Machine Learning Project

Master in Data Science and Advanced Analytics

NOVA Information Management School

Universidade Nova de Lisboa

Machine Learning Project

To Grant or Not to Grant: Deciding on Compensation Benefits

Group 47

Diogo Duarte - 20240525

Inês Araújo - 20240532

Luís Semedo - 20240852

Rui Luz - 20211628

Fall Semester 2024-2025

Abstract

The New York Workers' Compensation Board (WCB) oversees the claims process, having reviewed over 5 million cases since 2000. This project aims to automate the classification of claims by predicting the *Claim Injury Type* using Machine Learning models, leveraging labeled data from 2020 to 2022 for training and validation. The project workflow included exploratory data analysis (EDA), data cleaning, handling missing values and outliers, encoding categorical variables, feature engineering, and scaling. Feature selection techniques were applied to optimize model performance. Various predictive models were developed and evaluated to determine the most accurate approach for classifying claims. By implementing this solution, the decision-making process in workers' compensation is streamlined, significantly reducing manual effort while enhancing efficiency, accuracy, and consistency in claim classification.

Keywords: Machine Learning, Predictive Modeling, Data Preprocessing, Feature Engineering, Model Optimization, Claim Injury Type, Multi-Classification.

GitHub Repository: https://github.com/diogomrduarte/Machine_Learning_Group47

TABLE OF CONTENTS

1. Introduction

Labor compensation claims represent a significant challenge for insurance companies, requiring complex and carefully considered decisions regarding claim approvals. Accurate and timely prediction of claim outcomes can reduce operational costs, enhance efficiency, and benefit both insurers and claimants.

As part of a Machine Learning course, we developed a predictive model-based solution to automate claim classification and streamline decision-making processes for compensation requests. This project focuses on predicting the *Claim Injury Type* associated with new claims, leveraging labeled data collected between 2020 and 2022 for model training and validation.

The project is structured into several key stages. Initially, a comprehensive exploratory analysis of the datasets (training and test) was conducted, including the identification and correction of inconsistencies. Subsequently, a detailed analysis of the most relevant variables was performed, exploring their relationships with the target variable and with each other.

During the data preparation phase, rigorous methodologies were applied, including handling missing values, efficiently encoding categorical variables, scaling, and implementing robust strategies to address class imbalance. Following this, various supervised learning algorithms were tested and compared, with *Random Forest*, *LightGBM*, and *CatBoost* emerging as the key models evaluated for their performance throughout the study. Additionally, this project incorporates an innovative analysis in the Open-Ended section, where initially excluded variables were modeled and integrated into the predictive process.

In summary, this project highlights the critical role of Machine Learning in solving real-world problems, presenting a predictive model capable of optimizing decision-making for labor compensation claims. Furthermore, it underscores the importance of advanced data preparation and preprocessing techniques to ensure optimal model performance.

2. Data Exploration and Preprocessing

2.1. Overview of the Data

To begin, we identified the primary characteristics of the dataset. The training dataset comprises 593,471 rows and 33 columns, while the test dataset contains 387,975 rows and 30 columns. Upon reviewing the data types in both datasets, we observed that most variables are either of type object or float. However, some inconsistencies in data types were identified. Furthermore, we identified the target variable, *Claim Injury Type*, which represents the Workers' Compensation Board (WCB) decision regarding the benefits attributed to the claim. This numeric variable reflects the severity of the injury and is critical for developing the predictive model.

2.2 Data Inconsistencies

During the exploratory data analysis, several inconsistencies were observed, particularly concerning date variables. These discrepancies needed correction to ensure the dataset's validity and reliability.

For the *Assembly Date*, which represents the date when the claim was initially registered, it was identified that this date should always occur after the *Accident Date*, as a claim cannot logically be registered before the accident occurs. However, an analysis of the data revealed that the training dataset contains 1,407 invalid dates, while the test dataset includes 222 such occurrences.

We are interested in identifying cases where the variables *C-2 Date* and *C-3 Date* are earlier than the *Accident Date*, which may indicate inconsistent data. For the *C-2 Date* variable, in the training set, we have 982 invalid values, while in the test set, there are 181 incorrect records. Regarding the *C-3 Date* variable, for the training set, we found 1289 invalid values and 226 inconsistencies in the test set.

The *First Hearing Date*, which represents the date of the initial hearing for the case, was another variable requiring scrutiny. Given that hearings cannot logically precede the accidents they address, this date should always occur after the *Accident Date*. However, the analysis revealed 74 invalid dates in the training dataset and 9 in the test dataset where this condition was violated.

The *Average Weekly Wage* variable, used to calculate workers' compensation benefits, showed no negative values, which aligns with expectations. The data revealed 335,450 such occurrences in the training dataset and 316,549 in the test dataset. The *Birth Year* variable also presented invalid values, with some instances recorded as zero.

The *Gender Features* variable contained four distinct values: M, F, U, and X. Here, "M" corresponds to Male, "F" to Female, and "U" to Unknown, representing individuals who chose not to specify their gender.

Finally, the *ZIP Code* variable, which in the United States consists of a sequence of five digits, was found to include values containing letters or missing entries. Specifically, 8.16% of *ZIP Codes* in the training dataset and 5.06% in the test dataset were invalid.

2.3. Analysis of Numerical Variables

The *Age at Injury* variable exhibits a bimodal pattern in both the training and test sets, with peaks between 30-40 years and 50-60 years *(Fig. 1)*. After 60 years of age, there is a sharp decline, likely due to retirement or a decrease in active individuals at this age. Outliers, such as ages of 0 or above 90 years, were identified and should be addressed, as they are inconsistent with the context of the problem. The missing values rate in the training set is 3.27%, while there are no missing values in the test set.

The *Birth Year* variable has 8.17% of missing values in the training set and 5.01% in the test set. The distribution is concentrated in a specific range, with some very low outliers, possibly due to data entry errors *(Fig. 2)*. The distributions in both the training and test sets are consistent.

The boxplot for the *Average Weekly Wage* variable *(Fig. 3)* reveals a skewed distribution with extreme outliers far from the median. These high values may result from anomalous or incorrect data. The similarity between the training and test set distributions indicates consistency, contributing to the model's integrity.

The *IME-4 Count* variable represents the number of IME-4 forms received per claim, which can indicate the complexity or dispute of the cases. It shows a skewed distribution with many outliers outside the interquartile range (IQR) *(Fig. 4)*. Additionally, this variable has a high proportion of missing values (77.62% in the training set and 90.9% in the test set), which can be interpreted as unsubmitted forms.

The heatmap presented in *Figure 5* displays the correlations between the variables in the training and test datasets. The heatmap analysis reveals that the correlations are all very close to zero for both datasets, indicating very weak relationships between the various numerical variables and the target variable, *Claim Injury Type*. Furthermore, the correlation patterns are practically identical across both datasets. It is also worth noting that, as expected, the variable *OICS Nature of Injury Description* shows no correlation with any of the other variables, as it was empty in both datasets.

2.4. Analysis of Categorical Variables

The histogram analysis for the *Accident Date* variable *(Fig. 6)* reveals a strong concentration of records between 2016 and 2023, with similar distributions observed in both the training and test datasets. The boxplot highlights the presence of outliers at the beginning of the period (1960–1980), which may be due to errors. The asymmetric distribution, with a predominance of recent data *(Fig. 7)*, has the potential to introduce temporal bias in predictive models. Therefore, it is crucial to evaluate the relevance of these outliers and consider creating derived variables, such as year or month, to mitigate potential negative effects. Additionally, the variable shows 3.89% missing values in the training dataset and 0.62% in the test dataset.

Regarding the *Alternative Dispute Resolution* variable, there is a notable imbalance, with the category “N” (No) dominating in both datasets, as illustrated in *Fig. 8.* The “Y” (Yes) and “U” (Undetermined/Unknown) categories exhibit extremely low frequencies, which may limit the predictive utility of the variable. In the training dataset, the “Y” and “U” categories are nearly insignificant, while in the test dataset, they appear to be completely absent.

For the *Attorney/Representatives* variable, a significant imbalance is again observed. The “N” (No representative) category accounts for the majority of observations, with approximately 400,000 records in the training dataset and 300,000 in the test dataset. In contrast, the “Y” (With attorney) category is substantially smaller, with around 180,000 records in the training dataset and 100,000 in the test dataset *(Fig.9)*.

The *C-2 Date* variable exhibits a significant temporal misalignment between the training and test datasets *(Fig.10, 11)*. While the training data are concentrated between 2020 and 2023, the test dataset is predominantly focused on 2024. Furthermore, there are outliers in much older dates, going as far back as 1985. The median for the training dataset is positioned after 2020, reflecting the concentration of more recent records, whereas the median for the test dataset is clearly in 2024, further emphasizing the temporal discrepancy.

In the *C-3 Date* variable, the majority of the data in the training set is concentrated between 2020 and 2023, with a few outliers in earlier years *(Fig.12, 13)*. However, the high percentage of

missing values (69%) limits its reliability and suggests a possible bias toward recent years. In the test dataset, most values are concentrated in 2024, with rare outliers and an even higher percentage of missing values (79%).

The *Carrier Type* variable is distributed primarily among categories 1A, 2A, 3A, and 4A, with no missing values. On the other hand, the *COVID-19 Indicator* variable shows that, in both the training and test datasets, the “N” category is predominant, and there are no missing values.

For the *District Name* variable, both the training and test datasets contain eight unique values, with the New York district showing the highest concentration of records. A similar pattern is observed in the *Medical Fee Region* variable, where Region IV stands out in both datasets. This concentration suggests that Region IV may be associated with the state of New York, where a higher number of workplace incidents potentially occurred.

The *First Hearing Date* variable has a high percentage of missing values, which complicates the generation of coherent visualizations. However, these missing values may be interpreted as the absence of hearings during the process.

Regarding the *Gender* variable, the distribution indicates that men “M” are more likely to experience accidents compared to women, “F”. In the test dataset, this difference is less pronounced. Additionally, the presence of records with the “U” (Unknown) category may represent individuals who chose not to disclose their gender *(Fig.14)*.

Finally, the *WCIO Cause of Injury Code*, *WCIO Cause of Injury Description*, *WCIO Nature of Injury Code*, *WCIO Nature of Injury Description*, *WCIO Part of Body Code*, *WCIO Part of Body Description*, *Industry Code*, and *Industry Code Description* variables exhibit a high number of unique values. Nonetheless, their distributions remain consistent, oscillating among a few categories while maintaining stable patterns across both datasets.

The analyzed heatmaps *(Figs. 15, 16, and 17)* show that claims are primarily concentrated in "2. NON-COMP" (non-compensable) and "4. TEMPORARY" across all studied dimensions. The "1A. PRIVATE" insurance type and Region IV are the most impacted, with the highest number of cases, particularly in "NON-COMP". In terms of gender, men (M) record significantly more occurrences than women (F) across all injury categories. Regions such as I, II, III, and UK, as well as less representative insurers, show much lower volumes. Severe injuries, such as "7. PTD" (permanent total) and "8. DEATH", are rare but more frequent among men and in Region IV. Thus, it can be concluded that the concentration of claims is related to specific economic, demographic, and occupational factors.

2.5. Preprocessing

To begin this section about preprocessing, we started by dropping rows with missing values in the target variable. Additionally, we dropped the column *OIICS Nature of Injury Description* because it contains only missing values. We confirmed that there are no duplicated *Claim Identifier* entries after dropping rows with missing values in the target variable. We then set this column as the index because it works as an Identifier.

We separated categorical and numerical features. However, columns like *Industry Code*, *WCIO Cause of Injury Code*, *WCIO Nature of Injury Code,* and *WCIO Part of Body Code* despite being numerical, represent categories. Therefore, we moved them to the categorical features.

After organizing the data, we isolated the target variable columns and later used the *train_test_split()* method to divide the training data into training and validation sets, with the last one containing only 20% of the training data, using the Hold-Out technique.

2.5.1. Missing Values

Next, we handled missing values. We defined a function to display the presence of missing values across the three dataframes side by side, which was used throughout the data preprocessing phase.

We applied a customized strategy for addressing missing values in numerical features, considering the unique characteristics of each variable. For the *IME-4 Count* variable, missing values were imputed with 0, based on the feature's meaning, 0 indicates that no forms were received. For the *Average Weekly Wage* variable, missing values were imputed with the mean calculated from the training set *X_train* dataframe to avoid data leakage.

For handling missing values in categorical features, we took the following approach. For the *First Hearing Date* feature, we created a new variable to indicate whether a hearing session took place or not and then dropped the original *First Hearing Date* column. This decision was based on the assumption that missing values in this column imply that no hearing session occurred.

For the remaining date columns, we defined a function to calculate the difference in days between the *Accident Date* feature and each of these columns individually. For rows with missing values, where calculations were not possible, missing values were replaced with the median of the respective column but, again, just from the *X_train* dataset to avoid data leakage. For the previously identified codes, missing values were replaced with 0, and features related to the description of these codes were dropped to avoid redundancy, as they repeated already existing information.

To conclude this section and before proceeding to the encoding phase, we performed manual feature selection. We considered that geospatial features should not influence the predictive process in this challenge (To Grant or Not to Grant). Therefore, we dropped features like *Zip Code*, *County of Injury*, and *District Name*. Additionally, we dropped features such as *Birth Year* since we already have a column containing the person's age at the time of the accident. We also dropped *Carrier Name* as the carrier should not influence the predictive process of the model.

2.5.2. Encoding

In this section, we applied different encoding methods tailored to the characteristics of each variable. For features with 2 or 3 distinct values, we converted them to binary format by replacing the least frequent value with the most frequent one and assigning 0 and 1 to the remaining two values.

For features that, even after reducing the number of distinct values, still contained more than two unique categories, we applied *One-Hot Encoding.* While this method increased the number of

features by creating one new feature for each distinct value, it prevented unintended ordinal relationships from being passed to the model. This helped to avoid bias and improved the model's ability to accurately capture relationships between categories.

For the target variable, we used the *OrdinalEncoder()* since the classes in the target variable exhibit a defined order.

2.5.3. Scaling

Knowing that the goal of scaling is to ensure that all features are on the same scale, we implemented three different scalers (*MinMax Scaler*, *Standard Scaler*, and *Robust Scaler*). To use the *Naïve Bayes* model, we chose the *MinMax Scaler*, as the model does not handle negative values, and this scaler limits values between 0 and 1. *Min Max* was used in *Naïve Bayes* and in *Logistic Regression* to check how it could change the scores. For the rest of the models, we used the data not scaled.

3. Multiclass Classification

3.1. Feature Selection

Although manual feature selection had already been performed to remove features deemed irrelevant, in this section, we applied well-known techniques like *Filter Methods* and *Wrapper Methods.*

3.1.1. Filter Methods

In this subsection, we analyzed correlations between features using a correlation matrix. We identified some strong correlations, such as between *C-2 Date Day* and *Assembly Date Days*.

3.1.2. Wrapper Methods

Here, we used the model that yielded the best results below, *XGBClassifier*, to determine the top 15, 17, 20, and 23 features using Recursive Feature Elimination (RFE). *XGBClassifier* computes feature importance metrics such as gain, cover, and frequency during training. These metrics help identify the features that contribute the most to the model's performance, making the *XGBClassifier* suitable for RFE. This approach allowed us to observe the evolution of results as features were added.

4. Model Assessment

Intending to develop a reliable classification model, we implemented various machine learning algorithms. This process aimed to evaluate and compare the performance of diverse candidate models, focusing on their ability to predict the outcome of the target variable *Claim Injury Type*.

We implemented different key metrics, such as accuracy, precision, recall, and *F1-scores* (macro and weighted averages) to provide a comprehensive evaluation of model performance, focusing on their ability to handle class imbalances effectively. After testing multiple models, including *Logistic Regression*, *RandomForestClassifier*, *CatBoost*, *LightGBM,* and *Naïve Bayes*, the *XGBClassifier*

emerged as the best-performing model, achieving the highest scores according to the evaluation metrics performed. In the table below *(Table 1)*, we may observe the comparisons of the models scores.

Model Name	Train Score	Validation Score	F1-Score Train	F1-Score Val
Logistic Regression	0.7157	0.7149	0.2352	0.2344
Random Forest	0.9995	0.7921	0.9989	0.3870
XGBClassifier	0.8177	0.7951	0.7019	0.4496
CatBoost	0.8043	0.7956	0.5603	0.4461
LightGBM	0.7889	0.7827	0.4273	0.3957
Naïve Bayes	0.6316	0.6309	0.2037	0.2030
Logistic Regression*	0.64423	0.6440	0.2589	0.2614

Table 1 - Comparison of Model Scores

As we can observe, the results of the assessment reveal that the *XGBClassifier* delivered the best overall results across both training and validation datasets. *XGBClassifier* achieved the highest *F1-score* (closely followed by *CatBoost*) on the validation dataset while maintaining a reasonable training score and a small difference between training and validation scores, indicating it balances precision and recall effectively, with minimal overfitting.

Besides the *RandomForestClassifier* presenting a high train score, the discrepant and inferior validation score indicates overfitting, this means that the model performs greatly on training data but fails to predict unseen validation data. In opposition, the *Naïve Bayes* model, in which both training and validation scores are low and similar, indicates that exists underfitting. The *Logistic Regression*, which is a simple model, revealed decent scores for validation and training but low *F1-scores*, indicating limited performance in handling class imbalance.

The implementation of the models *Naïve Bayes* and *Logistic Regression* Variant (seen as Logistic Regression* in the table) have a slightly different component since we applied the *MinMaxScaler* to scale the data and then applied it to the models. We can easily observe that the scores obtained were quite low. The *Linear Regression* performed worse with the data scaled, which was not expected by us initially. As we can observe in the tables below *(Tables 2 and 3)*, the macro and weighted average metrics (precision, recall, and F1-score) also confirmed the *XGBClassifier* robustness compared to the other models.

Model Name	Macro AVG		
	Precision	Recall	F1-Score
Logistic Regression	0.34	0.25	0.23
Random Forest	0.53	0.37	0.39
XGBClassifier	0.67	0.42	0.45
Catboost	0.66	0.42	0.45
LightGBM	0.45	0.39	0.40
Naïve Bayes	0.22	0.21	0.2
Logistic Regression*	0.34	0.25	0.23

Table 2 - Macro Average Metrics Comparison

Model Name	Weighted AVG		
	Precision	**Recall**	**F1-Score**
Logistic Regression	0.64	0.71	0.65
Random Forest	0.76	0.79	0.75
XGBClassifier	0.76	0.80	0.76
Catboost	0.77	0.80	0.75
LightGBM	0.75	0.78	0.74
Naïve Bayes	0.53	0.63	0.57
Logistic Regression*	0.64	0.71	0.65

Table 3 -Weighted Average Metrics Comparison

While the macro average presents the average of the metrics equally across all classes, without considering class frequencies, the weighted average, averages the metrics weighted by the number of samples in each class. Thus, we can conclude that the lower scores for all models in the Macro AVG highlight struggles with classes with smaller samples. Also, *XGBClassifier* and *Catboost* dominate the scores in both tables, reinforcing they are robust models for our data.

The *XGBClassifier* model's strong performance in both macro and weighted averages, alongside its competitive validation and training scores, supported our selection as the best model.

Since the XGBClassifier is our best model, we ran the model for different numbers of features, as shown in *Figure 18*, and observed that increasing the number of features improved the model's performance.

5. Open-Ended

5.1. Objectives

In the Open-Ended section, we aimed to further improve the accuracy of predictions for *Claim Injury Type*. For this, we decided to make use of the variables we previously left out (*WCB Decision* and *Agreement Reached*). With this finality in mind, we set the objective to build a prediction model to estimate these values, given that the features under consideration aren't provided in the initial test dataset.

5.2. Implementation

5.2.1. Understanding the variables

To better understand the features in question, we should look at the distribution of their values. Across the initial train dataset, we find that the *Agreement Reached* column contains two distinct values, 0 and 1, for 95% and 5% of the rows, respectively. For its binary values, we take that this column indicates whether there is an agreement without the involvement of the *WCB* (0 if "No"

and 1 if “Yes”). On the other hand, the values for *WCD Decision* are not as diverse as expected from the metadata, having “Not Work Related” as the only observed value. Given this narrow feature, we decided to only work with *Agreement Reached* as we find that it would make no sense to target a variable with only one observed value to make predictions.

5.2.2. Modeling and Predicting *Agreement Reached*

Once we set *Agreement Reached* as a feature to be taken into consideration by the final model, adding the column to the train dataset (given the nature of this note that the values in question are already encoded), we now need to find a way to predict possible values for this column in the test dataset, which isn’t given by default.

With this in mind, we started by fitting new models, with previously used classifiers and algorithms, to the values of *Agreement Reached* on the same train and validation split performed before. Given the low rate of rows where there was an agreement without the involvement of the *WCB* (5% of rows), we also compared these models to others using weighted classes, resulting in the following scores:

Model Name	Accuracy	Precision	Recall	F1-Score
XGBClassifier	0.9555	0.9431	0.9555	0.9420
XGBClassifier Weighted	0.8556	0.9556	0.8556	0.8923
RandomForestClassifier	0.9561	0.9500	0.9561	0.9388
LogisticRegression	0.9498	0.9240	0.9498	0.9324
LogisticRegression Weighted	0.8075	0.9532	0.8075	0.8606

Table 4 - Comparison of Model Performance with Weighted Classes

Comparing the score metrics of each model, we take that, similar to what happened during the main objective, the model conceived with the *Extreme Gradient Boosting Classifier* is overall the best fit for this feature.

5.2.3. Adding the new feature to the test dataset

To be able to use *Agreement Reached* as a feature to predict the *Claim Injury Type*, in a test dataset that does not contain any information about the feature in question, we used the newly developed model with the *XGBClassifier* to make predictions on the test dataset about whether there is an agreement without the involvement of the *WCB*. To compare how this variable affects the efficiency of our main model, we assigned this prediction to a new dataset so that we have two different test sets.

To finally achieve our Open-Ended goal, we built a final model, using the same *XGBClassifier* used in the main model, fitted to our train and validation sets with *Agreement Reached* as a feature in each of them. After training this model, we used it to make predictions on *Claim Injury Type* in the new test set (also using the predicted *Agreement Reached* column as a feature). The table below

shows the differences between using *Agreement Reached* as a feature, comparing the scores of the two models:

Agreement Reached	Accuracy	Precision	Recall	F1 Score
Left out	0.795133	0.764530	0.795133	0.755869
Used as a feature	0.552689	0.520901	0.552689	0.429407

Table 5 - Comparison of Model Performance with and without Agreement Reached

5.3. Outcome and Discussion

Going against what we first expected from this variable, we take that adding *Agreement Reached* predictions to the best model yet reduced the performance of the model, by the lower score metrics. After considering some possibilities, we concluded that this decrease could be caused by the use of a prediction feature (which will increase possible errors and deviations) and the low variety of observed values in the provided train data. If we are to achieve the best possible model in terms of score metrics, we should keep using the main features set before and leave out the *Agreement Reached* and *WCB Decision* columns.

Conclusion

Throughout the project, we began with an exploratory data analysis to better understand the features and how they could be treated during the preprocessing phase. We used the Hold-Out Method to evaluate model performance by dividing the training dataset into two subsets: a training set and a validation set, resulting in three datasets: training set, validation set, and test set.

After this separation, we applied various preprocessing strategies to handle missing values, minimizing the risk of data leakage. Additionally, we implemented techniques such as Encoding and Scaling to ensure that the data was ready for model training, improving its effectiveness and robustness during predictions.

This approach allowed for a more detailed analysis and effective preparation of the data, resulting in a more reliable and accurate model.

Moving forward, we explored various feature selection methods, including Filter Methods and Wrapper Methods, such as Recursive Feature Elimination with XGBClassifier. After testing multiple models, we concluded that the best-performing model was XGBClassifier, considering the multiclass classification problem at hand.

Annexes

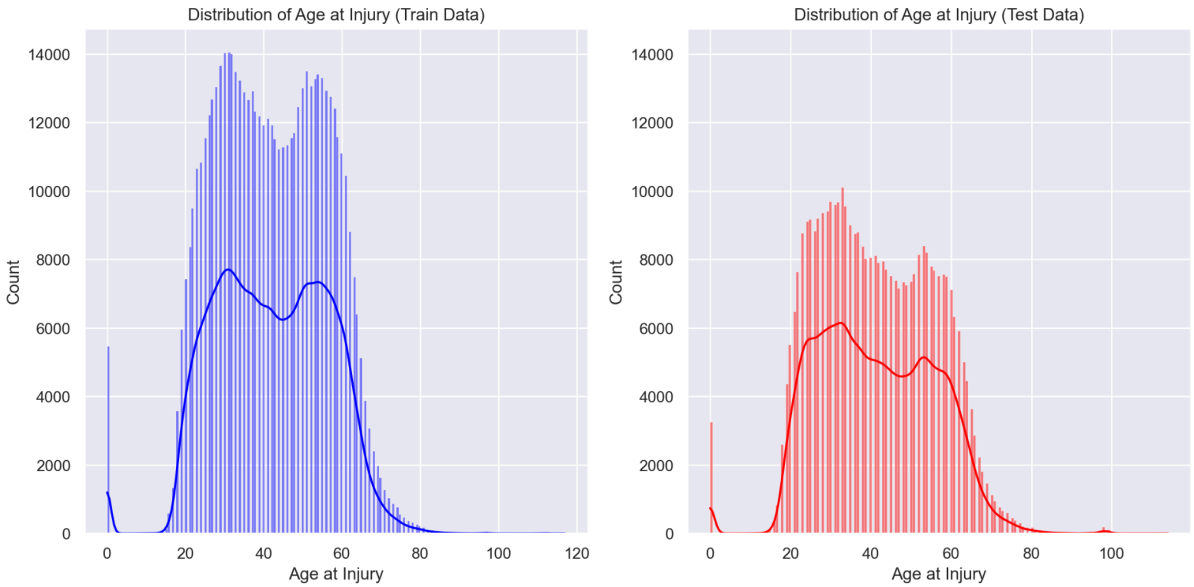

Figure 1 - Distribution of Age at Injury (Train and Test Data)

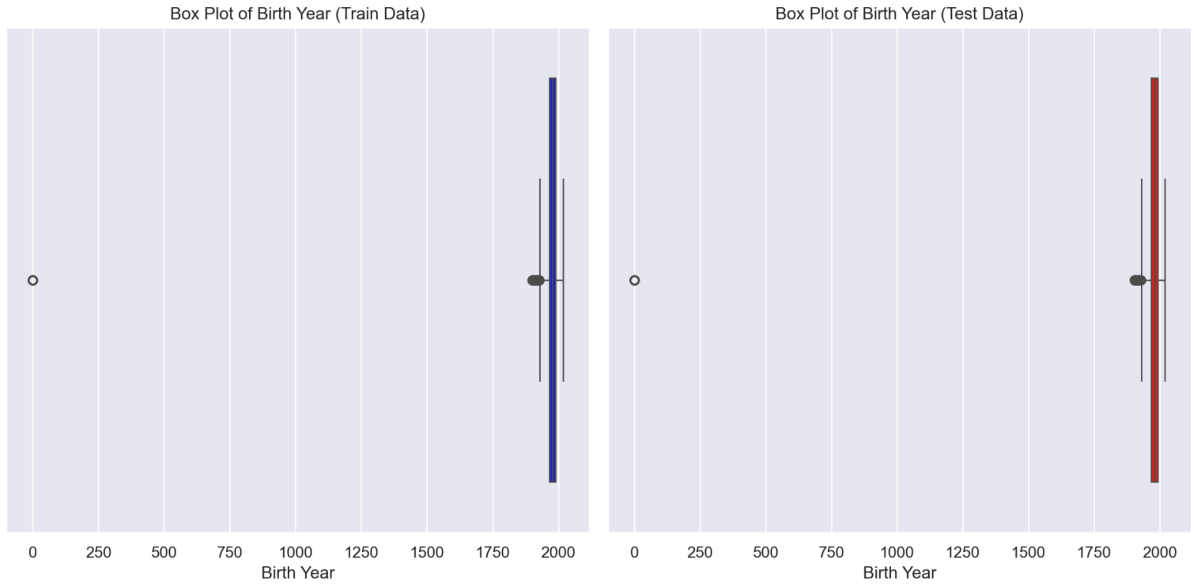

Figure 2 - Box Plot of Age at Injury (Train and Test Data)

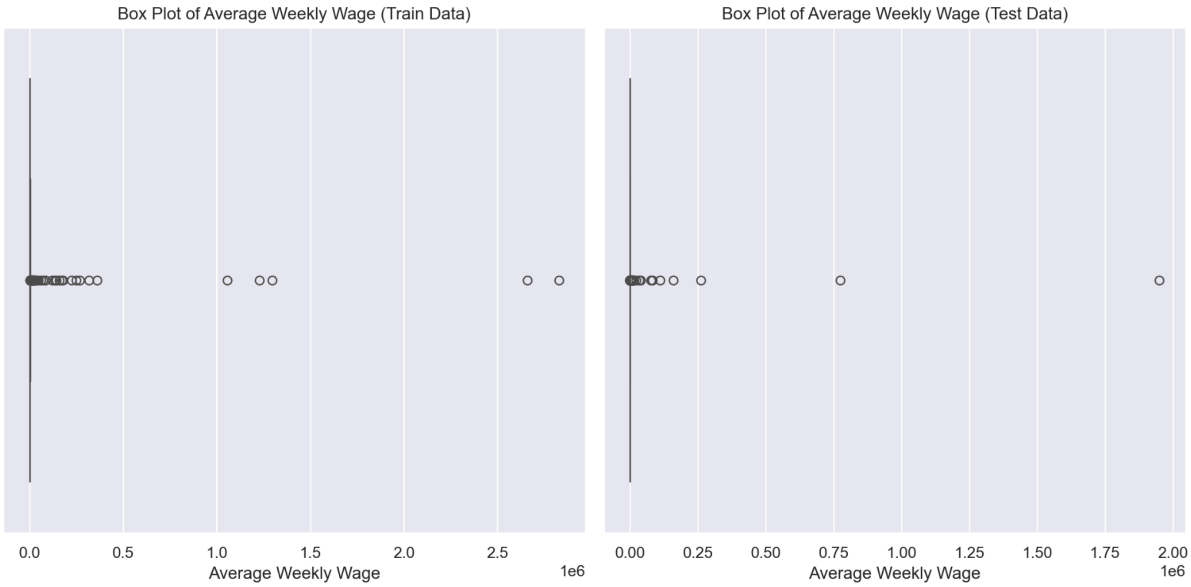

Figure 3 - Boxplot of Average Weekly Wage (Train and Test Data)

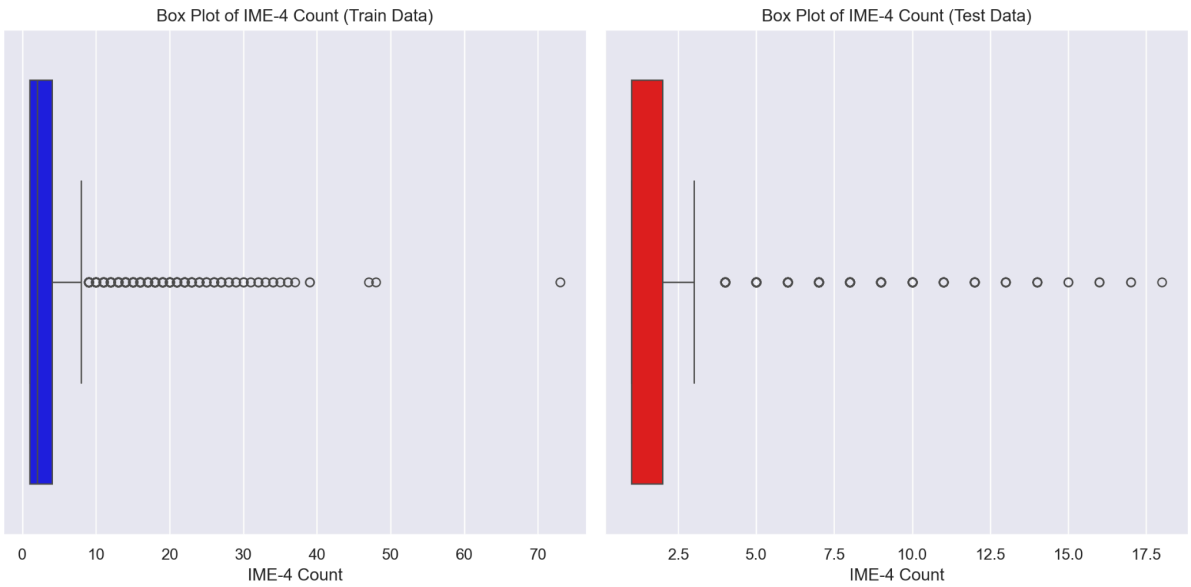

Figure 4 - Boxplot of IME-4 Count (Train and Test Data)

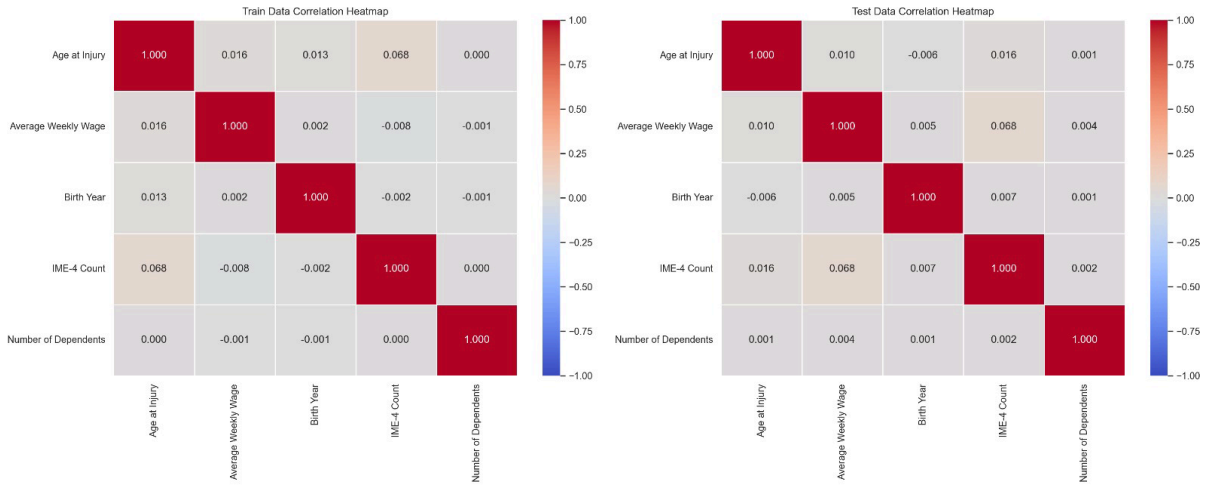

Figure 5 - Correlation Matrix for Numerical Features (Train and Test Data)

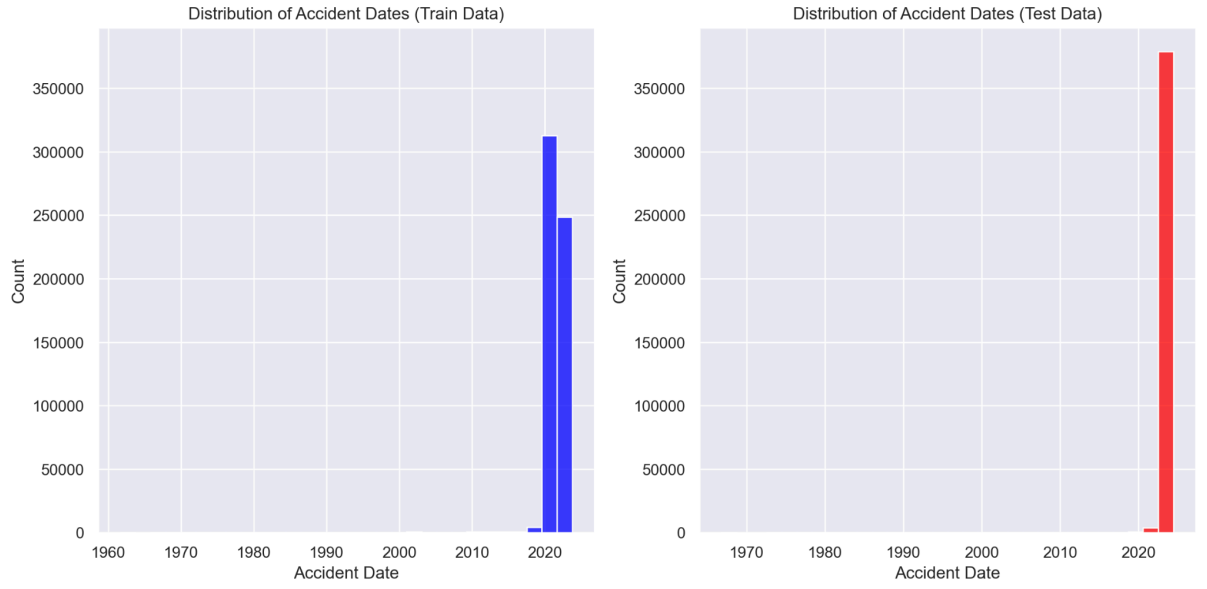

Figure 6 - Distribution of Accident Dates (Train and Test Data)

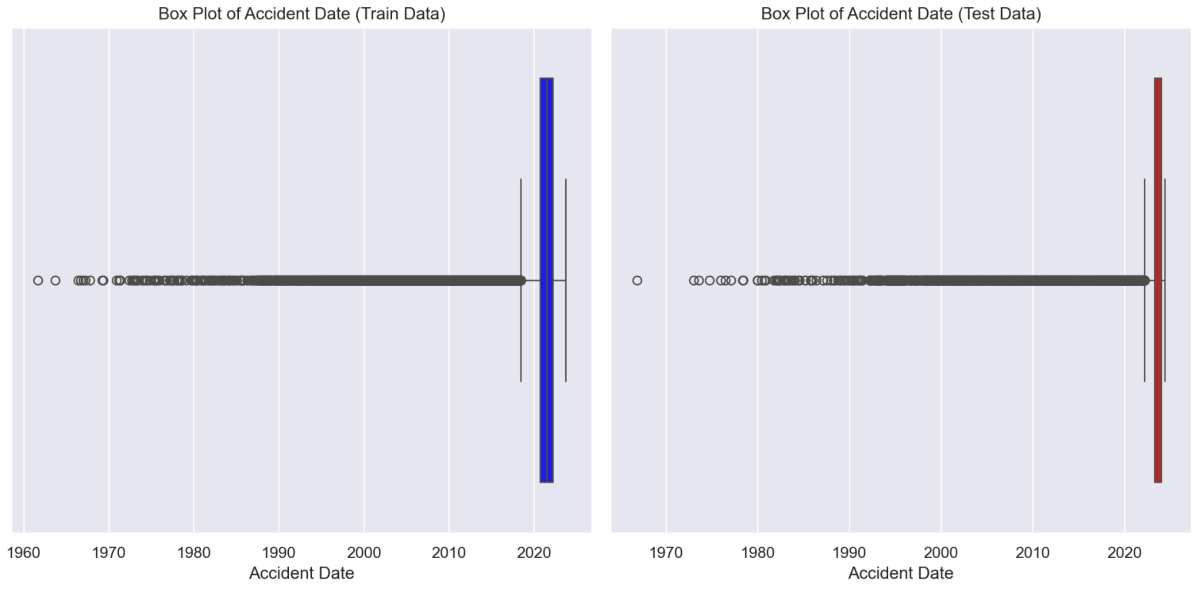

Figure 7 - Box Plot of Accident Dates (Train and Test Data)

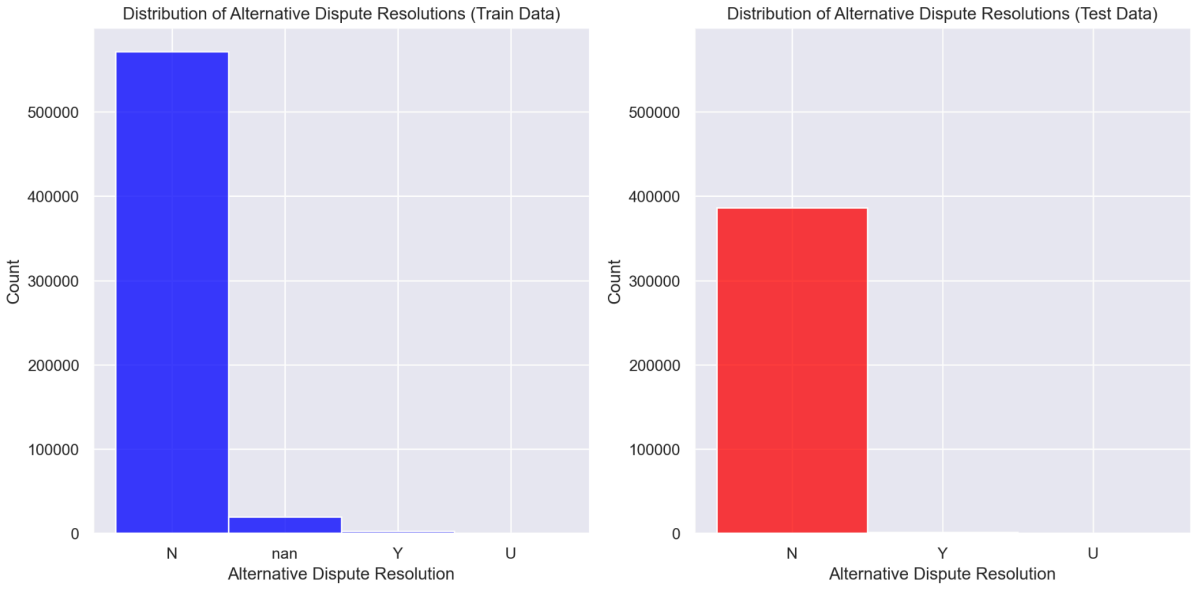

Figure 8 - Distribution of Alternative Dispute Resolution (Train and Test Data)

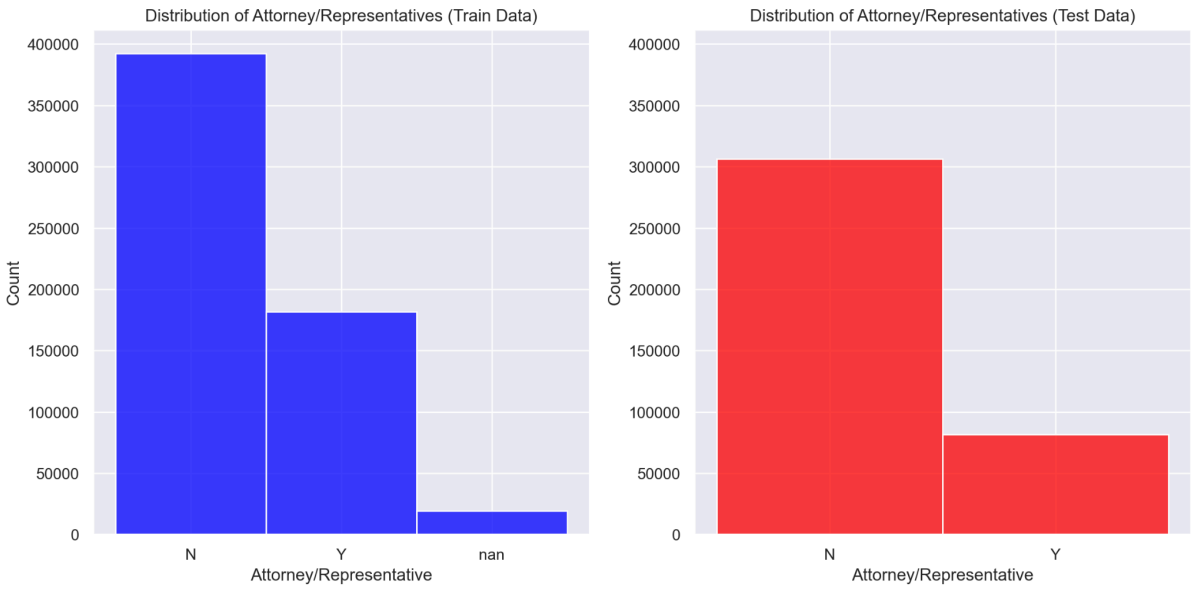

Figure 9 - Distribution of Attorney/Representatives (Train and Test Data)

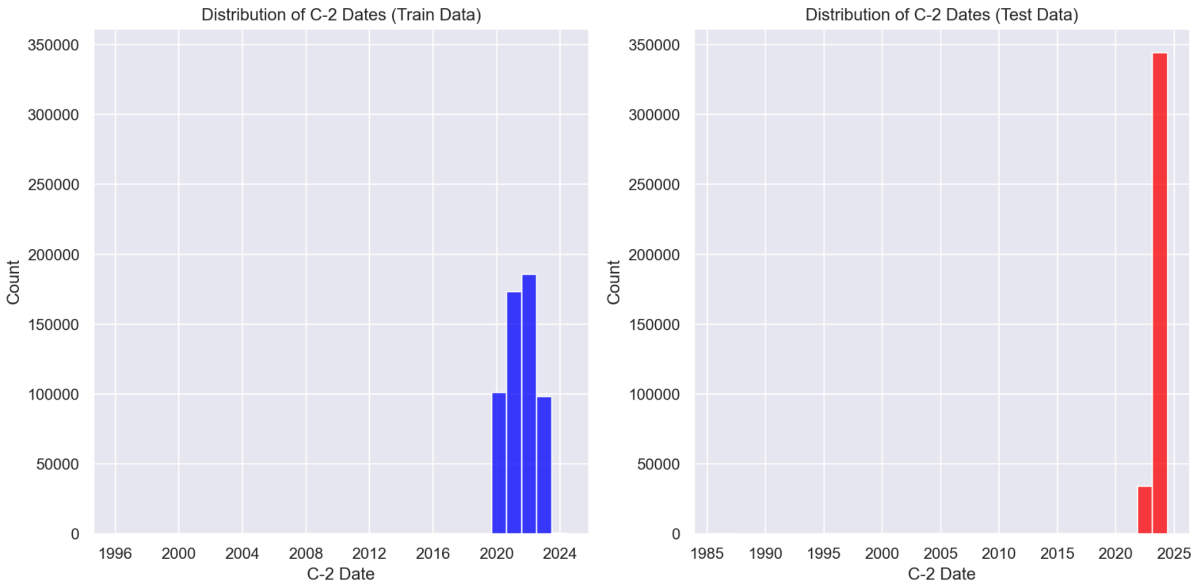

Figure 10 - Distribution of C-2 Dates (Train and Test Data)

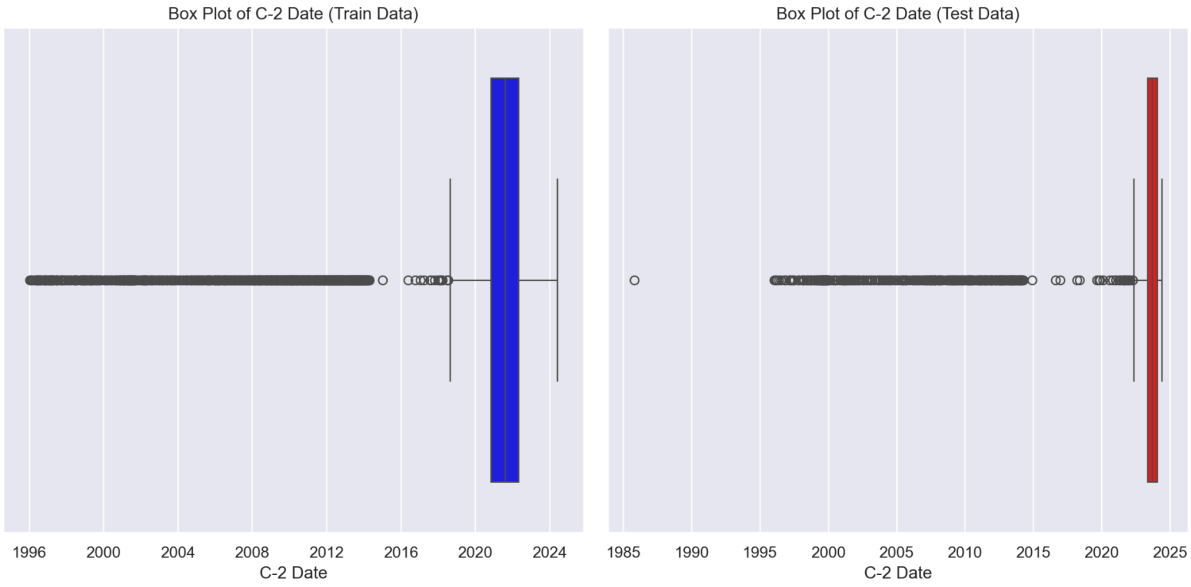

Figure 11 - Box Plot of C-2 Dates (Train and Test Data)

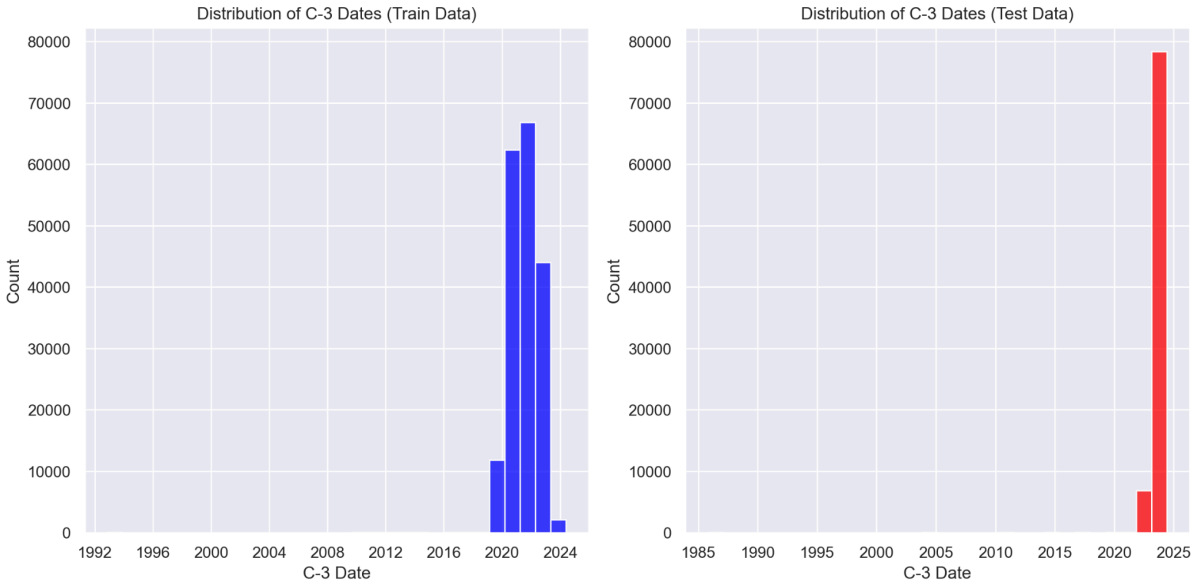

Figure 12 - Distribution of C-3 Dates (Train and Test Data)

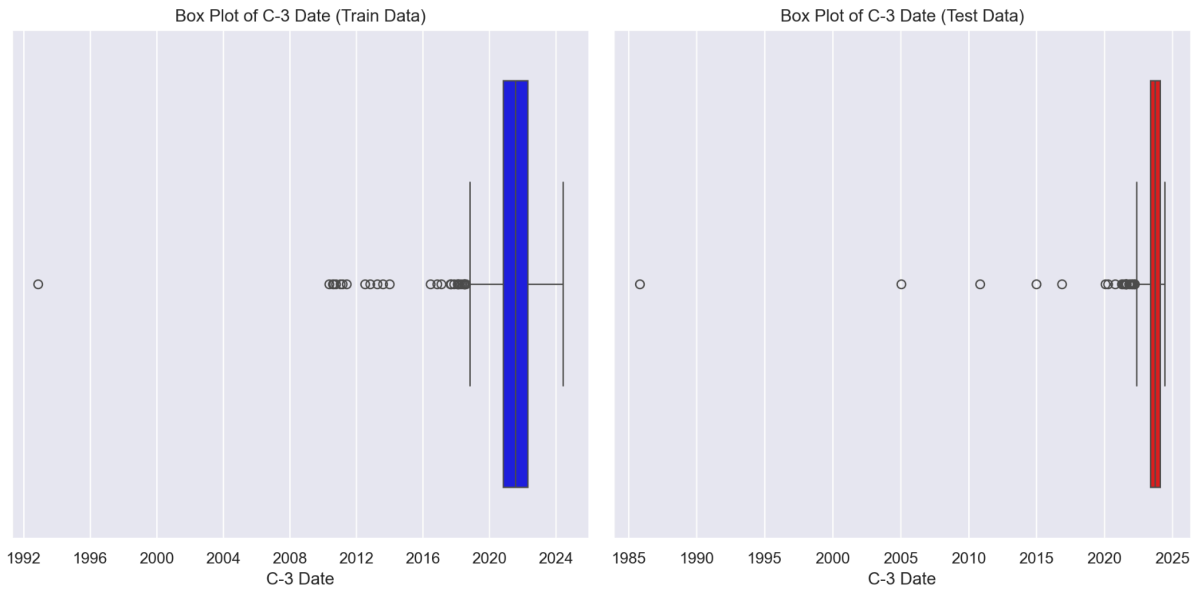

Figure 13 - Box Plot of C-3 Dates (Train and Test Data)

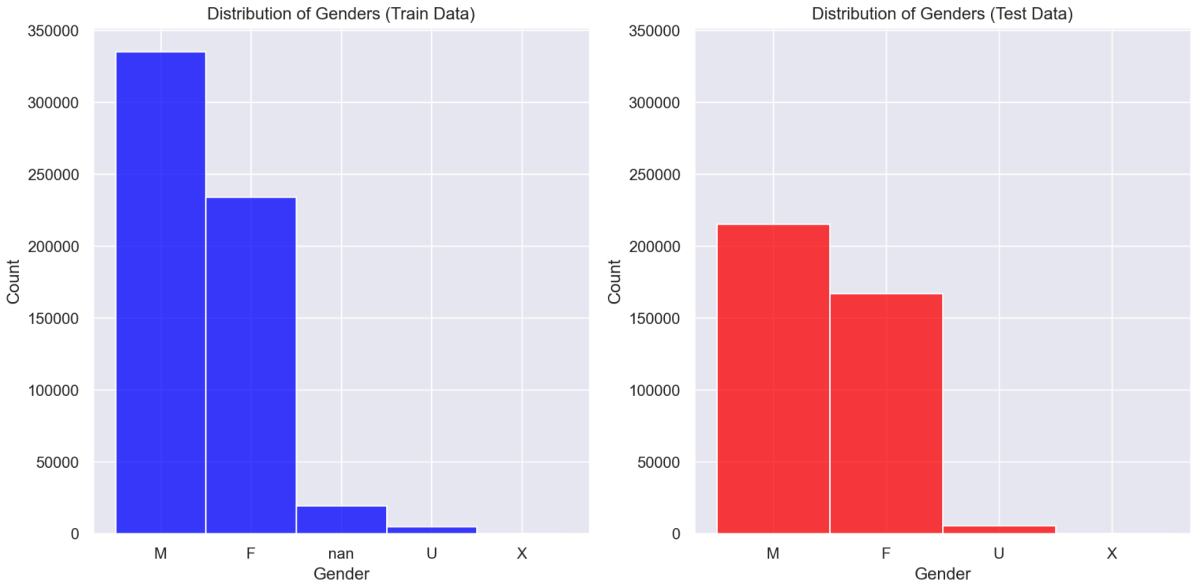

Figure 14 -Distribution of Genders (Train and Test Data)

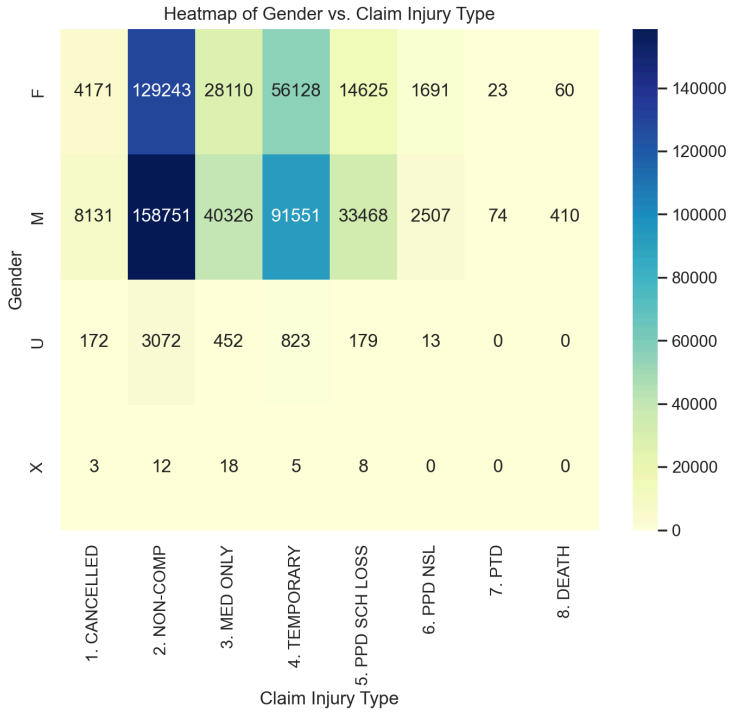

Figure 15 - Heatmap of Gender vs. Claim Injury Type

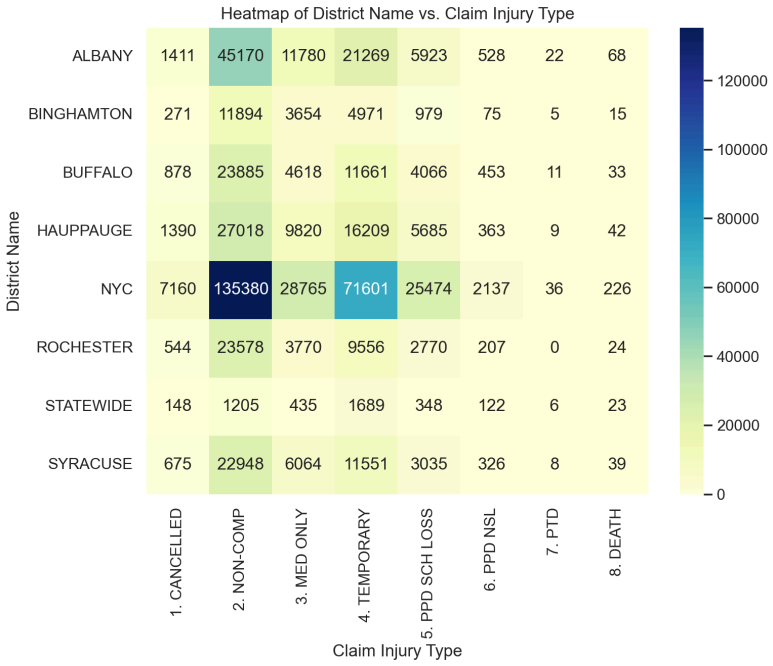

Figure 16 - Heatmap of District Name vs. Claim Injury Type

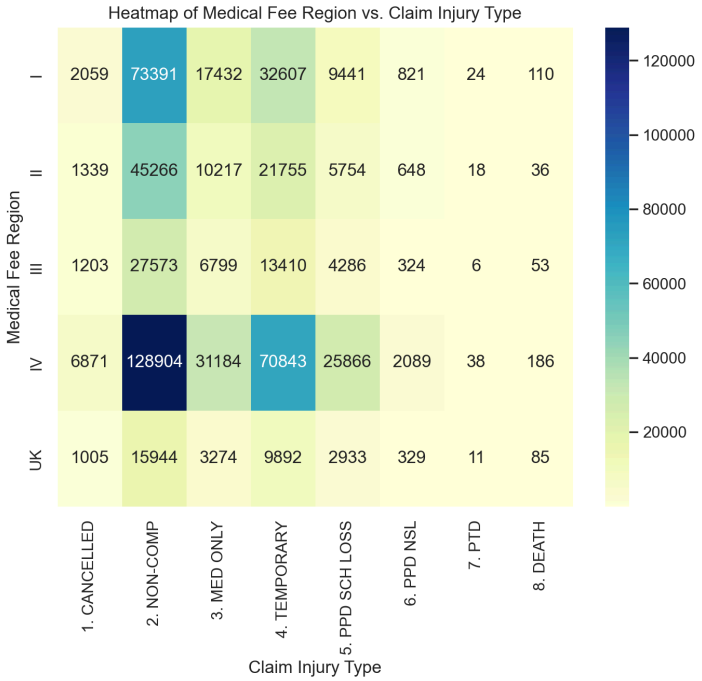

Figure 17 - Heatmap of Medical Fee Region vs. Claim Injury Type

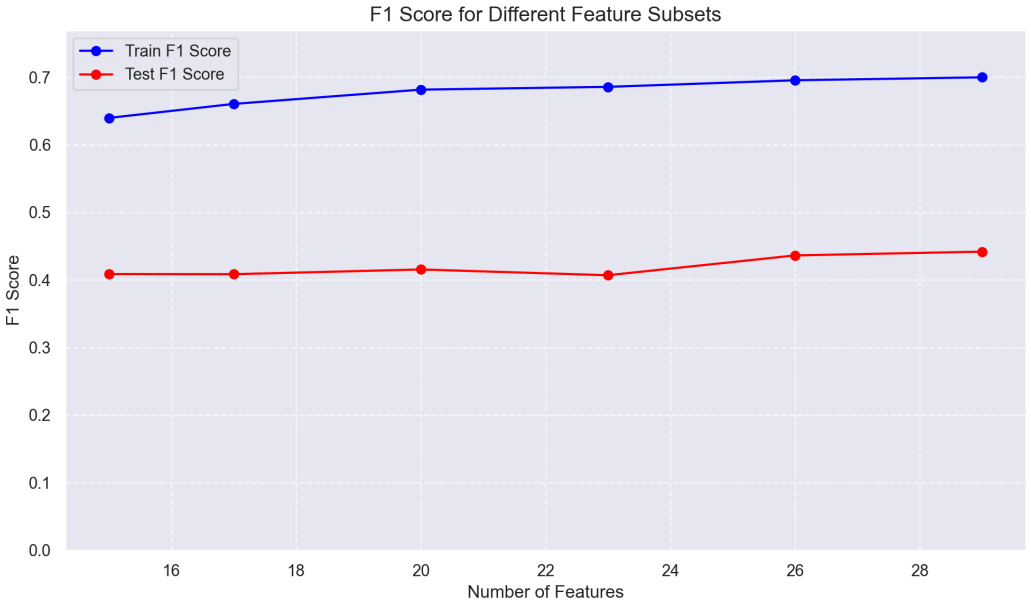

Figure 18 - F1-Score for Different Features Subsets