

**Project Final Report**

**Group Number- 5**

**Group members:**

1. **Pravallika Bollu**
2. **Sirisha Pathlawath**
3. **Ujjwal Vinnakota**
4. **Venkata Sri Ambica Mokarala**
5. **Yagya Pandey**

**Objective:**

Find out the best model/algorithm that predicts one out of three functionality labels (Functional, non-functional, and functional needs repairs) of water pipes in the given dataset using SAS enterprise miner.

**Methodology:**

To Predict the functionality of the water pipes we will use three classification algorithms in this project. The three classification models that are used to predict the output in this project are Logistic regression, Decision tree and Neural Networks.

As shown in the variable summary, there are 31 nominal variables and 9 interval variables, and timeId variable. There are a total of 47520 observations in the given dataset. The dataset contains different fields that explain different features of the water pipes. Features such as basin (Geographic water basin), scheme\_management (Who operates the waterpoint), source (The source of the water), construction\_year (Year the waterpoint was constructed) etc. are some of the helpful features to find out the functionality and location of the water pipes in the dataset. We can also reject “num private” and “recorded\_by” as there is not much useful information in these data fields.

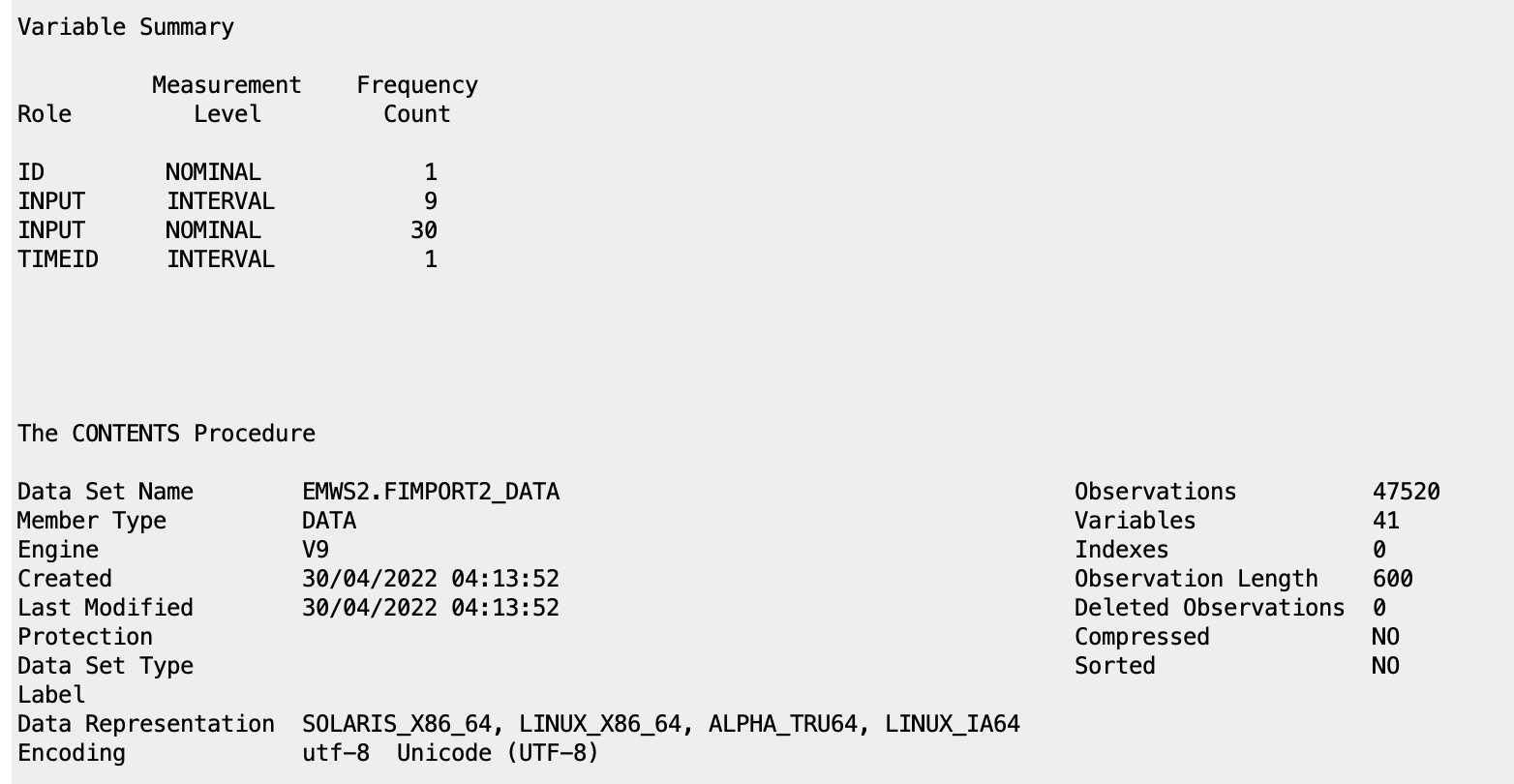


Fig. Variable summary of the dataset.

Before modeling our data, it is essential to understand and clean the data to get the best results. We perform basic EDA on the dataset to analyze the data, find out missing values, detect anomalies in the data, and find patterns. “StatExplorer” node is used to analyze the variables.

**Exploratory data Analysis:**

Dataset contains a total of 39 variables. The dataset consists of input variables of both nominal and interval datatypes and the target variable of the dataset is “status\_group” which has 3 labels that are to be predicted by the algorithms. The Status\_group variable of the dataset has 3 values functional, non-functional, and functional needs repair.

Graphical user interface, table

Description automatically generatedFig. StatExplorer output (class variables)

Number of levels:

There are some nominal input variables with more than 128 levels. The variables with more than 128 levels which are not required for the analysis are Funder (Who funded the well), installer (Organization that installed the well), Scheme\_name (Who operates the waterpoint), Subvillage (Geographic location), ward (Geographic location), wpt\_name (Name of the waterpoint if there is one).

Missing Values:

The variables funder, installer, permit, public\_meeting, scheme\_management, scheme\_name, subvillage are some of the variables with missing variables.

Table

Description automatically generated

Fig. StatExplorer output. (Interval variables)

Skewness:

There are some interval variables that do not have a skewness value between +3 to -3. Means these variables are not normally distributed. The variables are amount\_tsh, longitude, num\_private and population.

To run the data through the models, It is important to clean data of missing values because not all algorithms/models are good at handling the missing values.So, to handle the missing values and to clean the data we use “Replacement” node and “StatExplorer” as shown in below fig.

Text, chat or text message

Description automatically generated

Fig. Diagram.

Graphical user interface, table

Description automatically generated

Fig. Replacement editor window

The empty values in the dataset are set to “\_MISSING\_” in the replacement editor window of the replacement node. And the data/variables after the replacement can be explored using StatExplorer connected to the replacement node. The output of StatExplorer after replacing the missing values is shown below.

Table

Description automatically generated

Fig. Output of StatExplorer after replacement.

Variable worth and chi-square plots of the StatExplorer output are shown below.   
The larger the Chi-square value, the greater the probability that there really is a significant difference. From analyzing the plots, we can find quantity and quantity\_group are the two variables with the highest worth in the dataset. variables lga, Quantity and quantity\_group have the high chi-square values.

Chart, bar chart, histogram

Description automatically generated

Fig. Variable worth plot of statexplorer 2

Chart, histogram

Description automatically generated

Fig. Chi-Square plot of statexplorer 2

After rejecting the input variables which have more than 128+ levels, and variables with skewness not in the range of +3 to -3 and replacing the missing values. Now our data is ready to be modelled.

**Data Partitioning:**

Data partitioning is a technique used to distribute data into multiple datasets to improve the performance of the model. The main goal of any classification algorithm is to find out how accurately a model can predict unseen instances. So, we use data partitioning and validation datasets to find the accuracy of the model. In this project, cleaned data is partitioned 70% into training (for preliminary model fitting, to find the best model weights using this data set), 15% into validation (assessing the adequacy of the model, and for model fine-tuning), and 15% into test data. In this way, we can prevent our model from overfitting, and this will accurately evaluate our model. The data partition node is connected to the replacement node as shown below to check the discrepancy.

Diagram

Description automatically generatedFig. Diagram.

Now this partitioned data is used for the Logistic regression, Decision tree and Neural Networks algorithm to predict the target variable.

**Model 1- Decision Tree & Random Forest:**

A decision tree is a flowchart and a specific type of probability tree that starts with one main idea and divides it into branches based on the decisions. Tree construction is performed in top-down, recursive, divide-and-conquer manner. In complex decisions or when different factors including the uncertainty involved, then decision trees are the best model to deal with. It is very helpful in analyzing quantitative data and making a decision based on numbers. A decision tree includes some symbols like alternative branches, decision nodes, chance nodes, and end nodes. These symbols combinedly explain the outcomes or decisions of the model. In decision tree, if all the data belongs to one class, then we call it as a pure node. The color of the branch represents the purity level.

To clearly lay out the issue, analyze the possible consequences of our decision, and provide a framework to qualify our values of functional, nonfunctional, and the water pumps that need repair, let’s start with the Decision Tree for analysis.

A screenshot of a computer

Description automatically generated

Fig. Diagram representing connection of Decision Tree node

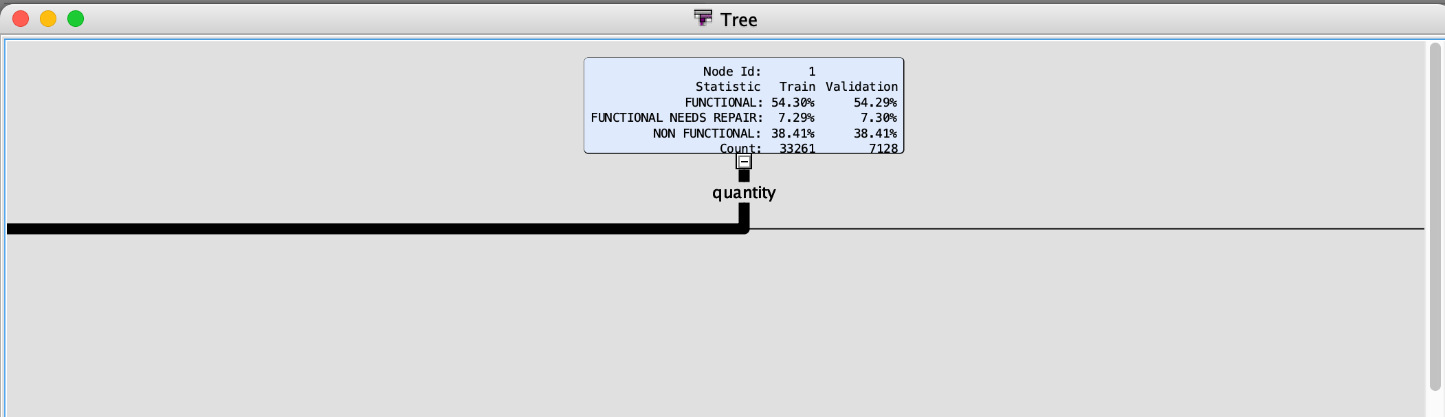
There are two types of decision trees.

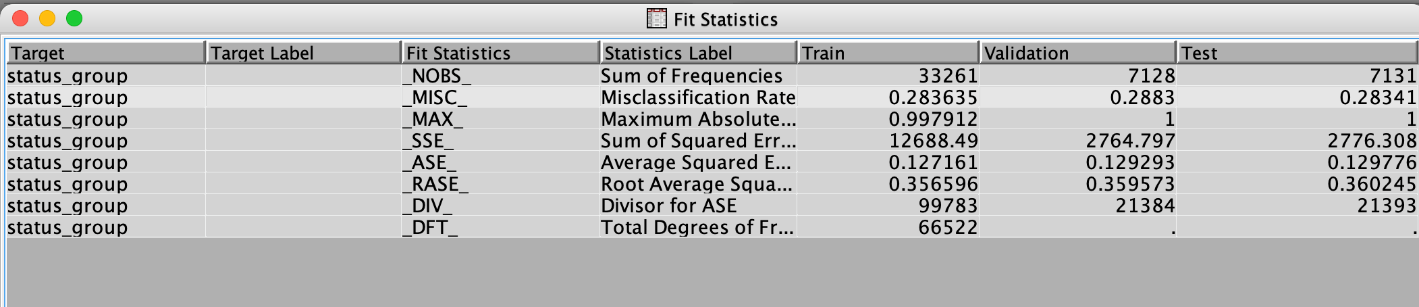
1. Full Tree

2. Pruned Tree

**1.1. Full-Tree:**

In the full tree, we run the algorithm completely and let it grow fully. In this analysis by default, we have the maximum branches to 2 and depth to 6. That means the tree will grow up to 6 generations of splits and we are analyzing the assessment measure based on the misclassification rate.

Fig. Tree window

From the results above in the Tree window, we can observe quantity is the first variable to split and we can also see that the ratio is divided into functional is 54.30%, functional needs repair is 7.29%, and nonfunctional is 38.42% for training data and for validation data set 54.29% as functional, 7.30% as functional needs repair and 38.42% as non-functional.  
  
  
Fig. Fit statistics

From the fit statistics window, we can observe the misclassification rate for training, validation, and test datasets. The Misclassification rate for training data is 0.283635 (28.3635%) so the accuracy is 1- 0.2836 = 0.7164 I.e., **1.64%**. Whereas the misclassification rate for validation data set is 0.2883(28.83%) so the accuracy can be calculated as 1- 0.2883= 0.7117 I.e.,**71.17%**  
Coming to the test dataset, the misclassification rate is 0.28341(28.341%) which means the accuracy is 1- 0.28341 = 0.71659 I.e.,**71.659%**.

The below figure represents the importance of the variables that are contributing more to our model prediction and the number of splitting rules for each variable. As “quantity” is the first variable to split it represents 1.0000 which means 100% of importance and the remaining variables follow in the order.

Table

Description automatically generatedFig. Variable importance

**1.2. Pruned Tree:**

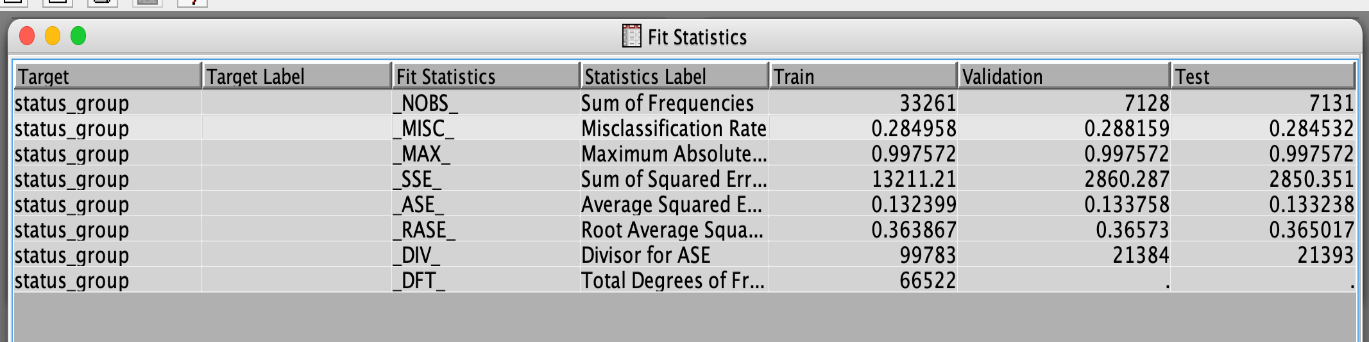
The only drawback in the full decision tree is “Overfitting”. As we allow the tree to grow fully, the number of branches increases which results in noises and outliers, and due to that Overfitting issue occurs. So, to stop that issue we use pruned tree which is the modification of full tree. In pruned tree, we reduce the number of child nodes from the branch nodes.

It is of two types:

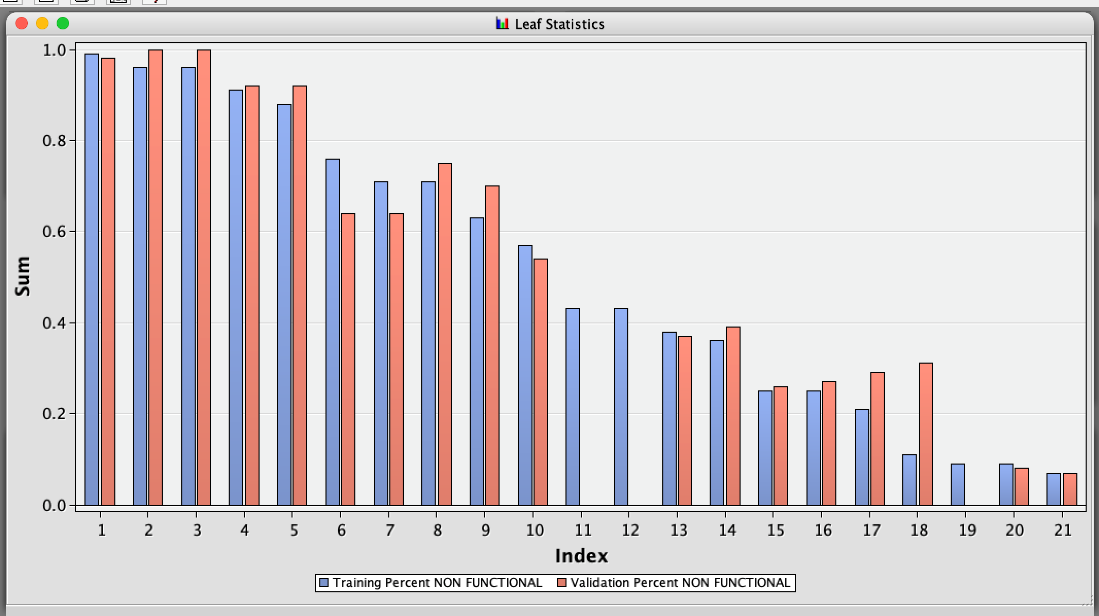
1. Pre-Pruning or early stopping
2. Post-Pruning

In Pre-Pruning, we stop the tree before it completes the classification whereas, in post-pruning we prune the tree after it completes the classification. Here in SAS enterprise miner, by selecting the assessment method on misclassification rate measure, it automatically performs the pruning process. It stops the growth of tree at a minimum misclassification rate in the validation set. So, in the end, it results in a smaller number of nodes compared to full tree model.

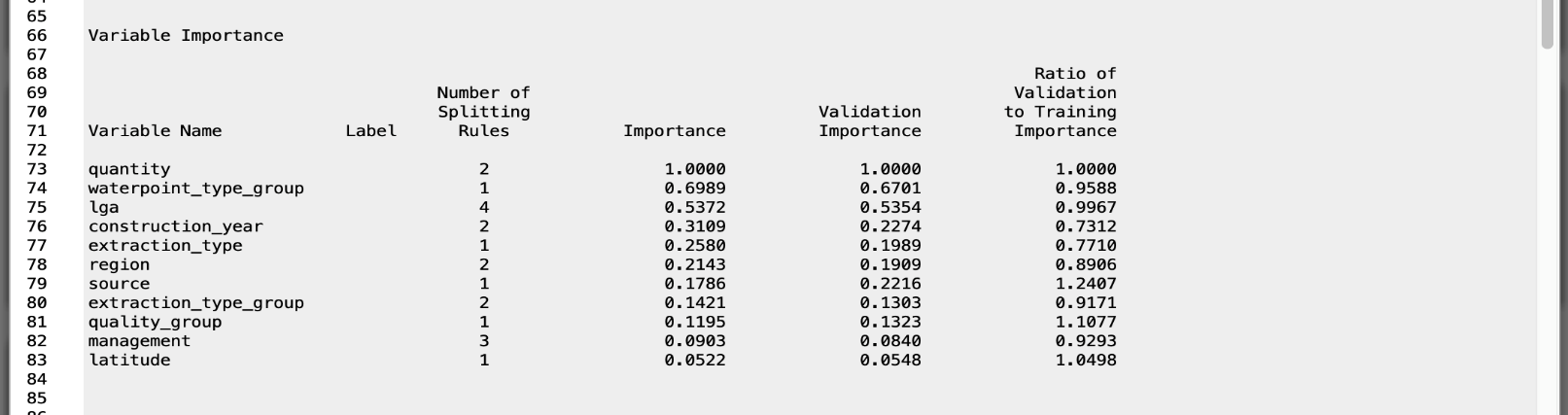
A screenshot of a computer

Description automatically generated  
Fig. Adding Pruned  
Fig. Fit statistics for Pruned tree

From the fit statistics of Pruned decision tree model, we can observe the misclassification rates. The misclassification rate for train data is 0.284958(28.4958%) which means the accuracy is 1-0.284958 = 0.715042 I.e., **71.50%**. For the validation data, it is 0.288159 (28.8159%) so the accuracy will be 1-0.288159 = 0.711841 I.e., **71.18%**. Whereas, for test data the misclassification rate is 0.284532 so the accuracy is 1-0.284532 = 0.715468 I.e., **71.5468%.**

  
Fig. leaf statistics shows leaves of the decision tree

Along with the number of branches, we can also observe the number of leaves in the model using leaf statistics. From the above fig, we can see the leaf statistics plot of the pruned decision tree. So, the index on the X-axis represents the number of leaves in the optimal tree. Here in this model, the number of leaves is 21.

Fig. Variable Importance after Pruned in output window

By comparing the variable importance of both full tree and pruned tree, we can observe the changes in the number of splitting rules (especially for lga variable) and minor changes in the importance of the variables as well.

**1.3. HP Forest:**

The Hp forest model is one kind of decision tree. As it uses the methodology of the decision tree and creates random forests in a high-performance environment. It creates a predictive model for regression. Random forest is an ensemble learning method for classification and regression tasks. The random forest will have several trees different from each other based on training, validation, and test sets.

Graphical user interface, application

Description automatically generated

Fig. Connecting HP Forest node

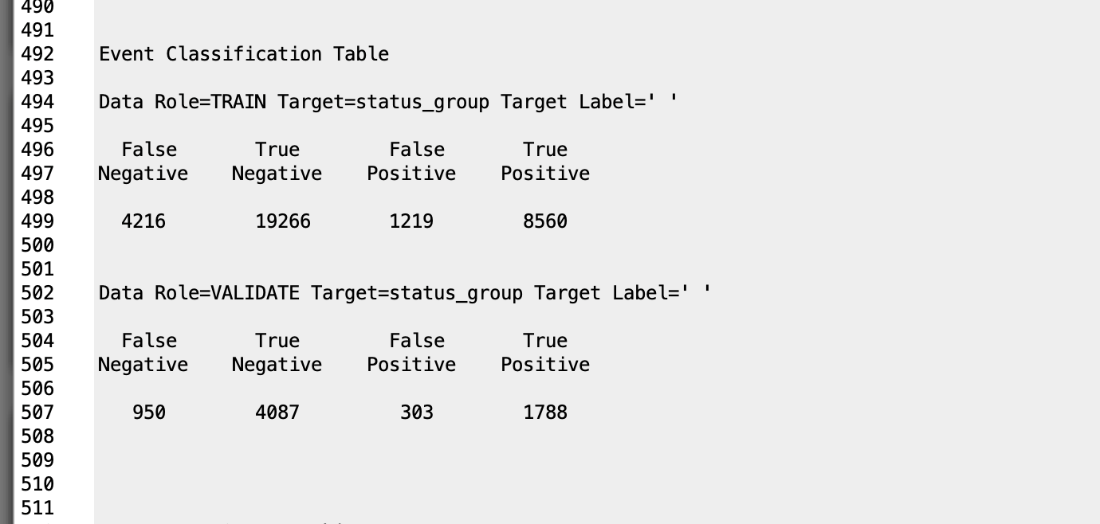
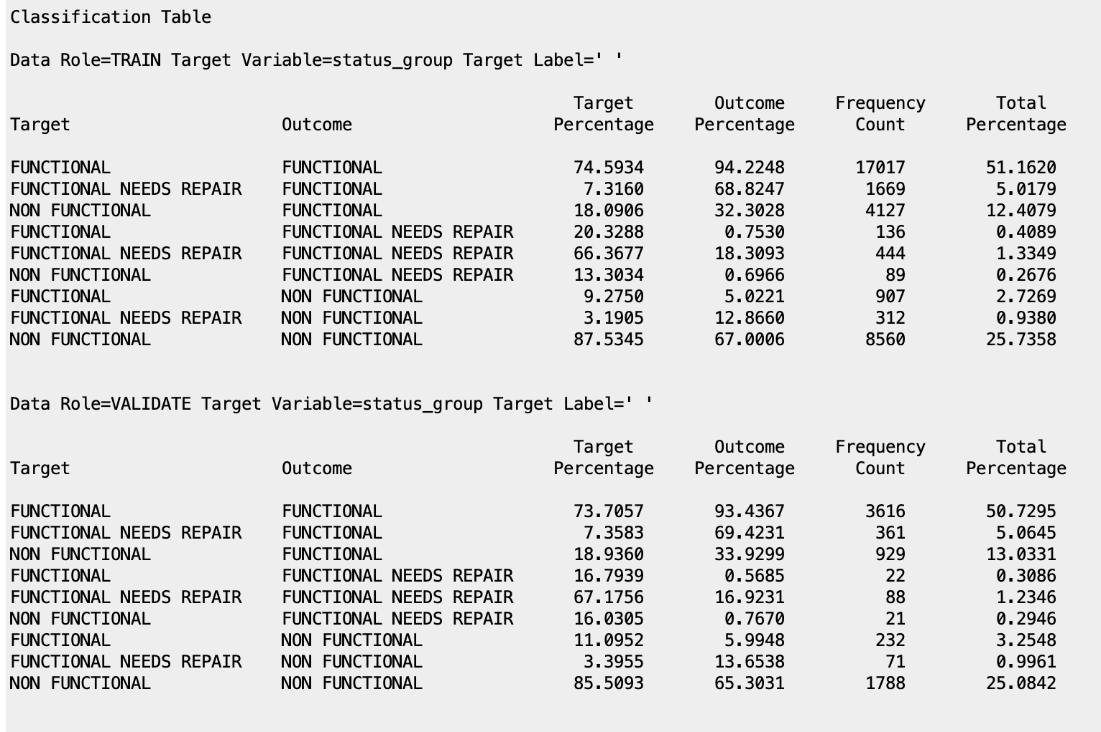
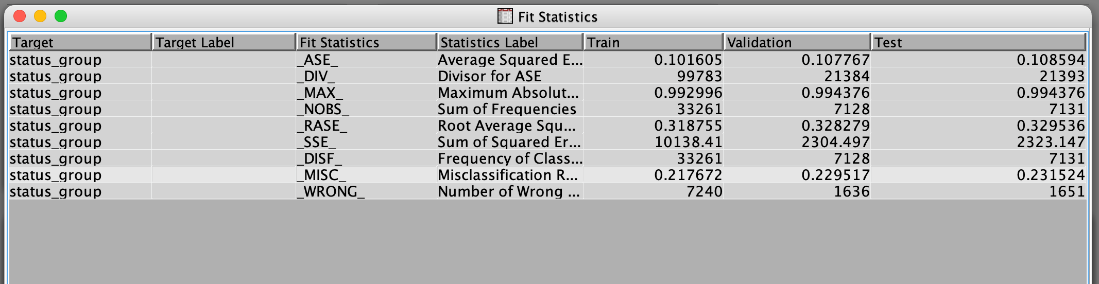
The Random branch assessment method is used to calculate variable importance (margin for a class target and absolute error for an interval target) based on the validation data when available. Hence, selecting the variable importance method to Random assessment.   
  


Fig. Confusion Matrix

The above confusion matrix displays as per training and validation set shows what’s predicted and what’s actual and how many are rightly and how many are wrongly classified.

  
Fig. Classification table for our target variable displaying result as per training and validation set  
  
  
  
Fig. Fit Statistics of HPforest

As we see the misclassification rate here for validation set, it’s lower than the decision and pruned tree, I.e., 0.229517 which makes the accuracy 1- 0.229517 = **77.04%.**

**Impute:**

Decision trees can handle the missing values automatically as they are robust to outliers as well, but Logistic regression and Neural networks are not good at handling the missing values. So, before we perform Logistic regression and Neural networks on our dataset, we need to impute the missing values in the dataset. The missing values in the dataset must be imputed by proper estimation methods like mean, median etc. before running the algorithm.

Diagram

Description automatically generated

Fig.Diagram

In this project, the missing values of the interval variables are imputed with median as it is more robust to outliers. And missing values of the class variables are imputed with tree surrogate. Tree surrogates are like decision trees for a particular class variable. A StatExplore node is used to analyze the output after imputing the variables. After imputing the missing values of the variables, the output of the StatExplorer is as shown below. We can see there are no missing values for the interval variables and skewness of the variables is in between +3 and –3.

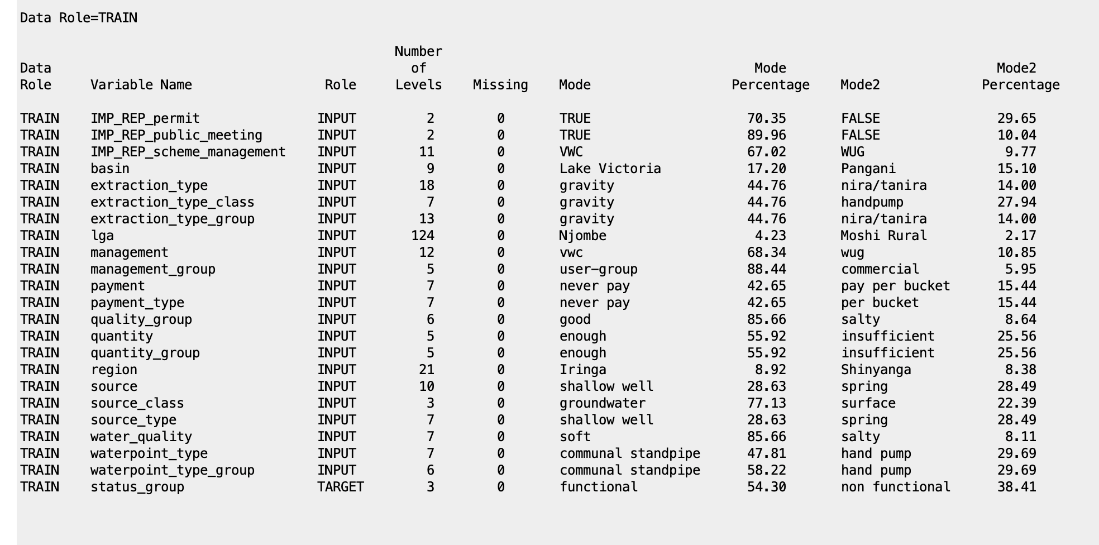


Fig. Output of StatExplorer3 node after imputing (class variables)

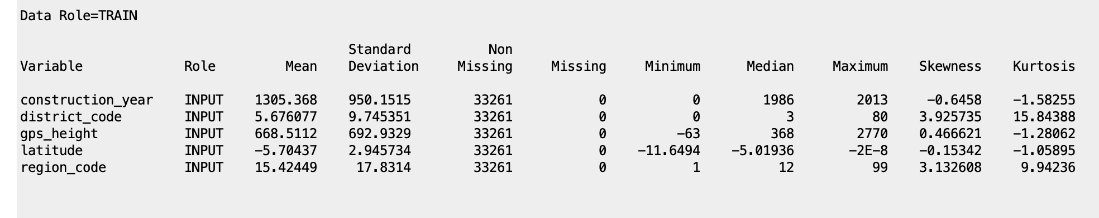


Fig. Output of StatExplorer3 node after imputing (interval variables)  
  
**Model 2. Logistic Regression:**

Logistic regression is a supervised algorithm model used to predict a dependent categorical target variable. It is a statistical analysis method to predict a binary outcome based on prior observations. It can also be used to predict one or more nominal data and that is called multinominal logistic regression. If an item can be classified into multiple classes, then it is represented as an ordinal type.

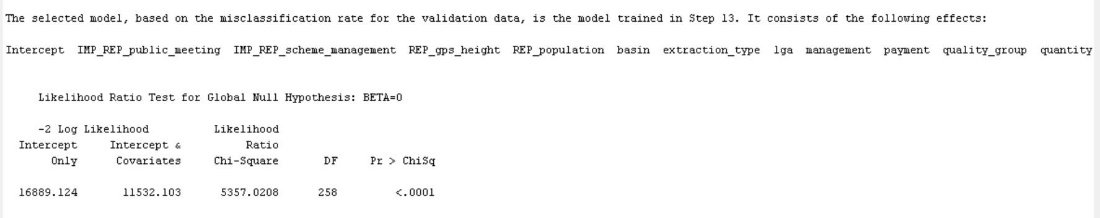
We have three different types of models in logistic regression. They are forward, backward and stepwise regression models. In forward regression, the model analyzes the input variables from top to bottom and rejects the variables at the end. The backward regression is the reverse of the forward process, it analyzes from bottom to top. The stepwise regression model is a combination of both forward and backward. It analyzes the input variables and rejects the unnecessary variables parallelly. In this project, we used a stepwise regression model.

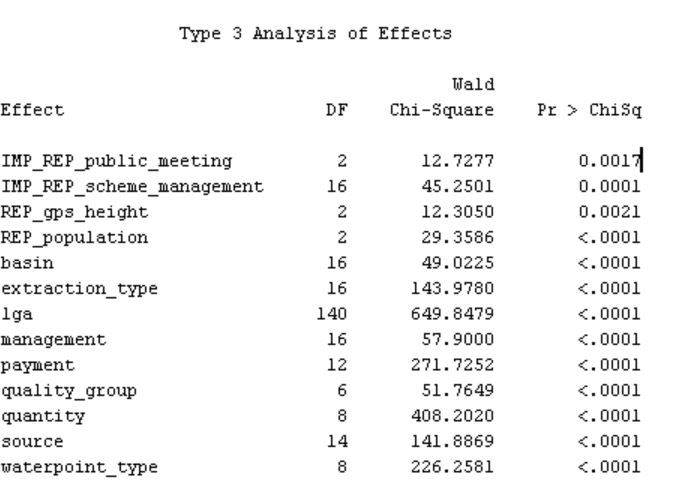
Diagram

Description automatically generated

Fig. Diagram

The output shows which iteration is selected by the model and as we can see all the variables/predictors selected by the model are significant and model. As per the output, the selected model is based on step 13.





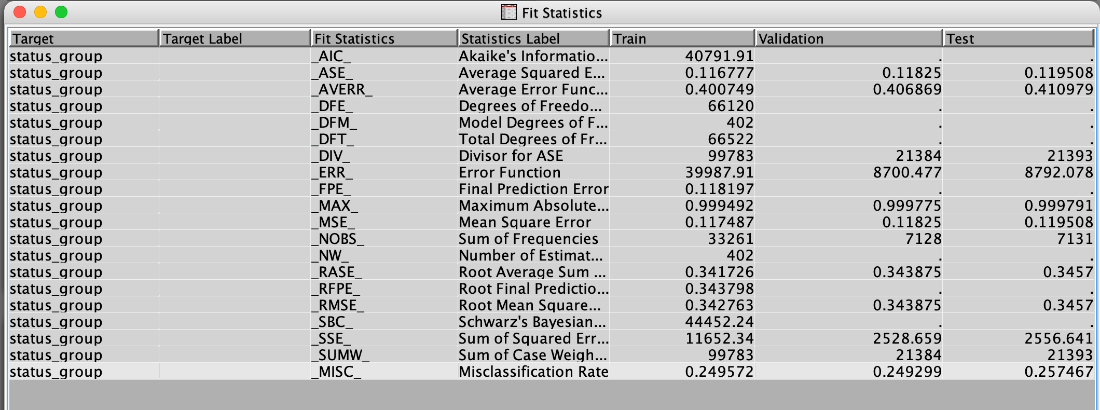


Fig. Fit statistics output of Logistic regression

From the above output of fit statistics, we can see the misclassification rate for training data is 0.249572 so the accuracy is **75.0428%.** The misclassification rate for the validation data set is 0.249299 so the accuracy is **75.0701%.** The misclassification rate for test data is 0.257467 and the accuracy will be **74.2533%.**

**Model 3-Neural Networks:**  
  
Neural networks are a class of flexible nonlinear regression, discriminant, and data reduction models. The Neural Network node provides a variety of feedforward networks that are commonly called backpropagation. Backpropagation refers to the method for computing the error gradient for a feedforward network, a straightforward application of the chain rule of elementary calculus.

Most connections in a network have an associated numeric value called a weight or parameter estimate. The training methods attempt to minimize the error function by iteratively adjusting the values of the weights. The value produced by the combination function is transformed by an activation function, which involves no weights or other estimated parameters.

The Neural Network node also provides a variety of conventional methods for nonlinear optimization that are usually faster and more reliable than the algorithms from the neural network literature. we also standardize the inputs before running the model to get better results.

For Neural network node:

The misclassification rate is 0.2493(24.93%) for training data, 0.2643(26.43%) for validation data and 0.2617(26.17%) for test data. That means the accuracy for validation set is **73.83%**

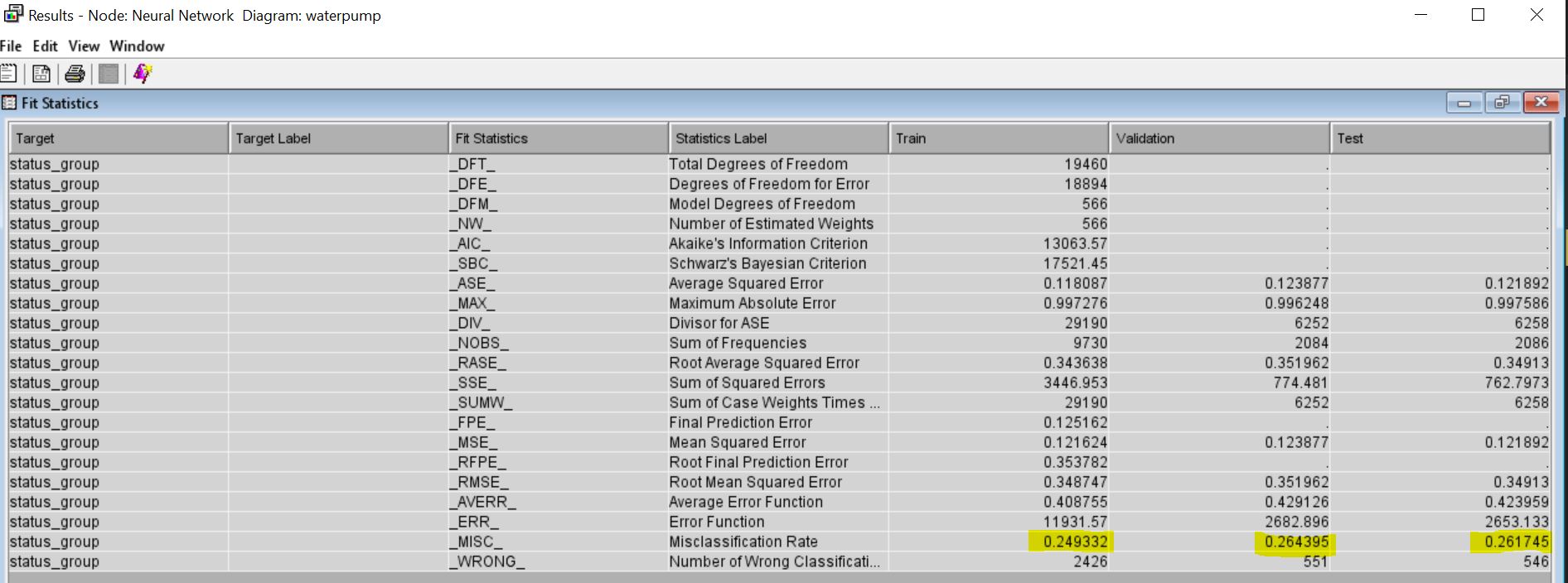


Fig: Fit statistics of neural networks

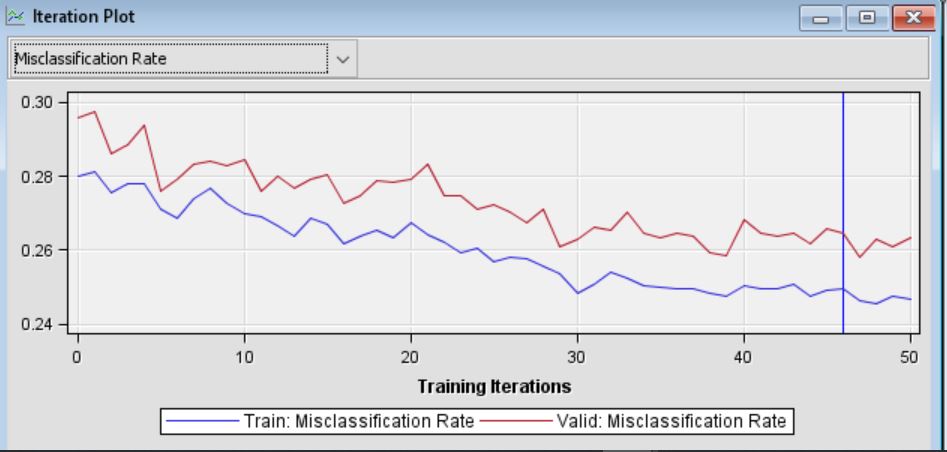


Fig: Iteration plot of misclassification rate

We can also observe the iteration plot for misclassification rate. From the above graph, we can observe that the iteration point is selected at point 46 to avoid overfitting of the model.

For Auto neural network node:

In the neural network model, it undergoes only a single hidden layer and no iterations. So, we prefer auto-neural networks to select a greater number of iterations and more hidden layers. More the hidden layers, the more accuracy and performance of the model.

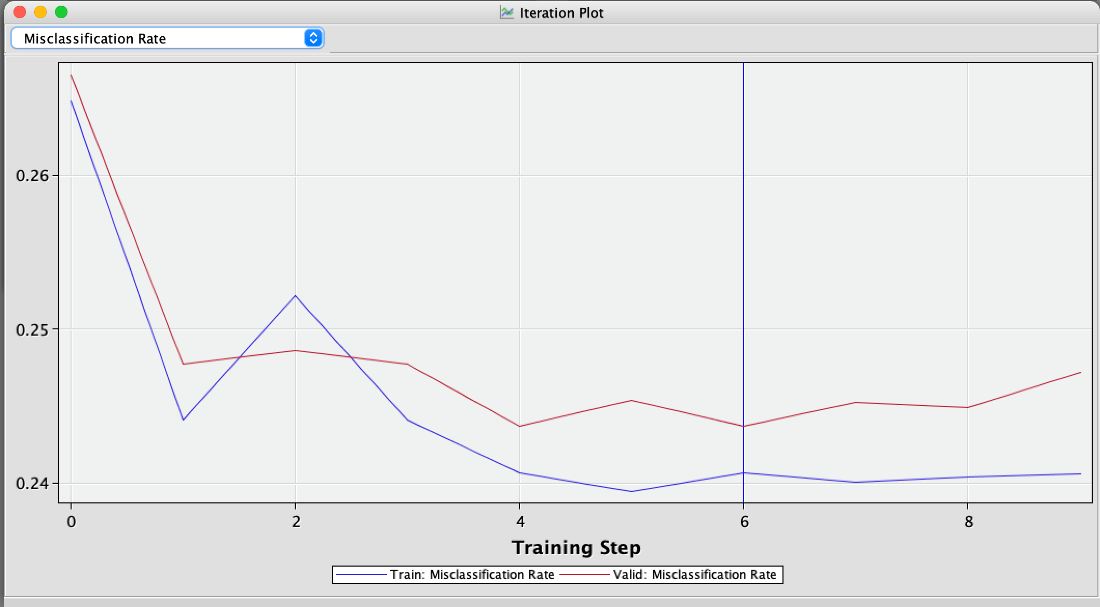


Fig: Iteration plot for Misclassification rate of Auto-Neural node

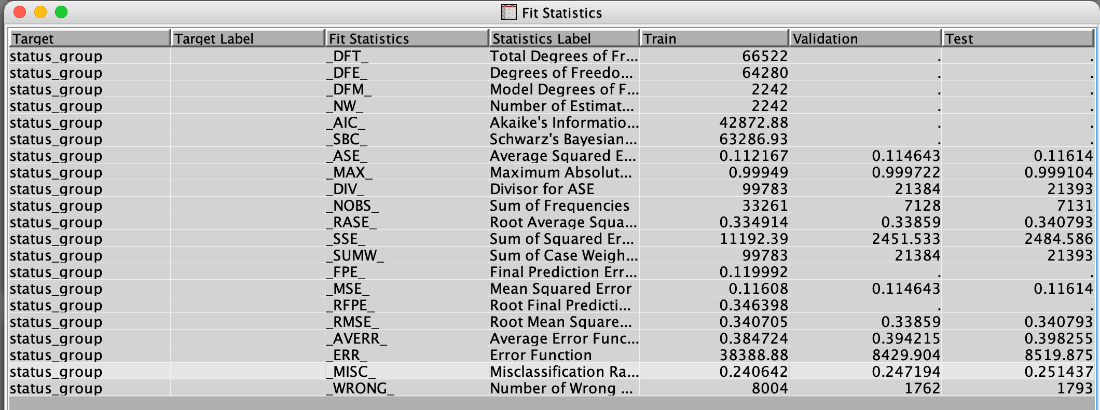


Fig. Fit statistics of Auto neural node

The misclassification rate for validation set of Autoneural network is 0.247194(24.71%). So, the accuracy is 1- 0.247194 i.e., **75.28%** for Autoneural Network.

**Conclusion:**

**Model Comparision:**

The model comparison node is used to assess which model is best as per validation classification rate.

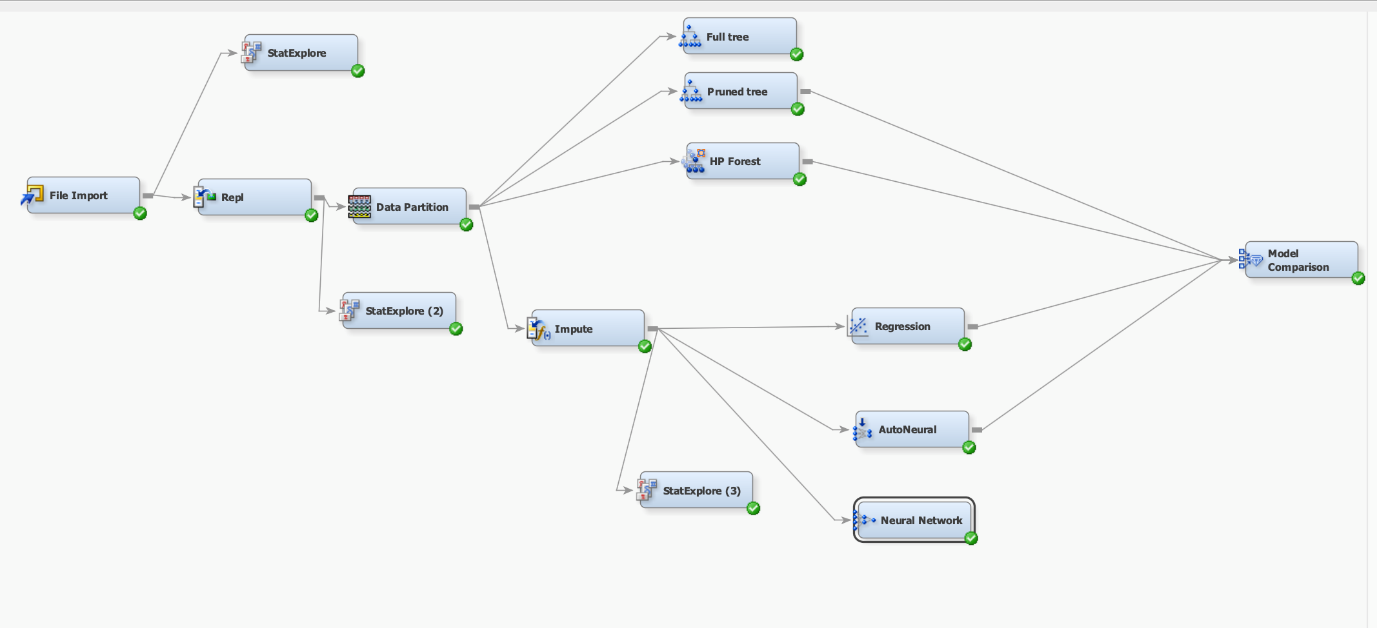


Fig. Diagram

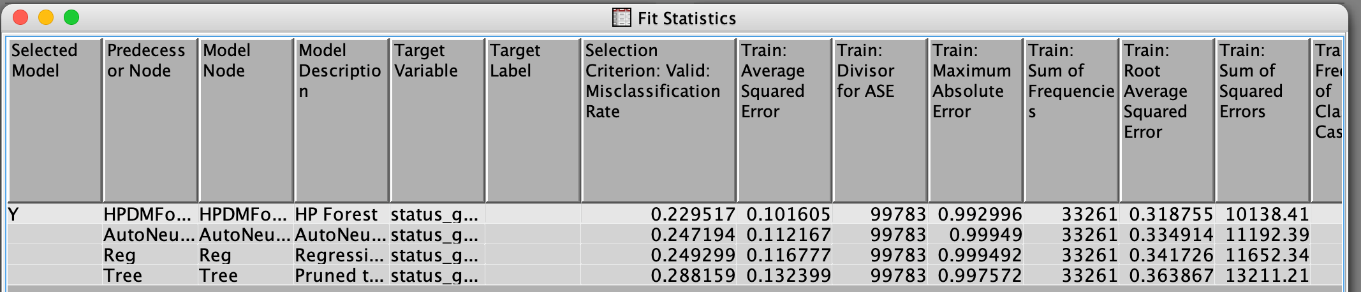


Fig: Fit Statistics of Model Comparison Node

The misclassification rates and accuracy of the models HPForest, Autoneural, Logistic regression and Decision tree are as follows

**HPForest** **AutoNeural** **Logistic regression** **Decision tree**

**Misclassification**  **22.95%**  **24.71%**  **24.92%** **28.85%**  
**rates%**

**Accuracy%** **77.05%**  **75.3%. 75.08%** **71.15%**

As we can see from the above statistics, we can see that HPForest accuracy is higher than all other models, and the model comparison node selected HPForest as shown in the above fit statistics fig.

Based on the model comparison node, we see that the misclassification rate for HP Forest algorithm is lowest, and Accuracy is higher among Decision/Prune tree, Logistic Regression and Neural Network.

Therefore, HP forest is the best model in our case.