[1]UTKFace dataset dataset link

About dataset :

UTKFace dataset is a large-scale face dataset with long age span (range from 0 to 116 years old). The dataset consists of over 20,000 face images with annotations of age, gender, and ethnicity.

The images cover large variation in pose, facial expression, illumination, occlusion, resolution, etc.

This dataset could be used on a variety of tasks, e.g., face detection, age estimation, age progression/regression, landmark localization, etc.

Labels:

The labels of each face image is embedded in the file name, formated like [age]_[gender]_[race]_[date&time].jpg

[age] is an integer from 0 to 116, indicating the age.

[gender] is either 0 (male) or 1 (female).

[race] is an integer from 0 to 4, denoting White, Black, Asian, Indian, and Others (like Hispanic, Latino, Middle Eastern.

[date&time] is in the format of yyyymmddHHMMSSFFF, showing the date and time an image was collected to UTKFace.

Implementation details :

We used two algorithms K-Means and Logistic Regression

Pre-Processing :

Extracts features by HOG algorithm , And reading age, gender, and race in lists, Compile a list of age, gender, race and photo features in Dataset.

We got 2190 features per each image , but we sum all this features in one new feature called features

Save this dataset into file .csv

Then normalize dataset by Min-Max Normalization algorithm.

Split data into features and target

features : age , race and image

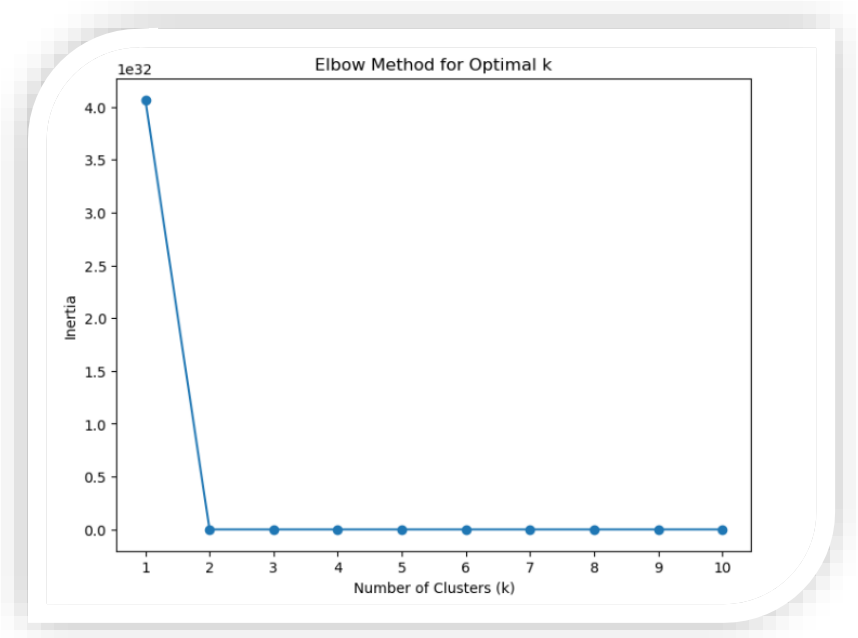
target : gender

And split dataset into 20% test and 80% train by train_test_slpit algoritm.

A)K-Means :

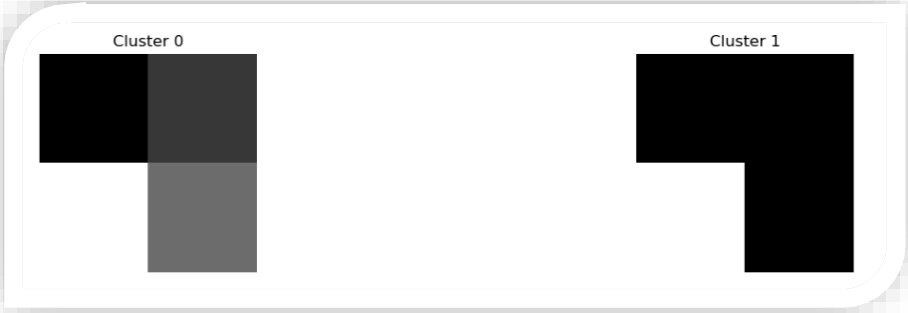
Goal : Classify humans into Male and Female (0 or 1)

Explain Code : select K randomly and loop from 2 to 11 , in each iteration save inertia in lest after end loop compare all inertias and plot Elbow to select right K.



Result:

-The Inertia: 15865.289326214694
-The silhouette score is: 0.9999067077152719
-Visualize the cluster :



B)Logistic Regression :

Goal : Detect Gender

Explain Code: We don't need to sum image features , we used dataset with all image features

Result :

Train Score : 0.8796616685890042

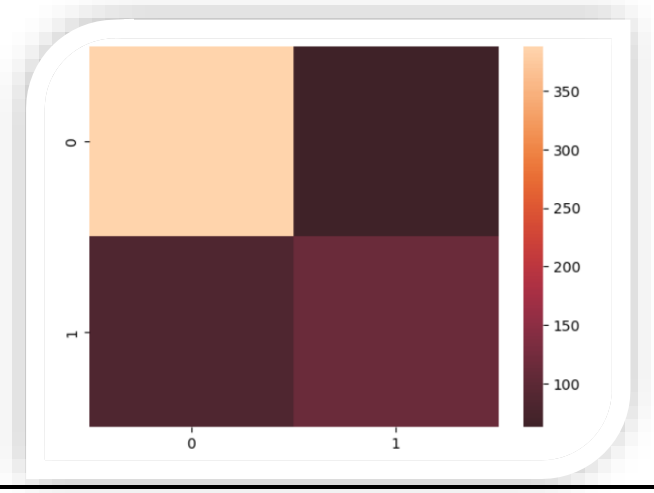
Test Score : 0.7741935483870968

Classes is : [0 1]

Iterations is : [100]

Intercept is : [-4.80416957]

Confusion Matrix :
[[388 63]
[84 116]]



Accuracy Score : 504

F1 Score is : 0.7741935483870968

Recall Score is : 0.7741935483870968

Precision Score is : 0.7741935483870968

Precision Recall Score is : (0.7741935483870968, 0.7741935483870968, 0.7741935483870968, None)

Precision Value is : [0.30721966 0.64804469 1.]

Recall Value is : [1. 0.58 0.]

Thresholds Value is : [0 1]

AUC Value : 0.7201552106430156

fpr Value : [0. 0.13968958 1.]

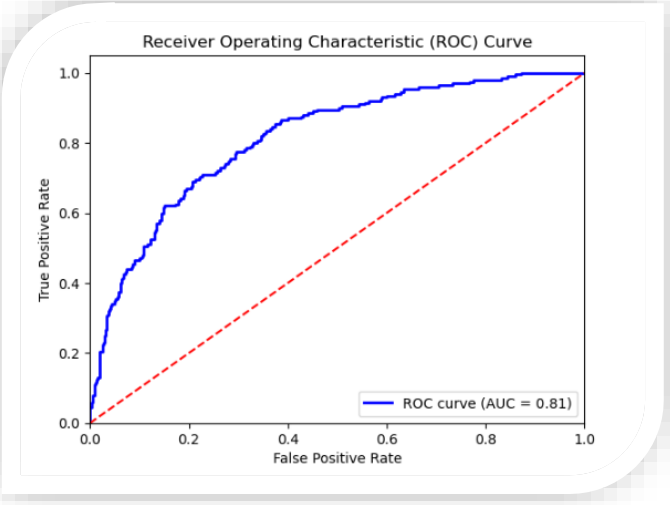
tpr Value : [0. 0.58 1.]

thresholds Value : [inf 1. 0.]

ROCAUC Score : 0.7201552106430156

Zero One Loss Value : 147

RCO curve :



[2]Data Science Salaries 2023 dataset link

About dataset :

These dataset are used to analyze trends such as how salaries vary based on location, experience, education, or skills.

They can help in making predictions about salary ranges based on different factors or in understanding the job market dynamics within the data science field.

Labels:

1) ***work_year***: The year the salary was paid.
2) ***experience_level***: The experience level in the job during the year
3) ***employment_type***: The type of employment for the role
4) ***job_title***: The role worked in during the year.
5) ***salary***: The total gross salary amount paid.
6) ***salary_currency***: The currency of the salary paid as an ISO 4217 currency code.
7) ***salaryinusd:*** The salary in USD
8) ***employee_residence***: Employee's primary country of residence in during the work year as an ISO 3166 country code.
9) ***remote_ratio***: The overall amount of work done remotely
10) ***company_location***: The country of the employer's main office or contracting branch
11) ***company_size***: The median number of people that worked for the company during the year

Implementation details :

Pre-Processing :

a) Encoding all text into integer by LabelEnocder algorithm
b) Fill all NaN values with the median of each column
c) Check if any outliers by IQR method
d) Normalize dataset by Min-Max Normalization
e) Split data into features and target values
f) Select features by features-selection algorithm
g) Split into train and test by train-test-split where test size is 20% and train size is 80%

a) K Nearest Neighbors (KNN)

Goal : detect salary

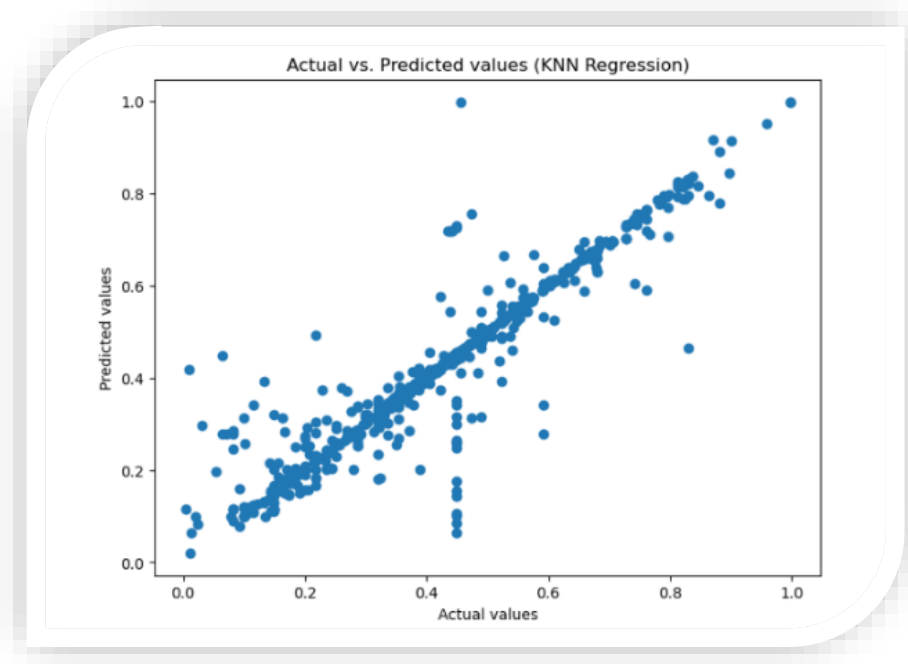
Code : set K randomly and loop in range 11 , after end loop compare all scores and select best K

Result :

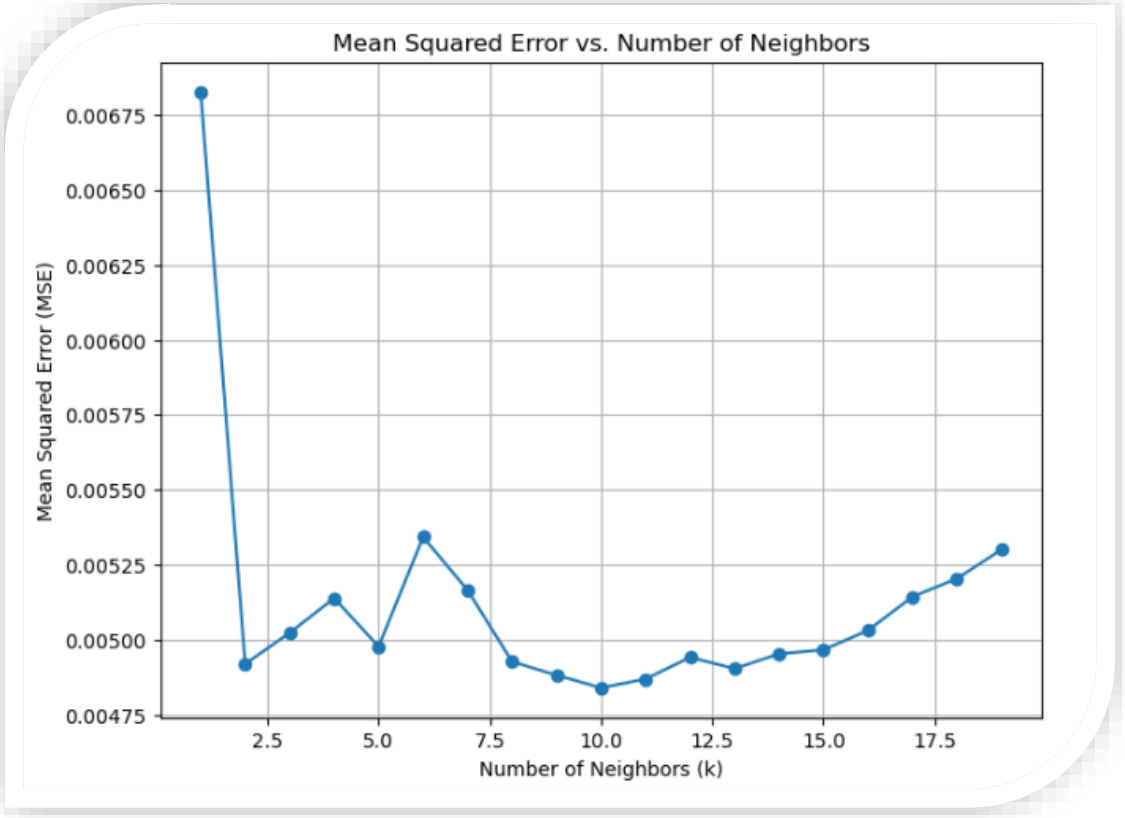
Test Score : 0.8666841697104013
Train Score : 0.9251391538190936
Cross-validation scores: [0.81268325 0.79926669 0.84402415 0.80113223 0.84796548]
Mean accuracy: 0.8210143603928592
Mean Absolute Error (MAE): 0.027364538891455386
Mean Squared Error (MSE): 0.004918238964241548
Root Mean Squared Error (RMSE): 0.07013015730940254
R-squared (R2): 0.8666841697104013
***ROC* curve** : not support in regression but we used alternative way to visualize the performance of a regression model's predictions.

The blow scatter plot represents the actual target values (y_test) against the predicted values (y_pred).

A perfect model would show a perfect linear relationship where all points align on a diagonal line, indicating that the predictions perfectly match the true values. However, in practical scenarios, deviations from this ideal line are expected.



loss curve: The blow plot illustrates how the Mean Squared Error (MSE) changes concerning the number of neighbors 'k' in the KNN regression model. It helps identify the 'k' value that yields the least error on the test set, providing insights into the model's performance with different neighborhood sizes.



b) Linear Regression

c) Goal : **detect salary**

d) Code : **used linear regression from sklearn.lineae_model**

e) Result :

Test Score : 0.8906410931985782

Train Score : 0.853805468396135

Cross-validation scores: [0.85101167 0.83872072 0.84639412 0.85076223 0.87594976]

