



MIS772 – Predictive Analytics

T2 2023

Assignment 1 – Individual

Student name: Sachin Bhat

Student number: 218676233

Executive summary

(1 page)

A legal firm known by the name DAX compensation lawyers has been facing a critical issue in the insurance sector. Identification of fraudulent claims is what we are looking at. A lot of claimants make ill use of the system's claiming procedure by presenting fraudulent personal injury claims. These are serious challenges that lead to extreme financial losses for the company while also putting the genuine claimants under suspicion.

Executive Problem Statement:

The firm has been tackling high and an increasing number of fraudulent claims under the category of personal injury insurance. Such fraudulent exist within the data of above 3000 claims which the firm has provided us with. Non-genuine claimants constantly aim to exploit the firm's claiming procedure by intentionally submitting claims with non-existent, unrelated, and exaggerated injuries in order to get a hold of non-deserving compensations. This is resulting in financial losses to the firm while also putting the genuine claimants holding the policies under the suspicious eye thereby questioning the whole system of insurance claims.

Based on the data presented to us, we are to provide the firm with an effective method enabling them to identify and inspect potential frauds in the **future** thereby protecting the rights of genuine claimants. We are also aiming at minimising the firm's financial losses and keeping our client's best interests in consideration, we want to assist them in practicing their business ethically. The data provided to us by DAX compensation lawyers can be utilised well to assist them with the issues they are facing.

Executive Solution Statement:

After examining various aspects of the claims, it is noticeable that quite a few factors are correlated. We have included visualisations for some of those factors in this report. While exploring the data using visualisations and careful inspection, it was noticed that fraudulent claims are more likely to occur where the claimants have mentioned: the **cause of injury** (Figure 2) as struck object, lifting and slip/fall, **nature of injury** (Figure 3) as contusion and sprain/strain, their **marital status** as married and widowed, **injured body parts** as back and head. Such frauds are also likely to occur among the claimants: between the **age group** of 30-40 years, who had no **witness present** (Figure 1) at the time of incident, and who reported that **motor vehicles** were not involved in the incident.

Out of all the approaches tried and tested thoroughly during the process, we would recommend the kNN modelling approach with its value = 5 (Figure 12) to DAX Compensation Lawyers. After following our recommendations, they would be able to make better predictions while identifying genuine or fraudulent claimants in the future. This would allow the firm to significantly reduce their financial losses while streamlining the claiming process. Genuine claimants' faith would be restored in the firm and eventually the industry. The customers would be happy and satisfied with the firm's fast decision-making process. The firm would get recognised even better in the sector. This would eventually generate more customers and revenue.



Data exploration, pattern discovery, and preparation

(2 pages)

Selecting the data:

The dataset was imported in the RapidMiner environment, and all the further steps were implemented over there. The dataset that has been provided to us by the firm contains 12 attributes. We had to proceed with relevant selection of these attributes and classify them as predictors and the label. Predictor attributes must be related to the label attribute. Based on the requirements and our logic, we chose the following attributes as the **predictors**: Claimant Age, Claimant Marital Status, Body Part, Nature of Injury, Cause of Injury, Witness Present, Vehicle Flag. We chose **Fraud Flag** as the **label attribute** because we want to predict whether a **claim was detected to be fraudulent or not**.

How the missing values were dealt with:

To ensure that our dataset is meaningful, we had to carefully deal with missing values. There are several techniques that can be used to deal with them such imputation, data transformation, taking mean or mode etc. In the provided dataset, 26 missing values were found amongst our relevant attributes. We used the **Filter Examples** operator to handle missing values. We asked it to **filter through the condition** class of **no_missing_attributes** which **matches us with examples with no missing values**.

Transforming the data:

Transformation of the relevant attributes is essential if and when the business problem requires it. We had to transform the Fraud Detection Flag attribute (1 = Yes, 0 = No) as it is our label attribute, and it has to be nominal. Using the Numerical to Binomial operator, we transformed it from integer to binomial. Now, the values which had the flag as 1 have been converted to True and the values with flag as 0 have been converted to False.

