

## 3. Data Mining with Azure ML Studio

### 3.1 Getting Started with Azure Machine Learning Studio

In this section we will be familiarizing ourselves with Azure Machine Learning (“ML”) Studio. We will create a dedicated storage account for our experiment and a workspace within our account, learn how to access the workspace from the Azure Portal, and finally create our first experiment. In order to get started and begin your first exercise with Azure ML, you must sign up for a free trial. You can register at: <http://azure.microsoft.com/en-us/pricing/free-trial/>

#### 3.1.1 Exercise: Creating an Azure Machine Learning Studio Workspace

Once you have a dedicated Azure storage account, you can create an Azure ML Studio workspace.

1. Create a new Azure ML Studio workspace by selecting:  
**+New > Data Services > Machine Learning > Quick Create** (Figure: 3.1, 3.2) .

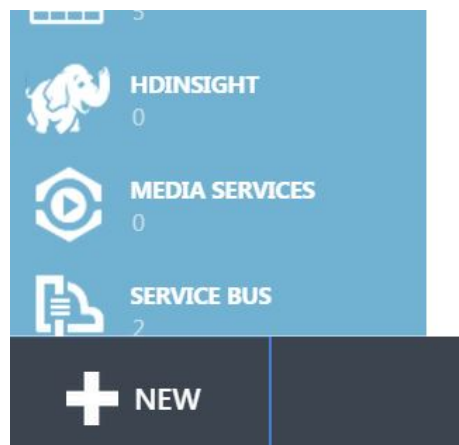


Figure 3.1: Create a new workspace

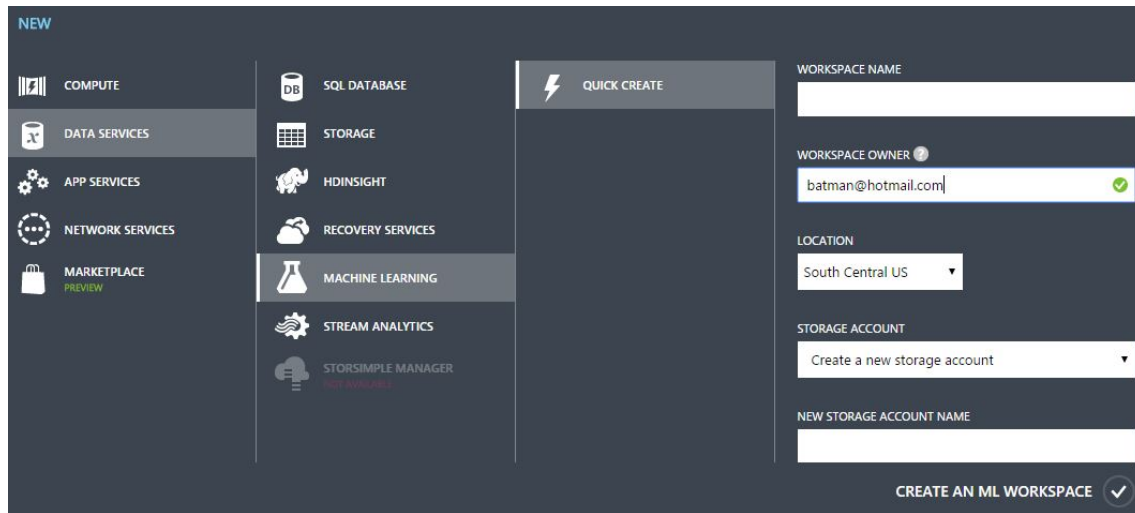


Figure 3.2: Create a new workspace

2. In the **Workspace Name** box, assign a globally unique name.
3. In the **Workspace Owner** box, input the administrative email for your Azure account, preferably a hotmail account.
4. In the **Storage Account** dropdown, select “Create a new storage account”.
5. In the **New Storage Account Name**, give your blob storage a globally unique name.

Click the check mark once the credentials have been populated to send off a workspace request to Azure. The workspace will take at least two minutes to setup. Accidentally deleting the blob storage associated with your Azure ML workspace will corrupt the workspace and render it unusable.

