HIV Neuroimaging Data Analysis and Supervised Machine Learning

SUPERVISOR: MATTHIAS TREDER

MODULE NUMBER: CM3202

MODULE TITLE: ONE SEMESTER INDIVIDUAL PROJECT

MODULE CREDITS: 40

Robert Taylor | C1630630 | 15/05/2020

Abstract

Project Description

HIV is one of the deadliest infections affecting almost 40 million people in the world. It is estimated that 20.4% of South African Women have been infected. As HIV causes neuroinflammation and neuronal death, it should be detectable in neuroimaging data. A neuroimaging dataset has been provided by Stellenbosch University, South Africa with 124 participants involved. This project will use data analysis and machine learning techniques in the python programming language to help understand the brain tissue and cognitive variables for HIV positive women in South Africa.

Methods

Pre-processing occurred on the dataset of HIV-positive (*1, n=62*) and HIV-negative (*0, n=62*) participant neuroimaging data to standardise the regional MRI values. Then data analysis techniques were applied to identify the descriptive statistical characteristics to find the count, mean, quartiles (25/50/75%) and range (min/max) and an independent t-test performed to find which brain regions had statistical significance.

Supervised machine learning models were then created to predict the HIV status of participants using a train/test (75/25%) data split. These predictive models explored a range of classification methods to obtain a reliable result. They were then tuned using cross-validation and a hyperparameter grid search to obtain the best classification performance when predicting the HIV status from the test data. To achieve a superior performance, feature selection was then implemented for these machine learning models by creating a train/test data split that only used the HIV significant brain regions.

Results

From the statistical analysis, four significant brain regions were identified with their magnitude (t-value) and probability of occurrence by chance (p-value). These MRI regions are the left hemisphere Frontal-lobe (*t=2.220, p=0.0283*), the total Corpus-callosum (*t=2.425, p=0.0168*), the left hemisphere Putamen (*t=2.414, p=0.0173*) and the right hemisphere Putamen (*t=2.034, p=0.0442*).

The supervised machine learning models were not capable of reliably predicting HIV status from the neuroimaging data. Classification accuracy on average was 54.27% with high variance F*(4,45)=14.12, p=1.527e-07 and* the model's ROC AUC performance at differentiating HIV status was only +1.29% from an undistinguished model (with unsubstantiated variation). When applying feature selection to the models, an average accuracy of 53.03% was obtained with high variation between the classifiers *F(*4,45*)=14.951, p=4.202e-08*. The feature selection models had a surprisingly worse accuracy but were able to differentiate the HIV status on average by +6.03% (5.03% higher than the initial models). The ROC AUC score only had slight variance between the models but was not significant enough statistically *F(*4,45*)=2.4854, p=0.05528*.

Acknowledgements

I would like to thank Dr Matthias Treder, my supervisor who proposed this project, helped guide me on this journey and offered countless advice over the course of this assignment.

I also thank my co-supervisor Dr Georgina Spies from the University of Stellenbosch, South Africa. Whom supplied the medical data necessary for me to use, allowing me this gratifying opportunity.

Finally, I would like to thank all the South African women who chose to participate in this study and help advance the research of HIV in neurology.

Table of Contents

Table of Figures

Introduction

Human Immunodeficiency Virus (HIV) is one of the deadliest infections in the world, with almost 40 million people being affected. [1] HIV is a virus that damages the immune systems' cells and weakens an individual's capability to combat most infections & diseases. Currently, there is no known cure for HIV. Individuals usually take drug treatments that facilitates them to live a prolonged and healthy life while still being infected with the virus. [2] The main form of testing for HIV is through the use of a blood test. A sample of blood is extracted from an individual by obtaining intravenous access to a vein using a medical needle. This is then sent for testing in a laboratory where the presence of the Virus' RNA particles is tested. [3]

Although detecting HIV using blood tests is often accurate, reliable, and cost-effective, it is not easy to understand the full effect of HIV on the brain. HIV positive individuals often have cognitive deficits known as HIV-associated neurocognitive disorders (HANDs). HANDs have been found to occur in many forms like psychomotor skills, verbal and visual memory and information processing speed. This is evidenced in HIV-infected women who scored poorly during testing for HANDs. We know that HIV penetrates the blood-brain barrier early in the course of infection and infects nerve cells, resulting in neuroinflammation and neuronal death. [4] Cognitive impairment, lower grey matter volume and white matter microstructural abnormalities are evident in HIV-positive individuals even with fully suppressive antiretroviral therapy. [5] So, we should expect to see changes within the grey matter volume for HIV infected individuals when separated by brain region.

HIV has been especially prevalent in South Africa which has the largest and most extreme HIV epidemic in the world. The UNAIDS organisation estimate that there are around 7.7 million people living with HIV in 2018 in this region of the world. [6] A third of all new HIV infections are contracted in South Africa, with 240,000 new infections and 71,000 deaths from AIDS-related illnesses in just 2018 alone. HIV prevalence persists across many regions of South Africa with over one in five (20.4%) people known to be enduring a life with HIV in 2018. [6]

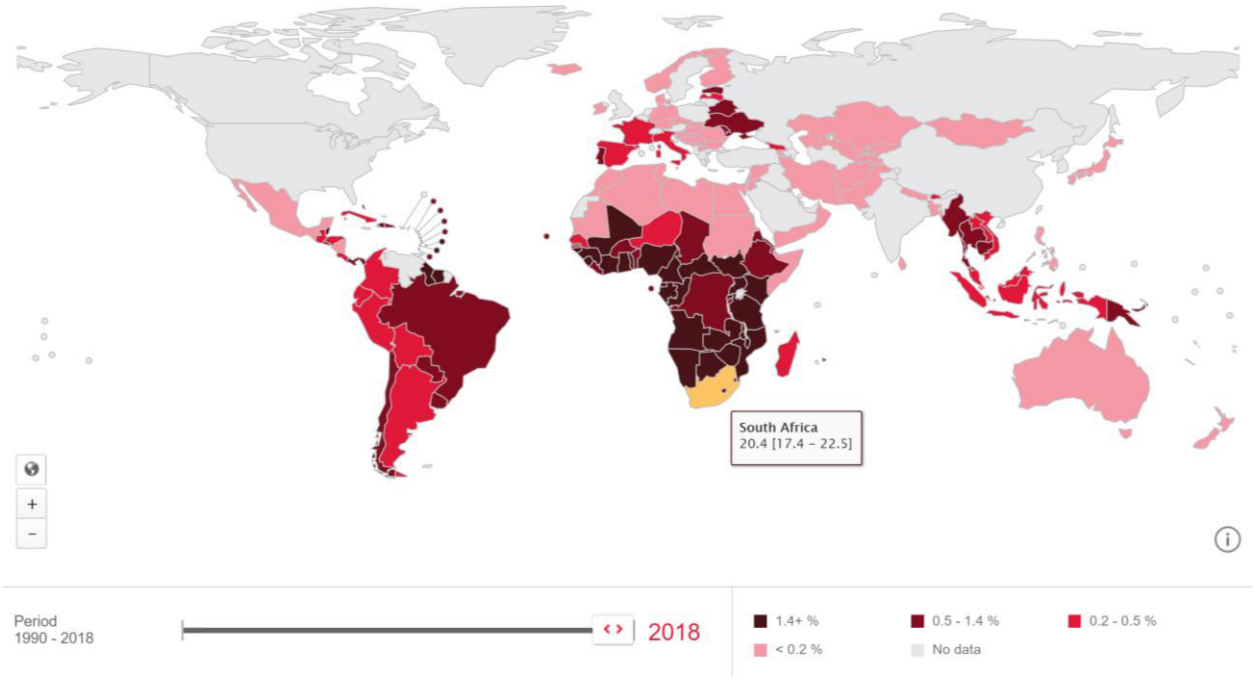

Figure 1: Global prevalence of HIV in 2018 [6]

In this report I will be discussing my final year research project, detailing the outcomes of my investigation; to apply supervised machine-learning models and data analysis techniques on a HIV neuroimaging dataset.

This project acquired a HIV neuroimaging dataset from Stellenbosch University, Tygerberg. Therefore, any results can help contribute to their own university projects by identifying strong inferences between the neuroimaging data and the HIV disease. MRI neuroimages are the most reliable form of brain imaging as it allows us to see the size and quantity of grey-matter and white-matter within the different areas of the brain. With a more advanced HIV disease stage and greater immunocompromise there is often more neuroimaging abnormalities and neurocognitive impairments. [7] This project will offer thoughtful insight towards understanding the brain tissue and cognitive variables for HIV positive women in South Africa.

The main aims of this project were to develop and further technical knowledge of data science techniques such as data analysis and supervised machine learning. This was done through the statistical analysis and visualisation of a neuroimaging dataset to help identify which key areas of the brain signify HIV infection. Methods such as descriptive analysis (averages, frequencies, quartiles) and statistical analysis (T-test, ANOVA) were employed to find correlations, associations, and variance between the neuroimaging data.

Several classical machine-learning predictive models were then developed and used to closely predict if participants are HIV positive or HIV negative as-well as if a positive participant has a detectable viral load, purely based on the supplied neuroimaging data. The use of supervised machine learning techniques was the focus for predicting HIV status & viral load detectability in this project. The main predictive models applied to the dataset used classification to best predict and visualise the results.

The next objective in this project is to evaluate the different models' top classification performance to determine if HIV & viral load detectability can be accurately predicted using the given dataset. Any tangible results that were identified would then be sent to Stellenbosch University and contribute to any of their relevant research projects that examines this dataset. To conclude the project, extensive evaluation and reflection on the overall processes and results were performed. This helped identify some of the limitations that were present and areas of the project that require more work in future.

This project was not able to create an accurate model that could differentiate between HIV negative and positive participants. This became apparent because of the limitations identified in the given dataset. Following these results, other hypotheses were investigated within the neuroimaging dataset to explore other areas for inference. These results also showed no significance, but the processes and data science methodologies learned was invaluable. Some statistically significant brain regions were also identified and then isolated to improve performance using feature selection methodology. The approach, implementation, and results are mentioned in the subsequent sections of this report, continue reading to find out more.

Background

Context: HIV and Neuroimaging

The HIV Disease

HIV is often a sexually transmitted infection and occurs when blood, pre-ejaculate, semen, and vaginal fluids is transferred between individuals. Research shows that HIV is un-transmittable through condom-less sexual intercourse, if a HIV-positive individual has a reliably undetectable viral load. [8] Non-sexual transmission can also occur from an infected mother to her infant during pregnancy, during childbirth by exposure to her blood or vaginal fluid, and through breast milk. Within these bodily fluids, HIV is present as both free virus particles and virus within infected immune cells. [9]

HIV attacks the immune system by destroying specific white blood cells called CD4 positive (**CD4+**) T cells that are vital to fighting off infection. The resulting shortage of these cells leaves people infected with HIV vulnerable to other infections, diseases, and additional complications. [10] CD4 cells are a type of white blood cell that are specific to the immune system and destroyed by HIV. Generally, the higher an individual's CD4 cell count, the stronger their immune system. So, analysing the CD4 count is a good indicator of how their immune system is performing. Treatment using **antiretroviral therapy** (**ART**) and medications that help control HIV is essential for HIV positive individuals as it prevents the risk of developing **acquired immunodeficiency syndrome** (**AIDS**), which is fatal. [3] After infection with HIV it is estimated to be 9 to 11 years without treatment to cause AIDS, a condition in which progressive failure of the immune system allows life-threatening opportunistic infections and cancers to thrive.

HIV **viral load** is the amount of HIV detectable in a sample of a person's blood. It is calculated by testing for the **HIV Ribonucleic acid** (**RNA**) and tracking how many HIV particles are in a sample of blood. The amount of detectable HIV RNA determines an individual's viral load score. HIV treatment aims to supress the viral load to a point where the virus cannot be detected by a viral load test, categorizing that individual as HIV positive with an undetectable viral load.

Current Status of HIV in South Africa

South Africa has made impressive progress in recent years in getting more people to test for HIV. In 2017, South Africa reached one of their core targets, with 90% of people living with HIV aware of their status, up from 85% in 2015. [6] Of the 90% aware, 68% are on HIV treatment, which still equates to 62% of all individuals living with HIV. And 54% of all HIV positive South Africans have successfully got the disease virally suppressed.

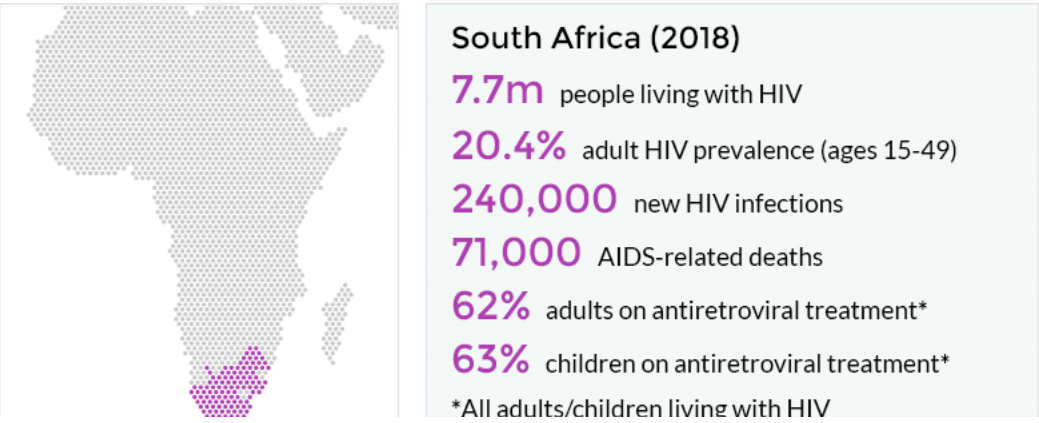

Figure 2: HIV Key Statistics in South Africa [49]

For South Africa there are subgroups of the population that are disproportionately affected by HIV. Specifically, South African Women suffer the highest frequency of HIV with an **estimated 26% of all South African Women having the virus**. Comparatively, this is in contrast to the 15% of the male population who are estimated to have HIV. [11] Aspects such as poverty, discrimination against women, and violence centred on gender are recognised as the main reasons for the HIV prevalence inequality. [12] This project will focus on **HIV positive women in South Africa** as they are currently one of the most severe subgroups affected and have prior research and data available to exploit.

Magnetic Resonance Imaging (MRI)

MRI scans use a powerful magnetic field and radio waves to generate detailed images of organs and tissues within the human body. Doctors and researchers use MRI practices to support medical research and operations. Doctors can use MRI scan on the brain (**neuroimaging**) to look for Blood vessel damage, brain injury, cancer, multiple sclerosis, spinal cord injuries, and stroke. [13]

The detection of physical matter (e.g. **grey matter**) in the neuroimages can then be measured and converted into a subsequent dataset through the use of MRI scans. This is done by applying **structural magnetic resonance imaging** (**sMRI**). This technique is a non-invasive method for examining the anatomy and pathology of the brain which produces images which can be used for clinical radiological reporting as well as for detailed analysis. This is a different technique from using **functional magnetic resonance imaging** (**fMRI**), which is applied to examine brain activity.

In this project, the neuroimaging dataset acquired, is from s**MRI with values of grey matter volume (mm^3) in 13 brain regions** as-well as the intercranial volume per participant. The analysed brain regions in the supplied dataset are:

- Intracranial volume (**ICV**)
- Left & Right hemisphere Frontal-lobe (**LH_Frontal, RH_Frontal**)
- Left & Right hemisphere Anterior Cingulate Cortex (**LH_ACC, RH_ACC**)
- Left & Right hemisphere Hippocampus (**LH_Hippo, RH_Hippo**)
- Total Corpus-callosum (**CC_Total**)
- Left & Right hemisphere Amygdala (**LH_Amygdala, RH_Amygdala**)
- Left & Right hemisphere Caudate (**LH_Caudata, RH_Caudata**)
- Left & Right hemisphere Putamen (**LH_Putamen, RH_Putamen**)

HIV's Effect on Neuroanatomy

Research into the effects on HIV on the brain has previously been studied in great detail. In-order to understand the outcomes expected from this project it is important to look at previous studies and see how they compare. Individuals' with HIV-infected brains are structurally different in comparison to equivalent healthy control brains. A considerable range of neuroimaging studies have assessed this with standard structural MRI showing subtle but significant relationships in people with HIV. [14]

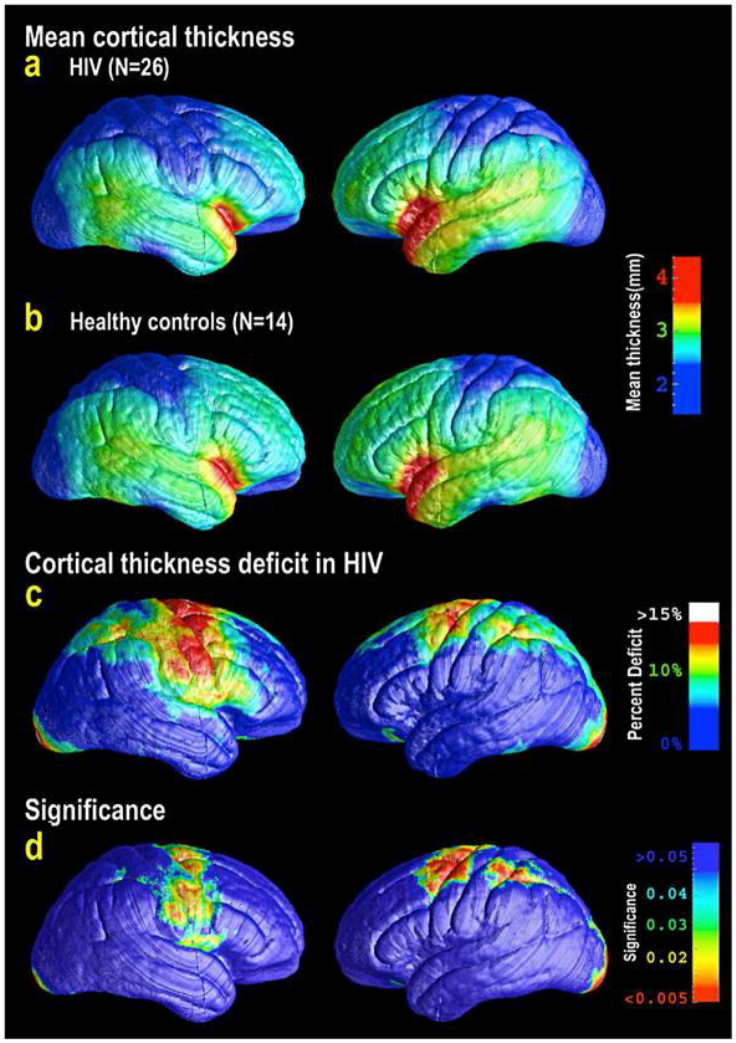

Figure 3: Cortical grey matter thinning shown in HIV+ individuals [15]

Figure 3 shows a study of cortical grey matter volume thickness using MRI for a group of HIV positive individuals, compared to a group of similar controls. The HIV positive group is shown to have a thinner cortex in the **sensorimotor regions** and some motor regions of the **frontal lobes**. The degree of **grey matter deterioration** also related to the neurocognitive performance of said patients. This research paper also shows that the intensity of grey matter reduction corresponded with a decreased CD4 count/detectable viral load in those participants. [15]

In a study published by Oxford University Press for the Infectious Diseases Society of America [5] we see that **HIV infection was associated with lower grey matter volume** and cognitive impairment. Overall grey matter volume was found to be lower in HIV positive individuals with deterioration taking place predominantly in the intracalcarine and supracalcarine cortices (within the Optical-lobe). This was **present in approximately 20% of cases** and is evidently associated with white matter abnormalities too.

Project Data

In this project, I analysed neuroimaging data acquired using a 3T Magnetom MRI scanner. Even with effective therapy, individuals who are HIV-infected continue to demonstrate ongoing aberrations in white and grey matter. An increase in brain white matter and subcortical grey matter abnormalities

are also linked to immunodeficiency recovery among infected individuals. [7] The HIV disease in South Africa has also had a population **increase of 0.3%** from 2014, when the participants MRI data was originally collected. [6] This makes finding details and inferences more important than ever, as there may be an ever increasing number of HIV positive women who suffer in South Africa.

Other projects have also performed analyses into the supplied dataset. [7] Where an investigation into "Effects of HIV and childhood trauma on brain morphometry and neurocognitive function" was carried out to see if the neuroimaging data collected showed any significant inference. This study proved inconclusive from their research data.

Although machine learning has been applied on neuroimaging in earlier projects, no clear investigation has been carried out for HIV participants in this sample dataset. And there are not many other HIV datasets that have been applied to machine learning within neurology research. A similar project applied machine learning to predict brain age deterioration as a result of HIV. [16] This example of HIV neurology allow me to shape my project to fit the scientific standard required of neurology research.

Concepts: Data Analysis and Supervised Machine Learning

Data Pre-processing

Scaling, or rescaling, means to add or subtract a constant and then multiply or divide by a constant, in-order to change the units of measurement of the data, for example, to convert a temperature from Celsius to Fahrenheit.

Normalising most often means dividing by a norm of the vector. It also often refers to rescaling by the minimum and range of the vector, to make all the elements lie between 0 and 1 thus bringing all the values of numeric columns in the dataset to a common scale.

Standardising usually means subtracting a measure of location and dividing by a measure of scale. For example, if the vector contains random values with a Gaussian distribution, you might subtract the mean and divide by the standard deviation, thereby obtaining a "standard normal" random variable with mean 0 and standard deviation 1. [17]

A technique used in this project is **Min-Max normalisation**, which performs a linear transformation on original data. It does this by changing the min and max boundaries of the data attributes and normalising them to a new scale of [0,1]. This technique is useful because it preserves the relationship between the original data values of the dataset. So that no input values can go out-of-bounds and beyond the limit of normalisation which would cause the data to skew. [18]

T-tests

A t-test is used to compare the mean of two given samples where the samples are assumed to be a normal distribution. When the population parameters such as mean and standard deviation are not easily known it is best to use a t-test. T-tests are best used for hypothesis testing in medical data such that we compare a null hypothesis with an alternate hypothesis using the difference in means. The t-test used in this project is an **independent t-test** which compares the mean of two groups of independent variables.

The statistic for this hypothesis testing is called the **t-value**. This statistic shows the relationship of the difference between the two groups and the difference within the groups. Therefore, a greater t-value suggests that there is more variance between the groups and supports an alternate hypothesis.

T-tests also have a p-value associated with each t-statistic. This **p-value** indicates the probability that the t-statistic occurred by chance due to the sample data. A p-value of **less than 5% (< 0.05) is statistically significant**. As it indicates strong evidence for the alternate hypothesis as there is less than a 5% probability that the results from the data are randomly against the null hypothesis. [19]

Analysis of Variance (ANOVA)

In this project a **one-way ANOVA** was performed to test the significance of the classification models' results. The one-way ANOVA tests the null hypothesis that two or more groups have the same population mean. The test is applied to samples from two or more groups, possibly with differing sizes. It calculates the **F-values** by assessing the magnitude of variance between the groups against the variance within each group of samples, as seen in figure 4.

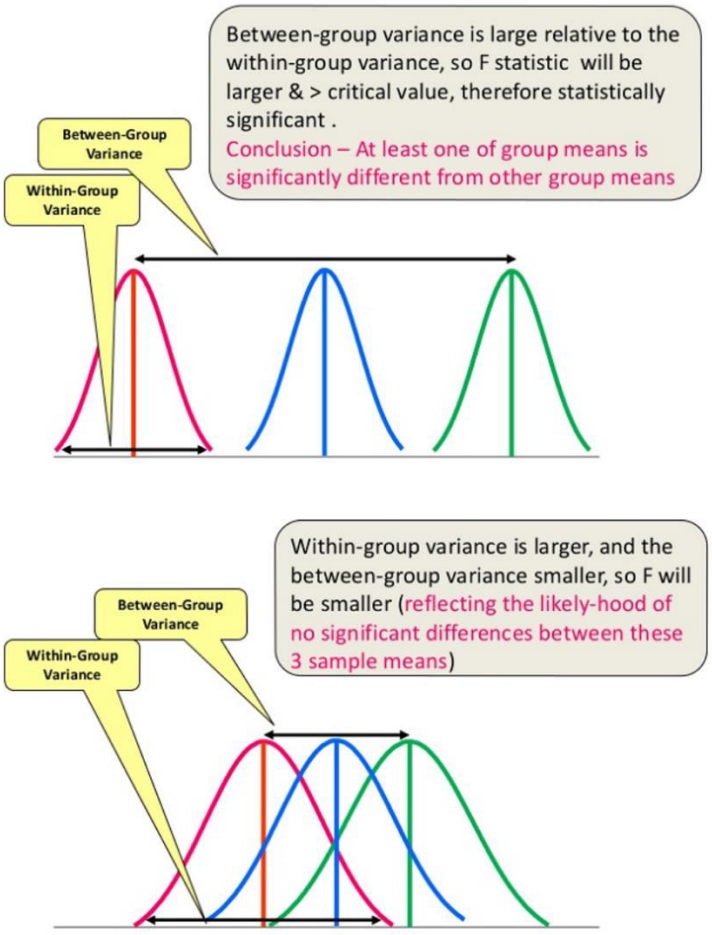

Figure 4: ANOVA Variance Between and Within Groups [20]

ANOVA notation depiction:

$$F(b, w) = x, \quad p = y$$

Where *b* represents the degrees of freedom between the groups, *w* signifies degrees of freedom within the groups. Whereas *x* represents the f-value and *y* signifies the p-value. [21]

Machine Learning

"Machine learning is the idea that there are generic algorithms that can tell you something interesting about a set of data without you having to write any custom code specific to the problem. Instead of writing code, you feed data to the generic algorithm and it builds its own logic based on the data." [22]

As a discipline in **Artificial Intelligence**, machine learning provides systems with the ability to understand and develop experience from given data without the need to be programmed in a precise manner. Machine Learning algorithms can be one of three separate categories: **reinforcement learning**, **unsupervised learning** & **supervised learning**.

Unsupervised Learning

Unsupervised learning uses neither labelled nor categorized data. This type of machine learning looks for previously undetected patterns in a data set without any pre-existing labels and with minimum human supervision. The goal in such unsupervised learning problems may be to discover groups of similar examples within the data, where it is called clustering, or to determine how the data is distributed in the space, known as density estimation. Therefore, it is self-organising which allows for density probability modelling. Although useful in certain circumstances, this project wants to find inferences for more in-depth analysis between the brain data of patients. Therefore, we want to look into supervised machine learning where we have influence over the results and more methodologies to employ.

Reinforcement Learning

"Reinforcement Learning (RL) is a type of machine learning technique that enables an agent to learn in an interactive environment by trial and error using feedback from its own actions and experiences." [23] Reinforcement learning uses mapping between input and output by employing rewards and punishments as signals for positive and negative behaviour within their models. Similarly to unsupervised learning, this project does not exercise reinforcement learning in its approach and implementation.

Supervised Learning

Supervised machine learning algorithms take what has been learned from experience with labelled data to make predictions about future data or events. When training a supervised learning algorithm, the training data of inputs and their corresponding correct outputs are given. The algorithm will investigate the data for patterns that indicate the desired output and then remembers to look for these patterns when testing. This trained model can then take in a new set of inputs and will attempt to determine which label the new inputs would be classified as by using the examples learned from the established training data. Supervised learning models are intended to predict the correct variable (**class**) for any new testing data using the experience gained from the training data. There are also different types of supervised machine learning. **Classification** and **Regression** together form supervised learning. This project will implement a supervised learning approach that focuses on classification techniques and methodologies.

Regression

Although not employed in this project, regression is a valuable form of machine learning. Regression algorithms aim at predicting input training data with a numerical output. A regression model attempts to find the important relationship between dependent and independent variables using a predictive statistical process. Regression algorithms can be used to predict a continuous number

such as sales, income, and test scores. Some regression models that can be used are: Linear Regression, Support Vector Regression and Random Forest Regressor.

Classification

Classification is the process of predicting the status of given data points. In this project we are trying to predict if a participant has HIV and if a HIV positive participant has a detectable viral load. During the training phase of a supervised learning model, the classification algorithm will look at the dataset of each category (**class**) and learn what trends and structure of the data apply for each class. The classification algorithm will then be able to take an input value and assign it to the class that it thinks the testing value fits into. It does this based on the tendencies identified in the training data. [24] Classification models can use various different algorithms to help classify the data. The focal classification algorithms highlighted in the project are: Linear Discriminant Analysis, Support Vector Machines, K-Nearest Neighbour, Logistic Regression and Random Forest.

Classes

In this project classes are the categorical variable that we are attempting to predict. This encompasses multiple variables over the course of the project. The main **class** in this project is the HIV status of participants (whether they are HIV negative or positive).

Features

This project uses various different variables from the obtained dataset and many of these variables will be used in the program as features. **Features** are the sets of variables that are used to make the prediction. For this project, the features are the MRI brain regions of participants, where we examine these values and learn their qualities. Then we attempt to predict the HIV status of other participants based on their MRI region values and qualities.

Classifiers

The **classifier** is an algorithm that maps from features to class. The classifiers in this project are classification functions imported from Scikit Learn. These classifiers attempt to make a prediction of class based on the feature data per participant. Each classifier uses a unique procedure to calculate the predicted class which will then be assessed against the actual class resulting in accuracy and performance metrics.

Fitting Data (Underfitting & Overfitting)

Overfitting and underfitting are problems often encountered by data scientists and can lead to a machine learning model with inadequate performance. **Underfitting** happens when a model is unable to model the training data, and unable to generalise new data. Whereas **overfitting** occurs when the model has studied the training data too much and learned every relationship to the point it decreases the performance. In-order to create a classification model with notable performance, it needs to be able to learn the relationships between the features and the patterns within the dataset to a significant intensity that is not too weighted towards the training data.

Performance Metrics

In-order to gauge a model's effectiveness, we need to evaluate its performance. Judging the top classification performance of any given model will differ depending on the problem being solved. Different metrics can be contrasted against other models to discover which of the classifiers are best at solving the problem. This project focuses on 5 key performance metrics:

- **Performance Accuracy**, the number of correct predictions divided by the total number of predictions (and multiplied by 100 to get percentage).

- **Precision**, the amount of predictions that were labelled as positive being predicted as positive.
- **Recall**, the predicted amount that were positive being actually positive.
- **F1 score**, a mean value from the precision and recall metrics.
- **ROC AUC score**, is how well the model is capable of distinguishing between classes, which is a more significant metric to use than accuracy alone.

Confusion Matrix

Confusion Matrix is another method to evaluate the performance of a given classifier. It can be used to find the correctness and accuracy of a model. It is visualised as a table with actual classifications as columns and predicted ones as rows. In a binary class confusion matrix using HIV status as the predictor, there are:

- **True positives** –number of correctly predicted samples that are HIV positive
- **True negatives** –number of correctly predicted samples that are HIV negative
- **False positives** –number of samples that are HIV negative but predicted as HIV positive
- **False negatives** –number of samples that are HIV positive but predicted as HIV negative

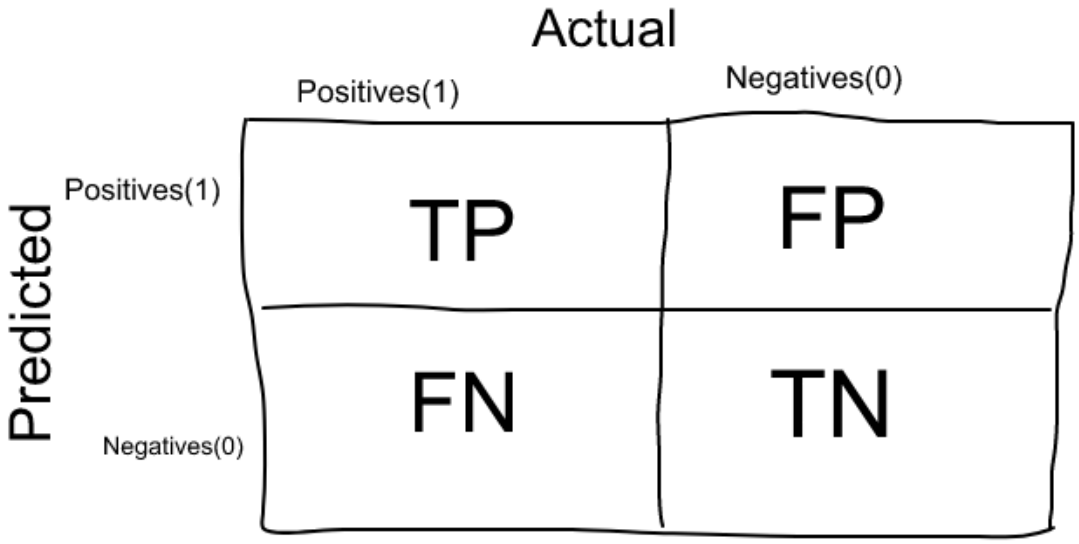

Figure 5: Confusion Matrix Example [25]

The 4 key values that can be determined from the 4 confusion matrix HIV categories are:

- Accuracy, in classification problems is the number of correct predictions made by the model over all predictions made
  Accuracy = (TP+TN)/(TP+FP+TN+FN)
- Precision measures what proportion of patients that we diagnosed as having HIV, have HIV
  Precision = TP/(TP+FP)
- Recall measures what proportion of patients that had HIV were predicted as having HIV
  Recall = TP/(TP+FN)
- F1 Score is a single score that represents both Precision and Recall
  F1 Score = Mean(Precision, Recall)

ROC AUC

Receiver Operating Characteristics (**ROC**) is the probability curve for our models and **Area Under the Curve** (**AUC**) represents the degree/measure of **separability**, detailing how well a model is at distinguishing the classes. A higher AUC means that the model is better at predicting HIV negative participants as negative and HIV positive participants as positive.

"An excellent model has AUC near to the 100 which means it has good measure of separability. A poor model has AUC near to the 0 which means it has worst measure of separability. In fact, it means it is reciprocating the result (predicting 0s as 1s and 1s as 0s). And when AUC is 50, it means model has no class separation capacity whatsoever." [26]

Therefore, we want our predictive classification models to have a score as far away from 50 as possible. If a score near 50 occurs, then it means the model has **no discrimination ability** to differentiate between HIV positive and HIV negative classes. In this project the ROC AUC scores are displayed on a scale of 0 to 100 rather than 0 to 1, with a corresponding value in parentheses adjacent to the ROC AUC score that depicts the value's distance from the inconclusive value of 50.

Python Programming

This project will be written in the programming language python. This project requires the utilisation of many core data science **Python libraries**, including: Anaconda, Jupyter Notebook, Scikit Learn, NumPy, pandas, Matplotlib, Seaborn & SciPy. More details can be found in the Implementation section: Python.

Cross-Validation

Overall, cross-validation facilitates better use of the MRI data, and it provides additional information for each of the implemented models' performance. This method helps us use the appropriate data in the different steps of the classification program. Allowing for real performance and helps mitigate the prospect of unsolicited side effects. Cross-Validation helps the project by giving assurance for the program in regard to challenges faced by many Data Science projects. [37]

Hyperparameters

All the classification algorithms used in this project have hyperparameters that allowed the behaviour of the algorithm to be tailored for the specific dataset. Hyperparameters are different from parameters, which are the internal coefficients or weights for a model found by the learning algorithm. Unlike parameters, hyperparameters are specified by the practitioner when configuring the model. [27] Implementing an exhaustive hyperparameter grid search allowed the sub-program to create the best model based on the training data and used cross-validation to prevent overfitting.

Approach

The initial aim for this project was to use data analysis and machine learning to help detect HIV in a neuroimaging dataset from south Africa. It was also decided to apply the same methods for viral load detectability. In this section of the report we will discuss the methodology, deliverables, assumptions, and hypothesis that took place before implementing a solution.

Project Methodology and Management

Data Science

The development process for data science projects will take a different approach to a standard software development lifecycle. This not only involves different techniques being implemented, but the use of a distinct design methodology as-well. When planning for this project I followed 8 steps that are commonly outlined in the of majority data science approaches. [28] [29] [30]

1. **Problem understanding**: Looking for a successful resolution of the problem. Provide an analytic solution by defining the problem, objectives, and requirements. Usually this is business focussed but for this project it focusses on more of a humanitarian project research perspective.
2. **Analytic approach**: After clearly establishing the problem, an analytic approach to solving it is created. Then a statistical and machine learning approach is designed to identify results and methods that achieve a desired outcome.
3. **Data requirements**: Depending on the approach, the data requirements are defined from the analytic methods required.
4. **Data collection/sanitisation**: Once identified the data is structured to be relevant for creating a solution to said problem.
5. **Data understanding**: basic statistics and visualisation techniques are used to better facilitate an understanding of the dataset. These results can then be assessed to determine the quality of the particular dataset.
6. **Data processing**: Here the data is prepared to be applied in the modelling phase. This can include processes such as data cleaning, data fusion, data transforming, feature engineering predictor enhancing, as-well as many other steps to prepare the data for detailed analysis and machine learning.
7. **Modelling data**: Using the processed data to train and test predictive and descriptive qualities for whichever analytical model is being applied.
8. **Evaluation of data**: Most importantly, the different model's quality is appraised against how well it solves the initial problem. These results can take the form of statistical metrics, inferences and examples using tables, graphs and other visualisations generated from the predictive model.

Workshops

Before this project commenced there were a couple of useful workshop tutorials that went through the key approaches and implementation processes. Facilitated by the project supervisor, these workshops helped significantly with understanding basic key processes and methods needed for a successful project. This allowed for the project to be approached with pre-existing knowledge of the functionality desired in the program.

Agile Development Lifecycle

This project used and practiced an Agile approach to development and execution. Agile development is an adaptable framework, which is implemented by adhering to established methods and practices. [31] In-order to sustain the agile lifecycle, the project had to adhere to the agile core values:

- Individuals and interactions over processes and tools
- Working program over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

These principles were upheld through the use of several processes, including weekly supervisor meetings. During these meetings it was possible to report any findings by sharing the program and tangible results that were identified over the week. These interactions allowed for supervisor collaboration and assessment where they could identify any changes needed in the demonstrated program. The processes would then adapt to address the criticism and advance the project to the next phase. These meetings were the primary measure of progress when assessing the program as it also helped maintain a constant tempo of work. Having this continuous consideration about the technical quality of the program made for progressive project alterations and allowed for changes in requirements, even late in development. An example of this happened after meeting with representatives from Stellenbosch University over a Skype conference call. Discussions to help understand any customer requirements, areas to explore and validation of work already performed took place with the representatives. One of the new emerging requirements was to investigate the viral load detectability using the MRI data rather than focussing solely on the HIV Status of the participants.

These meetings conveyed information through face-to-face conversations to begin with, however due to the situation of the Covid-19 pandemic [32] we had to adapt and harness changes to the weekly meetings using advantageous software like Skype and Microsoft Teams. By allowing a self-organizing approach, the agile process was able to promote a more sustainable implementation over the course of the project. This meant that program and developer behaviours could be tuned accordingly at the regular intervals.

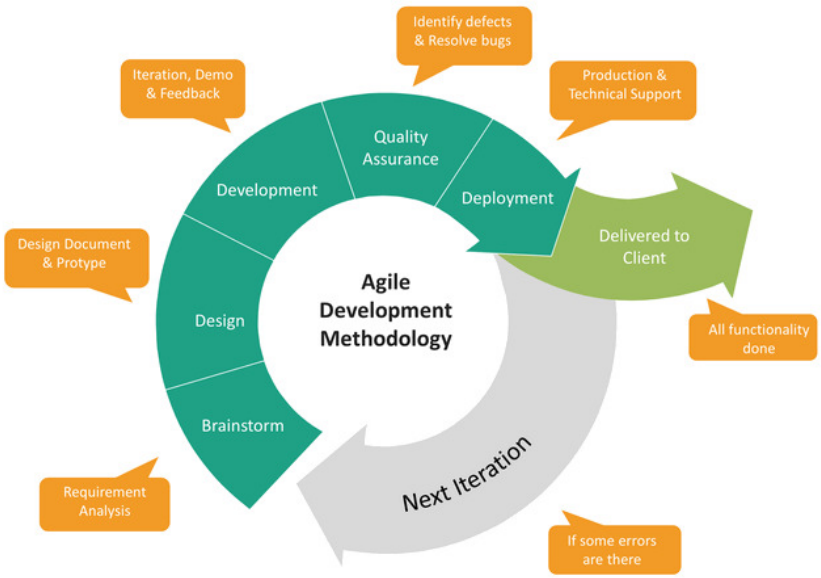

Figure 6: Agile Development Process [33]

In figure 6 we see the different phases involved in an agile development cycle. This project adhered to a similar approach, keeping the core values in mind. As outlined in the next sub-section, the project focussed on several key phases: **Plan**, **Explore/Model data**, **Implement Program**, **Develop Program**, **Evaluate** & **Document**.

It started by defining a model as simple as possible that carried out classification on the dataset. Next, a solution is developed to meets all minimum requirements for classification which was then tested. Finally, the performance of the program was evaluated, and reflection undertaken to identify areas to enhance. This software development lifecycle then starts again, with each new iteration of the cycle building upon the results of the last iteration. This agile approach allowed the project to avoid unnecessary work and **focus on the critical priorities** needed to meet the requirements in each iteration of the agile lifecycle.

Timeline

The initial plan for this project went over the predicted timescale, with a few changes needing to be applied later in the project. The main stages of this project consisted of 6 key phases adapted from the data science and agile methodology previously mentioned. These objectives are:

1. **Plan Project**: Here the HIV problem was identified and research into the disease and neuroimaging took place. The dataset was obtained, and an ethical consideration check took place. Objectives & milestones were also approved by the supervisor at this stage.
2. **Explore/Initialise Data**: This step involved looking through the sample dataset of participants and checking the demographics, size, and clinical characteristics (such as percentages, averages, frequency, standard deviation). The dataset was also sanitized to be relevant for the verdicts needed in the project.
3. **Initially Analyse Data**: data analysis and statistical techniques (Independent Students' T-test) used to find correlations, associations, and variance between the neuroimaging data. The preliminary data visualisations were then created from the data analysis using applications such as matplotlib & seaborn.
4. **Implement Predictive Learning**: Classification and regression machine learning models were then designed to handle the neuroimaging data. The predictive model was created, and a train/test split was applied from the HIV dataset.
5. **Develop Predictive Models**: In this phase the machine learning model is developed with applied hyperparameter tuning and cross validation to minimise error such as under/overfitting.
6. **Evaluate & Model Results**: This step was used for judging the model's peak classification performance and then compared with different classifiers (models) that have also undergone the same treatment. Findings are then evaluated and assessed as appropriate, using visualisations libraries like seaborn.
7. **Document**: The final step is to gather all the knowledge and significant discoveries to write up in this final report.

Deliverables

The initial deliverables for my project were:

- ***Initial Plan***
- ***Data Visualisations and Analytics from HIV Dataset***
- ***Functional Convolutional Neural Network Trained on the Neuroimaging Data***

- ***Results Identified from the Analysis and Predictive Learning***
- ***Final Report***

Some of these have since changed to efficiently streamline the project and focus in areas that are of more significance now that better requirements were established from the customer (Stellenbosch University research team). Excluding the initial plan that was already submitted and the final report that is currently being written, the updated deliverables are:

- **Initial Data Visualisations and Analytics from HIV Dataset**:
  Here we fulfilled the objective of initially analysing the data and produced the base analysis that progressed the project to the next stage. This was due to be completed by week 4 in the initial plan and remained the same in the final approach.

- **Functional Supervised Learning Classification Models Trained on the Neuroimaging Data**:
  As stated in the project title, we used supervised machine learning techniques to investigate neuroimaging data for HIV participants. This deliverable fulfils the requirement to implement multiple predictive learning models and develop them to best solve the classification problem. This deliverable was initially due for the start of Easter but ended-up moving due to the Covid-19 situation.

- **Results Identified, Visualized and Evaluated from the Analysis and Predictive Learning**:
  In this step of the approach the evaluation & modelling of results was completed. Accomplishing this objective is vital to the overall project as it is the foundation for communicating a solution to the problem. This objective was scheduled to be completed by week 10, however it was also pushed back as-well, due to further developments.

Assumptions

In this project many assumptions have been made including **customer strategy**, where it was assumed that the Stellenbosch university research team wanted a predictive learning model applied to the HIV data. This assumption was predominantly correct however other hypotheses to explore were established as extra functionality too. **Technology-based** assumptions such as access to hardware (laptop), software (Jupyter, Google colab) and the internet were also considered to be available over the course of this project and will not be restricted at any time. This proved to be the case and no adjustments were needed in this regard. It was also assumed that location and environmental factors were accessible over the course of this project and no impact to usual life would be present. However, due to the Covid-19 situation there have been many blocks to the regular scheduled management of this project. Further details can be found in the last section: Obstacles to Project.

The focal assumption of this project is that the dataset is an accurate reflection of previously proven theories. Therefore, this dataset will be able to show there is a clear difference in the neuroanatomy of HIV positive individuals by looking for neuronal inflammation and neuronal death. Inference from the data is assumed to be **sufficiently clear enough to prove the hypotheses** using machine learning models and data analysis.

The Dataset

In this project, there were two CSV files that contained the neuroimaging datasets. The main file is a baseline neuroimaging dataset with **124 participants**, with an even split of **HIV-positive (*1, n=62*)** and **HIV-negative (*0, n=62*)** sample size. The second dataset contains the follow-up neuroimaging

data for **60** of the original 124 participants with an uneven split of **HIV-positive (*1, n=26*) and HIV-negative (*0, n=34*)** sample size. This comma-separated values (CSV) file contains both the original values and the follow-up values for their current condition. The majority of the fields in these files were redundant, for example gender and marital status. Therefore, when the data is implemented into the program's dataframes, only the relevant features and classes are selected. These focal fields included, are displayed in **table 1**.

Class/Feature Name	Description	Variable Values
HIV_Status	HIV class condition	0 = HIV-negative, 1 = HIV-positive
ICV	Intracranial volume (ICV)	mm^3
LH_Frontal_vol	Left hemisphere Frontal-lobe (LH_Frontal)	grey matter mm^3
RH_Frontal_vol	Right hemisphere Frontal-lobe (RH_Frontal)	grey matter mm^3
LH_ACC	Left hemisphere Anterior Cingulate Cortex (LH_ACC)	grey matter mm^3
RH_ACC	Right hemisphere Anterior Cingulate Cortex (RH_ACC)	grey matter mm^3
LH_Hippo_vol	Left hemisphere Hippocampus (LH_Hippo)	grey matter mm^3
RH_Hippo_vol	Right hemisphere Hippocampus (RH_Hippo)	grey matter mm^3
CC_Total	Total Corpus-callosum (CC_Total)	grey matter mm mm^3
LH_Amygdala_vol	Left hemisphere Amygdala (LH_Amygdala)	grey matter mm^3
RH_Amygdala_vol	Right hemisphere Amygdala (RH_Amygdala)	grey matter mm^3
LH_Caudata_vol	Left hemisphere Caudate (LH_Caudata)	grey matter mm^3
RH_Caudata_vol	Right hemisphere Caudate (RH_Caudata)	grey matter mm^3
LH_Putamen_vol	Left hemisphere Putamen (LH_Putamen)	grey matter mm^3
RH_Putamen_vol	Right hemisphere Putamen (RH_Putamen)	grey matter mm^3
ARV_Treatment	Is the participant on ARV treatment (ART)	1 = Yes, 2 = No
CD4_Count	Number of CD4 cells detectable in participant blood	$CD4/ml^3$
Viral_Load	Detectability of participants HIV viral load	1 = Lower than the detectable, 1 = Slightly detectable (< 40 cps/ ml^3), Other values in cps/ ml^3

Table 1: Main Classes and Features in Dataset

Hypothesis

In this project the hypothesis is represented as a speculative statement regarding the relationship between several key variables. In this project we want to see if areas of the brain (neuroanatomy) signify HIV infection and if neuroanatomy can indicate viral load detectability. These statements attempt to predict an anticipated outcome that can be tested and revealed when evaluating the results. The hypothesis should directly correlate to the aims, therefore there are **2 core hypotheses** declared in this project. In-order to maintain excellence in this research endeavour, the project must make sure that the hypotheses adhere to 4 essential details, as defined by Amy Morin. [34]

i. "Does your hypothesis focus on something that you can actually test?"
ii. "Does your hypothesis include both an independent and dependent variable?"

iii. “Can you manipulate the variables?”
iv. “Can your hypothesis be tested without violating ethical standards?”

