Assignment 3

Assignment 3 - High-dimensional methods

Background

In this assignment we are going to use nearest shrunken centroid (NSC) classification method to classify cells as their proper types depending on which genes are expressed in the cells. The data is interesting as the number of features are almost seven times as many as the number of observations (300 x 2086). This is a task where common classification techniques are not appropriate, hence we need to use some regularization, here in the form of the NSC, where the classwise means are shrunk towards the overall mean for each non-significant feature (see Hastie et al. (2001). *The Elements of Statistical Learning*).

The distances from each class mean to the overall mean are standardized by subtracting the overall mean from each class mean and dividing by the pooled within-class standard deviation s_j (with a small correction to protect against extreme values using a small constant s_0).

$$d_{kj} = \frac{\bar{x}_{kj} - \bar{x}_j}{m_k(s_j + s_0)}$$

The shrinkage is done using soft thresholding, that is a shrinkage where a threshold parameter Δ is subtracted from each of the (absolute) values. This parameter can be selected using cross-validation. The resulting values that are negative are set to zero.

$$d'_{kj} = \text{sign}(d_{kj})(|d_{kj}| - \Delta)_+$$

To attain the new shrunken class centroids the following calculation is made:

$$\bar{x}'_{kj} = \bar{x}_j + m_k(s_j + s_0)d'_{kj}$$

These new values can then be used by a linear discriminant. As we use a threshold, some features may be set to zero and discarded, much like in the lasso, resulting in a simpler model.

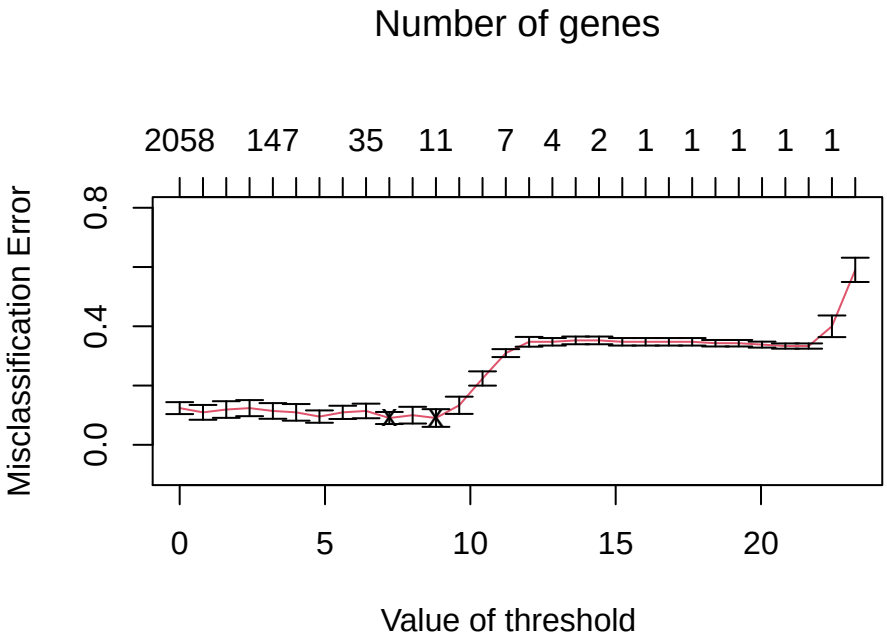
1. Nearest Shrunken Centroid Classification using Cross-Validation

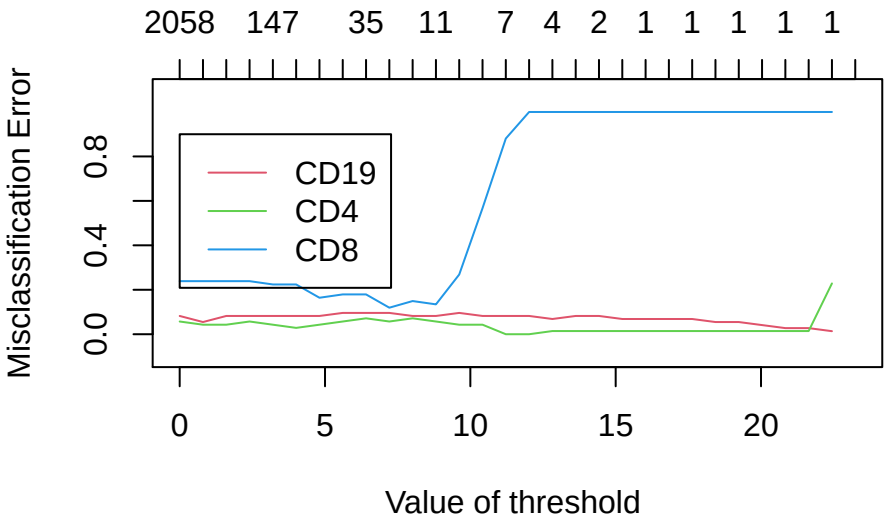
We are supposed to do NSC classification of the gene data. Data is first divided into training and test sets with proportions 70/30 without scaling and the model is then trained on the training data using package `pamr`.

Threshold and Classification Error The optimal value of the threshold is either 7.213 or 8.816 (both give the same number of errors). As less shrinkage could be argued to be less complex (the model is less sensitive), we have decided to use the former value for the threshold. In the output below one can see how the number of non-zero features are decreasing as the threshold value increases. The number of misclassifications seem to reduce a first, but then increase heavily as only a few features remain.

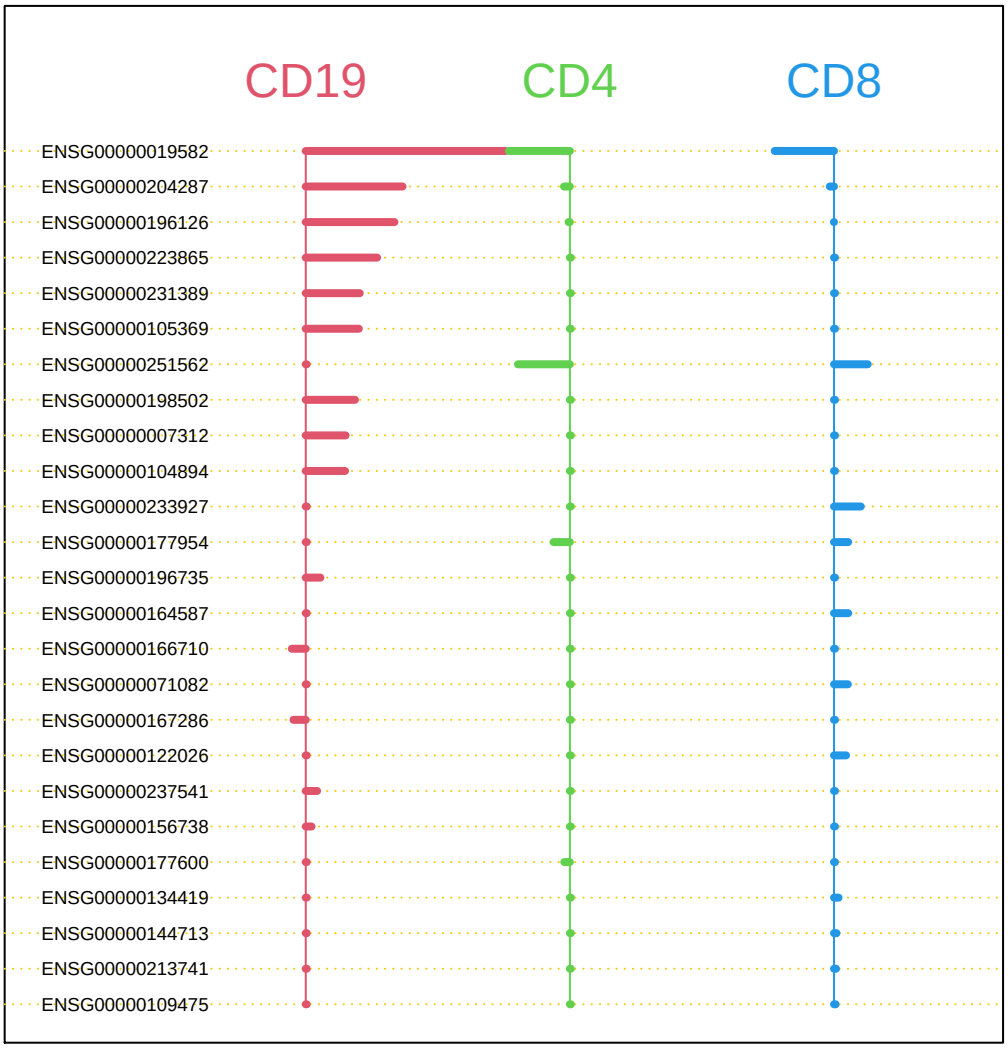
```
Call:
pamr.cv(fit = m1, data = training_data)
   threshold nonzero errors
1   0.000    2058    26
2   0.801    1040    23
3   1.603     374    25
4   2.404     226    26
5   3.206     147    24
6   4.007     102    23
7   4.809      78    20
8   5.610      54    23
9   6.412      35    24
10  7.213      25    19
11  8.014      19    21
12  8.816      11    19
13  9.617      10    28
14 10.419       8    47
15 11.220       7    65
16 12.022       4    73
17 12.823       4    73
18 13.624       3    74
19 14.426       2    74
20 15.227       1    73
21 16.029       1    73
22 16.830       1    73
23 17.632       1    73
24 18.433       1    72
25 19.235       1    72
26 20.036       1    71
27 20.837       1    70
28 21.639       1    70
29 22.440       1    84
30 23.242       0    124
```

Visual Inspection of Errors Inspecting the error rate and threshold visually reveals that it is the CD8 cell type that seems to be misclassified more often as the threshold increases.





Centroid Plot The centroid plot below shows how, for each cell type, the gene expression differs between each cell type's centroid and the overall centroid or the average expression. This means that not all cell type can have a positive (or negative) expression. The plot shows all the genes that survived the shrinkage, 25 in this case. "Surviving" in this case means that for at least one of the cell types, the shrunken centroid is non-zero. The CD19 cells are markedly different from the CD4 and CD8 cells, indicating the difference between these cells are greater.



Surviving Genes Below are the genes that survive the shrinkage, presented in a table format.

```
      id  name            CD19-score CD4-score CD8-score
 [1,] 2   ENSG00000019582 1.5153     -0.4617   -0.4489
 [2,] 15  ENSG00000204287 0.7328     -0.0448   -0.032
 [3,] 31  ENSG00000196126 0.6715     -0.0105   -0.001
 [4,] 32  ENSG00000223865 0.5403     0         0
 [5,] 37  ENSG00000231389 0.4089     0         0
 [6,] 138 ENSG00000105369 0.4019     0         0
 [7,] 1   ENSG00000251562 0          -0.3933   0.2553
 [8,] 90  ENSG00000198502 0.3727     0         0
 [9,] 172 ENSG00000007312 0.3016     0         0
[10,] 126 ENSG00000104894 0.2974     0         0
[11,] 79  ENSG00000233927 0          0         0.2028
[12,] 11  ENSG00000177954 0          -0.1239   0.1074
[13,] 110 ENSG00000196735 0.1109     0         0
[14,] 21  ENSG00000164587 0          0         0.1078
[15,] 3   ENSG00000166710 -0.1057    0         0
[16,] 50  ENSG00000071082 0          0         0.1051
[17,] 207 ENSG00000167286 -0.0933    0         0
[18,] 29  ENSG00000122026 0          0         0.0914
[19,] 192 ENSG00000237541 0.0862     0         0
[20,] 309 ENSG00000156738 0.0438     0         0
[21,] 28  ENSG00000177600 0          -0.0426   0
[22,] 38  ENSG00000134419 0          0         0.0346
[23,] 18  ENSG00000144713 0          0         0.0189
[24,] 127 ENSG00000213741 0          0         0.0156
[25,] 36  ENSG00000109475 0          0         0.0119
```

```
215313237138190172126791111021350207291923092838181273 6ENSG00000019582ENSG00000204287ENSG00000196126ENS
```

2. Most Contributing Genes

The two most contributing genes are given at the top of the previous table. These are, together with their synonyms:

- ENSG00000019582 - CD74/DHLAG
- ENSG00000204287 - HLA-DRA1

Both of these genes seem to appear in immune system cells according to https://panglaodb.se/markers.html (both B and T cells are lymphocytes).

Test Error of NSC Model The confusion matrix and calculated error for the test set are presented below.

```
NSC Confusion Matrix:
```

```
      Predicted
True   CD19 CD4 CD8
  CD19   27   0   0
  CD4     0  26   4
  CD8     0   6  27
```

```
NSC Error:

0.111

Number of non-zero parameters:

25
```

3. Elastic Net and Support Vector Machines

Elastic Net The elastic net technique combines L1 and L2 regularization, allowing for feature parameters to be reduced to zero. Here we are using elastic net with multinomial response and $\alpha = 0.5$. The penalty factor λ is selected through cross-validation. Package `glmnet` was used for model fitting with training data and prediction with test data.

```
Elastic Net Confusion Matrix:

      Predicted
True   CD19 CD4 CD8
  CD19   27   0   0
  CD4     0  29   1
  CD8     1   3  29

Elastic Net Error:

0.056

Number of non-zero parameters:

54
```

Support Vector Machine Here we are using support vector machines (SVM) to do the same classification task. SVM's can be used for both linear and non-linear classification, the latter by using the "kernel trick". In short, the SVM fits a decision boundary as a hyperplane to maximize the margin between classes. In this case we will be using the vanilladot kernel (linear) from the `ksvm` function from the package of the same name.

```
 Setting default kernel parameters

SVM Confusion Matrix:

      Predicted
True   CD19 CD4 CD8
  CD19   27   0   0
  CD4     0  30   0
  CD8     0   2  31

SVM Error:
```

```
0.022

Number of support vectors:

74
```

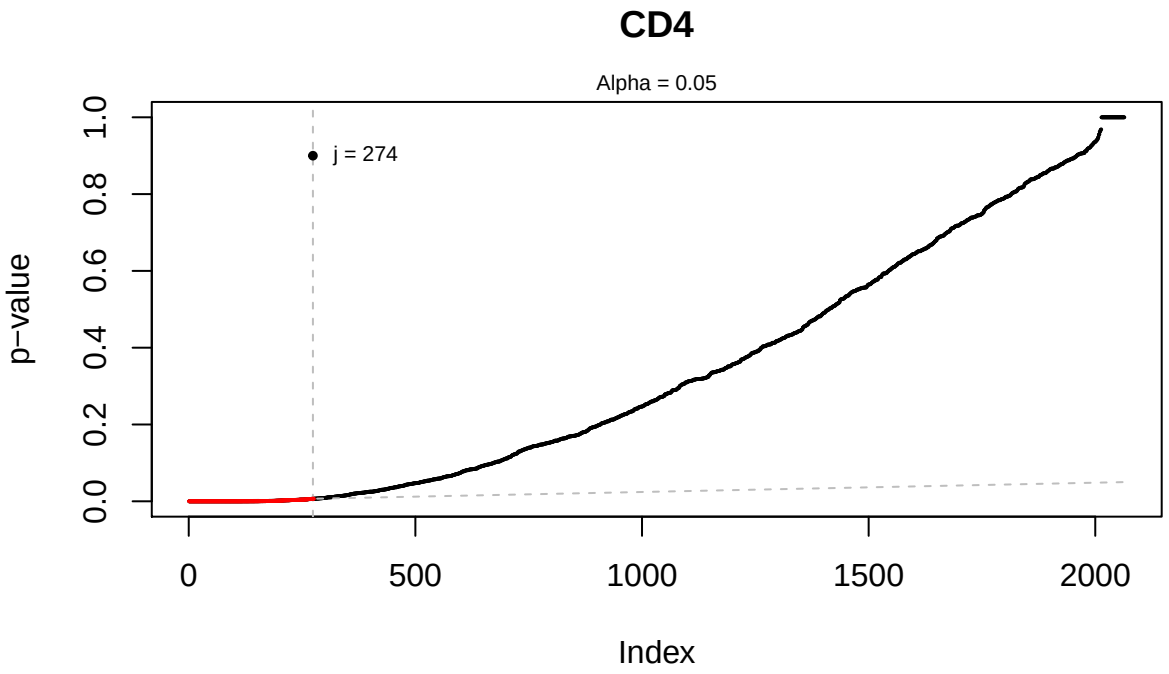
Model Comparison From the table below we can see that the best method regarding the error rate is SVM, but it also seems to be more complex with more support vectors used than the number of non-zero features in the other methods. Elastic net is the middle option here with a reasonable misclassification rate and a manageable number of features. When considering the time taken to run the different methods, the SVM comes out on top. Altogether the SVM seems to be the a great selection in this case.

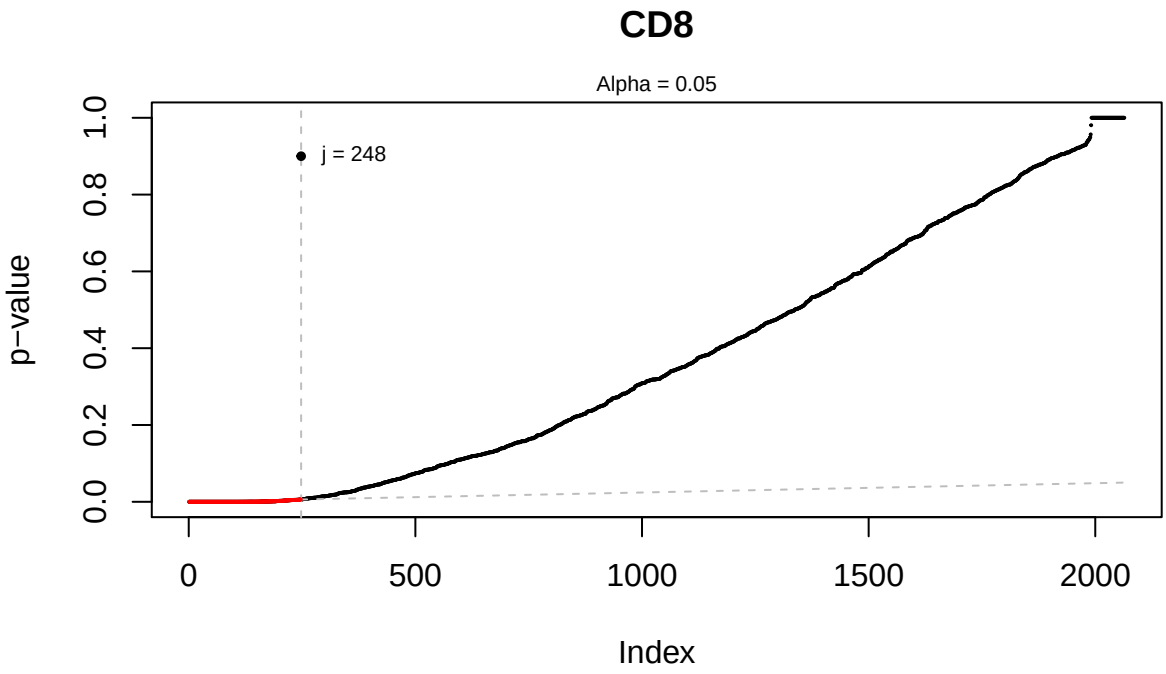
```
            Error Features Runtime
NSC         0.111       25   1.042
Elastic net 0.056       54   5.090
SVM         0.022       74   0.334
```

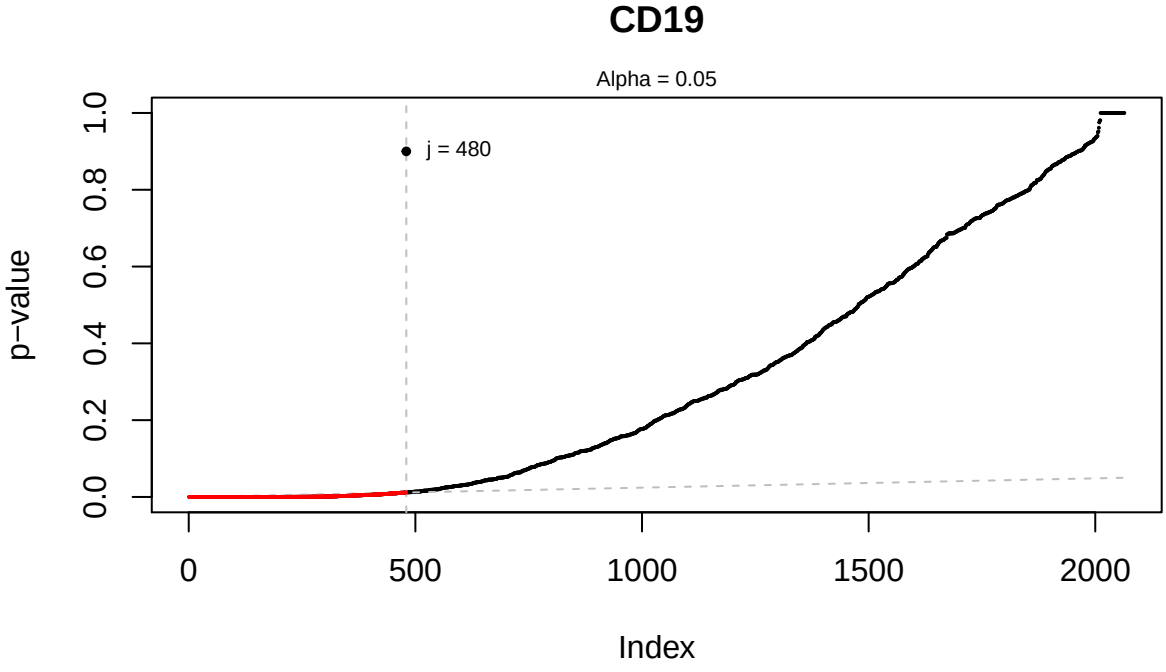
4. Benjamini-Hochberg method

For the final part of the assignment, we are going to use the Benjamini-Hochberg (BH) method to test each cell type against all others using all the features at hand (here we use t-tests). This should allow us to find the features that can discriminate well. The BH method allows us to fix a desired false discovery rate (FDR), denoted α, and utilise that it has been shown that the true FDR is smaller than α given that we order our p-values in ascending order and reject all null hypotheses that have p-values smaller than the BH rejection threshold.

Three plots are presented below, one for each cell type, using $\alpha = 0.05$ The p-values marked in red are those that are low enough to reject their corresponding null hypotheses, i.e. there is a statistically significant difference from the other cell types. It is clear that CD19 gets many more rejected hypotheses than the other types, indicating that it is easier to find features (genes) that can discriminate this cell type well from the others. It is interesting to note that this number of genes are much higher than in our previous attempts with NSC, SVM and elastic net.




CD8
Alpha = 0.05
j = 248
p−value
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
2000
Index


CD19
Alpha = 0.05
j = 480
p−value
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
2000
Index