Data Science and Machine Learning Essentials

Lab 2C – Data Quantization

By Stephen Elston and Graeme Malcolm

Overview

In this lab, you will learn how to sample and quantize data in Microsoft Azure Machine Learning (Azure ML).

Note: This lab builds on knowledge and skills developed in the previous labs. If you have little experience with using Python or R in Azure ML, and you did not complete the previous labs, you are advised to do so before attempting this lab.

What You'll Need

To complete this lab, you will need the following:

- An Azure ML account
- A web browser and Internet connection
- The lab files for this lab

Note: To set up the required environment for the lab, follow the instructions in the **Setup** document for this course. Then download and extract the lab files for this lab.

Categorizing Data with Azure ML Modules

In this exercise, you will use the **Metadata Editor** module to convert integer columns to *Categorical Features*. Creating categorical features from features with limited numbers of values is useful in many modeling and visualization situations. You will then use the **Quantize Data** module to assign ranges of values from continuous variables into a limited number of levels.

Create an Experiment

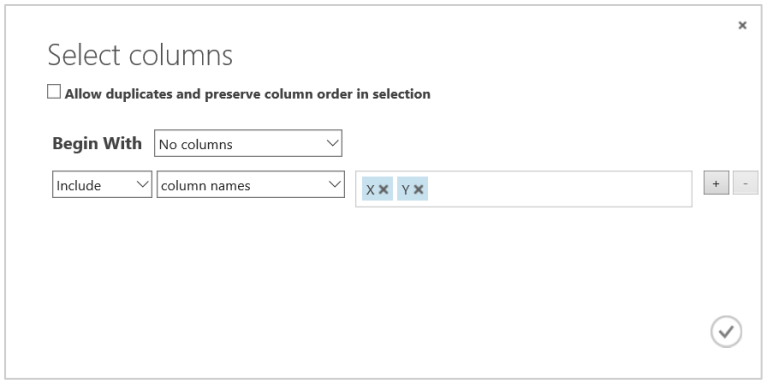
1. Open a browser and browse to https://studio.azureml.net. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment with the title **Forest Fires Quantization**.

Create Categorical Features

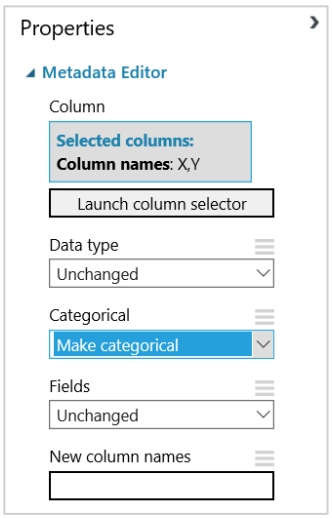
1. In the **Forest Fires Quantization** experiment, search for the **Forest fires data** dataset and drag it to the canvas. This dataset contains data about forest fires in Portugal.
2. Visualize the **dataset** output port of the **Forest fires data** dataset, and note the total number of rows it contains. Then note that the dataset contains the following columns:
    - **X** and **X**: The coordinates identifying the grid square location of the fire.
    - **month** and **day**: The month and day the fire occurred.
    - **FFMC**, **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 numeric value (or *label*) that can be predicted by the other columns. Some of the other columns (or *features*) represent numerical values, such as measurements of wind and rain; but others can be used to represent category indicators.

