IBM Cloud and IBM Data Science Experience

Using IBM Cloud and IBM Data Science Experience, to Create and deploy a scoring model to predict heart failure

Lab Exercise

© Copyright IBM Corporation 2017

IBM, the IBM logo and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal /copytrade.shtml.

This document is current as of the initial date of publication and may be changed by IBM at any time.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

Table of Contents

Overview

The objective of this lab is to demonstrate how to use IBM Data Science Experience to build a predictive model within a Jupyter Notebook.

You will deploy this predictive model to the Watson Machine Learning Service in IBM Cloud. Based on the data provided by User, User will get prediction about his heart failure. This task will be accomplished without writing any programming code.

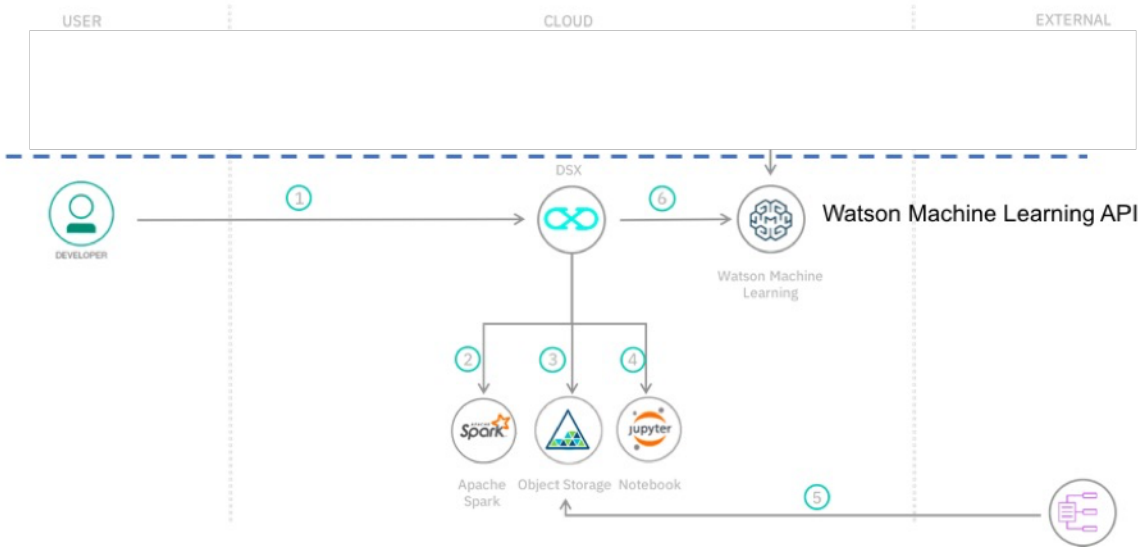
Some of the tasks that you will perform in this lab include:

- The developer creates an IBM Data Science Experience Workspace.
- IBM Data Science Experience depends on an Apache Spark service.
- IBM Data Science Experience uses Cloud Object storage to manage your data.
- This lab is built around a Jupyter Notebook, this is where the developer will import data, train, and evaluate their model.
- Import data on heart failure.
- Trained models are deployed into production using IBM's Watson Machine Learning Service.
- Accessing Watson ML Models and Deployments through API

Introduction

This lab is designed to demonstrate how to use IBM Data Science Experience to build a predictive model within a Jupyter Notebook. The predictive model is then deployed to the Watson Machine Learning Service in IBM Cloud

Flow



Pre-requisites

- IBM Cloud supported web browser
- An IBM Cloud Account

Duration

You should be able to complete this lab in approximately 60 minutes.

Using IBM Cloud and IBM Data Science Experience, to Create and deploy a scoring model to predict heart failure

Getting Started

To be able do this lab IBM Cloud account is necessary.

Not registered

If you don't have one yet -- or you did not complete the initial set up of your IBM Cloud account. Please follow instructions as per note sent to you while Registration for this LAB.

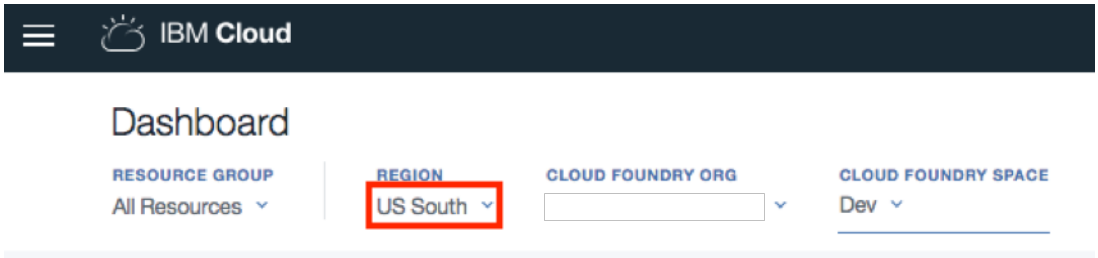
Your account must have enough resources available for at least 1 application (128MB) and 4 services.

Already registered and completed set-up

When you already registered, and completed the initial set-up of your IBM Cloud account, you directly jump to Create a space in **IBM Cloud US South region**.

Create a space in IBM Cloud US South Region

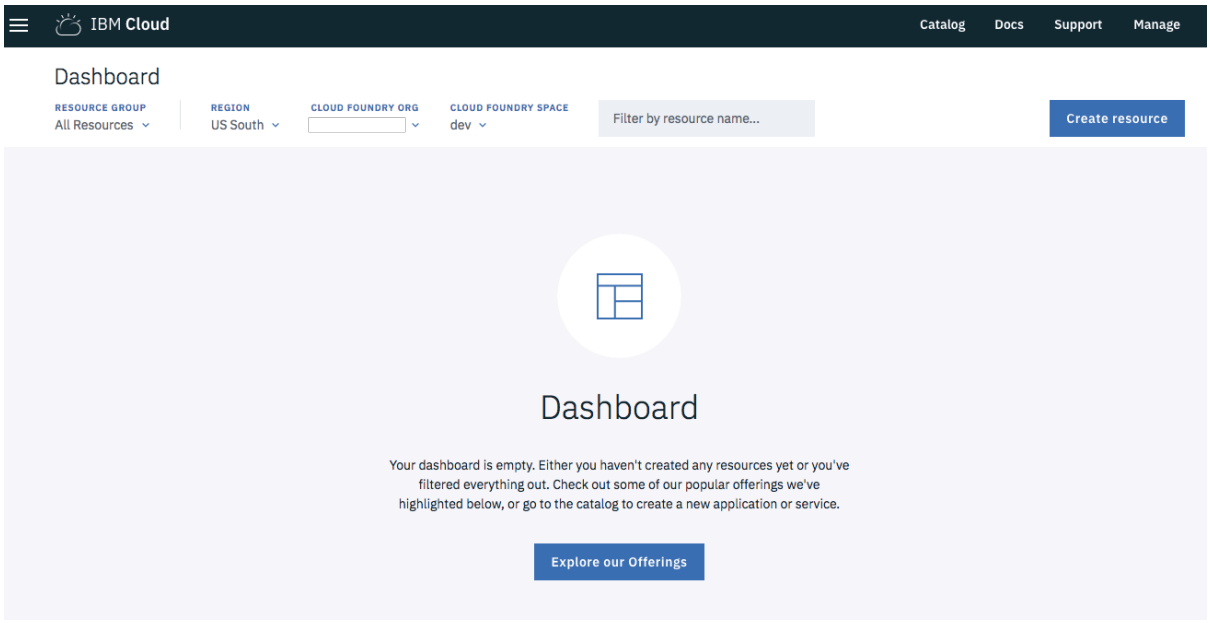
For the remainder of this lab we switch to the **US South Region** of IBM Cloud. For this, use Ctrl-click (or the equivalent for your system) to open the IBM Cloud dashboard. Click your account and choose **US South** as your active region.



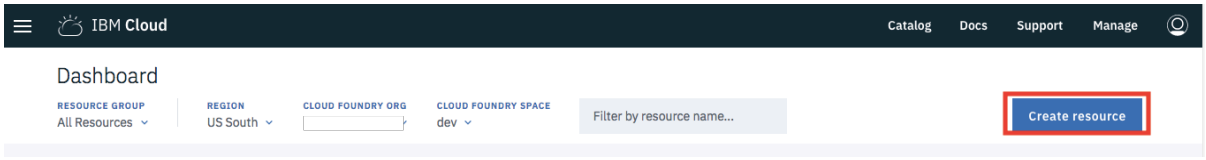
If you are all OK, you get the dashboard. Otherwise, you will be asked to create your first space in this region -- as depicted in the screenshot below. Typically, **Dev** would be a good name for your space.

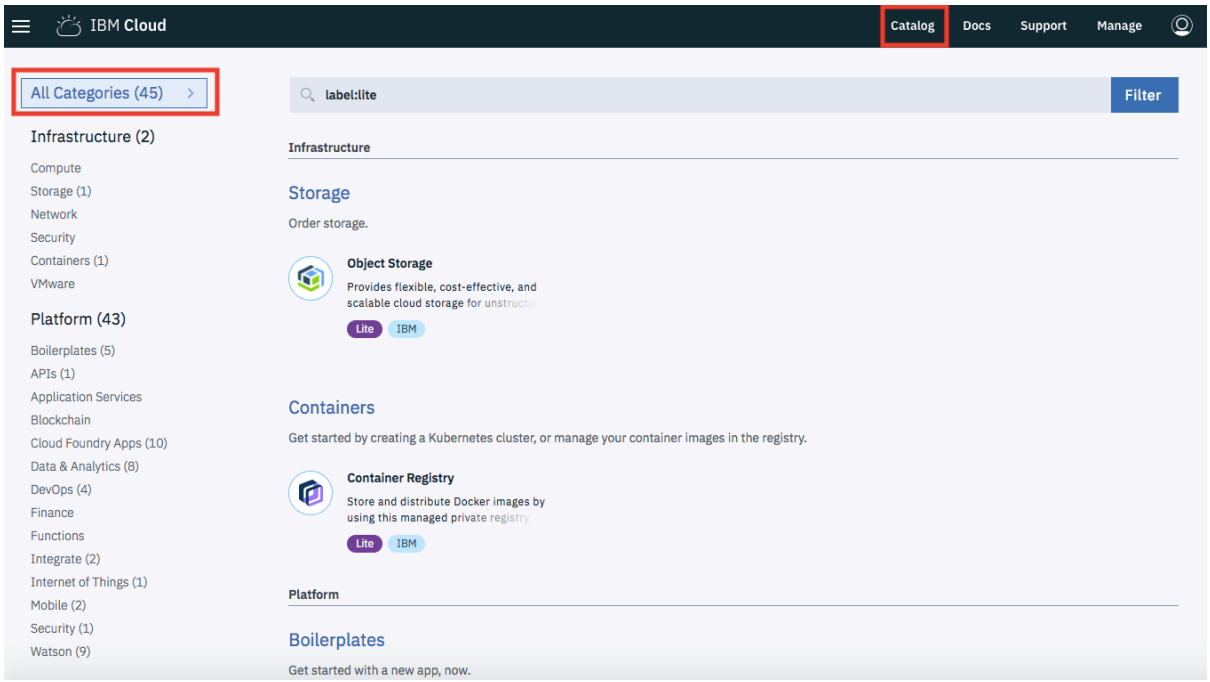
Step 1: Create IBM Data Science Experience Service

Once, you have logged in your IBM Cloud account, through http://console.bluemix.net , you will be landed on Dashboard.

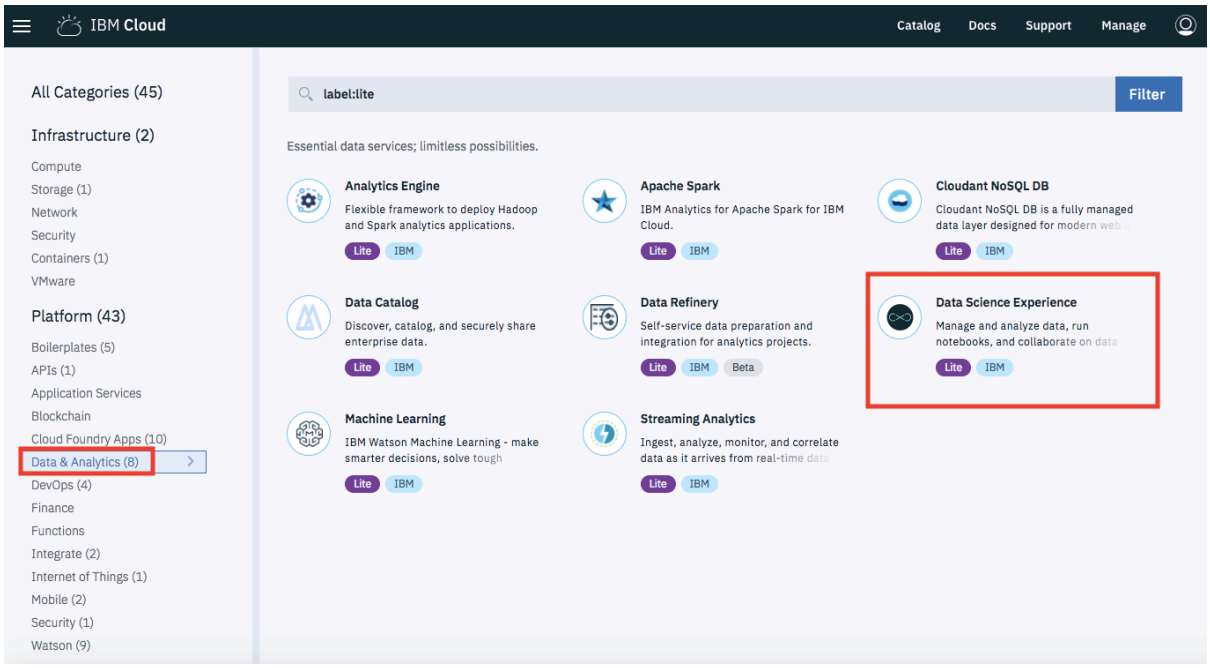


As you are logging for the first time, you will not have any applications on your dashboard. Click on "**Create resource"** link. You will be taken to a "*Catalog*" page



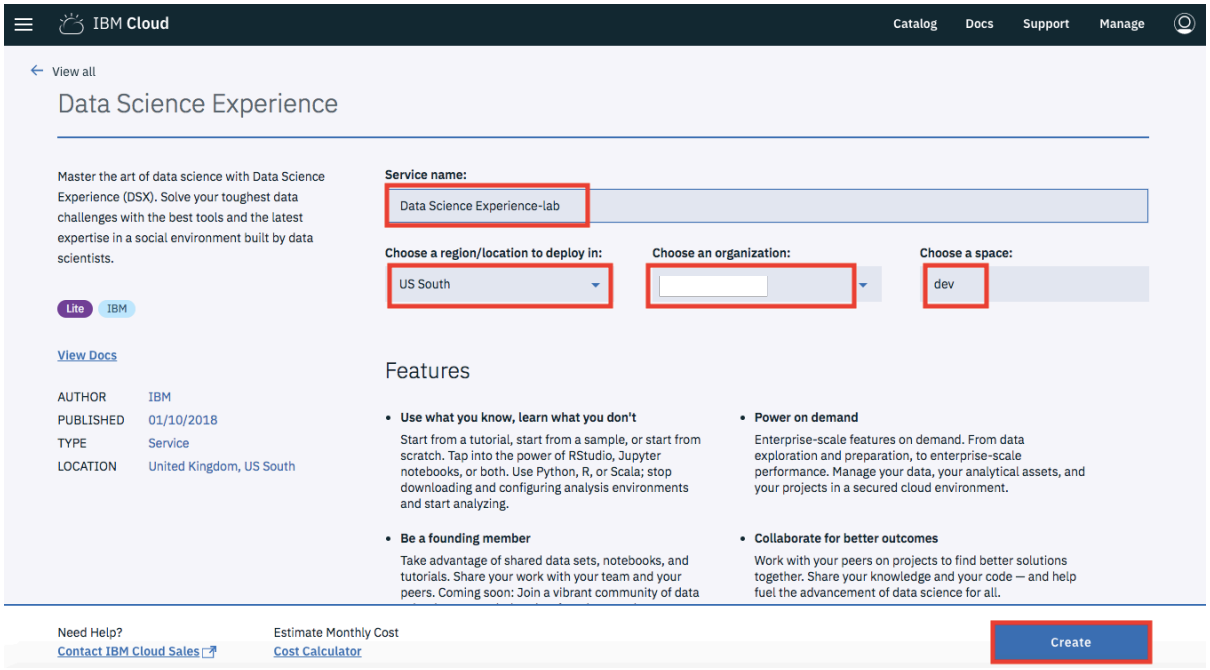


Select “Data & Analytics” Category from the left-hand side menu. Choose Data Science Experience

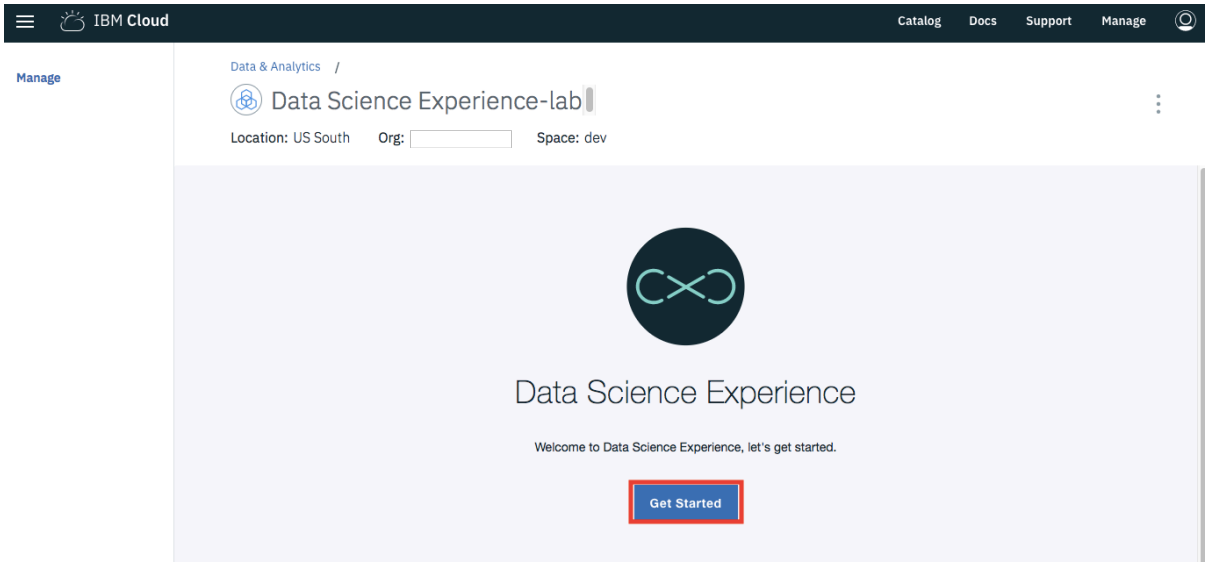


Once you click on Data Science Experience, you will be asked to give details on Service name and choose appropriate organization and space. After providing above details click on “Create” button.

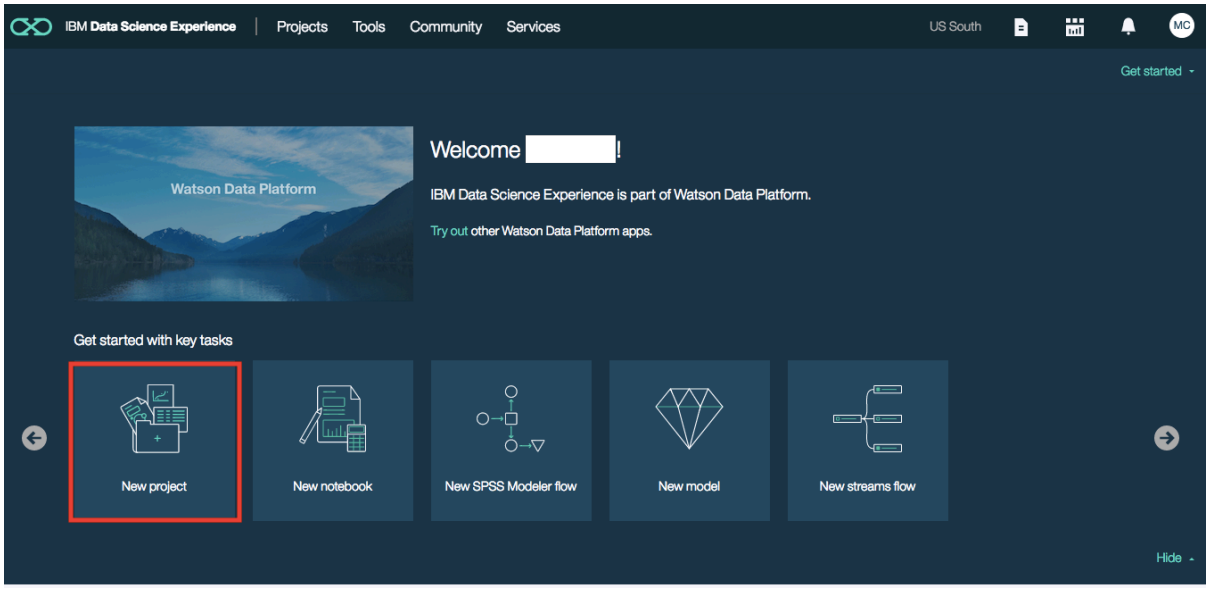
*Note: Please ensure you are using **US South** as region for all services being created for this lab*



On successful creation of service, you will get to “Data Science Experience” Console launch page.



As soon as the 'Get Started' button is clickable, click it and you should be directed to the Data Science Experience dashboard as shown below.

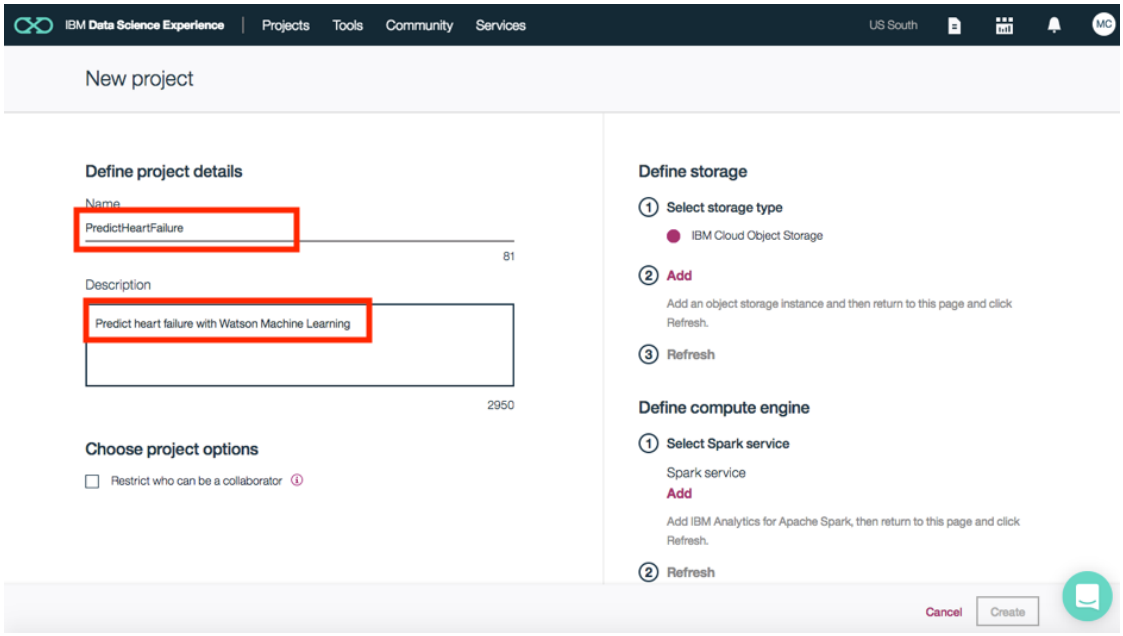


Note that two IBM Cloud services will be created for you -- a Cloud Object Storage service and an Apache Spark service. (In case those are not created, Step 2 will cover creation of those services)

Step 2: Create IBM Data Science Experience Service resources: Project, Services (Apache Spark and Cloud Object Storage)

In this part of the lab you will create a new project in IBM Data Science Experience and bind it to your instance of the Watson Machine Learning service.

1. From the dashboard, click on **"New project**" from the Get started with key tasks. Enter *"PredictHeartFailure"* as the project name and description Leave the other settings on their default value.

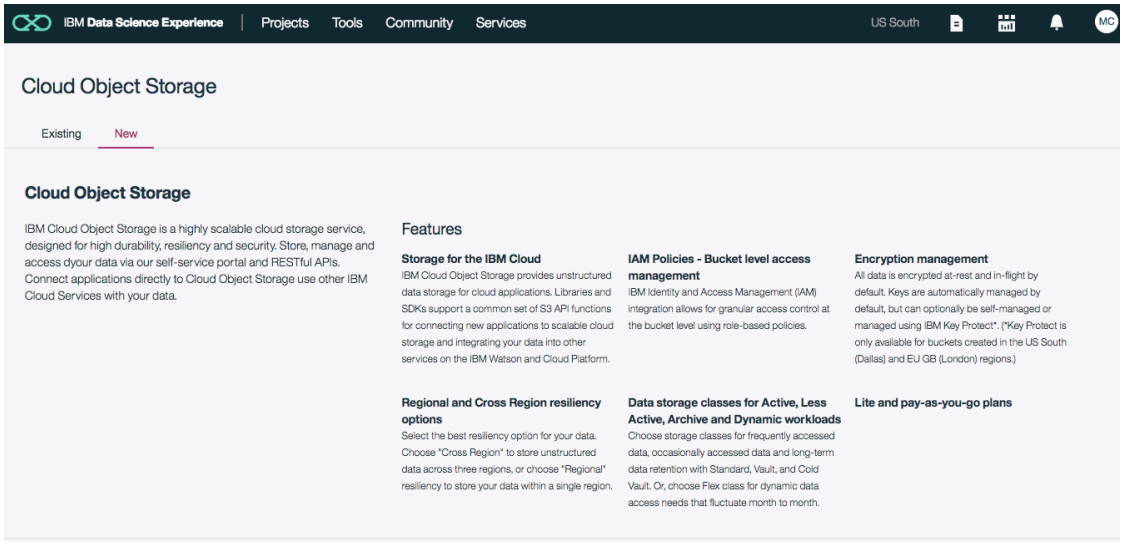


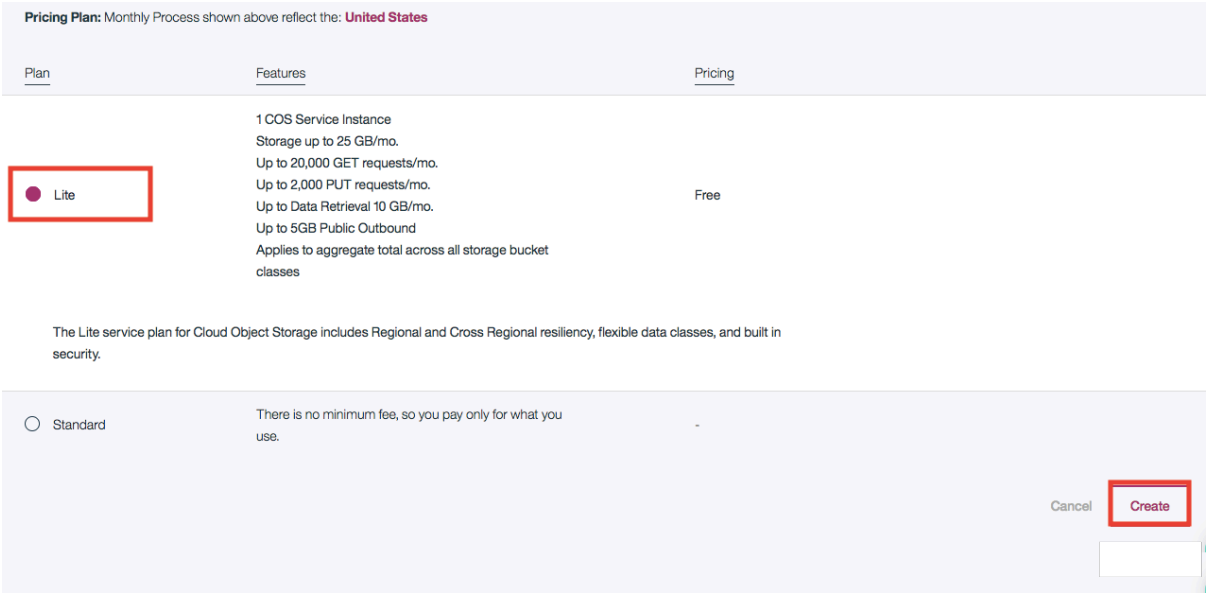
IBM Cloud services will be required

a) Cloud Object Storage service and
b) an Apache Spark service

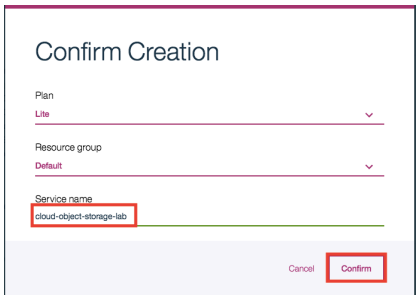
Let's start creating these services

a) Cloud Object Storage Service : Select Lite (Free) Plan for this Lab and Click on "Create"



