Final Project Part 1: Design of the Experiment

Qianli Wu

28 May, 2022

Question 1. Propose a fractional factorial design for the problem. In addition, propose an experimental design constructed using the optimal design approach.

The goal of this project is to identify the tuning parameters of random forest that affect the cross-validation accuracy.

- We have seven tuning parameters:

	Parameter	Low Level	High Level
A	ntree	100	1000
B	replace	0	1
C	mtry	2	6
D	nodesize	1	11
E	maxnodes	10	1000
F	classwt	0.5	0.9
G	cutoff	0.2	0.8

Fractional Factorial Design

Since there are 7 factors each with 2 levels, a complete factorial design 2^7 requires 128 runs. Only 7 degrees of freedom correspond to main effects, 21 correspond to two-factor interactions, and the rest of 99 are associated with three-factor and higher interactions.

Due to limitation of resources, we can only run at most 35 tests. We assume that certain high-order interactions are negligible.

Since

- The effect sparsity principle
- The projection property
- Sequential experimentation

We choose the One-Quarter Fraction of the 2^7 Design, i.e., 2^{7-2} Design.

Letting the 'FrF2' function recommend one design. Generally, the recommended designs are those in Table 8.14 in Chapter 8. The designs recommended by the function have the largest possible resolution. They also have the smallest aberration as defined in Section 8.4.1. Fractional factorial designs with minimum aberration are preferred because they minimize the aliasing among the factors' effects.

```
factors <- list(ntree = c(100, 1000),
                replace = c(0, 1),
                mtry = c(2, 6),
                nodesize = c(1,11),
                maxnodes = c(10, 1000),
                classwt = c(0.5, 0.9),
                cutoff = c(0.2, 0.8))
my.design <- FrF2(nruns = 32, nfactors = 7, randomize = F, factor.names = factors)
print(my.design)
```

```
##    ntree replace mtry nodesize maxnodes classwt cutoff
## 1    100       0    2        1       10     0.5    0.8
## 2   1000       0    2        1       10     0.9    0.2
## 3    100       1    2        1       10     0.9    0.2
## 4   1000       1    2        1       10     0.5    0.8
## 5    100       0    6        1       10     0.9    0.8
## 6   1000       0    6        1       10     0.5    0.2
## 7    100       1    6        1       10     0.5    0.2
## 8   1000       1    6        1       10     0.9    0.8
## 9    100       0    2       11       10     0.5    0.2
## 10  1000       0    2       11       10     0.9    0.8
## 11   100       1    2       11       10     0.9    0.8
## 12  1000       1    2       11       10     0.5    0.2
## 13   100       0    6       11       10     0.9    0.2
## 14  1000       0    6       11       10     0.5    0.8
## 15   100       1    6       11       10     0.5    0.8
## 16  1000       1    6       11       10     0.9    0.2
## 17   100       0    2        1     1000     0.5    0.2
## 18  1000       0    2        1     1000     0.9    0.8
## 19   100       1    2        1     1000     0.9    0.8
## 20  1000       1    2        1     1000     0.5    0.2
## 21   100       0    6        1     1000     0.9    0.2
## 22  1000       0    6        1     1000     0.5    0.8
## 23   100       1    6        1     1000     0.5    0.8
## 24  1000       1    6        1     1000     0.9    0.2
## 25   100       0    2       11     1000     0.5    0.8
## 26  1000       0    2       11     1000     0.9    0.2
## 27   100       1    2       11     1000     0.9    0.2
## 28  1000       1    2       11     1000     0.5    0.8
## 29   100       0    6       11     1000     0.9    0.8
## 30  1000       0    6       11     1000     0.5    0.2
## 31   100       1    6       11     1000     0.5    0.2
## 32  1000       1    6       11     1000     0.9    0.8
## class=design, type= FrF2
```

```
design.info(my.design)$catlg.entry
```

```
## Design:  7-2.1
##    32  runs,  7  factors,
##    Resolution  IV
##    Generating columns:  7 27
##    WLP (3plus):  0 1 2 0 0 ,  15  clear 2fis
##  Factors with all 2fis clear:  D E G
```

```
design.info(my.design)$aliased
```

```
## $legend
## [1] "A=ntree"    "B=replace"  "C=mtry"     "D=nodesize" "E=maxnodes"
## [6] "F=classwt"  "G=cutoff"
##
## $main
## character(0)
##
## $fi2
## [1] "AB=CF" "AC=BF" "AF=BC"
```

An Experimental Design Constructed using the Optimal Design Approach.

Suggestion is to start with 32 runs because multiples of 4 allows:

- orthogonal design with diagonal variance-covariance matrix.
- independent estimates of main effects, assuming interactions are ignored. (doubtful, since the upper left-corner of the 2.2 color map in Q2 is not 100% white)

3 runs are saved for confirmation experiments or follow-up experiment.

```
# Create a full design for 7 factors each with 2 levels
full.design <- FrF2(nruns = 128, nfactors = 7, randomize = F, factor.names = factors)
```

```
## creating full factorial with 128 runs ...
```

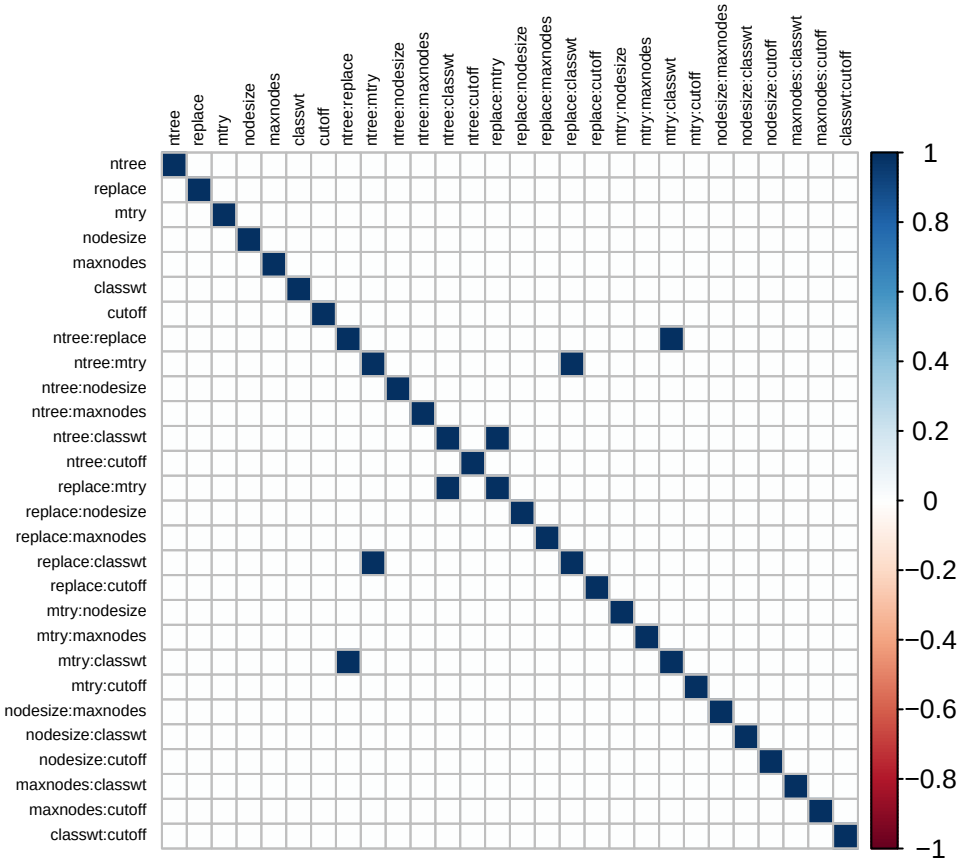
```
# Consider a model with 35 runs
alternative.design <- optFederov(~.^2,
                                 full.design, nTrials = 32, nRepeats = 1000)
print.data.frame(alternative.design$design)
```

```
##     ntree replace mtry nodesize maxnodes classwt cutoff
## 1     100       0    2        1       10     0.5    0.2
## 4    1000       1    2        1       10     0.5    0.2
## 6    1000       0    6        1       10     0.5    0.2
## 10   1000       0    2       11       10     0.5    0.2
## 15    100       1    6       11       10     0.5    0.2
## 18   1000       0    2        1     1000     0.5    0.2
## 27    100       1    2       11     1000     0.5    0.2
## 29    100       0    6       11     1000     0.5    0.2
## 34   1000       0    2        1       10     0.9    0.2
## 43    100       1    2       11       10     0.9    0.2
## 45    100       0    6       11       10     0.9    0.2
## 48   1000       1    6       11       10     0.9    0.2
## 49    100       0    2        1     1000     0.9    0.2
## 55    100       1    6        1     1000     0.9    0.2
## 60   1000       1    2       11     1000     0.9    0.2
## 62   1000       0    6       11     1000     0.9    0.2
## 66   1000       0    2        1       10     0.5    0.8
## 71    100       1    6        1       10     0.5    0.8
## 73    100       0    2       11       10     0.5    0.8
## 76   1000       1    2       11       10     0.5    0.8
## 78   1000       0    6       11       10     0.5    0.8
## 83    100       1    2        1     1000     0.5    0.8
## 85    100       0    6        1     1000     0.5    0.8
## 88   1000       1    6        1     1000     0.5    0.8
## 90   1000       0    2       11     1000     0.5    0.8
## 97    100       0    2        1       10     0.9    0.8
## 104  1000       1    6        1       10     0.9    0.8
## 106  1000       0    2       11       10     0.9    0.8
## 116  1000       1    2        1     1000     0.9    0.8
## 118  1000       0    6        1     1000     0.9    0.8
## 121   100       0    2       11     1000     0.9    0.8
## 127   100       1    6       11     1000     0.9    0.8
```

Question 2. Compare the optimal design with the fractional factorial design in practical and statistical terms. For instance, what is the performance of the designs for studying the main effects of the tuning parameters only? Can they estimate all two-parameter interactions? Why or why not? How do they compare in terms of multicollinearity?

1.1 Color Map for Fractional Factorial Design

```
# Visualize the aliasing in the design.
D.alt <- (desnum(my.design)) # Extract the design.
# Create the model matrix including main effects and two-factor interactions.
X.alt <- model.matrix(~.^2-1, data.frame(D.alt))

# Create color map on pairwise correlations.
contrast.vectors.correlations.alt <- cor(X.alt)
corrplot(contrast.vectors.correlations.alt, type = "full", addgrid.col = "gray",
         tl.col = "black", tl.srt = 90, method = "color", tl.cex=0.5)
```



From the color map above, we get there's no alias among main effects of the tuning parameters, which means the performance of our design for studying the main effects of the tuning parameters only will be great.

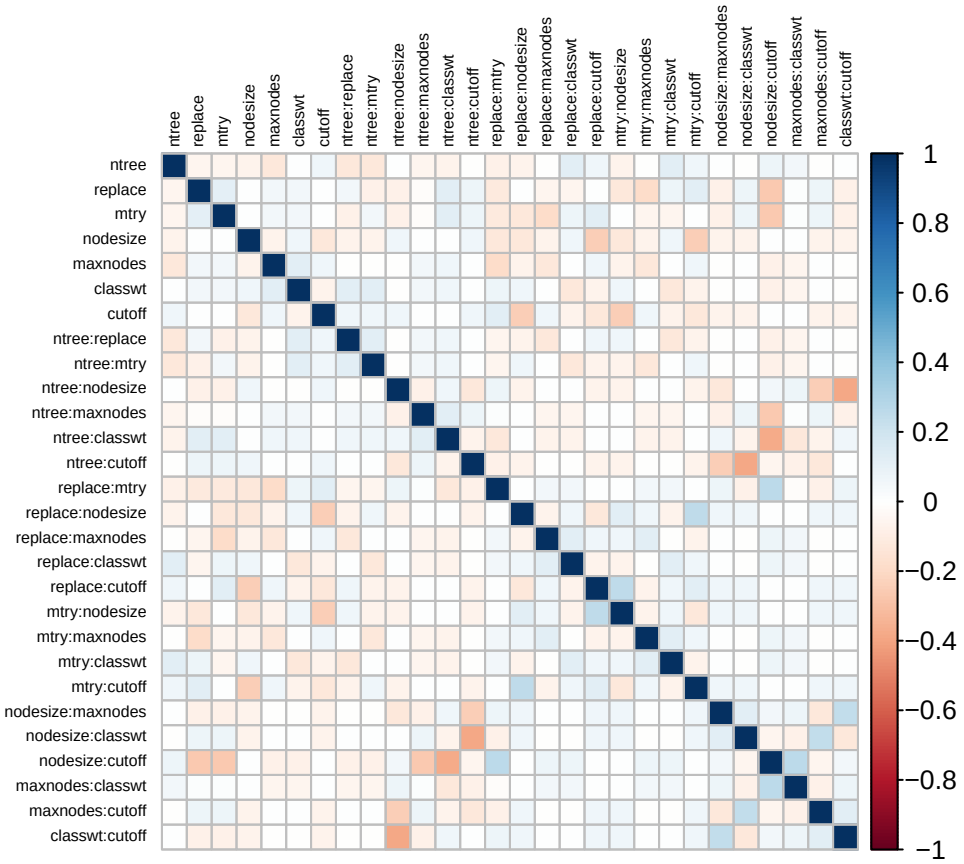
From the results of `design.info` in Question 1, we have AB=CF, AC=BF, and AF=BC, where "A=ntree", "B=replace", "C=mtry", "D=nodesize", "E=maxnodes", "F=classwt", and "G=cutoff". The aliasing is

also shown on the map above, as the corresponding dark cells for `mtry:classwt` with `ntree:replace`, `replace:classwt` with `ntree:mtry`, and `replace:mtry` with `ntree:classwt`.

Therefore, we have the fractional factorial design's performance is best for all main effects and two-factor interactions except these aliasing factors.

2.1 Color Map for D-optimal Design

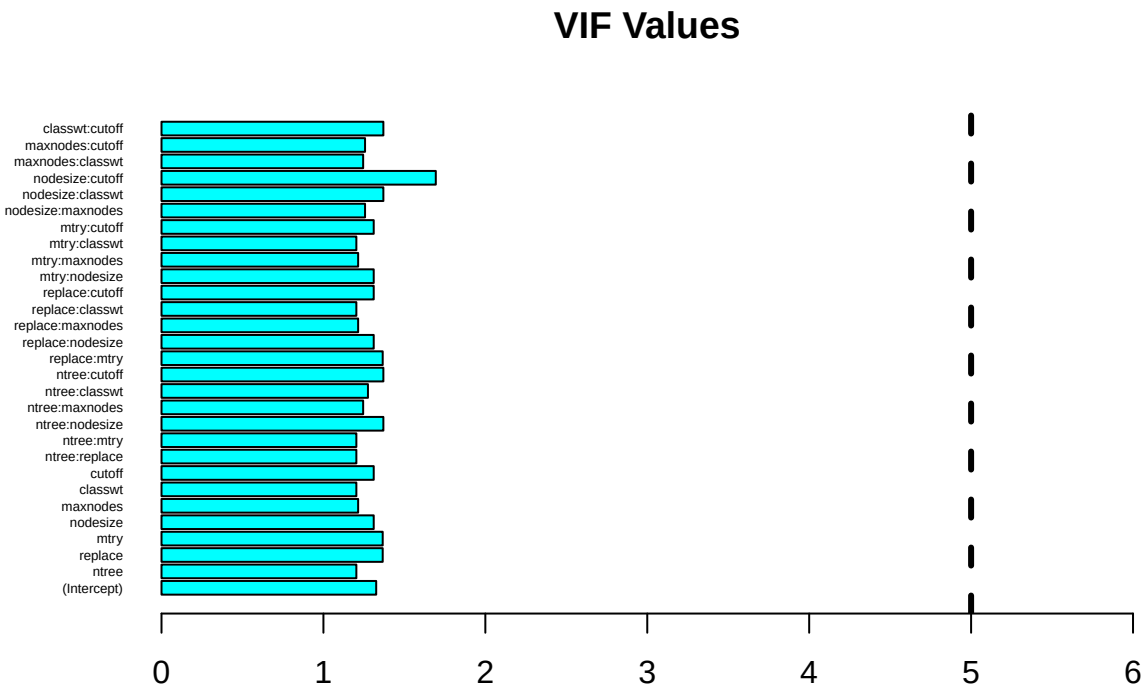
```
# Visualize the aliasing in the design.
D.alt <- data.frame(alternative.design$design) # Extract the design.
# Convert factors to -1 and 1
D.alt <- sapply(D.alt,function(x)(as.integer(x) -1.5)*2 )
X.alt <- model.matrix(~.^2-1, data.frame(D.alt))

# Create color map on pairwise correlations.
contrast.vectors.correlations.alt <- cor(X.alt)
corrplot(contrast.vectors.correlations.alt, type = "full", addgrid.col = "gray",
         tl.col = "black", tl.srt = 90, method = "color", tl.cex=0.5)
```



From the D-optimal Design above, we have most cells with light colors. Therefore, we have the alias among the main effects,between the main effects and the specific interactions, and among the specific interactions are fairly small.

2.2 VIF for Fractional Factorial Design

```
# We need to include the intercept when computing the VIF.
X.alt <- model.matrix(~.^2, data.frame(D.alt))
# Variance-covariance matrix of 46-run design. Assuming sigma^2 = 1
var.eff.one <- diag(solve(t(X.alt)%*%X.alt))
# Set the left margin of plot be 1
par(oma=c(0,1,0,0))
#create horizontal bar chart to display each VIF value
barplot(nrow(X.alt)*var.eff.one, main = "VIF Values", horiz = TRUE, col = "cyan", las=1, cex.names=0.4,
#add vertical line at 5
abline(v = 5, lwd = 3, lty = 2)
```



From the VIF plot above, we have there's no factor has VIF greater than 5. Thus, we don't have a severe multi-collineairty or aliasing issue.

3. Compare

- We have the fractional factorial design only suffers from multi-collinearity problem of these three pairs of aliased interactions. It will be great design if `mtry:classwt`, `ntree:replace`, `replace:classwt`, `ntree:mtry`, `replace:mtry`, and `ntree:classwt` are not potentially important effects.
- However, there's no serious multi-collinearity problem or aliasing issue for the D-optimal design. But the variances of more coefficient estimates are inflated trivially due to collinearity.

Question 3. Recommend one experimental design between the two options in Question 1. Motivate your decision.

Question 4. Using a commercial software, the TAs and I came up with the experimental design shown in Table 2. How does your recommended design in the previous question compare with this one?

```
source("CrossValidation_RandomForest/CrossValidation_RF.R")
```