Customer Churn Prediction

Group 4: Baihan Liu, Jiayin Ye, Yupeng Yang, Ying Wang, Yizhuo Li

12/7/2023

* The source code for the project is available at GitHub Repository.

1 Summary

This analysis utilizes a comprehensive dataset from IBM to explore customer behaviors and characteristics that contribute to churn. The dataset includes demographic, service-related, and geographic details for 7,043 customers, with 23 predictors (20 categorical, 3 numerical) and a binary response variable for customer service status. After exploratory data analysis and data preprocessing, we developed predictive models for estimating individual customer churn probabilities. Various classifier methods including SVM, Random Forest, XGBoost, Logistic Regression, and GAM, evaluated using AUC, BER, F1 score, and accuracy. An inference model-mixed model is also built to analyze the variability of Churn preference among cities. Findings suggest that GAM can be particularly effective, sometimes outperforming machine learning models in certain contexts capturing the non-linearity by spline. Variations are observed in the relationship between churn preference and tenure, as well as between churn and contract types across different cities. This comprehensive study not only enhances predictive accuracy but also informs strategic decision-making in customer retention.

2 Introduction

In the dynamic and competitive arena of modern business, customer retention emerges as a pivotal factor in securing a company's growth and long-term viability. A critical challenge in this domain is managing customer churn—the rate at which customers end their relationship with a service—which can profoundly affect the financial health of an organization. This analytical endeavor is centered on a rich data set, originally collected by IBM and made accessible by SIBELIUS5 on Kaggle, to unearth the intricacies of customer behaviors and characteristics.

Our team will investigate factors that could potentially contribute to the occurrence of churns. The dataset we will work with contains a combination of demographic data (such as age, gender, and marital status), service-related information (including subscription tenure and monthly expenses), and geographic specifics (like zip codes and latitude) for 7043 customers. With a collection of 23 predictors— 20 of which are categorical and 3 numerical—and a single response variable that records the customer's service status from the previous month, our analysis is well-equipped to identify the most impactful predictors of churn.

The ultimate goal of our analysis is twofold: to pinpoint the key factors that are indicative of a propensity to churn, and to devise a predictive classifier capable of calculating the probability of churn for individual customers. Accurately identifying customers at a high risk of churn will enable the company to deploy targeted retention strategies and enhance customer loyalty and satisfaction

3 Exploratory Data Analysis (EDA)

Before diving into the main analysis, it is crucial to explore the dataset to gain a better understanding of the data and identify any potential issues that may affect the subsequent steps. The EDA process involves the following steps:

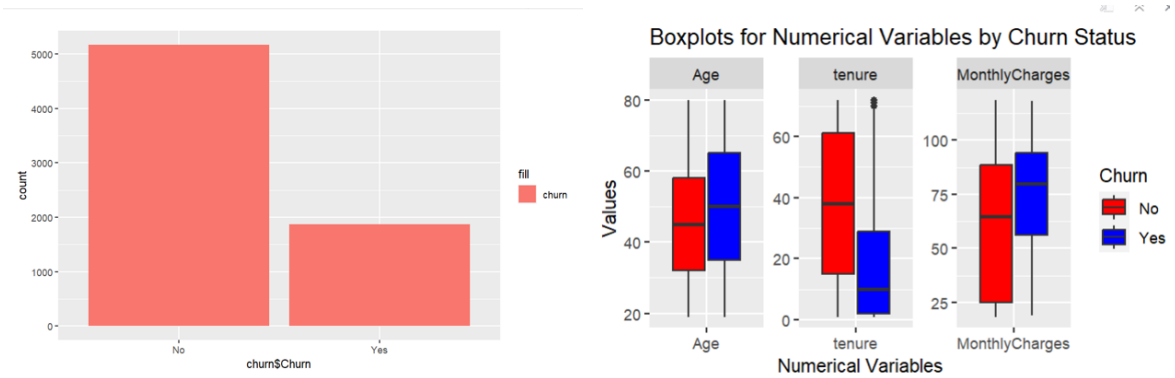
3.1 Data Integration Check

Our dataset contains 7,043 rows and 24 columns. After a brief summary, we found that 11 rows contain missing values. Because 11 is relatively small compared to 7,000+, we decided to delete those 11 rows. After further analysis, we found there are 20 categorical variables, 3 numerical variables, and 1 response (0-1 categorical) variable in our dataset. One can divide all predictors into the following clusters: demographic variables, service-related variables, payment-related variables, and tenure (the number of months that a customer stayed with a company).

Variable collinearity is a critical problem in model fitting, especially for a regression model. After calculating the correlation between categorical variables using Cramer's V coefficient, we found that some categories are completely linearly correlated. For example, when Internet Service is "No," Online Security will be "No Internet Service," indicating complete linear dependency. This issue also applies to Online Backup, Device Protection, Tech Support, Streaming TV, and Streaming movies, as these services are all related to Internet service. Our solution is to merge the "No Internet Service" category with the "No" category. It helps us avoid numerical issues and gain better interpretability of our model. Additionally, we found that the product of monthly charges and tenure is strongly correlated with the total charges (correlation is 0.99). We decided to delete yearly charges because it cannot provide us with additional information.

3.2 Variable Distribution Check

Exploring the distribution of predictors and response variables is quite important in Exploratory Data Analysis (EDA). By examining Figure 1, we can observe that the distribution of our response variable "Churn" is unbalanced. Therefore, we will use synthesized data points to improve our model in later sections.



(a) Churn

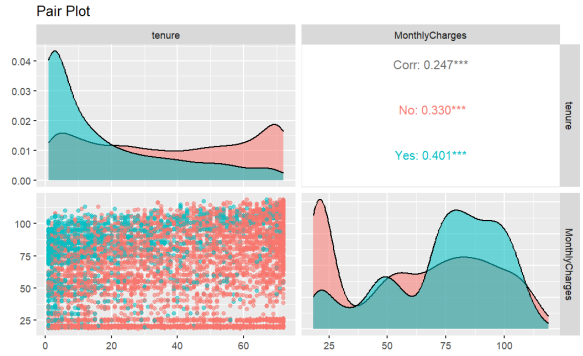
(b) Numerical Variales

Figure 1: missing data and imbalenced labels

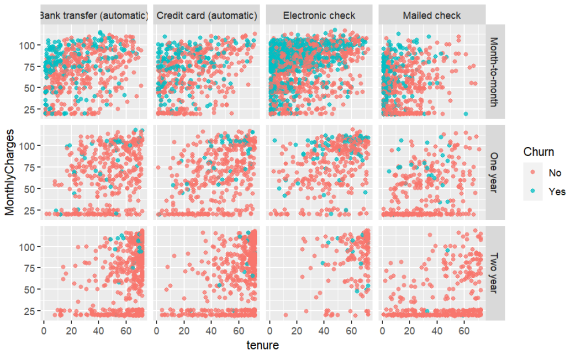
From Figure 2, we found a few vital insights. In the left panel, we can see the blue points concentrated on upper left corner, which means customers with high monthly charges and low tenure tend to churn. In the right panel, we can derive the following 5 insights.

1. The customers that have short-term (monthly) contract tend to churn.
2. The customers that have a larger tenure tend to sign a contract with a longer term.
3. The customers that use electronic check as the payment method tend to sign a monthly contract.
4. The customers that have a larger tenure tend to use automatic payment.
5. The majority of customers choose monthly contract.

In short, we also developed a KPI called customer loyalty to summarize the customer churn behavior. The details are in Figure 3



(a) Monthly Charges VS Tenure



(b) Facet Plot of Mixed Variables

Figure 2: missing data and imbalenced labels

High Loyalty	Low Loyalty
Yearly Contract	Monthly Contract
High Tenure	Low Tenure
Automatic Payment	Manual Payment
Minority of Customer	Majority of Customer
Tend to Stay	Tend to Churn

Figure 3: Customer Loyalty

4 Data Preprocessing

- Data Encoding

  We use dummy encoding to encode the categorical variables. In dummy encoding, we create new columns for each category in the original columns except the reference category and assign a value of 1 or 0 to indicate whether a particular category is present or absent in each row. If an object belongs to the reference category, then all values in the created columns will be 0.

- Data Splitting

  We randomly split our data into training set(80%) and test set(20%). Training set will be further divided into training and validation sets in the Cross-Validation approach in the our classification methods. Training set will be used to train our models, while validation set will be used to find hyperparameters of our models. Testing set will be used to evaluate the performance of our models.

- Data Augmentation

  We use Synthetic Minority Oversampling Technique (SMOTE) to handle the imbalanced data. It duplicates examples in the minority class.

5 Model Construction & Results

We will try different classifier methods including: Support Vector Machine, Random Forest, XGBoost, Logistic Regression, and Generalized Additive Model(GAM). Besides that our group will use ROC curve, AUC score, BER score, F1 score, as well as accuracy to evaluate the effectiveness of the machine learning models. We also build a mixed model to analyze the variability across cities.

5.1 Support Vector Machine

SVM is a popular ML algorithm for classification tasks. It handles linear and non-linear data efficiently, works well in high-dimensional spaces, and avoids overfitting. SVM is a versatile and effective tool for complex but small or medium-sized datasets.

Before fitting the Support Vector Machine, 'Satisfaction.Score' was dropped from the dataset since it is irrelevant to the classification. The model was fine-tuned through GridSearchCV. GridSearchCV function from scikit-learn systematically explores hyperparameter combinations using cross-validation to find the optimal combination resulting in the best performance (ROC AUC) on training data. This sets up a Support Vector Machine with a polynomial kernel, regularization strength of 0.1, automatically adjusted influence of training examples, probability estimates enabled, and reproducibility ensured with a random state set to 10. The Support Vector Machine (SVM) showed a high level of accuracy in its predictions, with an Area Under the Curve (AUC) of 0.8176. This score indicates that the model distinguishes between positive and negative classes. Additionally, the Balanced Error Rate (BER) of 0.3260 and F1 score of 0.5185 demonstrate the model's effectiveness in handling class imbalances, a common challenge in churn datasets.

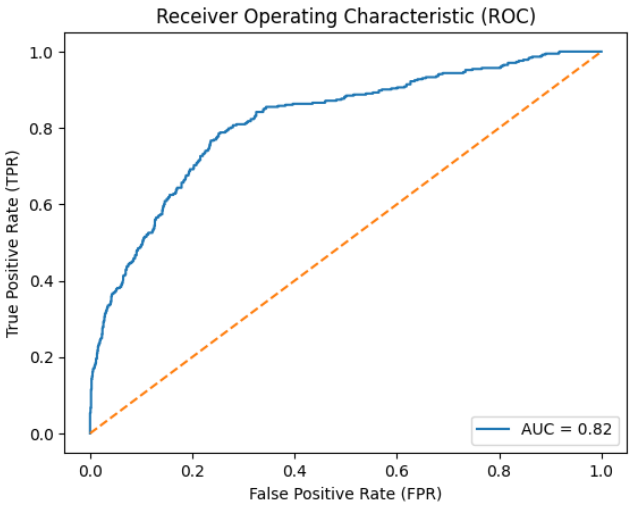


Figure 4: ROC of Support Vector Machine

A support vector classifier using a polynomial kernel has created a classification boundary plot against two numerical features, "Tenure" and "Monthly Charges." The plot displays two classes, represented by two different colors - commonly red and blue. The red region indicates the "Churn" class, while the blue region represents the other class. However, the overlap of red and blue points suggests that the model is facing challenges in fully separating the two classes linearly, even with the polynomial transformation. This misclassification is occurring around the boundary, indicating that the model is struggling to identify correctly which class some data points belong to.

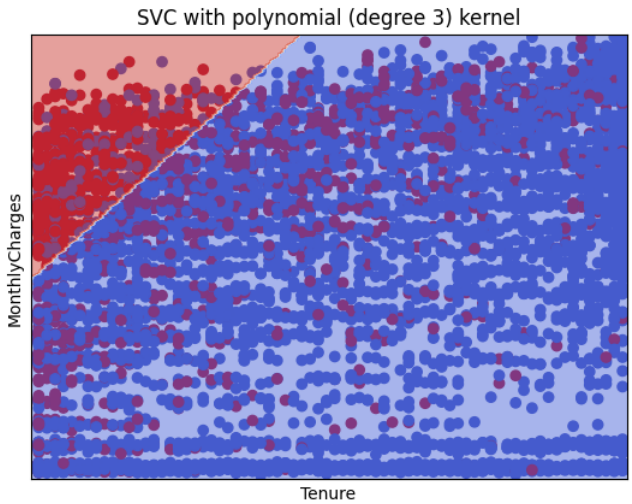


Figure 5: classification boundary of Support Vector Machine

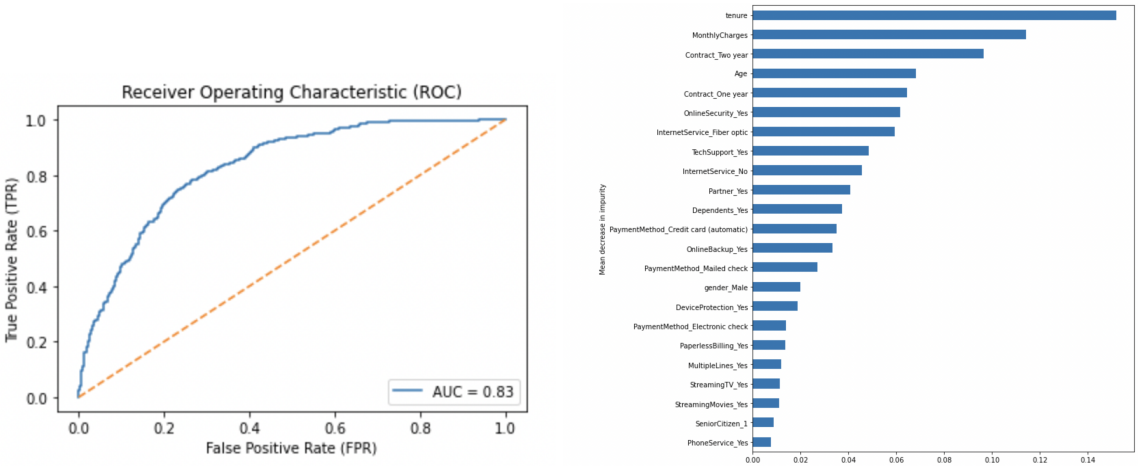
5.2 Random Forest

The Random Forest algorithm, a popular method in predictive analytics, operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes of individual trees. This ensemble approach enhances the predictive accuracy and controls overfitting, making it highly effective in complex tasks like churn prediction. The algorithm's ability to handle both categorical and continuous data, as well as its robustness against imbalanced datasets, makes it particularly suitable for analyzing customer churn, a scenario where data often exhibits such characteristics.

The model was fine-tuned through GridSearchCV, optimizing for parameters that best predict customer churn. This involved setting the maximum depth of trees at 12, using the square root of the number of features, and constructing 200 trees. The combination of these parameters was specifically tailored to enhance the model's performance in predicting churn. The Random Forest model exhibited high predictive accuracy with an AUC of 0.831. This high value indicates the model's strong capability in distinguishing between potential churners and loyal customers. The BER of 0.253 and F1 score of 0.611 signifies the model's effectiveness in managing class imbalances, a typical challenge in churn datasets.

The feature importance graph from the Random Forest algorithm provides valuable insights into which factors are most predictive of customer churn in the telecommunication sector. The model indicates that 'tenure', 'MonthlyCharges', and 'Contract Two year' are the most significant predictors, suggesting that customer loyalty and financial considerations play crucial roles in retention.

However, Random Forest's complexity can obscure the rationale behind its predictions, posing challenges in situations where interpretability is crucial for decision-making. Despite its predictive prowess, this opacity may restrict its use where decision transparency is paramount.



(a) (b)

Figure 6: ROC and Predictor Importance of Random Forest

5.3 XGboost

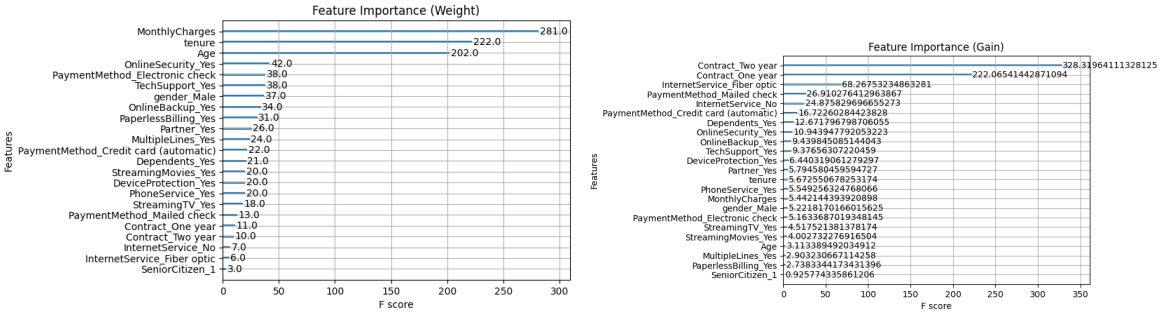
The next prediction model we explored is XGBoost, an advanced form of gradient boosting. Like Random Forest, XGBoost is a tree-based ensemble learning technique, but there are key differences in their approaches. While Random Forest employs a bagging algorithm, creating trees in parallel, XGBoost uses a boosting algorithm, building trees sequentially. Each new tree in XGBoost is designed to correct the errors made by the previous ones.

