ASSESSMENT TASK 2: DATA EXPLORATION AND PREPARATION

31250 Introduction to Data Analytics

John-Paul Martin
14508648

Contents

A.1 Attribute Type Identification

Attribute	Meaning	Type	Justification
Date	Date	Interval	Differences are measurable but there is no true zero.
Location	Name of the place	Nominal	Value acts as a label.
MinTemp	Minimum temperature in the 24 hours to 9am. Sometimes only known to the nearest whole degree.	Interval	Differences are measurable but there is no true zero.
MaxTemp	Maximum temperature in the 24 hours from 9am. Sometimes only known to the nearest whole degree.	Interval	Differences are measurable but there is no true zero.
Rainfall	Precipitation (rainfall) in the 24 hours to 9am. Sometimes only known to the nearest whole millimetre.	Ratio	Attribute has true zero point.
Evaporation	"Class A" pan evaporation in the 24 hours to 9am	Ratio	Attribute has true zero point.
Sunshine	Bright sunshine in the 24 hours to midnight	Ratio	Attribute has true zero point. E.g. night time has zero sunshine.
WindGusDir	Direction of strongest gust in the 24 hours to midnight	Nominal	Value acts as a label, e.g. North, East.
WindGusSpeed	Speed of strongest wind gust in the 24 hours to midnight	Interval	Attribute has a true zero point.
Temp	Temperature.	Interval	Differences are measurable but there is no true zero.
Humidity	Relative humidity.	Interval	Differences are measurable but there is no true zero.
Cloud	Fraction of sky obscured by cloud.	Ordinal	Can be ordered based on cloudiness, e.g. clear, partly cloudy, cloudy.
WindDir	Wind direction averaged over 10 minutes.	Nominal	Value acts as a label, e.g. North, East.
WindSpeed	Wind speed averaged over 10 minutes.	Ratio	Attribute has a true zero point.
Pressure	Atmospheric pressure reduced to mean sea level.	Ratio	Attribute has a true zero point.
RainToday	If it's rain then Yes. If it doesn't rain then No	Nominal	Value acts as a label.

RainTommorow	If it's rain then Yes. If it doesn't rain then No	Nominal	Value acts as a label.

Table 1. Attribute Types

A.2 Identification of Summarising Properties

Attribute	Mean	Range	Min	Max	Strd Deviation	Variance	Skewness	Kurtosis
Min Temp	12.21	37.9	-6.7	31.2	6.48	42.05	0.04	-0.49
Max Temp	23.18	46.4	-2.1	44.3	7.13	50.77	0.18	-0.33
RainFall	2.51	182.6	0.0	182.6	8.38	70.19	8.03	110.62
Evaporation	5.57	50.8	0.0	50.8	4.4	19.32	2.81	16.49
Sunshine	7.56	13.7	0.0	13.7	3.8	14.44	-0.5	-0.83
WindGustSpeed	40.0	93.0	7.0	100.0	13.2	174.15	0.73	0.82
WindSpeed9am	14.17	54.0	0.0	54.0	8.91	79.32	0.71	0.55
WindSpeed3pm	18.83	57.0	0.0	57.0	8.81	77.55	0.54	0.4
Humidity9am	68.97	96.0	4.0	100.0	18.94	358.77	-0.55	0.14
Humidity3pm	51.64	99.0	1.0	100.0	20.74	430.34	0.03	-0.55
Pressure9am	1017.7	51.0	989.6	1040.6	7.03	49.44	0.02	0.11
Pressure3pm	1015.26	53.5	984.4	1037.9	7.0	49.06	0.02	0.1
Cloud9am	4.55	8.0	0.0	8.0	2.91	8.47	-0.3	-1.51
Cloud3pm	4.59	8.0	0.0	8.0	2.75	7.57	-0.26	-1.45
Temp9am	16.97	46.1	-5.9	40.2	6.52	42.46	0.08	-0.37
Temp3pm	21.65	47.6	-5.1	42.5	6.98	48.67	0.19	-0.26

Table 2. Table of Summarising Properties

The following table reveals significant variation among the attributes within the given dataset. Max and MinTemp both fluctuate significantly, with ranges of 46.4 and 37.9 degrees. Additionally, their respective standard deviations are considerably high indicating a vastness of temperature throughout the dataset. Rainfall stands out with extreme variability, boasting a

range of 182.6 mm and a skewness of 8.03. WindGustSpeed's skewness of 0.73 indicates more consistent, lower speed winds. Evaporation and Sunshine appear to be more stable compared to other attributes with ranges of 50.8mm and 13.7 hours. Pressure is one of the more stabler attributes presenting itself with lower ranges, suggesting consistent patterns across Australia.

A.3 Data Exploration

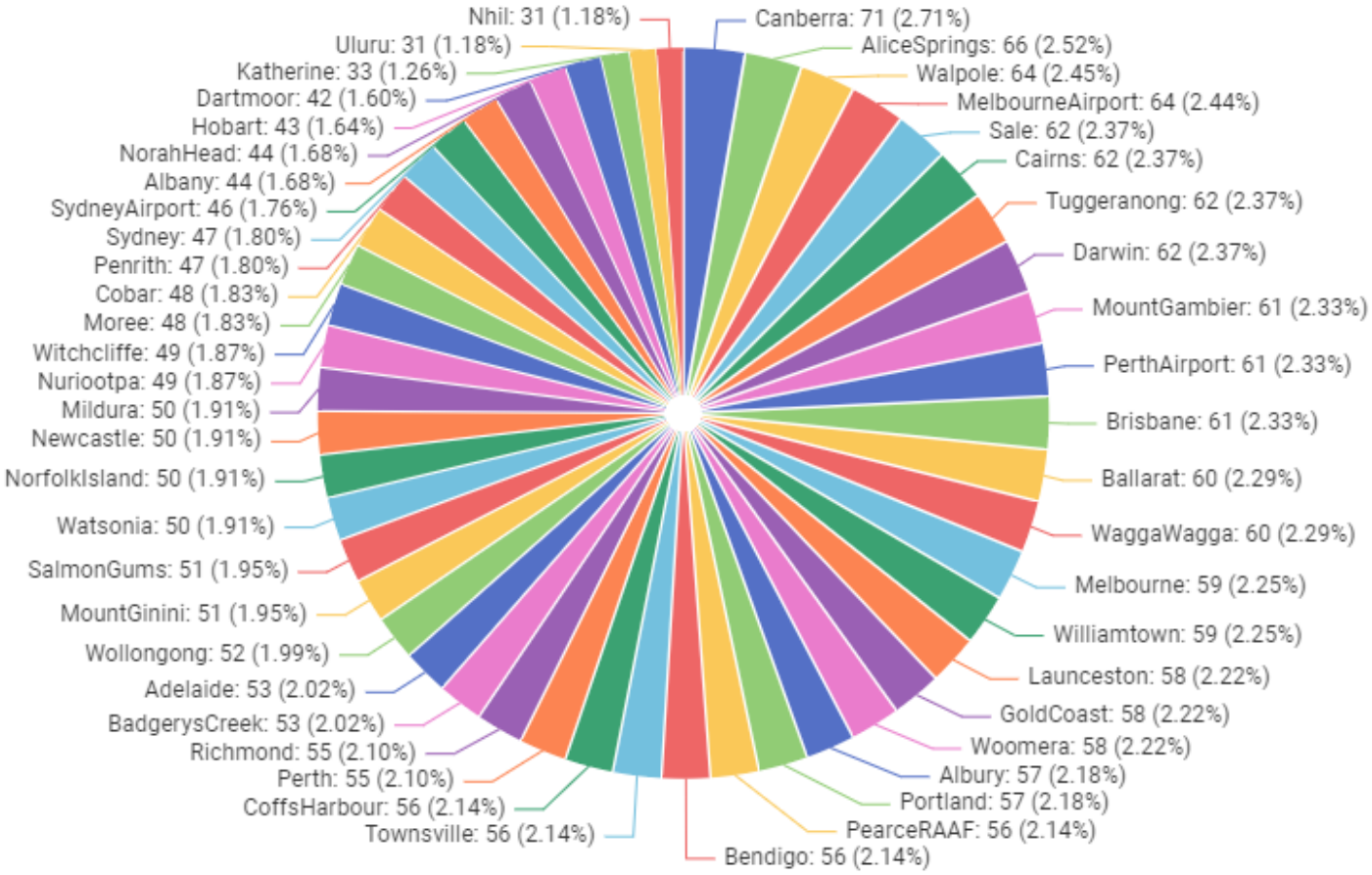


Figure 1. Occurrence of each location

Here we can effectively observe the occurrence of each location within the provided dataset. As it stands, there are a total of 49 unique location values that make up the pie chart. We can assume based on this figure, that less populated regions would be more likely to have fewer occurrences within the dataset as opposed to more populated regions. For example, Nhil is a small town with a population shy of 2500 as of 2021 and has the smallest presence within the graphic with 31 occurrences. On the other hand, Canberra, a region just under 400,000 people, has the highest number of occurrences at 71, making up 2.71% of all locations in the dataset.

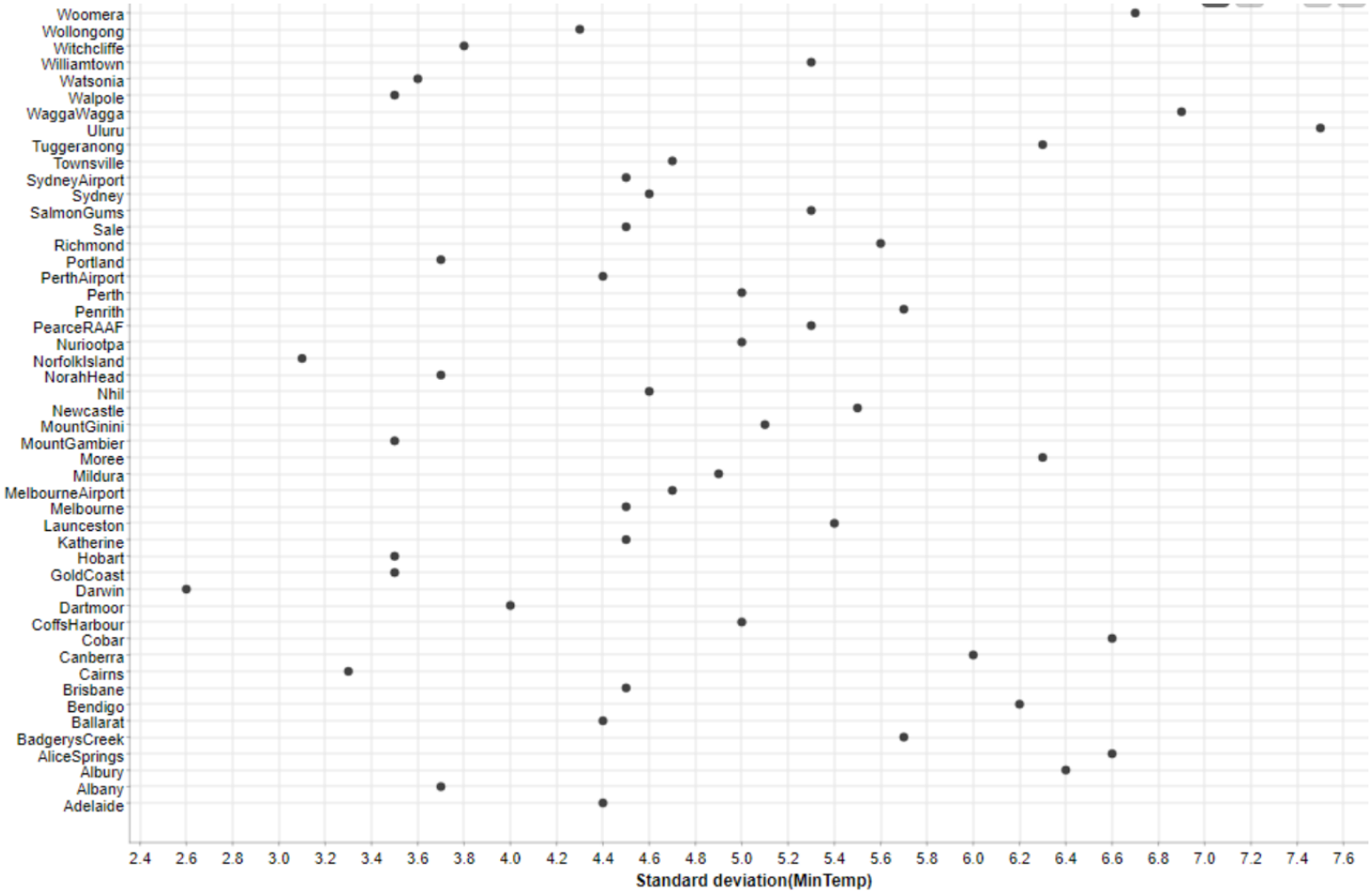


Figure 2. Scatter Plot of Standard deviation (MinTemp)

From the above figure and table, we can make the following observations:

- PearceRAAF, an air base located in Western Australia, has the largest range with a recorded range of 29.7 degrees.
- The location averaging the highest temperature by MinTemp was Darwin with an average of 23.3 degrees.
- A standard deviation of 7.5 at Uluru makes it the highest of all locations within the dataset. This indicates that Uluru experiences more fluctuation in its minimum temperature over time.
- This fact is reinforced through the provided scatter plot, indicating that Uluru is a clear outlier.

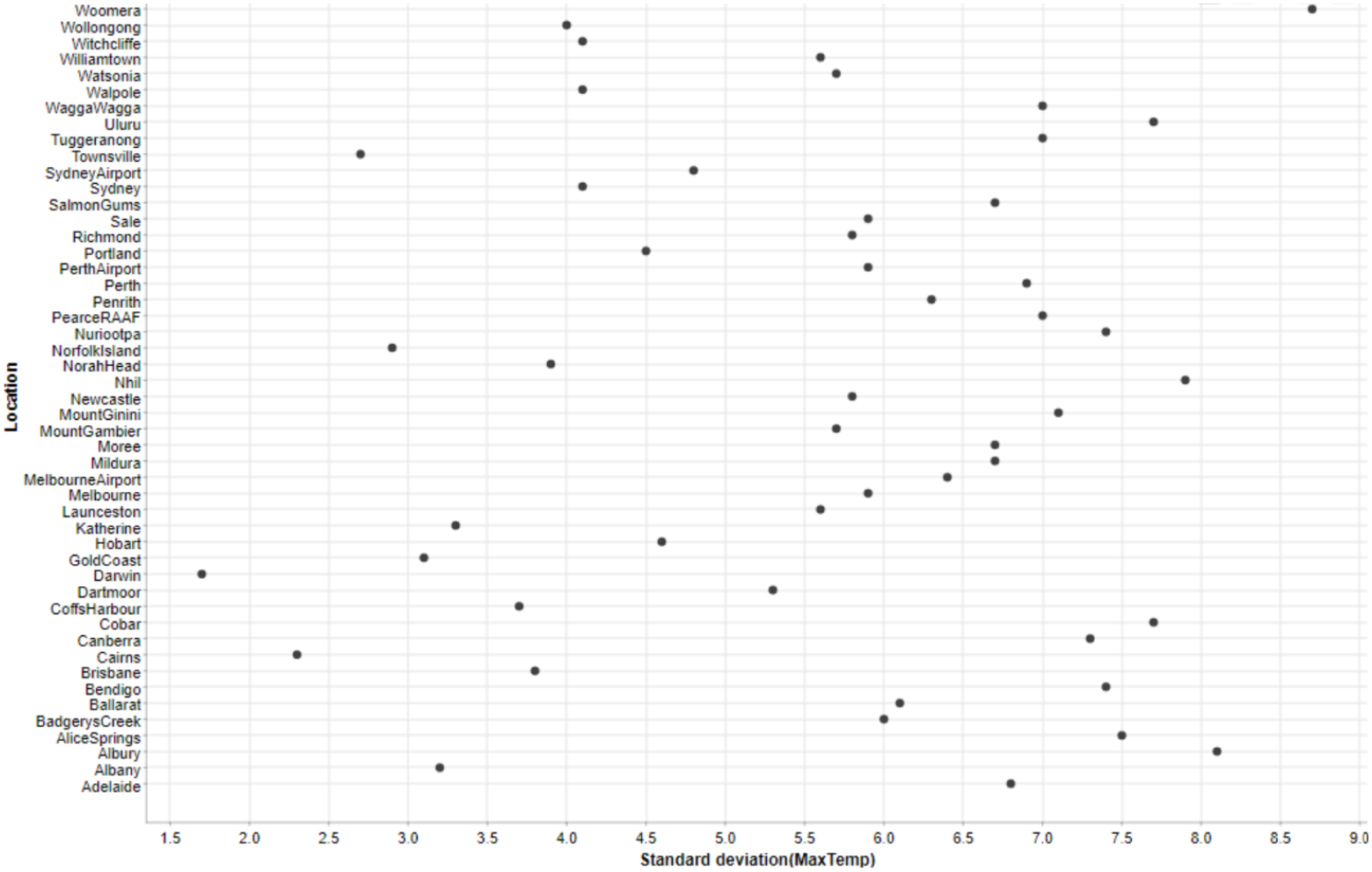


Figure 3. Scatter Plot of Standard deviation (MaxTemp)

From the above figure and table, we can make the following observations:

- Melbourne has the largest range with a recorded range of 29.7 degrees indicating a wide variation between the minimum and maximum recorded temperature in terms of MaxTemp.
- Additionally, Melbourne has the highest maximum temperature, and Darwin has the minimum.
- Darwin has the smallest range of 8.7, suggesting that their climate is very stable with consistently hot temperatures.
- A standard deviation of 8.7 at Woomera makes it the highest of all locations within the dataset. This indicates that Woomera experiences more fluctuation in its maximum temperature over time.

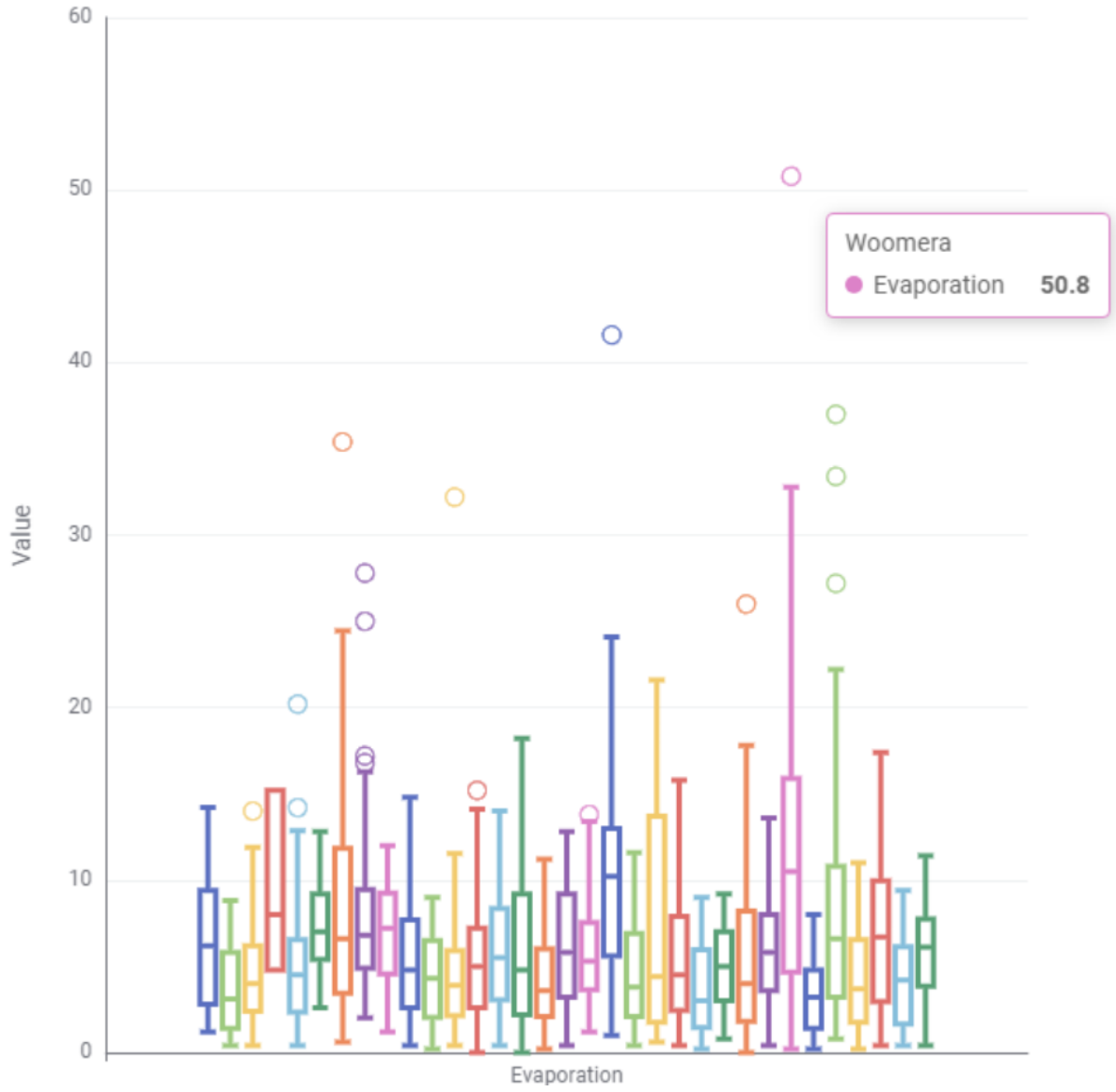


Figure 4. Box Plot of Evaporation Values by Location

The above figure reveals the overall distribution of evaporation values within the dataset, highlighting trends and potential outliers. It appears that the majority of locations record an evaporation value between 10 and 20, more specifically the bottom portion of the two. A clear outlier can be identified in Woomera, with a recorded value of 50.8, far greater than any other location. Furthermore, it appears that their recorded maximum is significantly higher than the rest with the only recorded maximum above a value of 30. This allows us to make the assumption that Woomera experiences considerably extreme evaporation compared to the norm.

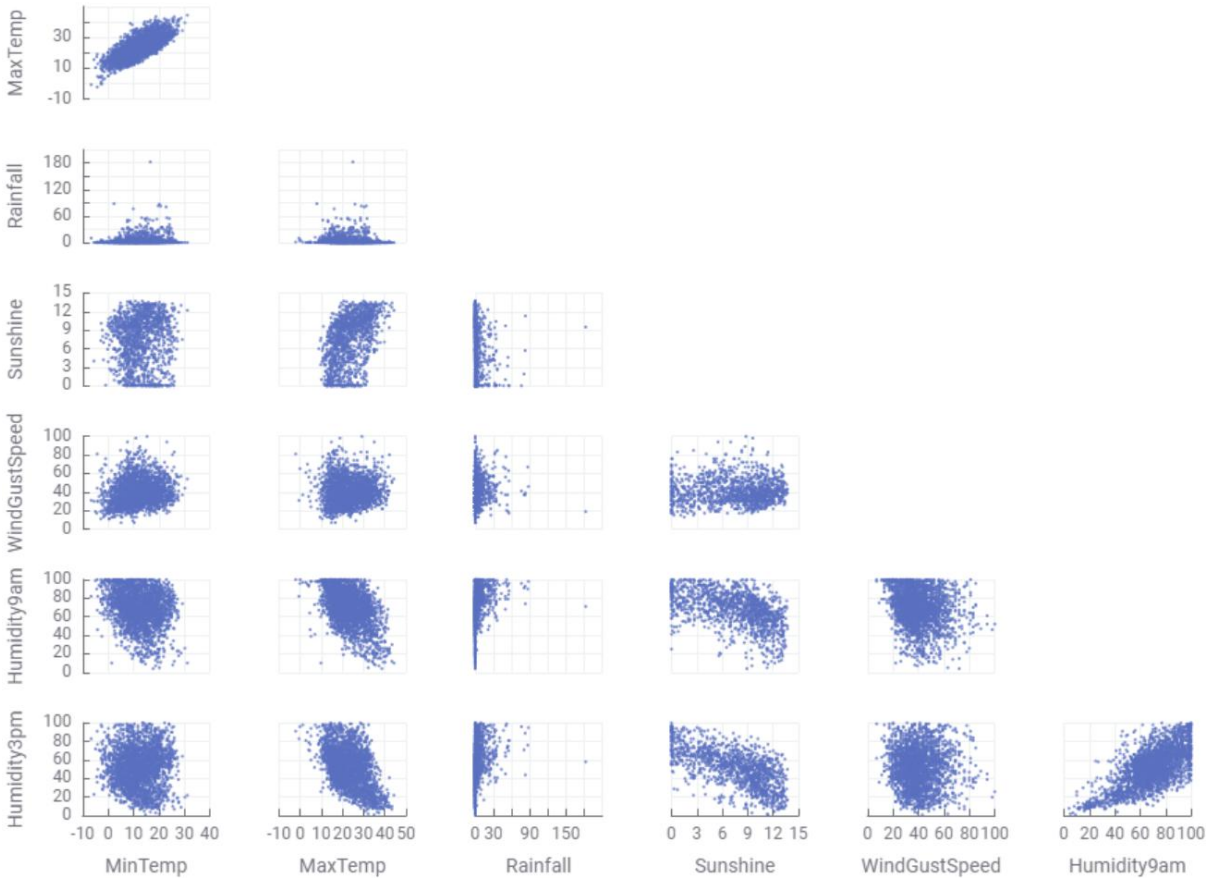


Figure 5. Scatter Plot Matrix

Here we can visualize the relationships between various attributes. This matrix reveals several findings:

- As MinTemp increases, the MaxTemp rises slightly as well. However, Rainfall does not display as much correlation as the two, or with any attributes for that matter, suggesting it is extremely independent, or its dependent attributes are not listed in the matrix.
- As expected, both Humidity attributes display a strong correlation, reflecting consistent recordings throughout the dataset.
- Additionally, Sunshine has some level of correlation with Min and MaxTemp, meaning that high temperatures are often linked with high levels of sunshine.

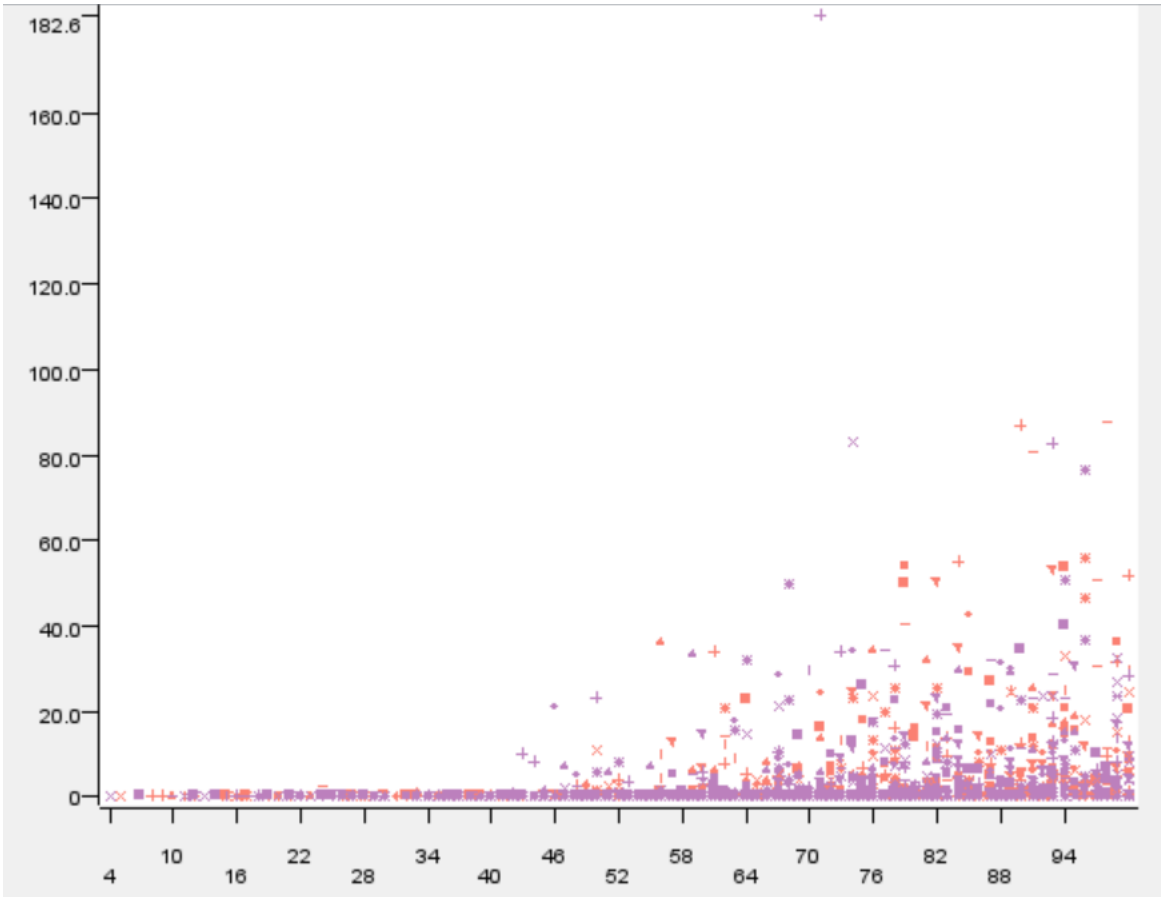


Figure 6. Scatter Plot Using Hierarchical Clustering

In this graphic, I utilise the Hierarchical Clustering node to observe the correlation between Humidity9am, and Rainfall. The node datapoints into clusters based on how similar they are. We can use this to identify any patterns and outliers. Along the X axis lies Humidity9am, and the Y axis contains Rainfall.

- As the value Humidity9am increases, so does the value of Rainfall.
- We only notice significant changes after Humidity9am reaches a value of 43, suggesting that this is the threshold for rain to occur.
- There is an anomaly present when humidity reaches 70, with a RainFall value well beyond the rest of the results, contradicting the fact that as humidity increases, rainfall also gradually increases. There is room for investigation into this datapoint.

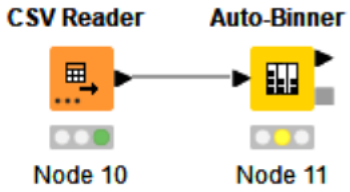
B1. Data Preprocessing: Equi-width and Equi-depth binning

Binning is a preprocessing technique used to group several continuous values into a smaller number of bins, making it easier to analyse or interpret data. I will be using two binning techniques to effectively smooth the values of the RainFall attribute. These are:

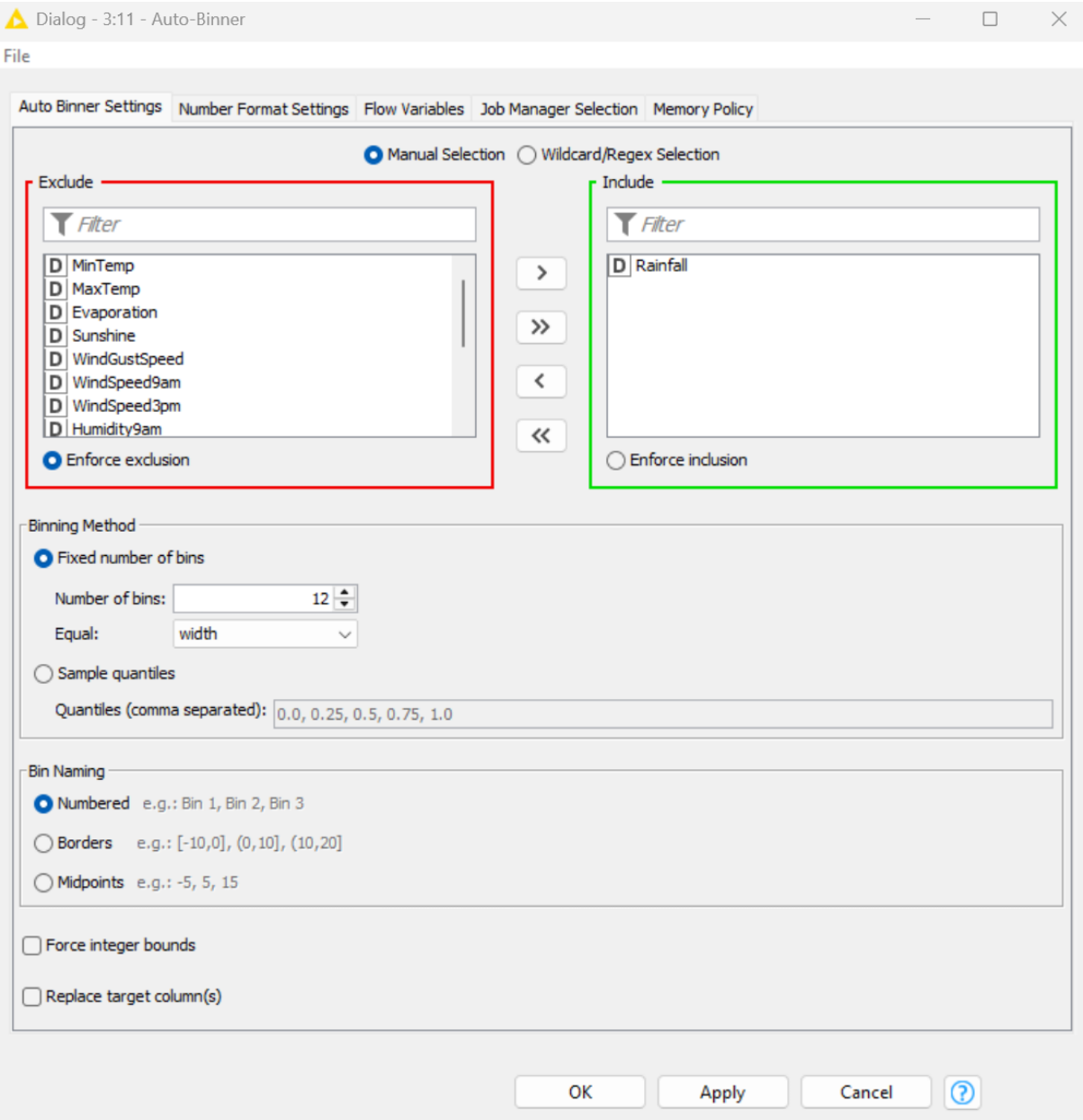
1. Equi-width: Used to split the whole range of numbers in intervals with equal size.
2. Equi-depth: Use intervals containing an equal number of values.

The following screenshots will demonstrate the process of binning (steps are the same for both techniques:

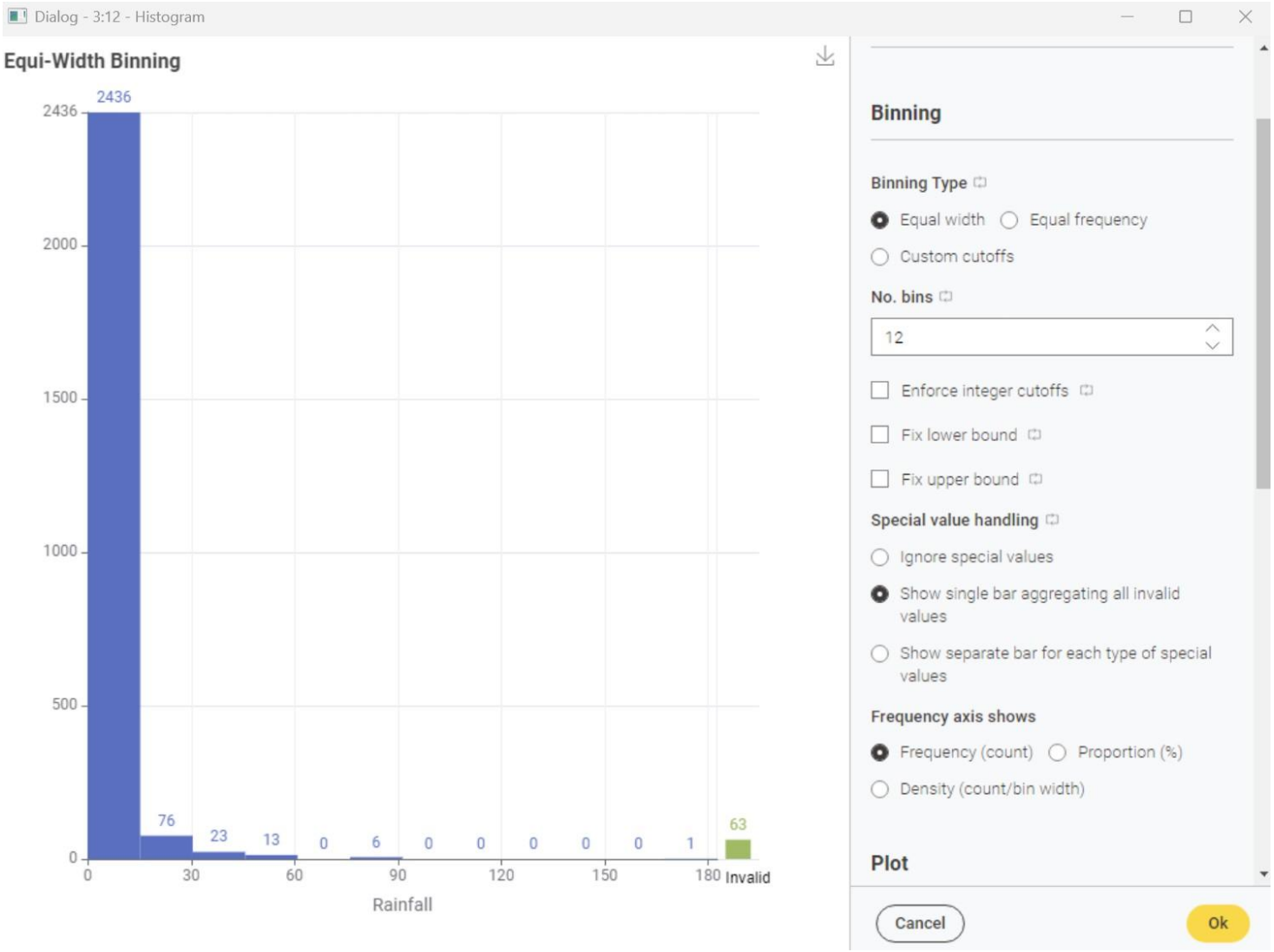
Importing CSV Reader node to read dataset.



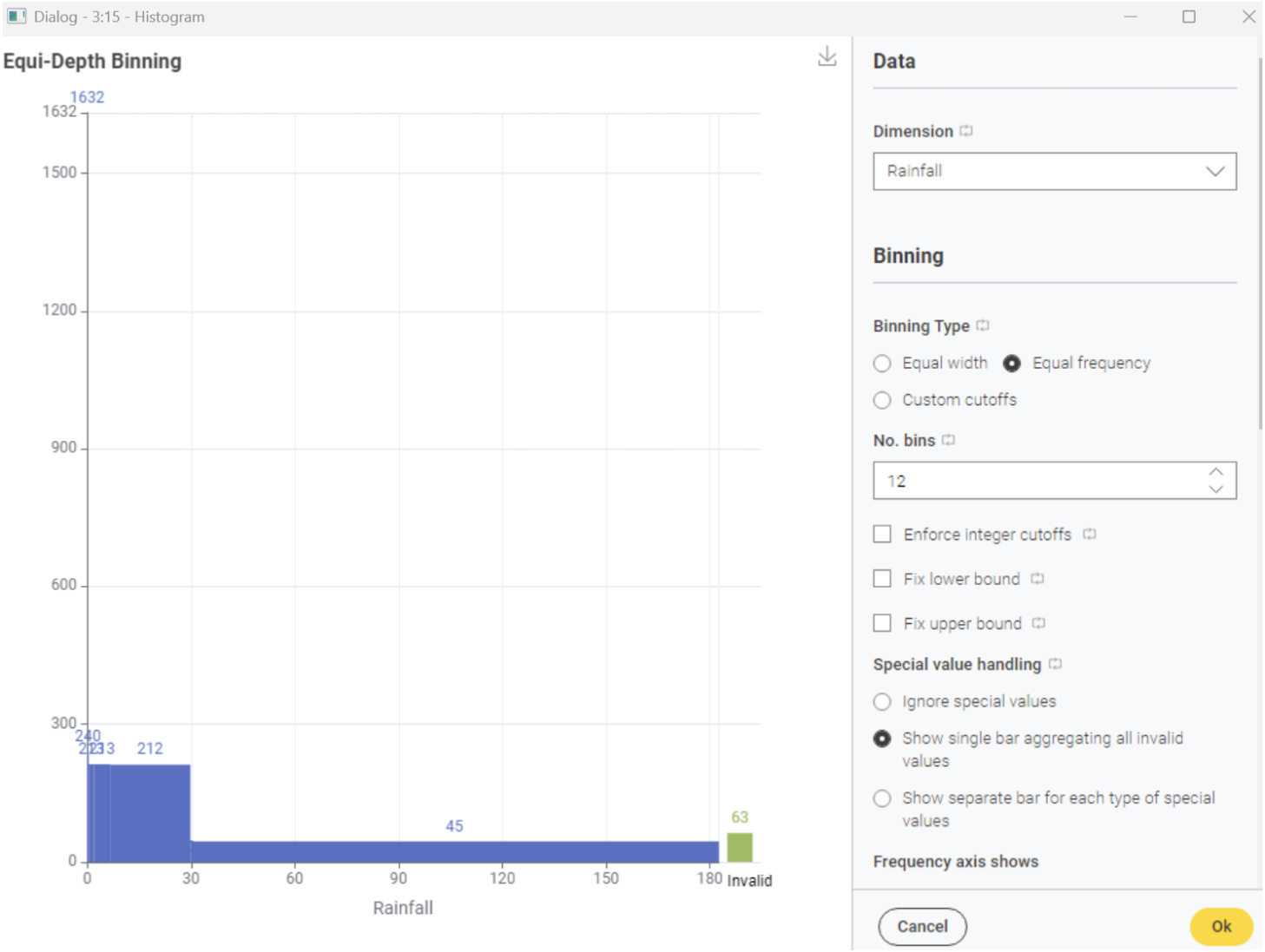
Import Auto-Binner node.



Configure Auto-Binner node. The number of bins was found using Sturge's Rule; $1 + \log_2(n)$, where n is the number of data points, which in this case is the number of rows, 2618. Using this formula, the number of bins chosen was 12. In 'Equal' menu, we choose width for Equi-width, and frequency for Equi-depth.



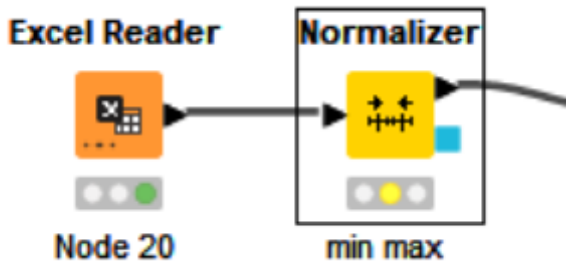
We can use the Histogram node to visualize the results of using the Equi-width binning technique. Along the Y axis we have the number of data points, and along the X axis are the bins, each with a width of 15. The vast majority of RainFall values lie in the first bin with a total of 2436, suggesting extreme rain events are extremely rare in many regions. However, there are a select few values that fall into Bins 2 and beyond, indicating that there are some instances of extreme rain fall in certain areas. Additionally, Bin 12 has 1 datapoint suggesting a catastrophic rain event. We can also observe the inclusion of missing values under the invalid bar for a total of 63.



We can also use a histogram to visualise the results of binning use Equi-depth. Here, the results do not vary as much as Equi-depth.

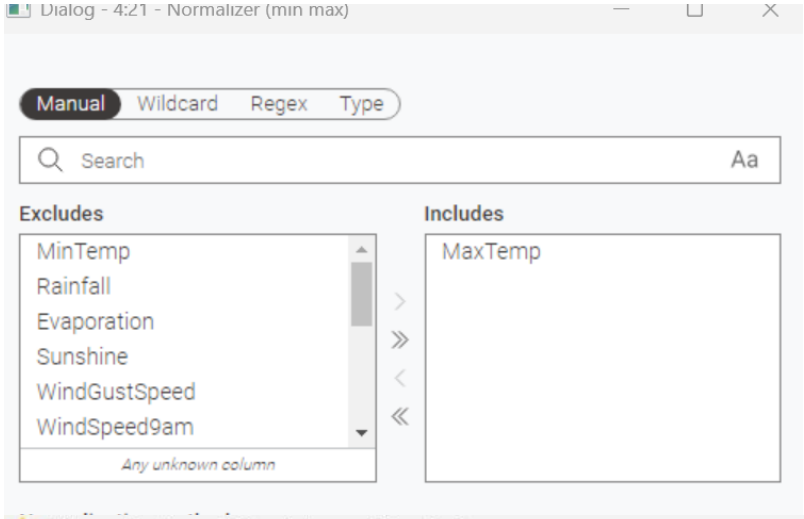
B.2 Normalisation

In this section, two widely used normalisation techniques will be applied to the attribute MaxTemp. These are Min-Max, and Z-Score normalisation and are a key step in the process of data preprocessing. The following screenshots will demonstrate how to do this in Knime.

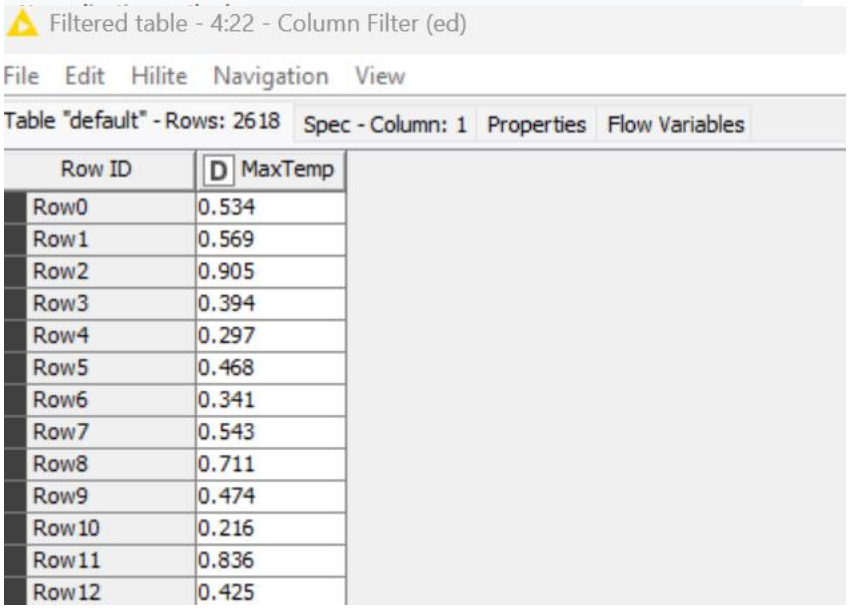


Ensure the Excel Reader is configured with the correct file before importing the Normalizer node.

MinMax:



From the configuration interface, we then ensure that we only include the required attribute, MaxTemp, before selecting Min-max as our normalisation method. From here, we adjust the minimum and maximum values as per the assignment requirements, before executing.

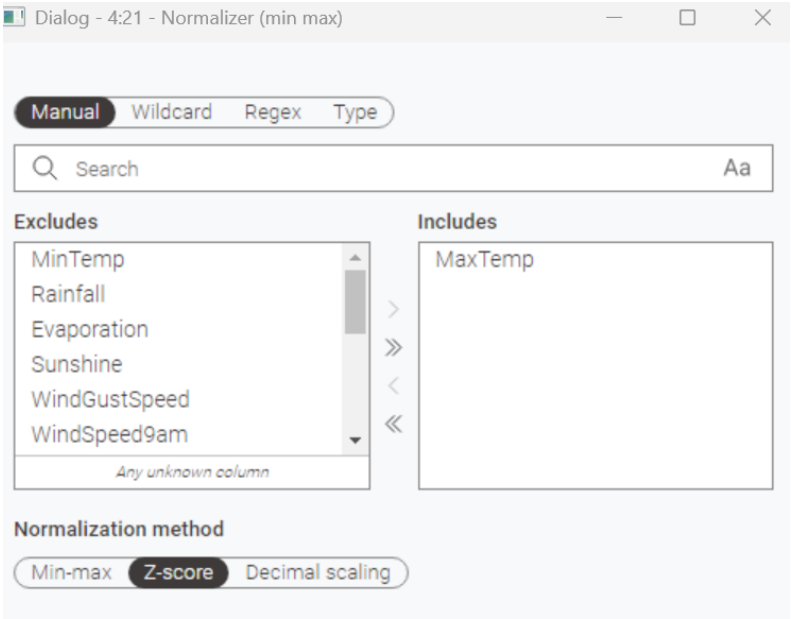


Row ID	D MaxTemp
Row0	0.534
Row1	0.569
Row2	0.905
Row3	0.394
Row4	0.297
Row5	0.468
Row6	0.341
Row7	0.543
Row8	0.711
Row9	0.474
Row10	0.216
Row11	0.836
Row12	0.425

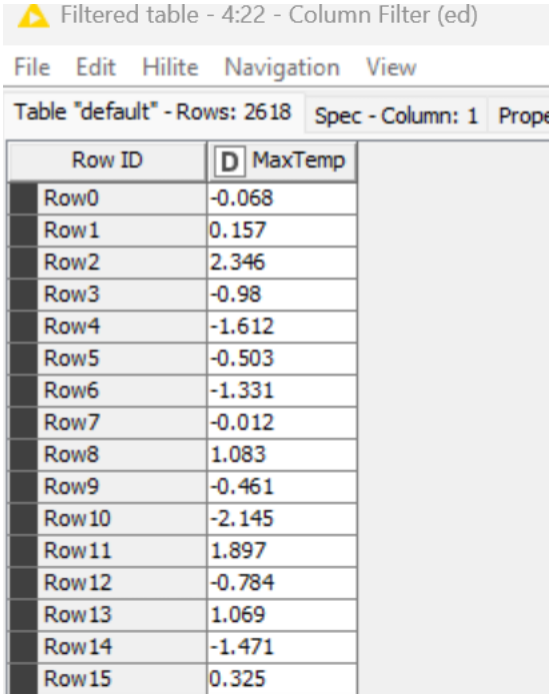
Here we have a snapshot of the normalised data using the Min-Max method.

Min-Max Normalization can be used to effectively scale data into a range. The specified values are adjust based on the minimum and maximum values of MaxTemp, transforming the data to a normalized scale.

Z-Score:



For Z-Score Normalization, we simply have to select the desired attribute and ensure that Z-Score is the selected Normalization method before executing.

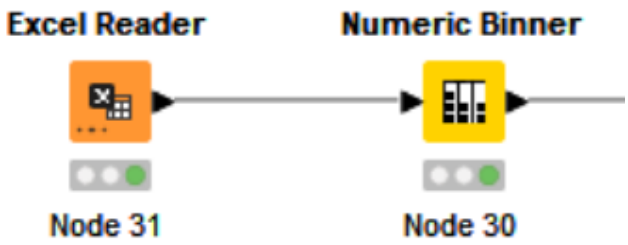


Using the Column Filter node, we can observe the result of the Normalized data.

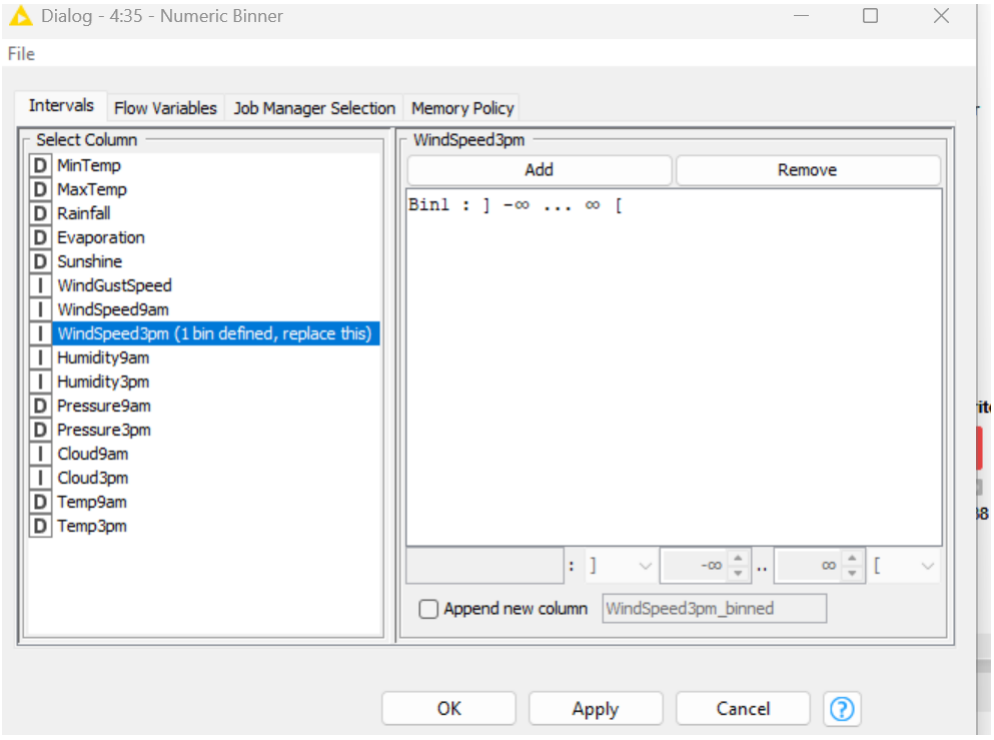
Much like MinMax, Z-Score transforms the data. However, the transformation is based on the data's mean and standard deviation. A value will be exactly equal to the mean if it is normalized to 0. Hence, if a value is negative, it is below the mean, and if it is positive, it is above the mean.

B.3 Discretisation

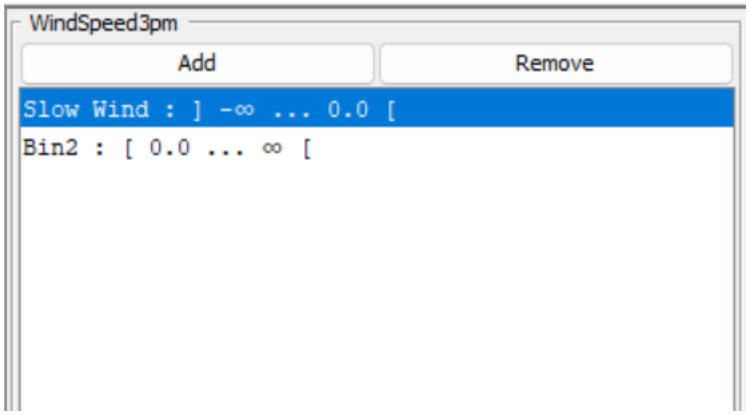
Discretisation is the process of converting data into different categories. This is especially useful when you wish transform numerical data into categorical data for analysis such as classification. Here, I will demonstrate how to discretise the variable, WindSpeed3pm into 4 distinct categories: Slow wind, Medium Wind, Fast Wind, Very Fast Wind.



Import the Numeric Binner node and connect it to the relevant reader with the appropriate dataset.

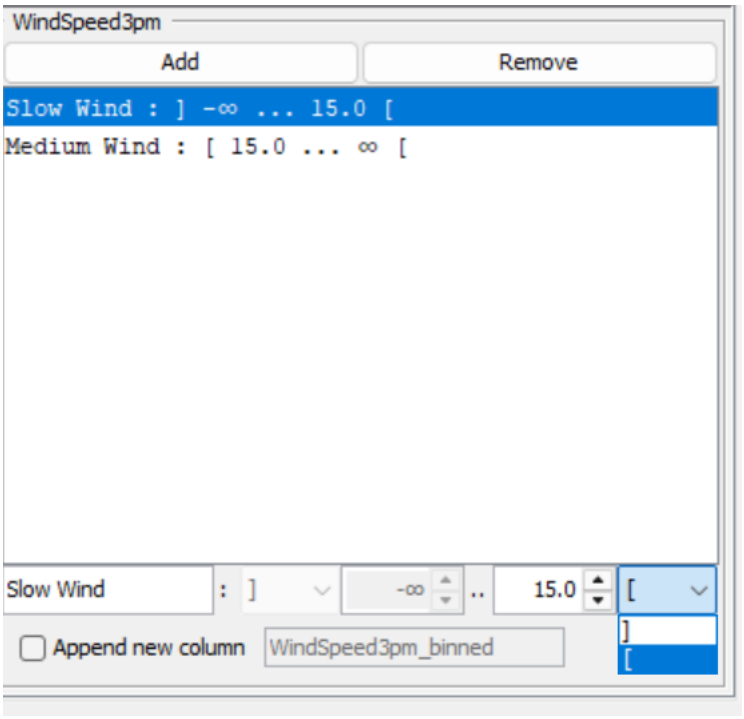


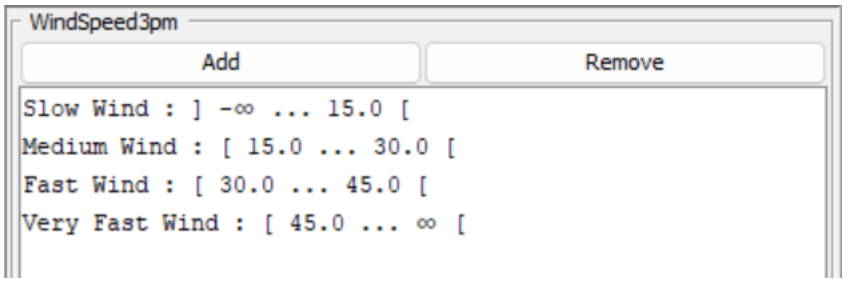
From the configuration screen, select the desired column and press “Add”. This will add a bin. Before proceeding, we need to determine the range for each bin. Looking at the previous summarisation table, we can see that the range its 57 for WindSpeed3pm. Hence, we can divide this value by the number of categories. Since we have 4 categories, we will reach a total of 14.25. However, since we can only do whole numbers, we will use the ceiling function and assign each bin a length of 15.



Here we rename Bin 1 to “Slow Wind” and add another bin so that we can edit the range for Slow Wind.

We can now adjust the Slow Wind range to 15. Additionally, in the bottom right is a menu. This gives us the option to either exclude or include the value of 15 in the first bin. We will choose to exclude as I only want values smaller than 15 under Slow Wind.

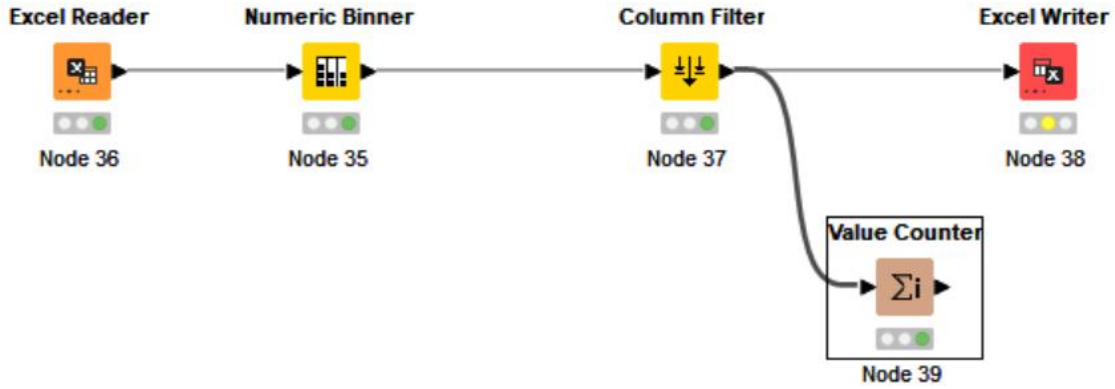




Ensure that we do the same for every other bin. It is clear that each category has a range of 15, thus each value within the dataset will be assigned a bin. Very Fast Wind, the last category, will account for all values greater than 45.

WindSpeed3pm	Discretisation
15	Medium Wind
11	Slow Wind
7	Slow Wind
13	Slow Wind
30	Fast Wind
22	Medium Wind
31	Fast Wind
15	Medium Wind
17	Medium Wind
20	Medium Wind
9	Slow Wind

Using Excel, we can format the discretisation results against the actual WindSpeed3pm values. All values under 15 are clearly marked with a value of "Slow Wind", and so on. Empty values are not accounted for as indicated by the empty space in the discretisation column.

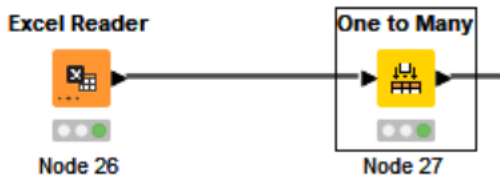


Row ID	count
?	44
Fast Wind	326
Medium Wind	1431
Slow Wind	799
Very Fast Wind	18

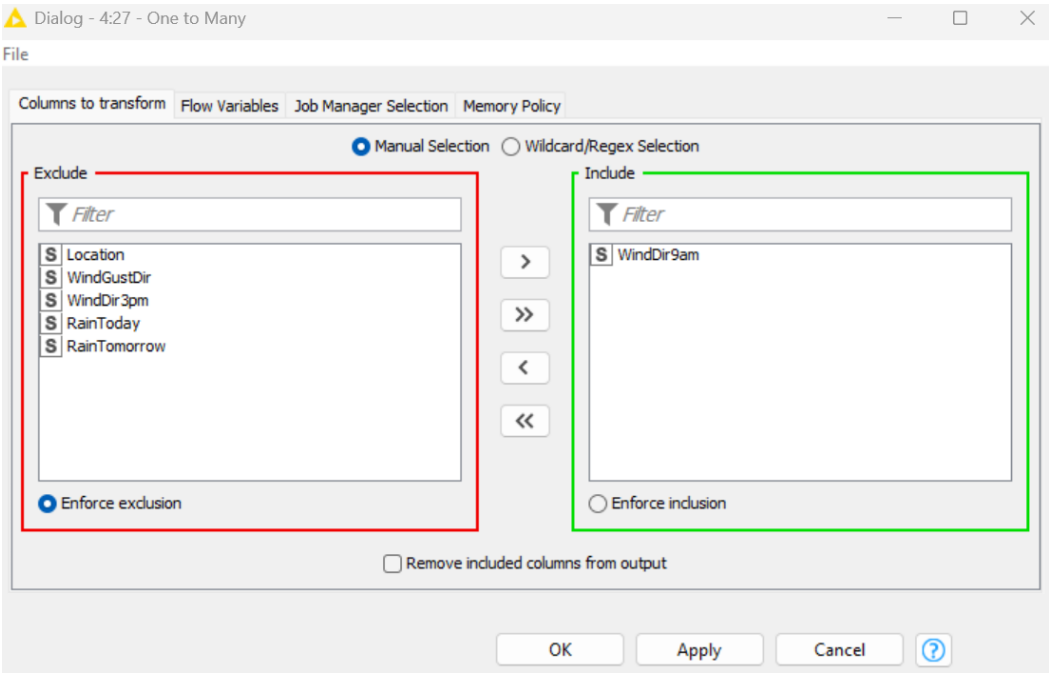
Additionally, through the use of the Value Counter node, we can determine the frequency of each category in the dataset. Medium Wind clearly occurs the most, with a count of 1431. Very Fast Wind however, has the least occurrences within the dataset, accumulating only 18 counts. 44 rows are also not discretised due to missing values has indicated by the "?".

B.4 Binarization

This section of the report will focus on applying a binarization technique to WindDir9am. Before proceeding, it is important to understand what binarization is. It is the process of simplifying data by converting it into binary form, making it easier to analyse. It is a commonly applied technique in preprocessing and is commonly used to improve machine learning models. To perform binarization, I completed the following:



Import the One-to-Many node and connect it to the appropriate reader with the relevant data.



We then configure the One-to-Many node to only Binarize the required variable, WindDir9am.

WindDir...	SSW	ESE	E	N	WNW	NW
SSW	1	0	0	0	0	0
ESE	0	1	0	0	0	0
E	0	0	1	0	0	0
N	0	0	0	1	0	0
WNW	0	0	0	0	1	0
NW	0	0	0	0	0	1
WSW	0	0	0	0	0	0
NW	0	0	0	0	0	1
NE	0	0	0	0	0	0
SW	0	0	0	0	0	0
W	0	0	0	0	0	0
SW	0	0	0	0	0	0
N	0	0	0	1	0	0

Using the column filter, we can see the result of the binarization. Here the categorical wind direction is converted into a binary matrix. Each direction is represented in a separate column,

with each row displaying either a value of 0 or 1 for its observed wind direction. For example, in row 1, the direction is SSW. Therefore, its corresponding column, SSW, should have a value of 1 in row 1. Similarly, the next row indicates a direction of ESE, hence in the SSW column, its respective row has a value of 0 since it does not match the corresponding columns direction.

C. Summary

Throughout the dataset, attributes of the ordinal type are extremely rare, as seen by the single instance, ‘Cloud’. While ordinal attributes may have a meaningful order, the lack of frequency of such attributes means they have little impact on analysis. On the other hand, interval and ratio attributes are prevalent throughout, providing detailed, qualitative and quantitative information, effectively impacting our analysis.

From Section A.2, we were given detailed insights into the provided data, with summaries for the entire dataset. As we can see, some attributes such as Cloud3pm, Sunshine, and Evaporation appear to be more stable, however the presence of highly unstable readings are present throughout. These include attributes such as Humidity9am and 3pm, as well as WindGustSpeed. In terms of variance, they are clear outliers with a recorded variance over 150 and ranges well over 80. This phenomenon indicates the need for further attention; hence I suggest that we closer attention to these attributes to determine the causes of their irregularities.

Moreover, special attention should be paid to the region of Woomera, a village located in South Australia, approximately 446 km from Adelaide. As detailed in section A.3, we can clearly see that Woomera is an outlier in each figure in terms of standard deviation by Min and MaxTemp, as well as the range of evaporation values. With such skewed recordings, it is crucial that we focus on understanding why regions such as Woomera face such risks and equip and prepare them accordingly.

Additionally, a large majority of regions are not impacted by torrential rainfall with only a select few instances recording more than 15ml of rain. Similarly, wind conditions are generally on the more conservative side with only 344 instances of fast wind or higher throughout the entire dataset.

In summary, while a significant portion of attributes exhibit stable behaviour, the dataset reveals clear outliers and irregularities, particularly in terms of humidity, wind, and regional variances. These anomalies, especially those associated with regions like Woomera, should be closely studied to provide deeper insights and help formulate more effective strategies for weather prediction and risk mitigation.

Introduction to Data Analytics

Assignment 3 – Data Mining in Action

John-Paul Martin, 14508648

Table of Contents

Problem Description

The problem is one that involves binary classification, which aims at predicting the attribute *'RainTommorow'.* The possible values for this attribute are 1 (Yes), if it does rain, and 0 (No), if it doesn't rain. This prediction holds significant value across multiple sectors such agriculture, transport and logistics, retail and supply chains, and even public safety. Accurately predicting rainfall for the next day allows for informed decisions and risk mitigation.

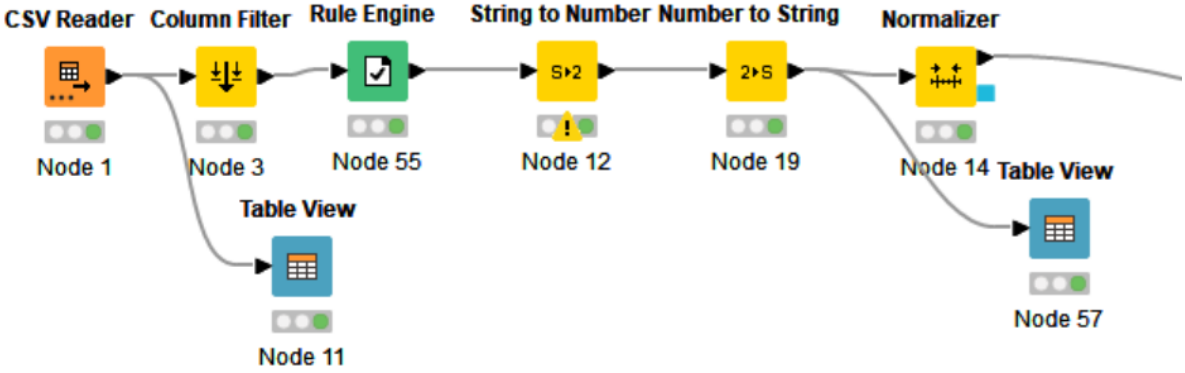
Input: Is made up of three datasets.

1. *WeatherData* (same dataset used in assignment 1, has statistics pertaining to weather in multiple regions throughout Australia).
2. *UnknownData* (will be used to assess the final model).

Output: The outcome will be a model that can accurately predict whether it will rain tomorrow based on the UnknownData dataset by creating a new column; "Predict-RainTommorow", and classifying the outcome as either:

- 1 (Yes) – it will rain tomorrow.
- 0 (No) – it will not rain tomorrow.

Data Preprocessing



CSV Reader:

- Allows us to import the relevant CSV file (WeatherData).

Column Filter:

- The evaporation and sunshine columns had many NA values, with 20,549 rows containing 'NA' in both. I removed these, along with Cloud9am, Cloud3pm, and other unrelated attributes like Location and WindGustDir, using the Column Filter.

Attribute	% of rows with NA value
Evaporation	42.3%
Sunshine	47.1%
Cloud9am	37.9%
Cloud3pm	40.3%

Table View:

- Check the attribute types; all except RainTomorrow are strings and need conversion.

Rule Engine:

- Converted Yes to 1 and No to 0 in RainToday column for model.

String to Number:

- Converts string-formatted numbers into numerical values for effective model training and evaluation.

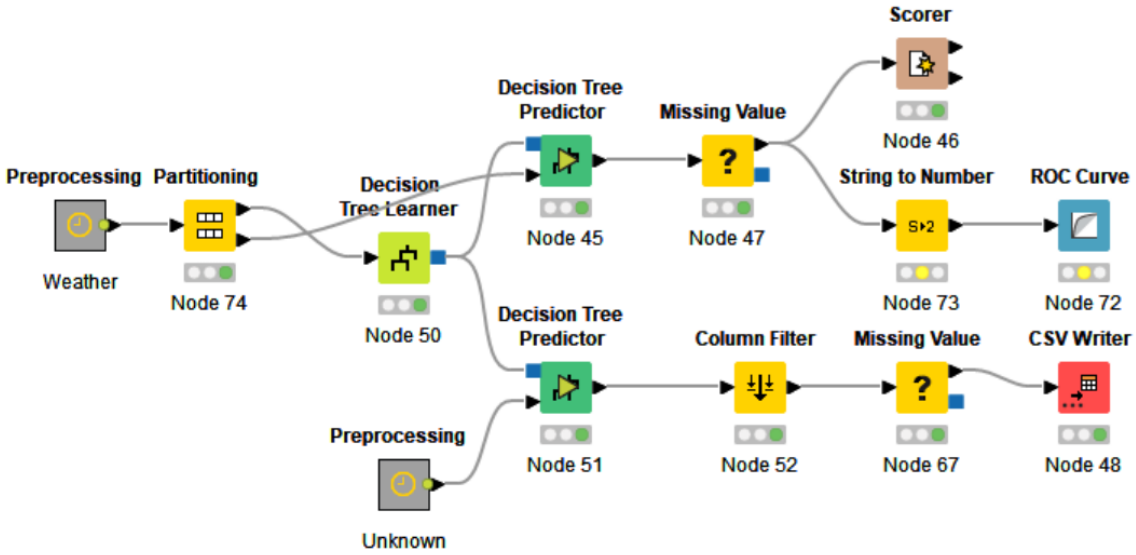
Number to String:

- Like String to Number node, however it converts numerical values into string-formatted attributes.

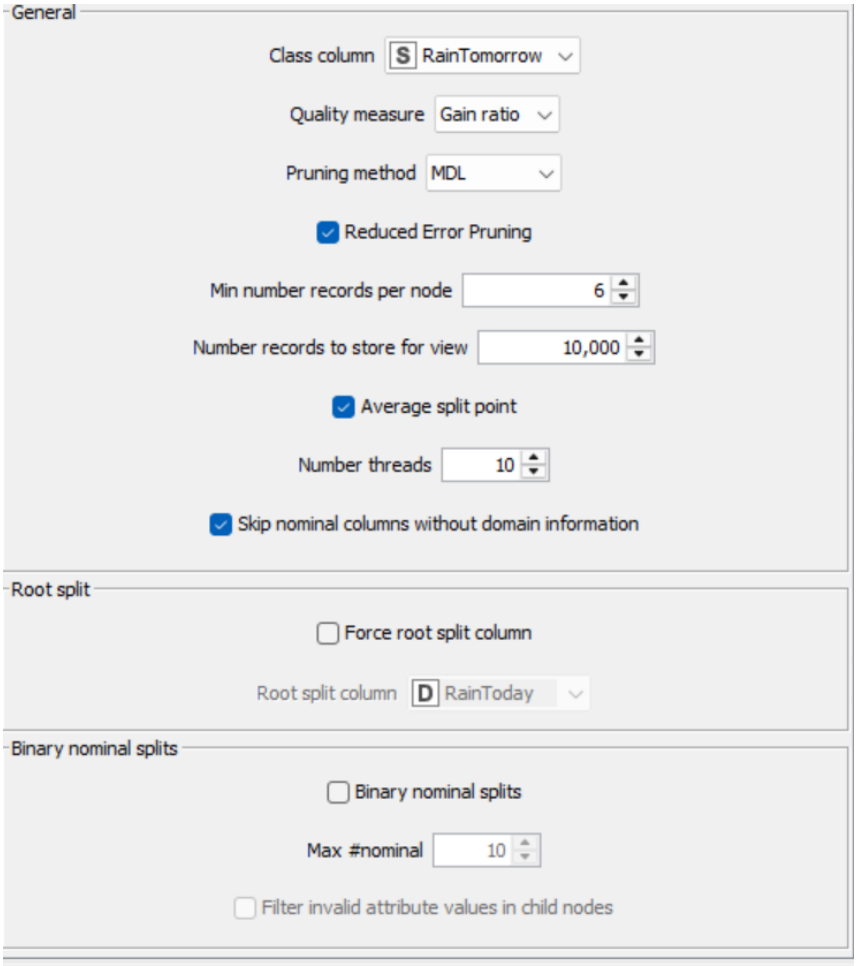
Normalizer:

- Configured to Min-max normalisation; 0 to 1.

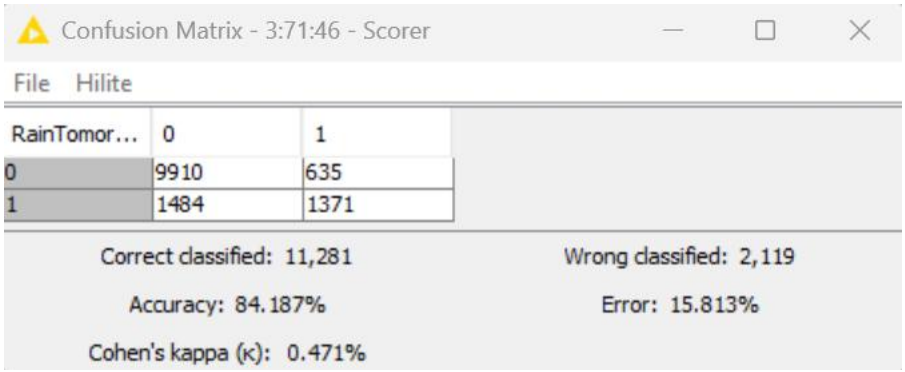
Decision Tree (DT)



After completing the data preprocessing steps, I load the dataset into the Partitioning node to divide it into training and test sets, allocating 70% of the data for training and the remaining 30% for testing. The option to Draw randomly is enabled to ensure random selection of data for each set. The training set is then used as input for the Decision Tree Learner node, which is configured with the following settings:

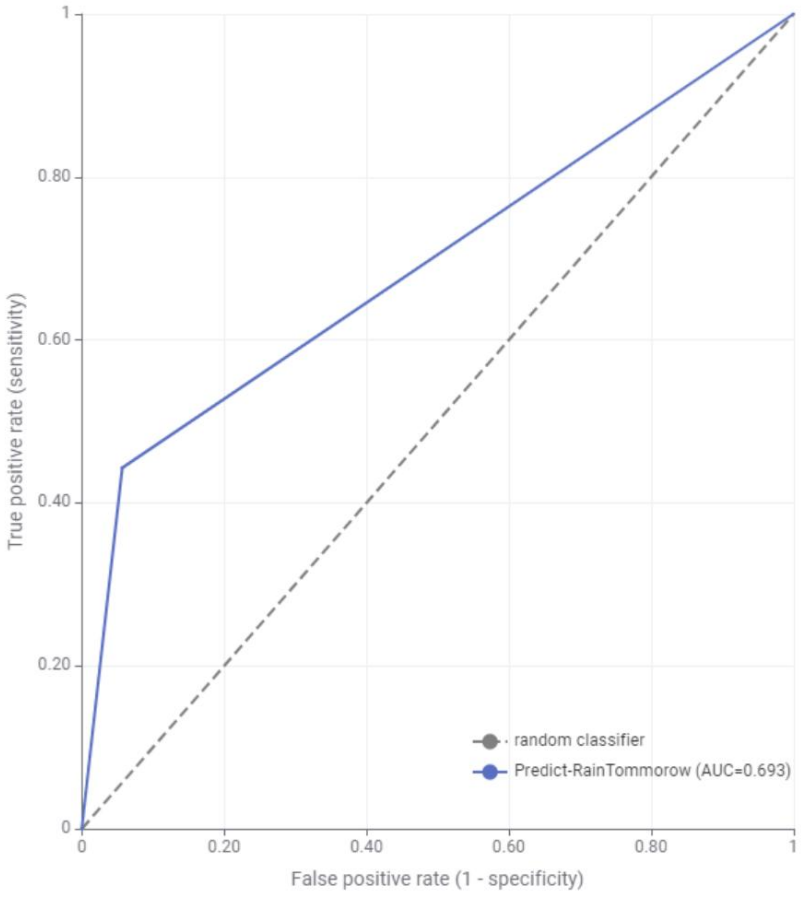


Using the Scorer node, I can visualize the results of this model:

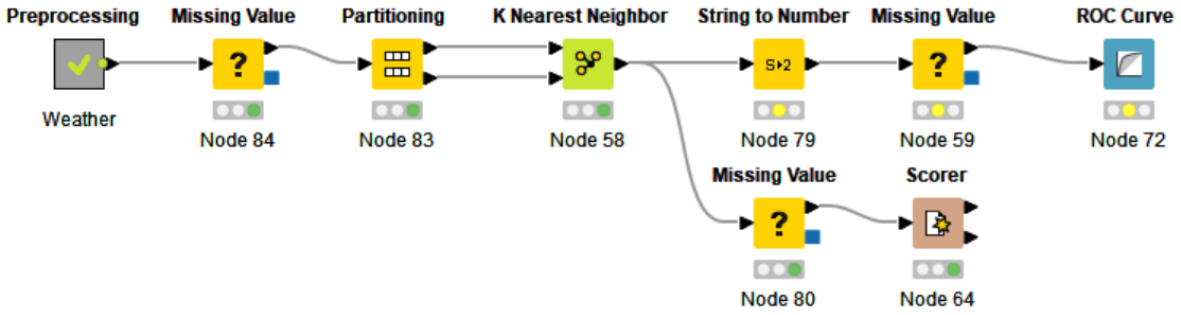


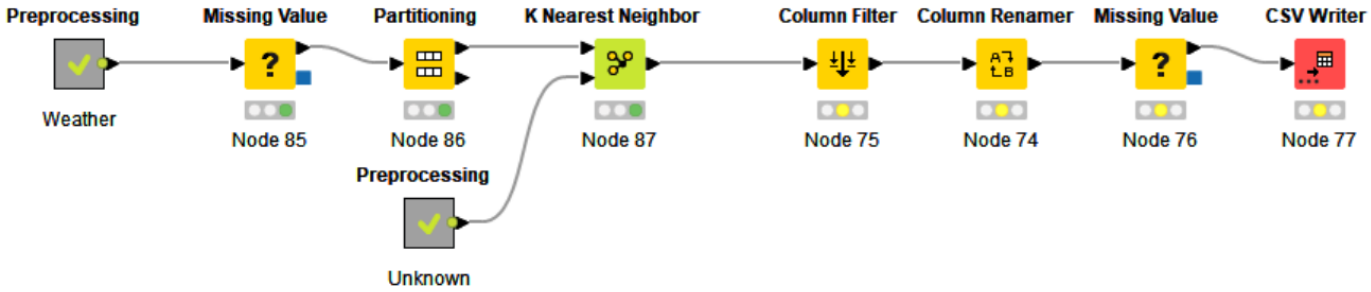
	TPos	FPos	TNeg	FNeg	Recall	Precision	Sensitivity	Specificity	F-Measure	Accuracy
0	9910	1484	1371	635	0.94	0.87	0.94	0.48	0.903	84.2%
1	1371	635	9910	1484	0.48	0.683	0.48	0.94	0.564	

Using RainTomorrow as the target column and Predict-RainTomorrow as the prediction column, the ROC Curve node reveals an AUC of 0.693 for the Decision Tree model. This value suggests the model has moderate predictive capabilities, performing noticeably better than random guessing (which would have an AUC of 0.5). However, an AUC of 0.693 indicates there is still substantial room for improvement, as a more effective model would achieve a higher AUC, closer to 1.0. Additionally, the model seems prone to generating a high number of false positives, which can impact its reliability by predicting rainfall when it may not occur.

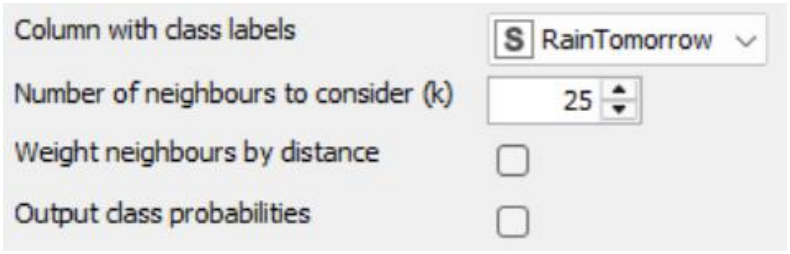


K-Nearest Neighbour

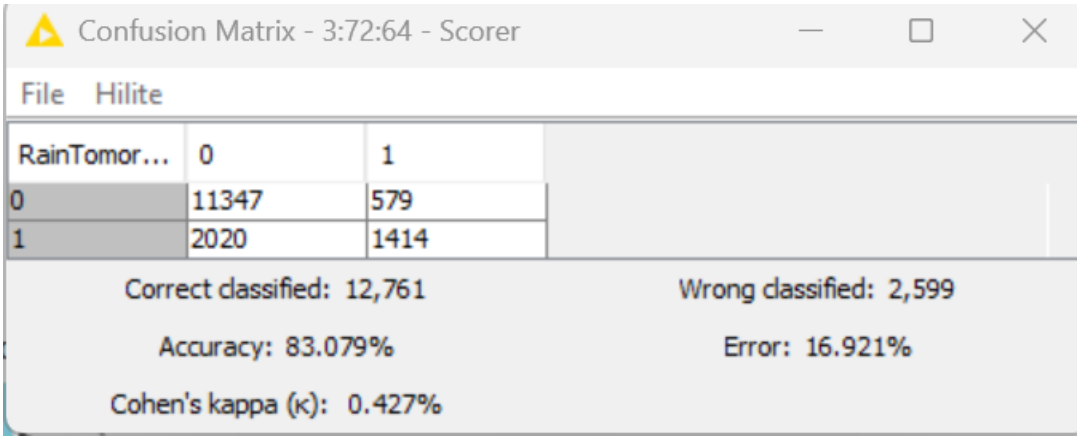




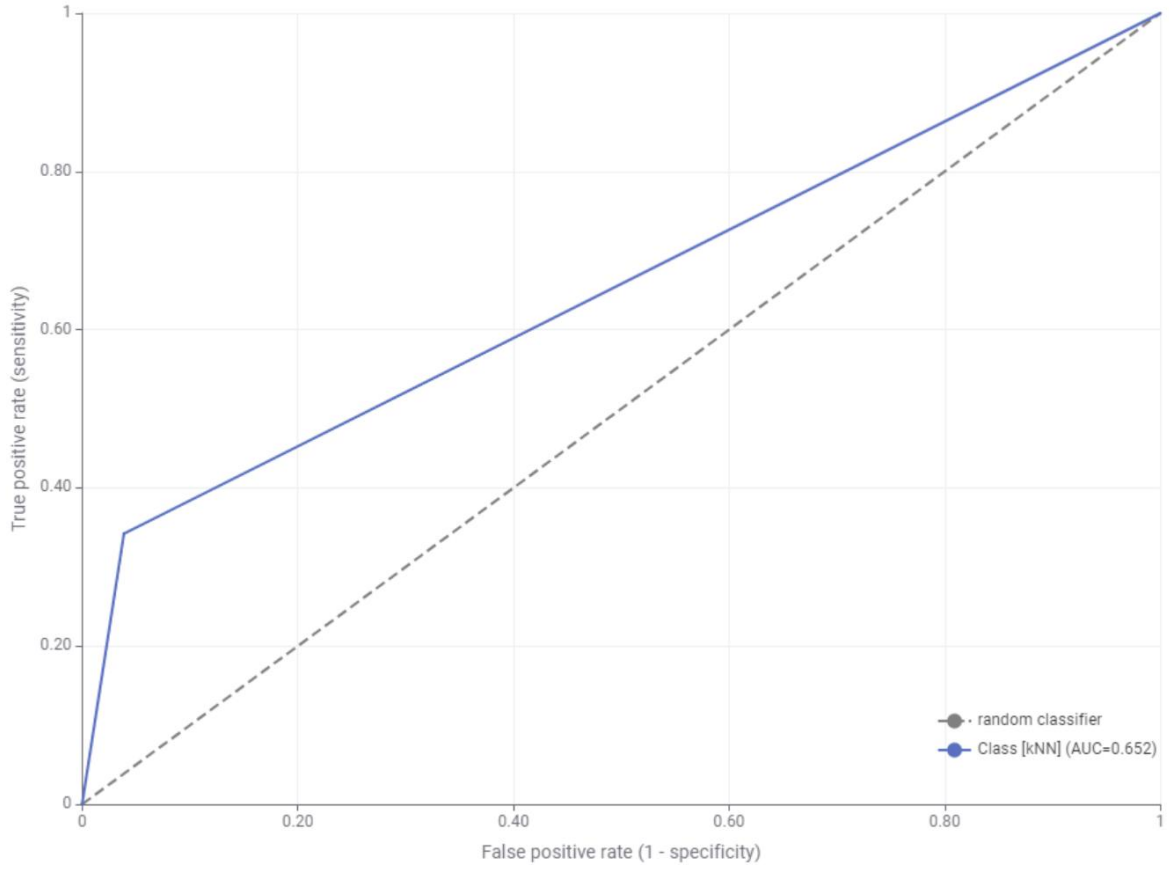
The setup for the K-Nearest Neighbours (KNN) model mirrors that of the Decision Tree, following the same preprocessing and partitioning steps, except instead of Draw Randomly, I use Linear Sampling. The training and test datasets from the Partitioning node are connected to their respective inputs in the KNN node, where the target column is set to RainTomorrow and the number of neighbours considered is 35. Before analysing the results, I ensure that any missing data is properly handled to avoid disruptions in the final model performance evaluation.



Using the Scorer node, I can visualize the result of this model:

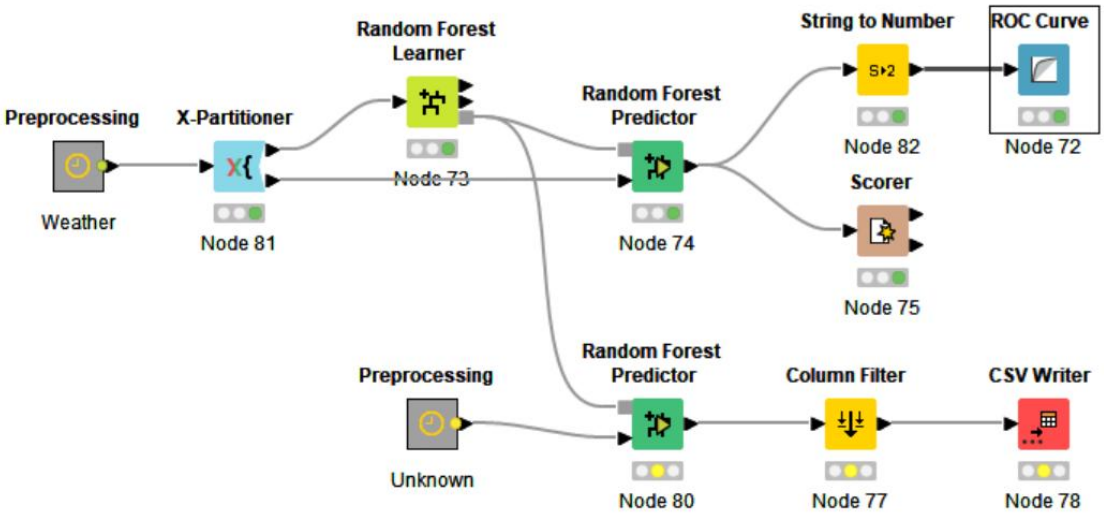


	TPos	FPos	TNeg	FNeg	Recall	Precision	Sensitivity	Specificity	F-Measure	Accuracy
0	11347	2020	1414	579	0.951	0.849	0.951	0.412	0.897	83.1%
1	1414	579	11347	2020	0.412	0.709	0.412	0.951	0.521	



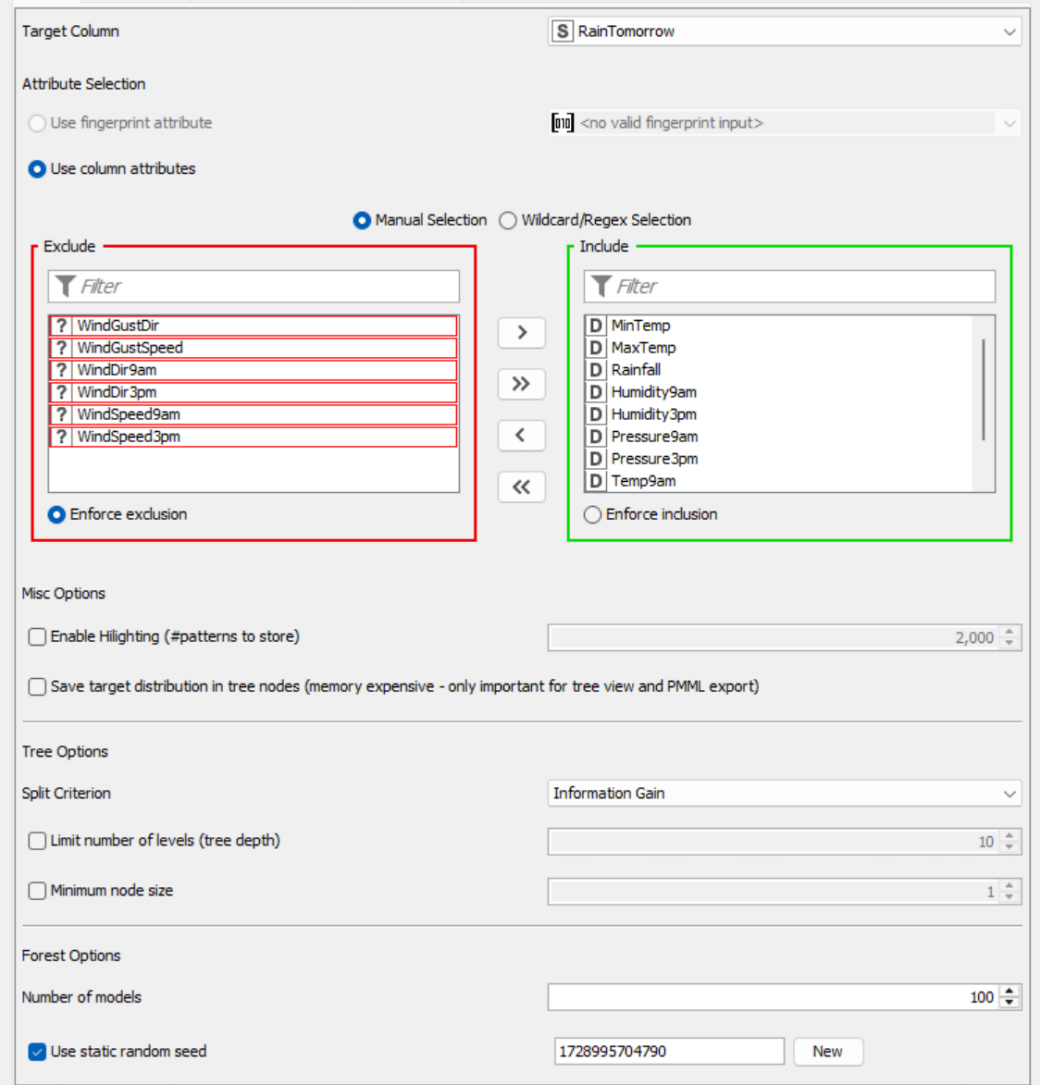
According to the ROC Curve node, the KNN classifier performs similarly to the Decision Tree classifier, with both showing a comparable AUC score of 0.652. The ROC plot for the KNN model closely resembles that of the Decision Tree, indicating that the KNN classifier also demonstrates a moderate balance between the true positive rate and false positive rate, with room for improvement in predictive accuracy.

Random Forest (RF)

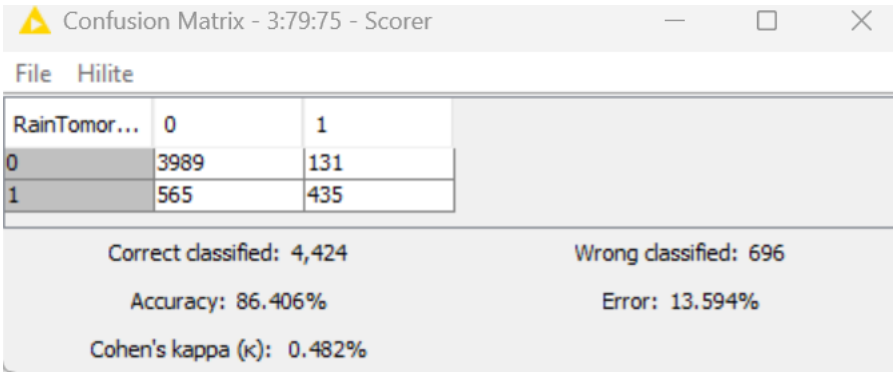


For this model, the preprocessing remains the same, however there is a slight change in terms of partitioning with the regular node being replaced by the X-Partitioner. This is to utilize cross-validation, allowing for multiple training and testing iterations across different folds of the dataset. This approach provides a more robust evaluation of the model's performance and helps reduce overfitting. I set the number of validations to 10 and use linear sampling. From here, the training set is fed to the Learner, while the testing set is fed to the predictor which inherits from the learner.

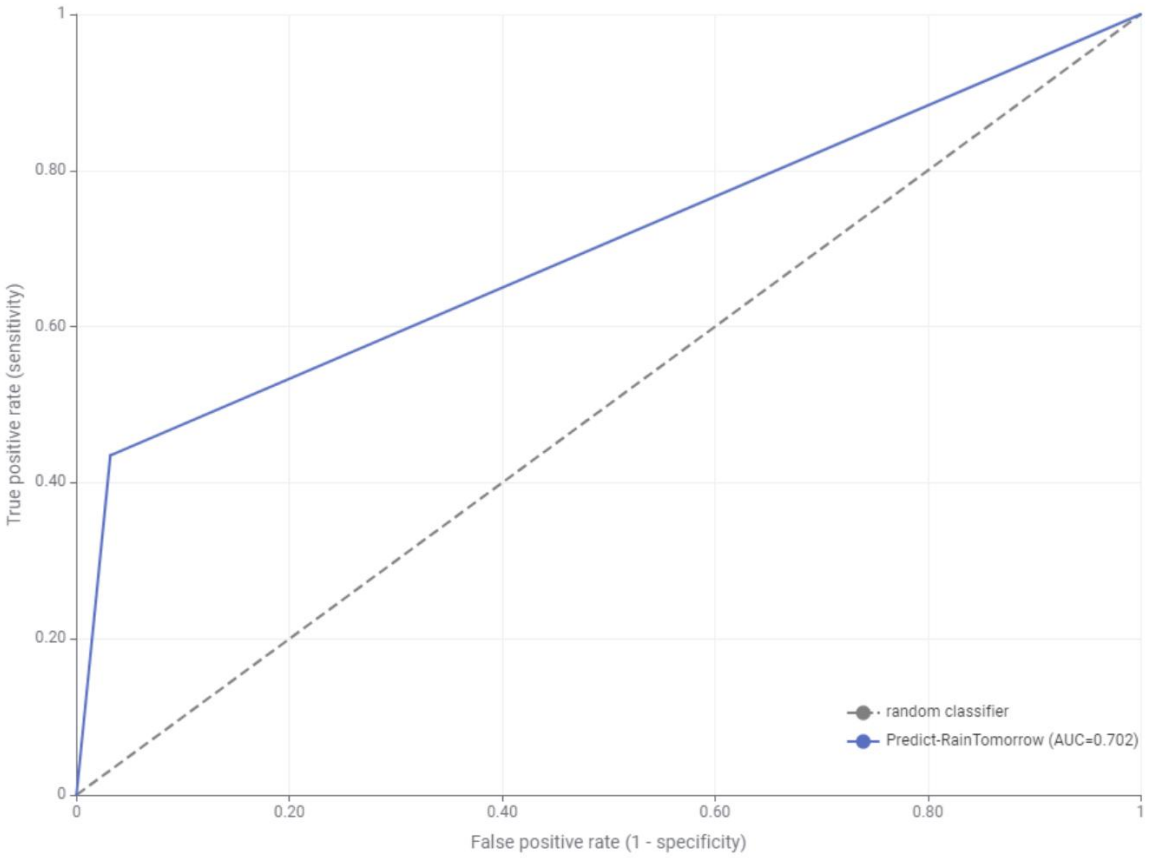
Here I use Information Gain as the Split Criterion since this produced the highest accuracy, and I also exclude any columns that are not numerical. The RF predictor then takes the data from the learner to make predictions on the dataset.



Using the Scorer and ROC Curve nodes, I can evaluate the model's performance.

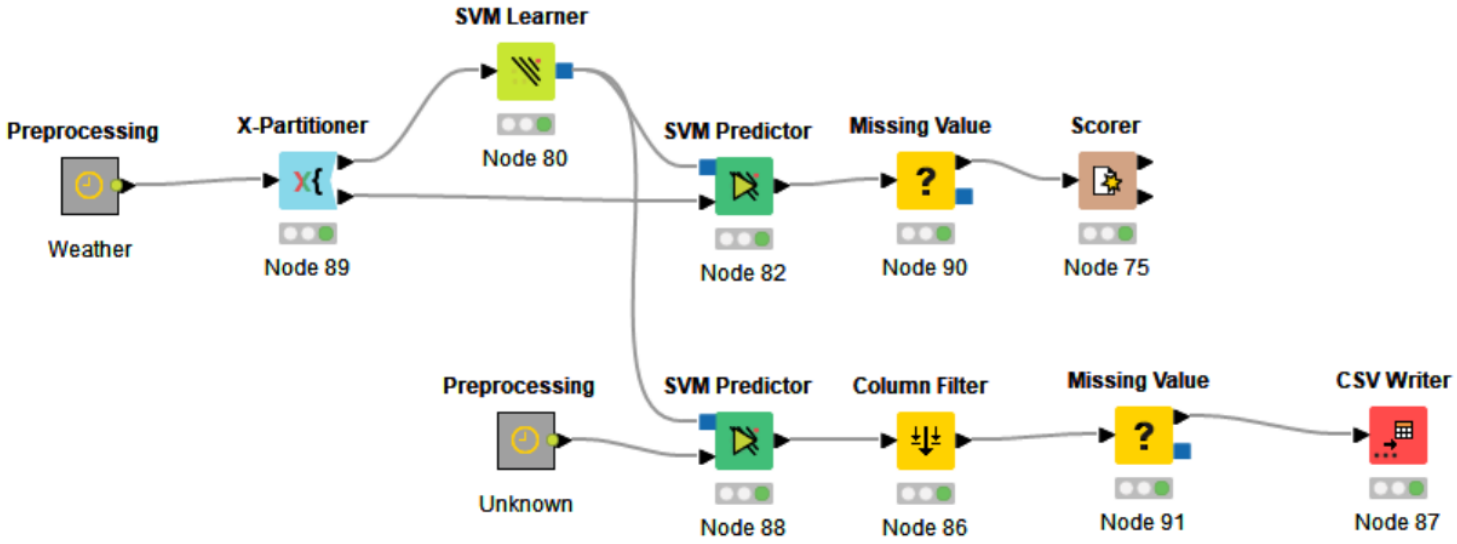


	TPos	FPos	TNeg	FNeg	Recall	Precision	Sensitivity	Specificity	F-Measure	Accuracy
0	3989	565	435	131	0.968	0.876	0.968	0.435	0.92	86.4%
1	435	131	3989	565	0.435	0.769	0.435	0.968	0.556	



This classifier has proven to be the most accurate among all those tested to date, achieving the highest accuracy and an AUC score of 0.702. This strong performance can be partially attributed to the model's evaluation on a smaller subset of data during each validation round, as implemented by the cross-validation method. By training and testing on various folds of the dataset, the classifier benefits from a more focused learning process, which can enhance its ability to generalize well. However, it is essential to consider that while the AUC score indicates good predictive capability, the apparent improvement in performance may be influenced by the reduced dataset size during validation, potentially masking how the model would perform on the complete dataset.

Support Vector Machine (SVM)

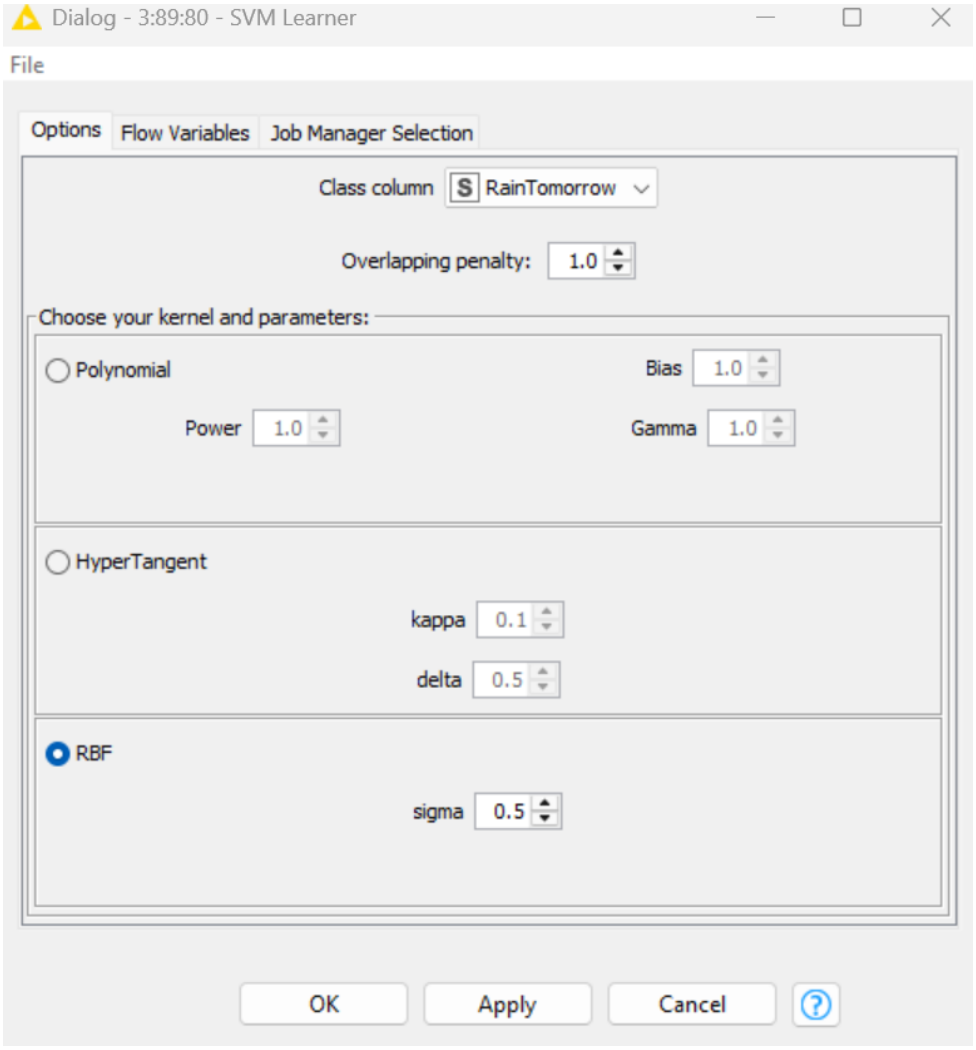


To prepare the data for the SVM model, I connect the pre-processed dataset to the X-Partitioner, configuring it with the following settings:

- Number of validations: 5
- Linear sampling

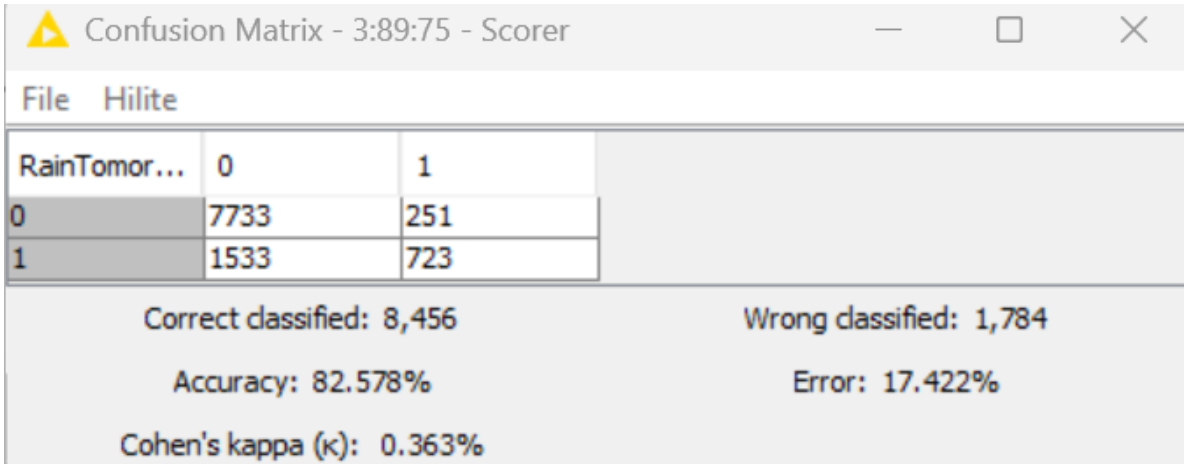
I chose linear sampling over other options to maintain class balance between the training and testing sets, which is crucial for achieving optimal model performance. Furthermore, linear sampling has demonstrated the highest accuracy score in previous evaluations.

Next, I connect the partitioned data to the SVM Learner, where the model will be trained on the input dataset to make predictions. I will operate the learner with the following parameters:

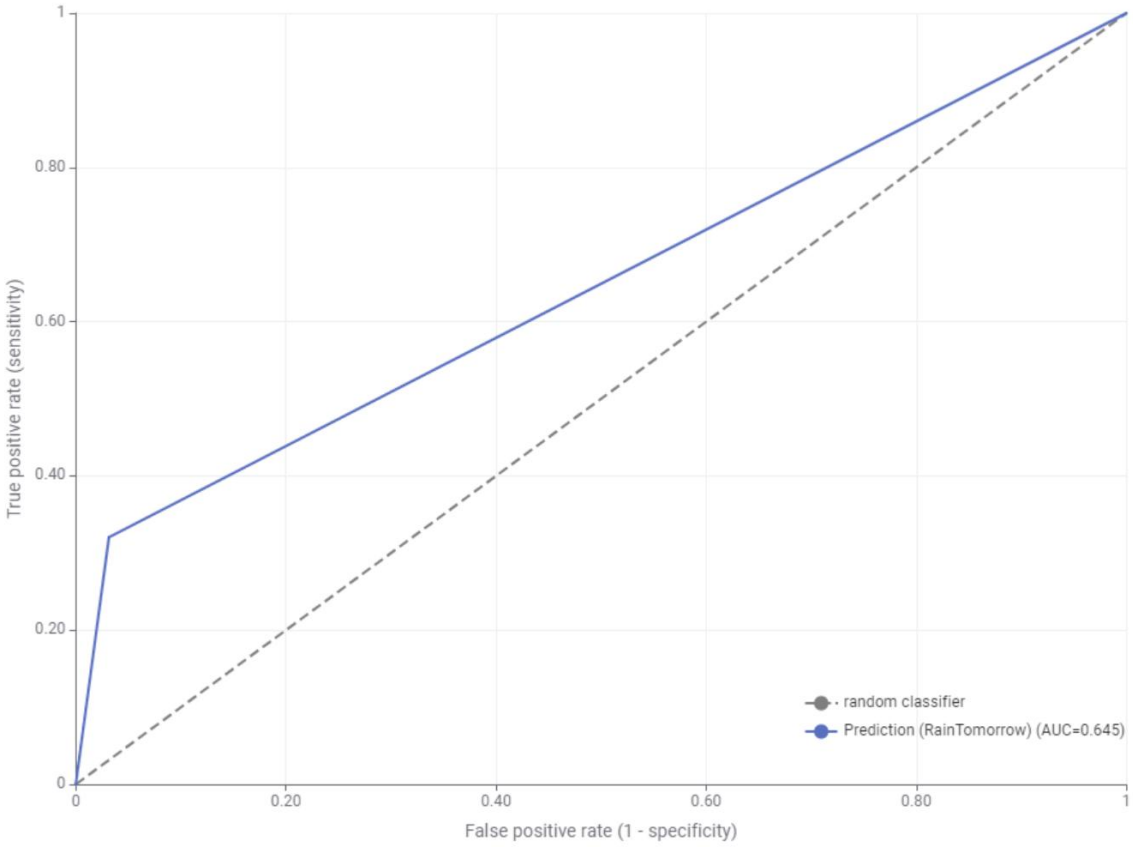


I use the RBF kernel with the sigma parameter set to 0.5 to balance model complexity and generalization, helping to prevent overfitting while effectively capturing key features in the data. Given that the prediction task is to forecast rain tomorrow, the class column is set to RainTomorrow, and I maintain an overlapping penalty of 1.0, which is the default value.

The results of this model:

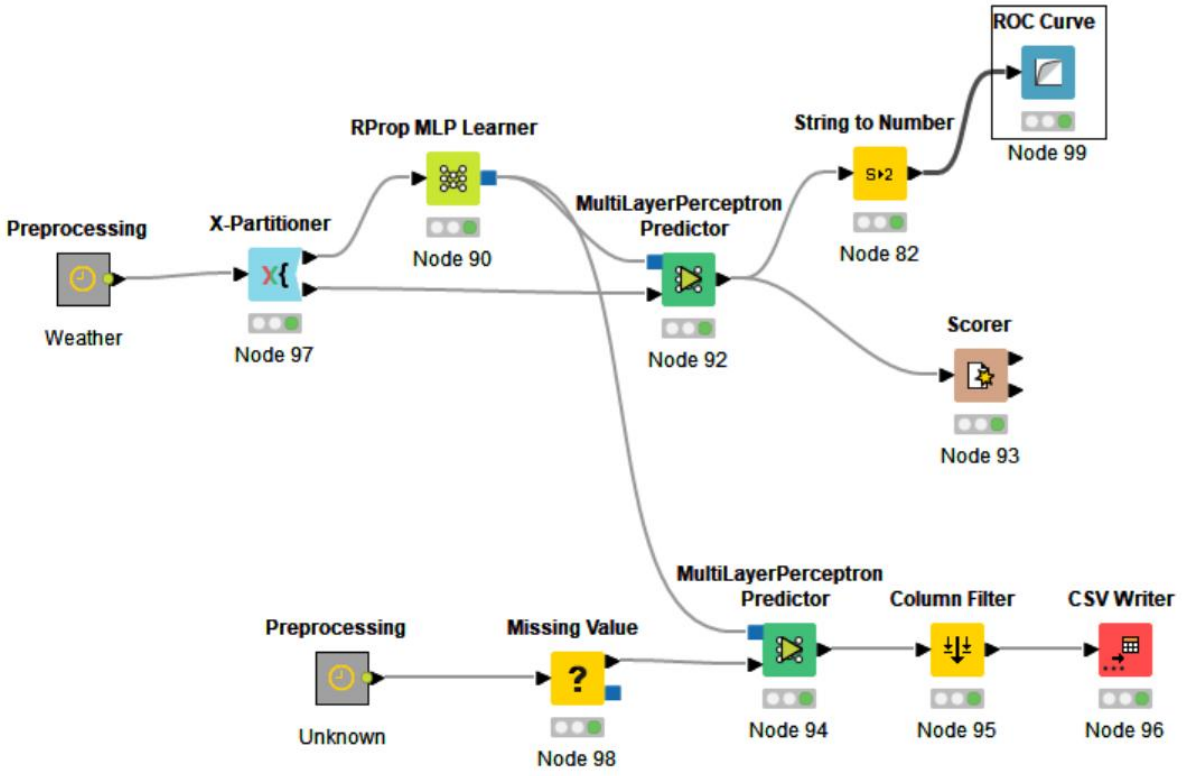


	TPos	FPos	TNeg	FNeg	Recall	Precision	Sensitivity	Specificity	F-Measure	Accuracy
0	7733	1533	723	251	0.969	0.835	0.969	0.32	0.897	82.6%
1	723	251	7733	1533	0.32	0.742	0.32	0.969	0.488	

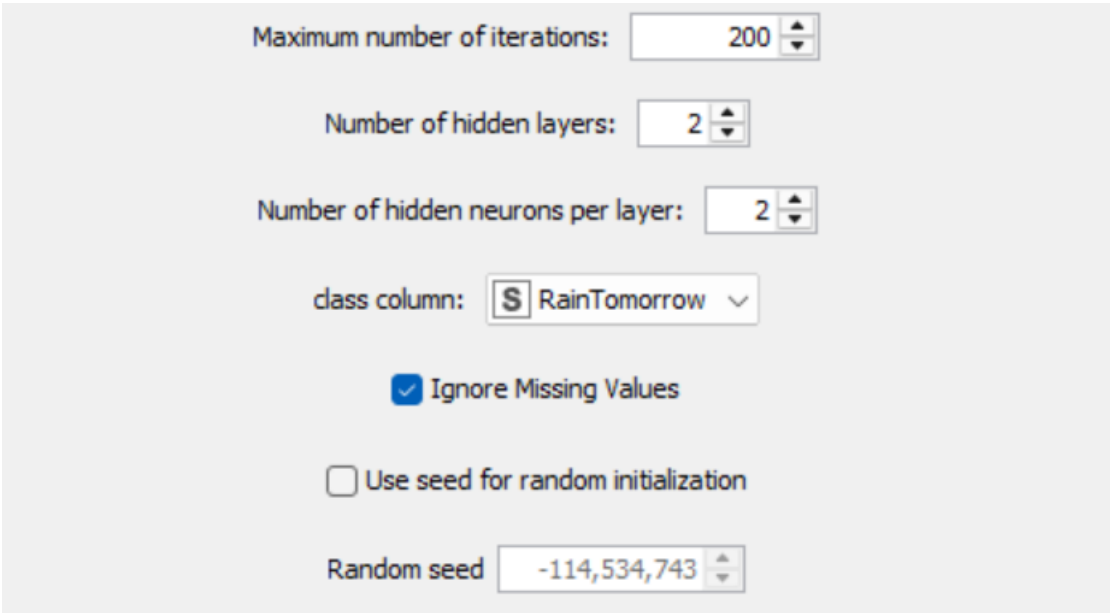


This model exhibits performance similar to the previous models; however, it underperforms in terms of true positivity compared to the Random Forest model. The AUC score also serves as a useful indicator of the model's effectiveness, showing a value comparable to those of the other models tested. Consequently, we can conclude that this model demonstrates moderate predictive capabilities.

Neural Network (MLP)

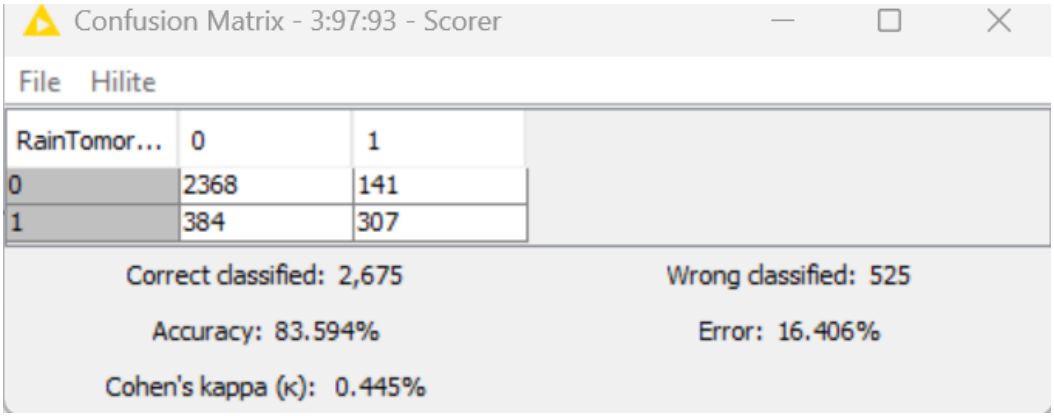


A Multilayer Perceptron (MLP) in KNIME is a type of neural network model that consists of multiple layers of interconnected neurons, designed for supervised learning tasks such as classification and regression. To prepare the data for this model, I follow a similar approach, however, I also include X-Partitioner which has been modified for 16 validations with Random Sampling. The training data is then connected to the Learner with the parameters:

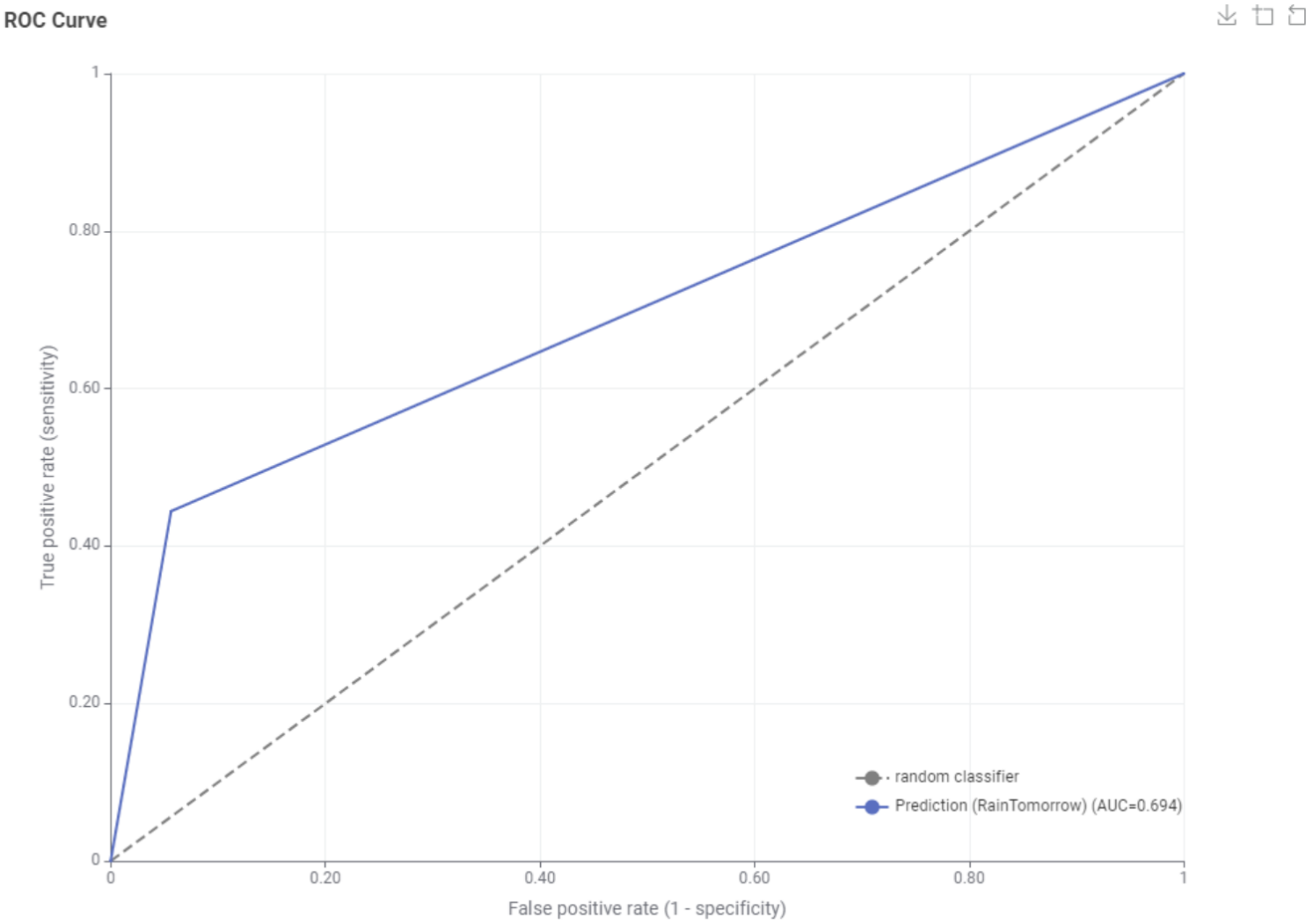


Following this, the testing data from the X-Partitioner as well as the MLP Learner are both connected to the MLP Predictor. The predictor applies a trained MLP neural network mode to new input data, in this case the test data, generating predictions for the Rain Tomorrow column.

Through the Scorer and ROC Curve nodes, we can analyse this model's performance.



	TPos	FPos	TNeg	FNeg	Recall	Precision	Sensitivity	Specificity	F-Measure	Accuracy
0	2368	384	307	141	0.944	0.86	0.944	0.444	0.9	83.6%
1	307	141	2368	384	0.444	0.685	0.444	0.944	0.539	



This model performs much like the previously discussed models, exhibiting a similar ROC Curve plot. With an AUC score of 0.694, the model does better than random guessing with a score over 0.5, but is not perfect. The curve being above the diagonal line means the model is finding some useful patterns to predict rainfall for the next day, but there's still room to improve its accuracy.

Results

To simplify the analysis, I have compiled the accuracy statistics for each model into a single table, allowing for easy comparisons and helping to identify the best course of action.

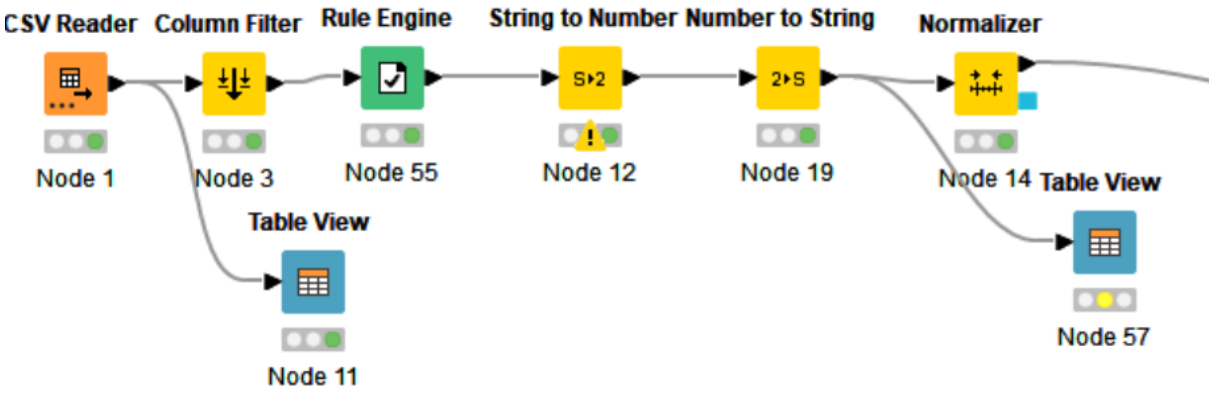
Decision Tree										
	TPos	**FPos**	**TNeg**	**FNeg**	**Recall**	**Precision**	**Sensitivity**	**Specificity**	**F-Measure**	**Accuracy**
0	9910	1484	1371	635	0.94	0.87	0.94	0.48	0.903	84.2%
1	1371	635	9910	1484	0.48	0.683	0.48	0.94	0.564	
K-Nearest Neighbour										
	TPos	**FPos**	**TNeg**	**FNeg**	**Recall**	**Precision**	**Sensitivity**	**Specificity**	**F-Measure**	**Accuracy**
0	11347	2020	1414	579	0.951	0.849	0.951	0.412	0.897	83.1%
1	1414	579	11347	2020	0.412	0.709	0.412	0.951	0.521	
Random Forest										
	TPos	**FPos**	**TNeg**	**FNeg**	**Recall**	**Precision**	**Sensitivity**	**Specificity**	**F-Measure**	**Accuracy**
0	3989	565	435	131	0.968	0.876	0.968	0.435	0.92	86.4%
1	435	131	3989	565	0.435	0.769	0.435	0.968	0.556	
Support Vector Machine										
	TPos	**FPos**	**TNeg**	**FNeg**	**Recall**	**Precision**	**Sensitivity**	**Specificity**	**F-Measure**	**Accuracy**
0	7733	1533	723	251	0.969	0.835	0.969	0.32	0.897	82.6%
1	723	251	7733	1533	0.32	0.742	0.32	0.969	0.488	
Neural Network										
	TPos	**FPos**	**TNeg**	**FNeg**	**Recall**	**Precision**	**Sensitivity**	**Specificity**	**F-Measure**	**Accuracy**
0	2368	384	307	141	0.944	0.86	0.944	0.444	0.9	83.6%
1	307	141	2368	384	0.444	0.685	0.444	0.944	0.539	

This table highlights a clear leader in performance. Random Forest outperforms all other models, achieving the highest scores in every column and the best accuracy overall. However, the Neural Network closely trails Random Forest in AUC score, making both models strong contenders for the final classifier selection.

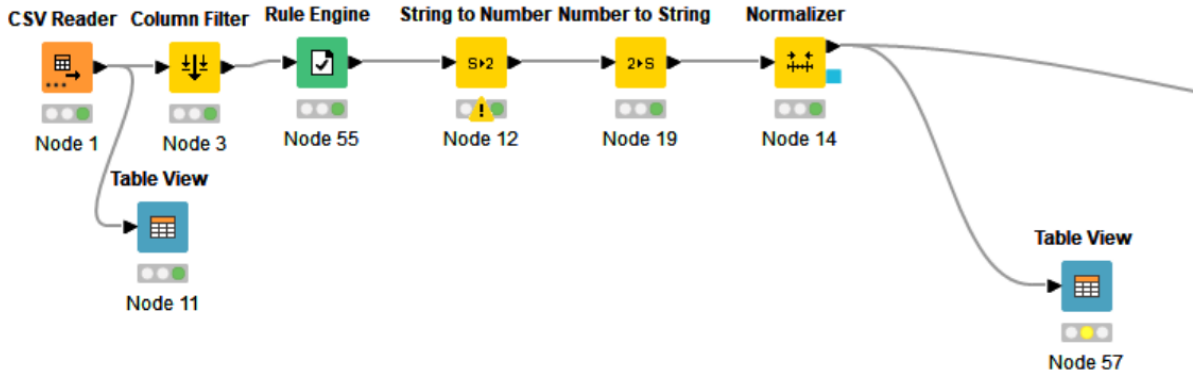
Kaggle Submission:

- Using the Random Forest Model, the Unknown Dataset returned a Kaggle score of 0.71219, meaning it correctly predicted 71% of the dataset.
- Neural Network on the other hand, returned a Kaggle score of 0.75932 making this model the highest performing.

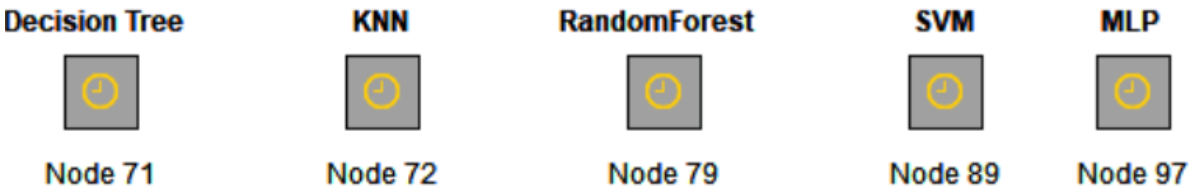
Appendix



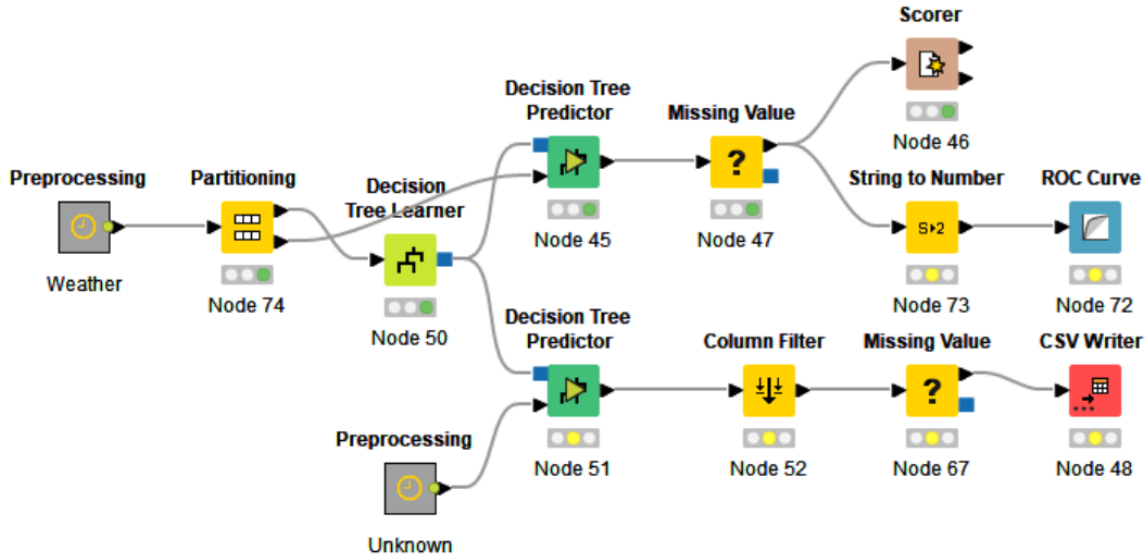
WeatherData pre-processing workflow.



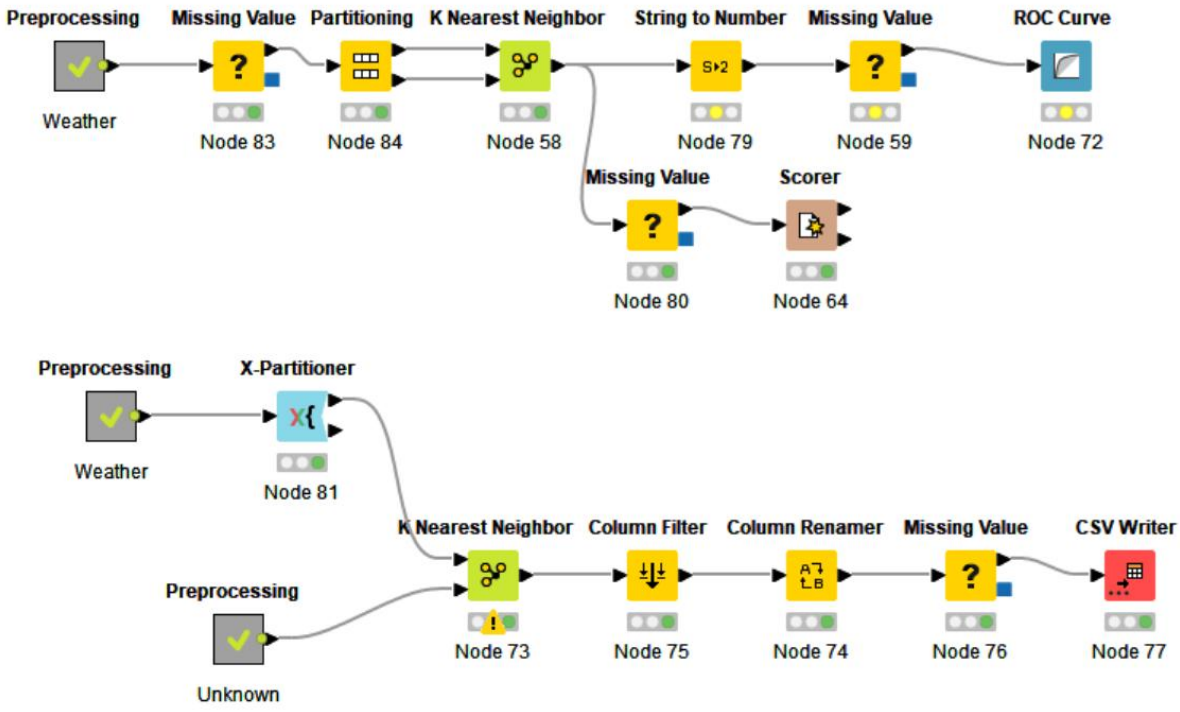
UnknownData pre-processing workflow.



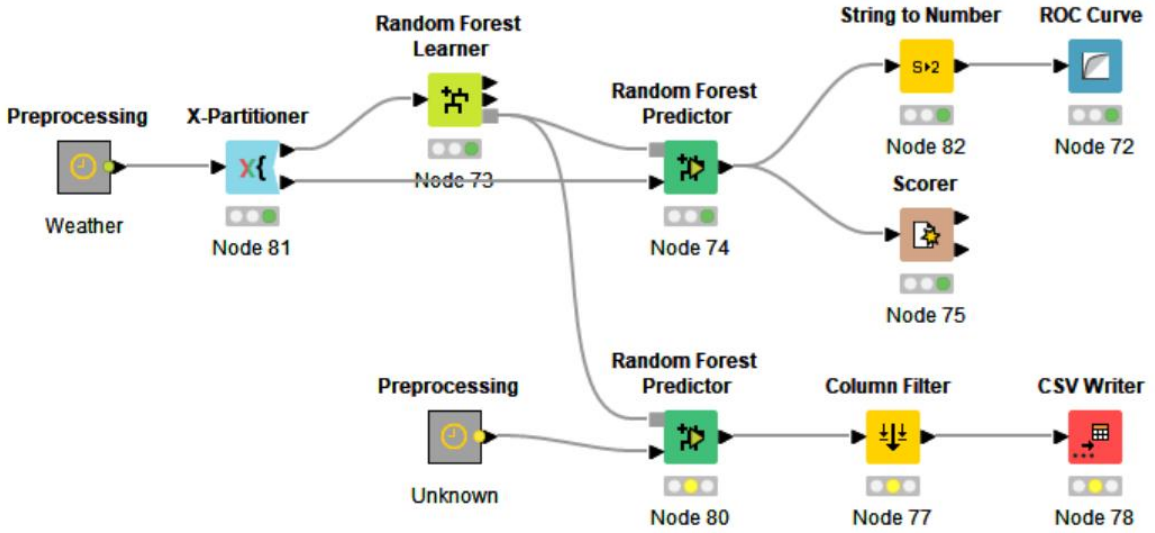
Models used, each combined into a meta node.



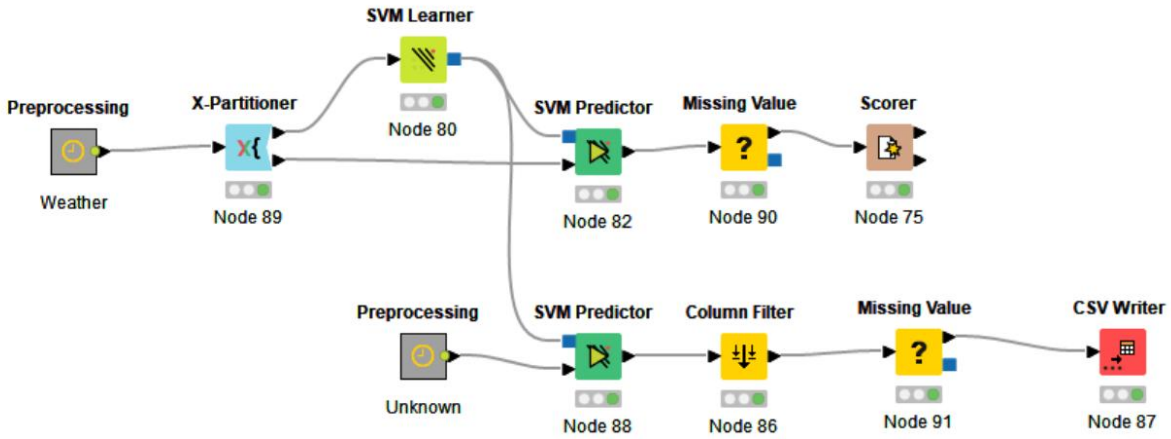
Decision Tree model workflow.



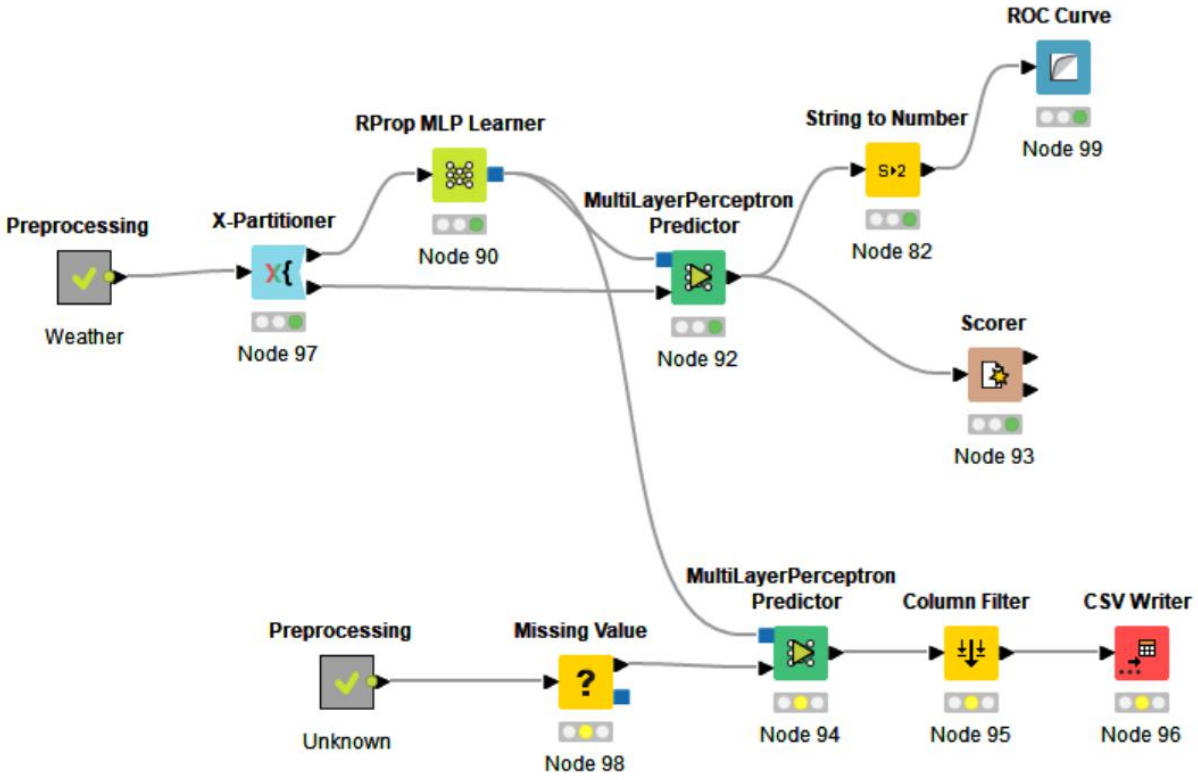
K Nearest Neighbour model workflow.



Random Forest model workflow.



Support Vector Machine model workflow.



Neural Network model workflow.