CS773-HOMEWORK-1

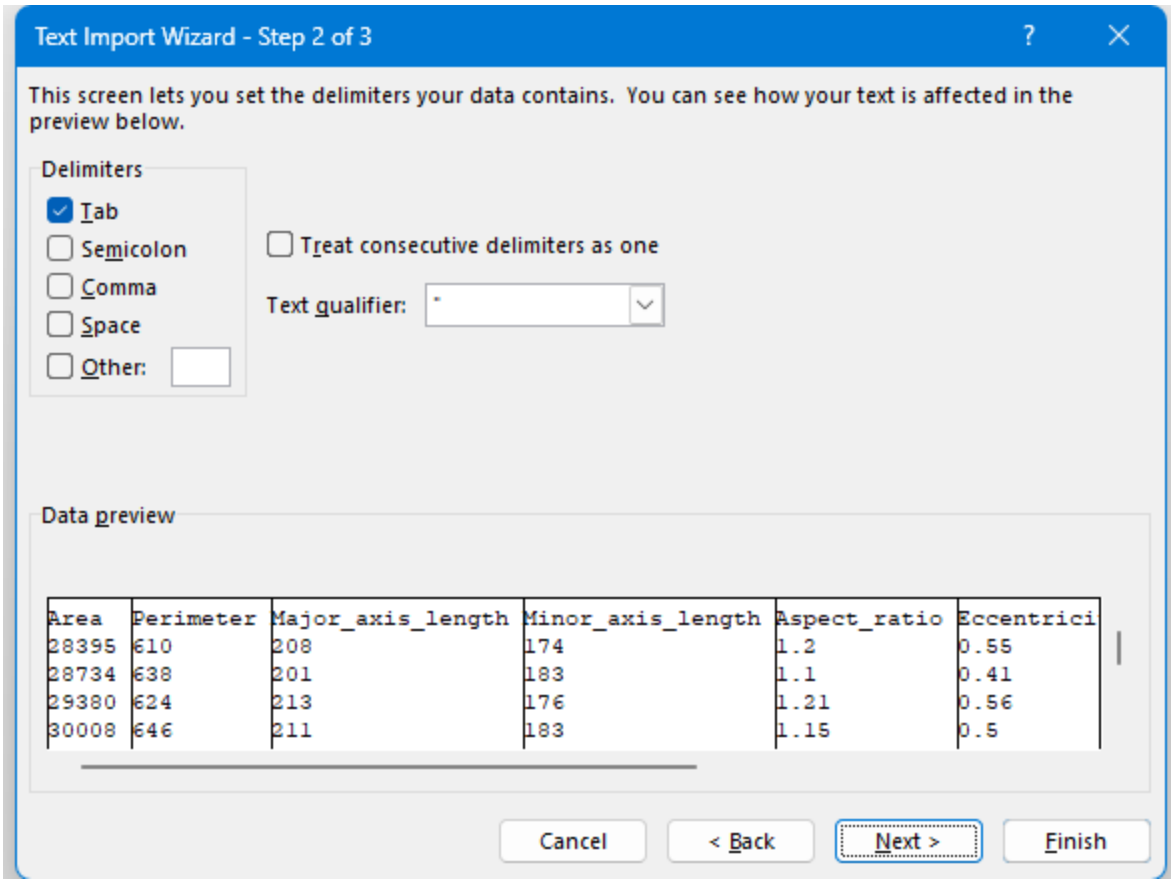
Step 1: Create an ARFF file bean_training_data.arff with the same 9 fields.

Solution

(a) Since the input file is in tab delimited, we need to first convert it into CSV so that Weka can read it. Open bean_training_data.txt with Microsoft Excel or R and convert the file into .CSV.

(b) Read the bean_training_data.csv into Weka and convert it into bean_training_data.arff as shown below. We can also change the field names wile exporting to Weka in ArffViewer.

Screenshots



Input CSV file

```
A,P,L,l,K,Ec,C,Ed,Class
28395,610,208,174,1.2,0.55,28715,190,SEKER
28734,638,201,183,1.1,0.41,29172,191,SEKER
29380,624,213,176,1.21,0.56,29690,193,SEKER
30008,646,211,183,1.15,0.5,30724,195,SEKER
30140,620,202,190,1.06,0.33,30417,196,SEKER
30279,635,213,182,1.17,0.52,30600,196,SEKER
30477,670,211,184,1.15,0.49,30970,197,SEKER
30519,630,213,183,1.17,0.51,30847,197,SEKER
30685,636,214,183,1.17,0.51,31044,198,SEKER
30834,632,217,181,1.2,0.55,31120,198,SEKER
30917,641,214,184,1.16,0.5,31280,198,SEKER
31091,639,210,188,1.12,0.45,31458,199,SEKER
31107,641,215,185,1.16,0.51,31423,199,SEKER
31158,643,216,184,1.18,0.53,31492,199,SEKER
31158,641,212,187,1.13,0.47,31474,199,SEKER
31178,637,213,187,1.14,0.48,31520,199,SEKER
31202,644,216,184,1.17,0.52,31573,199,SEKER
31203,640,215,185,1.16,0.51,31558,199,SEKER
31272,639,212,188,1.13,0.47,31593,200,SEKER
```

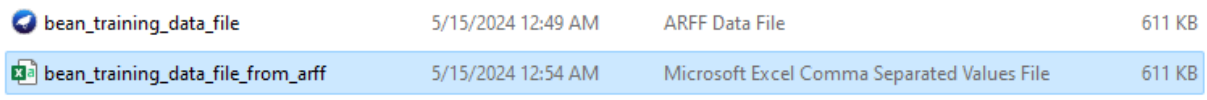
Output Arff File

```
@relation bean_training_data_file

@attribute Area numeric
@attribute Perimeter numeric
@attribute 'Major axis length' numeric
@attribute 'Minor axis length' numeric
@attribute 'Aspect ratio' numeric
@attribute Eccentricity numeric
@attribute 'Convex area' numeric
@attribute 'Equivalent diameter' numeric
@attribute Class {SEKER,BARBUNYA,BOMBAY,CALI,HOROZ,SIRA,DERMASON}

@data
20420,525,184,142,1.29,0.63,20684,161,DERMASON
20464,528,191,136,1.4,0.7,20772,161,DERMASON
20548,525,184,143,1.29,0.63,20825,162,DERMASON
20711,525,186,142,1.31,0.65,20988,162,DERMASON
20786,535,201,132,1.52,0.75,21057,163,DERMASON
20942,531,191,140,1.37,0.68,21191,163,DERMASON
21101,534,185,146,1.27,0.62,21462,164,DERMASON
21314,537,194,140,1.38,0.69,21587,165,DERMASON
21348,531,192,142,1.35,0.67,21590,165,DERMASON
21397,535,193,142,1.36,0.68,21731,165,DERMASON
21405,535,191,143,1.34,0.66,21737,165,DERMASON
21479,544,199,138,1.44,0.72,21762,165,DERMASON
21558,539,197,139,1.41,0.71,21808,166,DERMASON
21570,538,197,140,1.41,0.7,21820,166,DERMASON
```

Step 2. Create a corresponding csv file bean_training_data.csv

Solution

Either we can use Excel or R to convert tab delimited to .CSV file or we can use previously created bean_training_data.arff file into CSV using Arffviewer in Weka.



File Edit View

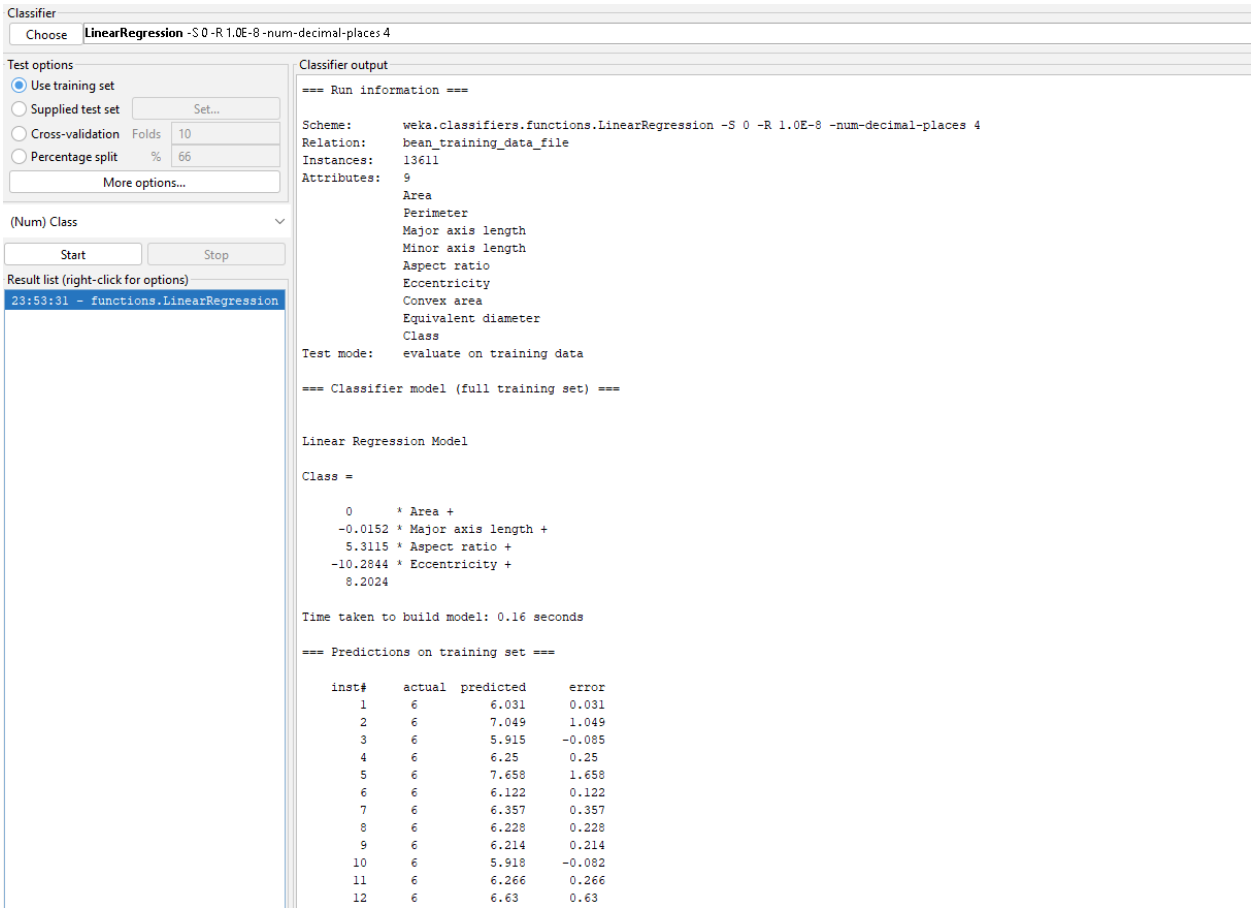
```
Area,Perimeter,'Major axis length','Minor axis length','Aspect ratio',Eccentricity,'Convex area','Equivalent diameter',Class
20420,525,184,142,1.29,0.63,20684,161,DERMASON
20464,528,191,136,1.4,0.7,20772,161,DERMASON
20548,525,184,143,1.29,0.63,20825,162,DERMASON
20711,525,186,142,1.31,0.65,20988,162,DERMASON
20786,535,201,132,1.52,0.75,21057,163,DERMASON
20942,531,191,140,1.37,0.68,21191,163,DERMASON
21101,534,185,146,1.27,0.62,21462,164,DERMASON
21314,537,194,140,1.38,0.69,21587,165,DERMASON
21348,531,192,142,1.35,0.67,21590,165,DERMASON
21397,535,193,142,1.36,0.68,21731,165,DERMASON
21405,535,191,143,1.34,0.66,21737,165,DERMASON
21479,544,199,138,1.44,0.72,21762,165,DERMASON
21558,539,197,139,1.41,0.71,21808,166,DERMASON
21570,538,197,140,1.41,0.7,21820,166,DERMASON
```

