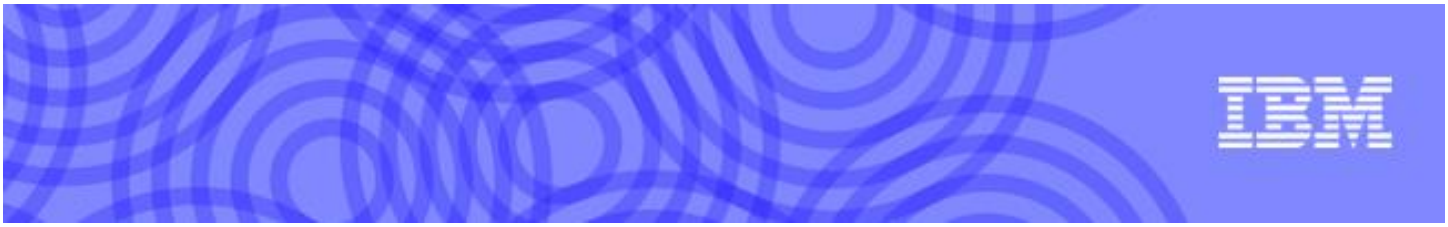


IBM Watson Machine Learning - Sample1 Deployment

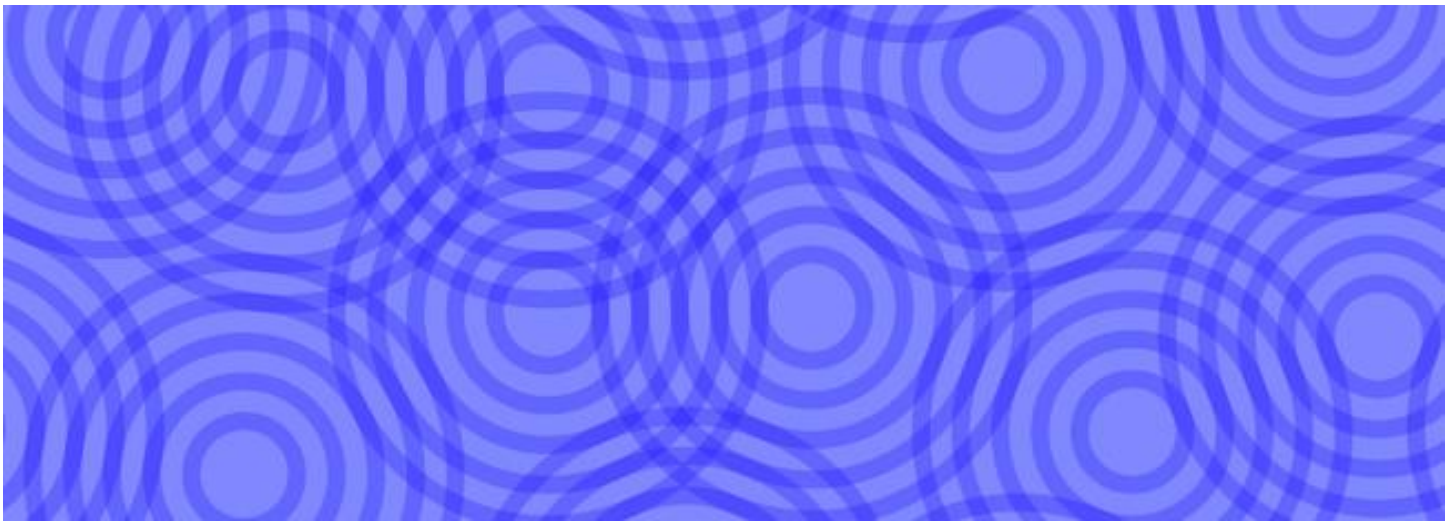


Table of Contents

1 Document Purpose 3

2 Prerequisites 4

2.1 Bluemix 4

2.2 Cloud Foundry 4

2.3 Node.js 5

2.4 Downloading Sample 1 6

3 IBM SPSS Modeler 7

4 Bluemix Application and Service 9

4.1 Creating an application 9

4.2 Creating the service 11

5 Customizing files 14

5.1 \app\manifest.yml 14

5.2 \app\app.js 14

5.3 \app\public\js\app.js 14

5.4 \app\public\js\srv.js 15

6 Porting source code 16

6.1 Node Package Manager 16

7 Uploading SPSS Models 18

8 Real-time scoring the sample 20

1 Document purpose

This document outlines how to configure and run IBM Machine Learning Service Sample1 (https://github.com/pmservice/predictive-modeling-samples).

- This document is meant to compliment the current documentation associated with Sample1.
- This document interfaces with an IBM SPSS Modeler stream, but the focus is on Bluemix. It is assumed that the reader has a basic understanding of predictive analytics used in IBM SPSS Modeler.
- While there are many supported runtimes in Bluemix (Liberty for Java, Python, Ruby, ASP.NET, Swift, Tomcat, etc.), this document will use SDK for Node.js.
- Other operating systems are supported but this example uses Microsoft Windows.

2 Prerequisites

2.1 Bluemix

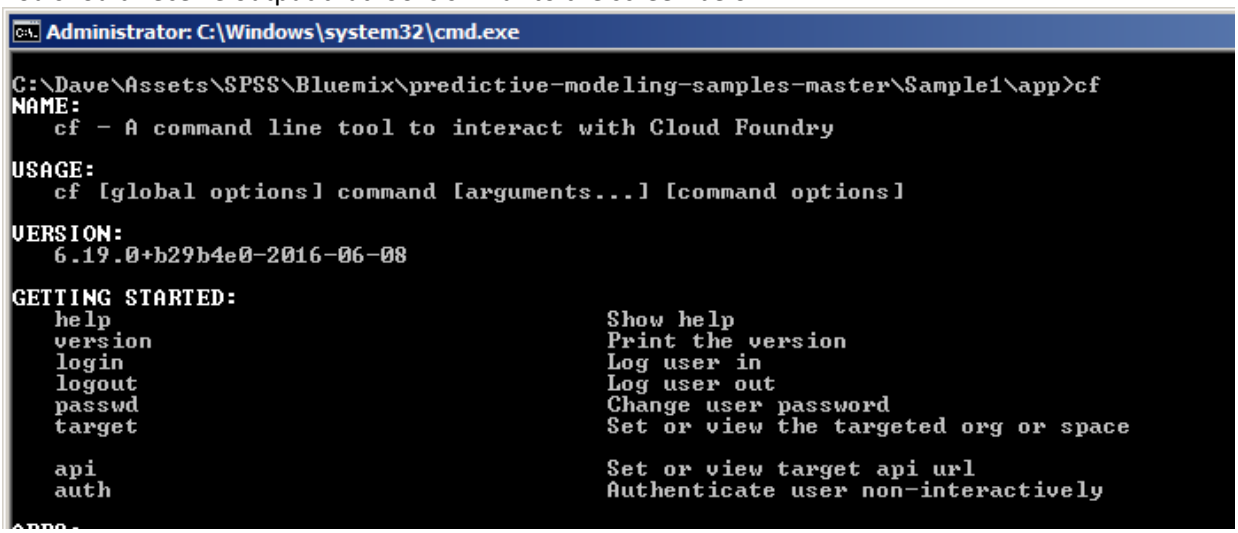
1. Ensure that you have a valid Bluemix account.

2.2 Cloud Foundry

2. Install the Cloud Foundry (CF) Command Line Interface (https://github.com/cloudfoundry/cli/releases).
3. Ensure that the tool is running properly by typing the following from a command prompt:

   cf

   You should receive output that looks similar to the screen below.



   Note: You may have to reboot your system in order for it to locate the "cf" executable. Do not proceed if the "cf" command does not run.

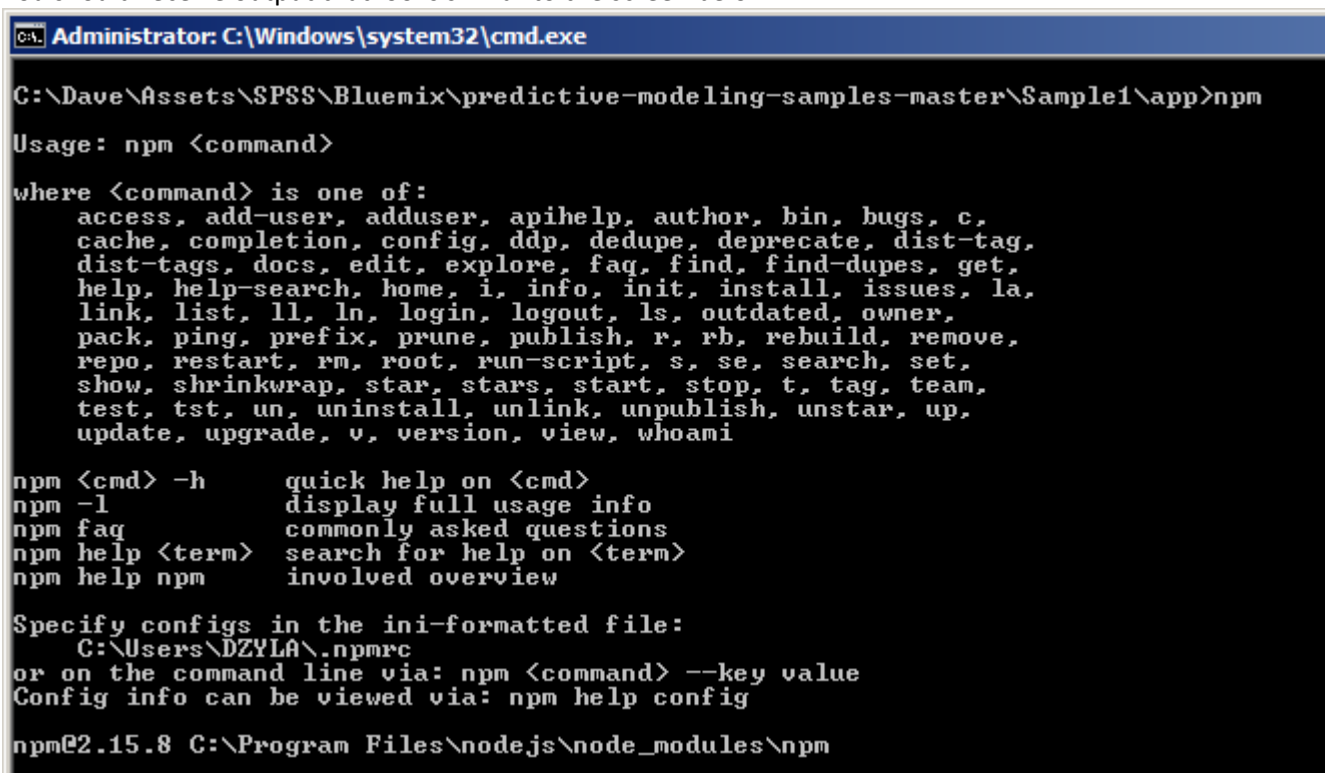
2.3 Node.js

Node.js is required to download supporting code from Bluemix to the local development environment. See https://nodejs.org/en/about/ for more information.

1. Download and install Node.js from https://nodejs.org/en/download/.
2. Validate that Node.js is running properly by typing the following from a command prompt:

   npm

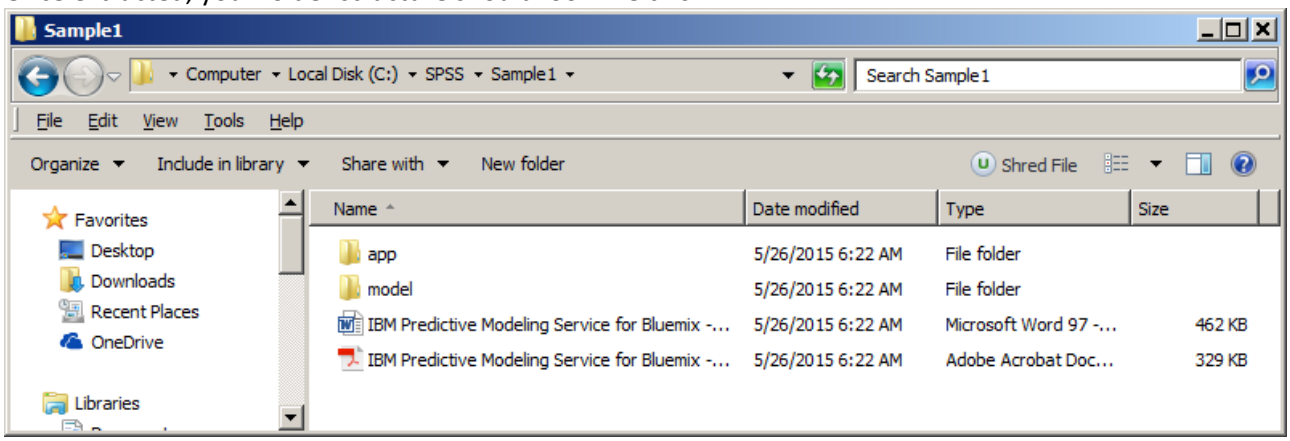
   You should receive output that looks similar to the screen below.



   Note: You may have to reboot your system in order for it to locate the “npm” executable. Do not proceed if the “npm” command does not run.

2.4 Downloading Sample 1

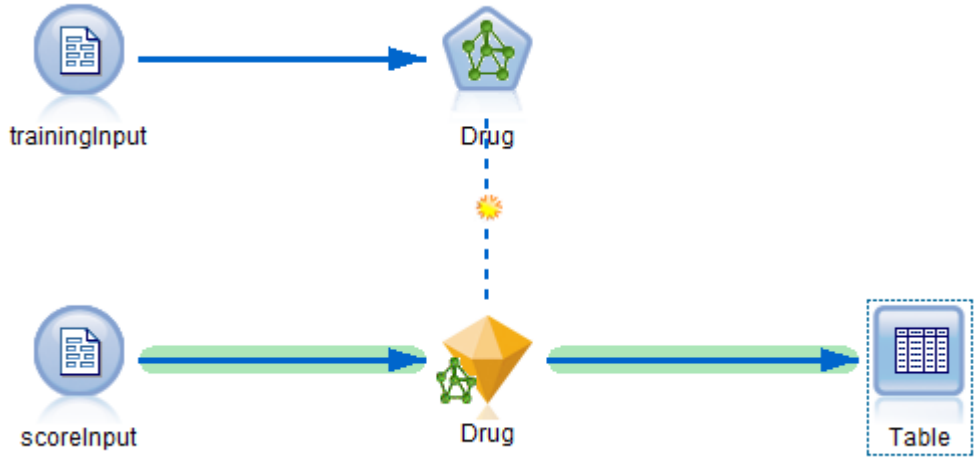
1. Download Sample 1 from https://github.com/pmservice/predictive-modeling-samples.
2. Extract the contents to a working directory. This example uses *C:\SPSS*.
   Once extracted, your folder structure should look like this:



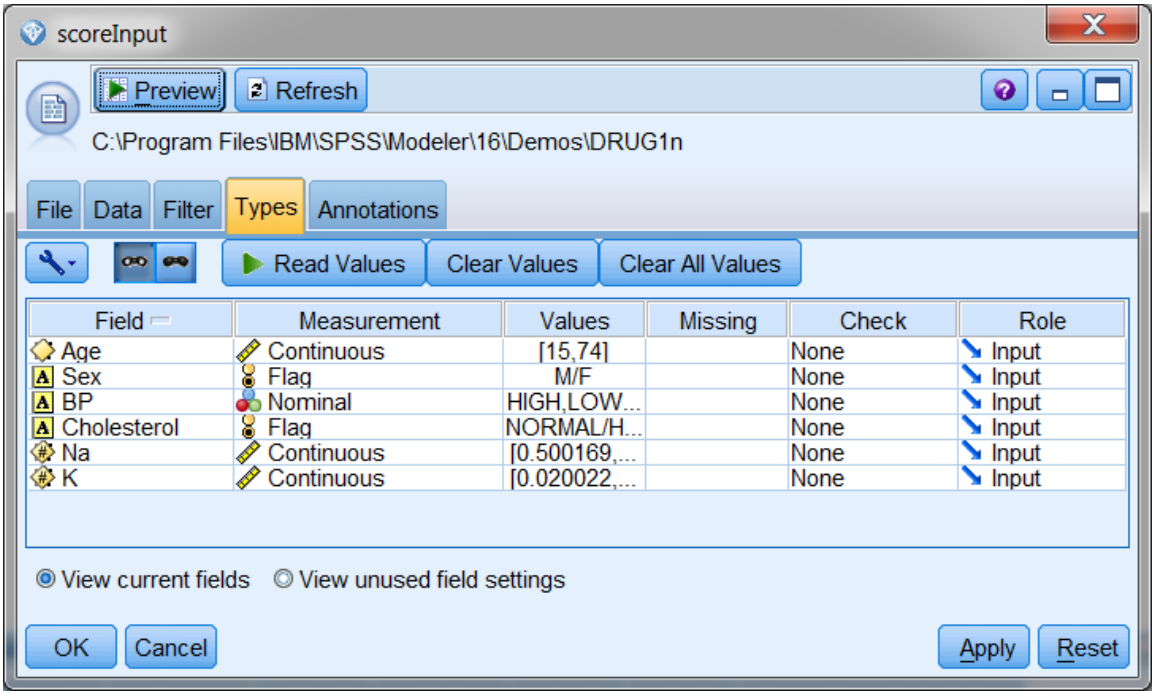
3 IBM SPSS Modeler

The sample comes with an SPSS Modeler stream which can be found at *C:\SPSS\Sample1\model\Drug1n.str*.

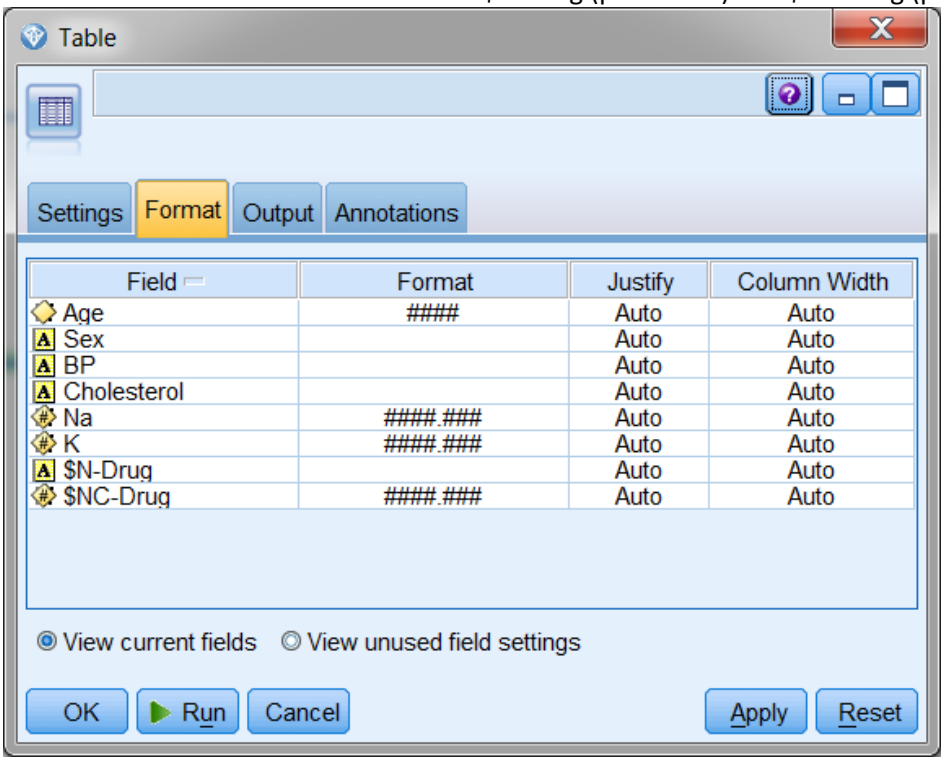
It is important to note that a scoring branch has been set in the stream. Any model used in the Bluemix Machine Learning node must have a default scoring branch set. This can simply be done by right-clicking the terminal node and selecting **Use as scoring branch**.



The Bluemix application will have to provide the same data items used in the input node. In this case it is the *scoreInput* node.



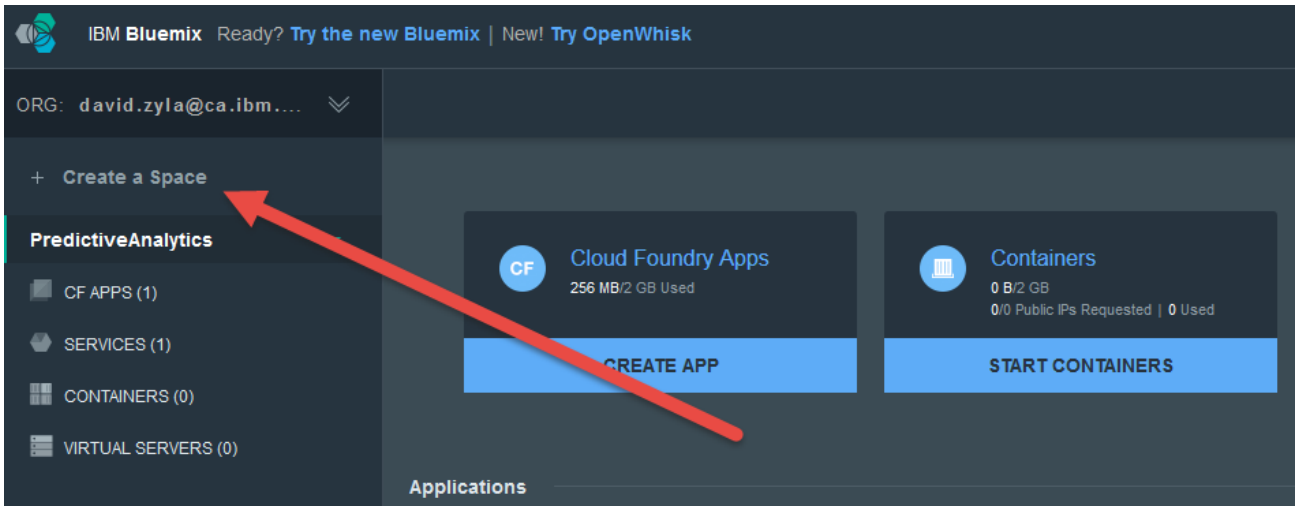
Bluemix will return scored results in the $N-Drug (prediction) and $NC-Drug (prediction confidence) columns.



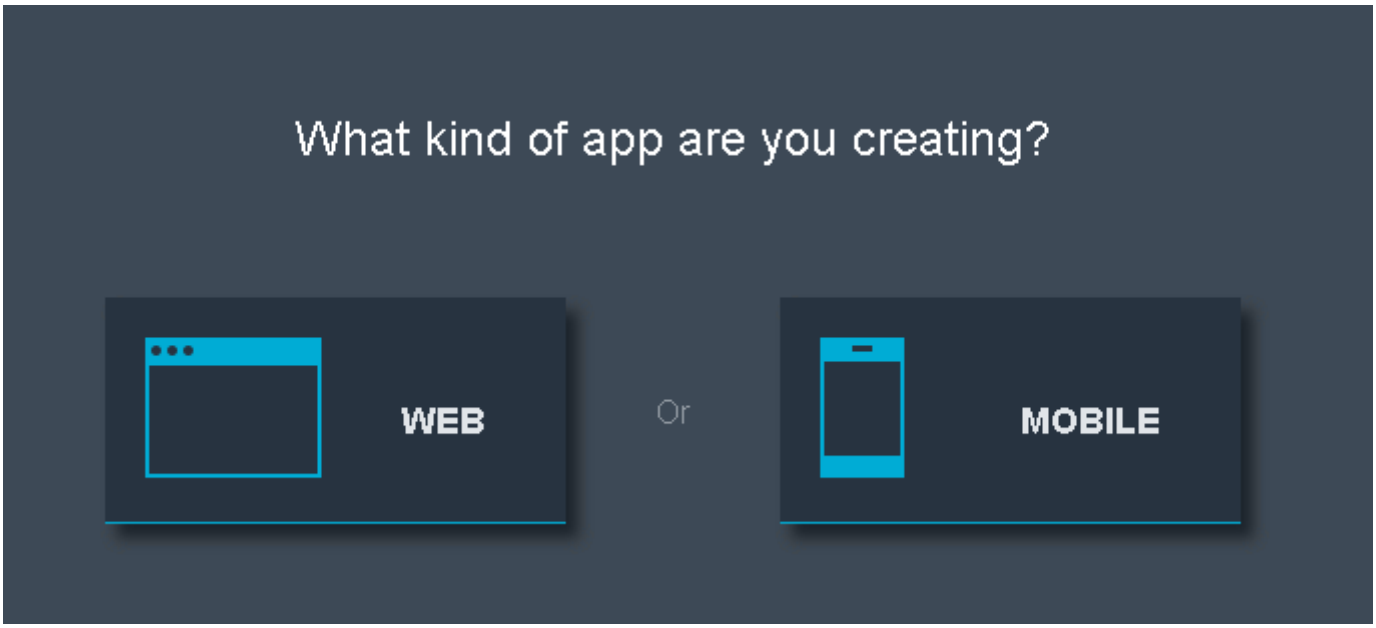
4 Bluemix application and service

4.1 Creating an application

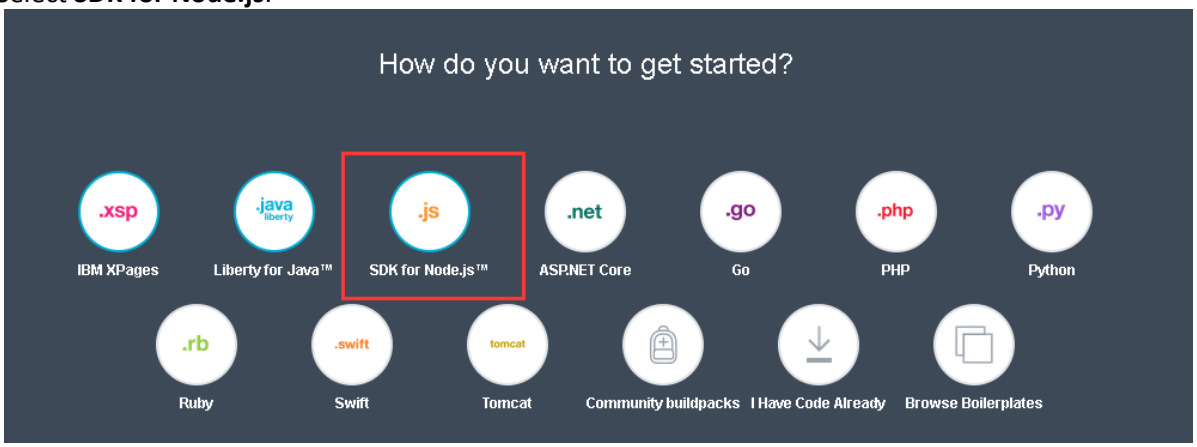
1. Log in to Bluemix.
2. If you don't already have a space created, create one now. Name the space however you like, but make note of the name for future reference.



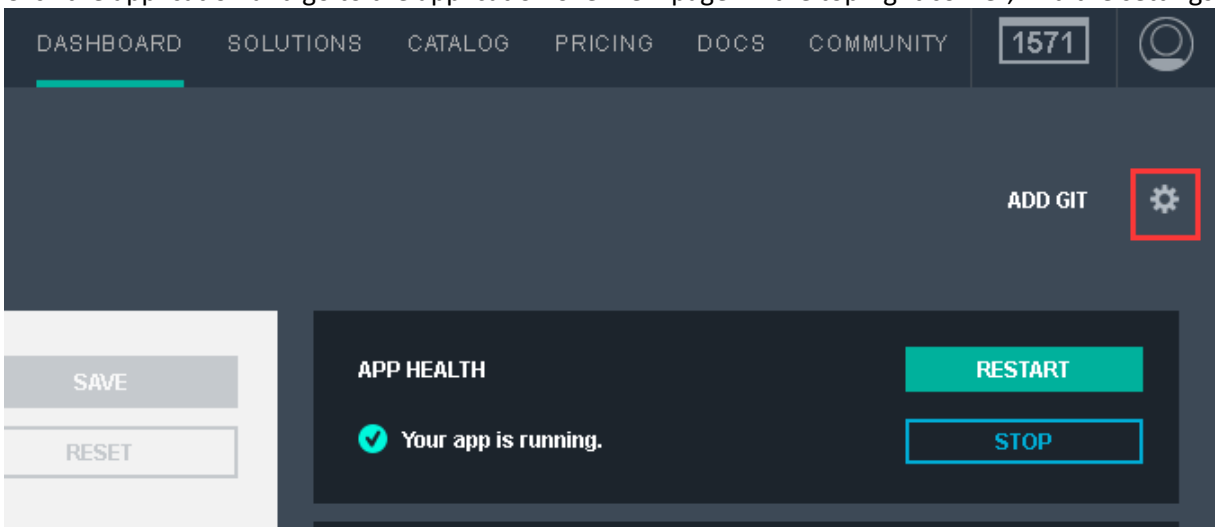
3. After creating the new space, click it from the left side of your screen.
4. On the top of the page, click **Create app**.
5. Choose **WEB**.



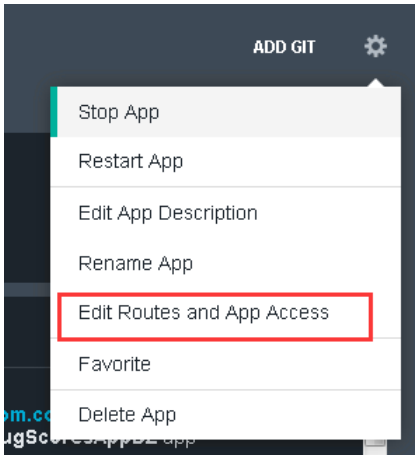
6. Select **SDK for Node.js**.



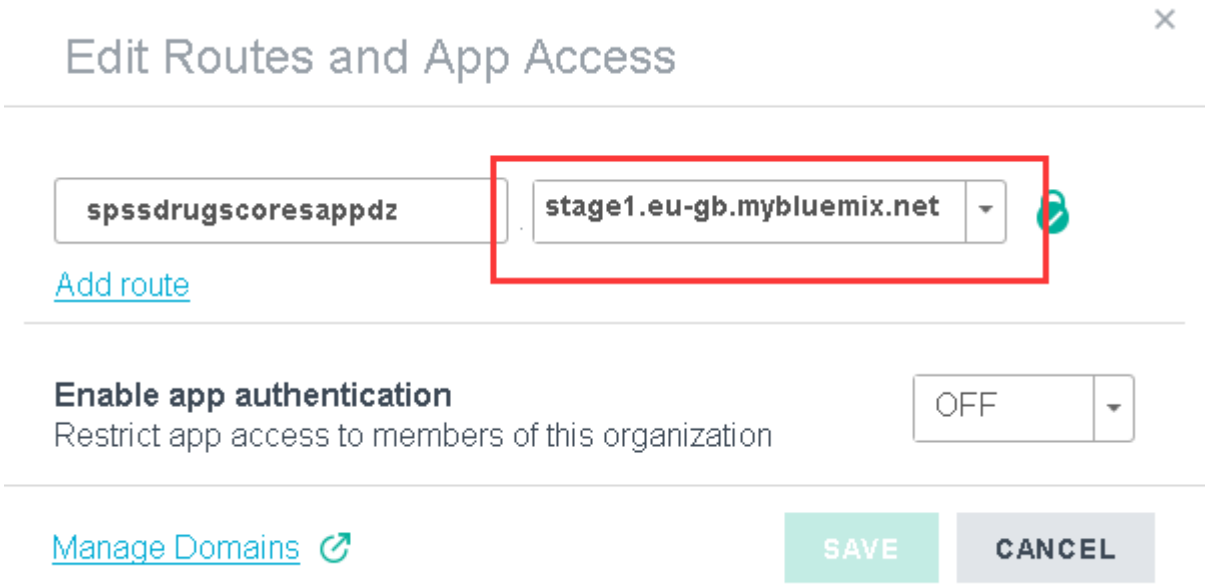
7. Name the new application however you'd like. This example uses the name **SPSSDrugScoresAppDZ**. We recommend using a short name.
8. Click **Create**. This will stage and create the application. This process may take a minute or two to complete.
9. After the application is created, go back to the dashboard. You should see the application you just created (for example, **SPSSDrugScoresAppDZ**).
10. Click the application and go to the application overview page. In the top right corner, find the settings button.



11. Click it and select **Edit Routes and App Access**.



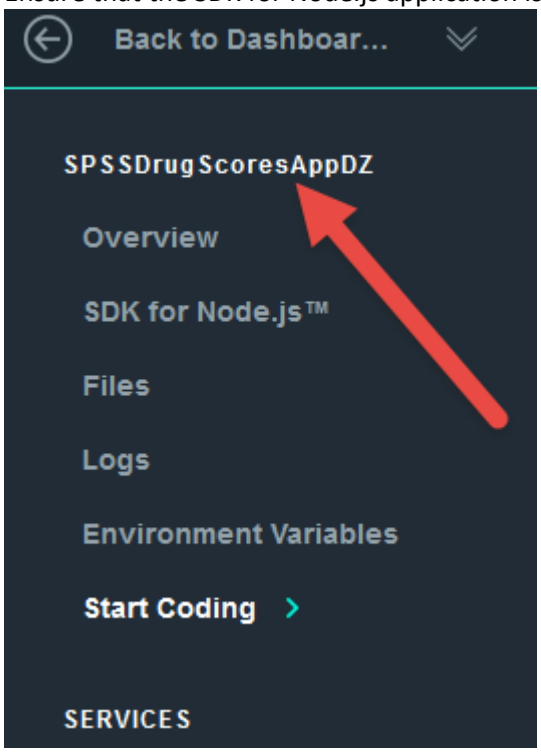
12. The second selection box shows this app's domain name.



13. Make note of the Space, App Name, Host, and Domain, as they will be referred to later on.

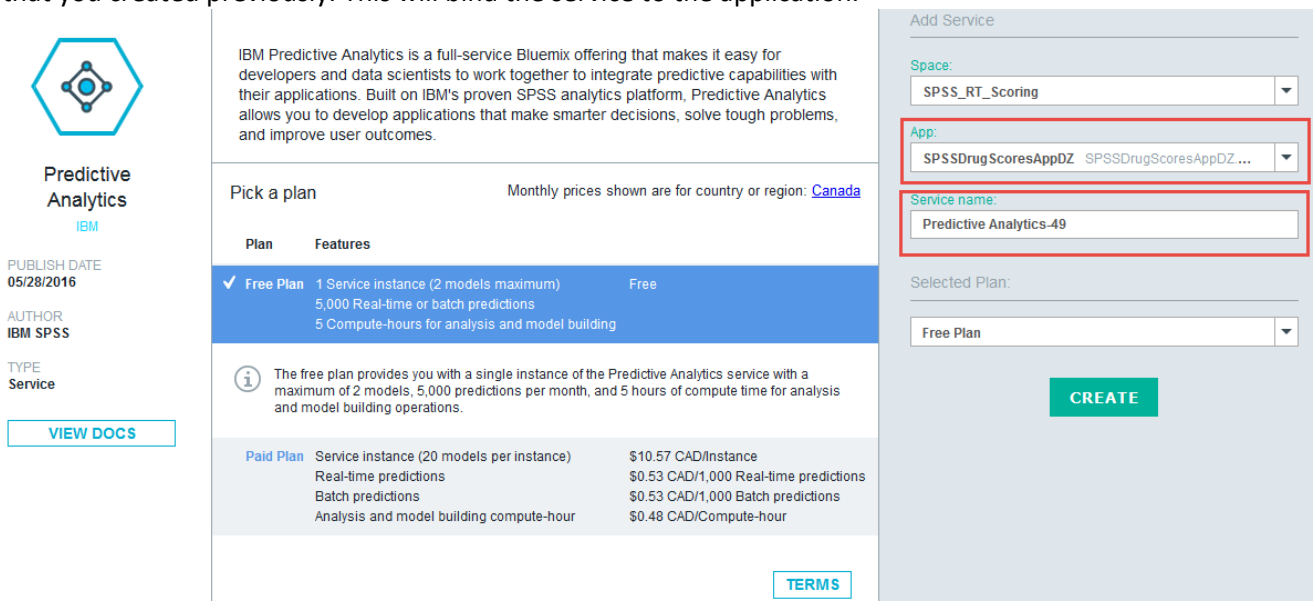
4.2 Creating the service

1. Ensure that the SDK for Node.js application is running and is selected on the left side of the page.

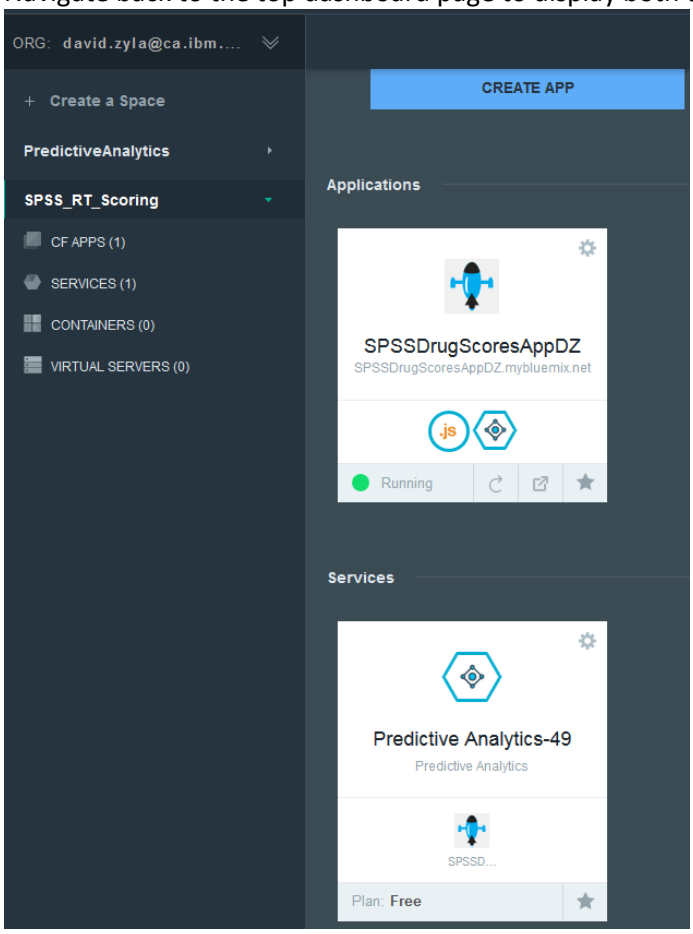


2. Click **Catalog** from the upper toolbar.
3. To filter the number of services displayed, select **Data and Analytics** under **Services** on the left side of the screen.
4. Click **IBM Watson Machine Learning**.

5. On the **Add Service** screen, ensure that the **App** drop down is changed from **Leave Unbound** to the application that you created previously. This will bind the service to the application.

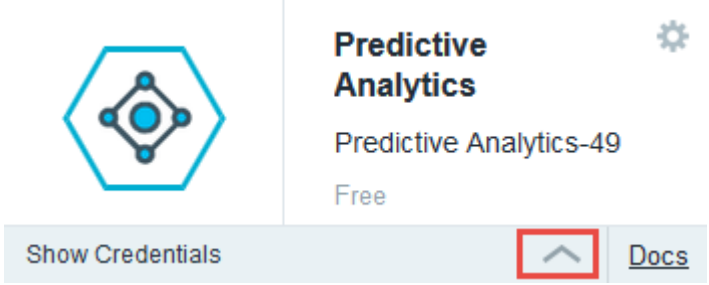


6. Make note of the App and Service names.
7. Click **Create**. If prompted to re-stage, select **Yes**.
8. Navigate back to the top dashboard page to display both the application and service.

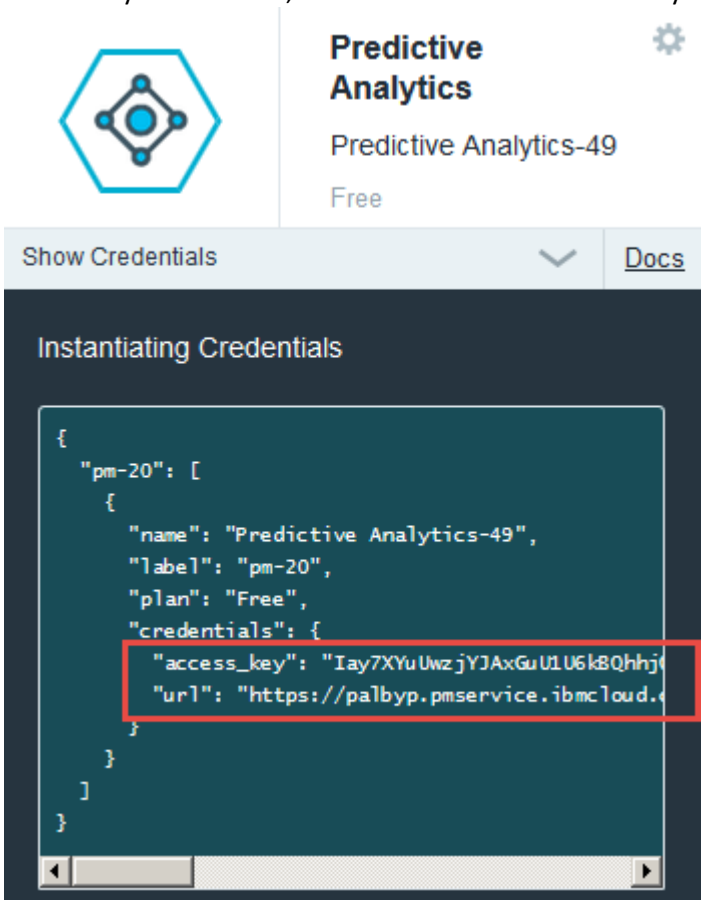


9. Click the application (**SPSSDrugScoresAppDZ**, for example).

10. When we score the input data in this example, credentials must be supplied to Bluemix at runtime. Click the up arrow to show the credentials.



11. In the fly-out window, make note of the the access key and URL. These will be needed later.



“access_key":
"Iay7XYuUwzjYJAxGuU1U6kBQhhjQLydoIFxLveWn3XUy98gODPY6vx3TabOtWZqeHxGxQ3pIogjgEOjN0TGDTcL0h32gVzPkwMbmHXNpi+EHoRceM3jXIle+y3OFM4+zBk8Cu9s4bKs2CqegHQphv+eqc4NgPgKMPJdgqshDkv4="

"url":
"https://palbyp.pmservice.ibmcloud.com/pm/v1"

12. At this point, the application and service containers are ready.

5 Customizing files

This section deals with customizing the files that were downloaded to the SPSS Working directory *C:\SPSS\Sample1.*

5.1 \app\manifest.yml

Manifest.yml contains information about the deployment and is typically used to reduce the number of deployment details that you must specify every time you deploy an application to Bluemix.

1. Open the file *C:\SPSS\Sample1\app\manifest.yml* in a text editor.
2. Change the value of the **host** variable to reflect what was used in the *Creating an application* section previously. In this example, *SPSSDrugScoresAppDZ* was used.
3. Change the value of the **name** variable to reflect what was used in the *Creating an application* section previously. In this example, *SPSSDrugScoresAppDZ* was used.
4. Change the value of the **domain** variable to reflect what was used in the *Creating an application* section previously. In this example, *mybluemix.net* was used.
5. Change the value of the **memory** variable to 256M.
6. Add the following line:
   disk_quota: 1024M
7. Save and close the file.

5.2 \app\app.js

1. Open the file *C:\SPSS\Sample1\app\app.js* in a text editor.
2. Navigate to line 21 and update the *defaultBaseURL* that was recorded in the *Creating the service* section previously. Note that the <> must be removed and the entire string value must be enclosed in single quotes (‘’).

   var defaultBaseURL = 'https://palbyp.pmservice.ibmcloud.com/pm/v1';

3. Similar to the previous step, update the *defaultAccessKey* value key on line 22.
4. Save and close the file.

5.3 \app\public\js\app.js

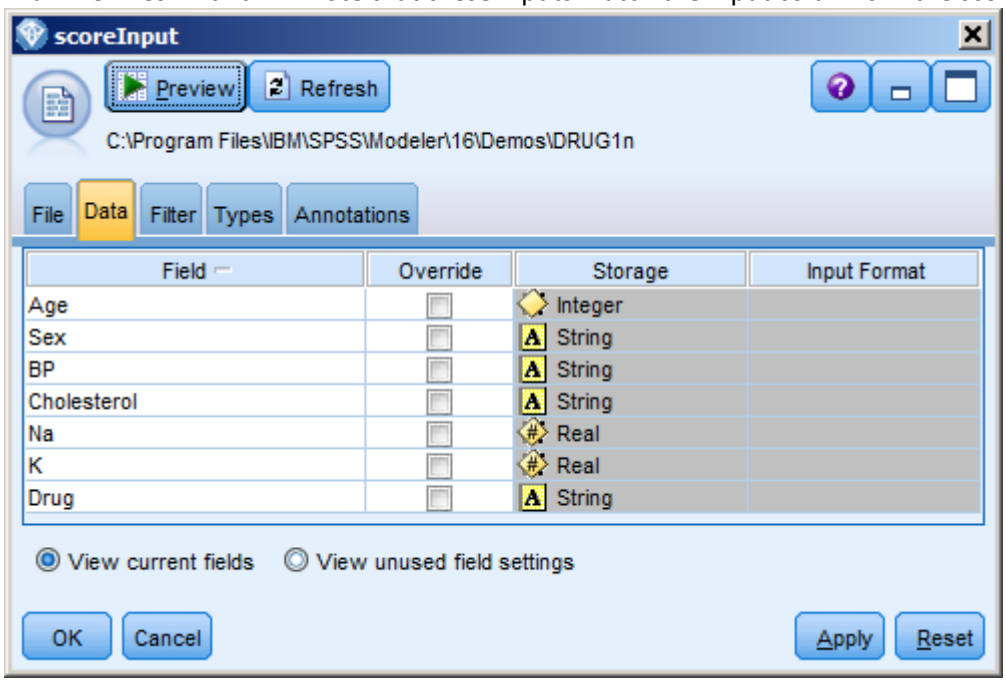
1. Open the file *C:\SPSS\Sample1\app\public\js\app.js* in a text editor.
2. Navigate to line 8.
3. Note that the value of the *$scope.context* variable is *drug1N*. While no change is required in this file, these steps are here to illustrate that every SPSS model scored in Bluemix must have a unique context ID. This will be discussed further in the *Uploading SPSS Models* section.
4. Close the file. No change is required.

5.4 \app\public\js\srv.js

1. Open the file *C:\SPSS\Sample1\app\public\js\srv.js* in a text editor:

2. Navigate to line 10. Note that the *tablename* variable matches the name of the source node in the SPSS Modeler model.

3. Examine lines 11 and 12. Note that these inputs match the input columns in the scoreInput node.



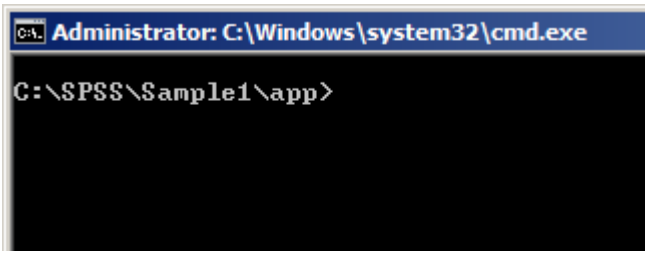
4. If you were to use a stream that used a different source node and had other parameters, they would need to be updated here.
5. Close the file. No change is required.

6 Porting source code

6.1 Node Package Manager

You can use the Node Package Manager utility to download supplemental files from Bluemix. The **Start Coding** section of the Bluemix application provides some of the commands that will help you with connecting to your Bluemix environment. It may be easier to use the commands there than to customize the steps below.

1. Launch a command prompt.
2. Change to the *C:\SPSS\Sample1\app* directory.

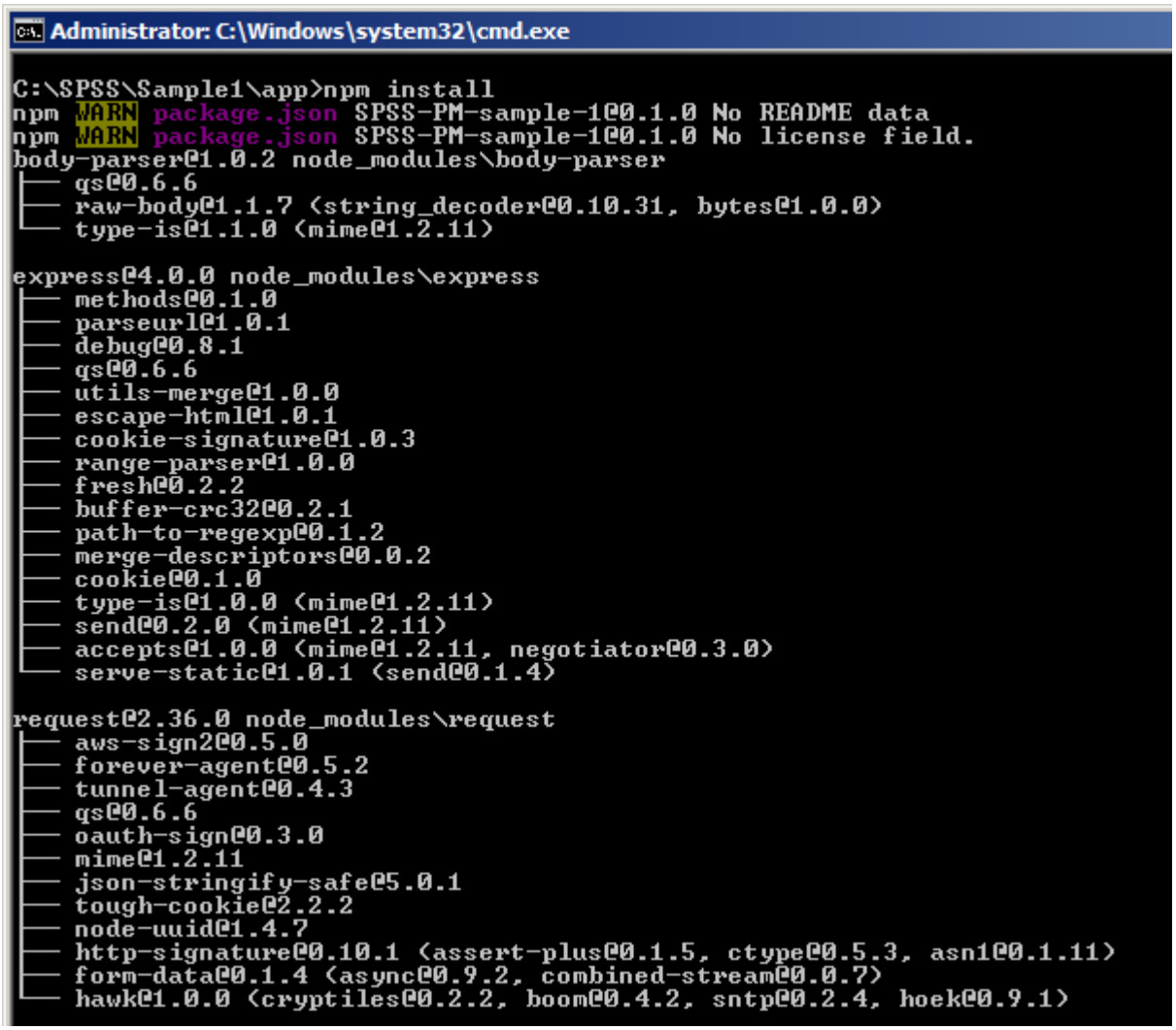


3. Connect to IBM Bluemix by typing the following command:
   bluemix api https://api.ng.bluemix.net

4. Log in to Bluemix by typing the following command:
   bluemix login -u david.zyla@ca.ibm.com -o david.zyla@ca.ibm.com -s SPSS_RT_Scoring
5. Type your password when prompted.

6. Download the supplemental files by typing:
   npm install

   This will create the directory *C:\SPSS\Sample1\app\node_modules*.



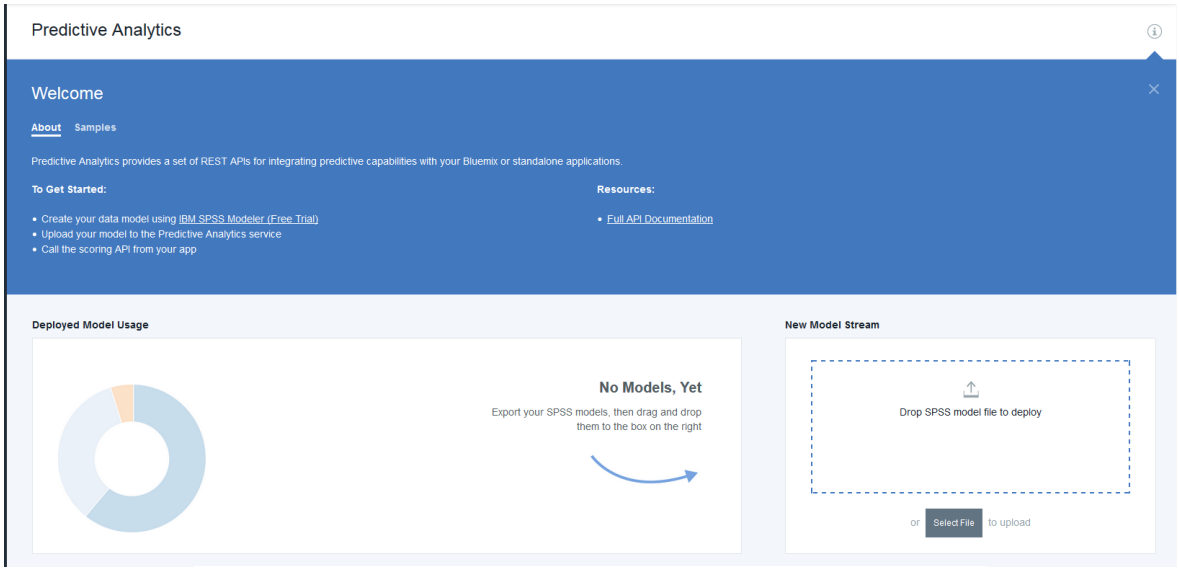
7. To push all files, including the customizations that were made, run the following command:
   cf push <Application Name>
   cf push SPSSDrugScoresAppDZ

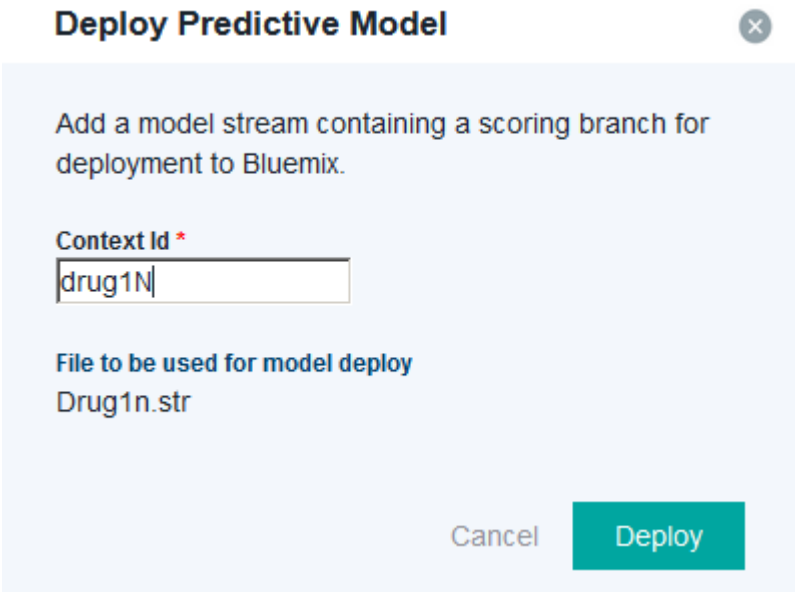
   The application is uploaded to Bluemix.

7 Uploading SPSS Models

At this point, you have moved your custom application to IBM Bluemix. But you still need to provide Bluemix with the SPSS models that you want to use.

1. Open a browser and log in to Bluemix.
2. From the dashboard, click the Machine Learning service that was created previously.
3. In the bottom right corner, note there is a **New Model Stream** section. The SPSS model we are going to provide can be found in *C:\SPSS\Sample1\model\Drug1n.str*.
   Upload the model by either dragging and dropping it or by using the **Select File** option.
4. You will be prompted to provide a Context Id. While this ID can be anything, this example uses **drug1N**.





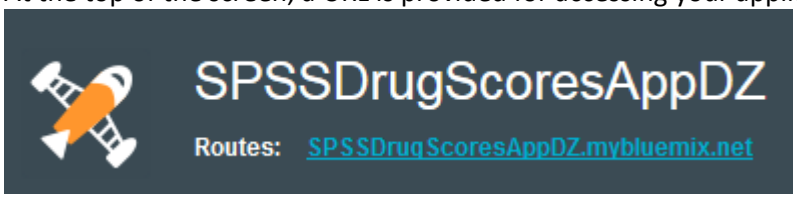
5. Type **drug1N** for the Context ID. If you choose a different Context Id, you must make the respective change in *C:\SPSS\Sample1\app\public\js\app.js* which was discussed earlier.
6. Click **Deploy**. The model is listed in the bottom section of the screen.

Manage Models - Status: Active

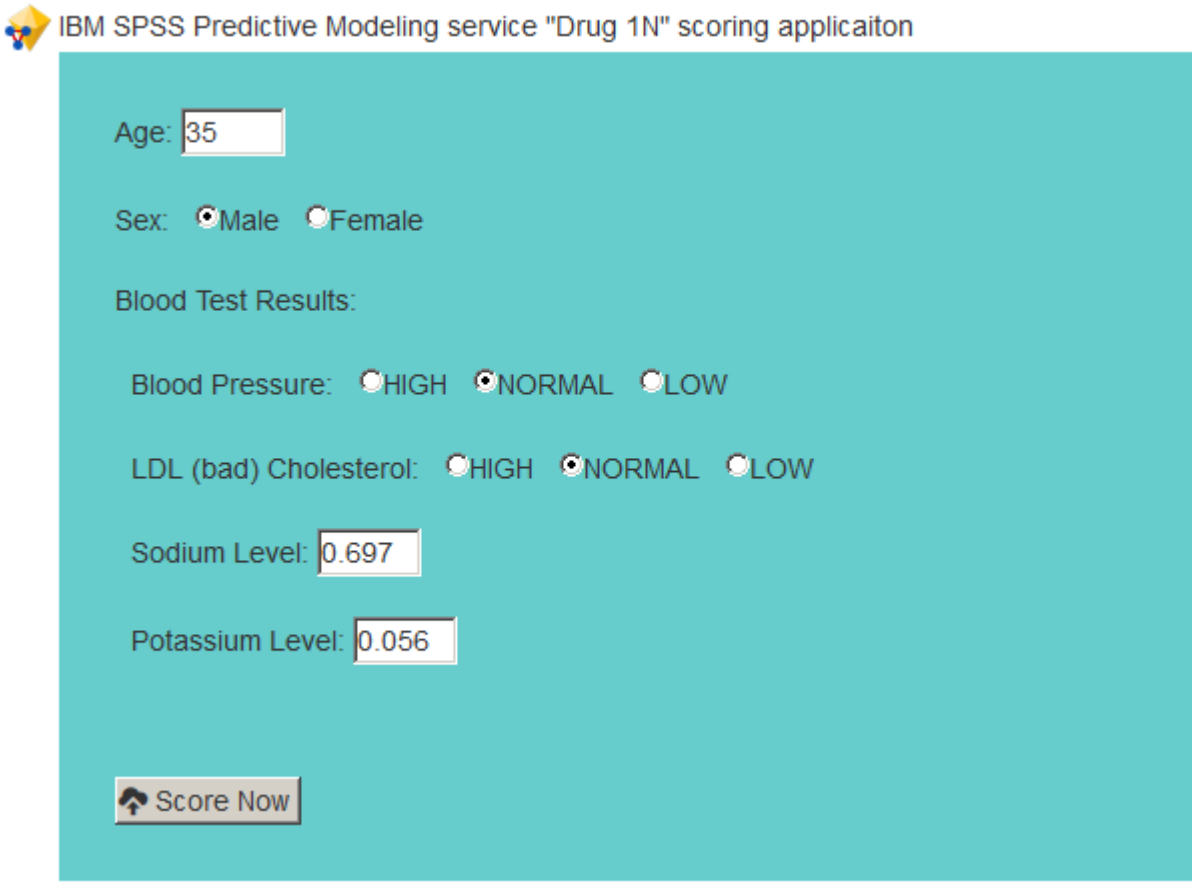
Context Id	File	Date Created	Date Updated	Action
drug1N	Drug1n.str	7/8/16	7/8/16	

8 Real-time scoring the sample

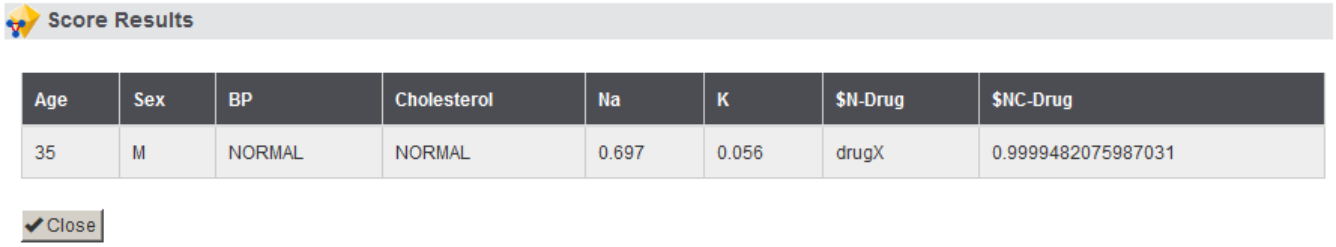
1. In Bluemix, navigate to your dashboard.
2. Click the application you created.
3. At the top of the screen, a URL is provided for accessing your application.



4. Click the URL. You are presented with your real-time scoring application.



5. You can change any of the values to simulate data being entered by a user of this application.
6. Click the **Score Now** button.



7. In this example, the input along with the predicted drug and predicted confidence are being displayed on the screen.

This document has walked through the process of:

- Setting up the necessary development tools
- Customizing the sample code
- Uploading an SPSS model
- Scoring the SPSS model with data provided in real-time