VINÍCIUS ALVES TEIXEIRA

BUILDING ACCESSIBLE AND INTERPRETABLE CARDIOVASCULAR RISK MODELS FOR PUBLIC USE IN BRAZIL

São Paulo

2024

VINÍCIUS ALVES TEIXEIRA

BUILDING ACCESSIBLE AND INTERPRETABLE CARDIOVASCULAR RISK MODELS FOR PUBLIC USE IN BRAZIL

Trabalho de Formatura apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do Diploma de Engenheiro de Produção.

São Paulo

2024

VINÍCIUS ALVES TEIXEIRA

BUILDING ACCESSIBLE AND INTERPRETABLE CARDIOVASCULAR RISK MODELS FOR PUBLIC USE IN BRAZIL

Trabalho de Formatura apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do Diploma de Engenheiro de Produção.
Orientador: Renato de Oliveira Moraes

São Paulo

2024

[página dedicada a catalogação]

ABSTRACT

Cardiovascular diseases (CVDs) are the leading cause of mortality in Brazil, accounting for nearly 400,000 deaths annually. Despite advancements in healthcare, disparities in prevention and diagnosis persist, driven by socioeconomic inequalities and the prevalence of modifiable risk factors. This thesis addresses these challenges by developing accessible and interpretable machine learning models for predicting CVD risk, tailored for public use in Brazil. Utilizing the PNS 2019 dataset, a comprehensive health survey by the Brazilian Institute of Geography and Statistics, the study explores logistic regression, K-nearest neighbors, Random Forests and XGBoost models. These models were rigorously optimized through feature selection, oversampling techniques, and hyperparameter tuning, prioritizing recall to enhance early detection of high-risk cases.

The research culminated in the deployment of a digital tool, designed to provide individuals with actionable health insights while adhering to ethical guidelines and prioritizing accessibility. By balancing accuracy with interpretability, the study ensures that the tool remains practical for non-specialist users while addressing critical issues like data privacy and healthcare equity. This work demonstrates the transformative potential of integrating machine learning into public health, offering a scalable framework that empowers individuals, supports healthcare systems, and contributes to reducing the burden of CVDs in Brazil. The findings underscore the importance of combining technical innovation with societal relevance to drive meaningful improvements in public health outcomes.

Keywords: Data analysis, Preprocessing techniques, Feature Selection, Machine learning model, Python

RESUMO

As doenças cardiovasculares são a principal causa de mortalidade no Brasil, responsáveis por cerca de 400 mil mortes anuais. Apesar dos avanços na área da saúde, ainda existem disparidades significativas na prevenção e no diagnóstico, impulsionadas por desigualdades socioeconômicas e pela alta prevalência de fatores de risco modificáveis. Este trabalho aborda esses desafios por meio do desenvolvimento de modelos de aprendizado de máquina acessíveis e interpretáveis para a predição de risco de doenças cardiovasculares, adaptados para uso público no Brasil. Utilizando o conjunto de dados da Pesquisa Nacional de Saúde (PNS) 2019, uma pesquisa abrangente realizada pelo Instituto Brasileiro de Geografia e Estatística (IBGE), foram explorados os modelos de regressão logística, K-vizinhos mais próximos (KNN), *Random Forests* e XGBoost. Esses modelos foram otimizados rigorosamente por meio de técnicas de seleção de variáveis, *oversampling* e ajuste de hiperparâmetros, priorizando o recall para melhorar a detecção precoce de casos de risco.

O estudo culminou no desenvolvimento de uma ferramenta digital projetada para fornecer informações de saúde acionáveis aos indivíduos, respeitando diretrizes éticas e priorizando a acessibilidade. Ao equilibrar precisão e interpretabilidade, o trabalho garante que a ferramenta seja prática para usuários não especializados, ao mesmo tempo em que aborda questões críticas, como privacidade de dados e equidade no acesso à saúde. Este trabalho demonstra o potencial transformador da integração de aprendizado de máquina na saúde pública, oferecendo um framework escalável que empodera indivíduos, apoia sistemas de saúde e contribui para a redução do impacto das doenças cardiovasculares no Brasil. Os resultados ressaltam a importância de combinar inovação técnica com relevância social para promover melhorias significativas nos desfechos de saúde pública.

LIST OF FIGURES

LIST OF TABLES

LIST OF CONTENTS

1. INTRODUCTION

Cardiovascular diseases (CVDs) are the leading cause of death worldwide, with 20 million deaths annually and 620 million individuals currently living with CVDs (MENSAH, 2023). In Brazil, this burden is particularly pronounced, as CVDs account for approximately one-third of all deaths, with nearly 400,000 fatalities annually (MANSUR & FAVARATO, 2021). Despite advancements in healthcare, the persistence of modifiable risk factors such as hypertension, diabetes, and obesity continue to challenge public health efforts, highlighting the critical need for innovative approaches to prevention and diagnosis (PATRIOTA, 2023).

This study focuses on addressing Brazil's CVD crisis by leveraging advanced machine learning and statistical learning methods to predict cardiovascular disease risk. Using the Behavioral Risk Factor Surveillance System 2015 survey dataset, which includes diverse health-related features, this research aims to identify the most effective predictive model and make it available to the Brazilian population. The study will systematically test various algorithms — logistic regression, k-nearest neighbors, and random forests, with variations for each model type with regards to oversampling and hyperparameter tuning techniques — adhering to established best practices in data analysis to ensure robust and generalizable results.

Beyond model development, this research seeks to deploy the best-performing model as a practical tool for public use in Brazil. The tool will prioritize accessibility, interpretability, and ethical application, ensuring that it empowers individuals to understand their cardiovascular health risks while adhering to medical guidelines. By following ethical principles and addressing concerns such as data privacy, health equity, and clinical validity, the tool aims to maximize public benefit and minimize risks of misuse. This initiative aspires to enhance public awareness, promote early intervention, and ultimately contribute to reducing the burden of CVD in Brazil.

Through this endeavor, the study not only demonstrates the potential of predictive modeling in addressing pressing public health challenges but also offers a framework for developing responsible, impactful digital health solutions tailored to regional contexts.

2. LITERATURE REVIEW

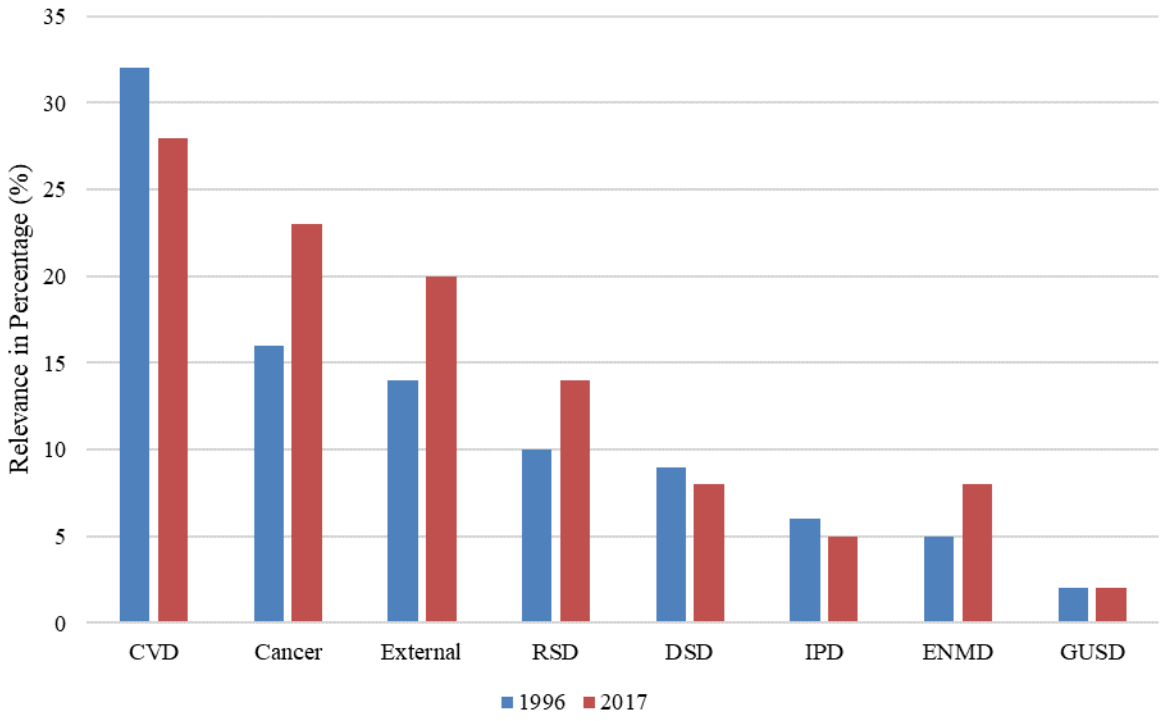
2.1 Cardiovascular Diseases in Brazil

Cardiovascular diseases (CVDs) represent the primary cause of mortality in Brazil, accounting for approximately one-third of all deaths (CASTRO et al, 2019). This prevalence reflects a persistent public health challenge despite ongoing interventions. In 2021, the prevalence of CVDs was estimated at 6.9% across both sexes, with men exhibiting a higher rate of 7.6% compared to women. In 2022, CVDs were responsible for nearly 400,000 deaths among Brazilians (MENSAH, 2023), with ischemic heart disease and stroke remaining the leading causes of CVD-related mortality since the 1960s (KRAUSKOPF, 2019).

The relevance of CVD as a cause of death, when compared to other conditions such as cancer and external causes, underscores its significant impact on public health in Brazil (Figure 1). Although the age-standardized mortality rate for CVD has decreased by 39.1% — from 345 deaths per 100,000 people in 1997 to 210 per 100,000 in 2017 (MANSUR & FAVARATO, 2021) — CVDs continue to place a substantial burden on Brazil's health system. Heart failure, in particular, has become the predominant cause of CVD-related hospitalizations, with over 222,000 admissions reported in 2019 alone (BERWANGER & SANTO, 2022).

Figure 1 underscores CVDs as the leading cause of death in the Brazilian population, with higher prevalence than Cancer, Diseases of the Respiratory System (RSD), Diseases of the Digestive System (DSD), Infectious and Parasitic Diseases (IPD), Endocrine, Nutritional and Metabolic Diseases (ENMD), and Diseases of the Genitourinary System (GUSD).

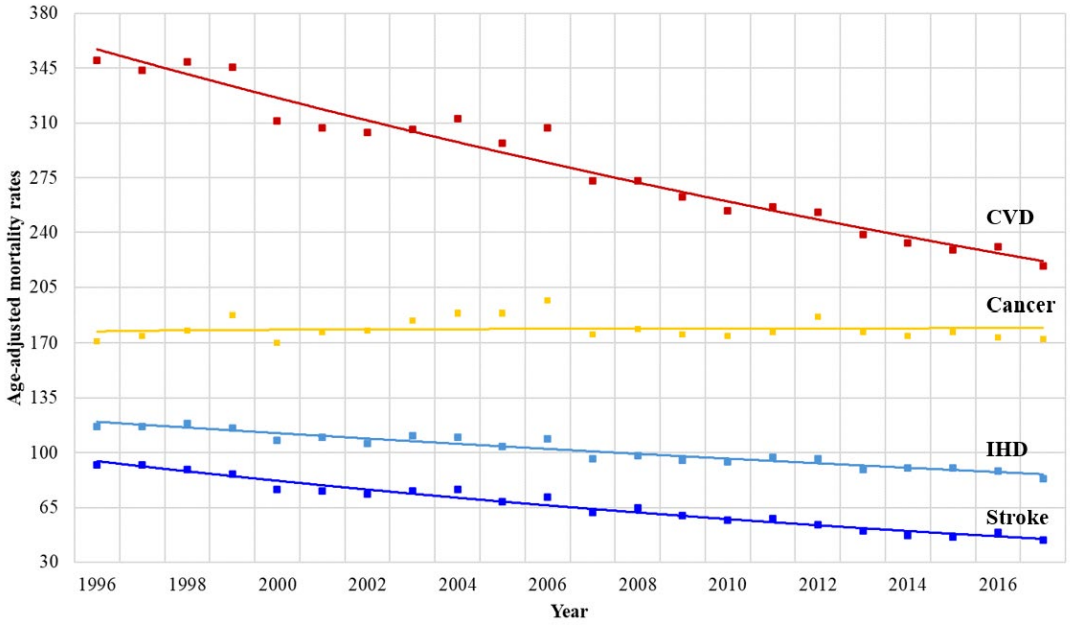
Figure 1: Relevance of main Death Causes in Brazil



Source: The Author, based on data from (MANSUR & FAVARATO, 2021)

This decreasing trend in mortality is illustrated in Figure 2, which highlights the shifts in age-adjusted mortality rates from 1997 to 2017. Two of the most common types of CVDs, Strokes and Ischemic Heart Diseases (IHD) have shown reduction over time. While improvements are evident, the high mortality rate linked to CVDs continues to reflect the persistence of risk factors within the population (MANSUR & FAVARATO, 2021). In fact, in 2019, about 83% of CVD mortality was attributed to modifiable risk factors (BRANDT, 2022). Key risk factors include hypertension, diabetes, dyslipidemia, obesity, smoking, physical inactivity, and an unhealthy diet. Notably, while Brazil has seen reductions in smoking and environmental risks, metabolic risk factors—such as diabetes and high cholesterol—have increased over time (PATRIOTA, 2023).

Figure 2: Trend in Mortality Rates in Brazil by Cause



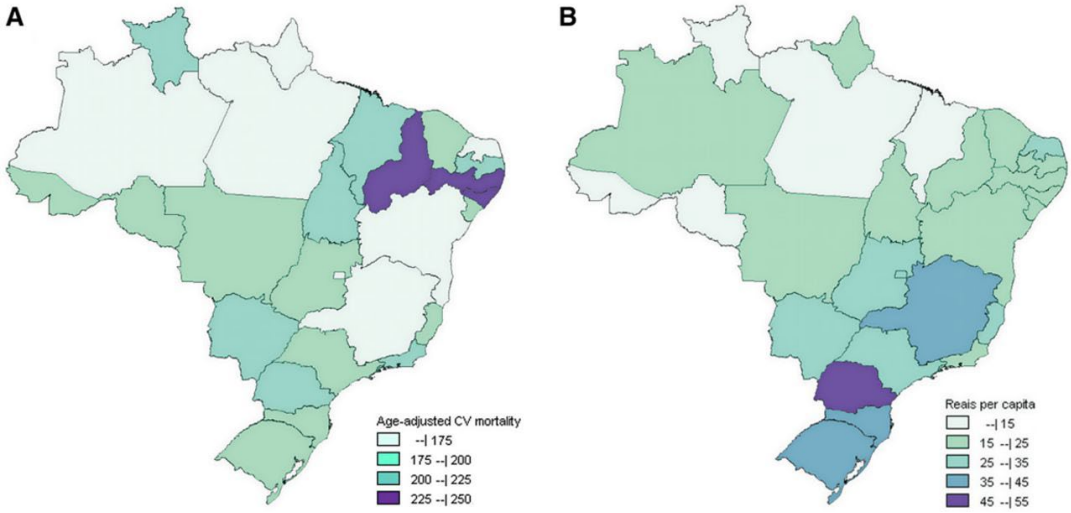
Source: Extracted from (MANSUR & FAVARATO, 2021)

Socioeconomic disparities also significantly influence CVD risk in Brazil. Wealthier, better-educated Brazilians report higher access to lifestyle recommendations for managing conditions like high cholesterol and hypertension than those from lower socioeconomic backgrounds. This disparity manifests in differing rates of hypertension, diabetes, obesity, and smoking across socioeconomic strata, further compounding the public health challenge (PATRIOTA, 2023).

The economic impact of CVDs in Brazil is profound. In 2015, the cost burden was estimated at R$37.1 billion, with 61% attributed to premature mortality and 39% linked to direct and indirect healthcare costs (ARAÚJO & RODRIGUES, 2022). Direct costs encompass expenses related to hospitalizations, monitoring, and treatment, while indirect costs are largely driven by productivity losses due to illness-related absenteeism and mortality.

Geographic disparities further exacerbate the CVD burden in Brazil, with states displaying varied rates of CVD mortality that correlate with socioeconomic development levels (RIBEIRO, 2016). As shown in Figure 3A, CVD mortality rates are particularly high in regions with lower socioeconomic indicators. Furthermore, the financial strain of CVD-related hospitalizations also varies across states, as illustrated in Figure 3B. This disparity in burden is partially attributed to the prevalence of risk factors like tobacco use, poor dietary habits, and elevated LDL cholesterol in states with lower Sociodemographic Indices (SDI) (BRANDT, 2022).

Figure 3: (A) CVD mortality across Brazilian States and (B) CVD-related cost of hospitalizations per capita



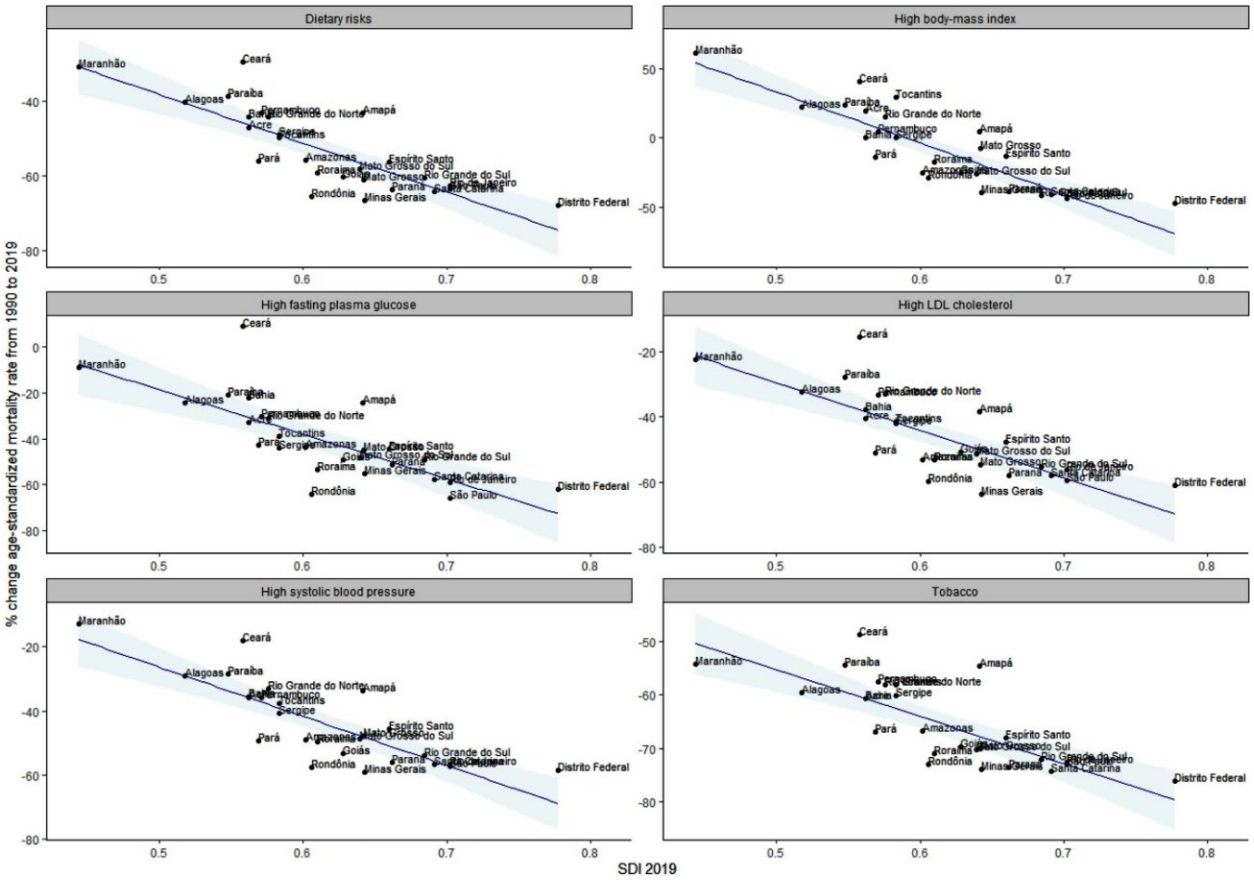
Source: Extracted from (RIBEIRO, 2016)

CVD Risk factors have been widely studied already on a global level. In particular, it is worth mentioning the INTERHEART study led by (YUSUF, 2004). This case-control analysis included more than 15 thousand cases over 52 countries, identifying nine modifiable risk factors that collectively account for over 90% of the global risk for myocardial infarction. These factors include abnormal lipids, smoking, hypertension, diabetes, abdominal obesity, psychosocial stress, inadequate consumption of fruits and vegetables, lack of regular physical activity, and excessive alcohol intake. The study underscores the significance of these risk factors across diverse populations, highlighting the potential for substantial reductions in cardiovascular disease incidence through targeted lifestyle and behavioral interventions. Another important observation

of (YUSUF, 2004) is that, despite the burden of CVD being more than 80% concentrated in low-income and middle-income countries, the studies about CVD risk factors have been mostly conducted in developed countries. This key observation supports the need of the work being done under this Thesis.

Finally, Figure 4 shows the relative change in mortality rates due to CVD attributed to selected Risk Factors for all Brazilian Federated Units, from 1990 to 2019. One can notice that while the mortality rates have generally improved for most Brazilian Federated Units and most Risk Factors, the improvements are far from being evenly distributed. Noticeably, the CVD-linked mortality rates improvements have been significantly greater for the Federated Units with the highest sociodemographic index (SDI) scores. This distribution emphasizes the need for targeted interventions to address both risk factors and structural inequalities contributing to CVD prevalence. In particular, CVD mortality attributed to high Body Mass Index (BMI) has grown for Brazilian Federated Units with the lowest SDI, including Maranhão, Alagoas, Paraíba and Ceará.

Figure 4: Relationship between sociodemographic index and CVD-related mortality rates



Source: Extracted from (BRANDT, 2022)

2.2 Predicting CVD Risk

The integration of predictive modeling in healthcare represents a transformative advancement in medical practice, leveraging data analytics and machine learning to enhance patient care and clinical decision-making. At its core, predictive modeling in healthcare involves the systematic analysis of diverse datasets to identify patterns and forecast future health outcomes (NWAIMO, 2024). This approach has become increasingly sophisticated with the advent of big data analytics, electronic health records (EHRs), and advanced computational capabilities.

The current healthcare landscape employs various predictive modeling techniques, ranging from traditional statistical methods to advanced machine learning algorithms. These include logistic regression, decision trees, random forests, support vector machines, and neural networks, each offering distinct advantages in different clinical contexts (ZHANG, 2020). The selection of appropriate modeling techniques depends on factors such as data characteristics, prediction objectives, and the specific healthcare domain under consideration.

One of the most promising applications of predictive modeling has emerged in cardiovascular disease (CVD) prevention and management. Given that CVD remains a leading cause of mortality worldwide, the development of accurate predictive models has become crucial for early intervention and risk stratification (DEEPA, 2024). Recent studies have demonstrated remarkable success in utilizing machine learning algorithms for CVD prediction, with some models achieving accuracy rates exceeding 80% (SANG, LEE, & LEE, 2019).

The implementation of predictive models in healthcare relies heavily on the quality and comprehensiveness of available data. Electronic Health Records (EHRs) serve as a primary data source, providing detailed patient histories, clinical measurements, and treatment outcomes (NWAIMO, 2024). However, the effective utilization of these data sources presents significant challenges, including data standardization, integration of disparate systems, and the need to address missing or incomplete information (ZHANG, 2020).

In the specific context of cardiovascular disease prediction, modern approaches have evolved to incorporate multiple data types, including clinical measurements, genetic information, lifestyle factors, and even social determinants of health (DEEPA, 2024). The XGBoost algorithm, in particular, has shown promising results in CVD prediction, demonstrating superior performance in handling complex medical data and providing accurate risk assessments (PENG, HOU, & CHENG, 2023).

Despite these advances, several limitations and challenges persist in healthcare predictive modeling. Data quality and standardization remain significant concerns, as does the need for model interpretability in clinical settings1. Healthcare professionals require not only accurate predictions but also clear explanations of the reasoning behind these predictions to make informed clinical decisions (BADAWY & RAMADAN, 2023).

In the realm of cardiovascular disease prediction, current research focuses on developing more sophisticated models that can account for the complex interplay of risk factors while maintaining clinical interpretability (OGUNPOLA, SAEED, & BASURRA, 2024). These efforts aim to bridge the gap between statistical accuracy and practical clinical utility, ensuring that predictive models serve as effective tools in cardiovascular disease prevention and management.

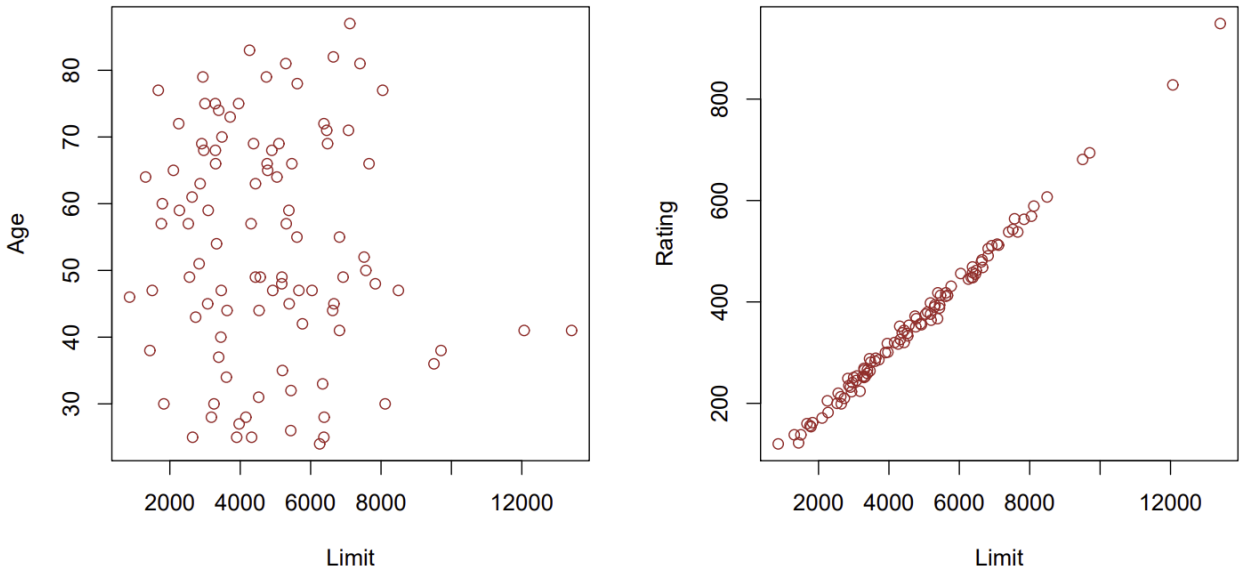
The successful implementation of predictive models in healthcare requires careful consideration of both technical and practical aspects. While the potential benefits are substantial, including improved patient outcomes and more efficient resource allocation, the challenges of data quality, model interpretability, and clinical integration must be systematically addressed (YANG, 2022). This understanding forms the foundation for developing more effective predictive models for cardiovascular disease, contributing to the broader goal of enhancing preventive care and reducing the burden of CVD in Brazil.

2.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a critical initial phase in predictive modeling, including applications in cardiovascular disease (CVD) prediction. EDA involves summarizing and visualizing data to uncover patterns, relationships, and potential anomalies before applying formal statistical or machine learning methods (JAMES, 2013). EDA helps in understanding the underlying data distribution, identifying significant variables, and exploring preliminary associations that may guide feature engineering and model selection in subsequent stages.

A fundamental component of EDA includes visualization techniques, such as histograms, scatter plots, and box plots, as illustrated by Figure 5. These techniques are instrumental in examining the distribution and potential outliers in predictor variables. These visual tools allow for detecting non-linear relationships and assessing the necessity for transformation or normalization, particularly essential in datasets with diverse variable types as seen in health-related studies (JAMES, 2013).

Figure 5: Sample scatterplots used during the EDA phase



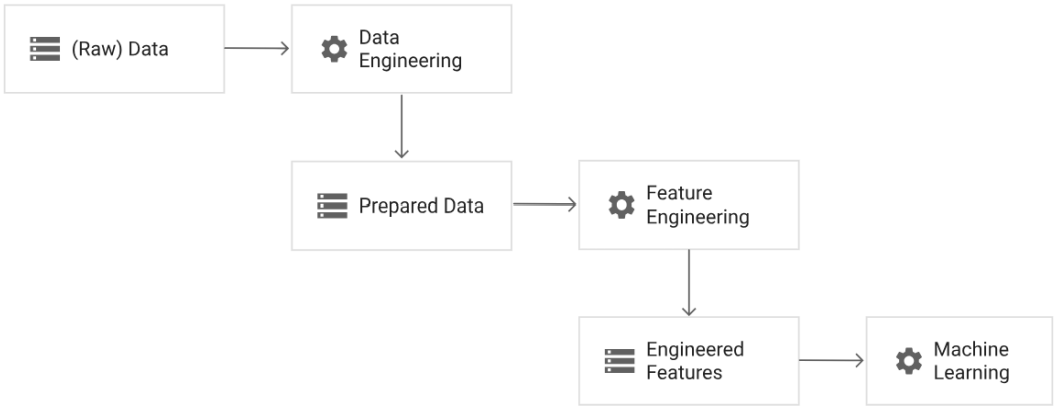
Source: Extracted from (JAMES, 2013)

2.4 Data Preprocessing

Data preprocessing is a foundational step in the machine learning workflow, directly impacting the quality and reliability of predictive models. It involves the transformation of raw, unstructured data into a clean and structured format that ensures consistency, accuracy, and suitability for analysis. This process addresses the challenges posed by real-world datasets, which often contain missing values, noise, and inconsistencies that can hinder the performance of algorithms. By systematically applying data preprocessing techniques, practitioners can ensure that models are trained on high-quality data, leading to more accurate and reliable predictions (TENSORFLOW, n.d.).

To illustrate the overall process, Figure 8 shows a typical workflow from raw data to machine learning. It emphasizes the sequential transformation stages, including data engineering and feature engineering, which culminate in prepared datasets ready for model training.

Figure 6: Data Preprocessing Workflow



Source: (TENSORFLOW, n.d.)

One key concept to explore on the Data Preprocessing side is Data Cleaning. Data cleaning is an essential first step in the machine learning pipeline, ensuring that data quality issues such as missing values, outliers, and inconsistencies are addressed. Missing values can be imputed using

techniques like mean or median replacement for numerical data or the most frequent category for categorical variables. (GÉRON, 2017)

Outliers, which can distort predictions, are typically identified using statistical techniques such as the interquartile range (IQR) or z-scores. For instance, an observation x_i can typically be considered an outlier if $|z_i| > 3$, where $z_i = \frac{x_i - \mu}{\sigma}$, with μ being the mean and σ the standard deviation for the dataset under consideration. Handling outliers involves either removing them or transforming the data to reduce their impact (GÉRON, 2017).

Another important step on the Data Preprocessing phase is the Data Transformation. Data Transformation ensures compatibility between raw data and machine learning algorithms. Scaling and Normalization are two key techniques used for continuous variables (GÉRON, 2017). Scaling, often achieved using min-max normalization, transforms each feature x to a standard range of [0,1].

$$x' = \frac{(x - x_{\min})}{x_{\max} - x_{\min}}$$

Where $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the feature, respectively. This approach is particularly useful when features have differing units or scales.

Normalization, on the other hand, adjusts features to have a mean of 0 and a standard deviation of 1, expressed mathematically as:

$$z = \frac{x - \mu}{\sigma}$$

Where μ is the mean and σ is the standard deviation of feature x. Since most supervised learning methods are sensitive to feature magnitudes, both Scaling and Normalization techniques are often employed on the Data Preprocessing step, facilitating faster convergence during model training and ensuring that features contribute evenly to the algorithm (GÉRON, 2017).

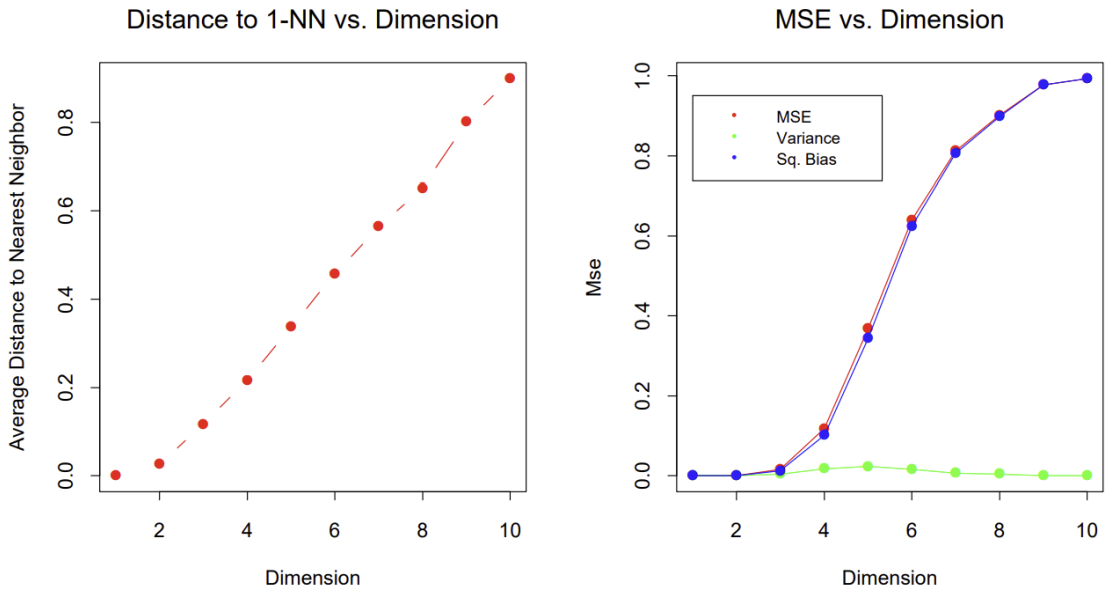
2.5 The Curse of Dimensionality

The "curse of dimensionality" refers to the various challenges that arise when working with high-dimensional data, as is the case in the present work. As the number of features or dimensions p increases, the volume of the feature space grows exponentially, leading to sparse data coverage even with a large dataset. This sparsity undermines the reliability of distance-based methods like k-nearest neighbors, as the distance between any two points becomes nearly uniform, thereby diminishing the distinctions between data points that are crucial for prediction (JAMES, 2013).

One primary issue in high-dimensional spaces is that the amount of data required to populate the feature space grows exponentially with the number of dimensions, making it difficult to estimate parameters accurately. Capturing a fixed proportion of the data requires neighborhoods that cover increasingly larger regions as dimensionality increases, which makes meaningful “local” analysis in high dimensions nearly impossible (FRIDMAN, HASTIE, & TIBSHIRANI, 2008). This often leads to overfitting, as models become too sensitive to the noise in the training data, capturing chance correlations that do not generalize well to new data.

Figure 7 shows how model performance quickly deteriorates in a distance-based model setting, for a fixed dataset size and increasing dimension. As predicted by the “Curse of Dimensionality”, the average distance to nearest neighbors grows as the feature complexity increases, with test error (as measured by the MSE) growing fast when dimension is greater than 3, likely due to overfitting.

Figure 7: Charts of Distance and MSE on the Curse of Dimensionality



Source: Extracted from (FRIDMAN, HASTIE, & TIBSHIRANI, 2008)

In practical terms, the curse of dimensionality often manifests as a tradeoff: while adding more features could, in theory, improve a model's predictive power, irrelevant or redundant features tend to degrade model performance. The inclusion of non-informative features increases the likelihood of overfitting without adding predictive value, exacerbating the model's variance without significantly reducing bias. Thus, techniques like regularization or feature selection are often necessary to manage high-dimensional data effectively.

2.6 Bias-Variance Tradeoff

In predictive modeling, bias refers to the error that results from overly simplistic assumptions in the model's structure. This occurs when a model fails to capture the true complexity of the data-generating process, often due to a restrictive framework that doesn't allow for sufficient flexibility in the relationship between input features and the target variable. For example, assuming a linear relationship where a non-linear one exists introduces systematic error, leading to consistently biased predictions. This simplification, while making the model easier to interpret and less prone to overfitting, often limits its accuracy on real-world data (JAMES, 2013).

Variance, on the other hand, reflects a model's sensitivity to fluctuations in the training data. High-variance models adapt closely to the specifics of the training data, capturing noise as if it were signal. This sensitivity typically occurs in more flexible models, which may perform well on training data but poorly on unseen data due to their tendency to "overfit" to the idiosyncrasies of the training set. High variance results in significant differences in model performance across different training datasets, undermining generalizability (FRIDMAN, HASTIE, & TIBSHIRANI, 2008).

The expected error of a model is measured by the Test Median Square Error (Test MSE). Test MSE can be decomposed in three fundamental quantities (JAMES, 2013):

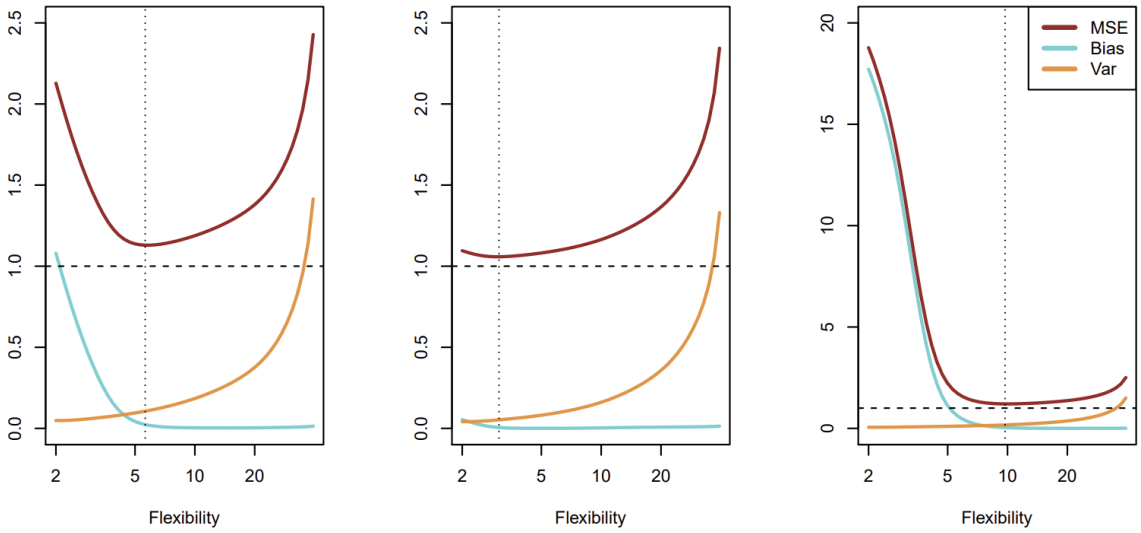
$$E(y_0 - \hat{f}(x_0))^2 = Var\left(\hat{f}(x_0)\right) + \left[Bias\left(\hat{f}(x_0)\right)\right]^2 + Var(\varepsilon)$$

As a result, one can observe there's an optimal level of model complexity (measured in terms of Flexibility) in order to minimize for test MSE or optimize model performance. In other words, a model that's too complex will start incorporating noise from the training data, ultimately predicting patterns that do not match reality.

Figure 6 illustrates the expected behavior for Test MSE, Bias and Variance. One can see there's an optimal level of flexibility (model complexity) that minimizes Test MSE. Since Test

MSE is our proxy of how the model will perform in real-world scenarios, we thus conclude that models of higher complexity are not always preferred. Finding out the optimal level of complexity across different model set ups will be critical to the current work.

Figure 8: Squared bias, Variance and Test MSE for three different datasets



Source: Extracted from (JAMES, 2013)

The bias-variance tradeoff is a central paradigm in supervised learning, underscoring the balance between underfitting and overfitting. Generally, as model flexibility increases, bias decreases but variance rises, and vice versa. The goal is to find a model that reduces both bias and variance to the extent possible, minimizing test error by aligning the model's complexity with the inherent patterns in the data while avoiding excessive sensitivity to training-specific noise (FRIDMAN, HASTIE, & TIBSHIRANI, 2008).

2.7 Logistic Regression

Logistic Regression is a foundational algorithm in supervised machine learning, being widely employed for classification tasks (problems with discrete output, like binary outputs), not being suitable for regression tasks (problems with continuous output).

Logistic Regression models the probability than an input belongs to a specific class, making it particularly effective for problems that require probabilistic predictions. Unlike linear regression, which predicts continuous values, logistic regression outputs probabilities between 0 and 1. These probabilities can then be converted into class predictions using a threshold (GÉRON, 2017).

The algorithm is based on modeling the log-odds (also known as the logit) of the dependent variable y as a linear combination of the independent variables X. This relationship is expressed mathematically as:

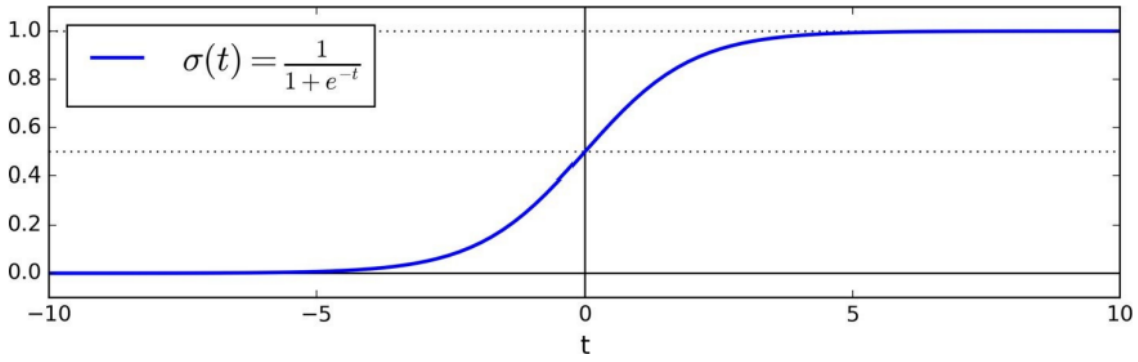
$$\log\left(\frac{P(y=1)}{P(y=0)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n$$

Where $P(y=1)$ represents the probability of the positive class, and $\beta_0, \beta_1, \ldots, \beta_n$ are the model coefficients. To ensure the predicted probabilities are confined to the range [0, 1], the log-odds are passed through the sigmoid function, given by (GÉRON, 2017):

$$P(y=1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n)}}$$

This transformation enables logistic regression to output probabilities and subsequently classify observations based on a chosen threshold. The visual of this important capability of the logistic function is shown in Figure 9, where one can see that the logistic function is able to compress any outcome t into a distribution [0,1], making it suitable to represent probabilities.

Figure 9: The Logistic Function



Source: Extracted from (GÉRON, 2017)

One of the key advantages of logistic regression is the interpretability of its coefficients. Each coefficient β_i quantifies the effect of a one-unit increase in the corresponding predictor X_i on the log-odds of the outcome, assuming all other predictors remain constant. Logistic regression is also computationally efficient, making it suitable for handling large datasets and high-dimensional feature spaces. Additionally, it provides more than just classification; the probability estimates allow for nuanced decision-making and insights into prediction confidence. To address overfitting, particularly in datasets with many features, regularization techniques such as L1 (Lasso) and L2 (Ridge) are often incorporated into logistic regression (GÉRON, 2017).

The algorithm is widely used in applications across diverse fields. In healthcare, it is employed to predict the likelihood of diseases based on patient features, while in finance, it is used to identify fraudulent transactions. In marketing, logistic regression aids in customer churn prediction and segmentation. Its flexibility and effectiveness in binary classification problems make it a cornerstone of predictive analytics.

Training a logistic regression model involves maximizing the likelihood of the observed data, which is achieved through Maximum Likelihood Estimation (MLE). The log-likelihood function, which forms the basis of the optimization process, is expressed as (JAMES, 2013):

$$L(\beta) = \sum_{i=1}^{N} [y_i + \log\big(P(y_i)\big) + (1 - y_i)\log\,(1 - P(y_i))]$$

Where N is the number of training examples. Optimization techniques such as Gradient Descent are used to iteratively adjust the coefficients β to maximize the log-likelihood function and fit the model to the training data.

While logistic regression is a powerful tool, it has some limitations. The model assumes a linear relationship between the predictors and the log-odds of the outcome, which may not hold in datasets with complex interactions. Additionally, logistic regression is inherently designed for binary classification tasks, and extensions such as multinomial logistic regression or one-vs-all strategies are required to handle multiclass problems.

2.8 K-Nearest Neighbors

The second algorithm to be explored in this thesis is the K-Nearest Neighbors (KNN) algorithm. The KNN algorithm is another widely used and foundational approach in supervised learning, being used for both classification and regression problems. As the problem under consideration for this Thesis is a classification problem, the following section will explore how KNNs are constructed and used on classification settings.

KNN is a non-parametric algorithm, meaning that it doesn't make assumptions about the underlying data distribution or involve fixed parameters in order to build a model. Instead, its foundation lies on the principle that similar observations tend to have similar outcomes, making predictions based on the proximity of data points in the feature space (JAMES, 2013).

That said, for a given test observation x_0 which we aim to classify, the KNN classifier first identifies the K points in the training data that are closest to x_0. On a multidimensional space, Distance can be defined in different ways, including the Euclidean Distance definition which is intuitively embedded into the real world, or other definitions such as the Manhattan or Minkowski distances, which can be preferred depending on the context. In the present work, distances will be calculated using the Euclidian definition, which for two points x and x' in an n-dimensional space can be defined as follows:

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2}$$

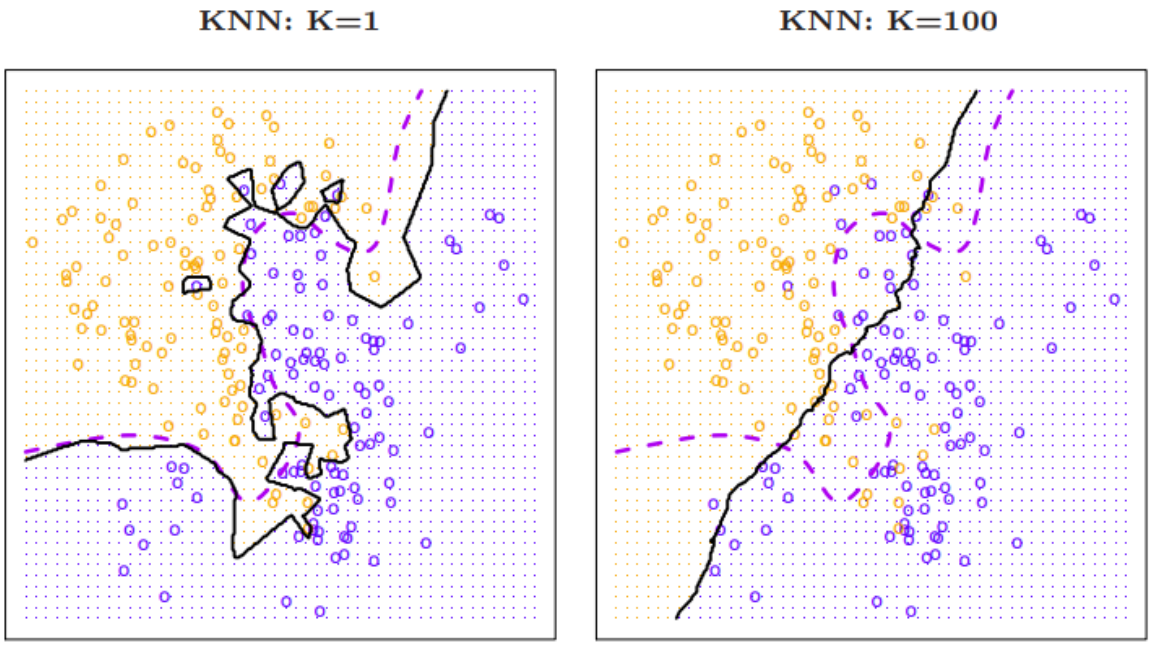
Once the distances between a test observation x_0 and all training observations are calculated, the K observations in the training dataset with the closest distance to x_0 will be considered the K nearest neighbors, represented by $\mathcal{N}_0$. The KNN classifier then estimates the conditional probability for a class j as the fraction of points in $\mathcal{N}_0$ whose response values are equal to j (JAMES, 2013). Mathematically, probabilities are assigned as follows:

$$\Pr(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j)$$

After calculating the probabilities for every class j in the setting above, the KNN classifier will simply classify x_0 as the class with the largest probability. By choosing the class with the highest probability, the KNN classifier minimizes the classification error, consistent with the Bayesian Decision Rule (SAMMUT, 2010).

It's important to highlight that the choice of K has a great effect on the result obtained by the KNN classifier. Figure 10 depicts two distinct scenarios, one for a KNN classifier constructed with $K = 1$, and other with $K = 100$. Both were trained and set to make predictions over the same training dataset. The decision boundaries for both models are shown by the solid black curves, with the true nature of the observations being differentiated by color (orange or purple), and the purple dashed line showing the Bayes decision boundary (an ideal classifier that minimizes test error but can't be achievable without the explicit conditional distribution of Y given X).

Figure 10: KNN decision boundaries for low and high K values



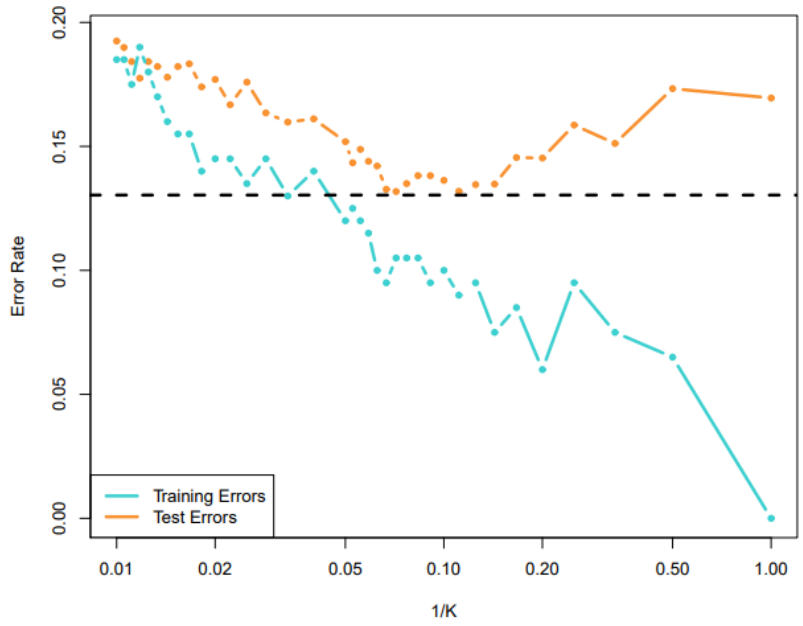
Source: Extracted from (JAMES, 2013)

One can notice that the decision boundary for when $K = 1$ tends to be overly flexible for this given problem, with the model adapting to patterns that don't represent the true behavior of the data, incorporating noise into the predictions. This results in a model with low bias but very high variance, that has an extremely low error rate in the training data set (in this case, and error rate of precisely 0), but not satisfactory error rates in the test data set. Given the Test Error Rate is the proxy for how well the model will perform in real-world scenarios, $K = 1$ leads to overfitting (JAMES, 2013).

On the opposite side, it's also important to clarify why attributing working with a K that is too high is also not desirable. By comparing the setting of $K = 100$ with Bayes decision boundary (the ideal classifier), it's noticeable how the model fails to capture some of the nuances present in the true training data distribution (JAMES, 2013).

Figure 11 shows how Training Errors generally keep going down as K decreases (1/K increases), but that Test Errors are minimized for an optimal value of K (in this case, around K=10 or 1/K = 0.10). The process of finding the best K for a model is very important and requires a structured approach to it, being part of the broader theme of Hyperparameter Tuning.

Figure 11: KNN error rates as a function of K



Source: Extracted from (JAMES, 2013)

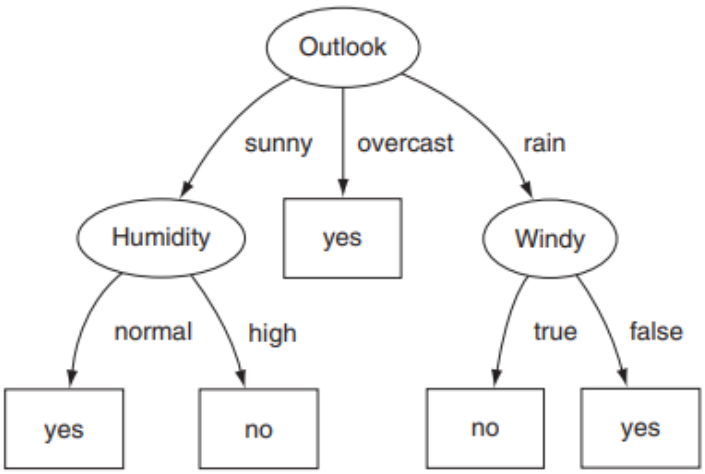
2.9 Random Forests

The third and last type of model explored in this Thesis is Random Forests. Random Forests widely differ from the approaches used by Logistic Regression and KNN models and is arguably the most complex and robust technique of all three.

The Random Forests technique was first introduced by (BREIMAN, 2001) and is built upon the concept of Decision Trees. A Decision Tree works by recursively partitioning the feature space into distinct regions, creating a hierarchical structure of decision rules (JAMES, 2013).

An anecdotic visualization of a Decision Tree is shown in Figure 12, for a problem in which the goal is understanding whether conditions are suitable for playing golf (target variable) based on weather conditions (features are Outlook, Humidity, Windy or Temperature).

Figure 12: Decision Tree Example



Source: Extracted from (SAMMUT, 2010)

The construction of a Decision Tree begins with a top node known as the Root. From the Root, decision rules are generated based on the features, creating additional internal nodes. This process continues until terminal nodes, called Leafs, are reached. The Leafs display the predicted values (in this case, "Yes" or "No" for favorable golfing conditions), with the predicted class determined the most frequent target value in the corresponding group of the training dataset (SAMMUT, 2010).

In more complex settings, however, the construction of Decision Trees requires careful consideration what are the best splits (decision rules) and how deep such tree should be. It is desirable that the Decision Tree separates the data into homogeneous and relevant groups, optimize model performance and avoid overfitting.

Formally, the homogeneity of the groups, and thus the quality of a given split, is commonly calculated by the Gini Impurity or the Cross-Entropy (also known as Information Gain). These measures assess the "purity" of the resulting groups after a split, guiding the algorithm to select the most informative feature and threshold at each step (GÉRON, 2017). The Gini Index (G) and Cross-Entropy (D) are calculated as follows (JAMES, 2013):

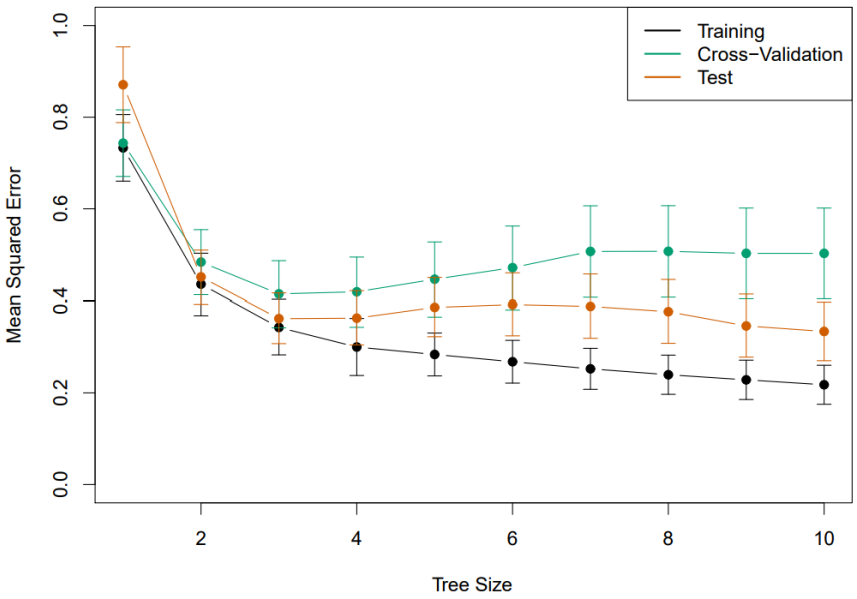
$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}); \;\; D = -\sum_{k=1}^{K} \hat{p}_{mk} log(\hat{p}_{mk})$$

Where $\hat{p}_{mk}$ represents the proportion of training observations in the mth region that belong to the kth class.

After formally defining how splits are optimally made, it's also important to explore how the depth of trees should be carefully chosen. To start exploring the optimal depth of trees, it is helpful to visualize how error rates behave in a real-world scenario, and that is shown in Figure 13. While too small tree sizes (like 1 in this case) might be not enough to create relevant segmentations to build meaningful predictions, too large tree sizes (like 10 in this given example) are also not optimal. Increasing depth eventually reduces the number of observations within each

group and creates arbitrary rules that do not capture meaningful behaviors on the ground truth data, resulting in overfitting.

Figure 13: Distribution of Error Rates by Tree Size



Source: Extracted from (JAMES, 2013)

While Decision Trees are great for their interpretability and simple structure, they often lack the robustness required to generate reasonable performance in modern problems. Despite leveraging Decision Trees as their fundamental building blocks, however, Random Forests are widely recognized as one of the most powerful Machine Learning algorithms available (GÉRON, 2017).

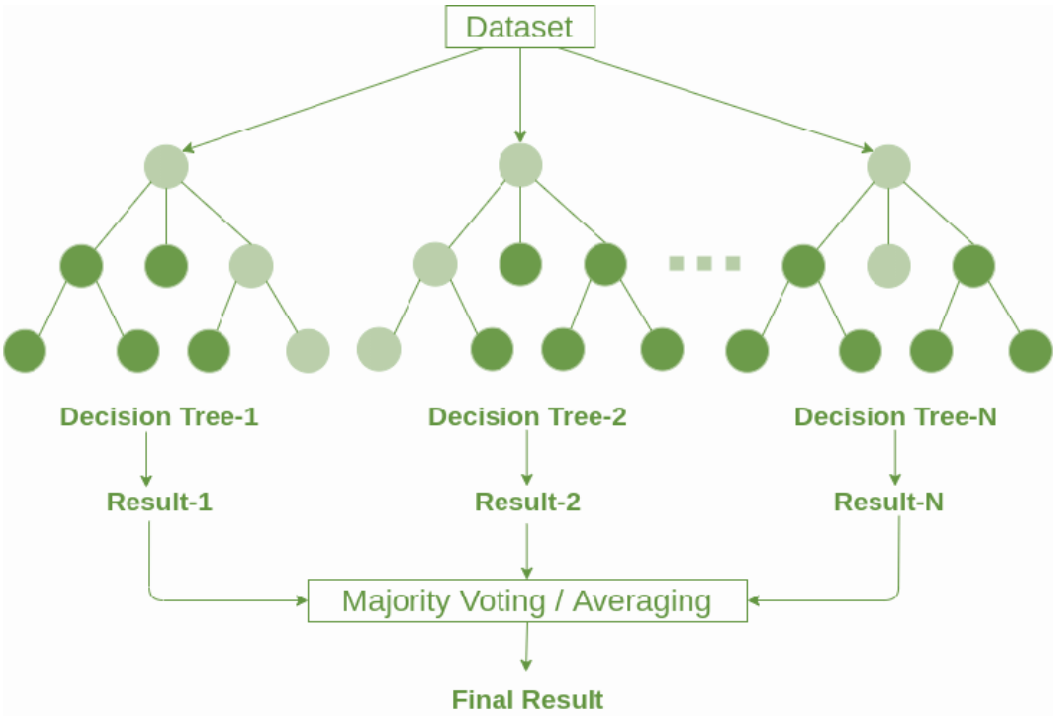
Random Forests are able to achieve such performance improvement by aggregating the outcomes of a large number of decision trees, using two key mechanisms: Bagging and Random Feature Selection (JAMES, 2013).

Bagging (also known as Bootstrap Aggregation) is a procedure for reducing the variance of a statistical learning model. Bagging begins by taking repeated samples from the original training data set, which will be used to create many different training data sets from the population

(a technique called Bootstrap). A new model is specifically trained for every set of Bootstrapped training sets, and the final prediction is done by averaging the resulting predictions (JAMES, 2013).

On top of Bagging, Random Forests make use of Random Feature Selection. Random Feature Selection consists of forcing each new split to consider only a small subset of m random predictors out of all p available. Given that the threes are now constructed over different subsets of features, they are much more decorrelated. Decorrelating the threes is an important step to ensure Bragging is able to efficiently reduce the resulting variance, and thus critical for model performance (JAMES, 2013). Figure 14 shows a visualization of the Random Trees classifier.

Figure 14: Visualization of the Random Trees classifier



Source: Extracted from (KOLAMBAGE & HEWAPATHIRANA, 2020)

2.10 XGBoost

After discussing the fundamentals behind Logistic Regression, K-Nearest Neighbors and Random Forests, one final model worth exploring is the XGBoost model. The XGBoost (or eXtreme Gradient Boosting) is a machine learning algorithm that builds on the concept of ensemble methods, similar to random forests, but with a distinct focus on sequential improvement (CHEN & GUESTRIN, 2016). While random forests create many independent decision trees in parallel and aggregate their predictions, XGBoost trains decision trees sequentially, where each new tree corrects the errors of the previous ones. This iterative process allows the model to focus on the hardest-to-predict data points, progressively improving its overall performance.

The algorithm is grounded in gradient boosting, where trees are added in a manner that minimizes a specified loss function, akin to the optimization process in logistic regression. This ensures that each subsequent tree refines the residuals—essentially, the mistakes—of the preceding trees (CHEN & GUESTRIN, 2016). What sets XGBoost apart is its efficiency and scalability, achieved through techniques like handling missing data, built-in regularization to prevent overfitting, and optimized parallelization. These enhancements make it particularly suited for structured data and have cemented its reputation as a top choice in machine learning competitions.

2.11 Model Evaluation Metrics

Evaluating the performance of predictive models is a critical aspect of statistical learning, as it ensures that models meet the requirements for accuracy, reliability, and relevance to the problem at hand. Different metrics are used to evaluate models depending on the context, especially for classification tasks. This chapter discusses some of the most commonly used model evaluation metrics: Accuracy, Precision, Recall, F1 Score, and the Area Under the Receiver Operating Characteristic Curve (AUC). All metrics will be defined and their significance discussed.

Accuracy is one of the simplest and most widely used evaluation metrics. It is defined as the proportion of correctly predicted instances out of the total number of instances (SAMMUT, 2010). Mathematically, accuracy is expressed as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where True Positives (TP) represent outcomes in which the data is true and the model correctly identifies it as true, True Negatives (TN) represent outcomes in which the data is false and the model correctly identifies it as false, False Positives (FP) represent outcomes in which the data is false but the model incorrectly identifies it as true, and False Negatives (FN) represent outcomes in which the data is true but the model incorrectly identifies it as false. Accuracy provides an overall measure of correctness but may not be suitable for imbalanced datasets, where a model can achieve high accuracy by simply predicting the majority class.

Precision, also known as positive predictive value, measures the proportion of true positive predictions among all positive predictions (SAMMUT, 2010). It is calculated as:

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

This metric is especially useful when false positives are costly, as in spam detection or medical diagnostics. Precision emphasizes the reliability of positive predictions.

Recall, or sensitivity, is the proportion of actual positive instances correctly identified by the model (SAMMUT, 2010). Its formula is:

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

Recall is vital in scenarios where false negatives carry severe consequences, such as in disease screening. It ensures that the model captures as many positives as possible, even if it results in some false positives.

Optimizing for one metric—such as accuracy, precision, or recall—often comes at the expense of another, depending on the nature of the problem and the chosen model. For instance, increasing recall (the ability to identify as many true positives as possible) may lead to a higher number of false positives, which in turn reduces precision (SAMMUT, 2010). Similarly, focusing on precision to minimize false positives can result in lower recall, as some true positives may be missed. This trade-off highlights why no single metric can universally define model performance. Different applications require optimizing different objectives; for example, a medical diagnostic model might prioritize recall to avoid missing critical cases, whereas a spam detection system might focus on precision to reduce false alarms.

One effective way to study these balances and trade-offs is through the confusion matrix, a table that summarizes the performance of a classification model (SAMMUT, 2010).

Table 1: Confusion Matrix

		Assigned Class	
		Positive	Negative
Actual Class	Positive	TP	FN
	Negative	FP	TN

Source: Extracted from (SAMMUT, 2010)

By providing a detailed breakdown of prediction outcomes, the confusion matrix allows practitioners to visualize how the model's decisions impact metrics like accuracy, precision, and recall. It explicitly shows the trade-offs: reducing false positives (increasing precision) may increase false negatives (decreasing recall), and vice versa. This clarity makes the confusion matrix an indispensable tool for understanding and interpreting a model's strengths and weaknesses in the context of its intended application.

There is one measure that aims to combine the balance between precision and recall, called F1 score. The F1 score combines precision and recall into a single metric by calculating their harmonic mean (SAMMUT, 2010):

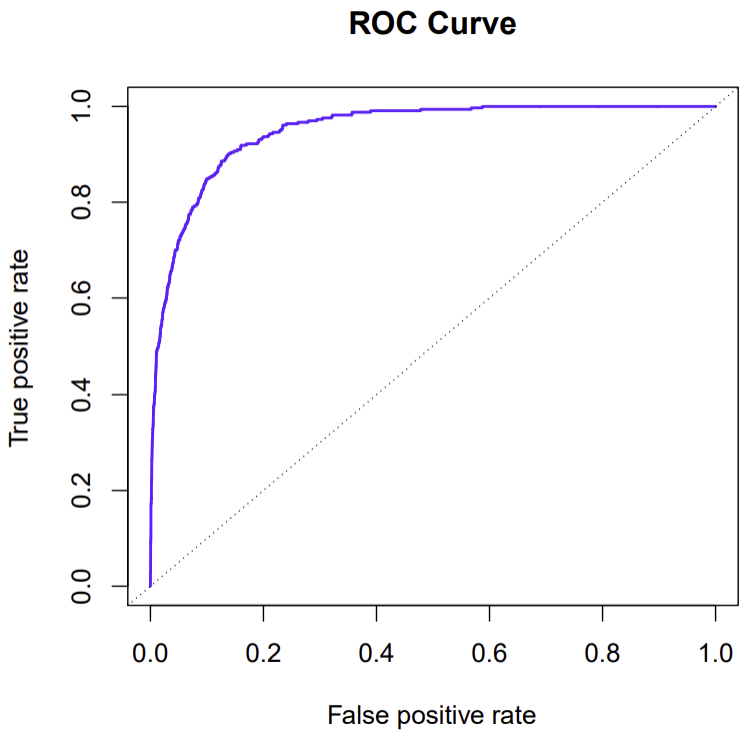
$$F1 = 2 * \frac{P * R}{P + R}$$

This score is particularly useful when there is a need to balance precision and recall, such as in binary classification problems with imbalanced datasets.

Finally, the last model evaluation metric important to define is the Area Under the Receiver Operating Characteristic Curve (AUC-ROC or AUC). The ROC (Receiver Operating Characteristic) curve is a graphical representation of a classifier's performance across various threshold settings, plotting the True Positive Rate (sensitivity) against the False Positive Rate (1-specificity). It is particularly useful for evaluating the balance between false positives and false negatives, helping to select the optimal threshold for classification tasks. The Area Under the Curve (AUC-ROC) quantifies the overall ability of the classifier to distinguish between classes; a

larger AUC indicates a better classifier, as it demonstrates higher sensitivity with lower false positive rates. An AUC-ROC of 1 represents a perfect classifier, while an AUC-ROC of 0.5 indicates random guessing. If the AUC-ROC is below 0.5, it often suggests a problem with the model, such as inverted predictions or data issues. A Sample of an ROC curve is shown in Figure 9.

Figure 15: Sample of ROC Curve



Source: Extracted from (JAMES, 2013)

Since the AUC-ROC is simply defined as the area under the ROC curve, it is calculated as the integral of the True Positive Rate (TPR) with respect to the False Positive Rate (FPR). Mathematically, it is expressed as (POLO & MIOT, 2020):

$$AUC = \int_0^1 TPR(x)d(FPR(x))$$

2.12 Cross-Validation

Beyond selecting the appropriate Model Evaluation Metrics and keeping in mind their meaning and limitations, modern problem settings often require more robust approaches than measuring performance over a single training and testing split of the data. With that in mind, Cross-Validation techniques are widely employed in Machine Learning problems to ensure models are not overfitting, but rather learning the important behaviors of the underlying data (JAMES, 2013).

Cross-Validation involves partitioning the dataset into multiple subsets (or "folds"), with the model being trained on a combination of these folds and validated on the remaining fold. The validation fold is then cycled through all subsets, with the model being re-trained and re-evaluated for each configuration (JAMES, 2013). The most common approach to Cross-Validation is the K-Fold Cross Validation, in which the data is divided into K equally sized folds. For better understanding of the K-Fold Cross Validation technique, a particular setting for when K=10 is shown in Figure 16. This figure shows how individual error values E_i are calculated for each of the $i = 1, 2, \dots, 10$ iterations, with the resulting error metric E being later calculated as the simple average across all individual error values.

Figure 16: Diagram of K-Fold Cross-Validation for K=10

Source: Extracted from (ROSAEN, 2016)

One important decision during the K-Fold Cross Validation technique is the decision for K. In practical applications, computational efficiency considerations should be considered, with high values of K potentially making the Cross-Validation step unfeasible if the model being explored is computationally intensive to train (JAMES, 2013). Additionally, Bias-Variance Trade-Off considerations also come into play, with higher K values resulting in lower bias by using more training data per fold but increasing variance due to smaller test sets. All factors considered, and empirical results suggest that K=5 or K=10 generally yield optimal results (JAMES, 2013).

2.13 Medical and Ethical considerations for self-service tools

The integration of digital health technologies in cardiovascular risk assessment requires careful consideration of both medical validity and ethical implications. Research has demonstrated that self-service cardiovascular risk assessment tools can provide results comparable to standard clinical methods when properly implemented (BARROSO, 2018). Studies have shown high clinical performance in ruling out intermediate or high cardiovascular risk, with particularly strong negative predictive values, indicating these tools can effectively identify individuals who don't require immediate clinical intervention (BARROSO, 2018).

A fundamental ethical consideration is the role of these tools as complementary resources rather than replacements for clinical judgment. Digital health tools should be positioned to empower individuals while maintaining the essential doctor-patient relationship (CAIANI, 2020). This complementary approach is particularly valuable for reaching individuals with multiple elevated CVD risk factors who might benefit from early intervention, while interventions are still viable (NEUFINGERL, 2014).

Data quality and accuracy represent critical medical considerations. Research has shown that missing or inaccurate information in self-assessment tools can lead to significant variations in risk calculations, with studies indicating risk overestimation by 2.1-4.5 years in heart age calculations when physiological risk factors are unknown (NEUFINGERL, 2014). This

underscores the importance of educating users about the tools' limitations and the necessity of professional medical validation (SHORE, 2020).

The ethical implementation of these tools must address several key domains, including access and usability considerations, privacy impacts, comprehensive risk-benefit assessment, and transparent data management practices (SHORE, 2020). To ensure ethical deployment, developers should invest time in building trust and communication channels with communities, partner with community health workers to bridge understanding gaps, and maintain transparent communication about data usage and sharing (SHORE, 2020). A visualization of the core ethical principles for digital health practices is shown in Figure 16, providing a general overview of the important aspects to be taken into consideration while developing and deploying new tools.

Figure 17: Factors influencing ethical practices in digital health

Source: Extracted from (SHORE, 2020)

Healthcare providers must ensure that any recommended digital health tool is safe, effective, and regulated to mitigate potential risks. The integration of these tools should follow

established clinical frameworks and maintain continuous feedback incorporation to ensure alignment with users' needs and lifestyles (SHORE, 2020). This includes regular validation and updates to ensure alignment with current clinical guidelines and best practices (NEUFINGERL, 2014).

A critical ethical consideration is the potential impact on healthcare disparities. While digital health technologies can extend clinical opportunities to historically excluded communities, they may also exacerbate existing disparities (SHORE, 2020). Users need sufficient digital skills and health literacy to properly utilize these tools and understand their limitations (CAIANI, 2020). To address this, community health workers can serve as bridges for helping individuals understand how technologies are used, how data are managed, and who has access (SHORE, 2020).

The development and implementation of self-service CVD risk assessment tools must be guided by established clinical frameworks while protecting participants through human-centered design principles (SHORE, 2020). These tools should incorporate validated risk calculation methodologies, such as the Framingham Risk Score or ASCVD risk calculator, to maintain clinical validity (NEUFINGERL, 2014). Regular validation and updates ensure alignment with current clinical guidelines and best practices, while clear communication about limitations and the complementary nature of these tools helps maintain appropriate expectations and usage (CAIANI, 2020).

3. METHODOLOGY

Building on the rigorous exploration of key concepts in the Literature Review – spanning the nature of CVD-linked mortality rates in Brazil, the theoretical foundation of supervised learning models, and the ethical and medical considerations for developing digital health tools – the Methodology section will outline the methodology designed to address the problem. It is worth reminding that our ultimate goal is building a highly accessible and interpretable machine learning model and make such model available to public benefit through an intuitive interface. To achieve this goal, we must work on top of a high-quality dataset, as well as experiment with a wide range of statistical learning techniques until satisfactory model performance can be achieved.

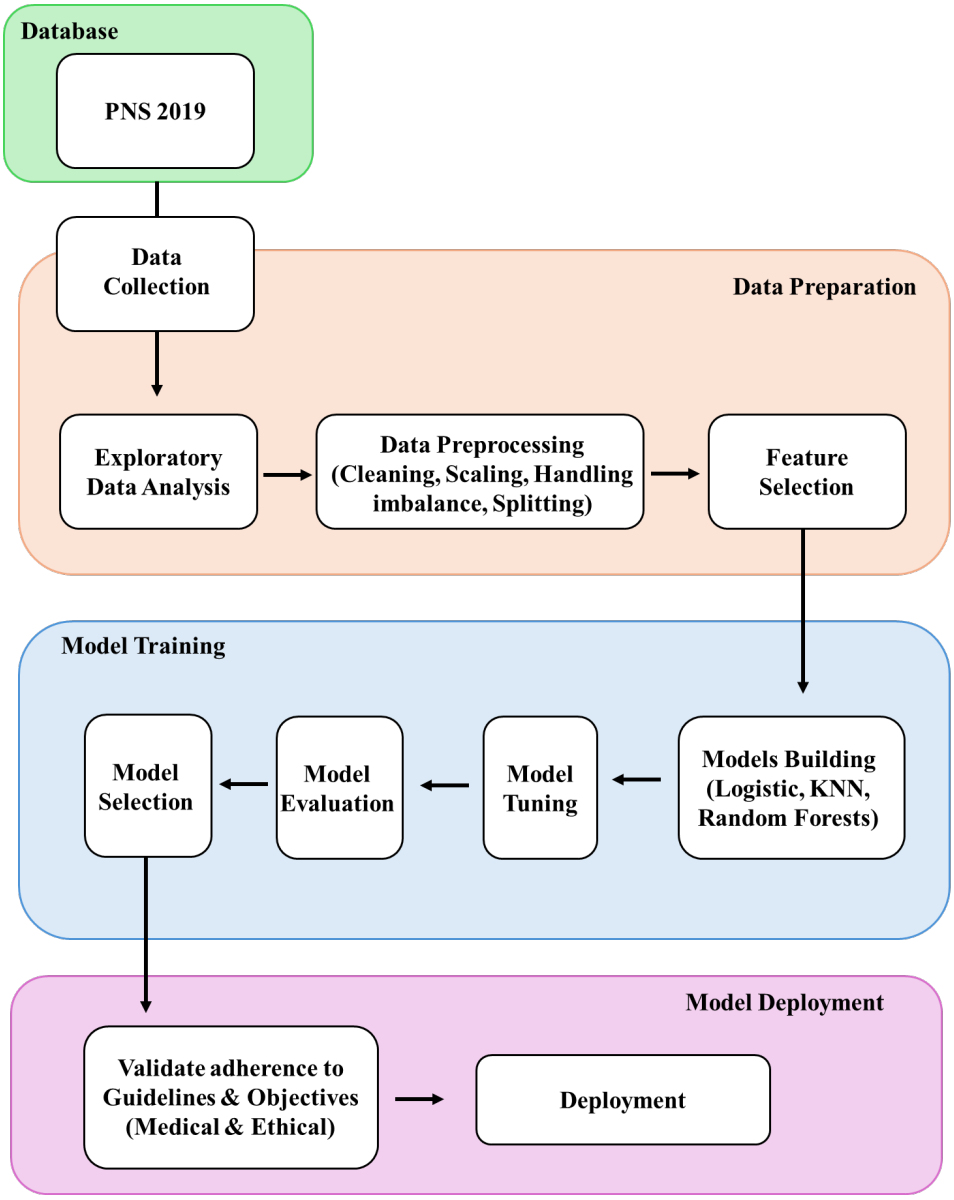
The Methodology section begins by outlining the Research Workflow adopted in this thesis, framing each step from data collection to model deployment in a structured manner. It then describes the Implementation Environment, highlighting the key computational tools used to build and test the models. This is followed by the Data Overview section, covering the dataset decision, its key characteristics, and the methods used to handle the data. Next, the Model Bulilding section dives into how the models were constructed, with a focus on their coding implementation. Finally, a subsection on Deployment explores the final step of releasing the tool for public use.

3.1 Research Workflow

The research workflow for this study was designed to ensure a systematic and rigorous approach, progressing logically from data acquisition until to model deployment. Structured as an interconnected pipeline, each stage builds upon the outcomes of the previous one, with the overarching goal of developing a predictive model for cardiovascular disease prediction that is both accessible and reliable.

This workflow draws inspiration from the Explore-Refine-Produce (ERP) framework proposed by (STOUDT, VÁSQUEZ, & MARTINEZ, 2021), which emphasizes a systematic progression from raw data exploration to actionable research products. While adhering to the ERP principles, specific adaptations were introduced by the Author to ensure compatibility with the unique challenges of CVD prediction, ensuring relevance and applicability. Figure 18 illustrates the composable and structured nature of this process.

Figure 18 Research Workflow Chart



Source: The author

The starting point of this workflow was the acquisition of the PNS 2019 dataset, a reliable and comprehensive source of health-related survey data collected by the Brazilian Institute of Geography and Statistics (IBGE). The PNS 2019 dataset will be later explored in detail under the "Data Overview" section. The collected dataset served as the foundation for the entire research, providing a rich collection of variables relevant to the prediction of CVD. The data collection phase was followed by an exploratory data analysis (EDA), which involved a thorough

examination of the dataset to understand variable distributions, identify missing or anomalous values, and uncover potential relationships among features. This step was critical for gaining initial insights into the data and informed subsequent preprocessing decisions.

The next phase, data preparation, was a cornerstone of the workflow. This stage addressed several preprocessing tasks, including cleaning, scaling, and handling class imbalance to ensure the dataset was suitable for machine learning models. Splitting the data into training, validation, and testing subsets was an integral part of this phase, allowing for robust model evaluation while minimizing the risk of overfitting. Feature selection techniques were employed to identify the most relevant variables, enhancing the efficiency and interpretability of the predictive models.

Model training marked the transition from data preparation to computational modeling. Three machine learning algorithms—logistic regression, K-nearest neighbors (KNN), and random forest—were selected based on their suitability for binary classification tasks and their balance between interpretability and predictive power. The training process involved iterative hyperparameter tuning to optimize model performance, with each iteration evaluated based on predefined metrics such as recall, precision, and the area under the ROC curve (AUC-ROC). Recall was prioritized due to its significance in identifying potential CVD cases, aligning with the overarching goal of early intervention.

Model evaluation played a critical role in validating the performance of the trained models. This phase was conducted iteratively, with feedback loops that informed refinements to the model tuning process. Each model's strengths and limitations were analyzed, ensuring that the final model achieved a balance between accuracy and recall. This iterative approach underscored the importance of continuous improvement in machine learning workflows, particularly for applications with significant public health implications.

The final stage of the workflow focused on model deployment, a crucial step in translating the research outcomes into practical applications. The deployment process involved integrating the predictive model into a web application using Anvil, a platform that simplified the transition from development to a user-friendly interface. This deployment ensured that the model could be

accessed by the general Brazilian population, reflecting the study's commitment to societal impact and accessibility. Validation steps were also incorporated into this phase to ensure adherence to ethical and medical guidelines, further solidifying the credibility and reliability of the final product.

Overall, this workflow provided a clear and structured pathway for achieving the research objectives. Its systematic design ensured that each stage was rigorously executed, with feedback loops promoting continuous improvement and adaptability. By following this comprehensive workflow, the research not only delivered a predictive model tailored to the dataset but also demonstrated a commitment to methodological rigor and practical applicability.

3.2 Implementation Environment

The implementation of this research relied on a robust computational stack, carefully selected to ensure efficiency, reproducibility, and accessibility throughout the data preprocessing, modeling, and deployment phases. Each tool used in this stack was chosen for its specific capabilities, open-source nature, and the community-driven innovation that accompanies such technologies. These tools facilitated the seamless handling of the dataset and the development of predictive models, while also enabling the deployment of the final model to make it accessible to the general Brazilian population. Each tool will be described in detailed, with a summary of all tools being shown in Table 2.

Python was the primary programming language used for all tasks in this research, from data preprocessing to model deployment. Its extensive ecosystem of libraries and widespread adoption in the machine learning community made it the ideal choice for this project. Developed by Guido van Rossum, Python has become one of the most versatile and accessible programming languages, with a strong emphasis on simplicity and readability. Its open-source nature has fostered a vibrant global community, ensuring continuous improvements and extensive documentation, which were instrumental in achieving the research objectives.

For data manipulation and preprocessing, Pandas and NumPy provided the foundational tools necessary to prepare the dataset for analysis. Pandas, created by Wes McKinney, offered robust support for handling structured data, including the ability to clean, transform, and analyze large datasets efficiently. NumPy, developed under the leadership of Travis Oliphant, facilitated high-performance numerical computations, particularly in handling multidimensional arrays and matrix operations. Together, these libraries formed a highly efficient and flexible framework for processing the PNS 2019 dataset, which was critical to ensuring data quality and consistency before modeling.

The machine learning models were implemented using scikit-learn, a widely respected open-source library developed by David Cournapeau and contributors from the French Institute for Research in Computer Science and Automation (INRIA). Scikit-learn was selected for its user-

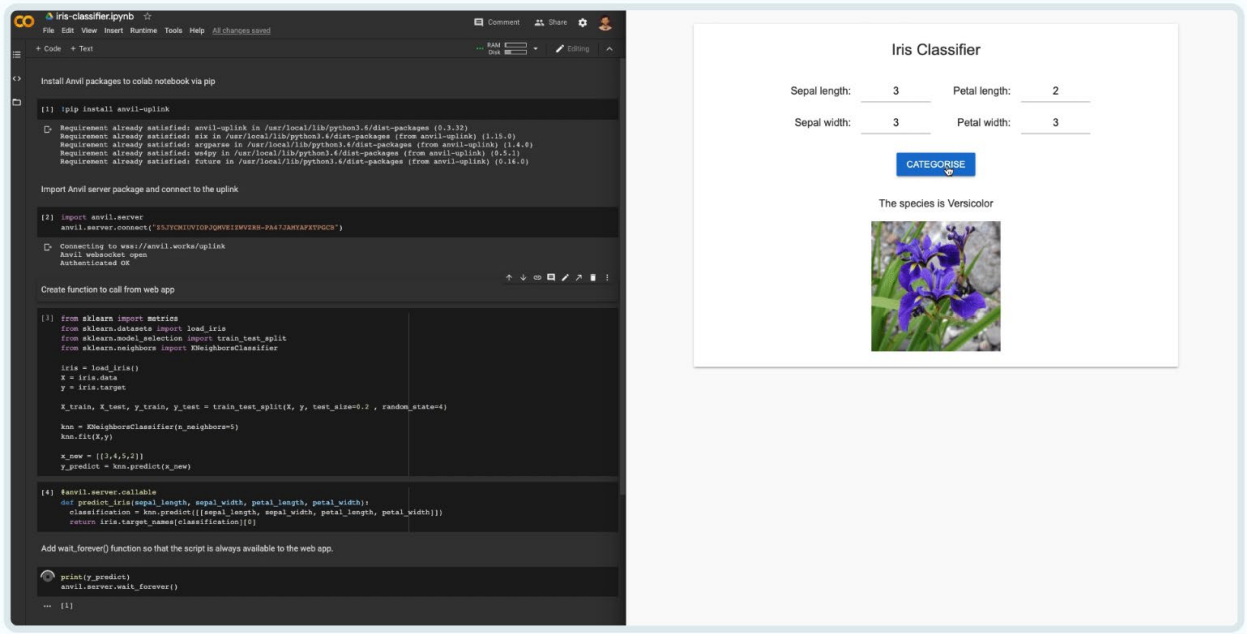
friendly API, extensive suite of algorithms, and strong integration with Python's data manipulation libraries. This tool enabled the development of logistic regression, K-nearest neighbors, and random forest models while simplifying the tasks of training, hyperparameter tuning, and performance evaluation. The open-source nature of scikit-learn ensured reliability and transparency, as its algorithms are rigorously validated by the scientific community.

Data visualization, an integral part of both exploratory data analysis and result presentation, was performed using Matplotlib and Seaborn. Matplotlib, initially developed by John D. Hunter, provided low-level control for creating customized visualizations, while Seaborn, built on top of Matplotlib by Michael Waskom, offered high-level abstractions for statistical data visualization. These tools allowed the generation of clear and informative visual representations of the dataset and model performance, aiding in deriving insights and effectively communicating findings.

To support the computational demands of training machine learning models on a large dataset, Google Colab was used as the coding environment. This cloud-based Jupyter Notebook service, developed by Google Research, provided free access to pre-configured libraries and hardware accelerators, including GPUs, which significantly enhanced the efficiency of computational tasks. Google Colab's integration with Python and its collaborative features also ensured a streamlined workflow and reproducibility, both of which are essential in academic research.

To make the deployment of the predictive model accessible to the general Brazilian population, Anvil was utilized (BRITNELL, n.d.). Anvil is a platform designed to simplify the process of deploying Python-based applications to the web. By integrating seamlessly with models built on Google Colab, Anvil allowed the creation of a user-friendly web interface for the predictive model, ensuring that it could be easily accessed and used by non-technical individuals. This deployment step reflects the research's commitment to translating technical outcomes into real-world impact, particularly for public health use cases in Brazil. A visualization of how Anvil works jointly with Colab is shown in Figure 18.

Figure 19: Avil Web-based Interface making it easy to interact with models hosted on Colab



Source: Extracted from (BRITNELL, n.d.)

By relying on an open-source, community-driven computational stack, this research not only ensured methodological rigor but also aligned with the principles of accessibility and transparency that are fundamental to academic inquiry. These tools collectively provided a powerful foundation for addressing the complexities of cardiovascular disease prediction, while the deployment via Anvil exemplified the broader goal of making research findings accessible and actionable for the general population.

Table 2: Summary of the Implementation Environment

Tool	Developer	Purpose	Rationale	Relevance to Research
Python	Guido van Rossum	Primary programming language for data preprocessing, modeling, and deployment.	Open-source, versatile, widely adopted, and supported by a large community.	Provided the foundation for integrating various libraries and ensured simplicity and reproducibility.
Pandas	Wes McKinney	Data manipulation and preprocessing of the BRFSS 2015 dataset.	Robust handling of structured data, open-source, and extensively documented.	Enabled efficient cleaning, transformation, and preparation of data for modeling.
NumPy	Travis Oliphant and contributors	Numerical computation and array manipulation.	High-performance operations on multidimensional arrays, essential for data transformations.	Simplified handling of numerical data during preprocessing and feature engineering.
scikit-learn	David Cournapeau and INRIA contributors	Development and training of machine learning models (logistic regression, KNN, random forest).	Comprehensive machine learning library with user-friendly API and validated algorithms.	Simplified model implementation, hyperparameter tuning, and performance evaluation.
Matplotlib	John D. Hunter	Data visualization for exploratory analysis and results presentation.	Provides granular control over custom visualizations, open-source, and widely used.	Enabled clear graphical representation of feature distributions and model performance.
Seaborn	Michael Waskom	High-level statistical data visualization.	Built on Matplotlib, simplifies the creation of complex visualizations with concise syntax.	Enhanced the clarity and interpretability of exploratory and statistical insights.
Google Colab	Google Research	Cloud-based coding environment for development and training of machine learning models.	Free access to pre-configured libraries and hardware accelerators like GPUs, ensuring computational efficiency.	Facilitated seamless execution of computationally intensive tasks, promoting reproducibility and collaboration.
Anvil	Anvil Works	Deployment platform to make the predictive model available as a web application.	Simplifies integration of Python-based models with user-friendly web interfaces for non-technical users.	Enabled accessible deployment of the predictive model to the general Brazilian population.

Source: The Author.

3.3 Data Analysis

The selection of a high-quality dataset tailored to the desired objectives is a foundational step in predictive modeling and data analysis. The quality, relevance, and structure of the dataset directly influence the accuracy, generalizability, and utility of the models developed. In the context of predicting cardiovascular disease (CVD), a dataset that captures a diverse range of health-related variables across a representative population is essential for ensuring robust and actionable insights.

The dataset used for this Thesis is the 2019 National Health Survey (PNS), or *"Pesquisa Nacional de Saúde"*. The PNS is a nationally representative survey conducted in Brazil that collects detailed information on health conditions, lifestyle factors, and healthcare utilization, being conducted by Brazil's Health Ministry and the Brazilian Institute of Geography and Statistics (IBGE) (MS, 2021). This dataset is particularly well-suited for the analysis of cardiovascular disease risk due to its comprehensive coverage of factors known to influence CVD, such as demographic attributes, behavioral patterns, pre-existing conditions, and access to healthcare.

In particular, the key characteristics of the PNS 2019 that make it a compelling candidate to our modelling purposes are its extensive scope, rich feature set, national representation and validated data collection. In Total, the PNS 2019 contains survey data on almost 300,000 individual observations with more than 1,000 features. These characteristics will be further explored on the Exploratory Data Analysis section.

Additionally, for a tool like the one being proposed under the current Thesis, transparency and reproducibility are essential. The PNS 2019 dataset is not only available to public use and can be accessed directly from the official website dedicated to the National Health Survey (PNS, 2021), but its methodology and data dictionary are also made available by the PNS initiative and easy to interpret. Such transparency will be crucial given the complexity of the dataset, helping us investigate features and fine tune the models throughout the process.

3.3.1 Exploratory Data Analysis

The Exploratory Data Analysis (EDA) is a critical step in understanding the nature and structure of a dataset, enabling the identification of patterns, relationships, and irregularities that influence subsequent modeling steps. EDA will provide a comprehensive overview of the PNS 2019 dataset, guiding decisions on cleaning, transformation, and feature selection while ensuring the data is optimized for predictive analysis. This phase focuses on examining the distributions of variables, detecting outliers, and identifying missing values or inconsistencies that require preprocessing.

The first thing to take note is the format in which data is initially structured, and whether it will require any transformation before further visualization. We can do this by loading the dataset into the Google Colab environment and creating a Pandas dataframe object to store the data in a structured and flexible object. Through Pandas native “info()” function, we observe the dataset is composed of 1,078 float64 columns and 9 int64 columns (1,087 available features in total), for a total of 293,725 unique observations. Even though a dataset of this complexity might look overwhelming at first, the robust and flexible implementation environment will allow for careful investigation of the meaning behind the data.

We proceed by noticing that PNS 2019 survey responses are structured in sections, named as Modules. Each Module is defined by a common response topic, ranging from standard identification and control data (such as the Brazilian Federated Unit in which the data was collected) to lifestyles or chronic diseases data. Conveniently, the features in the dataset also come identified by their module, which will help us bridge our domain knowledge of the problem built during the Literature Review section to narrowing our focus to the categories most likely to matter for our problem. Table 3 summarizes how features are distributed across categories, and it’s worth noticing how not all categories contribute evenly.

Table 3: Features categorized by Module

Module	Description	Feature Count
1	Identification and Control	12
A	Household information	43
B	Home visits by the Family Health Team and Endemic Agents	4
C	General characteristics of residents	20
D	Educational characteristics of individuals aged 5 years or older	18
E	Work of household residents	59
F	Household income	7
G	Individuals with disabilities	50
I	Health insurance coverage	14
J	Utilization of health services	65
K	Health of individuals aged 60 or older (…)	37
L	Children under 2 years	39
M	Work characteristics and social support	27
N	Perception of health status	16
O	Accidents	24
P	Lifestyles	146
Q	Chronic diseases	236
R	Women's health	45
S	Prenatal care	73
U	Oral health	19
Z	Paternity and partner prenatal care	17
V	Violence	45
T	Communicable diseases	11
Y	Sexual activity	8
H	Medical care	30
W	Anthropometry	7
	Others	15
	Total	**1087**

Source: The author

On top of that, careful inspection of the data dictionary for all features has shown many are framed as Yes/No questions, where 1 stands for Yes and 2 stands for No. By running code to aggregate feature count into similar types, it was possible to count that 493 features in total carry binary meaning (despite still being stored as float64 values), while the remaining features were either of small range categorical meaning or true continuous variables. Table 4 shows the summary of feature types, including examples. Given that features vary through significantly different ranges of values, rescaling will be needed to ensure distance-sensitive methods like KNN work properly.

Table 4: Feature Types

		Feature Example		
Category	**Count**	**Code**	**Description**	**Scale**
Binary	493	I00102	Do you have health insurance?	Yes/No
Small Range	433	N001	In general, how would you rate your health?	From 1=Very Good to 5=Very Bad
Medium or Wide Range	161	P00104	What is your weight?	From 1 to 599 in kg

Source: The Author

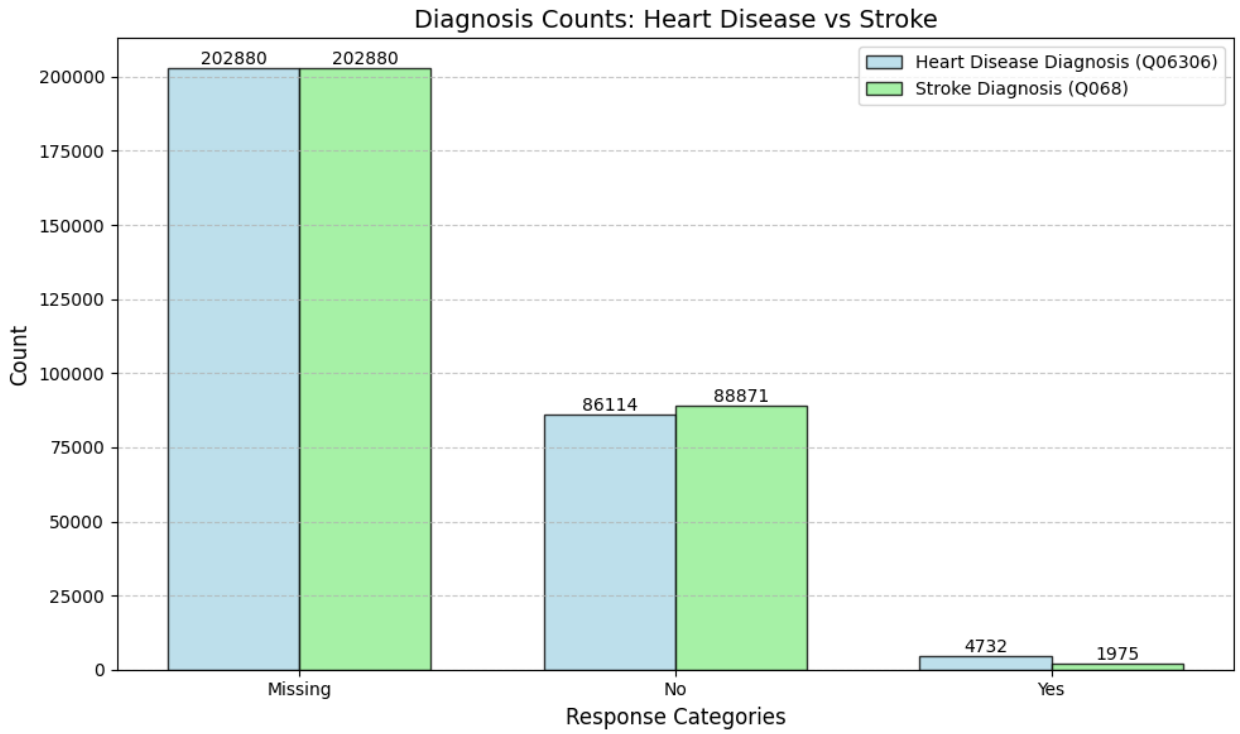
After building the initial understanding of how features are categorized within the dataset, it was possible to identify the candidates for the Target Variable within the dataset. For good modelling outcomes, it is crucial to select a meaningful Target Variable that is truly informative and in line with the model objectives. In our case, the features mostly tied to our problem are 'Q06306' and 'Q068', both within the 'Q' (Chronic Diseases) category. The interpretation for each feature is shown in Figure 20, followed by their respective frequency in the dataset in Figure 21.

Figure 20: Meaning of main CVD-related features

Feature	**Meaning**	**Values**
Q06306	Has a doctor ever diagnosed you with a Heart Disease? (Heart Attack, Angina, Cardiac Insufficiency, Arritmia, or Other)	1=Yes, 2=No, null
Q068	Has a doctor ever diagnosed you with Cerebrovascular Accident (CVA) or Stroke?	1=Yes, 2=No, null

Source: The author, (MS, 2021)

Figure 21: Distribution of responses for CVD-related features



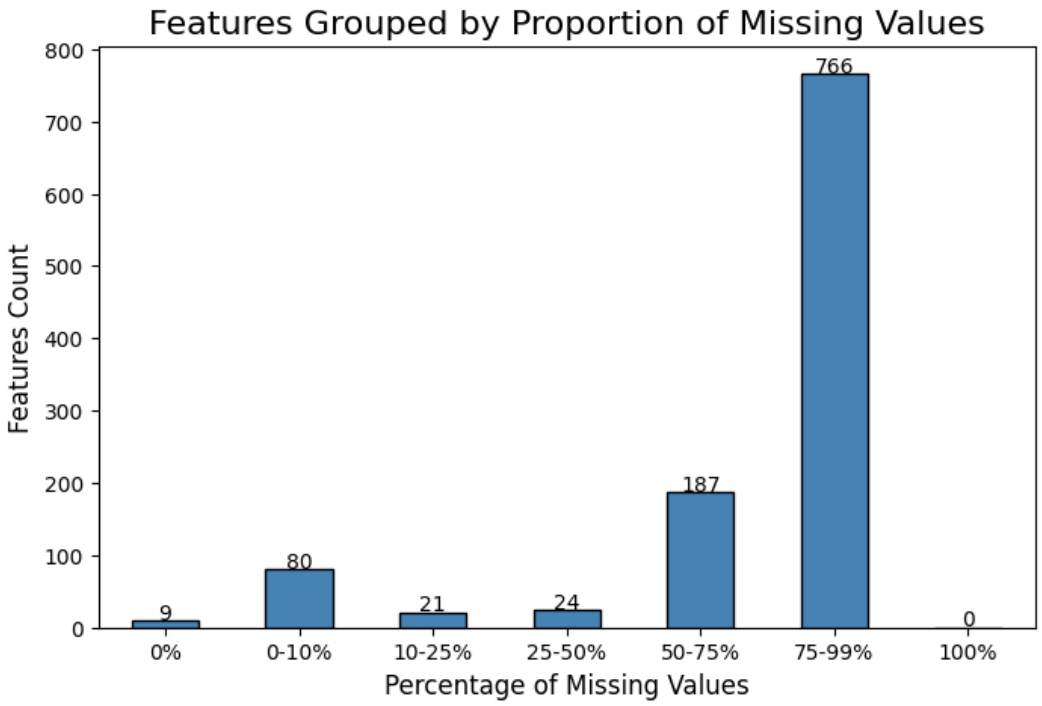
Source: The author

The visualization provided by Figure 21 highlights a significant class imbalance for the features we are interested, which is expected given CVDs only affect a minority of the general population (CASTRO et al, 2019). In our case, only about 7.4% of the non-null responses consist of Yes. This characteristic, however, creates the need for techniques that improve model performance under scenarios of severe class imbalance, which will be explored and implemented later.

Beyond the imbalance issue, Figure 21 also shows a significant presence of missing values. In this specific case, with precisely 202,808 observations containing null responses (69% of the total). This will be crucial to take note, as model performance might be sensitive to the presence of null values, with models like the Logistic Regression or K-Nearest Neighbors not able to natively handle null values. To further investigate the issue of Missingness (percentage of missing values within a feature), Figure 22 plots the count of features within a given Missingness range.

One can notice how while none of the features carry absolute Missingness (null values only), 766 features in total carry more than 75% of missing values.

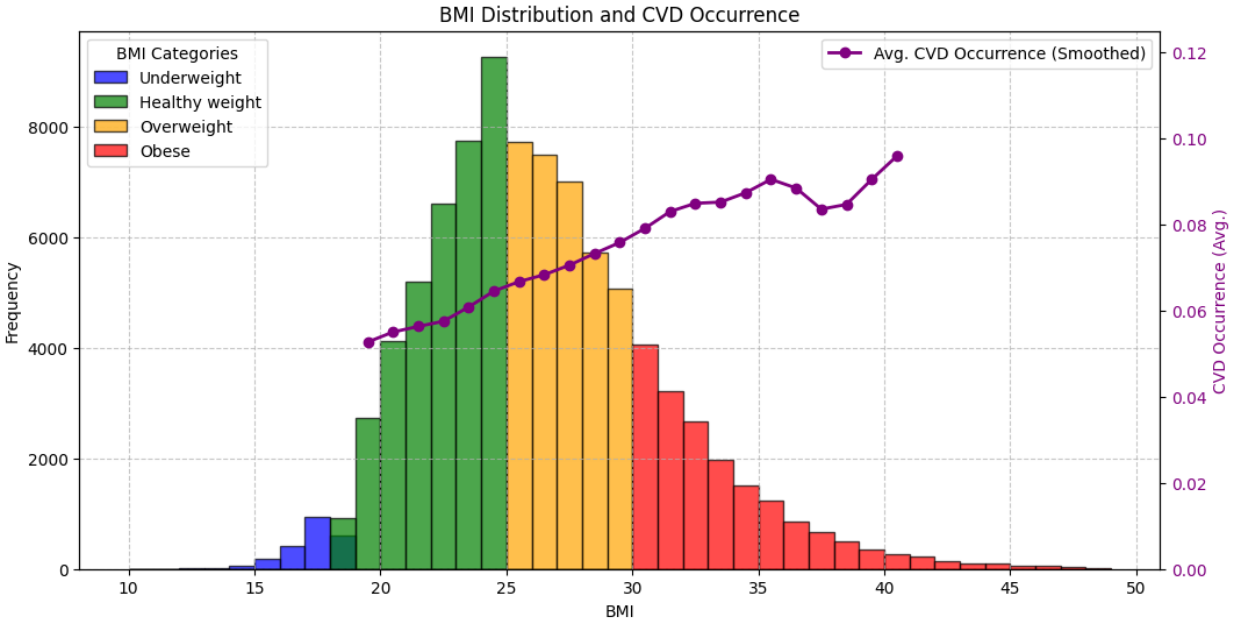
Figure 22: Missingness on the PNS2019 Dataset



Source: The Author

Another useful investigation exercised conducted on the dataset was studying how the Body Max Index (BMI) generally impacts the occurrence of CVDs. As pointed by (BRANDT, 2022) and explored during the Literature Review section, an abnormally high BMI should be tied with higher frequency of CVDs. Figure 23 shows how BMI affects CVD Occurrence rates within the PNS2019 dataset, pointing out to an increasing trend that validates our industry-specific knowledge.

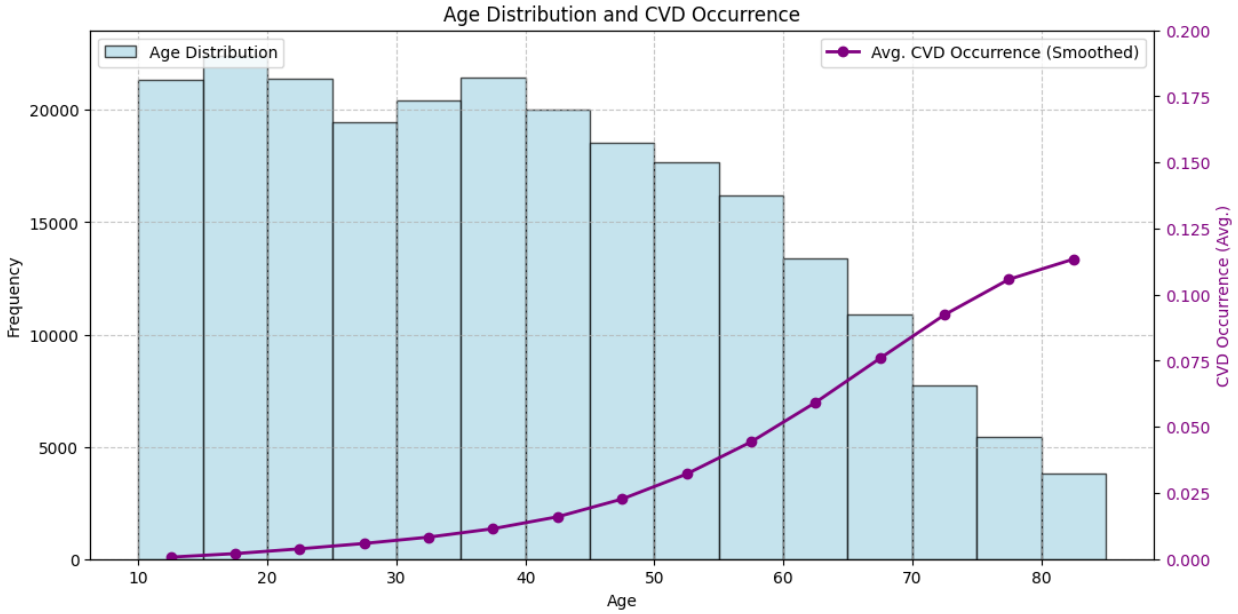
Figure 23: BMI Distribution relative to CVD Occurrence



Source: The author

Similarly to BMI, Age is also one of the core Risk Factors linked to CVD Occurrence. To proceed investigating with the effect of Age within the PNS2019 dataset, Figure 24 was constructed. One can notice, again, that the analytical conclusions go in line with the knowledge built during Literature Review.

Figure 24: Age Distribution relative to CVD Occurrence



Source: The author

3.3.2 Feature Pre-Processing and Selection

After diligently studying the PNS2019 Dataset and the most relevant features, the next stage on our pipeline is to conduct Feature Pre-Processing and Selection. This phase consists of exploring and implementing different feature decisions in order to achieve superior model performance. During this stage, 5 key techniques were applied: (i) min-max scaling, (ii) category-based feature filtering, (iii) missingness threshold-based filtering, (iv) feature engineering, and (v) exclusion of leakage-prone features.

As identified during the EDA stage, features of the PNS2019 dataset have widely different scales, which would compromise performance for methods sensitive to features' absolute values (distance-based methods), like K-Nearest Neighbors. To fix for this issue, simple min-max scaling, as defined in the Literature Review section, was implemented.

The second technique that has proven valuable in the current context after careful experimentation was the category-based feature filtering. Essentially, this step leverages the CVD-specific knowledge built during the Literature Review section, alongside the complex nature of the dataset, and narrows down the features available for training to those that are truly relevant to our problem. After experimenting with a wide range of choices, only the features belonging to 8 key modules have proven to be valuable for predictive purposes, on top of identification features. Those modules were: C (General Characteristics), I (Health Insurance Coverage), J (Utilization of Health Services), P (Lifestyles), Q (Chronic Diseases), U (Oral Health), H (Medical Care) and N (Perception of Health Status). Please refer to Table 3 for the full feature categorization table. This step has reduced the number of available features for training from 1087 to 619.

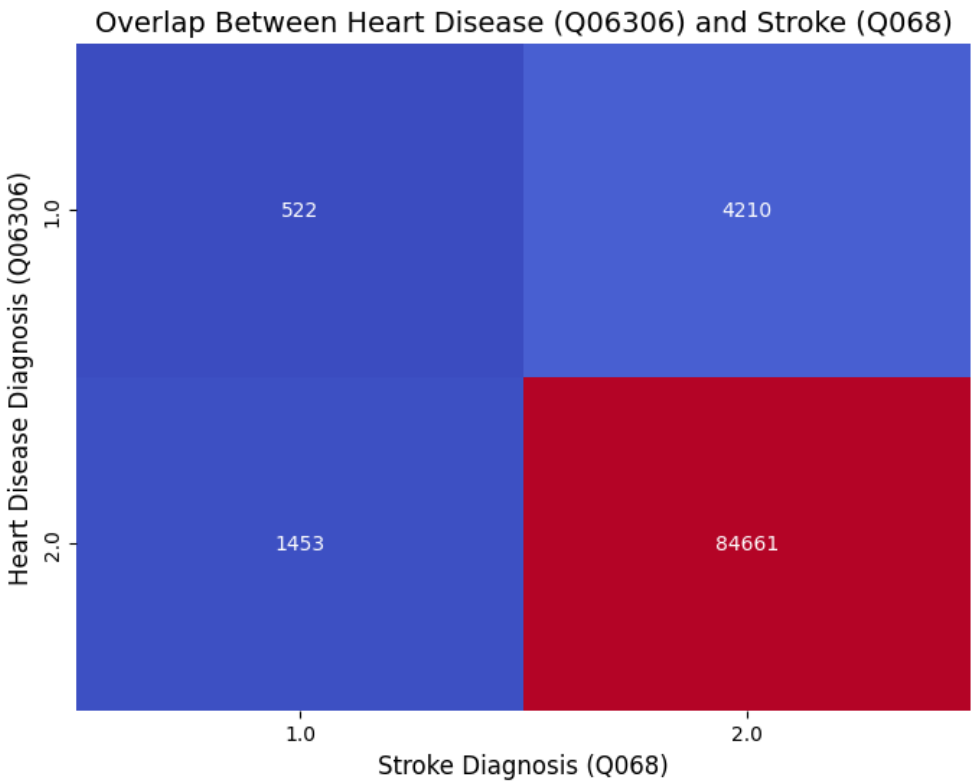
As previously identified during the EDA stage, missingness is extremely relevant on the PNS2019 dataset. To address and create reliability against this issue, several iterations of feature selection based on "missingness threshold" were conducted. The rationale for step is essentially experimenting to which degrees features with too many missing values are actually adding value in model training, instead of simply generating noise. The observed behavior here is that different types of models are able to deal with "missingness" complexity differently, making this tuning

parameter critical and specific to which model. While Logistic Regression models saw fast performance deterioration for thresholds above 60%, Random Forest models saw optimal performance for an 80% threshold. The intuition behind this is simply that Random Forests are better at handling more features and features with more missing values, while Logistic Regression models perform better when working at slightly lower feature counts.

In terms of feature engineering, there were experimentations around aggregating common features and whether this would improve model performance. Interestingly, the IPS2019 dataset does not natively hold any feature dedicated to BMI measuring, but it does collect data for both weight and height of respondents. That said, we experimented with an extra engineered feature to account specifically for BMI, in case this would perform better than features accounting for weight and height in a siloed manner. Our results indicated the additional BMI feature actually improved model performance, and it was added in the features used for model training by the 'P101' code.

Still within feature engineering, another important decision was made. While both individual Heart Disease Occurrence (Q06306) and Stroke Occurrence (Q068) could be useful as our Target Variable for modeling purposes, we used feature engineering to create a new feature, jointly accounting for the risk of CVDs as per a broader definition. This ensures broader applicability of the model, with the new synthetic feature (Q99) incorporating information from both of the previous features, while leveraging the knowledge built during Literature Review that the two disease categories are generally prone to similar risk factors (BRANDT, 2022). For clarification purposes of how Heart Disease Occurrence (Q06306) and Stroke Occurrence (Q068) are intertwined, Figure 25 plots the overlap between both features.

Figure 25: Intersection of Heart Disease and Stroke Occurrence.



Source: The author.

Removing leakage-prone features was a crucial task during the Feature Pre-Processing and Selection stage. Through meticulous inspection of the entire PNS2019 dataset and systematic evaluation of model performance across various scenarios, 20 leakage-prone features were identified. These features contained information closely tied to the target variable, which, if included in the training dataset, could lead models to exploit shortcuts, resulting in artificially high performance (CHOLLET, 2017). For example, such features included 'Q064' (indicating the age at first diagnosis of heart disease) and 'Q06310' (specific to arrhythmia rather than heart disease in general).

One additional process conducted during pre-processing stage was feature imputation for the models that can't natively handle empty values (Logistic Regression, KNNs and Random Forests). To solve for empty values, imputation was our only viable choice, given the relevance of missingness in the PNS2019 dataset (detailed during EDA stage), removing features with empty

values has proven to be an unfeasible approach, leaving too little features and implying non-satisfactory model performance. Nonetheless, different approaches to imputation were experimented with, including mean/median/mode imputation and random imputation. Across our experiments, random imputation has proven to be the most performant method. While random imputation being the most effective method might look surprising at first, this result simply tells our already built intuition about empty values in the PNS2019 – that they are simply missing at random. In this context, empty values don't carry any particular meaning – these are data points that were simply not collected, and with no deeper cause.

3.3.3 Model Training and Hyperparameter Tuning

After preparing the final features, we moved on to Model Training and Hyperparameter Tuning. This phase involves building models and systematically adjusting their parameters to achieve the best performance. The goal is to create a model that generalizes well to unseen data while addressing the specific challenges of the dataset, such as imbalanced classes or missing values.

The first model built was XGBoost, selected for its unique capability to handle missing values natively, as it learns the optimal way to split data even when values are absent. This feature simplifies preprocessing, eliminating the need for explicit imputation. XGBoost is also recognized for its state-of-the-art performance, making it a reliable benchmark for comparing other models. Its ability to efficiently handle large datasets and complex patterns further justified its use.

XGBoost was trained using GridSearchCV, which automates hyperparameter tuning with five-fold cross-validation to ensure consistent results across data splits. We optimized the model using the logloss metric, prioritizing recall to capture as many positive cases as possible, which is critical in imbalanced datasets. Additionally, we experimented with the scale_pos_weight parameter to adjust for class imbalance, finding that a value of 200 significantly improved the detection of minority class instances, providing a strong foundation for further modeling efforts.

The next step was training our Logistic Regression model, using the native classifier from the sklearn library and carefully tuning the parameters. After iterating over several parameters in terms of regularization strength (C), solvers and class weights, the best performing model was achieved with C=10, ‘lbfgs’ solver type and balanced class weight to account for class imbalance. GridSearchCV was used for cross-validation.

For our Random Forests model, we used Scikit-Learn's RandomForestClassifier from the ensemble module, optimizing its hyperparameters with GridSearchCV to improve recall. The hyperparameter grid included the number of estimators (n_estimators), maximum tree depth (max_depth), and class_weight to address class imbalance. A five-fold cross-validation was

applied. The best parameters identified were n_estimators=100, max_depth=10, and class_weight='balanced'. While higher maximum depths were experimented with, they were likely causing the model to overfit, making it interesting to see the bias-variance tradeoffs to occur in real tests.

Finally, the K-Nearest Neighbors model was built. For this last step, we again used Scikit-Learn, this time the KneighborsClassifier. Parameters tuning was done with GridSearchCV, choosing recall as the optimization metric. The hyperparameter grid included the number of neighbors (n_neighbors), the weighting scheme (weights), and the distance metric (metric). Cross-validation with 5 folds and parallel processing ensured an efficient and reliable search. The best parameters identified were n_neighbors=3, weights='distance', and metric='euclidean'. In this case, the distance-weighted voting scheme is essential to handle class imbalance, and n_neighbors=5 demonstrated to be a good choice when considering bias-variance tradeoffs.

4. RESULTS

This section presents the performance evaluation of the machine learning models, highlighting their predictive accuracy, ability to handle class imbalance, and overall suitability for the research objectives. Metrics such as accuracy, precision, recall, F1-score, and AUC were analyzed, alongside visual comparisons of ROC curves, to comprehensively assess the strengths and limitations of each model.

The performance metrics for all optimized models are summarized in Table 5. The Random Forest model achieved the highest AUC (0.8670), underscoring its superior ability to balance true positive and false positive rates. Additionally, it demonstrated robust recall (0.7203) and precision (0.8024), suggesting it effectively captures positive cases while maintaining reliable predictive accuracy. XGBoost closely followed with an AUC of 0.8365, showing strong performance overall but slightly lagging behind Random Forest in terms of recall and precision. Logistic Regression, with an AUC of 0.7557, presented competitive results, particularly given its simplicity and ease of interpretability. In contrast, the KNN model exhibited significant limitations, with the lowest AUC (0.5813) and recall (0.2261), likely reflecting its sensitivity to class imbalance and distance-based predictions in this context.

These results highlight Random Forest as the most reliable model, offering a well-rounded performance across all key metrics. XGBoost served as a valuable benchmark, validating the dataset's predictive potential, while Logistic Regression provided a simpler yet effective alternative. KNN, while intuitive and straightforward, struggled to match the performance of ensemble-based methods, particularly in identifying minority class cases.

Table 5: Summary of Model Evaluation Metrics (Optimized)

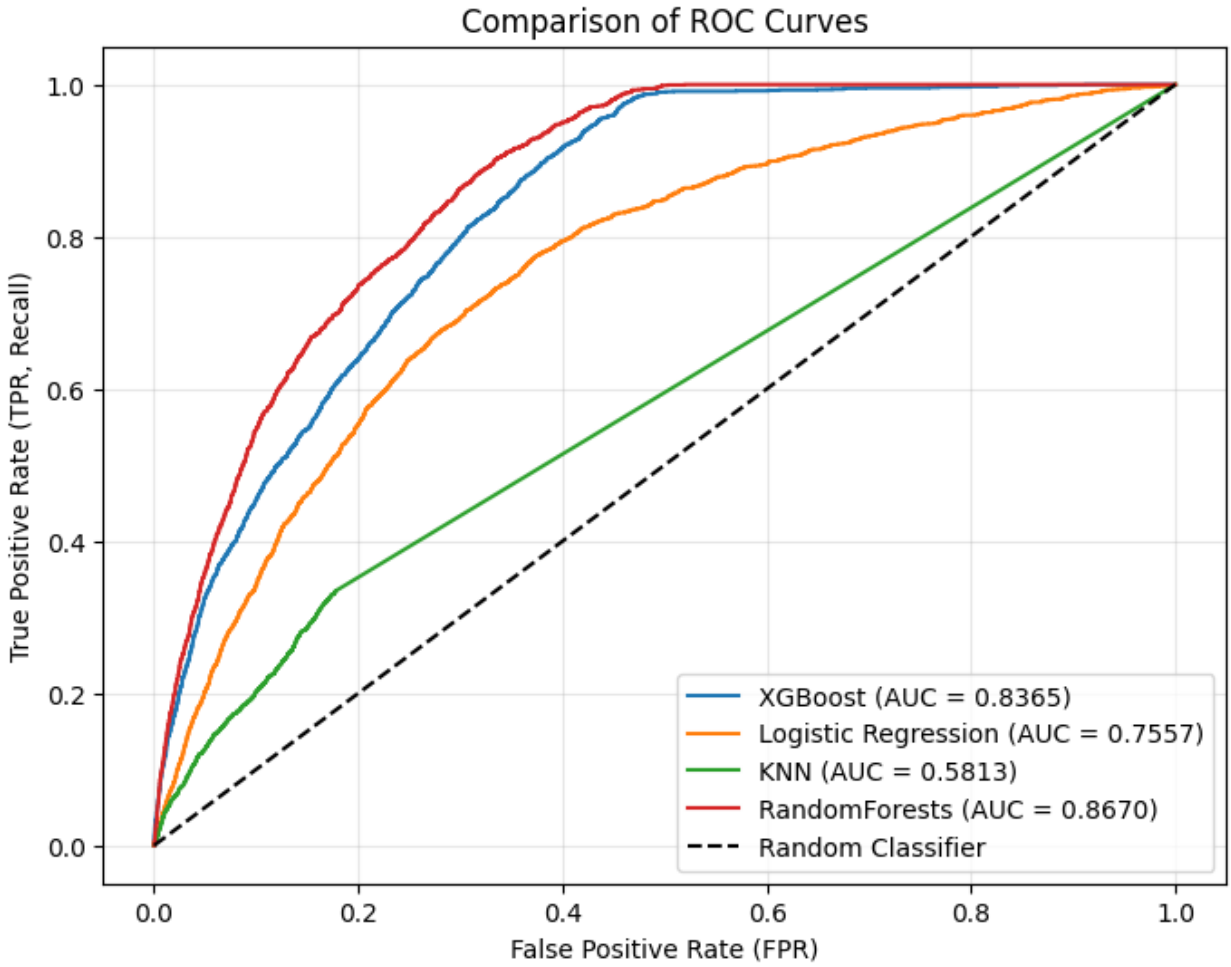
Model Types	AUC	Accuracy	Recall	Precision	F-1
RandomForests	0.8670	0.8024	0.7203	0.2102	0.3254
XGBoost	0.8365	0.7279	0.7590	0.1639	0.2696
LogReg	0.7557	0.6763	0.7225	0.1354	0.2280
KNN	0.5813	0.8391	0.2261	0.1201	0.1569

Source: The Author

The discriminative power of each model is further illustrated in Figure 26, which compares the Receiver Operating Characteristic (ROC) curves for all classifiers (their optimized versions post hyperparameter tuning). The Random Forest and XGBoost models stand out with curves closest to the top-left corner, reflecting their ability to maintain a high true positive rate (recall) while minimizing false positives. Random Forest achieved the steepest ascent, corroborating its leading AUC score and demonstrating its capacity to handle imbalanced data effectively. XGBoost exhibited a similar trajectory, albeit with a slightly reduced steepness, aligning with its marginally lower recall and precision values.

Logistic Regression maintained a solid curve, indicative of its competitive AUC and balanced performance across metrics. Conversely, KNN's ROC curve remained shallow, reflecting its difficulties in distinguishing between positive and negative cases. This underperformance can be attributed to its reliance on local neighbor relationships, which may falter in datasets with imbalanced classes or complex decision boundaries.

Figure 26: Comparison of ROC Curve for all Models (Optimized)



Source: The Author

5. CONCLUSION

This thesis demonstrates the potential of leveraging machine learning to address Brazil's significant public health challenge posed by cardiovascular diseases (CVDs). By developing predictive models tailored to the Brazilian population, the study emphasizes the importance of accessible and interpretable tools for early risk detection and intervention. Utilizing the PNS 2019 dataset, the research explored multiple machine learning algorithms, including logistic regression, K-nearest neighbors, and random forests. These models were rigorously evaluated to ensure high recall, reflecting the prioritization of early detection in healthcare. Through feature engineering and selection, key predictors of CVD risk were identified, bridging statistical modeling with actionable health insights. Additionally, the study addressed class imbalance and optimized performance metrics to ensure the models' reliability and validity.

Beyond the modeling, the deployment of a predictive tool on a web-based platform ensures accessibility for the general population while adhering to ethical guidelines. This tool aligns with Brazil’s public health goals by promoting health equity and informed decision-making. The results underscore the transformative potential of machine learning in public health, offering a framework that balances technical innovation with practical usability while addressing critical issues like data privacy and healthcare disparities. The study not only provides a foundation for CVD risk prediction in Brazil but also serves as a blueprint for leveraging machine learning in other public health challenges.

Despite its contributions, the study is not without limitations. The reliance on self-reported survey data introduces potential biases, and integrating clinical and genetic data could enhance accuracy in future research. Similarly, while the selected models balance interpretability and performance, advanced algorithms such as neural networks may improve predictive capabilities while maintaining usability. Expanding the tool’s reach and integrating user feedback mechanisms can further refine its impact, ensuring it meets the needs of diverse populations across Brazil.

In conclusion, this research illustrates a scalable approach to leveraging machine learning for CVD risk prediction, setting a precedent for similar applications in public health. By combining methodological rigor with societal relevance, the study contributes to reducing CVD mortality in Brazil and lays the groundwork for technology-driven solutions to pressing health challenges.

6. REFERENCES

ARAÚJO, J. M., & RODRIGUES, R. E. (2022). *The direct and indirect costs of cardiovascular diseases in Brazil*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC9778932/

BADAWY, M., & RAMADAN, N. &. (2023). *Healthcare predictive analytics using machine learning and deep learning techniques: a survey*. From https://jesit.springeropen.com/articles/10.1186/s43067-023-00108-y

BARROSO, M. (2018). *Validity of a method for the self-screening of cardiovascular risk*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC5953309/

BERWANGER, O., & SANTO, K. (2022). *Cardiovascular Care in Brazil: Current Status, Challenges, and Opportunities*. From https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.122.059320

BRANDT, L. e. (2022). *Burden of Cardiovascular diseases attributable to risk factors in Brazil: data from the "Global Burden of Disease 2019" study*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC9009428/

BREIMAN, L. (2001). *Random Forests.*

BRITNELL, R. (n.d.). *Turning a Google Colab Notebook into a Web App*. From https://anvil.works/learn/tutorials/google-colab-to-web-app

CAIANI, E. (2020). *Ethics of digital health tools*. From https://www.escardio.org/Journals/E-Journal-of-Cardiology-Practice/Volume-18/ethics-of-digital-health-tools

CASTRO et al. (2019). *Brazil's unified health system: the first 30 years and prospects for the future*. From https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(19)31243-7/abstract

CHAN, J. J. (2022). *Inequalities in the prevalence of cardiovascular disease risk factors in Brazilian slum populations: A cross-sectional study*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC10022010/

CHEN, T., & GUESTRIN, C. (2016). *XGBoost: A Scalable Tree Boosting System*. From https://arxiv.org/abs/1603.02754

CHOLLET, F. (2017). *Deep Learning with Python.*

DEEPA, R. (2024). *Early prediction of cardiovascular disease using machine learning: Unveiling risk factors from health records*. From

https://pubs.aip.org/aip/adv/article/14/3/035049/3279524/Early-prediction-of-cardiovascular-disease-using

FRIDMAN, J., HASTIE, T., & TIBSHIRANI, R. (2008). *The Elements of Statistical Learning: Data Mining, Inference and Prediction.*

GÉRON, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.*

JAMES, G. e. (2013). *An Introduction to Statistical Learning with Applications in R.*

KOLAMBAGE, N., & HEWAPATHIRANA, R. (2020). *Design, Development and Implementation of a Machine Learning-based Predictive Modelling Tool to Accurately Predict Thalassemia Carrier state using Full Blood Count Indices and Haemoglobin Variants*. From https://www.researchgate.net/publication/345813508_Design_Development_and_Implementation_of_a_Machine_Learning-based_Predictive_Modelling_Tool_to_Accurately_Predict_Thalassemia_Carrier_state_using_Full_Blood_Count_Indices_and_Haemoglobin_Variants

KRAUSKOPF, E. (2019). *Cardiovascular disease: The Brazilian research contribution*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC6852459/

MANSUR, A., & FAVARATO, D. (2021). *Cardiovascular and Cancer Death Rates in the Brazilian Population Aged 35 to 74 Years, 1996-2017*. From https://www.scielo.br/j/abc/a/cJzNdtHVN7PxzTg9BhnqWXb/?format=pdf&lang=en

MENSAH, G. e. (2023). *Global Burden of Cardiovascular Diseasesand Risks, 1990-2022*. From https://www.jacc.org/doi/10.1016/j.jacc.2023.11.007

MS. (2021). *Ministério da Saúde (MS). (2021). Pesquisa Nacional de Saúde: 2019: Informações sobre domicílios, acesso e utilização dos serviços de saúde (Vol. 39). IBGE.* From https://www.pns.icict.fiocruz.br/wp-content/uploads/2021/12/liv101846.pdf

NEUFINGERL, N. (2014). *Web-based self-assessment health tools: who are the users and what is the impact of missing input information?* From https://pubmed.ncbi.nlm.nih.gov/25261155/

NWAIMO, S. C. (2024). *Transforming healthcare with data analytics: Predictive models for patient outcomes*. From https://gsconlinepress.com/journals/gscbps/sites/default/files/GSCBPS-2024-0190.pdf

OGUNPOLA, A., SAEED, F., & BASURRA, S. (2024). *Machine Learning-Based Predictive Models for Detection of Cardiovascular Diseases*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC10813849/

PATRIOTA, P. e. (2023). *Reported recommendations to address cardiovascular risk factors differ by socio-economic status in Brazil. Results from the Brazilian National Health Survey 2019*. From https://www.sciencedirect.com/science/article/pii/S2211335523004187

PENG, M., HOU, F., & CHENG, Z. e. (2023). *Prediction of cardiovascular disease risk based on major contributing features*. From https://www.nature.com/articles/s41598-023-31870-8

PNS. (2021). From https://www.pns.icict.fiocruz.br/bases-de-dados/

POLO, T., & MIOT, H. (2020). *Use of ROC curves in clinical and experimental studies*. From https://www.scielo.br/j/jvb/a/8S8Pfqnz8csmQJVqwgZT8gH/?format=pdf&lang=en

RIBEIRO, A. L. (2016). *Cardiovascular Health in Brazil: Trends and Perspectives*. From https://www.ahajournals.org/doi/full/10.1161/circulationaha.114.008727

ROSAEN, K. (2016). *Learning Log*. From http://karlrosaen.com/ml/learning-log/2016-06-20/

SAMMUT, C. &. (2010). *Encyclopedia of Machine Learning.*

SANG, H., LEE, H., & LEE, M. e. (2019). *Prediction model for cardiovascular disease in patients with diabetes using machine learning derived and validated in two independent Korean cohorts*. From https://www.nature.com/articles/s41598-024-63798-y

SHORE, C. (2020). *Ethical and Regulatory Considerations for Digital Health Technologies*. From https://www.ncbi.nlm.nih.gov/books/NBK563599/

STOUDT, S., VÁSQUEZ, V. N., & MARTINEZ, C. C. (2021). *Principles for data analysis workflows*. From https://www.researchgate.net/publication/350161536_Principles_for_data_analysis_workflows

TENSORFLOW. (n.d.). From https://www.tensorflow.org/tfx/guide/tft_bestpractices

VARGAS, V. W. (2022). *Imbalanced data preprocessing techniques for machine learning: a systematic mapping study*. From https://link.springer.com/article/10.1007/s10115-022-01772-8

YANG, C. C. (2022). *Explainable Artificial Intelligence for Predictive Modeling in Healthcare*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC8832418/

YARO, S. A. (2023). *Outlier Detection in Time-Series Receive Signal Strength Observation Using Z-Score Method with Sn.* From https://www.mdpi.com/2076-3417/13/6/3900

YUSUF, S. (2004). *Effect of potentially modifiable risk factors associated with myocardial infarction in 52 countries (the INTERHEART study): case-control study*. From https://pubmed.ncbi.nlm.nih.gov/15364185/

ZHANG, Z. (2020). *Predictive analytics in the era of big data: opportunities and challenges*. From https://pmc.ncbi.nlm.nih.gov/articles/PMC7049053/