VIETNAM NATIONAL UNIVERSITY HCM CITY

UNIVERSITY OF ECONOMICS AND LAW

REPORT RESEARCH

SCIENTIFIC RESEARCH TOPIC OF STUDENTS

PARTICIPATING IN THE "YOUNG SCIENTIST UEL" AWARD IN 2022

Topic:

KNN, PCA, AND RANDOM FOREST ALGORITHMS FOR DATA IMPUTATION: A CASE STUDY WITH A VIETNAMESE COMMERCIAL BANK

Field: Economics

Specialization: Finance - banking - securities - accounting - auditing, insurance - credit

TP.HCM, April 2022

VIETNAM NATIONAL UNIVERSITY HCM CITY
UNIVERSITY OF ECONOMICS AND LAW

REPORT RESEARCH

SCIENTIFIC RESEARCH TOPIC OF STUDENTS

PARTICIPATING IN THE "YOUNG SCIENTIST UEL" AWARD IN 2022

Topic:

KNN, PCA, AND RANDOM FOREST ALGORITHMS FOR DATA IMPUTATION: A CASE STUDY WITH A VIETNAMESE COMMERCIAL BANK

Group of students

No.	Full name	ID	Faculty	Role	Phone number	Email
1.	Phan Thị Nhị	K194131685	Economic Mathematics	Leader	0965844321	nhipt19413@st.uel.edu.vn
2.	Phạm Tiến Đạt	K194131651	Economic Mathematics	Participant	0867590401	datpt19413@st.uel.edu.vn
3.	Nguyễn Thị Mai	K194131672	Economic Mathematics	Participant	0397923546	maint@st.uel.edu.vn
4.	Nguyễn Lê Minh Hằng	20050021	International Business and Economics	Participant	0934463550	20050021@vnu.edu.vn

TP.HCM, April 2022

DECLARATION

We hereby commit that the research paper titled "KNN, PCA, and Random forest algorithms for data imputation: A case study with a Vietnamese commercial bank" is our research work. All references are completely cited and clearly documented.

Group of authors

Abstract: *Missing data is a very common and critical aspect of the data world, especially for real-life applications. Missing relevant data or information can introduce bias in parameter estimation, affect statistical inferences and related works.*

In this study, we presented several approaches to deal with missing data, such as Mean/median Imputation, Regression Imputation, K Nearest Neighbors (KNN) and Random Forest (RF), etc. After that, we worked on a case study: comparing some common imputation methods (KNN, Random Forest, PCA) on a Vietnamese commercial bank data. As a result, our imputed values were very close to the original observations. In addition, the three methods helped to improve customer classification problems, much better than simple median imputation - which was widely used.

We've experimented with these methods before, but with tiny data, whereas this research is based on a dataset of other commercial banks with a considerably greater size. When compared to the prior data set, the methods utilized in this data set produced consistent results, indicating that the way of filling in the data used is feasible.

Overall, we contributed a synoptic review on data missing imputation; a comparison of some common methods and we applied the imputed result to classify good or bad customers in Vietnamese credit data. The imputation methods provided great support for many other situations, where our data had missing values.

Keywords: *missing data; data imputation; KNN; random forest; PCA.*

Tóm tắt: *Khuyết dữ liệu là một khía cạnh rất phổ biến và quan trọng trong thế giới dữ liệu, đặc biệt là đối với các ứng dụng trong cuộc sống thực. Dữ liệu hoặc thông tin liên quan bị khuyết có thể dẫn đến sai lệch trong ước lượng tham số, ảnh hưởng đến các suy luận thống kê và các công việc liên quan.*

Trong nghiên cứu này, chúng tôi đã trình bày một số phương pháp tiếp cận để xử lý dữ liệu bị khuyết, chẳng hạn như Mean/Median Imputation, Regression Imputation, K Nearest Neighbors (KNN), and Random Forest (RF), v.v. Sau đó, chúng tôi thực hiện một nghiên cứu điển hình: so sánh một số phương pháp áp đặt phổ biến (KNN, Random Forest, PCA) trên dữ liệu của một ngân hàng thương mại Việt Nam. Kết quả là các giá trị được điền khuyết của chúng tôi rất gần với các quan sát ban đầu. Ngoài ra, *ba phương pháp đã giúp cải thiện các vấn đề phân loại khách hàng, tốt hơn nhiều so với phương pháp điền khuyết trung vị đơn giản - vốn được sử dụng rộng rãi.*

Chúng tôi đã thử nghiệm các phương pháp này trước đây với dữ liệu nhỏ. Tuy nhiên, với nghiên cứu này, nhóm tiến hành thử nghiệm trên tập dữ liệu của một ngân hàng *thương mại khác với quy mô lớn hơn đáng kể. Và khi so sánh với kết quả của những tập dữ liệu trước đó, các phương pháp được sử dụng trong tập dữ liệu này tạo ra kết quả nhất quán, cho thấy rằng cách điền vào dữ liệu được sử dụng là khả thi.*

Nhìn chung, chúng tôi đã đóng góp một đánh giá khái quát về việc cung cấp dữ liệu bị khuyết; so sánh một số phương pháp phổ biến và áp dụng kết quả này để phân loại khách hàng tốt hay xấu trong dữ liệu tín dụng Việt Nam. Đặc biệt, các phương pháp này sẽ là một sự hỗ trợ tuyệt vời cho nhiều trường hợp khi dữ liệu bị khuyết giá trị.

Từ khóa: *missing data; data imputation; KNN; random forest; PCA.*

CONTENTS

LIST OF ACRONYMS

No.	Acronym	Meaning
1	MCAR	: Missing Completely at Random
2	MAR	: Missing at Random
3	MNAR	: Missing not at Random
4	KNN	: K Nearest Neighbors
5	PCA	: Principal Component Analysis
6	DIN	: Individual credit customers
7	DCO	: Corporate credit customers
8	Base_bal	: Base Balance
9	Duno_QD	: Debit balance
10	MIPCA	: Multiple PCA imputation
11	MIRAN	: Random Forest imputation

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: OVERVIEW

1.1. Motivation

Missing data is a highly critical aspect of data analysis, especially for applications in economics, finance, health sciences, etc. A lot of researchers, organizations and companies have to deal with this in practice (Ayilara et al., 2019; Florez-Lopez, 2010).

Many studies have investigated alternative PCA techniques in the lack of values in a collaborative fill-in task like the Netflix challenge. Filling in has been getting a lot of attention lately. It includes movie ratings given by 480189 customers for 17770 movies. There are 100480507 ratings from 1 to 5 given and the task is to predict 2817131 other ratings in the same customer group and movie. 1408395 ratings are intended for validation (or exploration). Note that 98.8% of the value is missing. The PCA method is one of the most popular techniques considered by Netflix contestants. (Tapani, 2010)

Missing data may be caused by record linkage, malfunction of the measuring system, badly designed surveys and beyond. Missing relevant data or explanatory information can bring strong bias in the estimation and influence to statistical inferences and more complex consequences.

To analyse the missing data sets, researchers sometimes simply delete all missing rows. However, removing missing data is not always a good idea since important information can be lost and inferences become distorted. For more sophisticated solutions, researchers are going to determine the best substitutions for missing values (Buuren, 2021).

Previous researchers have used a number of methods to fill in data such as KNN (K-nearest neighbour) imputation (Kowarik & Templ, 2016), random forest (Stekhoven, 2016), principal component analysis (PCA) by using a multiple correspondence analysis (MCA) model or a multiple factor analysis (MFA) model (Husson & Josse, 2020), etc. However, there was limited information on how effective these methods were for Vietnamese data. Only a few theoretical and empirical missing data studies have been done especially in the field of credit scoring.

That is why, in this work, we would like to apply the three above imputation methods to treat credit data coming from a Vietnamese commercial bank. By using the full data set, we are able to evaluate the three imputation methods. In addition, we compare their usages in two popular credit scoring models. After the experiments, we could come up with some suggestions for dealing with missing economic data, especially applicable for Vietnamese situations.

1.2. Aims of this study

- An overview of theoretical background about missing data and imputation, such as scientific literature, types of missing, popular imputation methods.

- Applying the three imputation methods for data from a Vietnamese commercial bank: K-nearest neighbour imputation (KNN), random forest, principal component analysis (PCA).

- A comparison of three methods with simple median imputation: how much they are similar to the original data and how they contribute to the output of credit scoring models. In this study, we use two popular classification methods: logit, random forest and xgb.

- Some suggestions for dealing with missing data in Vietnamese banks and financial institutions, especially while working with credit data.

1.3. Scope and Subject

Subject: Recognizing that the customer’s data (especially concerning credits) often have missing values, here our subjects are customers from financial institutions.

Scope: For this study, we focus on 95219 bank customers (both personal and corporate) from a Vietnamese commercial bank. In this set, we have information from their loan applications. Among them, the bank decided that there are 84932 good profiles and 10287 bad profiles.

1.4. Study methods

For the literature review, we filter and choose some highly cited relevant papers and works from Web of Science (Clarivate, previously Thomson and Reuters), Google Scholar with the keywords containing “missing data” and “imputation”.

The data for this research were collected from a Vietnamese commercial bank. After that data were explored and prepared in the suitable format for further analysis.

For empirical study and evaluation of imputation methods, we randomly remove some values from the full original data, and use three methods (KNN, Random Forest, PCA) to find their best estimations. All the imputed data were balanced and then used as inputs for two classification models (Logit, Random Forest and Xgb) to predict the good or bad customer profiles.

After that, we compare the imputed data with original data and the popular median imputation. There are two levels of comparisons. Direct comparison uses distance functions and objective-oriented comparison measures how precisely the imputed data could predict the good or bad customers compared to the original data.

All the implementation for data preparation, imputation methods and classification models have been done by using Python and R languages.

1.5. Contents and our contributions

This report consists of five chapters. After their description, the summary of our contributions are also given.

Chapter 1: *Overview*

The first chapter covers our motivation, the aims of the research, subject and scope, structure of the content and our contributions.

Chapter 2: *Theoretical background*

The second chapter provides a literature review on missing data imputation, three types of missing data and some well-known methods for handling them. Among them, we take three popular methods for further study.

Chapter 3: *Data preprocessing*

The third chapter is mainly for data description and relevant steps to prepare the suitable data. This part includes feature selections, balancing methods, etc.

Chapter 4: *Implementation and results*

This fourth chapter presents the details of our case study with three imputation methods: KNN, Random Forest, PCA. We provide here the criteria we would use for direct comparison and for objective-oriented comparison; as well as the corresponding output of all methods.

Chapter 5: *Discussion*

The last chapter contains the discussion of the output given in our previous chapter. We also suggest a potential procedure could be considered when dealing with missing values, especially for Vietnamese credit data.

Our contributions: In this study we contribute an overview on missing data literature; then we choose and apply three popular methods for Vietnamese data and compare the outcomes to see which one is the best in terms of data distance and in terms of credit scoring performances. As a result, for Vietnamese credit data, the three methods provide relatively good outputs, however the KNN - K nearest neighbors imputation is slightly better than Random Forest and PCA methods concerning credit scoring outcomes. All three methods are much better than the simple median imputation.

From our experiments, it is suggested that in the future research involving missing data, these above imputation methods can be used as reliable ways to estimate the un-available information. Especially for credit scoring problems, we have already tested with a typical data set. We are going to test some further Vietnamese data, in order to provide more empirical evidence for imputation results.

CHAPTER 2: THEORETICAL BACKGROUND

2.1. Theory of Credit Scoring

A credit scoring model is a mathematical model that is used to evaluate the probability of default, or the likelihood that a credit event will occur (i.e. bankruptcy, obligation default, failure to pay, and cross-default events). The chance of default is generally reported in the form of a credit score in a credit scoring model. The greater the score, the lesser the risk of default.

Although credit scoring models include a number of common credit variables, different types of loans may use distinct credit factors that are specific to the loan features. Payment history, age, number of accounts, and credit card utilization, for example, are credit factors for a credit card loan; credit considerations for a mortgage loan include down payment, job history and loan size.

Credit scoring algorithms that are accurate and predictive can assist a financial institution maximize its risk-adjusted return. During economic cycles such as recessions or booms, however, markets and consumer behavior can shift quickly. As a result, risk managers and credit analysts must not only construct but also promptly adjust and validate their models.

2.2. Literature reviews on missing data

Dealing with missing data is a critical problem in practice. While there are quite many foreign groups who have had valuable theoretical and empirical research, in Vietnam there are only a few relevant studies on this topic. In the following we highlight some relevant scientific papers.

- Ilin, A. & Raiko, T. (2010), Practical Approaches to Principal Component Analysis in the Presence of Missing Values. The author studied the problem of missing data and handled them using the Principal component analysis (PCA) method. In this study, the author pointed to a probability formula of the PCA, which provides a good foundation for dealing with missing values. Through the study, the author showed the algorithm's applicability in the Netflix problem.

- Ayilara, O. F., et al. (2019), Impact of missing data on bias and precision when estimating change in patient-reported outcomes from a clinical registry. In this study, the author studied the problem of patient-reported lack of information (PRO), which leads to deviations in estimating changes in results. Missing data can significantly affect the accuracy of the estimated change in PRO points from clinical registration data and the addition of information to MI models can increase accuracy and reduce deviations.

- Tang, F. & Ishwaran, H. (2016), Random forest missing data algorithms. Processing of missing data used by the author in this study is the Random forest (RF) method. The results show that RF is a machine learning method with high predicted performance and capable of processing a variety of data types.In addition, the author compared the RF and KNN methods in this study to better understand why the RF approach is more effective in low to medium correlation. This adaptability of RF can play a special role in exploiting correlations in data that may not necessarily be in other methods.

- Husson, F. (2012), Handling missing values in exploratory multivariate data analysis methods. This study refers mainly to the PCA method, the author first proceeded to describe the algorithm, then provided information about the variance and parameters, missMDA is a R package that implements recommended methods. The results show that the approaches are competitive, the PCA-based multiple fill method can be considered an alternative to other multiple filling methods.

- Kang, H. (2013), The prevention and handling of missing data. In this study, the author suggested that the missing data could reduce the statistical power of a study and could generate false estimates, leading to invalid conclusions. In addition, the author also generalized the types of data loss as missing completely at random (MCAR), missing at random (MAR) và missing not at random (MNAR). The techniques for processing missing data mentioned in this study are listwise or case deletion, pairwise deletion, mean substitution, regression imputation, expectation-maximization, multiple imputation, ect.

- Stekhoven, D. J. & Bühlmann, P. (2011), MissForest-nonparametric missing value imputation for mixed-type data. The author of this paper introduced the MissForest approach for properly processing missing values, particularly in data sets with various types of variables and multi-dimensional data. Furthermore, the author has demonstrated that MissForest outperforms alternative imputation approaches through comparison.

- Gajawada, S. & Toshniwal, D. (2012), Missing Value Imputation Method Based on Clustering and Nearest Neighbours. Based on the K-Means technique and nearest neighbors, the author developed a method to fill in the missing values in this study.

Furthermore, this approach has been tested on clinical data sets, with the findings indicating that the suggested method is more effective than basic procedures, albeit it is not advised for all incorrect data sets.

- Ispirova, G., Eftimova, T. & Seljaka, B. K. (2020), Evaluating missing value imputation methods for food composition databases. The absence of data in the Food Ingredients Database was addressed in this study (FCDB). The authors concentrated on missing data entry approaches based on the statistical prediction method of replacing missing values: Non-Negative Matrix Factorization (NMF), Multiple Imputations by Chained Equations (MICE), fill in asymmetrically missing data with Random Forest (MissForest) or K-Nearest Neighbors (KNN) and compare to frequently used techniques Mean/ Median values. The findings demonstrate that current impose methods produce better outcomes than older impose approaches.

- Minakshi, Rajan & Gimpy, V. (2014), Missing Value Imputation in Multi Attribute Data Set. The authors used a variety of methods to process lost values in this study, including Litwise deletion, Mean/Mode imputation, and KNN, and then conducted comparisons, evaluations, and conclusions, concluding that imputation and KNN methods were more accurate than the other two techniques. With faulty values, KNN imputation is a useful method.

- Buuren, S. van & Groothuis-Oudshoorn, K. (2010), mice: Multivariate Imputation by Chained Equations in R. This study mentions multi-variable data entry according to MICE V2.0, automatic predictor selection, data processing, processing after processing entered values, specialized synthesis, and model selection. In addition, it provides a practical, step-by-step approach to using mice to solve incomplete data problems in real data. The benefit is that the good tools in data visual display are lost, while the negative is that the application theory is poor and unclear.

