20240536 Inês Jacinto 20240536@novaims.unl.pt

20211639 Rui Lourenço 20211639@novaims.unl.pt

20240561 Antônio Ramos 20240561@novaims.unl.pt

20240662 Chiel Groeneveld 20240662@novaims.unl.pt

20240598 Sofia Jacinto 20240598@novaims.unl.pt

20240750 Marisa Marques 20240750@novaims.unl.pt

1. Introduction

The New York Workers' Compensation Board (WCB) project aims to automate the decision-making process for workplace injury claims. This analysis focuses on claims data assembled between 2020 and 2022, with the primary goal of developing a classification model to predict claim injury types. For that matter we will need to go through four phases: exploratory data analysis, clean and pre-process the dataset, feature selection, and finally build a model and assess its performance.

2. Data Exploration and Preprocessing

The main goal in this phase is to understand the distribution and the structure of the dataset and improve data quality before moving on to the model. We used the function info() for feature analysis, the function describe() for a descriptive statistical analysis, histograms to visualize numerical variables and plots to visualize the distributions of the categorical variables. We concluded that there are 33 features with ~574,026 records (entries), being 'Claim Injury Type' the target variable. Notable features include temporal data (Accident Date, Assembly Date), demographics (Age, Gender, Birth Year), injury details (WCIO codes for cause, nature, and body part), financial information (Average Weekly Wage) and administrative data (Attorney representation, IME-4 Count). This step is crucial to do the other ones once it gives us a deep understanding of our dataset and helps identify potential issues early.

For the temporal features (Accident Date, Assembly Date, C-2 Date, C-3 Date, First Hearing Date) the focus was mostly on reducing cardinality of variables. For that reason, temporal features created from date columns were derived:

- Month: *_Month (e.g., Accident Date_Month)
- Quarter: *_Quarter (e.g., Accident Date_Quarter)
- Year: *_Year (e.g., Accident Date_Year)

Several categorical variables were grouped to handle rare categories (occurring in less than 1% of records):

- Carrier_Category: Grouped rare carriers into 'OTHER_CARRIERS'
- County_Grouped: Grouped rare counties into 'OTHER_COUNTIES'
- Industry_Grouped: Grouped into major sectors like:
  - HEALTH CARE AND SOCIAL ASSISTANCE (19.9%)
  - PUBLIC ADMINISTRATION (16.1%)
  - RETAIL TRADE (10.7%)
  - TRANSPORTATION AND WAREHOUSING (9.4%)
  - EDUCATIONAL SERVICES (7.7%)

Injury Classification Grouping created grouped versions of injury classifications:

- WCIO Cause of Injury Description_Grouped
- WCIO Nature of Injury Description_Grouped

- WCIO Part Of Body Description_Grouped

Geographic Features:

- Zip_Region: Created from first 3 digits of ZIP code to group by geographic regions

The methods to clean categorical variables were:

- Gender_Clean: standardized gender categories (M, F, UNDEFINED)
- ADR_Clean: cleaned alternative dispute resolution values (N, Y)

For the numerical data cleaning it was applied restrictions to ensure data quality:

- Age at Injury: limited to 16-85 years, statistical support: 99% of data fell within this range (568,057 valid values from 574,026)
- Birth Year: based on the formula: current_year(2024) - age_limits
- IME-4 Count: limited to 1-15, IQR analysis showed most values between 1-4 (25th-75th percentile)
- WCIO Body Codes: limited to valid range 1-99, based on standard WCIO classification system

This last two steps helped us remove outliers and handled the missing values so we can go to the next phase.

3. Multiclass Classification

Feature selection is very important so we can remove irrelevant or redundant variables. In our dataset, a combination of XGBoost's tree-based feature selection and Principal Component Analysis (PCA) is applied. XGBoost's feature selection identifies important features through tree-splitting, while PCA reduces dimensionality. PCA is selected to minimize collinearity, capture 95% of the variance and retain meaningful patterns with a lower computational burden. Class imbalance can lead to a poor model performance so to address that issue, the Synthetic Minority Over-sampling Technique (SMOTE) was applied. Feature selection will be performed using F1-Score Optimization for imbalanced and multi-class data.

4. Open-ended section

Our open-ended section focuses on systematic feature engineering through XGBoost optimization. The methodology starts with the base XGBoost classifier optimized through k-fold gridsearch with macro F1-score as the loss function. By exploring permutations of the feature space through PCA decomposition, we identify eigenvalues that serve as quantitative measures of feature importance through explained variance. Setting a 95% variance preservation threshold allows us to significantly reduce dimensionality while maintaining predictive power. The implementation systematically tracks F1-score changes as we manipulate the feature space, ensuring that model performance isn't significantly degraded. The pipeline maintains reproducibility through consistent cross-validation strategies, with each feature selection decision validated against its impact on the macro F1-score. Neural network architectures are explored as an alternative to XGBoost, with hyperparameter optimization adjusted to focus on architectural variations rather than traditional parameter tuning, though still maintaining F1-score as the primary optimization metric.

5. Conclusion

We only got a 0,369 F1-score which means our model will need a lot of tuning, such as adjustment of modelling approach or further exploration of data preprocessing, due to the complexity of the training and testing dataset.