Question One:

Basic Statistical Results:

```
Sample Size: 506

Basic Statistical Data for Column CRIME
Mean         3.61352355731
Median       0.25651
Stand. Dev.  8.5930413513
Outliers     0
Quartiles    [ 0.081984   0.25651    3.6789645]

Basic Statistical Data for Column ZN
Mean         11.3636363636
Median       0.0
Stand. Dev.  23.2993956948
Outliers     0
Quartiles    [  0.    0.   12.5]

Basic Statistical Data for Column INDUS
Mean         11.1367786561
Median       9.69
Stand. Dev.  6.85357058339
Outliers     0
Quartiles    [  5.187   9.69   18.1  ]

Basic Statistical Data for Column CHAS
Mean         0.0691699604743
Median       0.0
Stand. Dev.  0.25374293496
Outliers     35
Quartiles    [ 0.  0.  0.]

Basic Statistical Data for Column NOX
Mean         0.554695059289
Median       0.538
Stand. Dev.  0.115763115407
Outliers     388
Quartiles    [ 0.449  0.538  0.624]

Basic Statistical Data for Column RM
Mean         6.28463438735
Median       6.2085
Stand. Dev.  0.701922514335
Outliers     30
Quartiles    [ 5.88495  6.2085   6.6252 ]
```

```
Basic Statistical Data for Column AGE
Mean         68.5749011858
Median       77.5
Stand. Dev.  28.1210325702
Outliers     0
Quartiles    [ 44.97  77.5   94.1 ]

Basic Statistical Data for Column DIS
Mean         3.79504268775
Median       3.20745
Stand. Dev.  2.10362835634
Outliers     0
Quartiles    [ 2.09941   3.20745   5.212035]

Basic Statistical Data for Column RAD
Mean         9.54940711462
Median       5.0
Stand. Dev.  8.69865111779
Outliers     0
Quartiles    [  4.   5.  24.]

Basic Statistical Data for Column TAX
Mean         408.23715415
Median       330.0
Stand. Dev.  168.370495039
Outliers     0
Quartiles    [ 279.  330.  666.]

Basic Statistical Data for Column PTRATIO
Mean         18.4555335968
Median       19.05
Stand. Dev.  2.16280519148
Outliers     0
Quartiles    [ 17.395  19.05   20.2  ]

Basic Statistical Data for Column B
Mean         356.674031621
Median       391.44
Stand. Dev.  91.2046074522
Outliers     0
Quartiles    [ 375.324   391.44    396.2305]

Basic Statistical Data for Column LSTAT
Mean         12.6530632411
Median       11.36
Stand. Dev.  7.13400163665
Outliers     0
Quartiles    [  6.9295  11.36    16.9665]
```

```
Basic Statistical Data for Column MEDV
Mean         22.5328063241
Median       21.2
Stand. Dev.  9.18801154528
Outliers     0
Quartiles    [ 16.99  21.2   25.  ]
```

Basic Stats Discussion:

There seems to be a considerable amount of outliers for Nitrates of Oxides, this could be due to some suburbs in the test being very close to motorways or traffic lights, anywhere where there are petrol based vehicles, or it could be due to lack of quality measurement devices, as NOX is a rather hard gas to reliably pickup, even with expensive devices (At least from my experience as an mechanic it was).

Both RM and CHAS have outliers but I think that it is safe to say they fit into the norm.
Note: the outliers were found outside 3 standard variations from the mean.

I could not include a crossover matrix in this document so it can be found in the prep folder under HouseData.png. The figure shows predictors 1-14 from top to bottom on the y axis and 1-14 from right to left along the x axis. There are a few points easily picked up on from this figure:

- There seems to be a strong positive correlation between Med-Price and Avg-Rooms
- Age vs Med-Price seems to have a distinct looking plot
- Crime vs Med-Price seems to have exponential decay (Which one would assume)

Linear Regression:

Note:
The data seems sorted so I attempted randomisation of the data on the fly, this got me into a bit of trouble with calculating the number of principal components, and leads to varied results each run. However on average (Repeated numerous times to get average stepwise and pcr models) the following models worked well:

Reverse Stepwise Regression: 0, 3, 5, 10, 11, 12
Forward Stepwise Regression: 0, 3, 5, 7, 10, 11, 12
Principal Component Number = 10
Reduced Model = 6, 8, 11, 13 (Did not work well)
Lambda for Ridge Regression = 32 (9 D.F)