- Newman, D. A. (2014), Missing Data: Five Practical Guidelines. This study shows that in order to enhance the best statistical method while optimizing the balance between ease of implementation and the degree of possible data deviations and deviations, the author gives five practical instructions and the decision tree for processing the missing data. If followed these practical principles will represent a significant step forward in eliminating missing data variations and variances. The benefit is that there are many fresh recommendations, which are better than those in past study articles.

- Pedersen, et al. (2017), Missing data and multiple imputation in clinical epidemiological research. The author discussed the advantages and disadvantages of different types of missing data, as well as methods that are commonly used to process missing data during the analysis phase, and their flaws. They also introduced multiple imputation as an alternative method, recognizing its advantages over "traditional" methods in clinical epidemiological research. The benefits of multiple imputation over other methods for addressing missing data are that they give impartial and accurate estimates of connections based on existing data. This technique, however, has an impact not just on cost estimates for variables with missing data, but also on estimates for variables with no missing data but no recommended resolution.

- Young, Rebekah; Johnson, David R. (2015), Handling Missing Values in Longitudinal Panel Data With Multiple Imputation . Provides a useful overview of major challenges and methods for vertically analyzing table data to manage missing values using Multiple Imputation. Multiple Imputation, according to the author, may allow the researcher to evaluate all available data.

- Enders, Craig K. (2017), Multiple imputation as a flexible tool for missing data handling in clinical research. A mixture of categorical and continuous variables, missing item-level data in surveys, meaningful testing, interaction effects, and multi-level missing data are some of the practical issues clinical researchers are likely to face when using multiple imputation, according to the study. In addition, the author shows how to use multiple imputation in existing software to encourage the use of this approach in a variety of practical research. Multiple imputation is a superior technique for behavioral science data because psychological data sets are typically complicated and difficult to handle within the existing framework of possibilities.

2.3. Types of missing data

There are three types of data deficiency: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR) (Buuren, 2021). The most common form of data loss, MCAR, occurs when a variable is missing data independently of all observations of variables, and has little to do with the variables in that data column. With

MAR, the missing data pattern doesn't depend on its own column, but they link to observed data in other columns; we can hope to estimate the missing values by the available data. With MNAR, the missingness in each column depends on that column itself and other columns as well.

Table 2.1: Example on missing data mechanisms

Age	Income	M_{Age}	M_{Inc}
24	1500	1	1
19	NA	1	0
29	4200	1	1
68	NA	1	0

We want to explain the Income according to the Age. There are missing values in the Income:

- If the observations are MCAR, the missingness of the Income does not depend on the Age and the Income.

- If the observations are MAR, the missingness of the Income does not depend on the Income. For example, it occurs if young and old people are less likely to give their incomes.

- If the observations are MNAR, the missingness of the Income depends on the Income itself and may depend on the Age. A possible interpretation is that very rich or poor people are less likely to give their incomes.

Missing data mechanisms: A toy example

Denote

$Y \in \{0, 1\}$: indicates presence of a feature, sometimes missing

$X \in \{A, B\}$: population groups, always observed

$M \in \{0, 1\}$: missingness pattern for Y

Full data: $P(M = 1|x, y) = 1$, $P(M = 0|x, y) = 0$

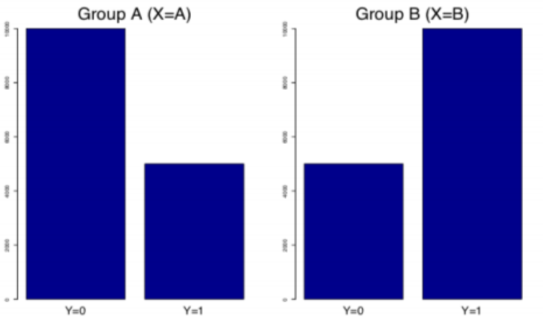


Figure 2.1: Example about missing value

- A toy example: Missing Completely At Random $P(M = 0|x, y) = P(M = 1) = 0.2$

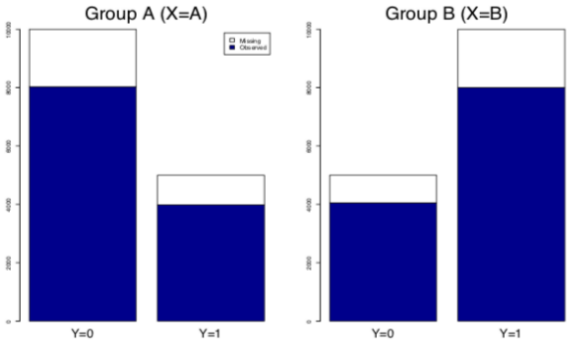


Figure 2.2: Example about MCAR

- A toy example: Missing At Random

$P(M = 0|x, y) = P(M = 0|x) = 0.2 \times 1(x = A) + 0.6 \times 1(x = B)$

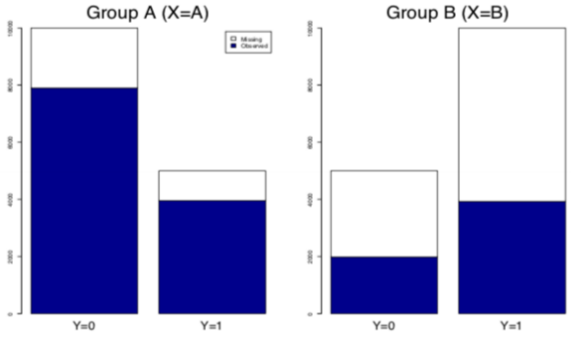


Figure 2.3: Example about MAR

- A toy example: Missing Not At Random

$P(M = 0|x, y) = P(M = 0|y) = 0.2 \times 1(y = 0) + 0.8 \times 1(y = 1)$

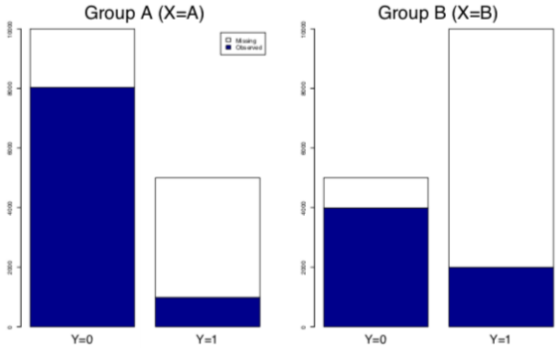


Figure 2.4: Example about MNAR

2.4. Methods of handling missing data

2.4.1. Complete - case analysis

With this method, we will delete faulty data, disregard them, and only use the remaining values of the data set to conduct study and evaluation.

This is a very simple approach, and it's cheap for dealing with missing values. However, the result is only accurate when the missing value ratio is small, the sample size is large enough to test and is MCAR or MAR. In many real-life situations, simple deletion affects the sample and decreases the reliability of the study results. It is totally unacceptable for data sets that lose a large amount of information. Particularly, where missing values are critical details, the study's findings can lead to deviations and fail to represent fact, which is a very dangerous issue. Therefore, depending on the area, considering a suitable method with high efficacy is important.

Example: Suppose we are interested in estimating the median income of some population. We send out an email asking a questionnaire to be completed, amongst which participants are asked to say how much they earn.

Gender	**Age**	**Income**	**…**
F	25	60000	…
M	?	?	…
?	30	?	…

=>

Gender	Age	Income	…
F	25	60000	…
~~M~~	~~?~~	~~?~~	~~…~~
~~?~~	~~30~~	~~?~~	~~…~~

F	?	150000	
…	…	…	…

F	?	150000	…
…	…	…	…

However, only a few proportion of the target sample returns the questionnaire, so we have missing incomes for the remainder . If those who returned an answer to the income question have systematically higher or lower incomes than those who did not , the median income of the complete cases will be biased.

2.4.2. Simple imputation

Missing values will be replaced with the sample means of the remaining values using the Mean Imputation process. Median imputation, Mode imputation and Zero imputation are also used in processing.

Using sample means to replace the missing values is an easy and popular technique. However, it has significant drawbacks, which may lead to a downward trend for variance. As a result, the use of average values is inappropriate, especially when the number of missing values accounts for a large percentage of the sample. This bias leads to an underestimation of the prepared variance, which affects the association between variables, changes the shape of the distribution, and leads to incorrect conclusions. Hence, this approach is only suitable for situations where the missing values are entirely random and account for a small percentage of the total.

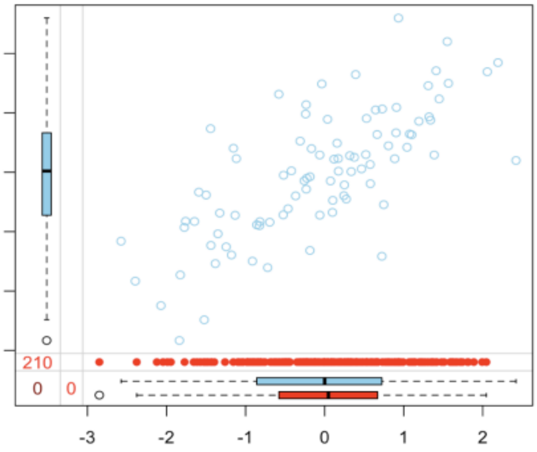
Example: Consider n couples (X_1, Y_1), . . . , (X_n, Y_n) where $X_i \sim$ N (μ_X, σ_X^2) and Y$\sim$ N (μ_Y, σ_Y^2). 70% of missing entries completely at random on Y.

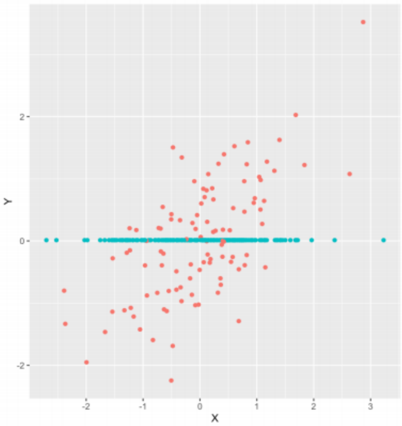
Simulated data: n = 300, $\mu_X = \mu_Y = 0$, $\sigma_X = \sigma_Y = 1$, $\rho_{XY} = 0.7$

Age	Income
25	60000
?	?
51	?
?	150300
…	…

=>

Age	Income
25	60000
$\widehat{\mu^1_{Age}}$	$\widehat{\mu^1_{Income}}$
51	$\widehat{\mu^1_{Income}}$
$\widehat{\mu^1_{Age}}$	150300
…	…





$\mu_Y = 0, \hat{\mu}_Y = 0.02$

$\sigma_Y = 1 \; \hat{\sigma}_Y = 0.58$

$\rho_{XY} = 0.7 \; \hat{\rho}_{XY} = 0.43$

Figure 2.5: Mean imputation

Comment: Mean imputation preserves the mean of the imputed variable, reduces variance; standard errors of estimates from filled - in data are too small, since standard deviations are underestimated and “sample size” is overstated. Besides, it distorts the correlation with other variables, deforms joint and marginal distributions.

2.4.3. Regression imputation

Regression with the complete cases: $\hat{Y}_i = \widehat{\beta_0} + \widehat{\beta_1} X_i$, i = 1,..,a

Imputation by the prediction of the regression model: $\hat{Y}_i = \widehat{\beta_0} + \widehat{\beta_1} X_i$, i = a + 1,....,n

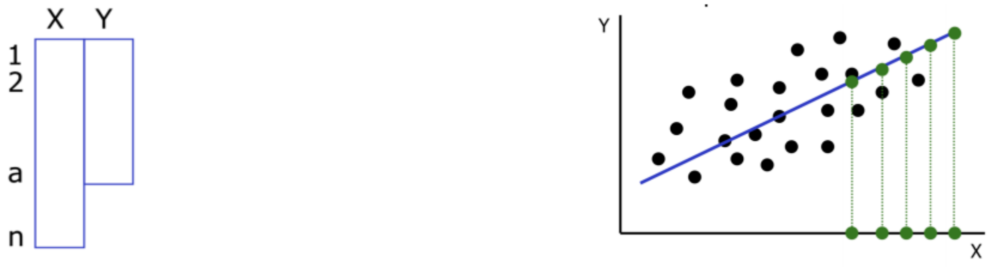
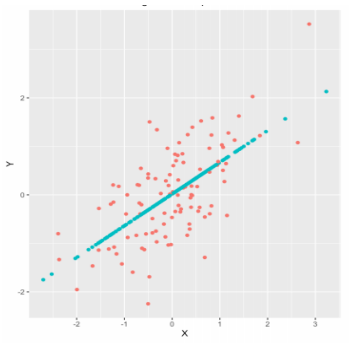


Figure 2.6: Regression imputation for missing Y



$\mu_Y = 0, \hat{\mu}_Y = 0.04$

$\sigma_Y = 1 \ \hat{\sigma}_Y = 0.81$

$\rho_{XY} = 0.7 \ \hat{\rho}_{XY} = 0.86$

Figure 2.7: Regression imputation

Comment: Imputation by regression takes into account the relationship variance underestimate and correlation overestimate.

2.4.4. Stochastic regression imputation

Estimate the coefficients β_0, β_1 and the variance σ^2, then impute from the predictive *Yi* $\sim N(\widehat{\beta_0} + \widehat{\beta_1} X_i; \widehat{\sigma}^2)$.

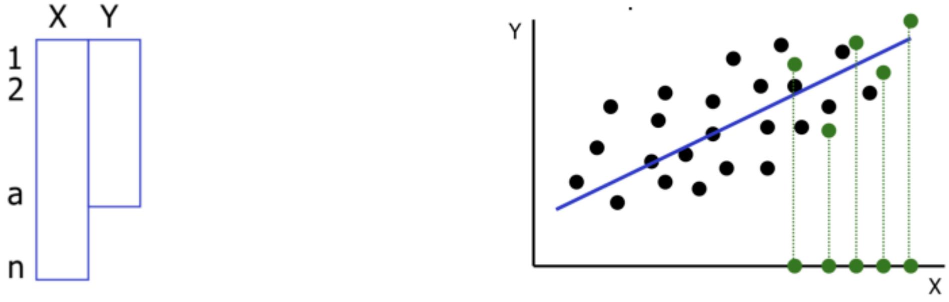
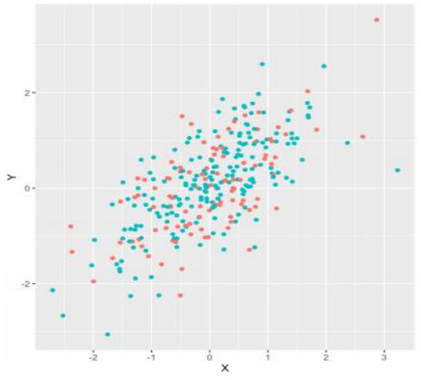


Figure 2.8: Stochastic regression imputation for missing Y

Regression with the complete cases: $\hat{Y}_i = \widehat{\beta_0} + \widehat{\beta_1} X_i$, i = 1,...., a and $\hat{\sigma}^2$

Imputation by the prediction of the regression model:

$$\hat{Y}_i = \widehat{\beta_1} + \widehat{\beta_2} X_i + \epsilon_i, \text{ i = a + 1,...., with } \epsilon_i \sim N(0; \hat{\sigma}^2)$$



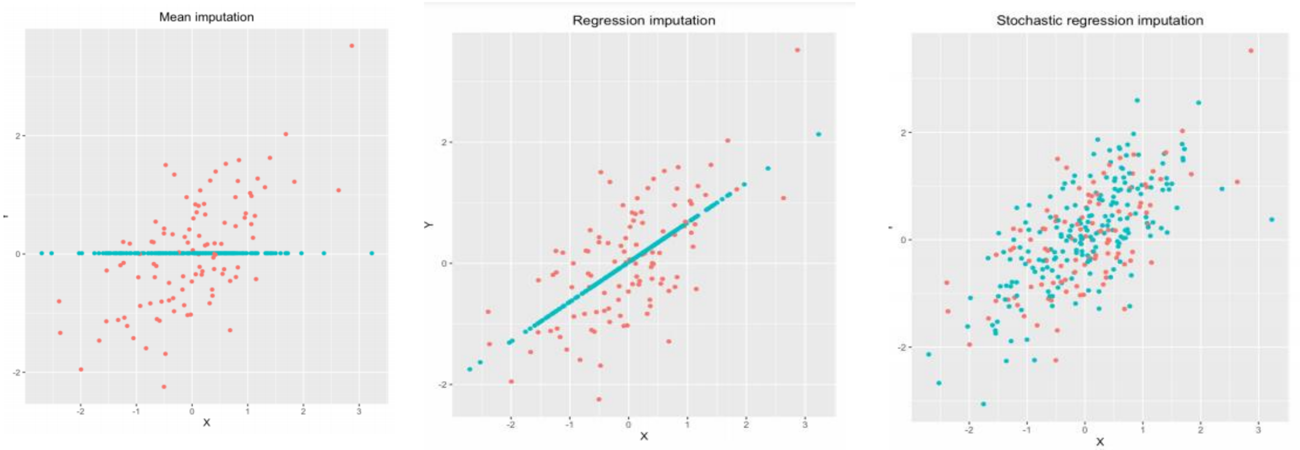
$\mu_Y = 0, \hat{\mu}_Y = 0.02$

$\sigma_Y = 1\ \hat{\sigma}_Y = 0.98$

$\rho_{XY} = 0.7\ \hat{\rho}_{XY} = 0.69$

Figure 2.9: Stochastic regression imputation

Comment: Stochastic regression imputation preserves distribution.



$\mu_Y = 0, \hat{\mu}_Y = 0.02$

$\sigma_Y = 1\ \hat{\sigma}_Y = 0.58$

$\rho_{XY} = 0.7\ \hat{\rho}_{XY} = 0.43$

$\mu_Y = 0, \hat{\mu}_Y = 0.04$

$\sigma_Y = 1\ \hat{\sigma}_Y = 0.81$

$\rho_{XY} = 0.7\ \hat{\rho}_{XY} = 0.86$

$\mu_Y = 0, \hat{\mu}_Y = 0.02$

$\sigma_Y = 1\ \hat{\sigma}_Y = 0.98$

$\rho_{XY} = 0.7\ \hat{\rho}_{XY} = 0.69$

Figure 2.10: Imputation with mean and regression

2.4.5. K Nearest Neighbors – KNN

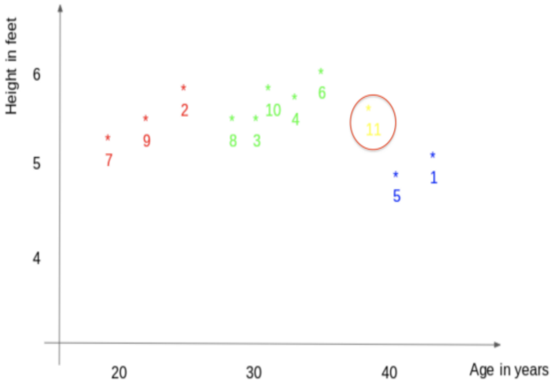
KNN imputation algorithm is a popular approach to missing data due to its simplicity and accuracy, in comparison with other methods. This classification tool is widely used in data grouping. KNN imputation improves its performance based on Mean Absolute deviation (MED) and Standard deviation (Ms.R.Malarvizhi & Dr.Antony Selvadoss Thanamani, 2012). In terms of using distance functions, missing data is treated by the most similar k given points to them.

There are significant advancements of KNN imputation algorithms. Firstly, it allows us to use both quantitative and qualitative values as estimated numbers for substituting. Secondly, cost and duration are supposed to be cut down during the process because of its predictive models – majority of them are the not-imposed. In contrast, if there is any lack of feature values, using distance function still gives us results without capability of finding them in similar calculations. Moreover, cost and duration may sharply be increased due to the amount of features.

Table 2.2: Example about KNN

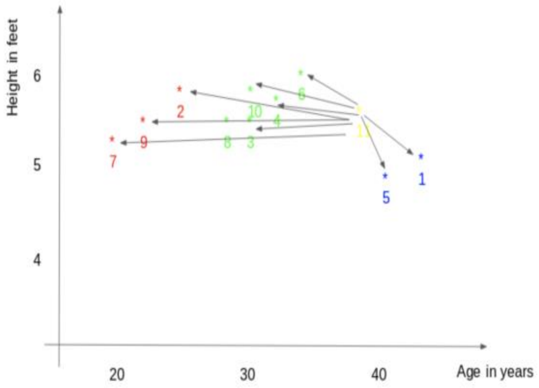
ID	Height	Age	Weight
1	5	45	77
2	5.11	26	47
3	5.6	30	55
4	5.9	34	59
5	4.8	40	72
6	5.8	36	60
7	5.3	19	40
8	5.8	28	60
9	5.5	23	45
10	5.6	32	58
11	5.5	28	?

Consider the following dataset with the weight value of ID11 is missing:

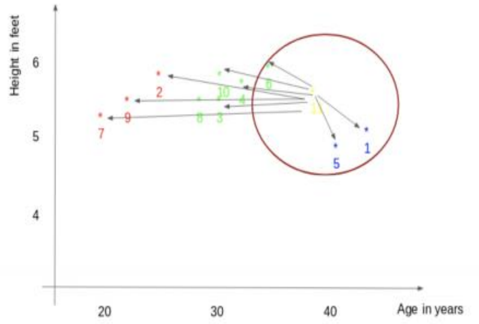


Step 1: Calculate the distance.

Euclidean distance: d(x, y) = $\sqrt{\sum_{j=1}^{p}(x_j + y_j)^2}$ for two vectors x = (x_1,…, x_p) and y = (y_1,…, y_p).



Step 2 & 3: Determine the k nearest neighbors (k closes points) based on the distance and compute the predicted value for ID11.

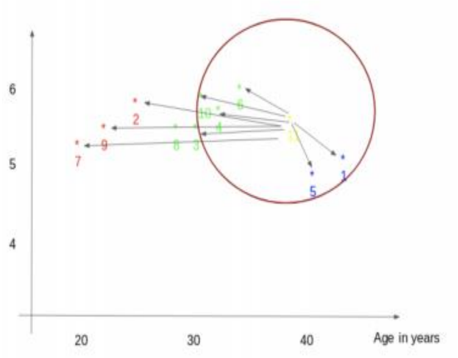


If we choose:

k = 3: ID11 = (77 + 72 + 60)/3 = 69.66

Table 2.3: Example data

ID	Height	Age	Weight
1	5	45	77
5	4.8	40	72
6	5.8	36	60



If we choose:

k = 5: ID11 = (77 + 59 + 72 + 60 + 58)/5 = 65.2

Table 2.4: Example data

ID	Height	Age	Weight
1	5	45	77
4	5.9	34	59
5	4.8	40	72
6	5.8	36	60
10	5.6	32	58

2.4.6. Random Forest

The data matrix X = (X_1, X_2,..., X_p) is assumed to be n p-dimensional. Because of its merits as a regression approach, we recommend employing a randomforest to impute missing data. After being trained on the originally mean imputed data set, the random forest algorithm contains a built-in process to handle missing values by weighting the frequency of the observed values in a variable with the random forest proximities (Breiman (2001)). However, in order to train the forest, this method necessitates a complete response variable.

Instead, we use a random forest trained on the observed parts of the data set to forecast the missing values directly. We can divide the data set into four pieces for an arbitrary variable $X_s$with missing values at entries $i_{mis}^{(s)} \subset \{1, \dots, n\}$:

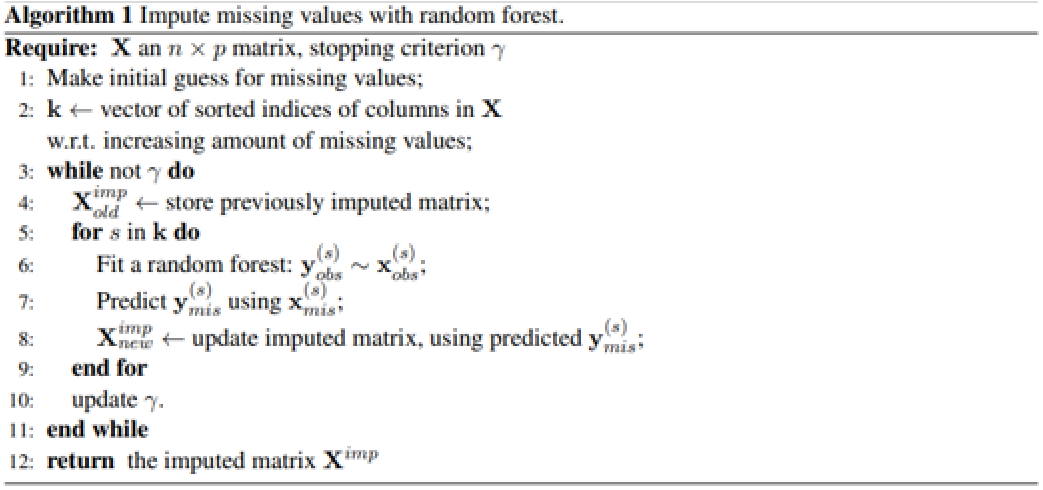
The observed values of variable X_s, denoted by $y_{obs}^{(s)}$;

The missing values of variable $X_{s,}$ denoted by $y_{mis}^{(s)}$

The variables other than X_s, with observations $i_{obs}^{(s)} = \{1, \dots, n\} \backslash i_{mis}^{(s)} denoted\ by\ X_{obs}^{(s)}$;

The variables other than X_s, with observations $i_{miss}^{(s)}\ denoted\ by\ X_{miss}^{(s)}$.

Because the index $i_{obs}^{(s)}$ corresponds to the seen values of the variable X_s, $X_{obs}^{(s)}$; is often not totally observed. Similarly, $X_{miss}^{(s)}$, isn't always fully absent. To begin, use mean imputation or another imputation approach to produce an educated guess for the missing values in X. Then, starting with the least amount of missing data, sort the variables X_s, s = 1, ..., p according to the quantity of missing values. The missing values are imputed for each variable X_s by first training a random forest with responder $y_{obs}^{(s)}$ and predictors $X_{obs}^{(s)}$, and then predicting the missing values $y_{mis}^{(s)}$ by applying the trained random forest to $X_{miss}^{(s)}$. Until a halting requirement is fulfilled, the imputation method is repeated. The missForest approach is represented by the pseudo algorithm 1.



Algorithm 1 Impute missing values with random forest.

```
Require: $\mathbf{X}$ an $n \times p$ matrix, stopping criterion $\gamma$
Make initial guess for missing values;
$\mathbf{k} \leftarrow$ vector of sorted indices of columns in $\mathbf{X}$
    w.r.t. increasing amount of missing values;
while not $\gamma$ do
    $\mathbf{X}_{old}^{imp} \leftarrow$ store previously imputed matrix;
    for $s$ in $\mathbf{k}$ do
        Fit a random forest: $\mathbf{y}_{obs}^{(s)} \sim \mathbf{x}_{obs}^{(s)}$;
        Predict $\mathbf{y}_{mis}^{(s)}$ using $\mathbf{x}_{mis}^{(s)}$;
        $\mathbf{X}_{new}^{imp} \leftarrow$ update imputed matrix, using predicted $\mathbf{y}_{mis}^{(s)}$;
    end for
    update $\gamma$.
end while
return the imputed matrix $\mathbf{X}^{imp}$
```

(Source: Stekhoven, Daniel J. (2011). MissForest - nonparametric missing value imputation for mixed-type data, Algorithm, page-3)

When the difference between the newly imputed data matrix and the prior one grows for the first time with regard to both variable types, the stopping requirement γ is met. For the set of continuous variables N, the difference is defined as:

$$\Delta_N = \frac{\sum_{j\in N}\left(X_{new}^{imp} - X_{old}^{imp}\right)^2}{\sum_{j\in N}\left(X_{new}^{imp}\right)^2}$$

as well as F as a set of categorical variables:

$$\Delta_F = \frac{\sum_{j \in F} \sum_{i=1}^{n} I_{X_{new}^{imp} \neq X_{old}^{imp}}}{\#NA}$$

where #NA is the number of categorical variables with missing values.

2.4.7. Principal Component Analysis – PCA

We used the PCA approach in this research to demonstrate that it can be used to estimate missing values.

PCA approach is used in this research to prove its availability in estimating missing values.

PCA is a technique for reducing the number of measurements in a data set of several variables which are connected to one another while preserving as much information as possible from the original data set. The transformation of the roof variables' set into a new one called "principal component" is used to reduce dimensionality. These new components are unrelated to one another and are arranged in such a way that the first few components preserve as much knowledge about the data set as possible, allowing researchers to investigate the data's aeration-heer error structure using the sample S sample misconception matrix or the R correlation system matrix.

PCA is also commonly used as a pre-processing technique for forecasting models, such as multiple regression, particularly when variables have a high correlation with one another (multicollinearity). PCA is also used in logistical revoicing models for classification problems, as well as for data compression and denoising. We use the PCA approach in this analysis to demonstrate that it can be used to estimate missing values.

Then, to evaluate the PCA method's efficacy, we compare and contrast those that have been completely filled out by the PCA with an original (simple) data set that we have defected on our own. Furthermore, we apply the approach above to larger and more complex data sets to see if pca is efficient with large data sets.

PCA in the complete case boils down to finding a matrix of low-rank S that give:

+ Best approximation of the data with projection.

+ Best representation of the variability.

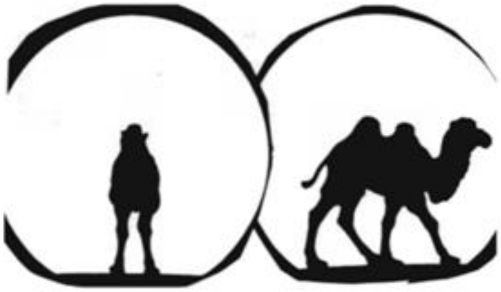


Figure 2.11: Camel or dromedary? (Source: J.P. Fénelon)

Identify principal components:

- Consider $y_j = a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jp}x_p = a_j'x, j = 1, \ldots, p$. With S is covariance matrix, then we have a sample mean Var(y_j)= $a_j'Sa_j$ and sample covariance Cov($y_j, yk) = a_j'Sa_k, \forall k \neq j$.
- To construct principle components $y_1, \ldots. \ , y_n$ we find coefficient vector $a_1, \ldots., a_p$ so that :

  The first principal component = linear combination $a_1'x$ with argmax{Var(y_1)} =$\boldsymbol{a_1'Sa_1}$, $\boldsymbol{a_1'a_1 = 1}$**.**

  The second principal component = linear combination $a_2'x$ with argmax{Var(y_2)} $\boldsymbol{= a_2'Sa_2}$, $\boldsymbol{a_2'a_2 = 1}$ and Cov($\boldsymbol{a_1'x}, \boldsymbol{a_2'x}$) = $\boldsymbol{a_1'Sa_2}$ = 0

At j-th step:

The j principal component = linear combination $a_j'x$ with argmax{Var(y_j)} = $\boldsymbol{a_j'Sa_j}$, $\boldsymbol{a_j'a_j = 1}$ and Cov($\boldsymbol{a_j'x}, \boldsymbol{a_k'x}$) = $\boldsymbol{a_j'Sa_k}$ = 0, $\forall k < j$.

Standard of maximization: max ($\boldsymbol{a'Sa}$).

We always be able to multiply y_1= $\boldsymbol{a'x}$ with a constant $|c| > 1$ for increasing variance :

$$\text{Var}(cy_1) = \text{Var}(c\boldsymbol{a'x}) = c^2\ \text{Var}(\boldsymbol{a'x}).$$

So that we need to normalize vector combination: $\boldsymbol{a'a}$ **= 1**

We need to find a_1 for:

$$\max_{\boldsymbol{a}} \frac{\boldsymbol{a'Sa}}{\boldsymbol{a'a}} = \mathrm{Var}(y_1)$$

We indicate that:

$$\max_{\boldsymbol{a}} \frac{\boldsymbol{a'Sa}}{\boldsymbol{a'a}} = \lambda_1$$

We can get this value when $\boldsymbol{a} = \boldsymbol{e_1}$. Note that λ_1 *and* $\boldsymbol{e_1}$ respectively are eigenvalue and eigen vector of matrix **S**.

We have the following result:

Consider S, sample covariance matrix corresponding to the vector $x' = (x_1, \ldots., x_p)$. **S** has a couple of eigenvectors and eigenvalues $(\lambda_1, \ \boldsymbol{e_1}), (\lambda_2, \ \boldsymbol{e_2}), \ldots, (\lambda_p, \ \boldsymbol{e_p})$, satisfies:

$\lambda_1 \geq \lambda_2 \geq \cdots \lambda_p \geq 0$. Therefore, the jth sample principal component should be:

$$y_j = \boldsymbol{e_j'x} = e_{j1}x_1 + \ e_{j2}x_2 + \cdots + e_{jp}x_p, \qquad j = 1 \, , \ldots., p$$

With $x' = [x_1, \ldots., x_p]$ is a random observation. We have:

$$\mathrm{Var}(y_j) = \lambda_j$$

And $\forall \ k \neq j$,

$$\mathrm{Cov}(y_j, yk) = 0$$

Consider $y_j = \boldsymbol{e_j'}x, y_j = \boldsymbol{e_j'}x, \ldots, y_j = \boldsymbol{e_j'}x$, we have

Total sample variance from the original dataset = $\sum_{i=1}^{p} s_{ii}$ = $\lambda_1 + \ \lambda_2 + \ldots + \lambda_p$,

With: $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_p \geq 0$ are eigenvalues of covariance matrix **S**.

Total sample variance is defined by the kth principle component is:

$$\frac{\lambda_k}{\lambda_1 + \ \lambda_2 + \ldots + \lambda_p} = \frac{\lambda_k}{\sum_{i=1}^{p} \lambda_i} \ , k = 1, 2, \ldots, p$$

The contribution of variable x_k to $y_j th$ principle component:

$$r_{y_j,x_k} = \frac{e_{jk}\sqrt{\lambda_j}}{s_{kk}}$$

If principle components is built by correlation matrix **R** ($s_{kk} = 1$), we shall have:

$$r_{y_j,x_k} = e_{jk}\sqrt{\lambda_j}$$

PCA reconstruction:

Minimizing the distance between observations and their projections.

Approximating the matrix $X_{n\times p}$ with a low-rank matrix S < p in the least square sense ($\|.\|$ the Frobenius norm: $\|X\|_2^2 = tr(XX^T)$):

$$argmin_Q\{\left\|X_{n\times p} - Q_{n\times p}\right\|_2^2 : rank(Q) \leq S\}$$

The PCA solution (Eckart & Young, 1936) is the truncated singular value decomposition (SVD) of X at the order S:

$$\hat{X} = U_{n\times s}\Lambda_{S\times S}^{1/2}\mathrm{V}_{S\times p}^T = F_{n\times S}\mathrm{V}_{S\times p}^T$$

$F = U\Lambda^{1/2}$: PC scores; V: principal axes – loadings.

PCA with incomplete data: weighted least squares (WLS)

$$argmin_Q\{\left\|W_{n\times p} \odot X_{n\times p} - Q_{n\times p}\right\|_2^2 : rank(Q) \leq S\}$$

Where $W_{ij} = 0$ if X_{ij} is missing and $X_{ij} = 1$ otherwise. $\odot$ stands for the element wise multiplication.

PCA imputation Step:

1. Initialization X_0. Missing elements are replaced by initial values such as for example the mean of each variable

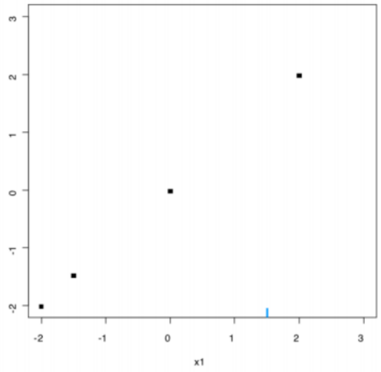
2. Iteration:

a. PCA is performed on the completed dataset to estimate the necessary parameters

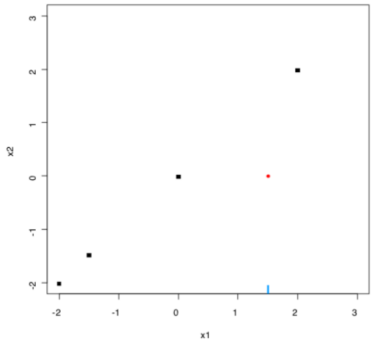
b. Missing values are imputed with the fitted values, observed values are kept the same, we get new X_0

3. Step 2 are repeated until convergence

X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98



X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**NA**
2.0	1.98

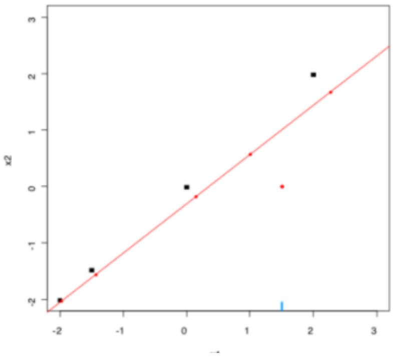


X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

Initialization t = 0: $X^{(0)}$ (mean imputation)

X_1	X_2

-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**NA**
2.0	1.98



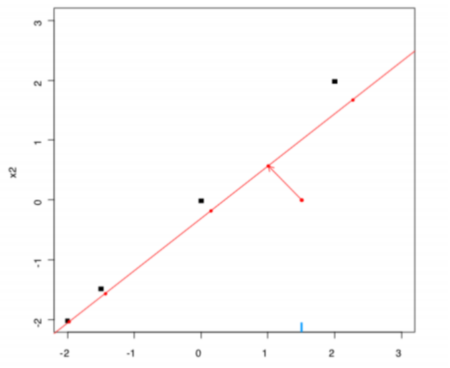
X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

$\widehat{x_1}$	$\widehat{x_2}$
-1.98	**-2.04**
-1.44	**-1.56**
0.15	**-0.18**
1.00	**0.57**
2.27	**1.67**

PCA on the completed dataset: ($U^{(t)}$, $\Lambda^{(t)}$, $V^{(t)}$)

X_1	X_2
-2.0	-2.01
-1.5	-1.48

0.0	-0.01
1.5	**NA**
2.0	1.98



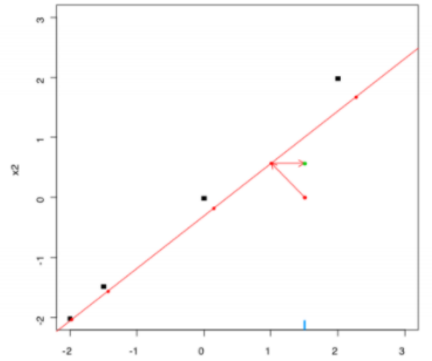
X1	X2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

$\hat{x}_1$	$\widehat{x_2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	0.57
2.27	1.67

Missing values imputed with the fitted matrix $\hat{X}^{(t)} = U^{(t)}\Lambda^{(t)^{1/2}}\mathrm{V}^{(t)^T}$

X1	X2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**NA**

2.0	1.98



X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

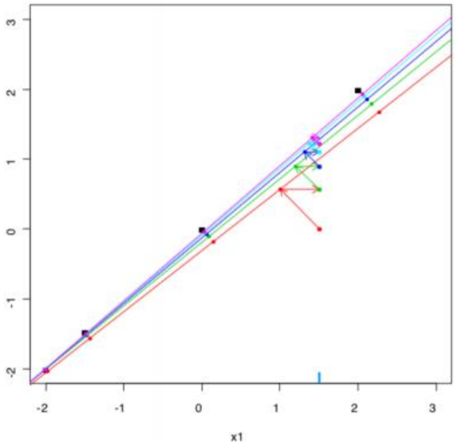
$\widehat{x_1}$	$\widehat{x_2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	0.57
2.27	1.67

X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98

The new imputed dataset is $X^{(t)}$ = W . X + $(1_{n.p} - W).\, \widehat{X^{(t)}}$

X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**NA**
2.0	1.98

X_1	X_2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98



$\widehat{x_1}$	$\widehat{x_2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	0.57
2.27	1.67

X_1	X_2
-2.0	-2.01
-1.5	-1.48

0.0	-0.01
1.5	**0.57**
2.0	1.98

Steps are repeated until convergence

2.4.8. Single imputation and multiple imputation

Imputation consists of single imputation and multiple imputation. Multiple imputation often brings better statistical results, but it is usually time-consuming and much costlier.

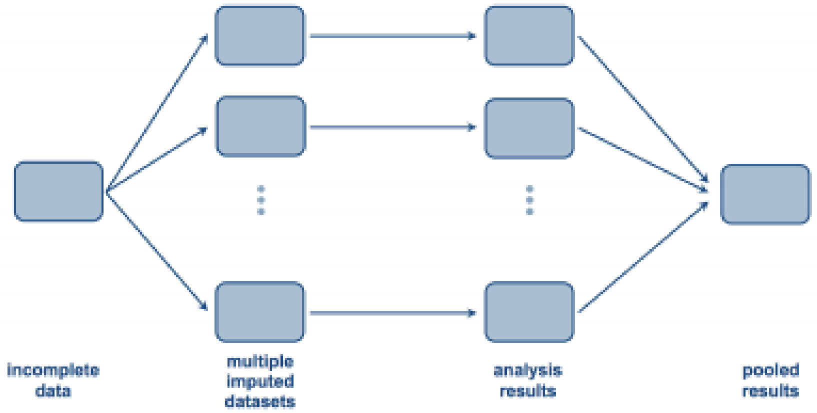


Figure 2.12: Multiple impute process

Three steps:

✓ Imputation: impute multiple times to get multiple completed datasets.

✓ Analysis: analyse each of the datasets.

✓ Pooling: combine results, taking into account additional uncertainty.

Multiple imputation is a technique for replacing missing values with a variety of results in order to find the best results by using calculation and inspection methods.

2.5. Models

2.5.1. Logistic regression

The logistic regression Equation is based on probability to decide the final value of y variable. Regarding the credit rating problems, variable y only carries 2 values such as good debt and bad debt. Throughout the research, we put y = 1 which represents customers who face difficulties in paying debt (bad/unrecoverable debt), and (y = 0) for all remaining positions (good/coverable debts).

P probability general equation:

$$P = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}}$$

- P(y = 1) = P: probability of the event (which leads to a bad unrecoverable debt).
- And P(y = 0) = 1 – P: probability of all other events (which lead to a coverable debt).

Odd is the ratio between an occurred event and a non-occurred event, or bad debt ratio.

Odds = P/(1 - P)

Considering a predictive model contains k inputs or forecast variable X and classification-result variable Y – represents for whether classification result is “convertible debt” or “unrecoverable debt.

Let P(x) is the probability defined by (the Value of Px range from 0 to 1.

The Equation is used for modelling the consequence/relation between Px and X by the formula below:

$$P(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_k X_k}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_k X_k}} \in (0,1)$$

Which is equivalent to:

$$\frac{P(X)}{1 - P(X)} = e^{\beta_0+\beta_1 X_1+\cdots+\beta_k X_k} \in (0, \infty)$$

For β_0, β_1, ... estimation, the above formula should be transformed into:

$$log\ \left(\frac{P(X)}{1-P(X)}\right)=\ \beta_0+\beta_1X_1+\cdots+\beta_kX_k$$

To reach the minimum difference between reality result and forecasted result, we use the Maximum Likelihood Estimation method for hyperparameter optimizing… instead of applying OLS in traditional rowar regression, because of the specific of variable y.

$$l(\beta_0,\beta_1,\dots,\beta_k)=\prod_{i:Y_i=1}^{m}P(X_i)\prod_{i:Y_j=0}^{m}P(X_j)$$

Which is equivalent to log likelihood function:

$$L(\beta_0,\beta_1,\dots,\beta_k)=\sum_{i:Y_i=1}^{m}log\ (P(X_i))+\sum_{i:Y_j=0}^{m}log\ (\ 1-P)$$

2.5.2. Random forest

A Random Forest is a collection of hundreds of Decision Trees. Resampling approaches (pick a component and build models from it) and random/random variable selection from the data sheet are used to generate the Tree. The outputs of a random forest are quite exact, but the structure is complex. As a result, it enables us to achieve high precision, although it is difficult to explain how it works. Furthermore, this paradigm either prevents researchers from highlighting the importance of factors to output or takes a long time to complete.

Random forest randomly selected different data sets to create different Decision trees in order to ensure that different Decision trees produced diverse outcomes. In other words, Random Forest will replace a portion of the data with another data set. Regardless of the little discrepancies between these sets, the Decision space is surely diverse. Random Forest's final anticipated result is more accurate as a result of these various Decisions. Bootstrapping is the term for this procedure.

CHAPTER 3: DATA PREPROCESSING

Chapter 3 begins with a description of the data and several charts to help readers comprehend the data's pattern. Furthermore, in this chapter, we will examine several feature correlations and compare the evaluation outcomes when we run data with the model on different sides. The implementation has been done using Python language.

3.1. Exploratory Data analysis

In this section, we'll go over set of credit data that was included in “experiment”. "Lending method, customer type, gender, basic balance, DUNO QD, term, Loan type, Parentorgname, loan purpose, Label" are among the 11 qualities that each bank client has.

In this section, we introduce a data credit set, which contains 95271 bank customer profiles from a Vietnamese bank, with 84981 customers accounting for about 89.2 percent of those with a good credit rating and 10290 customers accounting for approximately 10.8% of those with a low credit rating.

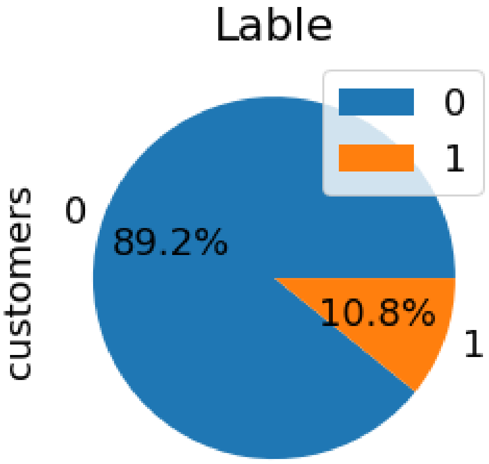
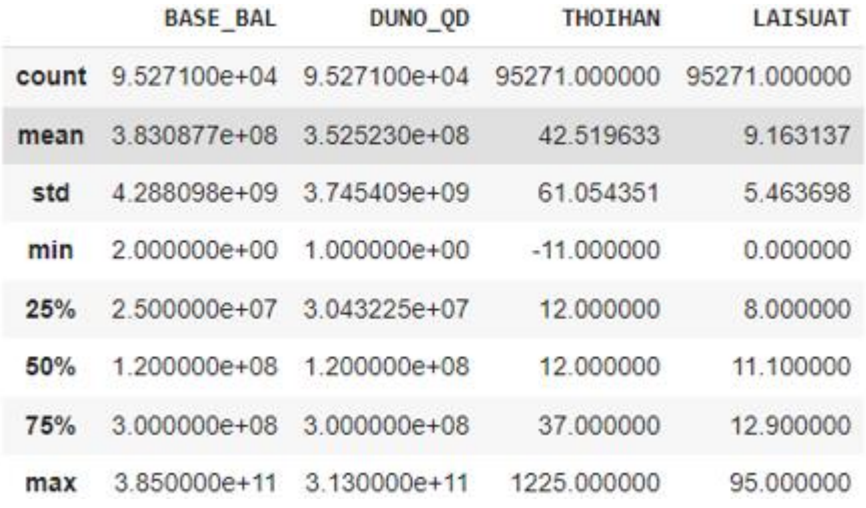


Figure 3.1: How imbalance of this data

Four of the eleven factors listed above are quantitative. This dataset is rather huge, and because it was compiled from several branches, there may be noise in the data during collection and processing. It's worth noting that some properties with missing values were eliminated in more than half of the cases. Table below shows descriptive statistics for four continuous variables in this dataset.

Table 3.1: Continuous features



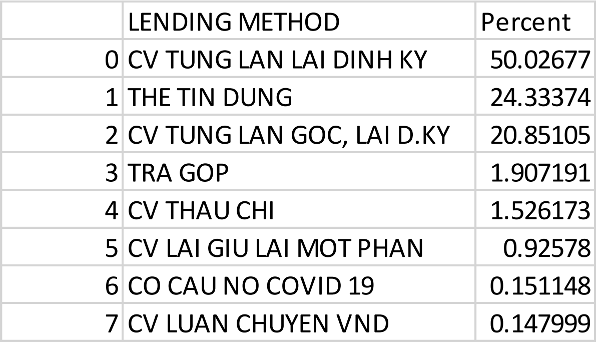
	BASE_BAL	DUNO_QD	THOIHAN	LAISUAT
count	9.527100e+04	9.527100e+04	95271.000000	95271.000000
mean	3.830877e+08	3.525230e+08	42.519633	9.163137
std	4.288098e+09	3.745409e+09	61.054351	5.463698
min	2.000000e+00	1.000000e+00	-11.000000	0.000000
25%	2.500000e+07	3.043225e+07	12.000000	8.000000
50%	1.200000e+08	1.200000e+08	12.000000	11.100000
75%	3.000000e+08	3.000000e+08	37.000000	12.900000
max	3.850000e+11	3.130000e+11	1225.000000	95.000000

3.2. Some data features

In the following we discuss important features in our data.

3.2.1. Lending method

Table 3.2: Frequency of Lending table



	LENDING METHOD	Percent
0	CV TUNG LAN LAI DINH KY	50.02677
1	THE TIN DUNG	24.33374
2	CV TUNG LAN GOC, LAI D.KY	20.85105
3	TRA GOP	1.907191
4	CV THAU CHI	1.526173
5	CV LAI GIU LAI MOT PHAN	0.92578
6	CO CAU NO COVID 19	0.151148
7	CV LUAN CHUYEN VND	0.147999

We can see that the majority of bank-approved loans are in the form of credit lending with periodic interest, which is frequently used for medium and long-term loans. Furthermore, three lending strategies, including "Recurring Interest Installment Loan," "Credit Card," and "Principal Loan, Periodic Interest," all fall into the low-risk lending category. The COVID 19 Debt Structure technique, on the other hand, has the highest default risk, although accounting for a minor portion of the data set (more than 90 percent).

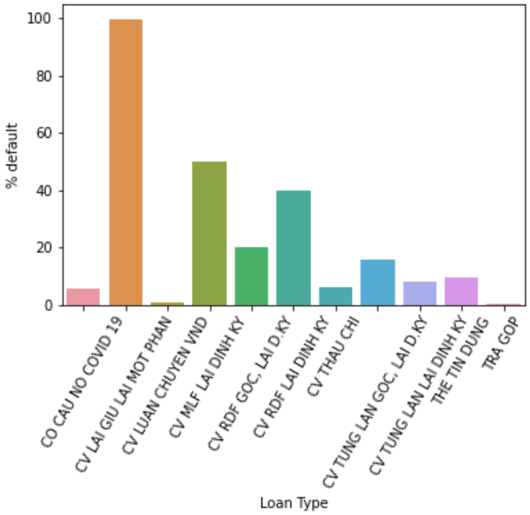


Figure 3.2: The default rate according to each kind of Lending method

3.2.2. Customer type

First and foremost, according to this commercial bank's customer-specified code type, DIN is used for individual customers, whereas DCO is utilized for corporate customers.

Individual customers (DIN) make up the majority of the borrowers, accounting for 98% of the total. Corporate clients, on the other hand, make up a small percentage of the total, 2%.

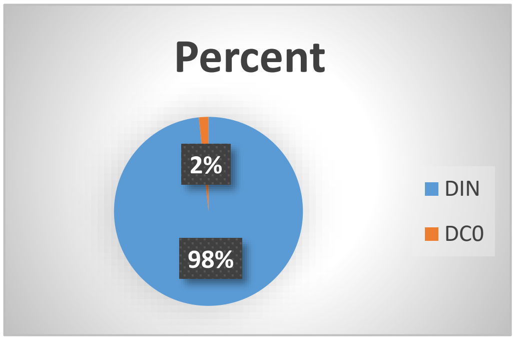


Figure 3.3: The percentage account for each kind of Customer type

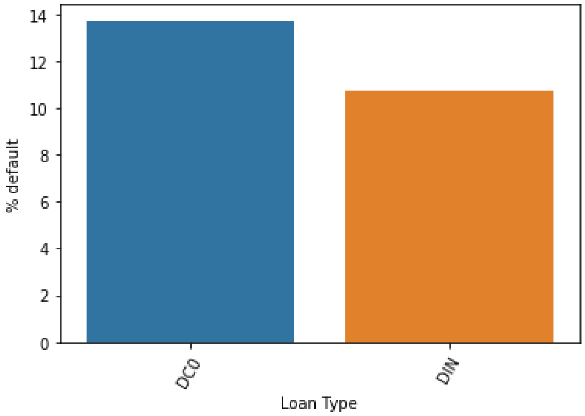


Figure 3.4: The default rate according to each kind of Customer type groups

Males account for 63% of the bank's customers, while females account for only 35%. The remaining minority category is made up of corporations seeking to borrow money.

As can be seen, the bank's loan risk ratio for corporate clients is significantly greater than that for individual clients. Furthermore, when each gender group is considered, the consumer group that belongs to companies has a higher risk rate than the other.

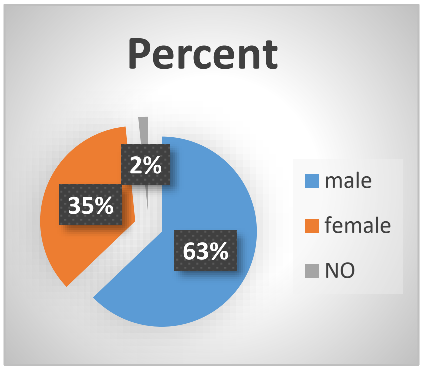


Figure 3.5: The percentage account for each kind of groups

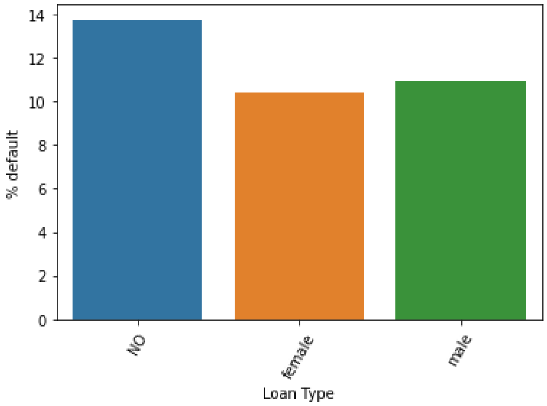
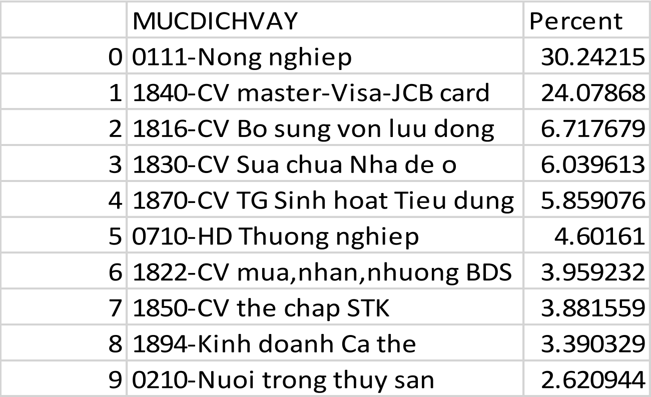


Figure 3.6: The default rate according to each kind of groups

3.2.3. Purpose

Table 3.3: Top ten highest frequency of Purpose feature



	MUCDICHVAY	Percent
0	0111-Nong nghiep	30.24215
1	1840-CV master-Visa-JCB card	24.07868
2	1816-CV Bo sung von luu dong	6.717679
3	1830-CV Sua chua Nha de o	6.039613
4	1870-CV TG Sinh hoat Tieu dung	5.859076
5	0710-HD Thuong nghiep	4.60161
6	1822-CV mua,nhan,nhuong BDS	3.959232
7	1850-CV the chap STK	3.881559
8	1894-Kinh doanh Ca the	3.390329
9	0210-Nuoi trong thuy san	2.620944

The majority of customers request for loans in order to keep their agriculture, forestry, and aquaculture businesses afloat. A number of consumers are then lent money for the purpose of making foreign installment payments using a master, Visa, or JCP card. Customers who borrow money to keep their businesses and finances afloat are in the minority at this bank, but there are many small groups that ask for loans for a variety of reasons.

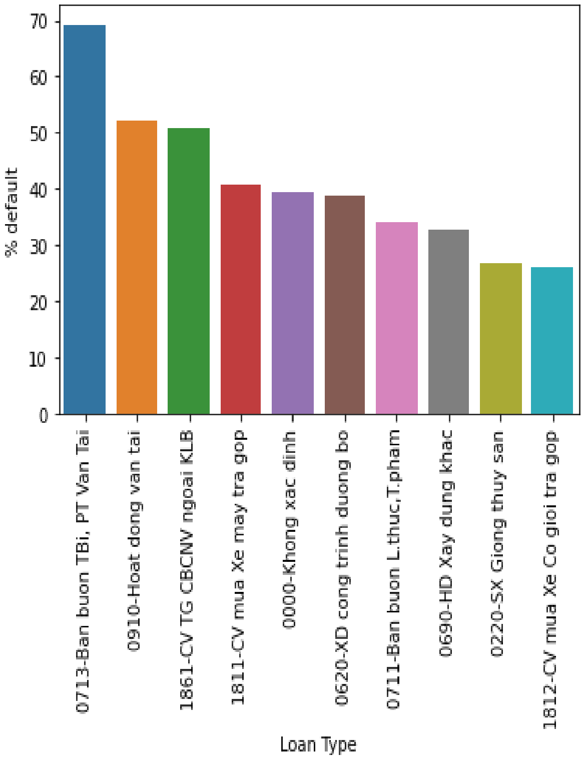


Figure 3.7: The default rate according to each kind of Purpose

The transportation and road transport sectors are the group of loan purposes with the highest credit risk ratio among these loan reasons. The trucking business by road has the largest risk ratio in this bank's group of lending objectives, with a risk ratio of up to 70%. Furthermore, this group of industries is closely followed by purposes such as installment loans for motorcycles (40.62 percent), transportation activities (52 percent), and road building (38.70 percent). Furthermore, credit lending activities for state officials outside of banks have a risk ratio of 50.92 percent.

3.2.4. Interest rate

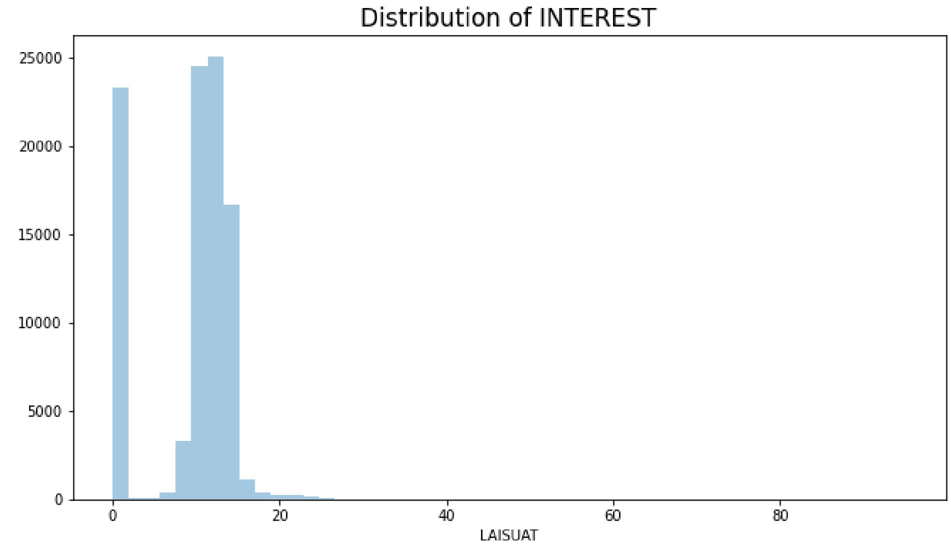


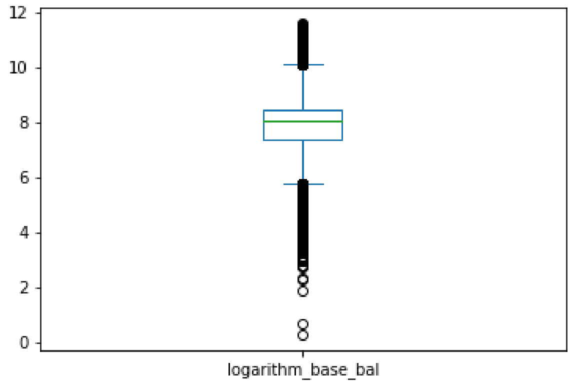
Figure 3.8: Distribution of Interest rate

Values range from a minimum of roughly 0% per month to a maximum of over 95% per month, with a median of 9.2% , as shown in the continuous table in table 3.1. Furthermore, the top and third quartiles, respectively, are 8% and 12.9% .The figure below describes the distribution of interest rate features in this dataset.

When it comes to the Bank's lending rates, there are two primary kinds of consumers. One set of borrowers has very low-interest rates, from 0% to about 5%, while the other group has higher interest rates ranging from around 6% to more than 20%.

3.2.5. Base Balance & Duno_QD

The aggregate of data types from numerous subbanks can result in noise in some variables. The distributions of the Base balance and Duno QD variables were severely skewed, so we transformed them to base 10 logarithms to smooth them down even more.



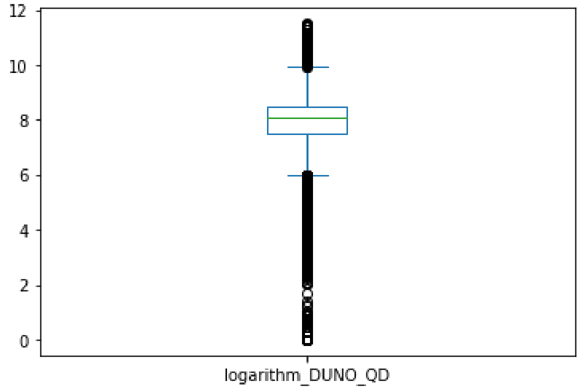


Figure 3.9: The box plot of logarithm_base_bal

Figure 3. 10: The box plot of DUNG_QD

It is estimated that nearly half of the customer group's base balance is in the billions of dollars or more. However, similar to Duno QD, there are still examples of having a very high base balance.

3.3. Correlation

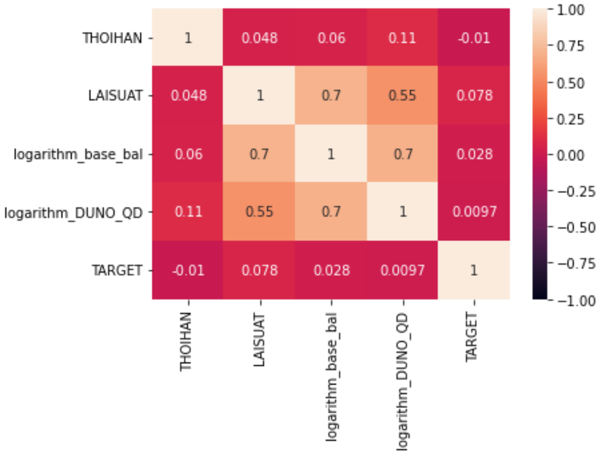


Figure 3.11: Correlation heatmap of salient features

We can evaluate the link between pairs of quantitative variables in this pair plot and determine whether or not to adjust the pairs of strongly correlated variables. The variables are all correlated from average or less, with the exception of the base balance and Duno qd logarithmic variables, which are strongly correlated.

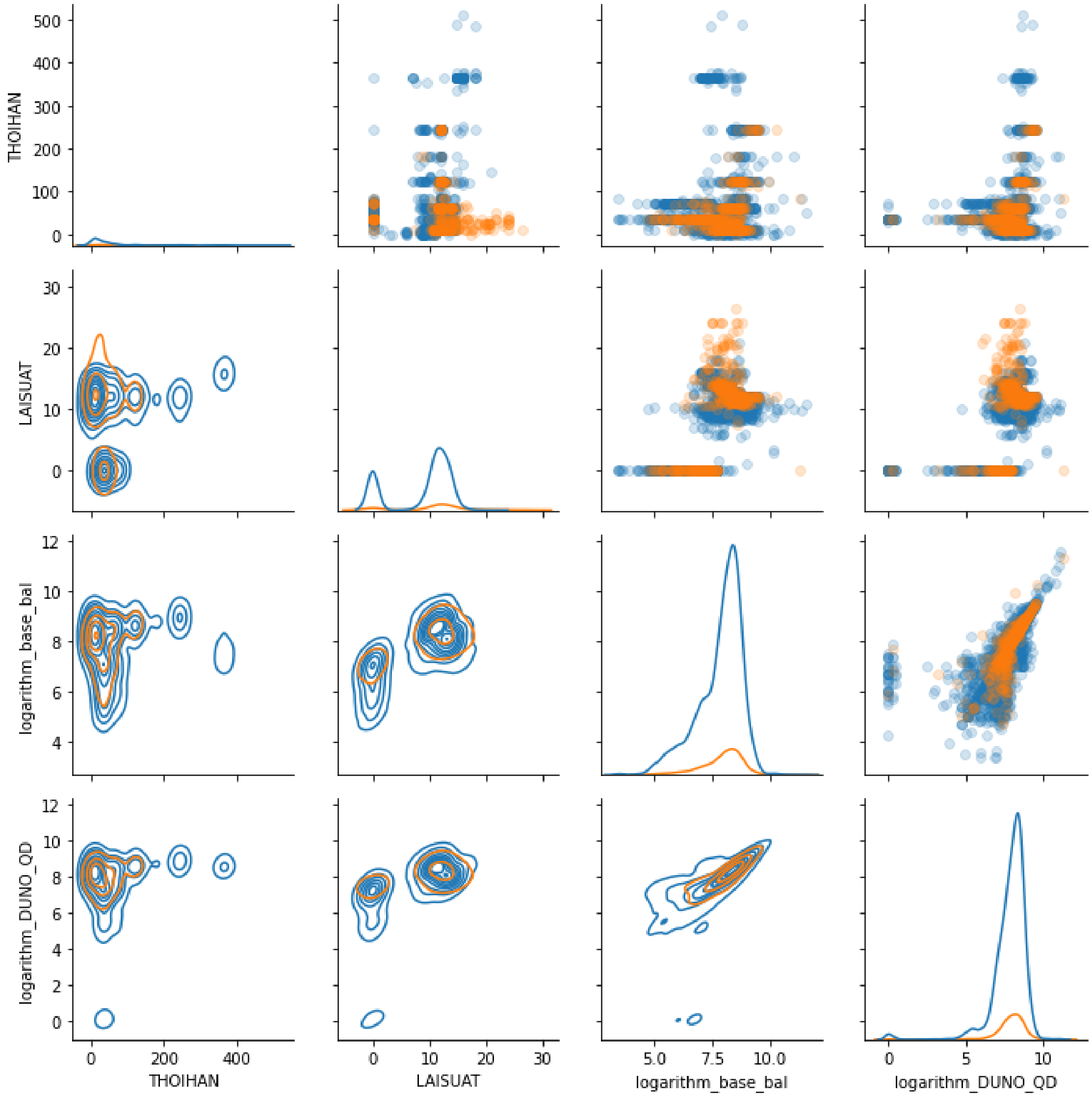
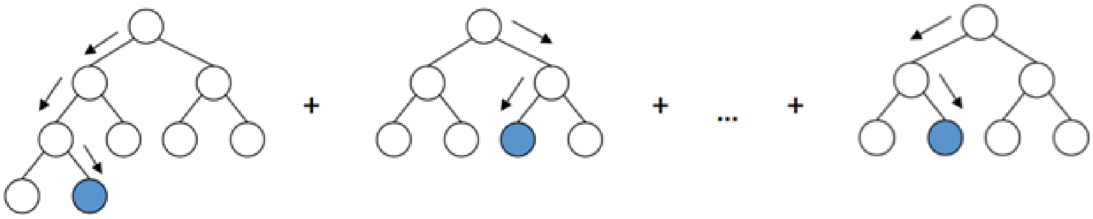


Figure 3.12: Correlation grid of salient features

Furthermore, when examining the association between the distributions of the pairs of variables based on the goal label. We discover that the data set has labels that are stacked on top of one other, indicating that this is a challenging data set to categorize because the quantitative variables do not clearly represent population distributions. There are both bad and nice customers. This is one of the factors that influences our models' training during the fill-in application and subsequent label categorization.

3.4. Feature selection

For classification and regression applications for tabular data and time series, gradient boosted decision trees such as XGBoost and LightGBM have become popular. Typically, the features that represent the data are extracted first, then used as the input for the trees.



Schematics of decision trees ensembling, author: Mohtadi Ben Fraj, source: medium

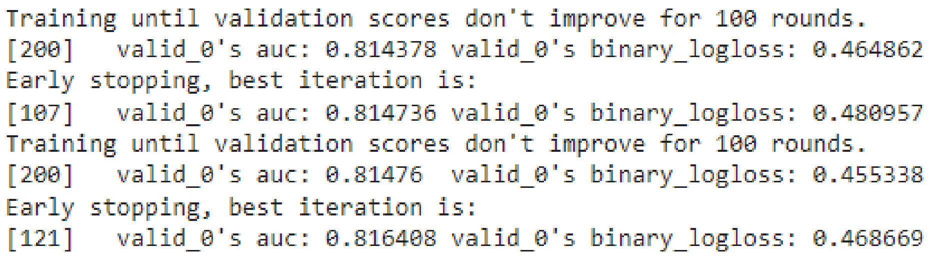
A feature in your data collection is an individual measurable quality or characteristic of a phenomena being observed. Various statistics (mean, standard deviation, median, percentiles, min, max, and so on), trends (rise and decay), peak analysis (periods, average peaks breadth, peaks number, frequency), autocorrelations and cross-correlations, and so on are just a few of the features that can be found. After the features have been retrieved from the data, they are fed into the gradient boosted decision trees (GBDT). However, because the GBDT is prone to overfit, it's vital to limit the number of features, leaving only those which aid the classifier, especially for small data sets. The features selection process is an important aspect of the decision tree pipeline. The features selection aids in reducing overfitting, removing redundant features, and preventing the classifier from becoming confused. Several prominent ways to select the most relevant qualities for the task are described below.

3.5. Recursive feature removal

The automatic tool for recursive feature reduction from the sklearn library [5] is an option for removing superfluous features. More often than the alternative without cross-validation, recursive feature elimination with cross-validation is utilized.

The purpose of this tool is to choose features by looking at smaller and smaller sets of features in a recursive manner.

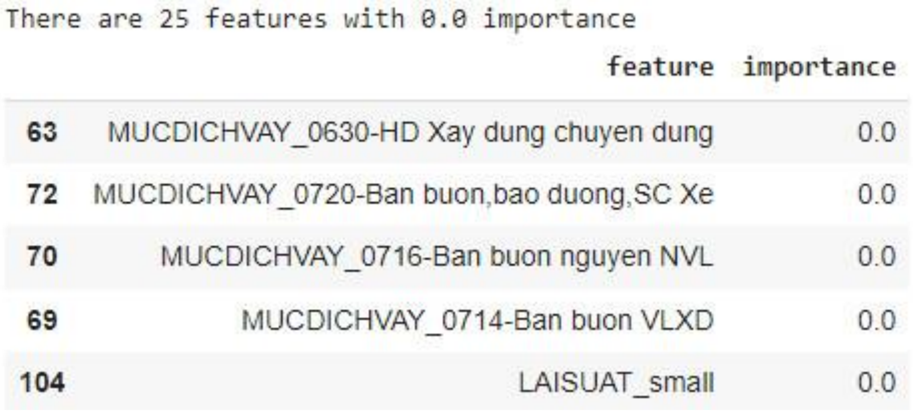
- The estimator is first trained on the initial set of features after the relevance of each feature is determined.



```
Training until validation scores don't improve for 100 rounds.
[200]   valid_0's auc: 0.814378 valid_0's binary_logloss: 0.464862
Early stopping, best iteration is:
[107]   valid_0's auc: 0.814736 valid_0's binary_logloss: 0.480957
Training until validation scores don't improve for 100 rounds.
[200]   valid_0's auc: 0.81476  valid_0's binary_logloss: 0.455338
Early stopping, best iteration is:
[121]   valid_0's auc: 0.816408 valid_0's binary_logloss: 0.468669
```

Output from training LGBM to get the best iteration

- The least important features are then eliminated from the current collection of features, and the classification metric is verified once more.



There are 25 features with 0.0 importance

	feature	importance
63	MUCDICHVAY_0630-HD Xay dung chuyen dung	0.0
72	MUCDICHVAY_0720-Ban buon,bao duong,SC Xe	0.0
70	MUCDICHVAY_0716-Ban buon nguyen NVL	0.0
69	MUCDICHVAY_0714-Ban buon VLXD	0.0
104	LAISUAT_small	0.0

Eliminate the non-importance features

- Recursively, the technique is repeated until the desired number of features to pick is attained.

The first approximation of the useful feature set is provided by this tool. However, automated feature deletion is not always ideal, and it frequently necessitates additional fine-tuning. Following the recursive elimination mentioned above to choose the first set of features, we use features importance to select features.

3.6. Feature importance

Techniques that assign a score to input features depending on how valuable they are at predicting a target variable are known as feature importance.

Statistical correlation scores, coefficients generated as part of linear models, decision trees, and permutation importance scores are some of the most common types and sources of feature importance scores.

In a predictive modelling project, feature relevance scores play a significant role in providing insight into the data, insight into the model, and the basis for dimensionality reduction and feature selection, which can increase the efficiency and efficacy of a predictive model on the problem. For this research, we apply the decision trees method for feature important scores, we use a package called lightgbm.plot_importance to those scores.

>>> norm_feature_importances = plot_feature_importances(feature_importances)

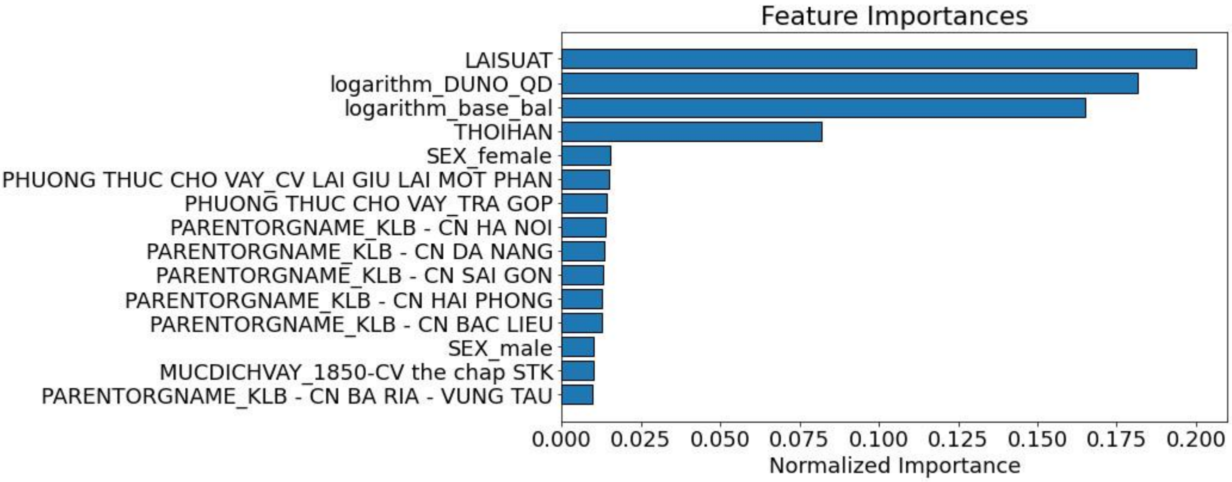


Figure 3.13: Feature importance scores after training LGBM

As we can see in figure 3.13, the LGBM feature importance scores shown, indicate that 3 variables “DRAWDOWN_AMOUNT”, “DURATION”, “INT_RATE” are 3 strongest feature affecting the credit scoring of this commercial bank.

3.7. Balancing method

It's tough to categorize bad and good consumers because of the problem of overlapping data points (Figure 3.12). Not to mention the issue of data disparity between the groups of satisfied and dissatisfied clients. To exclude the group of consumers with good labels who are close to customers with bad labels, we use the Tome Links approach. This eliminates the problem of excellent and bad observations overlapping.

SMOTE, on the other hand, is a newer version of the Subsampling approach. In contrast to Oversampling, which allows us to enlarge subgroups by repeating their observations, SMOTE employs the K algorithm, which generates new artificial elements (instead of iterating). However, because new components are not guaranteed, we combine Tomek

Links with SMOTE to raise the number of problematic labels while reducing the number of good labels that make up the majority of the dataset. This helps the model discriminate between two groups of excellent and bad consumers by emphasizing the range of values that reflect them.

3.8. Hyperparameter by optuna framework

Optuna is an open-source system for hyperparameter optimization that automates the search for hyperparameters. Optuna's primary characteristics include:

Define-by-run: programming that allows the user to construct the search space dynamically.Existing frameworks describe the search space and objective function separately. All hyperparameters are defined on the fly in Optuna, and the search spaces are defined inside the objective function. This capability allows for additional modulation and modification of Optuna coding.

Parallel distributed optimization: With near-linear scalability, Optuna can parallelize your optimization. Users can set up parallelization by running numerous optimization processes at the same time, and Optuna will automatically share trials in the background.

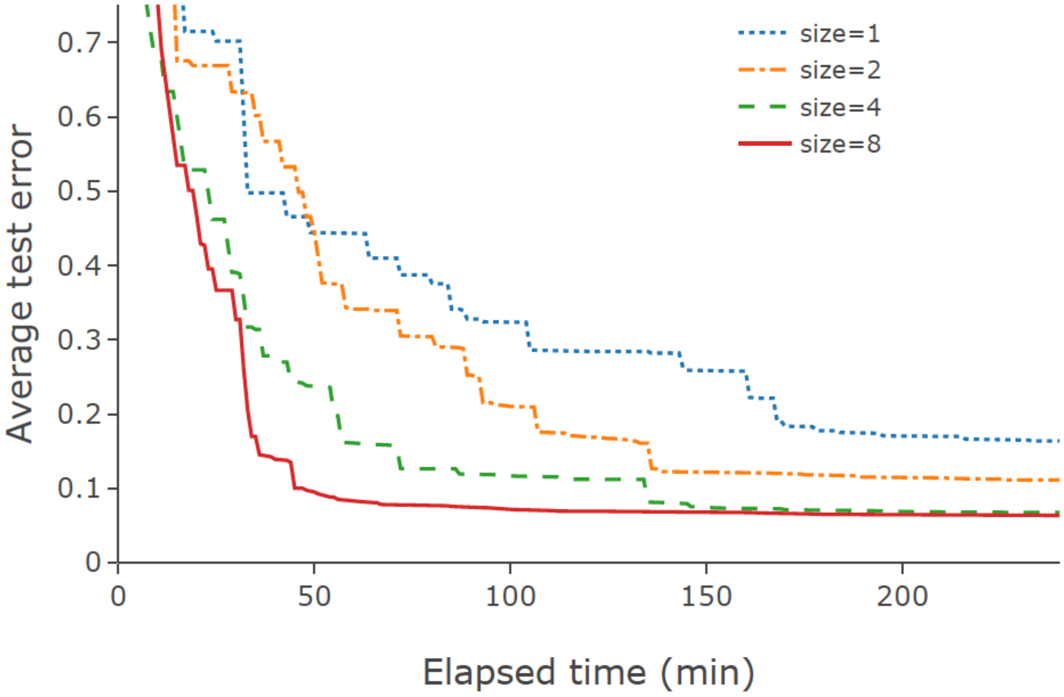
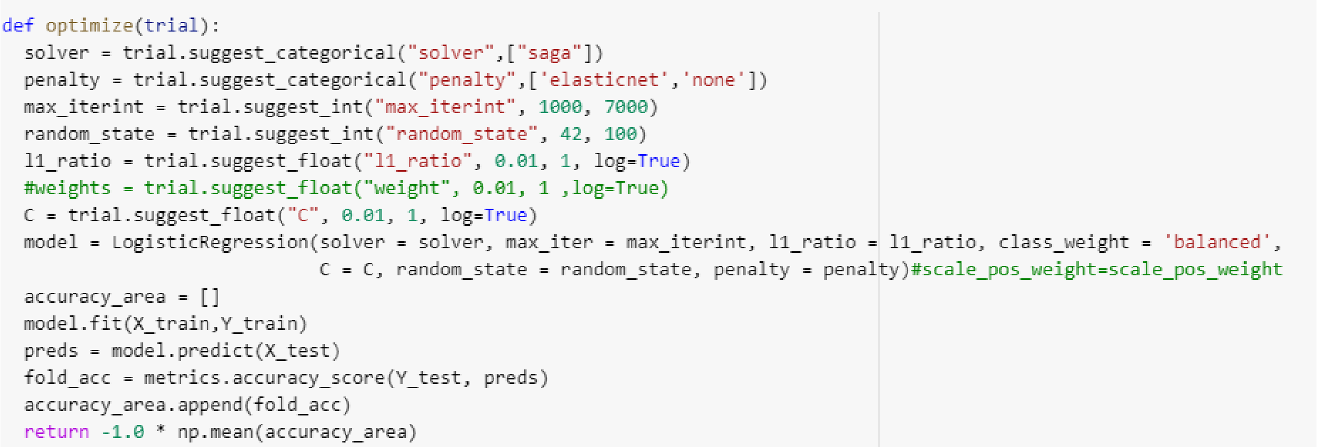


Figure 3.14: Effect of parallelization sizes of 1, 2, 4, and 8

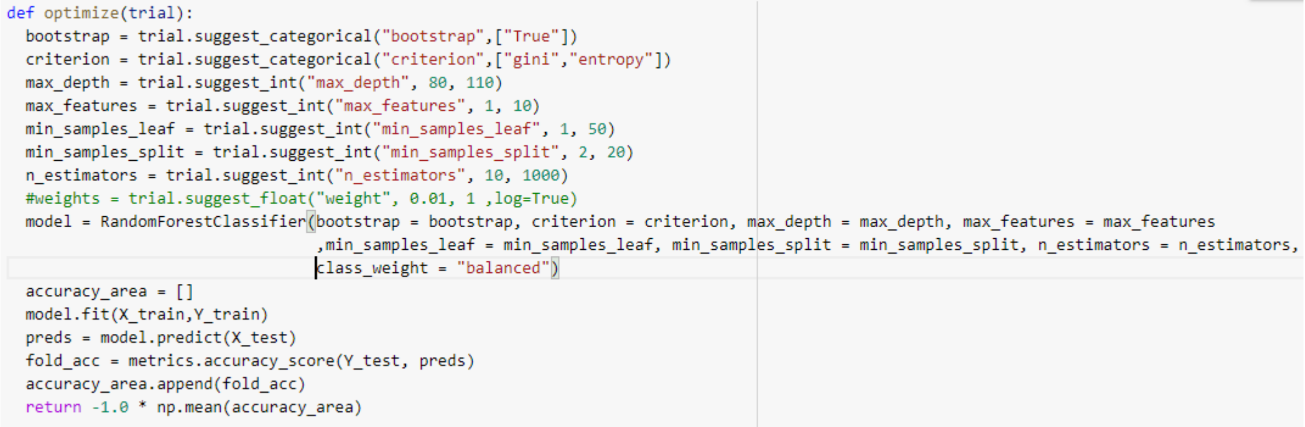
Pruning of unpromising trials: At the start of the training, the pruning feature automatically stops unpromising attempts (a.k.a., automated early-stopping). In iterative training algorithms, Optuna provides APIs for quickly implementing the pruning process.

For both logit and randomforest models, we create a search space and goal function (Define by run) in this study. Over 100 trials were completed for the self-training algorithm.



```
def optimize(trial):
  solver = trial.suggest_categorical("solver",["saga"])
  penalty = trial.suggest_categorical("penalty",['elasticnet','none'])
  max_iterint = trial.suggest_int("max_iterint", 1000, 7000)
  random_state = trial.suggest_int("random_state", 42, 100)
  l1_ratio = trial.suggest_float("l1_ratio", 0.01, 1, log=True)
  #weights = trial.suggest_float("weight", 0.01, 1 ,log=True)
  C = trial.suggest_float("C", 0.01, 1, log=True)
  model = LogisticRegression(solver = solver, max_iter = max_iterint, l1_ratio = l1_ratio, class_weight = 'balanced',
                             C = C, random_state = random_state, penalty = penalty)#scale_pos_weight=scale_pos_weight
  accuracy_area = []
  model.fit(X_train,Y_train)
  preds = model.predict(X_test)
  fold_acc = metrics.accuracy_score(Y_test, preds)
  accuracy_area.append(fold_acc)
  return -1.0 * np.mean(accuracy_area)
```

Figure 3. 15: Search space for logit



```
def optimize(trial):
  bootstrap = trial.suggest_categorical("bootstrap",["True"])
  criterion = trial.suggest_categorical("criterion",["gini","entropy"])
  max_depth = trial.suggest_int("max_depth", 80, 110)
  max_features = trial.suggest_int("max_features", 1, 10)
  min_samples_leaf = trial.suggest_int("min_samples_leaf", 1, 50)
  min_samples_split = trial.suggest_int("min_samples_split", 2, 20)
  n_estimators = trial.suggest_int("n_estimators", 10, 1000)
  #weights = trial.suggest_float("weight", 0.01, 1 ,log=True)
  model = RandomForestClassifier(bootstrap = bootstrap, criterion = criterion, max_depth = max_depth, max_features = max_features
                                 ,min_samples_leaf = min_samples_leaf, min_samples_split = min_samples_split, n_estimators = n_estimators,
                                 class_weight = "balanced")
  accuracy_area = []
  model.fit(X_train,Y_train)
  preds = model.predict(X_test)
  fold_acc = metrics.accuracy_score(Y_test, preds)
  accuracy_area.append(fold_acc)
  return -1.0 * np.mean(accuracy_area)
```

Figure 3.16: Search space for randomforest

CHAPTER 4: IMPLEMENTATION AND RESULTS

In this chapter, we randomly remove approximately 12% of observations from four columns of the original data set, causing some missing values. Then we'll use the previously discussed approaches to estimate the missings.

The chapter also introduces to readers some critical measures to evaluate the efficacy of fill-in based on the situation that the research team has identified. Finally, we will use the filled data sets to run two classification models and evaluate how the imputation affects the predictability of the customer's labels.

4.1. The artificial missing data

According to the MAR loss pattern, we introduce some missing observations in the three most critical columns in the dataset. The "ampute" function in the R software was used in data-preprocessing

```
>>> missing_data <- ampute (data_simlate2 , prop = 0.5, patterns = NULL, freq = NULL, mech = "MAR", weights = NULL, std = TRUE, cont = TRUE, type = NULL, odds = NULL, bycases = TRUE, run = TRUE)
```

```
>>> class(missing_data)
```

```
>>> head(missing_data$amp)
```

```
>>> data_simlate <- missing_data$amp
```

```
>>> View(data_simlate)
```

Table 4.1: First five entries of missing dataset

	LAISUAT	DUNG_QD	Logarithm_base_bal	THOIHAN
1	NA	8.93	8.9	112
2	NA	8.15	8.52	240
3	NA	8.55	8.49	243
4	24	7.55	7.55	37
5	NA	7.95	8.63	244

In addition, we use a variety of specialized libraries for data visualization supplied in this software so that readers may quickly grasp the dataset's missing rule.

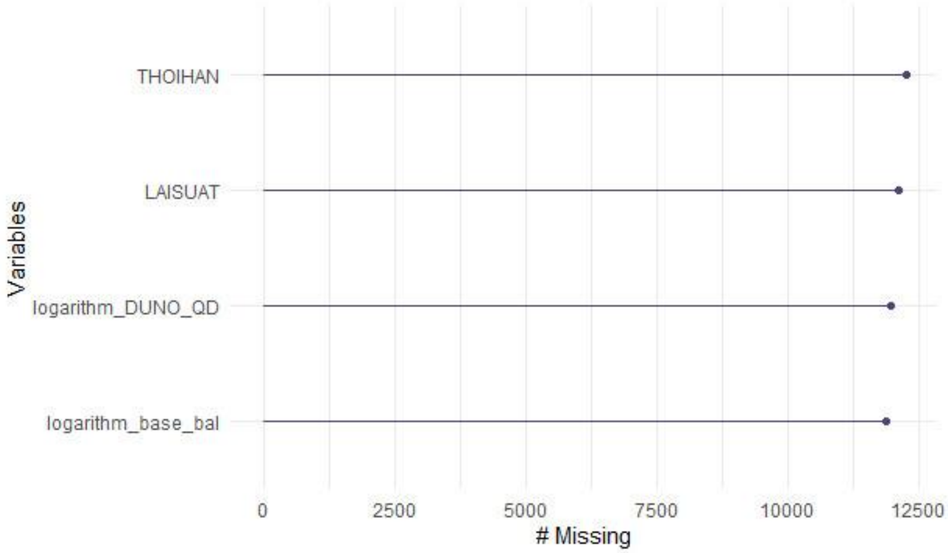


Figure 4.1: Number of missing observations

Table 4.2: Variables sorted by number of missing values

Variable	Number of missing values	Percent of missing
LAISUAT	**12361**	**12.98165**
DUNG_QD	**12055**	**12.66029**
Logarithm_base_bal	**11956**	**12.55632**
THOIHAN	**11888**	**12.48490**

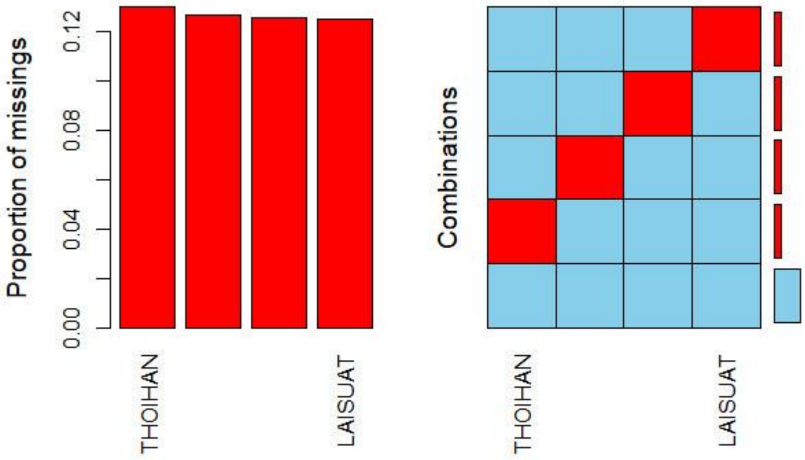


Figure 4.2: Proportion of missing values

The data flaws can be seen in the results that were exported from the tables and graphs above. This dataset contains 4659 missing values, or 49.32% of the missing dataset's observations. About 12% of the value of each column is lost. Moreover, figure 4.2 reflects the number of missing entries in each variable, as well as the number of missing entries in particular combinations of variables.

4.2. Missing data imputation

4.2.1. Multiple PCA imputation

We have scaled and normalized the data before using the PCA approach to avoid deviations while calculating the variance and covariance matrix for each variable.

```
>>> data_simlate1 <- scale(data_simlate, center = TRUE, scale = TRUE)

>>> View(data_simlate1)
```

Table 4.3: The normalized dataset

STT	LAISUAT	DUNG_QD	Logarithm_base_bal	THOIHAN
1	NA	0.97	1.12	1.17
2	NA	0.3	0.73	3.31
3	NA	0.64	0.7	3.36
4	2.71	-0.22	-0.28	-0.09
5	NA	0.12	0.85	3.37

Fill in the repetitive defect 100 times with the MIPCA feature (language R) and look for the best populated data set. We also set the number component equal 2 for evaluating the informational value of new the coordinate system.

```
>>> res.MIPCA <- MIPCA(data_simlate1, ncp = 2, nboot = 100)

>>> imputed <- unscale(res.MIPCA[["res.imputePCA"]])

>>> View(imputed)
```

Table 4.4: The first five rows of imputed dataset by multiple PCA

STT	LAISUAT	DUNG_QD	Logarithm_base_bal	THOIHAN
1	12	8.77	8.9	112
2	18	8.15	8.52	39.65

3	11.46	8.55	8.49	243
4	24	7.55	7.55	37
5	18	8.88	8.63	244

4.2.2. KNN imputation

For the KNN method, the first is that we need to prepare the necessary algorithm packages (Python language)

```
>>> import pandas as pd
>>> import sys
>>> import sklearn.neighbors._base
>>> sys.modules['sklearn.neighbors.base'] = sklearn.neighbors._base
>>> from sklearn.impute import KNNImputer
```

Then we assign the number of neighbors allowed by 2 (n_neighbors = 2) and proceed to fill in the defect.

```
>>> imputer = KNNImputer(n_neighbors=2)
>>> X_new = imputer.fit_transform(df)
```

As the data is filled in, it will be converted to array form. Therefore, we need the utility of the pandas library to transfer our output to the form of dataframe.

```
>>> col = list(df.columns)
>>> sgb_knn = pd.DataFrame(X_new, columns= col)
```

4.2.3. Miss Forest imputation

Like the KNN method, the first step of the Miss Forest method we also need to prepare the necessary algorithm packages is the first step of the Miss forest method (Python language).

```
>>> import pandas as pd
>>> from missingpy import MissForest
```

Then we start calling to assign inputs to the library to fill in the missing values.

```
>>> imputer2 = MissForest()
>>> X_new2  = imputer2.fit_transform(X)
```

Similar to KNN, after the data is filled out, it will be converted to array form, so we need the utility of the pandas' library to transfer our output to a dataframe form.

```
>>> sgb_ran = pd.DataFrame(X_new2, columns= col)
```

4.3. Performance measurement

4.3.1. Evaluate bias performance

Multiple imputation is a statistical technique for obtaining statistically accurate inferences from partial data. As a result, the imputation method's quality should be assessed in relation to this purpose. There are a number of indicators that can help us determine the statistical accuracy of a technique. These are the following:

- Raw bias (RB) and percent bias (%) are two types of bias (PB). The difference between the expected value of the estimate and truth is $\bar{Q}$ as the estimate's raw bias:$: RB = E(\bar{Q}) - Q$. RB should be as close to zero as possible. Bias can also be measured in percentages: $PB = 100 \times |E(\bar{Q}) - Q|$. We set a PB of 5% as the top limit for acceptable performance.

- Rate of coverage (CR). The coverage rate (CR) is the percentage of confidence intervals in which the true value is contained. The real rate should be the same as or higher than the nominal rate. The approach is too optimistic if CR goes below the nominal rate, resulting in false positives. A CR of less than 90% for a nominal 95% interval implies poor quality. A high CR (e.g., 0.99) may suggest that the confidence interval is excessively large, making the approach inefficient and resulting in overly conservative findings. Inferences that are "too conservative" are considered a smaller offense than those that are "too hopeful".

- Average width (AW). The breadth of the confidence interval on average is a measure of statistical efficiency. The length should be as short as possible, but not to the point when the CR falls below the nominal value.

Note that if all is well, then RB should be close to zero, and the coverage should be near 0.95.

After filling in the techniques indicated in the previous section, we calculate the deviation efficiency and present the findings in the table below:

Table 4.5: Bias evaluation missing on logarithm_base_bal

	RB	PB	CR	AW
MIPCA	-0.0017287	0.0220%	1	0.0133784
KNN	0.0018299	0.0233%	1	0.0136054
MIRAN	-0.0006556	0.0083%	1	0.0136555
Median imputation	-0.0063367	0.0806%	1	0.0130933

Table 4.6: Bias evaluation missing on logarithm_DUNO_QD

	RB	PB	CR	AW
MIPCA	0.000780978	0.0099%	1	0.01617711
KNN	0.002270398	0.0289%	1	0.01621913
MIRAN	-0.000831229	0.0106%	1	0.01628906
Median imputation	-0.01384717	0.1764%	0	0.01584629

Table 4.7: Bias evaluation missing on LAI SUAT

MIPCA	0.008352	0.09118%	1	0.077553
KNN	0.004232	0.04620%	1	0.0790
MIRAN	3.02E-05	0.00033%	1	0.078914
Median imputation	0.084974	0.92764%	0	0.076186

Table 4.8: Bias evaluation missing on THOI HAN

	RB	PB	CR	AW
MIPCA	-0.0375	0.0881%	1	0.81458
KNN	0.17080	0.4014%	1	0.88763
MIRAN	0.0736	0.1730%	1	0.88278
Median imputation	-4.1375	9.7239%	0	0.82381

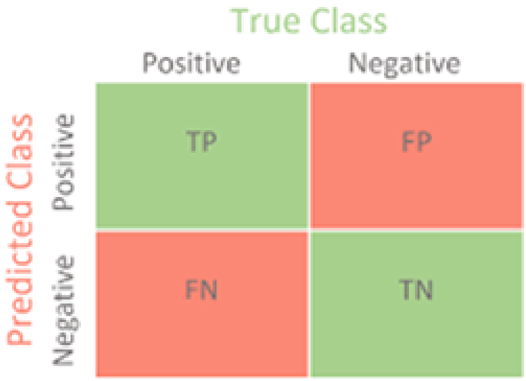
Inclusion: According to the table 4.8, PB value of duration variable is larger than 5%. Therefore the Median Imputation is unacceptable. Moreover, the CR value of duration variable of median dataset is 0%.meaning that this dataset does not have any possibility for covering the true value of complete dataset. However, the rest of the methods show that they can cover the true values. Their PBs are lower than 5% and their CR is 100%.

