In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import datasets, preprocessing
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn import svm
from sklearn import preprocessing, model_selection, neighbors, discriminant_analysis
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
from sklearn import datasets, linear_model
from sklearn.model_selection import train_test_split
from matplotlib import pyplot as plt
from sklearn.model_selection import KFold
```

In [2]:
```
file = ("glass.data")
df = pd.read_csv(file,delim_whitespace=False, header=None)
columns = ["Id", "Ri", "Na", "Mg", "Al", "Si", "K", "Ca", "Ba", "Fe", "Type"]
df.columns = columns
df
```

Out[2]:

	Id	Ri	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
0	1	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.00	0.0	1
1	2	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.00	0.0	1
2	3	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.00	0.0	1
3	4	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.00	0.0	1
4	5	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.00	0.0	1
...	...	...	...	...	...	...	...	...	...	...	...
209	210	1.51623	14.14	0.00	2.88	72.61	0.08	9.18	1.06	0.0	7
210	211	1.51685	14.92	0.00	1.99	73.06	0.00	8.40	1.59	0.0	7
211	212	1.52065	14.36	0.00	2.02	73.42	0.00	8.44	1.64	0.0	7
212	213	1.51651	14.38	0.00	1.94	73.61	0.00	8.48	1.57	0.0	7
213	214	1.51711	14.23	0.00	2.08	73.36	0.00	8.62	1.67	0.0	7

214 rows × 11 columns

In [3]:
```
x = df.iloc[:,1:10]
y = df.iloc[:,-1:]
x , y
```

Out[3]:
```
(          Ri     Na     Mg     Al     Si     K     Ca     Ba   Fe
 0    1.52101  13.64  4.49  1.10  71.78  0.06  8.75  0.00  0.0
 1    1.51761  13.89  3.60  1.36  72.73  0.48  7.83  0.00  0.0
 2    1.51618  13.53  3.55  1.54  72.99  0.39  7.78  0.00  0.0
 3    1.51766  13.21  3.69  1.29  72.61  0.57  8.22  0.00  0.0
 4    1.51742  13.27  3.62  1.24  73.08  0.55  8.07  0.00  0.0
 ..       ...    ...   ...   ...    ...   ...   ...   ...  ...
 209  1.51623  14.14  0.00  2.88  72.61  0.08  9.18  1.06  0.0
 210  1.51685  14.92  0.00  1.99  73.06  0.00  8.40  1.59  0.0
 211  1.52065  14.36  0.00  2.02  73.42  0.00  8.44  1.64  0.0
 212  1.51651  14.38  0.00  1.94  73.61  0.00  8.48  1.57  0.0
 213  1.51711  14.23  0.00  2.08  73.36  0.00  8.62  1.67  0.0
 [214 rows x 9 columns],
      Type
 0       1
 1       1
 2       1
 3       1
 4       1
 ..    ...
 209     7
 210     7
 211     7
 212     7
 213     7

 [214 rows x 1 columns])
```

In [4]:
```
x_train, x_test, y_train, y_test = model_selection.train_test_split(x, y, test_size=0.6,random_state=1)
```

In [5]:
```
x_train,y_train
```

Out[5]:
```
(          Ri     Na     Mg     Al     Si     K     Ca     Ba    Fe
 180  1.51299  14.40  1.74  1.54  74.55  0.00   7.59  0.00  0.00
 167  1.51969  12.64  0.00  1.65  73.75  0.38  11.53  0.00  0.00
 77   1.51627  13.00  3.58  1.54  72.83  0.61   8.04  0.00  0.00
 113  1.51892  13.46  3.83  1.26  72.55  0.57   8.21  0.00  0.14
 65   1.52099  13.69  3.59  1.12  71.96  0.09   9.40  0.00  0.00
 ..       ...    ...   ...   ...    ...   ...    ...   ...   ...
 203  1.51658  14.80  0.00  1.99  73.11  0.00   8.28  1.71  0.00
 137  1.51711  12.89  3.62  1.57  72.96  0.61   8.11  0.00  0.00
 72   1.51593  13.09  3.59  1.52  73.10  0.67   7.83  0.00  0.00
 140  1.51690  13.33  3.54  1.61  72.54  0.68   8.11  0.00  0.00
 37   1.51797  12.74  3.48  1.35  72.96  0.64   8.68  0.00  0.00
 [85 rows x 9 columns],
      Type
 180     6
 167     5
 77      2
 113     2
 65      1
 ..    ...
 203     7
 137     2
 72      2
 140     2
 37      1

 [85 rows x 1 columns])
```

In [6]:
```
x_train.shape, y_train.shape
```

Out[6]:
```
((85, 9), (85, 1))
```

In [7]:
```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.linear_model import LogisticRegression
```

In [8]:
```
x_train,y_train
```

Out[8]:
```
(          Ri     Na    Mg    Al     Si     K     Ca    Ba    Fe
 180  1.51299  14.40  1.74  1.54  74.55  0.00   7.59  0.00  0.00
 167  1.51969  12.64  0.00  1.65  73.75  0.38  11.53  0.00  0.00
 77   1.51627  13.00  3.58  1.54  72.83  0.61   8.04  0.00  0.00
 113  1.51892  13.46  3.83  1.26  72.55  0.57   8.21  0.00  0.14
 65   1.52099  13.69  3.59  1.12  71.96  0.09   9.40  0.00  0.00
 ..       ...    ...   ...   ...    ...   ...    ...   ...   ...
 203  1.51658  14.80  0.00  1.99  73.11  0.00   8.28  1.71  0.00
 137  1.51711  12.89  3.62  1.57  72.96  0.61   8.11  0.00  0.00
 72   1.51593  13.09  3.59  1.52  73.10  0.67   7.83  0.00  0.00
 140  1.51690  13.33  3.54  1.61  72.54  0.68   8.11  0.00  0.00
 37   1.51797  12.74  3.48  1.35  72.96  0.64   8.68  0.00  0.00

 [85 rows x 9 columns],
      Type
 180     6
 167     5
 77      2
 113     2
 65      1
 ..    ...
 203     7
 137     2
 72      2
 140     2
 37      1

 [85 rows x 1 columns])
```

In [9]:
```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
```

LINEAR

In [35]:
```
clf = LinearDiscriminantAnalysis()
```

In [36]:
```
scores = cross_val_score(clf,x_train,y_train,cv=10, scoring='accuracy')
```

```
was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [37]:
```
scores
```

Out[37]:
```
array([0.55555556, 0.55555556, 0.55555556, 0.44444444, 0.77777778,
       0.25      , 0.75      , 0.5       , 0.625     , 0.75      ])
```

In [38]:
```
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LinearRegression
from numpy import mean
from numpy import absolute
from numpy import sqrt
import pandas as pd
```

In [39]:
```
sqrt(mean(absolute(scores)))
```

Out[39]:
```
0.759202798262025
```

QUADRATIC

In [40]:
```
clf = QuadraticDiscriminantAnalysis()
```

In [41]:
```
scores = cross_val_score(clf,x_train,y_train,cv=10,scoring='accuracy')
```

```
was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/discriminant_analysis.py:878: UserWarning: Variables are collinear
  warnings.warn("Variables are collinear")
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/discriminant_analysis.py:878: UserWarning: Variables are collinear
  warnings.warn("Variables are collinear")
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/discriminant_analysis.py:878: UserWarning: Variables are collinear
  warnings.warn("Variables are collinear")
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/utils/validation.py:993: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/Users/georgepsaltakis/opt/anaconda3/lib/python3.9/site-packages/sklearn/discriminant_analysis.py:878: UserWarning: Variables are collinear
  warnings.warn("Variables are collinear")
```

In [42]:
```
scores
```

...

In [43]:
```
sqrt(mean(absolute(scores)))
```

Out[43]:
```
0.6656241849238619
```

In [ ]: