NOVA IMS
Information Management School

MACHINE LEARNING PROJECT

MASTER IN DATA SCIENCE AND ADVANCED ANALYTICS

Fall Semester 2024-2025

TO GRANT OR NOT TO GRANT: DECIDING ON COMPENSATION BENEFITS

REPORT

Group 14
André Moreira, 20222132
Andreas Follestad, 20240556
Carolina Silvestre, 20211512
Manuel Andrade, 20240571

TABLE OF CONTENTS

1. ABSTRACT

This report aims to show the readers how a student group at Nova IMS went by to develop a machine learning model aimed at automating the decision-making process for worker's compensation claims. In collaboration with the New York Worker Compensation Board (WCB), who are responsible for more than 5 million claims, we tested and tried multiple models for supervised learning to attain the highest possible accuracy on the predictions of a Test dataset. The first phase of the project was to understand the training data set, in other words do some basic exploration. A big part of the project was the pre-processing, which includes coherence checks, outlier identification, handling missing values, feature engineering and feature selection. As mentioned, we used classifiers such as K-Nearest Neighbors, support vector classifier, logistic regression, random forest, neural networks, and XGBoost. We conclude with the best result being from XGBoost, achieving a Kaggle score of 0.31619.

The notebooks used for data exploration, preprocessing, modeling, and evaluation, as well as the full reports are available on a GitHub Repository that can be visited here: TO GRANT OR NOT TO GRANT: DECIDING ON COMPENSATION BENEFITS.

2. INTRODUCTION

In an evolving digital world, companies are showing the need to elevate their analysis performance in a way that can exponentiate their results or reduce their costs. Through the project “To grant or not to grant: Deciding on compensation benefits” that the group was assigned as part of the Master's program in Data Science and Advanced Analytics at the Nova Information Management School, we plan to analyze part of the insurance sector and decision-making techniques. The focus on this project is on the New York Workers’ Compensation Board (WCB), as a regulatory authority, and how it deals with workplace injuries through claims proposed by the victims.

The main goals for this project are the understanding of the insurance sector and the US conjectures, the understanding of the given data, the creation of different models and their comparison leading to a final model and its optimization to get to predict the target variable “Claim Injury Type”. The last goal is to create a disruptive idea on how to advance after the accomplishment of the previous stated goals.

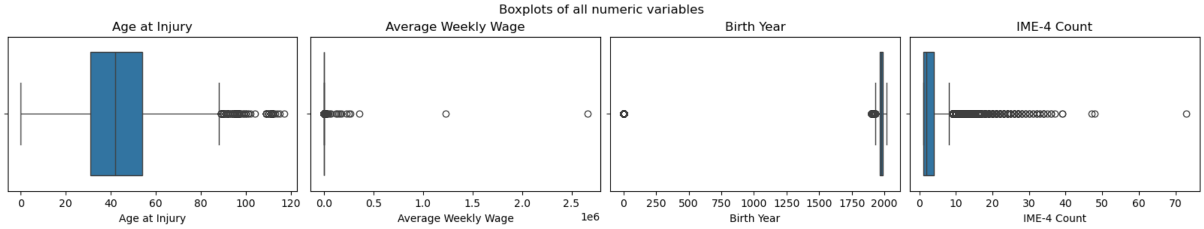
3. DATA EXPLORATION AND PREPROCESSING

3.1.1. Data exploration

Our dataset is divided into two sets: Training set which contains claims data from 2020 to 2022. It has 593471 claim records represented in 32 distinct features and a label, “Claim Injury Type” in this case. Each row should represent a uniquely proposed claim, and each column represents the characteristics of the claim, the person that suffered the injury and the injury itself; Test set with claims from 2023. This set only contained features and no label.

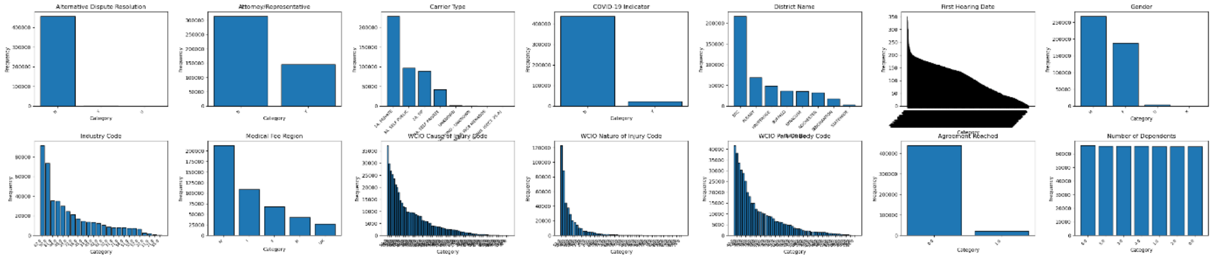
3.1.2. Numerical variables

Analyzing our numerical variables, we can identify that most of them are higly skewed and only “Age at Injury” shows a tendency to be normal. This confirms the need for further transformations on data preparation namely transform outliers



3.1.3. Categorical variables

Regarding categorical variables we can see the potential for binary variables, variables that show high cardinality and some values that must be dealt with like category “U” on “Alternative Dispute Resolution”



3.1.4. Variables correlation

As we can see on Figure 1 of the annex, we can’t identify any high correlation between numerical variables.

3.2. Data preprocessing

To apply feature selection steps and the models afterwards we had to execute data preprocessing before. The main reasons for data preprocessing are the existence of algorithms that might be intolerant to missing values, require normalization and/or need a specific handle of categorical variables such as the KNN imputer.

3.2.1. Outliers

Regarding outliers we found problems with variables that involve age like “Age at Injury” and “Birth Year” and for that reason we decided to apply the same bound on both. We only considered the records that have values between 14 and 88 years old admitting that it would be unrealistic to have a worker out of this age range. On figure 2 of the annex, we can check the transformed “Age at Injury” variable and it’s approximated normal distribution.

3.2.2. Missing Values

The main way to conduct the missing values filling was to use the mode as we did for 6 variables with an extra effort to maintain a better accuracy on “Medical Fee Region” as we searched for the mode of each medical fee region in each district and applied it. Then we used the mean as an imputer on 3 variables. On “Average Weekly Wage” we decide to apply an approach that would proportionately apply zeros to the percentage of zeros already there and the median to the percentage of values where the value was higher than 0. Regarding “C-2 Date”, as the dates are not in order, we decided to mitigate this via straight removal of missing value rows.

3.2.3. Encoding

We conducted encoding methods on time variables like “Assembly date” and categorical variables. Before advancing to the categorical ones, we decided to use the chi-square test to have a first insight into the importance of categorical variables on predicting the target variable and everyone shown importance. Using the one-hot encoding method on “District Name”, “County of Injury” and “Carrier Name” we ended up with 103 features. There were also categorical features that with the right encoding turned into binary features like “Attorney/Representative”.

3.2.4. Feature engineering

Regarding the creation of new features, we decided that would be relevant to have two new variables: “Assembly Date” and “Conclusion Time”. The first one is the result of the difference between features “Assembly Date” and “Accident Date” and the second one is the difference between “First Hearing Date” and “Accident Date”. Both variables were transformed into integer days. “Conclusion Time” was in the presence of missing values and we filled them with the mean.

3.2.5. Features Removal

There were some variables in this step that we chose to remove, even before feature selection, as they wouldn't any valuable information to the data set such as every description variable to code variable, as an example "Industry code" and "Industry Code Description". We also removed "WCB Decision" and variables with a high number of missing values like "C-3 Date", "IME-4 Count" and "First Hearing Date".

3.2.6. Pre feature selection

In order to advance to feature selection and then modelling we should split the training set in two different sets: X_train set and y_train set. In X_train we will present all features previously handled on preprocessing steps and y_train there will only be the target variable. To prevent data leakage, we decided to do this split before preprocessing. As changes were made in X_train and not on y_train we had to create an identifier variable "Tag" that at the end of preprocessing was used to keep on y_train the same rows as there was in X_train.

3.2.7. Scaling

To scale our sets, we used three different scaling methods: Min-Max scaler; Robust Scaler; Standard Scaler. We analyzed the three of them because on further modeling steps we will need scaling methods that offer resilience on outliers or standardize the data.

3.2.8. Validation and Test sets

Every previous modification on the training set was mirrored into validation and test sets in order to maintain equality on the handling. This ensures easier work on modelling as we create a static database from now on. After preprocessing, on the validation set we get to work with 114679 records and 387975 records on the test set.

4. MULTICLASS CLASSIFICATION

4.1. Feature selection

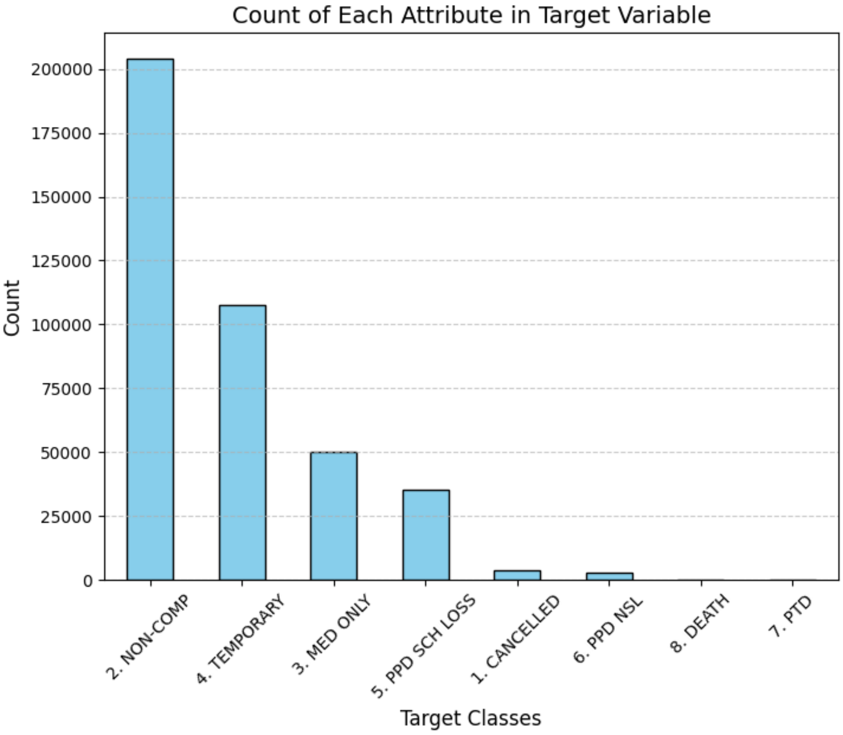
As a result of the original high number of features and then the encoding of categorical features we have reached a high-dimensional dataset that can lead us to several problems on modeling such as the curse of dimensionality, overfitting or even computational complexity. The last case would turn out to be the biggest barrier on this project not only because of high dimensionality but also the high number of records to work with.

For the previous reasons we decided to study feature relevancy through three different feature selection techniques. Regarding categorical variables we tested their independence with the chi-square methodology using also stratified k-fold to make sure we could deal with some imbalance that could be natural on a vast dataset like ours.

The second methodology we used was Lasso. On this method we needed to work with scaled data making us analyze which scaler was the best performer being it Standard scaler due to the need of standardized data. Lasso led us to a still high number of features when analyzing the ones that had a positive Lasso coefficient for alpha equal to 0.001, mostly because of counties encoding as we can see on figure 3 of the annex.

The last feature selection tool used was Recursive Feature Selection. It was tried with two different models, Logistic Regression and Support Vector Machines.

4.2. Target variable



As a last step before modeling, we will take a closer look at our target variable. It has 8 different classes, and they look unbalanced as we can see from the upper figure. For that reason, we will check each class weight on the target variable.

4.3. Modeling

The models we considered studying were the following: KNN classifier; Support Vector Classifier; Logistic Regression; Decision Trees; Random Forest; Neural Networks; XGBoost.

The KNN classifier was used on previously scaled data, specifically the min-max scaler, because, as it relies on distance metrics that we will study on the tuning part, it is sensitive to big variations on the data, and then we tried to find which were the parameters that would concede the best accuracy through a tuning phase and in this case it was 64.11%. We also used other metrics to study the model effect on classes such as precision, recall and the f1-score and that led us to find a bias on more populated classes in comparison to less populated ones.

Support Vector Classifier was utilized in a similar way to KNN. We first defined a pipeline to deal with scaling, we adopted min-max scaler again as SVC utilizes distance-based calculations, and the model itself, linearSVC considering the class weights previously studied. The next step was to find the hyperparameters that would deliver the best results. This time in terms of accuracy we were below the KNN mark as we could only achieve 58.26% and regarding the rest of the metrics, we had similar results showing a higher precision on higher numbered classes with emphasis on class 2 that has the highest precision, recall and f1-score. On a last analysis we tried to study the same metrics on the test set and the results improved for unsustainable numbers 78,24% with the rest of the metrics presenting even higher results for class 2. This is explained by only 3 classes being studied on the test set, classes 2, 4 and 5.

Logistic regression is the model that allows us to be the most efficient while retrieving outputs as we already used it on recursive feature selection. The main downside of it is its struggle to deal with high-dimensional data as we have. Looking into the results we achieved with it makes understandable the previous statements. The accuracy is not the best, 59.03%, and even higher populated classes don't show signs of acceptable precision nor f1-score.

Decision tree is the first model that doesn't need any type of scaling and it's easier to implement as it won't need any require less preprocessing steps. On the other hand, it's prone to overfit depending on how deep the trees are and will most certainly be biased toward the most dominant classes as we seen before. In terms of accuracy, it's less accurate than the models studied before, 50.63%, but as we look into the other metrics it continues the same pattern as before showing more precision, recall and f1-score on the most dominant classes.

As for random forests, we achieved the best accuracy from all previous models, 74.01%, and it might be due to a reduced overfit of data because it averages multiple decision trees. A con of working with random forest is the computational power needed to generate outputs. Analyzing the previous metrics we find a similar pattern on precision, recall and f1-score as before showing bias towards higher classes but this time f1-score improved.

Neural networks lead us to the best result when studying accuracy, with 76.73%, and it looks consistent on the other metrics analyzed being that a good indicator on the model. This high value in accuracy is due to the length of our dataset which can also lead to some overfitting.

Xgboost although it's more complex in terms of tuning for higher performance it can lead to better results as it supports different types of parameters such as regression and classification. In this case we don't have the best accuracy of all models, 66.44%, but this might be the better performer in terms of individual classes analysis. It shows an increase in precision among all classes except for 6 and 7 with zero precision.

5. OPEN-ENDED SECTION

This section of the project report is dedicated to exploring an open-ended analysis so we can have additional insights and enhance model performance. Among the suggestions provided, the chosen focus was on creating a model to predict the variable ‘Agreement Reached’ and assessing its potential as a feature to improve the performance of our primary models.

Building on the methodology applied to the variable ‘Claim Injury Type’, we followed a similar approach, where ‘Agreement Reached’ was used as the target variable for prediction, that has a binary nature (already encoded as 0 or 1).

Regarding the actions taken, we started with some data exploration and data preparation, where we split the data into train and validation (80-20), setting ‘Agreement Reached’ as the target. Then we performed data visualizations and moved on to data preprocessing. To preprocess, we changed data types, handled categorical features, removed outliers, treated missing values, dropped columns, performed feature engineering, and used one-hot encoding, as we did for ‘Claim Injury Type’. We also used the same scaling and feature selection methods as before and trained the same models, analysing them using ROC curves.

Among the trained models, the Support Vector Classifier (SVC) stood out as the best model, primarily due to it showcasing the highest Area Under the Curve (AUC) score out of all models. The AUC of 0.8732 demonstrates the model’s capability to distinguish between the positive and negative classes effectively, making it the most reliable model for predicting the ‘Agreement Reached’ variable. Additionally, the high AUC reflects the model’s robustness in balancing the true positive and false positive rates, reinforcing its selection as the best-performing model.

By predicting this variable and using it as an additional feature, we could evaluate its impact on the predictive accuracy of our primary models. Nevertheless, we also have in mind that the success of this approach would depend on the accuracy of the predictions and its correlation with the target variable of the main models.

6. CONCLUSION

This project aimed to address the challenge of predicting "Claim Injury Type" within the context of the New York Workers’ Compensation Board (WCB), leveraging data-driven decision-making techniques to optimize performance and provide actionable insights. Alongside the primary objective, we explored additional avenues, including predicting the variable ‘Agreement Reached’.

The initial objectives included analyzing the dataset, preprocessing and engineering features, building predictive models, and optimizing their performance. Each of these objectives was successfully accomplished. The data exploration revealed important characteristics, such as skewness in numerical features and imbalances in categorical ones, guiding the transformations and preprocessing steps required for effective modeling. Employing advanced feature selection techniques like Lasso helped us mitigate dimensionality issues, improving computational efficiency and reducing the risk of overfitting. Several models were trained and evaluated, with Neural Networks achieving the highest accuracy (76.73%) and Random Forest offering competitive performance with added robustness. The prediction of ‘Agreement Reached’ and its use as a feature demonstrated the potential for leveraging related variables to improve model accuracy, though the improvement was contingent on the quality and correlation of the predicted variable with the primary target. So, we can say that the findings aligned well with our initial expectations.

Some limitations that emerged during the project were related to the high dimensionality, the imbalanced target classes and the models complexity, that increase the risk of overfitting if not well conducted.

If we followed on our work some suggestions would be using SMOTE (Synthetic Minority Oversampling Technique) to better address target class imbalances, using ensemble methods that combine multiple models through stacking or blending to leverage the strengths of different approaches and achieve improved performance or add external data sources, such as economic or geographic indicators, to enrich the feature space and provide additional context for predictions.

In summary, this project has demonstrated the power of data-driven methods in addressing complex decision-making problems within the insurance sector. While the results meet our expectations, the identified limitations and opportunities show a clear path for future work to improve and build on the insights gained.

ANNEXES

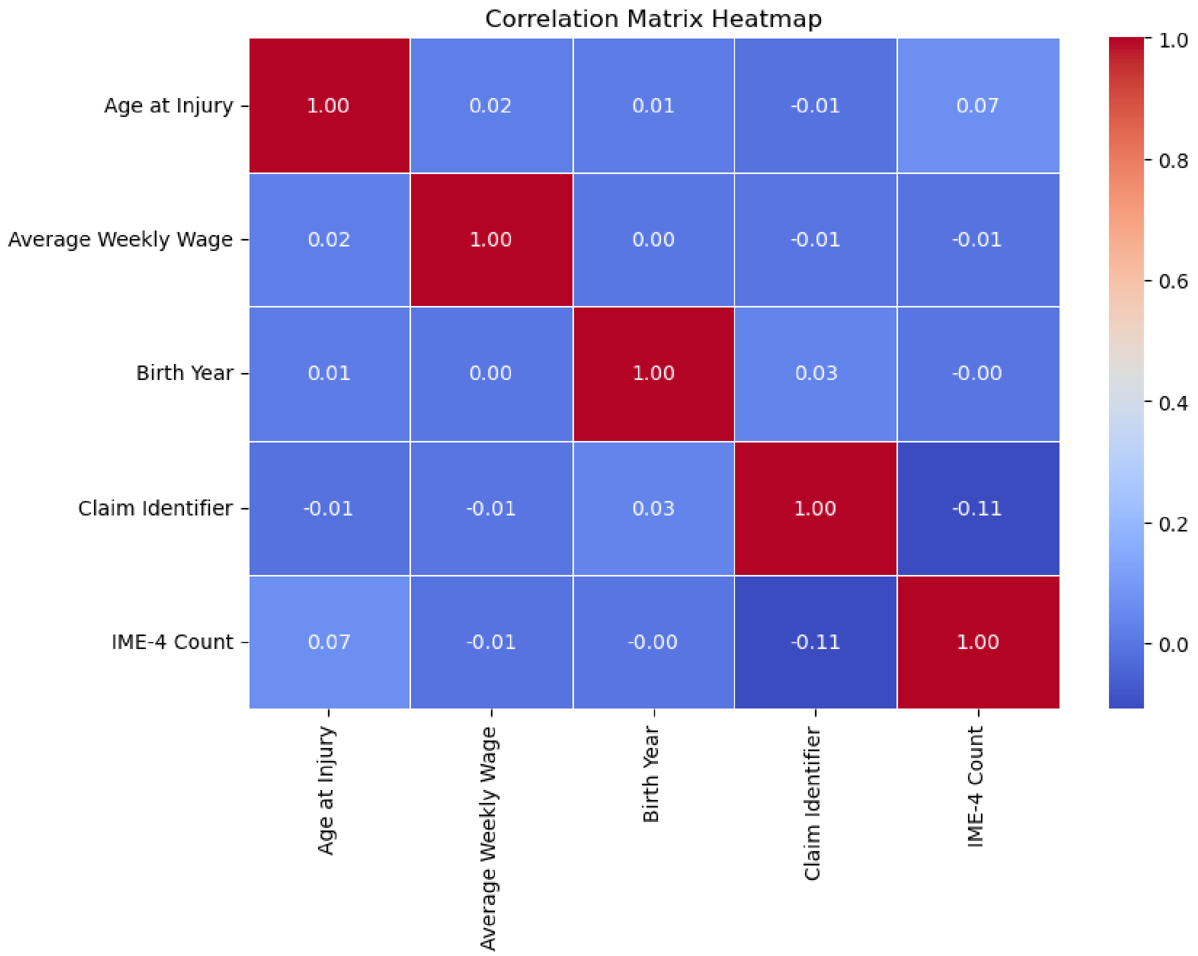


Figure 1 - Correlation Matrix

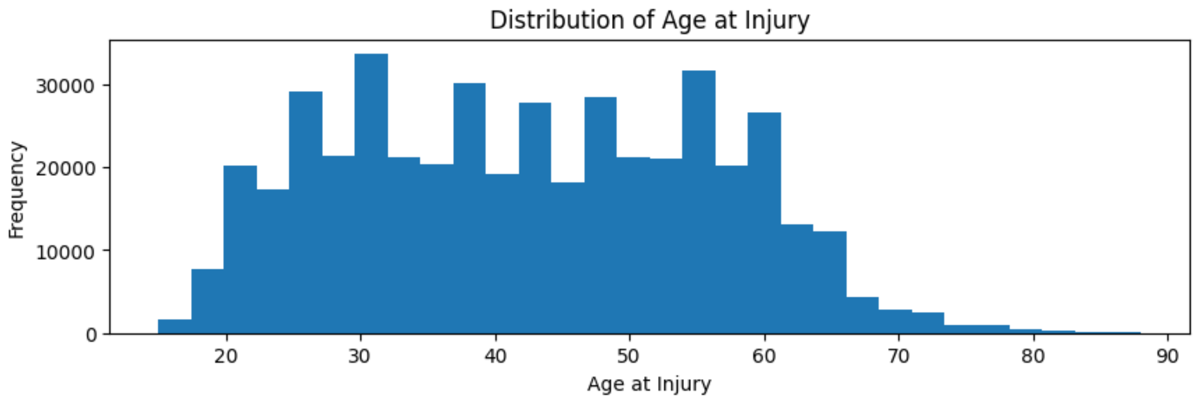


Figure 2 - Transformed Age at Injury

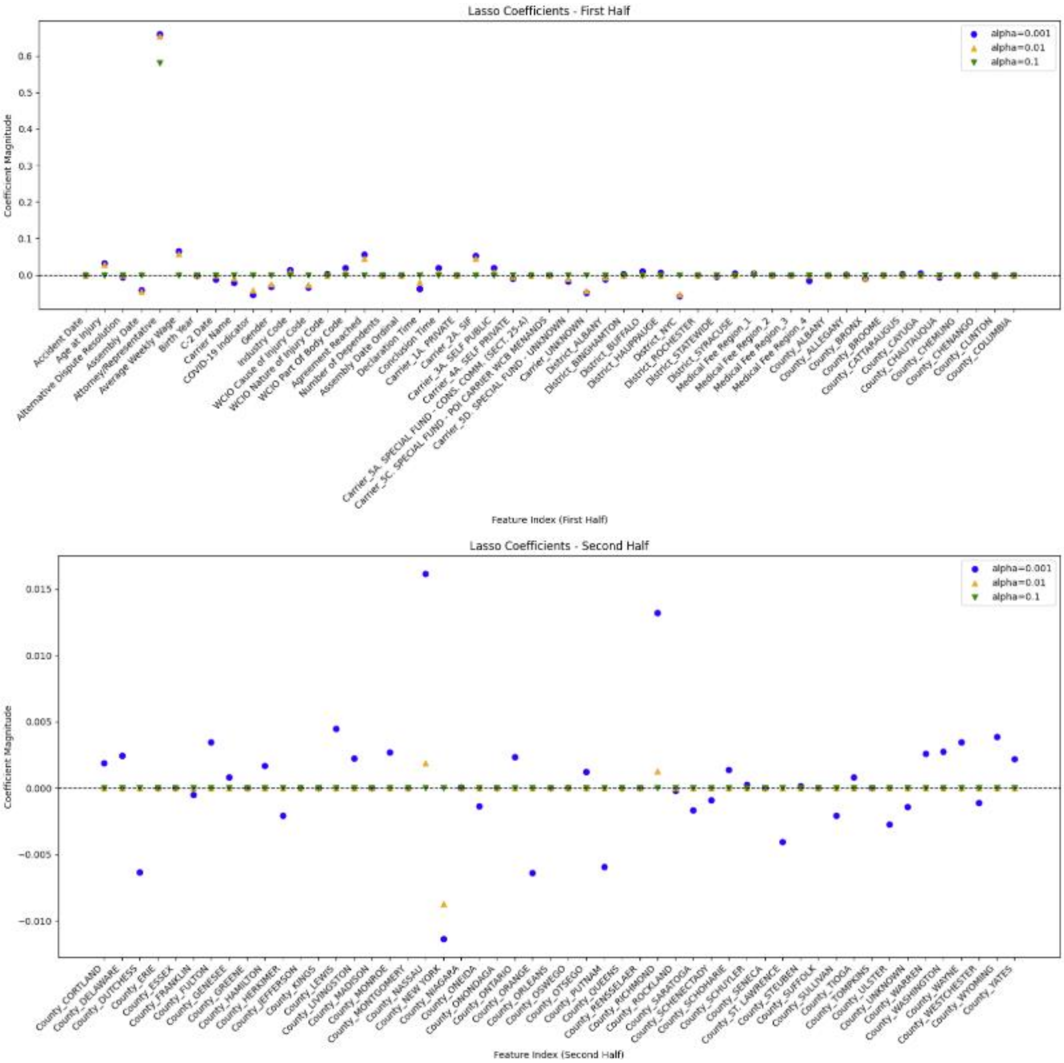


Figure 3 - Lasso coefficients (predicting ‘Claim Injury Type’)

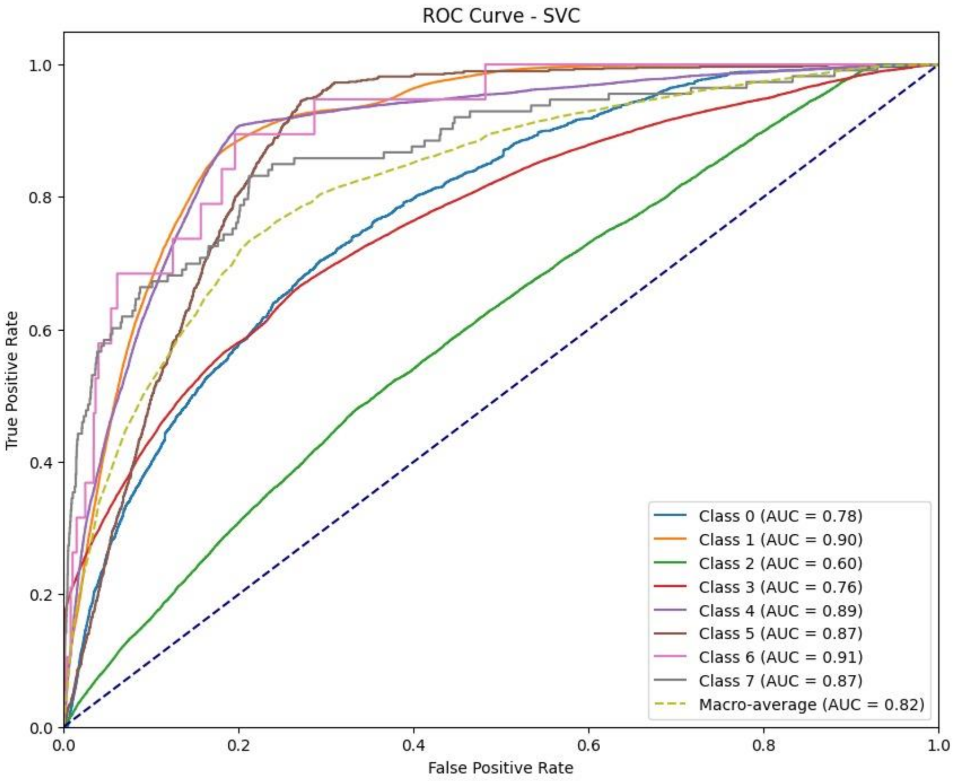


Figure 4 – ROC Curve: SVC (predicting 'Claim Injury Type')

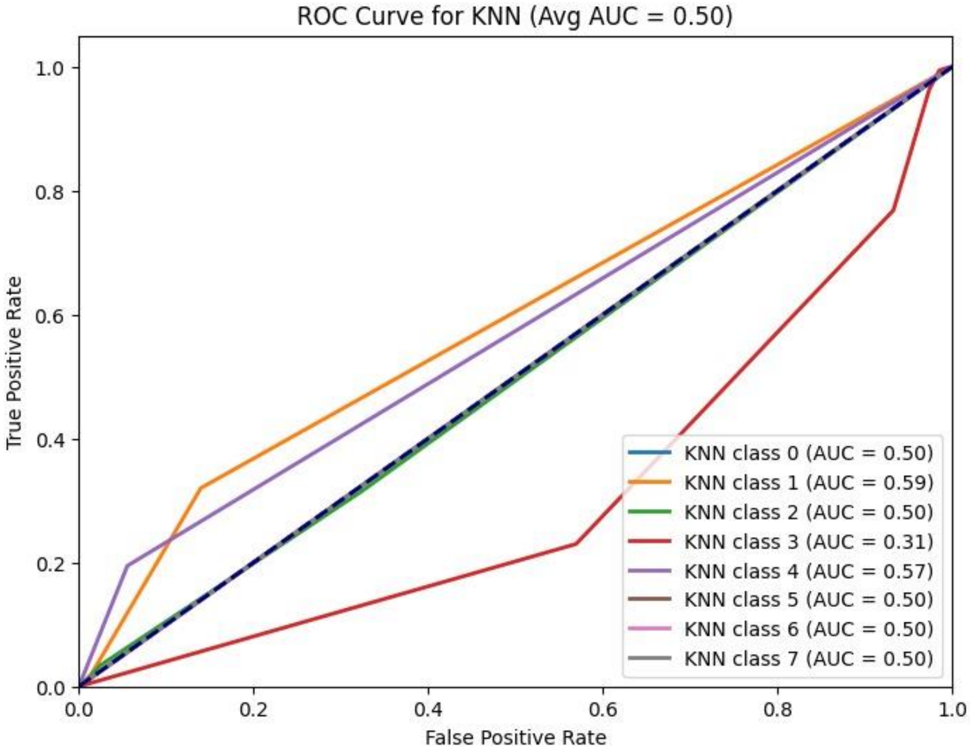


Figure 5 – ROC Curve: KNN (predicting 'Claim Injury Type')

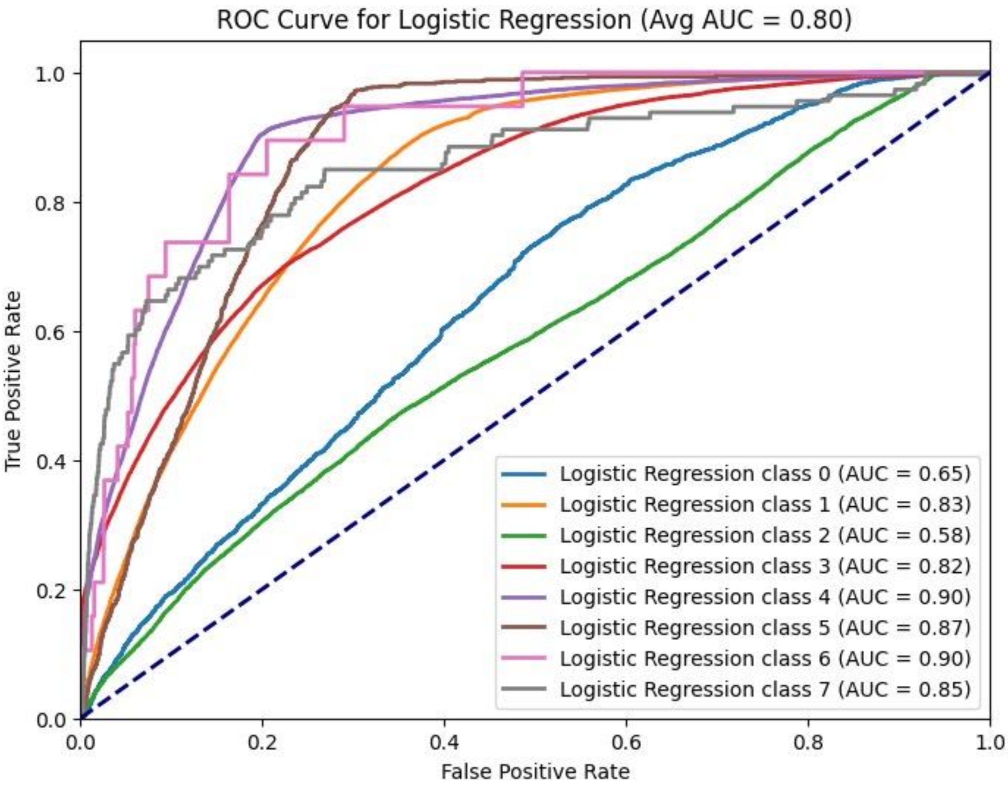


Figure 6 – ROC Curve: Logistic Regression (predicting 'Claim Injury Type')

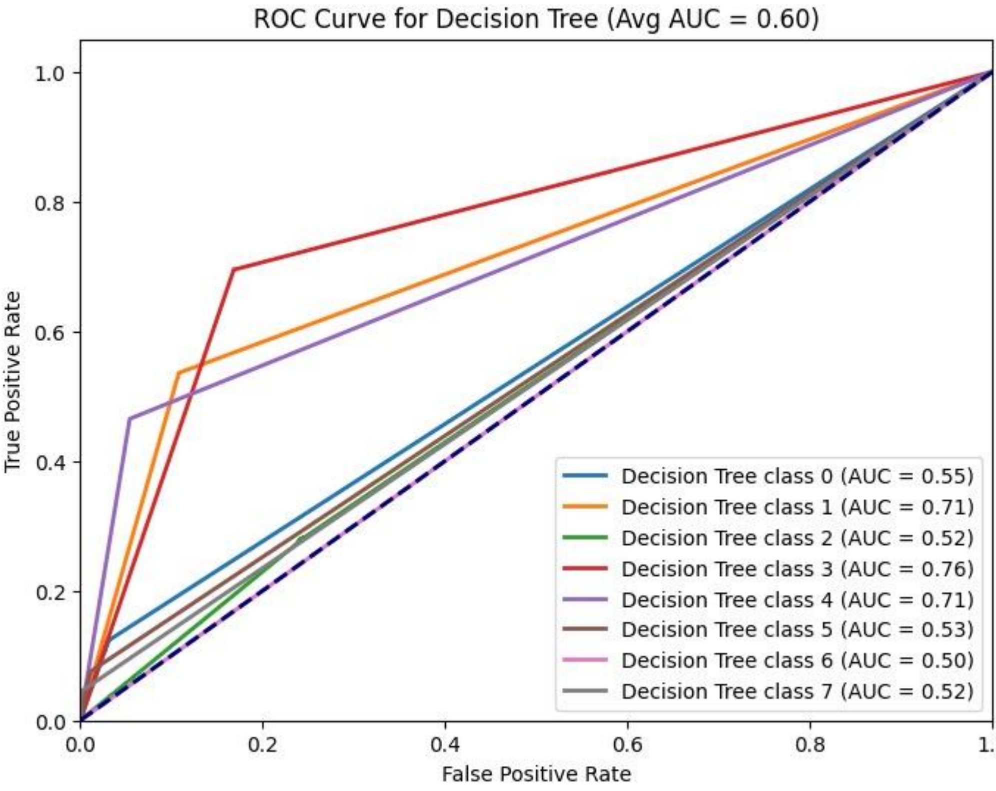


Figure 7 – ROC Curve: Decision Tree (predicting 'Claim Injury Type')

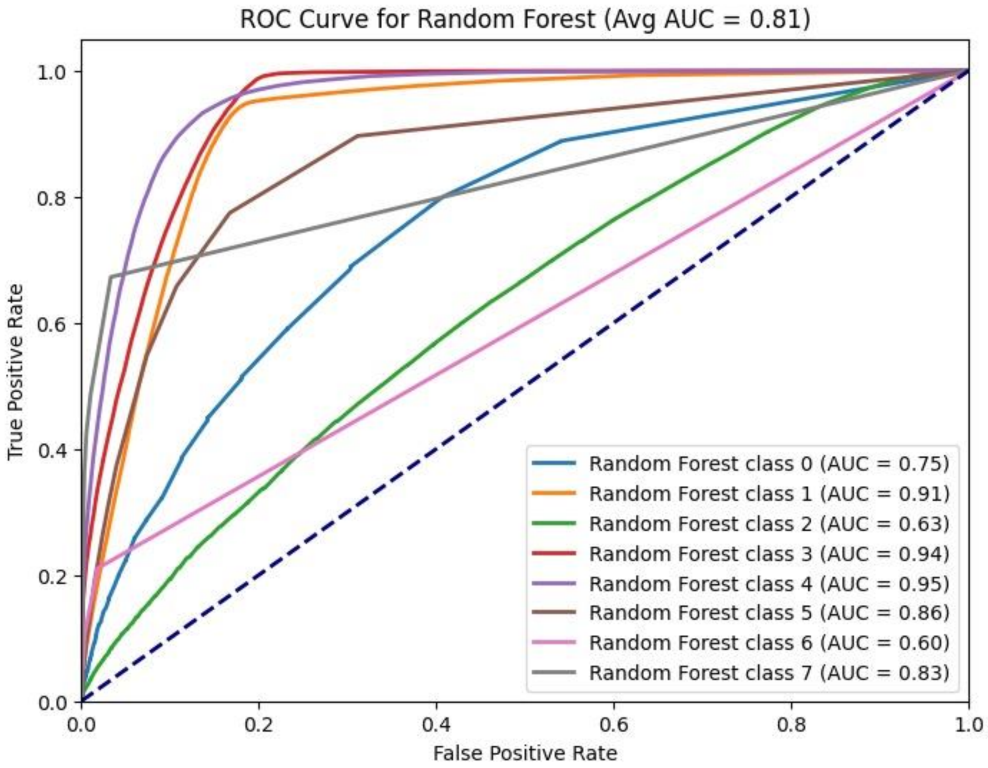


Figure 8 – ROC Curve: Random Forest (predicting 'Claim Injury Type')

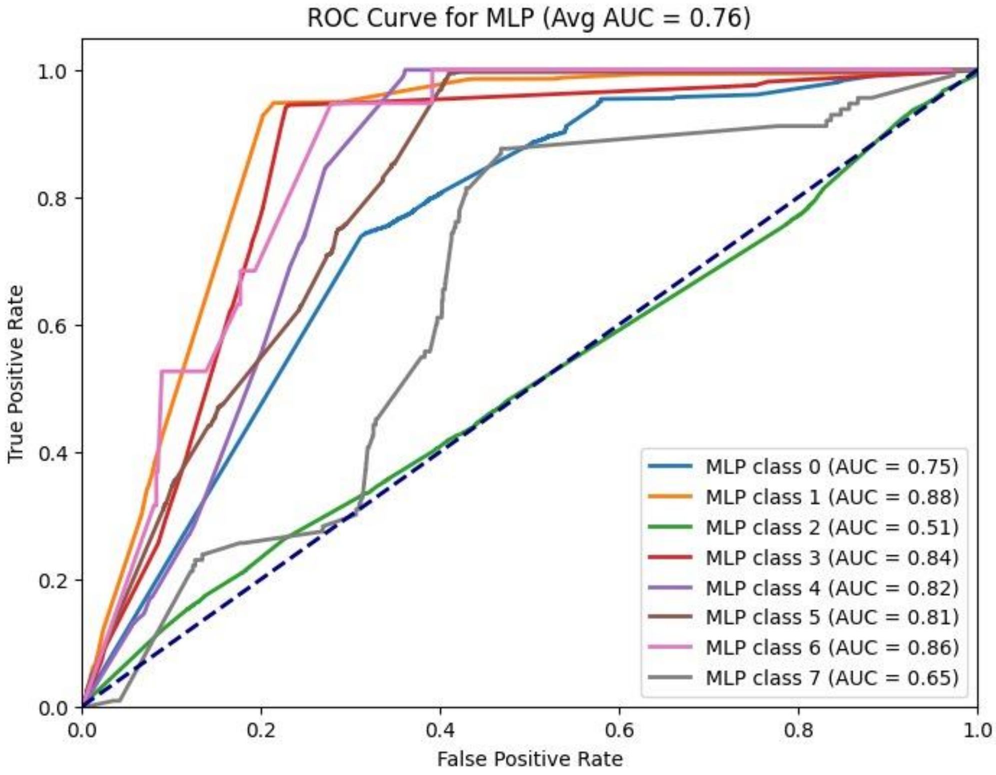


Figure 9 – ROC Curve: MLP (predicting 'Claim Injury Type')

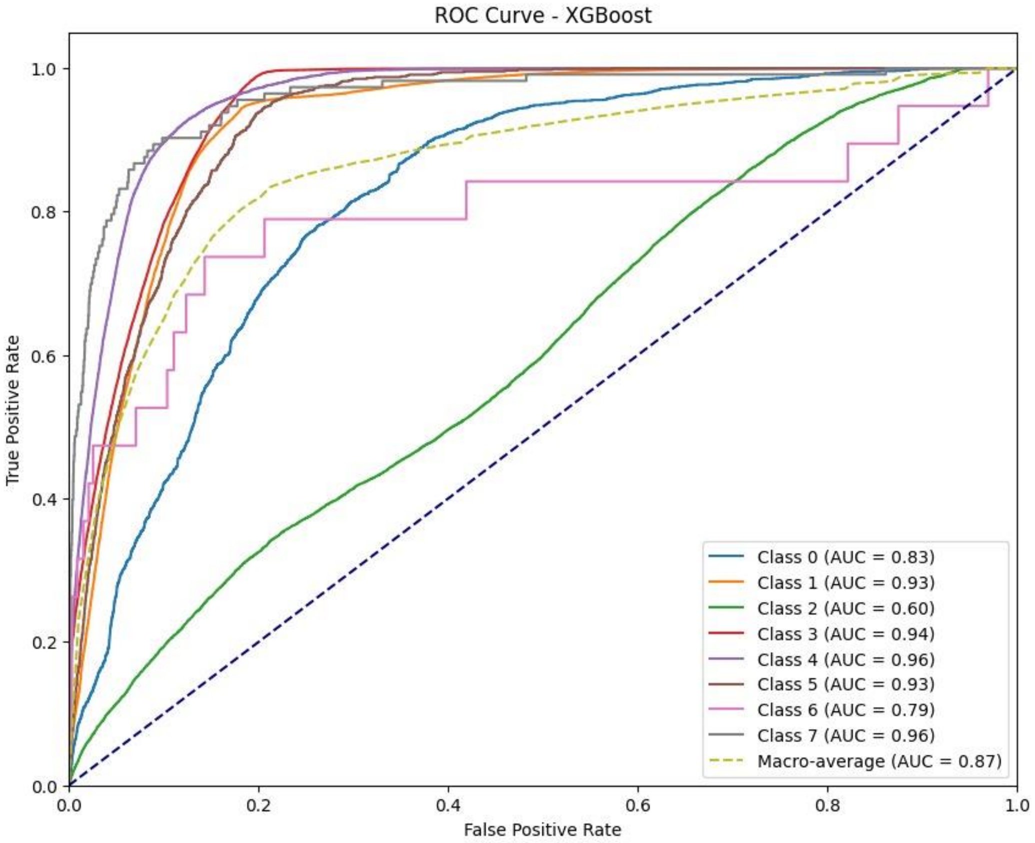


Figure 10 – ROC Curve: XGBoost (predicting 'Claim Injury Type')