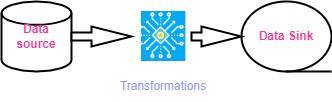
Data transformation: BigQuery

**Cloud Dataprep** is considered as a cloud data service to explore data visually, as well as prepare data for analysis.

**When to use:**

Major utility of Cloud Dataprep will be when you need to clean, explore, and prepare structured or unstructured data.

Dataflow : Dataflow is a process which contains the processes as read data from the source, transform and write it back into a sink.



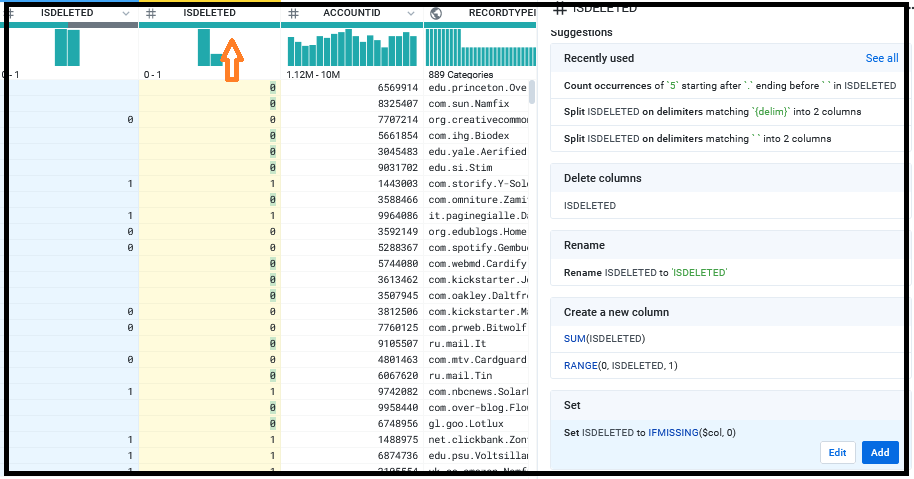
We can create dataflow jobs using the cloud console UI, gCloud CLI or APIs.

If we click “isdeleted” column from sales table, a new suggestion panel opens to the right of the Transformer page. This panel shows a series of suggestions relevant to the type of data contained in the specific column.

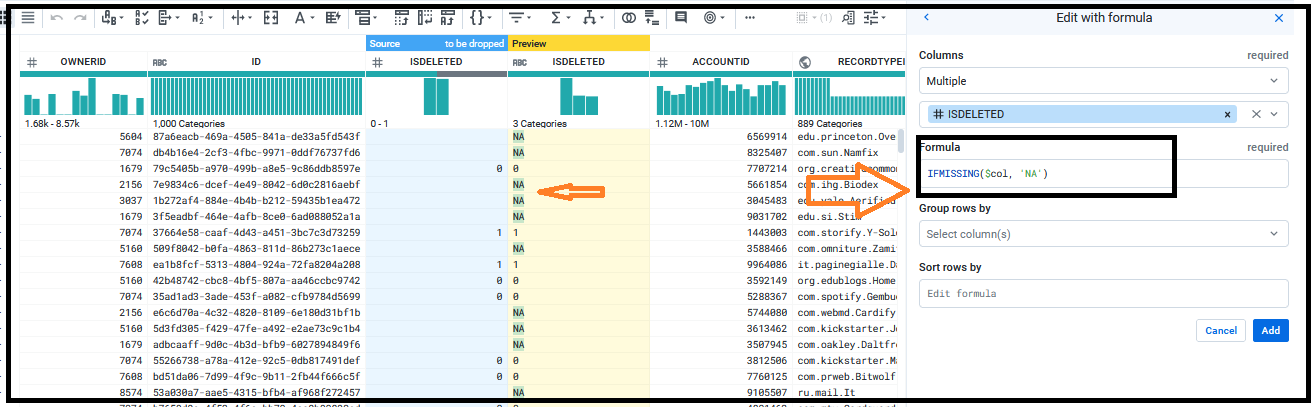
Several suggestions are proposed:

* Delete columns
* Rename
* Aggregate and group data
* Create a new column
* Set
* Values to columns

**Handle missing value**

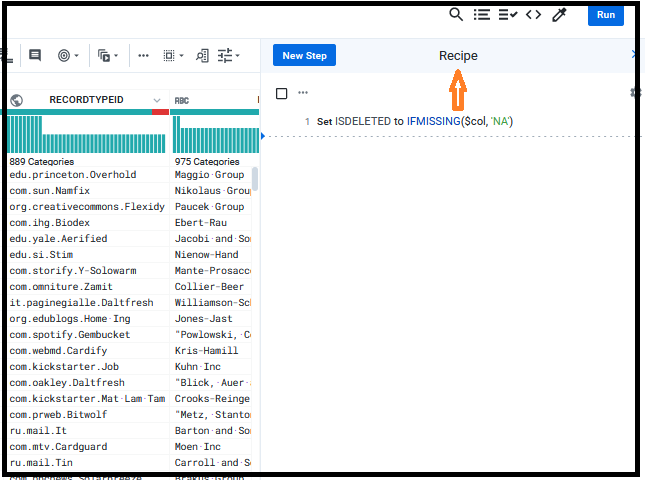


By clicking on the Set item, two buttons appear: Edit and Add. By clicking on the Edit button a new box is opened; a new formula is proposed in this box as follows:



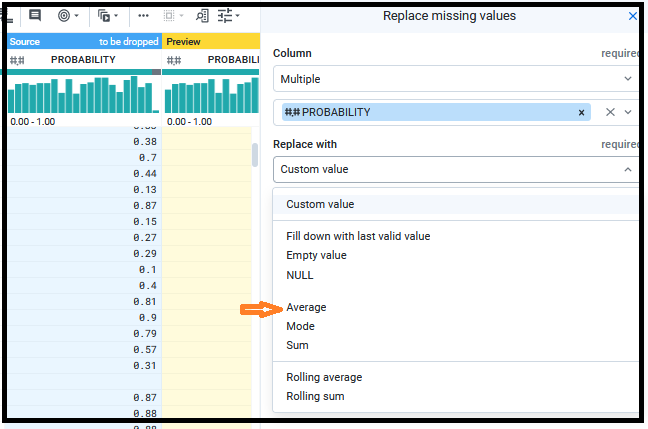
Now, to apply this suggestion, just click on the Add button in the suggestions panel. In this way, a new step will be added to the Recipe panel.

Recipe panel allows us to review and modify the steps of the recipe we have created so far.



**Replacing empty values with avg. value**

Column “Probability” has some empty rows. We can replace those values with average value.



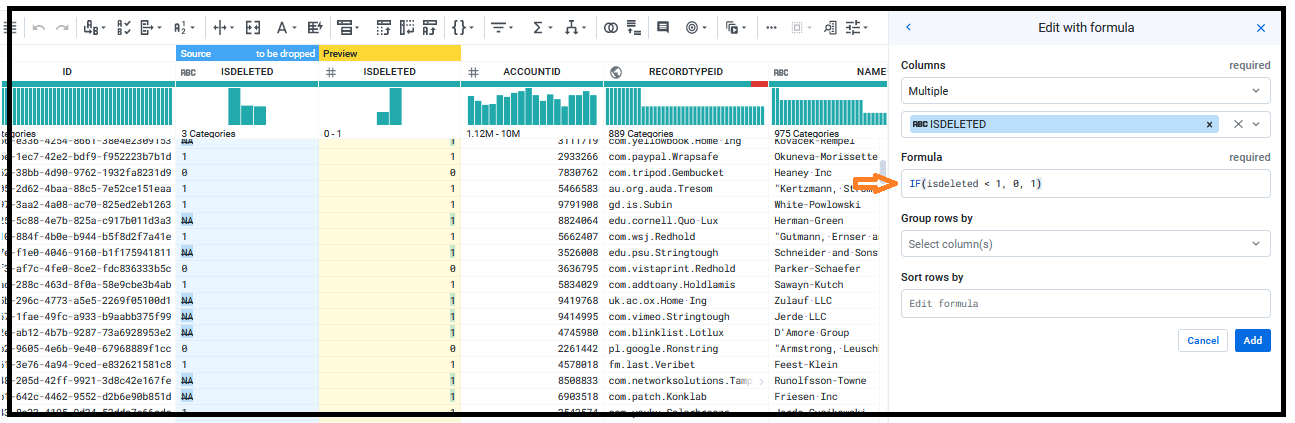
**Apply IF THEN statement**

Click on the respective column where do you want to make changes.

“isdeleted” column has some empty values. Apply the formula…

IF(isdeleted < 1, 0, 1)

The result shows:



We can replace this value with the missing value indicator, NA. To do this, we will write the following formula in the set suggestions:

IF(isdeleted < 1, 'NA', isdeleted)

# Mismatched values

A mismatched value is any value that seems to be different data type than the type specified for the column. In the data quality bar, mismatched values are identified in red. To fix mismatched data, there are several options available:

* Change the data type
  + Under change type, we have options as
    - String
    - Integer
    - Boolean
    - Decimal
    - Date/time
    - array
* Replace the values with constant values
* Set the values with other columns' values
* Transform the data with functions
* Delete rows
* Hide the column for now
* Drop the column

# Finding outliers in the data

Outliers are the values that, compared to others, are particularly extreme (a value clearly distant from the other available observations). The presence of outliers causes a hindrance because they tend to distort the results of data analysis**.** Unusual values or patterns in the data can be sources for the following:

* Missing data.
* Bad data.
* Poorly formatted data
* Mismeasured data
* Data that skews statistics

# This section provides guidance in how to locate these patterns of data in individual columns.

**Single column outliers**

To identify outliers in individual columns, Google Cloud Dataprep has

* visual functionality
* statistical information.

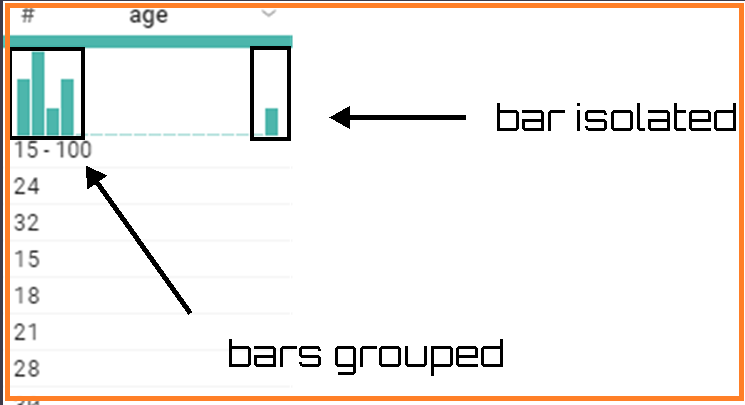
**Visual functionality**

Histograms displays on top of each column. This graph displays the count of each value detected in the column (for string data) or the count of values ​​within a numeric range (for numerical data).

The visual functionality offered by Google Cloud Dataprep refers precisely to the use of these histograms to identify unusual values ​​or outliers, which should be removed or corrected before performing any analysis on the entire dataset.

The type of histogram returned depends on the type of data contained in the column.

* Numeric data
  + for numeric data, each bar refers to a range of values ​​and the bars are sorted in numerical order.
* Categorical data
  + For categorical types, each vertical bar covers a single value, ordered by the values ​​that occur most frequently.



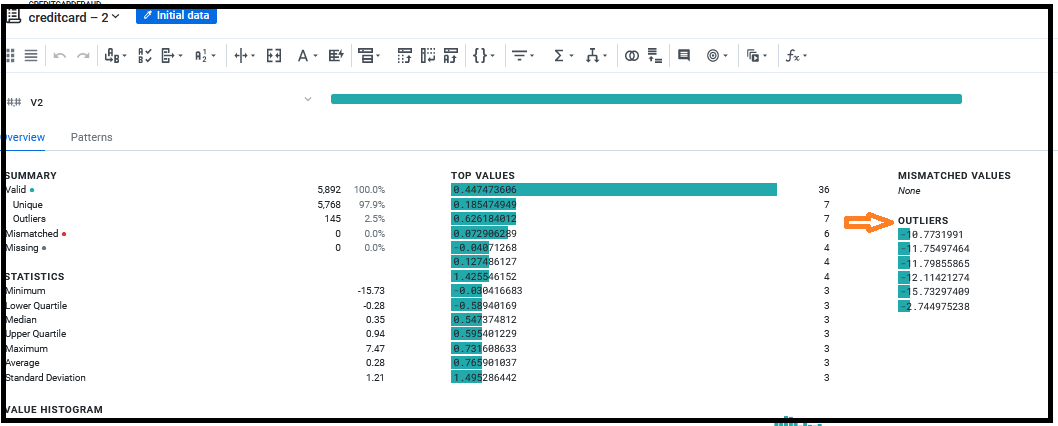
If a dataset contains multiple instances of outliers, it is necessary to review these values and their data in other columns before performing operations to modify or remove these rows, since the removal of these values can become statistically significant.

**Statistical information.**

The visual analysis we have done so far does not allow us to easily identify the presence of outliers in some cases. We can examine detailed statistics of the values in the currently selected column, including data on outliers, available in the Column Details panel. To open the Column Details panel, just select Column Details from the drop-down menu of a column.

From creditcard dataset, for V2 column…click on column…select column details

[..\..\csvFiles\creditcard.csv](file:///C:\Users\User\Documents\MYsqlDocument\csvFiles\creditcard.csv)



In the above panel, a lot of information is available, some of which is as follows:

* Summary stats the count of valid (unique and outliers), mismatched, and missing values
* Statistics such as, min, max, average, lowest and highest quartiles, median, and standard deviation
* Value histogram
* Top values
* Outliers

**Removing Outliers**

After identifying the values as outliers it is necessary to know whether these values ​​are valid or invalid for the dataset. In this case, any outliers are clearly identified and indicated. We can delete the entire row or can modify the row with outlier values.

* Modify the value(take the option Edit with Formula)

set col: V2 value: if(V2 == -10.7731991, 10, V2)

* If the removal of record does not effect on the dataset then we can remove the roe.
* To replace -10.7731991 with the average value of the column, use the following formula: if(v2 = -11.79855865, average(v2), v2)

Click on Run to update your changes.

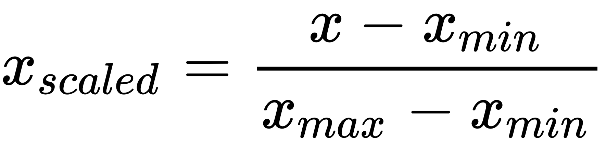
**Scale of features**

Data scaling is a preprocessing technique usually employed before feature selection and classification. Feature scaling is thus required to approximately equalize ranges of the features and make them have approximately the same effect in the computation of similarity. Two methods are usually known for scaling data: normalization, standardization.

