for FIM and SEP, so complexity is $C > 1$. As m increases, our distribution comes close to the Gaussian case.

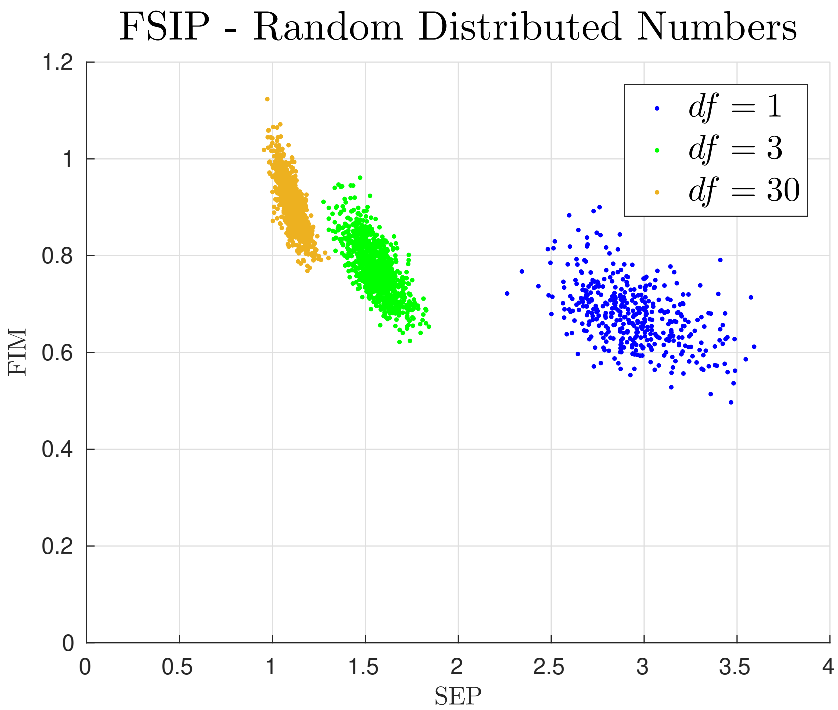


Figure 6.5: FSIP for Student-t Distributed Numbers

In Fig. 6.5, we can observe various situations in a large amount of time series. For df values 1 or 3 we observe a higher variance of FSIP values and that values are far from the Gaussian limit. As we reach a significantly higher value for degrees of freedom, we observe that the outcome of FSIP approaches the Gaussian case. Hence, the tails of the distribution become less heavy.

In comparison with the Gaussian case, we observe that the plot is over the Gaussian limit, and the curve has a lower rate of change.

6.3.3 Power Exponential Distribution

We remind the reader that the formula of the power exponential distribution is

$$f(x) = \frac{\gamma \lambda^{\frac{1}{\gamma}}}{2\Gamma\left(\frac{1}{\gamma}\right)} e^{-\lambda|x-\mu|^{\gamma}}, \text{ with } x, \mu \in \mathbb{R}; \lambda > 0, \gamma > 1, \tag{6.3}$$

where μ is the location parameter, $\lambda > 0$ is a scale parameter and $\gamma > 1$ is a shape parameter.

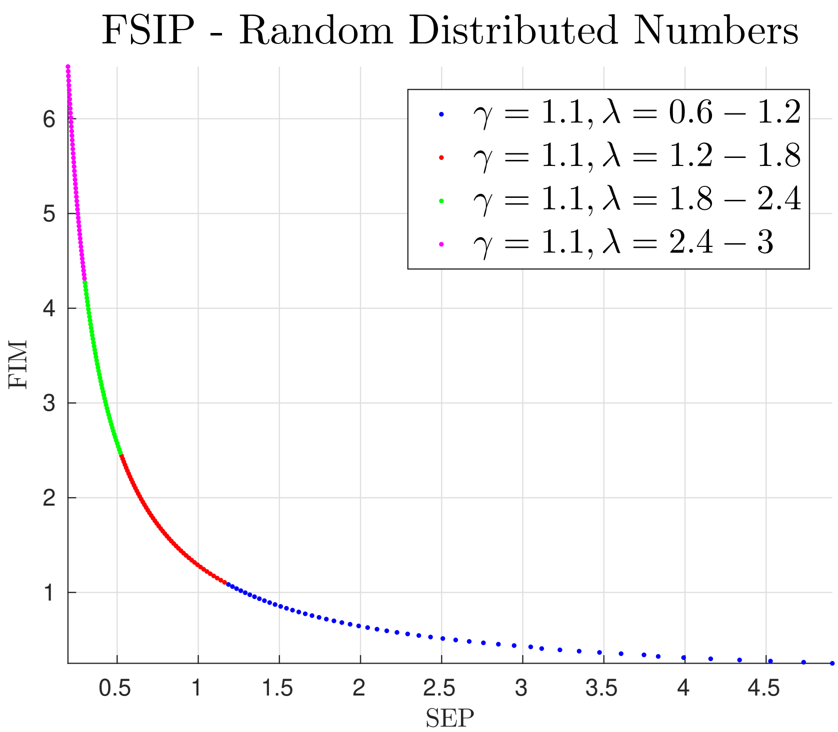


Figure 6.6: FSIP for Power Exponential Distributed Numbers

For the power exponential distribution, whose random numbers were generated using the inverse transform sampling method and with an approximate random number generator, we observe a significant deviation compared to the previous two distributions.

However, we can say that for various shape parameter γ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for γ values in range $[4.5, 3.5]$, the FSIP gives high FIM values and small SEP values. That combination in FSIP, represents distributions that are concentrated with high precision but low entropy, often seen in deterministic or highly structured systems where information is localized.

Within range $[1.5, 3.5]$ we see a situation that FIM and SEP values are close to each other. That indicates moderate precision and entropy.

For γ values in the range between $[0.5, 1.5]$ we observe that there is a minimization in FIM at first. A maximization on both FIM and SEP is following. That situation suggests broad and highly uncertain distributions with high precision.

6.3.4 Gamma Distribution

We remind the reader that the formula of the gamma distribution is

$$f(x) = \frac{\left(\frac{x-\mu}{\lambda}\right)^{\kappa-1} e^{-\frac{x-\mu}{\lambda}}}{\lambda\Gamma(\kappa)}, \text{ with } x \geq \mu; \kappa, \lambda > 0 \tag{6.4}$$

where κ is the shape parameter, μ is the location parameter, λ is the scale parameter and Γ is the Gamma function.

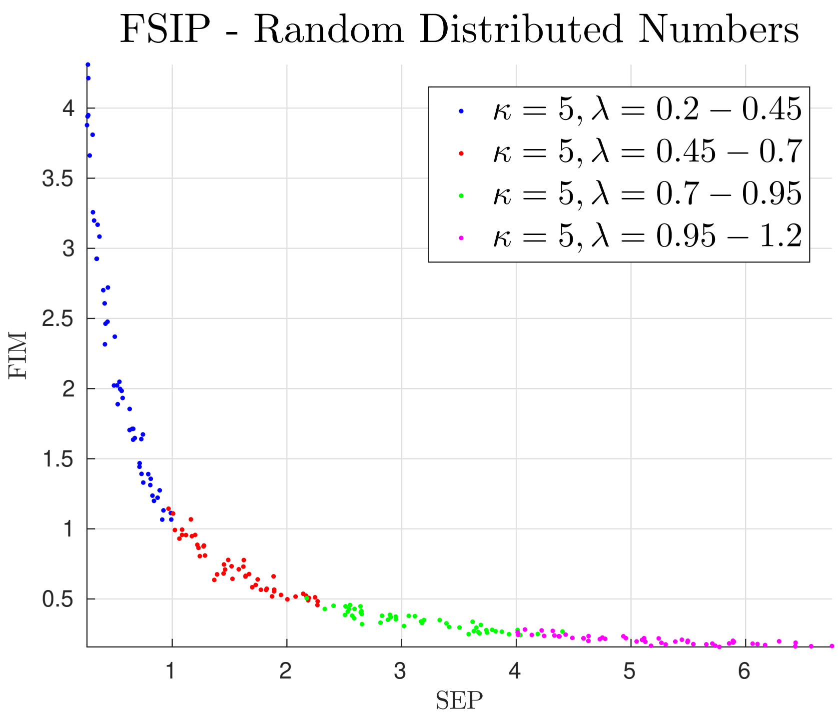


Figure 6.7: FSIP for Gamma Distributed Numbers

For the gamma distribution, we can see that for various scale λ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for small λ values, the FSIP gives high FIM values and small SEP values. That combination in FSIP, represents distributions that are concentrated with high precision but low entropy, often seen in deterministic or highly structured systems where information is localized.

As λ increases, for $\lambda \approx 0.5$, we observe that plane start to increase SEP values with a concurrently decrease on FIM values.

For λ values greater than 0.7 we observe that there is a minimization on FIM and a maximization on SEP. That situation suggests broad and highly uncertain distributions with low precision, which may indicate systems with significant randomness or uncertainty.

6.3.5 Weibull Distribution

We remind the reader that the formula of the Weibull distribution is

$$f(x) = \frac{\kappa}{\lambda}\left(\frac{x-\mu}{\lambda}\right)^{\kappa-1} e^{-\left(\frac{x-\mu}{\lambda}\right)^{\kappa}}, \text{ with } x \geq \mu; \kappa, \lambda > 0 \tag{6.5}$$

where κ is the shape parameter, μ is the location parameter and λ is the scale parameter.

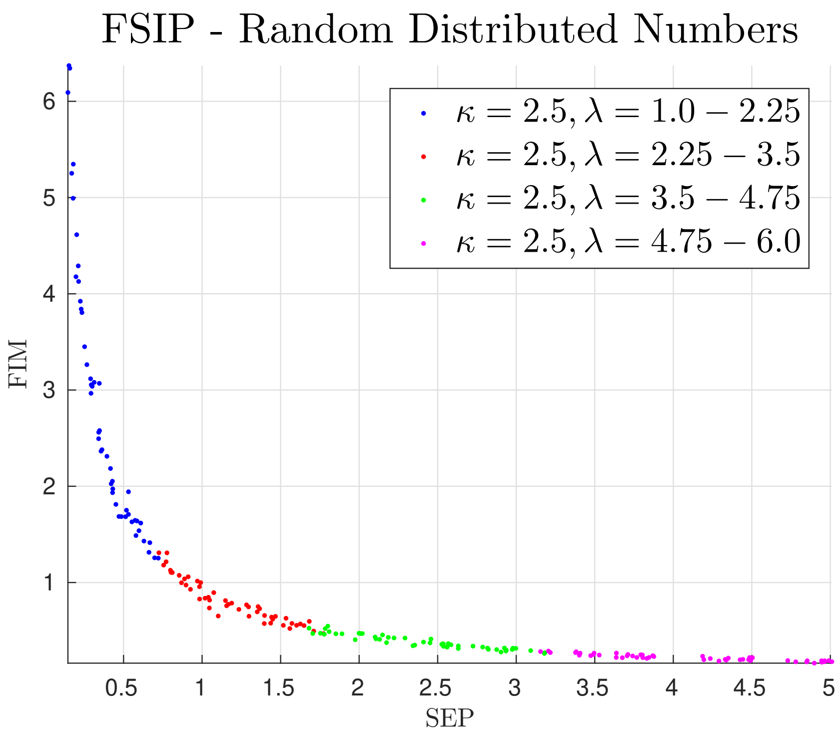


Figure 6.8: FSIP for Weibull Distributed Numbers

For the Weibull distribution, we can see that for various scale λ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for small λ values, the FSIP gives high FIM values and small SEP values. That combination in FSIP, represents distributions that are concentrated with high precision but low entropy, often seen in deterministic or highly structured systems where information is localized.

As λ increases, for $\lambda \approx 2.5$, we observe that plane start to increase SEP values with a concurrently decrease on FIM values.

For λ values greater than 2.5 we observe that there is a minimization on FIM and a maximization on SEP. That situation suggests broad, highly uncertain distributions with low precision, which may indicate systems with significant randomness or uncertainty.

6.3.6 Log-Normal Distribution

We have to remind the reader that the formula of log-normal distribution is

$$f(x) = \frac{e^{-\left(\frac{\ln\left(\frac{(x-\theta)}{m}\right)^2}{2\sigma^2}\right)}}{(x-\theta)\sigma\sqrt{2\pi}}, \text{ with } x > \theta; m, \sigma > 0 \tag{6.6}$$

where σ is the shape parameter (and is the standard deviation of the log of the distribution), θ is the location parameter and m is the scale parameter (and is also the median of the distribution).

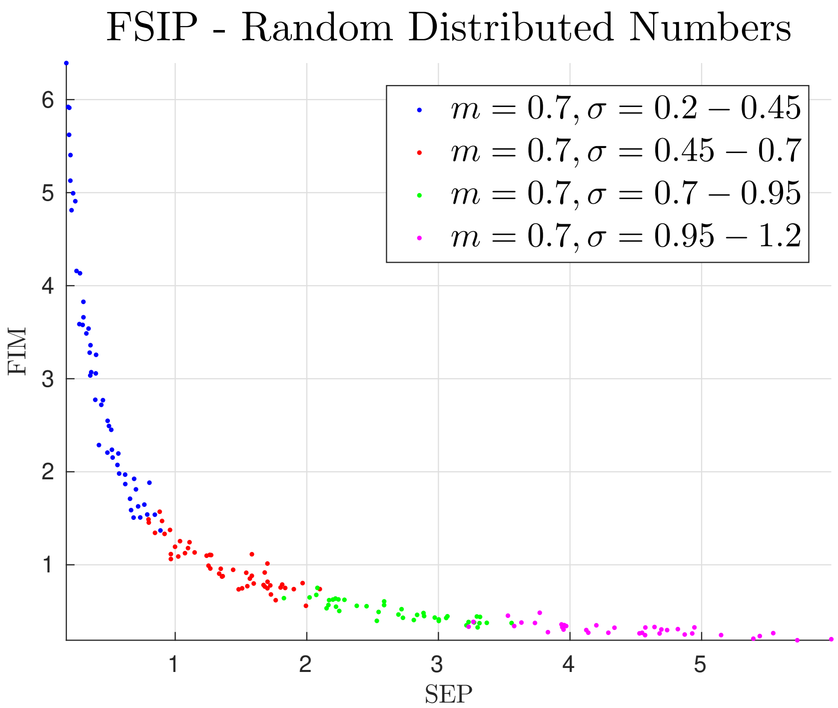


Figure 6.9: FSIP for Log-Normal Distributed Numbers

For the log-normal distribution, we can see that for various standard deviations of logarithmic σ values there is a different combination of FIM and SEP values. From the scatter plot, we can observe that for small σ values, the FSIP gives high FIM values and small SEP values. That combination in FSIP, represents distributions that are concentrated with high precision but low entropy, often seen in deterministic or highly structured systems where information is localized.

As σ increases, for $\sigma \approx 0.5$, we observe that plane start to increase SEP values with a concurrently decrease on FIM values.

For σ values greater than 0.7 we observe that there is a minimization on FIM and a maximization on SEP. That situation suggests broad and highly uncertain distributions with low precision, which may indicate systems with significant randomness or uncertainty.

6.3.7 Uniform Distribution

We remind the reader that the formula of log-normal distribution is

$$f(x; a, b) = \begin{cases} \frac{1}{b-a}, & \text{for } a \leq x \leq b \\ 0, & \text{otherwise} \end{cases} \tag{6.7}$$

where a is the lower bound of the interval (minimum value), b is the upper bound of the interval (maximum value), where $b > a$.

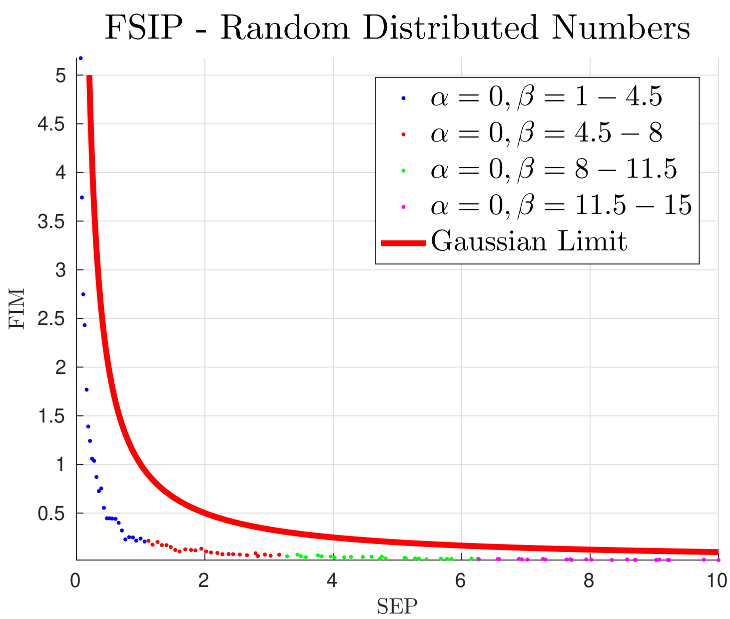


Figure 6.10: FSIP for Uniform Distributed Numbers

In Fig. 6.10 we observe that all FSIP values are below the Gaussian limit. The issue in that case aligned with the KDE and whether a PDF is differentiable. Functions with sharp corners (Delta function) or Uniform distribution with a sharp cut-off are not differentiable at certain points because they either have discontinuities or are not smooth.

6.4 FSIP for Synthetic fMRI Data using Linear Model

In that section we generate fMRI synthetic data using the linear model described in Sect. 3.3. Therefore, we use Eq. 3.9 and we study the FSIP behavior for a certain value α and parameterize different types of noise $n(t)$. That noise includes numbers randomly distributed in Gaussian, Student-t, Power Exponential, Gamma, Weibull and Log-Normal distributions in form

$$D(t) = \alpha \times E(t) + n(t) \tag{6.8}$$

and with computations using Matlab, we take out some significant outcome about behavior of Expected Response and the various distributions by using SEP, FIM and Fisher Shannon Information Plane.

6.4.1 Normally Distributed Noise

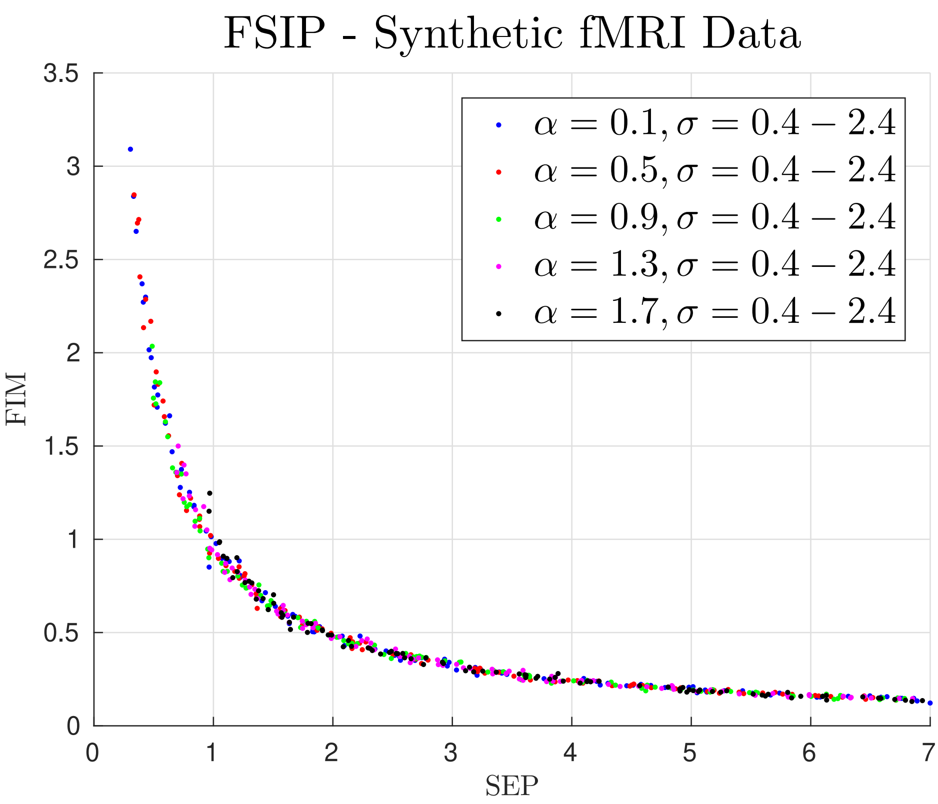


Figure 6.11: FSIP with normally distributed noise.

In Fig. 6.11 we can see expected fMRI response, mixed with normal noise with various standard deviations. As we can see in the graph, for small values of α there are significantly higher values of FIM. For every scalar α, when increased, we see that FIM values decreased and there is a maximization on SEP values. For every α we see that as the standard deviation of noise increases, we are driven from high FIM-low SEP values to low FIM -high SEP values. So we understand that we are going from a high-precision/low-entropy situation that is commonly seen in deterministic or highly structured systems to a distribution with low precision which may indicate significant randomness or uncertainty.

6.4.2 Student-t Distributed Noise

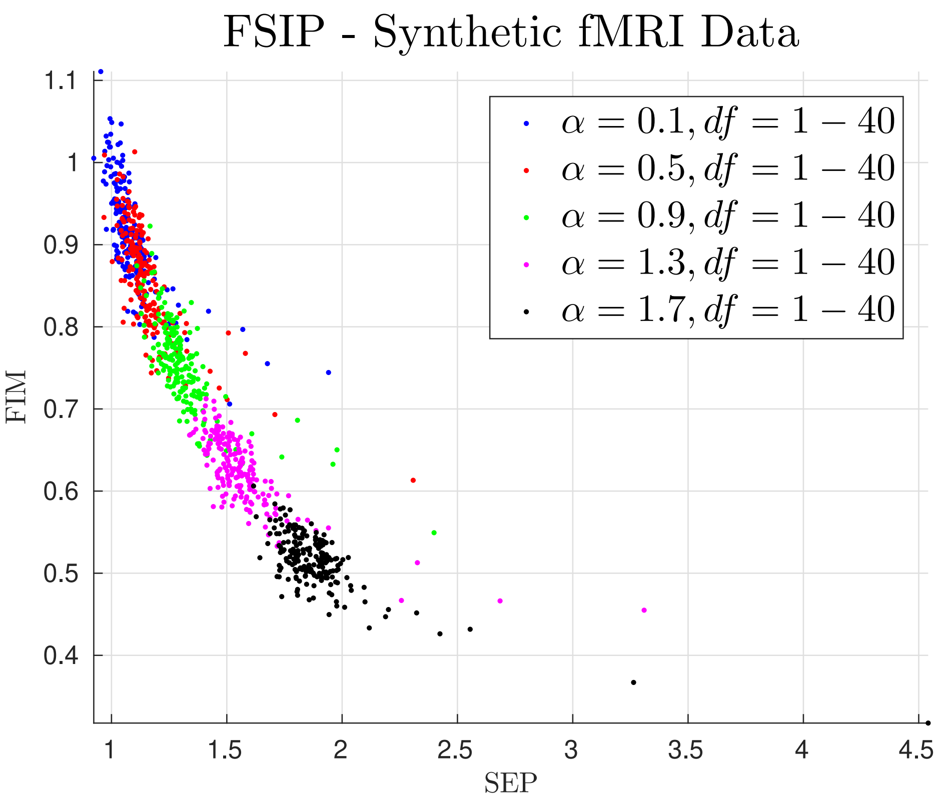


Figure 6.12: FSIP with Student-t Distributed Noise

In Fig. 6.12 we can see expected fMRI response, mixed with Student-t noise with various degrees of freedom. Like in every section with synthetic data, for small values of α we have a high FIM-low SEP values situation. As α increases we are going to a low FIM-high SEP situation. On the other hand, as degrees of freedom increase, we approach the Gaussian distribution, and so the values are likely to concentrate closer to the Gaussian limit. For small values for degree of freedom (e.g. 1-3) we have very heavy tails in the distribution, so that has an impact on the FSIP as these dots are far from the Gaussian case. Generally, increasing α and decreasing degrees of freedom simultaneously leads to a situation that indicates significant randomness or uncertainty.

6.4.3 Power Exponential Distributed Noise

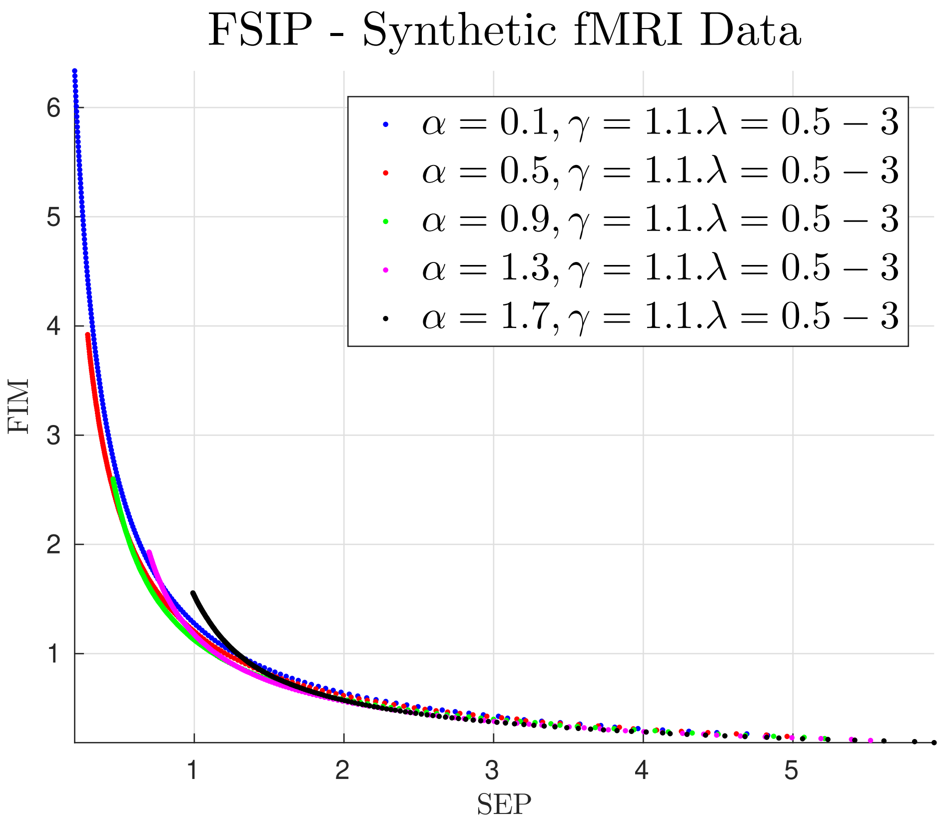


Figure 6.13: FSIP with Power Exponential Distributed Noise

In Fig. 6.13 we can see expected fMRI response, mixed with Power Exponential distributed noise with various values for the shape parameter. As we see in the plot, for small values of α we have a high FIM-low SEP values situation. As α increases, that leads to a low FIM-high SEP situation. Also, for small values of the shape parameter γ, we have a high FIM-low SEP situation. As γ increases, we are driven to low FIM-high SEP, which means significant randomness or uncertainty.

6.4.4 Gamma Distributed Noise

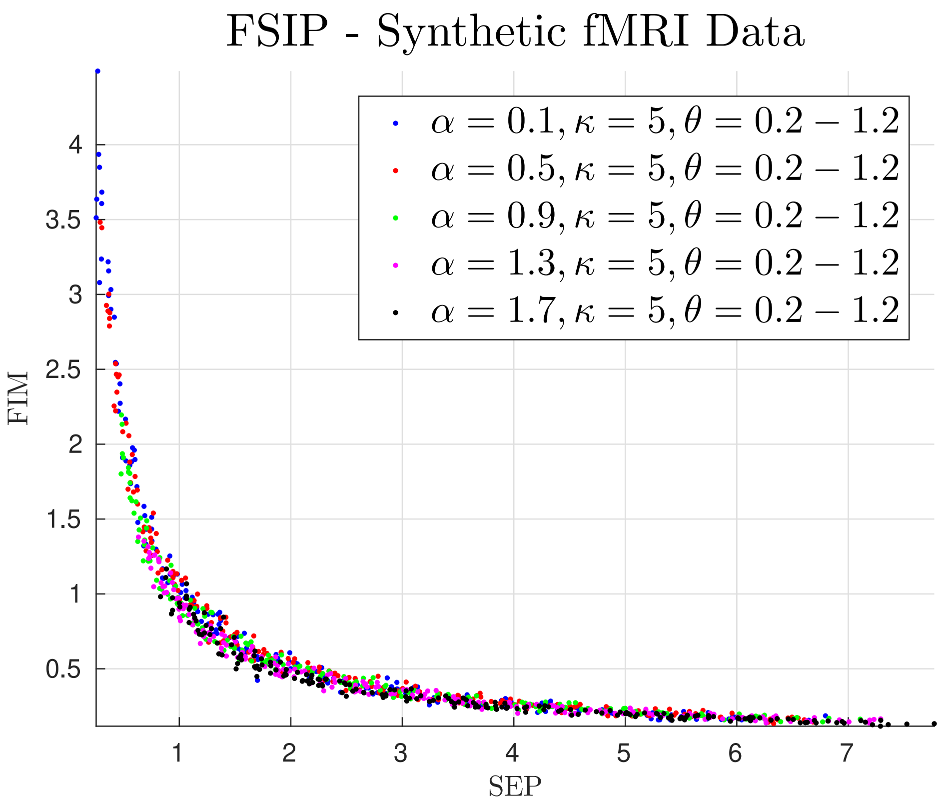


Figure 6.14: FSIP with Gamma Distributed Noise

In Fig. 6.13 we can see expected fMRI response, mixed with Gamma distributed noise with various values for the scale parameter. As we see in the plot, for small values of α we have a high FIM-low SEP values situation. As α increases, this leads to a low FIM-high SEP situation. Also, for small values of the scale parameter θ, we have a high FIM-low SEP situation. As θ increases, we are driven to low FIM-high SEP which means significant randomness or uncertainty.

6.4.5 Weibull Distributed Noise

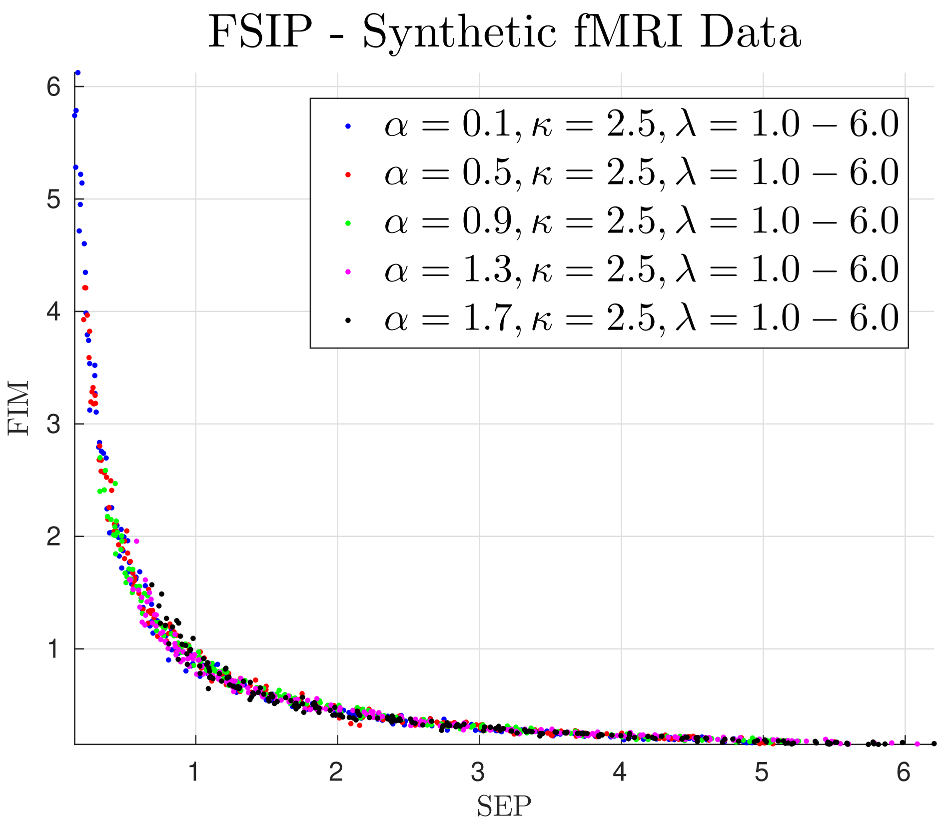


Figure 6.15: FSIP with Weibull Distributed Noise

In Fig. 6.15 we can see expected fMRI response, mixed with Gamma distributed noise with various values for the scale parameter λ. As we see in the plot, for small values of α we have a high FIM-low SEP values situation. As α increases, this leads to a low FIM-high SEP situation. Also, for small values of the scale parameter λ, we have a high FIM-low SEP situation. As λ increases, we are driven to low FIM-high SEP which means significant randomness or uncertainty.

6.4.6 Log-Normally Distributed Noise

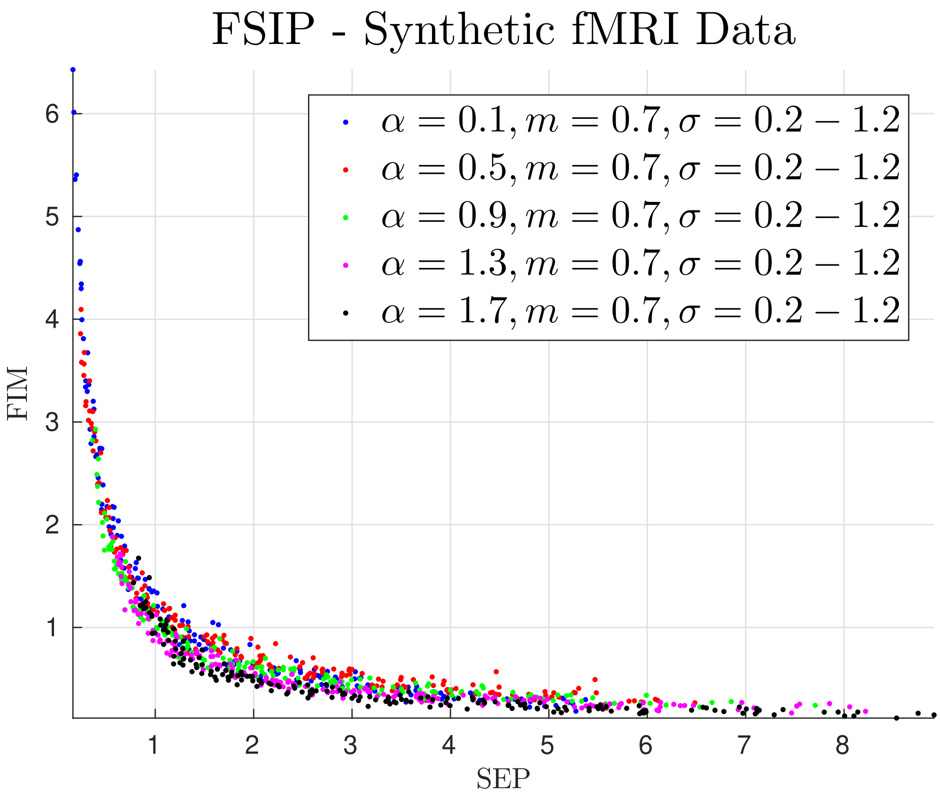


Figure 6.16: FSIP with Log-Normal Distributed Noise

In Fig. 6.16 we can see expected fMRI response, mixed with Gamma distributed noise with various values for parameter σ. As we see in the plot, for small values of α we have a high FIM-low SEP values situation. As α increases, this leads to a low FIM-high SEP situation. Also, for small values of the parameter σ, we have a high FIM-low SEP situation. As σ increases, we are driven to low FIM-high SEP which means significant randomness or uncertainty.

6.4.7 Uniformly Distributed Noise

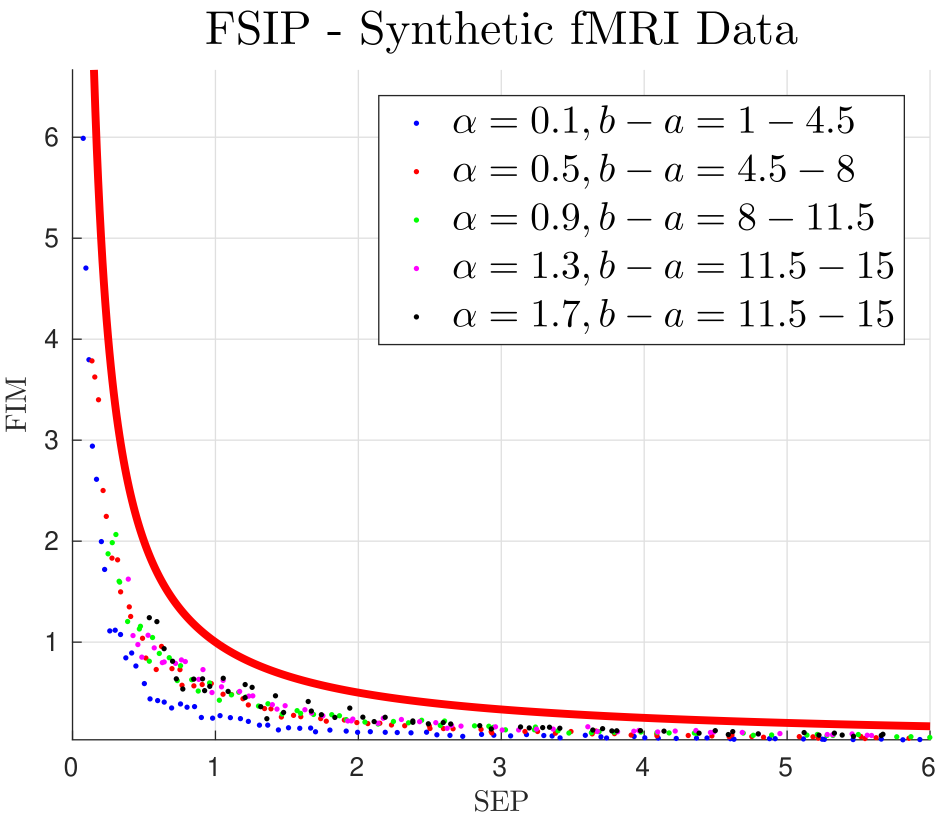


Figure 6.17: FSIP with Log-Normal Distributed Noise

In Fig. 6.17 we can see expected fMRI response, mixed with Gamma distributed noise with various values for the range $|\alpha - \beta|$. As we see in the plot, for small values of α we have a high FIM-low SEP values situation. As α increases, this leads to a low FIM-high SEP situation. Also, for small values of parameter $|\alpha - \beta|$, we have a high FIM-low SEP situation. As $|\alpha - \beta|$ increases, we are driven to low FIM-high SEP which means significant randomness or uncertainty. Furthermore, we see that for small $|\alpha - \beta|$ deltas, we have a bigger decrease in the FSIP regarding the FIM values and consequently a rapid increase on SEP values. The main characteristic for the distribution is that is located under the gaussian limit in the FSIP.

6.5 FSIP for Real FMRI Data

In this section, we utilized real fMRI data obtained through the procedure described in Sect. 2.2.4. For the preprocessed dataset, we applied the method outlined in Sect. 4.2.2 to compute KDE, followed by the calculation of SEP and FIM values. Figures Fig.6.18, Fig.6.20, Fig.6.22 and Fig.6.24 display the FSIP for the 25 subjects involved in the study. Each sub-figure contains 253,789 data points, representing combinations of SEP and FIM values. The axes of these sub-figures are limited to 5 on the y-axis and 1.5 on the x-axis to facilitate easier comparison between subjects, as the majority of FSIP values fall within this range. Figures Fig.6.19, Fig.6.21, Fig.6.23 and Fig.6.25 illustrate the FSIP values across all tasks included in the experiments, with each figure containing approximately 6.5 million SEP and FIM combination data points. In Figures Fig.6.26, Fig.6.27, Fig.6.28 and Fig.6.29, we attempted an average analysis by computing the average voxel value, resulting in figures with 253,789 data points.

6.5.1 FSIP per Subject

Fast to cup - Slow to person

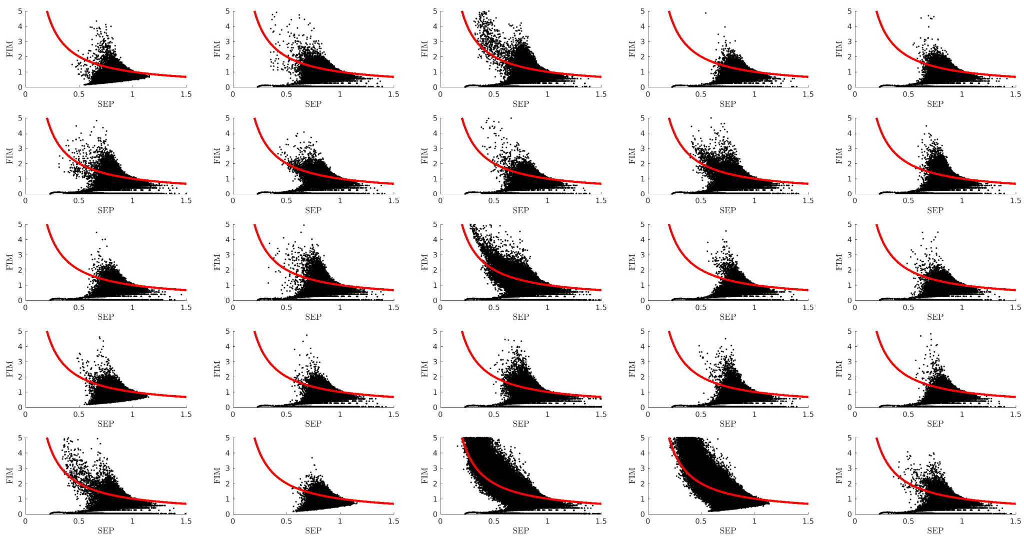


Figure 6.18: FSIP for Real FMRI Data-FS Experiment

In Fig. 6.18 we can see the FSIP for real fMRI data and also the gaussian limit on the red line. As we see we have an amount of FIM and SEP combination over the gaussian limit and also a significant amount, under the gaussian limit.

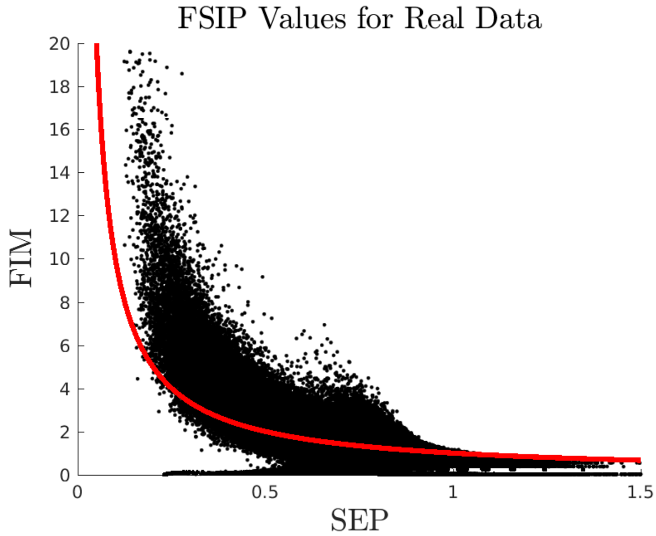


Figure 6.19: FSIP for 25 subjects-FS Experiment

In Fig. 6.19 we see the FSIP, for all voxels, for all subjects in a single plot. Every subject consist of 250.000 voxels approximately. For each voxel there is a time series to describe it and it is measured through the scanning session. There are 25 subjects, so the total number of points in that plot is 6.500.000. Furthermore, here we can see that a huge number of points, consisting of FIM and SEP values are over and under the Gaussian limit.

Slow to cup - Fast to person

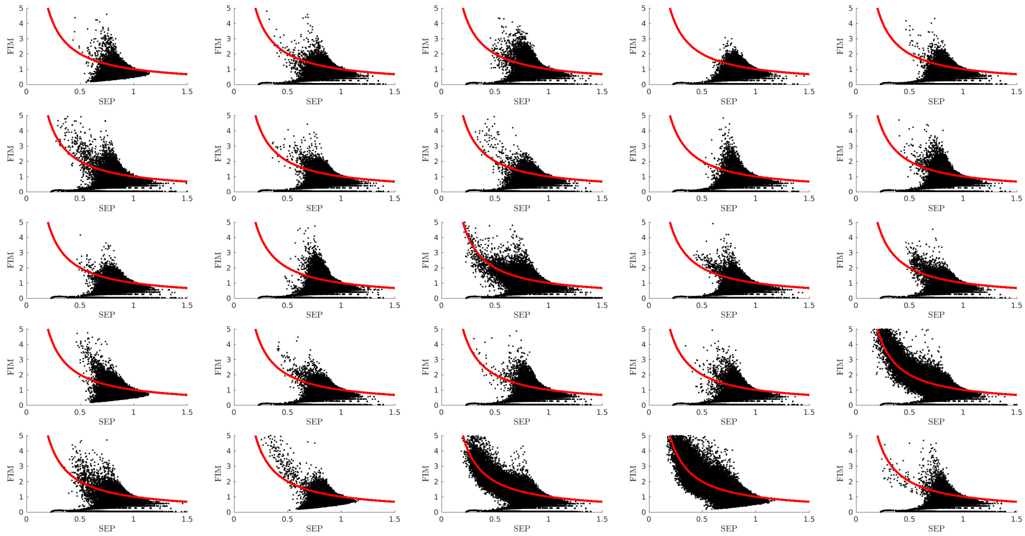


Figure 6.20: FSIP for Real FMRI Data-SF Experiment

In Fig. 6.20 we can see the FSIP for real fMRI data and also the Gaussian limit on the red line. As we can see, we have an amount of FIM and SEP combination over the Gaussian limit and also a significant amount, under the Gaussian limit.

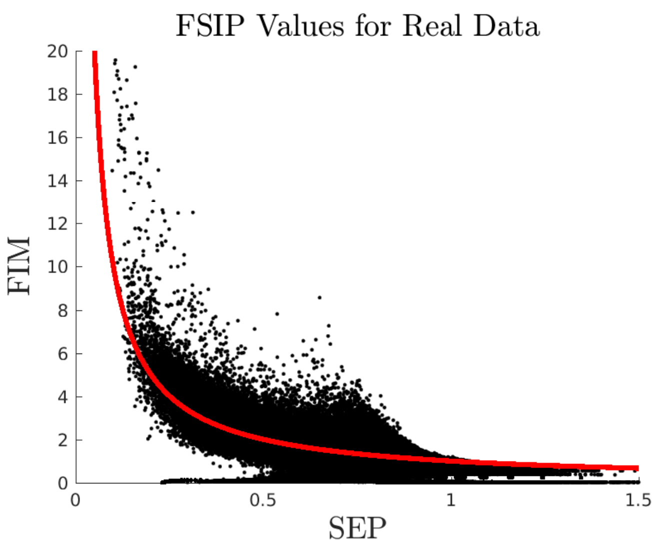


Figure 6.21: FSIP for 25 subjects-SF Experiment

In Fig. 6.21 we see the FSIP, for all voxels, for all subjects in a single plot. Each subject consists approximately of 250.000 voxels. For each voxel there is a time series to describe it and it is measured through the scanning session. There are 25 subjects, so the total number of points in that plot is 6.500.000. Furthermore, here we can see that a huge number of points, consisting of FIM and SEP values are over and under the Gaussian limit.

Fast to cup - Slow to bowl

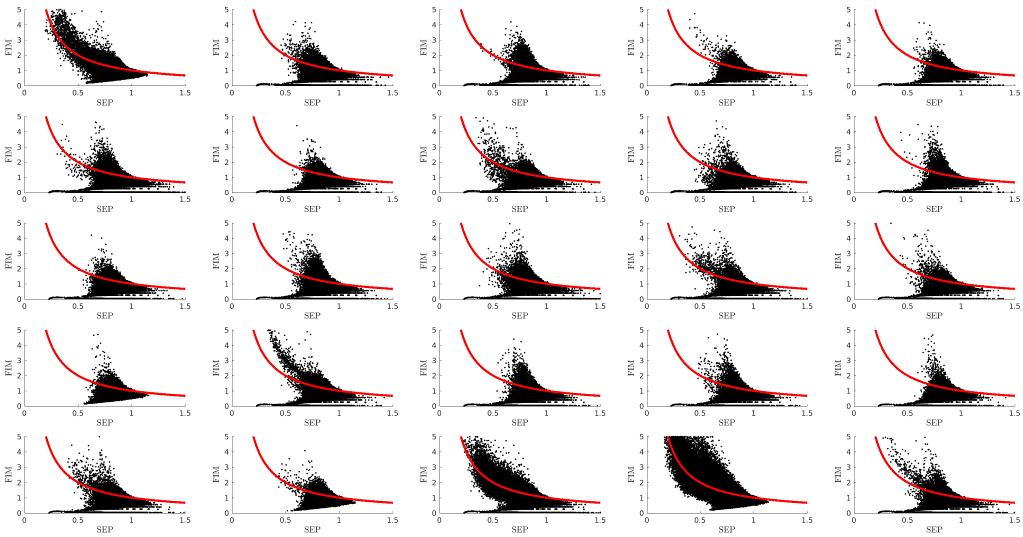


Figure 6.22: FSIP for Real FMRI Data-Bowl Experiment

In Fig. 6.22 we can see the FSIP for real fMRI data and also the Gaussian limit on the red line. As we see, we have an amount of FIM and SEP combination over the Gaussian limit and also a significant amount, under the Gaussian limit.

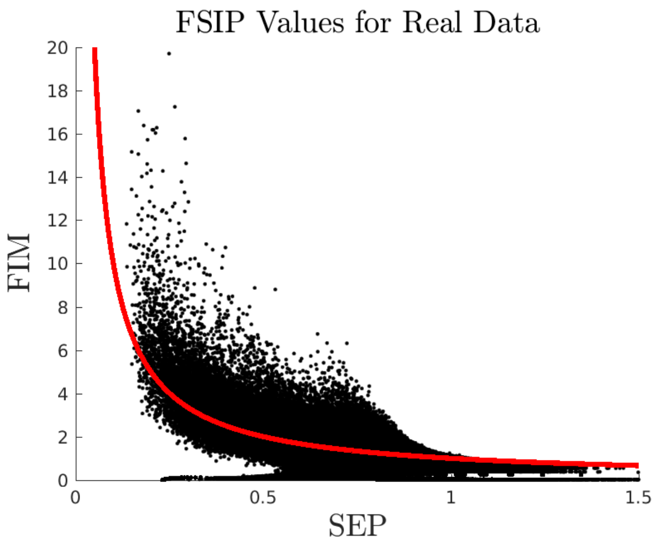


Figure 6.23: FSIP for 25 subjects-Bowl Experiment

In Fig. 6.23 we see the FSIP, for all voxels, for all subjects in a single plot. Each subject consists approximately of 250.000 voxels. For each voxel there is a time series to describe it, and it is measured through the scanning session. There are 25 subjects, so the total number of points in that plot is 6.500.000. Furthermore, here we can see that a huge number of points, consisting of FIM and SEP values are over and under the Gaussian limit.

Aimless action

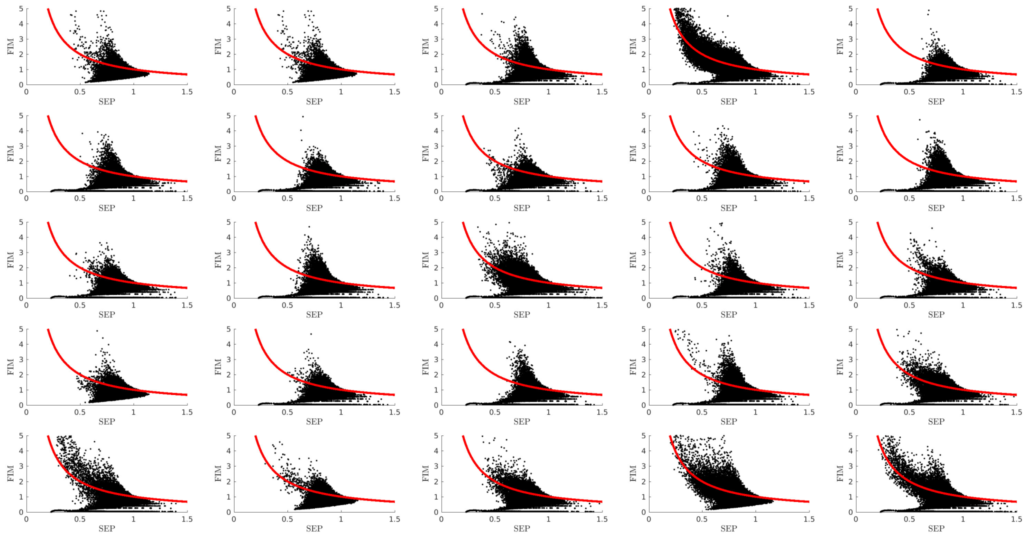


Figure 6.24: FSIP for Real FMRI Data-Aimless Experiment

In Fig. 6.24 we can see the FSIP for real fMRI data and also the Gaussian limit on the red line. As we see, we have an amount of FIM and SEP combination over the Gaussian limit and also a significant amount, under the gaussian limit.

