Azure Machine Learning Workshop for Data Scientists

|  |  |  |
| --- | --- | --- |
|  |  |  |

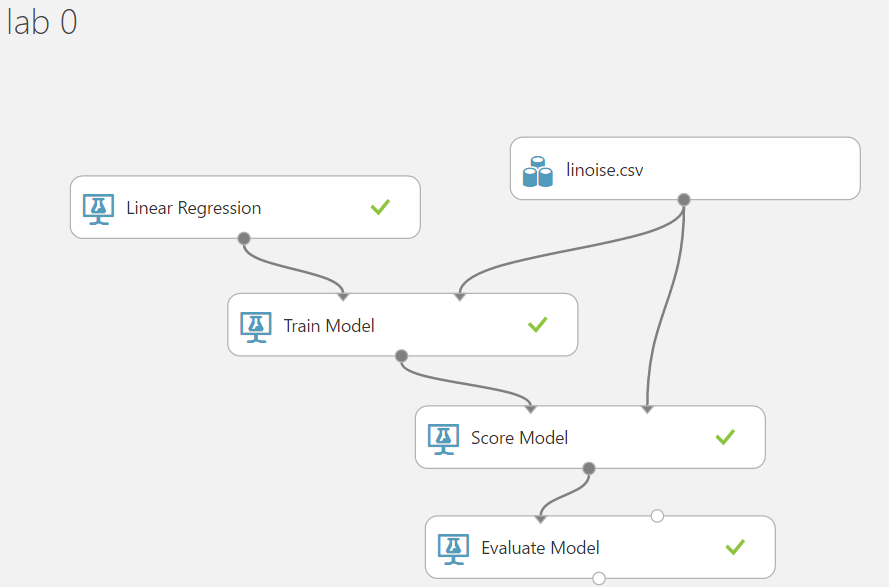
# Lab 0: Build a linear regression model and deploy it as a web service (Estimate: 1hr)

The purpose of this lab is to build a quick linear regression model and deploy it as a web service in Azure Machine Learning Studio. This lab is covering two important skills in using AML:

1. Build, Train and Deploy models as web services.
2. Customize a predictive experiment as need by removing any unneeded input of output values from the model.
3. Calling AML web services from R Studio.

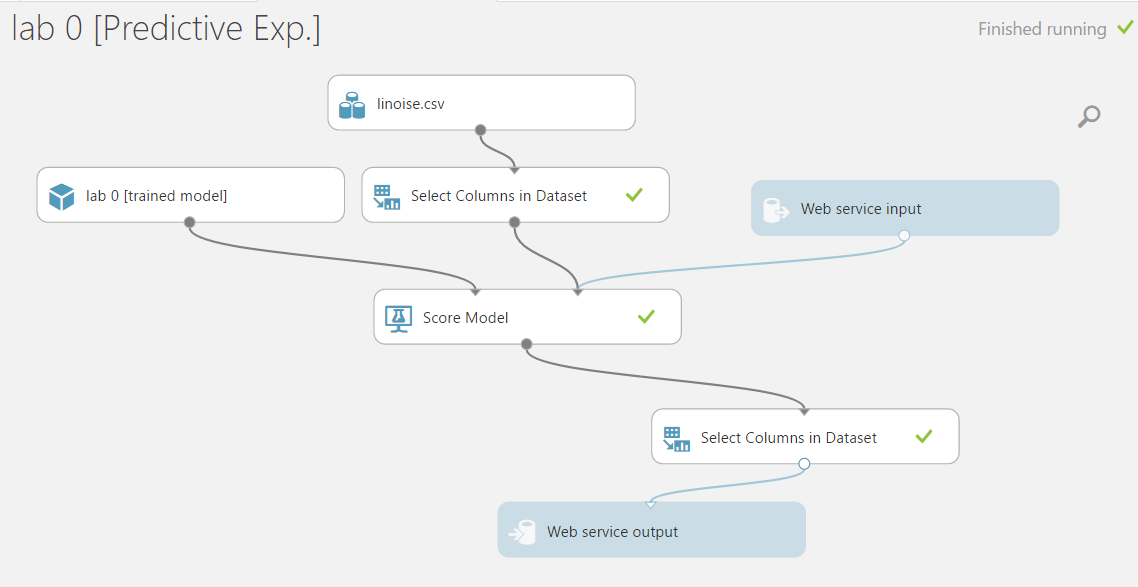
Lab 0 experiment is using a synthetic data file named “**linoise.zip**”, this file is in the workshop directory. We will build and publish this model in AML account.

1. Below figure shows the regression model experiment in AML you will build.



You will need to deploy this model as a web service after you finish building this experiment.

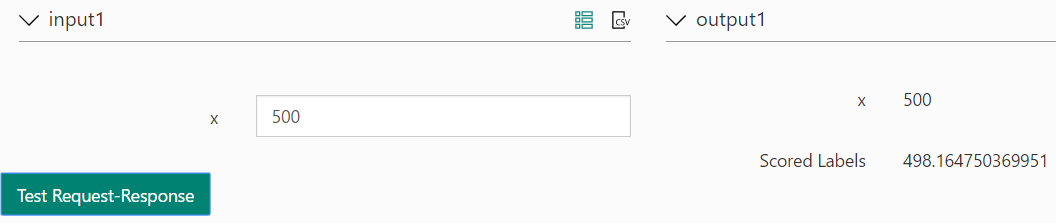
1. Below figure shows a customized predictive experiment that accepts one input (x) and provide the scored label as an output. To accomplish this, you need to add two “Select Columns” modules to the generated predictive experience. The first is to exclude “y” from the input data set, so we don’t need to enter “y” since we need to predict its value. The second is to exclude “y” from the output, so the model shows the scored label for any given “x” value.



Note: if you would like a walkthrough instructions for this lab. Please implement steps from 4.1 to 4.2.3. this lab is posted [here](https://github.com/melzoghbi/hol-azure-machine-learning/blob/master/004-lab-azureml-experiment.md).

**Testing the predictive experiment**

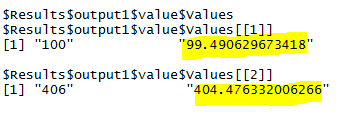
Using test endpoint (Preview) user interface in AML. The output for your predictive experiment should look like below screen shot.



1. **Calling AML WS from R Studio**

This section allows you to call published AML web services from R Studio. To call a published AML web services, follow below steps:

1. Open R Studio.
2. Copy and paste the code in “**callAMLWS.R**”, change the access key with the one you just published.
3. Run the R program and check out the output as shown in the below figure.



# **Lab 1**: [Monetizing Azure ML Web Services](https://github.com/melzoghbi/hol-azure-machine-learning/blob/master/009-lab-monetization.md) (Estimate: 1hr)

In the previous lab, we learned how to develop and deploy an Azure Machine Learning solution as a web service. Using the API key (password) and the web service URI anyone can access the solution and consume it from almost any type of application. Having single API key, how it is possible to share the service with multiple users, consumers? How to track the number of API calls per specific user? How to limit the API calls to a specific number? How to limit the access to your service with basic, OAuth etc. authentication? Is it possible for a user to consume the service with his/her facebook, microsoft or other accounts? etc.

# **Lab 2**: [Text Analytics](https://text-analytics-demo.azurewebsites.net/Home/SampleCode) (Estimate: 1hr)

You will build a console app that identify:

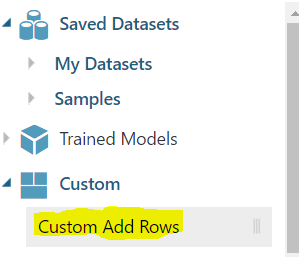
1. Key phrases
2. Language
3. Sentiment APIs
4. Topic Detection API
5. Provision Text Analytics Cognitive Services from Azure Portal.
   1. Sign in to the Azure portal.
   2. Click + NEW.
   3. Select Intelligence + Analytics and then Cognitive Services APIs (preview).
   4. Select Text Analytics API.
   5. Set the rest of portal fields and click on create to provision this service.

Need help? Here is an [article](https://docs.microsoft.com/en-us/azure/cognitive-services/cognitive-services-apis-create-account) on how to provision Cognitive Services in Azure portal.

1. Create a console app in Visual Studio 2015, follow these [instructions](https://text-analytics-demo.azurewebsites.net/Home/SampleCode) to integrate with Text Analytics API.

# **Lab 3**: Package and register a custom R module (Estimate: 1hr)

1. Extract the zip file and look at the structure of xml file and R source file. Examine the R function implementation, input and output arguments and XML manifest file.
2. To register them in your Machine Learning workspace, go to your workspace in the Machine Learning Studio, click the +NEW button on the bottom and choose MODULE -> FROM ZIP PACKAGE to upload the new Custom Add Rows module.
3. Create a new AML empty experiment, Drag the newly created custom module to an experiment. The new custom module will be listed under custom tab.



**Optional Labs:**

# Lab 4: [leverage the power of Power BI Embedded into LOB Apps](https://docs.microsoft.com/en-us/azure/power-bi-embedded/power-bi-embedded-get-started-sample)

With Microsoft Power BI Embedded, you can integrate Power BI reports right into your web or mobile applications. In this article, we'll introduce you to the Power BI Embedded get started sample.

# Lab 5: [Build Recommender Systems using Cognitive Services Recommendations API](https://github.com/melzoghbi/RecommenderApp)

In this lab, you are going to build a function app that uses content files; these files are stored in Azure blob storage to recommend items (Item to Item recommendation) or recommend user's items (User to item recommendation). The content files are restaurants catalog files which contains restaurant’s information and user’s transactions for visited restaurants.