MACHINE LEARNING PROJECT

Master in Data Science and Advanced Analytics

NOVA Information Management School

Universidade Nova de Lisboa

TO GRANT OR NOT TO GRANT

DECIDING ON COMPENSATION BENEFITS

AI generated image (Mistral, 2024)

Group 33

André Silvestre, 20240502

João Henriques, 20240499

Simone Genovese, 20241459

Steven Carlson, 20240554

Vinícius Pinto, 20211682

Zofia Wojcik, 20240654

Fall/Spring Semester 2024-2025

INDEX [1]

[1] **GitHub Project Repository:** github.com/Silvestre17/ML_24.25_Project_Group33

ABSTRACT

From 2021 to 2023, the total claims assembled, manually reviewed and adjudicated by the New York Work Compensation Board (NWCB) have been on an upwards trend (2023 Annual Report). To mitigate this time-consuming process, our project aims to deploy ML algorithms to predict the severity of workers' compensation claims.

From the NWCB's public database, our training data consisted of claims assembled between 2020 and 2022 and analysed and adjudicated by the NWCB. The critical features for determining the severity of compensation claims were analysed using the following methods. For numerical features, VIF, Spearman correlation, RFE with various base estimators and normalization approaches, Lasso, and Ridge were applied. For categorical features, Cramer's V and Chi-Square tests were used instead of VIF and Spearman. Based on the results from each method, a voting approach was adopted for the final feature selection. For categorical features, the Cramer's V and Chi-Square were used instead of the VIF and Spearman. Based on the results of each method, a 2/3's voting approach was followed to make the final selection. 27 important features were selected.

An initial dataset of 593,471 records was analysed using nine models: *Logistic Regression* (LR), *Naive Bayes* (NB), *K-Nearest Neighbours* (KNN), *Neural Networks* (NN), *Decision Tree* (DT), *Random Forest* (RF), *Catboost*, *ExtraTress* and *Stacking* combining RF and LR. To compare model performance, six performance metrics were used: *Accuracy*, *Precision*, *Recall*, *F1 Score (Macro)* and *AUROC Score*. Moreover, due to an unbalanced target feature in the training data, *K-means SMOTE* was selected as the resampling method.

For model comparison, *CatBoost* and RF outperformed remaining models achieving a higher F1 Macro Score and less overfitting, underlining the potential of ensemble methods using DT as the base model. The usage of *K-Means SMOTE* resulted in worse performing models with less generalization power. Our project contributes to automating the decision-making of the NWCB, following a data-driven approach capable of tackling the current upward trend on number of claims analysed.

Keywords

Workers' Compensation Claims; Machine Learning; Classification Models; Ensemble Learning; Random Forest; CatBoost

1. INTRODUCTION

With over five million injury claims analysed since the year 2000, the NWCB processes and adjudicates work claims in the New York State across different industry sectors. As the regulating authority, the NWCB is responsible for assembling and deciding on claims whenever it becomes aware of a workplace injury. Considering the NY State Workers' Compensation Board (2023), it was revealed that the total claims assembled increased from 161,742 to 169,961 maintaining the upward trend started in 2021. Considering this upward trend, this project aims to automate the decision-making of the NWCB through the development of a Multiclass Classification Model capable of predicting the severity of workers' compensation claims according to eight possible injury types.

Over the past years different academic papers have been dedicated to the prediction of injury severity and type of injury resulted from accidents in different industry sectors (**Table A1**). Based on the literature review different approaches were used for data preprocessing and feature selection: some authors opted for more simplistic approaches like the total removal of missing values (Khairuddin et al., 2022) or initial feature selection based on previous papers (Alkaissy et al., 2023) while others used more complex approaches like the usage of *Random Forest* (RF) for the imputation of missing values (Sarkar et al., 2020) and methods like the *Boruta Algorithm* (Sarkar et al., 2020) or the *Regularized Logistic Regression* with *Lasso* (Mathews, 2016) for the feature selection/importance process.

Most studies found the RF as the best predictor by outperforming other models with higher accuracy and F1-score (Khairuddin et al., 2022). Additionally, most studies revealed nature of injury, accident nature and affected body part as important features. Studies where the target feature showed a great imbalance benefited from the application of resampling methods, improving the classification performance achieved. In Sarkar et al (2020), KMSMOTE was the resampling method selected helping achieve better predictions results.

Based on these findings, what's expected is the RF to be one of the candidate models in this project and features like nature of injury and affected body part to be relevant for the predictive power of the algorithms. Regarding the class imbalance present in the training data, one should expect that the application of resampling methods will result in better performance.

The remainder of the paper is organized using the CRISP-DM methodology (**Annex A**) following the structure defined in **Appendix B**: Data Exploration Process and Initial Preprocessing applied in **Section 2**. In **Section 3**, the Feature Engineering and Feature and Model selection processes. **Section 4**, details the development of a Web Application that will be shared with the NWCB providing a live analytics interface that returns predictions for the severity of a new claim based on the inputs provided. Finally, the project's conclusions with the main findings, limitations encountered and scope for future work in this area will be provided in **Section 5**. (Provost & Fawcett, 2013)

2. DATA EXPLORATION AND PREPROCESSING

In this section, the *Data Understanding* stage of CRISP-DM framework was conducted. Upon importing both the train and test datasets, the dimensions of each were noted. The train dataset contained 593,471 rows and 33 columns. The test dataset had 387,975 rows and 30 columns. It was also identified that *Claim Identifier* is the unique identifier for each claim and *Claim Injury Type* is the target variable. Further analysis of the target variable revealed a significant class imbalance, as visualized in **Figure C1**, which may pose a challenge for the models. (New York State, 2024)

2.1. Initial Preprocessing

Table 2.2 summarizes the preprocessing steps performed.

Table 2.2 – Initial Preprocessing Decisions

Steps	Rows/Columns Affected	Justification
Set Index as *Claim Identifier*	Whole dataset	Since this is the identifier of each claim case that is being studied.
Drop from Train Dataset	*Agreement Reached*, *Claim Injury Type* & *WCB Decision*	These columns are present in train but not in test. They refer to decisions following *Claim Injury Type*
Check and Drop Missing Values by Columns	*OIICS Nature of Injury Description* (100%)	This column only has missing values in train dataset therefore it was dropped.
Check and Drop Missing Values (by Rows)	Rows with at least 29 columns with missing values: 19445 (3.28 % of the Train Dataset)	These rows just have the *Claim Identifier* and *Assembly Date* columns filled, so they will be dropped.
Checking Duplicates	Whole dataset	The two observations, marked as duplicates when ignoring the *Claim Identifier*, were kept as they represent initiated cases with *Claim Injury Type = 1. CANCELLED* and may hold valuable information for the model. Duplicate rows were not deleted since the test dataset includes all such observations requiring classification.
Checking Anomalies	Whole dataset	No cells with empty values.
Other Anomalies	*Accident Date*	There are 1701 (0.3%) cases where the *Accident Date* is greater than the other dates on the train dataset. They were retained because the same issue had 315 cases (0.08%) of the test dataset.

2.2. Data Exploration and Feature Creation

In this section, the datasets were further explored for deeper understanding of the data. Issues identified in the data were addresses and resolved. This was done based on the procedural order outlined in the CRISP-DM framework. The exploration included analysing feature distributions through visualizations such as boxplots and histograms. Patterns, unusual values and outliers were examined. Also, data type inconsistencies were resolved. **Table C1** summarizes the dropped features, the issues they presented, and the corresponding features created to resolve them.

Temporal Features

Starting with the date columns, *Accident Date*, *Assembly Date*, *C-2 Date*, *C-3 Date*, and *First Hearing Date* were first converted from object into datetime format. Also, for each column year, month, day, and weekday were extracted and stored as new columns. Thanks to that, these features could be used across different algorithms, including those that cannot handle date data. As shown in **Figures C2-C5**, there were no significant differences in the percentages of claims by month or day. Only on the 31 of each month the relative frequency was lower since not every month has 31 days. Frequency of claims peak in March possibly due to increased trend in seasonal jobs. Also, claims were less frequent on the weekends. Lastly, observe that the dates do not fall within the expected range (2020-2022), because the range specified in the project indications refers to the assembly date of the claims, and not to the date of the accident or the date of receipt of the Employer's Report of Work-Related Injury/Illness.

Outliers were identified in *Accident Date* (0.99% of the train set) and *C-2 Date* (0.21% of the train set). *C-3 Date* and *First Hearing Date* had a substantial number of missing values with 67.4% and 73.7% of missing values respectively. Handling these features involved deciding on whether the missing values

themselves carried meaningful information or if they needed to be imputed or completely removed. In an effort of avoiding the risk of losing the potential meaning of the missing values these features were transformed into binary variables. If the original value was null it was translated to 1 otherwise to 0. Similarly, binary features were created for *Accident Date*, *Assembly Date*, and *C-2 Date,* to see the correlation between them and the target variable for each class.

Age at Injury* and *Birth Year

First, both columns were converted from float64 into Int64 format to represent whole numbers. Missing values for *Birth* Year accounted for 5.1% of the train dataset. Moreover, as visible in **Figure C6**, there was a few cases where *Age at Injury* and *Birth Year* were 0. After filling in the missing values using interdependence of these two columns, results were stored as new variables – *Age at Injury Clean* and *Birth Year Clean*. **Figure C7** depicts how the distributions for these variables transformed thanks to the changes made. The distributions were relatively consistent between train and test, with most values concentrated around certain decades, suggesting demographic trends in the data. Still, both features had 0.05% of outliers in the test datasets. This included anomalies such as age of 117. The small percentage of observations (0.01% of the train dataset) had *Age at Injury* lower than the legal limit for workers in *New York State*. (NYS Department of Labor, n.d.) (NYS Department of Labor, 2023) The decision was made to keep these instances as valid points as these could represent kids working for family businesses. Finally, *Age at Injury* Groups columns was created, in which ages were categorized based on equal-width binning (Han et al., 2012). This was done to try capture any non-linear relationships between age and the target variable, providing a more interpretable representation of age in the model. As shown in **Figure C8**, there were differences in distribution of *Claim Injury Type* by *Age at Injury Group*. Specifically, *'8. DEATH'* was the most frequent in group 81-100 and claim of '*1. CANCELLED'* was significantly more frequent for the age group *Unknown*.

Gender

Gender had 4 unique values: F (Female), M (Male), X (Nonbinary), and U (Unknown gender). The reported gender distribution of injured workers shows that males represent the majority in both the training and test datasets, comprising approximately 58.4% and 55.5%, respectively, while females account for around 40.8% in the training dataset and 43.05% in the test dataset. The categories *U* and *X* each make up less than 1.5% in both datasets, as referenced in the **Figure C9**.

Alternative Dispute Resolution, Attorney/Representative* & *COVID-19 Indicator

Upon investigation of *Alternative Dispute Resolution,* it was revealed that this variable had 3 unique values: N (No), Y (Yes), and U (Unknown). As depicted in **Figure C10**, frequency of these values was highly imbalanced, with *N* appearing in over 99.5% cases in train and test datasets. *Attorney/Representative* and *COVID-19 Indicator* had two unique values of N (No) and Y (Yes). **Figure C11** visualizes the frequencies of both values for *Attorney/Representative* and for *COVID-19 Indicator*. Most of the cases were not represented by an attorney and a great majority of the cases ware not *COVID-19* related.

Average Weekly Wage* and *IME-4 Count

Upon inspecting *Average Weekly Wage*, it was determined that the feature how it was, was not suitable for our models because it contained a substantial number of missing values, 0s, and extreme outliers (**Figure C12**). It was concluded that it is very unlikely for these to be true values and instead it is highly probable that these zeros are a result of intentional misreporting to maximize the

compensation claims by the workers. To retain the information given by this feature, a new binary variable took on 1 if the *Average Weekly Wage* was reported as more than 0 and 0 otherwise. As shown in **Figure C13**, in most of the cases the wages were not reported. Similarly, *IME-4 Count* was transformed into a binary variable. It was due to the observation that the minimal value for that feature was 1 (**Figure C14**). It was concluded that the missing values could in fact be 0, which would represent no IME-4 forms received per claim. A new binary variable was created with value 1 if *IME-4 Count* was greater than 0, and 0 if the value was null. As depicted in **Figure C15**, most of the cases did not include any IME-4 forms.

Carrier Name* and *Carrier Type

Carrier Name had 2046 unique values consisting of text. Treating text data was out of scope, therefore the feature was dropped. As depicted in **Figure C16**, *Carrier Type* had 8 unique values, where one category accounted for around 50% of train and test datasets; and 4 smallest (by frequency) categories accounted for only 0.5% and 0.6% for train and test datasets respectively. To simplify the feature, three least common *Carrier Types* were bucketed together, as demonstrated by **Figure C17**. Still, the frequencies of the categories were not balanced. According to **Figure C18** *Carrier Type Bucket* distribution by *Claim Type in train dataset was* the same across 5 types and differed only for category '*0. Unknown*'.

Industry Code and *Industry Code Description*

As seen in both **Figures C19** and **C20**, *Industry Code* and *Industry Code Description,* are categorical columns with 24 and 20 distinctive values, respectively. The distribution of categories for both features are right skewed with long tails. This distribution indicates a few categories dominating the distribution. Both features had 1.7% missing values in the train dataset and 2.0% in the test dataset.

WCIO Codes and Descriptions (Cause of Injury, WCIO Nature of Injury, WCIO Part Of Body)

WCIO Cause of Injury Code and *WCIO Cause of Injury Description* had 77 and 74 unique values, respectively. This might not be ideal for the model due to complexity of the feature. A new column WCIO *Cause of Injury Bucket* was created with the values bucketed by injury type according to (Guaranty Support, Inc., 2019). Similarly, *WCIO Nature of Injury Bucket* and *WCIO Part of Body Bucket* were created. Bucketing was done based on (Guaranty Support, Inc., 2019) to preserve the potentially meaningful group-level distinctions, while reducing dimensionality. As depicted in **Figures C21-C26** the Original *WCIO Codes and Descriptions* had high cardinality and right-skewed distributions with long tails. Bucketing reduced dimensionality of these features. Refer to **Figures C27-C29**, which visualize frequencies of each bucket. In *WCIO Nature of Injury Bucket,* bucket '*1 – Specific*' dominated with around 90% frequency for both datasets.

Number of Dependents

Frequencies of each number for *Number of Dependents* are uniformly distributed across different injury types, with no clear trends observed, refer to **Figure C30**. Similarly, as depicted by **Figure C31**, proportion of *Number of Dependents* is the same for each *Claim Injury Type.*

Region information (*Zip Code, County of Injury, District Name, Medical Fee Region)*

Medical Fee Region had 5 unique values. Its frequencies are visualized by **Figure C32**. It must be noted that the categories in *Medical Fee Region* are nominal. '*Region IV*' is the most frequent one accounting for almost half of the cases in both datasets. Similarly in *District Name*, district *NYC*, accounted for

almost 50% of both datasets, as shown in **Figure C33**. The rest, 7 districts, ranged from frequency of around 15% to 0.6%. *County of Injury* had 63 distinctive values with only 15 of them accounting for more than 1% of the cases, as seen in **Figure C34**. **Figures C35** and **C36** map the cases according to *County of Injury*. It was noted that this feature is inconsistent with the metadata, since the counties of injuries include counties outside of New York State. However, distribution of cases analysed in the training and in the test are of the same regions, scattered across the country. Thus, all the cases (including the ones outside of New York State) were kept.

Zip Codes had 8286 distinct values. 5% of Zip Codes in the train data were not numeric. Another 5% of the data was missing for this variable in both datasets. In **Figure C35** Zip Codes are marked on the map. Coloured locations are, as expected, in the NY State. Due to this feature having many inconsistencies ranging from different number of digits in *Zip Code* to high dimensionality, it was dropped.

3. MULTICLASS CLASSIFICATION

3.1. Feature Engineering

Before starting the feature engineering process, appropriate features were dropped (**Table C1**) and the data was split into 75% training and 25% validation sets. This proportion split balances model training with enough data for robust performance evaluation. The validation set ensures accurate assessment of the developed models.

To maintain data integrity and avoid introducing potential biases, all features were analysed for outlier values. To determine the outliers, Interquartile Range method (IQR) was used. Values below or above 1.5 x IQR were considered outliers. They were detected in three features – *Accident Date Year*, *Age at Injury Clean,* and *C-2 Date Year.* After the analysis all outlier values were retained. Removing outliers' risks eliminating natural variability, potentially distorting the data's inherent distribution. Additionally, the process of outlier removal can be subjective, introducing bias by making arbitrary decisions on which data points to exclude. Although outliers were not removed directly, the creation of new features (such as *IME-4 Reported*) managed outliers by focusing on binary indicators rather than raw values. This approach simplified complex variables, reducing the impact of extreme values and enhancing the model's robustness against variability. Also, logarithmic transformation was applied to the variables which were identified as having a significant number of outliers. However, this strategy did not resolve the issue, therefore was not used after all.

When it comes to missing values, categorical and numerical features were handled using two different approaches. In *Industry Code Description*, the only categorical feature with missing values, the null values were replaced with a new category – '*Unknown*'. The numerical variables - *Accident Date Day, Accident Date Month, Accident Date Weekday, Accident Date Year, Age at Injury Clean, C-2 Date Day, C-2 Date Month, C-2 Date Weekday, C-2 Date Year* - were imputed using *KNN Imputer*.

Variables with an ordinal relationship, such as *Age at Injury Group*, were encoded using Ordinal Encoding to preserve the inherent order. Remaining categorical variables, including *Carrier Type*, *Gender*, and *District Name*, were encoded using *One-Hot Encoding* since they lacked an ordinal relationship. This approach was also applied to binary categorical variables like *COVID-19 Indicator* and *Attorney/Representative*.

3.2. Feature Selection

After *Feature Engineering*, variables were selected based on the strategy illustrated in **Figure 3.1**.

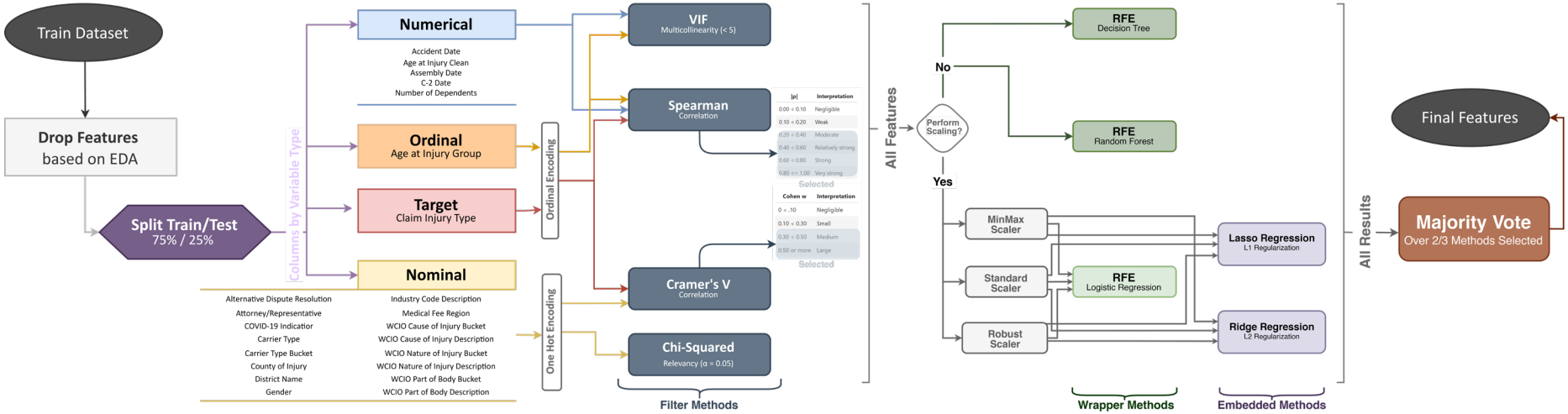


Figure 3.1 – Feature Selection process flowchart

Three scaling methods—*MinMax*, *Standard Scaler*, and *Robust Scaler* (**Annex D**) —were applied to ensure features were on similar scales, preventing dominance by larger-scale features. These scalers were chosen to handle outliers differently, allowing models that compute distances to perform effectively.[2]

The following section explores three feature selection methods: **Filter methods** focus on identifying intrinsic relationships among features or between features and the target variable. For *Cramer's V* (**Annex B**), features with a relevancy score above **0.3** were selected based on Cohen's rule-of-thumb (1988), which defines **0.3** as a medium effect size. A redundancy threshold of **0.8** was applied to eliminate highly correlated features. Similarly, *Spearman* correlation used a relevancy threshold of **0.2** and a redundancy threshold of **0.8**, justified by Rea and Parker's (2014) rule-of-thumb, which categorizes correlations of 0.2 as weak but meaningful. For multicollinearity, *Variance Inflation Factor* (VIF) (**Annex C**) was used, where features with a VIF score below **5** were selected.

[2] **Note:** The scalers have been applied also to *One-Hot Encoded* features. In the case of *MinMax* and *Robust* scalers the values are not changed. In the case of *Standard* scaler, the impact is irrelevant since what is important in *One-Hot-Encoding* is that the distance between positives and negatives is constant.

Wrapper methods base their selection on the validation F1 Macro Score. *Recursive Feature Elimination* (RFE) iteratively selects the most relevant subset of features in steps, evaluating model performance. The subset with the best validation score is retained, though this approach may introduce bias in future model evaluations. *Logistic Regression* RFE used *Ridge Regularization* by default, while *Decision Tree* and *Random Forest* RFE did not require feature scaling, completing their evaluation in a single iteration.

Embedded methods evaluate feature importance during model training without using validation data and features with a coefficient absolute value greater than **0** were selected. For *Ridge* and *Lasso Regularization*, feature importance was determined by the coefficients in Logistic Regression. L1 penalty (Lasso) and L2 penalty (Ridge) were applied to enhance the model's generalization ability.

The features have been selected by **majority of vote** of the methods, i.e. the features accepted at the end are the features that have been accepted by more than **2/3** (Nine *"Selected"* classifications) of the methods above. The final features selected for modelling are highlighted in **purple** (for numerical features) and **orange** (for categorical features) in the tables of **Appendix D.** [3]

3.3. Classification Model Strategy | Hold-Out Method

The *hold-out method* was used for model assessment. This was preferred over cross-validation due to the significant time that would have been required to iteratively perform the preprocessing steps for each fold of cross-validation. **Figure 3.1** summarizes the modelling process.

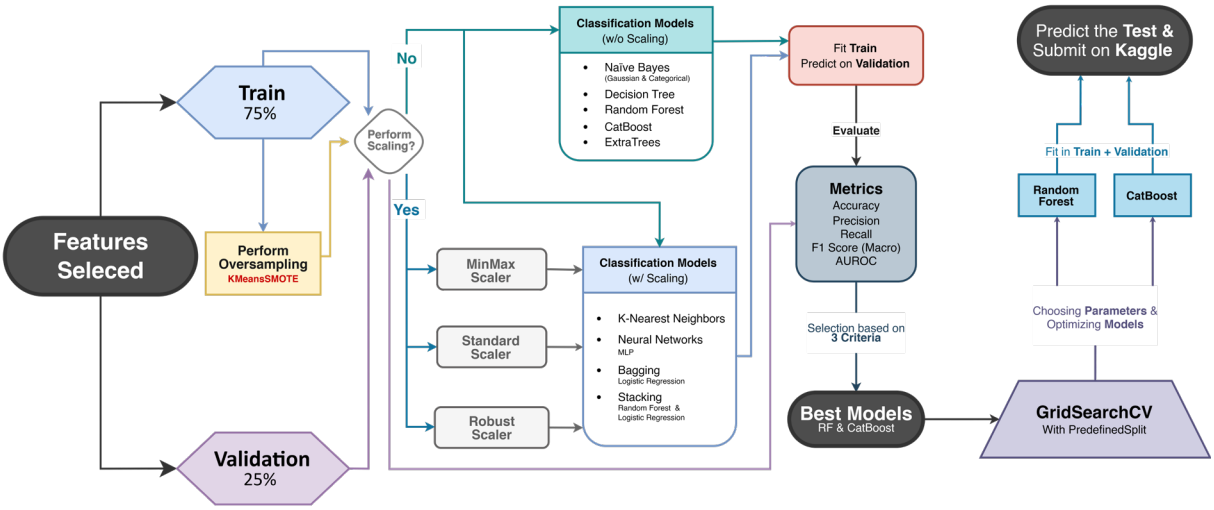


Figure 3.1 – Classification with Hold-out method Flowchart

The classification algorithms used were chosen from sources in the literature review (**Table A1**) and supplemented with additional algorithms chosen for their relevance and potential effectiveness in addressing the problem. To facilitate a robust modelling approach, the algorithms were applied to the original and/or scaled data, depending on each algorithm's sensitivity to feature scaling (**Table E1**). In keeping consistent with our feature selection approach, scaled data here refers to standard, normal, and robust scalers. Additionally, to address class imbalance, all algorithms were tested both with *KMeans-SMOTE* and without *KMeans-SMOTE* (**Annex F**).

[3] **Note:** Among the features, there are also the one-hot encoded variables. It may happen that some specific categories of variables are dropped, some others kept. That happens because the algorithm reckons as more relevant for the algorithm to know whether an observation belongs or not to a certain category rather than the exact category.

3.4. Models Evaluation

The following metrics were used to assess model performance: *Accuracy*, *Precision*, *Recall*, *F1-Score* (macro), and *AUROC Score*. Given the large number of total combinations run, single-score metrics were needed to efficiently compare model effectiveness. Multiple metrics were chosen to gain a robust and comprehensive measure of performance. Accuracy provides an overall measure of correctness, while Precision and Recall assess the model's ability to manage false positives and false negatives, respectively. The F1-Macro combines these into a single metric, which is particularly useful for the imbalanced classes in the dataset. Finally, AUROC score evaluates the model's ability to distinguish between classes by measuring its ranking performance across various thresholds, considering each class against others or between pairs of classes.

CatBoost with the original data was selected as the final model according to the selection process in **Table F1**. F1-macro score was chosen as the primary metric because of the class imbalance present in the dataset. After selecting for models with one of the top two highest F1 score values on validation data (**0.4** and **0.41**), *KMeans-Smote* models were removed because of overfitting difference greater than **0.1**. Next, models with the highest Accuracy and AUROC were retained. *CatBoost* scaled models were then removed due to a lack of improvement (both theoretical and observed) over the original data. Finally, both remaining models, *CatBoost* and *Random Forest*, were tested with the test data on *Kaggle* and *CatBoost* was selected as the highest performing model.

3.5. Fine-tuning Parameters (GridSearch)

To optimize the performance of *Random Forest,* hyperparameter tuning was conducted using *GridSearchCV* (**Table H1**). *GridSearch* - in contrast to random search, which only finds local optima - ensures the global optimum is reached by evaluating all possible combinations of the predefined hyperparameter values. Therefore, it guarantees parameter optimization based on the chosen hyperparameters. For the same reason as above, class imbalance, F1-macro was selected as the optimization parameter. A *PredefinedSplit* was employed for cross-validation to prevent data leakage.

F1-macro score on the validation set improved from **0.40** to **0.42** after optimization of *Random Forest*. For *Catboost*, applying *GridSearch* proved too computational expensive. Therefore, the parameters of the base model were used.

4. OPEN-ENDED SECTION

In this final phase of the CRISP-DM methodology, *Deployment*, the aim was to address the business problem – creation of a model that can automate the decision-making whenever a new claim is received – and ensure that the developed solutions are valuable for supporting strategic decision-making.

4.1. Web Application

To automate the NWCB's claim analysis, an end-to-end interactive web application was developed using the *Streamlit* (2024) library. This application allows users to explore the collected and cleaned data autonomously (**Figure I1**)

The web app has two main features:

1. **Prediction Interface**: Users can input claim-related data, and the application will generate predictions based on the best-performing model (*CatBoost*).
2. **Data Exploration Section**: This section allows users to interactively explore the dataset. Key insights, such as trends and relationships between variables, can be visualized. This functionality supports better understanding and analysis of claims data.

One of the challenges in creating this application was managing resource limitations during the implementation.[4] Due to the complexity of the model, the online application couldn't support and crashed. To solve this problem, the application's performance was optimized, and additional support was requested from the platform team. The improved resources were approved after demonstrating the importance of the application in an academic context.

The final version of the application is now available and serves as a practical tool for live claim predictions and data exploration, showcasing the project's value and applicability. [5]

4.2. LIME-based Interpretability Approach

Additionally, the *LIME* (marcotcr, 2019; Ribeiro et al., 2016) library was used to provide interpretability for the model's predictions. Individual predictions are explained (**Figure J1**) by identifying which features contribute most to the outcome. This approach aligns with the objective of making the model's results transparent and actionable. The analysis focuses on the most relevant features, offering clear insights into the decision-making process of the model.

The explanations generated by LIME are divided into three parts:

1. **Prediction Probability**: This section shows the probabilities assigned to each class, helping users understand the model's confidence in its predictions.
2. **Top 10 Features**: This highlights the ten most important variables influencing the prediction. For binary classification, features supporting class 1 are shown in **orange**, while those supporting class 0 are in **blue**.
3. **Actual Values**: The actual values of these top 10 features are displayed, making it easier to interpret the context behind the prediction.

This approach effectively bridges the gap between complex model outputs and user understanding, ensuring the results are both interpretable and aligned with the project's objectives.

[4] **GitHub Web Application Repository:** https://github.com/Silvestre17/ML_WebApp_Group33
[5] **Streamlit Application Link:** https://mlproject-wcb-group33.streamlit.app/

5. CONCLUSION

This project investigated the application of various ML models to predict the severity of the worker's compensation claims based on an initial dataset with around 600k injury claims. These were assembled between 2020 and 2022 by the NWCB covering different types of industry sectors. After an initial preprocessing and exploration stage, outliers and missing values were handled, and categorical features were encoded to obtain final pre-processed data later used as input for the ML algorithms.

A total of nine algorithms were used, namely LR, *NB*, *KNN*, *NN*, *DT*, *RF*, *CatBoost, Extra Trees*, and a *Stacked* algorithm combining LR and RF. The outputs were evaluated using five performance metrics: Accuracy, Precision, Recall, F1-macro and AUROC. To compare the results obtained, a model performance comparison strategy was used based on two criteria. Firstly, models with the highest scores on the validation set were prioritized, with F1-macro as the primary metric, followed by Accuracy and AUROC as secondary criteria. In case of ties, the model with the least overfitting was selected.

Based on the mentioned criteria, the two best models in predicting the type of injury claim were the RF and *Catboost* algorithms. This finding highlights the potential of ensemble learning as the ML technique of choice to achieve higher model performance. By combining a series of weak classifiers into a single strong classifier, it enhances the performance prediction (Khairuddin et al., 2022). Then, *GridSearch* was applied to determine best parameters for each algorithm. RF parameter optimization resulted in an average F1 macro score of **0.42**. For *Catboost*, applying *GridSearch* proved too computational expensive. Therefore, the parameters of the base model were used. Regarding feature importance, *Weekly Wage Reported*, *Age at Injury Clean* and *First Hearing Date Binary* were found to be the most critical in both models. Considering initial expectations based on the literature review, some materialized while other did not. As expected, RF was one of the top performing models. Nature of injury and affected body part features proved to be relevant, being in the top 15 most important features for both models. Regarding the usage of resampling algorithms, they were expected to improve the model performance, however their usage proved to generate more overfitting hence no models with *K-Means SMOTE* were selected.

Some limitations of this project should be highlighted. Firstly, *K-Means SMOTE* was the only resampling algorithm tested. Even though this decision was made based on the literature review, other methods like the normal *SMOTE* or *Borderline SMOTE* could have been used to attempt to improve model performance. Moreover, parameters used for *K-Means SMOTE* could have been further tuned. Secondly, this project focuses on injury claims specific to the New York State. Since each state has its own regulations, the project findings might be hard to generalize to other states. Thirdly, using the Hold out method instead of the K-Fold cross validation limited our ability to evaluate the algorithms generalization power and final model selection. Lastly, changes in the WCB's claim assembly process over recent years may have introduced inconsistencies in the dataset used.

Considering how the current project scope is limited to the New York State, it could be interesting to develop a more generalized model capable of being deployed in all states. Moreover, textual features initially dropped due to high number of unique values could be treated more in-depth using text mining algorithms to evaluate their predictive power. Additionally, exploring the usage of K-Fold cross validation could significantly improve our model selection. Finally, this study contributes to the claim severity literature by showcasing ML models' deployment to predict claim injury type and to automating the decision making of the NWCB.

BIBLIOGRAPHICAL REFERENCES

Aggarwal, C. C. (2017). *Outlier Analysis*. Springer International Publishing.

Alkaissy, M., Arashpour, M., Golafshani, E. M., Hosseini, M. R., Khanmohammadi, S., Bai, Y., & Feng, H. (2023). Enhancing construction safety: Machine learning-based classification of injury types. *Safety Science*, *162*, 106102. https://doi.org/10.1016/j.ssci.2023.106102

Brownlee, J. (2018, June 14). *A Gentle Introduction to the Chi-Squared Test for Machine Learning*. Machine Learning Mastery. https://machinelearningmastery.com/chi-squared-test-for-machine-learning/

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*(16), 321–357. https://doi.org/10.1613/jair.953

Chen, T., & Guestrin, C. (2016). XGBoost: a Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, 785–794. https://doi.org/10.1145/2939672.2939785

Cohen, J. (1988). Statistical power analysis for the behavioral sciences. *Journal of the American Statistical Association*, *73*(363), 223–225. https://doi.org/10.2307/2286629

Douzas, G., Bacao, F., & Last, F. (2018). Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Information Sciences*, *465*, 1–20. https://doi.org/10.1016/j.ins.2018.06.056

Glen, S. (2015, September 22). *Variance Inflation Factor*. Statistics How To. https://www.statisticshowto.com/variance-inflation-factor/

Guaranty Support, Inc. (2019). *WCIO Injury Code (Check WCIO website for current tables)*. https://www.guarantysupport.com/wp-content/uploads/2024/02/WCIO-Legacy.pdf

Haibo He, Yang Bai, Garcia, E. A., & Shutao Li. (2008, June 1). *ADASYN: Adaptive synthetic sampling approach for imbalanced learning*. IEEE Xplore. https://doi.org/10.1109/IJCNN.2008.4633969

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining : Concepts and Techniques*. Elsevier.

Hancock, J. T., & Khoshgoftaar, T. M. (2020). CatBoost for big data: an interdisciplinary review. *Journal of Big Data*, *7*(1). https://doi.org/10.1186/s40537-020-00369-8

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning : with applications in R*. Springer. https://static1.squarespace.com/static/5ff2adbe3fe4fe33db902812/t/6009dd9fa7bc363aa822d2c7/1611259312432/ISLR+Seventh+Printing.pdf

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.ISO 690

Kelleher, J. D., Brian Mac Namee, & Aoife D'arcy. (2015). *Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies* (pp. 460–480). The Mit Press.

Kelley, K., & Preacher, K. J. (2012). On effect size. *Psychological Methods*, *17*(2), 137–152. https://doi.org/10.1037/a0028086

Khairuddin, M., Lu, L., Hasikin, K., Razak, A., Wee Lai, K., Shakir, A., & Salwa Ibrahim, S. (2022). Occupational Injury Risk Mitigation: Machine Learning Approach and Feature Optimization for Smart Workplace Surveillance. *International Journal of Environmental Research and Public Health*, *19*(21). https://doi.org/10.3390/ijerph192113962

Mangiafico, S. S. (2016). *Summary and analysis of extension program evaluation in R* (1.15). RutgerCooperative Extension

marcotcr. (2019, July 5). *marcotcr/lime*. GitHub. https://github.com/marcotcr/lime

Martinez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernandez Orallo, J., Kull, M., Lachiche, N., Ramirez Quintana, M. J., & Flach, P. A. (2021). CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories. *IEEE Transactions on Knowledge and Data Engineering*, *33*(8).

Mathews, D. (2016, June 21). *Data Mining and Machine Learning Algorithms for Workers' Compensation Early Severity Prediction*. Mtsu.edu. https://jewlscholar.mtsu.edu/items/7f05bcc7-66e8-486e-94f5-94d13f9b509a

Menard, S. W. (2002). *Applied logistic regression analysis*. Sage Publications.

New York State. (2024, September 23). *Assembled Workers' Compensation Claims: Beginning 2000*. Ny.gov. https://data.ny.gov/Government-Finance/Assembled-Workers-Compensation-Claims-Beginning-20/jshw-gkgu/about_data

New York State Workers' Compensation Board. (2023). *2023 ANNUAL REPORT*. https://www.wcb.ny.gov/content/main/TheBoard/2023AnnualReport.pdf

NYS - Department of Labor. (n.d.). *Youth*. Department of Labor. https://dol.ny.gov/youth

NYS - Department of Labor. (2023, November). *Division of Labor Standards Worker Protection - Summary of New York State Child Labor Law, Permitted Working Hours for Minors Under 18 Years of Age*. Department of Labor. https://dol.ny.gov

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2019). CatBoost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, *31*. https://arxiv.org/abs/1706.09516

Provost, F., & Fawcett, T. (2013). *Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking*. O'Reilly.

Rawlings O, Pantula G and Dickey A (1998) *Applied Regression Analysis: A Research Tool (Second ed.).* NewYork: Springer, pp. 372–373. ISBN 0387227539. OCLC 54851769

Rea, L. M., & Parker, R. A. (2014). *Designing and conducting survey research : a comprehensive guide* (4th ed.). Jossey-Bass.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, February 16). *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. ArXiv.org. https://arxiv.org/abs/1602.04938

Sarkar, S., Pramanik, A., Maiti, J., & Reniers, G. (2020). Predicting and analyzing injury severity: A machine learning-based approach using class-imbalanced proactive and reactive data. *Safety*

Science, *125*. https://doi.org/10.1016/j.ssci.2020.104616

Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*(1). https://www.sciencedirect.com/science/article/pii/S1877050921002416

Scikit-Learn. (n.d.). *sklearn.preprocessing.RobustScaler — scikit-learn 0.24.2 documentation*. Scikit-Learn.org. Retrieved December 9, 2024, from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html

Snee R (1981) *Origins of the Variance In?ation Factor as Recalled by Cuthbert Daniel (Technical Report)*.12–65 Snee Associates. DOI: 10.13140/RG.2.1.3274.8562.

StatsModels. (2024, October 29). *statsmodels.stats.outliers_influence.variance_inflation_factor - statsmodels 0.15.0 (+302)*. Www.statsmodels.org. https://www.statsmodels.org/dev/generated/statsmodels.stats.outliers_influence.variance_inflation_factor.html

Streamlit. (2024). *Streamlit — The fastest way to create data apps*. Www.streamlit.io. https://www.streamlit.io/

Waqar, M. (2023, October 26). *Unlocking CRISP-DM: Your Path to Data Science Success*. Medium. https://medium.com/@mwaqarbatalvi/mastering-the-crisp-dm-framework-your-path-to-successful-data-science-projects-56f15d6f4c54

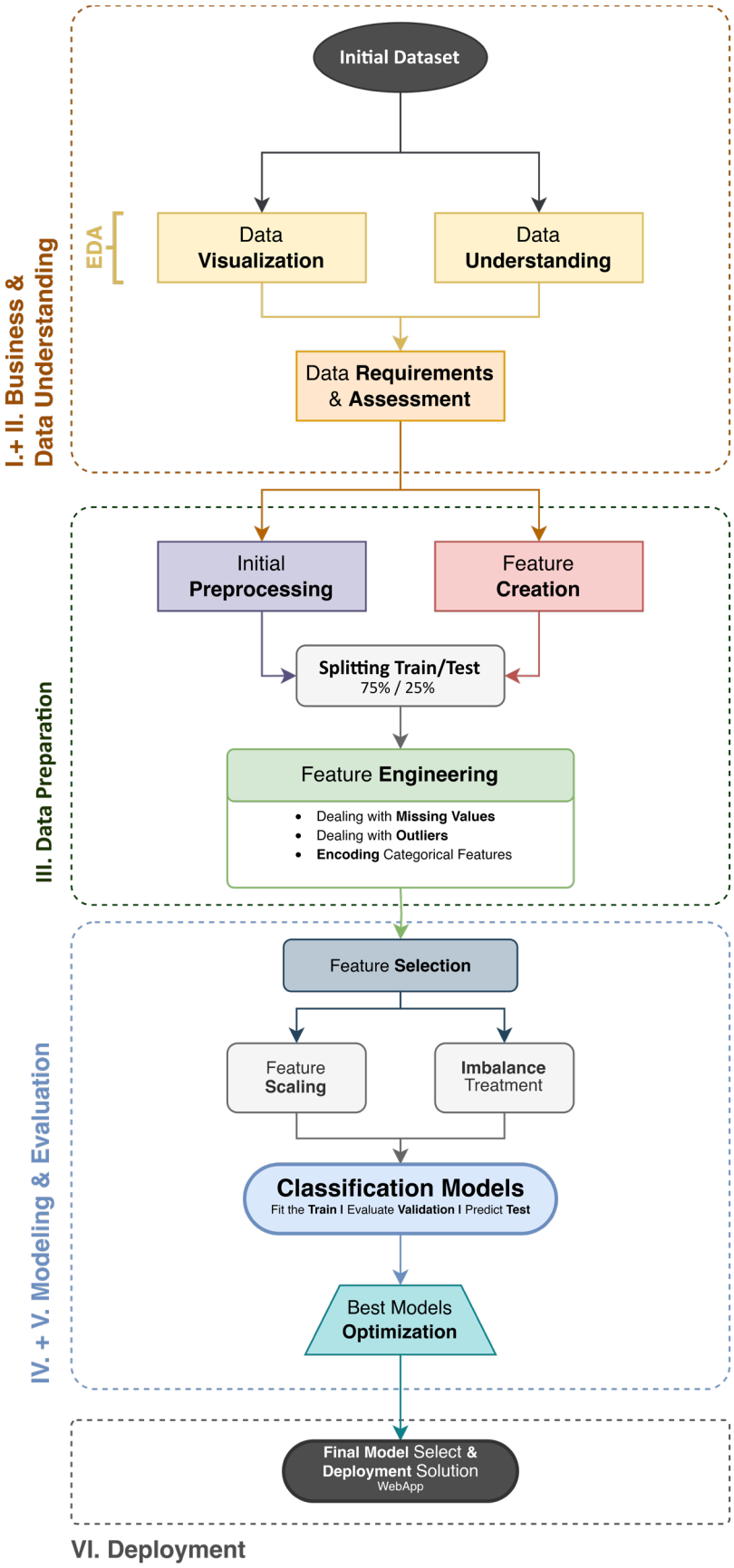
Yandex. (2019). *CatBoost - state-of-the-art open-source gradient boosting library with categorical features support*. Catboost.ai. https://catboost.ai/

APPENDIX A. LITERATURE REVIEW

Table A1 – Literature Review

Reference	Objective of the Study	Class Imbalance Handling	Feature Selection/Importance Techniques	Key Feature(s) Identified	Final Model Selected
Alkaissy et al., 2023	Predict four injury types: Upper limbs, Lower limbs, Head/Neck, and Back/Trunk	-	Based on literature review/domain knowledge followed by PCA	*Accident Nature* and *Accident Mechanism*	Random Forest
Zul et al., 2022	Predicting two binary target features: Hospitalization and Amputation	-	Based on literature review/domain knowledge	*Nature of Injury, Type of Event* and *Affected Body Part*	Random Forest
Sarkar et al., 2020	Predicting injury severity outcome as Fatal, Medical Case or First Aid	Different oversampling algorithms were used with K-Means SMOTE performing better	Boruta Algorithm	*Safety Standard* as the most important predictor	Random Forest
Mathews, 2016	Classify claims as Severe or Non-Severe based on incurred loss	Used the class weights parameter in SVM to assign higher weights to the minority class	Regularized Logistic Regression with Lasso penalty	*Cause of injury*, *Loss description topic*, *Target Body* part among others	Support Vector Machine (SVM)

APPENDIX B. PROJECT SCHEMA



APPENDIX C. EDA

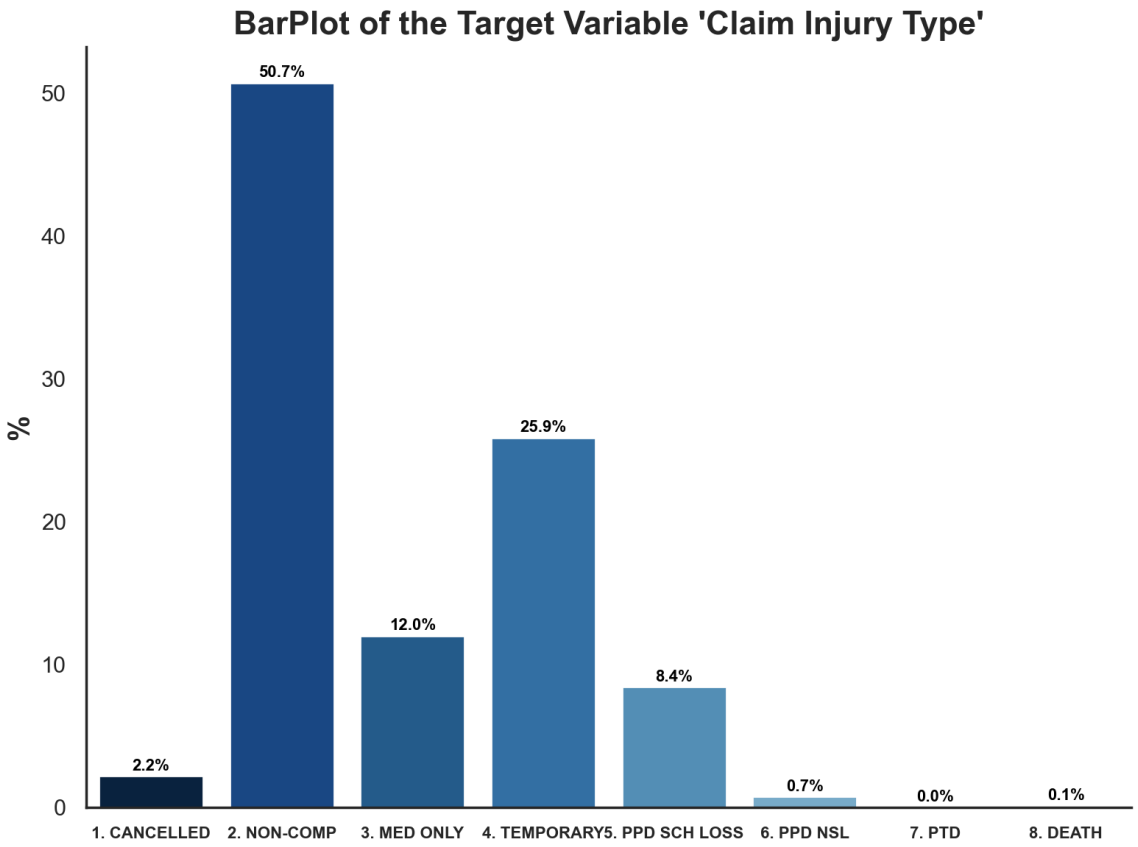


Figure C1 – Imbalanced Classes of the Target Variable.

Table C1. – Feature creation and dropping summary.

Feature Dropped	Reason/Issue	Feature Created
Accident Date, *Assembly Date*, *C-2 Date*, *C-3 Date*, *First Hearing Date*	Original features were replaced with derived features (Year, Month, Day, Weekday) to ensure consistency and capture granular temporal information across all features	Year, Month, Day, Weekday column for each feature
Accident Date, *C-2 Date*, *C-3 Date*, *First Hearing Date*	Missing values	Binary features (1 if observation had a missing value)
C-3 Date and First Hearing Date Year, Month, Weekday, Day related features	High number of missing values in the original features makes extracted columns *Year, Month, Weekday and Day* unsuitable for modelling	Binary features (1 if observation had a missing value)
Age at Injury	Missing Values	*Age at Injury Clean*
Average Weekly Wage	Missing Values	*Average Weekly Wage Reported*
Birth Year Clean	Redundant	*Age at Injury Clean*
Birth Year	Missing Values	*Birth Year Clean*
Carrier Name	Text data	-
Carrier Type	High Cardinality	Carrier Type Bucket
IME-4 Count	Missing Values	IME-4 Count Reported
Industry Code	Redundant (*Industry Code Description*)	-
WCIO Cause of Injury Code, WCIO Part of Body Code, WCIO Nature of Injury Code	Redundant (Decided to use Descriptions)	-
WCIO Cause of Injury Description, WCIO Part of Body Description, WCIO Nature of Injury Description	High Cardinality	WCIO Cause of Injury Bucket, WCIO Part of Body Bucket, WCIO Nature of Injury Bucket
Zip Code	High Cardinality and inconsistent	-

