

A
Report on
“Predictive Modeling Analytics”
(CS-18.338)

In partial fulfillment of requirements for the degree of

Bachelor of Technology (B.Tech.)
in
Computer Science and Engineering



Topic: Mobile Price Classification

Submitted by:
Ms. Akshita Kanther(180161)
Ms. Hiral Jain(180167)

Under the Guidance of
Ms. Khushbu Datwani

Department of Computer Science and Engineering
SCHOOL OF ENGINEERING AND TECHNOLOGY
Mody University of Science and Technology
Lakshmangarh, Distt. Sikar-332311
May 2021

ACKNOWLEDGMENT

We would like to express our gratitude to our project mentor **Ms. Khushbu Datwani** for her guidance and encouragement for successful completion of this project. Her kindness and help have been the source of encouragement for us. We would also like to extend special thanks to IBM for providing us such a passionate and motivated trainer.

Akshita Kanther

Hiral Jain

TABLE OF CONTENTS

S.No.	Topic	Pg. No.
1	Introduction	3
2	Dataset	4
3	Tools and Working Environment	4
4	Data Pre-processing	4
5	Data Visualization	19
6	Visual Recognition Service	22
7	AutoAI Experiment(Machine Learning model)	34
8	Python Notebook	74
9	Conclusion	82

INTRODUCTION

WHAT IS MACHINE LEARNING?

Machine learning is the science of getting computers to act without being explicitly programmed. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn. It focuses on the development of computer programs that can access data and use it to learn for themselves.

APPROACHES TO MACHINE LEARNING:-

- **Supervised learning** - can apply what has been learned in the past to new data using labelled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.
- **Unsupervised learning** - are used when the information used to train is neither classified nor labelled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabelled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabelled data.
- **Reinforcement learning** - is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behaviour within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.

Dataset: Mobile Price Classification

This project is to analyse a dataset related to mobiles and classify the mobile on the basis of different price ranges (low, medium, high and very high). It contains information about battery power, RAM, Bluetooth, Wi-Fi, height, width, price range, etc. of the mobiles of different types and companies. This data set contains 2000 rows and 21 columns.

Features of this project are as follows:-

- I. Explore the dataset, Data refining , cleaning and visualizations
- II. Visual Recognition Service
- III. Implementing the Machine Learning Model on selected data sets and deploying your model with the creation of a Job and testing
- IV. Concept of python using Jupyter notebook in IBM Watson

TOOLS AND WORKING ENVIRONMENT-

Project will be made using IBM Watson and Python Notebook on IBM cloud platform.

Python libraries used are as follows:

- Numpy
- Matplotlib
- Seaborn
- Pandas
- Scikit-Learn

DATA PREPROCESSING-

We use Data Refinery to cleanse and shape tabular data with a graphical flow editor. We can also use interactive templates to code operations, functions, and logical operators. When we cleanse data, we fix or remove data that is incorrect, incomplete, improperly formatted, or duplicated. When we shape data, we customize it by filtering, sorting, combining or removing columns, and performing operations.

We create a Data Refinery flow as a set of ordered operations on data. Data Refinery includes a graphical interface to profile our data to validate it and over 20 customizable charts that

give us perspective and insights into our data. When we save the refined data set, we typically load it to a different location than where we read it from. In this way, our source data remains untouched by the refinement process.

The Data Refinery service reduces the amount of time it takes to prepare data. Use pre-defined operations that we can use in our data flows to transform large amounts of raw data into consumable, quality data that's ready for analysis.

- **Required service**

Watson Studio or Watson Knowledge Catalog

- **Data format**

Avro, CSV, JSON, Parquet, TSV (read only), or delimited text data asset

Tables in relational data sources

- **Data size**

Any. Data Refinery operates on a sample subset of rows in the data set. The sample size is 1 MB or 10,000 rows, whichever comes first. However, when you run a job for the Data Refinery flow, the entire data set is processed.

With Data Refinery, we can:

- Interactively discover, cleanse, and transform our data with over 100 built-in operations. No coding is required.
- Understand the quality and distribution of our data using dozens of built-in charts, graphs, and statistics.
- Automatically detect data types and business classifications.
- Schedule data flow executions for repeatable outcomes.

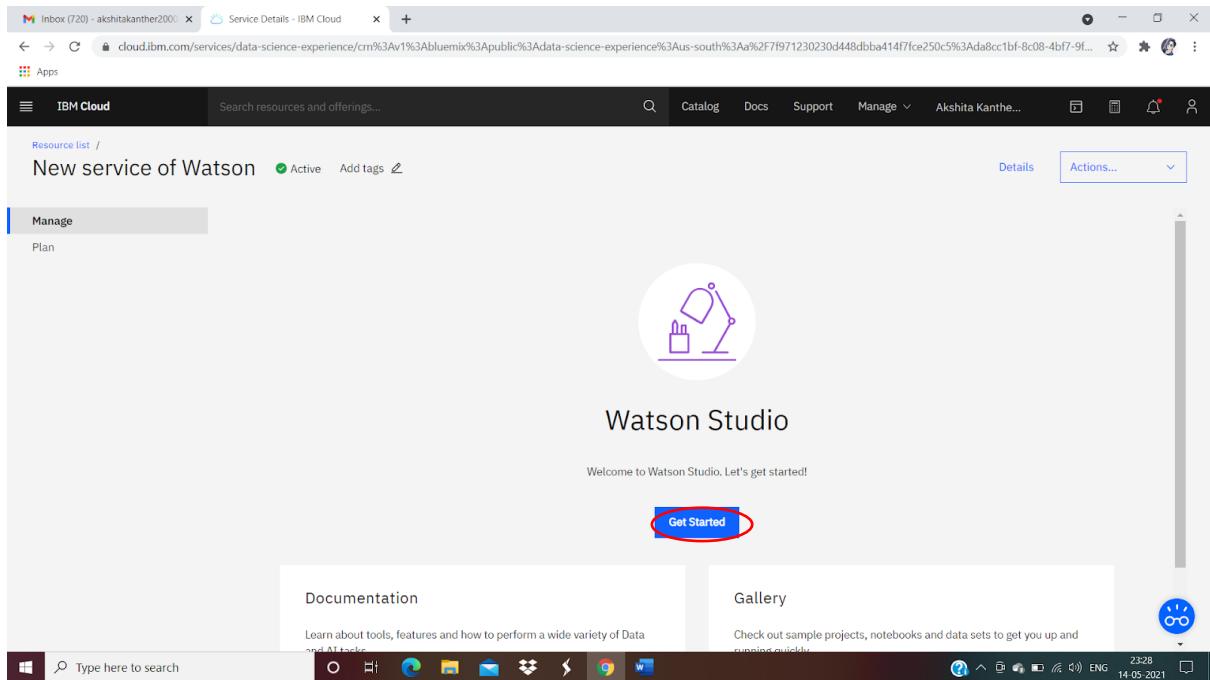
Step-1. Login to the IBM cloud platform and click on Services under Resource summary or click on Resource list from the Navigation pane on the left side of the screen.

The screenshot shows the IBM Cloud dashboard. The left sidebar has a red box around the 'IBM Cloud' icon. The main area features a 'Quick start' section with five cards: 'Build' (Explore IBM Cloud with this selection of easy starter tutorials and services), 'Monitor your resources' (Get visibility into the performance and health of your resources), 'Automate security and compliance' (Embed security checks into everyday workflows and centrally manage your compliance to organization and regulatory guidelines), 'Create an OpenShift cluster' (Deploy apps on highly available clusters with Red Hat OpenShift on IBM Cloud), and 'Browse, select, and create a database' (Choose from our range of available databases to store critical data and documents for your system. Includes deployment time). Below this is a 'Resource summary' section showing 6 resources (Cloud Foundry apps and Services), a 'Planned maintenance' section (Clear skies!), and a 'For you' section with links to host website assets in Cloud Object Storage and accelerate delivery of static files.

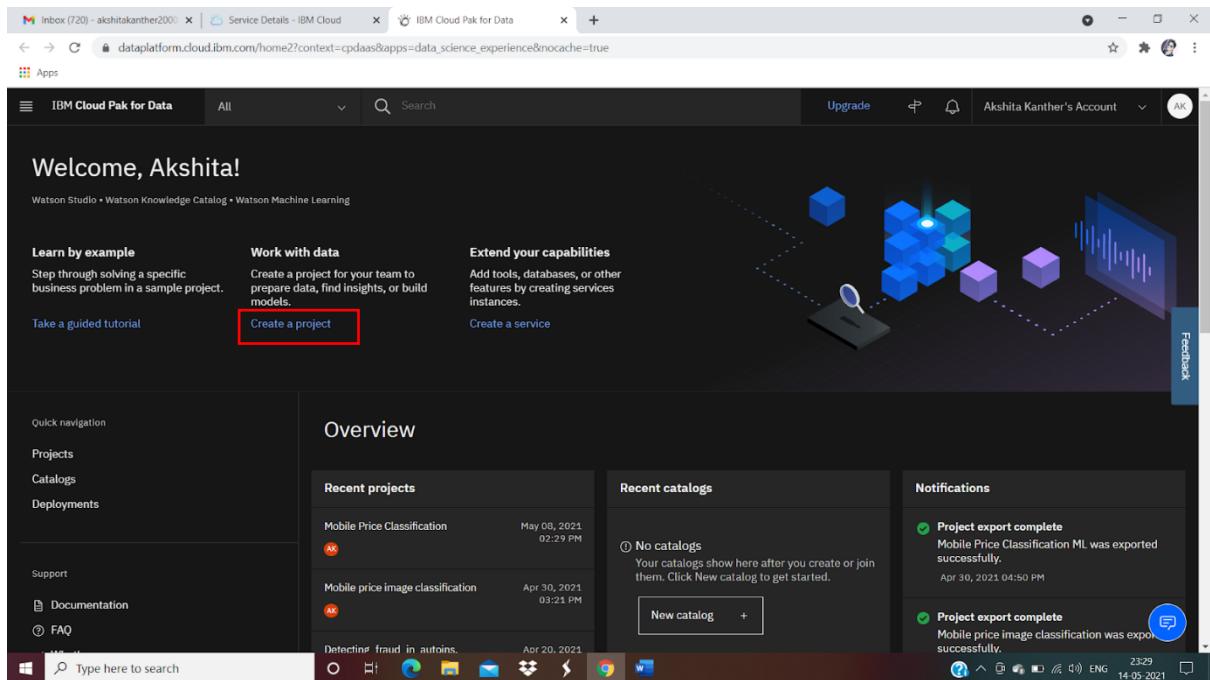
Step-2. Click on Watson service instance that is previously created. If Watson service instance is not previously created then create it.

The screenshot shows the 'Resource list' page in the IBM Cloud interface. The left sidebar has a red box around the 'IBM Cloud' icon. The main area displays a table of resources. In the 'Services' category, there are four entries: 'KnowledgeCatalog' (Default location, Watson Knowledge Catalog product, Active status), 'Machine Learning-ph' (Default location, Machine Learning product, Active status, with a tag 'cpdas'), 'New service of Watson' (Default location, Watson Studio product, Active status), and 'Visual Recognition-5m' (Default location, Visual Recognition product, Active status). The 'New service of Watson' row is highlighted with a red box.

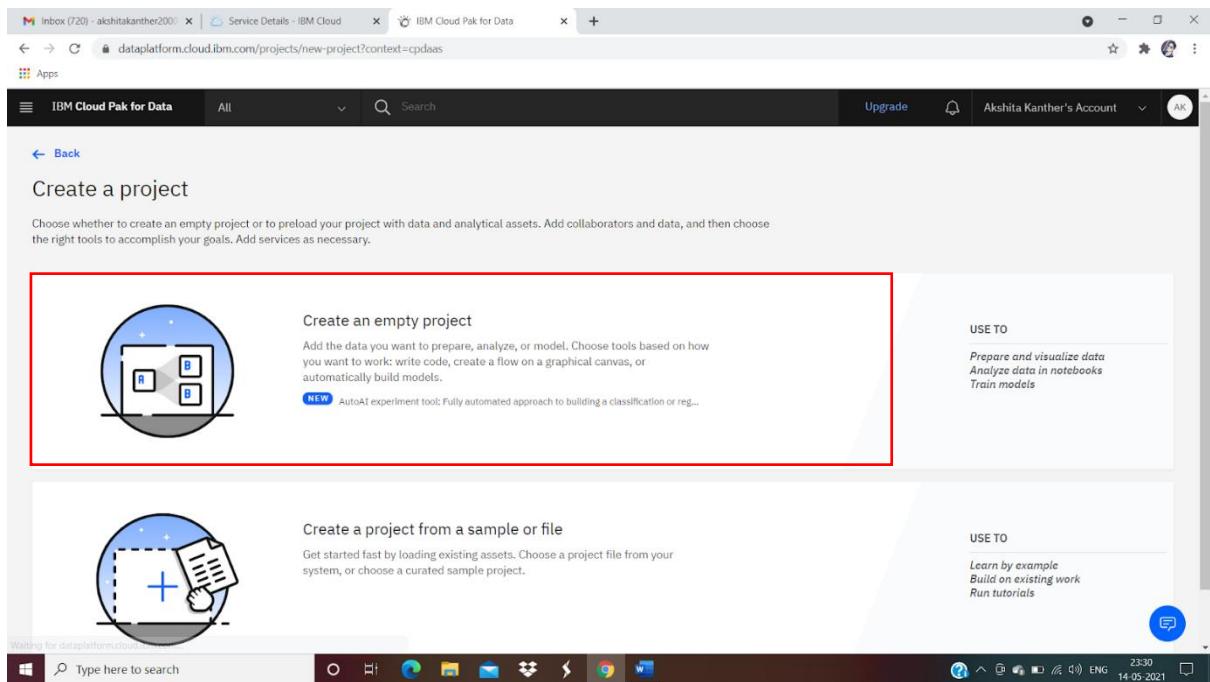
Step-3. Click on Get Started.



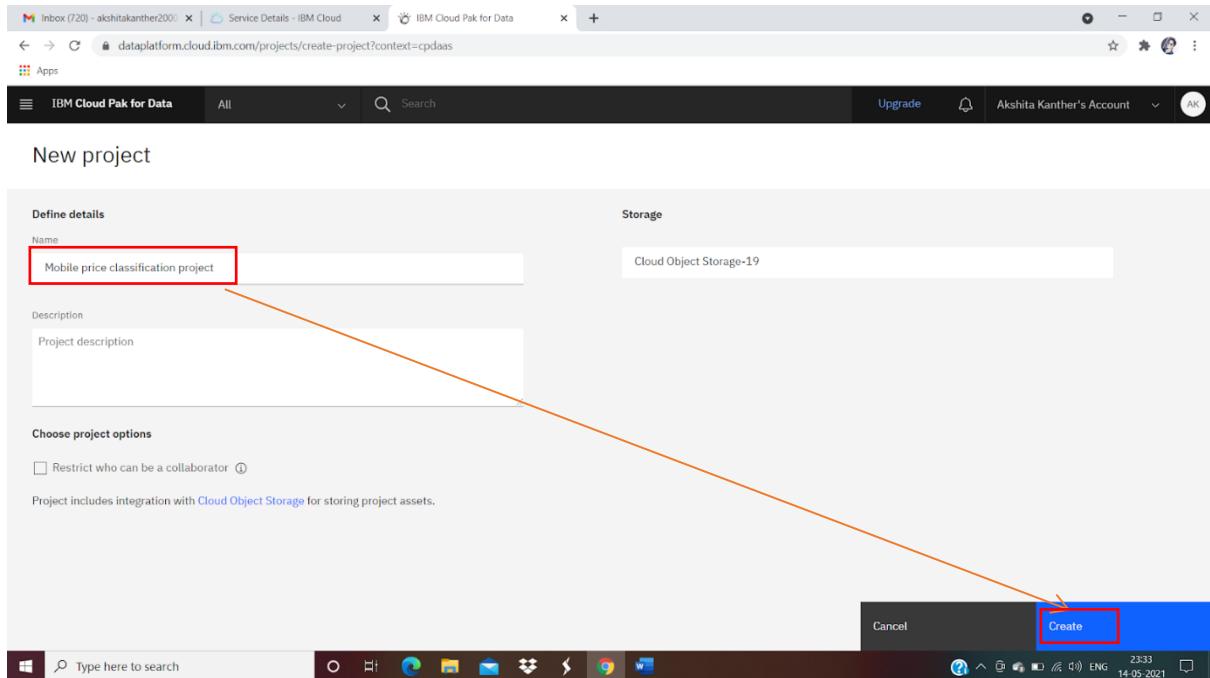
Step-4. Click on Create a project.



Step-5. Click on create an empty project.



Step-6. Enter your project name. Also ensure that an instance of Cloud Object Storage is created. Enter project description (optional). Then click on Create.



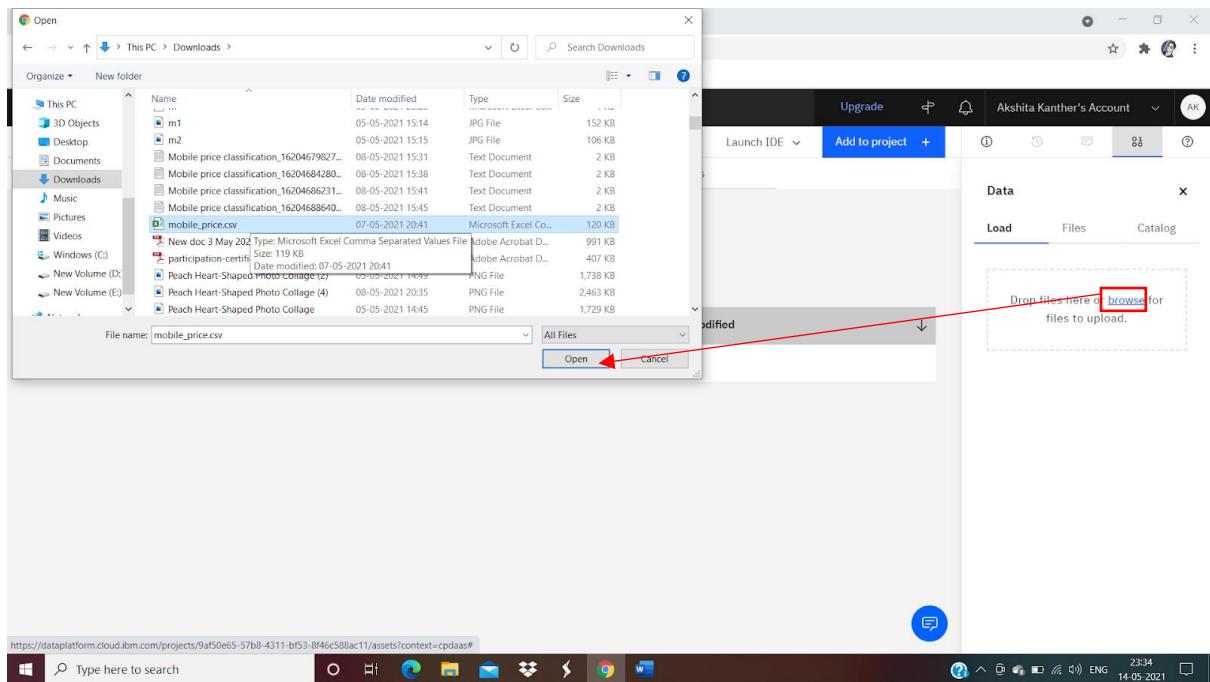
Step-7. Click on Assets.

The screenshot shows the 'IBM Cloud Pak for Data' interface. At the top, there are three tabs: 'Inbox (720) - akshitakanther2000', 'Service Details - IBM Cloud', and 'IBM Cloud Pak for Data'. Below the tabs, there's a search bar and a navigation bar with links for 'Upgrade', 'Launch IDE', 'Add to project', and account information ('Akshita Kanther's Account'). The main content area is titled 'Mobile price classification project'. It shows '0 Assets' and '1 Collaborators'. A red box highlights the 'Assets' tab in the navigation bar. The 'Assets' section contains an 'Overview' panel with details like 'Date created: May 14, 2021', 'Description: No description available', 'Storage: 0 Byte used (Cloud Object Storage)', and 'Collaborators: Akshita Kanther (Admin)'. To the right is a 'Recent activity' panel showing a small icon of a document with a plus sign and the text 'Alerts related to this project appear here when the project is active.' At the bottom, there's a 'Readme' section and a 'Back to top' button.

Step-8. Click on browse under load in the Data panel on the left side of the page.

This screenshot shows the same 'Assets' tab from the previous step. On the right side, there's a 'Data' panel with three tabs: 'Load' (which is highlighted with a red circle), 'Files', and 'Catalog'. A red arrow points to the 'browse' button in the 'Load' tab's interface, which says 'Drop files here or click browse to files to upload.' The rest of the interface is identical to the previous screenshot, including the navigation bar and the 'Mobile price classification project' overview.

Step-9. And upload your dataset in CSV format.



Step-10. Click on Overview to view that 1 Asset has been uploaded. Then switch to Assets.

A screenshot of the IBM Cloud Pak for Data interface. The top navigation bar shows 'Inbox (720) - akshitakanther2000', 'Service Details - IBM Cloud', and 'IBM Cloud Pak for Data'. The main area shows a project titled 'Mobile price classification project' last updated on May 14, 2021. The 'Overview' tab is active, indicated by a blue border. Other tabs include 'Assets', 'Environments', 'Jobs', 'Access Control', and 'Settings'. On the right, there are two large numbers: '1' under 'Assets' and '1' under 'Collaborators'. Below the tabs, there's an 'Overview' section with details like 'Date created: May 14, 2021', 'Description: No description available', 'Storage: 122.4 KB used (Cloud Object Storage)', and 'Collaborators: Akshita Kanther (Admin)'. There's also a 'Recent activity' section with a placeholder message: 'Alerts related to this project appear here when the project is active.' At the bottom, there's a 'Readme' section and a 'Back to top' button. The taskbar at the bottom shows various application icons and the date/time '14-05-2021 23:35'.

Step-11. Click on the name of csv file of dataset that has been uploaded.

IBM Cloud Pak for Data

Projects / Mobile price classification project

Overview Assets Environments Jobs Access Control Settings

What assets are you looking for?

Data assets

0 assets selected.

Name	Type	Created by	Last modified
CSV mobile_price.csv.csv	Data Asset	Akshita Kanther	May 14, 2021, 11:35 PM

Data

Load Files Catalog

Drop files here or [browse](#) for files to upload.

Stay on the page until upload completes.
Incomplete uploads are cancelled.

Step-12. Explore the dataset under preview.

IBM Cloud Pak for Data

Projects / Mobile price classification project / mobile_price.csv.csv

Preview Profile Activities

Schema: 21 Columns

Preview: First 1000 rows

Last refresh: just now

battery_p...	blue	clock_sp...	dual_sim	fc	four_g	int_mem...	m_dep	mobile_vt	n_cores
842	0	2.2	0	1	0	7	0.6	188	2
1021	1	0.5	1	0	1	53	0.7	136	3
563	1	0.5	1	2	1	41	0.9	145	5
615	1	2.5	0	0	0	10	0.8	131	6
1821	1	1.2	0	13	1	44	0.6	141	2
1859	0	0.5	1	3	0	22	0.7	164	1
1821	0	1.7	0	4	1	10	0.8	139	8
1954	0	0.5	1	0	0	24	0.8	187	4
1445	1	0.5	0	0	0	53	0.7	174	7
509	1	0.6	1	2	1	9	0.1	93	5
769	1	2.9	1	0	0	9	0.1	182	5
1520	1	2.2	0	5	1	33	0.5	177	8

Information

Data Asset
mobile_price.csv.csv

Description
No description is available for this asset.

Tags
No description is available for this asset.
Added: May 14, 2021, 11:35 PM
Size: 122.403 KB

Step-13. Scroll to the right side and view all the columns.

Screenshot of the IBM Cloud Pak for Data interface showing the 'mobile_price.csv.csv' data asset. The schema preview shows 21 columns with sample data for the first 1000 rows. A 'Refine' button is visible in the top right of the preview area.

Screenshot of the IBM Cloud Pak for Data interface showing the 'mobile_price.csv.csv_flow' data refinery flow. It displays three bar charts for 'n_cores', 'pc', and 'sc_w'. Below the charts, statistical details like Interquartile Range, Minimum, Maximum, Median, and Standard Deviation are listed for each column. On the right, the 'Information' panel shows the flow's name, steps, and output location.

Step-14. To use Data refinery tools, Click on Refine.

Schema: 21 Columns

px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_sc...	wifi	price_ran...
20	756	2549	9	7	19	0	0	1	1
905	1988	2631	17	3	7	1	1	0	2
1263	1716	2603	11	2	9	1	1	0	2
1216	1786	2769	16	8	11	1	0	0	2
1208	1212	1411	8	2	15	1	1	0	1
1004	1654	1067	17	1	10	1	0	0	1
381	1018	3220	13	8	18	1	0	1	3
512	1149	700	16	3	5	1	1	1	0
386	836	1099	17	1	20	1	0	0	0
1137	1224	513	19	10	12	1	0	0	0
248	874	3946	5	2	7	0	0	0	3
151	1005	3826	14	9	13	1	1	1	3

Last refresh: 20 seconds ago

Information

Data Asset mobile_price.csv.csv

Description No description is available for this asset.

Tags No description is available for this asset.

Added: May 14, 2021, 11:35 PM

Size: 122.403 KB

Step-15. You will find three tabs: Data, Profile and Visualizations. Click on Data.

Previewing the first 50 rows
Reading and processing data sample...

battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory
1	842	0	2.2	0	1	0
2	1021	1	0.5	1	0	1
3	563	1	0.5	1	2	1
4	615	1	2.5	0	0	0
5	1821	1	1.2	0	13	1
6	1859	0	0.5	1	3	0
7	1821	0	1.7	0	4	1
8	1954	0	0.5	1	0	0
9	1445	1	0.5	0	0	0
10	509	1	0.6	1	2	1
11	769	1	2.9	1	0	0
12	1520	1	2.2	0	5	1
13	1815	0	2.0	0	?	0

SOURCE FILE: mobile_price.csv.csv SAMPLE SIZE: First 50 rows

Step-16. Scroll to the right side and select the talk_time column, click on the three dots and Remove it as it is not required.

The screenshot shows the IBM Cloud Pak for Data interface. In the main workspace, a data table is displayed with columns: screen_height, screen_width, talk_time, three_g, and touch_screer. A context menu is open over the 'talk_time' column header, with options like 'Remove duplicates', 'Remove empty rows', 'Sort ascending', 'Sort descending', 'Substitute', and 'CONVERT COLUMN...' visible. The 'Information' panel on the right shows 7 steps, details about the project ('Mobile price classification project'), and a data refinery flow named 'mobile_price.csv.csv_flow'. The flow description is 'Enter a description of the Data Refinery flow'.

Step-17. Double click the header of blue column and rename it as Bluetooth, repeat the same for fc to rename it as front camera.

The screenshot shows the IBM Cloud Pak for Data interface. The data table now has columns: battery_power, bluetooth, clock_speed, dual_sim, and front_camera. The 'bluetooth' and 'front_camera' headers are highlighted with red boxes. The 'Information' panel on the right shows 8 steps, details about the project ('Mobile price classification project'), and a data refinery flow named 'mobile_price.csv.csv_flow'. The flow description is 'Enter a description of the Data Refinery flow'.

Step-18. Double click the header of n_cores column and rename it as no. of cores.

The screenshot shows the IBM Cloud Pak for Data interface. On the left, there's a data preview table with columns: battery_power, bluetooth, clock_speed, dual_sim, front_camera, and n_cores. A context menu is open over the 'n_cores' column header, listing 8 steps: Rename column, Renamed column blue to bluetooth, Rename column, Renamed column fc to front camera megapixel, Rename column, Renamed column front camera megapixel to front camera, Rename column, and Renamed column n_cores to no. of cores. To the right, the 'Information' pane displays the flow details: LOCATION (Mobile price classification project), DATA REFINERY FLOW NAME (mobile_price.csv.csv_flow), and STEPS (8). The bottom right corner shows the date and time: 14-05-2021 23:44.

Step-19. Double click the header of sc_w column and rename it as screen_width. Then click on save icon on the top, then click on Edit on the right side under Details in Information pane.

The screenshot shows the IBM Cloud Pak for Data interface. A context menu is open over the 'sc_w' column header, listing 8 steps: Rename column, Renamed column n_cores to no. of cores, Rename column, Renamed column sc_h to screen_height, Rename column, Renamed column sc_w to screen_width, Remove (JUST ADDED), and Removed talk_time. Above the table, there's a 'Save' button. To the right, the 'Information' pane displays the flow details: LOCATION (Mobile price classification project), DATA REFINERY FLOW NAME (mobile_price.csv.csv_flow), and STEPS (8). Red arrows point from the 'Save' button to the 'Edit' button in the 'Information' pane. The bottom right corner shows the date and time: 14-05-2021 23:45.

Step-20. Click on Edit Output.

The screenshot shows the 'IBM Cloud Pak for Data' interface. On the left, under 'DATA REFINERY FLOW DETAILS', it lists 'LOCATION' as 'Mobile price classification project', 'DATA REFINERY FLOW NAME' as 'mobile_price.csv.csv_flow', and 'STEPS' as '8'. On the right, under 'DATA REFINERY FLOW OUTPUT', there is a modal dialog titled 'Edit Output'. Inside the dialog, the 'Location' field shows 'Mobile price classification project/Data ...'. The 'Data set name' field contains 'mobile_price_csv_shaped'. Below these fields is a checkbox with the text: 'If the data set already exists, overwrite the data in the existing data set with the Data Refinery flow output.' At the bottom of the dialog is a 'Done' button.

Step-21. Make the changes and click on Done and then click on save icon and save .

The screenshot shows the 'IBM Cloud Pak for Data' interface. The 'Edit Output' dialog box is open, showing the 'Edit output' button circled in red. A red arrow points from this button to the 'Done' button at the bottom right of the dialog. The 'Data set name' field in the dialog contains 'mobile_price_csv_shaped'. The rest of the interface is similar to the previous screenshot, showing 'DATA REFINERY FLOW DETAILS' and 'DATA REFINERY FLOW OUTPUT' sections.

DATA REFINERY FLOW DETAILS

LOCATION
Mobile price classification project

DATA REFINERY FLOW NAME
mobile_price.csv.csv_flow

Enter a description of the Data Refinery flow

STEPS
8

DATA REFINERY FLOW OUTPUT

Edit Output

Location
Mobile price classification project/Data ...

Data set name
mobile_price_csv_shaped

Enter a description of the resulting data set.

If the data set already exists, overwrite the data in the existing data set with the Data Refinery flow output.

Review the Data Refinery flow details and the Data Refinery flow output details.

Done

Operation + Code an operation to cleanse and shape your data

Data Profile Visualizations

	battery_power	bluetooth	clock_speed	dual_sim	front_camera
	Integer	Integer	Decimal	Integer	Integer
1	842	0	2.2	0	1
2	1021	1	0.5	1	0
3	563	1	0.5	1	2
4	615	1	2.5	0	0
5	1821	1	1.2	0	13
6	1859	0	0.5	1	3
7	1821	0	1.7	0	4
8	1954	0	0.5	1	0
9	1445	1	0.5	0	0
10	509	1	0.6	1	2
11	769	1	2.9	1	0
12	1520	1	2.2	0	5
13	1915	0	2.9	0	?

SOURCE FILE: mobile_price.csv.csv FULL DATA SET: 2000 rows

Steps

8 Steps

Data Source
mobile_price.csv.csv

Convert column type AUTOMATIC

Automatically converted one or more columns to inferred data types. Strings that are converted to decimal use a dot (.) for the decimal symbol.

Rename column
Renamed column blue to bluetooth

Rename column
Renamed column fc to front camera megapixel

LOCATION
Mobile price classification project

DATA REFINERY FLOW NAME
mobile_price.csv.csv_flow

Enter a description of the Data Refinery flow

STEPS
8

DATA REFINERY FLOW OUTPUT

Location
Mobile price classification project/...

Data set name
mobile_price_csv_shaped

Step-22. Go to Assets and click on newly created Data refinery flow file under Data Refinery Flows.

The screenshot shows the 'Assets' tab in the IBM Cloud Pak for Data interface. The 'Data assets' section lists a single CSV file named 'mobile_price.csv.csv' created by Akshita Kanther on May 14, 2021, at 11:35 PM. The 'Data Refinery flows' section lists a single flow named 'mobile_price.csv.csv_flow' also created by Akshita Kanther on May 14, 2021, at 11:49 PM. A red box highlights the 'mobile_price.csv.csv_flow' entry.

After Refining

Data Refinery flow:-

The screenshot shows the 'Previewing' view of the refined Data Refinery flow. The interface displays the input file 'mobile_price.csv.csv' and the output file 'mobile_price_csv_shaped'. The right panel provides detailed information about the flow, including its location ('Mobile price classification project'), name ('mobile_price.csv.csv_flow'), description ('Enter a description of the Data Refinery flow'), steps (8), and output details.

The screenshot shows the IBM Cloud Pak for Data interface. The main area displays a table of data from a CSV file named 'mobile_price.csv.csv'. The table has columns: battery_power, bluetooth, clock_speed, dual_sim, front_camera, four_g, and int_memory. The data consists of 13 rows. The 'Information' panel on the right provides details about the project, including its name, location, and configuration.

	battery_power	bluetooth	clock_speed	dual_sim	front_camera	four_g	int_memory
1	842	0	2.2	0	1	0	7
2	1021	1	0.5	1	0	1	53
3	563	1	0.5	1	2	1	41
4	615	1	2.5	0	0	0	10
5	1821	1	1.2	0	13	1	44
6	1859	0	0.5	1	3	0	22
7	1821	0	1.7	0	4	1	10
8	1954	0	0.5	1	0	0	24
9	1445	1	0.5	0	0	0	53
10	509	1	0.6	1	2	1	9
11	769	1	2.9	1	0	0	9
12	1520	1	2.2	0	5	1	33
13	1015	0	2.0	0	2	0	32

DATA VISUALIZATION-

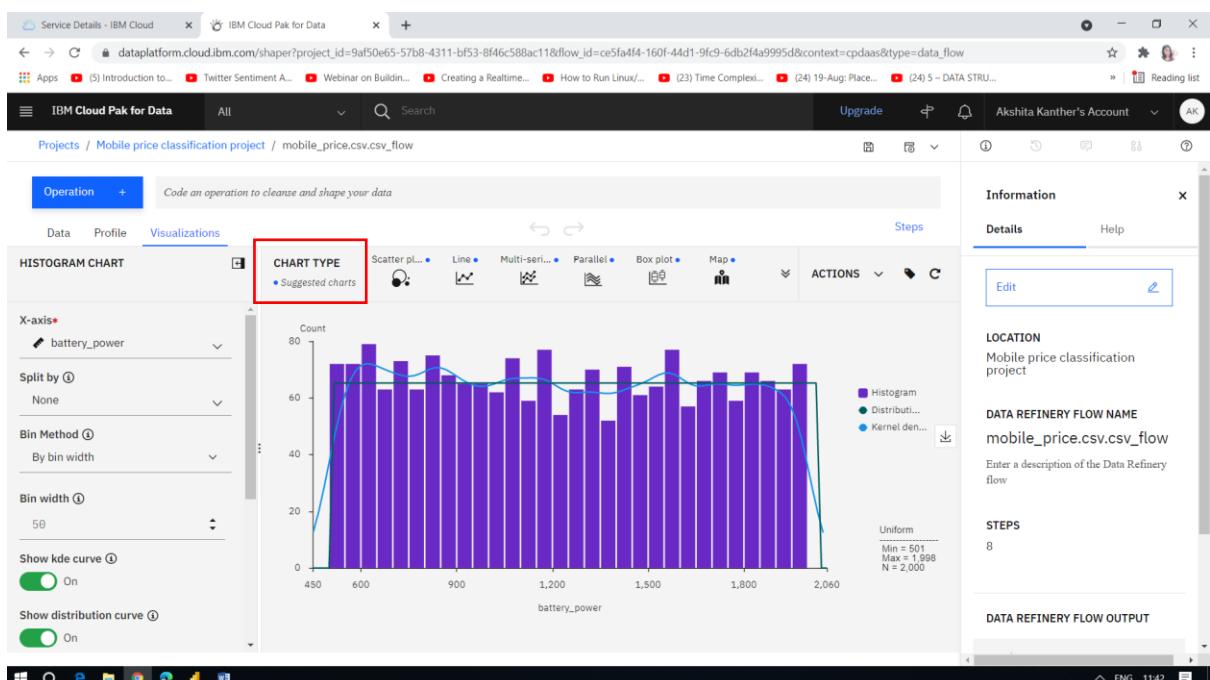
Data visualization is the representation of data through use of common graphics, such as charts, plots, infographics, and even animations. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand.

Step-1. Click on visualizations tab.

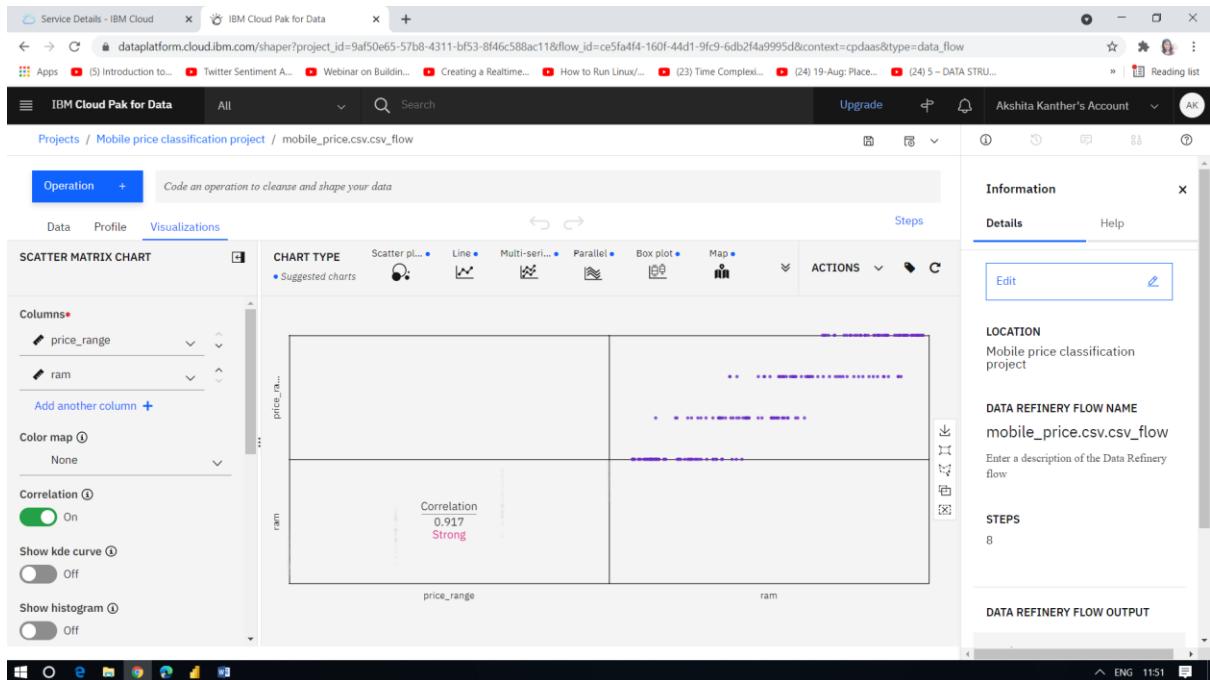
The screenshot shows the IBM Cloud Pak for Data interface. On the left, there's a data preview of a CSV file named 'mobile_price.csv.csv_flow' with 2000 rows. The columns include battery_power, bluetooth, clock_speed, dual_sim, front_camera, four_g, and int_memory. On the right, there's an 'Information' panel with tabs for 'Details' and 'Help'. Under 'Details', there are sections for 'LOCATION' (Mobile price classification project), 'DATA REFINERY FLOW NAME' (mobile_price.csv.csv_flow), and 'STEPS' (8). The 'Edit' button is highlighted.

Step-2. Select Histogram chart, select battery_power for x-axis and bin method as Bin width.

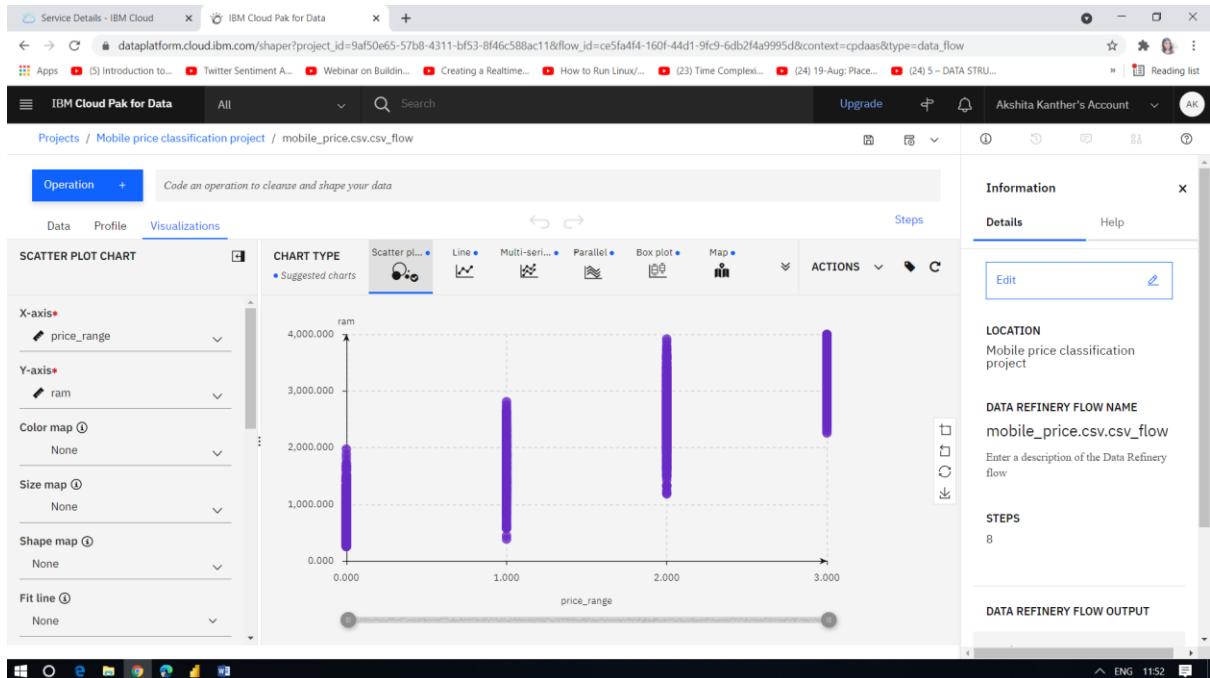
View the distribution for battery_power column.



Step-3. Select a SCATTER MATRIX CHART from the chart type and select price_range and ram as columns. We can observe strong correlation between them.



Step-4. Make a SCATTER PLOT CHART from chart type, select price_range for x-axis and select ram for y-axis.



VISUAL RECOGNITION SERVICE-

The IBM Watson Visual Recognition service uses deep learning algorithms to analyse images for scenes, objects, and other content. The response includes keywords that provide information about the content.

Available models

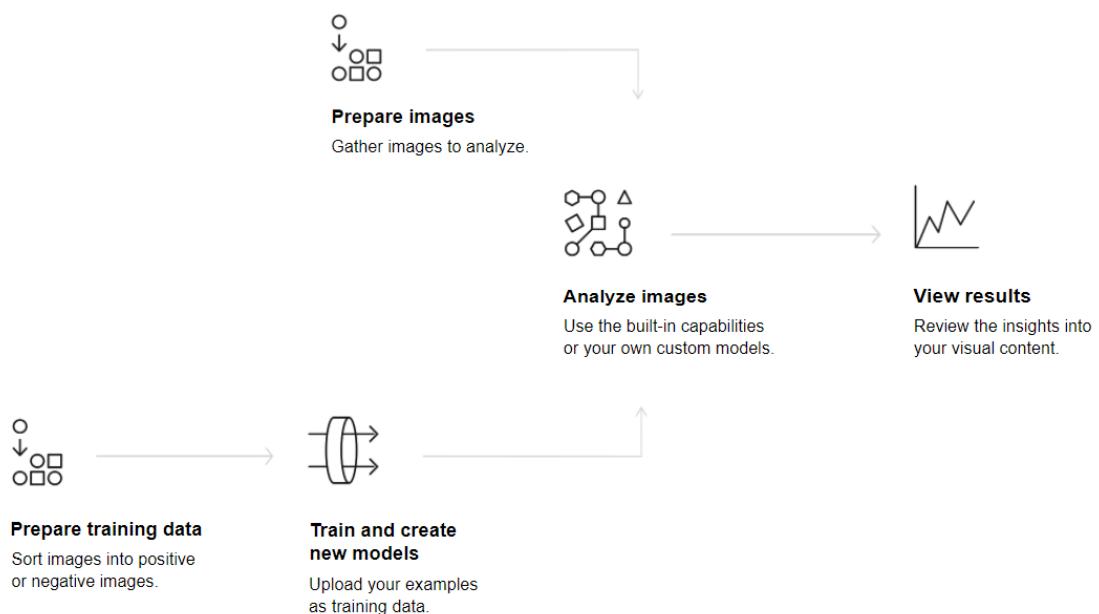
A set of built-in models provides highly accurate results without training:

- **General** model: Default classification from thousands of classes.
- **Explicit** model: Whether an image is inappropriate for general use.
- **Food** model: Specifically for images of food items.

We can also train **custom models** to create specialized classes.

How to use the service-

The following image shows the process of creating and using Visual Recognition:



Use cases

The Visual Recognition service can be used for diverse applications and industries, such as:

- **Manufacturing:** Use images from a manufacturing setting to make sure that products are positioned correctly on an assembly line
- **Visual auditing:** Look for visual compliance or deterioration in a fleet of trucks, planes, or windmills out in the field, train custom models to understand what defects look like
- **Insurance:** Rapidly process claims by using images to classify claims into different categories
- **Social listening:** Use images from your product line or your logo to track buzz about your company on social media
- **Social commerce:** Use an image of a plated dish to find out which restaurant serves it and find reviews, use a travel photo to find vacation suggestions based on similar experiences
- **Retail:** Take a photo of a favourite outfit to find stores with those clothes in stock or on sale, use a travel image to find retail suggestions in that area
- **Education:** Create image-based applications to educate about taxonomies

Step-1. Login to the IBM cloud platform, go to Resource list then click on Services and then click on instance of Visual Recognition Service. If an instance is not created previously then on the search bar of Search resources and offerings type visual recognition service and then create a visual recognition service instance.

The screenshot shows the IBM Cloud Resource list. On the left, there's a sidebar with categories like Devices, VPC infrastructure, Clusters, Satellite, Cloud Foundry apps, Cloud Foundry services, Services (4), and Storage. Under Services, 'Visual Recognition-5m' is listed. This item is highlighted with a red box. The main table has columns for Name, Group, Location, Product, Status, and Tags. The 'Visual Recognition-5m' row shows it's in the Default group, located in Dallas, part of Watson Knowledge Catalog, and active. A tag 'cpdaas' is attached to it.

Step-2. Click on Launch Watson Studio in Manage section.

The screenshot shows the IBM Watson Service Page for 'Visual Recognition-5m'. The top navigation bar includes Catalog, Docs, Support, Manage, and Akshita Kanthe. The main content area has tabs for Details and Actions. On the left, there's a 'Manage' sidebar with Service credentials, Plan, and Connections. The central area starts with a tutorial link 'Start by viewing the tutorial' and a 'Launch Watson Studio' button, which is also highlighted with a red box. Below that is a 'Credentials' section with an API key field and a URL field containing the value 'https://api.us-south.visual-recognition.watson.cloud.ibm.com/instances'. To the right, there's a 'Plan' section showing 'Lite' and an 'Upgrade' button.

Step-3. We have options to create two types of models, one is Custom Models and another one is Prebuilt Models. We will be working on both types of models.

The screenshot shows the IBM Cloud Pak for Data interface with the Visual Recognition service selected. The 'Custom Models' section is at the top, followed by the 'Prebuilt Models' section. Under 'Prebuilt Models', there are three categories: 'General', 'Food', and 'Explicit'. Each category has a 'Copy classifier ID' button and a 'Create Model +' button. Below each category is a brief description. A red box highlights the 'General' category, and a red arrow points to the 'Test' button located at the bottom of the 'General' section.

Working on Prebuilt models

Step-4. Go to General under Prebuilt Models and click on Test.

The screenshot shows the same interface as the previous one, but with a red box highlighting the 'Prebuilt Models' section. A red arrow points from this box to the 'Test' button located at the bottom of the 'General' category under 'Prebuilt Models'.

Step-5. Upload the images to Test and output will be displayed on the basis of multiple prebuilt classes.

General
Associated Service : Visual Recognition-5m

Prebuilt classes which are already present in the prebuilt models

Classes	Threshold
angle bracket	0.0
charcoal color	0.77
coal black color	0.65
controller	0.57
dado (dice)	0.50
doorbell	0.55
electric switch	0.63
electronic device	0.67
electronic equipment	0.59
flat panel display	0.50

Test

Implementation

Image	Label	Score
	charcoal color	0.82
	sage green color	0.94
	purple color	0.98
	controller	0.65
	electric switch	0.63
	push button	0.61
	doorbell	0.55
	angle bracket	0.54
	wall bracket	0.54
	support	0.54
	dado (dice)	0.50

According to these Prebuilt classes the results are shown here in the given fig.

Working on Custom models

Step-6. Go to Classify Images in Custom Model and click on Create Model + .

Services / Watson Services / Visual Recognition-5m

Custom Models

Classify Images
Create customized visual classifiers that go beyond the built-in images classes provided with the Watson Studio Visual Recognition tool.

Create Model +

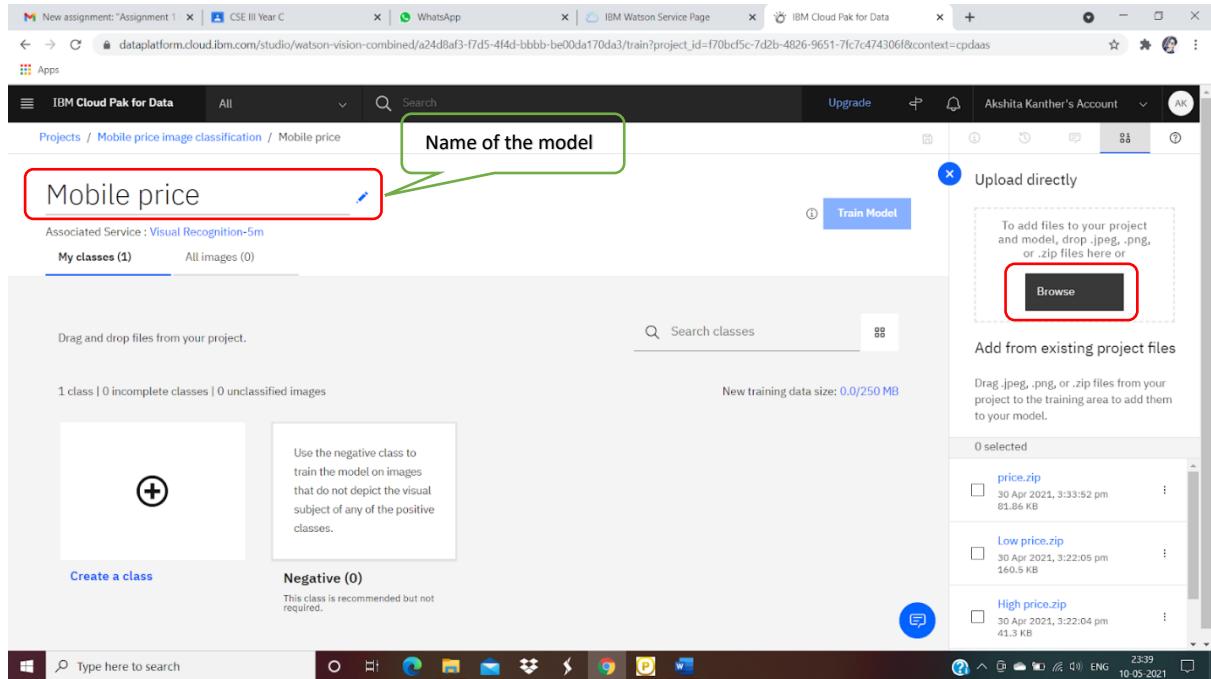
Detect Objects
Build custom image classifiers that detect objects within images using coordinates.

Create Model +

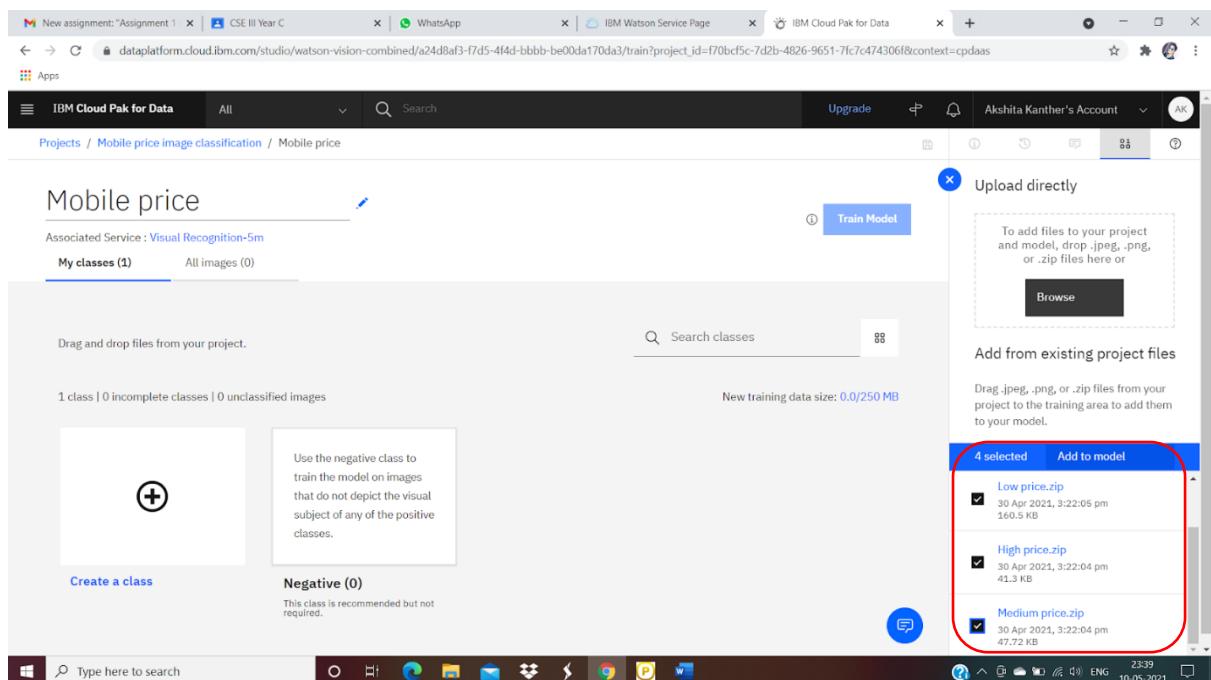
Prebuilt Models

- General**
Copy classifier ID
- Food**
Copy classifier ID
- Explicit**
Copy classifier ID

Step-7. Go to the Upload directly section on the right and then click on Browse to upload image folders or zip files of images of different classes: High price, low price and medium price and also we can name the model.

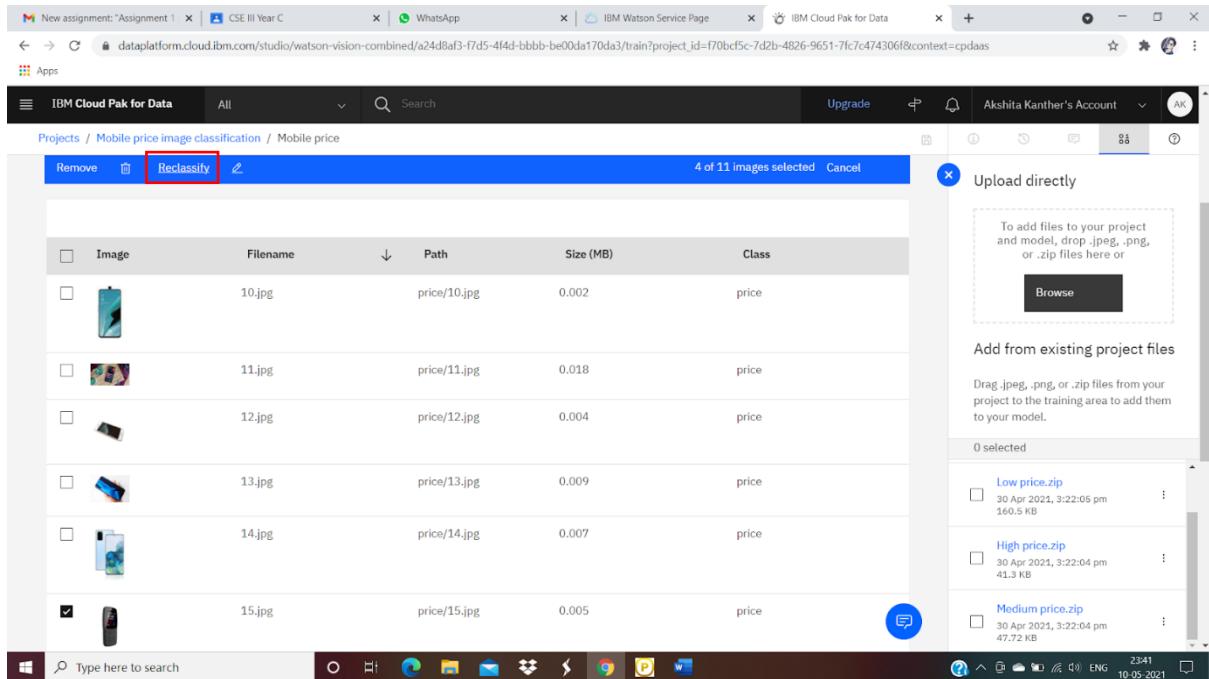


Step-8. Select all the image zip folders and then click on Add to model.



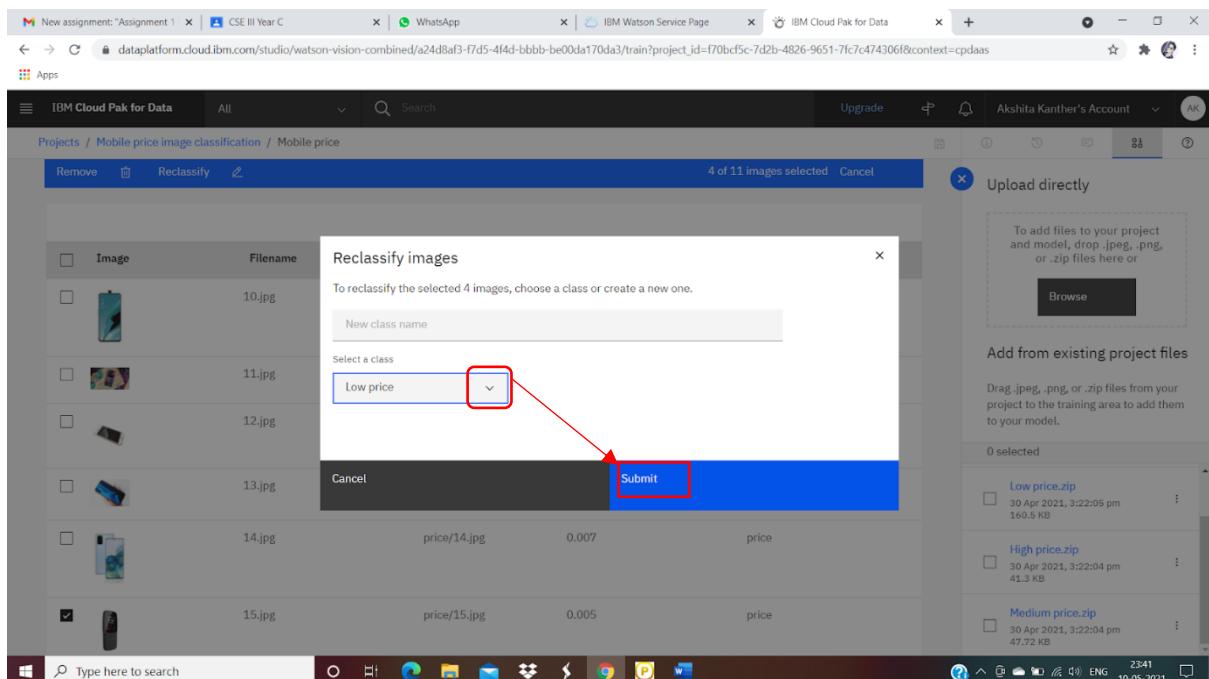
Reclassifying the images –

Step-9. Reclassify the images if they are in incorrect classes. Open each folder then select the images which shouldn't be present in that class, and then click on Reclassify on the top.



