

Importing the required packages

In [1]:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report,
precision_recall_fscore_support
```

Loading the dataset

In [2]:

```
import os, types
import pandas as pd
from boto3.client import Config
import ibm_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 = ibm_boto3.client(service_name='s3',
                              ibm_api_key_id='clIomr-o3mi4uya6zQO6eZ5Vn2OI0_5pifA9ZHIK6KJZ',
                              ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
                              config=Config(signature_version='oauth'),
                              endpoint_url='https://s3.private.us.cloud-object-
storage.appdomain.cloud')

bucket = 'flightdelay-donotdelete-pr-zwjtizzcthbqf1'
object_key = 'flightdata.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(
__iter__, body )

df = pd.read_csv(body)
df.head()
```

Out[2]:

| | Y | Q | M | D | D | UN | T | F | OR | O | C | A | A | A | C | D | CR | AC | D | U |
|---|---|---|---|----|---|----|---|---|-----|---|----|---|----|----|----|----|-----|------|---|---|
| | E | A | O | Y | Y | IQ | A | L | IGI | R | RS | R | R | R | A | I | S | TU | I | n |
| | R | R | N | OF | O | UE | I | _ | N | I | _A | R | R | R | N | V | E | AL | S | a |
| | | T | T | _ | F | _C | _ | N | AI | . | R | _ | D | D | C | E | LA | EL | T | m |
| | | E | H | M | W | AR | N | U | RP | . | TI | T | E | E | E | R | PS | ELA | A | e |
| | | R | | NT | E | RI | U | M | OR | I | M | I | L | L | L | T | ED | PSE | N | d |
| | | | | H | K | ER | M | | T_I | N | E | E | Y | 5 | D | D | ME | D_T | C | : |
| | | | | | | | | | D | | | | | | | | | IME | E | 2 |
| | | | | | | | | | | | | | | | | | | | | 5 |
| 0 | 2 | 1 | 1 | 1 | 5 | DL | N | 1 | 103 | A | . | 2 | - | 0. | 0. | 0. | 338 | 295. | 2 | N |
| | 0 | | | | | | 8 | 3 | | T | . | 1 | 41 | | | | | 1 | | a |

| | Y E A R | Q U A R T E R | M O N T H | D A Y _ O F _ M O N T H | D A Y _ O F _ W E E K | U N I Q U E _ C A R R I E R | T A I L _ N U M | F L _ N U M | O R I G I N _ A I R P O R T _ I D | O R I G I N | C R S _ A R R _ T I M E | A R R _ T I M E | A R R _ D E L E T A Y | A R R _ D E L E T 1 5 | C A N C E L L E D | D I V E R T E D | C R S _ E L A P S E D _ T I M E | A C T U A L _ E L A P S E D _ T I M E | D I S T A N C E | U n n a m e d : 2 5 | |
|---|------------------|---------------------------------|-----------------------|--|---|--|--------------------------------------|----------------------------|---|----------------------------|--|--------------------------------------|---|---|---|--------------------------------------|--|---|--------------------------------------|--|-------------|
| | 1 6 | | | | | | 3 6 D N | 9 9 | 97 | L | . | 43 | 0 2. 0 | .0 | 0 | 0 | 0 | .0 | 0 | 8 2. 0 | N |
| 1 | 2 0 1 6 | 1 | 1 | 1 | 5 | DL | N 9 6 4 D N | 1 4 7 6 | 114 33 | D T W | . | 14 35 | 1 4 3 9. 0 | 4. 0 | 0. 0 | 0. 0 | 0. 0 | 110 .0 | 115. 0 | 5 2 8. 0 | N a N |
| 2 | 2 0 1 6 | 1 | 1 | 1 | 5 | DL | N 8 1 3 D N | 1 5 9 7 | 103 97 | A T L | . | 12 15 | 1 4 2. 0 | - 33 .0 | 0. 0 | 0. 0 | 0. 0 | 335 .0 | 300. 0 | 2 1 8 2. 0 | N a N |
| 3 | 2 0 1 6 | 1 | 1 | 1 | 5 | DL | N 5 8 7 N W | 1 7 6 8 | 147 47 | S E A | . | 13 35 | 1 3 4 5. 0 | 10 .0 | 0. 0 | 0. 0 | 0. 0 | 196 .0 | 205. 0 | 1 3 9 9. 0 | N a N |
| 4 | 2 0 1 6 | 1 | 1 | 1 | 5 | DL | N 8 3 6 D N | 1 8 2 3 | 147 47 | S E A | . | 60 7 | 6 1 5. 0 | 8. 0 | 0. 0 | 0. 0 | 0. 0 | 247 .0 | 259. 0 | 1 9 2 7. 0 | N a N |

5 rows × 26 columns

In [3]:

```
df.info()

RangeIndex: 11231 entries, 0 to 11230
Data columns (total 26 columns):
#   Column              Non-Null Count  Dtype
---  -
0   YEAR                 11231 non-null  int64
1   QUARTER              11231 non-null  int64
2   MONTH                11231 non-null  int64
3   DAY_OF_MONTH         11231 non-null  int64
```

```

4   DAY_OF_WEEK      11231 non-null int64
5   UNIQUE_CARRIER  11231 non-null object
6   TAIL_NUM         11231 non-null object
7   FL_NUM           11231 non-null int64
8   ORIGIN_AIRPORT_ID 11231 non-null int64
9   ORIGIN            11231 non-null object
10  DEST_AIRPORT_ID   11231 non-null int64
11  DEST              11231 non-null object
12  CRS_DEP_TIME      11231 non-null int64
13  DEP_TIME          11124 non-null float64
14  DEP_DELAY         11124 non-null float64
15  DEP_DEL15         11124 non-null float64
16  CRS_ARR_TIME      11231 non-null int64
17  ARR_TIME          11116 non-null float64
18  ARR_DELAY         11043 non-null float64
19  ARR_DEL15         11043 non-null float64
20  CANCELLED         11231 non-null float64
21  DIVERTED          11231 non-null float64
22  CRS_ELAPSED_TIME  11231 non-null float64
23  ACTUAL_ELAPSED_TIME 11043 non-null float64
24  DISTANCE          11231 non-null float64
25  Unnamed: 25       0 non-null float64
dtypes: float64(12), int64(10), object(4)
memory usage: 2.2+ MB

```

Dropping unnecessary columns

```

In [4]:
df = df[['FL_NUM', 'MONTH', 'DAY_OF_MONTH', 'DAY_OF_WEEK', 'ORIGIN',
'DEST', 'DEP_DEL15', 'CRS_ARR_TIME', 'ARR_DEL15']]
df.head()

```

```

Out[4]:

```

| | FL_NUM | MONTH | DAY_OF_MONTH | DAY_OF_WEEK | ORIGIN | DEST | DEP_DEL15 | CRS_ARR_TIME | ARR_DEL15 |
|---|--------|-------|--------------|-------------|--------|---------|-----------|--------------|-----------|
| 0 | 1399 | 1 | 1 | 5 | ATL | SEA | 0.0 | 2143 | 0.0 |
| 1 | 1476 | 1 | 1 | 5 | DTW | MS P | 0.0 | 1435 | 0.0 |
| 2 | 1597 | 1 | 1 | 5 | ATL | SEA | 0.0 | 1215 | 0.0 |
| 3 | 1768 | 1 | 1 | 5 | SEA | MS P | 0.0 | 1335 | 0.0 |
| 4 | 1823 | 1 | 1 | 5 | SEA | DT W | 0.0 | 607 | 0.0 |

Handling Missing Values

Checking for null values

```
df.isnull().any()
```

In [5]:

Out[5]:

```
FL_NUM      False
MONTH        False
DAY_OF_MONTH False
DAY_OF_WEEK  False
ORIGIN       False
DEST         False
DEP_DEL15    True
CRS_ARR_TIME False
ARR_DEL15    True
dtype: bool
```

Replacing null values

```
df.fillna(df['DEP_DEL15'].mode()[0], inplace=True)
df.fillna(df['ARR_DEL15'].mode()[0], inplace=True)
```

In [6]:

Checking if the replacement is made

```
df.isnull().any()
```

In [7]:

Out[7]:

```
FL_NUM      False
MONTH        False
DAY_OF_MONTH False
DAY_OF_WEEK  False
ORIGIN       False
DEST         False
DEP_DEL15    False
CRS_ARR_TIME False
ARR_DEL15    False
dtype: bool
```

Encoding

One Hot Encoding

```
df = pd.get_dummies(df, columns=['ORIGIN', 'DEST'])
df.head()
```

In [8]:

Out[8]:

| | FL_NUM | MONTH | DAY_OF_MONTH | DAY_OF_WEEK | DEP_DEL15 | CRS_ARR_TIME | ARR_DEL15 | ORIGIN_ATL | ORIGIN_DTW | ORIGIN_JFK | ORIGIN_MSP | ORIGIN_SEA | DEST_ATL | DEST_DTW | DEST_JFK | DEST_MSP | DEST_SEA |
|---|--------|-------|--------------|-------------|-----------|--------------|-----------|------------|------------|------------|------------|------------|----------|----------|----------|----------|----------|
| 0 | 1399 | 1 | 1 | 5 | 0.0 | 2143 | 0.0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1476 | 1 | 1 | 5 | 0.0 | 1435 | 0.0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | 1597 | 1 | 1 | 5 | 0.0 | 1215 | 0.0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 1768 | 1 | 1 | 5 | 0.0 | 1335 | 0.0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 4 | 1823 | 1 | 1 | 5 | 0.0 | 607 | 0.0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

In [9]:

```
df.columns
```

Out[9]:

```
Index(['FL_NUM', 'MONTH', 'DAY_OF_MONTH', 'DAY_OF_WEEK', 'DEP_DEL15',
      'CRS_ARR_TIME', 'ARR_DEL15', 'ORIGIN_ATL', 'ORIGIN_DTW', 'ORIGIN_JFK',
      'ORIGIN_MSP', 'ORIGIN_SEA', 'DEST_ATL', 'DEST_DTW', 'DEST_JFK',
      'DEST_MSP', 'DEST_SEA'],
      dtype='object')
```

Splitting dataset into Independent and Dependent Variables

In [10]:

```
X = df.drop(columns=['ARR_DEL15'])
Y = df[['ARR_DEL15']]
```

Converting the Independent and Dependent Variables to 1D Arrays

In [11]:

```
X = X.values
Y = Y.values
```

Splitting dataset into Train and Test datasets

```
In [12]:
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2)

In [13]:
X_train.shape, X_test.shape, Y_train.shape, Y_test.shape

Out[13]:
((8984, 16), (2247, 16), (8984, 1), (2247, 1))
```

Building the Decision tree Machine Learning Model

```
In [14]:
from sklearn.linear_model import Decisiontree
model = Decisiontree (max_iter=800)
model.fit(X_train, Y_train.ravel())

Out[14]:
Decisiontree (max_iter=800)
```

Testing the Model

```
In [15]:
Y_pred_train = model.predict(X_train)
Y_pred_test = model.predict(X_test)
```

```
In [16]:
pd.DataFrame(Y_pred_train).value_counts()

Out[16]:
0.0    7750
1.0    1234
dtype: int64
```

```
In [17]:
pd.DataFrame(Y_pred_test).value_counts()

Out[17]:
0.0    1974
1.0     273
dtype: int64
```

Evaluating the ML Model using Metrics

Classification Report

```
In [18]:
print(classification_report(Y_test, Y_pred_test))
```

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0.0 | 0.96 | 0.96 | 0.96 | 1987 |
| 1.0 | 0.68 | 0.72 | 0.70 | 260 |
| accuracy | | | 0.93 | 2247 |
| macro avg | 0.82 | 0.84 | 0.83 | 2247 |
| weighted avg | 0.93 | 0.93 | 0.93 | 2247 |

Accuracy, Precision, Recall, F1 Score

In [19]:

```
acc = accuracy_score(Y_test, Y_pred_test)
prec, rec, f1, sup = precision_recall_fscore_support(Y_test, Y_pred_test)
print('Accuracy Score =', acc)
print('Precision =', prec[0])
print('Recall =', rec[0])
print('F1 Score =', f1[0])

Accuracy Score = 0.9283489096573209
Precision = 0.9625126646403243
Recall = 0.9562154001006542
F1 Score = 0.9593536985609694
```

Checking for Overfitting and Underfitting

In [20]:

```
train_acc = accuracy_score(Y_train, Y_pred_train)
test_acc = accuracy_score(Y_test, Y_pred_test)
print('Training Accuracy =', train_acc)
print('Testing Accuracy =', test_acc)

Training Accuracy = 0.9190783615316117
Testing Accuracy = 0.9283489096573209
```

There is no big variation in the training and testing accuracy. Therefore, the Logistic Regression model is not overfit or underfit.

Confusion Matrix

In [21]:

