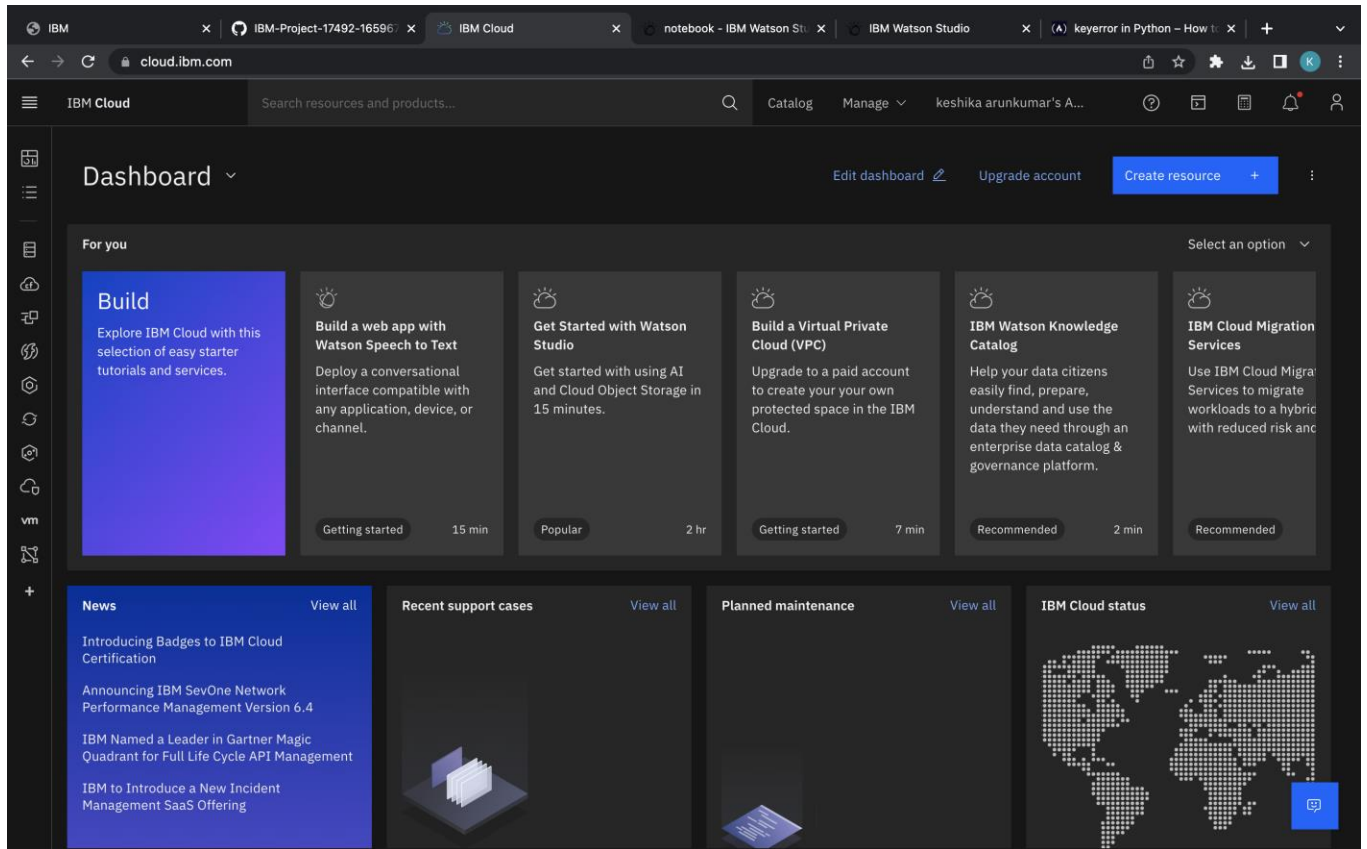


SCREENSHOTS

1. IBM cloud dashboard



2. Notebook.ipynb (model training)

The screenshot shows a web browser window with multiple tabs, including 'notebook - IBM Watson Studio'. The address bar shows the URL: `datapatform.cloud.ibm.com/analytics/notebooks/v2/fd757821-19f5-46f1-b8d0-225db55a93aa?projectId=ac053c85-7d36-4d15-839f-e9736bc40bfd&conte...`. The IBM Watson Studio interface is visible, with a search bar and a 'Buy' button. The notebook title is 'flight delay prediction / notebook'. The notebook content includes a title 'DEVELOPING A FLIGHT DELAY PREDICTION MODEL USING MACHINE LEARNING' and two code cells. The first cell, 'Data Preprocessing', imports various libraries including sys, numpy, pandas, seaborn, pickle, matplotlib, sklearn, and metrics. The second cell, 'Importing Data', imports os, types, pandas, and boto3, and defines an iterator function.

DEVELOPING A FLIGHT DELAY PREDICTION MODEL USING MACHINE LEARNING

Data Preprocessing

```
In [1]: # importing the required libraries
import sys
import numpy as np
import pandas as pd
import seaborn as sns
import pickle
import matplotlib inline
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
import sklearn.metrics as metrics
import warnings
warnings.filterwarnings('ignore')
```

Importing Data

```
In [2]:
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def iter (self): return 0
```

3. API key generation

IBM Watson Studio interface showing the deployment of a model named **flightdelay**.

The interface includes a sidebar with the deployment details for **flightdelay**, including its creation and update dates (Nov 19, 2022, 11:15 PM), deployment ID (**f26e6fbe-bb80-4bed-8366-278db09a3e06**), software specification (**runtime-22.1-py3.9**), and associated asset (**7aaa81a8-e0f7-4648-a9ab-28e4e...**).

The main content area displays the **API reference** for the deployment, showing the **Direct link** (Endpoint) and **Code snippets** for various languages (cURL, Java, JavaScript, Python, Scala).

The **Direct link** section shows the endpoint URL: `https://us-south.ml.cloud.ibm.com/ml/v4/deployments/f26e6fbe-bb80-4bed-8366-278db09a3e06/predict`, and the authentication method: **Bearer <token>** (IAM).

The **Code snippets** section provides a cURL example for making a prediction request:

```
# NOTE: you must set $API_KEY below using information retrieved from your IBM Cloud account.

curl --insecure -X POST --header "Content-Type: application/x-www-form-urlencoded" --header "Accept: application/json" \
  --data-urlencode "grant_type=urn:ibm:params:oauth:grant-type:apikey" \
  --data-urlencode "apikey=$API_KEY" "https://iam.cloud.ibm.com/identity/token"

# the above CURL request will return an auth token that you will use as $IAM_TOKEN in the scoring request below
# TODO: manually define and pass values to be scored below
curl -X POST --header "Content-Type: application/json" --header "Accept: application/json" --header "Authorization: \
  Bearer $IAM_TOKEN" -d '{"input_data": [{"fields": [$ARRAY_OF_INPUT_FIELDS], "values": [$ARRAY_OF_VALUES_TO_BE_SCORED, \
    $ANOTHER_ARRAY_OF_VALUES_TO_BE_SCORED]]}' "https://us-south.ml.cloud.ibm.com/ml/v4/deployments/f26e6fbe-bb80-4bed-8366-278db09a3e06/predict"
```

4. IBM App deployment

Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

app_ibm.py x app.py

Users > keshikaarunkumar > Desktop > ntp 2 > app > app_ibm.py

```
1 import numpy as np
2 import os
3 from flask import Flask, request, jsonify, render_template,json
4 import pickle
5 import requests
6
7 # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
8 API_KEY = "B_EVqatLl8m6PwfFYDpCpevQdnFSa0_NrmyPks-lUcUY"
9 token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
10 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
11 mltoken = token_response.json()["access_token"]
12
13 header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
14 app = Flask(__name__)
15 model = pickle.load(open('rfmodel.pkl', 'rb'))
16
17 @app.route('/')
18 def home():
19     return render_template('mainpage.html')
20
21 @app.route('/predict',methods=['POST'])
22 def predict():
23     '''
24     For rendering results on HTML GUI
25     '''
26     sm=[6,7,8]
27     wt=[9,10,11]
28     sp=[12,1,2,3]
29     fl=[4,5]
30     farr= [int(x) for x in request.form.values()]
31     if farr[1] in sm:
32         farr.append(0)
33     elif farr[1] in wt:
34         farr.append(1)
35     elif farr[1] in sp:
36         farr.append(2)
37     else:
38         farr.append(3)
39     farr += [1, 2, 3, 4]
40     final_features=[int(x) for x in farr]
41     print(final_features)
42     payload_scoring = {"input_data": [{"fields": [{"QUARTER", 'MONTH', 'DAY_OF_MONTH', 'DAY_OF_WEEK', 'FL_NUM', 'ORIGIN', 'DEST', 'CRS_DEP_TIME.1', 'CRS_ARR_TIME.1', 'CRS_ELA
43
44     response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/f26e6fbc-bb80-4bed-8366-278db09a3e06/predictions?version=2022-11-19', json=
```

Ln 48, Col 32 Spaces: 4 UTF-8 CRLF Python