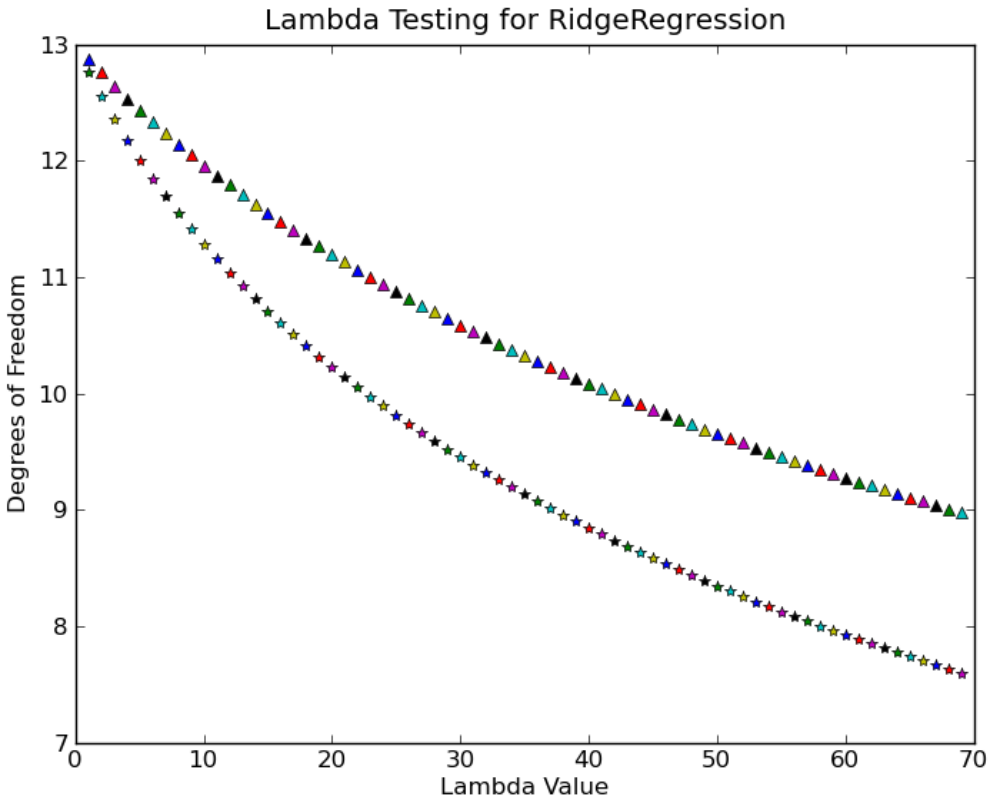

Figure 1.1 Showing Ridge Regression Analysis

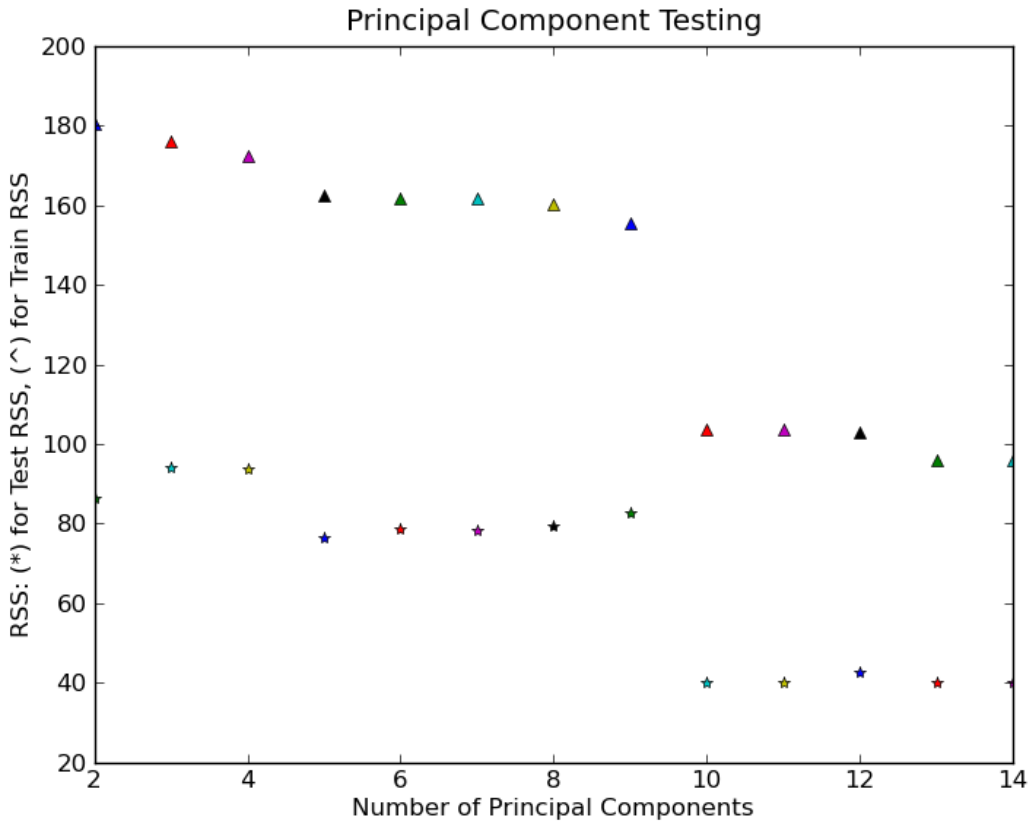

Figure 1.2 Principal Component Analysis

```
sigmahat2 =  0.297805588658
beta, z =  [[  4.5990604    6.53243156]
 [ -0.11671902  -3.3051134 ]
 [  0.13782576   2.96024983]
 [  0.0176207    0.28736047]
 [  0.07244691   2.53797768]
 [ -0.23044123  -3.55321756]
 [  0.24745733   6.53756738]
 [  0.05591792   0.98923581]
 [ -0.32655068  -5.16773702]
 [  0.33573282   4.01257292]
 [ -0.23821489  -2.55882553]
 [ -0.24603938  -6.06966442]
 [  0.08262169   2.35272271]
 [ -0.49701172 -10.01728311]]

BIC (full | pcr | backward | forward | ridge) = [[ 414.31399486]] [[ 417.62919098]]
[[ 441.74823894]] [[ 435.22105862]] [[ 405.09026067]]

Full Model Posterior probability                0 %
PCR Model Posterior probability                 0 %
```

```
Backwards Regression Model Posterior probability  0 %
Forwards Regression Model Posterior probability   0 %
Ridge Regression Model Posterior probability      98 %

(error | optimism | total) (full)       = [[ 0.28528523]] 0.0239879475683 [[
0.30927318]]
(error | optimism | total) (reduced)    = [[ 0.30802903]] 0.0239879475683 [[
0.33201698]]
(error | optimism | total) (backward)   = [[ 0.34859575]] 0.0102805489578 [[
0.3588763]]
(error | optimism | total) (brute)      = [[ 0.33802793]] 0.0119939737841 [[
0.3500219]]
(error | optimism | total) (ridge)      = [[ 0.2958387]] 0.0239879475683 [[
0.31982664]]

Full model     : RSS(train), RSS(test) = [[ 94.99998278]] [[ 38.14007939]]
Reduced model  : RSS(train), RSS(test) = [[ 261.97617256]] [[ 119.45387374]]
PCR model      : RSS(train), RSS(test) = [[ 102.57366843]] [[ 45.58937751]]
Ridge model    : RSS(train), RSS(test) = [[ 98.51428546]] [[ 36.09015199]]
Backward model : RSS(train), RSS(test) = [[ 116.08238581]] [[ 39.98709338]]
Forward model  : RSS(train), RSS(test) = [[ 112.56330036]] [[ 39.3866601]]
Bagged model   : RSS(train), RSS(test) = [[ 95.12222678]] [[ 38.07630031]]
Average Model  : RSS(train), RSS(test) = [[ 98.43289341]] [[ 36.09960615]]
```

Discussion:
From the above model, we see that most Z values for the full model are quite high, and due to the randomisation of the data every run, I found it impossible to find a suitable reduced model. I did not include the reduced model in the BIC as it would weigh it all down with inaccurate readings.

The ridge regression model seems to pick a good beta every run, which leads me to believe that for this data ridge regression would be the best selection. Model averaging selects Ridge 99% of the time, and sometimes adds a little from PCR and Backward depending on how lucky we get.

So the conclusion I have drawn from this is that either I have stuffed something up in my workings, or Ridge Regression selects the best weighting of the betahat. The bagged model does fairly well, So it would be a toss up between selection of those two.

Perceptron Analysis:
Before the data could be used with the perceptron, the response had to be categorised, this was done in increments of 5, which made 10 classes.

After this the Multi Layer Perceptron was straight forward enough to setup with initial normalised predictors.

Results of All 13 predictors:

```
[[ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.]
 [ 7.  1.  5.  2.  0.  1.  0.  0.  0.  0.]
 [ 8.  0.  1. 26.  6.  0.  0.  0.  0.  1.]
 [ 0.  0.  0.  7. 31.  8.  3.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  1.  2.  7.  4.  2.  3.]
 [ 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.  0.  0.  0.]]
```

Percentage Correct: 54.7619047619

Discussion:

55% does not look to good, better than no model at all, but I think its safe to say a linear regression model would suit this problem better.

Question Two:

Basic Statistical Results:

```
Sample Size: 214

Basic Statistical Data for Column ID
Mean         107.5
Median       107.5
Stand. Dev.  61.775804325
Outliers     0
Quartiles    [  53.95  107.5   161.05]

Basic Statistical Data for Column RI
Mean         1.51836542056
Median       1.51768
Stand. Dev.  0.00302975995483
Outliers     211
Quartiles    [ 1.5165195  1.51768    1.519161 ]

Basic Statistical Data for Column Na
Mean         13.4078504673
Median       13.3
Stand. Dev.  0.814693369343
Outliers     6
Quartiles    [ 12.8995  13.3     13.832 ]

Basic Statistical Data for Column Mg
Mean         2.68453271028
Median       3.48
```

Stand. Dev. 1.43903378681
Outliers   0
Quartiles   [2.0795 3.48   3.6005]

Basic Statistical Data for Column Al
Mean   1.44490654206
Median   1.36
Stand. Dev. 0.498101761789
Outliers   27
Quartiles   [1.19 1.36 1.63]

Basic Statistical Data for Column Si
Mean   72.6509345794
Median   72.79
Stand. Dev. 0.772733989255
Outliers   12
Quartiles   [72.279   72.79   73.0905]

Basic Statistical Data for Column K
Mean   0.497056074766
Median   0.555
Stand. Dev. 0.650666248509
Outliers   3
Quartiles   [0.12   0.555 0.61]

Basic Statistical Data for Column Ca
Mean   8.95696261682
Median   8.6
Stand. Dev. 1.41982446872
Outliers   1
Quartiles   [8.2395 8.6   9.1825]

Basic Statistical Data for Column Ba
Mean   0.175046728972
Median   0.0
Stand. Dev. 0.496056173017
Outliers   16
Quartiles   [0. 0. 0.]

Basic Statistical Data for Column Fe
Mean   0.0570093457944
Median   0.0
Stand. Dev. 0.0972107735392
Outliers   206
Quartiles   [0.   0.   0.1]

Basic Statistical Data for Column Type of Glass
Mean   2.78037383178

```
Median       2.0
Stand. Dev.  2.09881761339
Outliers     0
Quartiles    [ 1.  2.  3.]
```

Discussion (Basic Stats):

Outliers in the Iron category, this is most likely due to the mean being 0 so the 3* standard deviation does not work here. The rest of the data does not look to bad, small sample size has been noted also.

GlassData.png shows clustering of the data, and considering the data is categorical I will attempt to use Classification techniques to create a model for the data.

Response 4 (vehicle windows non float processed) was removed as it was not included in the database, leaving a total of 6 classes. This may cause headaches later on if a model is selected to use on data that does contain predictor, but I will ignore this in hope that I don't lose marks.

All types of Glass:

Naive Bayes Classifier / Linear Discriminant Analysis:

```
Mean Error Rate [NBC]      =  0.22222222
Mean Error Rate [LDA-train]  =  0.209589041096
Mean Error Rate [LDA-test]   =  0.158904109589
```

Multi-Layer Perceptron:

```
[[ 20.   0.   0.   0.   0.   0.]
 [  0.  19.   5.   4.   1.   4.]
 [  0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.]]
Percentage Correct:  73.5849056604
```

Self Organising Map / K-Means Algorithm:

```
K-means percentage correct = 0.415094339623
```

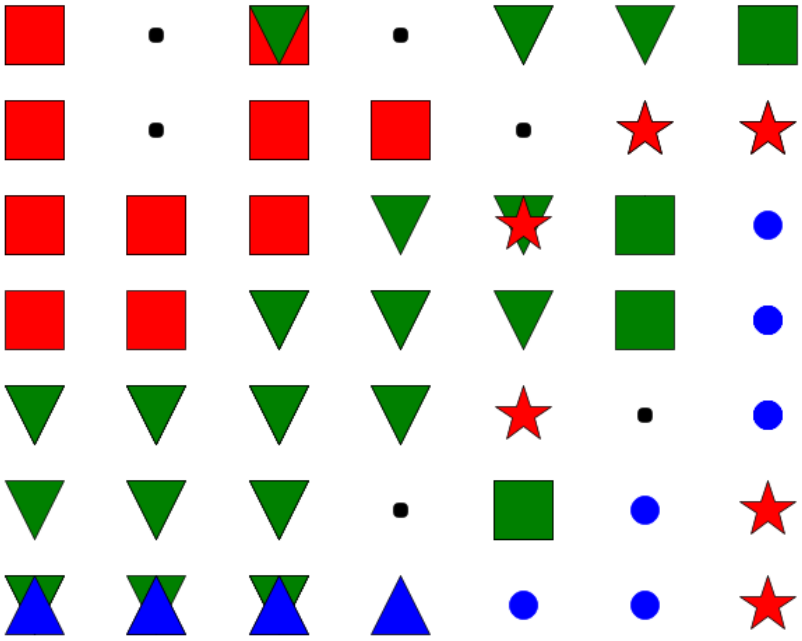

Figure 2.1: Self organising map for training data

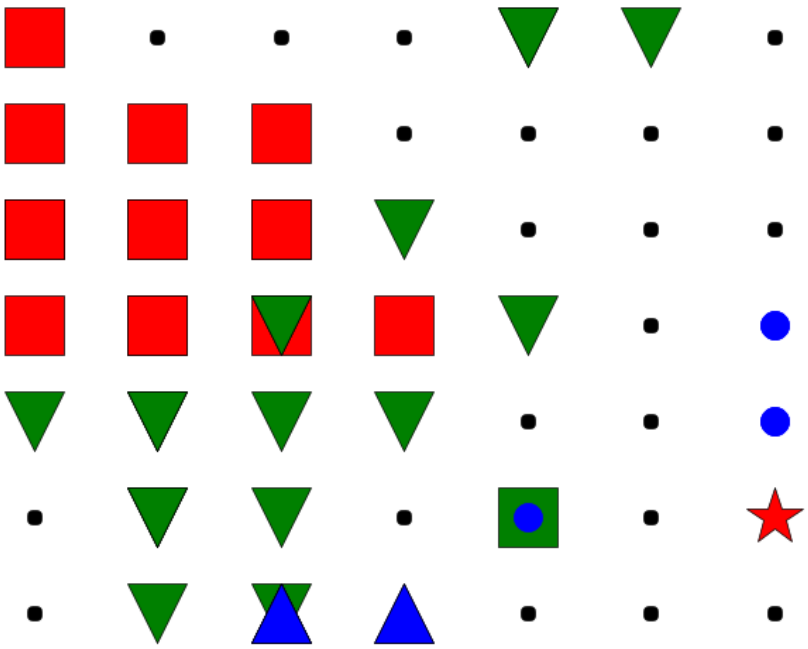

Figure 2.2: Self organising map for test data

Glass Vs. Non-Glass:

Naive Bayes:

```
Mean Error Rate [NBC] =  0.10347222
```

Multi-Layer Perceptron:

```
Confusion matrix is:
[[  0.   0.]
 [  8.  45.]]
Percentage Correct:  84.9056603774
```

Float vs. Non Float (And the rest)

Naive Bayes:

```
Mean Error Rate [NBC] =  0.12291667
```

Multi-Layer Perceptron:

```
Confusion matrix is:
[[ 53.   0.   0.]
 [  0.   0.   0.]
 [  0.   0.   0.]]
Percentage Correct:  100.0
```

Discussion:

Full Set:
The linear discriminant analysis produced the best looking results, with a test error of 15%, followed by the Naive Bayes Classifier, and last but not least the Multi-Layer Perceptron. I sampled with two unsupervised learning algorithms to assess their results of the data set, the K-means algorithm was very hit and miss (May be to do with improper implementation), however the Self Organising Map worked quite well on the set, from examination it classified quite a few points and had few over laps.

Split Set:
The two split sets definitely made better models, Float vs Non made a perfect Multi-layer perceptron at 100% correct classification on its test, and also helped the Naive Bayes achieve 12% error. Whereas the Glass vs Non made for a 10% Naive Bayes, a 5% reduction from the full set.