Implementing Predictive Analytics with Spark in Azure Databricks

Lab 2 – Building Supervised Learning Models

Overview

In this lab, you will use Spark to build classification and regression models.

What You'll Need

To complete the labs, you will need the following:

- A web browser
- A Microsoft account
- A Microsoft Azure subscription
- A Windows, Linux, or Mac OS X computer
- Azure Storage Explorer
- An Azure Databricks cluster
- An Azure Storage account
- The lab files for this course

Note: If you have not already done so, set up the required environment for the lab by following the instructions in the Setup document for this course. Then follow the instructions in Lab 1 to provision the required Azure resources.

Building a Classification Model

The first supervised learning model you will build is a classification model that predicts whether or not a flight will be late.

Upload Source Data to Azure Storage

In this lab, you will build predictive models for flight data. Before you can do this, you must store the flight data files in the **spark** container in your Azure storage account. The instructions here assume you will use Azure Storage Explorer to do this, but you can use any Azure Storage tool you prefer.

1. In the folder where you extracted the lab files for this course on your local computer, in the **data** folder, verify that the **flights.csv** file exists. This file contains flight data that has been cleaned and prepared for modeling.

2. Start Azure Storage Explorer, and if you are not already signed in, sign into your Azure subscription.
3. Expand your storage account and the **Blob Containers** folder, and then double-click the **spark** blob container.
4. In the **Upload** drop-down list, click **Upload Files**. Then upload **flights.csv** as a block blob to a folder named **data** in root of the **spark** container.

Create a Classification Model

You will use a Notebook to create your classification model. You can choose to work with Python or Scala.

1. In your Databricks workspace, import either **Python Classification.ipynb** or **Scala Classification.ipynb**, depending on your preferred choice of language, from the **Lab02** folder where you extracted the files for this course.
2. Open the notebook you uploaded and attach it to your cluster.
3. Read the notes and run the code it contains to build a classification model.

Building a Regression Model

Having built a classification model that predicts whether or not a flight will be late, you will now build a regression model that predicts how late (or early) flights will arrive.

Upload a Jupyter Notebook

You will use a Jupyter Notebook to create your regression model. You can choose to work with Python or Scala.

1. From the **Lab02** folder in the folder where you extracted the lab files, upload **Python Regression.ipynb** or **Scala Regression.ipynb**, depending on your preferred choice of language, to your Databricks workspace.
2. Open the notebook you uploaded, attach it to your cluster, and then read the notes and run the code it contains to build a regression model.

Building a Pipeline Model

So far you have built models by calling the **fit** method on the algorithm class. Now you will use a **Pipeline** class to encapsulate the transformers and estimators used by your model.

Upload a Jupyter Notebook

You will use a Jupyter Notebook to create your pipeline model. You can choose to work with Python or Scala.

1. From the **Lab02** folder in the folder where you extracted the lab files, upload **Python Pipeline.ipynb** or **Scala Pipeline.ipynb**, depending on your preferred choice of language, to your Databricks workspace.
2. Open the notebook you uploaded, attach it to your cluster, and then read the notes and run the code it contains to build a pipeline model.

Working with Text Features

All of your models to this point have used numeric columns for features. Now you will work with text features.

Upload Source Data to Azure Storage

In this exercise, you will explore tweet data. Before you can do this, you must store the data file in the shared storage used by your cluster.

1. In the folder where you extracted the lab files for this course on your local computer, in the **data** folder, verify that the **tweets.csv** file exists. This file contains tweets that have been classified as positive or negative.
2. Use Azure Explorer (or your preferred Azure storage tool) to upload **tweets.csv** as a block blob to a folder named **data** in root of the **spark** container in your Azure storage account.

Upload a Jupyter Notebook

You will use a Jupyter Notebook to create a classification model for text features. You can choose to work with Python or Scala.

1. From the **Lab02** folder in the folder where you extracted the lab files, upload **Python Text Analysis.ipynb** or **Scala Text Analysis.ipynb**, depending on your preferred choice of language, to your Databricks workspace.
2. Open the notebook you uploaded, attach it to your cluster, and then read the notes and run the code it contains to build a classification model for text features.

Clean Up

If you intend to proceed straight to the next lab, skip this section. Otherwise, follow the steps below to delete your cluster and avoid being charged for cluster resources when you are not using them.

Delete the Resource Group

1. Close the browser tab containing the databricks workspace if it is open.
2. In the Azure portal, view your **Resource groups** and select the resource group you created for your databricks workspace. This resource group contains your databricks workspace and your storage account.
3. In the blade for your resource group, click **Delete**. When prompted to confirm the deletion, enter the resource group name and click **Delete**.
4. Wait for a notification that your resource group has been deleted.
5. After a few minutes, a second resource group containing the resources for your cluster will automatically be deleted.
6. Close the browser.