HIV Status Hypotheses

The null HIV hypothesis is that: neuroimaging grey matter provided by the dataset is the same between HIV positive and negative participants.

The **alternate HIV hypothesis** is that: neuroimaging grey matter provided by the dataset is **different** between HIV positive and negative participants.

Viral Load Hypotheses

The null viral load hypothesis is that: neuroimaging grey matter provided by the dataset is the same between the undetectable and highly detectable viral loads of HIV positive participants.

The **alternate viral load hypothesis** is that: neuroimaging grey matter provided by the dataset is **different** between the undetectable and highly detectable viral loads of HIV positive participants.

In this project we are looking to prove the alternate hypothesis in both instances using the HIV dataset provided by Stellenbosch University. Considering the 4 essential details, we know that both hypotheses can be tested using data analysis and supervised learning techniques. The independent variables are the MRI grey matter regions (the features), and the dependant variables are either the HIV status or the viral load of the participant (the class) respectably. We can manipulate which variables are used by adjusting the data-frames for feature selection to choose which predictor we are looking for. Finally, both hypotheses have gone through ethical consideration and do not breach any ethical standards in this project.

Other Hypotheses Investigated

In this project there were 2 datasets that could be used: baseline MRI data & follow-up MRI data. After discussion with supervisors and clients, the main focus of this project would be on predicting the HIV status & viral load detectability from the baseline data due to its sample size. However, given time and significant enough reason, several other theories could be investigated using the follow-up dataset. These queries are:

- Can participants' whose **viral load detectability changed** between the base and follow-up acquisition be predicted from the difference in their grey matter?
- Are there enough changes in grey matter to predict participants' that **changed antiviral treatment (ART)** from the follow-up data?

If significant enough, testing these questions would allow us to use supervised machine learning and data analysis to see which brain regions altered as a result of their condition changing.

Implementation

In this section of the report I will be discussing the processes taken to implement several workable solutions that analyse and predict desirable results that prove/disprove the hypotheses. This project involves creating a python-based program that uses data analysis and supervised machine learning techniques to investigate if there are inferences of HIV status and viral load detectability in the neuroimages of South African women.

Program Architecture

The program implemented is designed using several popular methods that are utilized by data scientists from around the world.

Python

The program created during this project was written in the programming language Python. Specifically python **version 3.6.9**. The decision to use python as the language of choice was due to 3 main points:

1. The **project proposal** was established as being undertaken in python. This is because the project supervisor has an extensive knowledge of the python programming language and was able to deliver detailed workshops in the field of data science regarding python. This significantly helped to kickstart the project in the right direction.
2. Another reason python was chosen was due to the **pre-existing knowledge** that the project implementer already had regarding this programming language. This meant that extensive learning and research was not required before and during the project. Or at-least not as much as there would be if a new, unknown language had to be used for the project such as R.
3. The final justification as to why python was used for this project is because of its **reputation in the data science community**. Python is renowned by data scientists as it is known to be one of the easiest learn and develop when getting started as a data scientist. Python also has many substantially useful packages and support within the community that are free to use. This facilitated the project, as it can utilise detailed data analysis techniques, comprehensive supervised learning models and aesthetic graph visualisations.

On the other hand, python also has its own disadvantages. One of which is the fact that it does not have great documentation. This is especially the case when you compare it to other programming languages like PHP and Java. However, the advantages of using python in this project outweigh any disadvantages that it may have.

Anaconda

When initially implementing the python code, Anaconda was used to install and manage the majority of desirable packages that are being used in the project program. Installing anaconda allowed access to a multitude of useful packages. Using the Anaconda command prompt shell, the project could run Jupyter notebooks and the essential packages that were installed along with it.

Some of the more obscure packages had to be manually installed (e.g. Seaborn) as they are not installed by default using Anaconda. However, all the other required packages were already setup thanks to the assistance of Anaconda.

Initially, Anaconda was chosen because it was recommended by the project supervisor. It is also well known as an industry standard tool in data science that has been used to implement previous coursework projects.

Jupyter

The most notable package that is installed with Anaconda is Jupyter notebook. The program associated with this project is written exclusively inside of a Jupyter notebook as it has many advantageous functionalities. Jupyter notebook was initially developed for **data science applications** written in Python and is useful in variety of different projects. It allows for easy data visualisations as Jupyter notebook **developers often publish their techniques** and share their code and datasets. One of the most advantageous uses with the Jupyter notebooks is its **live code interactions**. The notebook's code is not just static due to the fact that it can be modified and then re-processed with the program returning the outcome instantly and directly into the notebook tab.

These processes are all very useful for implementing a program but when it comes to creating a project program, Jupyter notebooks allow for embedded documenting. In this project the code often has titles and comments that explains areas that were being investigated and their functions. This all happens while the user gets to see **dynamic feedback of results and visualisations**.

Google Colab

After the project started, the project supervisor wanted to use Google Colab as the main form of file sharing and version control in this project. This is due to Google Colab's functionality in code sharing capabilities, as they allow users with access to the repository to view code, execute it, and display the results directly in their web browser of choice. As suggested in the product name, it also **supports collaboration** between teams of developers working on Jupyter notebooks. This means that Google Colab could, in future, expand implementation to a wider team if necessary, in a later project.

Google Colab also has free inbuilt functionality that link the Jupyter notebook to a Google drive account where it will have back-ups and version control methodologies employed to a professional standard. One of the most practical characteristics that makes Google Colab a professional industry standard in the data scientist community, is that it has **dedicated GPU** (Graphics processing unit). This makes Google Colab especially useful to projects that require a significant enough chunk of processing power that may not be available on a user's own computer or restrict the user when processing a program. This is why Google Colab is especially useful when running programs that have machine learning and/or other exhaustive functions like deep learning.

I suggest that anyone who runs **this project's program to use Google Colab** for optimal results as it will ignore update errors and will not burden their own computer.

Program Design

In this project, the program was setup in a Jupyter notebook following the methodology that was decided in the approach. The structure of the program goes as follows:

- **Program Head**: import packages
- **Data Initialisation**: upload, characterise, normalize, define dataframes
- **Data Exploration and Analysis**: demographics, descriptive analysis (averages, frequency, range), neuroimaging data distribution, independent t-test (t-value, p-value), feature selection plot, pair-plot

- **Supervised Machine Learning**: train/test split & cross validation, linear discriminant analysis, support vector machine, k-nearest neighbour, logistic regression, random forest, classification accuracy (performance metrics)
- **MRI Feature Selection Supervised Machine Learning**: train/test split & cross validation, linear discriminant analysis, support vector machine, k-nearest neighbour, logistic regression, random forest, classification accuracy (performance metrics)

The sections *Data Exploration and Analysis*, *Supervised Machine Learning* are also duplicated for the viral load class in-order to determine the other core hypothesis. The first series of these sections investigate if HIV status can be predicted while the second series investigates if the viral load detectability can be predicted using the neuroimaging dataset. Full details, explanations, illustrations, and clarifications about the implementation can be found in the subsequent sections of this report.

Style and Formatting

When implementing diagrams, the program used the python library Seaborn in-order to adhere to presentation standards and **maintain interpretability**. Areas that were addressed when designing and creating the visualisation included: colour palettes, font size, font weight, labels, titles, subplots, regression lines, axis lines, error bar/confidence intervals and plot orientation (vertical/horizontal).

Changes from Initial Plan

The program adapted over the course of the project to fit a more suitable structure for the required results. The data science methodology of the newly designed program tied nicely with the proposed format that was originally mentioned within the initial plan. [35] the original format was to Explore/Initialise data which was performed in the first sections of the program. Then it was planned to model data visualisation which was performed in the data explorations and analysis section as-

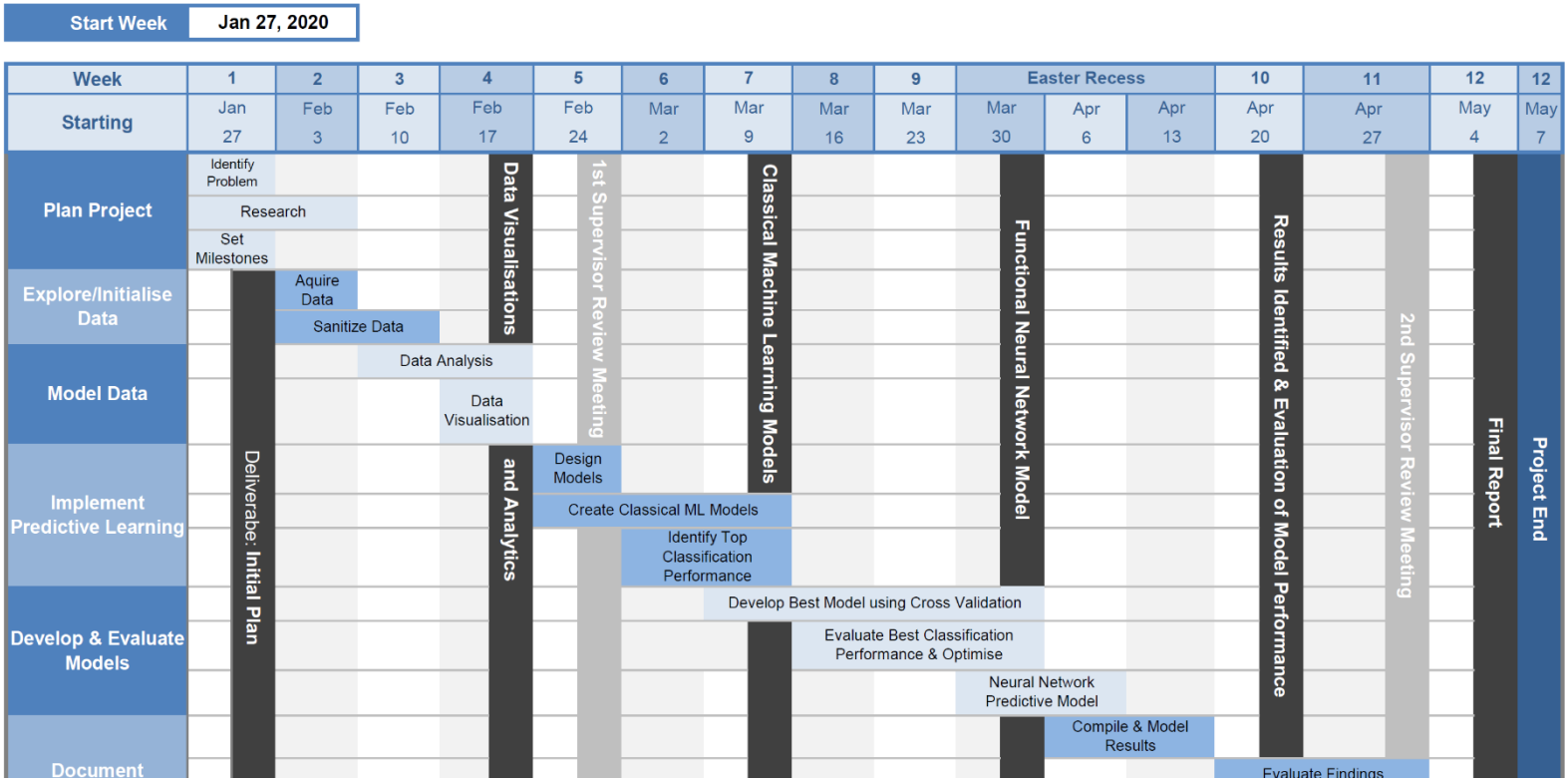

Figure 7: Project Timescale Gantt Chart [35]

well. The next stage of the plan involved implementing predictive learning using classical machine learning models and identifying the top classification performance. This objective was addressed in the supervised machine learning segment of the program where the different models were implemented and tested for their performance.

Each section had its own part in the program until the development phase of the plan. The development objectives and deliverables did change once the requirements needed in the program became clearer. The deliverable in this section changed from the creation of a neural network to the development of the pre-existing predictive classifiers. They were instead developed through the implementation of hyperparameter tuning, nested cross-validation, and feature selection. This new development process occurred within the supervised machine learning segment of the program over-riding the previous code that was used. And then a new part of the program "MRI Feature Selection Supervised Machine Learning" was added to investigate feature selection of the MRI data. This update to the deliverables was ill-fated but still aligned with the core aims of the project and accomplish the aim "Develop the predictive model to closely predict if a participant has HIV based on neuroimaging data".

Program Head

In the head section of the program, the required python libraries and packages are imported and defined for use throughout the program. Python is very advantageous, having large standard libraries that encompass many beneficial programming operations. Unlike other programming languages, the procedures coded into Python are already scripted. Python's simplicity is appealing for many data scientists who build machine learning libraries or develop existing ones. As a result of Python's extensive and effective library collections, it has become one of the best gateways for a data scientist to develop their machine learning skills.

```
[ ]  import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     import scipy as sp
     import sklearn
     import tensorflow as tf
     from tensorflow.keras import models, layers
     from google.colab import files

     from sklearn import datasets, metrics, preprocessing, linear_model, discriminant_analysis, model_selection, svm, ensemble, model_selection, neighbors
     from scipy import stats
```

Figure 8: Python libraries used in program [50]

These libraries are proven to be useful tools to any data scientist looking to use python programming in their projects. Most notably, Pandas, Seaborn and Scikit Learn (sklearn) are core packages that have been heavily integrated into this program.

Scikit Learn has many beneficial uses for data scientists due to its free usage, ease of use. Since scikit-learn is distributed under BSD license, it is free to use without any legal limitations for the program. Many industries and data science research projects use scikit-learn in their programs due to its easy use and lack of issues when performing complex processes.

Scikit Learn has also shown great versatility, support, and useful documentation from the data science community. The tool helps solve a variety of problems including classifying diseases and analysing neuroimages. The best feature available with scikit learn is it's detailed documentation, which is accessible on their website [36] and assists users in integrating scikit-learn with their own

datasets, programs and platforms. The tool also has a global community that are often able to help users if they happen to encounter any issues or errors with the tool.

Python alone is not convenient when it comes to investigating data analysis. However, Pandas offers many beneficial features and is the most common used library in this project as it enables data analysis techniques with collaboration from other tools. By combining pandas with libraries in this project an environment is created that supports analysis by enhancing the program's productivity and performance. Pandas has some of the most proficient data manipulation functions which allow the project to use dataframe structures for storing and displaying datasets. Pandas is also able to help solves regression and classification problems using statsmodels and scikit-learn by applying a dimensional data structures to the dataset. Pandas can also read and write data from formats including plain text, Comma Separated Values (CSV) and Relational Databases, making it the best data structure library for this project.

Seaborn is also one of the most important libraries used in the program, as it is used in the main visualisation code. The main advantage of using seaborn is its ease of use regarding graph plots and aesthetics. The aesthetics are much more visually appealing than matplotlib as it provides a wide library of easily customizable styles and pallets. This includes styles that differentiate qualitative, diverging, and sequential colour palettes for the satisfying visuals. It has plenty of valuable documentation that can be found on their website with many other useful tutorials and examples. [37]

Initializing Data

In the initializing section of the program, the neuroimaging data is uploaded from its source format type (CSV) and placed in Google Colab's temporary memory store (using Google Drive). Once placed in memory, the program can then read the neuroimaging dataset using pandas' *read_csv* function. This allows the program to save the uploaded data a variable that we can manipulate and store as the best structure.

Characteristics

The data characteristics can now be examined to make sure that it contains all data that is relevant to this project. Extracted is the number of participants, the amount of HIV positive participants and the columns/features (clinical characterises) that are available in the dataset.

This cell in the Jupyter notebook is useful for 2 main reasons. Firstly, it gives vital information about what data was contained in the uploaded file and which fields/columns of the data is being analysed in the program. The second useful note is that this tests the data to make sure it has all the necessary size, features and diversity to implement into the program. This includes knowing that there are **62 HIV positive** participants as indicated but the "HIV_Status" column and also that there are **13 MRI brain regions** we can use as features in this project.

Normalisation and Standardisation

When discussing the data with the project supervisor it remained important that the data be properly standardised for the participants when looking through the values that were present. In neuroanatomy there are many differences between brains, as this project is trying to prove with HIV. And it is important that the program addresses the fact that there is great deviation between the grey matter volume of each participant.

Because the **variation between participants intracranial volume (ICV) was significant**, a meeting was setup with the supervisors of the project to clear up any issues identified from the dataset. After discussion, it was determined that the ICV variable deviated significantly because the value is

derived from not only the grey and white matter volume, but from all the excess neuronal volume, fatty volume, plasma volume and other volume factors. However, they still recommended standardising the dataset using the ICV because the methodology and process of obtaining the ICV values remained constant for all participants and was reliable for regulating the data.

Therefore, it was decided that the program would implement a standardisation function that would assist in normalizing the dataset. This was done by **dividing the 13 core brain regions for each participant by their corresponding intracranial volume**, which was one of the key values stored in a column named "ICV". This was the first step to normalising the data, however, each of the 13 brain regions values were comparatively small when looking at the size of a participants ICV. This meant that the standardised variable for each of the brain regions in the dataset became uninterpretable due to how small the value was. The value for each brain region went from, for example 911 to 0.000940932 when divided by an ICV of 968189.35. Because the variables calculated from this standardisation were indistinct, the data values needed to be scaled as appropriate.

Scaling

Initially, the variables were scaled by converting the values to base 10 using a logarithmic function. This was attempted because one of the fields in dataset had scaled the ICV values to base 10 using the same methodology. This would mean that a value such as 0.000940932 would then become -3.03 (rounded) which is considerably more interpretable than it was previously, therefore it can be used to visualize the brain region distribution more appropriately. This was soon dismissed as a viable method of scaling when looking at the brain region distribution using the base 10 scale. This was because **the scaled data had become inaccurate** as a result of scaling to base 10. Scaling using this methodology caused the data to skew and become unbalanced.

Consequently, another method had to be employed. The data had already been standardized but now needed a new scaler instead of using logarithmic scaling. Fortunately, scikit learn have a pre-processing module that can use various normalisation techniques on the dataset. In this module there is a function called *MinMaxScaler* which scaled the brain regions (features) to lie between a minimum and maximum value, so that the maximum absolute value of each feature is scaled to unit size. [38] Here, the features were **scaled to between the values 0 and 1** for each brain region using a for loop over the indexed rows.

Dataframes

The final step when initialising the data was to define the features and store then in separate pandas dataframes, so that they are easily accessible when called later in the program. Although there are many more dataframes that are present throughout the program, these particular dataframes contain universal features that are not only relevant for many sections of the project but would need to be continuously redefined due to their excessive use. The main dataframe defined in this program is the *mriDF* which **contained the HIV status and MRI regions** for each of the 124 participants. This beginning of the focal dataframe was then displayed using the pandas *head* function.

Data Exploration & Analysis

In this stage of the program, the participants neuroimaging data was examined for key associations. This analysis was carried out for both HIV status inference as-well as viral load detectability, with initial investigations into the viability of the follow-up data.

Descriptive Statistical Characteristics

During this step of implementation, the program analysed the dataframes to calculate the statistical values that could show correlations, associations, and variance between the neuroimaging data.

First, the data is set into a pandas dataframe where the program extracts the core numerical data values for each feature, using the pandas *describe* function. In this function, a summary of the central tendencies, dispersion, and shape of distribution is displayed in a tabular format. This function also ignores not a number (NaN) values that would have been placed in fields like the participation ID. As a result, a table of useful values are displayed that depict the **count, mean, standard deviation, quartiles (0.25, 0.5, 0.75 percentiles) and range (min/max)** for each feature of the dataset.

Next, the program split the data by HIV Status (using pandas *groupby* function) and the mean values are calculated for each feature in the dataset. This will allow the user to see the fundamental difference for each brain region between the HIV positive and negative participants.

In-order to make the data more interpretable, the values from the descriptive tabular format were visualised in a **boxplot graph**. This was performed through the use of the visualisation library Seaborn, using the *boxplot* function. This plot describes the distribution of normalised MRI regions, therefore testing to see how much of a normal distribution is present and compares the standardised grey matter volume per brain region. After this step in the implementation, the brain region distribution was verified as normal by the project supervisors and researchers in South Africa. This was through the use of a Skype meeting, where discussions on the data characteristics allowed us to move ahead with the next stages of the data analysis.

Initial Investigations of Other Hypotheses

Along with predicting HIV, this project wants to see if there is a probability to predict viral load and changes in viral load/ART from the follow-up data. In this step of the implementation, the program counts the relevant variables (classes) that would be used for classification to determine whether those hypotheses are significant enough to be investigated in their entirety. This assessment was performed using a simplistic seaborn graph function called *countplot*. These visualisations will describe the quantity of participants that can be used in the train/test split for the classifier models. From here, if there are enough participants to use, the program will apply the data analysis and supervised learning techniques to test those hypotheses. After this investigation, the data limitations were made clear for several of the hypothesis. At this stage, any **investigation into the follow-up data was deemed insignificant** and therefore was not used in the rest of the implementation. However, even though the viral load baseline data had large variation between the sample size, it still had enough participants that could be investigated with a stratified implementation.

Independent T-Test

I used the *independent t-test* function from the stats module in the SciPy library. This function calculates a T-test from the means of two independent samples of scores. In this case it is for HIV positive and HIV negative brain regions. With this argument, we test for the 2 hypotheses mentioned in the approach section: HIV Status Hypotheses. This statistical test needed to be performed per brain region to **determine the significance that each brain region could help predict HIV status**. The original distribution of MRI regions was not Gaussian (bell curve) and therefore had to be normalised in-order to be subjected to the independent t-test. After the initialisation stage of the program, the data was normalised to create the required distributed as validated in the MRI region distribution visualisation. [17] The SciPy t-test functioned required 2 parameters, a list of values from the HIV positive MRI region, and the list of values from HIV negative participants for that same MRI region. The t-test would then compare the means of said MRI regions to determine the t-value & p-value. It would then perform another t-test for a different MRI region in the dataset until all brain regions have been statistically analysed for their significance.

In the program, this is done using pandas dataframes in a *for loop*. First the dataset is split by HIV status by defining a new dataframes for each status. Then the for loop indexes each relevant column of the dataframe to extract a list of the values for the positive and negative participants. Then these are statistically examined with the *stats.ttest_ind* function, where 2 values are returned and appended to a list so they can be visualised and evaluated in the next stage of the program. One list containing the **t-values** and the other list containing the **p-values** of each MRI region.

The same implementation was also applied using the viral load status as the split to help determine if there was any statistical significance between the MRI regions regarding viral load detectability.

A bar chart showing the significance of each brain region was then produced to visualise the t-values, p-values, and statistical significance for any of the MRI regions (features). These visualisations used Seaborn's *barplot* function and the statistical significance was indicated using a line positioned on the y axis at 0.05, indicating **the <0.05 statistical significance** of any brain regions. This line was then annotated with an arrow in-order to explain its meaning. At this point, there were only statistically significant MRI regions in regard to the HIV hypothesis. Therefore, **feature selection** would only be carried out on the HIV dataset.

Feature Selection Investigation

As a result of the independent t-test results, the **statistically significant brain regions** were identified and were further examined. Selecting and then analysing these features in greater detail allowed the project to check for further correlations that support the relevant hypothesis.

Feature selection is a widely used methodology in analysis and machine learning. Applying supervised learning to a dataset of just the 4 core features that were indicated by the t-test will increase performance. "[It] is one of the core concepts in machine learning which hugely impacts the performance of your model. The data features that you use to train your machine learning models have a huge influence on the performance you can achieve." [39]

There are many reasons why this program implemented feature selection. One of these benefits was the reduction in overfitting because there is less superfluous data and therefore less chance to draw conclusions from the noise of said data. Feature selection also **improves accuracy and reduces training time** as a result of having a reduced amount of data. As there is less undesirable data that requires processing in the program. [39]

More data analysis techniques were applied to the selected MRI regions to see the differences in a visualised format. This included a *scatterplot*, where the grey matter of the **4 significant brain regions** was plot against the intracranial volume separated by HIV status, and a line of regression added to help determine the degree of correlation. Another method used on the selected features was a *pairplot* graph that **shows the pairwise relationships** for those brain regions, separated by HIV status.

With the results of which brain regions showed statistical significance, a separate supervised machine learning segment was placed in the program to see if it performs in a more reliable and accuracy manner. This additional segment to the program uses the same code from the original supervised learning (classification) segment of the program. The data trained and tested in the program code consists of only the 4 significant brain regions rather than all 13 that were present in the previous data.

There was no statistical significance shown for the brain regions of undetectable and detectable viral loads in the HIV positive dataset. Therefore, when performing supervised learning to predict viral load detectability, feature selection was not implemented for any of the brain regions.

Supervised Machine Learning

During implementation, there were many different areas that needed to be included in the supervised machine learning stage of the program. The aim of this stage in implementation is to create predictive models that can predict **HIV status or Viral Load detectability** in the dataset.

In the initial plan, there was no mention of using feature selection for certain brain regions. Although, once statistical significance was identified that supported the hypothesis to predict HIV status from the MRI data, using feature selection seemed appropriate. Therefore, an MRI feature selection program was implemented to assess those particular brain regions for better results.

When implementing feature selection on the predictive models, there were minimal differences in the initially supervised learning models. The main changes required for feature selection was during the implementation of the training/testing data, where rather than allowing all brain region data to be used, the program **isolates the significant brain regions** to train and test in the different supervised machine learning models. The only other modifications from the initial predictive models were the variable names for the results. In-order to main integrity in the program, the variables that defined the final results were changed so that they can be easily called later in the program for evaluation, as required.

Defining Training and Testing Data Split

A well-established practice in the data science community is to split the data in-order to evaluate the model and make sure it performs appropriately on different elements. For this we use a test split which divides the original data into two groups. A training group that was used to train the different models and a test group using a new set derived from the rest of the data. So once the models are trained, it can be evaluated by checking if the model is consistent by accurately predicting the testing data. Typically, a data scientist will use a 80–20 or 70–30 percent train-test split as it is often reliable while being effective in performance. [40]

For the HIV status investigation, the features and class were defined by indexing the appropriate *mriDF* dataframe columns. When including feature selection, the same method was applied but only the 4 significant columns were indexed to be used as the features. The implementation of a train-test split began with random split using the Scikit Learn *model_selection.train_test_split* function. In this function the *test_size* was set to 0.25, meaning that the train-test split is **75% training data and 25% testing data**. So that the results created in the program were re-creatable the *random_state* of each of these functions was set to 1. This would be conducted with other random states in the Classification Accuracy sections of the program.

Defining Cross-Validation Methods

In the initial part of the supervised learning program, different cross-validation methods were also employed to test the accuracy of the classifiers in a way that approach the problem from multiple angles to prevent overfitting/underfitting of the predictive models. An advantage of using cross-validation include the utilising of all the MRI data, so **every participant will be used to train and test the classifiers** in the program. The usefulness of this method means that the program implements nested cross-validation functions when executing the pre-set functions with tuned hyperparameters.

In this program, 4 different types of cross-validation were used, each with their own approach to a train/test split. These cross-validation *model_selection* functions are also imported from the Scikit Learn *Metrics* module.

- **KFold**: which divides all the samples within the groups of samples known as folds. Training is learned using 3 of the 4 folds, and the last fold is excluded and then used the test split.
- **RepeatedStratifiedKFold**: repeats Stratified K-Fold n times with different randomisation in each repetition. It was used to run KFold n times, producing different splits in each repetition.
- **StratifiedShuffleSplit**: which is a variation of ShuffleSplit and returns stratified splits, i.e. which creates splits by preserving the same percentage for each target class as in the complete set.

Classification Models

The main predictive models that were implemented into the program are classification models which each use a different classifier function. In this program, each classifier has its own sub-program dedicated to hyperparameter tuning which is then evaluated against the other predictive model performances.

Initially, the viral load detectability was going to be a multiclass predictive model, where it would class the participants as **undetectable**, **slightly detectable** (< 40 cps/ml but still detectable), and **highly detectable** (> 40cps/ml). Though when running the train/test split, there were **not enough participants** who identified as slightly detectable (only 2 participants) to adequately use as a class in the classification models. Therefore, it was redundant to implement a multiclass system. So, the viral load detectability was turned into a binary class of undetectable & detectable viral load data.

Even though a binary class system was implemented, the sample data sizes between them was too great of a difference (with **8 undetectable** and **52 detectable** participants). Therefore, a stratified approach needed to be used to make sure that undetectable samples were being used in both the training and the testing split of the data. Supervised learning was still carried out even with the great disparity of the sampled data.

This project implemented many different classifiers that approached the problem using different algorithms and methodologies. The classifiers used in this program are:

- **Linear Discriminant Analysis (LDA)**
- **Support Vector Machine (SVM)**
- **K-Nearest Neighbour (KNN)**
- **Logistic Regression**
- **Random Forest**

These models were chosen as they are some of the most well-known and widely used classifiers in the data science community. [41] These classifiers on the surface can be easy, fast, and simple to implement and did not require extensive prior knowledge of the classifier or algorithms used in the classification sub-program. Several of the models also did not have many hyperparameters that required tuning. Random forest also would not require extensive pre-processing to be performed on the original dataset or make assumptions on the distribution of the data unlike other classifiers.

When implementing these sub-programs, a strict design was followed so that the program cells all had an engaging yet structured layout. This design was applied to each of the classifiers, with the only changing factors being the variable names, parameters, and the classifier function from Scikit

Learn itself. These sections use python comments to outline the headings and which part of the relevant code is being addressed.

The code structure for each of the classifier sub-programs is as follows:

1. **Defining Model and Parameters:**
   In this section of the sub-program, the imported classifier is defined along with any relevant parameters that need to be pre-defined for the hyperparameter grid searches (which are used to create the tuned models).
2. **Training Baseline Model with Random Split and Cross-Validation:**
   Here, the baseline classifier is trained (fit) with the random split and cross-validation methods specified in the test/train split cell before the classification sub-programs. With the cross-validation methods, the program returns a list of the scores for each of the implemented cross-validation processes. These baseline results are kept in their own variables to be called later in the sub-program.
3. **Set Hyperparameter Grid Search and Train:**
   In-order to create a model with optimised performance, the program uses the parameters from the first section in a dictionary to create a hyperparameter grid. This grid is then supplied to an exhaustive search function, to find the best estimators for the trained model. A parameter grid is also given to a random search function which allows for a comparative evaluation of a control result for the baseline and best estimator results. It does this by selecting a random set of parameters to use in the classification model. These grids searched models are then fit with the training data to minimise overfitting whilst finding the prime parameters for that model. These trained grid results are then set to their respective variable to be called later in the sub-program.
4. **Train and Assess the Tuned Models:**
   This section then takes the best estimator from the grid results and trains a model using the best hyperparameters from the exhaustive grid search. A cross-validation process is then employed with this tuned grid to evaluate a nested cross-validation method against the random split hyperparameter tuned model. Each of the models thus far have not be assessed against the test data to predict the desired class. The next part of this section produces the value for each score and assigns the results to their appropriate variables. Several of these variable results are then placed into corresponding dictionary dataframe which is will contain the key results for each classifier's tuned model. This is saved so it can be called in the accuracy and performance section of the sub-program.
5. **Display and Assign Results for Evaluation:**
   The last part of each sub-program is used to print out the results of the classifier models and concatenate the dataframe of results for each classifier to their overarching tables. These overarching tables are dataframe variable ***reportDF*** which contains the "*Precision*", "*Recall*", "*F1*", "*Accuracy*", "*ROC_AUC*" scores & ***resultsDF*** which contains the "*Baseline random split*", "*Baseline cross-validation*", "*Baseline repeated stratified CV*", "*Baseline shuffle split CV*", "*Random hyperparameter grid*", "*Tuned hyperparameter grid*", "*Tuned hyperparameter CV*" performance accuracy.

Classifiers' Accuracy & Performance

Once all the sub-programs have been executed, the results collected are analysed and visualised in this final part for each investigation. These include the HIV status investigation, the HIV status feature selection and the viral load detectability investigation.

The first measure of performance displayed in this section is a tabular dataframe of all the collected results from the different classification model accuracies. Here, the results are converted form decimal to percentages and then the dataframe is displayed. The program then groups the accuracy scores by *Classifier* to get the mean value of each classification mode and display those values.

These results are then visualised in a **classifier accuracy comparison barplot**. This was implemented by setting the x axis to the accuracy and the y axis to the classifier. This barplot is positioned in a horizontal orientation to improve interpretability as it is comparing percentages of accuracy. The error bars of this graph are set to include a confidence interval of 95%, which is the standard for bar chart statistical graphs. [42]

The results from the classification models were then used to create a set of **confusion matrix**. One confusion matrix was created from the results of each classifier using a *for loop* and seaborn *heatmap* function. In this cell 3 lists were defined. One to differentiate the classifiers, another to set the titles, and a final list to assign a sequential colour palette for each confusion matrix. The confusion matrix function looks at the *y_test* data (the actual class data) and compares it against the *y_predict* data that was calculated in each classifiers' sub-program. It then accesses the performance for each of the classifier results which can then be visualised in a heatmap.

The classification models thus-far have been set to use a random state of 1. This means that any results created would be replicable over the course of the project. However, this limits the data to only perform under one instance of the program whenever it is executed. For the final performance test of the investigation, the program uses a for loop to iterate over many **different random states** and takes the precision, recall, F1, accuracy and ROC AUC scores for each iteration and appends them to a consolidated dataframe.

These performance scores are then visualised in a tabular view and **barplot** graph to determine if the classification models were successful at proving the hypothesis. A 50% performance threshold was also added to the HIV investigation barplot to signify when the accuracy becomes a circumstantial possibility in an even sampled (50:50) predictive model. In data science, these 5 scores are exceedingly important and informative when evaluating the classification performance. Therefore, displaying them in this comparative view allows for efficient assessment of performance for each classifier.

After generating the results, the program finishes the investigation by performing a one-way analysis of variance (**ANOVA**). This ANOVA will test the significance of the obtained results by seeing if the performance values have the same population mean when applied to the ROC AUC score and accuracy of each classifier. This test will assess **any correlation and highlight the significance of means and interactions between the performance scores**. Implementing an ANOVA has benefits such as using an improved technique to analyse various factors in multidimensional data. The one-way ANOVA function was imported in the SciPy stats module, similarly to the t-test. However, this function requires at-least 3 parameters to be supplied for the analysis. Therefore, the program needed to separate the performance values by classifier and defined individual lists for each set of scores. The function *stats.f_oneway* then takes these lists as parameters and perform one-way ANOVA to return an f-value (Variance between Classifiers' Performance) and a p-value (Significance of Classifiers' Performance Variance).

Results and Evaluation

In this section of the report, I will be discussing the results identified from the project and evaluate how these solutions assisted in solving the problem. Some results showed promise to prove the project hypotheses, however there were no tangible solutions that reliably achieved the aim to predict HIV in a neuroimaging dataset. This part of the paper will go over the recorded results by evaluating their quantitative significances at solving the hypotheses. The project aims to find correlations, associations, and inferences within the neuroimaging data that supports the pre-existing theory that HIV positive individuals show signs of neuronal inflammation/death. In areas where the results contradict the existing concepts, a discussion about the reasons why the result do not satisfy the hypotheses is conducted.

Quantitative Result Methodology

This project predominantly took a quantitative approach to results by producing statistical numeric values regarding the analysis of the data and performance of the program. These arithmetic values are then transformed into diagrams to visualise, compare, and interpret the results identified by the program. This approach was preferred because this project uses python programming to employ analysis and algorithms which identify any concrete results. Using tables and graphs from the resulting data is also useful for the Stellenbosch University Research team, who will observe the results from an outside perspective (with insufficient knowledge about the python program).

HIV Status Investigation: Data Analysis

Descriptive Statistical Characteristics

The initial steps of data analysis identified the core statistical characteristics, which define the variables and attributes of the neuroimaging dataset supplied by Stellenbosch University. These initial statistics state the baseline characteristics of all **124 participants**, 62 of which are HIV negative and **62 HIV positive**. Demographic data did not describe many traits, but did reveal that the dataset consisted of entirely “black”/”coloured” [7] South African women between 18 & 50 (with a mean age of 30.27).

Other statistical qualities depicted in this stage of analysis include the mean ranges/quartiles and standard deviation of each feature. These details are useful regarding some of the characteristics in the dataset however to fully understand the distribution amongst the brain regions of the participants, a boxplot was created. In figure 9 we see the boxplot that **shows the normalised/standardised grey matter volume** as allocated between the separate MRI regions. Displaying this data using the seaborn *boxplot* function, we not only see the overall distribution of each MRI region, but it also depicts the median, inter-quartile range (lower/upper quartiles), the minimum & maximum correlating values as-well as any outliers present for the set of normalised data in each feature (MRI region).

The distribution of normalised MRI regions, as displayed in figure 9, was then shown to the research team at Stellenbosch University in South Africa. Here they could confirm that the MRI regions showed usual distribution for their respective features. Since the distribution of the participant neuroimaging data has been **verified by professional neurologists**, the next stage of analysis was permitted.

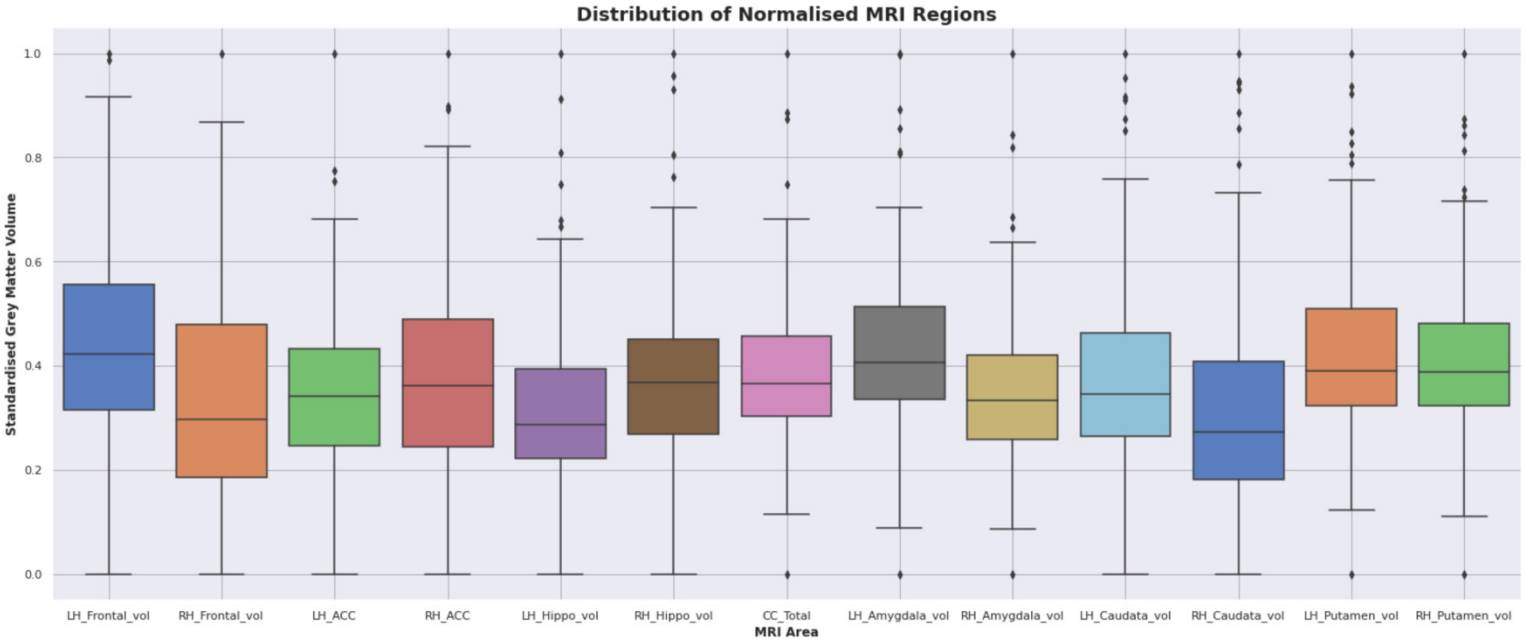

Figure 9: Distribution of Normalised MRI Regions

Independent T-Test

The main form of statistical analysis used in this stage of the program was an independent t-test. In this t-test we are attempting to test against the null hypothesis that: the normalised grey matter is the same between HIV positive and negative participants. By evaluating the mean values of the HIV negative and positive participants we can see how much variance there is per brain region. The independent t-test will give us 2 sets of results. A t-value, where the **magnitude of variation is measured**. The higher the t-value the more that brain region supports the alternate hypothesis. There is also the p-value, which gives the **probability that the t-value occurred by chance**. Therefore, we look for a p-value less than **0.05** which is **statistically significant** enough to say it was not produced coincidentally.

Figure 10 shows us the results of the initial t-test by displaying the t-values in a barplot per MRI region, which made analysis easier and enhanced interpretation. It is apparent from this graph that there are **5 MRI regions which show promise of solving the hypothesis**. These regions of interest are LH_Frontal, LH_ACC, CC_Total, LH_Putamen, RH_Putamen. These regions with the highest score test against the null hypothesis, so we can be sure to analyse them in more detail, in-order to see correlations and inferences when predicting the HIV disease in neuroimaging data.

Figure 10 shows the magnitude of each brain region against the null hypothesis but some of these values could have occurred by chance during the independent t-test. In-order to see which MRI regions truthfully have a chance of proving the alternate hypothesis I created another barplot.

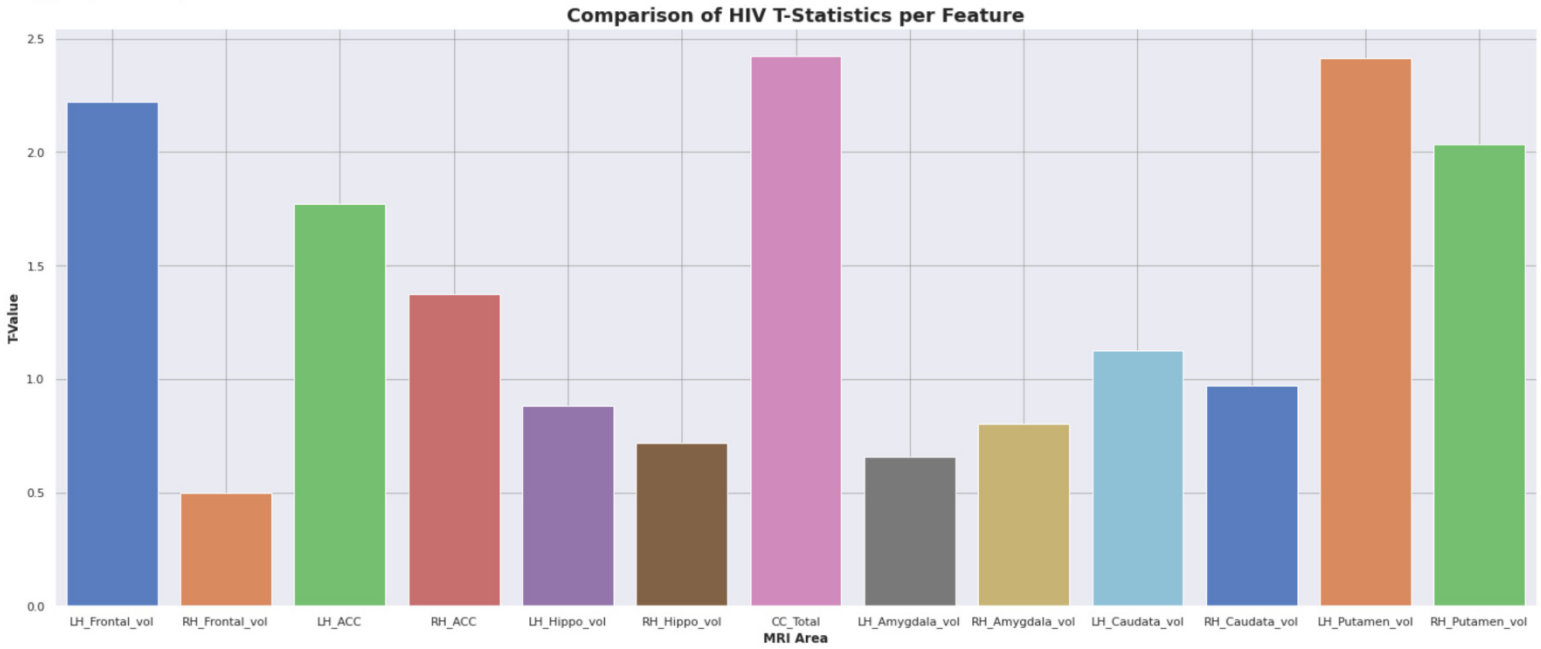

Figure 10: Comparison of HIV T-Statistics per Feature

Figure 11 compares the p-values of each feature analysed in the t-test. This barplot directly corresponds to the comparison of HIV t-values per feature diagram using the same style palette to represent each brain region. The p-values identified represent the likelihood that the t-value occurred by chance, which is very evident in some of the brain regions. In particular, the RH_Frontal, LH_Amygdala & RH_Hippo all have high chance, so those scores probably occurred by happenstance. In this barplot we are specifically look for the MRI regions that **have a p-value of less than 0.05** to signify their statistical significance. This probability threshold is shown using a red line at the x axis value of 0.05. Therefore, any MRI regions that fall below this line show an inconsequential enough probability of chance occurrence that we can scientifically assume they support their values in figure 10.