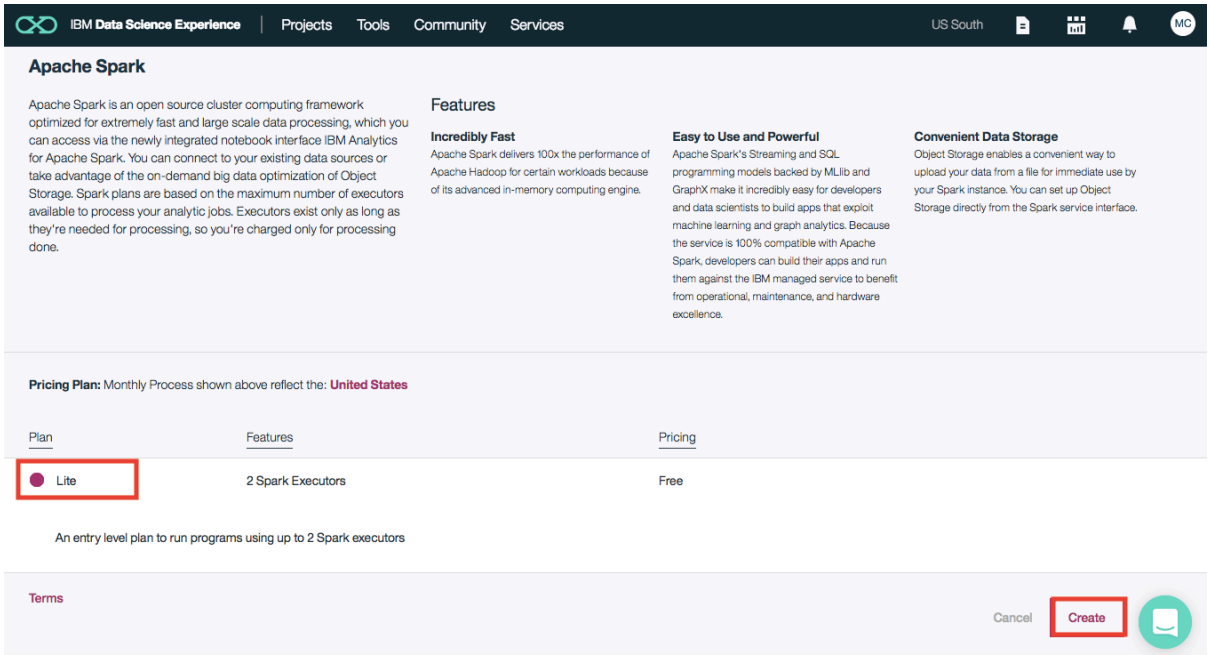


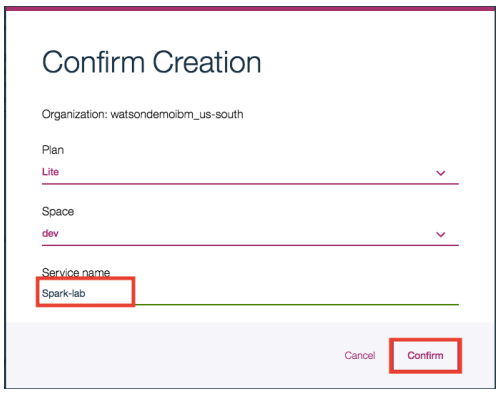
Click on “Create”



On Confirmation “cloud-object-storage-lab” will be created.

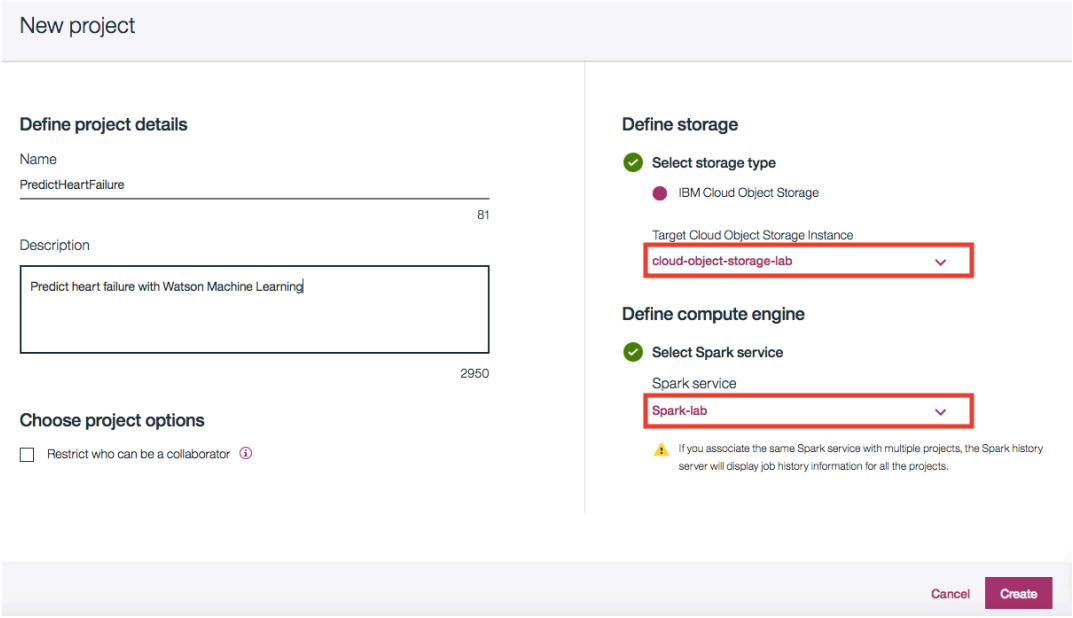
b) Apache Spark Service



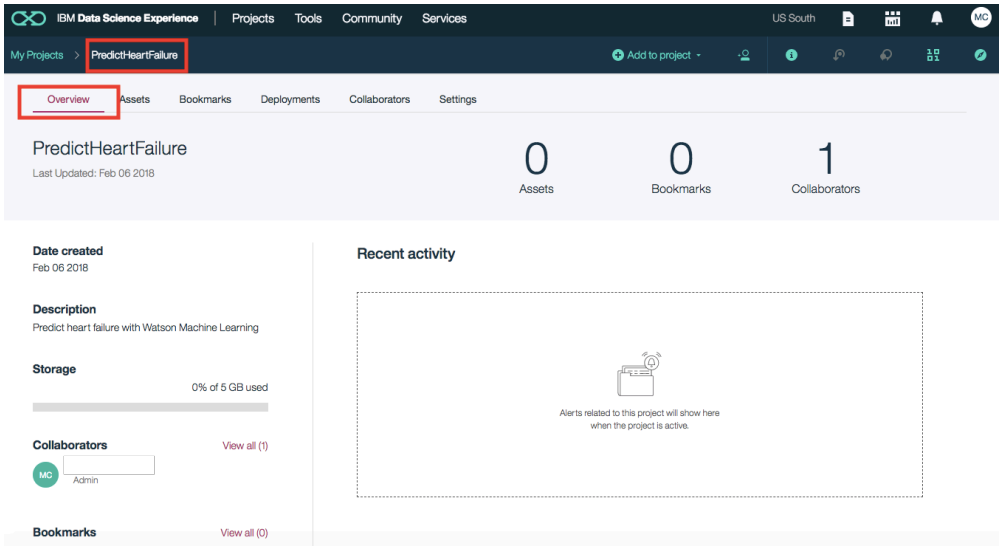


On Confirmation “spark-lab” will be created.

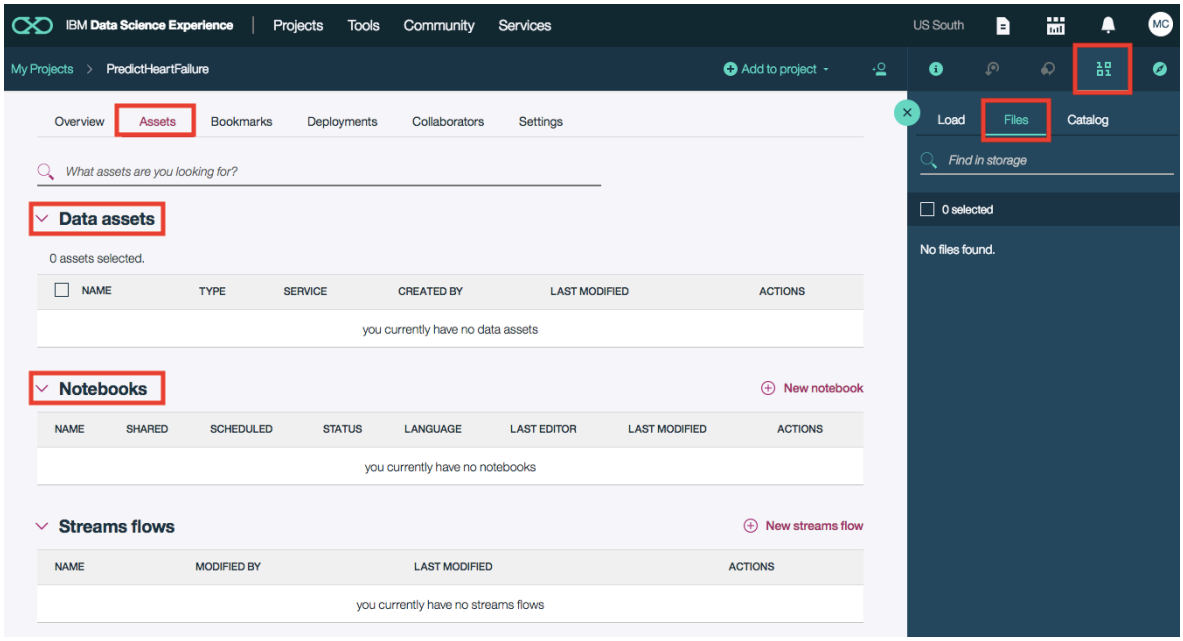
Go back to “View All Projects” and Select “PredictHeartFailure” Project, menu screen and Click on “Create” button



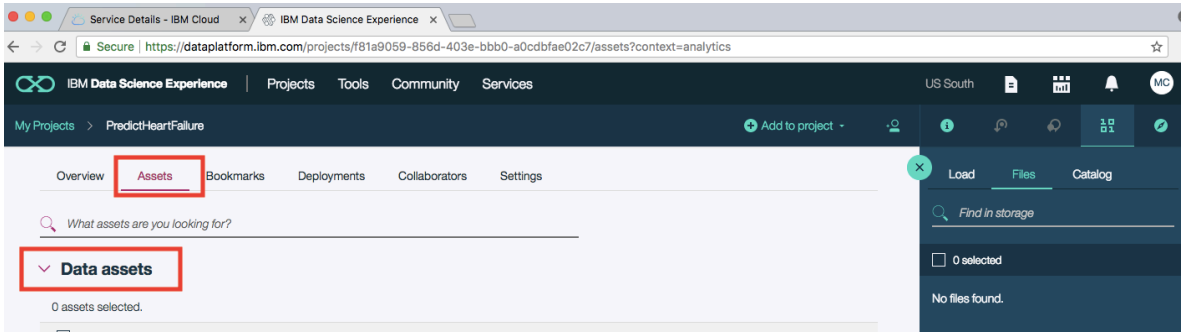
On successful creation of project, you will be on “Overview” page of created Project.

