Microsoft

Data Science and Machine Learning Essentials

Lab 1 – Getting Started with Azure Machine Learning

Overview

In this lab, you will learn how to open and navigate the Microsoft Azure Machine Learning (Azure ML) Studio. You will also learn how to create and run experiments in Azure ML.

Note: The goal of this lab is to familiarize yourself with the Azure ML environment and some of the modules and techniques you will use in subsequent labs. Details of how to visualize and manipulate data, and how to build and evaluate machine learning models will be discussed in more depth later in the course.

What You'll Need

To complete this lab, you will need the following:

- An Azure ML account
- A web browser and Internet connection
- The files for this lab
- Optionally, the Anaconda Python distribution or R and R Studio if you want to edit the code examples give in this lab

Note: To set up the required environment for the lab, follow the instructions in the Setup Guide for this course.

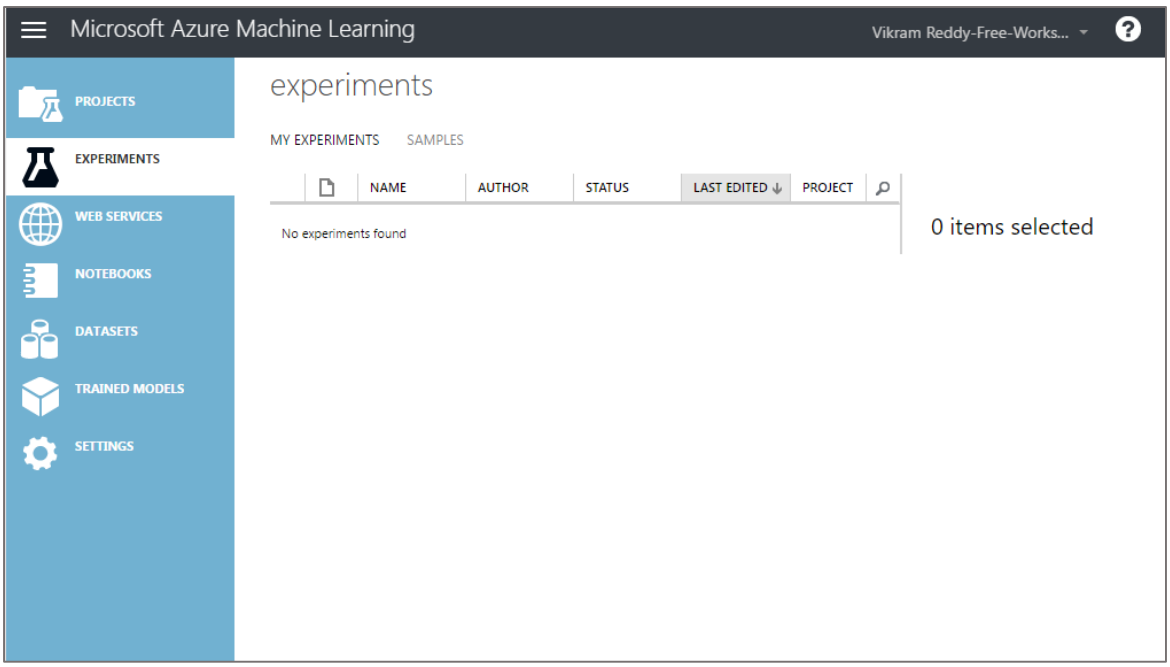
Creating an Azure ML Experiment

Azure ML enables you to create experiments in which you can manipulate data, create predictive models, and visualize the results. In this exercise, you will create a simple experiment to explore a sample dataset that contains data on bank customers. Your goal is to predict the creditworthiness of these customers.

Sign into Azure ML Studio

1. Browse to https://studio.azureml.net and sign in using the Microsoft account associated with your free Azure ML account.
2. If the **Welcome** page is displayed, close it by clicking the **OK** icon (which looks like a checkmark). Then, if the **New** page (containing a collection of Microsoft samples) is displayed, close it by clicking the **Close** icon (which looks like an X).

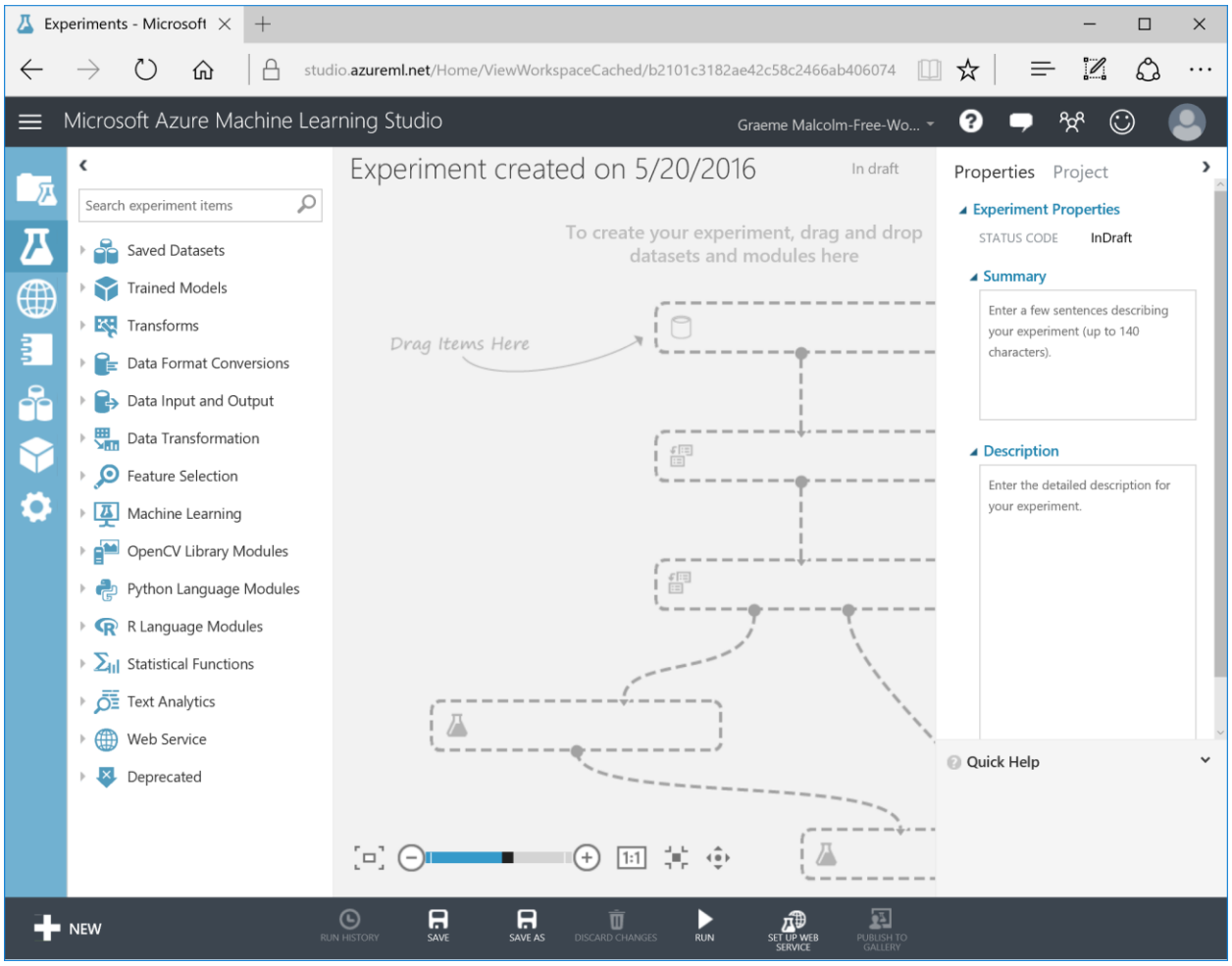
3. You should now be in Azure ML Studio with the **Experiments** page selected, which looks like the following image (if not, click the **Studio** tab at the top of the page).



Tip: To organize the labs for this course you can click **Projects** and create a new *project*. You can add your datasets and experiments to this project so they are easy to find in the future**.**

Create an Experiment

1. In the studio, at the bottom left, click **NEW**. Then in the **Experiment** category, in the collection of Microsoft samples, select **Blank Experiment**. This creates a blank experiment, which looks similar to the following image.

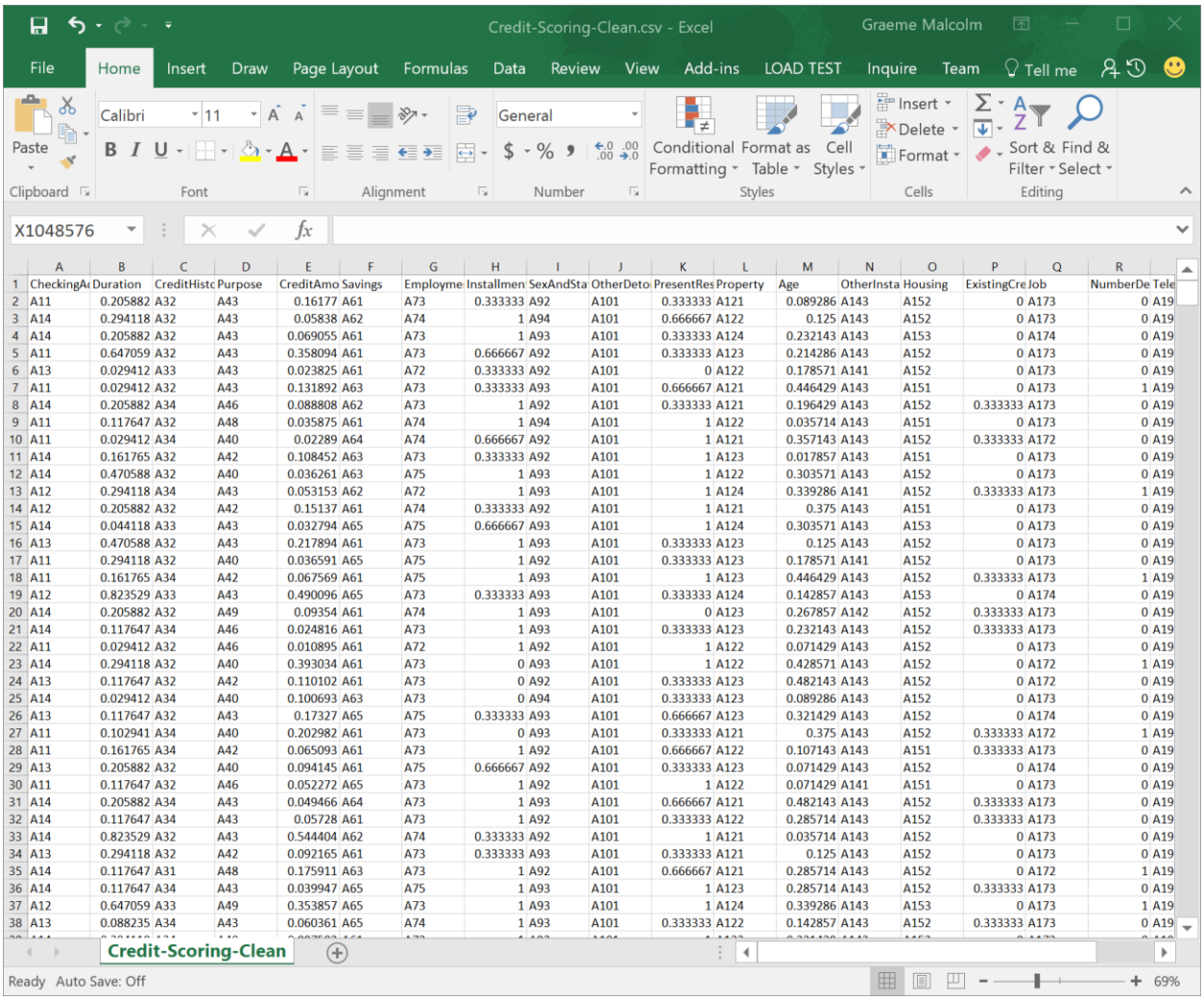


2. Change the title of your experiment from "Experiment created on *today's date*" to "**Bank Credit**"

Upload and Visualize the Dataset

> **Note**: The data set you will use in this lab has been cleaned and adjusted to make life easy for you while performing this lab. Later in this course, you will learn the important techniques required to clean and adjust data sets, and prepare them for analysis.

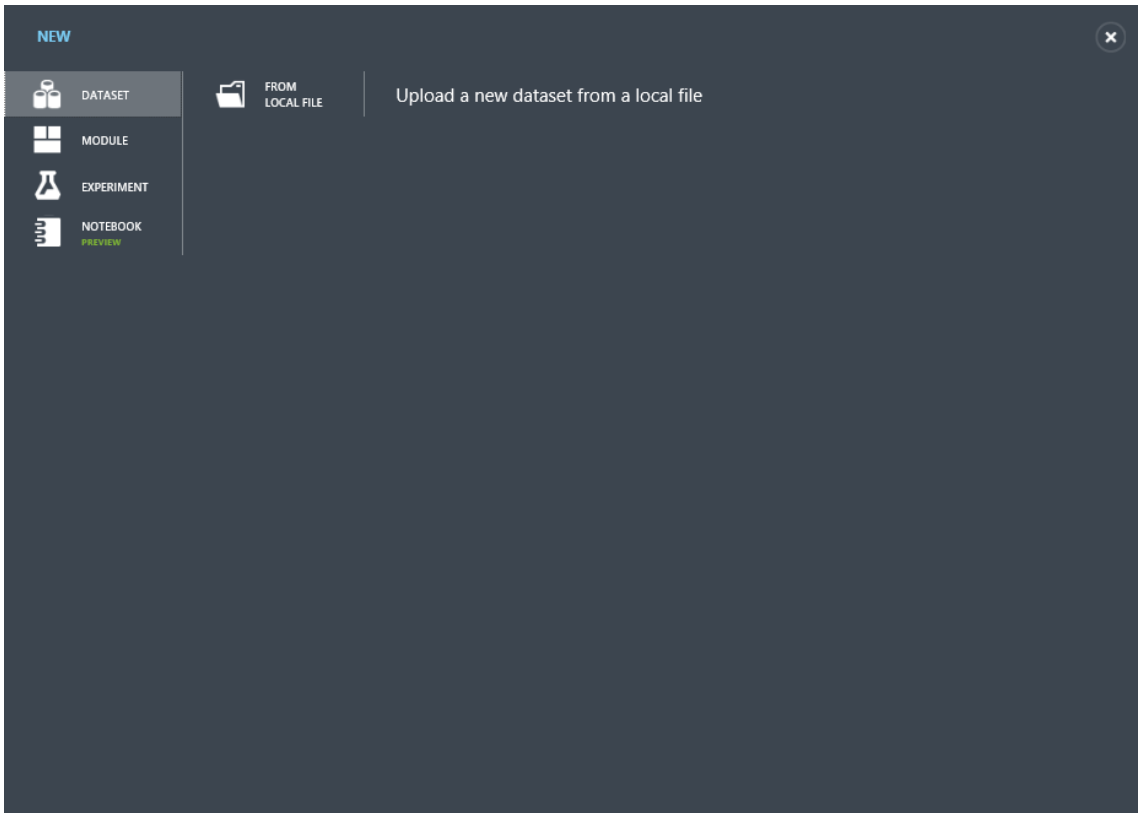
1. From the folder where you extracted the lab files for this module (for example, C:\DAT203.1x\Mod1), open the **Credit-Scoring-Clean.csv** file, using either a spreadsheet application such as Microsoft Excel, or a text editor such as Microsoft Windows Notepad.
2. View the contents of the **Credit-Scoring-Clean.csv** file, noting that it contains data on 950 customer cases. You can see the column headers for 20 features (data columns which can be used to train a machine learning model) and the label (the column indicating the actual credit status of the customers). Your data file should appear as shown here:



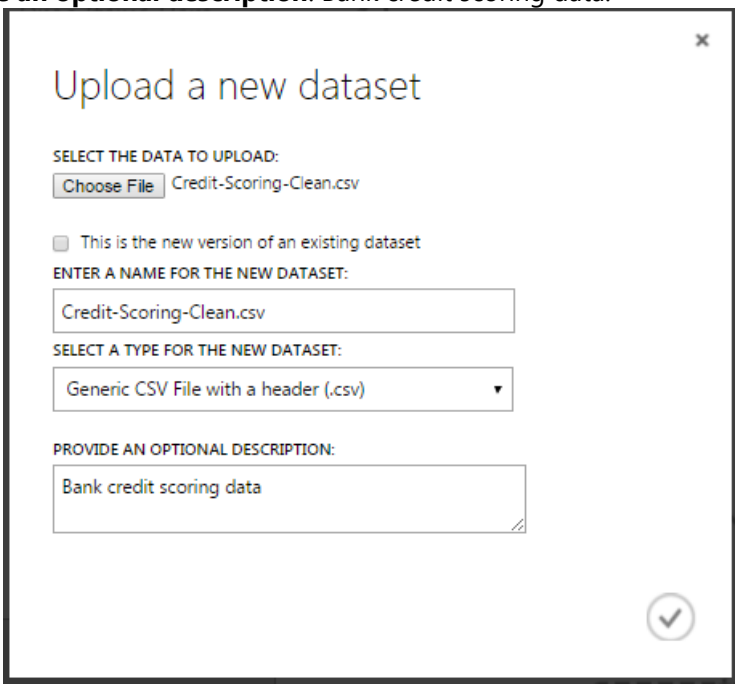
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	CheckingAc	Duration	CreditHistc	Purpose	CreditAmo	Savings	Employme	Installmen	SexAndSta	OtherDeto	PresentRes	Property	Age	OtherInsta	Housing	ExistingCre	Job	NumberDe	Tele
2	A11	0.205882	A32	A43	0.16177	A61	A73	0.333333	A92	A101	0.333333	A121	0.089286	A143	A152	0	A173	0	A19
3	A14	0.294118	A32	A43	0.05838	A62	A74	1	A94	A101	0.666667	A122	0.125	A143	A152	0	A173	0	A19
4	A14	0.205882	A32	A43	0.069055	A61	A73	1	A93	A101	0.333333	A124	0.232143	A143	A153	0	A174	0	A19
5	A11	0.647059	A32	A43	0.358094	A61	A73	0.666667	A92	A101	0.333333	A123	0.214286	A143	A152	0	A173	0	A19
6	A13	0.029412	A33	A43	0.023825	A61	A72	0.333333	A92	A101	0	A122	0.178571	A141	A152	0	A173	0	A19
7	A11	0.029412	A32	A43	0.131892	A63	A73	0.333333	A93	A101	0.666667	A121	0.446429	A143	A151	0	A173	1	A19
8	A14	0.205882	A34	A46	0.088808	A62	A73	1	A92	A101	0.333333	A121	0.196429	A143	A152	0.333333	A173	0	A19
9	A11	0.117647	A32	A48	0.035875	A61	A74	1	A94	A101	1	A122	0.035714	A143	A151	0	A173	0	A19
10	A11	0.029412	A34	A40	0.02289	A64	A74	0.666667	A92	A101	1	A121	0.357143	A143	A152	0.333333	A172	0	A19
11	A14	0.161765	A32	A42	0.108452	A63	A73	0.333333	A92	A101	1	A123	0.017857	A143	A151	0	A173	0	A19
12	A14	0.470588	A32	A40	0.036261	A63	A75	1	A93	A101	1	A122	0.303571	A143	A152	0	A173	0	A19
13	A12	0.294118	A34	A43	0.053153	A62	A72	1	A93	A101	1	A124	0.339286	A141	A152	0.333333	A173	1	A19
14	A12	0.205882	A32	A42	0.15137	A61	A74	0.333333	A92	A101	1	A121	0.375	A143	A151	0	A173	0	A19
15	A14	0.044118	A33	A43	0.032794	A65	A75	0.666667	A93	A101	1	A124	0.303571	A143	A153	0	A173	0	A19
16	A13	0.470588	A32	A43	0.217894	A61	A73	1	A93	A101	0.333333	A123	0.125	A143	A152	0	A173	0	A19
17	A11	0.294118	A32	A40	0.036591	A65	A75	1	A92	A101	0.333333	A123	0.178571	A141	A152	0	A173	0	A19
18	A11	0.161765	A34	A42	0.067569	A61	A75	1	A93	A101	1	A123	0.446429	A143	A152	0.333333	A173	1	A19
19	A12	0.823529	A33	A43	0.490096	A65	A73	0.333333	A93	A101	0.333333	A124	0.142857	A143	A153	0	A174	0	A19
20	A14	0.205882	A32	A49	0.09354	A61	A74	1	A93	A101	0	A123	0.267857	A142	A152	0.333333	A173	0	A19
21	A14	0.117647	A34	A46	0.024816	A61	A73	1	A93	A101	0.333333	A123	0.232143	A143	A152	0.333333	A173	0	A19
22	A11	0.029412	A32	A46	0.010895	A61	A72	1	A92	A101	1	A122	0.071429	A143	A152	0	A173	0	A19
23	A14	0.294118	A32	A40	0.393034	A61	A73	0	A93	A101	1	A122	0.428571	A143	A152	0	A172	1	A19
24	A13	0.117647	A32	A42	0.110102	A61	A73	0	A92	A101	0.333333	A123	0.482143	A143	A152	0	A172	0	A19
25	A14	0.029412	A34	A40	0.100693	A63	A73	0	A94	A101	0.333333	A123	0.089286	A143	A152	0	A173	0	A19
26	A13	0.117647	A32	A43	0.17327	A65	A75	0.333333	A93	A101	0.666667	A123	0.321429	A143	A152	0	A174	0	A19
27	A11	0.102941	A34	A40	0.202982	A61	A73	0	A93	A101	0.333333	A121	0.375	A143	A152	0.333333	A172	1	A19
28	A11	0.161765	A34	A42	0.065093	A61	A73	1	A92	A101	0.666667	A122	0.107143	A143	A151	0.333333	A173	0	A19
29	A13	0.205882	A32	A40	0.094145	A61	A75	0.666667	A92	A101	0.333333	A123	0.071429	A143	A152	0	A174	0	A19
30	A11	0.117647	A32	A46	0.052272	A65	A73	1	A92	A101	1	A122	0.071429	A141	A151	0	A173	0	A19
31	A14	0.205882	A34	A43	0.049466	A64	A73	1	A93	A101	0.666667	A121	0.482143	A143	A152	0.333333	A173	0	A19
32	A14	0.117647	A34	A43	0.05728	A61	A73	1	A92	A101	0.333333	A122	0.285714	A143	A152	0.333333	A173	0	A19
33	A14	0.823529	A32	A43	0.544404	A62	A74	0.333333	A92	A101	1	A121	0.035714	A143	A152	0	A173	0	A19
34	A13	0.294118	A32	A42	0.092165	A61	A73	0.333333	A93	A101	0.333333	A121	0.125	A143	A152	0	A173	0	A19
35	A14	0.117647	A31	A48	0.175911	A63	A73	1	A92	A101	0.666667	A121	0.285714	A143	A152	0	A172	1	A19
36	A14	0.117647	A34	A43	0.039947	A65	A75	1	A93	A101	1	A123	0.285714	A143	A152	0.333333	A173	0	A19
37	A12	0.647059	A33	A49	0.353857	A65	A73	1	A93	A101	1	A124	0.339286	A143	A153	0	A173	1	A19
38	A13	0.088235	A34	A43	0.060361	A65	A74	1	A93	A101	0.333333	A122	0.142857	A143	A152	0.333333	A173	0	A19

Note: the information in some of these features (columns) is in a coded format; e.g. A14, A11. You can see the meaning of these codes on the UCI Machine Learning repository at https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data).

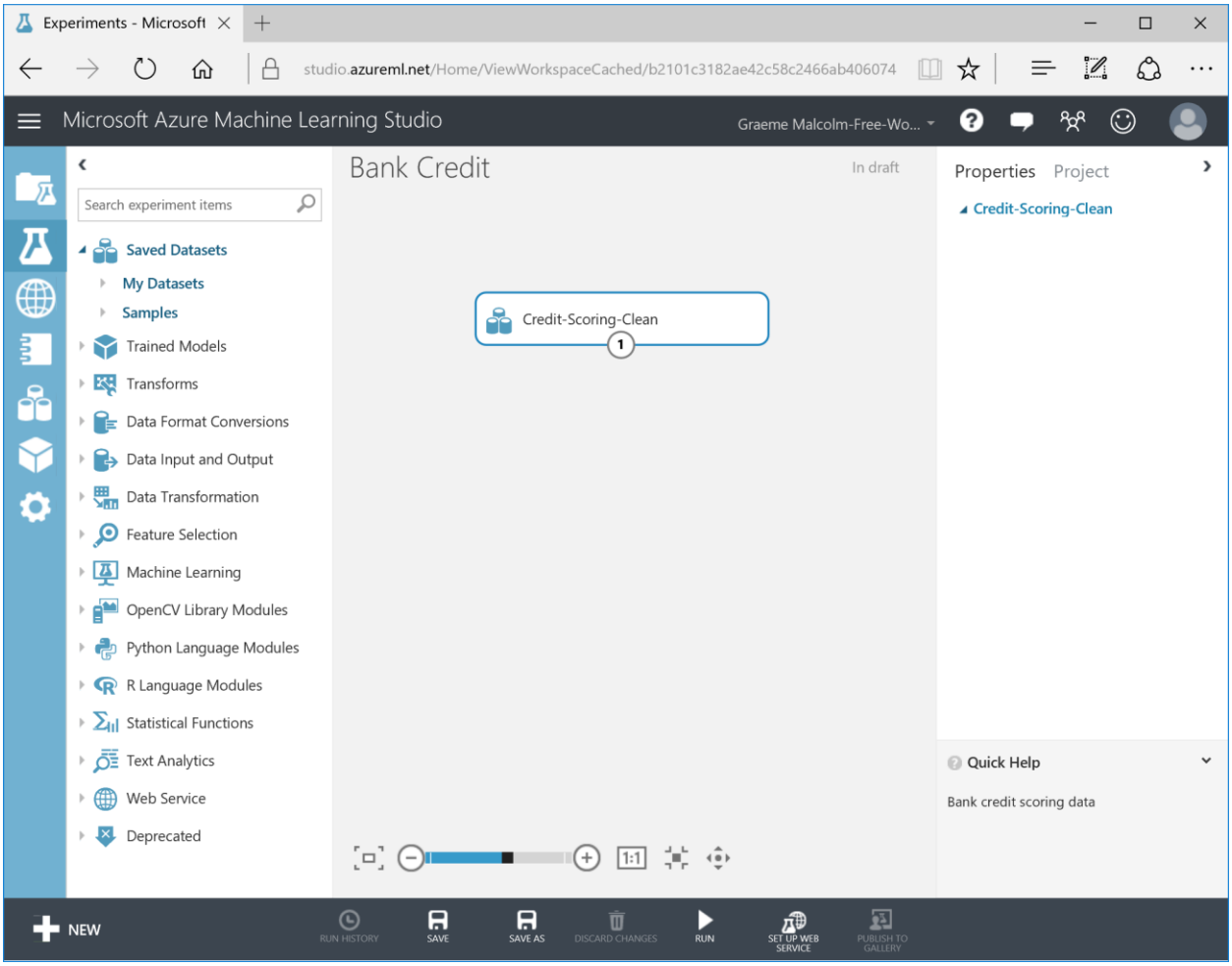
3. Close the text file and return to your browser where your experiment is displayed. At the bottom left, click **NEW**. Then in the **NEW** dialog box, click the **DATASET** tab as shown in the following image.



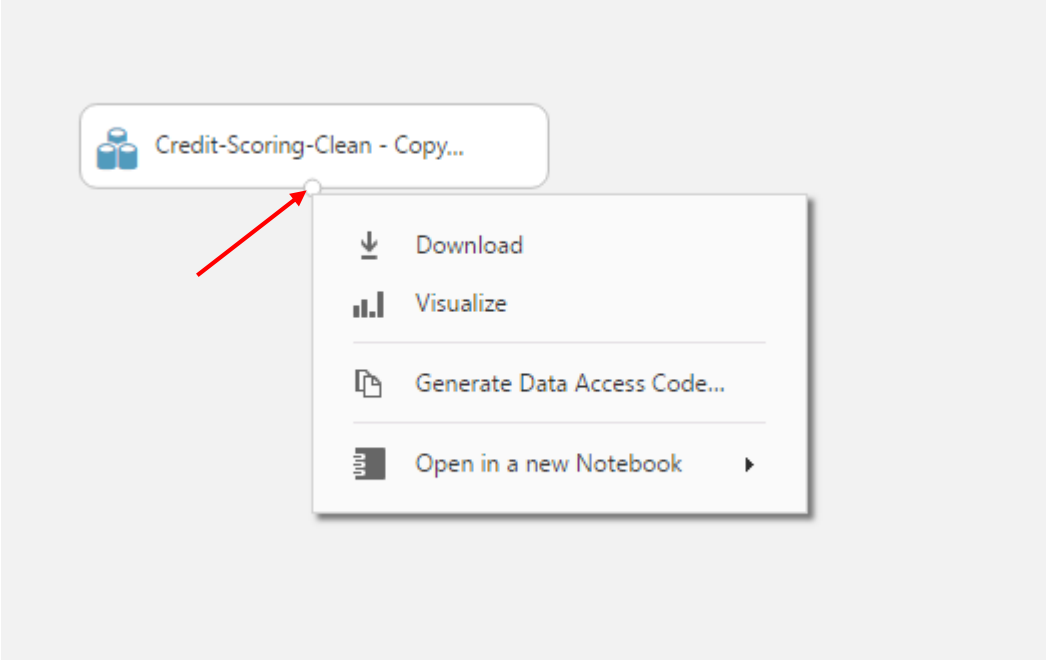
4. Click **FROM LOCAL FILE**. Then in the **Upload a new dataset** dialog box, browse to select the **Credit-Scoring-Clean.csv** file from the folder where you extracted the lab files on your local computer. Enter the following details as shown in the image below, and then click the ✓icon.
    - **This is a new version of an existing dataset**: Unselected
    - **Enter a name for the new dataset**: Credit-Scoring-Clean
    - **Select a type for the new dataset**: Generic CSV file with a header (.csv)
    - **Provide an optional description**: Bank credit scoring data.



5. Wait for the upload of the dataset to complete, then click **OK** on the status bar at the bottom of the AML Studio screen.
6. On the experiment items pane, expand **Saved Datasets** > **My Datasets** to verify that the **Credit-Scoring-Clean** dataset is listed.
7. Drag the **Credit-Scoring-Clean** dataset to the canvas for the **Bank Credit** experiment.
8. Verify that the Azure ML screen, which shows your experiment, now looks like the figure shown here:

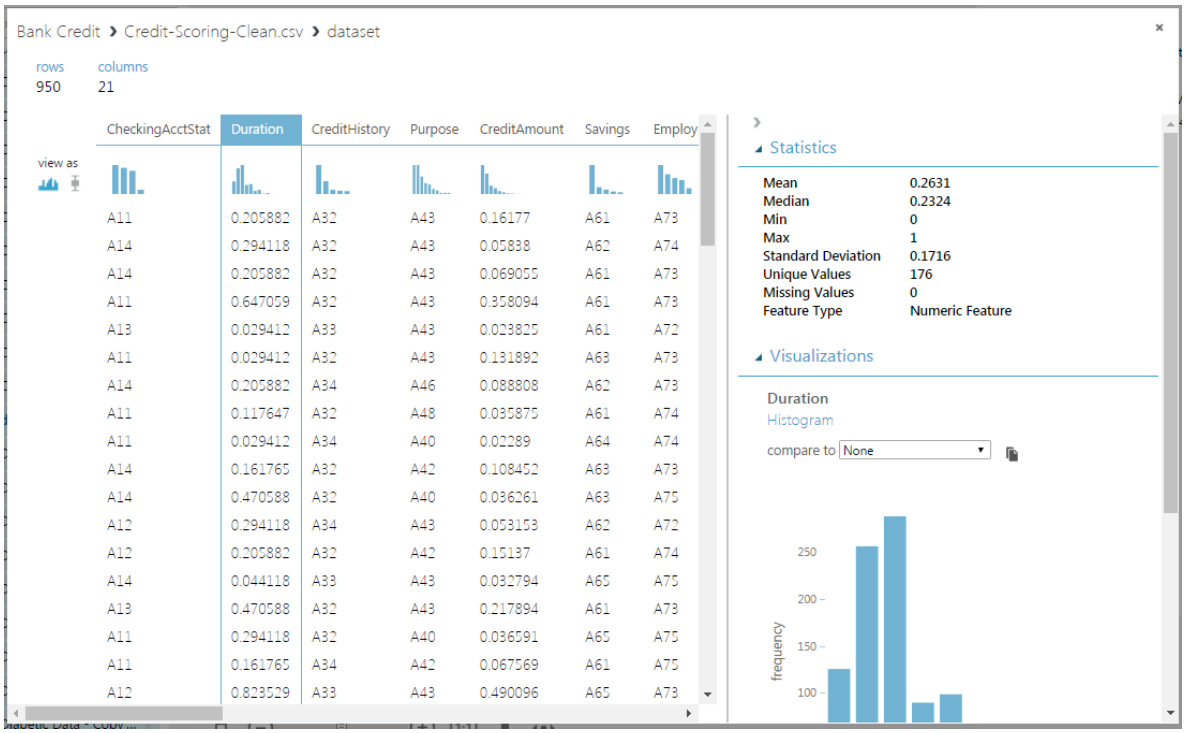


9. Click the output port for the **Credit-Scoring-Clean** dataset on the canvas and click **Visualize** to view the data in the dataset as shown in the figure:



