

Project#1

Student Name : Omar Majed Mohammad Badawi .

Student No. : 1202428.

Course : Comp4338 .

Instructor Name : D. Radi Jarrar .

Contents

[Abstract: 3](#_Toc154692838)

[Resources : 3](#_Toc154692839)

[Task 1: 4](#_Toc154692840)

[Code snippet : 5](#_Toc154692841)

[Task 2 : 5](#_Toc154692842)

[code snippet : 6](#_Toc154692843)

[Task 3 : 6](#_Toc154692844)

[Code snippet : 7](#_Toc154692845)

[Task 4 : 8](#_Toc154692846)

[Code snippet : 9](#_Toc154692847)

[Task 5 : 9](#_Toc154692848)

[Code snippet : 10](#_Toc154692849)

[Task 6 : 11](#_Toc154692850)

[Code snippet : 11](#_Toc154692851)

[Task 7 : 12](#_Toc154692852)

[Linear Regression Tasks : 12](#_Toc154692853)

[Code sinppet : 12](#_Toc154692854)

[Task 1 : 13](#_Toc154692855)

[Results : 13](#_Toc154692856)

[Task 2 : 13](#_Toc154692857)

[Results : 13](#_Toc154692858)

[Task 3 : 13](#_Toc154692859)

[Results : 14](#_Toc154692860)

[Comparison : 14](#_Toc154692861)

[Classification Task : 15](#_Toc154692862)

[Code Snippet : 15](#_Toc154692863)

[K = 1 16](#_Toc154692864)

[K = 7 17](#_Toc154692865)

[K = 14 18](#_Toc154692866)

[K = 60 19](#_Toc154692867)

[Conclusion of Classification Task : 20](#_Toc154692868)

[Data preprocessing : 20](#_Toc154692869)

[Code snippet : 20](#_Toc154692870)

# Abstract:

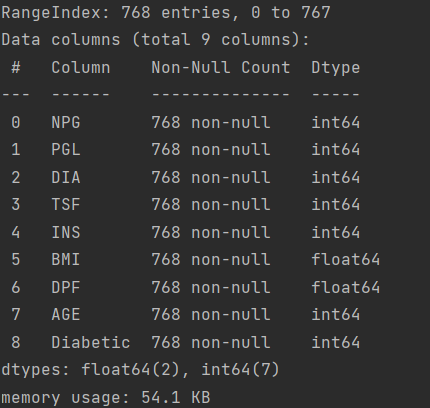
In this project, I used two machine learning techniques, Linear Regression and K Nearest Neighbors (KNN), to predict two features from the provided dataset—AGE and whether someone is diabetic or not. Before running these techniques, I did some data cleanup to make sure the information was in good shape. I created and tested three models for linear regression and four models for KNN with the dataset. The results of these models are explained in detail in different sections of this report.

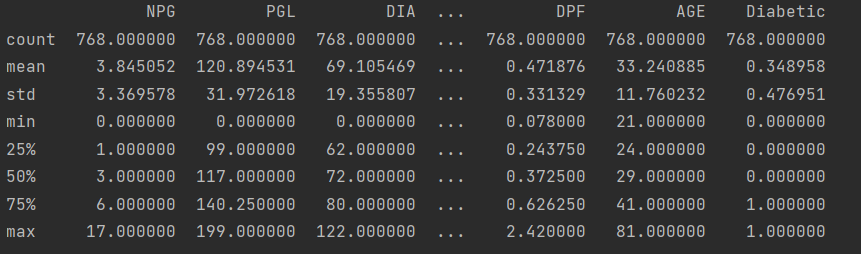
# Resources :

* The provided data set .
* Python libraries : pandas , matplotlib , seaborn and sklearn .
* YouTube.
* Stack overflow.
* ChatGPT.
* Course slides.

# Task 1:

Data set was explored and here is the result of statistics using info and describe methods from pandas library :



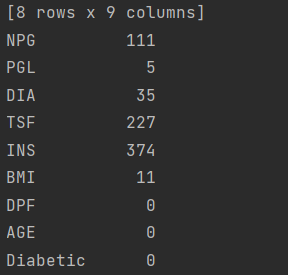


It’s noticed that it shows that there is no missing values ( NA’s )

Since the columns that have missing values use 0 instead of NA .

In order to solve this problem I had to replace all 0’s in all columns with NA’s excluding the last column ( Diabatic column ) where 0 represents a real value .

Here is the results of missing values after the replacement :



## Code snippet :

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

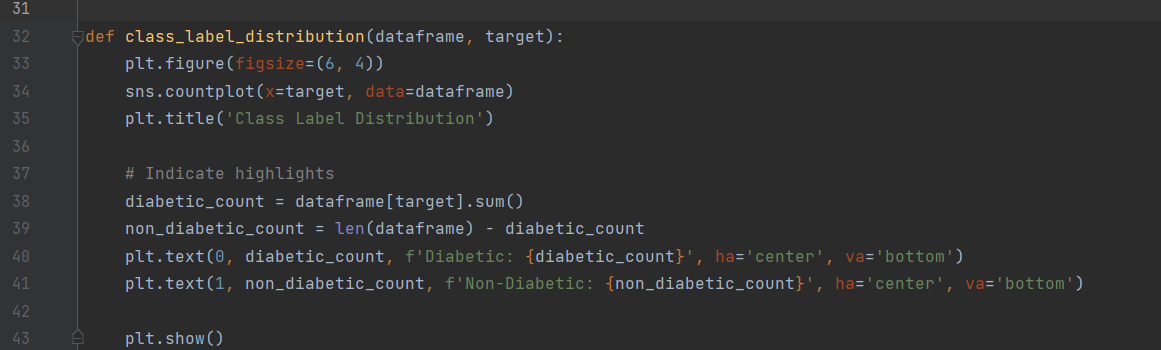
# Task 2 :

Using matplotlib library in python this plot was created which shows the diabetic vs non diabetic counts in the provided data :

A chart of diabetic and diabetes

Description automatically generated

## code snippet :



# Task 3 :

Using matplotlib library in python this plot was created which shows the diabetic vs non diabetic counts in the provided data but it separates them into subgroups based on the Age attribute :

A chart of diabetes

Description automatically generated with medium confidence

Code snippet : A computer screen shot of text

Description automatically generated

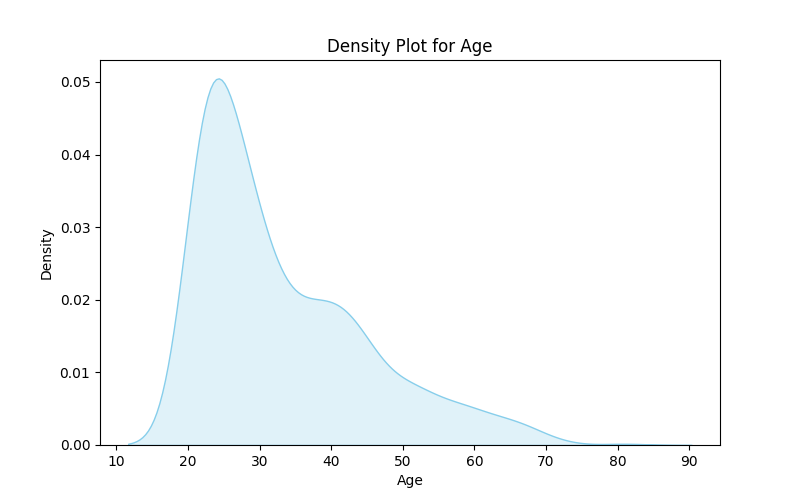
This was achieved by adding a new column called age group which is a categorical column not numeric.

Age groups consist of diabetic people within range of 10 years , this column was dropped later .

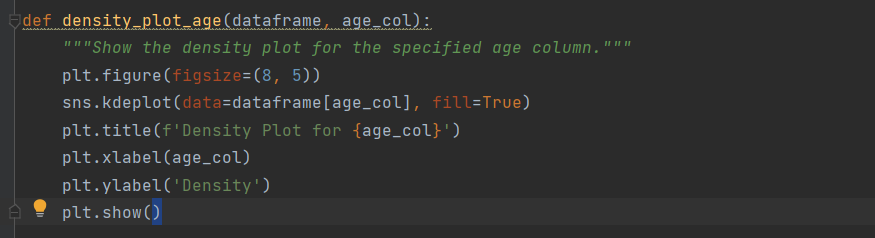
df.drop(columns=['AgeGroup'], inplace=True)

# Task 4 :

Using matplotlib this plot was created which shows the density of data according to age attribute:

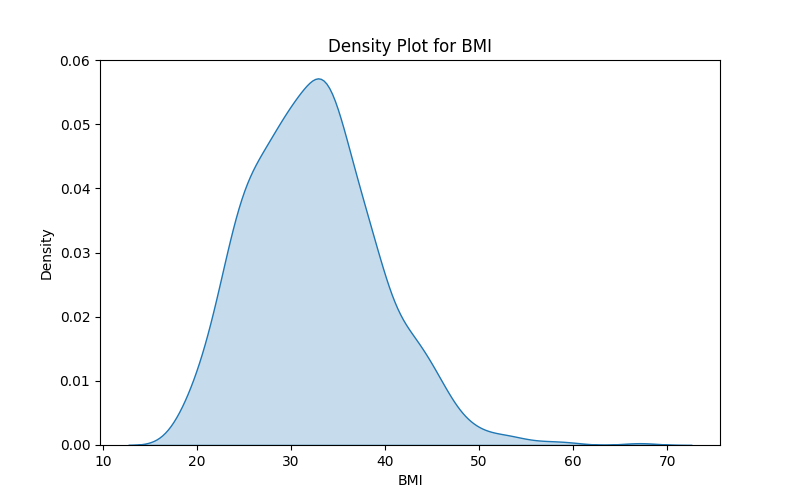


## Code snippet :



# Task 5 :

Using matplotlib this plot was created which shows the density of data according to BMI attribute:



## Code snippet :

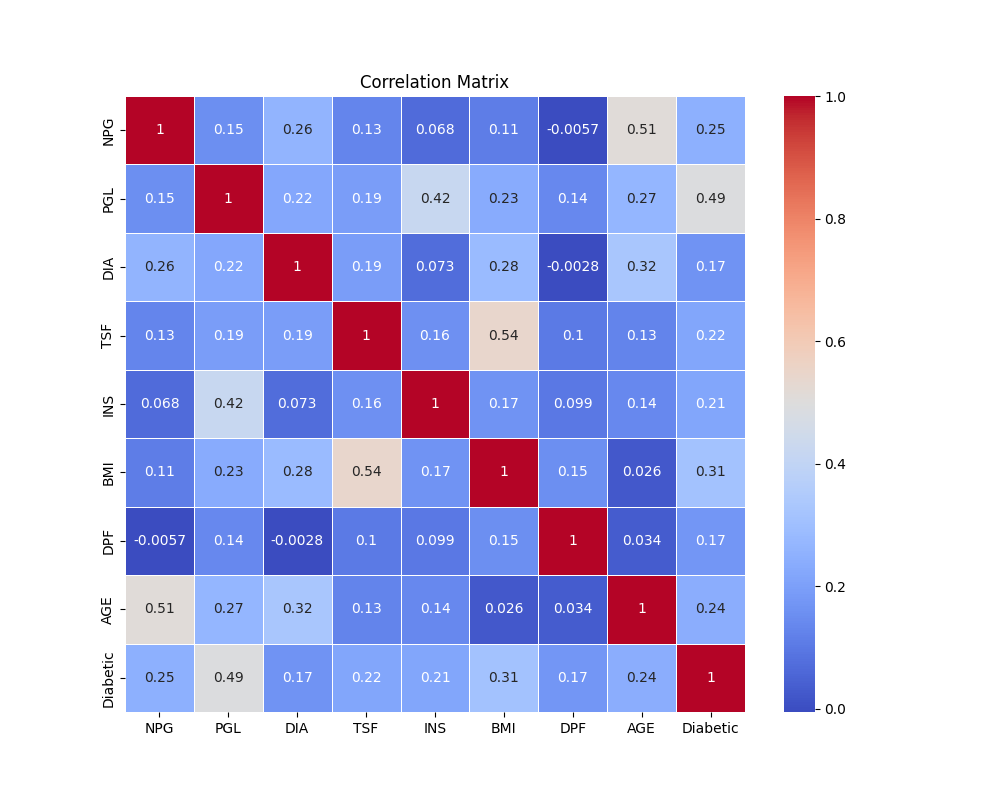
Same function was used by passing the target column name in order to reuse code .

A screen shot of a computer program

Description automatically generated

# Task 6 :

Using matplotlib this correlation matrix between features was created and visualized as a plot , this will be utilized later on in tasks of linear regression .



## Code snippet :

A screen shot of a computer code

Description automatically generated

# Task 7 :

Task 7 WILL BE DONE LATER ON WITH LINEAR REGRESSION AND KNN TASKS .

