

1 Introduction

Churn prediction is a crucial task in customer relationship management, aiming to identify customers who are likely to discontinue using a service. Accurately predicting churn allows companies to take proactive measures to retain these customers. In this task, we address bank churn from customer data using machine-learning techniques.

2 Data description and preprocessing

The dataset contains a total of 10,000 samples, where each sample represents a unique customer. Each sample contains both structured and unstructured data. The structured data includes features such as the number of products the customer owns, their account balance, salary, age, gender, and other information. The dataset exhibits a significant class imbalance, which can affect the performance of machine-learning models. To illustrate the class imbalance, the left panel on Fig.1 shows the distribution of customers leaving (1) and not leaving (0) the service.

To prepare the data for analysis, we first encode the categorical features (gender and country) into numerical labels. The encoding process transforms each unique category into an integer label. This step is essential because classical machine-learning algorithms require numerical input.

The next steps involve handling the unstructured data and integrating it with the structured data. Customer feedback is a valuable source of unstructured data that provides insights into customer satisfaction and potential churn risk. We perform sentiment analysis to classify the feedback as very positive, positive, neutral, negative, and very negative employing the BERT (Bidirectional Encoder Representations from Transformers) model. The first step is tokenization with the `BertTokenizer`. It is used to convert the text of each feedback into a format suitable for BERT. This involves breaking the text into tokens and mapping each token to an integer index in the BERT vocabulary. The tokenized text is then passed to the `BertForSequenceClassification` module. This module is fine-tuned to classify each piece of feedback into one of five categories, corresponding to star ratings ranging from 1 star (the poorest score) to 5 stars (the best score).

Once each feedback has been classified into a star rating, we encode these ratings into integer labels. We also include a derived variable from the sentiment analysis that indicates how certain BERT is in classifying the customer's feedback. This variable can help mitigate the impact of errors from BERT's predictions.

Once the data is all preprocessed, with a format that algorithms can read, we can look at the correlation between different variables and the probability of customer churn. The right panel on Fig. 1 shows the correlation between all features in the dataset and their current status (Exited/not Exited). It indicates that customer feedback is an important feature in predicting churn. However, we also found instances where customers gave excellent ratings according to BERT (5 stars) but still left the service. Manual verification confirmed the high ratings, suggesting that despite positive feedback, these customers decided to churn. Conversely, some customers gave poor ratings (1 or 2 stars) but did not leave the service. These findings highlight the complexity of human behavior. While customer feedback is a relevant feature, it is not an infallible predictor of churn. Unexpectedly, the second most correlated feature with churn is age.

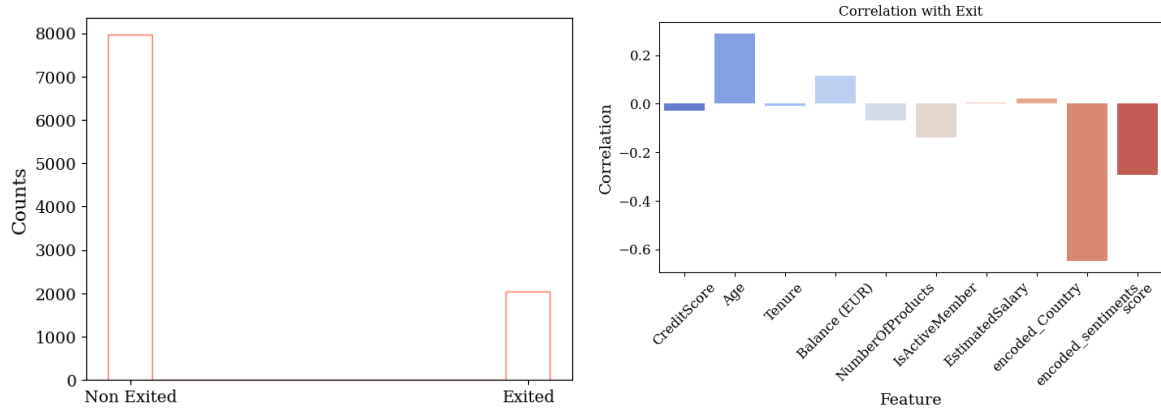


Figure 1: *Left*: Class distribution of the dataset (0: Non exited, 1:Exited). *Right*: Correlation between the features in the dataset and the exited outcome

3 Main results

To address the churn prediction task, we employed two machine learning approaches: Gradient Boosting and a Neural Network. Other algorithms were also tested, but Gradient Boosting provided the best results. Both algorithms were optimized using **Optuna**, a hyperparameter optimization framework. **Optuna** allows for efficient exploration of the hyperparameter space to find the best-performing settings for each model.

The performance of the models was evaluated using two primary metrics:

- **ROC Curve:** The Receiver Operating Characteristic (ROC) curve is a graphical representation of the true positive rate (sensitivity) versus the false positive rate (1-specificity) at various threshold settings. The area under the ROC curve (AUC) provides a single metric to evaluate the overall performance of the model, with higher values indicating better performance.
- **Confusion Matrix:** The confusion matrix provides a detailed breakdown of a model's performance by displaying the counts of true positives, true negatives, false positives, and false negatives. This matrix evaluates how well the model performs on each class. In the context of imbalanced datasets, understanding these metrics becomes critical since the model might achieve high overall accuracy by predominantly predicting the majority class while neglecting the minority class.

3.1 Gradient Boosting results

The Gradient Boosting model achieved an AUC of 0.91 (left panel on Fig.2). The confusion matrix (right panel on Fig.2) indicated the following performance:

- **True Negatives:** 95% of the customers who did not leave the service were correctly predicted.
- **True Positives:** 70% of the customers who left the service were correctly predicted.

3.2 Neural Network results

On the other hand, the Neural Network model yielded slightly worse results:

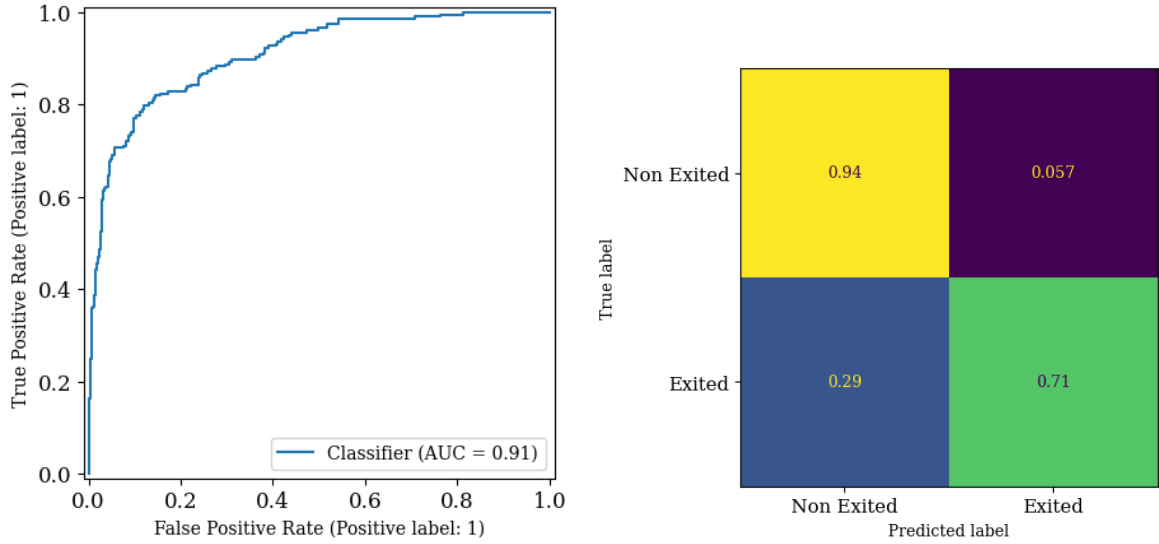


Figure 2: ROC Curve for Gradient Boosting

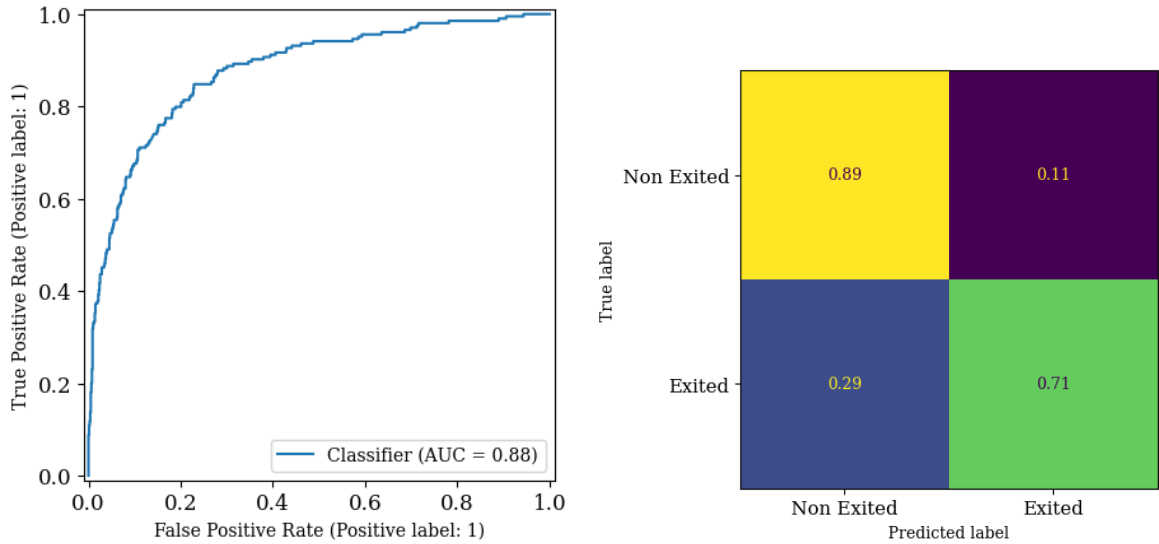


Figure 3: ROC Curve for Neural Network

- **AUC:** Approximately 0.86 (left panel on Fig.3).
- **True Negatives:** 90% of the customers who did not leave the service were correctly predicted (right panel on Fig.3)
- **True Positives:** 70% of the customers who left the service were correctly predicted.

The slightly lower performance of the Neural Network could be attributed to its sensitivity to dummy variables and the need for data normalization. Although normalization was performed, further optimization might improve results.

3.3 Discussion

Despite the overall performance of the models, churn prediction is not perfect. One significant challenge is the inherent unpredictability of human behavior. As previously noted, some customers with excellent feedback ratings still chose to leave the service, while others with poor

ratings decided to stay. This unpredictability introduces complexity that machine learning models may not fully capture.

Another factor contributing to the imperfection is the class imbalance in the dataset. Although methods like under-sampling, over-sampling, and bias adjustment for the neural network were applied to mitigate this issue, these techniques did not lead to a substantial improvement in the model's performance.

To enhance the accuracy and robustness of churn prediction, several avenues for future work are suggested:

- **Ensemble Methods:** Exploring ensemble methods that combine predictions from multiple models could potentially improve performance (bagging, boosting, and stacking).
- **Advanced Imbalance Handling:** Further investigation into advanced techniques for handling imbalanced data is warranted. This could include advanced cost-sensitive learning approaches.