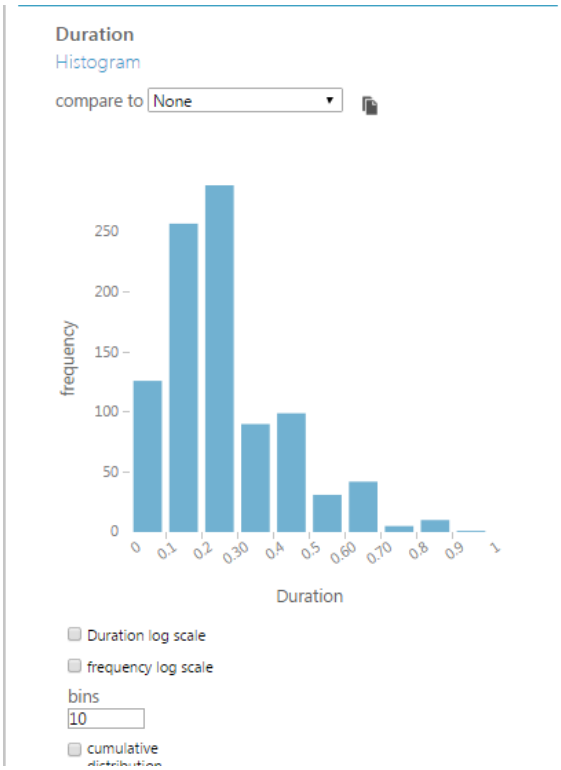
Note: The output port can be accessed by clicking on the small circle on the bottom of the module boxes, pointed to by the red arrow in the figure.

10. Click on the second column labeled **Duration**, which will display some properties of that feature (data column) on the right side of the display. These properties include summary statistics and the data type, as shown here:

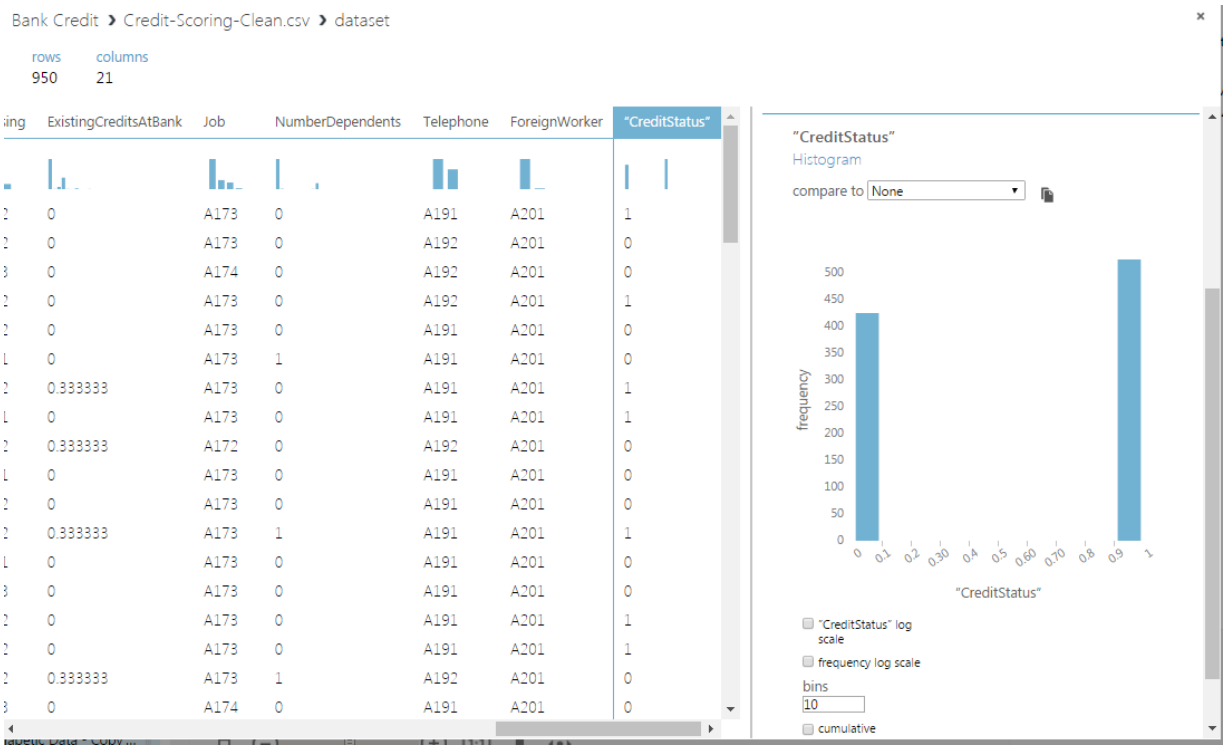


11. Verify that the dataset contains the data you viewed in the source file.

12. Using the scroll bar on the right side of the display, scroll down until you can see the histogram of the **Duration** feature as shown here:



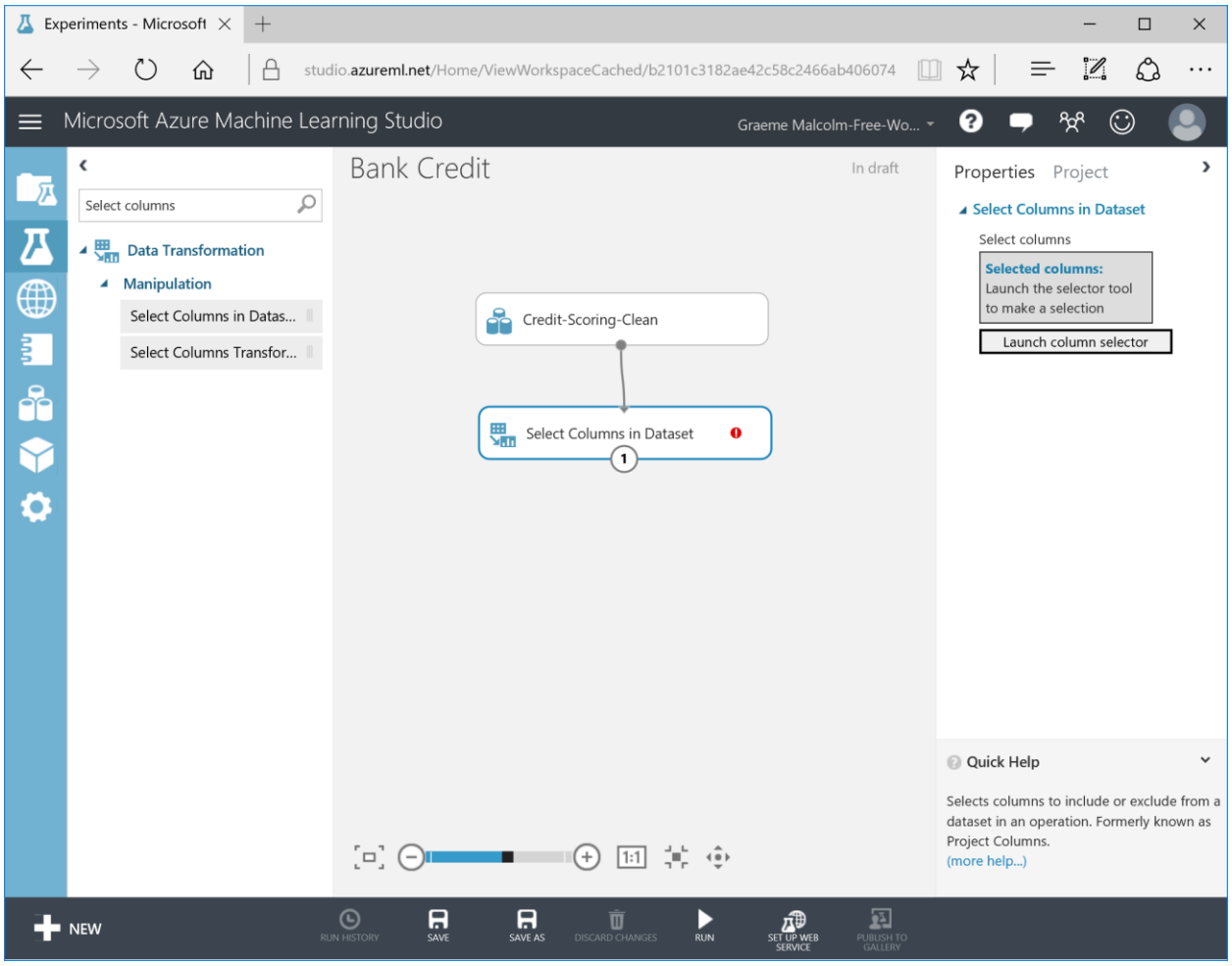
13. On the data display, scroll to the right and click **CreditStatus**. Scroll down in the pane on the right and observe the histogram, which should appear as shown below. Note that **CreditStatus** has two values, {0,1}, and that the number of cases with each value are approximately balanced.



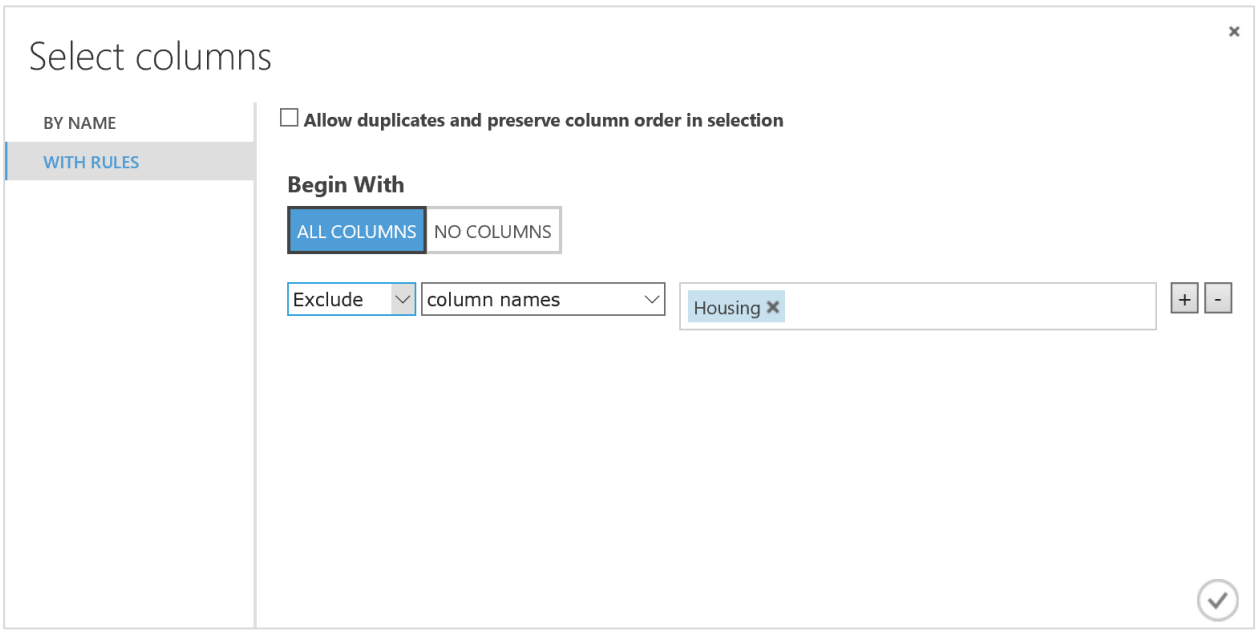
14. Click the '**x**' in the upper right corner to close the visualization.

Select Columns from the Dataset

1. In Azure ML Studio, search for the **Select Columns in Dataset** module, which is in the **Manipulation** category under **Data Transformation**, and drag it onto the canvas.
2. Connect the output of the **Credit-Scoring-Clean** dataset to the **Dataset** input of the **Select Columns in Dataset** module as shown here:



3. With the **Select Columns in Dataset** module selected, in the **Properties** pane, click **Launch Column** selector.
4. In the **Select columns** dialog box, note that on the **By Name** page, you can select individual columns by name; or alternatively, on the **With Rules** page you can specify rules to filter the columns. Many of the modules in Azure ML use this column selector, so you should familiarize yourself with it.
5. On the **With Rules** page, create a rule that starts with all columns and then excludes the **Housing** columns as shown here; then click the ✓ icon to apply the filter.



6. **Save** and **Run** your experiment by clicking on the icons at the bottom of the studio.
7. When your experiment has finished running, visualize the **Results Dataset** output of the **Select Columns in Dataset** module. Note that there are now 20 columns, because the **Housing** column has been removed.

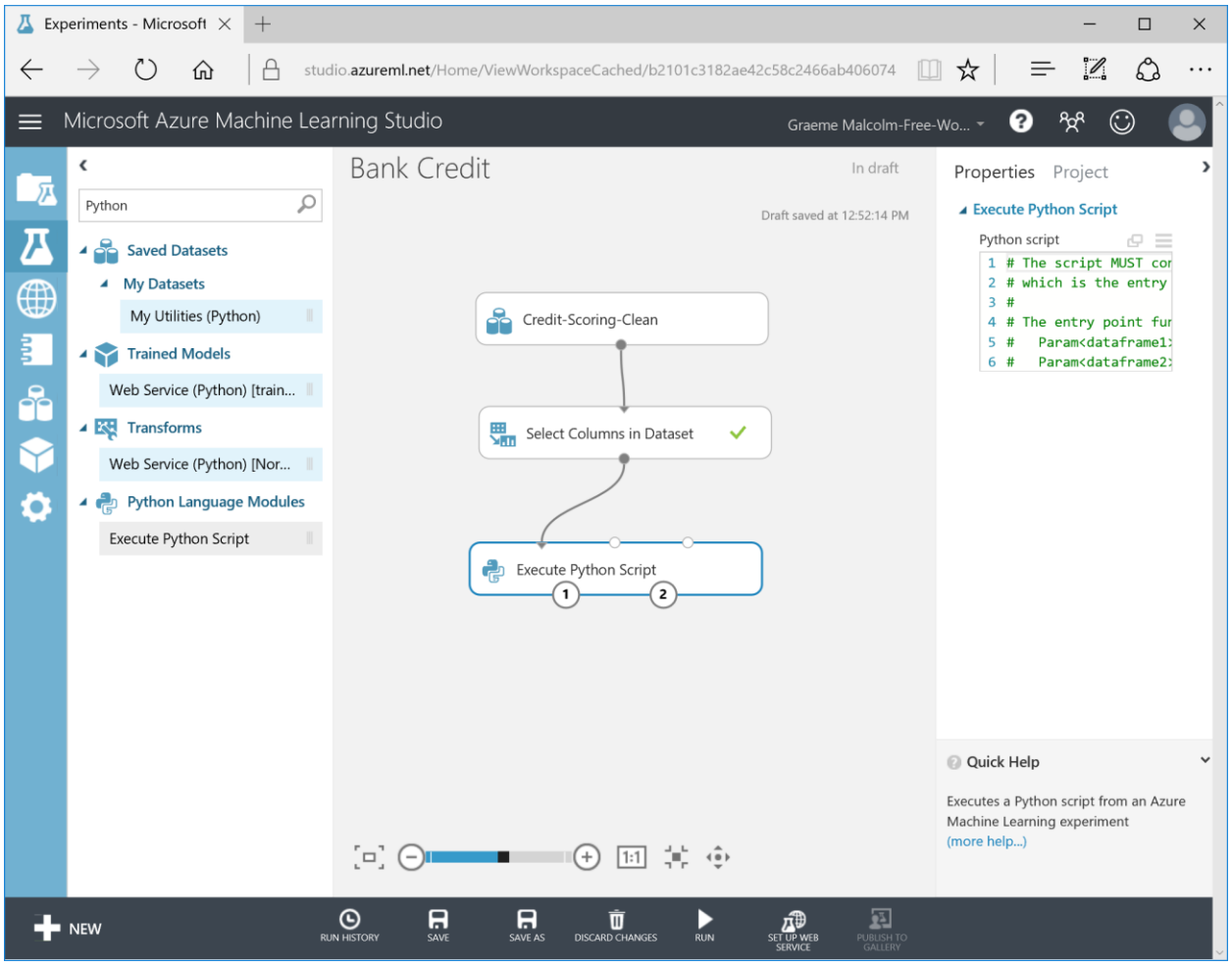
Using Custom Code in Azure ML

In this exercise you will use the Python, R and SQL to filter more columns from the **Credit-Scoring-Clean** data set.

Add an Execute Python Script Module

1. In Azure ML Studio, search for the **Execute Python Script** module, which is under **Python Language Modules**, and drag it onto the canvas.
   Note: Python is a commonly used scripting language in data science experiments; and like R, it enables you to include custom logic in an Azure ML experiment. You'll learn more about using Python in data science experiments in later modules. For now, use a simple Python script to remove some columns from the dataset. Throughout the rest of this course, you'll have the opportunity to choose either R or Python for scripting tasks.

2. Connect the **Results Dataset** output of the **Select Columns in Dataset** module to the **Dataset1** (left most) input of the **Execute Python Script** module as shown here:



3. Select the **Execute Python Script** module, and then replace the existing code in the code editor pane with the following code, which drops the **SexAndStatus** and **OtherDetorsGuarantors** columns. You can copy and paste this code from **dropcols.py** in the lab files folder for this lab.

```
def azureml_main(creditframe):
    drop_cols = ['SexAndStatus',
        'OtherDetorsGuarantors']
    creditframe.drop(drop_cols, axis = 1, inplace = True)

    return creditframe
```

> **Tip**: To paste code from the clipboard 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, to replace the existing code.

4. **Save** and **Run** the experiment, and when it has finished running, visualize the **Results Dataset** (left hand) output of the **Execute Python Script** module. Note that there are now 18 columns, as another two have been removed.

Add an Execute R Script Module

1. Search for the **Execute R Script** module, which is under the **R Language Modules**, and drag it onto the canvas.

Note: R is a commonly used scripting language in data science experiments, and it enables you to include custom logic in an Azure ML experiment. You'll learn more about using R in data science experiments in later modules. For now, use a simple R script to remove some more columns from the dataset. Throughout the rest of this course you'll have the opportunity to choose either R or Python for scripting tasks.

2. Connect the **Results Dataset1** (left) output of the **Execute Python Script** module to the **Dataset1** (left most) input of the **Execute R Script** module.
3. Replace the existing R code in the code editor window of the **Execute R Script** module with the following code. You can copy and paste this code from **dropcols.R** in the lab files folder for this lab.

```
credit.frame <- maml.mapInputPort(1)
drop.cols <- c('OtherInstallments',
               'ExistingCreditsAtBank')
out.frame <- credit.frame[, !(names(credit.frame) %in% drop.cols)]
maml.mapOutputPort("out.frame")
```

4. **Save** and **Run** the experiment. Then, when it has finished running, visualize the **Results Dataset** (left hand) output of the **Execute R Script** module. Note that there are now 17 columns.

Add an Apply SQL Transform

1. Search for the **Apply SQL Transform** module, under **Data Transformation > Manipulation,** and drag it onto the canvas.

   Note: The **Apply SQL Transformation** module enables you to write custom log in SQLite, a variant of the ANSI SQL language. If you are familiar with Transact-SQL in Microsoft databases such as SQL Server and Azure SQL Database, apply your SQL knowledge to work with data in an Azure ML experiment.

2. Connect the **Results Dataset** (left) output of the **Execute R Script** module to the **Table1** (left most) input of the **Apply SQL Transform** module.
3. Replace the existing SQL code in the code editor window of the **Apply SQL Transform** module with the following code. You can copy and paste this code from **selectcols.sql** in the lab files folder for this lab.

```
select
  CheckingAcctStat,
  Duration,
  CreditHistory,
  Purpose,
  Savings,
  Employment,
  InstallmentRatePecnt,
  PresentResidenceTime,
  Property,
  Age,
  Telephone,
  CreditStatus
  from t1;
```

4. **Save** and **Run** the experiment. Then, when it has finished running, visualize the **Results Dataset** output of the **Apply SQL Transform** module. Note it contains only the 12 columns named in the SQL select statement.

Creating and Evaluating a Machine Learning Model

Now that you have created a simple experiment that processes data, you can use the data to train a predictive model. In this exercise, you will use the data to create a model that tries to predict if a particular bank customer is a good or bad credit risk.

Note: The purpose of these exercises is to give you a feel for working with machine learning models in Azure Machine Learning. In subsequent chapters and in the next course we will explore the theory of operation and evaluation for machine learning models.

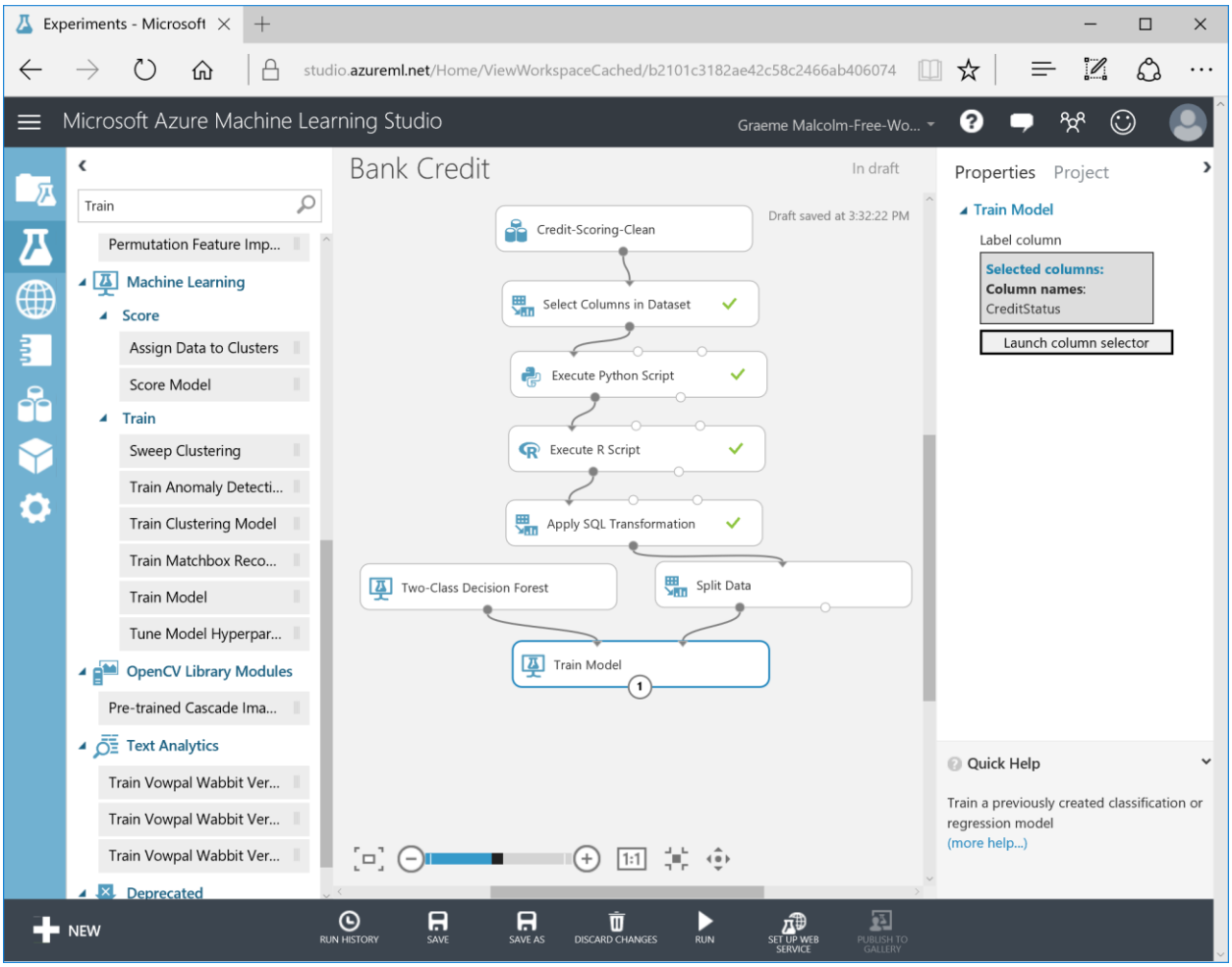
Split the Data

1. Search for the **Split Data** module and drag it onto the canvas under the existing modules.
   Note: The data are split to create independent, non-overlapping, randomly sampled subsets of the data to train and evaluate the performance of the machine learning model.

2. Connect the output of the **Apply SQL Transformation** module to the input of the **Split Data** module.
3. Select the **Split Data** module, and in the **Properties** pane, view the default split settings, which split the data randomly into two datasets. Set these properties as follows:
   - **Splitting mode:** Split Rows
   - **Fraction of rows in the first output dataset:** 0.7
   - **Randomized split:** checked
   - **Random seed:** 876
   - **Stratified split:** False

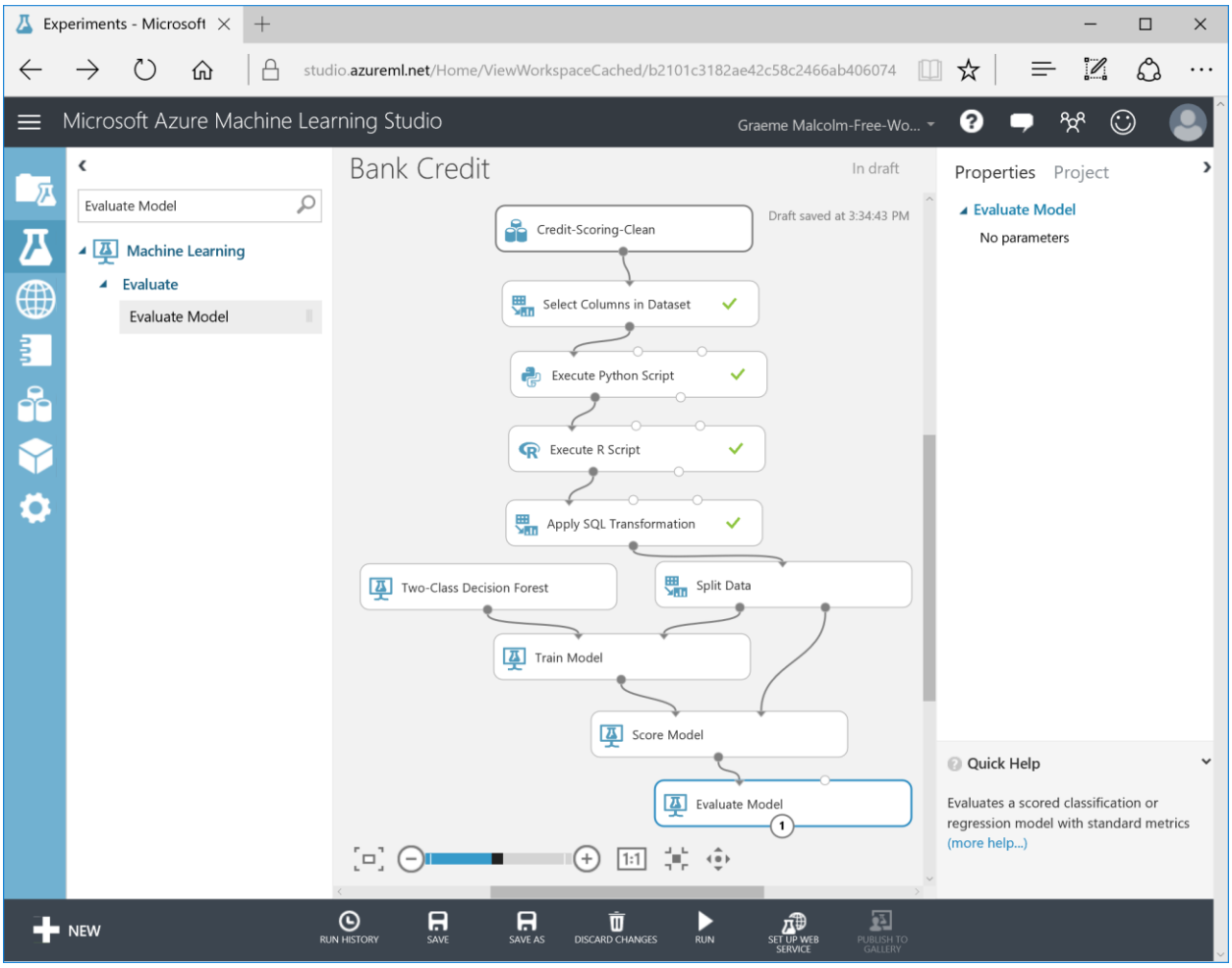
Add an Algorithm and a Train Model Module

1. In the experiment items pane, search for "Classification", and view the range of multi-class and two-class models that are supported in Azure ML. Find the **Two Class Decision Forest** module, which is under **Machine Learning > Initialize Model > Classification,** and drag it onto the canvas to the left of the **Split Data** module.
2. Select the **Two Class Decision Forest** module, and in the **Properties** pane configure the model parameters as follows:
   - **Resampling method:** Bagging
   - **Create trainer mode:** Single Parameter
     Number of Decision trees: 50
     Maximum depth of the decision tree: 32
   - **Number of random splits per node:** 32
   - **Minimum number of samples per leaf node:** 4
   - **Allow unknown values for categorical features:** checked
3. Search for the **Train Model** module, which is under **Machine Learning > Train**, and drag it onto the canvas beneath the existing modules.
4. Connect the output from the **Two-Class Decision Forest** module to the **Untrained model** (left) input of the **Train Model** module, and connect the **Results dataset1** (left-most) output port of the **Split Data** module to the **Dataset** (right) input of the **Train Model** module.
5. On the properties pane for the **Train Model** module, use the column selector to include only the **CreditStatus** column as the label for the model.
6. Verify that you experiment now looks like this:



Add Modules to Score and Evaluate the Trained Model

1. Search for the **Score Model** module, which is under **Machine Learning > Score**, and drag it onto the canvas under the existing modules.
2. Connect the output of the **Train Model** module to the **Trained model** (left) input of the **Score Model** module.
3. Connect the **Results Dataset2** (right) output of the **Split Data** module to the **Dataset** (right) input of the **Score Model** module.
4. Search for the **Evaluate Model** module, which is under **Machine Learning > Evaluate**, and drag it onto the canvas under the existing modules.
5. Connect the output of the **Score Model** module to the **Scored dataset** (left) input of the **Evaluate Model** module.
6. Verify that your experiment now looks like this:



Train and Evaluate the Model

1. **Save** and **Run** your experiment.
2. When the experiment has finished running, visualize the output of the **Score Model** module. Note the values of the **CreditStatus** column (the known label in the test dataset) and **Scored Labels** column (the prediction computed by the model). In most cases, the values in these columns are identical, indicating that the model has correctly predicted the label value. Cases where the value of the label and the prediction differ are errors.
3. Visualize the output of the **Evaluate Model** module. Scroll down until you see performance metrics, including values for **True Positive**, **False Negative**, **False Positive**, **True Negative**, **Accuracy**, **Precision**, **Recall**, **F1 Score**, and **AUC**. These metrics are used to measure the effectiveness of the model, and will be discussed later in this course.

Summary

This lab has familiarized you with the essentials of using the Azure ML Studio environment. In this lab you have used built-in Azure ML functionality, Python, R and SQL to select the features used for training a machine learning model. You then created, trained, and evaluated a first machine learning model to classify bank customers as good or bad credit risks.

In the rest of this course, you will learn how to employ a range of techniques to prepare data for modeling, to build effective models, and to evaluate model performance to create a suitably accurate predictive solution.