BHao_Final

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
kaggle = read.csv("C:/Users/bhao/Google Drive/CUNY/git/DATA605/Final/train.csv")
#str(kaggle)
```

Probability

Since P(X|Y) != P(X)P(Y), splitting the data this way does not make them independent. This is also confirmed by the chi-square test, which resulted in a very small p-value.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
df = kaggle %>% dplyr::select(SalePrice, LotArea)
X = df$LotArea
Y = df$SalePrice

# set x = min of 4th quartile and y = min of 2nd quartile
x = quantile(df$LotArea, 0.75)
y = quantile(df$SalePrice, 0.25)

#a. P(X>x | Y>y) = 31.23% probability that lot area is greater than 0.75 quantile when
#sale price is greater than 0.25 quantile
df %>% filter(LotArea > x & SalePrice > y) %>% summarise(n = n()) /
df %>% filter(SalePrice > y) %>% summarise(n = n())
```

```
##           n
```

```
## 1 0.3123288
```

```
#b. P(X>x, Y>y) = 23.42% probability that lot area is greater than 0.75 quantile AND
#sale price is greater than 0.25 quantile
df %>% filter(LotArea > x & SalePrice > y) %>% summarise(n = n()) /
nrow(df)
```

```
##           n
## 1 0.2342466
```

```
#c. P(X<x | Y>y) = 68.77% probability that lot area is less than 0.75 quantile when
#sale price is greater than 0.25 quantile
df %>% filter(LotArea < x & SalePrice > y) %>% summarise(n = n()) /
df %>% filter(SalePrice > y) %>% summarise(n = n())
```

```
##           n
## 1 0.6876712
```

```
#chi-square test
tbl = table(df %>% mutate(topQtrX = if_else(LotArea > x, 'TopQtr', 'Bottom3Qtr'),
              botQtrY = if_else(SalePrice > y, 'Top3Qtr', 'BottomQtr')) %>%
  dplyr::select(topQtrX, botQtrY))

chisq.test(tbl)
```

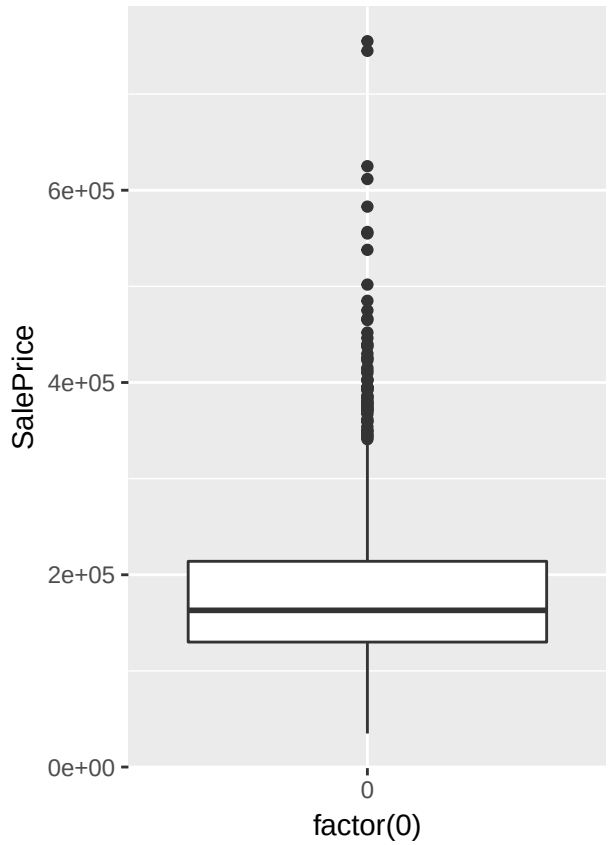
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tbl
## X-squared = 89.426, df = 1, p-value < 2.2e-16
```

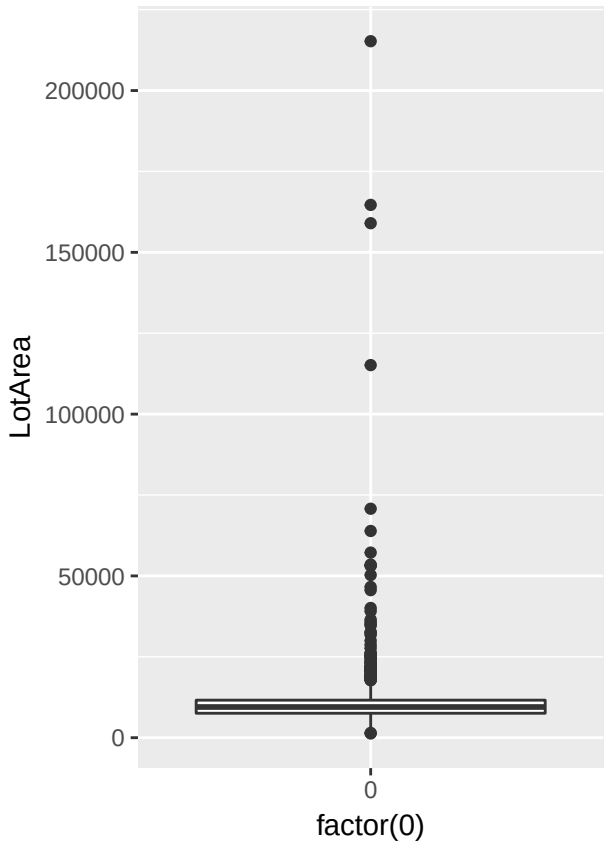
Descriptive and Inferential Statistics

```
summary(df)
```

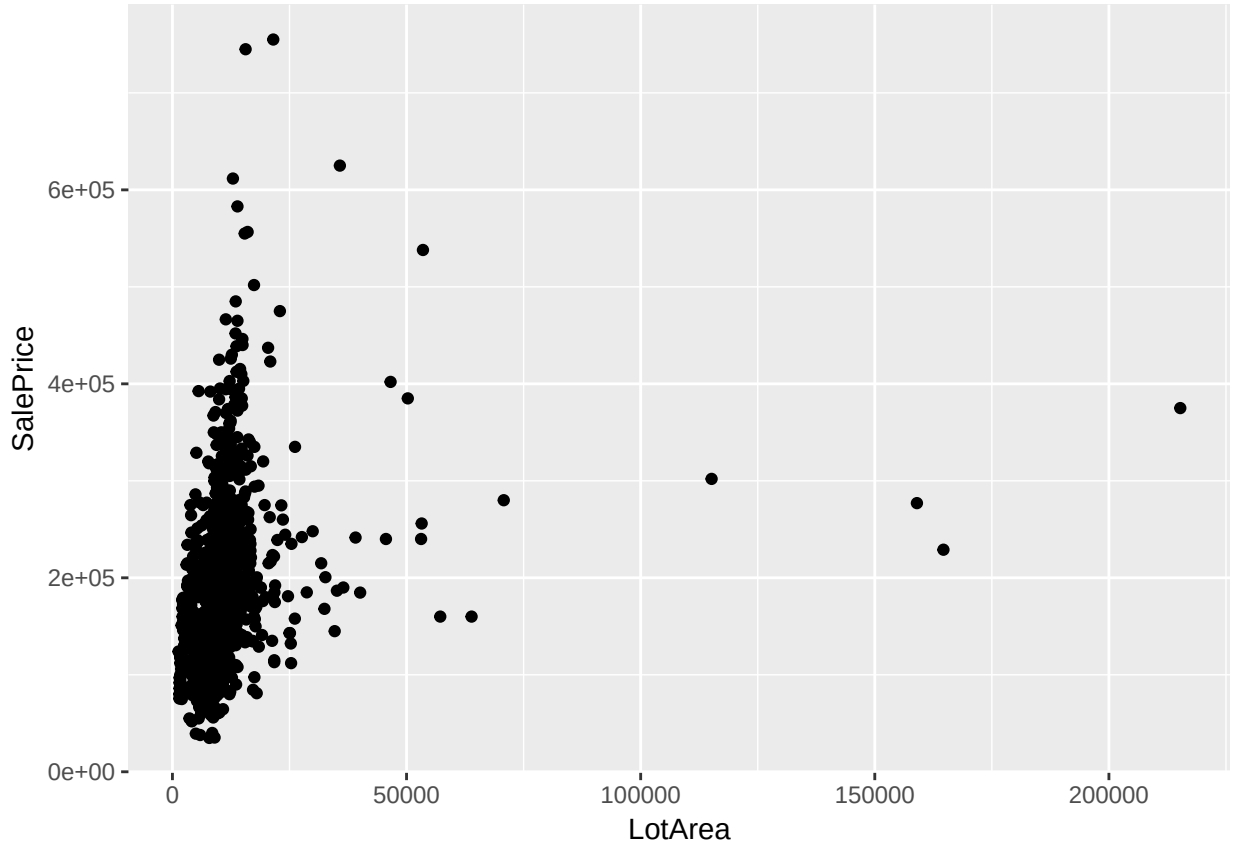
```
##    SalePrice         LotArea
##  Min.   : 34900   Min.   :  1300
##  1st Qu.:129975   1st Qu.:  7554
##  Median :163000   Median :  9478
##  Mean   :180921   Mean   : 10517
##  3rd Qu.:214000   3rd Qu.: 11602
##  Max.   :755000   Max.   :215245
```

```
p1 = df %>% ggplot(aes(x = factor(0), y = SalePrice)) + geom_boxplot()
p2 = df %>% ggplot(aes(x = factor(0), y = LotArea)) + geom_boxplot()
grid.arrange(p1, p2, ncol = 2)
```

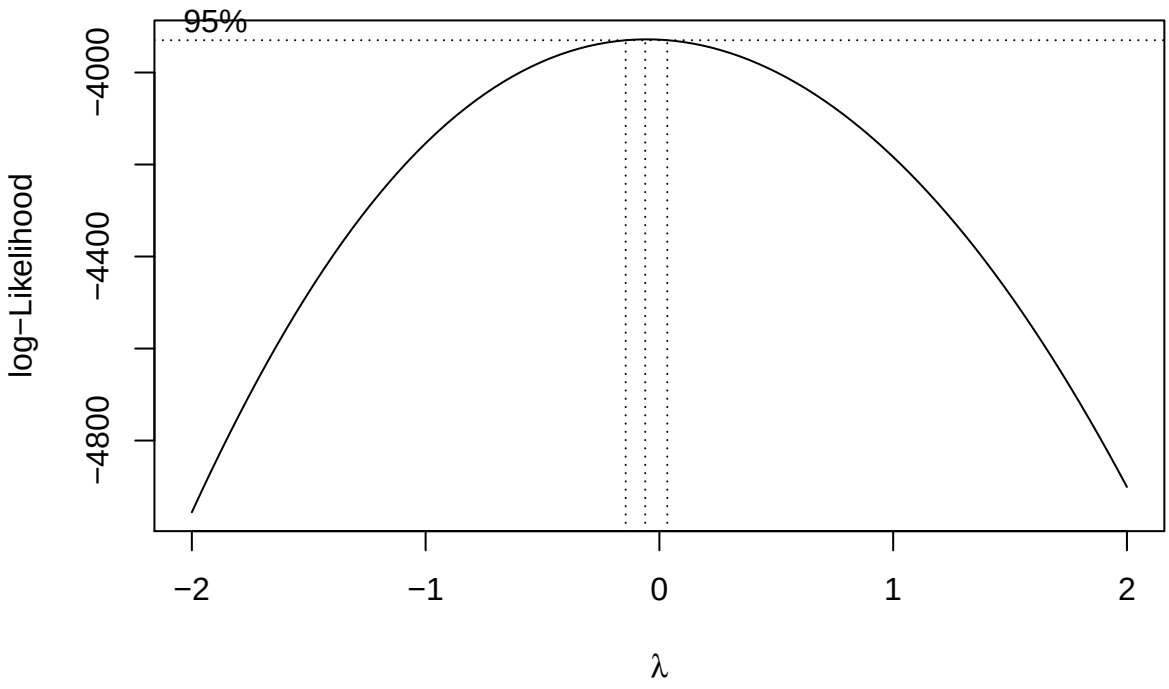





```
df %>% ggplot(aes(x = LotArea, y = SalePrice)) + geom_point()
```




```
#box-cox transformation
bc = boxcox(SalePrice ~ LotArea, data = df)
```




```
cor(bc$x, bc$y, method = 'spearman')
```

```
## [1] -0.01456946
```

```
cor(df$LotArea, df$SalePrice)
```

```
## [1] 0.2638434
```

Linear Algebra and Correlation

```
cor_mat = cor(df)
prec_mat = solve(cor_mat)  # invert correlation matrix
cor_mat %*% prec_mat  # results in identity matrix
```

```
##            SalePrice LotArea
## SalePrice          1       0
## LotArea            0       1
```

```
prec_mat %*% cor_mat  # results in identity matrix
```

```
##            SalePrice LotArea
## SalePrice          1       0
## LotArea            0       1
```

Calculus-Based Probability & Statistics

I fit a lognormal distribution to the lot area data; the actual data exhibits much more skew wit a fatter tail.

```
# fit lognormal distribution to data
fit = fitdistr(df$LotArea, densfun = 'lognormal')
meanLog = fit$estimate[1]
sdLog = fit$estimate[2]

samp = rlnorm(1000, meanLog, sdLog)
typeof(samp)
```

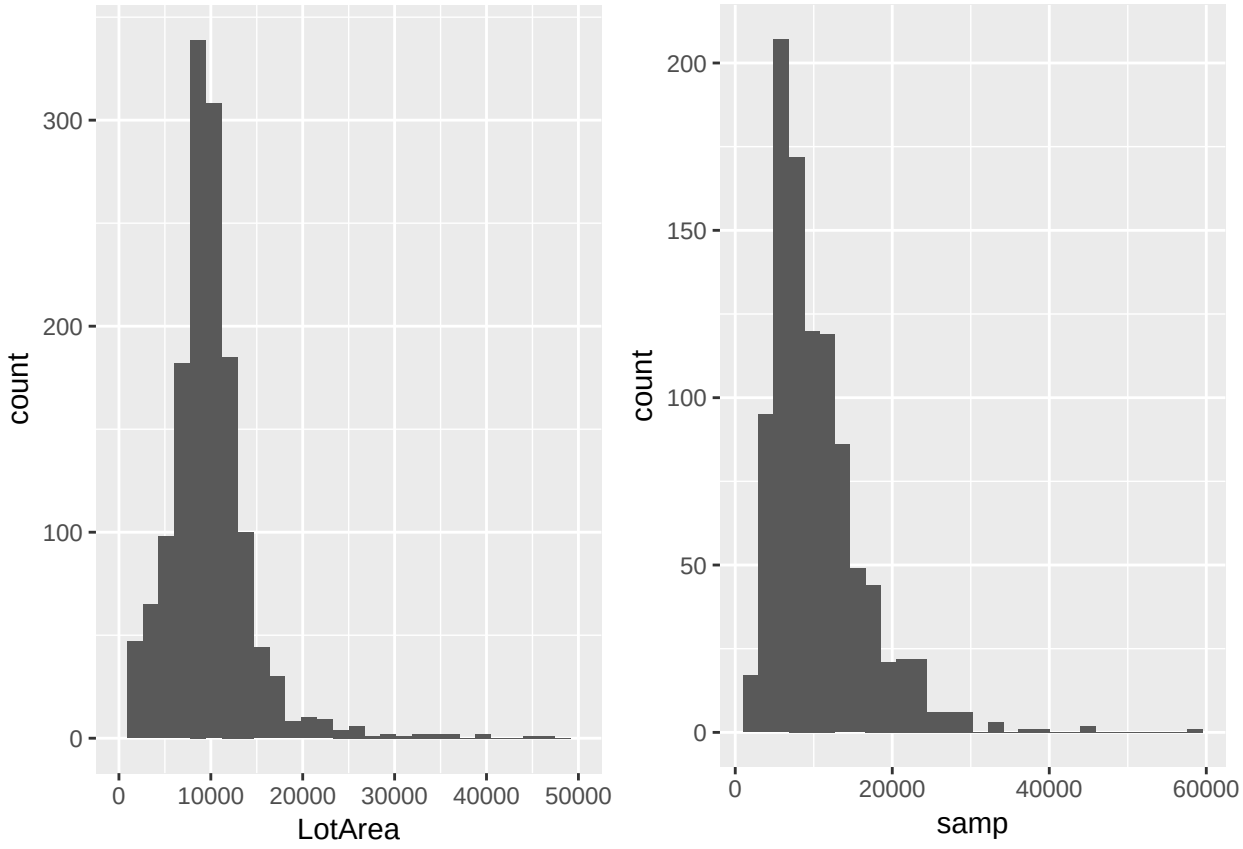
```
## [1] "double"
```

```
p1 = df %>% ggplot(aes(x = LotArea)) + geom_histogram() + xlim(c(0, 50000))
p2 = qplot(samp, geom = 'histogram')
grid.arrange(p1, p2, ncol = 2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 11 rows containing non-finite values (stat_bin).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Modeling

```
library(caret)
```

```
## Loading required package: lattice
```

```
set.seed(123)
myControl = trainControl(method = 'cv', number = 5, verboseIter = TRUE)

#try glmnet model
clean = imputeMissings::impute(kaggle, method = 'median/mode')  # impute medians for missing data

glmnet_model = train(SalePrice ~ ., data = clean, method = 'glmnet', trControl = myControl,
                     preProcess = c('center', 'scale'))
```

```
## Loading required package: glmnet

## Loading required package: Matrix

## Loading required package: foreach

## Loaded glmnet 2.0-5

## + Fold1: alpha=0.10, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## ElectricalMix

## - Fold1: alpha=0.10, lambda=12563
## + Fold1: alpha=0.55, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## ElectricalMix

## - Fold1: alpha=0.55, lambda=12563
## + Fold1: alpha=1.00, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## ElectricalMix

## - Fold1: alpha=1.00, lambda=12563
## + Fold2: alpha=0.10, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2RRAe,
## Condition2RRAn, RoofMatlMetal, RoofMatlRoll, HeatingOthW, HeatingQCPo,
## FunctionalSev

## - Fold2: alpha=0.10, lambda=12563
## + Fold2: alpha=0.55, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2RRAe,
## Condition2RRAn, RoofMatlMetal, RoofMatlRoll, HeatingOthW, HeatingQCPo,
## FunctionalSev

## - Fold2: alpha=0.55, lambda=12563
## + Fold2: alpha=1.00, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2RRAe,
## Condition2RRAn, RoofMatlMetal, RoofMatlRoll, HeatingOthW, HeatingQCPo,
## FunctionalSev
```

```
## - Fold2: alpha=1.00, lambda=12563
## + Fold3: alpha=0.10, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## NeighborhoodBlueste, Condition2PosA, Condition2PosN, RoofMatlMembran,
## Exterior1stAsphShn, Exterior1stCBlock, Exterior2ndCBlock

## - Fold3: alpha=0.10, lambda=12563
## + Fold3: alpha=0.55, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## NeighborhoodBlueste, Condition2PosA, Condition2PosN, RoofMatlMembran,
## Exterior1stAsphShn, Exterior1stCBlock, Exterior2ndCBlock

## - Fold3: alpha=0.55, lambda=12563
## + Fold3: alpha=1.00, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## NeighborhoodBlueste, Condition2PosA, Condition2PosN, RoofMatlMembran,
## Exterior1stAsphShn, Exterior1stCBlock, Exterior2ndCBlock

## - Fold3: alpha=1.00, lambda=12563
## + Fold4: alpha=0.10, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: MiscFeatureTenC

## - Fold4: alpha=0.10, lambda=12563
## + Fold4: alpha=0.55, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: MiscFeatureTenC

## - Fold4: alpha=0.55, lambda=12563
## + Fold4: alpha=1.00, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: MiscFeatureTenC

## - Fold4: alpha=1.00, lambda=12563
## + Fold5: alpha=0.10, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior1stImStucc,
## ExterCondPo

## - Fold5: alpha=0.10, lambda=12563
## + Fold5: alpha=0.55, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior1stImStucc,
## ExterCondPo

## - Fold5: alpha=0.55, lambda=12563
## + Fold5: alpha=1.00, lambda=12563

## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior1stImStucc,
## ExterCondPo
```

```
## - Fold5: alpha=1.00, lambda=12563
## Aggregating results
## Selecting tuning parameters
## Fitting alpha = 0.1, lambda = 12563 on full training set
```

```
#try random forest model
rf_model = train(SalePrice ~ ., data = clean, method = 'ranger', trControl = myControl,
                     preProcess = c('center', 'scale'))
```

```
## Loading required package: e1071
```

```
## Loading required package: ranger
```

```
## + Fold1: mtry=  2
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: RoofMatlRoll
```

```
## - Fold1: mtry=  2
## + Fold1: mtry=124
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: RoofMatlRoll
```

```
## - Fold1: mtry=124
## + Fold1: mtry=246
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: RoofMatlRoll
```

```
## - Fold1: mtry=246
## + Fold2: mtry=  2
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## MiscFeatureTenC
```

```
## - Fold2: mtry=  2
## + Fold2: mtry=124
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## MiscFeatureTenC
```

```
## - Fold2: mtry=124
## + Fold2: mtry=246
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Exterior2ndOther,
## MiscFeatureTenC
```

```
## - Fold2: mtry=246
## + Fold3: mtry=  2
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: Condition2RRNn, RoofMatlMetal
```

```
## - Fold3: mtry=  2
## + Fold3: mtry=124
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: Condition2RRNn, RoofMatlMetal
```

```
## - Fold3: mtry=124
## + Fold3: mtry=246
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19, uniqueCut
## = 10, : These variables have zero variances: Condition2RRNn, RoofMatlMetal
```

```
## - Fold3: mtry=246
## + Fold4: mtry=  2
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2PosA,
## Condition2RRAe, RoofMatlMembran, Exterior1stAsphShn, Exterior1stCBlock,
## Exterior1stImStucc, Exterior2ndCBlock, ExterCondPo, HeatingQCPo,
## ElectricalMix
```

```
## - Fold4: mtry=  2
## + Fold4: mtry=124
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2PosA,
## Condition2RRAe, RoofMatlMembran, Exterior1stAsphShn, Exterior1stCBlock,
## Exterior1stImStucc, Exterior2ndCBlock, ExterCondPo, HeatingQCPo,
## ElectricalMix
```

```
## - Fold4: mtry=124
## + Fold4: mtry=246
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: Condition2PosA,
## Condition2RRAe, RoofMatlMembran, Exterior1stAsphShn, Exterior1stCBlock,
## Exterior1stImStucc, Exterior2ndCBlock, ExterCondPo, HeatingQCPo,
## ElectricalMix
```

```
## - Fold4: mtry=246
## + Fold5: mtry=  2
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## Condition2RRAn, FunctionalSev
```

```
## - Fold5: mtry=  2
## + Fold5: mtry=124
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## Condition2RRAn, FunctionalSev
```

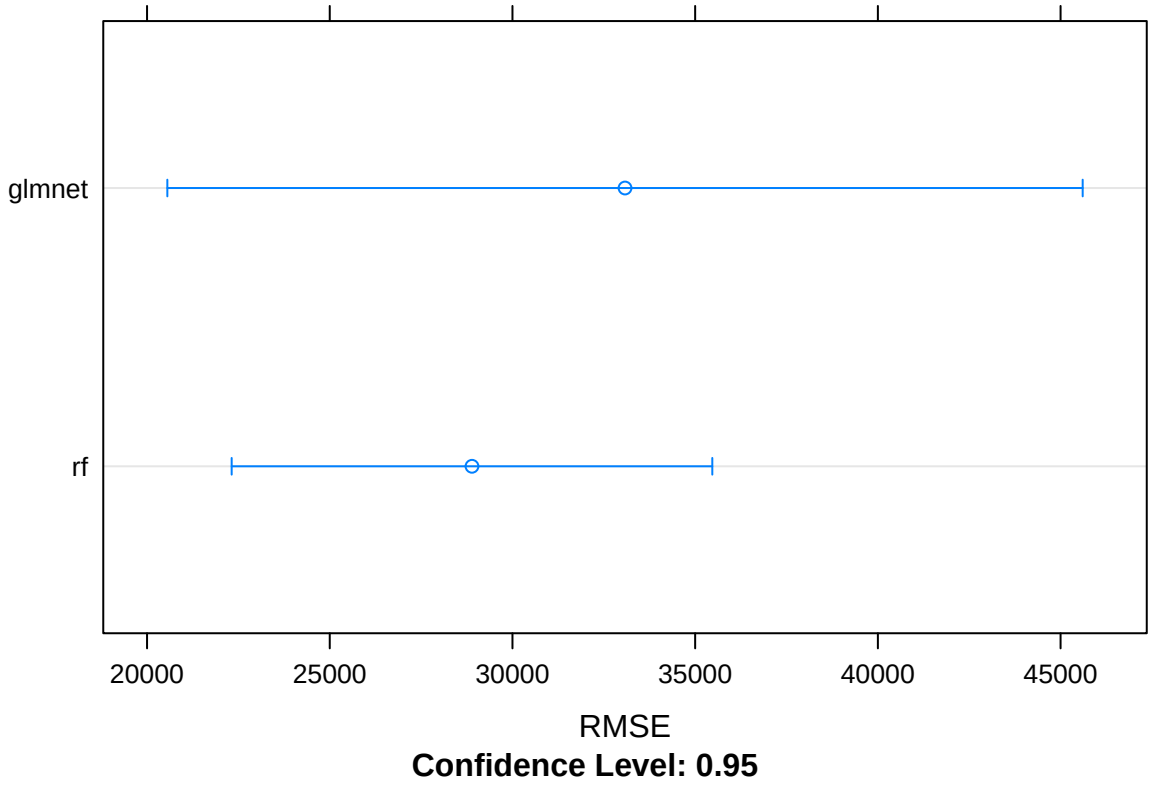
```
## - Fold5: mtry=124
## + Fold5: mtry=246
```

```
## Warning in preProcess.default(thresh = 0.95, k = 5, freqCut = 19,
## uniqueCut = 10, : These variables have zero variances: UtilitiesNoSeWa,
## Condition2RRAn, FunctionalSev
```

```
## - Fold5: mtry=246
## Aggregating results
## Selecting tuning parameters
## Fitting mtry = 124 on full training set
```

```
#compare models
model_list = list(glmnet = glmnet_model, rf = rf_model)
```

```
#collect resamples from the CV folds
resamps = resamples(model_list)
dotplot(resamps, metric = 'RMSE')
```




```
#load test data
test = read.csv("C:/Users/bhao/Google Drive/CUNY/git/DATA605/Final/test.csv")
test_clean = imputeMissings::impute(test, method = 'median/mode')  # impute medians for missing data
test_pred = predict(rf_model, newdata = test_clean)
test_pred = cbind(test, test_pred) %>% mutate(SalePrice = test_pred) %>% dplyr::select(Id, SalePrice)
write.csv(test_pred, "C:/Users/bhao/Google Drive/CUNY/git/DATA605/Final/rf_pred.csv")
```