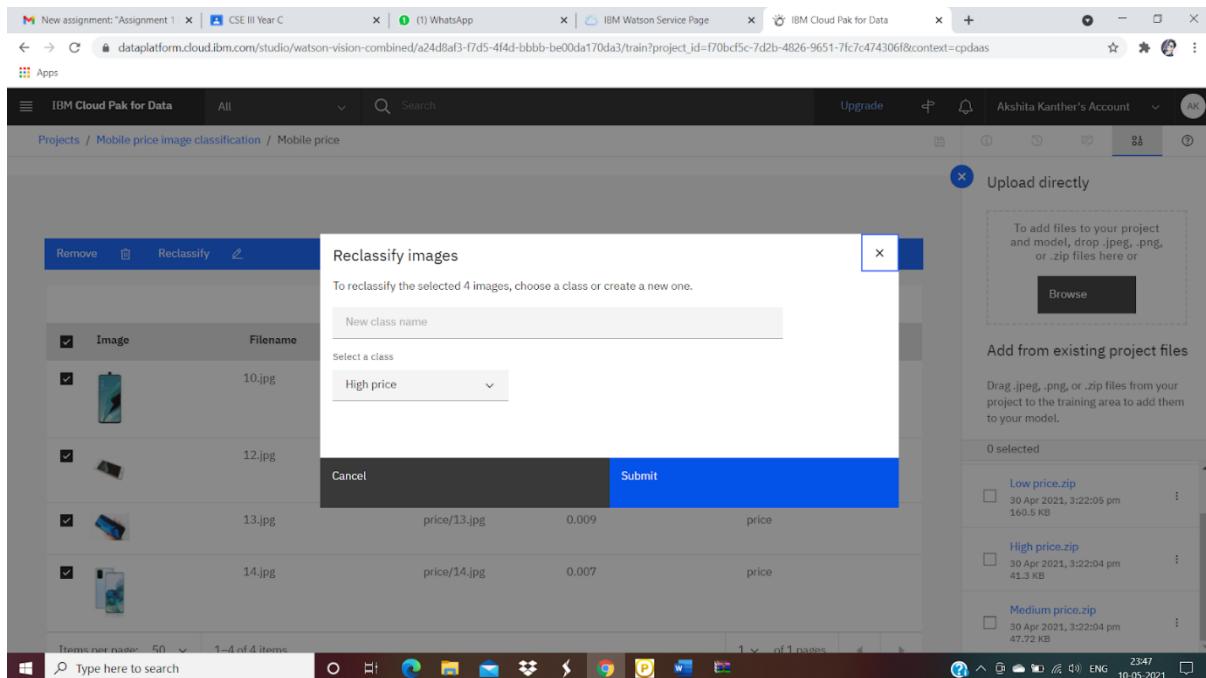
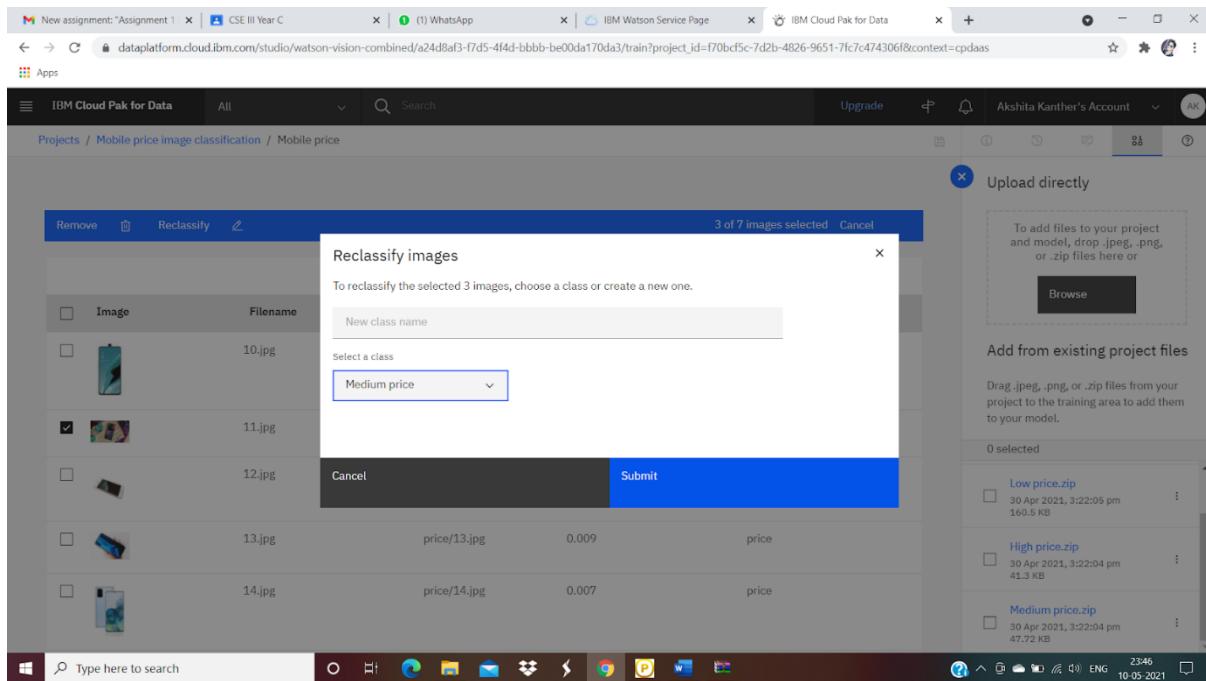
The screenshot shows the IBM Cloud Pak for Data interface. In the top navigation bar, there are several tabs: 'New assignment: "Assignment 1"', 'CSE III Year C', 'WhatsApp', 'IBM Watson Service Page', and 'IBM Cloud Pak for Data'. Below the tabs, there's a search bar and a user account section for 'Akshita Kanther's Account'. The main area is titled 'Mobile price image classification / Mobile price'. A blue header bar at the top of the list table contains the buttons 'Remove', 'Recategorize' (which is highlighted with a red box), and 'Cancel'. To the right of the table, there's a sidebar with sections for 'Upload directly' (with a 'Browse' button) and 'Add from existing project files' (listing 'Low price.zip', 'High price.zip', and 'Medium price.zip'). The main table lists 11 images with their details: Image, Filename, Path, Size (MB), and Class. The 15.jpg image is checked and has its class set to 'price'. The 15.jpg row also has a blue circular icon with a white question mark. The bottom of the screen shows a taskbar with various icons and a system tray indicating the date and time as '10-05-2021 23:41'.

Step-10. A dialog box will pop up. Select an appropriate class for those selected items and then click on Submit.

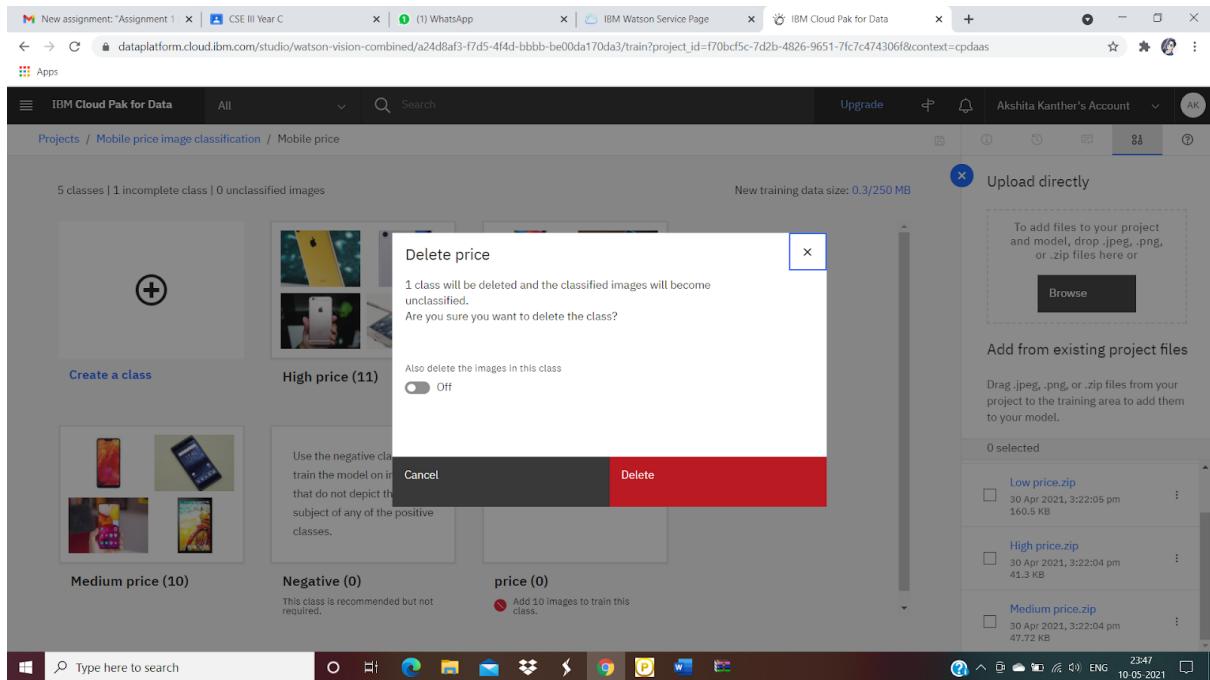


This screenshot shows the 'Reclassify images' dialog box overlaid on the main interface. The dialog has a title 'Reclassify images' and instructions: 'To reclassify the selected 4 images, choose a class or create a new one.' It includes a 'New class name' input field and a 'Select a class' dropdown menu. The 'Low price' option is selected in the dropdown and highlighted with a red box. A red arrow points from this highlighted dropdown to the 'Submit' button at the bottom of the dialog, which is also highlighted with a red box. The background shows the same interface as the previous screenshot, with the 'Mobile price' project and its files visible.

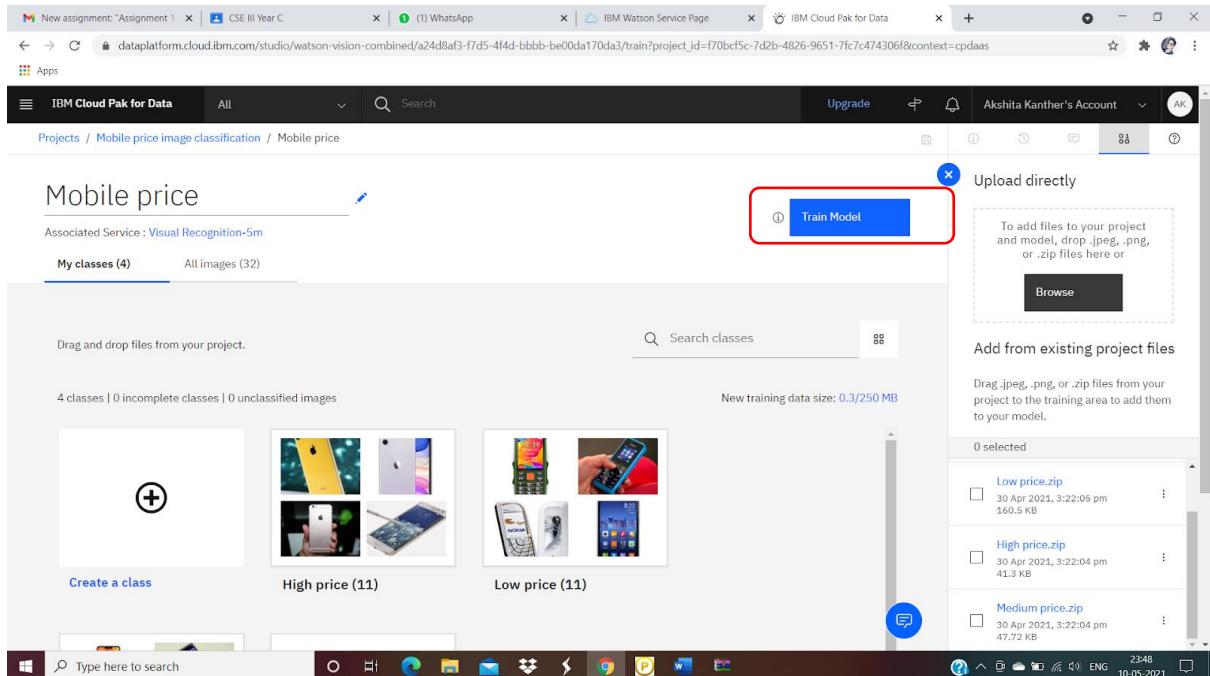
Step-11. Repeat the same for other classes also.

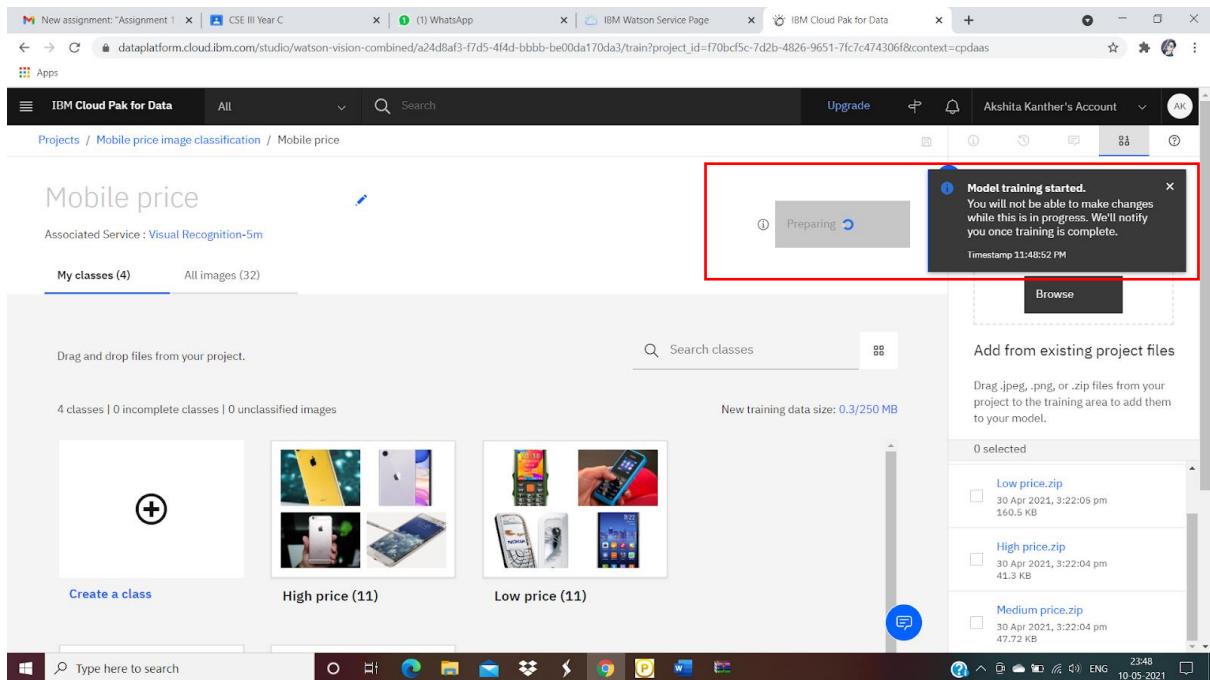
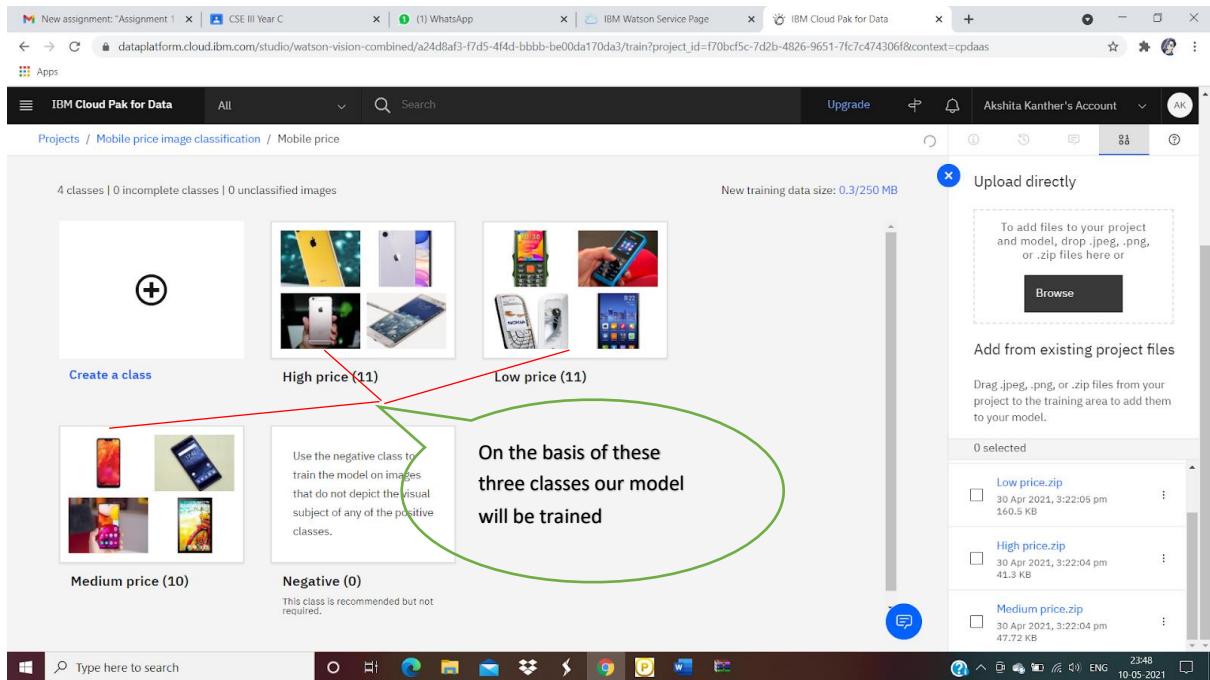


Step-12. Delete the unwanted class which has no images.

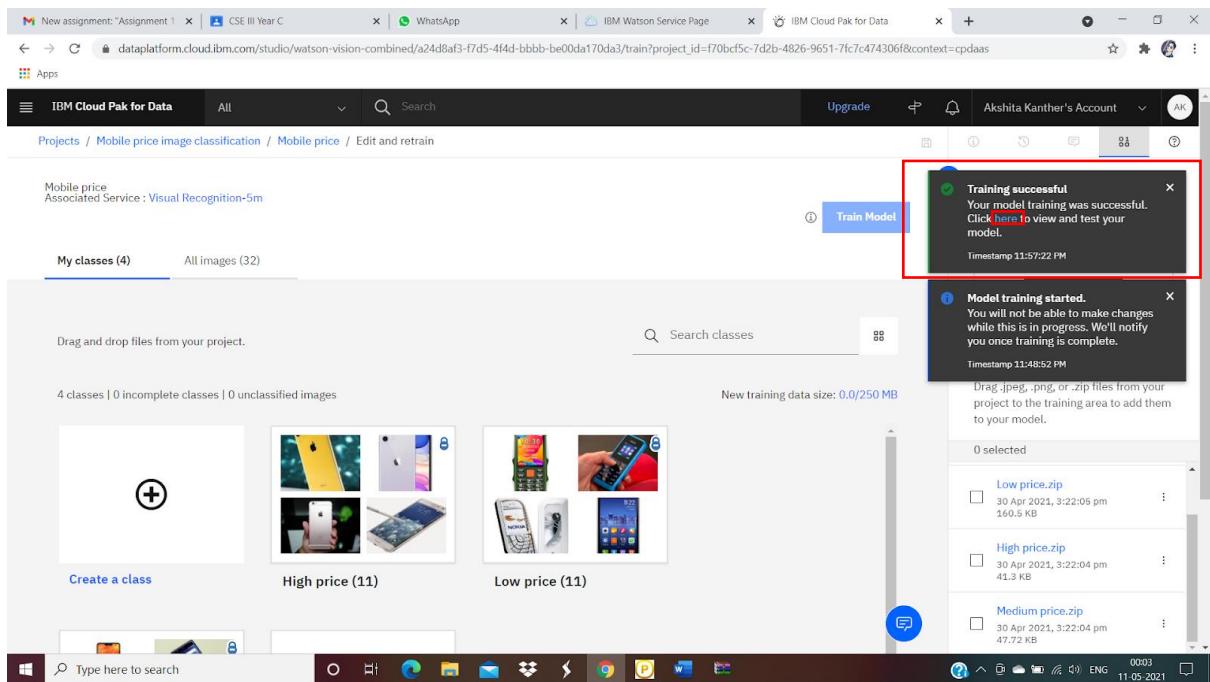


Step-13. Click on Train Model.





Step-14. After the Training has been completed successfully, a dialog box will pop up containing Training successful message, click on here.



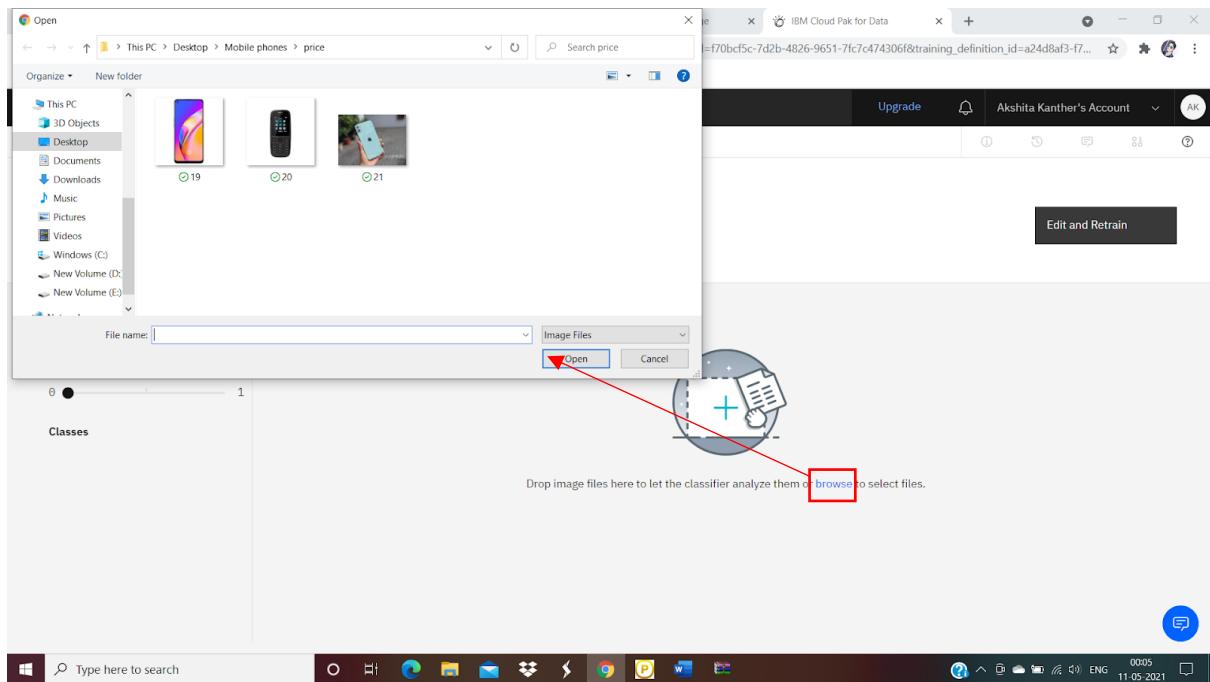
Step-15. Go to Test.

The screenshot shows the IBM Cloud Pak for Data interface with the "Test" tab highlighted in red. The main area displays a "Summary" table with the following data:

Model ID	Mobileprice_1934536116
Status	Ready
Explanation	This model is ready for use.
Created on	5/10/2021, 11:49:11 PM
Updated on	5/10/2021, 11:49:11 PM
Number of classes	3
Number of images	32

Below the summary, there's a "Classes" section. The bottom status bar shows the date and time as "11-05-2021 00:04".

Step-16. Upload the sample test images for different classes: select the images from local folder of your pc and then click on open.



The screenshot shows the 'Test' tab of the IBM Cloud Pak for Data interface. A yellow callout box highlights the text: 'The results are shown on the basis of classes which are created or uploaded by us.' A red box highlights the classification results for images 19.jpg, 20.jfif, and 21.jfif.

Image	Class	Probability
19.jpg	Medium price	0.65
19.jpg	Low price	0.36
19.jpg	High price	0.36
20.jfif	Low price	0.91
20.jfif	High price	0.00
20.jfif	Medium price	0.00
21.jfif	High price	0.90
21.jfif	Medium price	0.03
21.jfif	Low price	0.01

The output is displayed on the basis of different classes we have created. We can observe that our model has detected images correctly.

MACHINE LEARNING MODEL (AutoAI experiment)-

The AutoAI graphical tool in Watson Studio automatically analyses our data and generates candidate model pipelines customized for our predictive modelling problem. These model pipelines are created iteratively as AutoAI analyses your dataset and discovers data transformations, algorithms, and parameter settings that work best for our problem setting. Results are displayed on a leader board, showing the automatically generated model pipelines ranked according to our problem optimization objective.

- **Required service**

Watson Machine Learning service

- **Data format**

Tabular: CSV files, with comma (,) delimiter

- **Data size**

Less than 1 GB

- **AutoAI process**

Using AutoAI, we can build and deploy a machine learning model with sophisticated training features and no coding. The tool does most of the work for us.

AutoAI automatically runs the following tasks to build and evaluate candidate model pipelines:

- **Data pre-processing**

Most data sets contain different data formats and missing values, but standard machine learning algorithms work with numbers and no missing values. AutoAI applies various algorithms, or estimators, to analyse, clean, and prepare your raw data for machine learning. It automatically detects and categorizes features based on data type, such as categorical or numerical. Depending on the categorization, it uses hyper-parameter optimization to determine the best combination of strategies for missing value imputation, feature encoding, and feature scaling for your data.

- **Automated model selection**

The next step is automated model selection that matches your data. AutoAI uses a novel approach that enables testing and ranking candidate algorithms against small subsets of the data, gradually increasing the size of the subset for the most promising algorithms to arrive at the best match. This approach saves time without sacrificing performance. It enables ranking a large number of candidate algorithms and selecting the best match for the data.

- **Automated feature engineering**

Feature engineering attempts to transform the raw data into the combination of features that best represents the problem to achieve the most accurate prediction. AutoAI uses a unique approach that explores various feature construction choices in a structured, non-exhaustive manner, while progressively maximizing model accuracy using reinforcement learning. This results in an optimized sequence of transformations for the data that best match the algorithms of the model selection step.

- **Hyper parameter optimization**

Finally, a hyper-parameter optimization step refines the best performing model pipelines. AutoAI uses a novel hyper-parameter optimization algorithm optimized for costly function evaluations such as model training and scoring that are typical in machine learning. This approach enables fast convergence to a good solution despite long evaluation times of each iteration.

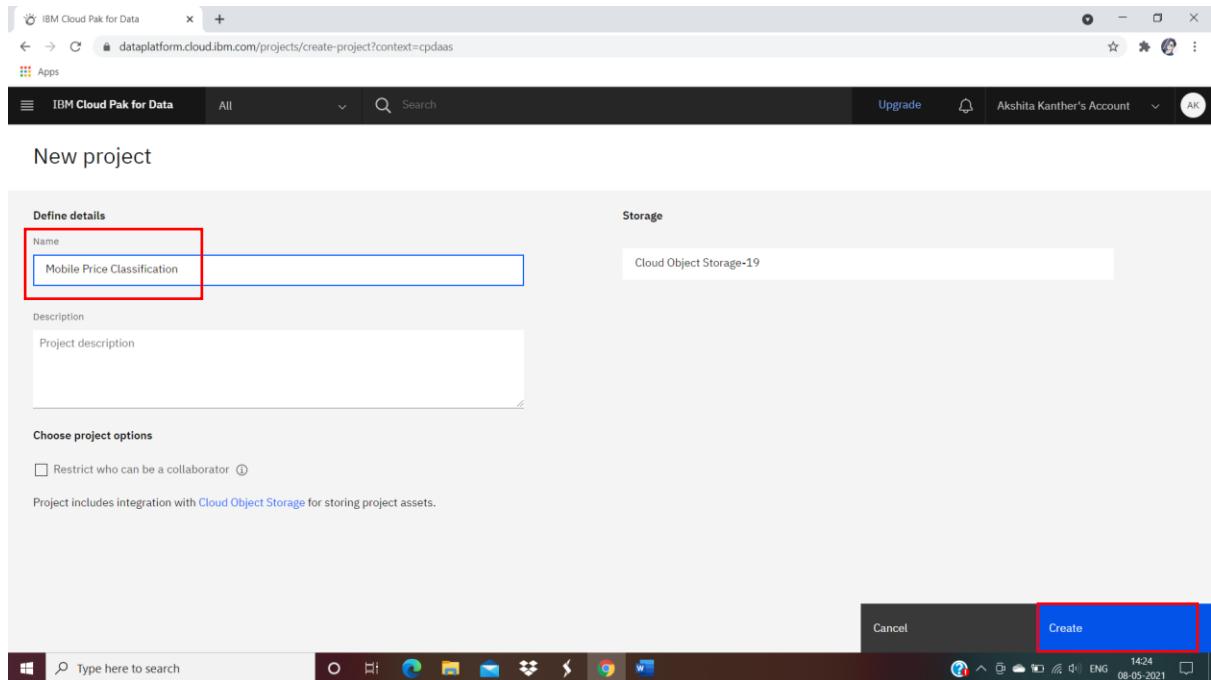
Step-1. Login to IBM cloud platform and **launch IBM Watson** service from resources (if IBM Watson service instance is not created then we will first create it and then launch it). Click on **Create a project**.

The screenshot shows the IBM Cloud Pak for Data homepage. At the top, there's a navigation bar with 'IBM Cloud Pak for Data', a search bar, and user account information. Below the header, there are three main sections: 'Learn by example', 'Work with data', and 'Extend your capabilities'. The 'Work with data' section contains a 'Create a project' button, which is highlighted with a red box. To the right of these sections is a decorative graphic of a 3D cube cluster with a magnifying glass over it. The left sidebar includes links for 'Quick navigation' (Projects, Catalogs, Deployments), 'Support' (Documentation, FAQ), and system status (Windows taskbar). The bottom of the screen shows a Windows taskbar with various icons.

Step-2. Click on create an empty project.

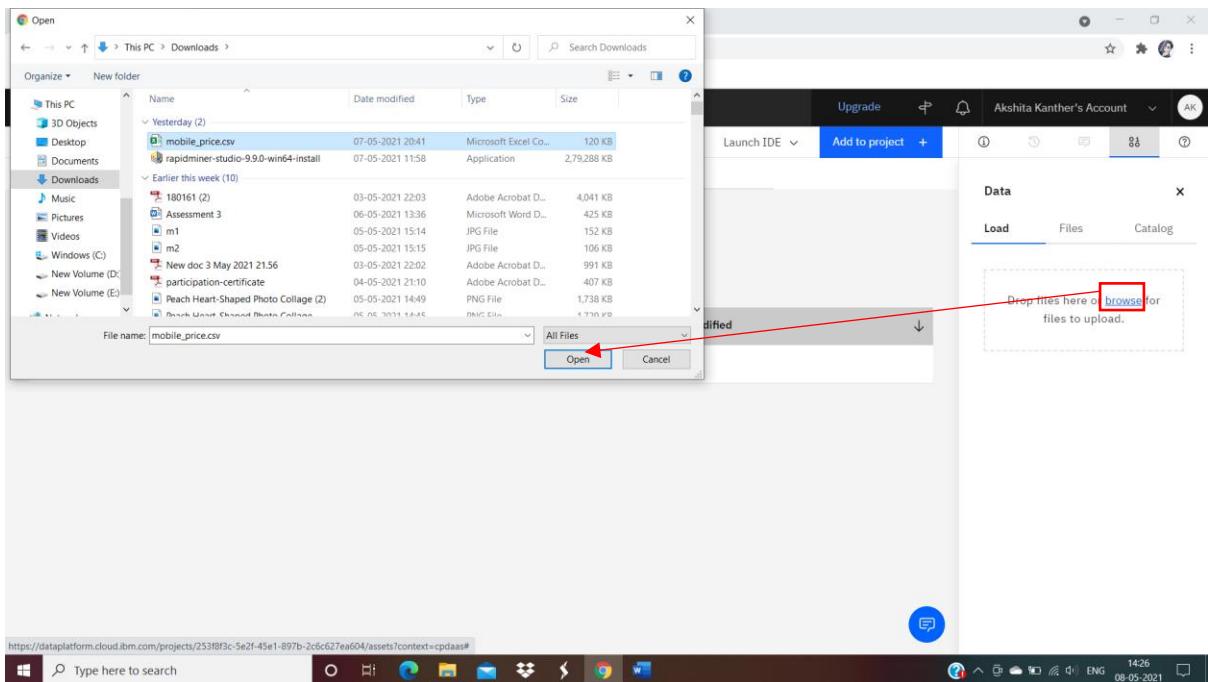
The screenshot shows the 'Create a project' page. At the top, there's a back button and a title 'Create a project'. Below that, a sub-section titled 'Create an empty project' is shown, also highlighted with a red box. This section contains a circular icon with a flowchart and some text about preparing data. To the right, there's a 'USE TO' section with options like 'Prepare and visualize data', 'Analyze data in notebooks', and 'Train models'. Below this, another section titled 'Create a project from a sample or file' is partially visible, featuring a circular icon with a document and plus sign. The bottom of the screen shows a Windows taskbar with various icons.

Step-3. Enter your project name. Also ensure that an instance of Cloud Object Storage is created. Enter project description (optional). Then click on Create.



Step-4. Click on Assets.

Step-5. Click on browse under load in the Data pane on the left side of the page. And upload your dataset in CSV format.



Step-6. Click on Overview .

The screenshot shows the 'Assets' tab in the IBM Cloud Pak for Data interface. The asset 'mobile_price.csv' is listed under 'Data assets'. The interface includes a search bar, navigation tabs (Overview, Assets, Environments, Jobs, Access Control, Settings), and a sidebar for data management.

Step-7. Now we can see that 1 Asset has been uploaded and the no. above Assets has been updated. Now click on Add to project which is on top right side of the page.

The screenshot shows the 'Mobile Price Classification' project page. At the top, there's a navigation bar with 'IBM Cloud Pak for Data' and a search bar. Below it, a toolbar has tabs for 'Overview', 'Assets', 'Environments', 'Jobs', 'Access Control', and 'Settings'. The 'Overview' tab is selected. In the center, there's a summary card with '1 Assets' and '1 Collaborators'. On the left, there's an 'Overview' panel showing creation date (May 08, 2021), description (No description available), storage (122.4 KB used in Cloud Object Storage), and collaborators (Akshita Kanther). On the right, there's a 'Recent activity' section with a placeholder message: 'Alerts related to this project appear here when the project is active.' A blue 'Add to project' button is highlighted with a red box at the top right of the page.

Step-8. Click on AutoAI experiment.

The screenshot shows the 'Assets' tab selected in the navigation bar. A modal window titled 'Choose asset type' is open, listing several options: 'AutoAI experiment' (highlighted with a red box), 'Connection', 'Notebook', 'Dashboard', 'Visual Recognition m...', 'Natural Language Cla...', 'Model from file', 'Federated Learning e...', 'Deep learning exper...', 'Modeler flow', 'Metadata Import', 'Data Refinery flow', and 'Decision Optimizatio...'. To the right of the modal, there's a 'Data' section with tabs for 'Load', 'Files', and 'Catalog', and a placeholder 'Drop files here or browse for files to upload.'

Step-9. Enter the name of the New AutoAI experiment, and click on Associate a Machine Learning service instance.

New AutoAI experiment

Define details

Name: AutoAI experiment

Description: Description of AutoAI experiment

Associate services

Watson Machine Learning Service Instance *

No Machine Learning service instances associated with your project.

Associate a Machine Learning service instance with your project on the project settings page, then click the reload button below to refresh the instances available for association with your new model builder instance.

Reload

Compute configuration * ⓘ

8 vCPU and 32 GB RAM

This compute configuration consumes 20 capacity units per hour for training. The consumption rate differs for joining multiple data sources, depending on your configuration. For details, see Watson Machine Learning plans.

Step-10. Select the Machine learning service instance, in Locations select Dallas and Global, then click on Associate service and then Reload the Machine learning service instance.

Associate service

Choose an existing or add a new service to associate with your project.

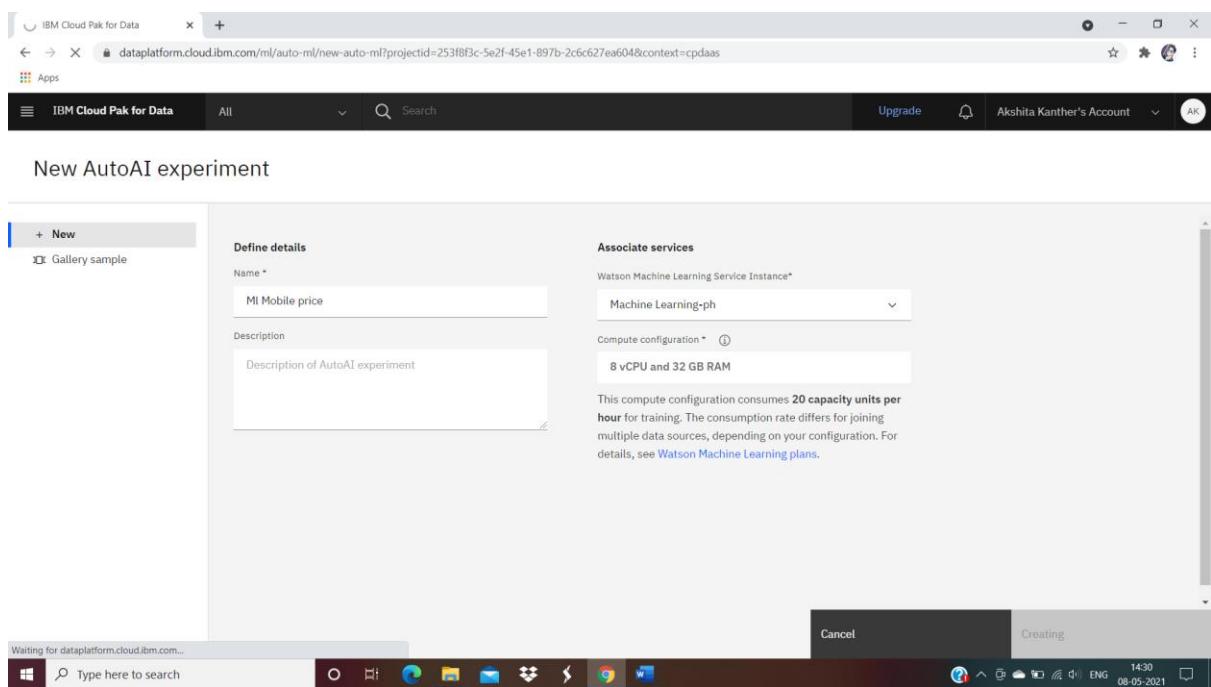
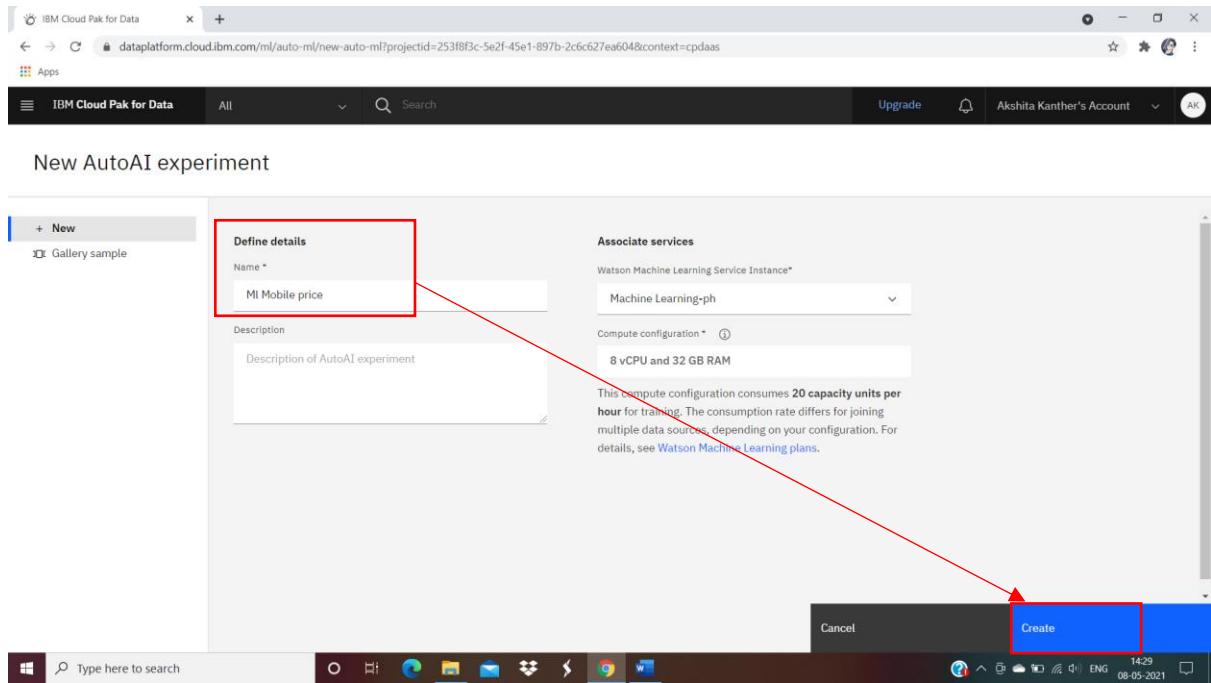
Filter by: Resource Groups ▾

2 x Locations ▾

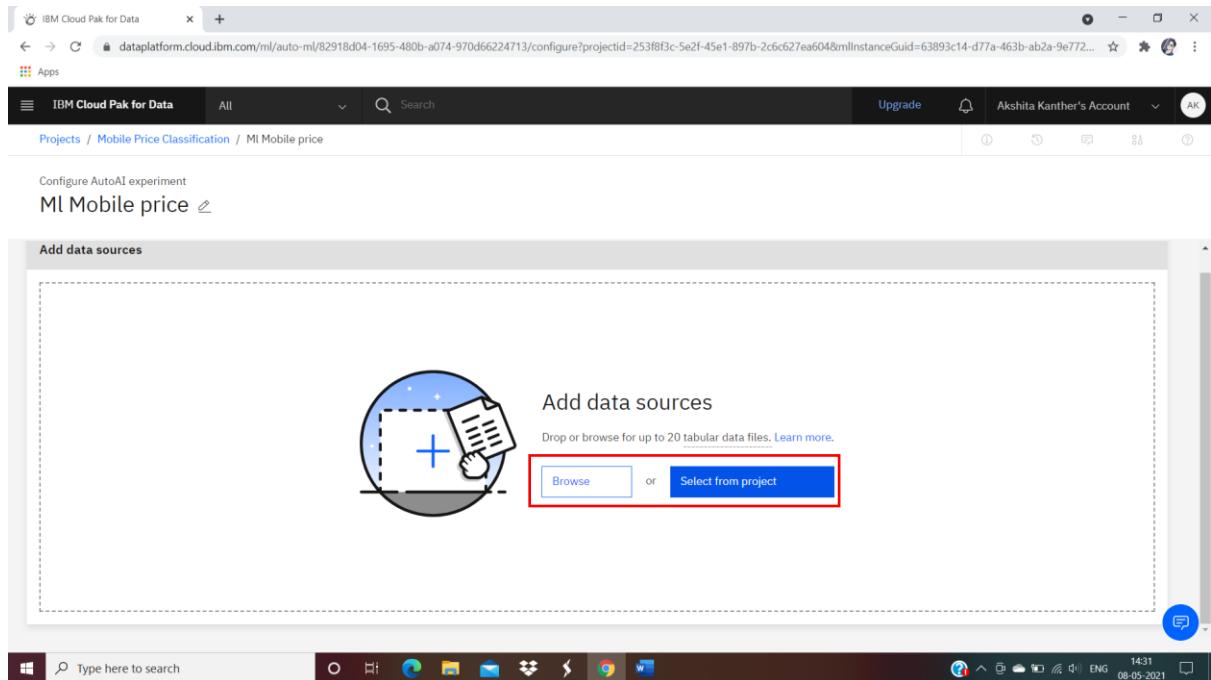
- None
- Dallas
- Global
- Frankfurt

Name	Type	Plan	Location	Status	Group
<input checked="" type="checkbox"/> Machine Learning-ph ⓘ	Machine Learning	Lite	Dallas	Not associated	Default
<input type="checkbox"/> Visual Recognition-5m ⓘ	Visual Recognition	Lite	Dallas	Not associated	Default

Step-11. Click on Create.



Step-12. Click on Browse and upload the dataset.



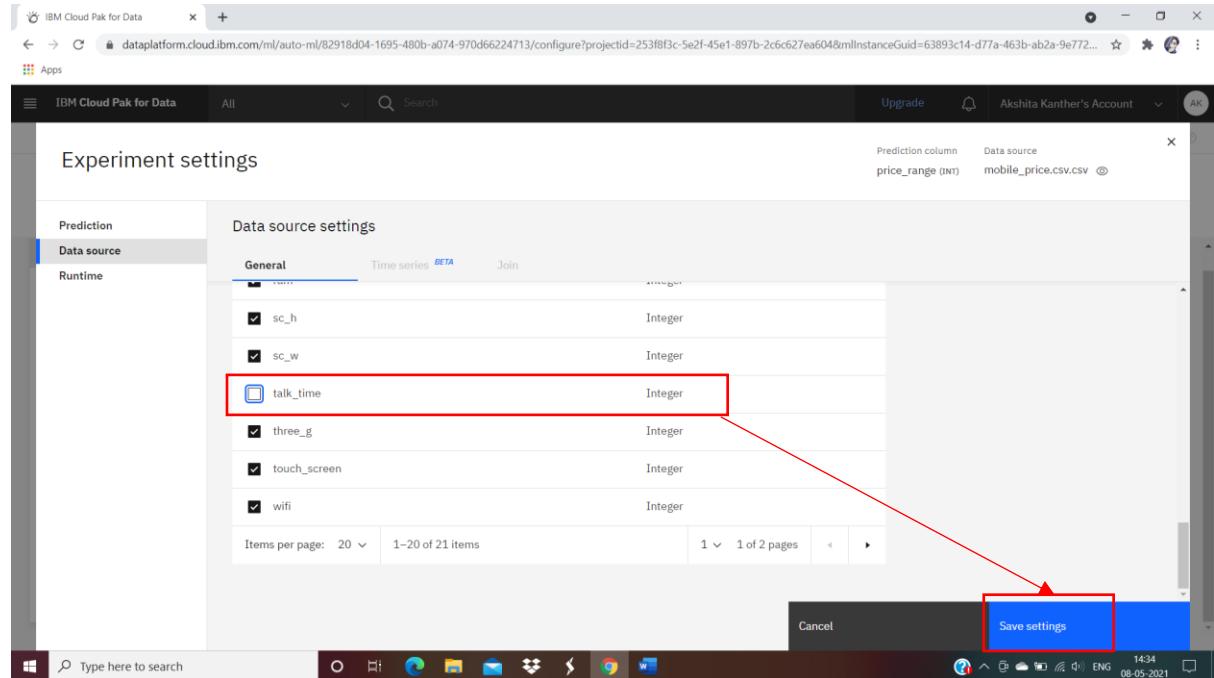
Step-13. After dataset has been uploaded, select No in Create a time series forecast? Then click on dropdown under What do you want to predict? Select the column that you want to predict.

The screenshot shows the 'Configure details' section of the IBM Cloud Pak for Data interface. On the left, the 'Add data sources' panel shows a file named 'mobile_price.csv' (Size: 0.12 MB, Columns: 21). On the right, the 'Configure details' panel includes a section for 'Create a time series forecast? (BETA)' with a 'Yes' and 'No' button, where 'No' is highlighted by a red box. Below it, a section titled 'What do you want to predict?' shows a dropdown menu with 'Select prediction column' and a list of columns: talk_time, three_g, touch_screen, wifi, and price_range, with 'price_range' highlighted by a red box.

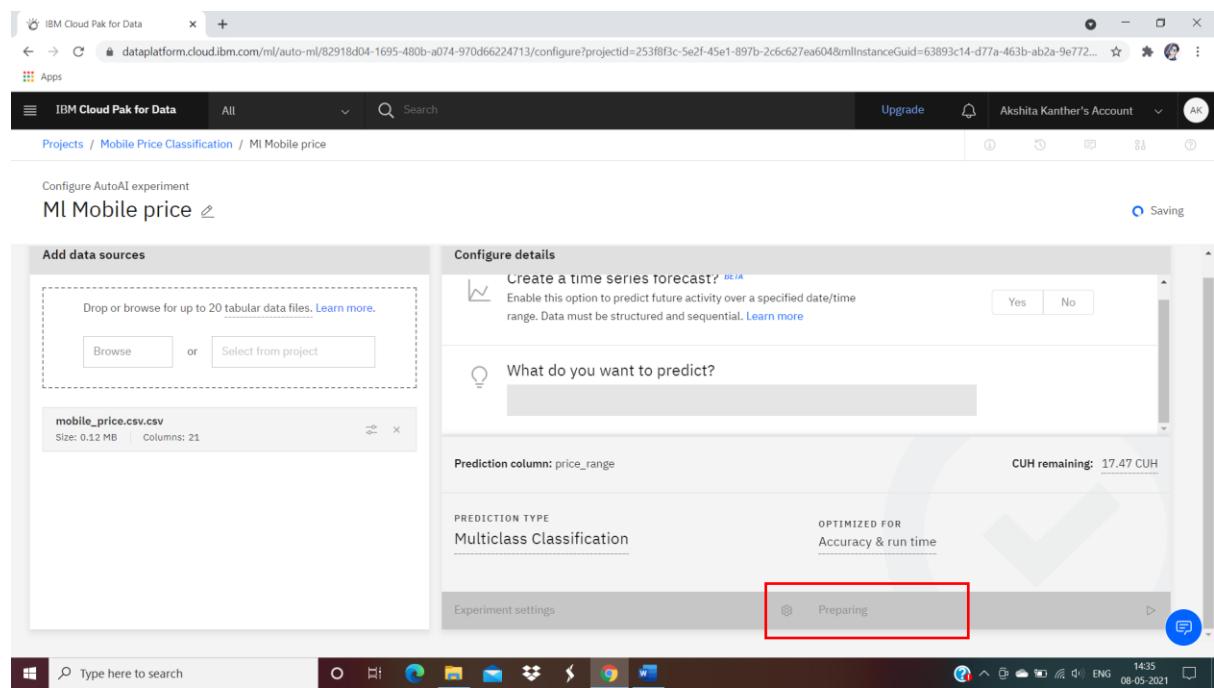
Step-14. It will automatically detect the PREDICTION TYPE, then click on Experiment settings at the bottom.

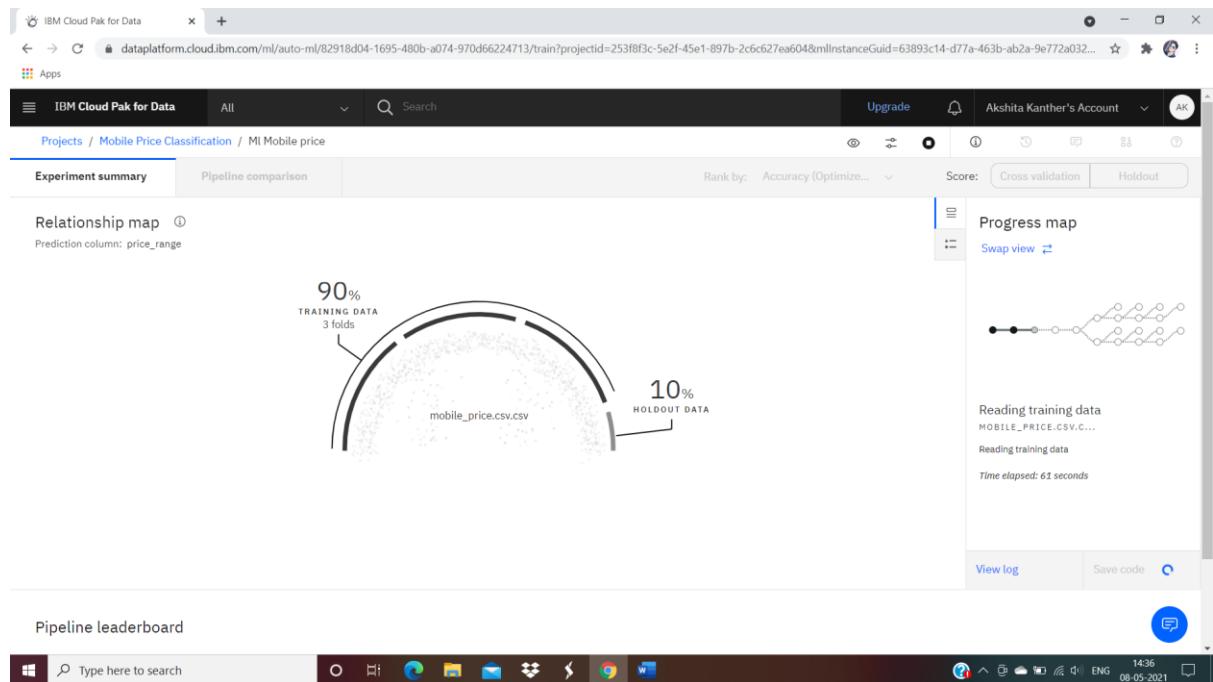
Step-15. Click on Data Source on the left side. By default training and testing data will be split in the ratio of 90% to 10%. This ratio is preferred, we can change according to our requirement by adjusting the scale.

Step-16. Unselect the column names which aren't required for making the model. And then click on Save settings.

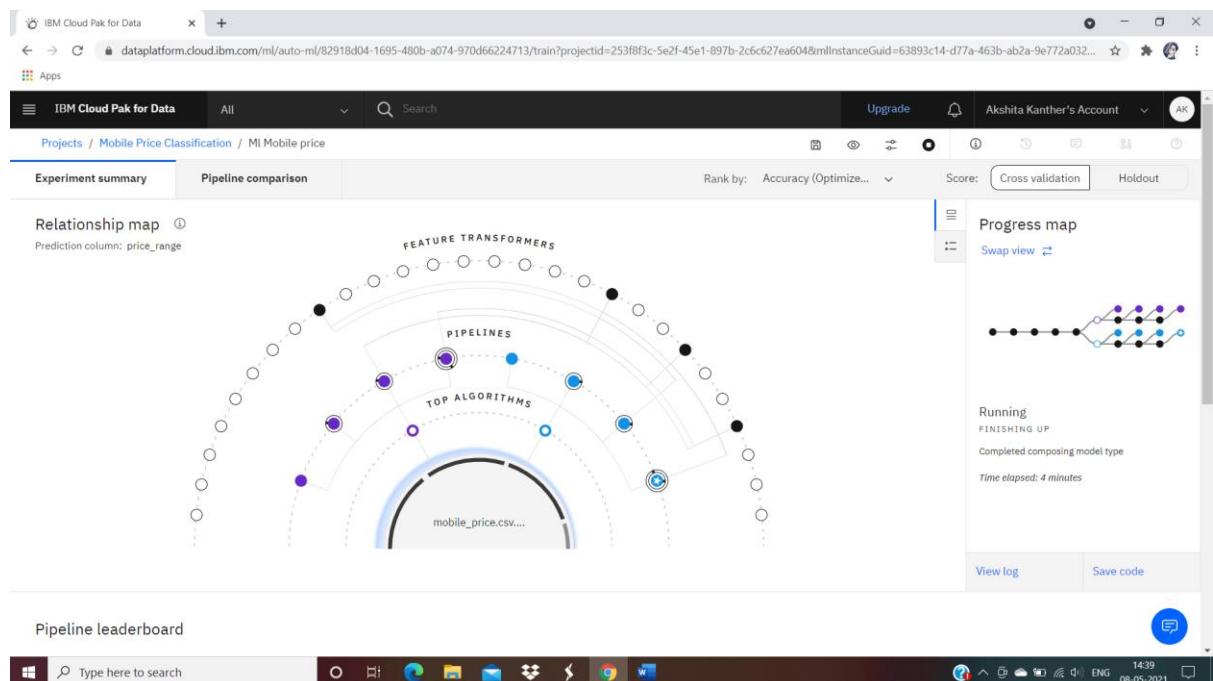


Step-17. Click on Run experiment at the bottom of the page.





Step-18. Pipelines have been generated for the model. Pipeline with * indicates the best pipeline. Click on Swap view under Progress map on the right side of the screen.



Step-19. Click on Pipeline comparison on the top.

Screenshot of the IBM Cloud Pak for Data interface showing the Pipeline comparison tab. A red box highlights the "Pipeline comparison" tab. The interface displays a "Progress map" showing the flow from "Read dataset" to "Model selection" through various stages like "Split holdout data", "Read training data", "Preprocessing", and "Hyperparameter optimization". A "Relationship map" shows connections between different pipelines. The "Experiment completed" section indicates 8 pipelines generated. The "Metric chart" section is visible at the bottom.

Screenshot of the IBM Cloud Pak for Data interface showing the Pipeline comparison tab. A red box highlights the "Pipeline comparison" tab. The interface displays a "Metric chart" comparing eight pipelines (P1-P8) across various metrics: Accuracy, Log loss, F1 macro, F1 micro, F1 weighted, Precision micro, Precision macro, Precision weighted, Recall micro, Recall macro, and Recall weighted. The chart shows that Pipeline P1 has the highest accuracy (~0.94) and lowest log loss (~0.02), while Pipeline P8 has the lowest accuracy (~0.78).

Step-20. Scroll down and view Pipeline leaderboard.

The screenshot shows the IBM Cloud Pak for Data interface. The top navigation bar includes 'IBM Cloud Pak for Data', 'All', 'Search', 'Upgrade', 'Akshita Kanther's Account', and various icons. Below the header, the URL is dataplatform.cloud.ibm.com/ml/auto-ml/82918d04-1695-480b-a074-970d66224713/train?projectId=253f8f3c-5e2f-45e1-897b-2c6c627ea604&mlInstanceGuid=63893c14-d77a-463b-ab2a-9e772a032.... The main content area displays the 'Experiment summary' tab, which is currently selected. It shows a 'Pipeline comparison' section with a 'Rank by: Accuracy (Optimize...)' dropdown set to 'Cross validation'. A table titled 'Pipeline leaderboard' lists eight pipelines, each with its rank, name, algorithm, and various performance metrics. Pipeline 8 is highlighted with a blue star icon and a rank of 1, with an accuracy of 0.937.

Rank	Name	Algorithm	Acc...	F1 ...	F1 ...	F1 ...	Log...	Pre...	Pre...	Pre...	Rec...	Rec...
1	Pipeline 8	RandomForestClassifier	0.937	0.937	0.937	0.937	0.182	0.937	0.937	0.937	0.937	0.937
2	Pipeline 7	RandomForestClassifier	0.932	0.932	0.932	0.932	0.210	0.933	0.932	0.933	0.932	0.932
3	Pipeline 4	DecisionTreeClassifier	0.914	0.914	0.914	0.914	1.043	0.915	0.914	0.915	0.914	0.914
4	Pipeline 3	DecisionTreeClassifier	0.912	0.912	0.912	0.912	3.032	0.913	0.912	0.913	0.912	0.912
5	Pipeline 6	RandomForestClassifier	0.864	0.864	0.864	0.864	0.409	0.864	0.864	0.864	0.864	0.864
6	Pipeline 1	DecisionTreeClassifier	0.823	0.823	0.823	0.823	6.121	0.823	0.823	0.823	0.823	0.823
7	Pipeline 2	DecisionTreeClassifier	0.823	0.823	0.823	0.823	6.121	0.823	0.823	0.823	0.823	0.823
8	Pipeline 5	RandomForestClassifier	0.776	0.777	0.776	0.777	0.772	0.779	0.776	0.779	0.776	0.776

Step-21. View pipeline with 1st rank, it is the best pipeline with a * and best results. Click on it. Then click on Model Evaluation which is on the left side of the screen. Model accuracy, ROC curve and Model Evaluation measures are displayed on the screen.

This screenshot is identical to the one above, but the row for Pipeline 8 has been highlighted with a red rectangular box. All other rows remain unhighlighted.

Pipeline 8

Rank 1

Random Forest Classifier

EVALUATION

Model Evaluation

Precision Recall Curve

Threshold Chart

MODEL VIEWER

Model Information

Feature Transformations

Feature Importance

Holdout Accuracy (Optimized) 0.955

Algorithm RandomForestClassifier

Enhancements HPO-1 FE HPO-2

Save as

Model Evaluation

TARGET : PRICE_RANGE

View: Multi-Class

Model Accuracy 0.955

ROC Curve

positive rate (sensitivity)

Model Evaluation Measures

	Holdout Score	Cross Validation Score
Macro Precision	0.955	0.937
Accuracy	0.955	0.937
Macro Recall	0.955	0.937

Step-22. Click on the dropdown which is present in front of the view and select 0(One v. Rest). View the output for one class of the predicted column.

Pipeline 8

Rank 1

Random Forest Classifier

EVALUATION

Model Evaluation

Precision Recall Curve

Threshold Chart

MODEL VIEWER

Model Information

Feature Transformations

Feature Importance

Holdout Accuracy (Optimized) 0.955

Algorithm RandomForestClassifier

Enhancements HPO-1 FE HPO-2

Save as

Model Evaluation

TARGET : PRICE_RANGE

View: 0 (One v. Rest)

Model Accuracy 0.995

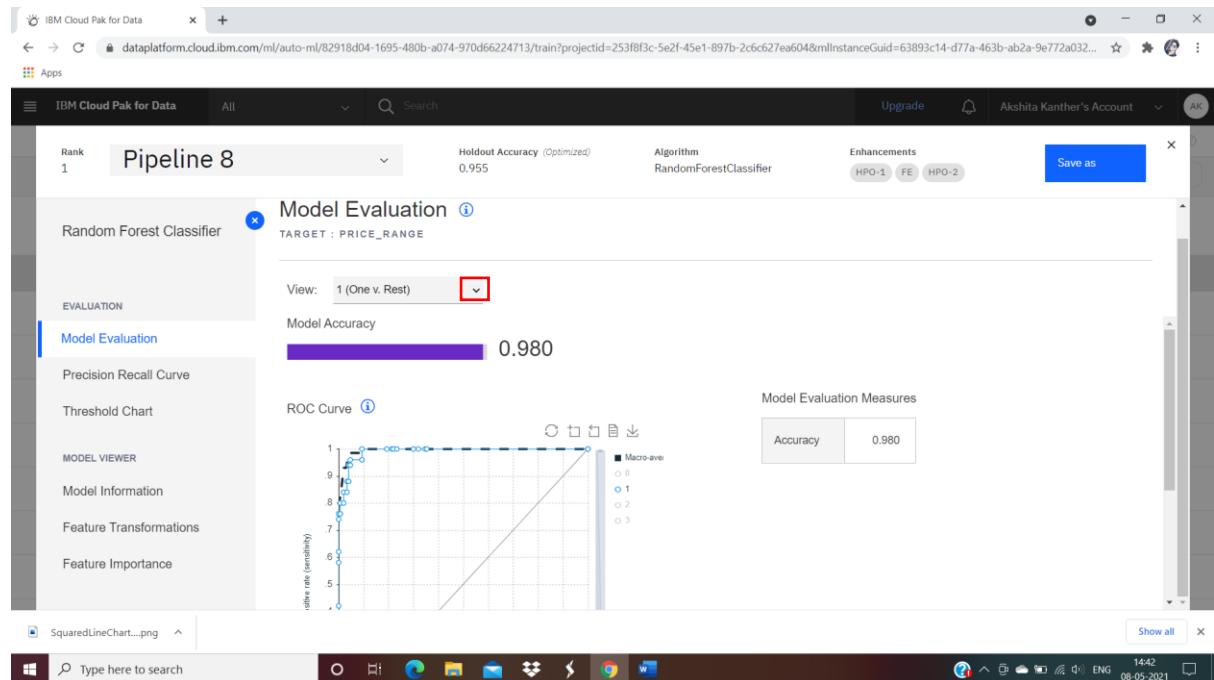
ROC Curve

positive rate (sensitivity)

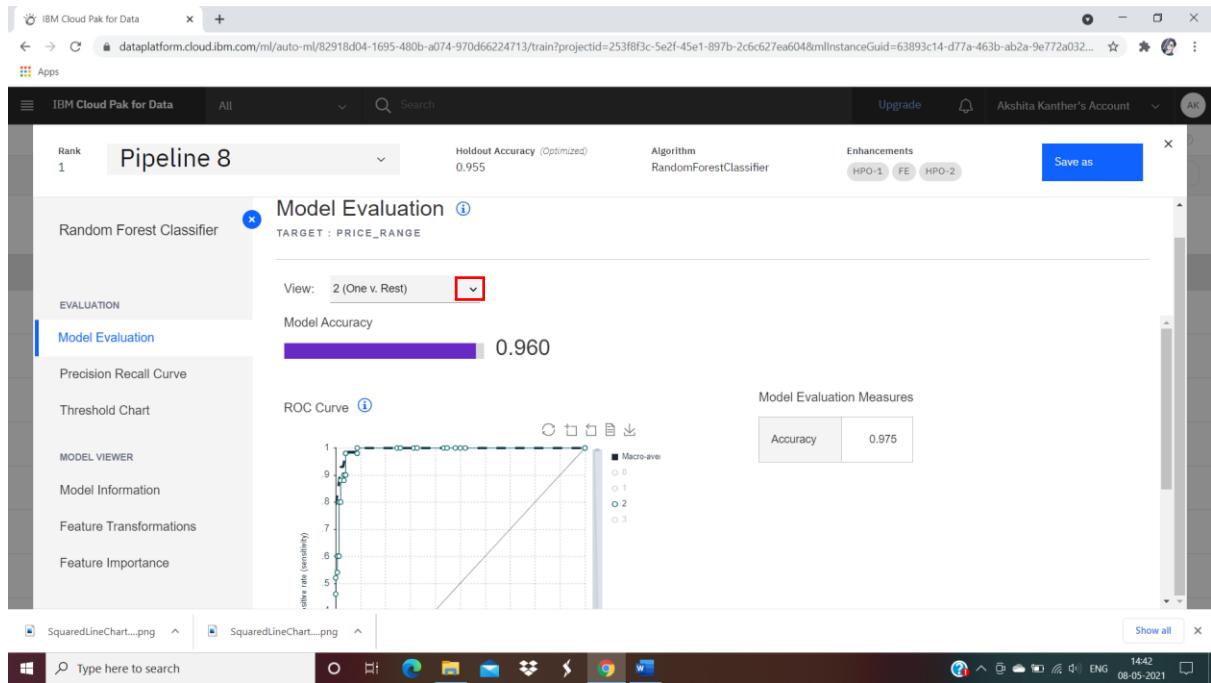
Model Evaluation Measures

	Holdout Score
Accuracy	0.995

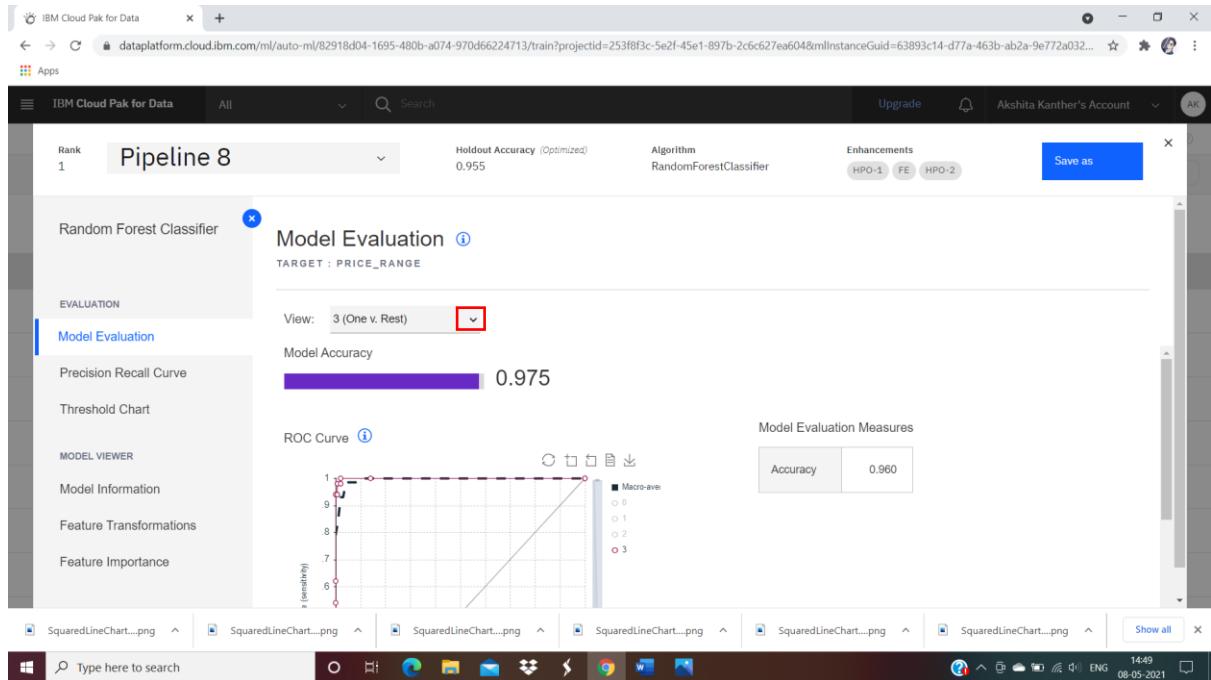
Step-23. Click on the dropdown which is present in front of the view and select 1(One v. Rest) .View the output for one class of the predicted column.

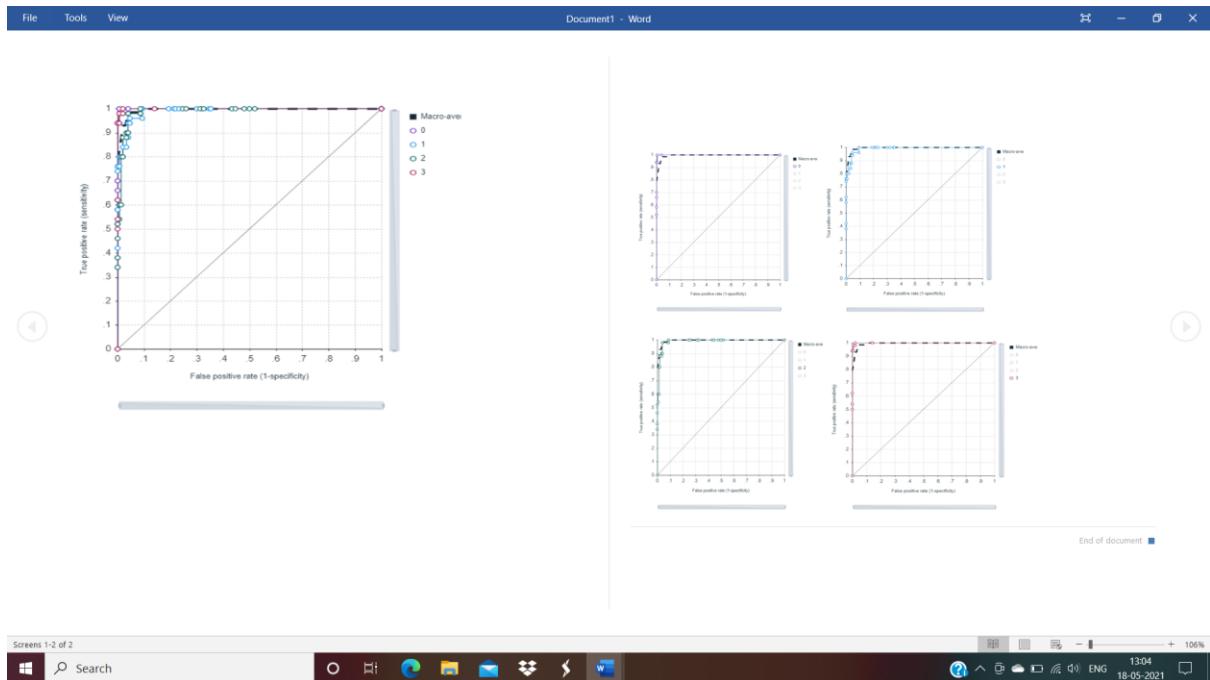


Step-24. Click on the dropdown which is present in front of the view and select 2(One v. Rest) .View the output for one class of the predicted column.

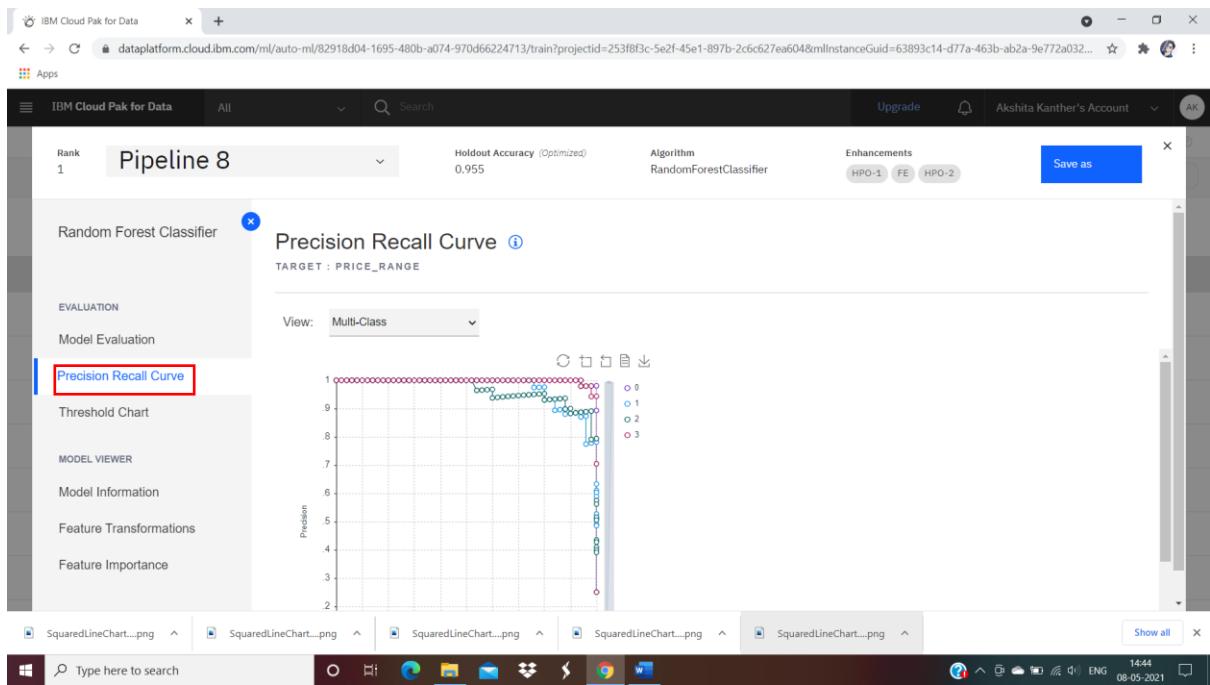


Step-25. Click on the dropdown which is present in front of the view and select 3(One v. Rest) .View the output for one class of the predicted column.

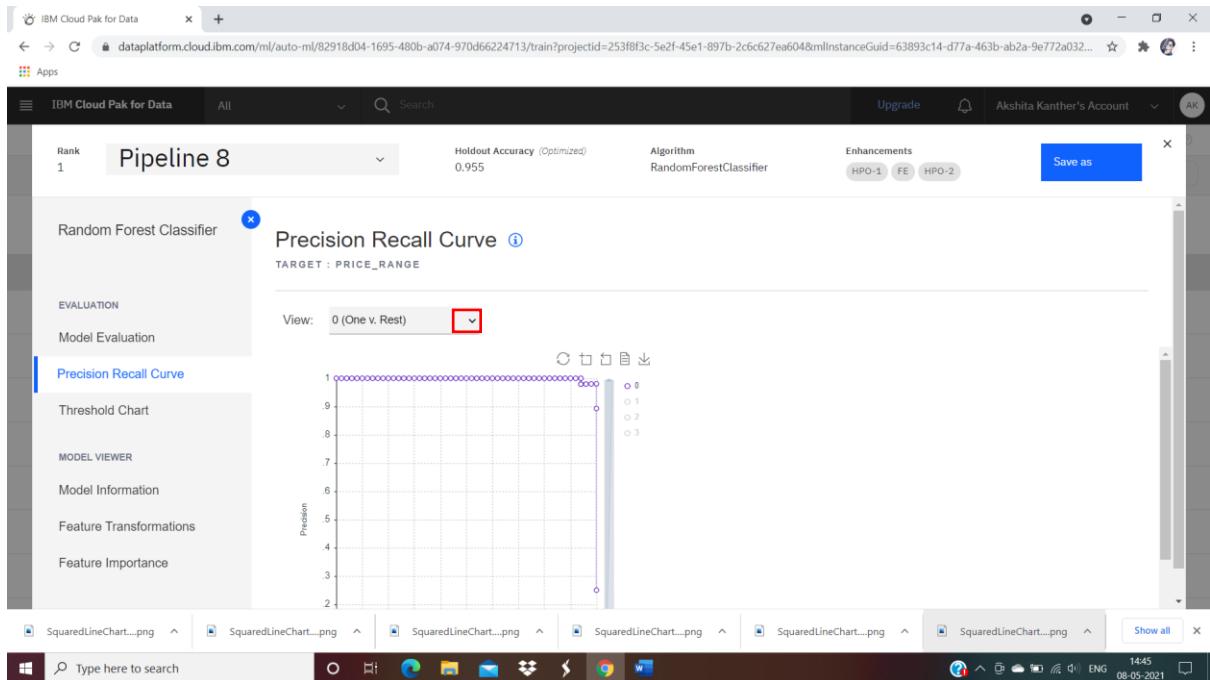




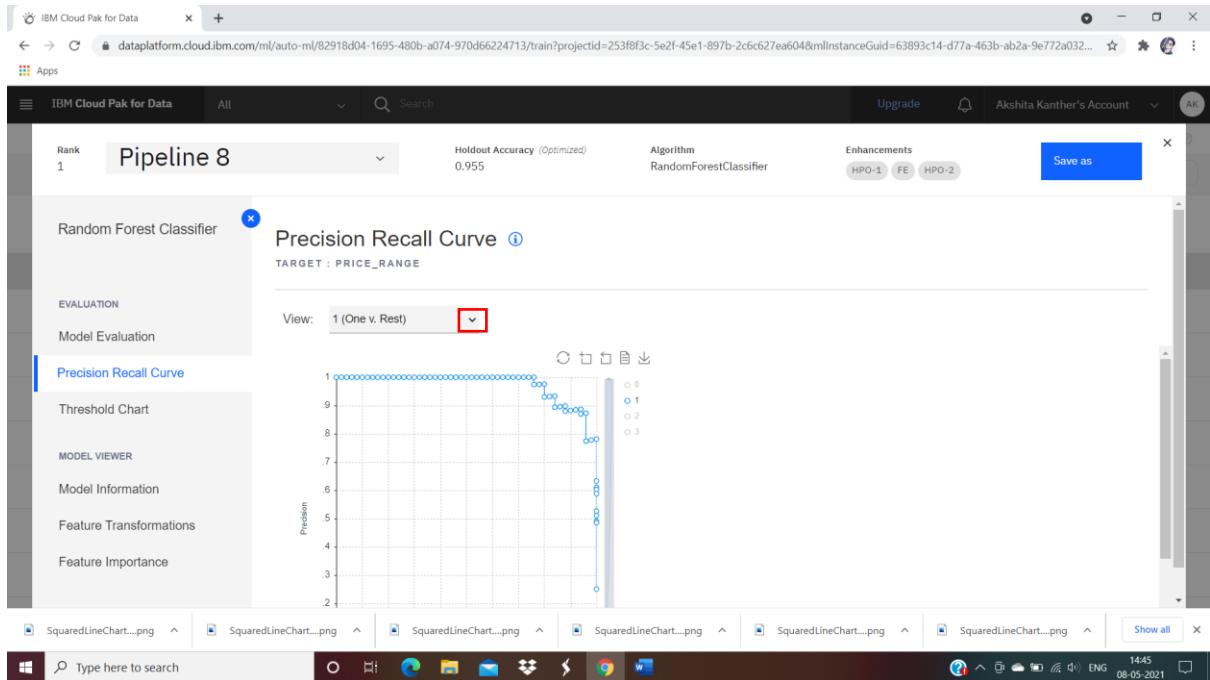
Step-26. Click on Precision Recall Curve on the left side and view it for Multi-Class view.



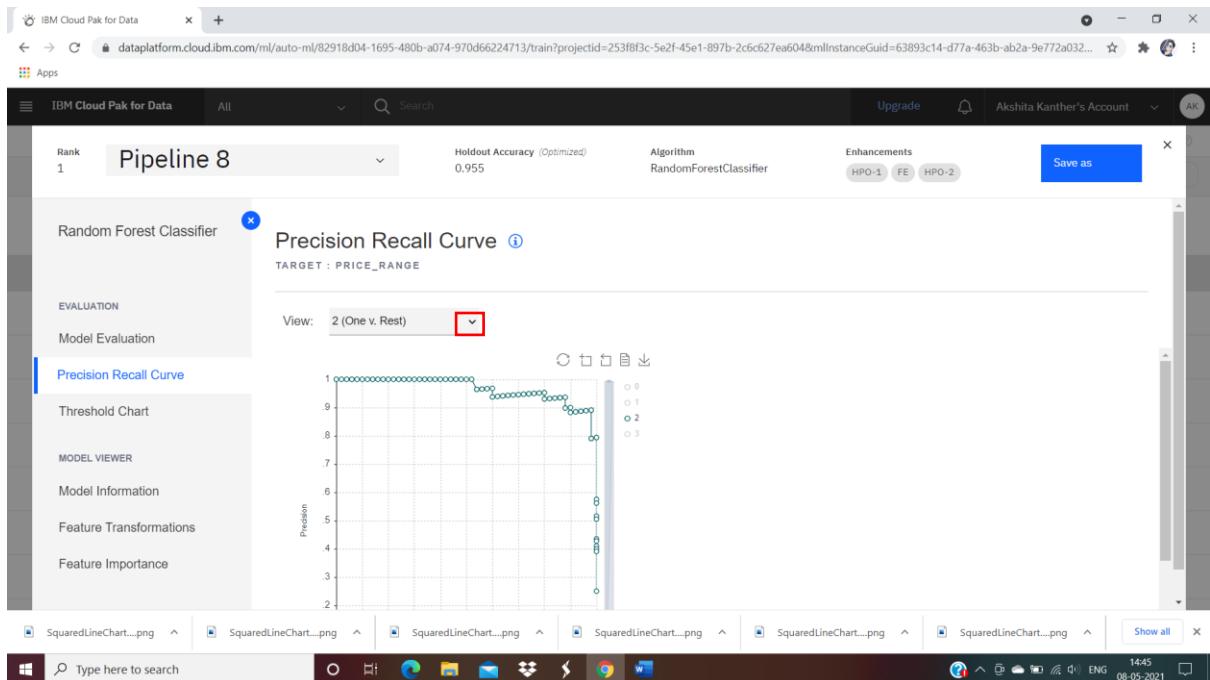
Step-27. Click on the dropdown which is present in front of the view and select 0(One v. Rest) .View the Precision-Recall curve for one class of the predicted column.



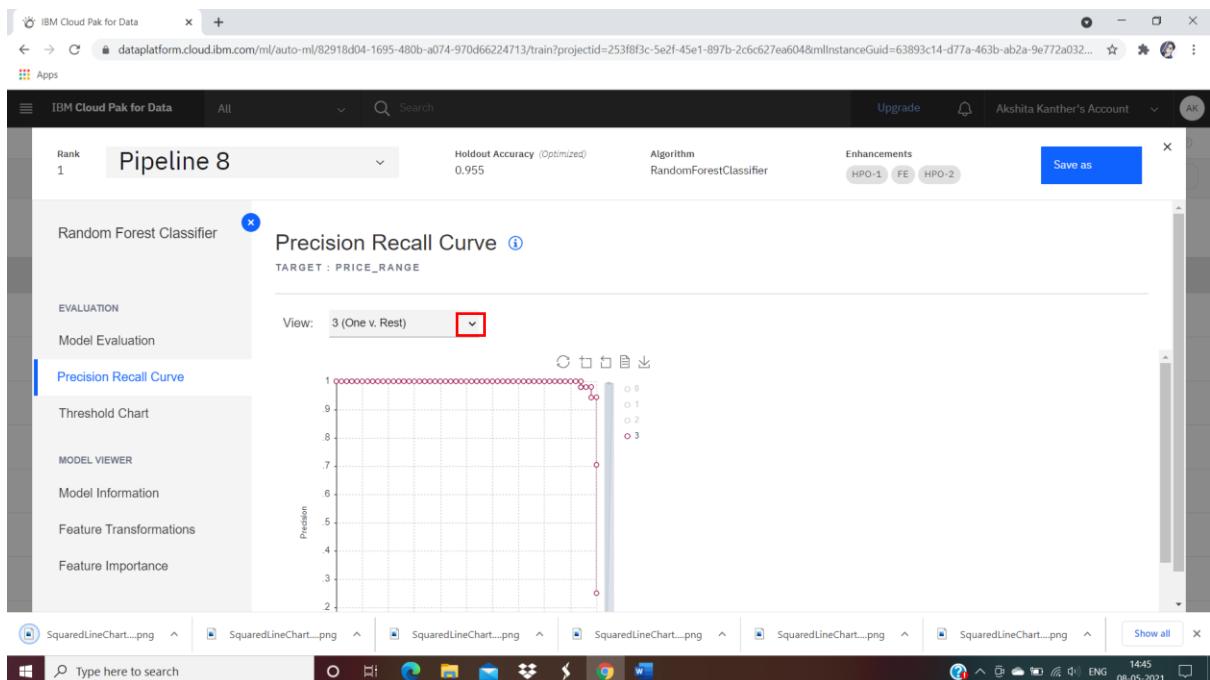
Step-28. Click on the dropdown which is present in front of the view and select 1(One v. Rest) .View the Precision-Recall curve for one class of the predicted column.

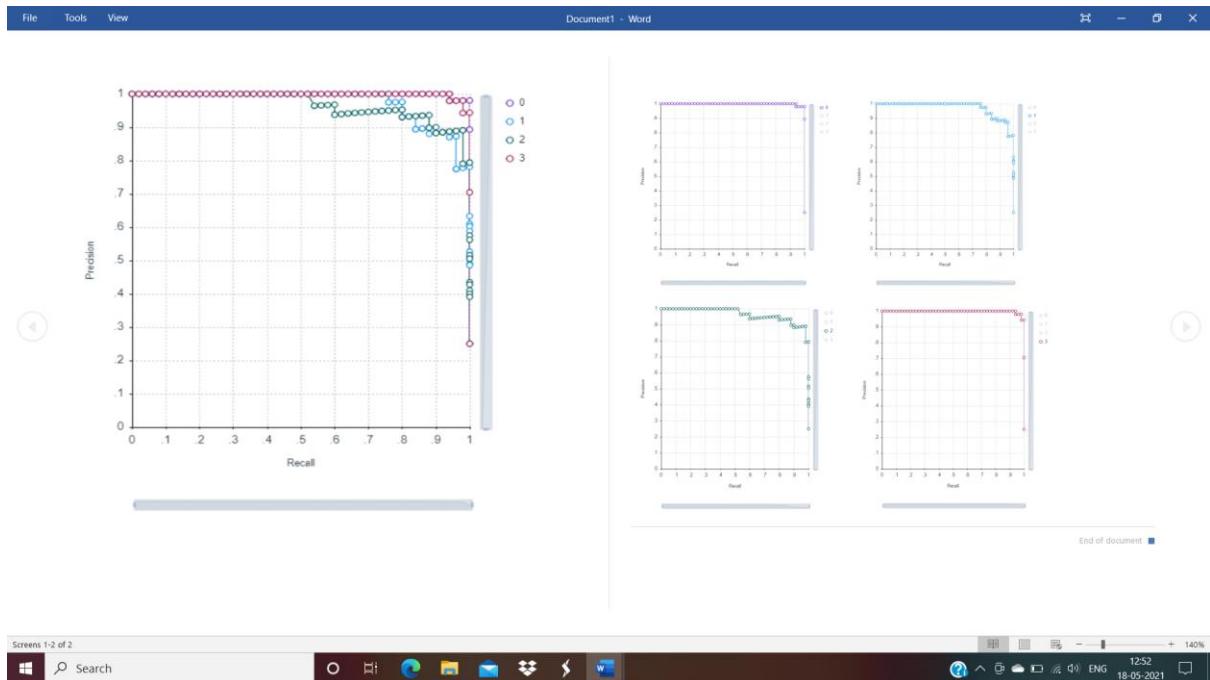


Step-29. Click on the dropdown which is present in front of the view and select 2(One v. Rest) .View the Precision-Recall curve for one class of the predicted column.

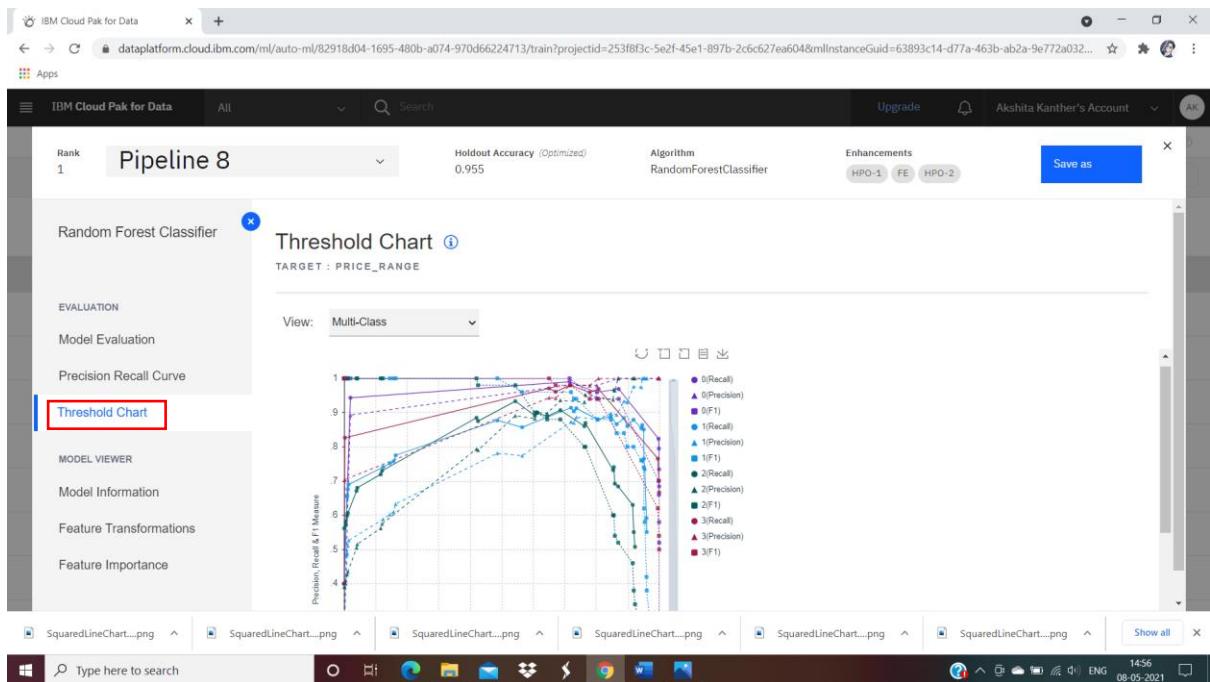


Step-30. Click on the dropdown which is present in front of the view and select 3(One v. Rest) .View the Precision-Recall curve for one class of the predicted column.

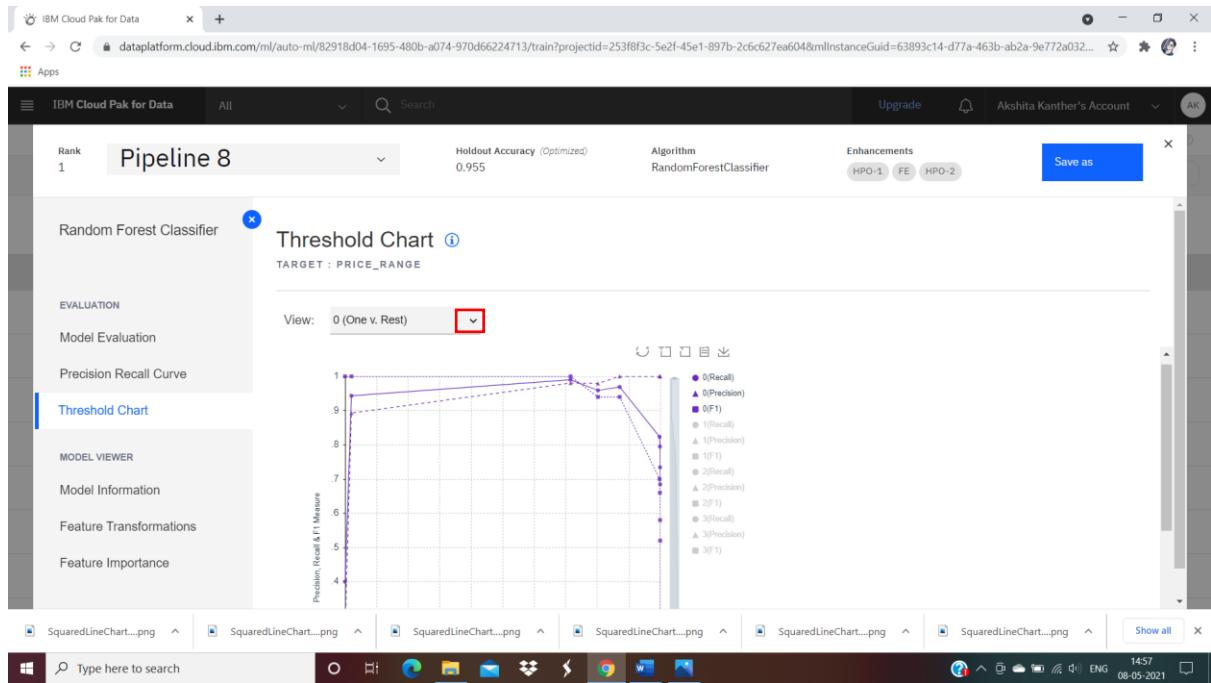




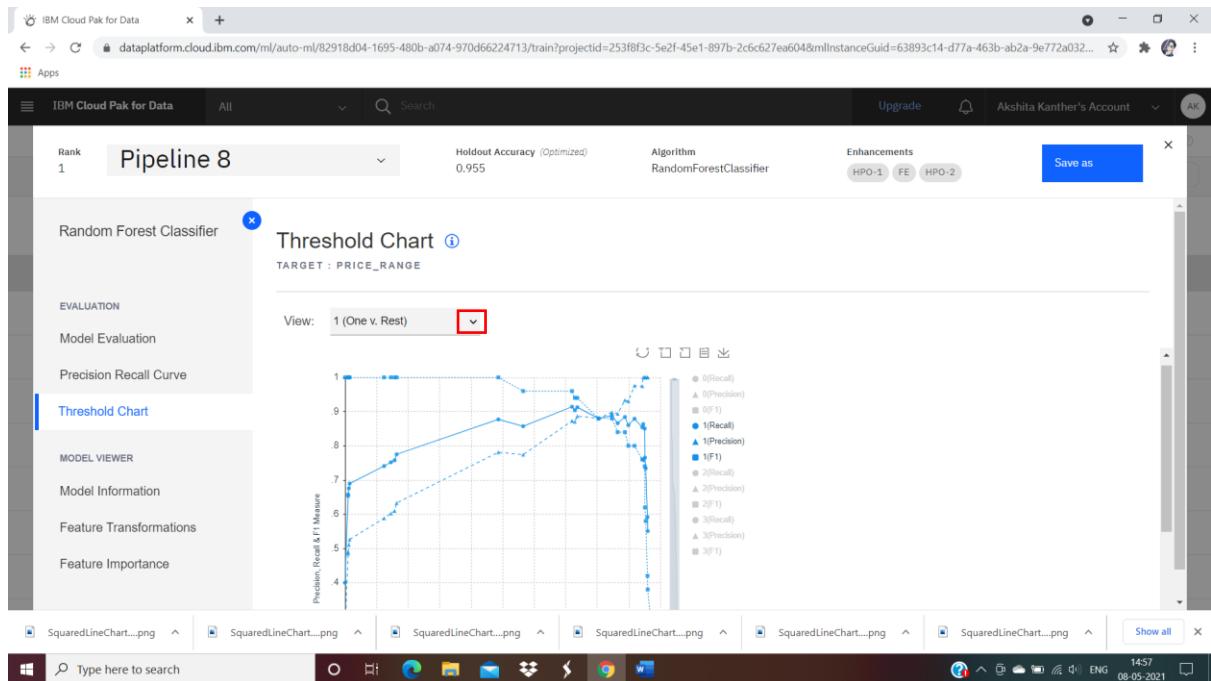
Step-31. Click on Threshold Chart on the left side and view it for Multi-Class view.



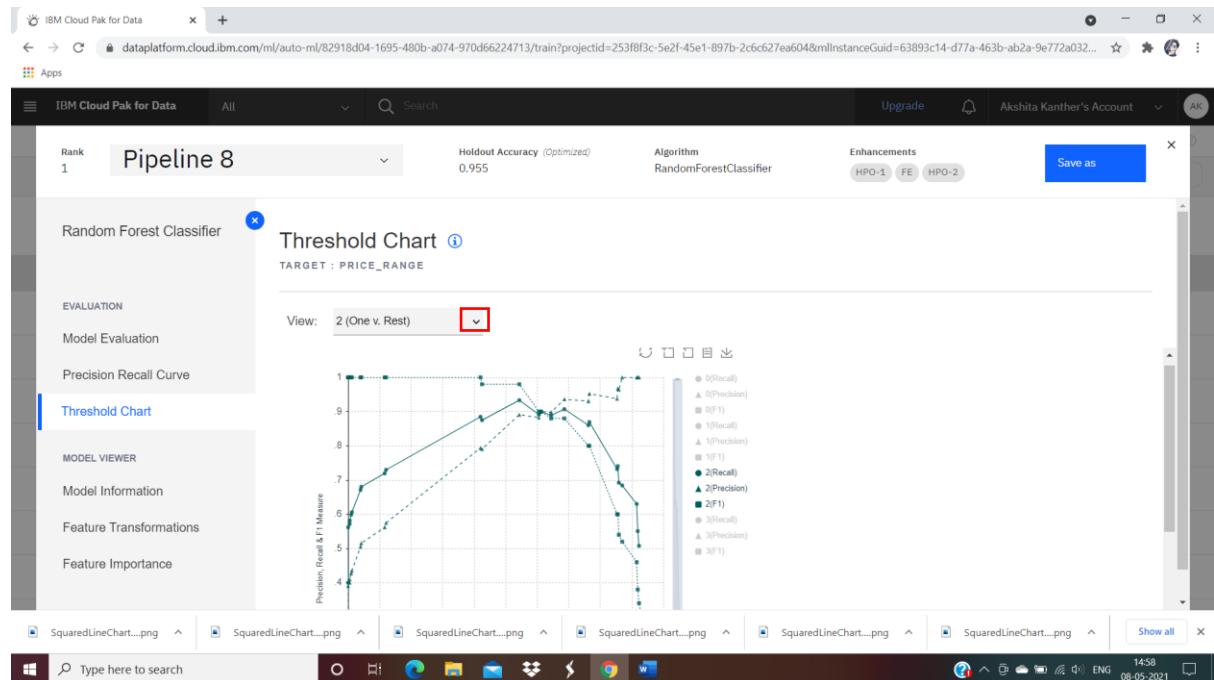
Step-32. Click on the dropdown which is present in front of the view and select 0(One v. Rest) .View the Threshold Chart for one class of the predicted column.



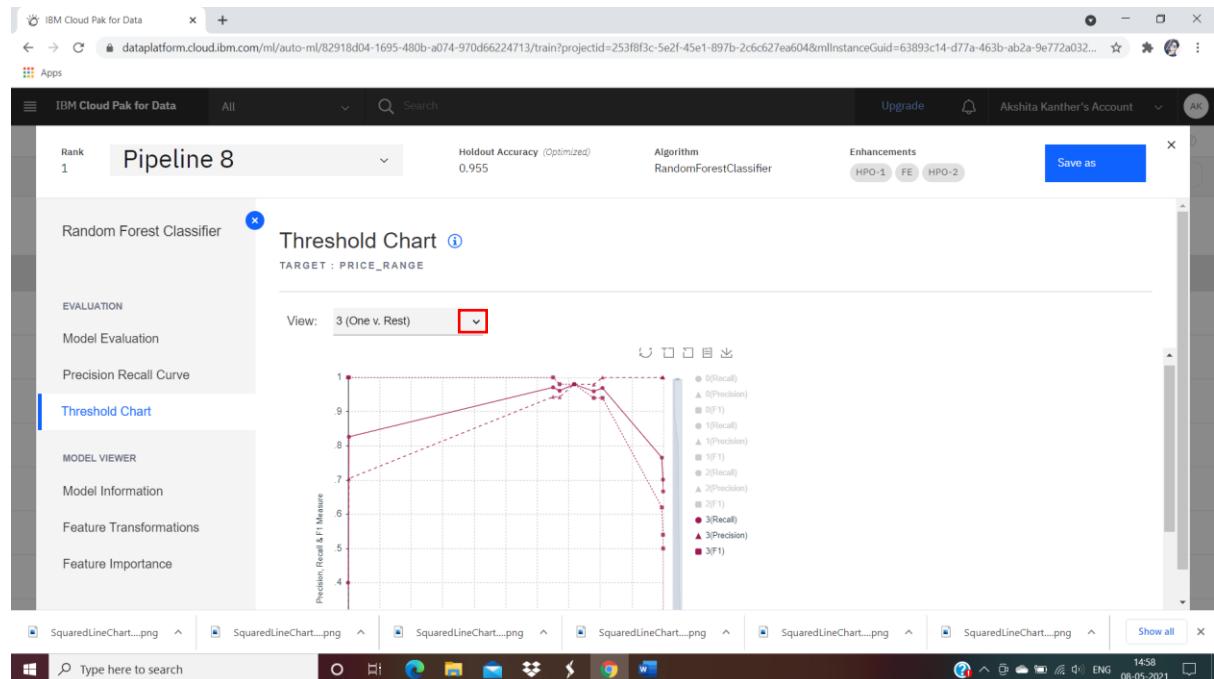
Step-33. Click on the dropdown which is present in front of the view and select 1(One v. Rest) .View the Threshold Chart for one class of the predicted column.

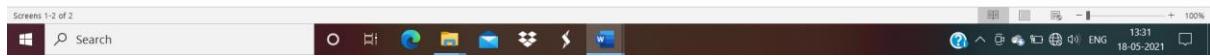
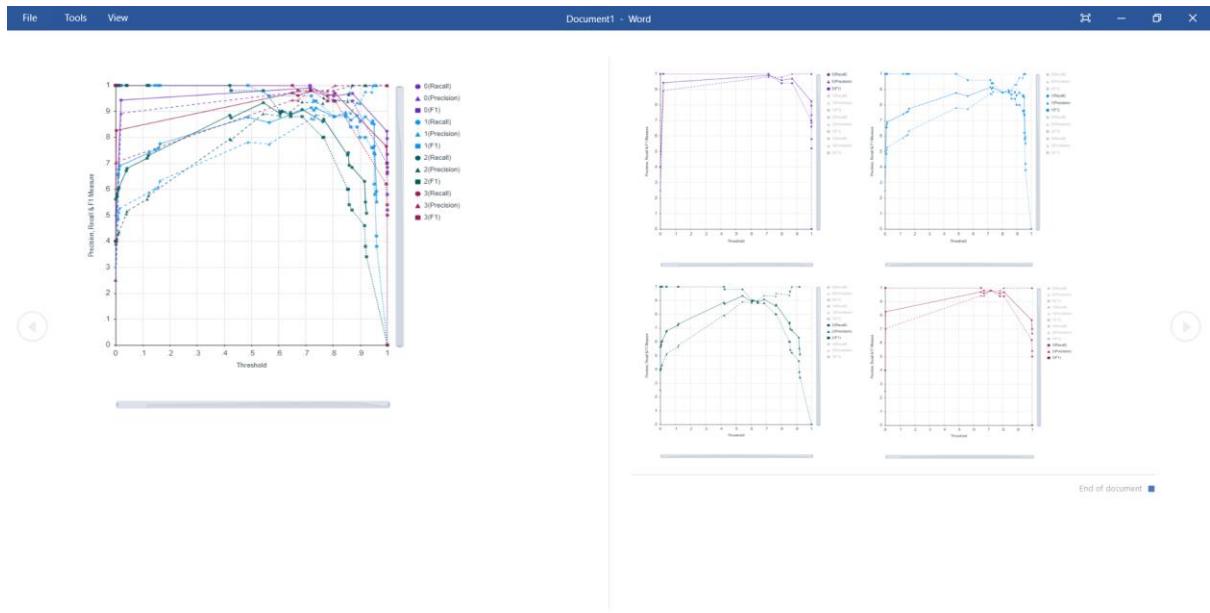


Step-34. Click on the dropdown which is present in front of the view and select 2(One v. Rest) .View the Threshold Chart for one class of the predicted column.



Step-35. Click on the dropdown which is present in front of the view and select 3(One v. Rest) .View the Threshold Chart for one class of the predicted column.





Step-36. Click on Model Information on the left side of the screen and view the Model Type, Number of Features, date of creation and Label.

Label (Target)	price_range
Model Type	Random Forest Classifier
Number of Features	34
Created At	5/8/2021, 2:39:42 PM

Step-37. Click on Feature Transformations on the left side of the screen and view the new features which are created or transformed by combining existing features or by applying Principal Component Analysis.

The screenshot shows the IBM Cloud Pak for Data interface. On the left, there's a sidebar with options like Rank, Pipeline 8, Random Forest Classifier, EVALUATION, MODEL VIEWER, and Feature Transformations (which is highlighted with a red box). The main area displays the 'Feature Transformations' panel for a target named 'PRICE_RANGE'. The panel includes a table with columns for New Feature, Original Feature, and Transformation. The table lists several entries, such as 'NewFeature_10' derived from 'All' using 'pca(ALL)', and 'NewFeature_2' derived from 'battery_power,ram' using 'product(battery_power,ram)'. A red arrow points from the 'Feature Transformations' link in the sidebar to the corresponding section in the main content area.

New Feature	Original Feature	Transformation
NewFeature_10	All	pca(ALL)
NewFeature_2	battery_power,ram	product(battery_power,ram)
NewFeature_11	All	pca(ALL)
NewFeature_1	battery_power,px_width	product(battery_power,px_width)
NewFeature_3	clock_speed,ram	product(clock_speed,ram)

Step-38. Scroll down and observe all the feature transformations.

This screenshot is similar to the previous one but shows the 'Feature Transformations' panel after scrolling down. The table now contains more rows, including 'NewFeature_9' (derived from 'cpu,ram' using 'product(cpu,ram)'), 'NewFeature_12' (derived from 'All' using 'pca(ALL)'), 'NewFeature_7' (derived from 'pc,ram' using 'product(pc,ram)'), 'NewFeature_5' (derived from 'int_memory,ram' using 'product(int_memory,ram)'), 'NewFeature_14' (derived from 'All' using 'pca(ALL)'), and 'NewFeature_6' (derived from 'mobile_bt,ram' using 'product(mobile_bt,ram)'). A vertical scroll bar is visible on the right side of the panel, indicating that there are more transformations listed below what is currently visible.

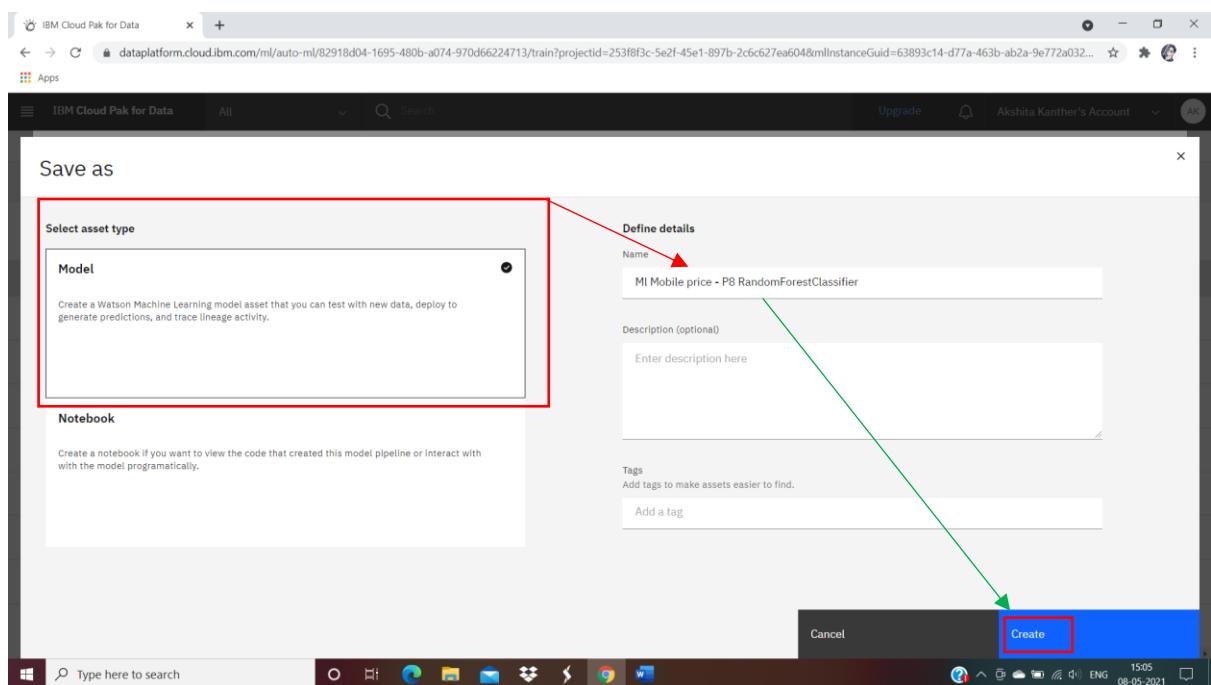
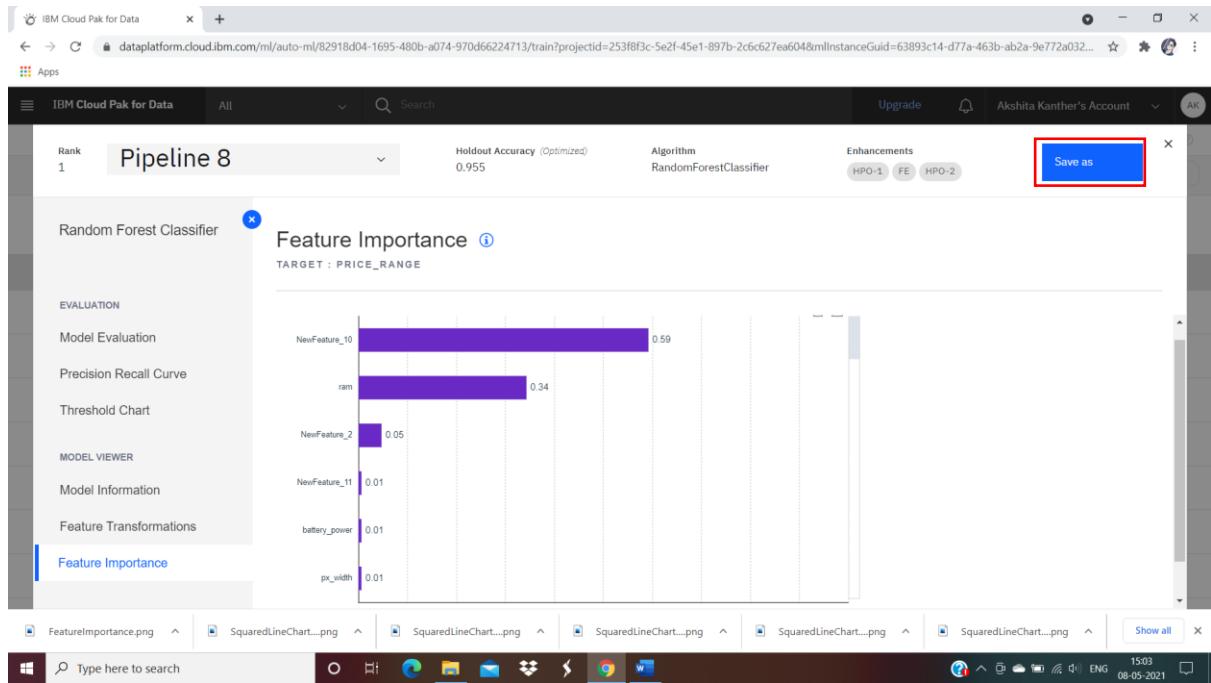
The screenshot shows the IBM Cloud Pak for Data interface. At the top, it displays "Pipeline 8" with a holdout accuracy of 0.955 and an algorithm of RandomForestClassifier. There are tabs for "EVALUATION" (Model Evaluation, Precision Recall Curve, Threshold Chart), "MODEL VIEWER" (Model Information, Feature Transformations, Feature Importance), and "Feature Transformations". The "Feature Transformations" section shows a table mapping new features to original features and their transformations:

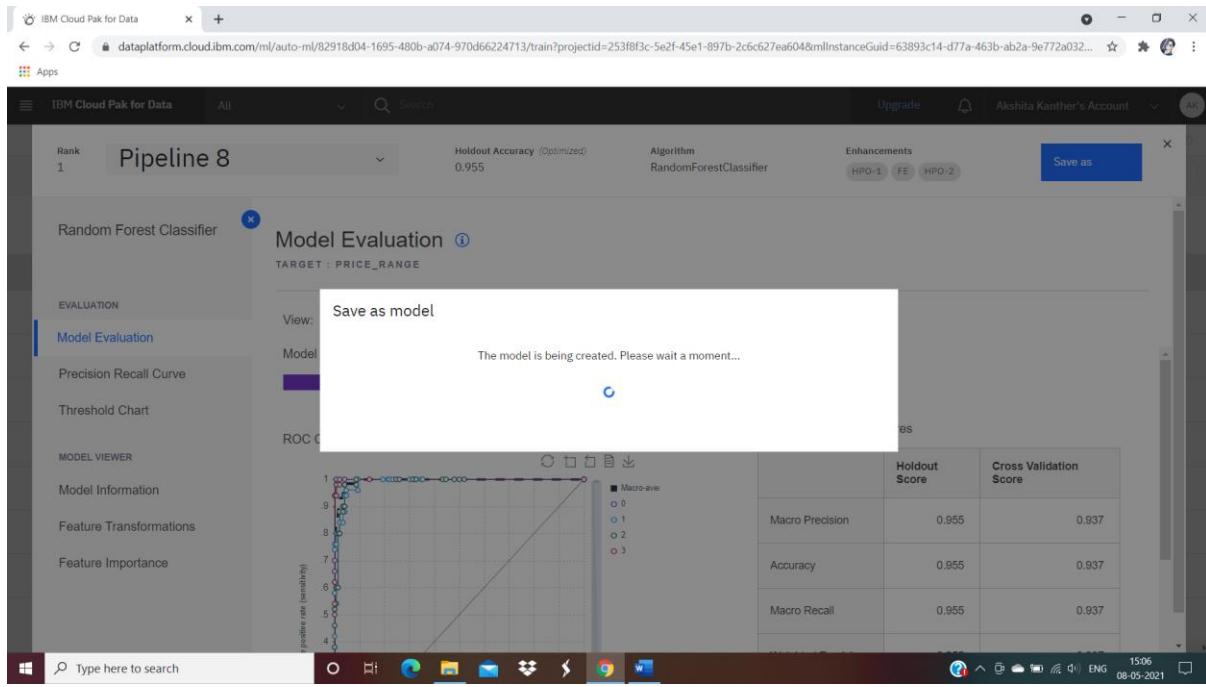
New Feature	Original Feature	Transformation
NewFeature_0	mobile_width	product(mobile_width,ram)
NewFeature_13	All	pca(ALL)
NewFeature_9	px_width,ram	product(px_width,ram)
NewFeature_0	battery_power,px_height	product(battery_power,px_height)
NewFeature_8	px_height,ram	product(px_height,ram)
NewFeature_4	fc,ram	product(fc,ram)

Step-39. Click on Feature Importance on the left side of the screen and observe the contribution of each feature in prediction.

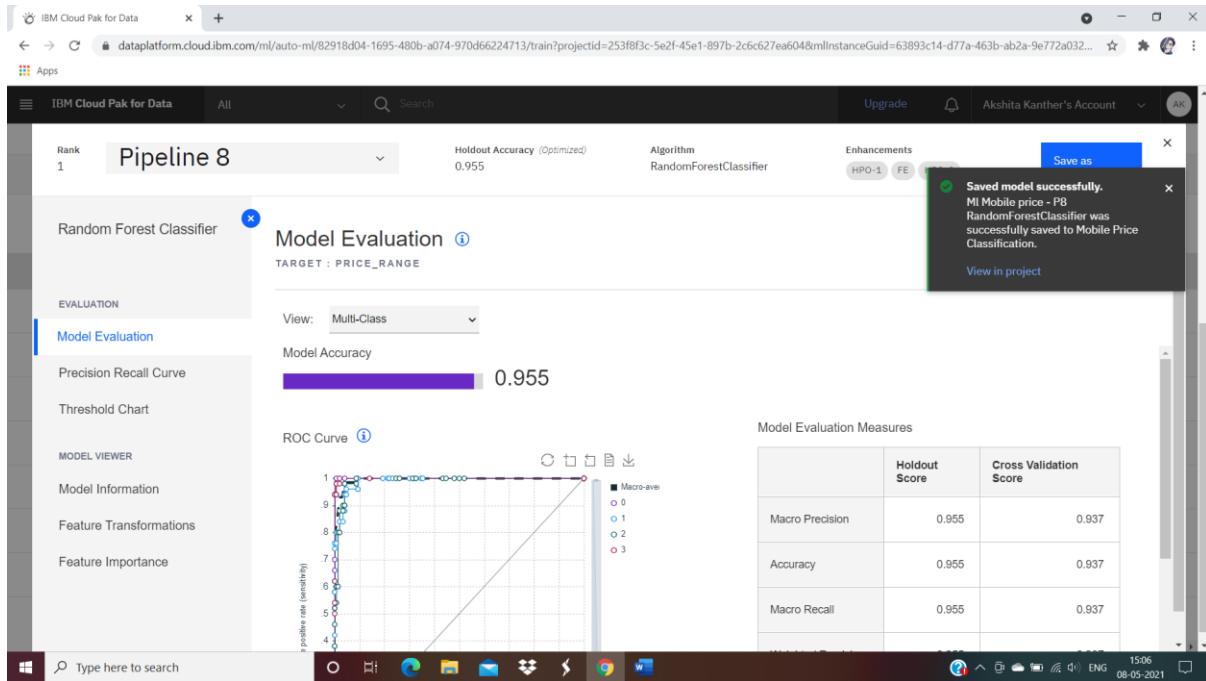
The screenshot shows the same interface as above, but the "Feature Importance" section is highlighted with a red arrow pointing to the "Feature Importance" button in the sidebar. The "Feature Importance" section displays a horizontal bar chart showing the contribution of various features to the prediction target "PRICE_RANGE". The chart shows that "NewFeature_10" has the highest contribution at 0.59, followed by "ram" at 0.34, and "NewFeature_2" at 0.05.

Step-40. Click on Save as on the top right corner. Select Model and then click on Create.

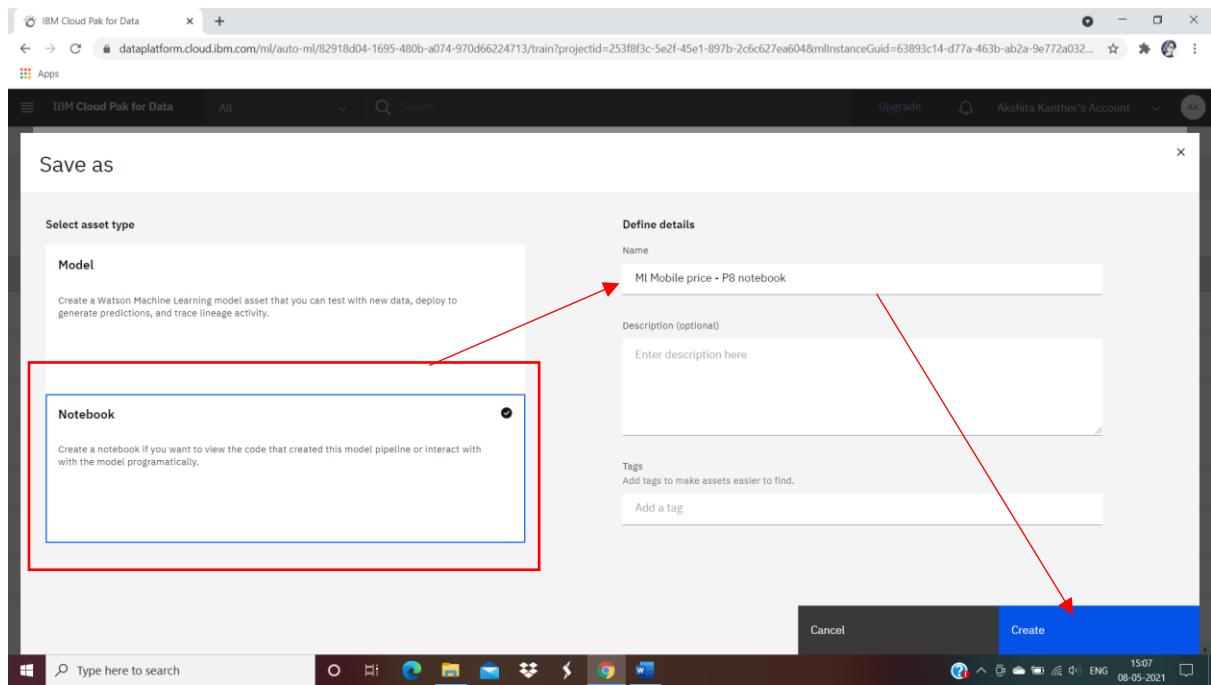




Step-41. A dialog box will pop up, it will display a message – “Saved the model successfully”. Close the pop up and again click on Save as.



Step-42. Select Notebook and click on Create.



Pipeline 8

Rank: 1 Holdout Accuracy (Optimized): 0.955 Algorithm: RandomForestClassifier Enhancements: HPO-1, FE, HPO-2 Save as

EVALUATION

Model Evaluation

Precision Recall Curve Threshold Chart ROC Curve

MODEL VIEWER

Model Information Feature Transformations Feature Importance

View: Model Evaluation

Save as notebook

The notebook is being created. Please wait a moment...

	Holdout Score	Cross Validation Score
Macro Precision	0.955	0.937
Accuracy	0.955	0.937
Macro Recall	0.955	0.937

Step-43. Click on project name on the top (breadcrumb link on the top). And scroll down.

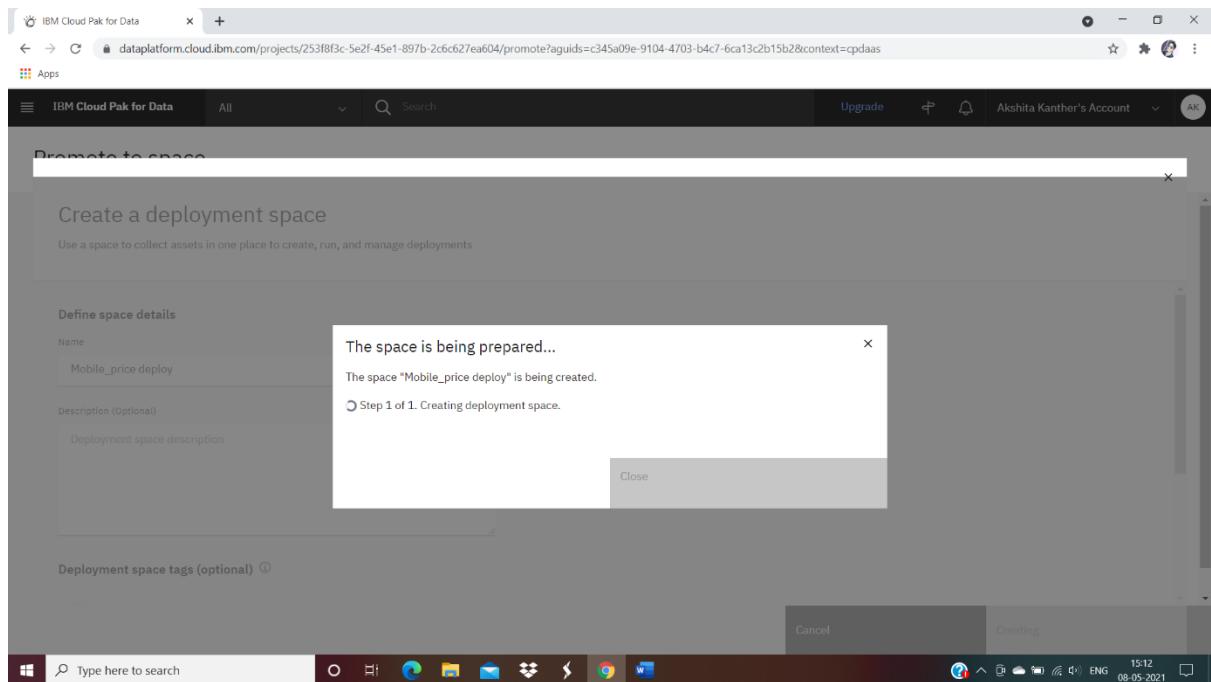
Then click on the model name under Models.

Step-44. Click on Promote to deployment space.

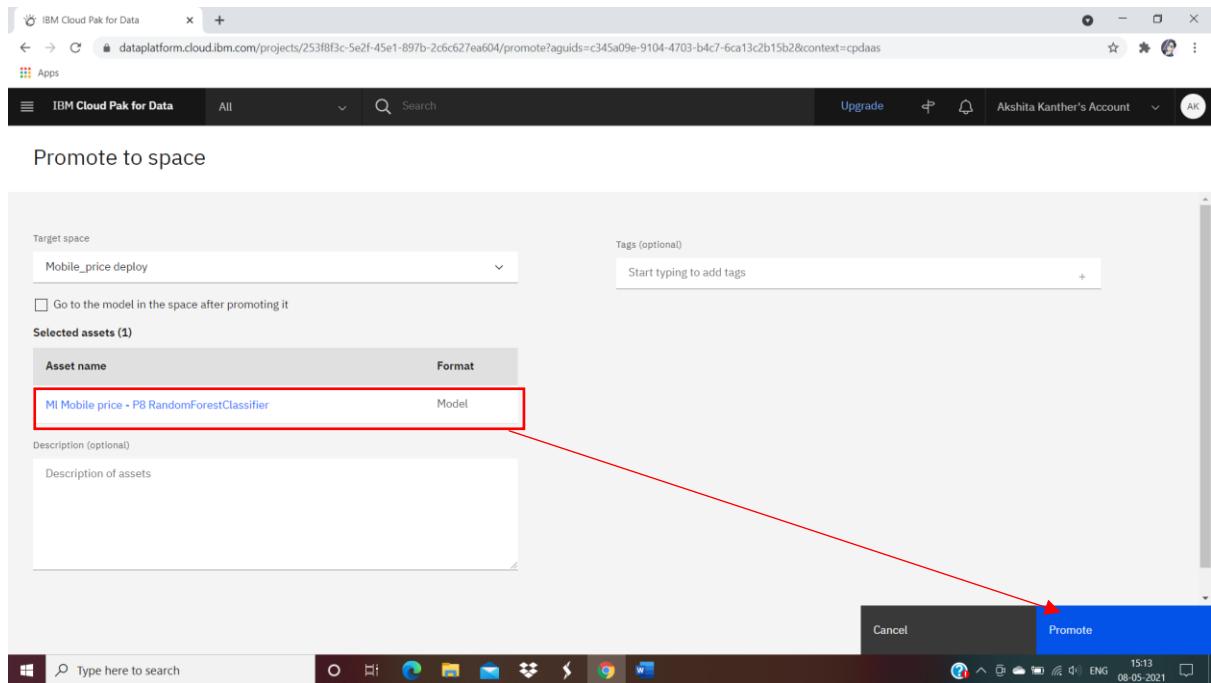
The screenshot shows the IBM Cloud Pak for Data interface. On the left, there's a navigation bar with 'IBM Cloud Pak for Data' and 'All'. Below it, 'My Projects' lists 'Mobile Price Classification / ML Mobile price - P8 RandomForestClassifier'. A red box highlights the 'Promote to deployment space' button. To the right, a detailed view of the model 'ML Mobile price - P8 RandomForestClassifier' is shown, including its description, creation date (May 8, 2021), type (wml-hybrid_0.1), model ID, software specification, and hybrid pipeline. A 'Tags' section allows adding tags for easier finding.

Step-45. Enter the name for deployment under Define space details and then click on Create.

The screenshot shows the 'Promote to space' dialog box. It has a title 'Create a deployment space' and a subtitle 'Use a space to collect assets in one place to create, run, and manage deployments'. Under 'Define space details', the 'Name' field is filled with 'Mobile_price_deploy', which is highlighted with a red box. A red arrow points from this field to the 'Create' button at the bottom right of the dialog. The 'Create' button is also highlighted with a red box.



Step-46. Select a Target space from the dropdown box and then click on Promote.



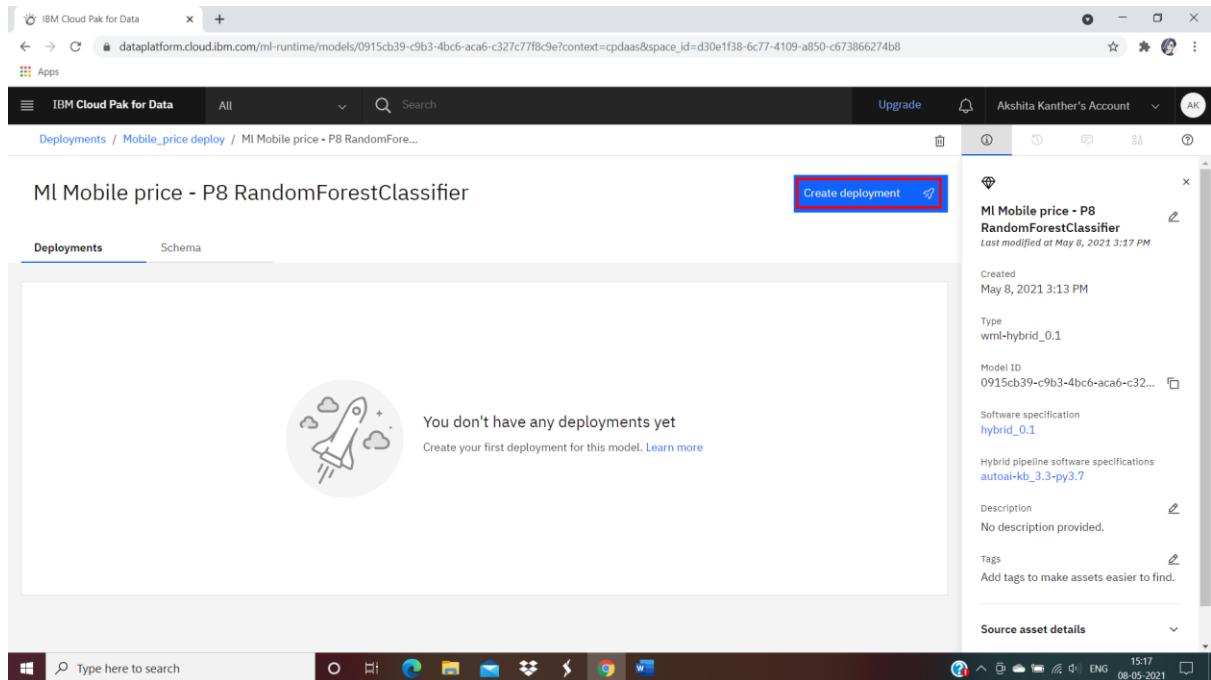
Step-47. A dialog box containing message of “Successfully promoted” will be displayed. Click on deployment space link in that message of the dialog box.

The screenshot shows the 'Assets' tab in the IBM Cloud Pak for Data interface. A prominent message box at the top right says: 'Successfully promoted MI Mobile price - P8 RandomForestClassifier to the associated deployment space. Go to the deployment space [1] to prepare the assets for deployment.' Below this, there are sections for 'Data assets', 'AutoAI experiments', and 'Notebooks'. The 'Data assets' section lists a single CSV file named 'mobile_price.csv'. The 'AutoAI experiments' section shows one experiment named 'MI Mobile price' which is completed. The 'Notebooks' section has a search bar and a list of notebook icons.

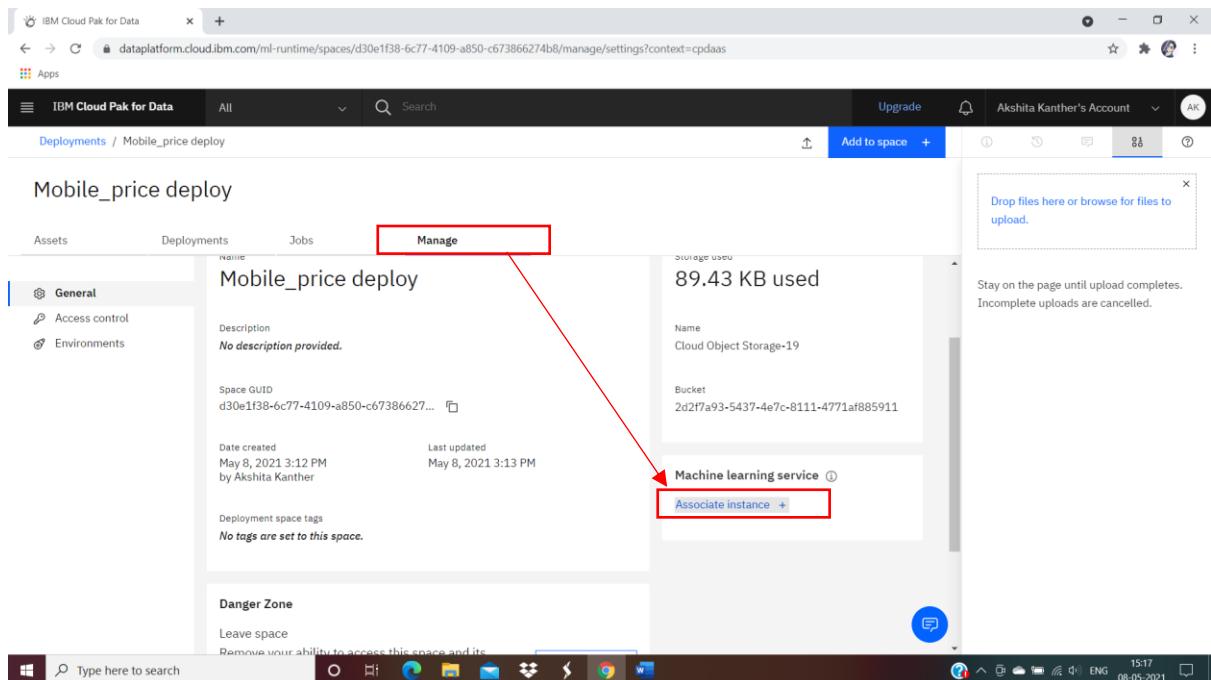
Step-48. Click on model name under Models (1) in the Assets tab.

The screenshot shows the 'Assets' tab in the IBM Cloud Pak for Data interface, specifically for the 'Mobile_price deploy' deployment. It displays a table of models with one entry: 'MI Mobile price - P8 RandomForestClassifier'. This entry is highlighted with a red box. To the right of the table, a message says: 'Stay on the page until upload completes. Incomplete uploads are cancelled.' The interface includes tabs for 'Assets', 'Deployments', 'Jobs', and 'Manage', along with a search bar and a file upload area.

Step-49. Click on Create deployment.



Step-50. Click on Manage. And then click on Associate instance + .



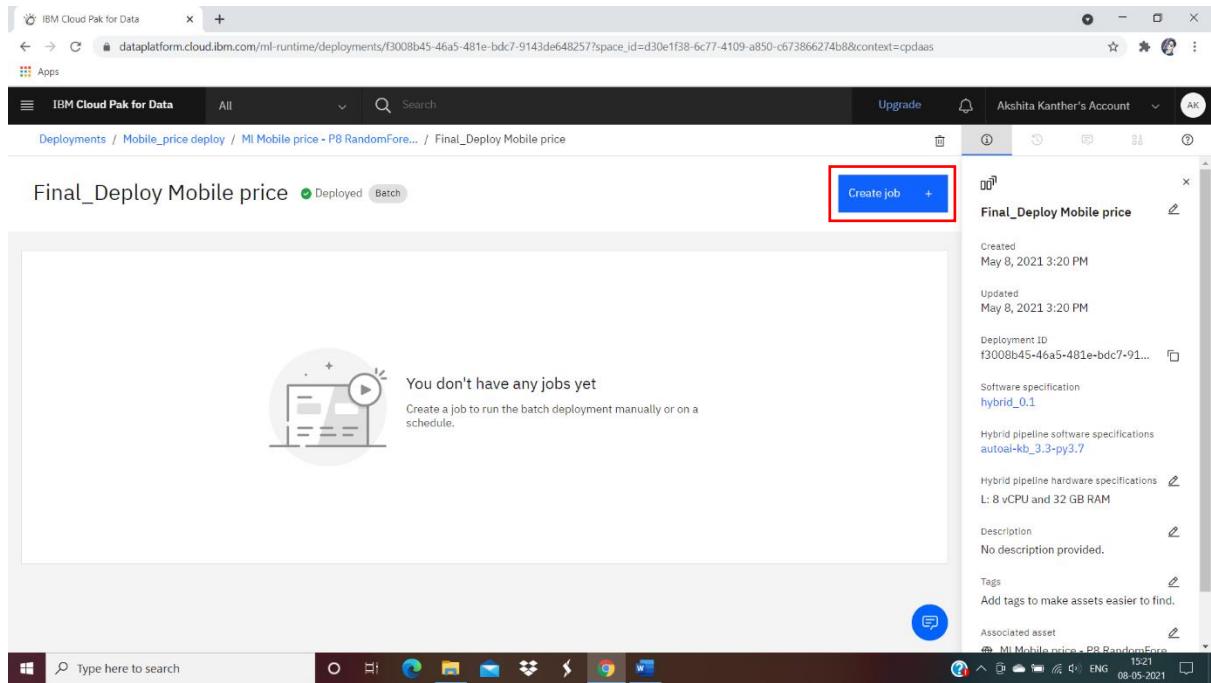
Step-51. After Machine learning service has been associated, switch to Deployments tab.

The screenshot shows the IBM Cloud Pak for Data interface. The top navigation bar includes 'IBM Cloud Pak for Data', 'Search', 'Upgrade', 'Akshita Kanther's Account', and a user icon. Below the navigation is a toolbar with 'Deployments / Mobile_price deploy', 'Add to space', and other icons. The main content area displays the 'Mobile_price deploy' configuration. The 'Deployments' tab is selected, highlighted with a red box. The deployment name is 'Mobile_price deploy'. The 'General' section shows a 'Description' field with the placeholder 'No description provided.' and a 'Space GUID' field containing 'd30e1f38-6c77-4109-a850-c673866274b8'. It also shows 'Date created' as May 8, 2021, 3:12 PM, and 'Last updated' as May 8, 2021, 3:18 PM, both by 'Akshita Kanther'. A 'Danger Zone' section contains a 'Leave space' link. To the right, there are sections for 'Storage usage' (89.43 KB used), 'Cloud Object Storage-19' (Bucket: 2d2f7a93-5437-4e7c-8111-4771af885911), and 'Machine learning service' (Machine Learning-ph). A large dashed box on the right allows file uploads with the instruction 'Drop files here or browse for files to upload.' A note says 'Stay on the page until upload completes. Incomplete uploads are cancelled.' The bottom of the screen shows a Windows taskbar with various icons and system status.

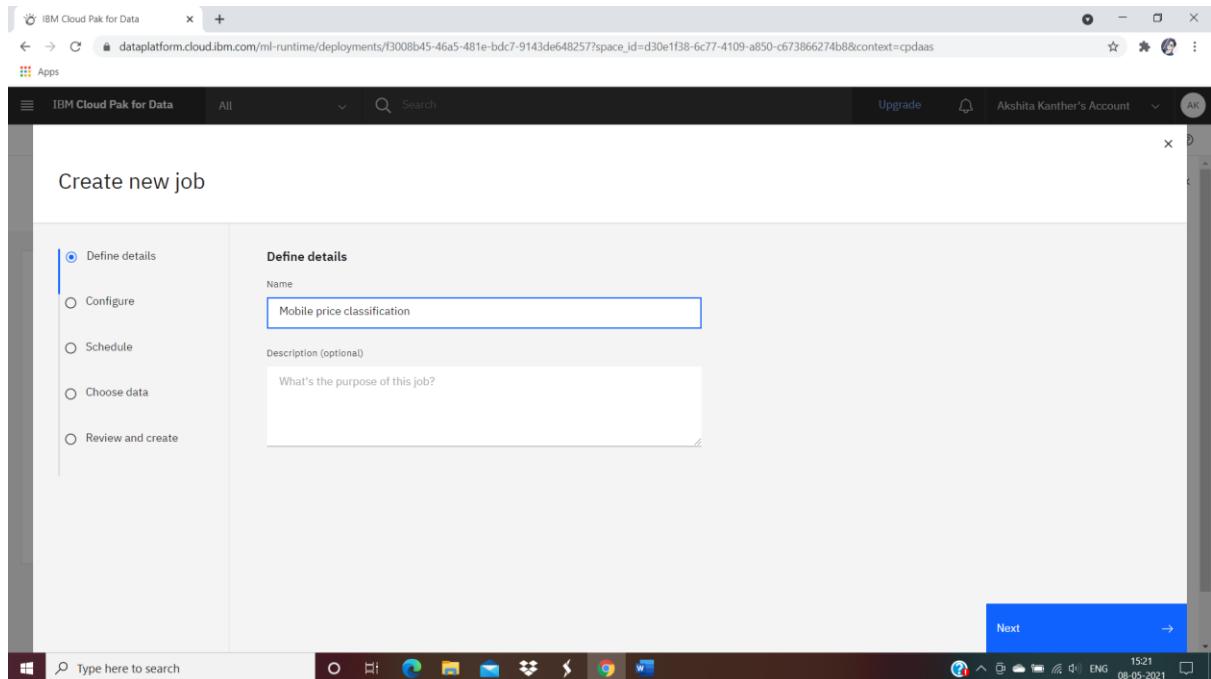
Step-52. Select Batch as Deployment type and then give a name and then click on Create.

The screenshot shows the 'Create a deployment' dialog box. At the top, it says 'Create a deployment'. Below that, there is a section for 'Associated asset' with 'MI Mobile price - P8 RandomForestClassifier'. Under 'Deployment type', there are two options: 'Online' (Run the model on data in real-time, as data is received by a web service) and 'Batch' (Run the model against data as a batch process). The 'Batch' option is selected and highlighted with a red box. A red arrow points from the 'Name' input field to the 'Create' button. The 'Name' field contains 'Final_Deploy Mobile price'. The 'Description' field has 'Deployment description'. At the bottom right of the dialog are 'Cancel' and 'Create' buttons, with a red arrow pointing to the 'Create' button.

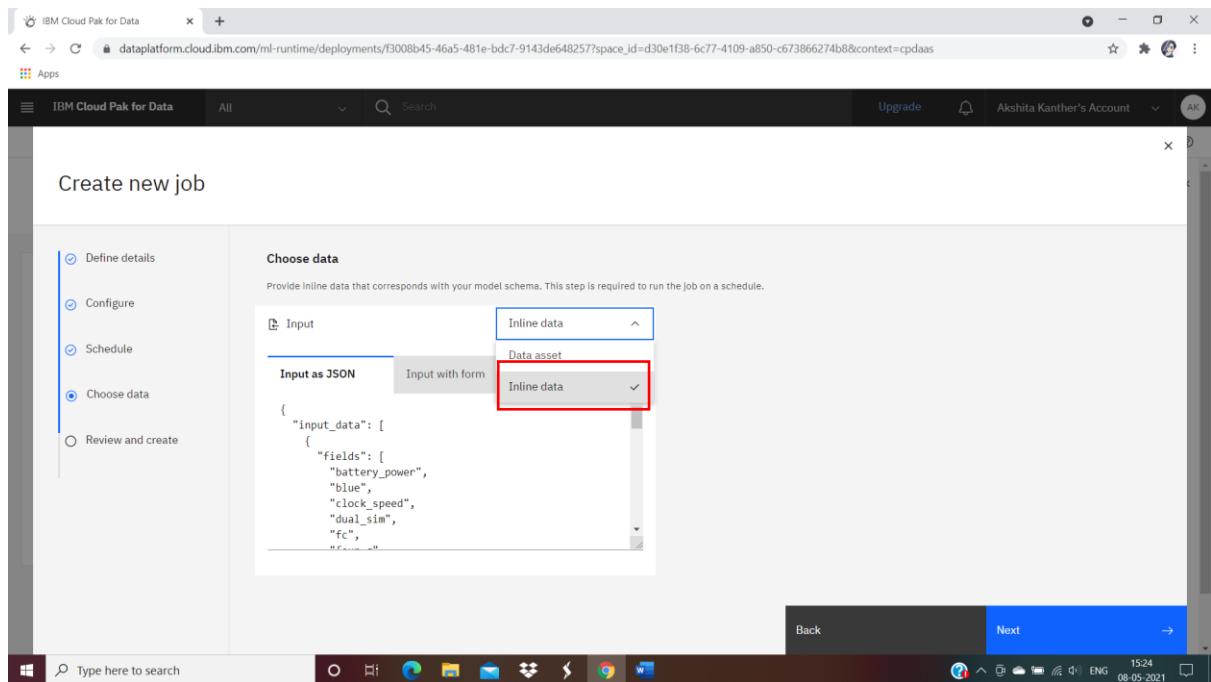
Step-53. Click on Create job + to create a job as we don't have any jobs as of now.



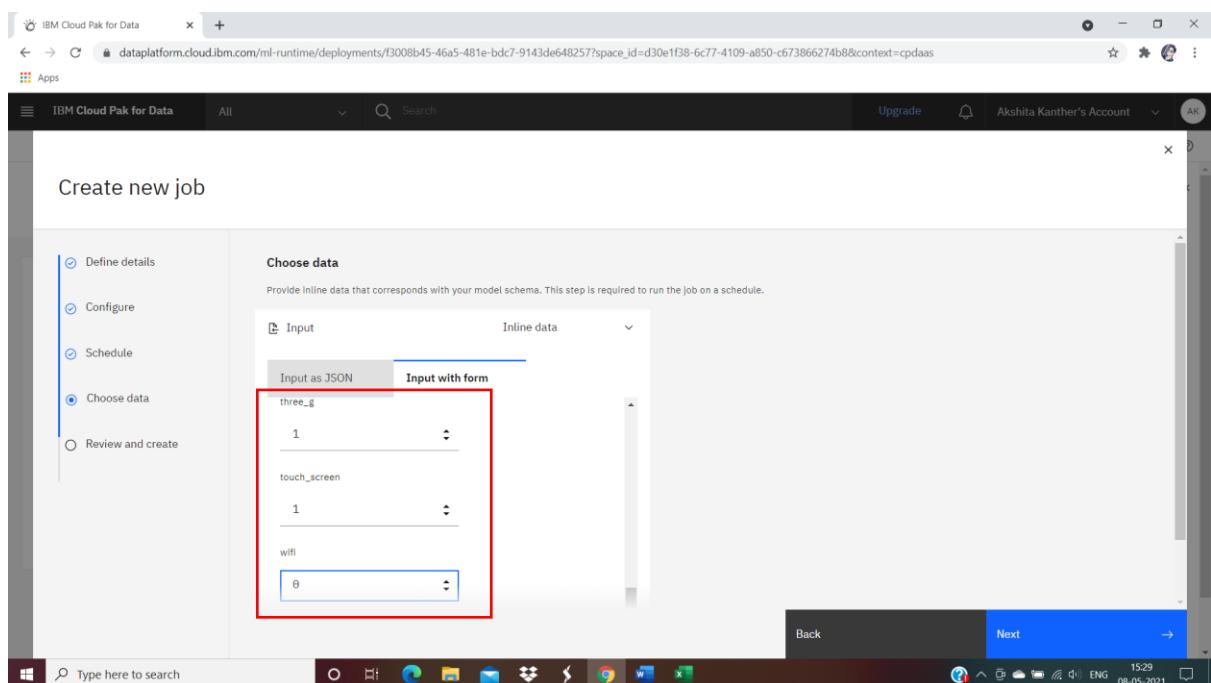
Step-54. Enter job name under Name column of Define details and then click on Next repeatedly for Define details, Configure and Schedule.



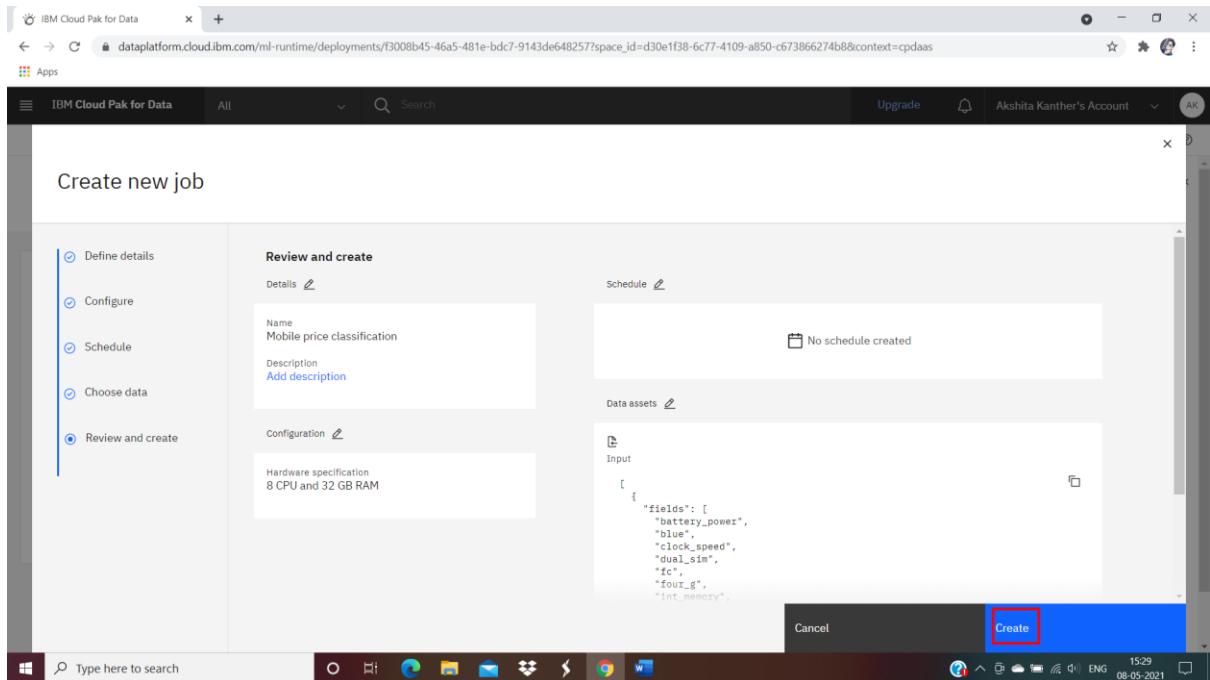
Step-55. Select Inline data for Input in Choose data.



Step-56. Select Input with form and enter input values of a row for prediction and click on Next.



Step-57. Click on Create.



Step-58. Click on job name (Mobile price classification).

Job name	Date created	Created by
Mobile price classification	May 8, 2021 3:29 PM	Akshita Kanther

Step-59. Repeat steps 56 to 58 for different input rows from sample data. When status of all the inputs for the job is completed under Runs then click on all the links one by one under Start Time and check the output.

IBM Cloud Pak for Data

dataplatform.cloud.ibm.com/ml-runtime/jobs/6ac850ea-2d3c-4f01-83ae-752d5c5f47a0?space_id=d30e1f38-6c77-4109-a850-c673866274b8&context=cpdaas

Associated Asset: Final Deploy Mobile price

Deployment Job Definition ID: 5a2426fc-41fb-43e4-9da6-e21d3040ade2

Job ID: 6ac850ea-2d3c-4f01-83ae-752d5c5f47a0

Scheduled to run: No schedule created

Environment definition: 8 CPU and 32 GB RAM

Input: {"input_data": [{"file..."]}

Output: No output asset for inline input data

Runs (4)

Start Time	Status	Duration	Started By
May 8, 2021 3:44 PM	Completed	4 seconds	Akshita Kanther
May 8, 2021 3:40 PM	Completed	5 seconds	Akshita Kanther
May 8, 2021 3:37 PM	Completed	5 seconds	Akshita Kanther
May 8, 2021 3:29 PM	Completed	5 seconds	Akshita Kanther

IBM Cloud Pak for Data

dataplatform.cloud.ibm.com/ml-runtime/jobs/runs?space_id=d30e1f38-6c77-4109-a850-c673866274b8&job_id=6ac850ea-2d3c-4f01-83ae-752d5c5f47a0&jobrun_id=c4d796a1-e673-482d-8519-d098...

Associated Asset: Mobile price classification

Deployment Job ID: 5a2426fc-41fb-43e4-9da6-e21d3040ade2

Job Run ID: c4d796a1-e673-482d-8519-d09801777612

About this run: Completed

Run details: Duration (seconds): 5, Started by: Akshita Kanther, Associated job: Mobile price classification

Job run details: May 08, 2021 3:29:42 PM

```
{
  "run": {
    "id": "c4d796a1-e673-482d-8519-d09801777612",
    "status": {
      "completed_at": "2021-05-08T10:00:04.256Z",
      "running_at": "2021-05-08T10:00:00.178Z",
      "state": "completed"
    },
    "predictions": [
      {
        "fields": [
          "prediction",
          "probability"
        ],
        "values": [
          [
            1,
            [
              0.0007027863093084525,
              0.0864659832153391,
              0.11863229047500064
            ]
          ]
        ]
      }
    ],
    "duration": 5
  }
}
```

IBM Cloud Pak for Data

May 08, 2021 3:37:08 PM

About this run

Completed

Run details

Duration (seconds): 5
Started by: Akshita Kanther
Associated job: Mobile price classification

Deployment Job ID: 5a2426fc-41fb-43e4-9da6-e21d3040ade2

Job Run ID: b10f1bfd-0551-4e2d-b6b1-3376923741b7

```
{
  "predictions": [
    {
      "fields": [
        "prediction",
        "probability"
      ],
      "values": [
        [
          0,
          [
            0.9380659949966471,
            0.061934805003352964,
            0,
            0
          ]
        ]
      ]
    },
    {
      "status": {
        "completed_at": "2021-05-08T10:07:30.224Z",
        "running_at": "2021-05-08T10:07:26.200Z",
        "state": "completed"
      }
    }
  ]
}
```

Show less ▾

Mobile price classifi...txt

IBM Cloud Pak for Data

May 08, 2021 3:40:23 PM

About this run

Completed

Run details

Duration (seconds): 5
Started by: Akshita Kanther
Associated job: Mobile price classification

Deployment Job ID: 5a2426fc-41fb-43e4-9da6-e21d3040ade2

Job Run ID: 71af7d2a-c8a8-4b5f-a3f2-572afd73f6f7

```
{
  "predictions": [
    {
      "fields": [
        "prediction",
        "probability"
      ],
      "values": [
        [
          2,
          [
            0,
            0.027229115323485664,
            0.966191571489495,
            0.006579313187019574
          ]
        ]
      ]
    },
    {
      "status": {
        "completed_at": "2021-05-08T10:44:783Z",
        "running_at": "2021-05-08T10:40:614Z",
        "state": "completed"
      }
    }
  ]
}
```

Show less ▾

Mobile price classifi...txt

May 08, 2021 3:44:24 PM

About this run

Completed

Run details

Duration (seconds): 4
Started by: Akshita Kanther
Associated job: [Mobile price classification](#)

Deployment Job ID: 5a2426fc-41fb-43e4-9da6-e21d3040ade2

Job Run ID: 6a499c89-9c9b-408e-be51-26f10bdbbe2a

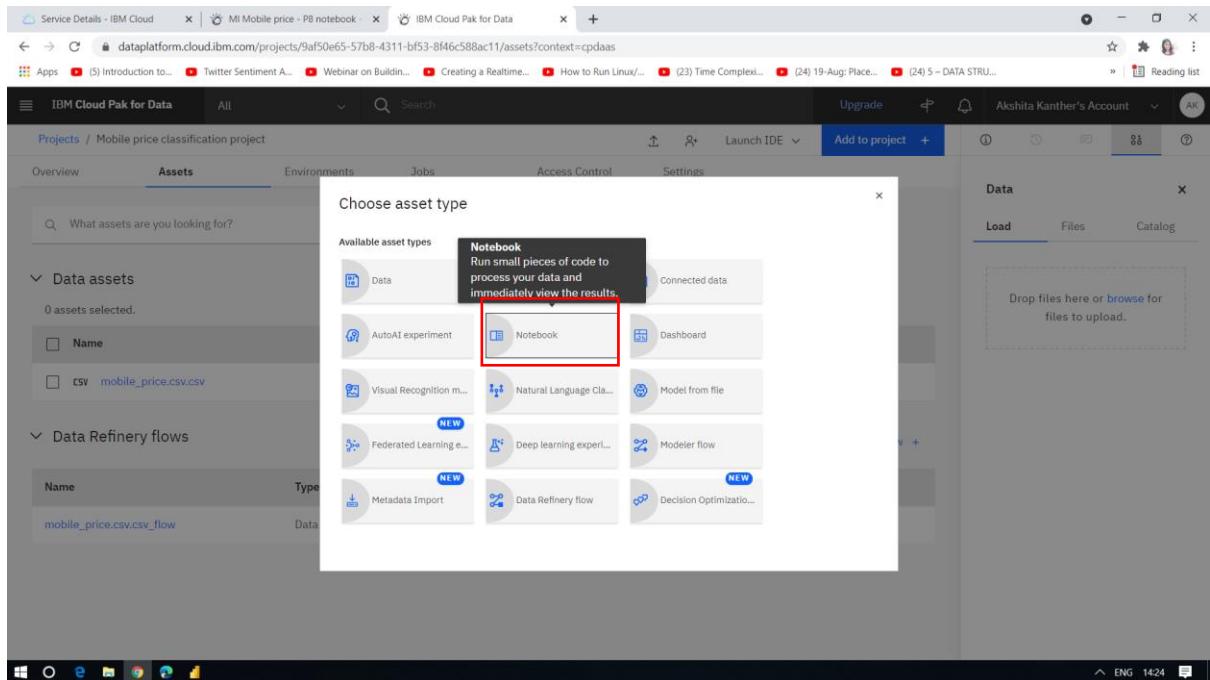
```
{
  "predictions": [
    {
      "fields": [
        "prediction",
        "probability"
      ],
      "values": [
        [
          3,
          [
            0,
            0,
            0.00033084617041718483,
            0.9996699538295827
          ]
        ]
      ]
    },
    {
      "status": {
        "completed_at": "2021-05-08T10:14:42.121Z",
        "running_at": "2021-05-08T10:14:38.785Z",
        "state": "completed"
      }
    }
  ]
}
```

Mobile price classifi....txt | Mobile price classifi....txt | Mobile price classifi....txt | Mobile price classifi....txt | Show all

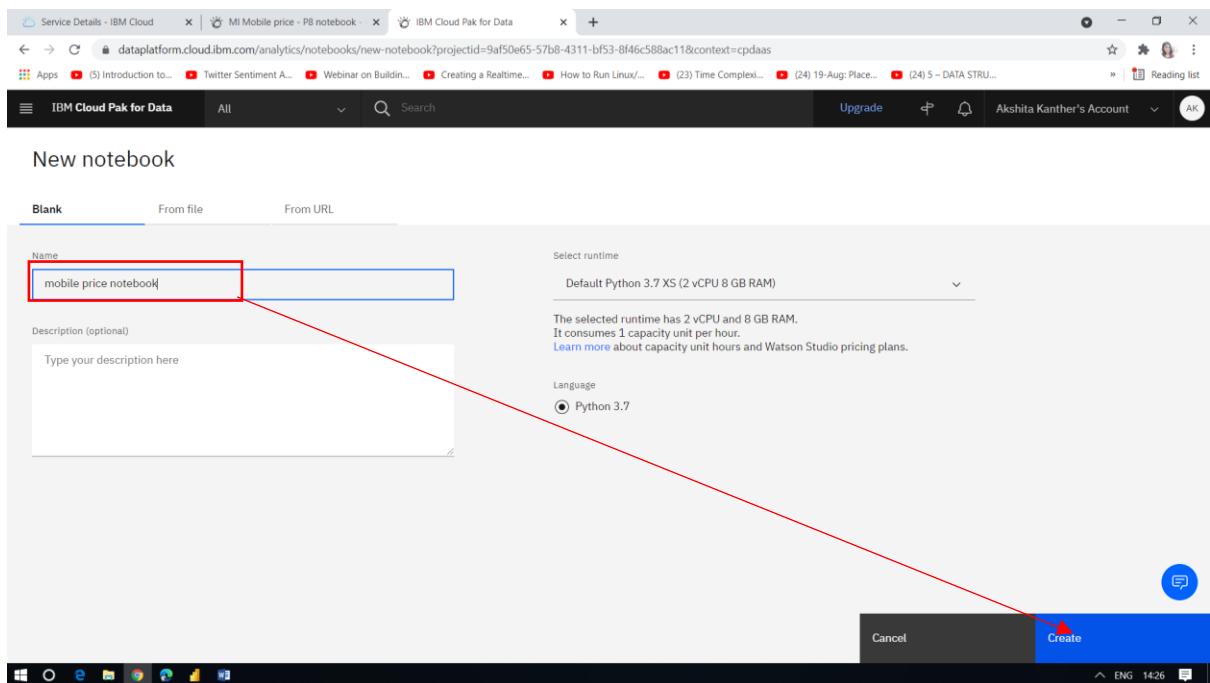
We can check predictions and probability of outputs of all the runs. After verifying it from sample data, we can conclude that since our outputs are predicted correctly with high probability for different classes, therefore our model will perform well on unknown data also.

PYTHON NOTEBOOK-

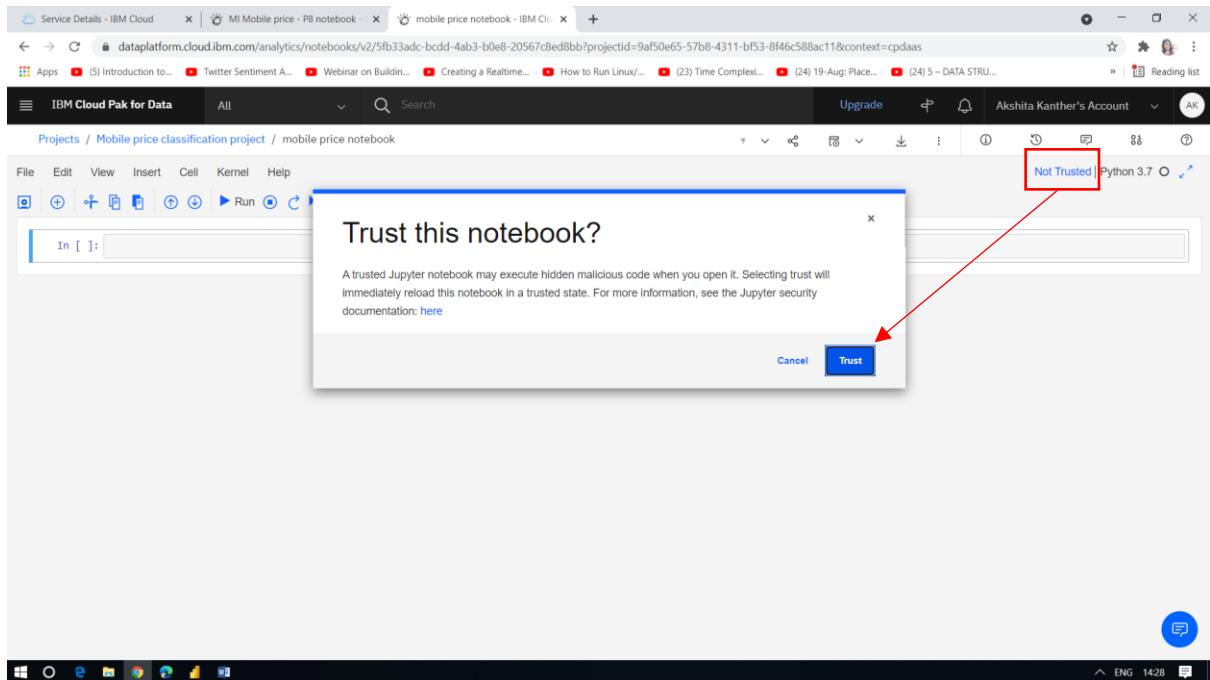
step-1. Select Add to project from Assets tab from a mobile price project. And click on Notebook.



step-2. Enter a name for the notebook and click on Create.



step-3. Click on Not Trusted on the top and then click on Trust.



step-4. Click on the name of the notebook under Notebooks.

Name	Type	Created by	Last modified
mobile price notebook	Data Asset	Akshita Kanther	May 14, 2021, 11:35 PM

step-5. Now we will write the code : firstly import the libraries , read the dataset and perform Exploratory Data Analysis.

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:

```
df = pd.read_csv('https://raw.githubusercontent.com/abhinav23dixit/Mobile-Price-Classification/master/train.csv')
```

In [3]:

```
df.head()
```

Out[3]:

#	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	px_height	px_width	ram	sc_h	sc_w	ta
0	842	0	2.2	0	1	0	7	0.6	188	2	... 20	756	2549	9	7	1E
1	1021	1	0.5	1	0	1	53	0.7	136	3	... 905	1988	2631	17	3	7
2	563	1	0.5	1	2	1	41	0.9	145	5	... 1263	1716	2603	11	2	9
3	615	1	2.5	0	0	0	10	0.8	131	6	... 1216	1786	2769	16	8	11
4	1821	1	1.2	0	13	1	44	0.6	141	2	... 1208	1212	1411	8	2	1E

5 rows × 21 columns

In [4]:

```
df.info()
```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2000 entries, 0 to 1999
Data columns (total 21 columns):
 # Column Non-Null Count Dtype

 0 battery_power 2000 non-null int64
 1 blue 2000 non-null int64
 2 clock_speed 2000 non-null float64
 3 dual_sim 2000 non-null int64
 4 fc 2000 non-null int64
 5 four_g 2000 non-null int64
 6 int_memory 2000 non-null int64
 7 m_dep 2000 non-null float64
 8 mobile_wt 2000 non-null int64
 9 n_cores 2000 non-null int64
 10 pc 2000 non-null int64
 11 px_height 2000 non-null int64
 12 px_width 2000 non-null int64
 13 ram 2000 non-null int64
 14 sc_h 2000 non-null int64
 15 sc_w 2000 non-null int64
 16 talk_time 2000 non-null int64
 17 three_g 2000 non-null int64
 18 touch_screen 2000 non-null int64
 19 wifi 2000 non-null int64
 20 price_range 2000 non-null int64
 dtypes: float64(2), int64(19)
 memory usage: 328.2 KB

In [5]:

```
df.describe().transpose()
```

Out[5]:

	count	mean	std	min	25%	50%	75%	max
battery_power	2000.0	1238.51850	439.418206	501.0	851.75	1226.0	1615.25	1998.0
blue	2000.0	0.49500	0.500100	0.0	0.00	0.0	1.00	1.0
clock_speed	2000.0	1.52225	0.816004	0.5	0.70	1.5	2.20	3.0
dual_sim	2000.0	0.50950	0.500035	0.0	0.00	1.0	1.00	1.0

step-6. Change the column names of required columns with the rename column.

The screenshot shows a Jupyter Notebook interface within the IBM Cloud Pak for Data environment. The notebook displays a table of mobile phone specifications and some initial data processing code.

Table Data:

	dual_sim	2000.0	0.50950	0.500035	0.0	0.00	1.0	1.00	1.0
fc	2000.0	4.30950	4.341444	0.0	1.00	3.0	7.00	19.0	
four_g	2000.0	0.52150	0.499662	0.0	0.00	1.0	1.00	1.0	
int_memory	2000.0	32.04650	18.145715	2.0	16.00	32.0	48.00	64.0	
m_dep	2000.0	0.50175	0.288416	0.1	0.20	0.5	0.80	1.0	
mobile_wt	2000.0	140.24900	35.399655	80.0	109.00	141.0	170.00	200.0	
n_cores	2000.0	4.52050	2.287837	1.0	3.00	4.0	7.00	8.0	
pc	2000.0	9.91650	6.064315	0.0	5.00	10.0	15.00	20.0	
px_height	2000.0	645.10800	443.780811	0.0	282.75	564.0	947.25	1960.0	
px_width	2000.0	1251.51550	432.199447	500.0	874.75	1247.0	1633.00	1998.0	
ram	2000.0	2124.21300	1084.732044	256.0	1207.50	2146.5	3064.50	3998.0	
sc_h	2000.0	12.30650	4.213245	5.0	9.00	12.0	16.00	19.0	
sc_w	2000.0	5.76700	4.356398	0.0	2.00	5.0	9.00	18.0	
talk_time	2000.0	11.01100	5.463955	2.0	6.00	11.0	16.00	20.0	
three_g	2000.0	0.76150	0.426273	0.0	1.00	1.0	1.00	1.0	
touch_screen	2000.0	0.50300	0.500116	0.0	0.00	1.0	1.00	1.0	
wifi	2000.0	0.50700	0.500076	0.0	0.00	1.0	1.00	1.0	
price_range	2000.0	1.50000	1.118314	0.0	0.75	1.5	2.25	3.0	

Code Snippets:

```
In [7]: df.rename(columns ={'blue':'bluetooth', 'n_cores':'no_of_cores', 'fc':'front_camera', 'm_dep':'mobile_depth'}, inplace=True)
In [8]: df.columns
Out[8]: Index(['battery_power', 'bluetooth', 'clock_speed', 'dual_sim', 'front_camera',
   'four_g', 'int_memory', 'mobile_depth', 'mobile_wt', 'no_of_cores',
   'pc', 'px_height', 'px_width', 'ram', 'sc_h', 'sc_w', 'talk_time',
   'three_g', 'touch_screen', 'wifi', 'price_range'],
  dtype='object')
```

step-7. We will check the correlation of all the columns with the price_range column and sort it in ascending order. We can observe that the ram column is highly correlated with the price_range column. We will drop the columns which are not required for analysis.

The screenshot shows a Jupyter Notebook interface within the IBM Cloud Pak for Data environment. The notebook displays correlation calculations and the subsequent dropping of columns.

Code Snippets:

```
Out[8]: Index(['battery_power', 'bluetooth', 'clock_speed', 'dual_sim', 'front_camera',
   'four_g', 'int_memory', 'mobile_depth', 'mobile_wt', 'no_of_cores',
   'pc', 'px_height', 'px_width', 'ram', 'sc_h', 'sc_w', 'talk_time',
   'three_g', 'touch_screen', 'wifi', 'price_range'],
  dtype='object')

In [13]: df.corr()['price_range'].sort_values()
Out[13]:
touch_screen    -0.030411
mobile_wt       -0.030302
clock_speed     -0.000060
mobile_depth     0.000853
no_of_cores      0.004399
four_g          0.014772
dual_sim         0.017444
wifi             0.018785
bluetooth        0.020573
talk_time        0.021859
front_camera     0.021998
sc_h             0.022986
three_g          0.023611
pc               0.033599
sc_w             0.038711
int_memory       0.044435
px_height         0.148858
px_width          0.165818
battery_power     0.200723
ram              0.917046
price_range      1.000000
Name: price_range, dtype: float64

In [14]: df.drop('talk_time', axis=1, inplace=True)
In [17]: df.columns
Out[17]: Index(['battery_power', 'bluetooth', 'clock_speed', 'dual_sim', 'front_camera',
   'four_g', 'int_memory', 'mobile_depth', 'mobile_wt', 'no_of_cores',
   'pc', 'px_height', 'px_width', 'ram', 'sc_h', 'sc_w'],
  dtype='object')
```

The screenshot shows a Jupyter Notebook interface in a browser window. The notebook has several cells:

- In [17]:** A code cell showing the creation of a correlation matrix from a DataFrame.
- Out[17]:** The resulting correlation matrix.
- In [18]:** A code cell calling `sns.heatmap(df.corr())`.
- Out[18]:** The generated heatmap visualization.
- In [19]:** A code cell showing the import of `LogisticRegression` from `sklearn.linear_model`.
- In [20]:** A code cell showing the creation of a logistic regression model object `logmodel`.
- In [21]:** A code cell showing the import of `train_test_split` from `sklearn.model_selection`.

step-8. Now split the data , train and evaluate the model. We will create a logistic regression model as we want to predict the output which will be in different classes (0,1,2,3 : low price, medium price, high price, very high price).

The screenshot shows a Jupyter Notebook interface in a browser window. The notebook has several cells:

- In [23]:** Code to drop the `'price_range'` column and split the remaining data into training and testing sets.
- In [24]:** Code to fit the logistic regression model `logmodel` on the training data.
- Out[24]:** A warning message from scikit-learn indicating convergence issues with the solver.
- In [25]:** Code to make predictions on the test set using `logmodel.predict(X_test)`.
- In [26]:** Code to import classification metrics from `sklearn.metrics`.
- In [27]:** Code to print the confusion matrix and classification report.
- Out[27]:** The printed confusion matrix and classification report data.

The screenshot shows two separate sessions of a Jupyter-style notebook within the IBM Cloud Pak for Data interface.

Session 1 (Top):

```
In [27]: print(confusion_matrix(y_test,y_pred))
print('\n')
print(classification_report(y_test,y_pred))

[[73 19  2  0]
 [20 56 22  4]
 [ 0 29 46 45]
 [ 0  0 17 67]]
```

	precision	recall	f1-score	support
0	0.78	0.78	0.78	94
1	0.54	0.55	0.54	102
2	0.53	0.38	0.44	120
3	0.58	0.80	0.67	84

	accuracy	macro avg	weighted avg
accuracy	0.60	0.61	0.60
macro avg	0.61	0.63	0.61
weighted avg	0.60	0.60	0.60

In [28]: X = df[['ram']]
y = df['price_range']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=101)

In [29]: logmodel.fit(X_train,y_train)

Out[29]: LogisticRegression()

In [30]: y_pred = logmodel.predict(X_test)

In [31]: print(confusion_matrix(y_test,y_pred))
print('\n')
print(classification_report(y_test,y_pred))

Session 2 (Bottom):

Out[31]: LogisticRegression()

In [32]: y_pred = logmodel.predict(X_test)

In [33]: print(confusion_matrix(y_test,y_pred))
print('\n')
print(classification_report(y_test,y_pred))

[[76 18 0 0]
 [17 62 23 0]
 [0 29 80 20]
 [0 0 14 70]]

	precision	recall	f1-score	support
0	0.82	0.81	0.81	94
1	0.62	0.61	0.61	102
2	0.68	0.67	0.68	120
3	0.78	0.83	0.80	84

	accuracy	macro avg	weighted avg
accuracy	0.72	0.72	0.72
macro avg	0.72	0.73	0.73
weighted avg	0.72	0.72	0.72

We have created two models. In the first model we have predicted output on the basis of all the columns except price_range column ,the accuracy and F1-score is not very good because price_range is weakly correlated with all other columns except ram , in the second model, we have predicted price range on the basis on only ram column because they were highly correlated. And we have observed significant increase in the F1-score and accuracy of the model.

CONCLUSION

Watson is IBM's portfolio of business-ready tools, applications and solutions, designed to reduce the costs and hurdles of AI adoption while optimizing outcomes and responsible use of AI. It operationalizes AI and transforms how work gets done with our proven capabilities and experiences .

After conducting the above project, we learned about how to work with the IBM Data Refinery tools, Visual Recognition service, Machine Learning model(Auto AI experiment) and Python Notebook on IBM cloud platform using IBM Watson Service . We were able to train the model for our dataset according to the requirement of the customer. We came to know how to work on a data set, how to classify the images and train the model according to classes needed and show results according to that classes. We came across different machine learning algorithms, we implemented Logistic Regression algorithm in Python Notebook as we had to predict the output in the form of classes/ categories. In IBM Auto AI experiment different pipelines were created , we evaluated different models on the basis of F1-score , Accuracy, Precision, Recall and various other model evaluation measures.