

# Build, Train, Save, Deploy and Test a Convolutional Neural Network Model using MNIST

## Introduction

This lab will use the [MNIST](#) computer vision data set to train a deep learning model to recognize handwritten digits. A simple convolutional neural network built using Keras will be submitted to Watson Machine Learning for training. The trained model will be saved in the model repository, deployed, and scored. The lab will use a notebook to programmatically accomplish these tasks.

## Objectives

Upon completing the lab, you will know how to programmatically:

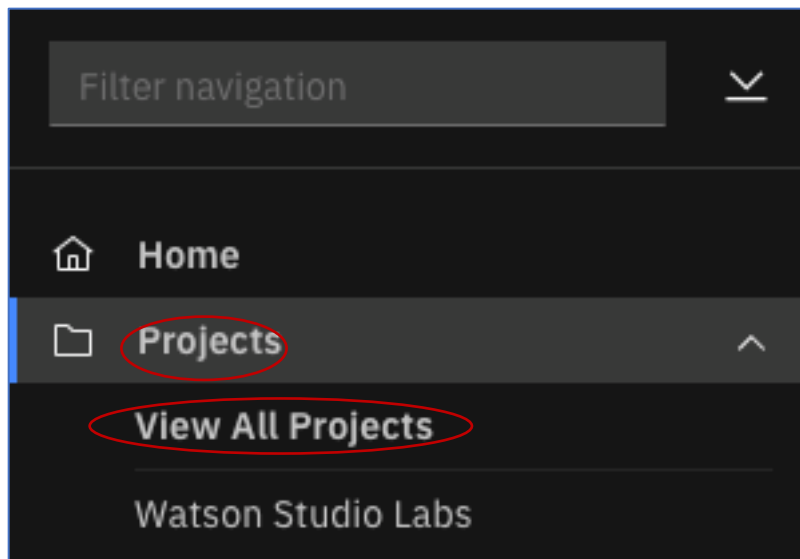
1. Set up the data and result buckets in IBM Cloud Storage
2. Set up the training definition
3. Train the model
4. Monitor the training progress and results
5. Save and Deploy the Trained Model
6. Test the Deployed Model

## Lab Steps

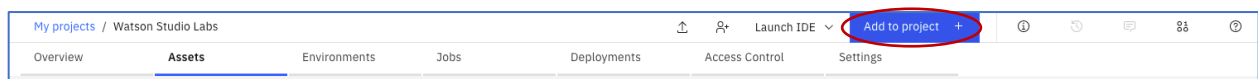
### Step 1 - Create a Jupyter Notebook

1. Click on the hamburger icon , then click on **Projects**, and then **Watson Studio Labs** (or whatever you named the project)

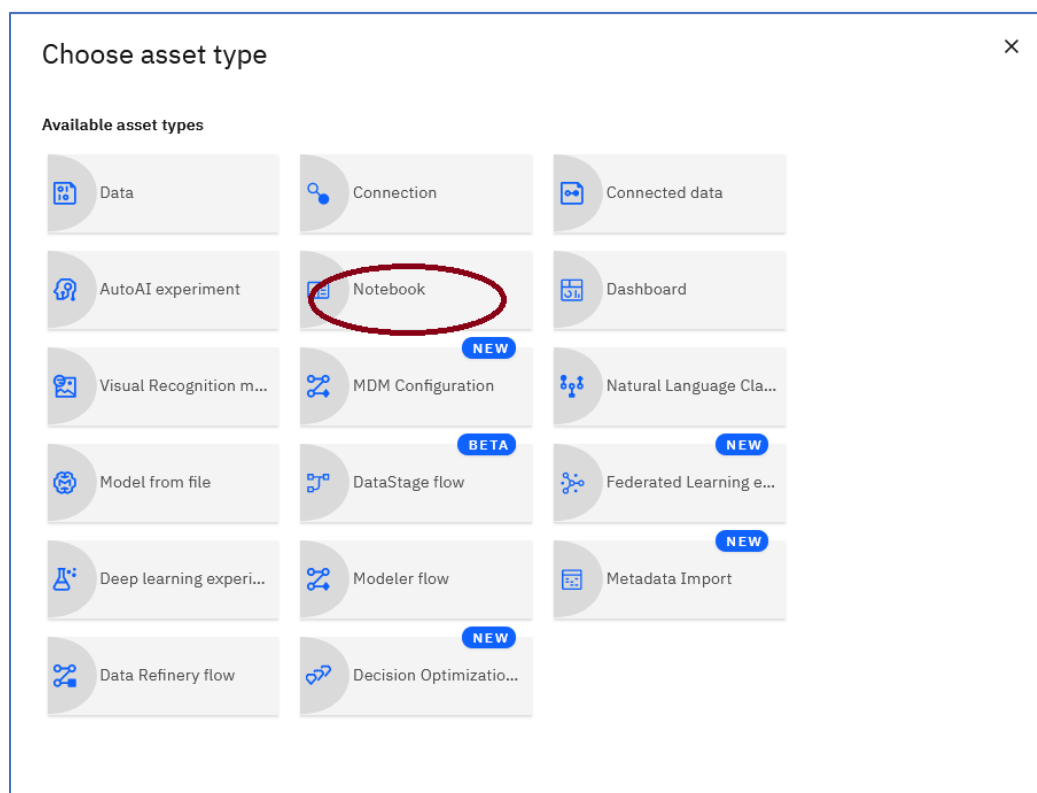




2. We are now going to create a notebook in our project. This notebook will be created from a url that points to the Keras-MNIST example notebook in the github repository. Click the **Add to project** link.



3. Click on **Notebook**



4. Click on **From URL** under **New Notebook**, enter **MNIST-Example** for the **Name**, and optionally enter a **Description**. Leave the default for the **Runtime**.

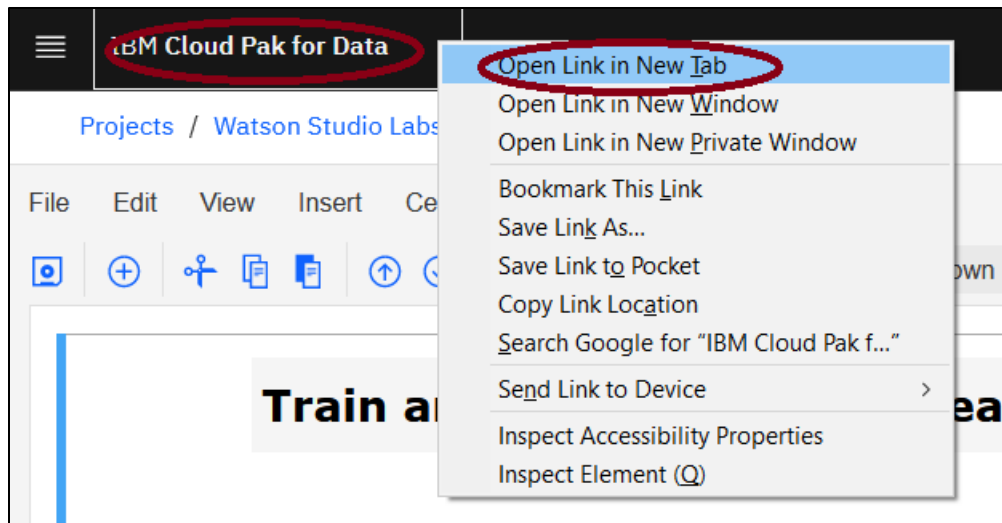
Use Copy Hyperlink and paste the following url into the **Notebook URL** field.

[https://github.com/bleonardb3/ML\\_POT\\_03-25-2021/blob/main/Lab-7/MNIST-Example.ipynb](https://github.com/bleonardb3/ML_POT_03-25-2021/blob/main/Lab-7/MNIST-Example.ipynb)

Click **Create Notebook**.

The screenshot shows the 'New Notebook' dialog in IBM Watson Studio. The 'From URL' tab is selected. The 'Name' field is 'MNIST-Example'. The 'Description (optional)' field is empty. The 'Select runtime' dropdown is set to 'Default Python 3.7 XS (2 vCPU 8 GB RAM)'. The 'Notebook URL' field contains the GitHub link. The 'Create' button is highlighted.

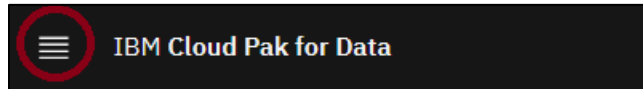
5. Before executing the code in the notebook, we need to do the following:
  - a. Obtain the Cloud Object Storage credentials.
  - b. Obtain an api key and the location of the WatsonMachineLearning service. We need to copy this information into a Notebook cell. This information is required to work with the Watson Machine Learning API. Note, we already did this in Lab-5. Go to the Lab-5 notebook to get these credentials. Note, if you skipped Lab-5, go to the Appendix below to follow the procedure.
6. To get the Cloud Object Storage credentials, right-click on **IBM Cloud Pak for Data**, and then click on **Open Link in New Tab**.

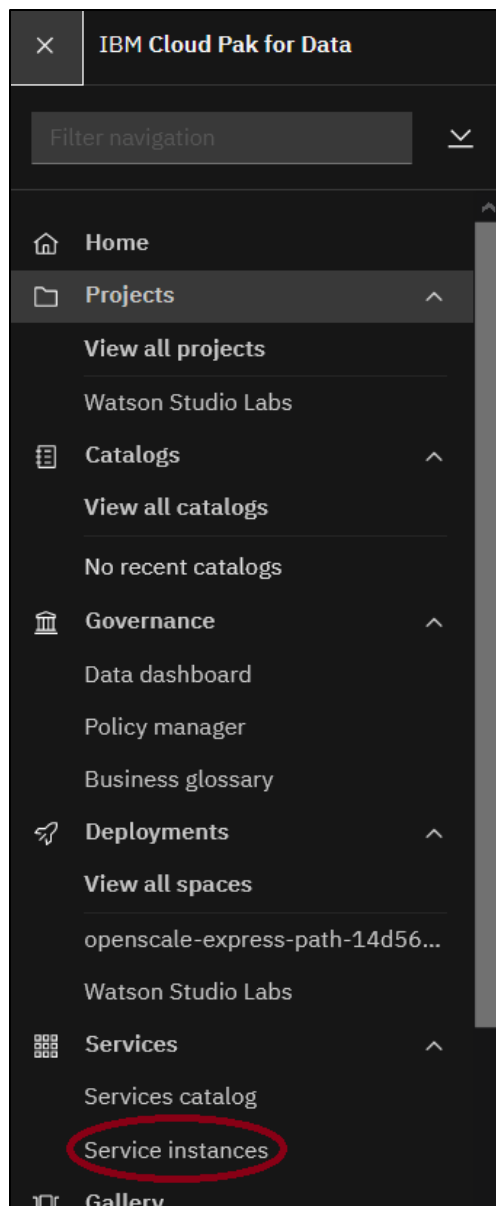



7. Click on the new **IBM Cloud Pak for Data** browser tab.



8. Click on the hamburger icon , and click on **Services**, and then **Service instances**.





9. Click on vertical ellipse  on the right side of the Cloud Object Storage service. Click on the **Manage in IBM Cloud** entry.

**Service instances**

Your existing service instances are listed below. You must [upgrade](#) from a Lite account to a paid account before upgrading individual services.

Filter by: Resource Groups 2 x Locations None Service Type Service Plan

Find service instances Add service +

Name	Type	Plan	Location	Group	
Cloud Object Storage-sv <a href="#">🔗</a>	Cloud Object Storage	Lite	Global	Default	
WatsonMachineLearning	Machine Learning	Lite	Dallas	Default	
KnowledgeCatalog	Watson Knowledge Catalog	Lite	Dallas	Default	
Watson OpenScale-9q <a href="#">🔗</a>	Watson OpenScale	Lite	Dallas	Default	
WatsonStudio	Watson Studio	Lite	Dallas	Default	

Upgrade service  
 Launch  
 Manage in IBM CL...  
 Delete

## 10. Click on **Service credentials**

[Resource list](#) / **Cloud Object Storage-sv** Active cpdaas [🔗](#) Aspera transfers Details Actions...

Getting started

**Buckets**

Endpoints

**Service credentials**

Connections

Usage details

Plan

Prefix filter 🔍

Create bucket +

Name	Public access	Location	Storage class	Created	Attributes
1864e385-fe7a-4638-8c02-4b5d996c9ce9		us-south	Standard	10/20/2020 11:45:25 PM	<a href="#">View</a>
528c3789-95a7-4834-9b39-881c119803ca		us-south	Standard	10/15/2020 7:07:18 AM	<a href="#">View</a>
watsonstudiolabs-donotdelete-pr-riv0xgsmajceyu		us-geo	Standard	10/14/2020 10:01:32 PM	<a href="#">View</a>

Items per page: 10 1-10 of all items page 1

## 11. Click on **New Credentials**.

[Resource list](#) / **Cloud Object Storage-sv** Active cpdaas [🔗](#) Aspera transfers Details Actions...

