Churn prediction for bank customers using binary classification models

Yazeed Alzahrani

1 Introduction

In this project a dataset of a bank's customers was used to develop classification models that predict which customers will churn based on their data. The dataset contains data about 11 variables for 10,000 customers. One of these variables is the churn label which takes two possible values; 1 if the customer churned, and 0 otherwise. This is the target variable that we aim to predict.

The aims of this project are to develop binary classification models that predict customer churn and to determine which customer features have the greatest effect on the likelihood of churn.

2 Analysis

All the variables were plotted against each other in order to see patterns in the data. The scatter plots and histograms in Figures 1 and 2 show that age has one of the most significant impacts on customer churn. There is a clear distinction between the distributions of the ages of customers who churned and those who did not. The mean age of churned customers is 44.8 years compared with 37.4 years for customers that did not churn. Therefore we expect to see evidence that age plays a major role in predicting churn in the classification models. We also observe in Figures 1 and 3 that customers who churn are more likely to have a balance in the range 100,000 to 150,000. On the other hand, the credit score and estimated salary variables do not exhibit a similar predictive potential because their distributions do not differ significantly between churned and non-churned customers.

The variables *tenure* and *credit_card* (which indicates whether the customer has a credit card or not) were not found to have any notable correlations with the target variable. The plots of the remaining variables are presented in Figure 5. It is evident in Figure 5 that Germans are the most likely to churn. 48% of Germans churned compared with a 19.5% churn rate for non-Germans. Also, 33.5% of females churned, whereas 19.6% of males churned. The churn rate of inactive customers is 36.7%, compared with 16.6% for active customers. Moreover, customers who purchased more than 2 products overwhelmingly churned, but they only represent 3% of the total number of customers. Additionally, customers with 1 product are 4 times more likely to churn than customers with 2 products. Therefore, the variable *products_number* is expected to have the most predictive potential out of the four variables

in Figure 5.

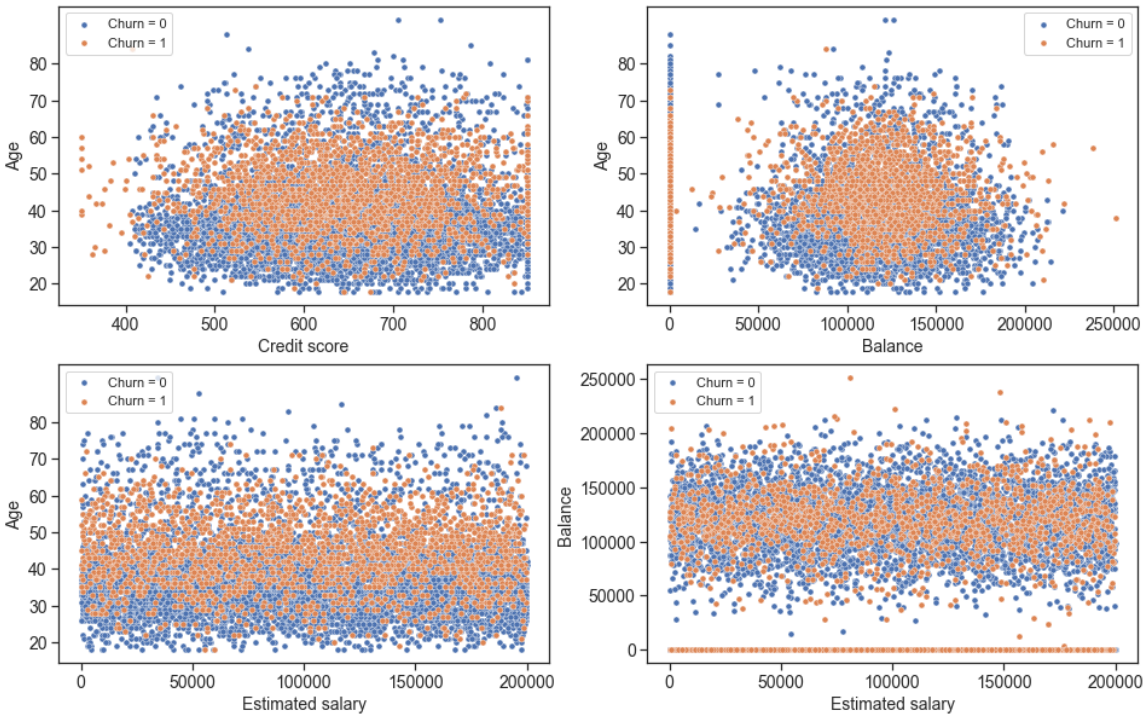

Figure 1: Scatter plots of some of the key variables of the dataset

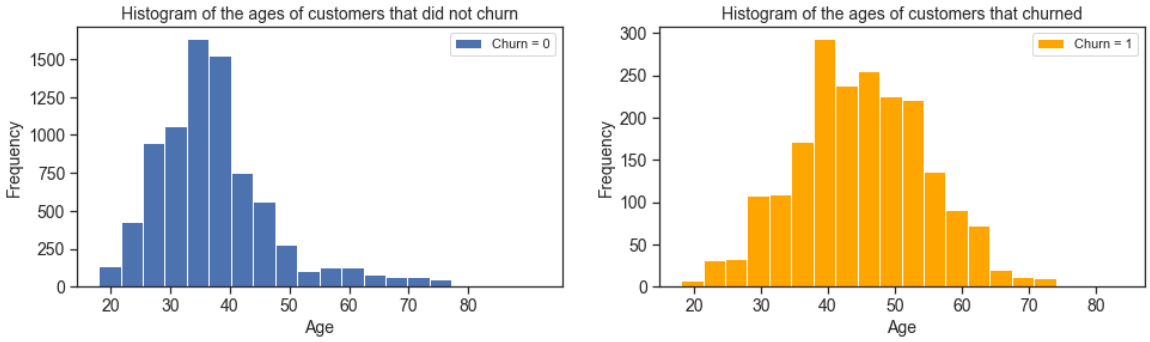

Figure 2: Histograms of the ages of customers

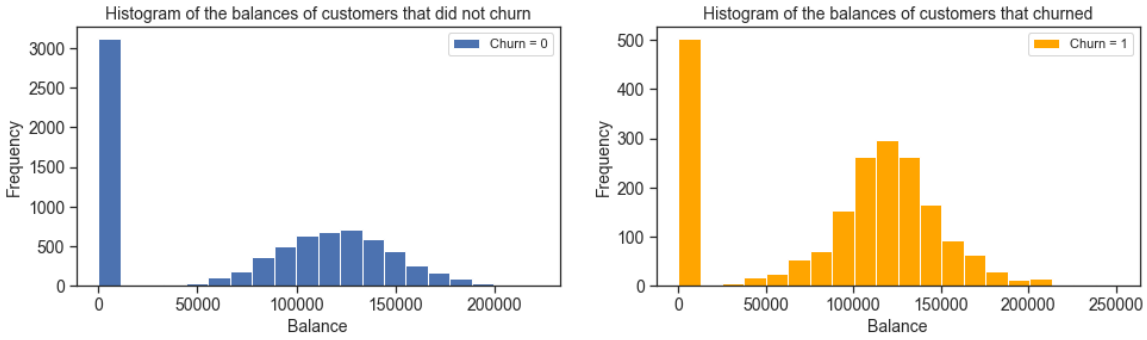

Figure 3: Histograms of the balances of customers

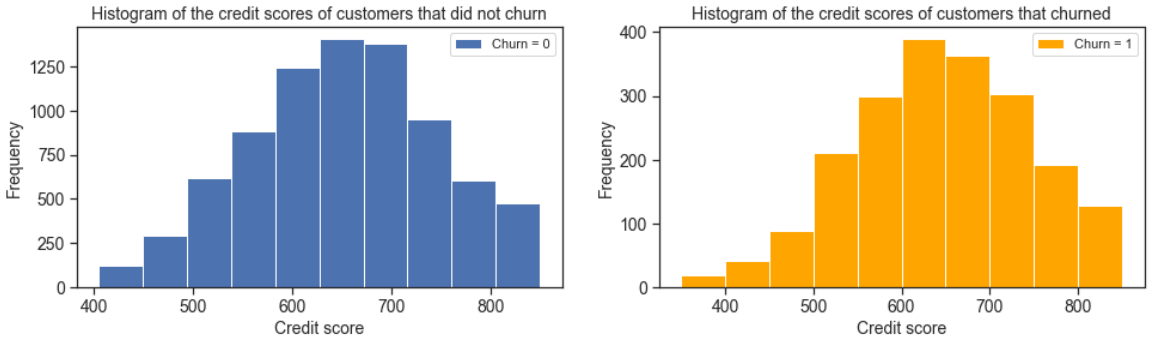

Figure 4: Histograms of the credit scores of customers

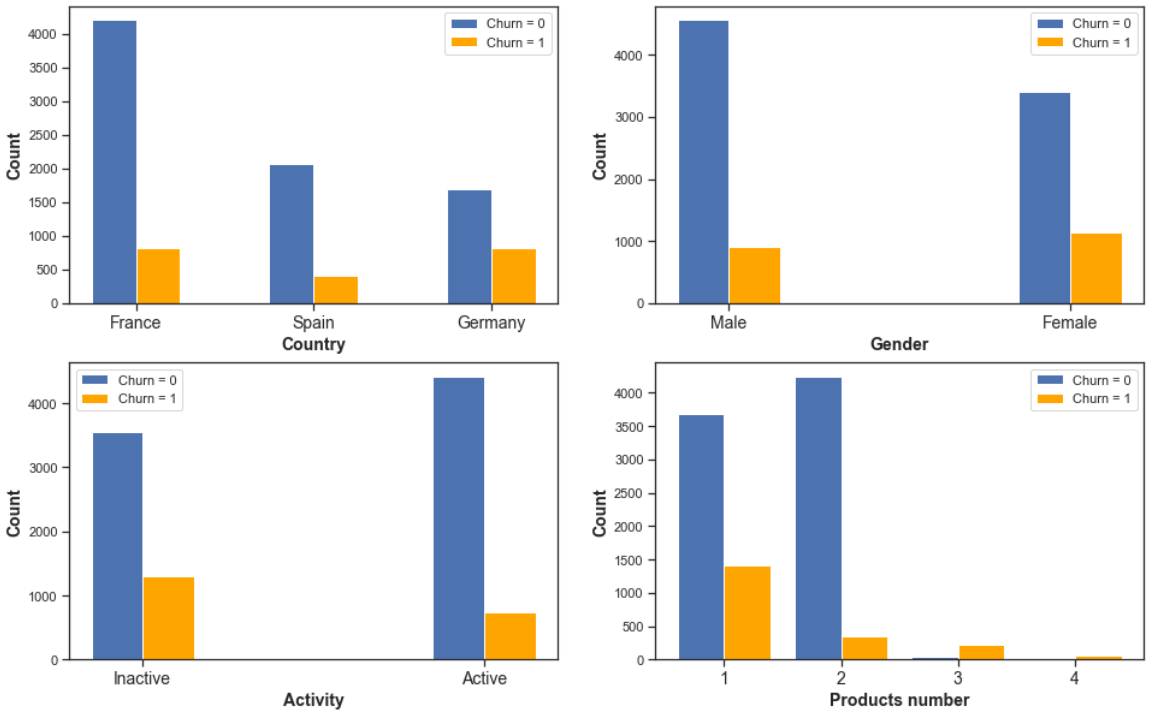

Figure 5: Bar graphs of the customers' count versus some key variables

3 Methods

3.1 Binary logistic ridge regression:

The first step in the process of implementing this method is standardising the data. Following that is the creation of the data matrix which has columns for each variable and a column of ones as its first column. K-fold cross-validation is then used to split the data into 5 equal folds, 4 folds for training the model and 1 fold to test it. The trained model is used to predict the class labels of the testing set and compare them with the true labels in order to calculate the classification accuracy. This process is then repeated four times, each time using a different fold as the testing set, and finally the optimal weights of the model and its true accuracy are estimated by averaging the weights and accuracies across all the folds.

To implement binary logistic regression we use the logistic function in Equation 1 which estimates the probability of occurrence of class label 1 for a data input $\mathbf{x}_i$ and a weights vector $\mathbf{w}$:

$$\sigma(\langle \mathbf{x}_i, \mathbf{w} \rangle) = \frac{1}{1 + e^{-\langle \mathbf{x}_i, \mathbf{w} \rangle}} \tag{1}$$

We then assign the predicted label a value of 1 if the output of the logistic function is greater than 0.5, and 0 otherwise. To find the optimal weights we use the gradient descent algorithm to minimise the cost function for binary logistic ridge regression given in Equation 2. This is because minimising the cost function maximises the likelihood of observing the true class label y_i for a given input $\mathbf{x}_i$.

$$L(\mathbf{w}) = \sum_{i=1}^{s} [\log(1 + e^{\langle \mathbf{x}_i, \mathbf{w} \rangle}) - y_i \langle \mathbf{x}_i, \mathbf{w} \rangle] + \frac{\alpha}{2} ||\mathbf{w}||^2 \tag{2}$$

$$\nabla L(\mathbf{w}) = \mathbf{X}^T(\sigma(\mathbf{X}\mathbf{w}) - \mathbf{y}) + \alpha \mathbf{w} \tag{3}$$

Where $\mathbf{X}$ is the data matrix, $\mathbf{y}$ is the true labels vector and α is the regularisation parameter. A grid search is performed to find the value of the regularisation parameter that maximises the classification accuracy.

3.2 K-nearest neighbours:

To apply this method we first standardise the data and prepare it by creating a Numpy array that consists of columns for the variables that will be included in the classification model, and rows for the data input for each customer. The data is then split into training and testing sets. After that the Euclidean distances between each row-vector (data input) in the testing set and all row-vectors in the training set are computed, and the label of the testing input is assigned according to the labels of the K nearest training inputs.

The estimated labels are compared with the true labels to obtain the classification error, and similar to the logistic regression method, K-fold cross-validation is performed to calculate the average error across all the folds. A grid search is also conducted to find the optimal number of neighbours that would minimise the classification error.

3.3 Support Vector Machine:

This classification method works by finding the optimal hyperplane that best separates the data points of two (or more) classes. When it is not possible to separate the data by a linear hyperplane, a method called the kernel trick is used to map the data into a higher-dimensional space where the data can be separated.

The scikit-learn library was used to perform the SVM method on the dataset. K-fold cross-validation was applied as before to estimate the true accuracy of the model. A grid search was performed to find the optimal regularisation parameter denoted by the letter C. This parameter controls how flexible the decision boundary is; a large C value can overfit the data and a smaller C value may result in underfitting.

4 Results

- **Binary logistic ridge regression:**

  When all the variables are selected for the model, the classification accuracy is 81.04% and the optimal regularisation parameter α is equal to 0. The optimal weights are given in Table 1.

Feature	Weight	Odds ratio
credit_score	-0.06	0.94
country_france	-0.12	0.89
country_spain	-0.09	0.91
country_germany	0.23	0.79
gender_male	-0.13	0.88
gender_female	0.13	1.14
age	0.76	2.14
tenure	-0.05	0.95
balance	0.16	1.18
products_number	-0.06	0.94
credit_card	-0.02	0.98
active_member	-0.54	0.58
estimated_salary	0.03	1.03

Table 1: Logistic regression results with all features selected.

  The odds ratio for a variable is calculated by taking the exponential of the weight of the variable. It represents the factor by which the odds of observing a class label 1 increase for a one unit increase in the variable.

  After removing the data inputs for which the *products_number* is greater than 2,

the classification accuracy rises to 84.11% and the odds ratio for *products_number* decreases from 0.94 to 0.46. This is because the vast majority of customers who purchased 3 and 4 products churned. Therefore, including the data inputs for these customers increases the odds ratio considerably. Consequently, this causes a reduction in the accuracy of the model because it over-represents those customers who constitute only 3% of the total number of customers.

When only the features with weights that have the highest absolute values were included in the model, it resulted in less accurate models. For example, when the selected features are *age*, *products_number* and *active_member* the classification accuracy is reduced to 83.64%.

Feature	**Weight**	**Odds ratio**
credit_score	-0.04	0.97
country_france	-0.15	0.86
country_spain	-0.11	0.89
country_germany	0.29	1.33
gender_male	-0.13	0.88
gender_female	0.13	1.14
age	0.74	2.10
tenure	-0.06	0.95
balance	-0.08	0.93
products_number	-0.78	0.46
credit_card	-0.03	0.97
active_member	-0.56	0.57
estimated_salary	0.01	1.01

Table 2: Logistic regression results with all features selected after adjusting the selection region for *products_number* and *credit_score*.

- **K-nearest neighbours:**

  This algorithm was executed in four cases with different feature selections. The results are presented in Table 3. The selected features are the ones that were found to have the greatest impact in the binary logistic regression model, with the exception of the last case. *estimated_salary* was chosen because it was expected to result in the lowest classification accuracy, which is indeed the case.

Selected features	**Classification accuracy (%)**
age	83.08
age, products_number	85.09
age, products_number, active_member	86.65
estimated_salary	80.90

Table 3: K-nearest neighbours results.

- **Support Vector Machine:**

SVM classification was performed for two cases. In the first case all the data features were selected, and the accuracy of the model was estimated to be 85.52%. In the second case the five features *country_germany*, *gender_male*, *age*, *products_number* and *active_member* which have the the highest absolute weights from the logistic regression model were selected. The resulting accuracy marginally increased to 85.88%.

5 Conclusions

K-nearest neighbours classification demonstrated the highest accuracy out of the three methods, although not by a significant margin. This may be because it has a better ability of capturing non-linearity in the data compared with logistic regression. Another possible reason is that KNN classification was less affected by outliers for this particular dataset. It was expected that SVM classification would achieve the highest accuracy, given its known excellent performance in classifying high-dimensional data. The reason this was not the case may be that the hyperparameter γ in the SVM classifier function was not tuned in conjunction with the regularisation parameter C. The parameter γ determines the reach of the training inputs on the decision boundary, with larger values of γ resulting in a more complex decision boundary and smaller values leading to a simpler boundary. A better accuracy may have been achieved if a grid search was conducted on C and γ together to find the combination that would give the highest possible accuracy.

The features' weights in the logistic regression model confirmed the initial assumptions about the predictive potential of some of those features. In particular, the weights of the age of the customer and the number of products purchased had the greatest magnitudes, meaning that they had the biggest impact on the churn likelihood. On the other hand, the weights of the estimated salary and credit card possession were negligible in comparison, suggesting little influence on the churn likelihood as was predicted by the initial data analysis.

6 References

scikit-learn (2024) 1.4. Support Vector Machines. Available at: https://scikit-learn.org/stable/modules/svm.html (Accessed 24 January 2024).