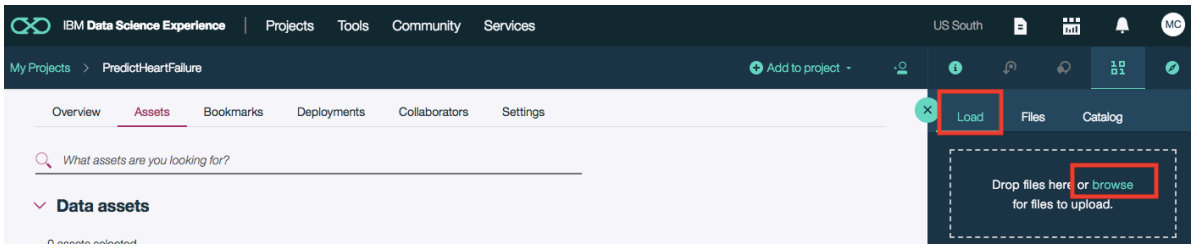


Click on “Assets” menu, all assets related to your project will be displayed here, like data assets, notebooks, etc.

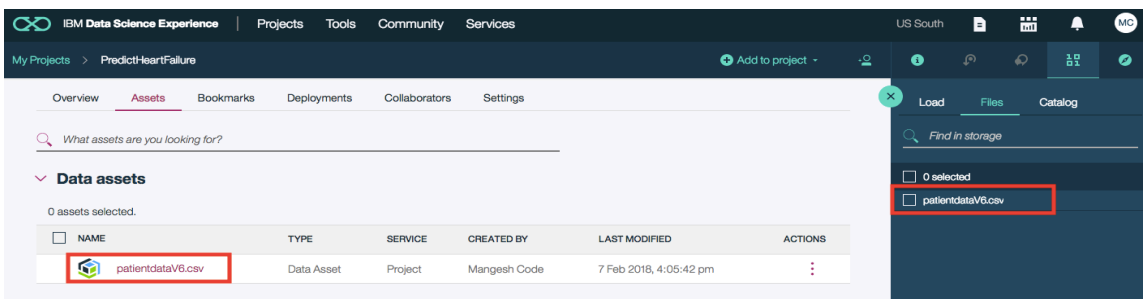


On your laptop, browse to the location where you downloaded the file **patientdataV6.csv** in the section Download patient data of this lab. Select the file and click on Open (or the equivalent action for your operating system).





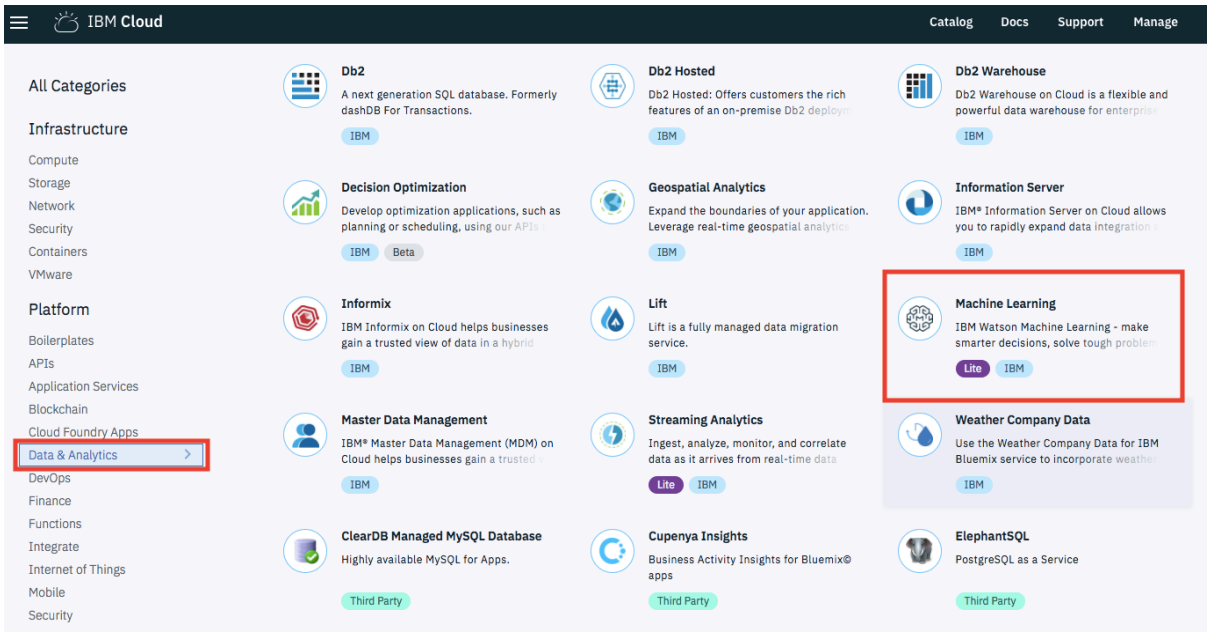
6. Once successfully uploaded, the file should appear in the **Data Assets** section.



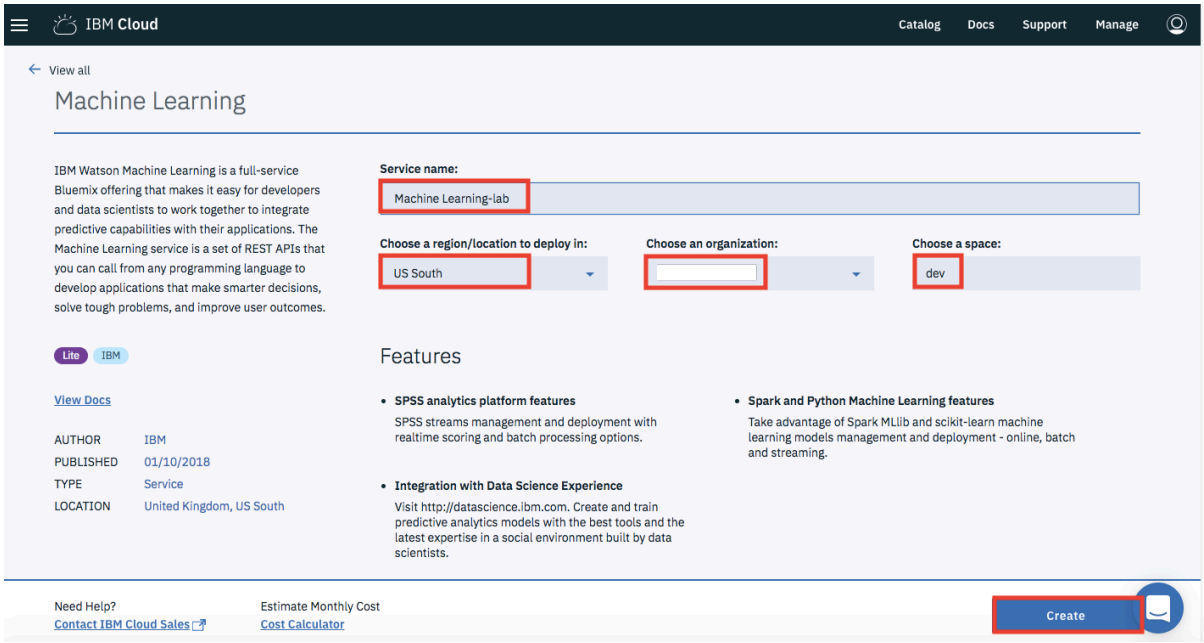
Step 3: Create an instance of the Watson Machine Learning Service

In this part of the lab, you'll create an instance of the Watson Machine Learning service In your browser go to the IBM Cloud Dashboard and click **Catalog**.

1. In the navigation menu at the left, select **Data & Analytics** (under **Platform**) and then select **Machine Learning**.



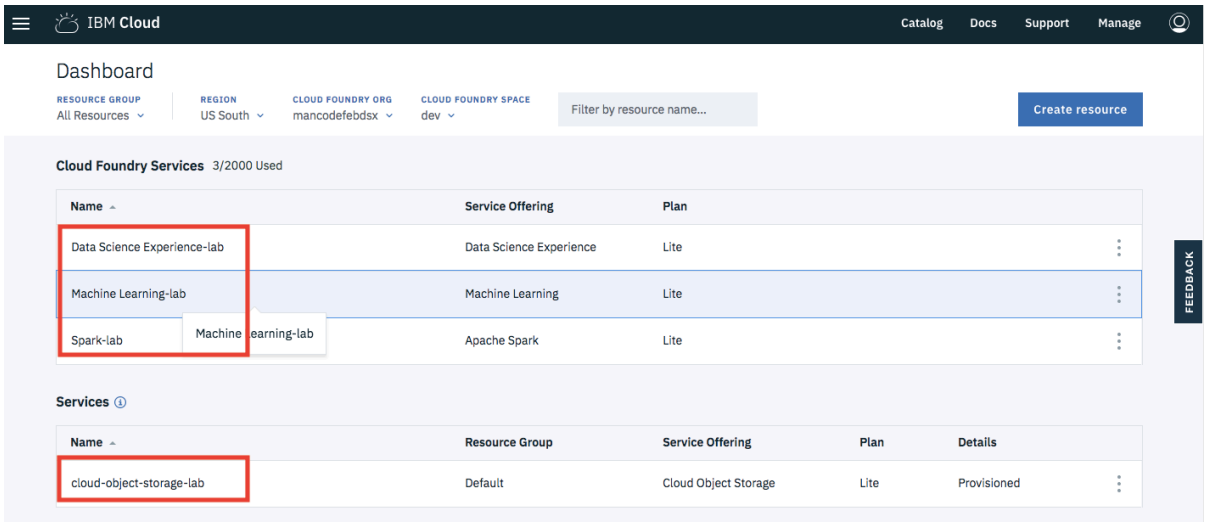
3. In the **Choose a region/location** drop-down, select the "**US South**" where deployed earlier Data Science Service.



4. Verify this service is being created in the same space as the app in Step 2.

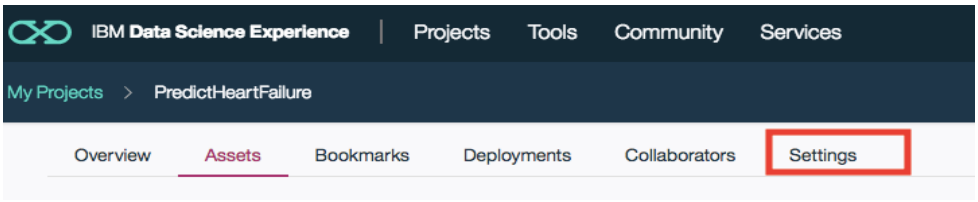
Verify :

5. On Successful creation of Step 1, Step 2 and Step3, your IBM Cloud Dashboard should have following services created.

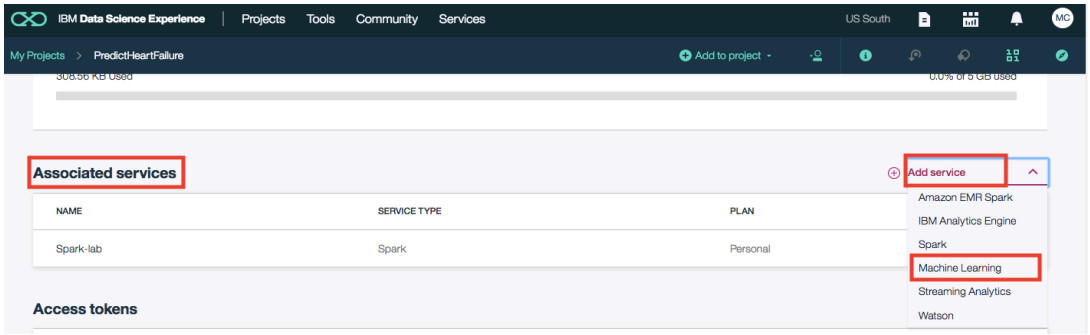


Step 4: Bind Data Science Experience Service to your Watson Machine Learning service instance

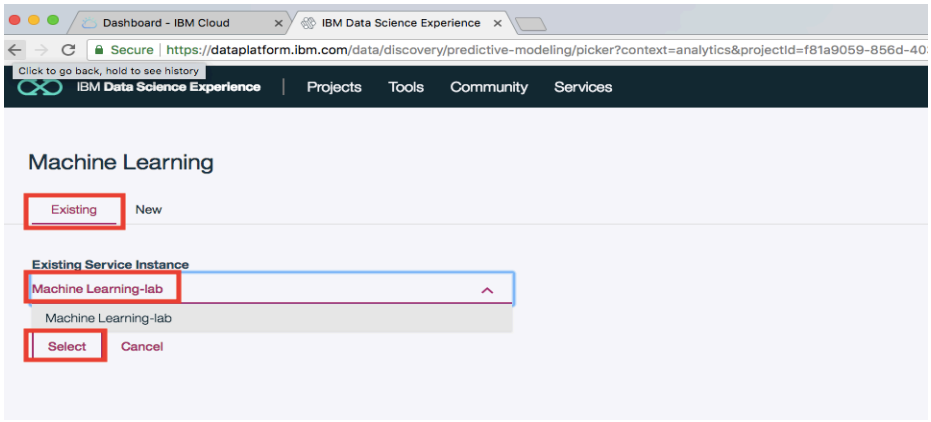
Go Back to IBM Data Science Exercise Home page. Go to “PredictHeartFailure” Project. Click on **Settings** for the project.



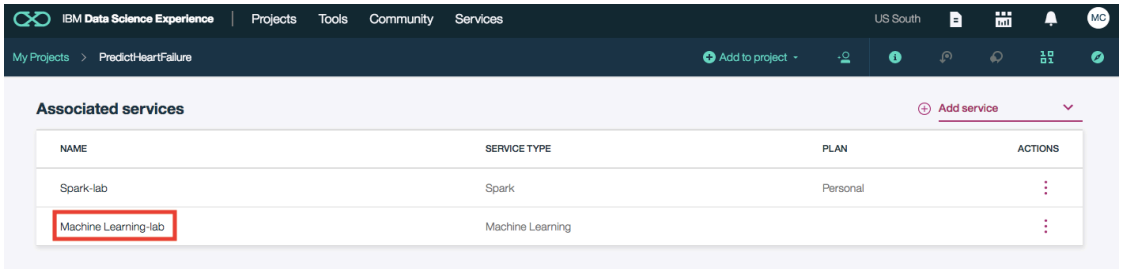
9. Click on add associated service and select **Machine Learning**



10. Choose your existing Machine Learning instance and click on **Select**.



11. Click on your browser’s Back button and verify that the Watson Machine Learning service is now listed as one of your **Associated Services**.

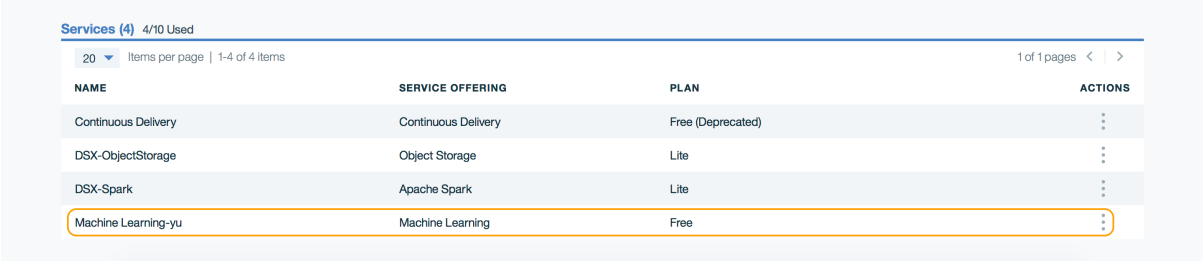


12. Leave the browser tab open for later.

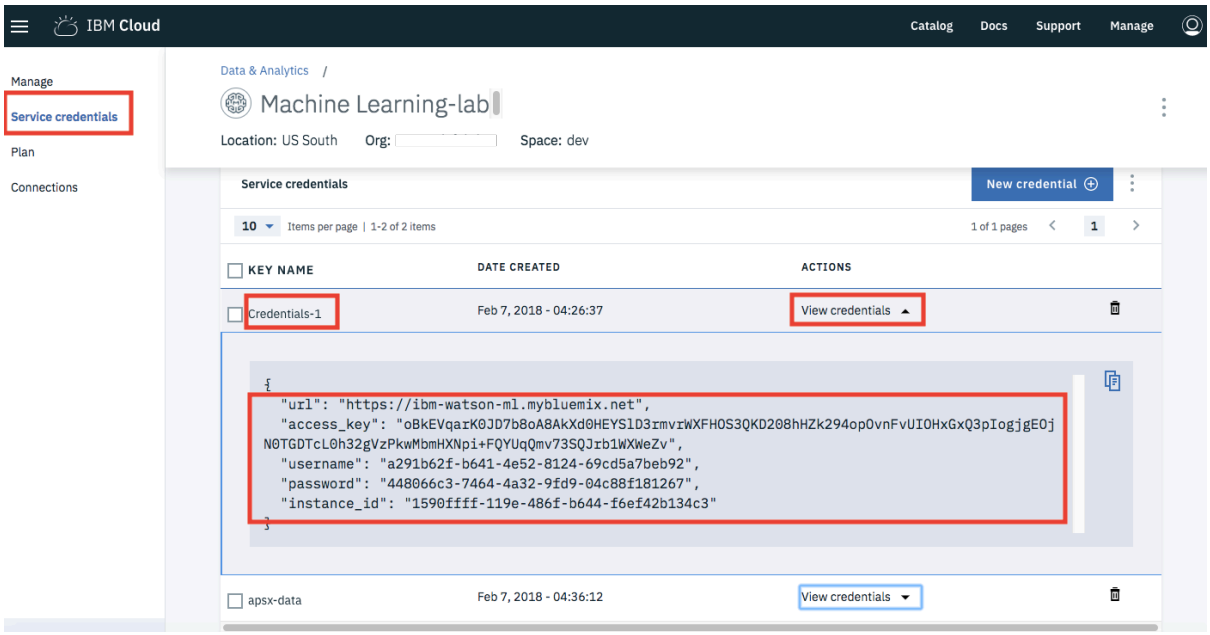
Step 5: Save the credentials for your Watson Machine Learning Service

In this part of the lab you'll save the credentials for your Watson Machine Learning instance so you can use it later in your code.

1. In a different browser tab go to http://console.bluemix.net and log in to the Dashboard
2. Click on your Watson Machine Learning instance under **Services**.



3. Click on **Service credentials** and then on **View credentials** to see the credentials.

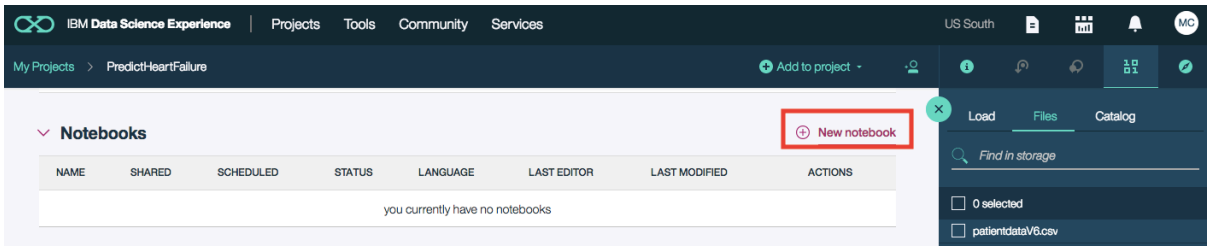


4. Save the username and password to a text file on your machine. You'll need this information later in your Jupyter notebook.

Step 6: Create a notebook in IBM Data Science Experience

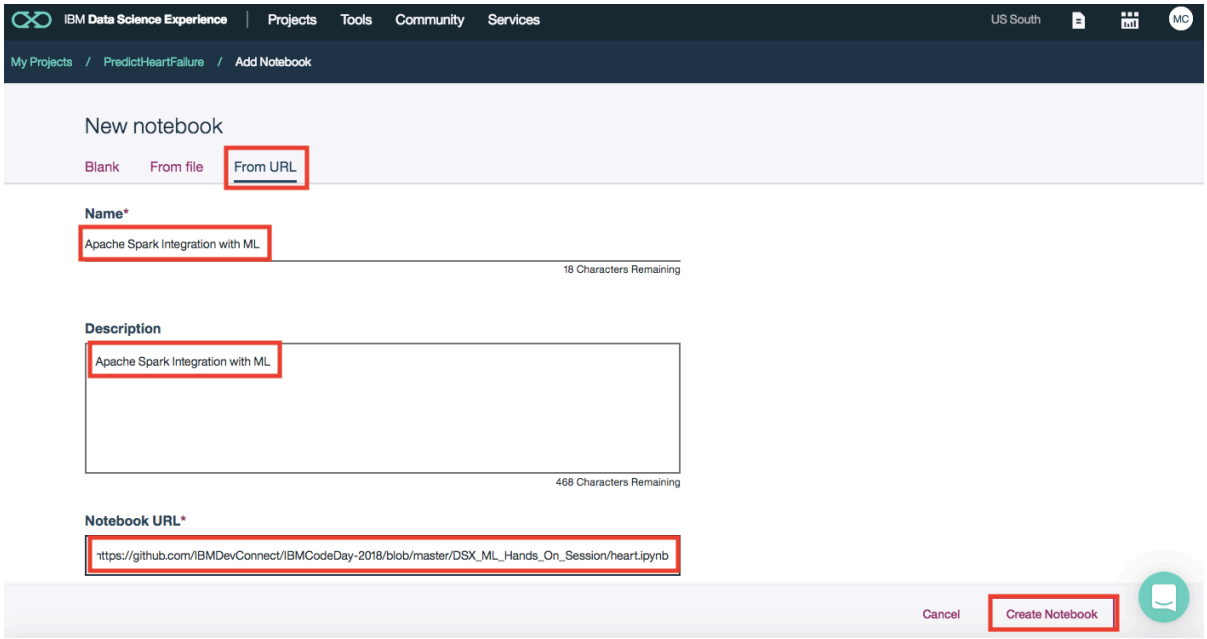
In this part of the lab you'll create a Jupyter notebook and import the code to create a predictive model.

1. Open created Project, first click **Projects -> View All Projects**, and then select your newly created project "*PredictHeartFailure*" from Step 4. Next, in the Data Science Experience browser tab click on **Overview** and then click **add notebooks**.



2. Click on **From URL** and name the notebook *Apache Spark integration with Watson ML*.

3. Under **Notebook URL** provide the following URL : Heart_Failure_Predictor.ipynb [https://github.com/IBMDevConnect/codeDay18-DSX_handson/Heart_Failure_Predictor.ipynb]



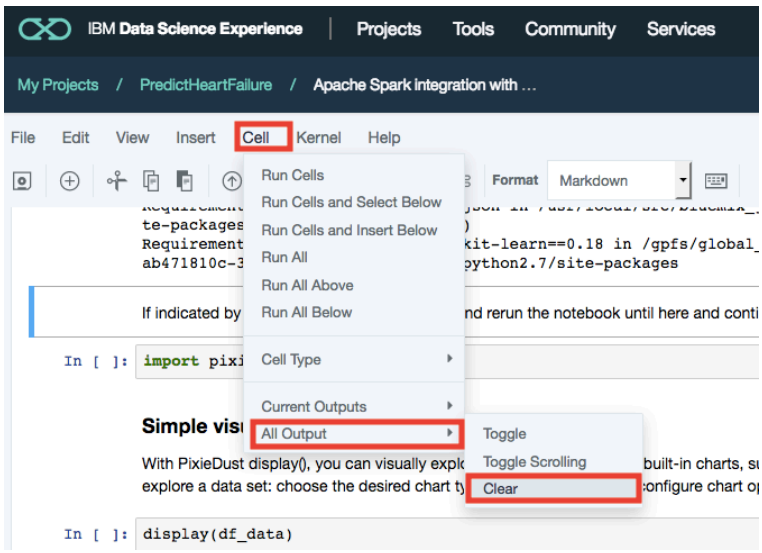
4. Click **Create Notebook** to create the new notebook.
5. Leave your browser tab open for the next part.

Step 7: Run the notebook in IBM Data Science Experience

In this part of the lab you will run the Jupyter Notebook code creating a predictive model, and save it in the Watson Machine Learning Service.

Note:

Clear the output in case ran before

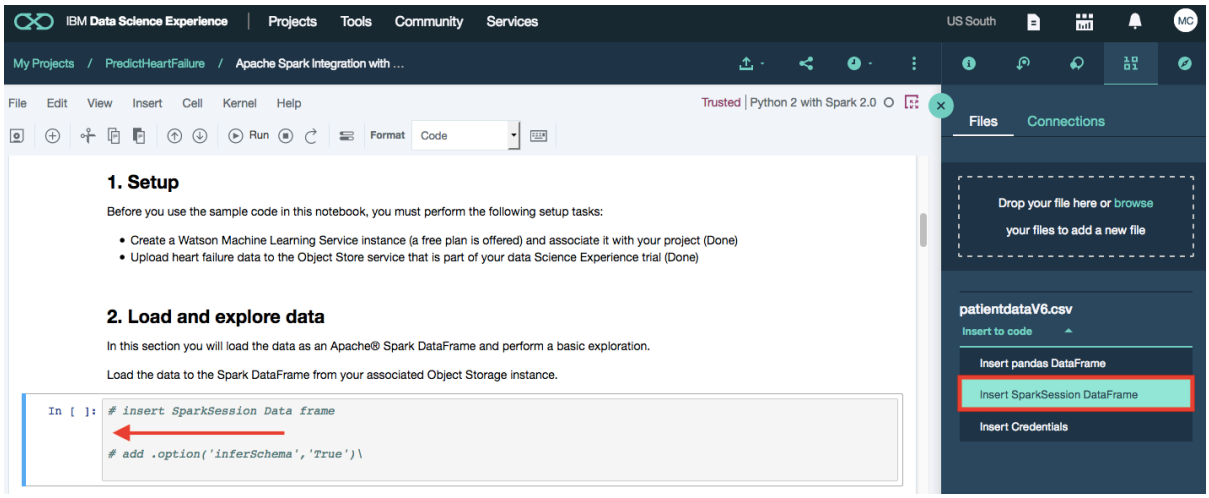


Following steps will be followed in Notebook, which was imported from URL.

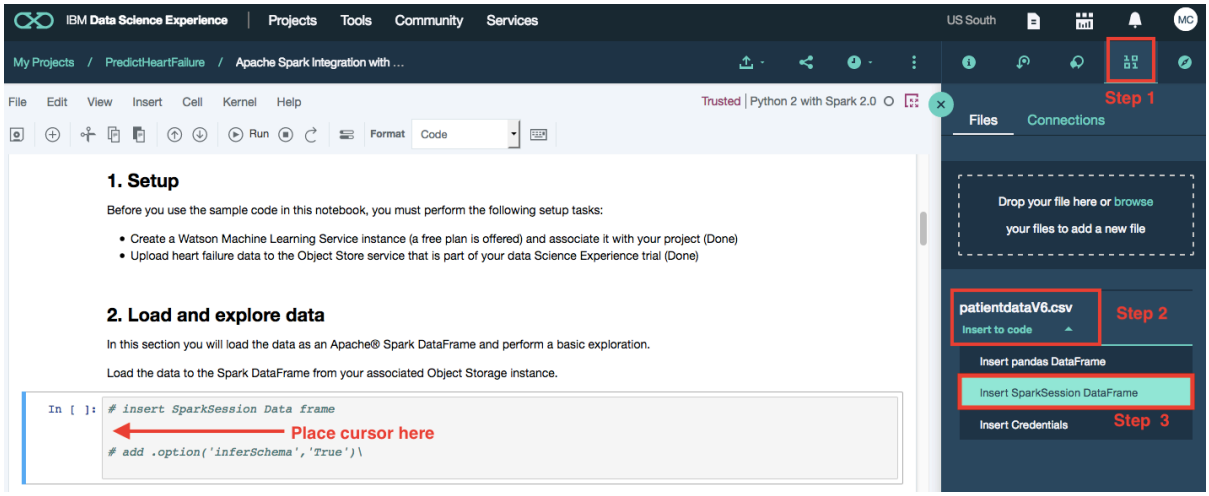
1. Setup (Completed) :

    - Create a Watson Machine Learning Service instance (a free plan is offered) and associate it with your project (Done)
    - Upload heart failure data to the Object Store service that is part of your data Science Experience trial (Done)

2. Place your cursor in this block in the notebook. Import existing data set.



Click on the **Find and Add** data icon -- see *step 1* in diagram below -- and then select **Insert to code** under the file **patientdataV6.csv**. This is *step 2* in diagram below. Finally select **Insert SparkSession DataFrame Credentials** -- which is *step 3* in diagram below.



After auto insertion of credentials, do following changes

i. change *df_data_1* to *df_data*
ii. add one more option as .option('inferSchema','True')

```
import ibmos2spark

# @hidden_cell
credentials = {
    'endpoint': 'https://s3-api.us-geo.objectstorage.service.networklayer.com',
    'api_key': 'H0wXyLtZZPqEIaXqaA9M50RNj-vD8wjJokVxErEKy7Ww',
    'service_id': 'iam-ServiceId-bc1d9355-b9d1-48be-968a-c1894ac37d7c',
    'iam_service_endpoint': 'https://iam.ng.bluemix.net/oidc/token'}

configuration_name = 'os_69e6f853ef384782888308cb979fb0b9_configs'
cos = ibmos2spark.CloudObjectStorage(sc, credentials, configuration_name, 'bluemix_cos')

from pyspark.sql import SparkSession
spark = SparkSession.builder.getOrCreate()
df_data = spark.read\
  .format('org.apache.spark.sql.execution.datasources.csv.CSVFileFormat')\
  .option('header', 'true')\
  .option('inferSchema','True')\
  .load(cos.url('patientdataV6.csv', 'predictheartfailure3e016fa9337a416ab99ded3a85a42b6d'))
df_data.take(5)
```

3. Load and explore data : In this section you will load the data as an Apache® Spark DataFrame and perform a basic exploration like, check schema , check number of records , check data.

   Explore the loaded data by using the following Apache® Spark DataFrame methods:

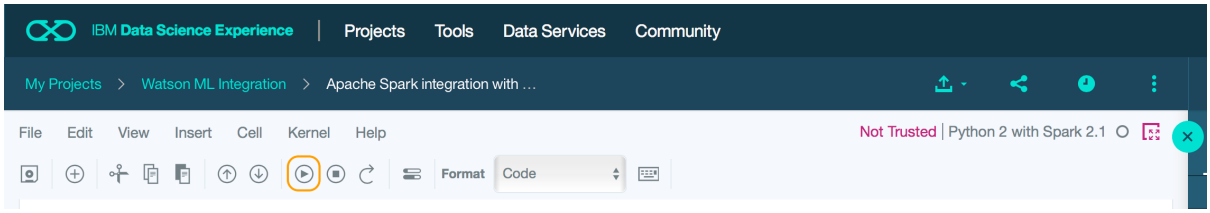
   - print schema

   Note: As you can see, the data contains ten fields. The HEARTFAILURE field is the one we would like to predict (label).

   - print top ten records
   - count all records : As you can see, the data set contains 10800 records.

   Load the data to the Spark DataFrame from your associated Object Storage instance.

