Predicting Customer Churn with Regression-Based and Tree-Based Methods

Baybayon, Darlyn Antoinette B. & Mayol, Jose Raphael J.

Submitted to: Gella, Frederick
Received on the 20th of May 2025

1. Introduction

In the telecommunications industry, customer churn—the phenomenon of customers discontinuing their services—is a persistent and costly challenge. With the market reaching saturation and competition intensifying, retaining existing customers has become more critical than ever. Research consistently shows that acquiring new customers is significantly more expensive than keeping current ones (Albérico & Casaca, 2024), making churn prediction an essential strategy for maintaining profitability and market share.

To address this challenge, many telecom companies have turned to data-driven methods to understand customer behavior and identify early warning signs of churn. Machine learning, in particular, offers powerful tools to uncover complex patterns in customer usage data that may not be apparent through traditional analysis. By leveraging historical customer activity and service interaction records, these models can predict the likelihood of churn and support timely, targeted interventions to retain high-risk customers.

This study uses the Orange Telecom Churn Dataset, which contains detailed customer behavior metrics along with churn labels indicating whether a customer has canceled their subscription. The dataset is split into two parts: a larger training set (churn-80) used for developing and cross-validating predictive models, and a smaller test set (churn-20) used for final evaluation. The dataset provides a rich array of features, including call usage during different times of day, number of customer service interactions, and international call patterns.

The primary objective of this study is to build and evaluate machine learning models that accurately predict customer churn. Both regression-based approaches (logistic, ridge, and lasso regression) and tree-based models (decision tree, random forest, and gradient boosting) are explored and compared in terms of predictive performance, interpretability, and real-world applicability. The insights generated from this study can help telecom companies proactively identify at-risk customers and implement effective retention strategies, ultimately reducing churn and enhancing customer satisfaction.

2. Methodology

2.1 Research Design

This study adopts an applied machine learning approach to build, train, and evaluate predictive models for customer churn. The goal is to compare the performance of various classification algorithms and determine the most effective model for accurately identifying customers at risk of discontinuing their telecom services. Both regression-based and tree-based classification models are considered to provide a balanced assessment of model performance, interpretability, and practical applicability.

2.2 Dataset Description

The study uses the Orange Telecom Churn Dataset, which includes anonymized data on customer usage behavior and churn status. The dataset is divided into two subsets:

- **churn-80**: Contains 80% of the data and is used for training and cross-validation.
- **churn-20**: Contains the remaining 20% and is reserved for final model testing and performance evaluation.

Each record includes features such as call durations (day, evening, night, and international), charge costs, number of customer service calls, and other service usage metrics, along with a binary target variable (Churn) indicating whether the customer left the service.

2.3 Data Preprocessing

Prior to model development, several preprocessing steps were applied:

- **Conversion of categorical variables**: All categorical variables were converted into dummy/indicator variables using one-hot encoding.
- **Feature scaling**: For regression-based models, feature scaling was applied where necessary.
- **Label encoding**: The target variable Churn was encoded as a binary factor (0 = no churn, 1 = churn).
- **Matrix transformation**: For models requiring matrix input (e.g., glmnet and xgboost), design matrices were created using model.matrix().

2.4 Exploratory Data Analysis

Following preprocessing, an exploratory data analysis (EDA) was conducted to understand the underlying structure and characteristics of the dataset. This step helped identify trends, distributions, and relationships that could inform model development and interpretation.

2.4.1 Descriptive Statistics and Visualizations

Key variables were summarized using descriptive statistics. Visualizations such as histograms and density plots were used to inspect the distribution of numerical features like churn rate, total call minutes (day, evening, night, and international), and service usage. Churn rate was calculated as the proportion of customers who discontinued service, providing a baseline understanding of the target variable's distribution. The churn rates for the training and test sets were 14.55% and 14.24%, respectively. Histograms and density plots for the total call minutes were also taken, and all were normally distributed. Service usage bar plots also showed that Area 408 had 669 observations, Area 415 had 1318, and Area 510 had 679 observations.

2.4.2 Correlation Analysis

A correlation matrix was generated to assess the linear relationships among numerical variables. High correlations were observed between variables like total minutes and total charges across different time periods, as expected due to actual billing practices. This would be incredibly detrimental to the model-building process, hence one of the variables must be dropped. Call minutes offer more valuable information and thus charge was eventually removed from the dataset.

2.4.3 Class Imbalance Check

The dataset was examined for class imbalance, revealing a disproportionate number of non-churning customers compared to churners. Specifically, the majority class (non-churn) comprised approximately 86% of the dataset. This imbalance was taken into account during model evaluation by placing emphasis on metrics such as sensitivity, specificity, F1-score, and AUC—beyond accuracy—to ensure models performed well on the minority class (churn).

2.4.4 Model Development

The following methods were developed and tuned:

Regression -Based Models

- **Logistic Regression:** A baseline model to assess linear relationships between predictors and churn. Class weights were used to address the imbalanced dataset.
- **Ridge Regression:** Utilized L2 regularization to address multicollinearity among predictor variables.
- **Lasso Regression:** Utilized L1 regularization to perform feature selection by shrinking irrelevant coefficients to zero.

Hyperparameters for ridge and lasso regression models were optimized using cross-validation with *cv.glmnet*.

Tree-Based Models

- **Decision Tree:** Implemented using the *rpart* package to produce an interpretable model based on recursive binary splits.
- **Random Forest:** A method that builds multiple decision trees and aggregates their outputs, lessening the variation.
- **Gradient Boosting:** A sequential ensemble technique that builds trees to correct previous errors, offering superior accuracy and handling of class imbalance.

For the tree-based models, hyperparameters such as tree depth, learning rate, and number of trees were tuned using grid search and early stopping criteria where applicable.

2.4.5 Model Evaluation

All models were evaluated using the *churn-20* test set. The following performance metrics were calculated to ensure a comprehensive assessment.

- **Accuracy:** Overall correctness of the model.
- **Sensitivity (Recall):** True positive rate for the non-churning class.
- **Specificity:** Ability to correctly identify churning customers.
- **Precision (Positive Predictive Value or PPV):** Proportion of predicted non-churns that were correct.
- **Balanced Accuracy:** Measures the average of sensitivity and specificity.
- **F1-Score:** Harmonic mean of precision and recall.
- **AUC (Area Under the ROC Curve):** Measure's the model's ability to discriminate between classes.

- **Confusion Matrix:** Summarized model predictions against actual labels.

2.4.6 Feature Importance Analysis

To improve model interpretability and provide actionable business insights, feature importance was examined for the tree-based models. This was computed using gain-based and frequency-based importance metrics. Key predictors identified included:

- **Total.day.minutes**
- **Total.eve.minutes**
- **Customer.service.calls**
- **Total.intl.minutes**

These features were interpreted as critical drivers of churn behavior and offer practical implications for customer retention strategies.

3. Results

This section presents the performance metrics of each model applied to the test dataset.

3.1 Logistic Regression

Figure 1. Logistic Regression Model Summary

```
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)           -3.7375767  0.4362936  -8.567  < 2e-16 ***
Account.length         0.0007981  0.0008827   0.904 0.365890
Area.code415          -0.0952939  0.0883866  -1.078 0.280968
Area.code510          -0.1299510  0.1012936  -1.283 0.199522
Number.vmail.messages  0.0401957  0.0108741   3.696 0.000219 ***
Total.day.minutes      0.0123277  0.0006464  19.070  < 2e-16 ***
Total.day.calls        0.0024739  0.0017370   1.424 0.154369
Total.eve.minutes      0.0054218  0.0007214   7.516 5.64e-14 ***
Total.eve.calls        0.0004061  0.0017745   0.229 0.818989
Total.night.minutes    0.0028918  0.0007343   3.938 8.21e-05 ***
Total.night.calls      0.0007615  0.0018234   0.418 0.676239
Total.intl.minutes     0.0787050  0.0131523   5.984 2.18e-09 ***
Total.intl.calls      -0.0854122  0.0146482  -5.831 5.51e-09 ***
Customer.service.calls 0.5822330  0.0262470  22.183  < 2e-16 ***
International.plan_No -2.4104429  0.1131892 -21.296  < 2e-16 ***
Voice.mail.plan_Yes   -2.0061683  0.3437203  -5.837 5.33e-09 ***
Voice.mail.plan_No            NA         NA      NA       NA
International.plan_Yes        NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 1 shows the coefficients of the logistic regression model. From the model summary, there are multiple predictors which are highly significant for the Churn equation. These are namely *Number.vmail.messages*, *Total.day.minutes*, *Total.eve.minutes*, *Total.night.minutes*, *Total.intl.minutes*, *Total.intl.calls*, *Customer.service.calls*, *International.plan*, and *Voice.mail.plan*.

Table 1. Logistic Regression Model Confusion Matrix & ROC Curve Statistics

Logistic Regression					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.7406	0.7290	0.8105	0.9586	0.8282	0.831
Balanced Accuracy		0.7698			

Figure 2. Logistic Regression Model ROC Curve

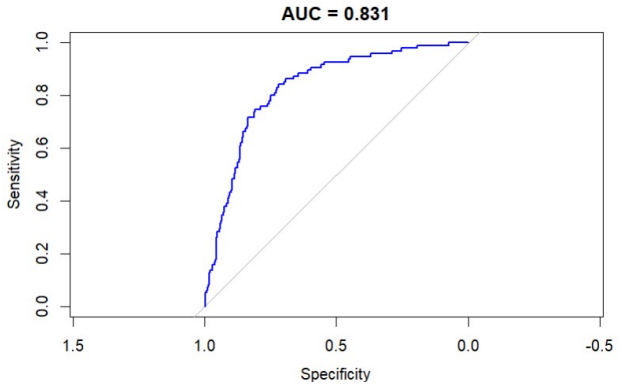

Table 1 and Figure 2 show the confusion matrix and ROC Curve statistics of the logistic regression model. The model achieved an accuracy of 74.06% with a balanced accuracy of 76.98% and an AUC of 0.831. While the model's sensitivity was relatively high at 72.90%, its specificity was also strong at 81.05%. However, the model struggled with negative predictive value (33.19%), indicating difficulty in correctly identifying customers who would not churn. The F1-score was 0.8282.

3.2 Ridge Regression

Table 2. Ridge Regression Model Confusion Matrix & ROC Curve Statistics

Ridge Regression					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.8546	0.9825	0.08421	0.86595	0.9206	0.764
Balanced Accuracy		0.5334			

Figure 3. Ridge Regression Model ROC Curve

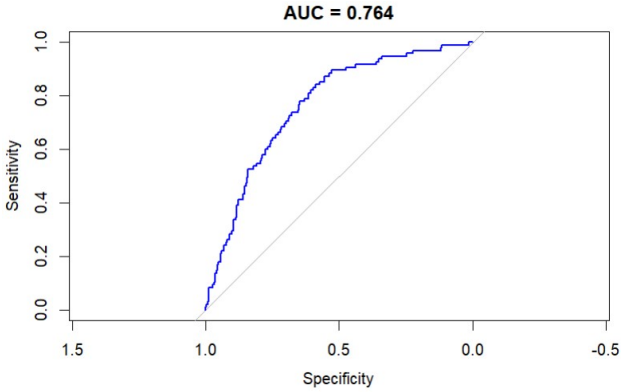

Table 2 and Figure 3 show the confusion matrix and ROC Curve statistics of the ridge regression model. The model showed a slightly improved accuracy of 85.46% but a significantly reduced balanced accuracy of 53.34%, due to a very low specificity of 8.42%. While the model demonstrated strong sensitivity (98.25%) and a high F1-score of 0.9206, it frequently misclassified non-churners as churners, reflecting

a bias toward the majority class. The AUC was 0.764, indicating moderate discriminatory ability.

3.3 Lasso Regression

Table 3. Lasso Regression Model Confusion Matrix & ROC Curve Statistics

Lasso Regression					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.8561	0.9773	0.1263	0.8707	0.9209	0.782
Balanced Accuracy		0.5518			

Figure 4. Lasso Regression Model ROC Curve

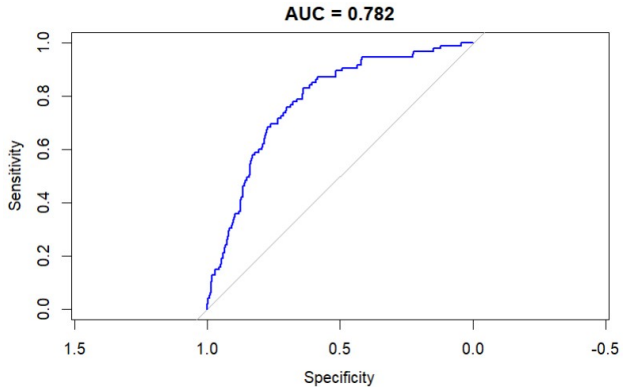

Table 3 and Figure 4 show the confusion matrix and ROC Curve statistics of the lasso regression model. The model performed similarly to ridge regression with an accuracy of 85.61%, specificity of 12.63%, and a balanced accuracy of 55.18%. Sensitivity remained high at 97.73%, but specificity was again low at 12.63%. The F1-score reached 0.9209, and the AUC was 0.782.

3.4 Decision Tree

Table 4. Decision Tree Model Confusion Matrix & ROC Curve Statistics

Decision Tree					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.9565	0.9843	0.7895	0.9657	0.9749	0.894
Balanced Accuracy		0.8869			

Figure 5. Decision Tree Model ROC Curve

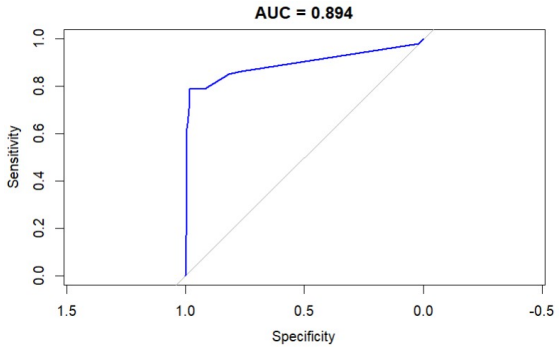

Table 4 and Figure 5 show the confusion matrix and ROC Curve statistics of the decision tree model. The model provided significant improvements, achieving an accuracy of 95.65% and a balanced accuracy of 88.69%. Sensitivity was high at 98.43%, while specificity reached 78.95%. The model also yielded a strong F1-score of 0.9749 and an AUC of 0.894.

3.5 Random Forest

Table 5. Random Forest Model Confusion Matrix & ROC Curve Statistics

Random Forest					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.949	0.9948	0.6737	0.9483	0.9710	0.931
Balanced Accuracy		0.8342			

Figure 6. Random Forest Model ROC Curve

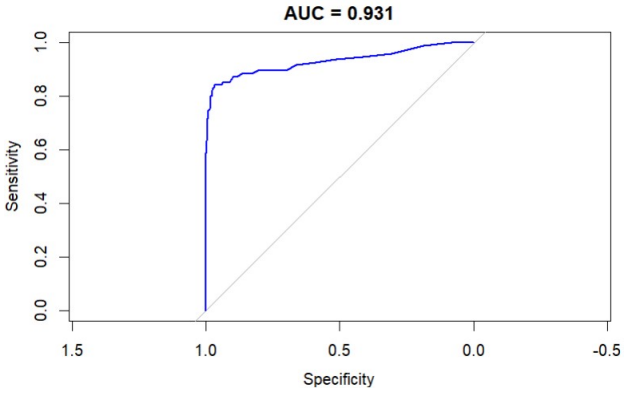

Table 5 and Figure 6 shows the confusion matrix and ROC Curve statistics of the Random Forest Model. The model showed robust performance with an accuracy of 94.90%, balanced accuracy of 83.42%, and an AUC of 0.931. It demonstrated excellent sensitivity (99.48%) and strong positive predictive value (94.83%), though specificity was slightly lower at 67.37%. The F1-score was 0.9710.

3.6 Gradient Boosting

Table 6. Gradient Boosting Model Confusion Matrix & ROC Curve Statistics

Gradient Boosting					
Accuracy	Recall	Specificity	Precision	F1	AUC
0.955	0.9878	0.7579	0.9609	0.9741	0.935
Balanced Accuracy		0.8728			

Figure 7. Gradient Boosting Model ROC Curve

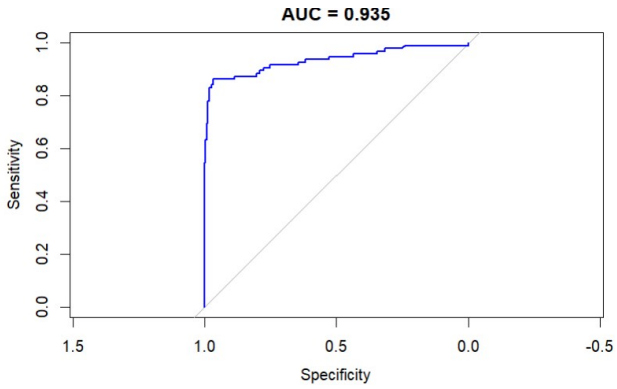

Table 6 and Figure 7 shows the confusion matrix and ROC Curve statistics of the Gradient Boosting model. The model emerged as the top-performing model, achieving an accuracy of 95.5%, a balanced accuracy of 87.28%, and an AUC of 0.935. It maintained high sensitivity of 98.78% and a relatively high specificity of 75.79%. The F1-score was 0.9741.

4. Discussion

The regression-based models—logistic, ridge, and lasso—showed moderate performance in identifying non-churners, which were designated as the positive class. Logistic regression, a straightforward and interpretable model, achieved an AUC of 0.831 and a balanced accuracy of 0.7698. It performed reasonably well in identifying non-churners, with a sensitivity of 72.9% and a high positive predictive value (95.86%). However, its specificity and negative predictive value were substantially lower, indicating a limited ability to correctly detect churners.

Ridge and lasso regression improved overall accuracy (both over 85%), but their extremely low specificity (8.4% and 12.6%, respectively) revealed a strong bias toward the majority class (non-churn). While both models exhibited high sensitivity and F1-scores, their capacity to identify churners (class 1) was weak. It is evident that there was poor agreement between predictions and actual classes. Thus, although regression models maintained decent overall classification rates, they failed to capture minority class behavior. The end goal is a churn predicting model; and when the model fails to effectively spot customers who will churn, then it will not be able to give substantial information to the company.

On the other hand, tree-based models clearly outperformed regression-based approaches in all key metrics. The decision tree model achieved a balanced accuracy of 0.8869 and a strong F1-score of 0.9749, suggesting high overall agreement with the true labels. It also exhibited excellent sensitivity (98.43%), indicating that it correctly identified non-churners, and a solid specificity (78.95%) in detecting churners. Its high interpretability makes it valuable for understanding which factors contribute to customer decisions, especially when coupled with the obtained important features from the Methodology section.

Random forest further enhanced generalization by reducing variance through ensemble averaging. It achieved an AUC of 0.934 and demonstrated exceptional sensitivity (99.48%), indicating its strength in identifying non-churners. Its specificity (67.37%), while lower, still reflected a fair ability to detect churners. Moreover, its balanced accuracy (0.8342) and high F1-score (0.9710) indicated a good trade-off between precision and recall. All in all, these values signify substantial agreement with the actual class labels in the model.

Gradient boosting offered the strongest performance overall. It matched the decision tree in accuracy (95.5%) and surpassed most models in F1-score (0.9741), but it also demonstrated an excellent balance between sensitivity (98.78%) and specificity (75.79%). This made it highly effective at identifying both non-churners and churners, a crucial advantage in practical applications. It has the highest AUC of 0.935, confirming its strong discriminative capability. This is exactly why it is a better model than the decision tree while having almost the same statistical values.

