**Lab for Chapter 2, Part :**

**Data Wrangling**

Contents

1 Exercise 0: Obtain the Diabetes dataset and reference admission data **.................................................3**

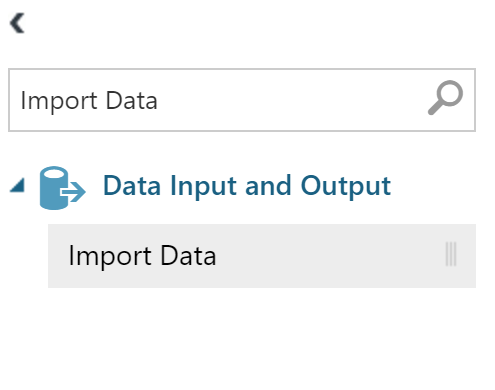
2 Exercise 1: Configure and Run the Missing Data Module **.......................................................................7**

3 Exercise 2: Remove Duplicate Rows **......................................................................................................12**

4 Exercise 3: Filter Outliers**..................................................................................................................... 13**

1 **Exercise 0: Obtain the Diabetes dataset and reference admission data**

1. Create AML experiment (Load Diabetic Data), Drag and drop ‘Import Data’ module from the menu on left.

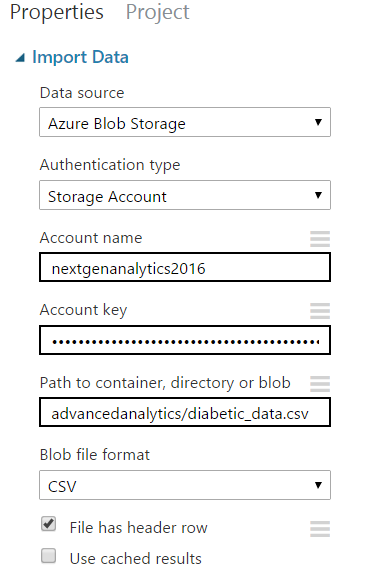


Input the sample blob storage account we have set up for you.

1. **Data Source:** Azure Blob Storage
2. **Authentication type:** Account
3. **Account name:** analyticsbootcamposlo
4. **Account key**:

WxHhL/+EhKva80Y3x25Id4gYndW0H6hzo1ChikRzaD21rtf3Dy2JIy4mNcSWw7ohqqDA1UPB29OTfoG7ogP2+w==

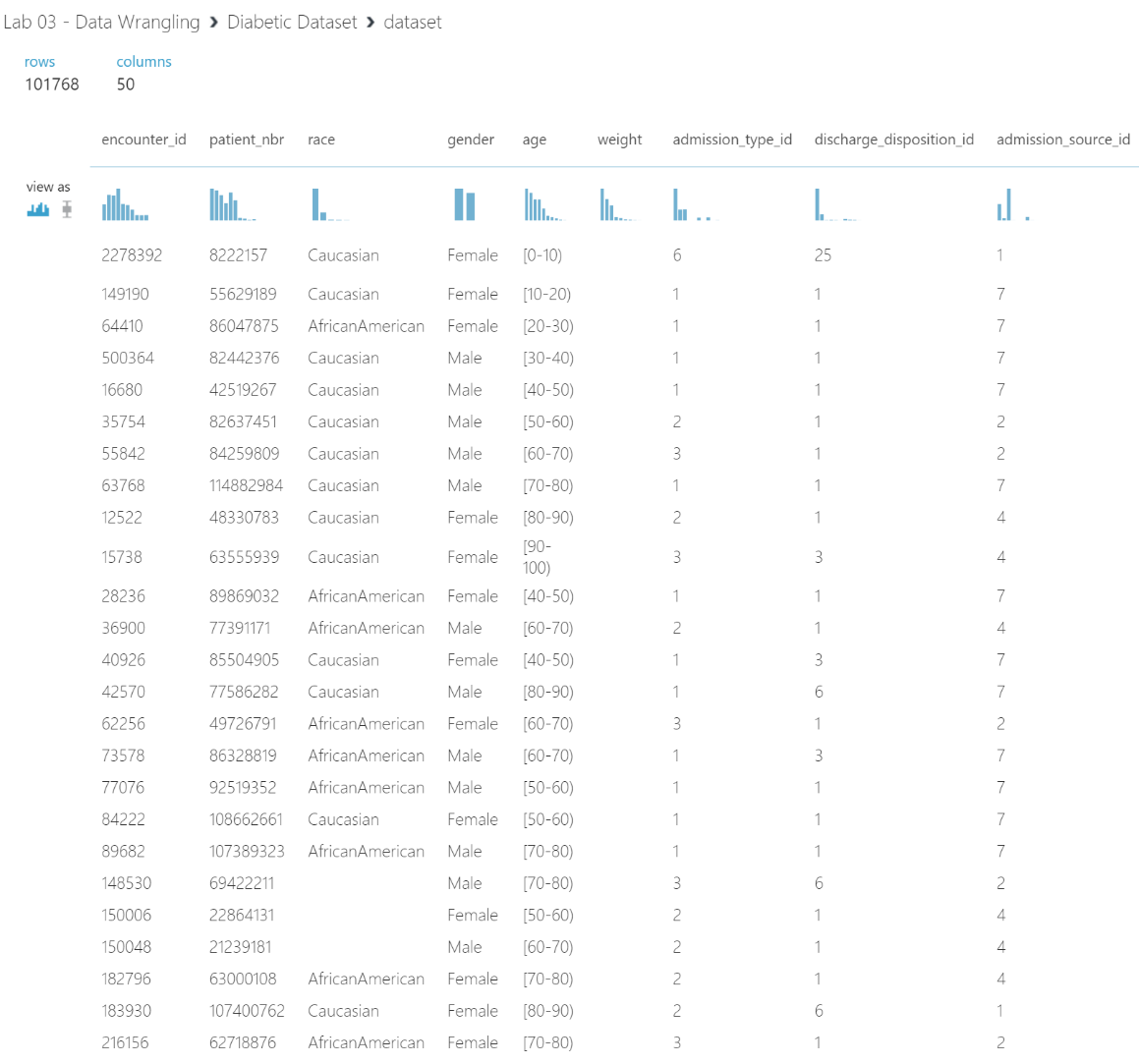
1. **Path to container, directory or blob:** advancedanalytics/diabetic\_data.csv
2. **Blob file format:** CSV
3. **File has header row:** Checked



2. Run the experiment to execute the import and parse

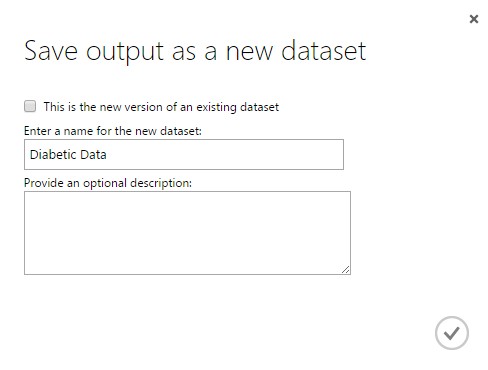


3. Preview the data by visualizing the output of the ‘Import Data’ module. Right click the bottom middle node of the ‘Import Data’ module to access the menu.



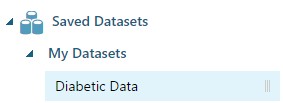
4. The data is not yet saved. Save it to the workspace by clicking “Save as Dataset” in the ‘Import Data’

output node. Name the dataset as “Diabetic Dataset”.



5. Go into any experiment and verify that the dataset has been imported.

The data will be under a directory called “Saved Datasets” within any experiment



6. Repeat the steps 1-5 to load the admissions\_mapping.csv available in blob storage.

7. Create a new AML Experiment called **Diabetic Cleansed data**

8. Drag and Drop Diabetic Data and Admissions\_mapping data set from My Datasets from the left panel

9. Join the Diabetic Data and Admission Mapping data set based on column

Select the **Join** module, and in the **Properties** pane, set the following properties:

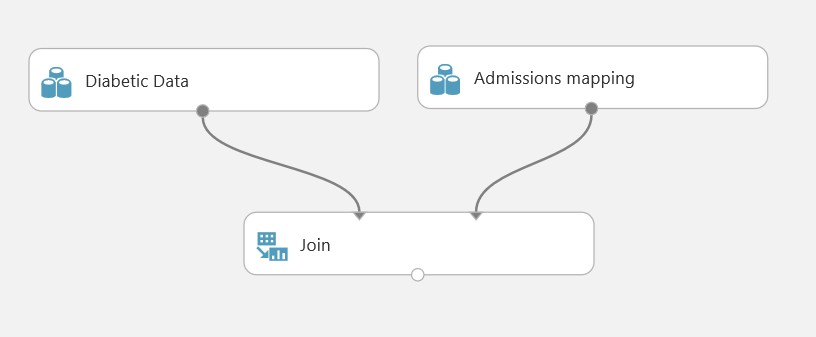
a. **Join key columns for L:** Launch the column selector and select **admission\_type\_id.**

b. **Join key columns for R:** Launch the column selector and select **admission\_type\_id.**

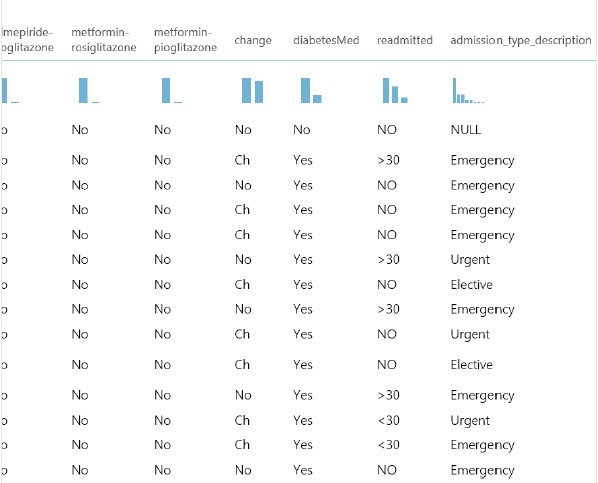
c. **Join type:** Left Outer Join

d. **Keep right key column:** *Unselected*

10. Verify that your experiment resembles the following image, and then save and run

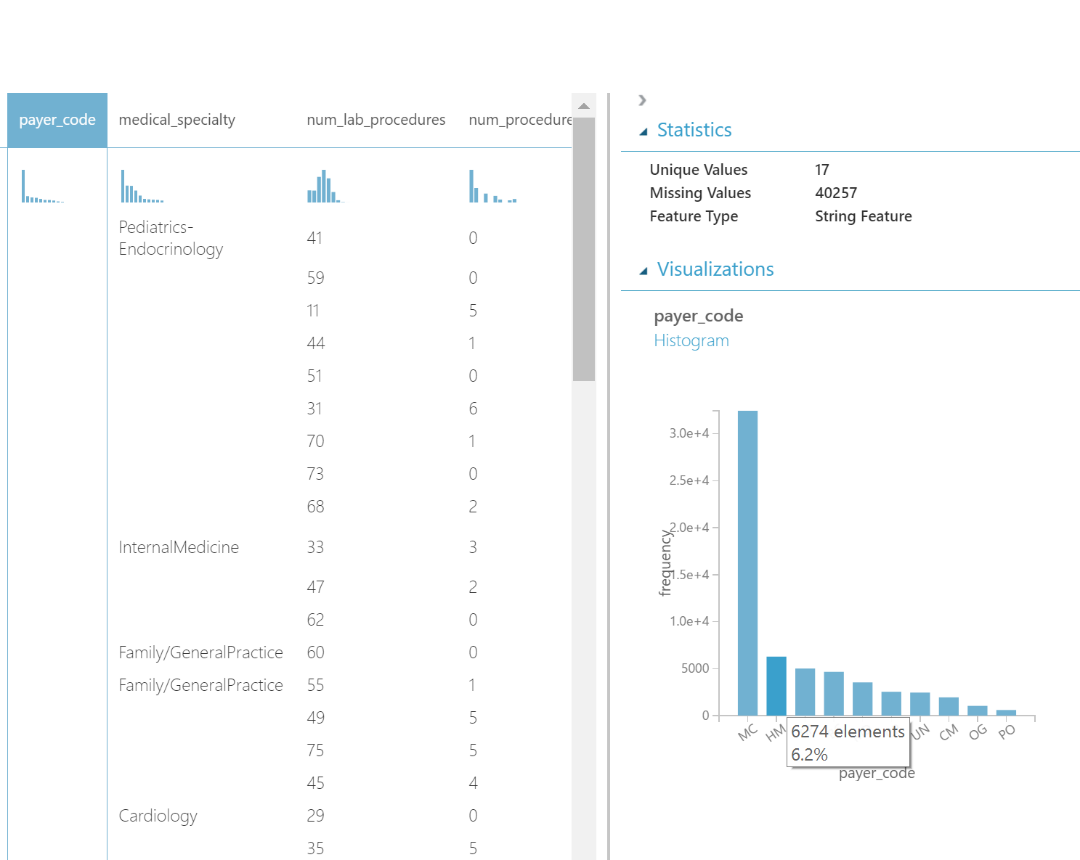


11. When the experiment has finished running, visualize the **Results dataset** output port of the **Join Data** module, and verify that it contains the cleansed patient admission data from the original **Diabetic Data** dataset and the **admission\_type\_description** column from the **Admissions Mapping** dataset as shown in the following image.



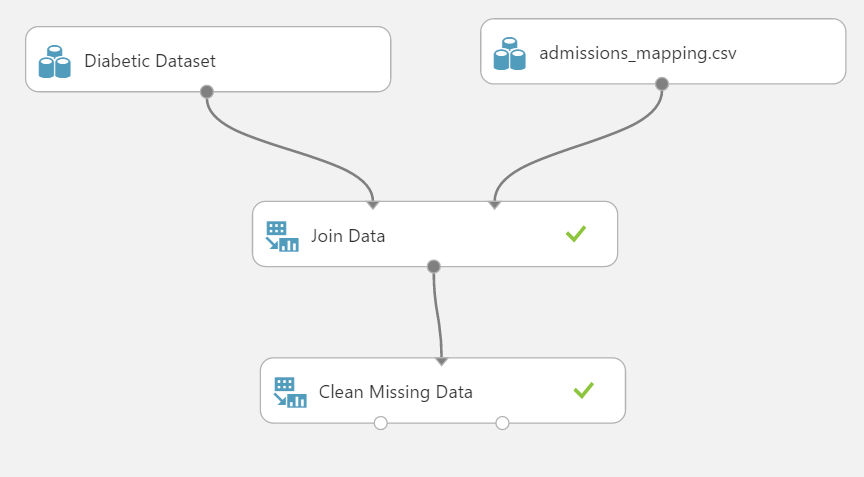
**2 Exercise 1: Configure and Run the Missing Data Module**

1. Visualize the **Dataset** result dataset of the **Join Data** module, and view the statistics for the **payer\_code** column as shown in the following image.
2. Highlight payer\_code as shown below

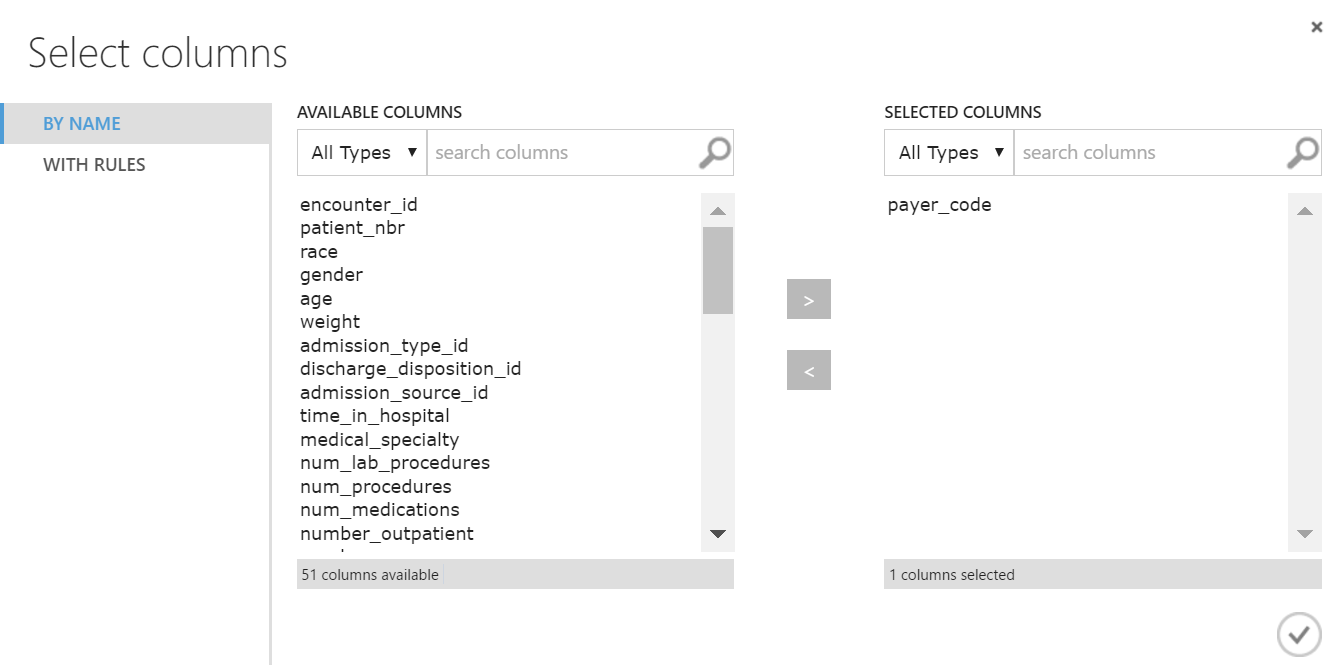


3. Note that the **payer\_code** column has a large number of missing values, and then close the dataset.

4. In the **Diabetes Cleansed Data** experiment, search for the **Clean Missing Data** module and drag it to the canvas under the **Join** module. Then connect the output from the **Join Data** module to the input port of the **Clean Missing Data** module. Your experiment should look similar to the following image.

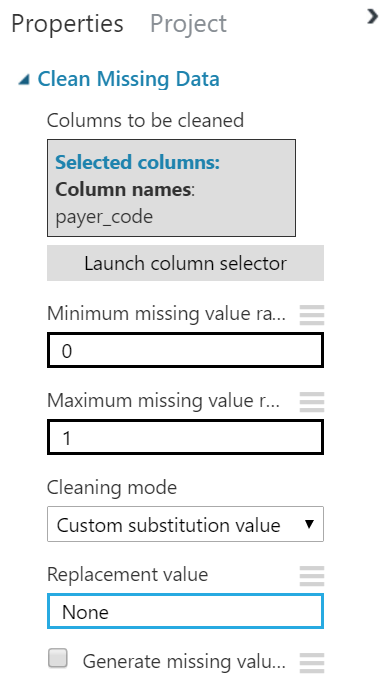


5. Select the **Clean Missing Data** module and in the **Properties** pane, launch the column selector. Then select the option to begin with no columns, and select only the **payer\_code** column as shown in the following image.



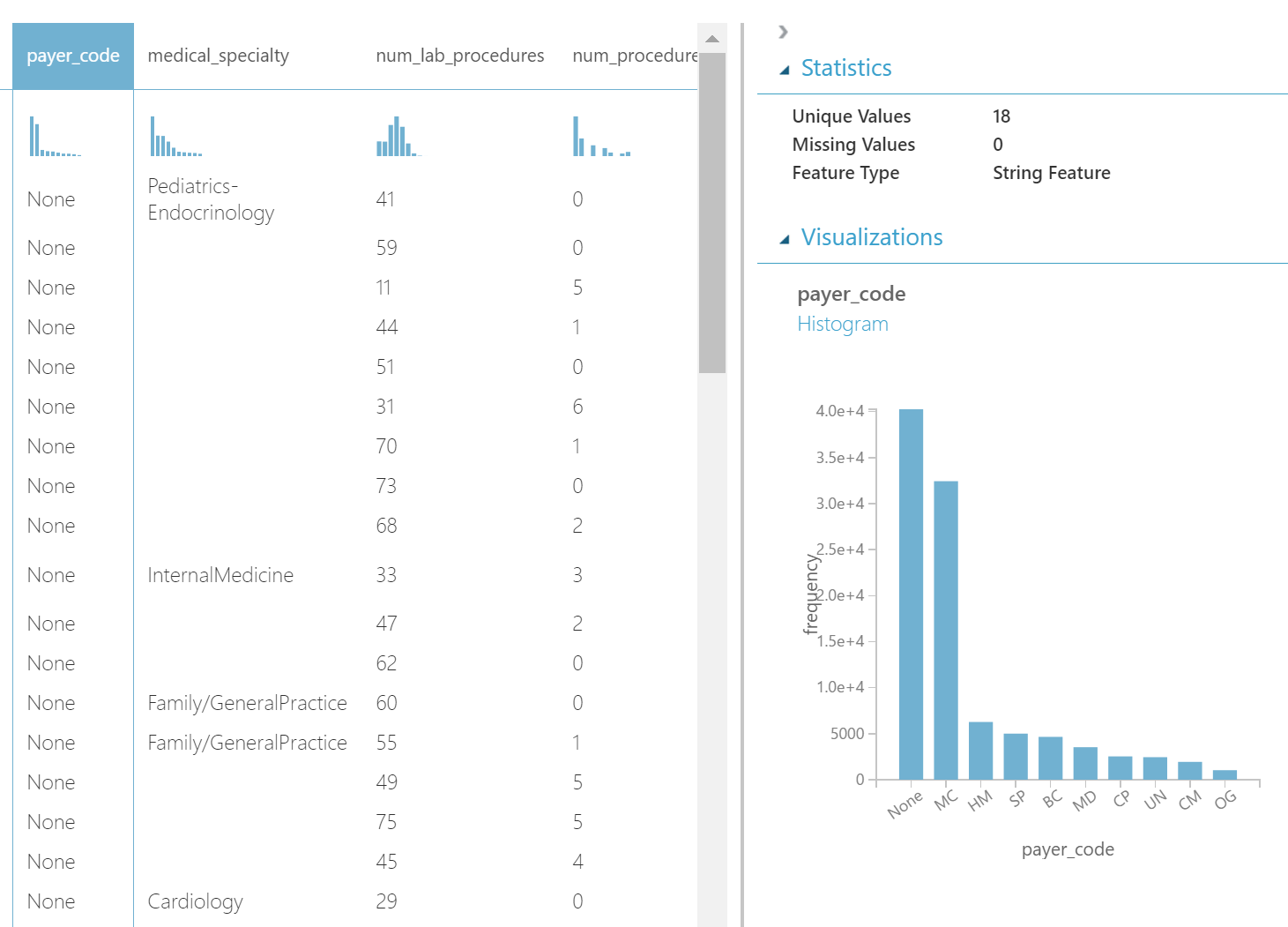
6. In the **Properties** pane, in the **Cleaning mode** list, select **Custom substitution value**, and in the

**Replacement value** textbox, type “None” as shown in the following image.



7. Save and run the experiment. Then, when the experiment has finished running, visualize the **Cleaned dataset** output of the **Clean Missing Data** module, and verify that the value “None” is now used to replace any missing **payer\_code** values; as shown in the following image.

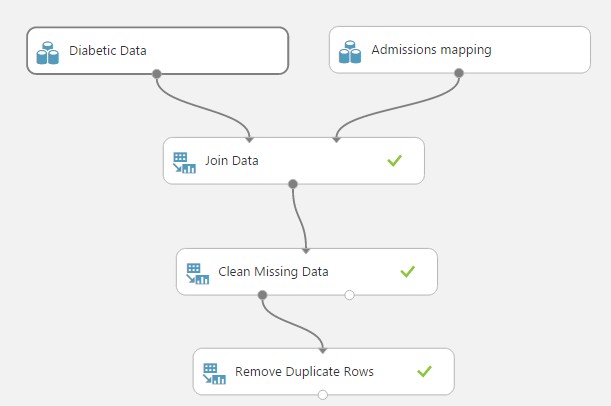
<Restricted> See Avanade’s [Data Classification and Protection Standard 9](https://at.avanade.com/organizations/Policies/Policies2/Forms/Document%20Set/docsethomepage.aspx?ID=670&FolderCTID=0x0120D52000634FE8B87F4B4141A21BFCB3CDC3E3D6&List=caf52708-714a-4e5f-ba50-5017dacf9744&RootFolder=/organizations/Policies/Policies2/Data%20Management)



8. Note the total number of rows (indicated at the top left), and then close the cleaned dataset

**3 Exercise 2: Remove Duplicate Rows**

1. In the **Diabetes Cleansed Data** experiment, search for the **Remove Duplicate Rows** module and drag it to the canvas under the **Clean Missing Data** module. Then connect the **Cleaned dataset** output from the **Clean Missing Data** module dataset to the input port of the **Remove Duplicate Rows** module
2. Select the **Remove Duplicate Rows** module, and in the **Properties** pane, launch the column selector to select the **encounter\_id** column. This column should be unique for each row. After you have selected this column, your experiment should resemble the following image.



3. Save and run the experiment. Then, when the experiment has finished running, visualize the **Results dataset** output of the **Remove Duplicate Rows** module, and verify that the total number of rows has changed (because rows containing duplicate **encounter\_id** values were removed).

4. Close the results dataset.

**4 Exercise 3: Filter Outliers**

In some cases, you may want to determine whether your data contains any values that deviate substantially from the norm, or *outliers*. Outliers can make the data difficult to interpret and make training a model less effective, so detecting the presence of outliers is an essential step in cleaning data for further analysis.

In this exercise you will examine a dataset containing information about forest fires in Portugal (which is provided as a sample dataset in Azure ML), detect the presence of outliers, and then use the **Clip Value** module in Azure ML to remove them.

**Visualize Outliers**

1. In your browser, in Azure ML Studio, create a new blank experiment and name it **Forest Fires Outliers.**

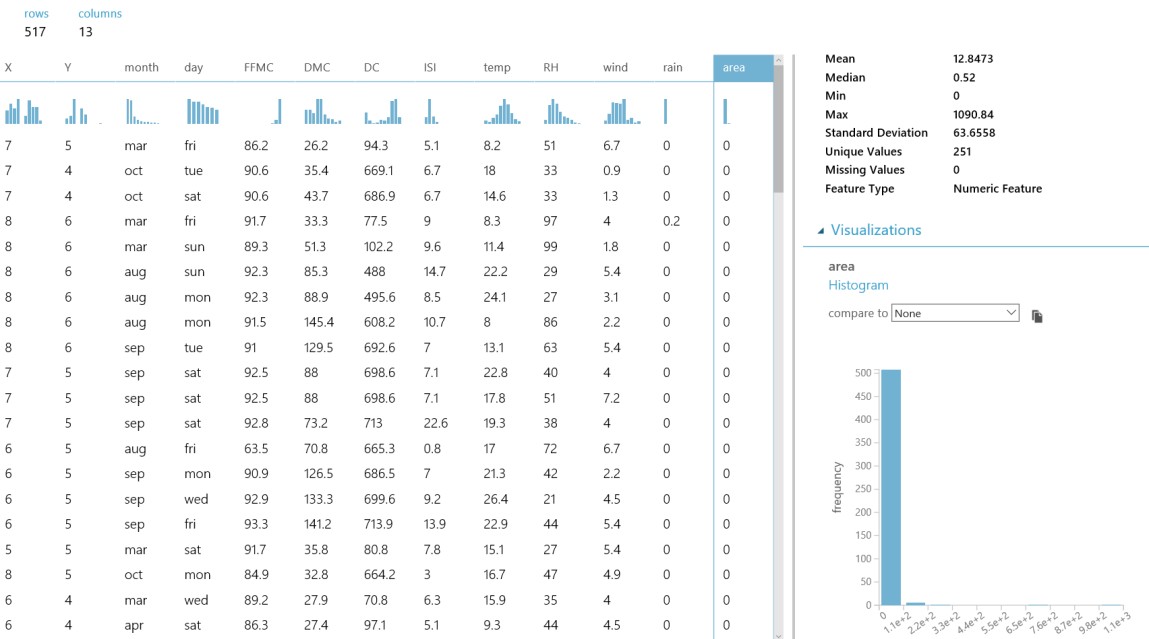
2. In the list of experiment items, search for the **Forest fires data** dataset, and drag it to the canvas.

3. Visualize the **dataset** output port of the **Forest fires data** dataset, and note the total number of rows it contains. Then note that the dataset contains the following columns:

* **X** and **Y**: The coordinates identifying the grid square location of the fire.
* **month** and **day**: The month and day the fire occurred.
* **FMC**, **DMC, DC,** and **ISI**: The numerical values for *Fine Fuel Moisture Code*, *Duff Moisture Code*, *Drought Code*, and *Initial Spread Index*. These are specialist measurements used in the study of forest fires.
* **temp, RH** (*relative humidity*), **wind**, and **rain**: Meteorological measurements at the time of the fire.
* **area:** The size of the area affected by the fire.

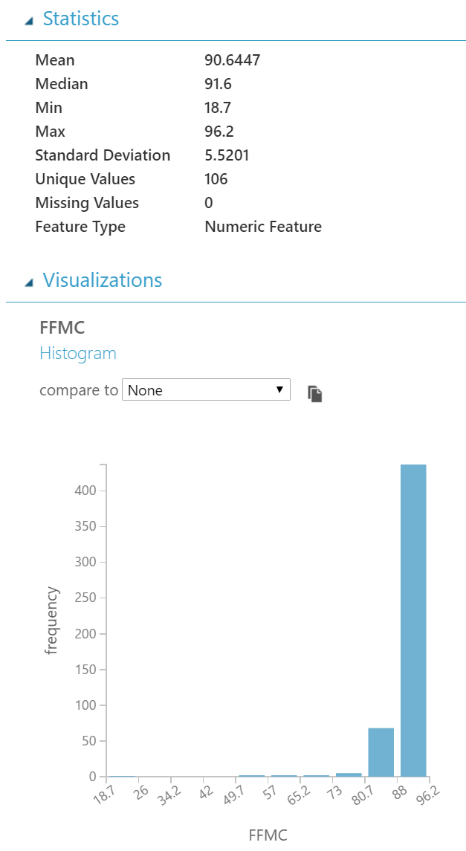
In this experiment, **area** represents the value that can be predicted by the other columns. However, before building a model, you should explore the relationships between this label column and the feature columns; and remove any outlier values that may skew the analysis.

