M2 Quantitative Finance

Machine Learning

Customer churn prediction with ensemble classifiers and fairness analysis

Paolo Bortun

Academic Year: 2025–2026

Abstract

This study investigates customer churn prediction using a real-world dataset of 8,000 observations from a mobile phone operator. We compare several ensemble methods: Boosting such as Gradient Boosting and Adaptive Boosting (AdaBoost), Bagging including Random Forests and ExtraTrees, and Stacking. They are all evaluated through F-score and AUC, using 5-fold cross-validation for all the models, except for Random Forest where we use the Out-of-Bag estimate. Although Gradient Boosting and Stacking achieve the strongest discriminative performance, Random Forest emerges as the preferred operational choice due to its stability, interpretability, and lower computational cost. Beyond predictive accuracy, we analyse the fairness of the Random Forest with respect to the sensitive attribute *Gender*. We examine three fairness criteria: Independence (Demographic Parity), Separation (Equalized Odds), and Sufficiency (Calibration). The results reveal small but systematic disparities: women exhibit slightly higher predicted churn rates, receive more false positives, and show minor deviations in calibration at low probability levels. These findings underscore the importance of integrating fairness diagnostics into churn prediction pipelines, ensuring that high predictive performance does not come at the expense of unequal treatment across demographic groups.

Keywords: Supervised Learning; Classification; K-fold cross-validation; Out-of-Bag; Ensemble Learning; Gradient Boosting; AdaBoost; Random Forest; ExtraTrees; Stacking; F-score; AUC; ROC; fairness criteria.

Index

1 Introduction

Customer churn prediction is a central problem in data-driven decision-making for service providers such as mobile network operators. Churn prediction can be naturally formulated as a binary supervised learning task, where the aim is to predict whether a customer will leave the company ($Y = 1$) or remain ($Y = 0$) based on a set of socio-demographic and financial features. Previous studies have shown that machine learning models can effectively identify customers at risk of churn, enabling companies to allocate retention resources more efficiently.

This project addresses two main questions. First, among the different classification methods studied, we aim to determine which model should be recommended for accurate churn prediction, based on empirical performance metrics such as **F-score** and the **Area Under the Curve** (AUC). Second, as the variable *Gender* is a sensitive attribute, we investigate the fairness of the recommended model by evaluating the three fundamental group fairness criteria: **Independence** (demographic parity), **Separation** (equalized odds), and **Sufficiency** (calibration).

2 Data Exploration

In this section, we present the dataset preparation used for the churn prediction task. The data consist of socio-demographic and service-related customer information. Each observation corresponds to a single customer and includes both continuous and categorical features. The dataset provided for this study contains approximately $n = 8000$ customers described by $p = 10$ features, including *CreditScore*, *Geography*, *Gender*, *Age*, *Tenure*, *Balance*, *NumOfProducts*, *HasCrCard*, *IsActiveMember*, and *Salary*, together with the binary target variable *Churn*.

2.1 Previous analysis

First, we have a solid dataset, free of missing values, eliminating the need for imputation techniques. However, the target variable (*Churn*) shows a strong **class imbalance** (80% *Loyal* vs. 20% *Churn*). As a result, simple Accuracy could be misleading, making it necessary to use more robust metrics such as F-score and the AUC.

Second, **histogram** analysis confirmed that numerical variables contain no "absurd" values (e.g., negative ages or salaries) and follow realistic patterns: *Age* is roughly normally distributed, *Salary* appears uniform, and *CreditScore* shows a bell-shaped distribution typical of financial data.

Finally, the **correlation analysis** (**heatmap**) showed that the independent variables (features) are poorly correlated with each other (values mostly below 0.1). This suggests the absence of significant multicollinearity, which could otherwise introduce redundancy and instability in predictive models. Among all variables, *Age* emerged as the one most strongly correlated (positively) with the target variable Churn, suggesting that it could play a key role in subsequent modeling phases.

In summary, the data is ready for the subsequent phases of pre-processing and modeling, with the awareness that managing class imbalance will be a focal point for achieving reliable performance.

2.2 Treatment of categorical variables and anomalies

The dataset contains two features that require categorical encoding. The variable **Gender** was transformed into a binary numerical feature, assigning 1 to *Male* and 0 to *Female*. The variable **Geography** includes three nominal categories (*France*, *Spain*, *Germany*) and therefore has no intrinsic ordering. We apply **one-hot encoding** to convert it into binary indicators. The reference category (*France*) is omitted to avoid **multicollinearity** and ensure identifiability of linear models.

To ensure the quality of the dataset before training the predictive models, we applied the **Isolation Forest** algorithm for **anomaly detection**. Unlike traditional statistical methods that define outliers based on deviations from a mean or a known distribution, Isolation Forest explicitly isolates anomalies by leveraging their rarity and difference in the feature space. The core idea is that anomalies are few in number and have attribute values that are distinct from normal instances, making them easier to isolate in a random tree structure (they have shorter path lengths).

We configured the algorithm with a **contamination rate** of 0.01, assuming that approximately 1% of the dataset might consist of anomalous observations. The algorithm identified 80 anomalies out of the 8000 total observations. The results revealed that these data points do not appear to be corrupted data, instead they represent valid but rare customer profiles. For example, we observed customers with: a maximum credit score of 850, holding the maximum number of products, and very high balances or salaries combined with specific geographic locations. These anomalies represent legitimate edge cases, and removing them could bias the model and harm generalization. Therefore, all 80 were retained to allow the models to learn from these rare but valid customer behaviours.

3 Methodology

In this section, we present the modelling framework adopted to address the churn classification task. Supervised classification methods can be grouped into several major families according to their modelling assumptions and learning strategies. We briefly review the main families relevant to this problem and justify the choice of the methods used in this project.

First, **Parametric methods** such as Logistic Regression, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) rely on explicit probabilistic assumptions about the data. They typically assume linear or quadratic decision boundaries and are easy to interpret, but their performance deteriorates when the true relationship between the predictors and the target is highly non-linear or when distributional assumptions are not satisfied. Similarly, **Gaussian Naive Bayes** applies Bayes' theorem under the strong assumption that features are conditionally independent given the class label. Therefore, we do not rely on these type of classifiers because their modelling assumptions are not well aligned with the nature of the churn prediction problem.

Next, **Classification Tree** methods recursively partition the feature space into homogeneous regions by applying binary splits. Decision trees are highly interpretable and capture non-linear interactions between features, but they tend to overfit and are therefore unstable on their own. To overcome the limitations of decision trees, we use **ensemble methods**, which combine multiple learners to reduce variance, correct bias, or integrate different models, achieving stronger predictive performance. In this project, we focus on the main ensemble families: **Boosting**, **Bagging** and **Stacking**.

On the one hand, **Boosting** is a *sequential ensemble method* designed to turn *weak learners* into a *strong learner*. Boosting exploits the dependence between the base learners, it builds its learners sequentially: each new learner focuses on the observations that were misclassified by the previous ones. This sequential dependence reduces the bias of the estimator. In this project, we consider two major Boosting algorithms: **Gradient Boosting (GB)** and **Adaptive Boosting (AdaBoost)**.

Gradient Boosting builds a predictive model sequentially, where each new tree is trained to correct the errors made by the previous ones. Instead of fitting all trees independently, the algorithm uses the negative gradient of the loss function to determine how the model should adjust to reduce error. Each tree learns to approximate these residuals, focusing on observations that the current model predicts poorly. By adding many small corrective trees, GB performs a form of functional gradient descent that gradually captures complex non-linear relationships.

AdaBoost works by training a sequence of weak classifiers, where each new classifier focuses increasingly on the observations that previous classifiers misclassified. At every iteration, AdaBoost updates a set of weights on the training data: misclassified points receive higher weight, while correctly classified points receive lower weight. The final model is a weighted combination of all weak learners, where more accurate learners receive larger influence. By sequentially correcting past mistakes, AdaBoost effectively reduces bias and builds a strong classifier from simple, weak base models.

On the other hand, **Bagging** (Bootstrap Aggregating) is a *parallel ensemble method* that improves predictive accuracy by training several base learners independently on different bootstrap samples of the data. In classifciation problems, their outputs are aggregated by **majority voting** to produce a final prediction. By exploiting the approximate independence of the learners, the variance reduces, making it particularly effective for inherently unstable models such as decision trees. In this project, the main Bagging methods considered are **Random Forest** and **ExtraTrees**.

In a **Random Forest**, each tree is trained on a bootstrap sample drawn with replacement from the original dataset. Contrary to standard bagging, RF introduces an additional source of randomness: at each split of each tree, only a random subset of features is considered for splitting. Since each tree uses different data (**bootstrap**) and different features (**feature subsampling**), trees become less correlated. Averaging uncorrelated trees leads to a strong variance reduction without increasing the bias too much. As a result, Random Forest models typically achieve **lower variance** than single decision trees and **lower bias** than shallow trees or *weak learners*.

In **ExtraTrees**, randomness is pushed further to increase tree diversity. Unlike Random Forests, trees are grown on the entire dataset (no bootstrap), and at each split both the subset of features and the split thresholds are chosen at random. Among these randomly generated thresholds, the algorithm selects the one that gives the best impurity reduction. This heavier randomization makes ExtraTrees faster to train and provides a strong regularizing effect, while the final averaging step stabilizes the predictions despite the noisier individual trees.

Finally, **Stacking** is an ensemble learning technique that combines several base learners through a meta-model. Unlike Bagging and Boosting, which aggregate models of the same family through averaging or sequential reweighting, Stacking integrates predictions from heterogeneous models in order to exploit their complementary strengths.

4 Model evaluation

In this section, we outline the framework used to assess the predictive quality of the models. We first apply appropriate **validation techniques** to ensure reliable generalisation, and then evaluate the classifiers through key **performance criteria**.

4.1 Model validation techniques

The goal of supervised learning is **generalization**, so a classifier must not only fit the training data, but also maintain a similar level of performance on unseen data. In other words, a model must work **without overfitting**. To guarantee this property, a systematic validation procedure is required.

4.1.1 K-fold cross-validation

Cross-validation (CV) is a widely used method for assessing a model's ability to generalise. It relies on splitting the dataset into two disjoint parts: a training set, used to fit the model and perform model selection, and a test set, reserved for an unbiased evaluation of its predictive performance. As this split may introduce a bias depending on how data fall into the train or test subsets, we partition the dataset using **K-fold CV** with $K = 5$, where the dataset is split into K different subsets (*folds*). In each iteration, one fold is used as the test set while the remaining folds form the training set. Repeating this procedure K times ensures that every observation is evaluated once, and the K performance estimates are then averaged to obtain a stable measure of the model's generalisation ability.

4.1.2 Out-of-Bag

An important feature of RF is the ability to estimate their generalisation error without the need for a separate validation set. The **Out-of-Bag (OOB)** error, which is obtained by evaluating each observation on the trees for which it was not included in the bootstrap sample.

4.2 Performance criteria

4.2.1 F-score

The **F-score** or **F1-score** is a metric designed for imbalanced datasets, as it focuses on the minority class by combining **Precision** (how many predicted churners are truly churners) and **Recall** (how many real churners are detected). It is high when the model both identifies churners correctly and avoids excessive false alarms. This makes it especially valuable in churn prediction, where the goal is to assess how well the model recognises customers likely to leave despite the majority being loyal.

4.2.2 AUC - ROC curve

The **Receiver Operating Characteristic** (ROC) curve provides a threshold-independent evaluation of a classifier. It represents the trade-off between the *True Positive Rate* (TPR) (called *Sensitivity*), and the *False Positive Rate* (FPR). The AUC summarizes the entire curve into a single scalar. It measures the overall predictive power of the model and it can be interpreted as the probability that the classifier assigns a higher score to a randomly chosen positive instance than to a randomly chosen negative one. Since the ROC curve and AUC don't depend on the class proportions, this criterion is particularly suitable for imbalanced problems such as churn prediction.

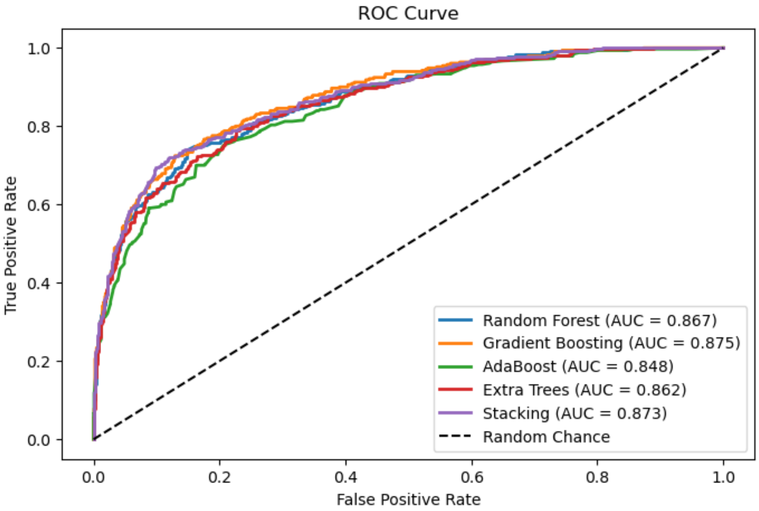
Table 1 presents the performance of the different ensemble methods. Gradient Boosting achieves the highest AUC and shows strong cross-validated results, confirming its effectiveness at ranking customers by churn risk. Stacking also proves strong predictive ability, achieving a AUC close to GB. Random Forest obtains the highest F-score, indicating that it is the most effective at correctly identifying churners while maintaining a good balance between precision and recall. ExtraTrees also perform competitively, offering solid F-scores and AUC values close to those of GB. AdaBoost performs reliably but remains below the other methods, consistent with its sensitivity to noisy or complex patterns.

Table 1: Performance comparison of classification models.

Model	F-score	AUC	Validation
Gradient Boosting	0.563	0.875	5-fold CV: 0.8599
Stacking	0.603	0.873	5-fold CV: 0.8507
Random Forest	0.627	0.867	OOB score: 0.8559
ExtraTrees	0.610	0.862	5-fold CV: 0.8507
AdaBoost	0.545	0.848	5-fold CV: 0.8559

Figure 1 shows the ROC curves for the three models. Gradient Boosting dominates across most thresholds, staying closer to the upper-left corner and confirming its superior discriminative ability. Random Forest also performs strongly, although slightly below Gradient Boosting. AdaBoost remains competitive but shows a more modest improvement over random guessing. Taken together, these results point to Gradient Boosting as the most reliable overall classifier in this setting.

Figure 1: ROC curves for all classification models.



Overall, churn prediction is a **high-dimensional** and **highly non-linear** classification problem, driven by heterogeneous predictors (combining continuous, discrete and categorical variables) and complex interactions across demographic, behavioural, and usage variables, making classical models unsuitable. In contrast, non-linear ensemble methods such as Gradient Boosting, Stacking and Random Forest are specifically designed to capture this structure.

Gradient Boosting attains the highest AUC, but requires careful hyperparameter tuning, is more sensitive to noise, and behaves as a black-box model whose decisions are difficult to interpret. Stacking shares this limited interpretability, as its meta-learner combines several complex base models, making the overall decision process less transparent. In contrast, Random Forest is more robust and stable: it works well with minimal tuning, provides an unbiased generalisation estimate through the OOB score, and offers clearer variable importance and interpretability thanks to its independently grown trees. For these reasons, Random Forest represents a more reliable and practical model.

4.3 Feature Importance

An attractive property of RF is their ability to quantify how much each predictor contributes to the model's performance. In this analysis, we rely on **impurity-based** variable importance, a metric that evaluates how much each variable reduces node impurity (Gini index in our case) across all splits and all trees in the forest. Although this method is computationally efficient and provides an interpretable ranking of predictors, it may slightly favour variables with many possible split points.

The feature importance results obtained from the Random Forest model are fully consistent with the correlation matrix. *Age*, which shows the strongest linear relationship with Churn ($r = 0.27$), is also the most influential predictor in the model (importance = 0.3139). Balance exhibits the second-highest correlation with the target ($r = 0.12$), which is reflected in its high importance score. *NumOfProducts*, despite showing only a weak linear correlation ($r = –0.04$), emerges as the second most important variable. This mismatch highlights the presence of strong non-linear effects: customers with a single product churn disproportionately more, whereas those with two products rarely churn. Such threshold behaviours are not captured by correlation but are fully exploited by tree-based models. Finally, *Salary* and *CreditScore* display near-zero correlation with Churn but still contribute to the model by improving splits through interaction effects. This again confirms that Random Forest importance captures more complex patterns than simple pairwise correlations.

5 Fairness analysis

In supervised classification, fairness concerns arise when predictions may systematically disadvantage particular social groups. Then, after selecting the best-performing classifier, we evaluate its fairness with respect to the sensitive attribute **Gender** with three complementary fairness criteria.

5.1 Independence (demographic parity)

The **Independence** criterion requires that the decision of the classifier be statistically independent of the sensitive attribute. Formally, a classifier satisfies Demographic Parity if:

$$P(\hat{Y} = 1 \mid \text{Male}) = 0.167, \qquad P(\hat{Y} = 1 \mid \text{Female}) = 0.267. \tag{1}$$

where $\hat{Y}$ is the predicted class. We estimate these probabilities by computing the proportion of positive predictions within each gender group. The model predicts churn = 1 almost twice as often for female customers, indicating a lack of demographic parity.

5.2 Separation (equalized odds)

The **Separation** criterion evaluates whether the classifier exhibits similar error rates across groups. A classifier satisfies Equalized Odds if both the TPR and the FPR are equal across sensitive groups:

$$\text{TPR}_{\text{Male}} = 0.61 \neq \text{TPR}_{\text{Female}} = 0.658 \quad \text{and} \quad \text{FPR}_{\text{Male}} = 0.079 \neq \text{FPR}_{\text{Female}} = 0.134. \tag{2}$$

The results show that the classifier does not satisfy the Separation criterion (Equalized Odds). While the True Positive Rate (TPR) is similar across groups, indicating equal opportunity in identifying churners, there is a significant disparity in the False Positive Rate (FPR). Specifically, women receive nearly twice as many false alarms (13.4%) as men (7.9%), revealing that prediction errors are not distributed uniformly across genders (violation of Predictive Equality).

5.3 Sufficiency (calibration by group)

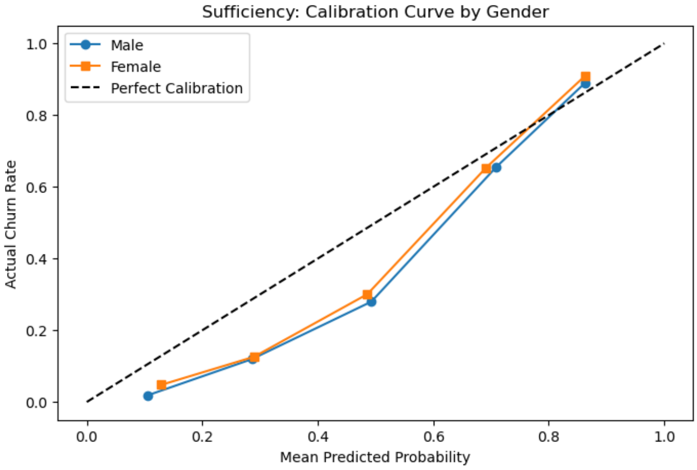
Sufficiency requires that the predicted probability scores be equally informative across groups. A classifier satisfies this criterion if:

$$P(Y = 1 \mid \hat{P} = p, S = \text{Male}) = P(Y = 1 \mid \hat{P} = p, S = \text{Female}), \tag{3}$$

for all probability levels p, where $\hat{P}$ denotes the predicted churn probability. To assess this, we compute *calibration curves* separately for Male and Female customers.

Figure 2 compares predicted probabilities with observed churn rates across bins for male and female customers. At low predicted probabilities, both groups show a slight deviation from the diagonal, indicating some degree of underestimation of the true churn rate. However, as the predicted probability increases, the calibration curves for males and females converge and closely track the line of perfect calibration. The two curves remain very close to each other throughout the entire range, and no systematic separation between gender groups is observed. Overall, the classifier is reasonably well calibrated and largely satisfies the Sufficiency criterion.

Figure 2: Calibration curve for the Random Forest model, showing the alignment between predicted churn probabilities and observed frequencies for Male and Female customers.



6 Conclusions

This project analysed a dataset of 8,000 customers described by ten socio-demographic and service-related variables with the goal of predicting customer churn. We compare several ensemble methods: Boosting such as Gradient Boosting and Adaptive Boosting (AdaBoost), Bagging including Random Forests and ExtraTrees, and Stacking, using F-score and AUC as performance metrics. All models were evaluated through stratified 5-fold cross-validation, except for Random Forest, which was assessed using its unbiased Out-of-Bag estimate.

The empirical results show that Gradient Boosting and Stacking achive the highest AUC, and its ROC curve confirms its strong discriminative ability. However, Random Forest, offers several practical advantages: it performs well with minimal hyperparameter tuning, is stable and robust due to the aggregation of many independent trees, handles noisy or irrelevant features effectively, and is computationally efficient thanks to its parallel structure. Although still a black-box model, it provides more transparent global interpretability through tools such as feature importance. For these reasons, **Random Forest** emerges as the most practical model for deployment.

Finally, the fairness analysis reveals moderate but systematic disparities across gender groups, Random Forest does not simultaneously satisfy Independence, Separation, or Sufficiency. This underscores the importance of complementing predictive performance with fairness diagnostics, ensuring that churn prediction systems operate not only effectively but also equitably across demographic groups.