What sets this model apart was its iterative learning process, which focused on correcting misclassified examples from previous rounds, enabling it to better capture complex patterns in the data. Furthermore, its ability to handle multicollinearity and nonlinear relationships made it particularly suitable for the structured, tabular nature of the dataset which often exhibit these characteristics.

There is also a good conversation point regarding the difference of approach regarding the bias-variance trade-off between the regression and tree-based models. Logistic regression, being a linear model, represents a low-variance but high-bias approach. It generalizes well in simple contexts but struggles with capturing complex relationships, as evidenced by its underwhelming performance. Decision

trees, while flexible and expressive, suffer from high variance and are prone to overfitting, particularly with noisy or imbalanced data. Ensemble techniques like random forest and gradient boosting strike a more favorable balance by reducing variance while maintaining low bias. Gradient boosting, in particular, excels due to its ability to iteratively reduce errors which significantly enhances the stability of the model and its accuracy.

Interestingly, a clear trade-off was observed between interpretability and predictive strength. Logistic regression and decision trees are inherently more transparent and easier to explain to stakeholders. However, they fall short in predictive performance compared to ensemble models. Random forest and Gradient Boosting, while more complex and harder to interpret, offer significant gains in accuracy and robustness. Fortunately, tools like confusion matrix & ROC Curve statistical values and feature importance metrics can help bridge the interpretability gap by providing usable insights into how individual variables influence predictions. In this case, call minutes during the day and evening as well as service call information are extremely essential in understanding whether a specific customer will churn or not.

In practical telecom scenarios, a churn prediction model with high sensitivity to non-churners (class 0) ensures that loyal customers are correctly identified and spared unnecessary retention efforts. At the same time, adequate specificity is crucial to correctly flag likely churners for proactive retention strategies. Arguably, predicting churners is the most important aspect to evaluate in a model for this specific use. Gradient boosting, with its balanced performance, minimizes the cost of both false positives and false negatives.

Feature importance analysis across tree-based models revealed that non-churn behavior is strongly influenced by total call minutes during the day and evening, number of customer service calls, and international call usage. These findings suggest that customers who use core services more consistently and experience fewer customer service interactions are less likely to churn.

Armed with this insight, telecom companies can design personalized interventions: targeting users with declining service usage or rising customer complaints with tailored offers, improving service quality, and enhancing engagement. Moreover, by integrating churn prediction models into customer relationship management systems, companies can automate alerts, prioritize high-risk segments, and allocate resources more efficiently. This predictive capacity not only protects revenue but also improves customer satisfaction and long-term loyalty.

5. Conclusion & Recommendations

Conclusion

Among the models evaluated for predicting customer churn, gradient boosting and decision tree methods delivered the strongest overall performance. Both achieved high accuracy—around 95.6%—with AUC values near 0.93 and F1-scores close to 0.98, demonstrating excellent ability to correctly identify customers likely to churn while minimizing false positives. Random forest also showed strong results with a good balance of accuracy and AUC. These tree-based models effectively capture nonlinear relationships and interactions from an extensive amount of variables which is extremely crucial in churn prediction. In contrast, linear models such as logistic regression, ridge, and lasso regression, while offering greater interpretability, showed lower accuracy (74% to 86%) and moderate AUC scores, reflecting their limited capacity to model complex patterns and class imbalance in churn data.

The bias-variance trade-off is evident in these results: linear models tend to have higher bias and lower variance, making them more stable but prone to underfitting. Ensemble tree methods like gradient boosting and random forest reduce bias while managing variance through aggregation, resulting in superior predictive performance, though at some cost to interpretability. However, feature importance analysis across these models consistently highlights total call minutes during the day and evening, the number of customer service calls, and total international call minutes as the most influential predictors.

For a telecom company aiming to reduce churn, gradient boosting or random forest models are recommended due to their strong predictive power and ability to identify at-risk customers accurately. These models enable focused retention efforts by targeting customers with high usage patterns or frequent service calls, maximizing marketing and operational effectiveness. At the same time, the consistent importance of specific usage metrics as mentioned prior offers interpretability that supports strategic decision-making. For instance, the number of customer service calls clearly shows that consistent complaints in service can lead to higher churn rates. If interpretability remains a priority, decision trees or linear models can help communicate churn drivers clearly to stakeholders. However, gradient boosting clearly stands out as the best choice to balance predictive accuracy with actionable insights.

Recommendations

Future researchers aiming to build predictive models for customer churn should consider several methodological and

practical improvements to enhance both model performance and generalizability. First, addressing class imbalance more rigorously is critical. While this study highlighted its effects on regression models, future work could benefit from implementing resampling strategies such as SMOTE or applying cost-sensitive learning approaches to better penalize misclassification of the minority class. Additionally, exploring hybrid models that combine the interpretability of linear models with the power of ensemble methods may offer a better trade-off between transparency and predictive accuracy.

Researchers are also encouraged to experiment with alternative boosting algorithms to improve performance on categorical variables and imbalanced datasets. Moreover, deeper exploration of feature engineering, such as generating time-based or interaction features, could reveal hidden patterns that will potentially improve model accuracy. More importanyly, this could also boost model power in predicting churners. Incorporating external datasets, such as customer demographics or market conditions, may also enhance the robustness of the churn predictions.

Lastly, reproducibility should be prioritized. Researchers should consider applying their models to different datasets or industries to test for generalizability and identify domain-specific factors that influence customer churn. One way in which this could be done is by validating models across multiple datasets through either different telecom datasets or synthetically benchmarked ones. These steps will help support the development of more reliable, interpretable, and actionable churn prediction systems.

References

Albérico, R., & Casaca, J. (2024). Relationship Marketing and Customer Retention - A Systematic Literature Review. *Studies in Business Economics, 18*(3), 44-66. http://dx.doi.org/10.2478/sbe-2023-0044

Appendices

Code (Paper): https://github.com/marklee1000/DATA-MINING-AND-WRANGLING/blob/54c3f7db1b2f98cd9fa56ec145ef5b4b85e2bda7/SA2/SEC-1-SA2-GROUP-4-BAYBAYON%2C-D--MAYOL%2C-J.md

Code (Bonus): https://github.com/marklee1000/DATA-MINING-AND-WRANGLING/blob/54c3f7db1b2f98cd9fa56ec145ef5b4b85e2bda7/SA2/Bonus.ipynb

Bonus: Fit a neural network to the Default data. Use a single hidden layer with 10 units, and dropout regularization. Have a look at Labs 10.9.1–10.9.2 for guidance. Compare the classification performance of your model with that of linear logistic regression.

Figure 8. Neural Network Performance

```
Neural Network Performance
[[1928    5]
 [  51   16]]
              precision    recall  f1-score   support

           0       0.97      1.00      0.99      1933
           1       0.76      0.24      0.36        67

    accuracy                           0.97      2000
   macro avg       0.87      0.62      0.67      2000
weighted avg       0.97      0.97      0.96      2000

AUC: 0.948
```

Figure 9 shows the performance of the neural network model. The model shows a high overall classification accuracy of 97% and an excellent AUC of 0.948, indicating strong discriminatory power between the two classes. However, the performance metrics reveal a significant class imbalance problem. The model achieves near-perfect precision and recall for the majority class (non-default, class 0), but it struggles to correctly identify the minority class (default, class 1). Specifically, it only captures 24% of actual defaults (recall), which is concerning since identifying defaulters is critical. The F1-score for class 1 is just 0.36, reflecting a poor balance between precision and recall for that class. This suggests that the model is biased toward predicting the majority class and is failing to capture the rare but important instances of default.

Figure 9. Logistic Regression Performance

```
Logistic Regression Performance
[[1925    8]
 [  46   21]]
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      1933
           1       0.72      0.31      0.44        67

    accuracy                           0.97      2000
   macro avg       0.85      0.65      0.71      2000
weighted avg       0.97      0.97      0.97      2000

AUC: 0.948
```

The logistic regression model, like the neural network, achieves a high overall accuracy of 97% and an identical AUC of 0.948. This indicates that both models are equally effective at distinguishing between the default and non-default classes when evaluated using the area under the ROC

curve. However, a closer look at the classification metrics again reveals the challenge posed by class imbalance. The logistic regression model performs very well on the majority class (non-default), with near-perfect precision and recall. For the minority class (default), the model achieves a recall of 31%, slightly better than the neural network's 24%, and a precision of 72%, which is comparable. Its F1-score for the minority class is 0.44, outperforming the neural network's 0.36 and indicating a better balance between precision and recall in identifying defaulters.

Model Comparison & Evaluation

Both models achieve the same overall accuracy of 97% and an identical AUC of 0.948, indicating strong and comparable ability to distinguish between defaulters and non-defaulters. However, important differences emerge when examining performance on the minority class (defaults). The neural network correctly identifies only 24% of defaulters (recall), with a corresponding F1-score of 0.36, despite having high precision (76%). This indicates that the model is conservative in predicting defaults and fails to detect most of them. In contrast, the logistic regression model achieves a recall of 31% and an F1-score of 0.44 for the same class, outperforming the neural network in its ability to capture actual default cases. While both models perform well on the majority class (non-default), logistic regression demonstrates slightly better balance between precision and recall for the minority class, making it the preferable choice in this case, especially when the ability to identify defaults is more critical than overall accuracy. A possible explanation could be that the neural network’s complexity may have caused it to overfit to the dominant class.