Perceptron and ADALINE

COMP 4211 - Tutorial 04

Chun-Kit Yeung

Hong Kong University of Science and Technology

2018-03-09

Perceptron and ADALINE

Both of them are *single-layer* neural network models.

Net Input

Both perceptron and ADALINE compute the so-called net input z

$$\begin{aligned} z &= w_0 x_0 + w_1 x_1 + \ldots + w_m x_m \\ &= \sum_{j=0}^{m} w_j x_j = \mathbf{w}^T \mathbf{x}, \end{aligned}$$

where $x_0 = 1$ and x_j is the input attribute.

Perceptron

In perceptron, the net input z is undergone a step function to make a prediction.

$$\hat{y} = \text{step}(z) = \text{step}(\mathbf{w}^T\mathbf{x}) = \begin{cases} 1, & \text{if } z > 0 \\ 0, & \text{if } z \leq 0 \end{cases}$$

where $\hat{y}$ is the prediction value.

Perceptron learns each weight iteratively by

$$w_j := w_j + \eta x_j (y - \hat{y}) \tag{1}$$

where y is the ground true and η is the learning rate.

ADALINE

Similar to perceptron, ADALINE makes a prediction by

$$\hat{y} = \text{step}(z) = \text{step}(\mathbf{w}^T\mathbf{x}) = \begin{cases} 1, & \text{if } z > 0 \\ 0, & \text{if } z \leq 0 \end{cases}$$

However, ADALINE learns the weights by gradient descent which aims to minimize the error $E = (y - z)^2$. Each weight is updated iteratively by

$$w_j := w_j - \eta \frac{\partial E}{\partial w_j}$$

where η is the learning rate.

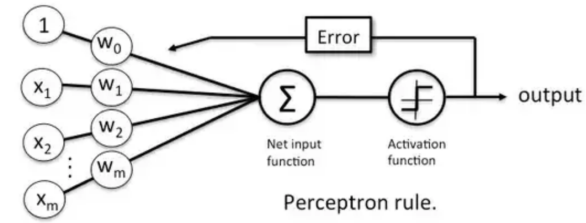
Perceptron vs ADALINE

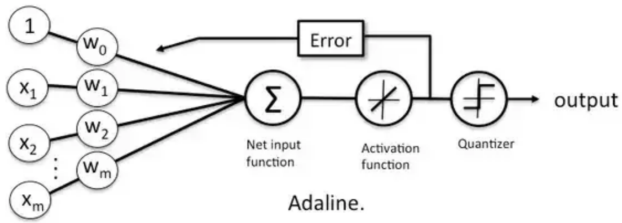
Commons:

- Binary classifier
- Linear decision boundary
- Learn iteratively, sample by sample
- Use a threshold function

Differences:

- Perceptron uses the class labels to learn model coefficients
- ADALINE uses continuous predicted values to learn the model coefficients, which is more “powerful” since it tells us by “how much” we were right or wrong.





Extracted from Quora.

Supplement: Linear Regression and Logistic Regression

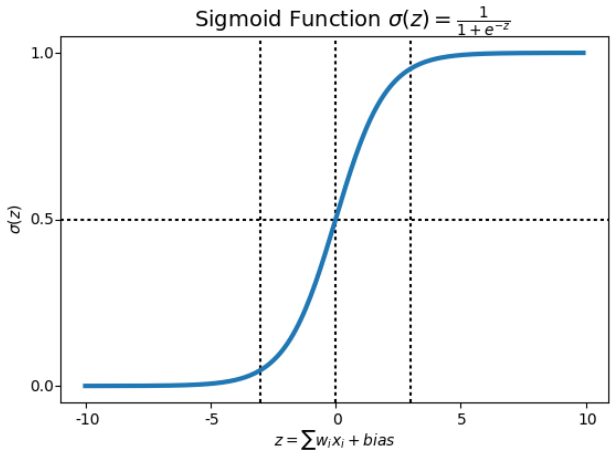
Linear regression

- It is simply $\hat{y} = z = \mathbf{w}^T \mathbf{x}$.

Logistic regression

- Passing the net input z to a sigmoid function.

$$\begin{aligned} \hat{y} &= \text{sigmoid}(z) \\ &= \frac{1}{1 + e^{-z}} \\ &= \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}} \end{aligned}$$



Extracted from Towards Data Science.

Let's code.

To better understand today tutorial, the following .ipynb is covered:

- T04_perceptron_and_adaline.ipynb

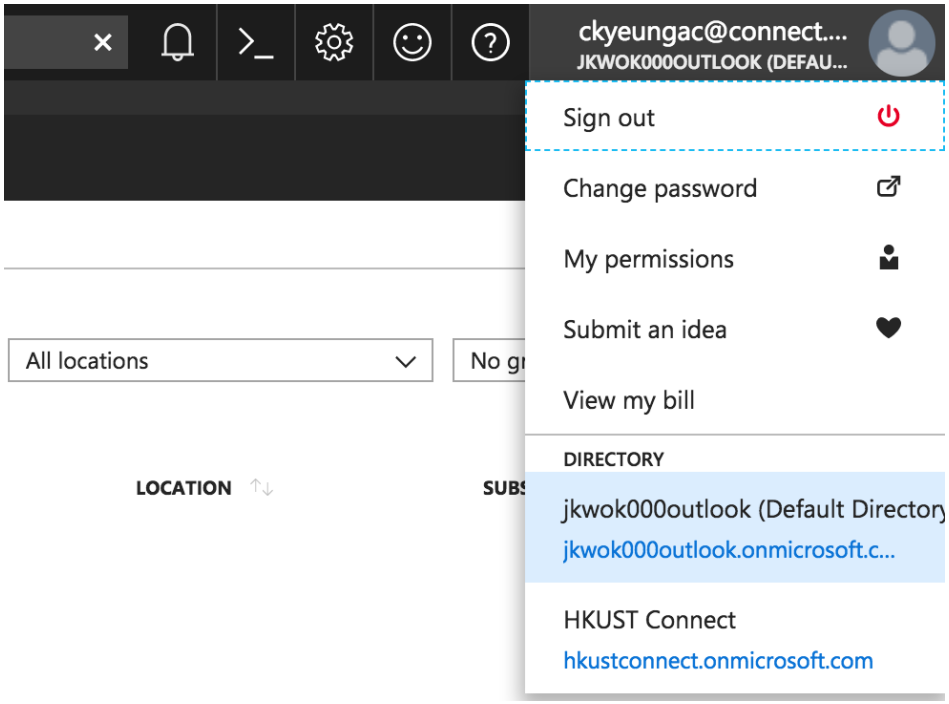
Miscellaneous

Create a new workspace for Azure Machine Learning Studio with credit offered by COMP 4211.

Create a new workspace for Machine Learning Studio

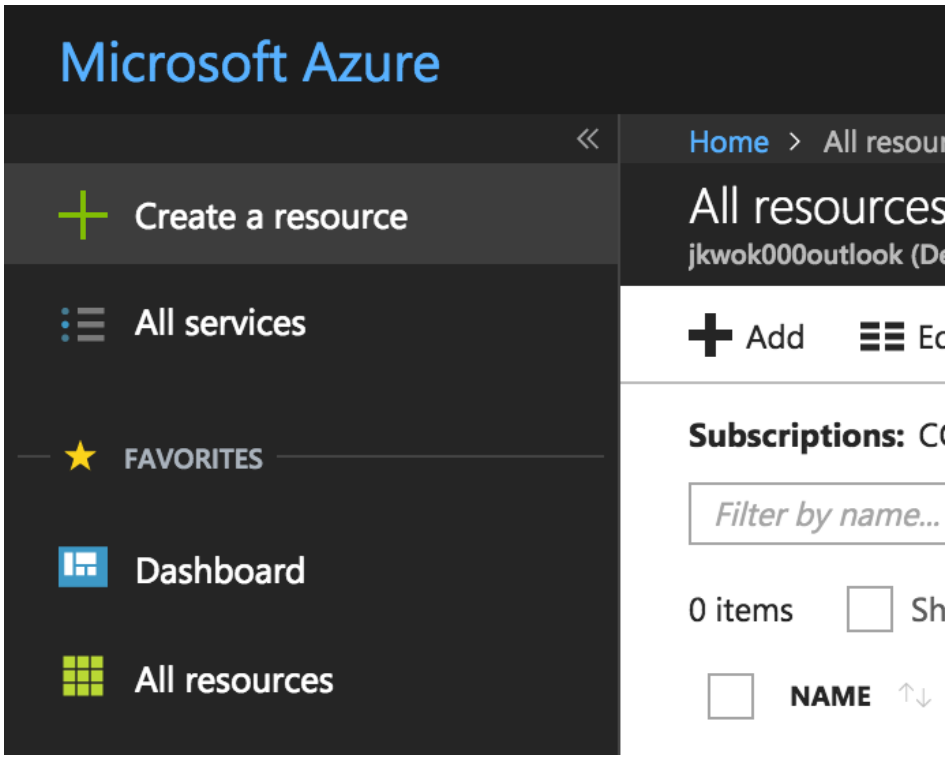
As in this course, you are offered with some Azure credits. You can use the standalone workplace in Azure. To set up the workplace, you are required to do the following steps:

1. Go to azure portal `https://portal.azure.com`
2. Change the 'directory' at up right corner to jamesk000outlook



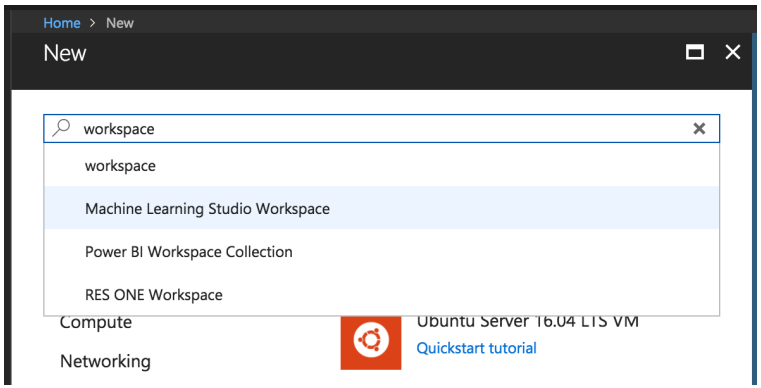
Create a new workspace for Machine Learning Studio

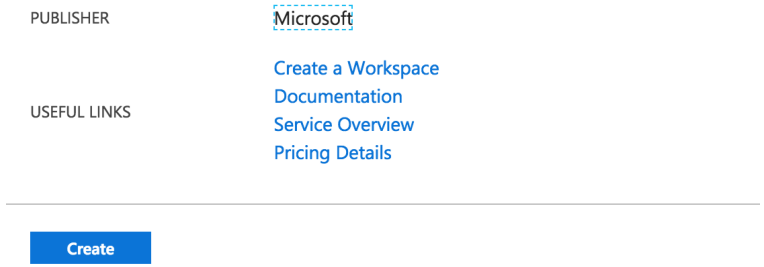
3. Click "+ Create a resource"



Create a new workspace for Machine Learning Studio

4. Type “workspace” in search bar and choose ML studio workspace. Then, go down to the bottom and click “Create”

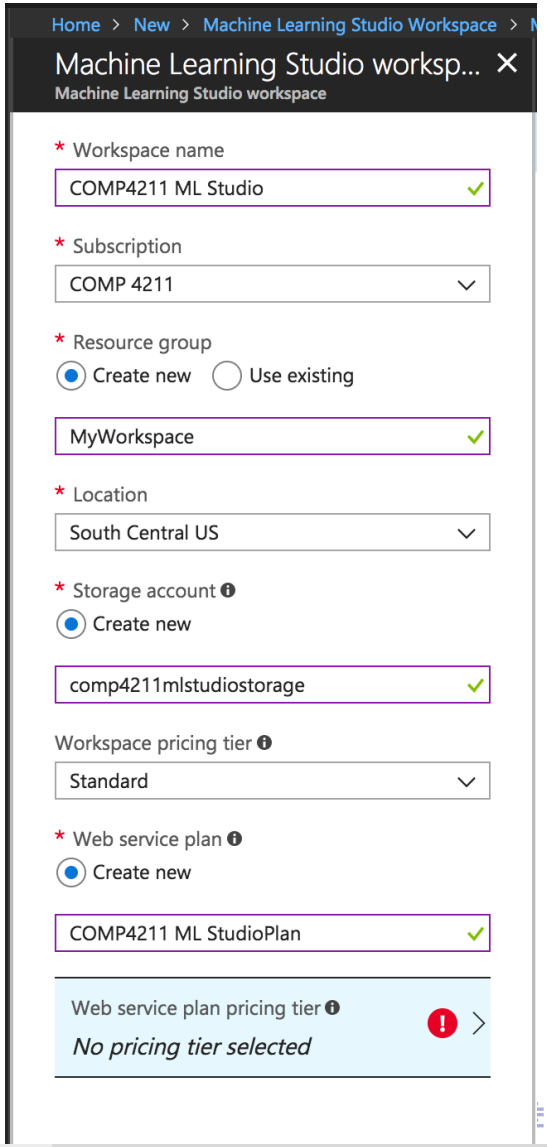




Create a new workspace for Machine Learning Studio

After clicking “Create”, you need to configure your workspace

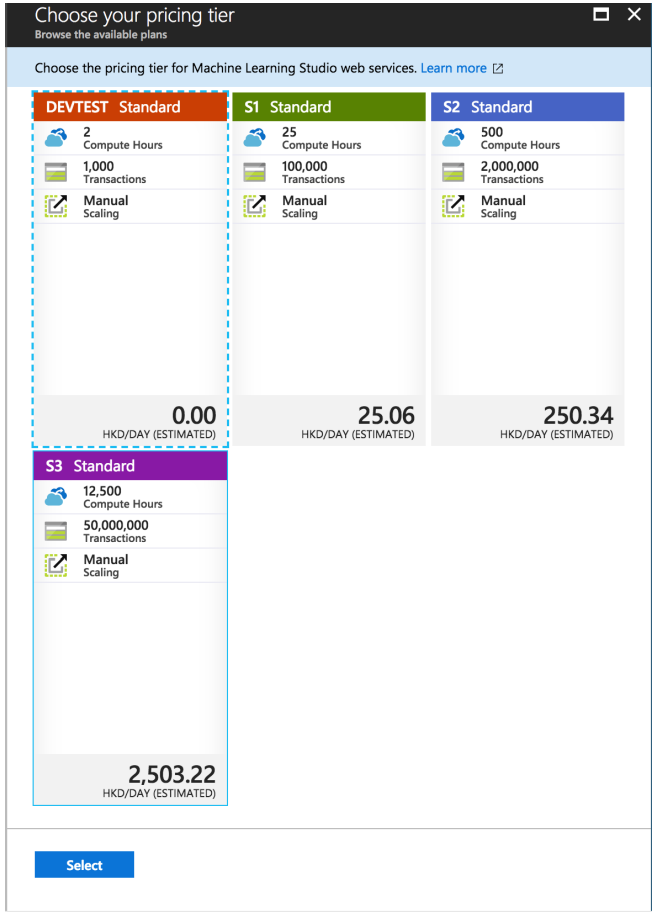
5. Do the configuration as the picture shown in the right. (The “Web service plan pricing tier” will be left in next slide)



Create a new workspace for Machine Learning Studio

For “Web service plan pricing tier”, click it and then

6. Choose **“DEVTEST Standard”** [IMPORTANT!!]
7. Click “Select” in this selection canvas.
8. Click “Create” in the configuration canvas.



Create a new workspace for Machine Learning Studio

After finishing the above steps, the new workspace should be created. To access the new workspace,

9. go to ML studio `https://studio.azureml.net/`
10. choose the newly created workspace. ("COMP4211 ML Studio")