NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCE
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATION

BSc THESIS

Analysis of monthly payment delays using machine learning

Efthymia C. Malesiou

Supervisor: **Ioannis Emiris,** Professor

ATHENS

OCTOBER 2021

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση της καθυστέρησης των μηνιαίων πληρωμών με χρήση μηχανικής μάθησης

Ευθυμία Χ. Μαλέσιου

Επιβλέπων: **Ιωάννης Εμίρης,** Καθηγητής

ΑΘΗΝΑ

ΟΚΤΩΒΡΙΟΣ 2021

BSc THESIS

Analysis of monthly payment delays using machine learning

Efthymia C. Malesiou
S.N.: 1115201500229

SUPERVISOR: **Ioannis Emiris,** Professor

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση της καθυστέρησης των μηνιαίων πληρωμών με χρήση μηχανικής μάθησης

Ευθυμία Χ. Μαλέσιου
Α.Μ.: 1115201500229

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Ιωάννης Εμίρης,** Καθηγητής

ABSTRACT

The prediction of the delay of monthly payments concerning long-term customers with or without contracts, is valuable to financial planning, cash-flow forecasting, making strategic choices to reduce losses and factoring in general. Especially for small and medium enterprises, it has been estimated that up to half of theιρ invoices are paid late, thus creating a significant problem. The estimation of the expected delay combined with the corresponding probability, allows the ranking of customers according to the risk of loss.

Usually, the type of product or service offered by the company, affects the available features, which consequently have different importance in the prediction process. Additionally, often the volume of data collected from the customers is huge and they are distributed on different databases and have varying quality.

In this thesis, both classification (late, non-late) and regression models (days until the bill is settled) are evaluated, using minimal information from the current bill and the customer's history. Furthermore, additional features are generated that summarize the customer's profile up to a specific date and capture recent trends, without including any information not known at the time of issuing the bill. Thus, the focus is on the customer's behaviour without a strict time component, as in the classic time-series.

Initially, basic machine learning algorithms that are often encountered in relevant applications in the literature are evaluated and then ensemble learning methods are tested, utilizing the basic models. Finally, their performance is compared to that of models that use classic time-series.

SUBJECT AREA: Machine Learning

KEYWORDS: late payment prediction, customer profile, time-series, classification, regression

ΠΕΡΙΛΗΨΗ

Η πρόβλεψη της καθυστέρησης πληρωμής των μηνιαίων τιμολογίων πελατών με συμβόλαια μακροχρόνιας δέσμευσης ή μακροχρόνια συνεργασία, βρίσκει εφαρμογή στον χρηματοοικονομικό σχεδιασμό, στην πρόβλεψη της ρευστότητας, στην επιλογή στρατηγικής για μείωση των απωλειών καθώς και γενικότερα στην αναδοχή επιχειρηματικών απαιτήσεων (factoring). Ειδικά για τις μικρομεσαίες επιχειρήσεις, έχει εκτιμηθεί πως έως και τα μισά τιμολόγια εξοφλούνται με καθυστέρηση, δημιουργώντας έτσι σημαντικό πρόβλημα. Η κατηγοριοποίηση (classification) ως προς την αναμενόμενη καθυστέρηση πληρωμής σε συνδυασμό με την εκτιμώμενη πιθανότητα αυτού του γεγονότος, επιτρέπουν την κατάταξη των πελατών ως προς τον κίνδυνο απωλειών.

Ο τύπος των προϊόντων ή υπηρεσιών που προσφέρονται από την επιχείρηση, συνήθως επηρεάζει τα διαθέσιμα χαρακτηριστικά, τα οποία κατ'επέκταση αποκτούν διαφορετική βαρύτητά στη διαδικασία πρόβλεψης ενώ συχνά ο όγκος των συνολικών δεδομένων που συλλέγονται για τους πελάτες είναι τεράστιος, κατανεμημένος σε διαφορετικές βάσεις και με διαβαθμιζόμενη ποιότητα.

Στην παρούσα πτυχιακή, ελέγχεται η αποτελεσματικότητα πρόβλεψης τόσο της κλάσης (πληρωμή με καθυστέρηση ή χωρίς καθυστέρηση) όσο και των ημερών που μεσολαβούν από την έκδοση του λογαριασμού έως την πληρωμή του, αξιοποιώντας ελάχιστα χαρακτηριστικά από το τρέχον τιμολόγιο και το ιστορικό των πελατών. Από αυτά, παράγονται πρόσθετα χαρακτηριστικά που συνοψίζουν το προφίλ του πελάτη έως τη δεδομένη στιγμή και πρόσφατες τάσεις, χωρίς να περιλαμβάνεται οποιαδήποτε πληροφορία δεν είναι γνωστή κατά τη στιγμή έκδοσης του λογαριασμού. Έτσι, το ενδιαφέρον εστιάζεται στη συμπεριφορά των πελατών χωρίς αυστηρή χρονική συνιστώσα, όπως στις κλασικές χρονοσειρές.

Αρχικά, αξιολογούνται βασικοί αλγόριθμοι μηχανικής μάθησης που συναντώνται συχνά σε σχετικές εφαρμογές στη βιβλιογραφία και στη συνέχεια ελέγχονται μέθοδοι συνολικής μάθησης (ensemble learning), αξιοποιώντας τα βασικά μοντέλα. Τέλος, η αποτελεσματικότητά τους συγκρίνεται και με μοντέλα που χρησιμοποιούν κλασικές χρονοσειρές.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Μηχανική Μάθηση

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: πρόβλεψη καθυστέρησης πληρωμής, προφίλ πελάτη, κατηγοριοποίηση, παλινδρόμηση, χρονοσειρές

«…πᾶσά τε ἐπιστήμη χωριζομένη δικαιοσύνης καὶ τῆς ἄλλης ἀρετῆς πανουργία, οὐ σοφία φαίνεται.», Μενέξενος, Πλάτωνας

"All sentient beings should have at least one right – the right not to be treated as property", Gary L. Francione

AKNOWLEDGMENTS

I would like to thank my supervisor, Professor Ioannis Emiris for giving me the opportunity to work on this topic and his guidance, and also the PhD student Emmanouil Christoforou for his advice.

CONTENTS

1. INTRODUCTION 19

1.1 Topic 19

1.2 Related work 19

1.3 Objectives 22

1.4 Organisation 22

2. THEORETICAL BACKGROUND 23

2.1 Feature selection criteria 23

2.1.1 Mutual information 23

2.1.2 Analysis of Variance (ANOVA) F-score 24

2.1.3 Chi-squared test 24

2.2 Dimensionality reduction algorithms 24

2.2.1 Principal Component Analysis (PCA) 24

2.2.2 Uniform Manifold Approximation and Projection (UMAP) 24

2.2.3 Gaussian random projection 24

2.3 Base models 25

2.3.1 Quadratic Discriminant Analysis 25

2.3.2 Gaussian Naïve Bayes 25

2.3.3 K-nearest neighbors 26

2.3.4 Support Vector Machine 26

2.3.5 Decision tree 26

2.3.6 Neural networks 27

2.4 Ensemble learning methods 28

2.5 Classic time-series forecasting 28

2.5.1 Prophet 28

2.6 Hyperparameter tuning 28

2.6.1 Genetic algorithm 28

2.6.2 Grid search 28

2.7 Evaluation metrics 29

2.7.1 Accuracy 29

2.7.2 RMSE 29

3. METHOD AND RESULTS 30

3.1 Dataset 31
3.1.1 Features 31
3.1.2 Data exploration 32

3.2 Preprocessing 52
3.2.1 Transformation / encoding 52
3.2.2 Derived features 52
3.2.3 Dimensionality reduction 54
3.2.4 Feature selection 56
3.2.5 Splitting methods 57

3.3 Models 59
3.3.1 Hyperparameter selection 59
3.3.2 Classification 59
3.3.3 Regression 60

3.4 Evaluation 61
3.4.1 Classification 61
3.4.2 Regression 68
3.4.3 Classic time-series forecasting - Prophet 73

3.5 Technical information 77

4. CONCLUSIONS AND FUTURE PROSPECTS 78

TABLE OF TERMINOLOGY 80

ABBREVIATIONS - ACRONYMS 81

REFERENCES 82

LIST OF FIGURES

Figure 1: Abstract representation of the quadratic discriminant analysis classifier. 25

Figure 2: Abstract representation of the naïve Bayes classifier. 25

Figure 3: Abstract representation of the k-NN classifier. 26

Figure 4: Abstract representation of the support vector machine classifier. 26

Figure 5: Abstract representation of the decision tree classifier. 27

Figure 6: Abstract representation of the neural network. 27

Figure 7: The workflow 30

Figure 8: Example of the functions used to train and tune the hyperparameters of a base model (random forest) using the 15 most important features and random split. ...31

Figure 9: Heatmap of the correlation between the features. As it was expected, the DaysToSettle, DaysLate and Late features are highly correlated (positively). A less strong correlation is observed with the `disputed` feature. 33

Figure 10: Barplot of the number of late and non-late bills (absolute frequency). Approximately, 33% of the invoices are late. 33

Figure 11: Histogram of the number of days of delay (date of settling - due date). Most of the invoices (64%) are settled with no delay and the vast majority (80%) of the late ones are settled within 15 additional days; the range is 0 to 45 days and the mean value 3.4 days 34

Figure 12: Histogram of the number of days until the invoice is settled (date of settling - issuing date). Most of the invoices (64%) are settled within 30 days and 80% are settled within 40 days; the range is 0 to 75 days and the mean value 26.4 days. The distribution is approximately normal. 34

Figure 13: Barplot of the number of disputed and non-disputed bills (absolute frequency). Approximately, 23% of the invoices are disputed. 35

Figure 14: Histogram of the new amount (not including any previous outstanding / unpaid amount) per invoice. The mean value is 59.9 and the amount ranges between 5.26 and 128.28. The distribution is approximately normal. 35

Figure 15: Barplot of the type of invoice (paper or electronic; absolute frequency). Approximately, 49% of the invoices are paper and 51% electronic. 36

Figure 16: Barplot of the number of disputed and non-disputed bills that are late or non-late (absolute frequency). Clearly, it is much more likely for disputed bills to be late; 68% of the disputed bills are late in contrast with only 26% of the non-disputed bills.....36

Figure 17: Histogram of the date (year-month) that the customers opted for electronic invoice. All the dates are within the range of the invoices' issuing dates.....37

Figure 18: Boxplots of the amount according to whether the invoice was disputed or not. The mean amount of the disputed bills is slightly greater. 37

Figure 19: Boxplots of the number of days the bills were late according to whether the invoice was disputed or not. The disputed bills have a greater mean value as well as a greater range.....38

Figure 20: Boxplots of the number of days until the bills were settled according to whether the invoice was disputed or not. The disputed bills have a greater mean value. 38

Figure 21: Barplot of the number of non-disputed and disputed electronic bills (absolute frequency).....39

Figure 22: Boxplots of the number of days the bills were late according to the type of invoice (paper, electronic). Electronic invoices tend to be late by more days compared to the paper ones with the mean values being 2.4 and 4.4, respectively.....39

Figure 23: Boxplots of the number of days to settle the bill according to the type of invoice (paper, electronic). Electronic invoices tend to be settled later compared to the paper ones with the mean values being 24.6 and 30.2, respectively.....40

Figure 24: Barplot of the number of invoices per year. 40

Figure 25: Barplot of the number of invoices per month, January to December (absolute frequency). They seem to be evenly distributed; the invoices of December mainly correspond to the year 2012 since the last invoice of 2013 was issued on December 3. 41

Figure 26: Barplot of the number of invoices per month per year (absolute frequency). They seem to be evenly distributed; the last invoice of 2013 was issued on December 3. 41

Figure 27: Barplot of the number of invoices per day of the month (absolute frequency). They seem to be evenly distributed; the 31st day has approximately half the frequency of the other days, as expected.....42

Figure 28: Barplot of the number of days the bill was late (absolute frequency), excluding the non-late bills. 42

Figure 29: Barplot of the number of days to settle the bill (absolute frequency). The distribution is approximately normal. 42

Figure 30: Barplot of the sum of the invoice amounts per year. The mean sum amount per month per year is approximately the same; 6,339 and 6,512, respectively. 43

Figure 31: Barplot of the sum of the invoice amounts per month. The values are approximately the same; the last invoice of 2013 was issued on December 3 and thus the sum is nearly half, as expected. 43

Figure 32: Scatter plot of the invoice amounts to the number of days the bills were late, coloring the points according to whether they were disputed or not. No correlation is observed.....44

Figure 33: Scatter plot of the invoice amounts to the number of days to settle the bills, coloring the points according to whether they were disputed or not. No correlation is observed.....44

Figure 34: Plot of the amount to the number of days the bill was late ratio. Most of the bills have a low ratio (< 10) while it ranges from 0.16 to 71.02. Greater values correspond to bills being paid immediately or within a few days.....45

Figure 35: Barplot of the number of invoices per country (absolute frequency). The first three countries (391, 406, 770) have more invoices than the last two (818, 897).....45

Figure 36: Barplot of the mean number of days the bills were late per country. The mean value of the first country (391) is approximately 50% lower compared to the other countries.....46

Figure 37: Barplot of the mean number of days until the bills are settled per country. The mean value of the first country (391) is slightly lower than that of the other countries but the difference is not as marked compared to the mean number of days the bills were late.....46

Figure 38: Barplot of the mean amount per country. The mean value of the fifth country (897) is much lower (23% - 43%) than that of the other countries.....47

Figure 39: Barplot of the number of late and non-late invoices per country. Clearly, the ratio of late bills is lower in the first country (391). 47

Figure 40: Barplot of the number of paper and electronic invoices per country. The last two countries (818, 897) have a lower ratio of paper to electronic bills. 48

Figure 41: Barplot of the number of disputed and non-disputed invoices per country. The second and fourth countries (406, 818) have a greater ratio of disputed to non-disputed bills. 48

Figure 42: Barplots of the number of bills (1st), the mean amount (2nd) and the percentage of electronic bills per customer (3rd). 49

Figure 43: Barplots of the mean number of days the bill was late (1st) and the mean number of days to settle the bill (2nd) along with the percentage of late and disputed bills per customer (3rd). 49

Figure 44: Plot depicting the date range of the invoices per customer along with their status (late, disputed). 50

Figure 45: Barplot of the number of unique customers per country (absolute frequency). The first three countries (391, 406, 770) have more customers than the last two (818, 897). This aligns with the number of invoices per country. 51

Figure 46: Barplot of the number of unique customers per month. The values are approximately the same. 51

Figure 47: Barplot of the number of unique customers per month per year (absolute frequency). They seem to be evenly distributed; the last invoice of 2013 was issued on December 3. 52

Figure 48: The ground truth (blue: non-late, red: late) in the three-dimensional space of the invoice amount bin x the number of months x the issuing month, and the results of the KMeans, Gaussian Mixture Model and kNN algorithms; kNN seems to perform better. 53

Figure 49: The variance explained per principal component. 54

Figure 50: The top contributing features to the variance of the first three principal components. The red line indicated the mean contribution across all the features. 54

Figure 51: PCA projections; the late (red) and non-late (blue) bills cannot be separated. 55

Figure 52: UMAP projections; the late (red) and non-late (blue) bills cannot be separated.....55

Figure 53: Gaussian random projection; the late (red) and non-late (blue) bills cannot be separated.....55

Figure 54: Barplot of the features' importance using the mutual information criterion. ..56

Figure 55: Barplot of the features' importance using the ANOVA F-score criterion. 56

Figure 56: Heatmap of features' rank using 2 criteria (mutual information, ANOVA F-score) for the purpose of classification. 57

Figure 57: The accuracy score of the classification models (ordered) on the training and test sets (random split, all features). 62

Figure 58: Confusion matrices of the top-performing models (random split, all features). 63

Figure 59: Histogram of the probability of delay using the XGBoost classifier (best model).....63

Figure 60: Histogram of the probability of the predicted class in the case of misclassification using the XGBoost classifier (best model). 64

Figure 61: The features' importance using the XGBoost classifier (best model, all features, random split). The gain, refers to the average gain across all the splits and the weight to the number of times the feature is used to split the data across all the trees. 64

Figure 62: Density scatter plot of the SHAP values [30] for each feature using the XGBoost classifier (gbtree booster instead of gblinear (best model), all features, random split), without (first plot) or without (second plot) random permutation of the target values. The features are sorted by the sum of the SHAP value magnitudes across all bills. 65

Figure 63: The features' importance using the XGBoost classifier (best model, all features, random split) and MDA (Mean Decrease Accuracy) metric. 66

Figure 64: The features' importance using the NN classifier (best model, all features, random split) and MDA (Mean Decrease Accuracy) metric. 67

Figure 65: The accuracy of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset. 68

Figure 66: The two-dimensional representation of customer's `w` history. 68

Figure 67: Histograms of the number of days until the bill is settled and the number of days the bill was late, along with the cardinality of the corresponding classes used to calculate the “accuracy” of the regression models.....69

Figure 68: Confusion matrices of the top-performing models (random split, all features). Although the “accuracy” is low, most of the misclassified cases belong to the previous or next “class”. 70

Figure 69: The absolute difference between the actual and predicted values of the test set, using the top-performing models (random split, all features). 71

Figure 70: The features’ importance using the XGBoost regressor (all features, random split) and MDA (Mean Decrease Accuracy) metric. 72

Figure 71: The features’ importance using the NN regressor (all features, random split) and MDA (Mean Decrease Accuracy) metric.....72

Figure 72: Prophet results - prediction of the 5th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. Even though floor and cap values are defined, the predictions are sometimes out of this range. The predictions having values >> cap, contribute to the higher accuracy observed in the last class. 74

Figure 73: Prophet results - prediction of the 10th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.....74

Figure 74: Prophet results - prediction of the 15th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.....75

Figure 75: Prophet results - prediction of the 20th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.....75

Figure 76: Prophet results - prediction of the 25th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. The predictions are much better now that 24 bills are available to fit the models. 76

Figure 77: Prophet results - prediction of the 30th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. There are only 19/100 customers having 30 bills.....76

LIST OF TABLES

Table 1: Summary of related papers. The best models are in bold. 20

Table 2: The basic models and the ensemble methods used in the papers and in the thesis. 23

Table 3: The accuracy score of the classification models (ordered) on the test set (random split, all features). 62

Table 4: The accuracy of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset. 67

Table 5: The RMSE score of the regression models (ordered) on the test set (random split, all features). 69

Tale 6: The results of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset. 73

1. INTRODUCTION

1.1 Topic

The delay of monthly payments concerning long-term customers with or without contracts, create significant problems to the stability and growth of businesses and is a great source of uncertainty. The prediction of that delay, is valuable to financial planning, cash-flow forecasting, making strategic choices to reduce losses and factoring in general. This is especially true for small and medium-sized businesses, for which it has been estimated that up to half of the invoices they issue are paid late [5]. The estimation of the expected delay combined with the corresponding probability, allows the ranking of customers according to the risk of loss. Subsequently, the financial planning may adapt and preventive actions can be taken to increase the revenue.

Often the volume of data collected for the customers is huge and the data are distributed on different databases with varying quality, since non-automatic entries are part of the process. In addition, depending on the type of the product or service, different characteristics, which sometimes require significant preprocessing, may be considered important.

A general purpose model that would use a limited number of common features, produced by the customer's history and recent trends, without significant loss of accuracy, would prove particularly useful.

1.2 Related work

Table 1, summarises the related literature, focusing on the type of data used and their origin, the models tested and the accuracy achieved:

- The data come from various sectors; national and international banks, energy organisations, telecommunication, broadcasting, logistics, oilfield services and high-tech equipment companies.
- The invoices span a period of 3 months to 2 years.
- The number of records ranges from 25,000 to 45,000,000 and the number of unique customers is between 700 and 1,600,000.
- The accuracy ranges from 66% to 97% and there is no apparent correlation with the number of records, the number of customers or the time period.
- The highest accuracy (97% [5]) was achieved with more complex data, immediately related to the actions of the customer and the company, millions of records, logistic regression and pre-clustering of the customers.
- XGBoost was most commonly the best model.
- Unified models (trained on different datasets), sometimes outperformed the individual models.

Table 1: Summary of related papers. The best models are in bold.

Year / Ref.	Data	Features	Models	Max. accuracy
2020 [4]	91,562 invoices, 2,229 customers, 6 countries, 2 years (international bank)	Paid invoice, Total paid invoices, Sum amount paid invoices,Total invoices late, Sum amount late invoices, Total outstanding invoices, Total outstanding late, Sum total outstanding, Sum late outstanding, Average days late, Average days outstanding late, Standard deviation invoices late, Standard deviation, Invoices outstanding late, Payment frequency difference	Logistic Regression, Naïve Bayes, Random Forest, **XGBoost**, k-NN, Deep Neural Network	81%
2020 [5]	45 million records, 1.6 million customers, 2 countries, 1 year (international telecommunications company)	Customer ID, Action type (e.g. e-mail, SMS, phone call), Action date, Stage changer flag (payment through the banking system, unpaid invoice occurrence or action timer flag)	**Logistic regression**, One Rule, SVM +/- pre-clustering of customers with DBSCAN	97%
2020 [6]	5.05 million bills, 1 country - Australia (energy organisation)	Age range, Month and year of a bill, Weekly median household income in the living area, Bill duration, Average household Size in the living area, Remoteness of the living area etc. (34 features)	**XGBoost**, Random Forest, Decision Tree, Bayesian Neural Network, Deep Neural Network, Logistic Regression, Naïve Bayes	68%
2019 [3]	175,552 invoices, 725 customers, 8 countries (multinational bank)	Paid invoice, total paid invoices, Sum amount paid invoices, Total invoices late, Sum amount late invoices, Total outstanding invoices, Total outstanding late, Sum total outstanding, Sum late outstanding, Average days late, Average days outstanding late, Standard deviation invoices late, Standard deviation invoices outstanding late, Payment frequency difference	Naïve Bayes, Logistic Regression, k-NN, Random Forest, **XGBoost**	77%
2019 [33]	25,000 records, 1 year, 1 country -	Amount of the given credit, Gender, Education, Marital status, Age, History of past payments,	Neural Networks, Decision Trees,	89%

	Taiwan (credit cards)	Amount of bill statement, Amount of previous payments	**Random Forest**	
2018 [11]	10,562 customers (logistics company)	Invoice, Intervention Actions, Revenue, Payment Type, Customer, Air bill, Billing cycle, Calculated pureness measure (17 features)	Probability Tree, Misclassification Tree, Regression, Polynomial Regression, Neural Network, Regression with **Neural Network,** Ensemble Model	N/A
2016 [25]	~38,000 customers, 4 months, 1 country - South Korea (cable broadcasting company)	Customer ID, Age, Gender, Status, Tenure, Payment methods, Product category, Bond category, Unpaid amount, Unpaid internet, Unpaid digital, Net cable, Net digital etc. (23 features)	**Decision tree**, Random forest, Neural network, SVM	N/A
2015 [22]	~210,000 invoices, 3 months, 90 countries (oilfield services company)	Amount, Customer number, Invoice date, Handler, Customer number, Document date, Posting date, Document currency, Clearing date, Entry date, Division, Payment term, Credit representative, No. paid invoices, No. delayed invoices, Total amount paid, etc.	Classification tree, **Random forest**, Adaptive boosting, Logistic regression, SVM	89%
2009 [38]	25,000 records, 1 year, 1 country - Taiwan (credit cards)	Amount of the given credit, Gender, Education, Marital status, Age, History of past payments, Amount of bill statement, Amount of previous payments	**k-NN**, Naïve Bayes, logistic regression, Discriminant analysis, Neural network, Classification tree	N/A
2008 [39]	~170,000 invoices, 4 firms (high-tech equipment companies)	Invoice base amount, Payment term, Category (under dispute or not), Number of total paid invoices, Number of invoices that were paid late, Ratio of paid invoices that were late, Sum of the base amount of total paid invoices, Sum of the base amount of invoices that were paid late, Ratio of sum of paid base amount that were late, Average days late of paid invoices being late, Number of total outstanding invoices, Number of outstanding invoices that were already late, Ratio of outstanding invoices that	**Decision tee**, Naïve Bayes, Logistic Regression, Boosting decision stumps, PART	96%

		were late, Sum of the base amount of total outstanding invoices, Sum of the base amount of outstanding invoices that were late, Ratio of sum of outstanding base amount that were late, Average days late of outstanding invoices being late		

1.3 Objectives

The objectives of the thesis, are to:

- Derive features that effectively summarise the customer's history and capture recent trends from minimal information of the invoices, making the process anonymous and context-independent.
- Select a subset of the features that does not significantly affect the performance, in order to reduce the dimensionality.
- Generate machine learning models and fine-tune their hyperparameters to predict the payment delay status of the invoices as well as the number of days until they are settled.
- Evaluate the performance of basic machine learning algorithms commonly utilized to predict the delay in payment along with ensemble learning methods, using the aforementioned data.

1.4 Organisation

The theoretical background of the algorithms / procedures used that are mentioned in the Method is summarised in Chapter 2. The method is described in Chapter 3, where the results are also presented. The general conclusions and the future prospects are presented in Chapter 4.

2. THEORETICAL BACKGROUND

2.1 Feature selection criteria

This chapter provides a brief overview of the methods mentioned in Chapter 3. Table 2, lists the models used in the literature and the corresponding frequency, indicating whether they were part of the tests performed here.

Table 2: The basic models and the ensemble methods used in the papers and in the thesis.

Type	Name	Frequency	Best model	Used
basic	Logistic Regression	8/10	1/10	Yes
basic	Decision Tree	7/10	2/10	Yes
basic	Neural Network	6/10	1/10	Yes
basic	Naïve Bayes	5/10	0/10	Yes
basic	Support Vector Machine	4/10	0/10	Yes
basic	k-Nearest Neighbors	3/10	1/10	Yes
basic	Discriminant Analysis	1/10	0/10	Yes
basic	OneR	1/10	0/10	No
basic	PART	1/10	0/10	No
basic	Gaussian Mixture Model	0/10	0/10	Yes (as feature)
basic	K-Means	0/10	0/10	Yes (as feature)
ensemble	Random Forest	6/10	2/10	Yes
ensemble	XGBoost	2/10	3/10	Yes
ensemble	AdaBoost	1/10	0/10	Yes
ensemble	Gradient Boosting	1/10	0/10	Yes
ensemble	Stacking	1/10	0/10	Yes
ensemble	Boosting Decision Stumps	1/10	0/10	No

2.1.1 Mutual information [26]

It is a measure of the mutual dependence of a set of variables, estimating the amount of information that can be obtained about one of them just by knowing the value of the other. The higher the value the less the uncertainty is, whereas a zero value means that the variables are independent, both linearly and nonlinearly. So, features irrelevant to the target can be eliminated.

2.1.2 Analysis of variance (ANOVA) F-score [15]

The ANOVA F-score is used to order the features according to their significance. It determines if the variance between the variables is significantly different, indicating if a particular variable changes significantly when the other variables change.

2.1.3 Chi-squared test [12]

The chi-squared test, determines if there is statistically significant difference between a non-negative feature and the target, evaluating the dependence between stochastic features. It assigns a higher value to those relevant to the target / class, allowing to sort the features accordingly.

2.2 Dimensionality reduction algorithms

2.2.1 Principal components analysis (PCA) [27]

It is a widely used linear method and relatively fast, based on the assumption that the data distribution is approximately normal. It has the ability to maintain both local and global distances and allows the comparison between clusters, facilitating the identification of outliers. The data are transformed and displayed in a new system of linearly uncorrelated principal components, based on variance; the first principal component corresponds to the maximum variation and each subsequent one is selected with the same criterion given that it is perpendicular to the previous ones.

2.2.2 Uniform manifold approximation and projection (UMAP) [24]

This is a stochastic, non-linear method aiming to learn a low dimensional manifold while partially maintaining regional (considered linear) relationships as well as global relationships according to the sensitive hyperparameters specified (e.g. perplexity, number of neighbors), using cross-entropy as the metric. The cost function has many local minima increasing the likelihood of convergence on one of them. The variance is increased in sparse regions as opposed to dense regions, thus enhancing local relationships and avoiding the concurrence of intermediate observations. Student's t-distribution is used for the distances in the low dimensional embedding.

2.2.3 Gaussian random projection [13]

The data are projected to a randomly generated matrix using components that result from the normal distribution N(0, 1 / no. components) and have unit length. The pairwise distances are approximately preserved.

2.3 Base models

2.3.1 Quadratic discriminant analysis [35]

This classifier uses a quadratic decision surface to determine the class of the observations, without assuming that the classes have the same covariance. It is essentially a generalised form of the linear classifier (figure 1).

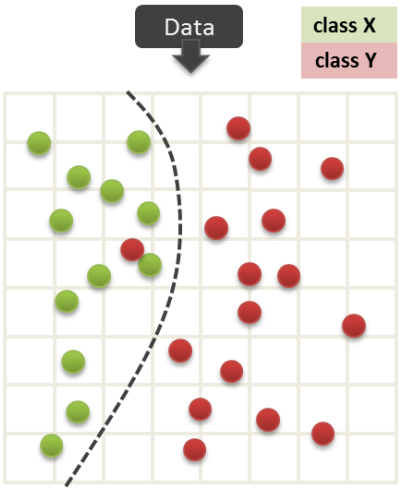

Figure 1: Abstract representation of the quadratic discriminant analysis classifier.

2.3.2 Gaussian naïve Bayes [19]

This Naïve Bayes classifier simply works on the assumption that each feature follows the normal distribution, is conditionally independent of any other and is as important in predicting the class as any other feature. The maximum posterior probability calculated determines the class of an observation (figure 2).

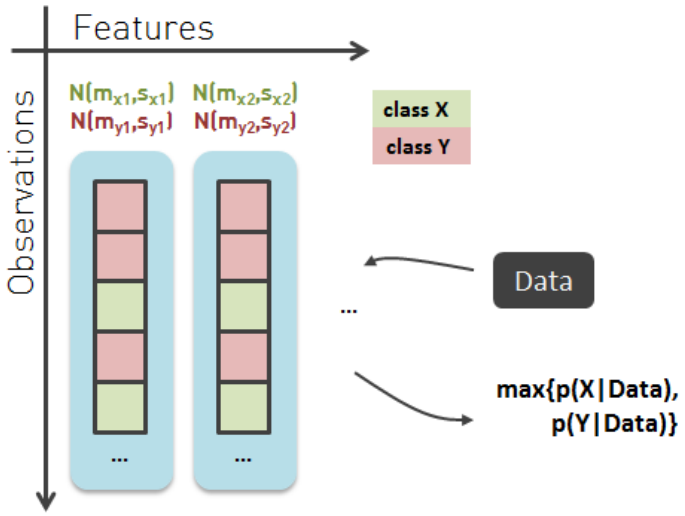

Figure 2: Abstract representation of the naïve Bayes classifier.

2.3.3 K-Nearest neighbors [2]

Using the k-NN algorithm, a new observation is assigned to the class that the majority of its k nearest neighbors of the training set belong to, using a specified metric function (figure 3).

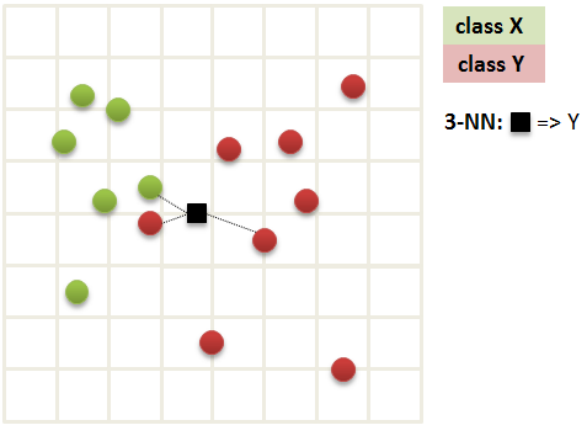

Figure 3: Abstract representation of the k-NN classifier.

2.3.4 Support vector machine [8]

The goal of this classifier is to find hyperplanes that make the observations of the two classes have the maximum possible gap between them, dividing the mapped feature space in regions assigned to a particular class (figure 4).

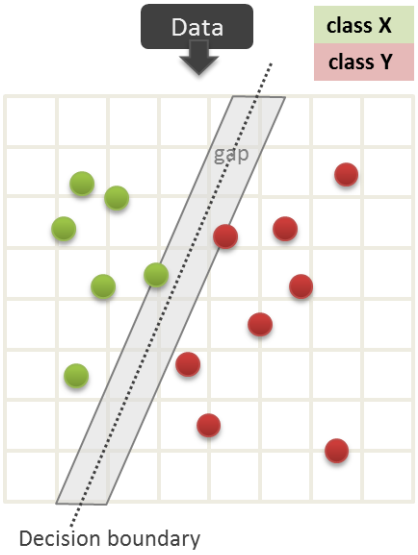

Figure 4: Abstract representation of the support vector machine classifier.

2.3.5 Decision tree [24]

A decision tree is a set of decision rules used at internal nodes of the tree-structure to determine how they split into branches and are optimised based on the training data in a greedy manner (figure 5). The bias is typically increased but the preprocessing

requirements are minimal. When it is restricted to just one level, it is called, decision stump or one-rule [23].

The PART algorithm [16] generates a partial pruned decision tree using a separate-and-conquer method.

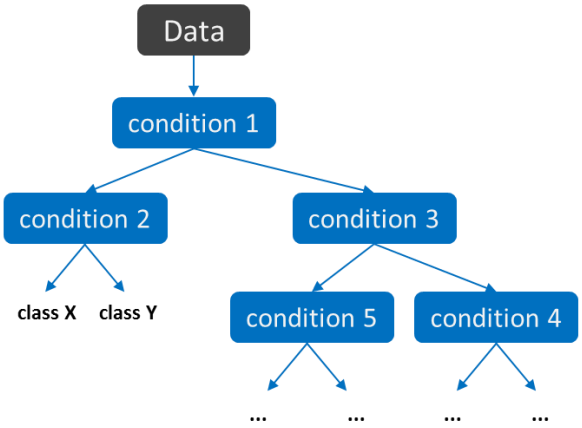

Figure 5: Abstract representation of the decision tree classifier.

2.3.6 Neural networks [21]

In its simple form, a neural network is a sequence of interconnected layers having varying number of nodes, where opermations are performed. The input layer receives the data and leads them via its connections through any existing hidden layers for further processing, until they finally reach the output layer where the final result is calculated (figure 6).

The perceptron [18] is a binary classifier having only one layer - the input layer. The nodes are assigned different weights and are directly connected to the output node. The output value results from the activation function (e.g. unit step) that uses the sum of the inner product of the data and the weights vector plus a bias value as its input.

A more complex type of NN used, is the LSTM (Long short-term memory) [20], where loops / feedback connections allow for information to be retained; there are gates to control what information is considered important and what should be forgotten, without being restricted to more recent events.

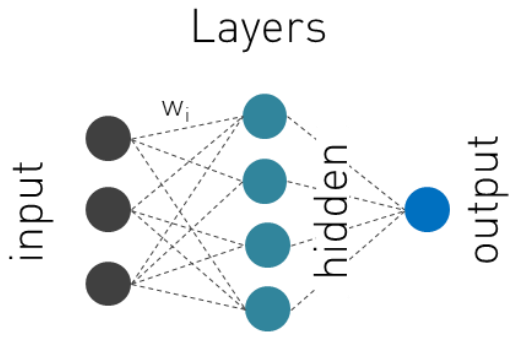

Figure 6: Abstract representation of the neural network.

2.4 Ensemble learning methods

These methods combine different base models, sometimes of different types also, and estimator models to collectively increase the robustness and overall performance. Three such approaches are boosting, stacking and bagging.

In the case of boosting [31], the base models are of the same type and are characterised by increased bias. They are fitted sequentially using a deterministic aggregation strategy (here, XGBoost [10], AdaBoost [17] and GradientBoosting [7]). The main goal of each step is to fix the errors observed in the previous step and finally lower the bias.

In the case of stacking [36], the base models (having one or more levels) are not of the same type and they are combined simultaneously, utilizing an estimator model such as the logistic or linear regression.

In bagging, a set of models of the same type are trained, with each one of them using random samples of the training data with replacement and randomly selecting to exclude some features. Finally their results are aggregated to form the output, leading to a decrease in variance (here, random forest [9] with the use of decision trees).

2.5 Classic time-series forecasting

2.5.1 Prophet [34]

It is a modular regression model that accepts time-series for forecasting, using either a linear or non-linear approach according to the growth type specified while being able to incorporate seasonal effects, handle missing values and detect outliers.

2.6 Hyperparameter tuning

2.6.1 Genetic algorithm [32]

This is a heuristic cross-validated method incorporating ideas of the evolution theory. More specifically, a simple approach is to randomly generate an initial population of models and determine their “fitness” with a specified metric. The top-performing models along with a random sub-population, are selected to produce the new members of the next generation of models, passing to them a randomly selected subset of their hyperparameter values. Additionally, there is a chance of mutation and when it occurs some hyperparameters are randomly altered, in order to have more diverse models. Finally, after a number of generations, no offspring models are significantly better than the ones of the previous generation, at which point the algorithm is considered to have converged.

2.6.2 Grid search [28]

It is an exhaustive scheme of cross-validated search for the set of hyperparameter values that maximize or minimize a score function. Essentially, all the possible combinations of the specified hyperparameter values are tested and the mean value of the cost function across the folds is calculated.

2.7 Evaluation metrics

2.7.1 Accuracy

The accuracy metric, representing the ratio of correctly predicted observations, is used to measure the performance of the classification models. Though, the confusion matrices provide additional insight regarding the class-wise accuracy which is especially important with imbalanced datasets.

$$Accuracy = \frac{No.correctly\ classified\ late\ +\ No.correctly\ classified\ non-late}{No.records}$$

2.7.2 RMSE

The root-mean-square error, the square root of the average squared error between the real and estimated values, is used to measure the performance of the regression models. So, zero RMSE would indicate perfect fit.

$$RMSE = \sqrt{\frac{\sum_i (predicted\ no.days_i - true\ no.days_i)^2}{No.samples}}$$

3. METHODS AND RESULTS

The steps of the workflow are presented in figure 7. Figure 8 provides an example of the function calls used to perform a test.

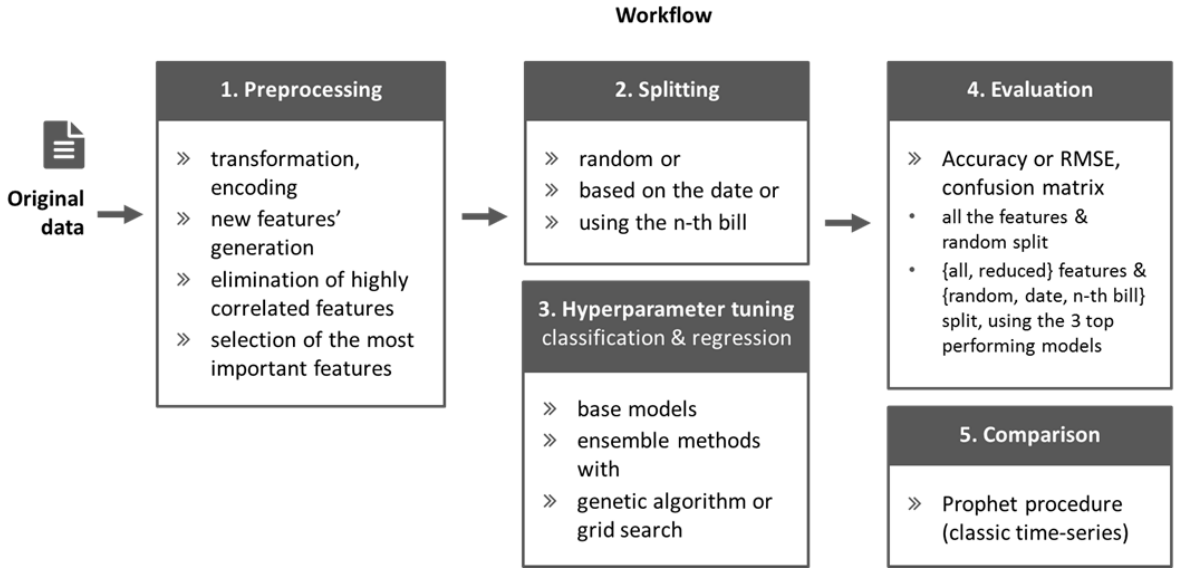

Figure 7: The workflow

```
# Read the IBM late payments data
ibm_data_file = "/content/drive/MyDrive/ibm_late_payments.data"
ibm_bills = read_bills(ibm_data_file)


# Date split
date_thr = '2013-05-01'
y_type = 'class'
# Get the training and test data (pandas dataframes) along with the
# corresponding dropped features (not known at the time of issuing the invoices)
# and their class (0: non-late, 1: late)
ibm_train, ibm_train_dr, y_train, \
  ibm_test, ibm_test_dr, y_test = date_split(ibm_bills, date_thr)

# Add features using clustering / classification algorithms
num_groups = 2
ibm_train, ibm_test = add_classes(ibm_train, y_train, ibm_test, num_groups)

# Remove highly correlated / redudant features
ibm_train, ibm_train_dr, ibm_test, ibm_test_dr = \
  remove_hcorr(ibm_train, ibm_train_dr, ibm_test, ibm_test_dr, corr_thr=0.85)

# Select the top-k features
 ibm_train, ibm_train_dr, ibm_test, ibm_test_dr = \
  sel_features_class(ibm_train, ibm_train_dr, ibm_test, ibm_test_dr, y_train, top_k=15)

```

```
27
28 # Single model evolution
29 method_use = 'rf' # Random forest
30 # Parameters to be tested
31 param_grid = {
32               "pipe__n_estimators": Integer(10, 1500),
33               "pipe__max_features": Categorical(["sqrt", "log2"]),
34               "pipe__criterion": Categorical(["gini", "entropy"]),
35               "pipe__bootstrap": Integer(0, 1),
36               "pipe__max_depth": Integer(1, 20),
37               "pipe__min_samples_split": Integer(2, 20),
38               "pipe__min_samples_leaf": Integer(1, 10),
39               "pipe__min_impurity_decrease": Continuous(0.0, 0.5)
40               }
41
42 # Evolved model (best)
43 evol_estim = baseModel_test(ibm_train, y_train, ibm_test, y_test,
44                             method_use, param_grid)
45
```

Figure 8: Example of the functions used to train and tune the hyperparameters of a base model (random forest) using the 15 most important features and random split.

3.1 Dataset

The dataset selected to train the models and perform the tests was provided by IBM and sourced from Kaggle Datasets [14]. It consists of 10 features (with 2 extra features that are derived from the original) corresponding to 100 unique customers, amounting to 2,466 entries spanning a period of roughly two years. Even though it is a relatively small dataset, it was preferred over bigger datasets of telecommunications and streaming companies, because the customers' behavior captured (features) more closely resemble the desired pattern of long-term business relationships with anonymous and context-free information, monthly invoices, fixed number of days allowed to settle them and varying amounts, regarding mainly one service / product that is not pre-paid allowing the bill to be disputed.

3.1.1 Features

Original features:

- countryCode: three-digit encoding of the country (5 unique countries)
- customerID: ten-character alphanumeric identifier of the customer (100 unique customers)
- PaperlessDate: date when the customer opted for electronic invoice
- invoiceNumber: a variable-length numeric identifier of the invoice
- invoiceDate: the date the invoice was issued (January 3 2012 to December 2 2013, ~1.2 invoices per customer per month)
- DueDate: the date the invoice is due (30 days after the issuing date)
- invoiceAmount: the amount of the current invoice only
- Disputed: a one-digit (true / false) indicator of whether the invoice was disputed
- SettledDate: the date the invoice was settled in full

- almost 1/3 of the invoices are settled after the due date
- PaperlessBill: the type of the current invoice (paper / electronic)

Extra features:

- DaysToSettle: the number of days between issuing and settling the invoice
- DaysLate: the number of days between the due date and the date the invoice was settled

3.1.2 Data exploration

Below, a number of plots (figures 9-47) are used to visually explore / summarize the initial data:

- Approximately, 33% of the invoices are late.
- Most of the invoices (64%) are settled with no delay (0-30 days) and the vast majority (80%) of the late ones are settled within 15 additional days; the range is 0 to 45 days and the mean value 3.4 days.
- The mean amount is 59.9 and it ranges between 5.26 and 128.28. Its distribution is approximately normal.
- The ratio of paper and electronic bills is nearly the same.
- It is much more likely for disputed bills to be late; 68% of the disputed bills are late in contrast with only 26% of the non-disputed bills.
- Electronic invoices tend to be settled later compared to the paper ones with the mean values being 24.6 and 30.2, respectively.
- Electronic invoices tend to be late by more days compared to the paper ones with the mean values being 2.4 and 4.4, respectively.
- The number of invoices per month does not differ.
- The distribution of the number of days to settle the bill is approximately normal.
- No correlation is observed between the invoice amount and the delay or the number of days to settle it.
- The number of unique customers per month is approximately the same.
- Three of the countries (391, 406, 770) have more invoices than the other two (818, 897), which aligns with the observation that these countries also have more customers.
- Two of the countries (406, 818) have a greater ratio of disputed to non-disputed bills.
- The ratio of late bills is comparatively lower in one country (391).
- Two of the countries (818, 897) have a lower ratio of paper to electronic bills.

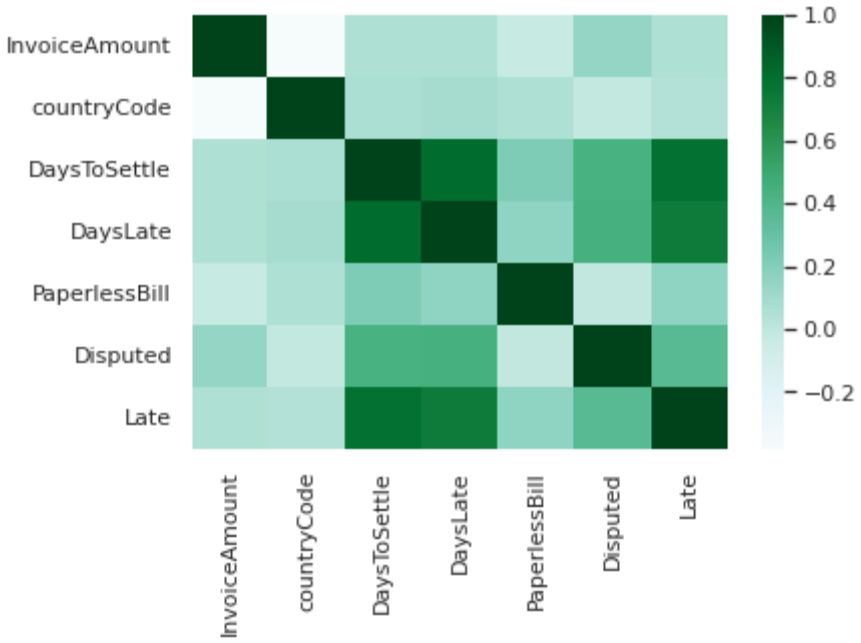

Figure 9: Heatmap of the correlation between the features. As it was expected, the DaysToSettle, DaysLate and Late features are highly correlated (positively). A less strong correlation is observed with the `disputed` feature.

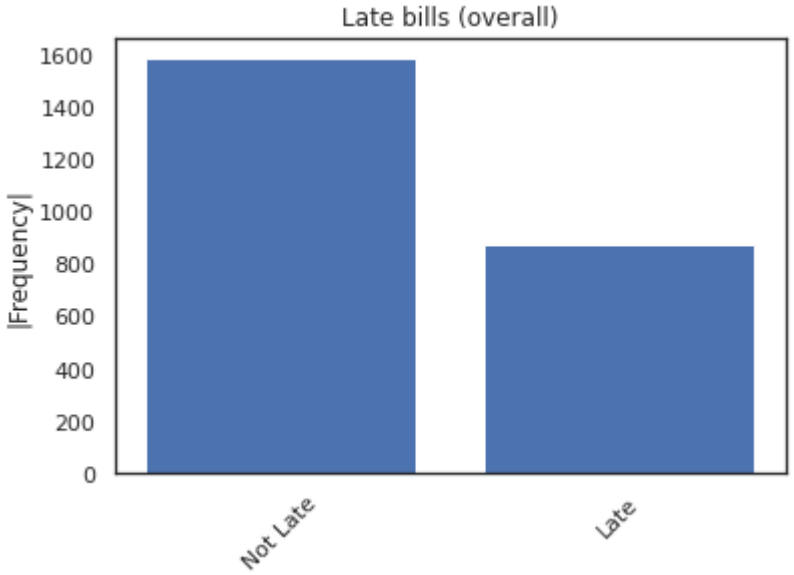

Figure 10: Barplot of the number of late and non-late bills (absolute frequency). Approximately, 33% of the invoices are late.

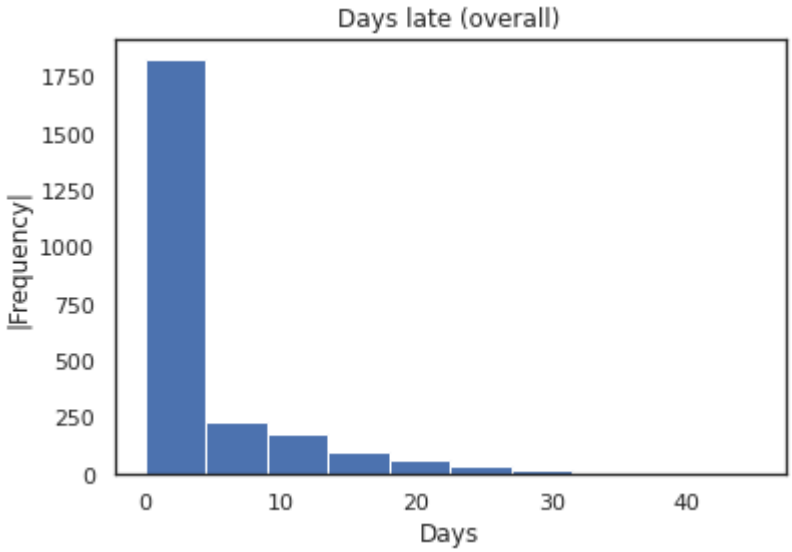

Figure 11: Histogram of the number of days of delay (date of settling - due date). Most of the invoices (64%) are settled with no delay and the vast majority (80%) of the late ones are settled within 15 additional days; the range is 0 to 45 days and the mean value 3.4 days.

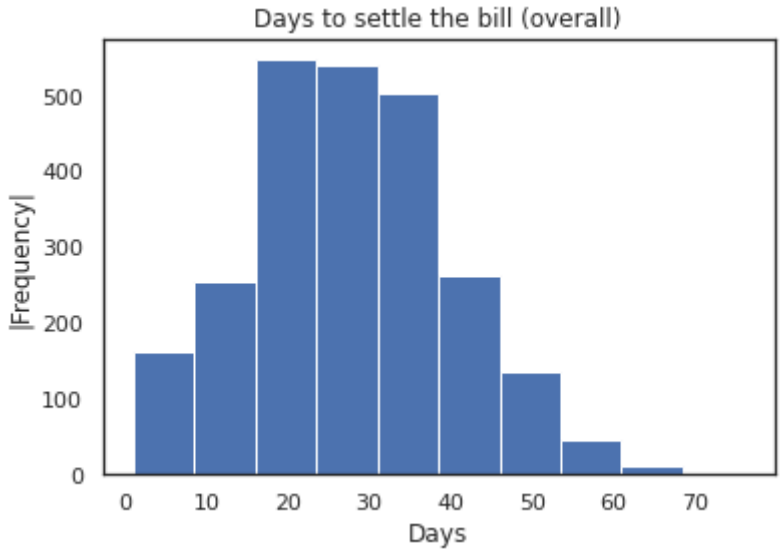

Figure 12: Histogram of the number of days until the invoice is settled (date of settling - issuing date). Most of the invoices (64%) are settled within 30 days and 80% are settled within 40 days; the range is 0 to 75 days and the mean value 26.4 days. The distribution is approximately normal.

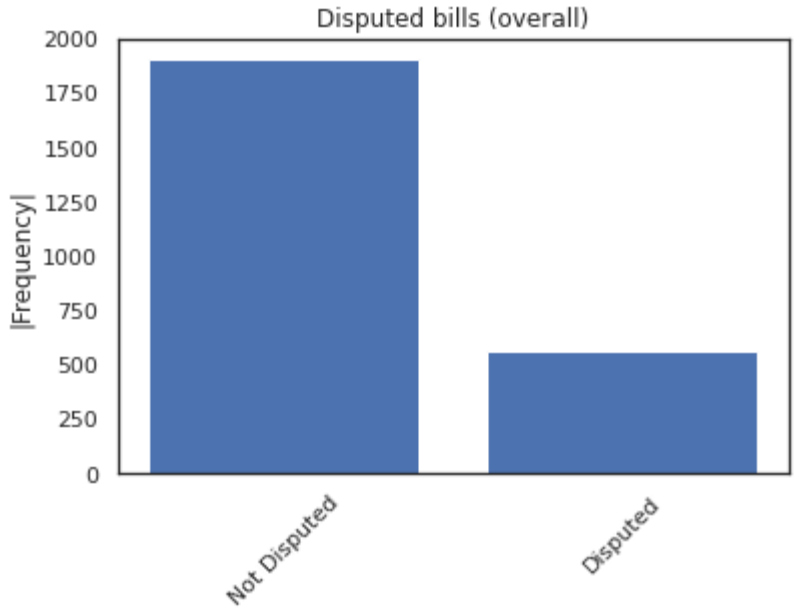

Figure 13: Barplot of the number of disputed and non-disputed bills (absolute frequency). Approximately, 23% of the invoices are disputed.

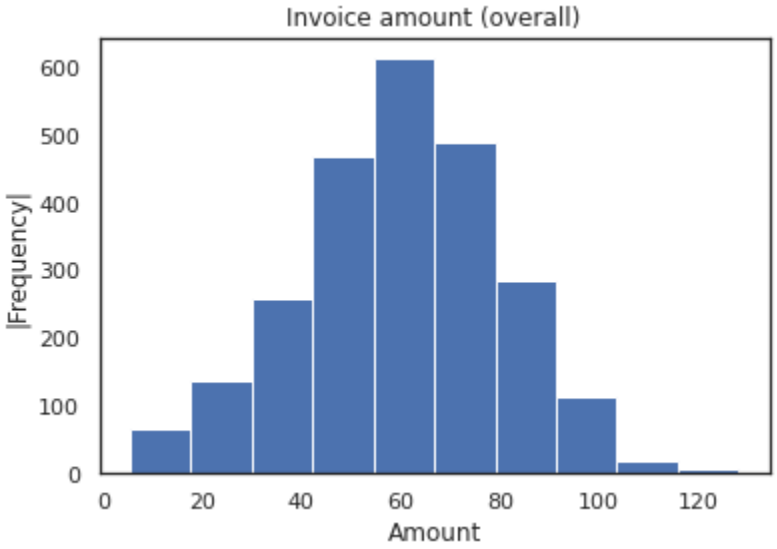

Figure 14: Histogram of the new amount (not including any previous outstanding / unpaid amount) per invoice. The mean value is 59.9 and the amount ranges between 5.26 and 128.28. The distribution is approximately normal.

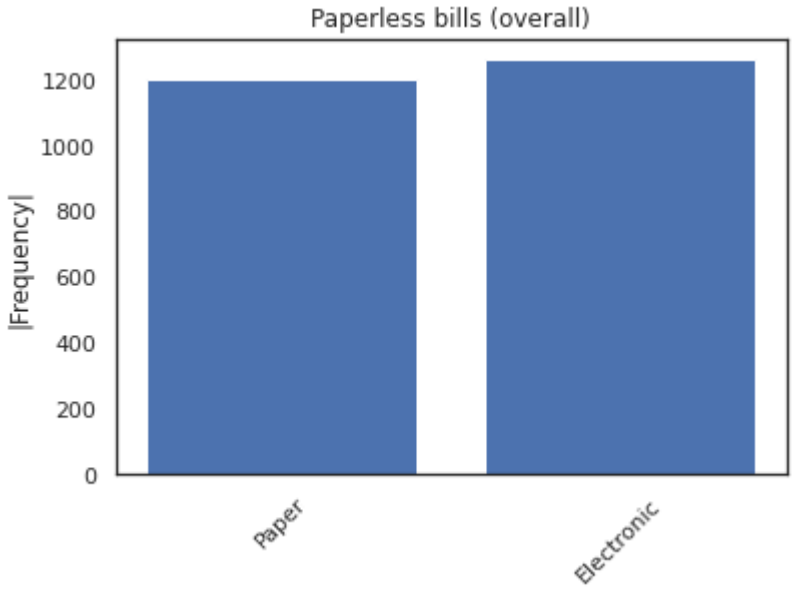

Figure 15: Barplot of the type of invoice (paper or electronic; absolute frequency). Approximately, 49% of the invoices are paper and 51% electronic.

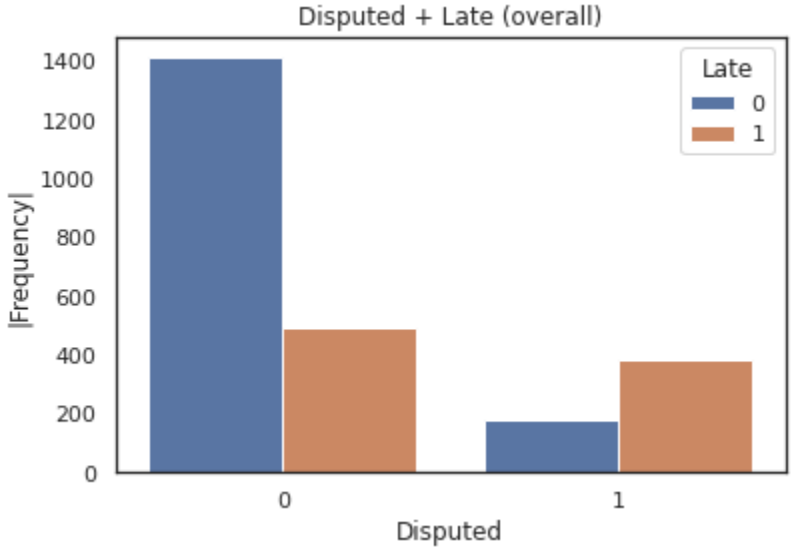

Figure 16: Barplot of the number of disputed and non-disputed bills that are late or non-late (absolute frequency). Clearly, it is much more likely for disputed bills to be late; 68% of the disputed bills are late in contrast with only 26% of the non-disputed bills.

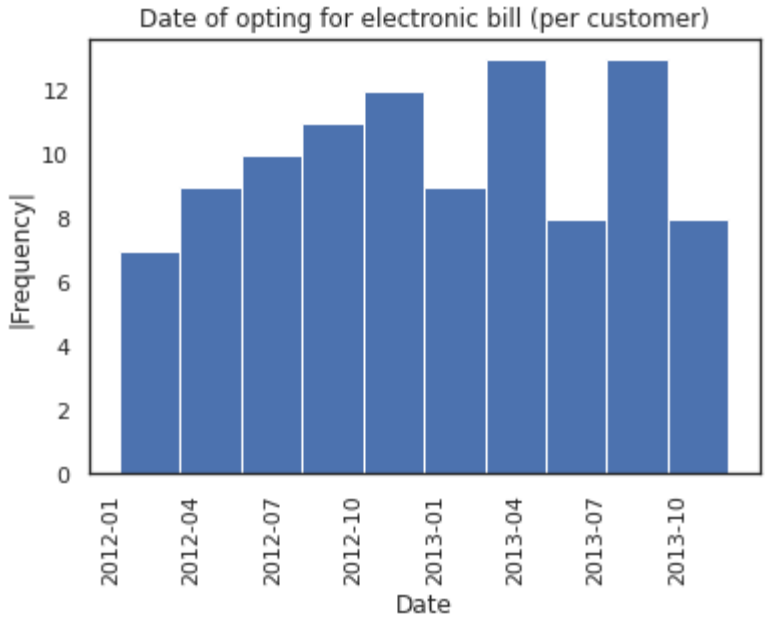

Figure 17: Histogram of the date (year-month) that the customers opted for electronic invoice. All the dates are within the range of the invoices' issuing dates.

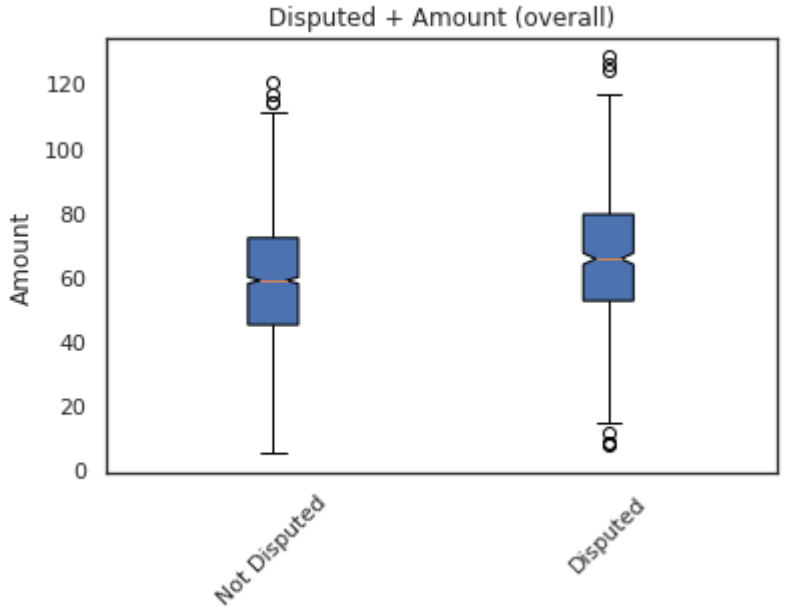

Figure 18: Boxplots of the amount according to whether the invoice was disputed or not. The mean amount of the disputed bills is slightly greater.

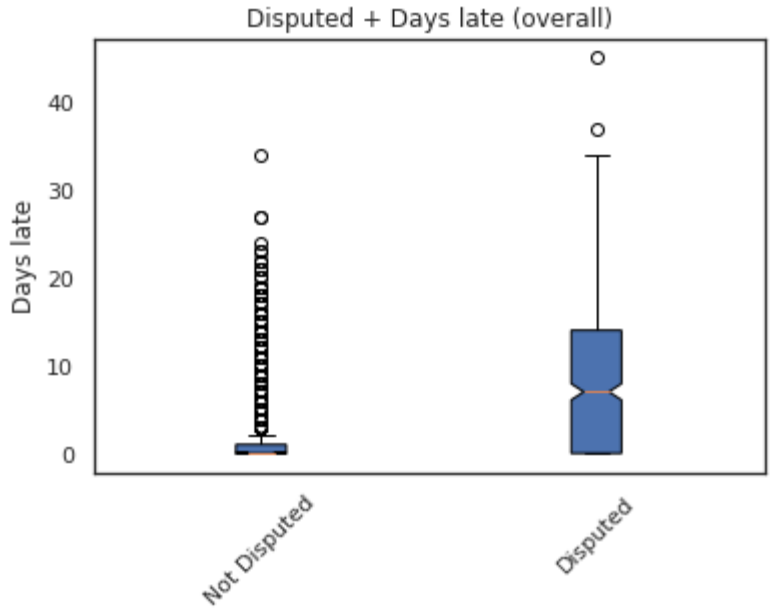

Figure 19: Boxplots of the number of days the bills were late according to whether the invoice was disputed or not. The disputed bills have a greater mean value as well as a greater range.

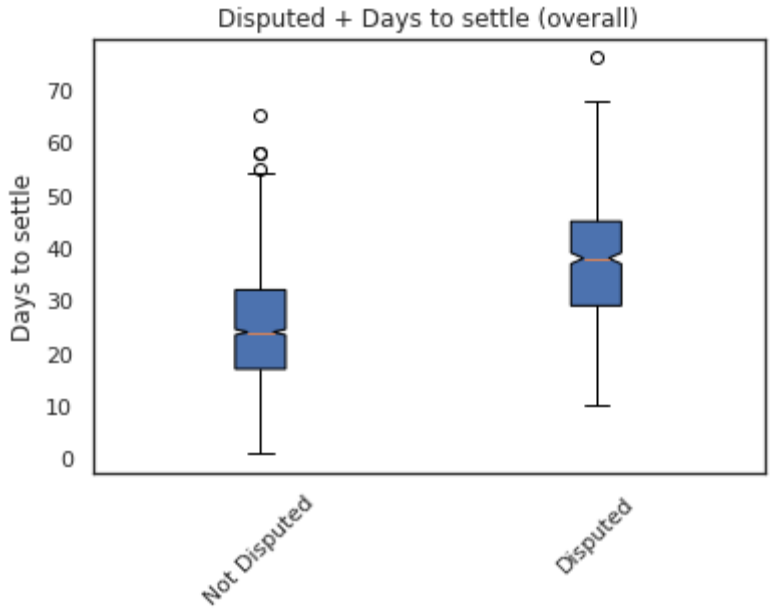

Figure 20: Boxplots of the number of days until the bills were settled according to whether the invoice was disputed or not. The disputed bills have a greater mean value.

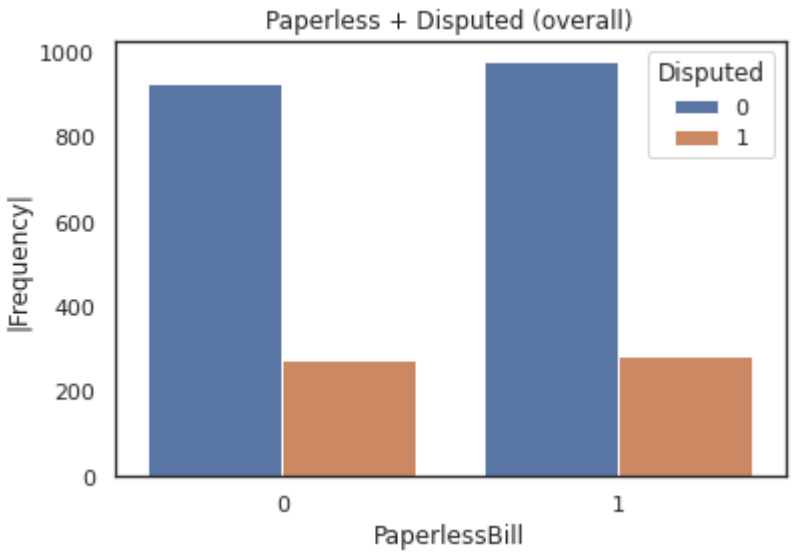

Figure 21: Barplot of the number of non-disputed and disputed electronic bills (absolute frequency).

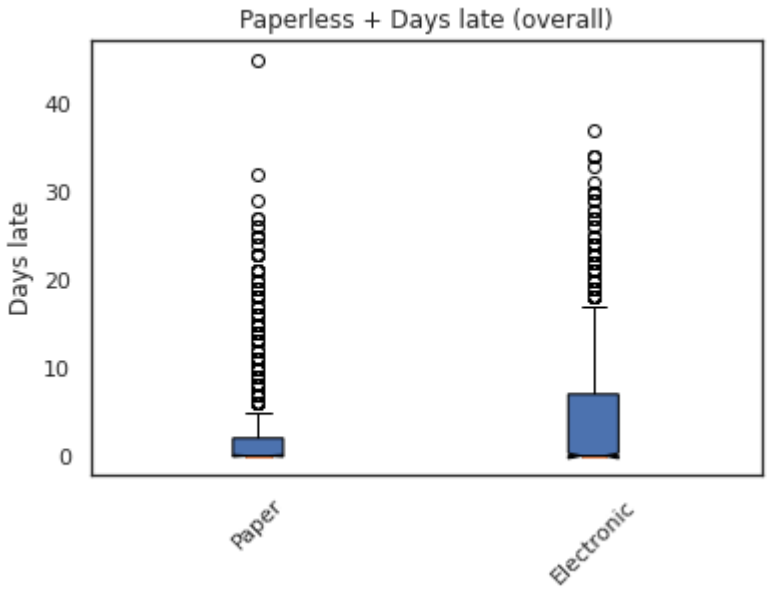

Figure 22: Boxplots of the number of days the bills were late according to the type of invoice (paper, electronic). Electronic invoices tend to be late by more days compared to the paper ones with the mean values being 2.4 and 4.4, respectively.

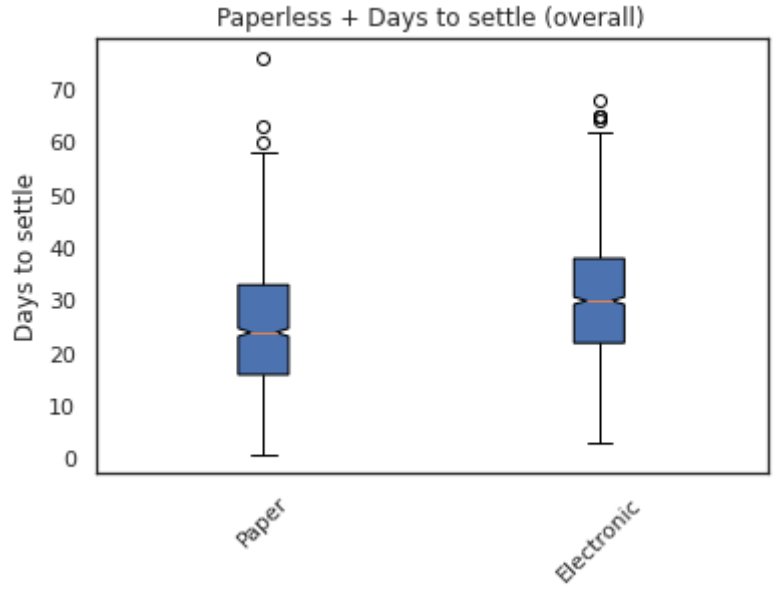

Figure 23: Boxplots of the number of days to settle the bill according to the type of invoice (paper, electronic). Electronic invoices tend to be settled later compared to the paper ones with the mean values being 24.6 and 30.2, respectively.

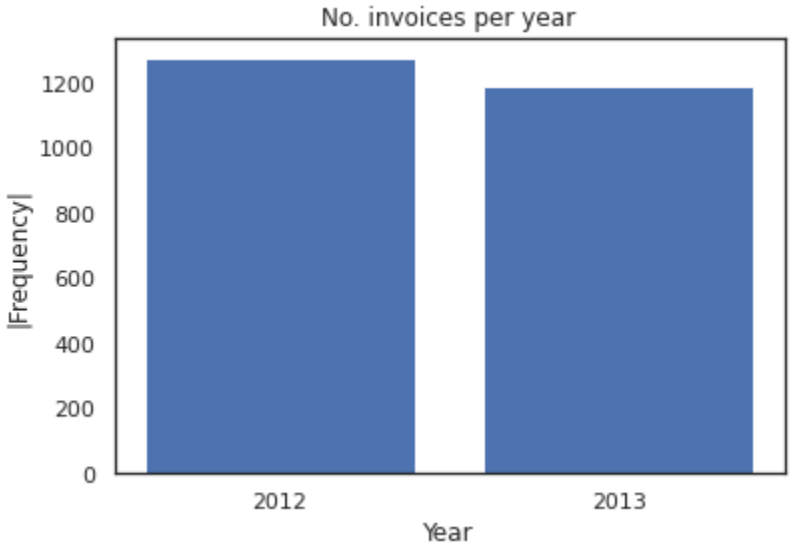

Figure 24: Barplot of the number of invoices per year.

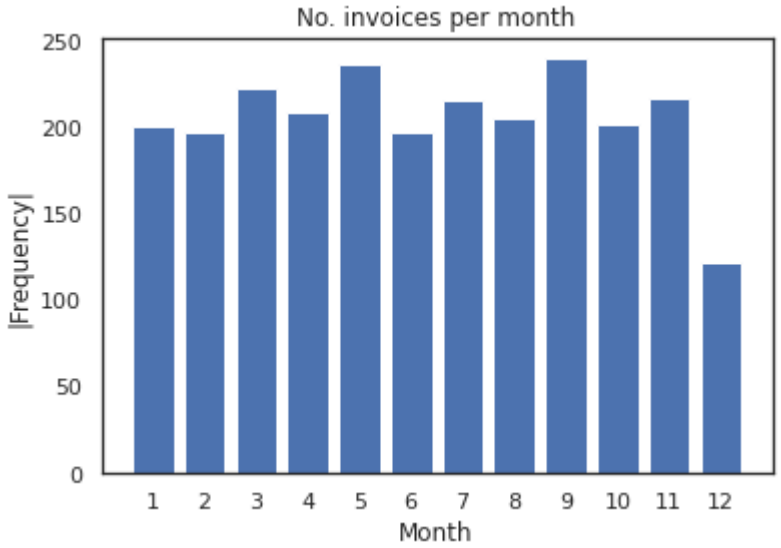

Figure 25: Barplot of the number of invoices per month, January to December (absolute frequency). They seem to be evenly distributed; the invoices of December mainly correspond to the year 2012 since the last invoice of 2013 was issued on December 3.

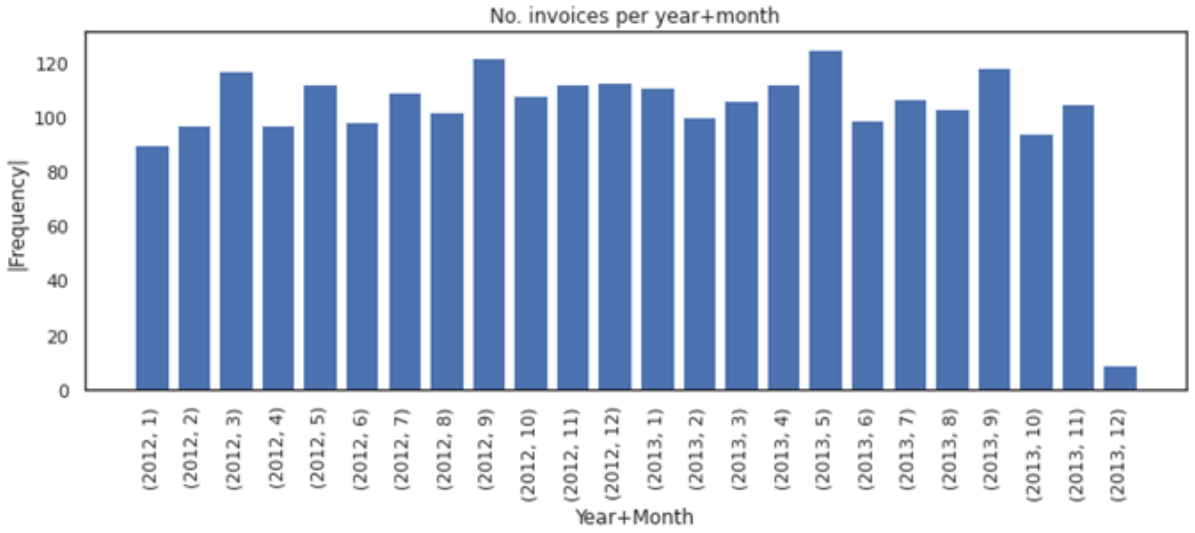

Figure 26: Barplot of the number of invoices per month per year (absolute frequency). They seem to be evenly distributed; the last invoice of 2013 was issued on December 3.

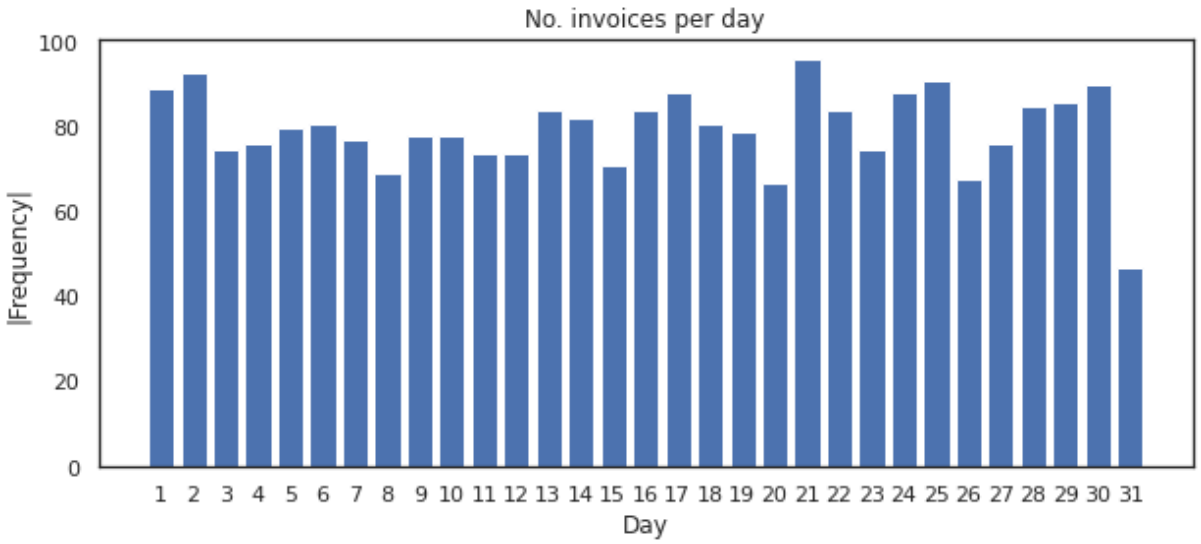

Figure 27: Barplot of the number of invoices per day of the month (absolute frequency). They seem to be evenly distributed; the 31st day has approximately half the frequency of the other days, as expected.