Mean accuracy: 0.8525677001123915

Mean Absolute Error (MAE): 0.026590301449671477

Mean Squared Error (MSE): 0.004034428884771201

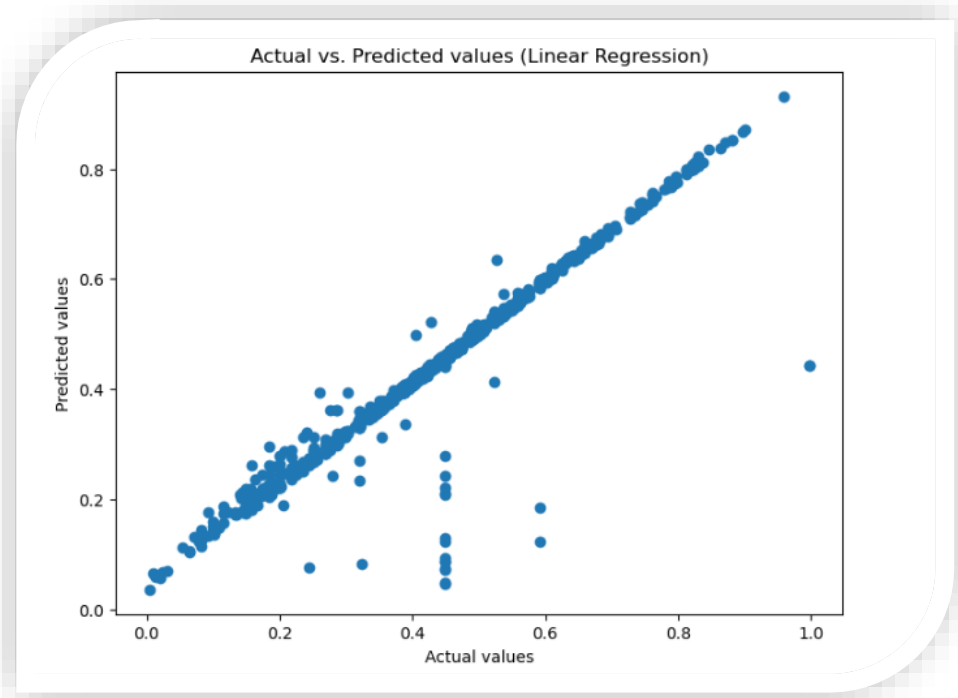
Root Mean Squared Error (RMSE): 0.06351715425592681

R-squared (R2): 0.8906410931985782

***ROC* curve** : not support in regression but we used alternative way to visualize the performance of a regression model's predictions.

The blow scatter plot represents the actual target values (y_test) against the predicted values (y_pred).

A perfect model would show a perfect linear relationship where all points align on a diagonal line, indicating that the predictions perfectly match the true values. However, in practical scenarios, deviations from this ideal line are expected.



Team members

Students of the Faculty of Computers and Artificial Intelligence at Helwan University, Department of CS & AI

Course : Machin Learning (fall 2023)

Name	ID	Dept.
Ossama Samir	20210140	CS
Abd-ulla Mohamed	20210556	CS
Marowa Omar	20210900	CS
Rana Essam	20210335	CS
Toka Mohamed	20210242	CS
Ahmed Mohamed	20210102	AI
Omar Nasser	20210628	AI

References :

Kaggle: Your Home for Data Science

Machine Learning Tutorial (geeksforgeeks.org)

What is Machine Learning? | Google for Developers

Python Machine Learning (w3schools.com)

Supervised Machine Learning: Regression and Classification | Coursera

(478) 10 القسم العاشر : مكتبة سايكيتليرن Sklearn Library - YouTube

Natural Language Processing With Python's NLTK Package – Real Python

(478) Machine Learning Algorithms بالعربى - YouTube

(478) Machine Learning New 2023 - YouTube

(478) Tabular Data Preprocessing For Machine Learning Models(in Arabic) - YouTube