IBM Watson Studio - Machine Learning Workshop

Prerequisites: IBM Cloud Setup, and a Watson Studio service instance. See the Setup and Installation Guide for more details.

This topic lists tutorials that demonstrate features of the Watson Machine Learning model in IBM Watson Studio. These tutorials all use the same sample data set to train different types of machine learning models: a binary classifier, a multiclass classifier, and a regression model. No coding is required to complete any of these tutorials. Through each of these three tutorials, you will be using Watson Studio to build machine learning models which will predict the likelihood of a customer buying camping equipment, based on sample historical customer data.

The sample data needed for this tutorial is listed below.

> **GoSales.csv** (also found here: https://dataplatform.cloud.ibm.com/exchange/public/entry/view/aa07a773f71cf1172a349f33e2028e4e)

You should be able to complete any one of these tutorials in less than 20 minutes depending on loading and running time.

To get started, open IBM Watson Studio through the IBM Cloud site at https://cloud.ibm.com and sign in.

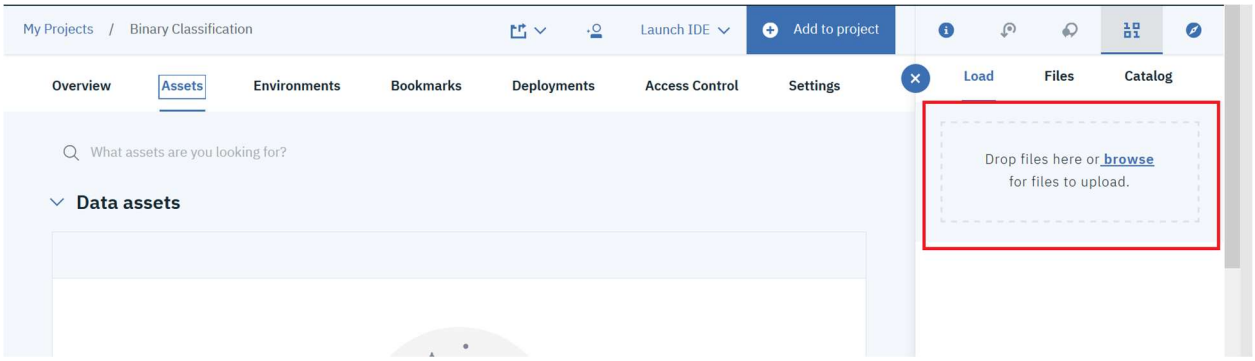
Tutorial 1 – Binary Classification

Binary classification is simple; our model will classify data into two discrete categories.

Part 1: Build and train the model

Step 1: Add training data

1. On the Watson studio start page, click the **Create a project** card.
2. Create an **Empty Project**.
3. Name the project “Binary Classification”.
4. Click on the **Assets** tab.
5. In the right-hand panel, **upload** the training data file “GoSales.csv” using either browse or drag-and-drop.



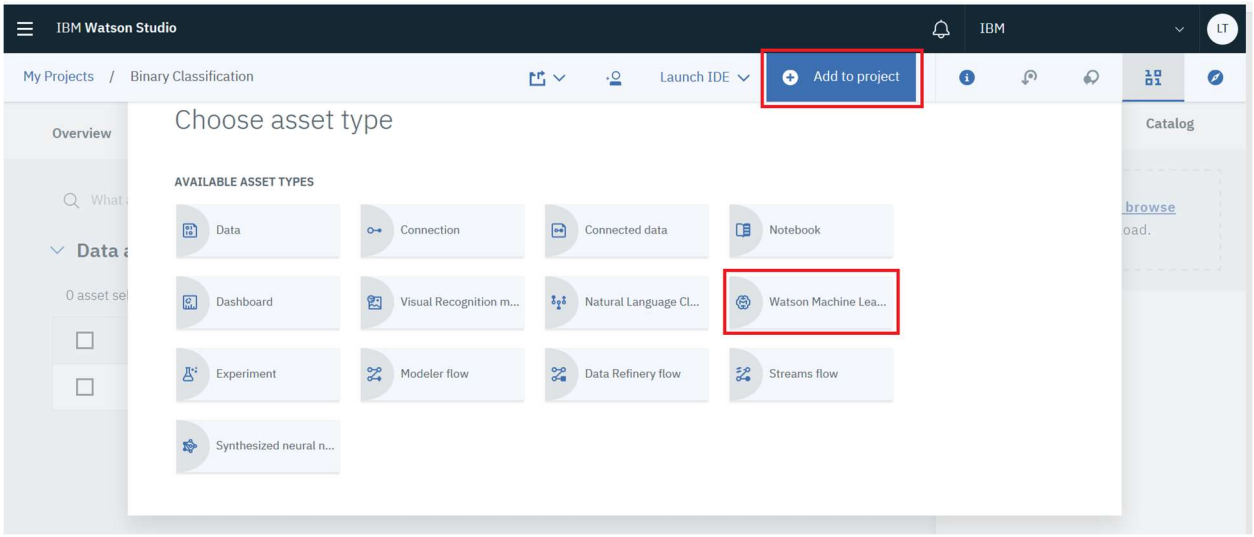
6. Under **Data Assets**, you should see your file appear.
7. If you do not already have an Machine Learning service instance associated with your project, create one now using the follow steps. If you already have one, skip to Step 2**.**
8. Creating and associating a Machine Learning service instance.
    a. Click on the **Settings** tab.
    b. Under Associated Services, click on the **Add service** drop-down menu.
    c. Select the **Watson** option.
    d. Select the **Machine Learning** card.
    e. Leave the auto-selected plan (Lite) as selected and click **Create**.
    f. On the **Confirm Creation** popup, double-check the following settings:

- Region: US-South
- Plan: Lite
- Resource Group: Default.
- Service Name: *Auto-Generated*.

g. Click the **Confirm** button.

Step 2: Specify basic model details

1. Click on **Add to project**.



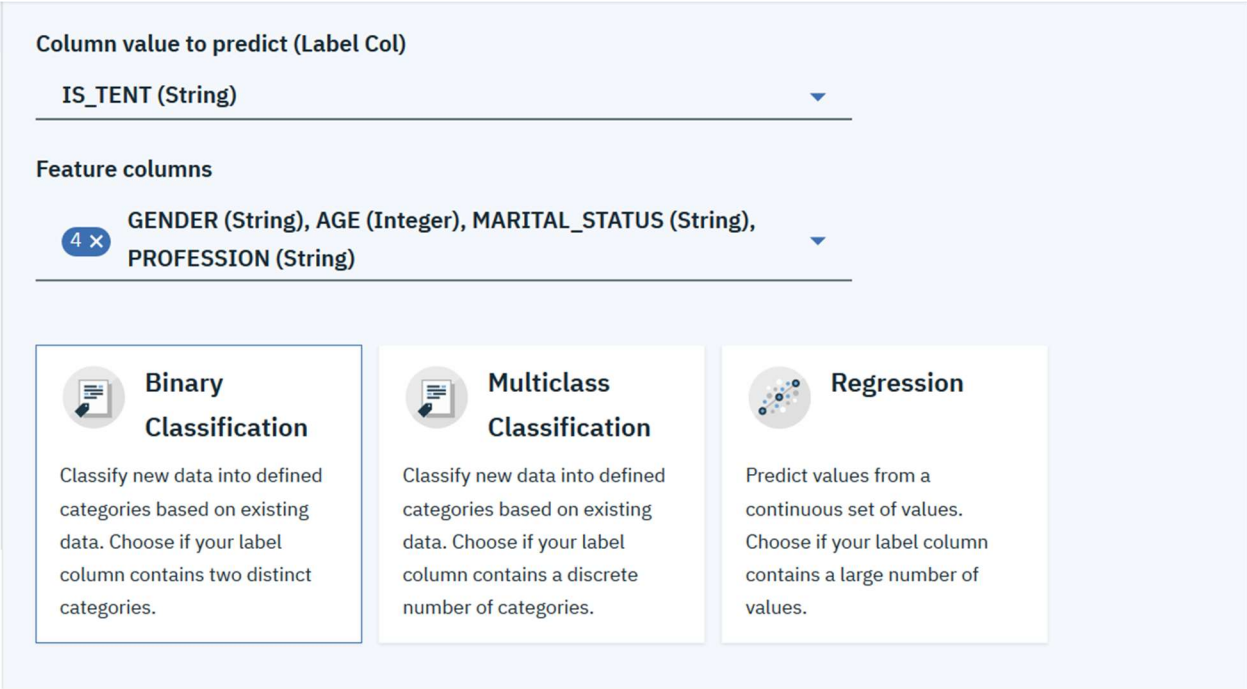
2. From the popup, choose the **Watson Machine Learning Model** option
    a. Name the model "**Binary Classification**".
    b. Feel free to leave the description box blank.
    c. Set the Machine Learning Service to the one you generated.
    d. Select Model Type: **Model Builder**.
    e. Runtime as "**Default Spark Scala 2.11**" form the dropdown menu
    f. Click the card labeled **Automatic**.
    g. Click the **Create** button.

Step 3: Train the model

1. Select the data asset which you uploaded earlier (**GoSales.csv**) and click **Next**.
2. For the label column drop-down menu, select the **IS_TENT** variable.

3. For the Feature Columns, select **GENDER, AGE, MARITAL_STATUS**, and **PROFESSION.**

The *label column* is what the model will predict. *Feature columns* contain the attributes on which the model will base predictions.



4. Select the **Binary Classification** card.
5. Click **Next**.
6. **Select** the model the you just made.
7. Click **Save**.
8. **Confirm** that your model has saved.

Part 2: Deploy the trained model

1. Click the **Deployments** tab.
2. Click the **Add Deployment** button.
3. Fill in the **Name** field and select “Web Service” as the **Deployment Type**.
4. Click the **Save** button.
5. Click on your deployment name to view the deployment details page.

Part 3: Test the deployed model

1. Click the **Test** tab of the deployment details page.
2. Fill in some values from the following feature columns: **GENDER**, **AGE**, **MARITAL_STATUS**, and **PROFESSION** (Example values can be found in the data file. Ignore the other fields).
3. Click the **Predict** button.

Now that you have built, trained, and tested your model, it will predict the likelihood of a customer with your entered values buying a tent. This has been a Binary Classification model, which only uses one variable in the label column, to predict customer choices in a binary (True/False) fashion. Next, we will build a Multiclass Classification model.

Tutorial 2 – Multiclass Classification

Multiclass classification is more nuanced; our model will classify data into multiple discrete categories.

Make a new **Empty Project** titled “Multiclass Classification”.

This project will use the same data as the other ones

Part 1: Build and train the model

Step 1: Add training data

1. Click on the **Assets** tab.
2. In the right hand panel, **upload** the training data file “GoSales.csv” using either browse or drag-and-drop.
3. Under **Data Assets**, you should see your file appear.
4. Associate a Watson Machine Learning service instance with your project. If you do not have one already, please refer to the relevant section in Tutorial 1.

Step 2: Specify basic model details

1. Click on **Add to project**.
2. From the popup, choose the **Watson Machine Learning Model** option
    a. Name the model “**Multiclass Classifier”**.
    b. Feel free to leave the description box blank.
    c. Set the Machine Learning Service to the one you generated.
    d. Select Model Type: **Model Builder**.
    e. Runtime as “**Default Spark Scala 2.11**” form the dropdown menu
    f. Click the card labeled **Manual**.
    g. Click the **Create** button.

Step 3: Train the model

1. Select the data asset which you uploaded earlier (**GoSales.csv**) and click **Next**.

2. For the label column drop-down menu, select the **PRODUCT_LINE** variable.
3. For the Feature Columns, select **GENDER, AGE, MARITAL_STATUS** and **PROFESSION.**

> The *label column* is what the model will predict. *Feature columns* contain the attributes on which the model will base predictions.

4. Select the **Multiclass Classification** card.
5. Click the **Add Estimators** button to view the choices of algorithm, and select the cards labeled "**Naïve Bayes**" and "**Random Forest Classifier**", then click the **Add** button.
6. Click the **Next** button to begin the model's training.

> Compare the training results of the two algorithms. Notice that the performance of *Naïve Bayes* is rated as Poor and the Random Forest Classifier is rated as Excellent.
>
> Watson machine learning model reserves some of the training data, doesn't use it to train the model, and then uses that reserved data to evaluate how well the model gets the correct answer.
>
> To find the best solution for a given machine learning problem, you sometimes have to experiment with your training data, the model design, or the algorithms used.

7. Select the **Random Forest Classifier**, and then click **Save**.

Part 2: Deploy the trained model

6. Click the **Deployments** tab.
7. Click the **Add Deployment** button.
8. Fill in the **Name** field and select "Web Service" as the **Deployment Type**.
9. Click the **Save** button.
10. Click on your deployment name to view the deployment details page.

Part 3: Test the deployed model

4. Click the **Test** tab of the deployment details page.
5. Fill in some values from the following feature columns: **GENDER**, **AGE**, **MARITAL_STATUS**, and **PROFESSION** (Example values can be found in the data file. Ignore the other fields).
6. Click the **Predict** button.

This time around, our model has predicted likelihoods for the different product lines to be purchased by a customer with your entered values. This type of model uses one label column with multiple properties. Our next and last tutorial will be the Regression Model.

Tutorial 3 – Regression Model

Regression Model Classification is complex; our model will classify data in a continuous manner.

Make a new **Empty Project** titled “Regression”. This project will use the same data as the previous ones.

Part 1: Build and train the model

Step 1: Add training data

1. Click on the **Assets** tab.
2. In the right hand panel, **upload** the training data file “GoSales.csv” using either browse or drag-and-drop.
3. Under **Data Assets**, you should see your file appear.
4. Associate a Watson Machine Learning service instance with your project. If you do not have one already, please refer to the relevant section in Tutorial 1.

Step 2: Specify basic model details

1. Click on **Add to project**.
2. From the popup, choose the **Watson Machine Learning Model** option
    h. Name the model “**Regression Model”**.
    i. Feel free to leave the description box blank.
    j. Set the Machine Learning Service to the one you generated.
    k. Select Model Type: **Model Builder**.
    l. Runtime as “**Default Spark Scala 2.11**” form the dropdown menu
    m. Click the card labeled **Manual**.
    n. Click the **Create** button.

Step 3: Train the model

1. Select the data asset which you uploaded earlier (**GoSales.csv**) and click **Next**.
2. For the label column drop-down menu, select the **PURCHASE_AMOUNT** variable.

3. For the Feature Columns, select **GENDER, AGE, MARITAL_STATUS** and **PROFESSION.**

The *label column* is what the model will predict. *Feature columns* contain the attributes on which the model will base predictions.

4. Select the **Regression** card.
5. Click the **Add Estimators** button to view the choices of algorithm, and select the card labeled "**Gradient Boosted Tree Regression**", then click the **Add** button.
6. Click the **Next** button to begin the model's training.
7. After the training completes, click **Save**.

Part 2: Deploy the trained model

1. Click the **Deployments** tab.
2. Click the **Add Deployment** button.
3. Fill in the **Name** field and select "Web Service" as the **Deployment Type**.
4. Click the **Save** button.
5. Click on your deployment name to view the deployment details page.

Part 3: Test the deployed model

1. Click the **Test** tab of the deployment details page.
2. Fill in some values from the following feature columns: **GENDER**, **AGE**, **MARITAL_STATUS**, and **PROFESSION** (Example values can be found in the data file. Ignore the other fields).
3. Click the **Predict** button.

First, our Binary Classifier was built to predict the likelihood that a customer would buy a tent. Then, our Multiclass Classifier was built to predict which product lines our customer was most likely to buy. Now, we have built a Regression Model. The customer values we entered are considered by our machine learning model, and it has given us a prediction of how much money the customer is likely to spend. These three kinds of classifiers have served as an introduction into machine learning models, but there are others available, which are addressed in our other workshops.