

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:

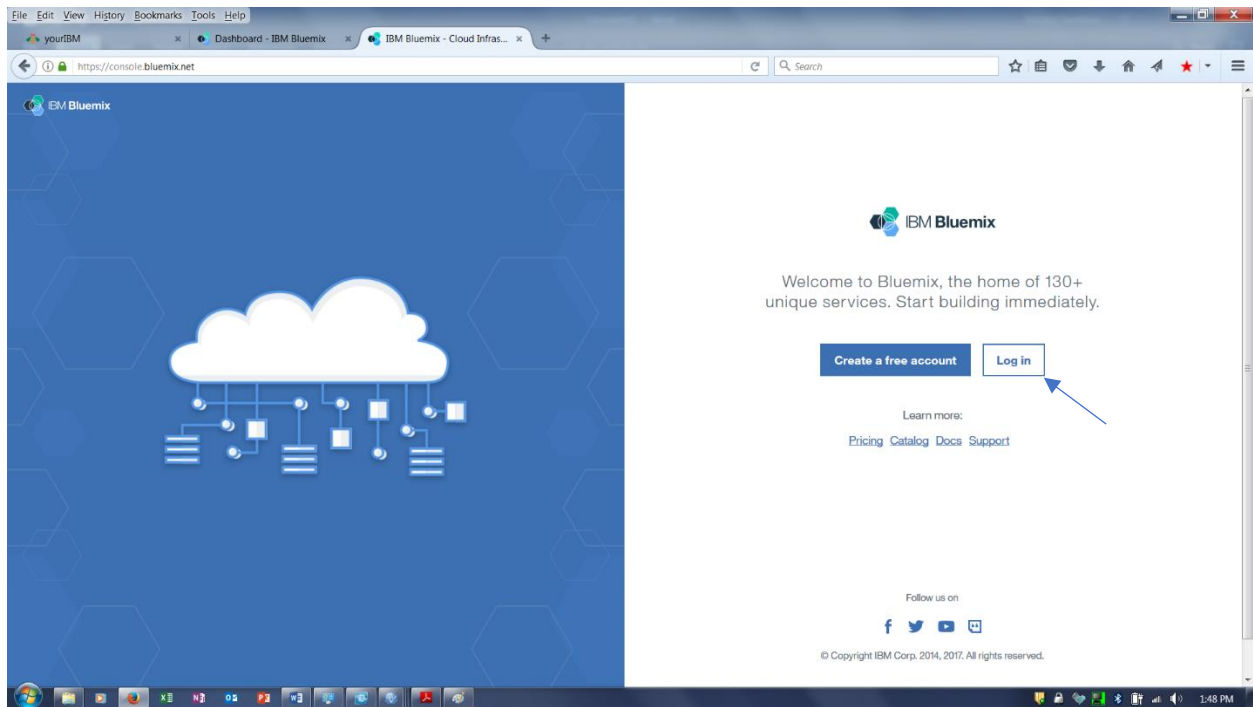
- Watson Machine Learning
- Object Storage
- Apache Spark

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.

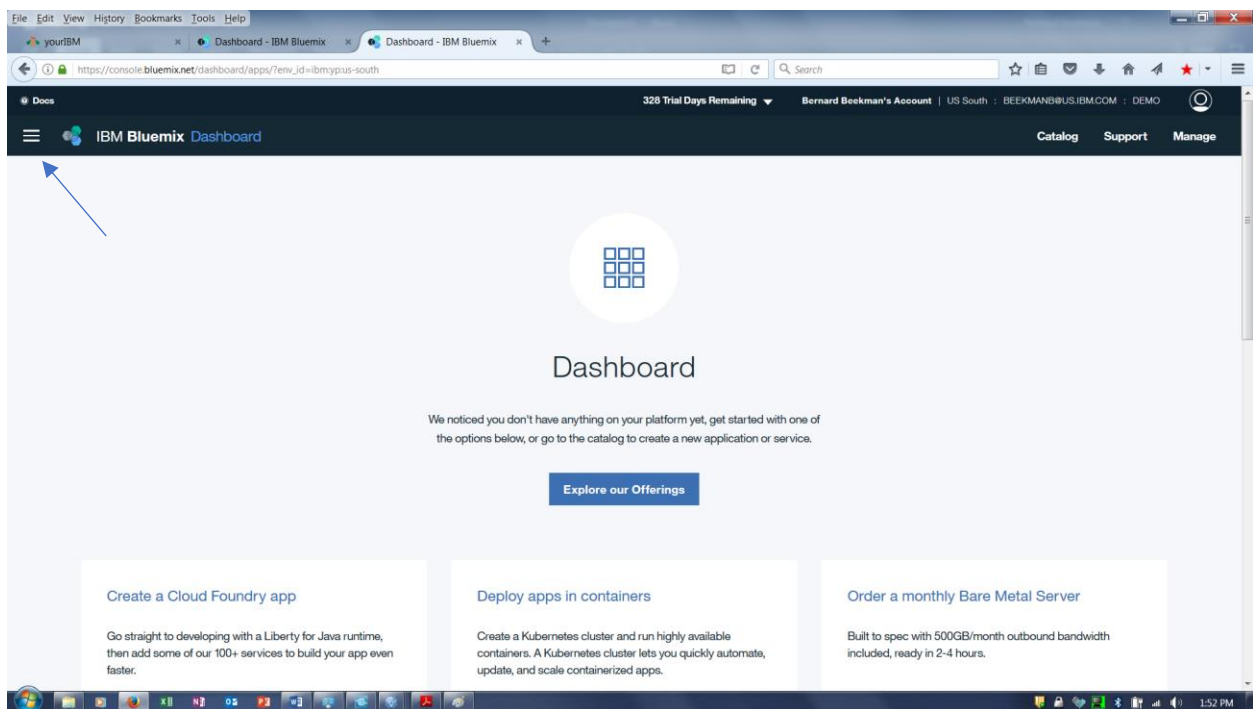
Step 1.1: Creating a Machine Learning Instance

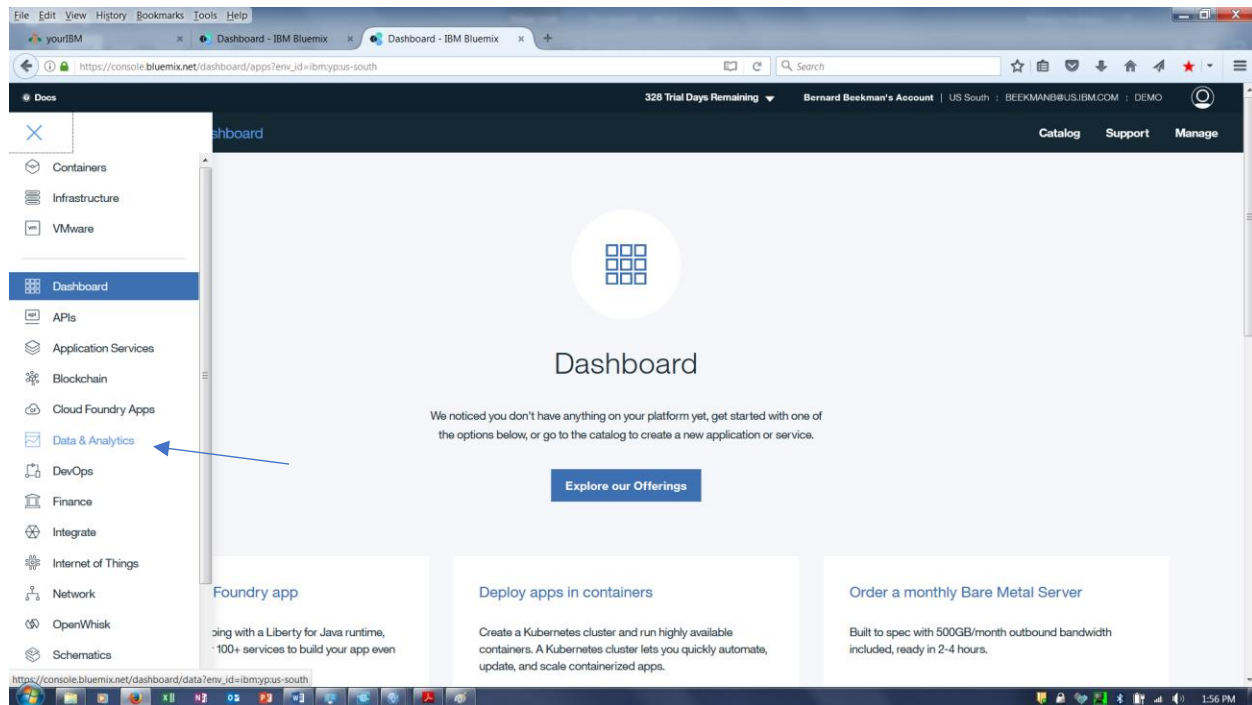
To create a Machine Learning service instance, you must perform the following steps:

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



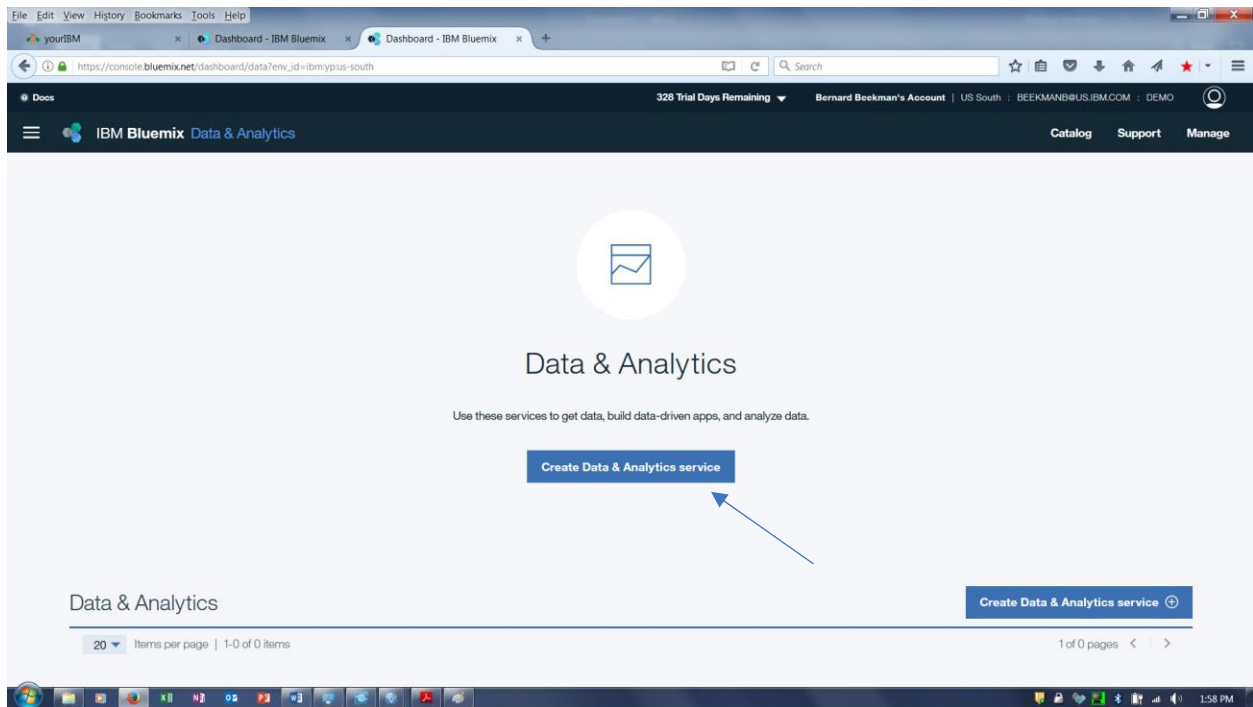
2. Once logged in, click on the hamburger icon, and from the navigation panel, click **Data & Analytics**.



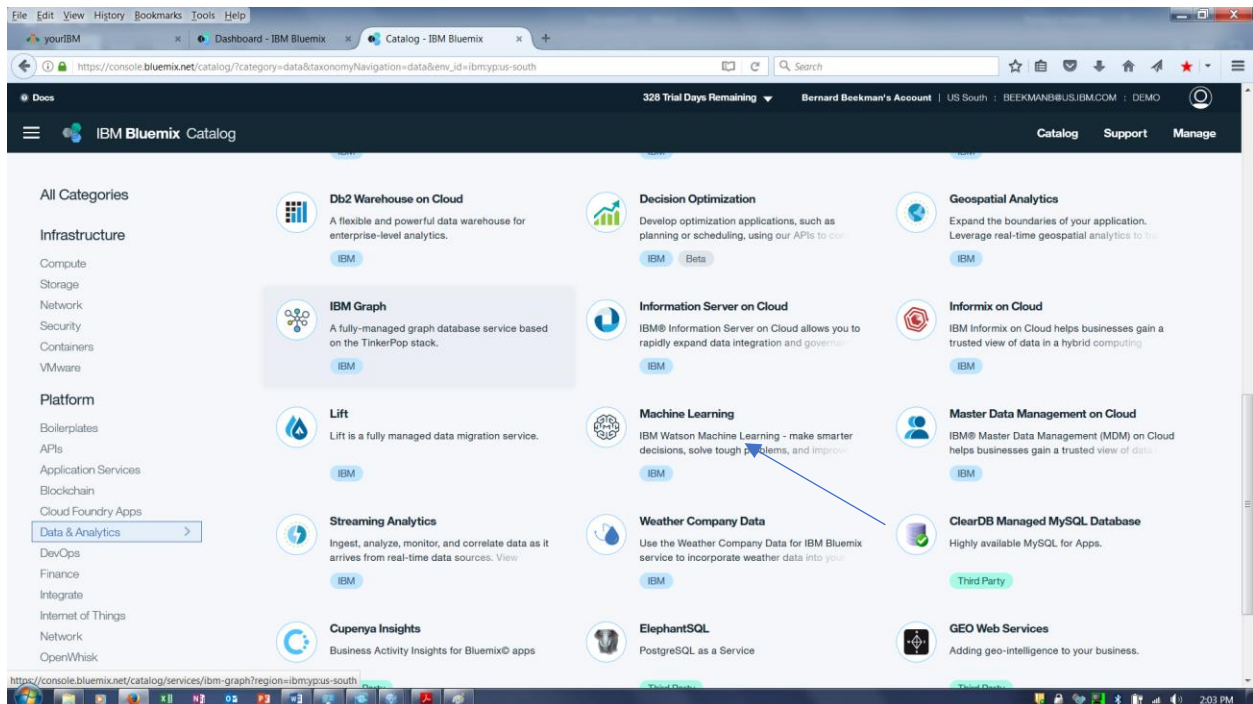


You see a screen centered on data services. You can return here regularly to work with your data and analytics services from one easy-to-use page. Check to see if a Machine Learning service already exists. If not, continue, otherwise go to Step 1.2: Adding existing Bluemix instances to a project in Data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

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



4. Scroll down to Machine Learning and click.



5. Configure service.

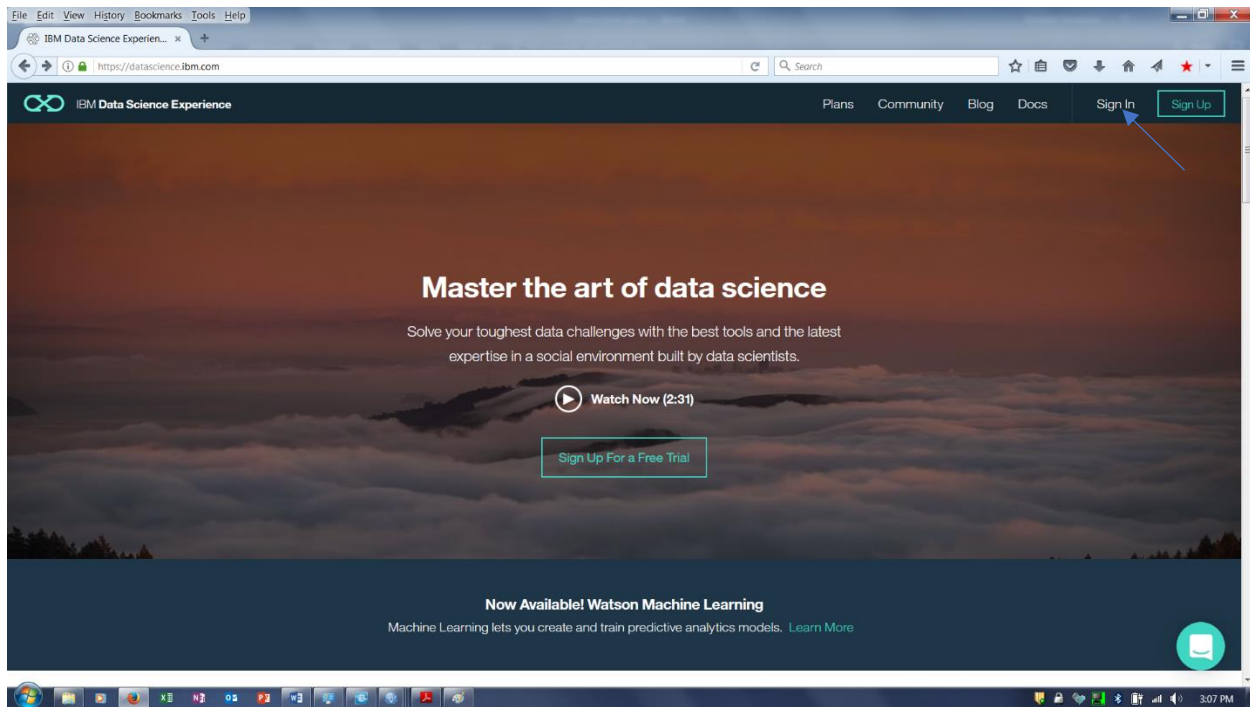
Enter a descriptive name for your service, choose a space, and select your data plan (find plan comparison and pricing details on this page). Click on **Create**.

The screenshot shows the IBM Bluemix Catalog interface for creating a new 'Machine Learning' service. The page includes a sidebar with 'View all' and 'View Docs' links, and a main content area with a description of the service. The 'Service name' field is set to 'Machine Learning'. The 'Credential name' field is set to 'Credentials-1'. The 'Select region to deploy in:' dropdown is set to 'US South'. The 'Choose an organization:' dropdown is set to 'BEEKMANB@US.IBM.COM'. The 'Choose a space:' dropdown is set to 'DEMO'. The 'Connect to:' dropdown is set to 'Leave unbound'. A 'Create' button is located at the bottom right of the form. Arrows point to the 'Machine Learning' service name, 'Credentials-1' credential name, 'US South' region, 'BEEKMANB@US.IBM.COM' organization, 'DEMO' space, and the 'Create' button.

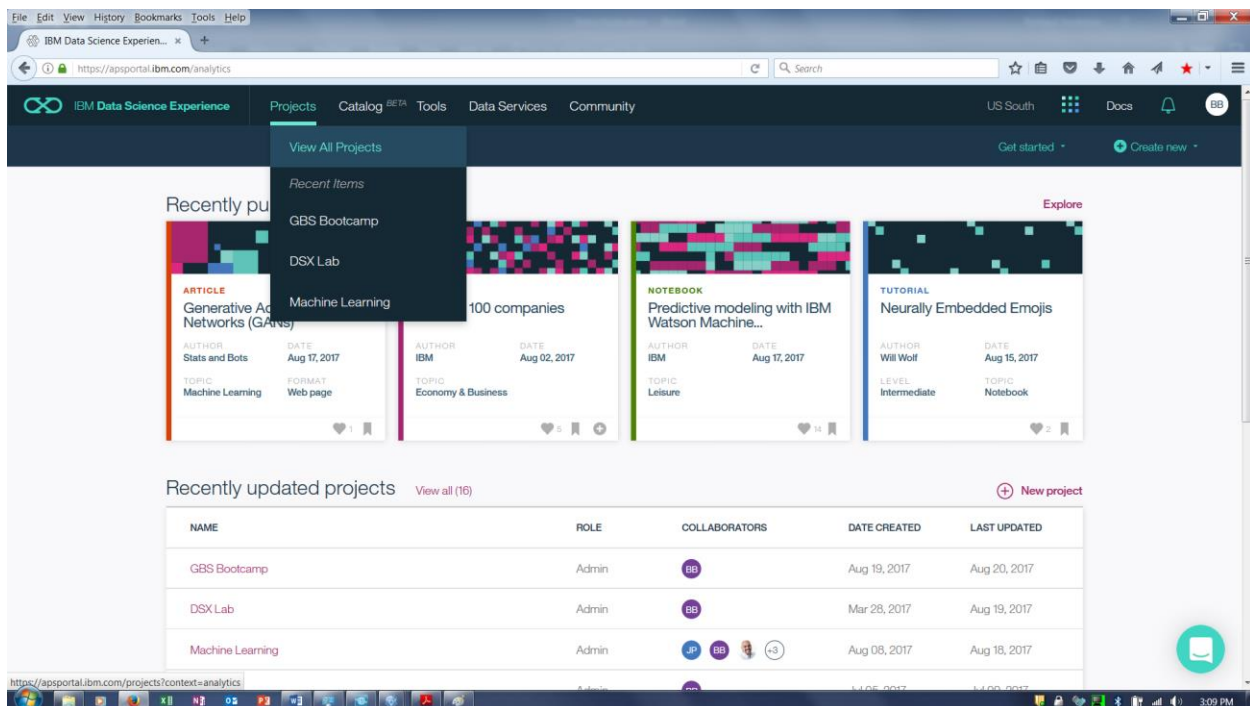
Step 1.2: Adding existing Bluemix instances to a project in Data Science Experience

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

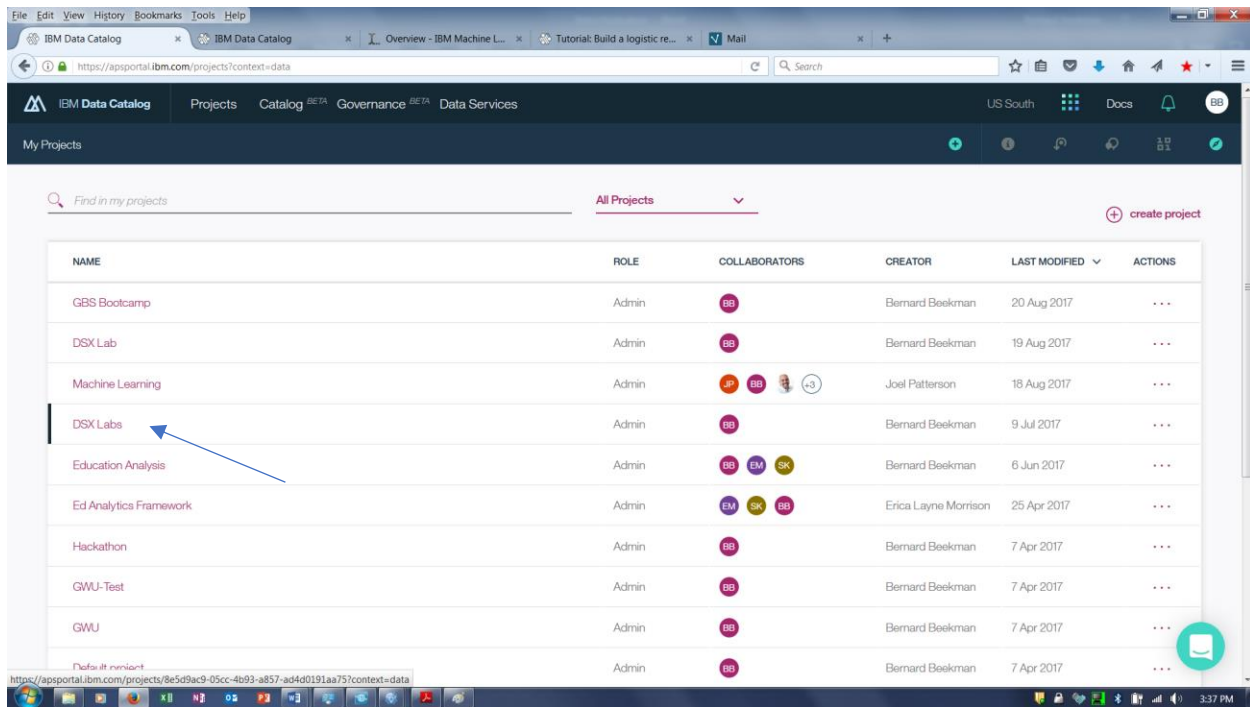
1. Log on to IBM Data Science Experience – <https://datascience.ibm.com>



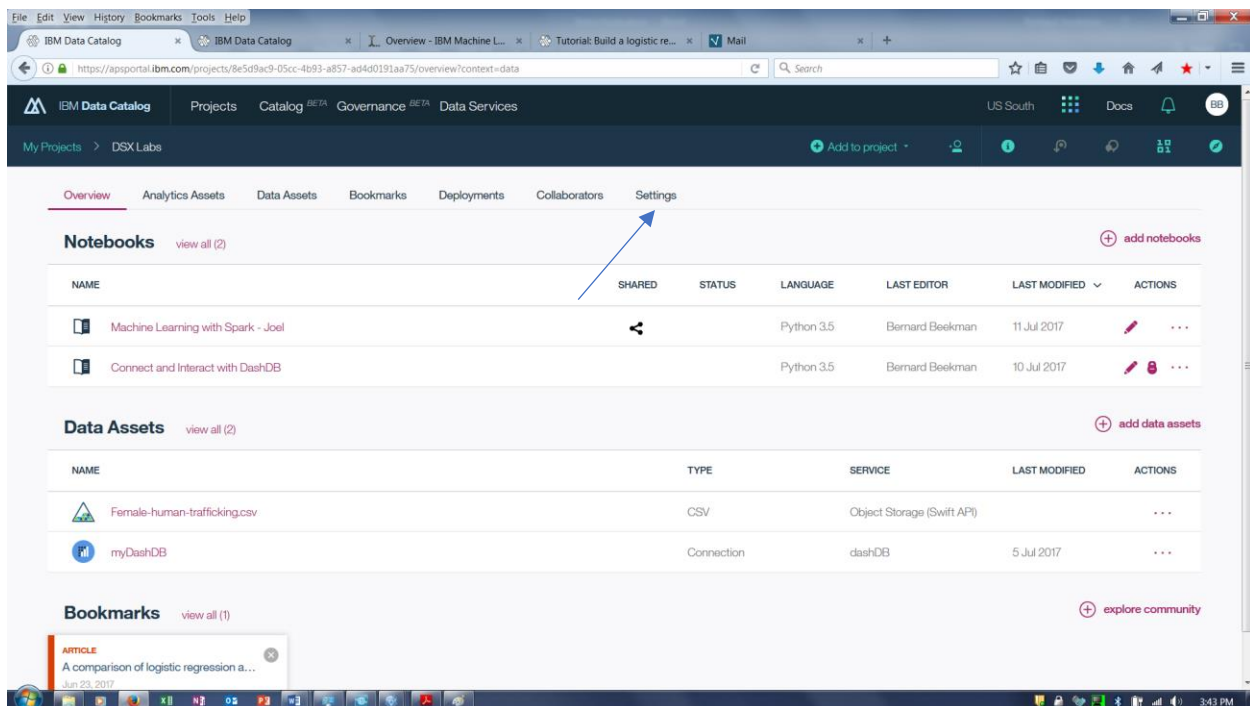
2. Click **Projects > View All Projects**.



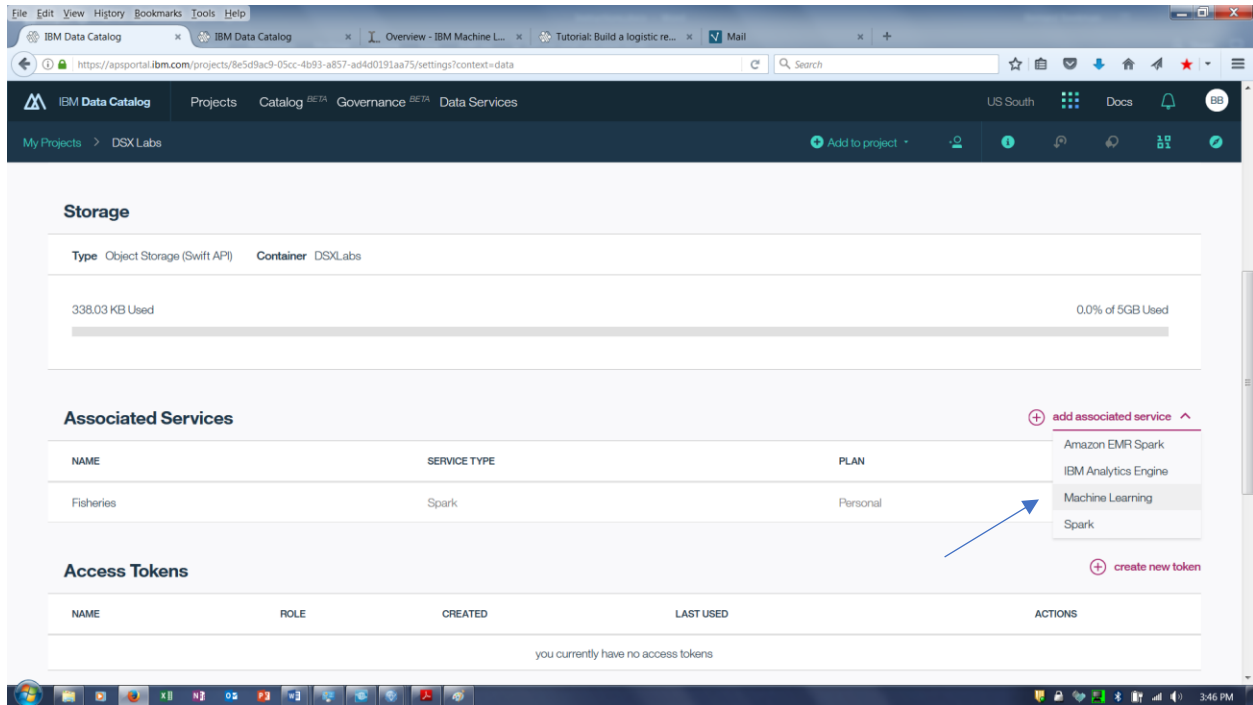
3. Click on the project that you created in the prerequisites, or if no project was created you can either create one, or click on the default project. (For the remainder of this document, I'm assuming the project name is DSX-Labs).



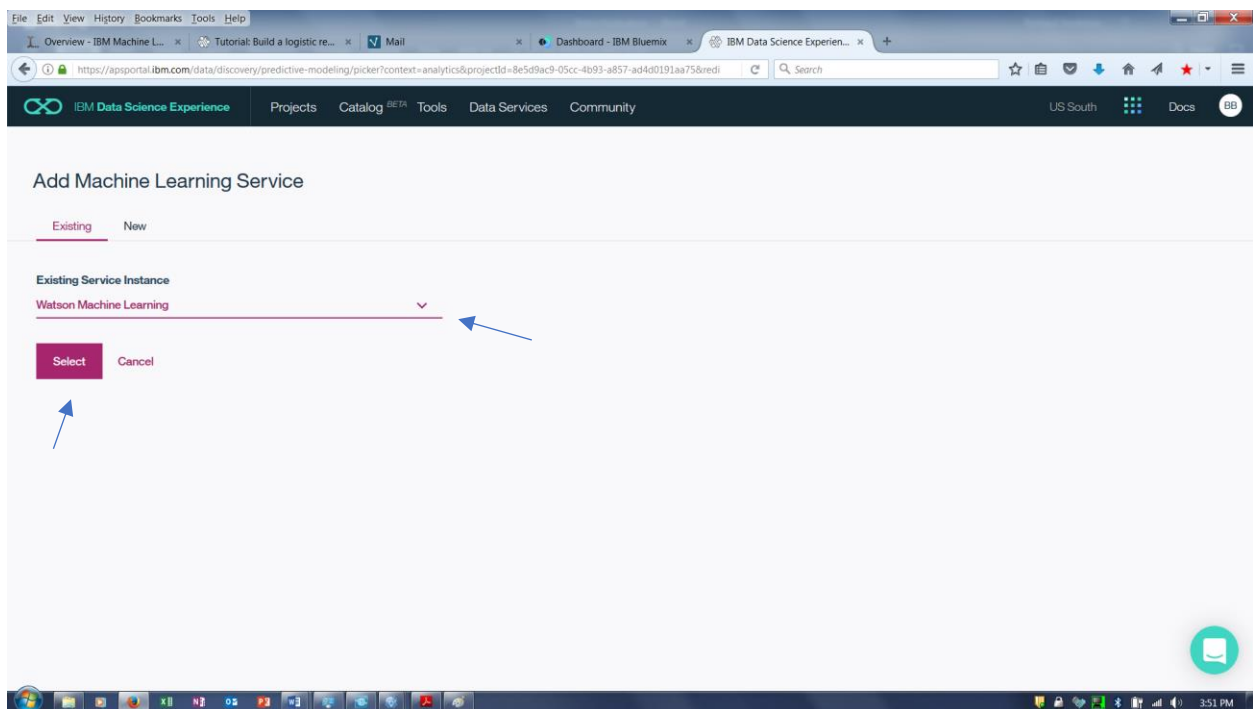
4. Select the **Settings** Tab.



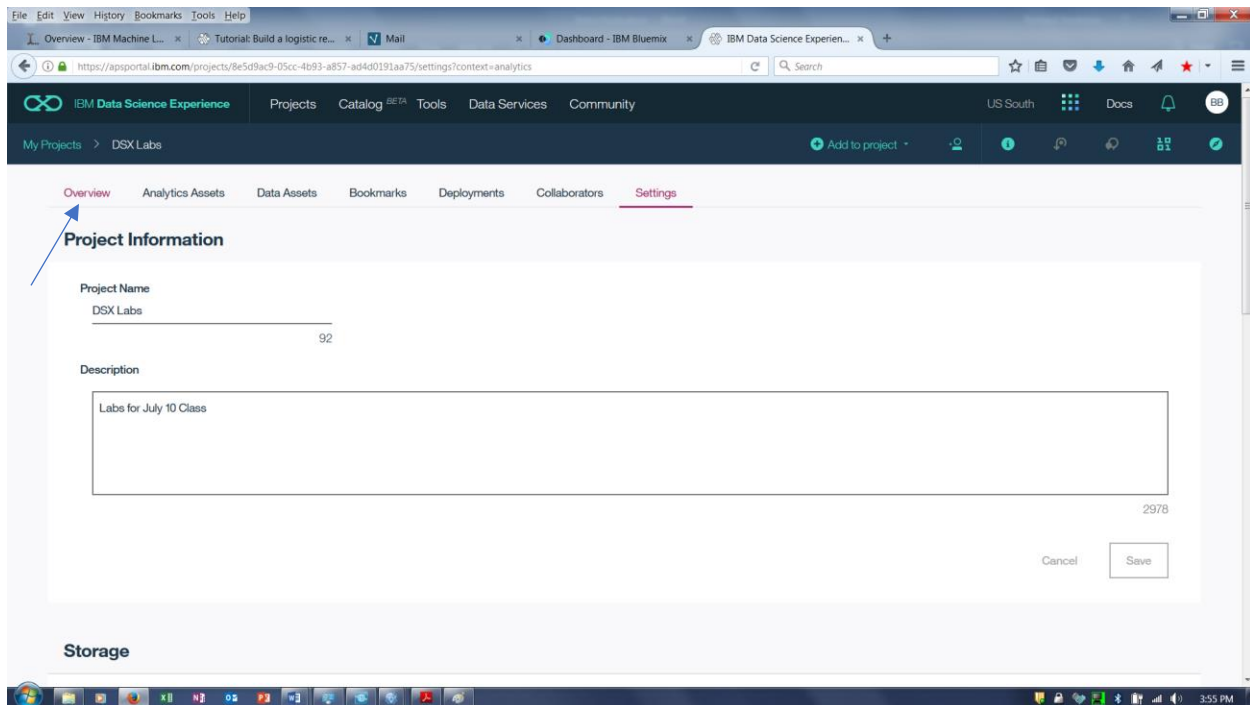
5. Scroll down to Associated Services. To add a service, in the **Associated Services** panel, click **add associated service**, select the Machine Learning service.



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



7. Click on the Project **Overview** tab.



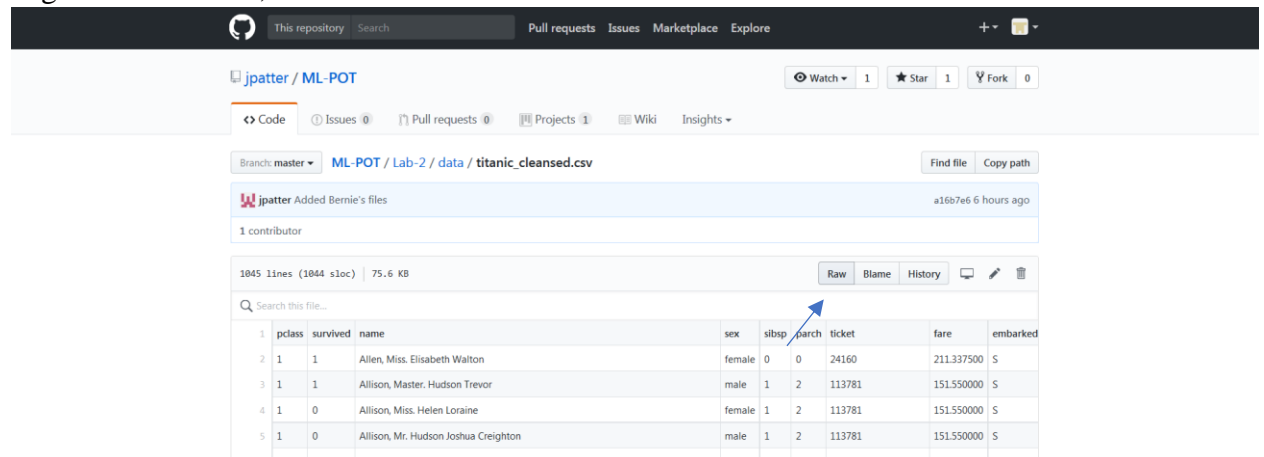
Step 2: Adding a Data Asset to the DSX Labs project

1. Download the Titanic data file from

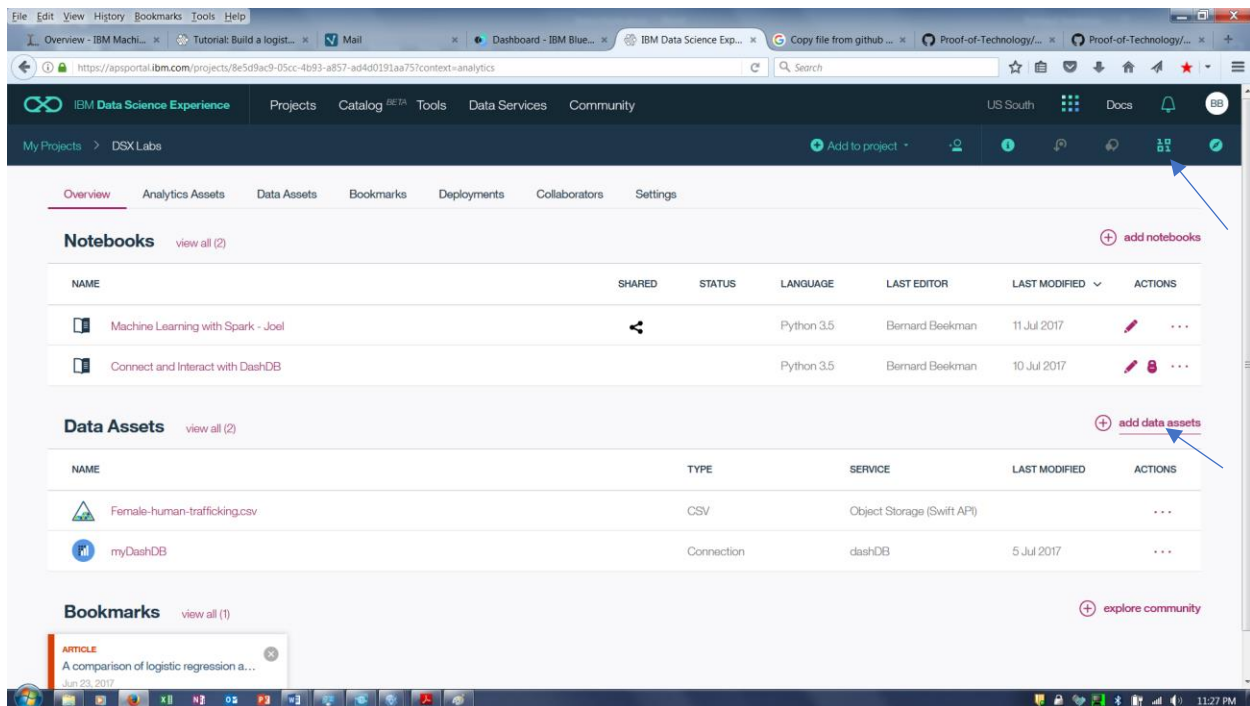
https://github.com/jpatter/ML-POT/blob/master/Lab-2/data/titanic_cleansed.csv

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



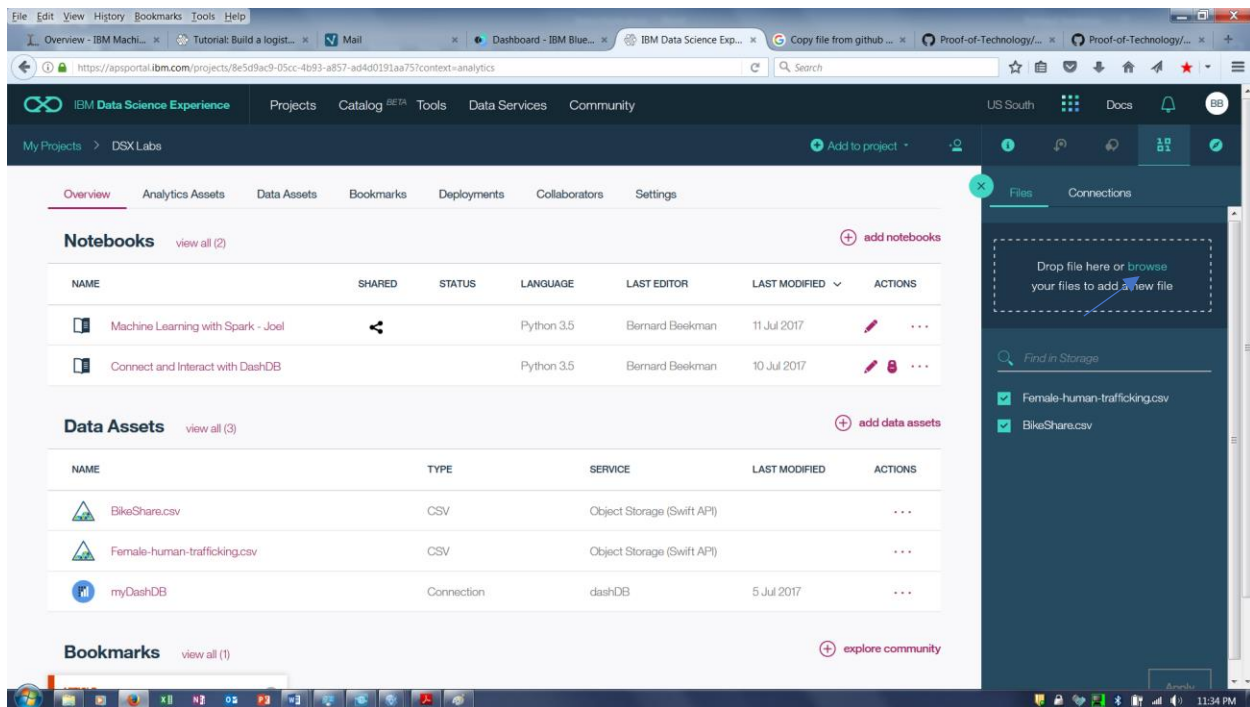
3. Go back to the DSX-Labs project. Click on **add data assets** or the  icon.



The screenshot shows the IBM Data Science Experience (DSX) interface for the 'DSX-Labs' project. The 'Data Assets' tab is selected, and the 'add data assets' button is highlighted with a blue arrow. The interface includes a top navigation bar with 'Projects', 'Catalog', 'Tools', 'Data Services', and 'Community'. The main content area displays a table of data assets and a sidebar with 'Notebooks' and 'Bookmarks'.

NAME	TYPE	SERVICE	LAST MODIFIED	ACTIONS
Female-human-trafficking.csv	CSV	Object Storage (Swift API)		...
myDashDB	Connection	dashDB	5 Jul 2017	...

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



The screenshot shows the IBM Data Science Experience (DSX) interface for the 'DSX-Labs' project. The 'Data Assets' tab is selected, and the 'add data assets' button is highlighted with a blue arrow. A file browser dialog is open, showing the 'titanic_cleansed.csv' file selected. The interface includes a top navigation bar with 'Projects', 'Catalog', 'Tools', 'Data Services', and 'Community'. The main content area displays a table of data assets and a sidebar with 'Notebooks' and 'Bookmarks'.

NAME	TYPE	SERVICE	LAST MODIFIED	ACTIONS
BikeShare.csv	CSV	Object Storage (Swift API)		...
Female-human-trafficking.csv	CSV	Object Storage (Swift API)		...
myDashDB	Connection	dashDB	5 Jul 2017	...

Step 3: Create a Model to predict survival

1. Click on the Analytics Assets Tab

The screenshot shows the IBM Data Science Experience (DSX) interface. The 'Analytics Assets' tab is selected, indicated by a blue arrow. The interface displays three sections: Notebooks, Data Assets, and Bookmarks. The Notebooks section lists two notebooks: 'Machine Learning with Spark - Joel' and 'Connect and Interact with DashDB'. The Data Assets section lists three assets: 'BikeShare.csv', 'Female-human-trafficking.csv', and 'myDashDB'. The Bookmarks section shows one bookmark: 'myDashDB'. A sidebar on the right shows a file upload area with the text 'Drop file here or browse your files to add a new file' and a list of files: 'Female-human-trafficking.csv' and 'BikeShare.csv'.

NAME	SHARED	STATUS	LANGUAGE	LAST EDITOR	LAST MODIFIED	ACTIONS
Machine Learning with Spark - Joel			Python 3.5	Bernard Beekman	11 Jul 2017	
Connect and Interact with DashDB			Python 3.5	Bernard Beekman	10 Jul 2017	

NAME	TYPE	SERVICE	LAST MODIFIED	ACTIONS
BikeShare.csv	CSV	Object Storage (Swift API)		
Female-human-trafficking.csv	CSV	Object Storage (Swift API)		
myDashDB	Connection	dashDB	5 Jul 2017	

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

2. Click on the add models.

The screenshot shows the IBM Data Science Experience (DSX) interface with the 'Analytics Assets' tab selected. The 'Models' section is visible, showing two models: 'Female Human Trafficking- Manual' and 'Female Human Trafficking'. A blue arrow points to the 'add models' button in the top right corner of the Models section.

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

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

Create new model BETA

Name
Titanic

Description
Machine Learning Model for the Titanic Data Set

Machine Learning Service
Watson Machine Learning

Spark Service
DSXSpark

Automatic
Prepare my data and create a model automatically

Manual
Let me prepare my data and select which models to train

Need something more flexible? Create a [notebook](#) or design a [flow](#).

