# Project_Report_Stats_Final

### Introduction

In the complex landscape of insurance services, complaints management plays a crucial role in customer satisfaction and operational excellence. In the realm of complaint management and fraud detection within organizations, accurately classifying and predicting the nature of complaints can be pivotal for timely and effective resolutions Every complaint presents a unique set of data points that, when systematically analyzed, can provide profound insights into the service's strengths and weaknesses. These complaints are characterized by various attributes that outline the specifics of each case, ranging from the type of insurance coverage involved to the details of how each complaint was resolved.

### Objectives

The primary objective of this project is to analyze insurance complaint data to understand the dynamics behind customer complaints and to identify patterns that may help predict future complaints or areas needing improvement. By dissecting data across several key dimensions such as complaint type, resolution efficacy, and the nature of the insurance coverage, the project aims to:

1. **Enhance Customer Service:** Analyze complaint resolution strategies to identify areas for improvement, ensuring quicker and more effective responses to future complaints.
2. **Improve Resource Allocation**: Understand the distribution of different types of complaints (e.g., "Customer Service" vs. "Denial of Claim") to better allocate resources and training where they are most needed.
3. **Risk Management**: Explore the relationship between complaint types, confirmed complaints, and other variables to identify potential risks and operational vulnerabilities.
4. **Product and Service Development**: Utilize insights from complaint data, including specific keywords and coverage details, to guide enhancements in insurance products and service offerings.

This objective can be achieved by examining the relationships between various attributes of complaints. By accurately predicting the complaint type, organizations can tailor their investigative and resolution approaches to be more effective and efficient.

### Dependent Variable (Y):

- **Complaint Type:** This categorical variable is crucial as it represents the outcome we aim to predict. The classification into INDV and ORG provides a binary yet significant differentiation in the nature of complaints, which can have varied implications and required actions.

### Independent Variables (X):

- **Complaint Filed Against**: Knowing against whom the complaint was filed can offer insights into potential targets or departments within an organization that are prone to complaints. This variable is essential for identifying patterns that may suggest systemic issues or areas vulnerable to misconduct.

- **Reason Complaint Filed**: This variable encapsulates the motivations or allegations behind the complaint, such as false claims or suspicious activities. It is a critical indicator of the underlying nature of the complaint and can be directly correlated to identifying fraudulent practices.

- **Coverage Type:** In contexts like insurance, the type of coverage can influence the propensity for disputes or complaints. Different coverage types may have different levels of complexity and potential for misunderstanding or misrepresentation, thereby impacting the likelihood of fraud.

- **Coverage Level**: This closely relates to the type of coverage but provides deeper insight into the scope or extent of the coverage involved in the complaint. Higher levels of coverage might be associated with higher stakes, potentially leading to more significant disputes or complexities.

- **Others Involved**: Identifying other parties involved in a complaint is crucial for understanding its context and scope. Inclusion of third parties or external entities can complicate the nature of the complaint and is often a red flag for potential fraudulent collusion.

- **Keywords**: The use of specific keywords in the complaint documentation can reveal much about the complaint's nature. Keywords such as "unauthorized", "excessive", or "denied" can be strong indicators of the issues at stake and are valuable for automated text analysis and classification.

- **complaint_number**: A unique identifier for each complaint, which could be used to track or reference the complaint in a database.

- **complaint_filed_against**: The name of the company or entity against which the complaint is filed. It is the insurance provider or network being complained about.

- **complaint_filed_by**: The individual or entity filing the complaint, such as a "Non-Contracted Provider". This identifies the complainant's relationship or role.

- **confirmed_complaint**: Indicates whether the complaint was confirmed as valid or not, often after preliminary investigation.

- **how_resolved**: Describes the method or outcome of how the complaint was addressed, such as "Information Furnished" indicating that providing additional information resolved the complaint.

- **received_date** and **closed_date**: The dates when the complaint was received and when it was officially closed, respectively. These are useful for tracking the duration and timing of complaint resolution.

- **respondent_id**: Identifier for the entity responding to the complaint, which could be used internally for tracking responses.

- **respondent_role** and **respondent_type**: These describe the role and type of the respondent (like "WC Healthcare Provider Ntwk" and "Organization"), providing context on who is handling or implicated in the complaint.
- **complainant_type**: Specifies the category of the complainant, such as "ORG" for an organization, indicating the nature of the complainant.
- **date_difference**: Represents the number of days between the received date and the closed date, providing a measure of how long it took to resolve the complaint.

## 2. Data Import

Data was imported using R's functionalities to read CSV files using the function read_csv. The script specified the path to the CSV file located on the user's desktop and managed initial data type specifications. During import, it also handled the renaming of columns with vague names to more descriptive ones, ensuring the data frame started with clear, understandable labels.

## 3. Data Cleaning

The data cleaning process in your script includes several steps: -

- **Column Name Standardization**: `clean_names()` from the janitor package was used to standardize column names by removing extra spaces, converting all names to lowercase, and replacing spaces and non-alphanumeric characters with underscores.
- **Missing Data Handling**: Identification and **imputation** of missing values were conducted with the help of **custom function**, with missing values in specific columns filled using the most frequent value (mode).
- **Filtering Specific Complaint Types**: The data set was filtered to include only rows where `complaint_type` matched specific predefined values, ensuring focus on relevant data. Following are the specific selected factor levels "Workers Compensation Network", "Independent Review Org", "Teacher Retirement System", "Portal".
- **Trimmed the columns:** with leading and trailing empty spaces as well as removed special characters such as "?", "/" from the data columns with the help of **regex pattern** module **gsub** for all the columns using the functions lapply.
- **Removed Special Characters:** Cleaned the columns with the function **lapply** where we removed special characters such as "&", "-" and so on.

## 4. Data Tidying

Data tidying steps involved: -

- **Unnesting and Expanding Lists**: Columns containing lists, such as multiple complaint types separated by commas with the help of strsplit, were split and **expanded using unnest()**, giving

each **list** element its own row for the columns how_resolved, others_involved, reason_complaint_filed, keywords, complaint_filed_against, complaint_filed_against.

- **Date Handling**: The `received_date` and `closed_date` were standardized and converted to R date objects to ensure consistency and facilitate calculations such as the difference between dates.
- **Factor Conversion**: Several columns were converted to factor data types, which is essential for certain types of statistical analysis in R using factor().

## 5. Data Exploration

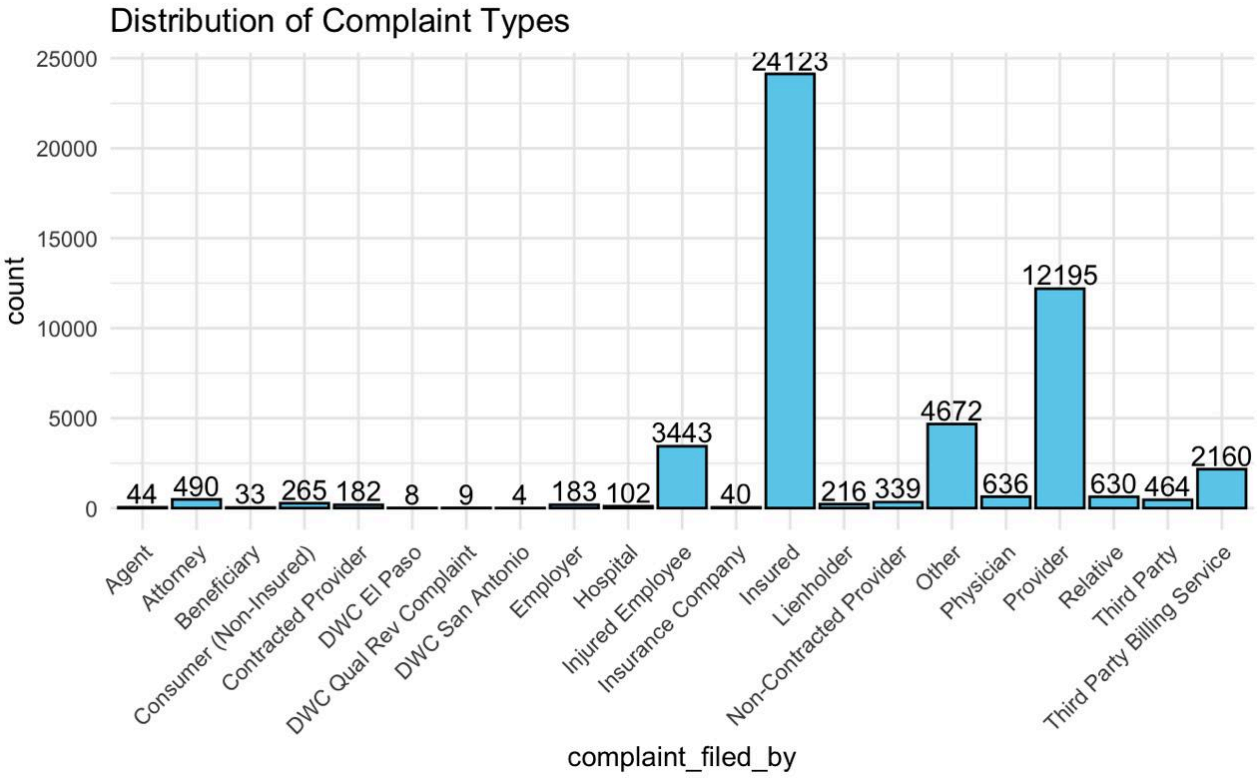
Key visualization and exploration techniques used: -

- **Unique Value Counts**: Extraction and plotting of unique values from specific columns to assess the variety and distribution of data.

- ![](images/clipboard-3962682488.png)

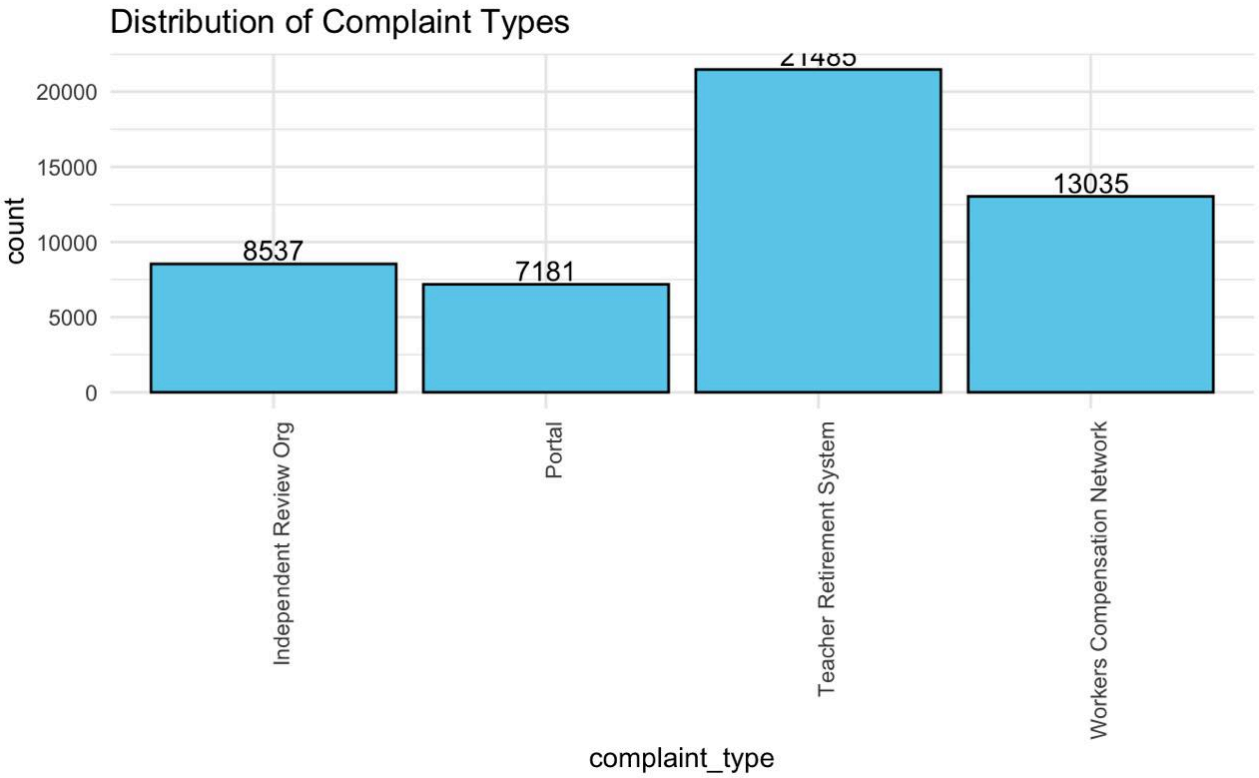
- The bar chart illustrates the unique value counts for various database columns, highlighting that the column "complaint_filed_by" has the highest count of unique values at 552, while the column "respondent_role" has the second highest at 447, and "coverage_type" follows with 406 unique entries.

- **Distribution Plots**: Visualized the distribution of complaints by type and coverage, aiding in identifying prevalent issues.

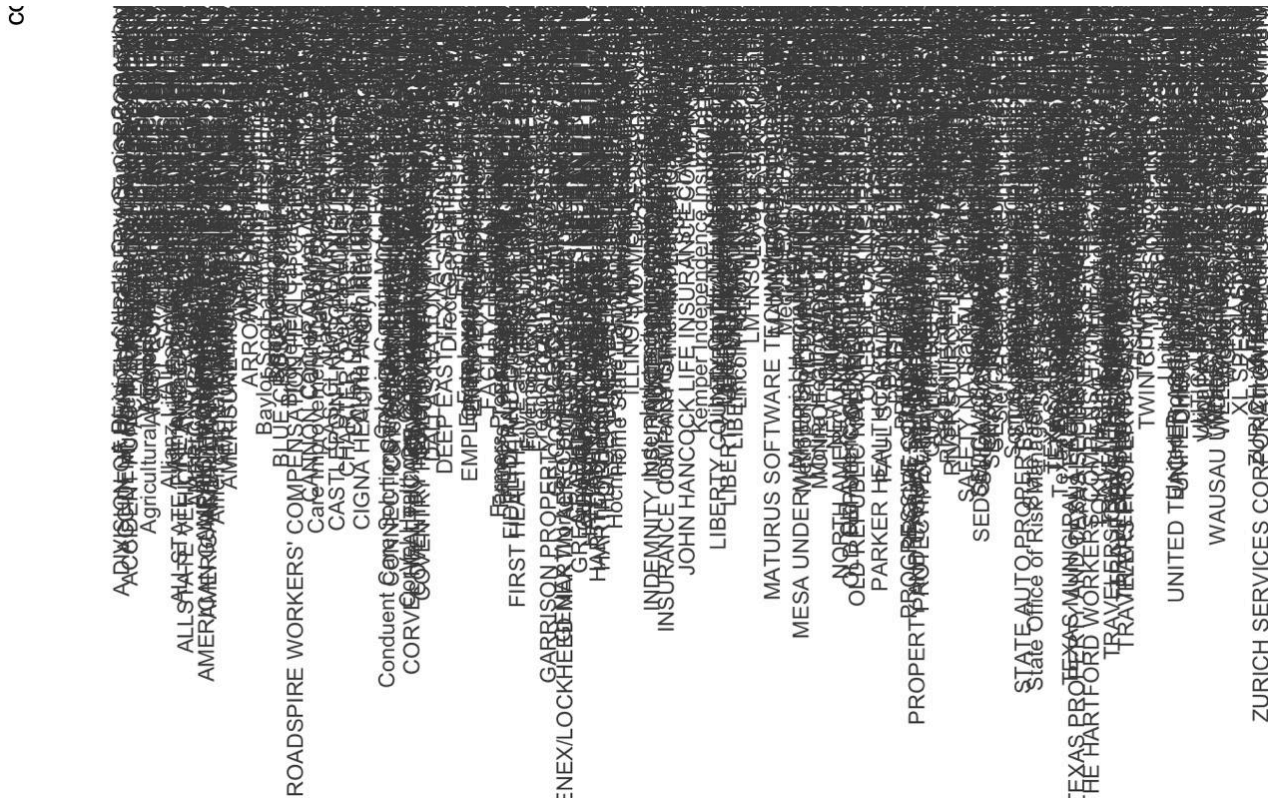



  - The bar chart shows the distribution of complaint types filed by various parties, with “Insured” filing the most complaints at 24,123, followed by “Provider” with 12,195 complaints, and much

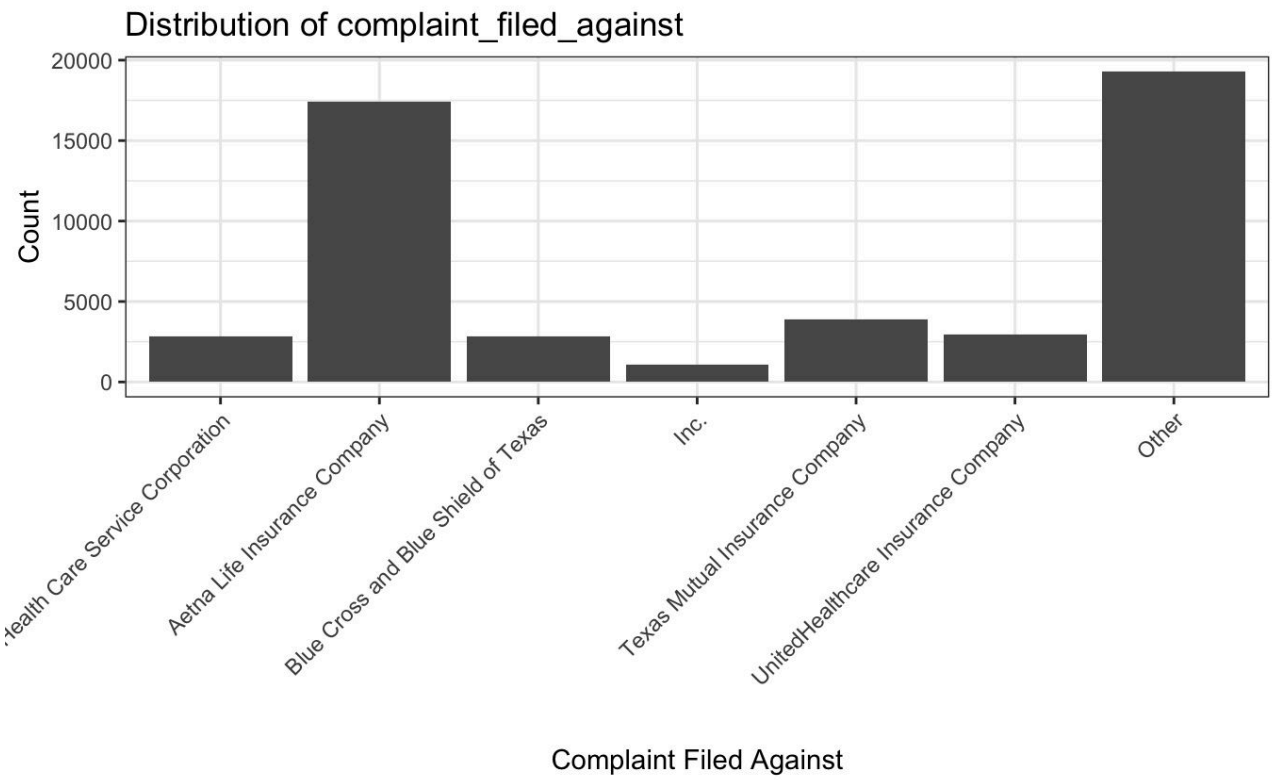
lower counts for other types such as “Agent,” “Attorney,” and “Beneficiary. Following is the plot to display the reduced levels in the response variable.




- The bar chart displays the distribution of complaint types with “Teacher Retirement System” receiving the highest number of complaints at 14,400, followed by “Workers Compensation Network” with 13,035 complaints, “Independent Review Org” with 8,537 complaints, and “Portal” with the fewest at 7,181 complaints.

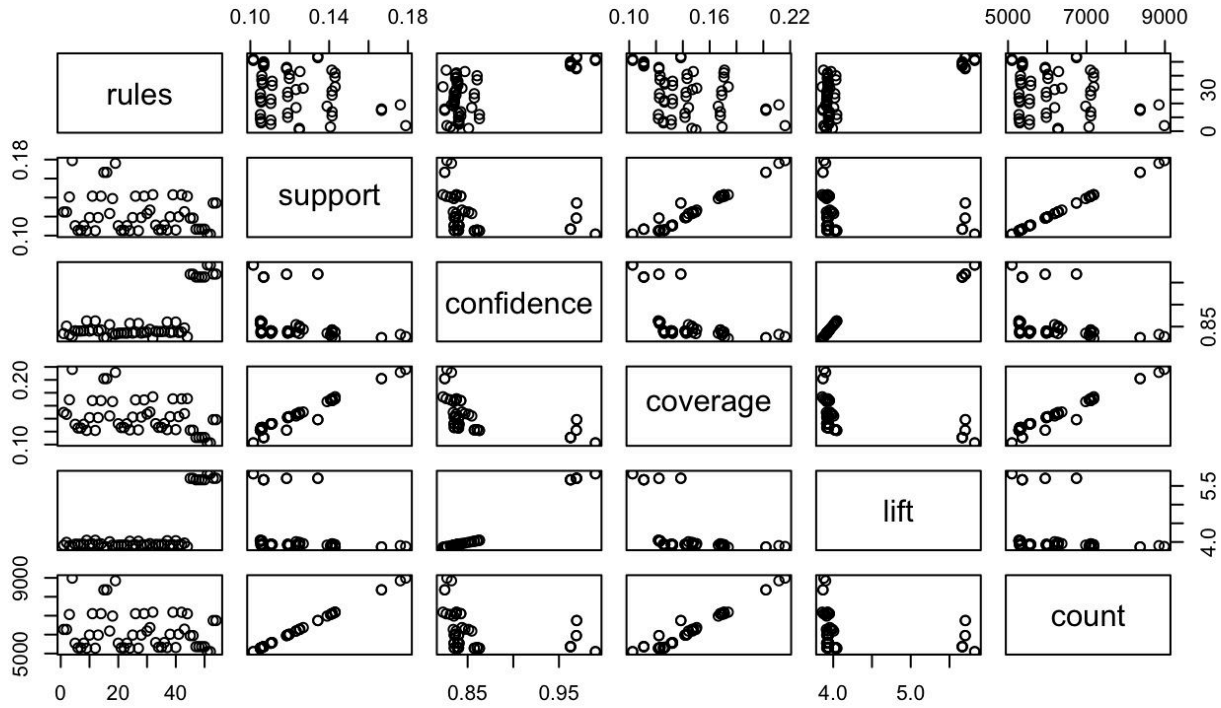


- Complaint_filled_against columns has around 552 factor levels which adding to the analysis reduces the statistical power even though we apply fancy models such as random forest.

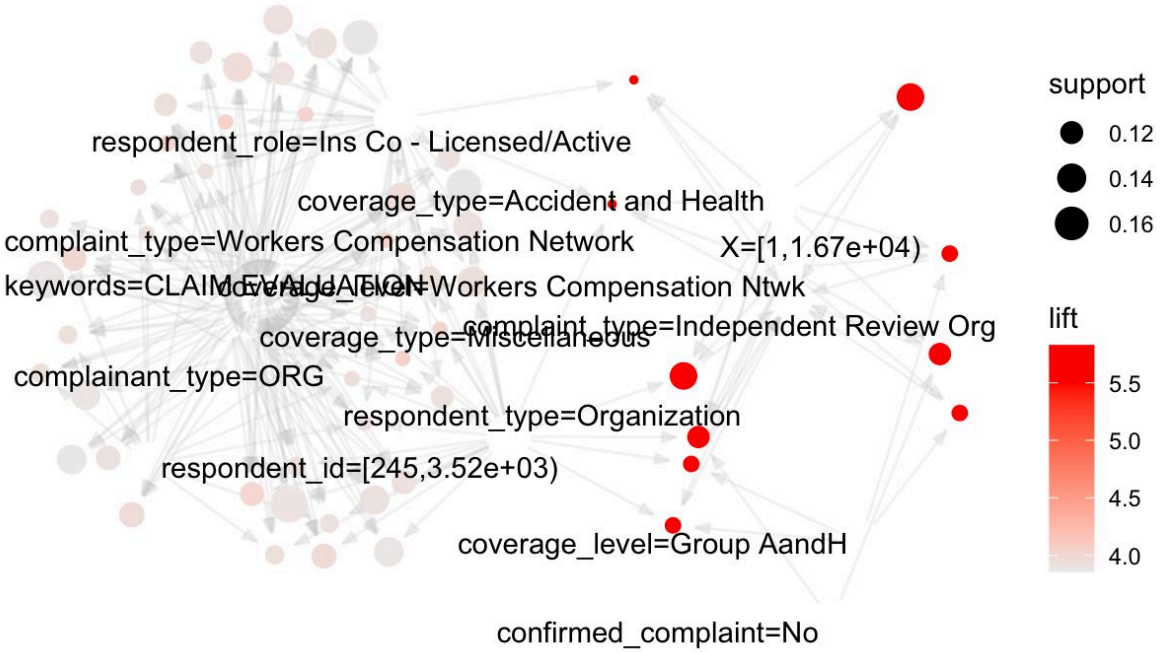



- Leveraged the functionality of step_other recipe of the workflow management to group the factors whose distribution is less than 2 percent.