```
pd.crosstab(Y_test.ravel(), Y_pred_test)
```

Out[21]:

| | col_0 | col_0 | col_1 |
|-------|-------|-------|-------|
| row_0 | | | |
| 0.0 | 1900 | 87 | |
| 1.0 | 74 | 186 | |

IBM Deployment

In [22]:

```
!pip install -U ibm-watson-machine-learning

Requirement already satisfied: ibm-watson-machine-learning in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.257)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (1.26.7)
Requirement already satisfied: importlib-metadata in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (4.8.2)
```

Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (1.3.4)

Requirement already satisfied: ibm-cos-sdk==2.11.* in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2.11.0)

Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (0.3.3)

Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2022.9.24)

Requirement already satisfied: packaging in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (21.3)

Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (0.8.9)

Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2.26.0)

Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (2.11.0)

Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (2.11.0)

Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (0.10.0)

Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (2.8.2)

Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-learning) (2021.3)

Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-learning) (1.20.3)

Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (1.15.0)

Requirement already satisfied: charset-normalizer~2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-learning) (2.0.4)

Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-learning) (3.3)

Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from importlib-metadata->ibm-watson-machine-learning) (3.6.0)

Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from packaging->ibm-watson-machine-learning) (3.0.4)

In [23]:

```
from ibm_watson_machine_learning import APIClient
import json
```

Authenticate and Set Space

In [24]:


```
wml_credentials = {
    "apikey": "I6vmW4nmyS35HD92jVtP81M_Ltw4dt5YoSFGBSpTvvSJ",
    "url": "https://us-south.ml.cloud.ibm.com"
}
```

In [25]:

```
wml_client = APIClient(wml_credentials)
```

In [26]:

```
wml_client.spaces.list()
```

Note: 'limit' is not provided. Only first 50 records will be displayed if the number of records exceed 50

```
-----
-
ID                                NAME                                CREATED
fe072024-e9ca-46c7-aa4f-461b5d334cfb  flight-delay  2022-11-17T15:04:27.398Z
84b0d9cd-0b5e-498f-97f0-80ce2781bd3c  B7-insurance  2022-10-19T13:45:09.759Z
4e12c9fc-81ad-491b-8d2e-211afa98c59a  iris-B7       2022-10-17T09:05:57.194Z
-----
-
```

In [27]:

```
SPACE_ID = "fe072024-e9ca-46c7-aa4f-461b5d334cfb"
```

In [28]:

```
wml_client.set.default_space(SPACE_ID)
```

Out[28]:

```
'SUCCESS'
```

In [29]:

```
wml_client.software_specifications.list(500)
```

```
-----
NAME                                ASSET_ID                                TYPE
default_py3.6                      0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
kernel-spark3.2-scala2.12          020d69ce-7ac1-5e68-ac1a-31189867356a  base
pytorch-onnx_1.3-py3.7-edt         069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6            09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12         09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
pytorch-onnx_rt22.1-py3.9          0b848dd4-e681-5599-be41-b5f6fccc6471  base
ai-function_0.1-py3.6              0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                         0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod       1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6                  10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl          111e41b3-de2d-5422-a4d6-bf776828c4b7  base
autoai-kb_rt22.2-py3.10            125b6d9a-5b1f-5e8d-972a-b251688ccf40  base
runtime-22.1-py3.9                 12b83a17-24d8-5082-900f-0ab31fbfd3cb  base
scikit-learn_0.22-py3.6            154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
default_r3.6                       1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
pytorch-onnx_1.3-py3.6             1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
kernel-spark3.3-r3.6               1c9e5454-f216-59dd-a20e-474a5cdf5988  base
pytorch-onnx_rt22.1-py3.9-edt      1d362186-7ad5-5b59-8b6c-9d0880bde37f  base
tensorflow_2.1-py3.6               1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
spark-mllib_3.2                    20047f72-0a98-58c7-9ff5-a77b012eb8f5  base
tensorflow_2.4-py3.8-horovod       217c16f6-178f-56bf-824a-b19f20564c49  base
runtime-22.1-py3.9-cuda            26215f05-08c3-5a41-a1b0-da66306ce658  base
-----
```

| | | |
|-------------------------------|--------------------------------------|------|
| do_py3.8 | 295addb5-9ef9-547e-9bf4-92ae3563e720 | base |
| autoai-ts_3.8-py3.8 | 2aa0c932-798f-5ae9-abd6-15e0c2402fb5 | base |
| tensorflow_1.15-py3.6 | 2b73a275-7cbf-420b-a912-eae7f436e0bc | base |
| kernel-spark3.3-py3.9 | 2b7961e2-e3b1-5a8c-a491-482c8368839a | base |
| pytorch_1.2-py3.6 | 2c8ef57d-2687-4b7d-acce-01f94976dac1 | base |
| spark-mllib_2.3 | 2e51f700-bca0-4b0d-88dc-5c6791338875 | base |
| pytorch-onnx_1.1-py3.6-edt | 32983cea-3f32-4400-8965-dde874a8d67e | base |
| spark-mllib_3.0-py37 | 36507ebe-8770-55ba-ab2a-eafe787600e9 | base |
| spark-mllib_2.4 | 390d21f8-e58b-4fac-9c55-d7ceda621326 | base |
| autoai-ts_rt22.2-py3.10 | 396b2e83-0953-5b86-9a55-7ce1628a406f | base |
| xgboost_0.82-py3.6 | 39e31acd-5f30-41dc-ae44-60233c80306e | base |
| pytorch-onnx_1.2-py3.6-edt | 40589d0e-7019-4e28-8daa-fb03b6f4fe12 | base |
| pytorch-onnx_rt22.2-py3.10 | 40e73f55-783a-5535-b3fa-0c8b94291431 | base |
| default_r36py38 | 41c247d3-45f8-5a71-b065-8580229facf0 | base |
| autoai-ts_rt22.1-py3.9 | 4269d26e-07ba-5d40-8f66-2d495b0c71f7 | base |
| autoai-obm_3.0 | 42b92e18-d9ab-567f-988a-4240ba1ed5f7 | base |
| pmml-3.0_4.3 | 493bcb95-16f1-5bc5-bee8-81b8af80e9c7 | base |
| spark-mllib_2.4-r_3.6 | 49403dff-92e9-4c87-a3d7-a42d0021c095 | base |
| xgboost_0.90-py3.6 | 4ff8d6c2-1343-4c18-85e1-689c965304d3 | base |
| pytorch-onnx_1.1-py3.6 | 50f95b2a-bc16-43bb-bc94-b0bed208c60b | base |
| autoai-ts_3.9-py3.8 | 52c57136-80fa-572e-8728-a5e7cbb42cde | base |
| spark-mllib_2.4-scala_2.11 | 55a70f99-7320-4be5-9fb9-9edb5a443af5 | base |
| spark-mllib_3.0 | 5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9 | base |
| autoai-obm_2.0 | 5c2e37fa-80b8-5e77-840f-d912469614ee | base |
| spss-modeler_18.1 | 5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b | base |
| cuda-py3.8 | 5d3232bf-c86b-5df4-a2cd-7bb870alcd4e | base |
| runtime-22.2-py3.10-xc | 5e8cddff-db4a-5a6a-b8aa-2d4af9864dab | base |
| autoai-kb_3.1-py3.7 | 632d4b22-10aa-5180-88f0-f52dfb6444d7 | base |
| pytorch-onnx_1.7-py3.8 | 634d3cdc-b562-5bf9-a2d4-ea90a478456b | base |
| spark-mllib_2.3-r_3.6 | 6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c | base |
| tensorflow_2.4-py3.7 | 65e171d7-72d1-55d9-8ebb-f813d620c9bb | base |
| spss-modeler_18.2 | 687eddc9-028a-4117-b9dd-e57b36f1efa5 | base |
| pytorch-onnx_1.2-py3.6 | 692a6a4d-2c4d-45ff-a1ed-b167ee55469a | base |
| spark-mllib_2.3-scala_2.11 | 7963efe5-bbec-417e-92cf-0574e21b4e8d | base |
| spark-mllib_2.4-py37 | 7abc992b-b685-532b-a122-a396a3cdbaab | base |
| caffe_1.0-py3.6 | 7bb3dbe2-da6e-4145-918d-b6d84aa93b6b | base |
| pytorch-onnx_1.7-py3.7 | 812c6631-42b7-5613-982b-02098e6c909c | base |
| cuda-py3.6 | 82c79ece-4d12-40e6-8787-a7b9e0f62770 | base |
| tensorflow_1.15-py3.6-horovod | 8964680e-d5e4-5bb8-919b-8342c6c0dfd8 | base |
| hybrid_0.1 | 8c1a58c6-62b5-4dc4-987a-df751c2756b6 | base |
| pytorch-onnx_1.3-py3.7 | 8d5d8a87-a912-54cf-81ec-3914adaa988d | base |
| caffe-ibm_1.0-py3.6 | 8d863266-7927-4d1e-97d7-56a7f4c0a19b | base |
| runtime-22.2-py3.10-cuda | 8ef391e4-ef58-5d46-b078-a82c211c1058 | base |
| spss-modeler_17.1 | 902d0051-84bd-4af6-ab6b-8f6aa6fdeabb | base |
| do_12.10 | 9100fd72-8159-4eb9-8a0b-a87e12eefa36 | base |
| do_py3.7 | 9447fa8b-2051-4d24-9eef-5acb0e3c59f8 | base |
| spark-mllib_3.0-r_3.6 | 94bb6052-c837-589d-83f1-f4142f219e32 | base |
| cuda-py3.7-opence | 94e9652b-7f2d-59d5-ba5a-23a414ea488f | base |
| nlp-py3.8 | 96e60351-99d4-5a1c-9cc0-473ac1b5a864 | base |
| cuda-py3.7 | 9a44990c-1aa1-4c7d-baf8-c4099011741c | base |
| hybrid_0.2 | 9b3f9040-9cee-4ead-8d7a-780600f542f7 | base |
| spark-mllib_3.0-py38 | 9f7a8fc1-4d3c-5e65-ab90-41fa8de2d418 | base |
| autoai-kb_3.3-py3.7 | a545cca3-02df-5c61-9e88-998b09dc79af | base |
| spark-mllib_3.0-py39 | a6082a27-5acc-5163-b02c-6b96916eb5e0 | base |
| runtime-22.1-py3.9-do | a7e7dbf1-1d03-5544-994d-e5ec845ce99a | base |
| default_py3.8 | ab9e1b80-f2ce-592c-a7d2-4f2344f77194 | base |
| tensorflow_rt22.1-py3.9 | acd9c798-6974-5d2f-a657-ce06e986df4d | base |

| | | |
|---------------------------------|--------------------------------------|------|
| kernel-spark3.2-py3.9 | ad7033ee-794e-58cf-812e-a95f4b64b207 | base |
| autoai-obm_2.0 with Spark 3.0 | af10f35f-69fa-5d66-9bf5-acb58434263a | base |
| runtime-22.2-py3.10 | b56101f1-309d-549b-a849-eea63f77b2fb | base |
| default_py3.7_opence | c2057dd4-f42c-5f77-a02f-72bdbd3282c9 | base |
| tensorflow_2.1-py3.7 | c4032338-2a40-500a-beef-b01ab2667e27 | base |
| do_py3.7_opence | cc8f8976-b74a-551a-bb66-6377f8d865b4 | base |
| spark-mllib_3.3 | d11f2434-4fc7-58b7-8a62-755da64fdaf8 | base |
| autoai-kb_3.0-py3.6 | d139f196-e04b-5d8b-9140-9a10ca1fa91a | base |
| spark-mllib_3.0-py36 | d82546d5-dd78-5fbb-9131-2ec309bc56ed | base |
| autoai-kb_3.4-py3.8 | da9b39c3-758c-5a4f-9cfd-457dd4d8c395 | base |
| kernel-spark3.2-r3.6 | db2fe4d6-d641-5d05-9972-73c654c60e0a | base |
| autoai-kb_rt22.1-py3.9 | db6afe93-665f-5910-b117-d879897404d9 | base |
| tensorflow_rt22.1-py3.9-horovod | dda170cc-ca67-5da7-9b7a-cf84c6987fae | base |
| autoai-ts_1.0-py3.7 | deef04f0-0c42-5147-9711-89f9904299db | base |
| tensorflow_2.1-py3.7-horovod | e384fce5-fdd1-53f8-bc71-11326c9c635f | base |
| default_py3.7 | e4429883-c883-42b6-87a8-f419d64088cd | base |
| do_22.1 | e51999ba-6452-5f1f-8287-17228b88b652 | base |
| autoai-obm_3.2 | eae86aab-da30-5229-a6a6-1d0d4e368983 | base |
| runtime-22.2-r4.2 | ec0a3d28-08f7-556c-9674-ca7c2dba30bd | base |
| tensorflow_rt22.2-py3.10 | f65bd165-f057-55de-b5cb-f97cf2c0f393 | base |
| do_20.1 | f686cdd9-7904-5f9d-a732-01b0d6b10dc5 | base |
| pytorch-onnx_rt22.2-py3.10-edt | f8a05d07-e7cd-57bb-a10b-23f1d4b837ac | base |
| scikit-learn_0.19-py3.6 | f963fa9d-4bb7-5652-9c5d-8d9289ef6ad9 | base |
| tensorflow_2.4-py3.8 | fe185c44-9a99-5425-986b-59bd1d2eda46 | base |
| ----- | ----- | ---- |

Save and Deploy the Model

