CASE 03: The Wheels in Motion

A public passenger vehicle refers to any vehicle operated by a public chauffeur to transport passengers for hire. These include licensed taxicabs (medallions), liveries, ambulances, medicars, charter and sightseeing buses, horse-drawn carriages, and pedicabs. Thus, this case study focuses on analysing data collected on registered public passenger vehicles in Illinois state.

For the analysis and visualization, you will be provided a folder "PassengerVehicle_Stats" containing 9 csv files. Each file corresponds to the fuel type of the vehicle. The files include several attributes related to each registered vehicle and the table below outlines the summary of these attributes.

Attribute	**Description**
Public Vehicle Number	License number for the vehicle
Status	Status of the vehicle license (ex: active)
Vehicle Make	Make of the vehicle (ex: Toyota)
Vehicle Model	Model of the vehicle (ex: Camry)
Vehicle Model Year	Year the vehicle was manufactured.
Vehicle Color	Color of the vehicle body
Vehicle Fuel Source	Type of fuel used to power the vehicle
Wheelchair Accessible	Whether the vehicle is equipped to accommodate passengers using wheelchairs
Company Name	The license holder
Address	Address of the company
City	The city where the company is located
State	State where the company is located
ZIP Code	Zip code of the city
Taxi Affiliation	The association the taxi vehicle is affiliated with
Taxi Medallion License Management	Description of who manages the taxi
Record ID	A unique identifier for each record (combination of vehicle type and the public vehicle number)

Apart from the above, you will also receive an additional file "car_reviews.csv". This contains 7000 customer reviews received for Kia vehicles listed on Edmunds automotive inventory. This file should be used to complete Task 03.

TASK 01: Maintain a GitHub Repository

- From the beginning, create and maintain a GitHub repository for the project.
- Follow proper version control practices and GitHub etiquettes (e.x: meaningful commits).
- We will limit our evaluation to the Python scripts and Jupyter notebooks present in the repository. Please ensure all your code is pushed promptly!
- Refer to the marking grid to ensure all necessary components are addressed for evaluation.

TASK 02: Data Preparation

To achieve the passing mark, the following tasks are mandatory. Implementing advanced techniques will earn extra credit. Carry out the below tasks in a Jupyter Notebook.

1. **Reading and combining data**
   - Load all 9 CSV files into a list.
   - Concatenate the files into a single DataFrame, named **vehicles_df**.

2. **Initial data exploration and cleaning**
   - Examine the DataFrame structure, including its features and data types.
   - Remove any duplicate records.
   - Remove null records only if it is required. Provide reasons for your decision.

3. **Handle outliers and missing values**
   - Perform outlier removal and missing value imputation only if necessary.
   - State the reason for any such actions (you can state the reasons within the notebook).

4. **Adding new columns to the Dataframe**
   1. Vehicle Type
      - A string column indicating the type of the public passenger vehicle.

      Hint: *Extract this information from the "Record ID" column. It is a combination of vehicle type and the public vehicle number.*

5. **Column Removal**
   - Drop the columns "Address" and "Public Vehicle Number".

TASK 03: Deploying a HuggingFace Model

Complete this task in a separate Jupyter notebook. Treat it as an independent task, and there's no need to consider it in relation to the rest of the tasks.

- Read the data from car_reviews.csv file. It has around 7000 customer reviews received for Kia vehicles.

- Select a suitable zero-shot classification model from HuggingFace and provide the rationale behind selecting the model.
- Using the selected model, classify the reviews into one of the below classes;
    1. talks about driving experience
    2. talks about features
    3. talks about value for money
    4. talks about issues
    5. other
- Fit a hugging face model to detect the sentiment of each excerpt and provide the rationale behind selecting the hugging face model.
- Add the predicted review category and the sentiment to the dataset as two new columns (name the columns "talks_about" and "sentiment" respectively).
- Visualize the sentiment spread using a suitable chart.
- Visualize the spread of the review category ("talks_about") using a suitable chart.
- Ensure to push both the updated dataset and the notebook to the GitHub repo.

TASK 04: Dashboard Creation

- Design a dashboard using Plotly Dash that tells an insightful story with the data.
- Be SMART!!! There are many different charts you can use to visualize data. Refer to Plotly documentation to decide the best and most interactive charts to showcase your story.
- Refer to the marking grid to cover all required aspects.

SUBMISSION GUIDELINES

- **Python scripts and notebooks:** Push to a public GitHub repository
- **Dashboard:** Screen record and submit as a video
- **Presentation:** A maximum of 5 slides explaining what you did in the analysis
- Upload the below items to the Google Form (will be shared on the 5th of Dec):
    1. GitHub repository link (public)
    2. Video clip of the dashboard
    3. PowerPoint Presentation

Deadline: 6th of December 2024 11:59 PM

MARKING GRID

	Task	Weight	Evaluation Criteria - minimum requirements	
Git	**Maintain a git repo for the project**	10%	1.1	All the team members should be added to the project
			1.2	Maintain branches for each component/member
			1.3	At least two commits per member
			1.4	At least one completed pull request
			1.5	Make commits on-the-spot (not at the end)
			1.6	Maintain proper branch naming conventions
			1.7	Maintain meaningful commits
			1.8	Main branch should be free of conflicts
Pandas	**Data preparation**	30%	2.1	Read data files
			2.2	Merge files
			2.3	Remove duplicate/null records
			2.4	Impute missing values (only if required)
			2.5	Outlier removal (only if required)
			2.6	Pivoting / Grouping
NLP	**Deploying a Hugging Face model**	20%	3.1	Pick a suitable model
			3.2	Reliability of the model
Visualization	**Dash Dashboard**	40%	4.1	Use correct charts to represent data
			4.2	Include at least 5 different types of charts
			4.3	Call the charts to a dashboard
			4.4	Use interactive features on the dashboard (ex: filters)
			4.5	Clarity of the dashboard
			4.6	Story-telling

To pass, you must score at least 65% of the allocated marks in each section.

If you have any queries reach out to us via:

uvini.ranaweera@acuitykp.com
samujitha.senaratne@acuitykp.com