Mohammed A. El-Agha

Data Mining

- Data mining is the process of discovering patterns in large data sets
- knowledge discovery in databases

- 2 main types
  - Supervised Learning
  - Unsupervised Learning

Data Mining

- Supervised
    - Classification
    - Regression

- Unsupervised
    - Clustering
    - Outlier Analysis

Python

- Why?
  - Simple syntax
  - Understandable semantic
  - A lot of ready-used libraries
  - Most of methods, techniques, metrics are single-line function

Project

- Dataset
  - Student Performance Data Set

- Paper
  - User Data Mining to Predict Secondary School Student Performance

- The used Data Mining methods are:
  - Decision Tree Classification
  - Nearest Neighbor Classification
  - Linear Regression
  - Kmeans clustering
  - Generalized ESD Outlier Analysis

Paper

- Using Data Mining to Predict Secondary School Student Performance
- 2008

- study students assessment in secondary schools in Portugal
  using their grades in two courses: Mathematics and Language

Paper

- The paper presents three supervised methods which are:
  - Binary Classification
  - 5-Level Classification
  - Regression

- Using
  - Decision Tree (DT)
  - Random Forest (RF)
  - Neural Networks (NN)
  - Support Vector Machine (SVM)

Dataset

- The data is from University of Minho in Portugal

- student assessment in Portuguese language course from two schools

- consists of two sub data sets; one of Language course and other for math course

Dataset

- 649 case with 31 feature, including
  - personal factors, such as: sex and age
  - living conditions, such as: urban or rural address and home to school travel time
  - health factor
  - social factors, such as: family size, quality of family relationships, parent's cohabitation status, student's guardian, mother's education, father's education, mother's job, father's job
  - entertainment factors, such as: romantic relationship, free time after school, going out with friends, Internet access at home
  - educational factors, such as: weekly study time, extra educational support, extra paid classes, desire to study higher

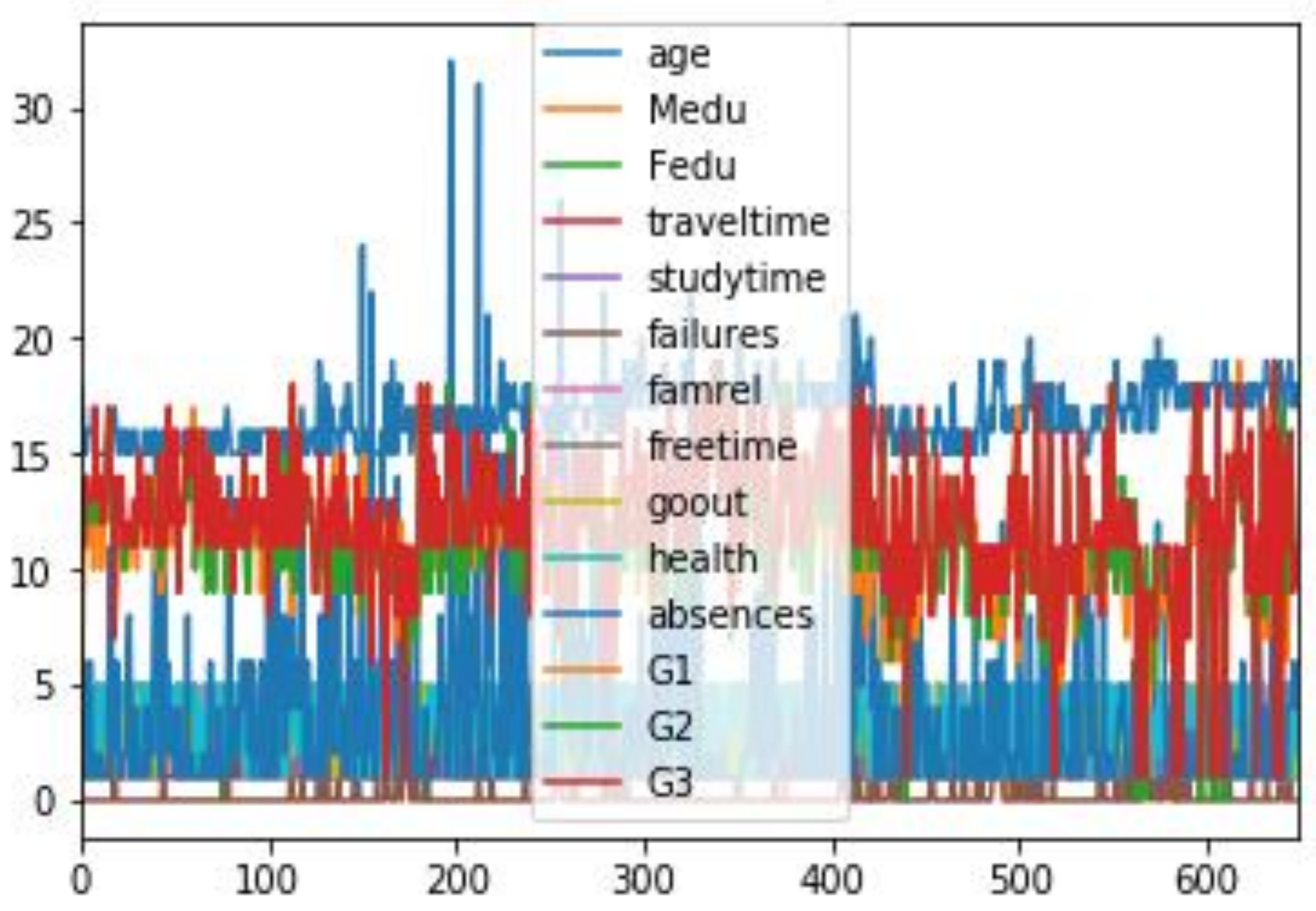
Data Preprocessing

- Configurations
- Remove irrelevant features
  - school, famsize, reason, traveltime, nursery, guardian
- Remove similar values features
  - age
- Remove redundant features
  - Medu, Fedu, Pstatus, G1, G2
- Fill NA/None by zero
  - failures, studytime, famrel, freetime, goout, health
- Convert nominal string to nominal integer
- Discretization
  - absences: bin 1 [0-24], bin 2 [25-49], bin 3 [50-74], bin 4 [75-100]
  - G3: Fail [0-9], Pass [10-14], Good [15-20]

Data Preprocessing

- After data preprocessing, 19 features are still. The 19th is G3 which is the target class.

Data Visualization



Used Data Mining Methods

- Decision Tree Classification
    - training set is 65% and the testing set is 35%
- Nearest Neighbors Classification
    - training set is 60% and the testing set is 40%
- Linear Regression
    - training set is 60% and the testing set is 40%
- K-Means Clustering
    - K = 3
- Generalized ESD Outlier Analysis
    - Number of output outliers is 10, and outlier ratio is 0.1

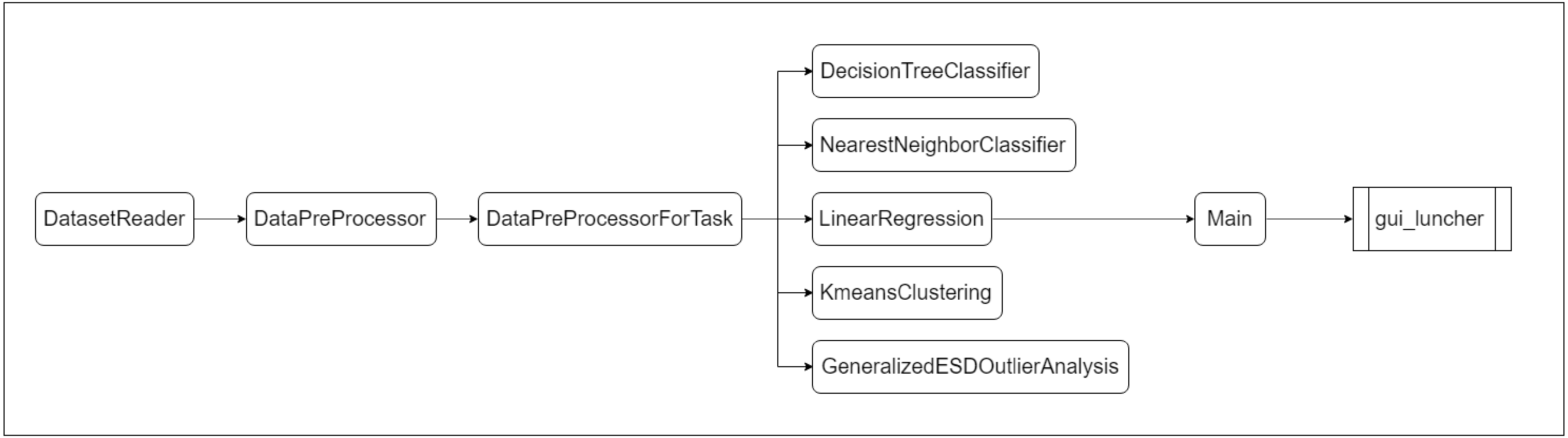
Used Python Libraries

- DataFrame
- Pandas
- Matplotlib
- Sklearn
- PyAstronomy
- tkinter

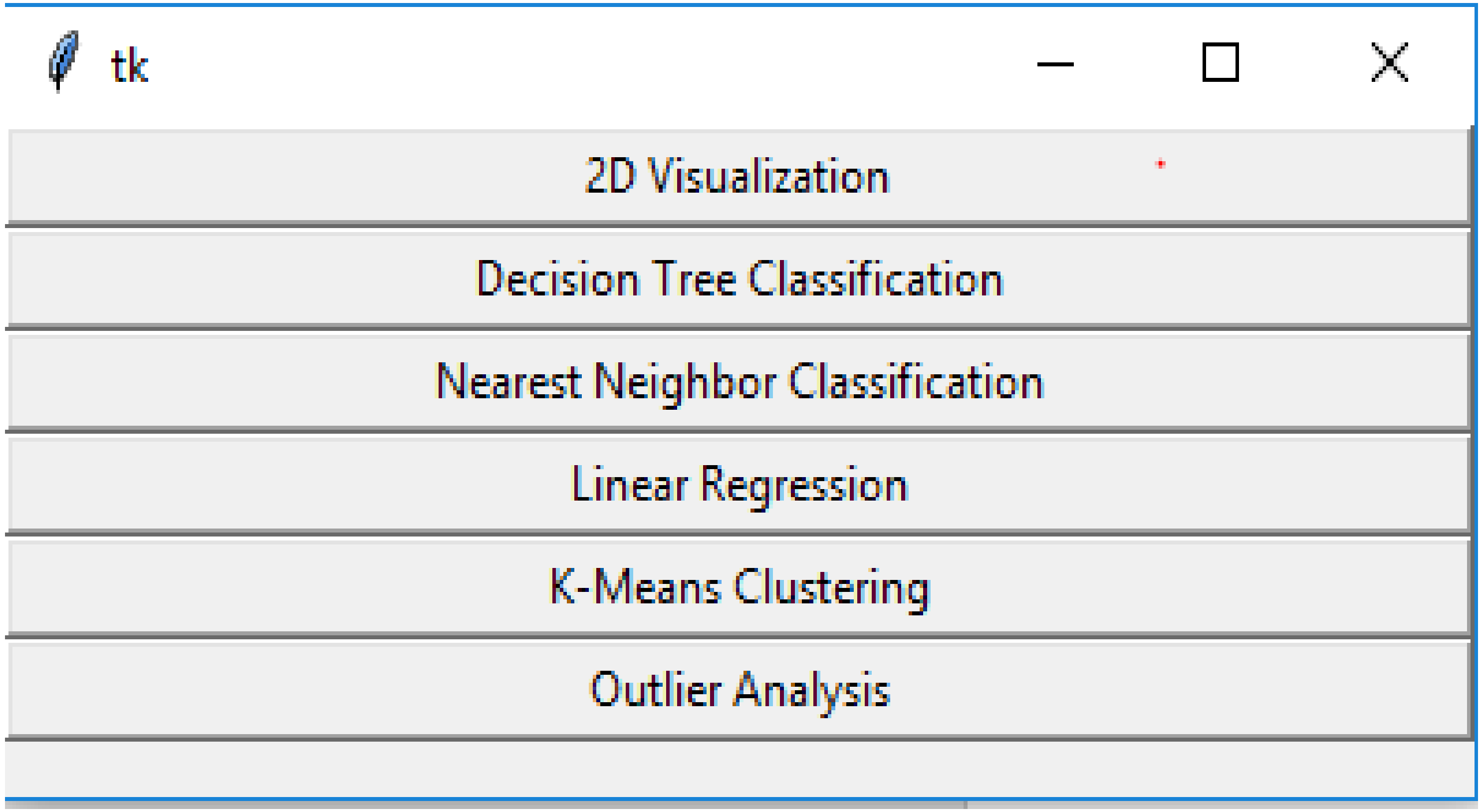
Code

File	Class/es
datasetreader	DatasetReader
preprocess	DataPreProcesses DataPreProcessorForTask
classification	DecisionTreeClassifier NearestNeighborClassifier
regression	LinearRegression
clustering	KmeansClustering
outlier_analysis	GeneralizedESDOutlierAnalysis
main	Main
gui_luncher	

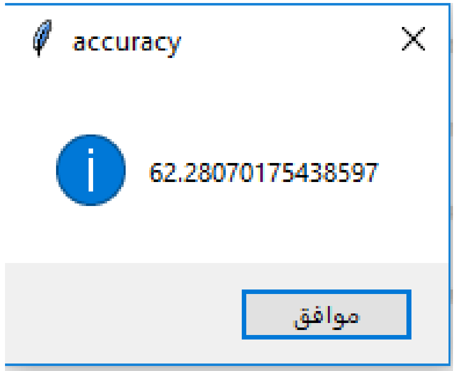
Code


DatasetReader
DataPreProcessor
DataPreProcessorForTask
DecisionTreeClassifier
NearestNeighborClassifier
LinearRegression
KmeansClustering
GeneralizedESDOutlierAnalysis
Main
gui_luncher

