Model 10 Predictions

Ethan Allavarpu (UID: 405287603)

11/02/2020

Transforming and Cleaning the Data

```
training <- read.csv("training.csv", stringsAsFactors = TRUE)
training$class <- factor(training$class)
levels(training$class) <- c("NG", "OG", "TSG")
outlier <- function(data) {
  low <- mean(data) - 3 * sd(data)
  high <- mean(data) + 3 * sd(data)
  which(data < low | data > high)
}
library(ggplot2)
scatter <- function(var) {
  ggplot(training, aes_string(var, "class")) +
    geom_jitter(width = 0.05, height = 0.1, size = 0.1,
                colour = rgb(0, 0, 0, alpha = 1 / 3))
}
scat_plot <- lapply(names(training)[-99], scatter)
library(gridExtra)
# grid.arrange(grobs = scat_plot[1:20], ncol = 4)
# grid.arrange(grobs = scat_plot[21:40], ncol = 4)
# grid.arrange(grobs = scat_plot[41:60], ncol = 4)
# grid.arrange(grobs = scat_plot[61:80], ncol = 4)
# grid.arrange(grobs = scat_plot[81:98], ncol = 4)
outlier_index <- sort(table(unlist(lapply(training[,-99], outlier))), decreasing = TRUE)
outlier_index[1:100]
```

```
 915 1280 2918  517 1914 2182 3052 1173 2215 3049  259  740 1749 1979 2998  417
  24   24   24   22   22   22   20   19   19   19   18   18   18   18   18   17
 441  806 2297  422  635 1258 1570 2278 2518 2729   80  150 2694  169  276  341
  17   17   17   16   16   16   16   16   16   16   15   15   15   14   14   14
1528 1556 1726 1809 1911 1955 2071 2624 2641 3120 3142   73  277  364  751 1244
  14   14   14   14   14   14   14   14   14   14   14   13   13   13   13   13
1330 2329 2787  343 1138 1171 1188 1372 1460 2031 2251 2968 2983 3166  352  634
  13   13   13   12   12   12   12   12   12   12   12   12   12   12   11   11
 907  923 1096 1858 2636  588 1137 1317 1463 1561 1740 1991 2487 2540 2555 2621
  11   11   11   11   11   10   10   10   10   10   10   10   10   10   10   10
2815 3029   74  144  657  789  857 1267 1610 1932 2022 2093 2142 2534 2666 2721
  10   10    9    9    9    9    9    9    9    9    9    9    9    9    9    9
2848 2900 3027  155
   9    9    9    8
```

```
training <- training[-as.numeric(names(outlier_index)[1:50]),]
sort(training$Missense_TO_Silent_Ratio, decreasing = TRUE)[1:10]
```

```
 [1] 384.98658 172.91420 135.59623  71.09712  23.21809  21.81193  20.37791
 [8]  19.42402  19.38769  15.84808
```

```
training <- training[-which(training$Missense_TO_Silent_Ratio > 100), ]
sort(training$Missense_KB_Ratio, decreasing = TRUE)[1:10]
```

```
 [1] 2063.9413 1296.6625 1060.0601  952.3810  931.4227  726.8519  594.7603
 [8]  593.3610  581.5085  516.8084
```

```
training <- training[-which(training$Missense_KB_Ratio > 2000), ]
sort(training$LOF_TO_Silent_Ratio, decreasing = TRUE)[1:10]
```

```
 [1] 81.177835  9.030120  6.470238  5.582840  4.741460  4.558252  4.176630
 [8]  4.058140  4.039062  4.021930
```

```
training <- training[-which(training$LOF_TO_Silent_Ratio > 5), ]
sort(training$Gene_expression_Z_score, decreasing = TRUE)[1:10]
```

```
 [1] 19.720  9.210  7.080  6.883  6.590  6.280  5.321  5.316  3.161  2.767
```

```
training <- training[-which(training$Gene_expression_Z_score > 4), ]
sort(training$dN_to_dS_ratio, decreasing = TRUE)[1:10]
```

```
 [1] 20.950  3.649  3.446  3.372  2.574  2.194  2.183  2.102  1.921  1.744
```

```
training <- training[-which(training$dN_to_dS_ratio > 5),]
sort(training$Silent_KB_Ratio, decreasing = TRUE)[1:10]
```

```
 [1] 474.4745 193.1684 174.0558 171.0362 166.4971 160.2273 158.7697 148.5800
 [9] 143.6782 135.2657
```

```
training <- training[-which(training$Silent_KB_Ratio > 200), ]
sort(training$Lost_start_and_stop_fraction, decreasing = TRUE)[1:10]
```

```
 [1] 0.333 0.167 0.118 0.087 0.074 0.071 0.071 0.068 0.067 0.067
```

```
training <- training[-which(training$Lost_start_and_stop_fraction > 0.2),]
sort(training$Synonymous_Zscore, decreasing = FALSE)[1:10]
```

```
 [1] -20.5110 -10.9780 -10.2960  -9.7346  -9.3720  -8.8090  -8.4062  -8.3918
 [9]  -8.1076  -8.1076
```

```
training <- training[-which(training$Synonymous_Zscore < -15), ]
numeric_training <- training[,-99]

n_zeroes <- rep(NA, nrow(numeric_training))

for(i in seq_len(nrow(numeric_training))){
  row_i_zeroes <- 0
  for(j in seq_len(ncol(numeric_training))){
    if(round(numeric_training[i,j], digits = 5) == 0){
      row_i_zeroes <- row_i_zeroes + 1
    }
  }
  n_zeroes[i] <- row_i_zeroes
}
```

```
training <- training[n_zeroes <= 50, ]
# Remove 1 of 2 variables that have r value of 1 or -1 (i.e. perfectly correlated)
perf_corr <- which(abs(cor(training[,-99])) == 1, arr.ind = TRUE)
which(perf_corr[,1] != perf_corr[,2])
```

```
Minus_Cell_proliferation_rate_CRISPR_KD       Cell_proliferation_rate_CRISPR_KD
                                     43                                      44
```

```
training <- training[, -43]
```

```
library(dplyr)
```

```
Attaching package: 'dplyr'

The following object is masked from 'package:gridExtra':

    combine

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
#function to calculate wca
score <- function (conf_mat) {
  print(sum(diag(conf_mat) * c(1, 20, 20)))
  print(sum(diag(conf_mat) * c(1, 20, 20)) / sum(apply(conf_mat, 2, sum) * c(1, 20, 20)))
}

# Set new threshold to account for unbalanced data
classify <- function(probs, k) {
  if (any(probs[2:3] > k)) {
    subset <- probs[2:3]
    output <- which(subset == max(subset))
    if (length(output) > 1) {
        output <- sample(1:2, 1)
    }
  } else {
    output <- 0
  }
  output
}
```

Multinom (Logistic Regression)

```
library(dplyr)
library(caret)
```

```
Loading required package: lattice
```

```
set.seed(12)
mn <- nnet::multinom(class ~ ., data = training, model = TRUE)
```

```
# weights:  297 (196 variable)
initial  value 3326.598010
iter  10 value 1352.245526
iter  20 value 1217.290354
iter  30 value 1085.878928
iter  40 value 942.253816
iter  50 value 839.781152
iter  60 value 772.029605
iter  70 value 678.667906
iter  80 value 550.944852
iter  90 value 446.725235
iter 100 value 355.375074
final  value 355.375074
stopped after 100 iterations
```

```
tidymn <- broom::tidy(mn) %>% arrange(p.value)
tidymn <- tidymn %>% filter(p.value < 0.05 / nrow(tidymn))
terms <- unique(tidymn$term)[-1]
terms
```

```
 [1] "Missense_Entropy"
 [2] "LOF_TO_Silent_Ratio"
 [3] "Splice_TO_Silent_Ratio"
 [4] "LOF_TO_Benign_Ratio"
 [5] "Splice_TO_Benign_Ratio"
 [6] "Missense_Damaging_TO_Benign_Ratio"
 [7] "Polyphen2"
 [8] "LOF_TO_Total_Ratio"
 [9] "Missense_TO_Total_Ratio"
[10] "Splice_TO_Total_Ratio"
[11] "LOF_TO_Missense_Ratio"
[12] "Silent_fraction"
[13] "Nonsense_fraction"
[14] "Missense_fraction"
[15] "Recurrent_missense_fraction"
[16] "Frameshift_indel_fraction"
[17] "Inframe_indel_fraction"
[18] "Lost_start_and_stop_fraction"
[19] "Inactivating_mutations_fraction"
[20] "CNA_deletion"
[21] "CNA_amplification"
[22] "Exon_Cons"
[23] "MGAentropy"
[24] "S50_score_replication_timing"
[25] "One_Minus_S50_score_replication_timing"
[26] "VEST_score"
```

```
[27] "Gene_expression_Z_score"
[28] "Gene_expression_Minus_Z_score"
[29] "Cell_proliferation_rate_CRISPR_KD"
[30] "Super_Enhancer_percentage"
[31] "BioGRID_betweenness"
[32] "BioGRID_clossness"
[33] "Promoter_hypermethylation_in_cancer"
[34] "Promoter_hypomethylation_in_cancer"
[35] "Gene_body_hypermethylation_in_cancer"
[36] "Gene_body_hypomethylation_in_cancer"
[37] "intolerant_pLI"
[38] "intolerant_pRec"
[39] "intolerant_pNull"
[40] "dN_to_dS_ratio"
[41] "Broad_H3K4me3_percentage"
[42] "Broad_H3K4me2_percentage"
[43] "Broad_H3K4me1_percentage"
[44] "Broad_H3K36me3_percentage"
[45] "Broad_H3K27ac_percentage"
[46] "Broad_H3K27me3_percentage"
[47] "Broad_H3K9me3_percentage"
[48] "Broad_H3K9ac_percentage"
[49] "Broad_H3K9me2_percentage"
[50] "Broad_H3K79me2_percentage"
[51] "Broad_H4K20me1_percentage"
[52] "Missense_TO_Benign_Ratio"
[53] "Missense_TO_Silent_Ratio"
[54] "N_Missense"
[55] "CDS_length"
[56] "ncGERP"
[57] "NonSilent_TO_Silent_Ratio"
[58] "ncRVIS"
[59] "BioGRID_log_degree"
[60] "N_LOF"
[61] "N_Splice"
[62] "Missense_Damaging_TO_Missense_Benign_Ratio"
[63] "Missense_Zscore"
[64] "H3K9me2_height"
```

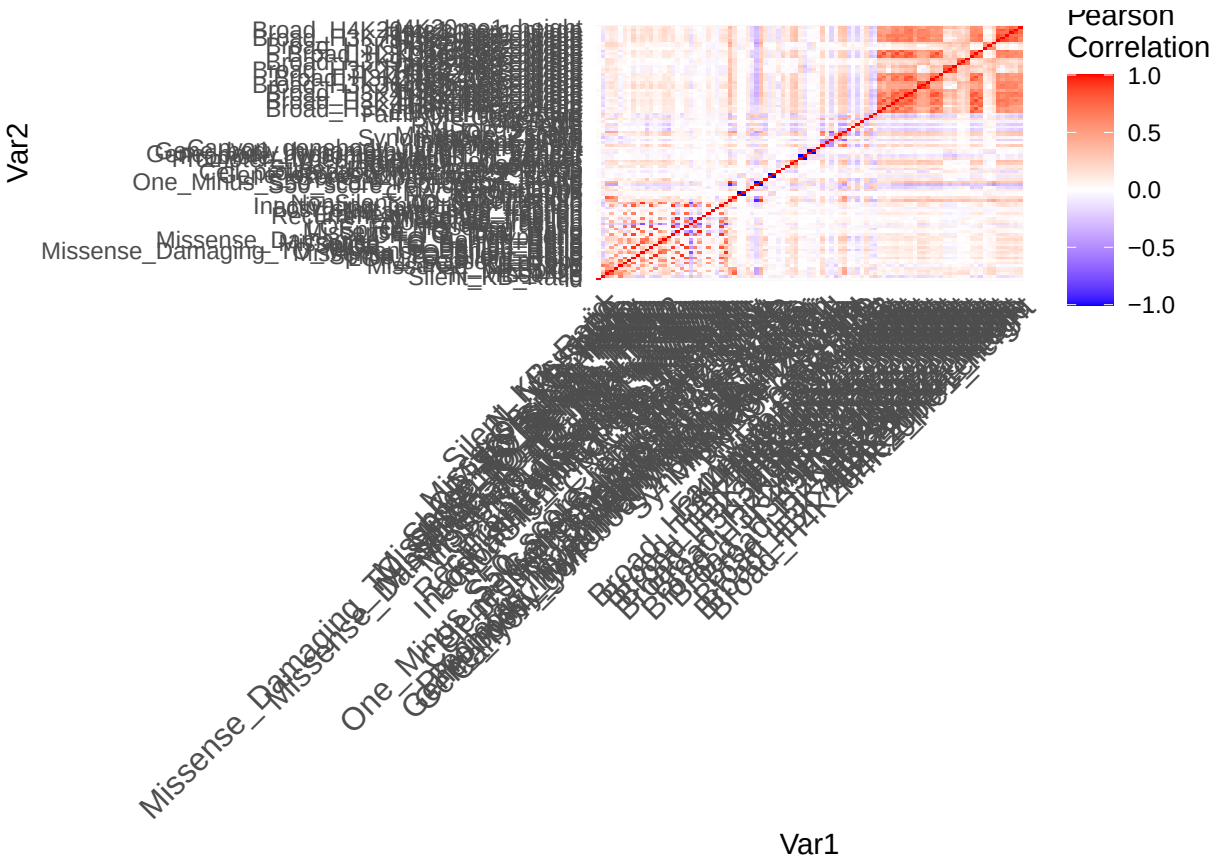
```
library(dplyr)
sig <- logical(97)
names(sig) <- names(training)[-98]
k <- 1
diffs <- logical(97)
for (var in names(training)[-98]) {
  model <- aov(training[[var]] ~ factor(training$class))
  sig[k] <- summary(model)[[1]][1, 5]
  diffs[k] <- all(TukeyHSD(model)$`factor(training$class)`[, 4] < 0.05 / 97)
  k <- k + 1
}
sort(sig[diffs])
```

```
Broad_H4K20me1_percentage
                2.563341e-146
                   VEST_score
```

1.141090e-121
Broad_H3K9ac_percentage
7.096289e-111
H3K79me2_height
1.252905e-110
H3K79me2_width
5.618333e-110
Missense_Entropy
1.254512e-109
Broad_H3K79me2_percentage
4.654407e-107
Missense_Damaging_TO_Benign_Ratio
1.510740e-106
Broad_H3K4me2_percentage
1.195197e-105
H4K20me1_width
6.869405e-105
Broad_H3K36me3_percentage
8.244495e-104
H3K36me3_width
4.633126e-103
H4K20me1_height
1.332273e-101
LOF_TO_Silent_Ratio
4.745882e-101
Broad_H3K27ac_percentage
5.173854e-97
Broad_H3K4me1_percentage
1.023092e-95
LOF_KB_Ratio
4.040183e-93
Missense_KB_Ratio
4.866503e-89
H3K4me1_width
6.433999e-80
LOF_TO_Benign_Ratio
1.552250e-78
Broad_H3K4me3_percentage
3.171192e-78
H3K36me3_height
2.795995e-75
H3K4me2_width
9.456016e-74
H3K9ac_width
1.107951e-71
Missense_Damaging_TO_Missense_Benign_Ratio
2.508133e-68
H3K9ac_height
1.751625e-65
Missense_TO_Benign_Ratio
5.840566e-64
H3K27ac_width
2.548087e-63
Length_H3K4me3

```
                          1.108383e-60
                       H3K27ac_height
                          2.528240e-58
                   LOF_TO_Total_Ratio
                          8.428823e-48
                LOF_TO_Missense_Ratio
                          1.945822e-42
                       H3K4me1_height
                          6.778589e-42
           Frameshift_indel_fraction
                          5.027810e-34
```

```r
score <- function (conf_mat) {
  sum(diag(conf_mat) * c(1, 20, 20)) / sum(apply(conf_mat, 2, sum) * c(1, 20, 20))
}

sig_terms <- names(sort(sig[diffs]))
combo_terms <- table(c(terms, sig_terms)) == 2
final_terms <- names(combo_terms[combo_terms])
term_mat <- cor(training %>% select(all_of(final_terms)))

ind_terms <- final_terms[-c(1, 3, 4, 5, 10, 11, 13, 14, 15, 17)]
vars <- training
cor_mtx = round(cor(vars[, names(vars) != "class"]), 2)
library(reshape2)
#reshape it
melted_cor_mtx <- melt(cor_mtx)

#draw the heatmap
cor_heatmap = ggplot(data = melted_cor_mtx, aes(x=Var1, y=Var2, fill=value)) + geom_tile()
cor_heatmap = cor_heatmap +
scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1,1), space =
theme_minimal() +
theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12, hjust = 1))

cor_heatmap
```




```
which(abs(cor_mtx) > 0.75, arr.ind = TRUE)
```

```
                                          row col
id                                          1   1
Silent_KB_Ratio                             2   2
N_Missense                                  3   3
N_LOF                                       4   3
CDS_length                                 32   3
N_Missense                                  3   4
N_LOF                                       4   4
N_Splice                                    5   5
Missense_KB_Ratio                           6   6
LOF_KB_Ratio                                7   7
Missense_Entropy                            8   8
LOF_TO_Silent_Ratio                         9   9
LOF_TO_Benign_Ratio                        13   9
LOF_TO_Total_Ratio                         18   9
LOF_TO_Missense_Ratio                      21   9
Inactivating_mutations_fraction            29   9
Splice_TO_Silent_Ratio                     10  10
Splice_TO_Benign_Ratio                     14  10
Splice_TO_Total_Ratio                      20  10
Missense_TO_Silent_Ratio                   11  11
Missense_TO_Benign_Ratio                   15  11
NonSilent_TO_Silent_Ratio                  30  11
Missense_Damaging_TO_Missense_Benign_Ratio 12  12
LOF_TO_Silent_Ratio                         9  13
```

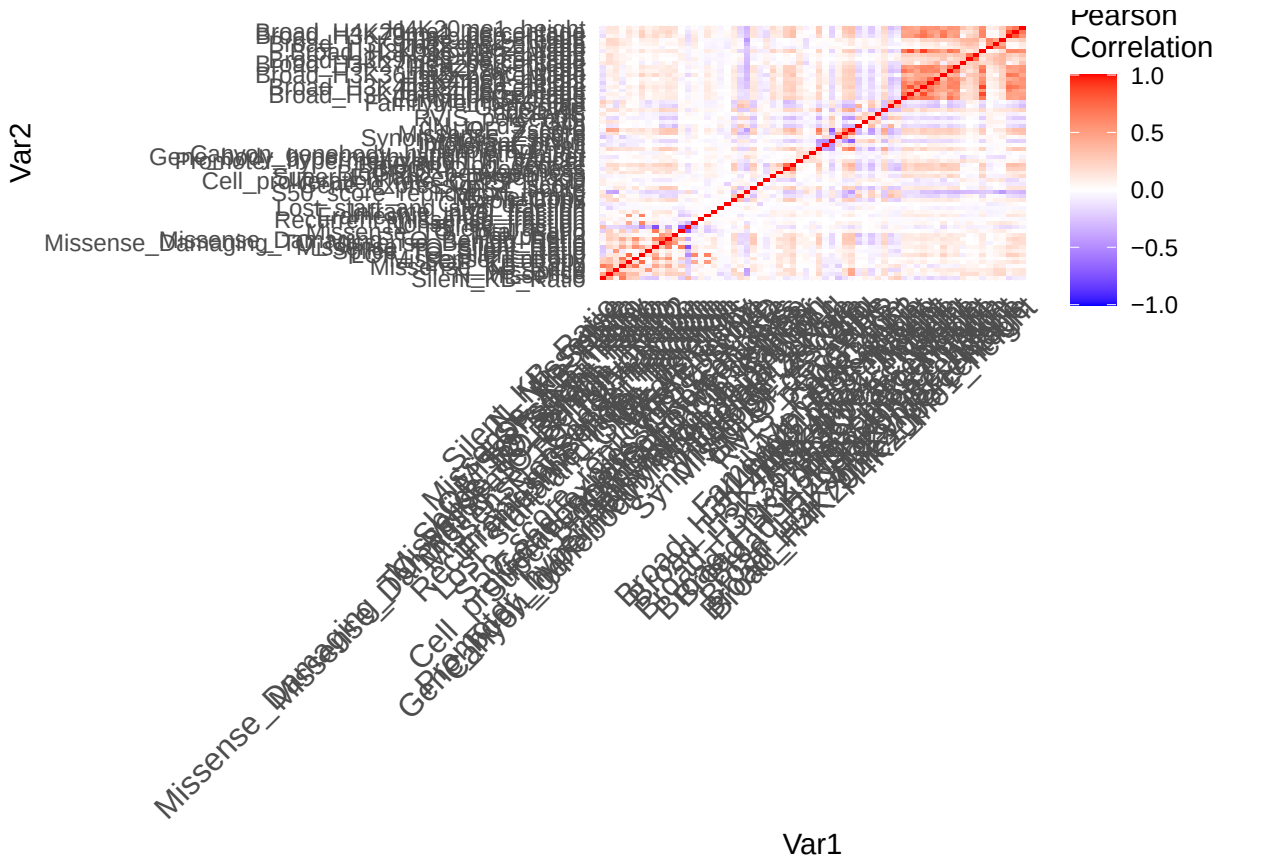
LOF_TO_Benign_Ratio	13	13
LOF_TO_Total_Ratio	18	13
LOF_TO_Missense_Ratio	21	13
Inactivating_mutations_fraction	29	13
Splice_TO_Silent_Ratio	10	14
Splice_TO_Benign_Ratio	14	14
Splice_TO_Total_Ratio	20	14
Missense_TO_Silent_Ratio	11	15
Missense_TO_Benign_Ratio	15	15
Missense_Damaging_TO_Benign_Ratio	16	16
Polyphen2	17	17
LOF_TO_Silent_Ratio	9	18
LOF_TO_Benign_Ratio	13	18
LOF_TO_Total_Ratio	18	18
LOF_TO_Missense_Ratio	21	18
Inactivating_mutations_fraction	29	18
Missense_TO_Total_Ratio	19	19
Missense_fraction	24	19
Splice_TO_Silent_Ratio	10	20
Splice_TO_Benign_Ratio	14	20
Splice_TO_Total_Ratio	20	20
LOF_TO_Silent_Ratio	9	21
LOF_TO_Benign_Ratio	13	21
LOF_TO_Total_Ratio	18	21
LOF_TO_Missense_Ratio	21	21
Inactivating_mutations_fraction	29	21
Silent_fraction	22	22
NonSilent_TO_Silent_Ratio	30	22
Nonsense_fraction	23	23
Missense_TO_Total_Ratio	19	24
Missense_fraction	24	24
Recurrent_missense_fraction	25	25
Frameshift_indel_fraction	26	26
Inframe_indel_fraction	27	27
Lost_start_and_stop_fraction	28	28
LOF_TO_Silent_Ratio	9	29
LOF_TO_Benign_Ratio	13	29
LOF_TO_Total_Ratio	18	29
LOF_TO_Missense_Ratio	21	29
Inactivating_mutations_fraction	29	29
Missense_TO_Silent_Ratio	11	30
Silent_fraction	22	30
NonSilent_TO_Silent_Ratio	30	30
log_gene_length	31	31
S50_score_replication_timing	37	31
One_Minus_S50_score_replication_timing	38	31
N_Missense	3	32
CDS_length	32	32
CNA_deletion	33	33
CNA_amplification	34	33
CNA_deletion	33	34
CNA_amplification	34	34
Exon_Cons	35	35
MGAentropy	36	36

log_gene_length	31	37
S50_score_replication_timing	37	37
One_Minus_S50_score_replication_timing	38	37
log_gene_length	31	38
S50_score_replication_timing	37	38
One_Minus_S50_score_replication_timing	38	38
VEST_score	39	39
Gene_expression_Z_score	40	40
Gene_expression_Minus_Z_score	41	40
Gene_expression_Z_score	40	41
Gene_expression_Minus_Z_score	41	41
Cell_proliferation_rate_CRISPR_KD	42	42
Super_Enhancer_percentage	43	43
BioGRID_betweenness	44	44
BioGRID_clossness	45	45
BioGRID_log_degree	46	46
Promoter_hypermethylation_in_cancer	47	47
Promoter_hypomethylation_in_cancer	48	47
Promoter_hypermethylation_in_cancer	47	48
Promoter_hypomethylation_in_cancer	48	48
Gene_body_hypermethylation_in_cancer	49	49
Gene_body_hypomethylation_in_cancer	50	49
Gene_body_hypermethylation_in_cancer	49	50
Gene_body_hypomethylation_in_cancer	50	50
Canyon_genebody_hypermethylation	51	51
intolerant_pLI	52	52
intolerant_pRec	53	53
intolerant_pNull	54	54
Synonymous_Zscore	55	55
Missense_Zscore	56	56
pLOF_Zscore	57	57
dN_to_dS_ratio	58	58
GDI	59	59
RVIS_percentile	60	60
ncRVIS	61	61
ncGERP	62	62
Gene_age	63	63
FamilyMemberCount	64	64
Length_H3K4me3	65	65
H3K4me2_width	68	65
H3K9ac_width	86	65
Broad_H3K4me3_percentage	66	66
Broad_H3K4me2_percentage	69	66
Broad_H3K9ac_percentage	87	66
H3K4me3_height	67	67
H3K9ac_height	88	67
Length_H3K4me3	65	68
H3K4me2_width	68	68
Broad_H3K4me2_percentage	69	68
H3K4me2_height	70	68
H3K4me1_width	71	68
H3K9ac_width	86	68
Broad_H3K4me3_percentage	66	69
H3K4me2_width	68	69

Broad_H3K4me2_percentage	69	69
Broad_H3K4me1_percentage	72	69
Broad_H3K27ac_percentage	78	69
Broad_H3K9ac_percentage	87	69
H3K4me2_width	68	70
H3K4me2_height	70	70
H3K9ac_height	88	70
H3K4me2_width	68	71
H3K4me1_width	71	71
Broad_H3K4me1_percentage	72	71
H3K27ac_width	77	71
Broad_H3K27ac_percentage	78	71
H3K9ac_width	86	71
Broad_H3K9ac_percentage	87	71
Broad_H3K4me2_percentage	69	72
H3K4me1_width	71	72
Broad_H3K4me1_percentage	72	72
Broad_H3K27ac_percentage	78	72
Broad_H3K9ac_percentage	87	72
H3K4me1_height	73	73
H3K36me3_width	74	74
Broad_H3K36me3_percentage	75	74
H3K36me3_height	76	74
H3K36me3_width	74	75
Broad_H3K36me3_percentage	75	75
H3K36me3_height	76	75
H3K36me3_width	74	76
Broad_H3K36me3_percentage	75	76
H3K36me3_height	76	76
H3K4me1_width	71	77
H3K27ac_width	77	77
Broad_H3K27ac_percentage	78	77
H3K9ac_width	86	77
Broad_H3K4me2_percentage	69	78
H3K4me1_width	71	78
Broad_H3K4me1_percentage	72	78
H3K27ac_width	77	78
Broad_H3K27ac_percentage	78	78
Broad_H3K9ac_percentage	87	78
H3K27ac_height	79	79
H3K27me3_width	80	80
Broad_H3K27me3_percentage	81	80
H3K27me3_height	82	80
H3K27me3_width	80	81
Broad_H3K27me3_percentage	81	81
H3K27me3_height	82	81
H3K27me3_width	80	82
Broad_H3K27me3_percentage	81	82
H3K27me3_height	82	82
H3K9me3_width	83	83
Broad_H3K9me3_percentage	84	83
H3K9me3_width	83	84
Broad_H3K9me3_percentage	84	84
H3K9me3_height	85	85

```
Length_H3K4me3                          65  86
H3K4me2_width                           68  86
H3K4me1_width                           71  86
H3K27ac_width                           77  86
H3K9ac_width                            86  86
Broad_H3K9ac_percentage                 87  86
Broad_H3K4me3_percentage                66  87
Broad_H3K4me2_percentage                69  87
H3K4me1_width                           71  87
Broad_H3K4me1_percentage                72  87
Broad_H3K27ac_percentage                78  87
H3K9ac_width                            86  87
Broad_H3K9ac_percentage                 87  87
H3K4me3_height                          67  88
H3K4me2_height                          70  88
H3K9ac_height                           88  88
H3K9me2_width                           89  89
Broad_H3K9me2_percentage                90  90
H3K9me2_height                          91  91
H3K79me2_width                          92  92
Broad_H3K79me2_percentage               93  92
H3K79me2_height                         94  92
Broad_H4K20me1_percentage               96  92
H3K79me2_width                          92  93
Broad_H3K79me2_percentage               93  93
H3K79me2_height                         94  93
Broad_H4K20me1_percentage               96  93
H3K79me2_width                          92  94
Broad_H3K79me2_percentage               93  94
H3K79me2_height                         94  94
H4K20me1_width                          95  95
Broad_H4K20me1_percentage               96  95
H3K79me2_width                          92  96
Broad_H3K79me2_percentage               93  96
H4K20me1_width                          95  96
Broad_H4K20me1_percentage               96  96
H4K20me1_height                         97  97
```

```r
vars <- vars[, -c(1, 4, 32, 13, 18, 21, 29, 14, 20, 15, 30, 15, 24, 31, 38, 34, 41, 48, 50, 68, 86, 88,
cor_mtx = round(cor(vars[, names(vars) != "class"]), 2)
library(reshape2)
#reshape it
melted_cor_mtx <- melt(cor_mtx)

#draw the heatmap
cor_heatmap = ggplot(data = melted_cor_mtx, aes(x=Var1, y=Var2, fill=value)) + geom_tile()
cor_heatmap = cor_heatmap +
scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1,1), space =
theme_minimal() +
theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12, hjust = 1))

cor_heatmap
```




```
which(abs(cor_mtx) > 0.6, arr.ind = TRUE)
```

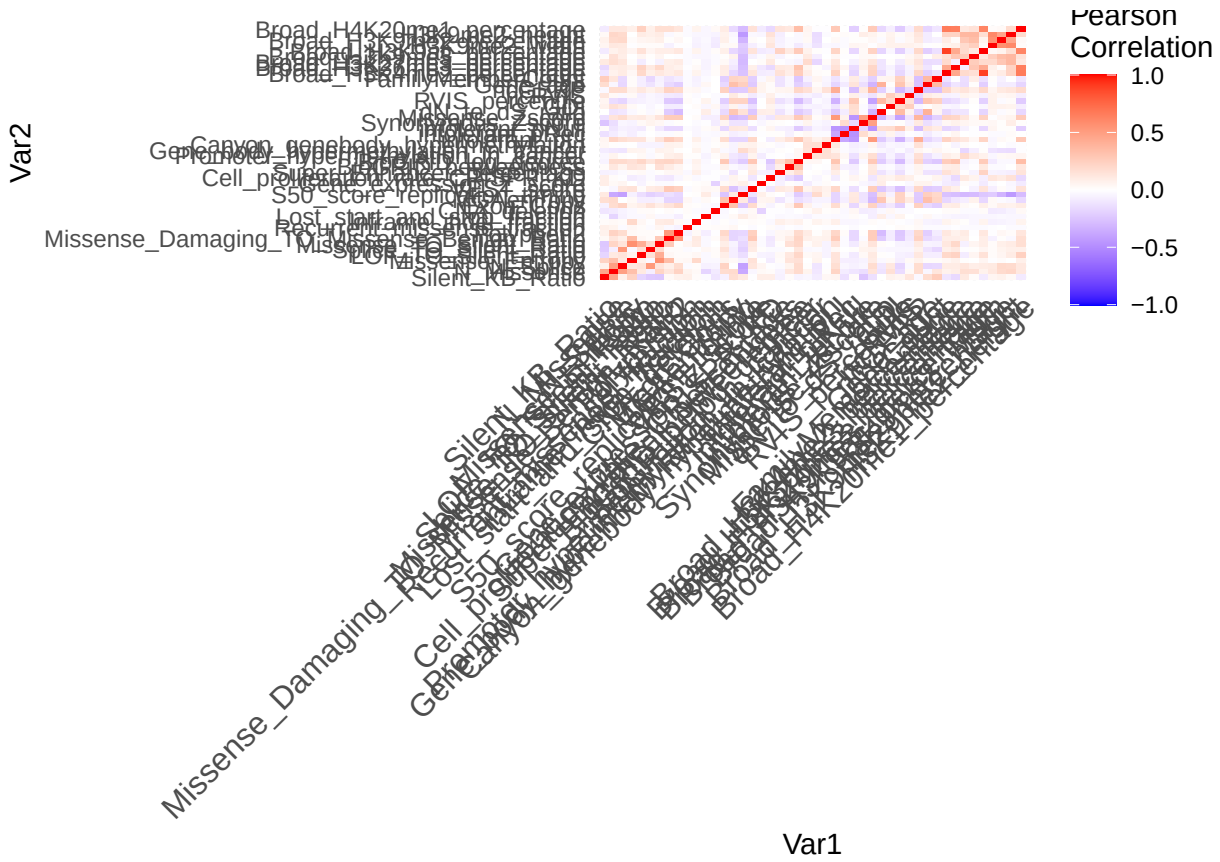
```
                                          row col
Silent_KB_Ratio                             1   1
Missense_KB_Ratio                           4   1
N_Missense                                  2   2
N_Splice                                    3   3
Silent_KB_Ratio                             1   4
Missense_KB_Ratio                           4   4
Missense_Entropy                            6   4
LOF_KB_Ratio                                5   5
LOF_TO_Silent_Ratio                         7   5
Missense_KB_Ratio                           4   6
Missense_Entropy                            6   6
LOF_KB_Ratio                                5   7
LOF_TO_Silent_Ratio                         7   7
Nonsense_fraction                          15   7
Frameshift_indel_fraction                  17   7
Splice_TO_Silent_Ratio                      8   8
Missense_TO_Silent_Ratio                    9   9
Missense_Damaging_TO_Benign_Ratio          11   9
Missense_TO_Total_Ratio                    13   9
Silent_fraction                            14   9
Missense_Damaging_TO_Missense_Benign_Ratio 10  10
Missense_Damaging_TO_Benign_Ratio          11  10
Missense_TO_Silent_Ratio                    9  11
Missense_Damaging_TO_Missense_Benign_Ratio 10  11
```

Missense_Damaging_TO_Benign_Ratio	11	11
Polyphen2	12	12
Missense_TO_Silent_Ratio	9	13
Missense_TO_Total_Ratio	13	13
Silent_fraction	14	13
Missense_TO_Silent_Ratio	9	14
Missense_TO_Total_Ratio	13	14
Silent_fraction	14	14
LOF_TO_Silent_Ratio	7	15
Nonsense_fraction	15	15
Recurrent_missense_fraction	16	16
LOF_TO_Silent_Ratio	7	17
Frameshift_indel_fraction	17	17
Inframe_indel_fraction	18	18
Lost_start_and_stop_fraction	19	19
CNA_deletion	20	20
Exon_Cons	21	21
MGAentropy	22	22
S50_score_replication_timing	23	23
VEST_score	24	24
Gene_expression_Z_score	25	25
Cell_proliferation_rate_CRISPR_KD	26	26
Super_Enhancer_percentage	27	27
BioGRID_betweenness	28	28
BioGRID_clossness	29	29
BioGRID_log_degree	30	29
BioGRID_clossness	29	30
BioGRID_log_degree	30	30
Promoter_hypermethylation_in_cancer	31	31
Gene_body_hypermethylation_in_cancer	32	32
Canyon_genebody_hypermethylation	33	33
intolerant_pLI	34	34
pLOF_Zscore	39	34
intolerant_pRec	35	35
intolerant_pNull	36	36
pLOF_Zscore	39	36
Synonymous_Zscore	37	37
Missense_Zscore	38	38
pLOF_Zscore	39	38
intolerant_pLI	34	39
intolerant_pNull	36	39
Missense_Zscore	38	39
pLOF_Zscore	39	39
dN_to_dS_ratio	40	40
GDI	41	41
RVIS_percentile	42	42
ncRVIS	43	43
ncGERP	44	44
Gene_age	45	45
FamilyMemberCount	46	46
Length_H3K4me3	47	47
Broad_H3K4me3_percentage	48	47
Broad_H3K4me2_percentage	50	47
H3K4me2_height	51	47

H3K27ac_width	54	47
Broad_H3K9ac_percentage	59	47
Length_H3K4me3	47	48
Broad_H3K4me3_percentage	48	48
Broad_H3K4me2_percentage	50	48
Broad_H3K9ac_percentage	59	48
H3K4me3_height	49	49
H3K4me2_height	51	49
Broad_H3K79me2_percentage	63	49
Length_H3K4me3	47	50
Broad_H3K4me3_percentage	48	50
Broad_H3K4me2_percentage	50	50
H3K4me2_height	51	50
H3K27ac_width	54	50
H3K27ac_height	55	50
Broad_H3K9ac_percentage	59	50
Broad_H3K79me2_percentage	63	50
Broad_H4K20me1_percentage	64	50
Length_H3K4me3	47	51
H3K4me3_height	49	51
Broad_H3K4me2_percentage	50	51
H3K4me2_height	51	51
H3K4me1_height	52	51
H3K27ac_height	55	51
H3K4me2_height	51	52
H3K4me1_height	52	52
H3K27ac_width	54	52
Broad_H3K36me3_percentage	53	53
Broad_H3K79me2_percentage	63	53
Broad_H4K20me1_percentage	64	53
Length_H3K4me3	47	54
Broad_H3K4me2_percentage	50	54
H3K4me1_height	52	54
H3K27ac_width	54	54
H3K27ac_height	55	54
Broad_H3K9ac_percentage	59	54
Broad_H3K4me2_percentage	50	55
H3K4me2_height	51	55
H3K27ac_width	54	55
H3K27ac_height	55	55
Broad_H3K9ac_percentage	59	55
Broad_H3K27me3_percentage	56	56
Broad_H3K9me3_percentage	57	57
H3K9me3_height	58	57
Broad_H3K9me3_percentage	57	58
H3K9me3_height	58	58
Length_H3K4me3	47	59
Broad_H3K4me3_percentage	48	59
Broad_H3K4me2_percentage	50	59
H3K27ac_width	54	59
H3K27ac_height	55	59
Broad_H3K9ac_percentage	59	59
Broad_H3K79me2_percentage	63	59
Broad_H4K20me1_percentage	64	59

```
H3K9me2_width                          60  60
Broad_H3K9me2_percentage               61  61
H3K9me2_height                         62  62
H3K4me3_height                         49  63
Broad_H3K4me2_percentage               50  63
Broad_H3K36me3_percentage              53  63
Broad_H3K9ac_percentage                59  63
Broad_H3K79me2_percentage              63  63
Broad_H4K20me1_percentage              64  63
Broad_H3K4me2_percentage               50  64
Broad_H3K36me3_percentage              53  64
Broad_H3K9ac_percentage                59  64
Broad_H3K79me2_percentage              63  64
Broad_H4K20me1_percentage              64  64
H4K20me1_height                        65  64
Broad_H4K20me1_percentage              64  65
H4K20me1_height                        65  65
```

```r
vars <- vars[, -c(4, 5, 15, 17, 11, 13, 14, 29, 39, 47, 48, 51, 49,
                  54, 55, 52, 58, 63, 65)]
cor_mtx = round(cor(vars[, names(vars) != "class"]), 2)
library(reshape2)
#reshape it
melted_cor_mtx <- melt(cor_mtx)

#draw the heatmap
cor_heatmap = ggplot(data = melted_cor_mtx, aes(x=Var1, y=Var2, fill=value)) + geom_tile()
cor_heatmap = cor_heatmap +
scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1,1), space =
theme_minimal() +
theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12, hjust = 1))

cor_heatmap
```



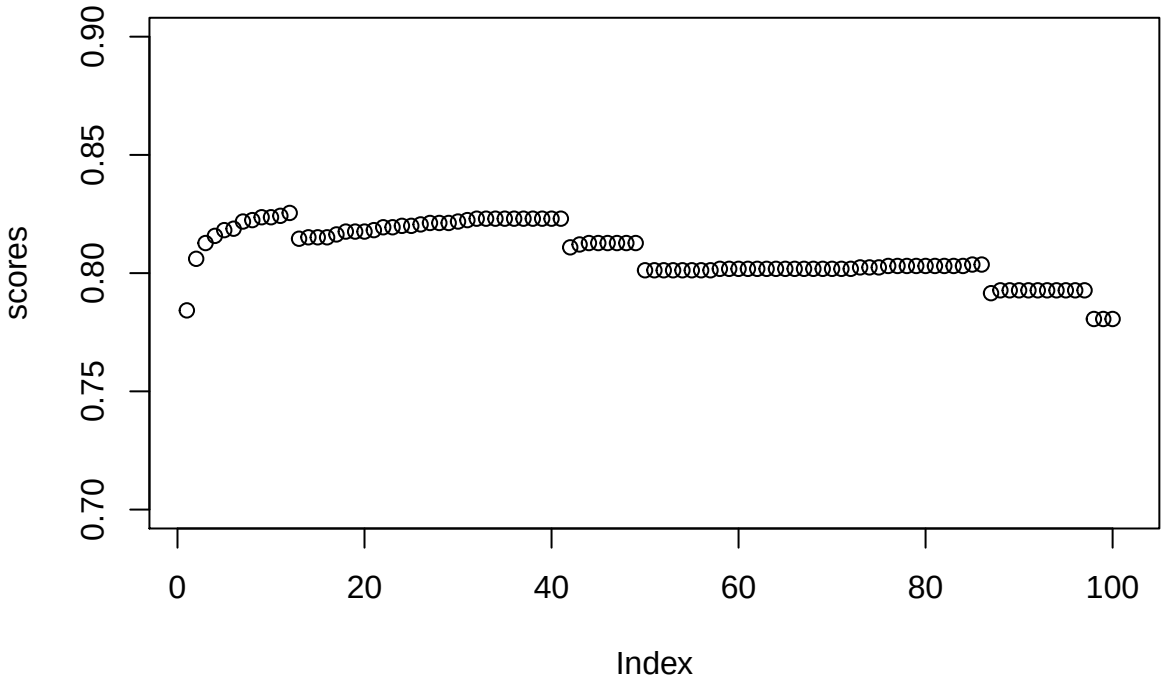

```
final_terms
```

```
 [1] "Broad_H3K27ac_percentage"
 [2] "Broad_H3K36me3_percentage"
 [3] "Broad_H3K4me1_percentage"
 [4] "Broad_H3K4me2_percentage"
 [5] "Broad_H3K4me3_percentage"
 [6] "Broad_H3K79me2_percentage"
 [7] "Broad_H3K9ac_percentage"
 [8] "Broad_H4K20me1_percentage"
 [9] "Frameshift_indel_fraction"
[10] "LOF_TO_Benign_Ratio"
[11] "LOF_TO_Missense_Ratio"
[12] "LOF_TO_Silent_Ratio"
[13] "LOF_TO_Total_Ratio"
[14] "Missense_Damaging_TO_Benign_Ratio"
[15] "Missense_Damaging_TO_Missense_Benign_Ratio"
[16] "Missense_Entropy"
[17] "Missense_TO_Benign_Ratio"
[18] "VEST_score"
```

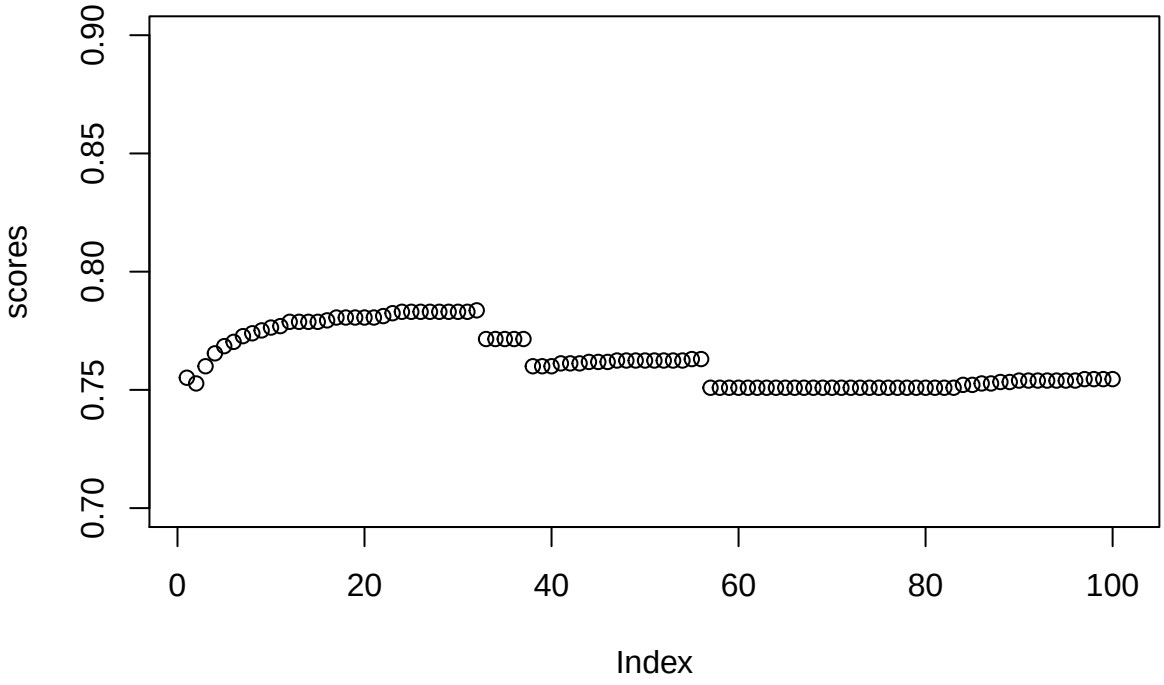
```
# vars <- training %>% select(Broad_H3K9ac_percentage, N_LOF, pLOF_Zscore,
#                             Missense_Entropy, VEST_score, BioGRID_log_degree, dN_to_dS_ratio,
#                             Broad_H3K79me2_percentage, Gene_expression_Z_score, S50_score_replication_
#                             Polyphen2, Broad_H3K36me3_percentage, class)
# vars <- training
set.seed(376524)
```

```r
thresh_list <- list()
length(thresh_list) <- 10
j <- 1
for (rand in ceiling(runif(10, min = 1, max = 10000000))) {
  set.seed(rand)
  vars_test <- createDataPartition(vars$class, p = 0.8,
                                   list = FALSE)
  vars_train <- vars[vars_test, ]
  vars_test <- vars[-vars_test, ]

  train_cont <- trainControl(method = "cv", number = 5, classProbs = TRUE, savePredictions = TRUE)
  lda_ft <- train(class ~ ., data = vars_train, method = "lda", preProc = c("center", "scale"),
                  trControl = train_cont)
  preds <- predict(lda_ft, newdata = vars_test, type = "prob")
  scores <- numeric(101)
  for (i in seq(from = 0, to = 0.1, by = 0.001)) {
    lda_mod <- table("pred"=apply(preds, 1, classify, k = i), "obs" = vars_test$class)
    lda_mod
    scores[i*1000] <- score(lda_mod)
  }
  names(scores) <- seq(from = 0, to = 0.1, by = 0.001)
  plot(scores, ylim = c(0.7, 0.9))
  best_thresh <- names(scores)[scores == max(scores)]
  max(scores)
  thresh_list[[j]] <- best_thresh
  j <- j + 1
}
```

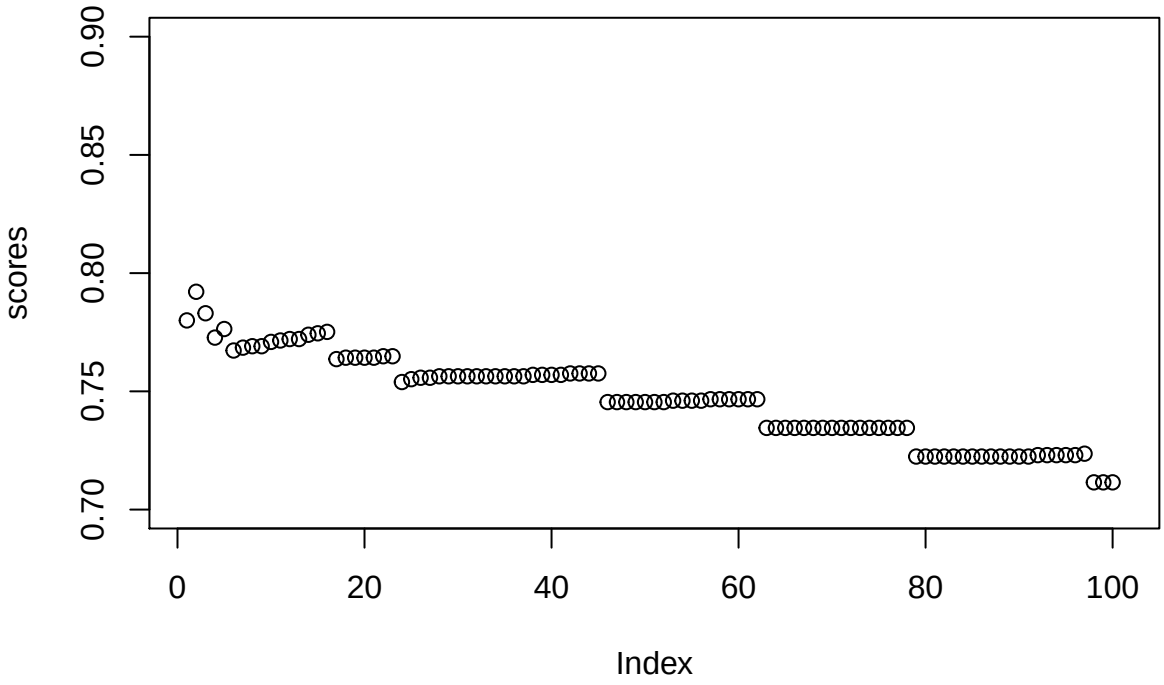
```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length

Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



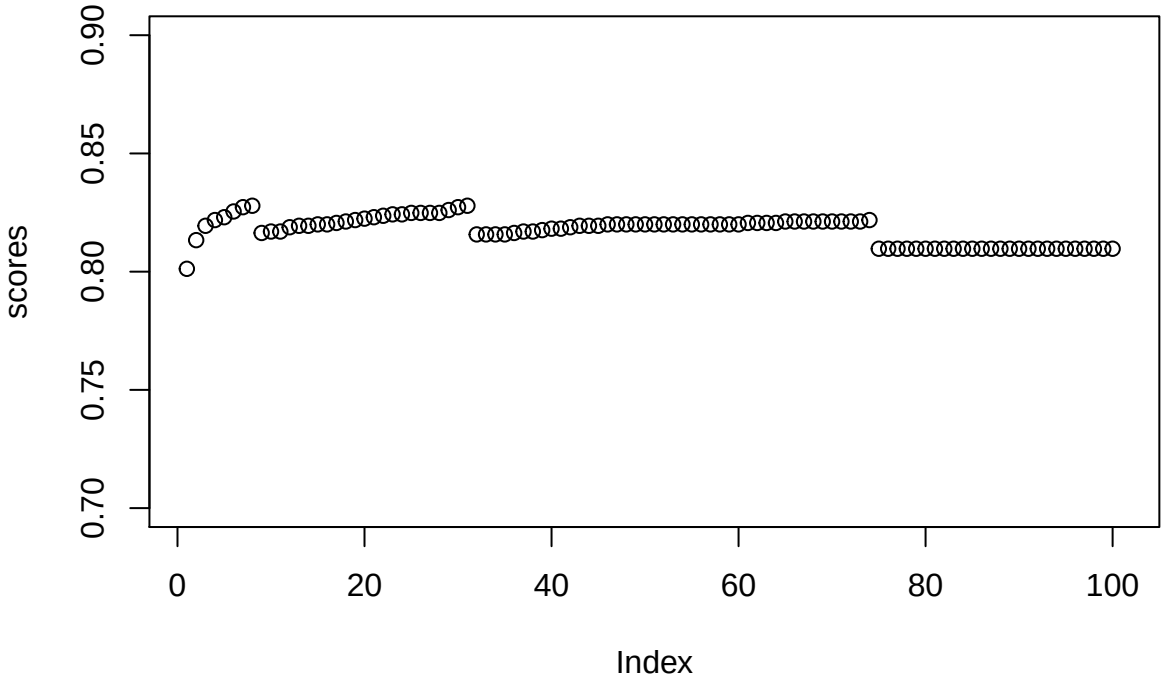

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



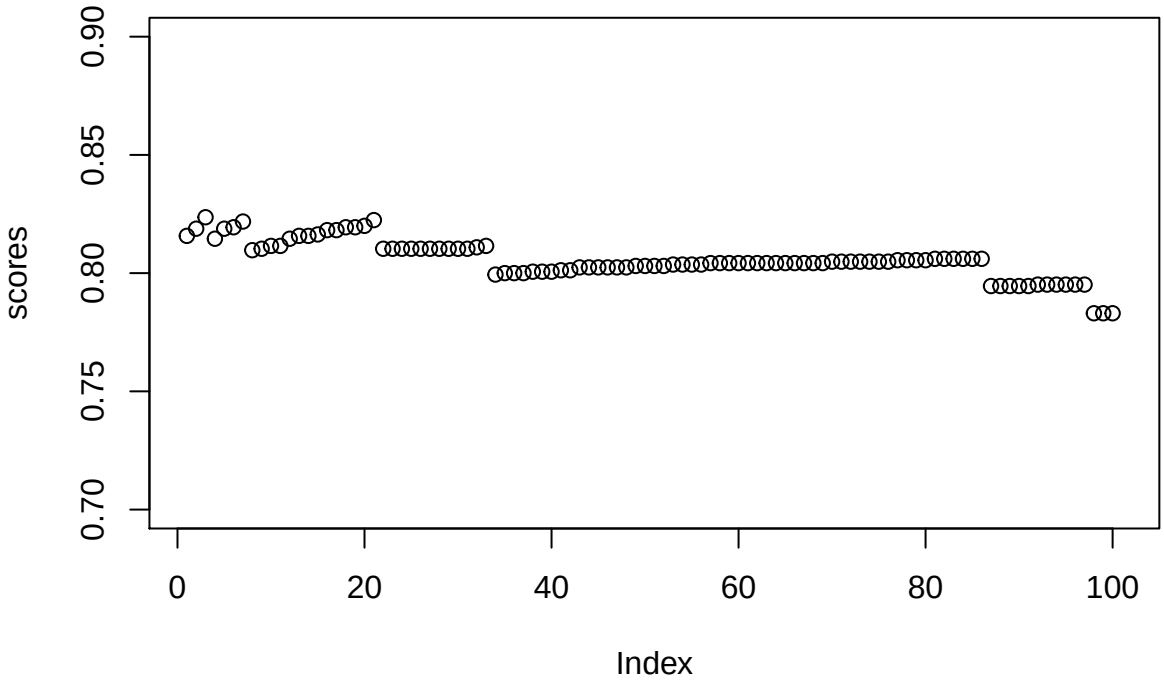

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



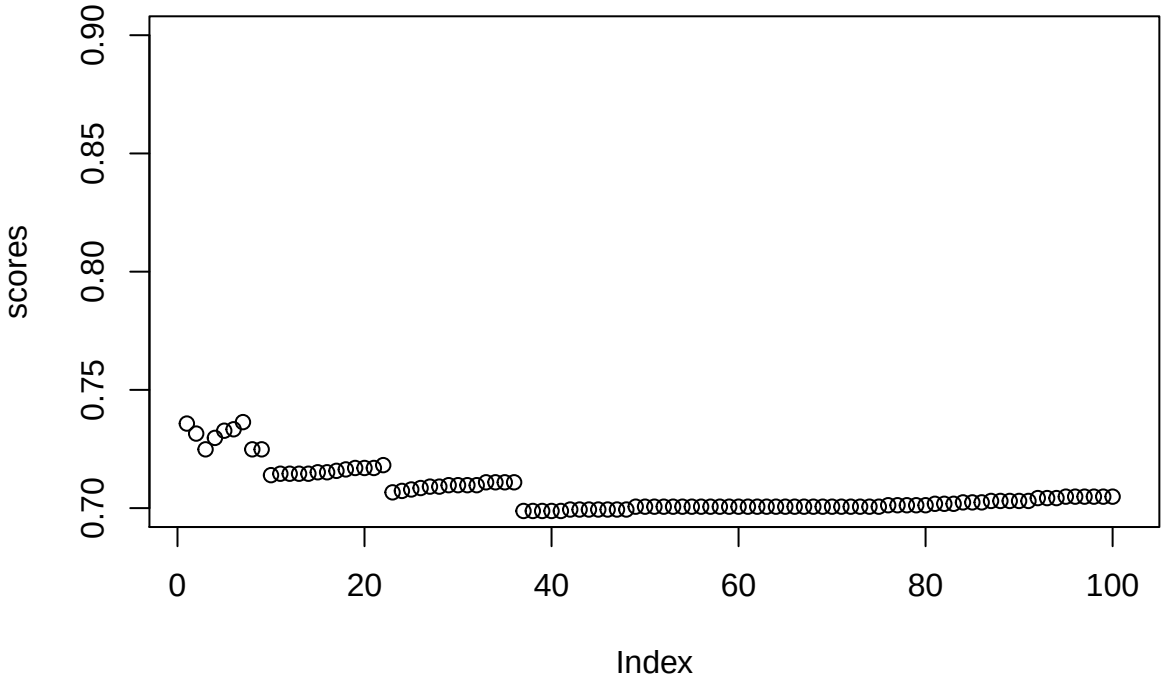

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



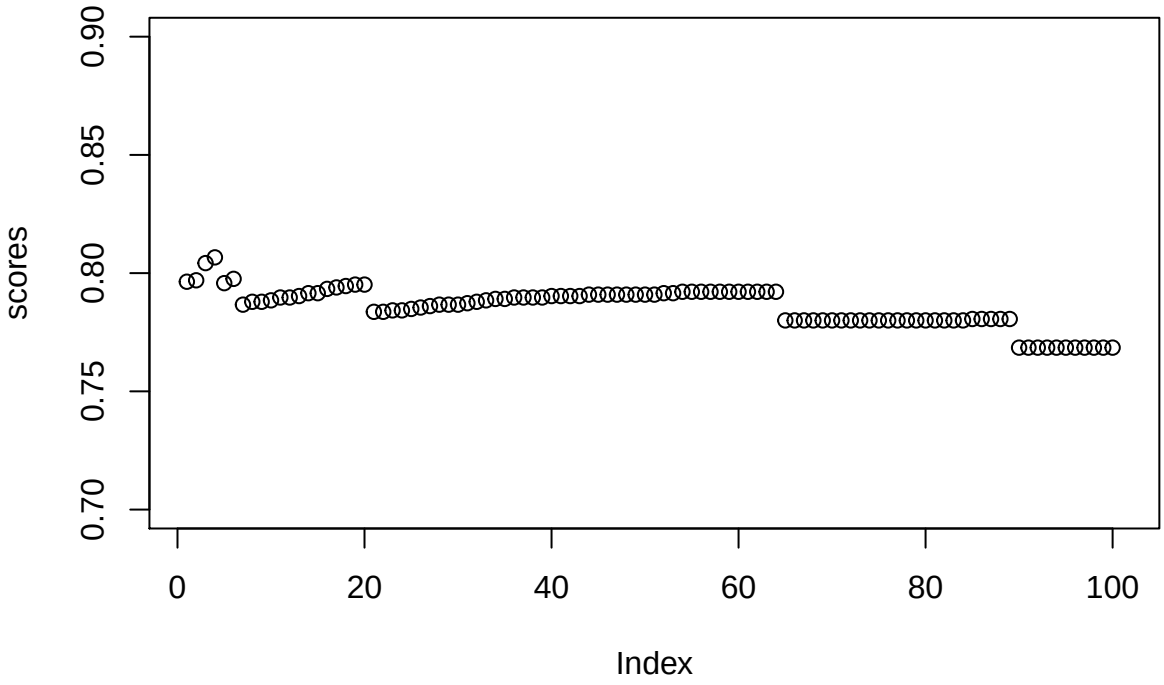

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



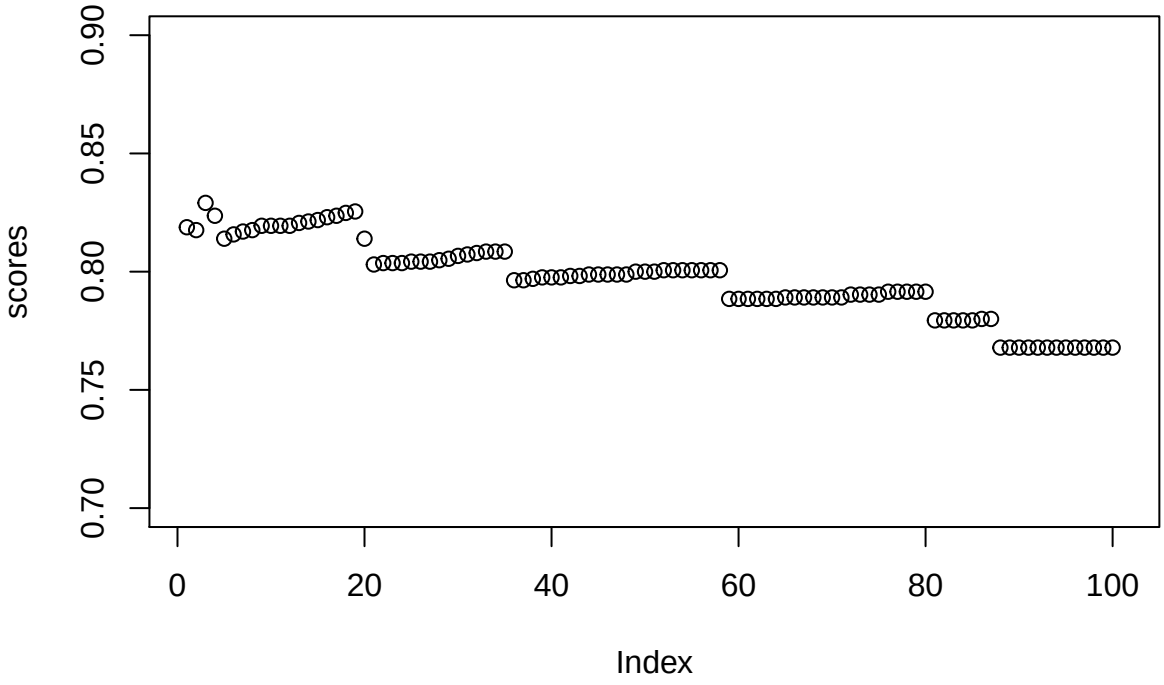

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```



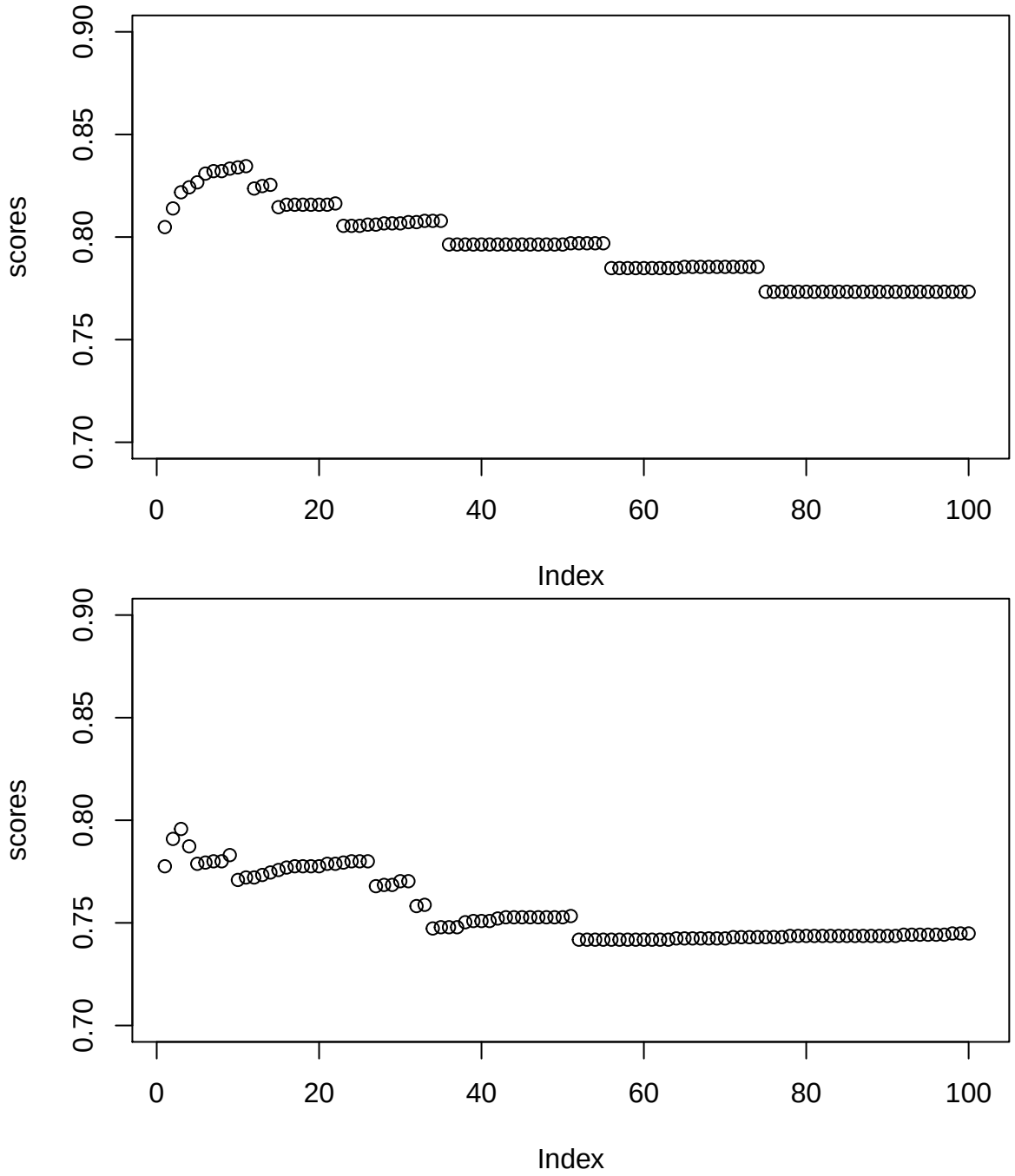

```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```




```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```




```
Warning in diag(conf_mat) * c(1, 20, 20): longer object length is not a multiple
of shorter object length
```




```
mean(as.numeric(unlist(thresh_list)))
```

```
[1] 0.009545455
```

```
median(as.numeric(unlist(thresh_list)))
```

```
[1] 0.006
```

```
set.seed(123)
  vars_test <- createDataPartition(vars$class, p = 0.8,
                                   list = FALSE)
  vars_train <- vars[vars_test, ]
  vars_test <- vars[-vars_test, ]
```

```
  train_cont <- trainControl(method = "cv", number = 5, classProbs = TRUE, savePredictions = TRUE)
  lda_ft <- train(class ~ ., data = vars_train, method = "lda", preProc = c("center", "scale"),
                  trControl = train_cont)
    preds <- predict(lda_ft, newdata = vars_test, type = "prob")
 lda_mod <- table("pred"=apply(preds, 1, classify, k = median(as.numeric(unlist(thresh_list)))), "obs" =
  lda_mod
```

```
    obs
pred  NG  OG TSG
   0 484   2   1
   1  31  23   3
   2  35   4  22
```

```
  score(lda_mod)
```

```
[1] 0.8387879
```

```
tests <- read.csv("test.csv")
preds <- predict(lda_ft, newdata = tests, type = "prob")
preds <- apply(preds, 1, classify, k = 0.01)
names(preds) <- tests$id
csv_file <- data.frame("id" = tests$id,
                       "class" = preds)
write.csv(csv_file, "modelpredictions9.csv", row.names = FALSE)
model_2_beta <- read.csv("modelpredictions2_beta.csv")
table("BEST MOD" = model_2_beta$class, csv_file$class)
```

```
BEST MOD    0    1    2
       0 1091   32   15
       1   10   63    7
       2   21    7  117
```