00-working

July 27, 2024

1 Car Sales Project

1.1 Working

I quickly reviewed the task, the time required for completion, and thought about the tools I would need to complete the task.

I knew the job role required MySQL and Python, and after seeing the `CSV` files contained tabular data, and looking at the task questions, I decided to use these as my main tools.

As the task needed to be done in 2 hours, I decided on the quickest workflow, using the following tools:

- `Jupyter` notebooks to write documentation and do exploratory data analysis (EDA)
- `pandas` for data importing and wrangling on the `CSV` files
- `pandasql` to run SQL queries on the `pandas` DataFrame
- `pygwalker` to create quick interactive data visualisations (EDA) for the stakeholders
- `streamlit` to make the visualisations accessible for non-tehnical stakeholders

1.2 Steps

1. Create project folder structure
2. Initialise `git` repository
3. Create `conda` environment with Python 3.10 and key packages
4. Start inspecting data

1.3 Discoveries

1.3.1 Tables

I know straight away that to answer the questions, I will need to do some SQL joins, because there is more than one table.

There are two tables (DataFrames) of car sales data: `purchase_data` and `vehicle_data`

`customer_id` is the primary key of the `purchase_data` table

`vehicle_id` is the primary key of the `vehicle_data` table

So, I will use `vehicle_id` as the key to join the tables on.

1.3.2 `purchase_data`

This table contains:

- Information about car purchases per customer
- `9` columns
- `2_000_000` purchases (rows)

1.3.3 `test_vehicle_data`

- Information about vehicles
- `19` columns
- `978` vehicles (rows)

1.3.4 Questions

Next, because there is a lot of data in the tables, I'll read the questions, to know which variables are needed to answer them.

I listed the questions, highlighted the variables, and then clarified which columns refer to which variables in the tables.

I need to do this to check if there are any issues with data quality, missing values, or outliers, only for the relevant variables. Otherwise, this will take too long, due to the number of columns.

1.4 TODO

DONE - Tables are already indexed using the `customer_id` (`purchase_data`) and `vehicle_id` (`vehicle_data`) columns, so I need to import the `CSV` to specify the index.

2 Evaluation

Strengths:

- Automated workflow
- Free and Open Source tools
- Version control

Weaknesses:

- Coding is slower than using GUI
- Choosing between SQL and pandas for queries
- Folium could not display map, so went with pygwalker and datashader/holoviews/bokeh

Ideas for next time:

- `Tableau` for quicker workflow
- `pygwalker` for entire project as interactive visualisations
- `duckdb` to speed up SQL queries