# Durham College - InnovateON.ca

**1202 – Data Analysis Tool Analytics**

Project Report

Submitted by: Group 9

Submitted to: Prof. Mizanur Rahman

Contents

[ 1](#_Toc184898327)

[Introduction 3](#_Toc184898328)

[Exploratory Data Analysis (EDA) 4](#_Toc184898329)

[Data Preprocessing and Feature Engineering 14](#_Toc184898330)

[Model Training and Evaluation 17](#_Toc184898331)

[ROC and AUC Analysis 26](#_Toc184898332)

[Adding Changes 30](#_Toc184898333)

[Conclusion 33](#_Toc184898334)

# Introduction

Customer churn is a critical issue for businesses, particularly in banking, where losing a customer can translate into significant financial and reputational losses. The goal of this project is to predict whether a customer will churn (leave the bank) or remain with the organization. This involves leveraging machine learning models to understand customer behavior and identify churn risk. Early detection enables the bank to deploy targeted retention strategies, improving profitability and customer satisfaction.

**Methodology**

The project follows the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework, ensuring a systematic approach:

1. **Business Understanding**: Define churn as a significant problem requiring predictive solutions.
2. **Data Understanding**: Explore the dataset to uncover patterns and correlations.
3. **Data Preparation**: Clean, preprocess, and engineer features for machine learning.
4. **Modeling**: Train and evaluate multiple models to identify the most effective one.
5. **Evaluation**: Use classification metrics (accuracy, precision, recall, F1-score, and AUC) to compare model performance.
6. **Deployment**: Propose the best model for real-world application.

# Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial phase in any data science project, as it helps uncover underlying patterns, relationships, and potential issues within the dataset. For this project, EDA was particularly significant in understanding the factors contributing to customer churn and identifying the key features that could drive predictive modeling.

**Objective of EDA**

The primary objectives of EDA in this project were:

1. To understand the distribution and characteristics of the data.
2. To identify patterns and trends in customer behavior that correlate with churn.
3. To detect and address potential data quality issues such as imbalance, outliers, or missing values.
4. To derive insights that can inform feature engineering and model selection.

**Dataset Overview**

The dataset consists of 10,000 rows and 12 columns, with churn as the target variable (1 = churn, 0 = retained). The features encompass a mix of numerical and categorical attributes, including demographic, financial, and behavioral data:

* Demographic: age, gender, country
* Financial: credit\_score, balance, estimated\_salary
* Behavioral: tenure, products\_number, active\_member, credit\_card

1. **Churn Distribution**

One of the first observations during EDA was the imbalance in the churn variable:

* Approximately 80% of customers are retained (class 0), while only 20% have churned (class 1).
* This imbalance is a common challenge in classification problems, as it can lead to models biased toward the majority class. Techniques like SMOTE were later applied to address this issue.

A graph with a bar and number

Description automatically generated with medium confidence

1. **Feature Exploration**

EDA provided insights into the individual and joint distributions of features. Key findings include:

* **Credit Score**:
  + Credit scores are normally distributed, with most customers scoring between 600 and 700.
  + No significant relationship was observed between credit score and churn, suggesting that other features might play a stronger role.

A graph of a credit score

Description automatically generated

* **Age**:
  + Older customers show a higher likelihood of churn. A positive correlation between age and churn was observed during correlation analysis.
  + This trend might indicate that older customers are less satisfied with the bank’s services or find it easier to switch providers.

A graph of a number of age

Description automatically generated with medium confidence

* **Balance**:
  + A notable portion of customers has a zero balance. These customers are more likely to churn, as a zero balance may indicate inactivity or disengagement with the bank’s services.
  + Customers with moderate to high balances tend to stay with the bank.
* **Tenure**:
  + Customers with longer tenure are less likely to churn. This feature highlights customer loyalty, with churn rates decreasing as the length of the relationship increases.
* **Products Number**:
  + Customers holding more products are less likely to churn, indicating that cross-selling and bundling strategies are effective retention mechanisms.

A graph of a bar chart

Description automatically generated

* **Active Membership**:
  + Active members are significantly less likely to churn compared to inactive ones, reinforcing the importance of customer engagement.
* **Country** and **Gender**:
  + Customers from different countries (France, Spain, and Germany) exhibited varying churn rates, with Germany showing slightly higher churn.