**Tip** You can invite others to collaborate in your workspace by adding them as users to the account under **Settings**. You can also copy and paste experiments across workspaces.

### 3.1.2 Exercise: Accessing your Azure Machine Learning Workspace

You may now access your Azure ML workspace.

1. Within the Azure Portal, select **Machine Learning** (Figure: 3.3).

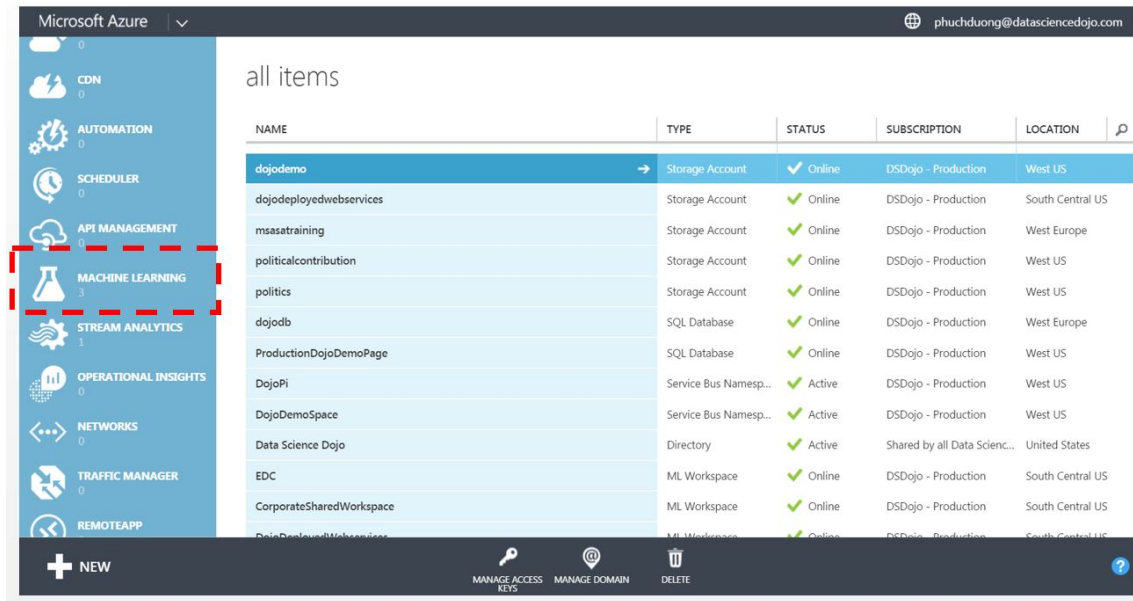


Figure 3.3: Access your machine learning workspace

2. Select the workspace that you just created in Exercise: Creating an Azure Machine Learning Studio Workspace.
3. Select “Access your Workspace” (Figure: 3.4) A new window will appear.

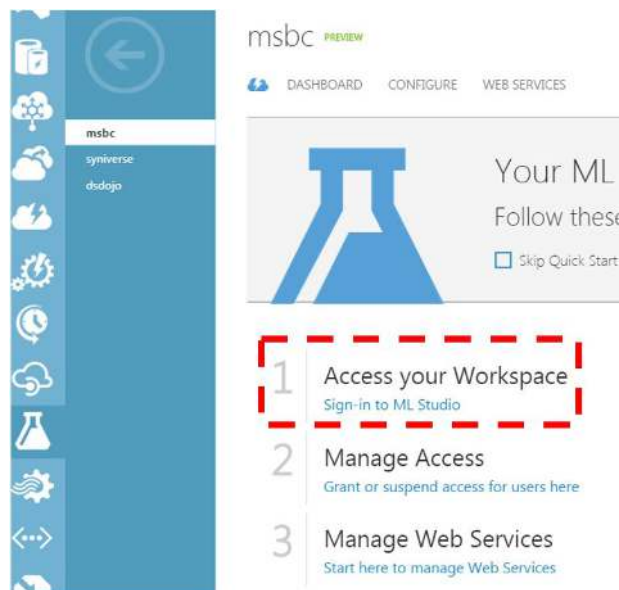


Figure 3.4: Access your workspace

### 3.1.3 Exercise: Creating your First Experiment

Data Science is an interdisciplinary art and science. It borrows terms from other disciplines, especially the sciences. In this tradition, a project in data science is called an experiment.