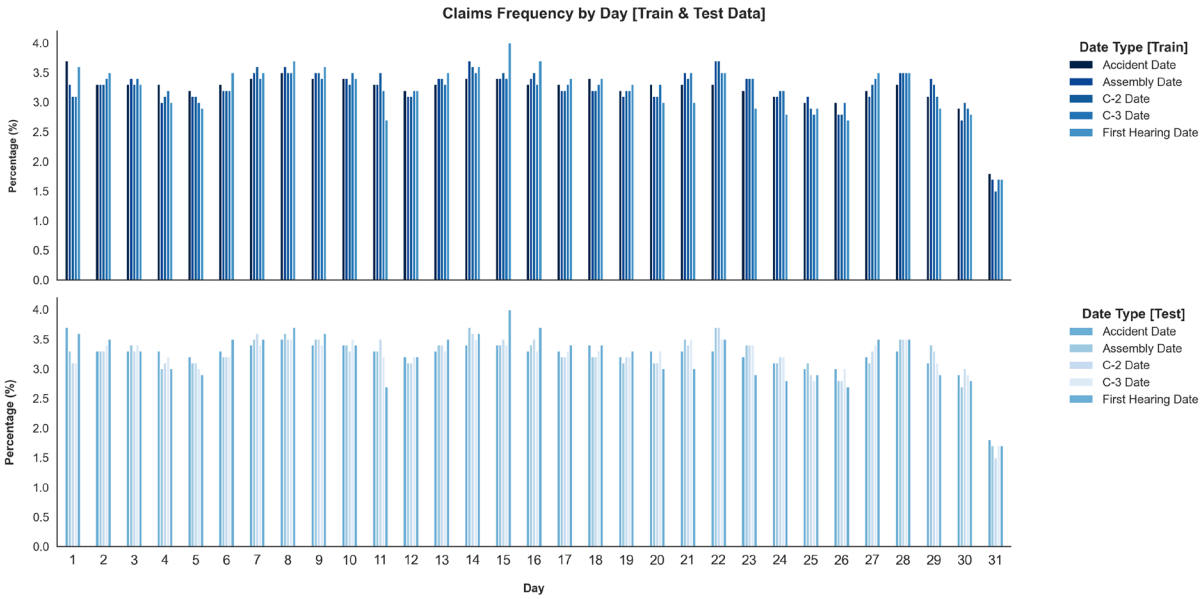


Figure C2. – Claims Frequency by Day

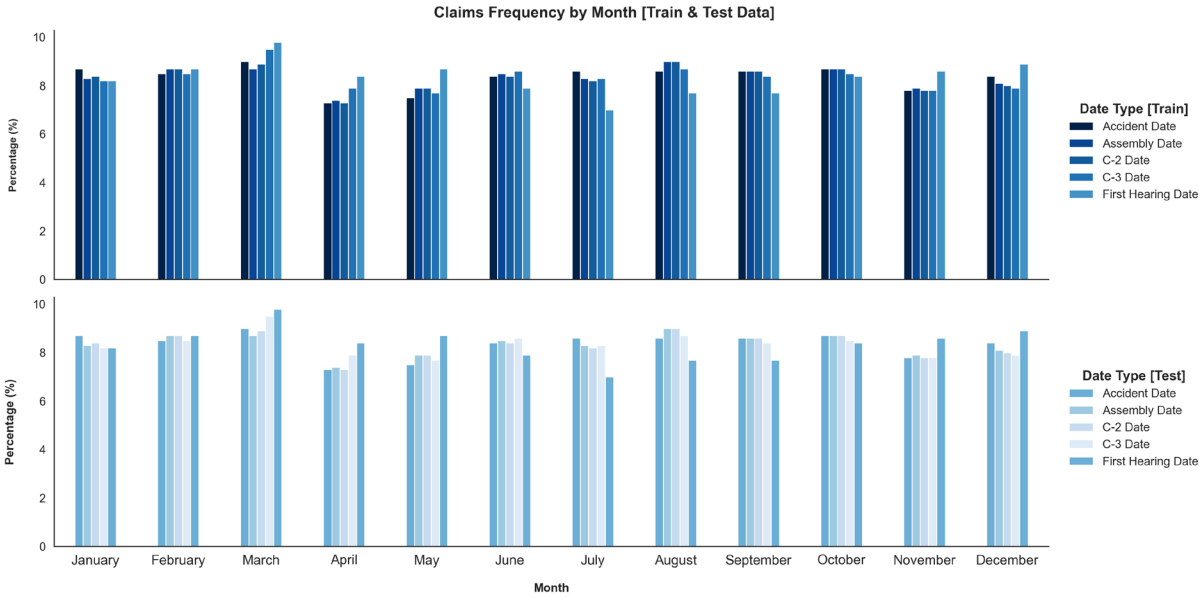


Figure C3. – Claims Frequency by Month

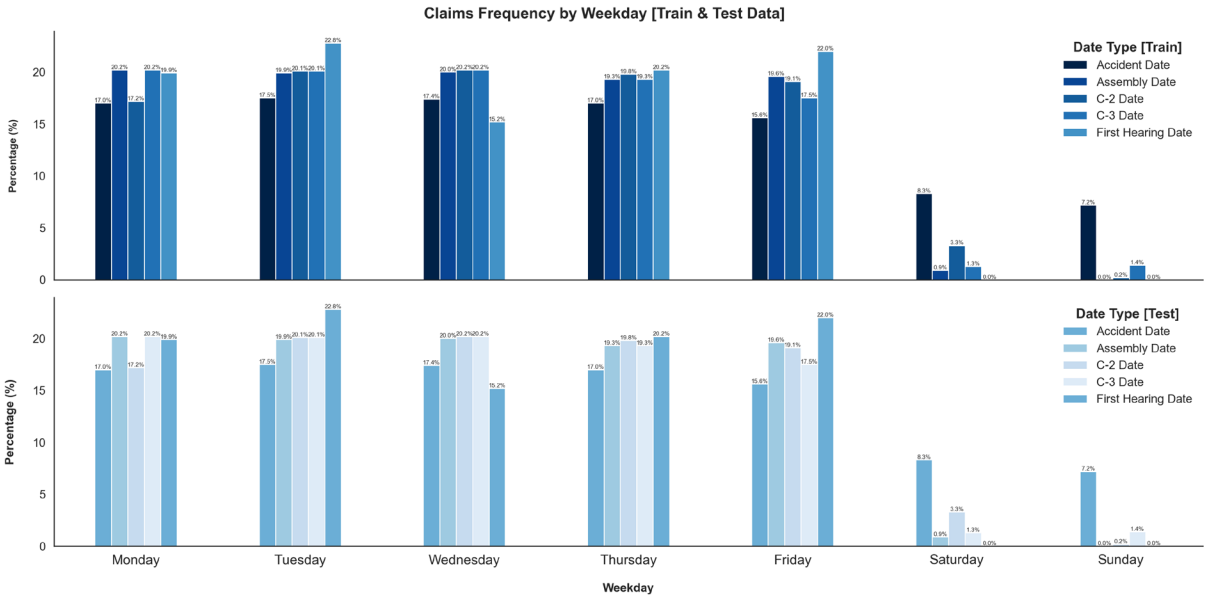


Figure C4. – Claims Frequency by Weekday

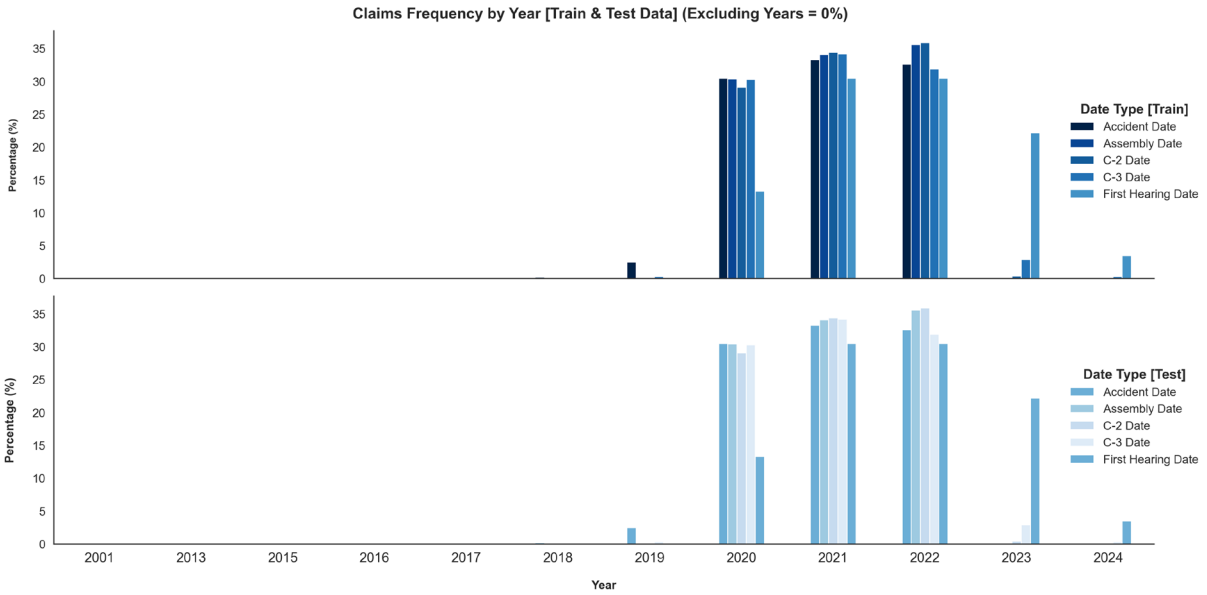


Figure C5. – Claims Frequency by Year

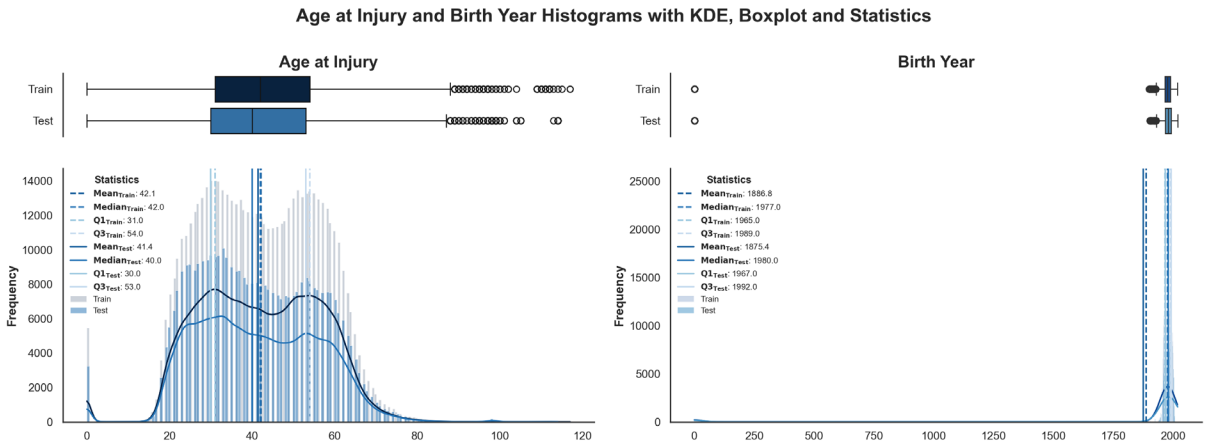


Figure C6. – Graphs of *Age at Injury* and *Birth Year*

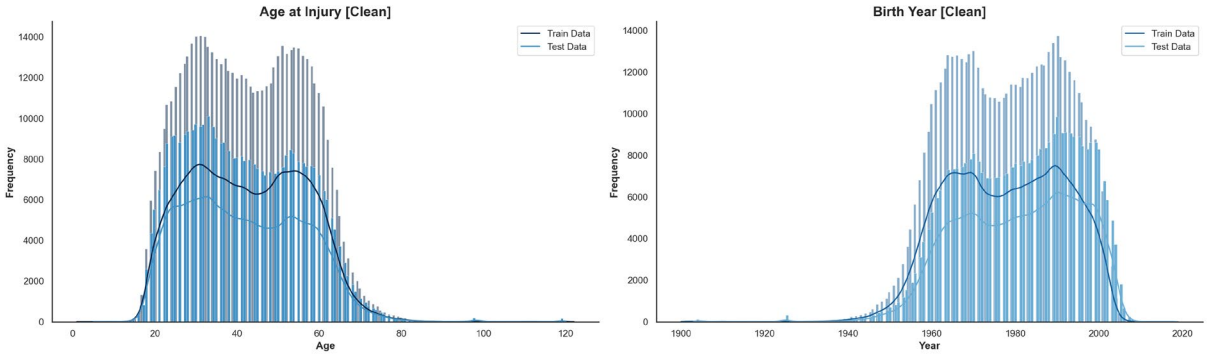


Figure C7. – Distributions of *Age at Injury Clean* and *Birth Year Clean*

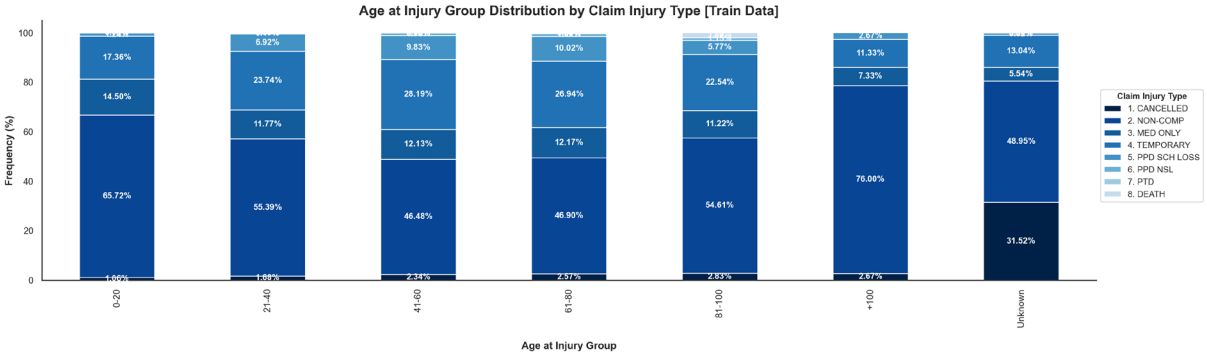


Figure C8. – Distributions of *Claim Injury Type* by *Age at Injury Group*

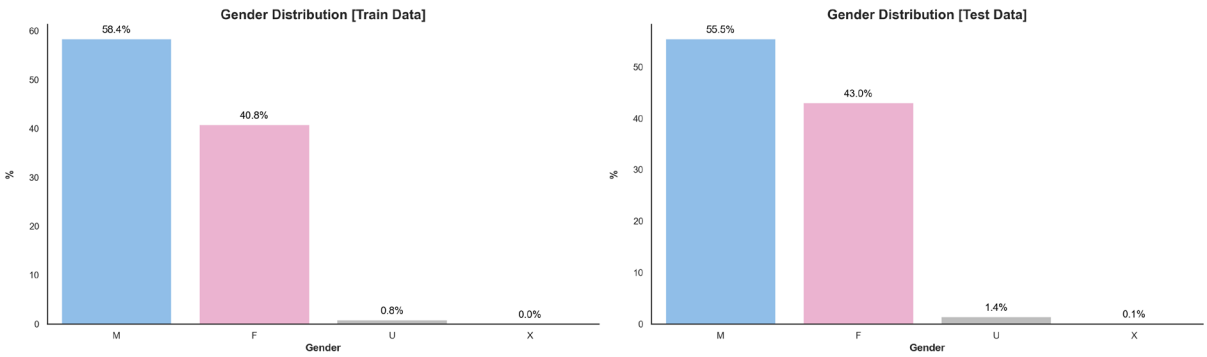


Figure C9. – *Gender* distributions

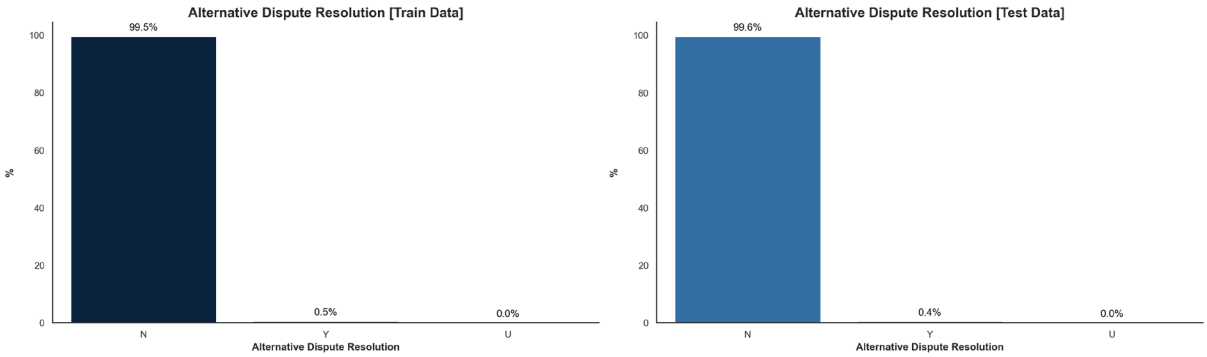


Figure C10. – Frequencies of each category in *Alternative Dispute Resolution*

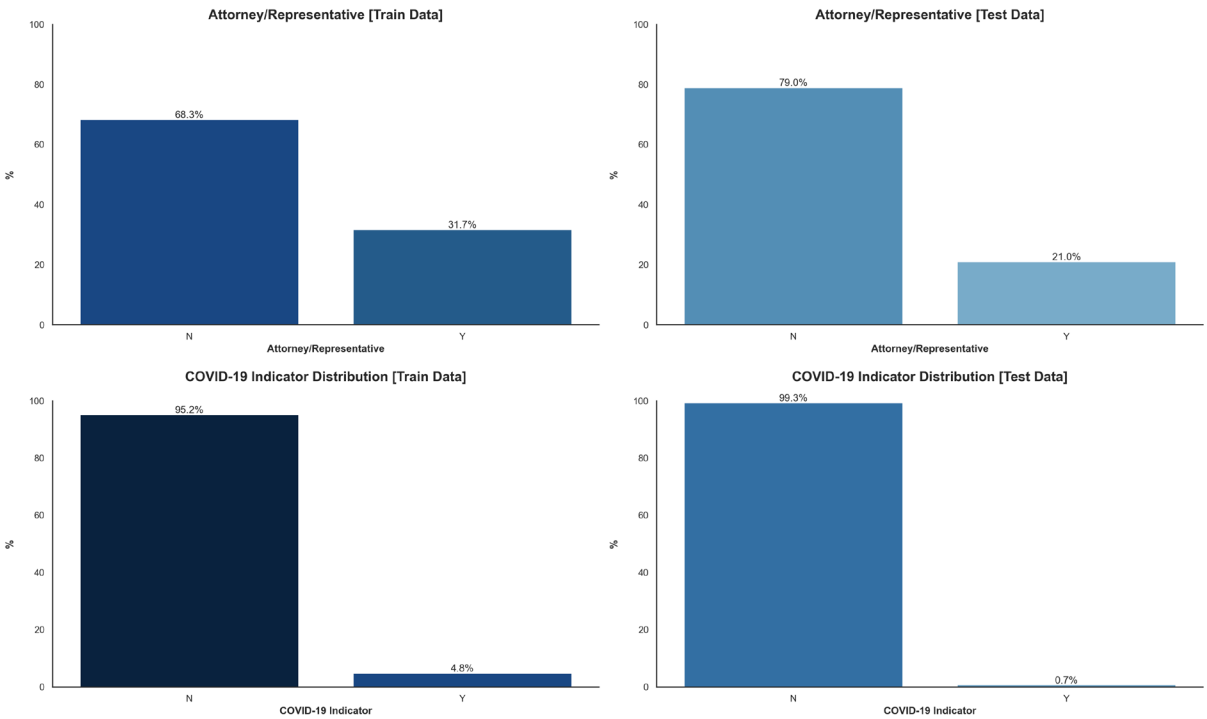


Figure C11. – Frequencies of each category in *Attorney/Representative* & *COVID-19 Indicator*

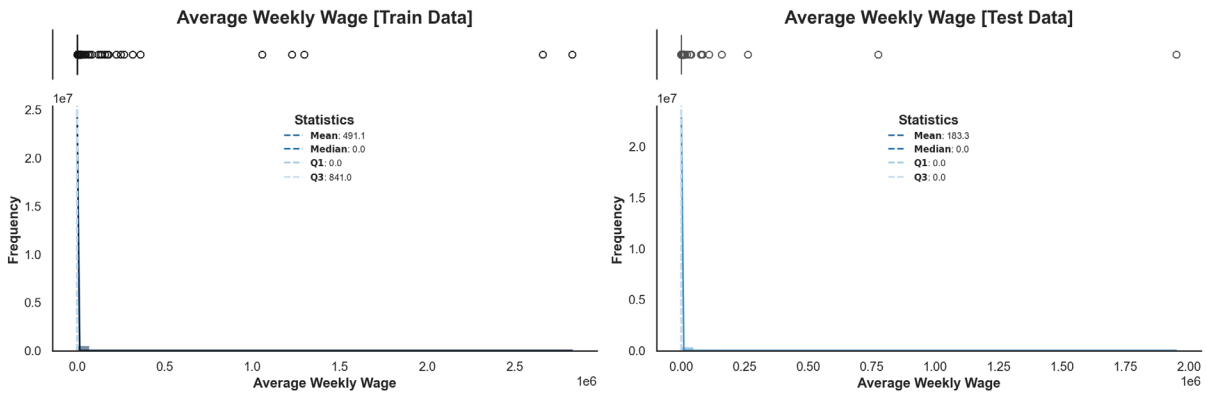


Figure C12. – Graphs of *Average Weekly Wage*

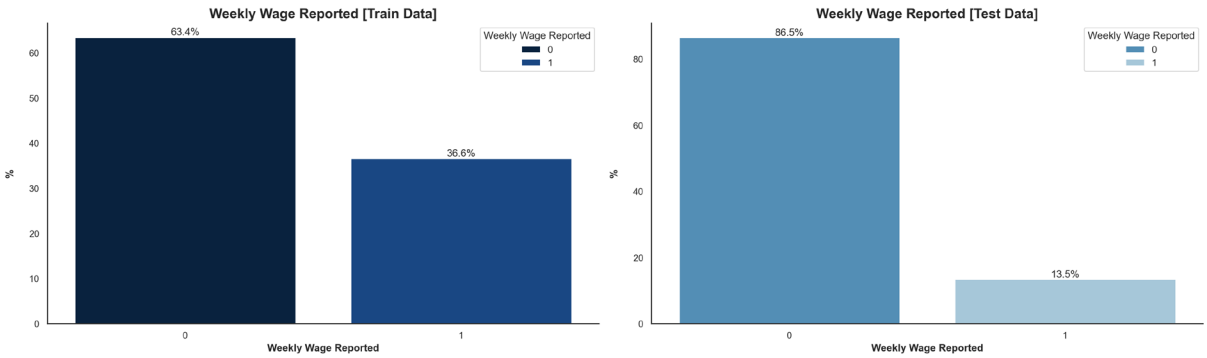


Figure C13. – Frequencies of each category in *Average Weekly Wage Reported*

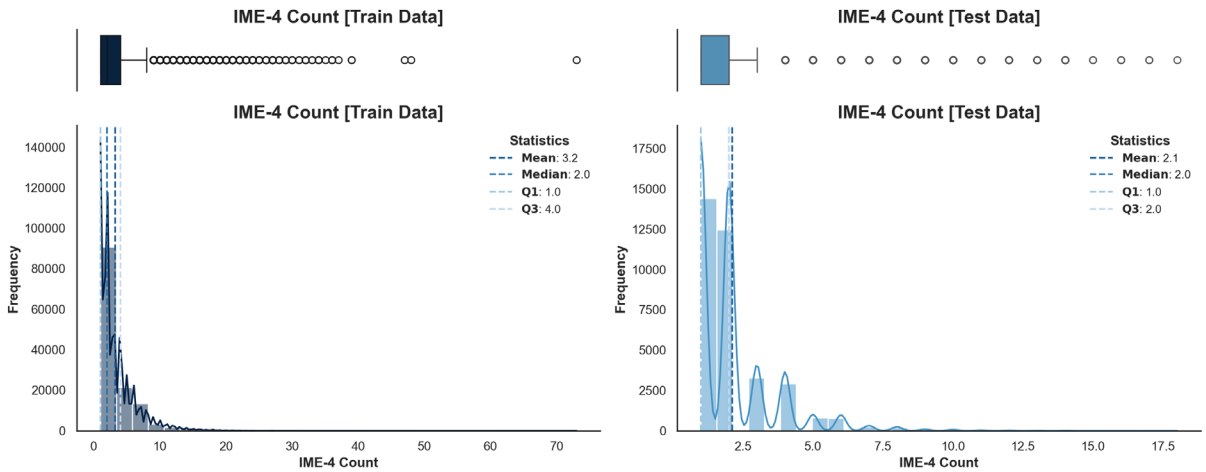


Figure C14. – Frequencies of *IME-4 Count*

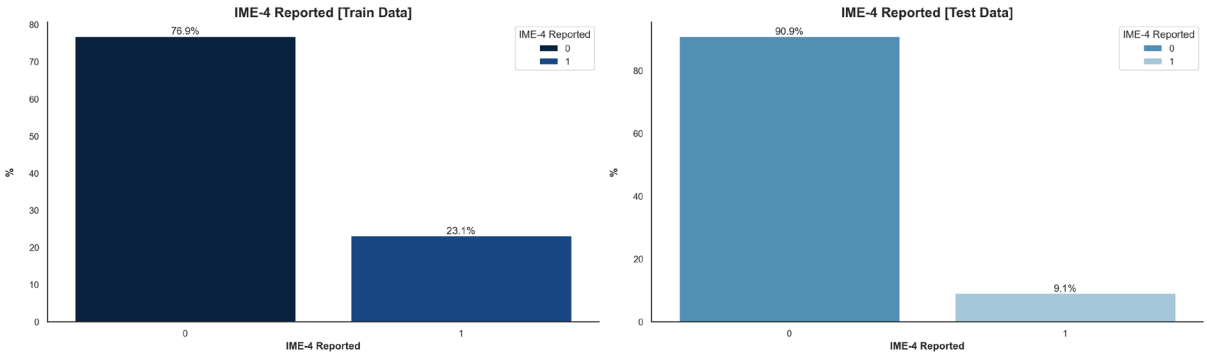


Figure C15. – Frequencies of each category in *IME-4 Reported*

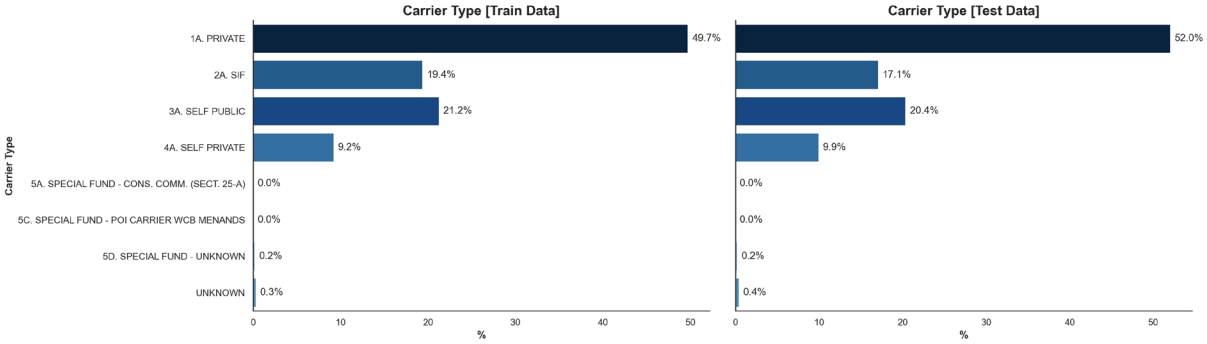


Figure C16. – Frequency of each *Carrier Type* category

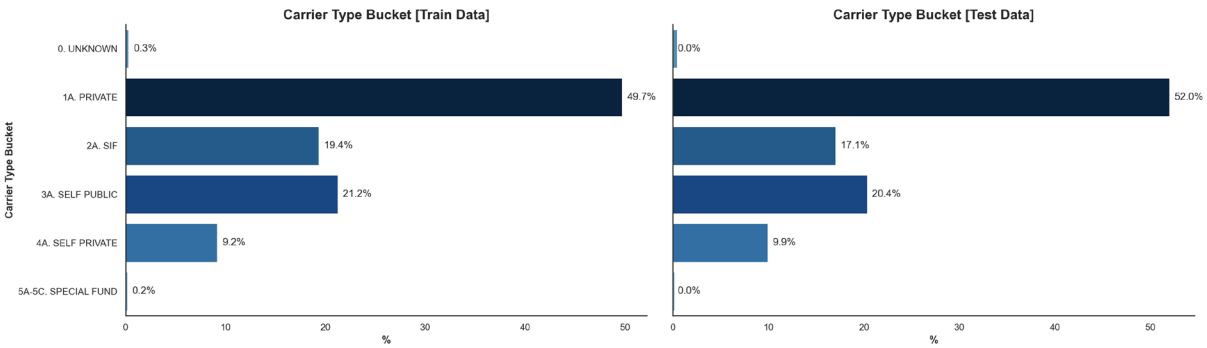


Figure C17. – Frequency of each *Carrier Type Bucket* category

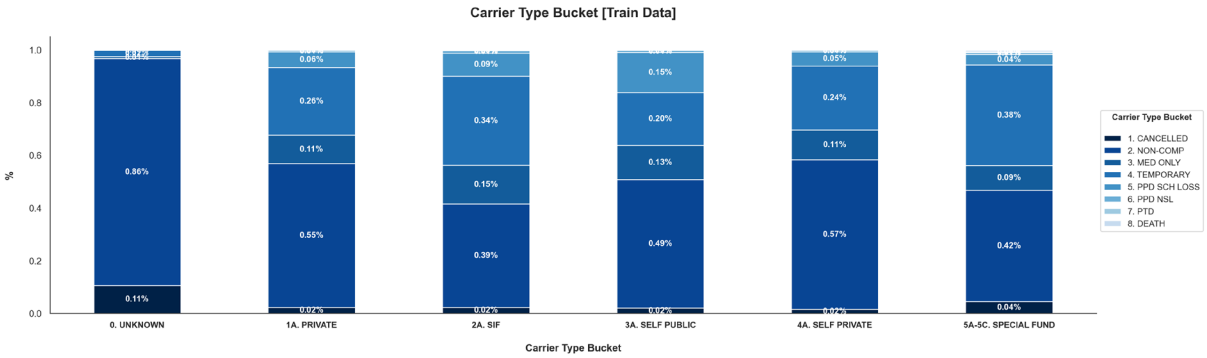


Figure C18. – Carrier Type Bucket Distribution by Claim Type (Train Data)

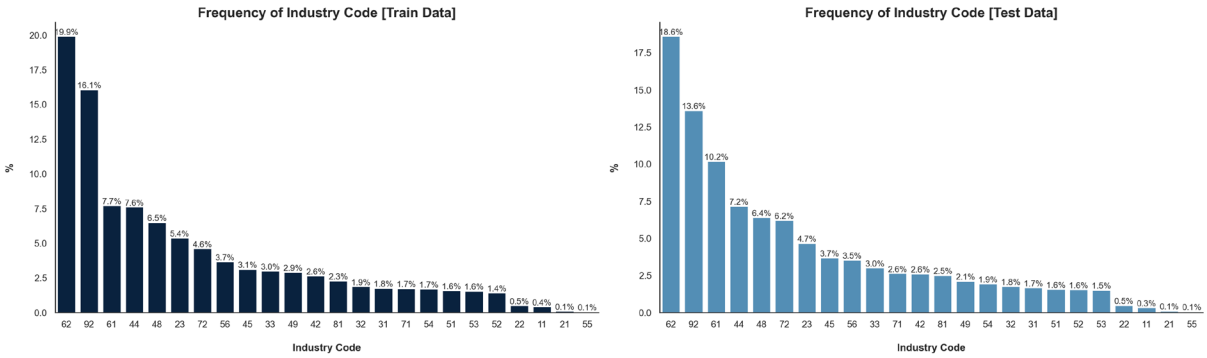


Figure C19. – Frequency of each category in *Industry Code*

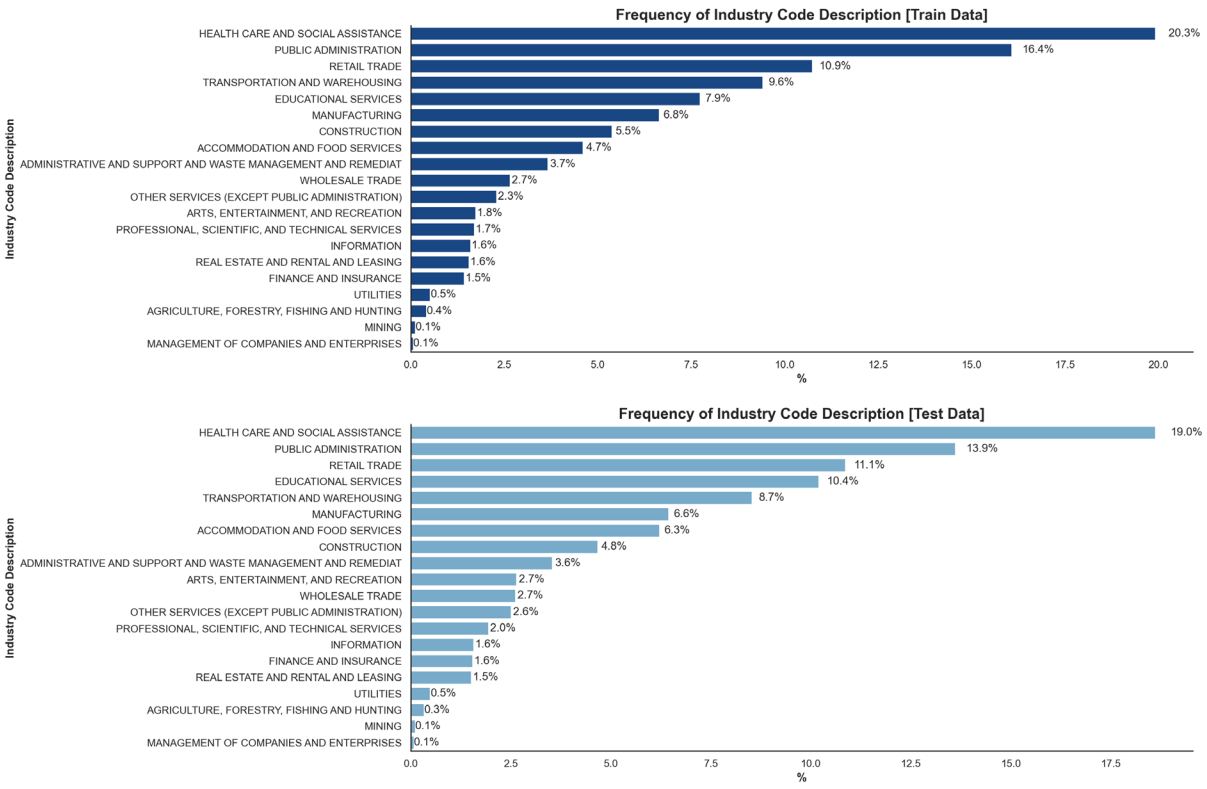


Figure C20. – Frequency of each category in *Industry Code Description*

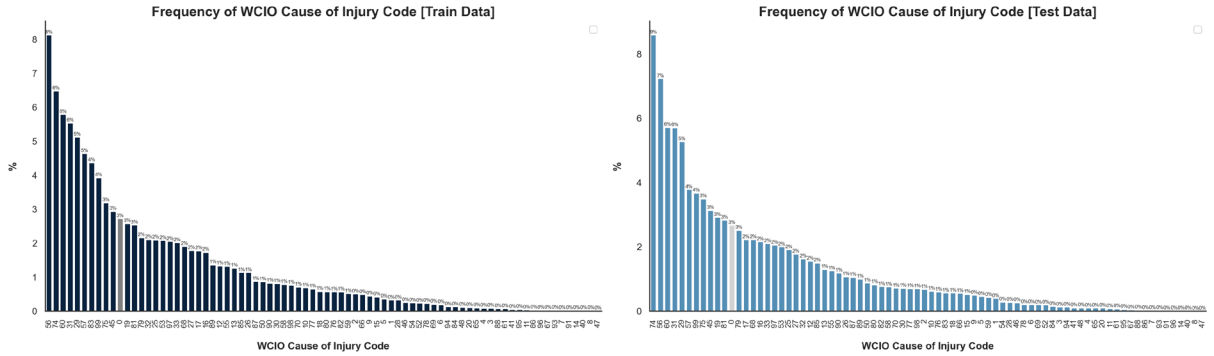


Figure C21. – Frequency of each category in *WCIO Cause of Injury Code*

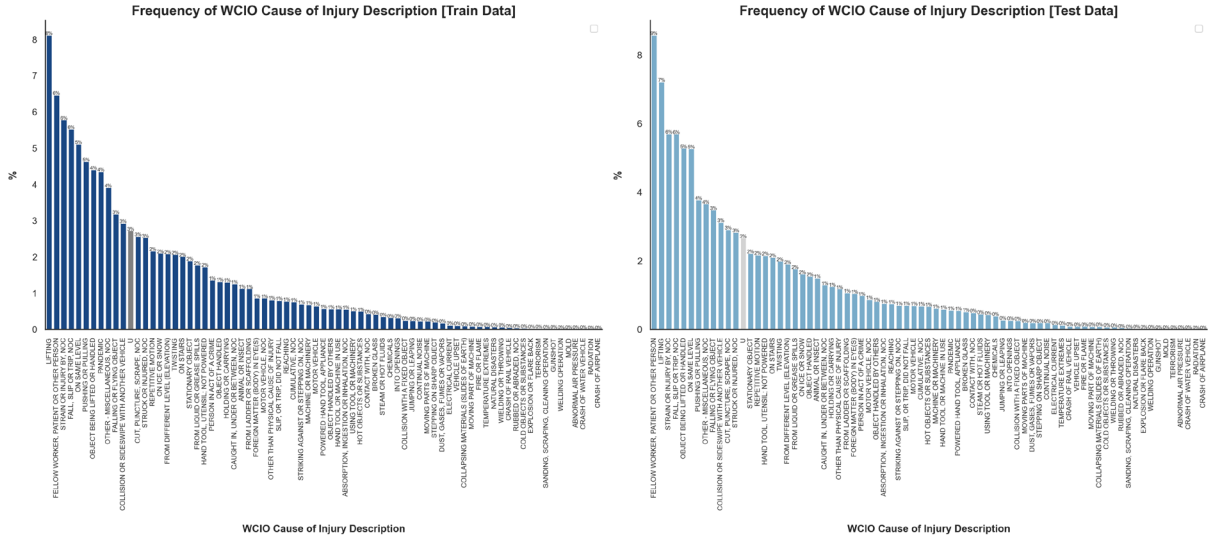


Figure C22. – Frequency of each category in *WCIO Cause of Injury Description*

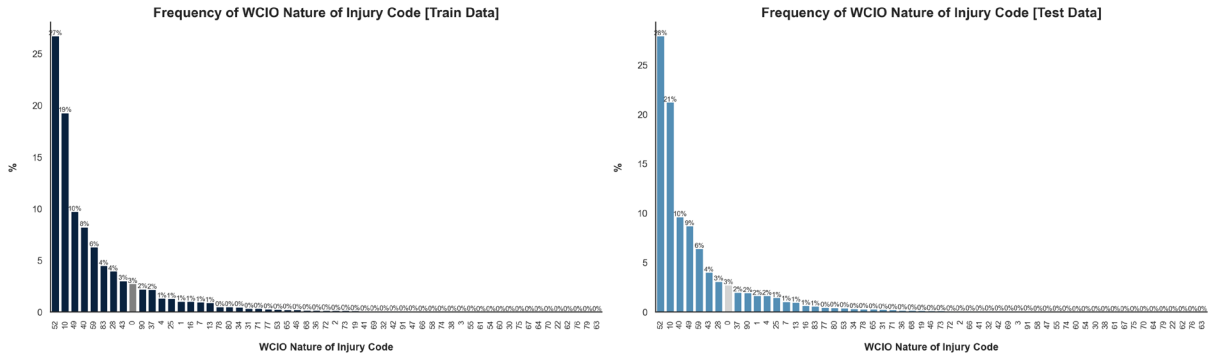


Figure C23. – Frequency of each category in *WCIO Nature of Injury Code*

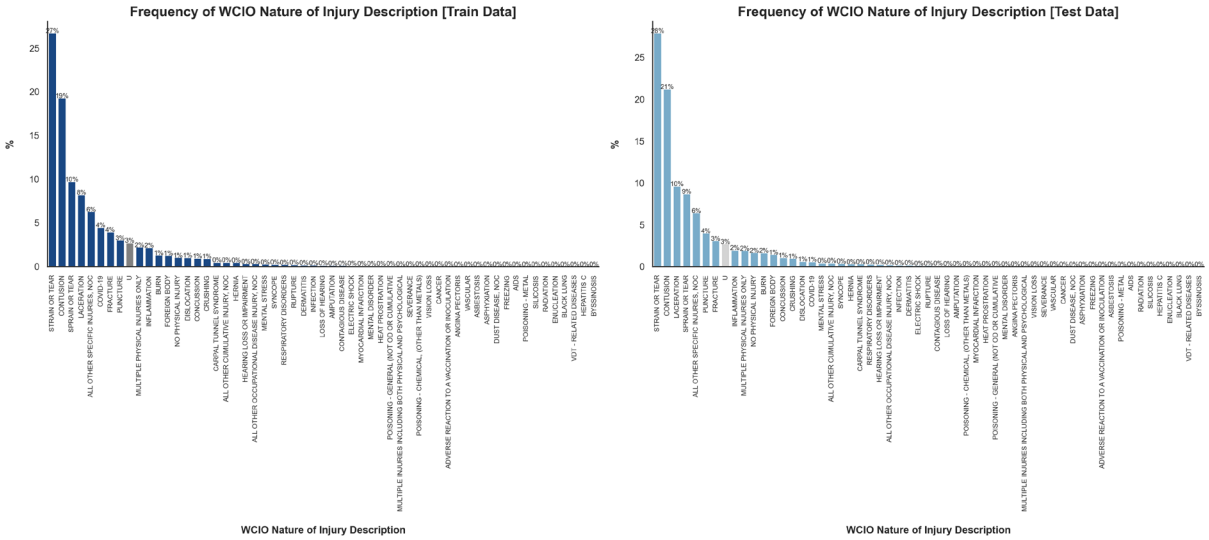


Figure C24. – Frequency of each category in *WCIO Nature of Injury Description*

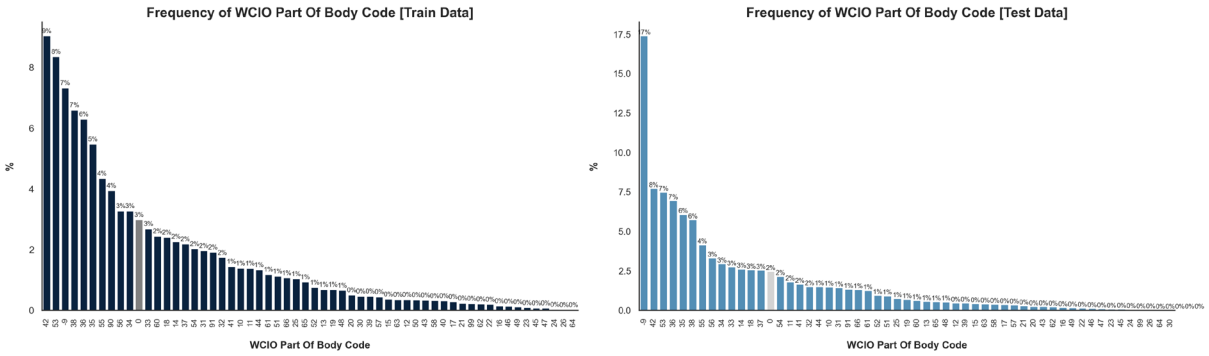


Figure C25. – Frequency for each category in *WCIO Part of Body Code*

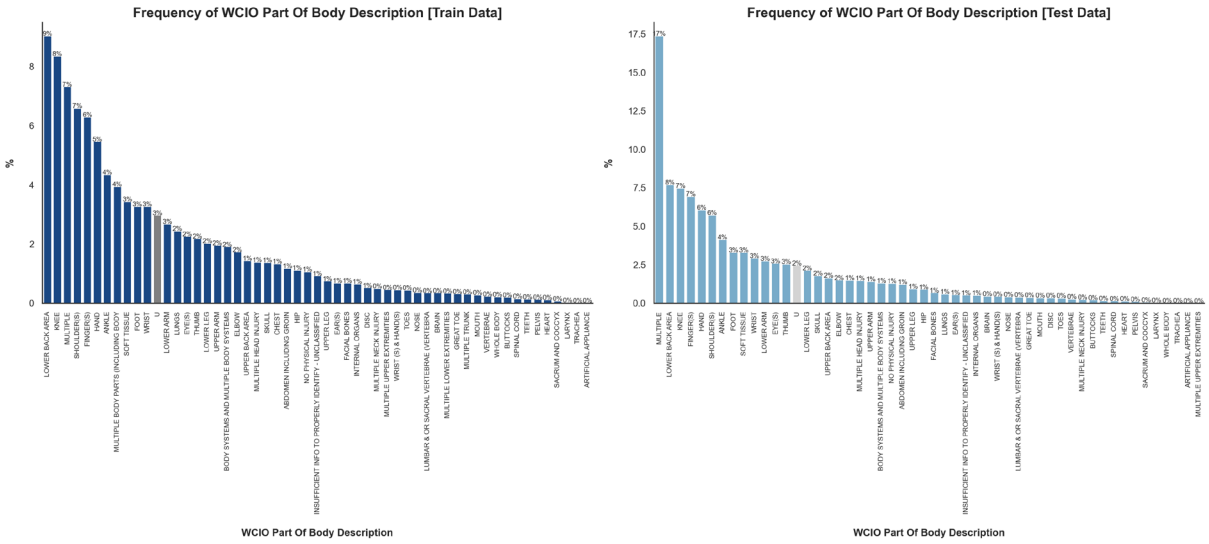


Figure C26. – Frequency of each category in *WCIO Part of Body Description*

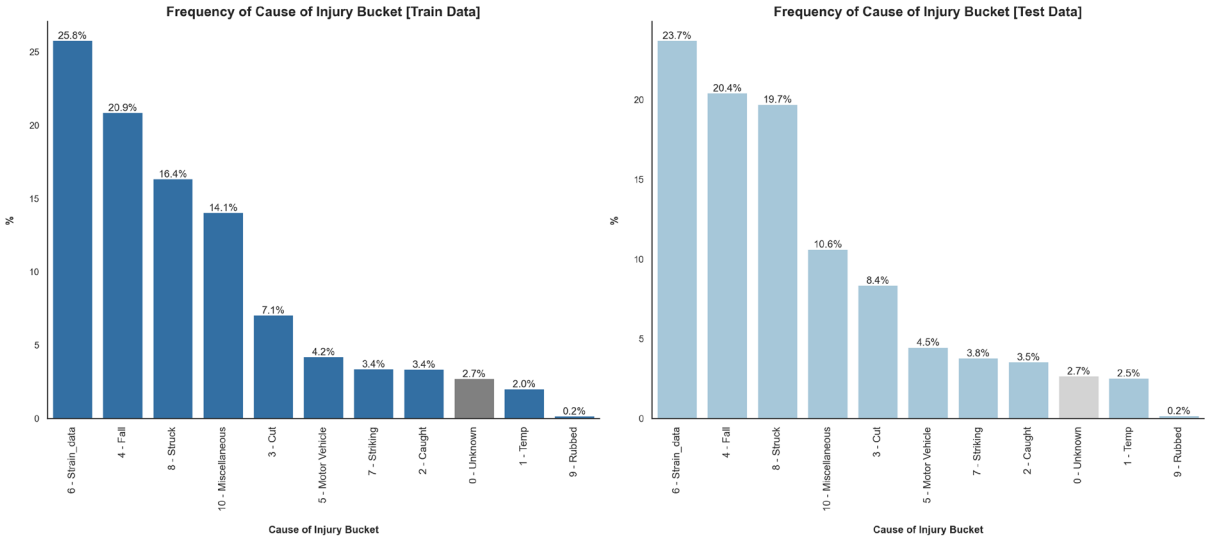


Figure C27. – Frequency of each category in *WCIO Cause of Injury Bucket*

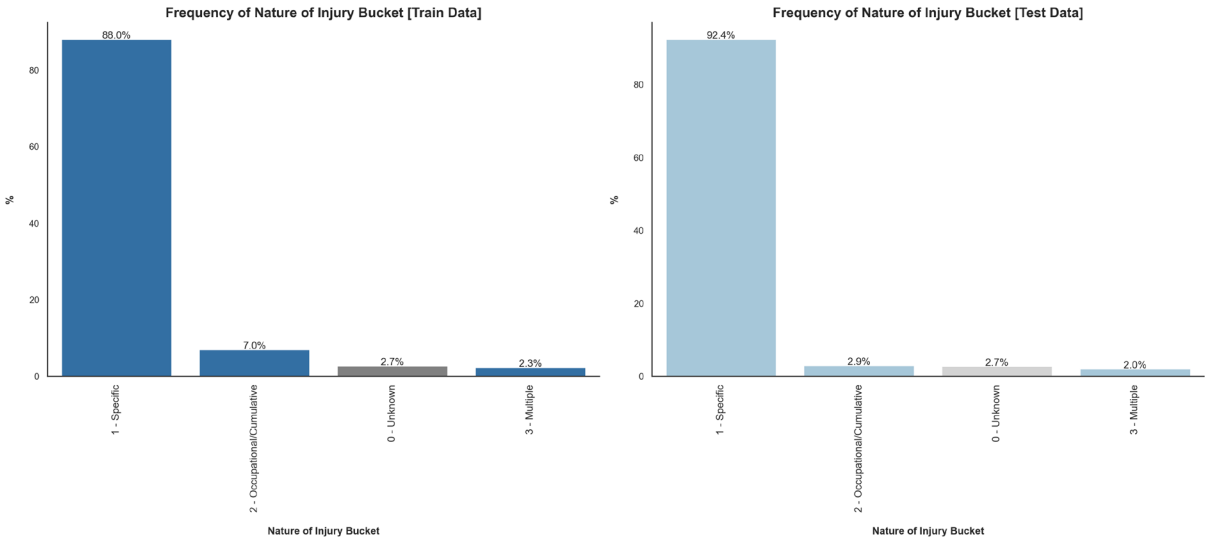


Figure C28. – Frequency for each category in *WCIO Nature of Injury Bucket*

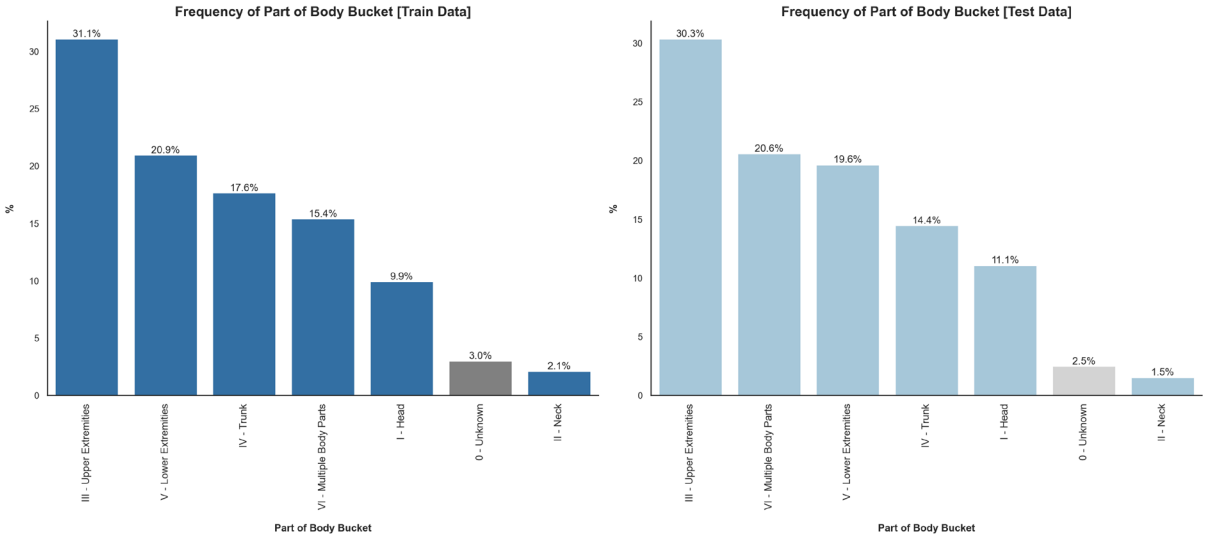


Figure C29. – Frequency for each category in *WCIO Part of Body Bucket*

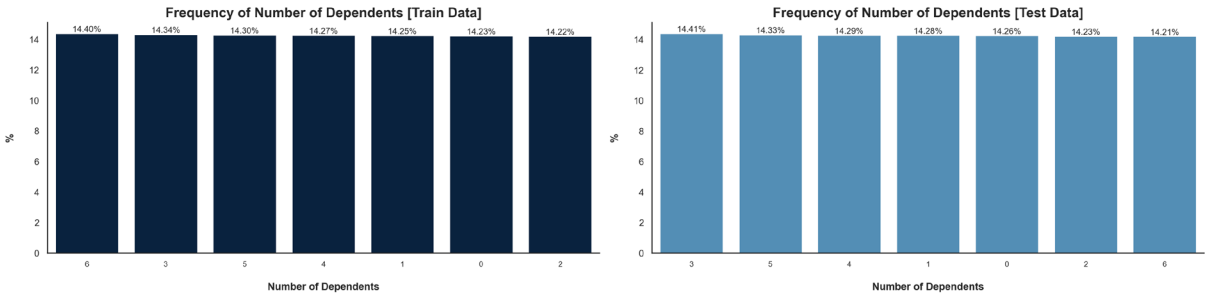


Figure C30. – Frequency of *Number of Dependents*

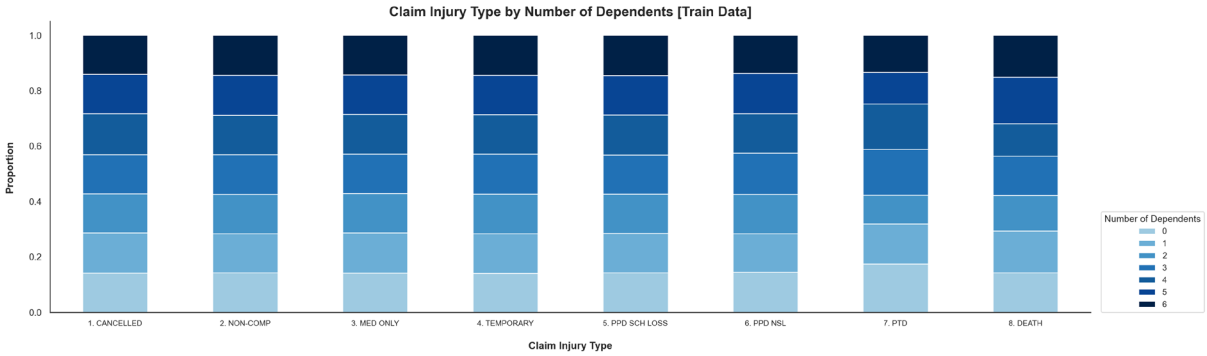


Figure C31. – Proportion of *Number of Dependents* per *Claim Injury Type*

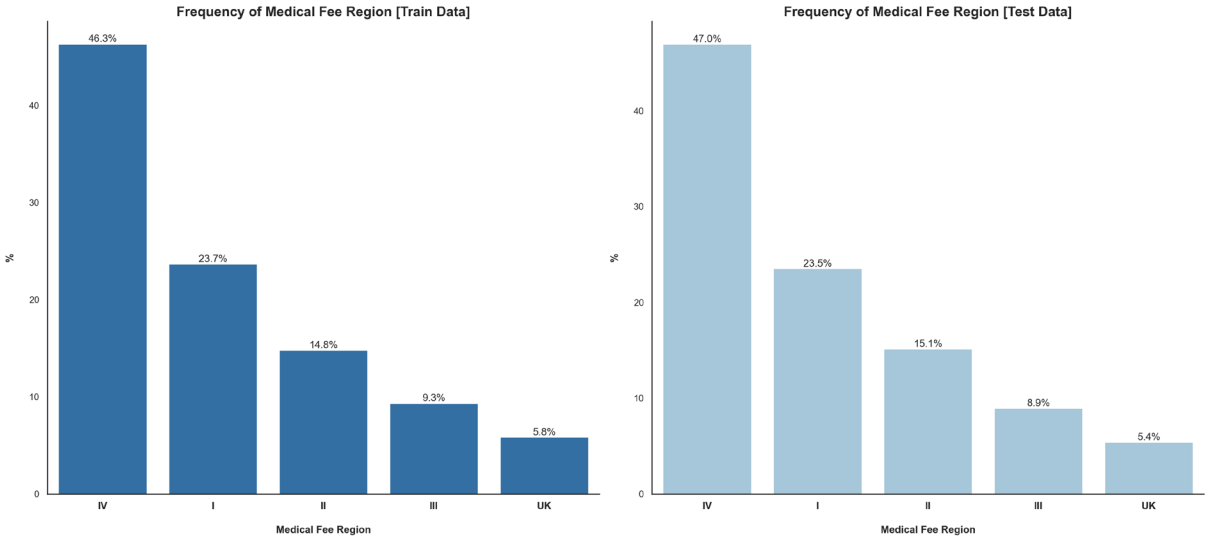


Figure C32. – Frequency for each category in *Medical Fee Region*

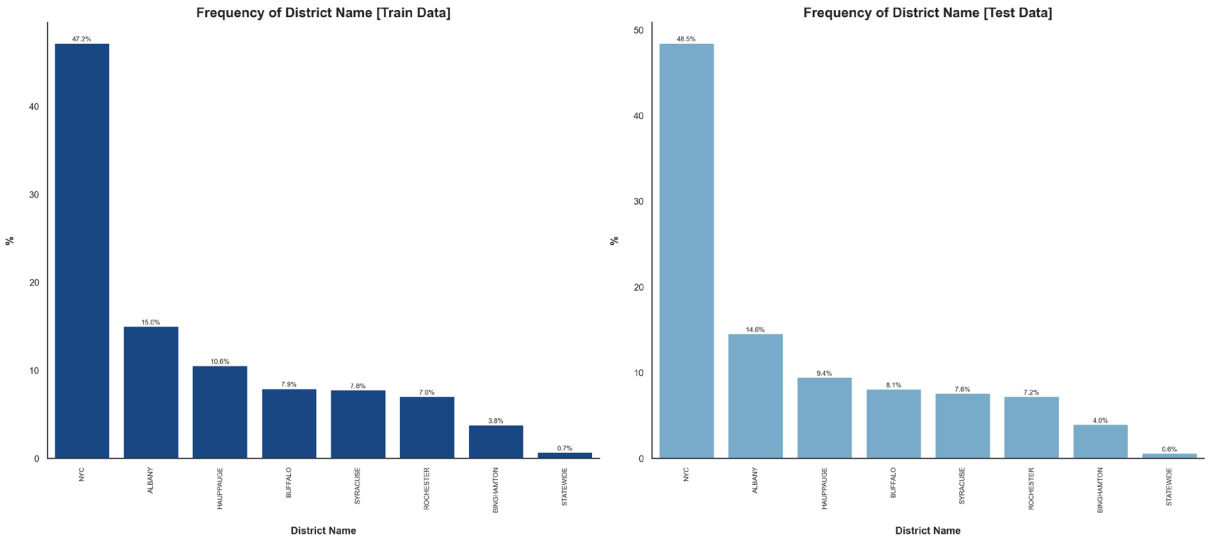


Figure C33. – Frequency of each category in *District Name*

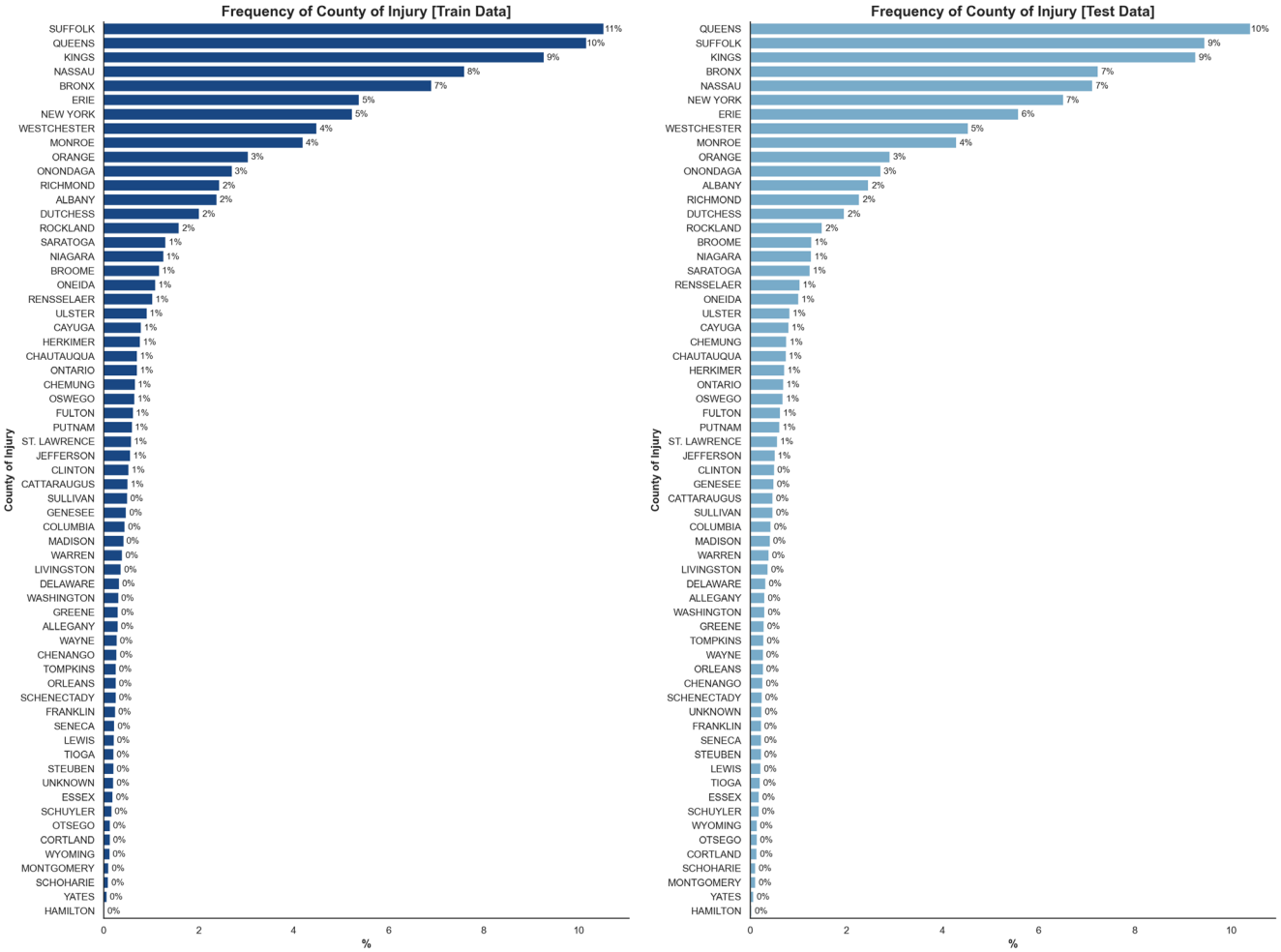


Figure C34. – Frequency for each category in *County of Injury*

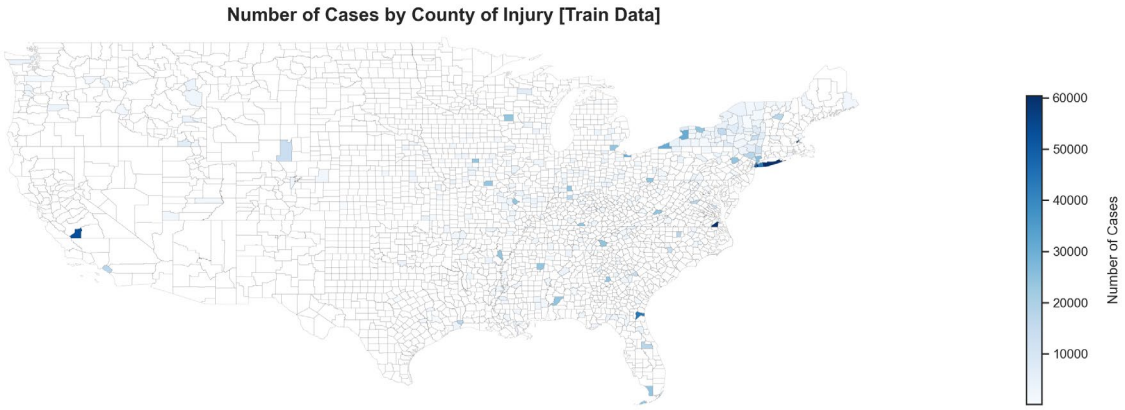


Figure C35. – Number of Cases per *County of Injury in Train Dataset*

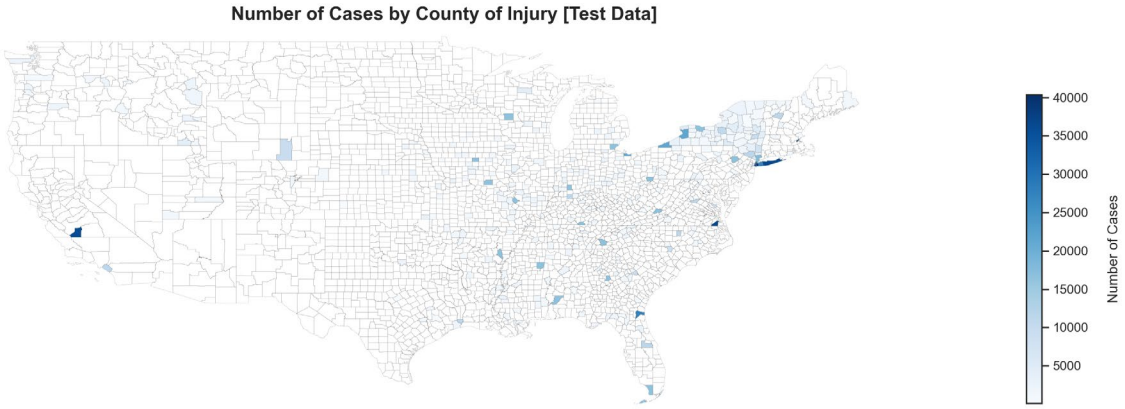


Figure C36. – Number of Cases per *County of Injury in Test Dataset*

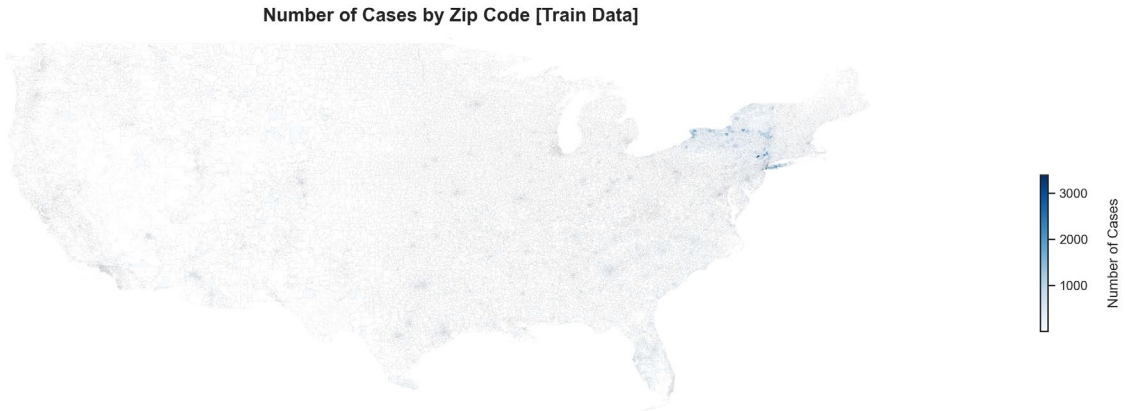


Figure C37. – Number of Cases per *Zip Code*

APPENDIX D. FEATURE SELECTION

Numerical Features

Features	Multicollinearity/ Redundancy (VIF, Spearman)	Correlation/ Relevancy (Spearman)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	What to Do?
Accident Date Day	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Accident Date Month	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Accident Date Weekday	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Accident Date Year	Not Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	**Drop**
Age at Injury Clean	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Not Selected	**Keep**
Assembly Date Day	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Assembly Date Month	Not Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Assembly Date Weekday	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
Assembly Date Year	Not Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	**Keep**
C-2 Date Day	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
C-2 Date Month	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
C-2 Date Weekday	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**
C-2 Date Year	Not Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	**Keep**
Number of Dependents	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	**Drop**

Categorical Features

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
Accident Date Binary	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Age at Injury Group	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Alternative Dispute Resolution_U	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Discard	**Drop**
Alternative Dispute Resolution_Y	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Attorney/Representative_Y	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
C-2 Date Binary	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Important	**Keep**
C-3 Date Binary	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
COVID-19 Indicator_Y	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Carrier Type Bucket_1A. PRIVATE	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Carrier Type Bucket_2A. SIF	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Carrier Type Bucket_3A. SELF PUBLIC	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Carrier Type Bucket_4A. SELF PRIVATE	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Important	**Keep**
Carrier Type Bucket_5A-5C. SPECIAL FUND	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Not Selected	Selected	Selected	Not Selected	Selected	Important	**Keep**
County of Injury_ALLEGANY	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_BRONX	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_BROOME	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CATTARAUGUS	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CAYUGA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CHAUTAUQUA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CHEMUNG	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CHENANGO	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
County of Injury_CLINTON	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_COLUMBIA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_CORTLAND	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_DELAWARE	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_DUTCHESS	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ERIE	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ESSEX	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_FRANKLIN	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_FULTON	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_GENESEE	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_GREENE	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_HAMILTON	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_HERKIMER	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_JEFFERSON	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_KINGS	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_LEWIS	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_LIVINGSTON	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_MADISON	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_MONROE	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_MONTGOMERY	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Discard	**Drop**
County of Injury_NASSAU	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
County of Injury_NEW YORK	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_NIAGARA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ONEIDA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ONONDAGA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ONTARIO	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ORANGE	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ORLEANS	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Discard	**Drop**
County of Injury_OSWEGO	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_OTSEGO	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_PUTNAM	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_QUEENS	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_RENSSELAER	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_RICHMOND	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ROCKLAND	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SARATOGA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SCHENECTADY	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SCHOHARIE	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SCHUYLER	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SENECA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ST. LAWRENCE	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_STEUBEN	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
County of Injury_SUFFOLK	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_SULLIVAN	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_TIOGA	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_TOMPKINS	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_ULSTER	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_UNKNOWN	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_WARREN	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_WASHINGTON	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_WAYNE	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_WESTCHESTER	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_WYOMING	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
County of Injury_YATES	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Discard	**Drop**
District Name_BINGHAMTON	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
District Name_BUFFALO	Not Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
District Name_HAUPPAUGE	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
District Name_NYC	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Important	**Keep**
District Name_ROCHESTER	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
District Name_STATEWIDE	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Important	**Drop**
District Name_SYRACUSE	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
First Hearing Date Binary	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Gender_M	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Important	**Keep**
Gender_U	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
Gender_X	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
IME-4 Reported	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Industry Code Description_ADMINISTRATIVE AND SUPPORT AND WASTE MANAGEMENT AND REMEDIAT	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_AGRICULTURE, FORESTRY, FISHING AND HUNTING	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_ARTS, ENTERTAINMENT, AND RECREATION	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_CONSTRUCTION	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Industry Code Description_EDUCATIONAL SERVICES	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_FINANCE AND INSURANCE	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_HEALTH CARE AND SOCIAL ASSISTANCE	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_INFORMATION	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_MANAGEMENT OF COMPANIES AND ENTERPRISES	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_MANUFACTURING	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_MINING	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_OTHER SERVICES (EXCEPT PUBLIC ADMINISTRATION)	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_PROFESSIONAL,	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
SCIENTIFIC, AND TECHNICAL SERVICES															
Industry Code Description_PUBLIC ADMINISTRATION	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Important	**Keep**
Industry Code Description_REAL ESTATE AND RENTAL AND LEASING	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_RETAIL TRADE	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_TRANSPORTATION AND WAREHOUSING	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_UTILITIES	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Industry Code Description_Unknown	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Important	**Keep**
Industry Code Description_WHOLESALE TRADE	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Medical Fee Region_II	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Medical Fee Region_III	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Medical Fee Region_IV	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
Medical Fee Region_UK	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_1 - Temp	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Important	**Drop**
WCIO Cause of Injury Bucket_10 - Miscellaneous	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_2 - Caught	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Important	**Drop**
WCIO Cause of Injury Bucket_3 - Cut	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Important	**Keep**
WCIO Cause of Injury Bucket_4 - Fall	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_5 - Motor Vehicle	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_6 - Strain_data	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Selected	Important	**Keep**

Features	Multicollinearity/ Redundancy (Cramer's V)	Correlation/ Relevancy (Cramer's V)	RFE Decision Tree	RFE Random Forest	RFE LR MinMax	RFE LR Standard	RFE LR Robust	Ridge MinMax	Ridge Standard	Ridge Robust	Lasso MinMax	Lasso Standard	Lasso Robust	Chi-Squared	What to Do?
WCIO Cause of Injury Bucket_7 - Striking	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_8 - Struck	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Cause of Injury Bucket_9 - Rubbed	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Nature of Injury Bucket_1 - Specific	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Nature of Injury Bucket_2 - Occupational/Cumulative	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Nature of Injury Bucket_3 - Multiple	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Part of Body Bucket_I - Head	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Part of Body Bucket_II - Neck	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Not Selected	Important	**Drop**
WCIO Part of Body Bucket_III - Upper Extremities	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Part of Body Bucket_IV - Trunk	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Part of Body Bucket_V - Lower Extremities	Selected	Not Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**
WCIO Part of Body Bucket_VI - Multiple Body Parts	Selected	Not Selected	Selected	Not Selected	Selected	Selected	Selected	Not Selected	Selected	Not Selected	Not Selected	Not Selected	Selected	Important	**Drop**
Weekly Wage Reported	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Selected	Important	**Keep**

APPENDIX E. PARAMETERS FOR CLASSIFICATION MODELS

Table E1 - Hyper-parameters used on standard ML classifiers.

Algorithm	Data Used	Parameters
Logistic Regression	Original, Scaled	**C**=1, **penalty**='l1', **solver**='saga', **max_iter**=1000, **n_jobs**=-1, **random_state**=2024
Naïve Bayes (Gaussian)	Original	**var_smoothing**=0.0001
Naïve Bayes, (Categorical)	Original	*Default parameters*
K-Nearest Neighbours	Scaled	**algorithm**='auto', **n_neighbors**=12, **p**=2 (*Euclidean Distance*), **weights**='uniform'
Neural Network (MLP)	Original, Scaled	**hidden_layer_sizes**=(9,9,9), **activation**='relu', **solver**='adam', **learning_rate_init**=0.001, **max_iter**=1000, **random_state**=2024
Decision Tree	Original	**criterion**='gini', **max_depth**=15, **random_state**=2024
Random Forest	Original	**criterion**='gini', **max_depth**=20, **min_samples_leaf**=2, **n_jobs**=-1, **random_state**=2024
Bagging (LG)	Original, Scaled	**estimator**=LogisticRegression(), **n_estimators**=100, **random_state**=2024

Algorithm	Data Used	Parameters
CatBoost (Annex E)	Original, Scaled	**iterations**=100, **loss_function**='MultiClass', **random_state**=2024
Extra Trees	Original, Scaled	**n_estimators**=1000, **max_depth**=20, **min_samples_leaf**=10, **random_state**=2024
Stacking (RF, LR)	Original, Scaled	**estimators**=[('**rf**', RandomForestClassifier(**max_depth**=20, **n_estimators**=1000, **min_samples_leaf**=10, **random_state**=2024))], **final_estimator**=LogisticRegression()

APPENDIX F. MODELS SELECTION CRITERIA

Table F1 - Model Selection Criteria and Retained Models.

Order	Criteria	Models Retained
1st	Models with F1 (macro) score **> 0.4** on validation data	Highlighted in **light turquoise** (**0.40**) and **light blue** (**0.41**) in **Table G1**
2nd	Models with overfitting (difference in F1 score between training and validation sets) **< 0.1** on validation data	**Random Forest (**Original**)**, **CatBoost (**Original, RobustScaler, MinMaxScaler**)**, **Extra Trees (**Original**)**
3rd	Highest accuracy (**0.78**) and AUROC (**0.92**) on validation data	**Random Forest (**Original**)**, **CatBoost (**Original, RobustScaler, MinMaxScaler**)**
4th	No significant performance improvement from scaling	**Random Forest (**Original**)**, **CatBoost (**Original**)**
5th	Highest F1-macro score on Kaggle test data	**CatBoost (**Original**)**

APPENDIX G. MODELLING RESULTS

Classification Models | Results

		Training Set						Validation Set				
	Models	Time of Execution	Accuracy	Precision	Recall	F1 Score	AUROC	Accuracy	Precision	Recall	F1 Score	AUROC
Original Data	**Logistic Regression**	807,09	0,76	0,60	0,29	0,29	0,85	0,76	0,60	0,29	0,29	0,84
	Naïve Bayes | Gaussian	0,24	0,64	0,34	0,51	0,35	0,90	0,64	0,34	0,52	0,35	0,89
	Naïve Bayes | Categorical	0,76	0,72	0,44	0,38	0,37	0,90	0,72	0,44	0,37	0,36	0,90
	K-Nearest Neighbors	0,08	0,77	0,67	0,35	0,37	0,95	0,74	0,46	0,33	0,33	0,76
	Neural Networks | MLP	122,81	0,74	0,55	0,25	0,22	0,84	0,74	0,55	0,25	0,22	0,84
	Decision Tree	1,58	0,80	0,80	0,45	0,48	0,94	0,77	0,51	0,38	0,39	0,84
	Random Forest	7,79	0,81	0,83	0,43	**0,46**	0,96	0,78	0,63	0,38	**0,40**	0,92
	Bagging	1291,50	0,51	0,31	0,13	0,09	0,68	0,51	0,31	0,13	0,09	0,66
	CatBoost	9,65	0,78	0,59	0,41	**0,43**	0,93	0,78	0,66	0,40	**0,41**	0,92
	ExtraTrees	100,03	0,85	0,91	0,58	0,65	0,96	0,77	0,46	0,38	**0,40**	0,91
	Stacking	1449,97	0,79	0,66	0,37	0,39	0,93	0,78	0,60	0,36	0,37	0,91
KMeansSMOTE (Original)*	**Logistic Regression**	4467,88	0,69	0,68	0,69	0,68	0,94	0,63	0,33	0,49	0,34	0,88
	Naïve Bayes | Gaussian	1,09	0,63	0,63	0,63	0,61	0,92	0,62	0,34	0,51	0,34	0,88
	Naïve Bayes | Categorical	2,83	0,68	0,68	0,68	0,67	0,94	0,64	0,34	0,51	0,35	0,89
	Neural Networks | MLP	60,82	0,12	0,12	0,12	0,03	0,50	0,51	0,51	0,12	0,08	0,50
	Decision Tree	6,29	0,80	0,80	0,80	0,80	0,97	0,67	0,35	0,46	0,37	0,82
	Random Forest	33,47	0,87	0,87	0,87	0,87	0,99	0,71	0,39	0,46	**0,41**	0,91
	Bagging	5662,89	0,46	0,44	0,46	0,42	0,82	0,38	0,23	0,23	0,15	0,66
	CatBoost	48,44	0,80	0,79	0,80	0,79	0,97	0,69	0,36	0,47	0,38	0,91
	ExtraTrees	524,96	0,85	0,85	0,85	0,84	0,98	0,70	0,39	0,45	**0,41**	0,90
	Stacking	8457,09	0,84	0,84	0,84	0,84	0,98	0,69	0,37	0,51	0,39	0,89
StandardScaler	**Logistic Regression**	437,17	0,77	0,61	0,37	0,39	0,92	0,77	0,59	0,37	0,38	0,92
	K-Nearest Neighbors	0,25	0,79	0,67	0,40	0,42	0,95	0,77	0,56	0,37	0,39	0,79
	Neural Networks | MLP	32,21	0,78	0,62	0,36	0,36	0,92	0,78	0,61	0,35	0,36	0,91
	Bagging	1042,66	0,77	0,53	0,37	0,39	0,92	0,77	0,60	0,37	0,38	0,92
	CatBoost	7,68	0,78	0,59	0,41	0,42	0,93	0,78	0,57	0,39	**0,41**	0,92
	Stacking	886,00	0,79	0,66	0,37	0,38	0,93	0,78	0,61	0,36	0,37	0,91
StandardScaler (with KMeansSMOTE)	**Logistic Regression**	2172,41	0,71	0,70	0,71	0,70	0,95	0,65	0,34	0,49	0,36	0,89
	K-Nearest Neighbors	0,27	0,75	0,78	0,75	0,75	0,95	0,72	0,39	0,42	**0,40**	0,77
	Neural Networks | MLP	144,14	0,75	0,75	0,75	0,75	0,96	0,65	0,34	0,47	0,36	0,89
	Bagging	4290,03	0,71	0,70	0,71	0,70	0,95	0,65	0,34	0,49	0,36	0,89
	CatBoost	31,24	0,80	0,79	0,80	0,79	0,97	0,69	0,36	0,48	0,38	0,91
	Stacking	3616,02	0,84	0,84	0,84	0,84	0,98	0,69	0,37	0,52	0,39	0,89

	Models	Training Set						Validation Set				
		Time of Execution	Accuracy	Precision	Recall	F1 Score	AUROC	Accuracy	Precision	Recall	F1 Score	AUROC
RobustScaler	Logistic Regression	1332,32	0,77	0,61	0,37	0,39	0,92	0,77	0,59	0,37	0,38	0,92
	K-Nearest Neighbors	3,71	0,79	0,69	0,40	0,42	0,95	0,77	0,61	0,38	0,39	0,80
	Neural Networks | MLP	881,02	0,78	0,64	0,38	0,39	0,92	0,78	0,63	0,38	0,39	0,92
	Bagging	3981,52	0,77	0,52	0,37	0,38	0,91	0,77	0,59	0,36	0,37	0,91
	CatBoost	23,33	0,78	0,69	0,41	0,43	0,93	0,78	0,66	0,39	**0,40**	0,92
	Stacking	2093,95	0,79	0,66	0,37	0,39	0,93	0,78	0,60	0,36	0,37	0,91
RobustScaler (with KMeansSMOTE)	Logistic Regression	6043,02	0,71	0,70	0,71	0,70	0,95	0,65	0,34	0,49	0,36	0,89
	K-Nearest Neighbors	1,09	0,75	0,78	0,75	0,75	0,95	0,73	0,38	0,42	0,39	0,77
	Neural Networks | MLP	1540,57	0,76	0,75	0,76	0,75	0,96	0,66	0,34	0,46	0,36	0,89
	Bagging	7419,21	0,71	0,70	0,71	0,70	0,95	0,65	0,34	0,50	0,36	0,89
	CatBoost	101,00	0,79	0,79	0,79	0,79	0,97	0,68	0,36	0,48	0,38	0,91
	Stacking	14441,73	0,84	0,84	0,84	0,84	0,98	0,69	0,37	0,51	0,39	0,89
MinMaxScaler	Logistic Regression	271,20	0,77	0,61	0,37	0,39	0,92	0,77	0,59	0,37	0,38	0,92
	K-Nearest Neighbors	0,24	0,79	0,69	0,40	0,43	0,95	0,77	0,57	0,38	0,39	0,80
	Neural Networks | MLP	188,75	0,78	0,67	0,36	0,38	0,92	0,78	0,64	0,36	0,37	0,92
	Bagging	2591,45	0,77	0,61	0,36	0,37	0,91	0,77	0,57	0,35	0,36	0,91
	CatBoost	55,48	0,78	0,66	0,41	0,43	0,93	0,78	0,57	0,39	**0,41**	0,92
	Stacking	12366,60	0,79	0,66	0,37	0,39	0,93	0,78	0,61	0,36	0,37	0,91
MinMaxScaler (with KMeansSMOTE)	Logistic Regression	6416,76	0,71	0,70	0,71	0,70	0,95	0,65	0,34	0,49	0,36	0,89
	K-Nearest Neighbors	0,30	0,75	0,78	0,75	0,75	0,95	0,72	0,39	0,43	**0,40**	0,77
	Neural Networks | MLP	1487,27	0,76	0,75	0,76	0,75	0,96	0,66	0,35	0,48	0,37	0,89
	Bagging	3945,65	0,70	0,70	0,70	0,70	0,95	0,65	0,34	0,51	0,36	0,89
	CatBoost	38,34	0,79	0,79	0,79	0,79	0,97	0,68	0,36	0,48	0,38	0,91
	Stacking	7682,26	0,84	0,84	0,84	0,84	0,98	0,69	0,37	0,51	0,39	0,89

***Missing KKN with K-Means SMOTE w/o Scaling**

This model was attempted; however, due to the nature of KNN relying on distance calculations and the data not being scaled, the process became computationally intensive. After running for over 9 hours without producing results, it was decided to discontinue this experiment. The lack of scaling caused significant inefficiencies in the algorithm, making it impractical for this analysis. Consequently, it was decided to ignore this approach and focus on scaled versions of the data for more feasible and reliable outcomes for this algorithm

APPENDIX H. HYPER-PARAMETER GRIDSEACRH

Table H1 - Hyper-parameters used on *GridSearch*.

Estimator for *GridSearch*	Hyper-parameter	Tested Parameters	Best Parameter
Random Forest	n_estimators	[100, 200, 500, 1000]	200
	max_depth	[10, 20, 30, 40]	40
	min_samples_leaf	[1, 2, 3, 4, 5]	2
	criterion	['gini', 'entropy', 'log_loss']	'gini'
	max_features	['sqrt', 'log2']	'sqrt'
	class_weight	['balanced', 'balanced_subsample', None]	'balanced_subsample'
CatBoost	depth	[10, 15, 20]	-
	iterations	[500, 1000, 2000]	-
	l2_leaf_reg	[0, 1, 2]	-
	bagging_temperature	[0, 1, 2]	-

APPENDIX I. OPEN-ENDED SECTION | WEBAPPLICATION

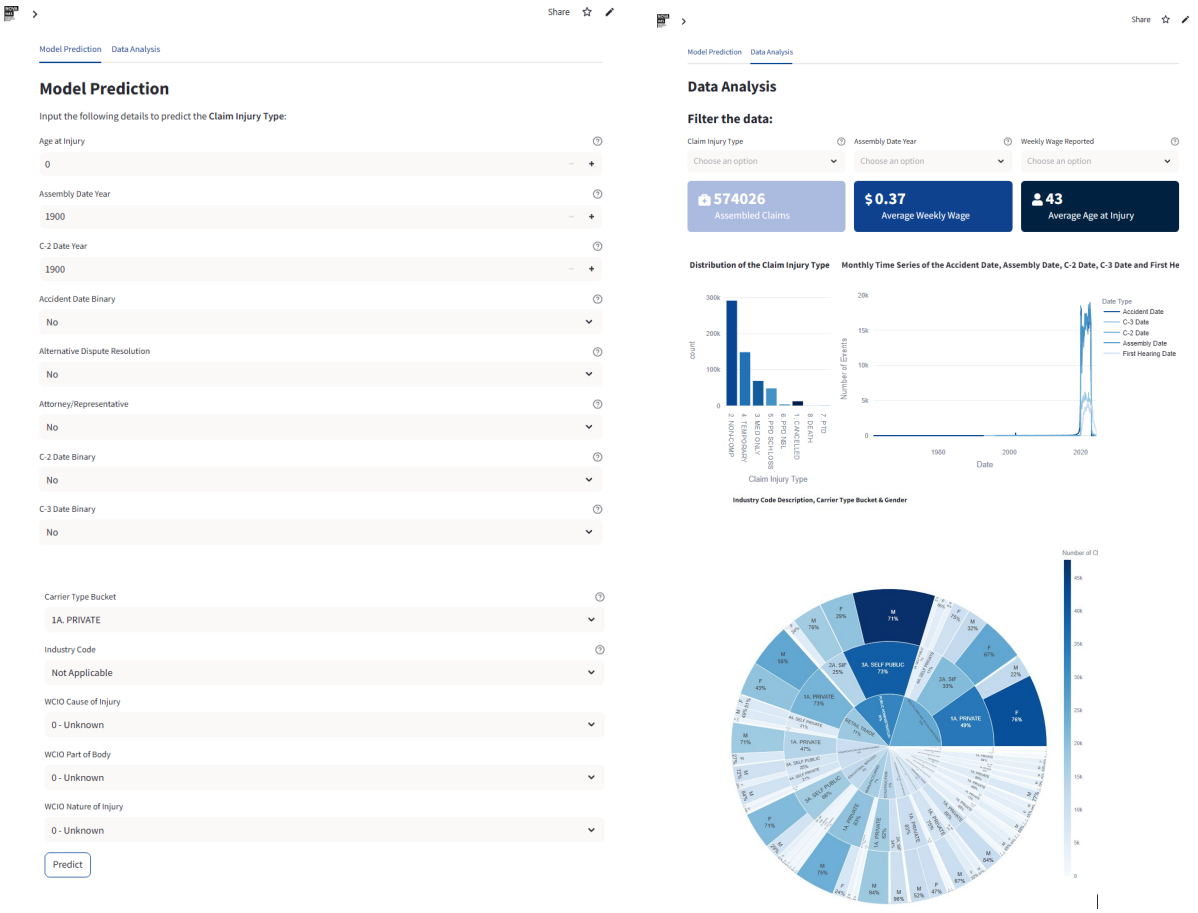


Figure I1 – Examples of parts that can be observed in the Web Application developed in *Streamlit*

APPENDIX J. OPEN-ENDED SECTION | LIME

Figure J1 – LIME result of observation 100

ANNEX A. CRISP-DM

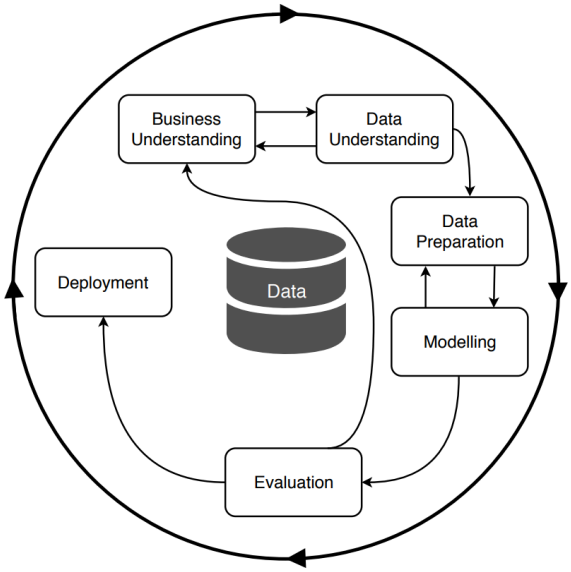


Figure A1 – CRISP-DM Methodology Cycle.
[Adapted from (Martinez-Plumed et al., 2021)]

The **CRISP-DM** (Cross Industry Standard Process for Data Mining) methodology is a widely accepted framework for conducting ML/Data Mining projects. This methodology consists of six iterative phases, each addressing critical aspects of a project and ensuring its success (Schröer et al., 2021) (Provost & Fawcett, 2013).

Business Understanding: This phase focuses on defining the project objectives and requirements from a business perspective. It includes determining the project goals, specifying success criteria and developing a comprehensive project plan.

Data Understanding: In this phase, data is collected, explored, and analysed to evaluate its quality and suitability for the project. Techniques such as descriptive statistics and visualizations are commonly used to gain insights into the data and its attributes.

Data Preparation: This phase involves selecting, cleaning, and transforming the data to create the final pre-processed dataset for modelling. Tasks such as handling missing values, outliers, and feature creation are critical for ensuring data quality.

Modelling: During this phase, the appropriate modelling techniques are selected and applied. This includes building, testing, and optimizing models. The choice of technique depends on the business problem and the data characteristics, with parameter tuning often playing an important role.

Evaluation: The evaluation phase compares model results against the defined business objectives to ensure they meet the project goals. Metrics such as accuracy and F1-scores are used to assess performance.

Deployment: The deployment phase involves integrating the developed model into an real-world application. This could include generating reports or deploying software components in a end-to-end application. Monitoring and maintaining the deployed solution is essential for its long-term success.

ANNEX B. CRAMER'S V

Cramer's V Correlation

Cramér's V, introduced by Cramér (1946), is a statistic used in the chi-square test for independence. It measures the strength of association between categorical variables. While it is primarily designed for the chi-square test for independence, it can also be adapted for the goodness-of-fit test (Kelley & Preacher, 2012, p. 145; Mangiafico, 2016, p. 474).

$$V_{gof} = \sqrt{\frac{\chi^2}{n \times df}} = \sqrt{\frac{\chi^2/n}{min(k-1, r-1)}}$$

Where:

- χ^2 is the chi-square value of the chi-square test
- n is the number of data points
- k is the number of columns
- r is the number of rows
- df is the degrees of freedom, which is the number of categories, minus one in for this

Cohen shown that Cramér's V can be converted to Cohen's w using (Cohen, 1988, p. 223):

$$w = V_{gof} \times \sqrt{df}$$

Interpretation of Cramer's V

- Cramer's V varies between 0 and 1.
  - If it is 0 there is no association between the variables,
  - If it is 1 there is a perfect association between the variables.

We could then use Cohen's rule-of-thumb for the interpretation (Cohen, 1988, pp. 224-225):

Cramer's V value (w)	Interpretation
$0 < w < 0.1$	Negligible
$0.1 < w < 0.3$	Small
$0.3 < w < 0.5$	Medium
$0.5 < w$	Large

ANNEX C. VIF

Variance Inflation Factor (VIF) is a statistical metric used to detect and quantify the severity of multicollinearity in a set of predictor variables within a multiple regression model. (Snee, 1981; Rawlings et al., 1998) In simpler terms, VIF quantifies the impact of multicollinearity—the situation where two or more independent variables are highly correlated—on the stability and interpretability of models.

The formula for VIF is:

$$\boldsymbol{VIF} = \frac{1}{1 - R^2}$$

Where:

- R^2 is the coefficient of determination obtained when regressing the variable of interest against all other predictor variables.

VIF values are always greater than or equal to 1.

Interpretation of VIF

- **VIF = 1**: Indicates no correlation between the predictor variable and the other predictors. There is no multicollinearity.
- **VIF > 1:** Indicates that the predictor variable is correlated with other variables in the model.

VIF Threshold

- **Threshold of 5:** A widely accepted threshold for moderate multicollinearity, suggested in practical regression diagnostics. It indicates when multicollinearity begins to interfere with model interpretation but may not yet be critical. (Menard, 2002) (James et al., 2013)
- **Threshold of 10:** Popularized as a strict cutoff by several applied regression texts and practitioners. When VIF exceeds **10**, multicollinearity is generally considered problematic. (Menard, 2002) (James et al., 2013)

ANNEX D. ROBUST SCALER

The **Robust Scaler** is a preprocessing method that scales features by removing the median and scaling them according to the interquartile range (IQR). (Aggarwal, 2017) (Scikit-Learn, n.d.)

The formula for the **Robust Scaler** transformation is:

$$x' = \frac{x - Q_2(x)}{Q_3(x) - Q_1(x)}$$

Where:

- x' is the scaled value
- x is the original feature
- $Q_2(x)$ is the median (50th percentile).
- $Q_1(x)$ and $Q_3(x)$ are the 25th and 75th percentiles, respectively.
- $Q_3(x) - Q_1(x)$ is the interquartile range (IQR).

Comparison with *MinMax Scaler* and *Standard Scaler*

	Central Tendency Statistic	Scaling Statistic	Sensitivity to Outliers
Standard Scaler	Mean	Standard Deviation	High
MinMax Scaler	Minimum	Range $(max - min)$	High
Robust Scaler	Median	Interquartile Range	Low

Key Advantages of *RobustScaler*:

- **Robust to Outliers:** Unlike *StandardScaler* and *MinMaxScaler*, which can be heavily influenced by extreme values, RobustScaler focuses on the median and IQR, reducing outlier impact.
- **Preserves Data Structure:** It maintains relative feature relationships while mitigating distortion caused by outliers.
- **Effective Preprocessing for Skewed Data:** *RobustScaler* is well-suited for features with skewed distributions.

ANNEX E. CATBOOST

CatBoost (Categorical Boosting) is a high-performance gradient boosting library specifically designed for tasks involving classification, regression, ranking, and other machine learning challenges. Developed by Yandex and launched in 2018, CatBoost is part of the family of **Gradient-Boosted Decision Trees (GBDT)** techniques and offers state-of-the-art performance, especially for datasets with a mix of numerical and categorical features. (Prokhorenkova et al., 2019) (Yandex, 2019)

Key Characteristics of *CatBoost*:

- **Automatic handling of categorical features:** Unlike traditional GBDT models, CatBoost internally encodes categorical variables, eliminating the need for one-hot encoding or label encoding.
- **Robustness to overfitting:** Utilizes ordered boosting and other regularization techniques to reduce overfitting.
- **Ease of use:** Requires minimal preprocessing and hyperparameter tuning for strong baseline performance.
- **Efficiency:** Optimized for fast training and prediction times.

Theoretical Foundation

CatBoost builds on the traditional **gradient boosting framework**, which incrementally builds an ensemble of weak learners (typically decision trees) to minimize a loss function. However, CatBoost introduces key innovations:

Ordered Boosting

- Traditional gradient boosting suffers from overfitting due to target leakage, as each tree in the ensemble is trained using the entire dataset.
- *CatBoost* employs **ordered boosting**, where data is split into separate subsets, and each subset trains the next model using only data that was not used in its own training. This prevents overfitting caused by target leakage.

Handling of Categorical Features

- *CatBoost* efficiently handles categorical features by employing a process called **ordered statistics encoding**, which computes category encodings based on data subsets to avoid data leakage.

Loss Function

CatBoost optimizes common loss functions used in machine learning, including:

- **Log Loss** for classification tasks.
- **RMSE (Root Mean Square Error)** for regression tasks.

Differences Between *CatBoost* and Other *GBDT Models*

The **Table E1.** outlines the main differences between *CatBoost*, *XGBoost*, and *LightGBM*.

Table E1 - *CatBoost* vs. *LightGBM* vs. *XGBoost*

Feature/Aspect	CatBoost	XGBoost	LightGBM
Release Year	2018	2014	2017
Handling of Categorical Data	Built-in; uses ordered statistics encoding	Requires manual encoding (e.g., one-hot, label)	Requires manual encoding (e.g., one-hot, label)
Target Leakage Prevention	Ordered boosting	Not explicitly addressed	Not explicitly addressed
Speed	Fast (optimized for categorical and numerical data)	Fast, but encoding adds overhead	Fastest (histogram-based approach for splitting)
Overfitting Prevention	Strong regularization, ordered boosting	Regularization techniques available	Regularization techniques available
Distributed Training	Supported Can be trained on GPUs for large datasets by setting parameter *'task type = GPU'*	Supported	Supported
Memory Usage	Moderate	Higher than LightGBM	Lower than both CatBoost and XGBoost
References	(Prok. et al., 2019) Yandex	(Chen & Guestrin, 2016) DMLC	(Ke, G., et al., 2017) Microsoft

Advantages of *CatBoost*

1. **Ease of Use:** Handles categorical data natively, requiring minimal preprocessing.
2. **Reduced Overfitting:** Innovative ordered boosting reduces target leakage and improves generalization.
3. **Performance:** Offers competitive accuracy and efficiency compared to *XGBoost* and *LightGBM*, especially on datasets with many categorical features.

ANNEX F. K-MEANS SMOTE

K-Means SMOTE is an advanced oversampling method designed to address class imbalance problems in machine learning. Proposed by Douzas, Bação, and Last (2018), this method combines the **K-Means clustering algorithm** with **Synthetic Minority Oversampling Technique (SMOTE)** to generate high-quality synthetic samples while mitigating the limitations of traditional oversampling techniques. The method aims at eliminating both between-class imbalances and within-class imbalances while at the same time avoiding the generation of noisy samples.

K-Means SMOTE operates in three main steps:

1. **Clustering:** The input dataset is divided into k clusters using the K-Means clustering algorithm. Clustering helps identify subregions in the feature space, isolating minority samples that are sparsely distributed from denser regions.
2. **Filtering:** Clusters are evaluated based on the proportion of minority class samples. Only clusters with a high proportion of minority samples are selected for oversampling. Sparse clusters with fewer minority samples are prioritized to ensure balanced coverage across the feature space.
3. **Oversampling:** SMOTE is applied within each selected cluster to generate synthetic samples. Synthetic samples are distributed across the clusters in proportion to their minority representation and sparsity.

Differences Between K-Means SMOTE and Other Oversampling Methods

The **Table F1.** outlines the main differences between *SMOTE, ADASYN* and *K-Means SMOTE.*

Table F1 - *SMOTE* vs. *ADASYN* vs. *K-Means SMOTE*

Aspect	SMOTE	ADASYN	K-Means SMOTE
Clustering	No clustering	No clustering	Uses K-Means to divide data into clusters
Focus Area	Uniformly oversamples across the feature space	Focuses on difficult-to-learn examples	Targets minority-dense and sparse regions selectively
Noise Reduction	Can create noisy samples in overlapping regions	Prone to generating noisy samples in noisy areas	Reduces noise by avoiding oversampling noisy regions
Synthetic Distribution	Balances only between-class imbalance	Balances between-class imbalance while focusing on hard examples	Balances between-class and within-class imbalances
Efficiency	Faster due to no clustering	Faster, but complexity increases with difficulty-weighting	Computationally more intensive due to clustering
Strengths	Simple and effective for balanced datasets	Better for datasets with highly complex decision boundaries	Effective for both global and local imbalances

Aspect	SMOTE	ADASYN	K-Means SMOTE
Weaknesses	Can fail on datasets with complex structures	May oversample noisy points	Increased computational cost
References	(Chawla et al., 2002)	(Haibo He et al., 2008)	(Douzas et al., 2018)

The advantages on choosing K-Means SMOTE are:

- **Targeted Oversampling:** By leveraging clustering, K-Means SMOTE can target regions of the feature space that require more representation, improving model performance.
- **Noise Reduction:** Avoids oversampling in noisy areas, minimizing the risk of creating synthetic samples that degrade model accuracy.
- **Within-Class Balance:** Addresses both between-class and within-class imbalances, leading to better generalization.