Summary for the best subset selection.

Due the high number of variables in the dataset and some issues with the variables (missing values, different formats of data and so on) we were not able to run the subset selection on the full dataset because different functions for subset selection in R gave us an error. It also can be caused by high correlation between some variables. Thus, we had to construct our regression model manually taking the logic of the problem into account. So, by repeatedly changing/adding/subtracting regressors we found the model that consists of 27 predictors and give the accuracy about 84% (adjusted R-squared). Also, we used na.omit() function to clean the data.

Code

```
> model1=lm(SalePrice~. ,data=na.omit(data_best_subset_sel))
> summary(model1)
```

Code Output:

```
Residual standard error: 31280 on 1345 degrees of freedom
Multiple R-squared:  0.8557, Adjusted R-squared:  0.8443
F-statistic: 75.26 on 106 and 1345 DF,  p-value: < 2.2e-16
```

For the subsect selection we decided randomly add 6 variables and to run best subset selection to check if we can improve the accuracy.

Code:

```
data_subset_sel_Clean=na.omit(data_subset_sel)
full.tr =  lm (SalePrice~., data=data_subset_sel_Clean) #full model
Intercept.tr=lm (SalePrice~1, data=data_subset_sel_Clean) #the smallest model, intercept only)
step.tr_both=stepAIC(Intercept.tr,scope=list(upper=full.tr,lower=Intercept.tr),direction="both")
step.tr_backward=stepAIC(full.tr,direction="backward")
step.tr_forward=stepAIC(Intercept.tr,scope=list(upper=full.tr,lower=Intercept.tr),direction="forward")
summary(step.tr_both)
summary(step.tr_forward)
summary(step.tr_forward)
```

“Both” and “forward” directions produced the most accurate model among the 3 and that is more precise that our model constructed manually.

```
summary(step.tr_both)

Call:
lm(formula = SalePrice ~ OverallQual + GrLivArea + Neighborhood + 
    RoofMatl + HouseStyle + ExterQual + BldgType + BsmtFinType1 + 
    YearBuilt + OverallCond + LotArea + Fireplaces + PoolArea + 
    MasVnrType + MasVnrArea + Condition1 + LotConfig + Foundation + 
    LandSlope + BsmtFinType2 + LowQualFinSF + YearRemodAdd + 
    Exterior1st, data = data_subset_sel_Clean)
```

```
Residuals:
Min     1Q  Median     3Q    Max
-331718  -12123      0  10925  203533

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)        -1.610e+06  1.711e+05  -9.410  < 2e-16 ***
OverallQual         9.947e+03  1.124e+03   8.847  < 2e-16 ***
GrLivArea           6.907e+01  2.958e+00  23.350  < 2e-16 ***
NeighborhoodClearCr -2.078e+04  1.026e+04  -2.027 0.042887 *
NeighborhoodCollgCr -1.971e+04  8.043e+03  -2.451 0.014391 *
NeighborhoodEdwards -2.756e+04  8.825e+03  -3.124 0.001826 **
NeighborhoodGilbert -2.560e+04  8.623e+03  -2.969 0.003039 **
NeighborhoodIDOTRR  -2.440e+04  1.021e+04  -2.390 0.017001 *
NeighborhoodMeadowV -8.154e+03  1.196e+04  -0.682 0.495359
NeighborhoodMitchel -2.605e+04  9.039e+03  -2.881 0.004023 **
NeighborhoodNAmes   -2.472e+04  8.563e+03  -2.887 0.003954 **
NeighborhoodNoRidge  1.709e+04  9.391e+03   1.820 0.068974 .
NeighborhoodNridgHt  2.448e+04  8.446e+03   2.898 0.003814 **
NeighborhoodNWAmes  -3.000e+04  8.869e+03  -3.382 0.000739 ***
NeighborhoodOldTown -2.506e+04  9.319e+03  -2.689 0.007266 **
NeighborhoodSawyer  -2.040e+04  8.992e+03  -2.268 0.023469 *
NeighborhoodSawyerW -1.824e+04  8.760e+03  -2.082 0.037522 *
NeighborhoodStoneBr  4.367e+04  9.400e+03   4.645 3.73e-06 ***
NeighborhoodSWISU   -2.703e+04  1.065e+04  -2.538 0.011270 *
RoofMatlCompShg      5.193e+05  3.204e+04  16.207  < 2e-16 ***
RoofMatlMembran      5.627e+05  4.549e+04  12.368  < 2e-16 ***
RoofMatlMetal        5.712e+05  4.529e+04  12.612  < 2e-16 ***
RoofMatlRoll         5.035e+05  4.322e+04  11.650  < 2e-16 ***
RoofMatlTar&Grv      5.060e+05  3.308e+04  15.296  < 2e-16 ***
RoofMatlWdShake      5.260e+05  3.510e+04  14.983  < 2e-16 ***
RoofMatlWdShngl      6.123e+05  3.380e+04  18.115  < 2e-16 ***
HouseStyle1.5Unf     1.407e+04  8.155e+03   1.726 0.084649 .
HouseStyle1Story     1.545e+04  3.283e+03   4.706 2.79e-06 ***
HouseStyleSFoyer     1.874e+04  6.219e+03   3.013 0.002632 **
ExterQualFa         -3.847e+04  1.111e+04  -3.464 0.000550 ***
ExterQualGd         -4.389e+04  5.048e+03  -8.694  < 2e-16 ***
ExterQualTA         -4.464e+04  5.662e+03  -7.884 6.62e-15 ***
BldgType2fmCon      -9.390e+03  5.611e+03  -1.674 0.094428 .
BldgTypeDuplex      -1.743e+04  5.333e+03  -3.269 0.001107 **
BldgTypeTwnhs       -4.409e+04  5.841e+03  -7.549 8.19e-14 ***
BldgTypeTwnhsE      -3.168e+04  3.781e+03  -8.379  < 2e-16 ***
BsmtFinType1GLQ      8.492e+03  2.891e+03   2.938 0.003364 **
BsmtFinType1LwQ     -8.721e+03  4.168e+03  -2.093 0.036574 *
BsmtFinType1Unf     -1.029e+04  2.762e+03  -3.726 0.000203 ***
YearBuilt            4.486e+02  7.270e+01   6.172 8.99e-10 ***
```

```
OverallCond          5.754e+03  9.032e+02   6.371 2.60e-10 ***
LotArea              6.668e-01  1.076e-01   6.195 7.78e-10 ***
Fireplaces           5.921e+03  1.516e+03   3.905 9.90e-05 ***
PoolArea             1.018e+02  2.041e+01   4.986 6.99e-07 ***
MasVnrTypeBrkFace    1.418e+04  7.723e+03   1.836 0.066612 .
MasVnrTypeNone       1.937e+04  7.766e+03   2.494 0.012757 *
MasVnrTypeStone      2.664e+04  8.174e+03   3.259 0.001147 **
MasVnrArea           2.875e+01  6.637e+00   4.332 1.59e-05 ***
Condition1Norm       9.756e+03  4.601e+03   2.120 0.034161 *
Condition1PosN      -1.559e+04  8.116e+03  -1.920 0.055016 .
Condition1RRAe      -1.831e+04  1.036e+04  -1.768 0.077330 .
LotConfigCulDSac     1.023e+04  3.665e+03   2.792 0.005322 **
FoundationPConc      6.979e+03  3.920e+03   1.781 0.075193 .
LandSlopeMod         7.259e+03  3.934e+03   1.845 0.065202 .
LandSlopeSev        -2.421e+04  1.127e+04  -2.149 0.031852 *
BsmtFinType2BLQ     -2.312e+04  8.334e+03  -2.775 0.005604 **
BsmtFinType2LwQ     -2.595e+04  8.104e+03  -3.202 0.001396 **
BsmtFinType2Rec     -2.436e+04  7.823e+03  -3.114 0.001885 **
BsmtFinType2Unf     -1.949e+04  6.829e+03  -2.854 0.004384 **
LowQualFinSF        -3.937e+01  1.981e+01  -1.988 0.047066 *
YearRemodAdd         1.134e+02  5.932e+01   1.913 0.056026 .
Exterior1stBrkComm  -5.299e+04  2.891e+04  -1.833 0.067027 .
Exterior1stBrkFace   1.406e+04  8.443e+03   1.665 0.096131 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 27730 on 1314 degrees of freedom
Multiple R-squared:  0.886,	Adjusted R-squared:  0.8775 
F-statistic: 103.2 on 99 and 1314 DF,  p-value: < 2.2e-16
```

So, our final model is:

```
lm(formula = SalePrice ~ OverallQual + GrLivArea + Neighborhood + 
    RoofMatl + HouseStyle + ExterQual + BldgType + BsmtFinType1 + 
    YearBuilt + OverallCond + LotArea + Fireplaces + PoolArea + 
    MasVnrType + MasVnrArea + Condition1 + LotConfig + Foundation + 
    LandSlope + BsmtFinType2 + LowQualFinSF + YearRemodAdd + 
    Exterior1st, data = data_subset_sel_Clean)
```

Next we are looking for MSE on the trainset

```
> Y_pr_tr=predict(model_final_tr, data=data_subset_sel_Clean)
> View(Y_pr_tr)
> View(data_subset_sel_Clean)
> err.tr=mean((data_subset_sel_Clean$SalePrice-Y_pr_tr)^2)
> err.tr
[1] 714564136
```

Predicting_test

```
vars_sub_select_exper_test=c("OverallQual","LotArea","OverallCond","YearBuilt","BldgType","YearRemodAdd","RoofStyle","PoolArea","MiscVal","Fireplaces","Neighborhood","EnclosedPorch","TotRmsAbvGrd","GrLivArea","LowQualFinSF","ExterQual", "MasVnrArea", "MasVnrType","Exterior1st","Exterior2nd","LotShape","LotConfig", "LandSlope","HouseStyle","Condition1", "BsmtCond","BsmtFinType2","ExterCond","HeatingQC","RoofMatl","Foundation","LandSlope","BsmtFinType1")
> data_test=test_set[vars_sub_select_exper_test]
> test_pr=predict(model_final_tr,vars_sub_select_exper_test)
Error in eval(predvars, data, env) : 
  invalid 'envir' argument of type 'character'
> test_pr=predict(model_final_tr,data_test)
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) : 
  factor Foundation has new levels Slab
> test_pr=predict(model_final_tr,data=data_test)
> view(test_pr)
Error in view(test_pr) : could not find function "view"
> View(test_pr)
> test_pr=predict(model_final_tr,newdata=data_test)
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) : 
  factor Foundation has new levels Slab
> test_pr=predict(model_final_tr,newdata=na.omit(data_test))
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) : 
  factor Exterior1st has new levels AsphShn
> data_test=na.omit(data_test)
> test_pr=predict(model_final_tr,newdata=data_test)
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) : 
  factor Exterior1st has new levels AsphShn
> data_test=data_test[!(data_test$Exterior1st=='AsphShn'),]
> test_pr=predict(model_final_tr,newdata=data_test)
> View(test_pr)
```

Unfortunately, this dataset came from Kaggle competition, so we don’t have the actual values for the target variable in the test set. However, if to take a look at the predicted values it looks like adequate predictions.

We also try to run log-linear regression (log-log requires too much time to process the data).

The code

```
> Copy_data_subset_sel_Clean$SalePrice=log(Copy_data_subset_sel_Clean$SalePrice)
> log_model_tr=lm(SalePrice~.,data=Copy_data_subset_sel_Clean)
> summary(log_model_tr)
```

```
Residual standard error: 0.1285 on 1278 degrees of freedom
Multiple R-squared:  0.9036, Adjusted R-squared:  0.8935
F-statistic: 88.77 on 135 and 1278 DF,  p-value: < 2.2e-16
```

As we see the log-linear regression has higher adjusted R-squared.