

OUTPUT OF DEPLOYMENT OF MODEL ON IBM CLOUD:

Deployments / pa_deploy_space / analysis_model /

analysis_deploy_model Deployed Online

API reference Test

Endpoint Bearer token

<https://eu-gb.ml.cloud.ibm.com/ml/v4/deployments/f4a6cc62-cd88-47a3-af62-6a940301e0> IAM

Code snippets

cURL Java JavaScript Python Scala

```
import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "your API key"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = [{"input_data": [{"fields": [array_of_input_fields], "values": [array_of_values_to_be

response_scoring = requests.post('https://eu-gb.ml.cloud.ibm.com/ml/v4/deployments/f4a6cc62-cd88-47a3-
headers={'Authorization': 'Bearer ' + mltoken})
```

Created Nov 15, 2022, 8:07 PM

Updated Nov 15, 2022, 8:07 PM

Deployment ID f4a6cc62-cd88-47a3-af62-6a940301e0

Software specification runtime-22.1-py3.9

Copies 1

Serving name No serving name.

Description No description provided.

Tags Add tags to make assets easier to find.

Associated asset

Projects / Machine Learning Based Vehicle ... / VehiclePerformanceAnalysisModel

Importing Libraries

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.formula.api as smf
```

Importing Dataset

```
In [4]: import os, types
import pandas as pd
from botocore.client import Config
import boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = boto3.client(service_name='s3',
    ibm_api_key_id='tsHfz5SCBRth06l3yrkIVT15GkgPZb7U8HQ1772kE76',
    ibm_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.eu.cloud-object-storage.appdomain.cloud')

bucket = 'machinelearningbasedvehicleperfor-donotdelete-pr-u0yvtfjysrhov'
object_key = 'car performance.csv'
```

```
scoringendpoint.py > ...
1 import requests
2
3 # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
4 API_KEY = "YIJAXb1Vp23FVn8FxaMhFEeCIBjRwptpHaal7JNGzUTE"
5 token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
6 API_KEY, "grant_type": "urn:ibm:params:oauth:grant-type:apikey"})
7 mltoken = token_response.json()["access_token"]
8
9 header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
10
11 # NOTE: manually define and pass the array(s) of values to be scored in the next line
12 payload_scoring = {"input_data": [{"fields": [{"f0", "f1", "f2", "f3", "f4", "f5"}], "values": [[8,160,380,3504,82,1]]}]}
13
14 response_scoring = requests.post('https://eu-gb.ml.cloud.ibm.com/ml/v4/deployments/f4a6cc62-cd58-47a3-af62-6a940381a611/pre
15 headers={'Authorization': 'Bearer ' + mltoken})
16 print("Scoring response")
17 print(response_scoring.json())
18 pred_response_scoring.json()
19 output_pred['predictions'][0]['values'][0][0]
20 print(output)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS D:\ibm\sprint\ibm cloud\Performance Prediction> python scoringendpoint.py

Scoring response

[{"predictions": [{"fields": {"prediction": "values": [[17.7]]}]]

17.7

PS D:\ibm\sprint\ibm cloud\Performance Prediction>

Ln 20, Col 14 Spaces: 4 UTF-8 CRLF Python 3.10.2 64-bit

```
File Edit Selection View Go Run Terminal Help
app.py - Performance Prediction - Visual Studio Code

EXPLORER
  PERFORMANCE PREDICTION
    static/css
    background.jpg
    favicon.ico
    style.css
    templates
      index.html
    app.py
    decision_model.pkl
    scoringendpoint.py

scoringendpoint.py
17 @app.route('/')
18 def home():
19     return render_template('index.html')
20
21 @app.route('/y_predict', methods=['POST'])
22 def y_predict():
23     ...
24     For rendering results on HTML GUI
25     ...
26     x_test = [[int(x) for x in request.form.values()]]
27     print(x_test)
28     #sc = load('scalar.save')
29     payload_scoring = {"input_data": [{"fields": ["f0", "f1", "f2", "f3", "f4", "f5"], "values": x_test}]}
30
31     response_scoring = requests.post('https://eu-gb.ml.cloud.ibm.com/ml/v4/deployments/f4a9cc62-cd58-47a3-af62-6a948391a611',
32     headers={"Authorization": 'Bearer ' + mltoken})
33     print("Scoring response")
34     print(response_scoring.json())
35     pred=response_scoring.json()
36     output=pred["predictions"][0]["values"][0][0]
37     print(output)
38     if(output<9):
39         ped="Worst performance with mileage " + str(output) + ". Carry extra fuel"
40     if(output>9 and output<=17.5):
41         ped="Low performance with mileage " + str(output) + ". Don't go to long distance"
42     if(output>17.5 and output<=29):
43         ped="Medium performance with mileage " + str(output) + ". Go for a ride nearby."
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

WARNING: This Follow link (ctrl + click) er. Do not use it in a production deployment. Use a production WSGI server instead.

* Running on http://127.0.0.1:5000

Press CTRL+C to quit

* Restarting with stat

* Debugger is active!

* Debugger PIN: 282-314-124

Ln 3, Col 14 Spacing: 4 UTF-8 CRLF Python 3.10.2 64-bit

Vehicle Performance Analyzer

Number of Cylinders
8

Displacement
304

Horse Power
150

Weight
3433

Acceleration
12

Model Year
70

Origin
1

Predict

{{Prediction}}. {{mpg}}

