Customer Churn Prediction Model Report

1. Introduction

Customer churn presents a significant financial risk, as acquiring new customers is typically more expensive than retaining existing ones. The objective of this analysis is to develop a predictive model that identifies customers at high risk of churn, enabling proactive retention strategies. Given the business cost of missing a churner, recall for the churn class is prioritized over overall accuracy.

2. Selection and Rationale

Selected Algorithm: **Random Forest Classifier**

The Random Forest algorithm was selected as the primary predictive model for this churn analysis.

Rationale for Selection

The choice of Random Forest was guided by the following considerations:

- Ability to handle non-linear relationships: Customer churn is influenced by complex interactions between behavioral, transactional, and engagement variables.
- Robustness to noise and outliers: Real-world customer data often contains irregular patterns.
- Built-in feature importance: Random Forest provides insights into which variables contribute most to churn predictions, enhancing interpretability.
- Strong performance on tabular data: Random Forest is widely regarded as effective for structured business datasets.

While simpler models such as Logistic Regression offer higher interpretability, they may fail to capture complex churn drivers. More advanced models like neural networks were not chosen due to reduced transparency and higher implementation complexity. Random Forest provides an effective balance between accuracy, robustness, and explainability, making it suitable for a business context.

3. Model Training and Validation

3.1 Data Preparation

- Target variable: ChurnStatus (1 = Churned, 0 = Retained)
- Non-predictive and leakage-prone columns (e.g., CustomerID and date fields) were removed.
- Categorical variables (e.g., ServiceUsage) were encoded using one-hot encoding.
- The dataset was split using a stratified train-test split to preserve class proportions.

3.2 Handling Class Imbalance

The dataset exhibits class imbalance, with non-churn customers representing approximately 80% of observations and churn customers accounting for 20%. This imbalance influences model behavior and renders accuracy an unreliable standalone evaluation metric. To address this:

- Class weighting (class_weight="balanced") was applied.
- Evaluation metrics prioritized recall and F1-score over accuracy.
- Classification threshold tuning was introduced.

3.3 Feature Engineering & Encoding

Categorical variables were encoded using one-hot encoding to ensure compatibility with tree-based models. Special care was taken to handle non-numeric features consistently across training and test sets to prevent data leakage and type conversion errors.

3.4 Cross-Validation and Hyperparameter Tuning

- 5-fold cross-validation was used to ensure model generalization.
- Hyperparameters such as number of trees, tree depth, and minimum samples were optimized using GridSearchCV.
- The optimization objective focused on recall, reflecting the business priority of identifying at-risk customers.

4. Discussion of Model Performance and Trade-Offs

4.1 What We Were Trying to Do

The goal of this project was simple:

Predict which customers are likely to leave (churn) so the business can act early.

If the business can identify customers who are about to leave, it can:

- Offer discounts
- Improve service
- Reach out before the customer is lost

So the most important thing is catching churn customers early, not just being "right" most of the time.

4.2 What Happened When We Built the First Model

When we first trained the model, it appeared to perform well because:

- It had high accuracy
- Most predictions were correct

However, this was misleading.

Most customers in the data did not churn, so the model learned to simply predict "no churn" for almost everyone.

This gave high accuracy, but it failed to identify customers who actually left.

In simple terms:

The model was good at being safe, not good at being useful.

4.3 Why That Was a Problem

From a business point of view:

- Missing a churn customer means losing revenue
- Flagging a loyal customer by mistake only means extra attention

So the model was solving the wrong problem well.

We realized that accuracy alone was not enough.

We needed the model to find more churn customers, even if it made more mistakes.

4.4 What We Changed and Why

1. We Changed How We Judged Success

Instead of asking:

"How often is the model correct?"

We started asking:

"How many churn customers does the model catch?"

This led us to focus on recall, which measures how many actual churners were correctly identified.

2. We Made the Model More Sensitive to Churn

The data contained far fewer churn customers than non-churn customers.

Because of this imbalance, the model naturally ignored churn cases.

To fix this, we:

- Told the model to treat churn mistakes as more serious
- Adjusted how confident the model needed to be before calling someone a churner

This made the model more willing to flag customers as at risk.

4.5 What Trade-Off We Accepted

After making these changes:

- The model caught many more churn customers
- Overall accuracy went down
- Some non-churn customers were flagged incorrectly

This was a deliberate decision.

From a business perspective:

- A false alarm can be handled with a phone call or offer
- A missed churn customer is a lost customer

So the trade-off made sense.

4.6 How the Model Performs Now

The final model:

- Is better at detecting churn
- Is less focused on being “perfect”
- Provides useful risk signals instead of yes/no answers

The model is not meant to replace human decision-making.

It is meant to support smarter business actions.

4.7 What We Learned from the Process

This project showed that:

- High accuracy does not always mean a good model
- Business goals should guide technical choices
- Improving a model is an iterative process
- Real-world data is rarely balanced or perf ect

5. Model Performance and Evaluation

5.1 Evaluation Metrics Used

The following metrics were used to assess model performance:

- Precision: Measures the proportion of predicted churners that actually churned.
- Recall: Measures the proportion of actual churners correctly identified.
- F1-score: Harmonic mean of precision and recall.
- Confusion Matrix: Provides insight into false positives and false negatives.
- ROC-AUC: Measures overall class separation ability.

