

SRPCE

2021-2025



NAME - Karthikeyan.A

DEPARTMENT-CSE

ROLL-NO - 12

SUBJECT - Cloud Application Development

REG-NO - 422021104012

SEMESTER- 5



IBM Cloud ML Sample



Explain a introduction and module and functions for machine learning and development with IBM cloud in sample program



Certainly! In machine learning and development with IBM Cloud, you typically work with modules, functions, and libraries to build and deploy your models. Here's a simple example of an introduction, modules, and functions you might use in a Python program for machine learning on IBM Cloud:

python

Copy code

```
# Introduction
# This is a sample program for machine learning and development on IBM Cloud
# We will use Python and various libraries to create a basic class

# Importing necessary modules and libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
import ibm_boto3
from ibm_botocore.client import Config

# Define functions for data preprocessing, training, and evaluation
def load_data():
    # Load your dataset, for example, from IBM Cloud Object Storage
    # You need to set up your credentials for IBM Cloud Object Storage
    cos = ibm_boto3.client(service_name='s3',
                           ibm_api_key_id='your_api_key',
                           ibm_service_instance_id='your_service_instance_id')
```

```

        config=Config(signature_version='oauth'
                        endpoint_url='your_endpoint_url')

cos.download_file(Bucket='your_bucket', Key='your_dataset.csv')
data = pd.read_csv('dataset.csv')
return data

def preprocess_data(data):
    # Perform data preprocessing, feature engineering, and handling missing values
    # Split data into features and labels
    X = data.drop('target', axis=1)
    y = data['target']
    return X, y

def train_model(X, y):
    # Split data into training and testing sets
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    # Train a machine learning model (Random Forest Classifier in this example)
    model = RandomForestClassifier(n_estimators=100, random_state=42)
    model.fit(X_train, y_train)
    return model

def evaluate_model(model, X_test, y_test):
    # Evaluate the model's accuracy
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    return accuracy

# Main program
if __name__ == '__main__':
    data = load_data()
    X, y = preprocess_data(data)
    model = train_model(X, y)
    accuracy = evaluate_model(model, X_test, y_test)
    print(f"Model Accuracy: {accuracy}")

```

In this sample program:

1. We start with an introduction that describes the purpose of the program.
2. We import various Python modules and libraries like NumPy, pandas, scikit-learn, and IBM Cloud Object Storage for data handling, machine learning, and cloud integration.
3. We define functions for different steps of the machine learning process, including loading data, preprocessing data, training a model, and evaluating the model.
4. In the main program, we load the data, preprocess it, train a machine learning model (Random Forest Classifier), and evaluate its accuracy.

Remember to replace placeholders like `'your_api_key'`, `'your_service_instance_id'`, `'your_endpoint_url'`, `'your_bucket'`, and `'your_dataset.csv'` with your actual IBM Cloud Object Storage credentials and dataset information. Additionally, you should adapt this code to your specific machine learning problem and dataset.