One of the notable strengths of XGBoost is that it does not require data standardization. Since it splits data based on conditions, scaling does not impact the outcome. Furthermore, XGBoost handles large datasets efficiently and provides valuable insights into feature importance, aiding in understanding the model's decision-making process. However, XGBoost can be prone to overfitting if the hyperparameters are not tuned carefully. This is particularly relevant when dealing with complex models or smaller datasets where the model might learn the training data too well, failing to generalize to new, unseen data.

In our implementation, the XGBoost model was configured with a maximum depth of 8 and a learning rate of 0.3. The performance was evaluated using the Area Under the Curve (AUC) metric, achieving a score of 0.8180. Other metrics, such as the Balanced Error Rate (BER) at 0.2730 and the F1 score at 0.5872, also indicated strong performance.

Feature importance analysis was a crucial part of our model evaluation. The weight-based feature importance plot revealed that numerical variables like MonthlyCharges, tenure, and age had higher weights compared to categorical variables. This is because numerical variables provide more potential split points. Additionally, the gain-based feature importance plot highlighted that features like contract type, internet service, and payment method significantly enhanced model performance. This gain value measures the improvement in accuracy brought by a feature to the branches it is on, offering a more informative perspective than just frequency of use in splits.

These insights from the XGBoost model not only affirm its predictive power but also provide a deeper understanding of the factors influencing the Churn perference, which is invaluable for making informed decisions based on the model's outputs.



(a) Weight (b) Gain

Figure 7: Weight and Gain Feature Importance of XGBoost

5.4 Logistic Regression

5.4.1 Model Diagnostic

Since we have a binary response, the regular residual plot will not work as well as for a continuous response. Thus, we started the diagnostic process by computing the linear predictor and predicted probabilities using the logistic regression model. Next, the data is grouped into bins based on the quantiles of the linear predictor, allowing for an organized analysis of the predicted probabilities. Within each bin, the observed proportions of actual churn occurrences are summarized, along with the mean predicted probabilities and the count of observations. Standard errors are then calculated to quantify the uncertainty in the predicted proportions. The plot illustrates the relationship between the predicted probabilities and the observed proportions of churn, which shows it is a good-fit to use logistic regression.

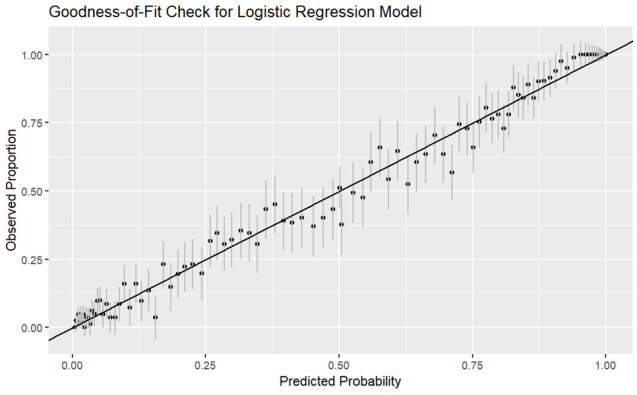


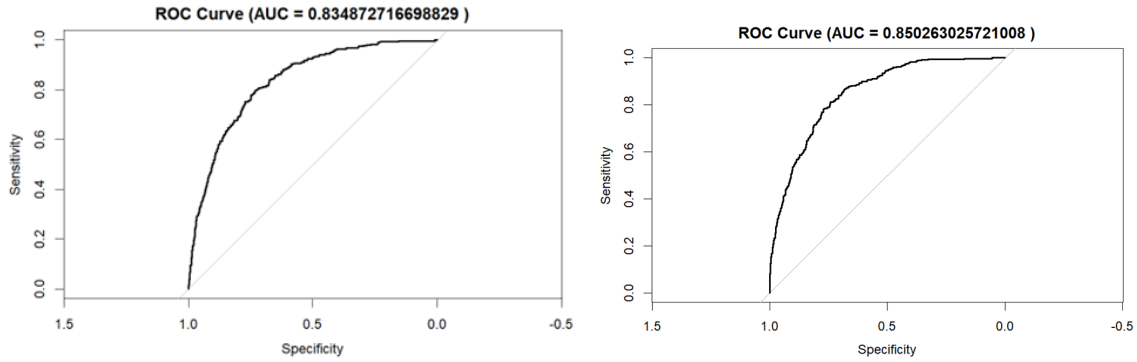
Figure 8: Goodness-of-Fit Check for Logistic Regression Model

5.4.2 Logistic Regression Analysis

We conducted a logistic regression analysis on a data set with balanced class weights being balanced due to the nature of the data set. Initially, all features, comprising 20 categorical predictors and 3 numerical predictors, were included in the model. The model, when applied to the full set of features, yielded a balanced error rate of 0.277 and a ROC area under the curve score of 0.834. Subsequently, quasi-binomial regression was employed, resulting in a dispersion parameter of 1.006, indicating the absence of over-dispersion. Following initial testing, we utilized a backward elimination method for feature selection, leading to the removal of two features: PaperBilling-Yes and Senior-Citizen-Yes. Subsequent to bootstrapping with a sample size of 200 and employing 10-fold cross-validation, the balanced error rate on the test data slightly decreased to 0.276, while the ROC area under the curve score saw a slight increase to 0.835.

5.5 GAM

A General Additive Model (GAM) was explored initially incorporating all features into the model. Then to reduce the complexity and dimension, a backward elimination process was implemented. We got rid of Paper-Billing-Yes, Senior-Citizen-Yes, Partner-Yes, but the resulting changes in AUC and BER were negligible. To reduce the complexity of the model, splines were exclusively applied to three numerical variables—Age, Total Monthly Charges, and Tenure. Introducing the parameter for the number of kernels, initial testing with k = 20 exhibited a substantial drop in effective degrees of freedom (edf). Further experimentation with k = 15 demonstrated minimal alterations in edf, leading to the decision to proceed with 15 kernels. During the process, the 'REML' method outperformed 'GCV' in terms of AUC, suggesting a potential scenario where numerous variables might be present, but information is comparatively sparse. To refine the spline, additional parameters were introduced. Employing b-spline for all three variables—Age, Tenure, and Monthly Charges—yielded superior results compared to cubic spline and other combinations of cs and bs. This also leads to the similar conclusion that our data needs simpler models instead of more complex ones. So I keep eliminating the categorical variables according to the rank of the importance given by the Random Forest approach, and it turns out the model with the best outcome is only with 6 categorical variables and 3 numerical variables. (Age, MonthlyCharges, tenure, InternetService-Fiber.optic, InternetService-No, OnlineSecurity-Yes, TechSupport-Yes,Contract-One.year, Contract-Two.year) The best outcome we get after Cross-Validation is having AUC = 0.85 and BER = 0.28.



(a) ROC of Logistic Regression

(b) ROC of General Additive Model

Figure 9: ROC of Logistic Regression and ROC of General Additive Model

5.6 Mixed Model

The dataset reveals that the majority of cities are concentrated in the southwest part of the United States, particularly around Los Angeles and San Diego. Notably, different cities may exhibit distinct Churn preferences. Our objective is to construct models that not only accommodate the variability across cities but also facilitate a detailed analysis of this variability. To this end, we have chosen to employ a mixed model approach. These models are termed 'mixed' due to their incorporation of both fixed and random effects, making them particularly suitable for analyzing data with dependent structures, such as grouped or hierarchical data.

