Banking Product Recommendation System for Banco Santander

Abstract

With the objective to improve Banco Santander's revenue by introducing an enhanced banking product recommendation system, our project aims to predict customer adoption of the top 3 popular financial products – current accounts, particular accounts, and direct debit. The methods used to predict the customer adoption will include Logistic Regression, KNN, Random Forest, Gradient Boosting, PCA and PCA in 2-Dimensions. Other than the accuracy of the classification outcome, the performance of the models will also be evaluated by the profits generated by the prediction. Our results indicated that the Logistic Regression model with a threshold of 0.1 produced the highest profit of $403,320 for the Current Accounts product. For the Particular Account product, the Gradient Boosting model with a threshold of 0.9 generated the highest profit of $69,610. Similarly, the Gradient Boosting model with a threshold of 0.9 produced the highest profit of $53,280 for the Direct Debit product. Opportunities for further research were identified and include developing additional classification models for: more regions and states of Spain, or the entirety of Spain; the full suite of available banking products; and predictor data using previous 3, 9, 12 and 18-month averages.

Introduction

In all types of business, attracting new customers, existing customer retention, customer up-sale (value add), and customer satisfaction, are all key factors on the customer-facing side of a business to maximize the potential for commercial success and to ensure the continuing operation of a business.

In early 2016, a Spanish bank, Banco Santander (2016), started a Kaggle competition (Kaggle is an online public data platform where data science challenges and competitions are hosted) to search for a better banking product recommendation system. At this time, Banco Santander had 24 different banking products available, and this data science competition was for competitors to develop computational methods of identifying banking products to market to new and existing customers. The input data is a range of bank-specific, financial, and socio-demographic information. As an example, given certain financial and socio-demographic information, I can identify those customers who would purchase our recommended credit card with an accuracy of 80%.

Further to the above, Banco Santander also identified this need to develop a new and better banking product recommendation system because, their current system, as of early 2016, created an uneven customer experience, where: small number of customers receive many recommendations; and a majority of customers receive very little recommendations.

Therefore, the key objective of this data mining and statical learning challenge is to create a more effective product recommendation system for Banco Santander to better understand the needs of the customers, as well as the potential for up-sale and new business, and to enhance the customer experience.

Problem Statement and Research Questions

The purpose of this study is to analyze the Santander bank customer data, which is comprised of customer banking, financial and socio-demographic data (the predictor variables), and current product subscription information (the response variables) for existing customer. The end goal of this study is to develop well-performing classification models to assist Santander bank staff with marketing efforts. To explain further, it is planned that these classification models will be able to flag potential customers for new bank product subscriptions, and then Santander banking staff can utilize this

information to approach existing customers with marketing efforts. It is also anticipated that these classification models will incorrectly flag an amount of existing customers with Type I errors (I.e. false positives: they are not truly interested in a new product, but will be marketed to) or Type II errors (I.e. false negatives: they are truly interested in a new product but will not be marketed to).

Please note that the scope of this study is limited to the development of flagging existing customers to be marketed to. It does not provide any recommendations on the marketing methods and channels that should be utilized to encourage customer sign-up to new products.

Analysis of the banking data will be undertaken using a range of data exploration techniques, domain expertise and knowledge, and a range of machine learning classification model architectures. Testing for a range of model hyperparameters, Monte Carlo Cross Validation and K-fold Cross Validation will be used to tune model hyperparameters to achieve high performance of classification models.

Before the main body of work for this study was undertaken, a preliminary study and project proposal was developed. As part of these preliminary works, research questions were identified and shortlisted. The most critical research questions identified include:

- Given there are 25 predictors, what predictor variables should be used in the final models?
- Given the large amount of qualitative predictors, is there a correlation between customer financial and socio-demographic data and banking products that the customer is actively using?
- Given there are 24 banking products (i.e. response variables), is it necessary to develop a separate model for each banking product? Or is it sufficient and feasible to have one model that works to predict multiple banking products at once?
- Given the likely predictive results can be represented as a confusion matrix, is it possible to evaluate model performance using an analysis of sensitivity and specificity analysis. I.e. where the end results of Type I and Type II errors are considered as costs (e.g. marketing costs, opportunity costs, revenue gained, and so on)?

Data Source, Data Preparation and Exploratory Data Analysis

Dataset Source

The *Santander product recommendation* dataset contains 1 and half years (18 months) of customer bank-specific information, financial and socio-demographic data from Banco Santander (2016). The dataset's period ranges from January 2015 to June 2016. The dataset is very large with 13.64 million data points, 956,645 unique customer ID numbers, and spans all regions and states in Spain. Further information on the full dataset can be found in the following link (Banco Santander, 2016).

Data Preparation and Exploratory Data Analysis

The Data Preparation process for real-world banking customer data proved to be very intensive. It required the application of both domain expertise, exploratory data analysis techniques and knowledge of data types and structures.

As expected with real-world data, there was a lot of information collected manually by bank employees that was redundant or duplicated in some analogous form. As expected with socio-demographic data, it was necessary to introduce many categorical dummy binary variables to address these qualitative and non-ordinal data types. Furthermore, the data size and large extent of regions and states covered in Spain also presented challenges. In addition, the distribution of response predictor values for customers for the different banking products was very sparse and presented

challenges. To elaborate, of the 24 banking products available, customers only tend to have 1-2 products in general, and this presented challenges when attempting to predict which banking products to market to customers.

The raw dataset contains 24 predictor variables and 24 response variables, for 13,647,309 data points over 18 months at monthly intervals. Full details of the variable descriptions can be found in the following link (Santander, 2016). **Please note that the descriptions of the variables selected from data preparation and exploratory data analysis can be found further below.*

Dataset variables were reviewed by the panel of team members with financial background knowledge application. In addition to knowledge and domain expertise application, preliminary Exploratory Data Analysis of variable occurrence and datatypes was used to eliminate 8 redundant predictor variables and 3 key response variables were selected to focus predictive classification efforts on.

To address categorical variables and missing variables in the predictor variables, 13 dummy variables were introduced (and their original categorical variables deleted). In some circumstances it was possible to infer missing variable values as zero values. But in general, no imputation was utilized as missing values were either zero or had a dummy value implemented. This data cleaning step brought the total number of predictor variables back to 29.

Due to the size of large dataset, memory allocation issues and very long run times by ensemble algorithms on this very large dataset (due to high time complexity) were an issue. Therefore, the panel of team members limited the scope of predictive efforts to focus on the region of 'Barcelona' or 'Madrid', the two largest cities in Spain. If the scope and availability of time allowed for further analysis later, more regions in Spain could be added to the dataset. Ultimately, 'Barcelona' was selected as the final subject for this study. This reduced the dataset to 1,271,771 data points (approx. 9.32% of the original dataset size). Furthermore, due to the large amount of response variables, the panel of team members also limited the scope of predictive efforts to the 3 most popular banking products at early stages in the study. If the scope and availability of time allowed for further analysis later, more banking products could be added to the dataset for analysis. Ultimately, the three most popular products were the final selection for this study.

After the data cleaning process described above was completed, the preliminary variables selected for further analysis and model prediction are described in **Table 1**, the banking products selected for predictive modeling (I.e. the 3 most popular products/response variables), and in **Table 2**, the banking customer characteristics used as model inputs (I.e 14 predictor variables). **Please note that the distributions of categorical values are shown in their original format.*

Table 1. Preliminary Variable Selection – Responses (Banking Products)

Variable	Description	Distribution
cco	Bank product of Current Account	65.55% occurrence in dataset. <Binary>
ctop	Bank product of Particular Account	12.91% occurrence in dataset. <Binary>
recibo	Bank product of Direct Debit	12.72% occurrence in dataset. <Binary>

Table 2. Preliminary Variable Selection – Predictors (Customer Characteristics)

Variable	Description
ind_empleado	Bank employee flag *(N: no ; A: yes ; B: former employee)*. <Categorical 3 Levels>
sexo	Gender *(V: female ; H: male ; not specified -> dummy)*. <Binary + Dummy>
ind_nuevo	It is a flag that shows if a customer joined in the last 6 months. <Binary>
indrel	Binary variable indicating if a customer left during the current month <Binary, except 99 = 0>

indrel_1mes	Customer type *(1: primary, 2: co-sign, 3: potential, 4: former, 5: former co-sign)* <Categorical 5 Levels>
indresi	Customer is a resident of Spain index. <Binary>
indext	Customer is a foreigner index. <Binary>
conyuemp	To indicate if the customer is a spouse of bank employee. <Binary>
tipodom	Index of the customer primary address *(1: primary address used)*. <Binary>
ind_actividad_clientele	Banking activity of customer *(1: active, 0: not active)* <Binary>
segmento	It shows the customer background. <Categorical 3 Levels>
age	The customer's age (years) <Integer>
antiguedad	The time customer has been with Santander bank (in months) <Integer>
renta	Annual household income <Float>

Data Exploration - Distribution Analysis

Exploratory data analysis began with review of the distributions and frequency of the 14 predictor variables for the Barcelona subset of data. Bar charts for binary variables and histogram plots for interval data were produced and can be seen below (refer **Figure 1**). A summary of findings revealed by interpretation of the exploratory data analysis of variable distributions can be seen further below in **Table 3**.

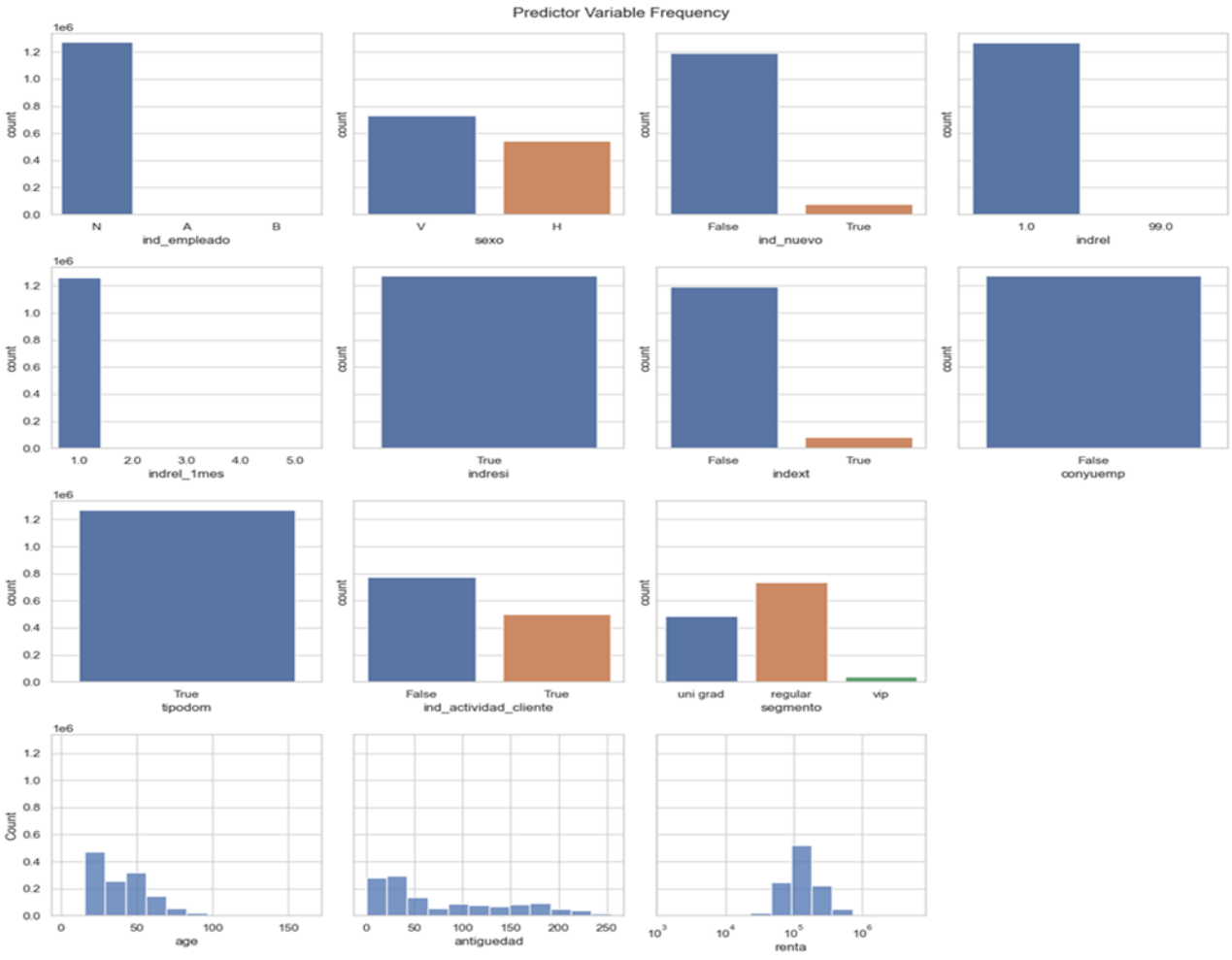


Figure 1. Preliminary Variable Selection – Responses (Banking Products)

Table 3. Data Exploration - Distribution Analysis Findings

Variable	Data Exploratory Analysis Comments
ind_empleado	All customers are non-employees of the bank.
sexo	There is a 57% to 43% skew of female to male bank account holders.
ind_nuevo	Only ~10% of customers joined in the last 6 months, all others have been customers for longer periods. Refer to **antiguedad** for further insights.
indrel	Every customer listed is not a customer who left at the beginning of the month. This possibly implies that when a customer leaves, they get de-listed, rather than have this data variable flag against them.
indrel_1mes	Every customer is a first/primary customer. There are no co-signatories, potential or former customers flagged.
indresi	All customers are residents of Spain.
indext	Some individuals (~15%) were born in a country other than Spain. However, most (~85%) were born in Spain.
conyuemp	No customers are spouses of bank employees.
tipodom	All customers are registered using their primary addresses (but how would the bank realistically know this. It only means that the customer isn't registering with an address the bank has flagged as an investment property through their loans).
ind_actividad_clientele	~60% of customers are not very active in the interactions with services at the bank.
segmento	More than 1/3 of customers are university graduates, and the number of VIPs is small (as expected). VIPs are most likely high net worth or high-income individuals.
age	The customer base is skewed towards younger people. Especially, young adults (and likely university graduates). Most customers are 18 to 60 years old, which makes logical sense in terms of natural human life expectancy and survivorship.
antiguedad	Many customers are very new (less than 5 years or 60 months), but then it approaches a uniform distribution after the 5-year mark. This appears logical, there is likely high recruitment of customers caused through initial interest, referrals, promotions and so on, and then people decide if they want to keep their account after evaluating the bank after a medium length of time (5 years). Customer retention follows what appears to be a fairly uniform distribution.
renta	Household incomes can start at 0, but the main distribution begins at the 40k-69k, with a reasonable proportion in the 70k-99k bracket, and with most customers falling in the 100k-129k bracket, a similar proportion to the 70k-99k bracket earns a household income of 130-159k bracket, and then the distribution trials off towards 1M incomes. The distribution appears normal on a log-scale.

As a result of this analysis, it would be expected that **ind_empleado**, **indrel**, **indrel_1mes**, **indresi**, **conyuemp**, and **tipodom** would be removed by stepwise regression and domain knowledge application as they offer no variability in describing the characteristics of customers. Furthermore, **renta** (household income) maybe benefit from being scaled on a log-scale or a normal kernel transformation at the modelling stages. For the purposes of the following correlation analysis, these redundant predictor variables are removed as they offer no variability in describing the characteristics of customers. This brings the total number of predictor variables down to 14.

Data Exploration - Correlation Analysis

Exploratory data analysis continued with a review of correlation coefficients between the 14 predictor variables narrowed down from the previous distribution analysis and the three response variables for the Barcelona dataset. Correlation coefficients are indicators of the strength of the linear relationship between variables. Therefore, a correlation analysis using a correlation matrix is a highly useful tool

for undertaking preliminary data analysis prior to final variable selection and model architecture selection. **Figure 2** depicts the strength of correlation coefficients between the 14 predictor variables and 3 binary response variables, where a light is a strong positive correlation and dark is a strong negative (or inverse) correlation. A summary of findings revealed by interpretation of the exploratory data analysis of variable correlation coefficients can be seen further below in **Table 4**.

As stated, the total number of predictor variables is 29, many of which are dummy binary variables created by the data preparation process. As can be seen below in **Figure 2**, a vast majority of the intra-correlation amongst variables is negligible, which suggests most variables are each explaining unique variability in the descriptive characteristics of the bank customers (in terms of linear relationships). Noting the low predictor variable correlations with the 3 final response variables, it is worth using stepwise regression or some alternative machine learning algorithm to computationally select the best response variables for use as model inputs.

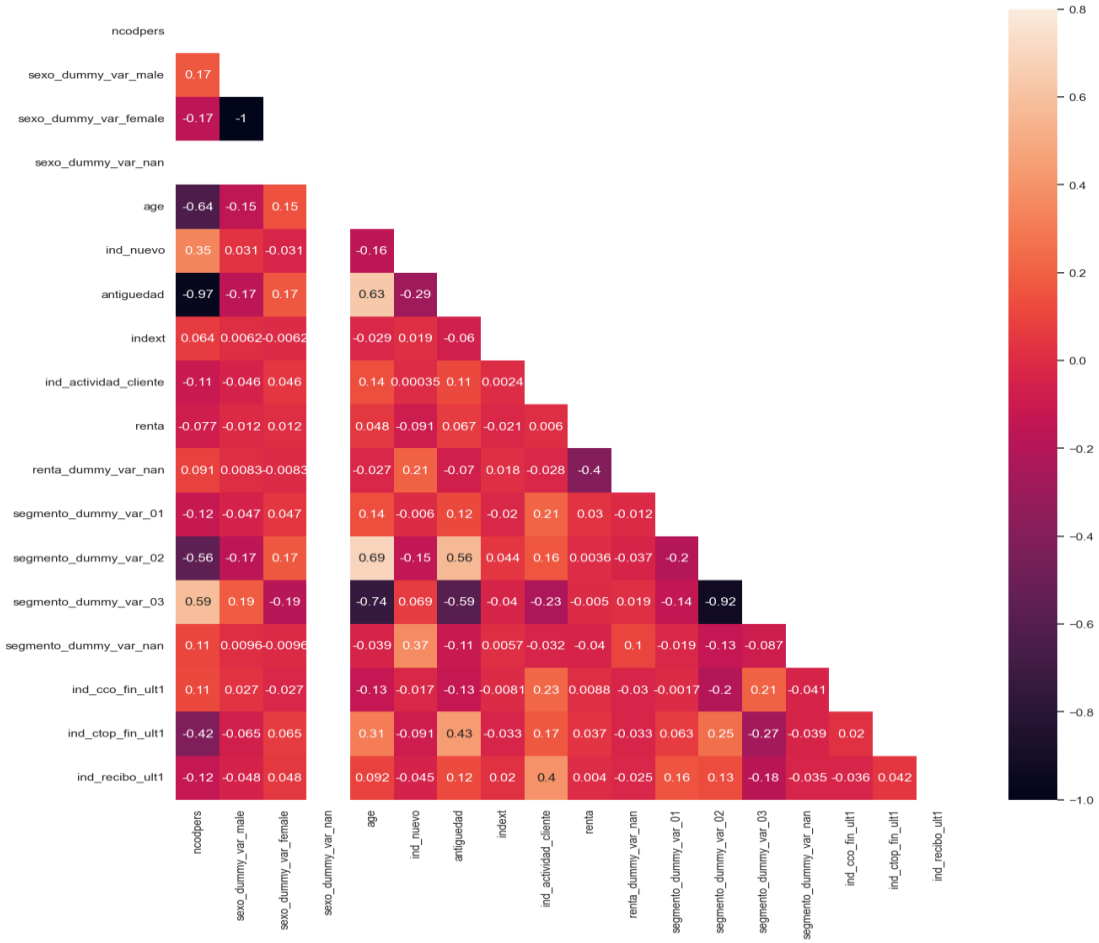


Figure 2. Correlation Matrix - Customer Characteristics and 3 Banking Products

Table 4. Data Exploration - Correlation Analysis Findings

Variable	Data Exploratory Analysis Comments
ncdopers (customer ID)	-0.97 correlation with **antiguedad** (customer seniority). -0.64 **age**, and –0.56 **segmento 02** (regular customer) i.e. the older and longer in time as a customer, the customer is, the earlier/lower value is for their customer ID, which is very logical. If someone is a regular customer, the more likely their customer ID will be lower/earlier.

sexo (gender)	Female is directly inversely-correlated with Male (-1) i.e. all non-females are males, unless 'unknown'.
age	+0.63 **antiguedad** (customer seniority) i.e. the older a customer is, the more likely they are to have held an account for a longer period. +0.69 correlation with being a regular customer (**segmento 02**), -0.74 correlation with being a university graduate (**segmento 03**) i.e. the older someone is, the more likely they are a regular customer; the younger someone is, the more likely they are a university graduate. Perhaps this phenomenon can be explained by the growing prevalence of university education in younger generations.
antiguedad (length of time as customer)	+0.56 **segmento 02** (regular customer), -0.59 **segmento 03** (university graduate) i.e. the longer someone has held an account, the more likely they are a regular customer; the younger someone is, the more likely they are a university graduate. Perhaps this phenomenon can be explained by the growing prevalence of university education in younger generations.
segmento (02) (regular customer)	Strongly inversely-correlated with university graduates -0.92 i.e. regular individuals holding accounts are very likely not university graduates.
ctop (particular account)	**age** +0.31, **antiguedad** +0.43 i.e. the particular account product is more likely to be held by older customers who have had their accounts for longer.
recibo (direct debit)	+0.4 **ind actividad clientele** (active customer) i.e. the direct debit product is more likely to be held by an active customer.

Proposed Methodology

There are a few different machine learning models that are explored and compared for this project. All models are trained on the training dataset (80% of the full dataset) and we used the testing dataset (20% of the full dataset) to predict the response. Please note that we use the most recent month or the average of the previous 6-month period of customer predictor variable data to build, train the model and run predictions on the most recent month of response variable data.

Furthermore, due to the sparsity of current banking products in the dataset (customers only tend to have 1-2 banking products in general) all machine models predict only one of the three response variables (Current Account, Particular Account or Direct Debit) at any given time. There are no multiple-predictor model architectures explored in this study due to the lack of relationship between many of banking products (i.e. a very sparse response variable matrix).

Data Preparation and Exploratory Data Analysis

The methodology for data preparation and exploratory data analysis can be seen in section **Data Source, Data Preparation and Exploratory Data Analysis**. It can be summarized as the following steps:

1. Preliminary variable selection, removal of redundant predictor variables using finance, consulting and data science domain knowledge.
2. Addressing missing values (NaNs) through zero values or dummy variables, or deletion
3. Addressing qualitative and non-ordinal categorical variables, using the introduction of dummy binary variables
4. Exploratory Data Analysis of Predictor variables and Response variables – Distribution
5. Exploratory Data Analysis of Predictor variables and Response variables – Correlation
6. Intermediate variable selection, removal of redundant predictor variables using knowledge gained from EDA.

For further discussion on the Data Cleaning and Exploratory Data Analysis process, please refer to sections **Data Source, Data Preparation and Exploratory Data Analysis** and **Lessons Learned**. *Please note that final variable selection can be unique to each machine learning model architecture below.

Logistic Regression and KNN

The objective is to build a Logistic Regression and KNN model to determine if a customer would purchase a financial product.

List of independent variables used are Employment Index *(A: active, B: ex-employed)*, Age, Sex, Seniority Index, Active Index *(1: active, 0: not active),* Customer type, Residency Status and Annual Household Income.

List of Dependent variables used are Ownership of Current Account, Particular Account or Direct Debit.

The steps taken for the Logistic Regression:

- Randomly split data in train and test dataset (0.8/0.2).
- Train the model on training dataset with stepwise glm to select the best independent variables by minimizing AIC.
- Predict three binary response variables separately on testing dataset.
- Set different threshold cut-off percentage (0.1, 0.2, 0.3, 0.5) for binary predicting results.
- Produce Confusion Matrix and compare test results.

The steps taken for the KNN:

- Randomly split data in train and test dataset (0.8/0.2).
- Train the model on training dataset with KNN model.
- Predict three binary response variables separately on testing dataset.

Random Forest and Gradient Boosting

The objective is to build a random forest and boosting model to determine if a customer would purchase a financial product. The random forest model will be tuned based on the number of variables, the number of trees and the node size to see which parameters would give the lowest training error. While the gradient boosting model will be built in a similar way where an initial model will be built and the model will be tuned based on the number of trees, the shrinkage (learning rate) and the depth (maximum number of leaves).

List of independent variables used are Employment Index *(A: active, B: ex-employed)*, Age, Sex, Seniority Index, Active Index *(1: active, 0: not active),* Customer type, Residency Status and Annual Household Income.

List of Dependent variables used are Ownership of Current Account, Particular Account or Direct Debit.

The steps taken for the Random Forest and Gradient Boosting:

- Train a default model for both the random forest and gradient boosting algorithms.
- Tune the model based on the training data to improve the performance.
    - Random forest: Number of variables, number of trees and node size.
    - Gradient boosting: Number of trees, shrinkage and depth.
- Select the best parameters based on the training error and apply it to the testing model.

KNN (k-nearest neighbours) with PCA (principal component analysis)

The objective of this method is to to classify if customer of the Santander Bank is a good candidate for Current Account *(ind_cco_fin_ult1)*, Particular Account *(ind_ctop_fin_ult1)*, and Direct Debit *(ind_recibo_ult1)* product by building a Principal Component Analysis (PCA) model and use the recommended principal component output as the final dataset for the KNN model. We would then use the KNN to run prediction on the testing dataset.

The list of independent variables used for this method are Employment Index *(A: active, B: ex-employed)*, Age, Sex, Seniority Index, Active Index *(1: active, 0: not active),* Customer type, Residency Status and Annual Household Income. Whilst the list of dependent variables used are ownership of Current Account, Particular Account, and Direct Debit product (*1: own the product, 0: does not own the product*).