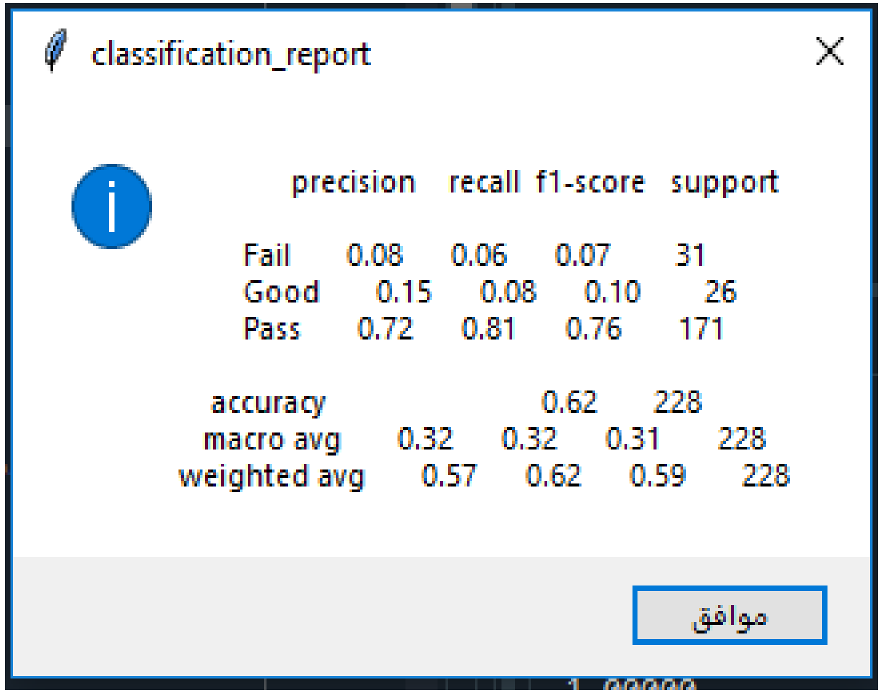
Results


tk
2D Visualization
Decision Tree Classification
Nearest Neighbor Classification
Linear Regression
K-Means Clustering
Outlier Analysis

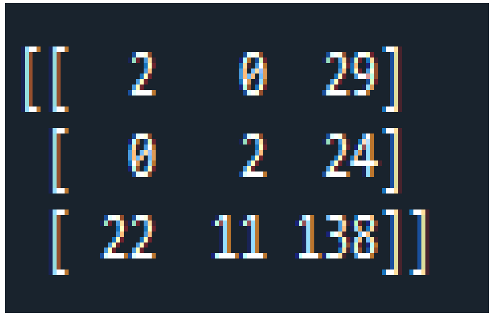
Results (DT)



62.28070175438597

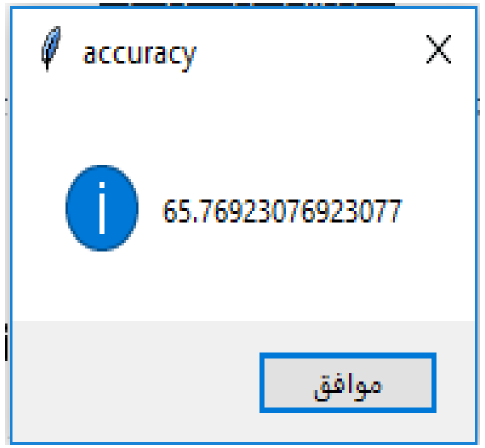


```
              precision    recall  f1-score   support

        Fail       0.08      0.06      0.07        31
        Good       0.15      0.08      0.10        26
        Pass       0.72      0.81      0.76       171

    accuracy                           0.62       228
   macro avg       0.32      0.32      0.31       228
weighted avg       0.57      0.62      0.59       228
```

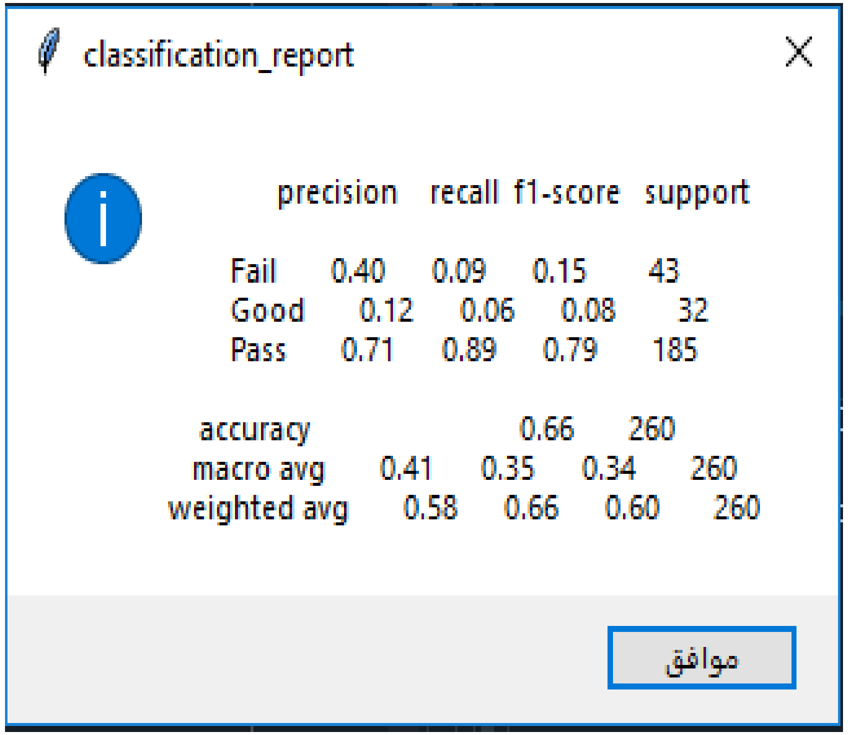


```
[[  2   0  29]
 [  0   2  24]
 [ 22  11 138]]
```

