Santander Bank Recommendation System

Final Report

GR 5291 Advanced Data Analysis

Group 6: Xinyi Chen (xc2464), Sitong Liu (sl4460), Shiwei Hua (sh3804), Caihui Xiao (cx2225), Zixiao Wang (zw2513), Bingyue He (bh2692), Xiaotong Li (xl2788), Yuli Hong (yh3044), Haiqing Xu (hx2259), Runjie Lyu (rl3032), Qingyu Zhang (qz2351), Xuran Jia (xj2222), Xiaomeng Huang (xh2407), Jie jin (jj2972)

Content

1- Introduction

Santander Bank originally started in Spain in 1902, and extended its services through United States since 2013. Santander Bank is based in Boston, and operates more than 650 banking offices among the U.S. As one of the top retail and commercial companies, Santander Bank not only has $65 billion in assets, but also has built financial relationships with 2.1 million clients. The mission of Santander Bank is to prioritize the customers' financial needs, and maximize the bank profit. Thus, to better treat their clients, the bank offers personalized product recommendations. However, the current recommendation system has a limit, which is suggesting a large number of financial products to a small number of clients. Santander needs a system that recommends financial products to each client, and filter the number of recommended products.

In order to solve the problem, 1.5 years of the Santander's dataset has been collected from Kaggle. After preprocessing the original dataset and performing explanatory data analysis, a Santander product recommendation system was built to predict which products that the bank clients would purchase in the next month based on their historical transactions.

Through the whole project, several recommendation methods were applied, including random guess, user-based method and matrix factorization method, and machine learning methods, especially the XGBoost model, on the training set. The performance of different methods was evaluated on the test set using precision, recall, and F-1 score.

2- Project Objective

The main objective of this project is to recommend the financial products that Santander's customers would like to purchase in the next month based on their transaction histories from

January 2015 to May 2016. A precise recommendation algorithm will effectively boost Santander's revenue and provide customers with personalized experiences.

3- Data Source and Description

In order to better understand and manage the dataset, we pre-processed the dataset to remove useless information and used data visualization techniques to explore the relationships among features.

3.1 - Data Preprocessing

The original dataset contains 14 million records and 48 features (24 customer features and 24 purchase features). Two features, the last date as primary customer and spouse index, were deleted since more than 99% of the values were missing. About 0.6% of customers in the dataset had missing values in product features and they were removed from the dataset as well. In addition, cases with unreasonable values, such as customers with age larger than 100, were also eliminated.

After data preprocessing, there are 8,574,273 records and 46 features (such as age, gender, location, credit card, debit account, etc.) in the final dataset. (See Appendix 1 for details)

3.2 - Exploratory Data Analysis

3.2.1- Product Popularity

There are 24 financial products from Santander bank, the popularity of each product is significantly different (Figure 1).