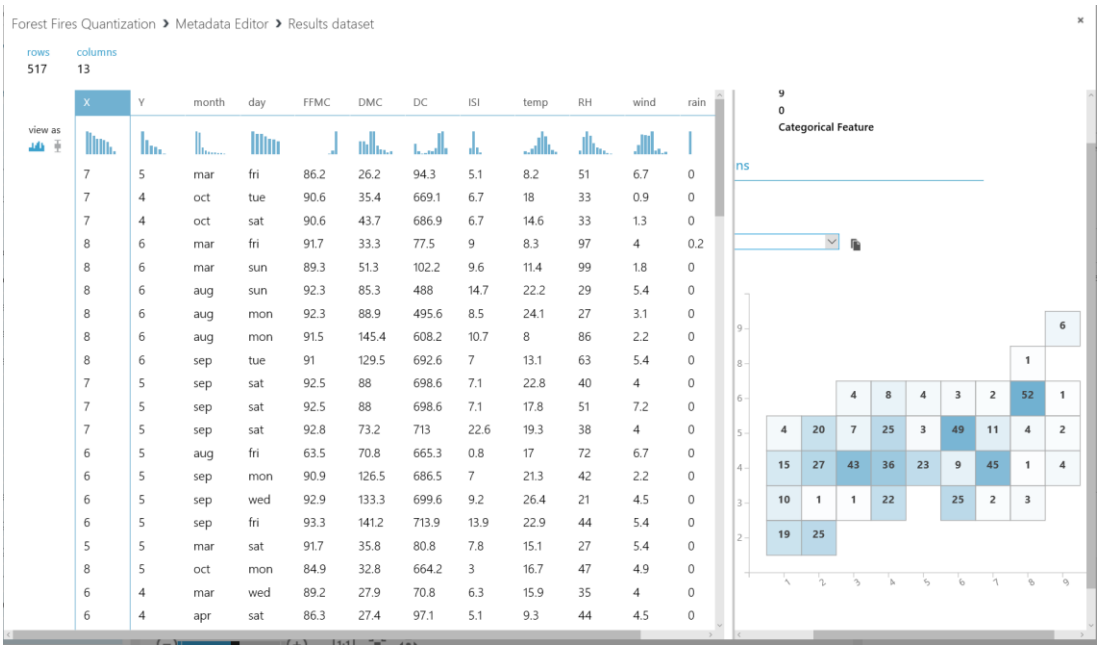
3. Then select the **X** column header and note in the **Statistics** area that this column is a **Numeric Feature** (in other words, it is considered by Azure ML to be a continuous variable)
4. Select the **Y** column, and note that it too is a **Numeric Feature**. Then close the dataset.
   The **X** and **Y** columns are coordinates that indicate the grid reference of each fire, and you want them to be treated as fixed, categorical values so you can easily compare fires on the same lateral and longitudinal planes.
5. Search for the **Metadata Editor** module and drag it to the canvas under the **Forest fires data** dataset. Then connect the output port from the **Forest fires data** dataset to the input port of the **Metadata Editor** module. Your experiment should look similar to the following image.
6. Select the **Metadata Editor** module and in the **Properties** pane, launch the column selector. Then select the **X** and **Y** columns as shown in the following image, and click **OK**.



7. In the **Properties** pane, in the **Categorical** list, select **Make categorical** as shown in the following image.



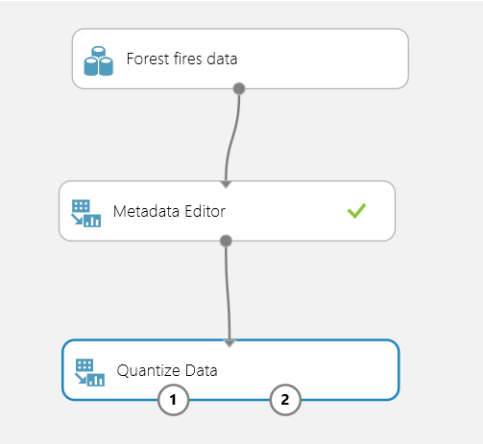
8. Save and run the experiment. Then, when the experiment has finished running, visualize the output of the **Metadata Editor** module, and select the **X** column to verify that it is now a **Categorical Feature**.
9. In the **Visualizations** area, in the **Compare to** list, select **Y** and note that because these values are categorical, it is now possible to count the number of fires at each coordinate as shown in the following image.