```
import sklearn
sklearn.__version__
```

In [30]:

```
'1.0.2'
```

Out[30]:

```
MODEL_NAME = 'flight-delay'
DEPLOYMENT_NAME = 'flight-delay'
DEMO_MODEL = model
```

In [31]:

Set Python Version

```
software_spec_uid =
wml_client.software_specifications.get_id_by_name('runtime-22.1-py3.9')
```

In [32]:

Setup Model Meta

```
model_props = {
    wml_client.repository.ModelMetaNames.NAME: MODEL_NAME,
    wml_client.repository.ModelMetaNames.TYPE: 'scikit-learn_1.0',
    wml_client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:
software_spec_uid
}
```

In [33]:

Save Model

In [34]:

```
model_details = wml_client.repository.store_model(  
    model=DEMO_MODEL,  
    meta_props=model_props,  
    training_data=X_train,  
    training_target=Y_train  
)
```

In [35]:

```
model_details
```

Out[35]:

```
{'entity': {'hybrid_pipeline_software_specs': [],  
  'label_column': 'l0',  
  'schemas': {'input': [{'fields': [{'name': 'f0', 'type': 'float'},  
    {'name': 'f1', 'type': 'float'},  
    {'name': 'f2', 'type': 'float'},  
    {'name': 'f3', 'type': 'float'},  
    {'name': 'f4', 'type': 'float'},  
    {'name': 'f5', 'type': 'float'},  
    {'name': 'f6', 'type': 'float'},  
    {'name': 'f7', 'type': 'float'},  
    {'name': 'f8', 'type': 'float'},  
    {'name': 'f9', 'type': 'float'},  
    {'name': 'f10', 'type': 'float'},  
    {'name': 'f11', 'type': 'float'},  
    {'name': 'f12', 'type': 'float'},  
    {'name': 'f13', 'type': 'float'},  
    {'name': 'f14', 'type': 'float'},  
    {'name': 'f15', 'type': 'float'}]},  
    'id': '1',  
    'type': 'struct'}],  
  'output': []},  
  'software_spec': {'id': '12b83a17-24d8-5082-900f-0ab31fbfd3cb',  
    'name': 'runtime-22.1-py3.9'},  
  'type': 'scikit-learn_1.0'},  
  'metadata': {'created_at': '2022-11-18T07:08:59.197Z',  
    'id': '9e04976f-e706-45ca-b2f0-06104acde153',  
    'modified_at': '2022-11-18T07:09:01.629Z',  
    'name': 'flight-delay',  
    'owner': 'IBMid-673000KZ14',  
    'resource_key': '49b1318b-e5d9-4ff9-88fa-8999f8da022a',  
    'space_id': 'fe072024-e9ca-46c7-aa4f-461b5d334cfb'},  
  'system': {'warnings': []}}
```

In [36]:

```
model_id = wml_client.repository.get_model_id(model_details)  
model_id
```

Out[36]:

```
'9e04976f-e706-45ca-b2f0-06104acde153'
```

Set Meta

In [37]:

```
deployment_props = {
```

```
wml_client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
wml_client.deployments.ConfigurationMetaNames.ONLINE: {}
}
```

Deploy

In [38]:

```
deployment = wml_client.deployments.create(
    artifact_uid=model_id,
    meta_props=deployment_props
)

#####
#####
```

Synchronous deployment creation for uid: '9e04976f-e706-45ca-b2f0-06104acde153' started

```
#####
#####
```

initializing

Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.

ready

```
-----
-----
Successfully finished deployment creation, deployment_uid='cec48201-70cc-4651-aa5d-7f49f99a586a'
-----
-----
```

In []: