Technical Skills Assessment Author: Shiraz Bheda

```
###load input files into a dataframe
df <- read.csv("clinical.csv", stringsAsFactors = F)

###assign "ID" column as the row names vector, remove from dataframe
rownames(df) <- df$ID
df$ID <- NULL

###rename certain columns for clarity
df <- df %>%
       rename("Tumor.Stage" = "T",
              "Metastasis.To.Lymph.Nodes" = "N",
              "Distant.Metastases" = "M")
```

Pre-processing checklist

#1. Check that field separators are splitting metadata in clinical.csv input file as intended

```
###check dimensions of dataframe
dim(df)
```

```
## [1] 190  15
```

```
head(df)
```

```
##   Outcome Survival.Months Age Grade Num.Primaries Tumor.Stage
## 1   Alive               9  67     4             0         UNK
## 2    Dead              19  73     2             0         UNK
## 3    Dead              13  72     3             0           2
## 4    Dead              15  69     9             1          1a
## 5    Dead              10  76     9             0         UNK
## 6    Dead              11  62     9             0           3
##   Metastasis.To.Lymph.Nodes Distant.Metastases Radiation Stage     Primary.Site
## 1                         2               NULL         0    IV  Left Lower Lobe
## 2                         2                  0         5    IV Right Upper Lobe
## 3                         2                  0         0  IIIA Right Upper Lobe
## 4                         0                  1         0    IA Right Upper Lobe
## 5                      NULL               NULL         0  IIIA       Left Hilar
## 6                         2               NULL         0   IVB       Left Hilar
##                 Histology Tumor.Size Num.Mutated.Genes Num.Mutations
## 1 Squamous cell carcinoma        1.4                 8             8
## 2          Adenocarcinoma       NULL                 2             2
## 3          Adenocarcinoma        1.5                 1             1
## 4          Adenocarcinoma       NULL                 4             4
## 5    Large-cell carcinoma       NULL                 3             3
## 6          Adenocarcinoma       NULL                 4             5
```

```
tail(df)
```

```
##     Outcome Survival.Months Age Grade Num.Primaries Tumor.Stage
## 185   Alive              33  71     4             0         UNK
## 186    Dead              32  82     9             0           4
## 187    Dead              10  62     4             0           3
## 188    Dead              23  72     3             0           3
## 189    Dead              32  67     4             1          1a
## 190    Dead              33  71     9             0          2a
##     Metastasis.To.Lymph.Nodes Distant.Metastases Radiation Stage
## 185                         0                  0         0    IV
## 186                         0                  1         0  IIIB
## 187                         2               NULL         0   IVB
## 188                      NULL               NULL         0    IA
## 189                      NULL                  0         0    IV
## 190                      NULL                  0         0  IIIB
##         Primary.Site               Histology Tumor.Size Num.Mutated.Genes
## 185 Right Lower Lobe          Adenocarcinoma        1.5                 2
## 186 Right Upper Lobe Squamous cell carcinoma          9                 2
## 187       Left Hilar    Large-cell carcinoma       NULL                 3
## 188 Right Upper Lobe Squamous cell carcinoma          2                 3
## 189 Right Upper Lobe          Adenocarcinoma         10                 3
## 190 Right Upper Lobe          Adenocarcinoma       NULL                 2
##     Num.Mutations
## 185             2
## 186             2
## 187             5
## 188             3
## 189             3
## 190             2
```

2. Exploratory data analysis: Check for any misspellings, blank spaces, formatting inconsistencies

```
###Column 'Survival.Months' has 3 instances of floating-point numbers (9.5 months)
table(df$Survival.Months)
```

```
## 
##   9 9.5  10  11  13  15  16  18  19  22  23  24  26  29  32  33  34  35  36  37 
##   7   3  23  27  21   7   8   2   6   7   6   1   1   3  11   8   4   6  18   1 
##  38  39  40  41  42  46  50  71 
##   9   2   1   1   3   1   1   2
```

```
###Column 'Stage' has an incorrect value that needs to be changed (1B instead of IB)
table(df$Stage)
```

```
## 
##   1B   IA   IB  IIA  IIB IIIA IIIB   IV  IVB 
##    1   32    1    8   11   43   24   45   25
```

```
df$Stage[df$Stage == '1B'] <- 'IB'
table(df$Stage)
```

```
## 
##   IA   IB  IIA  IIB IIIA IIIB   IV  IVB 
##   32    2    8   11   43   24   45   25
```

```
###Column 'Primary.Site' has incorrect spellings ('Righ Upper Lobe')
table(df$Primary.Site)
```

```
## 
##         Both Lung        Left Hilar   Left Lower Lobe   Left Upper Lobe 
##                 5                31                17                21 
##   Righ Upper Lobe       Right Hilar  Right Lower Lobe Right Middle Lobe 
##                 2                33                25                 3 
##  Right Upper Lobe 
##                53
```

```
df$Primary.Site[df$Primary.Site == 'Righ Upper Lobe'] <- 'Right Upper Lobe'
table(df$Primary.Site)
```

```
##
##         Both Lung        Left Hilar   Left Lower Lobe   Left Upper Lobe
##                 5                31                17                21
##       Right Hilar  Right Lower Lobe Right Middle Lobe  Right Upper Lobe
##                33                25                 3                55
```

3. Sanity-checks: Make sure that information in clinical.csv file is logically consistent

```
###(Assuming that mutations & genes referred to are all in coding regions) all Values in
###'Num.Mutations' should not be lower than corresponding values in 'Num.Mutated.Genes'
###column
check1 <- df[df$Num.Mutations < df$Num.Mutated.Genes, ]
nrow(check1) ###No selected rows indicates logic is ok between two columns
```

```
## [1] 0
```

```
###Since the data comes from de-identified databases in order to fulfill PHI compliance,
###there is a possibility that the same patient is accidentally duplicated within a
###dataset. So we add a check here to ensure that is not the case.
df$key<-apply(df,1,paste,collapse="")
check2 <- table(table(df$key))
check2 #all concatenated values are unique
```

```
##
##   1
## 190
```

```
###Load genomics.csv file, check to make sure that the number of genes per sample agrees
###with data in clinical.csv file
df2 <- read.csv("genomics.csv", stringsAsFactors = F)
df3 <- as.data.frame(table(df2$ID))
rownames(df3) <- df3$Var1
df3$Var1 <- NULL
check3 <- merge(df, df3, by.x = 0, by.y = 0)

check3$Num.Mutated.Genes - check3$Freq #data in clinical.csv & genomics.csv files match
```

```
##   [1] 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 0 0 0 0 0 0 0
##  [38] 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 0 0 0 0 0 0 0
##  [75] 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 0 0 0 0 0 0 0
## [112] 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 0 0 0 0 0 0 0
## [149] 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 0 0 0 0 0 0
```

Feature selection: Drop columns with greater than 20% missing data, check features for high correlations

```
###drop columns with missing data greater than 20%
df_dropped_features_with_missing_data <- df[,!names(df) %in%
      c("Grade", "Tumor.Stage", "Metastasis.To.Lymph.Nodes", "Distant.Metastases",
        "Tumor.Size")]

###checking pearson correlation between Num.Mutated.Genes & Num.Mutations
check4 <- df_dropped_features_with_missing_data[, c(9, 10)]
pcor(check4, method = "pearson") #91% correlation indicates redundancy
```

```
## $estimate
##                     Num.Mutated.Genes Num.Mutations
## Num.Mutated.Genes           1.0000000     0.9157352
## Num.Mutations               0.9157352     1.0000000
##
## $p.value
##                     Num.Mutated.Genes Num.Mutations
## Num.Mutated.Genes         0.00000e+00   2.25099e-76
## Num.Mutations             2.25099e-76   0.00000e+00
##
## $statistic
##                     Num.Mutated.Genes Num.Mutations
## Num.Mutated.Genes             0.00000      31.25056
## Num.Mutations                31.25056       0.00000
##
## $n
## [1] 190
##
## $gp
## [1] 0
##
## $method
## [1] "pearson"
```

```
###drop 'Num.Genes' and get rid of key
df_dropped_features_with_missing_and_redundant_data <-
df_dropped_features_with_missing_data[,!names(
  df_dropped_features_with_missing_data) %in%
      c("Num.Mutated.Genes", "key")]
```

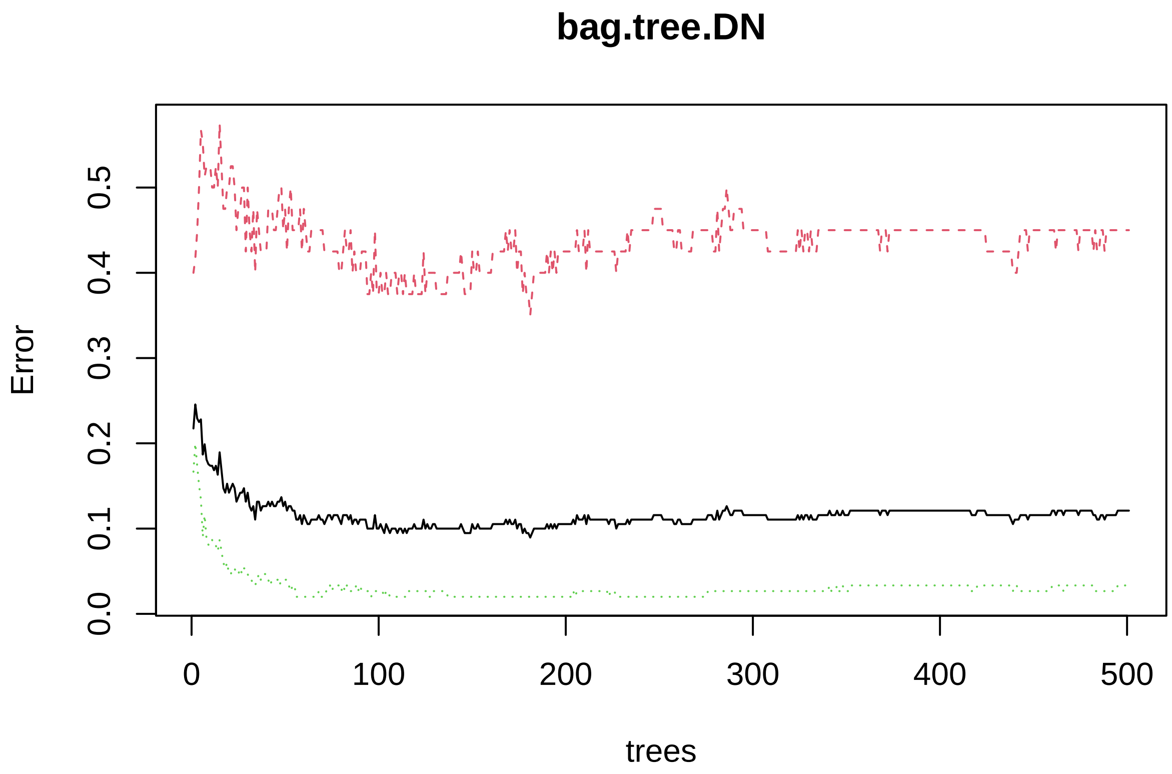
```
###Selection of inputs for the RF model
inputs <- df_dropped_features_with_missing_and_redundant_data

inputs <- as.data.frame(inputs)
inputs$Outcome <- as.character(inputs$Outcome)
inputs$Outcome <- as.factor(inputs$Outcome)
inputs$Age <- as.numeric(inputs$Age)
inputs$Num.Primaries <- as.numeric(inputs$Num.Primaries)
inputs$Radiation <- as.numeric(inputs$Radiation)
inputs$Stage <- as.factor(inputs$Stage)
inputs$Primary.Site <- as.factor(inputs$Primary.Site)
inputs$Histology <- as.factor(inputs$Histology)
inputs$Num.Mutations <- as.factor(inputs$Num.Mutations)

#####USING RF MODEL#####

#use bagging to avoid losing data while training model
set.seed(12345)
bag.tree.DN = randomForest(Outcome~., data = inputs, mtry = 3, ntree = 501,
                           importance = TRUE, na.action=na.roughfix, keep.inbag = TRUE,
                           norm.votes = FALSE, replace = FALSE)


###generate a visualization of per-category outcome error rate, as well as the OOB error
###rate, and how they change as the number of decision trees are increased
plot(bag.tree.DN, legend=TRUE)
```



```
###generate summary statistics of model
summary(bag.tree.DN)
```

```
##                 Length Class  Mode
## call               10  -none- call
## type                1  -none- character
## predicted         190  factor numeric
## err.rate         1503  -none- numeric
## confusion           6  -none- numeric
## votes             380  matrix numeric
## oob.times         190  -none- numeric
## classes             2  -none- character
## importance         32  -none- numeric
## importanceSD       24  -none- numeric
## localImportance     0  -none- NULL
## proximity           0  -none- NULL
## ntree               1  -none- numeric
## mtry                1  -none- numeric
## forest             14  -none- list
## y                 190  factor numeric
## test                0  -none- NULL
## inbag           95190  -none- numeric
## terms               3  terms  call
```

```
###generate confusion matrix to evaluate final error rates
bag.tree.DN$confusion
```

```
##       Alive Dead class.error
## Alive    22   18  0.45000000
## Dead      5  145  0.03333333
```

```
###save individual number of decision tree votes as a separate dataframe
bag.tree.DN_df <- as.data.frame(bag.tree.DN$votes)
dim(bag.tree.DN_df)
```

```
## [1] 190   2
```

```
bag.tree.DN_df
```

```
##     Alive Dead
## 1     150   37
## 2      10  174
## 3      19  162
## 4      42  122
## 5       0  180
## 6       0  203
## 7       2  169
## 8       1  181
## 9      18  138
## 10    166   28
## 11      4  167
## 12      0  197
## 13     42  136
## 14    135   59
## 15      0  176
## 16      0  196
## 17      0  165
## 18    106   88
## 19      9  167
## 20      0  175
## 21      1  201
## 22     61  121
## 23     24  160
## 24     21  152
## 25     23  147
## 26      0  175
## 27     29  159
## 28    183    8
## 29      3  166
## 30     13  164
## 31     15  172
## 32     47  136
## 33     44  142
## 34     84   93
## 35      8  176
## 36      5  176
## 37     99   97
## 38      0  184
## 39      0  186
## 40      32  159
## 41      10  172
## 42      83  114
## 43       8  164
## 44       0  169
## 45      36  160
## 46       0  180
## 47     167   13
## 48      24  168
## 49       0  212
## 50     101   88
## 51       6  179
## 52       0  188
## 53     117   57
## 54      33  145
## 55       0  186
## 56      46  114
## 57       0  176
## 58      22  156
## 59      48  148
## 60      16  176
## 61      93  101
## 62      32  158
## 63       0  170
## 64      52  133
## 65       0  169
## 66      42  137
## 67     130   60
## 68      14  165
## 69       0  186
## 70      25  167
## 71      14  169
## 72       0  181
## 73       0  176
## 74       6  161
## 75      10  169
## 76      54  127
## 77       0  178
## 78      54  128
## 79       5  162
## 80       6  190
## 81     125   50
## 82      90  109
## 83       0  172
## 84      15  179
## 85       4  185
## 86      75  102
## 87       0  183
## 88      65  134
## 89       7  177
## 90      49  136
## 91      92   81
## 92      12  164
## 93     157   22
## 94     100  101
## 95      40  147
## 96      54  129
## 97      80   95
## 98      60  129
## 99       3  175
## 100      0  178
## 101     27  158
## 102     49  148
## 103      0  178
## 104     53  150
## 105      5  158
## 106    100   96
## 107      0  170
## 108      0  179
## 109    155   37
## 110     90   86
## 111      0  192
## 112    103   77
## 113    101   69
## 114     48  140
## 115     29  154
## 116     11  185
## 117      0  192
## 118     19  161
## 119     90   91
## 120      0  171
## 121     83   88
## 122    154   30
## 123      0  195
## 124     40  143
## 125     13  160
## 126      0  195
## 127      0  172
## 128     93   92
## 129      3  165
## 130     89   92
## 131     24  163
## 132     95   79
## 133    102   77
## 134      0  181
## 135     20  161
## 136    110   91
## 137     30  164
## 138     22  150
## 139     33  150
## 140      0  170
## 141     52  129
## 142     47  128
## 143     14  159
## 144      0  188
## 145     14  162
## 146     38  159
## 147    105   81
## 148     4  159
## 149     0  203
## 150     1  191
## 151    11  179
## 152    42  135
## 153    26  161
## 154    35  152
## 155     0  168
## 156     0  182
## 157    12  170
## 158    30  143
## 159     8  178
## 160    22  162
## 161    29  154
## 162     0  179
## 163     0  171
## 164     0  171
## 165     0  192
## 166    11  161
## 167     0  151
## 168     0  202
## 169   161    5
## 170    14  165
## 171    18  153
## 172    40  152
## 173   160   17
## 174     0  181
## 175    96   89
## 176    86   96
## 177     0  191
## 178    78   93
## 179    55  124
## 180     0  182
## 181     6  151
## 182     0  179
## 183    79  104
## 184    42  126
## 185     3  170
## 186    58  124
## 187     0  179
## 188    57  107
## 189     9  189
## 190    16  170
```

```
###export to csv format
write.csv(bag.tree.DN_df,
          "votes.csv")
print(row.names(bag.tree.DN_df))
```

```
##   [1] "1"   "2"   "3"   "4"   "5"   "6"   "7"   "8"   "9"   "10"  "11"  "12"
##  [13] "13"  "14"  "15"  "16"  "17"  "18"  "19"  "20"  "21"  "22"  "23"  "24"
##  [25] "25"  "26"  "27"  "28"  "29"  "30"  "31"  "32"  "33"  "34"  "35"  "36"
##  [37] "37"  "38"  "39"  "40"  "41"  "42"  "43"  "44"  "45"  "46"  "47"  "48"
##  [49] "49"  "50"  "51"  "52"  "53"  "54"  "55"  "56"  "57"  "58"  "59"  "60"
##  [61] "61"  "62"  "63"  "64"  "65"  "66"  "67"  "68"  "69"  "70"  "71"  "72"
##  [73] "73"  "74"  "75"  "76"  "77"  "78"  "79"  "80"  "81"  "82"  "83"  "84" 
##  [85] "85"  "86"  "87"  "88"  "89"  "90"  "91"  "92"  "93"  "94"  "95"  "96" 
##  [97] "97"  "98"  "99"  "100" "101" "102" "103" "104" "105" "106" "107" "108"
## [109] "109" "110" "111" "112" "113" "114" "115" "116" "117" "118" "119" "120"
## [121] "121" "122" "123" "124" "125" "126" "127" "128" "129" "130" "131" "132"
## [133] "133" "134" "135" "136" "137" "138" "139" "140" "141" "142" "143" "144"
## [145] "145" "146" "147" "148" "149" "150" "151" "152" "153" "154" "155" "156"
## [157] "157" "158" "159" "160" "161" "162" "163" "164" "165" "166" "167" "168"
## [169] "169" "170" "171" "172" "173" "174" "175" "176" "177" "178" "179" "180"
## [181] "181" "182" "183" "184" "185" "186" "187" "188" "189" "190"
```