Selecting 'City' as a random effect is justified for several reasons. Initially, the data distribution across cities, as shown in the following table, indicates a reasonable sample size per city; most cities report 4 or 5 data points, with only 4 cities having as few as 3 data points. This sample size is adequate for considering 'City' as a random effect. Moreover, mixed models are adept at handling cluster groups of unequal sizes, a characteristic feature of our dataset. Furthermore, with over 1000 cities in the dataset, treating 'City' as a fixed effect is impractical. Our primary focus is not on pinpointing the precise differences in Churn results between specific regions, but rather on understanding broader regional trends and patterns.

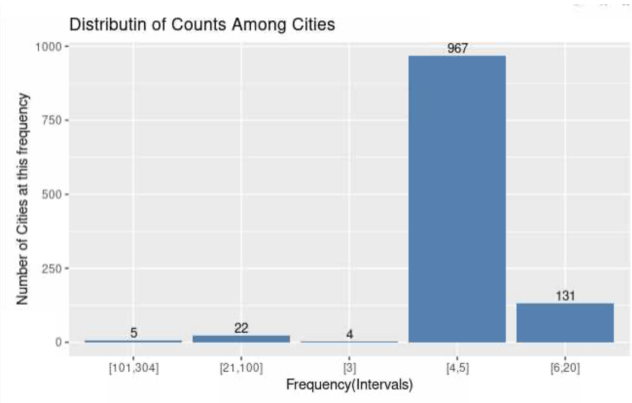


Figure 10: Distribution of Counts Among Cities

We select some important variables from the exploratory data analysis to fit the mixed models. We first fit a logistic mixed effect regression with random cities intercepts(M1).

$$y_{ij} = \beta' \mathbf{x}_{ij} + \theta_i + \epsilon_{ij} \quad (M1) \tag{1}$$

where $i = 1, \ldots, n$ indexes cities, and $j = 1, \ldots, n_i$ indexes observations within one city, with n_i being the size of city i. $\beta' \mathbf{x}_{ij}$ is the fixed effect, θ_i is the random effect with mean zero and variance σ_θ^2, and ϵ_{ij} is the residual with mean zero and variance σ_ϵ^2.

Analyzing the output, we observe that the p-values for the fixed effect variables are all below 0.05, indicating their statistical significance. Additionally, the random effect's standard deviation is 0.1183, which reflects the variability in the log odds of Churn across different cities. The 95% confidence interval for this standard deviation ranges from 0 to 0.30, suggesting that the lower 2.5% quantile of the random effect's standard deviation is 0. This indicates a possibility that the variance attributable to the random effect might be minimal.

To further explore the impact of cities on the relationship between each predictor and Churn preference, we implemented a logistic mixed effect regression model. This model includes both random city intercepts and random tenure slopes for cities (Model M2). In a similar vein, we also fitted models with random slopes for categorical predictors: PaperlessBilling (Model M3), InternetService (Model M4), SeniorCitizen (Model M5), and Contract (Model M6). These models help us understand how the influence of these predictors on Churn preference varies across different cities.

The equation for this kind of model is:

$$y_{ij} = \beta' \mathbf{x}_{ij} + \theta_i + \gamma_i x_{1ij} + \epsilon_{ij} \tag{2}$$

where $i = 1, \ldots, n$ indexes cities, and $j = 1, \ldots, n_i$ indexes observations within one city, with n_i being the size of city i. $\beta' \mathbf{x}_{ij}$ is the fixed effect, θ_i is the random effect with mean zero and variance τ^2. γ_i is a latent random city slope with mean zero and variance τ_1^2, and x_{1ij} is the variable that has a random city slope. ϵ_{ij} is the residual with mean zero and variance σ^2. Besides, θ_i and γ_i are uncorrelated, with $\text{cov}(\gamma_i, \theta_i) = 0$.

We use a likelihood ratio test to compare pairwise nested models.

Examining the results table, we note that the p-values for Anova comparisons between Models M1 and M2, and between M1 and M6, are less than 0.05. This indicates that incorporating an additional random slope for tenure (in M2) or for Contract (in M6) significantly improves the model fit compared to the model with only a random city intercept (M1). However, for other models with additional random slopes, the improvements in fit are not statistically significant. This suggests some variations in the relationship between churn preference and tenure, as well as between churn and contract types across different cities.

Model 1	Model 2	p-values of Anova(Model 1,Model 2)
M1	M2	0.007
M1	M3	0.2236
M1	M4	1
M1	M5	0.9083
M1	M6	3.321e-12

Table 1: Likelihood Ratio Tests

From M2's output, we find that the p-values for all fixed effect variables are below 0.05, signifying their significance. The standard deviations for the random city intercept and the random slope for cities are 0.01586 and 0.01440, respectively. M6's output reveals a standard deviation of 0.1773 for the random intercept, and for the random slopes of Contract types—Month-to-month, One year, and Two year—the standard deviations are 0.0118, 0.8890, and 10.1561, respectively.

The Intraclass Correlation Coefficient (ICC) is a measure used to gauge the degree of resemblance within units of the same group. It represents the proportion of variance between groups relative to the total variance (the sum of between-group and within-group variances) and ranges from 0 to 1. The ICC values do not necessitate equal group sizes, nor do they depend on the groups' ordering or labeling. Higher ICC values indicate greater disparities between groups. For our models, the ICCs for M1, M2, and M6 are 0.004, 0.000, and 0.009, respectively.

We have decided against adopting more complex models, such as those with correlated random intercepts and slopes, to avoid potential issues of singularity or overfitting. Our aim is to strike a balance between model fit and complexity, ensuring that our model remains robust and interpretable.

5.7 Performance Comparison

In our comprehensive evaluation, we utilized various metrics to assess the performance of five distinct models: Logistic Regression, General Additive Model (GAM), Random Forest, SVM and XGBoost. The metrics employed for evaluation include the Area Under the Curve (AUC) score, Balanced Error Rate (BER) score, F1 score, and overall Accuracy. The AUC score provides insights into the model's ability to distinguish between positive and negative classes, with higher values indicating superior performance. The BER score accounts for the balance between sensitivity and specificity. F1 score, the harmonic mean of precision and recall, showcases the model's precision in identifying positive instances. Lastly, Accuracy represents the overall correctness of the model's predictions.

Since we have unbalanced data, accuracy would take less important role. And when we compare Logistic Regression and Random Forest, we can see 3 of 4 metrics have similar results. But when we add in spline and move to a GAM model, both AUC and F1 have a increase. This indicates the relationship in the data has significant nonlinear components. The other potential reason can also be the optimal predictor selection to improve model proficiency and performance. At the same time, we cannot ignore the capability from Random Forest to handle imbalanced dataset since it reaches the lowest BER.

Table 2: performances of models on different metrics

Methods	AUC Score	BER Score	F1 score	Accuracy
Logistic Regression	0.834	0.277	0.611	0.785
GAM	**0.850**	0.280	**0.633**	0.764
Random Forest	0.831	**0.253**	0.611	0.775
Svm	0.818	0.326	0.519	**0.797**
XGBoost	0.818	0.273	0.587	0.7661

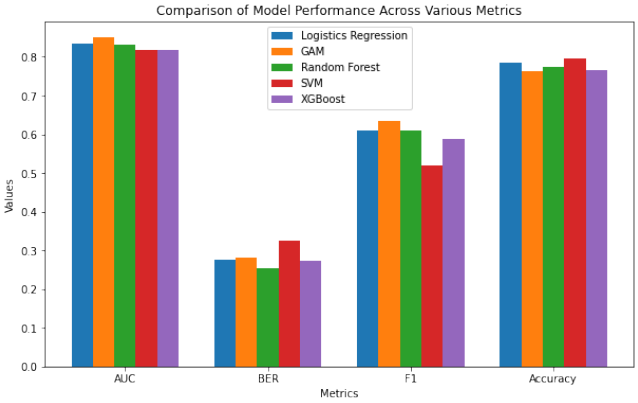


Figure 11: Comparison of Model Performance Across Various Metrics

6 Conclusion/Future Work

In certain scenarios, regression models like logistic regression or generalized additive models (GAM) may outperform machine learning models such as Random Forest, particularly when non-linearity exhibits a specific smooth structure that can be accurately represented using splines. However, it is always better to combine different methods. For example, leveraging insights from Random Forest predictions has proven to enhance the performance of GAM by reducing non-efficient information. Moreover, beyond predictive capabilities, gaining a deeper understanding of the data can open avenues for future research. We can also consider combining Random Forest with other techniques, like Support Vector Machine, to create hybrid models that can capture complex patterns in data more effectively. In the context of mixed models, variations are observed in the relationship between churn preference and tenure, as well as between churn and contract types across different cities.