Azure Machine Learning Studio

Tutorial 1

Chun-Kit Yeung

Hong Kong University of Science and Technology

COMP4211 (Spring 2018)

Who am I?

- Your TA for this course :)
- Graduated from HKUST and now a 2-nd year MPhil student.
- Interested in real world application of machine learning. Specifically, machine learning in education.
- Speak English, Mandarin, and native in Cantonese. Ask me any question in whatever language you like.

Who are you?

In order to give a right materials in this tutorial, I would like to know:

- Which major background are you from?
- Why are you taking this course?
- What are you expected to learn from this course?
- What are you expected to learn from the tutorial session? (You want to know more about the theoretical stuffs, or something practical? It is just a 50 minutes tutorial, I can't cover both in detail LoL)

What is your technical background?

Also, to give a right paced in this tutorial, I would like to know:

- What is your proficiency level in Python?
- Have you heard of Numpy, Pandas?
- What about Scikit-learn?
- Tensorflow, MXNet, PyTorch, CNTK?

What are the goals of tutorial in this course?

Throughout the course, we will mainly using Azure to manage our machine learning project. It provide the working environment on cloud, e.g. Azure Machine Learning Studio and Data Science Virtual Machine, and on desktop, e.g. Azure Machine Learning Workbench. You will be able to manage and run your machine learning project via Azure during this tutorial.

My personal goals, on behalf of a TA, are to cover not only the theoretical concepts, but also some practical experience in machine learning so that you can write on your CV.

What are we going to do in this tutorial?

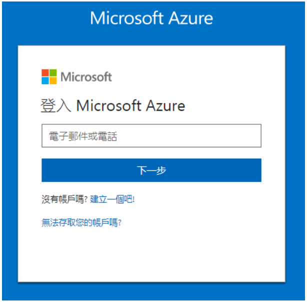
Throughout the course, we will be mainly using *Azure* powered by Microsoft to handle the machine learning project.

In this tutorial, you will

- Register an account on Azure
- Have a first touch on Azure Machine Learning (ML) Studio
- Build a Iris classification project using naive Bayes classifier (if time is available.)

Register Account in Azure

1. Open an browser and enter `https://azure.microsoft.com/en-us/`
2. Press “Start Free” until the login page.
3. Login Microsoft Azure with your email (preferably xxx@connect.ust.hk)



Microsoft Azure

Microsoft

登入 Microsoft Azure

電子郵件或電話

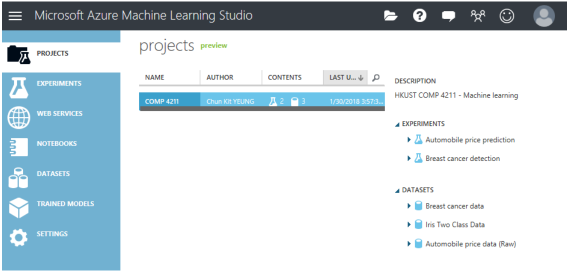
下一步

沒有帳戶嗎? 建立一個吧!

無法存取您的帳戶嗎?

Azure Machine Learning Studio

1. Enter `https://studio.azureml.net/`
2. Press “Sign In”
3. Login with your registered account.
4. After login, you will see something like following, except the project which I created earlier/



NAME	AUTHOR	CONTENTS	LAST U...
COMP 4211	Chun Kit YEUNG	2 3	1/30/2018 3:57:3...

DESCRIPTION

HKUST COMP 4211 - Machine learning

EXPERIMENTS
- Automobile price prediction
- Breast cancer detection

DATASETS
- Breast cancer data
- Iris Two Class Data
- Automobile price data (Raw)

Create an experiment

Create a model

1. Get data
2. Prepare the data (Not required this time.)
3. Define features

Train the model

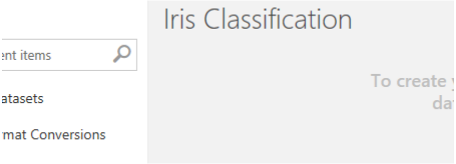
4. Choose and apply a learning algorithm

Score and test the model

5. Predict new iris labels

Step 1: Get Data

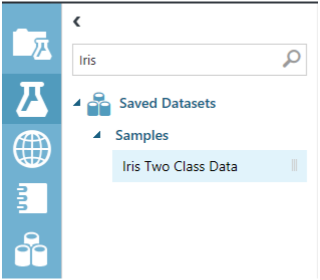
1. Create a new experiment by clicking **+NEW** at the bottom of the Machine Learning Studio window, select **EXPERIMENT**, and then select **Blank Experiment.**
2. The experiment is given a default name that you can see at the top of the canvas. Select this text and rename it to something meaningful, for example, **Iris Classification.** The name doesn't need to be unique.



Iris Classification

Step 1: Get Data

3. To the left of the experiment canvas is a palette of datasets and modules. Type **Iris** in the Search box at the top of this palette to find the dataset labeled **Iris Two Class Data.** Drag this dataset to the experiment canvas.



Iris

Saved Datasets

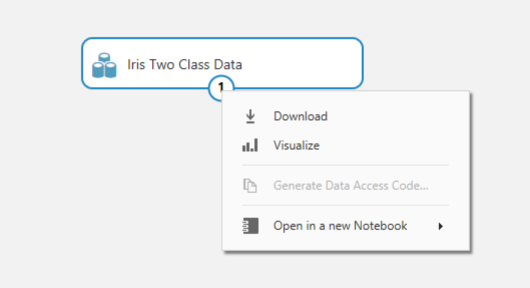
Samples

Iris Two Class Data

Step 2: Prepare Data

A dataset usually requires some preprocessing before it can be analyzed, as the data may be corrupted or inaccurate in the database, even missing values would be present in the columns of various rows. Fortunately, in the Iris dataset, the data is cleaned already and *do not require preprocessing*.

Let’s see what the data looks like in Azure. Click the output port at the bottom of the automobile dataset, and then select **Visualize**.



Step 3: Define the features

The goal of the iris classification problem is to find a machine learning model $h(\cdot)$ to predict whether the given sample $\mathbf{x}$ is an iris or not, i.e. $y \in \{0, 1\}$, where 1 indicates the sample is an iris.

Specifically, we are going to select a supervised machine learning model (or alternatively you can view it as a math function) $h(\cdot)$ which maps an input sample $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ to an output y, i.e. $y = h(\mathbf{x})$. Different machine learning algorithms have different approaches in finding such the model $h(\cdot)$.

Before to train a model, one thing has to be decided. That is what should the $\mathbf{x}$ be to feed in the algorithm.

Step 3: Define the features

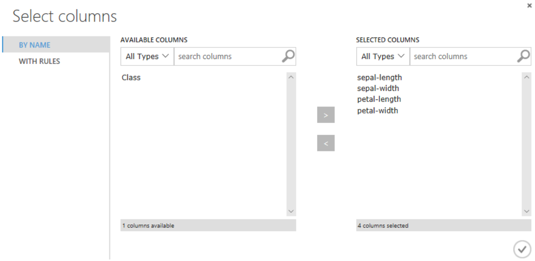
In Iris dataset, there are 5 columns: *class, sepal-length, sepal-width, petal-length, and petal-width*.

Obviously, the class indicates whether a sample is an iris or not, while the sepal-length, sepal-width, petal-length, and petal-width are the information about the sample.

Since they might all be relevant to classify whether a given sample is an iris, we will use them all to form the input x. We also call x to be *“features”*, as they are the features characterizing the sample.

Step 3: Define the features

1. Drag another **Select Columns in Dataset** module to the experiment canvas. Connect the left output port of the Clean Missing Data module to the input of the Select Columns in Dataset module.
2. Click **Launch column selector** in the **Properties** pane.
3. Select the features as follow:



Select columns

Class

1 columns available

sepal-length
sepal-width
petal-length
petal-width

4 columns selected

4. Click the check mark (OK) button.

Step 4: Choose and apply a learning algorithm

Now that the data is ready, constructing a predictive model consists of training and testing. We'll use our data to train the model, and then we'll test the model to see how closely it is able to predict the label.

1. Select and drag the "Split Data" module to the experiment canvas and connect it to the last 11Select Columns in Dataset" module.
2. Click the "Split Data" module to select it. Find the **Fraction of rows in the first output dataset** (in the **Properties** pane to the right of the canvas) and set it to 0.75. This way, we'll use 75 percent of the data to train the model, and hold back 25 percent for testing (later, you can experiment with using different percentages).
3. Visualize the two output ports to see the split result.

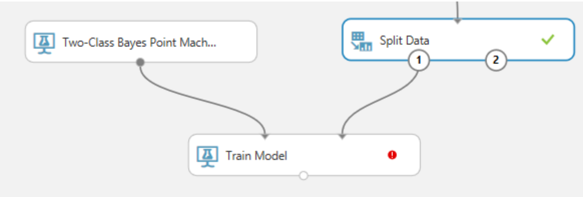
Step 4: Choose and apply a learning algorithm

4. To select the learning algorithm, expand the Machine Learning category in the module palette to the left of the canvas, and then expand Initialize Model. This displays several categories of modules that can be used to initialize machine learning algorithms.

   For this experiment, select the “Two-Class Bayes Point Machine” which is the naive Bayes classifier, and drag it to the experiment canvas.

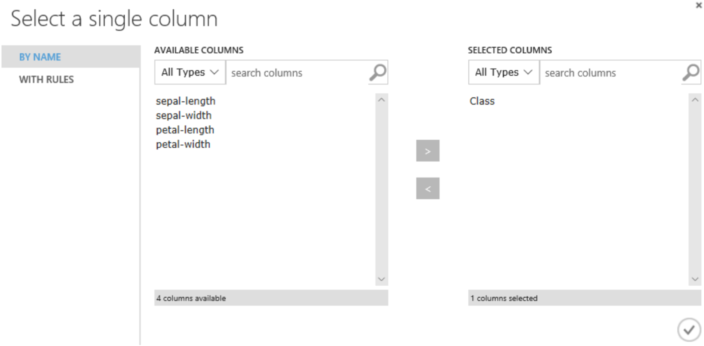
Step 4: Choose and apply a learning algorithm

5. Find and drag the “Train Model” module to the experiment canvas. Connect the output of the “Linear Regression” module to the left input of the “Train Model” module, and connect the training data output (left port) of the “Split Data” module to the right input of the “Train Model” module.



Step 4: Choose and apply a learning algorithm

6. Click the “Train Model” module, click **Launch column selector** in the **Properties** pane, and then select the **class** column. This is the value that our model is going to predict.



Select a single column

sepal-length
sepal-width
petal-length
petal-width

Class

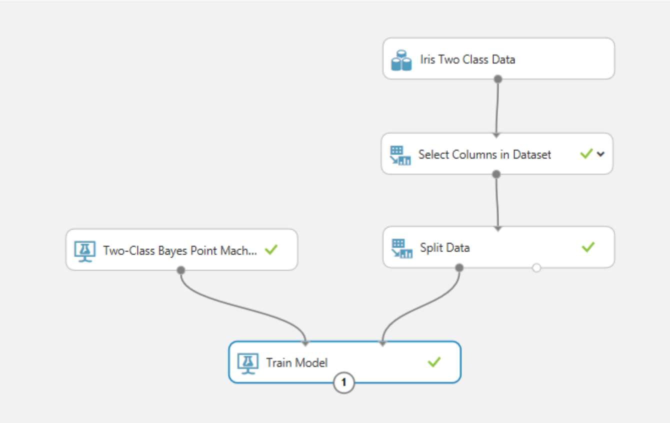
4 columns available

1 columns selected

7. Run the experiment.

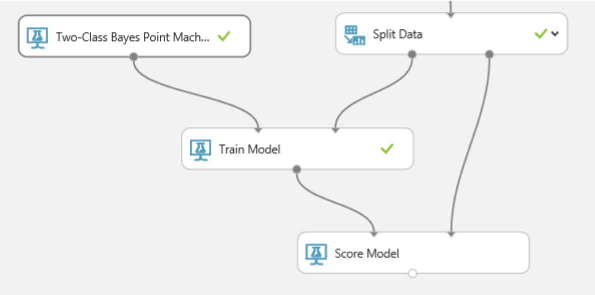
Step 4: Choose and apply a learning algorithm

The computation graph up to this moment:



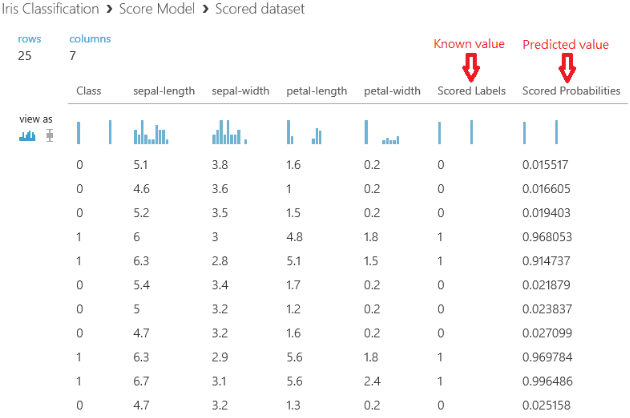
Step 5: Predict new iris label

1. Find and drag the Score Model module to the experiment canvas. Connect the output of the “Train Model” module to the left input port of “Score Model”. Connect the test data output (right port) of the Split Data module to the right input port of “Score Model”.



Step 5: Predict new iris label

2. Run the experiment and view the output from the “Score Model” module (click the output port of Score Model and select Visualize). The output shows the predicted values for price and the known values from the test data.



