Norwegian School of Economics
Bergen, Spring 2023

Balancing Data Protection and Model Accuracy

An Investigation of Protection Methods on Machine Learning Model Performance for a Bank Marketing Dataset

Jiahuan Du & Shravya Guruprasad

Supervisor: Nhat Quang Le

Master thesis, MSc in Economics and Business Administration
Major: Business Analytics

NORWEGIAN SCHOOL OF ECONOMICS

Abstract

The practice of sharing customer data among companies for marketing purposes is becoming increasingly common. However, sharing customer-level data poses potential risks and serious problems for businesses, such as substantial declines in brand value, erosion of customer trust, loss of competitive advantage, and the imposition of legal penalties (Schneider et al. 2017). These may eventually lead to financial loss and reputation damage for the companies. With the growing awareness of the value of personal information, more companies and customers are concerned about protecting data privacy.

In this paper, we used marketing data from a Portuguese bank to explore methods for balancing prediction accuracy and customer data privacy using various machine learning and data privacy techniques. The dataset includes observations from 45211 respondents and the observation period is from May 2008 to November 2010. Our goal is to find a method that enables third parties to share data with the bank while safeguarding customer privacy and maintaining accuracy in predicting customer behaviour.

We tested several machine learning models: Logistic Regression, Random Forest, and Neural Network (feedforward) on original data and then chose Random Forest, which gave the best prediction performance, as the model to proceed to explore. After using two different data privacy methods (Sampling and Random Noise) on the original data, we found the Random Forest model gives us accuracy levels that are very close to the accuracy before using the privacy methods. By doing this, we demonstrated a method for companies to protect customer data privacy without sacrificing predictive accuracy. The results of this study will have significant implications for companies that seek to share customer data while maintaining high levels of privacy and accuracy.

ACKNOWLEDGEMENT

This thesis is written as a part of the MSc in Economics and Business Administration at the Norwegian School of Economics (NHH), with a major in Business Analytics.

We would like to express our deepest gratitude and appreciation to all those who have supported us throughout the journey of completing this thesis. First and foremost, we are immensely grateful to our supervisor, Nhat Quang Le, for his guidance, expertise, and unwavering support. His invaluable insights, constructive feedback, and encouragement have played a pivotal role in shaping this thesis. We are truly fortunate to have had such a dedicated mentor who always pushed us to achieve our best.

We are also indebted to the lecturers and faculty members at NHH, who have contributed to our academic growth. Their lectures, seminars, and discussions have broadened our knowledge and provided a strong foundation for this thesis. We are grateful for their commitment to imparting knowledge and for their willingness to answer our questions and engage in meaningful academic dialogue.

Finally, we would like to express our sincere appreciation to our family, friends, and partners for their unwavering support and understanding throughout this endeavour. Their encouragement, patience, and belief in our abilities have been a constant source of motivation. Their presence in our lives has provided us with the strength and resilience to overcome challenges and persevere.

Table of Contents

1. INTRODUCTION

Effective marketing and advertising revolve around connecting with target audiences in a meaningful and relatable manner, distinguishing oneself from competitors, and crafting innovative and distinctive messages that not only reach customers but also drive conversions (Statista, 2023). Customer data is an important source for companies to analyse customer behaviours and to achieve such successful campaigns. Companies use customer data to predict customer behaviour and preferences for their marketing, conduct campaigns accordingly, customize their services, and create more customer value to boost sales and increase revenue. According to Brown et al. (2017), research shows that companies that utilize customer behavioural insights surpass their competitors by achieving an 85 percent increase in sales growth and a gross margin improvement of over 25 percent. A global survey conducted by McKinsey involving over 700 organizations revealed that investing in analytics for competitive intelligence, targeted customer engagement, and operational optimization resulted in operating-profit growth of around 6 percent (Brown et al., 2017).

Companies want to have more data from their customers such as social media data, to gain more insights. With these data combined, companies can make analyses from more aspects and can improve prediction accuracy. Schneider et al. (2017) mentioned a controlled experiment conducted by Cassidy et al. (2016), that shows that IKEA achieved significant results by partnering with Facebook. The collaboration led to increased footfall from new and existing customers, resulting in an overall lift of 11%. Specifically, the lift for the 22-25 year-old segment saw a remarkable increase of 31%. Moreover, the return on investment (ROI) for the paid media spend reached an impressive ratio of 6:1. According to Schneider et al. (2017), when a data provider's first-party data is combined with a media provider's first-party data, the resulting data is known as second-party data. This collaborative process offers benefits to both companies involved. The data provider gains two key advantages: firstly, access to an expanded range of demographic and media consumption profiles for their existing customers, which enables more personalized and relevant marketing. This leads to a richer set of first-party data. The data provider also gains access to valuable prospects and their demographic and media consumption profiles. Meanwhile, the media provider can benefit from charging the data provider for this collaboration. Sharing data is considered a vital marketing strategy in

numerous industries. As marketers often share their customer data with external parties to generate additional revenue from marketing activities, this process is a common way to add value for both parties involved (Schneider et al., 2017). The majority of marketing expenditures in the United States are allocated towards demographic, transactional, and behavioural third-party audience data (Statista, 2023). This phenomenon of companies sharing data has become more and more common. The estimated value of the global marketing-related data market was close to 17.7 billion U.S. dollars in 2021 (Statista, 2023).

However, sharing customer data has many potential risks and can result in many serious problems for businesses. A significant majority of Americans, approximately 64%, have been affected by a significant data breach, and a considerable proportion of the population expresses scepticism towards key institutions, such as the federal government and social media platforms, in safeguarding their personal data (Olmstead & Smith, 2017). As customers begin to harbour doubts regarding a company, the organization encounters a range of challenges, including diminishing brand value, erosion of competitive advantages, and even potential legal ramifications. Rahnama and Pentland (2022) mentioned that Facebook and Twitter are experiencing a drop in their daily active user numbers in their primary North American market, which is attributed to customers' growing realization that their data is being bought, sold, and utilized without their permission.

Since more companies and customers are becoming aware of the value of personal information, the importance of data privacy is rising. This makes protecting customer data critically important. According to Gimpel et al. (2018), data privacy has traditionally been seen as a cost to businesses, but it can be a source of competitive advantage. By implementing strong data privacy measures, businesses can enhance customer trust, which in turn can lead to increased sales and customer loyalty. Data privacy can be a win-win for both businesses and customers (Gimpel et al., 2018).

As concluded by Schneider and Iacobucci (2020), based on the current literature on data privacy in marketing and computing, there are two focus areas: firstly, is the control from the management aspect, such as limiting the amount and storage of data. Secondly, is to develop methods to protect customer data privacy by releasing model estimates (Holtrop et al. 2017) or a protected data set (Schneider et al., 2017). Releasing an estimated model poses several disadvantages. As cited by Schneider and Iacobucci (2020) from Little (1993), these disadvantages include "lack of flexibility in the choice of variables to be analyzed, and the

relative inability to do exploratory analysis and model-checking". Our paper is not trying to develop a fixed model nor to create a perfect privacy method for balancing but testing different machine learning models and two existing privacy methods and find a way to maintain data utility while protecting customer data privacy based on a specific bank marketing case. Through this, we contribute by preserving the data utility after using the privacy methods which are perceived as a "severe degradation of information (Gupta & Schneider, 2018)".

According to Schneider et al. (2017), the most effective approach to safeguarding data depends on a careful balance between risks and rewards. This balance takes into account factors such as the likelihood of misclassifying customers into segments, the potential drawbacks of wrongly assigning segment memberships, and the expected expenses associated with a data breach (Schneider et al., 2017). Therefore, managers need to have a methodology that can not only protect customers' data but also maintain prediction accuracy at the same time. As Ren et al. (2021) mentioned, over the past decades, numerous research endeavours have focused on striking a balance between data utility and information security. When it comes to statistical data analysis, the most prevalent approach is to introduce noise as a means to preserve certain statistical invariants, but this practice can potentially compromise the integrity of the data or dataset. In reality, it proves challenging to add Random Noise without significantly diminishing the utility of the data (Ren et al., 2021). However, there is a recent medical study conducted by Kusk and Lysdahlgaard (2022), which used machine learning (Neural Network) to make predictions after adding Gaussian noise at different noise levels to the original data. The study used a specific Chest X-rays image data set to train the models and it maintained (increased) the data utility after adding Random Noise (Gaussian noise) to the original data.

Our paper is trying to generate value within a business context by devising a technique that preserves prediction accuracy (data utility) after introducing noise to the original data. We will use the marketing campaign data of a Portuguese banking institution to train models. The best model is then chosen to make predictions after using a specific privacy method on original data to find a well-balanced strategy between privacy protection and data utility. Managers can use this strategy to protect customer data and develop customer trust while keeping good prediction accuracy and judgment of customer behaviours.

However, the findings of the study may not be universally applicable to other datasets and scenarios due to the specific nature of case study. The paper provides a way of thinking and suggests methods for companies to find the best strategy to balance privacy and accuracy.

In the following section, we will present a literature review to provide knowledge about existing research on statistical data privacy methods. Additionally, we will introduce some machine learning methods that are proven to be most effective and accurate in predicting binary variables and making analyses on customer behaviours. We justify our selection of specific models and data privacy methods, balancing prediction accuracy and data privacy considerations. In the last part of the literature review, we introduce the dataset used in our study and describe the specific scenario where our strategy can be applied. Based on the literature review, we choose specific models and data privacy methods to test. In the following data section, we introduce the dataset more in detail. Further, a methodology section presents implementations of applying all the models and privacy methods on the data. After the methodology section, we present our results, before discussing how well the results managed to achieve our expectation to protect customer privacy while maintaining the prediction accuracy. Within the discussion, we also include the implications for authors and readers. Finally, we conclude with how our results contribute to the field, and we discuss the limitations and possible future research possibilities.

2. LITERATURE REVIEW

2.1 Data Privacy Methods

Ensuring data privacy with a dataset involves taking appropriate measures to protect the sensitive and personal information contained within it. There have been studies that have explored different ways in which a dataset can be manipulated in order to ensure that data is secure.

One of the most common ways of protecting data is differential privacy. As Harvard University Privacy Tools Project (n.d.) has concluded, differential privacy can be understood as a precise mathematical concept pertaining to privacy. In its simplest form, it relates to an algorithm that examines a dataset and computes various statistics, such as mean, variance, median, mode, and so on. If such an algorithm is considered differentially private, it means that merely by observing the results, it is impossible to determine whether any specific individual's data was present in the original dataset or not. As Ren et al. (2021) cited from Soria-Comas et al. (2017),

differential privacy (DP) is a robust method for ensuring privacy, which aims to establish limits on the amount of information that can be exposed through the inclusion of an individual's data in a database. In this particular form of data protection method, Random Noise is added to data before it is analysed.

This method is similar to the one that this thesis explores, however, in differential privacy, noise is added to a dataset based on a privacy budget or parameter ε. The lower the value of ε, the higher the level of noise added and vice versa (Jain et al., 2018). Certain challenges arise as a result of employing this specific approach. One of the downsides of using differential privacy is that it is difficult to find the specific privacy parameter or an efficient value for ε. As quoted by Jain et al. (2018), "For a given computational task T and a given value of ε there will be numerous differentially private algorithms for achieving T in an ε-differentially private way. Some will have better accuracy than others." Finding a proper value for ε can be troublesome making it sensible to explore other methods of data privacy methods to protect consumer privacy. According to ETI (n.d.), differential privacy sacrifices accuracy for privacy; people who use differential privacy need to decide the extent to which they are willing to compromise accuracy to safeguard individuals' protection by setting parameter ε to reflect their values and priorities. Although advanced data perturbation techniques like differential privacy offer prospects for enhancing confidentiality, the pursuit of achieving optimal data privacy without compromising data utility remains an ongoing NP-hard (non-deterministic polynomial-time hardness) challenge (Mivule, 2012). Testing and exploring this method should be done in a separate paper, however, in this paper, we are focusing on trying more methods that are promising in maintaining good prediction accuracy while protecting data privacy.

Many other statistical data privacy methods are also being widely used. To contextualize our paper within the existing literature on data protection methodologies, we rely on the taxonomy developed by Schneider & Iacobucci (2020). Schneider & Iacobucci explored various privacy protection methods on survey data, including Aggregation, Sampling, Random Noise, Median Split, and Data Shuffling. This thesis focuses on further studying the methods explored by the authors of this literature. The authors form a proposed method in the literature, however, this particular method is not explored as it is out of scope for this thesis.

Aggregation includes replacing the values of one consumer with the average of all responses of one variable. Phls et al. (2014) also explored the use of *Aggregation* as a means of data protection and managed to achieve a well-protected dataset which was still "usable" for statistical predictions. According to Gupta & Schneider. (2018), U.S. Census Bureau and the Department of Agriculture collect sensitive data and use an approach to convert the original data into protected data and then release it. The methods these agencies use to perturb data include adding *Random Noise* and *Aggregation*. These methods are used to preserve the utility of the data while reducing the chance of privacy breaches by potential intruders (Gupta & Schneider, 2018). *Median Split* includes splitting the dataset based on rank order and setting one half to 1 and the other half to zero and *Data Shuffling* includes *Sampling* and replacing from a conditional distribution (Schneider & Iacobucci, 2020). *Median Split* is an attractive choice as they "make analyses easier to conduct and interpret", however, they are mainly used for continuous variables (Iacobucci et al., 2014). This would make it difficult to use on a dataset that is binary which constitutes most of the data types of the dataset used in this thesis. This is the same for the case of *Aggregation*.

Sampling refers to random Sampling from a select group and replacing this sample in another selected area of the responses. *Random Noise* refers to randomly adding noise to variables of a respondent at different percentage levels (Schneider & Iacobucci, 2020). Castro & Brankovic (1999) examined the risks associated with rapidly evolving data mining technology and the corresponding privacy concerns. They explored various methods of data protection, including the use of noise and swapping techniques. The study focuses on addressing the privacy issues arising from advancements in data mining and proposes strategies to mitigate these risks. Random Noise addition is used to protect data and uses decision trees as a means for their classification problem and measuring precision. The paper finds that Random Noise addition helps effectively protect data as well as keep the patterns derived in the original dataset implying that the data also maintains its utility and accuracy.

Kadampur & D.V.L.N (2010) also explore noise addition methods to protect data and find that noise addition methods specifically shuffling attribute values with certain probability handle the problem of data privacy issues as well as preserve the prediction accuracy. The literature handles datasets very similar to that of this thesis containing a mix of categorical and numeric

data types and focusing on a classification issue. The noise addition methods used in the literature "are effective in preserving the privacy of the data proper and producing prediction accuracies on par with the original dataset" (Kadampur & D.V.L.N, 2010).

This thesis focuses on a dataset with many binary variables and for simplicity, we have chosen two of the methods that are covered by Schneider's paper to explore the effects of how using data protection methods can affect the quality and utility of the information contained in the dataset. The two methods we will focus on are *Sampling* and *Random Noise.* Except for their proposed method, Sampling and Random Noise are the two methods that give the best trade-off between data privacy and data utility according to Schneider & Iacobucci. Therefore, we have chosen these two methods to utilize in our thesis.

Before using privacy methods on the data, we wanted to find out the model that can give the best possible predictions. Along with the problem of data privacy lies the issue of data utility, having a well-protected dataset would not be beneficial for the company if the data itself has no value and cannot be used for making business decisions. In order to make informed decisions for the future, businesses need to use predictions. By analysing past and current data, businesses can create accurate forecasts and anticipate potential outcomes. Predictive analytics can be used to identify trends and patterns in consumer behaviour and other metrics. This information can then be used to develop effective marketing strategies, improve operational efficiency, and reduce risk. Businesses can also use predictive models to anticipate changes in the market and adjust their strategies accordingly. By incorporating predictions into decision-making processes, businesses can make more informed decisions and stay ahead of the competition.

In order to make these predictions, businesses are shifting away from traditional methods to machine learning models to increase the efficiency and accuracy of the predictions. The rate at which data is evolving and increasing makes it difficult to continue using traditional methods.

2.2 Machine learning

Machine learning is a part of artificial intelligence that involves the development of algorithms and statistical models that enable computer systems to learn from data. Machine learning is based on the idea that machines can learn patterns and make predictions based on

data. Some areas where machine learning is widely used are image and speech recognition, medical diagnosis, and predictive analytics (Paratela, 2023).

Machine learning makes predictions on data through a process of training, where the algorithm receives a large set of data and is expected to learn patterns in that particular dataset to predict as accurately as possible on a new dataset. There is no need to explicitly program the machine to learn these patterns which can be time-consuming as it learns on its own from the information presented to it (Brown, 2021). There are various types of machine learning methods, however, this thesis focuses on supervised learning.

"In just the last five or 10 years, machine learning has become a critical way, arguably the most important way, most parts of AI are done."

- *Thomas W. Malone*

Supervised learning is a form of machine learning where the model is trained using labelled data. Labelled data refers to when there are meaningful labels associated with the raw data collected. Supervised learning is commonly used in regression and classification tasks. Regression explores correlations between dependent and independent variables in areas such as predicting house prices or market trends. Classification, on the other hand, categorizes data and divides it into classes based on certain parameters. For example, predicting customer churn or spam classification (Terra, 2023).

This thesis focuses on a classification problem. Classification problem is a type of supervised learning task that involves predicting a categorical label or class for a given input (Brownlee, 2020). It is used in a wide range of applications, including image recognition or text classification. To solve this classification problem and explore the trade-off between data privacy and data utility, the thesis dives into different machine learning models that can be utilized. The following subsections will explore the main aspects of the machine learning models used in this thesis. The models include Logistic Regression, Random Forest, and Neural Networks. This thesis explores these models and how data protection methods affect and influence the accuracy of the predictions. The accuracy measures the level of data utility. The higher the accuracy the higher the level of data utility and vice versa.

2.2.1 Logistic Regression

Generalized linear model (GLM) is a statistical framework that is an extension of the linear regression model to handle a wider range of data distributions, including non-normal distributions such as binomial, Poisson, and exponential. GLMs provide a flexible and powerful approach to modelling data that cannot be accommodated by standard linear regression.

There are 3 components to the Generalized Linear Model (Dobson & Barnett, 2018):

1. A probability distribution for the response variable $Y_1,\ldots,Y_N$

2. Set of parameters β and predictor variables

$$\mathrm{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_N^T \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & & \vdots \\ x_{N1} & & x_{Np} \end{bmatrix};$$

3. A link function g that links the linear predictor to the mean of the response variable.

$$g(\mu_i) = x_i^T \beta$$

GLM can be utilized with binary variables and more specifically classification problems (Dobson & Barnett, 2018). Logistic Regression is a subset of GLM and is mostly used in classification problems. The thesis will focus on Logistic Regression as the main model for GLM. In this type of regression, the dependent variable is assumed to have a binomial distribution, and the log odds link function, $\ln\left(\frac{p}{1-p}\right)$, is used to model the relationship between the independent variables and the probability of the dependent variable being a certain value.

The formula for Logistic Regression is as follows:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x$$

Logistic Regression is a simplified model which is used for classification models and hence we found it right to use this model to test our theory. The dataset in question is a classification problem and Logistic Regression fits in to solve this particular form of problem. Logistic Regression is relatively easy to interpret compared to other machine learning models and computationally efficient. Secondly, it isn't as easily prone to overfitting, which is a very common problem when it comes to training datasets. Lastly, Logistic Regression not only reveals the significance of a predictor in determining the final outcome but also the nature of its association, whether positive or negative (Jain, 2020).

2.2.2 Random Forest

Random Forest combines multiple decision trees to improve the accuracy and robustness of a classification task. Decision trees, more relevantly, classification trees are the foundation of Random Forest where the predictor space is segmented into simpler regions. To predict an observation, it's common to utilize the mean or mode of the training observations in the corresponding region (James et al. 2017). A decision tree's leaves show the final output, while its branches represent decision rules based on input features. The tree progresses from the root to the leaves, refining predictions at each node and branch.

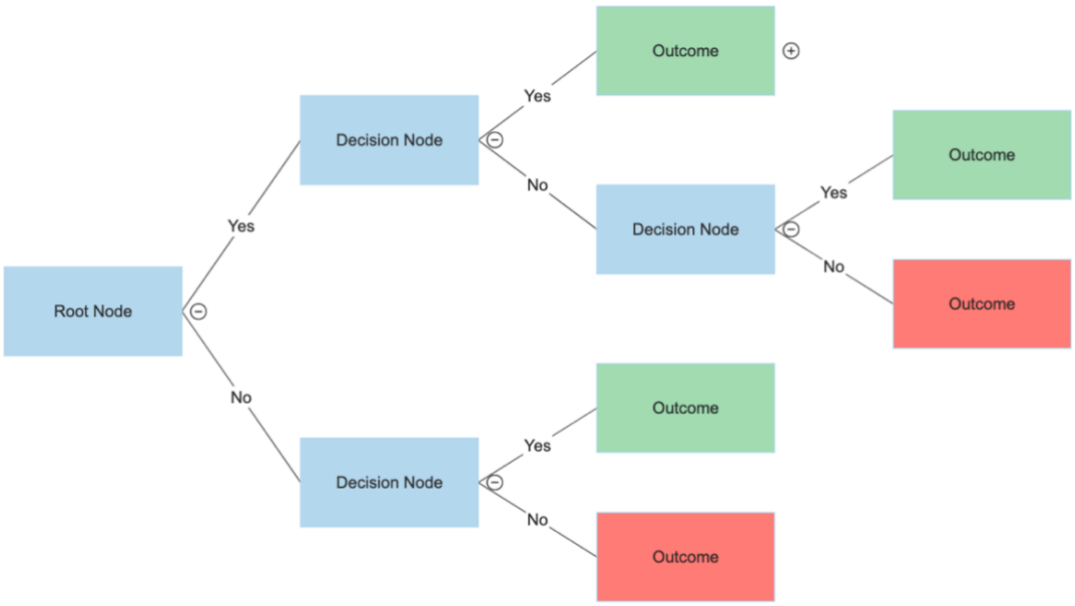

Figure 2.1: Example of a Decision Tree

Although decision trees are easy to interpret and explain, they do have poor prediction accuracy results due to high variance and are also non-robust to factors like noise. Changing the dataset in the smallest way could lead to a large change in the final estimated tree. Random Forest helps to solve this problem by producing multiple trees whose combined results form a final prediction (James et al. 2017). Random Forest works with the process of *bagging* or *bootstrap aggregation* where many noisy but approximately unbiased models are averaged. Usually, it is not possible to have access to many training sets, so the samples are repeatedly taken from the same training dataset *(bootstrapped*). This technique helps to reduce the variance of an estimated prediction function (Hastie et al., 2009).

This thesis explores the effect of noise on a dataset and hence requires the most robust of models to be able to handle this change in the original dataset. Random Forest is a good choice for our dataset because it can handle both numerical and categorical data, and it can capture non-linear relationships between the features and the target variable. Additionally, Random Forest can handle missing data and outliers, which can be very common in datasets. Overall, the flexibility, robustness, and high accuracy of Random Forest make it a strong choice for us to use in our thesis.

Random Forest has a history of achieving high prediction accuracy results as demonstrated by Couronne et al. (2018), where Random Forest outperformed Logistic Regression in terms of predictive accuracy 69% of the time. To be more relevant to our thesis, it is crucial to find a model that could also withstand noise. According to Schooltink (2020), Random Forest has a high robustness and lower sensitivity to datasets with introduced Gaussian noise as compared to other models such as Support Vector Classifiers. Elsewhere, Random Forest also performed the best in a synthetically generated dataset with generated noise, having the highest classification accuracy at almost all levels of noise (Lehtihet & Åryd, 2021).

2.2.3 Neural Network

Neural Network, a type of deep learning model, is inspired by the functional structure of the human brain. It consists of interconnected nodes called neurons. These neurons receive inputs, infer meaning, perform computations on the inputs, and subsequently generate an output.

Like Random Forest, Neural Network is highly flexible and known to be robust to noise. In one study, C*onvolutional Neural Network (CNN)*, a type of Neural Network, was used on image data where Gaussian noise was introduced and the prediction increased around 7% (Kusk & Lysdahlgaard, 2022). This study proved that Neural Network could withstand noise very well. However, according to Varma, S., & Das, S. (2019), a Convolutional Neural Network is invented for handling data that are consisted of images. The data used in this thesis is two-dimensional, therefore, if we want to explore with Neural Network, we need to explore other models other than CNN.

Feedforward Neural Network is the simplest form of Neural Network, where there is an information flow, through the network, from the input layer to the output layer in one direction. In the output layer, the output of each neuron is a weighted sum of its inputs. This output is passed through an activation function. This activation function decides whether or not the neuron will be activated or not by assessing the input value against a threshold value. The most commonly used activation function is ReLu (Agarap, 2019).

The formula for the ReLu activation function is as follow:

$$f(x) = \max(0,x)$$

Using a ReLu activation function for Neural Networks comes with its advantages. One of the advantages of ReLu is its computational efficiency. Unlike other mathematical functions, ReLU only requires a comparison and a maximum operation. ReLU requires less computation than sigmoid and hyperbolic tangent functions, making it faster and more efficient (Vivek, 2022). Utilizing this function also has the advantage of not activating all neurons simultaneously. Neurons are deactivated only when the linear transformation produces zero output (Sharma & Athaiya, 2020).

The thesis focuses on using *deep feedforward Neural Network*, which is also characterized by the unidirectional flow of information. However, unlike traditional feedforward networks, deep feedforward networks encompass multiple hidden layers positioned between the input and output layers.

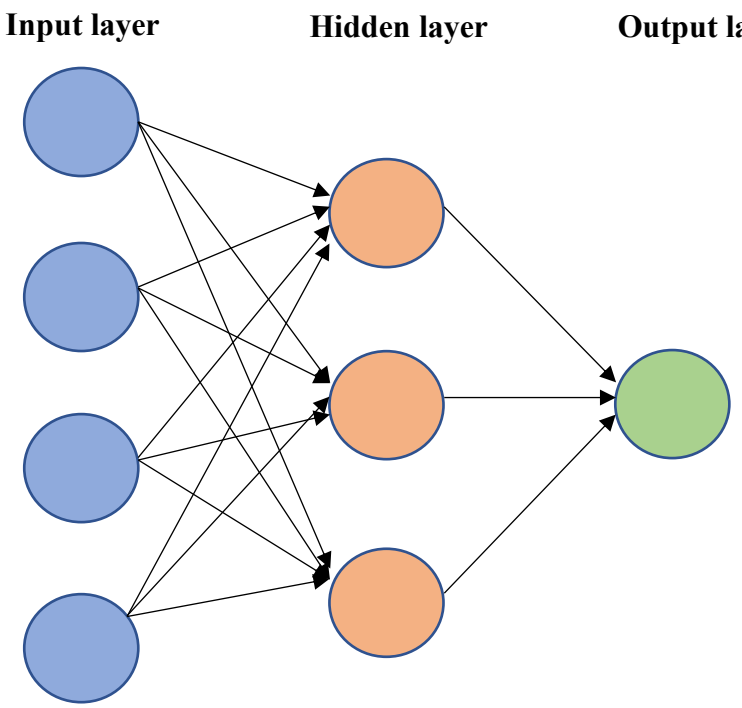

Figure 2.2: Feed Forward Neural Network for Classification (one hidden layer).

The data we used to explore the strategy is related with direct marketing campaigns of a Portuguese banking institution, retrieved from the UCI Machine Learning Repository which is publicly available. The dataset records the details and behaviours of 45211 customers between May 2008-November 2010. The marketing campaigns were based on phone calls. The bank wants to know if the client has subscribed to the product (bank term deposit) or not. If the client subscribed, the variable 'y' is yes. Otherwise, the variable y is no. The data also contains other information of the clients, such as age and job. In this paper, we take the 'y' variable as the dependent variable, which is the one that bank want to make predictions on. We take all other variables as independent variables and classified them into two groups. One is personal (sensitive) information such as age which is personal identity information and customers would not like companies to share with others. The other group is general information such as the number of contacts before this campaign, which is the type of variable that customers do not mind companies to share with other companies. This classification for the data is based on an investigation around the authors' surrounding people.

The reason we chose this dataset is based on the assumptions that if a company such as this bank wants to predict on the subscription of its clients, but they have limited data on the

customers. In order to make accurate predictions, the bank buys some data of these customers from a media provider and the data they bought including some personal data: 'Age', 'Housing', 'Balance', 'Education' and 'Marital'. The media provider should use the data privacy methods before selling it to the bank, at the same time, the media provider also needs to ensure the data is still valuable in predicting. Therefore, we used this data to explore based on these assumptions.

3. DATA

The original data used in this thesis is the marketing campaign data of a Portuguese banking institution retrieved from the UCI Machine Learning Repository[1] that involves exploring consumer behaviour through a bank marketing scheme (UCI Machine Learning Repository, 2014). The dataset that is available on the repository was initially used by Moro et al. (2014). There are 4 different datasets available on the repository. However, for this thesis, the older version that is titled 'bank-full' is used since the newer versions have extra inputs that are irrelevant to the goal of the thesis. The dataset contains 45211 observations and includes 17 variables. The observation period is between years 2008-2010. Apart from the one dependent variable, all the other variables are independent.

To ensure compatibility with the machine learning models, the data needs to undergo a cleaning phase prior to processing. This step of cleaning the data plays a crucial role in optimizing the performance of the models. By addressing any issues or inconsistencies in the data, it ensures accurate predictions of the desired variable. We will present the implementation of this step in the descriptive statistics section, however, prior to that we will investigate the original data.

3.1 Overview of Variables

3.1.1 Dependent Variable

The dependent variable is the variable that is dependent on and affected by changes in the independent variables. The dataset we are using contains one dependent variable which explores whether or not a customer of a bank will subscribe to a term deposit as a result of a

[1] https://archive.ics.uci.edu/ml/datasets/bank+marketing

marketing scheme carried out by the bank. The variable is titled 'y'. If the customer subscribes to a term deposit, the value of 'y' is 'yes', otherwise the value is 'no' (Moro et al., 2014).

3.1.2 Independent Variables

The independent variable is not affected by the changes in any other variable and is usually controlled to explore its effect on the dependent variable. The dataset contains 16 independent variables representing the characteristics of each customer. For the purpose of the thesis, independent variables were split into two groups; highly sensitive and personal information of a customer as well as more general info. The overview and descriptions of the variables are given in two separate tables below corresponding to their nature.

Independent Variable (Sensitive)	**Description**
Age	Age of the Customer (years) (continuous)
Job	Occupation of the customer *(management, technician, entrepreneur, blue-collar, admin, services, self-employed, unemployed, housemaid, student)*
Marital	Marital Status (s*ingle, married, divorced)*
Education	Education level held by customer (p*rimary, secondary, tertiary, unknown)*
Default	Whether or not the customer has credit in default *(yes, no)*
Balance	The balance of the customer at the bank (euros) (continuous)
Housing	Whether the customer has a housing loan *(yes, no)*
Loan	Whether the customer has a general loan *(yes, no)*

.

Table 3.1: Highly Sensitive Independent Variables

Table 3.1 explores the highly sensitive independent variables in the dataset. These variables are considered sensitive and personal to a customer and would be damaging if this information

was leaked. Sensitive variables contain information that could potentially cause harm or discrimination to individuals or groups. For example, if the age, job and marital status of a customer got leaked, it would be relatively simple to construct the identity of the exact customer. According to Jain and Kesswani (2021), determining the sensitivity of data attributes and who has the authority to make such decisions poses a significant challenge in data classification. In their proposed mechanism for data classification, they allow data owners to personalize their data privacy by configuring attribute sensitivity as either sensitive or non-sensitive at the application level. The data used in our thesis is secondary data, so we do not have the chance to gather this information from actual respondents. However, based on the same purpose of the classification – respecting respondents' personal preference, we investigated 30 people (friends, classmates and families) about their preferences by asking them to choose among the variables in our data that they perceive as sensitive information. In this small-scale research, we found there are five variables that have been mentioned by participants as personal information which may trigger higher level of privacy concerns than other information. The five variables are respectively: 'Age', 'Housing', 'Balance', 'Education' and 'Marital'. Therefore, we classify these five variables as sensitive information and the other variables in the data set as non-sensitive variables. In this paper, we are going to use this classification to make further explorations.

Independent Variable (non-sensitive)	**Description**
Contact	How the customer was contacted (*unknown. cellular, telephone*)
Day	The day of the month contacted (1-31)
Month	The month contacted (months)
Duration	Duration of the call (Continuous)
Campaign	Number of contacts performed in this campaign (Continuous)
pdays	Number of days passed since customer was contacted since previous campaign (Continuous)
Previous	Number of contacts for customer before this campaign (Continuous)

poutcome	Outcome of previous campaign (*success, failure, other, unknown)*

Table 3.2: General Independent Variables

3.2 Descriptive Statistics

Descriptive statistics play a crucial role in analysing and interpreting data. Exploring descriptive statistics include summarizing the essential characteristics of a dataset and looking at patterns and relationships within the data. This process allows us to identify and address any outliers or anomalies present in the data, facilitating the necessary data cleaning procedures to ensure its readiness and compatibility with the learning models.

The dataset used in this thesis has more categorical and binary variables as compared to numeric variables. However, the numeric variables are also important in the exploration of the data and the prediction of our outcome. The descriptive statistics of the numeric variables are given below.

	Variables						
	Age	**Balance**	**Day**	**Duration**	**Campaign**	**pdays**	**Previous**
count	45211	45211	45211	45211	45211	45211	45211
mean	40.94	1362.27	15.81	258.16	2.76	40.20	0.58
std	10.62	3044.77	8.32	257.53	3.10	100.13	2.30
min	18.00	-8019.00	1.00	0.00	1.00	-1.00	0.00
25%	33.00	72.00	8.00	103.00	1.00	-1.00	0.00
50%	39.00	448.00	16.00	180.00	2.00	-1.00	0.00
75%	48.00	1428.00	21.00	319.00	3.00	-1.00	0.00
max	95.00	102127.00	31.00	4918.00	63.00	871.00	275.00

Table 3.3: Descriptive Statistics of Numeric Variables

Table 3.3 outlines the descriptive statistics of the numeric variables. The minimum age of a customer contacted is 18, while the maximum is 95, however majority of the customers (75%) are above the age of 33. The *Balance* has a negative value which will be rectified in the

following section. The maximum balance is 102,127 euros. There are 50% of the customer have a balance between 72 and 1428 euros. The *Duration* and *pdays* columns have significant variations. The mean values of these columns are far away from the max values which shows the high level of variation. The *Campaign* column has a slight amount of variation as well. High levels of variations in a dataset can negatively impact machine learning models as they may struggle to find meaningful patterns or generalize well resulting in poor performance. These variations will be handled in Section 3.3, where the dataset will be pre-processed for the machine learning models.

After conducting a comprehensive examination of all independent variables within the dataset, only the variables we considered most relevant to the thesis will be displayed. The first set of variables that are displayed are the demographic variables. The demographic variables characterize an individual and in the case of our thesis, all the demographic variables are highly sensitive variables. Figure 3.1 shows the frequency of each demographic variable in the dataset.

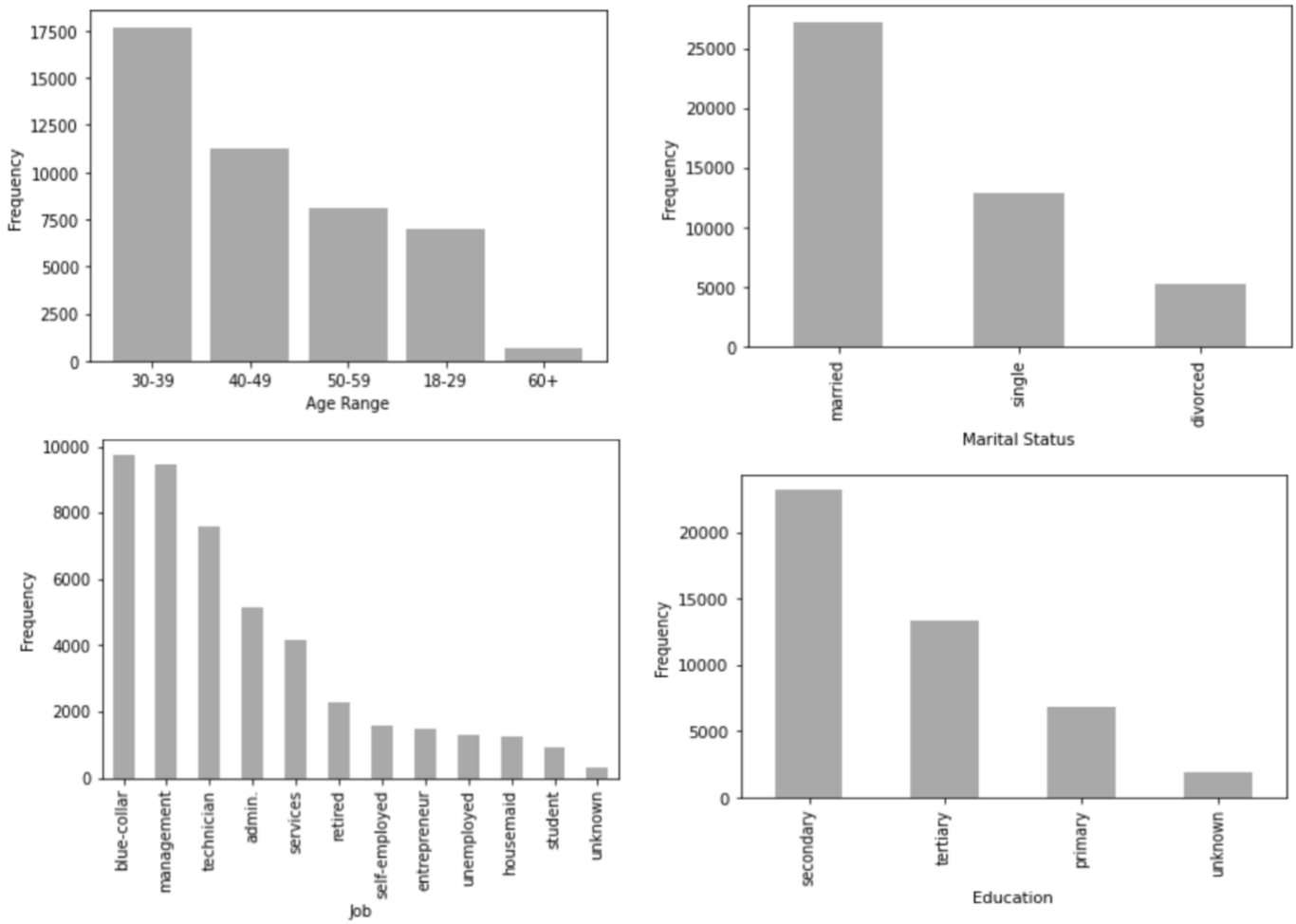

Figure 3.1: Demographic Variables

The highest age range is between 30-39 which was also highlighted in Table 3.3, while there are not much senior customers over the age of 60. Majority of the customers are also married with over half of the customers having at least a secondary level of education. There are a multitude of job categories with blue collar and management jobs taking the lead, followed by technician and admin roles. Unemployed customers and students are rare which is understandable as they would not be targeted by the bank for their campaign due to the lack of funds.

Figure 3.2 explores the default or loans a customer may have. These variables are not demographic but still considered sensitive.

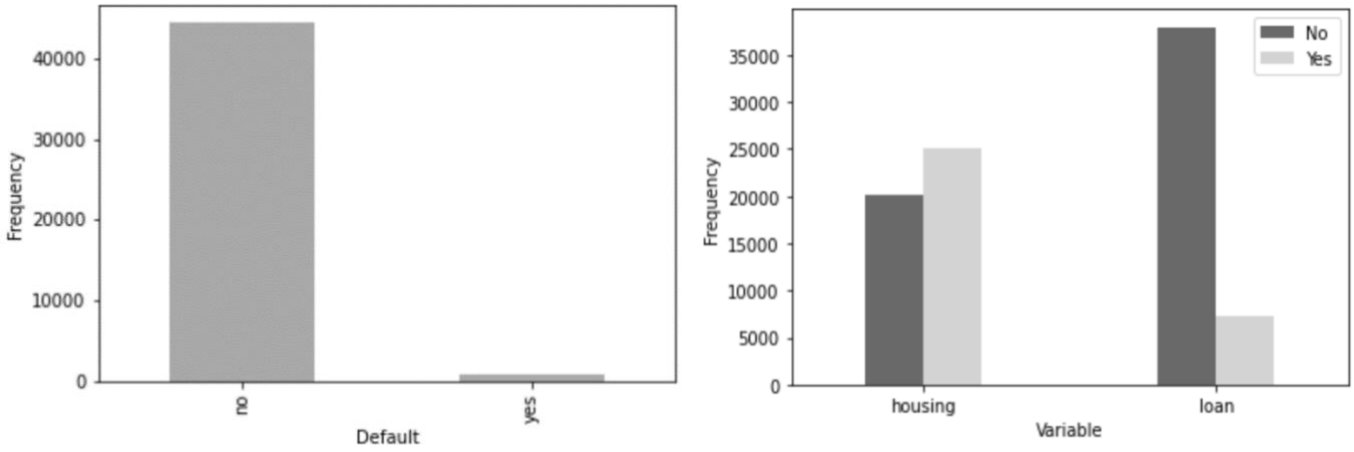

Figure 3.2: Number of Customers with Loans and Housing Loans.

There is not a high frequency of default in the dataset, which implies most customers are well on track of repaying loans. There are not a lot of people who have general loans from the bank, however, more than half of the customers do have a housing loan to pay off. Some of these sensitive variables will be explored further about how they affect the final subscription later in the paper.

Below shows the customer subscription levels for the deposit as a result of the marketing scheme.

Subscription Results	
yes	5289
no	39922

Table 3.4: Total Subscription

From Table 3.4, we can clearly see that the number of customers who do not subscribe to the deposit outweighs the number of customers who do. Nearly 88% of the data is classified under 'no' while 12% is 'yes'. This is a case of an imbalanced dataset. An *imbalanced dataset* refers to a situation where the number of instances in one class significantly outweighs the number of instances in the other class. Imbalanced datasets pose challenges for binary classification because most machine learning algorithms are designed to optimize overall accuracy, which can be misleading in the presence of class imbalance. The algorithm tends to favour the majority class, resulting in poor performance on the minority class. Hence, it would be crucial to explore other metrics of model evaluation other than only accuracy (hit rate) to compare model performance. This will be discussed further in the following sections.

Although the number of customers who have subscribed is low, it is crucial to explore the effects of the independent variables on the number of subscriptions. This helps to identify patterns and relationships among different variables in a dataset, which can provide valuable insights and inform decision-making processes. There are many independent variables and all of them have been thoroughly investigated, however only the most relevant ones are displayed in the following section.

Firstly, we detected the effects of variable 'Age'. Figure 3.3 shows the number of people that subscribe within a particular age group. The figure shows denser bubbles for the customers between age 30-39 as compared to the younger (below 30) and elder groups (above 40). This indicates that customers between 30 to 39 years old have been contacted more as part of the campaign, as compared to the younger and elder groups. The variable 'Campaign' is the number of contacts performed in this campaign. A higher campaign value means a customer (bubble) has been contacted more. With this aspect we can see that the customers have been

frequently contacted in the middle age group. This is probably due to the idea that younger and senior citizens would not have the ability to subscribe due to financial restrictions etc. However, in terms of the subscription, even though younger and elder age group are the least contacted, they still have a better subscription rate within the age group as compared to customers in age groups between 26-60. From the patterns we can conclude that variable 'Age' has an effect on subscription (dependent variable 'y').

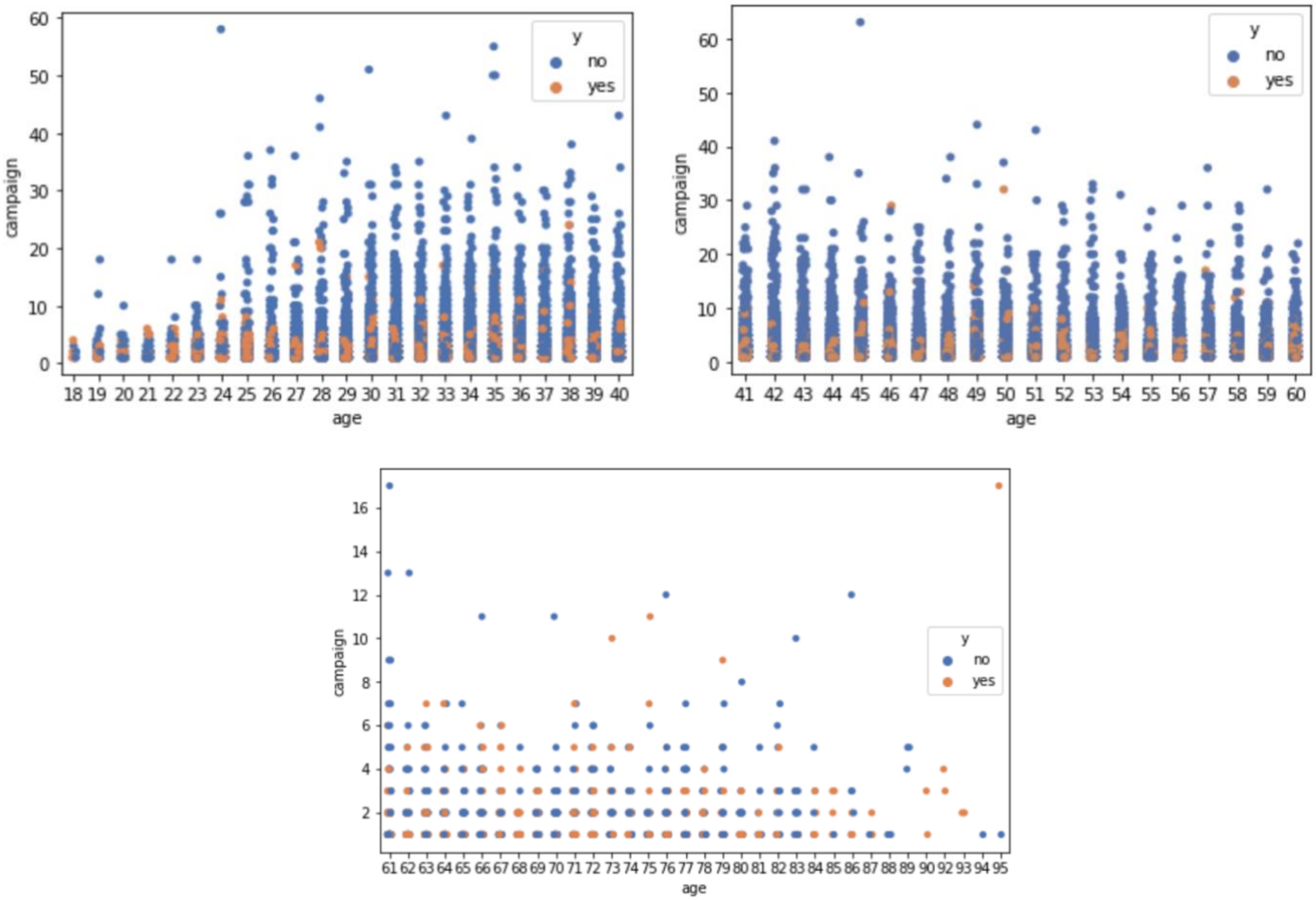

Figure 3.3: Effect of Age on Subscription

Figure 3.4 investigates the impact of general loans and housing loans on customer subscription. From the figure we can see that customers with these loans were less likely to subscribe to bank deposits compared to customers without these loans.

Specifically, we found that customers who had taken out general loans were less likely to subscribe to a bank deposit than those who did not have a general loan. Figure 3.4 shows the difference and it is clear that the orange portion (people who subscribe) is very difficult to see in customers who have taken a loan, as opposed to those who have not. This could be due to the fact that customers who have already borrowed money may not have enough disposable income to invest in a bank deposit.

Similarly, we found that customers with housing loans were also less likely to subscribe to bank deposits. This could be due to the fact that housing loans are a long-term investment, and customers may prefer to focus on paying off their housing loans rather than investing in bank deposits.

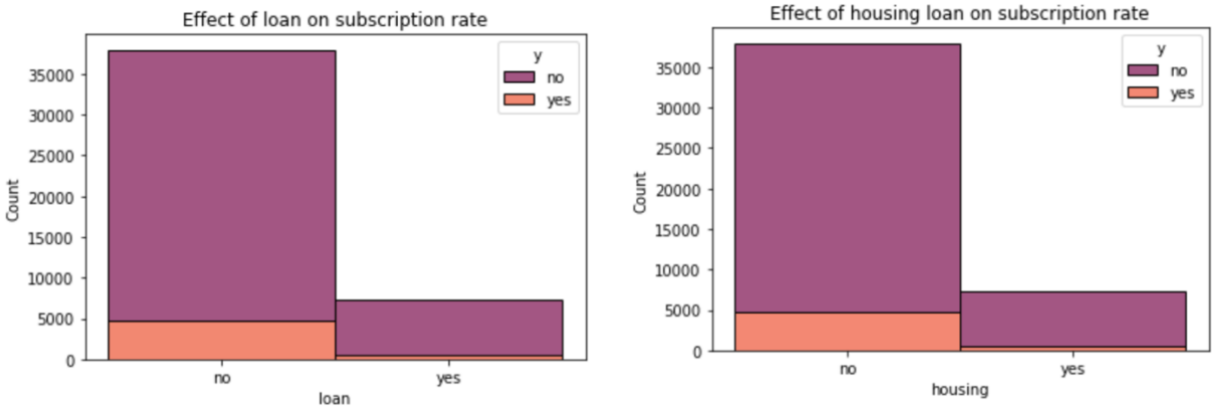

Figure 3.4: Effect of Loan and Housing Loan on Subscription

Figure 3.5 explores the relationship between the duration of a customer call and the customer subscription. Specifically, the graph shows that calls lasting more than 500 seconds have more subscriptions. This could be due to the fact that longer calls may indicate a deeper level of engagement with the customer, as the agent may have had more time to understand the customer's needs and provide tailored recommendations or solutions. Additionally, longer calls may indicate a higher level of satisfaction with the customer service experience, leading to a greater likelihood of subscribing to the service.

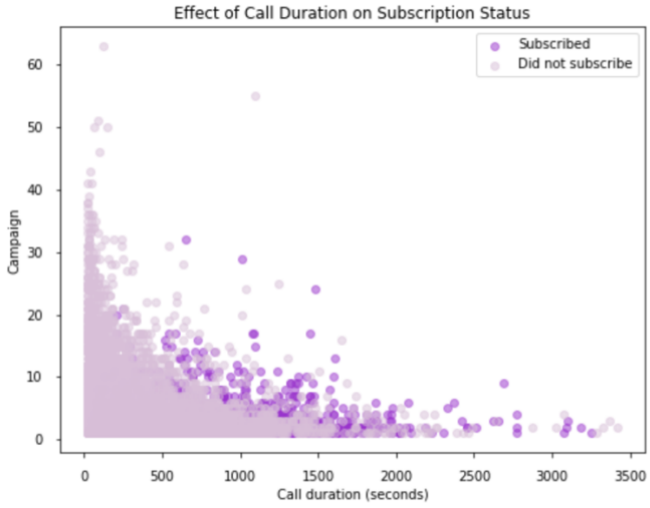

Figure 3.5: Effect of Duration on Subscription

However, it is important to note that correlation does not necessarily imply causation. Further analysis, such as regression analysis, would be needed to establish a causal relationship between call duration and subscription rate. The duration column will be further explored in the data pre-processing step in Section 3.3.

Based on the line graph in Figure 3.6, there seems to be a fluctuation in customer subscription rate over the course of a year. The graph shows that subscription rates increase until May, after which they start to decrease, reaching their lowest point in December and January. There could be several factors contributing to this trend. For example, customers may be more likely to subscribe during the spring and summer months, when they have more disposable income or when they are planning summer activities. Conversely, customers may be less likely to subscribe during the holiday season, when they have other priorities than on purchasing new subscriptions.

Additionally, external factors such as the overall economic climate or the state of the industry may be affecting subscription rates. Further analysis, such as examining changes in marketing or promotional strategies during different months, would be needed to determine the underlying causes of the subscription rate fluctuations.

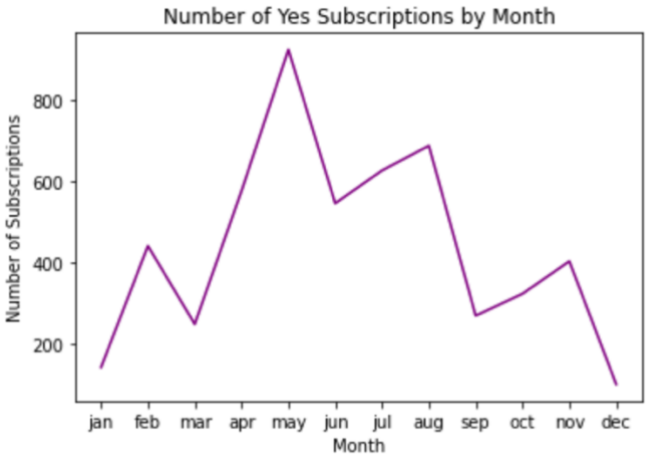

Figure 3.6: Number of Subscriptions by Month

3.3 Data Pre-processing

Before training machine learning models, the data has to be pre-processed. Data pre-processing is critical for machine learning because the performance of machine learning models depends heavily on the quality of the input data. Machine learning algorithms rely on large amounts of data to train, test, and evaluate their performance. Therefore, the quality and structure of the input data can significantly affect the accuracy and generalizability of the models. This process ensures that the input data is clean, consistent, and in the correct format so that machine learning algorithms can better identify the underlying patterns and relationships in the data. This ensures that we get better predictive accuracy, and better interpretability of the model.

The original dataset from UCI has no null or missing values, hence there was no need to change or impute values into the data. Table 3.1 and 3.2 display the types of each variable. All the variables are categorized either as integers or characters. For character variables, we have binary variables (‘yes’ and ‘no’) and categorical variables which have more than two categories, such as ‘Job’. Machine learning models usually need their input and output variables to be numerical (Brownlee, 2019). It is further proven as both our Random Forest and Neural Network models threw errors when we tried to pass the original dataset through them. As a result of this issue, we have processed the dataset in order to make the models perform

efficiently. The integer variables remained unchanged, and the first step was to convert the binary variables from 'yes' and 'no' to '1' and '0', respectively. Then we need to convert the categorical variables. For this we used the *one-hot encoding* to handle the original data for our Random Forest and Neural Network models.

Although the Logistic Regression model demonstrates proficiency in handling binary, categorical, and numerical variables, we ensured fair competition by utilizing the same pre-processed data that was used to train the Random Forest and Neural Network models for training our Logistic Regression model. To ensure there was no information loss to the dataset by processing the variables, we tried to train Logistic Regression model on both original data and the pre-processed data. The prediction accuracy was the same in both models, therefore, we used the same pre-processed data to train all three models.

One-hot encoding

Using one-hot encoding, categorical variables can be converted into numerical variables that can be used by machine learning algorithms. Each category is represented by a binary vector with one element set to '1' and the other element set to '0'. This creates a sparse matrix, where each row represents a data point and each column represents a category. One-hot encoding is useful because it preserves the distinction between categories and avoids the numerical assumptions that may occur with label encoding.

In our dataset, we have 6 categorical variables, 'Job', 'Marital', 'Education', 'Contact', 'Month', and 'poutcome'. After one-hot encoding, each separate category within one categorical variable was converted to a column with binary variables in each column. Once all the categorical variables were one hot encoded, the final dataset had 48 variables. A sample is shown below for the 'Marital' variable.

marital
married
single
married
married
single
married
single
divorced

Before Encoding

maritaldivorced	maritalmarried	maritalsingle
0	1	0
0	0	1
0	1	0
0	1	0
0	0	1
0	1	0
0	0	1

After encoding

Figure 3.7: Example of One-Hot Encoding (*Marital*).

Finally, we made changes to the *pdays* and *Balance* columns as well by removing the negative values of the variable. The minimum value for both *Balance* and *pdays* is '0'. The variation and outliers were removed for the *pdays* column as well. We kept the maximum *pdays* value to a year so 365 days at maximum. Table 3.3 shows that the *Duration* column has a large variation as well. However, in this thesis we have dropped the *Duration* column from the dataset. The *Duration* has the most importance in predicting the subscription rate, however, it is not the most realistic of a real-world situation because the value is not known before a call is performed. Once the call is performed, the 'y' value is automatically known whether or not the customer subscribed. To construct the model, it is impractical to await the completion of

the call and receive the duration inputs. Consequently, this column is omitted to ensure the development of a realistic predictive model.

4. METHODOLOGY

In this section, we will present the implementations of the models and methods discussed in the previous section. Initially, the pre-processed dataset is utilized to train three models: Logistic Regression, Random Forest, and Neural Network. Among these models, the best performing one is selected as the primary choice. Subsequently, this model (achieved algorithm) is trained on the datasets augmented with additional protection methods, resulting in multiple trained models (algorithms). Finally, the accuracies of these models are compared. The results will be shown in the next following chapter.

4.1 Software Used

This research utilized Python and R for the building, tuning and predicting of the models. R is used for the Logistic Regression, while Python is utilized for the Random Forest and Neural Networks, as well as for data cleaning, tuning of the parameters and data visualization. R is also used to add the data protection methods (Sampling and noise) on the dataset. Packages such as *scikit-learn (*Pedregosa et al., 2011) is used for Random Forest and Neural Network utilize the *tensorflow* (Géron, 2022) packages.

4.2 Data Protection Implementation

The data protection methods that this thesis focuses on is *Random Noise* and *Sampling.* These methods were applied on the dataset after it is pre-processed. The data protection methods are not applied to the entire dataset because the main purpose of the thesis is to preserve the data utility while protecting the data privacy. Hence, we will add privacy methods on the sensitive variables which we have mentioned before: ‘Age’, ‘Housing’, ‘Balance’, ‘Marital’ and ‘Education’. As the variables ‘Marital’ and ‘Education’ were one-hot encoded, there are more than one variable (column) within each variable. We choose only one variable (column) within each to add privacy methods. And in this case, they are respectively ‘education_tertiary’ and ‘Marital_married’. The details of how the privacy methods are applied on these sensitive variables are further discussed in the following section.

Sampling

The Sampling strategy employed on the dataset involved a method known as 'Sampling with replacement.' This approach entails the random selection of observations from the dataset, with each selected observation being returned to the dataset, thus allowing the potential for multiple selections of the same observation throughout the Sampling process. The sensitive variables are a mix of binary and numeric variables. Within each variable, a specific percentage of values was chosen for further processing. These selected values were subsequently shuffled and then reinserted at the same indexes, ensuring that each value was replaced only once and no duplicates were introduced during this operation. The different percentage levels that we have chosen to explore is 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100%. A sample code for Sampling the variable 'Age' is shown below:

```r
# Calculate the desired sample size by multiplying the length of the age column by 1.00
sample_size <- round(length(Bank$age) * 1.00)

# Generate random indices for sampling without replacement using the seq_along function
sampled_indices <- sample(seq_along(Bank$age), size = sample_size, replace = FALSE)

# Extract the age data corresponding to the sampled indices
sampled_data <- Bank$age[sampled_indices]

# Shuffle the sampled data randomly without replacement
shuffled_data <- sample(sampled_data, replace = FALSE)

# Replace the original age values with the shuffled data at the sampled indices
Bank$age[sampled_indices] <- shuffled_data
```

Figure 4.1: Code Snippet for Sampling Process

The first line of code calculates the desired sample size as a percentage of the total length of the 'Age' variable. This percentage (100%) is one of the 10 different percentage levels we have chosen for our Sampling method comparison. The second line of code generates a vector of randomly sampled indices based on the length of the 'Age' variable, without replacement (i.e.,

the same index cannot be selected twice). The third line of code selects the values in the 'Age' variable corresponding to the sampled indices, creating a new vector called 'sampled_data'.

The fourth line of code shuffles the values in the 'sampled_data' vector, again without replacement, and stores the shuffled values in a new vector called 'shuffled_data'. Finally, the last line of code replaces the original values in the 'Age' variable corresponding to the sampled indices with the shuffled values. This was repeated for the other 4 variables and for all the percentage levels. In total, there were 10 different datasets for Sampling generated, one for each percentage level.

Random Noise

In terms of Random Noise, the same variables were used and we chose 4 noise levels: 25%, 50%, 75% and 100%. In this particular example, the nature of the variables is important. In the case of binary variables, noise was added by flipping the values to the other.

	balance	housing
1	2143	1
2	29	1
3	2	1
4	1506	1
5	1	0

Before Noise

	balance	housing
1	2.142998e+03	1
2	2.899617e+01	0
3	1.889586e+00	0
4	1.505937e+03	0
5	8.335453e-01	1

After Noise

Figure 4.2: Noise Addition Example (75%)

For example, 25% noise would mean that 25% of values in a binary variable column were flipped to the other value. If the value was '1' it would flip to '0' and vice versa. We did the same for 50%, 75% and 100%. In the case of the numeric variables, we introduced *Gaussian noise* because we could not just flip it to another value as what we did to binary variables. Gaussian noise involves introducing random values from a Gaussian (normal) distribution to each data point. The standard deviation of the distribution controls the amount of noise added.

The mean of the Gaussian noise is zero while the standard deviation we chose was 0.1. For example, in the 25% noise level datasets, Gaussian noise was added on 25% of each continues sensitive variables. In total, 4 different new datasets for noise were generated, one for each percentage level. An example of how noise was added to the 'Balance' and 'Housing' columns is shown in Figure 4.2.

In order to delve a little deeper, Table 4.1 and 4.2 explores the statistical information of the 'Housing' and 'Balance' columns before and after 75% noise was added.

Housing	**0***(No)*	**1***(Yes)*
Before Noise	19872	24645
After Noise (75%)	43706	811

Table 4.1: Total Subscription Count for *Housing* Variable

Balance	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Before Noise	0	71	449	1391	1431	102127
After Noise (75%)	-0.36	71.26	449.09	1390.80	1431.00	102127.17

Table 4.2: Descriptive Statistics for *Balance* Variable

4.3 Model Building

4.3.1 Model Validation

Data partitioning is an important technique in machine learning for training and evaluating predictive models. The goal of data partitioning is to split a dataset into subsets that can be used for different purposes, such as training a model, tuning its hyperparameters, and testing its performance.

We split our dataset into 80% training and 20% test, where we divided the data into two subsets. The first subset, which contains 80% of the data, was used for training the machine learning model. During this process, the model learns patterns and relationships between the input features and the output labels, with the goal of making accurate predictions on new, unseen data.

The second subset, which contains the remaining 20% of the data, was used to evaluate the performance of the trained model. This subset, called the test set, serves as a proxy for new, unseen data that the model has not encountered before. By evaluating the model's performance on this test set, we can get a sense of how well the model will perform on new data in the future.

It is worth mentioning that, when addressing privacy methods, we implemented them on the entire dataset before splitting it into training and test data. Our approach is based on the objectives of the thesis and our underlying assumptions. The aim is to assist companies seeking to share data with a third party, such as the Portuguese bank in our case, in finding a way to apply privacy methods to the data while still maintaining prediction accuracy. Consequently, the dataset acquired by the bank has already undergone privacy measures, and the bank needs to perform analysis and predictions using this processed dataset. We utilized the training data to train models, with the intention of observing whether the bank can achieve satisfactory accuracy when employing a distinct dataset for predictions (which should differ from our training data). To assess this, we employed a test data set. Therefore, privacy methods were not only applied to the training data but also to the test data.

4.3.2 Logistic Regression

We built a Logistic Regression model on R to predict dependent variable 'y' using all the other independent variables by using the glm function.

We used the glm() function in R to fit a binomial Logistic Regression model to the training dataset. The dependent variable 'y' was regressed on all other variables in the dataset. The family argument was set to 'binomial' to specify that a binomial Logistic Regression model should be fit. This model was saved in a new object called 'logit_model'.

To generate predicted probabilities of the outcome variable for new observations, we used the predict() function in R with the 'logit_model' object and the 'test' dataset as inputs. The type of argument was set to 'response' to specify that the predicted probabilities should be output rather than the predicted log-odds. The predicted probabilities were then saved in a new variable called 'pred_prob' in the 'test' dataset. The Logistic Regression model was used to make predictions on new data using these predicted probabilities. Next, we used the ifelse() function to create a new variable called 'pred_choice', which took the value 1 if the predicted probability for a given observation was greater than 0.5, and 0 otherwise.

The table() function was used to create a contingency table of the predicted choices versus the actual outcomes in the test dataset. This allowed us to examine the model's accuracy in predicting the outcome variable. We used the diag() function to extract the diagonal elements of the table, which correspond to the number of correct predictions (i.e., when 'pred_choice' matches the actual outcome). By summing these diagonal elements and dividing by the total number of observations in the test dataset, we obtained the hit rate (accuracy), which is the proportion of correct predictions made by the Logistic Regression model on the test dataset. This evaluation metric will be further discussed in section 4.3.5.

4.3.3 Random Forest

The first step in implementing our Random Forest model was to import the necessary libraries, including *scikit-learn* (Pedregosa et al., 2011), *Pandas* (McKinney, 2010), and *Numpy* (Harris et al., 2020). Pandas is a data manipulation library that is useful for loading, cleaning, and transforming data, while numpy provides support for numerical computations. Scikit-learn is a machine learning library that provides a variety of algorithms for different tasks. The pre-processed dataset was then loaded into a Pandas data frame.

After loading the dataset, the data was split into training and test sets using the 'train_test_split' function from the scikit-learn library. After splitting the dataset, the Random Forest model is initialized. To initialize the Random Forest model, we created an instance of the *RandomForestClassifier* class in scikit-learn. The hyper parameters were then set as these need to be set before training. In this instance, a general hyperparameter set was used in order to start the training process. These hyperparameters were then tuned afterwards and compared to the general hyperparameter set. The general set included 100 estimators or decision trees and a *random state* of 42. The *random_state* parameter sets the random seed used by the algorithm, which allows us to reproduce the results if we run the same code multiple times.

Hyperparameter tuning

Choosing the optimal hyperparameter values for a model is an important step in machine learning. Rather than learning parameters from the data, hyperparameters are set beforehand. To achieve the best results, it is crucial to tune these factors to enhance the model's performance. Tuning hyperparameters involves searching for all possible hyperparameters and evaluating the model's performance with each set. Model accuracy, generalization, and robustness can be improved by optimizing hyperparameters. Moreover, it can help prevent overfitting, which occurs when a model performs well on training data but poorly on new data due to its complexity.

There are several hyperparameters that can be tweaked to optimize the performance of a Random Forest model, including the number of trees, the maximum depth of trees, the minimum number of samples required to split a node, and the criteria for splitting nodes. One common approach to tuning these hyperparameters is grid search. In grid search, each hyperparameter on the grid is exhaustively searched over all possible combinations of the hyperparameters. By defining grid values for n_estimators (*number of trees)*, max_depth (*the maximum depth of trees)* and min_samples_split (*minimum number of samples required to split a node),* we can train and evaluate Random Forest models for all possible hyperparameter combinations. However, if the dataset is large, it would be very time consuming to use grid search. Another approach is random search, which randomly samples hyperparameters from a predefined range or distribution. This approach doesn't evaluate all possible combinations of hyperparameters like the grid search, but rather focuses on areas of the hyperparameter space that are more likely to yield good results.

This thesis focuses on using grid search for our Random Forest model since the dataset we use is relatively small. Hence, we could explore every single combination in the hyperparameter space to yield the best results. The same hyperparameters derived from the tuning were then used on the Random Forest models to train all the new datasets created with added protection methods.

4.3.4 Neural Network

Important packages and libraries such as *Keras* (Géron, 2022) and *Tensorflow* (Géron, 2022) were first installed. *Tensorflow* is designed for building and training Neural Networks for a wide range of applications such as image classification, natural language processing etc. Keras is designed to provide a user-friendly interface for building and training deep learning models. Keras can run on top of TensorFlow and provides a simple and intuitive interface for building models.

After loading the dataset, the data was split into training and test sets using the 'train_test_split' function from the scikit-learn library. After splitting the dataset, the model was defined using the *tf.keras.Sequential* class, which allows for building a model by stacking layers sequentially. Considering the risk of overfitting when applying an excessive number of layers to a relatively small dataset, such as our data, we made the decision to construct an architecture comprising three layers: an input layer, a hidden layer, and an output layer. The number of layers is an initial parameter, that we are going to tune, which will be discussed in the later section.

The input layer has 100 neurons with a ReLU activation function and an input shape of 47, indicating the total number of variables in the dataset. The 100 neurons were initially randomly chosen and we planned to fine-tune this parameter in subsequent steps. The input shape is determined by the dataset's structure and variables. It is equal to the number of variables in our dataset, without the dependent variable, which in this case is 47. The hidden layer, which takes a value between the number of neurons in the input layer and the number of neurons in the output layer, has 50 neurons with a ReLU activation function. Since this is a classification problem, the output layer has a single neuron with a sigmoid activation function. The sigmoid function is commonly used in Neural Network for binary classification.

The training data gets processed through an epoch. The number of epochs typically refers to the number of times the entire training dataset has been processed by the model during training. In other words, one epoch corresponds to one complete pass through the entire training dataset. The number of epochs is typically a hyperparameter that needs to be tuned in order to achieve good performance on a given task. Setting too few epochs can result in underfitting, where the model does not learn enough from the data to make accurate predictions. On the other hand, setting too many epochs can result in overfitting, where the model becomes too specialized to the training data and performs poorly on new, unseen data. There are also two *dropout* layers with a rate of 0.5 applied after each of the dense layers to help to prevent overfitting.

Dropout layer

A dropout layer is a regularization technique that is used in deep learning models to prevent overfitting. The model can be simplified and improved by this method, thereby reducing its complexity and improving its generalization abilities. Dropout involves randomly dropping out (i.e., setting to zero) a fraction of neurons in a layer during training. As a result, the network is forced to learn more robust and useful features since no single neuron can always be present. The weights of all neurons are scaled during prediction to account for the dropouts during training (Srivastava et al., 2014). An example of how the dropout layer is implemented in our thesis is shown in Figure 4.3.

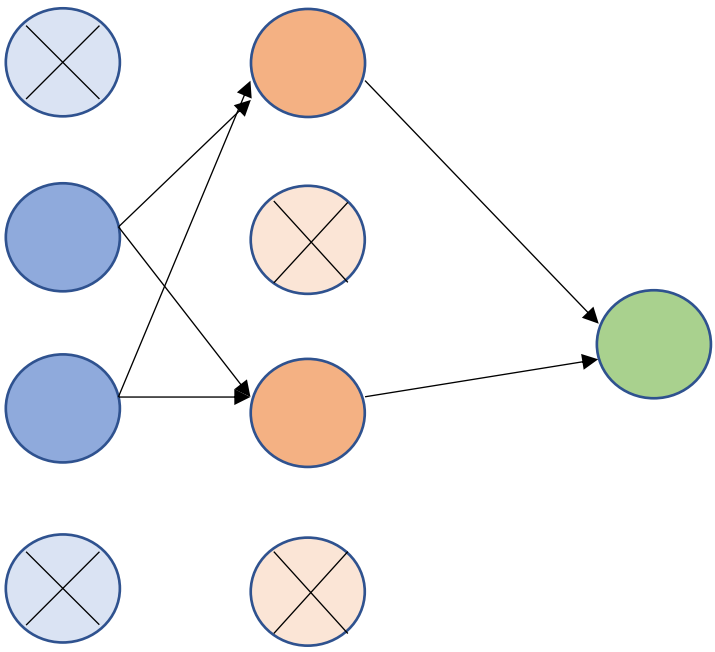

Figure 4.3: Dropout layer Implementation

The figure shows a hypothetical example with four neurons dropout rate of 0.5. which is the same rate we have used in our thesis, before tuning. There are two dropout layers, one for the input layer and one for the hidden layer each with a dropout rate of 0.5. This means that during training, 50% of the neurons in each of the dense layers will be randomly dropped out. The value for the dropout rate is also something that requires to be tuned in further steps and we will discuss this in the following section.

Hyper parameter tuning

As in Random Forest, hyperparameter tuning is an important step in the process of developing and training Neural Networks. The hyperparameters we focused on training in this thesis are the number of layers, the number of neurons in each layer as well as the number of epochs. The method we used for the hyperparameter is hyperband tuning. This tuning algorithm combines random search with adaptive resource allocation, allowing it to achieve better results than other hyperparameter tuning algorithms in a shorter amount of time.

In Hyperband tuning, hyperparameter configurations are randomly sampled and run for a small number of epochs. Some configurations are discarded based on their performance while others are promoted to the next round. Every round, the remaining configurations are run for a greater number of epochs, and the process is repeated until only the best configuration remains (Bhardwaj et al., 2020). Using this process, the best number of hidden layers, number of neurons, and epochs were generated, giving rise to the optimal Neural Network model. The same hyperparameters derived from the tuning were then used on all the datasets to see the effects of the data protection methods.

4.3.5 Prediction Evaluation Metrics

In order to evaluate the performance of our model we have explored two major metrics: *Accuracy/Hit Rate* which is generated from a *confusion matrix* and the *Receiver Operating characteristics (ROC).*

An effective method for assessing a classification model's effectiveness is a confusion matrix. This table displays the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) in a classification task. The number of cases where both the actual class and the predicted class are positive is known as true positives (TP). The number of occasions where the real class is negative but the predicted class is positive is known as false positives (FP). The number of cases where both the actual class and the predicted class are negative is known as true negatives (TN). The frequency of occasions where the actual class is positive while the predicted class is negative is known as false negatives (FN). The locations where the model performs well and poorly can be determined using a confusion matrix. Table 4. shows an example of how a confusion matrix would look:

	Actual Positive	**Actual Negative**
Predicted Positive	TP	FP
Predicted Negative	FN	TN

Table 4.3: Confusion Matrix

In the context of a confusion matrix, accuracy is a performance metric that measures the overall correctness of a classification model. It represents the proportion of correctly predicted instances out of the total number of instances in the dataset. The numbers of true positives and true negatives are added together, and this number is divided by the total number of instances to determine accuracy from a confusion matrix.

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

The goal is to maximize accuracy which would represent a positive performance of the model. However, there is one thing to consider especially in relevance to this thesis and our dataset. As mentioned in Section 3.2, our data is imbalanced and due to this, we cannot depend solely on accuracy to see how well our models perform. It would be beneficial to explore other metrics to provide a holistic view of model performance.

For binary classification tasks, *AUC-ROC ((Area Under the Receiver Operating Characteristic curve))* is a widely used evaluation metric in machine learning. A classification model's performance is assessed by the AUC-ROC, evaluating its ability to differentiate between positive and negative instances. An AUC-ROC metric summarizes how well the model performs across multiple classification thresholds. In order to determine the true positive rate (sensitivity), the Receiver Operating Characteristic (ROC) curve is plotted against the false positive rate (1-specificity) at different classification thresholds. "Sensitivity and specificity, are defined as the number of true positive decisions/the number of actually positive cases and the number of true negative decisions/the number of actually negative cases (Park et al., 2004)." The AUC-ROC is calculated as the area underneath this curve.

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

$$FPR(False\ Positive\ Rate) = 1 - Specificity = 1 - \frac{TN}{TN+FN}$$

Using the AUC-ROC value helped gain a much better insight into our models' performance due to the presence of our imbalanced dataset. AUC-ROC metric is less influenced by imbalanced datasets, where one class is significantly more prevalent than the other. In such cases, the AUC-ROC metric tends to provide a more reliable and robust evaluation compared to accuracy. It assesses the model's ability to rank instances correctly regardless of class distribution. Generally, an AUC of 0.5 suggests no discrimination, 0.7 to 0.8 is acceptable, 0.8 to 0.9 is excellent, and more than 0.9 is outstanding (Mandrekar, 2010).

