Assignment 3

Name: Vishesh Kalsi

SID: 13731652

Description

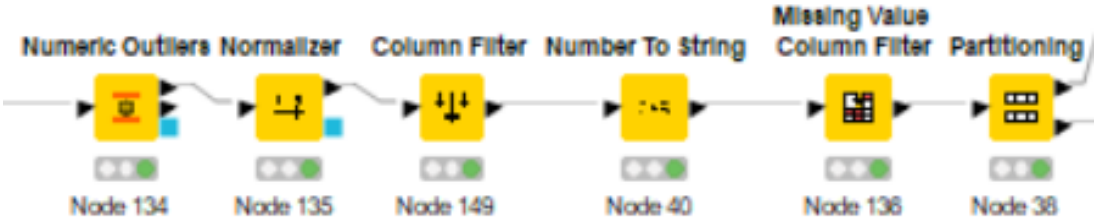
In the assigned task we were given datasets to explore and perform analytics on the same. We are provided with 3 datasets. The first one being a Training dataset using which we are supposed to make a model. The second one is a Unknown dataset which is basically a test dataset which needs to be predicted using the model obtained by the first one. The final one is a Kaggle submission random sample dataset which is an example of the required format of submission. The datasets given to us consists of a total of 38 attributes out of which one is 'Qualified' which we are supposed to predict. The datasets are considered as the inputs given to us whereas the final submission that we are supposed to do on Kaggle is considered as the output. We can make a total of 3 submissions for the competition of Kaggle.

How you went about solving the problem

After downloading the given dataset, I opened it using Excel and started to look through it. I then found few things regarding it but was not able to get a greater insight into it because of lack of options available on Excel. After that I imported the dataset into KNIME Analytics Platform and read the dataset into a file reader. I then started to play around with the different rows and columns present. I preprocessed the data to make it smooth and less redundant for the various classifiers that I put my hands on. Following that I got to know about my best classifier that was Random Forest Learner with an accuracy of 90.861% with the training dataset provided to me. The preprocessing and the different classifiers used during the process are explained in detail in upcoming sections.

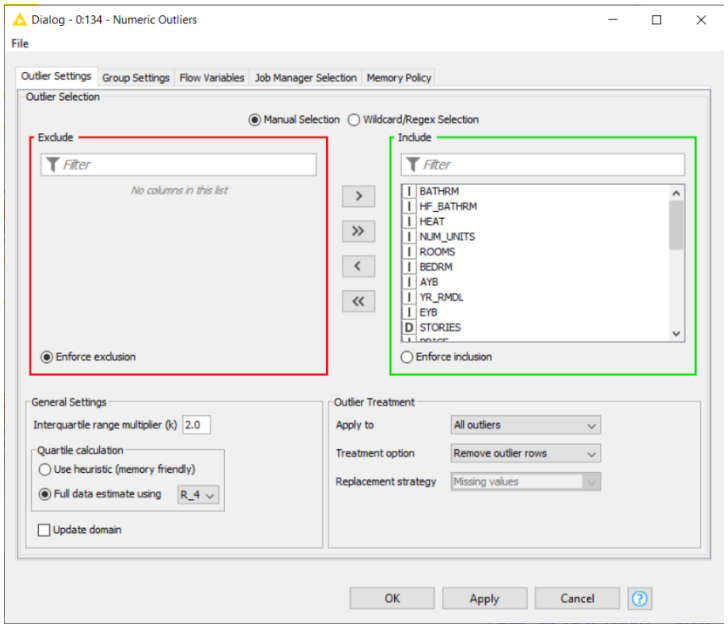
Data preprocessing and Transformation

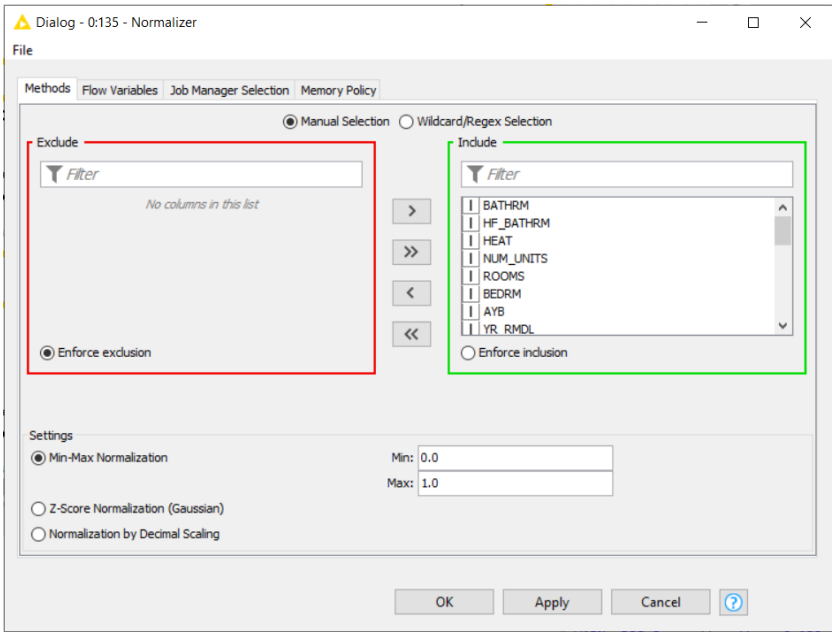
In this section I first read the given dataset: Training dataset into file reader and then did preprocessing differently for each of the classifiers used. Mainly there were a total of 6



nodes that I used for transforming the dataset and doing the preprocessing. These 6 nodes consisted of:

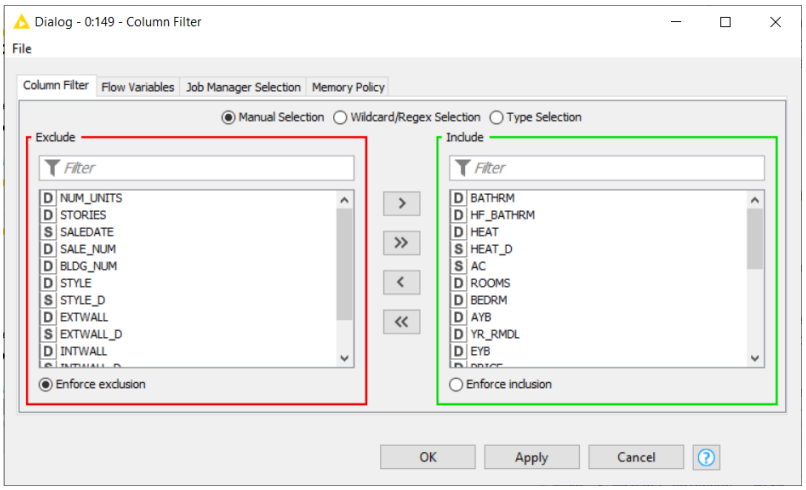
These nodes proved to be quite useful for me and helped me deal with the inconsistency of the given dataset and removed most of the errors from it. The node Numeric Outliers helps us to remove outliers based on Interquartile range multiplier(k) and by varying between different quartile calculation methods. When we remove the outliers then our dataset gives us results that are statistically more significant.

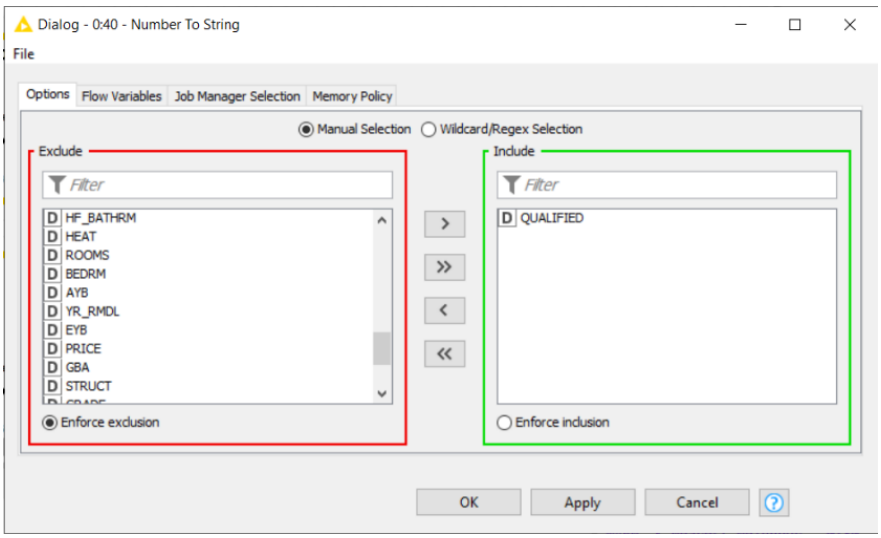




The next node I used was the Normalizer node which normalizes the dataset. It gives us a option to normalize the dataset by 2 methods mainly which are Min-Max Normalization and Z-Score Normalization. When our dataset is normalized it helps by increasing overall database organization, reducing redundant data, and making the database's design much more flexible. After normalizing the dataset, the overall speed of processing the data also becomes faster.

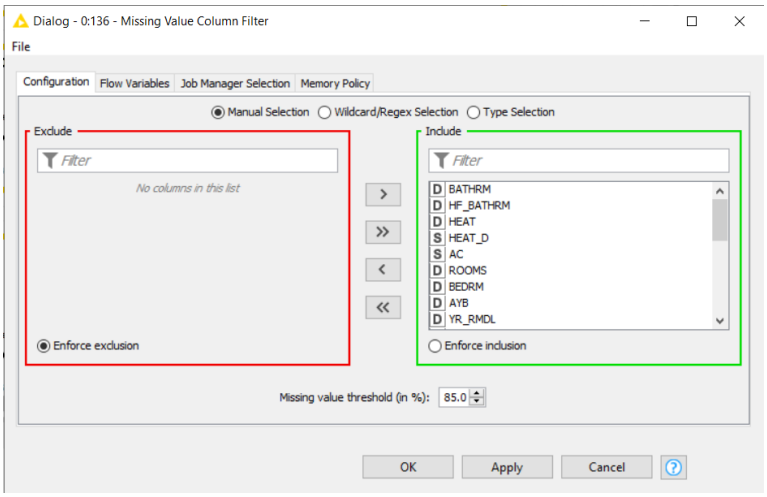
After this, I used the column filter node. It helped me to remove those columns which did not play any important role in the dataset.

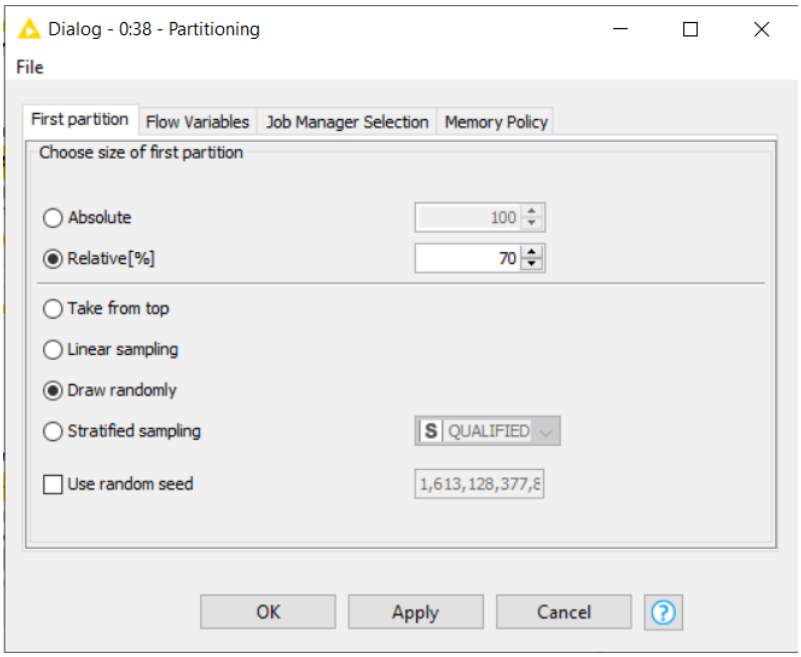




The next node I used following this was the Number to string Node which helped me convert the data on the "Qualified" column from number to a string value. I did this as the classifier can only predict on string values. Hence, our final task is to predict this column and of this would not have had been in string form then no prediction would have been possible.

Following this the next node used in the process was Missing value column filter. This helps us to remove the missing values from the columns with a variable threshold value. I had set this value as 85%. This node removes all columns from the input table which contain more missing values 85%. The filtering is only applied to the columns in the input list of the column filter panel.



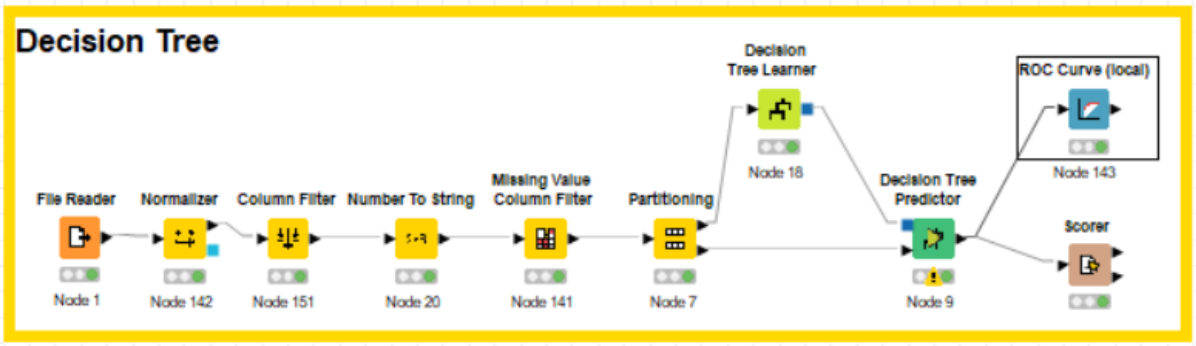


The last but not the least node that was used was a Partitioning node which spits the dataset into 2 partitions (i.e., row-wise), e.g., train and test data. The two partitions are available at the two output ports. The first one i.e., train set is sent to the learner of any model whereas the second one i.e., test set is sent to the test set. I had split the dataset into 70:30 i.e., 70% for training of the model and 30% for the test of the dataset.

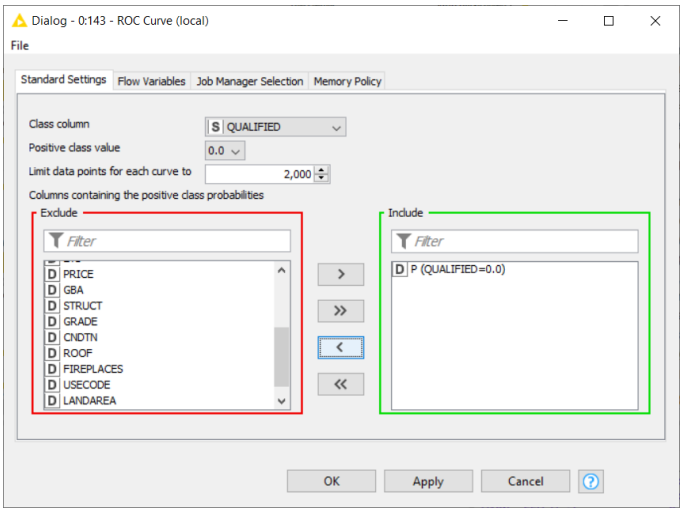
Covering across the 5 total classifiers used by me I did use all of them in 3 classifiers that were: Random Forest Learner, SVM and Multilayer Perceptron. In KNN I did not use the Normalizer node as it was decreasing the accuracy of my model. In Decision Tree I did not use the numeric outlier node as it was decreasing the accuracy of my model.

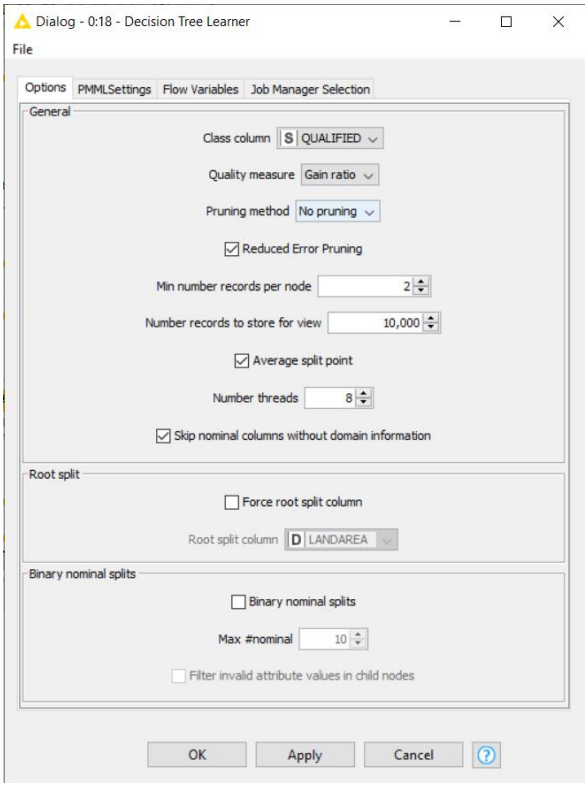
Different classification techniques used

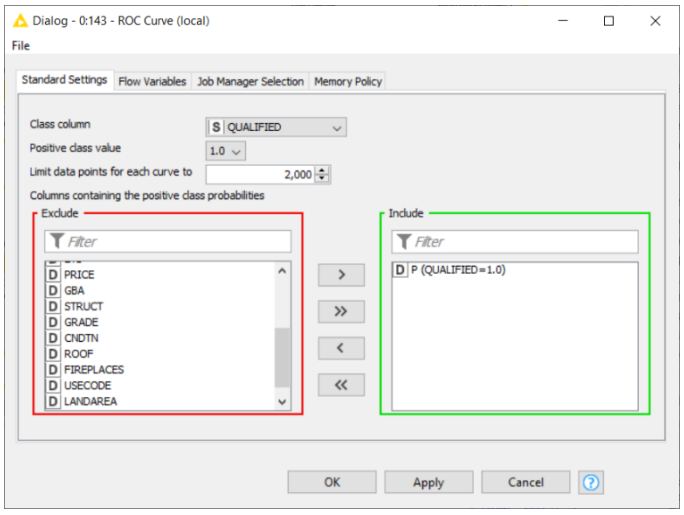
Over the course of the project, I have used a total of 5 classifiers and the **first one** being Decision Tree. After completing the pre-processing, I attached the obtained data to Decision Tree learner and Decision Tree predictor. The majority part was sent to the learner whereas the minority was sent to the predictor.

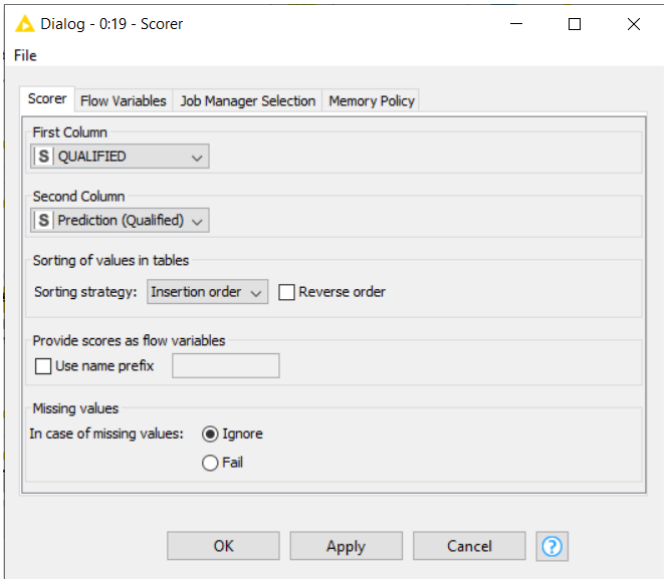


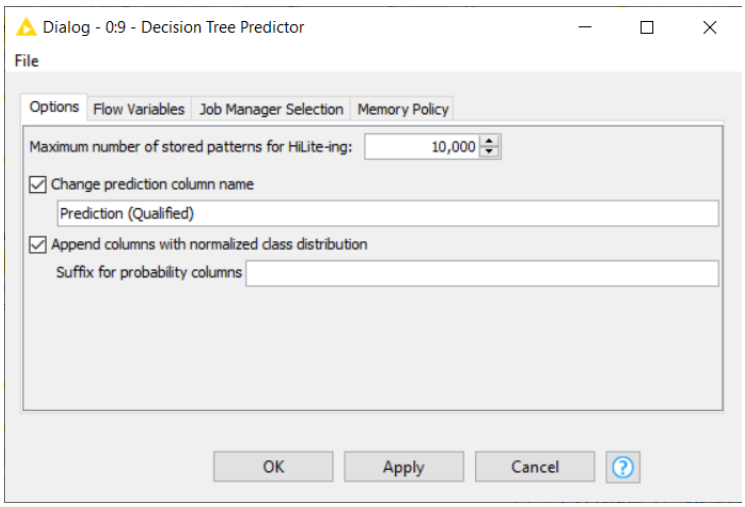
After this I attached a scorer and a ROC Curve to the data obtained from the predictor to calculate the accuracy of my model. I found out this model gave me an overall accuracy of 88.921%. Whereas the AUC for the ROC Curve was 0.9027 for both 0 and 1.

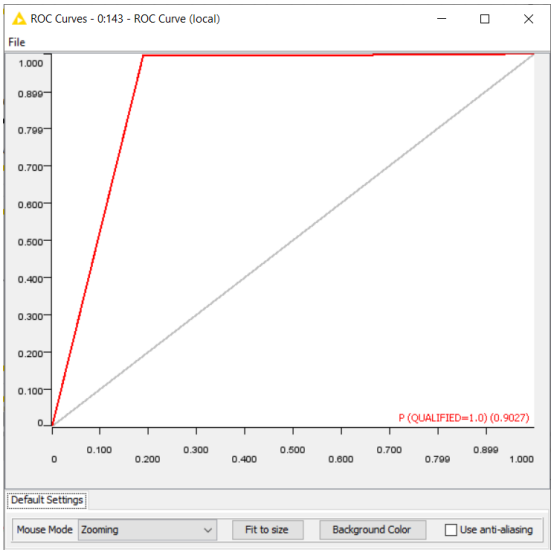


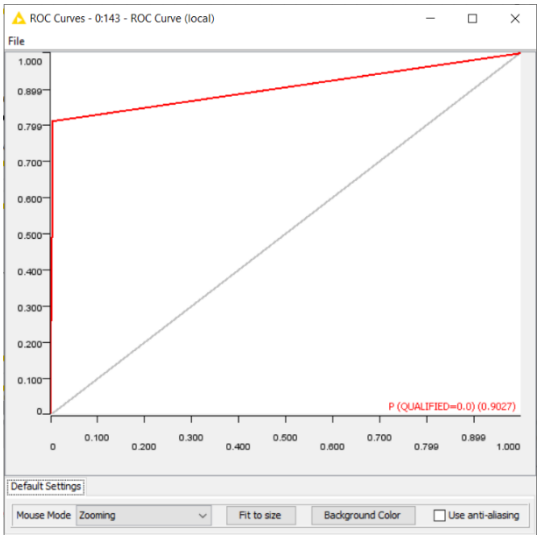




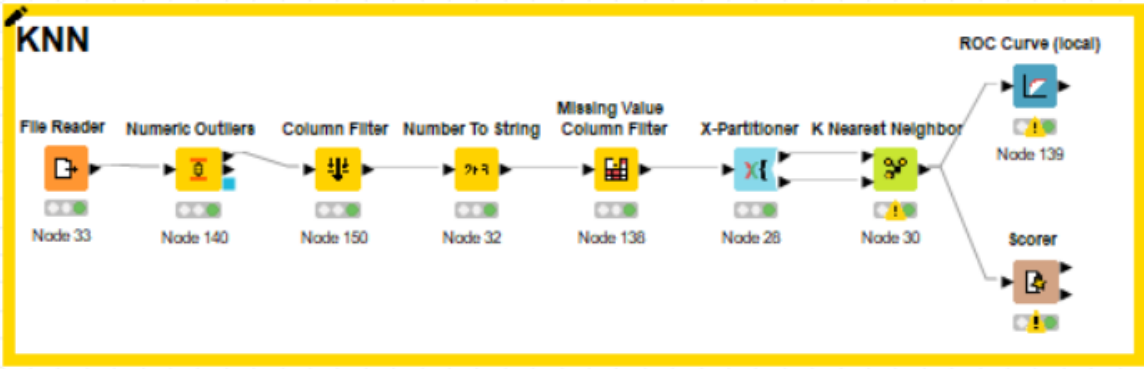




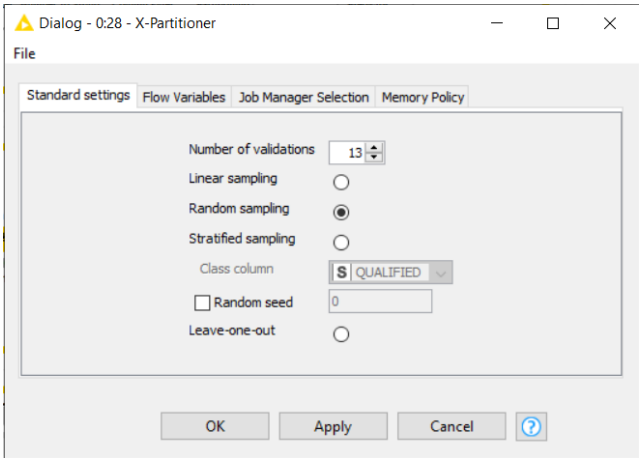


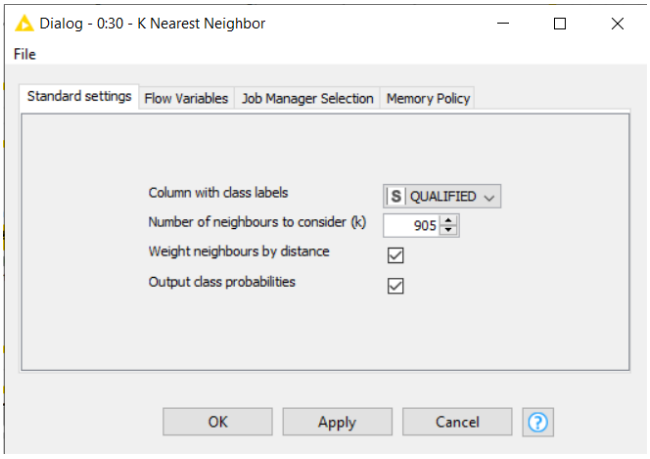


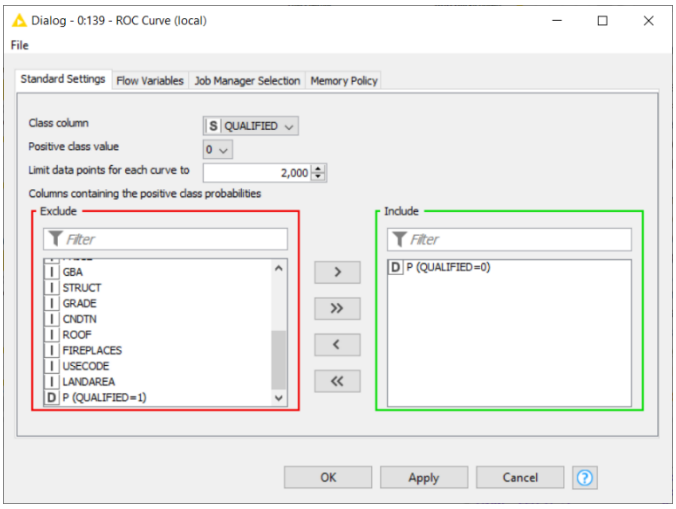
The **second classifier** that I made was KNN. After completing the pre-processing, I attached the obtained data to X-Partitioner along with K Nearest Neighbor. I used 13 validations for the X-Partitioner as this gave me the highest accuracy for my model. Both the test and training set was taken care by X-Partitioner and sent to the K-Nearest Neighbor. After this I attached a scorer and a ROC Curve to the data obtained from the predictor to calculate the accuracy of my model.

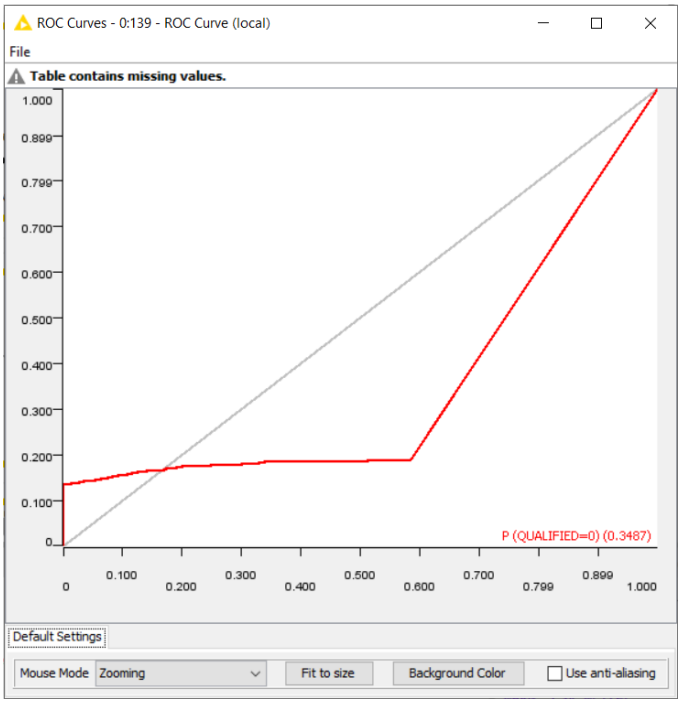


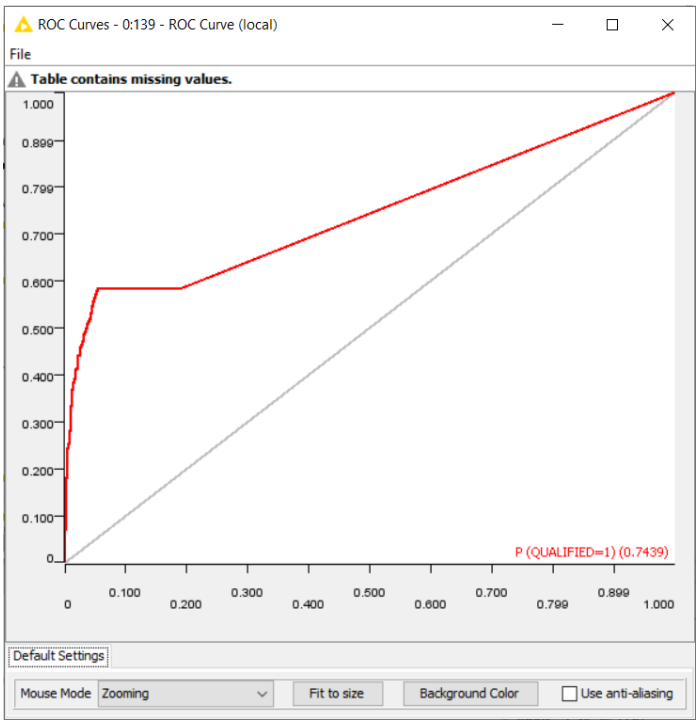
I found out this model gave me an overall accuracy of 87.5%. Whereas the AUC for the ROC Curve was 0.7439 and 0.3487 for 1 and 0, respectively.

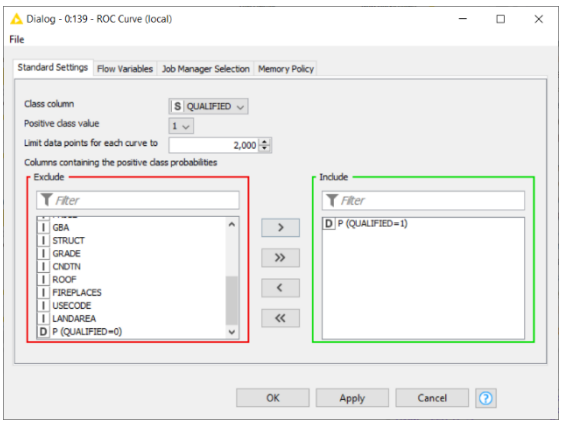

Dialog - 0:28 - X-Partitioner
File
Standard settings
Flow Variables
Job Manager Selection
Memory Policy
Number of validations
13
Linear sampling
Random sampling
Stratified sampling
Class column
QUALIFIED
Random seed
0
Leave-one-out
OK
Apply
Cancel

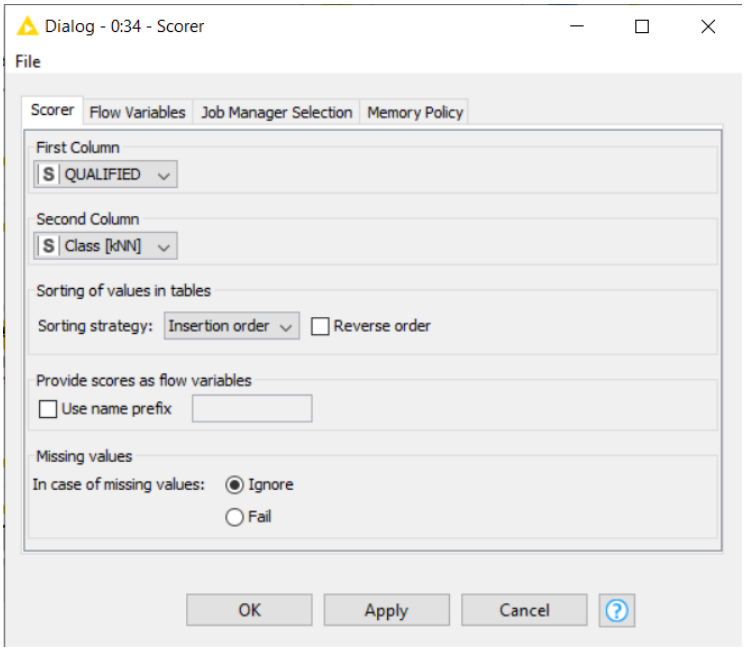

Dialog - 0:30 - K Nearest Neighbor
File
Standard settings
Flow Variables
Job Manager Selection
Memory Policy
Column with class labels
QUALIFIED
Number of neighbours to consider (k)
905
Weight neighbours by distance
Output class probabilities
OK
Apply
Cancel

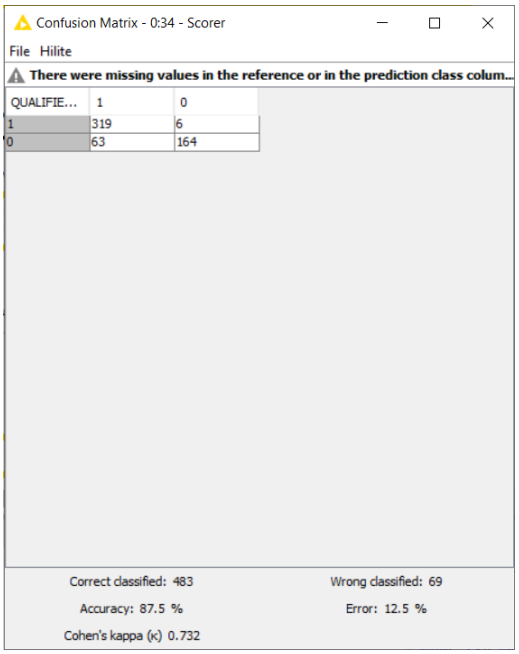

Dialog - 0:139 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
QUALIFIED
Positive class value
0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
GBA
STRUCT
GRADE
CNDTN
ROOF
FIREPLACES
USECODE
LANDAREA
P (QUALIFIED=1)
Include
Filter
P (QUALIFIED=0)
OK
Apply
Cancel


ROC Curves - 0:139 - ROC Curve (local)
File
Table contains missing values.
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=0) (0.3487)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing

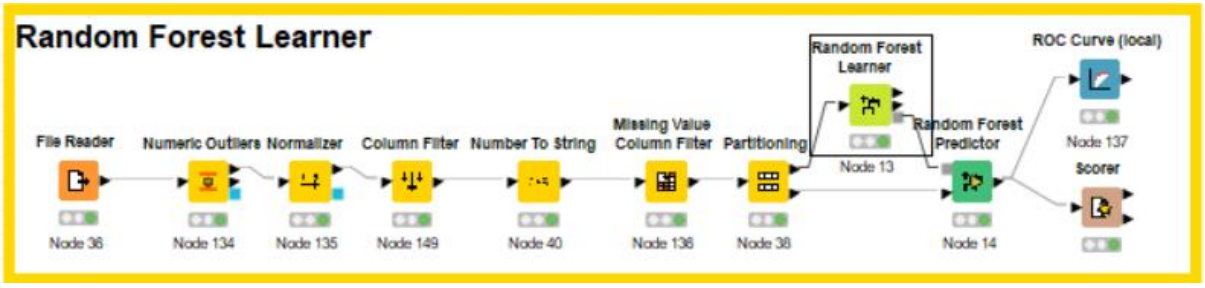

ROC Curves - 0:139 - ROC Curve (local)
File
Table contains missing values.
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=1) (0.7439)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing


Dialog - 0:139 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
QUALIFIED
Positive class value
1
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
GBA
STRUCT
GRADE
CNDTN
ROOF
FIREPLACES
USECODE
LANDAREA
P (QUALIFIED=0)
Include
Filter
P (QUALIFIED=1)
OK
Apply
Cancel

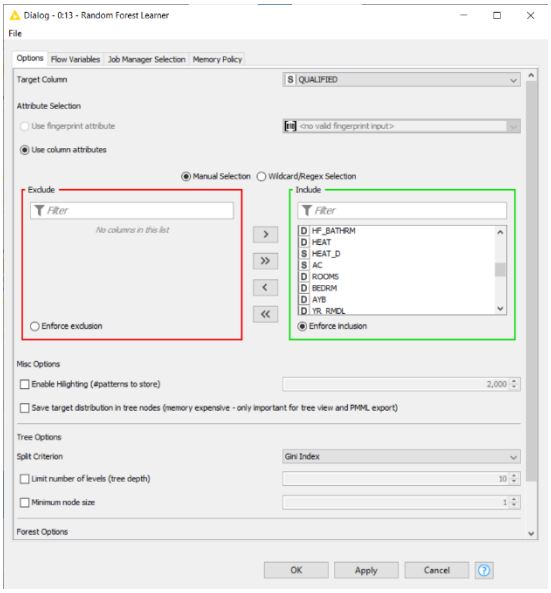


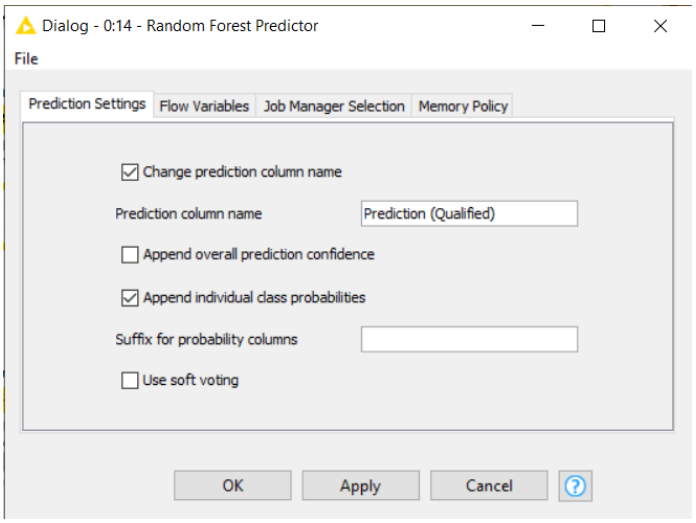


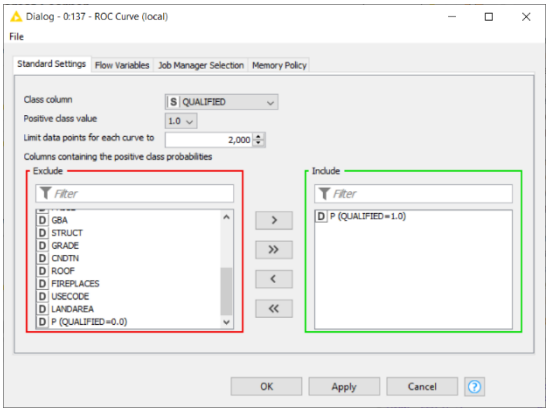
The **third classifier** that I used was Random Forest Learner. After completing the pre-processing, I attached the obtained data to Random Forest Learner along with Random Forest Predictor. The majority part was sent to the learner whereas the minority was sent to the predictor. I used the split criterion for the learner as Gini Index as it gave me a higher accuracy for my model in comparison to Information Gain and Information Gain Ratio. After this I attached a scorer and a ROC Curve to the data obtained from the predictor to calculate the accuracy of my model.

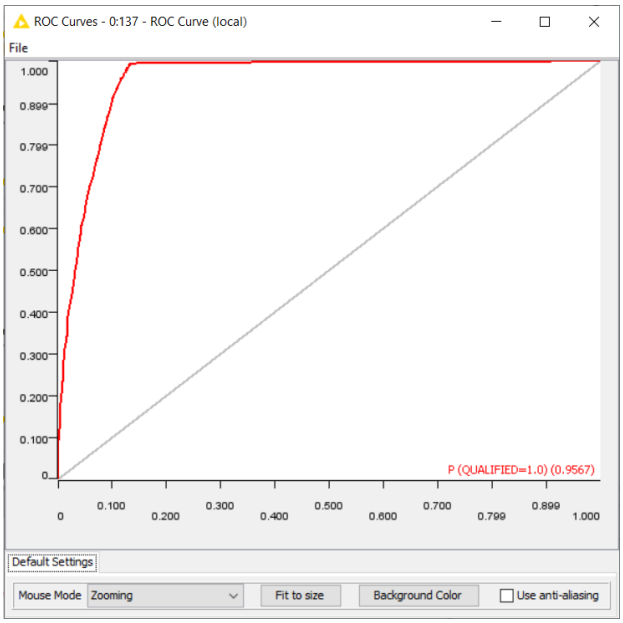


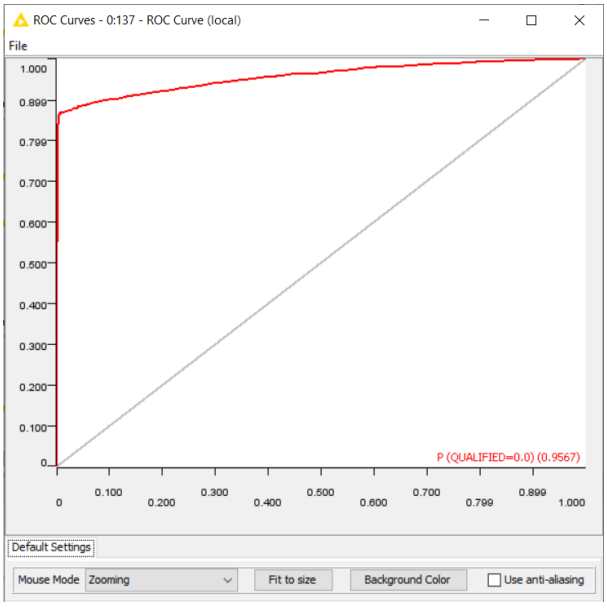
I found out this model gave me an overall accuracy of 90.861%. Whereas the AUC for the ROC Curve was 0.9567 for both 1 and 0.

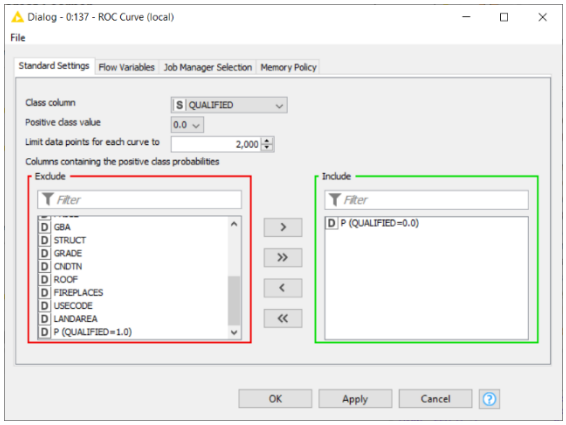

Dialog - 0:13 - Random Forest Learner
File
Options
Flow Variables
Job Manager Selection
Memory Policy
Target Column
QUALIFIED
Attribute Selection
Use fingerprint attribute
Use column attributes
Manual Selection
Wildcard/Regex Selection
Exclude
Filter
No columns in this list
Enforce exclusion
Include
Filter
HF_BATHRM
HEAT
HEAT_D
AC
ROOMS
BEDRM
AYB
YR_RMDL
Enforce inclusion
Misc Options
Enable Hilighting (#patterns to store)
2,000
Save target distribution in tree nodes (memory expensive - only important for tree view and PMML export)
Tree Options
Split Criterion
Gini Index
Limit number of levels (tree depth)
10
Minimum node size
1
Forest Options
OK
Apply
Cancel


Dialog - 0:14 - Random Forest Predictor
File
Prediction Settings
Flow Variables
Job Manager Selection
Memory Policy
Change prediction column name
Prediction column name
Prediction (Qualified)
Append overall prediction confidence
Append individual class probabilities
Suffix for probability columns
Use soft voting
OK
Apply
Cancel

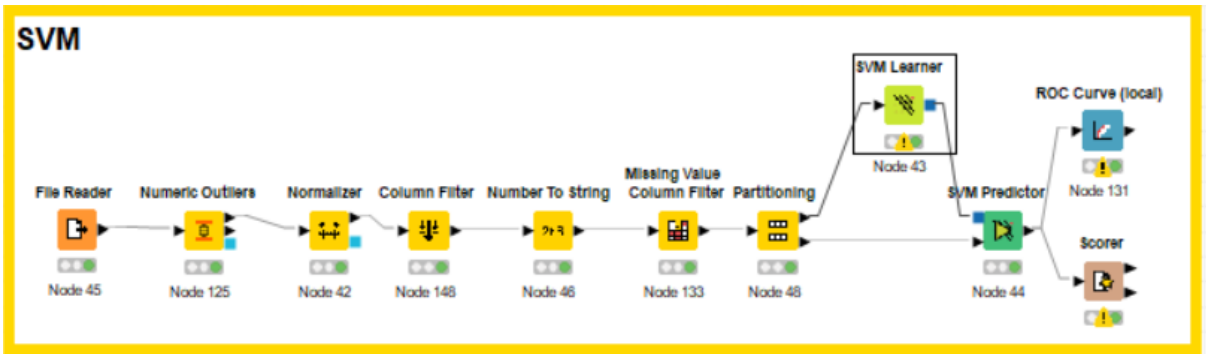

Dialog - 0:137 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
QUALIFIED
Positive class value
1.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
GBA
STRUCT
GRADE
CNDTN
ROOF
FIREPLACES
USECODE
LANDAREA
P (QUALIFIED=0.0)
Include
Filter
P (QUALIFIED=1.0)
OK
Apply
Cancel


ROC Curves - 0:137 - ROC Curve (local)
File
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=1.0) (0.9567)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing

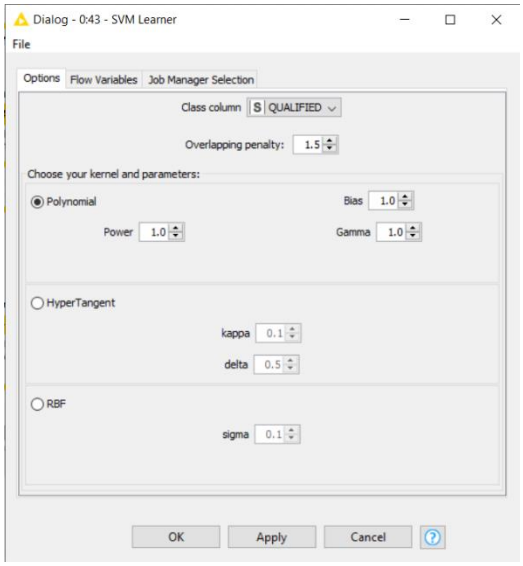

ROC Curves - 0:137 - ROC Curve (local)
File
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=0.0) (0.9567)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing

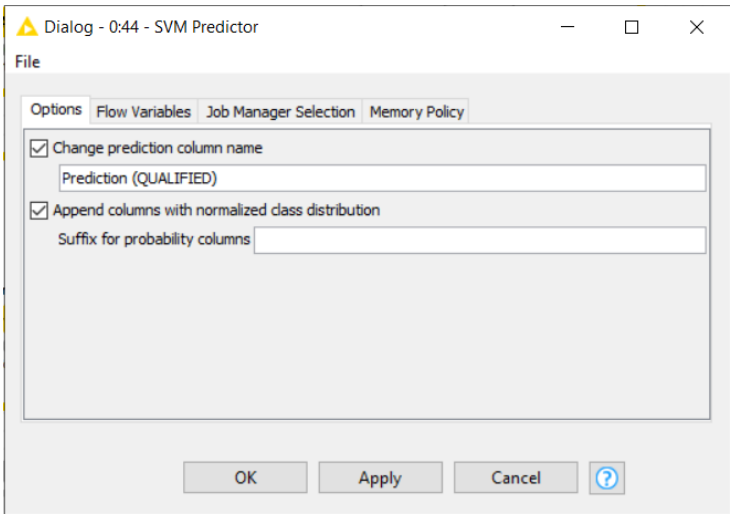

Dialog - 0:137 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
QUALIFIED
Positive class value
0.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
GBA
STRUCT
GRADE
CNDTN
ROOF
FIREPLACES
USECODE
LANDAREA
P (QUALIFIED=1.0)
Include
Filter
P (QUALIFIED=0.0)
OK
Apply
Cancel

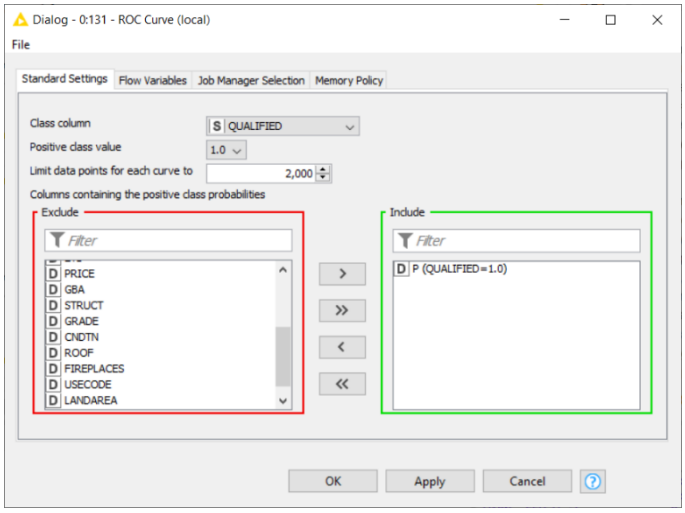
The **fourth classifier** that I used was SVM learner. After completing the pre-processing, I attached the obtained data to SVM Learner along with SVM Predictor. The majority part was sent to the learner whereas the minority was sent to the predictor. I used the split criterion for the learner as Polynomial as it gave me a higher accuracy for my model in comparison to HyperTangent and RBF. After this I attached a scorer and a ROC Curve to the data obtained from the predictor to calculate the accuracy of my model

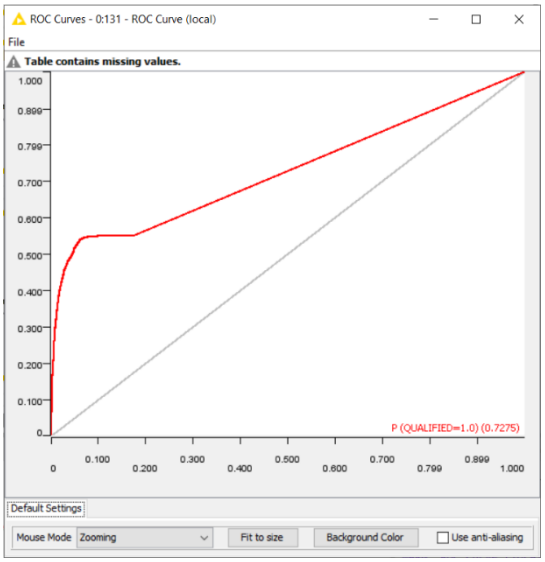


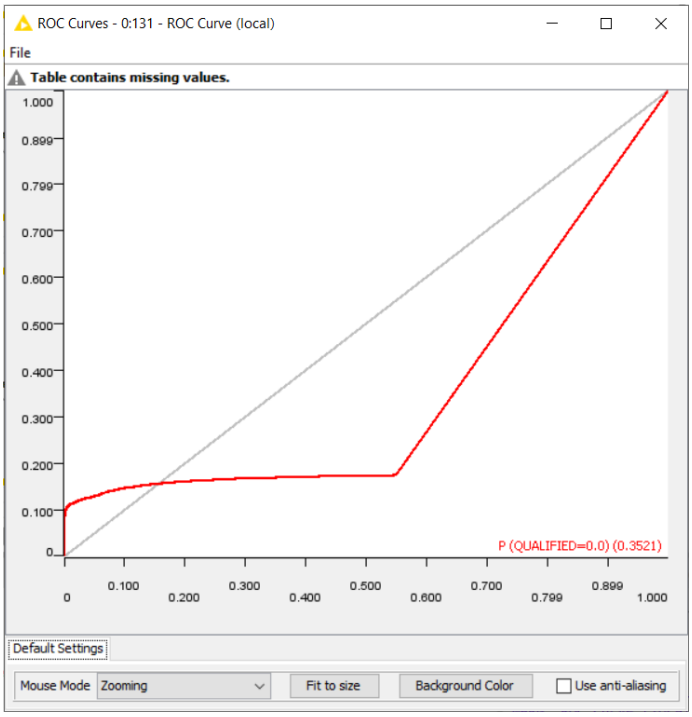
I found out this model gave me an overall accuracy of 84.474%. Whereas the AUC for the ROC Curve was 0.7275 and 0.3521 for 1 and 0, respectively.

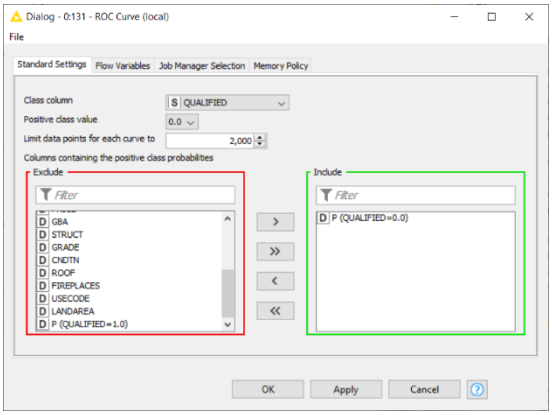


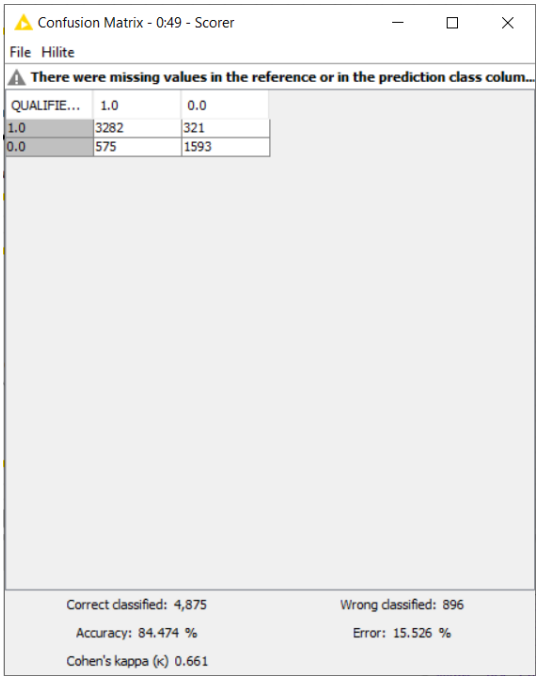


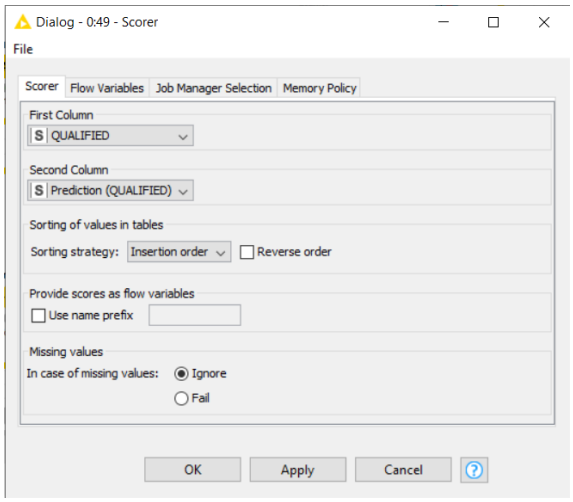

Dialog - 0:131 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
S QUALIFIED
Positive class value
1.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
D PRICE
D GBA
D STRUCT
D GRADE
D CNDTN
D ROOF
D FIREPLACES
D USECODE
D LANDAREA
Include
Filter
D P (QUALIFIED=1.0)
OK
Apply
Cancel


ROC Curves - 0:131 - ROC Curve (local)
File
Table contains missing values.
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=1.0) (0.7275)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing

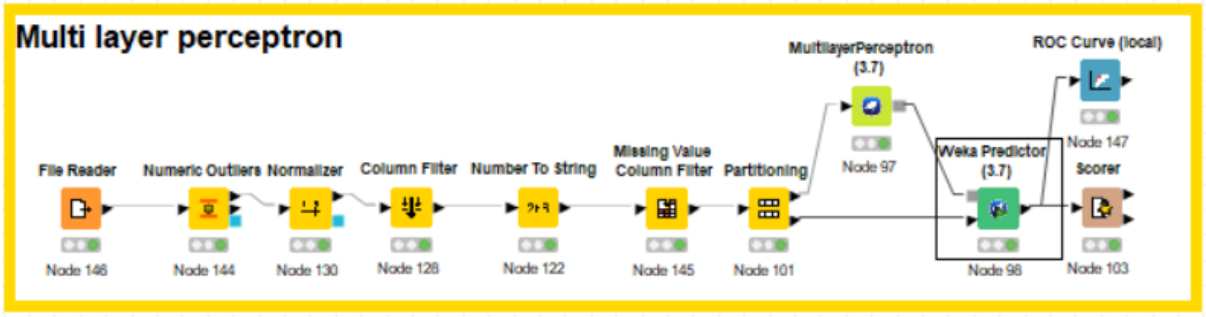

ROC Curves - 0:131 - ROC Curve (local)
File
Table contains missing values.
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=0.0) (0.3521)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing


Dialog - 0:131 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
S QUALIFIED
Positive class value
0.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
D GBA
D STRUCT
D GRADE
D CNDTN
D ROOF
D FIREPLACES
D USECODE
D LANDAREA
D P (QUALIFIED=1.0)
Include
Filter
D P (QUALIFIED=0.0)
OK
Apply
Cancel

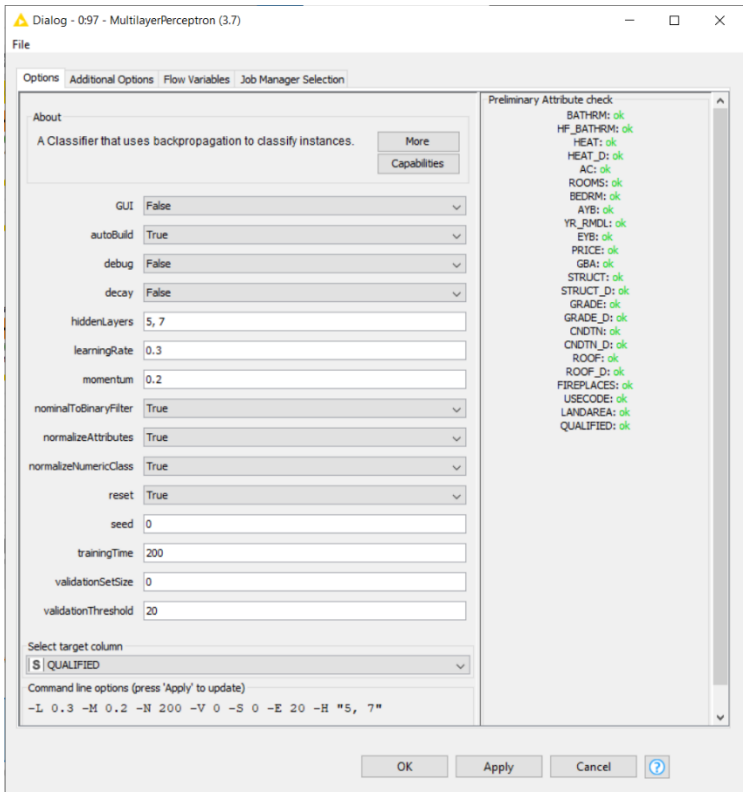

Confusion Matrix - 0:49 - Scorer
File
Hilite
There were missing values in the reference or in the prediction class colum...
QUALIFIE...
1.0
0.0
1.0
3282
321
0.0
575
1593
Correct classified: 4,875
Wrong classified: 896
Accuracy: 84.474 %
Error: 15.526 %
Cohen's kappa (κ) 0.661

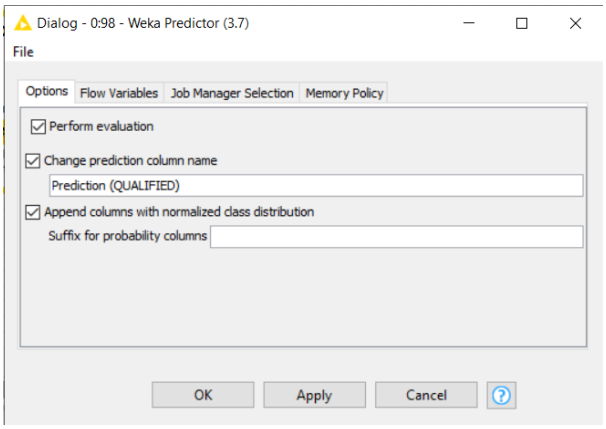

Dialog - 0:49 - Scorer
File
Scorer
Flow Variables
Job Manager Selection
Memory Policy
First Column
S QUALIFIED
Second Column
S Prediction (QUALIFIED)
Sorting of values in tables
Sorting strategy:
Insertion order
Reverse order
Provide scores as flow variables
Use name prefix
Missing values
In case of missing values:
Ignore
Fail
OK
Apply
Cancel

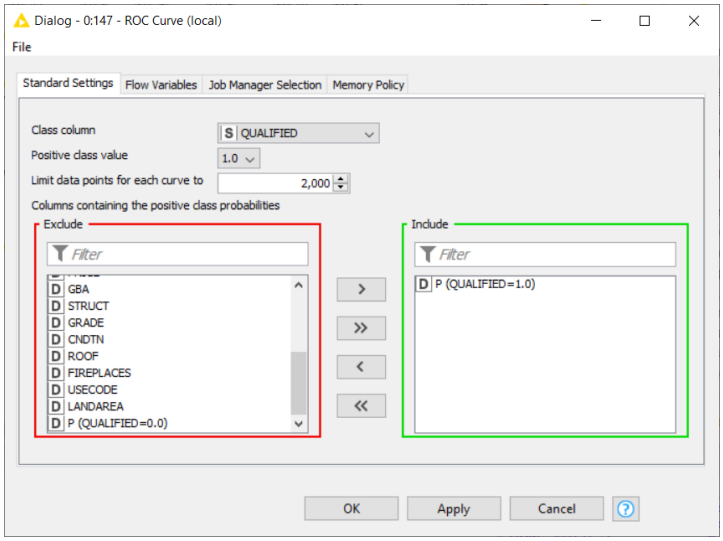
The **fifth classifier** that I used was a Multi-Layer Perceptron. After completing the pre-processing, I attached the obtained data to Multilayer Perceptron Learner along with Weka Predictor. The majority part was sent to the learner whereas the minority was sent to the predictor. After this I attached a scorer and a ROC Curve to the data obtained from the predictor to calculate the accuracy of my model.

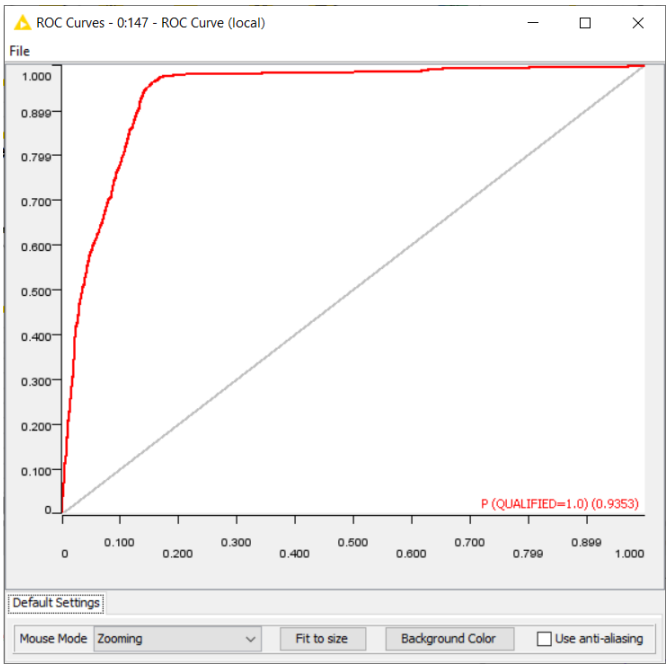


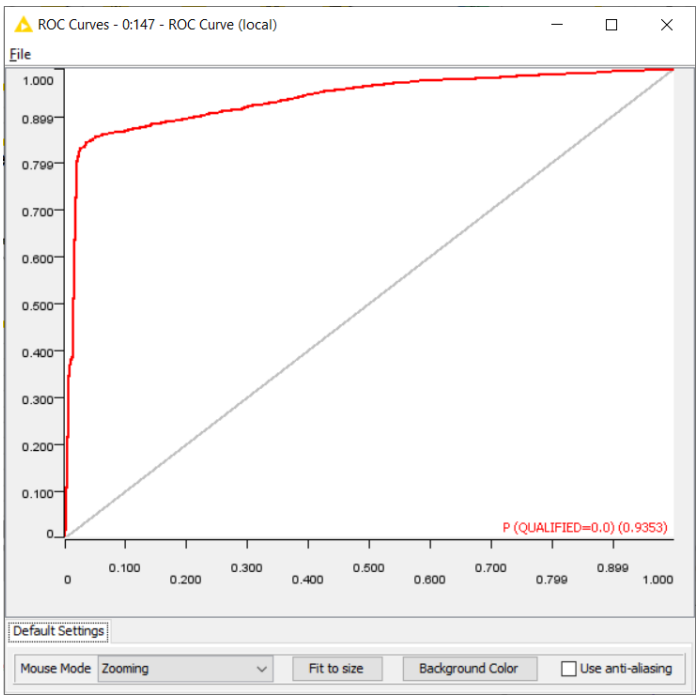
I found out this model gave me an overall accuracy of 88.487%. Whereas the AUC for the ROC Curve was 0.9353 for both 1 and 0.

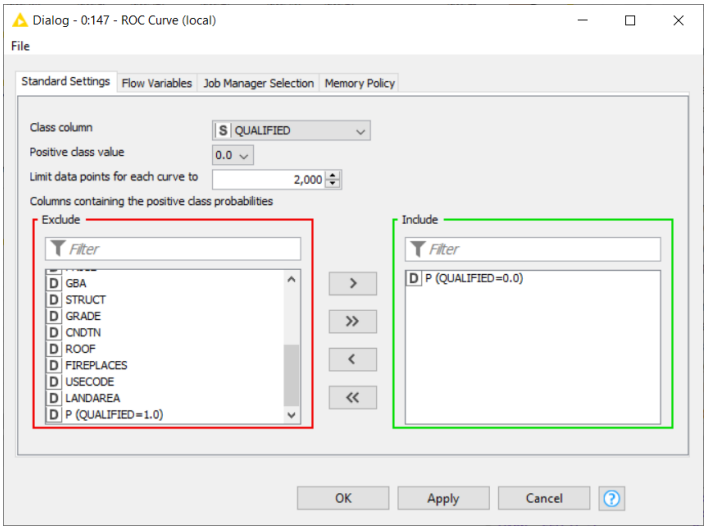


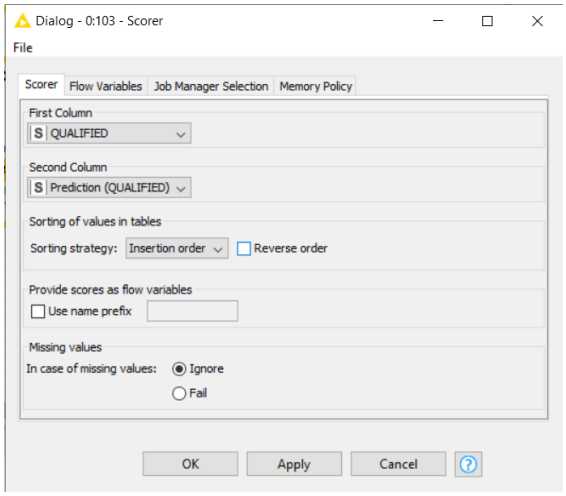


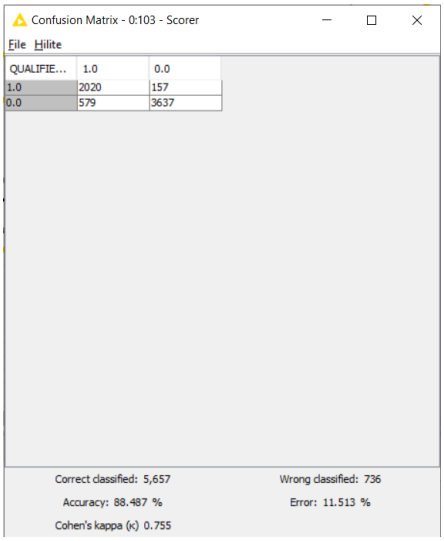

Dialog - 0:147 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
S QUALIFIED
Positive class value
1.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
D GBA
D STRUCT
D GRADE
D CNDTN
D ROOF
D FIREPLACES
D USECODE
D LANDAREA
D P (QUALIFIED=0.0)
Include
Filter
D P (QUALIFIED=1.0)
OK
Apply
Cancel


ROC Curves - 0:147 - ROC Curve (local)
File
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=1.0) (0.9353)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing


ROC Curves - 0:147 - ROC Curve (local)
File
1.000
0.899
0.799
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.799
0.899
1.000
P (QUALIFIED=0.0) (0.9353)
Default Settings
Mouse Mode
Zooming
Fit to size
Background Color
Use anti-aliasing

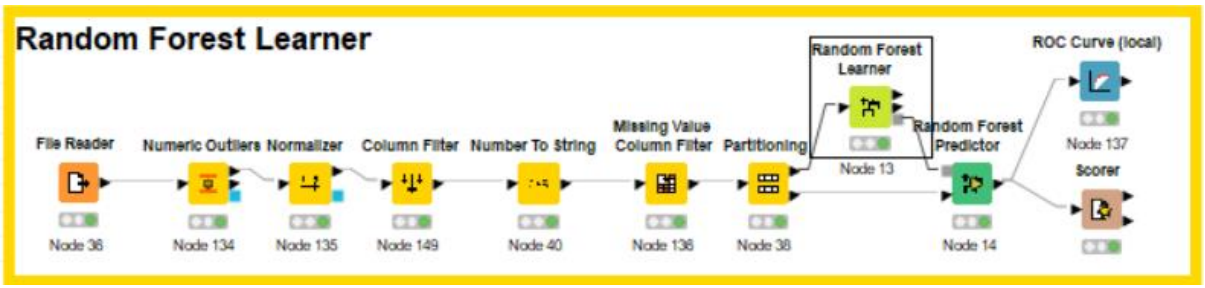

Dialog - 0:147 - ROC Curve (local)
File
Standard Settings
Flow Variables
Job Manager Selection
Memory Policy
Class column
S QUALIFIED
Positive class value
0.0
Limit data points for each curve to
2,000
Columns containing the positive class probabilities
Exclude
Filter
D GBA
D STRUCT
D GRADE
D CNDTN
D ROOF
D FIREPLACES
D USECODE
D LANDAREA
D P (QUALIFIED=1.0)
Include
Filter
D P (QUALIFIED=0.0)
OK
Apply
Cancel


Dialog - 0:103 - Scorer
File
Scorer
Flow Variables
Job Manager Selection
Memory Policy
First Column
S QUALIFIED
Second Column
S Prediction (QUALIFIED)
Sorting of values in tables
Sorting strategy:
Insertion order
Reverse order
Provide scores as flow variables
Use name prefix
Missing values
In case of missing values:
Ignore
Fail
OK
Apply
Cancel


Confusion Matrix - 0:103 - Scorer
File
Hilite
QUALIFIE...
1.0
0.0
1.0
2020
157
0.0
579
3637
Correct classified: 5,657
Wrong classified: 736
Accuracy: 88.487 %
Error: 11.513 %
Cohen's kappa (κ) 0.755

Best classifier

The best classifier for this project was Random Forest Learner. I got to know that not all classifiers were made for a dataset as such and only one was best suited. I preferred and chose this as it gave me the highest level of accuracy and helped me best predict the data. It is a very useful classifier. It solved the problem pretty easily and comfortably as compared with the others.



Reflection

After successful completion of this assignment, I found out that data mining is a very useful skill in today's world and to be successful and stay ahead of the league one must have an in-depth knowledge of the same. We must realize that in today's world the data is increasing at a very fast rate with each passing second and to cope up with the same one must have hands on experience with skills such as data mining. I also get to know that there is a share need to be able to process and convert data into useful form from an unorganized form. It is considered as one of many precious skills that one can have. We must also remember that it is upon us how much we develop our skill in the field of data mining that are required for our success. It is also considered to be one of the highest paying jobs of a data mining person. I also realize the importance of taking up courses such as this one to enhance my knowledge on the same and would like to thank my teacher Hesam Hesamian for providing me with such a brilliant opportunity to work and gain hands on experience on a platform like KNIME Analytics Platform. If I were given a chance to work again on this I would prefer finding and using many more nodes for a similar project and gain as much knowledge as possible using both the experience of my teacher as well as internet. I would watch more videos on YouTube and present a better model. I learnt a thing about myself that when and if I am faced with a problem such as this one where I am supposed to work on something that is totally new to me then I can handle it well when I concentrate to the fullest. I would like to thank my teacher once again for helping me discover this strength of mine. All in all, I feel privileged to have done such a project and performing to the best of my capabilities.