Iris Classification › Score Model › Scored dataset

rows 25

columns 7

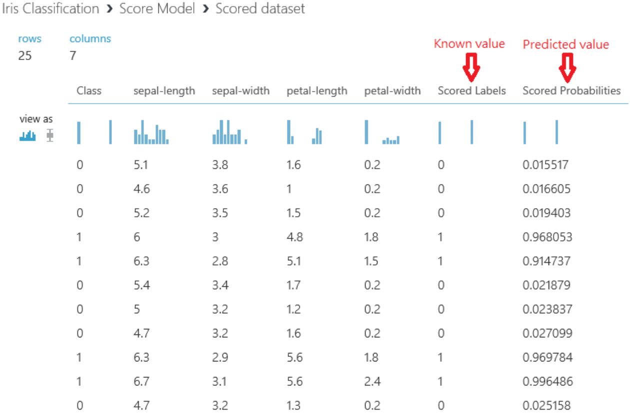
Known value

Predicted value

Class	sepal-length	sepal-width	petal-length	petal-width	Scored Labels	Scored Probabilities
0	5.1	3.8	1.6	0.2	0	0.015517
0	4.6	3.6	1	0.2	0	0.016605
0	5.2	3.5	1.5	0.2	0	0.019403
1	6	3	4.8	1.8	1	0.968053
1	6.3	2.8	5.1	1.5	1	0.914737
0	5.4	3.4	1.7	0.2	0	0.021879
0	5	3.2	1.2	0.2	0	0.023837
0	4.7	3.2	1.6	0.2	0	0.027099
1	6.3	2.9	5.6	1.8	1	0.969784
1	6.7	3.1	5.6	2.4	1	0.996486
0	4.7	3.2	1.3	0.2	0	0.025158

Step 5: Predict new iris label

3. Run the experiment and view the output from the “Score Model” module (click the output port of Score Model and select Visualize). The output shows the predicted values for price and the known values from the test data.

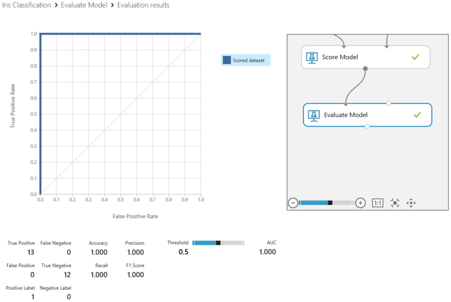


Iris Classification › Score Model › Scored dataset

Class	sepal-length	sepal-width	petal-length	petal-width	Scored Labels	Scored Probabilities
0	5.1	3.8	1.6	0.2	0	0.015517
0	4.6	3.6	1	0.2	0	0.016605
0	5.2	3.5	1.5	0.2	0	0.019403
1	6	3	4.8	1.8	1	0.968053
1	6.3	2.8	5.1	1.5	1	0.914737
0	5.4	3.4	1.7	0.2	0	0.021879
0	5	3.2	1.2	0.2	0	0.023837
0	4.7	3.2	1.6	0.2	0	0.027099
1	6.3	2.9	5.6	1.8	1	0.969784
1	6.7	3.1	5.6	2.4	1	0.996486
0	4.7	3.2	1.3	0.2	0	0.025158

Step 5: Predict new iris label

4. Finally, we test the quality of the results. Select and drag the "Evaluate Model" module to the experiment canvas, and connect the output of the "Score Model" module to the left input of "Evaluate Model".
5. Run the experiment.
6. Visualize the "Evaluate Model"

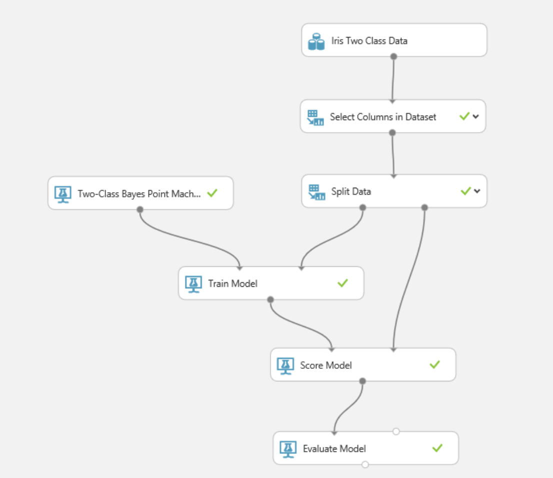


Iris Classification › Evaluate Model › Evaluation results

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
13	0	1.000	1.000	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
0	12	1.000	1.000		
Positive Label	Negative Label				
1	0				

Step 5: Predict new iris label

The computation graph eventually:



Optional Exercises

- What is the “Random seed” in “Split data”? Why is it useful?
- What would be the potential implications if we increase or decrease the fraction of rows split in “Split Data”.
- Try out other classification algorithms. (Further question: How can you compare the performances of two different models? (Refer to the demo of breast cancer detection in reference and further reading))
- Try out the Machine learning tutorial (in reference and further reading). See what is a regression problem.

Reference and Further Reading

- Machine learning tutorial: Create your first data science experiment in Azure Machine Learning Studio: available in `https://docs.microsoft.com/en-us/azure/machine-learning/studio/create-experiment`
- Demo of breast cancer detection using two-class naive Bayes classifier in Azure ML Studio: available in `https://gallery.cortanaintelligence.com/Experiment/Breast-cancer-detection`