605_Final_HomeSales

Eric Hirsch, Cameron Smith and Carlisle Fergusen

4/7/2022

Contents


```
# Load libraries
devtools::install_github("ericonsi/EHData", force=TRUE)
```

```
##       v  checking for file 'C:\Users\Eric\AppData\Local\Temp\RtmpeWkj9B\remotes3d7464f07968\ericonsi
##       -  preparing 'EHData':
##    checking DESCRIPTION meta-information ...     checking DESCRIPTION meta-information ...   v  check
##       -  checking for LF line-endings in source and make files and shell scripts
## -  checking for empty or unneeded directories
##    -  creating default NAMESPACE file
##    -  building 'EHData_0.1.0.tar.gz'
##
##
```

```
library(EHData)
library(data.table)
library(tidymodels)
library(vip)
library(tidyverse)
library(lmtest)
library(skimr)
library(mltools)
library(psych)
library(MASS)
library(broom)
```

Introduction

In this paper we analyze housing prices by comparing three prediction methodologies: OLS, Ridge regression, and Random Forest. The purpose is to compare the methodologies and draw conclusions about which are most effective and why. Regression alone is not necessarily the optimal strategy for predicting housing prices.[1] However, when data sets and/or analysis resources are limited, regression can perform adequately.

Background and Literature Review

The ability to accurately predict home prices is of tremendous value to a number of industries, including investors, real estate agents, and municipalities who depend upon property tax revenue. [1] Predictive models for home prices fall roughly into two kinds. First, there are those which predict market trends, busts, and booms. These predictions rely mainly on timeseries data and analysis of housing prices in the aggregate. The other type of prediction involves the capacity to predict individual house prices from a set of factors. These usually employ some form of regression and/or machine learning.[2]

For either sort of prediction, there is no consensus about the best method. Many researchers have sought to enhance the traditional models with other methodologies.[3] For example, Guan et. al. propose a "data stream" approach in which past sale records are treated as an evolving datastream.[4] Li et. al. introduce a "grey seasonal model" in which seasonal fluctuations are modeled using grey systems theory, which incorporates uncertainty.[5] Alfiyatin, et. el. use particle swarm optimization (PSO) to select independent variables.[6] (PSO is an optimization system in which population is initialized with random solutions and searches for optima by updating generations.) Finally, Liu et.al incorporate both spatial and temporal autocorrelation in their models by analyzing experience-based submarkets by real estate professionals.[7]

All of these researchers report that their innovations improve their regression models. Indeed, any real estate agent can tell you that a predictive model can be improved simply by knowing what other houses in the neighborhood sold for. The problem is, the data at the center of these enhancements is not always available. The researcher may have home sales from only a short time span, and neighborhoods that are not defined by real estate experts but by traditional boundary lines which may contain a mix of house types. Even when data is available, the complex models proposed may be computationally expensive and/or require data analysis expertise that is not generally available.

In this project we approach the question comparatively. Restricting ourselves to regression models, we compare three types of regression: OLS, Ridge, and Random Forest. At the data is drawn from the Advanced

[1] 1
[2] 2
[3] 3
[4] 4
[5] 5
[6] 6
[7] 7

Regression Techniques housing data set for Ames, Iowa. We test the accuracy of our models by submitting each to the Kaggle competition to see how they perform. We then discussed the merits of the different sorts of approaches.

Modeling

We are modeling a data set containing 1460 records of houses sold in the Ames, Iowa area between 2006 and 2010. The variables are mostly related to house features, such as square footage, the presense of a pool, etc. The response variable, "SalePrice", is a continuous variable representing the sale price of the house in dollars.

We examine the data:

```
dfTrain <- read.csv("https://raw.githubusercontent.com/ericonsi/CUNY_621/main/Final/train.csv", stringsA

dfTest <- read.csv("https://raw.githubusercontent.com/ericonsi/CUNY_621/main/Final/test.csv", stringsAsF
```

1. Dataset Description

```
summary(dfTrain)
```

A. Summary Statistics

```
##        Id           MSSubClass       MSZoning     LotFrontage
##  Min.   :   1.0   Min.   : 20.0   C (all):  10   Min.   : 21.00
##  1st Qu.: 365.8   1st Qu.: 20.0   FV     :  65   1st Qu.: 59.00
##  Median : 730.5   Median : 50.0   RH     :  16   Median : 69.00
##  Mean   : 730.5   Mean   : 56.9   RL     :1151   Mean   : 70.05
##  3rd Qu.:1095.2   3rd Qu.: 70.0   RM     : 218   3rd Qu.: 80.00
##  Max.   :1460.0   Max.   :190.0                   Max.   :313.00
##                                                  NA's   :259
##     LotArea        Street      Alley      LotShape  LandContour  Utilities
##  Min.   :  1300   Grvl:   6   Grvl:  50   IR1:484   Bnk:  63    AllPub:1459
##  1st Qu.:  7554   Pave:1454   Pave:  41   IR2: 41   HLS:  50    NoSeWa:   1
##  Median :  9478               NA's:1369   IR3: 10   Low:  36
##  Mean   : 10517                           Reg:925   Lvl:1311
##  3rd Qu.: 11602
##  Max.   :215245
##
##    LotConfig    LandSlope   Neighborhood   Condition1     Condition2
##  Corner : 263   Gtl:1382   NAmes  :225   Norm   :1260   Norm   :1445
##  CulDSac:  94   Mod:  65   CollgCr:150   Feedr  :  81   Feedr  :   6
##  FR2    :  47   Sev:  13   OldTown:113   Artery :  48   Artery :   2
##  FR3    :   4              Edwards:100   RRAn   :  26   PosN   :   2
##  Inside :1052              Somerst: 86   PosN   :  19   RRNn   :   2
##                            Gilbert: 79   RRAe   :  11   PosA   :   1
##                            (Other):707   (Other):  15   (Other):   2
##    BldgType      HouseStyle   OverallQual      OverallCond      YearBuilt
##  1Fam  :1220   1Story :726   Min.   : 1.000   Min.   :1.000   Min.   :1872
##  2fmCon:  31   2Story :445   1st Qu.: 5.000   1st Qu.:5.000   1st Qu.:1954
```

```
##  Duplex:  52   1.5Fin :154   Median : 6.000   Median :5.000   Median :1973
##  Twnhs :  43   SLvl   : 65   Mean   : 6.099   Mean   :5.575   Mean   :1971
##  TwnhsE: 114   SFoyer : 37   3rd Qu.: 7.000   3rd Qu.:6.000   3rd Qu.:2000
##                1.5Unf : 14   Max.   :10.000   Max.   :9.000   Max.   :2010
##                (Other): 19
##   YearRemodAdd    RoofStyle       RoofMatl     Exterior1st   Exterior2nd
##  Min.   :1950   Flat   :  13   CompShg:1434   VinylSd:515   VinylSd:504
##  1st Qu.:1967   Gable  :1141   Tar&Grv:  11   HdBoard:222   MetalSd:214
##  Median :1994   Gambrel:  11   WdShngl:   6   MetalSd:220   HdBoard:207
##  Mean   :1985   Hip    : 286   WdShake:   5   Wd Sdng:206   Wd Sdng:197
##  3rd Qu.:2004   Mansard:   7   ClyTile:   1   Plywood:108   Plywood:142
##  Max.   :2010   Shed   :   2   Membran:   1   CemntBd: 61   CmentBd: 60
##                                (Other):   2   (Other):128   (Other):136
##    MasVnrType    MasVnrArea     ExterQual ExterCond  Foundation  BsmtQual
##  BrkCmn : 15   Min.   :   0.0   Ex: 52    Ex:   3   BrkTil:146   Ex  :121
##  BrkFace:445   1st Qu.:   0.0   Fa: 14    Fa:  28   CBlock:634   Fa  : 35
##  None   :864   Median :   0.0   Gd:488    Gd: 146   PConc :647   Gd  :618
##  Stone  :128   Mean   : 103.7   TA:906    Po:   1   Slab  : 24   TA  :649
##  NA's   :  8   3rd Qu.: 166.0             TA:1282   Stone :  6   NA's: 37
##                Max.   :1600.0                       Wood  :  3
##                NA's   :8
##  BsmtCond    BsmtExposure BsmtFinType1   BsmtFinSF1     BsmtFinType2
##  Fa  :  45   Av  :221     ALQ :220     Min.   :   0.0   ALQ :  19
##  Gd  :  65   Gd  :134     BLQ :148     1st Qu.:   0.0   BLQ :  33
##  Po  :   2   Mn  :114     GLQ :418     Median : 383.5   GLQ :  14
##  TA  :1311   No  :953     LwQ : 74     Mean   : 443.6   LwQ :  46
##  NA's:  37   NA's: 38     Rec :133     3rd Qu.: 712.2   Rec :  54
##                           Unf :430     Max.   :5644.0   Unf :1256
##                           NA's: 37                      NA's:  38
##    BsmtFinSF2        BsmtUnfSF       TotalBsmtSF      Heating     HeatingQC
##  Min.   :   0.00   Min.   :   0.0   Min.   :   0.0   Floor:   1   Ex:741
##  1st Qu.:   0.00   1st Qu.: 223.0   1st Qu.: 795.8   GasA :1428   Fa: 49
##  Median :   0.00   Median : 477.5   Median : 991.5   GasW :  18   Gd:241
##  Mean   :  46.55   Mean   : 567.2   Mean   :1057.4   Grav :   7   Po:  1
##  3rd Qu.:   0.00   3rd Qu.: 808.0   3rd Qu.:1298.2   OthW :   2   TA:428
##  Max.   :1474.00   Max.   :2336.0   Max.   :6110.0   Wall :   4
##
##  CentralAir Electrical     X1stFlrSF      X2ndFlrSF     LowQualFinSF
##  N:  95     FuseA:  94   Min.   : 334   Min.   :   0   Min.   :  0.000
##  Y:1365     FuseF:  27   1st Qu.: 882   1st Qu.:   0   1st Qu.:  0.000
##             FuseP:   3   Median :1087   Median :   0   Median :  0.000
##             Mix  :   1   Mean   :1163   Mean   : 347   Mean   :  5.845
##             SBrkr:1334   3rd Qu.:1391   3rd Qu.: 728   3rd Qu.:  0.000
##             NA's :   1   Max.   :4692   Max.   :2065   Max.   :572.000
##
##    GrLivArea     BsmtFullBath     BsmtHalfBath        FullBath
##  Min.   : 334   Min.   :0.0000   Min.   :0.00000   Min.   :0.000
##  1st Qu.:1130   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1.000
##  Median :1464   Median :0.0000   Median :0.00000   Median :2.000
##  Mean   :1515   Mean   :0.4253   Mean   :0.05753   Mean   :1.565
##  3rd Qu.:1777   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:2.000
##  Max.   :5642   Max.   :3.0000   Max.   :2.00000   Max.   :3.000
##
##     HalfBath       BedroomAbvGr    KitchenAbvGr   KitchenQual  TotRmsAbvGrd
```

```
##  Min.   :0.0000   Min.   :0.000   Min.   :0.000   Ex:100      Min.   : 2.000
##  1st Qu.:0.0000   1st Qu.:2.000   1st Qu.:1.000   Fa: 39      1st Qu.: 5.000
##  Median :0.0000   Median :3.000   Median :1.000   Gd:586      Median : 6.000
##  Mean   :0.3829   Mean   :2.866   Mean   :1.047   TA:735      Mean   : 6.518
##  3rd Qu.:1.0000   3rd Qu.:3.000   3rd Qu.:1.000               3rd Qu.: 7.000
##  Max.   :2.0000   Max.   :8.000   Max.   :3.000               Max.   :14.000
##
##  Functional    Fireplaces    FireplaceQu   GarageType   GarageYrBlt
##  Maj1:  14   Min.   :0.000   Ex  : 24    2Types :  6   Min.   :1900
##  Maj2:   5   1st Qu.:0.000   Fa  : 33    Attchd :870   1st Qu.:1961
##  Min1:  31   Median :1.000   Gd  :380    Basment: 19   Median :1980
##  Min2:  34   Mean   :0.613   Po  : 20    BuiltIn: 88   Mean   :1979
##  Mod :  15   3rd Qu.:1.000   TA  :313    CarPort:  9   3rd Qu.:2002
##  Sev :   1   Max.   :3.000   NA's:690    Detchd :387   Max.   :2010
##  Typ :1360                               NA's   : 81   NA's   :81
##  GarageFinish   GarageCars      GarageArea     GarageQual  GarageCond
##  Fin :352     Min.   :0.000   Min.   :   0.0   Ex  :   3   Ex  :   2
##  RFn :422     1st Qu.:1.000   1st Qu.: 334.5   Fa  :  48   Fa  :  35
##  Unf :605     Median :2.000   Median : 480.0   Gd  :  14   Gd  :   9
##  NA's: 81     Mean   :1.767   Mean   : 473.0   Po  :   3   Po  :   7
##               3rd Qu.:2.000   3rd Qu.: 576.0   TA  :1311   TA  :1326
##               Max.   :4.000   Max.   :1418.0   NA's:  81   NA's:  81
##
##  PavedDrive   WoodDeckSF      OpenPorchSF     EnclosedPorch      X3SsnPorch
##  N:  90     Min.   :  0.00   Min.   :  0.00   Min.   :  0.00   Min.   :  0.00
##  P:  30     1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00
##  Y:1340     Median :  0.00   Median : 25.00   Median :  0.00   Median :  0.00
##             Mean   : 94.24   Mean   : 46.66   Mean   : 21.95   Mean   :  3.41
##             3rd Qu.:168.00   3rd Qu.: 68.00   3rd Qu.:  0.00   3rd Qu.:  0.00
##             Max.   :857.00   Max.   :547.00   Max.   :552.00   Max.   :508.00
##
##   ScreenPorch        PoolArea        PoolQC       Fence      MiscFeature
##  Min.   :  0.00   Min.   :  0.000   Ex  :   2   GdPrv:  59   Gar2:   2
##  1st Qu.:  0.00   1st Qu.:  0.000   Fa  :   2   GdWo :  54   Othr:   2
##  Median :  0.00   Median :  0.000   Gd  :   3   MnPrv: 157   Shed:  49
##  Mean   : 15.06   Mean   :  2.759   NA's:1453   MnWw :  11   TenC:   1
##  3rd Qu.:  0.00   3rd Qu.:  0.000               NA's :1179   NA's:1406
##  Max.   :480.00   Max.   :738.000
##
##     MiscVal             MoSold           YrSold        SaleType
##  Min.   :    0.00   Min.   : 1.000   Min.   :2006   WD     :1267
##  1st Qu.:    0.00   1st Qu.: 5.000   1st Qu.:2007   New    : 122
##  Median :    0.00   Median : 6.000   Median :2008   COD    :  43
##  Mean   :   43.49   Mean   : 6.322   Mean   :2008   ConLD  :   9
##  3rd Qu.:    0.00   3rd Qu.: 8.000   3rd Qu.:2009   ConLI  :   5
##  Max.   :15500.00   Max.   :12.000   Max.   :2010   ConLw  :   5
##                                                     (Other):   9
##  SaleCondition    SalePrice
##  Abnorml: 101   Min.   : 34900
##  AdjLand:   4   1st Qu.:129975
##  Alloca :  12   Median :163000
##  Family :  20   Mean   :180921
##  Normal :1198   3rd Qu.:214000
##  Partial: 125   Max.   :755000
```

```
##
```

```
str(dfTrain)
```

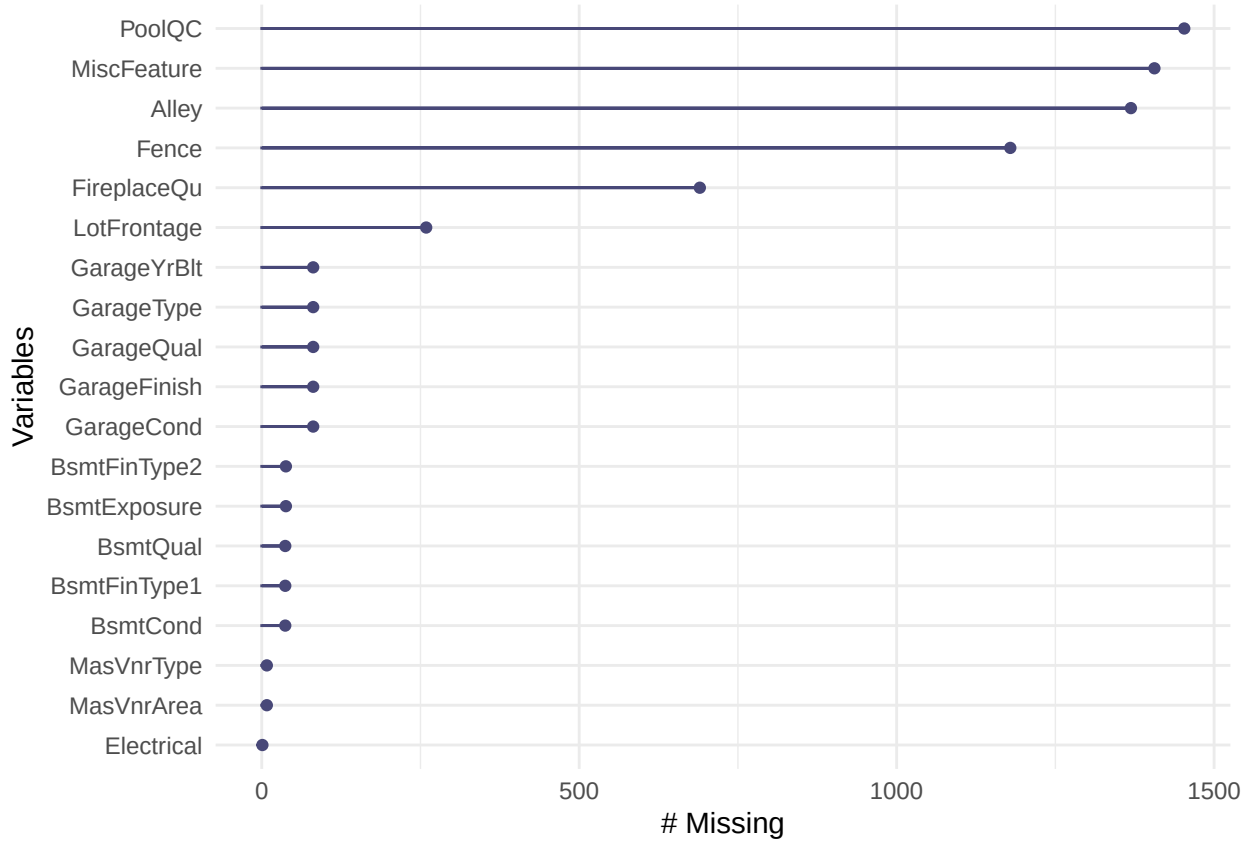
```
## 'data.frame':    1460 obs. of  81 variables:
##  $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
##  $ MSZoning     : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 4 4 4 4 5 4 ...
##  $ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
##  $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...
##  $ Street       : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Alley        : Factor w/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA NA NA NA ...
##  $ LotShape     : Factor w/ 4 levels "IR1","IR2","IR3",..: 4 4 1 1 1 1 4 1 4 4 ...
##  $ LandContour  : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Utilities    : Factor w/ 2 levels "AllPub","NoSeWa": 1 1 1 1 1 1 1 1 1 1 ...
##  $ LotConfig    : Factor w/ 5 levels "Corner","CulDSac",..: 5 3 5 1 3 5 5 1 5 1 ...
##  $ LandSlope    : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Neighborhood : Factor w/ 25 levels "Blmngtn","Blueste",..: 6 25 6 7 14 12 21 17 18 4 ...
##  $ Condition1   : Factor w/ 9 levels "Artery","Feedr",..: 3 2 3 3 3 3 3 5 1 1 ...
##  $ Condition2   : Factor w/ 8 levels "Artery","Feedr",..: 3 3 3 3 3 3 3 3 3 1 ...
##  $ BldgType     : Factor w/ 5 levels "1Fam","2fmCon",..: 1 1 1 1 1 1 1 1 1 2 ...
##  $ HouseStyle   : Factor w/ 8 levels "1.5Fin","1.5Unf",..: 6 3 6 6 6 1 3 6 1 2 ...
##  $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
##  $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
##  $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...
##  $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...
##  $ RoofStyle    : Factor w/ 6 levels "Flat","Gable",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ RoofMatl     : Factor w/ 8 levels "ClyTile","CompShg",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Exterior1st  : Factor w/ 15 levels "AsbShng","AsphShn",..: 13 9 13 14 13 13 13 7 4 9 ...
##  $ Exterior2nd  : Factor w/ 16 levels "AsbShng","AsphShn",..: 14 9 14 16 14 14 14 7 16 9 ...
##  $ MasVnrType   : Factor w/ 4 levels "BrkCmn","BrkFace",..: 2 3 2 3 2 3 4 4 3 3 ...
##  $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
##  $ ExterQual    : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 4 3 4 3 4 4 4 ...
##  $ ExterCond    : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ Foundation   : Factor w/ 6 levels "BrkTil","CBlock",..: 3 2 3 1 3 6 3 2 1 1 ...
##  $ BsmtQual     : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 3 3 4 3 3 1 3 4 4 ...
##  $ BsmtCond     : Factor w/ 4 levels "Fa","Gd","Po",..: 4 4 4 2 4 4 4 4 4 4 ...
##  $ BsmtExposure : Factor w/ 4 levels "Av","Gd","Mn",..: 4 2 3 4 1 4 1 3 4 4 ...
##  $ BsmtFinType1 : Factor w/ 6 levels "ALQ","BLQ","GLQ",..: 3 1 3 1 3 3 3 1 6 3 ...
##  $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
##  $ BsmtFinType2 : Factor w/ 6 levels "ALQ","BLQ","GLQ",..: 6 6 6 6 6 6 6 2 6 6 ...
##  $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
##  $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
##  $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
##  $ Heating      : Factor w/ 6 levels "Floor","GasA",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ HeatingQC    : Factor w/ 5 levels "Ex","Fa","Gd",..: 1 1 1 3 1 1 1 1 3 1 ...
##  $ CentralAir   : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Electrical   : Factor w/ 5 levels "FuseA","FuseF",..: 5 5 5 5 5 5 5 5 2 5 ...
##  $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
##  $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
##  $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...
##  $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
##  $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
```

```
##  $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
##  $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
##  $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
##  $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
##  $ KitchenQual  : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 3 3 4 3 4 4 4 ...
##  $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
##  $ Functional   : Factor w/ 7 levels "Maj1","Maj2",..: 7 7 7 7 7 7 7 7 3 7 ...
##  $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
##  $ FireplaceQu  : Factor w/ 5 levels "Ex","Fa","Gd",..: NA 5 5 3 5 NA 3 5 5 5 ...
##  $ GarageType   : Factor w/ 6 levels "2Types","Attchd",..: 2 2 2 6 2 2 2 2 6 2 ...
##  $ GarageYrBlt  : int  2003 1976 2001 1998 2000 1993 2004 1973 1931 1939 ...
##  $ GarageFinish : Factor w/ 3 levels "Fin","RFn","Unf": 2 2 2 3 2 3 2 2 3 2 ...
##  $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
##  $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
##  $ GarageQual   : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 2 3 ...
##  $ GarageCond   : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ PavedDrive   : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...
##  $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
##  $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
##  $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
##  $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
##  $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolQC       : Factor w/ 3 levels "Ex","Fa","Gd": NA NA NA NA NA NA NA NA NA NA ...
##  $ Fence        : Factor w/ 4 levels "GdPrv","GdWo",..: NA NA NA NA NA 3 NA NA NA NA ...
##  $ MiscFeature  : Factor w/ 4 levels "Gar2","Othr",..: NA NA NA NA NA 3 NA 3 NA NA ...
##  $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
##  $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
##  $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...
##  $ SaleType     : Factor w/ 9 levels "COD","Con","ConLD",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ SaleCondition: Factor w/ 6 levels "Abnorml","AdjLand",..: 5 5 5 1 5 5 5 5 1 5 ...
##  $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000 200000 129900 118000 ...
```

