DATA SCIENCE

INFLUENZA SYMPTOM PREDICTION

Table of Contents

Influenza Symptom Prediction Analysis

1. Introduction

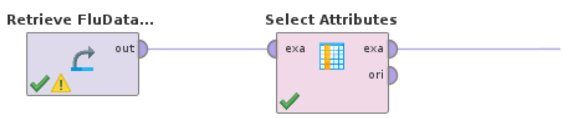
This assignment focuses on predicting the weekly number of patients with influenza symptoms in Germany using linear regression. The dataset, "FluData_basic_dataset.csv," encompasses crucial information about influenza cases, weather data, and Google Trends search data. The primary objective is to develop accurate regression models that utilize the available variables to forecast influenza symptom cases. The assignment begins with data preprocessing, including handling missing values, ensuring data quality, and transforming variables. Descriptive analytics will be conducted to analyze important variables, considering both single and multi-variable metrics. Subsequently, regression models will be built and evaluated using appropriate performance metrics. Additionally, an alternative regression model will be explored, followed by the enrichment of data using additional Google Trends search terms to improve the accuracy of the predictions.

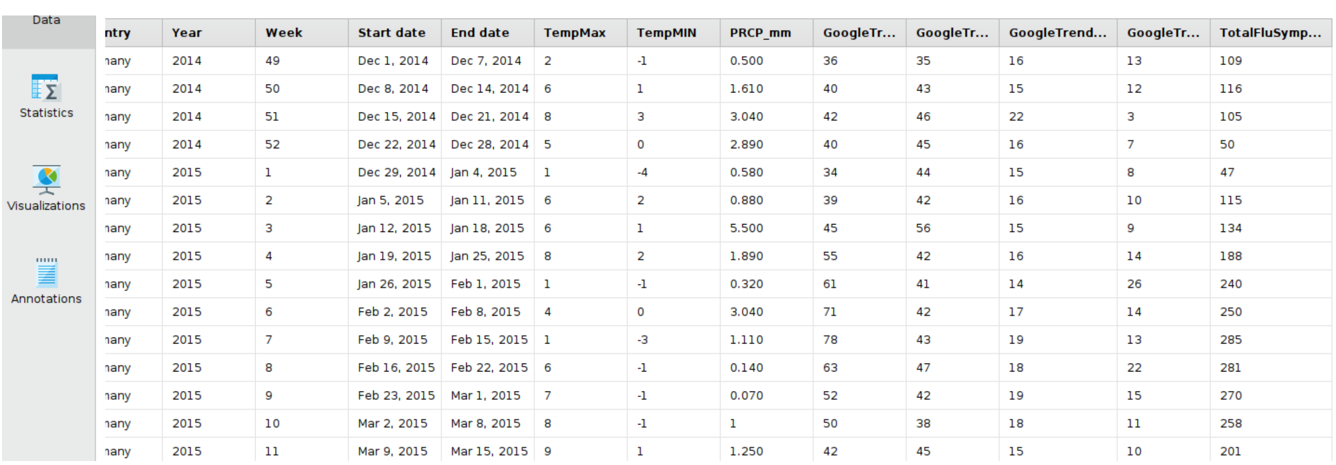
2. Description of Dataset

The dataset, "FluData_basic_dataset.csv," contains weekly data on influenza cases in Germany from December 1st, 2014 to February 2nd, 2020. It includes information on the number of patients with influenza symptoms reported by hospitals and doctors (TotalFluSympPatients) and the number of confirmed influenza cases (TotalFluPositiveVirusPatients). The dataset also incorporates weather data, including maximum and minimum temperatures (TempMax, TempMIN) and precipitation (PRCP_mm). Additionally, it includes Google Trends data on search volumes for specific terms related to influenza (GoogleTrends).

2.1. Retrieve Dataset

Import the dataset from your PC and add it into Rapid miner datasets section.





ntry	Year	Week	Start date	End date	TempMax	TempMIN	PRCP_mm	GoogleTr...	GoogleTr...	GoogleTrend...	GoogleTr...	TotalFluSymp...
ıany	2014	49	Dec 1, 2014	Dec 7, 2014	2	-1	0.500	36	35	16	13	109
ıany	2014	50	Dec 8, 2014	Dec 14, 2014	6	1	1.610	40	43	15	12	116
ıany	2014	51	Dec 15, 2014	Dec 21, 2014	8	3	3.040	42	46	22	3	105
ıany	2014	52	Dec 22, 2014	Dec 28, 2014	5	0	2.890	40	45	16	7	50
ıany	2015	1	Dec 29, 2014	Jan 4, 2015	1	-4	0.580	34	44	15	8	47
ıany	2015	2	Jan 5, 2015	Jan 11, 2015	6	2	0.880	39	42	16	10	115
ıany	2015	3	Jan 12, 2015	Jan 18, 2015	6	1	5.500	45	56	15	9	134
ıany	2015	4	Jan 19, 2015	Jan 25, 2015	8	2	1.890	55	42	16	14	188
ıany	2015	5	Jan 26, 2015	Feb 1, 2015	1	-1	0.320	61	41	14	26	240
ıany	2015	6	Feb 2, 2015	Feb 8, 2015	4	0	3.040	71	42	17	14	250
ıany	2015	7	Feb 9, 2015	Feb 15, 2015	1	-3	1.110	78	43	19	13	285
ıany	2015	8	Feb 16, 2015	Feb 22, 2015	6	-1	0.140	63	47	18	22	281
ıany	2015	9	Feb 23, 2015	Mar 1, 2015	7	-1	0.070	52	42	19	15	270
ıany	2015	10	Mar 2, 2015	Mar 8, 2015	8	-1	1	50	38	18	11	258
ıany	2015	11	Mar 9, 2015	Mar 15, 2015	9	1	1.250	42	45	15	10	201

3. Descriptive Analysis

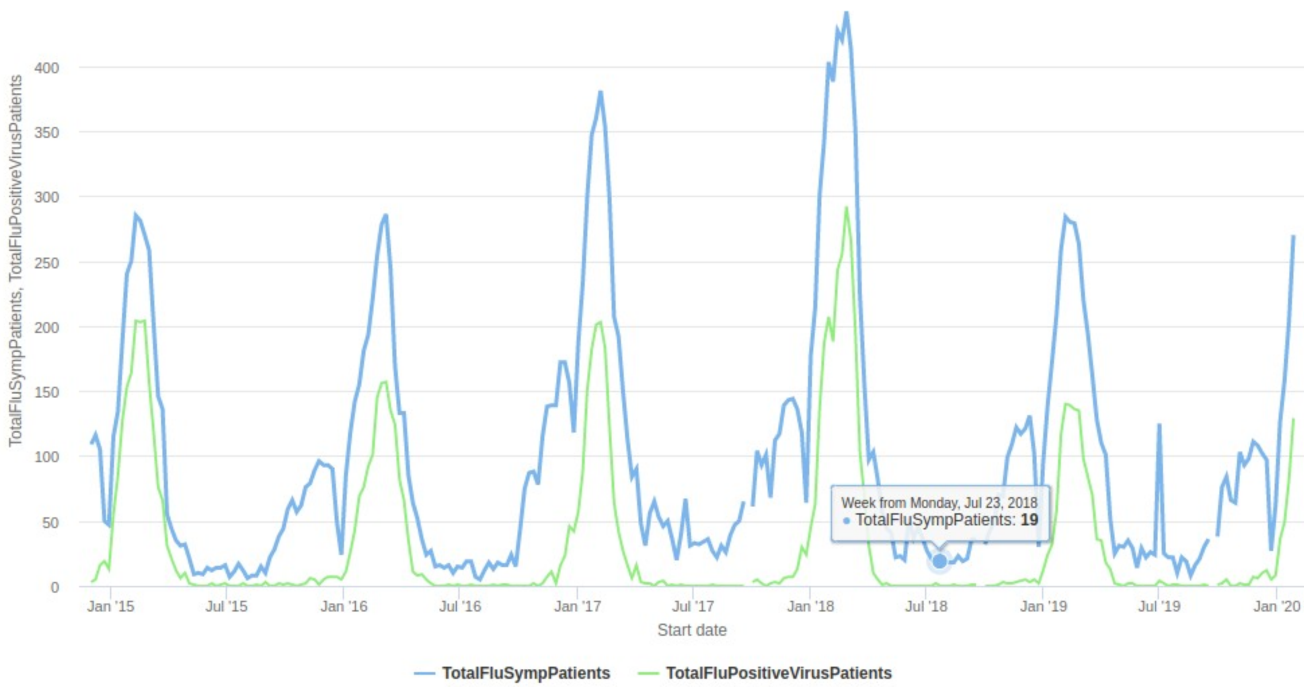
3.1. Statistics of datasets

The statistics of our dataset provide valuable insights into the characteristics of each variable. The mean represents the average value, indicating the typical or expected value for a given attribute. The median represents the middle value, which is useful for understanding the central tendency in the presence of outliers. The maximum and minimum values highlight the range of values observed in the dataset, indicating the upper and lower limits. Additionally, the average provides an overall summary of the dataset, capturing the collective tendencies of the variables. These statistics collectively offer a comprehensive overview of the dataset, aiding in data exploration and understanding its key attributes.

			Min	Max	Average
Year	Integer	0	2014	2020	2017.004
Week	Integer	0	1	53	26.621
TempMax	Integer	0	-3	32	14.898
TempMIN	Integer	0	-12	18	5.811
PRCP_mm	Real	0	0	16.460	1.560
GoogleTrendsFever	Integer	0	26	100	41.011
GoogleTrendsVomit	Integer	0	7	22	14.042
GoogleTrendsVaccine	Integer	0	3	30	12.330
TotalFluSympPatients	Real	0	5	442	98.390
TotalFluPositiveVirusPatients	Real	0	0	292	31.883
GoogleTrendsFatigue	Real	0	35	62	48.008

3.2. Data Visualization

In the graph, we can observe the total number of patients with a positive result for influenza and the total number of patients exhibiting symptoms of influenza. As we examine the data on a weekly basis, we can notice fluctuations in the number of influenza cases recorded, leading to an increase or decrease in the trends observed. These fluctuations are indicative of the varying prevalence of influenza over time, reflecting the dynamic nature of the dataset. By analyzing these trends, we can gain insights into the patterns and changes in influenza cases, allowing for a better understanding of the disease's occurrence and spread.



Upon observing the graph, it becomes evident that the majority of influenza cases are recorded during the month of January. This higher incidence can be attributed to the cold season in Germany. Cold weather conditions are known to contribute to the spread and prevalence of influenza, as people tend to spend more time indoors in close proximity to one another. The combination of cold temperatures and increased indoor interactions creates an environment conducive to the transmission of the influenza virus. Hence, the graph illustrates the seasonal pattern of influenza cases, with a notable concentration in the month of January due to the prevailing cold season in Germany.

4. Data Preprocessing

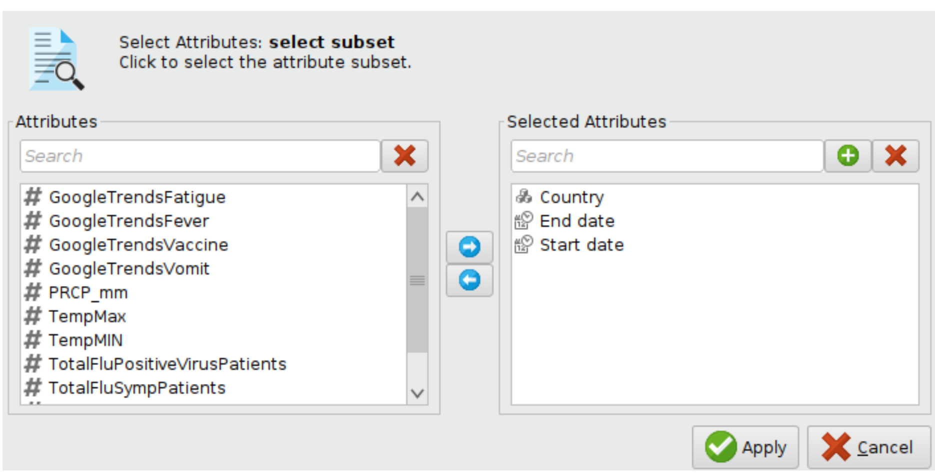
4.1. Data Cleaning

we understand that our dataset contains values representing the entire dataset for Germany, while each row corresponds to data for one week or six days as shown in below figure. In this case, there is no specific need for start date and end date columns as they may not provide crucial information for our analysis. Therefore, it is recommended to remove these unnecessary columns to streamline the dataset and focus on the relevant variables for predicting the weekly number of patients with influenza symptoms.