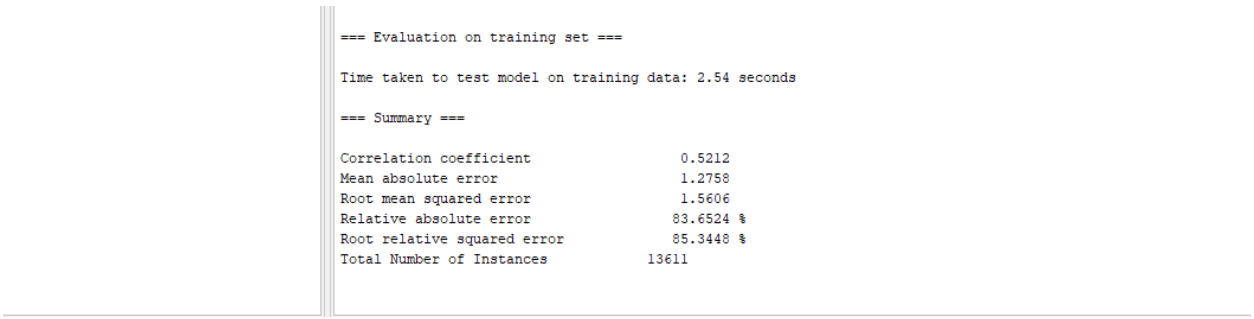
Step 3. Using Weka, with Output set to Plain text (under more options), runbean_training_data.csv file using (i) Linear Regression (ii) MultilayerPerceptron classifiers(available under function under classify). Show the predicted bean class for each of the 9 records in the test file (bean_test_file.txt or related bean_test_file.csv) along with the actual class shown

in the test file. You need to convert .txt to the corresponding formats.

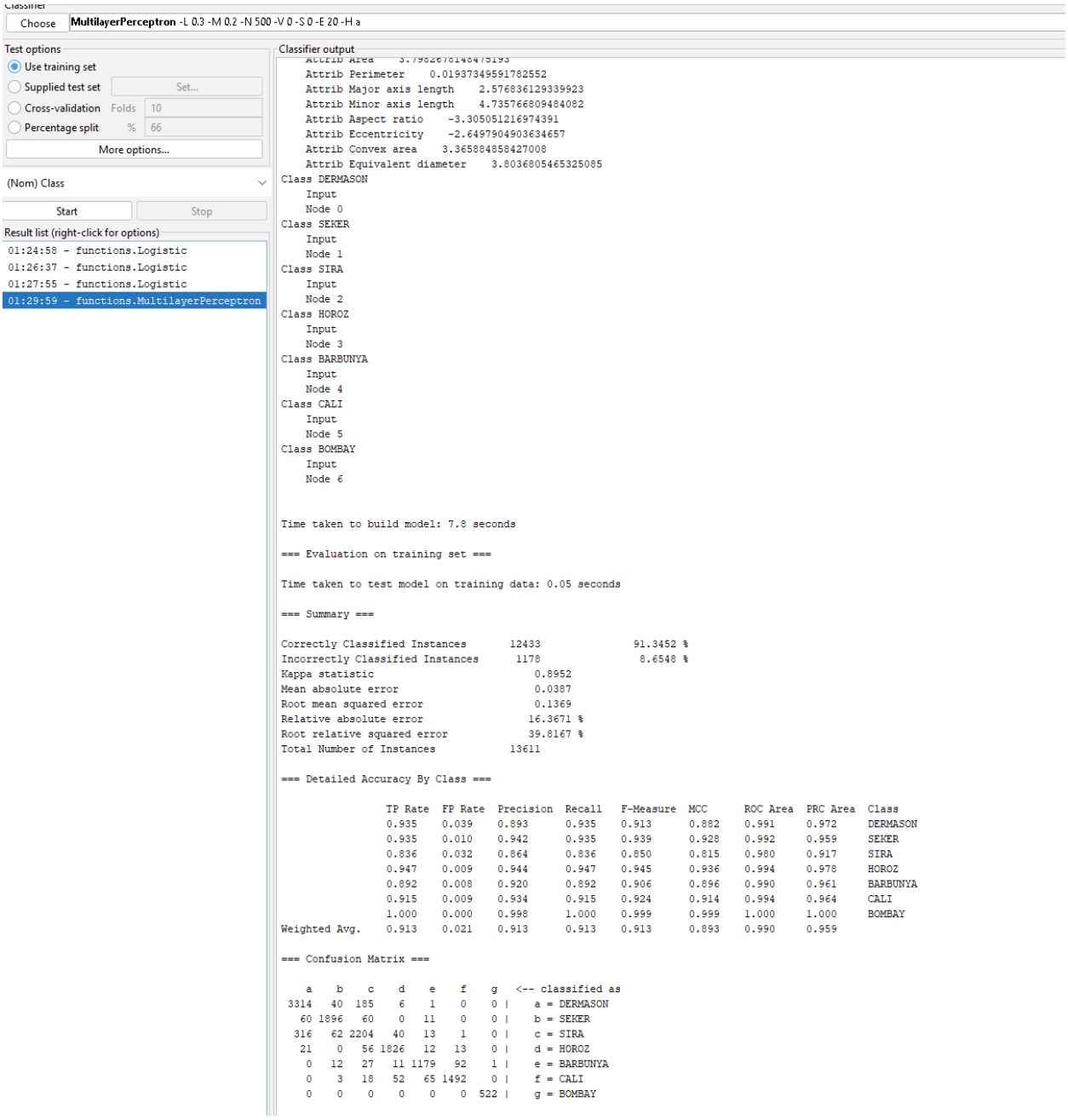
Solution

(a).Linear Regression





(b). MultilayerPerceptron



Convert bean_test_file.txt to bean_test_file.csv either using Excel or R

We need to save model for making prediction in Weka, save log.model and mlp.model for LogisticRegression and MLP.

(a)Prediction with Linear Regression

```
Linear Regression Model

Class =

     -0       * Convex area +
      6.4401

Time taken to build model: 0 seconds

=== Predictions on test set ===

    inst#     actual  predicted      error
        1          6          5.28      -0.72
        2          4          5.448      1.448
        3          6          5.251     -0.749
        4          5          4.308     -0.692
        5          1          3.945      2.945
        6          3          3.329      0.329
        7          2          1.476     -0.524
        8          7          4.922     -2.078
        9          1          3.935      2.935
       10          2          1.445     -0.555
       11          4          5.448      1.448
       12          7          4.928     -2.072
       13          3          3.338      0.338
       14          5          4.273     -0.727
       15          5          4.303     -0.697
       16          7          4.922     -2.078
       17          4          5.449      1.449

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds
```

(b)MLP Prediction

```
Time taken to build model: 0.02 seconds

=== Predictions on test set ===

    inst#     actual  predicted error prediction
        1    1:SEKER    1:SEKER       0.863
        2 2:DERMASON 2:DERMASON       0.848
        3    1:SEKER    1:SEKER       0.9
        4    3:HOROZ    3:HOROZ       0.903
        5 4:BARBUNYA 4:BARBUNYA       0.804
        6     5:CALI     5:CALI       0.846
        7   6:BOMBAY   6:BOMBAY       0.889
        8     7:SIRA     7:SIRA       0.854
        9 4:BARBUNYA 4:BARBUNYA       0.804
       10   6:BOMBAY   6:BOMBAY       0.882
       11 2:DERMASON 2:DERMASON       0.865
       12     7:SIRA     7:SIRA       0.86
       13     5:CALI     5:CALI       0.857
       14    3:HOROZ    3:HOROZ       0.894
       15    3:HOROZ    3:HOROZ       0.884
       16     7:SIRA     7:SIRA       0.861
       17 2:DERMASON 2:DERMASON       0.866

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds
```

Step 4. Repeat step 3 with bean_training_data.arff file. You should get exactly the same results as in Step 3

(a)Prediction with Linear Regression

```
Linear Regression Model

Class =

     -0       * Convex area +
      6.4401

Time taken to build model: 0 seconds

=== Predictions on test set ===

    inst#     actual  predicted      error
        1          6          5.28      -0.72
        2          4          5.448      1.448
        3          6          5.251     -0.749
        4          5          4.308     -0.692
        5          1          3.945      2.945
        6          3          3.329      0.329
        7          2          1.476     -0.524
        8          7          4.922     -2.078
        9          1          3.935      2.935
       10          2          1.445     -0.555
       11          4          5.448      1.448
       12          7          4.928     -2.072
       13          3          3.338      0.338
       14          5          4.273     -0.727
       15          5          4.303     -0.697
       16          7          4.922     -2.078
       17          4          5.449      1.449

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds
```

(b)MLP Prediction

```
Time taken to build model: 0.02 seconds

=== Predictions on test set ===

    inst#     actual  predicted error prediction
        1    1:SEKER    1:SEKER       0.863
        2 2:DERMASON 2:DERMASON       0.848
        3    1:SEKER    1:SEKER       0.9
        4    3:HOROZ    3:HOROZ       0.903
        5 4:BARBUNYA 4:BARBUNYA       0.804
        6     5:CALI     5:CALI       0.846
        7   6:BOMBAY   6:BOMBAY       0.889
        8     7:SIRA     7:SIRA       0.854
        9 4:BARBUNYA 4:BARBUNYA       0.804
       10   6:BOMBAY   6:BOMBAY       0.882
       11 2:DERMASON 2:DERMASON       0.865
       12     7:SIRA     7:SIRA       0.86
       13     5:CALI     5:CALI       0.857
       14    3:HOROZ    3:HOROZ       0.894
       15    3:HOROZ    3:HOROZ       0.884
       16     7:SIRA     7:SIRA       0.861
       17 2:DERMASON 2:DERMASON       0.866

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds
```

Step 5. Consider the raisin data (https://archive.ics.uci.edu/ml/datasets/Raisin+Dataset) predicting the type of raisins. You are provided the training set (Raisin_data_set) and the test set (raisin_test_data) for this data. By applying “Linear Regression” and “Gaussian Processes” options, predict the output for the test cases. Since output class is categorical, for linear regression, convert the output class to a numeric by mapping each class to an integer 1-7.Similarly, map the regression output back to an actual class based on nearest integer value.

Solution: Since the training and test datasets are in .xls format we need to convert them first to .csv

Csv Training data

File Edit View

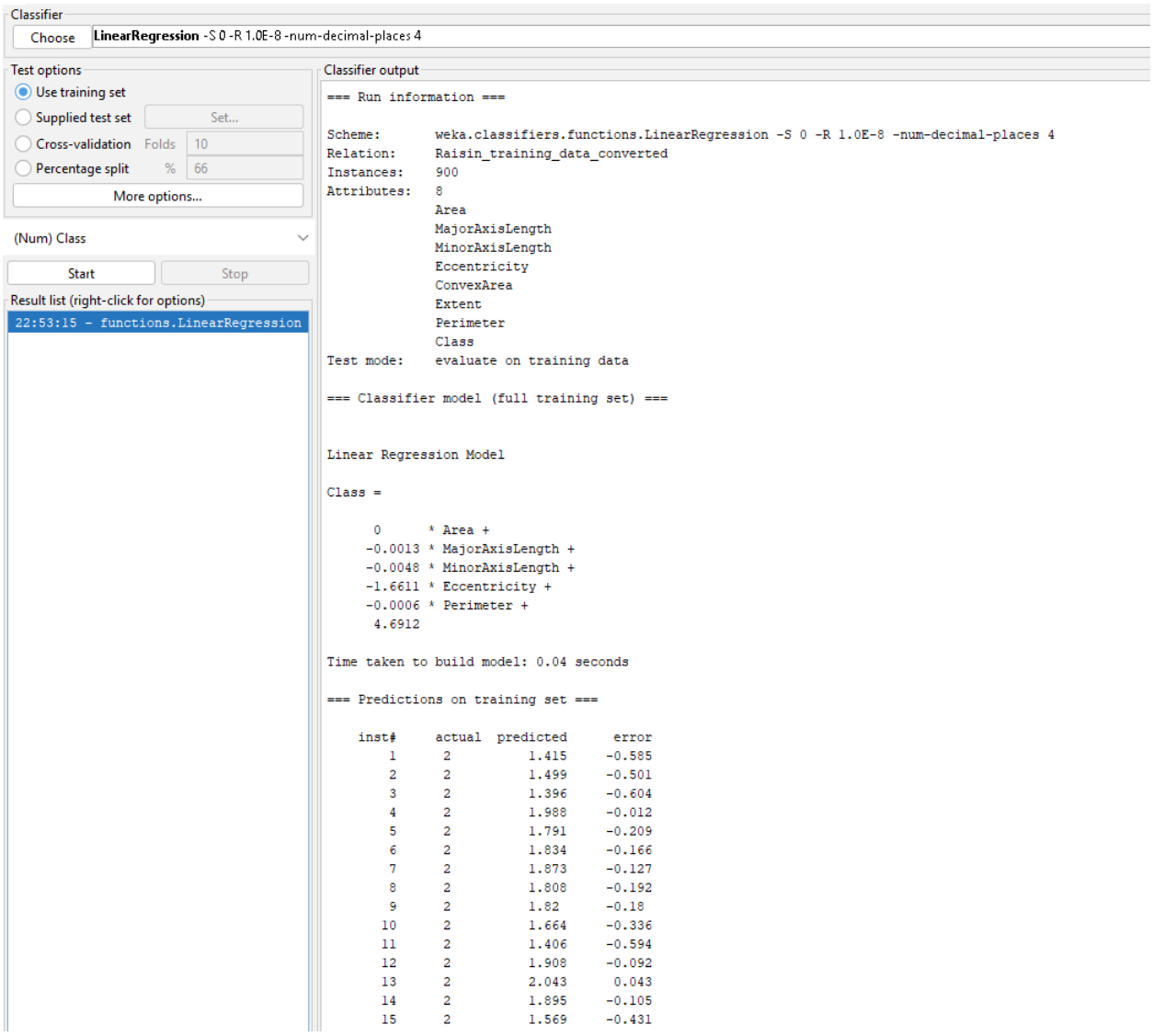
```
Area,MajorAxisLength,MinorAxisLength,Eccentricity,ConvexArea,Extent,Perimeter,Class
87524,442.2460114,253.291155,0.819738392,90546,0.758650579,1184.04,2
75166,406.690687,243.0324363,0.801805234,78789,0.68412957,1121.786,2
90856,442.2670483,266.3283177,0.798353619,93717,0.637612812,1208.575,2
45928,286.5405586,208.7600423,0.684989217,47336,0.699599385,844.162,2
79408,352.1907699,290.8275329,0.56401133,81463,0.792771926,1073.251,2
49242,318.125407,200.12212,0.777351277,51368,0.658456354,881.836,2
42492,310.1460715,176.1314494,0.823098681,43904,0.665893562,823.796,2
60952,332.4554716,235.429835,0.706057518,62329,0.74359819,933.366,2
42256,323.1896072,172.5759261,0.845498789,44743,0.698030924,849.728,2
64380,366.9648423,227.7716147,0.784055626,66125,0.664375716,981.544,2
80437,449.4545811,232.3255064,0.856042518,84460,0.674235757,1176.305,2
43725,301.3222176,186.9506295,0.784258452,45021,0.697068248,818.873,2
43441,276.6108288,201.8131355,0.683882337,45133,0.690855598,803.748,2
76792,338.8575454,291.3592017,0.510583813,78842,0.772322237,1042.77,2
74167,387.7989307,247.8581228,0.769089738,76807,0.680181585,1084.729,2
33565,261.5543311,167.7084908,0.767374275,35794,0.68155052,751.413,2
64670,403.0839752,206.4846437,0.858829168,66419,0.75677257,1028.445,2
64762,354.2939396,235.7524629,0.746473726,66713,0.694998015,981.509,2
43295,304.2844667,182.8110368,0.799406959,44714,0.713838189,814.68,2
70699,418.6985723,216.5960537,0.855799392,72363,0.728075054,1061.321,2
69726,354.1769124,252.529208,0.701160962,71849,0.734398534,1035.501,2
```

Csv Test Data

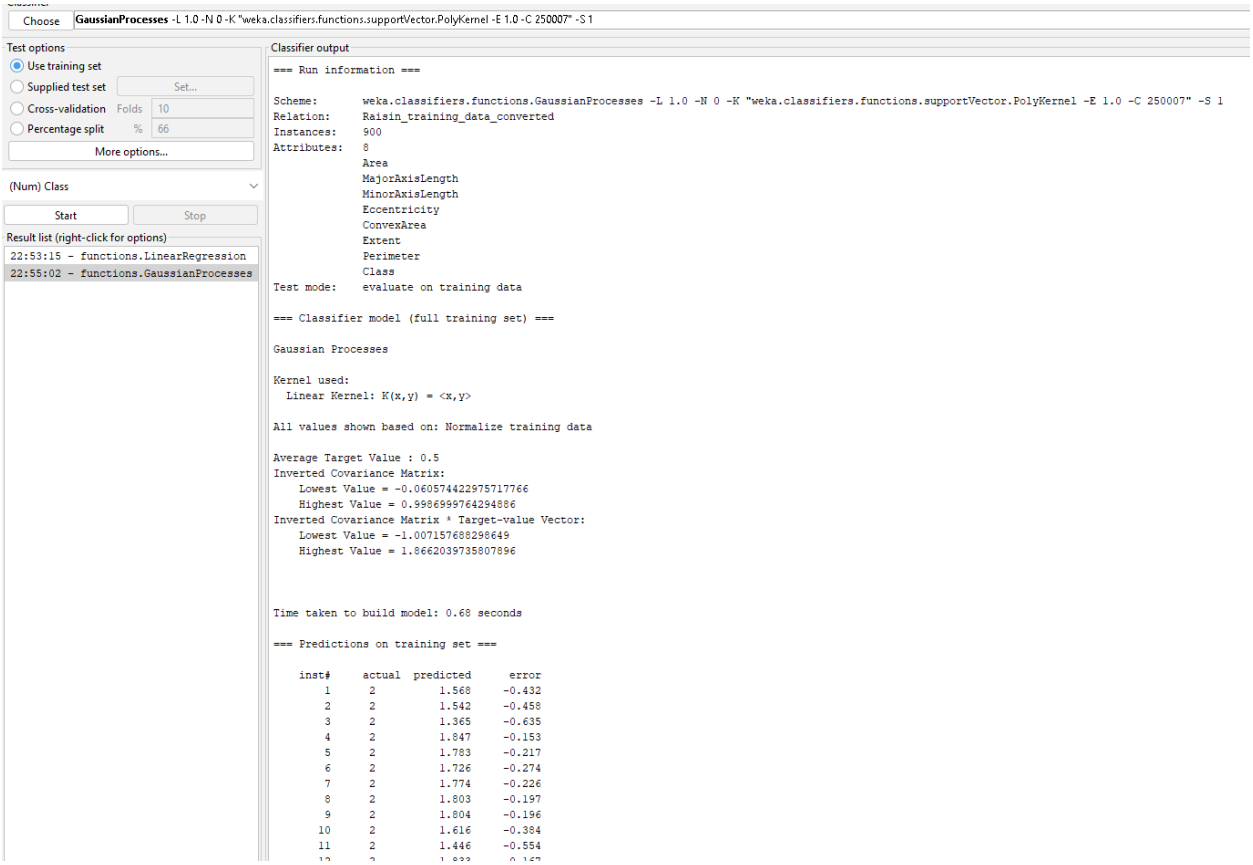
File Edit View

```
Area,MajorAxisLength,MinorAxisLength,Eccentricity,ConvexArea,Extent,Perimeter,Class
94211,450,269,0.80,96340,0.72,1195,Besni
87524,442,253,0.82,90546,0.76,1184,Kecimen
75166,407,243,0.80,78789,0.68,1122,Kecimen
86852,456,249,0.84,90550,0.61,1208,Besni
91464,433,273,0.78,93852,0.72,1182,Besni
93441,397,301,0.65,95370,0.72,1158,Besni
90856,442,266,0.80,93717,0.64,1209,Kecimen
45928,287,209,0.68,47336,0.70,844,Kecimen
```

Linear Regression on Train Data



Gaussian Processes on Train Data



Prediction with Linear Regression

inst#	actual	predicted	round-off	r actual_class	predicted_class
1	1	1.366	1	Besni	Besni
2	2	1.559	2	Kecimen	Kecimen
3	2	1.68	2	Kecimen	Kecimen
4	1	1.608	2	Besni	Kecimen
5	1	1.317	1	Besni	Besni
6	1	0.977	1	Besni	Besni
7	1	1.402	1	Besni	Besni

8	2	2.092	2	Kecimen	Kecimen

Prediction with GaussianProcess

inst#	actual	predicted	roundoff	actual_class	predicted_class
1	1	1.437	1	Besni	Besni
2	2	1.582	2	Kecimen	Kecimen
3	2	1.54	1	Kecimen	Besni
4	1	1.418	1	Besni	Besni
5	1	1.42	1	Besni	Besni
6	1	1.191	1	Besni	Besni
7	2	1.355	1	Kecimen	Besni
8	2	1.658	2	Kecimen	Kecimen