ANALYSIS OF MACHINE LEARNING ALGORITHMS IN EPILEPTIC SEIZURE PREDICTION BASED ON EEG DATA

THESIS

Submitted in partial fulfillment of the requirements of BITS F423T/BITS F421T, Thesis

by

Anubhav Chaturvedi

ID No - 2011A7TS079G

Under the supervision of

Dr. Raviprasad Aduri

Assistant Professor,

Dept. Of Biological Sciences

BITS, PILANI – K K BIRLA GOA CAMPUS

Acknowledgment

I would like to thank Dr. Raviprasad Aduri for his guidance and enthusiasm that allowed me to explore this inter-disciplinary topic in much details. His mentorship has been invaluable and has motivated me to learn more and solve problems that might initially appear to be a mammoth task.

I would also like to thank Prof. Bharat Deshpande, Mr. Ramprasad S. Joshi and the Dept. Of Computer Science, for their cooperation and allowing the access to the Kosambi compute cluster. This made parallel execution of experiments easy and made it viable to perform large amount of computation within the time constraints.

I would like to acknowledge the support from IEEG-Portal for providing data sets and making this study possible.

CERTIFICATE

This is to certify that the Thesis entitled, **ANALYSIS OF MACHINE LEARNING ALGORITHMS IN EPILEPTIC SEIZURE PREDICTION BASED ON EEG DATA**

is submitted by **Anubhav Chaturvedi** , ID No **. 2011A7TS079G**

in partial fulfillment of the requirements of BITS F423T/BITS F421T Thesis embodies the work done by him under my supervision.

Signature of the supervisor

Date : 8 May 2015

Name Dr. Raviprasad Aduri

Designation Assistant Professor

Dept. Of Biological Sciences

BITS Pilani, K K Birla Goa Campus

List of symbols & Abbreviations used

DBS	Deep Brain Stimulation
DFA	Detrended Fluctuation Analysis
ECG	Electrocardiogram
EEG	Electroencephalogram
FN	False Negative
FP	False Positive
iEEG	intracranial EEG
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
TN	True Negative
TP	True Positive

Abstract

The life of people suffering from epilepsy is affected by the seemingly random occurrence of seizures. A seizure that might occur in situations like driving a car may be fatal not only for the patient but also for those around them. In light of research on analyzing EEG signals for epilepsy patients, it can be shown that changes in brain activity, resulting in a seizure, start to develop much before the actual seizure. Previous studies on EEG data have shown that epileptic seizures are non-random events that can be predicted and provide hope for seizure forecasting using long term EEG recordings. This thesis work is mainly focused on developing prediction models that can distinguish pre-ictal events from the interictal events. Intracranial EEG data, from five Dogs suffering from naturally occurring Epilepsy was obtained from the UPenn - Mayo Clinic Seizure Detection Challenge. We trained and validated three different Machine Learning models: Support Vector Machine, k-Nearest Neighbors and Random Forest classifier, using 5-fold cross-validation. The resulting best models were tested on an isolated data set. The prediction performance remained better than chance performance in all the classifiers. One important factor to consider in assessing the performance of any seizure prediction method is to reduce the false negatives (i.e. a pre-ictal being classified as inter-ictal). Considering the same, our results clearly indicate that the k-Nearest Neighbors classifier performs better than the other two. Analysis of the different methods by varying the threshold also resulted in the k-Nearest Neighbors classifier being a superior one. Besides, considering the sensitivity of the prediction and the computational costs, our results showed that the k-Nearest Neighbors classifier is a better choice than Random Forest and SVM.

Contents

List of Tables

List of Figures

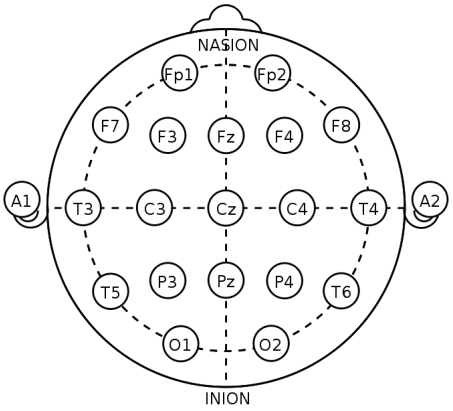
1 Introduction

Epilepsy is a neurological disorder that affects nearly 2% of the world population, seriously damaging the health and daily activities of those suffering. The disorder is characterized by interruptions of cerebral electric activities caused by abnormal hyper-synchronous discharges of neural populations referred to as seizures[4]. Seizures can result in lapse of attention or a whole-body convulsion. Anti-epileptic drugs form a major part of Epilepsy treatment but 20-40% of the patients continue to have seizure despite medication [8]. A significant cause of epileptic related disability for patients is the seemingly random onset of the seizure. The occurrence of such a seizure in public places could be harmful not only for the patient but also for those around them.

The most common way to detect a seizure activity is by analysis of the Electroencephalogram (EEG) data. EEG measures voltage fluctuations resulting from ionic current flows within the neurons of the brain. EEG can be of two types : Scalp EEG and intracranial EEG (iEEG) .

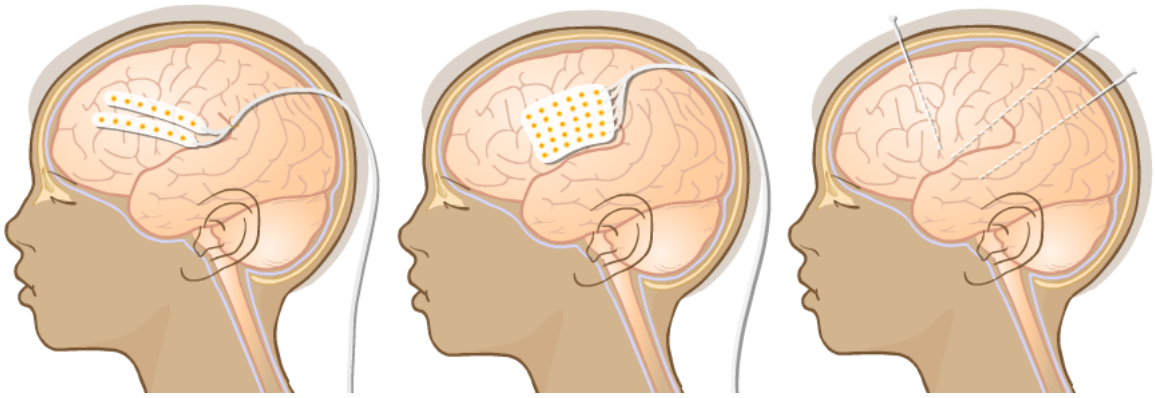
Scalp EEG is recorded by placing the electrodes on the scalp using some conductive gel after treating the scalp area with light abrasion in order to decrease the impedance resulting from dead skin cells. Generally, 19 electrodes are arranged around the scalp according to the specifications by the international 10-20 system.

Figure 1.1: The 10-20 system for Scalp EEG electrode placement



Intracranial EEG electrodes are placed directly in contact with the exposed brain surface during surgery to record the activity of the cerebral cortex. The iEEG electrodes can be in strip. grid or depth configuration. Recording using the scalp EEG is a non intrusive method but results in a data with higher noise compared to iEEG.

Figure 1.2: Types of iEEG electrodes



(a) Strip electrodes (b) Grid electrodes (c) Depth electrodes

If by analyzing the EEG data, we could predict the onset of a seizure, it will be very helpful as an automated seizure-prevention mechanism. It would be possible to deliver anti-epileptic drugs or provide Deep Brain Stimulation (DBS) using such an EEG-triggered on-demand therapy device [6, 5, 13].

On the level of neuronal networks, focal seizures are assumed to be initiated by abnormally discharging neurons, so-called bursters [14] that recruit and engulf the neighboring neurons into a critical mass. This build-up might be mediated by an increasing synchronization of neuronal activity that is accompanied by a loss of inhibition, or by processes that facilitate seizures by lowering the threshold for excitation or synchronization. In this context, the term 'critical mass' might be misleading in the sense that it implies an increasing number of neurons that are entrained into an abnormal firing pattern. This mass phenomenon would be easily accessible to conventional EEG analysis, which, to date, has failed to detect it. Rather, the seizure-initiating process might better be visualized as a process by which an increasing number of critical interactions between neurons in a focal region and connected units in an abnormal functional network unfold over time. On the basis of these concepts, a number of studies have been carried out aiming to characterize this collective neuronal behavior from the gross EEG in order to allow definition of a transitional pre-ictal phase [9].

Recent studies have also shown that channels in more remote regions of brain also carry useful information to make such predictions which is contrary to earlier beliefs. These findings would support the notion of an epileptic network whose interactions extend over large regions of the brain rather than the concept of a localized and well-defined epileptic focus.

While a number of studies employ Time domain, Frequency domain, Wavelet domain, Principal Component Analy-

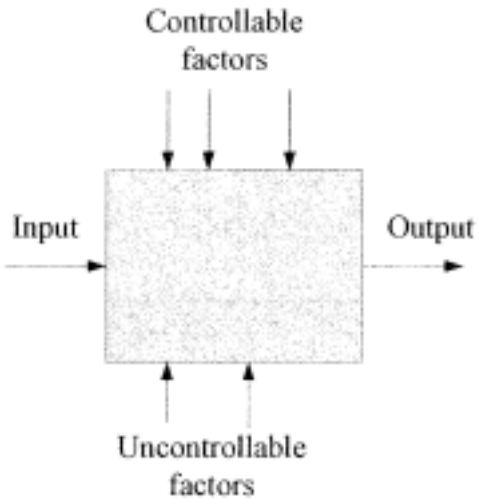
sis and Independent Component Analysis domain, Empirical Mode Decomposition and Singular Value Decomposition techniques to detect pre-ictal EEG recording have been published in the past, their result were not always reproducible [1]. This will be the focus of our study, to use a subset of these techniques to compare the performance of Random Forest, k-nearest neighbors and Support Vector Machine (SVM) classification to differentiate pre-ictal and interictal recordings and hence make predictions about seizure onset. Our method employs offline binary classification over pre-ictal and inter-ictal samples and the classification allows for a prediction within a 1 hour horizon.

2 Methodology

To study the effectiveness of machine learning techniques on any data set, a series of standard procedures are followed. These involve data preparation, feature extraction and feature selection, preparing train and test data, training of models and testing the results [2].

The problem under study is that of a binary classification. In a binary classification problem, a label (class) is assigned to a set of values (features). The data set provides large number of samples belonging to each class and a classifier is trained and tested using these samples. The goal is to correctly classify any new sample whose label is unknown.

Figure 2.1: Machine Learning Experiment Design



A binary classifier can be considered as a back box. During the training phase, a set of inputs are provided that assist in setting up the weights so as to minimize the error. Once trained, the model is now tested using a new set of inputs. These are the inputs that the model has not seen during the training phase and must assign a probability to the class to which the inputs belong to. Once the probability is assigned, then based on the thresholds, one can compute some very useful metrics that will help define the efficiency of the classifier. All these steps will be discussed in a more formal and detailed manner in the upcoming sections.

There are numerous algorithms that perform the task of binary classification. Logistic Regression, Support Vector Machines (SVM), k-Nearest Neighbors, Random Forest, AdaBoost, Artificial Neural Networks are few of the well known algorithms. In our study, we will focus on SVM, k-NN and Random forest and try to identify which one is the

best for this use case.

The data has been acquired from Kaggle competition organized by American Epilepsy society, University of Pennsylvania and Mayo Clinic during the period of 25th August 2014 to 17th November 2014
[https://www.kaggle.com/c/seizure-prediction]. The data is the long term iEEG recording of dogs with naturally occurring Epilepsy [7]. The implantable telemetry unit recorded voltages over 16 channels at 400 Hz sampling rate. The data has also been annotated and publicly accessible through the NIH-sponsored International Epilepsy Electrophysiology portal [http://ieeg.org].

The data consists of 10 minute EEG recordings that have been labeled by experts as pre-ictal or inter-ictal clip.

Table 2.1: Distribution of samples among different subjects

Subject	Number Of Channels	Number Of Interictal Records (10 Min)	Number Of Preictal Records (10 Min)	Total Records	Share of preictal samples
Dog 1	16	480	24	504	4.76%
Dog 2	16	500	42	542	7.75%
Dog 3	16	1440	72	1512	4.76%
Dog 4	16	804	97	901	10.77%
Dog 5	15	450	30	480	7.35%
Total		3766	301	4067	7.40%

2.1 Feature Extraction

A feature is quantifiable value associated to the phenomenon under observation. The features must be selected such that they are independent, uncorrelated and presents new information about the observation. For the purpose of this study we will be measuring important quantities from the raw EEG time-series and generate a feature vector that will be used for training and making predictions by the learning model.

What kind of features can be extracted from EEG time-series data and how does one determine them? Feature extraction from EEG time-series data has been a widely studied topic in statistics, digital signal processing and Neuroscience. Many journal articles were referred to in order to decide on what features are promising and how they can be computed. Computing some of these was computationally expensive and hence only a subset of features available have been considered for this experiment. PyEEG module [3] has been used for computation of these features.

An in-exhaustive list of the features that can be used is given below [1]. For this study we consider the statistical, frequency-domain, Hjorth parameters and features only.

Features for EEG time-series data:

- Variance
- Skewness
- Kurtosis
- Zero-crossing rate
- Eigen spectral features
- IMF-VoE
- largest Lyapunov exponent
- correlation dimension
- fractal dimension
- Shannon's entropy
- Time-domain energies
- Hurst exponent
- Cross-frequency coupling
- Wavelet coherence
- Phase entropy 1 and 2
- sample entropy
- approximate entropy
- Spike rate
- AR coefficients
- Energies of sub-bands

Throughout the discussion, we will follow the following notation:

F_k^i denotes feature F in channel i of data sample k.

2.1.1 Statistical Features

Min, Max, Mean and Variance The statistical features car computed for a given 10 minute EEG clip. These include the minimum value (min_k^i), maximum value (max_k^i), mean ($mean_k^i$), variance ($variance_k^i$), skewness ($skewness_k^i$) and kurtosis ($kurtosis_k^i$).

The minimum and the maximum voltages recorded during the 10 minute duration are represented by min_k^i and max_k^i respectively.

Mean and Variance in the voltage values throughout the 10 minute period is represented by $mean_k^i$ and $variance_k^i$ respectively.

$$mean = \frac{\sum y}{n}$$

where n is the number of data-points and y is the voltage measured

$$variance = \frac{\sum (y_i - \mu)^2}{n - 1}$$

Skewness Skewness is a measure of symmetry or non-symmetry in the data distribution . Here we compute the skewness of the voltage distribution in the 10 minute clip and use it as a feature. Skewness is a measure of the deviation of the sample distribution from that of a normal distribution. If skewness is positive, the data are positively skewed or skewed right, meaning that the right tail of the distribution is longer than the left. If skewness is negative, the data are negatively skewed or skewed left, meaning that the left tail is longer.

Skewness is computed by

$$kurtosis = \frac{\left[\frac{\sum(x-\bar{x})^3}{n}\right]}{\left[\frac{\sum(x-\bar{x})^2}{n}\right]^{3/2}}$$

Kurtosis The height and sharpness of the central peak relative to the rest of the data is measured by kurtosis . Here we compute the kurtosis of the voltage distribution in the 10 minute clip and use it as a feature.

Kurtosis is computed by

$$kurtosis = \frac{\left[\frac{\sum(x-\bar{x})^4}{n}\right]}{\left[\frac{\sum(x-\bar{x})^2}{n}\right]^{2}}$$

2.1.2 Energies of frequency sub-bands

The brain generates rhythms because of synchronous discharge or multiple neuron together. Such rhythms have been useful for clinicians and experts to identify not only seizure activity but also study sleep cycles, level of alertness and concentration and in brain computer interfacing. Experts have classified these rhythms based on the frequency at which they oscillate.

The commonly used bands are Delta, Theta, Alpha, Beta and Gamma.

Delta Band (0.5 - 4 Hz) These are the high amplitude and low frequency neural oscillations. The commonly associated feeling states with delta band frequencies are deep dreamless sleep, non-REM sleep, trance and unconscious. The behavior that corresponds to this is lethargic, not moving, not attentive.

Theta band (4 - 8 Hz) The commonly associated feeling states with theta band frequencies are intuitive, creative, recall, fantasy, imagery, creative, dreamlike, switching thoughts, drowsy. The behavior that corresponds to

this is creative, intuitive; but may also be distracted, unfocused.

Alpha band (8 - 13 Hz) The commonly associated feeling states with alpha band frequencies are relaxed, not agitated, but not drowsy. The behavior that corresponds to this is meditation, no action.

Beta band (13 - 30 Hz) The commonly associated feeling states with beta band frequencies are active, busy, or anxious thinking and active concentration. The behavior that corresponds to alert mental activity like problem solving.

Gamma band (30 Hz and above) The commonly associated feeling states with beta band frequencies are thinking, integrated thought. The behavior that corresponds to high-level information processing.

The power in a given frequency band can be computed using Fast Fourier Transform on the raw time-series data. This function computes the one-dimensional n-point discrete Fourier Transform on the real values and then aggregates the binned values to compute the power ratio of the given frequency bin the the signal. The power ratio helps normalize the distribution and provides a better estimate of changes in power over different frequency bands.

2.1.3 Hjorth Parameters

Hjorth parameters are the indicators of statistical properties of time-series signals introduced by Bo Hjorth. The parameters are Activity, Mobility and Complexity.

Hjorth Activity The activity parameter represents the signal power, the variance of a time function. This can indicate the surface of power spectrum in the frequency domain. This is represented by the following equation:

$$Activity(\, y(t)\,) = variance(\, y(t)\,)$$

Since the variance has already been accounted for, Hjorth Activity is not taken as a feature.

Hjorth Mobility The mobility parameter represents the mean frequency, or the proportion of standard deviation of the power spectrum. For a given time-series *y(t)*, mobility is defined as

$$Mobility(\, y(t)\,) = \sqrt{\frac{variance(\frac{dy}{dt})}{variance(y(t))}}$$

Hjorth Complexity The Complexity parameter represents the change in frequency. The parameter compares the signal's similarity to a pure sine wave, where the value converges to 1 if the signal is more similar. For a given time-series *y(t)*, mobility is defined as

$$Complexity(\, y(t)\,) = \sqrt{\frac{mobility(\frac{dy}{dt})}{mobility(y(t))}}$$

2.1.4 Detrended Fluctuation Analysis

Detrended Fluctuation Analysis was introduced by Peng et al[11] in 1994 and has since been used multiple times in analysis of EEG and Electrocardiogram (ECG) signals.The method of detrended fluctuation analysis has proven useful in revealing the extent of long-range correlations in time series. Briefly, the time series to be analyzed (with N samples) is first integrated. Next, the integrated time series is divided into boxes of equal length, n. In each box of length n, a least squares line is fit to the data (representing the trend in that box). The y coordinate of the straight line segments is denoted by $y_n(k)$. Next, we detrend the integrated time series, $y(k)$, by subtracting the local trend, $y_n(k)$, in each box. The root-mean-square fluctuation of this integrated and detrended time series is calculated by

$$F(n) = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[\, y(k) - y_n(k)\,]^2}$$

This computation is repeated over all time scales (box sizes) to characterize the relationship between F(n), the average fluctuation, and the box size, n. Typically, F(n) will increase with box size. A linear relationship on a log-log plot indicates the presence of power law (fractal) scaling. Under such conditions, the fluctuations can be characterized by a scaling exponent, the slope of the line relating log F(n) to log n.

2.2 Feature Matrix Encoding

For features to be presented to a machine learning model, they need to be encoded into a vector. In order to concentrate the features obtained at different channels of the same EEG clip, we use to following encoding scheme :

F_i^kdenotes the feature F in i^{th} channel of k^{th} EEG clip

We have already computed multiple such features and now we arrange these in a vector format

$$\begin{bmatrix} \begin{bmatrix} A_1^{k_1} & B_1^{k_1} & \dots \end{bmatrix}_{1\,\mathrm{x}\,m} \\ \begin{bmatrix} A_2^{k_1} & B_2^{k_1} & \dots \end{bmatrix}_{1\,\mathrm{x}\,m} \\ \vdots \\ \begin{bmatrix} A_{15}^{k_1} & B_{15}^{k_1} & \dots \end{bmatrix}_{1\,\mathrm{x}\,m} \\ \begin{bmatrix} A_{16}^{k_1} & B_{16}^{k_1} \end{bmatrix}_{1\,\mathrm{x}\,m} \\ \vdots \\ \begin{bmatrix} A_{16}^{k_n} & B_{16}^{k_n} & \dots \end{bmatrix}_{1\,\mathrm{x}\,m} \end{bmatrix}_{(16*n)\,\mathrm{x}\,1} \Rightarrow \begin{bmatrix} \begin{bmatrix} A_1^{k_1} & B_1^{k_1} & \dots & A_2^{k_1} & B_2^{k_1} & \dots & A_{15}^{k_1}B_{15}^{k_1} & \dots & A_{16}^{k_1}B_{16}^{k_1} \end{bmatrix}_{1\,\mathrm{x}\,(16*m)} \\ \vdots \\ \begin{bmatrix} A_1^{k_n} & B_1^{k_n} & \dots & A_2^{k_n} & B_2^{k_n} & \dots & A_{15}^{k_n}B_{15}^{k_n} & \dots & A_{16}^{k_n}B_{16}^{k_n} \end{bmatrix}_{1\,\mathrm{x}\,(16*m)} \end{bmatrix}_{n\,\mathrm{x}\,1}$$

where n is the number or 10 minute EEG recordings and m is the number of features. The number of channels have been assumed 16 which is true for all except Dog 5, which has 15 channels.

Algorithm 1: Encoding feature matrix

2.3 Generating training and testing data set

In any machine learning experiment, the integrity can be easily compromised if the testing is done using data samples that the learning model has already been exposed to. In order to make sure that this does not happen, we have divide the data set into training and testing data sets. The aim of doing so is that all the training and validation is performed using training data set. Once we have determined what feature-set, algorithm and hyper-parameters yield best result, only then is the final testing done using testing data set. This ensures sanctity of the data set and gives a better chance of selecting an unbiased learner.

In our study, we use the stratified shuffle split approach. We first split the feature vectors based on the label. Then both sets, interictal and preictal, are shuffled. Now we compute the index for each set that acts as a split dividing the entire set into two sets, one containing 70% and the other containing 30% data. We now merge them creating a 70-30% split ensuring that the class distribution is maintained in both training and testing set.

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix}$$ where x_k is the feature vector for the k$^{\text{th}}$ record.

STEP 1 : SEPARATE PREICTAL AND INTERICTAL FEATURE VECTORS

$$X_{preictal} = \{x_k \in \mathrm{X} \ : \ label\, of\, x_k\, is\, preictal\}$$

$$X_{interictal} = \{x_k \in X \ : \ label\, of\, x_k\, is\, interictal\}$$

↓

STEP 2 : SHUFFLE VECTORS

$$X_{preictal} \leftarrow Shuffle(X_{preictal})$$

$$X_{interictal} \leftarrow Shuffle(X_{interictal})$$

↓

STEP 3: GENERATE SPLIT

$$split\_index_{preictal} \leftarrow 0.7 * len(X_{preictal})$$

$$split\_index_{interictal} \leftarrow 0.7 * len(X_{interictal})$$

$$X\_train_{preictal} = X_{preictal}[0\, to\, split\_index_{precital}]$$

$$X\_train_{interictal} = X_{interictal}[0\, to\, split\_index_{intericital}]$$

$$X\_test_{preictal} = X_{preictal}[split\_index_{precital}\, to\, n]$$

$$X\_test_{interictal} = X_{interictal}[split\_index_{intericital}\, to\, n]$$

↓

STEP 4 : MERGING THE SPLITS

$$X\_train = X\_train_{preictal} \cup X\_train_{interictal}$$

$$X\_test = X\_test_{preictal} \cup X\_test_{interictal}$$

Algorithm 2: Method to generate training-testing data split

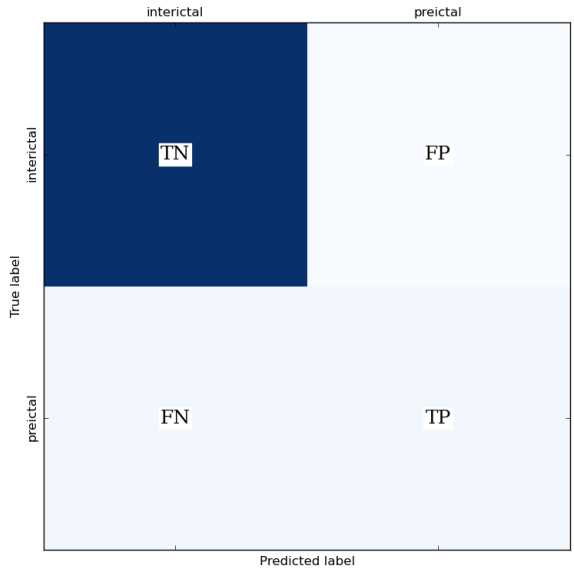
2.4 Training of Models

For training we use the models implemented using Scikit-Learn[10] library in Python. We have trained SVM, kNN and Random Forest classifiers on the data set using cross validation and hyper-parameter optimization and selected the best model for each subject to be used for evaluating final results. For evaluation, we use the trained model and expose it to the test set, plot the confusion matrices and evaluate the metric scores. In the following discussion we will discuss in detail about confusion matrix, scoring metrics and hyper-parameter optimization.

2.4.1 Confusion Matrix

A confusion matrix, also known as a contingency table or an error matrix, is a specific table layout that allows visualization of the performance of an algorithm. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. In binary classification we can generate a 2x2 matrix as shown in Figure 2.2 , and evaluate multiple scoring metrics to measure the performance of the classifier. *Note that for our study, the preictal samples are considered as the positive class and the interictal samples are considered as the negative class.*

Figure 2.2: Confusion Matrix



Numerous metrics can be computed using the confusion matrix but those that are of most significance for our

experiment and will be used to analyze the results are discussed bellow. We also need to define a few terms that will be used in the discussion. True Positive (TP) (or True Negative (TN)) is the number of samples that were predicted as positive (or negative) and did belong to the positive (or negative) class. On the other hand, False Positive (FP) (or False Negative (FN)) is the number of samples that were predicted as possitive (or negative) but actually belong to the negative (or positive) class.

Recall or Sensitivity or True Positive Rate Recall is the fraction of relevant samples predicted. In other words, it indicates that if a classifier has been trained for a given positive class, what fraction of positive test samples is it able to classify/predict. The score value lies between 0 (worst) and 1 (best).

$$recall = \frac{TP}{TP + FN}$$

For models predicting medical conditions that can be fatal or cause serious injuries to the patient, it is required that the classifier has a high recall score. If our classifier does not have a very high recall score, then it would imply that many of the seizure events were missed by the classifier and reported as normal brain activity. This score does not tell us anything about number of samples that were incorrectly classified as a positive class.

Precision or Positive Predictive Rate Precision is the fraction of positive predicted samples that are relevant. In other words, it indicates that if the classifier was trained for a given positive class, what fraction of samples classified as positive were actually positive. The score value lies between 0 (worst) and 1 (best).

$$precision = \frac{TP}{TP + FP}$$

A higher value of precision is desired. A high precision implies that if the classifier has classifier some sample as positive, then there is a high probability that it is correct. This score does not tell us anything about incorrectly classified positive classes.

Fall-out or False Positive Rate Fall-out is the fraction of negative samples that were incorrectly classified as positive. A high fall-out indicates that the classifier is frequently classifying negative samples as positive and hence leads to a reduced confidence in the warnings generated by classifier on encounter of a positive class sample. The score value lies between 0 (worst) and 1 (best).

$$false\,positive\,rate = \frac{FP}{FP + TN}$$

It is always desired to have a low false positive rate. In case of any mission critical system where the classifier generates warning, one can argue that it is okay to witness a few false alarms as long as the true positives are not being misclassified. Hence it is an added benefit to have a low false positive rate but not at the cost of recall or precision.

F_1Score F_1score is a measure of binary classifier's accuracy giving equal importance to precision and recall.It is a weighted average of precision and recall and lies in the range of 0 (worst) and 1 (best).

$$F_1\,score = \frac{2 * precision * recall}{precision + recall} = \frac{2 * TP}{(2 * TP) + FP + FN}$$

It is a special case of $F_\beta$score where a user attaches β times more importance to recall than precision.

$$F_\beta = (1 + \beta^2)\,\frac{precision * recall}{(\beta^2 * precision) + recall}$$

Matthews correlation coefficient It is a measure of the quality of a binary classifier. It was introduced by biochemist Brian W. Matthews in 1975 and since then has been considered as a good and balanced metric for binary classification even with a skewed class distribution. It takes into account both true and false positives and negatives.

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}}$$

In our experiment, the classes are highly skewed. The preictal samples for only 4.76% - 10.77% of the total samples. In such a distribution, metrics such as accuracy can be very misleading. Hence MCC will play a major part in the analysis.

Receiver Operating Characteristic (ROC) curve The ROC curve is the graphical representation of the performance of the classifier when the decision thresholds are varied. It is created by plotting True Positive rate (or recall or sensitivity) against False positive rate at various thresholds.

Figure 2.3: Representative ROC curve

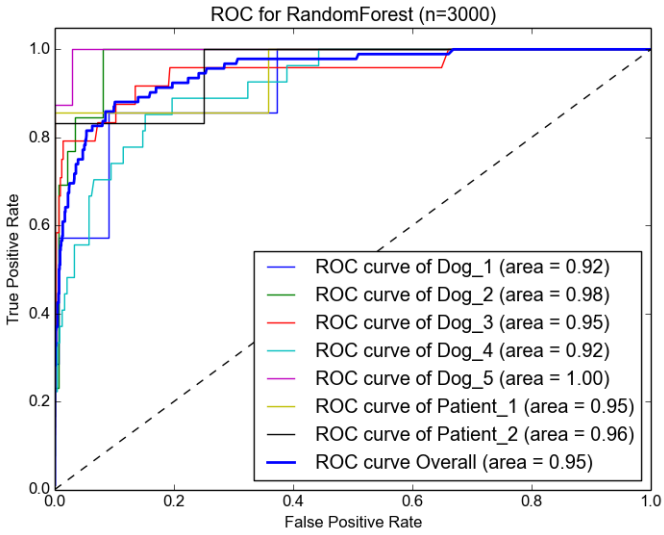
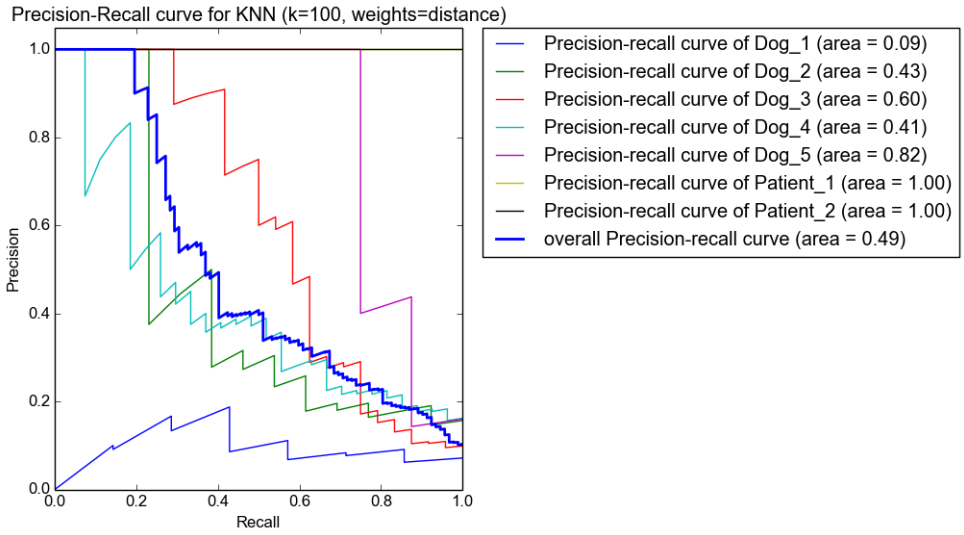


Figure 2.3 shows a sample ROC curve. Each point on the curve represents a threshold of decision making for the classifier. At each threshold the confusion matrix is generated and recall and False positive rate are computed and the point is plot. This gives an indication of the recall as a function of the false positive rate. In ideal scenario, the curve will touch the top left corner and the area under the curve will be 1.

These curves can be misleading when the class distribution is very skewed. In our experiment this is true and hence we will not put a lot of emphasis on ROC curves when analyzing the results.

Precision-Recall curve The PR curve is a graphical representation of the changes in precision and recall values as decision thresholds are changed. Unlike ROC curve that has a monotonically increasing nature, the PR curve has no monotonic properties. Figure 2.4 shows a sample PR curve. Notice the incresing-decreasing nature of the curve. The area under the PR curve is used as a metric for performance measurement. This curve is useful in cases where the class distribution is skewed and ROC is not very useful. The ideal curve will move towards the top-right corner and have an area under the curve of 1. It can be shown that a model that dominates in ROC curve will also dominate in the PR curve but the relationship is not straightforward if the curved intersect [12].

Figure 2.4: Representative Precision-Recall curve



2.4.2 Stratified k-fold Cross-validation

Stratified k-fold cross validation is a method in which the data set is split into k parts, each part itself having the similar class distribution as that of the complete data set. The model is then trained on k-1 splits and validated against the remaining one split. This process is repeated k times, each time using a different split for validation. This process reduces the bias in measuring the metric score and provides better estimation of the performance of the classifier even with a small number of samples.

In our experiment, we employ a 5-fold cross validation for all the configurations of the classifiers used during hyper-parameter optimization. We then take the mean of the scores and use it to compare and decide the best classifier.

2.4.3 Hyper-parameter optimization

Any machine learning algorithm is a function that takes in a few arguments as configuration parameters. Selecting these hyper-parameters can be a daunting task as they differ with different data sets. A number of techniques have been developed to zero-in to the best configuration. One of them is to start with one set of hyper-parameters and keep changing them one at a time in the direction where the score is improving. This is an intuitive but time consuming method.

A more exhaustive way is to run a Grid Search. In Grid search, we define the values that the hyper-parameters

can take. The task then can be run in parallel to compute the scores for each of the possible combinations. Once complete, the scores are compared and the optimal configuration can be used. For out study, we have employed Grid Search.

3 Results and Conclusion

We perform the computation on raw time series data to extract Statistical features, frequency sub-band energy values, Hjorth parameters and DFA scores as described in Section 2.1. Calculation of these features is computationally expensive and the programs were written to perform these computations in parallel to completely utilize to Kosambi cluster. Once computed, they were encoded into the feature vector as described in Section 2.2. This allowed the classifier to view the multi-channel record as a single vector.

The encoded feature vector was then split into the training and test set. For shuffling of the vectors, we used a pseudo-random number generator with a fixed seed. This ensured reproducibility of results. As described in Section 2.3, the data set was split in 7:3 ratio keeping class distribution constant. Hence the test results were not affected by the difference in the number of samples of each class present and give accurate reflection of the original data set.

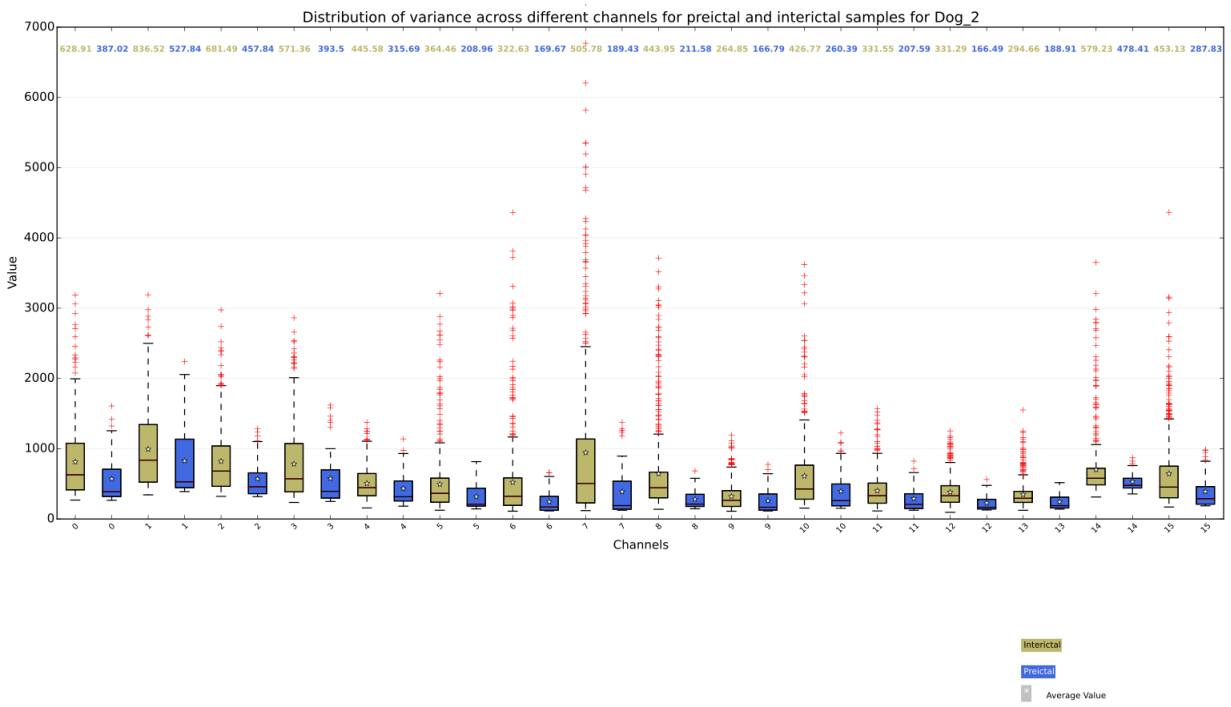
The training phase involved model selection using Grid Search and 5-fold stratified cross validation technique. Cross validation was necessary for evaluating the model because of the small number of samples present in the data set. It reduced the bias towards the folds generated and gave a more accurate estimate of the optimal model.

Grid search was used to select the optimal model. The metric that was used for optimizing the models was recall. In seizure forecasting, it is important that no seizure event is missed and hence high recall value is of great importance. Having performed the search, we evaluated the mean cross-validation scores of each model and the one with the highest was chosen to be tested.

3.1 Analysis of data distribution

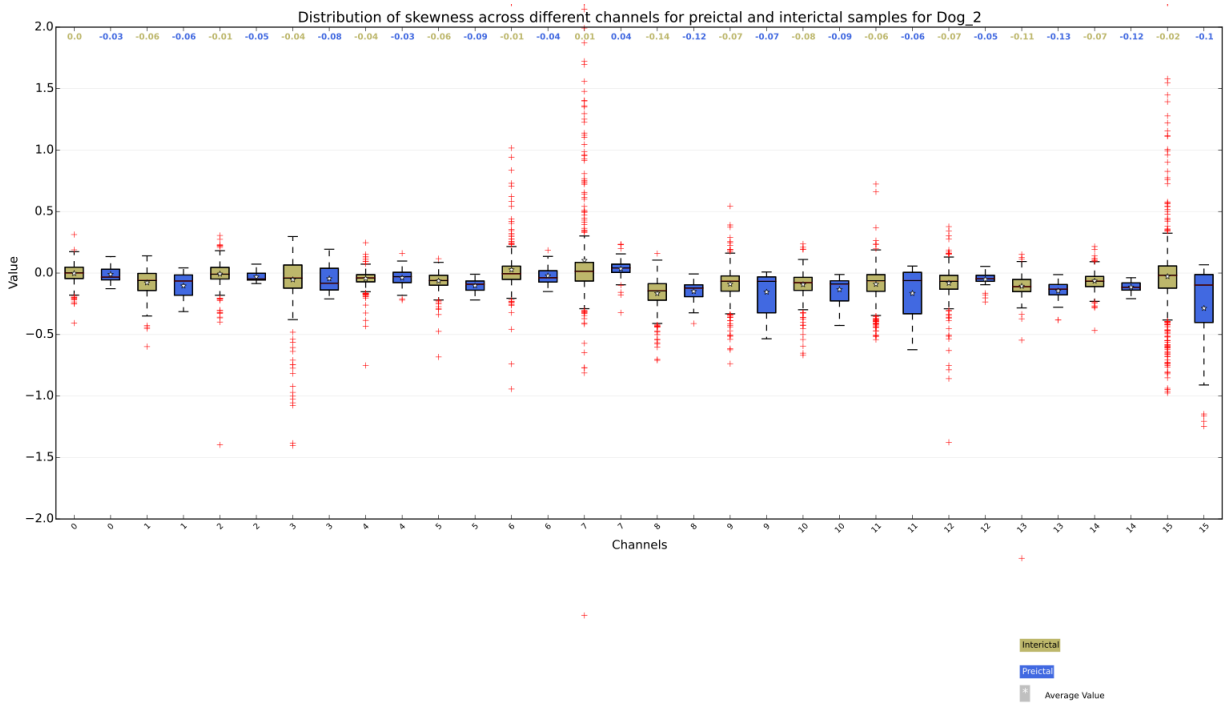
The hypothesis that brain activity before a seizure is significantly different from that of normal brain activity can be shown by viewing the distribution of feature values across different samples and channels. The features show similar trend across all channels for the two different classes. This distinction is much more pronounced in few features compared to the others.

Figure 3.1: The distribution of variance measured in recordings for Dog 2



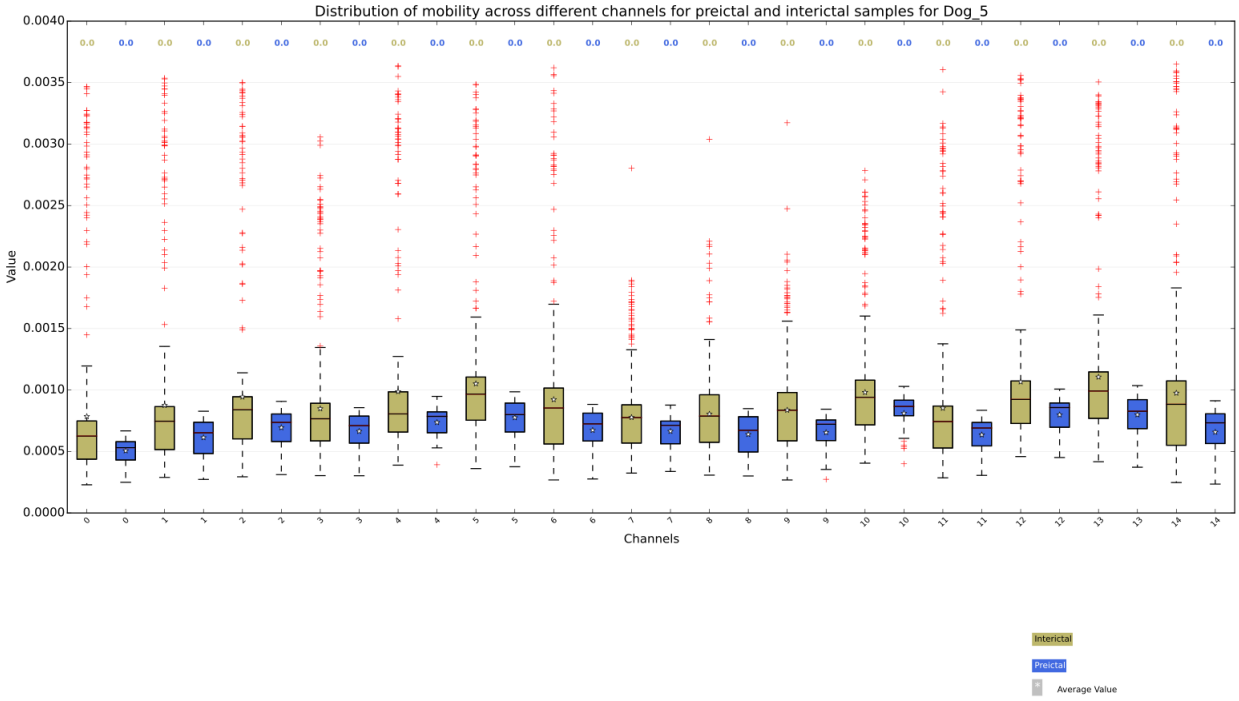
In Figure 3.1, we see the spread of the variance of voltage in a 10 minute clip for all the samples across all the channels. It can be clearly observed that the samples in the interictal clips demonstrate a higher degree of variance and outliers compared to those in the preictal samples. The median and the average variance also tends to be higher in the interictal clips. This behavior is consistent across all channels. The possible explanation for this is that during normal brain functioning, the brain is a very noisy dynamic system where all components are generating electrical activity irrespective of the other regions of the brain. However during a seizure event, the neurons fire synchronously and hence generate a more consistent electrical activity pattern.

Figure 3.2: The distribution of skewness measured in recordings for Dog 2



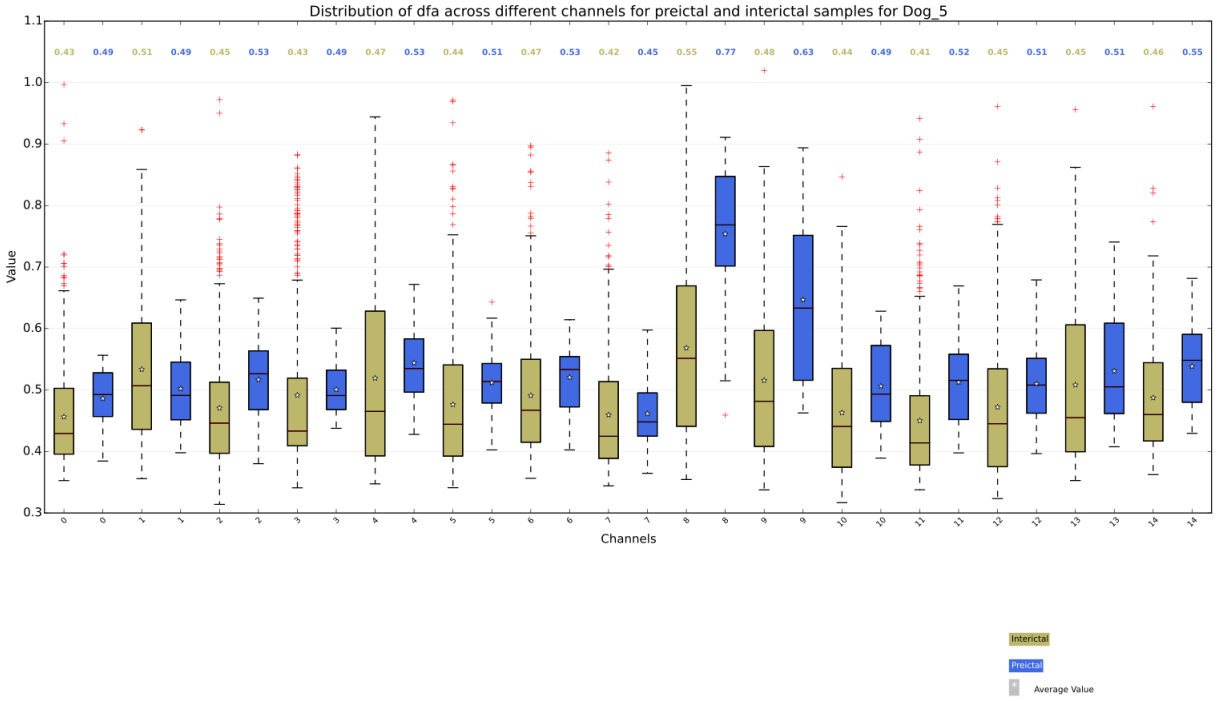
In Figure 3.2, we observer that the spread of skewness across different samples is larger in case of interictal. The average and median measure of skewness for any given channel is also higher for the interictal samples. The outliers are again more prominent in the case of interictal samples. Such distinctions provide better indications in separating the two types of samples.

Figure 3.3: The distribution of Hjorth Mobility measured in recordings for Dog 2



In Figure 3.3 , we see the distribution of the Hjorth mobility across different channels for Dog 5. A stark difference in the distribution can be observed with most of the samples in all channels being regarded as outliers in the interictal samples. Th average and median mobility for a given channel is also higher for interictal records compared to preictal records.

Figure 3.4: The distribution of DFA measured in recordings for Dog 5



In Figure 3.4, we observer a higher spread of value in interictal samples compared to that of preictal. There is also a noticible trend here that preictal samples have a higher median and average value than that of the interictal samples. Such trends allow the classifiers to generate much more accurate decision boundaries that are useful in the classification.

A complete list of distribution of features for all the subjects can be seen in Appendix A.

3.2 Model Selection

The model selection was performed using the Grid search on hyper-parameters as explained in Section 2.4.3 and the model with highest mean recall score for the 5-fold stratified cross validation was selected.

For SVM, the hyper-parameters are the soft margin parameter C, the type of kernel. We traverse over 0.1, 0.5, 1, 10, 100, 1000 for C and use a linear and radial basis function (RBF) kernel.

For Random Forest, the hyper-parameter is the number of estimators. We traverse over 10, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1200, 1400, 1600, 1800, 2000, 2200, 2400, 2600, 2800, 3000, 3200, 3400, 3600, 3800, 4000, 4200, 4400, 4600, 4800, 5000, 5200, 5400, 5600, 5800 as values of number of estimators in search for the best classifier.

For kNN, the hyper-parameters are the number of neighbors and the weight metric. We use 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 as the number of neighbors and 'uniform' and 'distance' as weight metric.

The hyper-parameters for best models selected for each subject is shown in Table 3.1.

Table 3.1: Best classifier hyper-parameters for each subject

(a) SVM

Subject	C	degree	gamma	Kernel
Dog 1	0.1	3	0	linear
Dog 2	0.1	3	0	linear
Dog 3	0.1	3	0	linear
Dog 4	0.1	3	0	linear
Dog 5	0.1	3	0	linear

(b) kNN

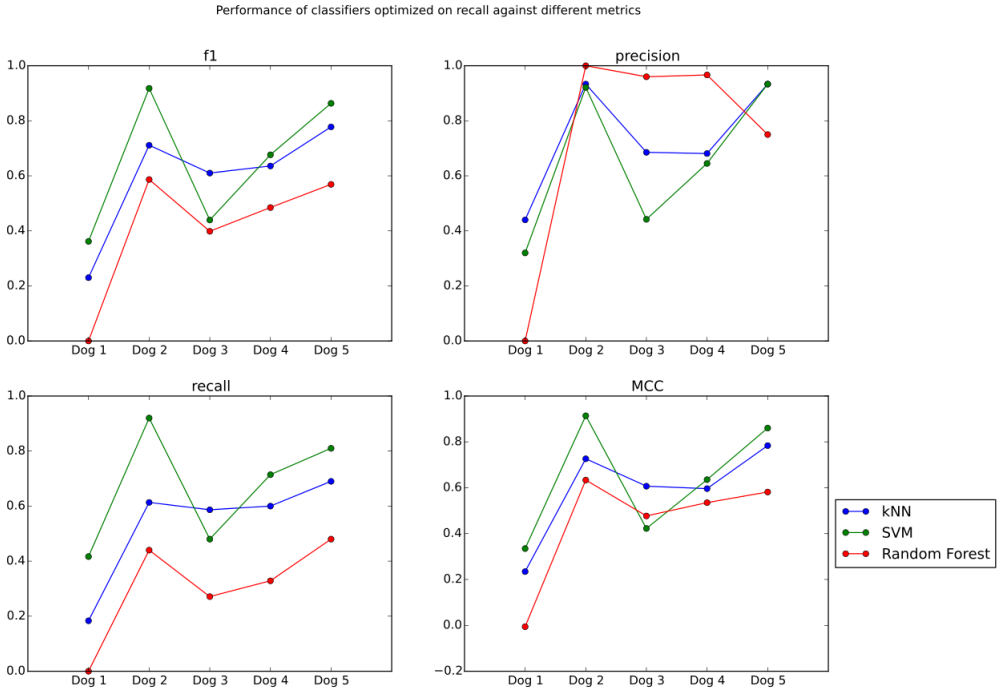
	Dog 1	Dog 2	Dog 3	Dog 4	Dog 5
neighbors	1	5	1	1	1
weight	uniform	distance	uniform	uniform	uniform

(c) Random Forest

	Dog 1	Dog 2	Dog 3	Dog 4	Dog 5
number of estimators	10	300	200	100	400

The comparison of training scores of the three classifiers can be seen in Figure 3.5. The scores are computed on the classifiers that have been optimized for recall and the hyper-parameters were computed using Grid Search. The higher F_1 and MCC scores of SVM and kNN suggests that they will perform better than Random Forest.

Figure 3.5: Cross-validation scores of optimal classifiers



3.3 Result on test data

Once the best classifier model was selected by cross-validation and hyper-parameter optimization, the models were then exposed to the test set. This is a set of samples that the classifier has not been trained or validated upon and will help in deciding the best classifier. For each sample, the classifiers return a probability score, indicating that the sample belongs to pre-ical stage. A high score indicates a higher confidence level and vice-versa. Based on these score and taking the decision threshold to be 0.5, we plot the confusion matrix for each subject as well as overall data set and analyze the results. Figure 3.6 shows the results for all the 3 classifiers on all the subjects combined. A more detailed view can be obtained by looking at the confusion matrix for each subject in Appendix B.

Since the class distribution is skewed, with preictal clips comprising only 4.76%-10.77% of the samples, the F_1score, the MCC score and the area under the PR curve (Appendix E) will play a major role is deciding the best classifier. The results have been optimized on recall as the scoring metric during the training phase because that is the most important metric to consider. Our aim is to correctly identify all the pre-ictal samples. The reduction in the false positive rate is desired but not at the cost of the recall.

Figure 3.6: Confusion Matrix for complete Test Data

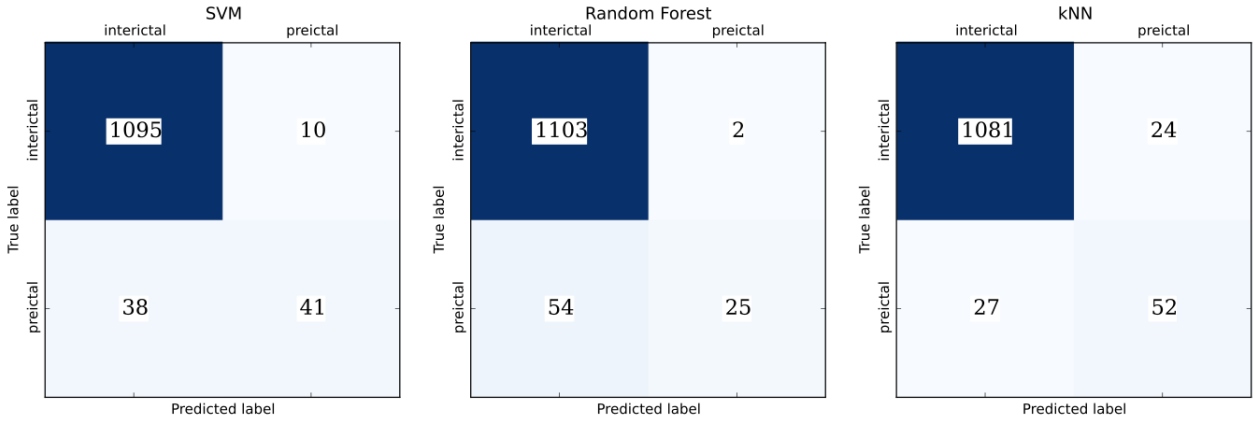


Table 3.2: Scores on test data at threshold=0.5

	Dog_1			Dog_2			Dog_3		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.95	0.95	0.93	0.94	1.00	0.94	0.97	0.95	0.97
F1	0.00	0.20	0.38	0.38	1.00	0.44	0.61	0.32	0.72
FPR	0.00	0.01	0.04	0.00	0.00	0.01	0.00	0.00	0.02
MCC	nan	0.19	0.34	0.47	1.00	0.47	0.64	0.37	0.70
Precission	0.00	0.33	0.33	1.00	1.00	0.80	0.92	0.71	0.69
Recall	0.00	0.14	0.43	0.23	1.00	0.31	0.46	0.21	0.75
	Dog_4			Dog_5			overall		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.92	0.94	0.94	0.98	0.99	0.99	0.95	0.96	0.96
F1	0.35	0.64	0.70	0.77	0.88	0.94	0.47	0.63	0.67
FPR	0.00	0.02	0.03	0.00	0.01	0.01	0.00	0.01	0.02
MCC	0.41	0.61	0.67	0.78	0.87	0.94	0.53	0.63	0.65
Precission	0.86	0.75	0.70	1.00	0.88	0.89	0.93	0.80	0.68
Recall	0.22	0.56	0.70	0.62	0.88	1.00	0.32	0.52	0.66

Table 3.2 shows the metric scores for individual subjects and overall data set. The Accuracy is of limited help and has been shown to strengthen the point that even bad classifiers can attain very high accuracy if the class distribution is highly skewed.

Random Forest shows a consistently low recall, MCC score and F_1score (except for Dog 3). This leads us to conclude that Random Forest is the least useful classifier in our arsenal. The other two, SVM and kNN have very good performance. kNN outperforms SVM in 4 out of the 5 subjects.

In terms of computation cost, kNN and Random Forest are both faster than SVM. This consideration is important

when we think of performing prediction on a potable device with limited resources in terms of memory and computation.

As shown in Appendix C , we have also analyzed the results at the decision thresholds of 0.2 and 0.8. Random Forest improves in performance at the threshold of 0.2 but it also reflects the dependence of its performance on thresholds. kNN on the other hand remain largely unaffected over this wide threshold range and is robust to such changes. It is an important characteristic to have in a classifier that is working with EEG data. EEG data is highly patient specific and very dynamic and noisy. If the performance of a classifier depends on very few and controllable factors which do not change with subject, it can make the optimization process easier.

Hence we recommend the use of kNN as it performs well across different metrics, is computationally less expensive and is robust to changes in decision thresholds. The better than chance performance of the classifiers is also indicative of the idea that seizure activities indeed start to show signs much before the actual seizure occurs. This provides great hope to the patients and researchers to improve their methods in order to help the patients.

4 Discussion and Future Work

Our study aimed at identifying the performance of seizure prediction using the three classifiers (SVM, kNN and Random Forest) on the same data set of 5 dogs with naturally occurring epilepsy. Earlier studies had focused on the identification of new features and presenting their results using a single classifier. As can be seen in the results presented in Section 3.3, even with rudimentary feature selection, the performance of the prediction system may vary significantly based on the algorithm used. There is strong evidence that seizure detection is possible by monitoring EEG signals and that has been shown by the classifiers performing better than chance predictions.

Metric selection plays an important role in analysis of results. One thing that is clear for this problem is the skewed distribution of classes. The seizure activities are few and even fewer in the time window when the intracranial EEG recording is performed. Most of the results published work on a single classification model and present the optimized recall and false positive rate obtained. There is a need to have a single metric that can be used as direct comparison for the performance of the classifiers. In our opinion, F_1 score and MCC provide a promising start but further study is necessary to define a metric that can be used across different data sets.

Another important consideration that needs to be addressed is the absence of a common data sets. It was found in literature survey that researchers have used records from The European Epilepsy Database, iEEG Portal and self monitored sources. In studies relating to highly dynamic systems like brain, where one subject's condition cannot be compared easily with another subject, it becomes impossible to compare the different prediction methods across different data sets. Further, the iEEG recordings of seizure patients is rare and scarce. This leads to another problem of too little data to work on. A unified data set that all the researchers can refer to, to present their results will be of much more utility. This data set can be updated periodically keeping the format consistent so that the problem of over-fitting the algorithm for a particular data set is avoided.

For this study we have used the three basic machine learning algorithms that are most widely used. Our next step will be to explore the performance of Artificial Neural Networks and Deep Learning Algorithms for the same task. These algorithms have already shown ground-breaking results in various fields. The prime motivation to explore these methods is their ability to extract hidden details about features from the raw data. This can be useful in exploring new relations in features that were earlier unknown and then can be used with classifiers that are computationally less expensive. Another factor to consider is the development of online prediction system. The models that we generated were trained for offline predictions. This is not something that is feasible in real-time situation. Hence there is a need for detailed study of the features and algorithms that can be used in embedded systems to develop a real-time

portable warning system.

We will continue with our work on this problem and are developing standards that will prove vital to the research community for faster and better development of algorithms. Going further on this path, we will develop a portable real-time system that can warn the patient and the caregivers about a seizure onset so that anti-epileptic drugs or Deep Brain Stimulation can be administered.

References

[1] Turkey N. Alotaiby, Saleh A. Alshebeili, Tariq Alshawi, Ishtiaq Ahmad, and Fathi E. Abd El-Samie. Eeg seizure detection and prediction algorithms: a survey. *EURASIP J. Adv. Sig. Proc.*, 2014:183, 2014.

[2] Ethem Alpaydin. *Introduction to Machine Learning.* The MIT Press, 2nd edition, 2010.

[3] Forrest Sheng Bao, Xin Liu, and Christina Zhang. Pyeeg: An open source python module for eeg/meg feature extraction. *Comp. Int. and Neurosc.*, 2011, 2011.

[4] M. Bikson, J. Lian, P. J. Hahn, W. C. Stacey, C. Sciortino, and D. M. Durand. Suppression of epileptiform activity by high frequency sinusoidal fields in rat hippocampal slices. *J. Physiol. (Lond.)*, 531(Pt 1):181–191, Feb 2001.

[5] R. S. Fisher. Therapeutic devices for epilepsy. *Ann. Neurol.*, 71(2):157–168, Feb 2012.

[6] R. S. Fisher. Deep brain stimulation for epilepsy. *Handb Clin Neurol*, 116:217–234, 2013.

[7] J. Jeffry Howbert, Edward E. Patterson, S. Matt Stead, Ben Brinkmann, Vincent Vasoli, Daniel Crepeau, Charles H. Vite, Beverly Sturges, Vanessa Ruedebusch, Jaideep Mavoori, Kent Leyde, W. Douglas Sheffield, Brian Litt, and Gregory A. Worrell. Forecasting seizures in dogs with naturally occurring epilepsy. *PLoS ONE*, 9(1):e81920, 01 2014.

[8] Patrick Kwan and Martin J. Brodie. Early identification of refractory epilepsy. *New England Journal of Medicine*, 342(5):314–319, 2000. PMID: 10660394.

[9] Florian Mormann, Ralph G. Andrzejak, Christian E. Elger, and Klaus Lehnertz. Seizure prediction: the long and winding road. *Brain*, 130(2):314–333, February 2007.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[11] C.-K. Peng, S. V. Buldyrev, S. Havlin, M. Simons, H. E. Stanley, and A. L. Goldberger. Mosaic organization of dna nucleotides. *Phys. Rev. E*, 49:1685–1689, Feb 1994.

[12] D.M.W. Powers. Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[13] W. H. Theodore and R. Fisher. Brain stimulation for epilepsy. *Acta Neurochir. Suppl.*, 97(Pt 2):261–272, 2007.

[14] J Wellmer, H Su, H Beck, and Y Yaari. Long-lasting modification of intrinsic discharge properties in subicular neurons following status epilepticus. *European Journal of Neuroscience*, 16(2):259–266, 2002.

Part I

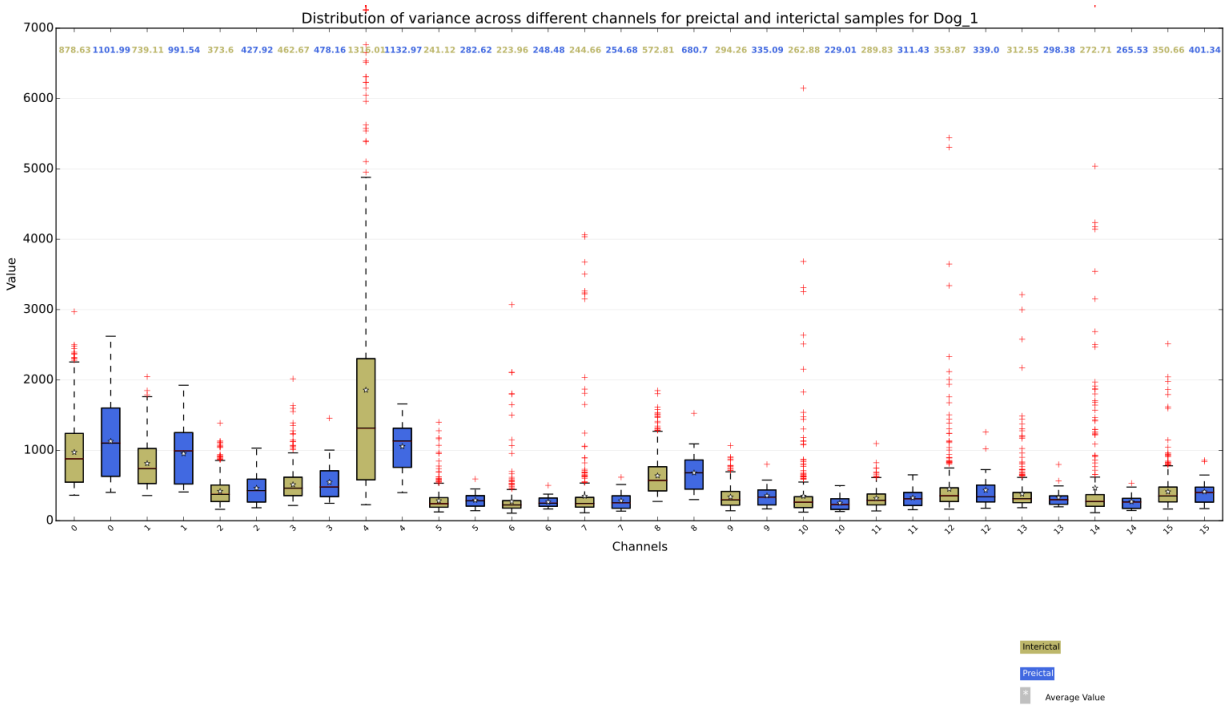
Appendix

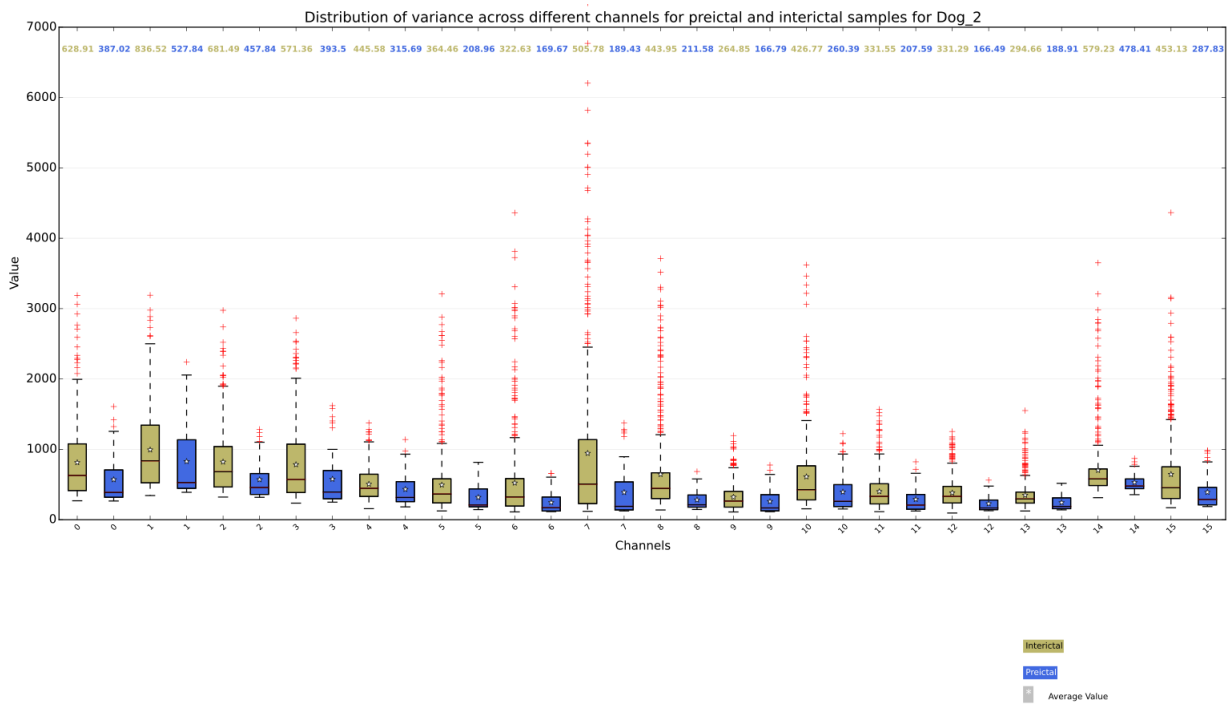
A Data distribution

A.1 Variance

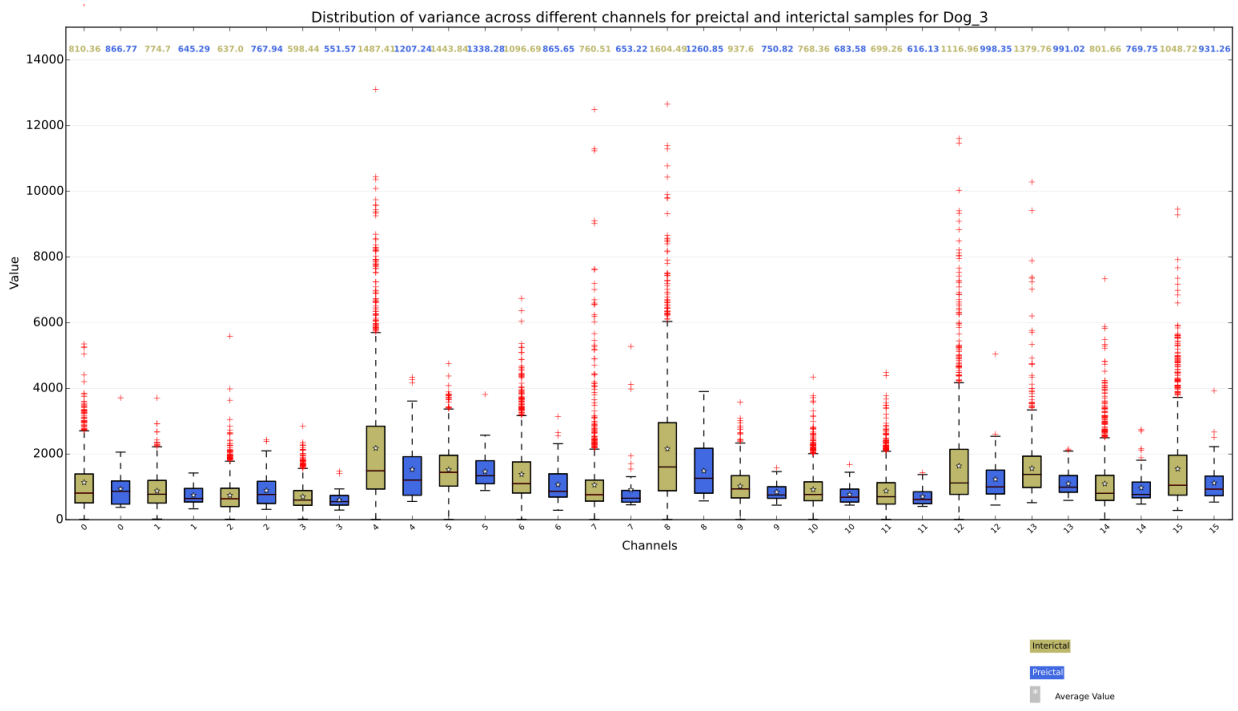
Figure A.1: Distribution of variance across different subjects

(a) Dog 1

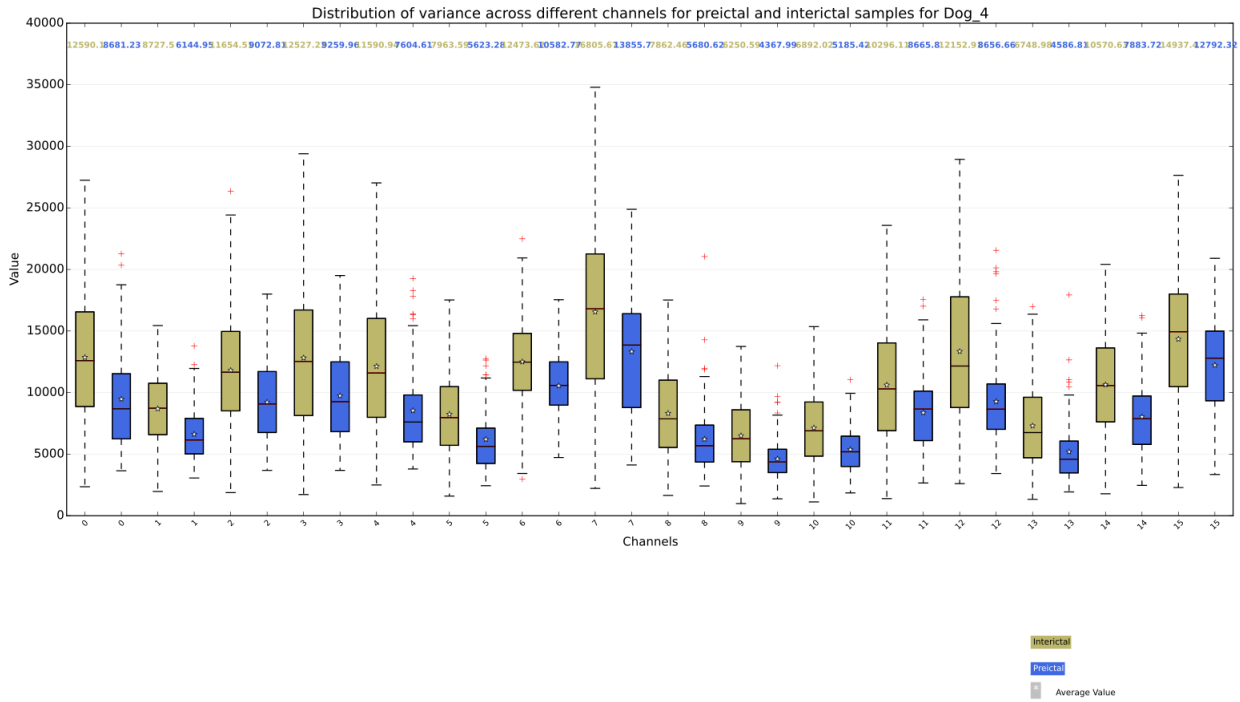



Distribution of variance across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

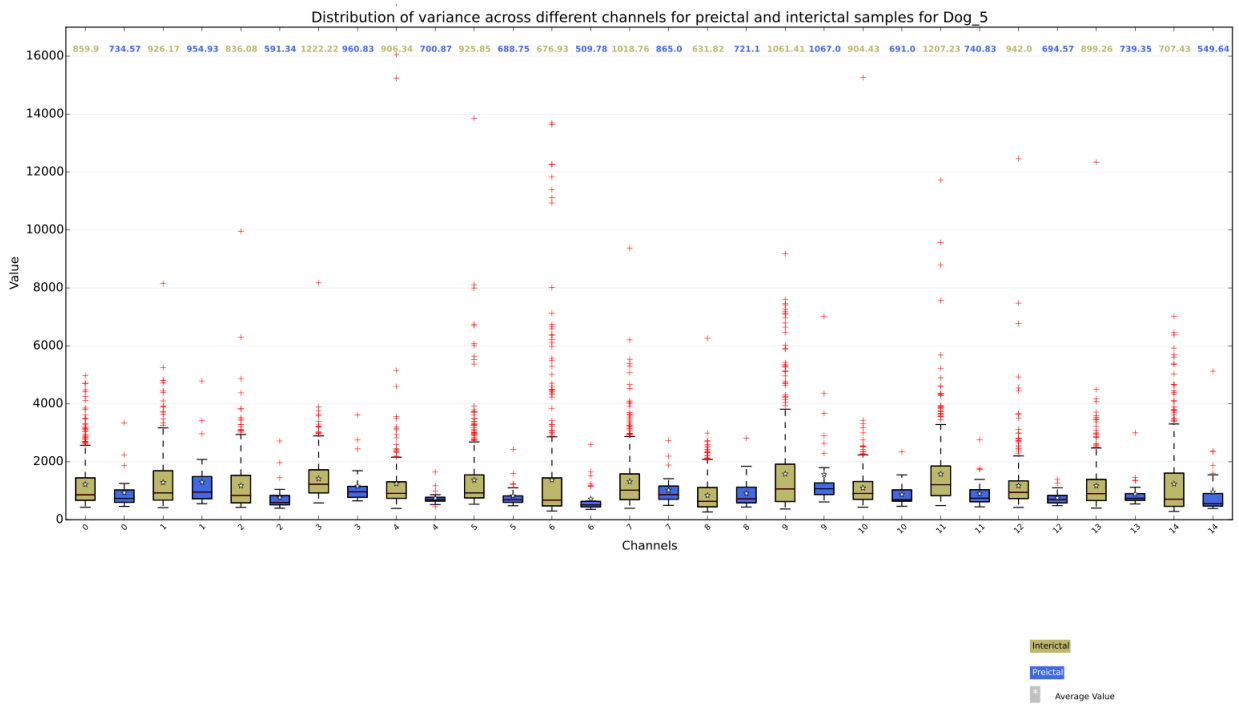
(b) Dog 2


Distribution of variance across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of variance across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

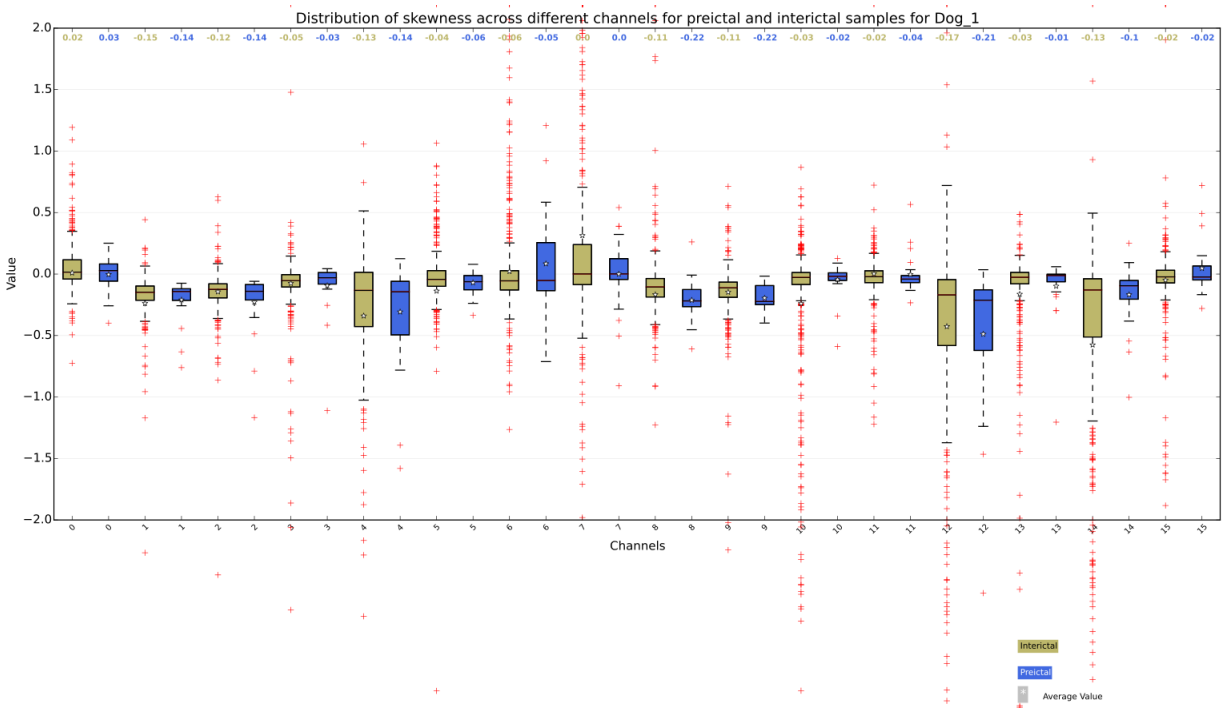

Distribution of variance across different channels for preictal and interictal samples for Dog_5
859.9 734.57 926.17 954.93 836.08 591.34 1222.22 960.83 906.34 700.87 925.85 688.75 676.93 509.78 1018.76 865.0 631.82 721.1 1061.41 1067.0 904.43 691.0 1207.23 740.83 942.0 694.57 899.26 739.35 707.43 549.64
Value
Channels
Interictal
Preictal
Average Value

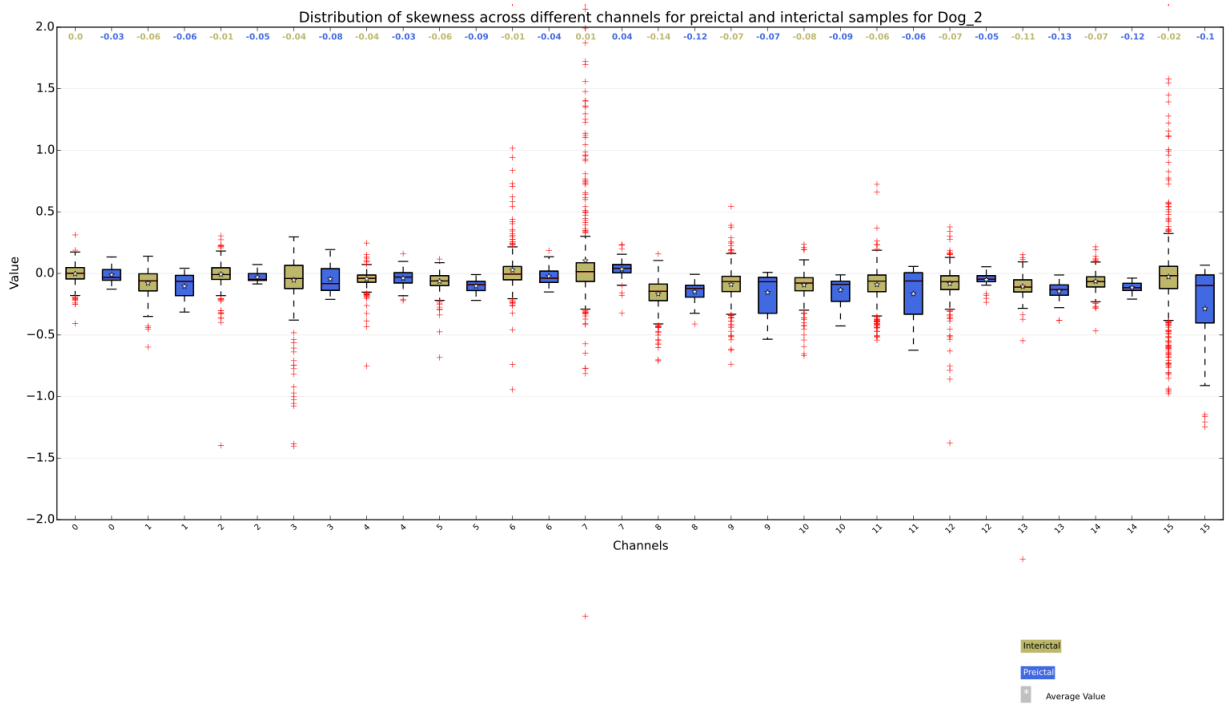
(e) Dog 5

A.2 Skewness

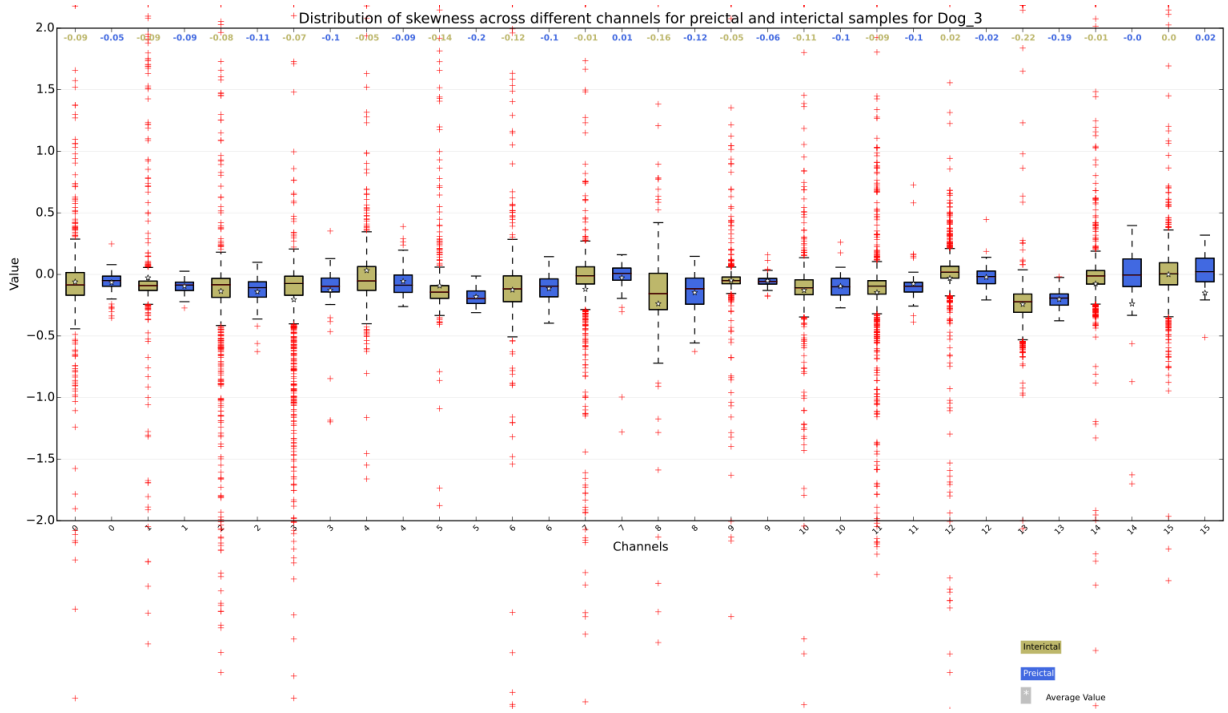
Figure A.2: Distribution of Skewness across different subjects

(a) Dog 1

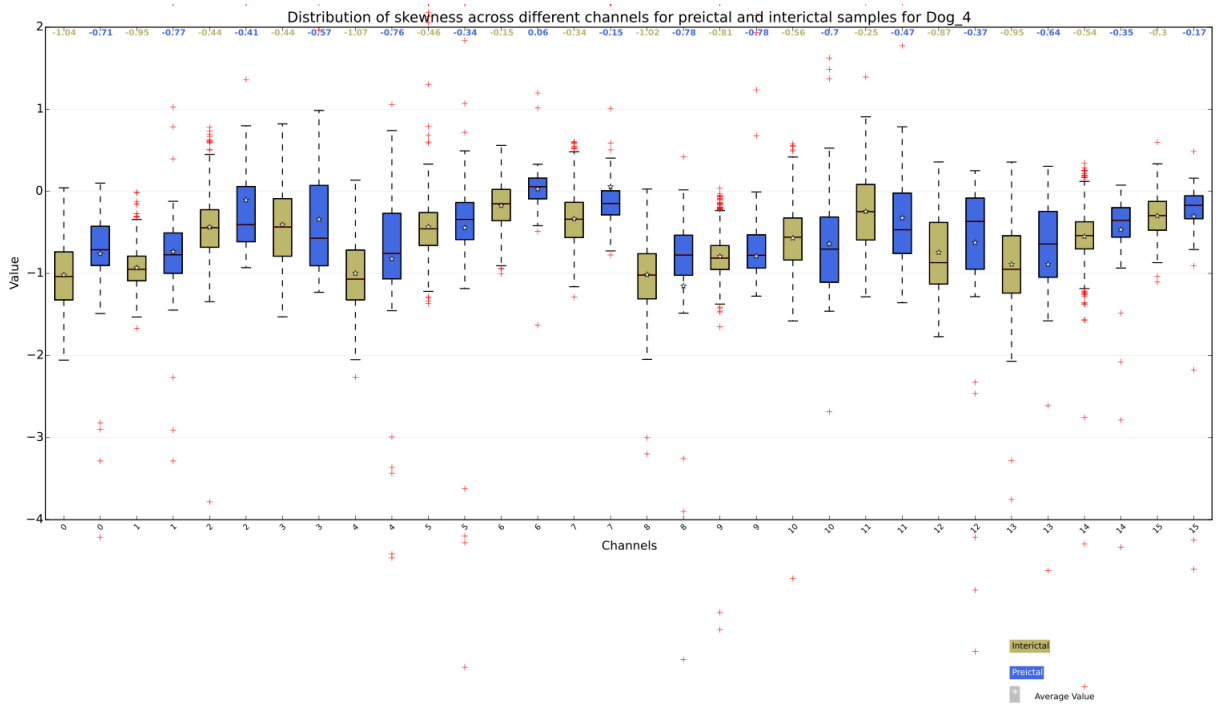



Distribution of skewness across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

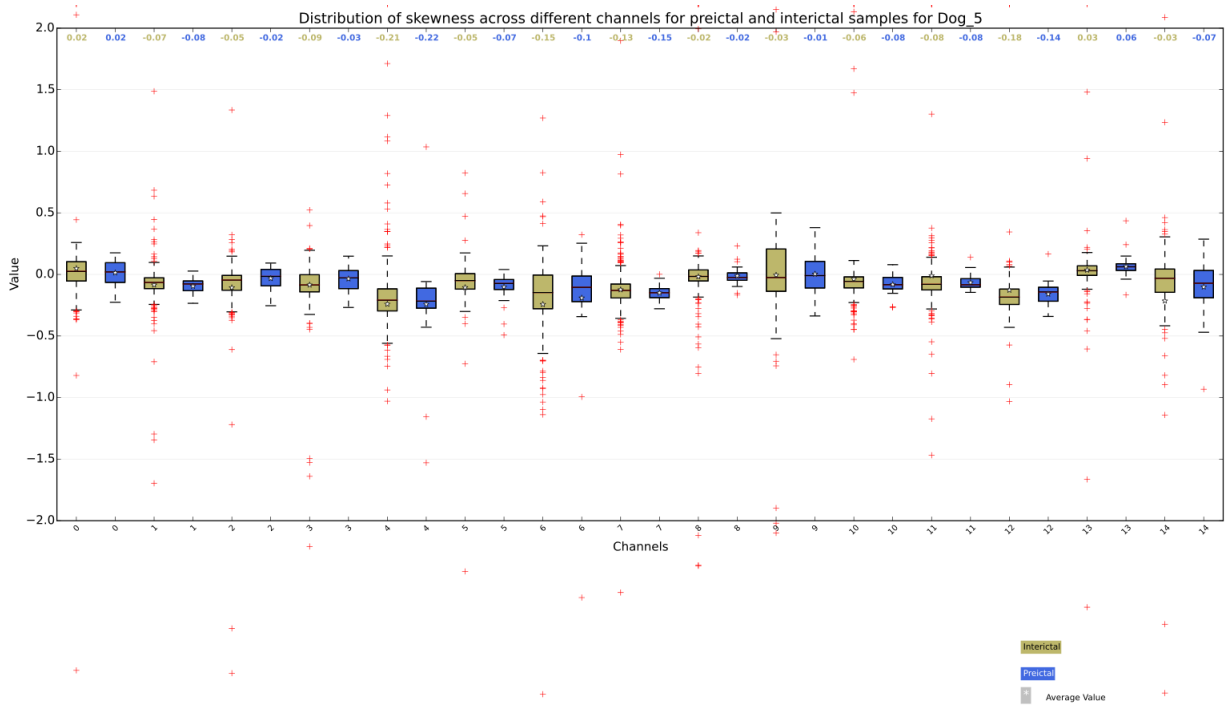
(b) Dog 2


Distribution of skewness across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of skewness across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

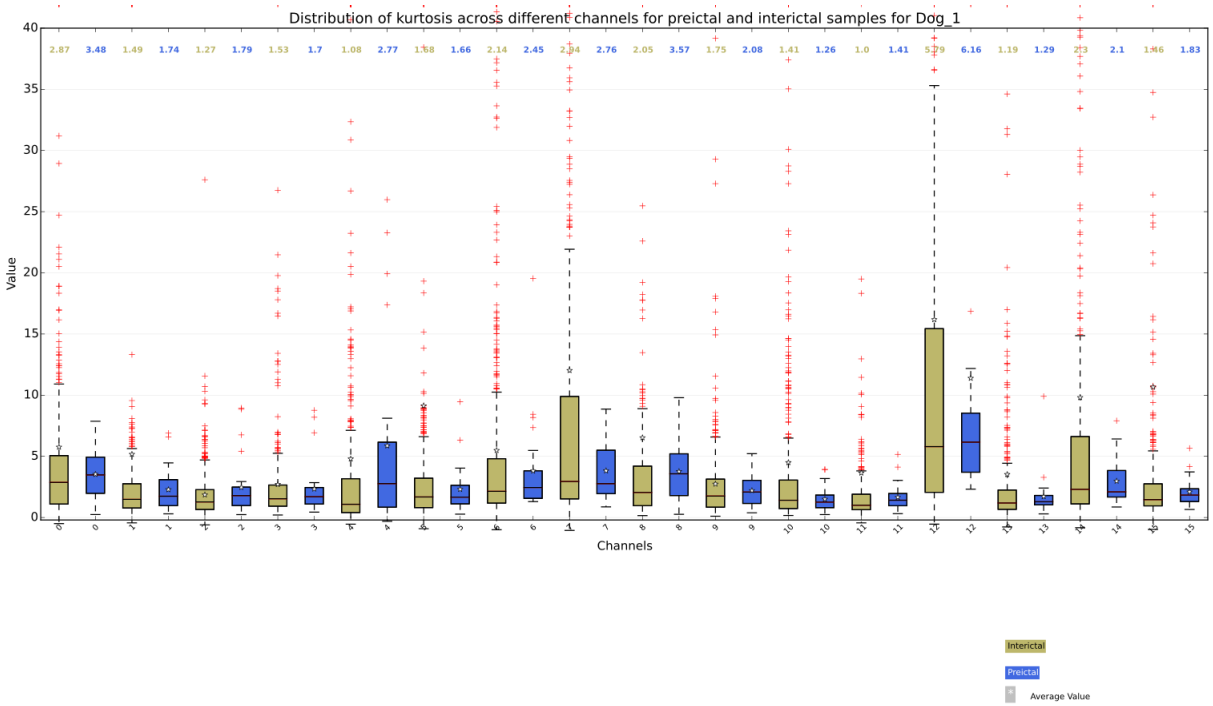

Distribution of skewness across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

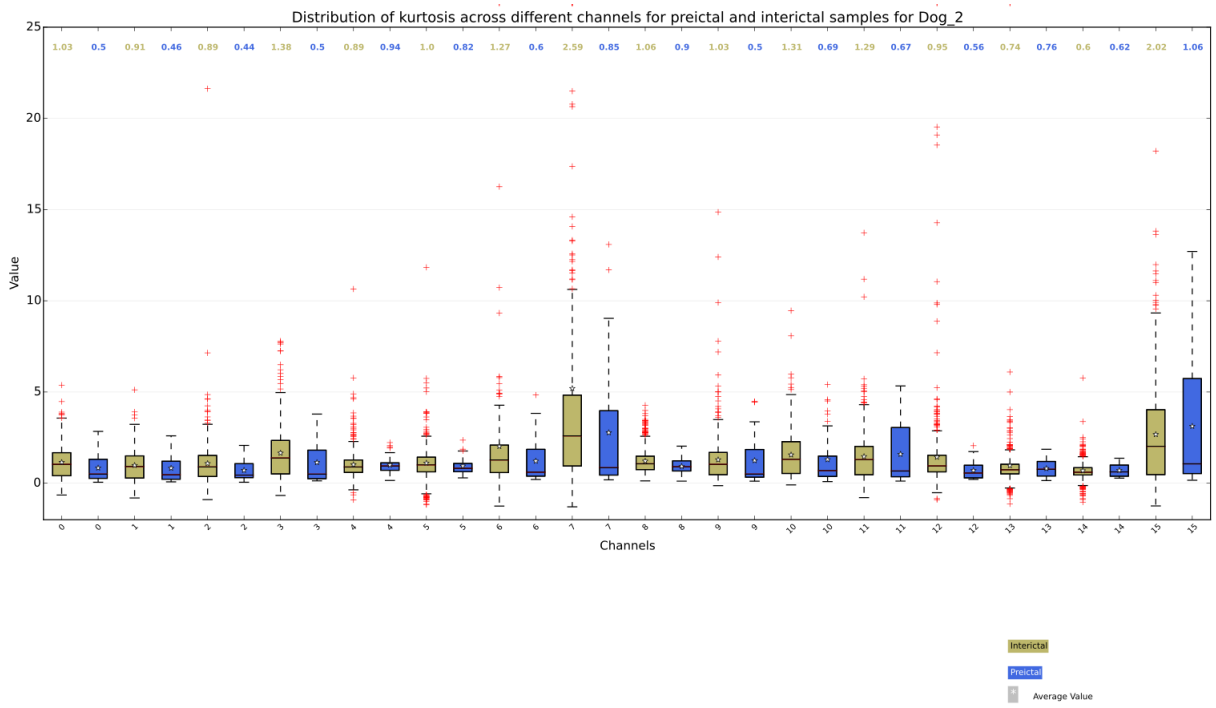
(e) Dog 5

A.3 Kurtosis

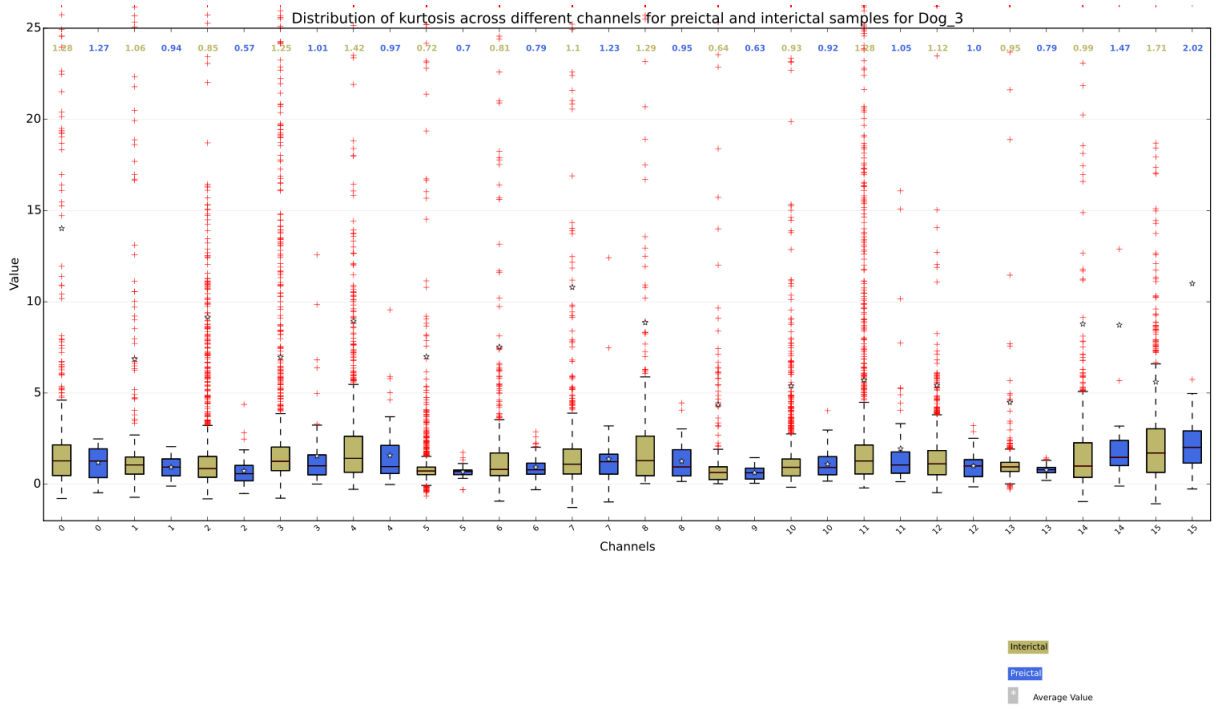
Figure A.3: Distribution of Kurtosis across different subjects

(a) Dog 1

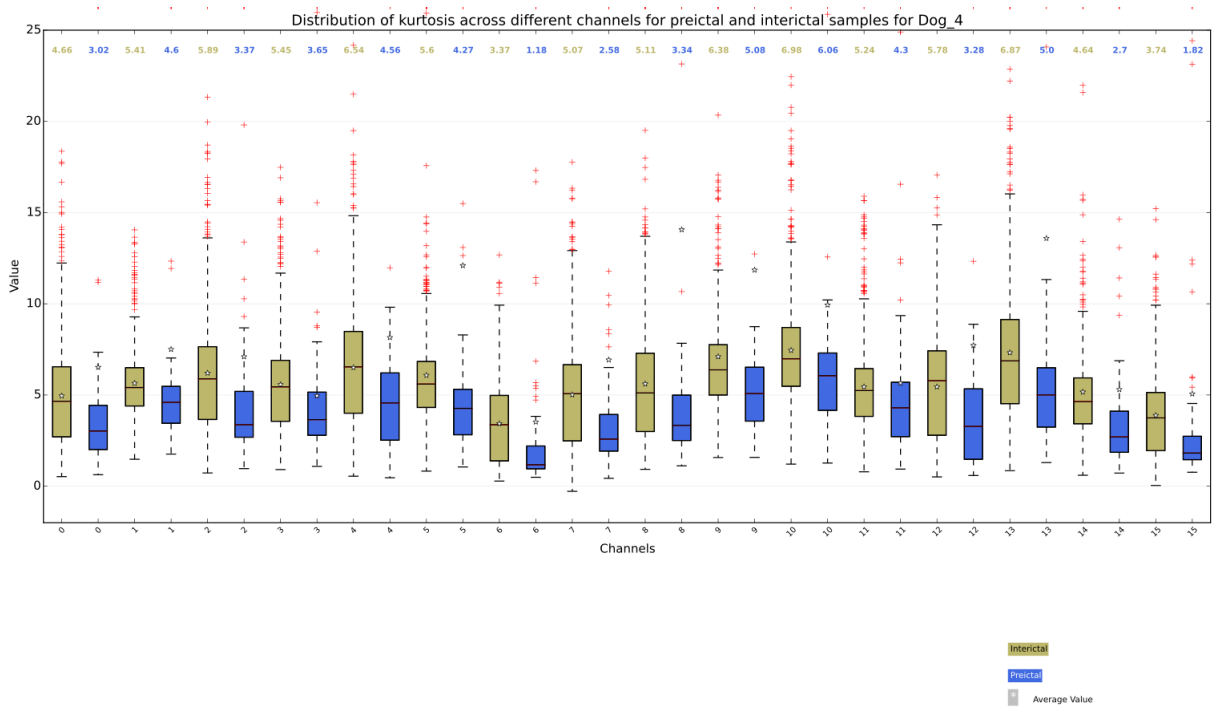



Distribution of kurtosis across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

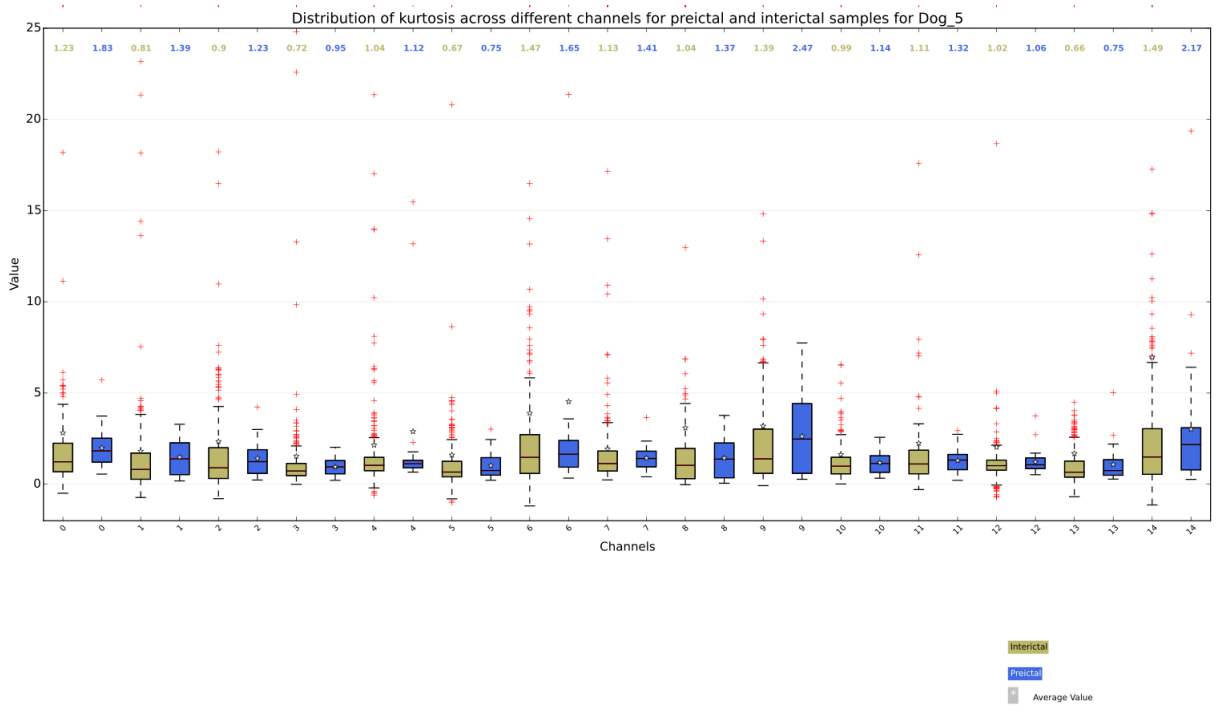
(b) Dog 2


Distribution of kurtosis across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of kurtosis across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4


Distribution of kurtosis across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

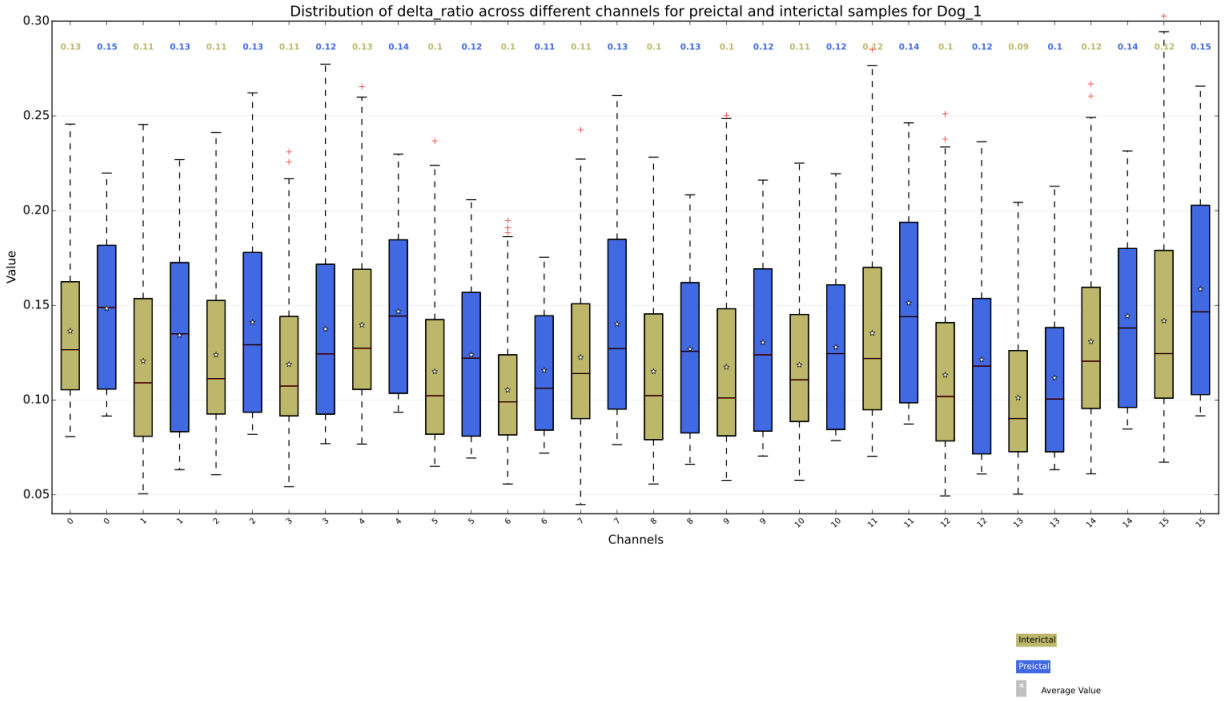
(e) Dog 5

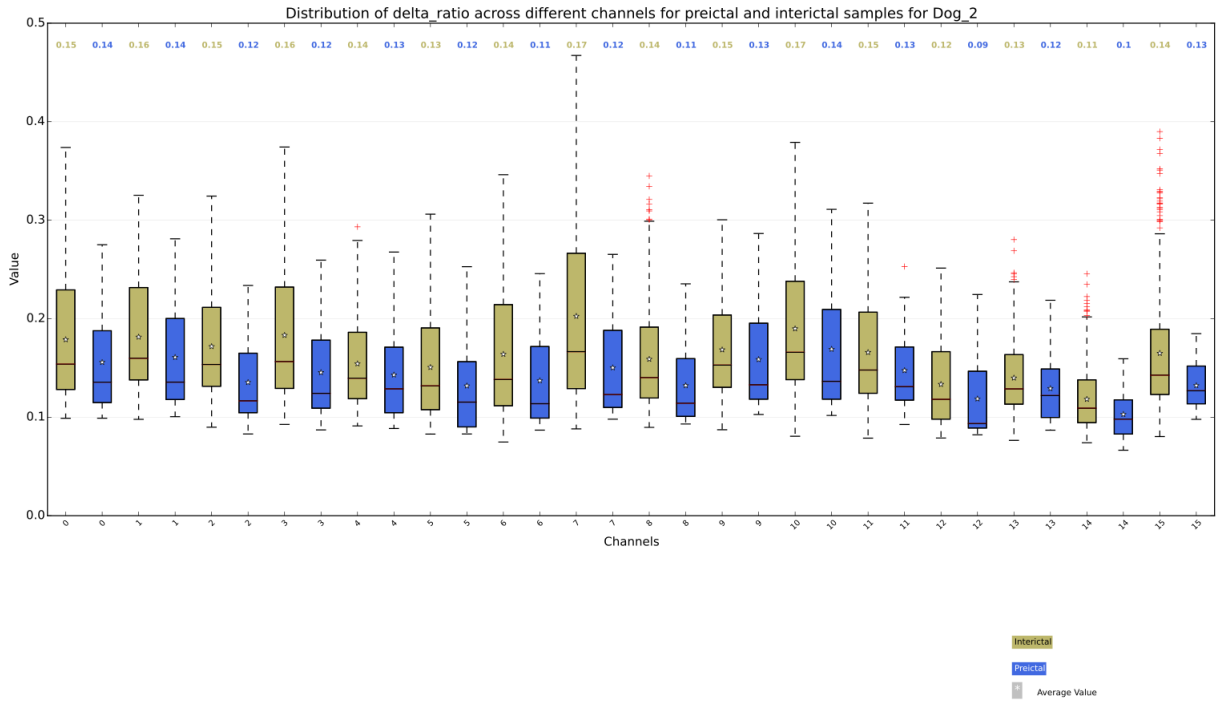
A.4 Frequency Sub-bands Energy

A.4.1 Delta Band

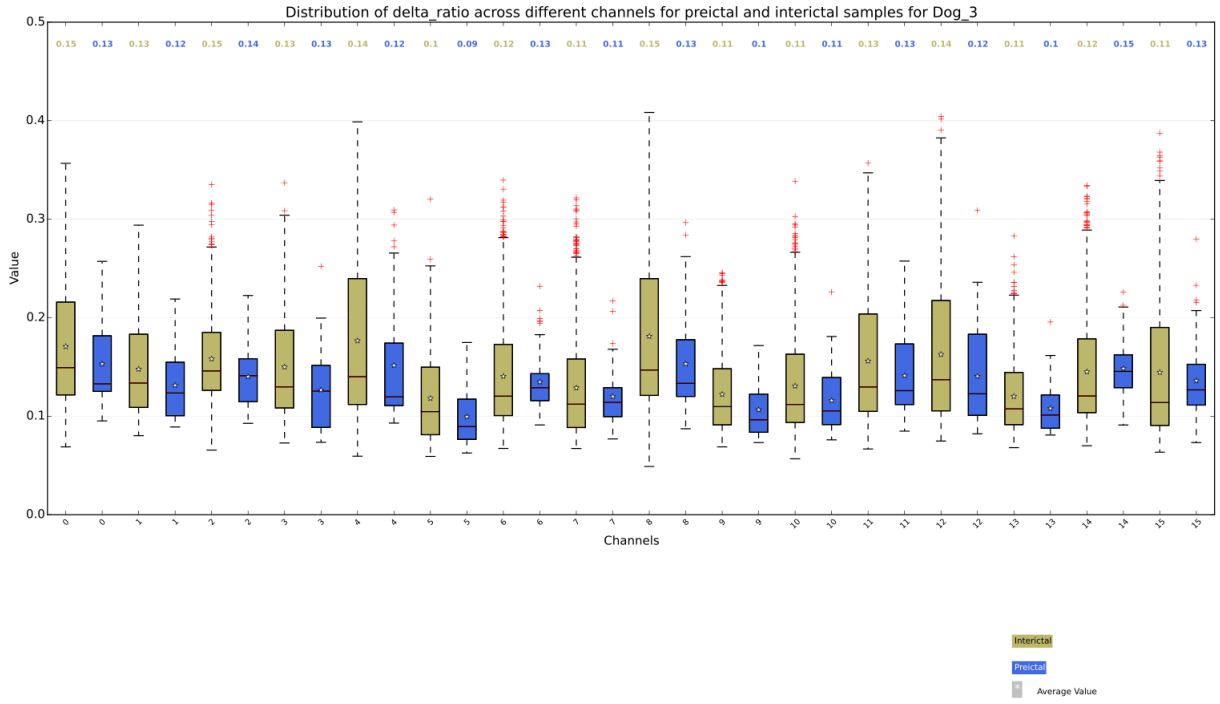
Figure A.4: Distribution of Delta Band across different subjects

(a) Dog 1

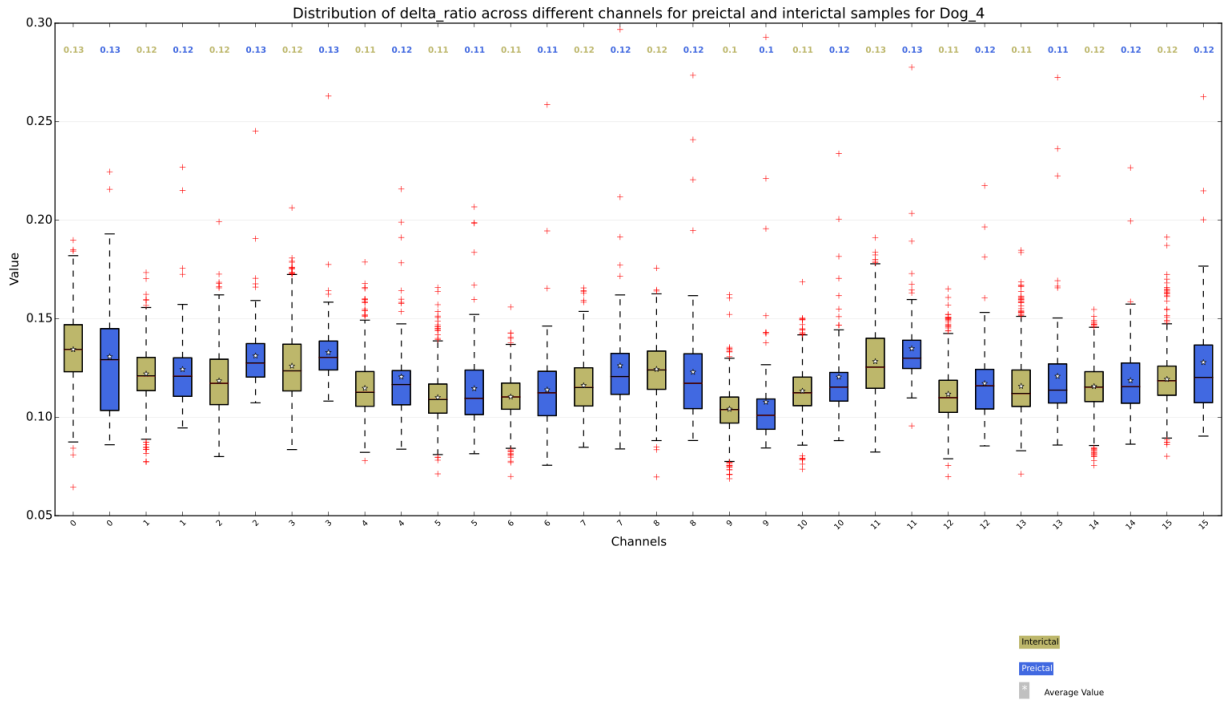



Distribution of delta_ratio across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

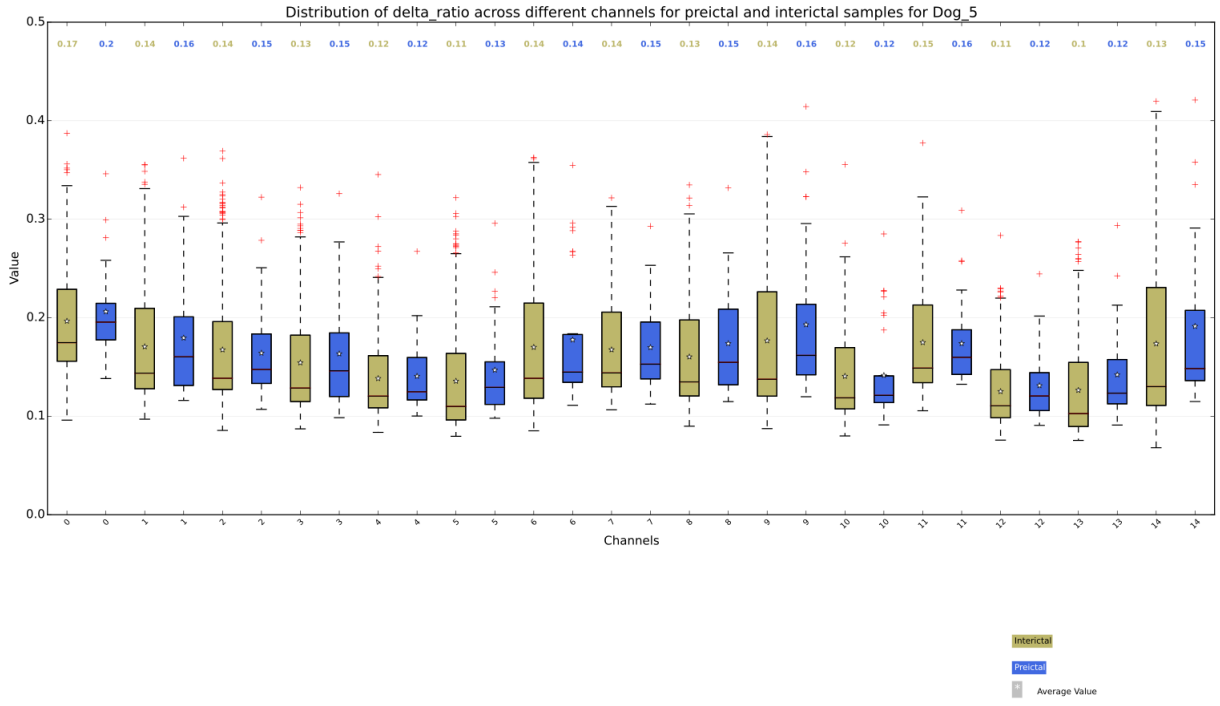
(b) Dog 2


Distribution of delta_ratio across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of delta_ratio across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

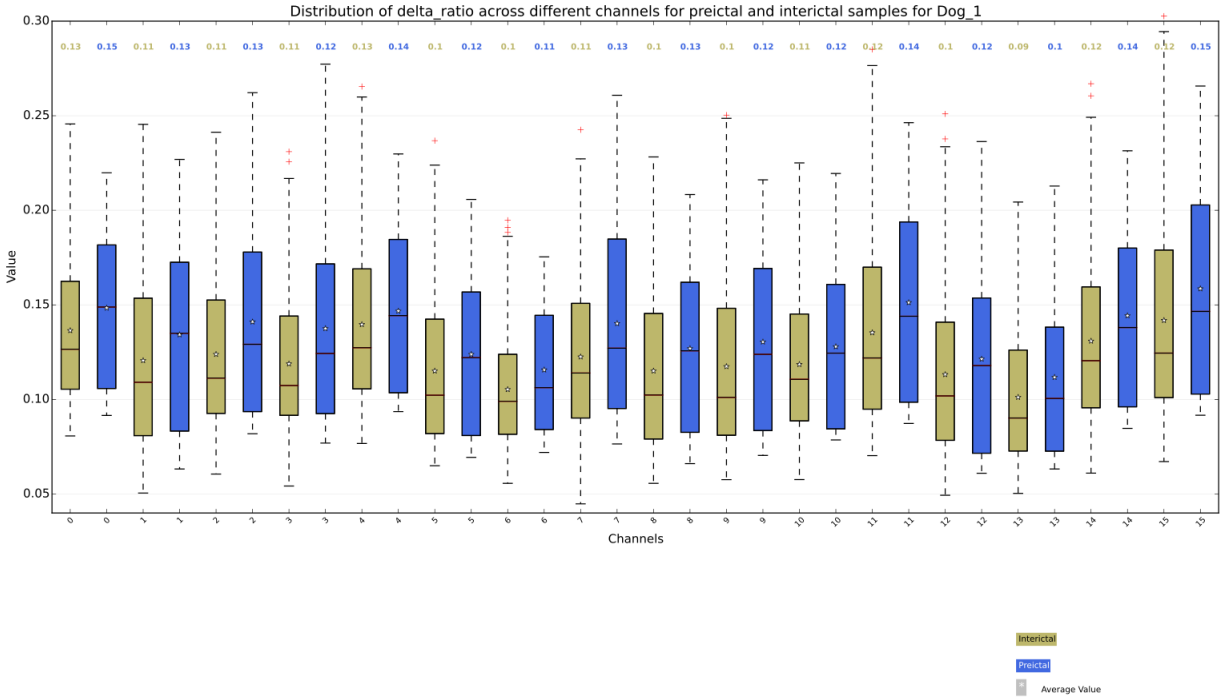

Distribution of delta_ratio across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

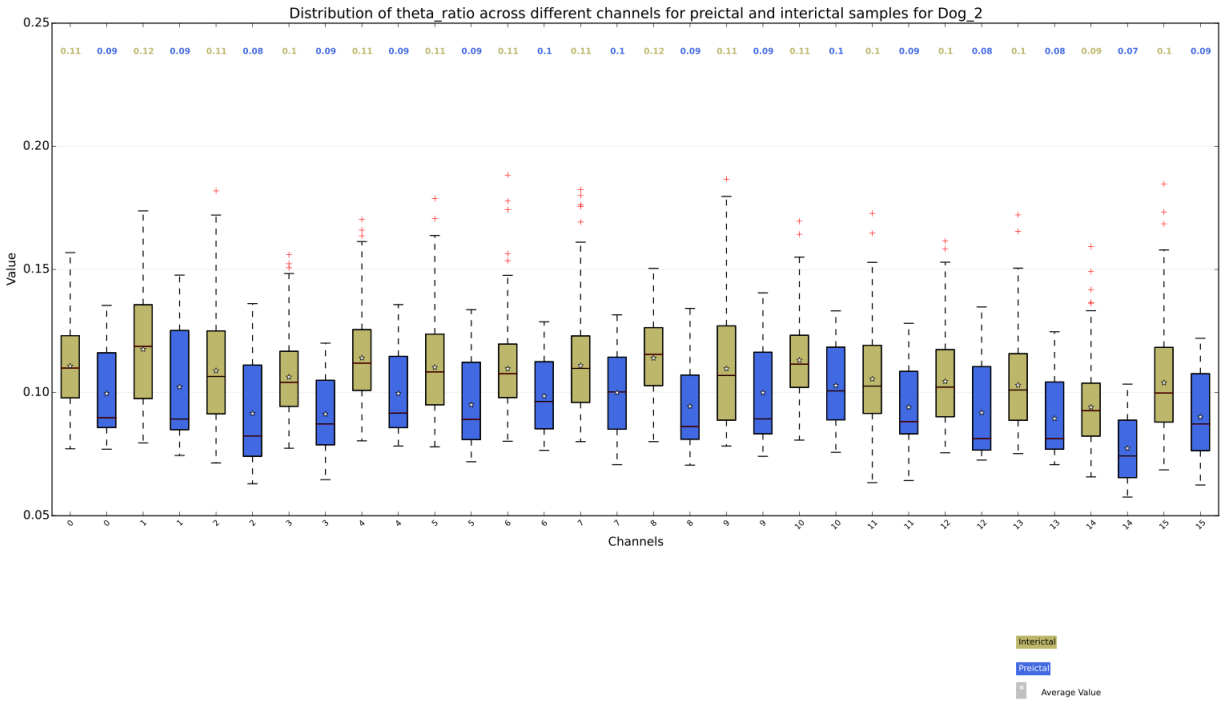
(e) Dog 5

A.4.2 Theta Band

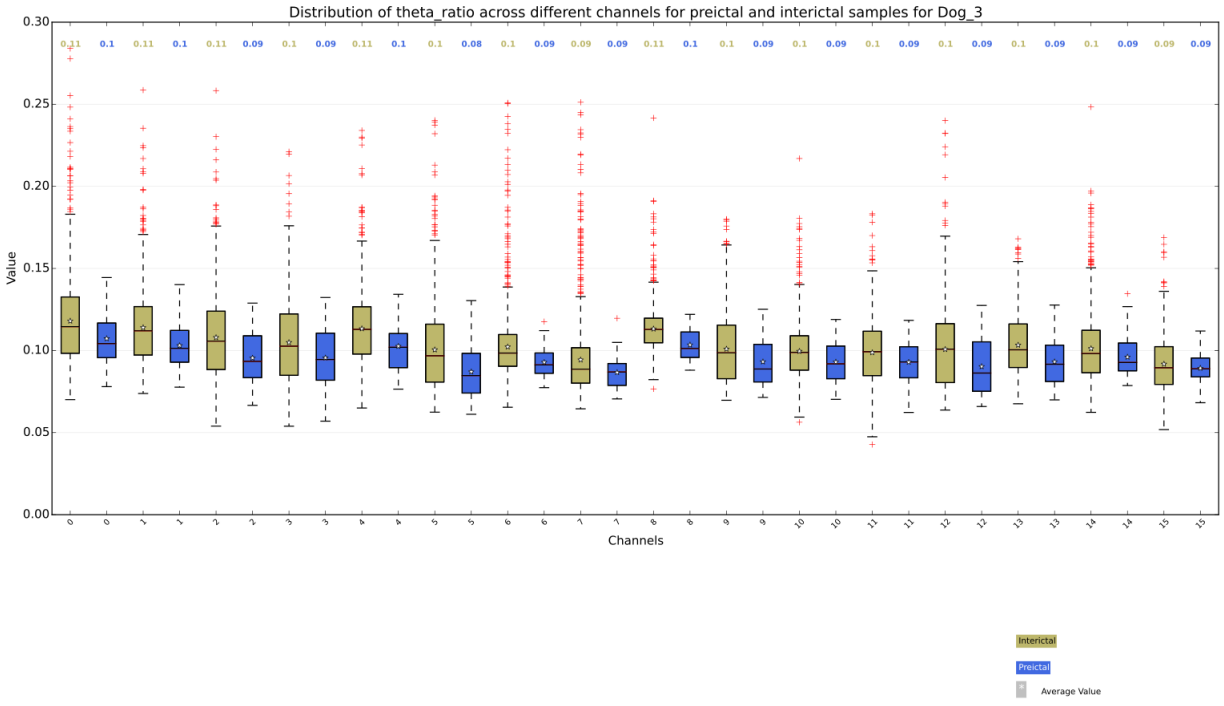
Figure A.5: Distribution of Theta Band across different subjects

(a) Dog 1

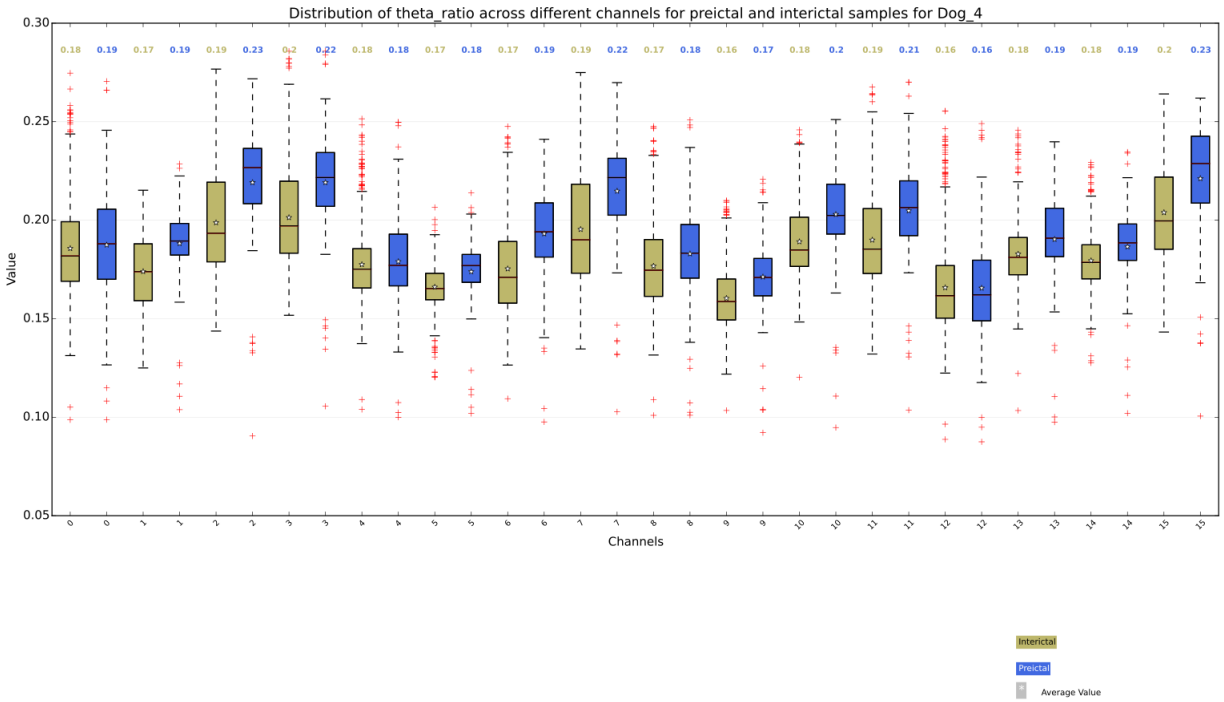



Distribution of theta_ratio across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

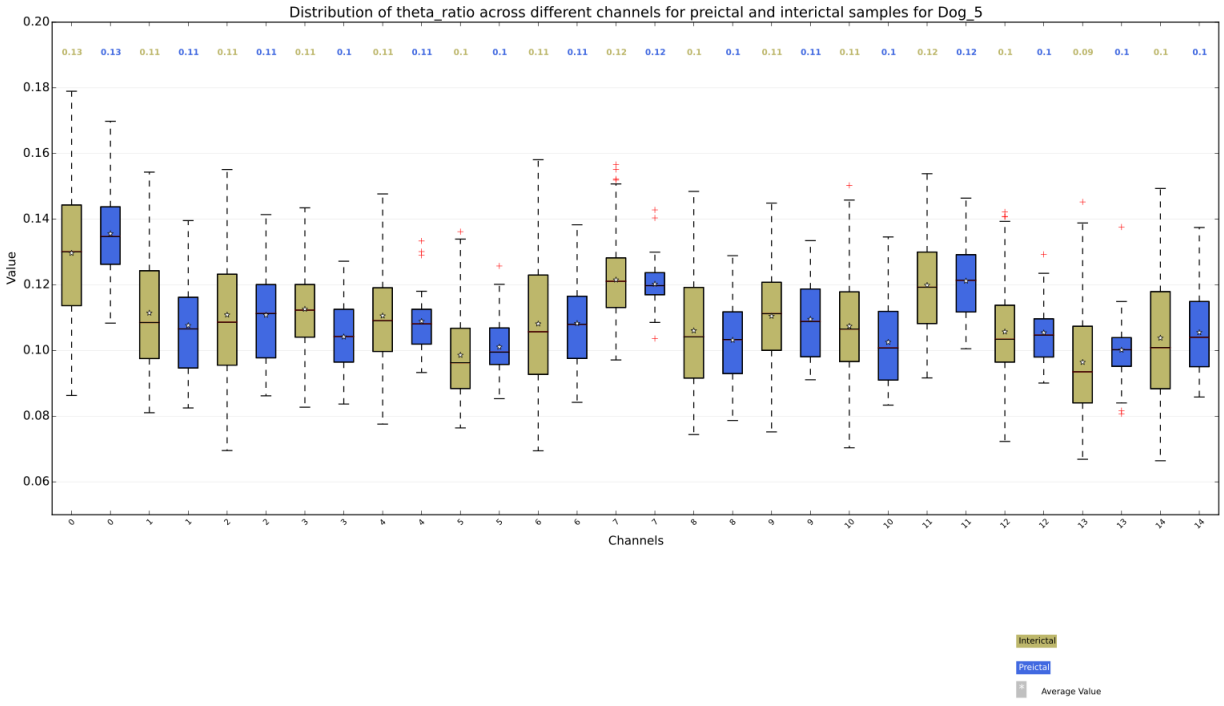
(b) Dog 2


Distribution of theta_ratio across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of theta_ratio across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

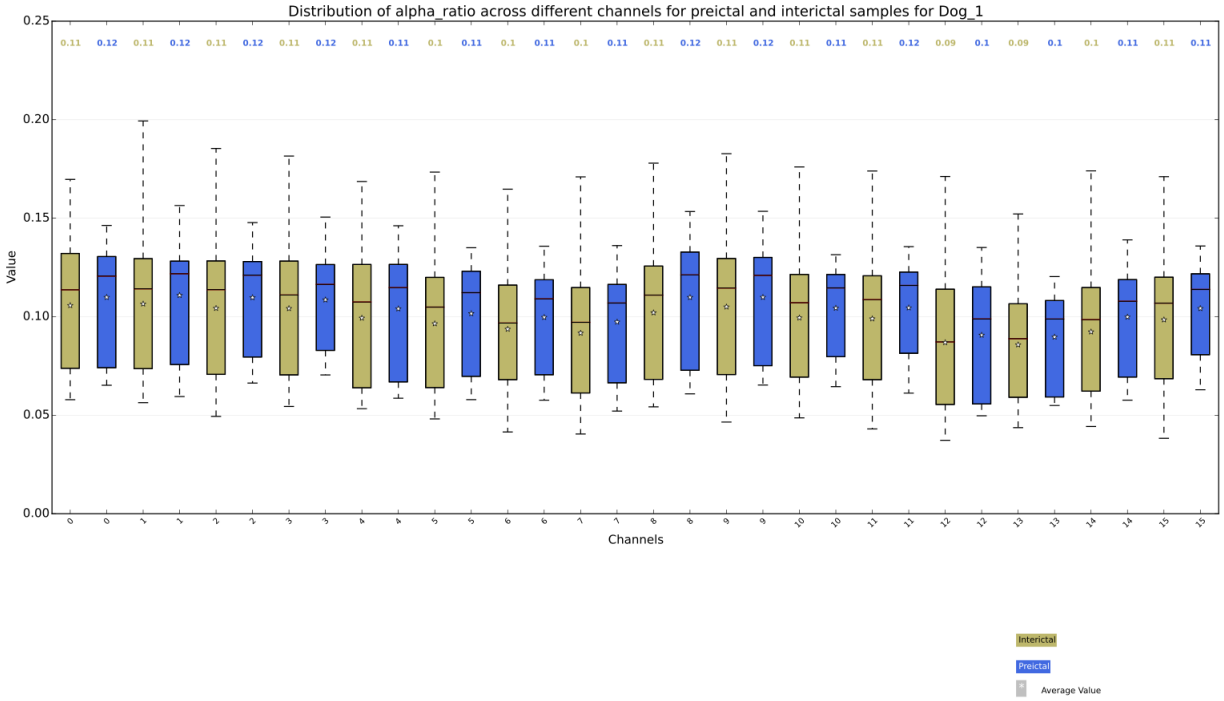

Distribution of theta_ratio across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

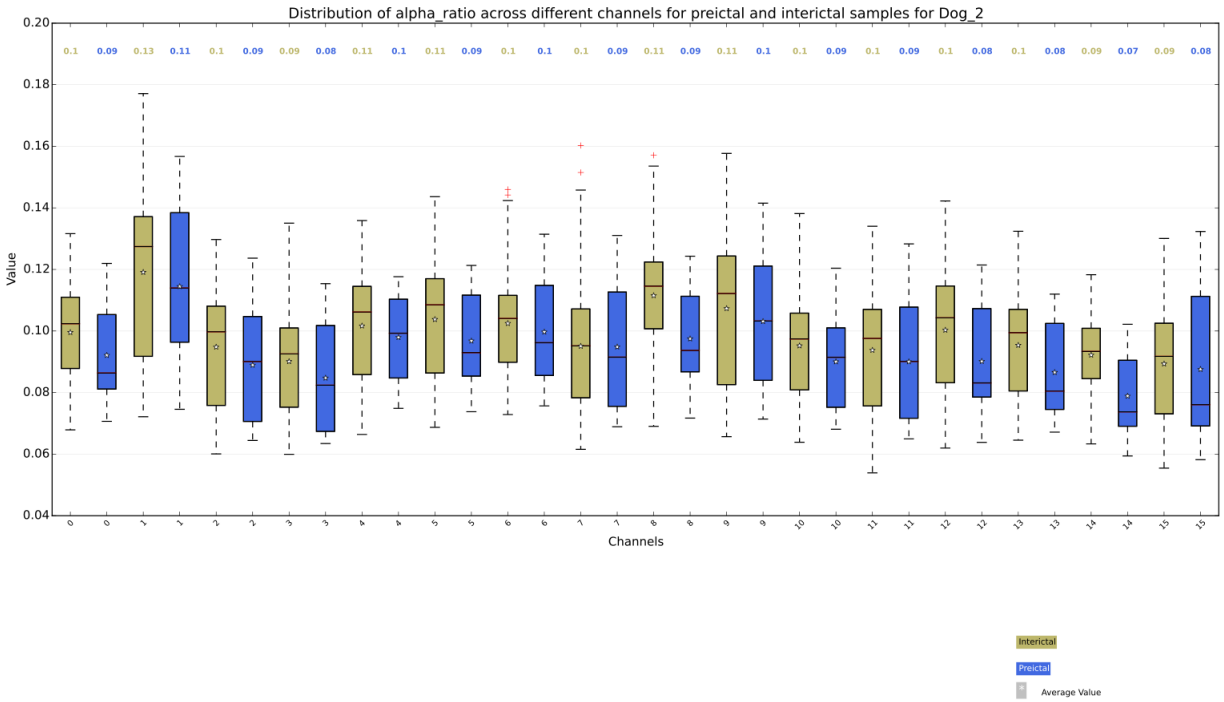
(e) Dog 5

A.4.3 Alpha Band

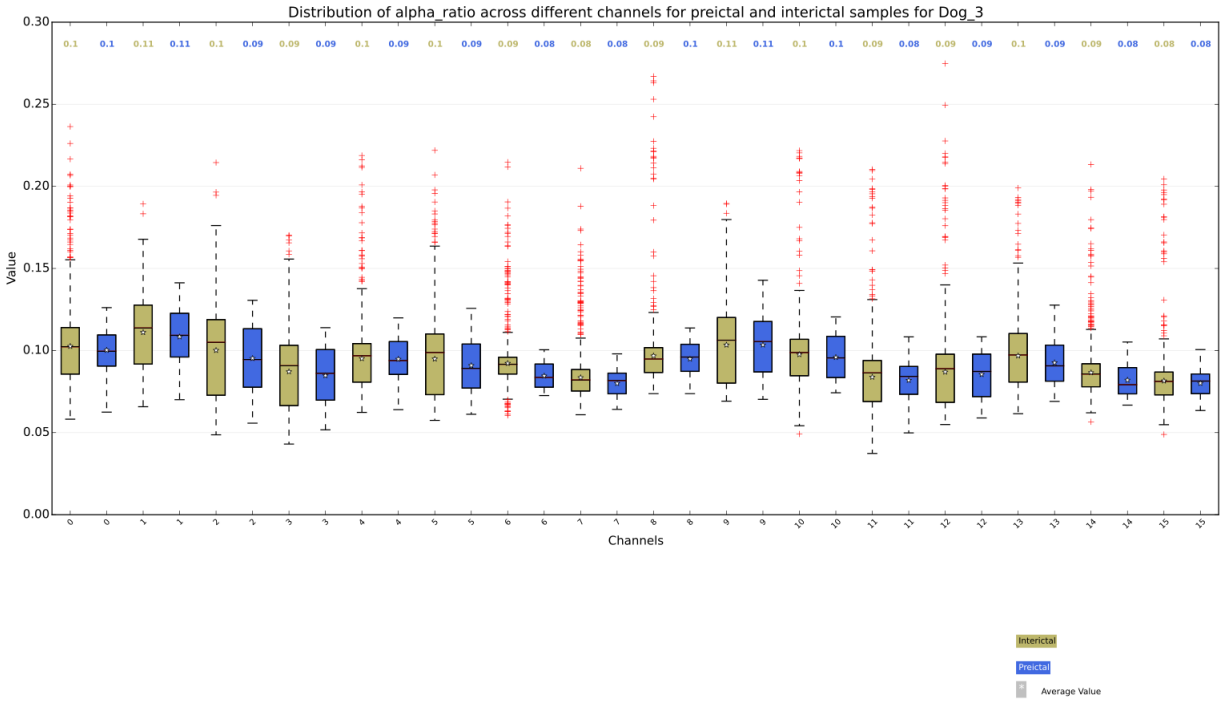
Figure A.6: Distribution of Alpha Band across different subjects

(a) Dog 1

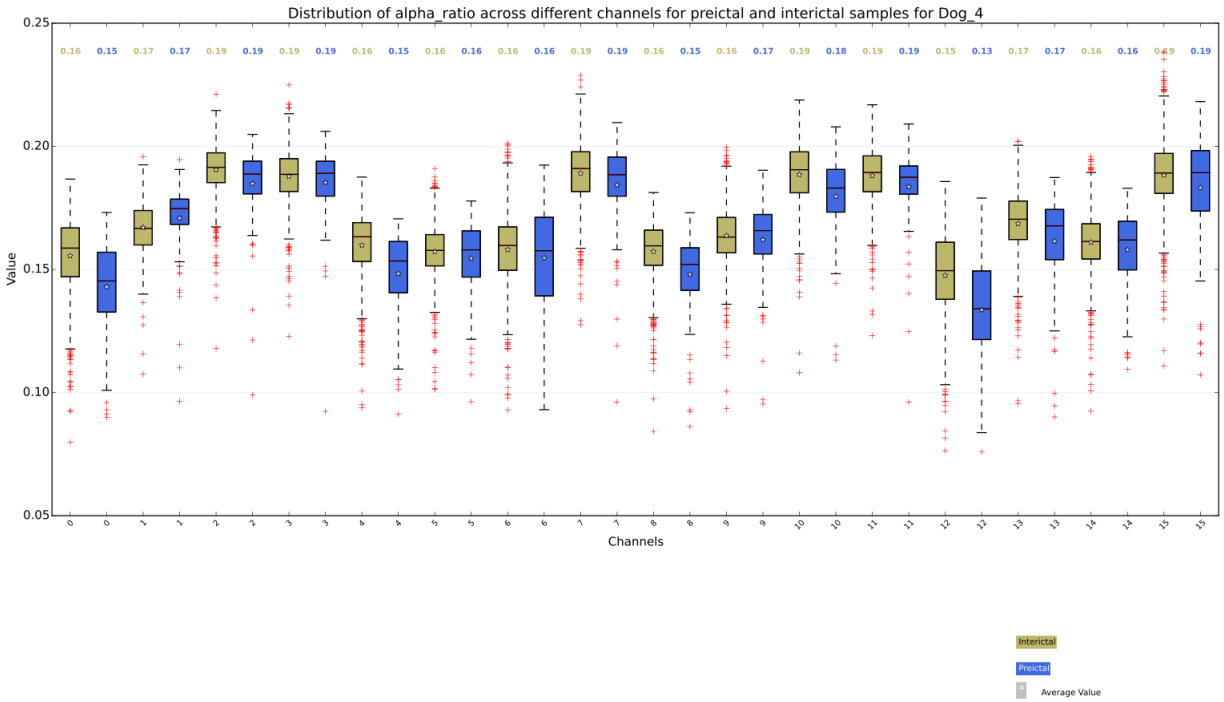



Distribution of alpha_ratio across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

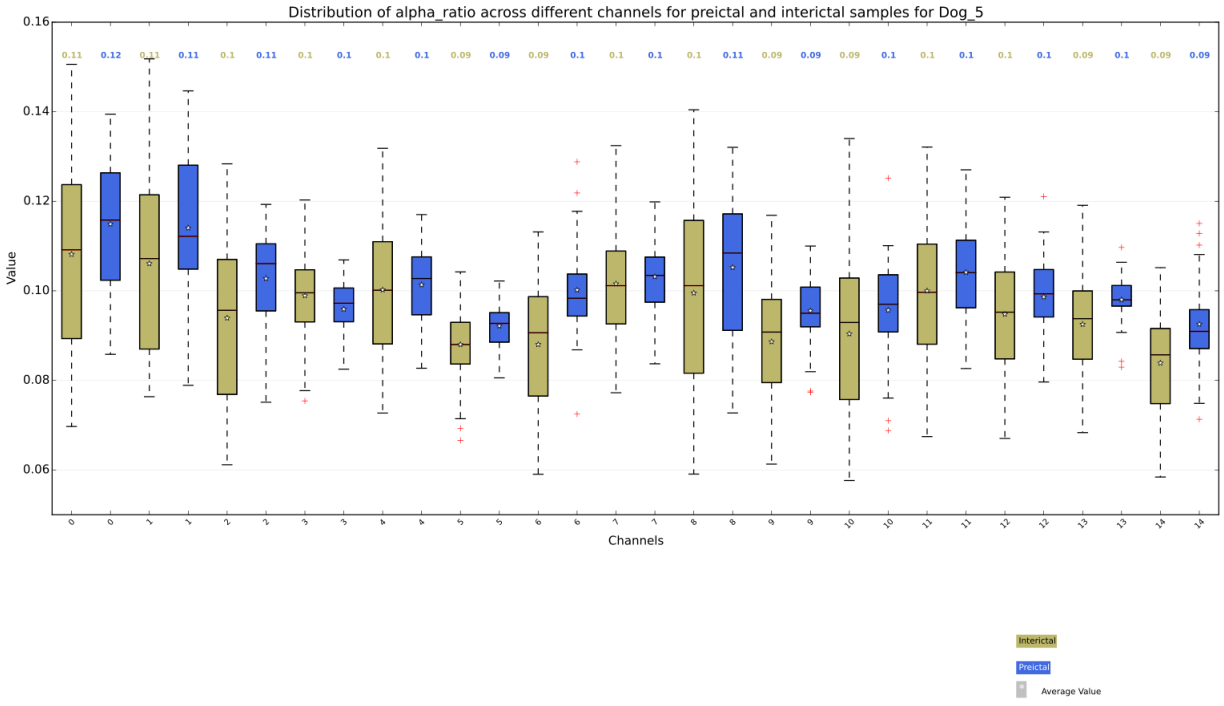
(b) Dog 2


Distribution of alpha_ratio across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of alpha_ratio across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

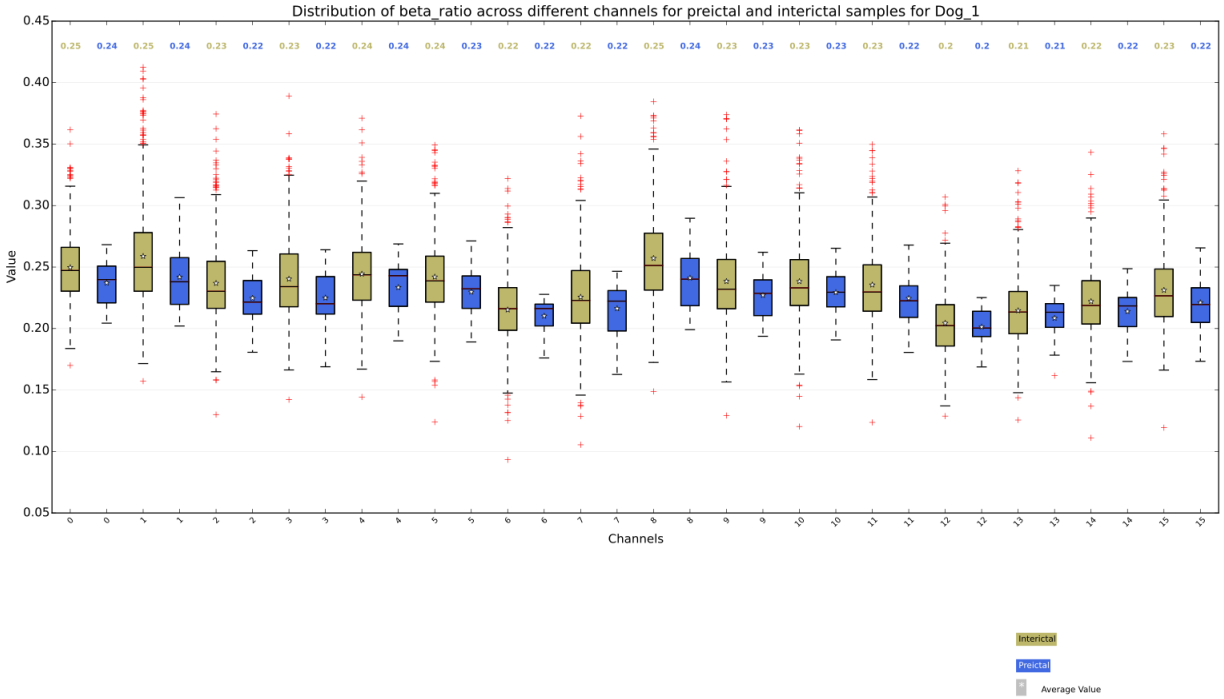

Distribution of alpha_ratio across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

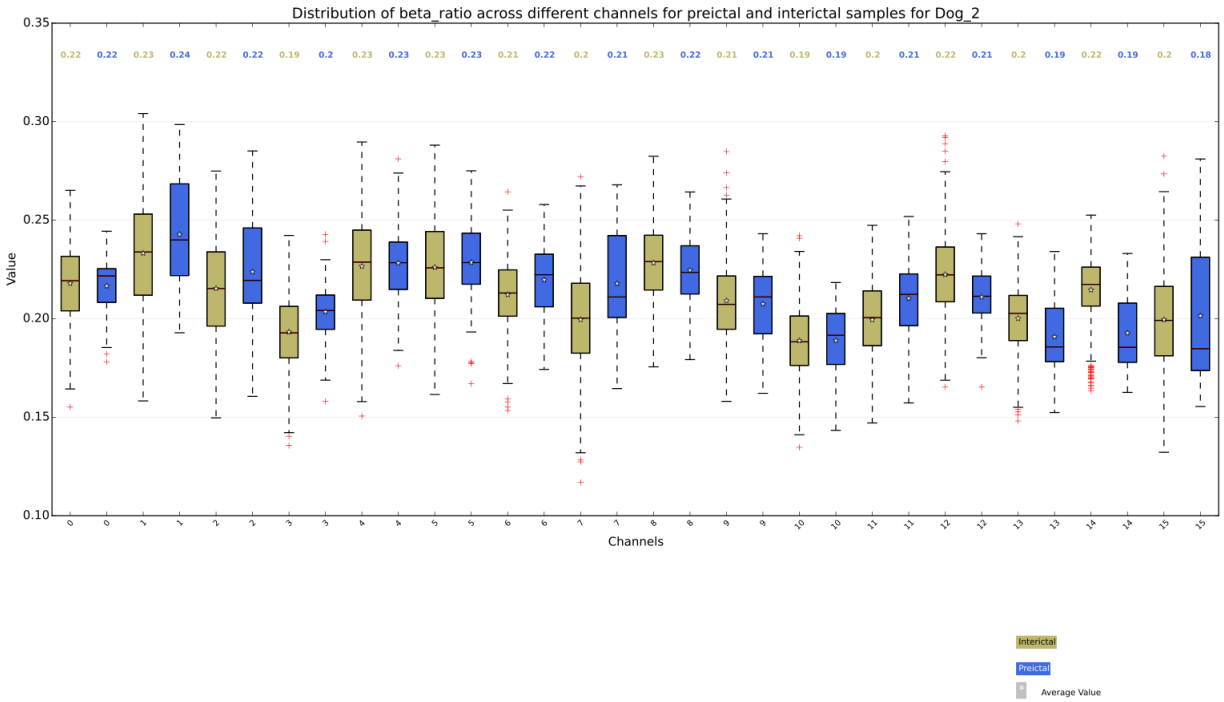
(e) Dog 5

A.4.4 Beta Band

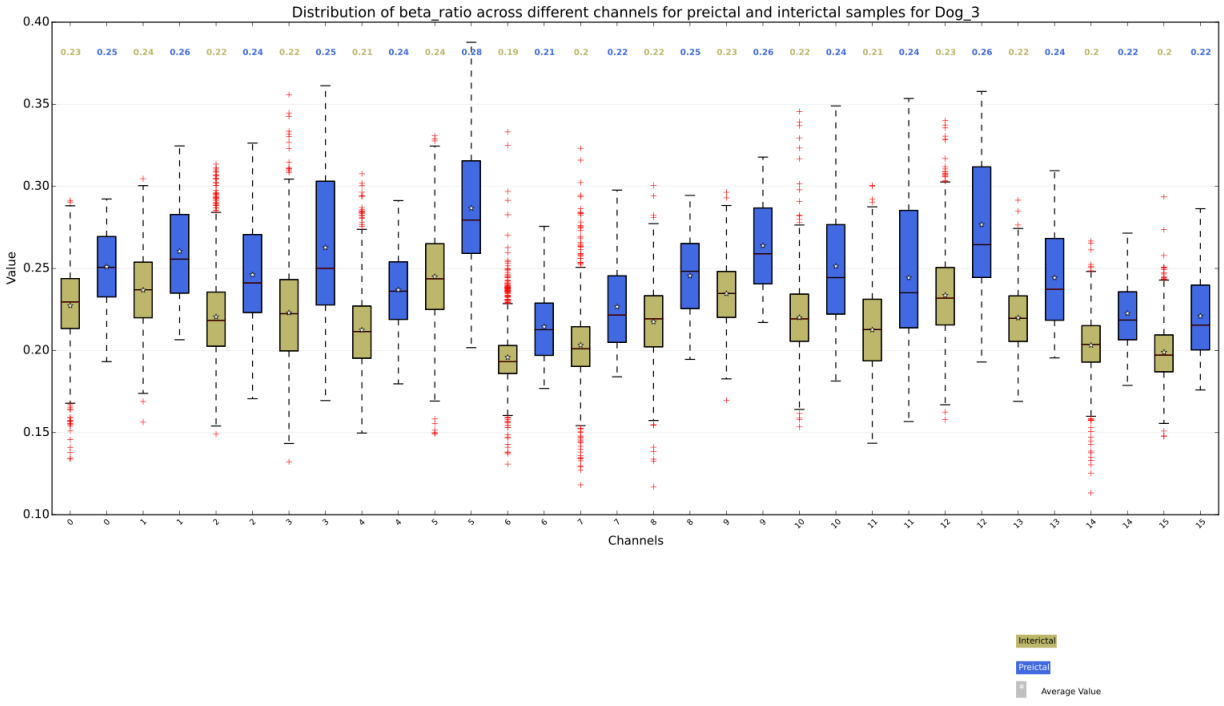
Figure A.7: Distribution of Beta Band across different subjects

(a) Dog 1

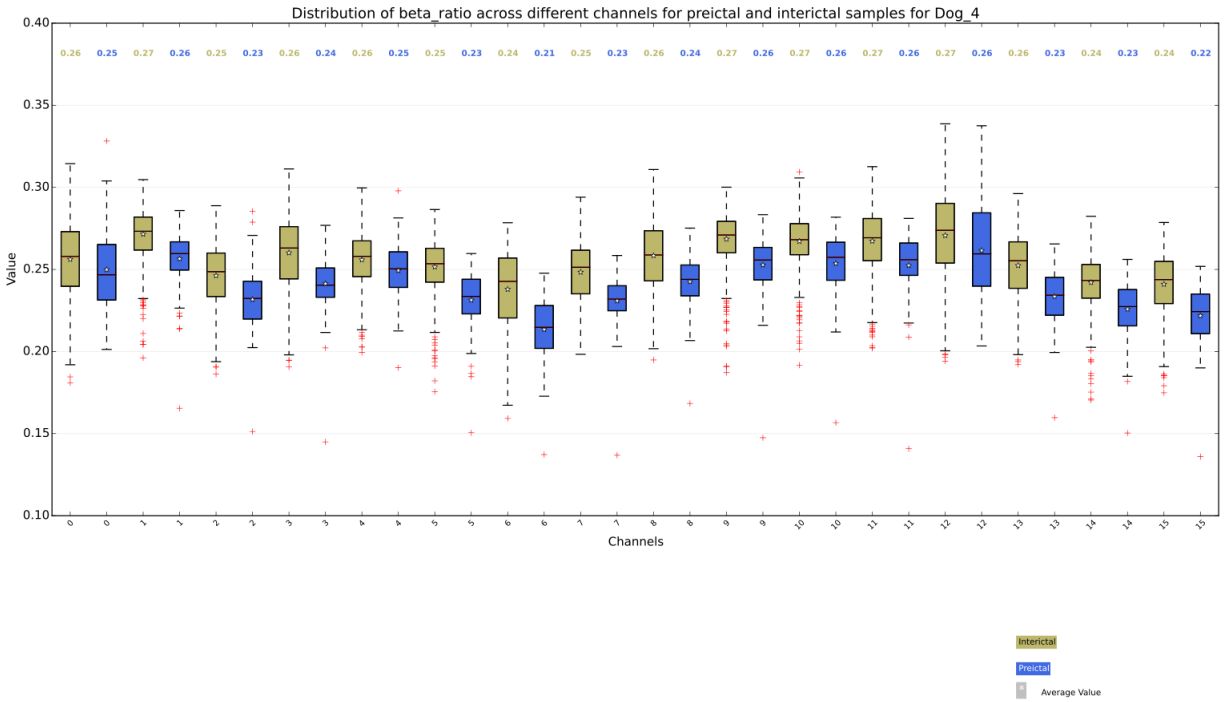



Distribution of beta_ratio across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

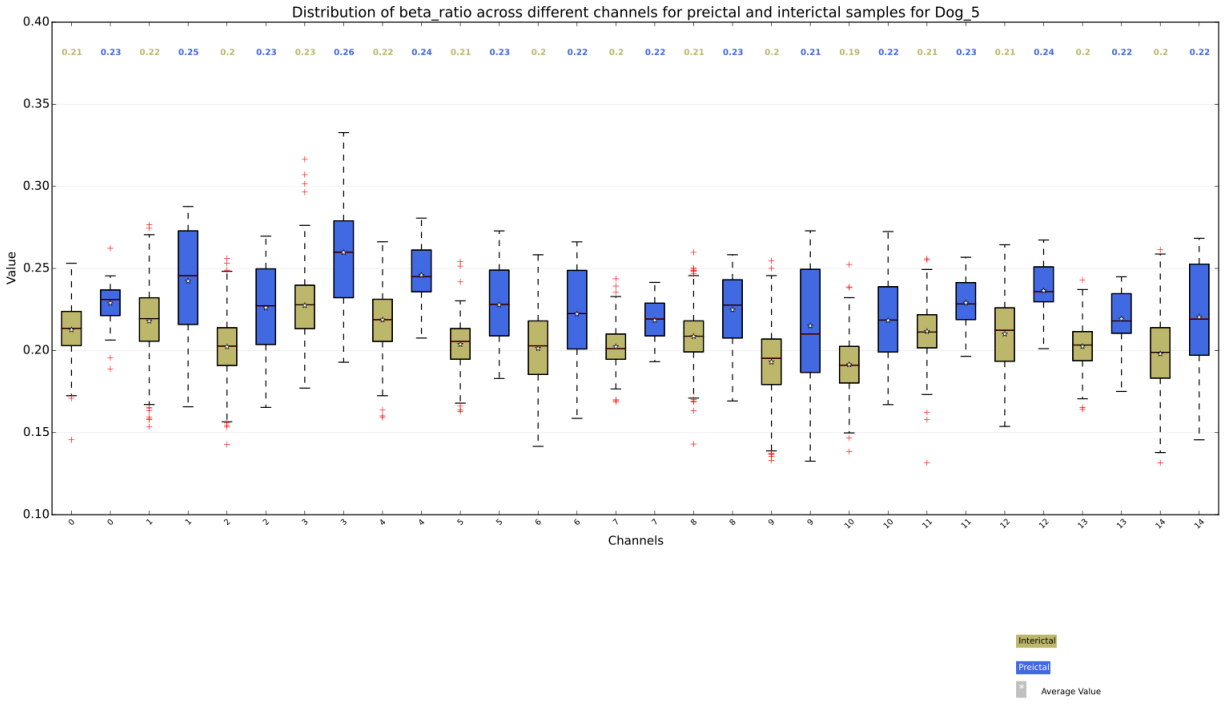
(b) Dog 2


Distribution of beta_ratio across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of beta_ratio across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

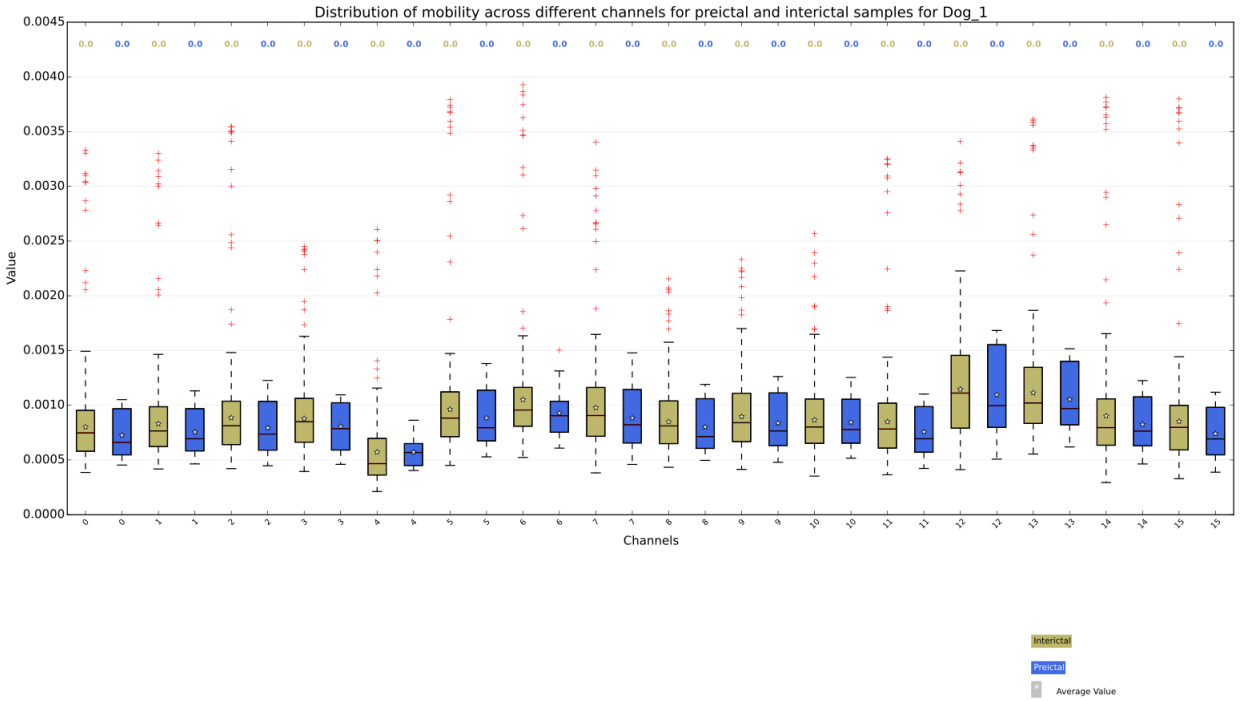

Distribution of beta_ratio across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

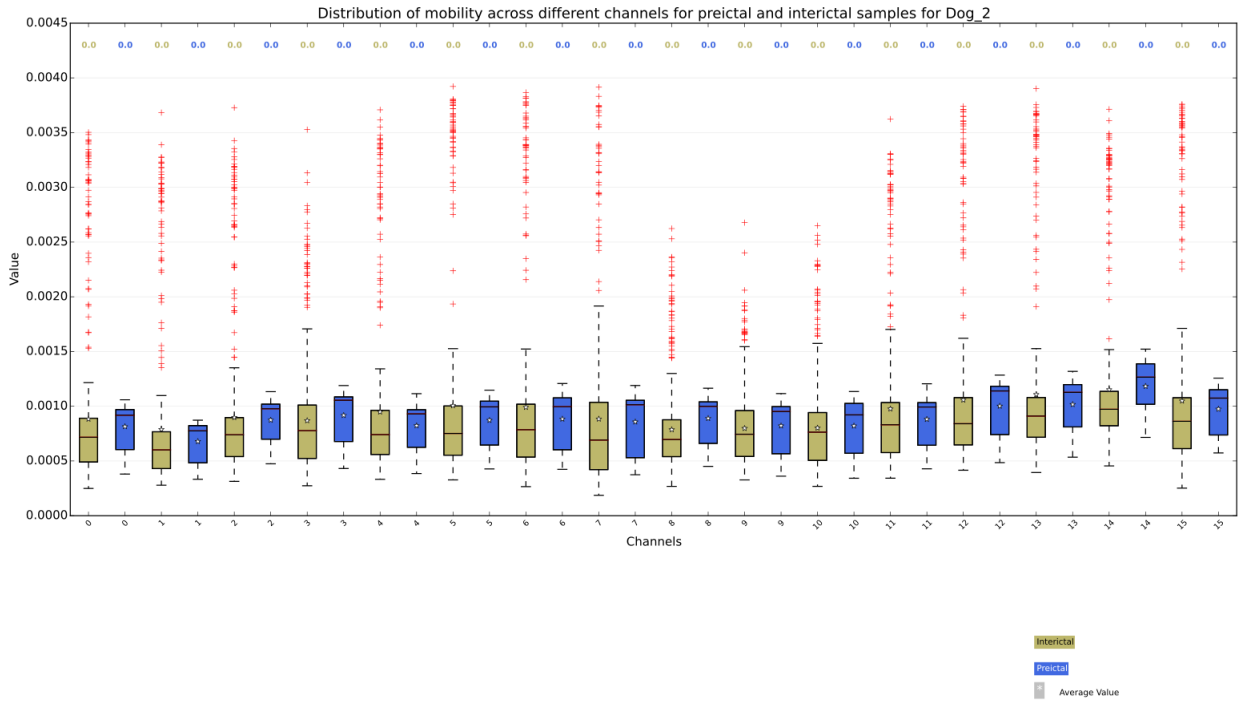
(e) Dog 5

A.5 Hjorth Mobility

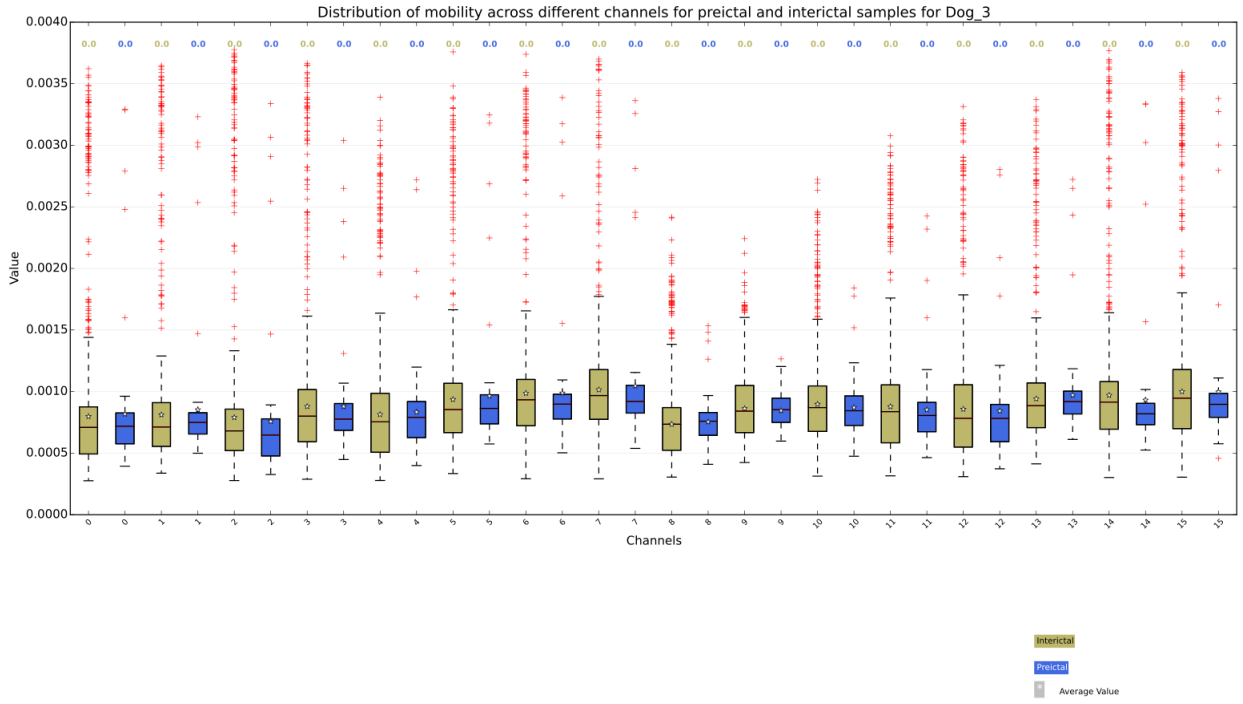
Figure A.8: Distribution of Hjorth mobility across different subjects

(a) Dog 1

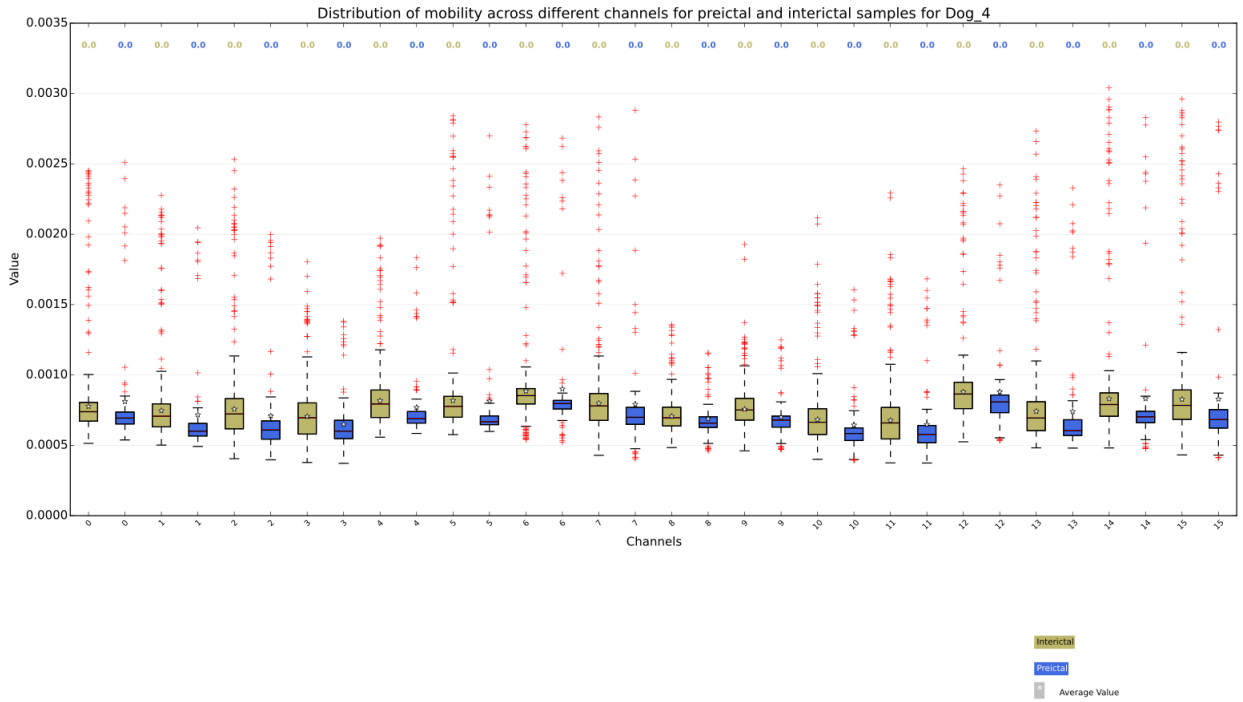



Distribution of mobility across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

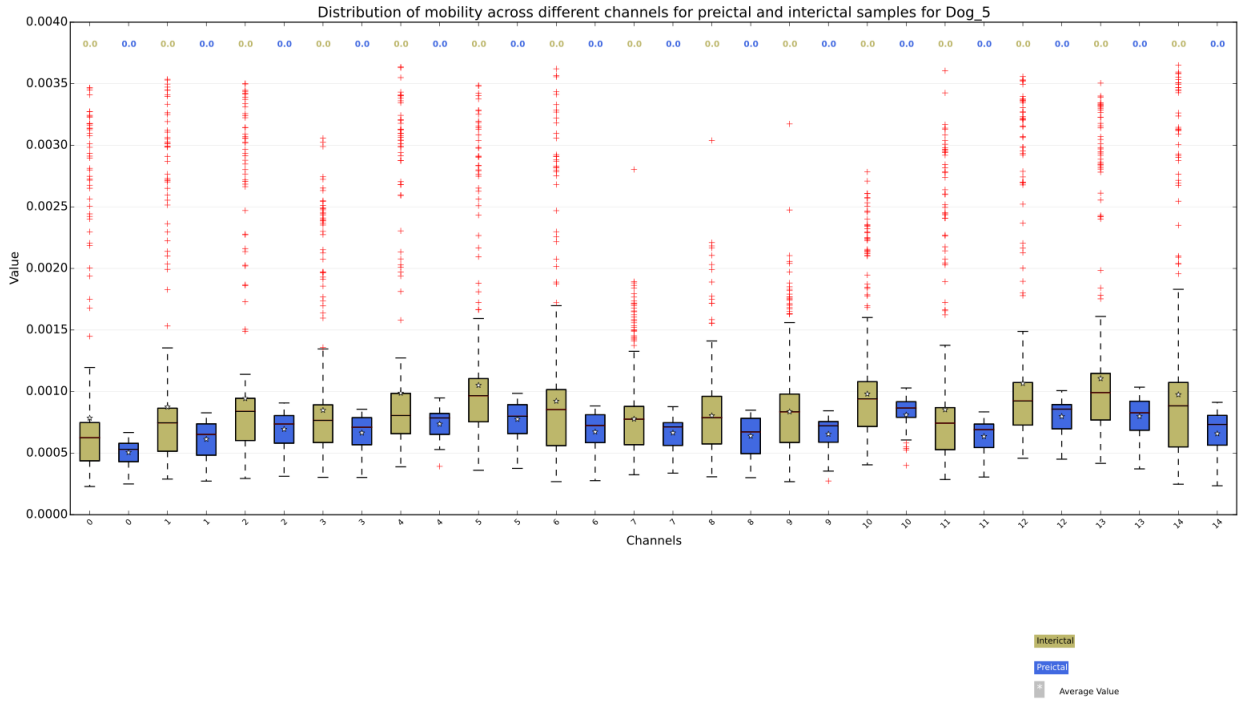
(b) Dog 2


Distribution of mobility across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of mobility across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

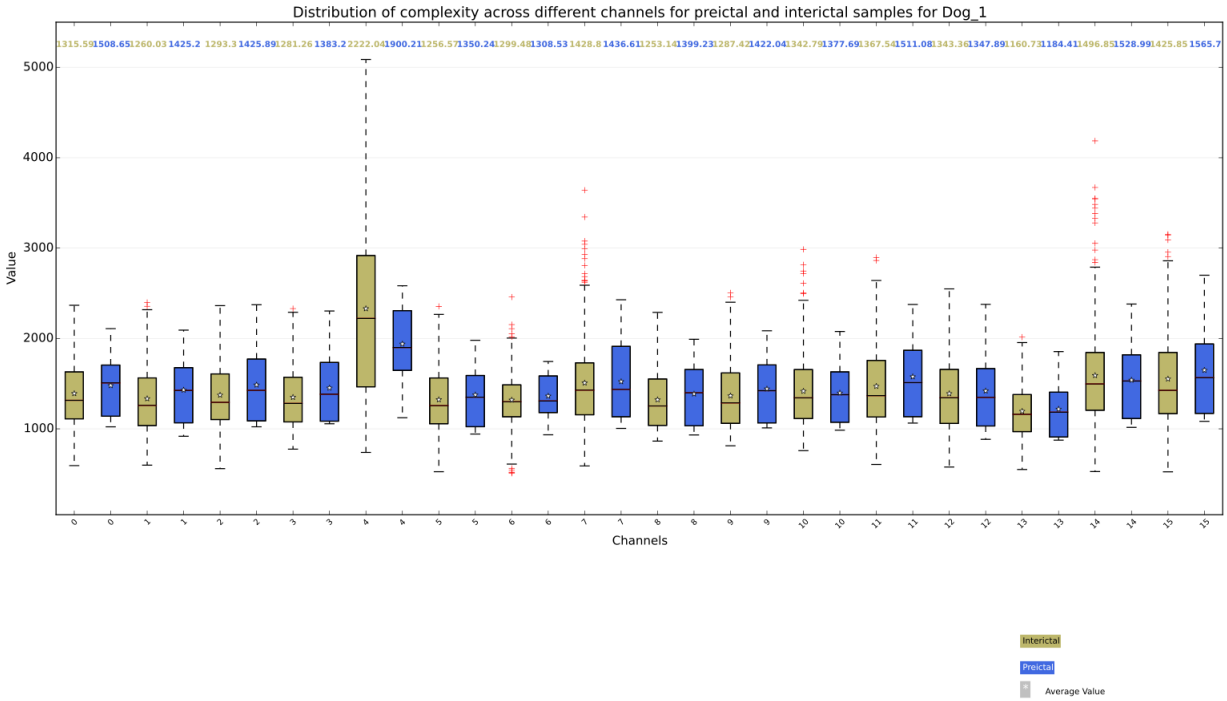

Distribution of mobility across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

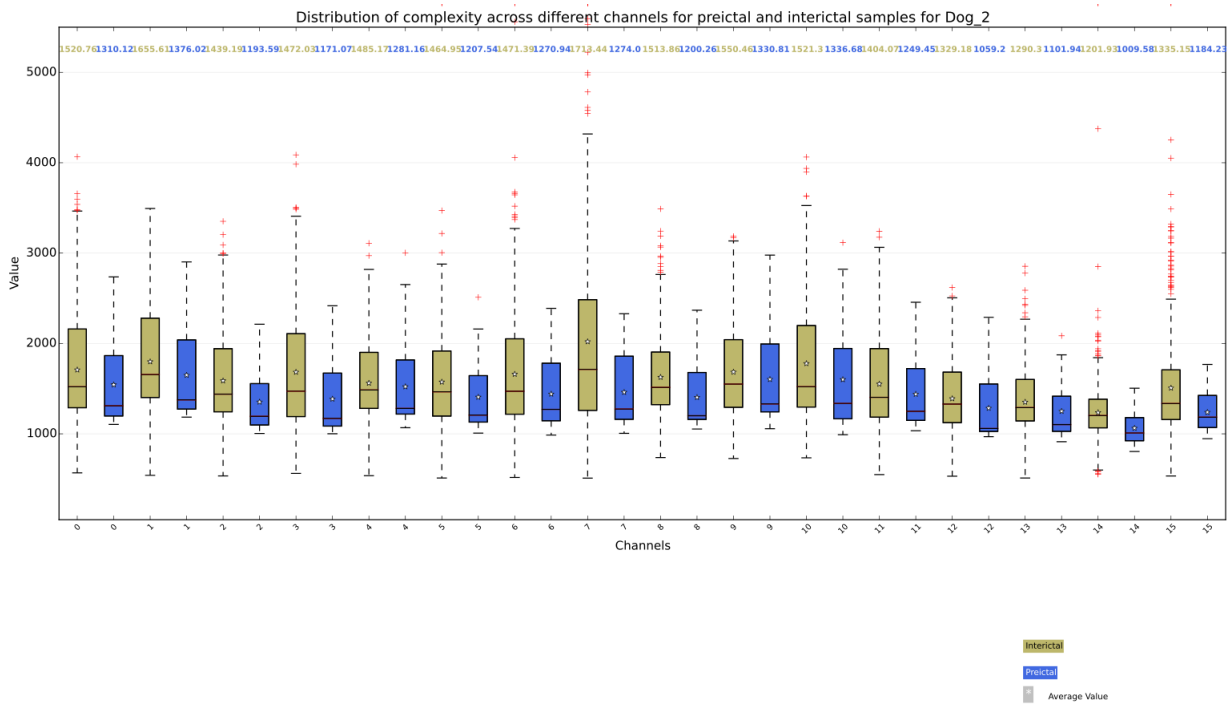
(e) Dog 5

A.6 Hjorth Complexity

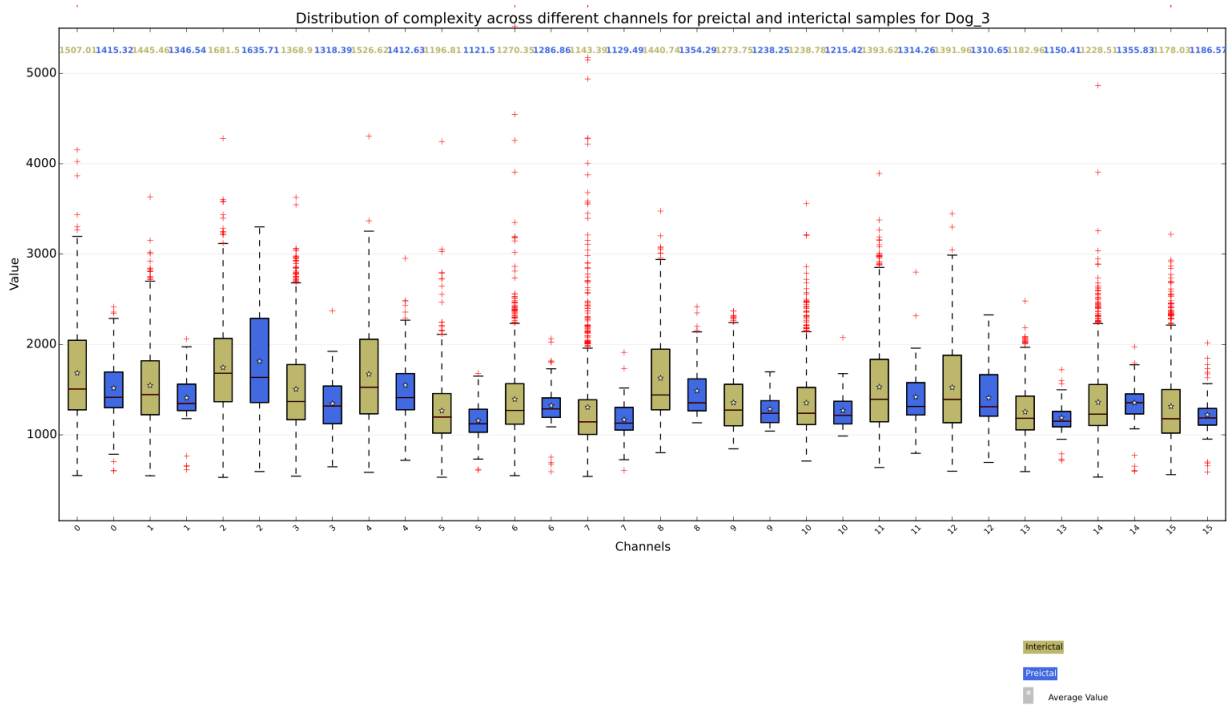
Figure A.9: Distribution of Hjorth complexity across different subjects

(a) Dog 1

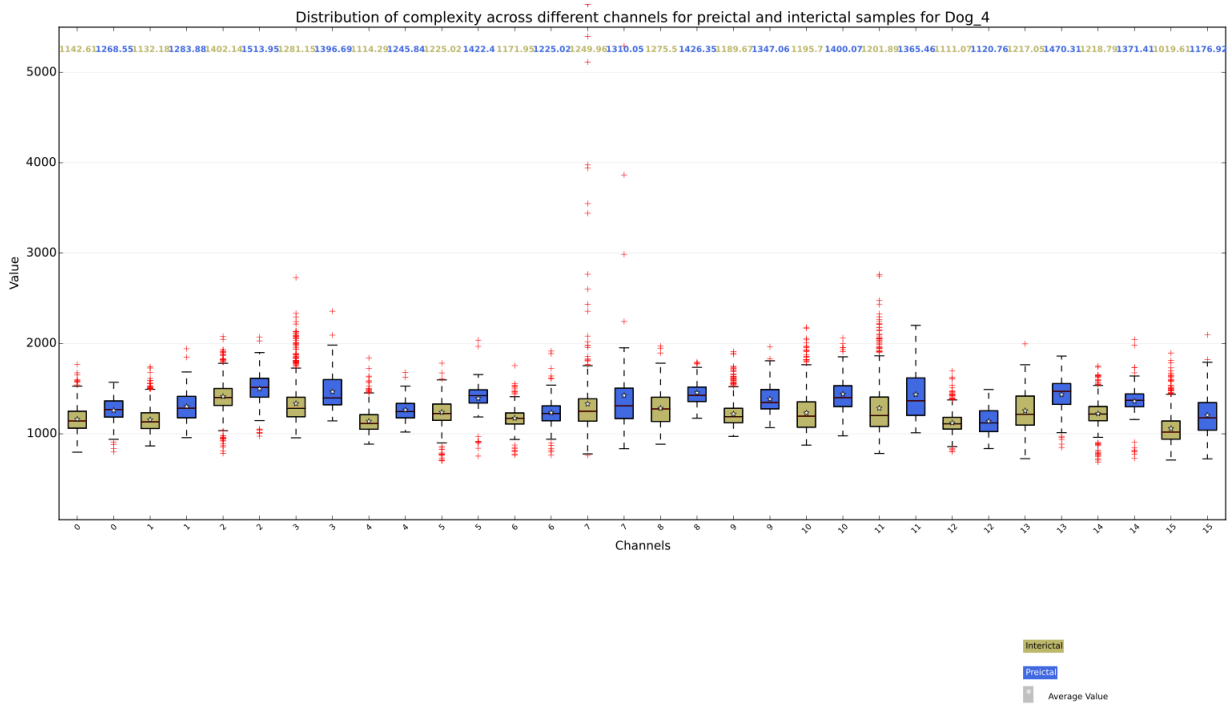



Distribution of complexity across different channels for preictal and interictal samples for Dog_2
Value
Channels
Interictal
Preictal
Average Value

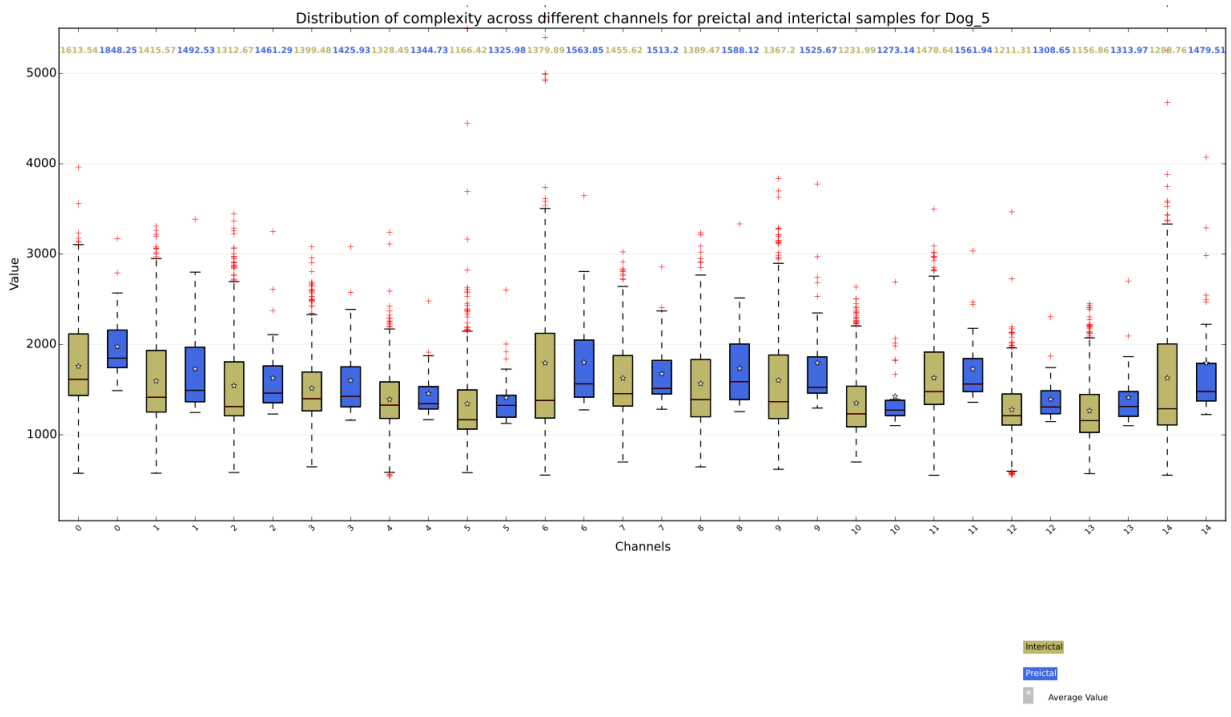
(b) Dog 2


Distribution of complexity across different channels for preictal and interictal samples for Dog_3
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of complexity across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

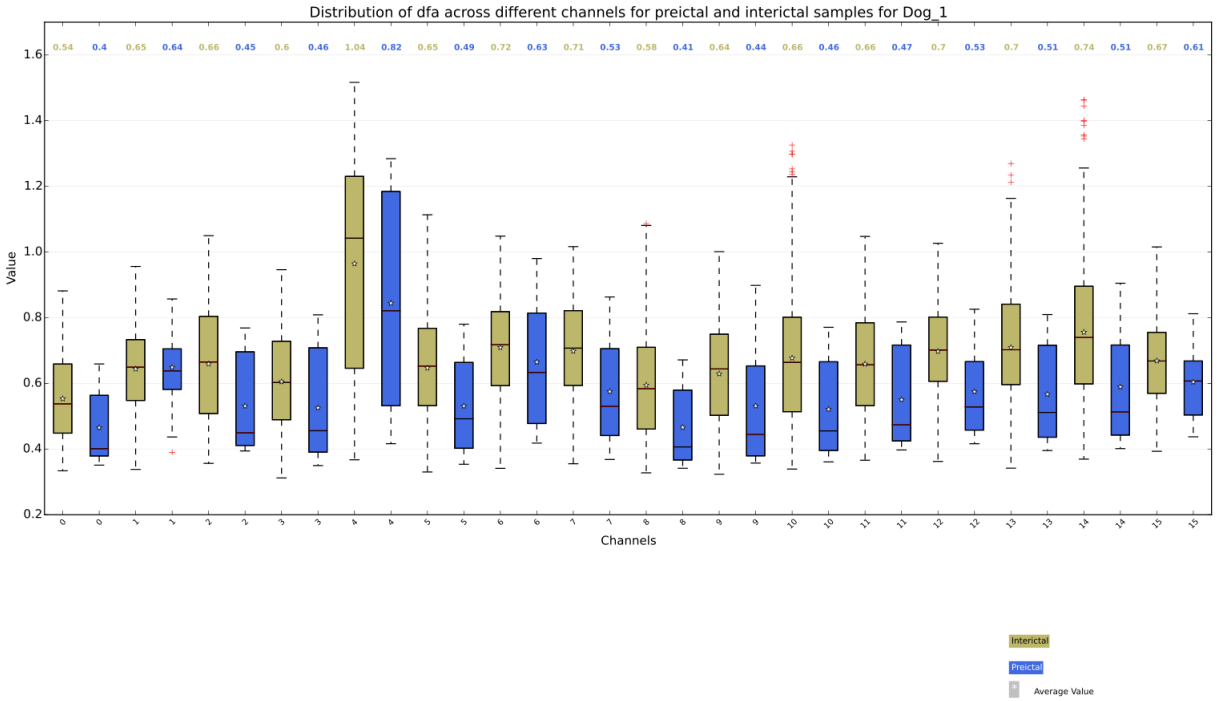

Distribution of complexity across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

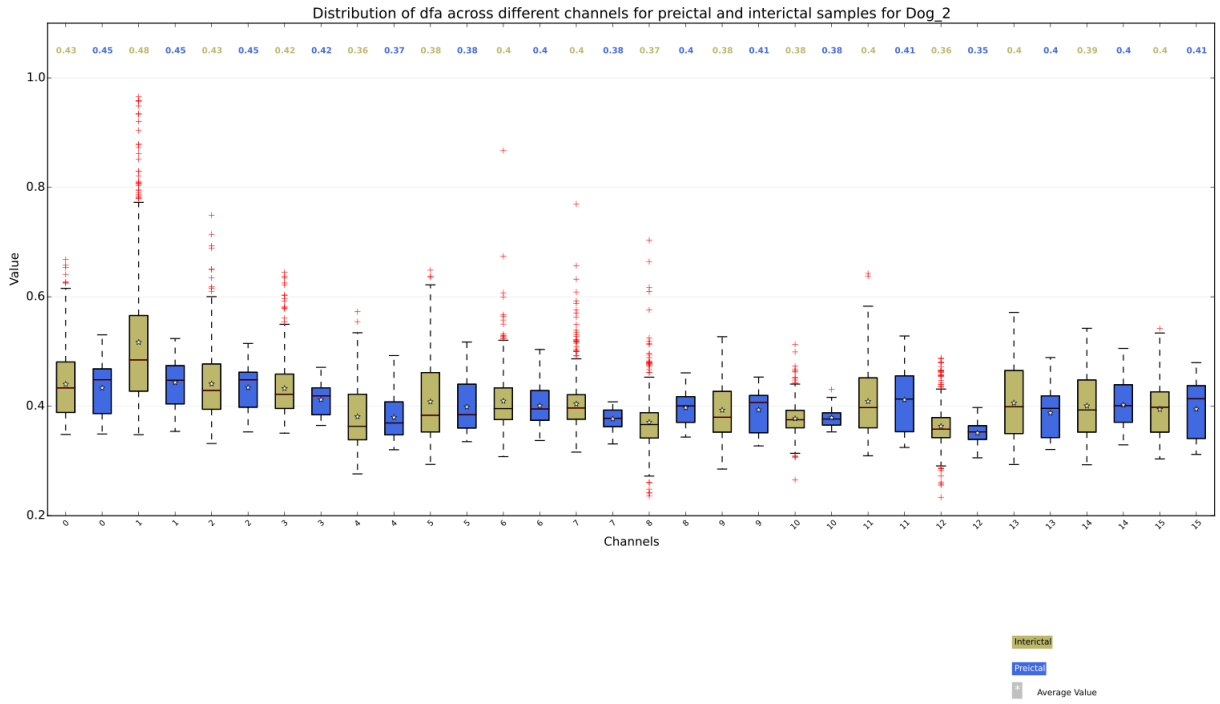
(e) Dog 5

A.7 Detrended Fluctuation Analysis

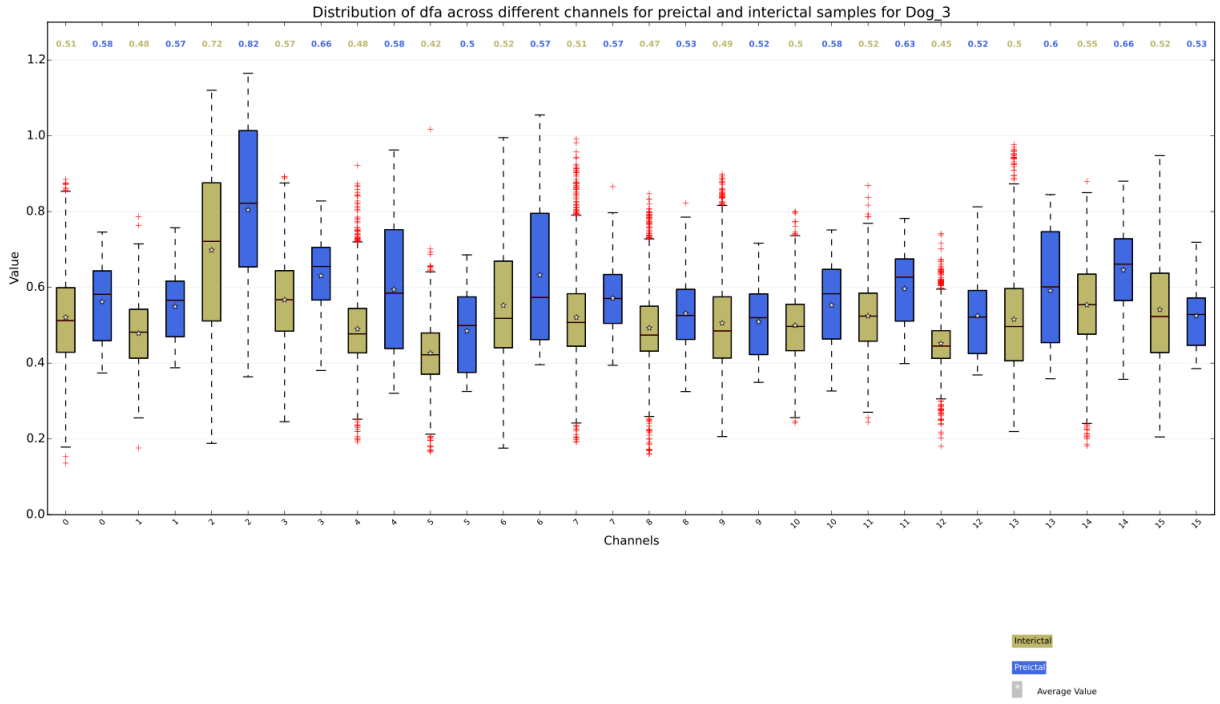
Figure A.10: Distribution of DFA across different subjects

(a) Dog 1

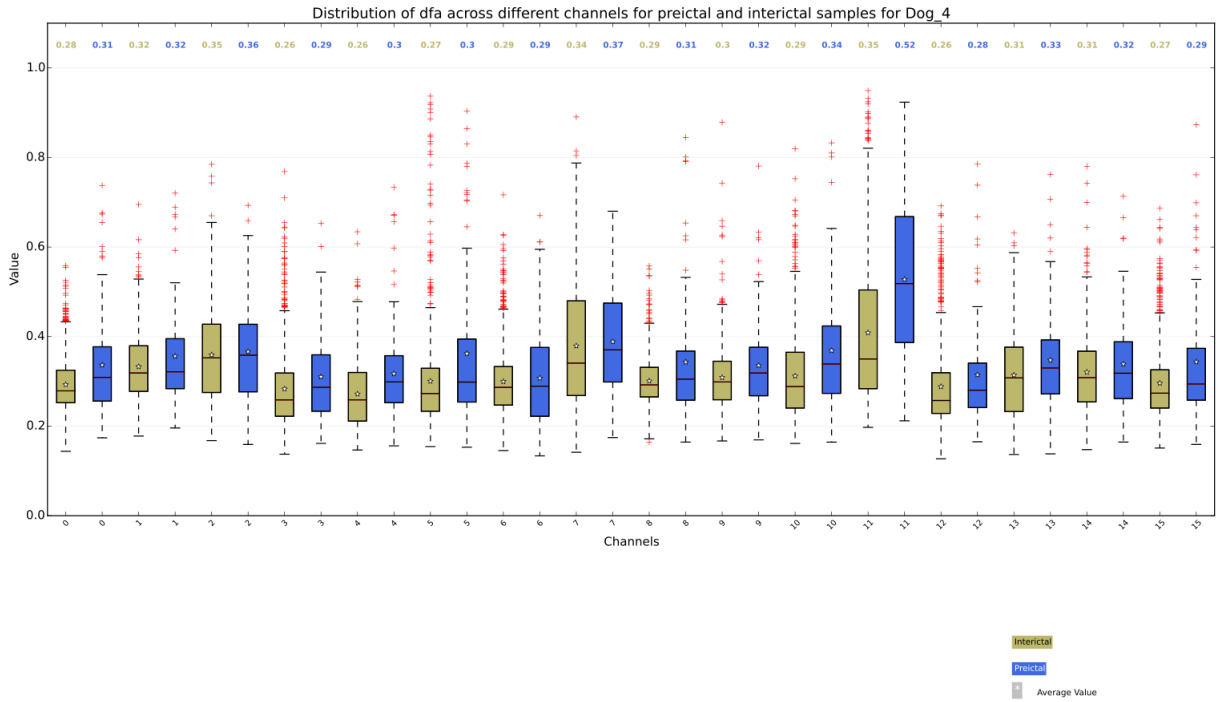



Distribution of dfa across different channels for preictal and interictal samples for Dog_2
0.43 0.45 0.48 0.45 0.43 0.45 0.42 0.42 0.36 0.37 0.38 0.38 0.4 0.4 0.4 0.38 0.37 0.4 0.38 0.41 0.38 0.38 0.4 0.41 0.36 0.35 0.4 0.4 0.39 0.4 0.4 0.41
Value
Channels
Interictal
Preictal
Average Value

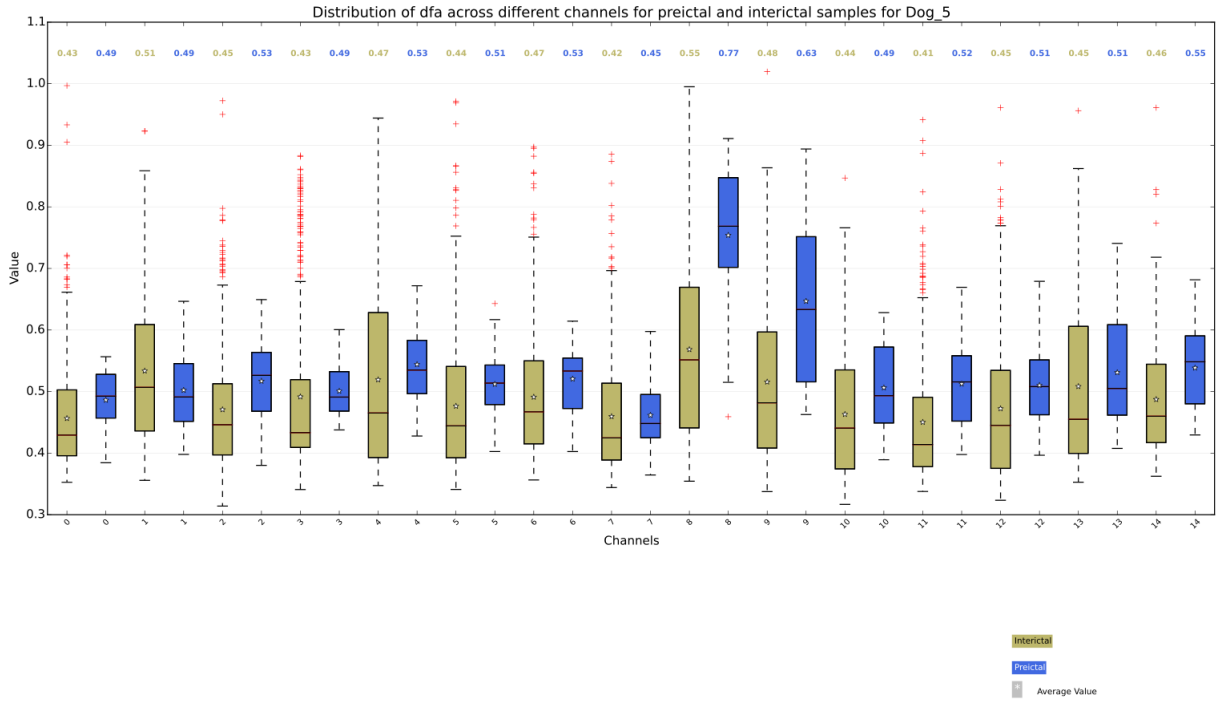
(b) Dog 2


Distribution of dfa across different channels for preictal and interictal samples for Dog_3
0.51 0.58 0.48 0.57 0.72 0.82 0.57 0.66 0.48 0.58 0.42 0.5 0.52 0.57 0.51 0.57 0.47 0.53 0.49 0.52 0.5 0.58 0.52 0.63 0.45 0.52 0.5 0.6 0.55 0.66 0.52 0.53
Value
Channels
Interictal
Preictal
Average Value

(c) Dog 3


Distribution of dfa across different channels for preictal and interictal samples for Dog_4
Value
Channels
Interictal
Preictal
Average Value

(d) Dog 4

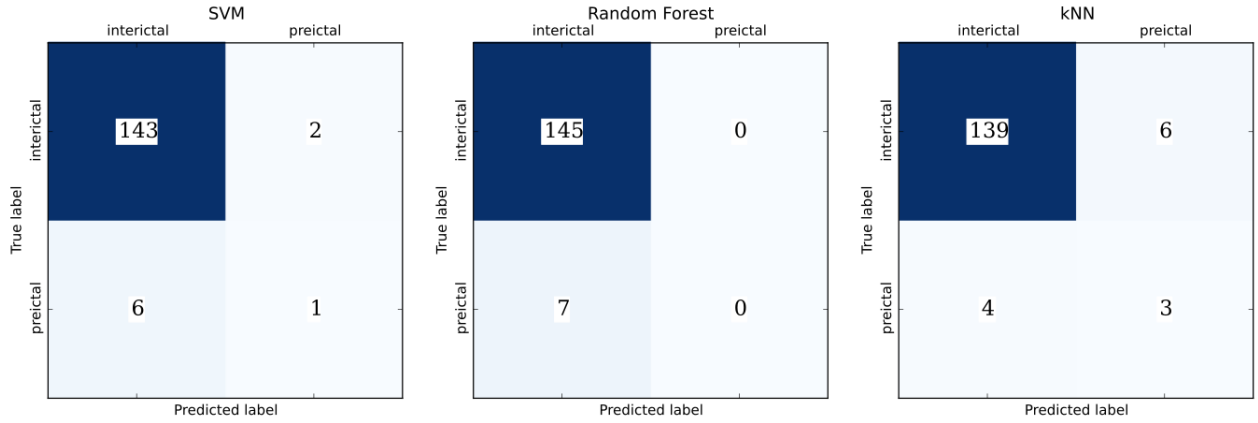

Distribution of dfa across different channels for preictal and interictal samples for Dog_5
Value
Channels
Interictal
Preictal
Average Value

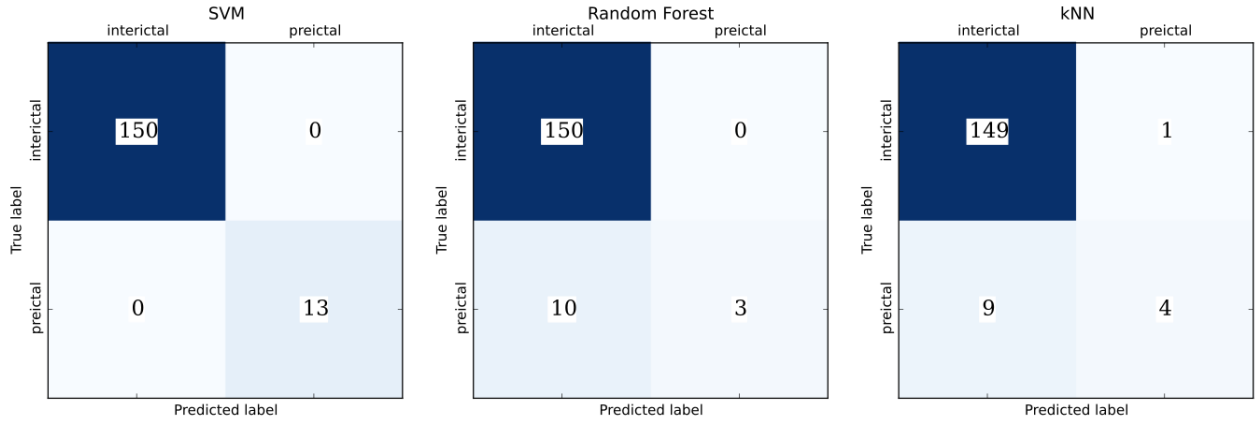
(e) Dog 5

B Confusion Matrix for Testing Data

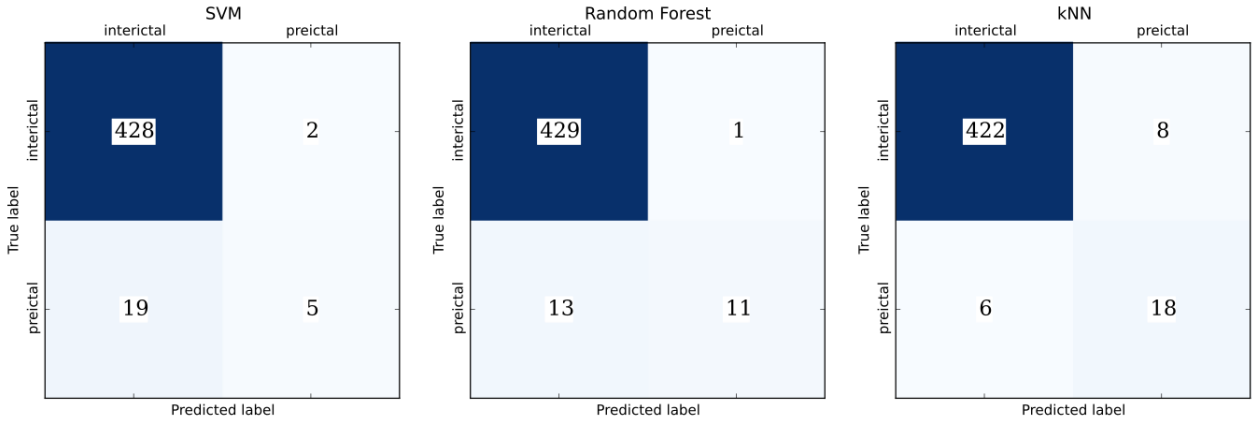
Figure B.1: Confusion matrix for test data across different algorithms

(a) Dog 1

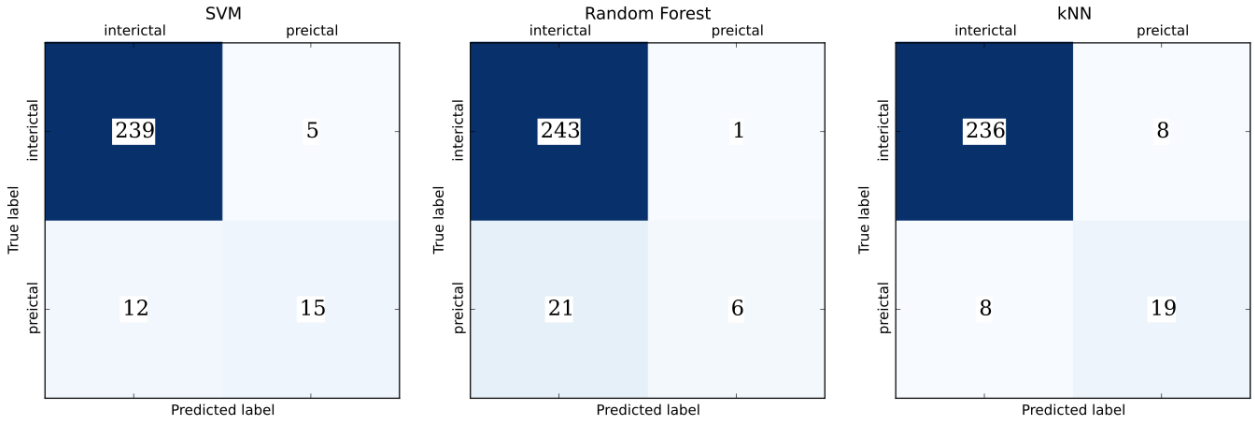




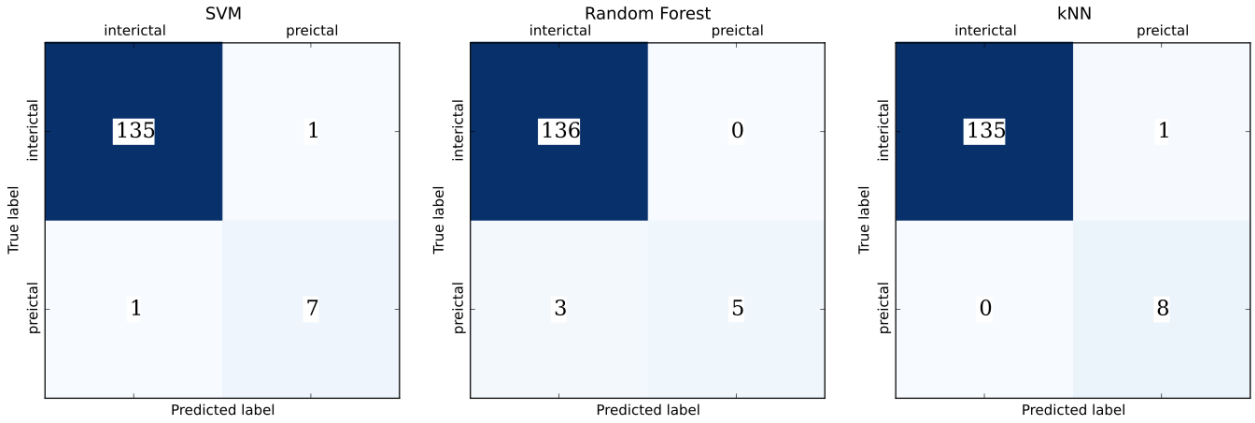
(b) Dog 2



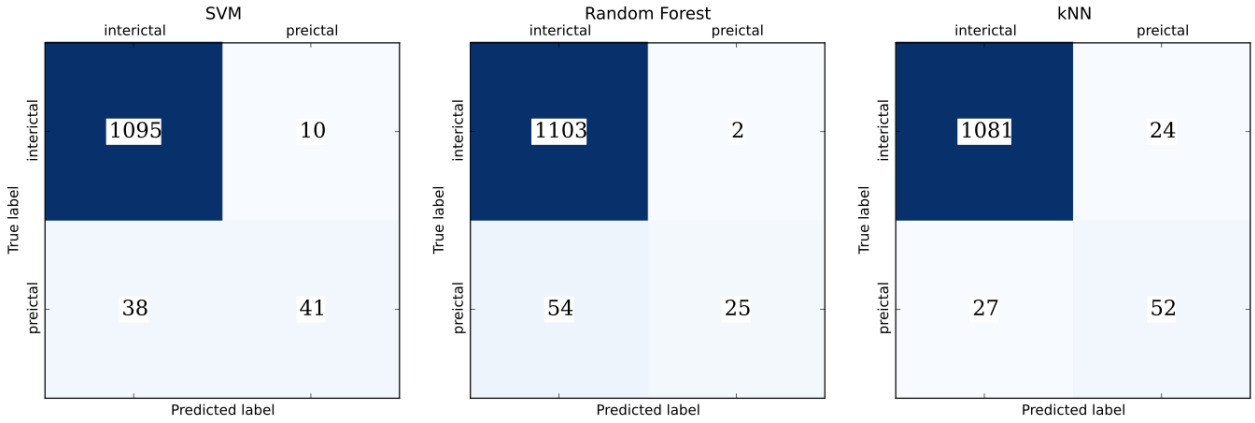
(c) Dog 3



(d) Dog 4



(e) Dog 5



(f) Overall

C Metric scores on testing data

Table C.1: Scores on test data at threshold=0.2

	Dog_1			Dog_2			Dog_3		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.91	0.94	0.93	0.96	0.99	0.94	0.98	0.96	0.97
F1	0.46	0.40	0.38	0.77	0.93	0.67	0.78	0.64	0.72
FPR	0.09	0.03	0.04	0.02	0.01	0.05	0.01	0.02	0.02
MCC	0.49	0.37	0.34	0.75	0.92	0.64	0.76	0.62	0.70
Precission	0.32	0.38	0.33	0.77	0.87	0.59	0.76	0.65	0.69
Recall	0.86	0.43	0.43	0.77	1.00	0.77	0.79	0.62	0.75
	Dog_4			Dog_5			overall		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.88	0.90	0.94	0.99	0.99	0.99	0.95	0.95	0.96
F1	0.57	0.59	0.70	0.93	0.89	0.94	0.66	0.67	0.69
FPR	0.11	0.09	0.03	0.00	0.01	0.01	0.05	0.03	0.03
MCC	0.54	0.55	0.67	0.93	0.89	0.94	0.65	0.65	0.67
Precission	0.44	0.49	0.70	1.00	0.80	0.89	0.56	0.61	0.66
Recall	0.81	0.74	0.70	0.88	1.00	1.00	0.81	0.75	0.73

Table C.2: Scores on test data at threshold=0.5

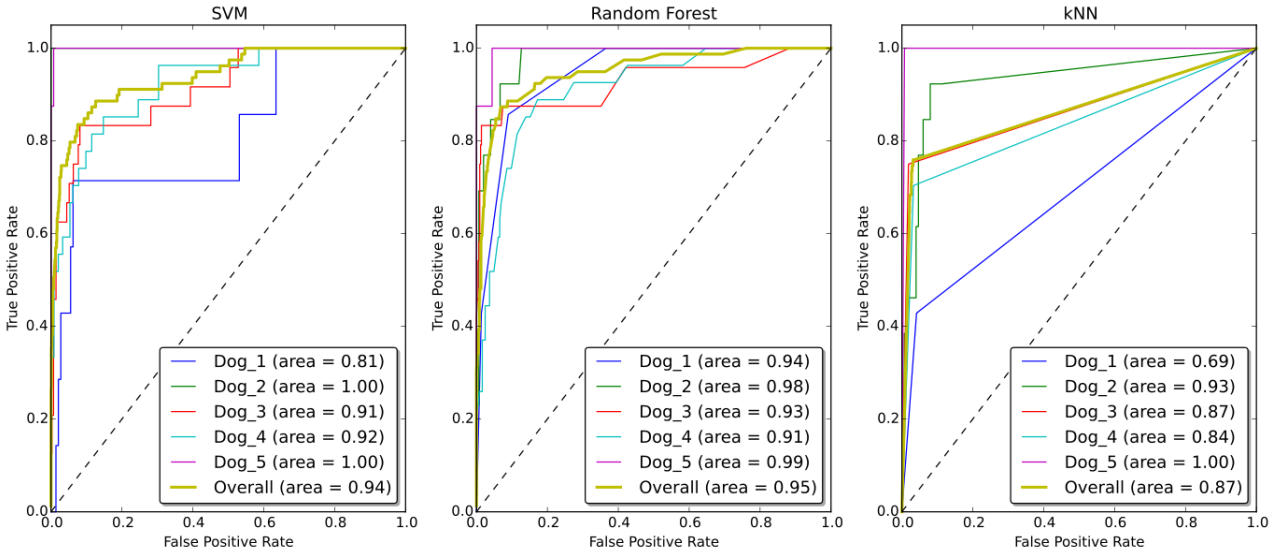
	Dog_1			Dog_2			Dog_3		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.95	0.95	0.93	0.94	1.00	0.94	0.97	0.95	0.97
F1	0.00	0.20	0.38	0.38	1.00	0.44	0.61	0.32	0.72
FPR	0.00	0.01	0.04	0.00	0.00	0.01	0.00	0.00	0.02
MCC	nan	0.19	0.34	0.47	1.00	0.47	0.64	0.37	0.70
Precission	0.00	0.33	0.33	1.00	1.00	0.80	0.92	0.71	0.69
Recall	0.00	0.14	0.43	0.23	1.00	0.31	0.46	0.21	0.75
	Dog_4			Dog_5			overall		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.92	0.94	0.94	0.98	0.99	0.99	0.95	0.96	0.96
F1	0.35	0.64	0.70	0.77	0.88	0.94	0.47	0.63	0.67
FPR	0.00	0.02	0.03	0.00	0.01	0.01	0.00	0.01	0.02
MCC	0.41	0.61	0.67	0.78	0.87	0.94	0.53	0.63	0.65
Precission	0.86	0.75	0.70	1.00	0.88	0.89	0.93	0.80	0.68
Recall	0.22	0.56	0.70	0.62	0.88	1.00	0.32	0.52	0.66

Table C.3: Scores on test data at threshold=0.8

	Dog_1			Dog_2			Dog_3		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.95	0.95	0.93	0.93	0.98	0.94	0.96	0.95	0.97
F1	0.00	0.00	0.38	0.14	0.82	0.38	0.40	0.08	0.72
FPR	0.00	0.01	0.04	0.00	0.00	0.00	0.00	0.00	0.02
MCC	nan	-0.02	0.34	0.27	0.82	0.47	0.49	0.20	0.70
Precission	0.00	0.00	0.33	1.00	1.00	1.00	1.00	1.00	0.69
Recall	0.00	0.00	0.43	0.08	0.69	0.23	0.25	0.04	0.75
	Dog_4			Dog_5			overall		
	Random Forest	SVM	kNN	Random Forest	SVM	kNN	Random Forest	SVM	kNN
Accuracy	0.90	0.93	0.94	0.94	0.99	0.99	0.94	0.95	0.96
F1	0.07	0.41	0.70	0.00	0.93	0.94	0.18	0.46	0.67
FPR	0.00	0.00	0.03	0.00	0.00	0.01	0.00	0.00	0.02
MCC	0.18	0.49	0.67	nan	0.93	0.94	0.31	0.53	0.64
Precission	1.00	1.00	0.70	0.00	1.00	0.89	1.00	0.96	0.69
Recall	0.04	0.26	0.70	0.00	0.88	1.00	0.10	0.30	0.65

D Receiver Operating Characteristic curves

Figure D.1: ROC curves for Test Data



E Precision-Recall curves

Figure E.1: PR curves for Test Data