AMERICAN UNIVERSITY OF BEIRUT

MAROUN SEMAAN FACULTY OF ENGINEERING & ARCHITECTURE

Department of Industrial Engineering and Management

INDE 535 – Data Analytics for Operations Research and Financial Engineering

Final Project – Car Price Prediction with Machine Learning

By - Ralph Mouawad, Lea Bou Sleiman, Michel Lamah

To - Inst. Mario Karam

Please Refer to the Python Collab File on Moodle

1- Introduction to the Problem:

GOM is a global online marketplace for buying and selling services and goods, such as cars and bikes. They are facing difficulties putting a price on the cars they are selling, because they might overprice or underprice them.

We, as data-driven consultants, are here to propose a solution.

2- Our Solution:

Given	Method	Result
• Historical data, including cars with their specific features and prices	• Develop machine and deep learning models to find the pattern between the features of the cars and their price, so that when we are given unlabeled cars with their features, our model will be able to estimate their price	• Close Estimates of Car prices depending on features available

3- Insights from our Data Exploration:

Given Features:

- **created_at_first** - time when the car was first listed on GOM application
- **Id**- anonymous id for every car
- **city** - city where the car is currently put for sale
- **brand** - brand of the car
- **region** - region where the car is currently put for sale
- **year** - year when the car was first introduced to the market
- **model** - model of listed car
- **body_type** - type of body of listed car
- **transmission_type** - type of transmission of listed car
- **kilometers** - distance traveled by car till date of listing on application
- **price** - price at which the car was listed for sale

How to proceed?

a) Remove irrelevant columns:
   - Remove **created_at_first, city, region** because we assumed these won't affect the price of a car. Including them would make our model more complex and might lead to errors
   - Remove the **body_type column** because more than 50% of the data was missing
   - Remove missing data from the columns **brand, model, and transmission_type** because imputing random values might lead to errors

b) Impute missing years and kilometers values with the most frequent ones

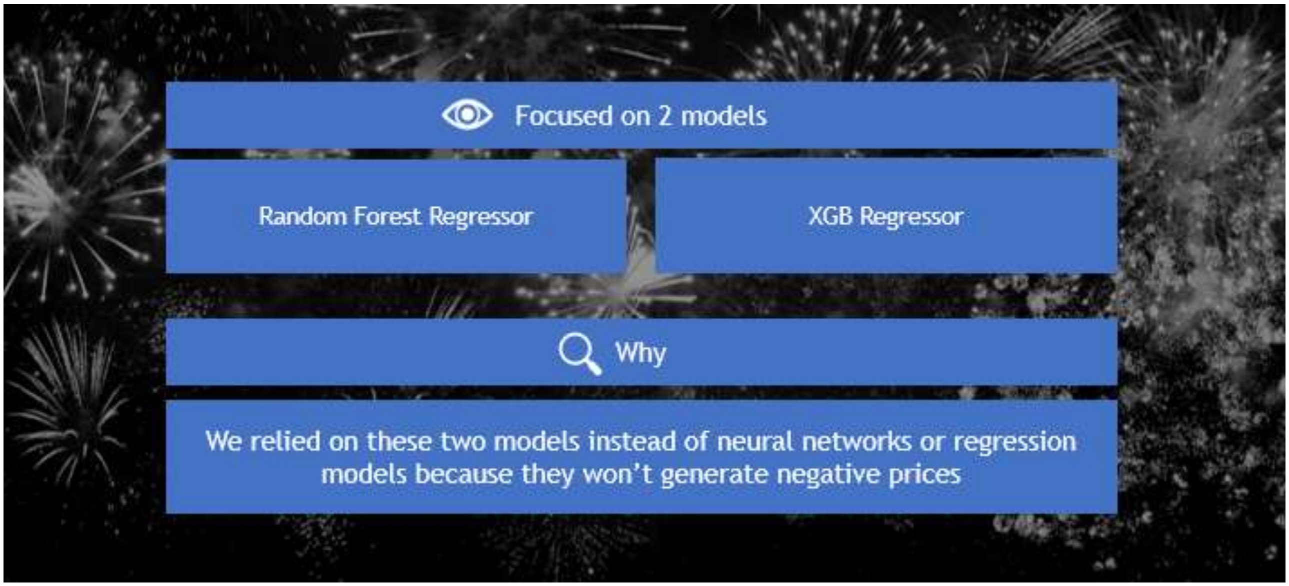
c) Transform the year column to an age column (2019 – year) to see how old the car is. We assume that this is more relevant.

d) Create 5 intervals for the kilometer's column: 'Very new', 'New', 'Somewhat', 'Old', 'Very Old'. This will prevent Overfitting especially for tree-based models because we had previously 22 intervals.

e) Data Pre-Processing:
   - Year column: performed min-max transformation and made sure to prevent data leakage. That would reduce the scale of the year column.

- Kilometers column: Ordinal Encoding because the more kilometers a car has traveled, the less price it should be.
- For the other columns: Dummy encoding because ML algorithms expect numbers.

4- Results of our Predictive Models:



Focused on 2 models

Random Forest Regressor

XGB Regressor

Why

We relied on these two models instead of neural networks or regression models because they won't generate negative prices

After performing Grid Search Cross Validation on both models, XGBRegressor performed the best and attained an

RMSLE = 0.25400.

5- Implementation Plan:

- Worked on Google Collab and employed many Machine Learning libraries and data pre-processing tasks.
- Faced difficulties in the training of Grid Search because it is computationally expensive.

Ranked TOP 4 of the competition, making our model successful.