Technical Assignment: Modeling Data Scientist

#1-amanotes

Version: `April, 2020`

> **BEFORE YOU START**
>
> 1. Please start the exercise as soon as possible, it's not a difficult but might take time.
> 2. DON'T worry if you can finish all tasks. Do as much as you can.
> 3. You can try and explore anything you want to do, but be selective in your approach to save your time, and add only critical points in the submission package.
> 4. We attempt to keep the scores, purposes, and expectations transparent. Please work on the tasks you are more interested first.
> 5. As this is a long exercise and the data is limited, DON'T spend too much time on tuning the model. We are more interested in considering the full workflow.

Submission Package for the Assignment

1. **Technical Report PDF:** Execution Summary, Presentation of Result, Discussions
2. **A zip file of the repository**: with all relevant scrips and files supporting your exercise

Dataset & Business Demand

Provided Dataset: This is a fictional dataset for e-commerce (dataset: `ml-technicaltest-ecommerce.csv`, columns includes:

1. `InvoiceNo`: Number of Invoice
2. `StockCode`: Code of Stock for the Product items
3. `Description`: Description of the Product items
4. `Quantity`: Count of items in the invoice
5. `UnitPrice`: Unit Price of items in the invoice
6. `CustomerID`: Customer ID of the invoice
7. `Country`: The country of customer

```
Data columns (total 7 columns):
InvoiceNo      541909 non-null object
StockCode      541909 non-null object
Description    540455 non-null object
Quantity       541909 non-null int64
UnitPrice      541909 non-null float64
CustomerID     406829 non-null object
```

```
Country        541909 non-null object
```

Business Demand: We want to build a recommender that once the customer choose a Product, it will recommend Top 5 other Products highly relevant for the customer.
(Alternative):
Considering the given dataset, you can be creative and come up with any other ML model which benefits the business case.

Task 1: Basics in Code Practices (Score: 20)

> **Purpose:** This task is to test the code practices of candidates in set-up and working in the projects.

- *Task 1.1 [Score: 10] – Set-up Repo*
  Create a repository contains a docker container with:
  1. Jupyter Notebook (with all relevant python packages)
  2. Tesseract OCR
  3. Postgres

  ➜ **Expectation:** The repo of your submission contains any `Dockerfile` and/or `docker-compose.yml` that it's ready to "up-and-run"
- *Task 1.2 [Score: 10] – Code Hygiene in the whole work*
  ➜ **Expectation:** The code is concise, efficient, readable with meaningful comments in code body.

Task 2: Model Ideation on Jupyter Notebook (Score: 50)

> **Purpose:** This task is to test the capabilities in data exploring, and formulate a ML model from business problem. Due to the limitation of the data, we would not emphasize on the performance of the model. Please do not spend much time on tuning the model.

- *Task 2.1 [Score: 10]– Exploring the data*
  ➜ **Expectation:** Please follow your current approach. Add important insights, together with visualizations into the Technical Report.
- *Task 2.2 [Score: 10]–* Cleaning data steps
  ➜ **Expectation:** The rational for this test should be from `Task 2.1` about the necessary steps to clean data
- *Task 2.3 [Score: 20]– Model Design*
  1. What is the raw input of the model?
  2. Preprocessing and any feature engineering steps

3. What are inputs/outputs in predicting?
4. ML Algorithms you choose, and why?

➜ **Expectation:** The model design should cover all above questions in Technical Report.

- *Task 2.4 [Score: 20]*– Please prototype the train/predict on Jupyter Notebook with the standard steps of ML modeling.

  ➜ **Expectation:** Code/Output on Jupyter Notebook + Summary in Technical Report.

Task 3: Scale & Productize (Score: 30)

> **Purpose:** This task is to test the capabilities in architecting ML Pipeline to training & serving model on Big Data.

- *Task 3.1 [Score: 20]– Design Blueprints for ML Pipelines*

  You can add any further assumption about the production environment to make it as realistic as possible (For example: "the data is loaded in a data lake by a third-party app" and "at the end, the users want an API that they can query to get a prediction") .

  The Design should include these pipelines (for each pipeline, please specify the tool/ language/engine/platform you use - pros, cons, alternative, recommendation)

  1. Ingesting
  2. Training
  3. Evaluating/Monitoring
  4. Predicting (Serving)

  ➜ **Expectation:** The diagrams of ML Pipeline and discussion.

- *Task 3.2 [Score: 10]– Write code for ML Pipelines*

  Choose at least 1 out of 4 pipelines in Task 3.1 and write the code to implement the pipeline on the tools you suggested (Notice that the code is expected to be scalable on Big Data and easy serving in prediction)

  ➜ **Expectation:** Coding scripts