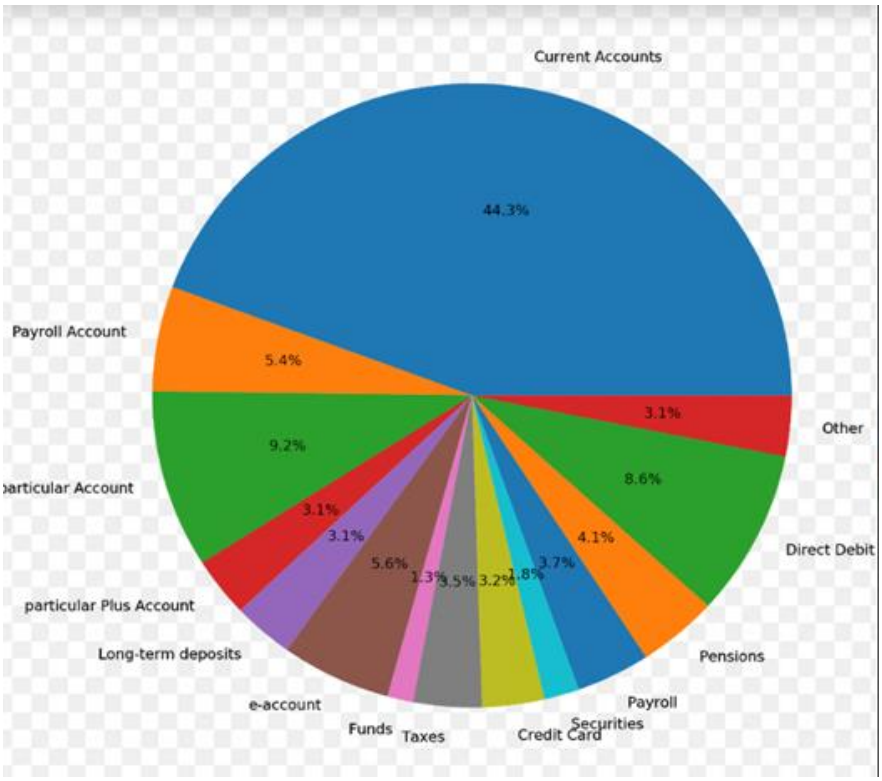

Figure 1. The above graph shows the percentage of products Santander clients purchased.

According to the graph, from 2015 to 2016, the top three purchased products are "current accounts", "particular account", and "direct debit". 44.3% of the clients have a current account with Santander bank, 9.2% of clients have a particular account, and 8.6% of clients have a direct debit. The percentage of the most popular product, the current account, is four times more than the second most popular product, a particular account. Moreover, there are eleven products with less than 1% of clients purchased, such as saving account, guarantees, medium-term deposit, etc.

Moreover, Santander clients purchase different number of financial products in each month. (Figure 2)

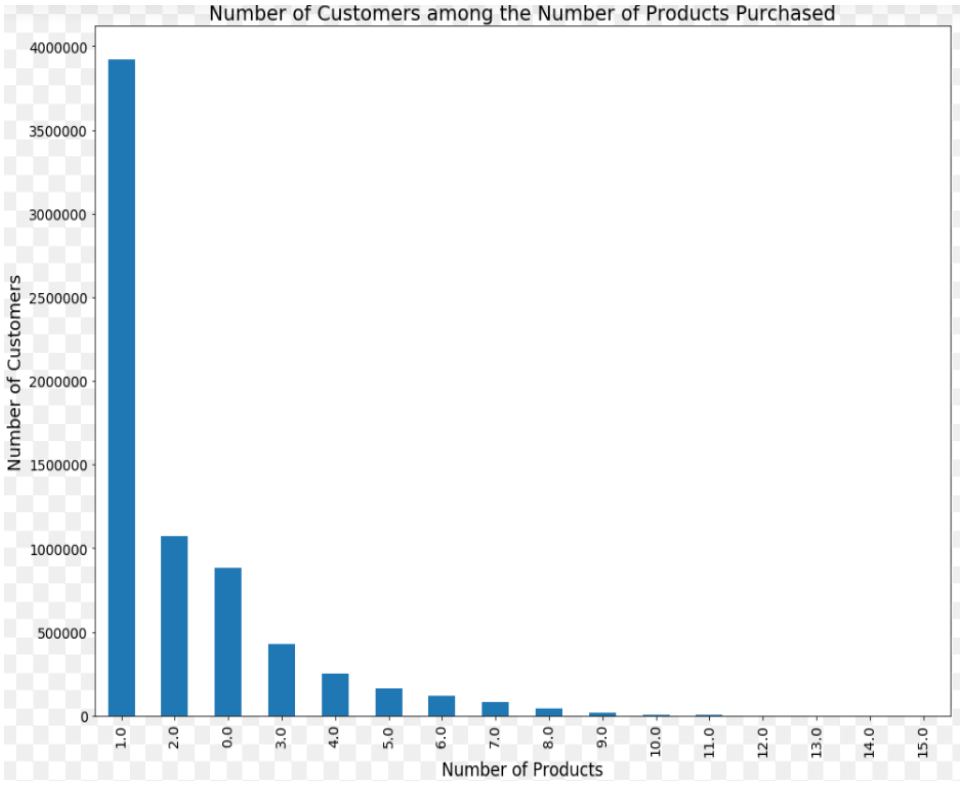

Figure 2. The above graph shows the number of financial products that a user is holding.

Based on the above graph, number of products purchased follows a right-skewed distribution. As the number of products increases, the number of customers decreases significantly. There is a significant difference: people who are holding one financial product at Santander is four times more than people who are holding two financial products. The maximum number of products that a user holds is eleven.

3.2.2- Age vs. Products

Among all the clients, their age appears to follow a bimodal distribution (Figure 3).

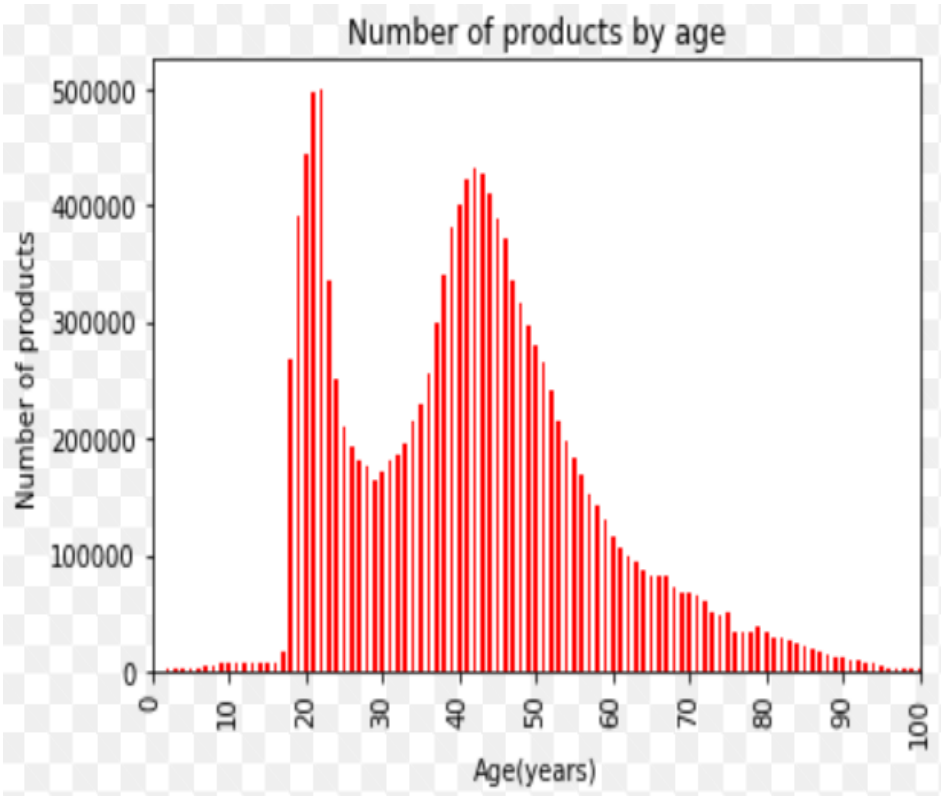

Figure 3. The above graph shows the age distribution among all customers.

From the bimodal age distribution, most of Santander's clients are from 18 years old to 70 years old. In addition, it has two modes, roughly around 20 years old and 45 years old.

Moreover, in the dataset, the age range is from 0 to 100, (all records with age larger than 100 years old have been removed from preprocessing step). Based on age, customers were separated into 5 groups, from 0-20, 20-40, 40-60, 60-80, 80-100. The differences in number of products purchased, and the preference over financial products among 5 groups were compared below (Figure 4).

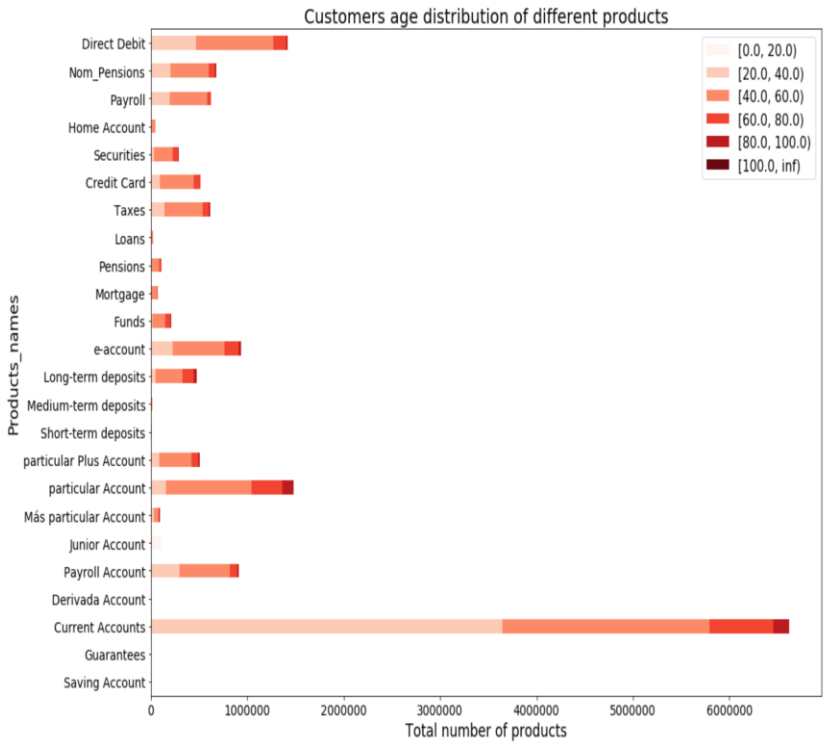

Figure 4: The above figure shows the customers' age distribution of different products.

Based on the above graph, the most popular account in the age group from 0 to 20 is a junior account. Most people from 20 to 40 years old purchased current accounts, direct debit, and e-account. The top three popular products are current account, particular account, and payroll account. People from 60 to 80 also purchased current account and particular account the most. Moreover, in the 80-100 age group, most people selected the current account.

3.2.3- Customer Relation Type (Active / Inactive) vs. Age

As mentioned, the client distribution among age returns a bimodal distribution. If compare the active customers and inactive customers using age distributions, both of them return bimodal distributions (Figure 5).

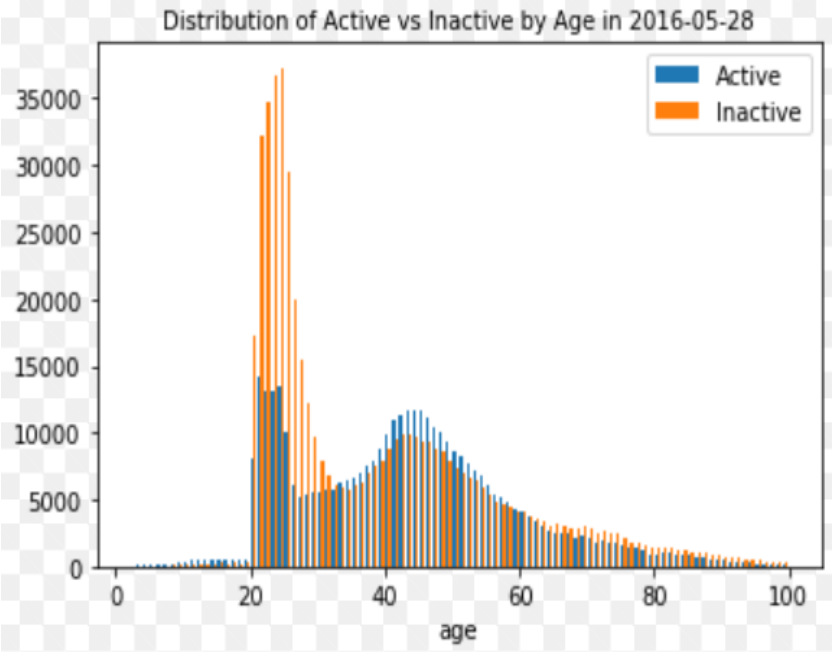

Figure 5: The above figure returns distribution of active vs. inactive by age.

According to the plot, it seems that most active customer age distribution have modes which are around 20 years old and 45 years old, but the mode at 20 years old are much higher than the mode at 45 years old. However, in the inactive customer's distribution, two modes are very close to each other.

4- Planned Methods of Analysis

4.1-Recommendation System Methods

For the recommendation system methods, user ID and all 24 financial product features were used. All customers' features such as age and sex were removed. In other words, these methods only consider the customer purchase history and ignore customers' personal information. Random Guess was adopted as a naïve approach, and popularity based method was chosen as the baseline. For further improvement, user-based method and matrix factorization method were applied.

4.1.1–Random Guess

The monthly dataset approximately contained 0.95 million data points and 24 financial product features. For each month, besides the products that Santander customers purchased in the previous month, 25,000 products were additionally purchased on average. If a product was randomly selected and recommended to a customer, the precision, which is the probability that this customer will purchase the product is 0.002668. However, the random guess method does not consider the specific transaction history of a certain client.

4.1.2 – Popularity Based Method (Baseline)

When applying the popularity based method, the popularity of all products were calculated using all data points, then the products were recommended for all customers based on the rank of product popularity. For example, if a client does not have a current account, the system would recommend the most popular product, a current account, to the client. On the other hand, if a client has a current account, the system would recommend the second most popular product, a particular account for the clients. This general popularity based method is the baseline method. To further improve the method, a user-based method was implemented.

4.1.3 – User-Based Method

User-based method is also known as user-user similarity filtering. It is one of the collaborative filtering methods, which considers the connection among clients. If there are many users who liked the same item then that item can be recommended to that user who hasn’t seen that item yet. The core idea of using the user-based method is to recommend products to new user based on what similar users have bought. To achieve this goal, there are 4 steps.

First, construct a user-item matrix. As shown, each row of the user-item matrix represents a user which contains purchase history of the user: '1' represents bought and '0' otherwise.

USER-ITEM Matrix

	I_1	I_2		I_j		I_{m-1}	I_m
U_1			...		...		
U_2			...		...		
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
U_i			...	A_{ij}	...		
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
U_{n-1}			...		...		
U_n			...		...		

Second, find similar users. Similarities between users are calculated based on user-item matrix using two ways: Cosine and Pearson Correlation. Here, two users will be similar on the basis of similar products they have bought. '1' represents highest similarity. Depending on all the information above, top p (self-selected parameter) most similar users could be found.

Third, pick products purchased by similar users. Top q (self-selected parameter) items bought by similar users that are not yet known/purchased by the target user.

Fourth, recommend those items to the target user.

Note: In this project, data is given monthly. Thus, the above steps are also repeated monthly.

4.1.4 - Matrix Factorization

Generally, most customers will not buy many products, so the user-item matrix is usually sparse, which contains a majority of missing values. Instead of treating this as a severe problem, matrix factorization takes advantage of the matrix sparsity to make recommendations: "decompose" user-item matrix (use 'A' in the following discussion) into two matrices U and I, where U*I can approximate the A the most. U and I could be found such that the following condition hold:

$$argmin_{(U,I)}\Sigma(A_{ij\{i,j\ such\ that\ a_{ij}\ is\ not\ empty\}} - U_i^T * I_j)$$

Hence, everything has turned into an optimization problem. Algorithms like Stochastics Gradient Descent was used to minimize the loss function above. Then, matrix A could be reconstructed using the product of U and I, where the new A' has no empty entries and each entry represents the predicted scores for a customer to purchase the product. Now recommending is nothing but looking up the complete matrix A'.

4.2- Machine Learning Method

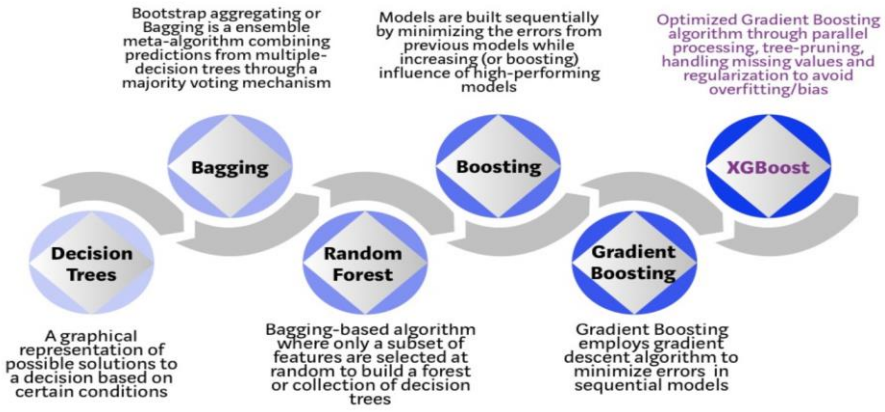

Figure 6. Evolution of the XGBoost model from Decision Tree

In the machine learning method, the XGBoost algorithm is used. Traditionally, in a multiclass classification problem, the XGBoost only predicts one product for each customer. However, in this case, each customer may purchase more than one product in each month, which makes it a multi-label prediction problem. To tackle the problem, "one versus rest" is used, which means training a single-label classifier for each product. There are three versions of input data. The first version only includes original customer features. The second version further includes the following

temporal features: products purchased last month (24 features) and number of products purchased last month (1 feature). The third version includes temporal features only.

5- Results

After evaluating all recommendation system methods and the XGBoost models using precision, recall and F-1 score, the final results are as follows (Table 1).

Model	**Precision**	**Recall**	**F1-Score**
Baseline (Popularity)	0.00345	0.00228	0.00275
Cosine Similarity	0.01267	0.01094	0.01174
Pearson Correlation Similarity	0.00358	0.00226	0.00277
Multifaceted Collaborative Filtering	0.02554	0.39845	0.04800
XGBoost (OriginalFeatures Only)	0.01629	0.56339	0.03166
XGBoost (Original Features and Temporal Effects)	0.23066	0.09883	0.13837
XGBoost (Temporal Effects Only)	0.22497	0.12176	0.15800

Table 4. The above table indicates the result of each model.

Note:

The evaluation metrics are as follows:

$$Precision = \frac{number\ of\ correctly\ predicted\ products}{number\ of\ recommended\ products}$$

$$Recall = \frac{number\ of\ correctly\ predicted\ products}{number\ of\ actually\ purchased\ products}$$

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

From the formula, it is clear that F1-Score is a weighted average of Precision and Recall.

6- Discussion/ Conclusion

The random guess approach gives a precision of 0.002668. The baseline model (popularity) shows slight improvement, yielding a precision value of 0.00345 and a recall value of 0.00228. Between two similarity-based methods, cosine similarity gives us a better result, with an F1-Score of 0.01174. Multifaceted Collaborative Filtering produces an F1-Score of 0.048. Among XGBoost models, the one with temporal features only performs the best, with an F1-Score of 0.158.

Based on these results, it is clear that temporal features have more predictive power compared with original features. For future studies, more temporal features could be considered to improve performance of the XGBoost model.

7- Bibliography

1. Koren, Yehuda. Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model. Florham Park, KDD '08, 2008, dl.acm.org/citation.cfm?id=1401944&preflayout=flat. Accessed 8 Dec. 2019.

2. Morde, Vishal. "XGBoost Algorithm: Long May She Reign!" *Towards Data Science*, Towards Data Science, towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d. Accessed 8 Dec. 2019.

3. "Santander Product Recommendation." *Kaggle*, 21 Dec. 2016, www.kaggle.com/c/santander-product-recommendation/overview. Accessed 8 Dec. 2019.

8- Appendix

Appendix : Data Description

Column Name	Description
fecha_dato	The table is partitioned for this column
ncodpers	Customer code
ind_empleado	Employee index: A active, B ex employed, F filial, N not employee, P pasive
pais_residencia	Customer's Country residence
sexo	Customer's sex
age	Age
fecha_alta	The date in which the customer became as the first holder of a contract in the bank
ind_nuevo	New customer Index. 1 if the customer registered in the last 6 months.
antiguedad	Customer seniority (in months)
indrel	1 (First/Primary), 99 (Primary customer during the month but not at the end of the month)
indrel_1mes	Customer type at the beginning of the month ,1(First/Primary customer), 2(co-owner), P(Potential), 3(former primary), 4(former co-owner)

tiprel_1mes	Customer relation type at the beginning of the month, A (active), I (inactive), P (former customer),R (Potential)
indresi	Residence index (S (Yes) or N (No) if the residence country is the same than the bank country)
indext	Foreigner index (S (Yes) or N (No) if the customer's birth country is different than the bank country)
canal_entrada	channel used by the customer to join
indfall	Deceased index. N/S
tipodom	Addres type. 1, primary address
cod_prov	Province code (customer's address)
nomprov	Province name
ind_actividad_cliente	Activity index (1, active customer; 0, inactive customer)
renta	Gross income of the household
segmento	segmentation: 01 - VIP, 02 - Individuals 03 - college graduated
ind_ahor_fin_ult1	Saving Account
ind_aval_fin_ult1	Guarantees
ind_cco_fin_ult1	Current Accounts
ind_cder_fin_ult1	Derivada Account

ind_cno_fin_ult1	Payroll Account
ind_ctju_fin_ult1	Junior Account
ind_ctma_fin_ult1	Más particular Account
ind_recibo_fin_ult1	particular Account
ind_ctpp_fin_ult1	particular Plus Account
ind_deco_fin_ult1	Short-term deposits
ind_deme_fin_ult1	Medium-term deposits
ind_dela_fin_ult1	Long-term deposits
ind_ecue_fin_ult1	e-account
ind_fond_fin_ult1	Funds
ind_hip_fin_ult1	Mortgage
ind_plan_fin_ult1	Pensions
ind_pres_fin_ult1	Loans
ind_reca_fin_ult1	Taxes
ind_tjcr_fin_ult1	Credit Card
ind_valo_fin_ult1	Securities

ind_viv_fin_ult1	Home Account
ind_nomina_ult1	Payroll
ind_nom_pens_ult1	Pensions
ind_recibo_ult1	Direct Debit