

Overview

This lab will introduce the Watson Machine Learning capability using the Titanic dataset. The lab will consist of the following steps:

1. Setting up the environment
2. Adding a data asset to the DSX Labs project
3. Creating a Model to predict whether a person would survive
4. Deploying and Test the Model
5. Creating a simple web front-end

Step 1: Setting up your environment

To use IBM Watson Machine Learning you must have the following service instances in your Bluemix dashboard:

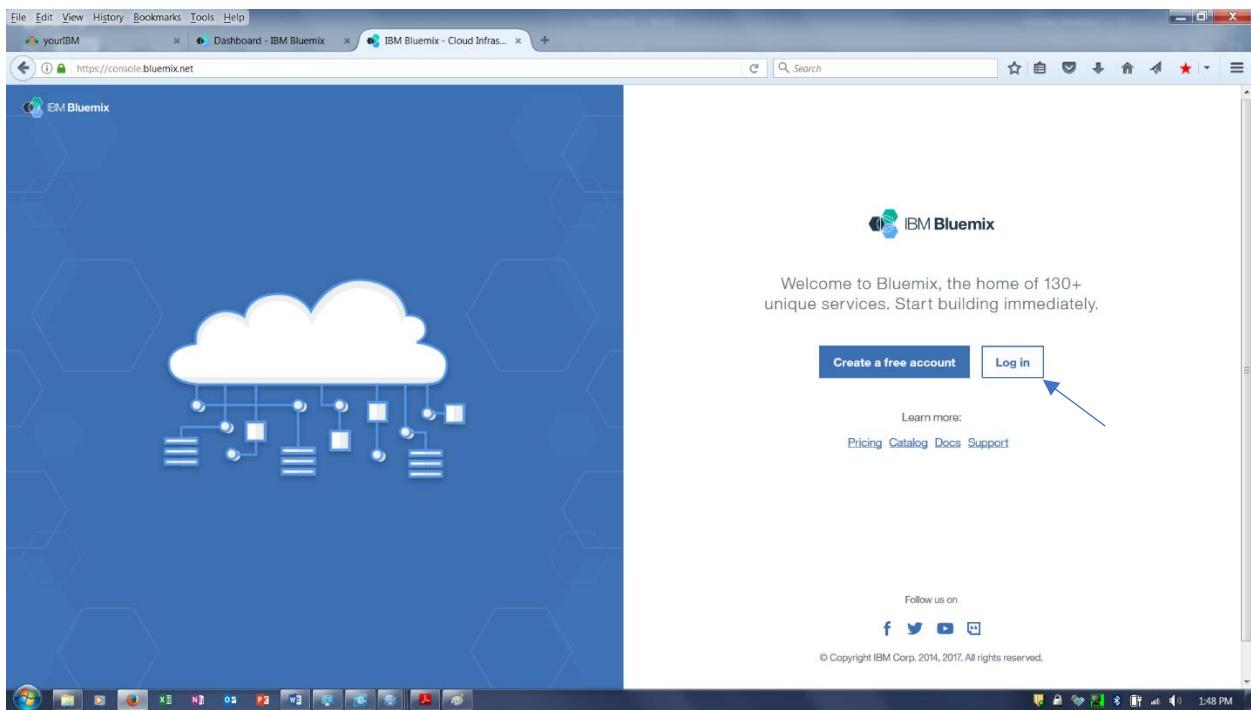
- Watson Machine Learning
- Object Storage
- Apache Spark

The Object Storage and Apache Spark service instances should already exist having been created when your DSX account was provisioned. We now need to provision a Machine Learning Service.

Step 1.1: Creating a Machine Learning Instance

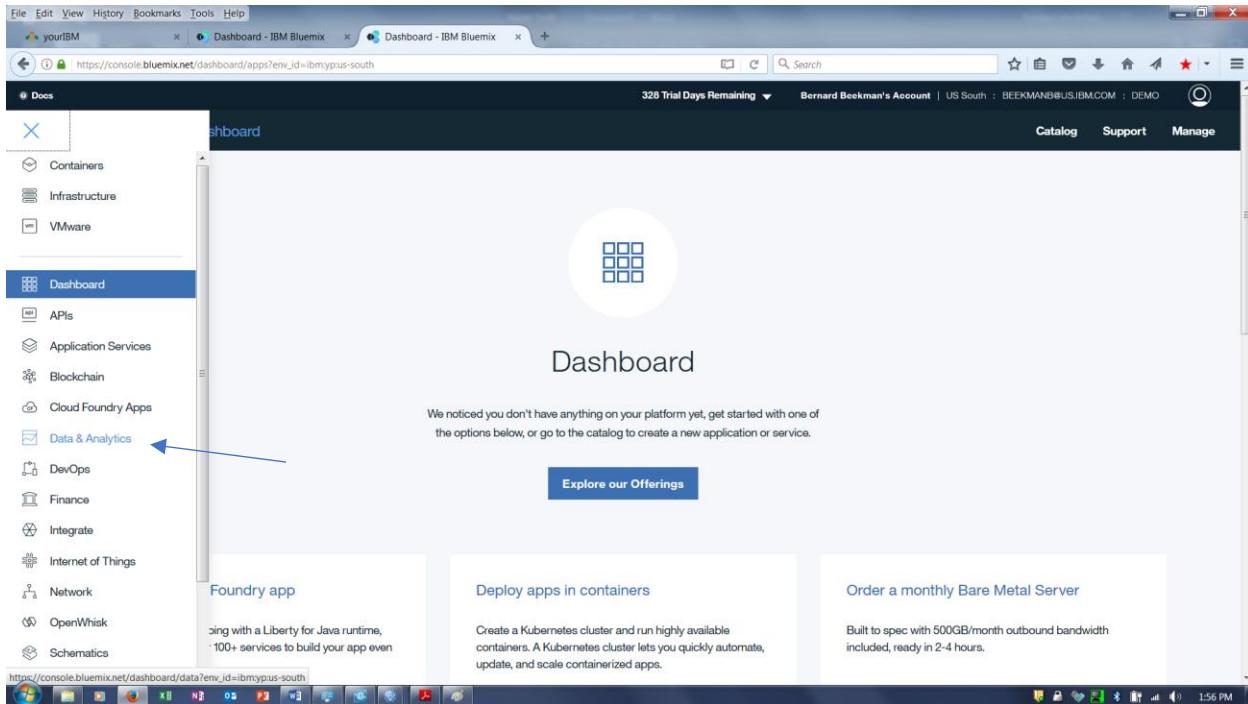
To create a Machine Learning service instance, you must perform the following steps:

1. Log into Bluemix at www.bluemix.net.



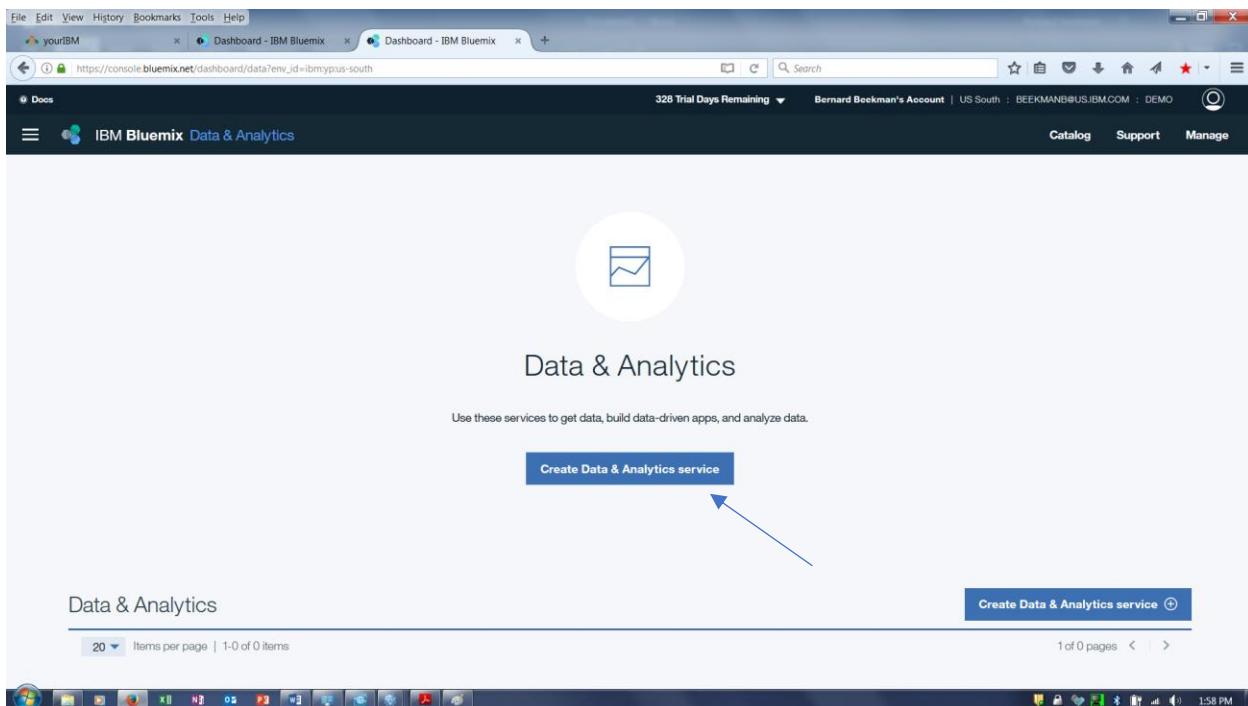
- Once logged in, click on the hamburger icon, and from the navigation panel, click **Data & Analytics**.

A screenshot of the IBM Bluemix Dashboard. The top navigation bar includes the IBM Bluemix logo, a "Docs" link, and "Catalog", "Support", and "Manage" buttons. A blue arrow points from the text "Once logged in, click on the hamburger icon, and from the navigation panel, click Data & Analytics." to the "Docs" link. The main dashboard area features a large circular icon with a grid of squares. Below it, the word "Dashboard" is centered. A message states: "We noticed you don't have anything on your platform yet, get started with one of the options below, or go to the catalog to create a new application or service." A blue button labeled "Explore our Offerings" is visible. At the bottom, there are three cards: "Create a Cloud Foundry app", "Deploy apps in containers", and "Order a monthly Bare Metal Server". The "Create a Cloud Foundry app" card includes the text: "Go straight to developing with a Liberty for Java runtime, then add some of our 100+ services to build your app even faster." The "Deploy apps in containers" card includes the text: "Create a Kubernetes cluster and run highly available containers. A Kubernetes cluster lets you quickly automate, update, and scale containerized apps." The "Order a monthly Bare Metal Server" card includes the text: "Built to spec with 500GB/month outbound bandwidth included, ready in 2-4 hours." The bottom of the screen shows the Windows taskbar with various pinned icons.

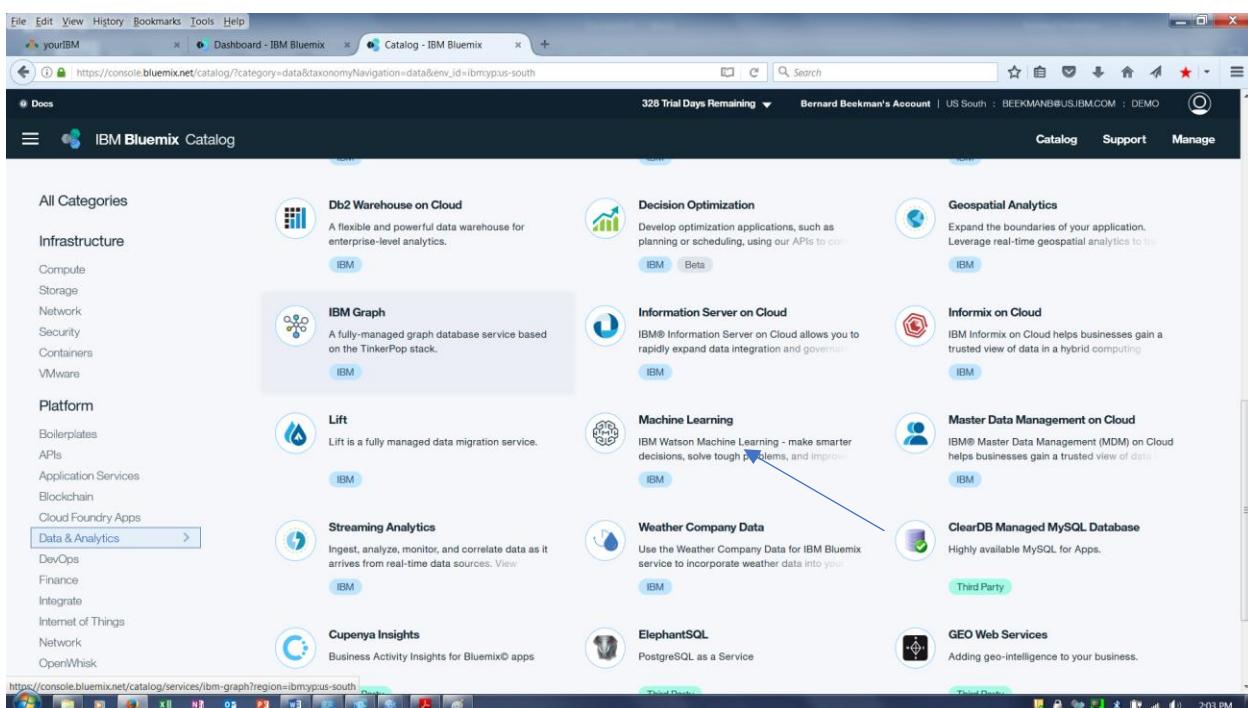


You see a screen centered on data services. You can return here regularly to work with your data and analytics services from one easy-to-use page. Check to see if a Machine Learning service already exists. If not, continue, otherwise go to Step 1.2: Adding existing Bluemix instances to a project in Data Science Experience

3. Click the **Create Data & Analytics Service** button.



4. Scroll down to Machine Learning and click.



5. Configure service.

Enter a descriptive name for your service, choose a space, and select your data plan (find plan comparison and pricing details on this page). Click on **Create**.