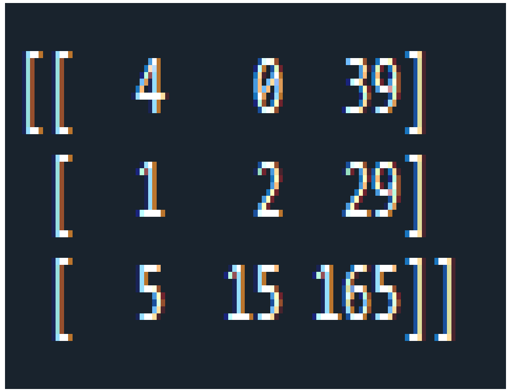
Results (KNN)



```
65.76923076923077
```

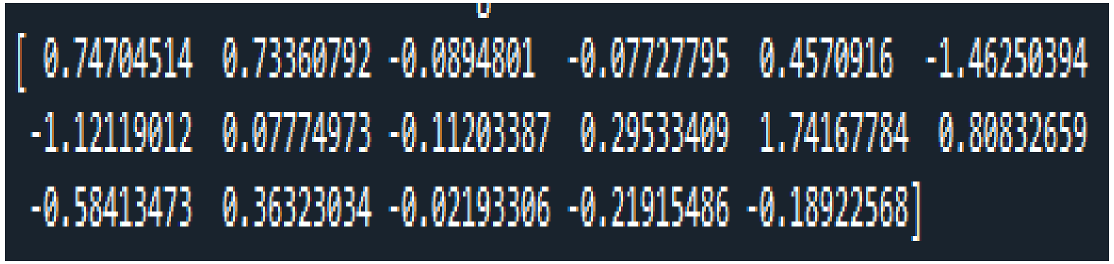


```
              precision    recall  f1-score   support

        Fail       0.40      0.09      0.15        43
        Good       0.12      0.06      0.08        32
        Pass       0.71      0.89      0.79       185

    accuracy                           0.66       260
   macro avg       0.41      0.35      0.34       260
weighted avg       0.58      0.66      0.60       260
```

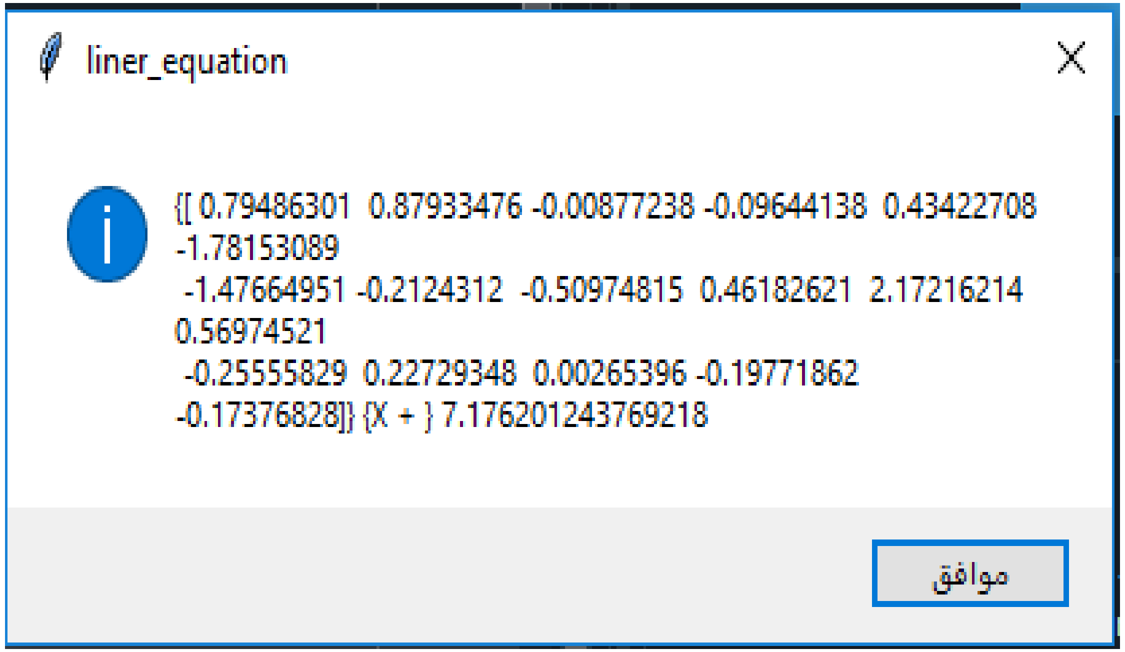


```
[[  4   0  39]
 [  1   2  29]
 [  5  15 165]]
```

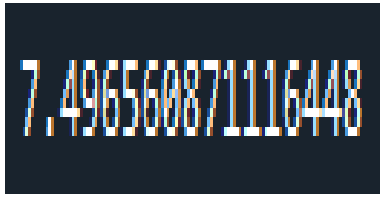
Results (Linear Regression)



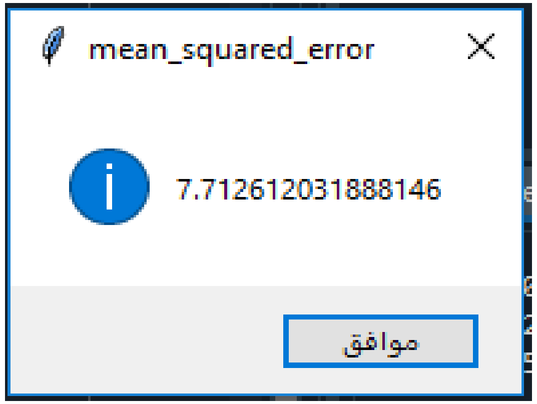
```
[ 0.74704514  0.73360792 -0.0894801  -0.07727795  0.4570916  -1.46250394
 -1.12119012  0.07774973 -0.11203387  0.29533409  1.74167784  0.80832659
 -0.58413473  0.36323034 -0.02193306 -0.21915486 -0.18922568]
```



```
{[ 0.79486301  0.87933476 -0.00877238 -0.09644138  0.43422708
-1.78153089
 -1.47664951 -0.2124312  -0.50974815  0.46182621  2.17216214
0.56974521
 -0.25555829  0.22729348  0.00265396 -0.19771862
-0.17376828]} {X + } 7.176201243769218
```

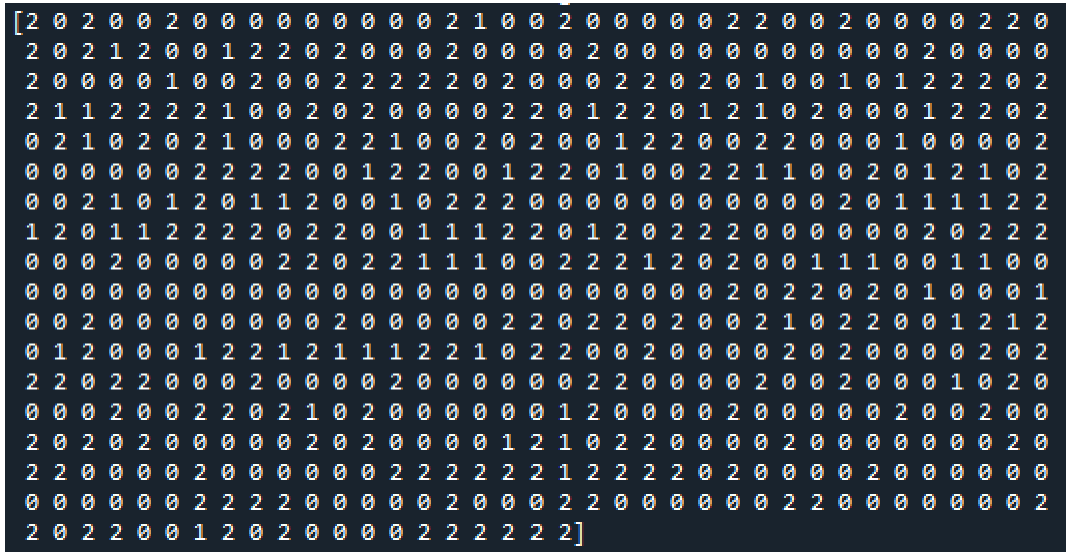


```
7.496560871116448
```

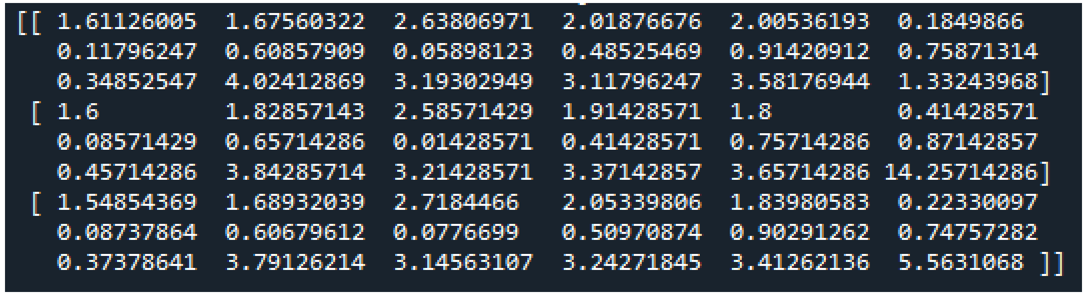


```
7.712612031888146
```

Results (K-Means)

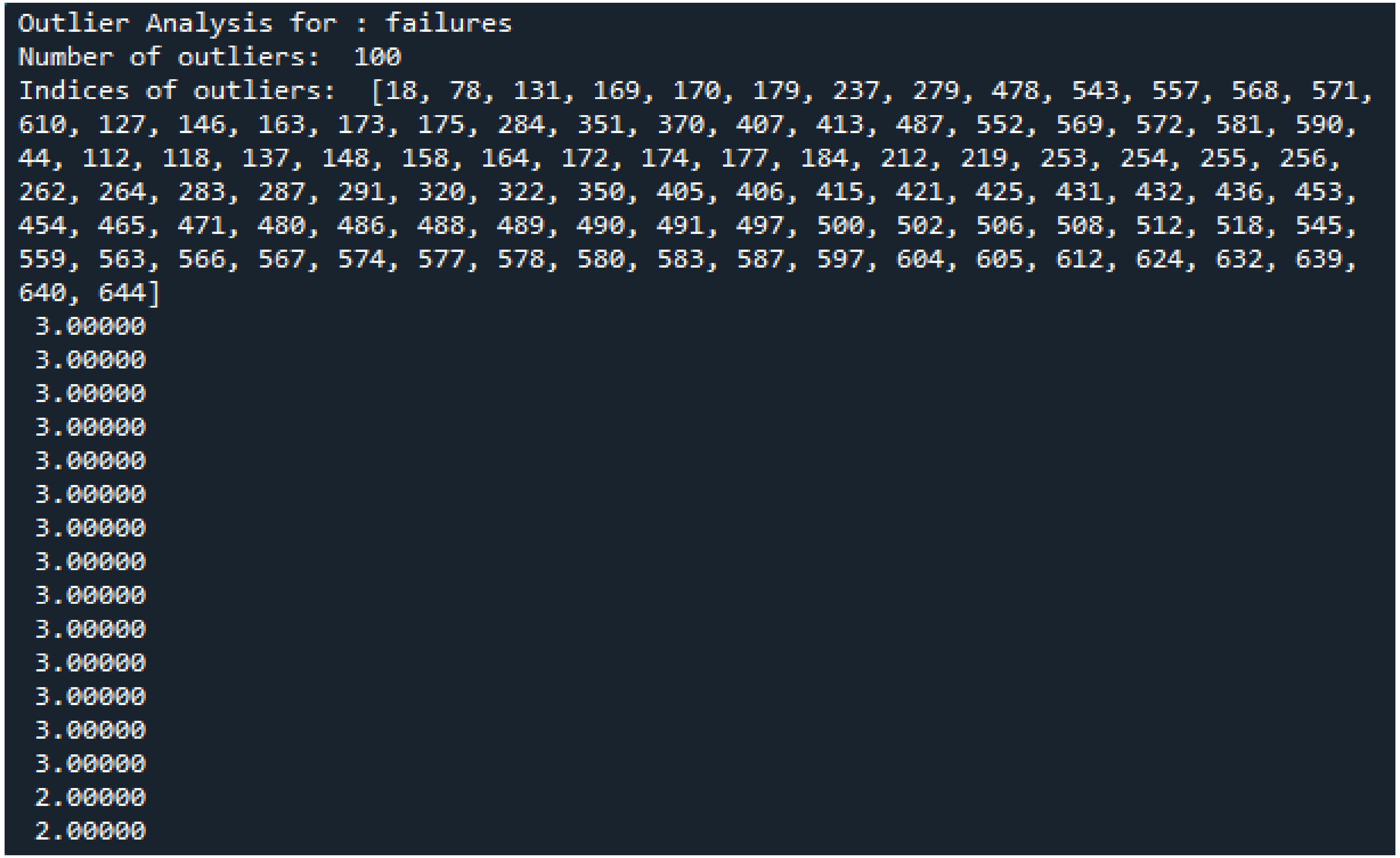


```
[2 0 2 0 0 2 0 0 0 0 0 0 0 0 0 2 1 0 0 2 0 0 0 0 0 2 2 0 0 2 0 0 0 0 2 2 0
 2 0 2 1 2 0 0 1 2 2 0 2 0 0 0 2 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0
 2 0 0 0 0 1 0 0 2 0 0 2 2 2 2 2 0 2 0 0 0 2 2 0 2 0 1 0 0 1 0 1 2 2 2 0 2
 2 1 1 2 2 2 2 1 0 0 2 0 2 0 0 0 0 2 2 0 1 2 2 0 1 2 1 0 2 0 0 0 1 2 2 0 2
 0 2 1 0 2 0 2 1 0 0 0 2 2 1 0 0 2 0 2 0 0 1 2 2 0 0 2 2 0 0 0 1 0 0 0 0 2
 0 0 0 0 0 0 2 2 2 2 0 0 1 2 2 0 0 1 2 2 0 1 0 0 2 2 1 1 0 0 2 0 1 2 1 0 2
 0 0 2 1 0 1 2 0 1 1 2 0 0 1 0 2 2 2 0 0 0 0 0 0 0 0 0 0 0 2 0 1 1 1 1 2 2
 1 2 0 1 1 2 2 2 2 0 2 2 0 0 1 1 1 2 2 0 1 2 0 2 2 2 0 0 0 0 0 0 2 0 2 2 2
 0 0 0 2 0 0 0 0 0 2 2 0 2 2 1 1 1 0 0 2 2 2 1 2 0 2 0 0 1 1 1 0 0 1 1 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 2 2 0 2 0 1 0 0 0 1
 0 0 2 0 0 0 0 0 0 0 0 2 0 0 0 0 0 2 2 0 2 2 0 2 0 0 2 1 0 2 2 0 0 1 2 1 2
 0 1 2 0 0 0 1 2 2 1 2 1 1 1 2 2 1 0 2 2 0 0 2 0 0 0 0 2 0 2 0 0 0 0 2 0 2
 2 2 0 2 2 0 0 0 2 0 0 0 0 2 0 0 0 0 0 0 2 2 0 0 0 0 2 0 0 2 0 0 0 1 0 2 0
 0 0 0 2 0 0 2 2 0 2 1 0 2 0 0 0 0 0 0 1 2 0 0 0 0 2 0 0 0 0 0 2 0 0 2 0 0
 2 0 2 0 2 0 0 0 0 0 2 0 2 0 0 0 0 1 2 1 0 2 2 0 0 0 0 2 0 0 0 0 0 0 0 2 0
 2 2 0 0 0 0 2 0 0 0 0 0 0 2 2 2 2 2 2 1 2 2 2 2 0 2 0 0 0 0 2 0 0 0 0 0 0
 0 0 0 0 0 0 2 2 2 2 0 0 0 0 0 2 0 0 0 2 2 0 0 0 0 0 0 2 2 0 0 0 0 0 0 0 2
 2 0 2 2 0 0 1 2 0 2 0 0 0 0 2 2 2 2 2 2]
```



```
[[ 1.61126005  1.67560322  2.63806971  2.01876676  2.00536193  0.1849866
   0.11796247  0.60857909  0.05898123  0.48525469  0.91420912  0.75871314
   0.34852547  4.02412869  3.19302949  3.11796247  3.58176944  1.33243968]
 [ 1.6         1.82857143  2.58571429  1.91428571  1.8         0.41428571
   0.08571429  0.65714286  0.01428571  0.41428571  0.75714286  0.87142857
   0.45714286  3.84285714  3.21428571  3.37142857  3.65714286 14.25714286]
 [ 1.54854369  1.68932039  2.7184466   2.05339806  1.83980583  0.22330097
   0.08737864  0.60679612  0.0776699   0.50970874  0.90291262  0.74757282
   0.37378641  3.79126214  3.14563107  3.24271845  3.41262136  5.5631068 ]]
```

Results (ESD)



```
Outlier Analysis for : failures
Number of outliers:  100
Indices of outliers:  [18, 78, 131, 169, 170, 179, 237, 279, 478, 543, 557, 568, 571,
610, 127, 146, 163, 173, 175, 284, 351, 370, 407, 413, 487, 552, 569, 572, 581, 590,
44, 112, 118, 137, 148, 158, 164, 172, 174, 177, 184, 212, 219, 253, 254, 255, 256,
262, 264, 283, 287, 291, 320, 322, 350, 405, 406, 415, 421, 425, 431, 432, 436, 453,
454, 465, 471, 480, 486, 488, 489, 490, 491, 497, 500, 502, 506, 508, 512, 518, 545,
559, 563, 566, 567, 574, 577, 578, 580, 583, 587, 597, 604, 605, 612, 624, 632, 639,
640, 644]
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 3.00000
 2.00000
 2.00000
```