4. Select the **area** column, and note the **Mean, Min, and Max** statistics for this column. Then look at the histogram, as shown here:

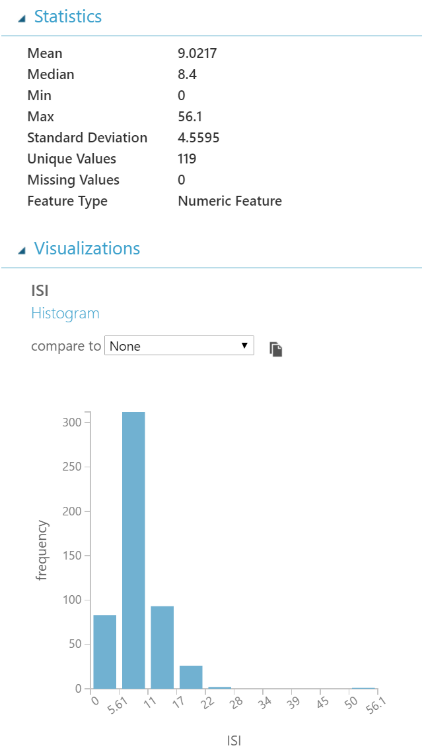


Note that the vast majority of fires are quite small – usually less than one acre; but there are also some very large fires.

5. In the list of columns, select the **FFMC** column and view the statistics for this column.

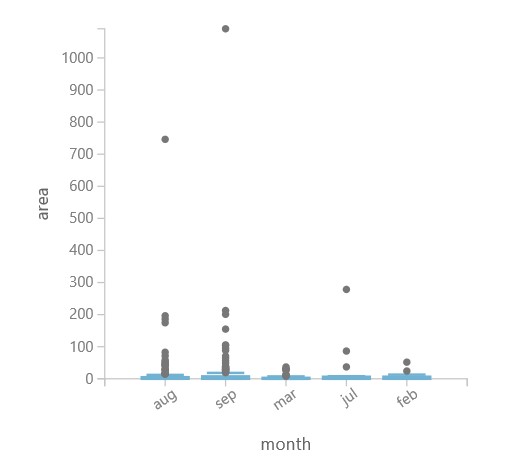


6. Note that the **Min** value for this column is significantly lower than the **Mean** and **Median** values, indicating that most **FFMC** values are high, but there are some low outliers.

7. In the list of columns, select the **ISI** column and view the statistics for this column.

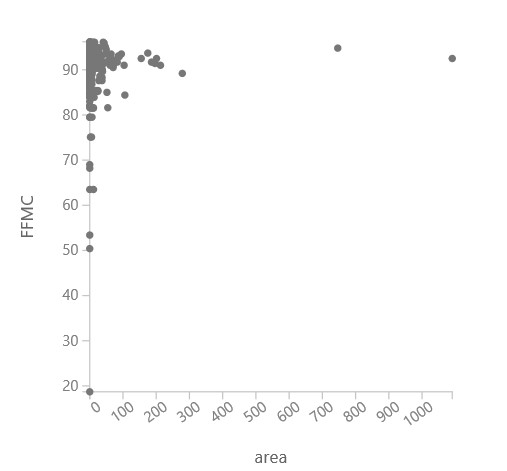
8. Note that the **Max** value for this column is significantly higher than the **Mean** and **Median** values, indicating that most **ISI** values are low, but there are some high outliers.

9. Select the **area** column again, and in the **Visualizations** area, in the **compare to** list, select **month**. This displays a scatter plot chart that shows **month** and **area** values as shown in the following image.



Note that while most plots are shown near the bottom of the chart, there are a few isolated plots that indicate extremely large fires during the late summer months of July, August, and September. This is not unexpected, so the outlier values for large area values have a reasonable explanation, and should remain in the dataset.

10. In the **compare** to list, select **FFMC** and view the resulting scatter plot:

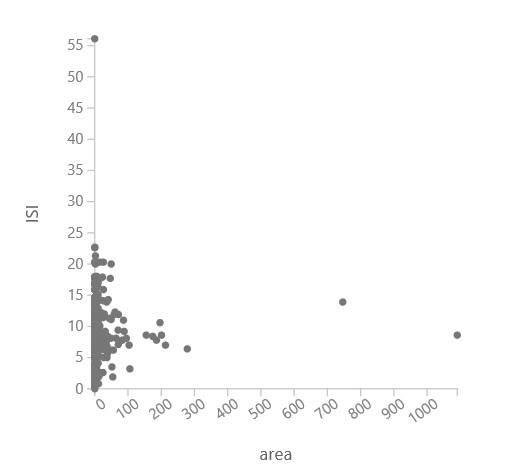


11. The chart shows most plots clustered around the top left, which corresponds to high **FFMC** values and small fires. There are a few plots on the right that indicate large fires with high **FFMC** values; but because we already know that while most fires are small there are a few large ones, this seems to be reasonable.

However, there are a few outlier plots that show low **FFMC** values for small fires; which could potentially

indicate some erroneous data.

12. In the **compare to** list, select **ISI** and view the resulting scatter plot:



13. As with **FFMC**, the chart shows most **ISI** values are clustered together, and we can ignore the outliers to the right that indicate large fires. However, there is an outlier that show a high **ISI** value for a small **area**; which again seems unusual, and may indicate some issues with the **ISI** data.

14. Close the dataset.

**Remove Outliers and Clean Missing Data**

1. In **Forest Fires Outliers** experiment, search for the **Clip Values** module, drag it to the canvas under the **Forest fires data** dataset, and connect the output from the **Forest fires data** dataset to the input port of the **Clip Values** module.

2. Select the **Clip Values** module and in the **Properties** pane, set the following property values:

**Set of thresholds**: ClipSubpeaks

**Lower threshold**: Percentile

**Percentile number for lower threshold**: 1

**Lower substitute value**: Missing

**List of columns**: Launch the column selector and select only the **FFMC** column name.

**Overwrite flag**: Selected

**Add indicator column**: Unselected

This removes the **FFMC** value for any rows where it is in the bottom one percentile.

3. Drag a second **Clip Values** module to the canvas under the first **Clip Values** module. Then connect the output from the first **Clip Values** module to the input port of the second **Clip Values** module.

4. Select the second **Clip Values** module and in the **Properties** pane, set the following property values:

**Set of thresholds**: ClipPeaks

**Upper threshold**: Percentile

**Percentile number for upper threshold**: 99

**Upper substitute value**: Missing

**List of columns**: Launch the column selector and select only the **ISI** column name.

**Overwrite flag**: Selected

**Add indicator column**: Unselected

This removes the **ISI** value for any rows where it is in the top one percentile.

5. In the list of experiment items, search for the **Clean Missing Data** module, and drag it to the canvas under the second **Clip Values** module. Then connect the output from the second **Clip Values** module to the import port of the **Clean Missing Data** module.

6. Select the **Clean Missing Data** module and in the **Properties** pane, set the following property values:

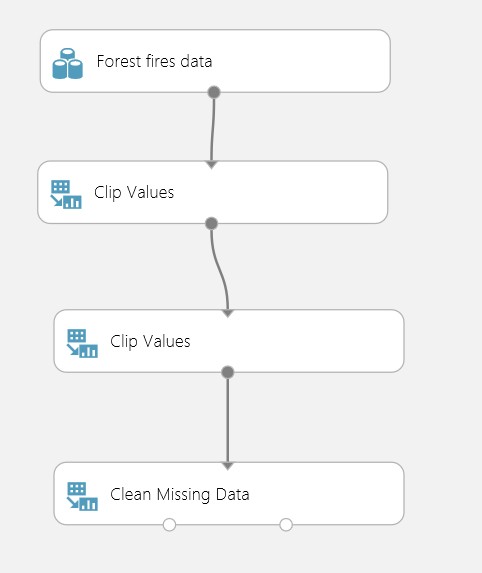
**Columns to be cleaned**: Launch the column selector and configure it to begin with no columns, and include only the **FFMC** and **ISI** column names.

**Minimum missing value ratio**: 0

**Maximum missing value ratio**: 1

**Cleaning mode**: Remove entire row

7. Verify that your experiment resembles the following image, and then save and run it.

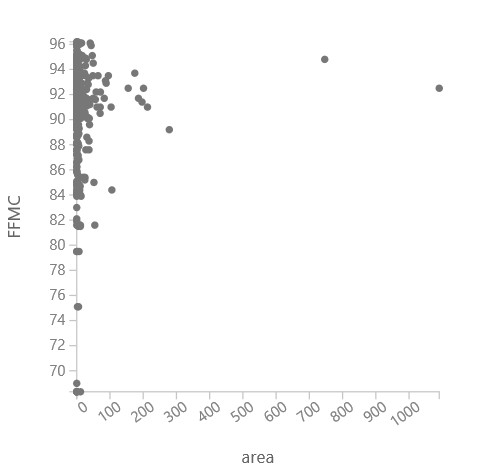


8. When the experiment has finished running, visualize the **Cleaned dataset** output of the **Clean Missing Data** module, and note the number of rows it contains. There should be fewer rows than in the source **Forest fires data** dataset because the rows containing outliers have been removed.

9. Select the **FFMC** column and note the **Mean**, **Min**, and **Max** statistics for this column. These should be higher than in the source data because low-valued outliers have been removed.

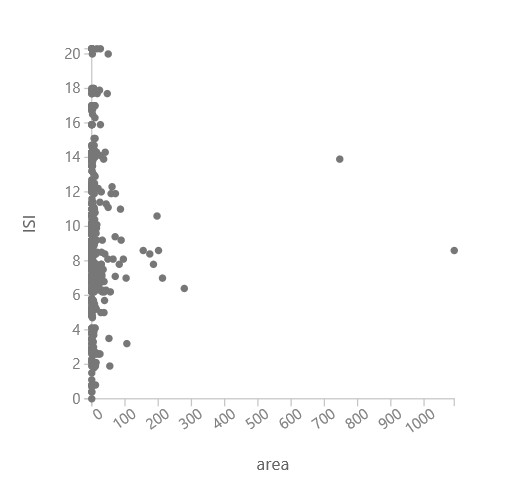
10. Select the **ISI** column and note the **Mean**, **Min**, and **Max** statistics for this column. These should be lower than in the source data because high-valued outliers have been removed.

11. Select the **area** column, and in the **Visualizations** area, in the **compare to** list, select **FFMC** and view the scatter plot chart:



Note that the very low outlier value that was previously present is no longer there.

12. With the **area** column selected, in the **Visualizations** area, in the **compare to** list, select **ISI**, and view the scatter plot chart:



Note that the outlier value that was previously present is no longer there.

13. Close the cleaned dataset.