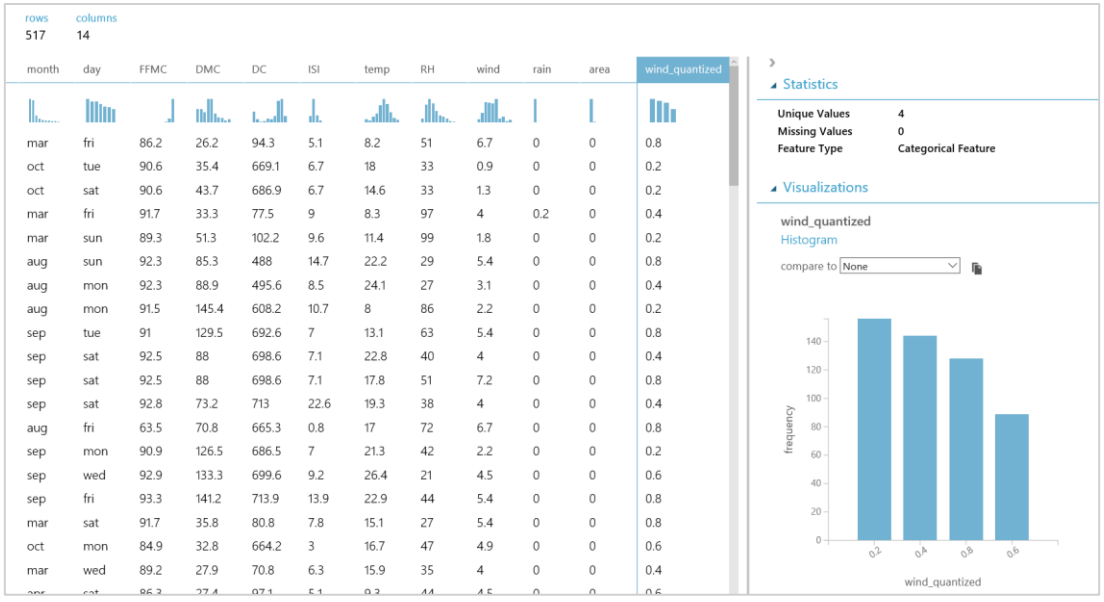
10. Select the **wind** column header. This column contains a measure of wind speed. Note that this column is a **Numeric Feature**, and that there are 21 unique values ranging from 0.4 to 9.4. To simply the analysis of the effect of wind on fire area, you will quantize these values into four categories, each representing a range of wind speeds.
11. Close the results dataset.

Quantize a Variable

1. In the **Forest Fires Quantization** experiment, search for the **Quantize Data** module and drag it to the canvas under the **Metadata Editor** module. Then connect the output from the **Metadata Editor** module to the **Quantize Data** module as shown in the following image.



2. Select the **Quantize Data** module, and in the **Properties** pane, select the following options:
    - **Binning mode**: Quantiles
    - **Number of bins**: 4
    - **Quantile normalization**: PQuantile
    - **Columns to bin**: Launch the column selector and configure it to begin with no columns and include only the **wind** column name.
    - **Output mode**: Append
    - **Tag columns as categorized**: Selected
3. Save and run the experiment. Then, when the experiment has finished running, visualize the **Quantized Dataset** output of the **Quantize Data** module and note that a new categorical feature named **wind_quantized** with four unique values has been added to the data as shown in the following image:



4. Note that the original numeric **wind** feature is retained in the dataset – to replace it, you could select an **Output mode** value of **InPlace**.
5. Close the quantized dataset.