A graph of a bar chart

Description automatically generated with medium confidence

* + Gender did not show a significant impact on churn likelihood.

A graph with different colored bars

Description automatically generated with medium confidence

1. **Correlation Analysis**

A correlation heatmap was generated to visualize the relationships between numerical features. Key observations include:

* Positive correlation between age and churn, suggesting that older customers are at higher risk.
* Negative correlation between balance and churn, indicating that customers with higher balances are less likely to leave.
* Other features, such as tenure and products\_number, also showed weak to moderate correlations with churn.

A screenshot of a graph

Description automatically generated

1. **Class Imbalance**

The imbalance in the churn variable was a critical challenge identified during EDA. Imbalanced datasets can lead to models that favor the majority class (non-churn) at the expense of the minority class (churn). This was later addressed using SMOTE, which generated synthetic examples for the minority class to balance the dataset.

1. **Joint Distributions**

Visualizations such as histograms, box plots, and pair plots were used to explore the joint distributions of features. For instance:

* **Age vs. Balance**: Older customers with lower balances showed a higher churn rate.
* **Products Number vs. Balance**: Customers with more products and higher balances exhibited lower churn.

1. **Outliers and Anomalies**

* Few outliers were detected in features like balance and estimated\_salary. These outliers were retained, as they likely represent real customer behaviors rather than data errors.
* The presence of zero balances for a large portion of customers was flagged as a significant feature for modeling.

1. **Insights Derived from EDA**

* **Key Predictors**: Age, balance, tenure, active membership, and products number emerged as potential key predictors of churn.
* **Inactivity Indicators**: Zero balance and inactive membership are strong signals of disengagement, which likely precede churn.
* **Customer Engagement**: Features indicating higher engagement (e.g., active membership, holding multiple products) correlate negatively with churn.

A graph of a diagram

Description automatically generated with medium confidence

A screenshot of a graph

Description automatically generated

# Data Preprocessing and Feature Engineering

Data preprocessing and feature engineering are pivotal stages in the development of any machine learning model, as they ensure that the raw data is transformed into a format suitable for analysis. In this project, the preprocessing steps addressed data quality issues, handled imbalances in the target variable, and prepared features for optimal performance in machine learning algorithms. Additionally, feature engineering was employed to extract meaningful insights from existing variables, further improving the predictive power of the model.

1. **Handling Categorical Features**

The dataset contained two categorical features, country and gender, which needed to be converted into numerical representations for compatibility with machine learning algorithms.

* **Country Encoding**:  
  The country variable, consisting of values France, Spain, and Germany, was mapped to numerical values: France (0), Spain (1), Germany (2). This ordinal encoding preserved the categorical nature of the variable while ensuring its usability in algorithms.
* **Gender Encoding**:  
  Similarly, the gender variable was mapped to numerical values: Female (0) and Male (1). This encoding allowed the model to differentiate between male and female customers during training.

The encoding of these categorical features ensured that their inherent information was preserved while making them compatible with numerical-based algorithms.

1. **Feature Scaling**

The dataset contained numerical features such as credit\_score, balance, age, and estimated\_salary, which had different ranges and distributions. To ensure uniformity, these features were standardized using a StandardScaler. Standardization transformed each feature to have a mean of 0 and a standard deviation of 1.

The importance of feature scaling cannot be overstated:

* Algorithms like Logistic Regression and Neural Networks are sensitive to the scale of input features. Without scaling, features with larger ranges (e.g., balance) would dominate the learning process, skewing the results.
* Standardized data also accelerates convergence during optimization, particularly for gradient-based algorithms.

1. **Addressing Class Imbalance**

One of the most critical challenges in this project was the significant imbalance in the target variable, churn. Approximately 80% of the dataset consisted of non-churners (class 0), while only 20% were churners (class 1). Without addressing this imbalance, the model would likely prioritize the majority class, leading to poor performance in identifying churners.

To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) was applied:

* **What is SMOTE?**  
  SMOTE is an oversampling technique that generates synthetic examples for the minority class by interpolating between existing samples. This increases the representation of the minority class without simply duplicating existing data.
* **Why SMOTE?**  
  SMOTE is particularly effective for imbalanced datasets as it reduces overfitting while improving the model’s ability to learn patterns for the minority class.