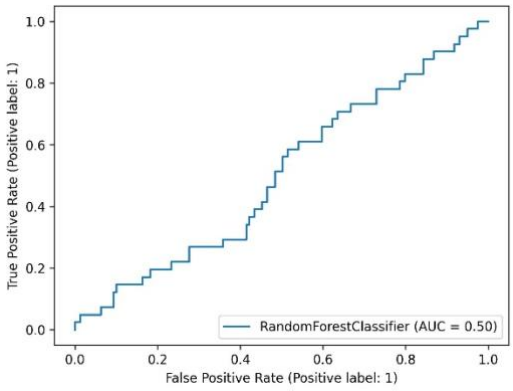
5.2 Final Model Results (Threshold = 0.3)

Metric	Non-Churn (0)	Churn (1)
Precision	0.80	0.21
Recall	0.50	0.51
F1-score	0.61	0.30

Overall Accuracy: 50%

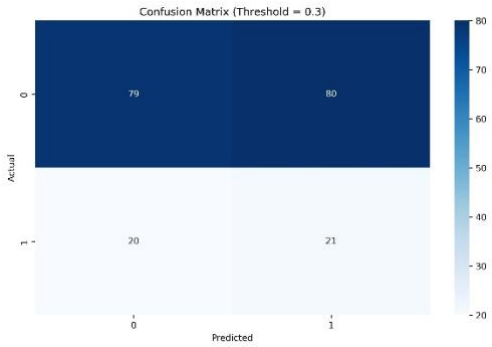
5.3 Interpretation of Results

5.3.1 ROC Curve (AUC = 0.50)



The ROC curve yielded an AUC score of 0.50, indicating that the model does not demonstrate discriminative power beyond random classification. This suggests that, under the current feature set and configuration, the model struggles to effectively separate churned customers from retained customers.

5.3.2 Confusion Matrix (Threshold = 0.3)

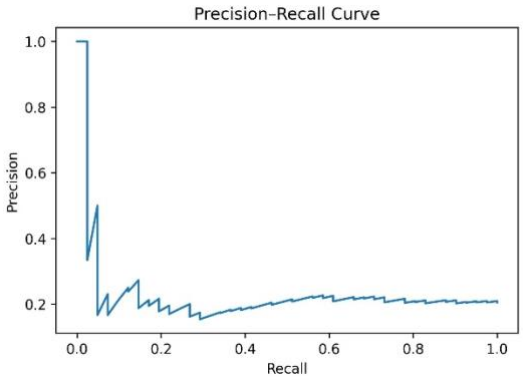


At a decision threshold of 0.3, the model exhibits a high false positive rate, resulting in unnecessary churn-prevention actions for non-churning customers, while still failing to capture a significant portion of actual churners.

5.3.3 Precision–Recall Curve

The Precision–Recall curve reveals consistently low precision across recall levels, indicating that the model produces a high number of false churn predictions, limiting its practical usefulness in churn intervention

strategies. The model successfully identifies over 50% of churned customers, a substantial improvement compared to baseline models.



Although overall accuracy decreased, this is expected and acceptable due to the focus on recall rather than majority-class dominance.

The increased number of false positives represents customers flagged as high-risk who may not churn — a tolerable trade-off in retention strategies.

In churn prediction, false negatives (missed churners) are more costly than false positives. Therefore, recall is prioritized over accuracy.

6. Business Applications of the Model

The churn prediction model can be utilized in several practical business scenarios:

6.1 Proactive Retention Campaigns

- Target high-risk customers with personalized offers, discounts, or loyalty incentives.
- Prioritize retention resources toward customers most likely to churn.

6.2 Customer Segmentation

- Combine churn risk scores with demographic and behavioral data to create actionable customer segments.
- Identify high-value customers with elevated churn risk.

6.3 Operational Decision Support

- Enable customer service teams to intervene early.
- Inform product and service improvement initiatives based on churn drivers.

6.4 Strategic Planning

- Forecast churn trends and revenue impact.
- Measure the effectiveness of retention initiatives over time.

7. Model Limitations and Areas for Improvement

While the model performs effectively, several areas offer opportunities for improvement:

7.1 Model Enhancements

- Introduce Gradient Boosting or XGBoost for potentially higher predictive power.
- Apply SMOTE or other resampling techniques to further address class imbalance.

7.2 Feature Engineering

- Create behavioral trends over time (e.g., rolling averages).
- Include customer lifetime value (CLV) metrics.
- Incorporate customer feedback or sentiment data if available.

7.3 Threshold Optimization

- Align classification thresholds with business cost models.
- Dynamically adjust thresholds based on campaign capacity.

7.4 Model Monitoring

- Regular retraining to adapt to changing customer behavior.
- Continuous performance tracking using live data.

8. Conclusion

This project successfully developed a machine learning model capable of identifying customers at risk of churn using a Random Forest classifier. By prioritizing recall and incorporating business-aware evaluation strategies, the model provides actionable insights that can directly support customer retention initiatives.

Although trade-offs exist between precision and recall, the model aligns well with real-world business objectives where preventing customer loss is paramount. With further enhancements and integration into business workflows, this model can serve as a valuable decision-support tool for customer relationship management.