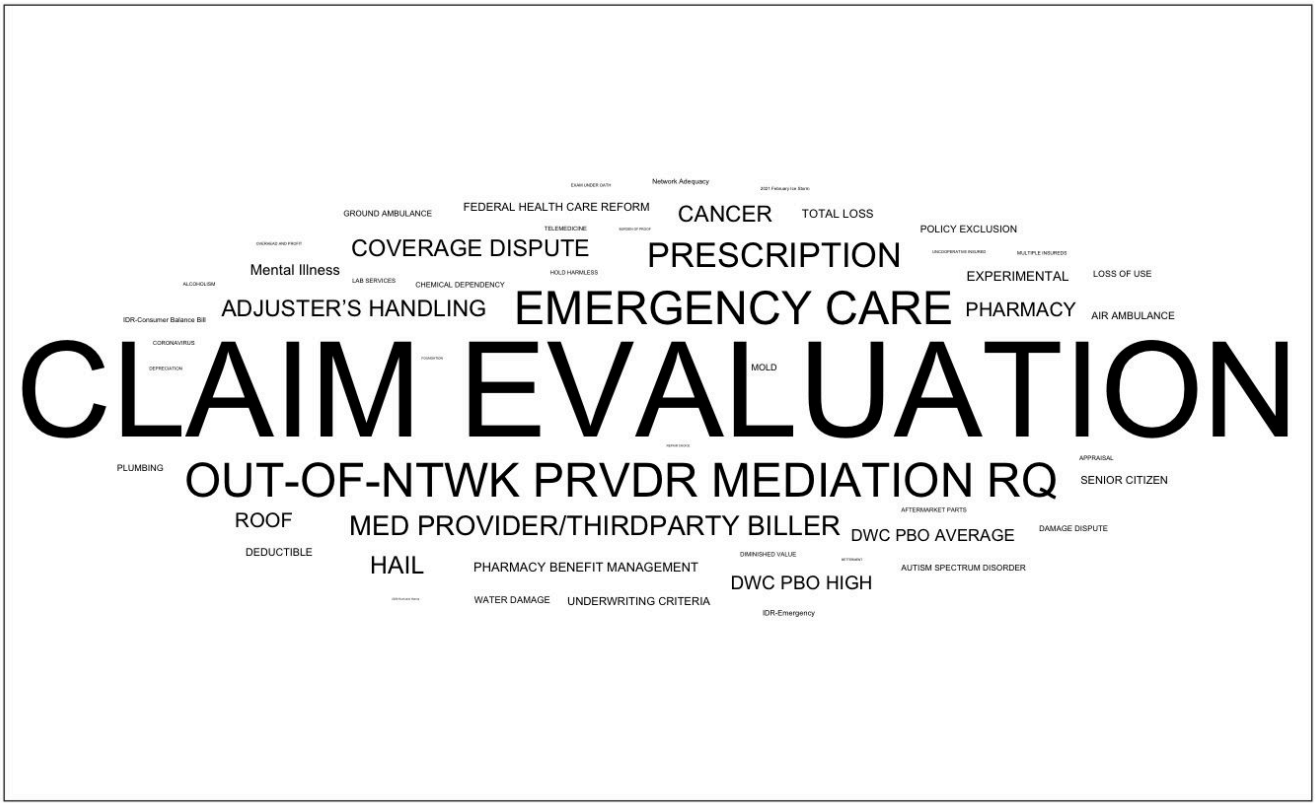
- **Scatter Plots for Confidence Intervals**: These plots help visualize the variability and confidence of certain estimates within the data.



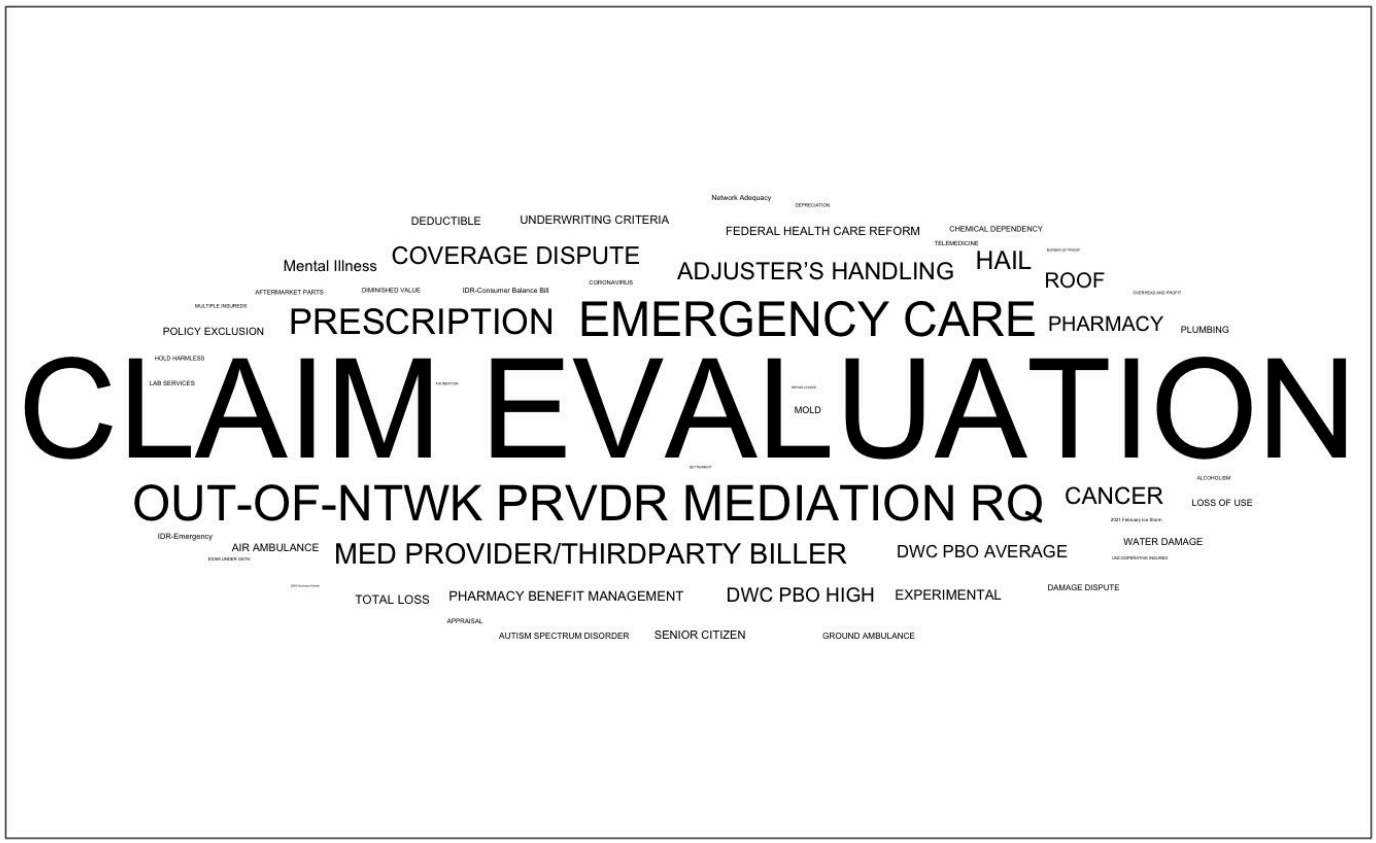

- **Apriori Algorithm Visualization**: Utilized to display the relationships between variables based on the strength of association rules.




- **Word Cloud Plots**: Generated from text data to visually summarize the most frequent terms associated with complaints, offering insights into common themes.




Following is the plot which was generated by eliminating the stop words from the above word cloud with the help of **anti-join command**




## 6. Modeling

The models implemented in the script to predict complaint types based on the dataset include:

- **Naive Bayes Classifier**: Effective for large datasets with independent predictors, it's often used for text classification due to its efficiency in handling multiple features.
- **Multinomial Logistic Regression**: Ideal for predicting categorical outcomes, it models the probabilities of different classes based on linear relationships among features.
- **K-Nearest Neighbors (KNN)**: A non-parametric method that classifies new cases based on the 'k' most similar instances in the training dataset, suitable for applications where patterns can be identified by proximity.
- **Random Forest**: An ensemble learning method that builds multiple decision trees and merges them to get a more accurate and stable prediction, great for handling large datasets with high dimensionality.
- **Decision Trees**: Useful for making strategic decisions based on hierarchical, sequential assessments of data features, it simplifies complex decision-making by breaking it down into a tree-like model of choices.

  These models were chosen for their ability to handle categorical data and their applicability to classification tasks, which are central to the analysis of complaint types.

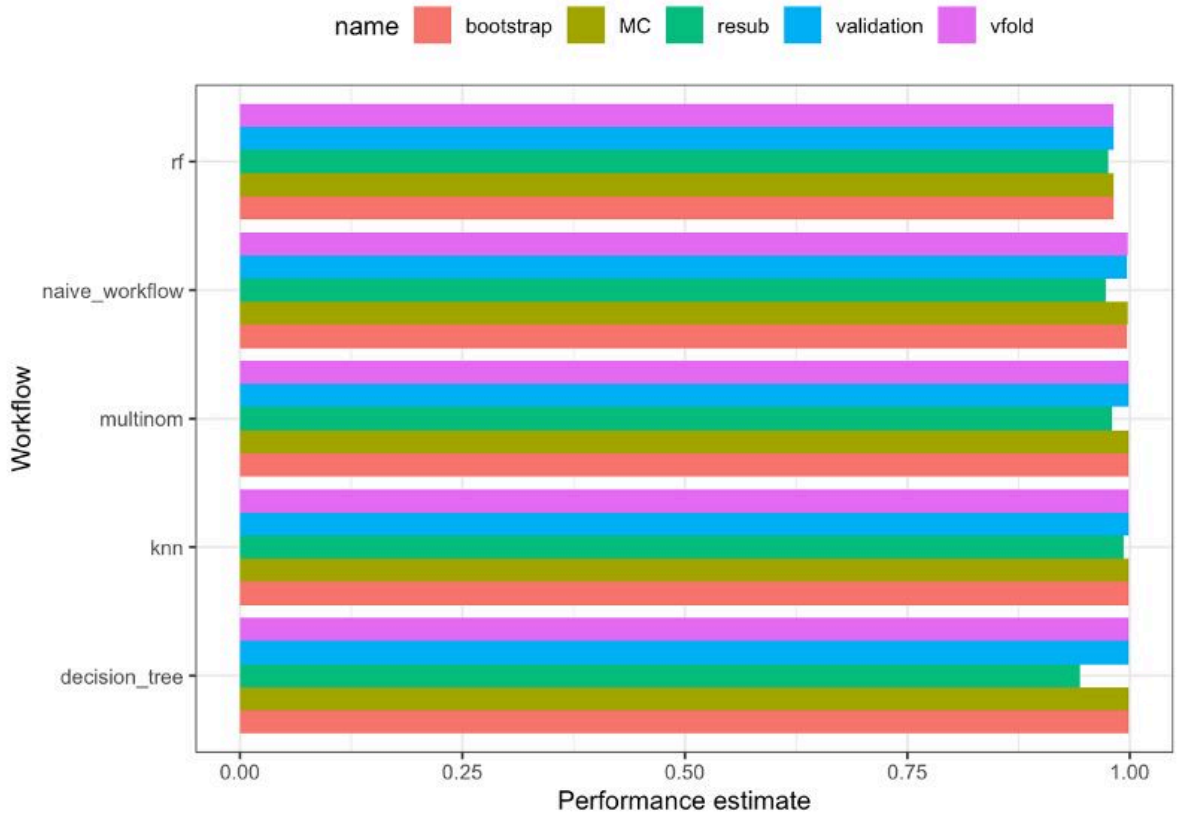
## 7. Model Selection

Model selection was rigorously performed using several resampling and validation techniques to ensure robustness on all the provided variables:

- **Validation Split**
- **K-Fold Cross-Validation**
- **Monte Carlo Cross-Validation**
- **Bootstrapping**

Each technique provided insights into how the models might perform on unseen data, helping in assessing model stability and generalizability. Following are the plots to visualize the performance of the

models based on each validation technique.

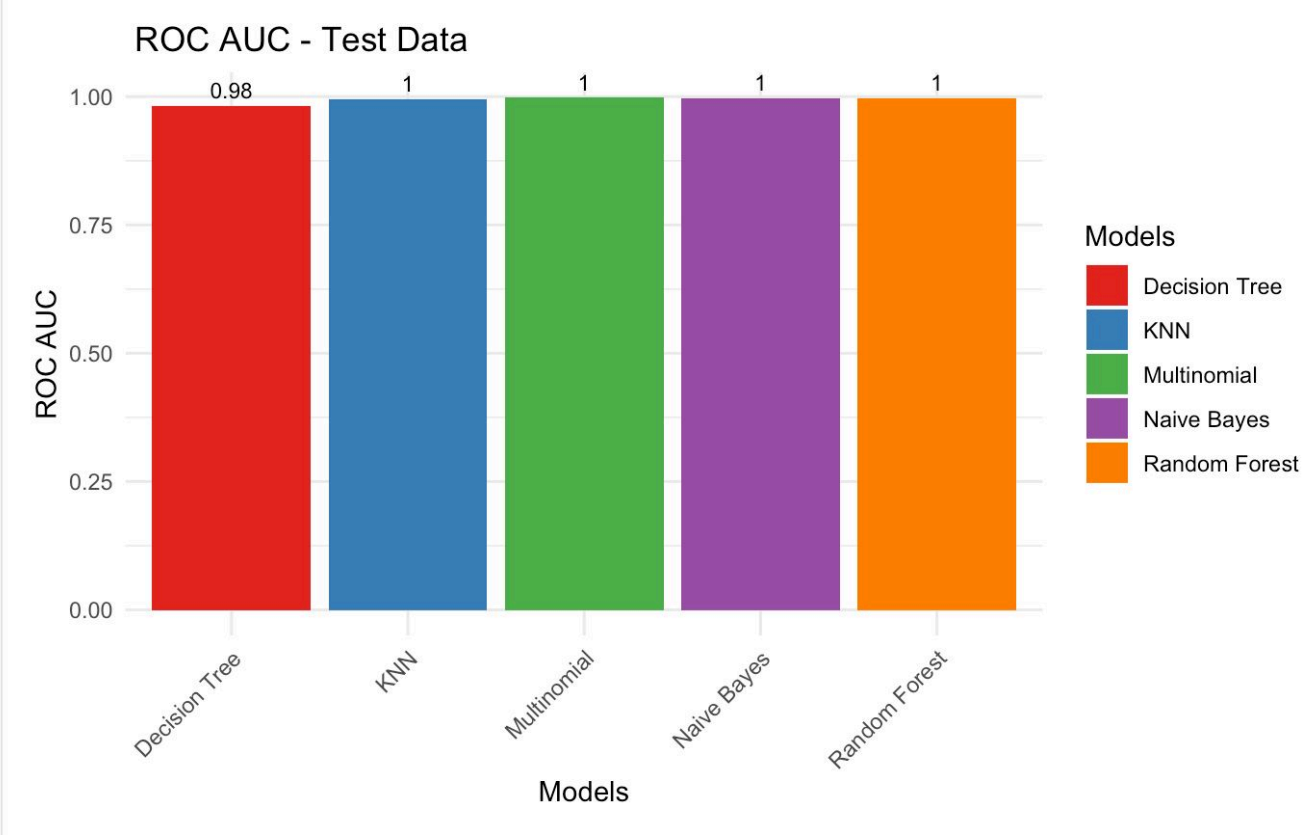



In assessing the performance of various machine learning models on the train data set using different validation methods, the random forest model consistently exhibits superior performance across all validation techniques, including bootstrap, Monte Carlo, resubstitution, validation, and k-fold cross-validation. This model's robustness and high performance metrics suggest it as the optimal choice for deployment, given its strong generalization capabilities and resistance to overfitting. The ability of the random forest to handle non-linear data effectively, along with providing insights into feature importance, further solidifies its suitability for addressing complex predictive tasks in the project.

## 8. Model Assessment and Interpretation

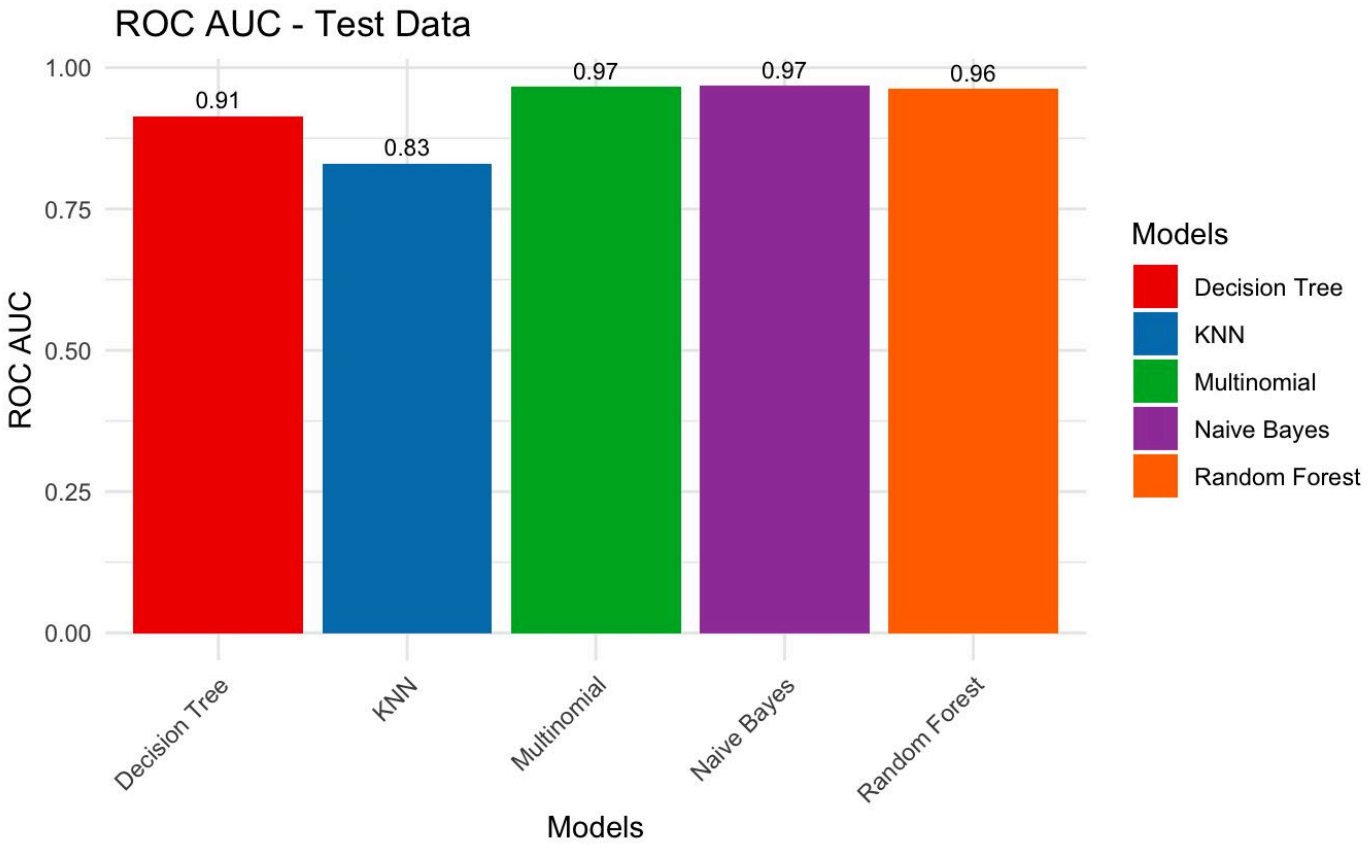
Performance of the models was assessed using metrics like ROC AUC and kappa scores. The analysis included:

- **ROC Plots**: Compared the true positive rate and false positive rate across models.




The bar chart illustrates the ROC AUC (Receiver Operating Characteristic Area Under the Curve) scores for various machine learning models tested on the same dataset. K-Nearest Neighbors (KNN), Multinomial Logistic Regression, Naive Bayes, and Random Forest models all achieved perfect scores of 1.00, indicating they perfectly distinguished between the classes. In contrast, the Decision Tree model scored slightly lower at 0.98, which is still very high but suggests it may not perform as well as the other models in some scenarios.

Model selection was rigorously performed using several re sampling and validation techniques to ensure robustness on selected the variables obtained by the correlation between the explanatory variables




The bar chart presents the ROC AUC scores for various machine learning models on test data, which measure each model's ability to distinguish between classes effectively. The Naive Bayes and Multinomial models both perform exceptionally well, with ROC AUC scores of 0.97, indicating very high predictive accuracy. The Random Forest model also performs strongly with a score of 0.96. In contrast, the KNN model scores lower at 0.83, suggesting it may be less effective at classification in this specific dataset compared to the others. The Decision Tree has the lowest performance with a score of 0.91, which, while respectable, indicates it might not capture the complexities in the data as well as the other models.

## 9. Uncertainty Quantification

Uncertainty in the dataset was quantified using the association rules generated by the Apriori algorithm. This method helped understand the relationships between different attributes and how they correlate with the complaint outcomes, providing a statistical basis for decision-making which was clearly visualized in the previous plots under the section **Apriori Algorithm Visualization.**

## 10. Results Communication

In this analysis, various machine learning models were evaluated to predict complaint types using a comprehensive dataset. The models considered included Naive Bayes Classifier, Multinomial Logistic Regression, K-Nearest Neighbors (KNN), Random Forest, and Decision Trees. These models were selected for their effectiveness in handling categorical data and their proven track record in classification tasks.

Through rigorous validation using methods such as Validation Split, K-Fold Cross-Validation, Monte Carlo Cross-Validation, and Bootstrapping, the Random Forest model emerged as the most robust across all resampling techniques. It demonstrated superior performance, consistently achieving high ROC AUC scores that nearly approached perfection. This suggests its strong capability in generalization, making it the optimal choice for deployment in this project.

The Random Forest model's ability to handle non-linear relationships and provide valuable insights into feature importance further validates its selection. On the other hand, while models like Naive Bayes and Multinomial Logistic Regression also showed high performance, with ROC AUC scores close to 0.97, the KNN model lagged slightly at 0.83, and the Decision Tree model, although scoring 0.91, showed limitations in capturing data complexities compared to the ensemble methods.

The uncertainty in the data was further explored using the Apriori algorithm to uncover association rules, which provided deeper insights into the relationships between different attributes and complaint outcomes. This statistical analysis supports the decision-making process by highlighting significant correlations that could influence the model's predictions.

The Random Forest model, with its robust performance and stability across various validation frameworks, is recommended for further deployment in predicting complaint types. This model not only ensures accuracy but also offers interpretability, which is crucial for ongoing monitoring and improvement of the predictive system deployed.

## Performance Reporting

In our analysis, we evaluated the performance of various machine learning models using two different sets of data: one including all variables and the other using a selected subset of variables. Performance metrics, displayed through ROC AUC scores, indicate that while models such as Random Forest, KNN, Multinomial Logistic Regression, and Naive Bayes achieved perfect scores of 1.00 with the full variable set, indicating flawless classification accuracy, their performance slightly varied with selected variables. Specifically, the Random Forest and Naive Bayes models showed a minor decrease in performance, and the KNN model notably dropped to 0.83, highlighting its sensitivity to variable selection. The consistent high performance of the Random Forest model across both datasets underscores its robustness and makes it a preferred choice for deployment in scenarios requiring high accuracy and reliability in classification tasks.

## Key Observations

1. **Model Performance:** The models generally perform well, with high accuracy and kappa scores. The Random Forest and Multinomial Logistic Regression models show particularly strong performance across different metrics and resampling methods.

2. **Validation Techniques:** Using multiple resampling methods helps in assessing the stability and reliability of model predictions across different subsets of data.

3. **ROC AUC Scores:** The ROC AUC scores are consistently high, suggesting that the models do a good job in distinguishing between different classes.

This detailed and structured approach in using `tidymodels` to train, evaluate, and validate multiple models on a dataset provides a robust framework for predicting outcomes based on insurance complaints, potentially guiding decisions in an operational setting.

**Conclusion**

The project successfully applied various statistical modeling techniques to the insurance complaints dataset, providing insightful predictions and analyses. The models highlighted areas where insurance companies could potentially focus to improve customer satisfaction and operational efficiency.

**Future Directions**

- Further Data Enrichment: Incorporating additional data sources such as customer feedback and financial records could enhance model accuracy and insights.
- Advanced Modeling Techniques: Exploring ensemble methods or deep learning could provide improvements in predictive performance.
- Real-Time Analytics Implementation: Developing a real-time complaint monitoring system to dynamically allocate resources and address complaints efficiently.

## Git Hub : https://github.com/lohitmarla-uconn/Insurance-Customer-Complaints-Classification