CS 699 Data Mining Final Project Report On:

"Practice of building and testing classification models using a real-world dataset - National Crime Victimization Survey: School Crime Supplement, 2013"

Prepared By:
Tejas Yogesh Pawar
(U 42297790)

Under the supervision and guidance of:
Ast. Prof. Dr. Jae Young Lee
BU - MET Computer Science Coordinator

(Signature)

Submitted On:
8th November 2023

• Table of Contents

1. Introduction

- **Purpose of the project**

The principal goal of this project is to precisely identify bullying victims by applying data mining techniques to create a variety of classification models. This project revolves around data mining, which offers the techniques and tools required to identify patterns in large, complicated datasets. In this instance, data mining is used to inform and direct practical solutions against school bullying, in addition to providing theoretical insights.
Our project aims to accomplish the following technological objectives and is methodologically based on data science principles:

- **Data Preparation:** Perform pre-processing operations on the dataset, such as normalisation, encoding, and treatment of class imbalances, to guarantee high-quality inputs for modelling.

- **Feature Selection:** Using methods such as statistical tests and attribute importance ranking, determine and pick the most informative features from the dataset that enhance the models' ability to predict future events.

- **Model Exploration:** Assess a variety of classification algorithms to see which ones work best for our dataset and problem environment. These algorithms range from more conventional methods like Decision Trees to more advanced ones like Support Vector Machines and Neural Networks. Model Selection and Optimisation: To improve the classification performance of the most promising models, fine-tune and optimise their parameters while keeping a close eye on the ratio of model complexity to predicted accuracy.

- **Validation and Testing:** Make sure the final model chosen works well not only on our data but also on data that can be reasonably predicted to perform well in the future by thoroughly testing the models to evaluate their robustness and generalizability.

The best-performing model will be identified and optimised as the project's completion.

- **Description of the dataset source (2013 school crime supplement of NCVS)**

Our data mining effort is empirically based on the 2013 School Crime Supplement of the National Crime Victimisation Survey (NCVS). This dataset, which consists of 4,947 observations with 204 numerical attributes each, offers a thorough understanding of school-related events, including demographic information, individual opinions, and reported incidences that may be associated with bullying victimisation.

This dataset presents a major problem in that it shows a large class disparity, with a heavy skew towards non-victim classes. If these imbalances are not properly addressed during the pre-processing stage, biased models may result. In addition, even with no missing values, the dataset's large dimensionality demands a thorough feature selection procedure in order to lessen the dimensionality curse and enhance the interpretability and performance of the model.

In the following sections, we will systematically unfold the steps taken to transform this raw data into actionable intelligence, demonstrating the pivotal role of data mining in solving real-world problems through the lens of our bullying classification task.

2. Data Mining Tools Used

I used R, an efficient programming language for statistical computation and graphics, as the main piece of software for this project in order to perform the data mining activities. The R scripts and commands were run in RStudio, an integrated development environment that improves the R user interface and facilitates data analysis, debugging, and coding in a more effective and efficient manner.

Weka is a Java-implemented collection of machine learning techniques for data mining activities that was used for exploratory data analysis (EDA). Weka's visual data exploration capabilities and ability to provide a basic grasp of the structure and properties of the dataset were quite useful.

In the course of this project, a number of R packages were utilized, each serving a specific purpose in the data mining process:

caret (Classification And REgression Training): This package was instrumental in streamlining the model training process by providing a unified interface for creating predictive models and pre-processing data. It offers extensive functions for data splitting, pre-processing, feature selection, model tuning, and performance assessment.

DMwR (Data Mining with R): Used primarily for its functions that assist with the handling of imbalanced classes, a challenge explicitly presented by our dataset. It also provides tools for smoothing data and other useful utilities for data mining.

tidyverse: A collection of R packages designed for data science, which makes it easy to import, tidy, transform, and visualize data. It includes ggplot2 for data visualization, dplyr for data manipulation, and purrr for functional programming, among others.

dplyr: A part of the tidyverse, it is particularly focused on providing a grammar for data manipulation and is used for filtering rows, selecting columns, and arranging data.

randomForest: This package implements the Random Forest algorithm for classification and regression tasks. It is known for its robustness and ability to handle a large number of input variables.

glmnet: Implements generalized linear and elastic-net regularized models. It is particularly useful when dealing with models that require regularization to prevent overfitting.

rpart: Provides functionality for recursive partitioning and regression trees, allowing us to fit decision tree models to our data.

gbm (Generalized Boosted Models): Used for fitting generalized boosted regression models which are an effective tool in improving model's predictive strength.

e1071: This package is used for fitting Support Vector Machines (SVMs), a powerful class of supervised learning algorithms for classification and regression.

class: Contains functions for k-nearest neighbor (k-NN) algorithm which is a simple yet effective technique for classification tasks.

naivebayes: Offers an implementation of the Naive Bayes classifier, an algorithm based on applying Bayes' theorem with strong independence assumptions between the features.

nnet: Used for fitting neural networks, which are particularly powerful in capturing complex patterns in data.

Through the strategic application of these tools and libraries, we were able to conduct an in-depth analysis of the dataset, pre-process the data effectively, train various classification models, and evaluate their performance with a high degree of accuracy and efficiency..

3. Data Pre-processing

In the initial phase of data preprocessing for this data mining project, I performed a series of steps to refine the dataset comprising 4,947 instances and 204 attributes. This process was crucial for ensuring the quality and relevance of the data before moving on to the modeling phase.

Summary of Pre-processing Steps

- **Removal of Zero-variance Features**: I used the nearZeroVar function in R to remove features that had the same value in all instances as they do not contribute any information for model discrimination.
- **Elimination of Low-variance Features**: I identified and removed attributes with very low variance using R's var function along with a threshold criterion because they are less likely to significantly influence the target variable.
- **Dealing with Highly Correlated Features**: With the findCorrelation function from the caret package in R, I eliminated features that were highly correlated with each other to reduce redundancy and mitigate the multicollinearity problem, which can affect model performance.
- **Correlation with Target Variable**: I excluded features with minimal correlation to the target variable by computing correlation metrics using the cor function in R, ensuring the dataset focused on more predictive attributes.
- **Re-evaluation of Features**: I carefully reviewed the remaining features to ensure no critical predictors were erroneously excluded during the initial steps.

Feature Selection Algorithms

I explored several feature selection methods and ultimately utilized ANOVA F-tests and LASSO as they provided the best results:

- **ANOVA F-test**
  - **Purpose**: The ANOVA F-test, performed using the aov function in R, helps to evaluate if there are statistically significant differences between the means of various groups, which is valuable when working with categorical inputs.
  - **Process**: I applied the ANOVA F-test to the dataset to compute F-scores for each feature against the target variable**.**
  - **Outcome**: This method brought to light features with high F-scores, suggestive of their potential predictive power.

- **LASSO**

  - **Purpose**: Utilizing the glmnet package in R, LASSO regression helps in performing variable selection and regularization to enhance model prediction accuracy and interpretability.
  - **Process**: I employed the cv.glmnet function for applying LASSO regression, which uses cross-validation to identify the best regularization parameter.
  - **Outcome**: LASSO was instrumental in pinpointing features that maintain a significant relationship with the target variable by zeroing out the coefficients of less influential features.

Results of Feature Selection:

ANOVA F-test Findings: Several key features emerged as statistically significant in predicting the target variable, with the top features being highlighted based on their F-scores.

LASSO Findings: LASSO's regularization identified a subset of features with significant coefficients, indicating their predictive importance
.

Selected Features: A consensus set of 31 features were selected by both ANOVA F-test and LASSO, reflecting their robustness as predictors.
The intersection of features determined by both the ANOVA F-test and LASSO resulted in a refined set of predictors that were considered robust for modelling the target variable, 'o_bullied.' These selected features are deemed essential for the subsequent development of predictive models and are expected to contribute to a model that is not only accurate but also generalizable and resilient to overfitting.

```
[1] "VS0059"    "V4526AA_1" "VS0115"    "V3041"     "VS0123"    "VS0131"    "V3072"
 [8] "V2025A"    "V3071"     "VS0005"    "V3045"     "V3014CAT"  "VS0137"    "VS0141"
[15] "V2026"     "V2041"     "VS0055"    "V3035"     "VS0011.1"  "V2040A"    "VS0026"
[22] "VS0026.1"  "VS0031"    "VS0126"    "VS0146"    "VS0007"    "V3012"     "VS0070"
[29] "VS0024.1"  "V2045"     "VS0147"
```

4. Classification Algorithms

I used 7 classification algorithms logistic regression, decision tree, random forest, SVM, KNN, Naïve Bayes and Neural Network. The following bar graph depicts the accuracy we achieved in each classification model.

1. **Logistic Regression:**
   The logistic regression model achieved an accuracy of 69.77% on the test set. This indicates that the model correctly classified approximately 70% of the cases. The model also has a kappa statistic of 0.3058, which suggests that the model is performing better than chance.

```
          Reference
Prediction   0   1
         0 538  74
         1 225 152
```

This confusion matrix shows that the logistic regression model made 538 correct predictions for class 0 (true negatives) and 152 correct predictions for class 1 (true positives). It also made 74 incorrect predictions for class 0 (false positives) and 225 incorrect predictions for class 1 (false negatives).

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.7051	0.3274	0.8791	0.7051	0.8029	0.5826	0.3058	0.7423596
Class 1	0.6726	0.2949	0.4032	0.6726	0.5242	0.4472	0.2759	0.7423596
Weighted Average	0.6888	0.3161	0.7404	0.6888	0.7146	0.5497	0.3058	0.7423596

In addition to accuracy and kappa, the following metrics were also calculated for the model:

2. **Decision Tree:**
   The decision tree model achieved an accuracy of 75.03% on the test set. This indicates that the model correctly classified approximately 75% of the cases. The model also has a kappa statistic of 0.2484, which suggests that the model is performing better than chance.

```
          Reference
Prediction   0   1
         0 658 142
         1 105  84
```

The Decision Tree model achieved 658 correct predictions for class 0 (true negatives) and 84 correct predictions for class 1 (true positives). However, it also made 142 incorrect predictions for class 0 (false positives) and 105 incorrect predictions for class 1 (false negatives).

In addition to accuracy and kappa, the following metrics were also calculated for the model:

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.8624	0.2376	0.8225	0.8624	0.8424	0.5854	0.2484	0.6170334
Class 1	0.3717	0.6283	0.4444	0.3717	0.4073	0.2794	0.1646	0.6170334
Weighted Average	0.6170	0.4330	0.7348	0.6170	0.6752	0.4366	0.2484	0.6170334

3. **Random Forest:**
   The Random Forest model achieved an accuracy of 76.34% on the test set. This indicates that the model correctly classified approximately 76% of the cases. The model also has a kappa statistic of 0.3096, which suggests that the model is performing better than chance.

```
          Reference
Prediction   0   1
         0 655 126
         1 108 100
```

The Random Forest model achieved 655 correct predictions for class 0 (true negatives) and 100 correct predictions for class 1 (true positives). However, it also made 126 incorrect predictions for class 0 (false positives) and 108 incorrect predictions for class 1 (false negatives).

In addition to accuracy and kappa, the following metrics were also calculated for the model:

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.8585	0.1613	0.8387	0.8585	0.8485	0.6471	0.3096	0.6504657
Class 1	0.4425	0.5575	0.4808	0.4425	0.4620	0.2941	0.1726	0.6504657
Weighted Average	0.6505	0.3594	0.6998	0.6505	0.6750	0.4706	0.3096	0.6504657

The Random Forest model achieved the highest performance among the three models evaluated. But TPR for class 0 is still very low.

4. **Support Vector Machine (SVM):**
   The SVM model achieved an accuracy of 65.62% on the test set. This indicates that the model correctly classified approximately 66% of the cases. The model also has a kappa statistic of 0.2194, which suggests that the model is performing better than chance.

```
          Reference
Prediction   0   1
```

```
  0 513  90
  1 250 136
```

The SVM model achieved 513 correct predictions for class 0 (true negatives) and 136 correct predictions for class 1 (true positives). However, it also made 90 incorrect predictions for class 0 (false positives) and 250 incorrect predictions for class 1 (false negatives).

In addition to accuracy and kappa, the following metrics were also calculated for the model:

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.6723	0.3982	0.8507	0.6723	0.7534	0.4355	0.2194	0.637058
Class 1	0.6018	0.2346	0.4852	0.6018	0.5301	0.3082	0.1798	0.637058
Weighted Average	0.6371	0.3164	0.7196	0.6371	0.6783	0.4199	0.2194	0.637058

5. K-Nearest Neighbors:

The KNN model achieved an accuracy of 62.79% on the test set. This indicates that the model correctly classified approximately 63% of the cases. The model also has a kappa statistic of 0.0918, which suggests that the model is performing slightly better than chance.

```
          Reference
Prediction   0   1
         0 527 132
         1 236  94
```

The KNN model achieved 527 correct predictions for class 0 (true negatives) and 94 correct predictions for class 1 (true positives). However, it also made

132 incorrect predictions for class 0 (false positives) and 236 incorrect predictions for class 1 (false negatives).

In addition to accuracy and kappa, the following metrics were also calculated for the model:

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.6907	0.5841	0.7997	0.6907	0.7404	0.3128	0.0918	0.5572438
Class 1	0.4159	0.3337	0.4159	0.4159	0.4159	0.1963	0.0745	0.5572438
Weighted Average	0.5533	0.4784	0.6852	0.5533	0.6142	0.2549	0.0918	0.5572438

6. Naïve Bayes:

The Naive Bayes model achieved an accuracy of 33.06% on the test set. The Naive Bayes model achieved a low level of performance on the test set. The model's accuracy and kappa statistic suggest that it is performing only slightly better than chance.

```
   Reference
Prediction   0   1
         0 125  24
         1 638 202
```

The Naive Bayes model achieved 125 correct predictions for class 0 (true negatives) and 202 correct predictions for class 1 (true positives). However, it also made 24 incorrect predictions for class 0 (false positives) and 638 incorrect predictions for class 1 (false negatives).

Here is TPR FPR PRECISION RECALL F-MEASURE MCC KAPPA for the Naive Bayes model

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	0.1638	0.1062	0.8389	0.1638	0.3023	0.1312	0.0295	0.5288162
Class 1	0.8938	0.6663	0.2285	0.8938	0.3650	0.2531	0.0980	0.5288162
Weighted Average	0.5288	0.3835	0.4722	0.5288	0.4998	0.2117	0.0295	0.5288162

7. Neural Network:

The Neural Network model achieved an accuracy of 77.45% on the test set. This indicates that the model correctly classified approximately 77% of the cases. However, the model has a kappa statistic of 0.0203, which suggests that the model is not performing much better than chance.

```
          Reference
Prediction   0   1
         0 763 223
         1   0   3
```

The Neural Network model achieved 763 correct predictions for class 0 (true negatives) and 3 correct predictions for class 1 (true positives). However, it made 223 incorrect predictions for class 0 (false positives) and did not make any incorrect predictions for class 1 (false negatives).

	(TPR)	(FPR)	Precision	Recall	F-Measure	MCC	Kappa	ROC
Class 0	1.00000	0.22851	0.77383	1.00000	0.87097	0.02033	0.02033	0.5066372
Class 1	0.01327	0.98673	0.01394	0.01327	0.02564	0.00042	0.00042	0.5066372
Weighted Average	0.77450	0.22550	0.25540	0.77450	0.41379	0.02033	0.02033	0.5066372

5. Best Model Parameters

The best-performing model in my analysis was obtained through logistic regression, a statistical method often favored for predicting binary outcomes. This logistic regression model was configured using the generalized linear model (GLM) function with a binomial family in R, signifying the binary target variable. The following is an elucidation of the model parameters and its performance:

- **Model Training**:

My logistic regression model was trained using the glm function in R, applying the formula o_bullied ~ ., wherein o_bullied signifies the dependent variable — whether an individual was subjected to bullying. The . represents the inclusion of all other predictor variables in the train_data dataset.

- **Family**:

The parameter family=binomial was selected to denote the binary nature of the response variable, which is typical for logistic regression models dealing with dichotomous outcomes.

- **Predictions**:

I generated predictions by using the predict function on the fitted model model_logreg, utilizing the test_data (minus the target column) as new input. The predictions were obtained as probabilities (type='response') of the positive class (bullying occurrence).

- **Thresholding**:

A decision threshold of 0.5 was employed to transform the predicted probabilities into binary classifications (0 or 1), indicative of the absence or presence of bullying.

- **Accuracy Calculation**:

The accuracy metric was computed by contrasting the binary predictions against the actual observations in test_data$o_bullied. I utilized the mean function to ascertain the correct prediction ratio.

- **Performance Measures**:

The confusionMatrix function from R's caret package was instrumental in producing a confusion matrix and related statistics to assess the model's efficacy. The performance output was as follows:

- **Accuracy**: The model attained an accuracy of about 69.77%.
- **Kappa**: The Kappa statistic stood at 0.3058, denoting a moderate level of agreement.
- **Sensitivity (True Positive Rate for class 0)**: Achieved 70.51%, surpassing the threshold of 70%.
- **Specificity (True Positive Rate for class 1)**: Reached 67.26%, which is above the minimum requirement of 65%.
- **Positive Predictive Value (Precision)**: Recorded at 87.91% for class 0.
- **Negative Predictive Value**: Stood at 40.32% for class 1.

The parameters of this logistic regression model and the resultant metrics confirm that it successfully meets the set criteria for performance, especially in terms of sensitivity and specificity for the classification of bullying victims. The model's significant sensitivity and specificity underscore its capability to accurately identify true positives in both classes, underscoring the effectiveness of the chosen approach.

6. Discussion:

- **Interpretation of the results**

My data mining project's findings offer a thorough analysis of how well the various classification algorithms identify bullying victims. The logistic regression model was the most successful in achieving a balance between specificity and sensitivity, indicating that it is appropriate for the given dataset.

With a Kappa statistic of 0.3058, the logistic regression model's accuracy was roughly 69.77%, indicating a moderate degree of agreement above chance. With a sensitivity of 70.51% for class 0 and 67.26% for class 1, our model showed a respectable True Positive Rate (TPR) for both classes, demonstrating its usefulness in properly identifying non-victims and victims of bullying, respectively.

- **Comparison of the models based on performance measures**

The logistic regression model outperformed the other seven classification models—decision tree, random forest, SVM, KNN, Naïve Bayes, and neural network—in terms of striking a better balance between sensitivity and specificity.

With accuracies over 75%, the decision tree and random forest models demonstrated promise; yet, their Kappa values revealed just a marginal increase over random chance, raising doubts about their dependability.

Lower accuracies and Kappa values were observed in the SVM, KNN, and neural network models, indicating less consistency in their prediction abilities.

With the lowest accuracy and a Kappa statistic that hardly indicated an improvement above guesswork, Naïve Bayes demonstrated severe limits, suggesting that it was the least appropriate model for the dataset.

- **Insights and findings from the data mining process**

LASSO and ANOVA F-tests were both used in the feature selection process, which helped to uncover a strong collection of predictors. Only the most relevant characteristics were kept after LASSO effectively penalised and removed non-contributory predictors. The ANOVA F-test revealed features with significant variations in means across groups.

The agreement between these two approaches was mirrored in the final selection of 31 features, highlighting their significance in accurately predicting the target variable.

The process of data mining brought to light the difficulties in identifying victims of bullying, highlighting the need for a sophisticated strategy to address data imbalances and subtleties. After investigating a number of classification algorithms, it was found that the logistic regression model performed the best, highlighting the value of choosing a model that is straightforward and easy to understand. Logistic regression demonstrated robustness in spite of the dataset's complexity, indicating that more intricate algorithms may not always produce superior outcomes.

In conclusion, the project demonstrated the value of thorough preprocessing and careful model selection. A logistic regression model that not only satisfies performance criteria but also provides insight into the dynamics of school bullying and has implications for the development of focused treatments is the result of meticulous dataset preparation and model evaluation.

7. Conclusion

The elements associated with bullying have been elucidated by the utilisation of data mining techniques, particularly logistic regression, in the examination of the presented dataset. Achieving an accuracy of 69.77%, the original logistic regression model had true positive rates of 67.26% for class 1 (bullied) and 70.51% for class 0 (not bullied). Although this is a good starting point, there is certainly space for development.

- **Implications of the results**

The findings suggest that the selected features have a measurable impact on the likelihood of experiencing bullying. Understanding these relationships is crucial for developing interventions and preventative measures. Variables that significantly influence the probability of being bullied can be targeted in awareness programs or monitored more closely in educational and social settings.

- **Limitations of the study**

The use of a logistic regression model, which is interpretable but may not be as good at capturing intricate non-linear correlations or interactions between features as more complex models, is one of its limitations. The quality of the data supplied is another factor that affects the model's performance; biases or mistakes in the dataset could have an effect on predictions.

- **Recommendations for future research or application**

In order to improve predictive accuracy even more, future studies can investigate the application of ensemble techniques such model stacking. Model stacking, which makes use of many modelling techniques, aggregates forecasts from several models to generate a final prediction that may be more accurate.