4.3.2. Performance on machine-learning models

In this part, we'll run both Logit and Random forest models using all of the blank-filled data sets and the beginning data sets. We use the following scales to assess the efficiency of accurate customer classification:

Confusion matrix

The confusion matrix (CM) is a universal matrix in which true and false classification results are generated by constructing a classification model and comparing reality output with processing data outcomes. CM is a N*N matrix, with N denoting the total number of output variables. A 2*2 matrix is produced by binary models with two sets of values: yes – no; 0 – 1.



There are four types of Confusion Matrix:

- True positive (TP) if the model correctly predicts the positive class.

- True negative (TN) if the model correctly predicts the negative class.

- False positive (FP) if the model incorrectly predicts the positive class.

- False negative (FN) if the model incorrectly predicts the negative class.

Precision is defined as the ratio of the number of true positive points to those classed as positive (TP + FP) when determining a class to be positive.

The ratio of genuine positives to those that are truly positives (TP + FN) is described as **recall** (also known as sensitivity).

Mathematically, Precision and Recall are two fractions with the same numerator but different denominators:

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

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

Precision and Recall are both non-negative integers that are less than or equal to one.

The term "high precision" refers to the correctness of the points discovered. Strong True Positive Rate suggests a low rate of missing truly positive points, hence high recall means a low True Positive Rate.

F1-score, is the harmonic mean of precision and recall (assuming the two are non-zero):

$$\frac{2}{F_1} = \frac{1}{precision} + \frac{1}{recall}\ hay F_1 = 2\frac{1}{\frac{1}{precision} + \frac{1}{recall}} = 2\frac{precision.recall}{precision + recall}$$

The general case of the F1 score is the Fβ score:

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

F1 is a special case of F_β when β=1. When β >1 recall is given more importance than precision, when β 1, precision is given more importance. Two commonly used β quantities are β=2 and β=0.5.

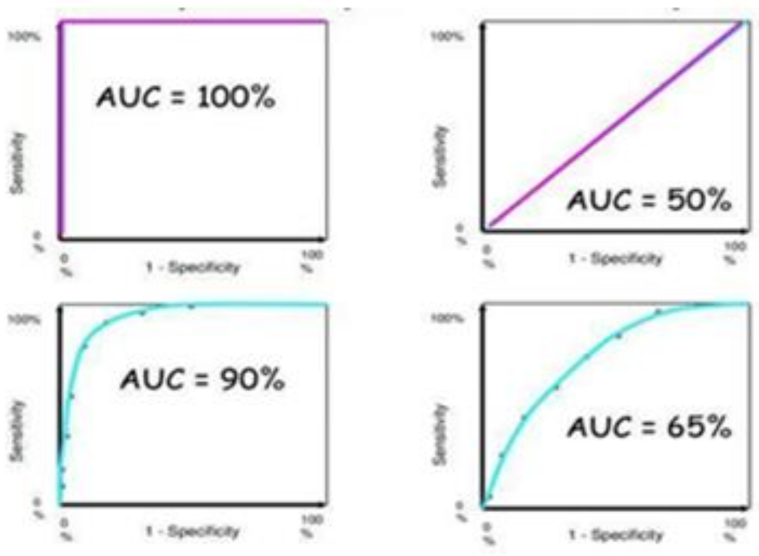
ROC curve

ROC (receiving operating characteristic) is a graphical plot. It is widely used in validating binary classification models. The curve is made by representing True positive rate (TPR), based on False positive rate (FPR) at different thresholds (we had explained about TPR and TFR above). As a result, the ROC curve shows the relationship, exchange and meaning of choosing the right model.

Area Under the ROC Curve (AUC)

By experimentalizing, in reality, if we only use ROC curve, we may find struggle to find out the differences between models, which one is closer to the left, the small gap but important of the two models.

Therefore, we look at AUC for efficiency measurement. The area is limited by the curve above and the horizontal axis.



AUC point will range from 0 to 1 (or 0% to 100%). The higher the AUC value is, the better the model is.

Table 4.9: Performance measure for logit

Logit	**Avg precise**	**Avg recall**	**AUC**	**Avg F1**	**Avg F2**	**Avg F0.5**
Complete data	91.11%	29.91%	81.99%	0.45038	0.345537	64.66%
KNN data	77.70%	**23.81%**	78.61%	**0.36455**	**0.27649**	**53.49%**
Missforest data	**95.10%**	17.08%	**78.63%**	0.289525	0.204274	49.69%
Multiple PCA data	84.63%	13.44%	72.79%	0.231912	0.161546	41.09%

Table 4.10: Performance measure of random forest

Logit	**Avg precise**	**Avg recall**	**AUC**	**Avg F1**	**Avg F2**	**Avg F0.5**
Complete data	92.85%	55.72%	90.78%	0.696467	0.605651	0.819323
KNN data	86.88%	**46.68%**	**88.24%**	**0.607351**	**0.514452**	**0.741195**
Missforest data	**99.10%**	29.50%	86.62%	0.454672	0.343223	0.673302
Multiple PCA data	89.51%	21.09%	79.42%	0.341389	0.248978	0.542885

Inclusion: Although the Precision and AUC metrics of the KNN method are slightly lower than Missforest, the three F-scores and recall are better in return showing that the KNN method can provide a suitable imputed dataset for credit classification. In summary, the KNN method provides a more balanced solution for imputing in data gaps, second is Missforest and third is Multiple PCA.

CHAPTER 5: DISCUSSION

5.1. Comparison of imputation methods

5.1.1. Bias performance

The filling impact, which can be seen in three tables 4.5-4.8, reveals that the Randomforest imputation approach delivers the lowest average (PB) deviation for the majority of missing values in the commercial banking data set (Average Percent Bias= 0.04806%). This method's PB value in all three data columns demonstrates a high level of efficiency. LAI SUAT - 0.00033%, DUNO_QD - 0.0106%, logarithm base_bal - 0.0083%, THOI HAN - 0.173%. In addition, the Randomforest recovering method's AW value is low, so that the data set's fill-in output impact is reliable.

Besides, MIPCA and KNN are two approaches with low deviation efficiency. And the median imputing method was the most simple method, however, produces the largest bias compared to the other ways. There are also data columns with an excessive percentage variance (5%) in this procedure (THOI HAN - 9.7239%). As a result, the approach is the worst method for imputing this commercial banking dataset.

5.1.2. Credit scoring performance

First, we can observe from the logit model that the KNN approach is best, it performed as good as the original data set. Table 4.9 shows that the majority of the scale values in the credit scoring of the KNN data set is the best in three imputed datasets. Furthermore, with an AUC of 78.63% and a precision of 95.10%, the Random forest technique is the second best for credit scoring. The MIPCA approach produces a lower AUC of 72.79%, and its precise value of 84.63% is also not so well performed.

The Random forest classification model is then used to evaluate the imputed datasets. The Random forest model's outcomes are listed in Table 4.10. As can be seen, the KNN approach results are still the best method categorizing credit clients. Moreover, the Random forest approach provides relatively good accuracy, with a precise value of 99.1% and an AUC of 86.62%. When the Random Forest model is used, the method's efficiency in forecasting customers to predict bad debt improves significantly. As can be observed, the PCA approach does not work so well with Random Forest models (precise-89.51%, auc-79,42%) as those models above, but it’s still good the main goal.

5.2. Discussion

Imputing in the data is a crucial part of data processing and model execution. If this step isn't done correctly, mistakes in evaluating and running label classification models will result.

This research not only assists the reader to understand the missing data categories, but it also presents a comparison in order to recommend the optimal strategy based on real data and simulated missing data. This research also introduces readers to essential data processing and variable selection techniques.

For the Vietnamese credit data, we can conclude that the KNN approach is the most successful method based on the results reported in the preceding section. This strategy not only delivers the best comprehension in training machine learning models for label classification, but it also produces a filled data set with very small bias. Furthermore, while both the MIPCA technique and the Random Forest produced quite good estimation of the missing values, both methods had limits when compared to each type of machine learning model. We also observed that from all the strategies, the median imputation method produced the worst outcomes.

At the same time, this study also indirectly shows the importance of financial variables

affecting the ability of customers to repay on time in the bank in the following order:,(I)

LAISUAT (interest rate); (II) DUNO_QD (debit balance); (III) Logarithm_base_bal (base balance); (IV) THOIHAN (duration); (V) SEX_female (female), (VI) PHUONG THUC CHO VAY_CV LAI GIU LAI MOT PHAN (partial retention loan type); (VII) PHUONG THUC CHO VAY _TRA GOP (installment loan type); ... With this, our study is beneficial not only for data analysts and data scientists in data processing, but also for credit rating organizations in determining how to classify consumers in order to avoid bad debt.

This research used a sophisticated data set that was many times larger than prior studies. Moreover, because this dataset has not only 2 labels but now 5 customer labels, we have altered and reorganized the labels to suit the classification model. In summary, we had to execute multiple phases of model pre-processing for new dataset that was complex, not clean, had a huge number of observations, and was very imbalanced. The results we present for both datasets, however, are consistent. As can be seen, the results we presented are reliable, and the method is not only suitable for small datasets but also for larger datasets.

However, the scope of this research is currently limited. Due to the large number of variables in the data set, the efficiency of advanced approaches such as PCA has not been proved. In fact, data imputations are time-dependent; KNN and Random forest algorithms will take a long time to fill in a huge and complicated data set, whereas PCA can do so quickly.

In the future, the team will perform tests with more advanced methods as well as further credit data. An imputation method test for a data set that has both qualitative and quantitative missing values should also be conducted.

5.3. Summary

To sum up, in this study we gave an overview on theoretical and empirical works on data imputation; we applied three popular imputation methods, K-nearest neighbors, random forest and principal component analysis for Vietnamese credit data. After that we used them as input for two credit scoring models: logit and random forest. After closely examining the outcomes, we could conclude that the three methods provide relatively good customer classification. However, the KNN is by far the most possible imputation method for our Vietnamese data set. All three methods show value estimations are close to original values and much better than the simple median imputation.

The only purpose for this is showing that these advanced methods can increase our ability to deal with uncertainty more than simple methods. We never hold ourselves accountable for missing values! We just anchor our ship in the'middle' of the sea whenever we discover a missing value, erroneously believing that our anchor has successfully fathomed the deepest pit of "uncertainties”. The goal is to keep the ship afloat by utilizing the resources at hand - wind speed and direction, star location, wave and tidal energy, and so on - to obtain the most 'diversified' catch for a higher return.

Our experiments suggested that in the future research these above imputation methods can be used as reliable ways to estimate the un-available information in missing data sets. If we have a big data set or time series-datasets, we could try first with PCA imputation because its running time is short. If the data is not so big, other two methods, especially the K-nearest-neighbors, are good to use. When using these imputed data for credit scoring problems, we have tested with typical Vietnamese data and the results were also close to the original set. We are going to check on some further data sets, in order to provide more empirical evidence for imputation results.

REFERENCES

Books

Bishop, C. M.. (2006). *Pattern recognition and machine learning*. Berlin: Springer.

Brownlee, J.. (2020). *How to Calculate Feature Importance With Python*. San Francisco: Machine Learning Mastery.

Articles

Ayilara, O. F., et al.. (2019). Impact of missing data on bias and precision when estimating change in patient-reported outcomes from a clinical registry. *Health and Quality of Life Outcomes*, *17*(1), 106. https://doi.org/10.1186/s12955-019-1181-2

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32. https://link.springer.com/article/10.1023/A:1010933404324

Buuren, S. van. (2021). Flexible Imputation of Missing Data. https://stefvanbuuren.name/fimd/sec-MCAR.html

Buuren, S. van & Groothuis-Oudshoorn, K.. (2010). mice: Multivariate Imputation by Chained Equations in R. *Journal of statistical software*, *10*(2), 1 – 68. https://dspace.library.uu.nl/handle/1874/44635

Demirtas, H., Freels, S. A. & Yucel, R. M.. (2008). Plausibility of Multivariate Normality Assumption When Multiply Imputing Non-Gaussian Continuous Outcomes: A Simulation Assessment. *Journal of Statistical Computation and Simulation, 78*(1), 69–84. https://doi.org/10.1080/10629360600903866

Enders, C. K.. (2017). Multiple imputation as a flexible tool for missing data handling in clinical research. *Behaviour Research and Therapy*, (), S0005796716301954–. https://doi.org/10.1016/j.brat.2016.11.008

Florez-Lopez, R.. (2010). Effects of missing data in credit risk scoring. A comparative analysis of methods to achieve robustness in the absence of sufficient data. *Journal of the Operational Research Society*, *61*(3), 486–501. https://doi.org/10.1057/jors.2009.66

Gajawada, S. & Toshniwal, D.. (2012). Missing Value Imputation Method Based on Clustering and Nearest Neighbours. *International Journal of Future Computer and*

Communication, *1*(2), 206-208. http://www.ijfcc.org/index.php?m=content&c=index&a=show&catid=34&id=299

Ilin, A. & Raiko, T.. (2010). Practical Approaches to Principal Component Analysis in the Presence of Missing Values. *Journal of Machine Learning Researc*h, 11, 1957-2000. https://jmlr.csail.mit.edu/papers/v11/ilin10a.html

Ispirova, G., Eftimova, T. & Seljaka, B. K.. (2020). Evaluating missing value imputation methods for food composition databases. *Food and Chemical Toxicology, 141*(), 111368-. https://doi.org/10.1016/j.fct.2020.111368

Josse, J. & Husson, F.. (2012). Handling missing values in exploratory multivariate data analysis methods. *Journal de la Société Française de Statistique*, *153*(2), 79-99. http://journal-sfds.fr/article/view/122

Kang, H.. (2013). The prevention and handling of the missing data. *Korean Journal of Anesthesiology*, *64*(5):402-406. https://doi.org/10.4097/kjae.2013.64.5.402

Kokla, M., et al.. (2019). Random forest-based imputation outperforms other methods for imputing LC-MS metabolomics data: a comparative study. *BMC Bioinformatics*, *20*(1), 492. https://doi.org/10.1186/s12859-019-3110-0

Kowarik, A., & Templ, M.. (2016). Imputation with the R Package VIM. *Journal of Statistical Software*, *74*(7). https://doi.org/10.18637/jss.v074.i07

Malarvizhi, R., Thanamani, A. S..(2012). K-Nearest Neighbor in Missing Data Imputation. *International Journal of Engineering Research and Development*, *5*(1), 05-07. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.375.925&rep=rep1&type=pdf

Minakshi, Rajan & Gimpy, V.. (2014). Missing Value Imputation in Multi Attribute Data Set. *International Journal of Computer Science and Information Technologies*, *5*(4) , 5315-5321. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.451.6606&rep=rep1&type=pdf

Newman, D. A.. (2014). Missing Data: Five Practical Guidelines. *Organizational Research Methods*, *17*(4), 372-411. https://doi.org/10.1177/1094428114548590

Pantanowitz, A. & Marwala, T.. (2009). Missing Data Imputation Through the Use of the Random Forest Algorithm. *Advances in Computational Intelligence*, *116*(), 53-62. DOI:10.1007/978-3-642-03156-4_6

Pedersen, A. B., et al.. (2017). Missing data and multiple imputation in clinical epidemiological research. *Clinical Epidemiology*, *9*(), 157-166. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5358992/

Phan Thị Thu Hồng. (2020). So sánh một số phương pháp xử lý dữ liệu thiếu cho chuỗi dữ liệu thời gian một chiều. *Tạp chí Khoa học Nông nghiệp Việt Nam*, *19*(4), 452-461. http://tapchi.vnua.edu.vn/wp-content/uploads/2021/04/tap-chi-so-4.2021.4.pdf

Stekhoven, D. J.. & Bühlmann, P.. (2011). MissForest-nonparametric missing value imputation for mixed-type data. *Bioinformatics*, *28*(1), 112–118. https://doi.org/10.1093/bioinformatics/btr597

Tang, F & Ishwaran, H.. (2016). Random forest missing data algorithms. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, (), 1-15. https://doi.org/10.1002/sam.11348

Young, R. & Johnson, D. R. (2015). Handling Missing Values in Longitudinal Panel Data With Multiple Imputation. *Journal of Marriage and Family*, *77*(1), 277–294. https://doi.org/10.1111/jomf.12144