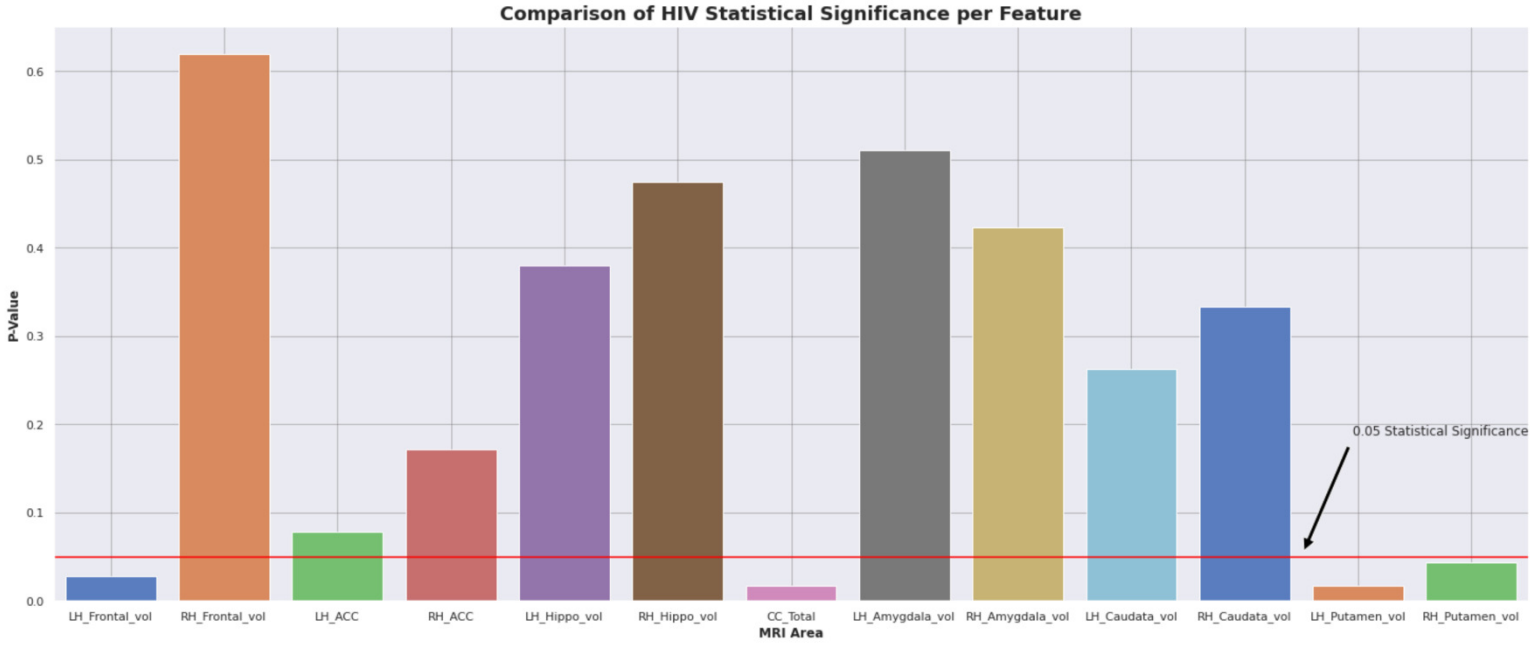

Figure 11: Comparison of HIV P-Values per Feature

Out of the 5 MRI regions identified in figure 10, we can see that **4 of these regions support the alternate hypothesis with statistical significance**. These are: LH_Frontal, CC_Total, LH_Putamen, RH_Putamen. However, looking at the probability of chance, we have had to discard the LH_ACC brain region as it has not shown to be statistically significant in this test. We can therefore move on with the 4 significant regions identified and investigate these brain regions further in future analysis.

Feature Selection Diagrams

Now that the significant brain regions have been identified, some rudimentary data analysis was carried out for these regions to look for clear relationships as-well as any disassociation between the sampled participant groups. In-order to perform this in-depth analysis, feature selection was used to isolate the 4 main MRI regions identified in the independent t-test.

The first form of feature selection data analysis used a scatterplot to see the general distribution and correlation of the 4 isolated MRI regions, separated by HIV status. Figure 12 depicts the results of this examination.

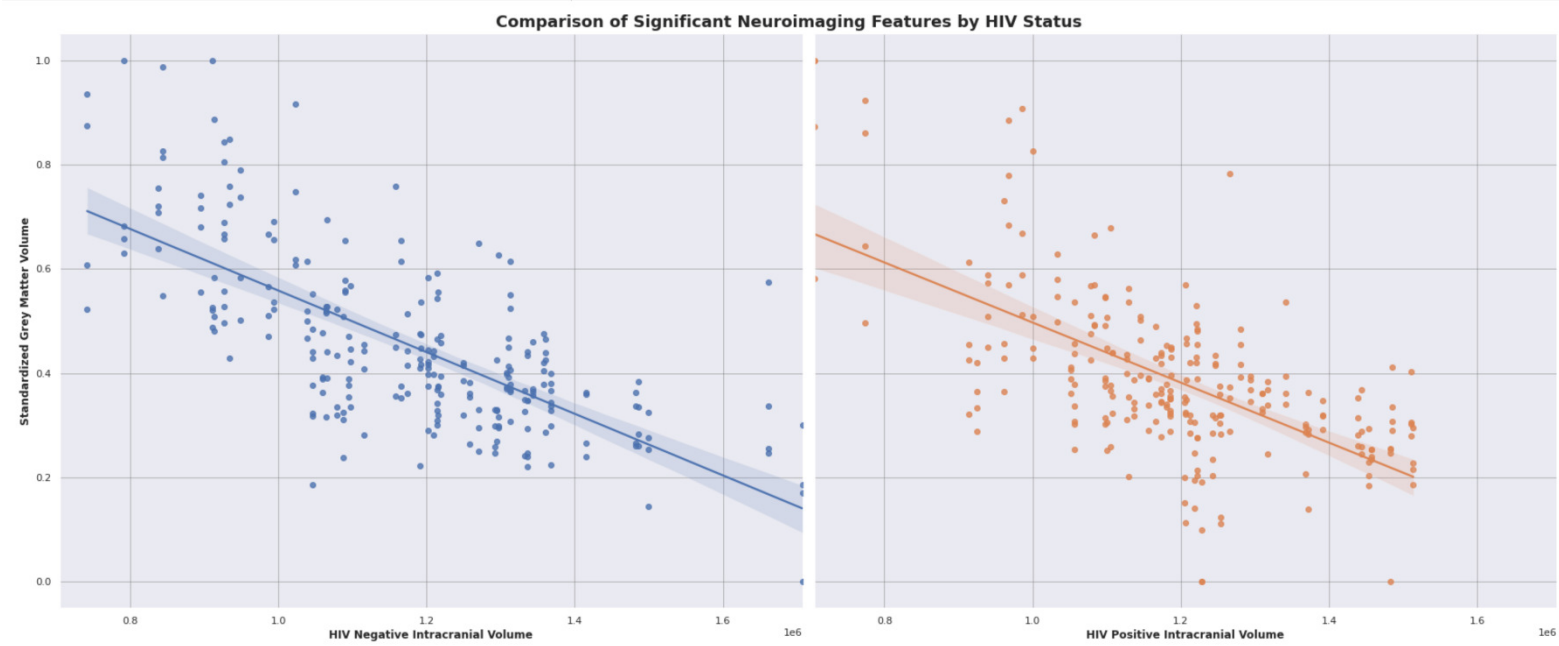

Figure 12: Comparison of Significant Neuroimaging Features by HIV Status

Figure 12 shows the general shape, correlation, and distribution of the MRI brain region grey matter against the intracranial volumes for both the HIV negative and positive participants. This columnized component analysis details the differences between the HIV groups showing several noteworthy elements in the selected data. Firstly, both have a similar negative correlation with a slightly steeper degree in the HIV negative samples. We also see clear limitation are present in the HIV positive MRI regions as there are no participants with a ICV higher than 1.6, whereas there are many outliers present in the HIV negative dataset which could have caused the MRI regions to support the alternative hypothesis in the independent t-test. These **visible limitations mean that there may not be enough inference in the dataset** to predict the HIV status for participants, even though we found some statistical significance for this project.

To finish the feature selection data analysis part of this project a pairplot was created to see the associations in the data per significant MRI region (separated by HIV Status). This assisted in demonstrating the divergence for said features. This pairplot is listed in the appendix as figure 23.

HIV Status Investigation: Supervised Machine Learning

Classification Accuracy

In table 2 we can see the individual classifier model accuracy results. Each method represents a model that was used to predict the HIV status of participants neuroimaging data. To observe the average accuracy for each classifier, the final column of table 2 includes this metric.

Classifier	Method	Accuracy %	Mean %
LDA	Baseline random split	58.06	58.96
	Baseline cross-validation	60.48	
	Baseline repeated stratified CV	59.68	
	Baseline shuffle split CV	57.89	
	Random hyperparameter grid	58.06	
	Tuned hyperparameter grid	58.06	
	Tuned hyperparameter CV	60.48	
SVM	Baseline random split	48.39	50.07
	Baseline cross-validation	45.16	
	Baseline repeated stratified CV	45.97	
	Baseline shuffle split CV	45.39	
	Random hyperparameter grid	48.39	
	Tuned hyperparameter grid	64.52	
	Tuned hyperparameter CV	52.69	
KNN	Baseline random split	48.39	50.43
	Baseline cross-validation	46.77	
	Baseline repeated stratified CV	54.57	
	Baseline shuffle split CV	51.97	
	Random hyperparameter grid	51.61	
	Tuned hyperparameter grid	45.16	
	Tuned hyperparameter CV	54.57	
Logistic Regression	Baseline random split	48.39	47.84
	Baseline cross-validation	45.97	
	Baseline repeated stratified CV	49.19	
	Baseline shuffle split CV	45.39	
	Random hyperparameter grid	48.39	
	Tuned hyperparameter grid	48.39	
	Tuned hyperparameter CV	49.19	
Random Forest	Baseline random split	54.84	56.80
	Baseline cross-validation	52.42	
	Baseline repeated stratified CV	53.49	
	Baseline shuffle split CV	57.24	
	Random hyperparameter grid	64.52	
	Tuned hyperparameter grid	64.52	
	Tuned hyperparameter CV	50.54	

Table 2: Accuracy of Classification Models per Classifier

The best classification model was **Random Forest** using a tuned hyperparameter grid which predicted the correct HIV status of **64.52**% of the test data. The least accurate method was a baseline **support vector machine** which had an accuracy of 45.16%. The over-all average accuracy for all classification models implemented in this stage of the project was **52.82**%. Due to the sample data being a 50% split of HIV positive and HIV negative participants, we can see there is a very small amount of inference. However, it is still an **insignificant conclusion** when attempting to classify the neuroimaging dataset. In-order to understand how the classifiers performed, we must investigate their overall performance and not just their accuracy. This was done through the use of a confusion matrix. Figure 14 in the subsequent section discusses this in more detail.

Figure 13 visualised the mean classification accuracy values with their respective error bars for a more distinct comparison and to strengthen interpretability of the results. The best classifier for average accuracy across the different classification models was **Linear Discriminant Analysis** with which falls within a small quantity of error (as depicted using the confidence interval). We can also see the worst functioning model was **Logistic Regression** with less than an average of 50% accuracy, which is the only classification model that goes against the core hypothesis of this project.

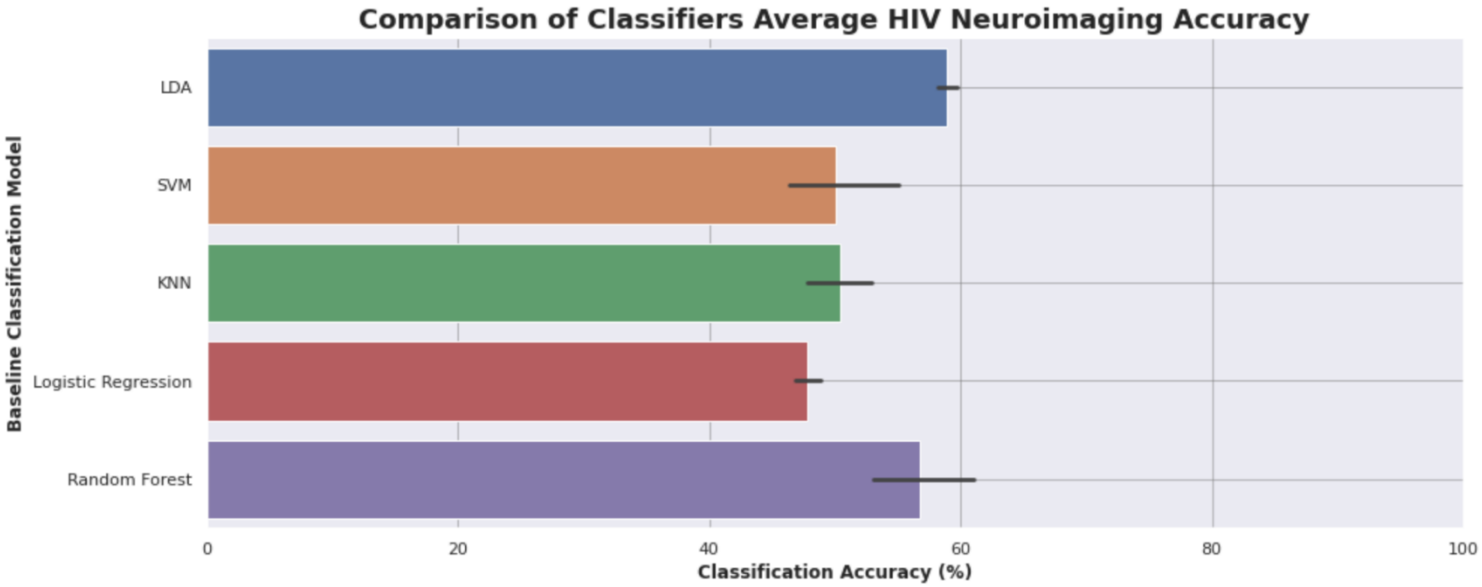

Figure 13: Comparison of Classifiers Average HIV Neuroimaging Accuracy

From the classification accuracy table (table 2), we can see that many classifiers' accuracy was **improved through cross-validation and hyperparameter tuning**. The tuned hyperparameter grid represents the exhaustive parameter grid search model and should be representative of the best accuracy for said classifier. However, this can facilitate overfitting in the model and therefore, a **hyperparameter tuned nested cross-validation model** is used with the best estimator parameters from the tuned grid model results. From the baseline split accuracy to the nested tuned cross-validation accuracy we see the following improvement:

- LDA: increase of **2.42**%
- SVM: increase of **4.30**%
- KNN: increase of **6.18**%
- Logistic Regression: increase of **0.80**%
- Random Forest: decrease of **4.30**%

The mean percentage accuracy increase is **1.88**% as a result of the tuned nested cross-validation models. This grid-search was used to find the optimal hyperparameters for each model which results in the most accurate predictions. Given a clear increase in classification accuracy as a result of the hyperparameter tuning, these tuned models will also be employed to classify the HIV status using different train/test data in a subsequent part of the program.

Overall, these results show an some suggestion that the HIV disease can be predicted using supervised machine learning as a vast majority of the classifiers had an accuracy better than 50%, which passes the accuracy threshold required as evidence in favour of proving the hypothesis that HIV status can be predicted with neuroimaging data. The accuracy of these supervised learning models does **indeed show some inference of HIV status from the neuroimaging data**. However, the

accuracy results are simply not substantial enough to clearly say that there is evidence of neuronal inflammation and death for HIV positive individuals. To prove this hypothesis, a significant correctness would be required where the majority of models have an accuracy result of at least 95% when predicting the HIV status of each participant.

Classification Performance

Classifier	Accuracy %
LDA	58.06
SVM	64.52
KNN	45.16
Logistic Regression	48.39
Random Forest	64.52

Table 3: Confusion Matrix Accuracy per Classifier

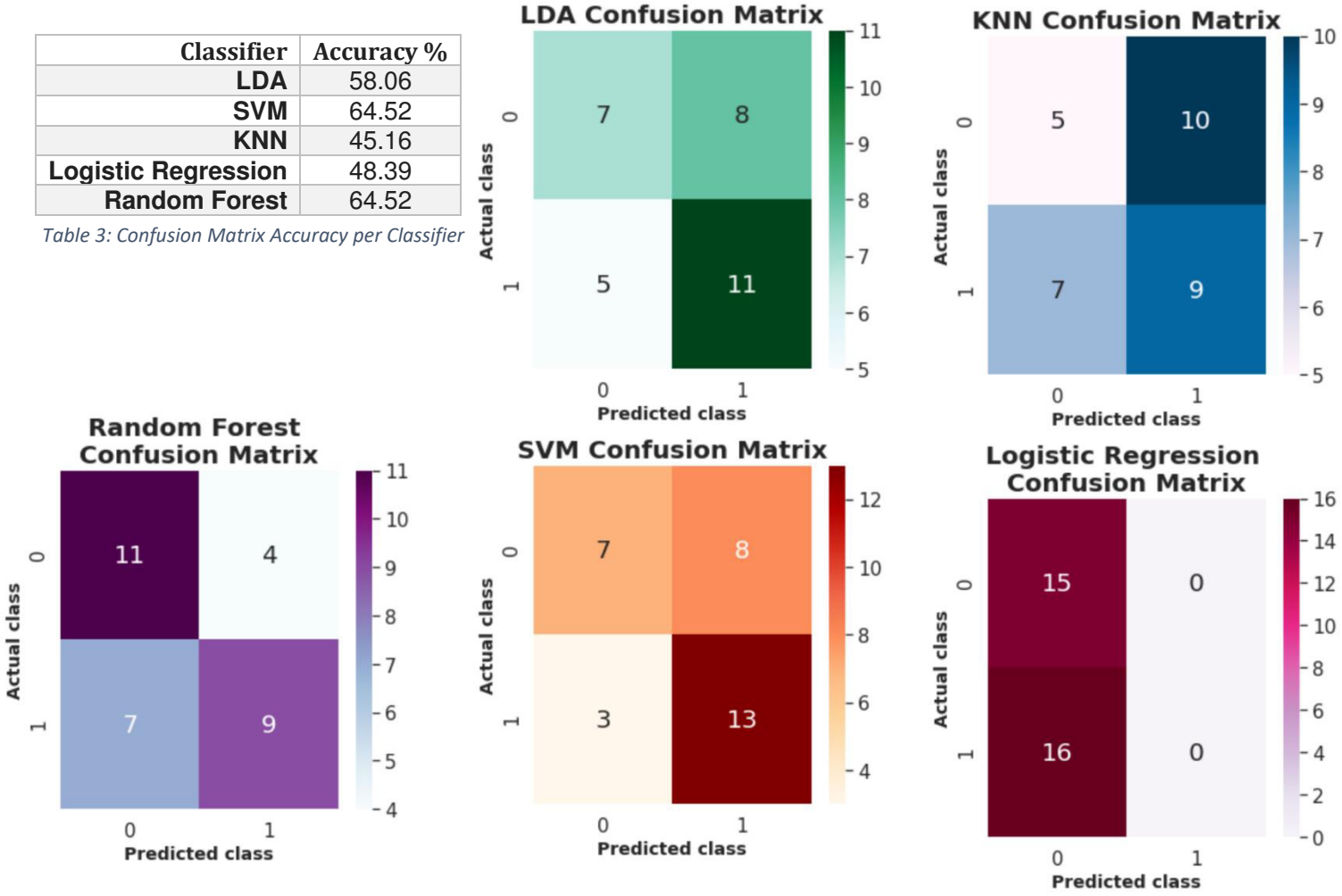

Figure 14: Confusion Matrix per Classifier

The training and testing data for each **confusion matrix** was the same, but the predicted data still differed between the classifiers. This meant that each confusion matrix had unique results and varying performance. The classifier with the best accuracy was both **SVM** and **Random Forest** which predicted **64.52**% (20/31 participant) of the test data's HIV status correctly. Using these confusion matrixes, we are also able to see which model performed the worst. Logistic Regression performed poorly as it predicted all the participants to be HIV negative even though more than 50% of the test samples were actually HIV positive. This demonstrated that **logistic regression has a negligible precision** score which raises concerns about the viability of the neuroimaging dataset. This is also corroborated with the fact that no individual classifier could predict the HIV status of the test data with more than 64.52% accuracy. The confusion matrix is a valuable performance metric because it doesn't just allow for the evaluation accuracy between the classifiers but it can also be used to calculate the **precision**, **recall** and **F1 score** which in-turn gives us the **ROC AUC score**.

Classifier	Metric	Score
KNN	Accuracy	51.71
	F1	48.59
	Precision	54.19
	ROC_AUC	51.96
	Recall	47.49
LDA	Accuracy	57.04
	F1	53.22
	Precision	55.27
	ROC_AUC	54.27
	Recall	53.08
Logistic Regression	Accuracy	45.18
	F1	23.58
	Precision	16.77
	ROC_AUC	50.00
	Recall	40.00
Random Forest	Accuracy	52.23
	F1	48.62
	Precision	54.21
	ROC_AUC	52.01
	Recall	46.46
SVM	Accuracy	52.04
	F1	47.80
	Precision	46.56
	ROC_AUC	48.20
	Recall	51.65

Table 4: Average Performance Score per Classifier

The classification models' results were appended to a list and the models then **assessed under 10 different random states** to produce a list of impartial results which maintained integrity. The performance scores are attained from the hyperparameter tuned nested cross-validation models already created when they are applied to the 10 different train/test splits. This includes the ROC AUC score where an average result was calculated from the different train/test splits. Table 4 depicts the mean score of all the results identified from the integrated train/test splits per classifier.

Figure 15 also reads these values and displays them in a comparative format using a barplot to visualise the scores, making it easier to understand which performances scored best across the different classifiers. From Table 4, we can identify that the most accurate classifier was **Linear Discriminant Analysis** with a prediction rate that was correct **57.04**% of the time across the different testing iterations. This still suggests that there is a small margin of inference that can be detected using the neuroimaging dataset, however there isn't a strong enough presence of neuronal inflammation/death in these participants to be able to accurately predict the HIV status based on their neuroimaging data. The average accuracy across the different classifiers stood at the value **53.22**%. Although above the 50% threshold needed to see any inference, this **result is still an inconsequential accuracy score**.

The **ROC AUC** score represents how capable the classification model is at **distinguishing between the HIV positive and negative participants**. A high score means the model is effective at separating the classes and predicting the right outcome. With the HIV sample size being a 50:50 split, a ROC AUC score of **50 is the worst-case performance** as it shows that the model does not actually distinguish a difference between the neuroimaging data for each class. [43] In table 4 it is obvious that there are no models that have learned enough from their training data to be able to

significantly predict the test data. The **highest ROC AUC score is only 54.27** (+**4.27** above an undistinguishing model). This is also representative of the other classifiers ROC AUC scores which all fall around the 50 mark, and therefore show that no inferences could be feasibly predicted using this neuroimaging data. A remarkable result identified using the ROC AUC score is from support vector machine classification model. This is because it has a score of **48.20, meaning that it predicted more HIV negative people as positive** and vice-versa. A score below 50 means that the model is able to differentiate between the classes but predicted them incorrectly (showing promise but also having a low accuracy). That is why we cannot use the accuracy or the ROC AUC score alone as the primary metric in Machine Learning. The mean ROC AUC for all classifiers is **51.29** (+**1.29**) which still is not enough to show potential to prove the hypothesis.

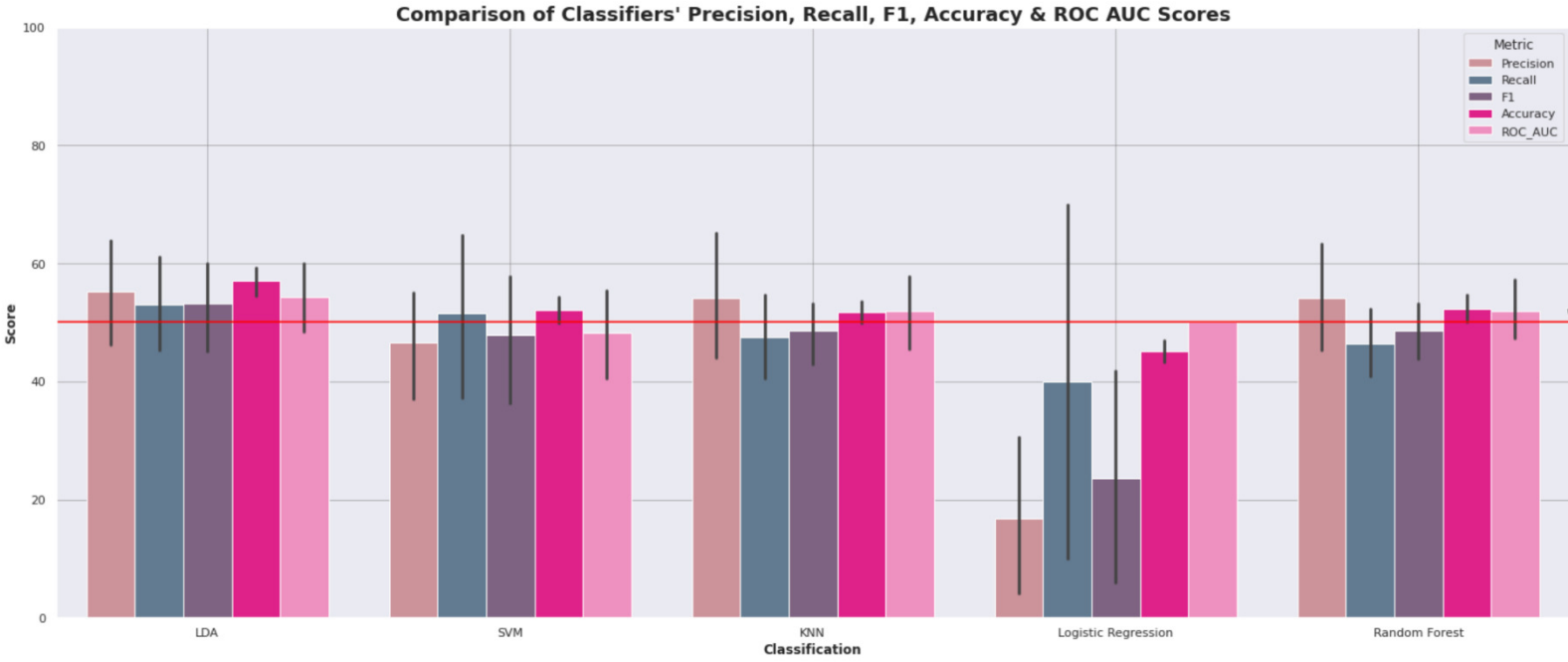

Figure 15:Comparison of Classifiers' Performance Scores

In figure 15 we can see a clear comparison between the classification model performance for each classifier, including the average precision, recall, F1, accuracy & ROC AUC scores. The red line depicted in the diagram represents the **50% performance threshold**. The higher the score is above this threshold, the better the classifier is at solving the hypothesis. Although a low ROC AUC score is more preferable than one near the 50% threshold. From this figure, it is clear that the best performing classifier was **Linear Discriminant Analysis** where all the performance scores are above the threshold and higher than the corresponding results from the other classifiers. The margin of error at 95% confidence interval still shows the **most reliability for LDA** when compared with a model of similar performance (Random Forest). We can also deduce that the worst classification model was Logistic Regression, where none of the performance metrics show any results that helped prove the hypothesis. It also had a great deal of error within its predicted values leading to the tremendous error bar seen for the recall metric in the graph. Overall, **these classification models did not show any significance** for the hypothesis.

To quantifiably evaluate the resultant performance scores, a statistical analysis was carried out using **one-way analysis of variance** (**ANOVA**). This will show how reliable the results are for each classification model in comparison to each other's performance scores. The classification model results (for both the ROC AUC and Accuracy scores) were supplied to the one-way ANOVA and statistically analysed for variance and probability of occurrence. ROC AUC ANOVA results:

- Variance between Classifiers ROC AUC Score (ANOVA F-Value)= **0.6006**
- Significance of Classifiers ROC AUC Score Variance (ANOVA P-Value)= **0.6641**

The low f-value suggests that we can assume that there is no significant variation between the ROC AUC scores per classifier and that the mean score depicted in the above diagrams are representative of the average ROC AUC score. Though because the **p-value is above 0.05**, this f-value could have easily occurred by chance (especially considering that the value is at a considerably high 0.6). Therefore, **we must not assume that the above mean is reflective of the average ROC AUC score** for the classifiers.

A one-way ANOVA was also carried out on the accuracy results of each classifier to determine their variance and significance:

- Variance between Classifiers Accuracy (ANOVA F-Value)= **14.12**
- Significance of Classifiers Accuracy Variance (ANOVA P-Value)= **1.527e-07 (0.0000001527)**

The ANOVA performed for the accuracy results had a high f-value and a low p-value (< 0.05). Therefore, the results obtained proves that **there is high variance between the classifiers' accuracy** results and that this variance is significant enough to be considered reflective of the scores produced for this dataset under most circumstances. Thus, making the **accuracy results inconsequential as a performance metric** in this project.

This predictive model was inaccurate because it was not able to prove the hypothesis using the given dataset from Stellenbosch University. **Limitation such as demographic characteristics** could play a big role in why this data is not able to adequately train these classification models. The given dataset does not account for all the relevant brain regions as there are **no features that represent the Optical-lobe**, which is known to be the region whose grey matter is most effected by HIV. [5]

HIV Status Investigation: MRI Feature Selection

After determining that supervised machine learning was not able to predict the HIV status of participants dependably, **feature selection was used in an attempt to boost the performance** of the classification models. Therefore, the 4 MRI brain regions that showed statistical significance in the independent t-test were used as the features for the classification models rather than using all the brain regions as was applied in the aforementioned machine learning sub-programs.

Classification Accuracy

Classifier	Initial Accuracy	Feature Selection Accuracy	Accuracy Change
LDA	58.96	58.88	-0.08
SVM	50.07	53.02	2.95
KNN	50.43	50.60	0.17
Logistic Regression	47.84	59.04	11.20
Random Forest	56.80	53.32	-3.48

Table 5: Classification Accuracy of Feature Selection Comparison

Table 5 gives us a comparative view of how the accuracy of the models improved/deteriorated since using feature selection. The most accurate model used with feature selection became **Logistic Regression** which had a substantial **increase of 11.2% accuracy**. The net increase in classification accuracy across all models was **2.15**%. In the initial classification models, we had an average accuracy of 52.82% overall. This increased to an overall average accuracy of **54.97**% after using feature selection. For more detail on each model's accuracy per classifier, refer to table 8 in the appendices.

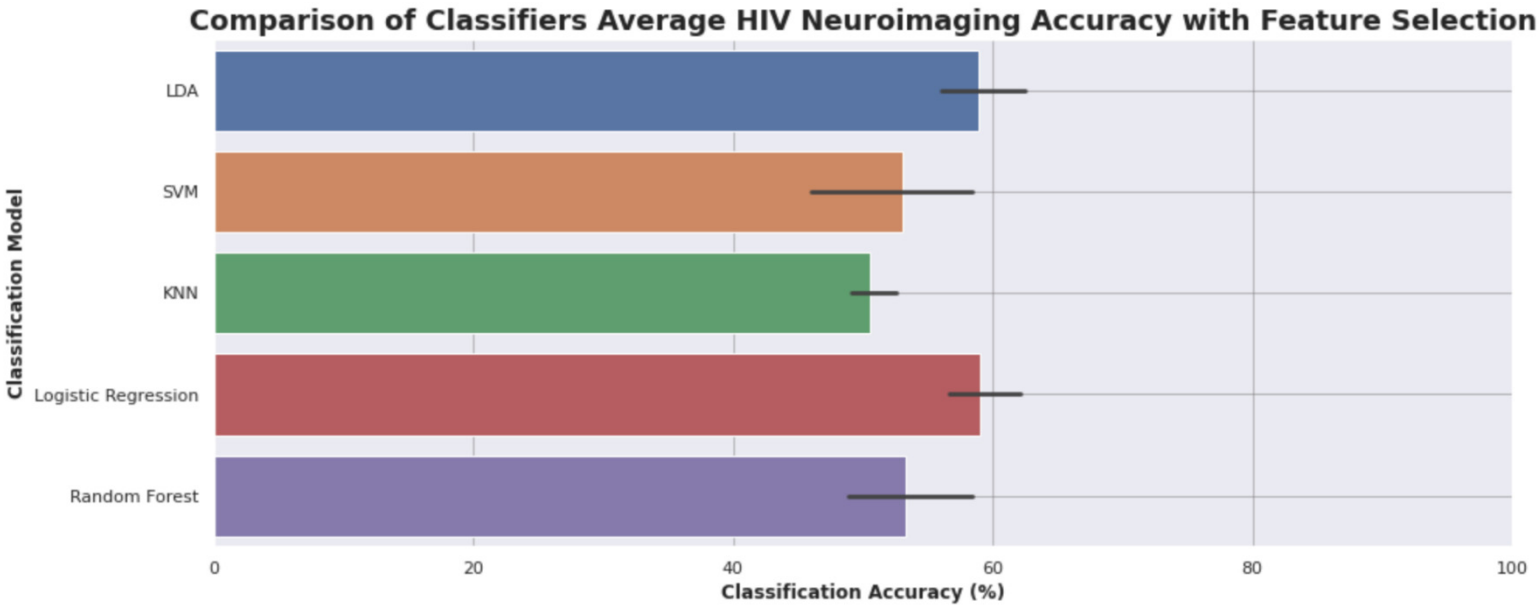

Figure 16: Comparison of Baseline Classifiers' HIV Neuroimaging Accuracy with Feature Selection

By comparing figure 16 with the initial results in figure 13, we can clearly see the drastic **increase in Logistical Regression classification accuracy** and the **decrease in Random Forest accuracy**. This visualisation also depicts the 95% confidence intervals between the different classification models. This shows that there is a general increase in doubt for the feature selection accuracy results, as the error bars are larger and more varied per classifier.

Classifier Performance

Feature selection has shown an overall improvement in accuracy, however in-order to determine if there is an improvement in performance, performance metrics were implemented to evaluate the classifier results. A confusion matrix for each feature selection classification model can be found in the appendices, figure 24. From these matrices, we can see that **LDA** and **SVM** had the **highest accuracy at 67.74**% (21/31 participants) using the hyperparameter tuned model. This means that those models **predicted 1 more participant correctly** than the initial models. However, the average accuracy across all models fell by **1.24%** (**from 54.27% to 53.03%**) when using feature selection.

Classifier	Metric	Feature Selection Performance
KNN	Accuracy	53.19 (+1.48)
	F1	50.7 (+2.11)
	ROC_AUC	54.95 (+2.99)
LDA	Accuracy	56.19 (-0.85)
	F1	61.19 (+7.97)
	ROC_AUC	59.87 (+5.60)
Logistic Regression	Accuracy	54.04 (+8.86)
	F1	51.03 (+27.45)
	ROC_AUC	57.89 (+7.89)
Random Forest	Accuracy	46.44 (-5.79)
	F1	49.69 (+1.07)
	ROC_AUC	50.81 (-1.20)
SVM	Accuracy	55.27 (+3.23)
	F1	56.54 (+8.74)
	ROC_AUC	58.08 (+9.88)

Table 6: Feature Selection Performance Score Changes per Classifier

Table 6 show the **increase/decrease in accuracy, F1 and ROC AUC scores** as a result of feature selection. **Logistic Regression** had the biggest change in accuracy, precision & recall (and therefore highest change in F1 score) whilst **SVM** had the largest increase in ROC AUC score. However, it is **LDA** which became the best model for distinguishing between the HIV positive and negative individuals with a **ROC AUC score of 59.87** (+**9.87**). Which is noticeably higher than the best score of 54.27 in the initial LDA model performance. The models had an average ROC AUC score of **56.32** which was **5.03** (6.32–1.29) **higher than the original models** because of feature selection. This means that the classifiers were able to differentiate the HIV status more precisely than the initial Classification models.

The values in table 6 corroborate with the results shown in figure 17, which is a revised diagram of the “Comparison of Classifiers' Performance Scores” from figure 15. These **performance metrics are evidently more integral per classifier** than the previous results. We can conclude this because the results are noticeably higher in relation to the 50% performance threshold outlined in both figures. From figure 17 we still see that **Linear Discriminant Analysis** has the best performance of any classification model with **SVM** and **Logistic Regression** following up with a better performance than the initial results.

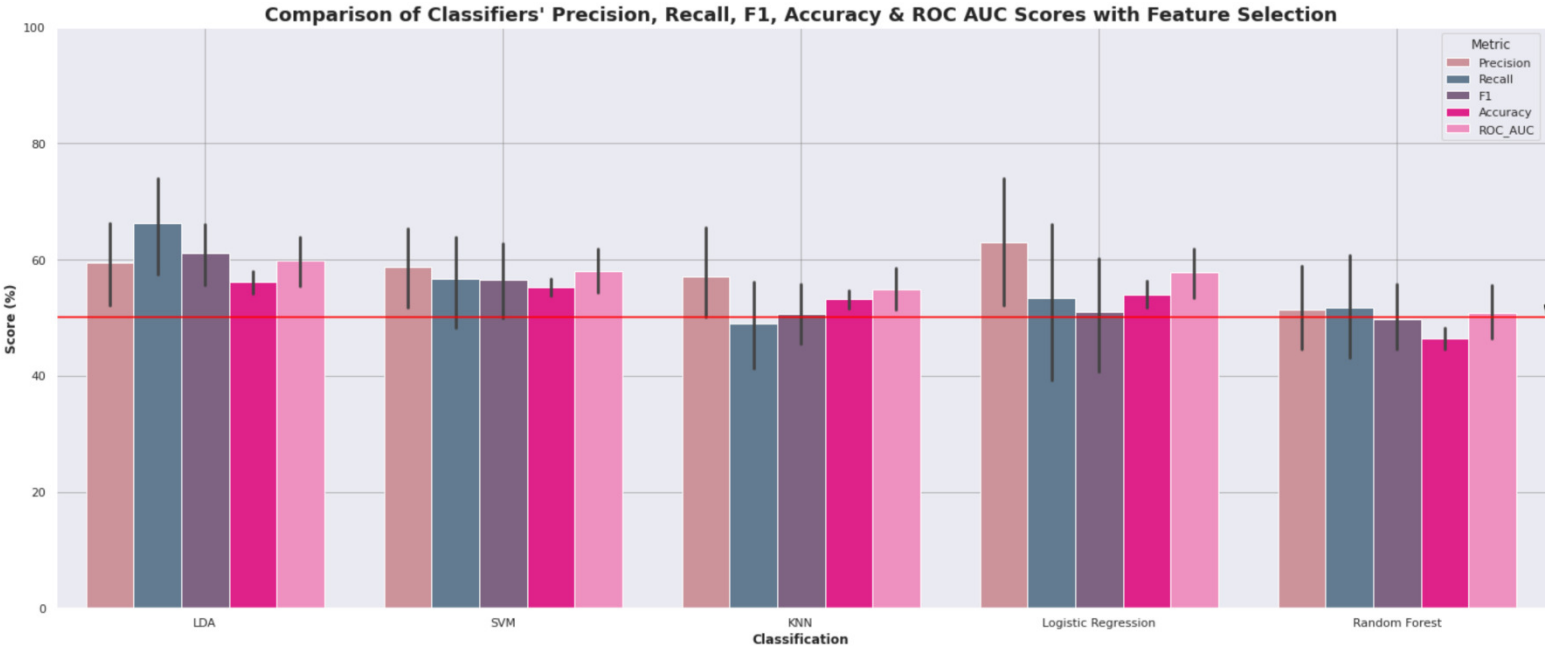

Figure 17: Comparison of Classifiers' Performance Scores for Feature Selection

From the previous results, we found that there was little variation between the ROC AUC scores with an insignificant probability of occurrence using a one-way ANOVA. The results from feature selection when applying statistical analysis:

- Variance between Feature Selection Classifiers' ROC AUC Score (ANOVA F-Value)= **2.4854**
- Significance of Feature Selection Classifiers' ROC AUC Score Variance (ANOVA P-Value)= **0.05528**

There is **some variation present** between these ROC AUC scores; however, it is not exceedingly prevalent because the f-value is high but not substantial. Nevertheless, in this ANOVA we **almost have statistical significance** as the p-value is just over the 0.05 threshold. This is still enough, to prove the significance of the given variation score, or a score close to it, to be specific enough. In conclusion, the feature selection ROC AUC scores likely have only **trivial variation** between each other and **likely reflect the outcomes of current and future analysis** with the classification models. This shows more potential than the initial results, but it displays more variation present in the performance score than originally expected.

When analysing the accuracy performance for the classification models, the one-way ANOVA shows that there is exceptionally high variation that also has a substantial chance of probability. Therefore, the results identified during the accuracy testing for the classifiers is likely to have significant variance when running the model. These results **do not help solve the hypothesis** of this investigation due to the **large probability of variation**.

- Variance between Feature Selection Classifiers Accuracy (ANOVA F-Value)= **14.951**
- Significance of Feature Selection Classifiers Accuracy Variance (ANOVA P-Value)= **4.202e-08**

In conclusion, **feature selection has improved the classification model's performance** in most aspects in comparison to the initial classification performances. Though, there is still not enough reliability for the investigated classifiers when predicting the HIV status using the neuroimaging dataset supplied by Stellenbosch University. Limitations in this dataset cause the feature selection **classification models to perform insufficiently**. This could be due to factors such as the limited feature data. Currently **the dataset only focuses on grey matter volume** per MRI region, whereas it has been proven that the **HIV disease most impactfully affects the white matter** of the brain as-well as other features that could have been present in the participant dataset.

Viral Load & ART Investigation: Data Analysis

The next step in this project was to look into the **Viral Load** and **antiretroviral treatment** (**ART**) data to see if data analysis and supervised machine learning could be used to investigate and predict these classes for the participants, rather than just the HIV status.

Preliminary Results

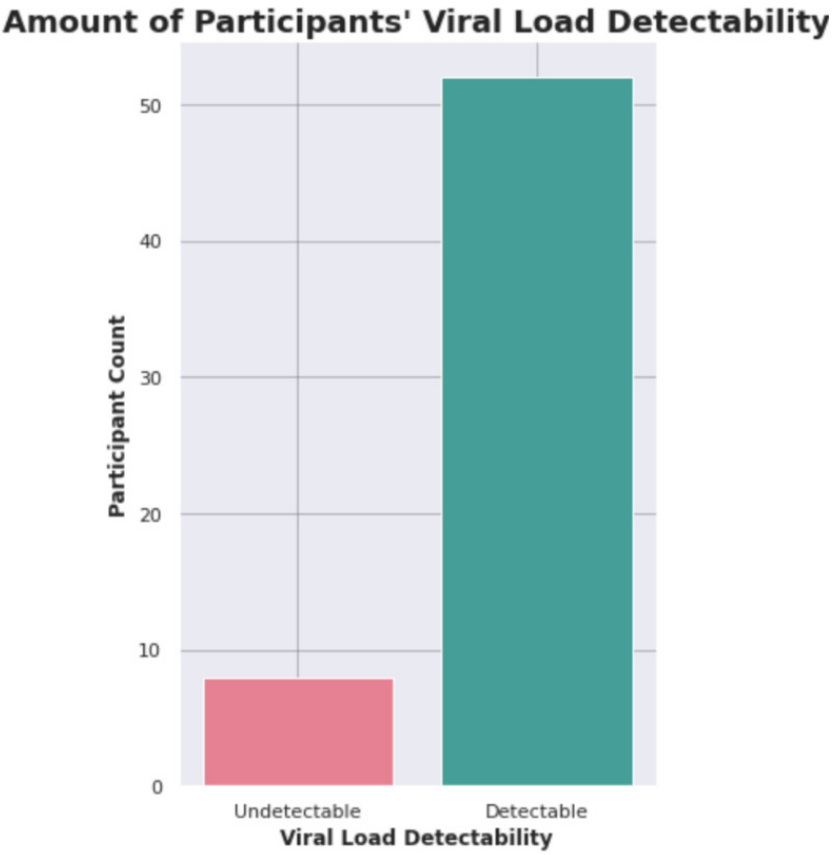

Figure 18: Quantity of Viral Load Participants

Before carrying out any statistical analysis, we want to **count the sample size of participants** who apply to the hypotheses. From this result we know if there is reason enough for an investigation into those classes. Below are figures 18, 19 & 20 which depict the sample sizes for the 3 areas we want to investigate in this part of the project.

The project supervisors from Stellenbosch University asked for the prediction of Viral Load and ART to be a potential deliverable in this project, as they were features present in the dataset. So, I analysed both the initial data and the follow-up data to determine if these factors were worth investigating. In figure 18, we see that there is a **substantial difference in the count** of detectable and undetectable viral load participants. However, there is still a total of 8 participants who have an undetectable viral load that could potentially be investigated further. Using statistical analysis to determine if there are any areas of significance and a stratified train/test split in machine learning, we may produce an accurate predictive model. Thus, we will investigate the viral load detectability in more detail to produce some results.

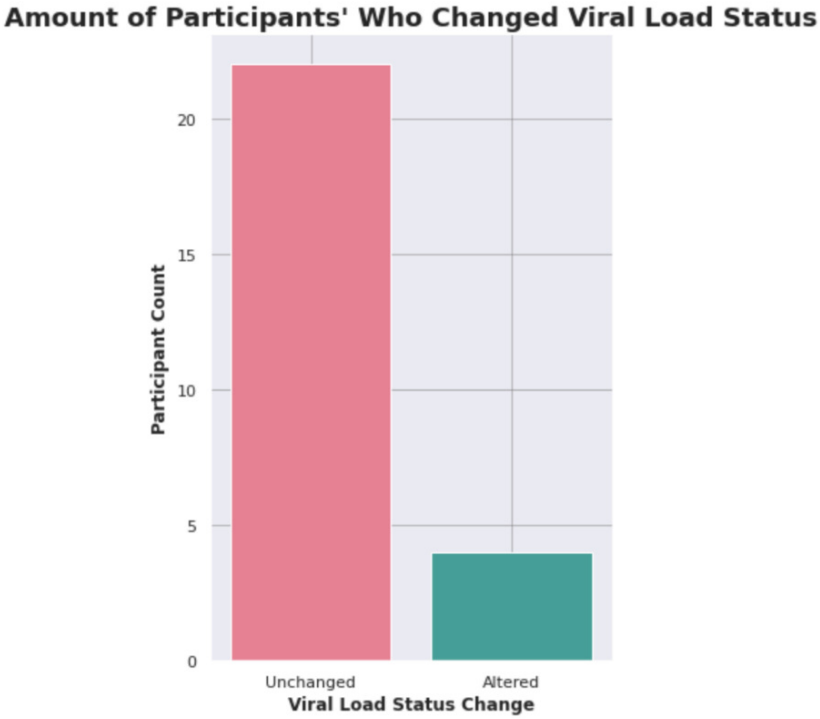

Figure 19: Quantity of Participants which Changed Viral Load Status

Looking at figure 19 the majority of HIV positive participants had a viral load that did not change between the initial and follow-up examinations. Here, only 4 participants changed their viral load status which is simply **too small of a sample size** to investigate. Because of this data limitation (and after identifying how limiting the data already is from the initial investigation) we decided to suspend the investigation into changing viral load status and its neuroanatomical implications. Looking at figure 20, there is even more of a disproportionate sample size which simply **cannot be investigated** until more neuroimaging data with changing ART status is produced.

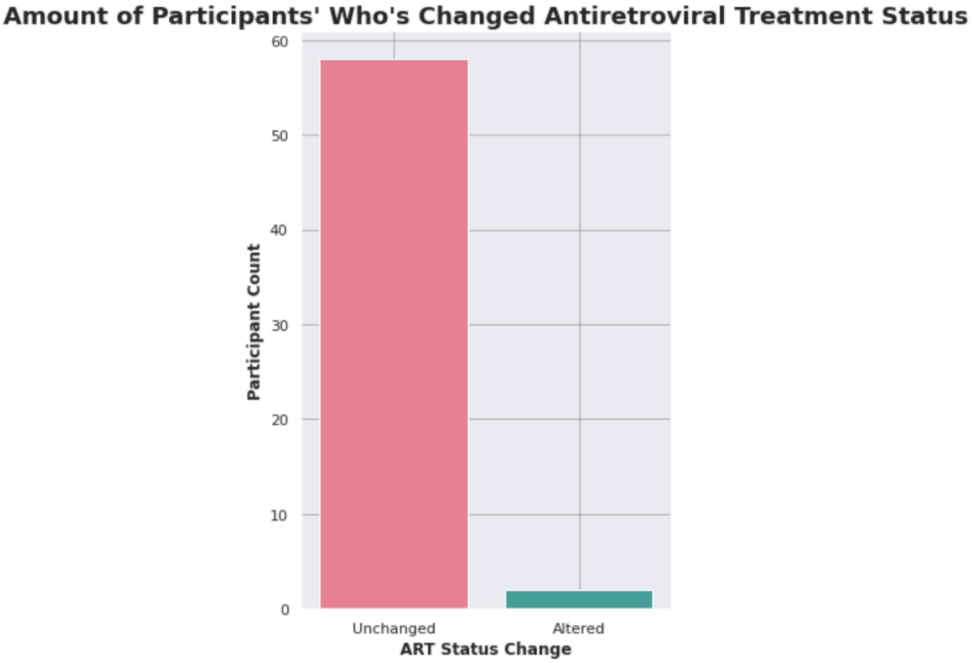

Figure 20: Quantity of Participants which Changed ART Status

Independent T-Test

An independent t-test was carried out on the MRI regions of HIV positive participants to see if there's any significant brain regions that contradict the null hypothesis: normalised grey matter is the same between detectable viral load participants and non-detectable viral load participants. The results of this t-test can be found in the appendices (figure 25 for the subsequent t-values barplot and figure 26 for the resultant p-values plot). Looking at these results we see the two MRI regions LH_Caudata and RH_Caudata have the most statistical significance of all the brain regions. Nevertheless, the probability that they occurred by chance is too high and not statistically significant enough to warrant the use of further feature selection analysis/classification.

Viral Load Investigation: Supervised Machine Learning

After running the classification models for the different classifiers, the results seem astonishingly high considering no statistical significance was identified. This originally came as a surprise when obtaining the results. But on further inspection, we can see that these results stemmed directly from the sample size of the undetectable participants. The sample size equated to **0.8667 (52/60)** and therefore the **accuracy threshold for this investigation is 87%** rather than the 50% split used in the HIV status investigation. The results shown in table 7 and figure 21 show that even with this high accuracy and a reflective train/test split for the supervised models, none of the classifiers showed potential. The average accuracy across all the classification models was therefore only **85.72**% due to the limitations in sample size.

Classifier	Accuracy
KNN	86.67
LDA	80.95
Logistic Regression	86.99
Random Forest	86.99
SVM	86.99

Table 7: Viral Load Classification Accuracy

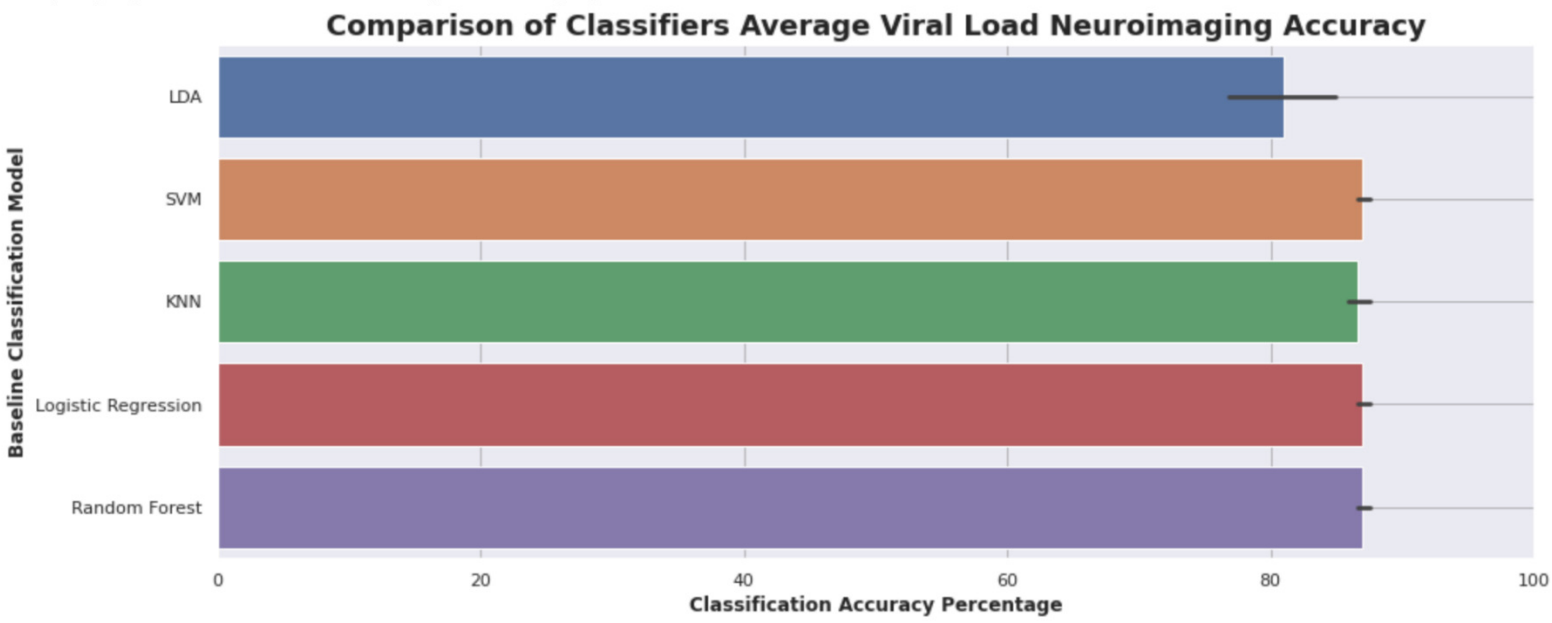

Figure 21: Comparison of Viral Load Performance Scores per Classifier

When looking at the confusion matrix (figure 27 in appendices), we see where the classification models are failing. The models predominantly learn how to classify detectable viral load neuroimaging data whilst only learning the data attributes of 8 undetectable participants. When the model predicted the viral load status in the test data, it classified them all as undetectable giving the model, on average, an 87% accuracy from just the distribution of the sample data. The models should therefore get a ROC AUC score near 50 as the **classifiers did not differentiate the data** at all.

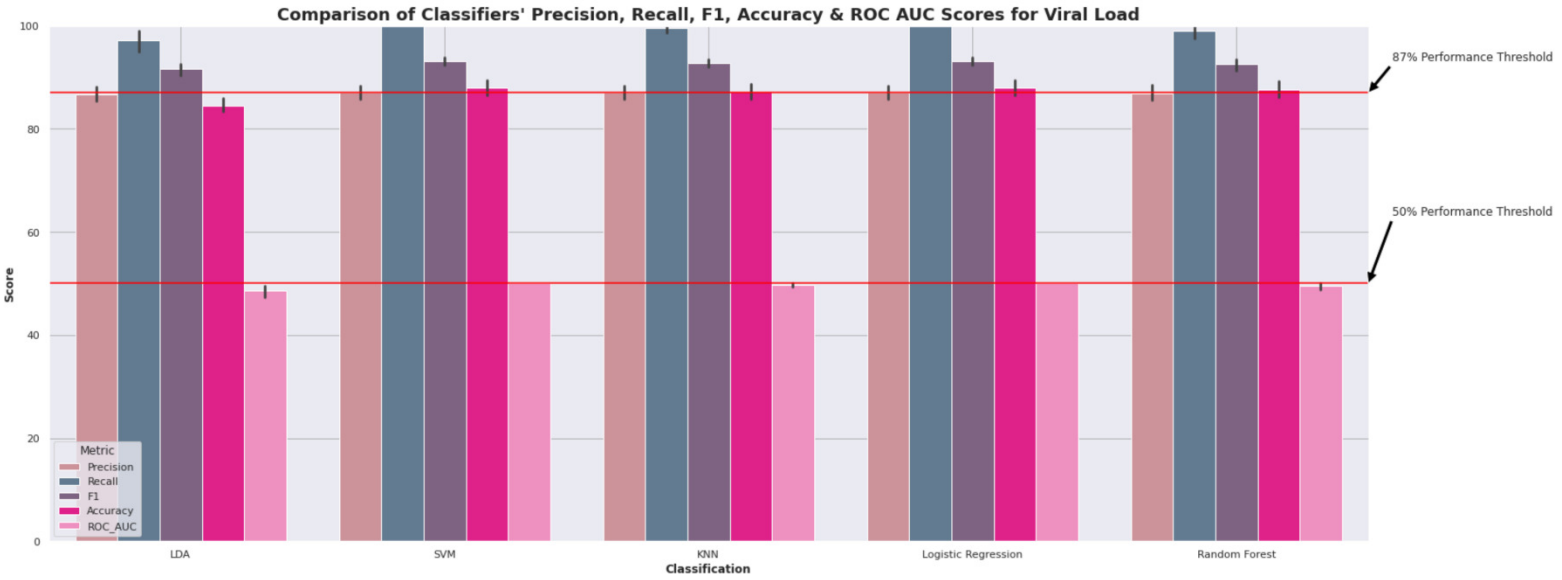

Figure 22: Comparison of Viral Load Classifiers' Accuracy

Figure 22 visualises the invalid results remarkably well. The results may look noteworthy however, the ROC AUC score for each classifier reveals its true performance (where a score closer to 50 cannot disseminate between the predictive classes). This is **because the disparity of samples is too great** to create accurate results from the models' performances. As a result of the undetectable viral load class only having 8 participants, all **results from this investigation are proved to be insignificant**.

To prove that these results are truly invalid an ANOVA was conducted in the same methodology as previously implemented. The results for both ROC AUC and Accuracy performance show that there is only slight variation between the scores. The p-value suggests that this small variation is statistically significant (<0.05). Because there is this little variation across the model's performance scores, then the **results are likely to be the standard** we can expect for these models when applying the viral load neuroimaging dataset.

- Variance between Viral Load Classifiers' **ROC AUC** Score (ANOVA F-Value)= **3.417**
- Significance of Viral Load Classifiers' **ROC AUC** Score Variance (ANOVA P-Value)= **0.01273**
- Variance between Viral Load Classifiers' **Accuracy** (ANOVA F-Value)= **3.533**
- Significance of Viral Load Classifiers' **Accuracy** Variance (ANOVA P-Value)= **0.01072**

This result identified and confirmed that there are limitations from the dataset which invalidate the classification accuracy due to the **limited sample size and data** present for these investigations. More participant data that highlights undetectable participants and participants whose HIV condition changed during the follow-up examination, is required in this project. Then a detailed discussion of evidence and results would be able to appropriately prove/disprove the core hypotheses.

Future Work

The core objectives of this project were fulfilled to completion, but the desired outcome was not achieved because of several major limitations identified in the project data. Because of this, the project should be continued with new data that helps prove or disprove the hypothesis to see if machine learning can be used to predict HIV status in neuroimaging data. Even with the current results identified, there are still many ways that this project could be expanded. These include:

1. Obtaining more data and increasing the sample size.
2. Optimising the current program by adding new functionality.
3. Collaborate with Stellenbosch University research team regarding their next publication.
4. Explore the deliverables dropped from the initial scope in more depth.

Applying a new dataset should strengthen support in favour of the theory that neuronal death/inflammation is present for HIV participants. The program created in this project could also be applied to this new dataset reasonably easily too. Especially if the CSV file(s) contain similar neuroimaging values obtained using the same MRI methodology. The new dataset would be sufficient if it contains several of the following factors that were identified as limitations in the current data. Including:

- Larger dataset of participants to limit the need for data stratification.
- Wider demographics to explore, not just South African women.
- More time relevant data, like disease stage and participants who changed viral load or ART status for a follow-up in a much larger sample.
- More neuroimaging data, white matter accounts for most neuronal death nor just the grey matter that was examined in this project.

The current program explores the classification side of supervised learning in great detail but does not examine any regression models that could be better applied to the dataset. The classification model was used because of the dataset's format, having a binary class of HIV positive and negative participants and a list of features detailing each of the participants MRI brain regions. Implementing regression models were not used in scope to help predict HIV status from the neuroimaging data. In reflection and given more time, regression models would be particularly insightful for this project. The models would be useful for predicting the CD4 count and Viral Load concentration for the HIV positive participants.

Further Research with Stellenbosch University can also be performed as future work inf this project, so that a fully informed publication is created to help tackle the challenge of HIV in neurology.

Over the course of this project several of the deliverables changed to best fit the data and approach desired. One of these deliverables was to produce and develop a neural network predictive model. Which after the initial analysis was deemed as unnecessary for the customers and too costly for the project scope (as it would take excessive time to implement). This project did begin to explore deep learning as a potentially useful tool when creating a predictive learning model, however the initial results showed the same limitations as the classification and statistical analysis. Therefore, it was not implemented into the program, and therefore not mentioned in this report as an implemented model. To corroborate the results in this project, other libraries and programming languages could be employed with the same methodology applied in those instances. Several other processes that could be explored include: Keras, TensorFlow, PyTorch, and OpenCV as they are reliable models that are often used in supervised machine learning and deep learning programs.

Conclusions

In the conclusion of this project, an appraisal of its success is evaluated against which aims were satisfied. The core aim of this project was to use data analysis and machine learning to predict HIV in a South African Dataset. Using statistical analysis and data science methodologies, a program with data analysis, supervised machine learning and feature selection was implemented. However, this program failed to create an accurate predictive model that would show clear inference for HIV in neurology. This project did prove that there are statistically significant brain regions when looking at the mean values for the HIV positive and negative participants in the South African data, because of the independent t-test. When performing supervised machine learning, the program was not able to actively predict HIV status of participants when applying proven data science methodologies. However, this project was able to deduce that these results were because of the limitations present in the obtained neuroimaging data. Other research projects have scientifically proven that there are discrepancies between HIV positive and negative individuals, therefore we can look into the restrictions of the given dataset. Limitations identified in the dataset applied to this project included:

- Small sample size, especially with follow-up participant data.
- Inadequate demographics, only focussing on one main ethnic group of South African women.
- Solitary focus on grey matter volume in the 13 MRI regions and not taking other neuroanatomy features into account (e.g. Optical-lobe & white matter distribution).

This project's initial aim was to "Further my technical knowledge of data science regarding data analysis and predictive machine learning". This was achieved through the learning and implementation of data science methodologies by fulfilling the other principal aims. I also aimed to "Analyse and visualise the neuroimaging dataset to identify key areas of the brain that signify HIV infection". Here, I found the descriptive characteristics and performed statistical analysis on the dataset to identify the statistical values of HIV positive participant. The results were then visualised using python programming with the Seaborn library functions.

Implementing the supervised learning models and then developing these model using hyperparameter tuning and nested cross validation accomplished the next two aims, to "Create classical machine learning models and evaluate the model's peak classification performance to determine the best model to utilize" & "Develop the predictive model to closely predict if a participant has HIV based on neuroimaging data". Although an effective predictive model was not able to be developed this project discovered the limitations present in the HIV dataset as-well as the difficult interpretability of the HIV disease within neurology. These results have been recorded and will be published for other research projects to reference and then distributed to the Stellenbosch research team which satisfies the aim to "Identify any tangible results that could contribute to the Stellenbosch University research project".

Completing this project has expanded personal knowledge regarding data science practices and allowed for independent contemplation and documentation of the overall project. Logging these processes and implementing the knowledge learned has help me accomplish my last aim to "Reflect, evaluate and document the one semester individual project" before it is submitted for examination. Whether or not the project was effective in the end is dwarfed by how much skill and experience that was ascertained over the course of this venture. Therefore, this individual project can be counted as a successful endeavour as a result of the investigation undertaken.

Reflection on Learning

Over the course of this project I have challenged myself to learn new processes and deliver an individual project that addresses a very impactful problem which affects millions of people on a daily basis . I find that providing sound analytical support on a professional level is extremely rewarding, especially when I am part of a group more at-risk of contracting this disease. This project had many challenges to overcome but still taught me valuable skills that I will be using in future projects.

Learning and Growth

This project was the most challenging academic venture I have undertaken while being an undergraduate student. Both the technical knowledge and the soft skills I have applied and internalised over the course of this project was phenomenal. Since my second year of study at Cardiff University, I have wanted to pursue a career in data science. Successful analysts must discover useful information from piles of data and provide tangible solutions. This project gave me the chance to investigate an impactful dataset and show my passion for statistical analysis and data visualisation. I learned how to implement data science methodologies in python using the incredible library, Scikit Learn. To not only understand this knowledge but apply it to an important project gave me the chance to identify new statistical techniques and learn how impactful the limitations of project data can be. After completing this report, I went through the individual sections to proofread, format, and add details where necessary. This project has helped me understand that I am a visual/auditory learner because reading through the entirety of this report was tough initially. I therefore decided to go through each section using Microsoft Word's text-to-speech functionality (Read Aloud). This drastically reduced the time used to proofread and was also more akin to my preferred learning styles. I hope to apply these newfound learning techniques in future large-scale projects.

I have experienced significant personal growth in aspects such as time management, motivation and dealing with taxing circumstances. This project took more time than initially anticipated. Originally the aims could have been met with 15 weeks' worth using the Easter recess to accomplish more objectives. The project started slowly, only having 10 to 20 hours of work a week to complete objectives. However, when the project got to the Easter break, the workload ramped up because results had to be identified and the report needed to be written. This led to more than 40 hours per week being allocated to complete this project over Easter and the weeks leading up to the hand-in date. I set myself objective deadlines for the remainder of the project which proved useful as to not overload myself with work when project blockades were present. I also made sure that I was not working more than 8 hours per day, as every day required myself to complete a different section of the project. A-lot of the time was spent waiting for the hyperparameter tuned models to be defined each time the Google Colab session was terminated (which usually happened daily). In this process I learned how to separate the objectives and prioritised the tasks that were critical to the project completion. This did mean that extra functionality was dropped from the final program, as the initial investigation into neural networks did not show any tangible results for the project.

Obstacles to Project

This project started off during a time when the university was open for business as usual. This changed after government restrictions were put in place to control the spread of a potentially fatal disease known as Coronavirus (Covid-19). Covid-19 is the greatest threat and challenge to everyone in this generation. The UK currently has the highest Covid-19 death rate of all countries in Europe. [44] Because of this, the UK government have enforced the closure of all unessential locations and

imposed heavy restrictions on all forms of travelling. The disruption caused by the virus has meant that many processes that would have been useful in the completion of this project became inaccessible for me. The foremost example is my utilisation of the university libraries. A personal methodology that I embody when working on university projects is to separate my workspace from my leisure places. But this legally enforced lockdown has meant that I am unable to visit the university libraries to use as my place of work. This therefore meant that I have since been confined to my small bedroom which in-turn caused intense productivity issues over the final weeks of the project. Simply because my leisure space then also became my workspace, exercise space and rest space. This added stress and lethargy to my life causing the end project to become less polished than originally intended. However, I still persevered and ended up creating an extensive report that highlighted all the areas I aimed for, even after a safety-net policy was deployed to make sure no graduates were disadvantaged.

In my initial plan, I created a risk matrix which partially accounted for this situation. Risk 1 was created in-case “Injury or Illness, having to take leave to recover” which is a generalised explanation of the situation we ended up facing in this project. The mitigation strategy adopted was to “Focus on critical deliverables (project crashing) and file for extenuating circumstances if injury/illness requires significant recovery”, which proved to be substantially useful when confronting these unprecedented circumstances. In this instance I focussed on delivering the core functionality for the project and abandoned the implementation of a deep learning model when the initial results were irrelevant for the project. No further action was required for extenuating circumstances because the university had already put practices in place which meant that grades would not suffer as a result of the virus. [45] Even with these challenges, I was able to overcome them to produce the report as the situation only slowed production and didn’t stop the project.

During Easter recess I was fortunate enough to get an interview with the Government Statistical Service (GSS). [46] This required me to take time away from the project to work on getting a career with the GSS. This meant that I did not have as much time to focus on the neural network deliverable that was originally meant to be produced by the end of the Easter break. Fortunately, I was successful in interview and will be working with government statistics after graduation. This process demonstrated that risks can be taken during a project to explore more lucrative rewards through the reprioritisation of work.

To close this report, I would like to express my gratitude to those involved in this process. I have found this project very exciting and have been lucky enough to be work on a medical dataset that has been used by professional researchers before. I hope that other students are able to find data science as rewarded as myself in their future projects.

Table of Abbreviations

Abbreviation	Full Form
HIV	Human Immunodeficiency Virus
HANDS	HIV-associated neurocognitive disorders
MRI	Magnetic Resonance Imaging
sMRI	Structural Magnetic Resonance Imaging
fMRI	Functional Magnetic Resonance Imaging
CD4	Cluster of Differentiation 4
AIDS	Acquired Immune Deficiency Syndrome
RNA	Ribonucleic Acid
ICV	Intracranial Volume
LH_Frontal	Left hemisphere Frontal-lobe
RH_Frontal	Right hemisphere Frontal-lobe
LH_ACC	Left hemisphere Anterior Cingulate Cortex
RH_ACC	Right hemisphere Anterior Cingulate Cortex
LH_Hippo	Left hemisphere Hippocampus
RH_Hippo	Right hemisphere Hippocampus
CC_Total	Total Corpus-callosum
LH_Amygdala	Left hemisphere Amygdala
RH_Amygdala	Right hemisphere Amygdala
LH_Caudata	Left hemisphere Caudate
RH_Caudata	Right hemisphere Caudate
LH_Putamen	Left hemisphere Putamen
RH_Putamen	Right hemisphere Putamen
ANOVA	Analysis Of Variance
TP	True Positive
TN	True Negative
FP	False Positive
FP	False Negative
GPU	Graphic Processing Unit
BSD	Berkeley Software Distribution license
LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
KNN	K-Nearest Neighbour
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
UK	United Kingdom
GSS	Government Statistical Service

Appendices

Figure 23: Pairplot of Feature Selection MRI Areas

Classifier	Method	Accuracy %	Mean %
LDA	Baseline random split	58.06	58.88
	Baseline cross-validation	52.42	
	Baseline repeated stratified CV	55.91	
	Baseline shuffle split CV	57.89	
	Random hyperparameter grid	61.29	
	Tuned hyperparameter grid	67.74	
	Tuned hyperparameter CV	58.87	
SVM	Baseline random split	54.84	53.02
	Baseline cross-validation	50.00	
	Baseline repeated stratified CV	52.42	
	Baseline shuffle split CV	55.26	
	Random hyperparameter grid	35.48	
	Tuned hyperparameter grid	67.74	
	Tuned hyperparameter CV	55.38	
Logistic Regression	Baseline random split	64.52	59.04
	Baseline cross-validation	54.84	
	Baseline repeated stratified CV	57.26	
	Baseline shuffle split CV	56.58	
	Random hyperparameter grid	58.06	
	Tuned hyperparameter grid	64.52	
	Tuned hyperparameter CV	57.53	
Random Forest	Baseline random split	51.61	53.32
	Baseline cross-validation	51.61	
	Baseline repeated stratified CV	45.16	
	Baseline shuffle split CV	50.66	
	Random hyperparameter grid	64.52	
	Tuned hyperparameter grid	61.29	
	Tuned hyperparameter CV	48.39	
KNN	Baseline random split	48.39	50.60
	Baseline cross-validation	51.61	
	Baseline repeated stratified CV	48.12	
	Baseline shuffle split CV	48.03	
	Random hyperparameter grid	51.61	
	Tuned hyperparameter grid	51.61	
	Tuned hyperparameter CV	54.84	

Table 8: Feature Selection Classification Accuracy per Model

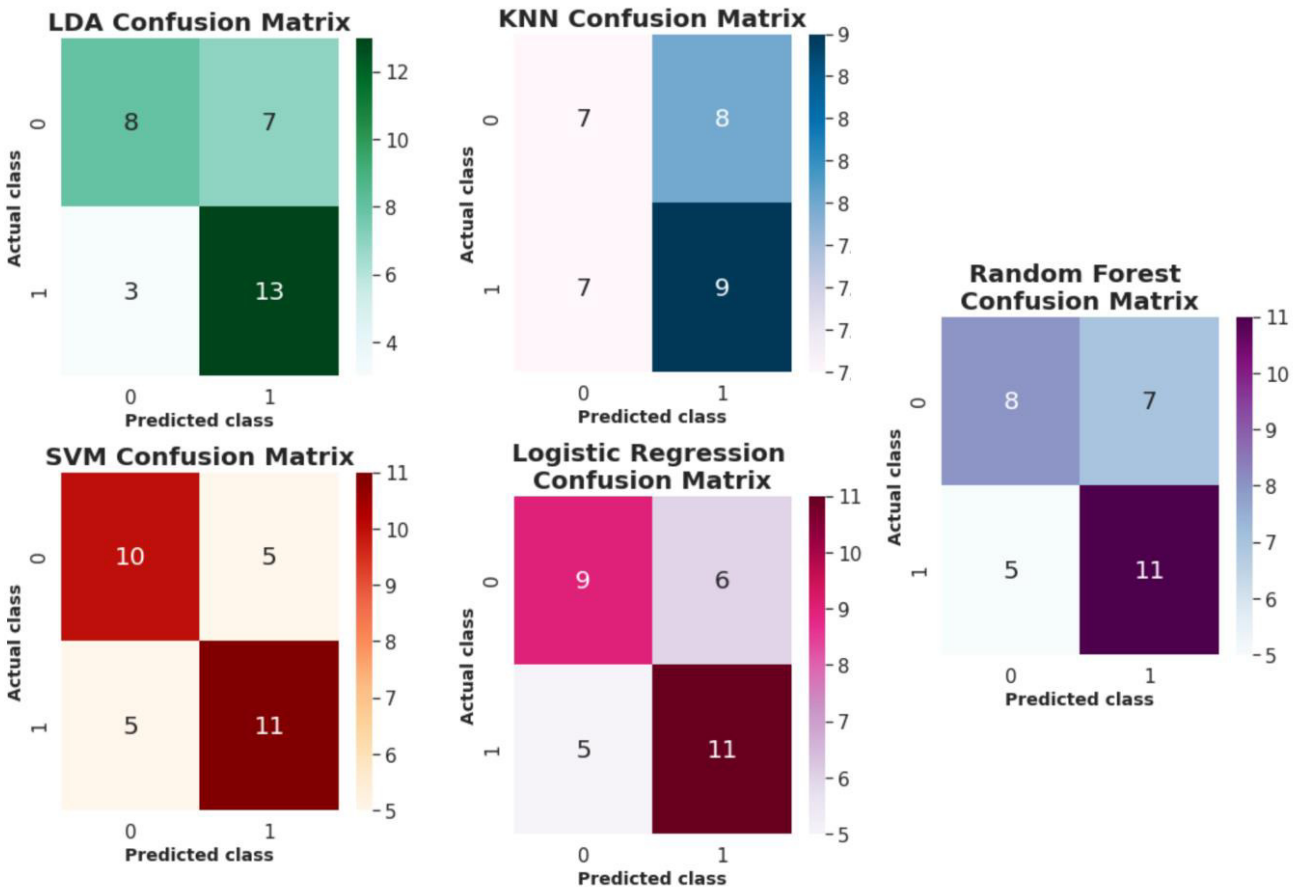

Figure 24: Confusion Matrix per Feature Selection Classifier

Classifier	Metric	Score
KNN	Accuracy	53.19
	F1	50.70
	Precision	57.08
	ROC_AUC	54.95
	Recall	49.01
LDA	Accuracy	56.19
	F1	61.19
	Precision	59.46
	ROC_AUC	59.87
	Recall	66.27
Logistic Regression	Accuracy	54.04
	F1	51.03
	Precision	62.93
	ROC_AUC	57.89
	Recall	53.43
Random Forest	Accuracy	46.44
	F1	49.69
	Precision	51.43
	ROC_AUC	50.81
	Recall	51.68
SVM	Accuracy	55.27
	F1	56.54
	Precision	58.70
	ROC_AUC	58.08
	Recall	56.71

Table 9: Feature Selection Performance Scores

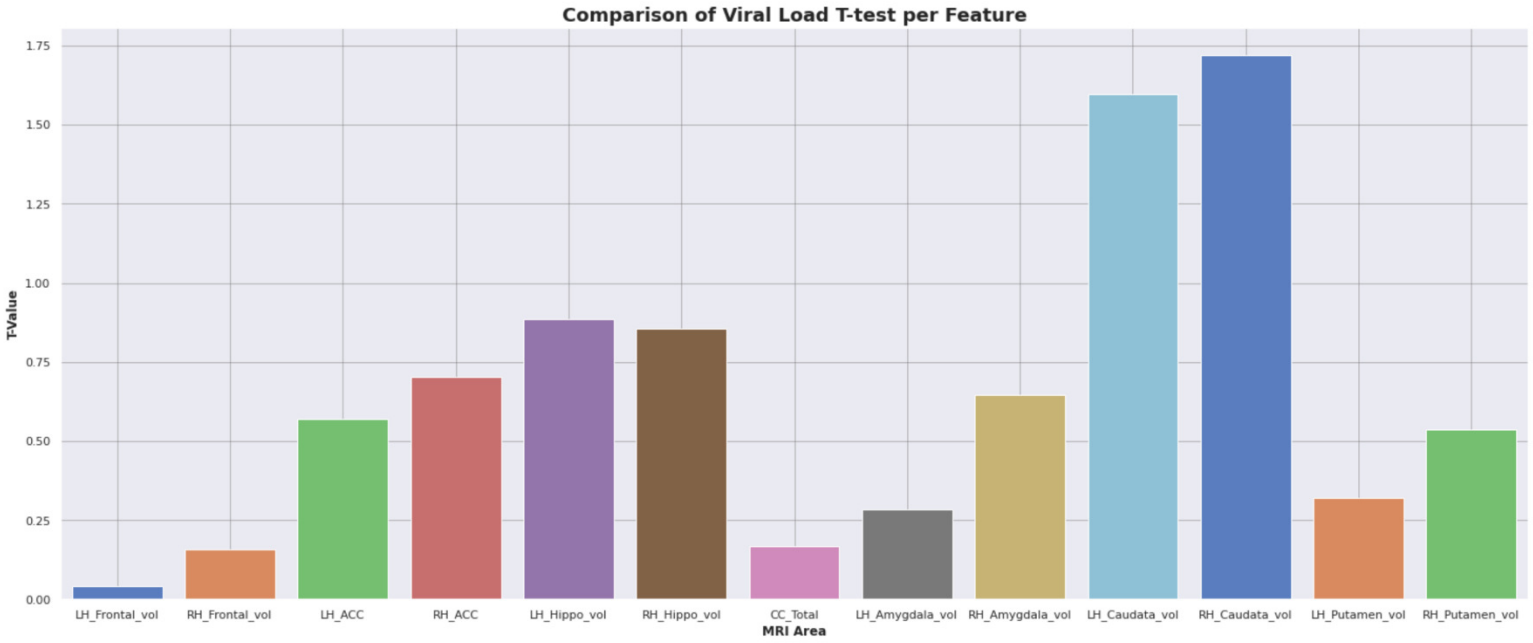

Figure 25: Comparison of Viral Load T-Values per Feature

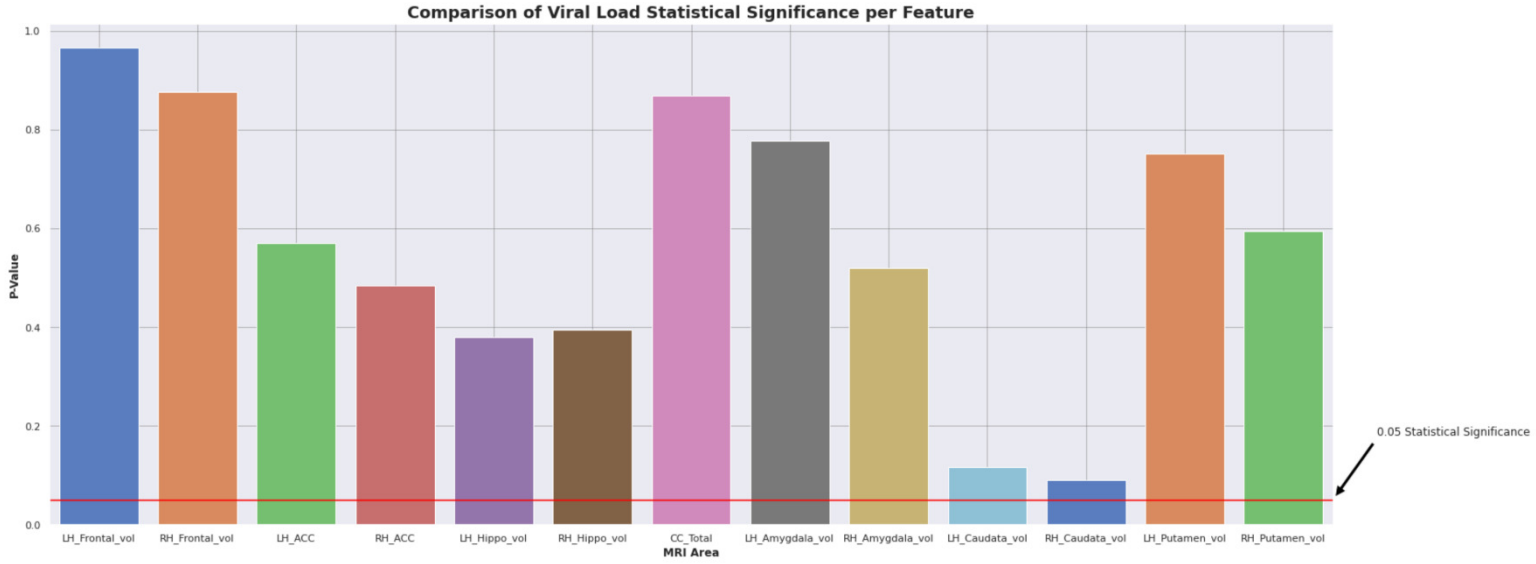

Figure 26: Comparison of Viral Load P-Values per Feature

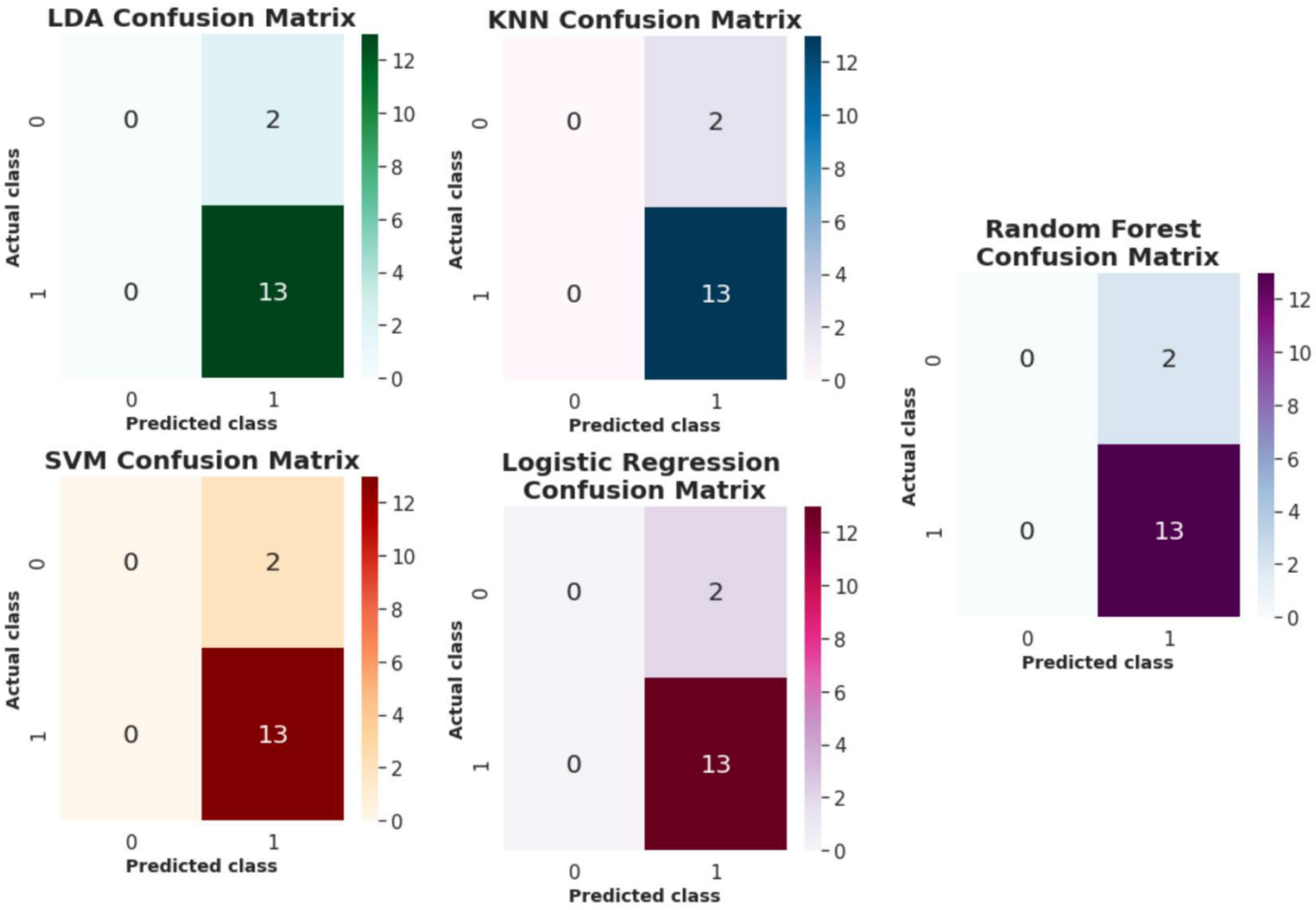

Figure 27: Viral Load Confusion Matrix per Classifier

References

[1] U. D. hiv.gov, “Global Statistics HIV.gov,” 2019. [Online]. Available: https://www.hiv.gov/hiv-basics/overview/data-and-trends/global-statistics. [Accessed 22 02 2020].

[2] NHS, “HIV and AIDS,” NHS, 03 04 2018. [Online]. Available: https://www.nhs.uk/conditions/hiv-and-aids/. [Accessed 23 04 2020].

[3] HIV.gov, “Lab Tests and Results,” U.S. Department of Health & Human Services, 14 02 2017. [Online]. Available: https://www.hiv.gov/hiv-basics/staying-in-hiv-care/provider-visits-and-lab-test/lab-tests-and-results. [Accessed 24 04 2020].

[4] D. Osowiecki, R. Cohen, K. Morrow, R. Paul, C. Carpenter and T. e. a. Flanigan, “Neurocognitive and psychological contributions to quality of life in HIV-1 infected women,” 2000.

[5] J. Underwood, “Grey and white matter abnormalities in treated HIV-disease and their relationship to cognitive function,” 01 08 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/28387814. [Accessed 23 04 2020`].

[6] UNAIDS, “AIDSINFO,” UNAIDS, 2019. [Online]. Available: http://aidsinfo.unaids.org/. [Accessed 24 02 2020].

[7] G. Spies, “Effects of HIV and childhood trauma on brain morphometry,” Springer, 2015.

[8] M. K. Powell, “Opportunistic Infections in HIV-Infected Patients Differ Strongly in Frequencies and Spectra between Patients with Low CD4+ Cell Counts Examined Postmortem and Compensated Patients Examined Antemortem Irrespective of the HAART Era,” 06 09 2016 . [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5017746/. [Accessed 24 04 2020].

[9] HIV.gov, “Preventing Mother-to-Child Transmission of HIV,” 19 12 2018. [Online]. Available: https://www.hiv.gov/hiv-basics/hiv-prevention/reducing-mother-to-child-risk/preventing-mother-to-child-transmission-of-hiv. [Accessed 25 04 2020].

[10] National_Institute_of_Neurological_Disorders_and_Stroke, “Neurological Complications of HIV and AIDS Fact Sheet,” National Institutes of Health, 06 2019. [Online]. Available: https://www.ninds.nih.gov/disorders/patient-caregiver-education/fact-sheets/neurological-complications-aids-fact-sheet. [Accessed 24 04 2020].

[11] U.S._President’s_Emergency_Plan_for_AIDS_Relief, “THE FIFTH SOUTH AFRICAN NATIONAL HIV PREVALENCE,” CDC, Cape Town, 2018.

[12] V. Damme_W, “Scaling-up antiretroviral treatment in Southern African countries with human resource shortage: how will health systems adapt?,” Social Science and Medicine, 2008.

[13] NHS, “MRI scan,” NHS, 09 08 2018. [Online]. Available: https://www.nhs.uk/conditions/mri-scan/. [Accessed 26 04 2020].

[14] P. M. Thompson, “Thinning of the cerebral cortex visualized in HIV/AIDS reflects CD4+ T lymphocyte decline,” 25 10 2005. [Online]. Available: https://www.pnas.org/content/102/43/15647.long. [Accessed 26 04 2020].

[15] R. Cohen, “Effects of nadir CD4 count and duration of human immunodeficiency virus infection on brain volumes in the highly active antiretroviral therapy era.,” 16 02 2010. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/20113183/. [Accessed 25 04 2020].

[16] T. Kuhn, “An Augmented Aging Process in Brain White Matter in HIV,” Department of Psychiatry and Biobehavioral Sciences, University of California, Los Angeles, 2016.

[17] S. Lakshmanan, “how-when-and-why-should-you-normalize-standardize-rescale-your-data,” 17 05 2019. [Online]. Available: https://medium.com/@swethalakshmanan14/how-when-and-why-should-you-normalize-standardize-rescale-your-data-3f083def38ff. [Accessed 27 04 2020].

[18] scikit-learn.org, “Preprocessing data,” scikit-learn developers (BSD License), 2019. [Online]. Available: https://scikit-learn.org/stable/modules/preprocessing.html. [Accessed 27 04 2020].

[19] Statistics_How_To, “T Test (Student’s T-Test): Definition and Examples,” Statistics How To, 2020. [Online]. Available: https://www.statisticshowto.com/probability-and-statistics/t-test/. [Accessed 27 04 2020].

[20] V. PAULSON, “comparison-between-one-way-and-two-way-anova,” 17 08 2018. [Online]. Available: https://stepupanalytics.com/comparison-between-one-way-and-two-way-anova/. [Accessed 07 05 2020].

[21] StatisticsHowTo, “F Statistic / F Value,” StatisticsHowTo, 2020. [Online]. Available: https://www.statisticshowto.com/probability-and-statistics/f-statistic-value-test/. [Accessed 13 05 2020].

[22] A. Geitgey, “machine-learning-is-fun,” 05 05 2014. [Online]. Available: https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471. [Accessed 27 04 2020].

[23] S. Bhatt, “Reinforcement Learning 101,” 19 03 2019. [Online]. Available: https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292. [Accessed 05 13 2020].

[24] S. Asiri, “machine-learning-classifiers,” 11 06 2018. [Online]. Available: https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623. [Accessed 27 04 2020].

[25] M. Sunasra, “performance-metrics-for-classification-problems-in-machine-learning,” Medium, 11 11 2017. [Online]. Available: https://medium.com/thalus-ai/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b. [Accessed 28 04 2020].

[26] S. Narkhede, “understanding-auc-roc-curve,” 26 06 2018. [Online]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-

68b2303cc9c5%20Receiver%20Operating%20Characteristics%20Area%20Under%20The%20Curve. [Accessed 13 05 2020].

[27] J. Brownlee, “Tune Hyperparameters for Classification Machine Learning Algorithms,” 13 12 2019. [Online]. Available: https://machinelearningmastery.com/hyperparameters-for-classification-machine-learning-algorithms/. [Accessed 13 05 2020].

[28] N. Logallo, “data-science-methodology-101,” towardsdatascience, 30 12 2019. [Online]. Available: https://towardsdatascience.com/data-science-methodology-101-ce9f0d660336. [Accessed 29 04 2020].

[29] S. Gajare, “data-science-methodology-and-approach,” geeksforgeeks, [Online]. Available: https://www.geeksforgeeks.org/data-science-methodology-and-approach/. [Accessed 29 04 2020].

[30] T. A. R Brachman, “IBM Corporation,” IBM Analytics, Somers, NY 10589, 2015.

[31] agilealliance, “agile101,” agilealliance, 2020. [Online]. Available: https://www.agilealliance.org/agile101/. [Accessed 29 04 2020].

[32] gov.uk, “Coronavirus (COVID-19): what you need to do,” gov.uk, 2020. [Online]. Available: https://www.gov.uk/coronavirus. [Accessed 29 04 2020].

[33] chercher, “agile-development-chart,” chercher, [Online]. Available: https://chercher.tech/images/jira/agile-development-chart.png. [Accessed 29 04 2020].

[34] A. Morin, “what-is-a-hypothesis,” 02 01 2020. [Online]. Available: https://www.verywellmind.com/what-is-a-hypothesis-2795239. [Accessed 29 04 2020].

[35] R. Taylor, “1630630-Initial_Plan-HIV_Neuroimaging_Data_Analysis_and_Supervised_Machine_Learning,” Cardiff, 2020.

[36] scikit-learn_developers, “scikit-learn,” scikit-learn.org, 2007 - 2019. [Online]. Available: https://scikit-learn.org/stable/. [Accessed 30 04 2020].

[37] M. Waskom, “what is seaborn and why you use it for data visualization?,” seaborn, 2012-2020. [Online]. Available: http://seaborn.pydata.org/. [Accessed 30 04 2020].

[38] scikit-learn_developers, “6.3. Preprocessing data,” scikit-learn, 2007-2019. [Online]. Available: https://scikit-learn.org/stable/modules/preprocessing.html. [Accessed 30 04 2020].

[39] R. Shaikh, “Feature Selection Techniques in Machine Learning with Python,” 28 10 2018. [Online]. Available: https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e. [Accessed 06 05 2020].

[40] G. F. Volpi, “6 amateur mistakes I’ve made working with train-test splits,” 07 08 2019. [Online]. Available: https://towardsdatascience.com/6-amateur-mistakes-ive-made-working-with-train-test-splits-916fabb421bb. [Accessed 06 05 2020].

[41] M. Sidana, “machine-learning-types-of-classification,” 28 02 2017. [Online]. Available: https://medium.com/@Mandysidana/machine-learning-types-of-classification-9497bd4f2e14. [Accessed 06 05 2020].

[42] R. Agarwal, “Confidence Intervals Explained Simply for Data Scientists,” 23 12 2019. [Online]. Available: https://towardsdatascience.com/confidence-intervals-explained-simply-for-data-scientists-8354a6e2266b. [Accessed 05 14 2020].

[43] S. Narkhede, “understanding-auc-roc-curve,” 26 06 2018. [Online]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5. [Accessed 10 05 2020].

[44] NHS, “Statistics > COVID-19 Daily Deaths,” NHS, 12 05 2020. [Online]. Available: https://www.england.nhs.uk/statistics/statistical-work-areas/covid-19-daily-deaths/. [Accessed 12 05 2020].

[45] C. University, “Coronavirus (COVID-19) information for current students,” Cardiff University, 30 04 2020. [Online]. Available: https://www.cardiff.ac.uk/coronavirus/current-students. [Accessed 12 05 2020].

[46] G. S. Service, “GSS - The Government Statistical Service,” Government Statistical Service, 2020. [Online]. Available: https://gss.civilservice.gov.uk/. [Accessed 12 05 2020].

[47] Statistics_South_Africa, “Mid-year population estimates,” 2014. [Online]. Available: http://www.statssa.gov.za/publications/P0302/P03022014.pdf.

[48] F. F. Kaiser, “The Global HIV/AIDS Epidemic,” 2019. [Online]. Available: https://www.kff.org/global-health-policy/fact-sheet/the-global-hivaids-epidemic/. [Accessed 29 01 2020].

[49] UNAIDS, “South Africa updated August2017_0.png,” Avert, 2018. [Online]. Available: https://www.avert.org/sites/default/files/styles/responsive_articlecustom_user_ipad_1x/public/South%20Africa%20updated%20August2017_0.png?itok=caAQWhb7&timestamp=1567005121. [Accessed 25 04 2020].

[50] R. Taylor, *Own Content,* Cardiff, 2020.

[51] A. Bronshtein, “train-test-split-and-cross-validation-in-python,” 17 05 2017. [Online]. Available: https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6. [Accessed 01 05 2020].