Country	Start date	End date
Germany	Dec 1, 2014	Dec 7, 2014
Germany	Dec 8, 2014	Dec 14, 2014
Germany	Dec 15, 2014	Dec 21, 2014
Germany	Dec 22, 2014	Dec 28, 2014
Germany	Dec 29, 2014	Jan 4, 2015
Germany	Jan 5, 2015	Jan 11, 2015
Germany	Jan 12, 2015	Jan 18, 2015
Germany	Jan 19, 2015	Jan 25, 2015
Germany	Jan 26, 2015	Feb 1, 2015
Germany	Feb 2, 2015	Feb 8, 2015
Germany	Feb 9, 2015	Feb 15, 2015
Germany	Feb 16, 2015	Feb 22, 2015
Germany	Feb 23, 2015	Mar 1, 2015

Select the Attributes

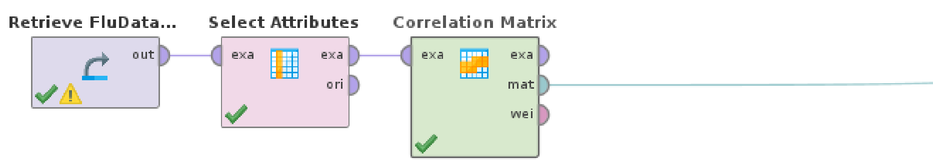
Now we select the those attributes or columns in rapid miner which is not necessary like country name, start date and end date. So exclude these columns from dataset for better result.



You can see in Figure after remove attributes.

Row No.	Year	Week	TempMax	TempMIN	PRCP_mm	GoogleTr...	GoogleTr...	GoogleTr...	GoogleTr...	TotalFlu...	TotalFlu...
1	2014	49	2	-1	0.500	36	35	16	13	109	3
2	2014	50	6	1	1.610	40	43	15	12	116	5
3	2014	51	8	3	3.040	42	46	22	3	105	16
4	2014	52	5	0	2.890	40	45	16	7	50	19
5	2015	1	1	-4	0.580	34	44	15	8	47	13
6	2015	2	6	2	0.880	39	42	16	10	115	55
7	2015	3	6	1	5.500	45	56	15	9	134	85
8	2015	4	8	2	1.890	55	42	16	14	188	127
9	2015	5	1	-1	0.320	61	41	14	26	240	153
10	2015	6	4	0	3.040	71	42	17	14	250	164
11	2015	7	1	-3	1.110	78	43	19	13	285	204
12	2015	8	6	-1	0.140	63	47	18	22	281	203
13	2015	9	7	-1	0.070	52	42	19	15	270	204
14	2015	10	8	-1	1	50	38	18	11	258	156

Correlation between the Attributes

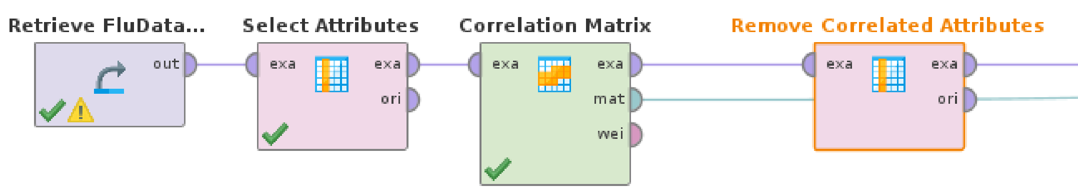


Next, we will examine the correlation between attributes in the dataset to identify which variables are highly correlated with each other. This analysis will help us understand the relationships between different attributes and how changes in one variable may impact others. By assessing the correlation matrix, we can identify the strongest positive or negative correlations.

Attributes	Year	Week	Temp...	Temp...	PRCP...	Goog...	Goog...	Goog...	Goog...	Total...	Total...
Year	1	-0.111	0.049	0.058	-0.032	0.236	0.343	-0.665	0.235	0.111	0.024
Week	-0.111	1	0.352	0.421	0.095	-0.467	0.133	0.094	-0.245	-0.475	-0.585
TempMax	0.049	0.352	1	0.947	0.079	-0.621	-0.126	-0.046	-0.021	-0.731	-0.590
TempMIN	0.058	0.421	0.947	1	0.194	-0.565	0.001	-0.058	-0.011	-0.684	-0.580
PRCP_mm	-0.032	0.095	0.079	0.194	1	-0.071	0.231	-0.040	-0.063	-0.089	-0.092
GoogleTrendsFever	0.236	-0.467	-0.621	-0.565	-0.071	1	0.021	-0.144	0.167	0.903	0.891
GoogleTrendsFatigue	0.343	0.133	-0.126	0.001	0.231	0.021	1	-0.177	-0.001	-0.003	-0.118
GoogleTrendsVomit	-0.665	0.094	-0.046	-0.058	-0.040	-0.144	-0.177	1	-0.176	-0.069	-0.006
GoogleTrendsVaccine	0.235	-0.245	-0.021	-0.011	-0.063	0.167	-0.001	-0.176	1	0.122	0.141
TotalFluSympPatients	0.111	-0.475	-0.731	-0.684	-0.089	0.903	-0.003	-0.069	0.122	1	0.926
TotalFluPositiveVirusP...	0.024	-0.585	-0.590	-0.580	-0.092	0.891	-0.118	-0.006	0.141	0.926	1

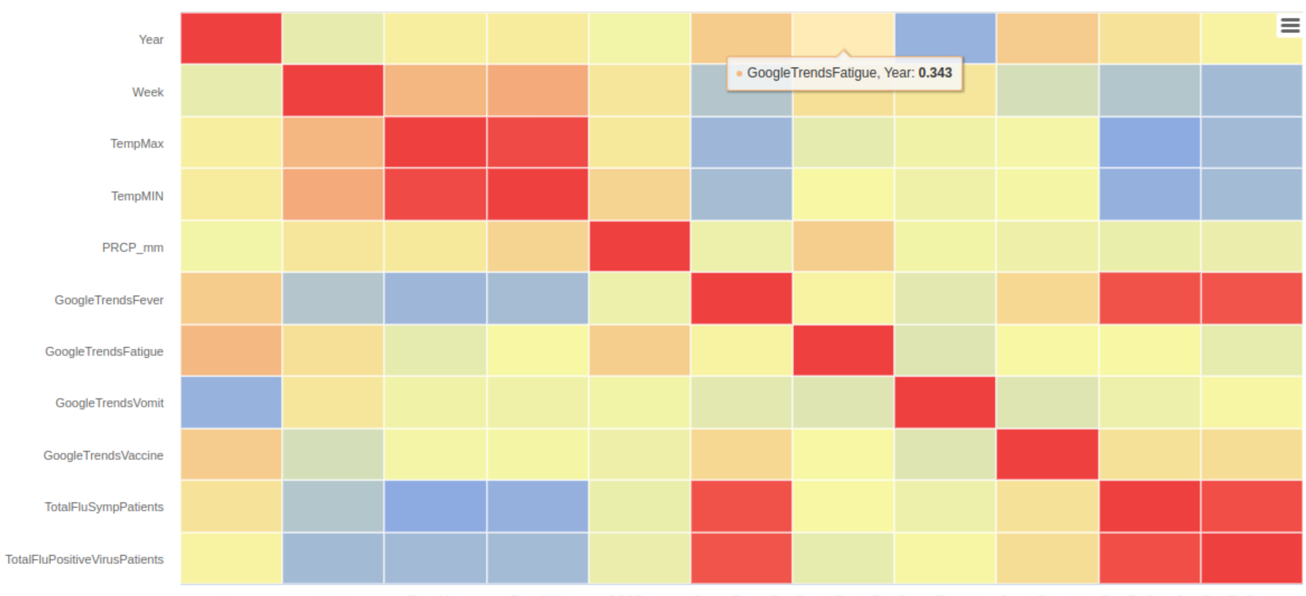
Remove Correlation Attributes

To achieve better results, we will remove attributes that have a correlation greater than 0.95 with other attributes. This can be done using the "Remove Correlated Attributes" operator. By eliminating highly correlated attributes, we can reduce redundancy in the dataset and potentially improve the performance of our analysis or modeling tasks.



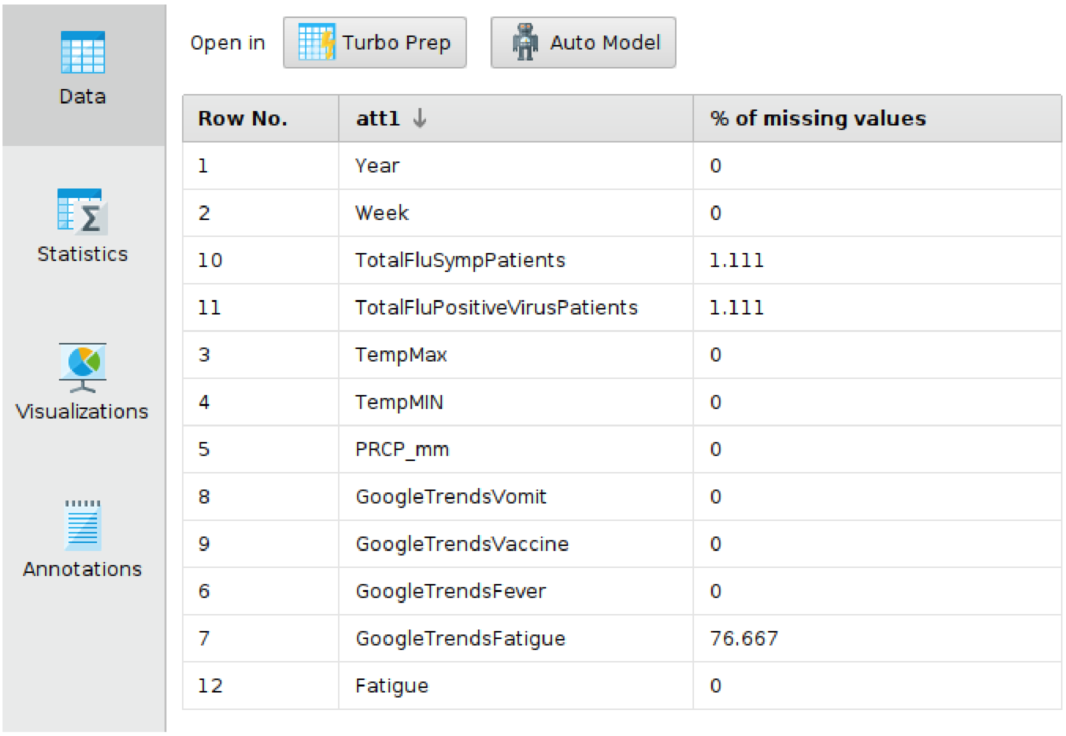
Now set the parameter for remove correlated attributes operator greater then 0.95.

After remove the highly correlated attributes.



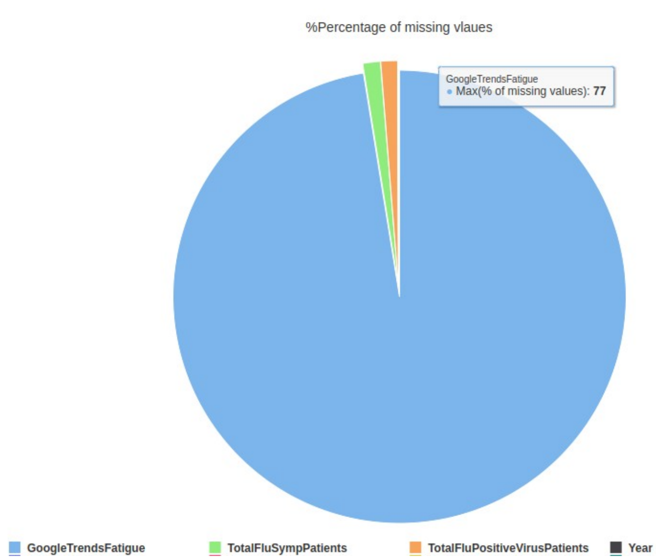
4.1. Handle the missing values

Now we handle missing values in our dataset and handling it with different techniques. With missing value we cannot train a Machine Learning model. As shown in table the percentage of missing values in out datasets. As you can see GoogleTrendsFatigue attribute have more 76% missing values.