The screenshot shows the IBM Bluemix Catalog interface. A blue arrow points from the 'Service name' field ('Machine Learning') to the 'Create' button at the bottom right. Another blue arrow points from the 'Select region to deploy in' dropdown ('US South') to the same 'Create' button. A third blue arrow points from the 'Choose a space' dropdown ('DEMO') to the 'Create' button. A fourth blue arrow points from the 'Connect to' dropdown ('Leave unbound') to the 'Create' button.

File Edit View History Bookmarks Tools Help
yourIBM Dashboard - IBM Bluemix Machine Learning - IBM Bl...
https://console.bluemix.net/catalog/services/machine-learning?env_id=ibmypus-south&taxonomyNavigation=data
328 Trial Days Remaining Bernard Beekman's Account | US South : BEEKMANB@US.IBM.COM : DEMO
Docs IBM Bluemix Catalog Catalog Support Manage
View all Machine Learning

IBM Watson Machine Learning is a full-service Bluemix offering that makes it easy for developers and data scientists to work together to integrate predictive capabilities with their applications. The Machine Learning service is a set of REST APIs that you can call from any programming language to develop applications that make smarter decisions, solve tough problems, and improve user outcomes.

Service name: Machine Learning
Credential name: Credentials-1
Select region to deploy in: US South Choose an organization: BEEKMANB@US.IBM.COM Choose a space: DEMO
Connect to: Leave unbound

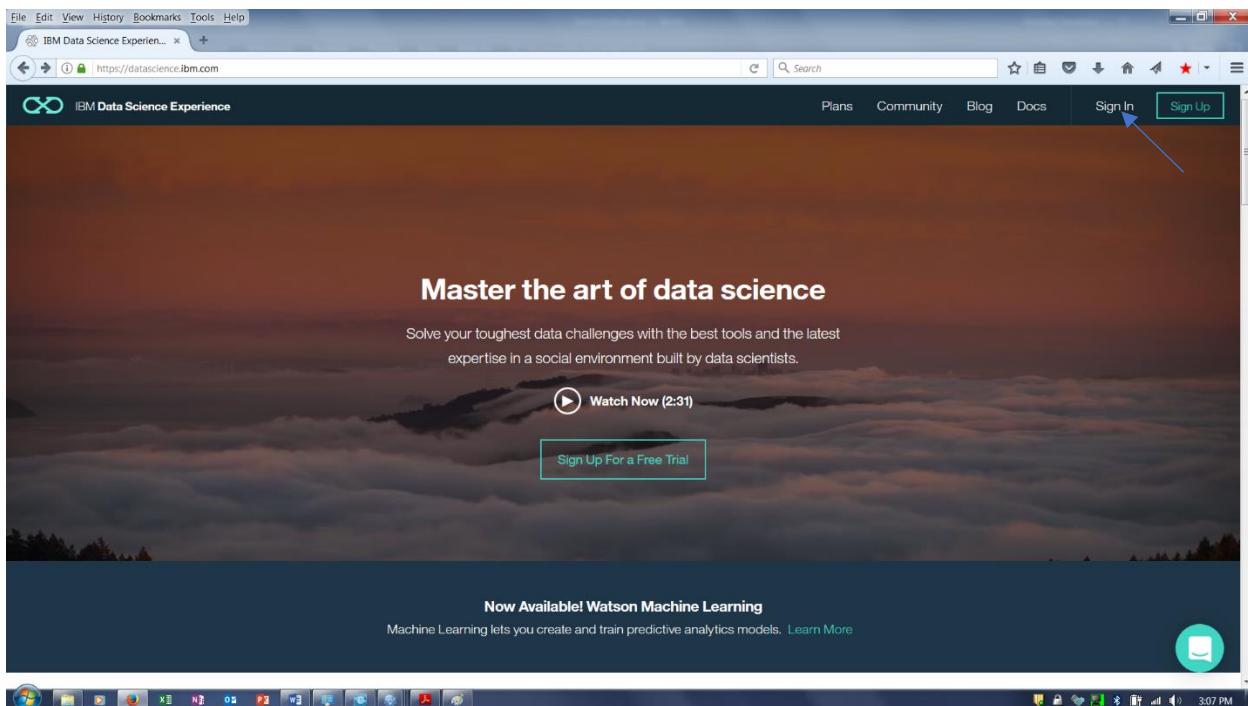
AUTHOR IBM PUBLISHED 08/01/2017 TYPE Service LOCATION US South, United Kingdom

Features
Need Help? Contact Bluemix Sales Estimate Monthly Cost Cost Calculator Create

Step 1.2: Adding existing Bluemix instances to a project in Data Science Experience

If you already have instances, but have not linked them to a project in Data Science Experience, you must perform the following steps:

1. Log on to IBM Data Science Experience – <https://datascience.ibm.com>



2. Click Projects > View All Projects.

Name	Role	Collaborators	Date Created	Last Updated
GBS Bootcamp	Admin	BB	Aug 19, 2017	Aug 20, 2017
DSX Lab	Admin	BB	Mar 28, 2017	Aug 19, 2017
Machine Learning	Admin	BB, SP, SP, SP, +3	Aug 08, 2017	Aug 18, 2017

- Click on the project that you created in the prerequisites, or if no project was created you can either create one, or click on the default project. (For the remainder of this document, I'm assuming the project name is DSX-Labs).

The screenshot shows the IBM Data Catalog interface. At the top, there's a navigation bar with tabs like 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Tools', and 'Help'. Below the navigation bar, the main header says 'IBM Data Catalog' and 'Overview - IBM Machine L...'. The main content area is titled 'My Projects' and contains a table of projects. The table has columns for 'NAME', 'ROLE', 'COLLABORATORS', 'CREATOR', 'LAST MODIFIED', and 'ACTIONS'. Projects listed include 'GBS Bootcamp', 'DSX Lab', 'Machine Learning', 'DSX Labs' (which is selected and highlighted with a blue border), 'Education Analysis', 'Ed Analytics Framework', 'Hackathon', 'GWU-Test', and 'GWU'. A blue arrow points from the text 'Select the Settings Tab.' to the 'Settings' tab in the navigation bar.

4. Select the Settings Tab.

The screenshot shows the IBM Data Catalog interface for the 'DSX Labs' project. The top navigation bar is identical to the previous screenshot. The main content area shows the 'DSX Labs' project details. It includes sections for 'Overview', 'Analytics Assets', 'Data Assets', 'Bookmarks', 'Deployments', 'Collaborators', and 'Settings'. The 'Settings' tab is highlighted with a blue border. Below the tabs, there are three main sections: 'Notebooks' (with two items: 'Machine Learning with Spark - Joel' and 'Connect and Interact with DashDB'), 'Data Assets' (with two items: 'Female-human-trafficking.csv' and 'myDashDB'), and 'Bookmarks' (with one item: 'A comparison of logistic regression a...'). A blue arrow points from the text 'Scroll down to Associated Services...' to the 'Associated Services' section in the bottom right corner.

5. Scroll down to Associated Services. To add a service, in the **Associated Services** panel, click **add associated service**, select the Machine Learning service.

The screenshot shows the IBM Data Catalog interface. In the top navigation bar, there are tabs for 'IBM Data Catalog' (selected), 'Overview - IBM Machine L...', 'Tutorial: Build a logistic re...', and 'Mail'. Below the tabs, the URL is https://apisportal.ibm.com/projects/8e5d9ac9-05cc-4b93-a857-ad4d0191aa75/settings?context=data. The main content area has a dark header with 'IBM Data Catalog' and 'Projects Catalog Governance Data Services'. A sub-header 'My Projects > DSX Labs' is visible. The 'Storage' section shows 'Type Object Storage (Swift API)' and 'Container DSXLabs'. It displays '338.03 KB Used' and '0.0% of 5GB Used'. The 'Associated Services' section lists 'Fisheries' with 'Spark' under 'Service Type' and 'Personal' under 'PLAN'. A dropdown menu on the right shows options like 'Amazon EMR Spark', 'IBM Analytics Engine', 'Machine Learning', and 'Spark'. A blue arrow points from the text 'Select the Machine Learning service instance from the drop down list and then click Select.' to the 'Machine Learning' option in the dropdown. The 'Access Tokens' section is empty, showing a message 'you currently have no access tokens'. The bottom of the screen shows a Windows taskbar with various icons.

6. Select the Machine Learning service instance from the drop down list and then click **Select**.

The screenshot shows the IBM Data Science Experience interface. In the top navigation bar, there are tabs for 'Overview - IBM Machine L...', 'Tutorial: Build a logistic re...', 'Dashboard - IBM Bluemix', and 'IBM Data Science Experience' (selected). Below the tabs, the URL is https://apisportal.ibm.com/data/discovery/predictive-modeling/picker?context=analytics&projectId=8e5d9ac9-05cc-4b93-a857-ad4d0191aa75&redi. The main content area has a dark header with 'IBM Data Science Experience' and 'Projects Catalog Tools Data Services Community'. A sub-header 'Add Machine Learning Service' is visible. There are two tabs: 'Existing' (selected) and 'New'. Under 'Existing Service Instance', 'Watson Machine Learning' is listed. A blue arrow points from the text 'Select the Machine Learning service instance from the drop down list and then click Select.' to the 'Select' button. The bottom of the screen shows a Windows taskbar with various icons.

7. Click on the Project Overview tab.

Project Name
DSX Labs

Description
Labs for July 10 Class

Cancel Save

Step 2: Adding a Data Asset to the DSX Labs project

1. Download the Titanic data file from

https://github.com/jpatter/ML-POT/blob/master/Lab-2/data/titanic_cleaned.csv

The data in this file has already been prepared and it ready to be input into the Modeling step.

2. Right click on Raw, and click on Save link as

jpatter / ML-POT

Branch: master

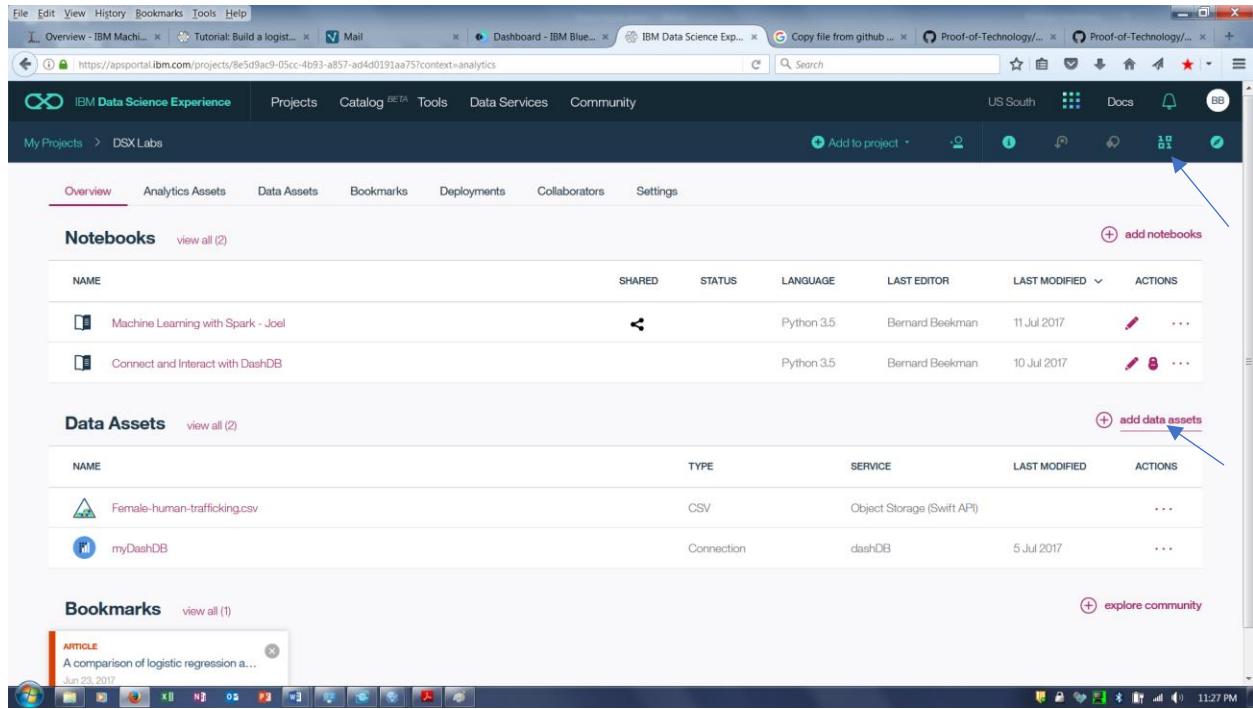
ML-POT / Lab-2 / data / titanic_cleaned.csv

1845 lines (1844 sloc) | 75.6 KB

Raw Blame History

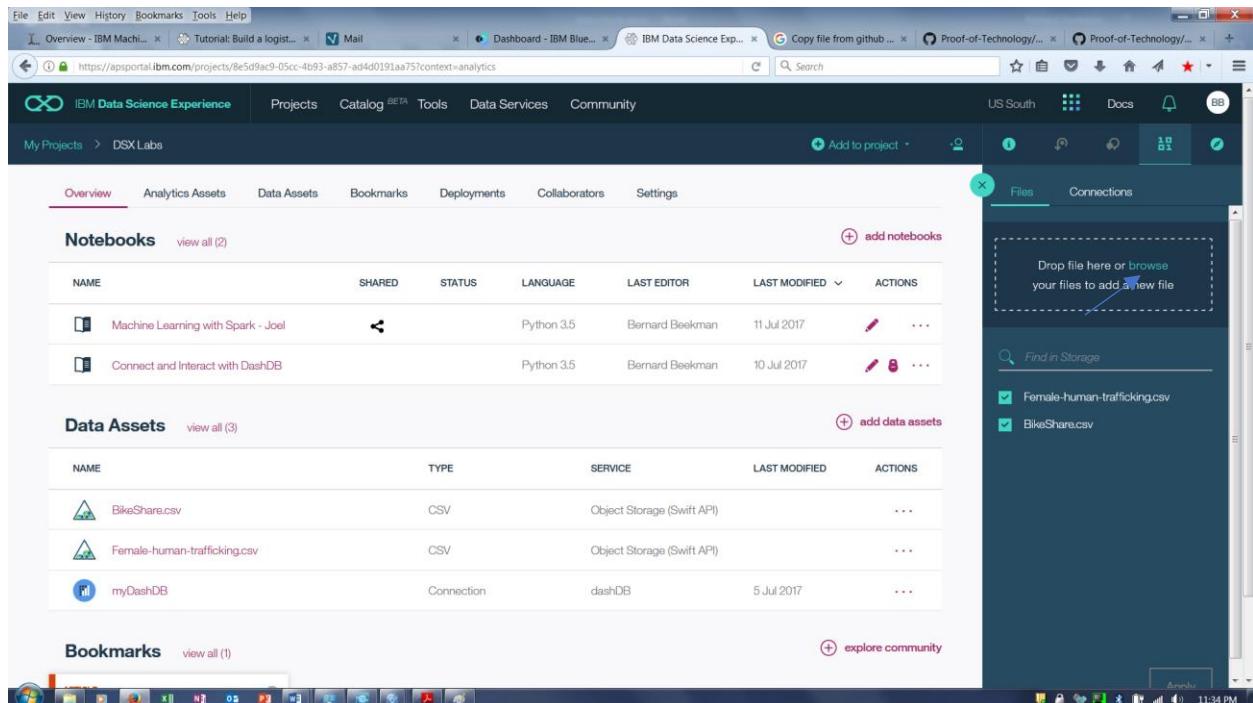
	pclass	survived	name	sex	sibsp	parch	ticket	fare	embarked
1	1	1	Allan, Miss. Elisabeth Walton	female	0	0	24160	211.337500	S
2	1	1	Allan, Master. Hudson Trevor	male	1	2	113781	151.550000	S
3	1	0	Allan, Miss. Helen Loraine	female	1	2	113781	151.550000	S
4	1	0	Allan, Mr. Hudson Joshua Creighton	male	1	2	113781	151.550000	S
5	1	0	Allan, Mr. Hudson Joshua Creighton	female	1	2	113781	151.550000	S

3. Go back to the DSX-Labs project. Click on **add data assets** or the  icon.



The screenshot shows the IBM Data Science Experience interface. In the top navigation bar, there is a 'Projects' tab and a 'Data Assets' tab. Below the navigation bar, there is a 'Notebooks' section, a 'Data Assets' section, and a 'Bookmarks' section. The 'Data Assets' section contains two items: 'Female-human-trafficking.csv' (CSV, Object Storage (Swift API)) and 'myDashDB' (Connection, dashDB). On the right side of the 'Data Assets' table, there is a blue button labeled '+ add data assets'. A blue arrow points to this button.

4. Click on browse and then go to the folder where the titanic_cleaned.csv is stored. Select titanic_cleaned.csv and then click Open.



The screenshot shows the same IBM Data Science Experience interface as the previous one, but with a file browser sidebar on the right. The sidebar has two tabs: 'Files' (selected) and 'Connections'. Under the 'Files' tab, there is a 'Drop file here or browse' area with a dashed box. Below it is a 'Find in Storage' section showing files 'Female-human-trafficking.csv' and 'BikeShare.csv'. A blue arrow points to the 'Drop file here or browse' area.

Step 3: Create a Model to predict survival

1. Click on the Analytic Assets Tab

The screenshot shows the IBM Data Science Experience web interface. The top navigation bar includes 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Tools', and 'Help'. Below the bar, there are several tabs: 'Overview - IBM Machi...', 'Tutorial: Build a logist...', 'Dashboard - IBM Blue...', 'IBM Data Science Exp...', 'Copy file from github ...', 'Proof-of-Technology/...', and 'Proof-of-Technology/...'. The main content area has a dark header with 'IBM Data Science Experience' and a navigation menu: 'Projects', 'Catalog BETA', 'Tools', 'Data Services', and 'Community'. On the right side, there's a sidebar titled 'US South' with sections for 'Docs', 'Community', and 'Analytics'. The main content area is titled 'My Projects > DSX Labs'. It features a tab bar with 'Overview' (selected), 'Analytics Assets' (with a red arrow pointing to it), 'Data Assets', 'Bookmarks', 'Deployments', 'Collaborators', and 'Settings'. Under 'Analytics Assets', there are three sections: 'Notebooks' (with 2 items), 'Data Assets' (with 3 items), and 'Bookmarks' (with 1 item). A sidebar on the right contains 'Files' and 'Connections' sections, with a 'Drop file here or browse your files to add a new file' area and a 'Find in Storage' section listing 'Female-human-trafficking.csv' and 'BikeShare.csv'. The bottom of the screen shows a Windows taskbar with various icons and the time '11:34 PM'.

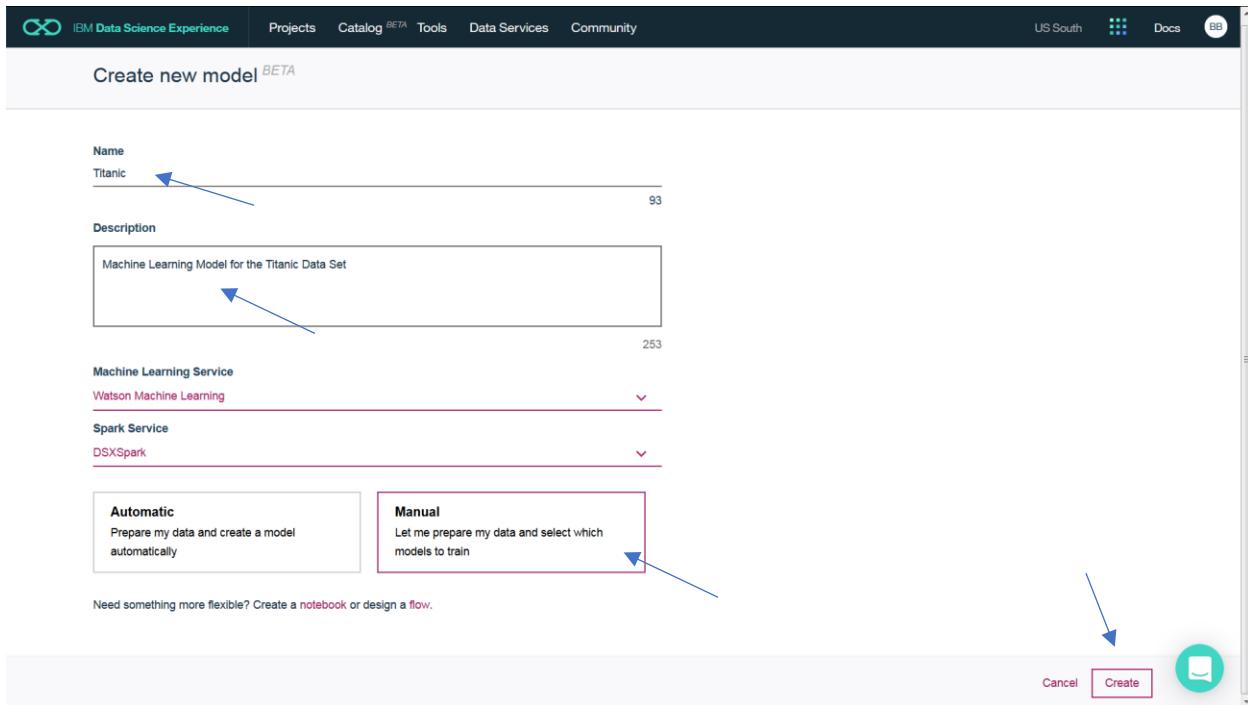
2. Click on the add models.

This screenshot shows the same IBM Data Science Experience interface as the previous one, but with a different focus. The 'Analytics Assets' tab is still selected. In the 'Models' section, there is a table with two rows:

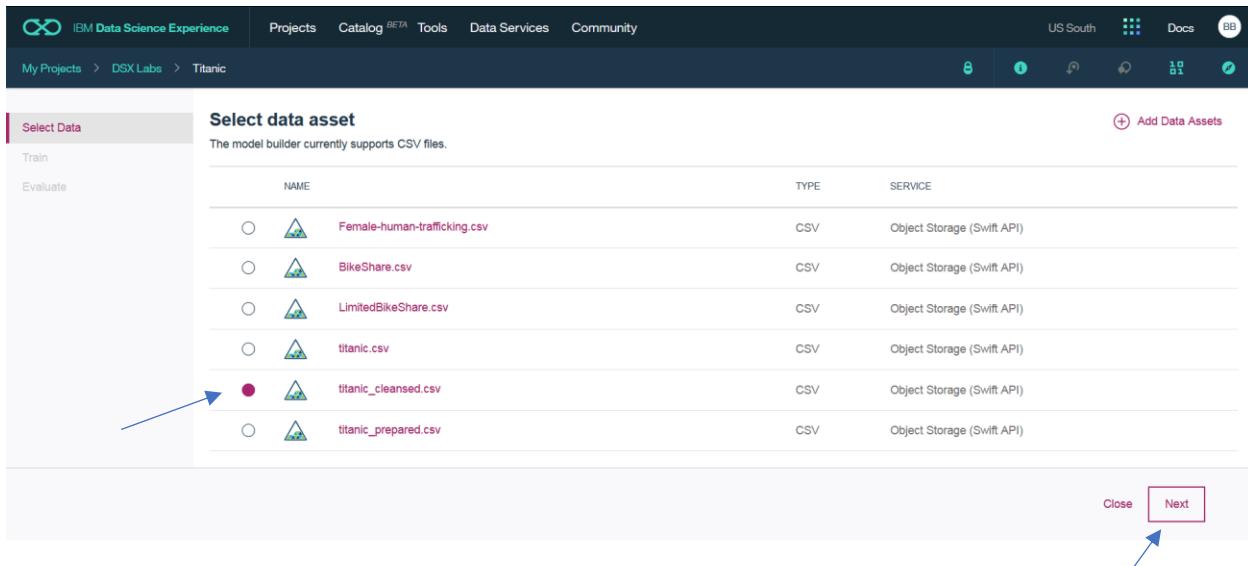
NAME	STATUS	RUNTIME	LAST MODIFIED	ACTIONS
Female Human Trafficking- Manual	untrained		21 Aug 2017	
Female Human Trafficking	untrained		9 Jul 2017	

To the right of the table, there is a blue arrow pointing to a red '+' icon labeled '+ add models'.

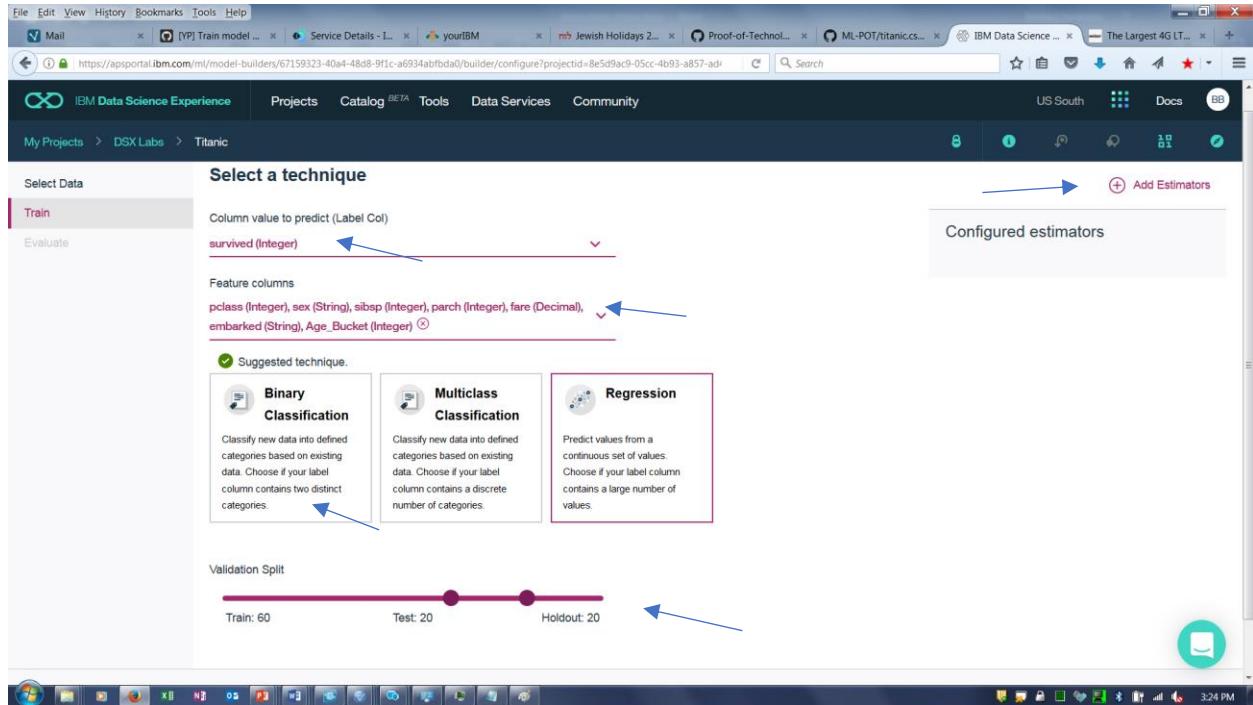
3. Enter the Model Name, Description, Select Manual, and click on Create.



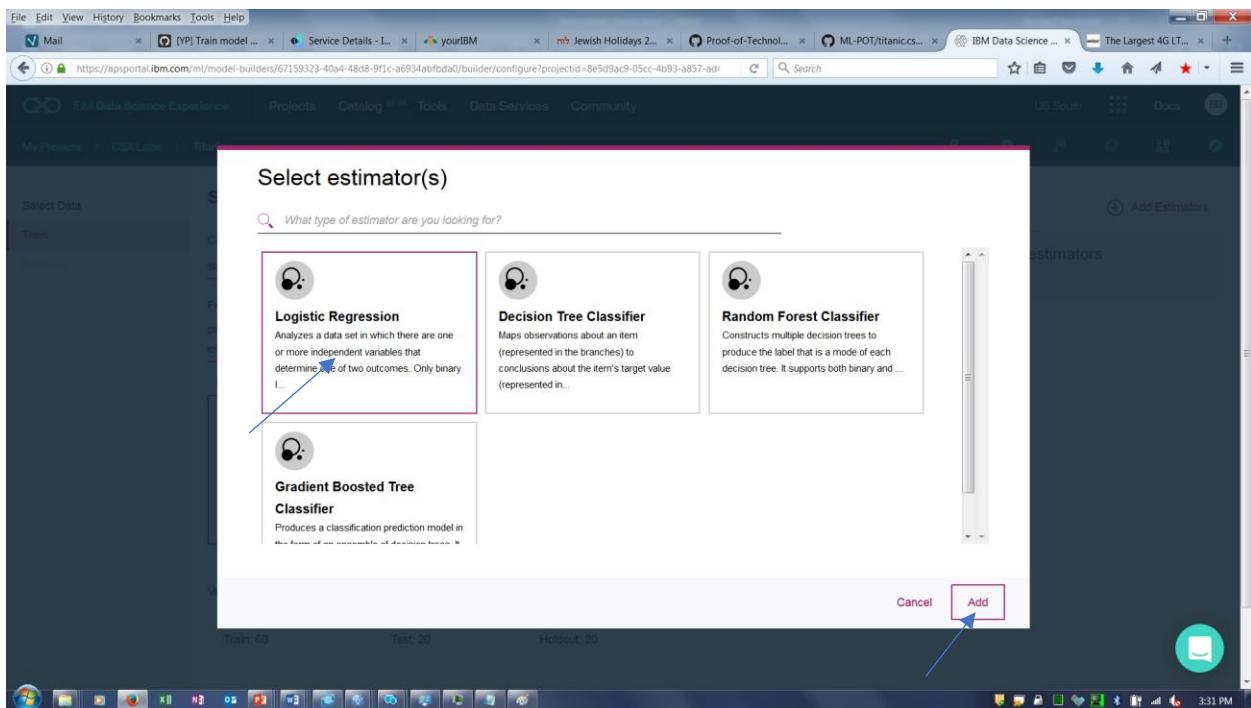
4. Click on the titanic_cleansed.csv and click on Next



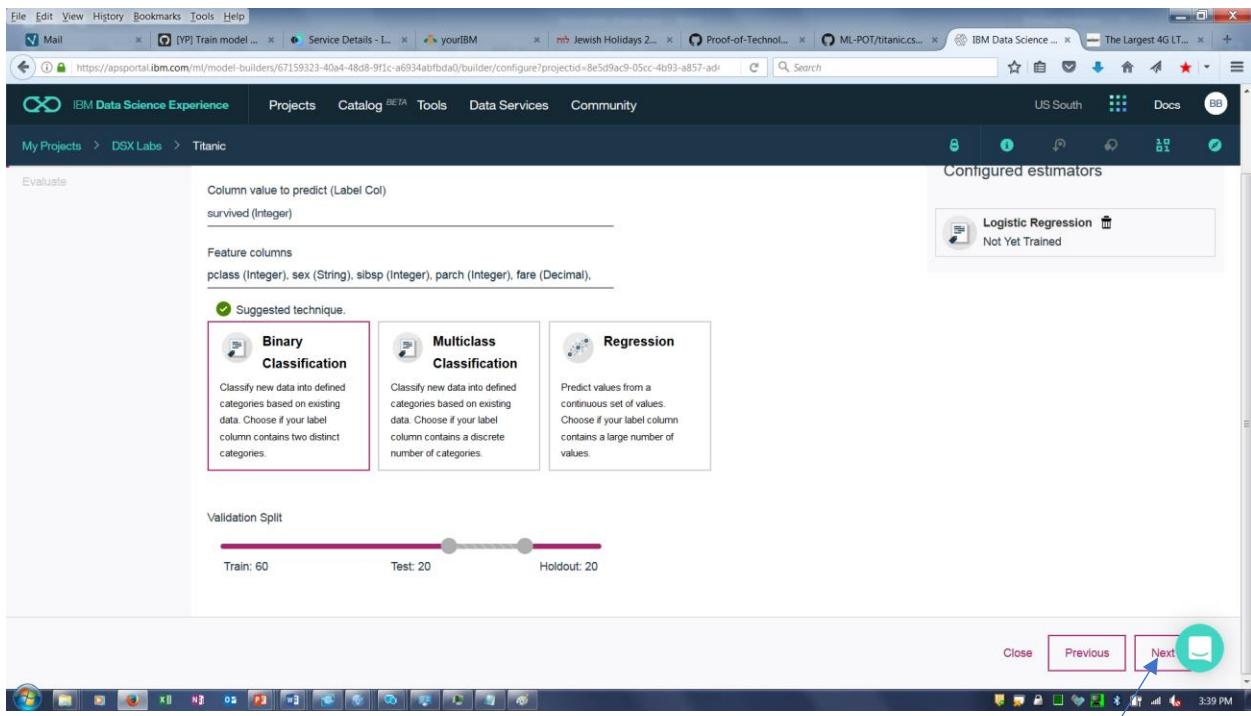
5. For **Column value to predict (Label Col)** select **survivor**. For **Feature columns** select the following features (**pclass,sex,sibsp,parch,fare,embarked,Age_Bucket**) . Click on the **Binary Classification Box** (which is suggested by the service). Adjust the **Validation Split** as desired. Click on **Add Estimators** to add the specific models to use.



6. Select the **Logistic Regression**. Select **Add**.



7. Select the Next button.



8. The system trains and evaluates each model. If more than one model was selected, the models would be listed in descending order of quality with the best result at the top. Click on the **Logistic Regression** and then click **Save**.

The screenshot shows the 'Select model' page in IBM Data Science Experience. At the top, there are navigation links: Projects, Catalog BETA, Tools, Data Services, and Community. On the right, there are icons for US South, Docs, and BB. Below the header, the path is My Projects > DSX Labs > Titanic. The main area has tabs for Select Data, Train, and Evaluate, with Evaluate selected. A table lists a single model entry:

	ESTIMATOR TYPE	STATUS	PERFORMANCE	AREA UNDER ROC CURVE	AREA UNDER PR CURVE	LAST EVALUATION	ACTIONS
●	LogisticRegression	Trained & Evaluated	Good	0.81287	0.81673	26 Aug 2017, 6:29 PM	...

At the bottom right are buttons for Close, Previous, and Save, with Save highlighted by a blue arrow.

9. The system displays the model training summary. To run a sample prediction, select the **Predictions** tab

The screenshot shows the 'Predictions' tab for the Titanic model. The top navigation bar and path are identical to the previous screenshot. The 'Predictions' tab is selected, indicated by a red underline. Below the tabs, there is a section for the 'Titanic' model with a delete icon. A table provides details about the model configuration:

Machine learning service	Machine Learning-s2
Label column	survived
Model builder details	View
Training data schema	View
Input data schema	View
Runtime environment	spark-2.0
Training date	26 Aug 2017, 6:30 PM

Below the table, there is a 'Deployments' section.

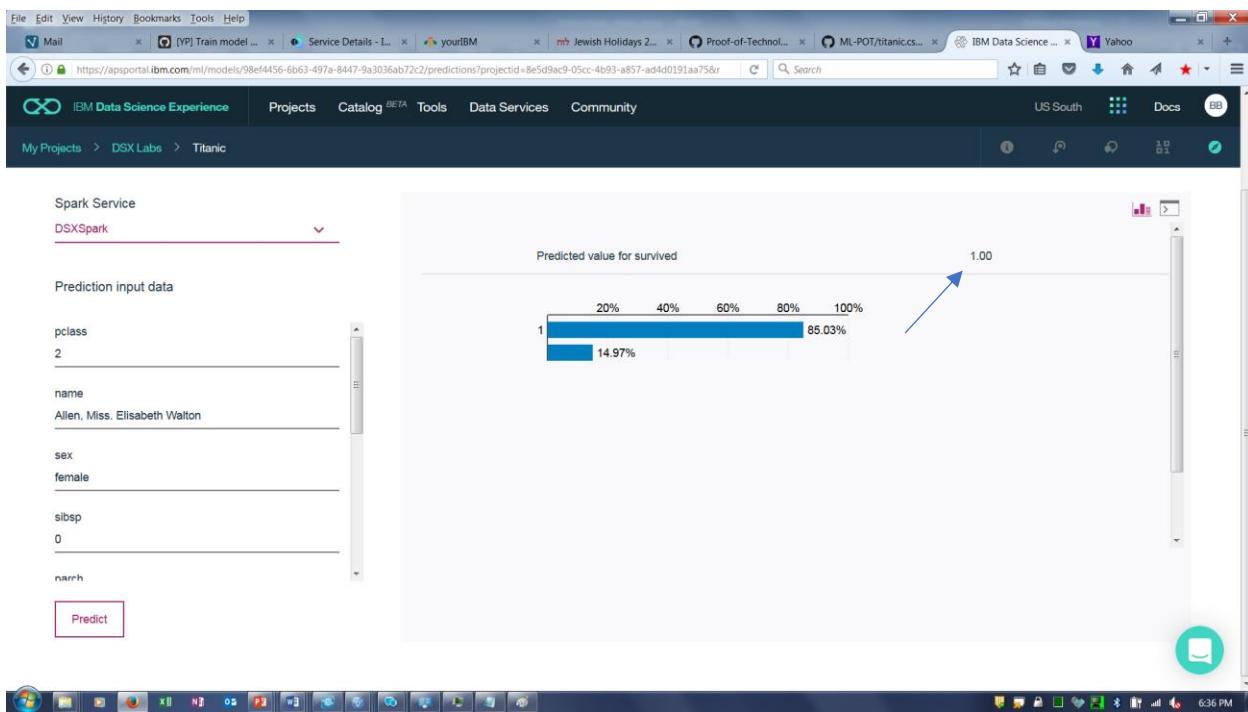
10. Enter values for the input features and then click on **Predict**.

The screenshot shows the IBM Data Science Experience interface. At the top, the navigation bar includes 'My Projects > DSX Labs > Titanic'. Below the navigation is a header for 'Spark Service' and 'DSXSpark'. The main area is titled 'Prediction input data' and contains the following fields:

- pclass: 2
- name: Allen, Miss. Elisabeth Walton
- sex: female
- sibsp: 0
- narrh

A blue arrow points from the 'Predict' button at the bottom left to the 'Predict' button in the screenshot below. Another blue arrow points from the 'Predict' button to the predicted result.

11. The prediction for survivor is displayed along with the confidence in the prediction.



Step 4: Deploying a Model

We can deploy the model to enable applications to invoke it via an API call.

1. Select the **Details Tab**
2. Scroll down to the **Add Deployments** option. Click on **Add Deployments**

The screenshot shows the IBM Data Science Experience interface. At the top, there are navigation links: Projects, Catalog BETA, Tools, Data Services, and Community. On the right, it shows 'US South' and various icons. Below the header, the path 'My Projects > DSX Labs > Titanic' is visible. The main content area has two sections: 'Model details' and 'Deployments'. The 'Model details' section contains a table with the following rows:

Label column	survived
Model builder details	View
Training data schema	View
Input data schema	View
Runtime environment	spark-2.0
Training date	26 Aug 2017, 6:30 PM

The 'Deployments' section shows a table with one row: 'Your model is not deployed.' At the top right of this section is a button labeled '+ Add Deployment' with a blue arrow pointing to it.

3. Select Online for **Deployment Type**, enter **Titanic_Deployment** for **Name**, and click on **Deploy**.

The screenshot shows the IBM Data Science Experience interface with a browser window open to the same 'Titanic' project page. A modal dialog box titled 'Deploy model' is displayed in the center. It contains the following fields:

- Deployment Type: A dropdown menu set to 'Online' with a blue arrow pointing to it.
- Name: A text input field containing 'Titanic_Deployment' with a blue arrow pointing to it.
- Buttons at the bottom: 'Close' and 'Deploy' (which is highlighted with a red box and a blue arrow pointing to it).

4. The system responds with an acknowledgement that the model was successfully deployed. Click on **Titanic_Deployment** to test the deployed API.

The screenshot shows the IBM Data Science Experience web interface. At the top, there are multiple tabs open in a browser. Below the tabs, the main interface has a header with 'IBM Data Science Experience' and navigation links for 'Projects', 'Catalog BETA', 'Tools', 'Data Services', and 'Community'. A search bar is also present.

In the main content area, there is a message box stating 'Your deployment was successfully created.' with a green checkmark icon. Below this, there is a table titled 'Deployments' with columns for 'NAME', 'DEPLOYMENT TYPE', and 'ACTIONS'. One row in the table is highlighted with a blue arrow pointing to it, showing the deployment named 'Titanic_Deployment' with a status of 'Online'.

5. The system displays information about the deployed service including the endpoint to invoke by an application (e.g. web application predicting survival). Click on **Test API** to test out the API.

The screenshot shows the 'Deployment Details' page for the 'Titanic_Deployment'. The page has a header with 'IBM Data Science Experience' and navigation links. The URL in the address bar is <https://apsportal.ibm.com/ml/deployments/00c07a3c-61ec-45ef-8cd8-213c18f1b448/details?projectid=8e5d9ac9-05cc-4b93-a857-ad4d0191aa75&mlins>.

The main content area is titled 'Deployment Details' and contains several sections with input fields:

- Details** (selected tab)
- Test API**

The 'Test API' section contains the following details:

Name	Titanic_Deployment
Type	online
Scoring End Point	https://ibm-watson-ml.mybluemix.net/v3/wml_instances/374817e5-8365-42da-a434-cb20e3d1fba4/published_models/98ef4456-6b63-497a-8447-9a3036ab72c2/deployments/00c07a3c-61ec-45ef-8cd8-213c18f1b448/online
Status	ACTIVE
Associated Model	Titanic

6. Enter values for the input fields and then click on **Predict**. Note that the values inputted for any of the fields not included in the model parameters (e.g. name) will not affect the prediction.

My Projects > DSX Labs > Titanic_Deployment

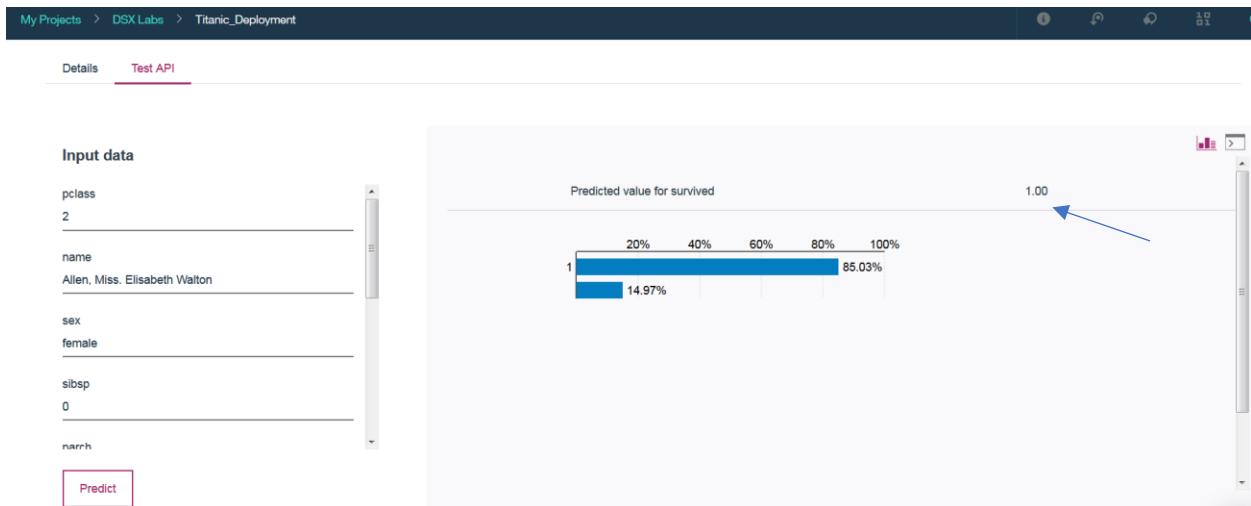
Details Test API

Input data

pclass	2
name	Allen, Miss. Elisabeth Walton
sex	female
sibsp	0
narch	

Predict

7. The predicted result is returned.



Step 5: Deploy a simple web front-end to invoke the Watson Machine Learning service

This section will provide an example of a simple Python Flask front-end that invokes the Titanic scoring API demonstrating embedding machine learning in a web app. You will click on a link below that will deploy the sample Python web application into your bluemix account. A toolchain will be set up for continuous delivery of the application. The application code will be cloned from a public Git repository into a private Git repo in your account that will be set up as part of the toolchain. Each time you commit changes to the repo, the app will be built and deployed.

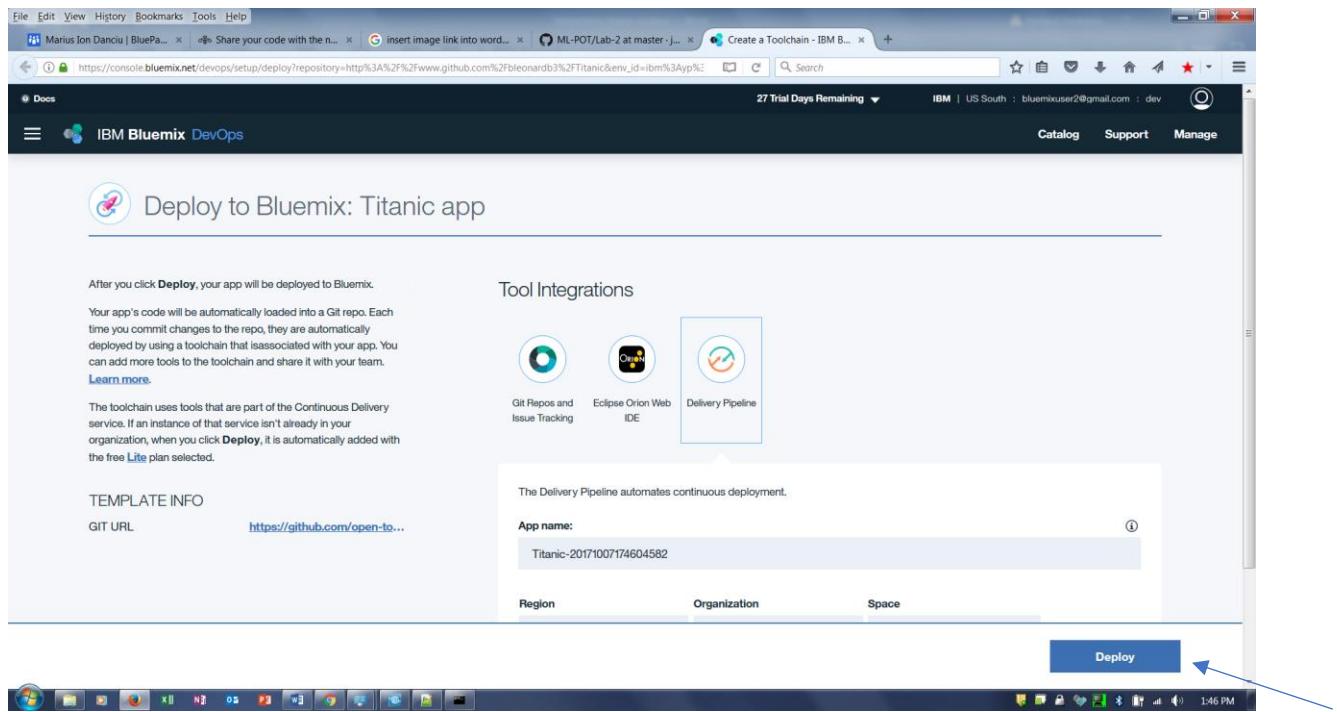
The toolchain uses tools that are part of the Continuous Delivery service. If an instance of that service isn't already in your organization, when you click **Deploy**, it is automatically added with the free [Lite](#) plan selected.

You will need to customize the application to provide the credentials for your Watson Machine Learning service, and to provide the scoring endpoint.

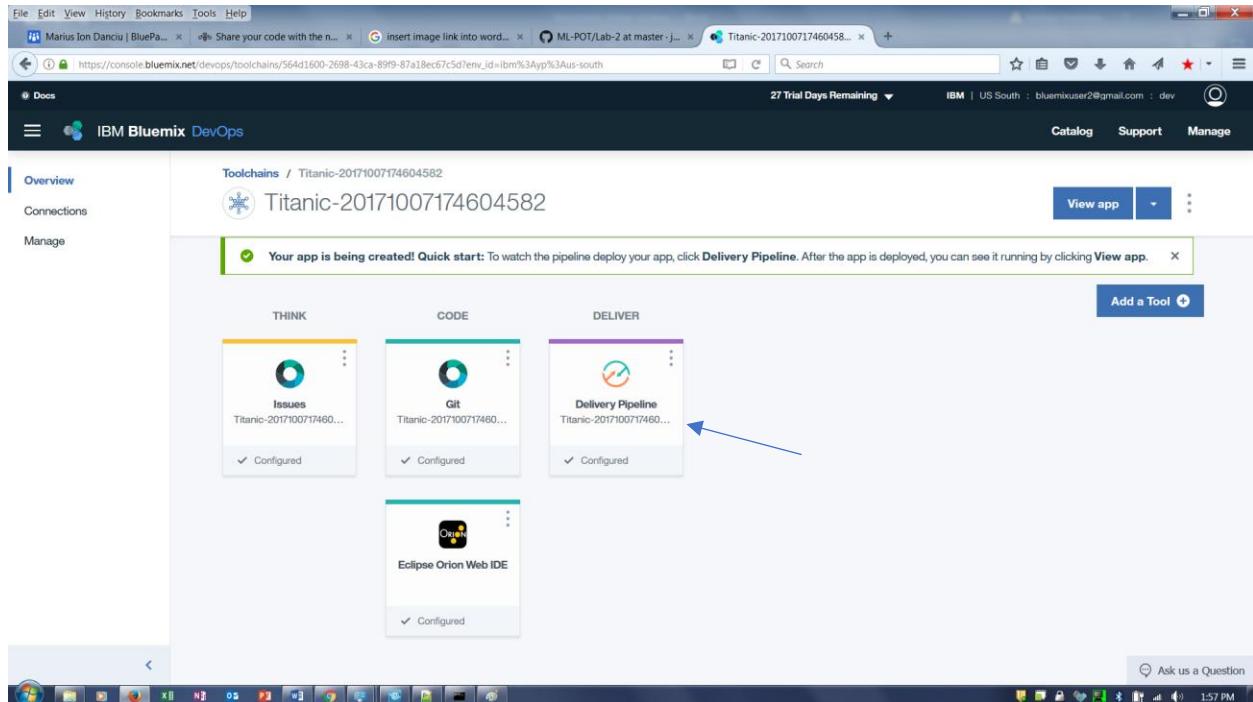
1. Click on the following to deploy a sample Python Flash web application into your bluemix account. Note you may get a message to sign in to Bluemix.

Deploy to Bluemix

2. Click on the **Deploy** button.

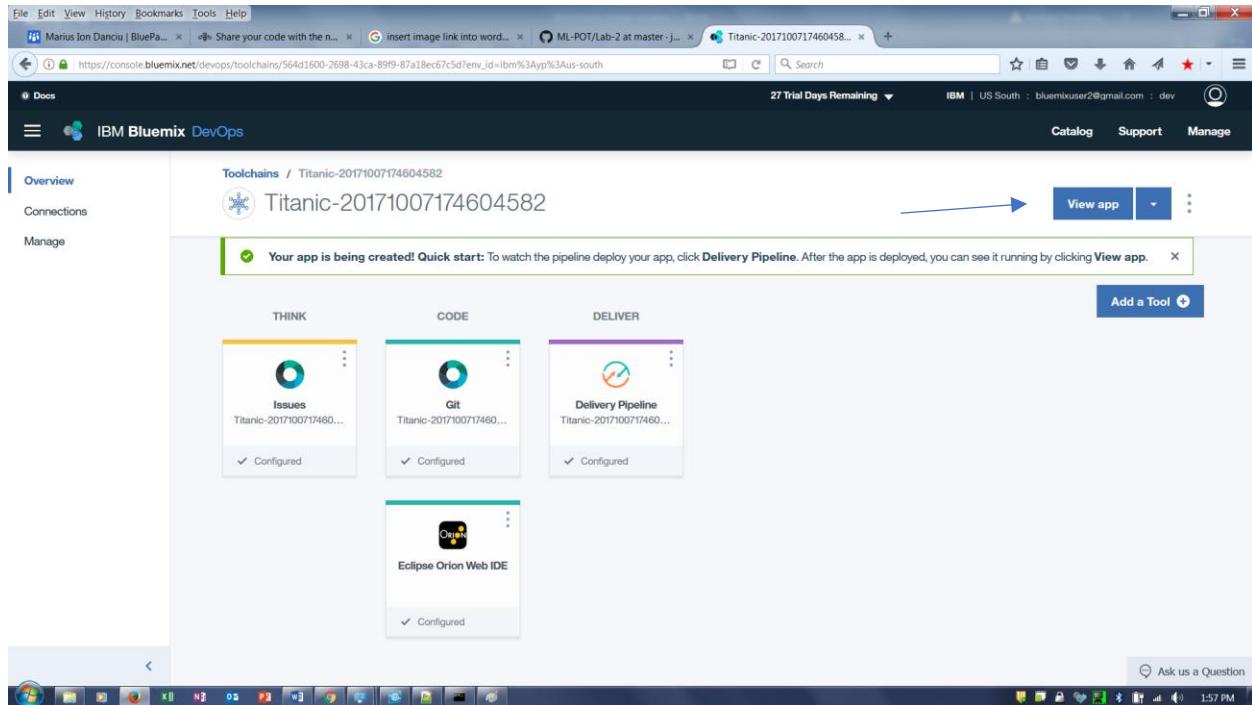


3. Your app is being created! To watch the pipeline deploy your app, click **Delivery Pipeline**.

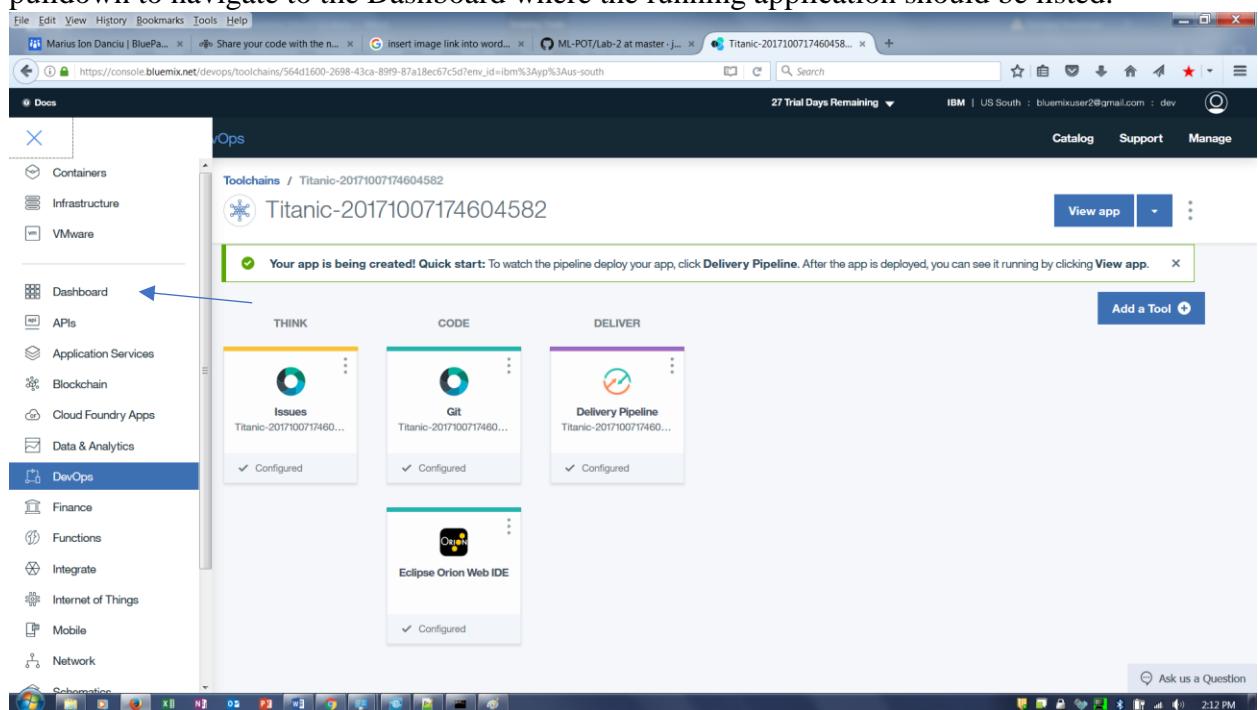


4. After the app is deployed successfully (should say Deployed passed now in the Deploy stage), return to the Delivery Pipeline by clicking on the vertical ellipse and click on View Toolchain.

5. You can see the running app by clicking **View app**. The web form collecting the Titanic passenger data should appear. Note that the application is not functional until we connect it to the Watson Machine Learning service.



6. Close the Titanic prediction app tab, and click on the  icon and Dashboard in the pulldown to navigate to the Dashboard where the running application should be listed.



7. We are now going to connect the application to the Watson Machine Learning service that was created earlier. Click on the application name.

The screenshot shows the IBM Bluemix Dashboard with the 'Cloud Foundry Apps' section selected. The table lists one application:

NAME	ROUTE	MEMORY (MB)	INSTANCES	RUNNING	STATE	ACTIONS
Titanic-20171007174604582	titanic-20171007174604582-jackleg-pirogue.myblu...	128	1	1	Running	

Below the main table is a 'Services' section:

NAME	SERVICE OFFERING	PLAN	ACTIONS
Continuous Delivery	Continuous Delivery	Free (Deprecated)	
Data Science Experience-gz	Data Science Experience	Lite	
DSX-ObjectStorage	Object Storage	Lite	
DSX-Spark	Apache Spark	Lite	
Watson Machine Learning	Machine Learning	Free	

8. Scroll down until you see the Connections panel. Click on Connect Existing.

The screenshot shows the 'Cloud Foundry Apps' page for the application 'Titanic-20171007174604582'. The left sidebar has 'Overview' selected. The main area displays application details and runtime cost information. The 'Connections' panel shows:

- Buildpack: Python
- Instances: 1 (All instances are running, Health is 100%)
- MB Memory per Instance: 128
- Total MB Allocation: 1,875 GB still available

The 'Connections' panel contains the following text and buttons:

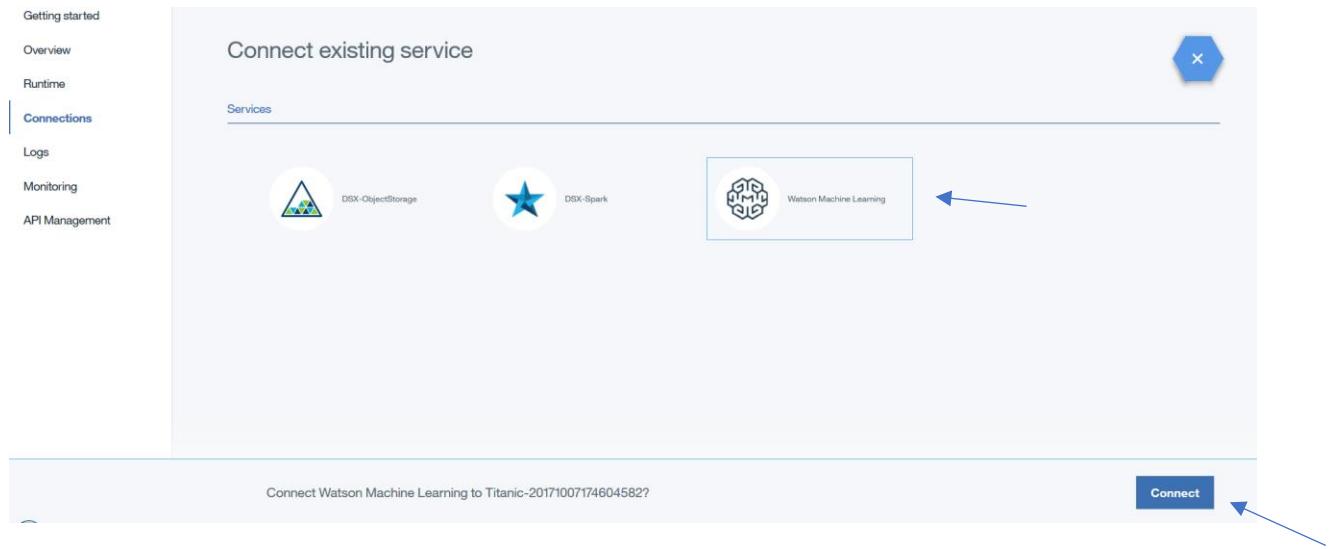
No services are connected to this app
You can create or bind a service:

Connect new **Connect existing**

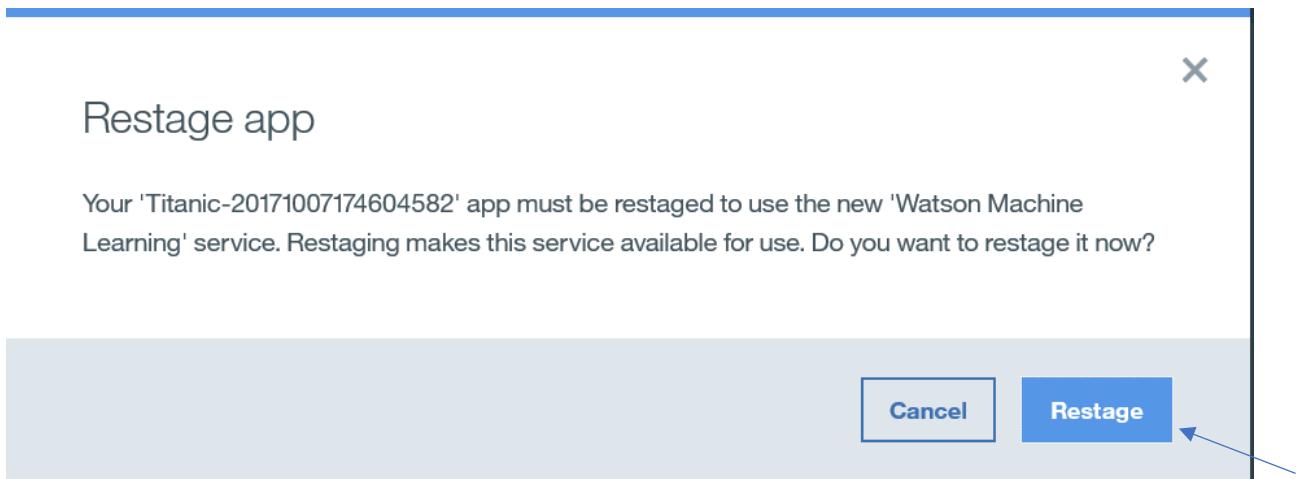
Runtime cost:

\$0.00	\$0.00
Current charges for billing period	Estimated total for billing period (Oct 1, 2017 - Oct 31, 2017)
Current and estimated cost excludes connected services.	
View full usage details	

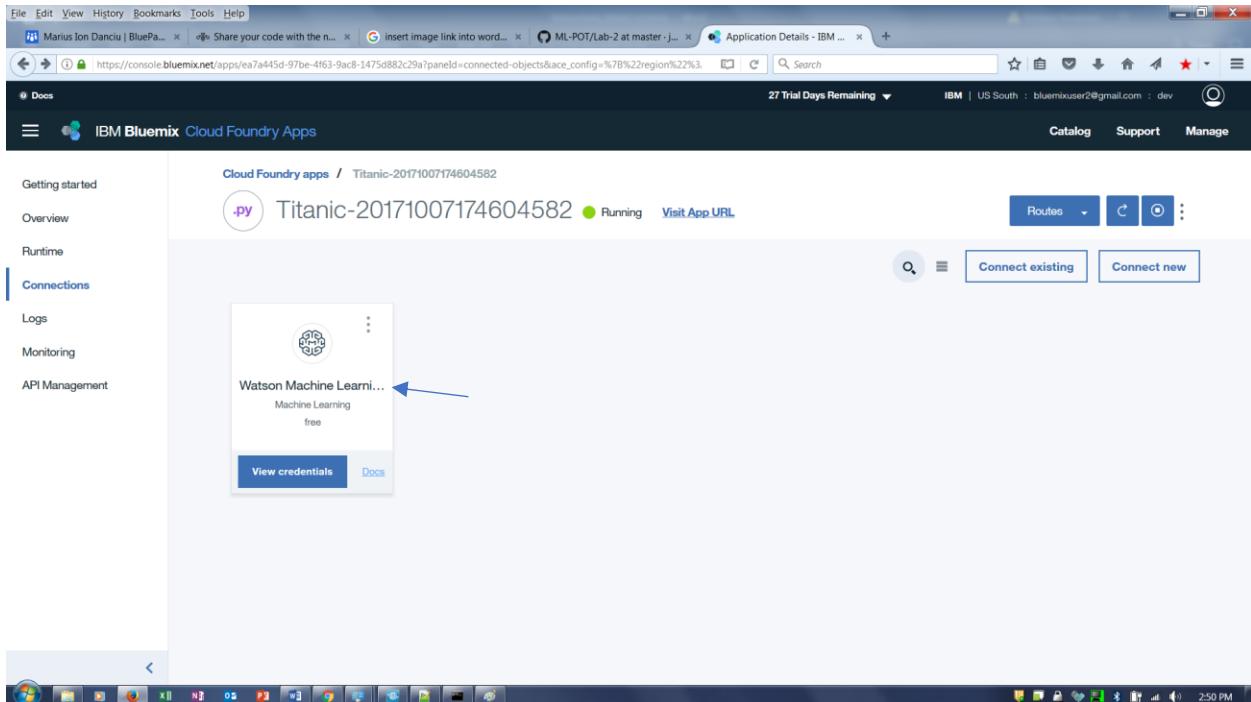
9. You should see at least 3 services listed. Two services are automatically provisioned for the Data Science Experience (Spark service, and Object Storage service). Click on the **Watson Machine Learning** service, and then click on **Connect**.



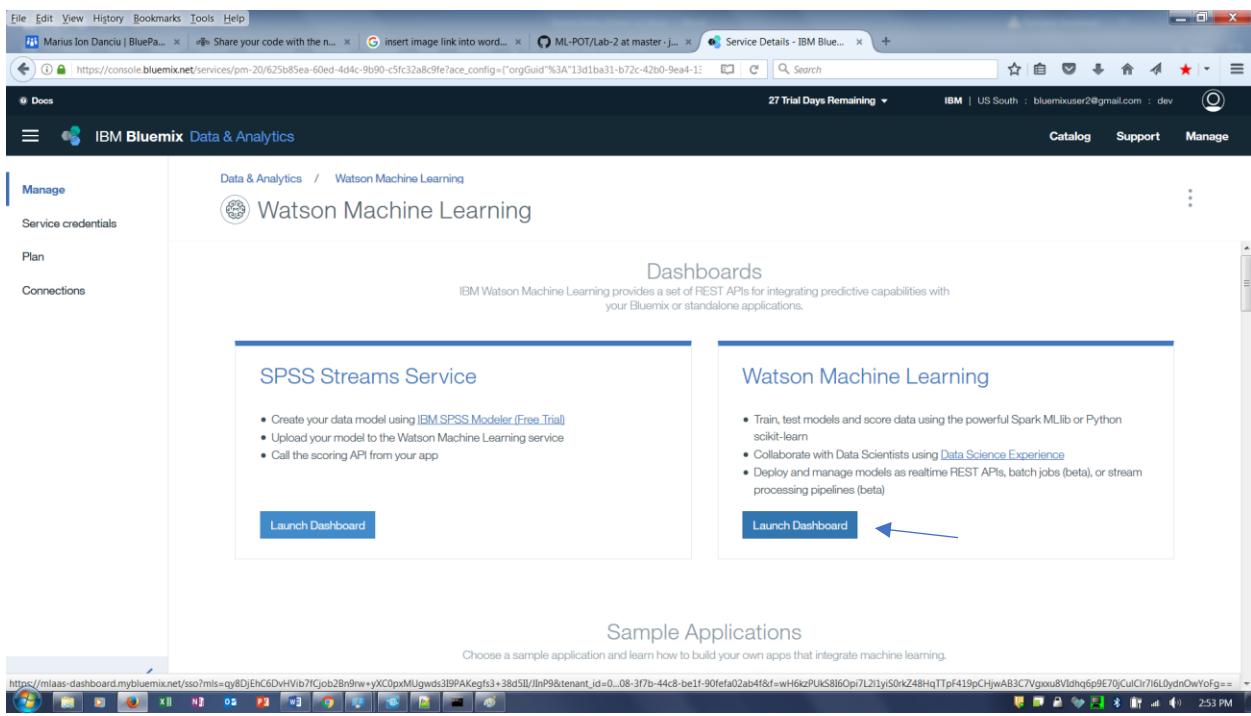
10. You will get a pop up that asks to Restage the application. Click on **Restage**.



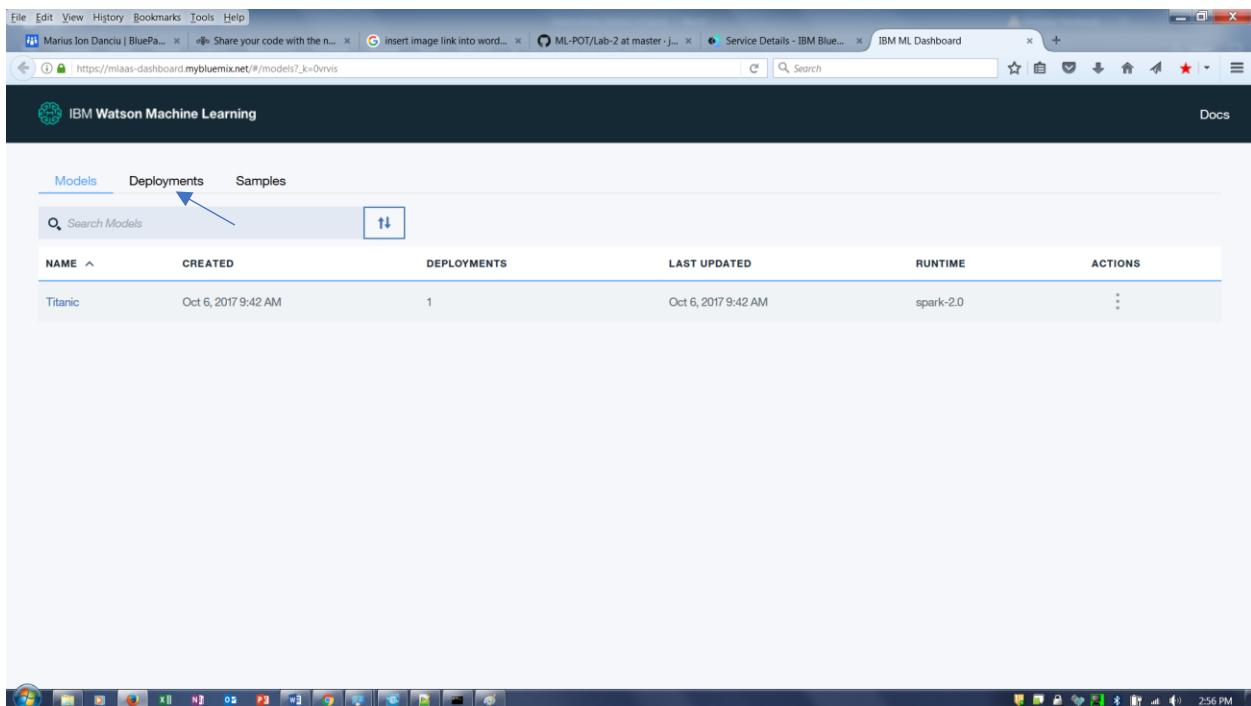
11. We now need to copy the scoring endpoint to the clipboard. Click on the **Watson Machine Learning** service.



12. Click on the Watson Machine Learning service **Launch Dashboard**.

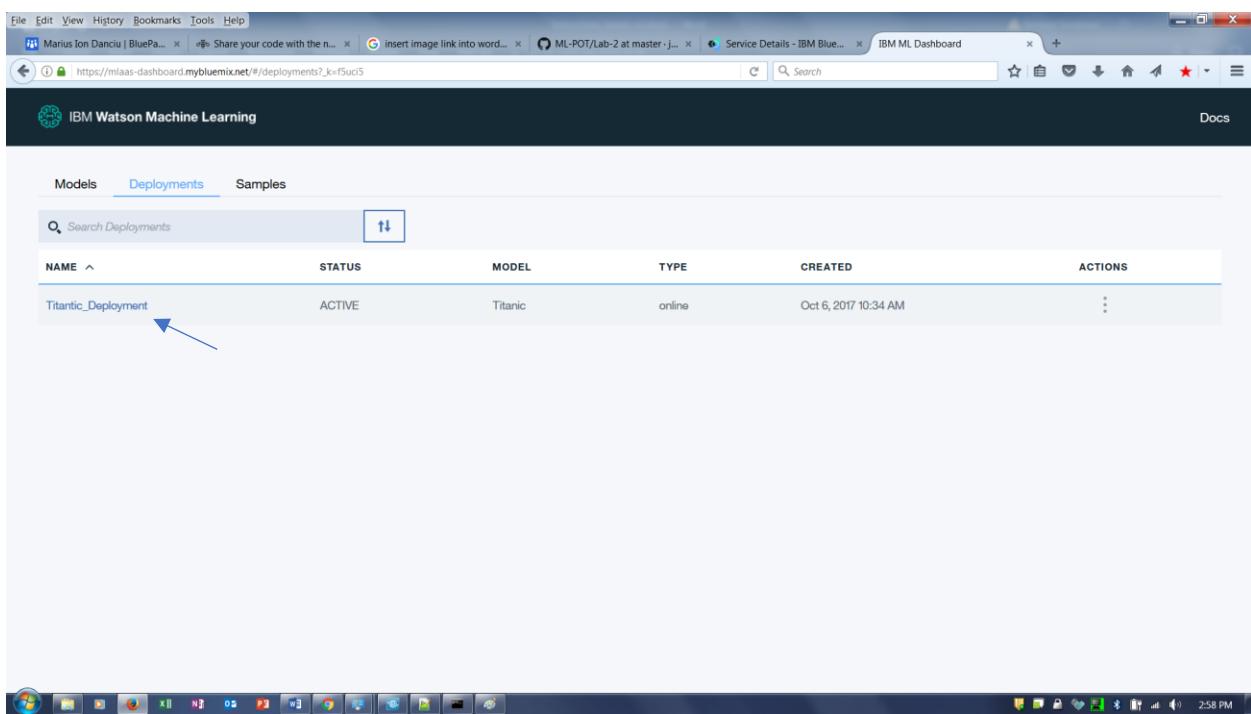


13. Click on the **Deployment** tab.



The screenshot shows the IBM Watson Machine Learning dashboard. The top navigation bar includes links for 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Tools', and 'Help'. The main title is 'IBM Watson Machine Learning'. Below the title, there are three tabs: 'Models' (selected), 'Deployments' (highlighted with a blue arrow), and 'Samples'. A search bar labeled 'Search Models' is present. The main content area displays a table with columns: NAME, CREATED, DEPLOYMENTS, LAST UPDATED, RUNTIME, and ACTIONS. One row is visible for 'Titanic', created on Oct 6, 2017 at 9:42 AM, with 1 deployment, last updated on Oct 6, 2017 at 9:42 AM, using spark-2.0 runtime. The bottom status bar shows various icons and the time as 2:56 PM.

14. Click on the Titanic Deployment.



The screenshot shows the IBM Watson Machine Learning dashboard with the 'Deployments' tab selected. The top navigation bar and title are identical to the previous screenshot. The main content area displays a table with columns: NAME, STATUS, MODEL, TYPE, CREATED, and ACTIONS. One row is visible for 'Titanic_Deployment', which is ACTIVE, uses the 'Titanic' model, is of type 'online', was created on Oct 6, 2017 at 10:34 AM, and has an 'Actions' button. The bottom status bar shows various icons and the time as 2:58 PM.

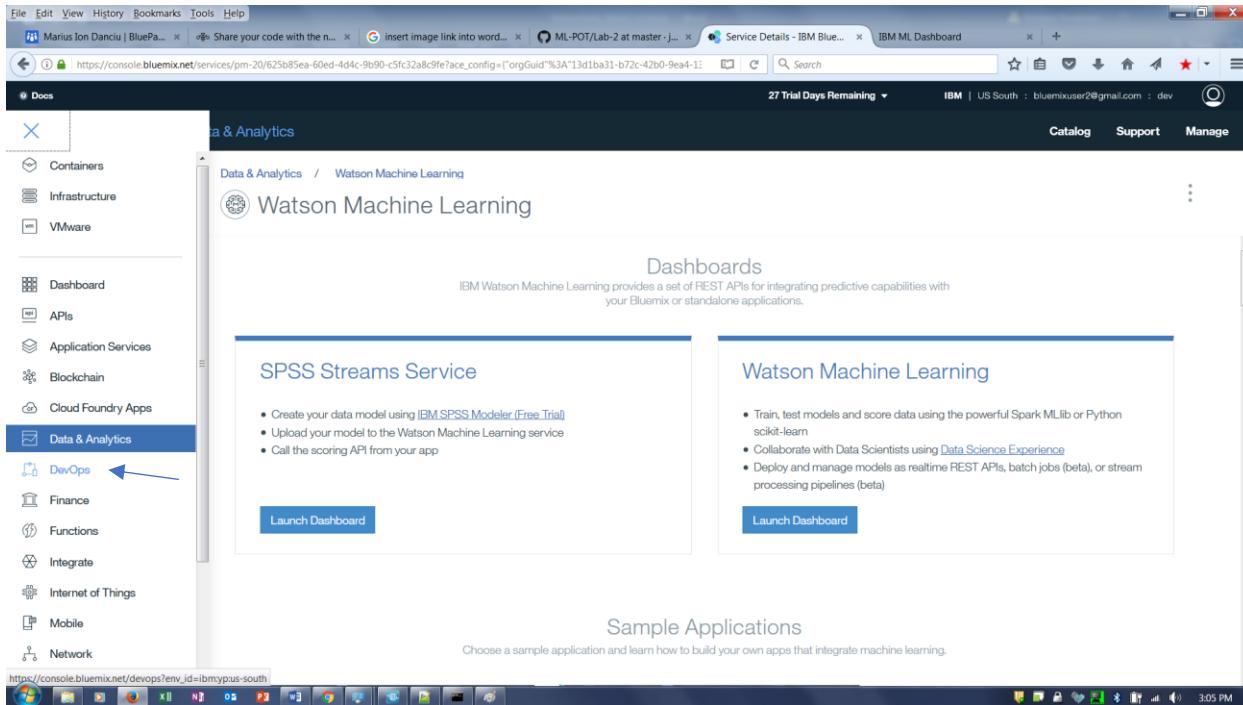
15. Scroll down to API details and select the Scoring Endpoint “http....etc”

Deployment Name	Titanic_Deployment
Deployment Type	online
Deployment Id	7e591f82-e004-46e7-b35d-c9fb062ac8a
Deployment Status	ACTIVE
URL	https://ibm-watson-ml.mybluemix.net/v3/wml_instances/625b85ea-60ed-4d4c-9b90-c5fc32a8c9fe/published_models/0788b520-0f19-4f7c-9ee5-03a9f7ed899d/deployments/7e591f82-e004-46e7-b35d-c9fb062ac8a
Model Name	Titanic
Model URL	https://ibm-watson-ml.mybluemix.net/v3/wml_instances/625b85ea-60ed-4d4c-9b90-c5fc32a8c9fe/published_models/0788b520-0f19-4f7c-9ee5-03a9f7ed899d
Created	Oct 6, 2017 10:34 AM
Last Updated	Oct 6, 2017 10:34 AM

API Details

Scoring Endpoint: https://ibm-watson-ml.mybluemix.net/v3/wml_instances/625b85ea-60ed-4d4c-9b90-c5fc32a8c9fe/published_models/0788b520-0f19-4f7c-9ee5-03a9f7ed899d/deployments/7e591f82-e004-46e7-b35d-c9fb062ac8a/online

16. Close the Deployment screen, and Select the Service Details tab. Click on the  icon, and click on DevOps in the pulldown to navigate to the Toolchain.



The screenshot shows the IBM ML Dashboard interface. The top navigation bar includes 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Tools', and 'Help'. The current tab is 'Service Details - IBM Blue...'. The main content area displays deployment details for 'Titantic_Deployment' with a scoring endpoint URL. Below this, there's a 'Dashboards' section for Watson Machine Learning. On the left, a sidebar lists various service categories: 'Containers', 'Infrastructure', 'VMware', 'Dashboard', 'APIs', 'Application Services', 'Blockchain', 'Cloud Foundry Apps', 'Data & Analytics', 'DevOps' (which is highlighted with a blue arrow), 'Finance', 'Functions', 'Integrate', 'Internet of Things', 'Mobile', and 'Network'. A second blue arrow points to the 'DevOps' link in the sidebar.

17. We are now going to paste the scoring endpoint into the application code. Click on the Toolchain.

The screenshot shows the IBM Bluemix DevOps interface. The left sidebar has 'Toolchains' selected. The main area is titled 'Toolchains' and shows a table with one row: 'Titanic-20171007174604582'. A blue arrow points to this row. The top right of the main area has a 'Create a Toolchain' button.

18. Click on the Eclipse Orion Web IDE.

The screenshot shows the 'Toolchains / Titanic-20171007174604582' page. It displays three tool components: 'Issues' (configured), 'Git' (configured), and 'Delivery Pipeline' (configured). Below them is an 'Eclipse Orion Web IDE' component, also marked as configured. A blue arrow points to the 'Eclipse Orion Web IDE' component.

19. Click on the Titanic_v1 file.

This repository contains a Python Flask program that invokes a Watson Machine Learning service to predict survival based on passenger information from the Titanic.

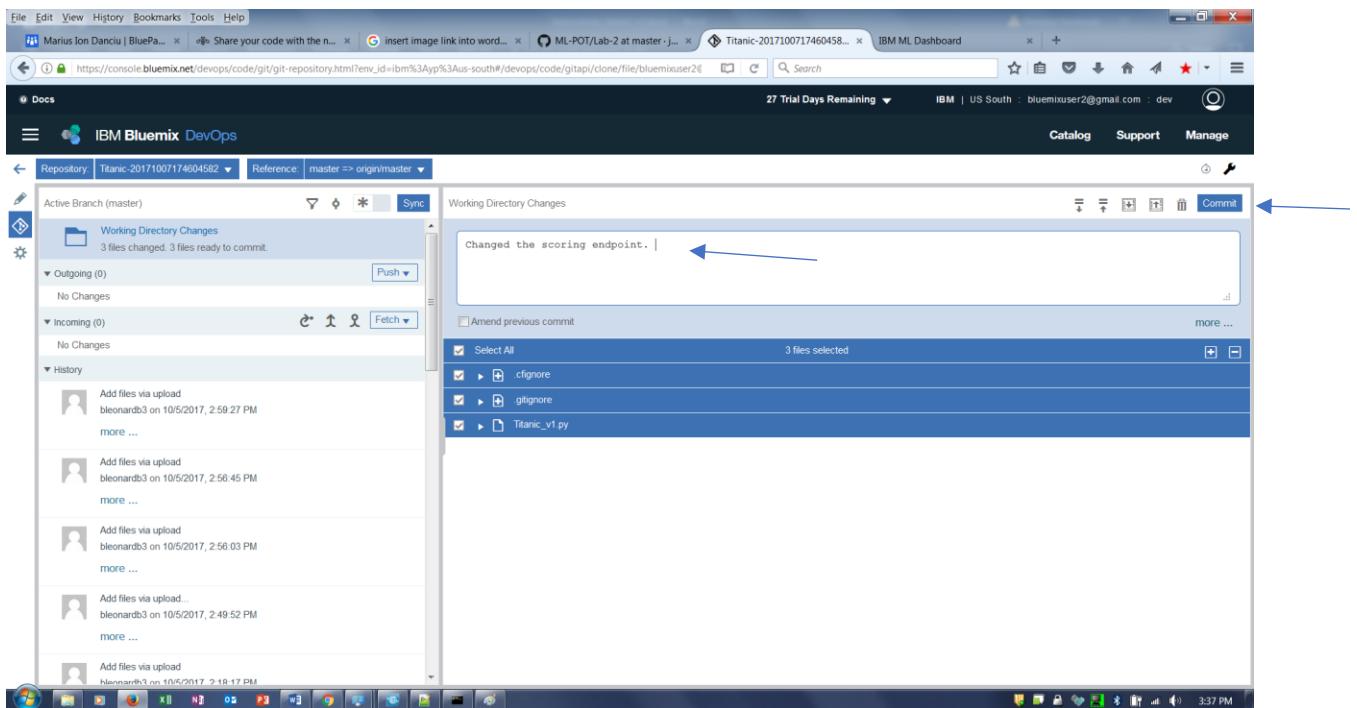
- static
- templates
- cignore
- gitignore
- LICENSE
- manifest.yml
- NOTICE
- Profile
- README.md
- requirements.txt
- Titanic_v1.py

20. Look around line 20 for the “scoring endpoint =”. Select the current scoring endpoint (may want to use Shift-End to get to the end of the line. Make sure not to select the endpoint quote – if you do just make sure to put it back in). Enter Ctrl-V to paste the new scoring endpoint from your Watson Machine Learning service. Enter Ctrl-S to save the file. Then click on the  icon on the top left.

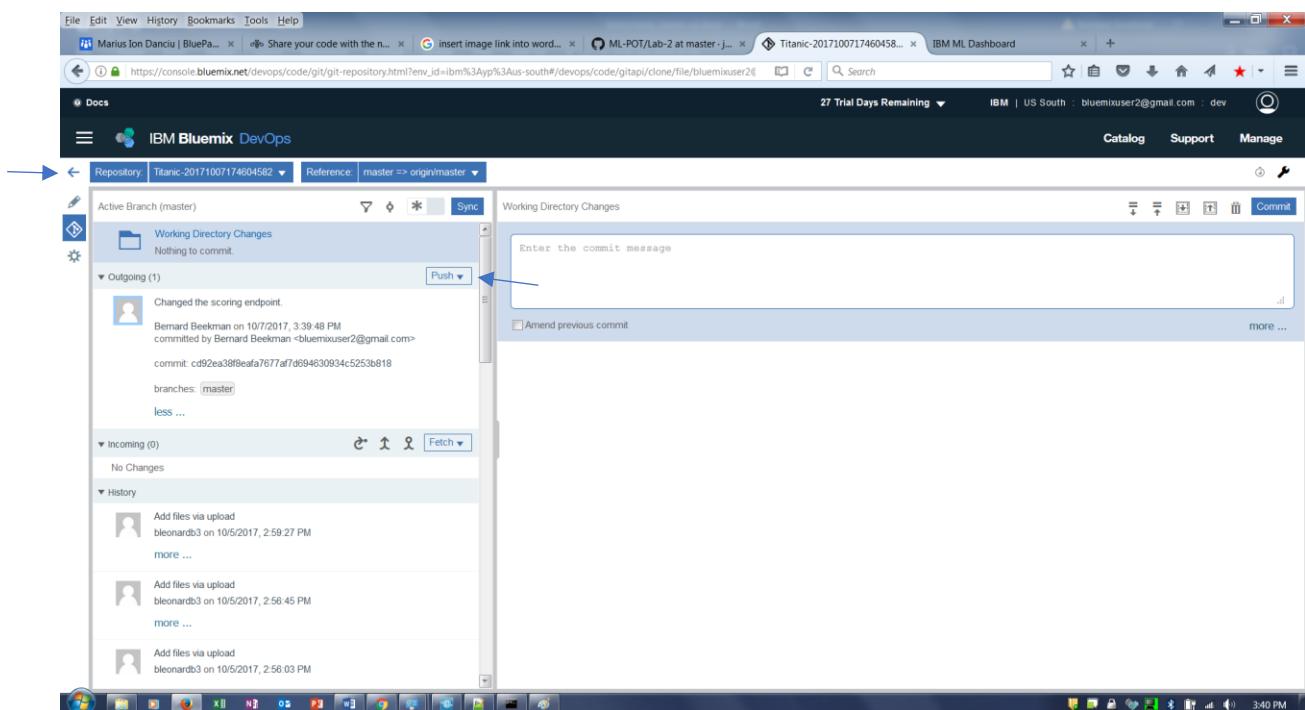
```

1 import urllib3, requests, json, os
2 app = Flask(__name__, static_folder='static')
3 from flask_bootstrap import Bootstrap
4 from flask_wtf import FlaskForm
5 from wtforms import StringField, SubmitField, RadioField, FloatField, IntegerField
6 from wtforms.validators import Required, Length, NumberRange
7
8 #url = 'https://ibm-watson-ml.mybluemix.net'
9 #username = '1e1f23c-ad34-4927-8283-a55f66caec00'
10 #password = 'ec34cd2-3809-4378-a24f-ee7e59b1dd3d'
11
12 if 'VCAP_SERVICES' in os.environ:
13     vcap = json.loads(os.getenv('VCAP_SERVICES'))
14     print('Found VCAP_SERVICES')
15     if 'wml': 
16         creds = vcap['wml'][0]['credentials']
17         username = creds['username']
18         password = creds['password']
19         url = creds['url']
20
21 scoring_endpoint = url + 'v3/wml_instances/625b85ea-60ed-4d4c-9b90-c5fc32a8c9fe/published_models/0788b520-0f19-4f7c-9ee5-03a9f7ed899d'
22 app.config['SECRET_KEY'] = 'secretpassw0rd'
23 bootstrap = Bootstrap(app)
24 class TitanicForm(FlaskForm):
25     pclass = RadioField('Passenger Class:', coerce=int, choices=[('1','First'),('2','Second'),('3','Third')])
26     name = StringField('Name:')
27     sex = RadioField('Sex:', coerce=str, choices=[('male','Male'),('female','Female')])
28     age = RadioField('Age:', coerce=int, choices=[('0','0-5'),('1','6-11'),('2','12-17'),('3','18-39'),('4','40-64'),('5','65-79'),('6','80+')])
29     ticket = StringField('Ticket:')
30     fare = FloatField('Fare:')
31     sibsp = IntegerField('Number of siblings/spouses:')
32     parch = IntegerField('Number of parents/children:')
33     embarked = RadioField('Embarked location:', coerce=str, choices=[('S','South Hampton'),('C','Cherbourg'),('Q','Queenstown')])
34     submit = SubmitField('Submit')
35 @app.route('/', methods=['GET', 'POST'])
36 def index():
37     form = TitanicForm()
38     if form.is_submitted():
39         sex = form.sex.data
40         name = form.name.data
41
    
```

21. The next step is to commit the change to the repository. Enter “Changed the Scoring Endpoint” in the Enter Commit Message field, and then click on **Commit**.



22. Then click on **Push** to push the changes to the Git repo which will start the build and deploy of the application. Click on the left arrow to return to the Toolchain.



23. Click on the **Delivery Pipeline** to view status of the deployment as before. Once the Deployment status shows **Deploy passed now**, click on the vertical ellipsis and then click

on the **View Toolchain** option to return to the Toolchain screen. Click on the **View Apps** button. (see Steps 3,4,5 above as a reminder if necessary). The web form should appear. Enter data in all the fields and click on the **Submit** button. (the submit button is located at the bottom of the web form – you may need to scroll).

To determine the survival prediction,please enter the following:

Passenger Class:

First
 Second
 Third

Name: Bernie Beekman

Gender:

Male
 Female

Number of siblings/spouses: 1

Number of parents/children: 1

Ticket: 1234

Fare: 23

Embarc Location:

South Hampton
 Cherbourg
 Queenstown

Age:

0-5
 6-11
 12-17

24. You should see something similar to the following depending on the values of the input fields that you entered. Click on the **Try Again!**, if you want to experiment with different inputs.

Titanic Prediction

```
prediction:survived
probability: 0.827966430684
```

[Try Again!](#)