Goal: to improve the prediction accuracy and not allow any particular feature to impact the prediction due to a large numeric value range.

**Max-min Normalization**

Performs a linear transformation on the original data. This technique gets all the scaled data in the range (0, 1). The formula to achieve this is the following:



We will use a dataset contained in the Airquality.csv file. It has following fields as

 Ozone: Mean ozone in parts per billion from 1300 to 1500 hours at Roosevelt island

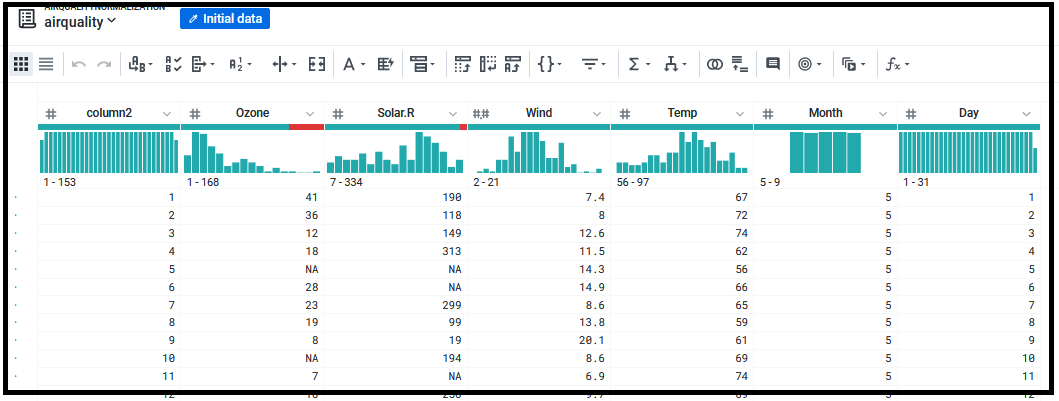
 Solar: Solar radiation in Langleys in the frequency band 4000–7700 Angstroms from 0800 to 1200 hours at central park

 Wind: Average wind speed in miles per hour at 0700 and 1000 hours at LaGuardia airport

 Temp: Maximum daily temperature in degrees Fahrenheit at LaGuardia airport Month

* Day
* Month

Now load the data into flow.



By analysing the upper part of columns

The four columns ozone, solar,wind,temp have different units of measure that is their ranges of values are very different. We can obtain the following ranges:

Ozone: 1 to 168

Solar.R: 7 to 334

Wind: 2 to 21

Temp: 56 to 97

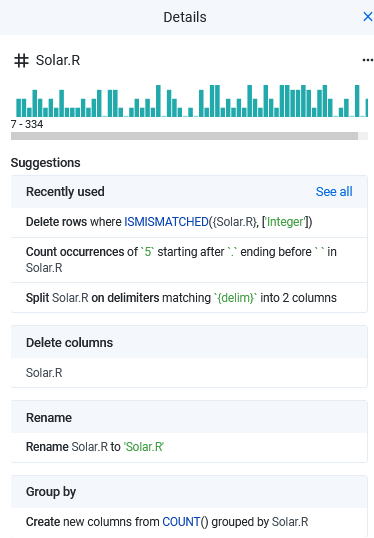
Before proceeding with standardization, we eliminate mismatched values.

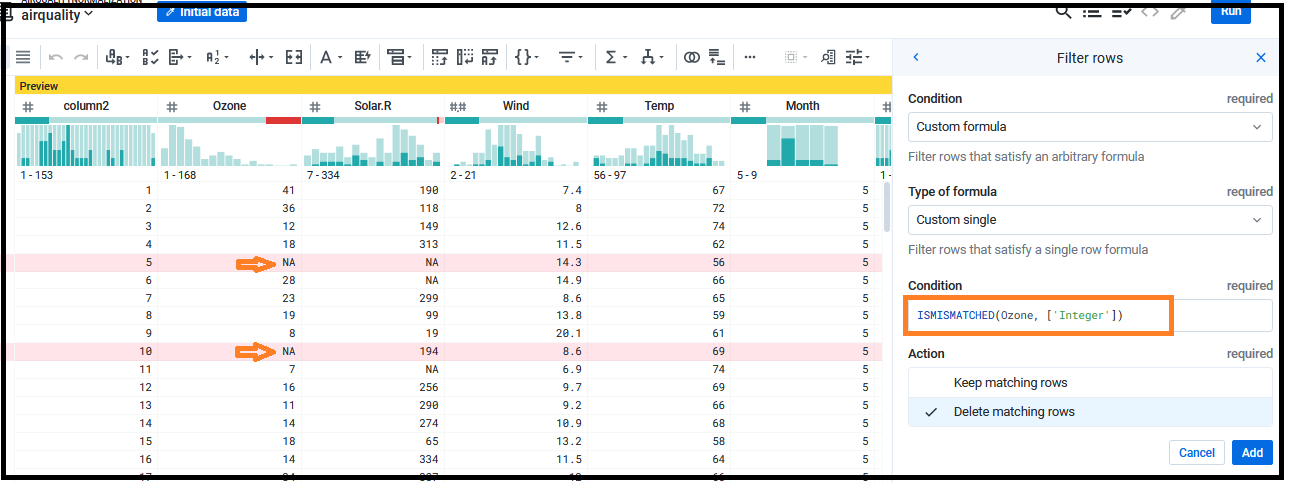
* **Ozone**: 37 mismatched values
* **Solar.R**: 7 mismatched values

Before proceeding with standardization, we have to eliminate some problems highlighted in the data quality bar. Eliminate mismatched values that have been identified.

