Assignment 3 v2.0

AdaptiveAlgo Systems Inc. has invited you to be partners in helping them implement data science solutions. Your company was selected after rigorous review and is held in high esteem for the quality data science solutions you have been developing. However, the AdaptiveAlgo is concerned if your team of three can deliver high performance solutions in a timely manner. Note that AdaptiveAlgo wants to make a decision on which team they want to partner with based on the deliverable you submit by April 13th 11.59pm. They have pre-selected datasets that are here: https://docs.google.com/spreadsheets/d/1n_bHTkz6G76rgH1Vhu_065ZWEP2a8DVND6tk8Pz6WcM/edit?usp=sharing and will be made available through Amazon S3. Since AdaptiveAlgo has all solutions on the cloud, you should also implement all solutions on the cloud. You have a choice of cloud.

Here are the requirements:

Part1: Model design and building

1. The dataset is on Amazon S3. Access the data assigned to you from S3
2. You should build a pipeline using **Luigi/Airflow/Sklearn** (See the google link for your team's allocated method. This pipeline incorporates:
    1. Data ingestion into **Pandas**
    2. Cleanup the Data if needed
    3. Exploratory Data Analysis with **Plotly/seaborn/matplotlib**
    4. Feature Engineering on the data
    5. Feature Selection or any transformation on the dataset
    6. Run Different Machine learning models (at-least 3) for the problem assigned
    7. Get the Accuracy and Error metrics for all the models and store them in a csv file with Ranking of the models
    8. Pickle all the models
    9. Upload the error metric csv and models to s3 bucket
3. Dockerize this pipeline using **Repo2Docker** or write your own docker file
4. Note:
    1. Properly document your code
    2. Use Python classes and functions when needed for replicability and reuse.
    3. You should try and use configuration files when possible to ensure you can make modifications and your solution is generic.
    4. You should also write a comprehensive Readme.md to detail your design, implementation, results and analysis
    5. Use any other Python package when needed

Part2: Model Deployment

1. Create a Web application using **Flask** that uses the models created (in Pickle format) in Part1 and stored on S3
2. Build a web page which takes user inputs. The application should allow submission of data for prediction via Forms as well as REST Api calls using JSON
3. The application should allow submission on single record of data as well as batch of records data upload to get single/bulk responses.
4. The Result should be provided to the user as a csv file and a table with results should be displayed
5. You need to use the models saved in S3 to run your models.

6. Create Unit tests for the user and test your application.
7. Dockerize this using **repo2docker** or write your own docker file. Whenever your run your docker image, your application should get the latest models from S3 and do predictions on all the three (or any number of models you developed) and present outputs.
8. Note that your webapp should get the latest models whenever the models change. You implement this using Amazon Lambda. See the following resources on how to accomplish it:
    1. https://aws.amazon.com/lambda/
    2. https://github.com/aws-samples/lambda-refarch-fileprocessing
    3. https://docs.aws.amazon.com/lambda/latest/dg/python-programming-model-handler-types.html
    4. https://s3.amazonaws.com/awslambda-reference-architectures/file-processing/lambda-refarch-fileprocessing.pdf
9. When you have more than 10 inputs, use **Dask** to setup a cluster and divide the load of computation
10. Write a Readme.md detailing your application

Bonus Points:

Do Auto Scalability and create more worker nodes as needed?

Email analyticsneu@gmail.com and the TA the following deliverables:

1. Urls for fully hosted applications

2. Github respository with all files

3. Submission deadline April 13th 11.59 pm

4. You will have a Demo on April 14th in class. Plan to present the project. You will have 10 minutes