Cancel Create

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

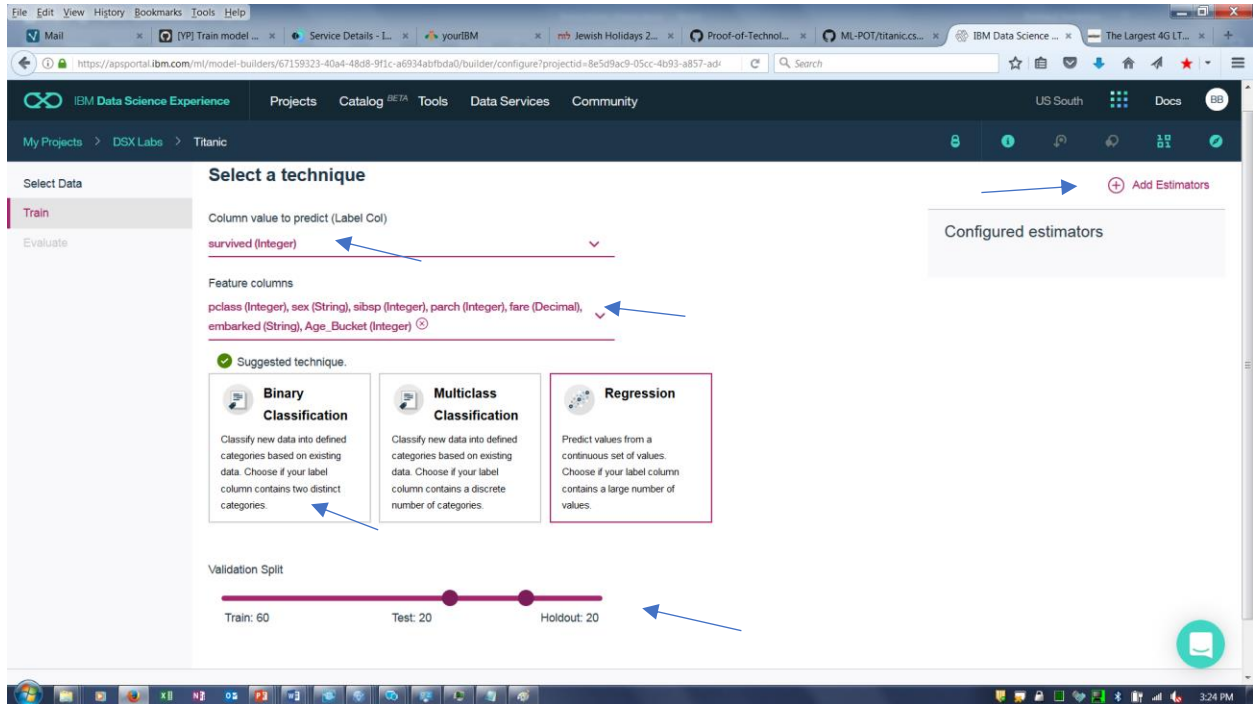
Select data asset

The model builder currently supports CSV files.

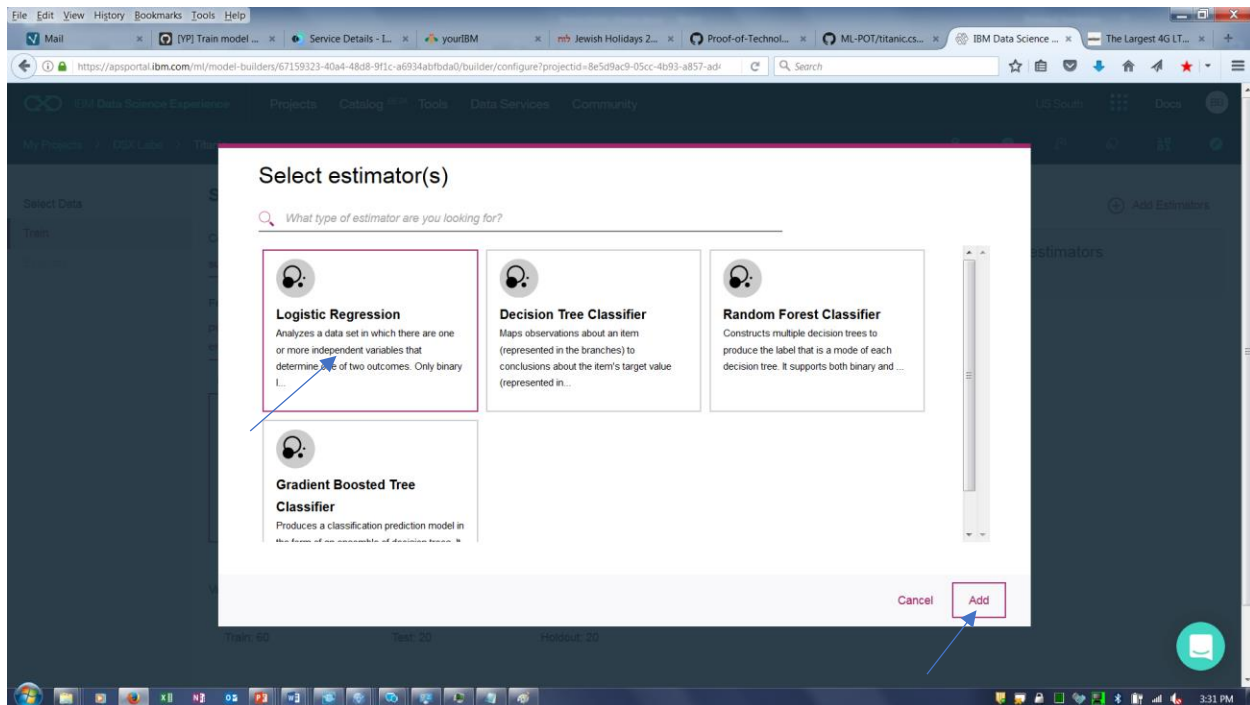
	NAME	TYPE	SERVICE
<input type="radio"/>	Female-human-trafficking.csv	CSV	Object Storage (Swift API)
<input type="radio"/>	BikeShare.csv	CSV	Object Storage (Swift API)
<input type="radio"/>	LimitedBikeShare.csv	CSV	Object Storage (Swift API)
<input type="radio"/>	titanic.csv	CSV	Object Storage (Swift API)
<input checked="" type="radio"/>	titanic_cleansed.csv	CSV	Object Storage (Swift API)
<input type="radio"/>	titanic_prepared.csv	CSV	Object Storage (Swift API)

Close Next

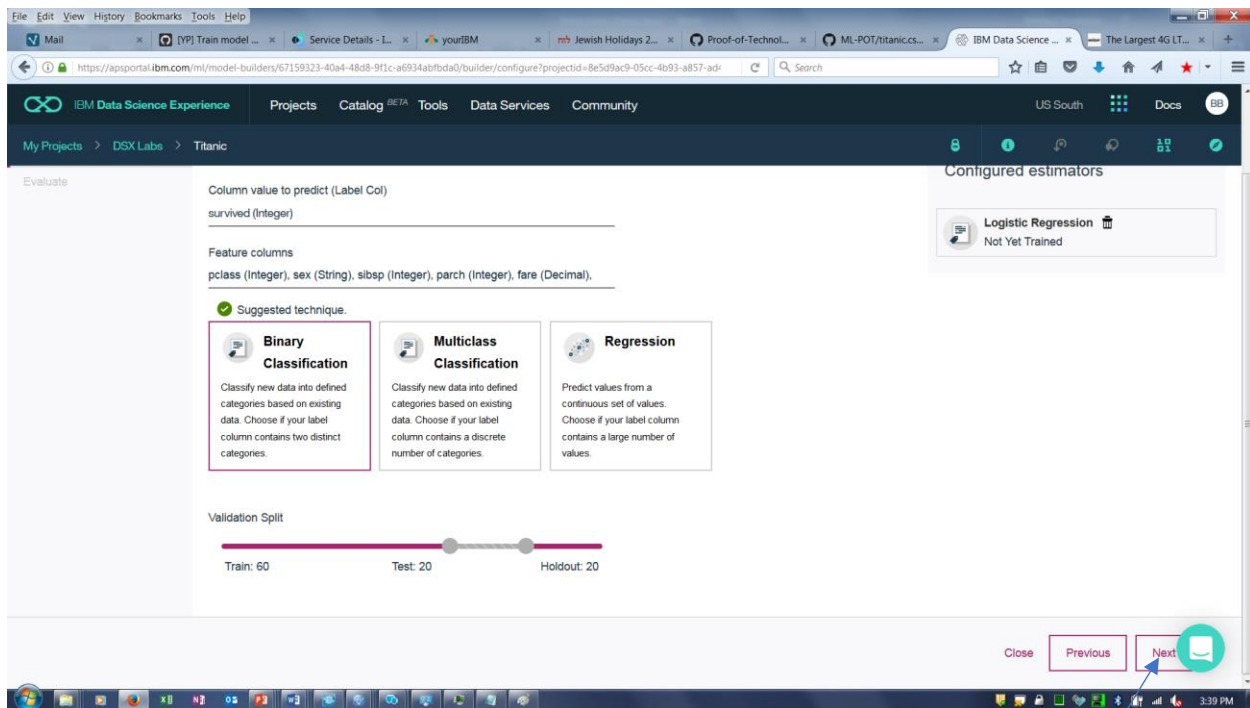
5. For **Column value to predict (Label Col)** select **survivor**. For **Feature columns** select the following features (**pclass,sex,sibsp,parch,fare,embarked,Age_Bucket**) . Click on the **Binary Classification** Box (which is suggested by the service). Adjust the **Validation Split** as desired.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

IBM Data Science Experience

Projects Catalog ^{BETA} Tools Data Services Community

US South Docs 68

My Projects > DSX Labs > Titanic

Select Data

Train

Evaluate

Select model

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	...

Close Previous Save

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

My Projects > DSX Labs > Titanic

Details Predictions

Titanic

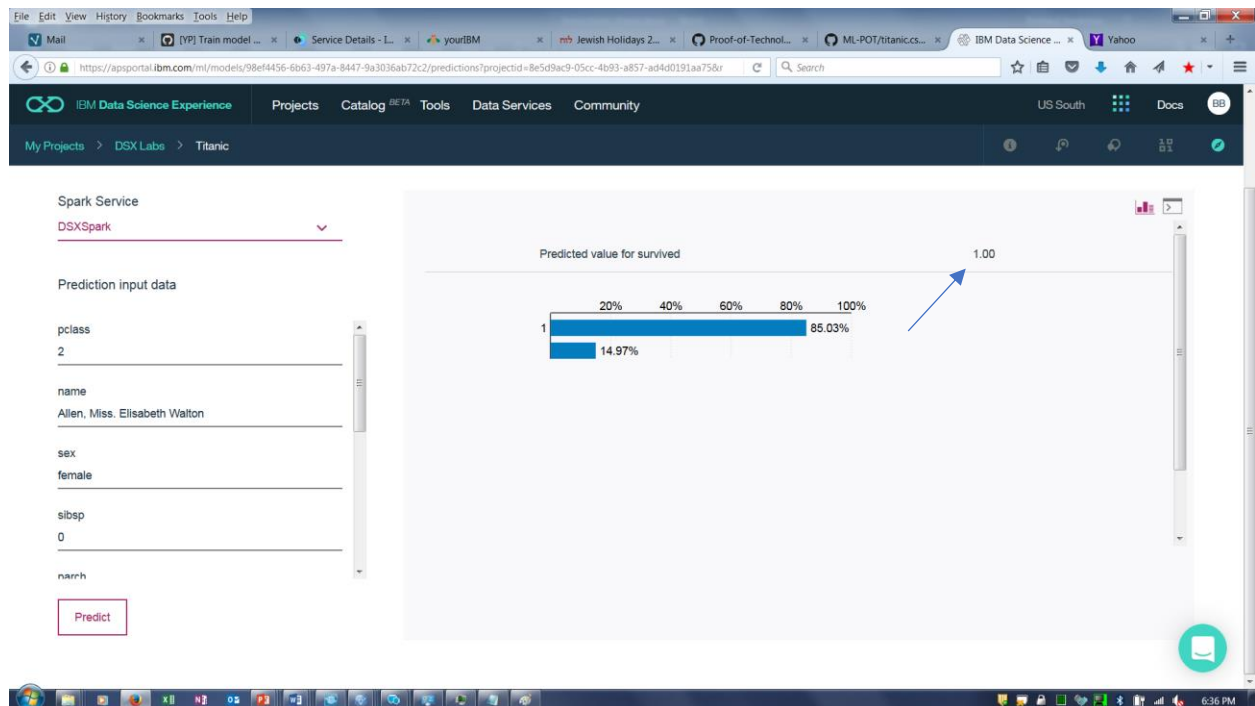
Machine learning service	Machine Learning-s2
Label column	survived
Model builder details	View
Training data schema	View
Input data schema	View
Runtime environment	spark-2.0
Training date	26 Aug 2017, 6:30 PM

Deployments

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



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**

IBM Data Science Experience

Projects Catalog BETA Tools Data Services Community

US South Docs 98

My Projects > DSX Labs > Titanic

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

Deployments

[+ Add Deployment](#)

NAME	DEPLOYMENT TYPE	ACTIONS
Your model is not deployed.		

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

File Edit View History Bookmarks Tools Help

Mail [VP] Train model ... Service Details - L... yourIBM Jewish Holidays 2... Proof-of-Techno... ML-POT/titanic.cs... IBM Data Science ... Yahoo

https://appsportal.ibm.com/ml/models/98ef4456-6b63-497a-8447-9a3036ab72c2/details?projectId=8e5df9ac9-95cc-4b93-a857-ad4d0191aa75&mlins

IBM Data Science Experience Projects Catalog BETA Tools Data Services Community

US South Docs 98

My Projects > DSX Labs > Titanic

Label column

Model builder details

Training data schema

Input data schema

Runtime environment

Training data

Deployments

[+ Add Deployment](#)

NAME

DEPLOYMENT TYPE

ACTIONS

Your model is not deployed.

Deploy model

Deployment Type

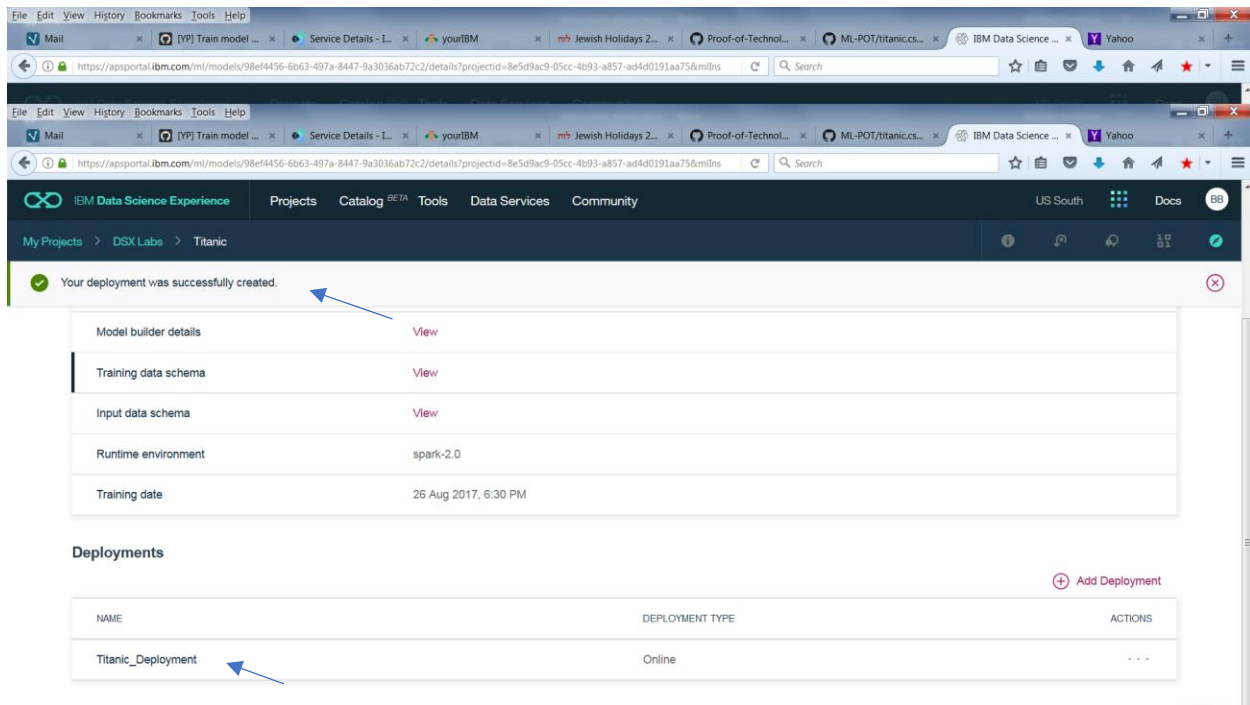
Online

Name

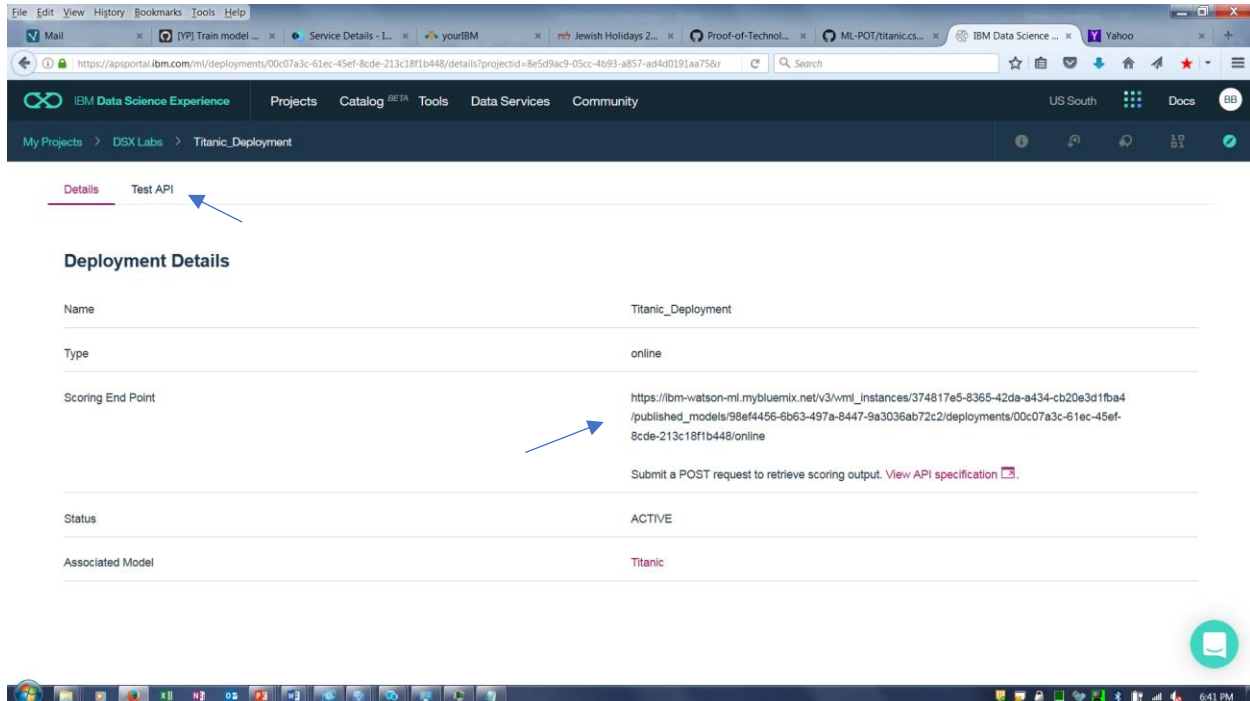
Titanic_Deployment

Close Deploy

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



- 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.



- 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

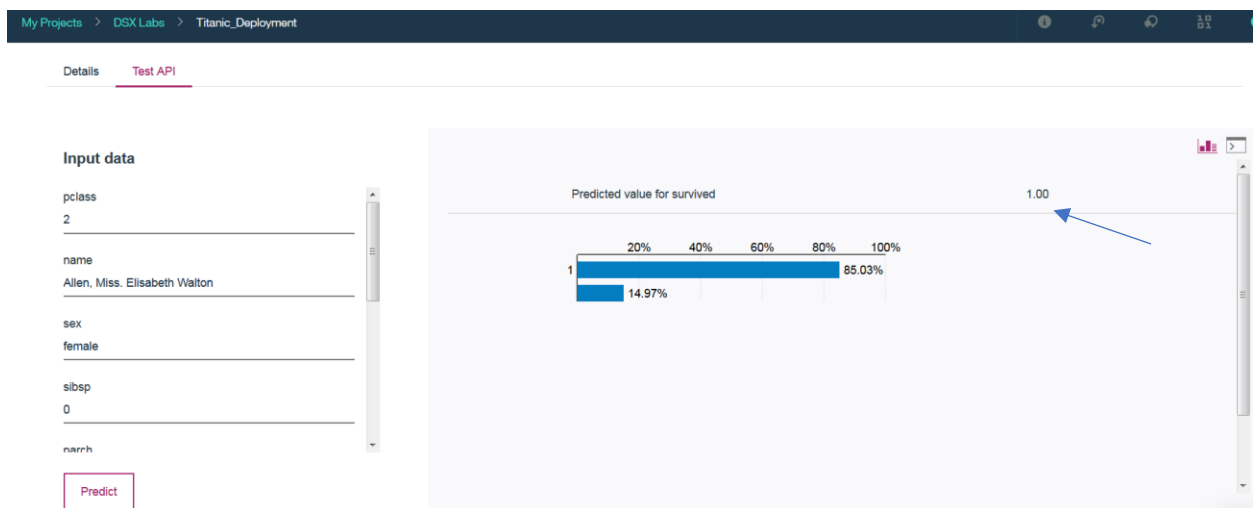
sibsp
0

narch

Predict



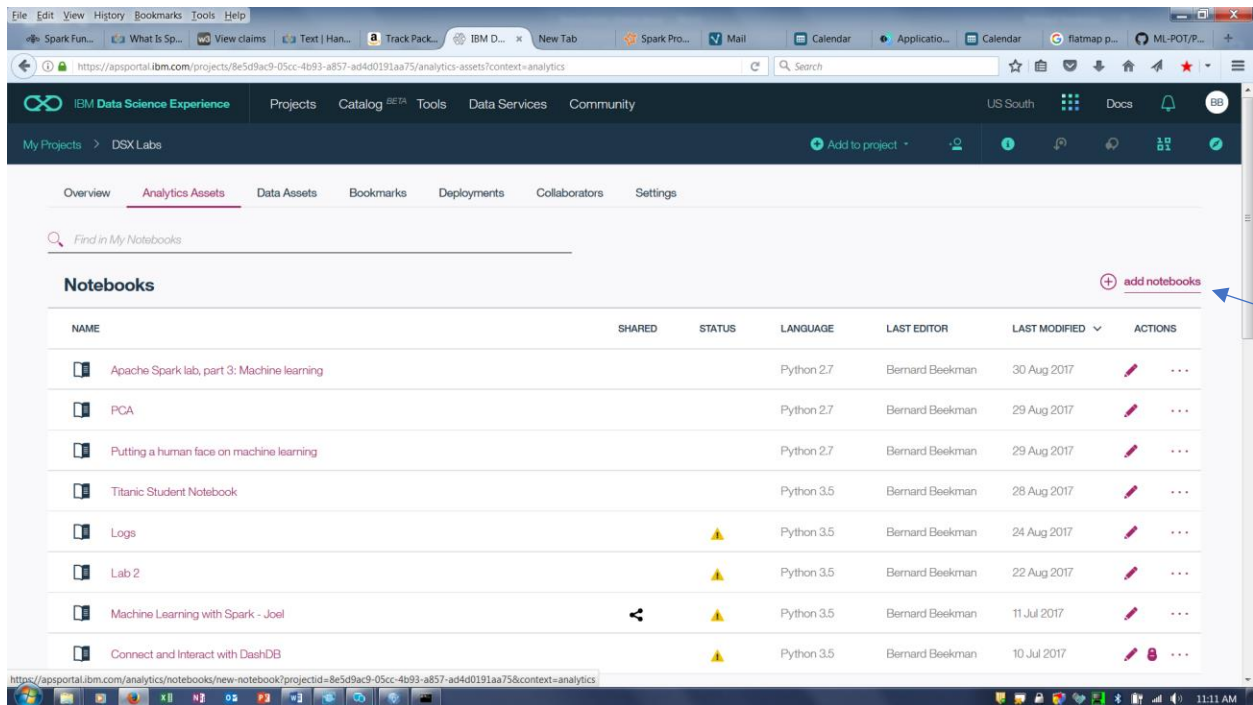
7. The predicted result is returned.



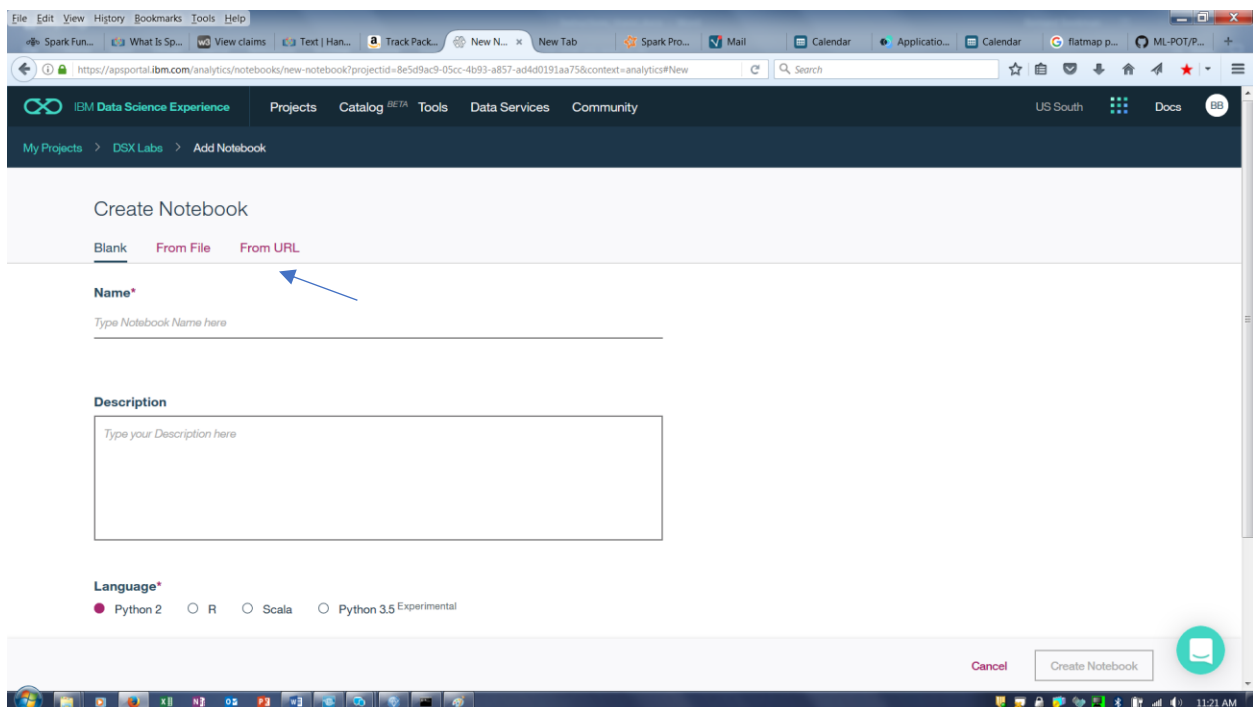
Step 5: Create a simple web front-end

This section will provide an example of creating a simple Python Flask front-end to invoke the Titanic scoring API. The steps will be described in a notebook on the github repository.

1. Go to the DSX Labs Project and click on **add notebook**.



2. Click on **From URL**.



3. Enter a **Name** for the notebook and optionally a **Description**. Cut and paste the url below into the Notebook URL field, select a Spark service, and then click on **Create Notebook**.

<https://github.com/jpatter/ML-POT/blob/master/Lab-2/Titanic%2BWeb%2BFront-End.ipynb>

IBM Data Science Experience | Projects | Catalog BETA | Tools | Data Services | Community | US South | Docs |

My Projects > DSX Labs > Add Notebook

Create Notebook

Blank | **From File** | **From URL**

Name*

Titanic Web Front-End 29 Characters Remaining

Description

Type your Description here

Notebook URL*

<https://github.com/jpatter/ML-POT/blob/master/Lab-2/Titanic%2BWeb%2BFront-End.ipynb>

Spark Service*

DSXSpark

Associate this notebook with the Spark Service of your choice.

Cancel Create Notebook

4. Follow the instructions in the notebook.