The steps taken for the KNN with PCA method are:

- Build the PCA regression model by fitting all predictor variables.
- Plot the summary of the model to decide the number of principal components to use for the KNN model.
    - We select 10 principal components as the below plot shows adding more than 10 principal components does not increase the percentage of explained variance by much.

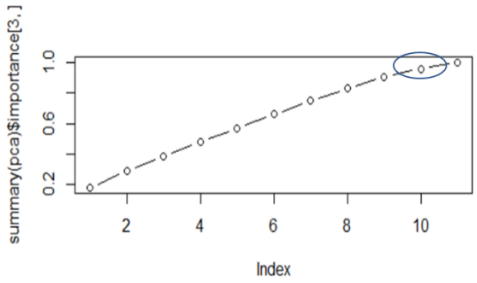


Figure 3 – PCA Summary Plot

- The below plot shows the interpretation of PCA objects by its categorical response value and eigenvectors for Direct Debit *(ind_recibo_ult1)*. The plots for Current Accounts *(ind_cco_fin_ult1)* and Particular Account *(ind_ctop_fin_ult1)* are available in the Appendix.

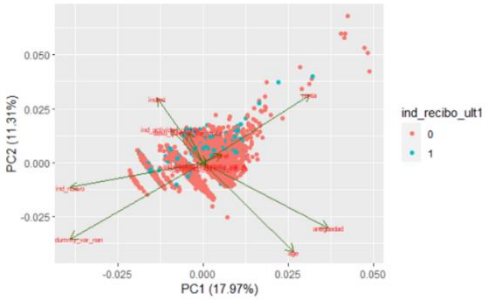


Figure 4 – PCA Objects Interpretation

- Based on the above plot, we created a new data frame with the selected 10 principal components.
- We would then build K-Nearest Neighbors model where we fit the new dataset.

- In building the KNN model, we fit different values of K: 1, 3, 5, 7, 9, 11, 13, and 15 to find the lowest testing error.
  - The below plot shows the model built for Direct Debit *(ind_recibo_ult1)* has the lowest testing error, *0.1249136, when* K = 15. The same plots for Current Accounts and Particular Account are available in the Appendix.

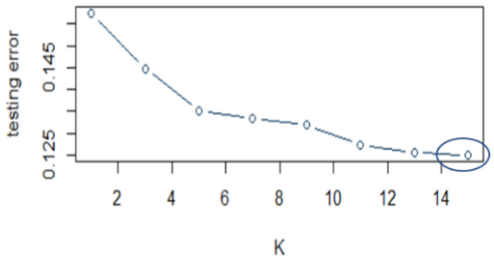


Figure 5 – Testing Error by number of K

- Run the final prediction on the testing dataset by using the best value of K value *(i.e. K = 15 for Direct Debit)*.
- To further asses the robustness of this method, we ran Monte Carlo cross-validation:
  - Randomly select 80% observations from the full dataset as the training dataset and the rest as the testing dataset.
  - Train models for each response variables and calculate the CV testing error.
  - Repeat the process 100 times and calculate the average of the testing error.

Decision Boundary Visualization - PCA 2-Dimension Approximations

For the purposes of visualizing the decision boundaries of the above model architectures, 2-dimension approximations of the model architectures will be modeled. To achieve this, the following steps will be taken:

- The predictor training and testing data will be projected into the first two principal directions of a Principal Component Analysis (PCA) i.e. the predictor variables will be combined through PCA dimensionality reduction into 2 dimensions. This also requires pre-processing scaling of any non-binary variables onto a 0 to 1 scale.
- Scatterplots of the PCA 2-dimensional data will be explored to understand if the 2 principal directions make logical sense in terms of the original variables.
- Logistic Regression, K-Nearest Neighbor, Random Forest, AdaBoost and Gradient Boosting models will be trained on the projected predictor training data and a single response variable data using hyperparameters obtained from previous tuning (the other methods above) or by taking advantage of multi-processing sklearn's GridSearchCV in Python.
- The models trained on the 2-dimensional predictor data will then predict the response label (0 or 1) over a grid at equal intervals e.g. from –6 to 6 in the X-direction and –8 to 16 in the Y-direction, at 100 intervals each, which results in 10,000 equally-spaced data points.
- Each model's grid predictions will then be compared to the true labels for each of the three predictor variables I.e. there will be 15 2D-approximated model visualizations (5 model architecture X 3 predictors). These can be viewed at the end of the **Appendix**.

Results

Profit Calculation

To determine which model performs the best for every selected response variable, we will use cost/profit analysis. Before we get into the results for each model, we will start with the overview of the cost/profit calculation as explained below.

- Dealing with a financial marketing task, it is necessary to penalize potential lost revenue (through opportunity costs) and calculate potential profits in order to evaluate model performance. Model accuracy/test error/misclassification rate is not the sole indicator of model performance in this application.
- **Calculation of Profit**: With the models in place, the estimated profit will be calculated to determine thresholds for each model to determine if a customer should be selected to be advertised to.
- **True Positive** – This customer will purchase the product and will be assigned a **revenue of $50** (composed of +$60 profit and -$10 advertising cost).
- **True Negative** – This customer will not purchase the product and will not be advertised to, thus **no revenue or cost** is assigned.
- False Positive – The customer is predicted not to purchase but ended up not doing so, an advertising **cost of $10** (i.e. -$10 revenue) will be assigned.
- False Negative – The customer is predicted not to purchase a product but has the intention of purchasing, this will be assigned a **cost of $60** (i.e. -$60 revenue, the opportunity cost).

Logistic Regression and KNN

- Current Accounts

Test Error is smallest at 43% when Threshold = 0.5. However, threshold = 0.1 has the smallest FP, suggesting the best prediction for the Current Accounts customer. After stepwise analysis, the best selected variables are ind_impleado_dummy_var_B, sexo_dummy_var_male, age, ind_nuevo, indrel, indrel_1mes_dummy_var_1, indext, ind_actividad_cliente, renta, renta_dummy_var_nan, segmentom_dummy_var_01 , segmentom_dummy_var_02, segmento_dummy_var_03.

	Threshold	Test Error	TN	FN	TP	FP	Profit
Logistic Regression	0.1	0.4523	58	2	9,676	8,036	403,320
	0.2	0.4481	143	13	9,665	7,951	402,960
	0.3	0.3953	1,570	502	9,176	6,524	363,440
	0.5	0.3149	4,282	1,784	7,894	3,812	249,540

Test Error is smallest at 44% when K = 9.

	K	Test Error	TN	FN	TP	FP	Profit
KNN	1	0.4436	4,079	3,868	5,810	4,015	18,270
	2	0.4500	3,991	3,895	5,783	4,103	14,420
	3	0.4443	3,903	3,705	5,973	4,191	34,440
	9	0.4375	3,610	3,291	6,387	4,484	77,050

- Particular Accounts

Test Error is smallest at 9% When Threshold = 0.5. However, threshold = 0.1 has the smallest FP, suggesting the best prediction for the Particular Accounts customer. After stepwise analysis, the best

selected variables are ind_impleado_dummy_var_A, ind_impleado_dummy_var_B, sexo_dummy_var_male, age, ind_nuevo, antiguedad, indrel_1mes_dummy_var_3, indrel_1mes_dummy_var_nan, ind_actividad_cliente, renta, segmentom_dummy_var_01, segmentom_dummy_var_02, segmento_dummy_var_03.

	Threshold	Test Error	TN	FN	TP	FP	Profit
Logistic Regression	0.1	0.2865	11,049	19	1,631	5,073	29,680
	0.2	0.1733	13,309	266	1,384	2,813	25,110
	0.3	0.1167	15,080	1,032	618	1,042	-41,440
	0.5	0.0928	16,122	1,650	0	0	-99,000

Test Error is smallest at 10% when K = 9.

	K	Test Error	TN	FN	TP	FP	Profit
KNN	1	0.1224	15,039	1,092	558	1,083	-48,450
	2	0.1271	15,005	1,142	508	1,117	-54,290
	3	0.1107	15,438	1,283	367	684	-65,470
	9	0.1009	15,840	1,511	139	282	-86,530

- Direct Debit

Test Error is smallest at 9% When Threshold = 0.5. However, threshold = 0.1 has the smallest FP, suggesting the best prediction for the Direct Debit customer. After stepwise analysis, the best selected variables are ind_impleado_dummy_var_B, sexo_dummy_var_male, age, ind_nuevo, antiguedad,indrel_1mes_dummy_var_1,indext,ind_actividad_cliente,renta, renta_dummy_var_nan, segmentom_dummy_var_01, segmento_dummy_var_03.

	Threshold	Test Error	TN	FN	TP	FP	Profit
Logistic Regression	0.1	0.2673	11,448	4	1,574	4,746	31,000
	0.2	0.2537	11,787	101	1,477	4,407	23,720
	0.3	0.0996	15,830	1,406	172	364	-79,400
	0.5	0.0888	16,194	1,578	0	0	-94,680

Test Error is smallest at 9% when K = 9.

	K	Test Error	TN	FN	TP	FP	Profit
KNN	1	0.1479	14,927	1,362	216	1,267	-83,590
	2	0.1483	14,937	1,379	199	1,257	-85,360
	3	0.1065	15,793	1,492	86	401	-89,230
	9	0.0901	16,161	1,568	10	33	-93,910

Random Forest

For each of the random forest models, for each financial product, is tuned over the number of variables, the number of trees and the node size. The following are the iterations used for each tuning parameter:

1. Number of variables (1 to 11)
2. Number of trees (50, 100, 200, 300, 400)
3. Node Size (1, 2, 3, 4, 5)

Final Tuned Model for each Product

Product	Number of Variables	Number of Trees	Node Size	Training Error	Testing Error
Current Account	4	50	1	0.277	0.284
Particular Account	4	500	5	0.118	0.129

Direct Debit	2	400	5	0.118	0.120

Once the final model has been chosen, a confusion matrix is created to calculate the profit attained by using a particular model. The thresholds, that determine the values of the confusion matrix, are adjusted to find the threshold that would provide the most profitable model.

Final Threshold for each Product

Product	Threshold	True Positive	False Positive	True Negative	False Negative	Profit
Current Account	0.9	6,828	5,044	4,550	1,350	209,960
Particular Account	0.9	1,771	1,421	13,869	711	31,680
Direct Debit	0.99	790	1,346	14,294	1,343	-54,540

Gradient Boosting

For each of the gradient boosting models, for each financial product, is tuned over the number of trees, shrinkage and depth. The following are the iterations used for each tuning parameter:

1. Number of trees (50, 100, 200, 300, 400)
2. Shrinkage (0.001, 0.01, 0.1, 1)
3. Depth (1, 2, 3, 4, 5)

Final Tuned Model

Product	Number of Trees	Shrinkage	Depth	Training Error	Testing Error
Current Account	300	1	4	0.275	0.307
Particular Account	400	0.1	5	0.126	0.131
Direct Debit	400	0.01	5	0.114	0.120

Once the final model has been chosen, a confusion matrix is created to calculate the profit attained by using a particular model. The thresholds, that determine the values of the confusion matrix, are adjusted to find the threshold that would provide the most profitable model.

Final Threshold for each Product

Product	Threshold	True Positive	False Positive	True Negative	False Negative	Profit
Current Account	0.9	7,939	8,380	1,214	239	298,810
Particular Account	0.9	2,724	4,145	10,184	419	69,610
Direct Debit	0.9	2,090	4,870	10,770	42	53,280

KNN with PCA

The table below shows the result for the KNN with PCA model for all banking products with its profit calculation.

Product	Best K	Testing Error	CV Testing Error	TP	FP	TN	FN	Profit ($)
Current Account	15	0.2994964	0.3020196	3077	1583	4017	1450	$51,020
Particular Account	13	0.1367631	0.1414724	483	1008	8259	377	-$8,550
Direct Debit	15	0.1246174	0.1254296	145	1070	8720	192	-$14,970

Note: TP = True positive; FP = False Positive; TN = True Negative; FN = False Negative.

The confusion matrix for each banking product is available in the Appendix.

Decision Boundary Visualization - PCA 2-Dimension Approximations

A thorough analysis of the two principal directions of the dataset using the 14 selected predictor variables can be seen towards the end of the **Appendix**. This analysis, which is summarized below in

Figure 6, shows that the 1st principal direction (the X-dimension) corresponds to **age** (where older is to the right), **antiguedad** (customer seniority, where longer is to the right), and **segmento** (where university graduate is more left and regular customer is more right). Further, the 2nd principal direction (the Y-dimension) corresponds mostly to **renta** (household income, where higher is more to the top) and **indrel** (recent customer, where negative values are more likely to be a recent customer). If one compresses the Y-dimension to 1D it would look very similar to the distribution of household income identified in **Exploratory Data Analysis**, where most households earn $40k-$160k, but there are some that earn $1M incomes. These $1M incomes are represented by the sparse data points towards the top of the PCA 2-D plot. Further, it should be noted that VIPs are a small percentage of sample population and are generally negligible in the PCA analysis. But it can be inferred from the analysis that VIPs tend to be older and have lower household income. It is possible that VIPs are retired and have high net worths, but low reported household income due to being business owners or having other financial vehicles to receive income from. It stands to reason that high net worth individuals may be utilizing financial vehicles to reduce their income and thereby income tax. Therefore, dimensionality reduction of the predictor variables into PCA 2-Dimensions is a reasonable choice for visualization purposes.

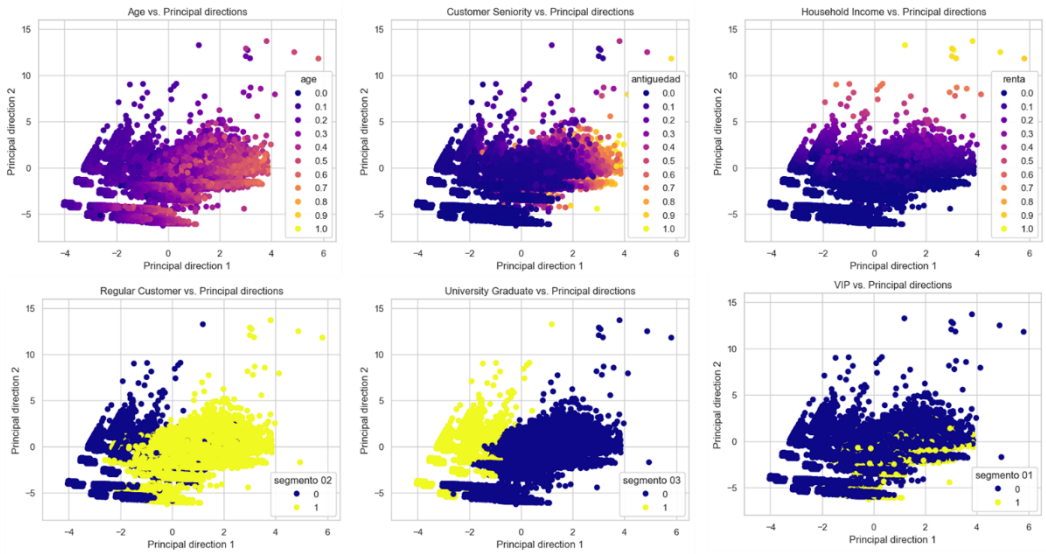


Figure 6. PCA 2-Dimensional Approximation of Predictor Variables

PCA 2-dimension visualizations of the decision boundaries for Logistic Regression, K-Nearest Neighbor, Random Forest, AdaBoost and Gradient Boosting models for each of the three selected predictors (the three most popular banking products) can be found at the end of the **Appendix**. For brevity, only the 2-dimension decision boundary approximations of the most successful model architecture for each of the three predictors will be discussed.

Firstly, for the Current Accounts product (**cco**), the Logistic Regression model with threshold 0.1 has the highest profit of $403,320. A 2-d approximation of its decision boundary can be viewed in **Figure 7** below. Of particular note, Logistic Regression has a linear decision boundary and is easy to interpret. It is assuming that people who are younger and university graduates or those of high household income are most likely to desire a Current Account. This appears to make logical sense since younger people and those with high incomes may be looking for new banking opportunities.

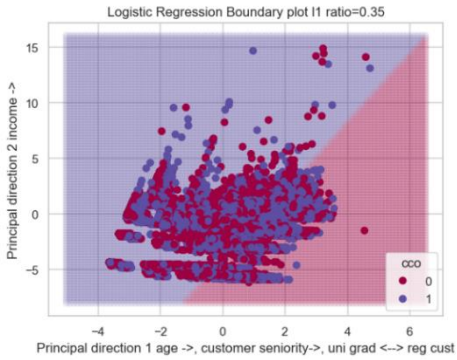


Figure 7. PCA 2-D Approximation Decision Boundary for cco Logistic Regression 'best' Model

Secondly, for the Particular Account product (**ctop**), the Gradient Boosting model with threshold 0.9 has the highest profit of $69,610. A 2-d approximation of its decision boundary can be viewed in **Figure 8** below. Of note, boosting algorithms segment the solution space in rectangular boxes and regions (in the 2D case). This Gradient Boosting model is able to create isolated boundaries to capture the true labels that are nested inside the right-hand side of the sea of false labels. The section of true labels can be characterized by customers who are older and have typical household incomes (perhaps within 1 standard deviation of the mean if we consider the distribution of incomes to be normal as identified in **Exploratory Data Analysis**).

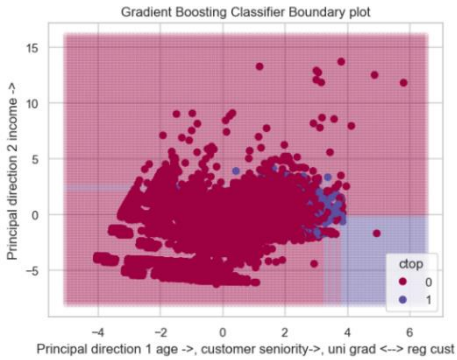


Figure 8. PCA 2-D Approximation Decision Boundary for ctop Gradient Boosting 'best' Model

Finally, for the Direct Debit product (**recibo**), the Gradient Boosting model with threshold 0.9 has the highest profit of $53,280. A 2-d approximation of its decision boundary can be viewed in **Figure 9** below. In this particular visualization, the 2D approximation of the best Gradient Boosting model for this predictor is somewhat poor and it assumes most points are a false label. Even with decreased opacity, it is obvious there are no clear rectangular sections of true labels. This may be because the true labels are so well-mixed together with false labels in this 2D projected solution space.

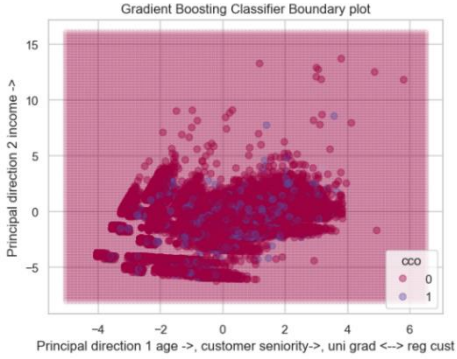


Figure 9. PCA 2-D Approximation Decision Boundary for recibo Gradient Boosting 'best' Model

Cross Validation

Cross validation was performed on the final models of each product. The results show us that the models are robust as there is little difference between the initial testing error and the one with cross validation. The final models are not overfitted to the training data and are not sensitive to different resamples of the available dataset.

Product	Testing Error	Testing Error (CV)
Current Account	0.4523	0.4531
Particular Account	0.131	0.132
Direct Debit	0.120	0.122

Conclusions

Kindly find the conclusion based on the results and explanation in the above sections:

- For Current Accounts product, the Logistic Regression model with a threshold 0.1 has the highest profit of $403,320.
- For Particular Account product, the Gradient Boosting model with a threshold 0.9 has the highest profit of $69,610.
- For Direct Debit product, the Gradient Boosting model with a threshold 0.9 has the highest profit of $53,280.

Further Research

As mentioned in earlier stages of data exploration, the scope of this study was limited to the Barcelona region of Spain and for the three most popular banking products. Furthermore, predictive modelling efforts were predominantly based on data from the most recent month or previous 6-month averages to predict labels for the most recent month. Therefore, the following opportunities for further research are:

- Develop additional classification models for more regions and states of Spain, or the entirety of Spain. As stated, the scope of this study was limited to the Barcelona region.
- Develop additional classification models for the full suite of available banking products. As stated, the scope of this study was limited to the three most popular banking products. There are 24 available banking products, therefore it is possible to develop additional classification models for the current dataset, or new datasets where there are even more available banking products recommendation.
- Develop a range of classification models for different X-month averages. The models in this study are limited to previous 6-month averages for predictor variables. It stands to reason that additional prediction models should be developed and compared for 3-month, 6-month (already developed), 9-month, 12-month and 18-month averages. These intervals are commonplace in trend analysis and the financial sector.

Lessons Learned

This section lists the lessons that we learned from this project. We break the lessons learned into two categories: Exploratory Data Analysis and Modelling.

Exploratory Data Analysis

- The Data Cleaning process for real-world banking customer data proved to be very intensive. It required the application of both domain expertise and knowledge of data types and structures. As expected with real-world data, there was a lot of information collected manually by bank employees that was redundant or duplicated in some analogue form. In addition, the response labels for customers for different products was very sparse and presented difficulties. To explain, of the 24 banking products available, customers only tend to have 1-2 products in general, and this presented challenges when attempting to predict other products to promote to customers.
- Dealing with large amounts of predictor variables. Voted in a group based on research and domain knowledge to remove redundant predictor variables.
- Dealing with large amounts of response variables. Voted in a group based on research and domain knowledge to select most popular responses: savings account, credit card and mortgage.
- Dealing with a very large data size: 13,647,309 rows x 48 columns of various datatypes.
    - Attempting to pass large datasets to algorithms in R threw memory allocation issues e.g. *"cannot allocate 2.4gb of memory"* and so on.
    - Needed to limit dataset from whole of Spain to one region. Voted as a group for 'Barcelona', as the capital of the Catalan area of Spain. 1,271,771 rows (approximately 9.3188% of the total dataset).
- Dealing with time series data. Some models deemed infeasible.
    - Due to memory allocation issues and difficulty of using Autoregressive integrated moving average (ARIMA) or Exponential-smoothing/Cumulative Sum (CUSUM) -like models, it was decided by the group to focus on 'one-pass' model architectures, rather than 'moving average', 'sequential cumulative sum' or other recursive model architectures such as reinforcement learning.
    - Voted as a group to use: the last month of data as response values; and last month of data or average previous 3 months or average previous 6 months or average previous 12 months as predictor values. This allowed us to appropriately train and test many of the different types of one-pass model architectures covered as part of the course.
- In general, of the 24 response variables, their occurrence in the dataset of 13.6 million data points, is incredibly sparse. That is, customers and previous customers only have 1 to 2 products at any given time period. Therefore, it was infeasible to pursue researching how to create a product recommendation algorithm that uses the occurrence of existing products amongst similar customers to make recommendations.
- Low frequency of positive occurrence (response = 1) for the response variables: savings account, credit card and mortgage; causes poor performance of predictive classification models. In fact, a model simply guessing that all data points are 0 will result in very attractive and accurate performance metrics. Therefore, it was necessary to switch predictive classification models to the top 3 response variables by occurrence.
    - All percentages are by occurrence in the entire dataset Before: '**ahor**' Savings Account – 0.0102%, '**tjcr**' Credit Card – 4.4389%, '**hip**' Mortgage – 0.5887% , Current: '**cco**' Current Account – 65.5484%, '**ctop**' Particular Account – 12.9008%, '**recibo**' Direct Debit – 12.7916%.
- After a reduction of 8 redundant predictor variables, dropping the total from 24 to 16 predictor variables, there was a need to a create various dummy binary variables in-place of each level categorical variable and any instance of missing values ('nan' (not a number)) which

could not be rationalized to a zero-value. For example, a missing value of 'household income' cannot be rationalized as a zero-value, it is simply 'unknown' and thus should have a dummy binary variable. On the other hand, a missing categorical value of someone being the spouse of an employee can be rationalized as a zero-value: it is safe to assume that this person is not a spouse of an employee.

- With the creation of many dummy binary variables, this brought the total number of predictor variables back to 29 variables. Of which, only only three variables 'age', 'antiguedad' (months as a customer) and 'renta' (household income) have a range of variability. All other predictor variables are binary variables.
- Difficulties with datatypes input and output when working across Python and R. For example, boolean 'True' and 'False' labels for boolean values from Python being read in as 'char' or 'string' in R.

Modeling

- The time complexity of running ensemble methods through large CV (resamples = 100+) numbers on a large dataset (1.3M entries) and performing Grid Search on a range of hyperparameters values in order to find 'optimal' hyperparameter values is too high. Even with a powerful performance desktop PC (not a laptop as some other group members have), runtime is more than 1 hour. In some cases, it was possible to take advantage of Scikit-Learn's GridSearchCV and multi-processing. But in general, performing a search over a large variation of hyperparameters and large CV numbers for a large dataset was simply infeasible.
- It is important for data analysts and/or data scientists to perform cross-validation or other methods to find the best combination of model parameters instead of just depending on or using the default model parameters when building the final model. It is proven in this project, the model built with a combination of fine-tuned parameters tends to perform the best (highest accuracy and lowest testing error).

Our Team

Chen, Lulu (lchen612) - Graduated in Actuarial Science from Simon Fraser University in Canada. Now work full-time in Prudential Hong Kong as an actuary.

Chua, Christian (cchua8) – Graduated in Business Management from Singapore Management University. Currently working in PwC on data management projects.

Wallace, Bradley (bwallace35) – Graduated as a civil engineer from the University of Adelaide. Now working and studying Korean part-time, alongside the Master's degree, while living in South Korea.

Wiratama, Ardy (awiratama3) - Works as a Senior Industry Process Consultant for Dassault Systemes and serves the energy and resources industry for over 10 years. He is based in Perth, Australia.

Zhang, Yichi (yzhang3825) – Graduated as a computer engineering student from National University of Singapore. Now working in Accenture as a consultant specializing in wealth management and advisory solutions.

References

1. Banco Santander (2016) *Santander product recommendation*, *Kaggle*. Banco Santander. Available at: https://www.kaggle.com/competitions/santander-product-recommendation/data (Accessed: February 17, 2023).

Appendix

Appendix: Random Forest Calculations

Random Forest (Current Account)

mtry	Training Error
1	0.442316
2	0.283743
3	0.283718
4	**0.283101**
5	0.286508
6	0.300309
7	0.317047
8	0.320158
9	0.325244
10	0.326528
11	0.324602

Ntree	Training Error
50	**0.283669**
100	0.283718
200	0.283693
300	0.283716
400	0.283743

Node Size	Training Error
1	**0.283767**
2	0.283718
3	0.283718
4	0.283792
5	0.283817

Results (4, 50, 1)
Final Training Error: 0.277
Final Testing Error: 0.284

Random Forest (Particular Account)

mtry	Training Error
1	0.1511665
2	0.1510925
3	0.1387483
4	**0.1363535**
5	0.1393902
6	0.1450932
7	0.1493149
8	0.1500802
9	0.1511665
10	0.1512653
11	0.1524503

Ntree	Training Error
50	0.1359585
100	0.1363535
200	0.1352673
300	0.1348229
400	**0.1346994**

Node Size	Training Error
1	0.1353168
2	0.1346994
3	0.1346994
4	0.1349957
5	**0.1345019**

Results (4, 400, 5)
Final Training Error: 0.118
Final Testing Error: 0.129

Random Forest (Direct Debit)

mtry	Training Error
1	0.1183311
2	**0.1183311**
3	0.1183558
4	0.1188989
5	0.1222318
6	0.1293667
7	0.134255
8	0.1349463
9	0.1361807
10	0.1368226
11	0.1365757

Ntree	Training Error
50	0.1183311
100	0.1183311
200	0.1183311
300	0.1183311
400	**0.1183311**

Node Size	Training Error
1	0.1183311
2	0.1183311
3	0.1183311
4	0.1183311
5	**0.1183311**

Results (2, 400, 5)
Final Training Error: 0.118
Final Testing Error: 0.120

Appendix: Gradient Boosting Calculations

Gradient Boosting (Current Account)

Ntree	Training Error
50	0.2839649
100	0.2839649
200	0.2839649
300	**0.2838168**
400	0.2838909

Shrinkage	Training Error
0.001	0.2839649
0.01	0.2838662
0.1	0.2821133
1	**0.2784841**

Depth	Training Error
1	0.2836193
2	0.2802617
3	0.2813727
4	**0.2747068**
5	0.2790273

Results (300, 1, 4)
Final Training Error: 0.275
Final Testing Error: 0.307

Gradient Boosting (Particular Account)

Ntree	Training Error
50	0.1511665
100	0.1511665
200	0.1462289
300	0.1344278
400	**0.1327984**

Shrinkage	Training Error
0.001	0.1511665
0.01	0.1355141
0.1	**0.1305024**
1	0.1327737

Depth	Training Error
1	0.1336625
2	0.1319096
3	0.1301568
4	0.1295149
5	**0.1274657**

Results (400, 0.1, 5)
Final Training Error: 0.126
Final Testing Error: 0.131

Gradient Boosting (Direct Debit)

Ntree	Training Error
50	0.1183311
100	0.1183311
200	0.1183311
300	0.1183311
400	**0.1183311**

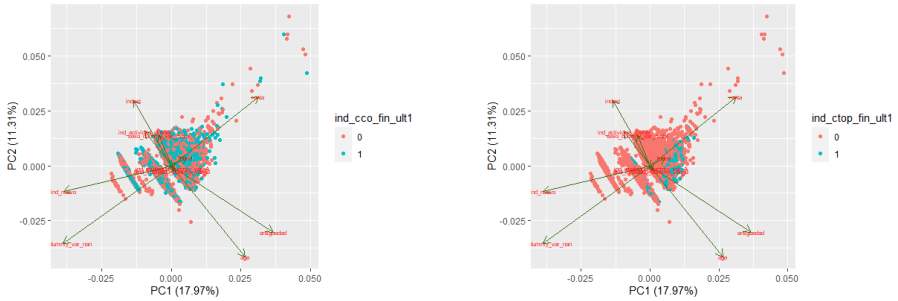
Shrinkage	Training Error
0.001	0.1183311
0.01	**0.1183311**
0.1	0.1168251
1	0.1390939

Depth	Training Error
1	0.1183311
2	0.1183311
3	0.1183311
4	0.1183311
5	**0.1183311**

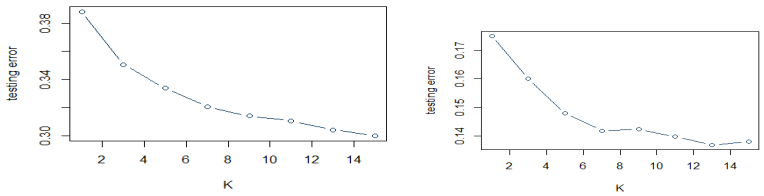
Results (400, 0.01, 5)
Final Training Error: 0.114
Final Testing Error: 0.120

Appendix: KNN with PCA methodology

- PCA interpretation by its response value and eigenvectors for Current Account *(left plot)* and Particular Account *(right plot)*.



- Best-K for Current Account (left plot) and Particular Account (right plot).



- Confusion Matrix for Current Account *(left)*, Particular Account *(middle)*, and Direct Debit *(right)*.

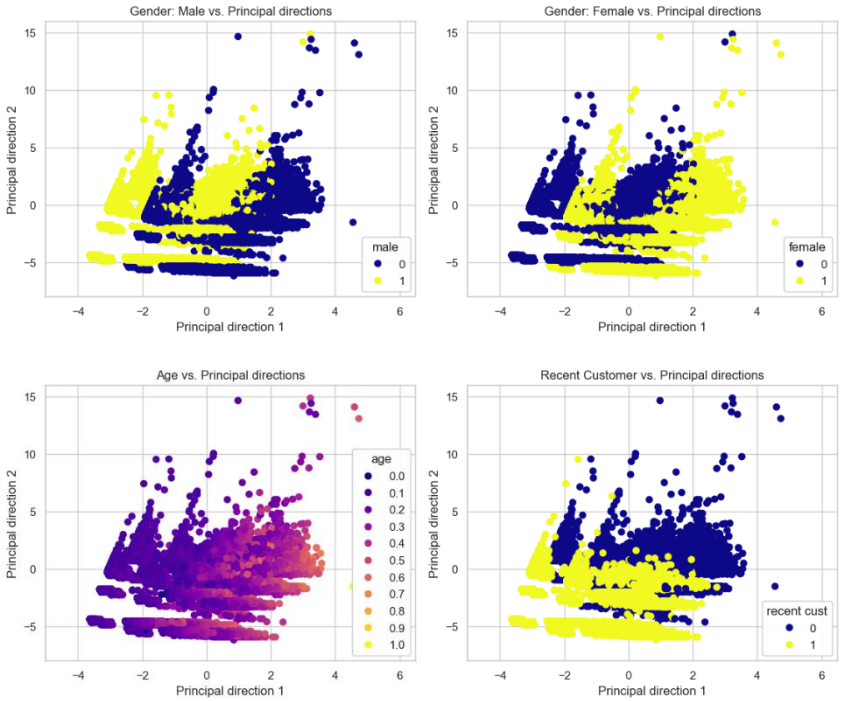
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 4017 1450
##          1 1583 3077
##
##                Accuracy : 0.7005
##                  95% CI : (0.6915, 0.7094)
##     No Information Rate : 0.553
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.3959
##
##  Mcnemar's Test P-Value : 0.01654
##
##             Sensitivity : 0.6797
##             Specificity : 0.7173
##          Pos Pred Value : 0.6603
##          Neg Pred Value : 0.7348
##              Prevalence : 0.4470
##          Detection Rate : 0.3038
##    Detection Prevalence : 0.4602
##       Balanced Accuracy : 0.6985
##
##        'Positive' Class : 1
##
```

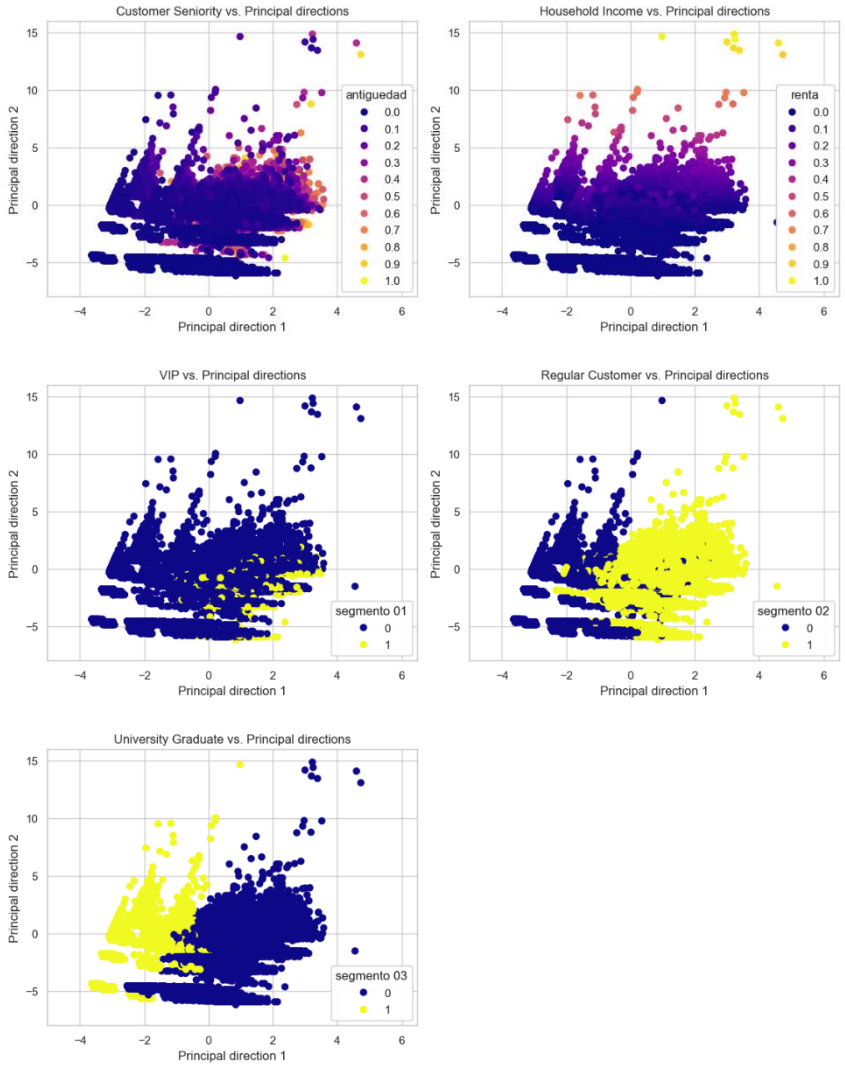
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 8259  377
##          1 1008  483
##
##                Accuracy : 0.8632
##                  95% CI : (0.8564, 0.8699)
##     No Information Rate : 0.9151
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.3398
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.56163
##             Specificity : 0.89123
##          Pos Pred Value : 0.32394
##          Neg Pred Value : 0.95635
##              Prevalence : 0.08492
##          Detection Rate : 0.04769
##    Detection Prevalence : 0.14723
##       Balanced Accuracy : 0.72643
##
##        'Positive' Class : 1
##
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 8720  192
##          1 1070  145
##
##                Accuracy : 0.8754
##                  95% CI : (0.8688, 0.8818)
##     No Information Rate : 0.9667
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1422
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.43027
##             Specificity : 0.89070
##          Pos Pred Value : 0.11934
##          Neg Pred Value : 0.97846
##              Prevalence : 0.03328
##          Detection Rate : 0.01432
##    Detection Prevalence : 0.11998
##       Balanced Accuracy : 0.66049
##
##        'Positive' Class : 1
##
```

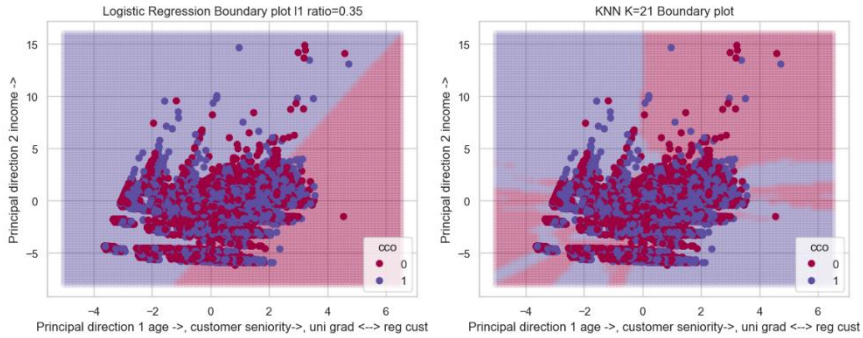
Appendix: Decision Boundary Visualization - PCA 2-Dimension Approximations

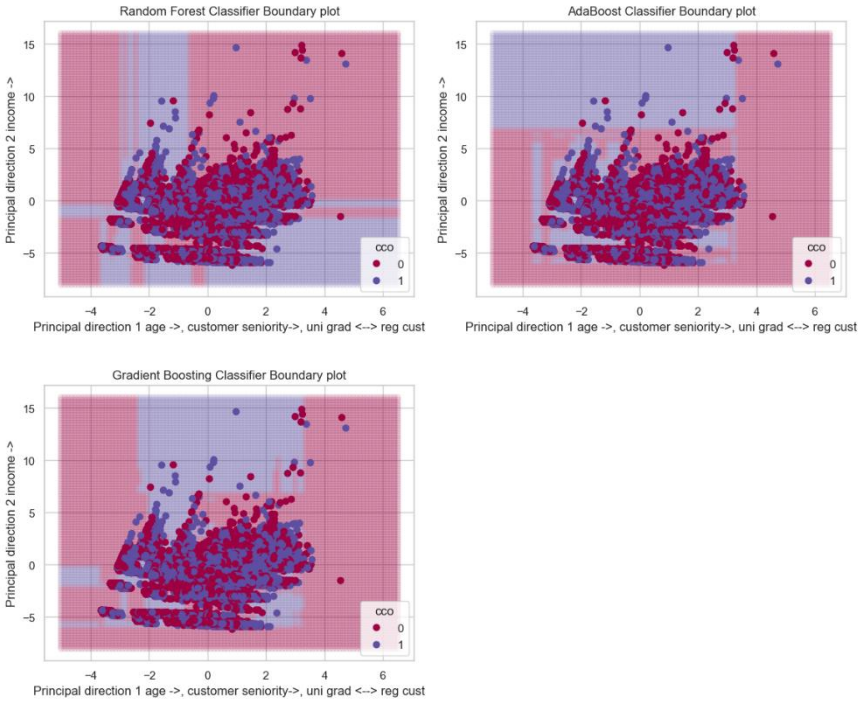
Predictor Variable Projection into 2D Feasibility Analysis



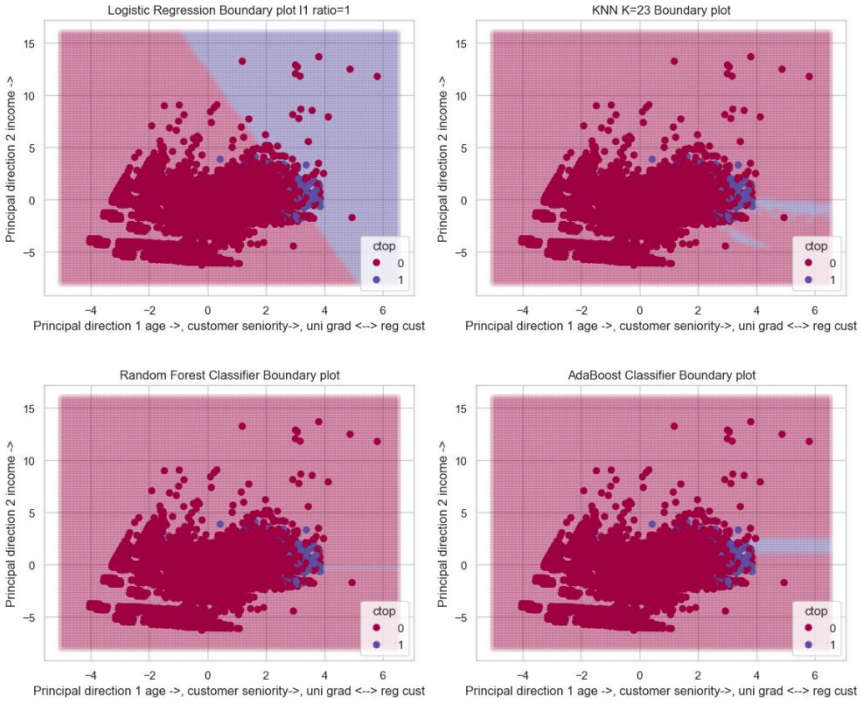


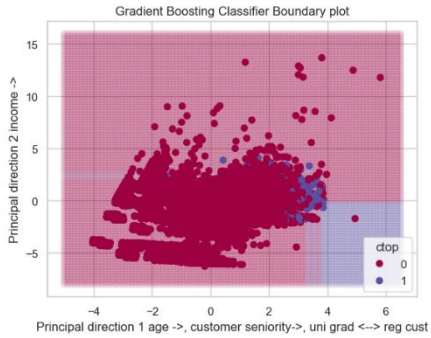
*2D Estimations of Model Architectures – Current Account (**cco**)*





*2D Estimations of Model Architectures – Particular Account (**ctop**)*





*2D Estimations of Model Architectures – Direct Debit (**recibo**)*