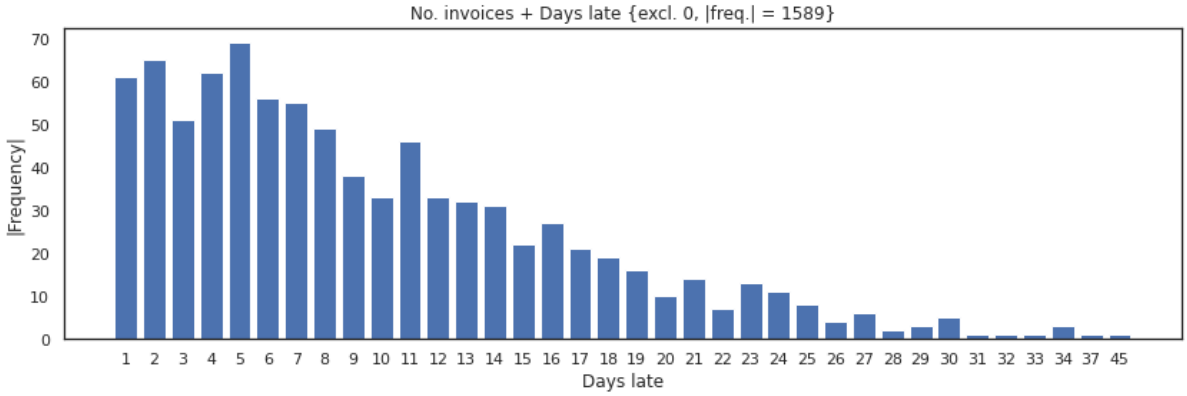

Figure 28: Barplot of the number of days the bill was late (absolute frequency), excluding the non-late bills.

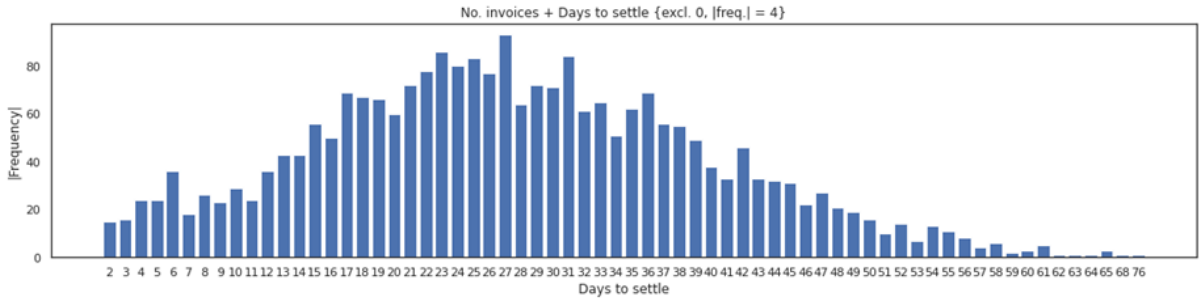

Figure 29: Barplot of the number of days to settle the bill (absolute frequency). The distribution is approximately normal.

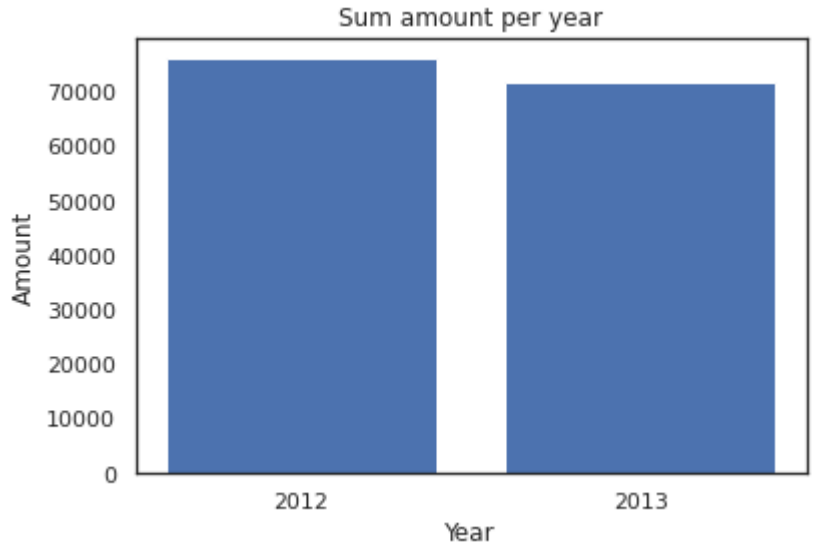

Figure 30: Barplot of the sum of the invoice amounts per year. The mean sum amount per month per year is approximately the same; 6,339 and 6,512, respectively.

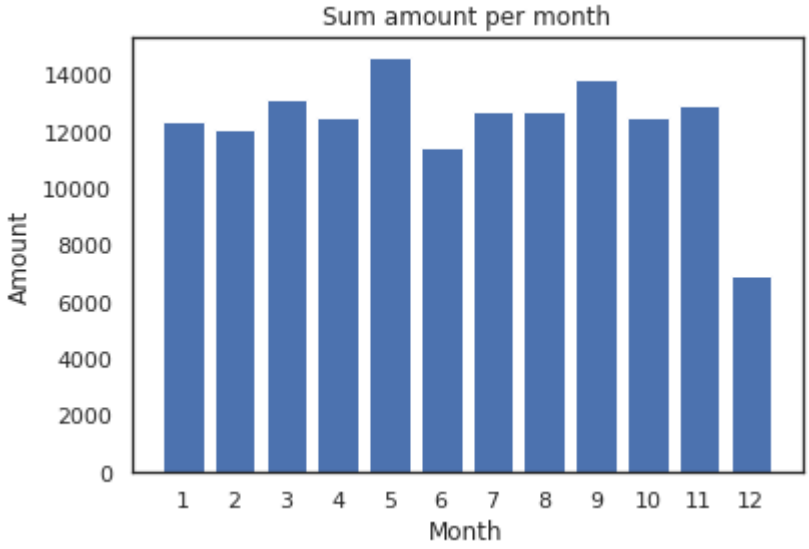

Figure 31: Barplot of the sum of the invoice amounts per month. The values are approximately the same; the last invoice of 2013 was issued on December 3 and thus the sum is nearly half, as expected.

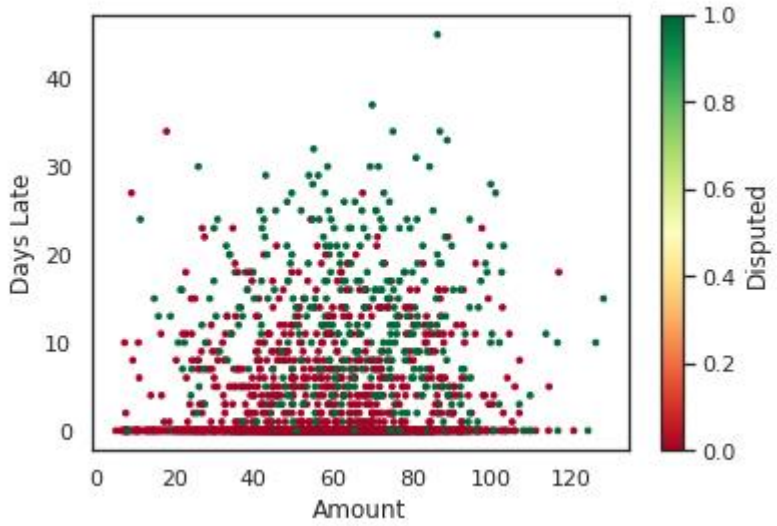

Figure 32: Scatter plot of the invoice amounts to the number of days the bills were late, coloring the points according to whether they were disputed or not. No correlation is observed.

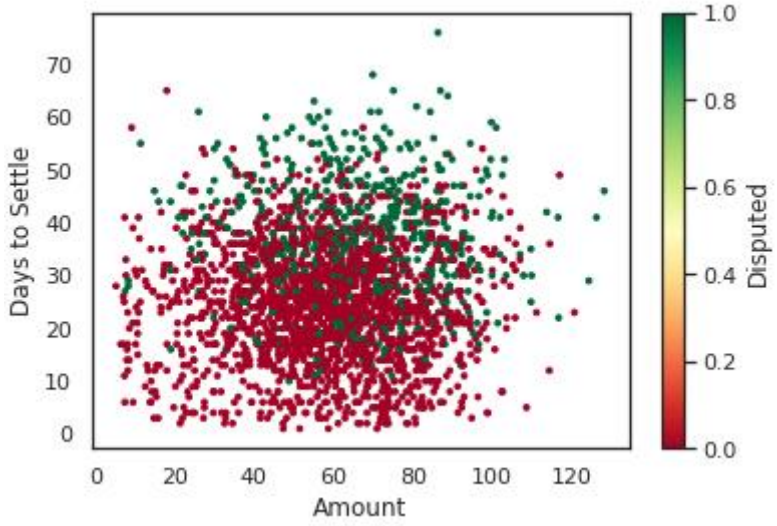

Figure 33: Scatter plot of the invoice amounts to the number of days to settle the bills, coloring the points according to whether they were disputed or not. No correlation is observed.

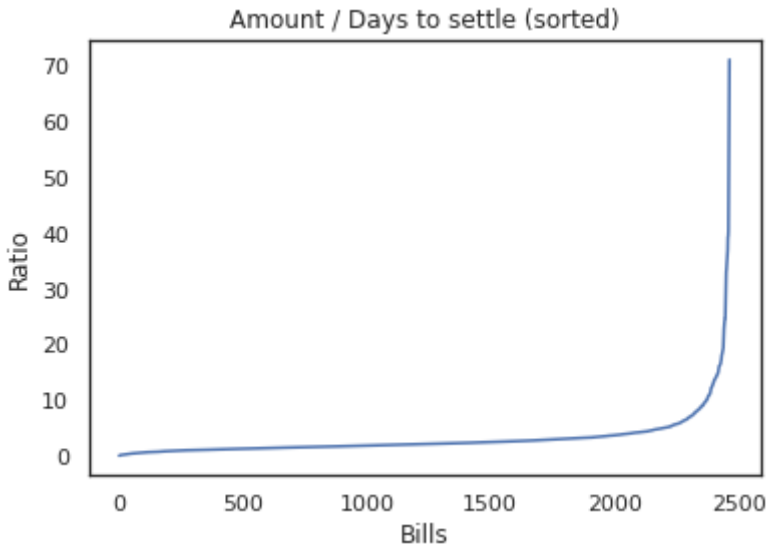

Figure 34: Plot of the amount to the number of days the bill was late ratio. Most of the bills have a low ratio (< 10) while it ranges from 0.16 to 71.02. Greater values correspond to bills being paid immediately or within a few days.

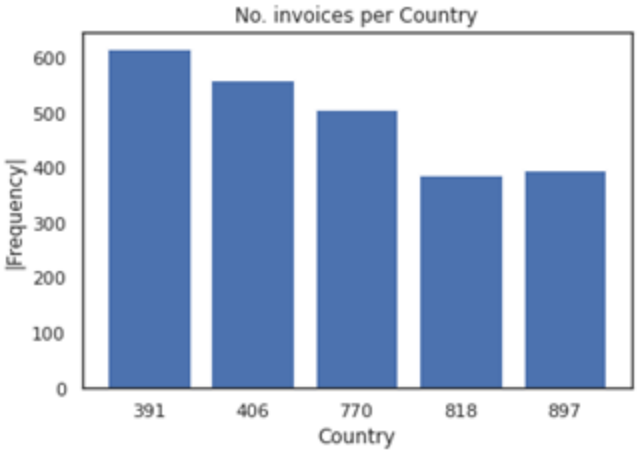

Figure 35: Barplot of the number of invoices per country (absolute frequency). The first three countries (391, 406, 770) have more invoices than the last two (818, 897).

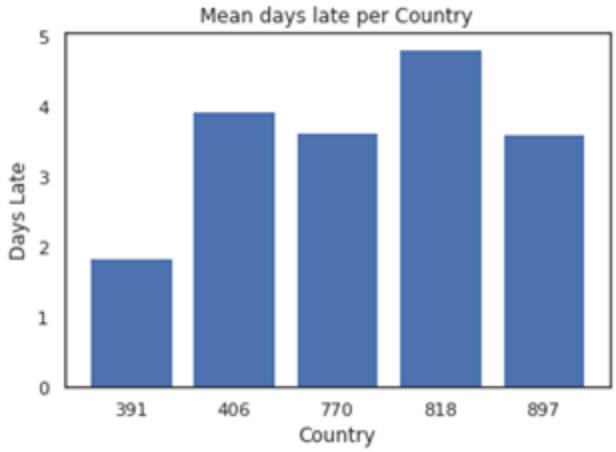

Figure 36: Barplot of the mean number of days the bills were late per country. The mean value of the first country (391) is approximately 50% lower compared to the other countries.

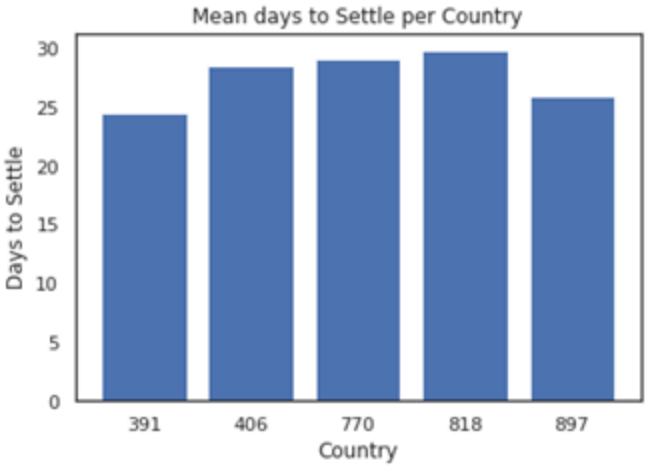

Figure 37: Barplot of the mean number of days until the bills are settled per country. The mean value of the first country (391) is slightly lower than that of the other countries the difference is not as marked compared to the mean number of days the bills were late.

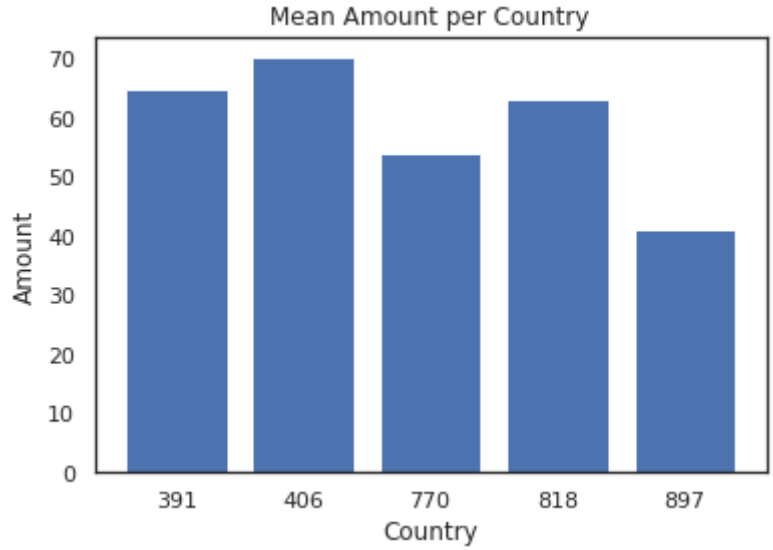

Figure 38: Barplot of the mean amount per country. The mean value of the fifth country (897) is much lower (23% - 43%) than that of the other countries.

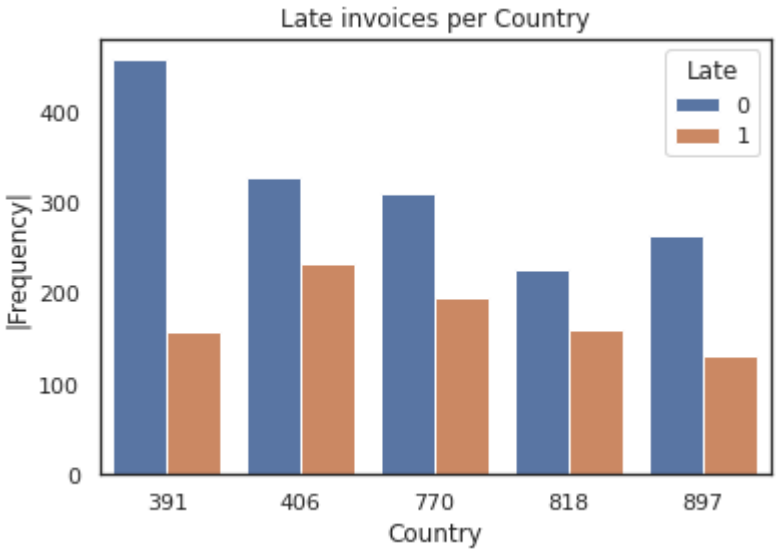

Figure 39: Barplot of the number of late and non-late invoices per country. Clearly, the ratio of late bills is lower in the first country (391).

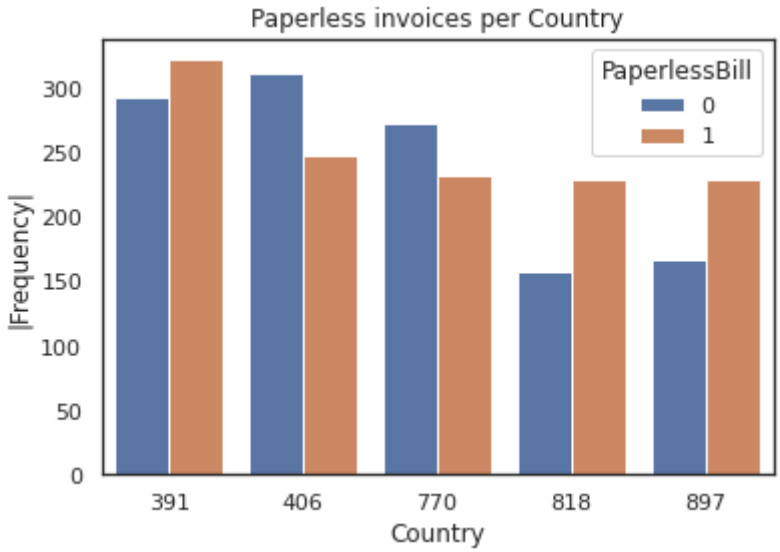

Figure 40: Barplot of the number of paper and electronic invoices per country. The last two countries (818, 897) have a lower ratio of paper to electronic bills.

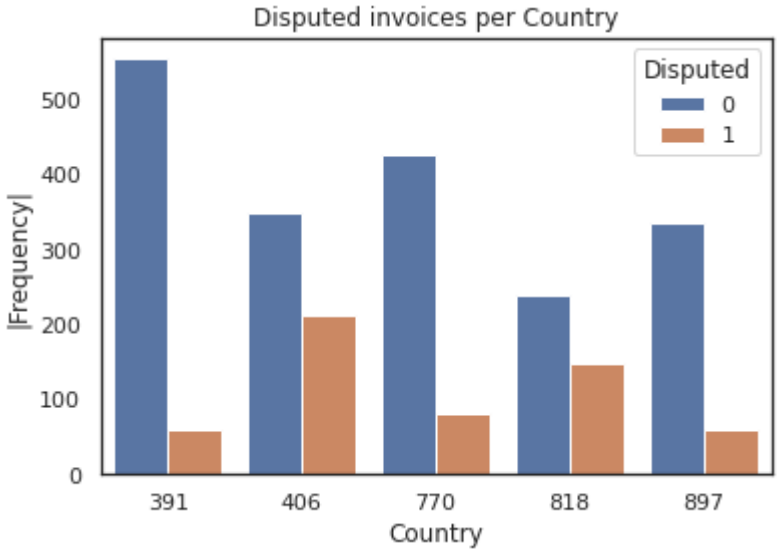

Figure 41: Barplot of the number of disputed and non-disputed invoices per country. The second and fourth countries (406, 818) have a greater ratio of disputed to non-disputed bills.

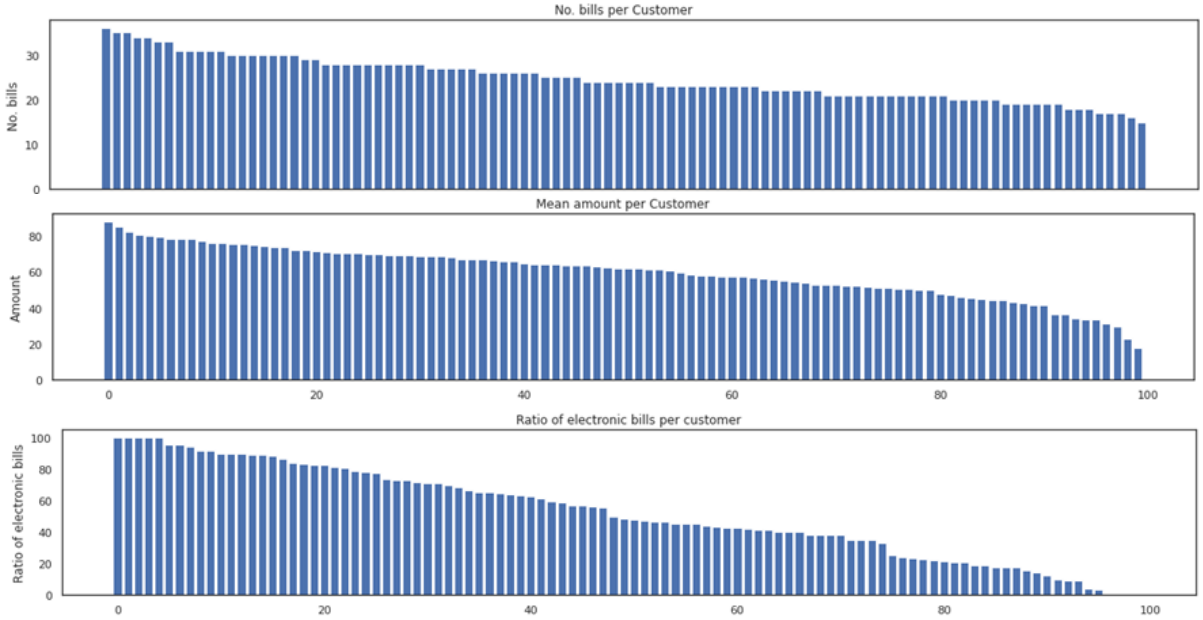

Figure 42: Barplots of the number of bills (1st), the mean amount (2nd) and the percentage of electronic bills per customer (3rd).

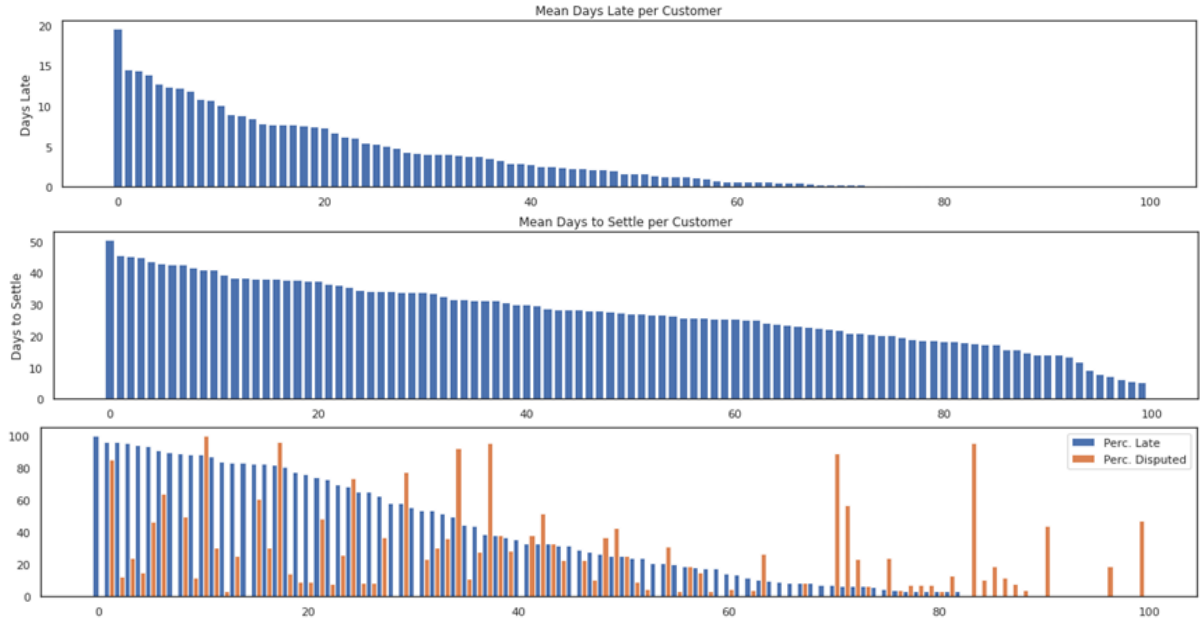

Figure 43: Barplots of the mean number of days the bill was late (1st) and the mean number of days to settle the bill (2nd) along with the percentage of late and disputed bills per customer (3rd).

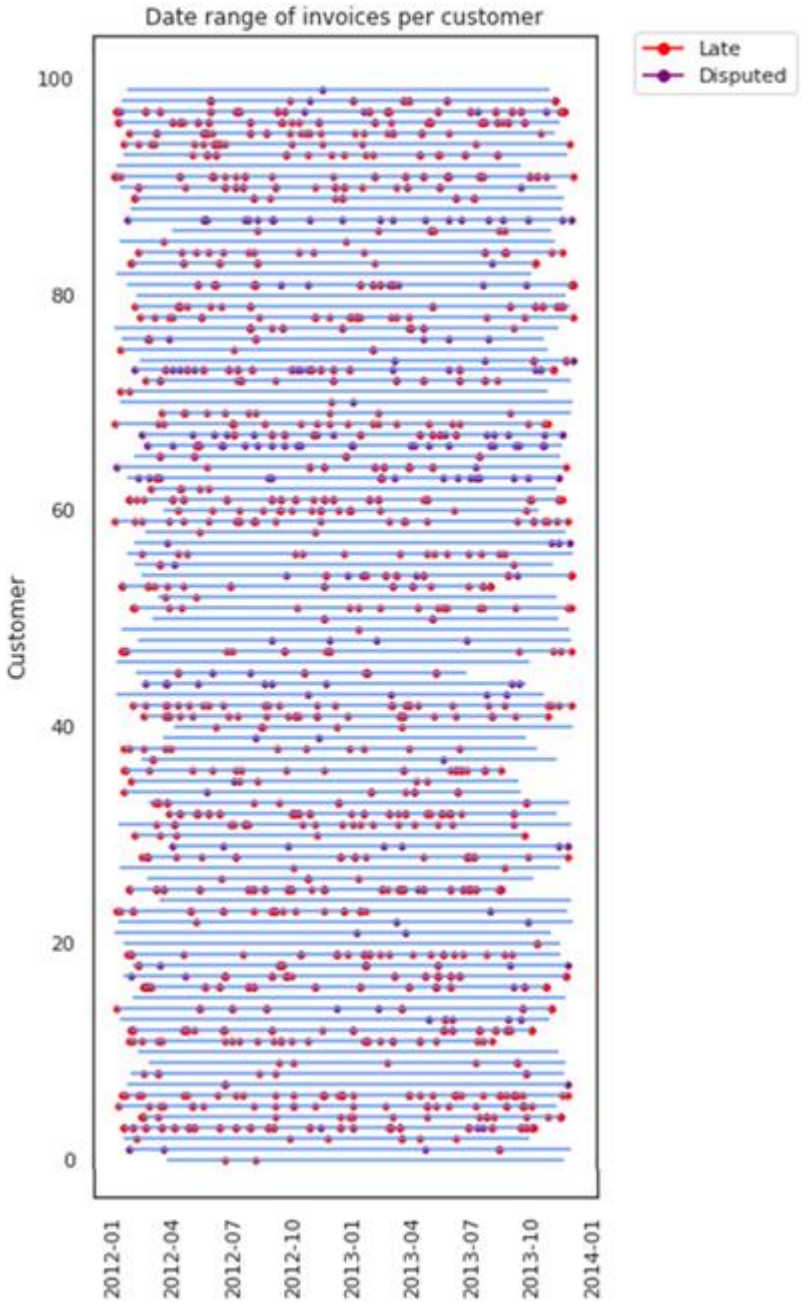

Figure 44: Plot depicting the date range of the invoices per customer along with their status (late, disputed).

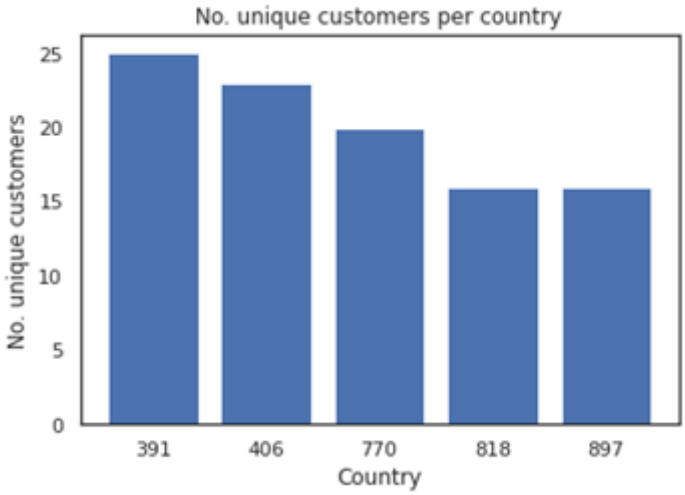

Figure 45: Barplot of the number of unique customers per country (absolute frequency). The first three countries (391, 406, 770) have more customers than the last two (818, 897). This aligns with the number of invoices per country.

Figure 46: Barplot of the number of unique customers per month. The values are approximately the same.

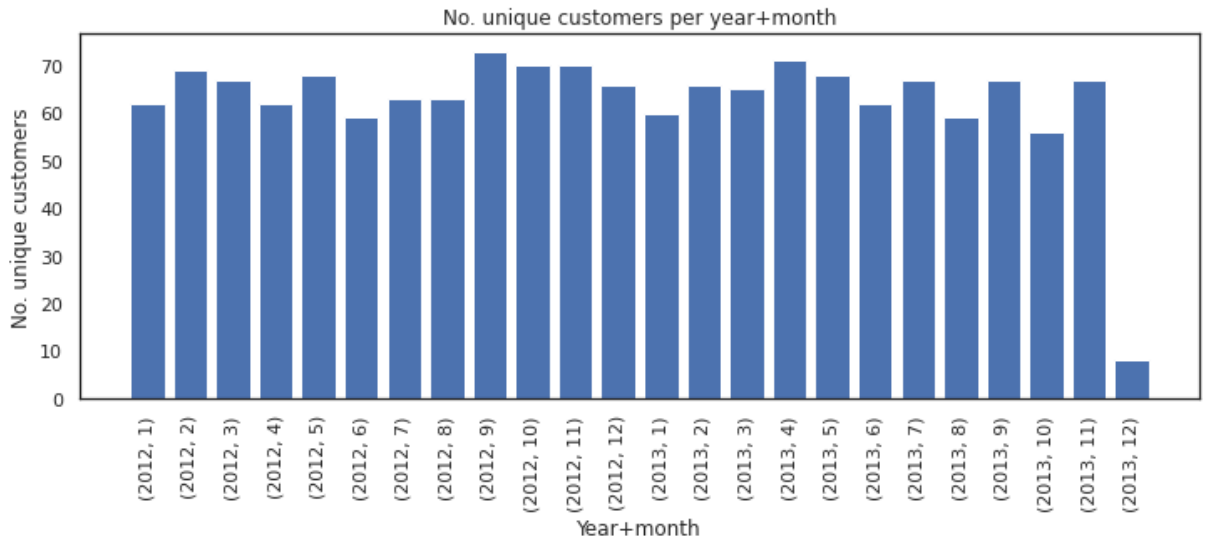

Figure 47: Barplot of the number of unique customers per month per year (absolute frequency). They seem to be evenly distributed; the last invoice of 2013 was issued on December 3.

3.2 Preprocessing

The preprocessing consists of three steps: transformation / encoding, generation of new features and feature selection. There were no missing values in the initial data. Furthermore, three methods of splitting the data into training, validation and test sets were utilized, each aiming to evaluate a different aspect of the models.

3.2.1 Transformation / encoding

The alphanumeric features (InvoiceDate => month, day, year, customerID, PaperlessDate, InvoiceDate, DueDate, SettledDate, Disputed, PaperlessBill), were converted to numeric and then the categorical features were encoded using either one-hot (countryCode) or binary (Disputed, PaperlessBill) transformation. The features concerning the invoice that are not known at the time of issuing were omitted (SettledDate, Late, DaysLate, DaysToSettle).

3.2.2 Derived features

New features were generated in order to summarise the customer's history up to the date of issuing the current invoice and also capture more recent trends, taking into account the status of the previous two or three bills only.

The features added, are:

- No. {all, late, disputed} bills
- Ratio {late, disputed} bills
- Ratio {late/disputed, disputed/late}
- Sum, mean, std amount {all, late, disputed}

- Mean, std {days late, days to settle}
- Mean, std ratio amount/days to settle
- No. {paper, electronic} bills
- Ratio {paper, electronic} bills
- Ratio {paper/electronic, electronic/paper}
- No. {disputed+paper, disputed+electronic}
- Ratio {disputed /paper. disputed/electronic}
- Ratio {disputed+paper/disputed+electronic, disputed+electronic/disputed+paper}
- No. {late+paper, late+electronic}
- Ratio {late+paper, late+electronic}
- Ratio {late+paper/late+electronic, late+electronic/late+paper}
- No. months since the 1st bill
- No. months since opting for electronic bills
- Bin of InvoiceAmount, InvoiceDate & Month => 4 bins, each
- AmountToSettle = Amount / Days to Settle
- Clusters / Classes => K-means, kNN, GMM

In total, the number of features now is 133.

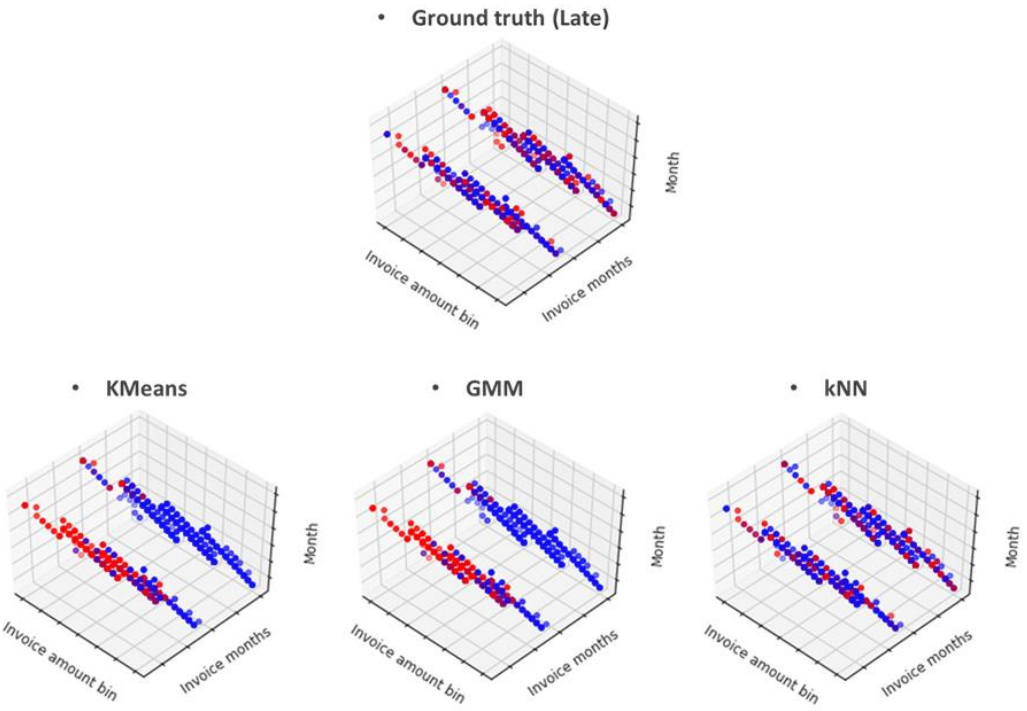

Figure 48: The ground truth (blue: non-late, red: late) in the three-dimensional space of the invoice amount bin x the number of months x the issuing month, and the results of the KMeans, Gaussian Mixture Model and kNN algorithms; kNN seems to perform better.

3.2.3 Dimensionality reduction

PCA was performed after the highly correlated features were removed. The first three principal components explain ~95% of the variance (figure 49) and no feature contributes disproportionately to that variance as it can be seen in figure 50.

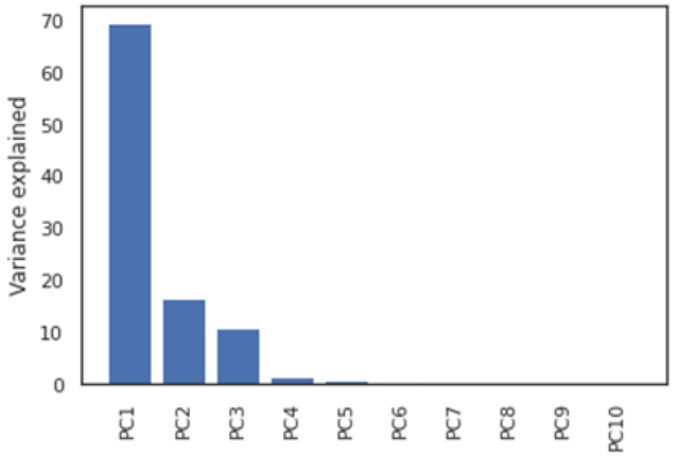

Figure 49: The variance explained per principal component.

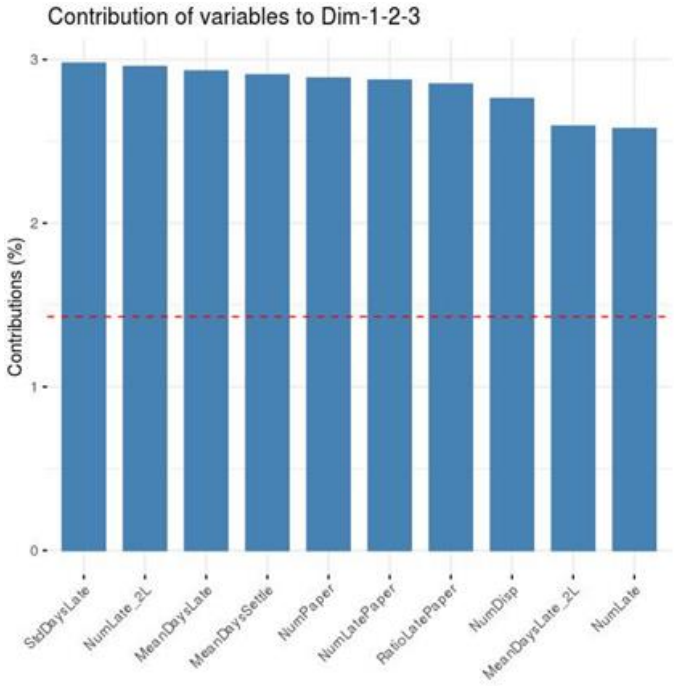

Figure 50: The top contributing features to the variance of the first three principal components. The red line indicated the mean contribution across all the features.

As it can be observed in figure 51, the two classes cannot be separated.

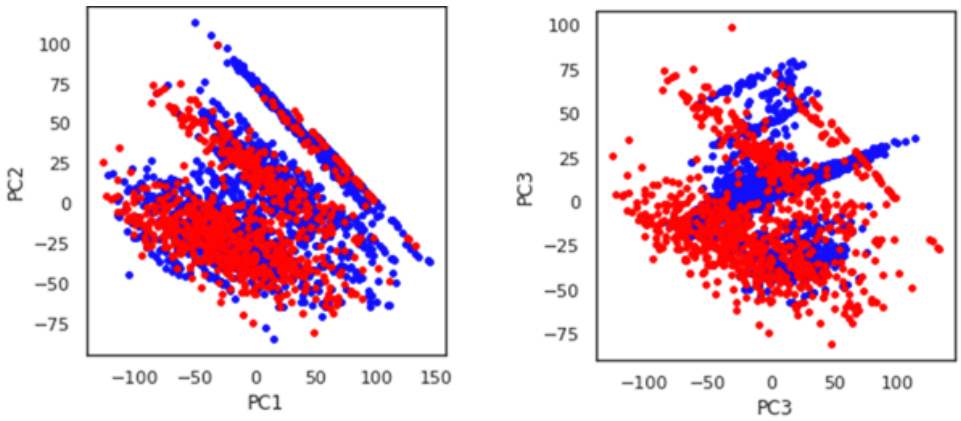

Figure 51: PCA projections; the late (red) and non-late (blue) bills cannot be separated.

Similarly, UMAP was performed with hyperparameters: components = 2, neighbors = 30 and metric = euclidean. Again, the late (red) and non-late (blue) bills cannot be separated.

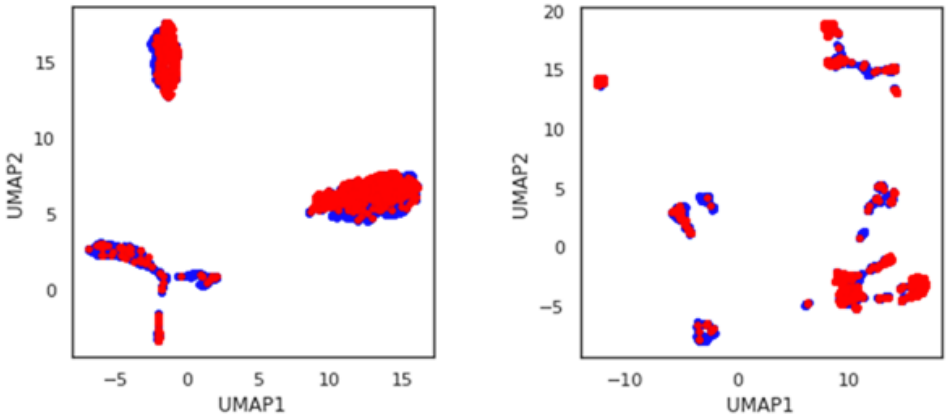

Figure 52: UMAP projections; the late (red) and non-late (blue) bills cannot be separated.

Likewise, the gaussian random projection was also unable to separate the late (red) and non-late (blue) bills although the results are better.

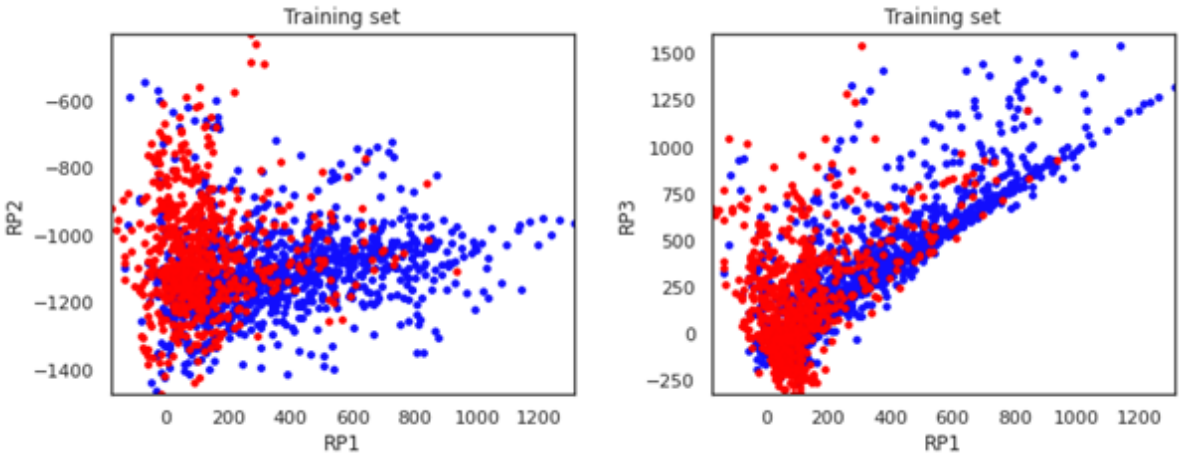

Figure 53: Gaussian random projection; the late (red) and non-late (blue) bills cannot be separated.

3.2.4 Feature selection

In order to reduce the dimensionality, at first highly correlated features (Pearson, > 0.85) are removed and next, only the top-15 features are kept which are those having the lowest mean rank of importance, using 2 methods: mutual information and ANOVA F-score in the case of classification, or 3 methods: mutual information, ANOVA F-score and X2 in the case of regression. An example of the results is provided in figures 54-56.

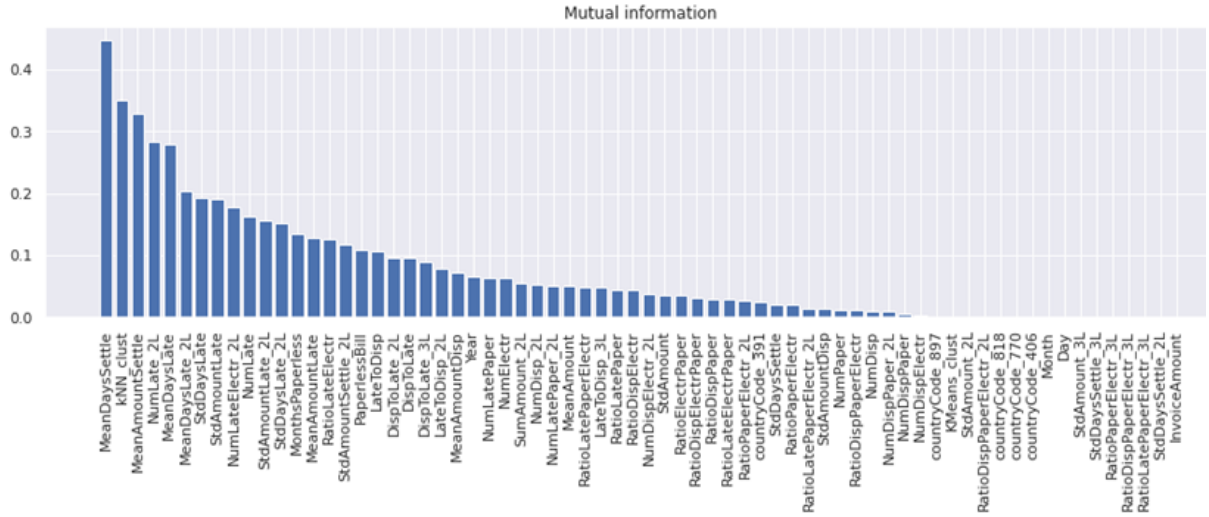

Figure 54: Barplot of the features' importance using the mutual information criterion.

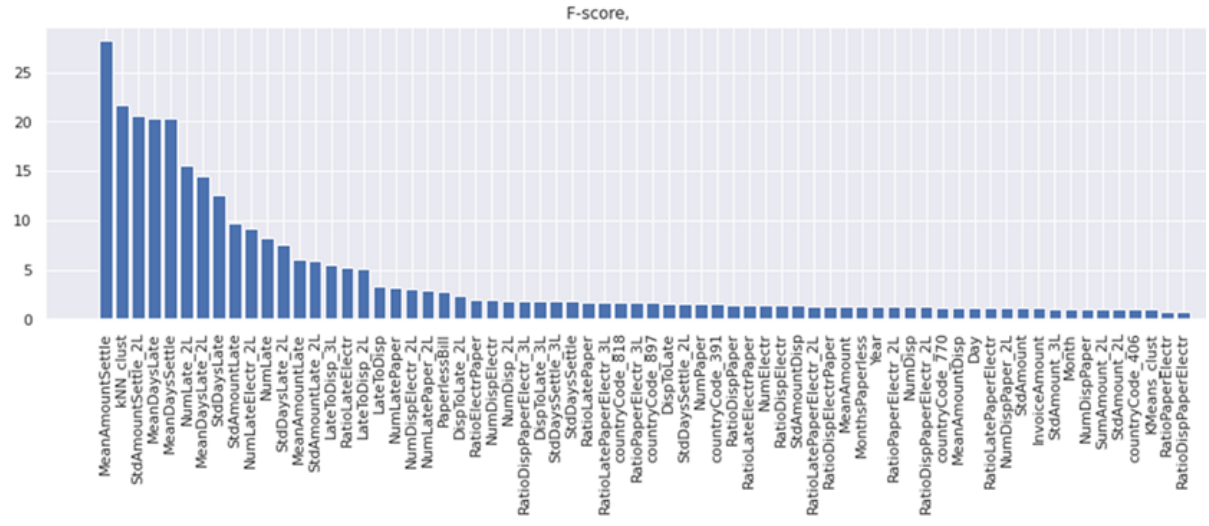

Figure 55: Barplot of the features' importance using the ANOVA F-score criterion.

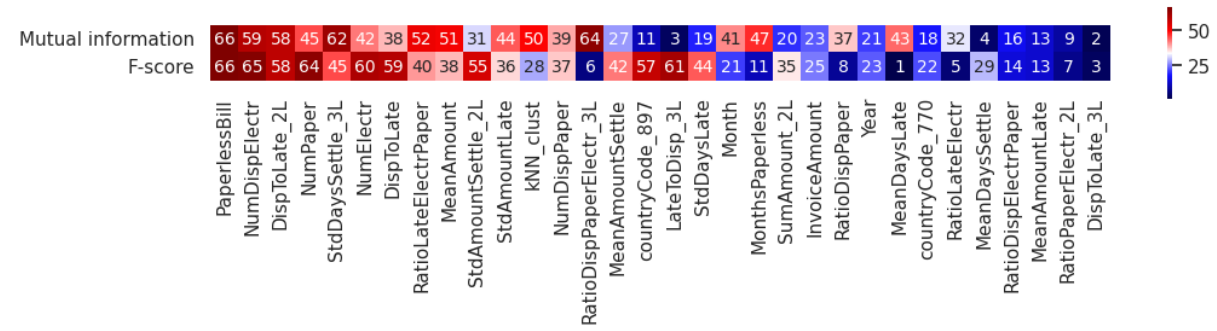

Figure 56: Heatmap of features' rank using 2 criteria (mutual information, ANOVA F-score) for the purpose of classification.

3.2.5 Splitting methods

"Random split":

The bills are randomly assigned to the training and test set according to a specified ratio (default: 80% training / validation, 20% test).

"N-split":

The test set consists of the n-th bill of each customer and all the previous bills are used to form the training set.

"Date-split":

The training set consists of all the bills up to a specific date and the test set consists of the next bill available per customer.

The last two methods are more "natural" since the splitting follows the chronological order.

The features selected using each method are:

<u>Random split:</u>

Classification:

- PaperlessBill
- RatioLatePaperElectr_3L
- NumDispElectr
- RatioPaperElectr
- DispToLate_2L
- StdAmountDisp
- NumPaper
- RatioPaperElectr_3L
- StdDaysSettle_3L
- StdDaysLate_2L
- NumElectr

Regression:

- RatioPaperElectr_3L
- NumDisp_2L
- countryCode_818
- LateToDisp_2L
- StdAmountDisp
- NumDisp
- SumAmount_2L
- RatioDispPaperElectr_3L
- MeanAmount
- RatioDispPaper
- DispToLate_3L

- NumLate
- DispToLate
- RatioDispElectr
- RatioDispPaperElectr_2L
- RatioLatePaper
- MeanAmountSettle
- MeanDaysLate
- countryCode_391

Date split (2013-05-01):

Classification:

- RatioLateElectrPaper
- MeanAmountLate
- RatioPaperElectr
- RatioDispPaperElectr_2L
- NumDispElectr
- StdDaysSettle
- StdAmountDisp
- NumLatePaper_2L
- NumDisp_2L
- StdAmountSettle_2L
- MonthsPaperless
- StdAmountLate
- DispToLate
- PaperlessBill
- RatioPaperElectr_2L

Regression:

- RatioLateElectrPaper
- MeanAmountLate
- RatioPaperElectr
- RatioDispPaperElectr_2L
- NumDispElectr
- StdDaysSettle
- StdAmountDisp
- NumLatePaper_2L
- NumDisp_2L
- StdAmountSettle_2L
- MonthsPaperless
- StdAmountLate
- DispToLate
- PaperlessBill
- RatioPaperElectr_2L

NumBills split (10):

Classification:

- NumLateElectr_2L
- RatioDispPaperElectr_3L
- RatioPaperElectr_2L
- MonthsPaperless
- RatioLatePaperElectr_2L
- RatioLateElectr
- NumDisp_2L
- LateToDisp_3L
- MeanAmount
- countryCode_818
- RatioLatePaper
- countryCode_391
- PaperlessBill
- MeanAmountLate
- StdDaysLate

Regression:

- RatioDispElectrPaper_3L
- NumDisp_2L
- countryCode_818
- MeanDaysSettle
- StdAmount_3L
- NumLate
- StdDaysLate_3L
- MeanAmountDisp
- RatioPaperElectr
- kNN_clust
- NumLate_2L
- StdAmountDisp
- RatioDispPaperElectr_3L
- countryCode_391
- MeanAmount

3.3 Models

3.3.1 Hyperparameter selection

The hyperparameters of the models were selected using either a genetic algorithm (classification) or grid search (regression).

Genetic algorithm

- 10 models
- 10 generations
- 10% mutation probability
- 80% crossover probability
- top 3 models (scoring: accuracy) kept per generation
- + 2 random models
- 5-fold cross-validation

Grid search

- metric: neg_root_mean_squared_error
- 5-fold cross-validation

Below are the hyperparameters that resulted in the best accuracy score (classification) or RMSE (regression), separately and in the context of stacking, using random splitting of the data and all the features remaining after those highly correlated are eliminated.

3.3.2 Classification

Base estimators:

- AdaBoostClassifier
    - base: DecistionTree, estimators: 77, algorithm: SAMME.R
- DecisionTreeClassifier
    - criterion: entropy, splitter: random, min. split: 3
- GradientBoostingClassifier
    - loss: deviance, learning rate: 0.01, estimators: 100, criterion: mse
- KNeighborsClassifier
    - n: 5, leaf size: 90, algorithm: kd tree
- LinearSVC
    - loss: hinge, C: 0.16

- MLPClassifier
    - hidden layer size: 36, activation: logistic, solver: adam, learning rate: adaptive
- RandomForestClassifier
    - estimators: 100, criterion: gini, max. features: sqrt, bootstrap: True
- SGDClassifier
    - loss: hinge, penalty: l2, alpha: 0.0001
- SVM
    - kernel: RBF, degree: 3, decision function shape: ovo
- XGBoostClassifier
    - max_depth: 7, learning_rate: 0.001, n_estimators: 10000, objective: binary:hinge, booster: gblinear, num_boost_round: 100
- QuadraticDiscriminantAnalysis
- Perceptron
- GaussNB
- Neural network (Tensorflow)
    - activation: relu / softplus, optimizer: adam, loss: BinaryCrossentropy
    - 300 nodes, 51 epochs, 0.36 dropout rate

Final estimator:

- LogisticRegression
    - C: 0.79, solver: lbfgs

3.3.3 Regression

Base estimators:

- AdaBoostRegressor
    - base: DecistionTree, estimators: 50, loss: linear, learning_rate: 0.1
- DecisionTreeRegressor
    - criterion: mae, splitter: random, min. split: 5
- KNeighborsRegressor
    - n: 10, leaf size: 10, algorithm: brute, weights: distance
- MLPRegressor
    - hidden layer size: 10, activation: relu, solver: adam, learning rate: constant

- RandomForestRegressor
    - estimators: 500, criterion: mae, max_depth: 14, max. features: sqrt, bootstrap: True
- XGBoostRegressor
    - n_estimators: 10000, objective: reg:tweedie, booster: gblinear, num_boost_round: 1000

Final estimator:

- LinearRegression

3.4 Evaluation

Initially, random splitting and all the features (only removing the highly correlated) are used to determine the top-performing types of models that are then tested using all the combinations of: {random (test set: 20%), 10-th bill, date} split x {all, top-15} features.

3.4.1 Classification

The classes used to train the models are two and correspond to the `late` status (0: not late, 1: late). The results presented in tables 3-4 and figures 57-65, show that:

- The accuracy achieved ranges from 59% to 84% (table 3). Except for perceptron, all the other models had at least 74% accuracy, though it differs class-wise (figure 57).
- The top models are tree-based using boosting (XGBoost, AdaBoost) or neural networks (NN, MLP), with the NN model only having significant difference between the training and test set, indicating overfitting.
- The late class prediction accuracy ranges from 63% to 73% among the top-performing models (figure 58).
- The probabilities assigned to the predicted class by the top model, XGBoost, are mainly quite high (figure 59) even in the event of misclassification (figure 60).
- XGBoost and NN consider different features more important (figures 57-58), sharing half of their top-10.
- The impact of the most important feature, the mean number of days to settle the bill, is greater when others have little impact and is more important for non-late bills.
- Three features, the ratio of electronic to paper bills, the invoice amount and the standard deviation of the late bills' amount are more important for late bills.
- After the ground truth labels are randomly permuted, the features considered important change (figures 56-57), giving some credibility to the original models.
- The use of the top-15 features (~35% of them concern recent trends) only led to better accuracy with the "date split".

Table 3: The accuracy score of the classification models (ordered) on the test set (random split, all features).

	Accuracy (test set)
XGBoost	84%
NN	83%
AdaBoost	82%
MLP	81%
GaussianProcess	81%
Stacking	80%
KNeighbors	79%
LinearSVC	79%
SVM	79%
RandomForest	79%
SGD	79%
LR	79%
QDA	77%
DecisionTree	76%
GaussianNB	74%
Perceptron	59%

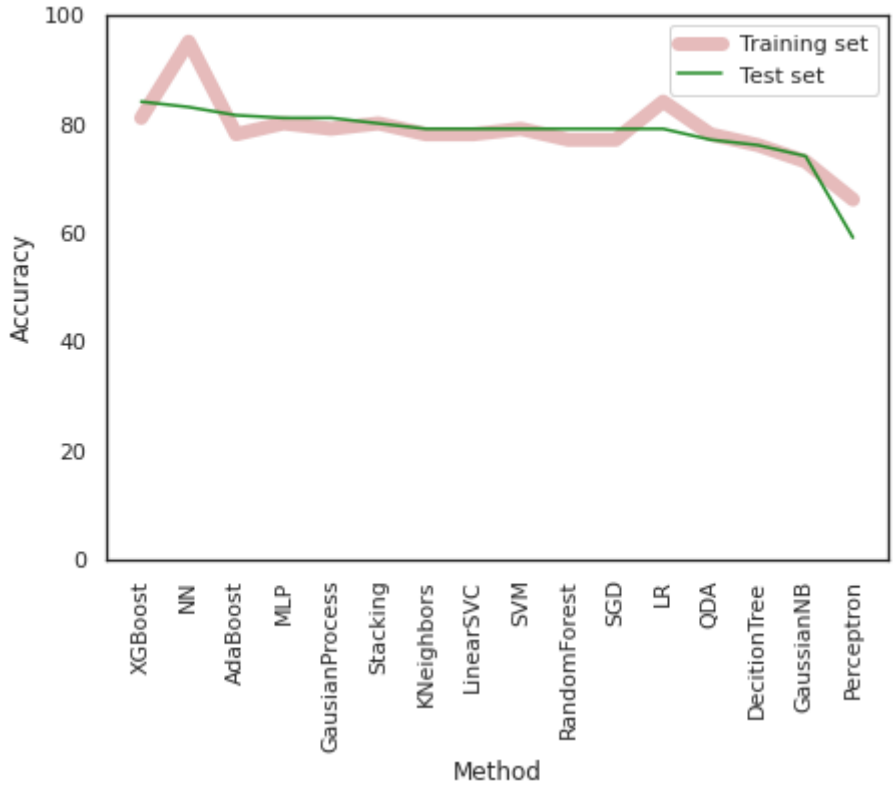

Figure 57: The accuracy score of the classification models (ordered) on the training and test sets (random split, all features).

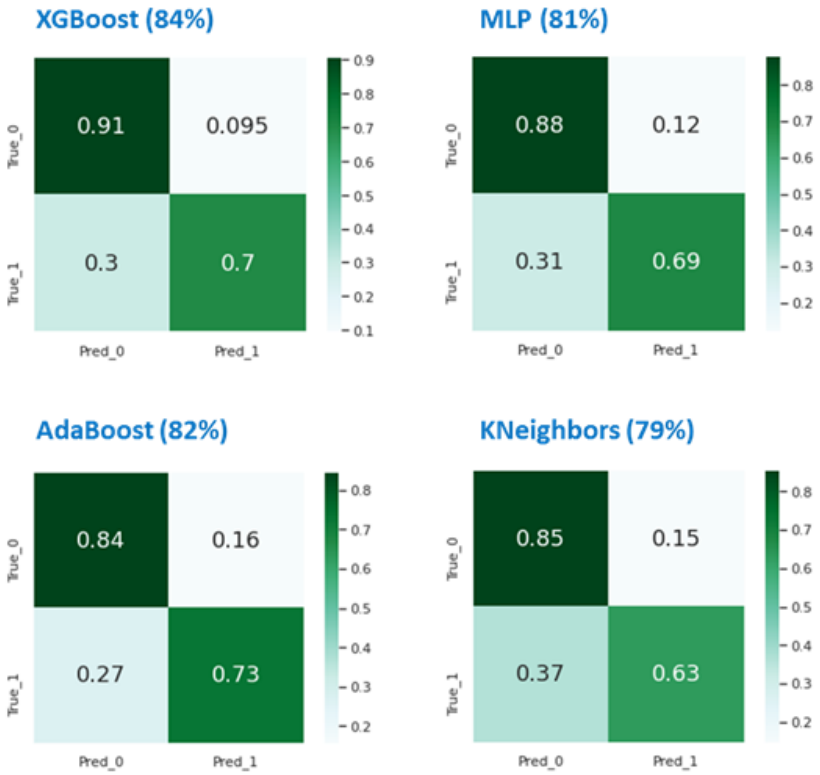

Figure 58: Confusion matrices of the top-performing models (random split, all features).

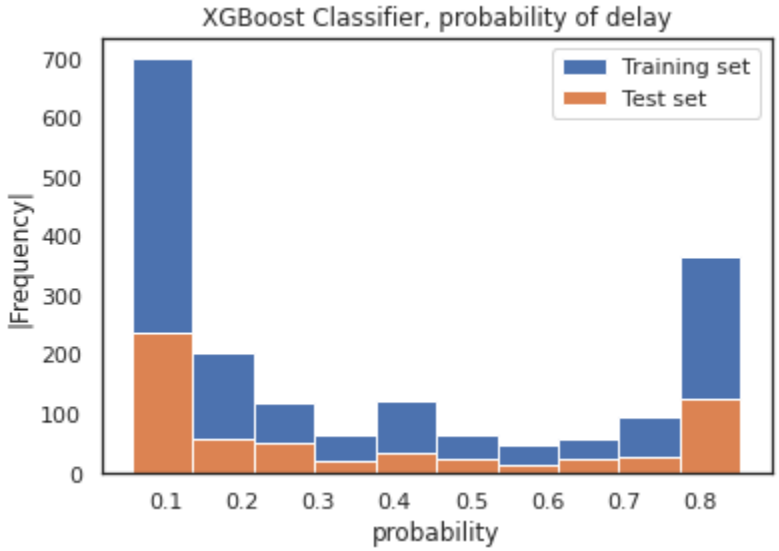

Figure 59: Histogram of the probability of delay using the XGBoost classifier (best model).

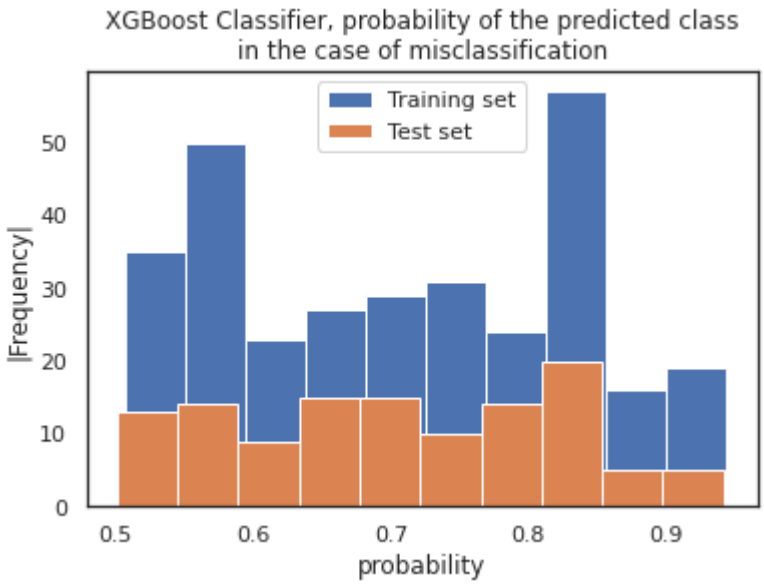

Figure 60: Histogram of the probability of the predicted class in the case of misclassification using the XGBoost classifier (best model).

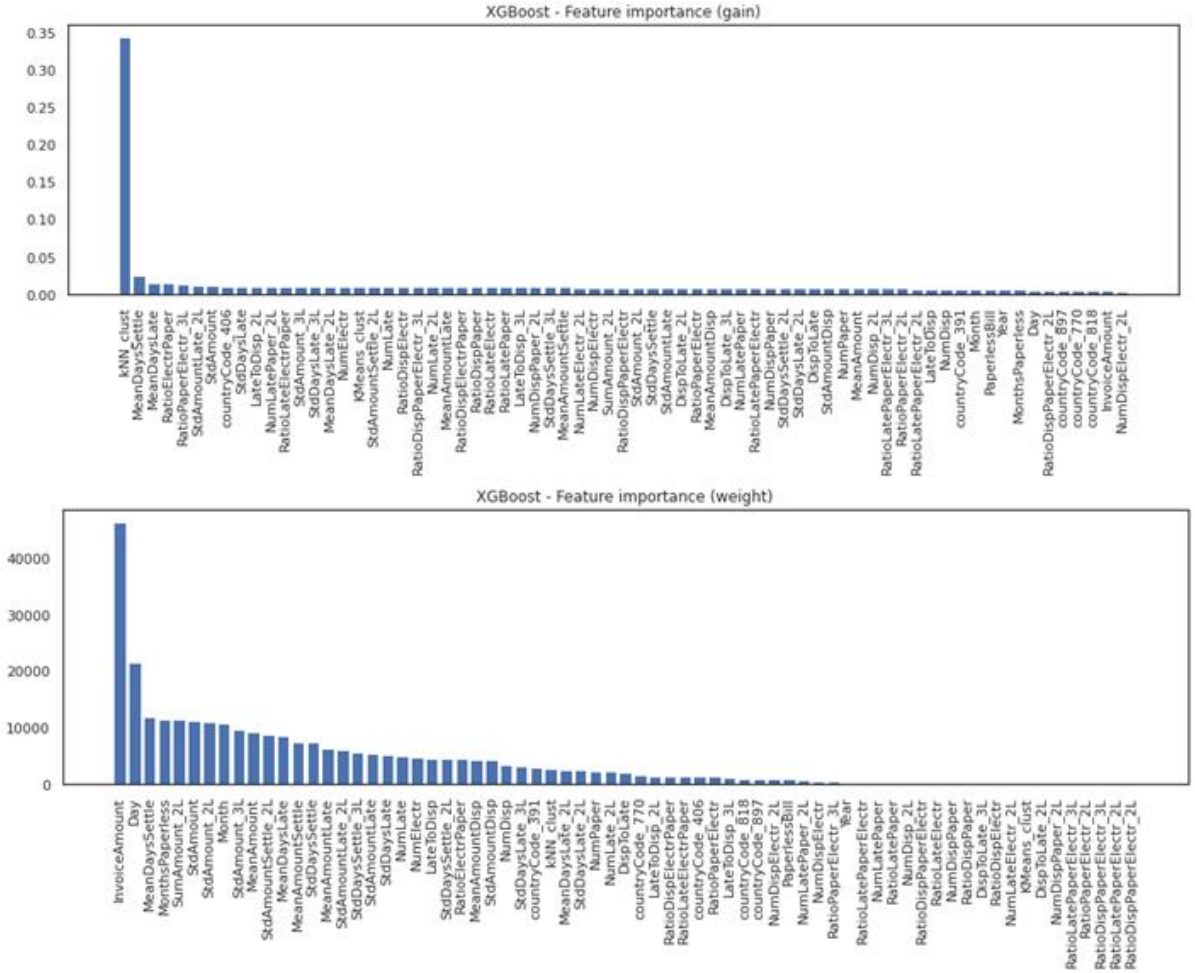

Figure 61: The features' importance using the XGBoost classifier (best model, all features, random split). The gain, refers to the average gain across all the splits and the weight to the number of times the feature is used to split the data across all the trees.

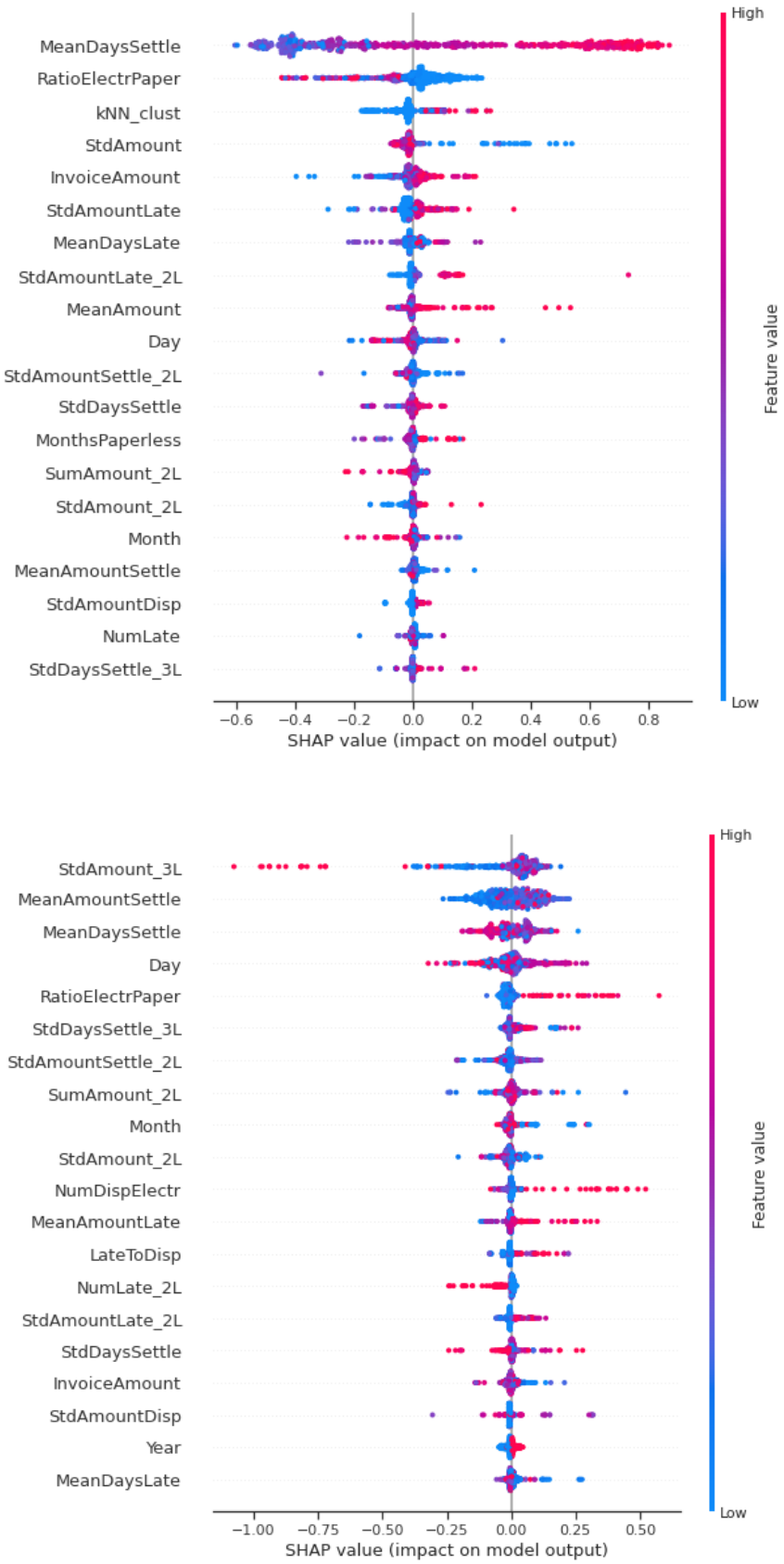

Figure 62: Density scatter plot of the SHAP values [30] for each feature using the XGBoost classifier (gbtree booster instead of gblinear (best model), all features, random split), without (first plot) or without (second plot) random permutation of the target values. The features are sorted by the sum of the SHAP value magnitudes across all bills.

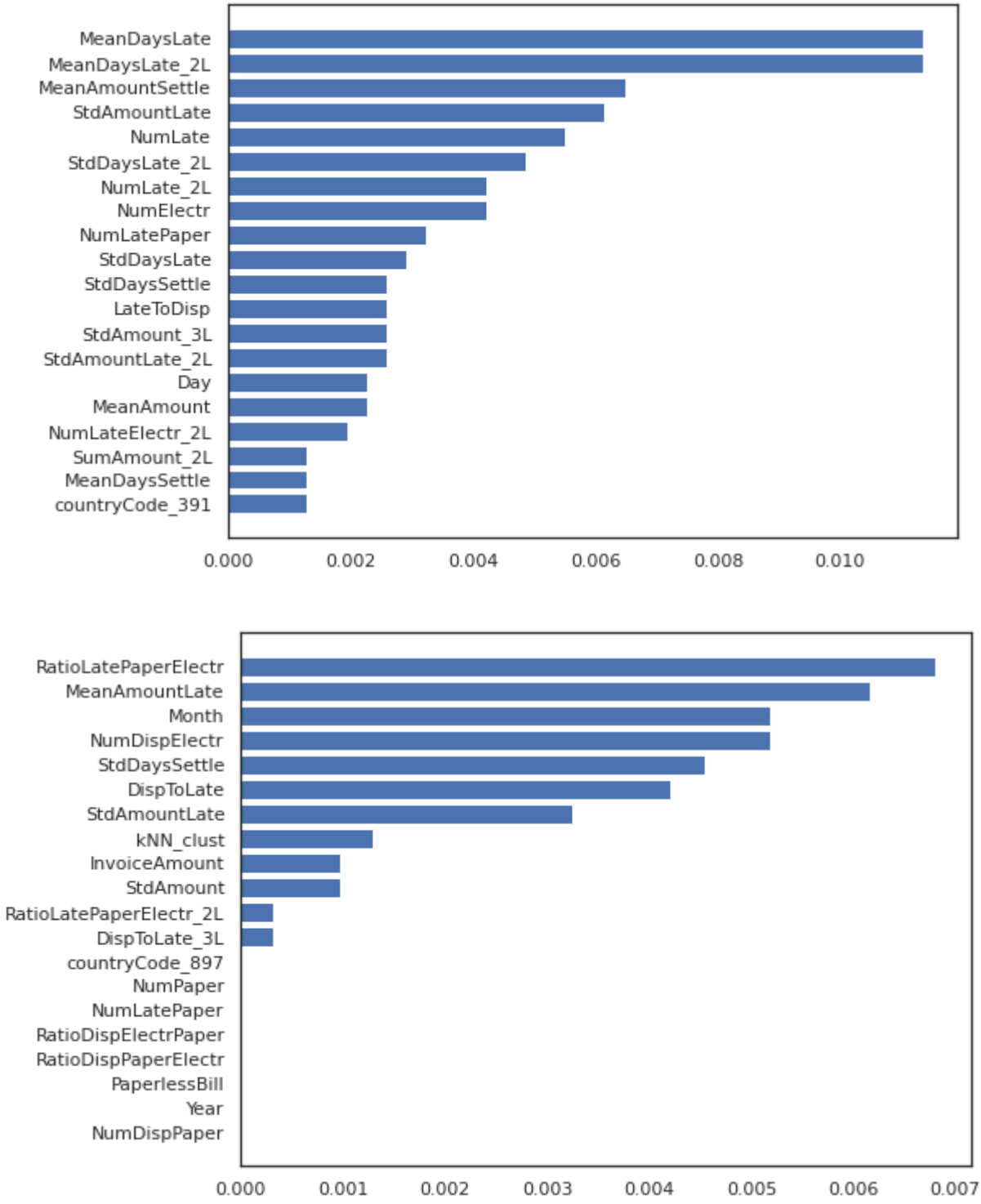

Figure 63: The features' importance using the XGBoost classifier (best model, all features, random split) and MDA (Mean Decrease Accuracy) metric.

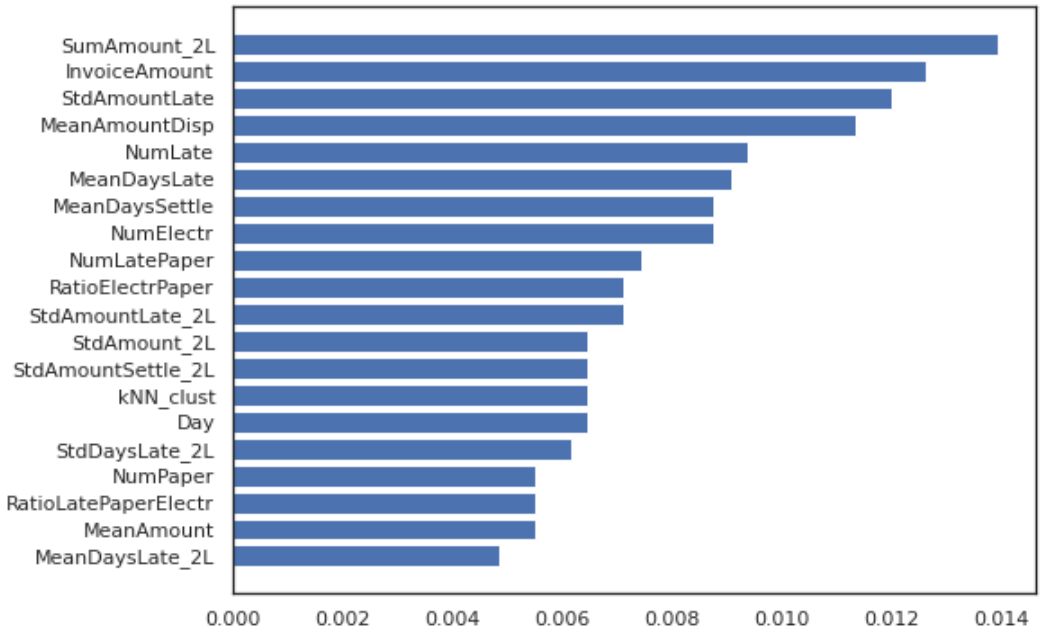

Figure 64: The features' importance using the NN classifier (best model, all features, random split) and MDA (Mean Decrease Accuracy) metric.

Table 4: The accuracy of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset.

	Split type	Features	Accuracy (test set)
XGBoost	Random	All	84%
	Random	Top-15	74%
	Date	All	78%
	Date	Top-15	83%
	NumBills	All	75%
	NumBills	Top-15	60%
NN	Random	All	83%
	Random	Top-15	78%
	Date	All	78%
	Date	Top-15	80%
	NumBills	All	78%
	NumBills	Top-15	76%
AdaBoost	Random	All	82%
	Random	Top-15	74%
	Date	All	78%
	Date	Top-15	83%
	NumBills	All	75%
	NumBills	Top-15	60%

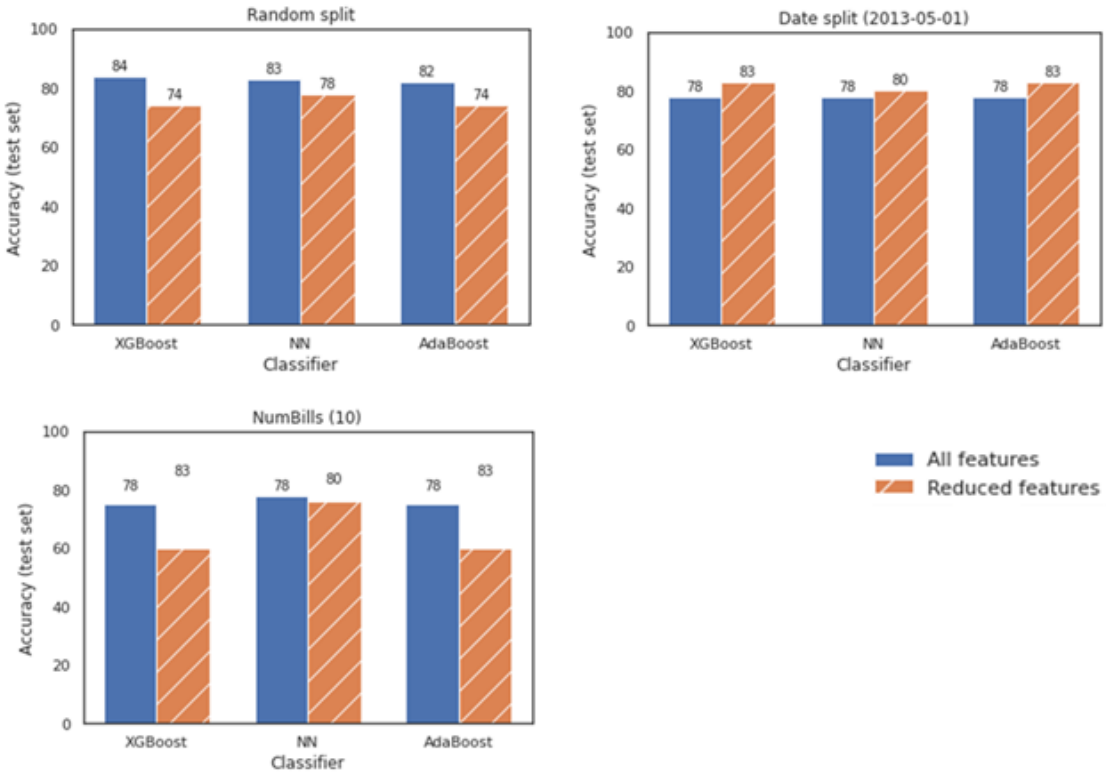

Figure 65: The accuracy of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset.

Also, CNN LSTM models were used to extract intermediate representations but the accuracy achieved was +0.5% at best. The input of these models is a set of two-dimensional matrices consisting of the customer's information at each time point (figure 66).

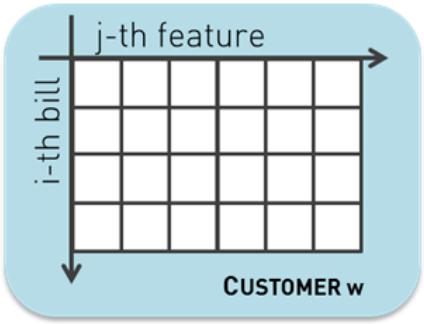

Figure 66: The two-dimensional representation of customer's `w` history.

3.4.2 Regression

The predicted value is the number of days until the bill is settled. The "classes" mentioned below, correspond to the following intervals: 0: [0,10] days, 1: (10,20], 2: (20,30], 3: (30,40], 4: (40,∞) days. The results presented in tables 5-6 and figures 68-71, show that:

- The RMSE ranges from 7.29 to 11.01 (table 5).
- The top models are tree-based using boosting (XGBoost, RandomForest) and stacking or neural network (MLP).

- Even though the "accuracy" calculated (figure 68) is low (51%-55%), most of the misclassified cases belong to the previous or next “class”. It is greatly reduced when the delay exceeds 40 days.
- The absolute difference in days between the actual and predicted values was as high as 30 days.
- The use of the top-15 features (~35% of them concern recent trends), improved the RMSE with the “n-th bill split” and sometimes with the “date split”.
- XGBoost and NN share most (7/10) of the features they consider more important (figures 57-58).

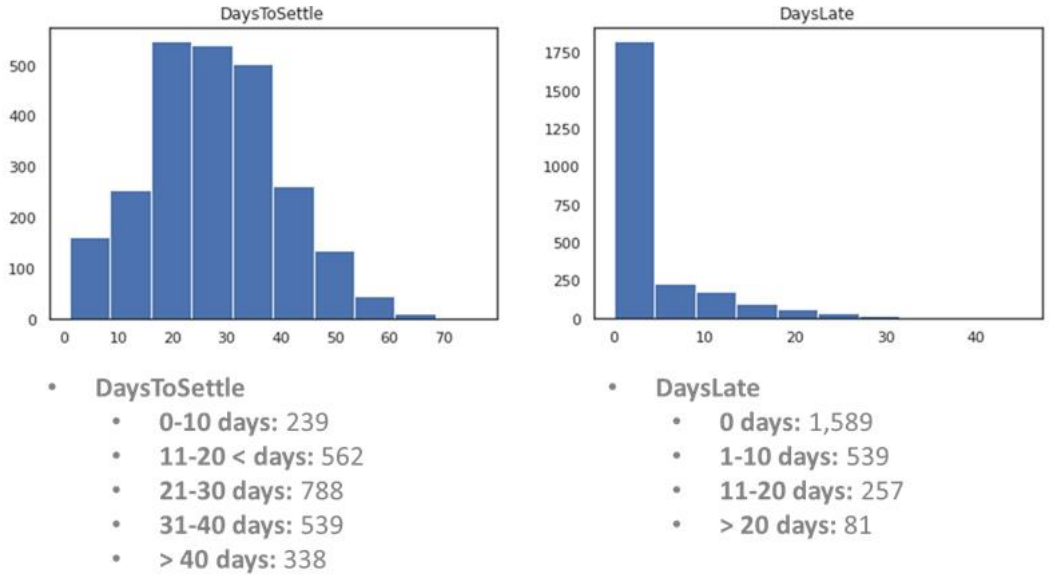

Figure 67: Histograms of the number of days until the bill is settled and the number of days the bill was late, along with the cardinality of the corresponding classes used to calculate the “accuracy” of the regression models.

Table 5: The RMSE score of the regression models (ordered) on the test set (random split, all features).

	RMSE (test set)
Stacking	7.29
XGBoost	7.32
MLP	7.41
RandomForest	7.55
AdaBoost	7.77
KNeighbors	8.27
DecisionTree	11.01

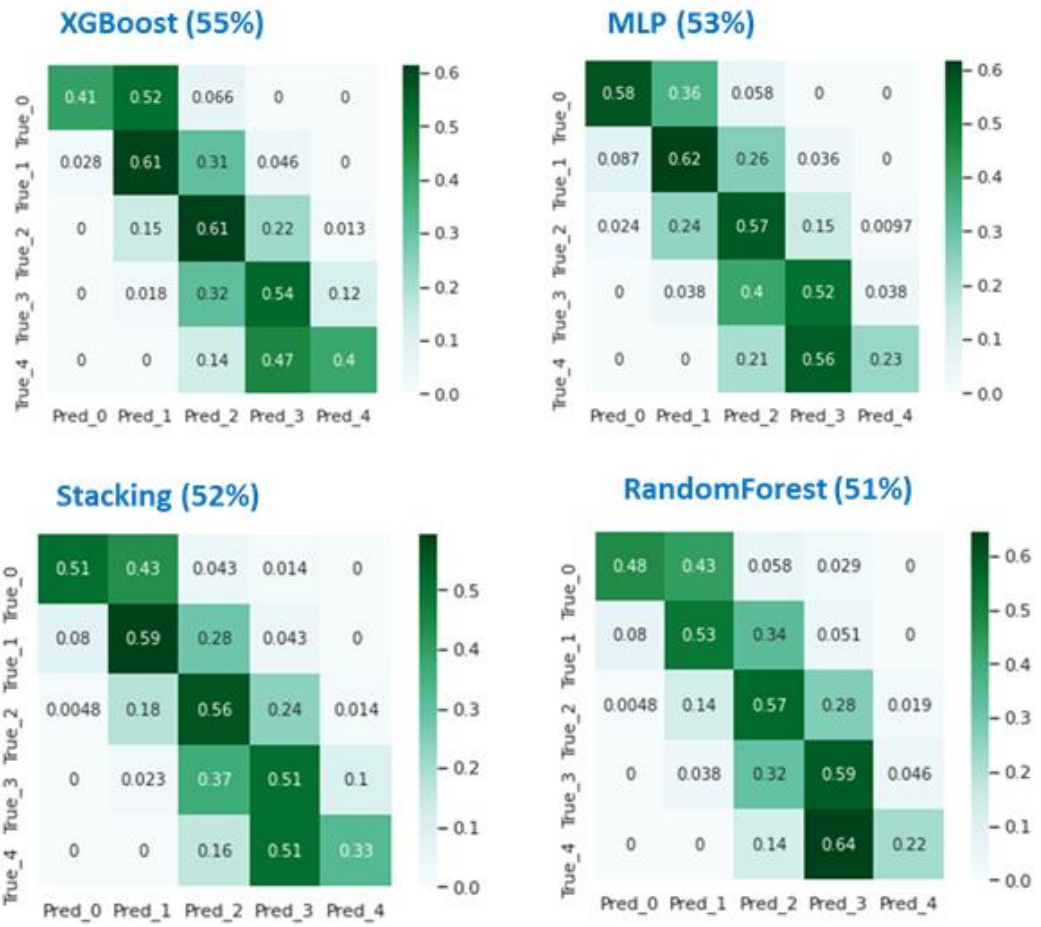

Figure 68: Confusion matrices of the top-performing models (random split, all features). Although the "accuracy" is low, most of the misclassified cases belong to the previous or next "class".

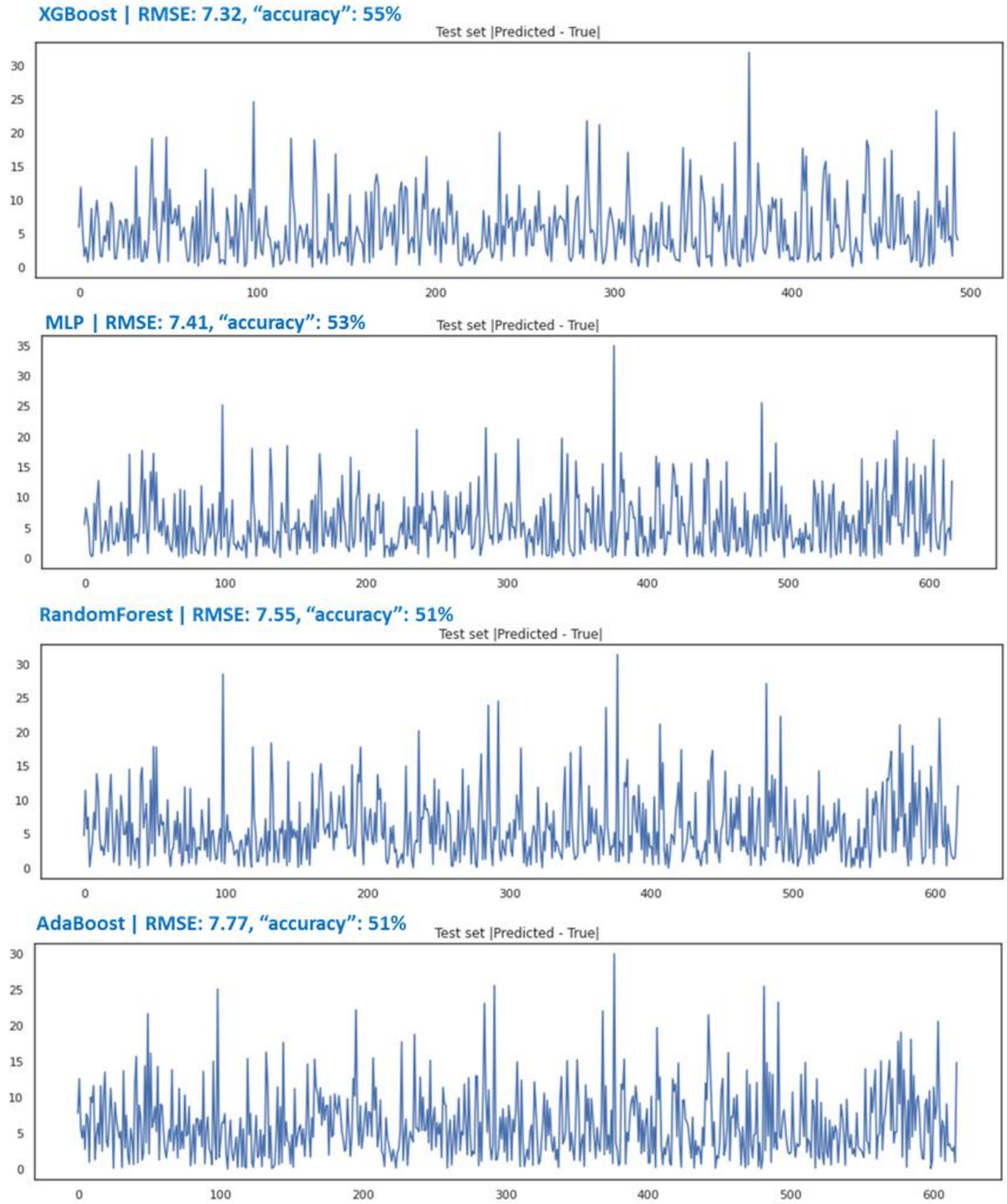

Figure 69: The absolute difference between the actual and predicted values of the test set, using the top-performing models (random split, all features).

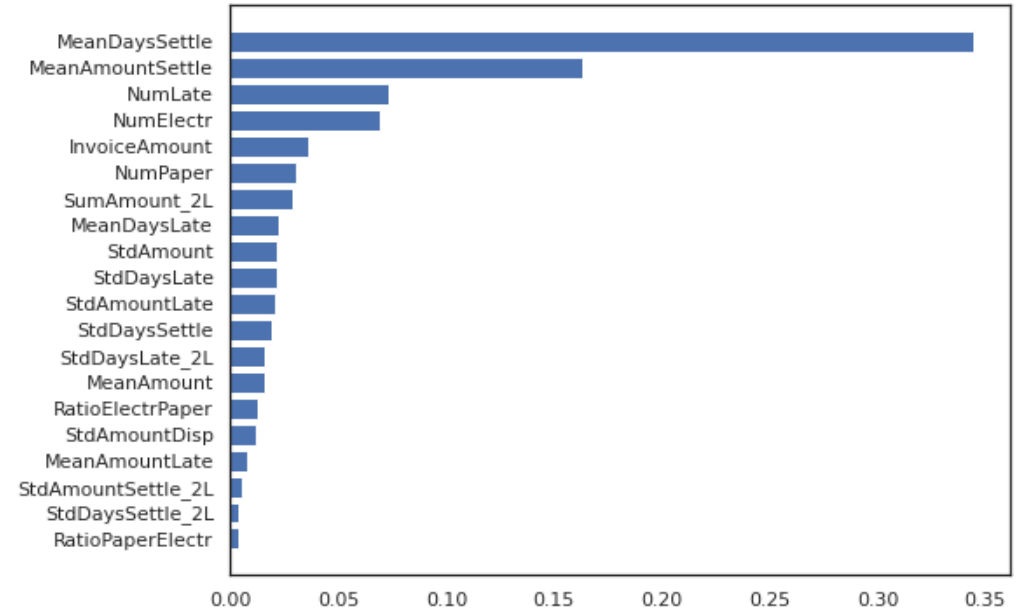

Figure 70: The features' importance using the XGBoost regressor (all features, random split) and MDA (Mean Decrease Accuracy) metric.

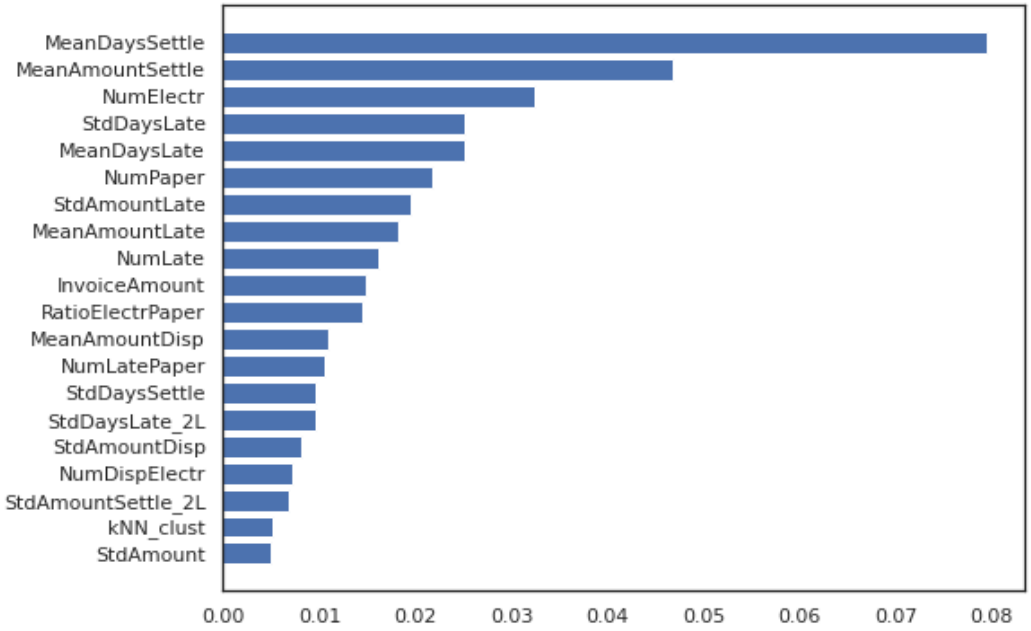

Figure 71: The features' importance using the NN regressor (all features, random split) and MDA (Mean Decrease Accuracy) metric.

Table 6: The results of the top-performing models, using all the splitting methods and all or the top-15 features of the dataset.

	Split type	Features	RMSE (test set)	"Accuracy" (test set)
Stacking	Random	All	7.29	52%
	Random	Top-15	7.55	51%
	Date	All	7.59	55%
	Date	Top-15	7.49	53%
	NumBills	All	7.73	55%
	NumBills	Top-15	8.28	50%
XGBoost	Random	All	7.32	55%
	Random	Top-15	7.90	50%
	Date	All	7.44	53%
	Date	Top-15	9.06	43%
	NumBills	All	7.89	55%
	NumBills	Top-15	7.74	50%
MLP	Random	All	7.41	53%
	Random	Top-15	8.85	43%
	Date	All	7.98	47%
	Date	Top-15	9.45	44%
	NumBills	All	8.35	55%
	NumBills	Top-15	7.28	60%
RandomForest	Random	All	7.55	51%
	Random	Top-15	7.68	50%
	Date	All	7.19	55%
	Date	Top-15	7.45	54%
	NumBills	All	8.26	50%
	NumBills	Top-15	7.14	65%

3.4.3 Classic time-series forecasting - Prophet

Finally, the Prophet procedure was used to predict the days until the n-th bill was settled (n: 5, 10, 15, 20, 25, 30), providing the status of all the previous bills per customer for the fitting along with the date they were issued. Only one growth type, "logistic", allowed for both the floor (0) and ceiling (cap, 80) values to be specified and so it was the one selected to carry out the tests since the alternatives were yielding negative values.

Even though the floor and cap values were defined, the predictions were sometimes out of this range. Generally, the results (figures 72-77) show that at least 24 past bills are required to make its performance comparable to that of the previous section's models. Though, only a few customers (19) have 30 or more bills, making the evaluation of its performance problematic. The higher accuracy in the last class, when using less than 25 bills, is attributed to the predictions having systematically much greater values than the cap.

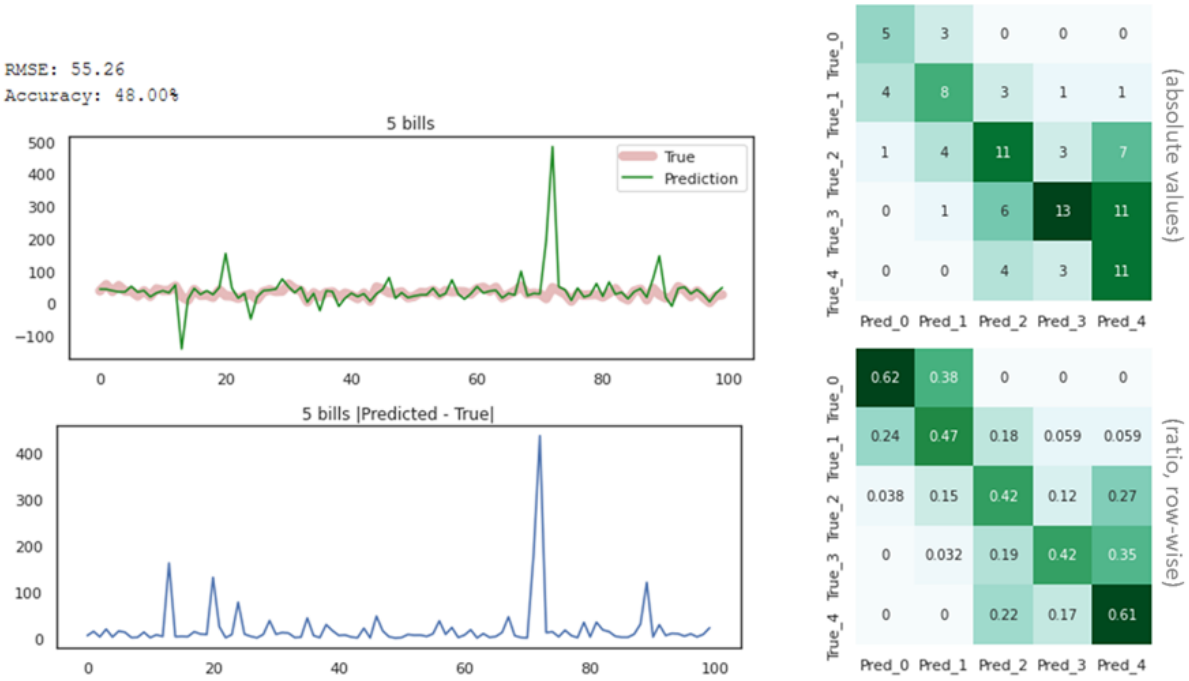

Figure 72: Prophet results - prediction of the 5th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. Even though floor and cap values are defined, the predictions are sometimes out of this range. The predictions having values >> cap, contribute to the higher accuracy observed in the last class.

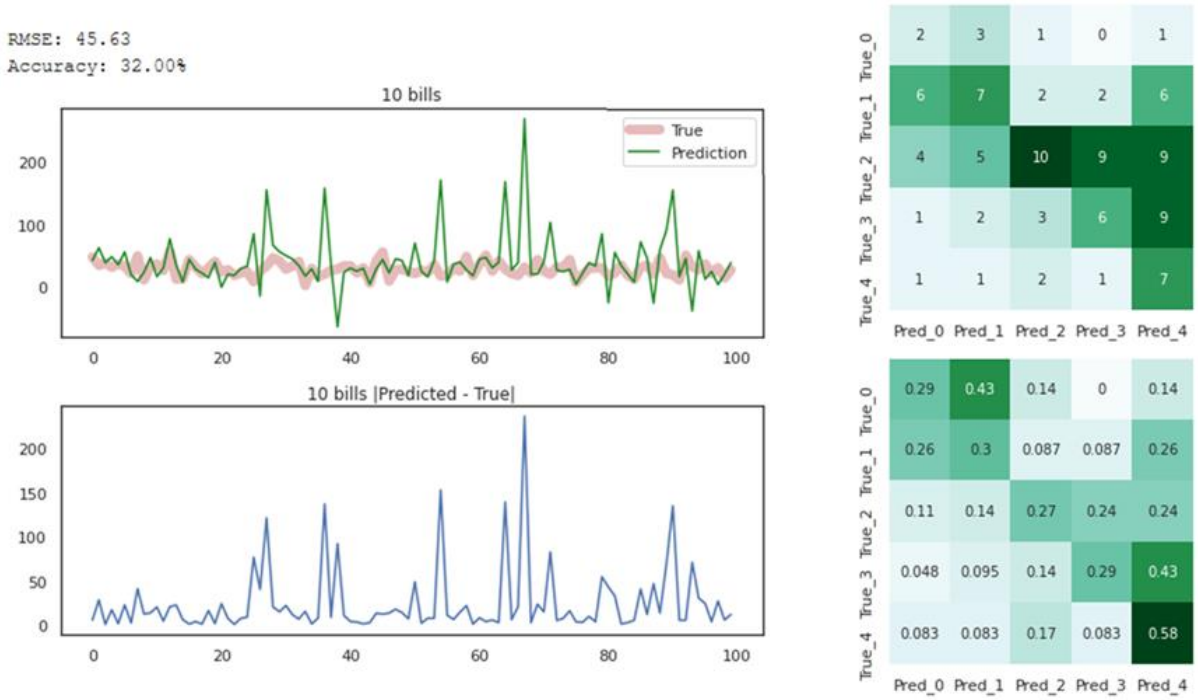

Figure 73: Prophet results - prediction of the 10th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.