Getting started

Buckets

Endpoints

**Service credentials**

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)

Search credentials...

**New credential** +

## 12. Type in **keras** for the Name and click on the down arrow icon adjacent to **Advanced options**.

Create credential

Name:

keras

Role: ⓘ

Writer

Advanced options

Cancel

Add

13. Enable the **Include HMAC Credentials**. This is required for Watson Machine Learning interface to Cloud Object Storage. Click on **Add**.

Create credential

Name:

keras

Role: ⓘ

Writer

Advanced options

Select Service ID (Optional) ⓘ

Auto Generate

Include HMAC Credential ⓘ

On

Provide service-specific configuration parameters in a valid JSON object (Optional)


Choose file

Add inline configuration parameters (Optional)


```
{
  "HMAC": true
}
```

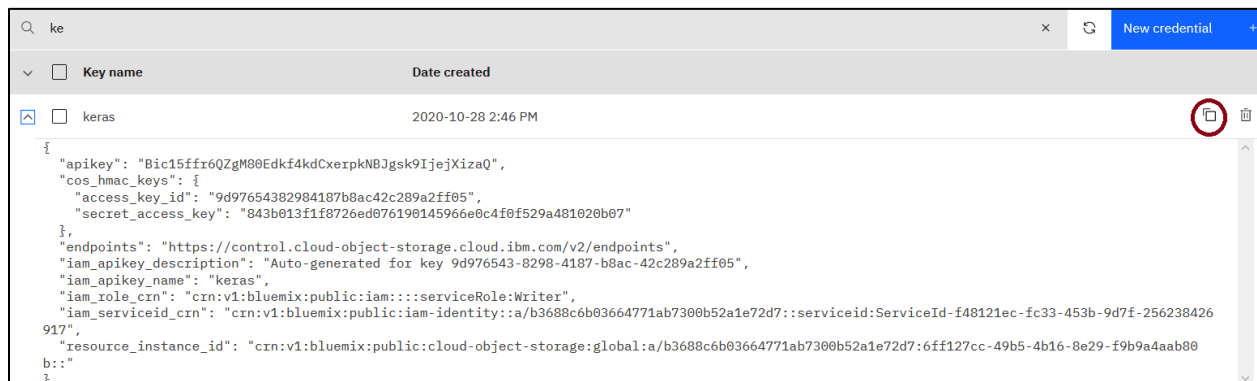
Cancel

Add

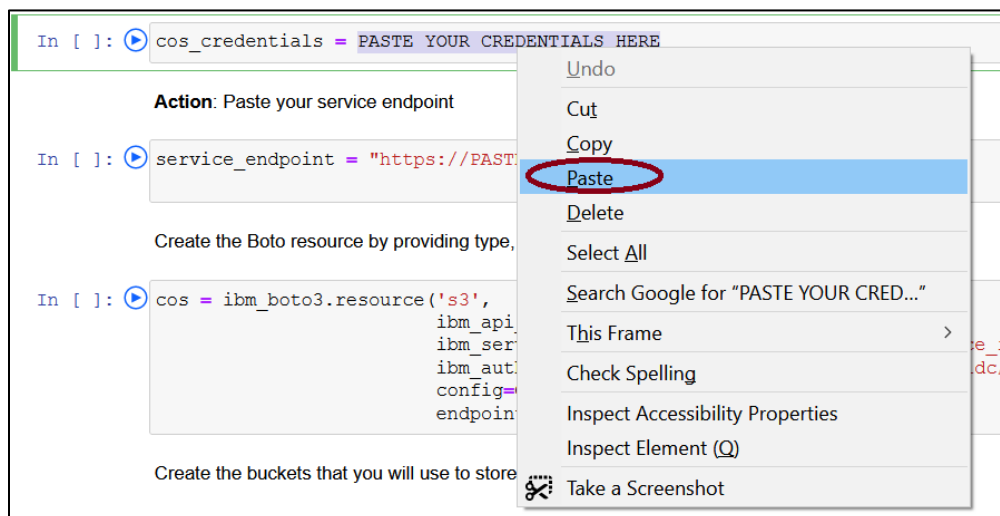
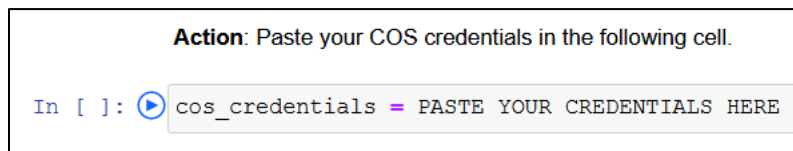
14. Enter keras in the Search field. Click on the down arrow icon  adjacent to keras.