Data Distribution Analysis:

From several visualisations available to us, we made use of different bar charts to explore the data and possible relationships among our attributes. Some of the relevant predictors are visualised below.

Figure (1). Witness Present.

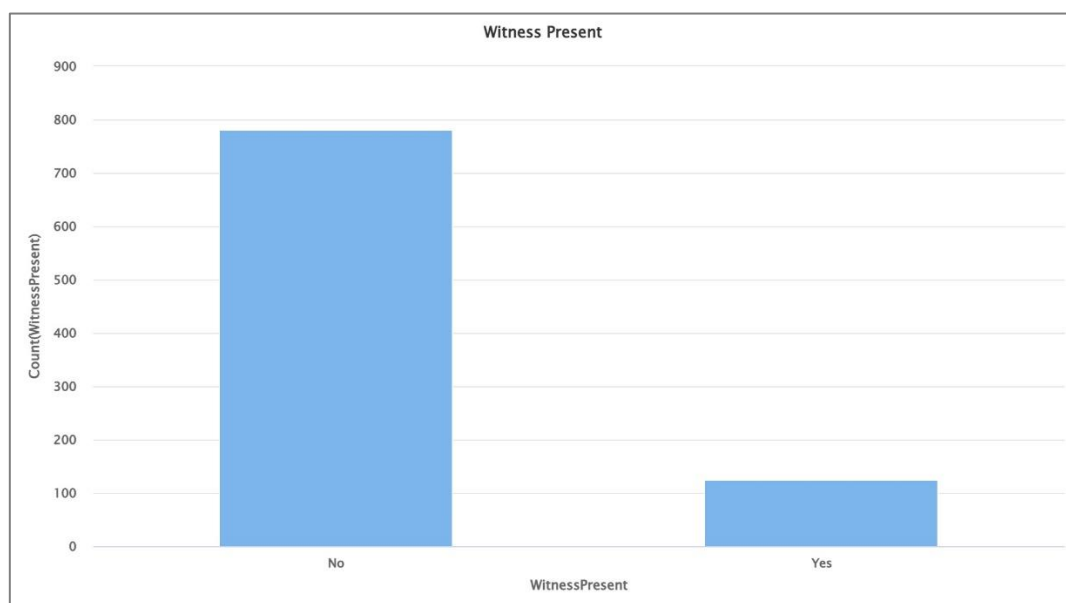




Figure (2). Cause of Injury

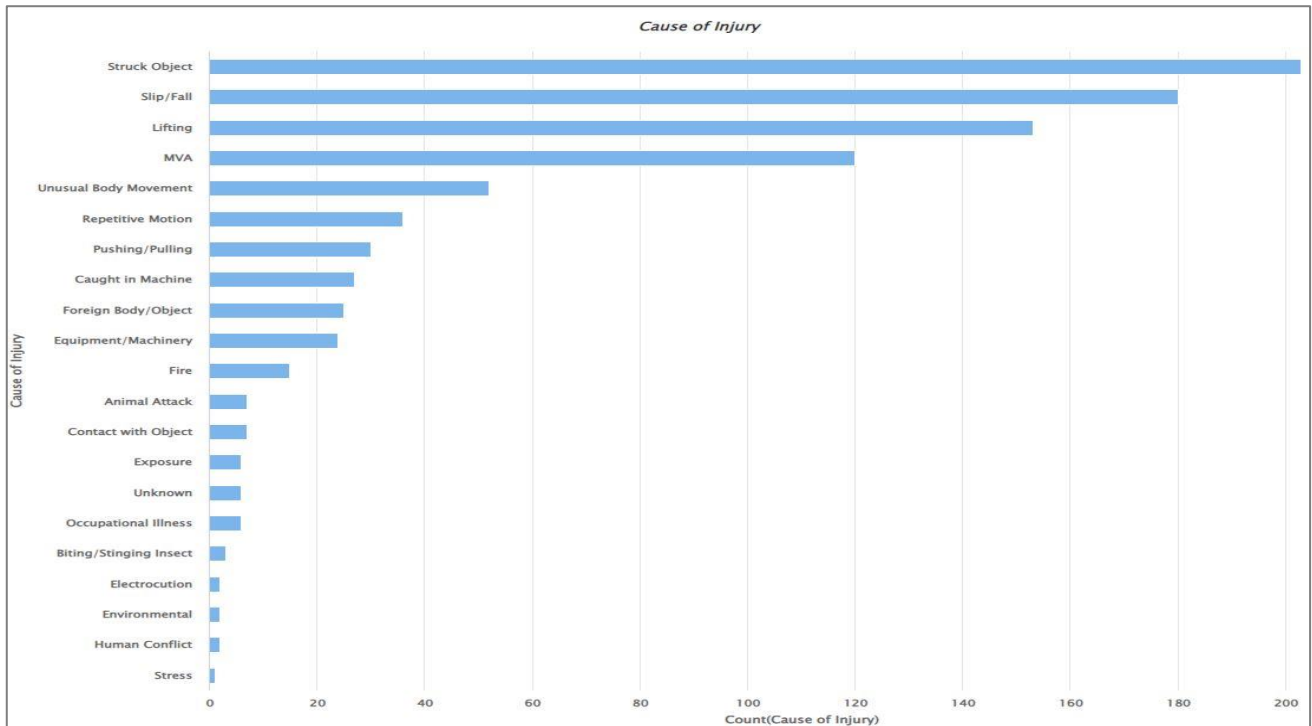
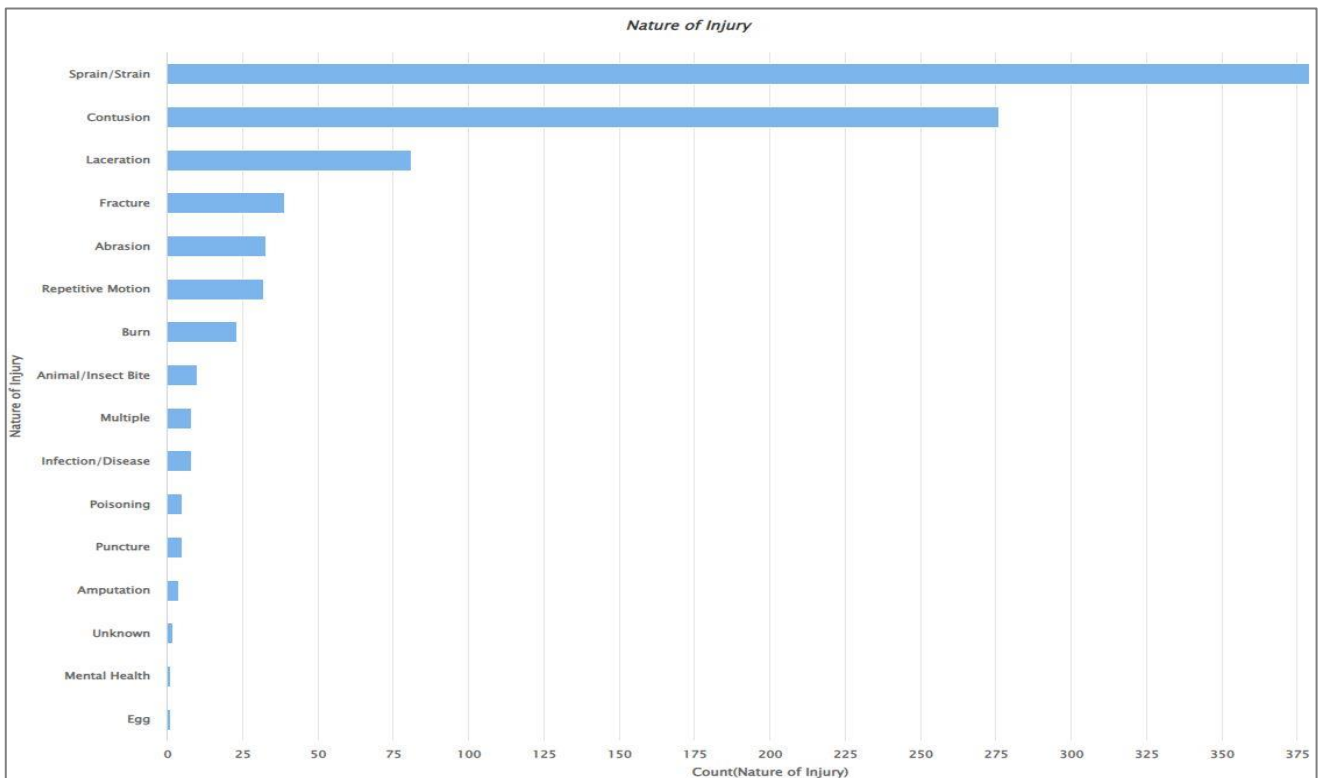


Figure (3). Nature of Injury



In Figure (1): 781 claims did not have witnesses present during the incident while 126 did. In Figure (2): Struck object, slip/fall, and lifting were the most submitted causes of injury. In Figure (3): Sprain/strain and contusion were the most submitted natures of injury.

Predictive modelling

(2 pages)

After data exploration, pattern discovery and data preparation, we have achieved a clean and meaningful dataset. To perform predictive modelling on this dataset, we have utilised RapidMiner's kNN classification and Gradient Boosted Trees models.

kNN approach goes well with our scenario as the dataset contains a combination of categorical and numerical attributes and it is known to perform well in such cases. It is efficient in identifying patterns which is quite useful in identifying fraudulent claims. **Gradient Boosted trees** are known to provide robust and precise predictive models. It works out complicated interactions between different attributes and can efficiently get hold of non-linear relationships which is quite useful for our case.

Figure (4). kNN with weight 3

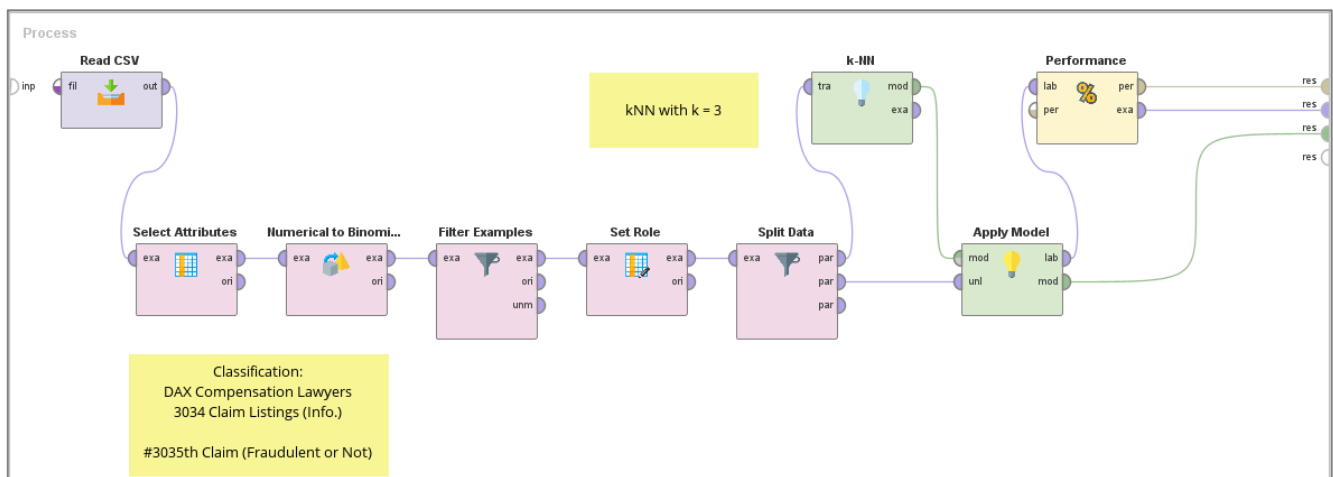
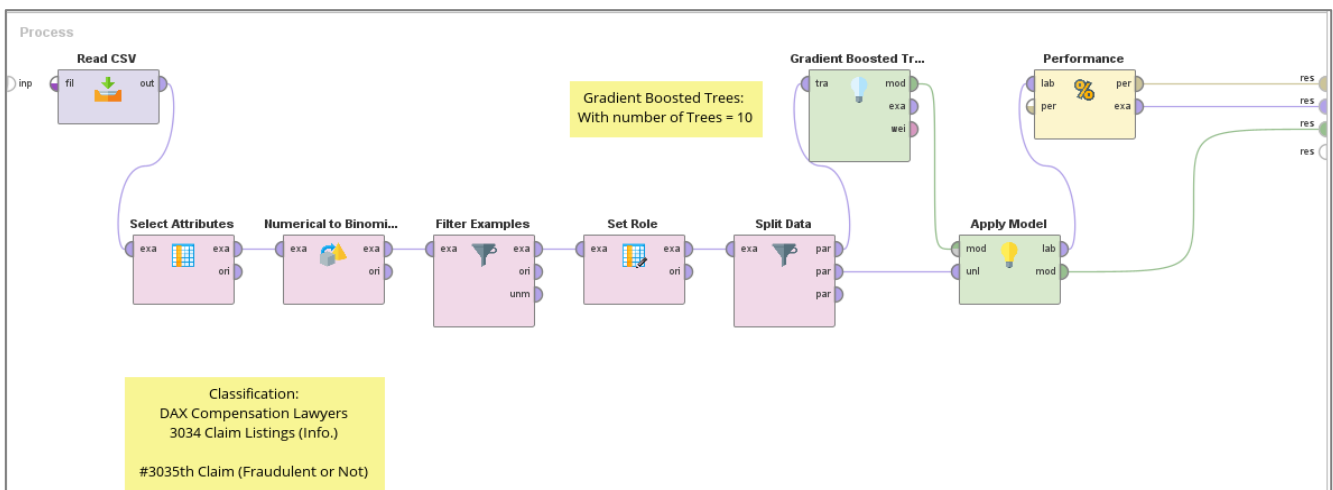


Figure (5). Gradient Boosted Trees with number of trees = 10



In Figure (4) and (5), similar process and steps were utilised **except** the predictive model used. In Figure (4) the kNN approach has been demonstrated and in Figure (5) the Gradient Boosted Trees approach has been shown. The Read CSV operator was used to import the dataset into the RapidMiner environment. Using Select Attributes (Figure 6) we let RapidMiner know about what the predictor attributes should be. Using Set Role, we established Fraud Flag as the label attribute. But as it was of integer type (with values 0 and 1), we had to convert it to binomial using Numerical to Binomial operator (Figure 9) because label attribute cannot handle non-nominal attributes. Before proceeding further, we dealt with missing values using the Filter Examples (Figure 7) operator. We instructed it to match us with examples having no missing attributes set as its parameter. Using the Split Data operator (Figure 8), we divided the dataset into 2 parts: 70% of data was used to train the model meanwhile 30% of the data was kept for testing using the Apply Model operator. We

set the **local random seed** (Figure 8) to a constant i.e., 2023. Then, as shown in Figure (4) we used kNN and in Figure (5) we used Gradient Boosted Trees to train that 70% of the data. Trained output from the predictive models was passed through the Apply Model operator along with the test data so that it could be applied. Then, the trained output was passed through the Performance operator to test its performance based on the accuracy, kappa, and AUC. In the results, we would achieve details of the model, the performance vector and the example set.

Figure (6). Select Attributes Parameters

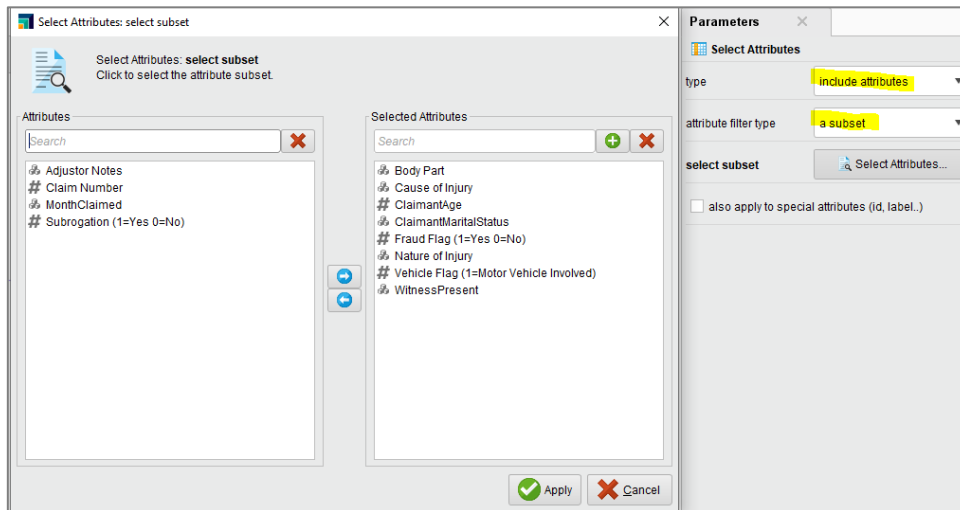


Figure (7). Filter Examples Parameters

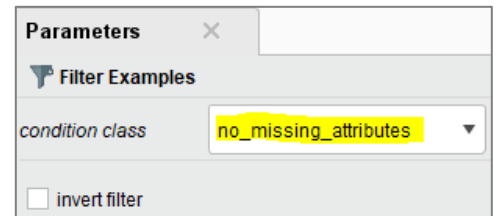


Figure (8). Split Data Parameters

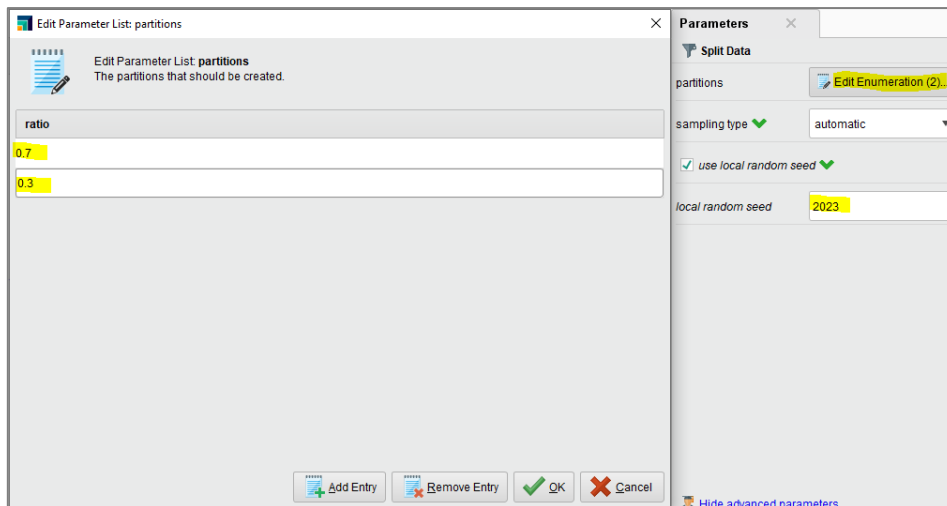
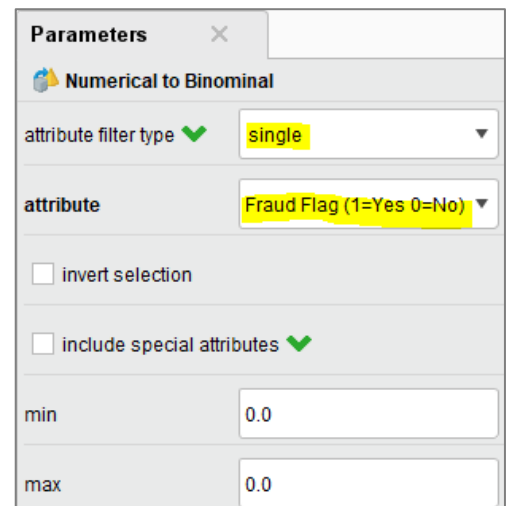


Figure (9). Numerical to Binomial Parameters



For kNN, k values of 3 and 5 have been used to test and compare model's performance. A lower value of k such as 3 is comparatively more sensitive than 5 while identifying local patterns in the data. If the dataset has outliers, a higher value such as 5 would be less prone to misclassifications due to underfitting.

For Gradient Boosted Trees, the number of trees have been set to 10, 30 and 50 to test and compare model's performance. A higher value of the trees such as 50 delivers a complex and powerful model as compared to 30 and its efficiency goes even lower when set to 10. Higher number of trees ensure better potential to identify crucial interactions and patterns in the data.



Model evaluation and improvement

(2 pages)

Hold-out sampling (Split Data into 70% for training and 30% for testing) was conducted in the Predictive Modelling section using kNN and Gradient Boosted Trees (GBT). For the predictive models created previously, the results based on accuracy, kappa and AUC are provided below:

For **kNN** with **k value = 3**: accuracy was 96.36 %, kappa was -0.009 and AUC was 0.562. With **k value = 5**: accuracy was 96.91 %, kappa was 0 and AUC was 0.587. For **GBT** with **trees = 10**: accuracy was 93.05 %, kappa was 0.103 and AUC was 0.711. With **trees = 30**: accuracy was 94.05 %, kappa was 0.069 and AUC was 0.710. With **trees = 50**: accuracy was 94.27 %, kappa was 0.074 and AUC was 0.729.

We will now try to evaluate and compare different predictive models that we have created.

For kNN with k=3:

- 874 claimants were actually genuine and predicted as genuine.
- 5 claimants were actually genuine and predicted as fraud.
- 0 claimants were actually fraud and predicted as fraud.
- 28 claimants were actually fraud and predicted as genuine.

For kNN with k=5:

- 879 claimants were actually genuine and predicted as genuine.
- 0 claimants were actually genuine and predicted as fraud.
- 0 claimants were actually fraud and predicted as fraud.
- 28 claimants were actually fraud and predicted as genuine.

For GBT with trees = 10:

- 839 claimants were actually genuine and predicted as genuine.
- 40 claimants were actually genuine and predicted as fraud.
- 5 claimants were actually fraud and predicted as fraud.
- 23 claimants were actually fraud and predicted as genuine.

For GBT with trees = 30:

- 850 claimants were actually genuine and predicted as genuine.
- 29 claimants were actually genuine and predicted as fraud.
- 3 claimants were actually fraud and predicted as fraud.
- 25 claimants were actually fraud and predicted as genuine.

For GBT with trees = 50:

- 852 claimants were actually genuine and predicted as genuine.
- 27 claimants were actually genuine and predicted as fraud.
- 3 claimants were actually fraud and predicted as fraud.
- 25 claimants were actually fraud and predicted as genuine.

Then we will perform cross validation on the original dataset while keeping the number of folds as 3 and 9. We obtained the following outcomes:

With number of folds = 3: accuracy was 94.38% +/- 1.58%, kappa was 0.085 +/- 0.067, and AUC was 0.584 +/- 0.047.

- 2843 claimants were actually genuine and predicted as genuine.
- 87 claimants were actually genuine and predicted as fraud.
- 10 claimants were actually fraud and predicted as fraud.
- 83 claimants were actually fraud and predicted as genuine.

With number of folds = 9: accuracy was 94.01% +/- 1.30%, kappa was 0.086 +/- 0.121, and AUC was 0.583 +/- 0.093.

- 2830 claimants were actually genuine and predicted as genuine.
- 100 claimants were actually genuine and predicted as fraud.
- 12 claimants were actually fraud and predicted as fraud.
- 81 claimants were actually fraud and predicted as genuine.

Figure (10). Example of Cross Validation. No. of folds = 9

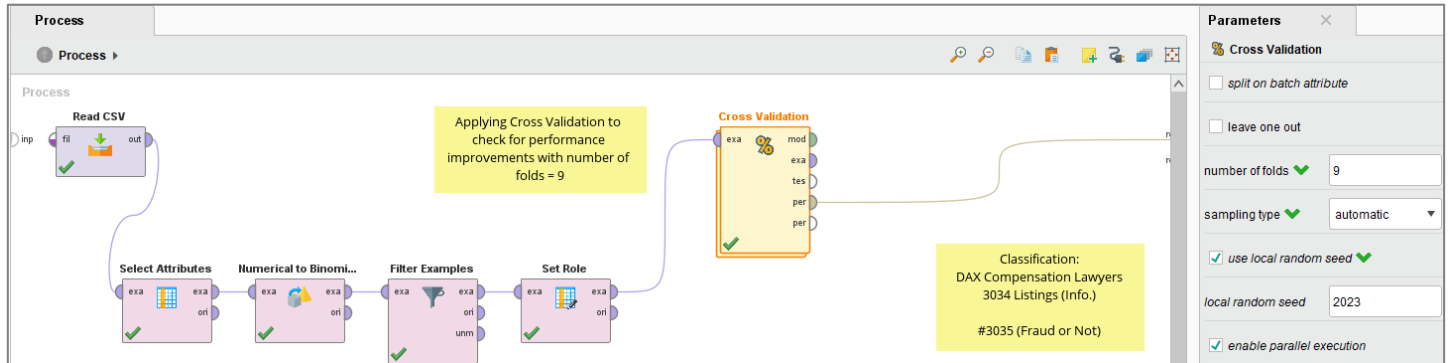
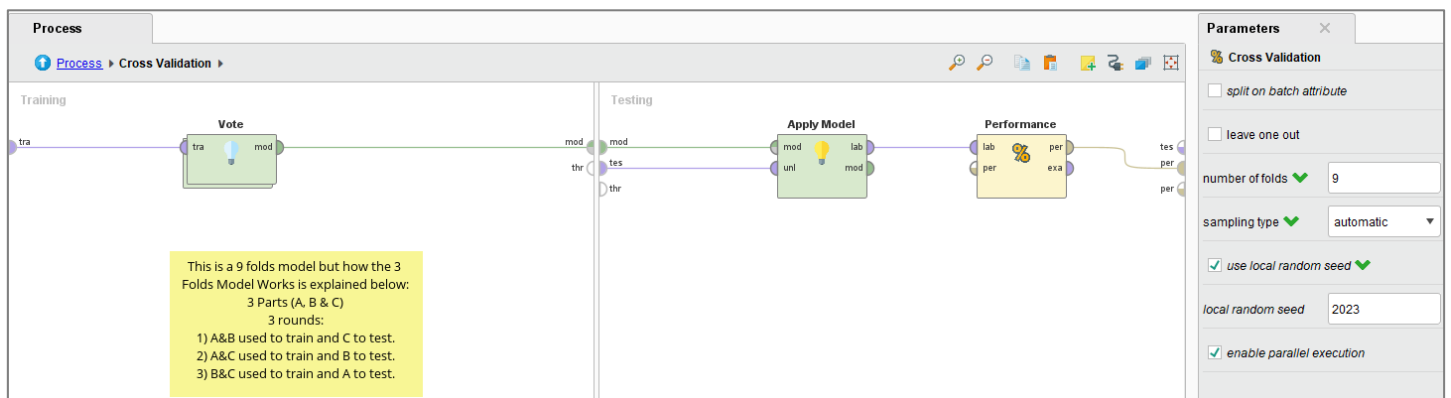


Figure (11). Constituents of Cross Validation. No. of folds = 9



Among the Cross Validation approach, we utilised the method of voting but both the models (no. of folds=3 and 9) perform almost similarly. No significant differences could be observed among the results.

kNN Model: Value of $k=5$ performs better than $k=3$ as it delivers higher number of actually genuine and predicted as genuine claimants. Also, it comparatively has better accuracy value.

GBT Model: Value of trees = 50 performs better than 30 and 10 as it collectively delivers higher number of actually genuine and predicted as genuine claimants along with actually fraud and predicted as fraud claimants. Also, it comparatively has better accuracy value.

Figure (12). kNN with weight = 5

accuracy: 96.91%			
	true false	true true	class precision
pred. false	879	28	96.91%
pred. true	0	0	0.00%
class recall	100.00%	0.00%	

Out of kNN and GBT models, the kNN model with value of $k=5$ performs the best based on its overall results. We will recommend this model to the DAX Compensation Lawyers firm.