FAST, FURIOUS AND INSURED: HACKER EARTH MACHINE LEARNING CHALLENGE

→Technology Stack: Python, Jupyter notebook

→Task 1:

Image Classification: Predict if the vehicle provided in the image is damaged or not.

Approach 1:

- Used the images of the vehicle along with techniques such as data augmentation and passed those images through the pre-trained CNN models for predicting the condition of the vehicles in the images.
- Used the pre-trained CNN models such as 'MobileNet',' InceptionResNetV2', 'MobileNetV2' , 'Xception' ,'InceptionV3', 'ResNet50V2' ,'VGG16' ,'VGG19' and 'EfficientNetB1'.

Approach 2:

- Converted the images of the vehicle into numpy arrays using OpenCV library and saved the labels for these images into list.
- Passed these converted images in numpy arrays through the pre-trained CNN models such as 'MobileNet',' InceptionResNetV2', 'MobileNetV2' , 'Xception' ,'InceptionV3', 'ResNet50V2' ,'VGG16' ,'VGG19' and 'EfficientNetB1'.

Final Approach:

- Validation Accuracy for all the pre-trained for both the approaches was in the range of 94% to 96%.
- Predictions from Approach 2 with pre-trained CNN models - 'MobileNet' gave better score on the leader board.

→Task 2:

Regression: Based on the condition of a vehicle, predict the insurance amount of the cars that are provided in the dataset

Approach:

- Handled missing values by imputing them with '0' and dropped the outliers and other unwanted data.
- Feature Engineering :
    1. Created new features using 'group by' and 'transform' functions along with summary statistics like 'mean', 'max' and 'min'.
    2. Created features using binning technique for numerical and categorical features.
    3. Also created features using python date functions for 'Expiry Date' column.
- For categorical variable tried to convert them into numerical variable by using both label encoding and by creating python's dummy function but results were better when using python's dummy function for the categorical variables.
- Algorithms Used: Linear Regression, LassoCV, Extra Trees, Decision Tree, Random Forest, Gradient Boosting, XGBoost, XGBRFRegressor, CatBoost, Light GBM.
- Random Forest model gave better results out of all the results.