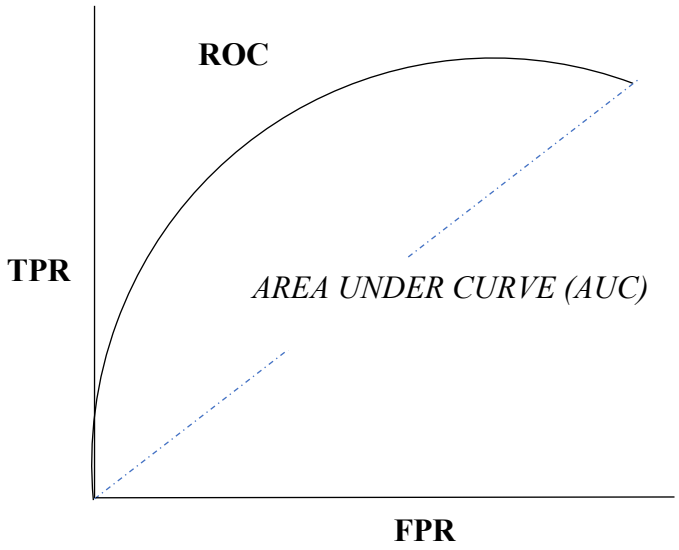

Figure 4.4: ROC Curve

There are two widely used statistical criteria for selecting and comparing models which were considered as well for the evaluation of the models used in this thesis: AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion). AIC measures the trade-off between model fit and complexity in order to select the model with the best balance between goodness of fit and parsimony. A similar criterion is BIC, which incorporates a penalty for model complexity. In both AIC and BIC, lower values indicate better models. Random Forest models do not typically use AIC or BIC, which are commonly used for model selection and comparison in statistics. The reason is that AIC and BIC are derived from the likelihood function, which assumes that the model has a certain parametric form (Burnham & Anderson, 2004). Random Forest, on the other hand, is an ensemble of decision trees and does not have a traditional likelihood function. Neural Network also involves a substantial number of parameters and cannot be constrained to a specific parametric form which is the case for AIC and BIC.

The evaluation of machine learning models in this thesis relies on accuracy/hit rate and AUC-ROC. Initially, these metrics were utilized to identify the best-performing machine learning model. Subsequently, the selected model underwent further exploration. In order to ensure privacy protection, the original dataset was subjected to privacy methods, resulting in the creation of multiple new datasets. These datasets were used to train the best model, generating several distinct models (algorithm). The performance comparison among these models was

based solely on the hit rate (accuracy) metric. This approach aligns with the research goal of assessing the impact of various data protection methods on accuracy at different levels of protection. Thus, accuracy/hit rate was employed to compare the efficacy of different protection methods within the best model.

5. RESULTS

With the pre-processed dataset, we trained Logistic Regression, Random Forest and Neural Network models. The accuracy/ hit rate of the predictions from the three models are 68.2%, 89.6% and 88.1% respectively. Random Forest is the model that gives us the highest prediction accuracy. Neural Network is 1.5% less accurate than Random Forest.

We proceeded by tuning parameters for the most promising models which are Random Forest and Neural Network. The computer specs involved in the implementation of this process includes an Apple M1 Chip with 8-core CPU and GPU along with an 8 GB RAM and 256 GB storage. The time for tuning for Neural Networks took around an hour and the best parameters are shown in Table 5.1. The best number of hidden layers is 2 while the best number of epochs to run through is 17.

Hyper Parameter	**Optimal Value**
Number of hidden layers	2
Number of Neurons in First Layer	50
Dropout rate between First and Second Layer	0.0
Number Of Neurons in Second Layer (First hidden layer)	110
Dropout rate between Second and Third Layer	0.30
Third layer (Second hidden layer)	110
Dropout rate between Third and Final Layer	0.30
Epochs	17

Table 5.1: Best Hyperparameters for Neural Network.

The hyperparameter tuning process for Random Forest took slightly longer at about 90 minutes and the best hyperparameters are shown in Table 5.2. The Random Forest classifier has a maximum depth of 20. This specifies the maximum number of nodes allowed from the root to the deepest leaf of the tree, with each leaf having at least 1 sample. Each internal node will need at least 5 samples to be split, and there are 100 trees in the forest.

Hyper Parameter	**Optimal Value**
Number Of Trees	100
Minimum number of samples required to split an internal node	5
Minimum number of samples required to be present in a leaf node	1
Maximum depth / levels in the decision tree.	20

Table 5.2: Best Hyperparameters for Random Forest.

After tuning the parameters for the two models, the accuracy/hit rates for Random Forest and Neural Network were 90.4% and 89.2%. Though the difference between them becomes slightly less, the Random Forest is still the model that give the best prediction accuracy.

Hit Rate/Accuracy	Logistic Regression	Neural Network	Random Forest
Before parameter tuning	66.9%	88.1%	89.6%
After tuning	-	89.2%	90.4%

Table 5.3: Hit Rate/Accuracy for the Three Different Models.

AUC-ROC		
Logistic Regression	Neural Network	Random Forest
0.67	0.71	0.78

Table 5.4: AUC-ROC for the Three Different Models.

In the study, the performance of various models was also evaluated using the AUC-ROC metric. From Table 5.4 we can see that the Random Forest model outperformed the others, demonstrating superior performance in terms of AUC-ROC as well. Therefore, we chose Random Forest to train the model and make predictions after using the two different privacy methods on the data.

We used the achieved Random Forest model to explore the predicting importance of independent variables. Figure 5.1 shows the most important variables for predicting customer subscription. From the figure, we can see that the sensitive variables have high values of importance in predicting, which means that the sensitive variables we are protecting also have a significant influence on prediction accuracy. This has enhanced the value of our research. As we know, when the value of a variable changes, the more important a variable is, the higher the effect it will have on prediction accuracy.

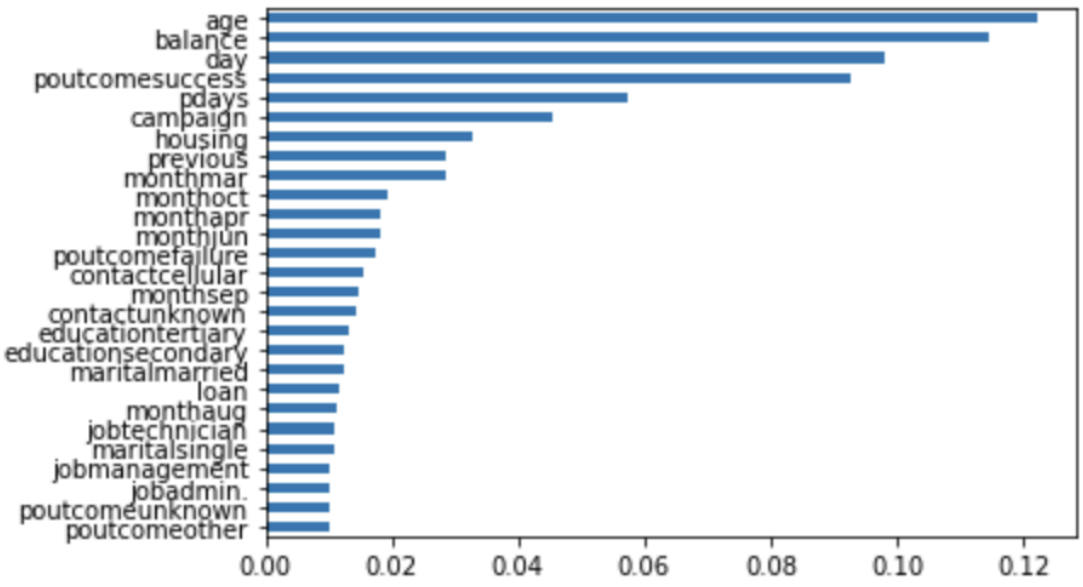

Figure 5.1: Variable Importance for Random Forest

Next, we applied privacy methods on original (pre-processed) data. As mentioned before, for the sensitive variables, we tried different percentages of each variable to add Random Noise and Sampling. When using Sampling, we experimented levels of 25%, 50%, 75% and 100% for each sensitive variable. With the Random Noise method, we experimented 10 different levels from 10% to 100%, in increments of 10%. Therefore, we got 14 new datasets in total.

Subsequently, we ran the achieved Random Forest algorithm with the same tuned hyperparameters on all the 14 datasets. Therefore, we got 14 Random Forest algorithms: 4 algorithms are trained for four different Random Noise levels, and 10 algorithms are trained for 10 different Sampling levels. The accuracies were calculated and were used as performance measures for each Random Forest algorithm. Figure 5.2 shows the accuracies.

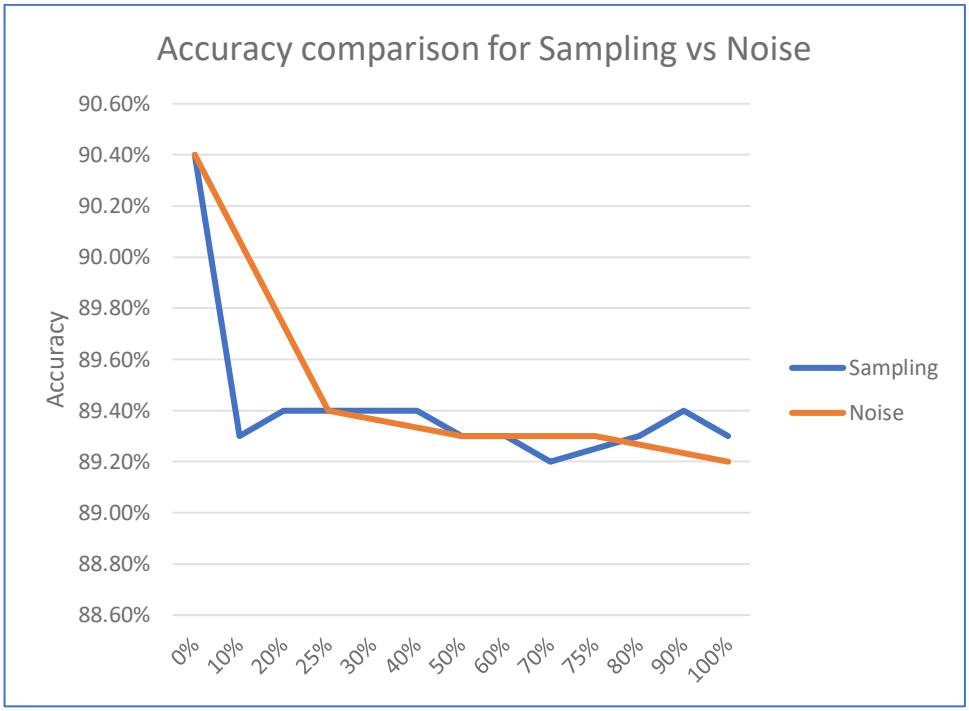

Figure 5.2: Noise and Sampling Accuracies at Different Protection Levels

From the graph above we can clearly see that before 25%, the accuracy shows decreasing trend. However, after 25%, the accuracy becomes stable and fluctuates only slightly. This trend

applies to both privacy methods. The accuracy for 100% noise is 89.2% while the accuracy for 100% Sampling is 89.3%.

6. DISCUSSION

In terms of the privacy level, as Liu et al. (2021) mentioned, there does not exist a unified privacy metric or notion. In this paper, we are not comparing each methods' privacy level but finding the best balance point for each privacy methods. Applying Random Noise to personal information variables such as 'Age', 'Balance', and 'Marital' can make it difficult to reconstruct the original data. This is because there is no access to the key that was used to add the random perturbations to the data. Moreover, even if access to the modified data was granted, the process of reconstructing the original data is computationally expensive and time-consuming. There would need to be multiple calculations and statistical analyses to estimate the original values of the data. In the case of our thesis, the noise and Sampling methods applied were random. There was no particular key or metric to which we added the protection methods. In a way, it would be close to impossible to reconstruct the data and this is exactly what is the aim in terms of preserving customers privacy.

In terms of the prediction accuracy level, the paper explored by Schneider & Iacobucci (2020) showcases the results of using data protection methods and is different from the results achieved by our analysis. Our method surpassed our expectation where we barely had a loss in accuracy levels from using data protection methods even as the percentage of protection added increased. This is different from the result achieved by Schneider & Iacobucci (2020), where they had a maximum loss in data utility using 100% Sampling whereas our model reduced by around 1% at that rate. Our hypothesis for this difference is that variations in the nature of the datasets lead to differences in the implementation of the protection method. Additionally, it should be noted that Schneider and Iacobucci's paper did not focus on machine learning implementation, which is the main topic of interest in this thesis.

To discuss the balance point, we also have to understand the needs of companies, which means the acceptable level of decreasing prediction accuracy due to adding noise to customer data. This acceptable level can vary depending on the specific needs and requirements of the company. In this specific scenario, the purpose of the prediction is marketing. In general, 89.3%

and 89.2% (the accuracies after applying privacy methods at their maximum level) are both relatively high prediction accuracies for marketing.

Therefore, we can conclude the best points to balance model accuracy and data protection for two privacy methods is when applying 100% privacy level. This paper proved that companies could protect customer data privacy while maintaining prediction accuracy. In this case, we found that the bank can use Random Forest to train the model and use Sampling or Random Noise to achieve the balance between privacy and prediction accuracy. Although this strategy is based on a specific case and companies cannot directly use the same machine learning model and privacy methods to apply on their own case to get the best balance, the paper suggests a way for companies who are trading customer data to find the best model and privacy method to achieve the balance.

7. CONCLUSION

This thesis has studied three different machine learning methods and two data privacy methods to protect customer data privacy while still maintaining good prediction accuracy and data utility. To investigate the method, we have used the marketing campaign data of a Portuguese banking institution. Specifically, the paper aims to find a method for a third party to protect their customers' data privacy when they share the data to the bank, while still maintaining the prediction value of the data. The major contribution of the paper is that we managed to maintain the data utility after using the privacy methods which are perceived as the methods that can significantly decline the quality or integrity of the information.

In order to achieve this, we first explained the machine learning models and privacy methods we chose and the reasons why we chose these methods to conduct the research in the literature review and theory sections. Based on the data we chose, we trained the models and almost maintained the prediction accuracy after using privacy methods on the original data.

The research tested several machine learning methods and privacy methods and explored the different combinations on a real-world dataset in marketing area which has barely been investigated, therefore, the results are a contribution to the research community in the marketing field to protect customer data privacy.

8. LIMITATIONS AND FURTHER RESEARCH

In addition to the results and discussion presented above, there are several other factors that could impact the choice of privacy methods for a given use case. One such factor is the size of the dataset being used. In our study, we used a relatively small dataset. It is possible that for larger datasets, the difference in prediction accuracy between the two privacy methods could become more pronounced. This is because the privacy methods are proportional to the size of the dataset, and larger datasets may require more protection to be added to achieve the same level of privacy.

In the classification of sensitive and non-sensitive variables, we conducted an investigation by researching individuals within our social circles, including family and friends. To enhance precision in this regard, future studies may consider collecting data directly, obtaining first-hand privacy preference information, and subsequently conducting classifications based on this information.

We have proved that the pre-processing data process has not cause information loss. However, this is not always the case. When using different datasets, if this process causes large amounts of information loss, then companies cannot use this way to process data and compare the three models.

Moreover, we have tried limited privacy methods, however, there are better ways to balance the trade-off between data protection and model accuracy which are tested by other literature. Therefore, further studies can build on this paper to test other methods such as differential privacy, the proposed method created by Schneider and Iacobucci (2020) and generating synthetic data explored by Gupta and Schneider (2018).

The acceptable level of decreasing prediction accuracy due to adding noise to customer data can vary depending on the specific needs and requirements of the company. Based on this case study, further research can be done to find ways to measure how much influence the 1% accuracy difference will cause for the bank, maybe try to convert the accuracy loss into cost so that the performance between different methods can be straighter for companies, which means companies can compare and make decisions based on cost difference instead of accuracy difference.

REFERENCES

Agarap, A. F. M. (2019, February 7). Deep learning using rectified linear units (ReLU) . https://arxiv.org/pdf/1803.08375.pdf

Agarwal, R. (2019, September 18). *The 5 classification evaluation metrics every data scientist must know*. Medium. https://towardsdatascience.com/the-5-classification-evaluation-metrics-you-must-know-aa97784ff226

Bhardwaj, A., Mangat, V., & Vig, R. (2020). Hyperband tuned deep Neural Network with well posed stacked sparse AutoEncoder for detection of ddos attacks in cloud. *IEEE Access*, *8*, 181916–181929. https://doi.org/10.1109/access.2020.3028690

Brown, B., Kanagasabai, K., Pant, P., & Serpa Pinto, G. (2017, March 15). Capturing value from your customer data. McKinsey & Company. Retrieved from https://www.mckinsey.com/capabilities/quantumblack/our-insights/capturing-value-from-your-customer-data

Brown, S. (2021, April 21). *Machine Learning, explained*. MIT Sloan. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

Brownlee, J. (2019, August 26). *3 ways to encode categorical variables for deep learning*. MachineLearningMastery.com. https://machinelearningmastery.com/how-to-prepare-categorical-data-for-deep-learning-in-python/

Brownlee, J. (2020, August 19). *4 types of classification tasks in machine learning*. MachineLearningMastery.com. https://machinelearningmastery.com/types-of-classification-in-machine-learning/

Burnham, K. P., & Anderson, D. R. (2004). Multimodel inference. *Sociological Methods & Research*, *33*(2), 261–304. https://doi.org/10.1177/0049124104268644

Castro, V. E., & Brankovic, L. (1999). Data Swapping: Balancing Privacy against Precision in Mining for Logic Rules. https://www.researchgate.net/publication/220802575_Data_Swapping_Balancing_Privacy_against_Precision_in_Mining_for_Logic_Rules

Couronne, R., Probst, P., & Boulesteix, A.-L. (2018, July 17). *Random Forest versus Logistic Regression: A large-scale benchmark experiment - BMC Bioinformatics*. BioMed Central. https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-018-2264-5

Dobson, A. J., & Barnett, A. G. (2018). *An introduction to generalized linear models*. CRC Press.

ETI. (n.d.). What is Differential Privacy? Retrieved from http://eti.mit.edu/what-is-differential-privacy/

Géron, A. (2022). *Hands-on machine learning with scikit-learn, keras and tensorflow: Concepts, tools, and techniques to build Intelligent Systems*. O'Reilly Media Inc.

Gimpel, H., Kleindienst, D., Nüske, N., Rau, D., & Schmied, F. (2018). The upside of data privacy – delighting customers by implementing data privacy measures. *Electronic Markets*, *28*(4), 437–452. https://doi.org/10.1007/s12525-018-0296-3

Gupta, S., & Schneider, M. J. (2018, June 01). Protecting Customers' Privacy Requires More than Anonymizing Their Data. Harvard Business Review. Retrieved from https://hbr.org/2018/06/protecting-customers-privacy-requires-more-than-anonymizing-their-data

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Harvard University Privacy Tools Project (n.d.). Differential Privacy. Retrieved from https://privacytools.seas.harvard.edu/differential-privacy

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of Statistical Learning, second edition: Data Mining, Inference, and prediction*. Springer.

Hill, M., & Swinhoe, D. (2022, November 8). *The 15 biggest data breaches of the 21st Century*. CSO Online. https://www.csoonline.com/article/2130877/the-biggest-data-breaches-of-the-21st-century.html

Holtrop, Niels, Jaap E. Wieringa, Maarten J. Gijsenberg, and Peter C. Verhoef. (2017). No future without the past? Predicting churn in the face of customer privacy. International Journal of Research in Marketing 34 (1): 154–172.

Iacobucci, D., Posavac, S. S., Kardes, F. R., Schneider, M., & Popovich, D. L. (2014). Toward a more nuanced understanding of the statistical properties of a median split. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2663427

Jain, A. (2020, July 22). *Advantages and disadvantages of Logistic Regression in machine learning*. Medium. https://medium.com/@akshayjain_757396/advantages-and-disadvantages-of-logistic-regression-in-machine-learning-a6a247e42b20

Jain, P., Gyanchandani, M., & Khare, N. (2018). Differential Privacy: Its technological prescriptive using Big Data. *Journal of Big Data*, *5*(1). https://doi.org/10.1186/s40537-018-0124-9

Jain, S.K., Kesswani, N. A noise-based privacy preserving model for Internet of Things. Complex Intell. Syst. (2021). https://doi.org/10.1007/s40747-021-00489-5

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). An Introduction to Statistical Learning.

https://static1.squarespace.com/static/5ff2adbe3fe4fe33db902812/t/6009dd9fa7bc363aa822d2c7/1611259312432/ISLR+Seventh+Printing.pdf

Kadampur, M. A., & D.V.L.N, S. (2010, January). A Noise Addition Scheme in Decision Tree for Privacy Preserving Data Mining. https://arxiv.org/ftp/arxiv/papers/1001/1001.3259.pdf

Klarreich, E. (2012, December 31). Privacy by the Numbers: A New Approach to Safeguarding Data. Quanta Magazine. Retrieved from https://www.scientificamerican.com/article/privacy-by-the-numbers-a-new-approach-to-safeguarding-data/

Kusk, M. W., & Lysdahlgaard, S. (2022). The effect of gaussian noise on pneumonia detection on chest radiographs, using Convolutional Neural Networks. *Radiography*, *29*(1), 38–43. https://doi.org/10.1016/j.radi.2022.09.011

Lehtihet, O. S., & Åryd, V. (2021). A Comparison of Performance and Noise Resistance of Different Machine Learning Classifiers on Gaussian Clusters. http://www.diva-portal.se/smash/get/diva2:1593442/FULLTEXT01.pdf

Little, Roderick J.A. 1993. Statistical analysis of masked data. Journal of Official Statistics 9 (2): 407–426

Liu, B., Ding, M., Shaham, S., Rahayu, W., Farokhi, F., & Lin, Z. (2021). When Machine Learning Meets Privacy. ACM Computing Surveys, 54(2), 1-36.

Mandrekar, J. (2010). Receiver operating characteristic curve in diagnostic test assessment. *Journal of Thoracic Oncology*, *5*(9), 1315–1316. https://doi.org/10.1097/jto.0b013e3181ec173d

Marvin, M. (2022, May 13). *New Cyber Threats & vulnerabilities brought on by the rise of IOT devices*. Portnox. https://www.portnox.com/blog/new-cyber-threats-with-the-rise-of-iot-devices/

McKinney, W. (2010). Data Structures for Statistical Computing in python. *Proceedings of the Python in Science Conference*. https://doi.org/10.25080/majora-92bf1922-00a

Mivule, Kato. (2012). Utilizing Noise Addition for Data Privacy, an Overview. 10.13140/2.1.4629.2482.

Moro, S., Cortez, P., & Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, *62*, 22–31. https://doi.org/10.1016/j.dss.2014.03.001

Olmstead, K., & Smith, A. (2017, January 26). *Americans' experiences with Data Security*. Pew Research Center: Internet, Science & Tech. https://www.pewresearch.org/internet/2017/01/26/1-americans-experiences-with-data-security/

Paratela, N. (2023, April 16). *In-depth analysis of artificial intelligence.* LinkedIn. https://www.linkedin.com/pulse/in-depth-analysis-artificial-intelligence-norton-paratela

Park, S. H., Goo, J. M., & Jo, C.-H. (2004, March 31). *Receiver operating characteristic (ROC) curve: Practical Review for Radiologists*. Korean Journal of Radiology. https://synapse.koreamed.org/articles/1027596

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. & others (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825--2830.

Phls, H. C., Mssinger, M., Petschkuhn, B., & Rcker, J. (2014). Aggregation and Perturbation in Practice: Case-Study of Privacy, Accuracy & Performance. http://henrich.poehls.com/papers/2014_PoehlsMoessingerPetschkuhnRueckert_AggregationAndPerturbationInPractice_CAMAD2014.pdf

Rahnama, H., & Pentland, A. S. (2022, February 25). *The new rules of Data Privacy*. Harvard Business Review. https://hbr.org/2022/02/the-new-rules-of-data-privacy

Ren, W., Tong, X., Du, J. et al. Privacy Enhancing Techniques in the Internet of Things Using Data Anonymisation. Inf Syst Front (2021). https://doi.org/10.1007/s10796-021-10116-w

Schneider, M. J., Jagpal, S., Gupta, S., Li, S., & Yu, Y. (2017). Protecting customer privacy when marketing with second-party data. *International Journal of Research in Marketing*, *34*(3), 593–603. https://doi.org/10.1016/j.ijresmar.2017.02.003

Schneider, M.J., Iacobucci, D. Protecting survey data on a consumer level. *J Market Anal* **8**, 3–17 (2020). https://doi.org/10.1057/s41270-020-00068-6

Schooltink, W. T. (2020). Testing the Sensitivity of Machine Learning Classifiers to Attribute Noise in Training Data. https://essay.utwente.nl/82072/1/Schooltink_BA_EEMCS.pdf

Sharma, S., Sharma, S., & Athaiya, A. (2020). ACTIVATION FUNCTIONS IN NEURAL NETWORKS. https://ijeast.com/papers/310-316,Tesma412,IJEAST.pdf

Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., Megías, D. (2017). Individual differential privacy: a utility-preserving formulation of differential privacy guarantees. IEEE Transactions on Information Forensics and Security, 12(6), 1418.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfittin. https://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf

Statista. (2023). Data usage in marketing and advertising - statistics & facts. Retrieved from https://www.statista.com/topics/4654/data-usage-in-marketing-and-advertising/#topicOverview

Tene, O., & Polenetsky, J. (2012, February 12). *Privacy in the age of big data*. Stanford Law Review. https://www.stanfordlawreview.org/online/privacy-paradox-privacy-and-big-data/

Terra, J. (2023, February 22). *Regression vs. classification in Machine Learning for Beginners: Simplilearn*. Simplilearn.com. https://www.simplilearn.com/regression-vs-classification-in-machine-learning-article

UCI Machine Learning Repository: Bank Marketing Data Set. (2014). https://archive.ics.uci.edu/ml/datasets/bank+marketing

Varma, S., & Das, S. (2018). Deep learning. https://srdas.github.io/DLBook/

Vivek, S. (2022, March 7). *Introductory guide on the activation functions*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/03/introductory-guide-on-the-activation-

Whitney, L. (2021, August 17). *Data privacy is a growing concern for more consumers*. TechRepublic. https://www.techrepublic.com/article/data-privacy-is-a-growing-concern-for-more-consumers/

Wickham et al., (2019). *Welcome to the tidyverse.* Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686