Click on the column and from the suggestions panel opened on the right of the window…

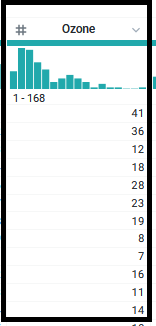
* Delete the lines with mismatched values in Ozone.
* Select column data…go to suggestion box…then Edit the option Delete rows.



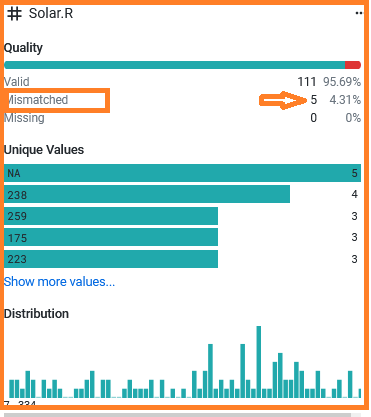


ISMISMATCHED(Ozone, ['Integer'])

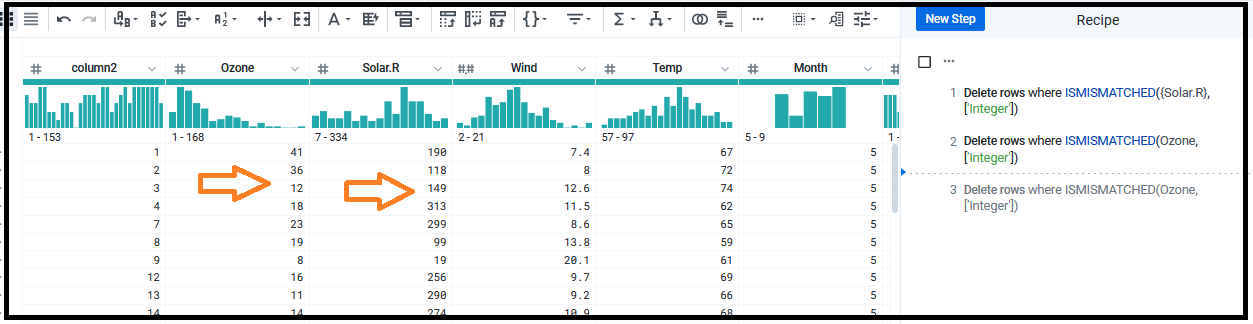
Click on Add. The result comes with removed ‘NA’ values.



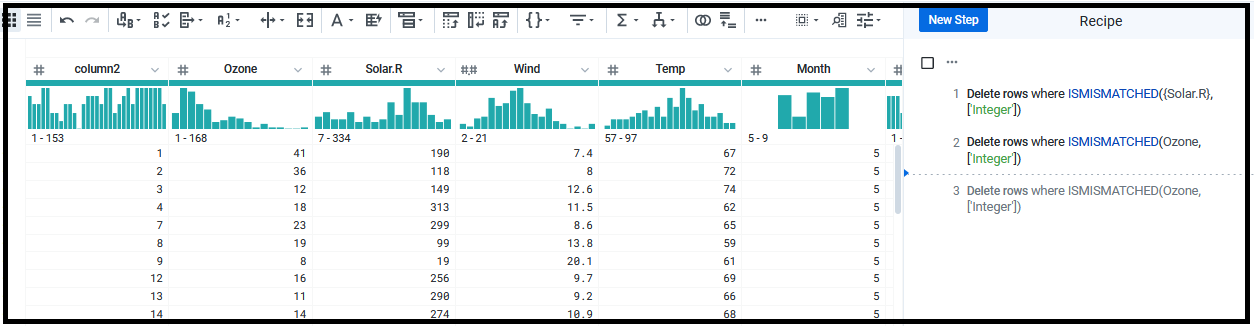
We perform the same operation for the **Solar\_R** column.



ISMISMATCHED({Solar.R}, ['Integer'])



The flow shows as



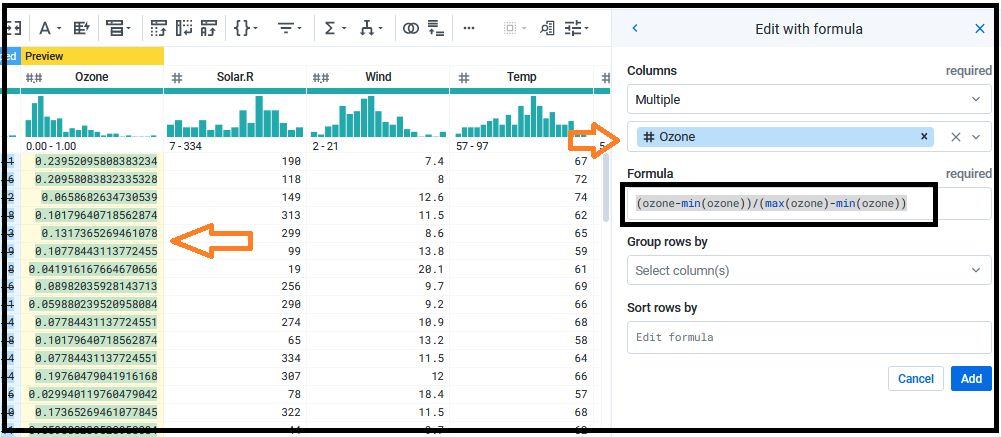
The variables ranges for 4 columns are very varied. We want to eliminate this feature through min-max normalization. We have to calculate the minimum and maximum for each variable. To do this, we can apply two functions available in Google Cloud Dataprep: MIN and MAX.

To apply the formula for normalization, the following steps:

* In the Recipe panel, click on the NEW STEP button.
* From the Transformation drop-down menu, select Edit with formula item.

(ozone-min(ozone))/(max(ozone)-min(ozone))

* Just click on the Add button.



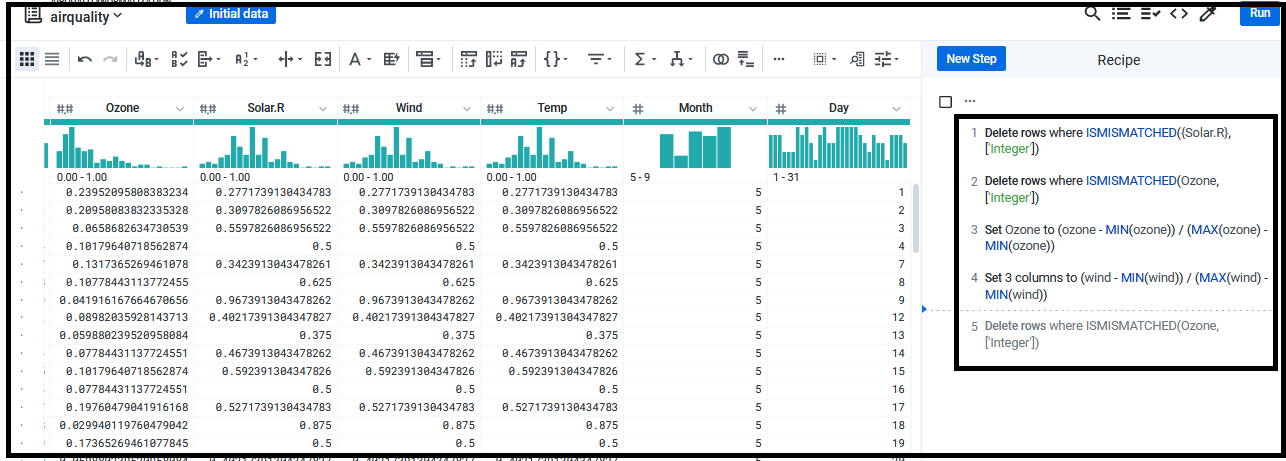
We can apply the same formula to other 3 columns.

* Select columns where you want apply normalizations.

The Transformer page has been changed as follows:

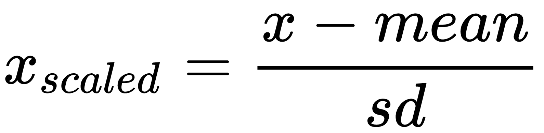


It is apparent that now the data is all between zero and one; this happens for each column of the dataset and then for each variable.



**z score standardization**

This technique consists of subtracting the mean of the column from each value in a column, and then dividing the result by the standard deviation of the column.



Here features are scaled so that they have the properties of a standard normal distribution.

* *μ=0*
* *σ=1*

**z-score** defines, how data differ from mean value of what is observed or measured. Values more than the mean have positive scores, while values less than the mean have negative z score(mean: population mean). Two functions require here

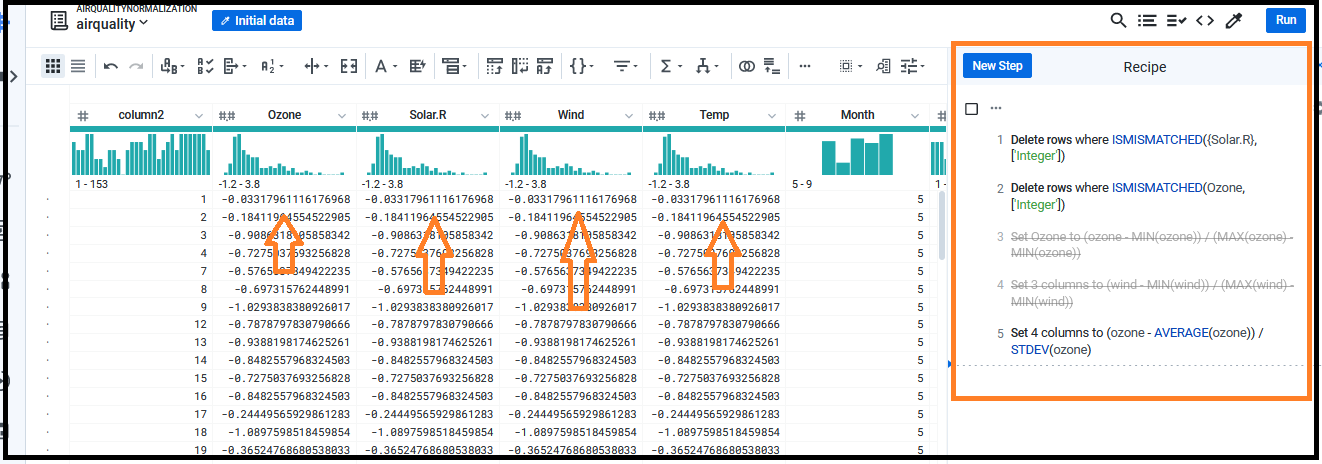
* average/mean : Calculates the average for each column
* stdev: Calculates the standard deviation for each column

Here I referred a dataset called Airquality.csv. Steps given below are

* In the Recipe panel, click on the NEW STEP button.
* From the Transformation drop-down menu, select Apply formula
* From the column box, select the Ozone column.

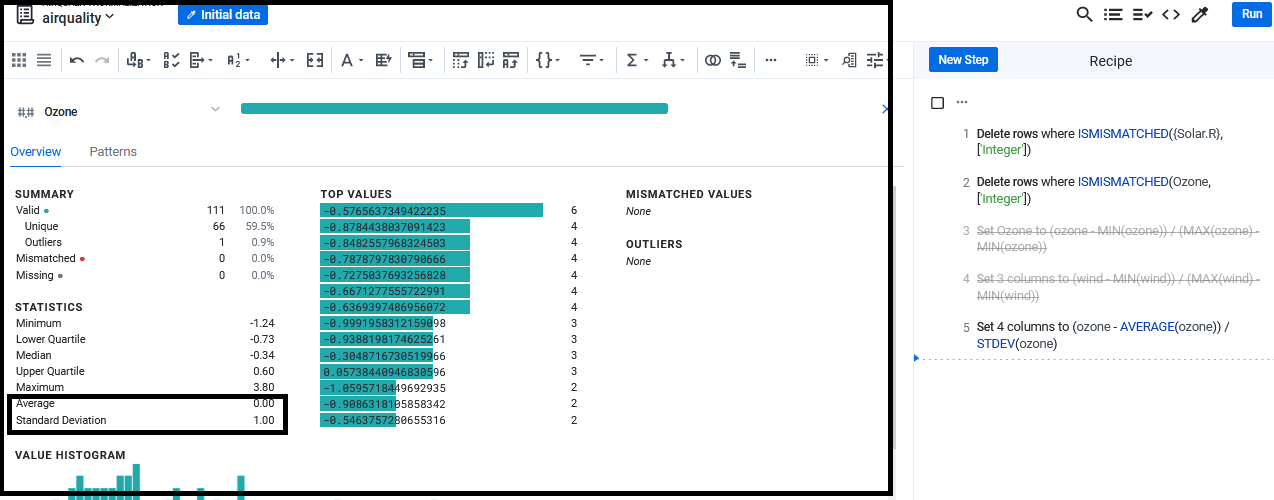
(ozone-average(ozone))-stdev(ozone)

* Click Add.



According to the assumptions, all variables must have average=0 and stdev=1. Let’s verify this …

* Open the Column Details panel
* Select Column Details from the drop-down menu of a column. The following panel is opened:



From above figure we can check that the **Ozone** variable has an average of zero and a standard deviation of one.

Some of other operations can be

* Remove mismatched values

ifmismatched($col, ['Gender'], 'F')

**case( )**

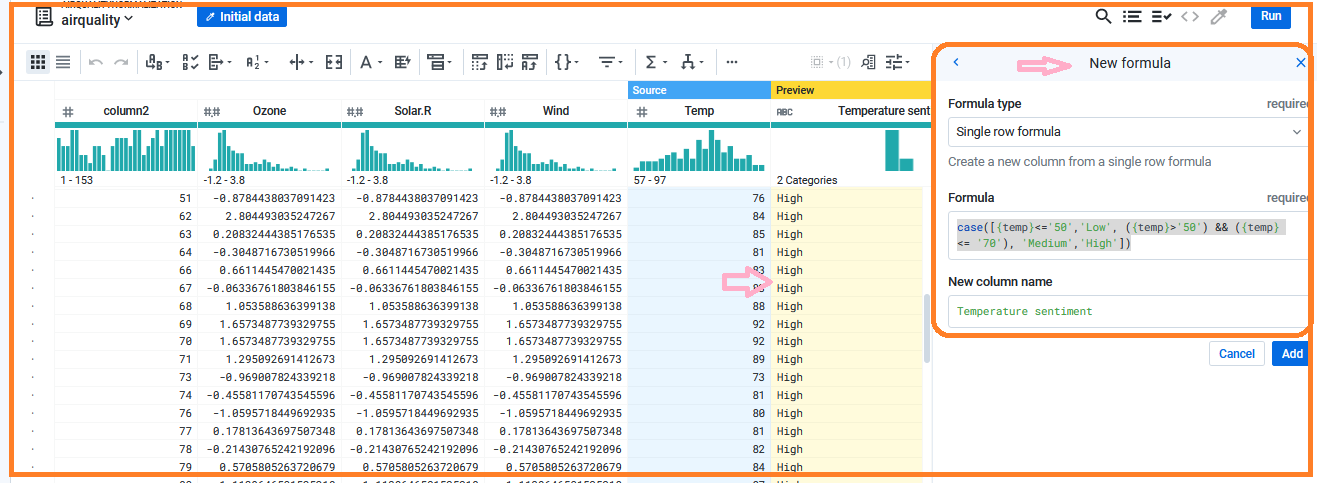
* Case( ) function allows to create multiple conditional tests on asset of expressions with a statement.

Create a new data column.

* Select the column
* select custom formula option
  + New formula message bar

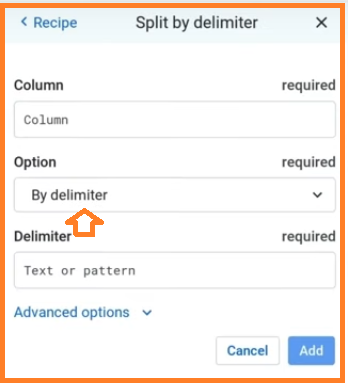
case([{temp}<='50','Low', ({temp}>'50') && ({temp} <= '70'), 'Medium','High'])

* Give a new column name
* Click Add



**By delimeter**

* Separate a column value by ‘/’
* Go to New step.



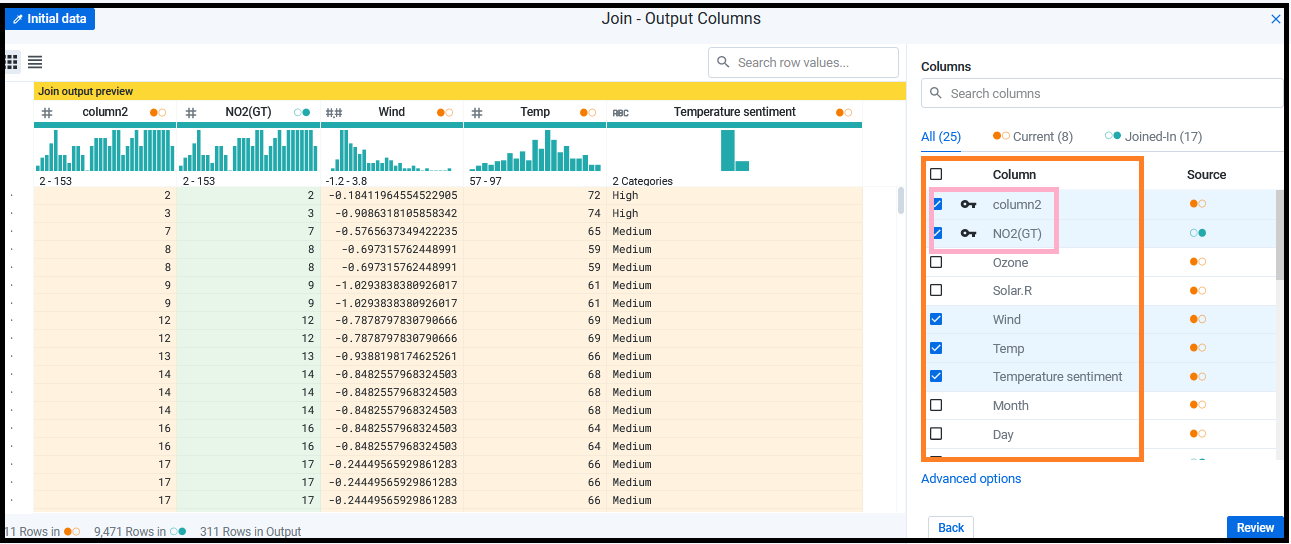
Rename the column

**Join**

* Go to Recipe and click on New step
* Search join
* Select datasets which you want to take for join. It has following options
  + Inner
  + Left
  + Outer
  + Cross
  + right
* I have selected airQuality and AirQualityUCI files. Here column2 and NO2(GT) are automatically detected as the join key.



* Here we can choose the columns which you want to display for both the datasets post the join function.
* Here for this I have choosen Wind, Temp, temperature sentiment columns with column2 and NO2(GT).



* Click on Add to Recipe.

**Remove any symbol from column values**

* Go to NEW Recipe
* Search for text format….do select the column and click add.

**Change data types**

* Click on change type option (select the column and from drop down menu).

End