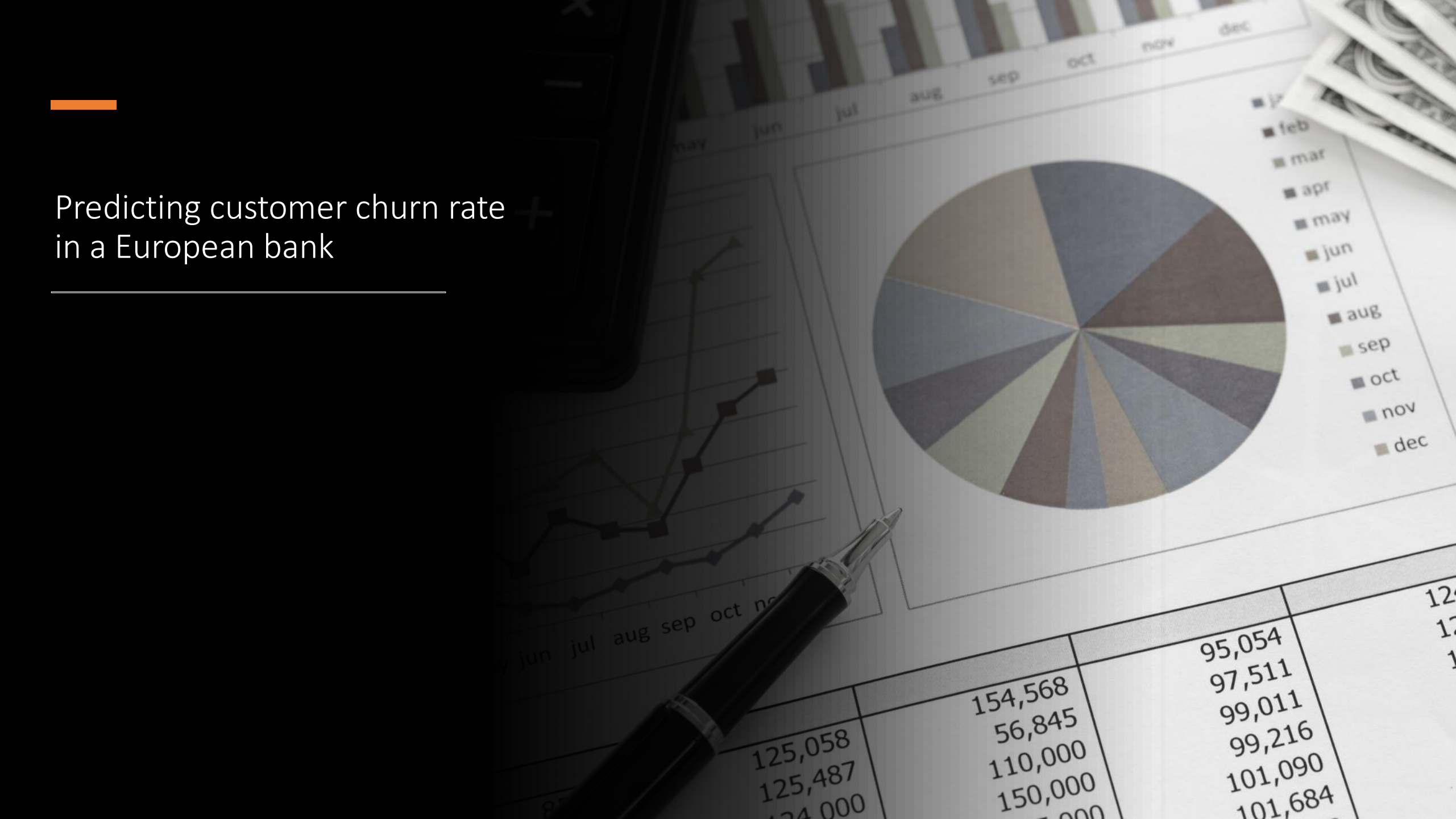
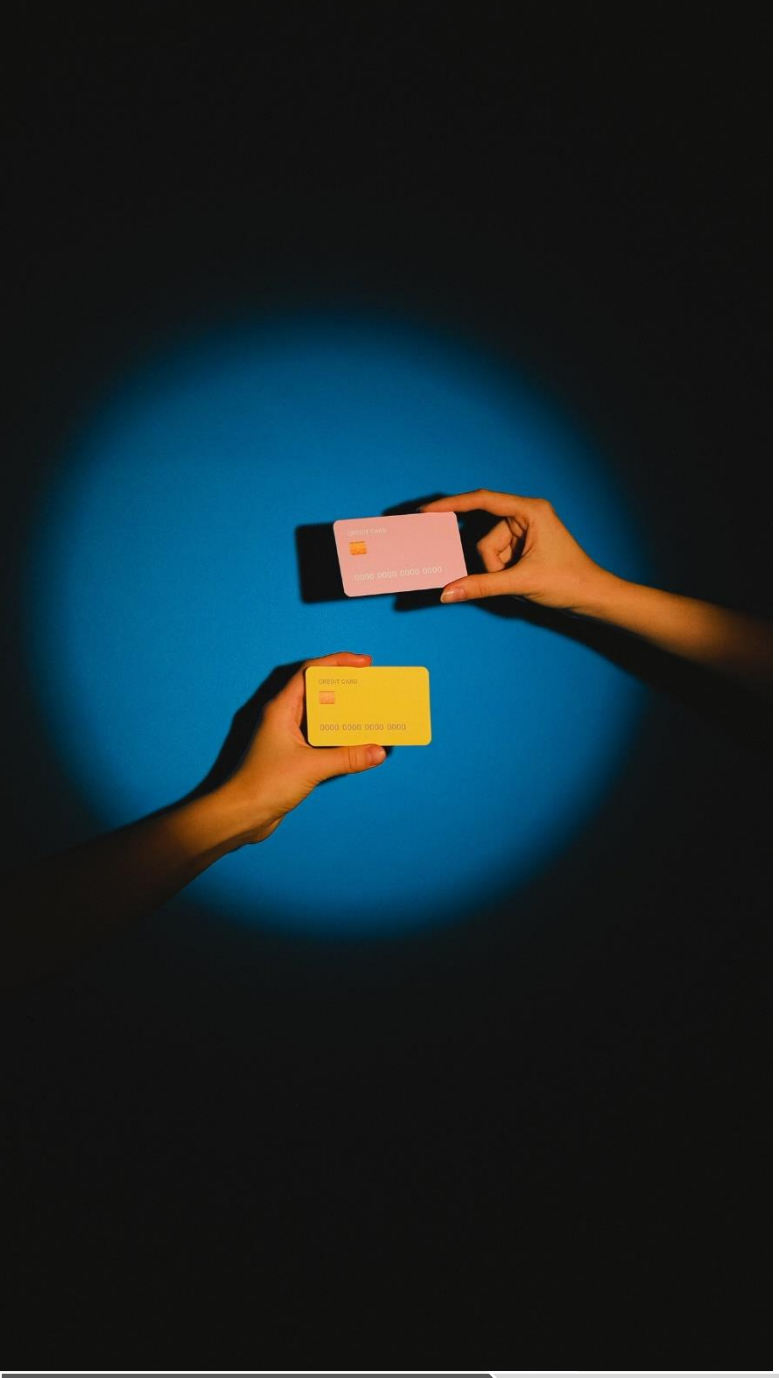
Predicting customer churn rate in a European bank





Case Introduction

Customer loyalty within the banking sector has recently seen a decline due to the increasing number of options available to customers to satisfy their banking needs[1]. This is a matter of concern for banks, considering that acquiring new customers can be 5 to 25 times more expensive than retaining existing ones[2].

Our client, the European multinational bank ABC, currently has a churn rate of 20% which they are committed to decrease both to enhance the customer experience and the company's bottom line. That's why they've appointed us to **develop a model to predict which customers are likely to churn.**

Our approach involves the creation of four distinct predictive models to address this binary classification problem. These models will assess whether customers with specific characteristics are likely to churn. Subsequently, we will evaluate the performance of these models based on key metrics such as recall and accuracy rates, enabling us to select the most effective model for implementation.

What is Customer Churn? — Credit: KPI Max

[1] https://www.bain.com/insights/customer-behavior-and-loyalty-in-banking-global-edition-2023/

[2] https://hbr.org/2014/10/the-value-of-keeping-the-right-customers

Business Motivation

With the help of the model the bank and their marketing team can identify customers in the risk of churning and focus their marketing efforts towards these customer instead of marketing for all.

Decreasing customer churn rate will lead to savings in the form of not having to attain new customers.

By preventing churn the bank can both avoid the costs of new customers
acquisition and also increase the overall customer lifetime value.

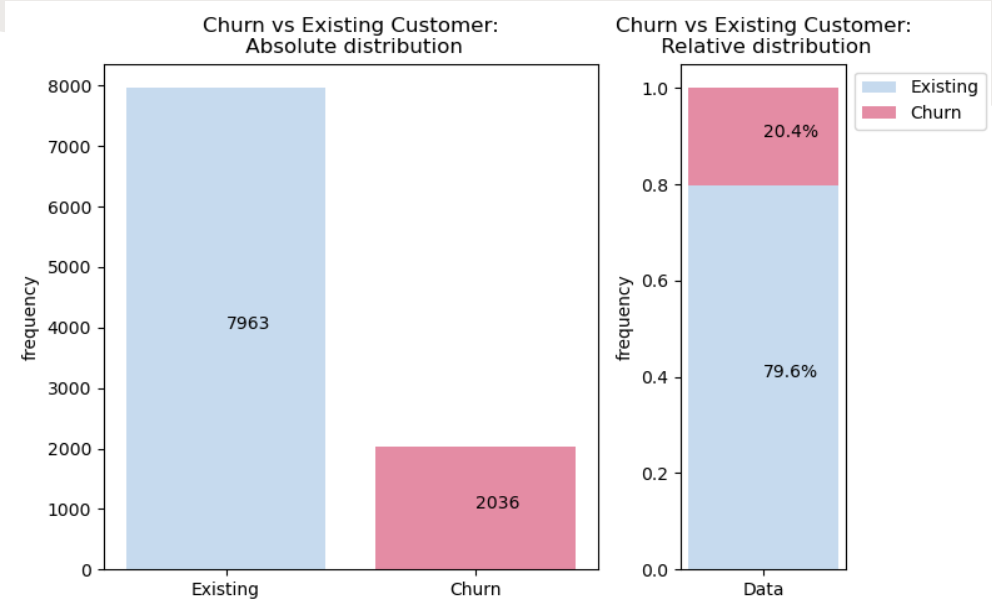
This will also help the bank to use their marketing budget in the most efficient manner

Data Understanding

NO.	FEATURE_NAME	DESCRIPTION	FEATURE_TYPE	DATA_TYPE
1	customer_id	Account Number	categorical	int
2	credit_score	Credit Score	numerical	int
3	country	Customer's Country of Residence	categorical	object
4	gender	Male/ Female	categorical	object
5	age	Customer's Age	numerical	int
6	tenure	Duration of customer relationship	numerical	int
7	balance	Account Balance	numerical	float
8	products_number	Number of products from bank	numerical	int
9	credit_card	Customer has a credit card	categorical	int
10	active_member	An active customer of bank	categorical	int
11	estimated_salary	Customer's Estimated Salary	numerical	float
TARGET 12	**churn**	**1 = Churn (left the bank) 0 = Non churn (did not leave bank)**	**categorical**	**int**

- **10 features** and **10.000 observations**
- **No missing values** (all variables show 10.000 entries)
- **No NaN values**
- Mainly **int** and **float**; **categorical encoding** for *'country. gender'* and **transformation** from object to float for *'estimated_salary'*
- **Categorized features**: *'age', 'tenure', 'estimated_salary',* and *'credit_score'*

Exploratory Data Analysis

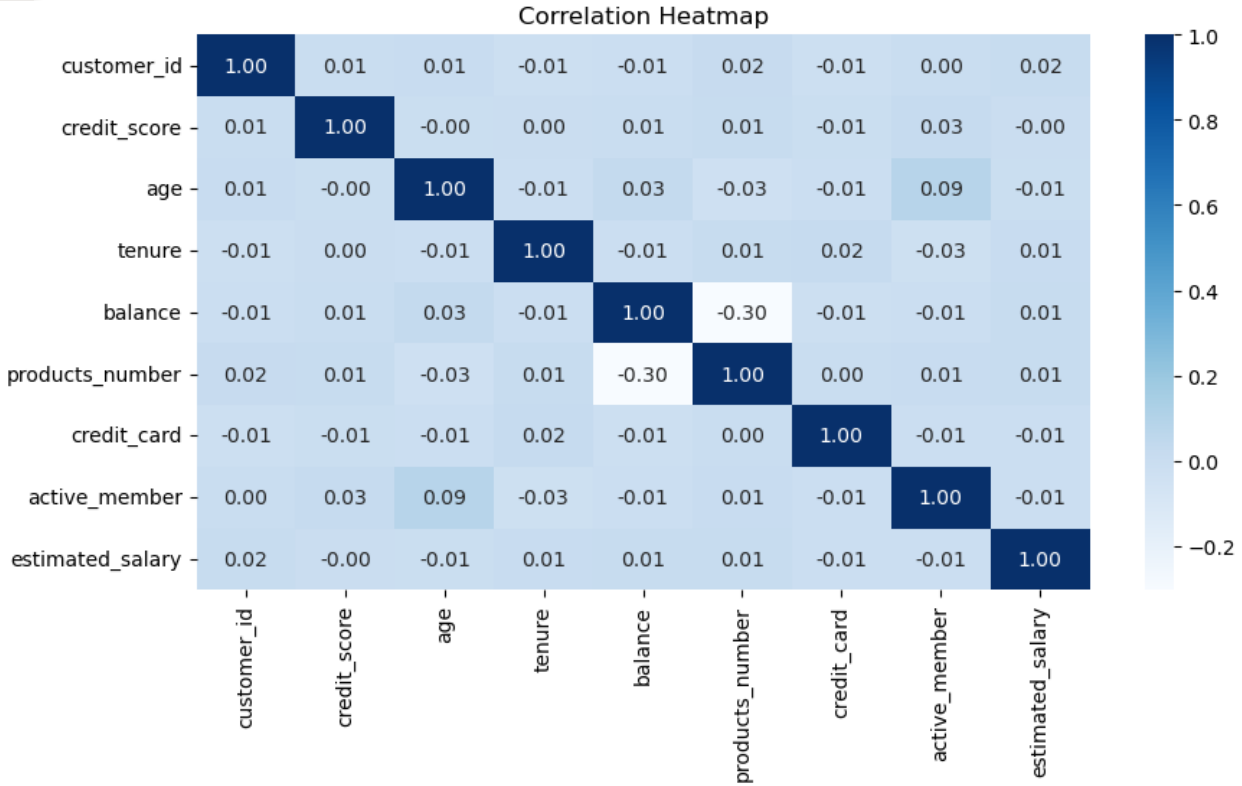


Data imbalance: **7.963 (79.6%) non-churned** vs. **2.037 (20.4%) churned customers**.
Consider special model training for balance.

The explanatory variables could be divided into two primary groups:

- **Continuous numeric** features
- **Categorical** features

Leverage the heatmap to visualize **numeric feature correlations,** guiding selection and **preventing multicollinearity** in models.



There are **no strongly correlated features** in our feature matrix.
Thus, no columns need to be removed.

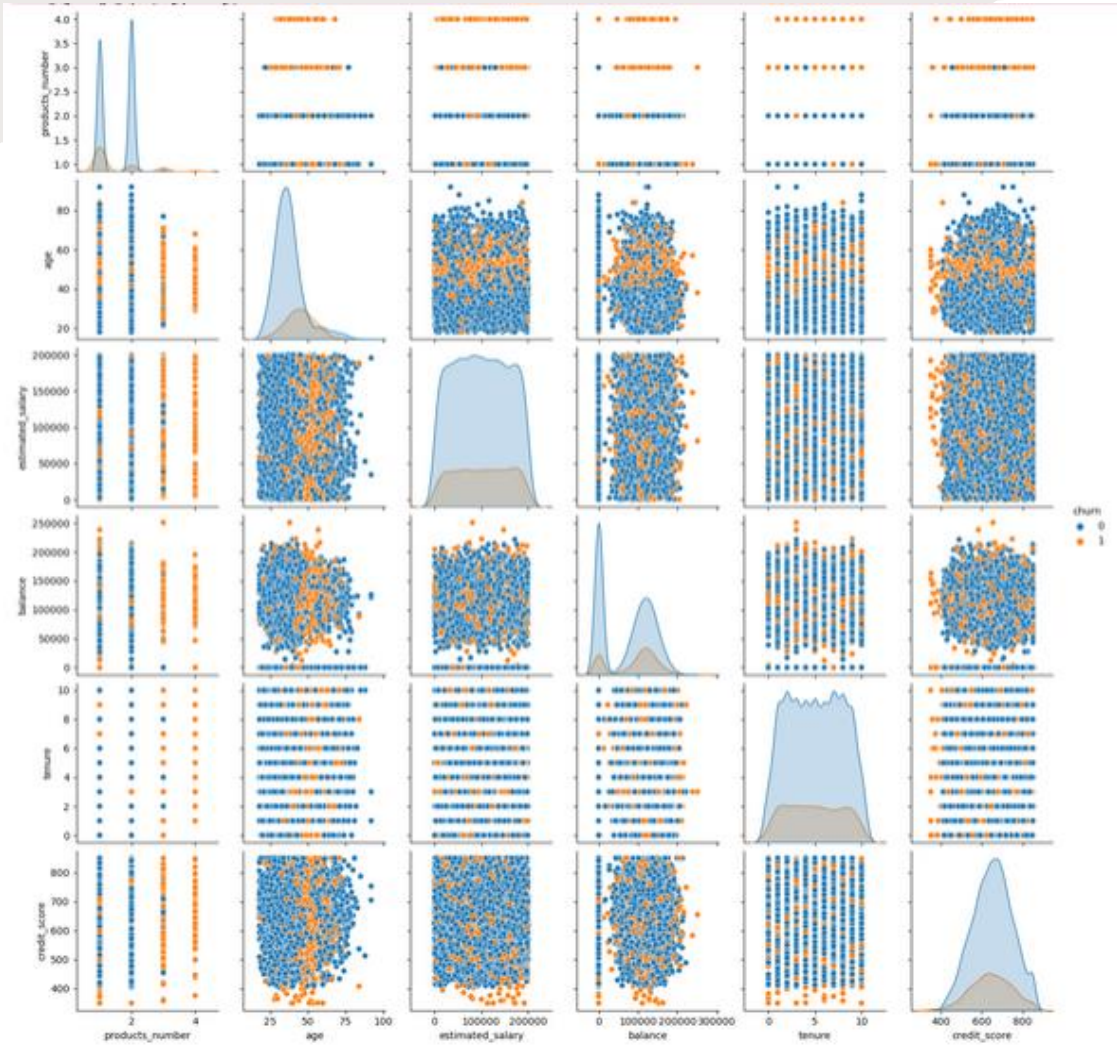
Exploratory Data Analysis

Analyzing the distribution of *numeric features* yields insights:

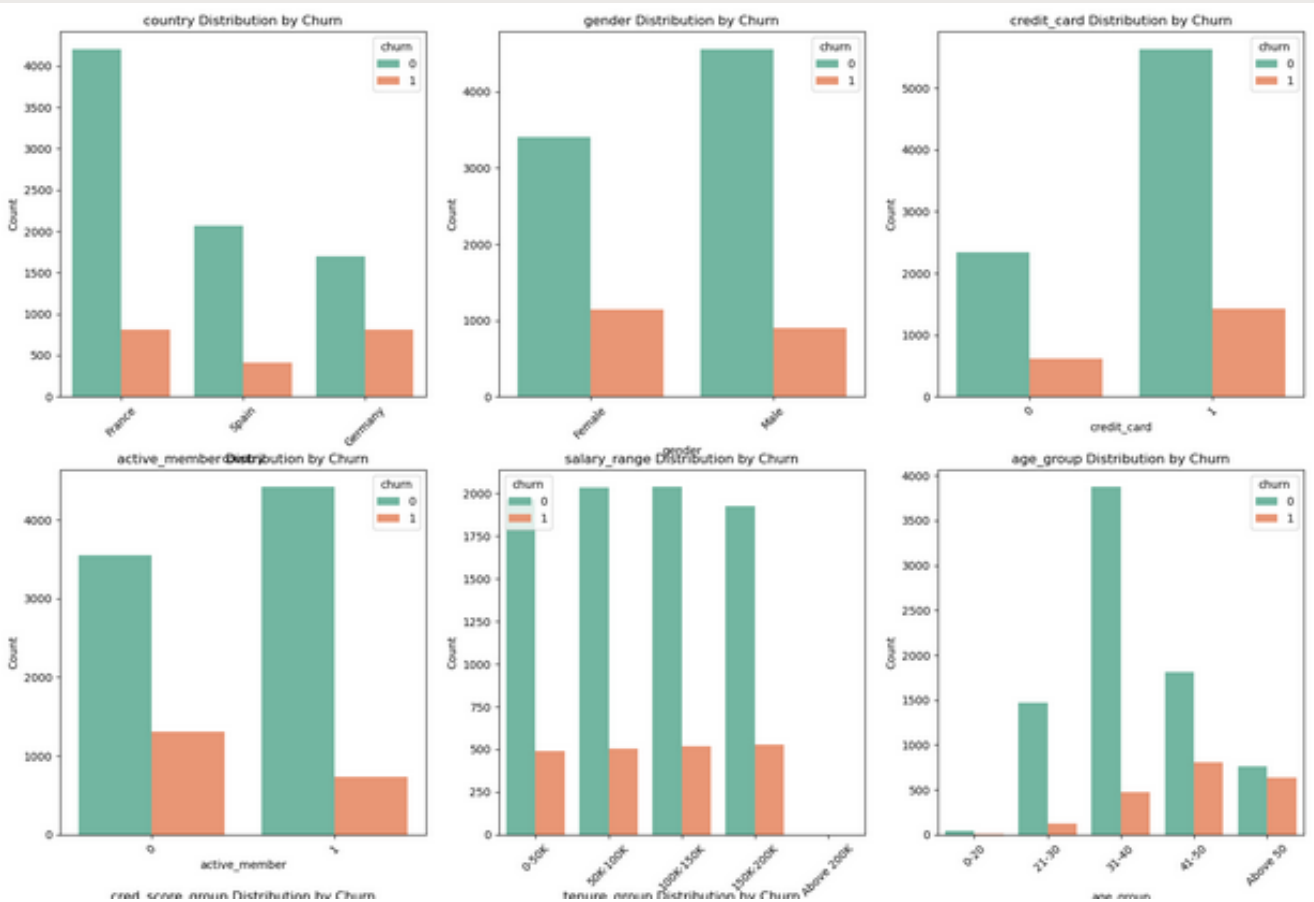
- **Credit score** approximates a **normal distribution. Age** has a skewed distribution (and hence has been transformed to logarithm).
- Most individuals in the dataset are **aged 20 to 40**.
- A significant portion has **no recorded balance**.
- **Salary** values are **well-distributed**.
- **Most churn customers has 1 product,** but who having **more than 3 have 100% churn rate**

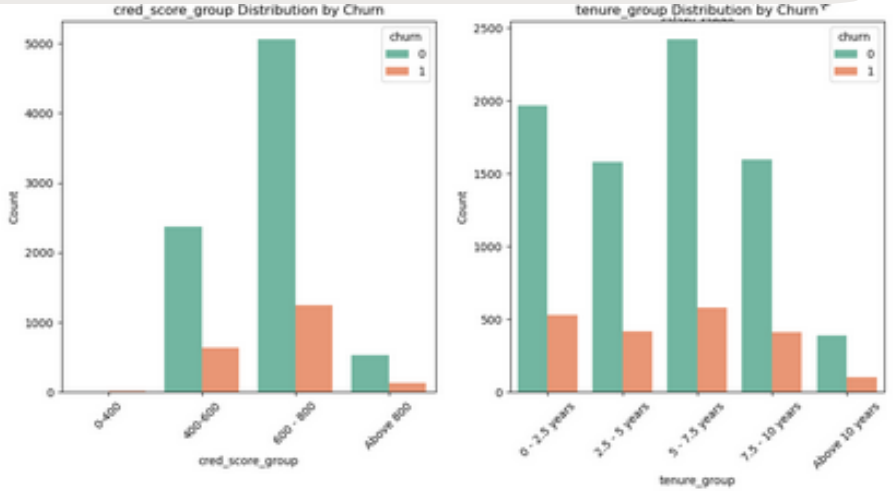
The **non-linear nature** of the dataset impacts model parameter choices. such as in Support Vector Machine.

- Data is re-balanced by SMOTE
- Use StandardScaler to scale the data used with SVM and Logistic Regression



Exploratory Data Analysis





Pinpoint potential churn-prone customer segments:

1. From **Germany**
2. Who are **Female**
3. Using **Credit Card**
4. Who are **Non-active members**
5. Above **41 years old**
6. With **credit scores below 400 or above 800**

Binary Classification: Model selection

Logistics Regression

Simple and interpretable

Can serve as a baseline model

Decision Tree

Simple and interpretable

Easy to visualize for presenting result

Random Forest

Ensemble method

Resistant to overfitting

Support Vector Machine

Able to find complex decision boundaries

Able to handle non-linear relationships

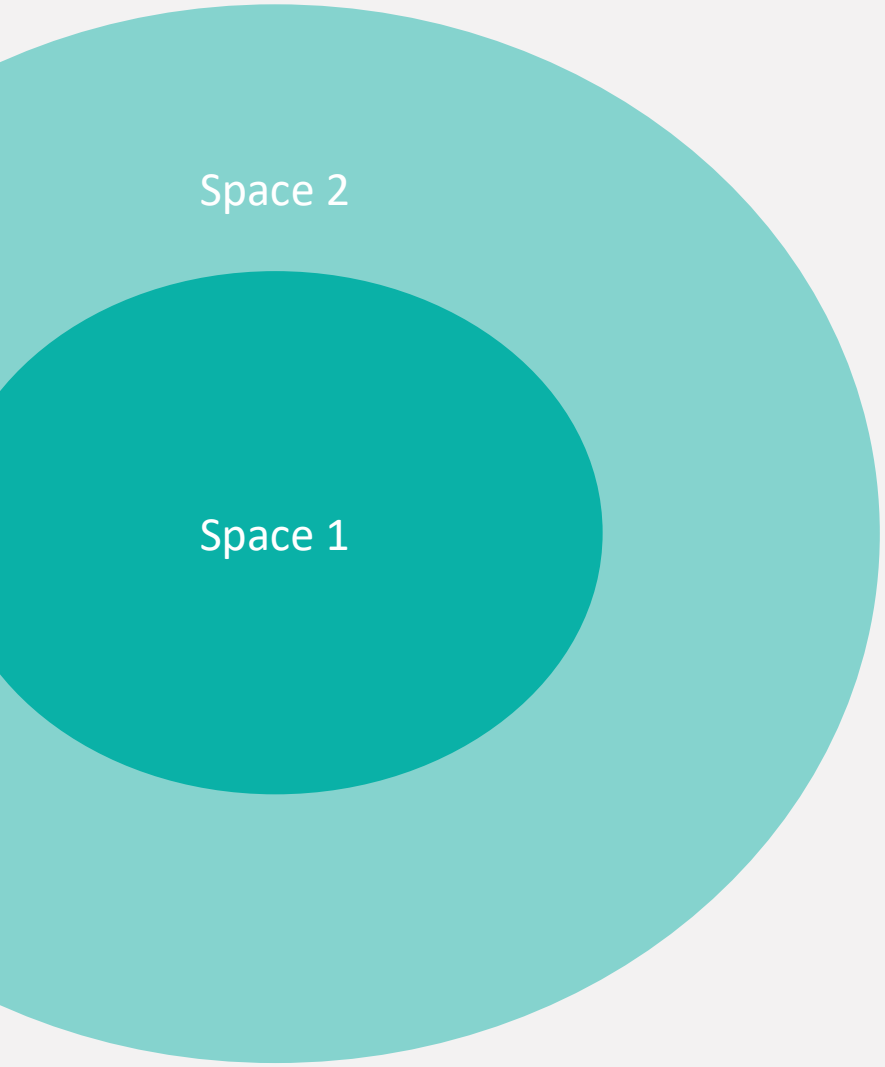
Modeling process:

- Run 4 models on original dataset and rebalanced data set (with SMOTE)
- Each model get optimized to reach highest performance
  - Hyperparameter tuning is done for Decision tree, Random Forest and SVM
  - Regularization is done for Logistic Regression
- Best model is selected based on the combination of metrics including precision, recall, F1-score and accuracy

Classification Reports for all models

		precision	recall	f1-score	accuracy
LogisticRegression	LogisticRegression Default	0.00	0.00	0.00	0.80
(baseline model)	LogisticRegression Balanced	0.24	0.66	0.35	0.51
	LogisticRegression Scaled - Optimized	0.64	0.20	0.30	0.82
DecisionTree	DecisionTree Default	0.83	0.27	0.41	0.84
	DecisionTree Balanced	0.47	0.65	0.55	0.78
	DecisionTree Default - Optimized	0.77	0.40	0.52	0.85
SVM	SVM Default	0.00	0.00	0.00	0.80
	SVM Scaled	0.21	0.11	0.15	0.73
	SVM Scaled - Optimized	0.85	0.39	0.53	0.86
RandomForest	RandomForest Default	0.87	0.27	0.41	0.84
(best model)	RandomForest Balanced	0.49	0.60	0.54	0.79
	RandomForest Default - Optimized	0.77	0.45	0.57	0.86

Hyperparameter tuning for RandomForests



Original hyperparameters:

- ✓*n_estimators=100*
- ✓*criterion='gini'*
- ✓*max_depth=4.*
- ✓*min_samples_leaf=3*

Optimized hyperparameters:.

- ✓*n_estimators: [100. 200. 300. 350. 400. 450. 500. 550. 600. 650. 700. 750]*
- ✓*criterion:['gini'. 'entropy']*
- ✓*max_depth: [1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12]*
- ✓*min_samples_leaf: [1. 3. 10. 15. 20. 25. 30. 35]*
- ✓*max_features: [1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12]*

Model parameters:

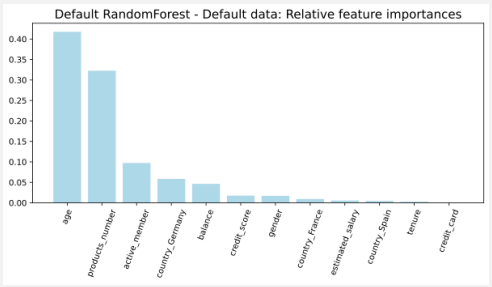
- ✓ *n_iter=50*
- ✓ *cv=StratifiedKFold*

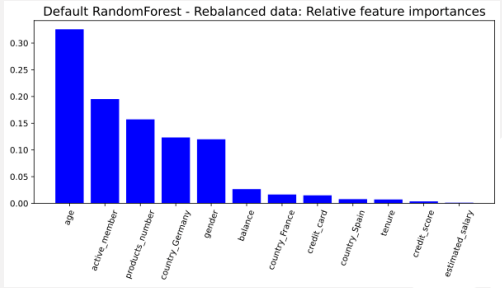
Feature importance before and after tuning

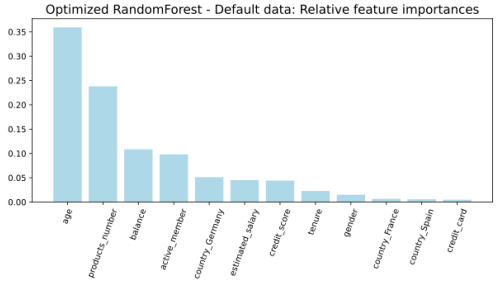
FEATURE RANKING	DEFAULT	
Age	0.417673	0.325745
Number of products	0.322676	0.157207
Active member	0.097335	0.195169
Country: Germany	0.058391	0.123344
Balance	0.046373	0.026674
Credit score	0.017599	0.003712
Gender	0.017117	0.119918
Country: France	0.009321	0.016623
Estimated salary	0.005662	0.001321
Country: Spain	0.004701	0.007928
Tenure	0.002737	0.007316
Credit card	0.000416	0.015042

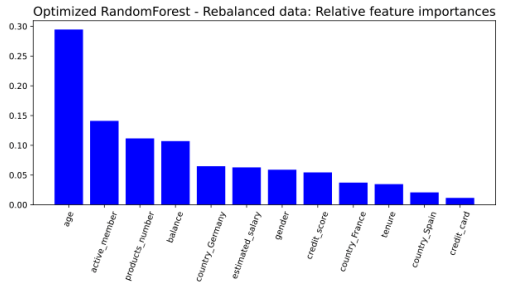
FEATURE RANKING	OPTIMIZED	
Age	0.359242	0.294691
Number of products	0.238068	0.111612
Active member	0.051312	0.141080
Country: Germany	0.108630	0.058968
Balance	0.098100	0.064822
Credit score	0.015054	0.011589
Gender	0.045146	0.037279
Country: France	0.005908	0.020835
Estimated salary	0.044225	0.107079
Country: Spain	0.022795	0.054432
Tenure	0.006572	0.034736
Credit card	0.004948	0.062876

- Since the metrics still had room for improvement, it is interesting to see how the importances of the features had evolved. As you can see from the highlighted cells there virtually now change in the importances.
- What has also stayed consistent is that 5 of the bottom do not seem to have significant impact on the predictions of the models.
- The most prominent features continue to be Age, Number of products, Balance and whether they are an active member or not.





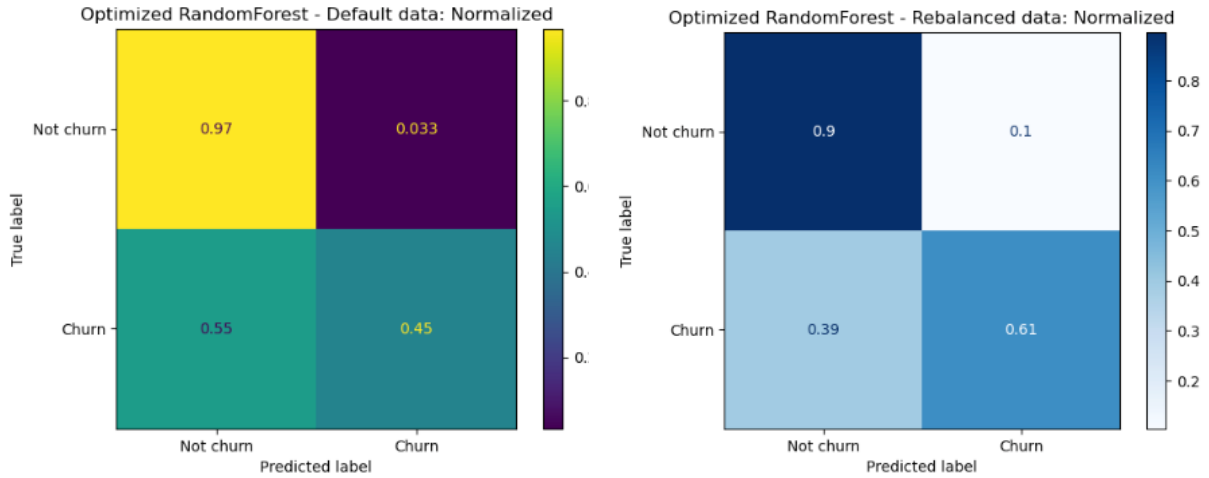


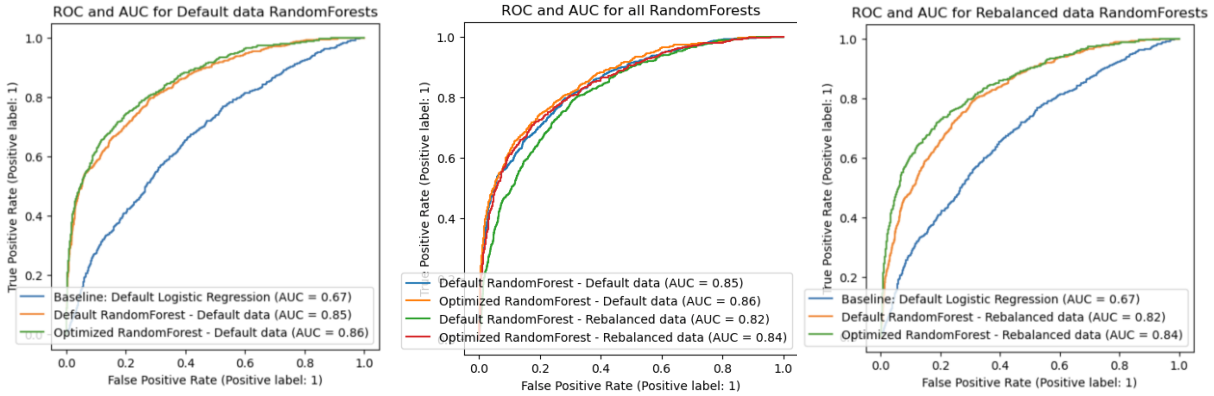


Evaluation of RandomForests after Hyperparameter tuning

DEFAULT DATA	precision	recall	f1-score	accuracy
Default RandomForest – Default Data	0.87	0.27	0.41	0.84
Optimized RandomForest – Default Data	0.77	0.45	0.57	0.86

REBALANCED DATA	precision	recall	f1-score	accuracy
Default RandomForest – Rebalanced Data	0.49	0.60	0.54	0.79
Optimized RandomForest – Rebalanced Data	0.60	0.61	0.60	0.84





- From the Classification Reports we see that with optimization of the hyperparameters we are able to make the performance of the models a little bit better.
- Accuracy for both default data and rebalanced data was already quite high so it is quite difficult to keep improving this. From ConfusionMatrices we can see that Accuracy is high because of high TrueNegative rate.
- From the ROC curves we can see the models are all performing better than our base model and that there are no significant changes in the performance even after hyperparameter tuning.
- Based on these metrics The model with rebalanced data and optimization would be the model we recommend. Although that model does not have the highest accuracy rate it does have the highest recall and f1-score.

Recommendations and Cost analysis

Preliminary information			
	Absolute value	**%**	
Total no. of customers	10 000		
Churn rate	2 000	20 %	Information bank
Cost of Churn	375,00 €		Money needed to spend to attain a new customer = savings
Cost marketing	- 75,00 €		
Aggregated cost of churn now	750 000,00 €		
Predictions from Model			
	Absolute value	**%**	
FalsePostitive rate	806	40,3 %	Lose money from marketing
TruePostive rate	1 194	59,7 %	Make saving for marketing if they take offer
TruePositive who take offer and not Churn	836	70,0 %	Assuming a 70% acceptance rate
Saving	313 470,87 €		
Cost of marketing	- 150 000,00 €		
Total saving	163 470,87 €		

- The bank is recommended to use **the model with optimized and rebalanced data to predict customer churn**
- With the model the bank can figure out which customers to target with marketing measures to prevent them from churning
- For the cost and benefit analysis, it is assumed that the acquisition cost of a new customer is 375.00€, the marketing cost to target a likely-to-churn customer is 75.00€ and that 70 % of customers who are sent targeted marketing will take the offer
- The cost and benefit analysis concludes that our model combined with targeted measures to "predicted churn" customers can provide **cost savings of 160,000 € in reduced acquisition costs**, as we manage to keep more customers