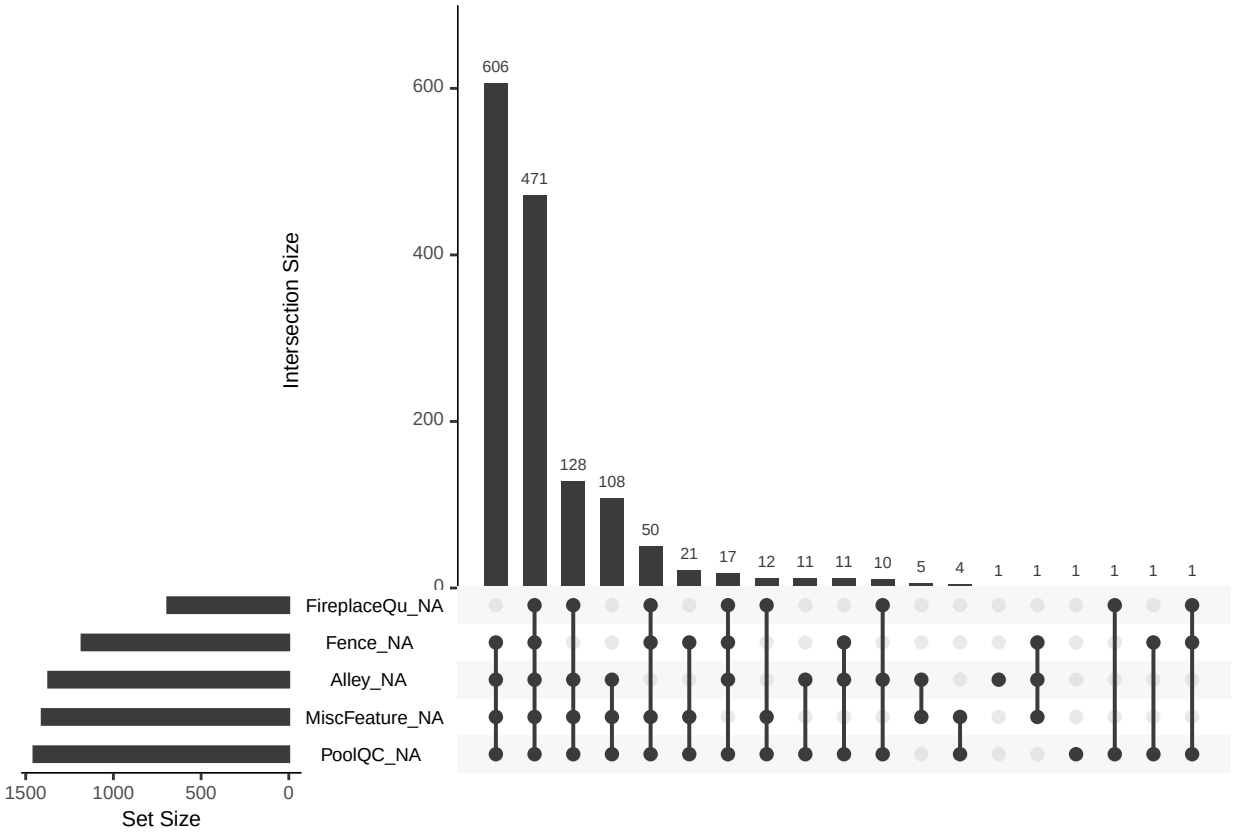
The dataset consists of 1460 observations and 81 variables, some numeric and some categorical. The target variable has a minimum of 34,950 and a maximum of 7,550,000. The low median compared to the mean suggests some skew.

B. Missing values There are missing values scattered throughout the dataset. We analyse them:

```
dfMissing <- dfTrain %>%
  dplyr::select(which(colMeans(is.na(.)) >0))

dfMissing2 <- dfTrain[rowSums(is.na(dfTrain)) > 0, ]

mm <- EHSummarize_MissingValues(dfMissing)

mm[[1]]
```


PoolQC
MiscFeature
Alley
Fence
FireplaceQu
LotFrontage
GarageYrBlt
GarageType
GarageQual
GarageFinish
GarageCond
BsmtFinType2
BsmtExposure
BsmtQual
BsmtFinType1
BsmtCond
MasVnrType
MasVnrArea
Electrical
Variables
0
500
1000
1500
Missing

```
mm[[3]]
```



A few categorical features like fireplace, fence, etc. take up the bulk of missings. They do not appear to be important enough to retain so we delete them (FireplaceQu, Fence, Alley, MiscFeature, PoolQC, and LotFrontage). We impute the mean for the rest.

```
dfTrain1 <- dfTrain %>%
  dplyr::select(-FireplaceQu, -Fence, -Alley, -MiscFeature, -PoolQC, -LotFrontage)

dfTest1 <- dfTest %>%
  dplyr::select(-FireplaceQu, -Fence, -Alley, -MiscFeature, -PoolQC, -LotFrontage)

dfTrain2 <- EHPrepare_MissingValues_Imputation(dfTrain1)
dfTest2 <- EHPrepare_MissingValues_Imputation(dfTest1)
```

C. Create dummy variables Now we create dummy variables for all of the character variables. Categorical NA's will be handled by adding a dummy variable for NA.

```
library(tidytable)

dfTrain3 <- EHPrepare_CreateDummies(dfTrain2)
dfTest3 <- EHPrepare_CreateDummies(dfTest2)
```

D. Reconcile training and test sets We check if the dataset is missing columns from the test dataset and if so, drop them from the training set. This way we don't risk making predictions on training set variables not found in the test set.

```
g <- EHPrepare_RestrictDataFrameColumnsToThoseInCommon(dfTrain3, dfTest3, exclude=c("SalePrice"))

dfTrain4 <- g[[1]]
dfTest4  <- g[[2]]
```

E. Multicollinearity We examine multicollinearity in the database We look at all of the pairs of correlations over .8 There are 24 pairs.

```
mult6 <- EHExplore_Multicollinearity(dfTrain4, printHighest=TRUE, threshold=.8, printHeatMap=FALSE)
```

```
##                    col1                  col2 correlation
## 1           TotalBsmtSF             X1stFlrSF   0.8195300
## 3             GrLivArea          TotRmsAbvGrd   0.8254894
## 5            GarageCars            GarageArea   0.8824754
## 7           MSZoning_FV  Neighborhood_Somerst   0.8628071
## 9        RoofStyle_Flat      RoofMatl_Tar.Grv   0.8349139
## 11 Exterior1st_AsbShng   Exterior2nd_AsbShng   0.8479167
## 12 Exterior1st_CemntBd   Exterior2nd_CmentBd   0.9741711
## 13 Exterior1st_HdBoard   Exterior2nd_HdBoard   0.8832714
## 14 Exterior1st_MetalSd   Exterior2nd_MetalSd   0.9730652
## 15 Exterior1st_Wd.Sdng   Exterior2nd_Wd.Sdng   0.8592439
## 21     Foundation_Slab           BsmtQual_NA   0.8017334
## 22     Foundation_Slab           BsmtCond_NA   0.8017334
## 23     Foundation_Slab       BsmtFinType1_NA   0.8017334
## 25         BsmtQual_NA           BsmtCond_NA   1.0000000
## 26         BsmtQual_NA       BsmtExposure_NA   0.9864076
## 27         BsmtQual_NA       BsmtFinType1_NA   1.0000000
## 28         BsmtQual_NA       BsmtFinType2_NA   0.9864076
## 31         BsmtCond_NA       BsmtExposure_NA   0.9864076
## 32         BsmtCond_NA       BsmtFinType1_NA   1.0000000
## 33         BsmtCond_NA       BsmtFinType2_NA   0.9864076
## 36     BsmtExposure_NA       BsmtFinType1_NA   0.9864076
## 37     BsmtExposure_NA       BsmtFinType2_NA   0.9729810
## 42     BsmtFinType1_NA       BsmtFinType2_NA   0.9864076
## 47        SaleType_New SaleCondition_Partial   0.9868190
```

