R setup

```
library(readr)
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(fitdistrplus)
```

```
## Loading required package: MASS
```

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

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

```
## Loading required package: survival
```

```
library(ggplot2)

library(olsrr)
```

```
##
## Attaching package: 'olsrr'
```

```
## The following object is masked from 'package:MASS':
##
##     cement
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
library(rpart)
library(rpart.plot)
library(caret)
```

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

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:survival':
##
##     cluster
```

```
library(randomForest)
```

```
## randomForest 4.7-1.2
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

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

Problem 3

Setup

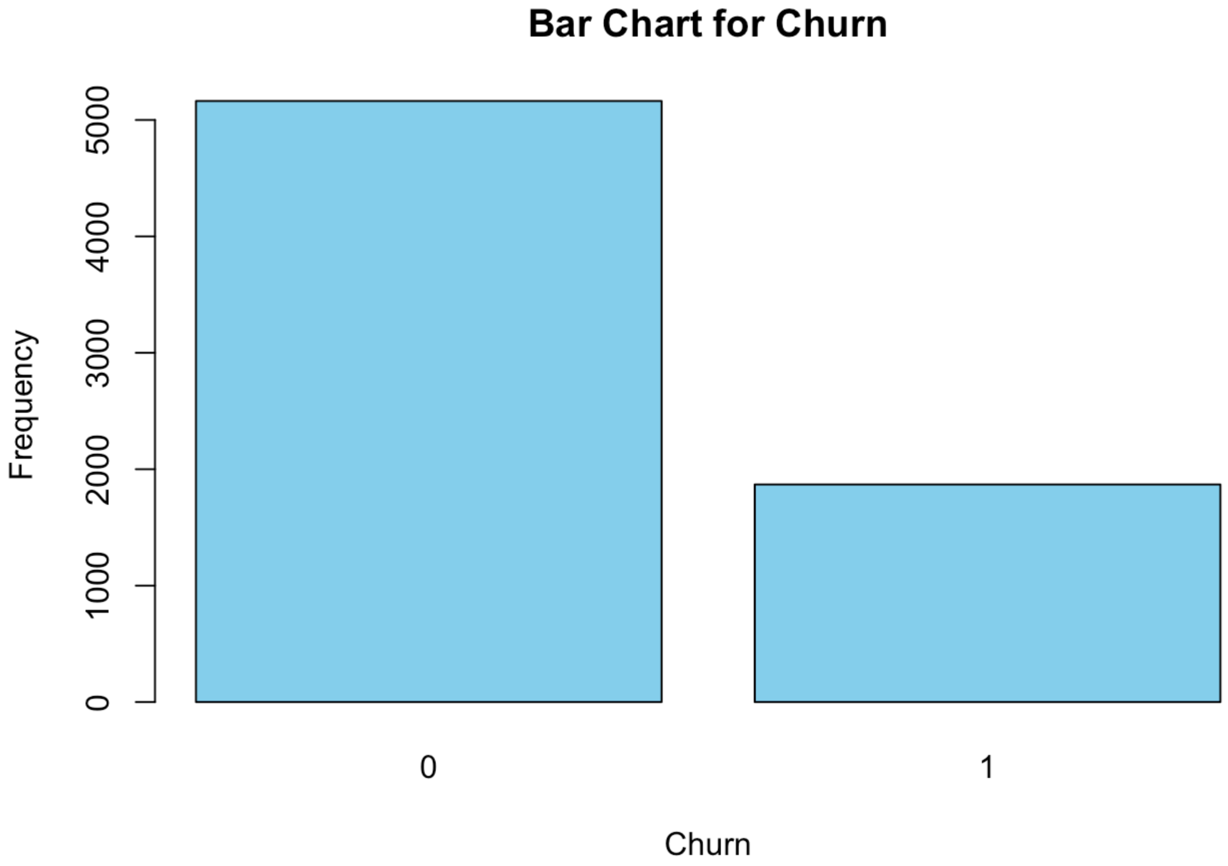
```
df_churn <- read_csv("/Users/riyaparikh_computeracct/Downloads/MIT/15.072_AdvancedAnalyt
icsEdge/deliverable2-analyticsedge-mit/customerchurn.csv", show_col_types = FALSE)
```

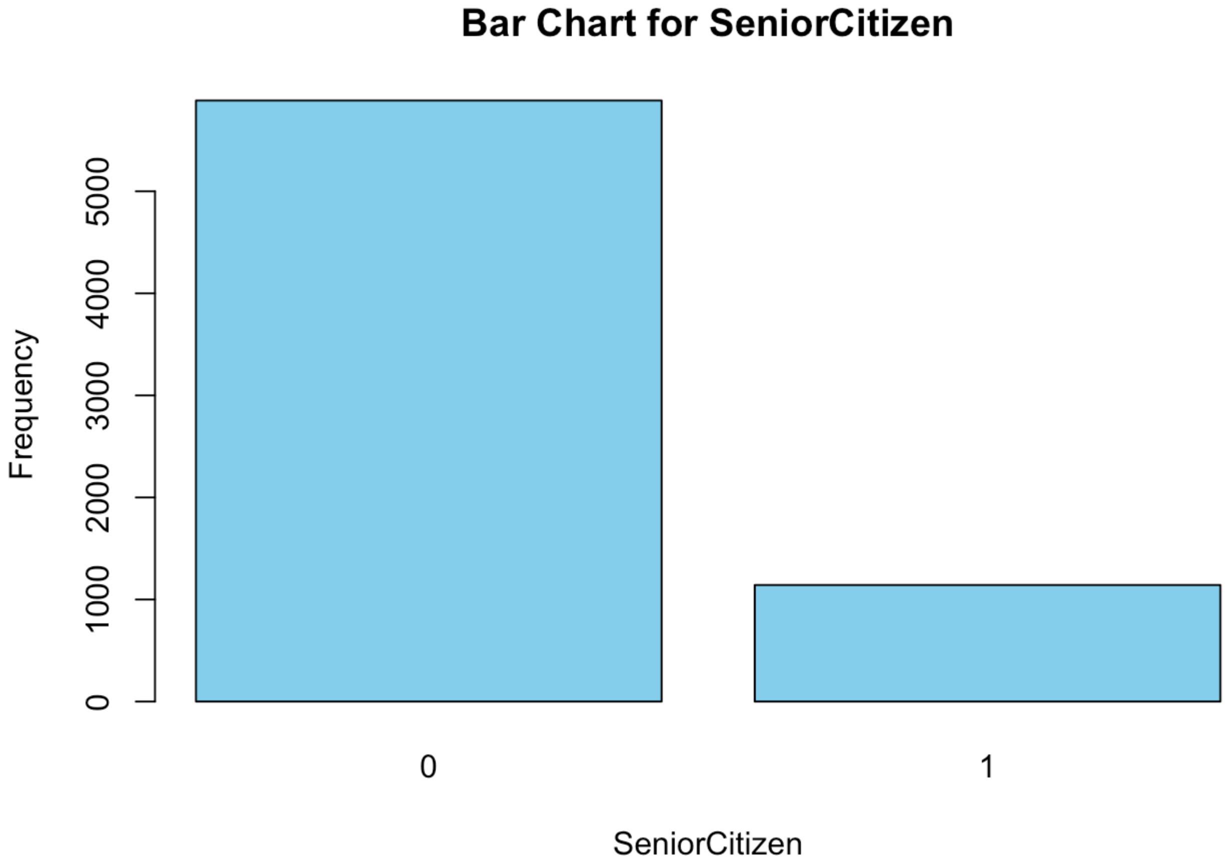
Part a

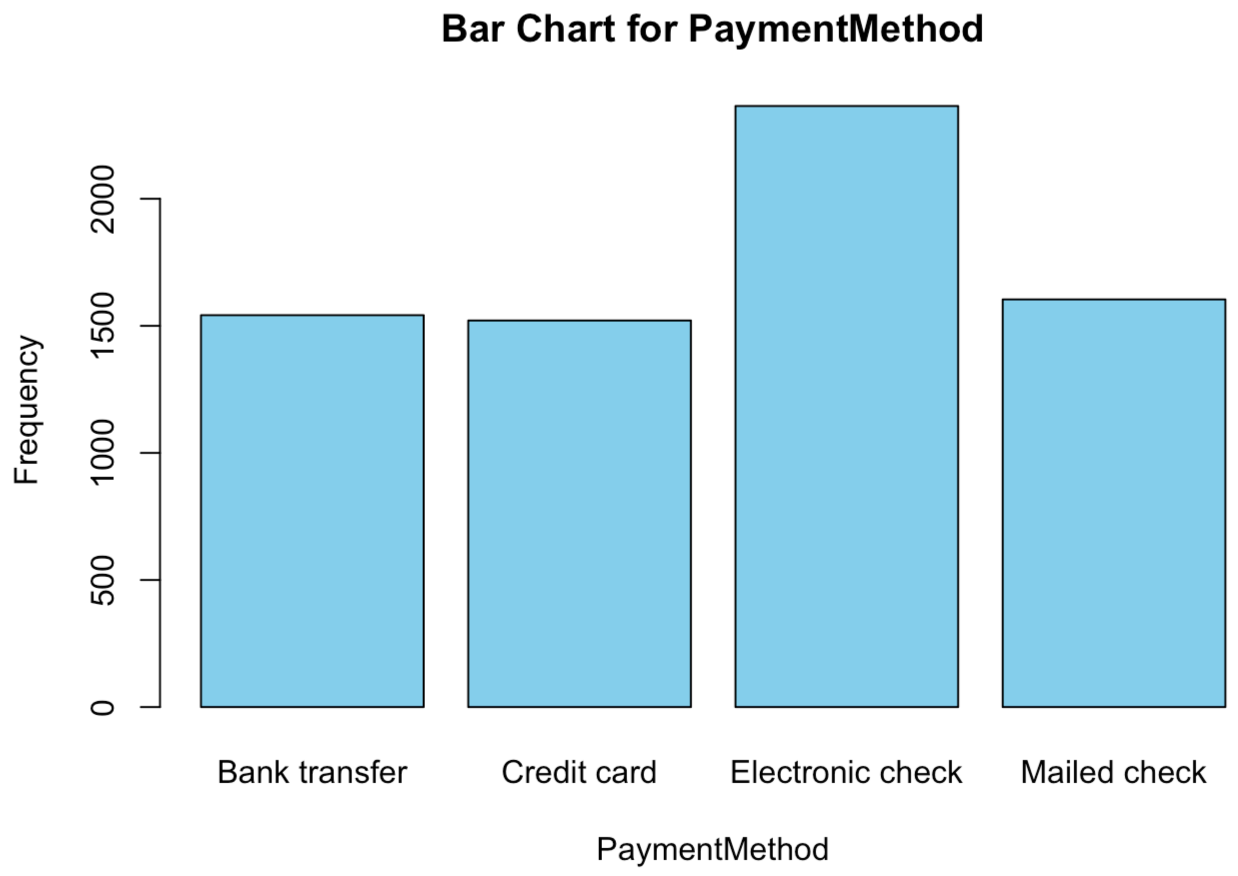
```
summary(df_churn)
```

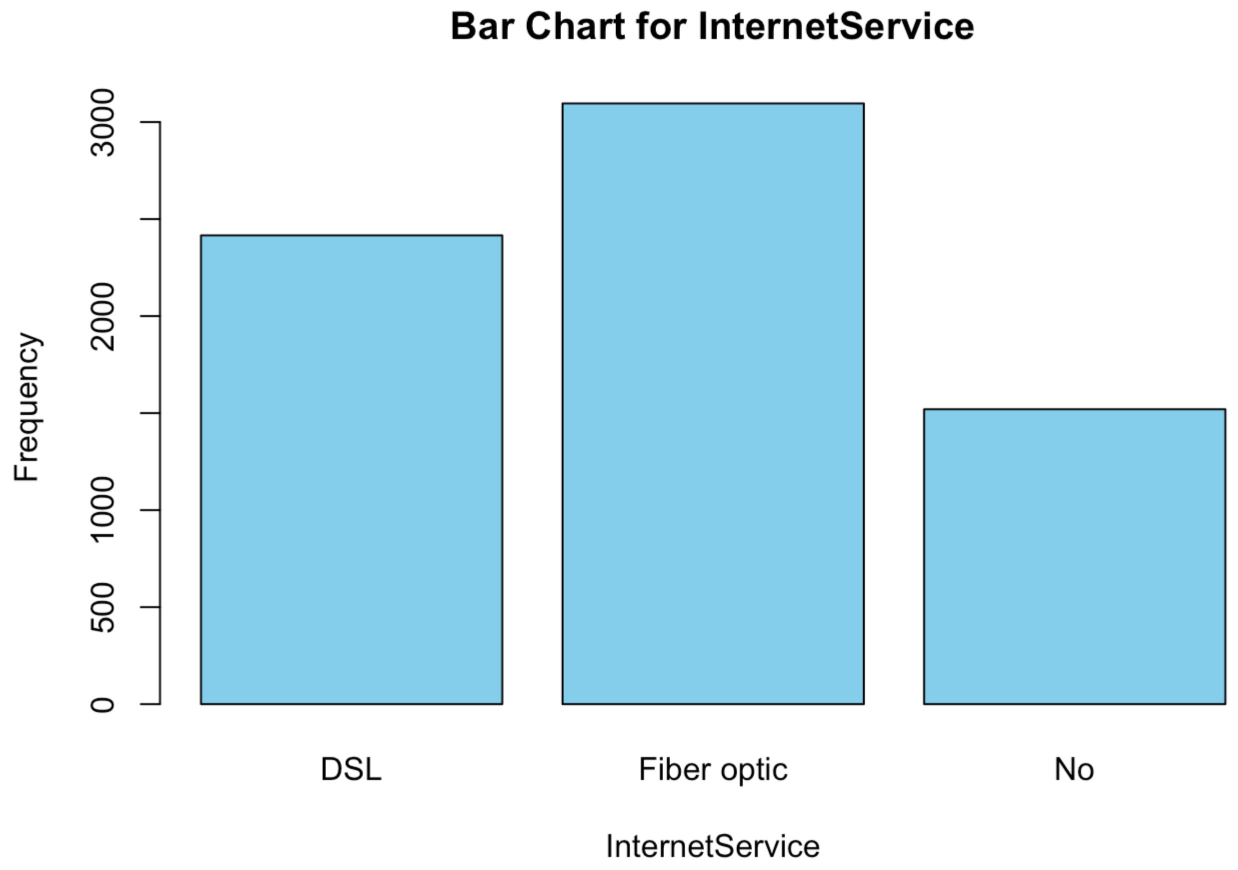
```
##      Churn        MonthlyCharges   SeniorCitizen    PaymentMethod     
##  Min.   :0.0000   Min.   : 18.25   Min.   :0.0000   Length:7032       
##  1st Qu.:0.0000   1st Qu.: 35.59   1st Qu.:0.0000   Class :character  
##  Median :0.0000   Median : 70.35   Median :0.0000   Mode  :character  
##  Mean   :0.2658   Mean   : 64.80   Mean   :0.1624                     
##  3rd Qu.:1.0000   3rd Qu.: 89.86   3rd Qu.:0.0000                     
##  Max.   :1.0000   Max.   :118.75   Max.   :1.0000                     
##  InternetService        tenure       Contract        
##  Length:7032        Min.   : 1.00   Length:7032       
##  Class :character   1st Qu.: 9.00   Class :character  
##  Mode  :character   Median :29.00   Mode  :character  
##                     Mean   :32.42                     
##                     3rd Qu.:55.00                     
##                     Max.   :72.00                     
```

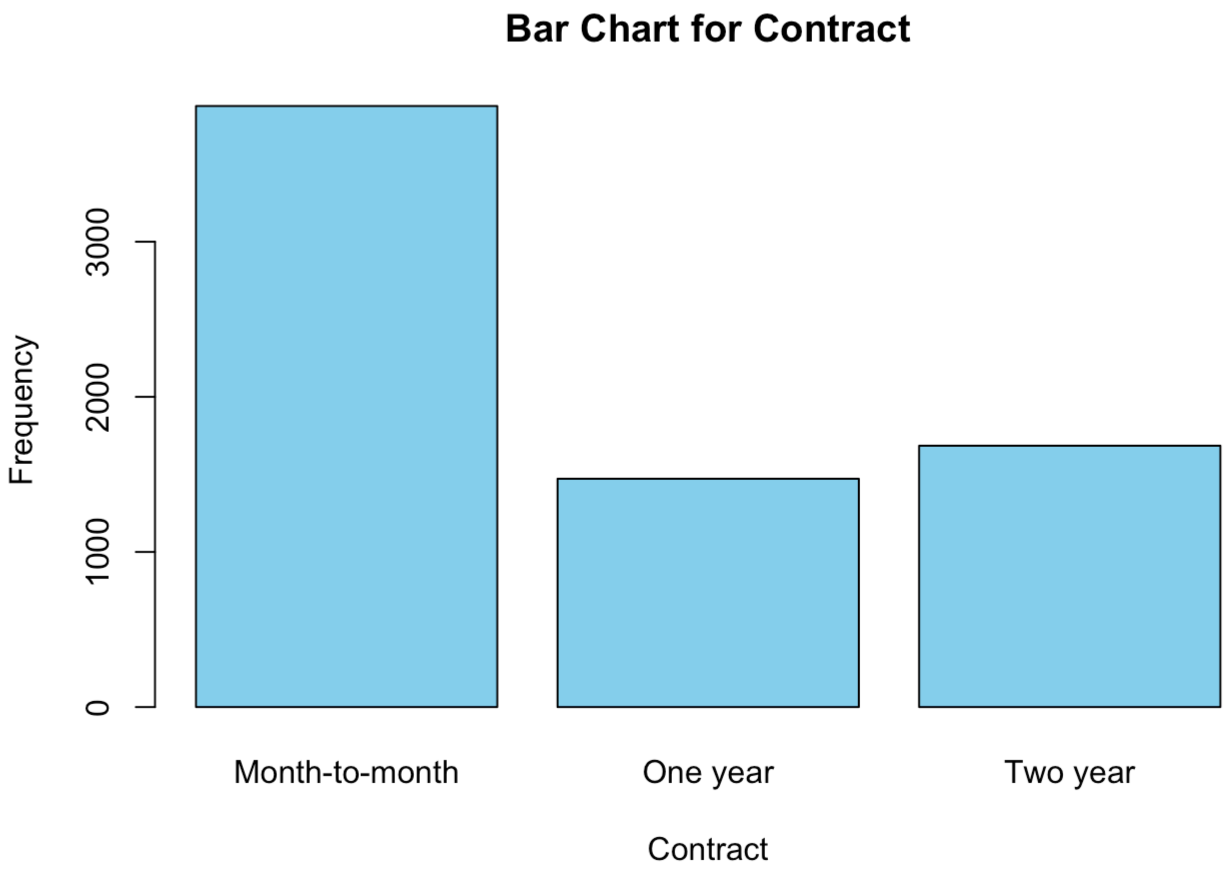
```
barchartvector = c("Churn", "SeniorCitizen", "PaymentMethod", "InternetService", "Contra
ct")
for (col_name in barchartvector) {
  counts <- table(df_churn[[col_name]])

  barplot(counts,
          main = paste("Bar Chart for", col_name),
          xlab = col_name,
          ylab = "Frequency",
          col = "skyblue")
}
```


Bar Chart for Churn
Frequency
0
1000
2000
3000
4000
5000
0
1
Churn

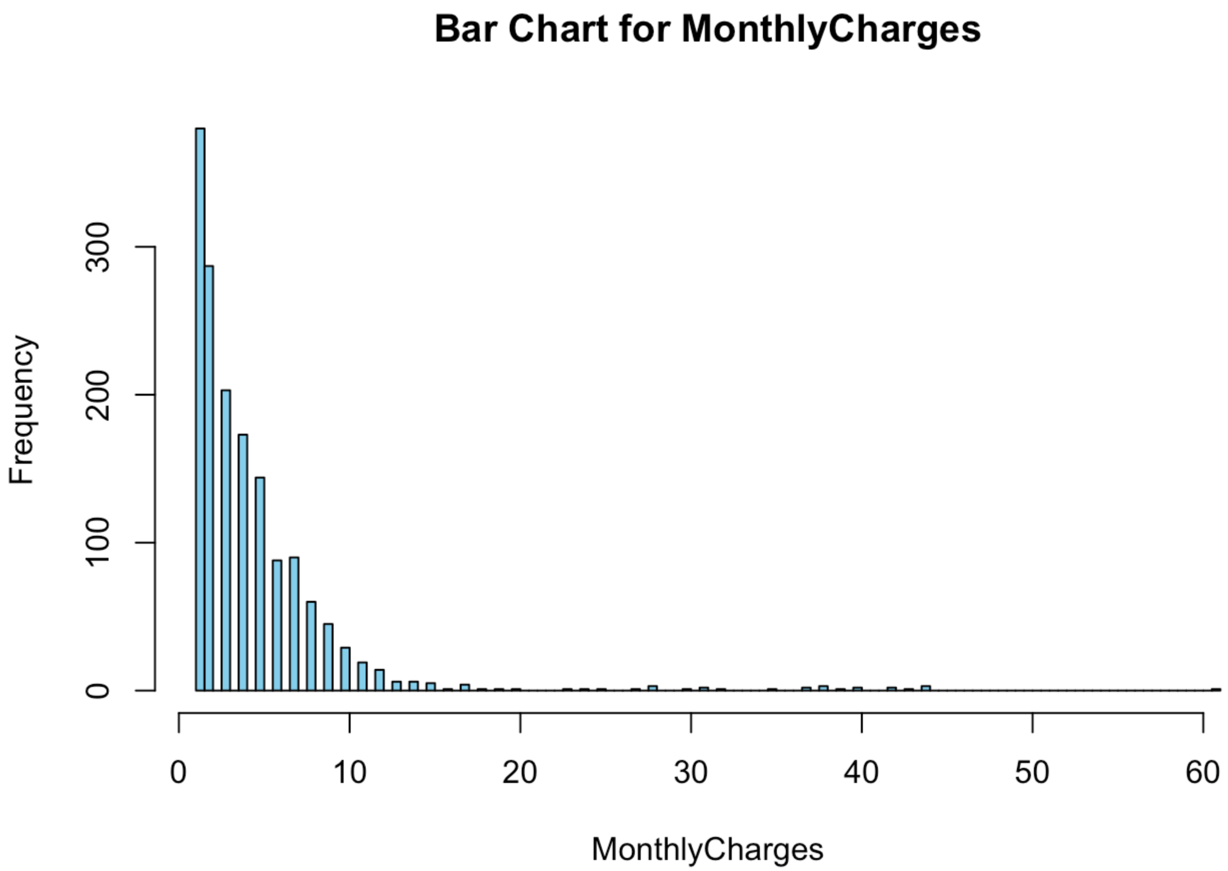

Bar Chart for SeniorCitizen
Frequency
0
1000
2000
3000
4000
5000
0
1
SeniorCitizen

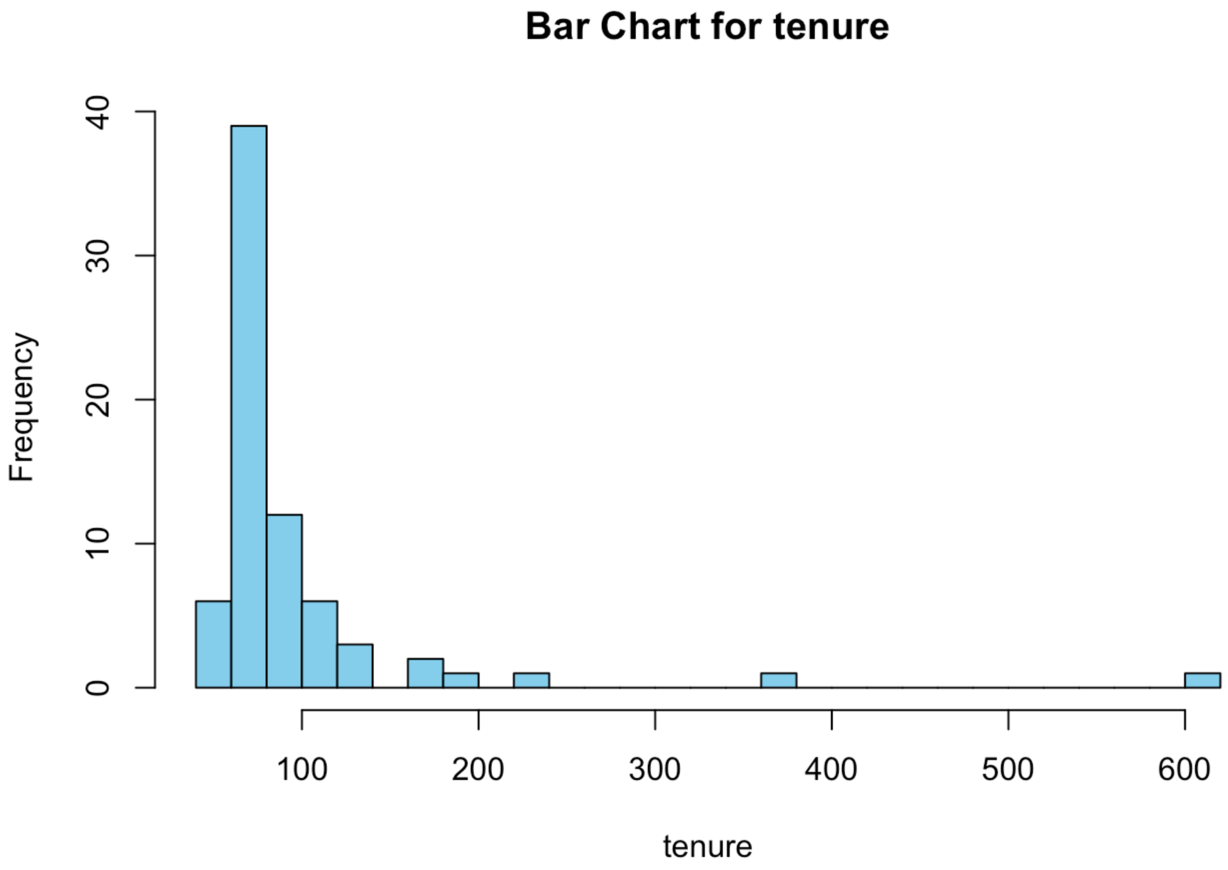

Bar Chart for PaymentMethod
Frequency
0
500
1000
1500
2000
Bank transfer
Credit card
Electronic check
Mailed check
PaymentMethod


Bar Chart for InternetService
Frequency
0
500
1000
2000
3000
DSL
Fiber optic
No
InternetService



```r
histvector = c("MonthlyCharges", "tenure")
for (col_name in histvector) {
  counts <- table(df_churn[[col_name]])

  hist(counts,
          main = paste("Bar Chart for", col_name),
          xlab = col_name,
          ylab = "Frequency",
          col = "skyblue", breaks = "FD")
}
```


Bar Chart for MonthlyCharges
Frequency
0
100
200
300
0
10
20
30
40
50
60
MonthlyCharges


Bar Chart for tenure
Frequency
0
10
20
30
40
100
200
300
400
500
600
tenure

```
churnpercentage <- df_churn %>% filter(Churn == "1") %>% nrow() / length(df_churn$Churn)
nochurnpercentage <- 1-churnpercentage
cat("Churn percentage:", churnpercentage*100, "%", "and non churn percentage:", nochurnp
ercentage*100, "%")
```

```
## Churn percentage: 26.5785 % and non churn percentage: 73.4215 %
```

From this output, we have a few key takeaways: the churn percentage is 26.5785% and the non churn percentage is 73.4215%, meaning we have approx 3x more observations of non churners than churners. It might be useful to balance the data set in following questions, we will see if necessary. We also realize that our data set is also unbalanced in terms of age demographics, having only 16.24% senior citizens. For the payment methods, users have the choice between 4 different payment methods including electronic check which is the most popular choice, and bank transfer/credit card/mailed check which all have pretty similar utilization. To provide more color on internet service, we see that the most popular service is fiber optic followed by DSL, but many people report not having a service. Moreover, the majority of people pay for these services month to month rather than being locked in for 1 or 2 years. With these services, we see that users pay on average of $64.80 per month, with a high of $118.75 and a low of $18.25. Charges are skewed right as is the histogram of tenure.

Part b

```
set.seed(15072)

# creating a binary dependent variable for churn (0 = no, 1 = churn)
df_churn$Churn <- as.factor(df_churn$Churn)

# training (70%) and test (30%) partition
smp_size <- floor(0.70 * nrow(df_churn))
train_ind <- sample(seq_len(nrow(df_churn)), size = smp_size, replace = FALSE)
train_churn <- df_churn[train_ind, ]
test_churn <- df_churn[-train_ind, ]

#  exclude payment method from predictors in model
model <- glm(Churn ~ . - PaymentMethod, data = train_churn, family = "binomial")

summary(model)
```

```
## 
## Call:
## glm(formula = Churn ~ . - PaymentMethod, family = "binomial", 
##     data = train_churn)
## 
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)    
## (Intercept)               -0.492422   0.188257  -2.616   0.0089 ** 
## MonthlyCharges             0.003523   0.003632   0.970   0.3321    
## SeniorCitizen              0.460054   0.095973   4.794 1.64e-06 ***
## InternetServiceFiber optic  1.041354   0.153243   6.795 1.08e-11 ***
## InternetServiceNo         -0.886428   0.177866  -4.984 6.24e-07 ***
## tenure                    -0.033687   0.002497 -13.492  < 2e-16 ***
## ContractOne year          -0.829613   0.124337  -6.672 2.52e-11 ***
## ContractTwo year          -1.736190   0.212439  -8.173 3.02e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 5763.5  on 4921  degrees of freedom
## Residual deviance: 4227.2  on 4914  degrees of freedom
## AIC: 4243.2
## 
## Number of Fisher Scoring iterations: 6
```

The coefficient of 0.460054 for SeniorCitizen indicates that being a senior increases the log-odds of churn by 0.460054 ($p < 0.001$). In terms of odds, senior citizens are about 1.59 (=exp(0.460054)) times more likely to churn compared to non-seniors, holding all else constant.

Part c

```
fifth <- df_churn[5, ]

predictions <- predict(model, newdata = fifth)
print(predictions)
```

```
##         1 
## 0.7306284
```

User is 73.06284% likely to churn.

Part d

```
predictionsfull <- predict(model, newdata = test_churn, type = "response")
custom_threshold <- 0.3
predicted_classes <- ifelse(predictionsfull > custom_threshold, 1, 0)

confusion_matrix <- table(predicted_classes, test_churn$Churn)
confusion_matrix
```

```
##                  
## predicted_classes    0    1
##                 0 1158  119
##                 1  423  410
```

Part e

False positive - we inaccurately predicted that a user will churn even though they did not actually.
False negative - we inaccurately predicted that a user will not churn at the end of the year despite the fact that they did.

Part f

If I was a business analyst at Watson Analytics, I would focus on minimizing false negatives. Churn refers to the loss of customers, subscribers, or clients over a given period. It’s a key metric for assessing customer satisfaction, loyalty, and the overall health of a business, especially in subscription-based models like we are looking at here. In the case of a false negative, we thought that a user would stay with us after the end of the year by assuming they were happy with our services. However, they in fact were unsatisfied and chose to leave us. This leaves us with a lost customer and an additional cost of having to expend time, money, and effort to gain new customers. While a false positive also isn’t an accurate prediction from our model, its implications are far less harmful. It only means that a user ended up staying with our company despite the fact that we thought they’d leave.

Part g

```
# increase threshold to 0.4
custom_threshold2 <- 0.4
predicted_classes2 <- ifelse(predictionsfull > custom_threshold2, 1, 0)
confusion_matrix2 <- table(predicted_classes2, test_churn$Churn)
dimnames(confusion_matrix2) <- list(Actual = c("No Churn", "Churn"), Pred = c("No Chur
n", "Churn"))
confusion_matrix2 <- confusion_matrix2/sum(confusion_matrix2)

# increase threshold to 0.5
custom_threshold3 <- 0.5
predicted_classes3 <- ifelse(predictionsfull > custom_threshold3, 1, 0)
confusion_matrix3 <- table(predicted_classes3, test_churn$Churn)
dimnames(confusion_matrix3) <- list(Actual = c("No Churn", "Churn"), Pred = c("No Chur
n", "Churn"))
confusion_matrix3 <- confusion_matrix3/sum(confusion_matrix3)

# increase threshold to 0.6
custom_threshold4 <- 0.6
predicted_classes4 <- ifelse(predictionsfull > custom_threshold4, 1, 0)
confusion_matrix4 <- table(predicted_classes4, test_churn$Churn)
dimnames(confusion_matrix4) <- list(Actual = c("No Churn", "Churn"), Pred = c("No Chur
n", "Churn"))
confusion_matrix4 <- confusion_matrix4/sum(confusion_matrix4)

confusion_matrix_rate <- confusion_matrix/sum(confusion_matrix)

confusion_matrix_rate
```

```
##                  
## predicted_classes         0         1
##                 0 0.5488152 0.0563981
##                 1 0.2004739 0.1943128
```

```
confusion_matrix2
```

```
##           Pred
## Actual       No Churn      Churn
##   No Churn 0.61990521 0.09194313
##   Churn    0.12938389 0.15876777
```

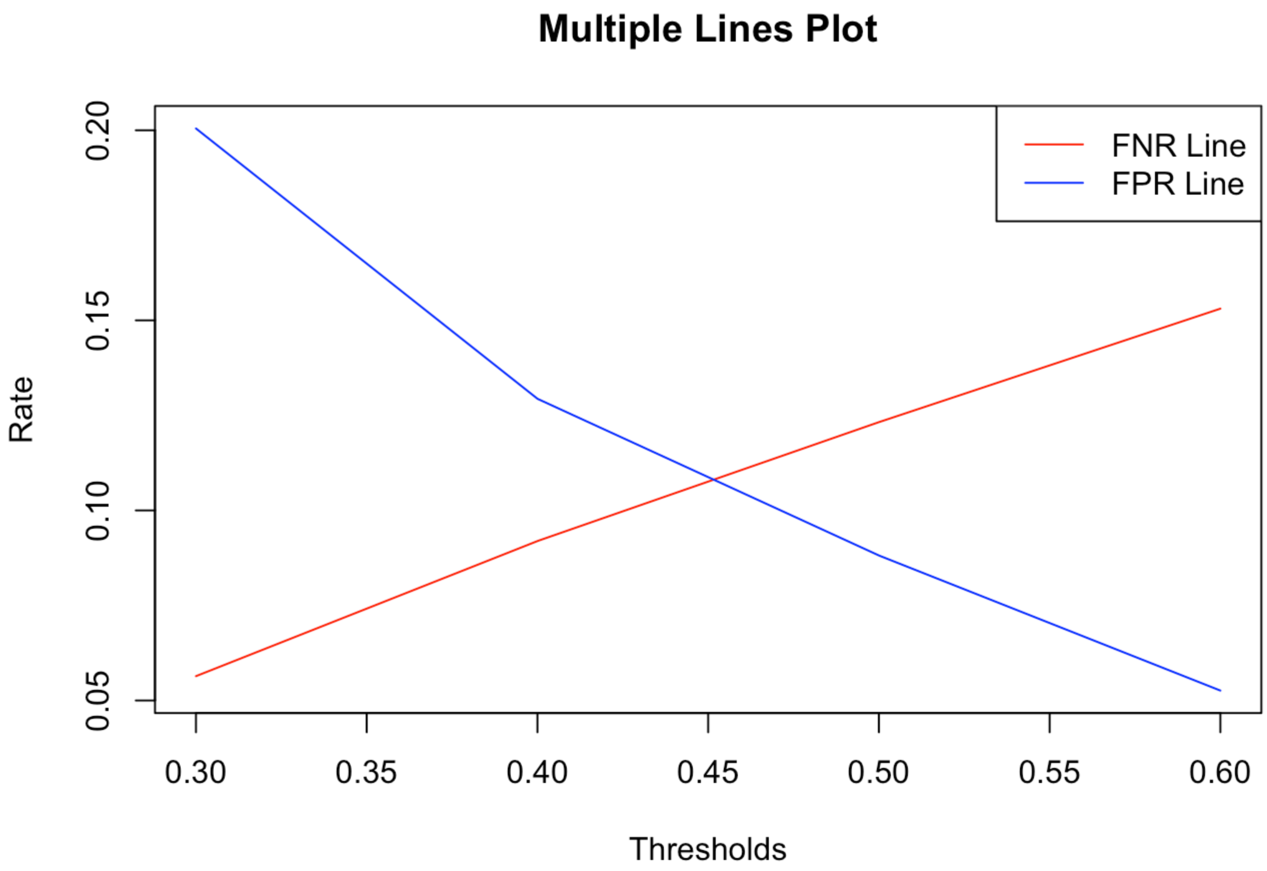
```
confusion_matrix3
```

```
##           Pred
## Actual       No Churn      Churn
##   No Churn 0.66113744 0.12322275
##   Churn    0.08815166 0.12748815
```

```
confusion_matrix4
```

```
##            Pred
## Actual       No Churn      Churn
##   No Churn 0.69668246 0.15308057
##   Churn    0.05260664 0.09763033
```

```
# rates
rates = seq(from = .3, to = .6, length.out = 4)

# false negatives
fnr = c(0.0563981, 0.09194313, 0.12322275, 0.15308057)

# false positives
fpr = c(0.2004739,0.12938389, 0.08815166, 0.05260664)

# plot multiple lines using matplot
matplot(rates, cbind(fnr, fpr), type = "l", lty = 1,
        col = c("red", "blue"), xlab = "Thresholds",
        ylab = "Rate", main = "Multiple Lines Plot")
legend("topright", legend = c("FNR Line", "FPR Line"),
       col = c("red", "blue"),
       lty = 1)
```



As you can see in the plot, as the threshold increases, the FPR increases and the FNR decreases. They both move in opposite directions. This is because the threshold signifies with what probability we must have confidence in our prediction for it to be classified “1”. If this threshold keeps getting higher and higher, it is harder for us to have confidence of that high of a level. Thus, we label as “1” less often and the fpr goes down.

Part h

```
thresholds <- seq(0, 1, length.out = 20)
total_value_list <- c()

for (i in thresholds) {
  predicted_classes <- ifelse(predictionsfull > i, 1, 0)
  confusion_matrix <- table(
  factor(predicted_classes, levels = c(0,1)),
  factor(test_churn$Churn, levels = c(0,1))
)

  # Payoff calculation
  TN <- confusion_matrix[1,1] * 3000
  FP <- confusion_matrix[2,1] * -1000
  FN <- confusion_matrix[1,2] * -6000
  TP <- confusion_matrix[2,2] * 2000

  total_val <- TN + FP + FN + TP
  total_value_list <- c(total_value_list, total_val)
}

# find best threshold
best_idx <- which.max(total_value_list)
best_threshold <- thresholds[best_idx]
best_profit <- total_value_list[best_idx]

cat("Best index: ", best_idx)
```

```
## Best index:  8
```

```
cat("Best threshold: ", best_threshold)
```

```
## Best threshold:  0.3684211
```

```
cat("Best profit: ", best_profit)
```

```
## Best profit:  3237000
```

We'd optimize the probability threshold by finding out which threshold leads to max profit. By looping through 20 values from 0 to 1 (probability(Churn=1)=0 to probability(Churn=1)=1), we can see how each threshold produces different # FP, FN, TP, and TN Since each of these have an associated cost/profit with them, we can then multiply # * cost for all 4 result types. Then, by finding the one with the max profitability, we can backtrack and find which threshold was associated with that profitability.
Here, the best threshold is 0.3684211 associated with a profit of $3,237,000.

Problem 4

Setup

```
df_ames <- read_csv("/Users/riyaparikh_computeracct/Downloads/MIT/15.072_AdvancedAnalyticsEdge/deliverable2-analyticsedge-mit/ames.csv", show_col_types = FALSE)

set.seed(15072)

# training (70%) and test (30%) partition
smp_size <- floor(0.70 * nrow(df_ames))
train_ind <- sample(seq_len(nrow(df_ames)), size = smp_size, replace = FALSE)
train_ames <- df_ames[train_ind, ]
test_ames <- df_ames[-train_ind, ]
```

Part a

```
mod_linear_intial <- lm(SalePrice ~ ., data = train_ames)
summary(mod_linear_intial)
```

```
##
## Call:
## lm(formula = SalePrice ~ ., data = train_ames)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -304836  -10699     -42   10472  129099
##
## Coefficients: (7 not defined because of singularities)
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -3.177e+05  8.923e+05  -0.356 0.721876
## MSZoningRL               4.069e+03  3.950e+03   1.030 0.303126
## MSZoningRM               4.903e+03  4.666e+03   1.051 0.293482
## LotFrontage             -1.302e+02  4.290e+01  -3.035 0.002437 **
## LotArea                  1.527e-01  1.702e-01   0.897 0.369689
## StreetPave              -9.429e+02  1.511e+04  -0.062 0.950249
## AlleyNo Alley            4.545e+03  3.329e+03   1.365 0.172329
## AlleyPave                2.907e+03  5.056e+03   0.575 0.565398
## LotShapeMod+ IR         -1.345e+03  3.536e+03  -0.380 0.703705
## LotShapeReg             -6.858e+02  1.367e+03  -0.502 0.616051
## LandContourHLS           1.889e+04  4.199e+03   4.498 7.31e-06 ***
## LandContourLow           4.018e+03  5.384e+03   0.746 0.455535
## LandContourLvl           1.167e+04  3.084e+03   3.782 0.000161 ***
## LotConfigCulDSac         6.310e+02  3.035e+03   0.208 0.835326
## LotConfigFR2            -5.913e+03  3.624e+03  -1.632 0.102931
## LotConfigFR3            -4.961e+02  7.475e+03  -0.066 0.947086
## LotConfigInside         -1.746e+03  1.552e+03  -1.125 0.260796
## LandSlopeNot Gtl         9.969e+03  3.339e+03   2.986 0.002866 **
## NeighborhoodBlueste     -7.305e+03  1.301e+04  -0.561 0.574572
## NeighborhoodBrDale       9.959e+03  9.714e+03   1.025 0.305392
## NeighborhoodBrkSide     -6.574e+03  8.148e+03  -0.807 0.419900
## NeighborhoodClearCr      7.643e+03  8.260e+03   0.925 0.354913
## NeighborhoodCollgCr      2.730e+03  6.628e+03   0.412 0.680465
## NeighborhoodCrawfor      2.022e+04  7.442e+03   2.717 0.006654 **
## NeighborhoodEdwards     -1.947e+04  7.089e+03  -2.746 0.006086 **
## NeighborhoodGilbert     -6.953e+03  6.904e+03  -1.007 0.313999
## NeighborhoodGreens       1.098e+04  1.394e+04   0.788 0.430975
## NeighborhoodIDOTRR      -1.297e+04  8.623e+03  -1.504 0.132676
## NeighborhoodMeadowV     -1.733e+04  8.862e+03  -1.955 0.050728 .
## NeighborhoodMitchel     -7.251e+03  7.151e+03  -1.014 0.310751
## NeighborhoodNAmes       -8.649e+03  7.014e+03  -1.233 0.217709
## NeighborhoodNoRidge      3.915e+04  7.527e+03   5.201 2.21e-07 ***
## NeighborhoodNPkVill      1.368e+04  9.349e+03   1.464 0.143501
## NeighborhoodNridgHt      4.249e+04  6.880e+03   6.175 8.15e-10 ***
## NeighborhoodNWAmes      -4.111e+03  7.192e+03  -0.572 0.567597
## NeighborhoodOldTown     -1.447e+04  8.170e+03  -1.771 0.076773 .
## NeighborhoodSawyer      -8.414e+03  7.225e+03  -1.165 0.244366
## NeighborhoodSawyerW     -1.783e+03  6.879e+03  -0.259 0.795571
## NeighborhoodSomerst      2.453e+04  7.379e+03   3.325 0.000903 ***
## NeighborhoodStoneBr      4.084e+04  7.629e+03   5.353 9.79e-08 ***
## NeighborhoodSWISU       -1.278e+04  8.391e+03  -1.523 0.127936
## NeighborhoodTimber       1.118e+04  7.199e+03   1.553 0.120521
## NeighborhoodVeenker      3.282e+04  9.467e+03   3.467 0.000539 ***
## Condition1Feedr          2.851e+03  4.172e+03   0.683 0.494457    
## Condition1Norm           1.107e+04  3.486e+03   3.176 0.001521 ** 
## Condition1PosA           2.523e+04  7.255e+03   3.478 0.000518 ***
## Condition1PosN           1.302e+04  5.864e+03   2.221 0.026490 *  
## Condition1RRAe          -7.330e+02  6.691e+03  -0.110 0.912777    
## Condition1RRAn           3.548e+03  5.709e+03   0.621 0.534376    
## Condition1RRNe          -3.459e+03  1.142e+04  -0.303 0.762068    
## Condition1RRNn           7.658e+03  1.165e+04   0.657 0.510981    
## Condition2Other         -9.149e+03  5.933e+03  -1.542 0.123246    
## BldgType2fmCon          -4.286e+03  4.744e+03  -0.904 0.366333    
## BldgTypeDuplex          -9.778e+03  5.290e+03  -1.848 0.064702 .  
## BldgTypeTwnhs           -3.281e+04  4.805e+03  -6.829 1.17e-11 ***
## BldgTypeTwnhsE          -2.633e+04  3.242e+03  -8.120 8.56e-16 ***
## HouseStyle1.5Unf         2.420e+03  7.672e+03   0.315 0.752518    
## HouseStyle1Story         2.689e+02  3.201e+03   0.084 0.933060    
## HouseStyle2.5Fin        -6.680e+03  1.322e+04  -0.505 0.613509    
## HouseStyle2.5Unf        -8.216e+02  7.093e+03  -0.116 0.907809    
## HouseStyle2Story        -2.559e+02  2.709e+03  -0.094 0.924765    
## HouseStyleSFoyer        -7.635e+02  4.698e+03  -0.163 0.870919    
## HouseStyleSLvl          -3.142e+03  4.035e+03  -0.779 0.436318    
## YearBuilt                1.014e+02  6.223e+01   1.630 0.103226    
## YearRemodAdd             2.293e+02  4.299e+01   5.335 1.08e-07 ***
## RoofStyleGable           6.241e+03  1.026e+04   0.608 0.542992    
## RoofStyleGambrel         5.263e+03  1.210e+04   0.435 0.663666    
## RoofStyleHip             9.114e+03  1.029e+04   0.885 0.376122    
## RoofStyleMansard         1.070e+04  1.463e+04   0.731 0.464798    
## RoofStyleShed            2.681e+04  1.498e+04   1.790 0.073599 .  
## RoofMatlOther            2.234e+03  6.935e+03   0.322 0.747375    
## Exterior1stMetalSd      -3.430e+03  7.028e+03  -0.488 0.625576    
## Exterior1stOther         7.347e+03  3.580e+03   2.052 0.040284 *  
## Exterior1stVinylSd      -1.209e+04  7.327e+03  -1.650 0.099182 .  
## Exterior1stWd Sdng      -2.111e+03  4.512e+03  -0.468 0.639957    
## Exterior2ndMetalSd       8.355e+03  7.041e+03   1.187 0.235517    
## Exterior2ndOther        -3.036e+03  3.545e+03  -0.856 0.391840    
## Exterior2ndVinylSd       1.525e+04  7.359e+03   2.072 0.038427 *  
## Exterior2ndWd Sdng       6.332e+03  4.589e+03   1.380 0.167758    
## MasVnrTypeBrkFace        5.037e+03  6.667e+03   0.755 0.450049    
## MasVnrTypeNone           5.767e+03  6.673e+03   0.864 0.387601    
## MasVnrTypeStone          3.461e+03  6.933e+03   0.499 0.617665    
## MasVnrArea               4.298e+00  5.231e+00   0.822 0.411420    
## ExterQualFa             -1.902e+04  8.283e+03  -2.296 0.021803 *  
## ExterQualGd             -1.110e+04  4.087e+03  -2.715 0.006681 ** 
## ExterQualTA             -2.006e+04  4.527e+03  -4.430 9.98e-06 ***
## ExterCondFa             -1.396e+04  1.118e+04  -1.248 0.212233    
## ExterCondGd             -4.832e+03  1.011e+04  -0.478 0.632622    
## ExterCondTA             -4.194e+03  1.006e+04  -0.417 0.676711    
## FoundationCBlock         4.958e+03  2.542e+03   1.950 0.051307 .  
## FoundationPConc          5.283e+03  2.828e+03   1.868 0.061925 .  
## FoundationSlab           6.251e+03  7.628e+03   0.819 0.412638    
## FoundationStone          3.633e+03  9.178e+03   0.396 0.692304    
## FoundationWood           7.703e+02  1.254e+04   0.061 0.951014    
## BsmtQualFa              -1.975e+04  4.962e+03  -3.981 7.14e-05 ***
## BsmtQualGd              -1.771e+04  2.817e+03  -6.286 4.07e-10 ***
## BsmtQualNo Basement     -2.763e+04  9.904e+03  -2.790 0.005326 **
## BsmtQualTA              -1.560e+04  3.503e+03  -4.454 8.93e-06 ***
## BsmtCondGd               3.513e+03  4.220e+03   0.832 0.405304
## BsmtCondNo Basement             NA         NA      NA       NA
## BsmtCondPo              -3.550e+02  1.783e+04  -0.020 0.984114
## BsmtCondTA               1.987e+03  3.266e+03   0.608 0.542935
## BsmtExposureGd           1.072e+04  2.555e+03   4.193 2.89e-05 ***
## BsmtExposureMn          -6.323e+03  2.564e+03  -2.466 0.013746 *
## BsmtExposureNo Basement         NA         NA      NA       NA
## BsmtExposureNo Exposure -4.749e+03  1.907e+03  -2.490 0.012858 *
## BsmtFinType1BLQ         -6.945e+02  2.341e+03  -0.297 0.766775
## BsmtFinType1GLQ          5.324e+03  2.119e+03   2.512 0.012088 *
## BsmtFinType1LwQ         -6.536e+03  2.935e+03  -2.227 0.026072 *
## BsmtFinType1No Basement         NA         NA      NA       NA
## BsmtFinType1Rec         -3.041e+03  2.407e+03  -1.263 0.206743
## BsmtFinType1Unf         -3.161e+03  2.423e+03  -1.304 0.192289
## BsmtFinSF1               1.083e+01  3.943e+00   2.748 0.006057 **
## BsmtFinType2BLQ         -2.744e+03  5.329e+03  -0.515 0.606691
## BsmtFinType2GLQ          3.599e+03  6.266e+03   0.574 0.565824
## BsmtFinType2LwQ         -4.531e+03  5.176e+03  -0.875 0.381506
## BsmtFinType2No Basement         NA         NA      NA       NA
## BsmtFinType2Rec         -2.265e+03  4.986e+03  -0.454 0.649663
## BsmtFinType2Unf          7.536e+02  5.028e+03   0.150 0.880878
## BsmtFinSF2               1.401e+01  6.868e+00   2.039 0.041547 *
## BsmtUnfSF                8.281e+00  3.673e+00   2.255 0.024268 *
## TotalBsmtSF                     NA         NA      NA       NA
## HeatingHotW              3.684e+03  6.128e+03   0.601 0.547788
## HeatingOther            -5.718e+02  9.070e+03  -0.063 0.949734
## HeatingQCFa             -7.305e+03  3.544e+03  -2.061 0.039442 *
## HeatingQCGd             -1.609e+03  1.704e+03  -0.944 0.345282
## HeatingQCTA             -4.376e+03  1.683e+03  -2.599 0.009413 **
## CentralAirY              1.661e+03  3.011e+03   0.552 0.581256
## ElectricalFF            -3.990e+01  4.859e+03  -0.008 0.993449
## ElectricalFP             7.554e+03  1.145e+04   0.660 0.509590
## ElectricalSB             3.966e+01  2.449e+03   0.016 0.987081
## X1stFlrSF                4.387e+01  4.207e+00  10.429  < 2e-16 ***
## X2ndFlrSF                4.113e+01  4.771e+00   8.621  < 2e-16 ***
## LowQualFinSF             2.020e+01  1.331e+01   1.517 0.129328
## GrLivArea                       NA         NA      NA       NA
## BsmtFullBath             6.939e+03  1.581e+03   4.389 1.21e-05 ***
## BsmtHalfBath            -1.128e+03  2.367e+03  -0.476 0.633849
## FullBath                 4.710e+03  1.821e+03   2.586 0.009783 **
## HalfBath                -7.456e+02  1.693e+03  -0.440 0.659707
## BedroomAbvGr             4.360e+02  1.103e+03   0.395 0.692567
## KitchenAbvGr            -1.196e+04  4.621e+03  -2.589 0.009693 **
## KitchenQualFa           -2.634e+04  5.315e+03  -4.955 7.92e-07 ***
## KitchenQualGd           -2.299e+04  3.207e+03  -7.170 1.09e-12 ***
## KitchenQualTA           -2.741e+04  3.550e+03  -7.721 1.91e-14 ***
## TotRmsAbvGrd            -2.736e+02  7.780e+02  -0.352 0.725121
## FunctionalMaj2          -2.363e+04  1.270e+04  -1.861 0.062964 .
## FunctionalMin1           4.314e+03  8.237e+03   0.524 0.600515    
## FunctionalMin2           2.858e+02  8.181e+03   0.035 0.972138    
## FunctionalMod           -3.856e+02  8.921e+03  -0.043 0.965529    
## FunctionalTyp            1.336e+04  7.400e+03   1.805 0.071181 .  
## Fireplaces               8.233e+03  2.242e+03   3.673 0.000247 ***
## FireplaceQuFa           -2.211e+04  5.698e+03  -3.881 0.000108 ***
## FireplaceQuGd           -1.648e+04  4.547e+03  -3.624 0.000298 ***
## FireplaceQuNo Fireplace -1.560e+04  5.286e+03  -2.952 0.003195 ** 
## FireplaceQuPo           -1.267e+04  6.688e+03  -1.894 0.058365 .  
## FireplaceQuTA           -1.763e+04  4.659e+03  -3.784 0.000159 ***
## GarageTypeA              1.774e+04  6.539e+03   2.713 0.006728 ** 
## GarageTypeBI             1.719e+04  7.156e+03   2.402 0.016395 *  
## GarageTypeBM             1.422e+04  8.279e+03   1.718 0.086021 .  
## GarageTypeCP             5.891e+03  9.526e+03   0.618 0.536416    
## GarageTypeD              1.255e+04  6.554e+03   1.915 0.055707 .  
## GarageTypeNone          -2.353e+05  9.418e+04  -2.499 0.012557 *  
## GarageYrBlt             -1.305e+02  4.873e+01  -2.679 0.007449 ** 
## GarageFinishNone                NA         NA      NA       NA    
## GarageFinishRFn         -7.442e+03  1.691e+03  -4.401 1.14e-05 ***
## GarageFinishUnf         -5.400e+03  2.007e+03  -2.691 0.007192 ** 
## GarageCars               9.267e+03  1.911e+03   4.850 1.34e-06 ***
## GarageArea               1.634e+01  6.968e+00   2.344 0.019169 *  
## GarageQualTA             3.165e+03  2.824e+03   1.121 0.262591    
## PavedDriveP             -1.757e+03  4.410e+03  -0.398 0.690339    
## PavedDriveY              1.357e+03  2.834e+03   0.479 0.632227    
## WoodDeckSF               1.186e+01  5.013e+00   2.366 0.018095 *  
## OpenPorchSF             -8.345e+00  9.283e+00  -0.899 0.368790    
## EnclosedPorch            1.303e+01  1.028e+01   1.268 0.205029    
## X3SsnPorch               3.304e+01  2.217e+01   1.490 0.136281    
## ScreenPorch              3.349e+01  1.008e+01   3.322 0.000910 ***
## PoolArea                 1.516e+03  1.493e+02  10.153  < 2e-16 ***
## PoolQCNo Pool            9.924e+05  9.212e+04  10.773  < 2e-16 ***
## PoolQCTA                 1.287e+05  3.064e+04   4.200 2.80e-05 ***
## FenceGdWo               -1.410e+03  4.091e+03  -0.345 0.730437    
## FenceMnPrv              -3.792e+03  3.213e+03  -1.180 0.238039    
## FenceMnWw               -3.097e+03  9.594e+03  -0.323 0.746854    
## FenceNo Fence           -3.780e+03  2.890e+03  -1.308 0.190940    
## MiscFeatureNo Feature    1.257e+03  3.037e+03   0.414 0.679002    
## MoSold                  -4.768e+01  2.062e+02  -0.231 0.817148    
## YrSold                  -4.858e+02  4.362e+02  -1.114 0.265620    
## SaleTypeOther           -7.106e+03  3.553e+03  -2.000 0.045670 *  
## SaleTypeWarranty Deed   -6.050e+03  2.470e+03  -2.449 0.014414 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 23040 on 1791 degrees of freedom
## Multiple R-squared:  0.901,  Adjusted R-squared:  0.8911 
## F-statistic: 90.57 on 180 and 1791 DF,  p-value: < 2.2e-16
```

```
mod_linear_olsrr <- ols_step_backward_p(mod_linear_intial, p_val = 0.05, progress = TRU
E)
```

```
## Backward Elimination Method
## ---------------------------
## 
## Candidate Terms:
## 
## 1. MSZoning
## 2. LotFrontage
## 3. LotArea
## 4. Street
## 5. Alley
## 6. LotShape
## 7. LandContour
## 8. LotConfig
## 9. LandSlope
## 10. Neighborhood
## 11. Condition1
## 12. Condition2
## 13. BldgType
## 14. HouseStyle
## 15. YearBuilt
## 16. YearRemodAdd
## 17. RoofStyle
## 18. RoofMatl
## 19. Exterior1st
## 20. Exterior2nd
## 21. MasVnrType
## 22. MasVnrArea
## 23. ExterQual
## 24. ExterCond
## 25. Foundation
## 26. BsmtQual
## 27. BsmtCond
## 28. BsmtExposure
## 29. BsmtFinType1
## 30. BsmtFinSF1
## 31. BsmtFinType2
## 32. BsmtFinSF2
## 33. BsmtUnfSF
## 34. TotalBsmtSF
## 35. Heating
## 36. HeatingQC
## 37. CentralAir
## 38. Electrical
## 39. X1stFlrSF
## 40. X2ndFlrSF
## 41. LowQualFinSF
## 42. GrLivArea
## 43. BsmtFullBath
## 44. BsmtHalfBath
## 45. FullBath
## 46. HalfBath
## 47. BedroomAbvGr
## 48. KitchenAbvGr
## 49. KitchenQual
## 50. TotRmsAbvGrd
## 51. Functional
## 52. Fireplaces
## 53. FireplaceQu
## 54. GarageType
## 55. GarageYrBlt
## 56. GarageFinish
## 57. GarageCars
## 58. GarageArea
## 59. GarageQual
## 60. PavedDrive
## 61. WoodDeckSF
## 62. OpenPorchSF
## 63. EnclosedPorch
## 64. X3SsnPorch
## 65. ScreenPorch
## 66. PoolArea
## 67. PoolQC
## 68. Fence
## 69. MiscFeature
## 70. MoSold
## 71. YrSold
## 72. SaleType
##
##
## Variables Removed:
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Street
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => MoSold
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => RoofMatl
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => TotRmsAbvGrd
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => BedroomAbvGr
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Electrical
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Heating
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => MiscFeature
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => HalfBath
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => CentralAir
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => LotShape
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => HouseStyle
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => BsmtCond
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => BsmtHalfBath
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => MasVnrType
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => MasVnrArea
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Fence
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => PavedDrive
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => MSZoning
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => BsmtFinType2
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Foundation
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => Alley
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => LotArea
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => OpenPorchSF
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => LotConfig
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => YrSold
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b * sx: longer object length is not a multiple of shorter object
## length
```

```
## => RoofStyle
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
## Warning in b ∗ sx: longer object length is not a multiple of shorter object
## length
```

```
## => Condition2
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b ∗ sx: longer object length is not a multiple of shorter object
## length
```

```
## => EnclosedPorch
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b ∗ sx: longer object length is not a multiple of shorter object
## length
```

```
## => X3SsnPorch
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b ∗ sx: longer object length is not a multiple of shorter object
## length
```

```
## => GarageQual
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
## Warning in b ∗ sx: longer object length is not a multiple of shorter object
## length
```

```
## => ExterCond
```

```
## Note: model has aliased coefficients
##       sums of squares computed by model comparison
```

```
##
## No more variables to be removed.
```

```
mod_linear_final <- mod_linear_olsrr$model
summary(mod_linear_final)
```

```
## 
## Call:
## lm(formula = paste(response, "~", paste(c(include, cterms), collapse = " + ")), 
##     data = l)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -311695  -10836     -12   10820  130403 
## 
## Coefficients: (5 not defined because of singularities)
##                           Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             -1.380e+06  1.613e+05  -8.556  < 2e-16 ***
## LotFrontage             -1.126e+02  3.877e+01  -2.904 0.003722 ** 
## LandContourHLS           1.915e+04  4.065e+03   4.712 2.63e-06 ***
## LandContourLow           6.996e+03  5.042e+03   1.388 0.165435    
## LandContourLvl           1.233e+04  2.949e+03   4.181 3.03e-05 ***
## LandSlopeNot Gtl         1.027e+04  3.157e+03   3.252 0.001168 ** 
## NeighborhoodBlueste     -3.016e+03  1.223e+04  -0.247 0.805276    
## NeighborhoodBrDale       1.363e+04  8.777e+03   1.553 0.120491    
## NeighborhoodBrkSide     -7.414e+03  7.324e+03  -1.012 0.311506    
## NeighborhoodClearCr      1.071e+04  7.804e+03   1.372 0.170333    
## NeighborhoodCollgCr      3.121e+03  6.344e+03   0.492 0.622784    
## NeighborhoodCrawfor      2.171e+04  7.096e+03   3.059 0.002254 ** 
## NeighborhoodEdwards     -1.824e+04  6.751e+03  -2.702 0.006965 ** 
## NeighborhoodGilbert     -6.415e+03  6.523e+03  -0.983 0.325541    
## NeighborhoodGreens       1.243e+04  1.343e+04   0.926 0.354797    
## NeighborhoodIDOTRR      -1.205e+04  7.549e+03  -1.597 0.110533    
## NeighborhoodMeadowV     -1.432e+04  7.940e+03  -1.804 0.071419 .  
## NeighborhoodMitchel     -6.924e+03  6.783e+03  -1.021 0.307530    
## NeighborhoodNAmes       -6.920e+03  6.657e+03  -1.040 0.298703    
## NeighborhoodNoRidge      4.115e+04  7.134e+03   5.768 9.40e-09 ***
## NeighborhoodNPkVill      1.416e+04  8.923e+03   1.587 0.112595    
## NeighborhoodNridgHt      4.279e+04  6.551e+03   6.532 8.36e-11 ***
## NeighborhoodNWAmes      -2.902e+03  6.855e+03  -0.423 0.672064    
## NeighborhoodOldTown     -1.385e+04  7.103e+03  -1.950 0.051347 .  
## NeighborhoodSawyer      -6.016e+03  6.874e+03  -0.875 0.381581    
## NeighborhoodSawyerW     -1.484e+03  6.612e+03  -0.224 0.822474    
## NeighborhoodSomerst      2.036e+04  6.296e+03   3.234 0.001241 ** 
## NeighborhoodStoneBr      4.217e+04  7.359e+03   5.730 1.17e-08 ***
## NeighborhoodSWISU       -1.208e+04  7.936e+03  -1.522 0.128178    
## NeighborhoodTimber       1.179e+04  6.912e+03   1.705 0.088300 .  
## NeighborhoodVeenker      3.512e+04  9.046e+03   3.882 0.000107 ***
## Condition1Feedr          2.651e+03  4.016e+03   0.660 0.509246    
## Condition1Norm           1.154e+04  3.341e+03   3.453 0.000567 ***
## Condition1PosA           2.593e+04  7.001e+03   3.703 0.000219 ***
## Condition1PosN           1.298e+04  5.640e+03   2.301 0.021512 *  
## Condition1RRAe           2.077e+02  6.489e+03   0.032 0.974463    
## Condition1RRAn           3.297e+03  5.473e+03   0.602 0.546950    
## Condition1RRNe           1.628e+03  1.106e+04   0.147 0.882978    
## Condition1RRNn           8.799e+03  1.110e+04   0.793 0.427956    
## BldgType2fmCon          -5.139e+03  4.473e+03  -1.149 0.250782    
## BldgTypeDuplex          -9.638e+03  4.856e+03  -1.985 0.047326 *  
## BldgTypeTwnhs          -3.403e+04  4.435e+03  -7.674 2.68e-14 ***
## BldgTypeTwnhsE         -2.741e+04  2.872e+03  -9.545  < 2e-16 ***
## YearBuilt               1.588e+02  5.222e+01   3.041 0.002394 **
## YearRemodAdd            2.268e+02  4.045e+01   5.607 2.36e-08 ***
## Exterior1stMetalSd     -3.585e+03  6.875e+03  -0.522 0.602066
## Exterior1stOther        6.388e+03  3.482e+03   1.835 0.066720 .
## Exterior1stVinylSd     -1.260e+04  7.060e+03  -1.785 0.074384 .
## Exterior1stWd Sdng     -1.980e+03  4.389e+03  -0.451 0.652033
## Exterior2ndMetalSd      8.563e+03  6.872e+03   1.246 0.212913
## Exterior2ndOther       -1.972e+03  3.451e+03  -0.571 0.567786
## Exterior2ndVinylSd      1.548e+04  7.107e+03   2.178 0.029547 *
## Exterior2ndWd Sdng      6.260e+03  4.459e+03   1.404 0.160522
## ExterQualFa            -2.368e+04  7.579e+03  -3.125 0.001806 **
## ExterQualGd            -1.119e+04  3.972e+03  -2.818 0.004881 **
## ExterQualTA            -2.048e+04  4.378e+03  -4.677 3.13e-06 ***
## BsmtQualFa             -1.946e+04  4.734e+03  -4.111 4.10e-05 ***
## BsmtQualGd             -1.771e+04  2.727e+03  -6.492 1.08e-10 ***
## BsmtQualNo Basement    -2.751e+04  6.527e+03  -4.215 2.62e-05 ***
## BsmtQualTA             -1.531e+04  3.338e+03  -4.587 4.81e-06 ***
## BsmtExposureGd          1.083e+04  2.479e+03   4.367 1.33e-05 ***
## BsmtExposureMn         -5.892e+03  2.411e+03  -2.444 0.014610 *
## BsmtExposureNo Basement        NA         NA      NA       NA
## BsmtExposureNo Exposure -4.508e+03  1.709e+03  -2.637 0.008426 **
## BsmtFinType1BLQ        -2.425e+02  2.247e+03  -0.108 0.914074
## BsmtFinType1GLQ         4.403e+03  2.027e+03   2.172 0.030014 *
## BsmtFinType1LwQ        -5.687e+03  2.783e+03  -2.043 0.041146 *
## BsmtFinType1No Basement        NA         NA      NA       NA
## BsmtFinType1Rec        -2.398e+03  2.293e+03  -1.046 0.295675
## BsmtFinType1Unf        -3.488e+03  2.313e+03  -1.508 0.131748
## BsmtFinSF1              1.085e+01  3.660e+00   2.964 0.003075 **
## BsmtFinSF2              1.052e+01  4.839e+00   2.173 0.029900 *
## BsmtUnfSF               8.669e+00  3.394e+00   2.554 0.010729 *
## TotalBsmtSF                    NA         NA      NA       NA
## HeatingQCFa            -9.912e+03  3.226e+03  -3.072 0.002155 **
## HeatingQCGd            -1.881e+03  1.657e+03  -1.135 0.256552
## HeatingQCTA            -4.831e+03  1.610e+03  -3.000 0.002740 **
## X1stFlrSF               4.509e+01  3.556e+00  12.682  < 2e-16 ***
## X2ndFlrSF               4.055e+01  2.048e+00  19.801  < 2e-16 ***
## LowQualFinSF            1.584e+01  1.186e+01   1.336 0.181847
## GrLivArea                      NA         NA      NA       NA
## BsmtFullBath            7.627e+03  1.435e+03   5.316 1.19e-07 ***
## FullBath                4.151e+03  1.575e+03   2.635 0.008474 **
## KitchenAbvGr           -1.253e+04  4.232e+03  -2.961 0.003110 **
## KitchenQualFa          -2.714e+04  5.059e+03  -5.364 9.14e-08 ***
## KitchenQualGd          -2.289e+04  3.132e+03  -7.307 4.04e-13 ***
## KitchenQualTA          -2.700e+04  3.457e+03  -7.810 9.46e-15 ***
## FunctionalMaj2         -2.492e+04  1.202e+04  -2.073 0.038282 *
## FunctionalMin1          3.770e+03  7.911e+03   0.476 0.633784
## FunctionalMin2         -1.472e+03  7.770e+03  -0.189 0.849799
## FunctionalMod          -3.358e+03  8.544e+03  -0.393 0.694328
## FunctionalTyp           1.288e+04  7.080e+03   1.819 0.069083 .
## Fireplaces              8.399e+03  2.161e+03   3.886 0.000106 ***
## FireplaceQuFa          -2.154e+04  5.565e+03  -3.871 0.000112 ***
## FireplaceQuGd          -1.556e+04  4.452e+03  -3.495 0.000486 ***
## FireplaceQuNo Fireplace -1.530e+04  5.156e+03  -2.967 0.003041 **
## FireplaceQuPo          -1.282e+04  6.540e+03  -1.960 0.050145 .
## FireplaceQuTA          -1.714e+04  4.567e+03  -3.753 0.000180 ***
## GarageTypeA             1.699e+04  6.257e+03   2.716 0.006667 **
## GarageTypeBI            1.574e+04  6.750e+03   2.332 0.019794 *
## GarageTypeBM            1.193e+04  7.967e+03   1.498 0.134372
## GarageTypeCP            2.455e+03  9.143e+03   0.269 0.788321
## GarageTypeD             1.127e+04  6.275e+03   1.796 0.072728 .
## GarageTypeNone         -2.317e+05  8.915e+04  -2.599 0.009426 **
## GarageYrBlt            -1.269e+02  4.595e+01  -2.761 0.005819 **
## GarageFinishNone               NA         NA      NA       NA
## GarageFinishRFn        -7.585e+03  1.642e+03  -4.618 4.13e-06 ***
## GarageFinishUnf        -5.435e+03  1.940e+03  -2.802 0.005129 **
## GarageCars              9.235e+03  1.822e+03   5.069 4.40e-07 ***
## GarageArea              1.753e+01  6.689e+00   2.621 0.008845 **
## WoodDeckSF              1.126e+01  4.796e+00   2.347 0.019011 *
## ScreenPorch             3.098e+01  9.800e+00   3.161 0.001600 **
## PoolArea                1.504e+03  1.393e+02  10.802  < 2e-16 ***
## PoolQCNo Pool           9.862e+05  8.637e+04  11.419  < 2e-16 ***
## PoolQCTA                1.292e+05  2.935e+04   4.401 1.14e-05 ***
## SaleTypeOther          -7.814e+03  3.425e+03  -2.282 0.022625 *
## SaleTypeWarranty Deed  -5.956e+03  2.365e+03  -2.518 0.011883 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22930 on 1860 degrees of freedom
## Multiple R-squared:  0.8983, Adjusted R-squared:  0.8922
## F-statistic: 147.9 on 111 and 1860 DF,  p-value: < 2.2e-16
```

```
# out of sample Rsqd calc
out_of_sample_predictions <- predict(mod_linear_final, newdata = test_ames)

# calculate out-of-sample R-squared
out_of_sample_r_squared <- 1 - (sum((test_ames$SalePrice - out_of_sample_predictions)^2)
/
                                sum((test_ames$SalePrice - mean(test_ames$SalePrice))^
2))
out_of_sample_r_squared
```

```
## [1] 0.825442
```

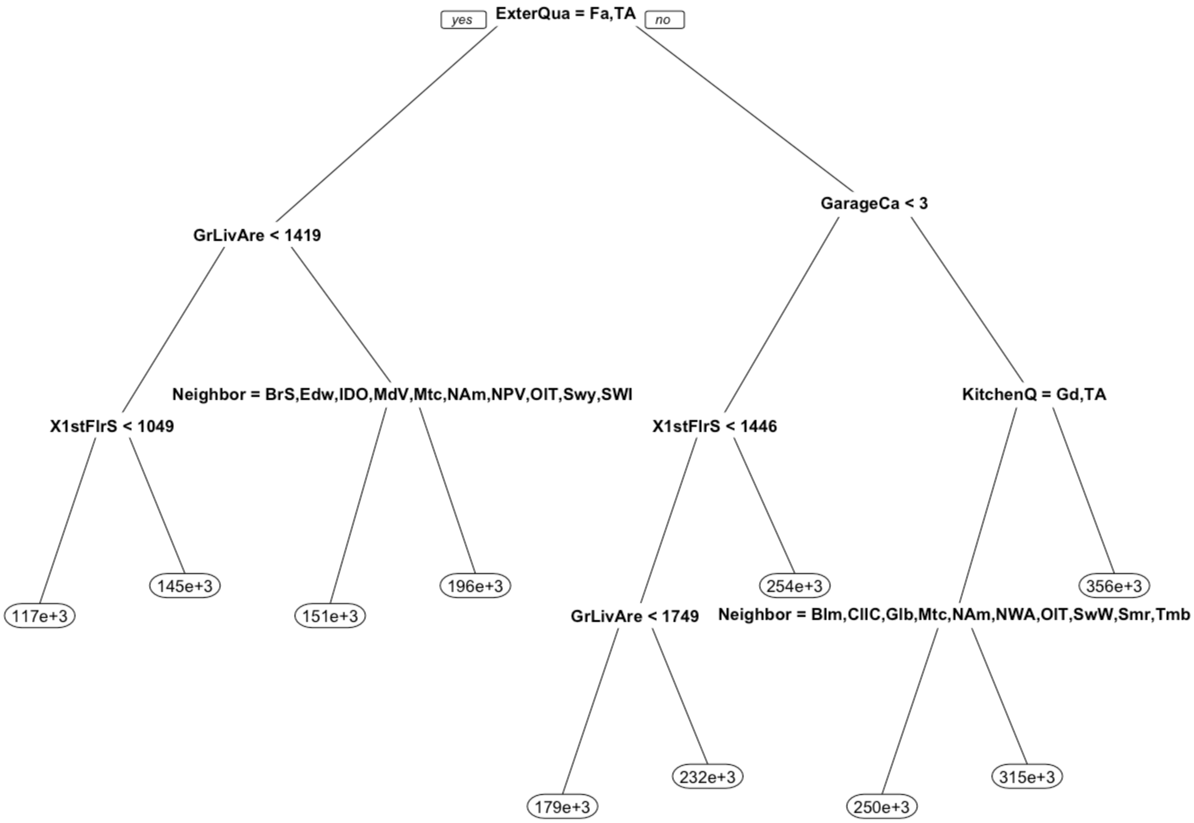
For mod_linear_final: In-sample R-squared (taken from outputted summary): 0.8983 Out-of-sample R-squared: 0.8254482

Part b

```
library(rpart)
modelcart = rpart(data = train_ames, SalePrice ~ .)
modelcart
```

```
## n= 1972
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 1972 9.608319e+12 178139.2
##    2) ExterQual=Fa,TA 1245 1.900199e+12 143154.8
##      4) GrLivArea< 1419 785 5.444631e+11 127737.4
##        8) X1stFlrSF< 1049 481 2.328889e+11 116714.7 *
##        9) X1stFlrSF>=1049 304 1.606640e+11 145178.0 *
##      5) GrLivArea>=1419 460 8.507225e+11 169464.9
##       10) Neighborhood=BrkSide,Edwards,IDOTRR,MeadowV,Mitchel,NAmes,NPkVill,OldTown,S
awyer,SWISU 274 3.673466e+11 151150.2 *
##       11) Neighborhood=ClearCr,CollgCr,Crawfor,Gilbert,NridgHt,NWAmes,SawyerW,Somers
t,Timber,Veenker 186 2.560787e+11 196444.6 *
##    3) ExterQual=Ex,Gd 727 3.574869e+12 238050.7
##      6) GarageCars< 2.5 510 1.269908e+12 209839.7
##       12) X1stFlrSF< 1446 373 5.675778e+11 193570.8
##         24) GrLivArea< 1748.5 269 2.177148e+11 178686.4 *
##         25) GrLivArea>=1748.5 104 1.361211e+11 232069.8 *
##       13) X1stFlrSF>=1446 137 3.348141e+11 254134.0 *
##      7) GarageCars>=2.5 217 9.451414e+11 304353.0
##       14) KitchenQual=Gd,TA 142 3.937172e+11 277137.0
##         28) Neighborhood=Blmngtn,CollgCr,Gilbert,Mitchel,NAmes,NWAmes,OldTown,Sawyer
W,Somerst,Timber 83 1.250205e+11 249986.9 *
##         29) Neighborhood=NoRidge,NridgHt,StoneBr,Veenker 59 1.214454e+11 315331.3 *
##       15) KitchenQual=Ex 75 2.471010e+11 355881.9 *
```

```
library(rpart.plot)
prp(modelcart)
```



```
# in-sample predictions
in_sample_predictions <- predict(modelcart, newdata = train_ames)

# calculate in-sample R-squared
in_sample_r_squared <- 1 - (sum((train_ames$SalePrice - in_sample_predictions)^2) /
                            sum((train_ames$SalePrice - mean(train_ames$SalePrice))^2))

# out-of-sample predictions
out_of_sample_predictions <- predict(modelcart, newdata = test_ames)

# calculate out-of-sample R-squared
out_of_sample_r_squared <- 1 - (sum((test_ames$SalePrice - out_of_sample_predictions)^2)
/
                                sum((test_ames$SalePrice - mean(test_ames$SalePrice))^
2))

cat("In-sample R²:", round(in_sample_r_squared, 4), "\n")
```

```
## In-sample R²: 0.7711
```

```
cat("Out-of-sample R²:", round(out_of_sample_r_squared, 4), "\n")
```

```
## Out-of-sample R²: 0.691
```

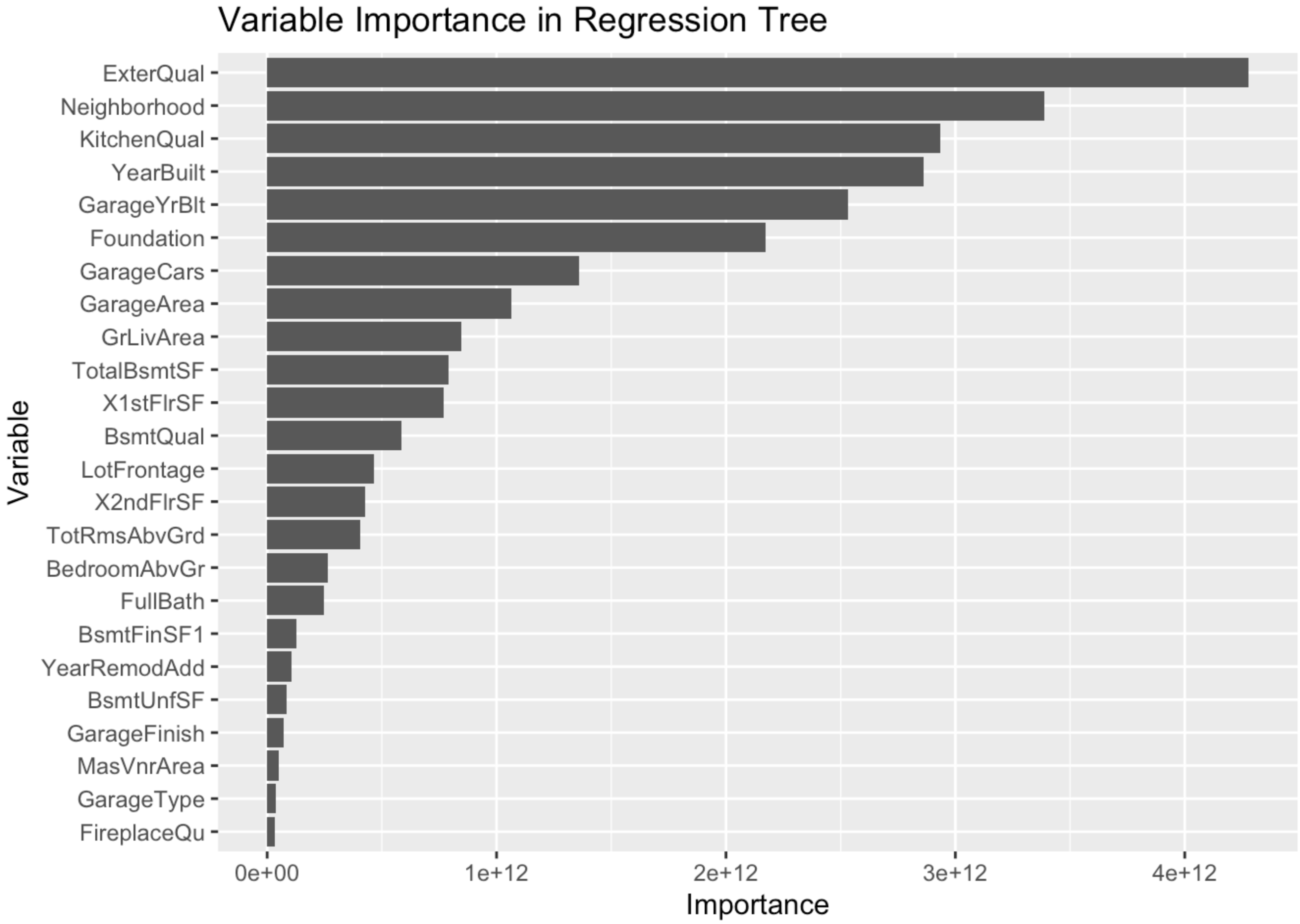
```
importance_scores <- modelcart$variable.importance
print(importance_scores)
```

```
##    ExterQual Neighborhood  KitchenQual    YearBuilt  GarageYrBlt   Foundation 
## 4.279327e+12 3.389753e+12 2.936123e+12 2.862491e+12 2.532263e+12 2.174458e+12 
##   GarageCars   GarageArea    GrLivArea  TotalBsmtSF    X1stFlrSF     BsmtQual 
## 1.359819e+12 1.065296e+12 8.466353e+11 7.909058e+11 7.674546e+11 5.851225e+11 
##  LotFrontage    X2ndFlrSF TotRmsAbvGrd BedroomAbvGr     FullBath   BsmtFinSF1 
## 4.642459e+11 4.272400e+11 4.075687e+11 2.623970e+11 2.459194e+11 1.287648e+11 
## YearRemodAdd    BsmtUnfSF GarageFinish   MasVnrArea   GarageType  FireplaceQu 
## 1.048548e+11 8.584321e+10 7.209965e+10 4.991570e+10 3.772757e+10 3.493857e+10
```

```
sorted_importance <- sort(importance_scores, decreasing = TRUE)
print(sorted_importance)
```

```
##    ExterQual Neighborhood  KitchenQual    YearBuilt  GarageYrBlt   Foundation 
## 4.279327e+12 3.389753e+12 2.936123e+12 2.862491e+12 2.532263e+12 2.174458e+12 
##   GarageCars   GarageArea    GrLivArea  TotalBsmtSF    X1stFlrSF     BsmtQual 
## 1.359819e+12 1.065296e+12 8.466353e+11 7.909058e+11 7.674546e+11 5.851225e+11 
##  LotFrontage    X2ndFlrSF TotRmsAbvGrd BedroomAbvGr     FullBath   BsmtFinSF1 
## 4.642459e+11 4.272400e+11 4.075687e+11 2.623970e+11 2.459194e+11 1.287648e+11 
## YearRemodAdd    BsmtUnfSF GarageFinish   MasVnrArea   GarageType  FireplaceQu 
## 1.048548e+11 8.584321e+10 7.209965e+10 4.991570e+10 3.772757e+10 3.493857e+10
```

```
library(ggplot2)
importance_df <- as.data.frame(sorted_importance)
importance_df$variable <- rownames(importance_df)
ggplot(importance_df, aes(x = reorder(variable, sorted_importance), y = sorted_importanc
e)) +
  geom_col() +
  coord_flip() +
  labs(x = "Variable", y = "Importance", title = "Variable Importance in Regression Tre
e")
```



In sample R^2 is 0.7711155 and out of sample R^2 is 0.6909755. We have included our decision tree visualization above, and we can see from this plot of feature importances that the 5 most important variables are ExterQual, Neighborhood, KitchenQual, YearBuilt, and GarageYrBlt. This makes sense because the initial splits in our tree do occur based on ExterQua, GrLivAre, and GarageCa, and Neighbor shows up as well in following splits. These features must provide the most information gain in the tree, hence they are some of the most important features that tell us more about how to predict SalePrice based on other attributes provided.

Part c

```
coef(mod_linear_intial)["CentralAir"]
```

```
## <NA>
##   NA
```

In order to see if it is worth it to have central air installed in order to increase the value of her home, we would want to compare the cost of installation ($15,000) to the predicted value added when a home has central air but all other factors remain constant. If the predicted value add is more than the cost, then yes it is worth it. Else, it is not going to be enough of a reason to spend the cost on the install. When we try to find the coefficient for “CentralAir” from our initial linear model, we get NA. When we try to see if “CentralAir” was one of the important features in our plot of Important Features in our decision tree, we see it doesn’t show up. Meaning, “CentralAir” is not being valued in either of our models - the noninclusion suggests 0 value add. One thing I noticed was that, during the construction of models in part a, I was getting the error of aliased coefficients. I investigated further and found that a model has “aliased coefficients” when there is linear dependency among its predictor variables, meaning one or more predictors can be perfectly expressed as a linear combination of others, resulting in redundant information and making it impossible to uniquely estimate their coefficients. It indicates potential multicollinearity issues that require addressing by removing redundant variables or using other modeling techniques. However, we were told to assume no multicollinearity, which is obviously not the case in this dataset.

Part d

```
# defining the cp values to evaluate
cp_values <- c(5e-6, 5e-5, 5e-4, 5e-3, 5e-2)
tune_grid <- expand.grid(cp = cp_values)

# set up 10-fold cross-validation
ctrl <- trainControl(method = "cv", number = 10)

# perform cross-validation to find the best cp
set.seed(42)
tree_model_cv <- train(
  SalePrice ~ .,
  data = train_ames,
  method = "rpart",
  trControl = ctrl,
  tuneGrid = tune_grid
)

# result of cross validation
print(tree_model_cv)
```

```
## CART
##
## 1972 samples
##   72 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1775, 1775, 1776, 1774, 1774, 1776, ...
## Resampling results across tuning parameters:
##
##   cp     RMSE      Rsquared   MAE
##   5e-06  31986.16  0.7903454  21667.53
##   5e-05  31979.45  0.7904077  21687.35
##   5e-04  32106.24  0.7875481  21791.22
##   5e-03  35342.98  0.7451652  25173.56
##   5e-02  44268.82  0.5986164  32732.44
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was cp = 5e-05.
```

```
# reporting the best cp value selected by cross-validation based on lowest rmse
best_cp <- tree_model_cv$bestTune$cp
cat("The optimal cp value is:", best_cp, "\n")
```

```
## The optimal cp value is: 5e-05
```

```
# define the final model using the best cp
final_model <- rpart(SalePrice ~ ., data = train_ames, cp = best_cp)
print(final_model)
```

```
## n= 1972
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##    1) root 1972 9.608319e+12 178139.20
##      2) ExterQual=Fa,TA 1245 1.900199e+12 143154.80
##        4) GrLivArea< 1419 785 5.444631e+11 127737.40
##          8) X1stFlrSF< 1049 481 2.328889e+11 116714.70
##           16) Neighborhood=BrDale,BrkSide,Edwards,IDOTRR,MeadowV,OldTown,SawyerW,SWIS
U 265 1.076367e+11 106900.90
##             32) GrLivArea< 1146 170 5.917118e+10  99635.68
##               64) TotalBsmtSF< 685 87 2.257950e+10  92467.70
##                128) GrLivArea< 952 40 5.243056e+09  84340.00
##                  256) Neighborhood=BrkSide,IDOTRR,MeadowV 20 1.435466e+09  79685.00 *
##                  257) Neighborhood=Edwards,OldTown,SWISU 20 2.940830e+09  88995.00 *
##                129) GrLivArea>=952 47 1.244523e+10  99384.89
##                  258) Neighborhood=BrkSide,MeadowV 16 1.338001e+09  89111.88 *
##                  259) Neighborhood=BrDale,Edwards,IDOTRR,OldTown 31 8.547155e+09 1046
87.10
##                    518) GrLivArea< 995.5 12 2.040862e+09  96025.00 *
##                    519) GrLivArea>=995.5 19 5.037246e+09 110157.90 *
##               65) TotalBsmtSF>=685 83 2.743616e+10 107149.10
##                130) PavedDrive=N,P 24 7.332318e+09  93667.71
##                  260) MoSold>=8.5 7 6.045305e+08  79896.43 *
##                  261) MoSold< 8.5 17 4.853618e+09  99338.24 *
##                131) PavedDrive=Y 59 1.396754e+10 112633.10
##                  262) YearRemodAdd< 1962 34 7.175998e+09 107064.70
##                    524) BsmtFinType1=ALQ,LwQ,Unf 23 4.352235e+09 102360.90
##                     1048) YrSold>=2008.5 8 1.514859e+09  92162.50 *
##                     1049) YrSold< 2008.5 15 1.561560e+09 107800.00 *
##                    525) BsmtFinType1=BLQ,GLQ,Rec 11 1.250800e+09 116900.00 *
##                  263) YearRemodAdd>=1962 25 4.303582e+09 120206.00
##                    526) GarageYrBlt< 1957.5 14 1.645409e+09 112607.10 *
##                    527) GarageYrBlt>=1957.5 11 8.209068e+08 129877.30 *
##             33) GrLivArea>=1146 95 2.343486e+10 119901.90
##               66) KitchenQual=Fa,TA 80 1.659940e+10 117384.80
##                132) GarageYrBlt< 1934 21 3.071808e+09 107935.90
##                  264) LotFrontage< 51 8 7.546400e+08  99050.00 *
##                  265) LotFrontage>=51 13 1.296771e+09 113404.20 *
##                133) GarageYrBlt>=1934 59 1.098537e+10 120747.90
##                  266) GrLivArea< 1379 52 7.050902e+09 118198.60
##                    532) HeatingQC=Fa,Gd,TA 36 4.431531e+09 115261.80
##                     1064) GrLivArea< 1338 27 2.576234e+09 112145.40
##                       2128) YearRemodAdd< 1977.5 19 1.677012e+09 109311.90 *
##                       2129) YearRemodAdd>=1977.5 8 3.843750e+08 118875.00 *
##                     1065) GrLivArea>=1338 9 8.063889e+08 124611.10 *
##                    533) HeatingQC=Ex 16 1.610309e+09 124806.20 *
##                  267) GrLivArea>=1379 7 1.086029e+09 139685.70 *
##               67) KitchenQual=Gd 15 3.625209e+09 133326.70 *
##           17) Neighborhood=Blueste,ClearCr,CollgCr,Crawfor,Gilbert,Mitchel,NAmes,NPkV
ill,NWAmes,Sawyer,Somerst,Timber 216 6.841824e+10 128754.70
```

```
##             34) HouseStyle=1.5Unf,1Story 153 3.940272e+10 123441.70
##               68) TotalBsmtSF< 776.5 19 2.222815e+09 101400.00 *
##               69) TotalBsmtSF>=776.5 134 2.664019e+10 126567.00
##                138) BsmtFullBath< 0.5 72 1.485336e+10 121405.90
##                  276) YearRemodAdd< 1975.5 53 1.037829e+10 118127.80
##                    552) X1stFlrSF< 1007 46 8.371702e+09 115897.80
##                     1104) GarageFinish=None,RFn 8 1.819469e+09 106687.50 *
##                     1105) GarageFinish=Unf 38 5.730725e+09 117836.80
##                       2210) BsmtFinSF2>=154 8 3.080425e+09 106985.40 *
##                       2211) BsmtFinSF2< 154 30 1.457062e+09 120730.50
##                         4422) BsmtUnfSF>=678.5 11 3.348141e+08 114790.90 *
##                         4423) BsmtUnfSF< 678.5 19 5.095047e+08 124169.30 *
##                    553) X1stFlrSF>=1007 7 2.745834e+08 132782.10 *
##                  277) YearRemodAdd>=1975.5 19 2.316865e+09 130550.00 *
##                139) BsmtFullBath>=0.5 62 7.641811e+09 132560.50
##                  278) YearRemodAdd< 1966.5 27 1.641385e+09 127253.70
##                    556) BsmtFinType1=BLQ,LwQ,Rec 18 8.423311e+08 124122.20 *
##                    557) BsmtFinType1=ALQ,GLQ 9 2.695200e+08 133516.70 *
##                  279) YearRemodAdd>=1966.5 35 4.653482e+09 136654.30
##                    558) TotalBsmtSF< 959.5 21 1.885678e+09 132073.80 *
##                    559) TotalBsmtSF>=959.5 14 1.666314e+09 143525.00 *
##             35) HouseStyle=1.5Fin,2Story,SFoyer,SLvl 63 1.420769e+10 141657.80
##               70) Neighborhood=Blueste,Mitchel,NAmes,Sawyer 41 6.721785e+09 135721.00
##                140) LotFrontage< 61 13 1.500369e+09 124707.70 *
##                141) LotFrontage>=61 28 2.912528e+09 140834.30
##                  282) GarageType=D,None 9 1.460094e+08 132095.60 *
##                  283) GarageType=2T,A,BI,BM 19 1.753672e+09 144973.70 *
##               71) Neighborhood=ClearCr,CollgCr,Crawfor,Gilbert,NPkVill,NWAmes,Somers
t,Timber 22 3.347702e+09 152721.90
##                142) GarageType=D 12 6.820892e+08 144308.30 *
##                143) GarageType=A,BI 10 7.968026e+08 162818.20 *
##          9) X1stFlrSF>=1049 304 1.606640e+11 145178.00
##           18) Neighborhood=BrkSide,Crawfor,Edwards,IDOTRR,MeadowV,NAmes,NPkVill,OldTo
wn,Sawyer,SawyerW,SWISU 236 9.176543e+10 139985.40
##             36) YearBuilt< 1953.5 41 1.257992e+10 121856.60
##               72) LotFrontage< 79.5 31 4.707429e+09 114568.40
##                144) YearBuilt< 1924.5 10 8.649676e+08 105942.10 *
##                145) YearBuilt>=1924.5 21 2.743978e+09 118676.20
##                  290) BsmtFinType1=GLQ,LwQ,No Basement,Rec 10 7.167890e+08 112390.00
*
##                  291) BsmtFinType1=ALQ,BLQ,Unf 11 1.272789e+09 124390.90 *
##               73) LotFrontage>=79.5 10 1.121225e+09 144450.00 *
##             37) YearBuilt>=1953.5 195 6.287762e+10 143797.10
##               74) TotalBsmtSF< 472 11 3.389282e+09 112027.30 *
##               75) TotalBsmtSF>=472 184 4.772208e+10 145696.40
##                150) X1stFlrSF< 1187.5 106 2.130203e+10 140895.90
##                  300) LotArea< 10082 77 1.375078e+10 137484.70
##                    600) YearBuilt< 1977.5 61 1.018108e+10 134775.70
##                     1200) Neighborhood=Edwards,OldTown,SawyerW 12 1.749682e+09 12147
5.00 *
##                     1201) Neighborhood=Crawfor,IDOTRR,MeadowV,NAmes,NPkVill,Sawyer 49
5.788586e+09 138033.10
```

```
##                        2402) BsmtFinType1=BLQ,LwQ,Unf 15 9.198066e+08 130504.70 *
##                        2403) BsmtFinType1=ALQ,GLQ,Rec 34 3.643562e+09 141354.40
##                          4806) LotArea< 8071 14 7.326609e+08 135489.30 *
##                          4807) LotArea>=8071 20 2.092188e+09 145460.00
##                            9614) BsmtUnfSF>=404 7 3.715486e+08 138214.30 *
##                            9615) BsmtUnfSF< 404 13 1.155251e+09 149361.50 *
##                     601) YearBuilt>=1977.5 16 1.415438e+09 147812.50 *
##                   301) LotArea>=10082 29 4.276110e+09 149953.40
##                     602) BsmtFinSF1< 511.5 10 3.750360e+08 138920.00 *
##                     603) BsmtFinSF1>=511.5 19 2.042983e+09 155760.50 *
##                 151) X1stFlrSF>=1187.5 78 2.065788e+10 152220.00
##                   302) ScreenPorch< 132 68 1.747439e+10 150281.80
##                     604) FullBath< 1.5 60 1.383249e+10 148209.30
##                      1208) MoSold< 5.5 29 5.328240e+09 143472.30
##                        2416) BsmtFinSF1>=818.5 7 2.535733e+09 132749.60 *
##                        2417) BsmtFinSF1< 818.5 22 1.731582e+09 146884.10
##                          4834) YearBuilt< 1958.5 8 3.209522e+08 139306.20 *
##                          4835) YearBuilt>=1958.5 14 6.887321e+08 151214.30 *
##                      1209) MoSold>=5.5 31 7.244762e+09 152640.70
##                        2418) Condition1=Norm 23 4.077532e+09 148552.20
##                          4836) BsmtFinType1=LwQ,Rec,Unf 15 2.217732e+09 144246.70 *
##                          4837) BsmtFinType1=ALQ,BLQ 8 1.060375e+09 156625.00 *
##                        2419) Condition1=Feedr,PosA,PosN 8 1.677405e+09 164395.20 *
##                     605) FullBath>=1.5 8 1.451455e+09 165825.00 *
##                   303) ScreenPorch>=132 10 1.190900e+09 165400.00 *
##            19) Neighborhood=ClearCr,CollgCr,Gilbert,Mitchel,NWAmes,Timber 68 4.045119e
+10 163199.20
##              38) BsmtExposure=Av,Mn,No Exposure 61 2.654873e+10 159246.70
##                76) GarageArea< 334.5 8 3.448840e+09 134800.00 *
##                77) GarageArea>=334.5 53 1.759708e+10 162936.80
##                 154) KitchenQual=TA 40 1.253121e+10 158918.70
##                   308) BsmtFinSF1< 836 26 8.924332e+09 154476.90
##                     616) LotFrontage>=82.5 7 4.787336e+09 143314.90 *
##                     617) LotFrontage< 82.5 19 2.943547e+09 158589.20 *
##                   309) BsmtFinSF1>=836 14 2.141228e+09 167167.90 *
##                 155) KitchenQual=Ex,Gd 13 2.433040e+09 175300.00 *
##              39) BsmtExposure=Gd 7 4.644937e+09 197642.90 *
##        5) GrLivArea>=1419 460 8.507225e+11 169464.90
##         10) Neighborhood=BrkSide,Edwards,IDOTRR,MeadowV,Mitchel,NAmes,NPkVill,OldTow
n,Sawyer,SWISU 274 3.673466e+11 151150.20
##           20) LotArea< 12561.5 221 1.569281e+11 143602.90
##             40) GarageQual=Other 50 2.758929e+10 120574.40
##               80) LotFrontage>=87.5 7 3.201500e+09  96500.00 *
##               81) LotFrontage< 87.5 43 1.967031e+10 124493.50
##                162) GarageYrBlt< 1948.5 36 1.536284e+10 121036.60
##                  324) YearBuilt< 1910.5 12 5.134983e+09 107155.80 *
##                  325) YearBuilt>=1910.5 24 6.759635e+09 127977.10
##                    650) TotalBsmtSF< 895 7 1.398412e+09 112207.10 *
##                    651) TotalBsmtSF>=895 17 2.903570e+09 134470.60 *
##                163) GarageYrBlt>=1948.5 7 1.664894e+09 142271.40 *
##             41) GarageQual=TA 171 9.507018e+10 150336.30
##               82) YearRemodAdd< 1961 59 2.011788e+10 138366.90
##                164) LotFrontage< 51.5 10 3.789469e+09 123790.00 *
##                165) LotFrontage>=51.5 49 1.376989e+10 141341.80
##                  330) HalfBath< 0.5 35 5.646199e+09 137510.00
##                    660) GrLivArea>=1522.5 23 3.219239e+09 134378.30
##                     1320) Exterior1st=HdBoard,VinylSd,Wd Sdng 11 1.567602e+09 127127.
30 *
##                     1321) Exterior1st=MetalSd,Other 12 5.431425e+08 141025.00 *
##                    661) GrLivArea< 1522.5 12 1.769021e+09 143512.50 *
##                  331) HalfBath>=0.5 14 6.325024e+09 150921.40 *
##               83) YearRemodAdd>=1961 112 6.204682e+10 156641.60
##                166) GrLivArea< 1932.5 88 3.678549e+10 151508.70
##                  332) BsmtFinSF1< 467 54 1.588062e+10 143912.30
##                    664) LotFrontage< 84 47 9.861887e+09 140184.30
##                     1328) TotalBsmtSF< 569 8 1.896279e+09 124662.50 *
##                     1329) TotalBsmtSF>=569 39 5.642820e+09 143368.30
##                       2658) FireplaceQu=No Fireplace,Po 24 2.657544e+09 138748.50
##                         5316) BsmtFinType1=GLQ,Unf 12 1.064485e+09 133409.50 *
##                         5317) BsmtFinType1=ALQ,BLQ,LwQ,Rec 12 9.089406e+08 144087.50
*
##                       2659) FireplaceQu=Gd,TA 15 1.653496e+09 150760.00 *
##                    665) LotFrontage>=84 7 9.798371e+08 168942.90 *
##                  333) BsmtFinSF1>=467 34 1.283974e+10 163573.50
##                    666) Neighborhood=MeadowV,NPkVill,OldTown,Sawyer 10 3.527916e+09 1
46225.00 *
##                    667) Neighborhood=BrkSide,Edwards,Mitchel,NAmes 24 5.048062e+09 17
0802.10
##                     1334) TotalBsmtSF< 1157 11 1.136962e+09 162027.30 *
##                     1335) TotalBsmtSF>=1157 13 2.347463e+09 178226.90 *
##                167) GrLivArea>=1932.5 24 1.444138e+10 175462.50
##                  334) GarageYrBlt>=1957 17 6.755721e+09 165641.20 *
##                  335) GarageYrBlt< 1957 7 2.063509e+09 199314.30 *
##           21) LotArea>=12561.5 53 1.453373e+11 182621.20
##             42) KitchenQual=Fa,TA 41 8.432780e+10 167486.00
##               84) Fireplaces< 1.5 30 3.194658e+10 148714.20
##                168) Fireplaces< 0.5 9 7.021340e+09 119766.70 *
##                169) Fireplaces>=0.5 21 1.415150e+10 161120.20
##                  338) GrLivArea< 1657 9 6.210722e+09 143444.40 *
##                  339) GrLivArea>=1657 12 3.019944e+09 174377.10 *
##               85) Fireplaces>=1.5 11 1.297864e+10 218681.80 *
##             43) KitchenQual=Ex,Gd 12 1.952767e+10 234333.30 *
##         11) Neighborhood=ClearCr,CollgCr,Crawfor,Gilbert,NridgHt,NWAmes,SawyerW,Somer
st,Timber,Veenker 186 2.560787e+11 196444.60
##           22) GrLivArea< 2093.5 155 1.341352e+11 187091.70
##             44) BsmtFinSF1< 593.5 99 5.366151e+10 176118.30
##               88) KitchenAbvGr>=1.5 8 1.897482e+09 131600.90 *
##               89) KitchenAbvGr< 1.5 91 3.451583e+10 180031.90
##                178) TotalBsmtSF< 786.5 33 1.196113e+10 170088.00
##                  356) BsmtFinSF1< 217.5 20 4.757086e+09 159856.00
##                    712) Neighborhood=ClearCr,Crawfor,SawyerW 7 1.217269e+08 140045.60
*
##                    713) Neighborhood=Gilbert,NWAmes,Timber 13 4.089631e+08 170523.10
*
```

```
##                  357) BsmtFinSF1>=217.5 13 1.888763e+09 185829.60 *
##                179) TotalBsmtSF>=786.5 58 1.743503e+10 185689.70
##                  358) MasVnrArea< 293 49 1.036308e+10 182604.10
##                    716) YearRemodAdd>=1967.5 41 5.424490e+09 179878.00
##                     1432) OpenPorchSF< 38.5 17 1.811141e+09 170841.20 *
##                     1433) OpenPorchSF>=38.5 24 1.241660e+09 186279.20 *
##                    717) YearRemodAdd< 1967.5 8 3.072415e+09 196575.00 *
##                  359) MasVnrArea>=293 9 4.065509e+09 202488.90 *
##             45) BsmtFinSF1>=593.5 56 4.747746e+10 206491.20
##               90) Neighborhood=Gilbert,NWAmes,SawyerW,Veenker 29 8.277487e+09 190848.
80
##                180) X1stFlrSF< 1605.5 21 3.662546e+09 185243.60
##                  360) HeatingQC=Gd,TA 12 1.719695e+09 179676.30 *
##                  361) HeatingQC=Ex 9 1.075000e+09 192666.70 *
##                181) X1stFlrSF>=1605.5 8 2.223219e+09 205562.50 *
##               91) Neighborhood=ClearCr,CollgCr,Crawfor,Somerst,Timber 27 2.448267e+10
223292.30
##                182) Exterior1st=HdBoard,VinylSd,Wd Sdng 18 6.053178e+09 208951.60 *
##                183) Exterior1st=Other 9 7.324022e+09 251973.80 *
##           23) GrLivArea>=2093.5 31 4.059107e+10 243208.80
##             46) Neighborhood=ClearCr,Crawfor,Gilbert,Timber 19 1.508205e+10 226305.30
*
##             47) Neighborhood=CollgCr,NridgHt,NWAmes,SawyerW 12 1.148445e+10 269972.80
*
##      3) ExterQual=Ex,Gd 727 3.574869e+12 238050.70
##        6) GarageCars< 2.5 510 1.269908e+12 209839.70
##         12) X1stFlrSF< 1446 373 5.675778e+11 193570.80
##           24) GrLivArea< 1748.5 269 2.177148e+11 178686.40
##             48) X1stFlrSF< 1274 194 1.194392e+11 169870.00
##               96) Neighborhood=BrkSide,ClearCr,Edwards,Mitchel,NAmes,OldTown,Sawyer,S
WISU 38 1.386876e+10 140578.90
##                192) LotArea< 7136.5 16 2.259849e+09 128606.20 *
##                193) LotArea>=7136.5 22 7.647366e+09 149286.40
##                  386) GarageArea< 501.5 15 2.641149e+09 141193.30 *
##                  387) GarageArea>=501.5 7 1.918494e+09 166628.60 *
##               97) Neighborhood=Blmngtn,Blueste,CollgCr,Crawfor,Gilbert,Greens,NridgH
t,NWAmes,SawyerW,Somerst,StoneBr,Timber,Veenker 156 6.502606e+10 177005.00
##                194) GrLivArea< 1204 37 8.034938e+09 157738.70
##                  388) Neighborhood=Blmngtn,CollgCr,SawyerW,Somerst 30 3.403489e+09 15
2961.10
##                    776) X1stFlrSF< 1141.5 21 1.080043e+09 148453.00 *
##                    777) X1stFlrSF>=1141.5 9 9.008288e+08 163480.00 *
##                  389) Neighborhood=Blueste,Gilbert,Greens,NWAmes,StoneBr,Timber 7 1.0
11929e+09 178214.30 *
##                195) GrLivArea>=1204 119 3.898687e+10 182995.30
##                  390) TotalBsmtSF< 769 49 6.439587e+09 172881.50
##                    780) BedroomAbvGr< 2.5 12 1.457755e+09 161619.40 *
##                    781) BedroomAbvGr>=2.5 37 2.966202e+09 176534.00
##                     1562) X2ndFlrSF< 777 29 2.017196e+09 174442.90
##                       3124) WoodDeckSF< 28 9 7.143366e+08 168185.60 *
##                       3125) WoodDeckSF>=28 20 7.918961e+08 177258.70 *
##                     1563) X2ndFlrSF>=777 8 3.625001e+08 184114.40 *
##                   391) TotalBsmtSF>=769 70 2.402650e+10 190075.00  
##                     782) X2ndFlrSF< 840.5 54 1.512165e+10 186514.20  
##                      1564) Neighborhood=Blmngtn,Gilbert,Greens,NWAmes,SawyerW,Timber 2
5 5.224790e+09 179762.80  
##                        3128) GrLivArea< 1267.5 9 1.637578e+09 167308.90 *
##                        3129) GrLivArea>=1267.5 16 1.406120e+09 186768.10 *
##                      1565) Neighborhood=CollgCr,Crawfor,Somerst,StoneBr,Veenker 29 7.7
74948e+09 192334.40  
##                        3130) GarageArea>=478.5 20 2.746510e+09 187695.00  
##                          6260) GarageArea< 548 8 5.744088e+08 178287.50 *
##                          6261) GarageArea>=548 12 9.920867e+08 193966.70 *
##                        3131) GarageArea< 478.5 9 3.641326e+09 202644.20 *
##                     783) X2ndFlrSF>=840.5 16 5.909338e+09 202092.80 *
##              49) X1stFlrSF>=1274 75 4.419064e+10 201491.50  
##                98) Neighborhood=Blmngtn,CollgCr,Gilbert,NAmes,NWAmes,SawyerW 33 9.6452
13e+09 186641.80  
##                 196) Neighborhood=Gilbert,NAmes,NWAmes,SawyerW 18 1.627613e+09 178988.
80 *
##                 197) Neighborhood=Blmngtn,CollgCr 15 5.698282e+09 195825.50 *
##                99) Neighborhood=ClearCr,Greens,Mitchel,NridgHt,Somerst,StoneBr,Timber,
Veenker 42 2.155091e+10 213159.10  
##                 198) GarageFinish=RFn,Unf 32 6.788699e+09 204612.60  
##                   396) LotArea< 8419 23 3.573489e+09 199952.80  
##                     792) LotFrontage>=59.5 7 7.962521e+08 189742.90 *
##                     793) LotFrontage< 59.5 16 1.728287e+09 204419.70 *
##                   397) LotArea>=8419 9 1.439511e+09 216521.00 *
##                 199) GarageFinish=Fin 10 4.945231e+09 240508.00 *
##            25) GrLivArea>=1748.5 104 1.361211e+11 232069.80  
##              50) TotalBsmtSF< 1052.5 69 6.478386e+10 218543.70  
##               100) Neighborhood=CollgCr,Edwards,Gilbert,NoRidge,NWAmes,OldTown,Sawyer,
SawyerW,SWISU,Timber 59 3.856651e+10 211397.60  
##                 200) BsmtQual=Fa,No Basement,TA 7 5.305994e+09 165628.60 *
##                 201) BsmtQual=Gd 52 1.662296e+10 217558.80  
##                   402) BsmtFinType1=ALQ,Unf 20 4.520155e+09 205864.50  
##                     804) X2ndFlrSF< 1037.5 9 1.398326e+09 197627.80 *
##                     805) X2ndFlrSF>=1037.5 11 2.011661e+09 212603.60 *
##                   403) BsmtFinType1=GLQ 32 7.658241e+09 224867.70  
##                     806) TotalBsmtSF< 917.5 15 1.162517e+09 216653.30 *
##                     807) TotalBsmtSF>=917.5 17 4.590530e+09 232115.60 *
##               101) Neighborhood=Crawfor,Somerst 10 5.427956e+09 260705.80 *
##              51) TotalBsmtSF>=1052.5 35 3.382601e+10 258735.60  
##               102) GrLivArea< 2184 20 7.177199e+09 240141.00  
##                 204) WoodDeckSF< 130 8 7.239483e+08 224891.20 *
##                 205) WoodDeckSF>=130 12 3.352519e+09 250307.50 *
##               103) GrLivArea>=2184 15 1.051333e+10 283528.50 *
##          13) X1stFlrSF>=1446 137 3.348141e+11 254134.00  
##            26) Neighborhood=Blmngtn,CollgCr,Edwards,Mitchel,NAmes,NWAmes,OldTown,Sawye
rW,Somerst,StoneBr,Timber 99 1.529120e+11 239198.80  
##              52) GrLivArea< 1592.5 47 3.941796e+10 221856.80  
##               104) BsmtUnfSF>=1096.5 23 1.318671e+10 205102.20  
##                 208) Neighborhood=Blmngtn,CollgCr,Mitchel,NAmes,SawyerW,Timber 15 2.42
6446e+09 193305.90 *
```

```
##                 209) Neighborhood=Somerst 8 4.759368e+09 227220.10 *
##              105) BsmtUnfSF< 1096.5 24 1.358725e+10 237913.30
##                 210) YearBuilt< 2004.5 17 3.519644e+09 227705.30 *
##                 211) YearBuilt>=2004.5 7 3.993999e+09 262704.30 *
##             53) GrLivArea>=1592.5 52 8.658317e+10 254873.30
##              106) Neighborhood=CollgCr,Edwards,Mitchel,NAmes,NWAmes,OldTown,Timber 31
3.667409e+10 239100.40
##                 212) BsmtExposure=Mn,No Exposure 19 1.346494e+10 226073.20 *
##                 213) BsmtExposure=Av,Gd 12 1.487934e+10 259726.80 *
##              107) Neighborhood=SawyerW,Somerst,StoneBr 21 3.081187e+10 278157.10
##                 214) YearRemodAdd< 1998 9 8.326120e+09 251600.00 *
##                 215) YearRemodAdd>=1998 12 1.137756e+10 298075.00 *
##           27) Neighborhood=ClearCr,Crawfor,NoRidge,NridgHt,Veenker 38 1.022877e+11 29
3044.00
##             54) TotalBsmtSF< 1567 13 1.464072e+10 259128.50 *
##             55) TotalBsmtSF>=1567 25 6.491780e+10 310680.00
##              110) GrLivArea< 1856.5 16 4.158005e+10 296965.00 *
##              111) GrLivArea>=1856.5 9 1.497764e+10 335062.30 *
##        7) GarageCars>=2.5 217 9.451414e+11 304353.00
##         14) KitchenQual=Gd,TA 142 3.937172e+11 277137.00
##           28) Neighborhood=Blmngtn,CollgCr,Gilbert,Mitchel,NAmes,NWAmes,OldTown,Sawye
rW,Somerst,Timber 83 1.250205e+11 249986.90
##             56) TotalBsmtSF< 1797.5 76 8.073447e+10 243560.60
##              112) GrLivArea< 2197.5 64 5.961513e+10 237688.50
##                 224) BsmtFinSF1< 1145 56 4.376146e+10 232125.50
##                   448) OpenPorchSF< 22 12 5.776200e+09 208114.90 *
##                   449) OpenPorchSF>=22 44 2.918042e+10 238673.80
##                     898) LotArea< 10062.5 12 1.105174e+10 222167.90 *
##                     899) LotArea>=10062.5 32 1.363334e+10 244863.50
##                      1798) TotalBsmtSF< 1618.5 21 8.119032e+09 237315.10
##                        3596) LotArea>=11697.5 9 1.123250e+09 224868.60 *
##                        3597) LotArea< 11697.5 12 4.555850e+09 246650.00 *
##                      1799) TotalBsmtSF>=1618.5 11 2.033418e+09 259274.20 *
##                 225) BsmtFinSF1>=1145 8 1.989257e+09 276629.80 *
##              113) GrLivArea>=2197.5 12 7.142676e+09 274878.70 *
##             57) TotalBsmtSF>=1797.5 7 7.072257e+09 319757.10 *
##           29) Neighborhood=NoRidge,NridgHt,StoneBr,Veenker 59 1.214454e+11 315331.30
##             58) TotalBsmtSF< 1743 43 5.455331e+10 301056.70
##              116) GrLivArea< 2390.5 23 1.552357e+10 282136.30
##                 232) OpenPorchSF< 63 15 5.911989e+09 270826.70 *
##                 233) OpenPorchSF>=63 8 4.095510e+09 303342.00 *
##              117) GrLivArea>=2390.5 20 2.132769e+10 322815.00
##                 234) YearBuilt< 1995.5 9 5.237556e+09 301777.80 *
##                 235) YearBuilt>=1995.5 11 8.848133e+09 340027.40 *
##             59) TotalBsmtSF>=1743 16 3.458255e+10 353694.40 *
##         15) KitchenQual=Ex 75 2.471010e+11 355881.90
##           30) BsmtUnfSF>=599 34 1.088776e+11 326350.40
##             60) Neighborhood=CollgCr,Edwards,NoRidge,Somerst 8 3.067148e+10 282116.60
*
##             61) Neighborhood=NridgHt,StoneBr,Timber 26 5.773678e+10 339960.80
##              122) LotArea< 12217.5 12 1.055214e+10 310238.30 *
##              123) LotArea>=12217.5 14 2.749692e+10 365437.10 *
##           31) BsmtUnfSF< 599 41 8.398221e+10 380371.50
##             62) BsmtFinSF1< 1270 18 3.566146e+10 347678.90 *
##             63) BsmtFinSF1>=1270 23 1.402603e+10 405957.00
##              126) GrLivArea< 1958 9 1.945442e+09 384827.70 *
##              127) GrLivArea>=1958 14 5.479498e+09 419540.20 *
```

```
# in-sample R-squared
in_sample_pred <- predict(final_model, newdata = train_ames)
SST_in <- sum((train_ames$SalePrice - mean(train_ames$SalePrice))^2)
SSR_in <- sum((in_sample_pred - mean(train_ames$SalePrice))^2)
R2_in_sample <- SSR_in / SST_in
cat("In-sample R-squared:", R2_in_sample, "\n")
```

```
## In-sample R-squared: 0.9285921
```

```
# out-of-sample R-squared from cross-validation results
out_of_sample_R2 <- max(tree_model_cv$results$Rsquared)
cat("Out-of-sample R-squared:", out_of_sample_R2, "\n")
```

```
## Out-of-sample R-squared: 0.7904077
```

RMSE was used to select the optimal model using the smallest value. The final value used for the model was cp = 5e-05 with n= 1972. This model has following stats: MAE of 21687.35, RMSE of 31979.45, in-sample R-squared: 0.9285921, and out-of-sample R-squared of 0.7904077.

Part e

```
# The mtry parameter in a Random Forest model refers to the number of variables (or predictors) that are randomly sampled as candidates at each split when growing a decision tree within the forest

# cv control
control <- trainControl(method = "cv", number = 5)

# mtry tuning grid
mtry_values <- data.frame(mtry = 1:73)

# train rf w 80 trees
rf_model_tuned <- train(
  SalePrice ~ .,
  data = train_ames,
  method = "rf",
  tuneGrid = mtry_values,
  trControl = control,
  ntree = 80,
  importance = TRUE
)

# display selected mtry
selected_mtry <- rf_model_tuned$bestTune$mtry
cat("Selected mtry value:", selected_mtry, "\n")
```

```
## Selected mtry value: 37
```

```
# insample R²
pred_train <- predict(rf_model_tuned, newdata = train_ames)
SSE_train <- sum((train_ames$SalePrice - pred_train)^2)
SST_train <- sum((train_ames$SalePrice - mean(train_ames$SalePrice))^2)
R2_train <- 1 - SSE_train / SST_train

# oos R²
pred_test <- predict(rf_model_tuned, newdata = test_ames)
SSE_test <- sum((test_ames$SalePrice - pred_test)^2)
SST_test <- sum((test_ames$SalePrice - mean(train_ames$SalePrice))^2)
R2_test <- 1 - SSE_test / SST_test

cat("In-sample R²:", round(R2_train, 4), "\n")
```

```
## In-sample R²: 0.9797
```

```
cat("Out-of-sample R²:", round(R2_test, 4), "\n")
```

```
## Out-of-sample R²: 0.8733
```

Selected mtry value: 36 In-sample R^2: 0.9806 Out-of-sample R^2: 0.872

Part f

Out of the four models constructed, I would recommend my model from e - a random forest model with 80 trees, a nodesize of 25, and selected mtry value of 36. I believe this is the most robust model compared to the other linear regression and CART models. Out of all 4 models, this model had the highest out of sample R^2 at 0.872, meaning that 87.2% of the variation in SalePrice from the test dataset could be explained by the model built on 36 variables chosen from the training dataset. The other models we looked at all had lower OOS R^2 at 0.83, 0.69, and 0.79 respectively for a, b, and d. Despite most of those models showing promising in sample R^2 (all above 0.75 and two around 0.90), they didn't perform as well with the test data, meaning that there could be some sort of overfitting occurring that isn't allowing for generalization towards unseen data.

Also, I prefer my random forest model because it is very flexible compared to the other models. RF algorithms can effectively capture relationships and complex patterns within data by creating numerous decision trees, which can model non-linear boundaries and interactions between features. Specifically, with our mtry feature, at each node of a tree, the algorithm does not consider all available predictors. Instead, it randomly selects mtry number of predictors from the full set of predictors. This makes it less likely to select 2 highly correlated variables because one will be a more important predictor than the other (reducing the rmse more.) Unlike linear models that assume a straightforward relationship, the ensemble nature of a random forest allows it to learn intricate patterns that might be difficult for other algorithms to detect, making it a powerful and flexible tool for many machine learning tasks. The one thing that's not great about the RF is that it isn't interpretable. While individual trees are interpretable, the fact that we have a forest of trees with aggregate decision-making is difficult to follow. However, we can make plots and use techniques like feature importance to gain insights into the model's behavior and identify key drivers of its predictions.