15. Click on the copy icon .



16. Highlight “PASTE YOUR CREDENTIALS HERE” (under the **Action:** Paste your COB credentials in the following cell). Right-click and then click on **Paste**.





17. The credentials should appear similar to below with different values.

```
**Action**: Paste your COS credentials in the following cell.

In [2]: cos_credentials = {
    "apikey": "Bic15ffr6QZgM80Edkf4kdCwexpkNBjgsk9IjejXizaQ",
    "cos_hmac_keys": {
        "access_key_id": "9d97654382984187b8ac42c289a2ff05",
        "secret_access_key": "843b0131f8726ed076190145966e0c4f0f529a481020b07"
    },
    "endpoints": "https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints",
    "iam_apikey_description": "Auto-generated for key 9d976543-8298-4187-b8ac-42c289a2ff05",
    "iam_apikey_name": "keras",
    "iam_role_crn": "crn:vl:bluemix:public:iam:::serviceRole:Writer",
    "iam_serviceid_crn": "crn:vl:bluemix:public:iam-identity:a/b3688c6b03664771ab7300b52a1e72d7::serviceid:ServiceId-f48121ec-fc33-453b-9d7f-256238426917",
    "resource_instance_id": "crn:vl:bluemix:public:cloud-object-storage:global:a/b3688c6b03664771ab7300b52a1e72d7:6ff127cc-49b5-4b16-8e29-f9b9a4aab80b::"
}
```

18. Return to the Cloud Object Storage tab and click on **Endpoints**.

Resource list / Cloud Object Storage-sv Active cpdaas

Getting started

Buckets

**Endpoints**

Service credentials

Connections

Usage details

Plan

Service credentials

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)

Key name	Date created
keras	2020-10-28 2:46 PM

19. Click on the appropriate geo where you account has been set up.

Resource list / Cloud Object Storage-sv Active cpdaas Aspera transfers Details Actions...

Getting started

Buckets

**Endpoints**

Service credentials

Connections

Usage details

Plan

Endpoints

Endpoints are used with your credentials (Bucket name, API Key, SDK) to tell your service where to look for your bucket. Choose an endpoint URL that is located in the same region as your service or application. [Learn more](#)

Select resiliency: Cross Region

Select location: **ap-geo**

	Public	Private	Direct
ap-geo	s3.ap.cloud-object-storage.appdomain.cloud	s3.private.ap.cloud-object-storage.appdomain.cloud	s3.direct.ap.cloud-object-storage.appdomain.cloud
Tokyo	s3.tok.ap.cloud-object-storage.appdomain.cloud	s3.private.tok.ap.cloud-object-storage.appdomain.cloud	s3.direct.tok.ap.cloud-object-storage.appdomain.cloud
Seoul	s3.seo.ap.cloud-object-storage.appdomain.cloud	s3.private.seo.ap.cloud-object-storage.appdomain.cloud	s3.direct.seo.ap.cloud-object-storage.appdomain.cloud
Hong Kong	s3.hkg.ap.cloud-object-storage.appdomain.cloud	s3.private.hkg.ap.cloud-object-storage.appdomain.cloud	s3.direct.hkg.ap.cloud-object-storage.appdomain.cloud

20. My account is in us-geo. Copy the Public endpoint corresponding to your geo.

Resource list / **Cloud Object Storage-sv** Active cpdaas

Getting started  
Buckets  
**Endpoints**  
Service credentials  
Connections  
Usage details  
Plan

### Endpoints

Endpoints are used with your credentials (Bucket name, API Key, SDK) to tell your service where to look for your bucket. Choose an endpoint URL that is located in the same region as your service or application. [Learn more](#)

Select resiliency: Cross Region Select location: us-geo

	Public	Private
us-geo	s3.us.cloud-object-storage.appdomain.cloud	s3.private.us.cloud-object-storage.appdomain.cloud
Dallas	s3.dal.cloud-object-storage.appdomain.cloud	s3.private.dal.us.cloud-object-storage.appdomain.cloud
Washington	s3.wdc.cloud-object-storage.appdomain.cloud	s3.private.wdc.us.cloud-object-storage.appdomain.cloud
San Jose	s3.sjc.cloud-object-storage.appdomain.cloud	s3.private.sjc.us.cloud-object-storage.appdomain.cloud
Legacy Endpoints		

A right-click context menu is open over the 'us-geo' public endpoint, with the 'Copy' option highlighted.

21. Return to the notebook, and highlight “PASTE SERVICE ENDPOINT HERE” (underneath the Action: Paste your service endpoint). Right-click and click on **Paste**.

**Action: Paste your service endpoint**

```
In [6]: service_endpoint = "https://PASTE SERVICE ENDPOINT HERE"
```

```
In [6]: service_endpoint = "https://PASTE SERVICE ENDPOINT HERE"
```

Create the Boto resource by providing type, endpoint\_url


```
In [7]: cos = ibm_boto3.resource('s3',
                                ibm_api_key_id=co
                                ibm_service_insta
                                ibm_auth_endpoint
                                config=Config(sig
                                endpoint_url=serv
```

Create the buckets that you will use to store training data

**Note::** Bucket name has to be unique - please update following ones to any unique name.

A right-click context menu is open over the 'PASTE SERVICE ENDPOINT HERE' text, with the 'Paste' option highlighted.

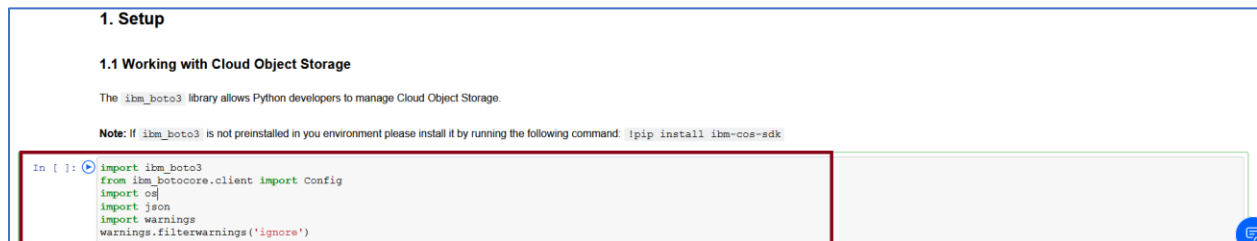
22. The service endpoint should appear similar to below depending on the geo.



**Action:** Paste your service endpoint

```
In [6]: service_endpoint = "https://s3.us.cloud-object-storage.appdomain.cloud"
```

23. Return to the top of the notebook, read through the documentation in the beginning and then select the first code cell to execute.



**1. Setup**

**1.1 Working with Cloud Object Storage**

The `ibm_boto3` library allows Python developers to manage Cloud Object Storage.

**Note:** If `ibm_boto3` is not preinstalled in your environment please install it by running the following command: `!pip install ibm-cos-sdk`

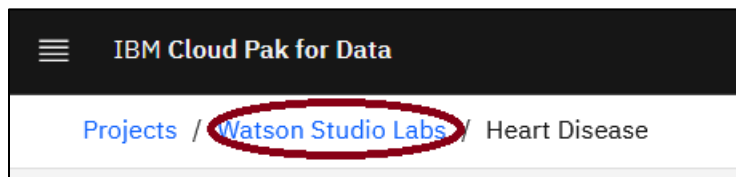
```
In [ ]: import ibm_boto3
        from ibm_botocore.client import Config
        import os
        import json
        import warnings
        warnings.filterwarnings('ignore')
```

For those not familiar with Jupyter notebooks, read below.

A Jupyter notebook consists of a series of cells. These cells are of 2 types (1) documentation cells containing markdown, and (2) code cells (denoted by a bracket on the left of the cell) where you write Python code, R, or Scala code depending on the type of notebook. Code cells can be run by putting the cursor in the code cell and pressing **<Shift><Enter>** on the keyboard. Alternatively, you can execute the cells by clicking on **Run icon** on the menu bar that will run the current cell (where the cursor is located) and then select the cell below. In this way, repeatedly clicking on **Run** executes all the cells in the notebook. When a code cell is executed the brackets on the left change to an asterisk **\*** to indicate the code cell is executing. When completed, a sequence number appears. The output, if any, is displayed below the code cell.

Execute each of the notebook cells in order (either by typing in **<Shift><Enter>** or using the **Run** menu option). Read the notebook documentation to gain an understanding of the code that is executing.

24. Click Watson Studio Labs to exit out of the notebook.



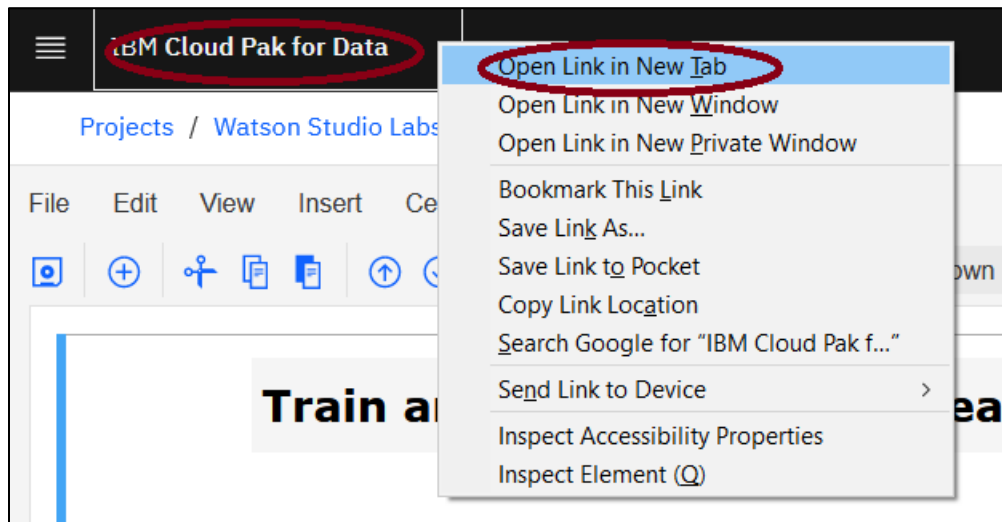
## You have completed the Lab!

- ✓ Set up the data and result buckets in IBM Cloud Storage
- ✓ Set up the training definition
- ✓ Trained the model
- ✓ Monitored the training progress and results
- ✓ Saved and Deployed the Trained Model
- ✓ Tested the Deployed Model

## Appendix


If you have skipped Lab-5, here are the steps to get the Watson Machine Learning api key and location.

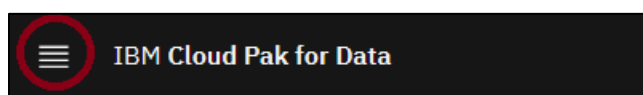
1. Right-click on **IBM Cloud Pak for Data**, and then click on **Open Link in New Tab**.

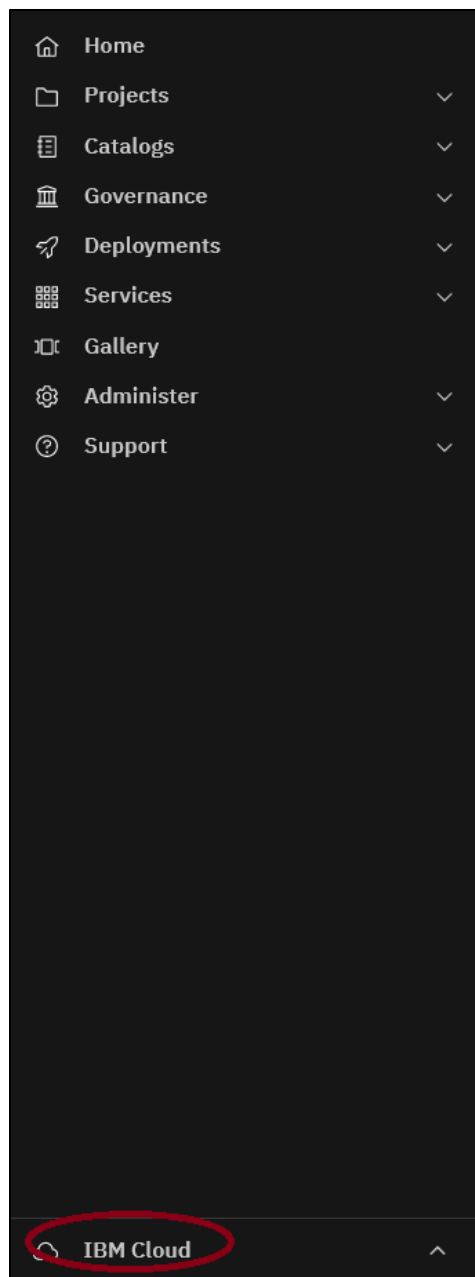


2. Click on the new **IBM Cloud Pak for Data** browser tab.

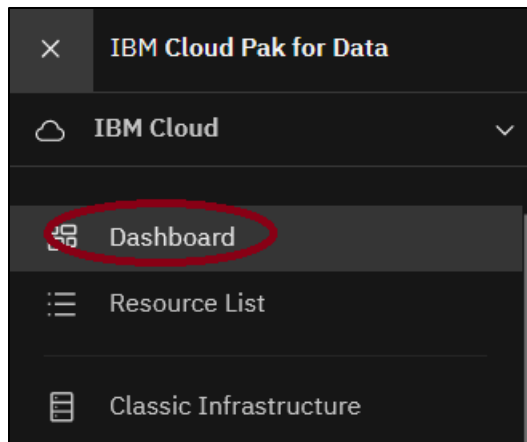


3. Click on the hamburger icon , and then scroll down to the bottom to click on **IBM Cloud**.





4. Click on **Dashboard**



5. Click on the IBM Command Shell icon .



6. Wait for the shell to be initialized. To create an api key, copy and paste the following line next to the command shell prompt and press the <Enter> key.

```
ibmcloud iam api-key-create API_KEY_NAME
```

You can view this entry in the screen image below highlighted in **maroon**.

The newly created api key is shown highlighted in **blue** below. This key is unique to my account. Your key will be unique to your account.

h

To get the location of the WatsonMachineLearning service instance, copy and paste the following line next to the command shell prompt and press the <Enter> key.

```
ibmcloud resource service-instance WatsonMachineLearning
```

You can view this entry in the screen image below also highlighted in **maroon**.

The location of the WatsonMachineLearning instance is shown highlighted in **blue** below. My instance location is shown to be in us-south.

```
Welcome to IBM Cloud Shell!
Image version: 1.0.9
Help us improve future releases by clicking Feedback to share your experience!

Note: Your Cloud Shell session is running in Dallas (us-south). Your workspace includes 500 MB of temporary storage. This session will close after
, your workspace data is removed. To track your usage, go to Usage quota in the Cloud Shell menu.

Tip: Enter 'ibmcloud' to use the IBM Cloud CLI. The Dallas (us-south) region is targeted by default. You can switch the region by running 'ibmcloud

wsuser56000@cloudshell:~$ ibmcloud iam api-key-create API_KEY_NAME
Creating API key API_KEY_NAME under 1aab03f27a8b47e2be36e68574e8fcce as wsuser56000@gmail.com...
OK
API key API_KEY_NAME was created

Please preserve the API key! It cannot be retrieved after it's created.

ID          ApiKey-413e0d2a-cadc-4388-b4e7-2da4d02b8de9
Name        API_KEY_NAME
Description
Created At  2020-10-28T03:24:0000
API Key     WeqpHa-Av0wn2PhEE_CXGcGK11qkcJewwPQI51mrFT19t
Locked     false

wsuser56000@cloudshell:~$ ibmcloud resource service-instance WatsonMachineLearning
Retrieving service instance WatsonMachineLearning in all resource groups under account Steven Doe's Account as wsuser56000@gmail.com...
OK

Name:        WatsonMachineLearning
ID:          crn:v1:bluemix:public:pm-20:us-south:a/1aab03f27a8b47e2be36e68574e8fcce:93d46d98-4f8a-4547-8838-cbf1f9065455::
GUID:        93d46d98-4f8a-4547-8838-cbf1f9065455
Location:    us-south
Service Name: pm-20
Service Plan Name: lite
Resource Group Name: Default
State:       active
Type:        service instance
```