4. Click on the **Run** icon to run the code in the cell.



Move your cursor to each code cell and run the code in it. Read the comments for each cell to understand what the code is doing. **Important** when the code in a cell is still running, the label to the left changes to **In [*]**:. Do **not** continue to the next cell until the code is finished running.

5. You can also see some visualizations on the existing data set with Pixie.

   For that Interactive Visualizations w/PixieDust

   a) PixieDust Install packages
   b) Simple visualization using bar charts

   With PixieDust display(), you can visually explore the loaded data using built-in charts, such as, bar charts, line charts, scatter plots, or maps. To explore a data set: choose the desired chart type from the drop down, configure chart options, configure display options.

6. Create an Apache® Spark machine learning model

   In this section, you will learn how to prepare data, create and train an Apache® Spark machine learning model.

   4.1: Prepare data : In this subsection, you will split your data into: train and test data sets.

   You will split the data set into training set and test set because once we trained the model we would want to test it with some data. This is done as a random split.

   You will see output as:

   Number of training records: 8637

   Number of testing records : 2163

   4.2: Create pipeline and train a model

   In this section, you will create an Apache® Spark machine learning pipeline and then train the model. In the first step you need to import the Apache® Spark machine learning packages that will be needed in the subsequent steps.

   A sequence of data processing is called a *data pipeline*. Each step in the pipeline processes the data and passes the result to the next step in the pipeline, this allows you to transform and fit your model with the raw input data

   you can train your Random Forest model by using the previously defined **pipeline** and **training data**. You can check your **model accuracy** now. To evaluate the model, use **test data**.
   Accuracy = 0.867314
   Test Error = 0.132686

   You can tune your model now to achieve better accuracy. For simplicity of this example tuning section is omitted.

Continue running each cell until you reach to a step where it says, **Stop here !!!!** in a notebook.

7. Persist model

   In this section, you will learn how to store your pipeline and model in Watson Machine Learning repository by using Python client libraries.

8. Authenticate to Watson Machine Learning service on IBM Cloud.

   When you get to the cell that says **Stop here !!!!** insert the username and password that you saved from your Watson Machine Learning instance into the code before running it.

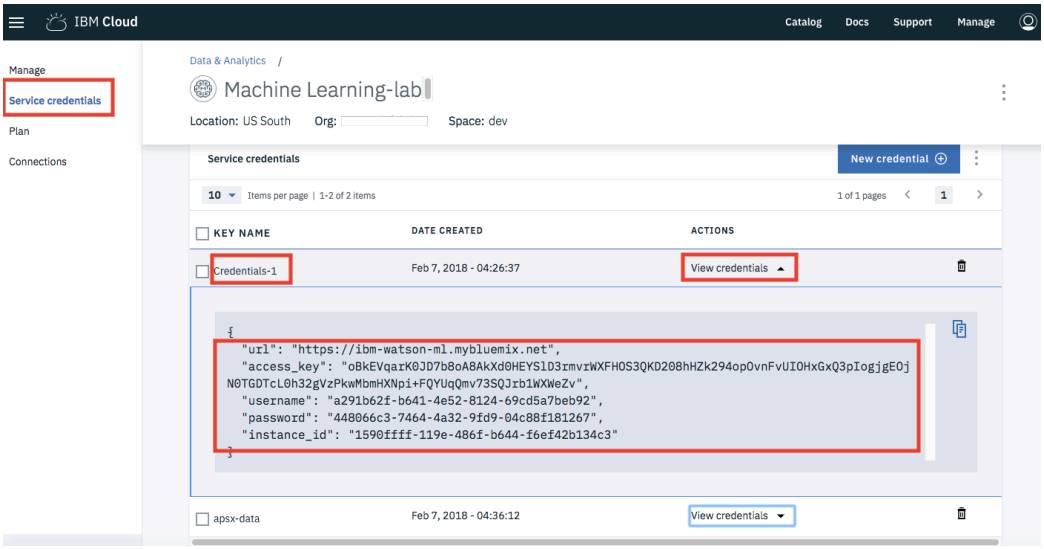
Authenticate to Watson Machine Learning service on IBM Cloud.

STOP here !!!!:

Put authentication information (username and password) from your instance of Watson Machine Learning service here.

```
In [ ]: wml_url = 'https://ibm-watson-ml.mybluemix.net'
        username = 'a291b62f-b641-4e52-8124-69cd5a7beb92'
        password = '448066c3-7464-4a32-9fd9-04c88f181267'
        instance_id= "1590ffff-119e-486f-b644-f6ef42b134c3"
```

Note this credentials from Watson Machine Learning Service created earlier in Step 3



At end of this step

Heart Failure Prediction Model

Congratulations. You've successfully created a predictive model and saved it in the Watson Machine Learning service.

Step 8: Accessing Watson ML Models and Deployments through API

Get Details about currently published model id

Get the deployment details as this stage

```
In [34]: print('{} model(s) are available in your Watson ML Service'.format(published_models['count']))
         for model in published_models['resources']:
             print('\t- name:       {}'.format(model['entity']['name']))
             print('\t  model_id:    {}'.format(model['metadata']['guid']))
             print('\t  deployments: {}'.format(model['entity']['deployments']['count']))

         1 model(s) are available in your Watson ML Service
                 - name:       Heart Failure Prediction Model
                   model_id:    c5e100fb-3892-4056-8c90-f1d7849f96b3
                   deployments: 0
```

Create a new deployment of the Model

Create a new deployment of the Model

Change the model_id to above model_id and Run this Step

```
Create a new deployment of the Model

In [ ]: model_id = 'c5e100fb-3892-4056-8c90-f1d7849f96b3'

deployment_url = wml_url + "/v3/wml_instances/" + instance_id + "/published_models/" + model_id + "/deployments"

payload = "{\"name\": \"Heart Failure Prediction Model \", \"description\": \"First deployment of Heart Failure Prediction Model\", \"type\": \"online
headers = {'authorization': 'Bearer ' + watson_ml_token, 'content-type': "application/json" }

response = requests.request("POST", deployment_url, data=payload, headers=headers)

print(response.text)
```

Monitor the status of Deployment

```
In [34]: deployment = json.loads(response.text)

         print('Model {} deployed.'.format(model_id))
         print('\tname: {}'.format(deployment['entity']['name']))
         print('\tdeployment_id: {}'.format(deployment['metadata']['guid']))
         print('\tstatus: {}'.format(deployment['entity']['status']))
         print('\tscoring url: {}'.format(deployment['entity']['scoring_url']))

         Model c5e100fb-3892-4056-8c90-f1d7849f96b3 deployed.
                 name: Heart Failure Prediction Model
                 deployment_id: b53bc50b-2db5-4640-abc9-b92c78dc0d99
                 status: INITIALIZING
                 scoring url: https://ibm-watson-ml.mybluemix.net/v3/wml_instances/1590ffff-119e-486f-b644-f6ef42b134c3/published_models/c5e100fb-3892-4056-
         8c90-f1d7849f96b3/deployments/b53bc50b-2db5-4640-abc9-b92c78dc0d99/online
```

Monitor the status of deployment

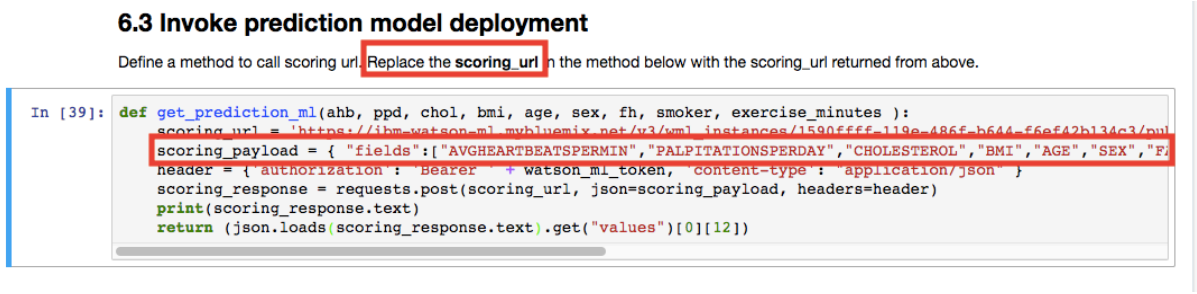
Note scoring_url from this step

```
In [38]: deployment_details = json.loads(response.text)

         for resources in deployment_details['resources']:
             print('name: {}'.format(resources['entity']['name']))
             print('status: {}'.format(resources['entity']['status']))
             print('scoring url: {}'.format(resources['entity']['scoring_url']))

         name: Heart Failure Prediction Model
         status: ACTIVE
         scoring url: https://ibm-watson-ml.mybluemix.net/v3/wml_instances/1590ffff-119e-486f-b644-f6ef42b134c3/publis
         hed_models/c5e100fb-3892-4056-8c90-f1d7849f96b3/deployments/0fe68bcf-aaf3-4890-bbb2-d6b5abfc1ae7/online
```

Invoke prediction model deployment



And Finally, Call get_prediction_ml method exercising our prediction model

print('Is a 44 year old female that smokes with a low BMI at risk of Heart Failure?: {}'.format(get_prediction_ml(100,85,242,24,44,"F","Y","Y",125)))

RESULT:
Is a 44 year old female that smokes with a low BMI at risk of Heart Failure?: 1.0

If the value is 1.0: Means There are chances of Heart Failure and If the value is 0.0 Then No Heart Failure predicted.

Test this for another set of data:

print('Is a 42 year old male that smokes with a low BMI at risk of Heart Failure?: {}'.format(get_prediction_ml(100,85,178,22,42,"M","N","N",45)))

Is a 42 year old male that smokes with a low BMI at risk of Heart Failure?: 0.0

RESULT:
No Heart Failure predicted as value is 0.0

Congratulations, you successfully created a predictive model in Apache Spark and deployed and tested it using the Watson Machine Learning Service in IBM Cloud 😃!

Optional (Explore more Estimators)

This is optional part of the Lab, You can explore this notebook, that guides you through comparing various estimators and tweaking with the hyper parameter

Exploration of Models and Hyperparameters.ipynb
Note: Notebook has been commented with the steps to be followed

Additional links

- More data science journeys on IBM Code: https://developer.ibm.com/code/journey/category/data-science/
- IBM Data Science Experience: https://www.ibm.com/analytics/us/en/watson-data-platform/data-science-experience/
- Watson Data Platform: https://www.ibm.com/analytics/us/en/watson-data-platform/