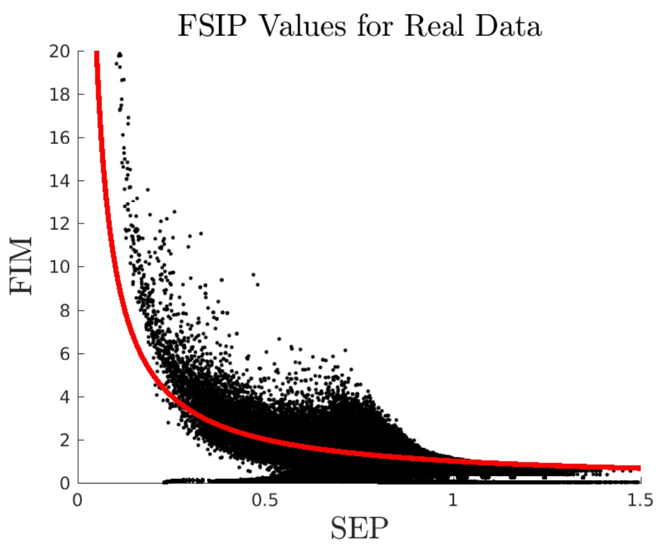


Figure 6.25: FSIP for 25 subjects-Aimless Experiment

In Fig. 6.25 we see the FSIP, for all voxels, for all subjects in a single plot. Each subject consists approximately of 250.000 voxels. For each voxel there is a time series to describe it and it is measured through the scanning session. There are 25 subjects, so the total number of points in that plot is 6.500.000. Furthermore, here we can see that a huge number of points, consisting of FIM and SEP values are over and under the Gaussian limit.

6.5.2 FSIP Analysis for Average of Subjects

In that section, we implement an average analysis regarding each voxel. Through the fMRI data acquirement it is unsurprising for different subjects to see smaller or larger differences in the shape of the human brain. With the task of Spatial Normalization it is believed that every voxel for each person can be aligned with the voxels of every other person. We believe that we can do a kind of average analysis and it is presented below along with the Gaussian limit.

Fast to cup - Slow to person

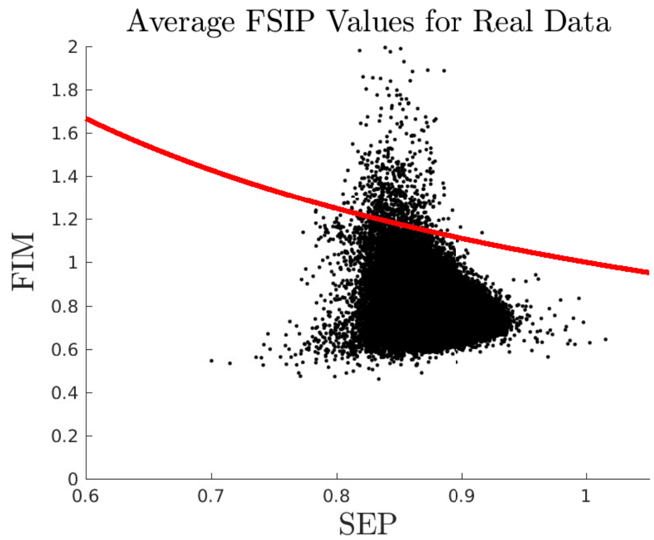


Figure 6.26: Average FSIP for Real FMRI Data-FS Experiment

Slow to cup - Fast to person

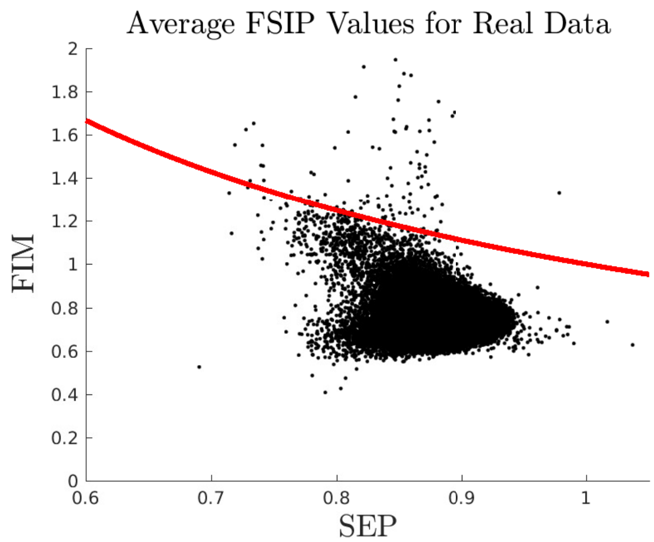


Figure 6.27: Average FSIP for Real FMRI Data-SF Experiment

Fast to cup - Slow to bowl

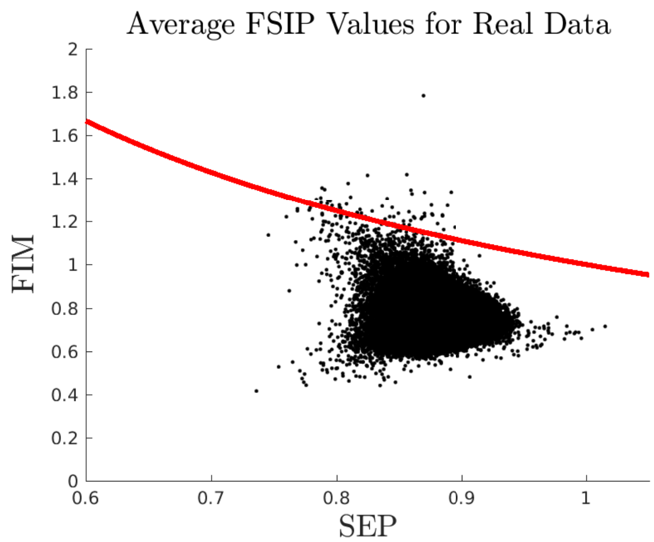


Figure 6.28: Average FSIP for Real FMRI Data-Bowl Experiment

Aimless action

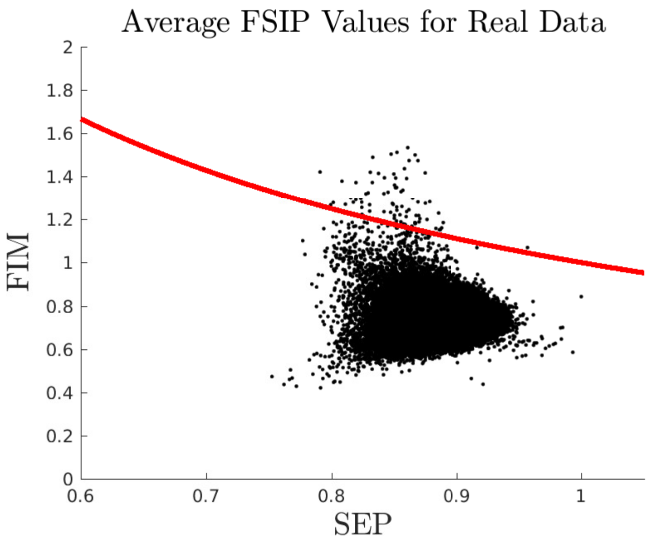


Figure 6.29: Average FSIP for Real FMRI Data-Aimless Experiment

6.5.3 FSIP Analysis for Median of Subjects

A similar analysis with Sec. 6.5.2 is done in this section but with median values. Median-based methods can be particularly useful in scenarios where we like less sensitivity to outliers, noise reduction, and we can observe voxel intensity distribution. Also, it is useful for group analysis, where the median can offer insights into central tendency, but also on temporal dynamics. With the task of Spatial Normalization, it is believed that every voxel for each person can be aligned with the voxels of every other person. We believe that we can do a kind of median analysis and it is presented below along with the Gaussian limit.

Fast to cup - Slow to person

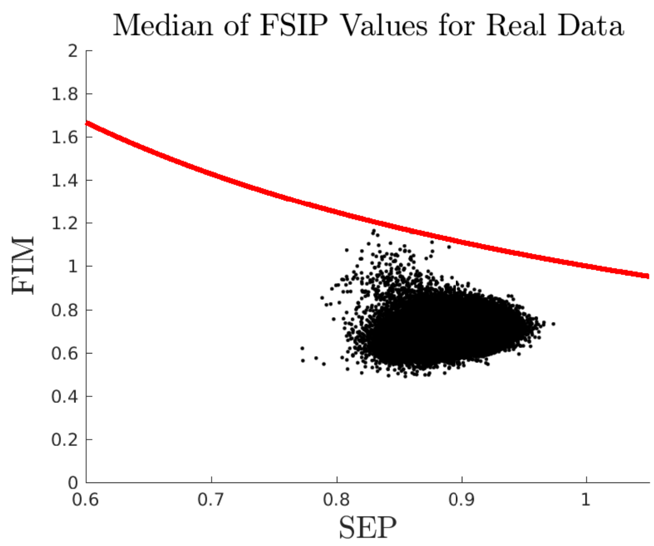


Figure 6.30: Median FSIP for Real FMRI Data-FS Experiment

Slow to cup - Fast to person

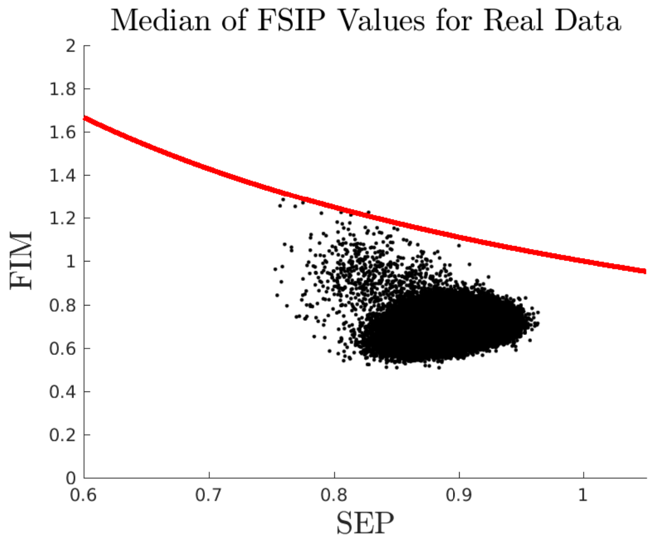


Figure 6.31: Median FSIP for Real FMRI Data-SF Experiment

Fast to cup - Slow to bowl

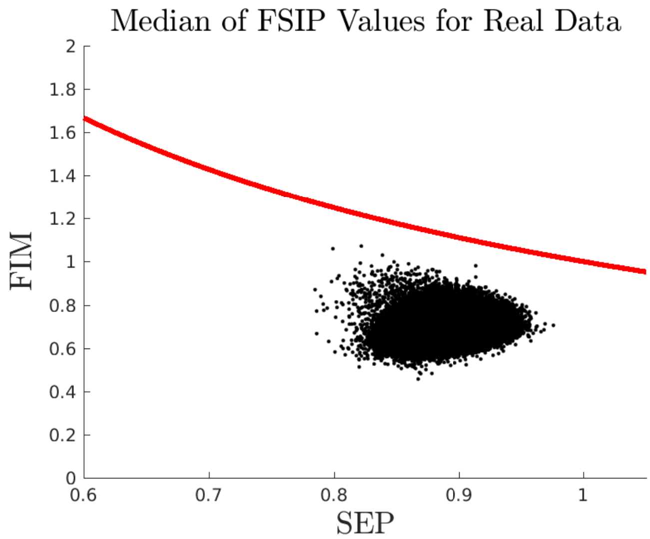


Figure 6.32: Median FSIP for Real FMRI Data-Bowl Experiment

Aimless action

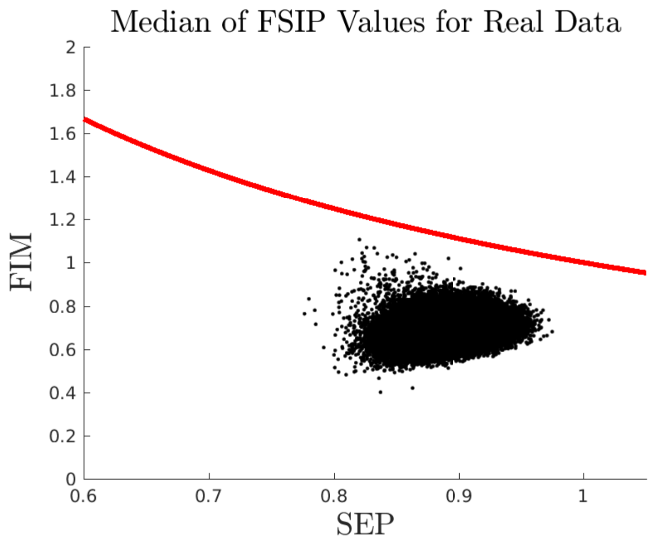


Figure 6.33: Median FSIP for Real FMRI Data-Aimless Experiment

6.6 Discussion

The following sections present the results of applying the Fisher-Shannon Information Plane (FSIP) framework to both synthetic and real fMRI data sets, building on the methodologies outlined in the preceding sections. Given the complex nature of fMRI data characterized by high-dimensional time-series signals and susceptibility to various artifacts, the need for advanced analytical techniques is paramount. The FSIP, which combines the FIM and SEP, offers insights through which to examine the complex dynamics of neural activity.

This chapter is structured to first address the application of FSIP to various distributions and synthetic data, where controlled conditions allow for the validation and calibration of the methodology against known probability distribution models. This is followed by an analysis of real fMRI data, where the robustness and versatility of FSIP are further tested against the natural complexities and variability present in human brain imaging.

The findings in this study are crucial for evaluating how well the FSIP can distinguish between different signal patterns, especially considering the challenges of noise and non-linearities in fMRI data. Moreover, these results highlight the potential benefits for the broader field of neuroimaging, such as improving data analysis and making neural activity interpretations more accurate.

6.6.1 FSIP Application to Random Numbers

In our investigation of random numbers, we examine how the FSIP responds to various parameter changes. Most distributions display behavior consistent with findings from existing studies and literature. A particularly intriguing case is the uniform distribution, which, given its characteristics, falls under the Gaussian limit. Interestingly, the uniform distribution shows a response similar to theoretical expectations and patterns observed in real fMRI data. However, when using the Kernel Density Estimator—the statistical tool we rely on to derive FSIP values—we encounter differentiation issues specific to the uniform distribution, which could affect the precision of these extracted values.

6.6.2 FSIP Application to Synthetic Data

In our investigation of the fMRI response with added noise, we examine how various distributions affect the FSIP. By scaling the expected response with different scalar multiples and introducing different types of noise, we observe the behavior of the FSIP across multiple distributions. For each distribution, we adjust the parameters to analyze how the FSIP responds to changes. In particular, we find correlations between the parameters and the FIM and SEP, revealing a shift from highly precise and deterministic distributions to more random and uncertain ones. In certain edge cases, we observe highly structured data that align under the Gaussian limit. However, the kernel density estimator, used to extract FSIP values, plays a critical role in these cases. The differential properties of certain distributions, along with discontinuities, often lead to vague outcomes, which can be challenging to interpret accurately.

6.6.3 FSIP Application to fMRI Data

The FSIP analysis of real fMRI data does not provide a clear representation of the underlying situation. Generally, the data shows that most subjects cluster within a similar region in the Fisher Shannon Information Plane, with the majority concentrated along the x-axis values between 0.5 and 1.2 and the y-axis values between 0.5 and 2. This concentration near the Gaussian limit suggests a notable lack of variability and may indicate the presence of systematic errors. Furthermore, many subjects exhibit significant artifacts and noise, which drastically alter the results and imply that some measurements may be unreliable. For several subjects, the FSIP plots appear anomalous, raising concerns about possible measurement errors and suggesting that some data points might need exclusion from further analysis.

The signals used in the FSIP analysis are also subject to compression, which could have a substantial impact on the final outcomes. Such compression might distort the data and affect the reliability of the FSIP results, leading to an excessive concentration of data points around (x, y) = (0.9, 0.8) in the average analysis. This clustering undermines the effectiveness of the average analysis, making it challenging to extract meaningful insights. Given these considerations, further validation and refinement of the pre-processing and analysis methods are essential for improving the accuracy and reliability of the findings.

Chapter 7

Conclusions

7.1 Conclusions

In this thesis, we have explored the application of the FSIP to various types of signals, examining how it reveals insights into both the structure and uncertainty of the data.

By applying the FSIP to multiple types of signals, we can gain a deeper understanding of their underlying structures and behaviors. FSIP provides a powerful framework for analyzing the relationship between information(via FIM) and uncertainty (via SEP). By investigating how different signals cluster within the FSIP, we can uncover distinctive patterns that may not be immediately apparent in traditional time or frequency domain analysis. This approach could indeed offer new ways to classify signals more accurately, as it allows us to evaluate both the precision of the signal and its inherent uncertainty, offering a more holistic view than simply looking at statistical properties like mean and variance.

When FSIP is applied to actual data, we often encounter more complexities than with synthetic models. Real-world data, such as fMRI data, tend to have noise, structural dependencies, and non-idealities that synthetic models may not account for. The behavior of the FSIP can vary significantly due to these factors in the real world, which presents challenges in the interpretation of the results. In contrast, synthetic models, designed with controlled parameters, often provide clearer and more predictable behavior in the FSIP. The parameters within probabilistic models, such as the type of distribution and noise characteristics, play a crucial role in shaping these outcomes. For example, the degree of randomness or structure in the data can shift how it is represented in the FSIP, with distributions ranging from highly deterministic to highly uncertain affecting the FSIP's response.

The Gaussian limit serves as a boundary or reference point in the FSIP, as Gaussian distributions exhibit maximum entropy for a given variance. This limit could influence signal classification by serving as a baseline for more structured or less structured distributions. Signals that align closely with the Gaussian limit may display behaviors that are more predictable and exhibit less uncertainty, making them easier to classify. On the other hand, signals deviating from the Gaussian limit—such as those with more extreme randomness or noise—may exhibit more complex or ambiguous FSIP representations, challenging classification. Understanding how signals behave with respect to this Gaussian boundary could help in distinguishing between different signal types or identifying anomalies in the data.

One of our most significant findings is the observation that, in cases where the signal dominates, the FSIP results tend to fall below the Gaussian limit. Conversely, when noise is the dominant factor, the FSIP values align with or exceed the Gaussian limit. This helps explain why, for real-world data, a substantial portion of the points lie below the limit, while there are also numerous values above it. This leads to the hypothesis that active regions are associated with values below the Gaussian limit.

Despite its compact nature and limited set of parameters, the FSIP has proven to be a versatile and powerful tool for characterizing a wide range of distributions and signal types, as we see in many related works. The combination of FIM and SEP allows the FSIP to capture both the precision and uncertainty of a signal, making it applicable to diverse scenarios. While its simplicity is an advantage, it also means that the FSIP may not fully capture every nuance in highly complex data, as fMRI data. However, for many common signal types, including those that are Gaussian-like or exhibit moderate noise, FSIP can provide valuable insights. Its ability to reveal underlying patterns through a relatively straightforward framework allows for broad applicability across different fields and types of data.

7.2 Future Work

While this thesis has demonstrated the potential of the Fisher-Shannon Information Plane (FSIP) for analyzing and classifying various signal types, several avenues for future work remain. In future work, a voxel-per-voxel analysis may be more useful for interpreting fMRI data in a more effective way. Differences in the same voxel between different subjects in the same experiment is the key for that. One key direction is the further exploration of FSIP's application to other complex, real-world data sets, such as EEG or those with higher noise levels or nonstationary behaviors, to better understand its limitations and refine its applicability. Furthermore, the development of advanced kernel density estimation techniques could help mitigate differentiation issues encountered in edge cases such as the uniform distribution, improving the robustness of FSIP in diverse scenarios. The current study demonstrates the potential of the FSIP in characterizing various distributions and signal types. However, there remain several opportunities to further extend and refine this framework, with higher-dimensional probabilities and other complementary measures. Finally, integrating FSIP with other machine learning and data analysis methods could open up new possibilities to improve signal classification accuracy and enable more precise predictions in fields such as neuroscience, communications and beyond.

Bibliography

[1] F. G. Ashby, "An introduction to fmri," in *An introduction to model-based cognitive neuroscience.* Springer, 2015, pp. 91–112.

[2] S. Ogawa, T. M. Lee, A. R. Kay, and D. W. Tank, "Brain magnetic resonance imaging with contrast dependent on blood oxygenation." *Proceedings of the National Academy of Sciences*, vol. 87, no. 24, pp. 9868–9872, 1990. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.87.24.9868

[3] S. Ogawa, D. W. Tank, R. Menon, J. M. Ellermann, S. G. Kim, H. Merkle, and K. Ugurbil, "Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging." *Proceedings of the National Academy of Sciences*, vol. 89, no. 13, pp. 5951–5955, 1992. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.89.13.5951

[4] K. K. e. a. Kwong, "Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation." *Proceedings of the National Academy of Sciences of the United States of America vol. 89,12 (1992): 5675-9.*, 1992. [Online]. Available: https://www.pnas.org/doi/pdf/10.1073/pnas.89.12.5675

[5] R. P. Friedland and C. Iadecola, "Roy and sherrington (1890)," *Neurology*, vol. 41, no. 1, pp. 10–10, 1991. [Online]. Available: https://www.neurology.org/doi/abs/10.1212/WNL.41.1.10

[6] N. Logothetis, J. Pauls, M. Augath, T. Trinath, and A. Oeltermann, "Neurophysiological investigation of the basis of the fmri signal," *Nature*, vol. 412, pp. 150–7, 08 2001.

[7] N. K. Logothetis, “The neural basis of the blood-oxygen-level-dependent functional magnetic resonance imaging signal.” *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, vol. 357 1424, pp. 1003–37, 2002. [Online]. Available: https://api.semanticscholar.org/CorpusID:14227983

[8] N. K. Logothetis and B. A. Wandell, “Interpreting the bold signal,” *Annu. Rev. Physiol.*, vol. 66, pp. 735–769, 2004.

[9] N. K. Logothetis, “What we can do and what we cannot do with fmri,” *Nature*, vol. 453, pp. 869–878, 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:4403097

[10] E. Formisano, D.-S. Kim, F. Di Salle, P.-F. van de Moortele, K. Ugurbil, and R. Goebel, “Mirror-symmetric tonotopic maps in human primary auditory cortex,” *Neuron*, vol. 40, no. 4, pp. 859–869, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S089662730300669X

[11] J. H. Lee, R. Durand, V. Gradinaru, F. Zhang, I. Goshen, D.-S. Kim, L. E. Fenno, C. Ramakrishnan, and K. Deisseroth, “Global and local fmri signals driven by neurons defined optogenetically by type and wiring,” *Nature*, vol. 465, no. 7299, pp. 788–792, 2010.

[12] J. B. Goense and N. K. Logothetis, “Neurophysiology of the bold fmri signal in awake monkeys,” *Current Biology*, vol. 18, no. 9, pp. 631–640, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960982208004429

[13] T. Thomsen, K. Specht, Å. Hammar, J. Nyttingnes, L. Ersland, and K. Hugdahl, “Brain localization of attentional control in different age groups by combining functional and structural mri,” *Neuroimage*, vol. 22, no. 2, pp. 912–919, 2004.

[14] G. Buzsáki, K. Kaila, and M. Raichle, “Inhibition and brain work,” *Neuron*, vol. 56, no. 5, pp. 771–783, 2007.

[15] B. Stefanovic, J. M. Warnking, and G. B. Pike, "Hemodynamic and metabolic responses to neuronal inhibition," *Neuroimage*, vol. 22, no. 2, pp. 771–778, 2004.

[16] H. S. Palmer, "Optogenetic fmri sheds light on the neural basis of the bold signal," *Journal of neurophysiology*, vol. 104, no. 4, pp. 1838–1840, 2010.

[17] J. M. Soares, R. Magalhães, P. S. Moreira, A. V. de Sousa, E. Ganz, A. Sampaio, V. Alves, P. Marques, and N. J. Sousa, "A hitchhiker's guide to functional magnetic resonance imaging," *Frontiers in Neuroscience*, vol. 10, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:12010779

[18] R. Adhikari and R. K. Agrawal, "An introductory study on time series modeling and forecasting," *arXiv preprint arXiv:1302.6613*, 2013.

[19] I. Bankman, *Handbook of medical image processing and analysis.* Elsevier, 2008.

[20] W. D. Penny, K. J. Friston, J. T. Ashburner, S. J. Kiebel, and T. E. Nichols, *Statistical parametric mapping: the analysis of functional brain images.* Elsevier, 2011.

[21] J. Ashburner and K. J. Friston, "Nonlinear spatial normalization using basis functions," *Human brain mapping*, vol. 7, no. 4, pp. 254–266, 1999.

[22] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal, "Automated multi-modality image registration based on information theory," in *Information processing in medical imaging*, vol. 3, no. 6. Citeseer, 1995, pp. 263–274.

[23] R. S. Frackowiak, *Human brain function.* Elsevier, 2004.

[24] B. de Haan and C. Rorden, "Introduction to functional mri," *Functional MRI*, 2018.

[25] G. M. Boynton, S. A. Engel, G. H. Glover, and D. J. Heeger, "Linear systems analysis of functional magnetic resonance imaging in human v1," *Journal of Neuroscience*, vol. 16, no. 13, pp. 4207–4221, 1996. [Online]. Available: https://www.jneurosci.org/content/16/13/4207

[26] N. Lange and S. Zeger, "Non-linear fourier analysis of magnetic resonance functional neuroimage time series," *Applied Statistics*, vol. 46, no. 1, p. 1 – 29, 1996, cited by: 4. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-52649134402&partnerID=40&md5=6c6626550ebc92130398502a8f228d17

[27] K. J. Friston, P. Jezzard, and R. Turner, "Analysis of functional mri time-series," *Human Brain Mapping*, vol. 1, no. 2, pp. 153–171, 1994. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/hbm.460010207

[28] R. S. et al., "Exploring the temporal boundaries of fmri: Measuring responses to very brief visual stimuli," *Book of Abstracts, Society for Neuroscience 24th Annual Meeting, Miami*, vol. 1264, 1994.

[29] M. S. Cohen, "Parametric analysis of fmri data using linear systems methods," *NeuroImage*, vol. 6, no. 2, pp. 93–103, 1997. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811997902780

[30] J. C. Rajapakse, F. Kruggel, J. M. Maisog, and D. Yves von Cramon, "Modeling hemodynamic response for analysis of functional mri time-series," *Human brain mapping*, vol. 6, no. 4, pp. 283–300, 1998.

[31] K. J. Worsley, C. H. Liao, J. Aston, V. Petre, G. Duncan, F. Morales, and A. C. Evans, "A general statistical analysis for fmri data," *Neuroimage*, vol. 15, no. 1, pp. 1–15, 2002.

[32] K. Friston, P. Fletcher, O. Josephs, A. Holmes, M. Rugg, and R. Turner, "Event-related fmri: Characterizing differential responses," *NeuroImage*, vol. 7, no. 1, pp. 30–40, 1998. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811997903062

[33] G. H. Glover, "Deconvolution of impulse response in event-related bold fmri1," *NeuroImage*, vol. 9, no. 4, pp. 416–429, 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811998904190

[34] M. Lindquist, "The statistical analysis of fmri data," *Stat. Sci.*, vol. 23, 06 2009.

[35] C. Vignat and J.-F. Bercher, "Analysis of signals in the fisher–shannon information plane," *Physics Letters A*, vol. 312, no. 1, pp. 27–33, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S037596010300570X

[36] F. Guignard, M. Laib, F. Amato, and M. Kanevski, "Advanced analysis of temporal data using fisher-shannon information: Theoretical development and application in geosciences," *Frontiers in Earth Science*, vol. 8, 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/feart.2020.00255

[37] F. Guignard, D. Mauree, M. Lovallo, M. Kanevski, and L. Telesca, "Fisher–shannon complexity analysis of high-frequency urban wind speed time series," *Entropy*, vol. 21, no. 1, 2019. [Online]. Available: https://www.mdpi.com/1099-4300/21/1/47

[38] L. Telesca and M. Lovallo, "On the performance of fisher information measure and shannon entropy estimators," *Physica A: Statistical Mechanics and its Applications*, vol. 484, pp. 569–576, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378437117303783

[39] A. Dembo, T. M. Cover, and J. A. Thomas, "Information theoretic inequalities," *IEEE Transactions on Information theory*, vol. 37, no. 6, pp. 1501–1518, 1991.

[40] M. Thomas and A. T. Joy, *Elements of information theory.* Wiley-Interscience, 2006.

Appendix A

Appendix Code

A.1 Theoretical FSIP Computation

A.1.1 Gamma Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%% GAMMA  %%%%%%%%%%%%%%%%%%%%%%%%%
theta =linspace(0.23,1.2,1000);
k=5   %linspace(2.1,7,1000);
Nx = ( (theta.^2 * gamma(k)^2) ./ (2*pi*exp(1)) ) * exp(2 *
    ((1-k) * psi(k) +k));
Ix = 1./((k-2)*theta.^2);

plot(Nx, Ix, 'linewidth',3)
grid on;
axis tight;
tt=title('FSIP of Gamma Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');

 print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_gamma_th
    '

Cx = Nx.*Ix;
figure();
plot(theta, Cx, 'linewidth',1)
tt=title('Complexity of Gamma Distribution', 'fontsize',18)
```

```
xx=xlabel('$\theta$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;


 print -deps '/home/ioannis/Desktop/FSIP Theory/
    Compl_gamma_th'
```

For Gamma Distribution the computation of FS Complexity done with two different ways [36], [35].

A.1.2 Weibull Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%% WEIBULL %%%%%%%%%%%%%%%%%%%%%%%%

% Gamma Euler-Mascheroni constant
gama= 0.5772156649;
lambda=linspace(1.32,7.7,1000);
k = 2.5 %linspace(2.1,6,1000);
a=(k-1)./k;

Nx = (((1-a)^2 * lambda.^2 * exp(1)) / (2*pi) ) * exp(2 * a *
    gama);
Ix = ((a^2) ./ ( (1-a)^2 * lambda.^2 )) * gamma(2*a-1);

plot(Nx, Ix, 'linewidth',2)
grid on;
axis tight;
tt=title('FSIP of Weibull Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
```

```

print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_Wei_th'

Cx = Nx.*Ix;
figure();
plot(lambda, Cx)
tt=title('Complexity of Weibull Distribution', 'fontsize',18)
xx=xlabel('$\lambda$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;

print -deps '/home/ioannis/Desktop/FSIP Theory/Compl_Wei_th'

kk=linspace(2.2,16,1000);
aa=(kk-1)./kk
cxx_paper= (((aa.^2)*exp(1))/(2*pi)) .* gamma(2*aa-1).*exp(2*
    aa*gama)
figure();
plot(kk, cxx_paper)
tt=title('Complexity of Weibull Distribution', 'fontsize',18)
xx=xlabel('$\kappa$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;

print -deps '/home/ioannis/Desktop/FSIP Theory/Compl2_Wei_th'
```

For Weibull Distribution the computation of FS Complexity done with two different ways [36], [35].

A.1.3 Log-Normal Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%% LOG-NORMAL %%%%%%%%%%%%%%%%%%%%%
mi = .7;
sigma = linspace(0.2,1.2,1000);

Nx = sigma.^2 * exp(2*mi);
Ix = (1 + 1./sigma.^2) .* exp(2.*(sigma.^2-mi));

plot(Nx, Ix, 'linewidth',2)
grid on;
axis tight;
tt=title('FSIP of Log-Normal Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');

print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_LN_th'

Cx = Nx.*Ix;
figure();
plot(sigma, Cx)
tt=title('Complexity of Log-Normal Distribution', 'fontsize'
    ,18)
xx=xlabel('$\sigma$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;

print -deps '/home/ioannis/Desktop/FSIP Theory/Compl_LN_th'
```

For Log-Normal Distribution the computation of FS Complexity done with two different ways [36], [35].

A.1.4 Student-t Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%% STUDENT-T  %%%%%%%%%%%%%%%%%%%%%

% q > 0 extensivity parameter
% m degrees of freedom
% m = (1+q)/(1-q)
m=3;
sigma = linspace(0.575,3,1000); %variance of Normal Distribution from X=N/Xm

Nx = (1 / (2*pi*exp(1)) ) * ((( sigma * sqrt(m - 2) * gamma(m/2) * gamma(1/2) )/ gamma((m+1)/2) ).^2) * exp((m+1) * (psi((m+1)/2)-psi(m/2)));
Ix = (1 ./ sigma.^2) * (m *(m+1) / ( (m-2) * (m+3) ) );

figure();
plot(Nx, Ix, 'linewidth',2)
grid on;
axis tight;
tt=title('FSIP of Student-t Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');

print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_stud_th'

Cx = Nx.*Ix;
figure();
plot(sigma, Cx)
tt=title('Complexity of Student-t Distribution', 'fontsize',18)
xx=xlabel('$\sigma$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;
```

```
print -deps '/home/ioannis/Desktop/FSIP Theory/Compl_stud_th'
```

A.1.5 Power Exponential Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%% POWER EXPONENTIAL %%%%%%%%%%%%%

%lambda scale
%gama shape
gama =3;
lambda = linspace(0.015,2.5,1000);

Nx = (2 / (pi * exp(1)) ) *  ( (1 / gama) * gamma(1 / gama) * ( exp(1) ./ lambda ).^(1 / gama) ).^2;
Ix = (gamma(1-(1/gama)) / gamma(1/gama)) * gama * (gama - 1) * (lambda.^ (2/gama));

plot(Nx, Ix, 'linewidth',2)
grid on;
axis tight;
tt=title('FSIP of Power Exponential Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');

print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_pe_th'

% g=2.9 %linspace(1.001,10,1000);
% hpe = (2*exp((2./g)-1)/pi).*gamma(1./g).*(gamma(2-(1./g)));
% figure();
% plot(g, hpe)
% axis tight;
```

```

Cx = Nx.*Ix;
figure();
plot(lambda, Cx)
tt=title('Complexity of Power Exponential Distr.', 'fontsize'
    ,18)
xx=xlabel('$\lambda$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;

print -deps '/home/ioannis/Desktop/FSIP Theory/Compl_pe_th'
```

A.1.6 Normal Distribution

Computation of I_x, N_x and FS Complexity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%% GAUSSIAN %%%%%%%%%%%%%

%mu     mean
%sigma std
sigma = linspace(0.4082,2.45,1000);

Nx = sigma.^2
Ix = 1./sigma.^2

plot(Nx, Ix, 'linewidth',2)
grid on;
axis tight;
tt=title('FSIP of Gaussian Distribution', 'fontsize',18)
xx=xlabel('SEP', 'fontsize',18);
yy=ylabel('FIM', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');

```

```
print -deps '/home/ioannis/Desktop/FSIP Theory/FSIP_norm_th'

Cx = Nx.*Ix;
figure();
plot(sigma, Cx)
tt=title('Complexity of Gaussian Distribution', 'fontsize'
    ,18)
xx=xlabel('$\sigma$', 'fontsize',18);
yy=ylabel('Complexity', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
axis tight;

print -deps '/home/ioannis/Desktop/FSIP Theory/Compl_norm_th'
```

A.2 Probability Distributions Analysis

A.2.1 Gamma Distribution

Computations

```
%%%%%%%%% GAMMA %%%%%%%%%%
x = 0:0.01:20;
a = [0.5 1 2 5]          %shape
b =  1                   %scale

for i=1:4
    pdf_gamma(i,:) = pdf('Gamma',x,a(i),b);
    cdf_gamma(i,:) = cdf('Gamma',x,a(i),b);
    icdf_gamma(i,:) = icdf('Gamma',x,a(i),b);
    hazard_gamma(i,:) = pdf_gamma(i,:)./(1-cdf_gamma(i,:));
    c_hazard_gamma(i,:) = -(log10(1-cdf_gamma(i,:))/log10(exp
       (1)));
    survival_gamma(i,:) = 1- cdf_gamma(i,:);
    xd = 1-x;
    i_survival_gamma(i,:) = icdf('Gamma',x,a(i),b);

```

```
end

figure();
plot(x, pdf_gamma, 'linewidth',2)
grid on;
axis tight;
ylim([0 1]);
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('PDF of Gamma Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/pdf_gamma'

figure();
plot(x, cdf_gamma, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('CDF of Gamma Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/cdf_gamma'

figure();
plot(x, icdf_gamma, 'linewidth',2)
grid on;
axis tight;
```

```
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('PPF of Gamma Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/icdf_gamma'

figure();
plot(x, hazard_gamma, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('Hazard function of Gamma Distribution')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/hazard_gamma'

figure();
plot(x, c_hazard_gamma, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('Cumulative HF of Gamma Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
```

```
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/c_hazard_gamma'

figure();
plot(x, survival_gamma, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('Survival function of Gamma Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/survival_gamma'

figure();
plot(xd, i_survival_gamma, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\kappa=0.5$','$\kappa=1$','$\kappa=2$','$\kappa=5
    $')
tt=title('Inverse Survival function of Gamma Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    GAMMA/i_survival_gamma'
```

A.2.2 Weibull Distribution

Computations

```
%%%%%%%% WEIBULL %%%%%%%%
x = 0:0.01:5;
a = 1          %scale param
b = [.5 1 2 5];        %shape param

for i =1:4
    pdf_Weibull(i,:) = pdf('Weibull',x,a,b(i));
    cdf_Weibull(i,:) = cdf('Weibull',x,a,b(i));
    icdf_Weibull(i,:) = icdf('Weibull',x,a,b(i));
    hazard_Weibull(i,:) = pdf_Weibull(i,:)./(1-cdf_Weibull(i
        ,:));
    c_hazard_Weibull(i,:) = -(log10(1-cdf_Weibull(i,:))/log10
        (exp(1)));
    survival_Weibull(i,:) = 1- cdf_Weibull(i,:);
    xd = 1-x;
    i_survival_Weibull(i,:) = icdf('Weibull',x,a,b(i));
end

figure();
plot(x, pdf_Weibull, 'linewidth',2)
grid on;
axis tight;
ylim([0 2]);
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('PDF of Weibull Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/pdf_weibull'

figure();
plot(x, cdf_Weibull, 'linewidth',2)
```

```
grid on;
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('CDF of Weibull Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/cdf_weibull'

figure();
plot(x, icdf_Weibull, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('PPF of Weibull Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/icdf_weibull'

figure();
plot(x, hazard_Weibull(1:3,:), 'linewidth',2)
hold on;
plot(x(1:120), hazard_Weibull(4,1:120), 'linewidth', 2)
grid on;
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('Hazard function of Weibull Distribution')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
```

```
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/hazard_weibull'

figure();
plot(x, c_hazard_Weibull, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('Cumulative HF of Weibull Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/c_hazard_weibull'

figure();
plot(x, survival_Weibull, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('Survival function of Weibull Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/survival_weibull'

figure();
plot(xd, i_survival_Weibull, 'linewidth',2)
grid on;
```

```
axis tight;
ll=legend('$\beta=0.5$','$\beta=1$','$\beta=2$','$\beta=5$')
tt=title('Inverse Survival function of Weibull Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    WEIBULL/i_survival_weibull'
```

A.2.3 Log-Normal Distribution

Computations

```
%%%%%%%%%% LOG-NORMAL %%%%%%%%%%%
x = 0:0.01:2;
xx=0:0.01:1;
mu = 0      %mean of logarithmic values
sigma = [0.5 1 2 5];   %standard deviation of logarithmic
    values

for i =1:4
    pdf_LN(i,:) = pdf('Lognormal',x,mu,sigma(i));
    cdf_LN(i,:) = cdf('Lognormal',x,mu,sigma(i));
    icdf_LN(i,:) = icdf('Lognormal',x,mu,sigma(i));
    hazard_LN(i,:) = pdf_LN(i,:)./(1-cdf_LN(i,:));
    c_hazard_LN(i,:) = -(log10(1-cdf_LN(i,:))/log10(exp(1)));
    survival_LN(i,:) = 1- cdf_LN(i,:);
    xd = 1-x;
    i_survival_LN(i,:) = icdf('Lognormal',x,mu,sigma(i));
end


figure();
plot(x, pdf_LN, 'linewidth',2)
grid on;
axis tight;
```

```
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
   $')
tt=title('PDF of Lognormal Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
   LOGNORMAL/pdf_LN'

figure();
plot(x, cdf_LN, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
   $')
tt=title('CDF of Lognormal Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
   LOGNORMAL/cdf_LN'

figure();
plot(x, icdf_LN, 'linewidth',2)
grid on;
axis tight;
ylim([0 5])
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
   $')
title('PPF of Lognormal Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
```

```
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/LOGNORMAL/icdf_LN'

figure();
plot(x, hazard_LN, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5$')
tt=title('Hazard function of Lognormal Distribution')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/LOGNORMAL/hazard_LN'

figure();
plot(x, c_hazard_LN, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5$')
tt=title('Cumulative HF of Lognormal Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/LOGNORMAL/c_hazard_LN'

figure();
```

```
plot(x, survival_LN, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Survival function of Lognormal Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    LOGNORMAL/survival_LN'

figure();
plot(xd, i_survival_LN, 'linewidth',2)
grid on;
axis tight;
ylim([0 5])
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Inverse Survival function of Lognormal Distribution
    ')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    LOGNORMAL/i_survival_LN'
```

A.2.4 Student-t Distribution

Computations

```
%%%%%%%%%% STUDENT-T %%%%%%%%%%
x = -6:0.01:6;
```

```
df = [1 5 10 30];      %degrees of freedom

for i=1:4
    pdf_t(i,:) = pdf('T',x,df(i));
    cdf_t(i,:) = cdf('T',x,df(i));
    icdf_t(i,:) = icdf('T',x,df(i));
    hazard_t(i,:) = pdf_t(i,:)./(1-cdf_t(i,:));
    c_hazard_t(i,:) = -(log10(1-cdf_t(i,:))/log10(exp(1)));
    survival_t(i,:) = 1- cdf_t(i,:);
    xd = 1-x;
    i_survival_t(i,:) = icdf('T',x,df(i));
end

figure();
plot(x, pdf_t, 'linewidth',2)
grid on;
axis tight;
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('PDF of Student-t Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/pdf_t'

figure();
plot(x, cdf_t, 'linewidth',2)
grid on;
axis tight;
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('CDF of Student-t Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
```

```
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/cdf_t'

figure();
plot(x, icdf_t, 'linewidth',2)
grid on;
axis tight;
ylim([-6 6]);
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('PPF of Student-t Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/icdf_t'

figure();
plot(x, hazard_t, 'linewidth',2)
grid on;
axis tight;
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
title('Hazard function of Student-t Distribution')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/hazard_t'

figure();
plot(x, c_hazard_t, 'linewidth',2)
grid on;
axis tight;
```

```
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('Cumulative HF of Student-t Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/c_hazard_t'

figure();
plot(x, survival_t, 'linewidth',2)
grid on;
axis tight;
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('Survival function of Student-t Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/survival_t'

figure();
plot(xd, i_survival_t, 'linewidth',2)
grid on;
axis tight;
ylim([-6 6]);
ll=legend('$df=1$','$df=5$','$df=10$','$df=30$')
tt=title('Inverse Survival function of Student-t Distribution
    ')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
```

```
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    STUDENTT/i_survival_t'
```

A.2.5 Power Exponential Distribution

Computations

```
%%%%%%%%% POWER EXPONENTIAL %%%%%%%%%
x = -3:0.01:3;
a = 1                       %scale
b = [.5 1 2 5 10]           %shape

for i=1:5
    pdf_pe(i,:) = b(i)/(2*a*gamma(1/b(i)))*exp(-(abs(x)/a).^b
        (i));
    pdf_pe_f = @(x) b(i)/(2*a*gamma(1/b(i)))*exp(-(abs(x)/a)
        .^b(i));
    cdf_pe(i,:) = zeros(size(x));

    for j=1:length(x)
        cdf_pe(i,j) = integral(pdf_pe_f,-Inf,x(j));
    end
    v=(2*abs(x-0.5));
    icdf_pe(i,:) = sign(x-0.5).* (a^b(i) * (icdf('Gamma',v
        ,1,1/b(i)))).^(1/b(i));
    hazard_pe(i,:) = pdf_pe(i,:)./(1-cdf_pe(i,:));
    c_hazard_pe(i,:) = -(log10(1-cdf_pe(i,:))/log10(exp(1)));
    survival_pe(i,:) = 1- cdf_pe(i,:);
    xd = 1-x;
    vd=(2*abs(xd-0.5));
    i_survival_pe(i,:) = sign(xd-0.5).* (a^b(i) * (icdf('
        Gamma',vd,1,1/b(i)))).^(1/b(i));
end

figure();
plot(x, pdf_pe, 'linewidth',2)
grid on;
axis tight;
```

```
tt=title('PDF of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5
    $','$\gamma=10$')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/
    pdf_pe'

figure();
plot(x, cdf_pe, 'linewidth',2)
grid on;
axis tight;
tt=title('CDF of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5
    $','$\gamma=10$')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/
    cdf_pe'

figure();
plot(x, icdf_pe, 'linewidth',2)
grid on;
axis tight;
tt=title('PPF of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5
    $','$\gamma=10$')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
```

```
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/icdf_pe'

figure();
plot(x(1:380), hazard_pe(:,1:380), 'linewidth',2)
grid on;
axis tight;
tt=title('Hazard function of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5$','$\gamma=10$')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/hazard_pe'

figure();
plot(x(1:380), c_hazard_pe(:,1:380), 'linewidth',2)
grid on;
axis tight;
tt=title('Cumulative HF of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5$','$\gamma=10$')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/c_hazard_pe'

figure();
plot(x, survival_pe, 'linewidth',2)
```

```
grid on;
axis tight;
tt=title('Survival function of Power Exponential Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5$','$\gamma=10$')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex', 'fontsize',18);
set(yy,'interpreter', 'latex', 'fontsize',18);
set(tt,'interpreter', 'latex', 'fontsize',18);
set(ll,'interpreter', 'latex', 'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/survival_pe'

figure();
plot(x, i_survival_pe, 'linewidth',2)
grid on;
axis tight;
tt=title('Inverse Survival Function of PE Distribution')
ll=legend('$\gamma=0.5$','$\gamma=1$','$\gamma=2$','$\gamma=5$','$\gamma=10$')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex', 'fontsize',18);
set(yy,'interpreter', 'latex', 'fontsize',18);
set(tt,'interpreter', 'latex', 'fontsize',18);
set(ll,'interpreter', 'latex', 'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/PPE/i_survival_pe'
```

A.2.6 Normal Distribution

Computations

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% NORMAL %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mu = 0          % mean
sigma = [0.5 1 2 5];        % std
```

```
x = [-3:.01:3];

for i =1:4
    pdf_normal(i,:) = pdf('Normal',x,mu,sigma(i));

    cdf_normal(i,:) = cdf('Normal',x,mu,sigma(i));

    icdf_normal(i,:) = icdf('Normal',x,mu,sigma(i));

    hazard_normal(i,:) = pdf_normal(i,:)./(1-cdf_normal(i,:));

    c_hazard_normal(i,:) = -(log10(1-cdf_normal(i,:))/log10(exp(1)));

    survival_normal(i,:) = 1- cdf_normal(i,:);

    xd = 1-x;
    i_survival_normal(i,:) = icdf('Normal',x,mu,sigma(i));
end


figure();
plot(x, pdf_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5$')
tt=title('PDF of Normal Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex');
print -depsc '/home/ioannis/Desktop/Distributions Theory/NORMAL/pdf_normal'

figure();
```

```
plot(x, cdf_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('CDF of Normal Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex', 'fontsize',18);
set(yy,'interpreter', 'latex', 'fontsize',18);
set(tt,'interpreter', 'latex', 'fontsize',18);
set(ll,'interpreter', 'latex', 'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/cdf_normal'

figure();
plot(x, icdf_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('PPF of Normal Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex', 'fontsize',18);
set(yy,'interpreter', 'latex', 'fontsize',18);
set(tt,'interpreter', 'latex', 'fontsize',18);
set(ll,'interpreter', 'latex', 'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/icdf_normal'

figure();
plot(x, hazard_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Hazard function of Normal Distribution')
xx=xlabel('Failure time');
```

```
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/hazard_normal'

figure();
plot(x, c_hazard_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Cumulative HF of Normal Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/c_hazard_normal'

figure();
plot(x, survival_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Survival function of Normal Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/survival_normal'
```

```

figure();
plot(xd, i_survival_normal, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\sigma=0.5$','$\sigma=1$','$\sigma=2$','$\sigma=5
    $')
tt=title('Inverse Survival function of Normal Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/i_survival_normal'
```

A.2.7 Uniform Distribution

Computations

```
%%%%%%%%%%%%%%%%%%%%%%%%%%% Uniform
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
a=0;
b=[1,2,5,10];

x = [-1:.01:11];

for i =1:4
    pdf_uniform(i,:) = unifpdf(x, a, b(i));

    cdf_uniform(i,:) = unifcdf(x, a, b(i));

x1 = [0:.01:1];
    icdf_uniform(i,:) = unifinv(x1, a, b(i));

    hazard_uniform(i,:) = pdf_uniform(i,:)./(1-cdf_uniform(i
        ,:));

```

```
    c_hazard_uniform(i,:) = -(log10(1-cdf_uniform(i,:))/log10(exp(1)));

    survival_uniform(i,:) = 1- cdf_uniform(i,:);

    q = 1-x1;
    i_survival_uniform(i,:) =a + (1 - q) .* (b(i) - a);
end


figure();
plot(x, pdf_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$','$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('PDF of Uniform Distribution')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex');
print -depsc '/home/ioannis/Desktop/Distributions Theory/Uniform/pdf_uniform'

figure();
plot(x, cdf_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$','$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('CDF of Uniform Distribution')
xx=xlabel('x');
yy=ylabel('CDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
```

```
print -depsc '/home/ioannis/Desktop/Distributions Theory/Uniform/cdf_uniform'

figure();
plot(x1, icdf_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$','$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('PPF of Uniform Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/Uniform/icdf_uniform'

figure();
plot(x, hazard_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$','$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('Hazard function of Uniform Distribution')
xx=xlabel('Failure time');
yy=ylabel('Hazard rate');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/Uniform/hazard_uniform'

figure();
plot(x, c_hazard_uniform, 'linewidth',2)
grid on;
axis tight;
```

```
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$'
    ,'$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('Cumulative HF of Uniform Distribution')
xx=xlabel('x');
yy=ylabel('Cumulative Hazard');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    Uniform/c_hazard_uniform'

figure();
plot(x, survival_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$'
    ,'$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('Survival function of Uniform Distribution')
xx=xlabel('x');
yy=ylabel('Probability');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    Uniform/survival_uniform'

figure();
plot(q, i_survival_uniform, 'linewidth',2)
grid on;
axis tight;
ll=legend('$\alpha = 0, \beta = 1$','$\alpha = 0, \beta = 2$'
    ,'$\alpha = 0, \beta = 5$','$\alpha = 0, \beta = 10$')
tt=title('Inverse Survival function of Uniform Distribution')
xx=xlabel('Probability');
yy=ylabel('x');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
```

```
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);
print -depsc '/home/ioannis/Desktop/Distributions Theory/
    Uniform/i_survival_uniform'
```

A.2.8 Comparison between Normal & Student-t Distribution

Computations

```
%% COMPARISON NORMAL, STUDENT-T
mu = 0          % mean
sigma = 1;      % std
x = [-5:.01:5];
pdf_normal = pdf('Normal',x,mu,sigma);
df=1;
pdf_t(1,:) = pdf('T',x,df);
df=5;
pdf_t(2,:) = pdf('T',x,df);
df=30;
pdf_t(3,:) = pdf('T',x,df);

figure();
plot(x, pdf_normal, x, pdf_t(1,:), x, pdf_t(2,:), x, pdf_t
    (3,:), 'linewidth',2)
hold on;
plot(x, pdf_t(3,:),'g', 'linewidth',2)
grid on;
axis tight;
tt=title('Normal and Student t Distribution Comparison')
ll=legend('Normal $(\mu=0, \sigma=1)$','Student-t $(df=1)$','
    Student-t $(df=5)$','Student-t $(df=30)$')
xx=xlabel('x');
yy=ylabel('PDF');
set(xx,'interpreter', 'latex',  'fontsize',18);
set(yy,'interpreter', 'latex',  'fontsize',18);
set(tt,'interpreter', 'latex',  'fontsize',18);
set(ll,'interpreter', 'latex',  'fontsize',18);

```

```

print -depsc '/home/ioannis/Desktop/Distributions Theory/
    NORMAL/Norm-stud-t_comparison'
```

A.3 Generation of Random Distributed Numbers

```
%Dimensions
% 1.normal
% 2.student t
% 3.gamma
% 4.weibull
% 5.log-normal
% 6.uniform
% 7.power exponential (possible)
%%%%%%%%%%%%%% ONLY RANDOM DISTRIBUTED NUMBERS  ----WITHOUT
    EXPECTED RESPONSE %%%%%%%%%%%%%%%%%%%%%%
% LINE 1-96 %
%%%%%%%%%%%%%% LINEAR MODEL E = a*X  +  distributed_Noise for
     various noises %%%%%%%%%%%%%%%%%%%%%%%
% LINE 97- TILL END
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all;
clear all;
clc
%%%%%%%normal%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mu = 0
Normal_distributed = [];
for sigma=linspace(0.4,2.4,200)    % [0.4:2.4]
    r = random('Normal',mu,sigma,[1,1000]);
    Normal_distributed = cat(1,Normal_distributed,r);
end
Z=Normal_distributed;
%save('Normal_distributed.mat','Normal_distributed','-v7.3');
%%%%%%%%%%%student-t%%%%%%%%%%%%%%%%%%%%%%%%%%
t_distributed = [];
for df=linspace(1,40,40)      % [0.4:2.4]
```

```
    r = random('T',df,[5,1000]);
    t_distributed = cat(1,t_distributed,r);
end
Z=cat(3,Z,t_distributed);
 %save('t_distributed.mat','t_distributed','-v7.3');
%%%%%%%%%%%gamma%%%%%%%%%%%%%%%%%%%%%%%
k=5; %shape
gamma_distributed = [];
for theta=linspace(0.23,1.2,200)     % [0.23:1.2] scale
    r = random('Gamma',k,theta,[1,1000]);
    gamma_distributed = cat(1,gamma_distributed,r);
end
Z=cat(3,Z,gamma_distributed);
 %save('gamma_distributed.mat','gamma_distributed','-v7.3');
%%%%%%%%%%%weibull%%%%%%%%%%%%%%%%%%%%%%%%
lambda=linspace(1.32,7.7,1000);
k = 2.5 %linspace(2.1,6,1000);
weibull_distributed = [];
for lambda=linspace(1.0,6.0,200)    % [1.32:7.0] 1->0
   megalwnei fim, 7->apeiro megalwnei sep
    r = random('Weibull',lambda,k,[1,1000]);
    weibull_distributed = cat(1,weibull_distributed,r);
end
Z=cat(3,Z,weibull_distributed);
 %save('weibull_distributed.mat','weibull_distributed','-v7
     .3');
%%%%%%%%%%%LN%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mi = .7;
LN_distributed = [];
for sigma=linspace(.2,1.2,200)  % [0.2:1.2]
    r = random('LogNormal',mi,sigma,[1,1000]);
    LN_distributed = cat(1,LN_distributed,r);
end
Z=cat(3,Z,LN_distributed);
%save('LN_distributed.mat','LN_distributed','-v7.3');
%%%%%%%%Uniform %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
low_a = 0;
uni_distributed = [];
for high_b=linspace(1,30,200)     % [0.4:2.4]
```

```
    r = random('Uniform',low_a,high_b,[1,1000]);
    uni_distributed = cat(1,uni_distributed,r);
end
Z=cat(3,Z,uni_distributed);
%save('uni_distributed.mat','uni_distributed','-v7.3');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%https://en.wikipedia.org/wiki/Inverse_transform_sampling
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
PE=[]; a = 1;  %scale
x = -10:0.001:10;
 for b=linspace(.5,4.5,200);    % [0.5:4.5]
    rng(333);

    P=b/(2*a*gamma(1/b))*exp(-(abs(x)/a).^b);
    % create cdf
    cdf= cumtrapz(x, P);
    % number of required random draws
    n= 1;
    % generate uniformly distributed random numbers from
       [0,1]
    r= rand(n,1000);
    % generate random numbers from the desired pdf; inverse
       transform sampling
    [cdf, index]= unique(cdf);
    laplrnd= interp1(cdf, x(index), r);
    PE=cat(1,PE,laplrnd);
 end
Z=cat(3,Z,PE);
%save('PE_distributed.mat','PE_distributed','-v7.3');

% rng(333)
% x = -100:0.001:100;
% a = 1;       %scale
% b = 1;
% P=b/(2*a*gamma(1/b))*exp(-(abs(x)/a).^b)
% % create cdf
% cdf = cumtrapz(x, P);
```

```
% n = 1;
% r = rand(n,860);
% [cdf, index] = unique(cdf);
% laplrnd = interp1(cdf, x(index), r);
% PE_distributed = laplrnd ;
% %save('PE_distributed.mat','PE_distributed','-v7.3');

%%%  POWER EXPONENTIAL THROUGH GENERALIZED NORMAL
    DISTRIBUTION %%
% close all;
% clear all;
% clc
%
% PE_distributed=[];
% mu = 0;           % Mean
% sigma = 1;        % Scale
% %beta = 2;         % Shape
% n = 860;          % Number of random samples
%
% for beta=linspace(10,14.5,1999);
% % Generate random numbers
% r = power_exp_rnd(0, 1, 10000000, 1001);
% PE_distributed = cat(1,PE_distributed,transpose(r));
% end
% %save('PE_distributed_TEST22_091.mat','PE_distributed','-v7
    .3');

%hist(Z(100,:,2),30)
ZZ=[];
t_distributed_a = [];
for df=2:3:5     % [0.4:2.4]
    r = random('T',df,[1000,1000]);
    t_distributed_a = cat(1,t_distributed_a,r);
end
r = random('T',30,[1000,1000]);
t_distributed_a = cat(1,t_distributed_a,r);
ZZ=cat(3,ZZ,t_distributed_a);
```

A.4 FSIP of Random Distributed Numbers

```
Y = Z;
is = 1; %   Set subject index
Nind = size(Y, 3); % Number of individuals
Nc = size(Y, 1); % Number of voxels
N = size(Y, 2); % Number of time points

%======================================================

sepa = zeros(Nind, Nc);
fima = zeros(Nind, Nc);

% For testing set Nind = 1


for i=1: Nind
    for j=1: Nc
        xij = squeeze(Y(j, :,i));
        [ ~, sep, fim ] = FSIP2( xij, N );
        sepa(i,j) = sep;
        fima(i,j) = fim;
    end;
end


%Normal Dist FSIP Scatter
figure();
scatter(sepa(1,1:50), fima(1,1:50), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(1,51:100), fima(1,51:100), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(1,101:150), fima(1,101:150), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(1,151:200), fima(1,151:200), '.', 'm', '
```

```
linewidth', 3);
axis tight;
ll=legend({'$\sigma=0.4 - 0.9$', '$\sigma=0.9 - 1.4$', '$\
    sigma=1.4 - 1.9$', '$\sigma=2.0 - 2.4$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    RandomNumbers/Normal'

%Stud Dist FSIP Scatter
figure();
scatter(sepa(2,1:50), fima(2,1:50), '.', 'b', 'linewidth', 3)
    ; %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(2,51:100), fima(2,51:100), '.', 'r', 'linewidth'
    , 3);
hold on;
scatter(sepa(2,101:150), fima(2,101:150), '.', 'g', '
    linewidth', 3);
hold on;
scatter(sepa(2,151:200), fima(2,151:200), '.', 'm', '
    linewidth', 3);
axis tight;
ll=legend({'$df=1 - 10$', '$df=10 - 20$', '$df=20 - 30$', '
    $df=30 - 40$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');
```

```

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/RandomNumbers/studentt'

%Gamma Dist FSIP Scatter
figure();
scatter(sepa(3,1:50), fima(3,1:50), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(3,51:100), fima(3,51:100), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(3,101:150), fima(3,101:150), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(3,151:200), fima(3,151:200), '.', 'm', 'linewidth', 3);
axis tight;
ll=legend({'$\lambda=0.2 - 0.45$', '$\lambda=0.45 - 0.7$', '$\lambda=0.7 - 0.95$', '$\lambda=0.95 - 1.2$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/RandomNumbers/Gamma'

%Weibull Dist FSIP Scatter
figure();
scatter(sepa(4,1:50), fima(4,1:50), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
```

```
scatter(sepa(4,51:100), fima(4,51:100), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(4,101:150), fima(4,101:150), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(4,151:200), fima(4,151:200), '.', 'm', 'linewidth', 3);
axis tight;
ll=legend({'$\lambda=1.0 - 2.25$', '$\lambda=2.25 - 3.5$', '$\lambda=3.5- 4.75$', '$\lambda=4.75 - 6.0$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/RandomNumbers/Weibull'

%log-normal Dist FSIP Scatter
figure();
scatter(sepa(5,1:50), fima(5,1:50), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(5,51:100), fima(5,51:100), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(5,101:150), fima(5,101:150), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(5,151:200), fima(5,151:200), '.', 'm', 'linewidth', 3);
axis tight;
ll=legend({'$\sigma=0.2- 0.45$', '$\sigma=0.45 - 0.7$', '$\
```

```
    sigma=0.7 - 0.95$', '$\sigma=0.95 - 1.2$'}, 'FontSize',15)
    ;
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    RandomNumbers/LN'

X=linspace(0.2,10,1000);
Y=1./X;

%uniform Dist FSIP Scatter
figure();
scatter(sepa(6,1:25), fima(6,1:25), '.', 'b', 'linewidth', 3)
    ; %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(6,26:50), fima(6,26:50), '.', 'r', 'linewidth',
    3);
hold on;
scatter(sepa(6,51:75), fima(6,51:75), '.', 'g', 'linewidth',
    3);
hold on;
scatter(sepa(6,76:100), fima(6,76:100), '.', 'm', 'linewidth'
    , 3);
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
xlim([0 10])
ll=legend({'$\alpha=0, \beta= 1-4.5$', '$\alpha=0, \beta
    =4.5-8$', '$\alpha=0, \beta =8-11.5$', '$\alpha=0, \beta
    =11.5-15$', 'Gaussian Limit'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
```

```
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    RandomNumbers/Uniform'


%power exponential Dist FSIP Scatter
figure();
scatter(sepa(7,1:50), fima(7,1:50), '.', 'b', 'linewidth', 3)
    ; %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(7,51:100), fima(7,51:100), '.', 'r', 'linewidth'
    , 3);
hold on;
scatter(sepa(7,101:150), fima(7,101:150), '.', 'g', '
    linewidth', 3);
hold on;
scatter(sepa(7,151:200), fima(7,151:200), '.', 'm', '
    linewidth', 3);
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
%xlim([0 10])
ll=legend({'$b -a= 1-4.5$', '$b -a=4.5-8$', '$b -a=8-11.5$',
    '$b -a=11.5-15$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
```

```
RandomNumbers/PE_gaussian'

%Stud Dist FSIP Scatter df 1 vs 3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%FSIP COMPUTATION %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Y = ZZ;
is = 1; %   Set subject index
Nind = size(Y, 3); % Number of individuals
Nc = size(Y, 1); % Number of voxels
N = size(Y, 2); % Number of time points

%==========================================

sepa = zeros(Nind, Nc);
fima = zeros(Nind, Nc);

% For testing set Nind = 1


for i=1: Nind
    for j=1: Nc
        xij = squeeze(Y(j, :,i));
        [ ~, sep, fim ] = FSIP2( xij, N );
        sepa(i,j) = sep;
        fima(i,j) = fim;
    end;
end

x=linspace(0.053,4,999);
y=1./x;

figure();
scatter(sepa(1,1:1000), fima(1,1:1000), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(1,1001:2000), fima(1,1001:2000), '.', 'g', 'linewidth', 3);
```

```
hold on;
scatter(sepa(1,2001:3000), fima(1,2001:3000), '.', 'o', '
    linewidth', 3);
% hold on;
% plot(x,y, 'r', 'linewidth', 3);
xlim([0 4])
ylim([0 1.2])
ll=legend({'$df=1$', '$df=3$', '$df=30$', 'Gaussian Limit'},
    'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Random Distributed Numbers', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    RandomNumbers/studentt_1_3_30'
```

A.5 Generation of Synthetic Data

```
%Dimensions
% 1.normal
% 2.student t
% 3.gamma
% 4.weibull
% 5.log-normal
% 6.uniform
% 7.power exponential (possible)
%%%%%%%%%%%%%%%%%% normal noise %%%%%%%%%%%%%%%%%%%%%%%
% random('Normal',mu,sigma,[1,1000]);

close all;
clear all;
clc
dbclear all

```

```
load Expected_responses
X(1:140) = [Expected_responses, Expected_responses(21:80)];
V = uint32(1):uint32(260);
XY=[X,X(21:140),X(21:140),X(21:140),X(21:140),X(21:140),X
    (21:140)];


Z1= [];
n = 860;          % Number of random samples
mu = 0;
for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for sigma=linspace(0.5,4.5,200);
                                                    %[200 values] for
        every k we try different values for sigma
        r = random('Normal',mu,sigma,[1,860]);
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp +
            e_gaussian
        Z1 = cat(1,Z1,rr);
                                                           %we add
            the new value into the vector
    end
end


Z=Z1;

%plot([1:140],Z(1,1:140))
%save('Exp_normal_distributed.mat','Z','-v7.3');

%%%%%%%%%%%%%%%%%% student t noise %%%%%%%%%%%%%%%%%%%%%%%%%
% random('T',1,[1000,100]);

Z1 = [];
n = 860;          % Number of random samples

for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for df=linspace(1,200,200); %degrees of freedom
        r = random('T',df,[1,860]);
```

```
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp + e_gaussian
        Z1 = cat(1,Z1,rr);       %we add the new value into the vector
    end
end

Z=cat(3,Z,Z1);


%plot([1:140],Z(1,1:140))
%save('Exp_student_distributed.mat','Z','-v7.3');


%%%%%%%%%%%%%%%%%% gamma noise %%%%%%%%%%%%%%%%%%%%%%%
%random('Gamma',k,theta,[1,1000]);

Z1 = [];
n = 860;          % Number of random samples
kappa=5;

for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for theta=linspace(0.23,1.2,200); %degrees of freedom
        r = random('Gamma',kappa,theta,[1,860]);
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp + e_gaussian
        Z1 = cat(1,Z1,rr);  %we add the new value into the vector
    end
end

Z=cat(3,Z,Z1);

%plot([1:140],Z(1,1:140))
%save('Exp_gamma_distributed.mat','Z','-v7.3');

%%%%%%%%%%%%%%%%%% weibull noise %%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%random('Weibull',lambda,k,[1,1000]);

Z1 = [];
n = 860;         % Number of random samples
kappa =2.5;
for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for lambda=linspace(1.0,6.0,200); %degrees of freedom %[200 values] for every k we try different values for sigma
        r = random('Weibull',lambda,kappa,[1,860]);
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp + e_gaussian
        Z1 = cat(1,Z1,rr);                                    %we add the new value into the vector
    end
end

Z=cat(3,Z,Z1);

%plot([1:140],Z(1,1:140))
%save('Exp_weibull_distributed.mat','Z','-v7.3');


%%%%%%%%%%%%%%%%%% LogNormal noise %%%%%%%%%%%%%%%%%%%%%%%
%random('LogNormal',mi,sigma,[1,1000]);

Z1 = [];
n = 860;         % Number of random samples
mi=.7;
for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for sigma=linspace(.2,1.2,200); %degrees of freedom %[200 values] for every k we try different values for sigma
        r = random('LogNormal',mi,sigma,[1,860]);
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp + e_gaussian
```

```
        Z1 = cat(1,Z1,rr);                                              %we add the new value into the vector
    end
end

Z=cat(3,Z,Z1);

%plot([1:140],Z(1,1:140))
%save('Exp_ln_distributed.mat','Z','-v7.3');


%%%%%%%%%%%%%%%%%% Uniform noise %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%random('Uniform',low_a,high_b,[1,1000]);

Z1 = [];
n = 860;          % Number of random samples
low_a=0;
for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
    for high_b=linspace(1,30,200); %degrees of freedom                  %[200 values] for every k we try different values for sigma
        r = random('Uniform',low_a,high_b,[1,860]);
        %r = transpose(r);
        rr =k.*XY + r;          %we make a new E(t) = k*Exp + e_gaussian
        Z1 = cat(1,Z1,rr);                                              %we add the new value into the vector
    end
end

Z=cat(3,Z,Z1);

%plot([1:140],Z(1,1:140))
%save('Exp_Uniform_distributed.mat','Z','-v7.3');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% POWER
```

```
    EXPONENTIAL %%%%%%%%%%%% (attention)
    %%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%https://en.wikipedia.org/wiki/Inverse_transform_sampling
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5
a = 1;       %scale
%b = 1;
Z1=[];
x = -10:0.001:10;
for k=0.1:0.4:1.8        %5 values 0.1 0.5 0.9 1.3 1.7
 for b=linspace(.5,8.5,200);    % [0.5:4.5]
    rng(333);
    P=b/(2*a*gamma(1/b))*exp(-(abs(x)/a).^b);
    % create cdf
    cdf          = cumtrapz(x, P);
    % number of required random draws
    n = 1;
    % generate uniformly distributed random numbers from
        [0,1]
    r = rand(n,860);
    % generate random numbers from the desired pdf; inverse
        transform sampling
    [cdf, index] = unique(cdf);

    r = interp1(cdf, x(index), r);
    rr =k.*XY + r;
    Z1=cat(1,Z1,rr);
 end
end
 Z=cat(3,Z,Z1);

% PE_distributed =PE ;
% hist(PE(201,:),100)
% save('PE_distributed.mat','PE_distributed','-v7.3');
%
% rng(333)
% x = -100:0.001:100;
```

```
% a = 1;        %scale
% b = 1;
% P=b/(2*a*gamma(1/b))*exp(-(abs(x)/a).^b)
% % create cdf
% cdf = cumtrapz(x, P);
% n = 1;
% r = rand(n,860);
% [cdf, index] = unique(cdf);
% laplrnd = interp1(cdf, x(index), r);
% PE_distributed = laplrnd ;
% save('PE_distributed.mat','PE_distributed','-v7.3');
%
% %%%%%%  POWER EXPONENTIAL THROUGH GENERALIZED NORMAL
    DISTRIBUTION %%
% close all;
% clear all;
% clc
%
% PE_distributed=[];
% mu = 0;           % Mean
% sigma = 1;        % Scale
% %beta = 2;          % Shape
% n = 860;         % Number of random samples
%
% for beta=linspace(10,14.5,1999);
% % Generate random numbers
% r = power_exp_rnd(0, 1, 10000000, 1001);
% PE_distributed = cat(1,PE_distributed,transpose(r));
% end
% save('PE_distributed_TEST22_091.mat','PE_distributed','-v7
    .3');
%
% hist(PE_distributed,100)
```

A.6 FSIP of Synthetic Data

```
%% expected responses FSIP for synthetic data

```

```
Y = Z;
is = 1; %   Set subject index
Nind = size(Y, 3); % Number of individuals
Nc = size(Y, 1); % Number of voxels
N = size(Y, 2); % Number of time points

%===========================================

sepa = zeros(Nind, Nc);
fima = zeros(Nind, Nc);

% For testing set Nind = 1


for i=1: Nind
    for j=1: Nc
        xij = squeeze(Y(j, :,i));
        [ ~, sep, fim ] = FSIP2( xij, N );
        sepa(i,j) = sep;
        fima(i,j) = fim;
    end;
end

