


# Deploying a Predictive Model with Azure Machine Learning



# Contents

Overview .....	3
Copy an Experiment from the Gallery .....	5
Explore the Training Experiment.....	11
Publish a Model as a Web Service.....	15
Make Predictions using the Web Service .....	18
Exercises.....	20
Terms of Use .....	21

# Overview

## Summary

This lab explores unsupervised learning in Azure Machine Learning and how to deploy a predictive model as a web service.

The lab will walk through copying an experiment from the Azure Machine Learning Gallery into the ML Studio, creating a scoring experiment, deploying a model as a web service, and interacting with the API using the included web interface.

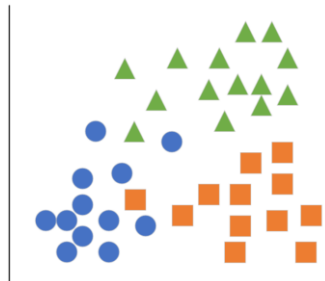
## Business Case

“Where should I open my next restaurant location?”

This question is often very difficult to answer. The right choice could lead to increased revenue and profit, but the wrong choice could lead to losing a major investment. Trying to make this decision by manually sifting through hundreds or even thousands of possible cities or neighborhoods can be almost impossible. Machine learning can help with this task by analyzing large volumes of data about different locations, finding common characteristics among locations, and grouping those like-attributed locations together. These groups can then be compared to previously successful restaurant locations to help narrow the choices for where to open next.

In this lab, you will work with a dataset that includes geographic, economic, and demographic data about different US cities.

The model you will explore uses a K-Means algorithm to cluster cities into distinctive buckets. K-Means Clustering is an unsupervised learning approach that uses iterative techniques to analyze cases and then group cases with similar characteristics together. These groups can then be used for labeling, exploration, identifying anomalies, or even further prediction.



## Learning Objectives

Upon completing this lab, you will have hands-on experience with the following functions and concepts related to Azure Machine Learning:

- Copying an experiment from the Azure Machine Learning Gallery to the ML Studio
- Normalizing data
- Creating a Scoring Experiment
- Deploying a model as a web service

- Modifying Web service inputs and outputs
- Testing the web service via the web UI

#### **Lab Requirements/Prerequisites**

- A Microsoft account is required to access an Azure Machine Learning workspace. If you don't already have a Microsoft account, you can obtain one for free by following the link below:

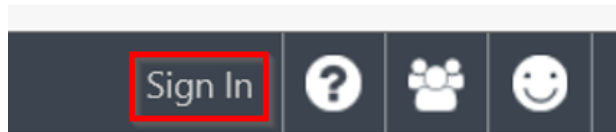
<https://www.microsoft.com/en-us/account/default.aspx>

# Copy an Experiment from the Gallery

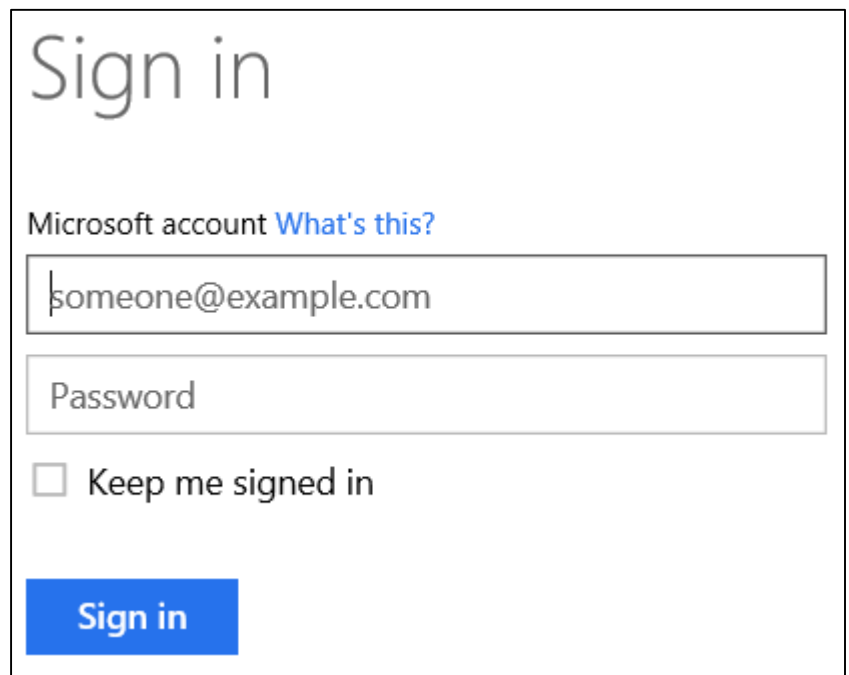
## Access an Azure Machine Learning Workspace

To get started, you will need to log in to a free Azure Machine Learning workspace. A workspace is like an all-inclusive development environment with the tools to create, manage, and publish machine learning models.

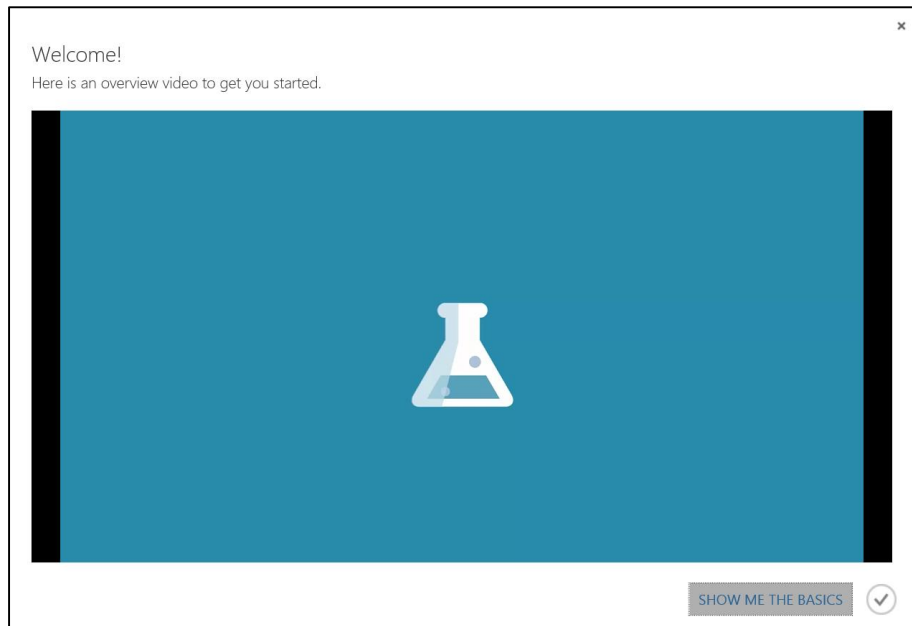
1. Open **Internet Explorer** using the shortcut on the desktop taskbar.
2. Go to the **ML Studio** website by typing <http://studio.azureml.net> in the address bar.
3. Click **Sign In** on the top right corner of the web page.



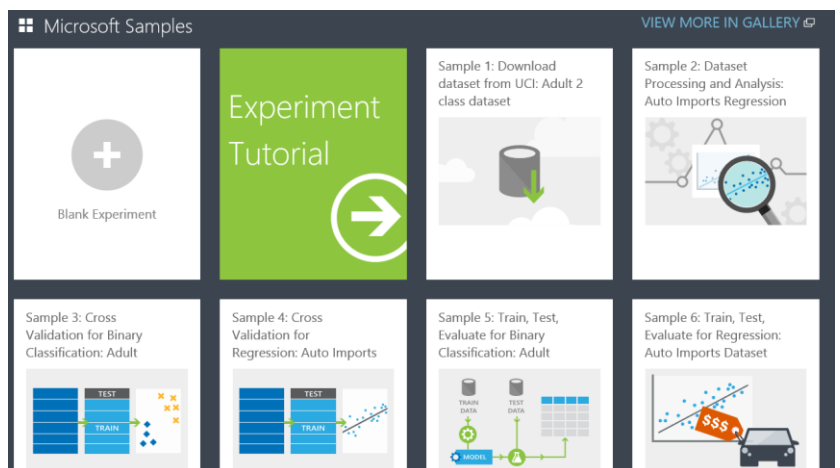
4. Enter the **email address** and **password** associated with your Microsoft ID, and click the **Sign In** button.

A screenshot of the Microsoft Sign in page. The page has a white background. At the top, the text 'Sign in' is displayed in a large, light gray font. Below it, the text 'Microsoft account' is followed by a blue link 'What's this?'. There are two input fields: the first contains the email address 'someone@example.com', and the second is labeled 'Password'. Below the password field is a checkbox labeled 'Keep me signed in'. At the bottom, there is a blue button with the text 'Sign in' in white.

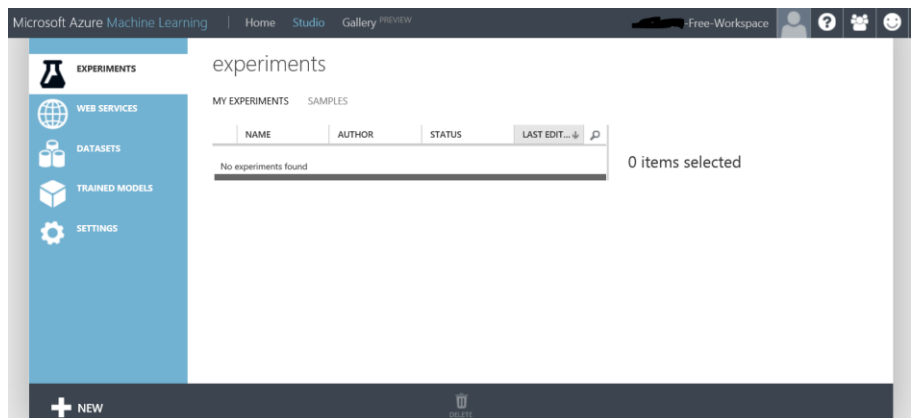
5. If upon logging in, a **Welcome** video is displayed (usually displays on the first login), click the **X** at the top right of the video to close it.



6. If the **Microsoft Samples** dialogue box is displayed (usually displays on the first login), go ahead and close it by choosing the **X** in the top right corner of the pane.



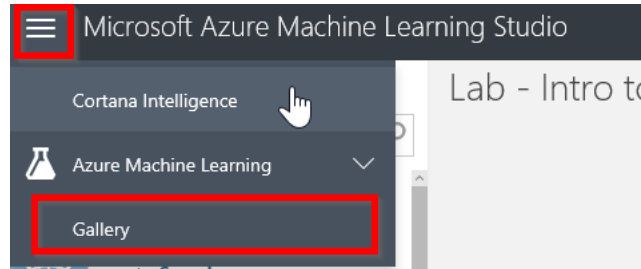
You are now logged into the free workspace associated with your Microsoft ID.



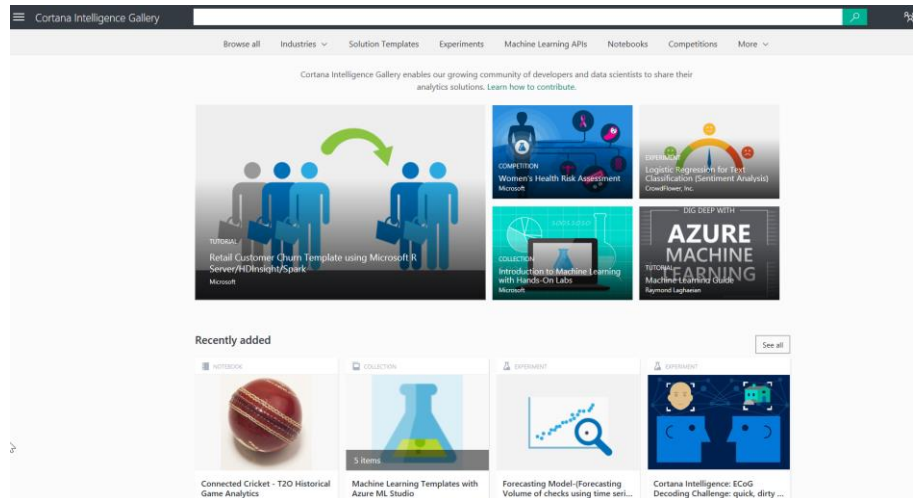
## Copy Experiment from Cortana Intelligence Gallery

Next, you will go to the Cortana Intelligence Gallery to find an experiment that has already been created for you, and copy it into your workspace.

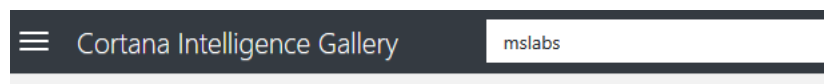
1. Click on the  icon in the top left hand corner of Azure Machine Learning and then click the **Gallery** link.



The *Cortana Intelligence Gallery* website will open. The gallery is Microsoft's community portal for Cortana Intelligence Suite (of which Azure Machine Learning is a component). Here, users can upload experiments and share tutorials and comments with each other.



2. In the search box near the top of the website, type **mslabs** and hit **Enter**.

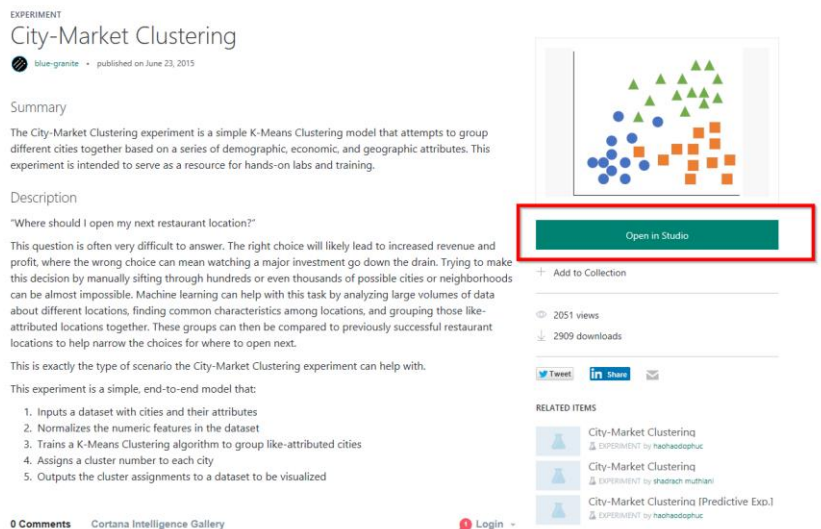


3. Find the experiment titled **City-Market Clustering** and click either the **title** or **picture** on the experiment.



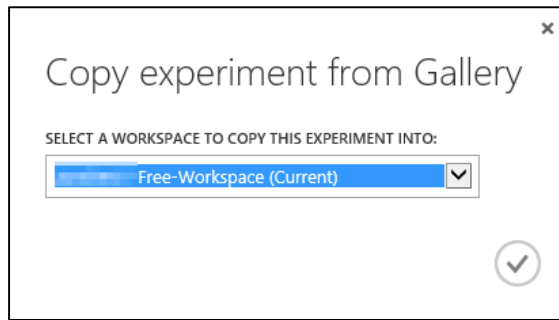
This will open the Experiment page in the gallery. Notice the user who contributed the experiment, a *Summary* and *Description* for the experiment, and some metadata about the experiment in the panel on the right.

4. Click the **OPEN IN STUDIO** button found in the panel on the right side of the screen.



5. In the **Copy experiment from Gallery** dialogue box, choose the **workspace** from the dropdown box you want to copy the experiment into (this should be the same workspace you logged into at the beginning of the lab).





6. Click the  button on the **Copy experiment from Gallery** dialogue box.

If you receive the following message:

This experiment contains modules and/or datasets which have newer versions. Please wait while your experiment is upgraded.

Click OK.

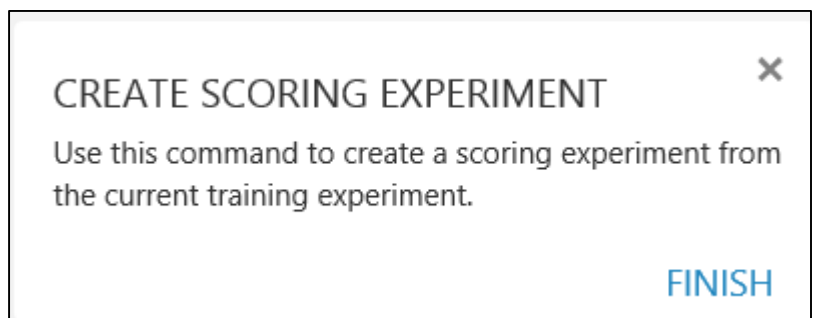
After a few moments, the experiment you selected in the Gallery will open in ML Studio under the workspace you selected. This is a copy of the experiment from the gallery, so you are free to run/modify it as you would like. It will also be saved as a training experiment in the workspace.

7. Click **RUN** at the bottom of the page to execute the training experiment.




After the experiment finishes executing, a *CREATE SCORING EXPERIMENT* dialogue box might be displayed.

8. If the **CREATE SCORING EXPERIMENT** dialogue box is displayed, click the **X** in the top right corner of the **CREATE SCORING EXPERIMENT** dialogue box to close it.

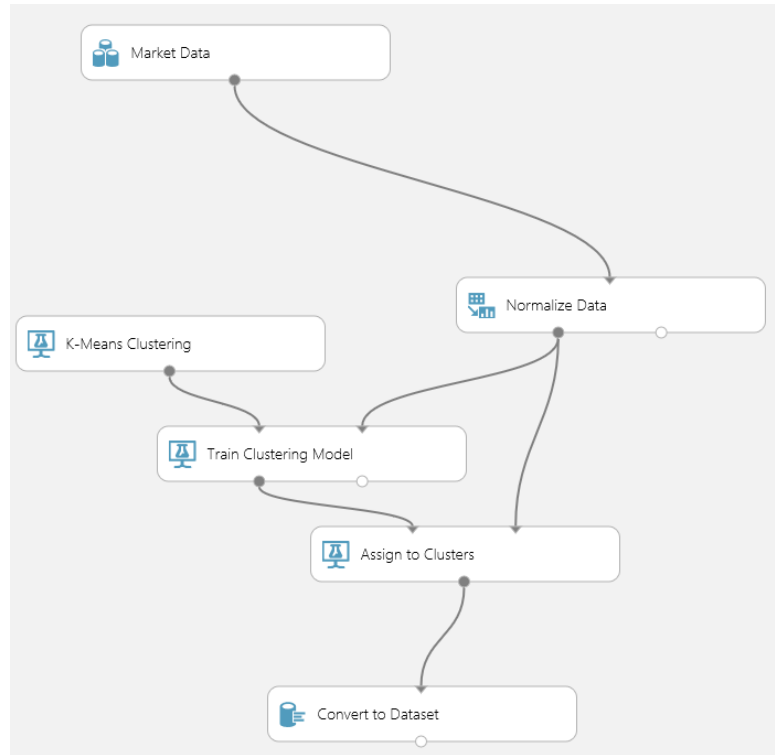


9. Click the **SAVE AS** button at the bottom of the page.



10. In the **Save As** dialogue box, change the **Experiment name** to **Lab – Deploying a Predictive Model**, and click the  button.

The ML Studio canvas should now look similar to the below image.

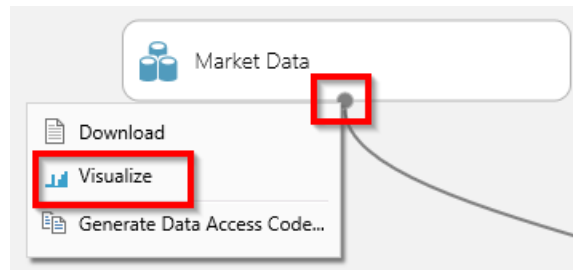


# Explore the Training Experiment

## Visualize the Input Dataset

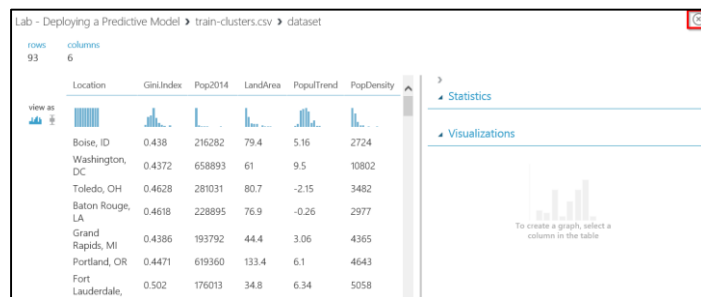
Next, you will take a quick look at the dataset used for the experiment.

1. Click **RUN** at the bottom of the page to execute the training experiment.
2. Once the experiment finishes running, click the **output** port on the **Market Data** dataset and select **Visualize** from the displayed menu

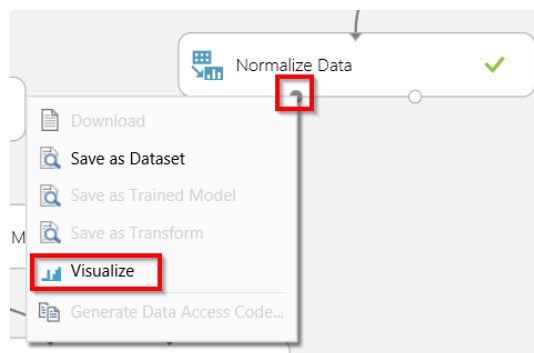


Notice each row in the data set represents a location and there are a series of geographic, economic, and demographic features describing each location.

3. Click the **X** in the top right corner of the **Visualize** dialogue box to close it.



4. Click the **left output** port on the **Normalize Data** module and select **Visualize** from the displayed menu.



Notice each of the features in the dataset has been scaled to a value between -1 and 1.

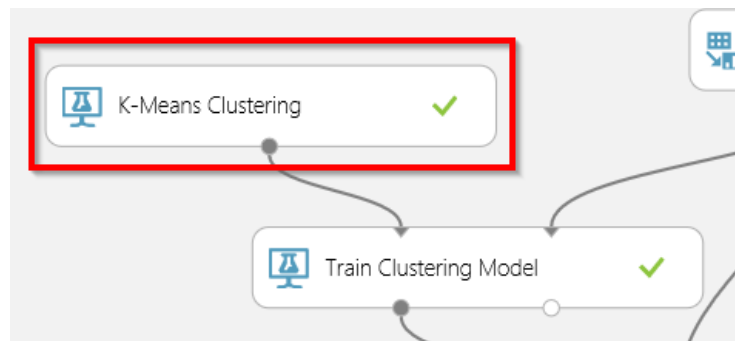
Normalizing features is the process of setting all of the numeric features in a dataset to a similar scale (usually somewhere between -1 and 1). This often prevents features with larger scales from dominating influence in the model and typically makes features easier to visualize. Having normalized data is necessary for K-Means Clustering.

5. Click the **X** at the top right of the **Visualize** dialogue box to close it.

### Explore the Clustering Model

The next series of modules trains the K-Means Clustering model. New input data is then assigned to the clusters using the trained model.

1. Click the **K-Means Clustering** module on the canvas.

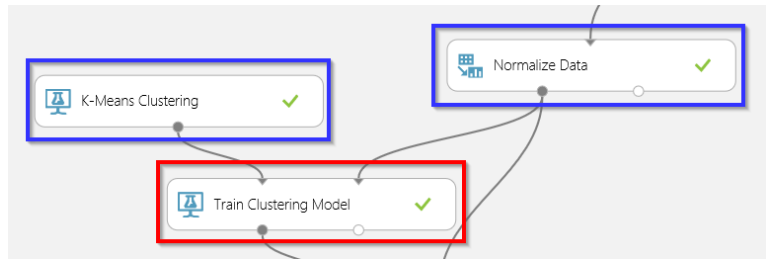


Notice the different parameters that can be configured in the Properties pane, including the number of clusters (centroids) to be created.

The screenshot shows the 'Properties' pane for the 'K-Means Clustering' module. The 'Number of Centroids' is set to 4, which is highlighted with a red rectangle. Other settings include 'Metric' set to 'Euclidean', 'Initialization' set to 'K-Means++', and 'Iterations' set to 100. At the bottom, there are status fields: 'START TIME', 'END TIME', 'ELAPSED TIME', 'STATUS CODE' (Finished), and 'STATUS DETAILS' (None). A link 'View output log' is also present.

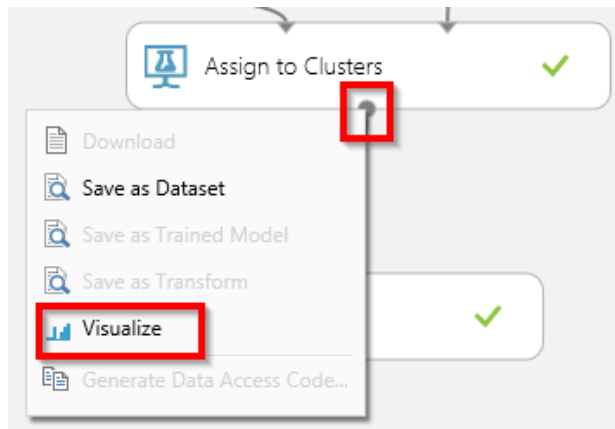
Property	Value
Number of Centroids	4
Metric	Euclidean
Initialization	K-Means++
Iterations	100
START TIME	
END TIME	
ELAPSED TIME	
STATUS CODE	Finished
STATUS DETAILS	None

The next module in the data flow, *Train Clustering Model*, will train the clustering model using the K-Means algorithm and the input dataset.

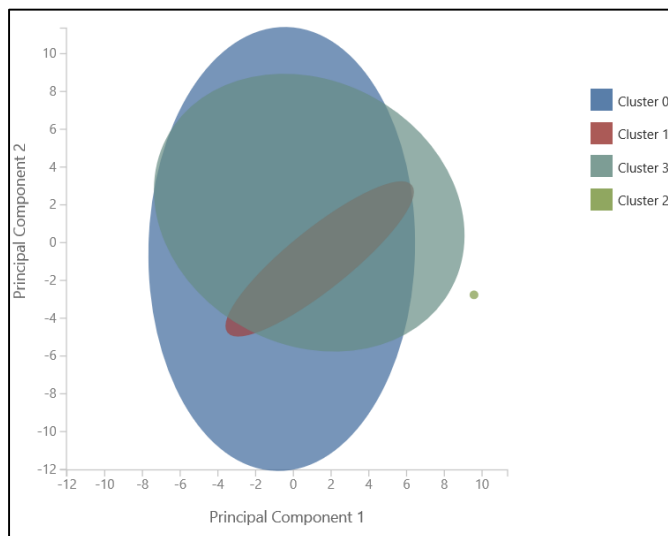


Next, records are assigned to clusters using the *Assign to Clusters* module.

2. Click the **output** port on the **Assign to Clusters** module and select **Visualize** on the displayed menu.

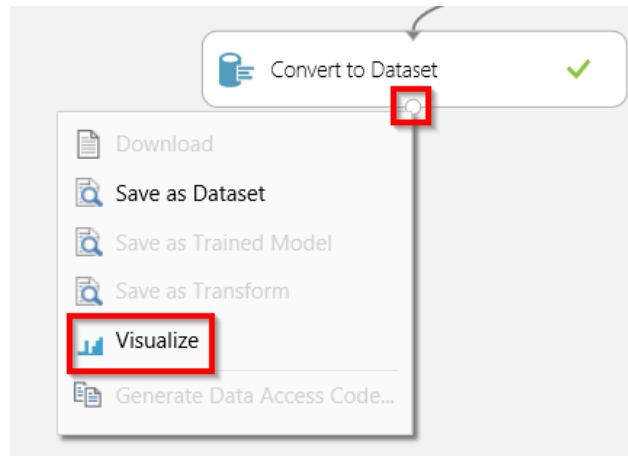


The visual that is displayed shows the size and relative position among features of each of the clusters.



3. Click the **X** in the top right corner of the **Visualize** dialogue box to close it.

- Click the **output** port on the **Convert to Dataset** module and select **Visualize** from the displayed menu to view how each individual city-market was clustered.



The *Assignments* column has the numeric value for the assigned cluster for each row in the data set.

PopulTrend	PopDensity	Assignments
0.056362	-0.067104	0
0.27132	0.227606	0
-0.305698	-0.03945	0

This numeric value does not provide any context into what type of cities/locations are in each cluster.

At this point, you would typically download the assigned dataset to a tool like Excel to analyze and try to label each cluster. Upon doing this, you would likely find clusters for average/established cities, quickly growing cities, extremely large and dense cities, and cities that are very geographically spread out.

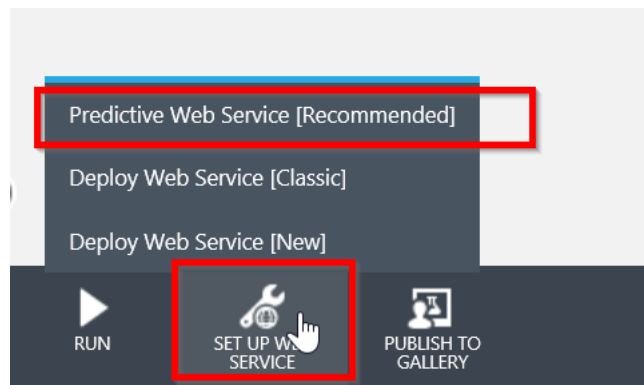
- Click the **X** in the top right corner of the **Visualize** dialogue box to close it.

# Publish a Model as a Web Service

## Create a Scoring Experiment from the Training Experiment

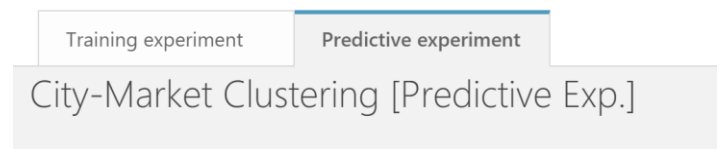
Once you are satisfied with the results of a training experiment, you need to convert that training experiment into an experiment that is optimized for scoring/predicting new data. For this, you will create a Scoring Experiment.

1. Click **RUN** at the bottom of the page to execute the training experiment.
2. Once the experiment finishes executing, click **SET UP WEB SERVICE** > **PREDICTIVE WEB SERVICE** at the bottom of the page.

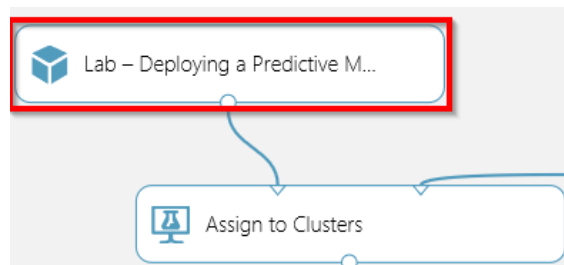


Notice a few things:

- a. There are now 2 tabs at the top of the ML Studio canvas: one for the *Predictive experiment*, and one for the *Training experiment*.

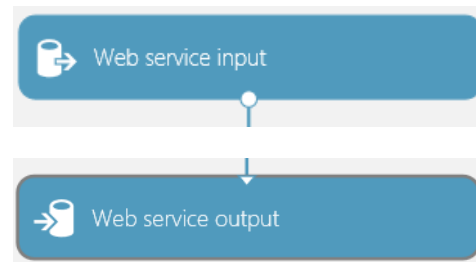


- b. The modules for training the model have all been replaced with a new module representing the trained clustering model. The new trained model module feeds the *Assign to Clusters* module.



- c. Modules for *Web service input* and *Web service output* have been added to the scoring experiment. These

modules indicate where data would be passed into and out of an experiment that is published as a web service.



3. Click **RUN** at the bottom of the page to execute the experiment.

The experiment will execute with the data set that is attached as the input data set (which is the same data as the Training experiment). It will score all of the rows using the trained clustering model.

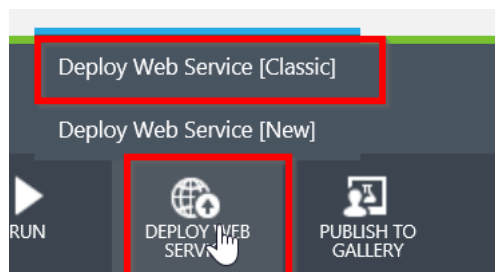
4. If the **PUBLISH WEB SERVICE** dialogue box pops up, close it by clicking the **X** in the top right corner.



### Publish the Scoring Experiment as a Web Service

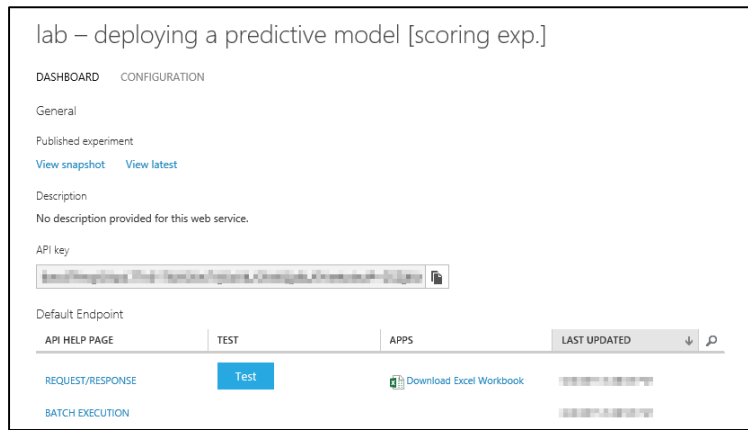
Next, you will publish your scoring experiment as a web service so that predictions can be made from other applications outside of the ML Studio.

1. Click **DEPLOY WEB SERVICE** at the bottom of the page > **Deploy Web Service [Classic]**.



The web service is created, and you now see the web service dashboard.





Here you will find:

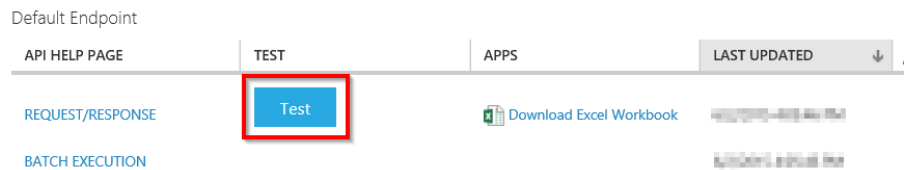
- a. *Links* to the latest experiment that was used to create the web service
- b. *A description* of the web service
- c. The *API key* used for authentication when other applications call the web service
- d. Other information about the *Default Endpoint* including:
  - a. help pages for the *REQUEST/RESPONSE* and *BATCH EXECUTION* APIs
  - b. a *TEST* option
  - c. an *Excel Workbook* to download
  - d. a data indicating when the endpoint was *LAST UPDATED*

# Make Predictions using the Web Service

## Test the API using the Web Interface

Finally, we will test our API using the provided Web UI on the web service dashboard page.

1. Click the **Test** button.



2. Enter the following values in their corresponding text boxes on the **Enter data to predict** dialogue box:

LOCATION: **Houston, TX**


GINIINDEX: **0.4802**

POP2014: **2239558**

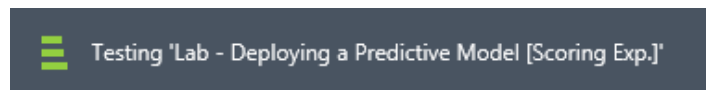
LANDAREA: **599.6**

POPULTREND: **6.63**

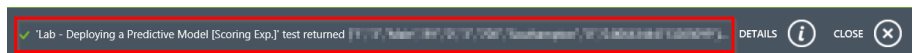
POPDENSITY: **3735**

3. Click the  at the bottom of the **Enter data to predict** dialogue box.

Notice a status bar at the bottom of the page is displayed indicating the web service call is being made.



Once the web service call is complete, the status bar will have a *green checkmark* next to the web service name along with the returned prediction from the API.



4. Click **DETAILS** on the status bar.



The status bar will expand to show the entire result body that was returned from the API including the cluster number.

```
← 'Lab - Deploying a Predictive Model [Scoring Exp.]' test returned ["Houston, TX","0.518043303929431","0.25510737963... CLOSE X]
Result: {"Results":{"output":{"type":"table","value":{"ColumnNames":{"Location","GiniIndex","Pop2014","LandArea","PopulTrend","PopDensity","Assignments"},"ColumnTypes":
["String","Double","Double","Double","Double","Double","Nullable 1"],"Values":{"Houston,
TX","0.518043303929431","0.255107379637065","0.798358413132695","0.561168895492818","0.11255016417365 P,"0"]}}} }
```

5. Click **CLOSE** on the status bar to return to the web service dashboard.

```
← 'Lab - Deploying a Predictive Model [Scoring Exp.]' test returned ["Houston, TX","0.518043303929431","0.25510737963... CLOSE X]
Result: {"Results":{"output":{"type":"table","value":{"ColumnNames":{"Location","GiniIndex","Pop2014","LandArea","PopulTrend","PopDensity","Assignments"},"ColumnTypes":
["String","Double","Double","Double","Double","Double","Nullable 1"],"Values":{"Houston,
TX","0.518043303929431","0.255107379637065","0.798358413132695","0.561168895492818","0.112550164173659","0"]}}}}}
```

# Exercises

Try the exercises outlined below.

1. If you have Excel 2010 or greater, download the Excel spreadsheet example and run the web API (hint: in the Excel Azure ML add-in there is a “Sample Data” button).
2. If you are familiar with C# or Python – try executing the sample code in the API help page.
3. Can you Deploy your first Azure ML experiment as a web API? i.e. in AALAB01 (Introduction to Azure Machine Learning) you created a linear regression model on the MPG dataset – can you publish this as a web API?
4. Read about the Azure ML Connector for Logic Apps - <https://azure.microsoft.com/en-us/documentation/articles/app-service-logic-connector-azureml/>. The Azure ML Connector for Logic Apps enables calling Azure ML APIs for batch scoring (Batch Execution Service) and retraining. These features in combination with Logic App Triggers enable scheduling batch jobs and setting up scheduled retraining of models.

## Conclusion

This concludes the *Deploying a Predictive Model with Azure Machine Learning* lab. To recap, you have successfully copied an experiment from the Azure Machine Learning Gallery into ML Studio, created a scoring experiment, deployed a model as a web service, and interacted with the API via the included web interface.

Now that you have a deployed web service, you can use the provided web UI or the provided Excel Worksheet to predict with new data. You can also create custom applications and data flows that access the API for automated prediction.

To answer our earlier question of where to open your next restaurant location, you would start by downloading the cluster assignments into a tool like Excel to try to label the type of city-market each cluster represents. Then you could take those cluster labels and compare them to your restaurant locations that have done well in the past. This would give you a list of prime city-markets for future restaurant locations.

# Terms of Use

© 2016 Microsoft Corporation. All rights reserved.

By using this Hands-on Lab, you agree to the following terms:

The technology/functionality described in this Hands-on Lab is provided by Microsoft Corporation in a “sandbox” testing environment for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the Hands-on Lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this Hands-on Lab or any portion thereof.

COPYING OR REPRODUCTION OF THE HANDS-ON LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS HANDS-ON LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS HANDS-ON LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCTIONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

**FEEDBACK.** If you give feedback about the technology features, functionality and/or concepts described in this Hands-on Lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE HANDS-ON LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF THE VIRTUAL LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE VIRTUAL LAB FOR ANY PURPOSE.

## DISCLAIMER

This lab contains only a portion of the features and enhancements in Microsoft Azure Machine Learning. Some of the features might change in future releases of the product.