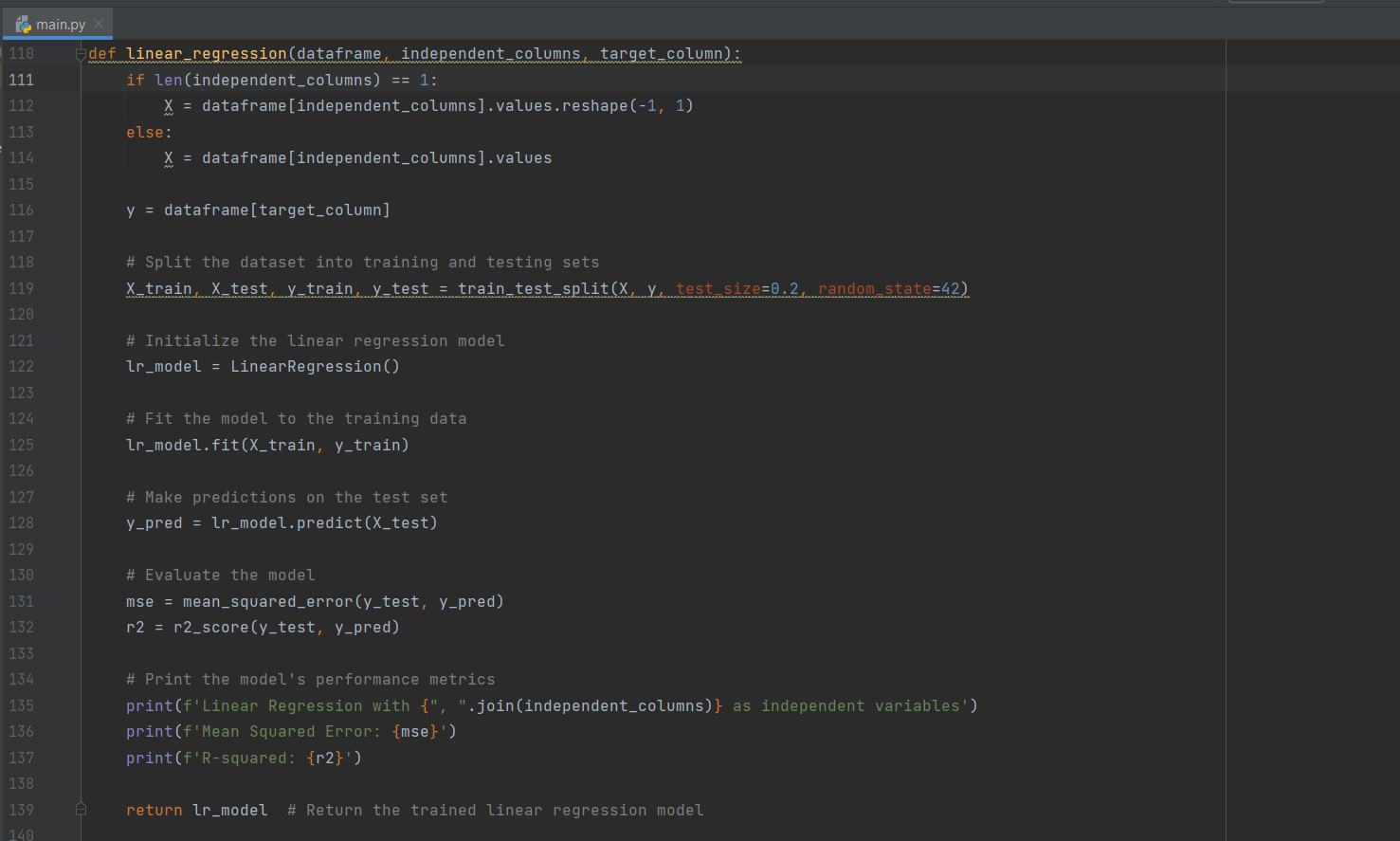
# Linear Regression Tasks :

In this task we utilized skilearn library from python to create linear regression machine learning models

We used the same function that returns the trained and tested model also it prints the testing results and the dataset is split into 0.2 , 0.8 ratio for testing and training .

The same function was used in the 3 tasks with different parameters since the function can take an array of features as independent columns and the target feature .

## Code sinppet :



## Task 1 :

In this task we created a machine learning model and called it LR1 which relies on linear regression technique , this model can predict the AGE based on all other features in the data set .

### Results :

A number on a black background

Description automatically generated

## Task 2 :

In this task we created a machine learning model and called it LR2 which relies on linear regression technique , this model can predict the AGE based on NPG ( Which AGE is most connected to this feature ) according to the previously created Correlation matrix .

### Results :

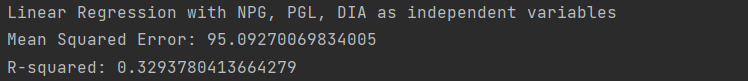
A black background with white text

Description automatically generated

## Task 3 :

In this task we created a machine learning model and called it LR2 which relies on linear regression technique , this model can predict the AGE based on NPG , PGL and DIA ( Which AGE is most connected to these features ) according to the previously created Correlation matrix .

### Results :



## Comparison :

As shown in results we can see that LR1 gave us the best results

While LR2 gave the worst results , on the other hand LR3 gave us some kind of balanced result .

We cay objectify these results in the following way , in LR2 we faced what is called underfitting since we used only 1 feature to predict the AGE which caused our model to not recognize the whole patterns in the data set , but it still demonstrates a strong connection between NPG and AGE .

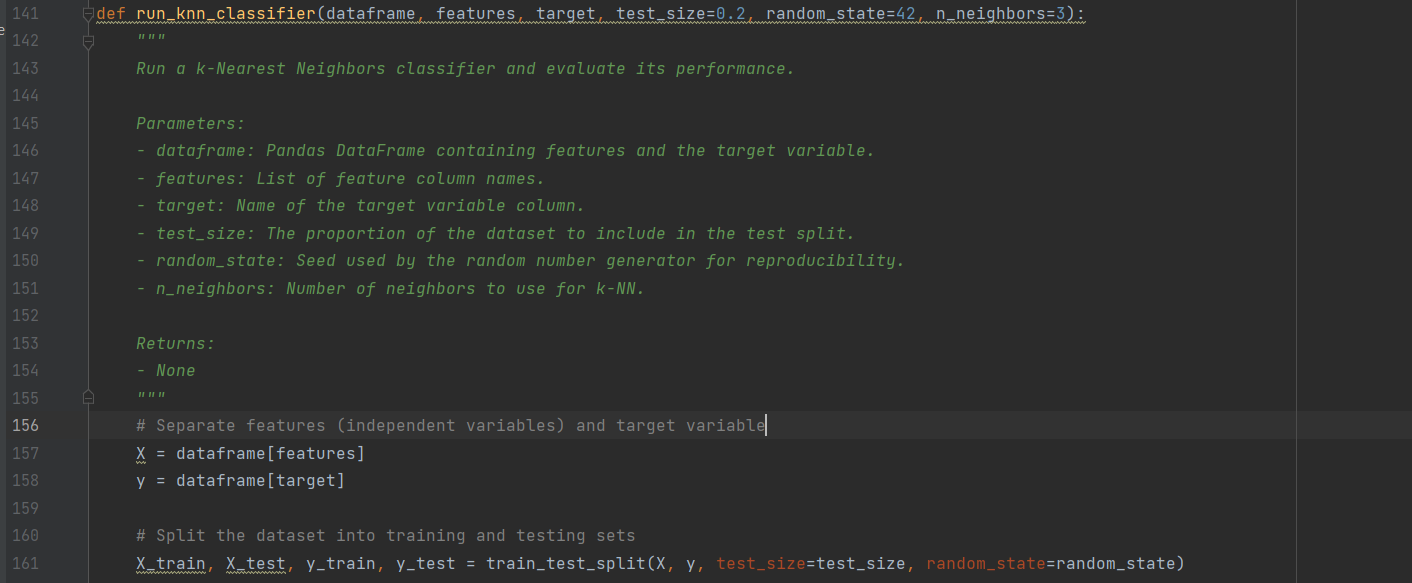
In LR1 our model can expose most of the patterns laying there in data so it occurs to be has the best results still we can’t say we faced overfitting since the results aren’t super accurate .

In LR3 it was someway balanced it’s neither overfitting nor underfitting when we used the 3 most important features to predict the AGE feature .

# Classification Task :

A function was wrote to implement the KNN technique and used for the 4 created models , where we can pass parameters to the function Including K or ( n = number of neighbors ) .

## Code Snippet :





A screen shot of a computer code

Description automatically generated

## K = 1

A screenshot of a computer screen

Description automatically generated

A graph of a line with blue and orange lines

Description automatically generated

## K = 7

A screenshot of a computer

Description automatically generated A graph of a curve

Description automatically generated

## K = 14

A screenshot of a computer

Description automatically generated

A graph of a curve

Description automatically generated

## K = 60

A screenshot of a computer screen

Description automatically generated

A graph of a curve

Description automatically generated

## Conclusion of Classification Task :

Number of neighbors can effectively change the accuracy of the model and it should be balanced in order to avoid underfitting and overfitting , also that there is no optimal number of negihbors but it actually depends on the data provided ant it’s size and its characteristics .

# Data preprocessing :

Since our data set is relatively small the approach of dropping rows that contain missing values in its columns wasn’t a good choice , so I followed another approach to replace all NA’s with the column mean

Which will keep things balanced and we will avoid us losing a lot of our data .

## Code snippet :