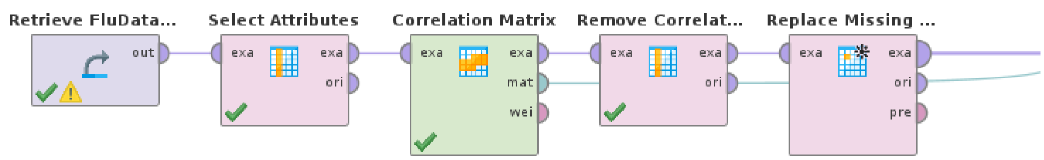
Row No.	att1 ↓	% of missing values
1	Year	0
2	Week	0
10	TotalFluSympPatients	1.111
11	TotalFluPositiveVirusPatients	1.111
3	TempMax	0
4	TempMIN	0
5	PRCP_mm	0
8	GoogleTrendsVomit	0
9	GoogleTrendsVaccine	0
6	GoogleTrendsFever	0
7	GoogleTrendsFatigue	76.667
12	Fatigue	0

GoogleTrendsFatigue exhibits substantial missing data (over 76%) based on the Pie chart. In contrast, other attributes have minimal missing values (1.11%), significantly lower than GoogleTrendsFatigue.

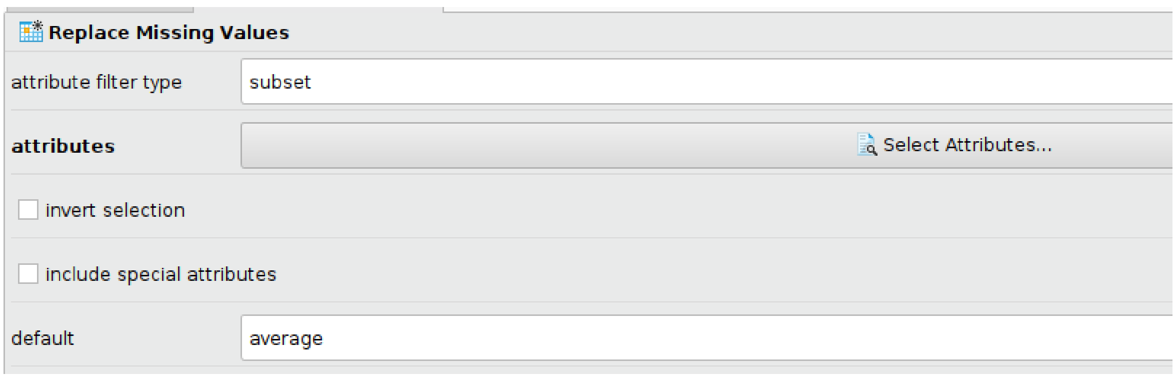


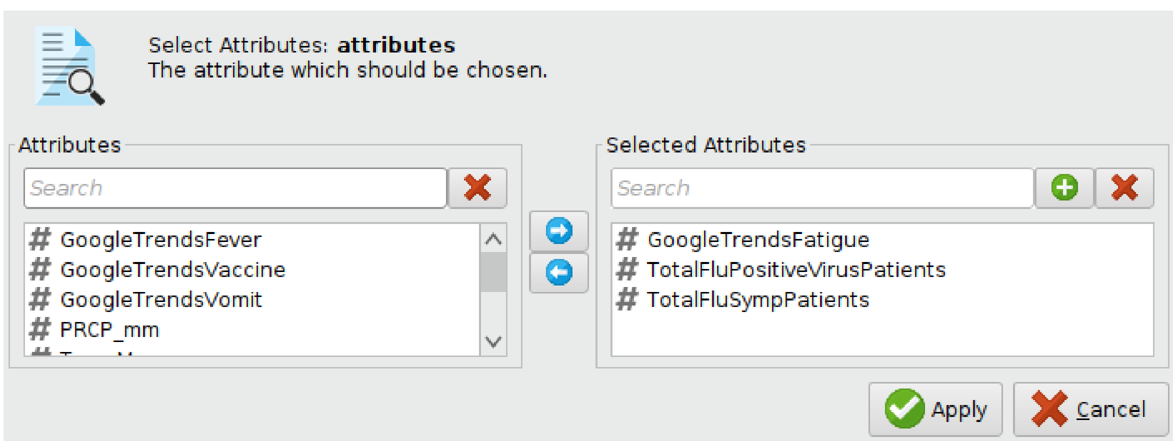
Replace Missing Values

So, for filling the missing values in our datasets, we will use different techniques. The columns TotalFluSympPatients and TotalFluPositiveVirusPateints have a 1.11% missing value rate, and GoogleTrendsFatigue has a substantial 77% missing value rate. To address this, filling the missing values with mean or median is a suitable approach. By applying mean or median, we can observe the changes in the dataset after filling the missing values and evaluate their impact."



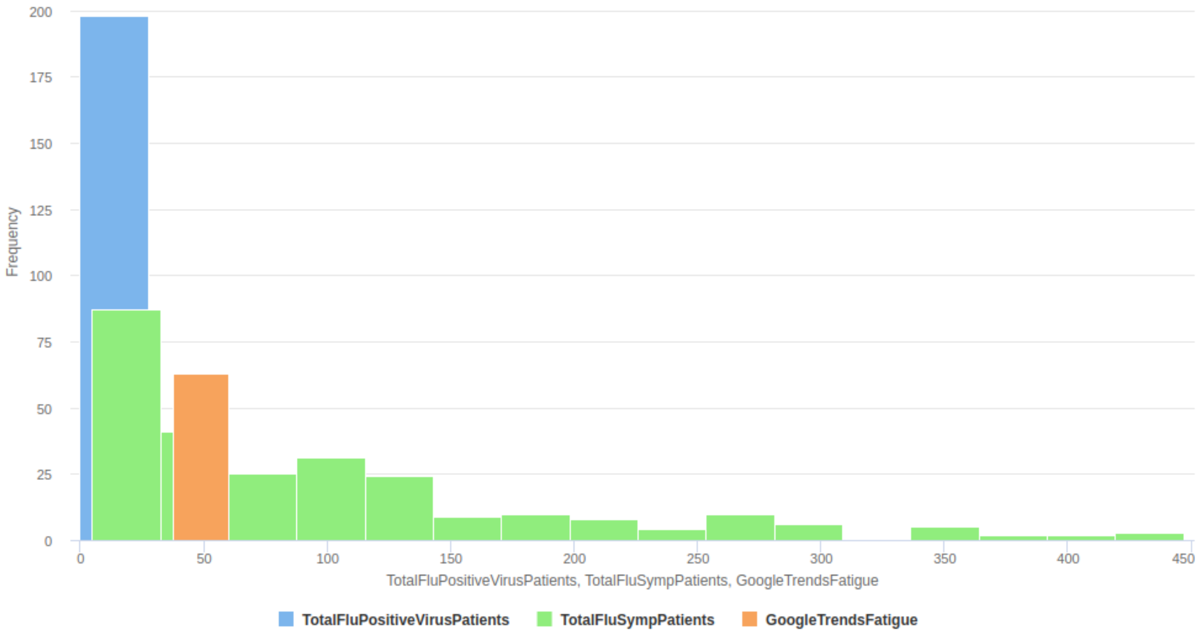
For replace the missing value we use replace missing value operator in rapid miner and update his parameter, set the attribute filter type, add attribute name and run the operator.



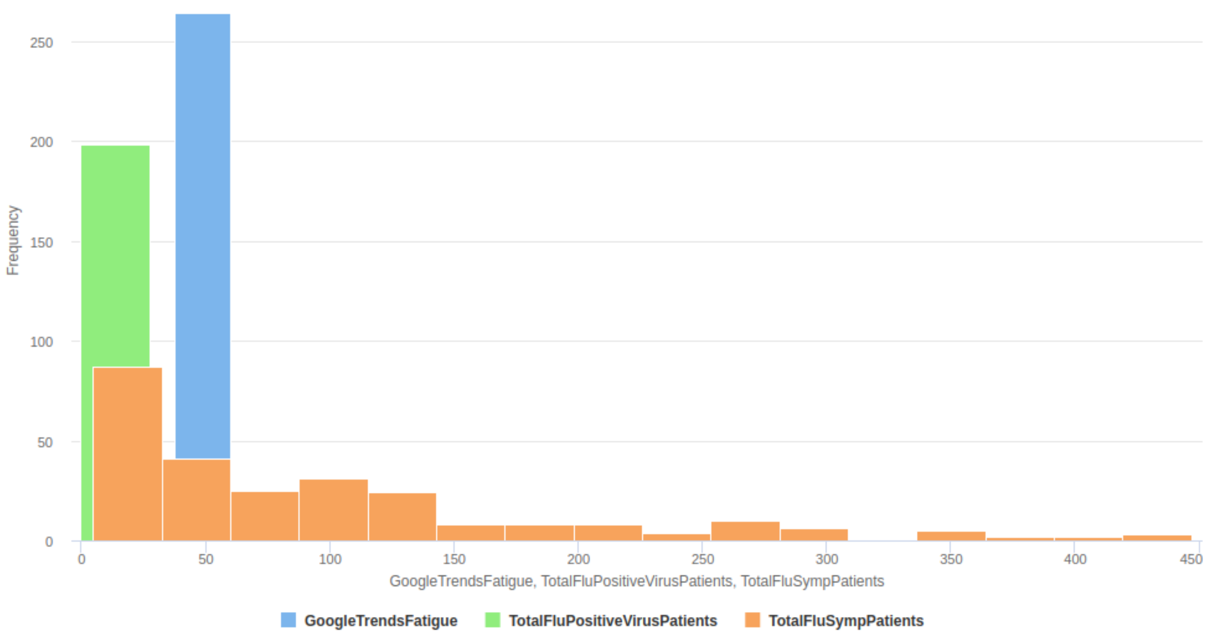


Overview of dataset before and after replace missing values

Here is the histogram depicting the missing attributes in the dataset prior to replacing the missing values. The graph highlights that the column "GoogleTrendsFatigue" has only 30% of the data available, indicating a significant amount of missing values within that column. This information is valuable for understanding the completeness of the dataset and determining the necessary steps to address the missing values, such as imputation techniques or considering the impact of missing data on subsequent analyses.



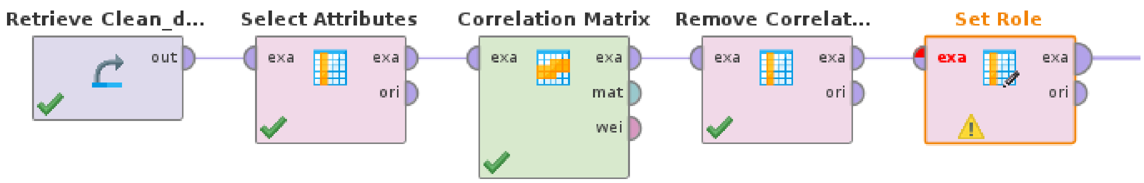
Before Replace missing values



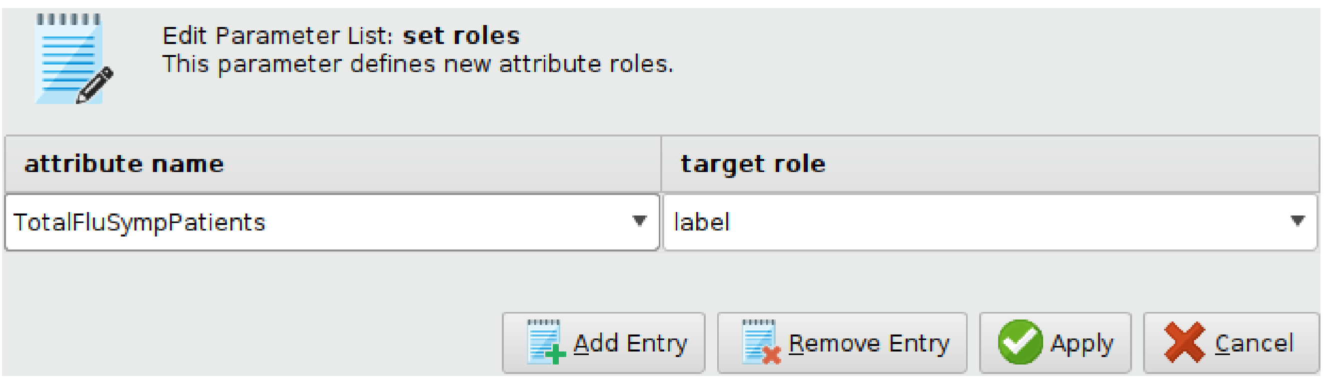
After Replace missing values

4.2. Set Target Values

To set the target variable for our predictive model in Rapid Miner, we will utilize the "Operator Set Role." This operator allows us to assign the role of the target variable, which is the variable we aim to predict. By setting the target variable appropriately, we can focus our modeling efforts on accurately predicting its values.



In our case we will set target variable is Weekly Flu Symptom in Patients.

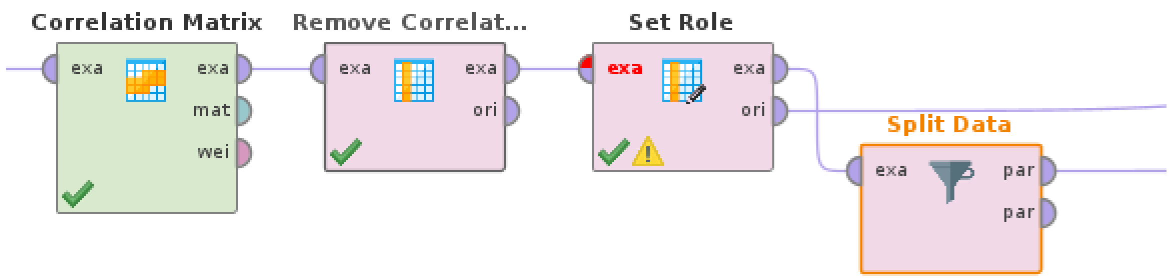


Model is now configured to predict the values of the designated target variable based on the input data and the defined model structure.

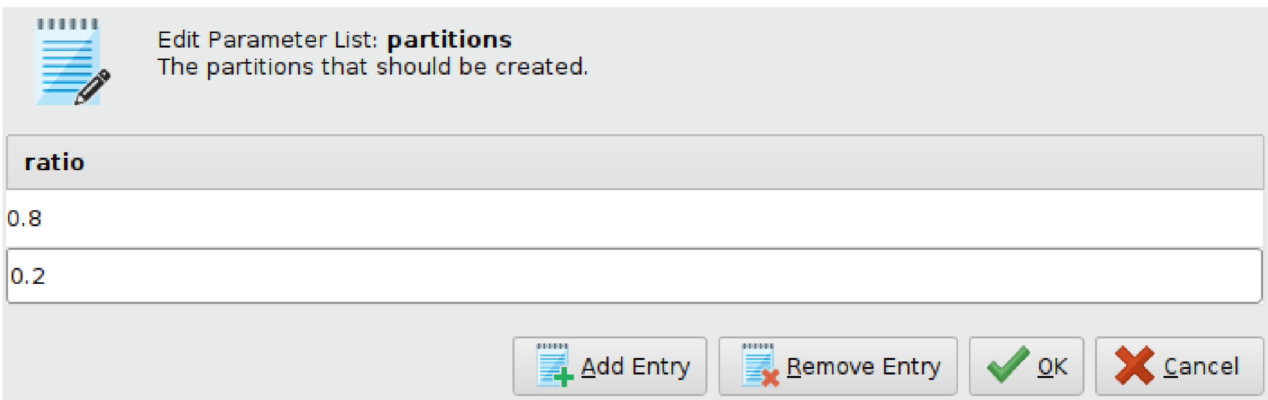
GoogleTrendsVaccine	GoogleTrendsFatigue	TotalFluPositiveVirusPatients	TotalFluSympPatients
13	35	3	109
12	43	5	116
3	46	16	105
7	45	19	50
8	44	13	47
10	42	55	115
9	56	85	134
14	42	127	188
26	41	153	240

4.3. Split the Dataset

To perform the dataset split for testing and training purposes in Rapid Miner, we will utilize the "Split Data" operator as shown in figure. This operator allows us to divide the dataset into two separate subsets: one for testing and the other for training. The dataset will be split into 20% for testing and 80% for training, ensuring that an adequate portion of the data is reserved for evaluating the model's performance on unseen instances. This division enables us to train the model on the majority of the data while assessing its effectiveness on the withheld testing data.



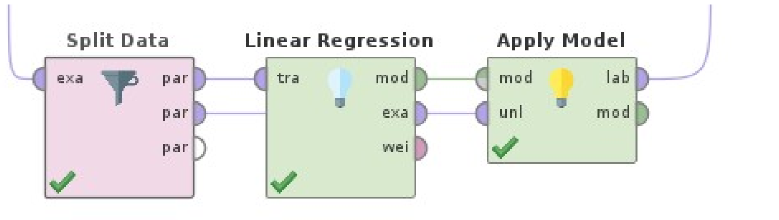
Set Partitions Ratio.



5. Apply Predictive Model

5.1. Predictive Values

We proceed with the application of a linear regression model on our training and testing datasets. Linear regression is a statistical modeling technique that aims to establish a linear relationship between the independent variables and the dependent variable. By fitting the model to the training dataset, we can estimate the coefficients and intercept that best represent this relationship.



After applying the machine learning model to our dataset, we obtain the predicted values for the target variable. The first column represents the actual values of the target variable, while the second column contains the predicted values generated by the model.

TotalFluSympPatients	prediction(TotalFluSympPatients)
79	61.246
49	93.006
85	83.030
133	138.527
36	57.593
14	11.877
19	29.639
18	34.716
16	27.668
44	34.616
78	80.174
139	87.758
187	151.522
233	192.665

These coefficients represent the weight assigned to each variable in the linear regression model for predicting the target variable. The constant term (-11237.735) is the intercept or the baseline prediction.

Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value
Year	5.605	1.437	0.086	1.000	3.901	0.000
Week	0.512	0.160	0.080	0.726	3.208	0.002
TempMax	-4.418	0.726	-0.389	0.464	-6.088	0.000
TempMIN	2.216	1.051	0.131	0.464	2.108	0.036
GoogleTrendsFever	0.771	0.413	0.099	0.147	1.869	0.063
TotalFluPositiveVirus...	1.183	0.088	0.729	0.192	13.495	0
(Intercept)	-11237.735	2891.400	?	?	-3.887	0.000

5.2. Error Calculation

This is the performance vector of our model.

Metric	Value	Standard Deviation
root_mean_squared_error	20.686	0.000
absolute_error	15.803	13.349
relative_error	31.16%	34.02%
root_relative_squared_error	0.219	-
squared_error	427.931	631.801

- In this case, the value is 20.686, indicating the average root mean squared error of the model's predictions. A lower value indicates better accuracy.
- The absolute error represents the absolute difference between the actual and predicted values. The value of 15.803 implies the average absolute error of the model's predictions. A lower value indicates better accuracy.
- The relative error measures the percentage difference between the actual and predicted values. The value of 31.16% suggests that, on average, the predictions have a relative error of 31.16%. A lower value indicates better accuracy.
- This metric represents the square root of the relative squared error. The value of 0.219 indicates the average root relative squared error of the model's predictions. A lower value indicates better accuracy.
- The squared error measures the squared difference between the actual and predicted values. The value of 427.931 represents the average squared error of the model's predictions. A lower value indicates better accuracy.

6. Apply Predictive Model includes new trends

Now, we have updated our dataset by incorporating additional features from Google Trends(2014-2020) that are related to influenza, as depicted in the dataset provided below.

GoogleTrendsChickenpox	GoogleTrendsRubella	GoogleTrendsSmallpoxVaccine	GoogleTrendsInfertility	GoogleTrendsCampylobacter	GoogleTrendsTyphoidFever
40	72	0	29	57	28
37	76	0	36	58	29
40	61	1	27	81	24
41	72	1	31	47	28
45	59	1	34	68	25
40	53	0	35	83	26
40	65	1	31	73	23
40	68	0	39	54	16
39	55	0	44	54	26
37	50	0	35	87	31
37	51	1	31	62	34
40	54	1	34	78	34
34	52	1	25	59	21
30	59	1	35	45	24
40	62	1	38	34	26

Attributes	Google...	Goog...	Google...	Goog...	Goog...	Goog...	Goog...	Goog...	Goog...	Goog...
GoogleTrendsFever	1	0.021	-0.123	-0.255	-0.057	0.015	0.282	-0.077	-0.144	0.167
GoogleTrendsFatigue	0.021	1	0.087	-0.100	-0.033	0.074	0.156	-0.023	-0.177	-0.001
GoogleTrendsChickenpox	-0.123	0.087	1	0.544	0.434	-0.432	0.123	0.178	-0.067	0.076
GoogleTrendsRubella	-0.255	-0.100	0.544	1	-0.057	-0.183	0.081	0.105	0.078	-0.026
GoogleTrendsSmallpoxVacci...	-0.057	-0.033	0.434	-0.057	1	0.031	-0.167	-0.052	-0.128	-0.058
GoogleTrendsInfertility	0.015	0.074	-0.432	-0.183	0.031	1	-0.087	-0.250	-0.047	-0.106
GoogleTrendsCampylobacter	0.282	0.156	0.123	0.081	-0.167	-0.087	1	-0.042	0.201	-0.099
GoogleTrendsTyphoidFever	-0.077	-0.023	0.178	0.105	-0.052	-0.250	-0.042	1	-0.142	0.105
GoogleTrendsVomit	-0.144	-0.177	-0.067	0.078	-0.128	-0.047	0.201	-0.142	1	-0.176
GoogleTrendsVaccine	0.167	-0.001	0.076	-0.026	-0.058	-0.106	-0.099	0.105	-0.176	1

6.1. Predictive Values

These coefficients represent the weights assigned to each variable in the linear regression model for predicting the target variable. The constant term (-9334.154) is the intercept or the baseline prediction.

Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value
Year	4.656	2.049	0.067	0.983	2.272	0.024
Week	0.717	0.134	0.107	0.755	5.359	0.000
TempMax	-4.251	0.722	-0.360	0.640	-5.890	0.000
TempMIN	1.976	1.034	0.113	0.660	1.911	0.057
PRCP_mm	-0.945	0.839	-0.019	0.993	-1.127	0.261
GoogleTrendsFever	0.206	0.244	0.027	0.289	0.845	0.399
GoogleTrendsFatigue	0.619	0.362	0.042	0.994	1.709	0.089
GoogleTrendsChicke...	-0.219	0.269	-0.028	1.000	-0.813	0.417
GoogleTrendsSmallp...	0.285	0.268	0.048	0.996	1.063	0.289
GoogleTrendsInfertility	0.277	0.146	0.044	0.991	1.895	0.059
GoogleTrendsCampyl...	0.202	0.136	0.036	0.938	1.486	0.139
GoogleTrendsTyphoi...	-0.631	0.182	-0.097	1.000	-3.458	0.001
GoogleTrendsVomit	-1.039	0.716	-0.030	0.989	-1.451	0.148
GoogleTrendsVaccine	0.532	0.545	0.017	0.982	0.975	0.331
TotalFluPositiveVirus...	1.297	0.041	0.834	0.386	31.388	0
(Intercept)	-9334.154	4123.794	?	?	-2.263	0.025

6.2. Error Calculation

These metrics provide insights into the performance of the regression model. Lower values for root_mean_squared_error, absolute_error, relative_error, root_relative_squared_error, and squared_error, along with a narrower range, indicate better model performance. Additionally, a smaller range for prediction_average suggests greater consistency in the predicted values.

Metric	Value	Standard Deviation
root_mean_squared_error	27.589	0.000
absolute_error	18.525	20.444
relative_error	36.74%	52.84%
root_relative_squared_error	0.289	
squared_error	761.137	1899.517
prediction_average	97.944	95.547

6.3. Comparison between Model1 and Model2

Performance Metric	PerformanceVector1	PerformanceVector2
root_mean_squared_error	20.686 +/- 0.000	27.589 +/- 0.000
absolute_error	15.803 +/- 13.349	18.525 +/- 20.444
relative_error	31.16% +/- 34.02%	36.74% +/- 52.84%
root_relative_squared_error	0.219	0.289
squared_error	427.931 +/- 631.801	761.137 +/- 1899.517
prediction_average	97.944 +/- 95.547	N/A

To determine which performance is better, it depends on the specific evaluation metric or criteria that is most important for your task.

- A lower RMSE value indicates better model performance. Predictive Model1 has a lower RMSE (20.686) compared to Predictive Model2 (27.589), suggesting better accuracy in Predictive Model1.
- Predictive Model1 has a lower average absolute error (15.803) compared to Predictive Model2 (18.525), indicating better accuracy in Predictive Model1.
- Predictive Model1 has a lower average relative error (31.16%) compared to Predictive Model2 (36.74%), suggesting better accuracy in Predictive Model1.
- Predictive Model1 has a lower root relative squared error (0.219) compared to Predictive Model2 (0.289), indicating better accuracy in Predictive Model1.
- Predictive Model1 has a lower average squared error (427.931) compared to Predictive Model2 (761.137), suggesting better accuracy in Predictive Model1.

Based on these metrics, Predictive Model1 generally demonstrates better model performance compared to Predictive Model2.

7. Overview of all processes