Statistical Machine Learning - Assignment 2

Alex Gibson - 07120141

Question 1

Use cross-validation on the prostate cancer training data to select the best model from the reduced regression model (predictors 1,2,4 and 5), ridge regression, and principal components regression. Compare your results to those obtained by fitting the models to the full training data and then evaluating them on the test data.

Results:

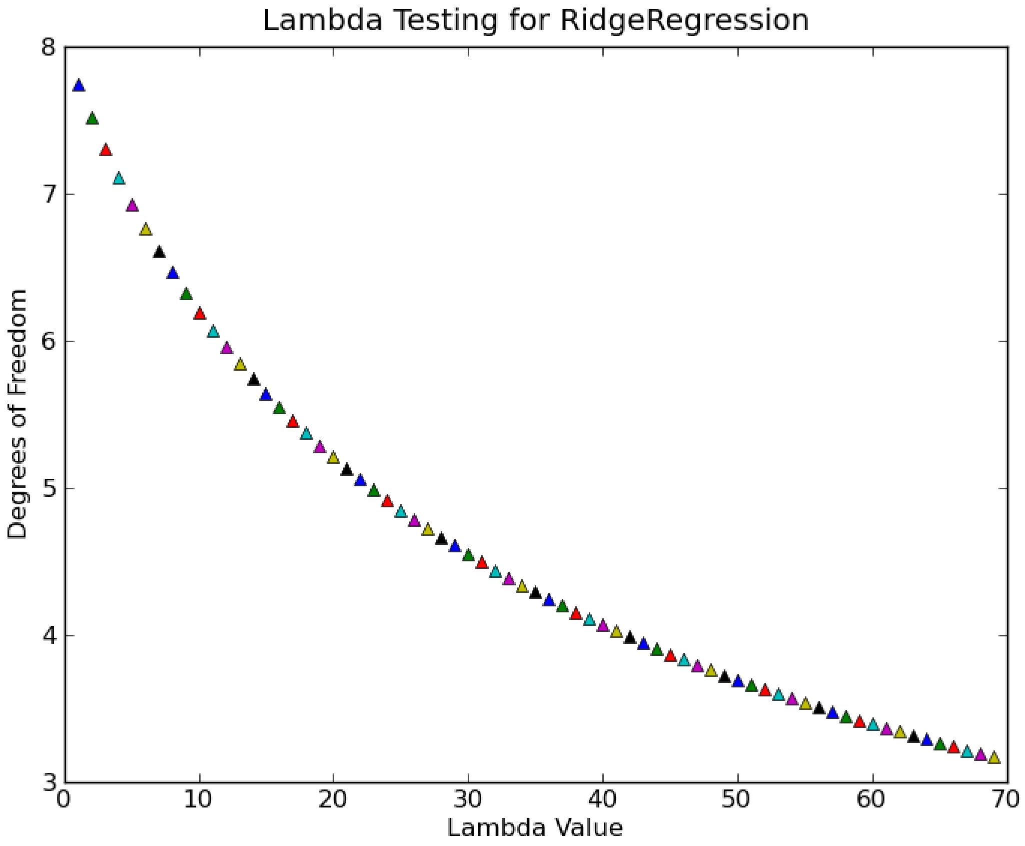

Figure 1.1: Showing D.F for different selections of Lambda in Ridge Regression.

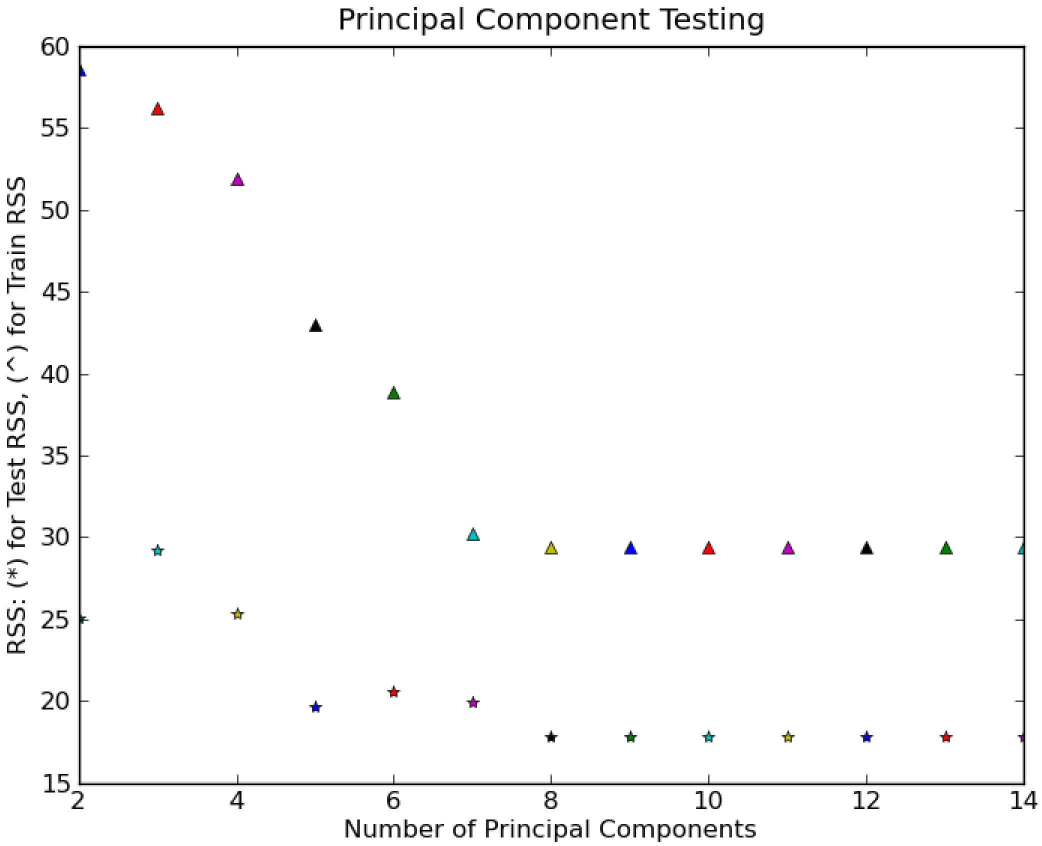

Figure 1.2: Showing RSS for test and training with different numbers of principal components.

Selected Models:

```
Full model              : RSS(train), RSS(test) = [[ 29.42638396]] [[
17.58986697]]
Reduced model     : RSS(train), RSS(test) = [[ 32.90740736]] [[ 14.9072031]]
Ridge model             : RSS(train), RSS(test) = [[ 35.07993745]] [[
16.4449807]]
PCR model               : RSS(train), RSS(test) = [[ 43.00079757]] [[
19.49481604]]

Cross-validation:

Total Loss (CV) [full | reduced | ridge | principal components] =
 [  7.49292037   8.53763376  12.40211575  13.4733518 ]
 [  5.70202694   7.08715434  14.49647256  12.16981918]
 [  6.6799404    5.5569764    7.36802944  10.25797902]
 [ 15.1673882   13.65123752  11.62022532  10.93942506]
 [  5.52807603   4.88276409   6.47209691   9.20825788]

Cross-validation mean:
```

```
Mean Loss RSS [full | reduced | ridge | principal components] =
[  8.11407039   7.94315322  10.471788    11.20976659]
```

Discussion:

Assumptions:
The data-set was assumed to have variance of around 4.2 d.f, so and attempt was made to set the lambda in the ridge regression equation to 4.2 d.f, figure 1.1 shows the lambda values and their effect on the d.f, selecting a lambda of 32 gives a close approximation to 4.2 so that amount has been used.

Figure 1.2 shows the test and training rss with different amounts of principal components, you can see from this that 8 principal components uses all 8 predictors in the data-set, selecting 5 for the principal components gives the best rss for the test set, so 5 principal components has been used in this problem.

The size of K for the cross-validation has been left at 5, this is because of the explanation on page 39, study guide 6, stating that the accuracy vs the size of k plateaus around k = 5.

Results:
The cross validation favours the reduced model, showing that the prediction error is 7.94, and the full model coming second with 8.11.

Question 2

For the prostate cancer data, consider the the full regression model, the reduced regression model (only predictors 1,2,4 and 5), and the ridge regression model (all from Lecture Slides 5), the backward regression model from Exercise 5.5, and the best model obtained in Question 4 of Assignment 1.

Part (a):

Results:

```
BIC (full | reduced | backward | brute | ridge) =
[[ 104.84223357]] [[ 91.74460222]] [[ 96.53023798]] [[ 94.69069069]] [[
98.54413106]]

Full Model Posterior probability                                       0 %
Reduced Model Posterior probability                                    73 %
Backwards Regression Model Posterior probability      6 %
Brute Force Model Posterior probability                          16 %
Ridge Regression Model Posterior probability               2 %

(err | opt | Err) (full)          = [[ 0.43919976]] 0.117993965 [[
0.5571937]]
(err | opt | Err) (reduced)       = [[ 0.49115533]] 0.052441762 [[
0.5435971]]
(err | opt | Err) (backward)      = [[ 0.43983817]] 0.091773084 [[
0.5316113]]
(err | opt | Err) (brute)         = [[ 0.53803028]] 0.039331321 [[
0.5773616]]
(err | opt | Err) (ridge)         = [[ 0.52358116]] 0.117993965 [[
0.6415751]]
```

Part (b):

Results:

```
Average Beta =  [[-0.00452591]
 [ 0.63710802]
 [ 0.22115244]
 [-0.00962624]
 [ 0.22173544]
 [ 0.23934457]
 [-0.01909365]
 [ 0.00113409]
 [ 0.02103813]]

Bagged Beta = [[ 0.61333686]
 [ 0.71221954]
 [ 0.27157489]
 [-0.14182764]
 [ 0.241543  ]
 [ 0.30483097]
 [-0.28685729]
 [-0.02775604]
 [ 0.29752732]]


Average Model    : RSS(Test) = [[ 15.42861529]]
Bagged Model     : RSS(Test) = [[ 17.94870054]]
```

Part (c):

Discussion:

Assumptions:
As with the previous question, the ridge regression model has been selected to have close to 4 degrees of freedom (Lambda of 32).

Results:
From the calculated BIC of all the models, we see that both the reduced model and the brute force model are preferred, this is likely because the BIC method penalises heavily for small data-sets, leaning towards the models with less parameters. However, from the posterior probability we can see that the reduced model is also more preferable (As well as having the best BIC), I'm not sure how many models are needed to calculate a reliable posterior probability, but I am assuming that the model set is large enough and varied to get an accurate result.

The optimism is showing that the brute force model and reduced model both have a low expect difference between the test and training errors, and the Err shows that all models except the ridge regression have quite similar estimated error.

Comparing the bagged model against the Bayesian average model, shows that the average model has a considerably lower rss on the test data, looking at the Beta-hat of both also shows that they are quite different from one another.

The average model is most persuaded by the reduced set, which has bias due to a data point anomaly in the set, it would be interesting to see the results if that were not there, considering the average model should be less sensitive to individual model assumptions.

Question 3

When you arrive at the pub, your five friends already have their drinks on the table. Jim has a job and buys the round half of the time. Jane buys the round a quarter of the time, and Sarah and Simon buy a round one eighth of the time. John hasn’t got his wallet out since you met him three years ago.

Part (a):

Compute the entropy of each of them buying the round and work out how many questions you need to ask (on average) to find out who bought the round.

Results:

```
Entropy for Jim buying a round is         1.0
the information gain for knowing this is  0.75

Entropy for Jane buying a round is        0.811278124459
the information gain for knowing this is  0.686278124459

Entropy for Sarah buying a round is             0.5435644432
the information gain for knowing this is  0.4966894432

Entropy for Simon buying a round is       0.5435644432
the information gain for knowing this is  0.4966894432

Entropy for John buying a round is        0.0
the information gain for knowing this is  0.0

Average amount of questions to ask :      1.875
But if Simon and Sarah were asked as a couple it would be : 1.75
```

Discussion:

The only curve ball that could screw up the equation would be if John did decide to buy a drink one day.

Part (b):

Two more friends now arrive and everybody spontaneously decides that it is your turn to buy a round (for all eight of you). Your friends set you the challenge of deciding who is drinking beer and who is drinking vodka according to their gender, whether or not they are students, and whether they went to the pub last night. Use ID3 to work it out, and then see if you can prune the tree.

Results:

```
  Gender
  Male
      Pub Last Night
      Yes
                  ->    Beer
      No
                  ->    Vodka
  Female
            ->    Vodka

---------------Pruned Tree------------------
  Gender
  Male
            ->    Beer
  Female
            ->    Vodka
```

Discussion:
Using your algorithm was the easy part, implementing a pruning algorithm was not so easy. My approach after a bit of research into decision tree pruning was to use pessimistic pruning, which does not require a training and test set. The idea is to make a single pass up from the leaves of the decision tree towards the root, and at every internal node make the decision whether or not to prune.

The decision is base on the comparison of the error at the node if pruned to the error of the node sub-tree, the decision is shown in formula 2.1.

$$\boxed{\frac{e2}{nv} < \frac{e1}{nv} + \sqrt{c\,\frac{k}{n}}}$$

Formula 2.1: e2 = number of errors if v is replaced by a leaf
e1 = number of errors at the sub-tree

n = size of sample set
k = number of nodes in the sub-tree
nv = number of examples that arrive at node v
c = likeliness of pruning, leverage variable

The only way I could prune the tree using the the previously mentioned formula was to set c to 1.1, anything lower would cause the pruning to favour the full set, this is because removing the ‘Pub Last Night’ classifier causes the same amount of error as the function sqrt(k/n).

Question 4

Devise a scoring scheme that evaluates how well each map size does based on these two criteria, plus anything else that you think is important. Use it to optimise the size of the network for the iris data.

Results:

```
Network Size : 2  has average penalty of : 0.215
Network Size : 3  has average penalty of : 0.253333333333
Network Size : 4  has average penalty of : 0.13849702381
Network Size : 5  has average penalty of : 0.125071428571
Network Size : 6  has average penalty of : 0.114318783069
Network Size : 7  has average penalty of : 0.091481624908
Network Size : 8  has average penalty of : 0.112809409734
Network Size : 9  has average penalty of : 0.131500229903
The Best Network Size for the data is : 7
```

Discussion:

Method:
The has two loops, one loops 8 times, creating network sizes ranging from 2 to 10, the inner loop repeats 10 times averaging the penalty results obtained from each network size.

The formula for calculating the results consists of the following:

Non-Classified Points:
Count the number of non-classified points divide by the total set size, each point is only worth .8 as this is not such a bad thing to have.

Paired Set Intersection:
A pair of sets, calculate the overlap between the two, divide by the size of the two sets combined.

Total Intersection:
Take all the sets and find the intersection, divide by the total set size, each point in the overlap is worth 2, as this is a bad situation to have.

The results for this test show a local minimum around 7.