Machine Learning Analysis of Neuroimaging (MRI) Data to Distinguish Individuals with Focal Cortical Dysplasia Type II

A Major Qualifying Project Report submitted to the faculty of

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the Degree of
Bachelor of Science

Authors:

Jonathan Golden

Vivek Kandasamy

Abstract

This project investigates the utilization of machine learning techniques for analyzing neuroimaging (MRI) data to differentiate patients diagnosed with Focal Cortical Dysplasia Type II (FCD Type II). FCD Type II presents challenges in accurate diagnosis, prompting the exploration of alternative approaches. The study involves preprocessing MRI data, extracting relevant features, and training various machine learning models for classification. Performance evaluation metrics are employed to assess the models' efficacy in distinguishing patients with FCD Type II from healthy individuals or those with other neurological conditions. The research aims to contribute to improved diagnosis and management of FCD Type II through the integration of machine learning analysis into neuroimaging practices.

Table of Contents

Acknowledgments

We extend our gratitude to the individuals who offered invaluable support and guidance during the completion of this MQP. We are thankful to our advisor Professor Benjamin Nephew for his unwavering support and constructive feedback that shaped this project. He consistently made himself available, demonstrating patience as we navigated the challenges of project completion. Additionally, we are appreciative of Professor Dmitry Korkin for being a co-advisor to us and giving us helpful tips when working on our project.

Chapter 1. Introduction

With the unprecedented growth of artificial intelligence (AI), its integration into almost every field is undeniable. The field of medicine is no different, and one promising avenue in which to apply the use of AI is in evaluating and diagnosing many different disorders. One such disorder is focal cortical dysplasia (FCD).

FCD is a neurological disorder characterized by various abnormal developments in the brain's cerebral cortex (Kabat & Król, 2012). FCD is typically split into three subcategories: focal cortical dysplasia Type I, Type II, and Type III respectively. FCD can have a variety of symptoms, including cognitive deficits, and motor impairments, but the main symptom of FCD is epilepsy and epilepsy-related seizures. Due to the severity of these symptoms, it is imperative that FCD be diagnosed early.

Traditional methods for diagnosing FCD rely on experts to inspect MRI brain scans against brain scans of cognitively normal individuals. However, this method is both time consuming and subjective, leading to FCD being underdiagnosed. AI, specifically machine learning, offers a potential solution to these issues.

The goal of our research is to develop a machine learning pipeline that could successfully classify magnetic resonance imaging (MRI) brain scans of individuals with FCD Type II when compared to cognitively normal individuals. This report documents our research, findings, and analysis of our investigation. Finally, we discuss the impact that research like this could have on the medical community as a whole.

Chapter 2. Background

2.1 Epilepsy

Epilepsy is a neurological disorder characterized by recurrent and unprovoked seizures, which are sudden and abnormal bursts of electrical activity in the brain. These seizures can manifest in various ways, ranging from momentary lapses of awareness to convulsions and loss of consciousness. Epilepsy affects people of all ages, and its causes are diverse, including genetic factors, brain injuries, infections, and structural abnormalities in the brain. The prevalence of epilepsy is significant globally, with millions of people living with the condition. It has a profound impact on individuals' quality of life, affecting not only their physical health but also their social and psychological well-being. (World Health Organization [WHO], 2024). The unpredictability of seizures can lead to limitations in daily activities, restrictions on driving and employment, and increased stigma associated with the condition.

One of the challenges in managing epilepsy is the considerable heterogeneity in its causes and manifestations. The identification and classification of various epilepsy syndromes have evolved over time, aiding in both diagnosis and treatment planning. Advancements in medical imaging techniques, particularly that of MRI, have played a crucial role in revealing structural abnormalities in the brain that may be associated with epileptic seizures (Garner et al., 2022).

2.2 Focal Cortical Dysplasia

Focal cortical dysplasia (FCD) is a common structural abnormality linked to epilepsy. It involves malformations in the development of the cerebral cortex, leading to localized areas of abnormal neuronal organization (Kim & Choi, 2019). FCDs are often challenging to detect through conventional MRI analysis, necessitating advanced tools such as machine learning algorithms to enhance diagnostic accuracy.

In recent years, there has been a growing emphasis on personalized medicine in epilepsy care, tailoring treatment plans based on an individual's specific characteristics and the underlying causes of their seizures. Advances in neuroimaging, genetics, and computational techniques offer new opportunities to unravel the complexities of epilepsy, leading to more targeted interventions and improved outcomes for those living with this challenging neurological disorder. The intersection of machine learning (ML) and magnetic resonance imaging (MRI) has emerged as a promising frontier in the understanding and management of epilepsy, particularly in the context of FCD. As discussed earlier, FCDs often present a diagnostic challenge, frequently eluding detection through conventional MRI analysis. This limitation has prompted the exploration of advanced technologies like ML to enhance the accuracy and efficiency of FCD detection.

2.3 Focal Cortical Dysplasia Type II: In Depth

As previously mentioned, FCD can be classified into various neuropathological subtypes. These subtypes are based on the types of malformation present in the cerebral cortex (Kim & Choi, 2019). Our dataset was on those with Focal Cortical Dysplasia Type II. In order to create an accurate machine learning model, it is important to first understand the unique properties of FCD Type II, so as to know what properties the model should consider important.

Focal Cortical Dysplasia Type II is a particularly severe form of cortical developmental malformations that significantly disrupts the normal neuronal organization. This disorder is categorized into two specific subtypes: FCD Type IIa and FCD Type IIb. Both subtypes are marked by the presence of dysmorphic neurons, which are neurons with abnormal size, shape, and orientation, disrupting the normal laminar pattern of the cortex. The distinguishing factor between FCD Type IIa and FCD Type IIb is the presence of balloon cells in FCD Type IIb, which are abnormal, large cells with an eosinophilic, glassy appearance (Blumcke et al., 2017).

The disruption caused by these abnormal cells in FCD Type II leads to significant alterations in the cortical lamination. This malformed cortical organization is crucial for understanding the clinical manifestations often associated with FCD Type II, particularly severe forms of epilepsy that are frequently resistant to pharmaceutical treatments. The presence of balloon cells, especially, is associated with more severe epileptic symptoms and often necessitates surgical intervention for management (Represa, 2019).

The pathological features of FCD Type II, such as dysmorphic neurons and balloon cells, disrupt the normal six-layered structure of the cortex, leading to the clinical challenges observed in affected individuals. These features are pivotal not only in diagnosis but also in determining the prognosis and potential treatment options for patients with this condition. Surgical outcomes, for instance, are heavily influenced by the precise localization and extent of the cortical dysplasia, which are typically assessed through comprehensive neuroimaging studies (Abbasi et al., 2019).

2.4 MRI Images

Magnetic Resonance Imaging (MRI) is a non-invasive imaging technology that produces detailed anatomical images of the human body, particularly useful in the visualization of soft tissues. This technology utilizes powerful magnets and radio waves to align the nuclear magnetization of hydrogen atoms in water within the body, and then uses radiofrequency fields to alter the alignment of this magnetization. The subsequent return to equilibrium emits a radio signal, which is captured by the scanner to create cross-sectional images of the body (Smith & Nichols, 2018).

Recent advances have further enhanced MRI technology, particularly through the use of high-field MRI scanners, such as 3 Tesla (3T) machines. These high-field scanners improve image clarity and detail, thereby facilitating more accurate diagnosis and better planning for potential surgical interventions in patients with FCD Type II. Additionally, the advent of functional MRI (fMRI) allows clinicians to observe brain activity in real time by detecting changes in blood flow associated with neuronal activity, thus adding a functional dimension to the morphological images (Li et al., 2021).

Despite its widespread use and benefits, MRI technology does come with limitations. The quality of MRI images can be degraded by motion, and certain patient populations, including those with specific types of medical implants, pacemakers, or claustrophobia, may face challenges with MRI scans. Furthermore, the high cost of MRI technology and the need for specialized facilities and trained personnel limit its accessibility in some settings (Smith & Nichols, 2018; Douglas et al., 2016).

MRI stands as a cornerstone of modern medical imaging, particularly in the diagnosis and management of FCD Type II. Its detailed visualization capabilities and ongoing advancements continue to make it a crucial tool in the neurological sciences, improving the safety and efficacy of medical diagnostics.

2.4 Machine Learning Algorithms in Medical Imaging

Machine learning algorithms, particularly those based on deep learning architectures like convolutional neural networks (CNNs), have shown remarkable capabilities in learning complex patterns from medical imaging data (Liu et al., 2022). In the case of epilepsy, leveraging ML on MRI datasets allows for the automated identification of the subtle structural abnormalities indicative of FCD. This not only addresses the limitations of human visual inspection but also paves the way for a more nuanced understanding of the intricacies associated with epileptic disorders. The utilization of ML in epilepsy imaging is driven by the need for robust and reliable tools to assist clinicians in early and accurate diagnosis. The integration of ML algorithms with MRI data offers the potential to uncover subtle abnormalities that might be overlooked in traditional analyses, contributing to a more comprehensive evaluation of the structural changes in the brain associated with epilepsy.

In this context, Focal Cortical Dysplasia Type II presents a significant challenge for diagnosis due to its subtle and often diffuse characteristics. FCD Type II is characterized by specific histopathological features, including dysmorphic neurons and, in more severe cases, balloon cells, which are crucial for its classification (Balestrini et al., 2023). These features disrupt the normal cortical layering and are associated with severe forms of epilepsy that are often resistant to medication. Machine learning models, particularly CNNs, have been effectively trained to detect these abnormalities by analyzing layers of MRI data to identify patterns not readily visible to the human eye.

Recent advancements in ML have focused on enhancing the sensitivity and specificity of these models to differentiate between FCD Type II and other types of cortical dysplasias. By training on large datasets of annotated MRI scans, these models learn to recognize the unique signatures of dysmorphic neurons and balloon cells. This capability is crucial for neurologists and neurosurgeons, as accurate identification of FCD Type II can significantly influence treatment decisions, including the need for surgical intervention. The use of ML in medical imaging extends beyond diagnosis to include predictive analytics, potentially forecasting patient outcomes based on the identified imaging features. This aspect of machine learning is particularly valuable in planning treatment strategies and in personalized medicine, where tailored interventions can be designed based on predicted disease progression and response to treatment (Lee et al., 2019).

The integration of machine learning algorithms with medical imaging, particularly in the diagnosis and management of epilepsy due to FCD Type II, represents a transformative advancement in medical technology. These tools not only enhance diagnostic accuracy but also offer new insights into the disease mechanisms, supporting clinicians in providing more targeted and effective treatments.

Chapter 3. Methodology

3.1 Neuroimaging Data

The use of data with known output parameters (labeled data) is required by machine learning algorithms to train a model for a specific purpose. The OpenNeuro dataset, which includes MRI data from individuals diagnosed with epilepsy due to Focal Cortical Dysplasia Type II as well as healthy controls, was utilized as a resource for training and validating machine learning models (OpenNeuro Dataset ds004199, 2023). The inclusion of T1 and FLAIR weighted images, along with manually labeled lesion masks and clinical features, provides a rich and diverse set of information from which the algorithms can learn. This dataset is aimed at catalyzing advancements in computer-aided FCD detection, enabling the validation of existing algorithms and the development of novel approaches that can enhance diagnostic accuracy. By harnessing the power of machine learning in conjunction with MRI, the goal to improve FCD detection is possible.

3.2 Preprocessing and Feature Extraction

Quantitative values must be translated from MRI images for use in training ML algorithms. This process, known as preprocessing, involves the identification and computation of relevant measurements from each image. FastSurfer has been chosen as the tool for preprocessing due to its exceptional performance in brain MRI analysis. It is recognized as a sophisticated neuroimaging pipeline optimized for the analysis of structural human brain MRIs (Henschel et al., 2020). Various convolutional neural networks are integrated by it to process different MRI orientations, thereby enhancing its segmentation accuracy. By incorporating deep learning models, tasks traditionally requiring extensive manual labor and computational resources are automated. Its efficiency and precision provide significant improvements over conventional methods, delivering outputs that meet the standards of similar applications, such as FreeSurfer, but in a reduced timeframe.

High-quality T1-weighted MRI images, preferably obtained with a 3T scanner, are required by FastSurfer. The input images must meet specific dimension and quality standards akin to those required by FreeSurfer, with isotropic voxel sizes ideally between 1mm and 0.7mm. By utilizing the `run_fastsurfer.sh` script, FastSurfer is executed with various flags to define the MRI data, processing options, and output settings. Segmentation and surface reconstruction are also performed, with adjustments based on image acquisition parameters like 3T imaging. FastSurfer's segmentation employs the FastSurferCNN, a deep neural network that processes MRI data through multiple pathways, capturing a comprehensive representation of brain structures. This method produces highly reliable segmentation outcomes validated against standard datasets.

3.3 Output Files from FastSurfer

After segmentation, various quantitative data points such as volumetric measurements and cortical thickness are extracted by FastSurfer. These metrics are essential for the project and aid in the development of a machine learning model to identify patients with Focal Cortical Dysplasia Type II. Several output files are generated by FastSurfer, each providing valuable neuroanatomical insights crucial for subsequent analysis. The aseg+DKT file, which contains summary statistics for volume estimates per anatomical structure, encompasses both cortical and subcortical segmentation statistics. Similarly, summary statistics for surface area and thickness estimates per anatomical structure of cortical parcellation statistics are presented in the aparc.DKTatlas.mapped file, with mapping from ASEGDKT segmentation to the surface (Reuter et al., 2010).

Additionally, the BA_exvivo.stats and BA_exvivo.thresh.stats files are produced by FastSurfer, entailing statistical analyses performed at high resolution, particularly for Broadmann ex vivo studies (Desikan et al., 2006; Fischl et al., 2002). These files offer detailed insights, with the BA_exvivo.thresh.stats applying a threshold to the analysis. Moreover, a table of curvature statistics, which provides information on surface curvature characteristics, is contained in the curv.stats file. The w-g.pct.stats file presents the Gray to White Matter signal intensity ratio with a surface overlay, aiding in the visualization of structural differences. Lastly, insights into white matter structures are provided in the wmparc.DKTatlas.mapped.stats file, which offers a table of white matter segmentation statistics (Reuter et al., 2010).

3.4 Organizing Output Data into Tabular Form

FastSurfer produces output in several files for each subject. These need to be collated into a single tabular representation which can be used for ML model training. ChatGPT was used to assist in this process, through a process known as prompt engineering. This method involves crafting and refining specific instructions, or prompts, which guide the creation of a Python script capable of executing the desired task. The target scenario involves multiple subjects, each with their own `.stats` file containing valuable measurements.

The prompt engineering process was initiated with a precise identification of the required statistics—names such as 'NVertices', 'Area_mm2', and 'Mean', among others. For each subject, the path to their respective file was dynamically generated using a for-loop, accounting for the range of subject numbers. Within the Python script, each of these paths was programmed to be visited in turn, thus opening the door to the rich data within. The script was formulated to read through the contents of each file, ensuring the careful extraction of numerical treasures line by line. Special attention was paid to the format and spacing within the files, as these aspects can vary as widely as the data they contain. Regular expressions, a stalwart tool in text parsing, were utilized to ensure that even amidst inconsistent white space or text, the statistics were accurately captured.

As the script traversed the list of files, it gathered the statistics into a comprehensive dictionary for each subject. This dictionary held keys corresponding to the data points of interest, each prefixed with 'left' to signify their anatomical origin. With the data collated, the script then proceeded to transfer it into a CSV file—a tabular expanse prepared to receive the influx of information. Each row was dedicated to a subject, with their name enshrined in the first column, followed by the columns of statistics, standing in orderly fashion like columns in a Grecian temple (Appendix A).

3.5 Data Cleaning

Prior to conducting any analysis, a thorough data cleaning process was performed on the raw neuroimaging data to ensure the integrity and reliability of subsequent analyses. This process comprised two main steps: the elimination of subjects with missing '.stats' files from the FastSurfer output and the handling of missing values within the remaining dataset. Initially, two subjects were identified as having missing '.stats' files, resulting in their exclusion from the dataset to maintain data consistency. Subsequently, the remaining dataset was scrutinized for any instances of missing values.

In cases where missing values were detected, they were imputed conservatively with a value of 0. This method assumes that missing values signify absence rather than unknown or unrecorded data, ensuring the inclusion of all available information in subsequent analyses while minimizing potential biases. By adhering to these rigorous data cleaning procedures, the aim was to enhance the robustness and reliability of the analyses, thereby facilitating meaningful insights from the neuroimaging data (Appendix B).

3.6 Python and Jupyter Notebook

For the development of machine learning models in this study, the Python programming language was selected as the primary coding platform due to its versatility, extensive libraries, and robust ecosystem for machine learning and data analysis tasks. Python is endowed with a wide range of libraries such as scikit-learn, TensorFlow, and Keras, which provide efficient implementations of various machine learning algorithms and neural network architectures, enabling streamlined development and experimentation (Abadi et al., 2016). Additionally, collaborative coding efforts are facilitated and rapid prototyping of machine learning pipelines is aided by Python's simplicity and readability (Igual et al., 2017).

In conjunction with Python, Jupyter Notebook was chosen as the integrated development environment (IDE) for coding and experimentation. An interactive computing environment is provided by Jupyter Notebook, allowing for the creation of executable documents containing code, visualizations, and explanatory text, fostering reproducible research practices and enhancing the documentation of model development workflows (Kluyver et al., 2016). The combination of Python and Jupyter Notebook offers a user-friendly interface for iteratively exploring data, building machine learning models, and visualizing results, thereby facilitating a seamless and efficient development process.

3.7 Scikit-learn

For the implementation of machine learning models in this study, scikit-learn, a powerful Python library, was selected as the primary toolset. A comprehensive collection of machine learning algorithms and utilities that facilitate efficient development, training, and evaluation of models is offered by scikit-learn (Pedregosa et al., 2011). Its user-friendly interface, extensive documentation, and emphasis on code readability make it an ideal choice for both novice and experienced practitioners in the field of machine learning (Appendix C). Additionally, seamless integration with other Python libraries such as NumPy and pandas is facilitated, enabling easy data manipulation and preprocessing tasks required for model development (Van Der Walt et al., 2011). Support for various machine learning tasks including classification, regression, clustering, and dimensionality reduction is provided by the library, thereby offering a versatile platform for addressing diverse research questions and experimental designs (Pedregosa et al., 2011).

3.8 Choosing Machine Learning Metrics for Classififcation

The selection of appropriate evaluation metrics was pivotal in assessing the performance of machine learning models for classification of FCD Type II. A comprehensive set of metrics including accuracy, recall, F1 score, and area under the receiver operating characteristic curve (AUC-ROC) was utilized to evaluate the efficacy of the classification models in this study.

Accuracy, which measures the proportion of correctly classified instances over the total number of instances, provides an overall assessment of model performance but may not be optimal for imbalanced datasets (Sokolova, 2011).

Recall, also known as sensitivity, quantifies the model's ability to correctly identify positive instances from the total number of actual positive instances, making it crucial in medical diagnosis tasks where missing a positive case (false negative) can be consequential.

The F1 score, being the harmonic mean of precision and recall, balances both precision and recall and is especially useful for imbalanced datasets.

Precision, commonly used in classification tasks, was not chosen for this study due to the nature of the problem. Since missing a case of Focal Dysplasia Type 2 (FD2) is more concerning than false positives in the task of distinguishing FD2 patients from a control group using neuroimaging data, metrics like recall and F1 score, prioritizing the minimization of false negatives, are more suitable.

Additionally, the AUC-ROC metric evaluates the model's performance across various threshold settings, indicating its discrimination capability (Powers, 2020). This selection of metrics aligns with best practices in machine learning evaluation, ensuring a comprehensive assessment of model performance across different aspects of classification accuracy and robustness.

Chapter 4. Results

The outcomes of employing machine learning techniques for the analysis of neuroimaging data, specifically magnetic resonance imaging (MRI), to discern patients diagnosed with Focal Dysplasia Type 2 from a control group were recorded. Through a careful evaluation of various machine learning models and feature selection methods vs test size, we assess the efficacy of our approach in accurately classifying individuals with FCD Type II.

4.1 Machine Learning Model #1: Naive Bayes

In the assessment of a Naive Bayes classifier's performance, we monitored the classifier across different test set sizes (Figure 1).

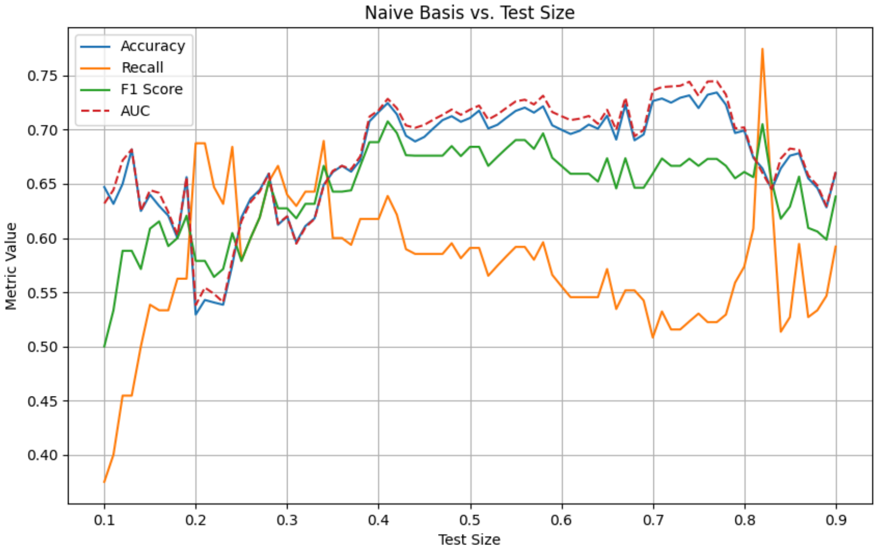


Figure 1: Plot of the performance metrics of Naive Bayes as a function of the test size

The stability of the classifier's accuracy, with values averaging around 70%, regardless of the proportion of data used for testing, was observed. This consistency suggests that the classifier maintains its ability to correctly identify outcomes across various test sizes. However, significant variability in the recall metric, critical for the identification of true positives, was noted with larger test sizes. This fluctuation points to a potential sensitivity of the model to the composition of the test data, which could be critical for applications where the cost of a false negative is high. Variability was also observed in the F1 score, which generally trended lower than accuracy. The AUC remained above 0.7 for most test sizes, but the metric exhibited some peaks and dips, signaling that certain test sizes or data compositions might influence the classifier's discriminative capabilities. Anomalous results were observed at a test size of approximately 0.8, where recall spiked and AUC dipped simultaneously. This anomaly may reveal an area where the model's performance diverges significantly from the norm and warrants further investigation to understand the underlying factors contributing to this behavior.

The general effectiveness of the Naive Bayes classifier was affirmed by its stable accuracy across diverse test sizes. However, the notable variability observed in the recall and F1 score metrics highlights areas for potential improvement. While the classifier performs reliably well in a general context, its ability to balance the precision-recall trade-off could be optimized for more specific applications, particularly those where the correct identification of all positive cases is crucial. Moving forward, strategic adjustments such as fine-tuning the decision threshold, addressing any dataset imbalances, or modifying the model's hyperparameters may enhance its performance, particularly in the areas indicated by the volatility in the recall and F1 scores.

4.2 Machine Learning Model #2: Decision Tree

In the evaluation of the Decision Tree model's predictive capabilities, performance metrics were plotted against various test set sizes to observe trends and stability (Figure 2).

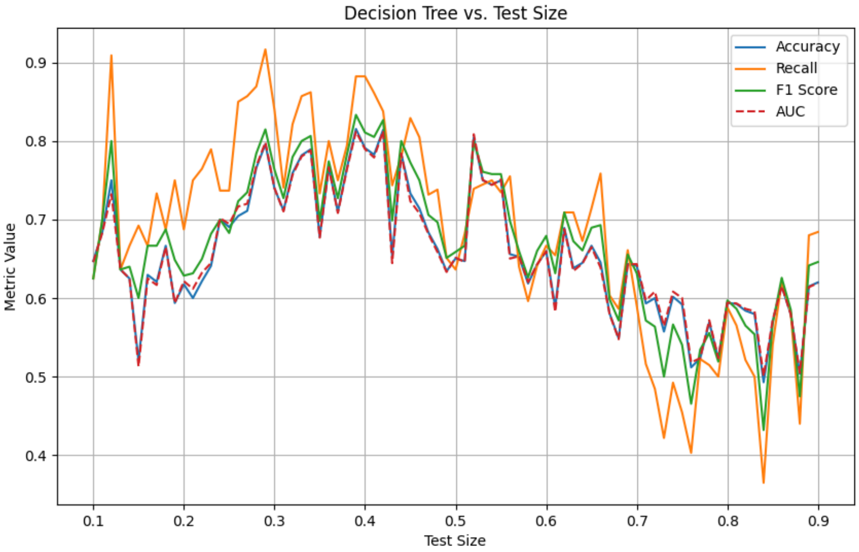


Figure 2: Plot of the performance metrics of Decision Tree as a function of the test size

The accuracy of the Decision Tree model exhibited variability, yet maintained a generally high performance, peaking at just below 90% and rarely falling below 60%. This indicates that while the model's predictions are mostly correct, there are instances where the model's performance can deviate significantly. The fluctuations might be attributed to the Decision Tree's sensitivity to specific splits in the data or to the variance within the test sets. Recall showed a similar trend to accuracy, with occasional spikes that suggest certain configurations of the test data might be particularly well-suited or ill-suited to the model's method of classification. This metric is particularly sensitive to how the model identifies true positives and its performance could be reflective of the Decision Tree's ability to generalize from the training data to unseen instances. The F1 score aligned closely with recall, indicating that the balance between precision and recall is somewhat proportional. The peaks in the F1 score typically correspond with the peaks in recall, reinforcing the model's strength in certain test set compositions. However, the dips in the F1 score suggest there are areas where the model might be improved, particularly in its ability to maintain a balance between the precision and recall across diverse test sizes. The AUC remained fairly stable, with most values around the 0.7 to 0.8 range, demonstrating a consistent discriminative ability for the model between the classes. The occasional troughs in AUC point to test set sizes where the model may have difficulty distinguishing between classes as effectively.

Overall, the Decision Tree model demonstrated a robust ability to classify correctly across varied test sizes. However, the observed variability in performance metrics suggests that the model may benefit from further tuning. Strategies to address the fluctuating accuracy and recall might include pruning the tree to prevent overfitting, adjusting the depth of the tree, or incorporating ensemble methods to stabilize predictions. The relatively steady AUC implies that the model has a solid foundation for distinguishing classes, but fine-tuning could enhance its performance, particularly in the areas where the other metrics fluctuate. The results provide valuable directions for future optimization to increase the model's reliability and predictive accuracy.

4.3 Machine Learning Model #3: Logistic Regression

The evaluation of the Logistic Regression model's performance over varying test sizes was conducted to determine how well the model generalizes to new data and how its performance metrics are impacted by the size of the test dataset (Figure 3).

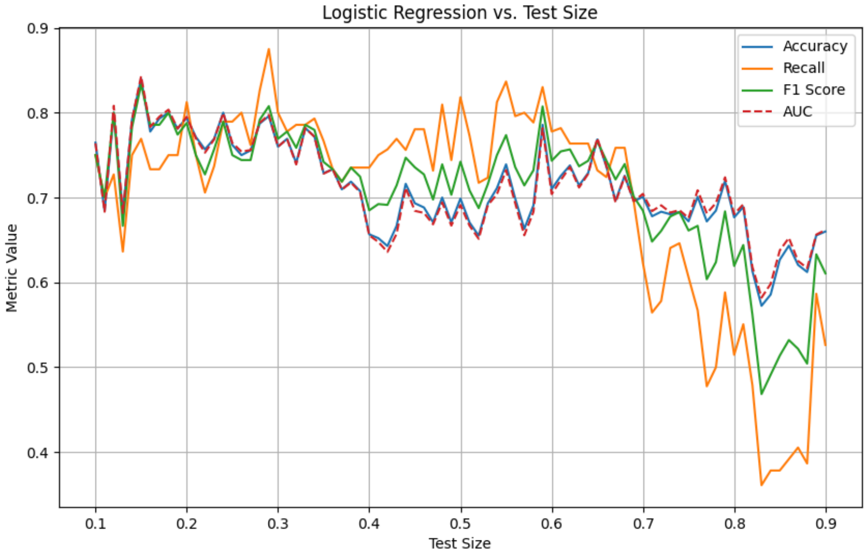


Figure 3: Plot of the performance metrics of Logistic Regression as a function of the test size

Accuracy fluctuated across test sizes, with some peaks exceeding 80%. This suggests that while Logistic Regression can achieve high levels of accuracy, its performance is not uniformly stable across different test set proportions. The recall metric also demonstrated variability but showed a tendency to track closely with accuracy. Peaks in recall corresponded to peaks in accuracy, denoting instances where the model was particularly adept at identifying true positives. The F1 score mirrored the trends of these metrics, suggesting that as the model's ability to detect true positives increased, so did its precision in not misclassifying negative instances. However, the F1 score's oscillations indicate that there are test conditions under which the model's balance of precision and recall can be optimized. AUC exhibited less dramatic fluctuations than the other metrics, maintaining values above 0.6 for most test sizes. This indicates a reasonably consistent ability of the model to discriminate between the classes across various test sizes.

Despite some inconsistencies, the Logistic Regression model generally exhibited good performance, with the capability to reach high levels of accuracy and recall in certain conditions. The relatively steady AUC suggests that the model has a solid foundation for distinguishing between classes.

4.4 Machine Learning Model #4: Random Forest

In the examination of the Random Forest classifier, its performance was analyzed through metrics against a range of test sizes to determine the model's predictive performance (Figure 4).

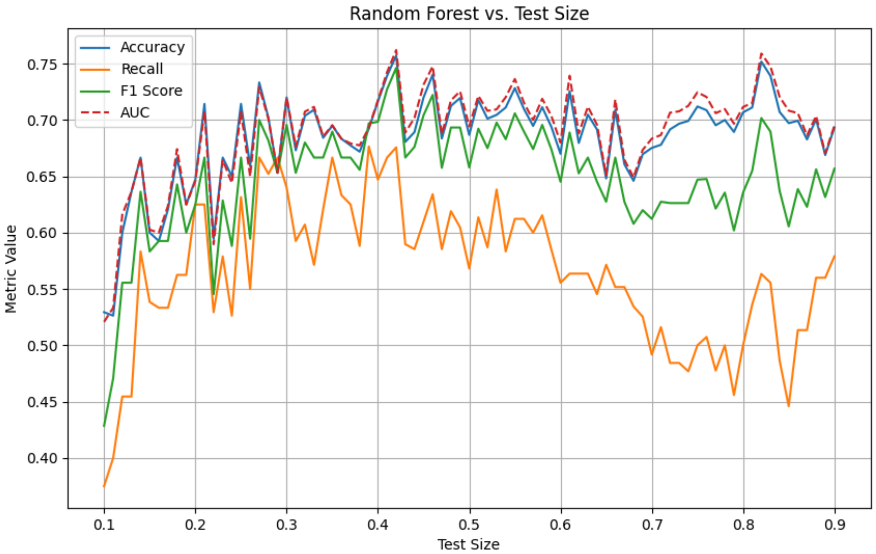


Figure 4: Plot of the performance metrics of Random Forest as a function of the test size

The classifier’s accuracy displayed commendable consistency, mostly staying within the 70-75% range across the spectrum of test sizes. This suggests a stable performance, indicative of the model's general aptitude in producing correct predictions. The slight undulations in accuracy are within an acceptable range, hinting at the model's resilience to changes in the test set size. Recall, a measure critical for the identification of all positive instances, revealed more pronounced fluctuations. This could imply that while the Random Forest classifier is generally reliable, its performance in identifying all relevant cases is more sensitive to the composition of the test data. The F1 score demonstrated a similar trend to recall, suggesting that the balance between precision and recall is subject to variation. However, the F1 score remained within a moderate band, indicative of a generally harmonious balance between the precision and recall of the model. AUC, indicative of the classifier's ability to differentiate between classes, remained relatively stable, but with a declining trend as the test size increased. This suggests that while the Random Forest classifier is generally good at distinguishing between classes, there might be a slight decrease in this ability as the proportion of the data used for testing grows.

The performance of the Random Forest classifier across varying test sizes reflects its strength as a robust and generally reliable predictive model. However, the observed fluctuations, particularly in recall and F1 score, point towards the possibility of further fine-tuning to enhance the model's sensitivity and precision-recall balance. Future directions could include exploring feature selection, hyperparameter optimization, and ensemble strategies to mitigate the variability and improve the classifier’s performance, especially for larger test sets where a slight decline in discriminative power was noted. The goal would be to not only maintain high levels of accuracy but also to ensure consistent performance across all metrics irrespective of the test size.

4.5 Machine Learning Model #5: Support Vector Machine

In evaluating the Support Vector Machine (SVM) classifier, results elucidate the model's predictive stability and accuracy across different proportions of data allocated for testing (Figure 5).

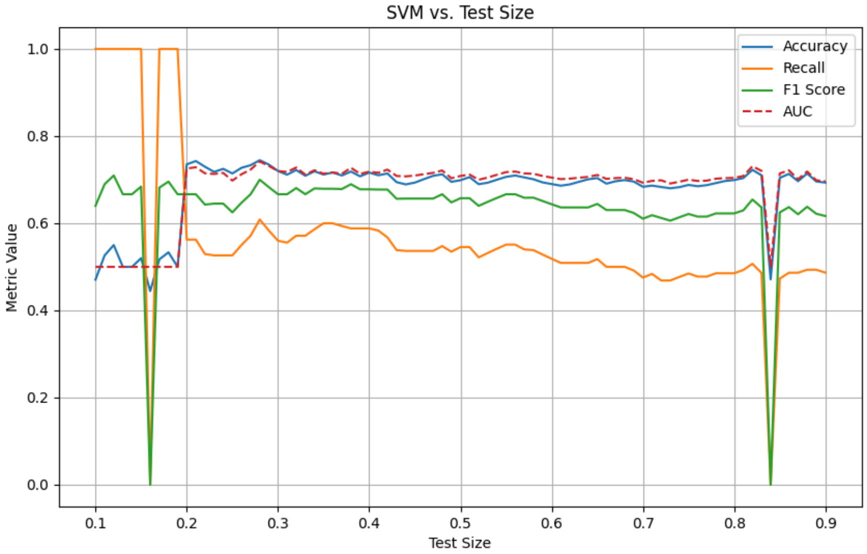


Figure 5: Plot of the performance metrics of SVM as a function of the test size

The accuracy metric for the SVM remained fairly consistent, with most values residing in the upper 60% to lower 70% range, suggesting a decent baseline of predictive correctness that is not heavily influenced by the size of the test data. Despite a few abrupt variations, which likely stem from anomalies or particularities in the test splits, the model's accuracy indicates a solid level of stability. Recall displayed some degree of variability, which is not uncommon for SVMs due to their sensitivity to the choice of kernel and regularization parameters. However, outside of the pronounced spikes, the recall tended to level off, suggesting that, in general, the SVM classifier could reliably identify a majority of the positive cases. The F1 score, oscillating in tandem with recall, suggests a consistent balance between precision and recall was maintained. While there are points where the F1 score dips, indicating moments where the model could neither precisely identify true positives nor avoid false positives, the score mostly trends in a stable band, implying a satisfactory balance in most cases. AUC, representing the model's ability to discriminate between classes, shows an overall consistency with a few notable exceptions. This consistent performance, particularly in the range of 0.6 to 0.7, underscores the SVM's robustness in distinguishing class labels across varied test sizes.

The Support Vector Machine classifier has proven to be quite effective in terms of accuracy, demonstrating a commendable level of consistency across various test sizes. However, the lower recall observed may be a cause for concern in contexts where missing out on positive cases carries a significant penalty. This could be particularly critical in domains like medical diagnosis or fraud detection, where failing to identify true positives could have serious consequences.

4.6 Machine Learning Model #6: XGBoost

Finally, the XGBoost model performance was examined through performance metrics against a range of test sizes to determine the model's predictive performance (Figure 6).

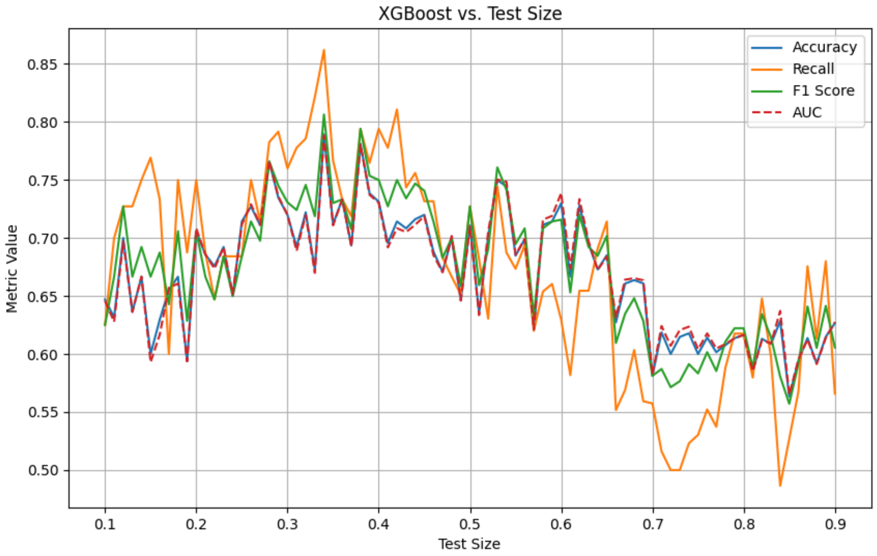


Figure 6: Plot of the performance metrics of XGBoost as a function of the test size

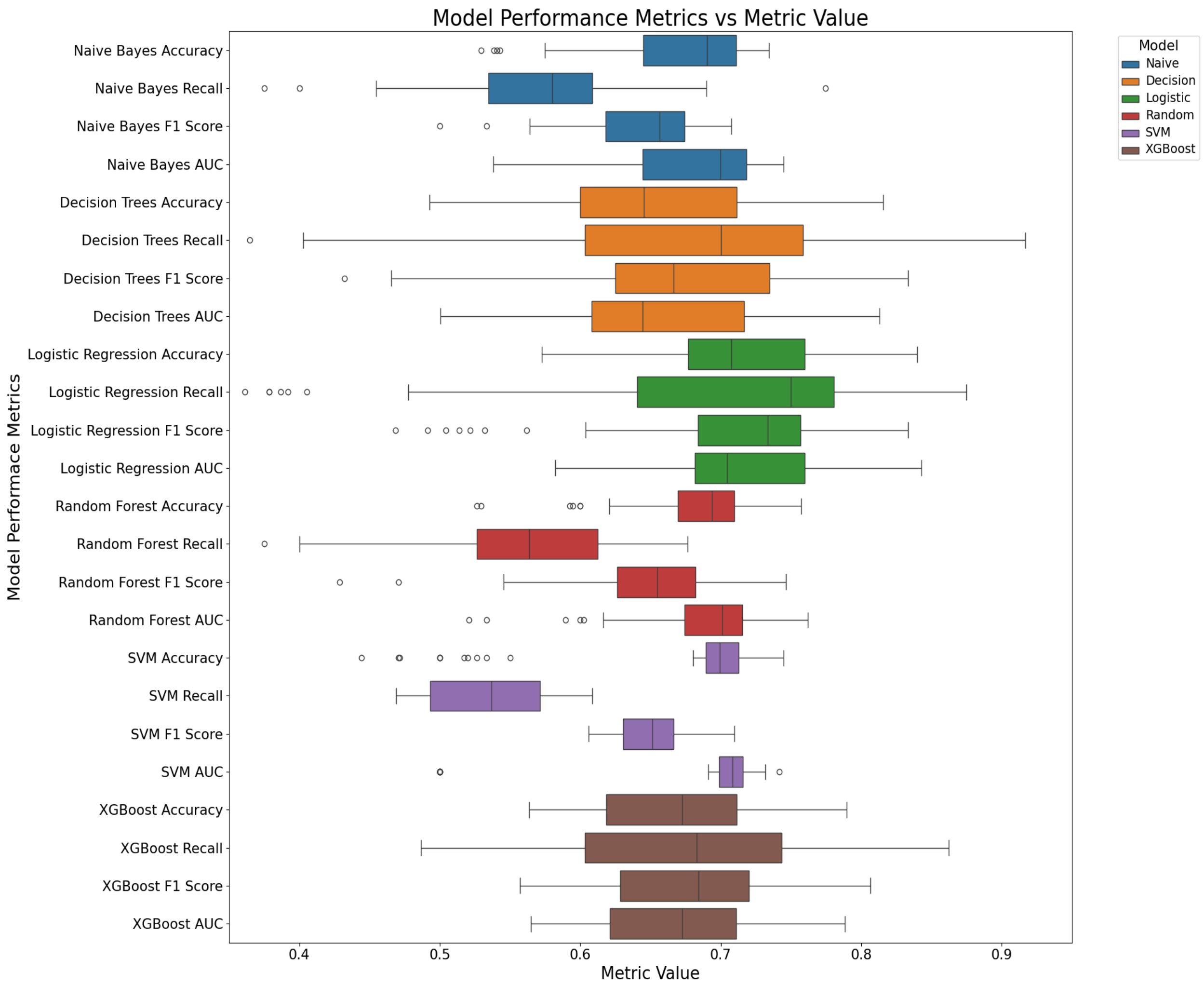
The XGBoost model's accuracy was observed to be strong, with most values lying in the range of 70-85%, indicating a high level of correct predictions across different test set sizes. This performance attests to the robust nature of XGBoost and its ability to handle a variety of data partitions without a significant loss in prediction capability. The recall metric, while less consistent than accuracy, showed the model's proficiency in identifying true positive cases, albeit with some fluctuation. This variability in recall may reflect the model's sensitivity to the specific distribution of classes within different test sets, an important consideration in applications where identifying all positive instances is critical. The F1 score, which balances precision and recall, also displayed fluctuations but remained within a relatively high range, suggesting that the model maintained a reasonable balance between identifying relevant instances and limiting false positives. In cases where the recall dipped, the F1 score also trended downwards, highlighting areas where model tuning might improve performance. AUC demonstrated the model's discriminative ability, remaining fairly stable across test sizes. While there were moments where

AUC peaked or dipped, these were not drastic, and the metric consistently hovered around the 0.7 to 0.8 mark, showcasing the model's overall good classification ability.

The XGBoost model proved to be an effective tool for classification tasks, with its high accuracy and relatively stable performance across metrics. However, the observed variability, particularly in recall and F1 score, indicates opportunities for enhancement, potentially through more granular hyperparameter tuning or by addressing any imbalances in the training data. The stable AUC suggests that the model is a reliable classifier; further refinements could optimize its performance, particularly in the recall metric, to ensure that the model not only predicts accurately but also minimizes the number of missed positive cases, which is of paramount importance in many practical applications.

4.7 All Models Evaluation

In the evaluation of several classification models, we analyzed their performance by considering metrics such as accuracy, recall, F1 score, and Area Under the Curve (AUC) against varying test sizes (Figure 7)



.Figure 7: Box plot of the performance metrics of all the models

The Naive Bayes classifier demonstrated commendable accuracy yet experienced fluctuations in recall and F1 score, indicating inconsistent variability in its performance. The Decision Trees showed substantial accuracy and AUC, suggesting overall reliability, but were subject to variability that could be mitigated with model adjustments such as tree pruning. Logistic Regression presented stable accuracy and AUC, with some variability in recall and F1 score, indicating a need for careful tuning to enhance the balance between precision and recall.

The Random Forest classifier exhibited consistent accuracy and a relatively stable AUC, although the variability in recall and F1 score suggested room for fine-tuning, particularly in hyperparameter optimization to enhance overall reliability. The SVM stood out for its high accuracy but showed a lower recall, which is a potential concern in areas where missing true positives is particularly costly; thus, improvement efforts could be directed towards increasing the model's sensitivity. XGBoost, known for its robust accuracy, maintained a stable AUC but with fluctuating recall and F1 scores, signifying potential for better precision in identifying positive cases.

Across the board, the models maintained a commendable level of accuracy, and most exhibited stable AUC, signifying good class differentiation. However, the notable variability in recall and F1 scores across various models points to the necessity for strategic enhancements. These could include implementing adjustments to decision thresholds, addressing class imbalance, or refining hyperparameters, especially in scenarios with high costs for false negatives. Such targeted improvements are poised to bolster the efficacy of these classifiers in more specialized and demanding domains.

Chapter 5. Conclusion

In conclusion, the choice of a machine learning model should be informed by the particular performance metric that is most critical for the task at hand. While XGBoost and Logistic Regression may offer the most reliable performances across most test sizes, models like SVM might require additional consideration of their trade-offs. Decision Trees and Random Forest could be less desirable due to their variability, and the Naive Bayes model may need scrutiny for performance stability. Each model's characteristics should be carefully weighed against the requirements of the specific problem being addressed. From our analysis, we are not able to rule any one specific model for testing FDC Type II. Rather, our findings indicate that future research into this topic is required, specifically testing the full extent of Logistic Regression’s feasibility in categorizing brainscans.

5.1 Future Work

In future work, the incorporation of both machine learning models and traditional 'eye test' analysis of MRI images should be prioritized to enhance the detection of Focal Cortical Dysplasia (FCD) Type II. Employing a dual-approach that combines the interpretative skills of clinicians with the analytical power of algorithms could provide a more comprehensive and accurate diagnostic tool. This strategy would leverage the strengths of both human expertise, which is adept at nuanced pattern recognition, and machine learning models, which excel at processing large volumes of data for pattern detection.

Moreover, the accumulation of a larger and more diverse dataset would be highly beneficial, particularly for employing deep learning techniques that require substantial data to achieve optimal performance. The additional data would not only provide a richer basis for training but also improve the models' generalization capabilities to unseen data. It would also allow for the application of more complex deep learning architectures that might capture subtleties and complexities associated with FCD Type II.

Despite the allure of deep learning, it's important to acknowledge and further explore the advantages that machine learning models hold over their deep learning counterparts. Machine learning can offer greater transparency, require less computational power, and can be more easily interpreted by clinicians. These benefits are particularly significant in clinical settings, where explainability is crucial for trust and adoption by healthcare professionals.

Lastly, there is a valuable opportunity to improve MRI feature extraction applications. Current applications like FastSurfer are designed to balance efficiency with performance; however, there might be room to enhance the quality of feature extraction without significantly compromising on time. By focusing on advanced image processing and machine learning techniques, future applications could extract more nuanced features relevant to FCD Type II, which might lead to better diagnostic performance. Research into these areas could be pivotal in advancing the reliability and efficacy of FCD Type II detection tools, ultimately contributing to better patient outcomes.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Matthieu Devin, Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Manjunath Kudlur, … Zheng, X. (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv.Org*. https://doi.org/10.48550/arxiv.1603.04467

Abbasi, B., & Goldenholz, D. M. (2019). Machine learning applications in epilepsy. Epilepsia (Copenhagen), 60(10), 2037–2047. https://doi.org/10.1111/epi.16333

Balestrini, S., Barba, C., Thom, M., & Guerrini, R. (2023). Focal cortical dysplasia: a practical guide for neurologists. Practical Neurology, 23(4), 293–302. https://doi.org/10.1136/pn-2022-003404

Blumcke, I., Spreafico, R., Haaker, G., Coras, R., Kobow, K., Bien, C. G., Pfäfflin, M., Elger, C., Widman, G., Schramm, J., Becker, A., Braun, K. P., Leijten, F., Baayen, J. C., Aronica, E., Chassoux, F., Hamer, H., Stefan, H., Rössler, K., … Avanzini, G. (2017). Histopathological Findings in Brain Tissue Obtained during Epilepsy Surgery. The New England Journal of Medicine, 377(17), 1648–1656. https://doi.org/10.1056/NEJMoa1703784

Desikan, R. S., Ségonne, F., Fischl, B., Quinn, B. T., Dickerson, B. C., Blacker, D., Buckner, R. L., Dale, A. M., Maguire, R. P., Hyman, B. T., Albert, M. S., & Killiany, R. J. (2006). An automated labeling system for subdividing the human cerebral cortex on

MRI scans into gyral based regions of interest. NeuroImage (Orlando, Fla.), 31(3), 968–980. https://doi.org/10.1016/j.neuroimage.2006.01.021

Douglas, P. S., Cerqueira, M. D., Berman, D. S., Chinnaiyan, K., Cohen, M. S., Lundbye, J. B., Patel, R. A. G., Sengupta, P. P., Soman, P., Weissman, N. J., & Wong, T. C. (2016). The Future of Cardiac Imaging: Report of a Think Tank Convened by the American College of Cardiology. JACC. Cardiovascular Imaging, 9(10), 1211–1223. https://doi.org/10.1016/j.jcmg.2016.02.027

Garner, G. L., Streetman, D. R., Fricker, J. G., Bui, N. E., Yang, C., Patel, N. A., Brown, N. J., Shahrestani, S., Rangel, I. C., Singh, R., & Gendreau, J. L. (2022). Focal cortical dysplasia as a cause of epilepsy: The current evidence of associated genes and future therapeutic treatments. *Interdisciplinary Neurosurgery*, *30*, 101635. https://doi.org/10.1016/j.inat.2022.101635

Fischl, B., Salat, D. H., Busa, E., Albert, M., Dieterich, M., Haselgrove, C., van der Kouwe, A., Killiany, R., Kennedy, D., Klaveness, S., Montillo, A., Makris, N., Rosen, B., & Dale, A. M. (2002). Whole Brain Segmentation: Automated Labeling of Neuroanatomical Structures in the Human Brain. Neuron (Cambridge, Mass.), 33(3), 341–355. https://doi.org/10.1016/S0896-6273(02)00569-X

Henschel, L., Conjeti, S., Estrada, S., Diers, K., Fischl, B., & Reuter, M. (2020). FastSurfer—A fast and accurate deep learning based neuroimaging pipeline. *NeuroImage*, *219*, 117012. https://doi.org/10.1016/j.neuroimage.2020.117012

Igual, L., Seguí, S., Vitrià, J., Puertas, E., Radeva, P., Pujol, O., Escalera, S., Dantí, F., & Garrido, L. (2017). Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications (1st ed. 2017.). Springer Nature. https://doi.org/10.1007/978-3-319-50017-1

Kabat, J., & Król, P. (2012). Focal cortical dysplasia – review. *Polish Journal of Radiology*, *77*(2), 35–43.

Kim, S. H., & Choi, J. (2019). Pathological Classification of Focal Cortical Dysplasia (FCD): Personal Comments for Well Understanding FCD Classification. *Journal of Korean Neurosurgical Society*, *62*(3), 288–295. https://doi.org/10.3340/jkns.2019.0025

Kluyver, T., Ragan-Kelley, B., Perez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., & Willing, C. (2016). *Jupyter Notebooks-a publishing format for reproducible computational workflows*.

Lee, H. M., Gill, R. S., Bernasconi, N., & Bernasconi, A. (2023). Machine Learning in Neuroimaging of Epilepsy. In Machine Learning for Brain Disorders (pp. 879–898). Springer US. https://doi.org/10.1007/978-1-0716-3195-9_27

Li, S., Wang, Y., Hu, Z., Guan, L., Hai, Y., Zhang, H., He, L., Jiang, W., & Guo, H. (2021). High-fidelity diffusion tensor imaging of the cervical spinal cord using point-spread-function encoded EPI. NeuroImage (Orlando, Fla.), 236, 118043–118043. https://doi.org/10.1016/j.neuroimage.2021.118043

Liu, S., Masurkar, A. V., Rusinek, H., Chen, J., Zhang, B., Zhu, W., Fernandez-Granda, C., & Razavian, N. (2022). Generalizable deep learning model for early Alzheimer's disease detection from structural MRIs. *Scientific Reports*, *12*(1), 17106. https://doi.org/10.1038/s41598-022-20674-x

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. https://doi.org/10.5555/1953048.2078195

Powers, D. M. W. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv.Org*. https://doi.org/10.48550/arxiv.2010.16061

Represa, A. (2019). Why Malformations of Cortical Development Cause Epilepsy. Frontiers in Neuroscience, 13, 250–250. https://doi.org/10.3389/fnins.2019.00250

Reuter, M., Rosas, H. D., & Fischl, B. (2010). Highly accurate inverse consistent registration: A robust approach. NeuroImage (Orlando, Fla.), 53(4), 1181–1196. https://doi.org/10.1016/j.neuroimage.2010.07.020

Schuch, F., Walger, L., Schmitz, M., David, B., Bauer, T., Harms, A., Fischbach, L., Schulte, F., Schidlowski, M., Reiter, J., Bitzer, F., von Wrede, R., Rácz, A., Baumgartner, T., Borger, V., Schneider, M., Flender, A., Becker, A., Vatter, H., … Rüber, T. (2023). An open presurgery MRI dataset of people with epilepsy and focal cortical dysplasia type II. *Scientific Data*, *10*(1), 475–475. https://doi.org/10.1038/s41597-023-02386-7

Schuch, F., Walger, L., Schmitz, M., David, B., Bauer, T., Harms, A., Fischbach, L., Schulte, F., Schidlowski, M., Reiter, J., Bitzer, F., von Wrede, R., Rácz, A., Baumgartner, T., Borger, V., Schneider, M., Flender, A., Becker, A., Vatter, H., … Rüber, T. (2023). An open presurgery MRI dataset of people with epilepsy and focal cortical dysplasia type II. OpenNeuro. https://openneuro.org/datasets/ds004199/versions/1.0.5

Smith, S. M., & Nichols, T. E. (2018). Statistical Challenges in "Big Data" Human Neuroimaging. Neuron (Cambridge, Mass.), 97(2), 263–268. https://doi.org/10.1016/j.neuron.2017.12.018

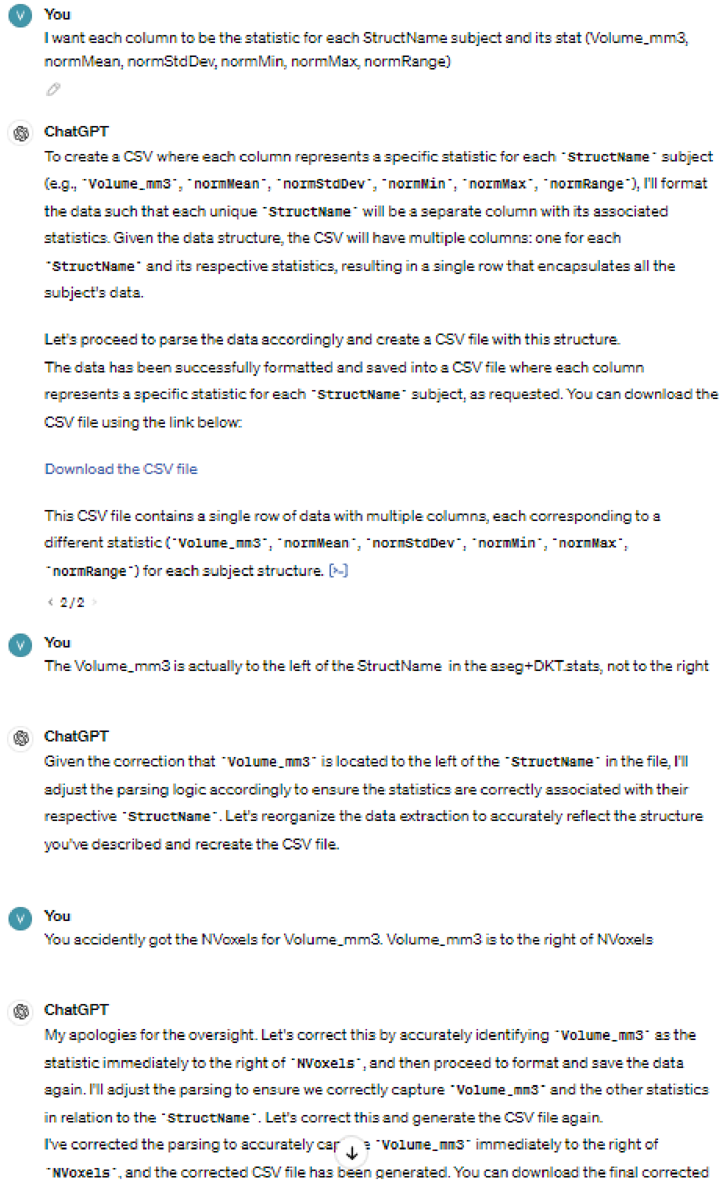
Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. Computing in Science & Engineering, 13(2), 22–30. https://doi.org/10.1109/MCSE.2011.37

World Health Organization. (2024, February 7). *Epilepsy*. https://www.who.int/news-room/fact-sheets/detail/epilepsy

Appendix

Appendix A: An Example of using ChatGPT as Prompt Engineering to Generate Python Scripts to Format the Stats Files



You

I want each column to be the statistic for each StructName subject and its stat (Volume_mm3, normMean, normStdDev, normMin, normMax, normRange)

ChatGPT

To create a CSV where each column represents a specific statistic for each `StructName` subject (e.g., `Volume_mm3`, `normMean`, `normStdDev`, `normMin`, `normMax`, `normRange`), I'll format the data such that each unique `StructName` will be a separate column with its associated statistics. Given the data structure, the CSV will have multiple columns: one for each `StructName` and its respective statistics, resulting in a single row that encapsulates all the subject's data.

Let's proceed to parse the data accordingly and create a CSV file with this structure.
The data has been successfully formatted and saved into a CSV file where each column represents a specific statistic for each `StructName` subject, as requested. You can download the CSV file using the link below:

Download the CSV file

This CSV file contains a single row of data with multiple columns, each corresponding to a different statistic (`Volume_mm3`, `normMean`, `normStdDev`, `normMin`, `normMax`, `normRange`) for each subject structure.

‹ 2 / 2 ›

You

The Volume_mm3 is actually to the left of the StructName in the aseg+DKT.stats, not to the right

ChatGPT

Given the correction that `Volume_mm3` is located to the left of the `StructName` in the file, I'll adjust the parsing logic accordingly to ensure the statistics are correctly associated with their respective `StructName`. Let's reorganize the data extraction to accurately reflect the structure you've described and recreate the CSV file.

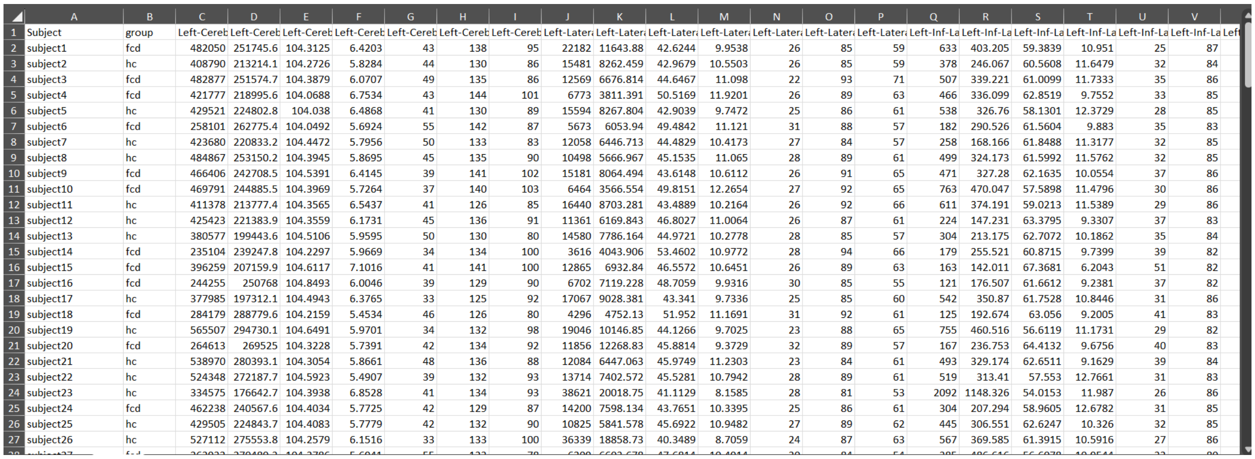
You

You accidently got the NVoxels for Volume_mm3. Volume_mm3 is to the right of NVoxels

ChatGPT

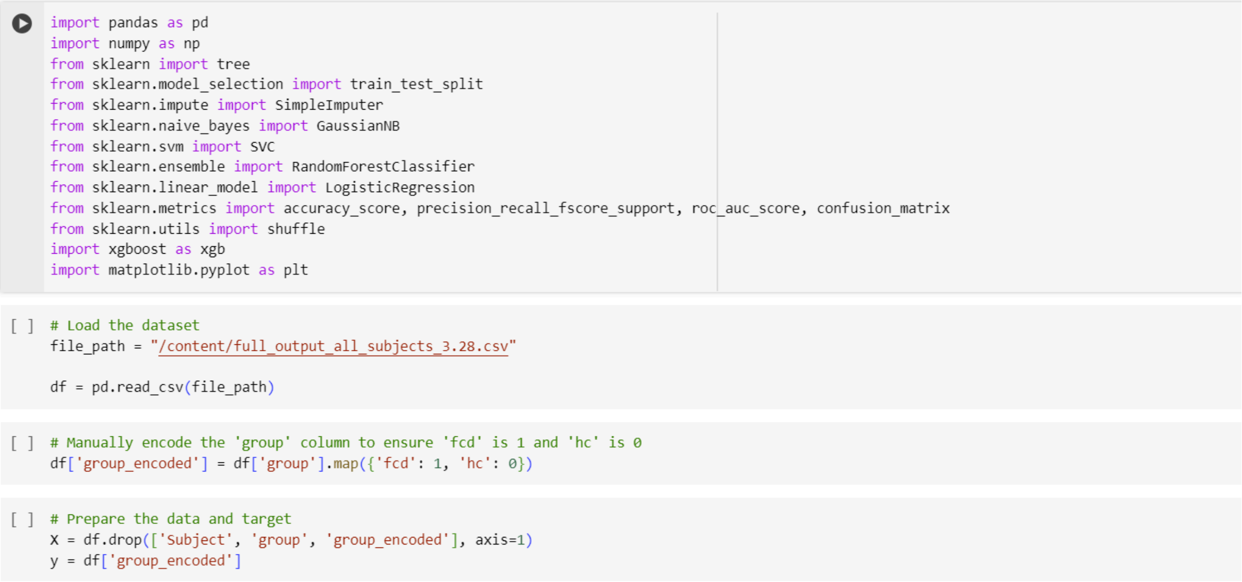
My apologies for the oversight. Let's correct this by accurately identifying `Volume_mm3` as the statistic immediately to the right of `NVoxels`, and then proceed to format and save the data again. I'll adjust the parsing to ensure we correctly capture `Volume_mm3` and the other statistics in relation to the `StructName`. Let's correct this and generate the CSV file again.
I've corrected the parsing to accurately cap e `Volume_mm3` immediately to the right of `NVoxels`, and the corrected CSV file has been generated. You can download the final corrected

Appendix B: Final Dataset after Prompt Engineering Stats Files from FastSurfer



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	
1	Subject	group	Left-Cereb	Left-Cereb	Left-Cereb	Left-Cereb	Left-Cereb	Left-Cereb	Left-Cereb	Left-Latera	Left-Latera	Left-Latera	Left-Latera	Left-Latera	Left-Latera	Left-Latera	Left-Inf-La	Left-Inf-La	Left-Inf-La	Left-Inf-La	Left-Inf-La	Left-Inf-La	Left
2	subject1	fcd	482050	251745.6	104.3125	6.4203	43	138	95	22182	11643.88	42.6244	9.9538	26	85	59	633	403.205	59.3839	10.951	25	87	
3	subject2	hc	408790	213214.1	104.2726	5.8284	44	130	86	15481	8262.459	42.9679	10.5503	26	85	59	378	246.067	60.5608	11.6479	32	84	
4	subject3	fcd	482877	251574.7	104.3879	6.0707	49	135	86	12569	6676.814	44.6467	11.098	22	93	71	507	339.221	61.0099	11.7333	35	86	
5	subject4	fcd	421777	218995.6	104.0688	6.7534	43	144	101	6773	3811.391	50.5169	11.9201	26	89	63	466	336.099	62.8519	9.7552	33	85	
6	subject5	hc	429521	224802.8	104.038	6.4868	41	130	89	15594	8267.804	42.9039	9.7472	25	86	61	538	326.76	58.1301	12.3729	28	85	
7	subject6	fcd	258101	262775.4	104.0492	5.6924	55	142	87	5673	6053.94	49.4842	11.121	31	88	57	182	290.526	61.5604	9.883	35	83	
8	subject7	hc	423680	220833.2	104.4472	5.7956	50	133	83	12058	6446.713	44.4829	10.4173	27	84	57	258	168.166	61.8488	11.3177	32	85	
9	subject8	hc	484867	253150.2	104.3945	5.8695	45	135	90	10498	5666.967	45.1535	11.065	28	89	61	499	324.173	61.5992	11.5762	32	85	
10	subject9	fcd	466406	242708.5	104.5391	6.4145	39	141	102	15181	8064.494	43.6148	10.6112	26	91	65	471	327.28	62.1635	10.0554	37	86	
11	subject10	fcd	469791	244885.5	104.3969	5.7264	37	140	103	6464	3566.554	49.8151	12.2654	27	92	65	763	470.047	57.5898	11.4796	30	86	
12	subject11	hc	411378	213777.4	104.3565	6.5437	41	126	85	16440	8703.281	43.4889	10.2164	26	92	66	611	374.191	59.0213	11.5389	29	86	
13	subject12	hc	425423	221383.9	104.3559	6.1731	45	136	91	11361	6169.843	46.8027	11.0064	26	87	61	224	147.231	63.3795	9.3307	37	83	
14	subject13	hc	380577	199443.6	104.5106	5.9595	50	130	80	14580	7786.164	44.9721	10.2778	28	85	57	304	213.175	62.7072	10.1862	35	84	
15	subject14	fcd	235104	239247.8	104.2297	5.9669	34	134	100	3616	4043.906	53.4602	10.9772	28	94	66	179	255.521	60.8715	9.7399	39	82	
16	subject15	fcd	396259	207159.9	104.6117	7.1016	41	141	100	12865	6932.84	46.5572	10.6451	26	89	63	163	142.011	67.3681	6.2043	51	82	
17	subject16	fcd	244255	250768	104.8493	6.0046	39	129	90	6702	7119.228	48.7059	9.9316	30	85	55	121	176.507	61.6612	9.2381	37	82	
18	subject17	hc	377985	197312.1	104.4943	6.3765	33	125	92	17067	9028.381	43.341	9.7336	25	85	60	542	350.87	61.7528	10.8446	31	86	
19	subject18	fcd	284179	288779.6	104.2159	5.4534	46	126	80	4296	4752.13	51.952	11.1691	31	92	61	125	192.674	63.056	9.2005	41	83	
20	subject19	hc	565507	294730.1	104.6491	5.9701	34	132	98	19046	10146.85	44.1266	9.7025	23	88	65	755	460.516	56.6119	11.1731	29	82	
21	subject20	fcd	264613	269525	104.3228	5.7391	42	134	92	11856	12268.83	45.8814	9.3729	32	89	57	167	236.753	64.4132	9.6756	40	83	
22	subject21	hc	538970	280393.1	104.3054	5.8661	48	136	88	12084	6447.063	45.9749	11.2303	23	84	61	493	329.174	62.6511	9.1629	39	84	
23	subject22	hc	524348	272187.7	104.5923	5.4907	39	132	93	13714	7402.572	45.5281	10.7942	28	89	61	519	313.41	57.553	12.7661	31	83	
24	subject23	hc	334575	176642.7	104.3938	6.8528	41	134	93	38621	20018.75	41.1129	8.1585	28	81	53	2092	1148.326	54.0153	11.987	26	86	
25	subject24	fcd	462238	240567.6	104.4034	5.7725	42	129	87	14200	7598.134	43.7651	10.3395	25	86	61	304	207.294	58.9605	12.6782	31	85	
26	subject25	hc	429505	224843.7	104.4083	5.7779	42	132	90	10825	5841.578	45.6922	10.9482	27	89	62	445	306.551	62.6247	10.326	32	85	
27	subject26	hc	527112	275553.8	104.2579	6.1516	33	133	100	36339	18858.73	40.3489	8.7059	24	87	63	567	369.585	61.3915	10.5916	27	86	

Appendix C: Jupyter Notebook Code to Train and Test Machine a SVM Model



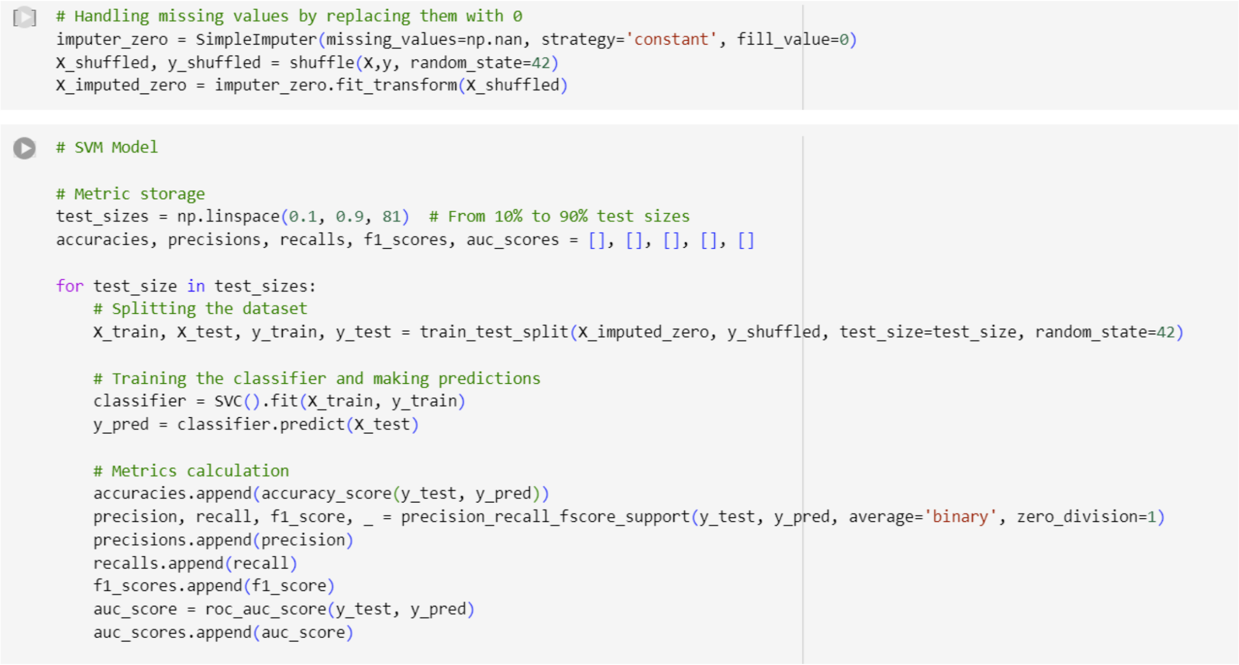
```
import pandas as pd
import numpy as np
from sklearn import tree
from sklearn.model_selection import train_test_split
from sklearn.impute import SimpleImputer
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, precision_recall_fscore_support, roc_auc_score, confusion_matrix
from sklearn.utils import shuffle
import xgboost as xgb
import matplotlib.pyplot as plt
```

```
# Load the dataset
file_path = "/content/full_output_all_subjects_3.28.csv"

df = pd.read_csv(file_path)
```

```
# Manually encode the 'group' column to ensure 'fcd' is 1 and 'hc' is 0
df['group_encoded'] = df['group'].map({'fcd': 1, 'hc': 0})
```

```
# Prepare the data and target
X = df.drop(['Subject', 'group', 'group_encoded'], axis=1)
y = df['group_encoded']
```



```
# Handling missing values by replacing them with 0
imputer_zero = SimpleImputer(missing_values=np.nan, strategy='constant', fill_value=0)
X_shuffled, y_shuffled = shuffle(X,y, random_state=42)
X_imputed_zero = imputer_zero.fit_transform(X_shuffled)
```

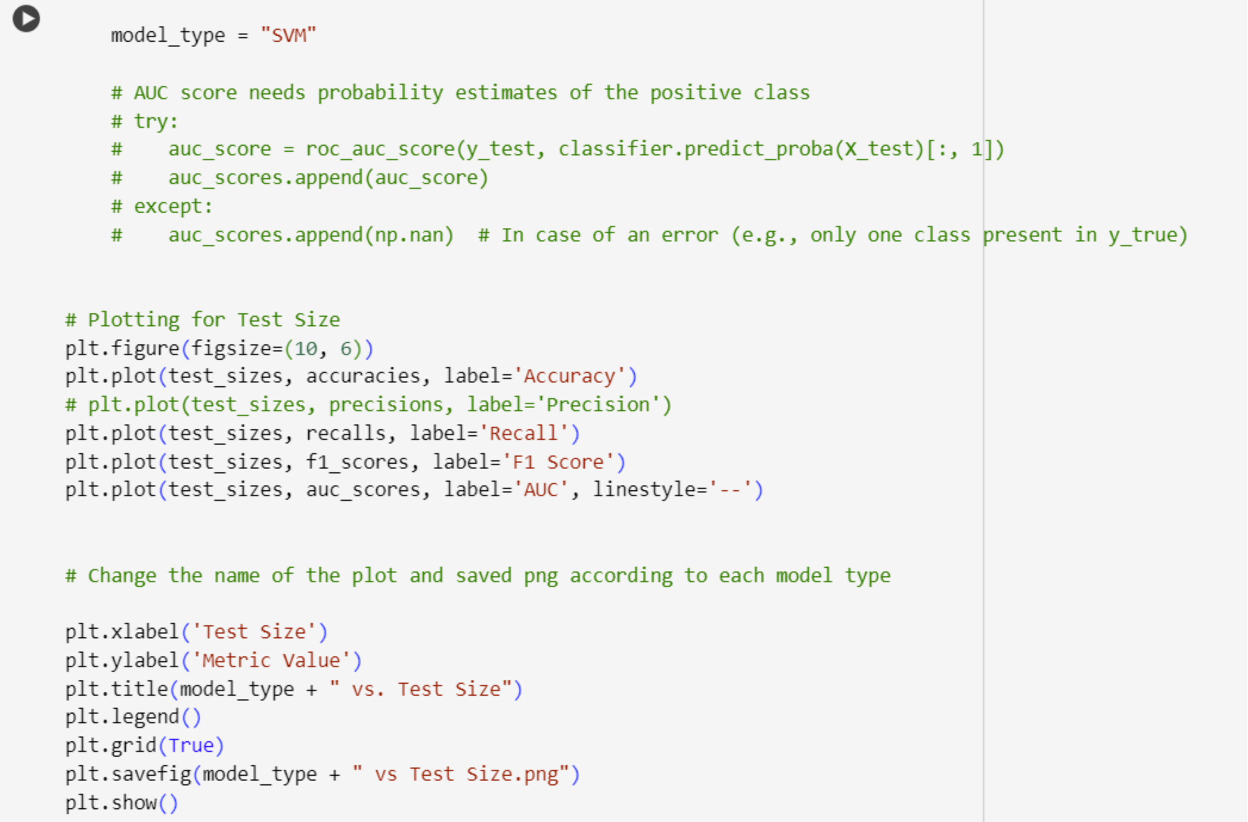
```
# SVM Model

# Metric storage
test_sizes = np.linspace(0.1, 0.9, 81)  # From 10% to 90% test sizes
accuracies, precisions, recalls, f1_scores, auc_scores = [], [], [], [], []

for test_size in test_sizes:
    # Splitting the dataset
    X_train, X_test, y_train, y_test = train_test_split(X_imputed_zero, y_shuffled, test_size=test_size, random_state=42)

    # Training the classifier and making predictions
    classifier = SVC().fit(X_train, y_train)
    y_pred = classifier.predict(X_test)

    # Metrics calculation
    accuracies.append(accuracy_score(y_test, y_pred))
    precision, recall, f1_score, _ = precision_recall_fscore_support(y_test, y_pred, average='binary', zero_division=1)
    precisions.append(precision)
    recalls.append(recall)
    f1_scores.append(f1_score)
    auc_score = roc_auc_score(y_test, y_pred)
    auc_scores.append(auc_score)
```



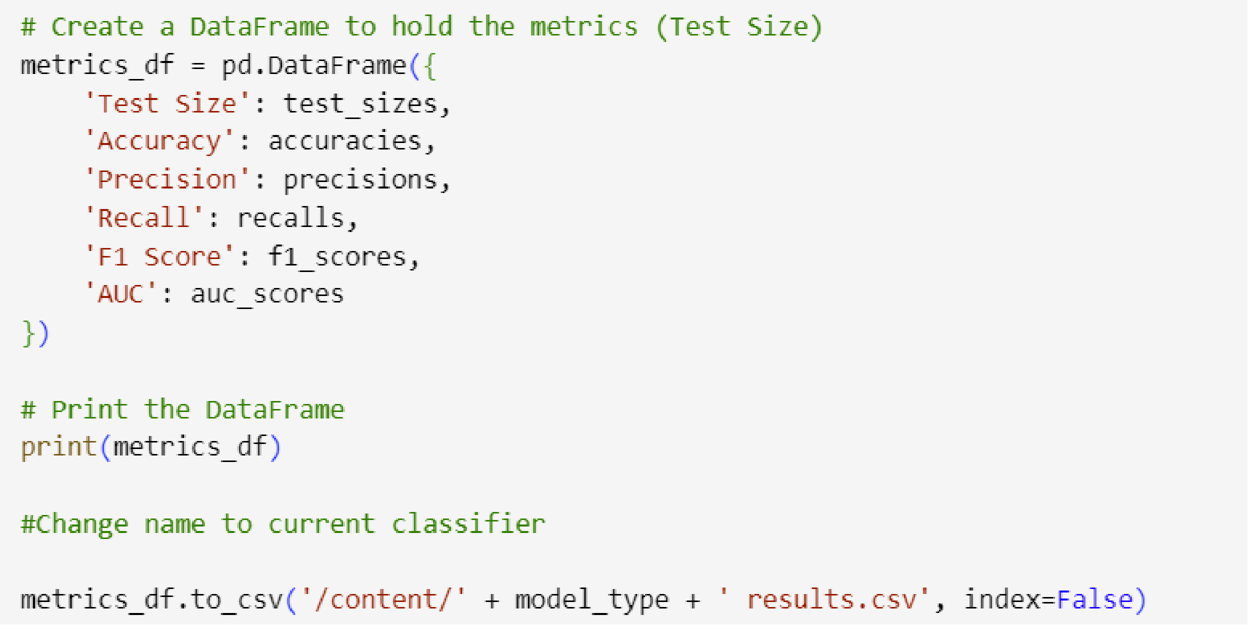
```
        model_type = "SVM"

        # AUC score needs probability estimates of the positive class
        # try:
        #    auc_score = roc_auc_score(y_test, classifier.predict_proba(X_test)[:, 1])
        #    auc_scores.append(auc_score)
        # except:
        #    auc_scores.append(np.nan)  # In case of an error (e.g., only one class present in y_true)


    # Plotting for Test Size
    plt.figure(figsize=(10, 6))
    plt.plot(test_sizes, accuracies, label='Accuracy')
    # plt.plot(test_sizes, precisions, label='Precision')
    plt.plot(test_sizes, recalls, label='Recall')
    plt.plot(test_sizes, f1_scores, label='F1 Score')
    plt.plot(test_sizes, auc_scores, label='AUC', linestyle='--')


    # Change the name of the plot and saved png according to each model type

    plt.xlabel('Test Size')
    plt.ylabel('Metric Value')
    plt.title(model_type + " vs. Test Size")
    plt.legend()
    plt.grid(True)
    plt.savefig(model_type + " vs Test Size.png")
    plt.show()
```

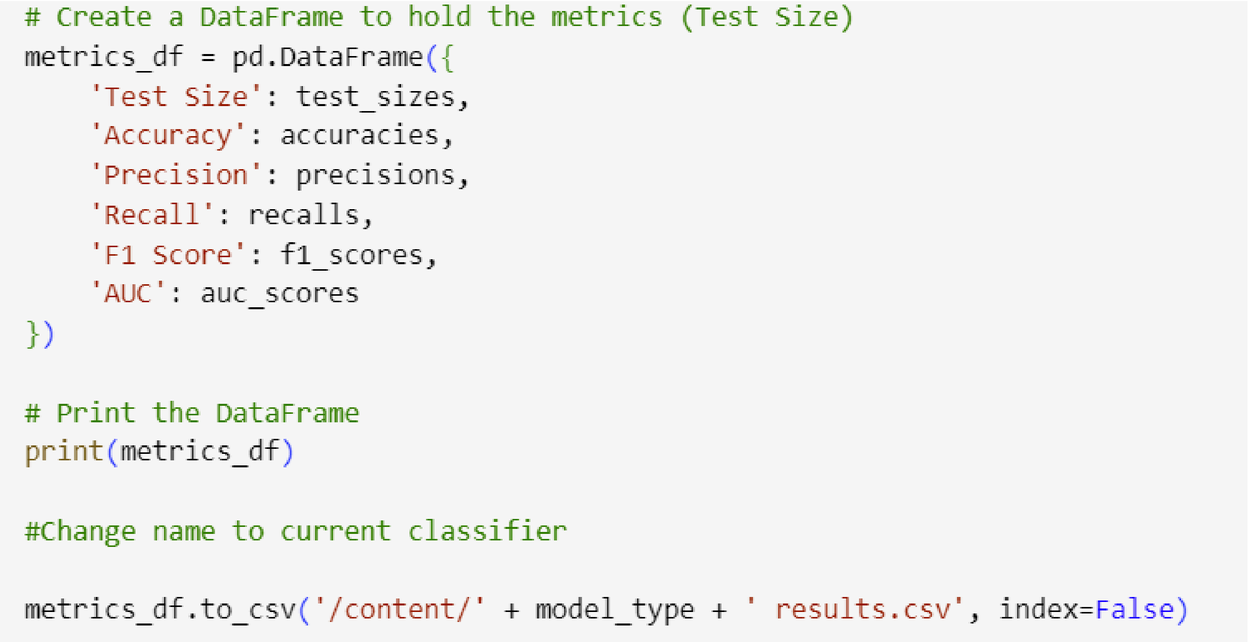


```
# Create a DataFrame to hold the metrics (Test Size)
metrics_df = pd.DataFrame({
    'Test Size': test_sizes,
    'Accuracy': accuracies,
    'Precision': precisions,
    'Recall': recalls,
    'F1 Score': f1_scores,
    'AUC': auc_scores
})

# Print the DataFrame
print(metrics_df)

#Change name to current classifier

metrics_df.to_csv('/content/' + model_type + ' results.csv', index=False)
```

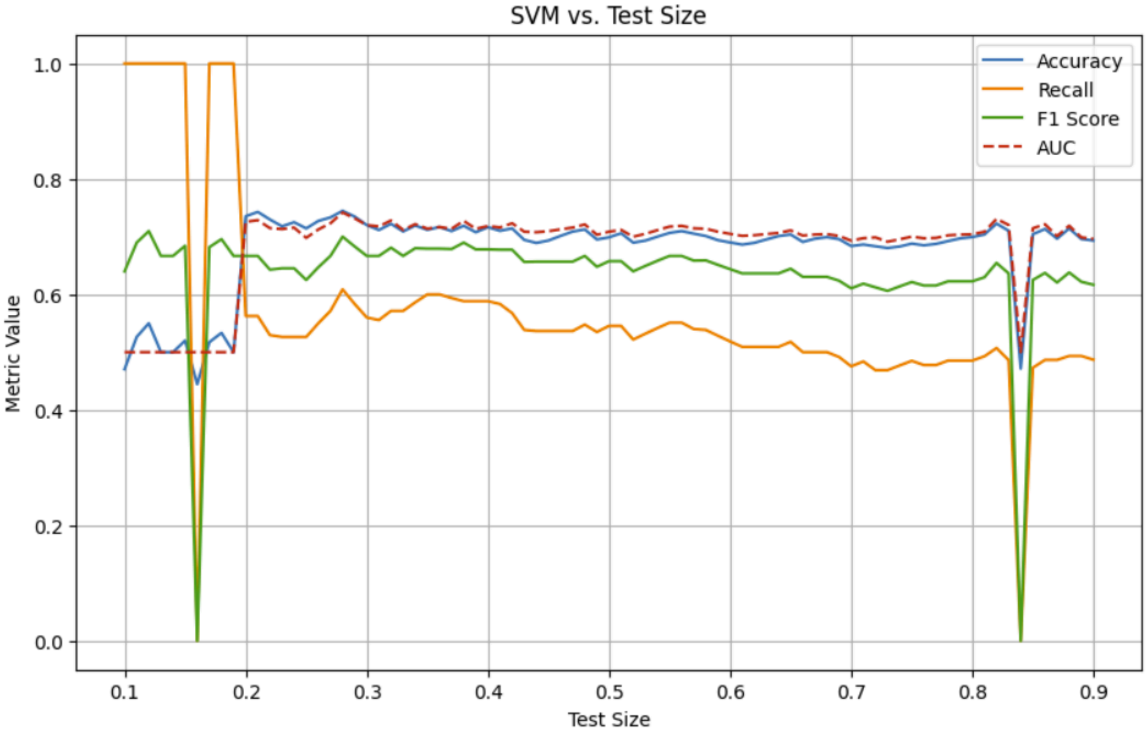


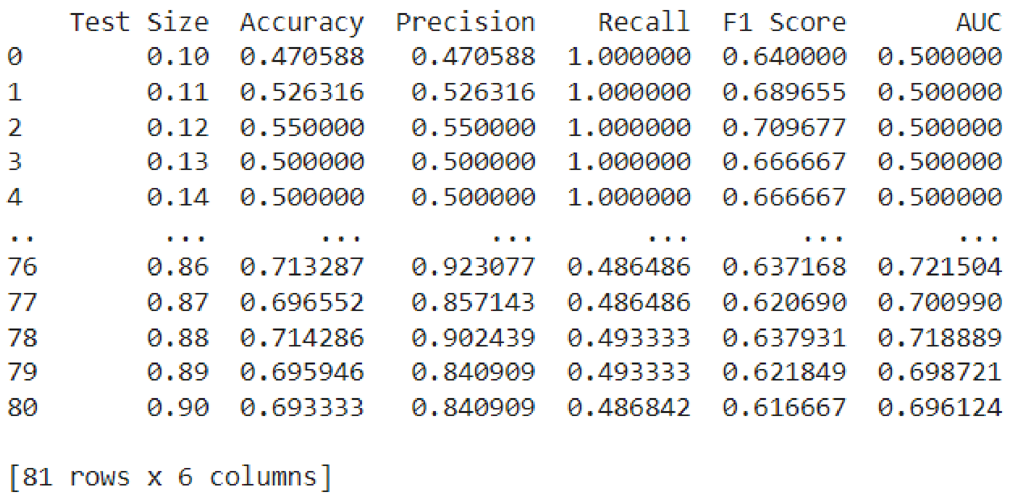
```
# Create a DataFrame to hold the metrics (Test Size)
metrics_df = pd.DataFrame({
    'Test Size': test_sizes,
    'Accuracy': accuracies,
    'Precision': precisions,
    'Recall': recalls,
    'F1 Score': f1_scores,
    'AUC': auc_scores
})

# Print the DataFrame
print(metrics_df)

#Change name to current classifier

metrics_df.to_csv('/content/' + model_type + ' results.csv', index=False)
```





```
    Test Size  Accuracy  Precision    Recall  F1 Score       AUC
0        0.10  0.470588   0.470588  1.000000  0.640000  0.500000
1        0.11  0.526316   0.526316  1.000000  0.689655  0.500000
2        0.12  0.550000   0.550000  1.000000  0.709677  0.500000
3        0.13  0.500000   0.500000  1.000000  0.666667  0.500000
4        0.14  0.500000   0.500000  1.000000  0.666667  0.500000
..        ...       ...        ...       ...       ...       ...
76       0.86  0.713287   0.923077  0.486486  0.637168  0.721504
77       0.87  0.696552   0.857143  0.486486  0.620690  0.700990
78       0.88  0.714286   0.902439  0.493333  0.637931  0.718889
79       0.89  0.695946   0.840909  0.493333  0.621849  0.698721
80       0.90  0.693333   0.840909  0.486842  0.616667  0.696124

[81 rows x 6 columns]
```