After applying SMOTE, the dataset was balanced, ensuring that both classes were equally represented during model training.

1. **Train-Test Split**

To evaluate model performance, the dataset was split into training and testing subsets:

* **Training Set**: 80% of the data was used to train the machine learning models.
* **Testing Set**: 20% of the data was reserved for evaluating model performance on unseen data.

Stratified splitting was used to maintain the original distribution of the target variable in both subsets. This ensured that the class proportions in the training and testing sets were representative of the entire dataset.

1. **Feature Engineering**

Feature engineering involved creating new variables or transforming existing ones to enhance the dataset's predictive power. While the dataset was relatively well-structured, certain features were analyzed and refined:

* **Balance as an Indicator of Activity**:  
  Customers with a balance of 0 were flagged as potentially inactive. This binary indicator was considered during modeling to capture disengagement patterns.
* **Interaction Terms**:  
  Interaction terms between variables, such as age \* balance and products\_number \* active\_member, were explored to capture non-linear relationships.
* **Normalization of Products Number**:  
  Customers holding an unusually high number of products were flagged, as this might indicate unique engagement behavior.

These transformations added depth to the dataset, allowing models to capture more nuanced patterns in the data.

1. **Outlier Treatment**

While most features had well-distributed values, certain outliers in features like balance and estimated\_salary were identified. However, these outliers were retained during modeling for two reasons:

* They likely represent real customer behaviors rather than data errors.
* Models like Random Forest and Neural Networks are robust to outliers and can handle such anomalies effectively.

1. **Challenges in Preprocessing**

Several challenges were encountered and addressed during preprocessing:

* **Categorical Features**: While simple encoding methods worked well for country and gender, more advanced techniques like one-hot encoding could be considered for larger datasets with diverse categories.
* **Class Imbalance**: The application of SMOTE resolved the imbalance but introduced synthetic data, which might not perfectly represent real-world patterns. This trade-off was accepted to improve model performance.
* **Scaling and Interpretability**: While standardization improved model performance, it slightly reduced interpretability for certain models like Logistic Regression, where coefficients directly correspond to feature importance.

1. **Impact of Preprocessing**

The preprocessing steps significantly enhanced the quality of the dataset, ensuring that it was ready for machine learning:

* The balanced dataset provided fair learning opportunities for both classes.
* Scaled features ensured consistent performance across different algorithms.
* Engineered features added complexity and depth, improving the model's ability to identify patterns.

# Model Training and Evaluation

The core objective of this project was to build a predictive model for customer churn that is both accurate and interpretable. To achieve this, multiple machine learning algorithms were trained and evaluated on the dataset. Each model was chosen for its ability to address specific challenges such as handling class imbalance, capturing non-linear relationships, or providing interpretability. The models were rigorously evaluated using standard metrics to ensure robust performance, particularly for the minority class (churn).

1. **Models Trained**

A diverse set of machine learning models was selected to ensure a comprehensive exploration of potential solutions:

* **Logistic Regression**: A simple and interpretable baseline model that works well for linearly separable data.
* **Naive Bayes**: A probabilistic model leveraging conditional probabilities to predict outcomes.
* **Decision Tree**: A non-linear model capable of capturing intricate relationships between features.
* **Random Forest**: An ensemble learning method that combines multiple decision trees to improve accuracy and generalizability.
* **Neural Network (Multi-Layer Perceptron)**: A deep learning approach for capturing complex patterns in high-dimensional data.

Each model was implemented using Python’s scikit-learn library, and hyperparameters were set to their default values for initial training. Advanced tuning was later applied to the top-performing models.

1. **Training Process**

The training process involved several key steps:

* **Data Input**: The preprocessed dataset, with balanced classes after applying SMOTE, was split into training (80%) and testing (20%) subsets.
* **Feature Scaling**: Numerical features were standardized to ensure that algorithms sensitive to scale (e.g., Logistic Regression, Neural Networks) performed optimally.
* **Model Fitting**: Each algorithm was trained on the training set using the balanced data. The models learned patterns and relationships between the input features and the target variable (churn).

1. **Evaluation Metrics**

To evaluate the performance of each model, a combination of metrics was used, focusing on both overall accuracy and the model’s ability to predict the minority class (churn):

* **Accuracy**: The proportion of correctly classified instances out of the total instances.
* **Precision**: The proportion of true churn predictions out of all churn predictions. High precision indicates fewer false positives.
* **Recall (Sensitivity)**: The proportion of actual churn cases correctly identified. High recall ensures the model captures most churners.
* **F1-Score**: The harmonic mean of precision and recall, balancing the trade-off between the two.
* **ROC-AUC**: The Area Under the Receiver Operating Characteristic Curve measures the model’s ability to discriminate between churners and non-churners across various thresholds.

These metrics ensured that the evaluation was comprehensive, capturing both the model’s accuracy and its effectiveness in identifying the minority class.

1. **Model Performance**
   1. **Logistic Regression**:

* Accuracy: 80%
* Precision: 0.59
* Recall: 0.17
* F1-Score: 0.27
* ROC-AUC: 0.75 Logistic Regression served as a baseline model. It performed well for the majority class (non-churn) but struggled with the minority class due to the linear separability assumption.

A diagram of a confused matrix

Description automatically generated

1. **Naive Bayes**:

* Accuracy: 82%
* Precision: 0.68
* Recall: 0.27
* F1-Score: 0.39
* ROC-AUC: 0.79 Naive Bayes improved recall for the minority class but still lacked sufficient discriminatory power.

A diagram of a confused matrix

Description automatically generated

* 1. **Decision Tree**:
* Accuracy: 78%
* Precision: 0.47
* Recall: 0.48
* F1-Score: 0.47
* ROC-AUC: 0.68 The Decision Tree model achieved a balanced precision and recall, but its performance suffered from overfitting due to its tendency to capture noise in the training data.

A graph showing a comparison of a number of labels

Description automatically generated with medium confidence

1. **Random Forest**:

* Accuracy: 85%
* Precision: 0.76
* Recall: 0.43
* F1-Score: 0.55
* ROC-AUC: 0.84 Random Forest emerged as one of the top performers, offering a good balance between accuracy, precision, and recall. Its ensemble nature mitigated overfitting while capturing non-linear relationships effectively.

A diagram of a confused matrix

Description automatically generated

* 1. **Neural Network**:
* Accuracy: 85.3%
* Precision: 0.75
* Recall: 0.43
* F1-Score: 0.55
* ROC-AUC: 0.84 The Neural Network achieved similar results to Random Forest, demonstrating strong generalizability and the ability to capture complex patterns. However, its interpretability was limited compared to simpler models.

A diagram of a confused matrix

Description automatically generated

1. **Insights from Model Evaluation**

* **Random Forest and Neural Network** emerged as the best-performing models, achieving high accuracy and balanced precision and recall. These models effectively handled class imbalance and non-linear relationships.
* Simpler models like Logistic Regression and Naive Bayes struggled with the minority class, highlighting the need for more sophisticated techniques.
* Decision Tree, while interpretable, was prone to overfitting, limiting its reliability for generalization.

1. **Comparative Analysis**

A comparison of the models using evaluation metrics showed that Random Forest and Neural Network were the most reliable options for deployment. Their high ROC-AUC scores (0.84) confirmed their ability to discriminate between churners and non-churners effectively.

A graph of different colored bars

Description automatically generated

1. **Challenges Encountered**

* **Class Imbalance**: Despite applying SMOTE, some models still struggled with recall for the minority class. Ensemble methods like Random Forest mitigated this issue.
* **Overfitting**: The Decision Tree model overfit the training data, necessitating the use of ensemble techniques.
* **Interpretability vs. Performance**: While Neural Networks offered high accuracy, their lack of interpretability posed challenges for business stakeholders.

1. **Impact of Evaluation**

The evaluation process provided a clear understanding of each model’s strengths and limitations, enabling an informed selection of the best-performing algorithms. Random Forest and Neural Network were chosen as the final candidates for deployment, balancing performance and scalability

A chart of blue squares with white text

Description automatically generated

# ROC and AUC Analysis

The Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) are essential tools for evaluating the performance of classification models, especially in datasets with imbalanced classes like this churn prediction project. These metrics provide insights into how well the models distinguish between the two classes—churn (1) and non-churn (0)—across various classification thresholds.

1. **Importance of ROC and AUC**

The ROC curve plots the **True Positive Rate (TPR)** (also known as recall) against the **False Positive Rate (FPR)** for different threshold values:

* **True Positive Rate (TPR)**: The proportion of correctly identified churners out of all actual churners.
* **False Positive Rate (FPR)**: The proportion of incorrectly predicted churners out of all non-churners.

The AUC represents the area under the ROC curve and provides a single scalar value summarizing the model’s ability to distinguish between the classes:

* AUC = 1: Perfect classification.
* AUC = 0.5: No discriminatory ability (equivalent to random guessing).
* AUC between 0.7 and 0.9: Represents good to excellent discriminatory power.

1. **Application in This Project**

In this project, the ROC curves and AUC scores were used to compare the discriminatory capabilities of the machine learning models:

* A higher AUC indicates better performance in distinguishing between churners and non-churners.
* ROC analysis allows for an in-depth understanding of the trade-offs between TPR and FPR at different thresholds, helping in selecting an optimal threshold for classification.

1. **ROC Curves for Models**

**Logistic Regression**:

* **AUC**: 0.75
* **Insights**: Logistic Regression achieved moderate discriminatory power, indicating that it performed better than random guessing but struggled with accurately predicting churners due to the linearity of the model.

**Naive Bayes**:

* **AUC**: 0.79
* **Insights**: Naive Bayes showed a slight improvement over Logistic Regression. Its probabilistic nature helped capture some patterns in the data, but it was less effective for the minority class (churn).

**Decision Tree**:

* **AUC**: 0.68
* **Insights**: The Decision Tree model had the lowest AUC score. While it performed well on the training data, overfitting reduced its ability to generalize, leading to a weaker discriminatory ability on the test data.

**Random Forest**:

* **AUC**: 0.84
* **Insights**: Random Forest was one of the top-performing models, with excellent discriminatory power. Its ensemble nature, which combines multiple decision trees, enabled it to effectively capture complex relationships and avoid overfitting.

**Neural Network (Multi-Layer Perceptron)**:

* **AUC**: 0.84
* **Insights**: The Neural Network matched Random Forest in terms of AUC, demonstrating its strength in capturing intricate patterns. However, its complexity and lack of interpretability limited its usability for business stakeholders.

A graph of a curve

Description automatically generated

1. **Interpretation of ROC Curves**

The ROC curves highlighted the trade-offs between TPR and FPR:

* Logistic Regression and Naive Bayes showed relatively steep curves at the initial thresholds, indicating better performance for the majority class (non-churn).
* Random Forest and Neural Network had more balanced curves, achieving high TPR with a low FPR across thresholds. This indicates strong performance for both classes.
* The Decision Tree curve demonstrated overfitting, as the model struggled to generalize beyond the training data.

**Comparison of AUC Scores**

| **Model** | **AUC Score** |
| --- | --- |
| Logistic Regression | 0.75 |
| Naive Bayes | 0.79 |
| Decision Tree | 0.68 |
| Random Forest | 0.84 |
| Neural Network | 0.84 |

Random Forest and Neural Network clearly outperformed other models, confirming their suitability for deployment.

1. **Selecting the Optimal Threshold**

While the default classification threshold (0.5) was used initially, the ROC curves provided insights into adjusting the threshold to balance precision and recall:

* For business-critical applications like churn prediction, a slightly lower threshold might prioritize recall, ensuring that more churners are identified even if false positives increase.
* This trade-off is particularly useful in scenarios where the cost of missing a churner is higher than the cost of falsely identifying a non-churner.

1. **Challenges in ROC Analysis**

* **Imbalance in the Dataset**: Despite using SMOTE, the original imbalance in the dataset influenced the ROC curves. Models like Logistic Regression and Naive Bayes showed weaker performance due to their inherent limitations in handling imbalance.
* **Threshold Selection**: While ROC provides a visual tool for threshold selection, it requires careful consideration of business priorities (e.g., prioritizing recall vs. precision).

1. **Insights from ROC and AUC**

* Random Forest and Neural Network demonstrated the best ability to differentiate between churners and non-churners, as reflected by their high AUC scores (0.84).
* While Logistic Regression and Naive Bayes provided moderate discriminatory power, their performance was not sufficient for practical deployment in a highly imbalanced setting.
* Decision Tree was the least effective, with an AUC of 0.68, primarily due to overfitting.

# Adding Changes

Throughout the project, several changes were implemented to address challenges, enhance model performance, and ensure that the solutions aligned with the project objectives. These adjustments were primarily driven by insights from Exploratory Data Analysis (EDA), model evaluations, and the need to balance the trade-offs between accuracy and interpretability. This section elaborates on the key changes made, their impact, and the rationale behind them.

1. **Addressing Class Imbalance with SMOTE**

One of the most significant challenges identified during EDA was the imbalance in the target variable (churn), where only 20% of the records belonged to the minority class (churners). Without addressing this imbalance, models would be biased toward predicting the majority class (non-churners), resulting in poor recall for churners.

* **Change Implemented**:  
  The Synthetic Minority Over-sampling Technique (SMOTE) was applied to generate synthetic samples for the minority class. This technique interpolates between existing samples in the minority class to create new, realistic data points.
* **Impact**:  
  SMOTE balanced the dataset, enabling models to learn patterns from both classes effectively. Recall and F1-scores for the minority class improved significantly after applying SMOTE, particularly for models like Random Forest and Neural Network.
* **Rationale**:  
  SMOTE was chosen over simple oversampling or undersampling to avoid duplication of minority class samples (which can lead to overfitting) and to retain the full set of majority class data.

1. **Hyperparameter Tuning**

Default hyperparameters were used in the initial training phase to establish baseline performance. However, advanced tuning was necessary to optimize the top-performing models (Random Forest and Neural Network) and achieve the best possible results.

* **Changes Implemented**:
  + **Random Forest**:
    - The number of trees (n\_estimators) was increased to 200 to improve accuracy.
    - The maximum depth (max\_depth) was limited to prevent overfitting.
    - The minimum samples per leaf (min\_samples\_leaf) were adjusted to ensure generalization.
  + **Neural Network**:
    - The architecture was modified to include three hidden layers with nodes (10, 5, 3) to balance complexity and training time.
    - The learning rate was fine-tuned to improve convergence during training.
    - Maximum iterations (max\_iter) were increased to 1,000 for better training stability.
* **Impact**:  
  Hyperparameter tuning significantly improved both models’ precision, recall, and F1-scores while maintaining high AUC values (0.84).
* **Rationale**:  
  Fine-tuning ensures that the models are neither overfitting the training data nor underperforming due to suboptimal configurations.

1. **Feature Engineering**

Feature engineering was performed to enhance the dataset’s predictive power. This involved creating new features and modifying existing ones based on insights from EDA.

* **Changes Implemented**:
  + **Binary Activity Indicator**:  
    A binary feature was created to flag customers with a balance of 0, representing potential inactivity.
  + **Interaction Terms**:  
    Interaction terms such as products\_number \* active\_member were introduced to capture non-linear relationships between features.
  + **Normalization**:  
    Features like products\_number were normalized to reduce the impact of extreme values.
* **Impact**:  
  These engineered features provided additional signals for the models, improving their ability to capture subtle patterns. Random Forest and Neural Network, in particular, benefited from these changes due to their ability to handle complex interactions.
* **Rationale**:  
  Feature engineering adds depth to the dataset, allowing the models to leverage previously hidden relationships between variables.

# Conclusion

The Bank Customer Churn Prediction project successfully leveraged machine learning to address the critical issue of customer churn. Through structured data analysis, preprocessing, and model training, the project identified Random Forest and Neural Network as the best-performing models, achieving high accuracy (85%+) and balanced metrics.

Key insights revealed age, balance, and active membership as strong predictors of churn, providing actionable strategies for targeted retention efforts, such as loyalty programs and engagement campaigns. Addressing challenges like class imbalance with SMOTE and refining models through hyperparameter tuning ensured robust and reliable predictions.

Future improvements include incorporating additional data sources, exploring advanced ensemble methods, and deploying the model for real-time churn prediction. This project demonstrates the potential of data-driven solutions to improve customer retention and enhance profitability, positioning the bank for sustainable success.