Churn Bank Customer Database

Analysis, Modelisation & Scoring

Full-stack customer churn prediction pipeline for banking, from exploratory data analysis to internal scoring

- Objective / Context
- Data Gathering / Cleaning
- SQL Work
- API Work
- Statistical analysis
- Modélisation
- Model Limits
- Scoring
- Combining
- Conclusion

Thomas Giret

Objective and Context

Main objectives

- Working on a churn database with the support of python, sql and Tableau.
- Working on the development of a prediction model.

Context

- Data Analyst for a retail bank
- Working with dataset of 10,000 customers (including demographic details and financial behaviors)
- Tasked to build machine learning pipeline
- Predict the largest volume churners customers

Data Gathering and Cleaning

Data Source

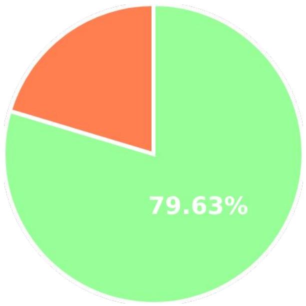
- Kaggle
- CSV
  - Pandas (jupyter Notebook)
  - Traceability, standard data analytics practices

Data Cleaning

- No null data
- No duplicates data
- Column deleted : RowNumber

Unbalanced dataset

Exited variable unbalanced :



20,37% of churn customers

SQL Work

ERD

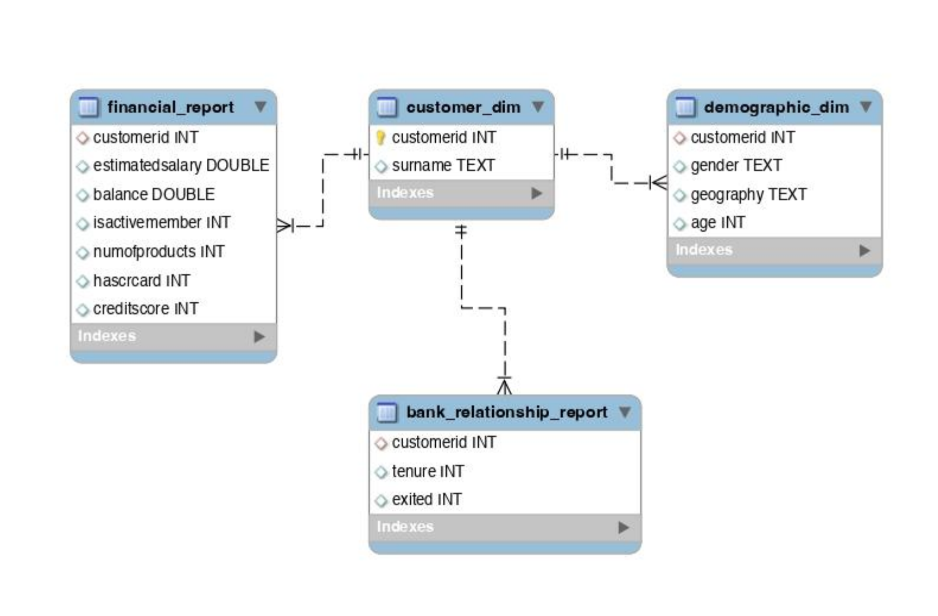
- 4 tables (from jupyter notebook) exported in csv
- Imported in My SQL Workbench

financial_report
Information about customers activities

demography_dim
Demography information

bank_relationship_report
Information about relation between customers and bank

customer_dim
main informations about customer



Query example 1

- Churn rate by geography by gender

geography	gender	churn_rate	total_customers
France	Female	0.20	2727
Spain	Female	0.21	1142
Germany	Female	0.24	1195
France	Male	0.15	2753
Spain	Male	0.13	1297
Germany	Male	0.17	1383

Query example 2

- Average âge and number of product by exited

exited	avg_age	avg_num_products	total_customers
0	37	1.54	7963
1	45	1.48	2037

API Development

Ressource Customer

GET /api/customers

List all customers with pagination and filtering

→ **Key Parameters** : page, limit, filters

GET /api/customers/{id}

Retrieve individual customer details

→ **Key Parameters** : `{id}` (customer ID)

```
### 2. UNIQUE CUSTOMER (Nested Details)
Status: OK (200). Details for H?:
  > First Name: N/A
  > Age: 27
  > Balance: 134603.88
  > Churn: False
```

Ressource Analytics

GET /api/analytics

Return a index of all 5 available analysis report (query)

GET /api/customers/{report_id}

Execute the corresponding query

→ **Key Parameters** : `{report_id}` (query ID)

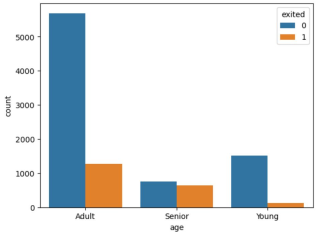
```
### 3. ANALYTICS LIST (List of Reports)
Status: OK (200). Number of available reports: 5
Report #4 Name: Multi-Product Analysis

### 4. ANALYTICS EXECUTION (Query 4 - Multi-Product)
Status: OK (200). Report ID 4 loaded.
Description: Balance and Tenure for customers with >1 product, ordered by balance
Query Result (Avg Balance by number of products):
    avg_balance  avg_tenure  numofproducts  total_customers
0  93733.135000    5.300000              4               60
1  75458.328195    5.003759              3              266
2  51879.145813    5.051852              2             4590

========================================
```

Statistical Analysis

Variables for model

Âge

Grouped into **bins**: **Young** (<30), **Adult** (30-50), and **Senior** (>50) (T-test and countplot)



Geography

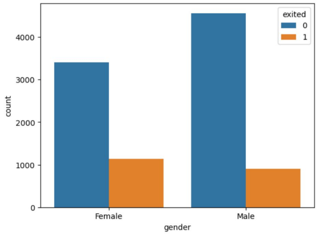
German have higher churn rate

Z-test showed no significant difference in churn between **France** and **Spain**.

Gender

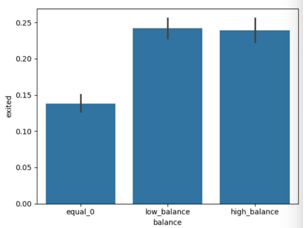
Women churn more than men

(chi-test and countplot)



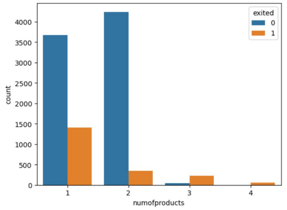
Balance

T-test showed churners have higher balances.
Converted to **Binary**: 0 (Zero Balance) vs 1 (Positive Balance).



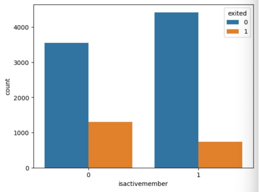
Number of Product

Chi-2 showed a strong relationship (high Cramer's V).



Active Member

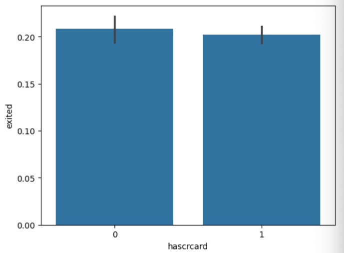
Z-test confirmed inactive members are ~2x more likely to churn.



Variables dropped

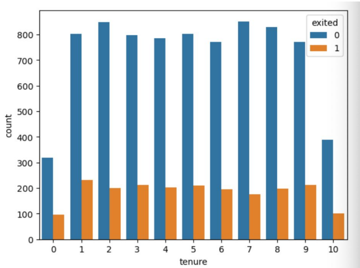
Has Credit Card

Dropped: Tests confirmed the absence of a significant relationship with churn.



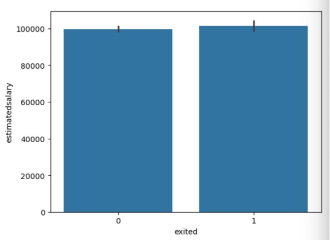
Tenure

Dropped: Even after a categorisation of tenure, Chi-2 and Z-tests showed no significant relationship with churn.



Estimated Salary

Dropped: T-test and categorization showed no difference in churn proportion.



Modelisation

Preparation

- Encoding : LabelEncoder and one hot encoding for categorical variables
- Scaling : MinMaxScaler to normalize
- Sample : split 70/30

Why XGBoost ?

- Good model for tables prediction (including bank database)
- Learn by iteration about previous prediction (learn from him mistake)
- Allow quickly to get prediction with quality

Unbalanced Management

Only **20,37%** of customers churn

→ **SMOTE** : Give more importance to churn person

→ **Moving Threshold**

Why not other model ?

- **Logistic Regression:** Too linear
- **SVM (Support Vector Machine):** Too slow, not easily interpretable
- **Random Forest:** Solid, but less performant and less optimizable than XGBoost

Best Management and Performance Model

Not using Smote and Threshold 0,6

~0.76	~0.71	~0.56	~0.48
Accuracy	Recall	F1-score	Precision

- Accuracy : Not trusted (because of imbalance dataset)
- F1-score : Show that generates a lot of false positive
- Precision : Not matter because it does not cost the bank to find false positive
- Recall : **~0.71** = show that found a good part of true positive churner

Why Recall privileged ?

- Our goal is to find the largest number of churn customers
- Recall is the most important because it costs a lot to the bank to miss churn customers.

$$\text{Recall} = \frac{\text{True Positives } (TP)}{\text{True Positives } (TP) + \text{False Negatives } (FN)}$$

Model Limits

XGBoost limit

Even with results that can be satisfying, the model shows some limit :

2052 mistake in total
So almost 21% mistake in the prediction

560 False Negative (missed churner)
Worst scenario cause it cost a lot to the bank to miss a churner

1492 False Positive
It cost but it’s tolerable

Own Scoring Creation

Objective :
Creation of a simple score, understandable and able to reduce the False Negative number with a simple method

- Few Scoring test have been realised

Method

- Using a sample
- Choosing a dataset form
- Using the principle of Impact Encoding

  Impact Encoding = replace each variable with the **mean probability of the target variable** for that variable.
- Some all variable for each customer and Normalize 0→1.
- Classification into 4 levels of risk :

Very Low	Moderate	High	Very High
<0.25	0.25-0.5	0.5-0.7	>0.7

- Compare distribution of churn customers in risk level

Differents Scoring

Test 1

- Same dataset as the model
- Impact Encoding for each values

Score too flat

insufficient spread/differentiation
→ Probably because of a problem in my categorisation/encoding

Test 2

- Unprepared dataset (no encoding and no categorisation)
- Only variable with relationship with exited
- Impact Encoding for each values

Best Conclusion

- 97% of non-churned customers <0.25
- 0% of non-churned customers >0.5
- 13% of churned customers <0.25
- 69% of churned customers >0.5

→ **Conclusion:** Everything above 0.5 is churned.

Test 3

- Unprepared Dataset
- All variables
- Impact Encoding for each values

Conclusion more complexe

- 99% of non-churned customers <0.25
- 1% of non-churned customers >0.25
- 18% of churned customers <0.25
- 82% of churned customers >0.25

→ **Too much churned customer < 0.25**

Combining Model and Score

Combining Method

- Few Scoring test have been realised

Combining Test 1 → Best combining

Simple rule :

- **If customer score ≥ 0.25** → Predict churn
- **If customer score < 0.25** → Model prediction

+ =

This combining have for objective to reduce the number of False Negative

Combining Result

560 Missed Churn (FN) Before	**195** Missed Churn (FN) After
1492 False Positive (FP) Before	**1493** False Positive (FP) After

365 churns customers have been predicted right thanks to the combining

Conclusions and Next Steps

Conclusion on model prediction

The good result are due to :

- A comprehensive pipeline creation all around the data source
- Avoids searching complex parameters & Uses simple methods to maximize churn detection
- Hard to create and maintain in a company environment

Next steps

- Multivariate analysis
- Using only a model or a scoring for the churn prediction

Dashboard Demo

[Dashboard link]

Thank You !