%synthetic with normal noise
figure();
scatter(sepa(1,1:200), fima(1,1:200), '.', 'b', 'linewidth',
    3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(1,201:400), fima(1,201:400), '.', 'r', '
    linewidth', 3);
hold on;
scatter(sepa(1,401:600), fima(1,401:600), '.', 'g', '
    linewidth', 3);
hold on;
scatter(sepa(1,601:800), fima(1,601:800), '.', 'm', '
    linewidth', 3);
hold on;
scatter(sepa(1,801:1000), fima(1,801:1000), '.', 'k', '
```

```
    linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, \sigma=0.4 - 2.4$', '$\alpha=0.5, \
    sigma=0.4 - 2.4$', '$\alpha=0.9, \sigma=0.4 - 2.4$', '$\
    alpha=1.3, \sigma=0.4 - 2.4$', '$\alpha=1.7, \sigma=0.4 -
    2.4$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
xlim([0 7]);
ylim([0 3.5]);
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    Synthetic/Exp_Norm'

%synthetic with student noise
figure();
scatter(sepa(2,1:200), fima(2,1:200), '.', 'b', 'linewidth',
    3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(2,201:400), fima(2,201:400), '.', 'r', '
    linewidth', 3);
hold on;
scatter(sepa(2,401:600), fima(2,401:600), '.', 'g', '
    linewidth', 3);
hold on;
scatter(sepa(2,601:800), fima(2,601:800), '.', 'm', '
    linewidth', 3);
hold on;
scatter(sepa(2,801:1000), fima(2,801:1000), '.', 'k', '
    linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, df=1 - 40$', '$\alpha=0.5, df=1 - 40
    $', '$\alpha=0.9, df=1 - 40$', '$\alpha=1.3, df=1 - 40$',
```

```
    '$\alpha=1.7, df=1 - 40$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/
    Synthetic/Exp_t'

%synthetic with gamma noise
figure();
scatter(sepa(3,1:200), fima(3,1:200), '.', 'b', 'linewidth',
    3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(3,201:400), fima(3,201:400), '.', 'r', '
    linewidth', 3);
hold on;
scatter(sepa(3,401:600), fima(3,401:600), '.', 'g', '
    linewidth', 3);
hold on;
scatter(sepa(3,601:800), fima(3,601:800), '.', 'm', '
    linewidth', 3);
hold on;
scatter(sepa(3,801:1000), fima(3,801:1000), '.', 'k', '
    linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, \theta=0.2 - 1.2$', '$\alpha=0.5, \
    theta=0.2 - 1.2$', '$\alpha=0.9, \theta=0.2 - 1.2$', '$\
    alpha=1.3, \theta=0.2 - 1.2$', '$\alpha=1.7, \theta=0.2 -
    1.2$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
```

```
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/Synthetic/Exp_Gam'

%synthetic with weibull noise
figure();
scatter(sepa(4,1:200), fima(4,1:200), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(4,201:400), fima(4,201:400), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(4,401:600), fima(4,401:600), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(4,601:800), fima(4,601:800), '.', 'm', 'linewidth', 3);
hold on;
scatter(sepa(4,801:1000), fima(4,801:1000), '.', 'k', 'linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, \lambda=1.0 - 6.0$', '$\alpha=0.5, \lambda=1.0 - 6.0$', '$\alpha=0.9, \lambda=1.0 - 6.0$', '$\alpha=1.3, \lambda=1.0 - 6.0$', '$\alpha=1.7, \lambda=1.0 - 6.0$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/Synthetic/Exp_Wei'
```

```
%synthetic with log-normal noise
figure();
scatter(sepa(5,1:200), fima(5,1:200), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(5,201:400), fima(5,201:400), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(5,401:600), fima(5,401:600), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(5,601:800), fima(5,601:800), '.', 'm', 'linewidth', 3);
hold on;
scatter(sepa(5,801:1000), fima(5,801:1000), '.', 'k', 'linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, \sigma=0.2 - 1.2$', '$\alpha=0.5, \sigma=0.2 - 1.2$', '$\alpha=0.9, \sigma=0.2 - 1.2$', '$\alpha=1.3, \sigma=0.2 - 1.2$', '$\alpha=1.7, \sigma=0.2 - 1.2$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/Synthetic/Exp_LN'

X=linspace(0.15,10,1000);
Y=1./X;

%synthetic with uniform noise
figure();
scatter(sepa(6,1:200), fima(6,1:200), '.', 'b', 'linewidth',
```

```
3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
grid on;
scatter(sepa(6,201:400), fima(6,201:400), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(6,401:600), fima(6,401:600), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(6,601:800), fima(6,601:800), '.', 'm', 'linewidth', 3);
hold on;
scatter(sepa(6,801:1000), fima(6,801:1000), '.', 'k', 'linewidth', 3);
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
xlim([0 6])
ll=legend({'$\alpha =0.1, b -a= 1-4.5$', '$$\alpha =0.5, b -a=4.5-8$', '$$\alpha =0.9, b -a=8-11.5$', '$$\alpha =1.3, b -a=11.5-15$', '$$\alpha =1.7, b -a=11.5-15$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/Synthetic/Exp_uni'


%synthetic with power exponential noise
figure();
scatter(sepa(7,1:200), fima(7,1:200), '.', 'b', 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
```

```
grid on;
scatter(sepa(7,201:400), fima(7,201:400), '.', 'r', 'linewidth', 3);
hold on;
scatter(sepa(7,401:600), fima(7,401:600), '.', 'g', 'linewidth', 3);
hold on;
scatter(sepa(7,601:800), fima(7,601:800), '.', 'm', 'linewidth', 3);
hold on;
scatter(sepa(7,801:1000), fima(7,801:1000), '.', 'k', 'linewidth', 3);
axis tight;
ll=legend({'$\alpha=0.1, \sigma=0.2 - 1.2$', '$\alpha=0.5, \sigma=0.2 - 1.2$', '$\alpha=0.9, \sigma=0.2 - 1.2$', '$\alpha=1.3, \sigma=0.2 - 1.2$', '$\alpha=1.7, \sigma=0.2 - 1.2$'}, 'FontSize',15);
xx=xlabel('SEP');
yy=ylabel('FIM');
tt=title('FSIP - Synthetic fMRI Data', 'fontsize',18);
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
set(tt,'interpreter', 'latex');
set(ll,'interpreter', 'latex');

print -depsc '/home/ioannis/Desktop/THESIS NEO/7. Last_11_11/Synthetic/Exp_PE'
```

A.7 Creation of Expected Response

```
a1=6;
a2=12;
b1=.9;
b2=.9;
d1 =a1*b1;
d2=a2*b2;
c=0.25;
t=[0:1:80]
```

```
tt=[t t]

h=(t./d1).^a1.*exp(-((t-d1)/b1)) - c*(t./d2).^a2.*exp(-((t-d2
    )/b2));

figure();
x=linspace(0,8,1002);
y=0.15;
stimulus=y*square(5*x)+y;
plot(t,h,x,stimulus)
xlim([0 25])
w=conv(stimulus,h);
```

A.8 FSIP of Expected Response

```
%% FSIP for Expected Response
% scalar a multiplication with ideal expected response
y = linspace(0.01, 10.001, 10000);
for j=1:10000
    X(1:80)= y(j).*Expected_responses;
    Vox_10000(j,:) = X;
end


load('a_0.1_to_2_Exp_10000samples.mat')
Y = Vox_10000;

is = 1; %   Set subject index
% Nind = size(Y, 1); % Number of individuals
Nc = size(Y, 1); % Number of voxels
N = size(Y, 2); % Number of time points
%==============================================
sepa = zeros(1, Nc);
fima = zeros(1, Nc);
% For testing set Nind = 1
for i=1: 1
    for j=1: Nc
        xij = squeeze(Y(j, :));
```

```
        [ ~, sep, fim ] = FSIP2( xij, N );
        sepa(i,j) = sep;
        fima(i,j) = fim;

    end;

end

% gaussian limit
x=linspace(0.053,2.84,999);
y=1./x;

figure();
plot(sepa(100:2001), fima(100:2001), 'linewidth', 3); %%
    1:201 :: 1005 for exp with norm distrib
hold on;
plot(sepa(2002:4002), fima(2002:4002), 'linewidth', 3);
hold on;
plot(sepa(4003:6003), fima(4003:6003), 'linewidth', 3);
hold on;
plot(sepa(6004:8004), fima(6004:8004), 'linewidth', 3);
hold on;
plot(sepa(8005:10000), fima(8005:10000), 'linewidth', 3);
hold on;
plot(x,y, 'r', 'linewidth', 3);
grid on;
% legend({'snr=10', 'snr=8', 'snr=6', 'snr=4', 'snr=2', 'snr
    =0', 'snr=-2', 'snr=-4', 'snr=-6'}, 'FontSize',18);
% legend({'a=0.01', 'a=0.1', 'a=0.5', 'a=1', 'a=1.5', 'a
    =2.5', 'a=3.5', 'a=4.5', 'a=5'}, 'FontSize',18);
ll=legend({'$\alpha=0.15 - 2$', '$\alpha=2 - 4$', '$\alpha=4
    - 6$', '$\alpha=6 - 8$', '$\alpha=8 - 10$', 'Gaussian
    Limit'}, 'FontSize',15);
set(ll,'interpreter', 'latex');
axis tight;
xx=xlabel('SEP');
yy=ylabel('FIM');
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
```

```
tt=title('FSIP for Expected FMRI Response', 'fontsize',18);
set(tt,'interpreter', 'latex');

% print -depsc '/home/ioannis/Desktop/FSIP CALCULATIONS/exp_mult_factor'
=log(x); %*********************************
y=log(y); %*********************************

fima=log(fima); %*********************************
sepa=log(sepa); %*********************************

figure();
plot(sepa(100:2001), fima(100:2001), 'linewidth', 3); %% 1:201 :: 1005 for exp with norm distrib
hold on;
plot(sepa(2002:4002), fima(2002:4002), 'linewidth', 3);
hold on;
plot(sepa(4003:6003), fima(4003:6003), 'linewidth', 3);
hold on;
plot(sepa(6004:8004), fima(6004:8004), 'linewidth', 3);
hold on;
plot(sepa(8005:10000), fima(8005:10000), 'linewidth', 3);
hold on;
plot(x,y, 'r', 'linewidth', 3);
grid on;
% legend({'snr=10', 'snr=8', 'snr=6', 'snr=4', 'snr=2', 'snr=0', 'snr=-2', 'snr=-4', 'snr=-6'}, 'FontSize',18);
% legend({'a=0.01', 'a=0.1', 'a=0.5', 'a=1', 'a=1.5', 'a=2.5', 'a=3.5', 'a=4.5', 'a=5'}, 'FontSize',18);
ll=legend({'$\alpha=0.15 - 2$', '$\alpha=2 - 4$', '$\alpha=4 - 6$', '$\alpha=6 - 8$', '$\alpha=8 - 10$', 'Gaussian Limit'}, 'FontSize',15);
set(ll,'interpreter', 'latex');
axis tight;
xx=xlabel('SEP');
yy=ylabel('FIM');
set(xx,'interpreter', 'latex');
set(yy,'interpreter', 'latex');
tt=title('FSIP for Expected FMRI Response [log]', 'fontsize'
```

```
    ,18);
set(tt,'interpreter', 'latex');
```

A.9 FSIP of Real fMRI Data

A.9.1 FSIP of 25 Subjects in 5x5 Figure

```
%% 25 SUBJECTS 1BY1 WITH GAUSSIAN LIMIT
close all;
clear all;
clc
dbclear all

X=linspace(0.15,10,1000);
Y=1./X;

X=linspace(0.05,1.5,1000);
Y=1./X;

load('FSIP_measures_SF.mat')

figure();
for i = 1:25
    smplot(5,5,i,'axis','on')
    scatter(sepa(i,:), fima(i,:), 'k.');
    hold on;
    plot(X,Y, 'r', 'linewidth', 3);
    xlim([0 1.5])
    ylim([0 5])
    xx=xlabel('SEP', 'fontsize',12);
    yy=ylabel('FIM', 'fontsize',12);
    set(xx,'interpreter', 'latex');
    set(yy,'interpreter', 'latex');
    hold on;
end


load('FSIP_measures_FS.mat')
```

```

figure();
for i = 1:25
    smplot(5,5,i,'axis','on')
    scatter(sepa(i,:), fima(i,:), 'k.');
    hold on;
    plot(X,Y, 'r', 'linewidth', 3);
    xlim([0 1.5])
    ylim([0 5])
    x=xlabel('SEP', 'fontsize',12);
    y=ylabel('FIM', 'fontsize',12);
    set(x,'interpreter', 'latex');
    set(y,'interpreter', 'latex');
end

load('FSIP_measures_bowl.mat')

figure();
for i = 1:25
    smplot(5,5,i,'axis','on')
    scatter(sepa(i,:), fima(i,:), 'k.');
    hold on;
    plot(X,Y, 'r', 'linewidth', 3);
    xlim([0 1.5])
    ylim([0 5])
    x=xlabel('SEP', 'fontsize',12);
    y=ylabel('FIM', 'fontsize',12);
    set(x,'interpreter', 'latex');
    set(y,'interpreter', 'latex');
end

load('FSIP_measures_aimless.mat')

figure();
for i = 1:25
    smplot(5,5,i,'axis','on')
    scatter(sepa(i,:), fima(i,:), 'k.');
    hold on;
    plot(X,Y, 'r', 'linewidth', 3);
```

```
    xlim([0 1.5])
    ylim([0 5])
    x=xlabel('SEP', 'fontsize',12);
    y=ylabel('FIM', 'fontsize',12);
    set(x,'interpreter', 'latex');
    set(y,'interpreter', 'latex');
end
```

A.9.2 FSIP of 25 Subjects in 1 Figure

```
%% ALL 25 SUBJECTS' VOXEL IN ONE PLOT

load('FSIP_measures_SF.mat')


sepv = sepa(:);
fimv = fima(:);
figure();
scatter(sepv, fimv, 'k.')
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
xlim([0 1.5])
ylim([0 20])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
tt=title('FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');

load('FSIP_measures_FS.mat')

sepv = sepa(:);
fimv = fima(:);
figure();
scatter(sepv, fimv, 'k.')
hold on;
plot(X,Y, 'r', 'linewidth', 3);
```

```
axis tight;
xlim([0 1.5])
ylim([0 20])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
tt=title('FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');

load('FSIP_measures_bowl.mat')

sepv = sepa(:);
fimv = fima(:);
figure();
scatter(sepv, fimv, 'k.')
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
xlim([0 1.5])
ylim([0 20])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
tt=title('FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');

load('FSIP_measures_aimless.mat')

sepv = sepa(:);
fimv = fima(:);
figure();
scatter(sepv, fimv, 'k.')
hold on;
plot(X,Y, 'r', 'linewidth', 3);
axis tight;
xlim([0 1.5])
ylim([0 20])
```

```
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
tt=title('FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');
```

A.9.3 FSIP of 25 Subjects Average per Voxel

```
%% AVERAGE PER VOXEL OF 25 SUBJECTS

load('FSIP_measures_SF.mat')
for i=1:253789
       fima_all(i)=sum(fima(:,i));
       fima_all(i)=fima_all(i)/25;
       sepa_all(i)=sum(sepa(:,i));
       sepa_all(i)=sepa_all(i)/25;
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Average FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');

load('FSIP_measures_FS.mat')
for i=1:253789
       fima_all(i)=sum(fima(:,i));
       fima_all(i)=fima_all(i)/25;
       sepa_all(i)=sum(sepa(:,i));
       sepa_all(i)=sepa_all(i)/25;
end
figure();
```

```
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Average FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');

load('FSIP_measures_bowl.mat')
for i=1:253789
       fima_all(i)=sum(fima(:,i));
       fima_all(i)=fima_all(i)/25;
       sepa_all(i)=sum(sepa(:,i));
       sepa_all(i)=sepa_all(i)/25;
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Average FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');

load('FSIP_measures_aimless.mat')
for i=1:253789
       fima_all(i)=sum(fima(:,i));
       fima_all(i)=fima_all(i)/25;
       sepa_all(i)=sum(sepa(:,i));
       sepa_all(i)=sepa_all(i)/25;
end
```

```
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Average FSIP Values for Real Data', 'fontsize',18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
```

A.9.4 FSIP of 25 Subjects Median per Voxel

```
%%  MEDIAN PER VOXEL OF 25 SUBJECTS
%median(A,dim)

X=linspace(0.05,1.5,1000);
Y=1./X;

load('FSIP_measures_SF.mat')
for i=1:253789
        fima_all(i)=median(fima(:,i));
        sepa_all(i)=median(sepa(:,i));
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Median of FSIP Values for Real Data', 'fontsize'
    ,18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
```

```
load('FSIP_measures_FS.mat')
for i=1:253789
       fima_all(i)=median(fima(:,i));
       sepa_all(i)=median(sepa(:,i));
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Median of FSIP Values for Real Data', 'fontsize'
    ,18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');

load('FSIP_measures_bowl.mat')
for i=1:253789
       fima_all(i)=median(fima(:,i));
       sepa_all(i)=median(sepa(:,i));
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Median of FSIP Values for Real Data', 'fontsize'
    ,18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');

```

```
load('FSIP_measures_aimless.mat')
for i=1:253789
       fima_all(i)=median(fima(:,i));
       sepa_all(i)=median(sepa(:,i));
end
figure();
scatter(sepa_all, fima_all, 'k.');
hold on;
plot(X,Y, 'r', 'linewidth', 3);
xlim([0.6 1.05])
ylim([0 2])
x=xlabel('SEP', 'fontsize',18);
y=ylabel('FIM', 'fontsize',18);
tt=title('Median of FSIP Values for Real Data', 'fontsize'
    ,18)
set(tt,'interpreter', 'latex');
set(x,'interpreter', 'latex');
set(y,'interpreter', 'latex');
```