Machine Learning Project

Regression: Car Price Prediction

Classification: 18- Fashion Item Classification

Team Member:

ID	Name
20220180	Ziad Adel Farghaly Mehran
20220179	Ziad Tarek Galal Elsayed
20220191	Ziad Waleed Mohamed
20220181	Ziad Atef Ali Abdelsalam
20220167	Rawan Hesham Mostafa Abdelhafez
20220104	Bassel Waleed hamd Mohamed Ibrahim

Numerical Dataset

- Name: Car Price Prediction Challenge
- Link: https://www.kaggle.com/datasets/deepcontractor/car-price-prediction-challenge
- Description:
    - CSV file: 19237 rows x 18 columns (Price Columns as Target)
    - Attribute Column Feature:
        1. ID
        2. Price: price of the care (Target Column)

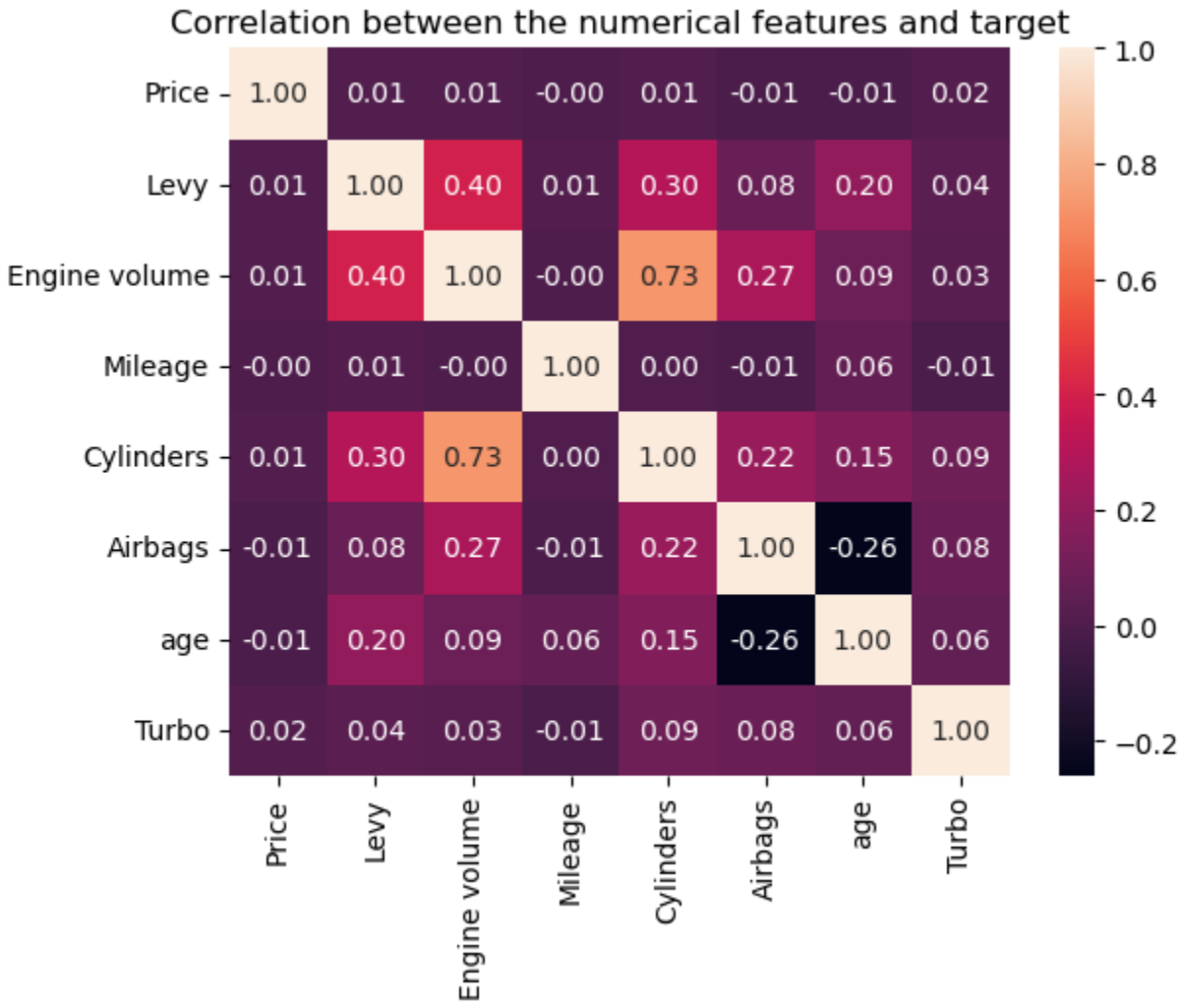
3. Levy
4. Manufacturer
5. Model
6. Prod. Year
7. Category
8. Leather interior
9. Fuel type
10. Engine volume
11. Mileage
12. Cylinders
13. Gear box type
14. Drive wheels
15. Doors
16. Wheel
17. Color
18. Airbags

The dataset not contain any empty values, ensuring consistency and completeness.

However, during preprocessing, adjustments were made to address **overfitting concerns**. Specifically, certain test set entries contained features not encountered during training, such as rows with rare or unique values like "number of airbags" being 13 or 15, which appeared only once. These rows were removed to prevent the model from being biased toward such anomalies.

Additionally, the "production year" column was replaced with the "age of the car," as it provides more relevant information for analysis and modeling. Outliers were also identified and removed to ensure the dataset's quality and to improve model performance by avoiding skewed predictions caused by extreme values.

And get the correlation between feature to be know which best Feature to select and which one have high correlation to split these feature (remove one to get accurate result).



Algorithms

Linear regression:

is a algorithm that models the relationship between a dependent variable (target) and one or more independent variables (features) by fitting a linear equation to the observed data. The goal is to find the line of best fit that minimizes the residuals (differences between actual and predicted values).

Steps:

- Split the data into training, validation, and test sets.
- Trained the model using the training set to learn the relationship between features and the target variable.
- Evaluated model performance on the validation set to check for underfitting or overfitting.
- Tested the model on the test set to measure its generalization to unseen data.

```
MSE 0.725694099349343
MAE 0.6466471144077001
Rscore for x_train: 0.2951543235366917
Rscore x_train: 0.27835961697748013
```

K-Nearest Neighbors Regression (KNN Regression):

Is a non-parametric algorithm that predicts the target value for a data point by averaging the target values of its nearest neighbors in the feature space. The number of neighbors (k) is a hyperparameter that controls the trade-off between bias and variance.

Steps:

- Applied grid search (param_grid) to define a range of potential values for hyperparameters (e.g., the number of neighbors k).
- Used cross-validation to evaluate multiple configurations of hyperparameters and select the one with the best performance.
- Trained the final model using the optimal hyperparameters and validated its performance on the test set.

```
Best Cross-Validated MSE: 0.45810619826128
Validation MSE: 0.49462636441941416
Testing MSE: 0.48936879679183715
Validation MAE: 0.4630530835779153
Testing MAE: 0.47011700594038874
Rscore for x-test: 1
Rscore for x-vaild: 0.5001517873929724
Rscore for x_test: 0.5133648099484843
```

Compare in Results between Two Algorithm

	Linear Regression prediction	KNN predictions	Actual Values
0	18922.37	21926.72	19736.00
1	6257.26	4715.36	3136.00
2	7040.34	5840.13	6272.00
3	13518.21	12572.18	13150.00

Classification Dataset

- Name: Fashion Item Classification
- Link: https://github.com/zalandoresearch/fashion-mnist
- Description: consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes **We use (7).**

Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

In the preprocessing stage for image data, we applied normalization to the photos to ensure that the pixel values were scaled to a consistent range, typically between 0 and 1. This step is crucial for improving the convergence of machine learning models and reducing the impact of varying pixel intensity ranges. After preprocessing, the dataset was split into training and validation sets. The training set was used to train the model, while the validation set was reserved to evaluate the model's performance and fine-tune its parameters, ensuring a robust and generalizable solution.

Algorithms

Logistics regression results:

```
Accuracy score in training for Logistics: 0.9547080370609783
Accuracy score in test for Logistics: 0.9370748299319728
```

KNN Classification results:

```
Accuracy score in training for KNN: 0.945054945054945
Accuracy score in test for KNN: 0.9275510204081633
```

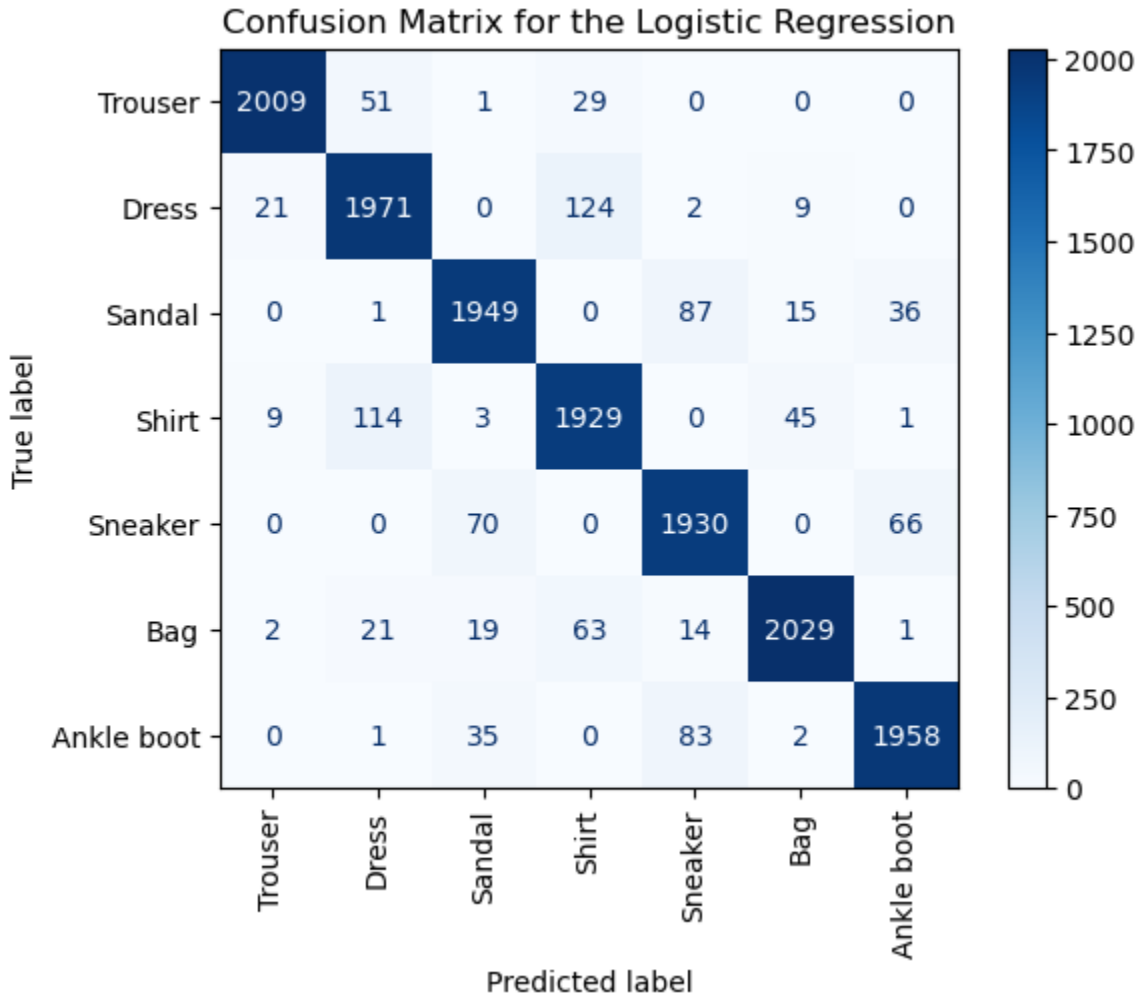
Precision and Recall Result for Logistics regression:

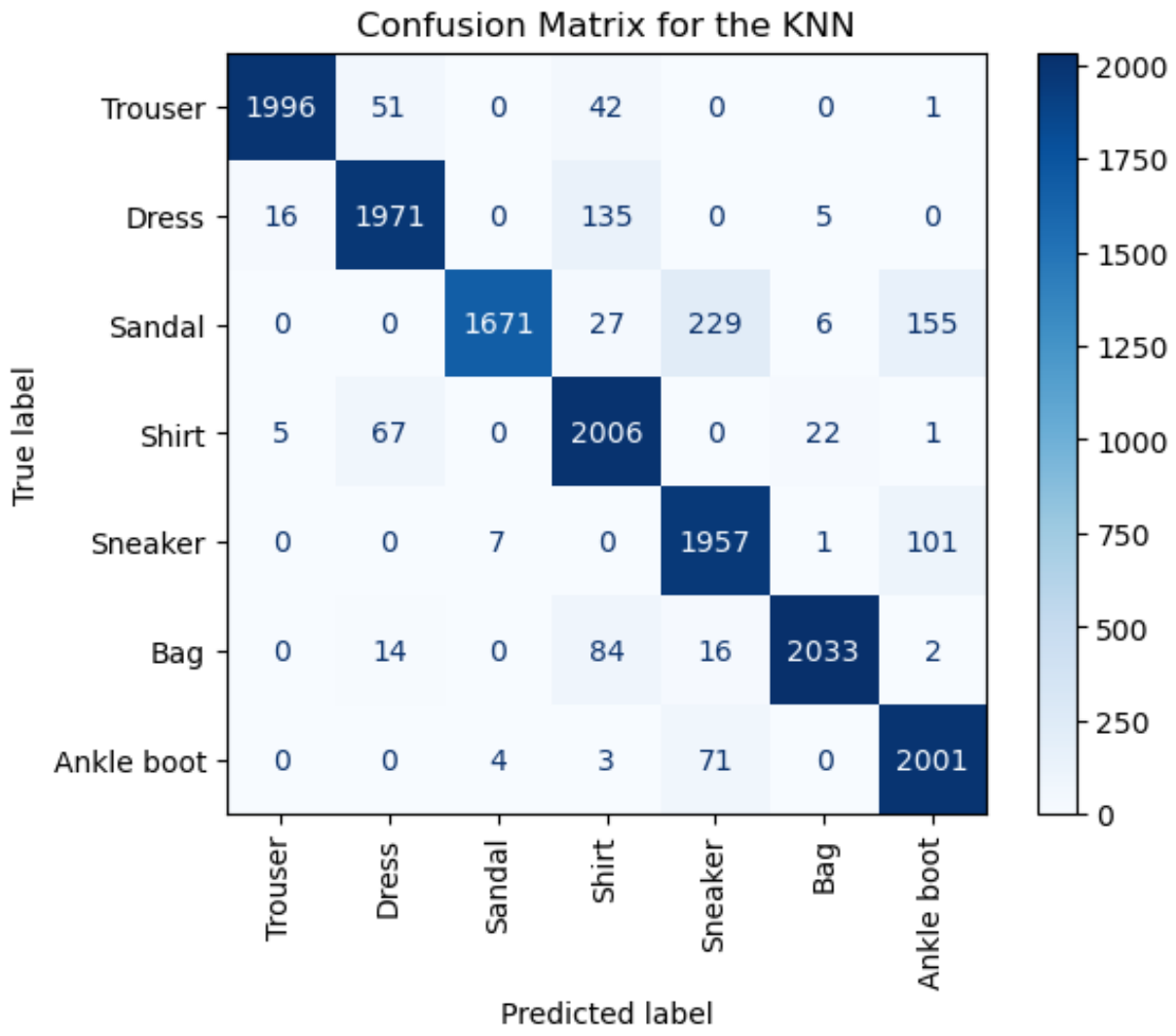
	precision	recall	f1-score	support
Trouser	0.98	0.96	0.97	2090
Dress	0.91	0.93	0.92	2127
Sandal	0.94	0.93	0.94	2088
Shirt	0.90	0.92	0.91	2101
Sneaker	0.91	0.93	0.92	2066
Bag	0.97	0.94	0.96	2149
Ankle boot	0.95	0.94	0.95	2079
accuracy			0.94	14700
macro avg	0.94	0.94	0.94	14700
weighted avg	0.94	0.94	0.94	14700

Precision and Recall Result for KNN:

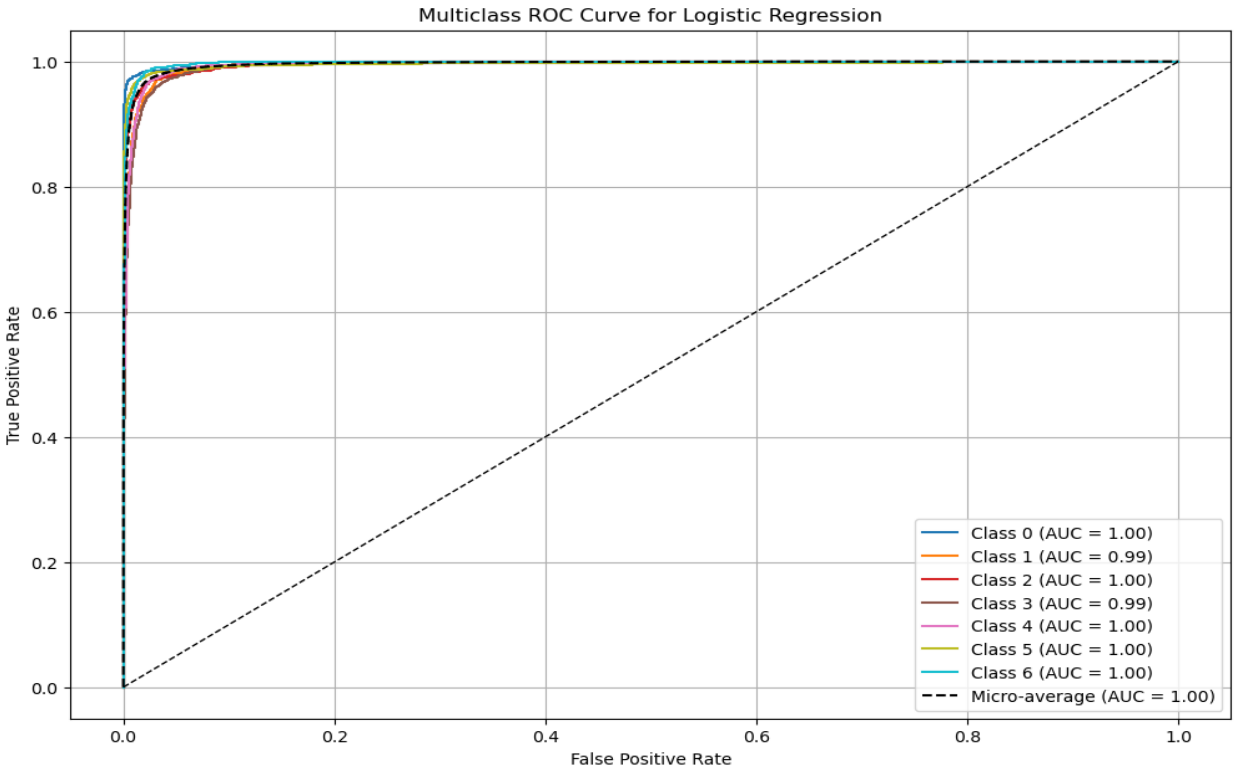
	precision	recall	f1-score	support
Trouser	0.99	0.96	0.97	2090
Dress	0.94	0.93	0.93	2127
Sandal	0.99	0.80	0.89	2088
Shirt	0.87	0.95	0.91	2101
Sneaker	0.86	0.95	0.90	2066
Bag	0.98	0.95	0.96	2149
Ankle boot	0.89	0.96	0.92	2079
accuracy			0.93	14700
macro avg	0.93	0.93	0.93	14700
weighted avg	0.93	0.93	0.93	14700

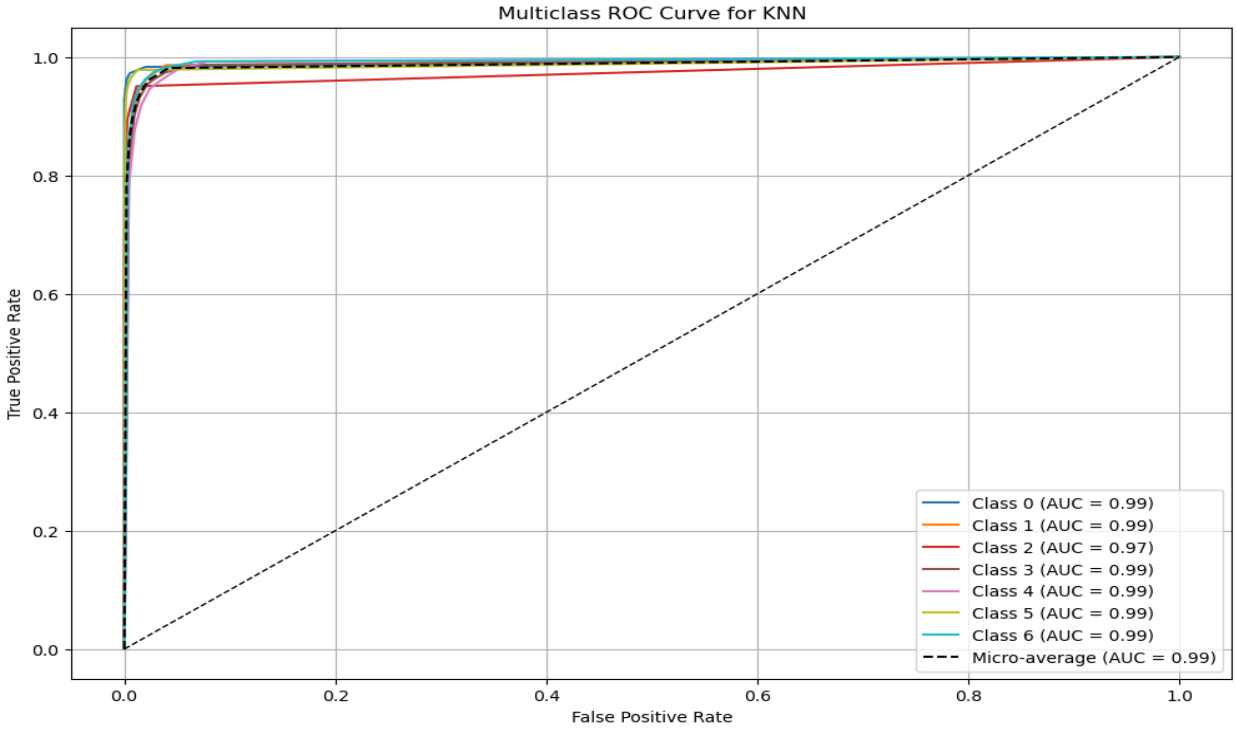
Confusion Matrix:




Confusion Matrix for the KNN
Trouser
Dress
Sandal
Shirt
Sneaker
Bag
Ankle boot
True label
Predicted label
1996 51 0 42 0 0 1
16 1971 0 135 0 5 0
0 0 1671 27 229 6 155
5 67 0 2006 0 22 1
0 0 7 0 1957 1 101
0 14 0 84 16 2033 2
0 0 4 3 71 0 2001
2000
1750
1500
1250
1000
750
500
250
0

ROC and AUC Result:





Loss Curve and Accuracy Curve: