

# MACHINE LEARNING APPLICATION ON RAPIDMINER AND WEKA

## *Team Members:*

Name: Wajd Bandar Alharbi  
ID: 2007057

Name: Renad Baghdadi  
ID: 2006538

# Task Assignment

Task	Wajd Alharbi	Renad Baghdadi
Introduction	50%	50%
The chosen dataset	50%	50%
The machine learning	100%	0%
Linear Regression algorithm	100%	0%
Multilayer Perceptron algorithm (Neural Net)	0%	100%
Conclusion	50%	50%

## Table of Contents

1-	Introduction .....	4
2-	Purpose of the project .....	4
3-	The chosen dataset .....	4
•	Metro Interstate Traffic Volume Data Set.....	4
•	Attribute Information .....	5
4-	The machine learning method .....	6
5-	Linear Regression Algorithm: .....	7
•	Result of the experiment .....	8
6-	Multilayer Perceptron Algorithm (Neural Net): .....	12
•	Result of the experiment .....	13
7-	Conclusion .....	17
8-	References .....	19
9-	Appendix .....	20
•	Linear Regression .....	20
•	Multilayer Perceptron.....	26
•	Designs and Steps .....	32

# 1-Introduction

Artificial Intelligence and computer science's field of machine learning is concerned with utilizing algorithms and data to mimic how individuals learn in order to increase accuracy over time. Machine learning algorithms are used to generate a prediction, regression, or classification. Based on certain input data, which may be labeled or unlabeled, our algorithms will offer an estimate about a pattern in the data.

## 2-Purpose of the project

This project's objective is to employ the tools to apply machine learning algorithms to massive amounts of data. The methods we chose include the Multilayer Perceptron algorithm and the Linear Regression algorithm, both of which leverage machine learning tools like Rapid-miner and Weka. Examining how various validations use the same method and the same data to get different results that rely on the qualities of the validations is the aim.

## 3-The chosen dataset

- Metro Interstate Traffic Volume Data Set

Traffic volume is determined by observing the temperature, weather, if there is snow, clouds, or rain, and if there is a holiday or not. This data set uses anomaly detection, a method that makes use of AI to spot unusual activity in comparison to a pre-existing pattern. An anomaly is something that deviates from the accepted baseline pattern. The AI in Dynatrace automatically creates baselines, finds abnormalities, fixes underlying causes, and notifies users. Metro Interstate Traffic Volume Data Set is a multivariate, sequential, time-Series type of data set. The characteristics of its 9 attributes are Integer and Real for the most part, but it also included date and time, as well as polynomial attributes. The tasks associated with the data is Regression, for this reason, we chose regression algorithms for our regression problem.

- Attribute Information

1. holiday Categorical US National holidays plus regional holiday, Minnesota State Fair
2. temp Numeric Average temp in kelvin
3. rain\_1h Numeric Amount in mm of rain that occurred in the hour
4. snow\_1h Numeric Amount in mm of snow that occurred in the hour
5. clouds\_all Numeric Percentage of cloud cover
6. weather\_main Categorical Short textual description of the current weather
7. weather\_description Categorical Longer textual description of the current weather
8. date\_time DateTime Hour of the data collected in local CST time
9. traffic\_volume Numeric Hourly I-94 ATR 301 reported westbound traffic volume

- Sample of data set on Rapid Miner

Row No.	traffic_volu...	holiday	temp	rain_1h	snow_1h	clouds_all	weather_m...	weather_de...	date_time
1	5545	None	288.280	0	0	40	Clouds	scattered clo...	Feb 10, 2012 ...
2	4516	None	289.360	0	0	75	Clouds	broken clouds	Feb 10, 2012 ...
3	4767	None	289.580	0	0	90	Clouds	overcast clou...	Feb 10, 2012 ...
4	5026	None	290.130	0	0	90	Clouds	overcast clou...	Feb 10, 2012 ...
5	4918	None	291.140	0	0	75	Clouds	broken clouds	Feb 10, 2012 ...
6	5181	None	291.720	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
7	5584	None	293.170	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
8	6015	None	293.860	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
9	5791	None	294.140	0	0	20	Clouds	few clouds	Feb 10, 2012 ...
10	4770	None	293.100	0	0	20	Clouds	few clouds	Feb 10, 2012 ...
11	3539	None	290.970	0	0	20	Clouds	few clouds	Feb 10, 2012 ...
12	2784	None	289.380	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
13	2361	None	288.610	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
14	1529	None	287.160	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
15	963	None	285.450	0	0	1	Clear	sky is clear	Feb 10, 2012 ...
16	506	None	284.630	0	0	1	Clear	sky is clear	Mar 10, 2012 ...
17	321	None	283.470	0	0	1	Clear	sky is clear	Mar 10, 2012 ...
18	273	None	281.180	0	0	1	Clear	sky is clear	Mar 10, 2012 ...

## 4-The machine learning method

Regression is a method for determining how independent traits or variables relate to a dependent feature or result. It is a technique for machine learning predictive modeling, where an algorithm is utilized to estimate continuous outcomes. One of the most popular uses of machine learning models, particularly in supervised machine learning, is to solve regression issues. The link between an outcome or result of events occurring and independent factors is something that algorithms are designed to grasp. The model may then be used to forecast the results of fresh, unforeseen input data or to complete a data gap.

There are many error metrics that are commonly used for evaluating and reporting the performance of a regression model; they are:

- Correlation Coefficient
- Root Mean Squared Error (RMSE).
- Mean Absolute Error (MAE)
- Mean Squared Error (MSE).
- Mean Absolute error.
- Relative Absolute Error.

However, the most common ones are **Root Mean Squared Error**, and **Correlation**, which are the ones we will be using for comparison. The lower the Root mean squared error, the better the model is. The correlation coefficient ranges between 0 – 1, and the closer the correlation is to 1, the better the model. A correlation of 1 means 100% accuracy.

In this project we are going to use Two regression task suitable algorithms:

## 5- Linear Regression Algorithm:

---

An algorithm for machine learning based on supervised learning is linear regression. Given that it is one of the most extensively utilized regression analysis approaches, it is frequently employed for tasks and issues involving regression. Based on independent variables, regression models a goal prediction value. Finding the connection between variables and predicting is its main purpose. The linear regression procedure, often known as linear regression, illustrates a linear connection between one or more independent variables ( $x$ ) and a dependent variable ( $y$ ). As a result of displaying a linear connection, linear regression may be used to determine how the values of the dependent variable variable changes in proportion to the values of the independent variable.

- Result of the experiment

1- Split validation:

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Split validation</b>	
	Rapid Miner	Weka
<b>Split Ratio</b>	70%	70%
<b>Root Mean Squared Error</b>	1968.796	1960.8132
<b>Correlation Coefficient</b>	0.146	0.1589
<b>Mean Absolute Error</b>	1717.258	1709.0488
<b>Relative absolute error</b>	215.70%	98.168 %



## 2- Split validation :

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Split validation</b>	
	Rapid Miner	Weka
<b>Split Ratio</b>	80%	80%
<b>Root Mean Squared Error</b>	1972.306	1960.5496
<b>Correlation Coefficient</b>	0.142	0.1477
<b>Mean Absolute Error</b>	1717.684	1707.4839
<b>Relative absolute error</b>	187.75%	98.3212 %

1- Cross validation :

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Cross validation</b>	
	Rapid Miner	Weka
<b>FOLDS</b>	10	10
<b>Root Mean Squared Error</b>	2945.127	
<b>Correlation Coefficient</b>	0.137	0.0076
<b>Mean Absolute Error</b>	1730.589	1730.2904
<b>Relative absolute error</b>	295.54%	99.1521 %

## 2- Cross validation :

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Cross validation</b>	
	Rapid Miner	Weka
<b>FOLDS</b>	20	20
<b>Root Mean Squared Error</b>	2676.623	4143.6945
<b>Correlation Coefficient</b>	0.144	0.0076
<b>Mean Absolute Error</b>	1730.178	1730.3893
<b>Relative absolute error</b>	295.48%	99.1574 %

## **6-Multilayer Perceptron Algorithm (Neural Net):**

A feed-forward artificial neural network that produces a set of outputs from a collection of inputs and it is a supervised learning algorithm called a multilayer perceptron (MLP), MLP is characterized by several layers of inputs, it consists of three types of layers, the input layer, output layer and hidden layer. MPL is a deep learning method and it trains the network through backpropagation. MPL was created to approximate any continuous function and can resolve issues that cannot be divided linearly. Pattern classification, recognition, prediction, and approximation are the main applications of MLP.

- Result of the experiment

1- Split validation :

In Rapid Miner and In Weka “Look at the appendix “

## Rapid Miner vs Weka

	<b>Split validation</b>	
	Rapid Miner	Weka
<b>Split Ratio</b>	70%	70%
<b>Root Mean Squared Error</b>	1982.307	1966.8953
<b>Correlation Coefficient</b>	0.164	0.1415
<b>Mean Absolute Error</b>	1730.491	1719.0456
<b>Relative absolute error</b>	196.56%	98.7422 %

## 2- Split validation :

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Split validation</b>	
	Rapid Miner	Weka
<b>Split Ratio</b>	80%	80%
<b>Root Mean Squared Error</b>	1972.886	2082.3326
<b>Correlation Coefficient</b>	0.156	0.129
<b>Mean Absolute Error</b>	1722.592	1756.4698
<b>Relative absolute error</b>	179.24%	101.1419 %

1- Cross validation :

In Rapid Miner and In Weka “Look at the appendix “

## Rapid Miner vs Weka

	<b>Cross validation</b>	
	Rapid Miner	Weka
<b>FOLDS</b>	10	10
<b>Root Mean Squared Error</b>	2029.831	2108.8495
<b>Correlation Coefficient</b>	0.171	0.0464
<b>Mean Absolute Error</b>	1750.541	1805.6282
<b>Relative absolute error</b>	287.83%	103.4693 %

## 2- Cross validation :

In Rapid Miner and In Weka “Look at the appendix “

### Rapid Miner vs Weka

	<b>Cross validation</b>	
	Rapid Miner	Weka
<b>FOLDS</b>	20	20
<b>Root Mean Squared Error</b>	2007.699	2156.026
<b>Correlation Coefficient</b>	0.167	0.0369
<b>Mean Absolute Error</b>	1734.916	1827.4707
<b>Relative absolute error</b>	290.46%	104.7205 %



## 7-Conclusion

		For Linear Regression			
		Split validation 70%	Split validation 80%	Cross validation 10 folds	Cross validation 20 folds
<b>Rapid Miner</b>	<b>Root Mean Squared Error</b>	<b>1968.796</b>	<b>1972.306</b>	<b>2945.127</b>	<b>2676.623</b>
<b>Weka</b>	<b>Root Mean Squared Error</b>	<b>1960.8132</b>	<b>1960.54</b>	<b>4161.0686</b>	<b>4143.6945</b>

As the previous result for the linear regression algorithm, we approved a difference between each program in predicting the target. In Split validation, the Root mean squared error values are better in Weka than in RapidMiner. However, in Cross validation, we noticed that RapidMiner was better than Weka in Root mean square error.

		For Multilayer Perceptron			
		Split validation 70%	Split validation 80%	Cross validation 10 folds	Cross validation 20 folds
<b>Rapid Miner</b>	<b>Root Mean Squared Error</b>	<b>1982.307</b>	<b>1972.886</b>	<b>2029.831</b>	<b>2007.699</b>
<b>Weka</b>	<b>Root Mean Squared Error</b>	<b>1966.8953</b>	<b>2082.3326</b>	<b>2108.8495</b>	<b>2156.026</b>

As the previous result for the Multilayer Perceptron Algorithm (Neural Net) we approved a difference between each program in predicting the target. In Split validation and the Root mean squared error we noticed both values were better in RapidMiner than Weka

In conclusion, Linear regression seems to perform better in split validation than Multilayer perception on both RapidMiner and Weka. On the contrary, Multilayer perceptron performs better in cross validation than linear regression did on all programs. However, the root mean squared error in cross validation in Weka for linear regression spiked to the worst case, so we will only be looking at the good side, (split validation). Linear regression has noticeably performed better in split validation, in comparison to multilayer perceptron's best-case performance in any of the validation methods and programs. For this reason, we have considered linear regression to be the better algorithm in performance.

## 8-References

*Anomaly detection powered by ai.* Dynatrace. (2022, October 13). Retrieved November 4, 2022, from [https://www.dynatrace.com/monitoring/platform/artificial-intelligence/anomaly-detection/?utm\\_source=google&utm\\_medium=cpc&utm\\_term=ai%20anomaly%20detection&utm\\_campaign=me-aiops-aiops&utm\\_content=none&gclid=Cj0KCQjwteOaBhDuARIsADBqReiqBeH0CLIUAGmZzawbp4wQZsyzkL\\_zbDu\\_FtJ\\_GYL1VVROjlraZx0aAvX0EALw\\_wcB&gclidsrc=aw.ds](https://www.dynatrace.com/monitoring/platform/artificial-intelligence/anomaly-detection/?utm_source=google&utm_medium=cpc&utm_term=ai%20anomaly%20detection&utm_campaign=me-aiops-aiops&utm_content=none&gclid=Cj0KCQjwteOaBhDuARIsADBqReiqBeH0CLIUAGmZzawbp4wQZsyzkL_zbDu_FtJ_GYL1VVROjlraZx0aAvX0EALw_wcB&gclidsrc=aw.ds)

*Metro Interstate Traffic Volume Dataset.* UCI Machine Learning Repository: Metro Interstate Traffic Volume Data Set. (n.d.). Retrieved November 4, 2022, from <https://archive.ics.uci.edu/ml/datasets/Metro+Interstate+Traffic+Volume>

Seldon. (2021, October 29). Machine learning regression explained. Seldon. Retrieved November 4, 2022, from <https://www.seldon.io/machine-learning-regression-explained>

*Multilayer Perceptron.* Multilayer Perceptron - an overview | ScienceDirect Topics. (n.d.). Retrieved November 4, 2022, from <https://www.sciencedirect.com/topics/computer-science/multilayer-perceptron>

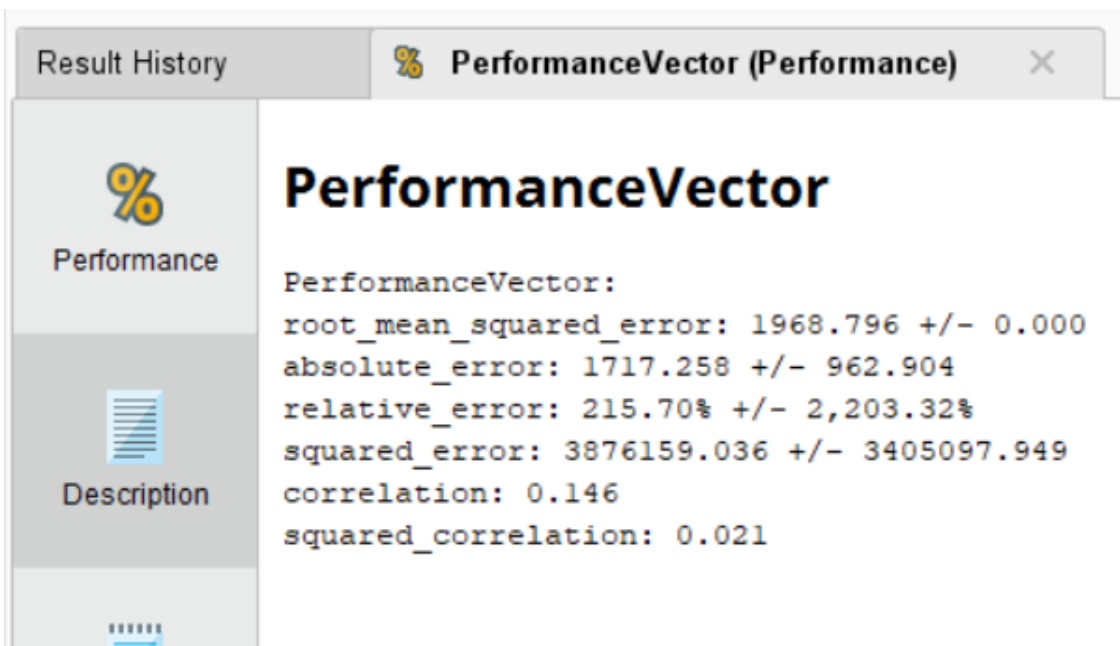
*Multi-layer perceptron learning in tensorflow.* GeeksforGeeks. (2021, November 5). Retrieved November 4, 2022, from <https://www.geeksforgeeks.org/multi-layer-perceptron-learning-in-tensorflow/>

## 9-Appendix

- **Linear Regression**

1- Split validation for **Linear Regression**:

In Rapid Miner With split ratio 0.7

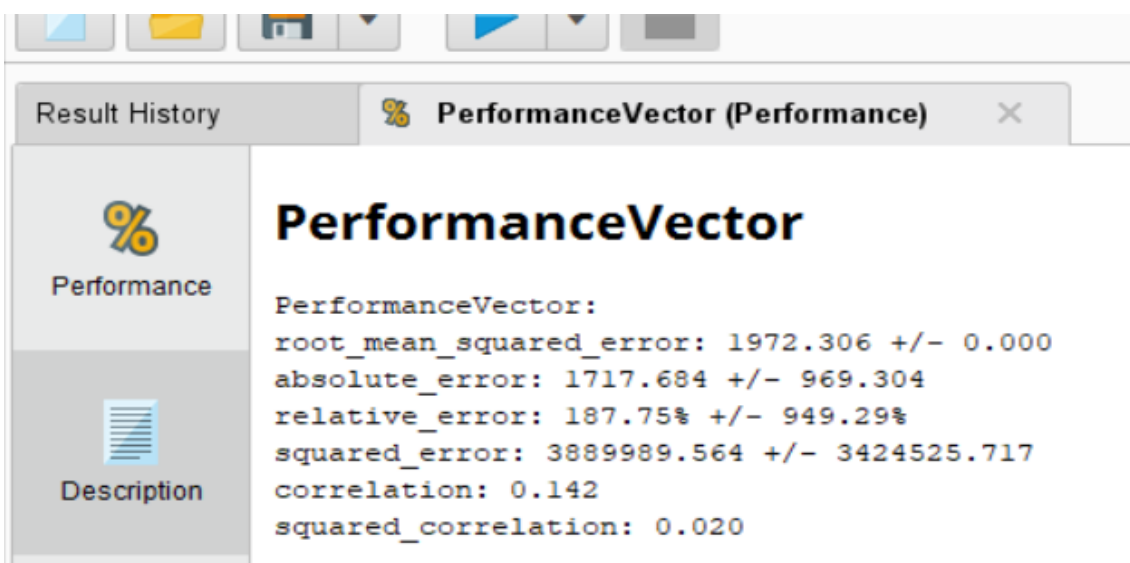


The screenshot shows the 'PerformanceVector (Performance)' window in Rapid Miner. The left sidebar has 'Performance' selected. The main area displays the following metrics:

```
PerformanceVector:
root_mean_squared_error: 1968.796 +/- 0.000
absolute_error: 1717.258 +/- 962.904
relative_error: 215.70% +/- 2,203.32%
squared_error: 3876159.036 +/- 3405097.949
correlation: 0.146
squared_correlation: 0.021
```

1- Split validation for **Linear Regression**:

In Rapid Miner With split ratio 0.8



The screenshot shows the 'PerformanceVector (Performance)' window in Rapid Miner for a split ratio of 0.8. The left sidebar has 'Performance' selected. The main area displays the following metrics:

```
PerformanceVector:
root_mean_squared_error: 1972.306 +/- 0.000
absolute_error: 1717.684 +/- 969.304
relative_error: 187.75% +/- 949.29%
squared_error: 3889989.564 +/- 3424525.717
correlation: 0.142
squared_correlation: 0.020
```

## 2- Split validation for **Linear Regression**:

In weka With split ratio 0.7

The screenshot shows the Weka software interface. At the top, there are three tabs: "Associate", "Select attributes", and "Visualize". Below these, a command line shows: `> 0 -R 1.0E-8 -num-decimal-places 4`. The main window is titled "Classifier output". It displays the following text:

```
temp
rain_1h
snow_1h
clouds_all
traffic_volume
Test mode:    split 70.0% train, remainder test

=== Classifier model (full training set) ===

Linear Regression Model

traffic_volume =

    20.6425 * temp +
    4.1343 * clouds_all +
    -2749.0607

Time taken to build model: 0.25 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0.04 seconds

=== Summary ===

Correlation coefficient          0.1589
Mean absolute error             1709.0488
Root mean squared error         1960.8132
Relative absolute error         98.168 %
Root relative squared error     98.7324 %
Total Number of Instances      14461
```

On the left side of the window, there is a vertical list of buttons: "on", "ion", "ion", "ion".

## 2- Split validation for **Linear Regression**:

In weka With split ratio 0.8

### Classifier output

```
temp
rain_1h
snow_1h
clouds_all
traffic_volume
Test mode:    split 80.0% train, remainder test

=== Classifier model (full training set) ===

Linear Regression Model

traffic_volume =

    20.6425 * temp +
    4.1343 * clouds_all +
    -2749.0607

Time taken to build model: 0.13 seconds

=== Evaluation on test split ===

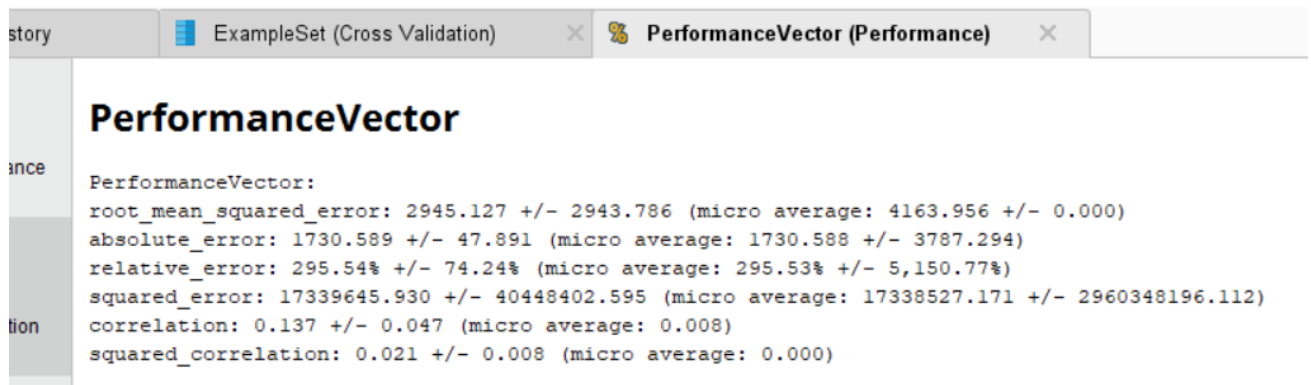
Time taken to test model on test split: 0.01 seconds

=== Summary ===

Correlation coefficient          0.1477
Mean absolute error             1707.4839
Root mean squared error         1960.5496
Relative absolute error         98.3212 %
Root relative squared error     98.9048 %
Total Number of Instances      9641
```

## 1- Cross validation for **Linear Regression**:

In Rapid Miner With 10 Folds

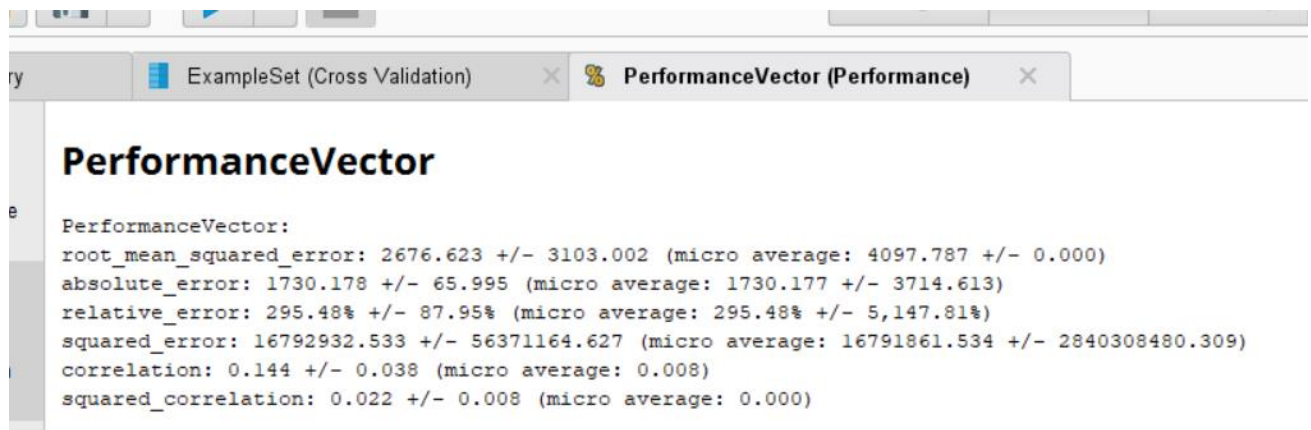


The screenshot shows the 'PerformanceVector (Performance)' window in Rapid Miner. The window title bar includes 'ExampleSet (Cross Validation)' and 'PerformanceVector (Performance)'. The main content area is titled 'PerformanceVector' and displays the following performance metrics:

```
PerformanceVector:
root_mean_squared_error: 2945.127 +/- 2943.786 (micro average: 4163.956 +/- 0.000)
absolute_error: 1730.589 +/- 47.891 (micro average: 1730.588 +/- 3787.294)
relative_error: 295.54% +/- 74.24% (micro average: 295.53% +/- 5,150.77%)
squared_error: 17339645.930 +/- 40448402.595 (micro average: 17338527.171 +/- 2960348196.112)
correlation: 0.137 +/- 0.047 (micro average: 0.008)
squared_correlation: 0.021 +/- 0.008 (micro average: 0.000)
```

## 1- Cross validation for **Linear Regression**:

In Rapid Miner With 20 Folds



The screenshot shows the 'PerformanceVector (Performance)' window in Rapid Miner. The window title bar includes 'ExampleSet (Cross Validation)' and 'PerformanceVector (Performance)'. The main content area is titled 'PerformanceVector' and displays the following performance metrics:

```
PerformanceVector:
root_mean_squared_error: 2676.623 +/- 3103.002 (micro average: 4097.787 +/- 0.000)
absolute_error: 1730.178 +/- 65.995 (micro average: 1730.177 +/- 3714.613)
relative_error: 295.48% +/- 87.95% (micro average: 295.48% +/- 5,147.81%)
squared_error: 16792932.533 +/- 56371164.627 (micro average: 16791861.534 +/- 2840308480.309)
correlation: 0.144 +/- 0.038 (micro average: 0.008)
squared_correlation: 0.022 +/- 0.008 (micro average: 0.000)
```

## 2- Cross validation for **Linear Regression**:

In Weka With 10 Folds

```
Classifier output

Relation:      Metro_Interstate_Traffic_Volume-weka.filters.unsupervised.attribute.Remove-R1,6-8
Instances:     48204
Attributes:    5
               temp
               rain_1h
               snow_1h
               clouds_all
               traffic_volume
Test mode:     10-fold cross-validation

=== Classifier model (full training set) ===

Linear Regression Model

traffic_volume =

    20.6425 * temp +
    4.1343 * clouds_all +
    -2749.0607

Time taken to build model: 0.04 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient      0.0076
Mean absolute error         1730.2904
Root mean squared error     4161.0686
Relative absolute error      99.1521 %
Root relative squared error  209.4266 %
Total Number of Instances   48204
```



## 2- Cross validation for **Linear Regression**:

In Weka With 20 Folds

```
Classifier output

Relation:      Metro_Interstate_Traffic_Volume-weka.filters.unsupervised.attribute.Remove-R1,6-8
Instances:     48204
Attributes:    5
               temp
               rain_1h
               snow_1h
               clouds_all
               traffic_volume
Test mode:     20-fold cross-validation

=== Classifier model (full training set) ===

Linear Regression Model

traffic_volume =

    20.6425 * temp +
    4.1343 * clouds_all +
    -2749.0607

Time taken to build model: 0.06 seconds

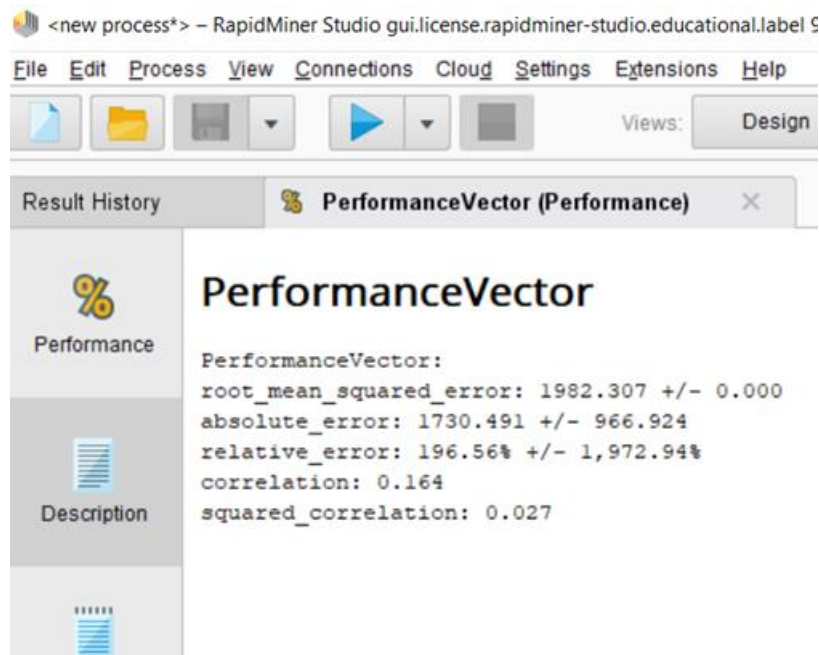
=== Cross-validation ===
=== Summary ===

Correlation coefficient          0.0076
Mean absolute error             1730.3893
Root mean squared error         4143.6945
Relative absolute error         99.1574 %
Root relative squared error     208.552 %
Total Number of Instances      48204
```

## • Multilayer Perceptron

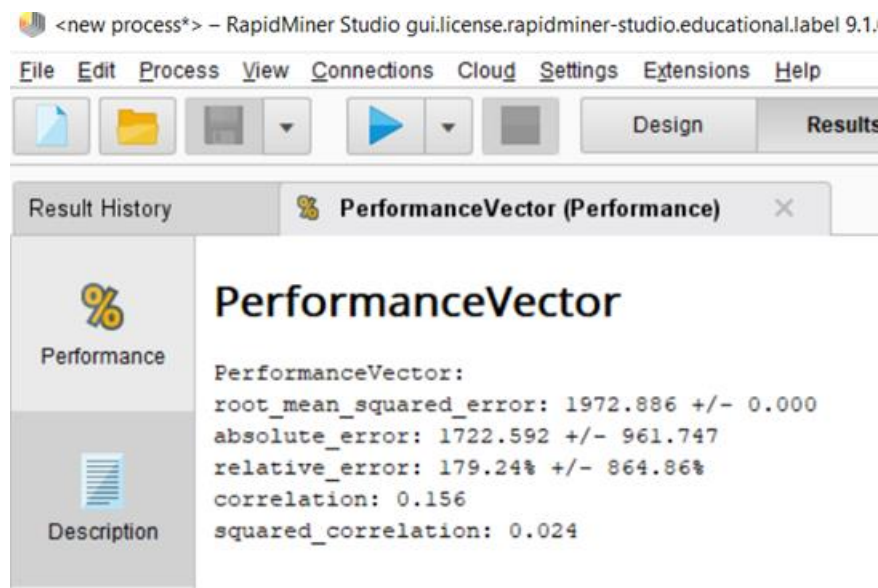
### 1- Split validation for **Multilayer Perceptron**:

In RapidMiner With 0.7 split



### 1- Split validation for **Multilayer Perceptron**:

In RapidMiner With 0.8 split



## 2- Split validation for **Multilayer Perceptron**:

In Weka With 0.7 split

```
Select attributes Visualize

.2 -N 500 -V 0 -S 0 -E 20 -H a

Classifier output

traffic_volume
Test mode: split 70.0% train, remainder test

=== Classifier model (full training set) ===

Linear Node 0
  Inputs  Weights
  Threshold  0.07949686244370083
  Node 1    -2.4872501441034323
  Node 2    -0.6195500043980948
Sigmoid Node 1
  Inputs  Weights
  Threshold  -5.64880418197142
  Attrib temp  -0.8722895208103282
  Attrib rain_lh  5.450529133924751
  Attrib snow_lh  5.473903002102157
  Attrib clouds_all  -13.351790725775903
Sigmoid Node 2
  Inputs  Weights
  Threshold  -6.33121224083205
  Attrib temp  -8.312460826540777
  Attrib rain_lh  6.276934718911213
  Attrib snow_lh  6.351653454677328
  Attrib clouds_all  -23.81242223784559
Class
  Input
  Node 0

Time taken to build model: 15.82 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0.15 seconds

=== Summary ===

Correlation coefficient          0.1415
Mean absolute error             1719.0456
Root mean squared error        1966.8953
Relative absolute error         98.7422 %
Root relative squared error     99.0386 %
Total Number of Instances      14461
```

## 2- Split validation for **Multilayer Perceptron**:

In Weka With 0.8 split

Select attributes | Visualize

1.2-N 500-V 0-S 0-E 20-H a

**Classifier output**

```
traffic_volume
Test mode: split 80.0% train, remainder test

=== Classifier model (full training set) ===

Linear Node 0
  Inputs  Weights
  Threshold  0.07949686244370083
  Node 1    -2.4872501441034323
  Node 2    -0.6195500043980948

Sigmoid Node 1
  Inputs  Weights
  Threshold  -5.64880418197142
  Attrib temp  -0.8722895208103282
  Attrib rain_lh  5.450529133924751
  Attrib snow_lh  5.473903002102157
  Attrib clouds_all  -13.351790725775903

Sigmoid Node 2
  Inputs  Weights
  Threshold  -6.33121224083205
  Attrib temp  -8.312460826540777
  Attrib rain_lh  6.276934718911213
  Attrib snow_lh  6.351653454677328
  Attrib clouds_all  -23.8124223784559

Class
  Input
  Node 0

Time taken to build model: 10.73 seconds

=== Evaluation on test split ===

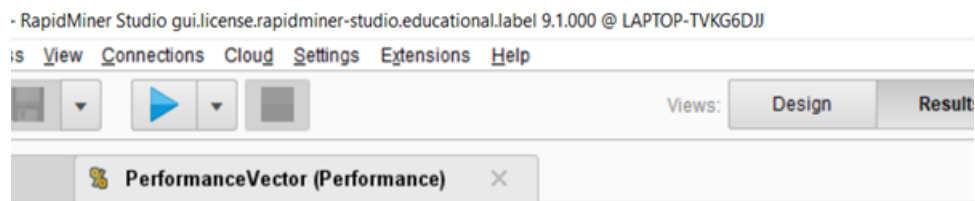
Time taken to test model on test split: 0.07 seconds

=== Summary ===

Correlation coefficient      0.129
Mean absolute error        1756.4698
Root mean squared error    2082.3326
Relative absolute error     101.1419 %
Root relative squared error 105.0484 %
Total Number of Instances  9641
```

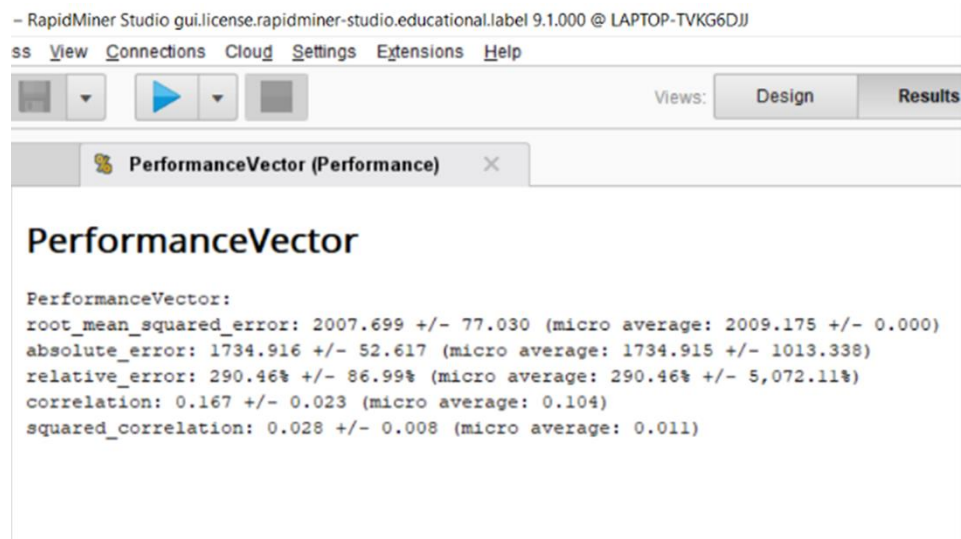
## 1- Cross validation for **Multilayer Perceptron**:

In RapidMiner With 10 Folds



## 1- Cross validation for **Multilayer Perceptron**:

In RapidMiner With 20 Folds



## 2-Cross validation for **Multilayer Perceptron**:

In Weka With 10 Folds

```
Select attributes Visualize
.2 -N 500 -V 0 -S 0 -E 20 -H a

Classifier output

rain_lh
snow_lh
clouds_all
traffic_volume
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Linear Node 0
Inputs  Weights
Threshold  0.07949686244370083
Node 1    -2.4872501441034323
Node 2    -0.6195500043980948
Sigmoid Node 1
Inputs  Weights
Threshold  -5.64880418197142
Attrib temp  -0.8722895208103282
Attrib rain_lh  5.450529133924751
Attrib snow_lh  5.473903002102157
Attrib clouds_all  -13.351790725775903
Sigmoid Node 2
Inputs  Weights
Threshold  -6.33121224083205
Attrib temp  -8.312460826540777
Attrib rain_lh  6.276934718911213
Attrib snow_lh  6.351653454677328
Attrib clouds_all  -23.81242223784559
Class
Input
Node 0

Time taken to build model: 14.74 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient      0.0464
Mean absolute error        1805.6282
Root mean squared error    2108.8495
Relative absolute error     103.4693 %
Root relative squared error 106.1384 %
Total Number of Instances  48204
```

## 2- Cross validation for **Multilayer Perceptron**:

In Weka With 20 Folds

```
ite | Select attributes | Visualize |
=====
W 0.2-N 500-V 0-S 0-E 20-H a

Classifier output

rain_lh
snow_lh
clouds_all
traffic_volume
Test mode: 20-fold cross-validation

=== Classifier model (full training set) ===

Linear Node 0
Inputs      Weights
Threshold   0.07949686244370083
Node 1      -2.4872501441034323
Node 2      -0.6195500043980948
Sigmoid Node 1
Inputs      Weights
Threshold   -5.64880418197142
Attrib temp  -0.8722895208103282
Attrib rain_lh  5.450529133924751
Attrib snow_lh  5.473903002102157
Attrib clouds_all -13.351790725775903
Sigmoid Node 2
Inputs      Weights
Threshold   -6.33121224083205
Attrib temp  -8.312460826540777
Attrib rain_lh  6.276934718911213
Attrib snow_lh  6.351653454677328
Attrib clouds_all -23.81242223784559
Class
Input
Node 0

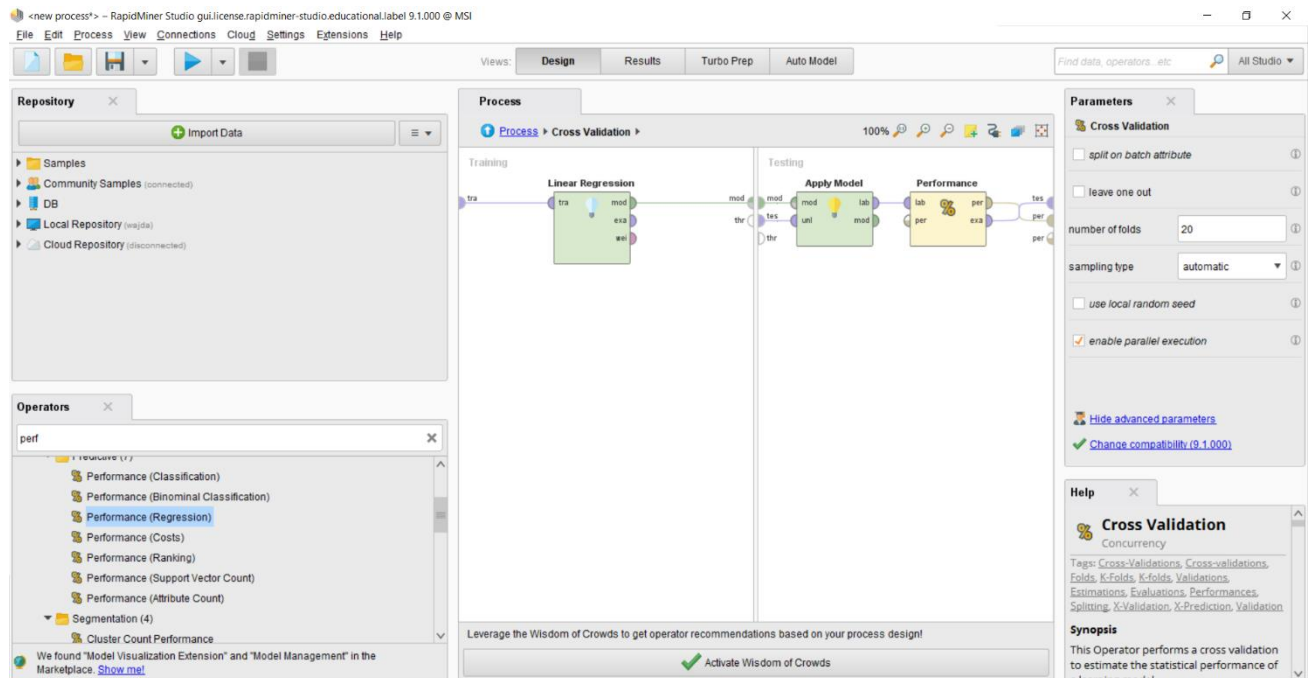
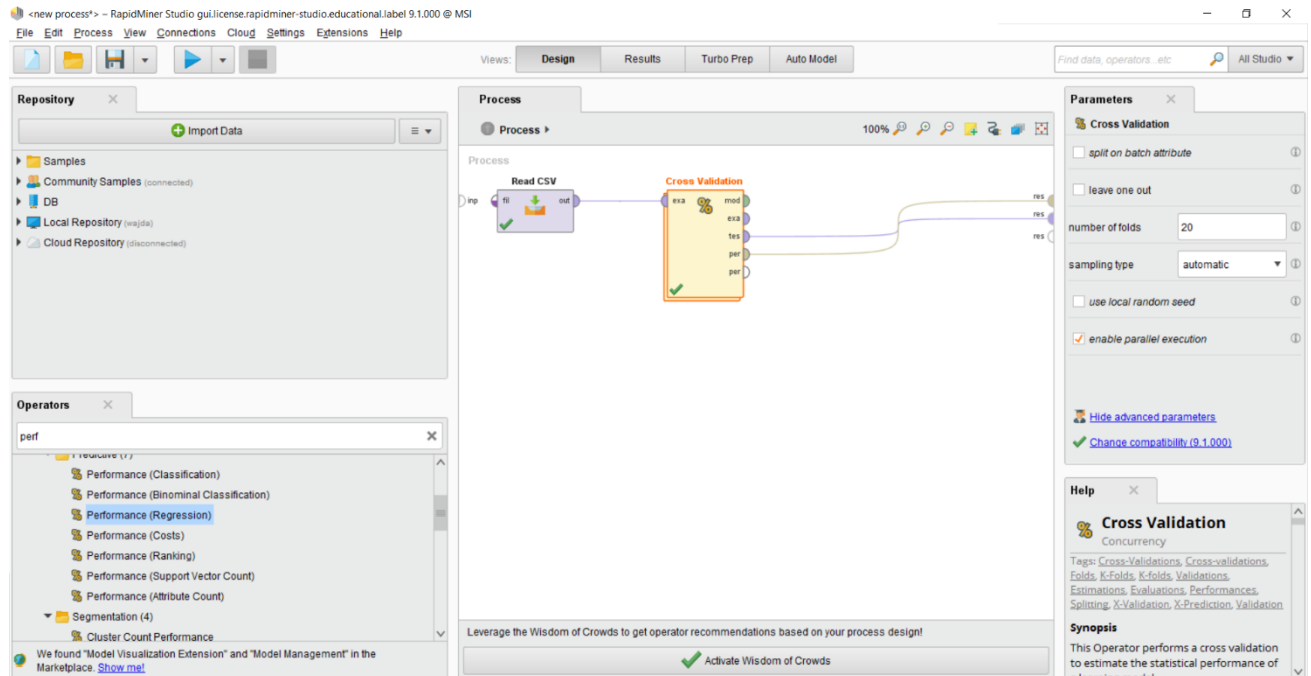
Time taken to build model: 15.66 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient      0.0369
Mean absolute error         1827.4707
Root mean squared error     2156.026
Relative absolute error      104.7205 %
Root relative squared error  108.5127 %
Total Number of Instances   48204
```

# • Designs and Steps

## Cross Validation RapidMiner Design with regression performance:





## Split validation RapidMiner Design with regression performance:

The screenshot shows the RapidMiner Studio interface with the following components:

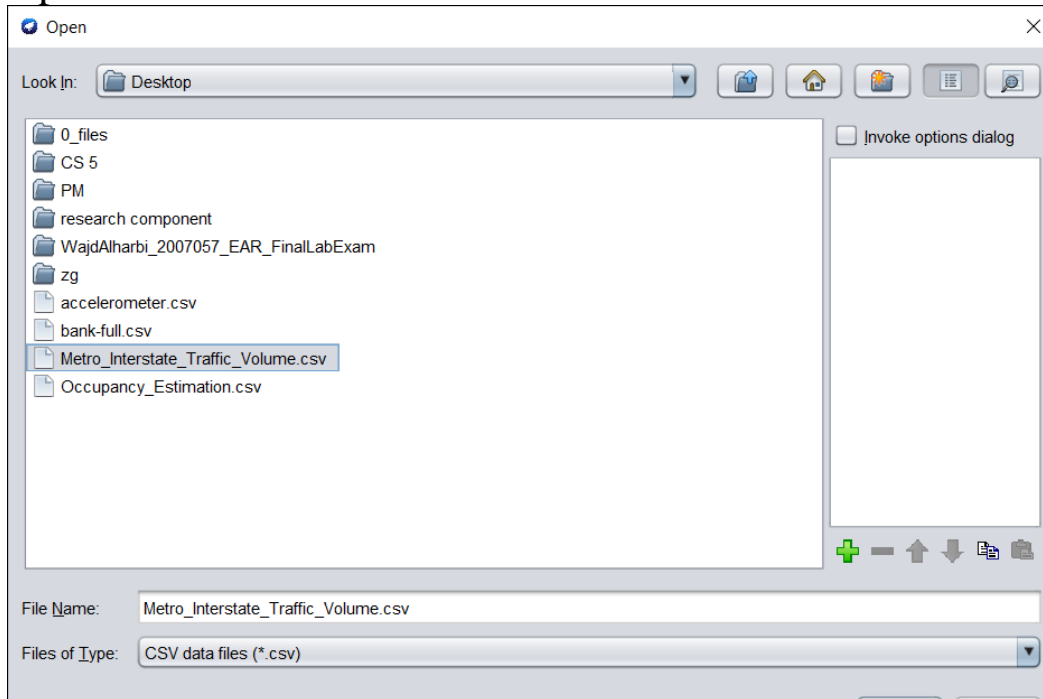
- Repository:** Contains 'Import Data' and a list of data sources: Samples, Community Samples (connected), DB, Local Repository (wajda), and Cloud Repository (disconnected).
- Operators:** A search bar with 'cross' and a list of operators including 'Create Association Rules', 'Similarities (2)', 'Data to Similarity', 'Cross Distances', and 'Validation (1)'. 'Cross Validation' is selected.
- Process:** A workflow diagram showing a 'Read CSV' operator connected to a 'Validation' operator. The 'Validation' operator has ports for 'tra' (training) and 'mod' (model), and 'ave' (average) and 'res' (results).
- Parameters:** A panel for 'Validation (Split Validation)' with settings: 'split' set to 'relative', 'split ratio' set to '0.8', and 'sampling type' set to 'automatic'. There is a checkbox for 'use local random seed' and links for 'Hide advanced parameters' and 'Change compatibility (9.1.000)'.
- Help:** A panel titled 'Split Validation' with a description and tags: 'Divide, Separate, Part, Training, Testing, Holdout, Partitions, Validations, Evaluations, Validation'.

The screenshot shows the RapidMiner Studio interface with the following components:

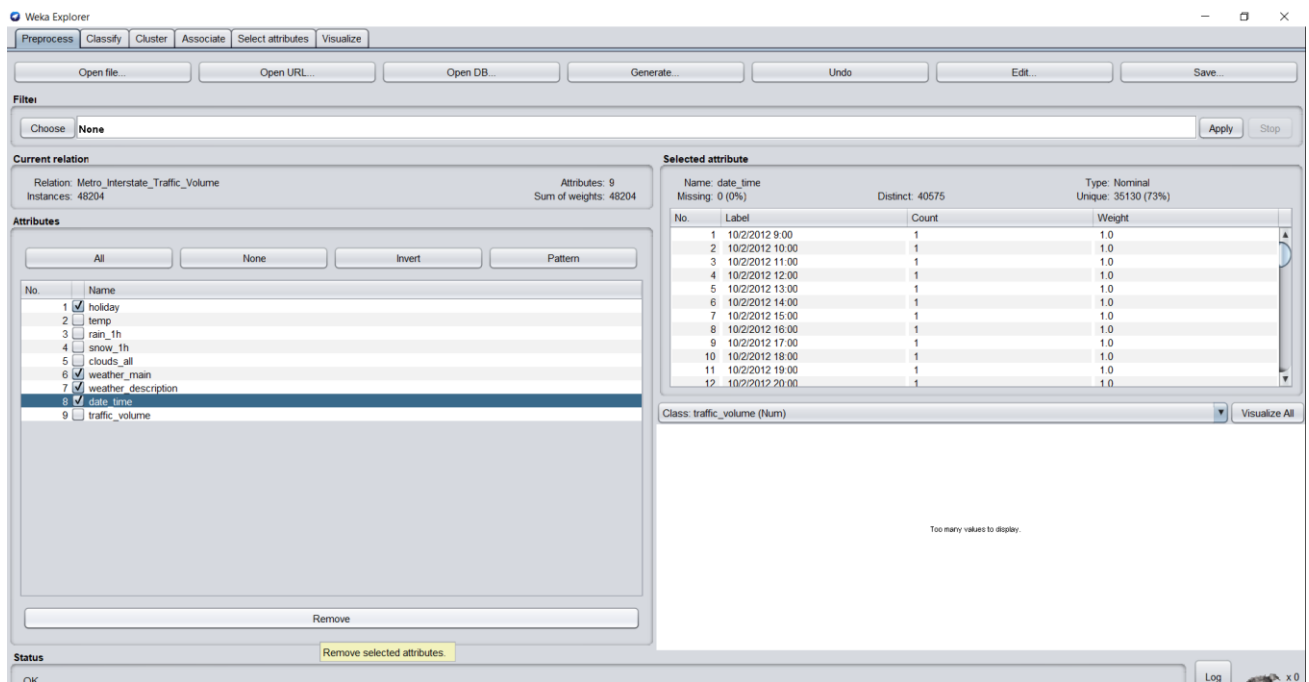
- Repository:** Same as the first screenshot.
- Operators:** Same as the first screenshot.
- Process:** A workflow diagram showing a 'Linear Regression' operator connected to an 'Apply Model' operator, which is then connected to a 'Performance' operator. The 'Linear Regression' operator has ports for 'tra' (training), 'mod' (model), and 'exa' (example). The 'Apply Model' operator has ports for 'mod' (model), 'lab' (label), and 'mod' (model). The 'Performance' operator has ports for 'per' (performance) and 'exa' (example).
- Parameters:** Same as the first screenshot.
- Help:** Same as the first screenshot.

## Weka Steps:

### Open file



### Remove Unwanted Attributes:



## Attributes are removed:

The screenshot shows the Weka Explorer interface with the 'Select attributes' tab active. The 'Current relation' panel displays 'Relation: Metro\_Interstate\_Traffic\_Volume-weka.filters.unsupervised.attribute.Remove-R1,6-8' and 'Instances: 46204'. The 'Attributes' list on the left shows five attributes: 'temp', 'rain\_1h', 'snow\_1h', 'clouds\_all', and 'traffic\_volume'. The 'Selected attribute' panel on the right shows details for 'temp', including its statistics (Minimum: 0, Maximum: 310.07, Mean: 281.206, StdDev: 13.338) and a histogram of its distribution. The 'Class' is set to 'traffic\_volume (num)'.

## Pick Algorithm (here I chose linear regression):

The screenshot shows the Weka Explorer interface with the 'Classify' tab active. The 'Classifiers' list on the left shows various algorithms, with 'LinearRegression' selected. A yellow tooltip box provides details about the 'LinearRegression' algorithm, including its description, capabilities, and attributes. The 'Class' is set to 'traffic\_volume (num)'. The 'build model' button is visible, and the 'Log' button is at the bottom right.

## Pick Label (I Chose traffic volume):

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4

Test options

☐ Use training set

☐ Supplied test set

☒ Cross-validation Folds 20

☐ Percentage split % 80

(Num) traffic\_volume

(Num) temp

(Num) rain\_1h

(Num) snow\_1h

(Num) clouds\_all

(Num) traffic\_volume

11:10:02 - functions.LinearRegression

11:10:22 - functions.LinearRegression

Classifier output

Relation: Metro\_Interstate\_Traffic\_Volume-weka.filters.unsupervised.attribute.Remove-R1,6-8

Instances: 48204

Attributes: 5

temp

rain\_1h

snow\_1h

clouds\_all

traffic\_volume

Test mode: 20-fold cross-validation

=== Classifier model (full training set) ===

Linear Regression Model

traffic\_volume =

20.6425 \* temp +

4.1343 \* clouds\_all +

-2745.0607

Time taken to build model: 0.06 seconds

=== Cross-validation ===

=== Summary ===

Correlation coefficient	0.0076
Mean absolute error	1730.3893
Root mean squared error	4143.6945
Relative absolute error	59.1574 %
Root relative squared error	206.552 %
Total Number of Instances	48204

Status

OK

Log