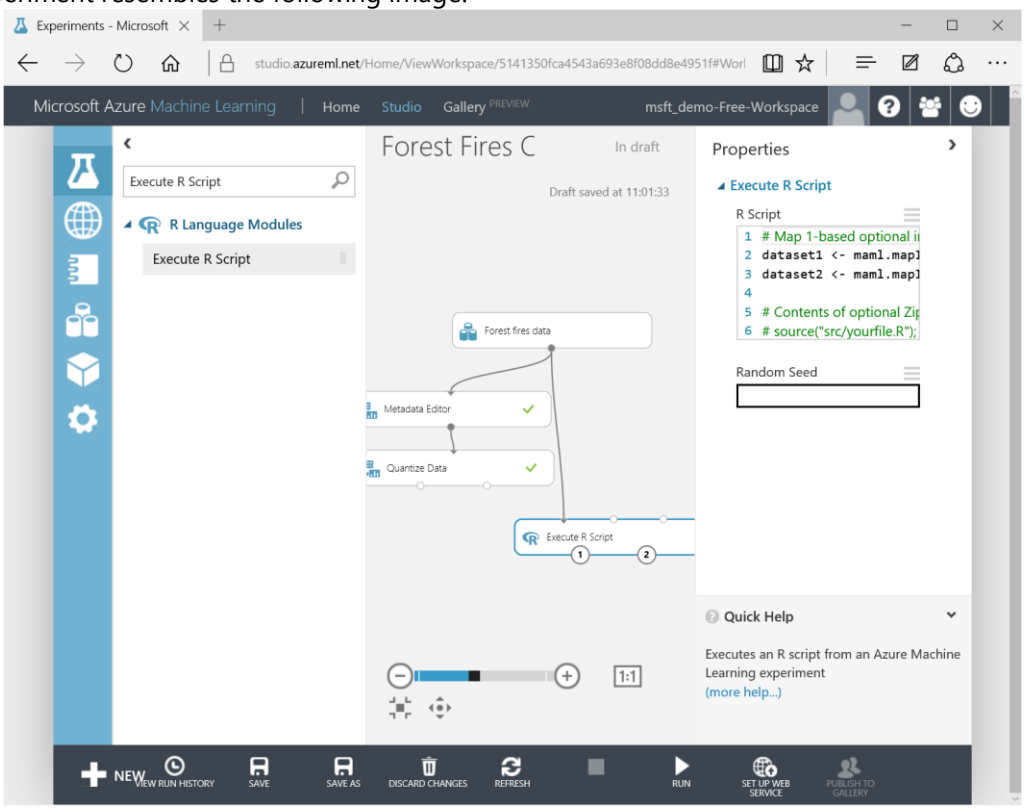
Using Custom Code to Quantize Data

In this exercise, you will use a custom R or Python script to quantize and sample data.

Quantize Data with R

If you prefer to work with Python, skip this procedure, and complete the next procedure: *Quantize data with Python*.

1. In the **Forest Fires Quantization** experiment, search for the **Execute R Script** module and drag it to the canvas under the **Forest fires data** dataset. Then connect the output port of the **Forest fires data** dataset to the **Dataset1** input port of the **Execute R Script** module so that your experiment resembles the following image:



2. Select the **Execute R Script** module, and in the **Properties** pane, replace the default R script with the following code, which you can find in the **Quantize.R** script in the lab files.

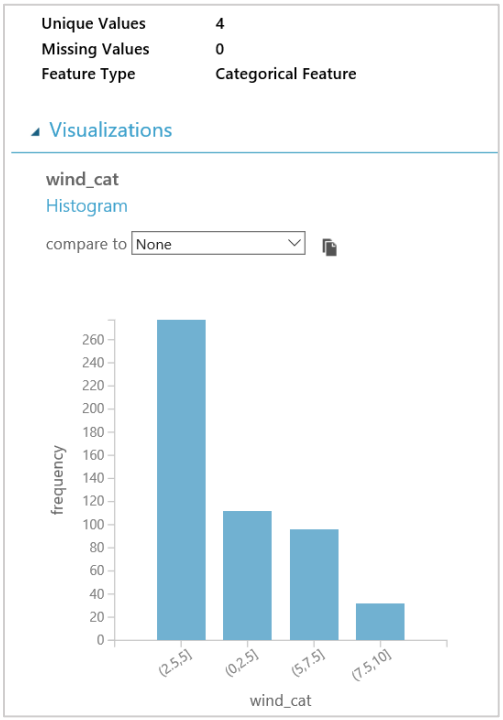
```
frame1 <- maml.mapInputPort(1)
## Bin the wind column into 4 categories.
bins <- c(0, 2.5, 5, 7.5, 10)
frame1[, "wind_cat"] <- cut(frame1[, "wind"], breaks = bins)

## Create categorical variables from the X and Y columns.
frame1[, c("X", "Y")] <- lapply(frame1[, c("X", "Y")], as.factor)

## Output the data frame.
maml.mapOutputPort('frame1')
```

Tip: To copy code in a local code file to the clipboard, press **CTRL+A** to select all of the code, and then press **CTRL+C** to copy it. To paste copied code into the code editor in the Azure ML **Properties** pane, press **CTRL+A** to select the existing code, and then press **CTRL+V** to paste the code from the clipboard, replacing the existing code.

This code uses the **cut** function to bin the **wind** column into four categories based on specific values. Alternatively, you could use the R **quantile** function to automatically compute the bin edges at specific percentage levels.

The code also converts the **X** and **Y** columns to categorical (factor) columns by applying the **as.factor** function. This approach works since X and Y have only a small number of unique integer values.

3. Save and run the experiment. When the experiment has finished running, visualize the **Results dataset** output of the **Execute R Script** module, and select the X and T columns in turn to verify that they are categorical features. Then select the new **wind_cat** column and note that it contains four values that represent ranges of wind speed:

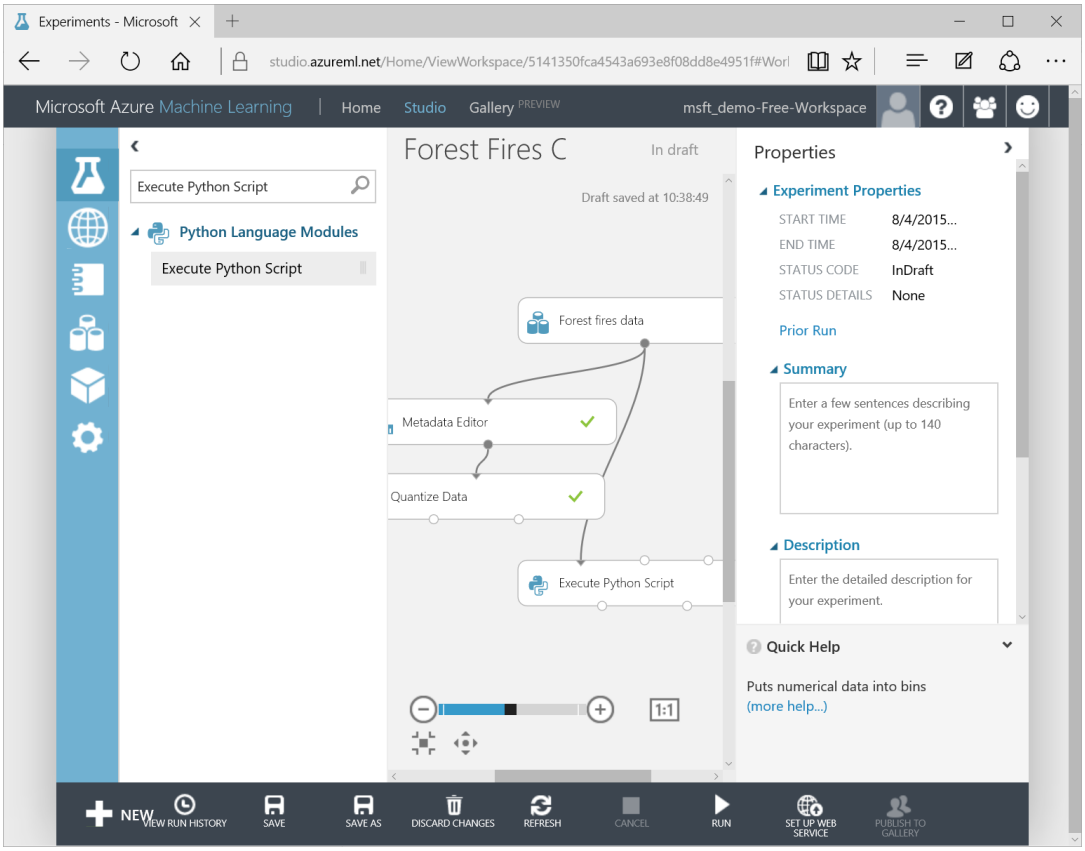


4. Close the Results dataset.

Quantize Data with Python

If you prefer to work with R, skip this procedure, and complete the previous procedure *Quantize Data with R*.

1. In the **Forest Fires Quantization** experiment, search for the **Execute Python Script** module and drag it to the canvas under the **Forest fires data** dataset. Then connect the output port of the **Forest fires data** dataset to the **Dataset1** input port of the **Execute Python Script** module so that your experiment resembles the following image:



2. Select the **Execute Python Script** module, and in the **Properties** pane, replace the default Python script with the following code, which you can find in the **Quantize.py** script in the lab files.

```
def azureml_main(frame1):

## Quantize the wind into 4 categories using cut with explicit columns.
  import pandas as pd
  bins = [0, 2.5, 5, 7.5, 10]
  frame1['wind_cat'] = pd.cut(frame1['wind'], bins)
  return frame1
```

Tip: To copy code in a local code file to the clipboard, press **CTRL+A** to select all of the code, and then press **CTRL+C** to copy it. To paste copied code into the code editor in the Azure ML **Properties** pane, press **CTRL+A** to select the existing code, and then press **CTRL+V** to paste the code from the clipboard, replacing the existing code.

This code uses the Pandas **cut** function to bin the **wind** column into four categories based on specific values. Alternatively, you could use the **quantile** function to automatically compute the bin edges at percentage levels.

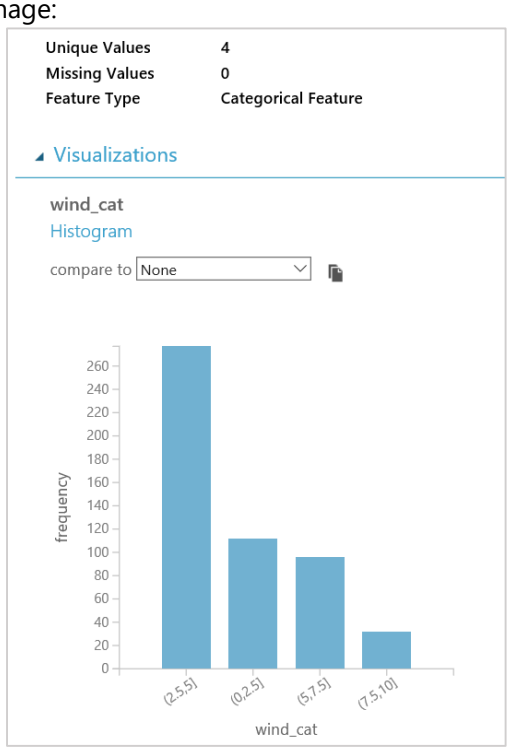
Note: Conversion to categorical variables was introduced in Pandas 0.15.0. Presently Azure ML is running Pandas 0.14.0, so this capability is not available. You can display the current version of Pandas by running the following Python command (after importing the pandas module as **pd**):

```
print("The Pandas version is " + pd.__version__)
```

When Pandas 0.15.0 is available in Azure ML, you will be able to extend this script to convert the X and Y columns to category features by using the following code:

```
frame1["X "] = frame1["X"].astype('category')
frame1["Y "] = frame1["Y"].astype('category')
```

3. Save and run the experiment. When the experiment has finished running, visualize the **Results dataset** output of the **Execute Python Script** module, and note that a category feature column named **wind_cat** that contains four unique values has been created by your Python code, as shown in the following image:



4. Close the Results dataset.

Summary

In this lab, you have used built-in Azure ML modules and custom script to quantize data.

Note: The experiment created in this lab is available in the Cortana Analytics library at http://gallery.cortanaanalytics.com/Collection/5bfa7c8023724a29a41a4098d3fc3df9.