1. *What models or machines did you attempt to fit?*

A.

```
# LS
fit.ls = lm(Y ~ ., data = data.train)
```

B.

```
#Hybrid Stepwise

fit.start = lm(Y ~ 1, data = data.train)
fit.end = lm(Y ~ ., data = data.train)

step.BIC = step(fit.start, list(upper = fit.end), k = log(n.train),
                trace = 0)
```

C.

```
#ridge regression

fit.ridge = lm.ridge(Y ~ ., lambda = lambda.vals,
                     data = data.train)
```

D.

```
### LASSO
all.LASSOs = cv.glmnet(x = matrix.train, y = Y.train)

### Get both 'best' lambda values using $lambda.min and $lambda.1se
lambda.min = all.LASSOs$lambda.min
lambda.1se = all.LASSOs$lambda.1se

### Get the coefficients for our two 'best' LASSO models
coef.LASSO.min = predict(all.LASSOs, s = lambda.min, type = "coef")
coef.LASSO.1se = predict(all.LASSOs, s = lambda.1se, type = "coef")
```

E.

```
## GAM
fit.gam = gam(Y ~ s(X1, k=5) + s(X2, k=5) + s(X3, k=5) + s(X4, k=5) + s(X5, k=5) + s(X6, k=5) + s(X7, k=5) + s(X8, k=5)
              + s(X9, k=5) + s(X10, k=5) + s(X11, k=5) + s(X12, k=5) + s(X13, k=5) + s(X14, k=5) +s(X15, k=5),
              data = data.train)
```

F.

```
# Full-tree
fit.tree = rpart(Y ~ ., data = data.train, cp=0)
```

G.

```
# Min-cv tree

fit.tree.min = prune(fit.tree, cp = CP.min)
```

H.

```
#"1-se tree"

fit.tree.1se = prune(fit.tree, cp = CP.1se)
```

I.

```
# Random-forest

fit.rf.2 = randomForest(Y ~ ., data = data.train, importance = T,
                        mtry = 9, nodesize = 8)
```

2. *What process(es) did you use to evaluate and compare models and to select your final model?* I am thinking of Lecture 3, specifically:

   I used 10-kfolds, fit all models to each training set. After that, I found the model with smallest MSPE.

3. *Did you tune any methods?* If so, (a) what process(es) did you use to evaluate and compare models and to select your final model (b) **for each method list all parameter values that were considered**

   a. For “ridge regression” to choose a lambda values, I use a sequence from 0 to 100 by 0.05.

   b. For “GAM” to choose a k value, I used for loop from 1 to 10 but when k=6, an error occurred.

   c. For “random-forest”, I used a grid of values mtry = (1… 13) and nodesize = (2, 5, 8, 11, 15)

4. *What was your chosen prediction machine?*

```
set.seed(2385660, kind="Mersenne-Twister")

#Read data
data = read.csv("Data2020.csv")

data2 = read.csv("Data2020testX.csv")

fit.gam2 = gam(Y ~ s(X1, k=5) + s(X2, k=5) + s(X3, k=5) + s(X4, k=5) + s(X5, k=5) + s(X6, k=5) + s(X7, k=5) +
                 s(X8, k=5) + s(X9, k=5) + s(X10, k=5) + s(X11, k=5) + s(X12, k=5) + s(X13, k=5) + s(X14, k=5) +s(X15, k=5),
               data = data)

pred.gam2 = predict(fit.gam2, data2)

write.table(pred.gam2, "prediction1.csv", sep = ",", row.names = F, col.names = F)
```