1. To create a new experiment, select **+New > Experiment > “Blank Experiment”** (Figure: 3.5) .

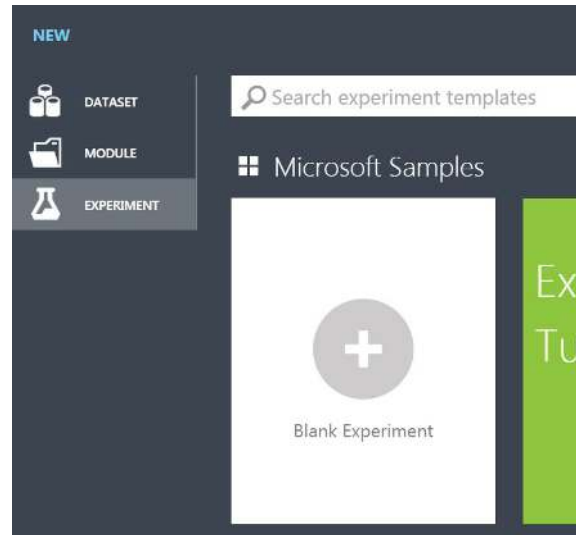


Figure 3.5: Create a new experiment

2. Name your experiment in the “Experiment Name” field.

We aren’t ready to save our experiment yet, so for now we will move on.

## 3.2 Methods of Ingress and Egress with Azure Machine Learning Studio

### 3.2.1 Exercise: Reading a Dataset from a Local File

The first dataset we will be using is the go-to database when getting started with data science. The data describes features of an iris plant in an attempt to predict its class.

To retrieve the dataset, Google “UCI Iris Data” or go to:

<http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data>

Notice how commas differentiate each value. This allows us to know that the elements can be read as comma separated values (“CSV”). Excel files and delimited text files can be read as CSV as well. Also notice that the data does not have headers. The model will eventually require headers but we will define these later on.

1. Download and save the text as a CSV file. For example “filename.csv”.
2. In Azure ML Studio, select **+New > Dataset > From Local File**.
3. Please note that by default, Azure ML ships with a dataset called “Iris Two Class Data”. To avoid confusion, give your dataset a unique name, then import.
4. To verify that your data has been imported, go into any experiment and look under the directory **Saved Datasets** (Figure: 3.6). You should see the name you chose for your data listed.

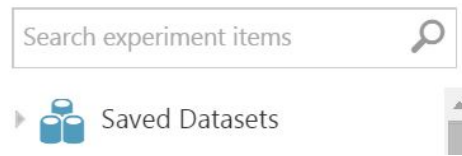


Figure 3.6: Saved dataset directory

5. Now that your experiment has a module in it, you can now save your experiment. Select “Save As” on the menu at the bottom of your screen (Figure: 3.7).



Figure 3.7: Save the experiment

### 3.2.2 Exercise: Reading a Dataset from a URL

1. To begin, use the search bar to find the **Reader** module within your experiment. Drag and drop the module from the menu on the left (Figure: 3.8).



Figure 3.8: Search for the Reader module

2. In the **Reader** settings for “Please specify data source” select “Http”.
3. In the “URL” box, enter the URL of the iris data set:  
`http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data`.
4. In the “Date format” drop down, select “CSV”.
5. Leave “CSV or TSV has header row” unchecked (Figure: 3.9).

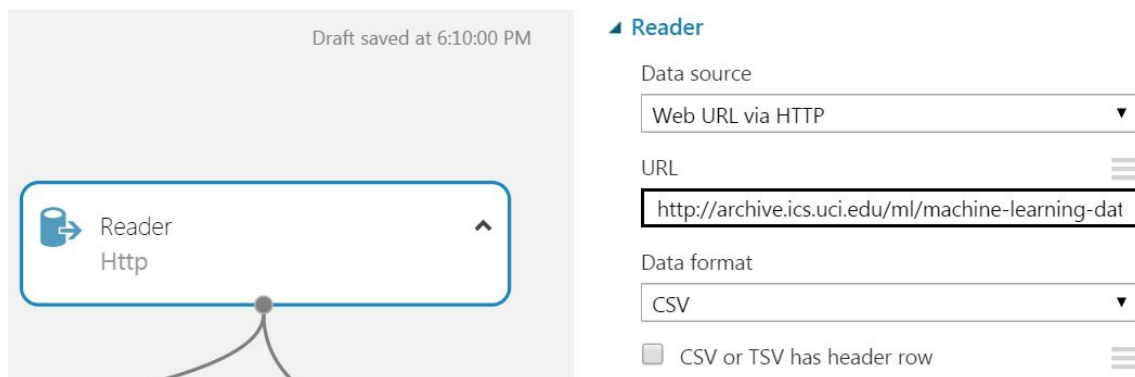


Figure 3.9: Reader module settings

6. Select **Run** to import and parse the experiment (Figure: 3.10).



Figure 3.10: Run the experiment

7. In order to preserve the dataset, we must save our work. Save the output of your experiment by right-clicking the bottom middle node of the **Reader** module again. Select “Save as Dataset”. Please note that by default, Azure ML ships with a dataset called “Iris Two Class Data”. To avoid confusion, give your dataset a unique name, then import.
8. To verify that your dataset has been successfully imported, go into any experiment and look under the directory **Saved Datasets**. You should see the name you chose for your data listed.

### 3.2.3 Exercise: Reading a Dataset from Azure Blob Storage

1. To begin, use the search bar to find the **Reader** module within your experiment. Drag and drop the module from the menu on the left.
2. For the required fields, input the information from Table: 6.2.

Required Field	Input
<b>Data source</b>	Azure Blob Storage
<b>Authentication type</b>	Account
<b>Account name</b>	dojoattendeestorage
<b>Account key</b>	aKQOxU3As1BsS3yT2bh HkJ/icCICJPpL1tdWKxQ+tP BNk6DbykV4qd3HGIFPZ N/3TdiUHuM/Quk 9DPUEQu7M8A==
<b>Path to container, directory or blob</b>	datasets/iris.three.class.csv
<b>Blob file format</b>	CSV
<b>File has header row</b>	Unchecked

Table 3.1: Azure Blob Storage Log-In Details

**Tip** Note that “dojoattendeestorage” is the container. Containers contain blobs which are essentially files in the Azure Cloud itself. For those who are familiar with web development, this is equivalent to an FTP.

Figure: 3.11 depicts a sample of what your **Reader** module will look like after all of the above steps have been followed.

▲ **Reader**

Data source  
Azure Blob Storage ▼

Authentication type  
Account ▼

Account name ☰  
dojoattendeestorage

Account key ☰  
.....

Path to container, directory or blob ☰  
datasets/iris.three.class.csv

Blob file format  
CSV ▼

File has header row ☰

Figure 3.11: Reader module settings

3. Select **Run** to import and parse the experiment.
4. In order to preserve the dataset, we must save our work. Save the output of your experiment by right-clicking the bottom middle node of the **Reader** module again. Select “Save as Dataset”. Please note that by default, Azure ML ships with a dataset called “Iris Two Class Data”. To avoid confusion, give your dataset a unique name, then import.
5. To verify that your dataset has been successfully imported, go into any experiment and look under the directory **Saved Datasets**. You should see the name you chose for your data listed.

### 3.2.4 Exercise: Writing a Dataset to Azure Blob Storage

1. Go into the directory **Saved Datasets** and drag any dataset into your workspace.
2. Search for the **Writer** module in the search box. Drag the module into your workspace and connect it to your dataset.
3. For the required fields, input the information from (Table: 3.2).

Required Field	Input
<b>Please specify data destination</b>	Azure Blob Storage
<b>Please specify authentication type</b>	Account
<b>Azure account name</b>	dojoattendeestorage
<b>Azure account key</b>	aKQOxU3As1BsS3yT2b hHkJ/icCICJPpL1tdWKxQ+tPB Nk6DbykV4qd3HGIFPZN/3 TdiUHuM/Quk9DPueQu7M8A==
<b>Path to blob beginning with container</b>	attendee-uploads/<file-name>.csv
<b>Azure blob storage write mode</b>	Overwrite
<b>Azure blob storage write mode</b>	CSV
<b>Write blob header row</b>	Unchecked

Table 3.2: Azure Blob Storage Log-In Details


 Normally when prompted for the “Path to blob beginning with container”, you can choose any file name you would like. However, since many people will be writing to this blob during this exercise, do not name the file iris.csv. Name the file with your first initial, last name, then iris as one word (i.e. John Smith or jSmithiris.csv).



Figure: 3.12 depicts a sample of what your **Writer** module will look like after all of the above steps have been followed.

▲ **Writer**

Please specify data destination

Azure Blob Storage ▼

Please specify authentication type

Account ▼

Azure account name ☰

dojoattendeestorage

Azure account key ☰

.....

Path to blob beginning with container ☰

attendee-uploads/pDuongIris.csv

Azure blob storage write mode ☰

Overwrite ▼

File format for blob file

CSV ▼

Figure 3.12: Sample Writer module settings

### 3.3 Visualizing, Exploring, Cleaning, and Manipulating Data

#### 3.3.1 About the Data

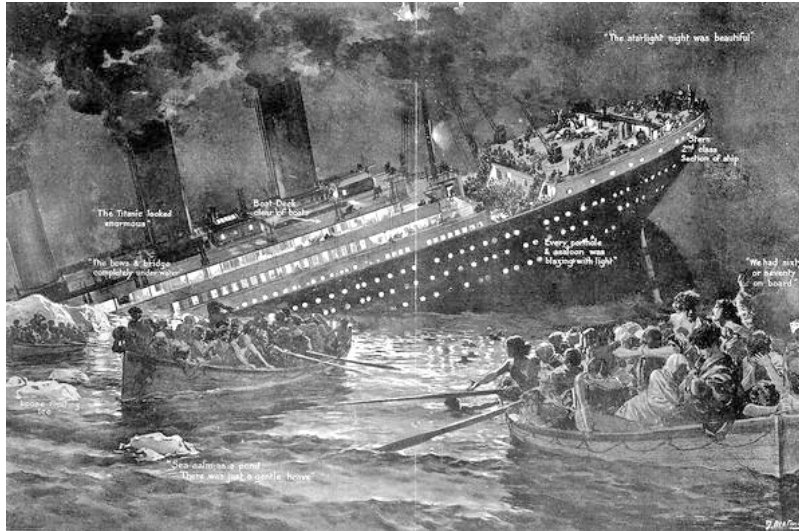


Figure 3.13: Illustration of the sinking of the Titanic. Source: National Maritime Museum

For most of this chapter we will be working with the Titanic dataset. The Titanic data is a good beginner's dataset to start learning how to data mine. The sinking of the RMS Titanic occurred in 1912 and is one of the most infamous shipwrecks in history. 1,502 out of 2,224 passengers were killed in the tragedy and the incident caused an international backlash for ship safety reform. Although the mass loss of life is mainly attributed to the lack of life vessels and other elements of chance, some groups were more likely to survive than others. We will use the power of machine learning to uncover which types of individuals were more likely to survive. To begin, we will first procure the dataset.

#### 3.3.2 Exercise: Obtaining the Titanic sample data

You will be reading in the dataset from our online GitHub repository by using a **Reader** module.

1. Drag and drop a **Reader** module from the menu on left (Figure: 3.14).



Figure 3.14: Search for the Reader module

2. Set the **Reader** module with the settings found in 3.3.

Field	Setting and Inputs
<b>Data Source</b>	Web URL via HTTP
<b>URL</b>	<code>https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.csv</code>
<b>Data format</b>	CSV
<b>CSV or TSV has header row</b>	Checked

Table 3.3: Reader module settings

Reader

Data source  
Web URL via HTTP

URL  
`https://raw.githubusercontent.com/datasciencedojo/data`

Data format  
CSV

CSV or TSV has header row

Figure 3.15: Reader module settings

Figure: 3.16 depicts a sample of what your **Reader** module will look like after all of the above steps have been followed.

Reader  
AzureBlob-Titanic

Reader

Data source  
Azure Blob Storage

Authentication type  
Account

Account name  
`dojoattendeestorage`

Account key  
.....

Path to container, directory or blob  
`datasets/titanic.csv`

Blob file format  
CSV

File has header row

Figure 3.16: Reader module settings

3. Select **Run** to execute the import and parse.
4. In order to preserve the dataset, we must save our work. Save the output of your experiment by right-clicking the bottom middle node of the **Reader** module again. Select “Save as Dataset”.

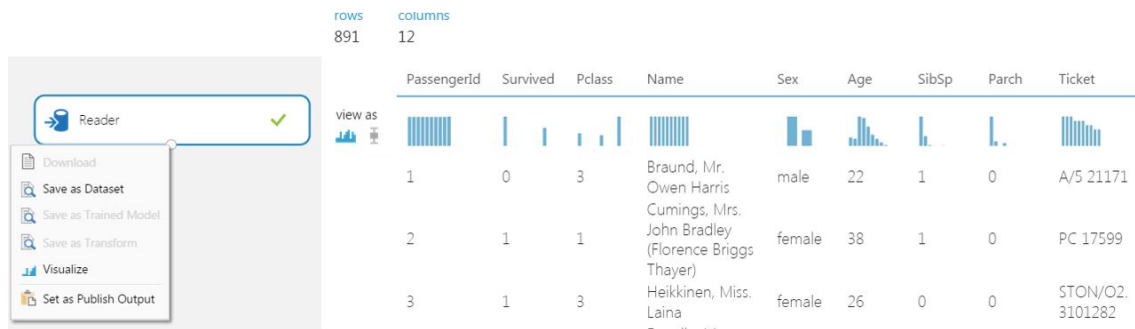


Figure 3.17: Visualize the Reader module

- To verify that your dataset has been successfully imported, go into any experiment and look under the directory **Saved Datasets**. You should see the name you chose for your data listed.

### 3.3.3 Titanic Dataset Key

The Titanic dataset is made up of qualitative and quantitative information. The Titanic key describes the meaning variables and their corresponding values. The key can also be found at:

<https://www.kaggle.com/c/titanic/data>.

#### Variable Descriptions

Column Name	Meaning	Notes
survival	Survival	(0 = No; 1 = Yes)
pclass	Passenger Class	(1 = 1st; 2 = 2nd; 3 = 3rd)
name	Name	
sex	Gender	
age	Age	
sibsp	Number of Siblings/Spouses Aboard	
parch	Number of Parents/Children Aboard	
ticket	Ticket Number	
fare	Passenger Fare	In 1910 USD
cabin	Cabin	
embarked	Port of Embarkation	(C = Cherbourg; Q = Queenstown; S = Southampton)

#### Special Notes

- Pclass is a way to infer socio-economic status (SES)  
1st Upper; 2nd Middle; 3rd Lower
- Age is in Years; age is fractional if the passenger age is less than one  
If the age is Estimated it is in the form “xx.5”
- With respect to the family relation variables (i.e. sibsp and parch) some relations were ignored. The following are the definitions used for sibsp and parch.
  - Sibling: Brother, sister, stepbrother, or stepsister of the passenger
  - Spouse: Husband or wife of the passenger (mistresses and fiances Ignored)
  - Parent: Mother or father of the passenger
  - Child: Son, daughter, stepson, or stepdaughter of the passenger
- Other family relatives excluded from this study include cousins, nephews, nieces, aunts, uncles, and in-laws. Some children travelled only with a nanny, therefore parch=0 for them.

In addition, some passengers travelled with close friends or neighbors in a village, however, the definitions do not support such relations.

### 3.3.4 Exercise: Casting Columns

Although the Titanic dataset contains categorical data types, by default Azure will treat them as sequential numbers. Therefore we must tell Azure which columns are categorical.

1. Go into **Saved Datasets** and find the Titanic dataset you just obtained. Drag the dataset into your experiment's workspace (Figure: 3.18)

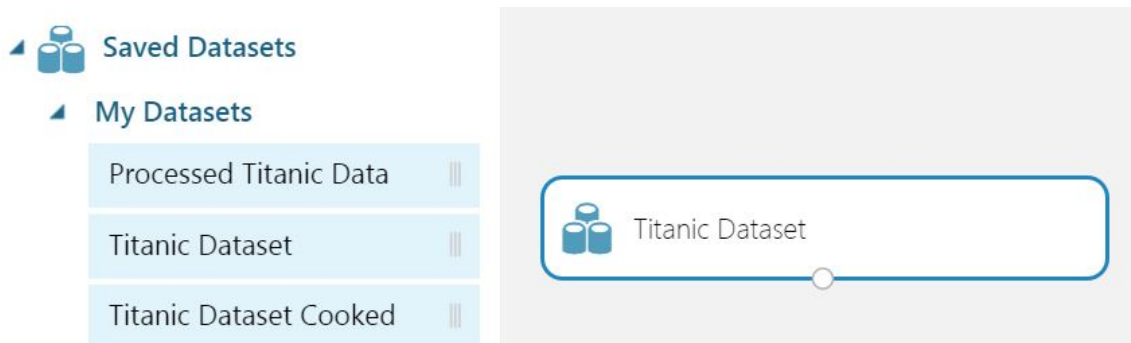


Figure 3.18: Drag the Titanic Dataset into the workspace

2. Right-click on the bottom center node of the dataset. Select “Visualize” to see the output. Verify that it looks like Figure: 3.20.

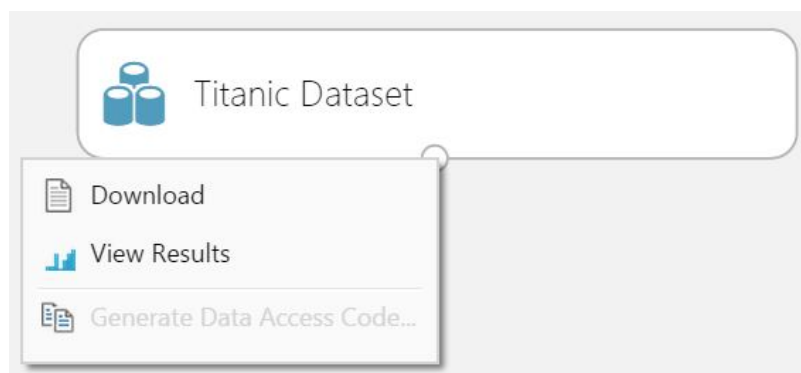


Figure 3.19: Visualize the Titanic Dataset



Figure 3.20: Visualize the Titanic Dataset

3. After verifying the output, we will cast categorical values to the corresponding columns. To begin, search for the **Metadata Editor** in the left menu and drag the module into the workspace.
4. Connect the **Metadata Editor** to the dataset and launch the “column selector” within the editor.
5. Select “Launch column selector”. For the box of chosen values, add “Survived”, “Sex”, “Pclass”, “Embarked”, and “PassengerId”. Leave all of the other fields unchanged (Figure: 3.21).

## Select columns

Allow duplicates and preserve column order in selection

**Begin With**

**Include**

Survived ✕ Sex ✕ Pclass ✕ Embarked ✕  
PassengerId ✕

Figure 3.21: Column selector settings

6. After applying your settings in the “column selector”, change the “Categorical” field to “Make Categorical” in the **Metadata Editor**. Keep all other fields as they are (Figure: 3.22).