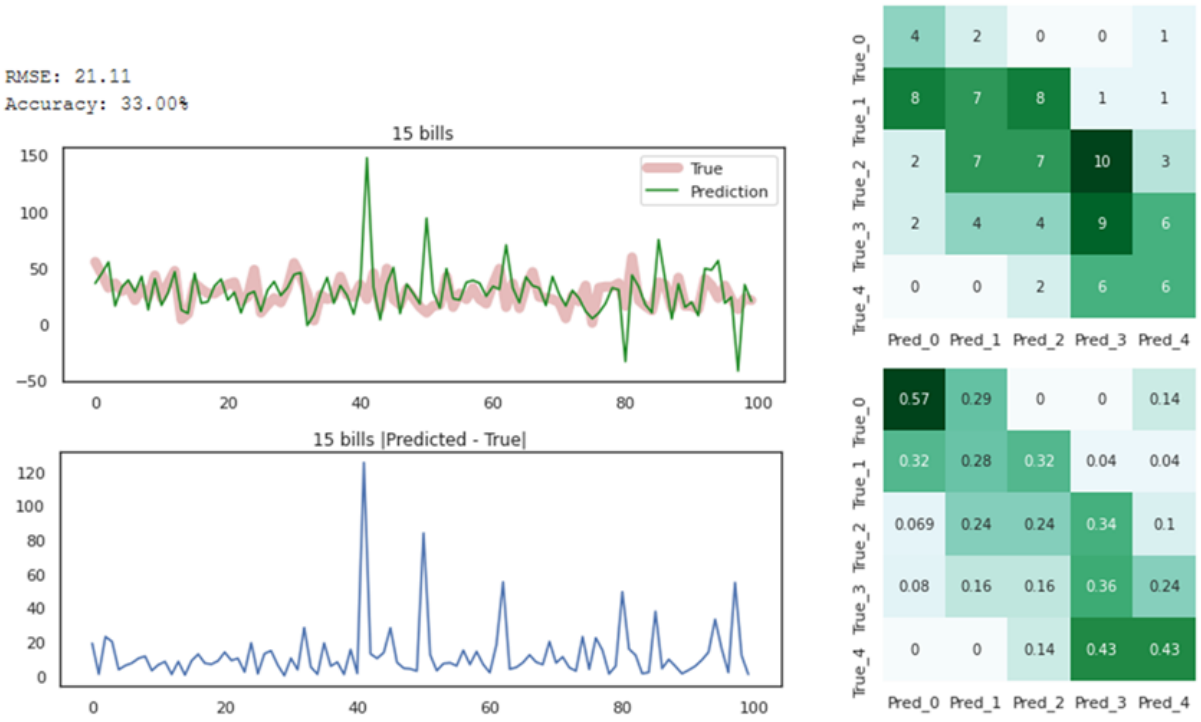

Figure 74: Prophet results - prediction of the 15th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.

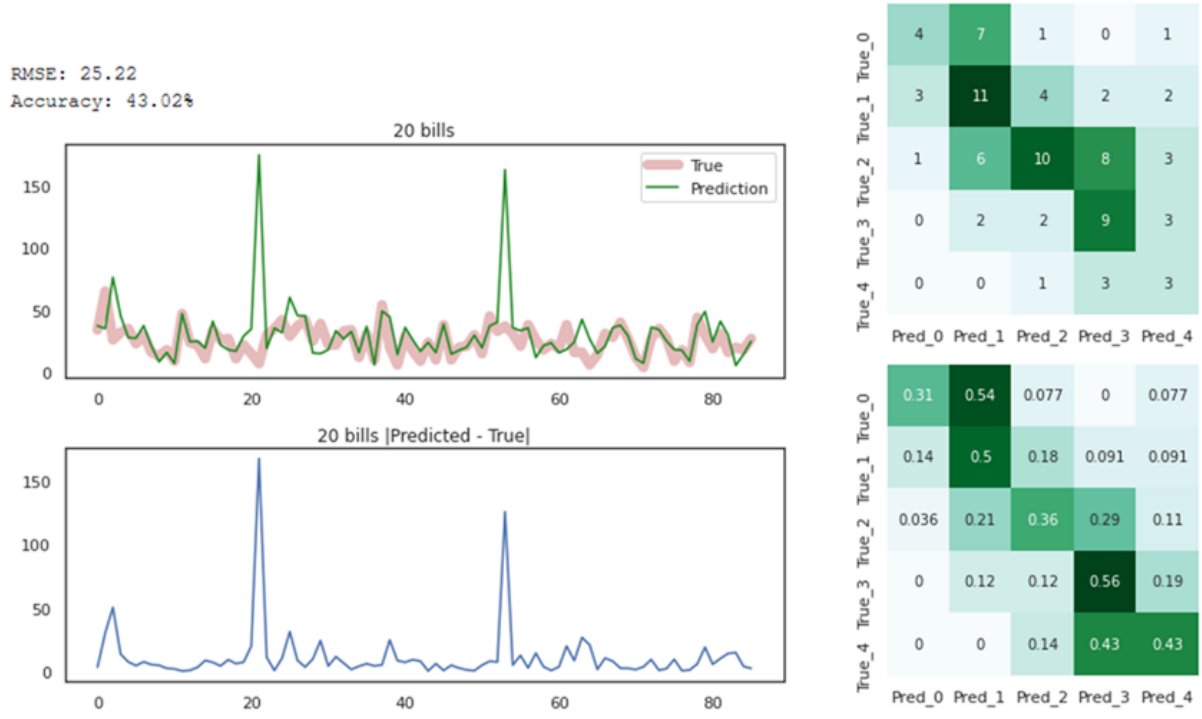

Figure 75: Prophet results - prediction of the 20th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios.

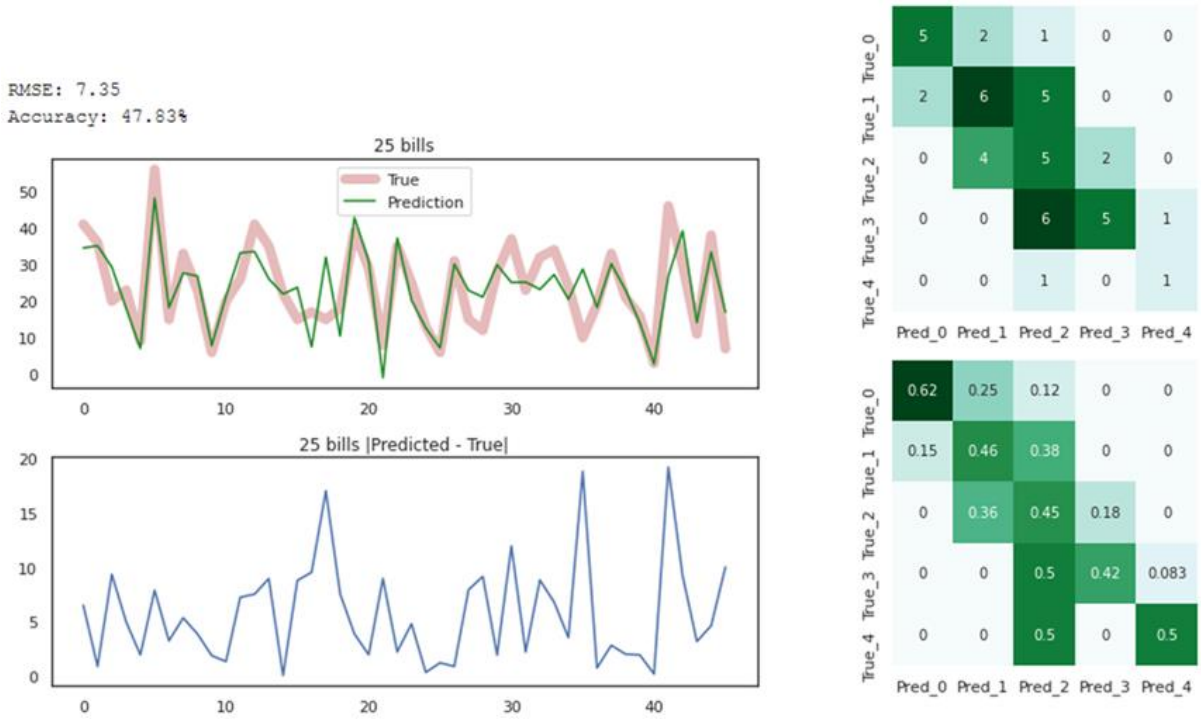

Figure 76: Prophet results - prediction of the 25th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. The predictions are much better now that 24 bills are available to fit the models.

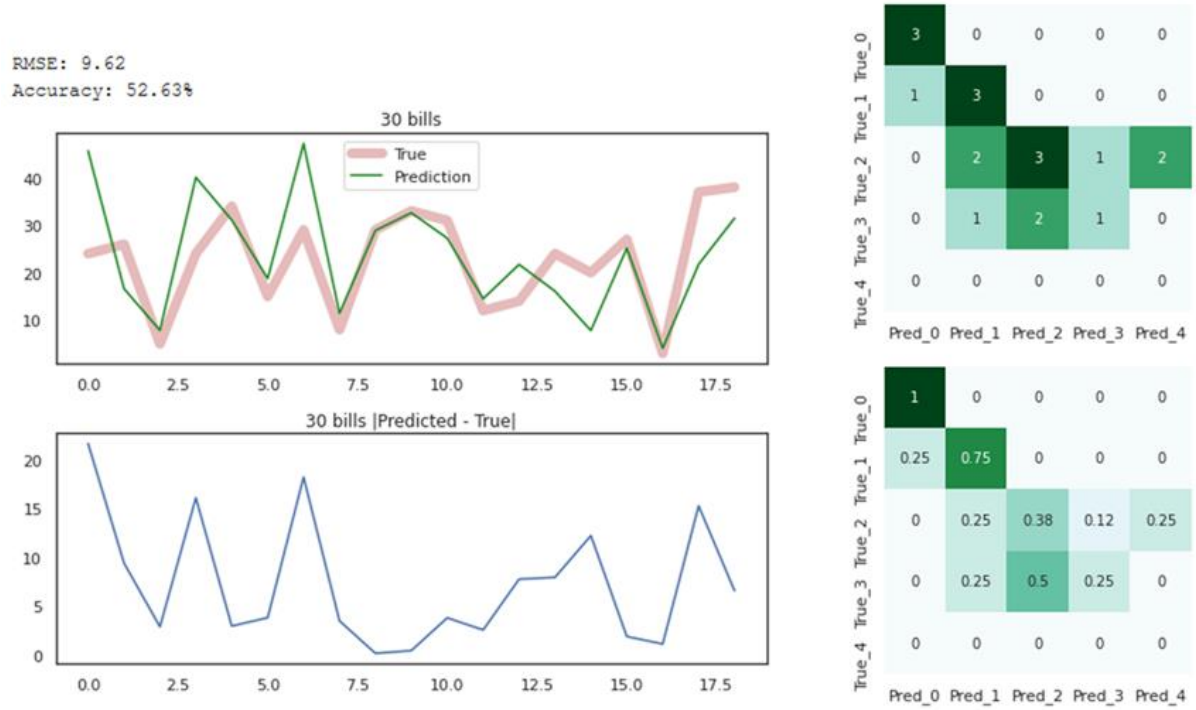

Figure 77: Prophet results - prediction of the 30th bill, absolute difference between the predicted and actual values, heatmap using the actual values and heatmap using the row-wise ratios. There are only 19/100 customers having 30 bills.

3.5 Technical information

The experiments were performed using: Python 3.7.12 [29], scikit-sklearn 0.22.2, Tensorflow 2.6.0 [1], XGBoost 1.4.1, sklearn-genetic-opt 0.6.1 [32], umap-learn-0.5.1 and Prophet 0.7.1.

4. CONCLUSIONS AND FUTURE PROSPECTS

In the case of classification, the accuracy of the three best models (XGBoost, NN, AdaBoost) on the test set was 82%-84%, using all the features and “random split”. The greatest difference between the training and test set accuracy was observed in the NN, with the training set being almost perfectly classified, indicating some degree of overfitting. Looking at the confusion matrices, AdaBoost was best at classifying the late bills (73%) and XGBoost was best at classifying the non-late bills (91%). Also, the probabilities XGBoost assigned to the predictions were mainly high and the same was true for about half of the misclassified bills. The lowest accuracy across the top-performing models involved the “n-th bill split”. Consistently, the use of the top-15 features (~35% of them concern recent trends) only led to better accuracy with the “date split”.

Based on how much the accuracy decreases when a feature is not available, XGBoost and NN consider different features more important (figures 63-64); half of their ten most important features are common: mean days late, number of late bills, standard deviation of the late bills’ amount, number of late paper bills and the number of electronic bills. Furthermore, figure 62 shows the impact of the features having the largest sum of SHAP value magnitudes across all the bills. We observe that the most important feature, the mean number of days to settle the bill, has greater impact when the other features have little to no impact and its magnitude in determining the class is greater for non-late bills. On the opposite side, the ratio of electronic to paper bills, the invoice amount and the standard deviation of the late bills’ amount are more important for late bills. Furthermore, if the ground truth labels are randomly permuted, the features considered important change drastically (figures 62-63), indicating the explanation of the original models about the relationship between the input and the prediction has some validity.

In the case of regression, the RMSE of the three best models (Stacking, XGBoost, MLP) on the test set was 7.29-7.41, using all the features and “random split”. The corresponding “accuracy” was up to 55%, but most of the misclassified bills belong to either the previous or the next “class”. The absolute difference in days between the actual and predicted values was as high as 30 days. With the use of the top-15 features (~35% of them concern recent trends), the RMSE consistently improved with the “n-th bill split” and sometimes with the “date split”. This time, XGBoost and NN have more common features (seven) among the ten most important (figures 70-71): mean days to settle the bill, mean amount settled, num of late bills, number of electronic bills, the invoice amount and mean days late, but their impact is much greater (up to seven times) in XGBoost. It should be noted that the “accuracy” is greatly reduced when the delay exceeds 40 days, leading to the hypothesis that other features are important for this group and possibly not available. It is characteristic that if we know whether the current bill is disputed, the same models have RMSE 5.8 and 6.3 instead of 7.3 and 7.4, and “accuracy” 63% and 59% instead of 55% and 53%, respectively. More interestingly, the “accuracy” at predicting the last “class” doubles in NN but in XGBoost the greatest increase concerns the first “class” (1.7 times). The results are similar if we add that information to the classification models; the accuracy of XGBoost and NN increases up to 90% and 85% instead of 84% and 82%, respectively. At the same time, the models are almost equally good at discerning the two classes even though the non-late bills’ class is underrepresented. Thus, factors associated with the invoice being disputed or the handling process after it is disputed should be tested.

The classic time-series method, Prophet, required to know 24 past bills (approximately 2 years) per customer in order to have comparable performance (RMSE: 7, “accuracy”: 48%, ~45 customers), but more data did not lead to better RMSE.

Overall, the accuracy achieved can be considered good / promising, given the limited data available and the relatively low ratio of non-late to late bills. The ensemble learning methods, which are entirely decision tree-based or include decision trees, performed better in both cases, representing two of the three top models. This is in accordance with the papers described in section 1.2, where 70% of the best models use decision trees and 70% of them involve ensemble learning (random forest, XGBoost). Though, in contrast, only 10% of the best models involved neural networks, but here one out of the three best models per case did. Also, the best accuracy these models acheived, ranged from 68% to 96% and the dataset with the lowest cardinality had ten times more records than the dataset used here. Tree-based methods tend to perform better than NN with tabular data [37] and they also make the interpretation of the decisions more straightforward; ensemble learning methods are employed to counter their variance. The linear regression models commonly used in business analysis cannot handle nonlinear relationships well.

In a real application, a bigger dataset, having more unique customers and records and a greater ratio of late bills, spanning a longer time period, potentially leading to the derivation of more, useful features, would be needed to establish the validity of the model. It would also allow to better understand the impact of certain characteristics on customer subgroups with different behaviour and eliminate outliers. This could be especially important if low cardinality subgroups responsible for disproportionate loss are identified. Then, the customers could be pre-clustered and used to separately train base models, since the importance of the features may vary. After more data are accumulated over time, some features may prove to be degrading the performance.

Apart from having a better training set, the test set would be possible to consist of completely unseen customer records and allow better assessment of how small perturbations of the input affect the results, making the process more reliable and robust. Furthermore, the more natural splitting methods (“n-th split” and “date-split”) could be utilized to exhaustively test the models and evaluate how the performance changes when the number of bills available changes. Additionally, different time windows could be considered for capturing recent trends, followed by comparison of the features with the highest impact.

Finally, sector-specific or global economic factors may influence the ability or willingness of the customers to pay on time and such features could be incorporated along with the cost of misclassification according to its impact on revenue.

TABLE OF TERMINOLOGY

Ξενόγλωσσος όρος	**Ελληνικός Όρος**
Classification	Κατηγοριοποίηση
Ensemble learning	Συνολική μάθηση
Factoring	Αναδοχή επιχειρηματικών απαιτήσεων

ABBREVIATIONS - ACRONYMS

AdaBoost	Adaptive boosting
ANOVA	Analysis of variance
CNN	Convolutional neural network
DBSCAN	Density-based spatial clustering of applications with noise
DT	Decision tree
GA	Genetic algorithm
GMM	Gaussian mixture model
ID	Identity
k-NN	k-Nearest neighbors
LR	Logistic regression
LSTM	Long short-term memory
MLP	Multi-layer perceptron
NB	Naïve Bayes
NN	Neural network
OneR	One rule
PART	Partial decision tree
PCA	Principal component analysis
QDA	Quadratic discriminant analysis
Ref	Reference
RF	Random forest
RMSE	Root-mean-square error
SGD	Stochastic gradient descent
SHAP	Shapley additive explanations
SMS	Short message service
SVC	C-Support Vector Classification
SVM	Support vector machine
UMAP	Uniform Manifold Approximation and Projection
XGBoost	Extreme gradient boosting

REFERENCES

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems", arXiv:1603.04467v2 [cs.DC], 2018.

[2] N. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression", *The American Statistician*, vol. 46, no. 3, pp. 175-185, 1992. Available: 10.1080/00031305.1992.10475879.

[3] A.P. Appel, V. Oliveira, B. Lima, G.L. Malfatti, V.G. de Santana and R. de Paula, "Optimize Cash Collection: Use Machine learning to Predicting Invoice Payment", arXiv:1912.10828v1 [cs.LG], 2019.

[4] A.P. Appel, G.L. Malfatti, R.L. de Freitas Cunha, B. Lima and R. de Paula, "Predicting Account Receivables with Machine Learning", arXiv:2008.07363v1 [cs.LG], 2020.

[5] M. Bahrami, B. Bozkaya and S. Balcisoy, "Using Behavioral Analytics to Predict Customer Invoice Payment", Big Data, vol. 8, no. 1, pp. 25-37, 2020. Available: 10.1089/big.2018.0116.

[6] M.A. Bashar, A.W. Kieren, H. Kerina and R. Nayak, "Propensity-to-Pay: Machine Learning for Estimating Prediction Uncertainty", arXiv:2008.12065v1 [cs.LG], 2020.

[7] B. Boehmke and B. Greenwell, Hands-on machine learning with R.

[8] B. Boser, I. Guyon and V. Vapnik, "A training algorithm for optimal margin classifiers", Proceedings of the fifth annual workshop on Computational learning theory - COLT '92, 1992. Available: 10.1145/130385.130401.

[9] L. Breiman, Machine Learning, vol. 45, no. 1, pp. 5-32, 2001. Available: 10.1023/a:1010933404324.

[10] T. Chen and C. Guestrin, "XGBoost", Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016. Available: 10.1145/2939672.2939785.

[11] M.L.F. Cheong and S.H.I. Wen, "Customer level predictive modeling for accounts receivable to reduce intervention actions", ICDATA' 18, 2018.

[12] W. Cochran, "The X^2 Test of Goodness of Fit", *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 315-345, 1952. Available: 10.1214/aoms/1177729380.

[13] S. Dasgupta, "Experiments with random projection", Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence (UAI'00), San Francisco, CA, USA, pp.143-151, 2000.

[14] "Finance Factoring - IBM Late Payment Histories", Kaggle.com, 2021. [Online]. Available: https://www.kaggle.com/hhenry/finance-factoring-ibm-late-payment-histories.

[15] R.A. Fisher, "The Correlation Between Relatives on the Supposition of Mendelian Inheritance", *Philosophical Transactions of the Royal Society of Edinburgh*, vol. 52, pp. 399-433, 1918.

[16] E. Frank, and H.I. Witten, "Generating Accurate Rule Sets Without Global Optimization", Fifteenth International Conference on Machine Learning, pp. 144-151, 1998.

[17] Y. Freund and R. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119-139, 1997. Available: 10.1006/jcss.1997.1504.

[18] Y. Freund and R. Schapire, Machine Learning, vol. 37, no. 3, pp. 277-296, 1999. Available: 10.1023/a:1007662407062.

[19] R. Gupta and D. Richards, "The History of the Dirichlet and Liouville Distributions", *International Statistical Review / Revue Internationale de Statistique*, vol. 69, no. 3, p. 433, 2001. Available: 10.2307/1403455.

[20] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997. Available: 10.1162/neco.1997.9.8.1735.

[21] J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities.", *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554-2558, 1982. Available: 10.1073/pnas.79.8.2554.

[22] P. Hu, "Predicting and Improving Invoice-to-Cash Collection Through Machine Learning", MSc, Massachusetts Institute of Technology, 2013.

[23] W. Iba and P. Langley, "Induction of One-Level Decision Trees", ML92: Proceedings of the Ninth International Conference on Machine Learning, Aberdeen, Scotland, 1–3 July 1992, San Francisco, CA: Morgan Kaufmann, pp. 233–240, 1992.

[24] B. Kamiński, M. Jakubczyk and P. Szufel, "A framework for sensitivity analysis of decision trees", *Central European Journal of Operations Research*, vol. 26, no. 1, pp. 135-159, 2017. Available: 10.1007/s10100-017-0479-6.

[25] J. Kim and P. Kang, "Late payment prediction models for fair allocation of customer contact lists to call center agents", Decision Support Systems, vol. 85, pp. 84-101, 2016. Available: 10.1016/j.dss.2016.03.002.

[26] J. Kreer, "A question of terminology", *IRE Transactions on Information Theory*, vol. 3, no. 3, pp. 208-208, 1957. Available: 10.1109/tit.1957.1057418.
[27] A. Maćkiewicz and W. Ratajczak, "Principal components analysis (PCA)", *Computers & Geosciences*, vol. 19, no. 3, pp. 303-342, 1993. Available: 10.1016/0098-3004(93)90090-r.
[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, "Scikit-learn: Machine Learning in Python", *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.
[29] G. Rossum and F. Drake, Python 3. United States: SohoBooks, 2009.
[30] S. Lundberg et al., "From local explanations to global understanding with explainable AI for trees", *Nature Machine Intelligence*, vol. 2, no. 1, pp. 56-67, 2020. Available: 10.1038/s42256-019-0138-9.
[31] R. Schapire, "The strength of weak learnability", *Machine Learning*, vol. 5, no. 2, pp. 197-227, 1990. Available: 10.1007/bf00116037.
[32] "sklean-genetic-opt — sklearn genetic opt 0.6.1 documentation", Sklearn-genetic-opt.readthedocs.io, 2021. [Online]. Available: https://sklearn-genetic-opt.readthedocs.io/en/stable/.
[33] A. Subasi and S. Cankurt, "Prediction of default payment of credit card clients using Data Mining Techniques," *International Engineering Conference (IEC)*, 2019, pp. 115-120, doi: 10.1109/IEC47844.2019.8950597.
[34] Taylor, S. J., and B. Letham, "Forecasting at scale", *PeerJ Preprints*, vol. 5, 2017.
[35] A. Tharwat, "Linear vs. quadratic discriminant analysis classifier: a tutorial", *International Journal of Applied Pattern Recognition*, vol. 3, no. 2, p. 145, 2016. Available: 10.1504/ijapr.2016.079050.
[36] D. Wolpert, "Stacked generalization", *Neural Networks*, vol. 5, no. 2, pp. 241-259, 1992. Available: 10.1016/s0893-6080(05)80023-1.
[37] X. Dastile and T. Celik, "Making Deep Learning-Based Predictions for Credit Scoring Explainable", *IEEE Access*, vol. 9, pp. 50426-50440, 2021. Available: 10.1109/access.2021.3068854.
[38] I. Yeh and C. Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients", Expert Systems with Applications, vol. 36, no. 2, pp. 2473-2480, 2009. Available: 10.1016/j.eswa.2007.12.020.
[39] S. Zeng, P. Melville, C. Lang, I. Boier-Martin and C. Murphy, "Using predictive analysis to improve invoice-to-cash collection", Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 08, 2008. Available: 10.1145/1401890.1402014.