Most of the pairs make sense - siding on the first floor will match siding on the sencond floor, the number of cars a garage can hold will be related to its area. We will address the multicollinearity more closely when we run the analysis.

2. Transformations

A. Log of SalePrice The skew in the dependent variable suggests a log transformation.

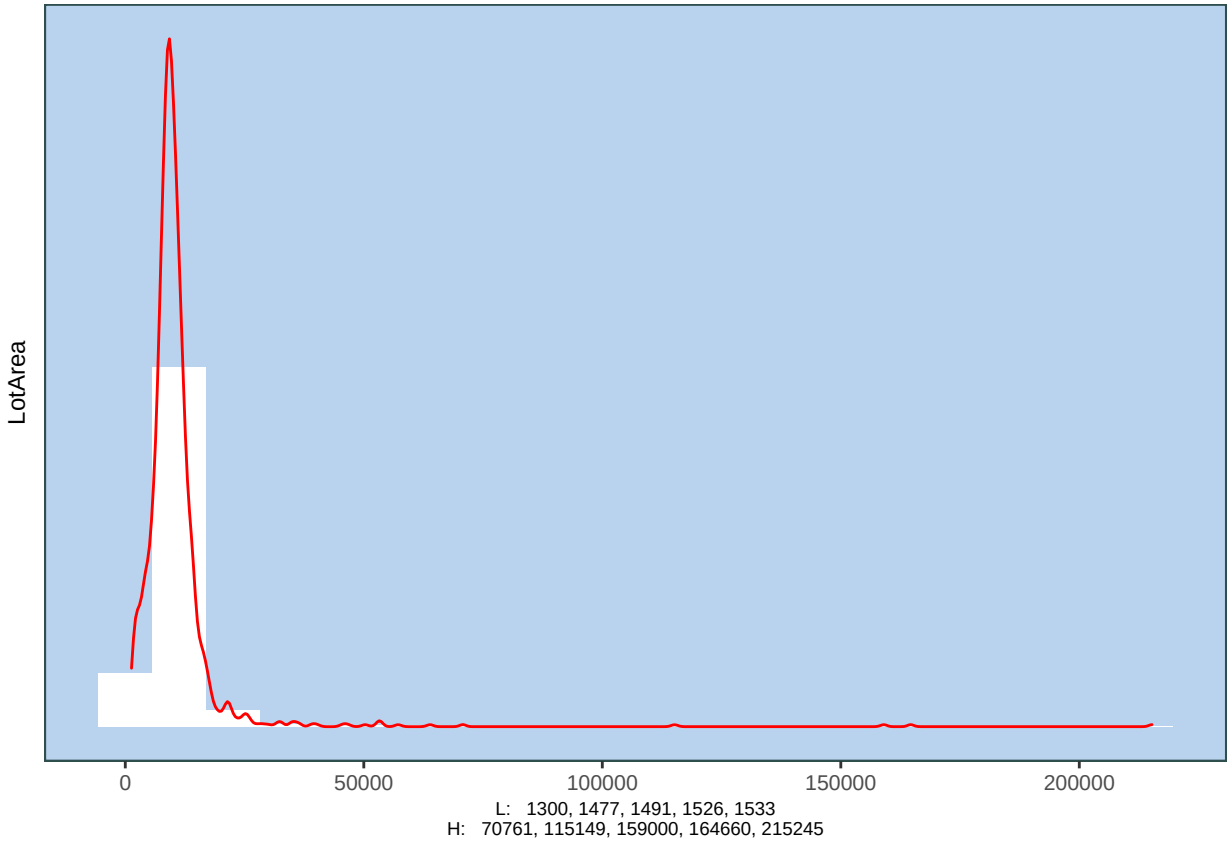
```
dfTrain5 <- na.omit(dfTrain4)
library(caTools)
library(Metrics)
dfTrain5$SalePrice <- log(dfTrain5$SalePrice)

dfTest5 <- dfTest4
```

```
#EHSummarize_StandardPlots(df6, "SalePrice")
```

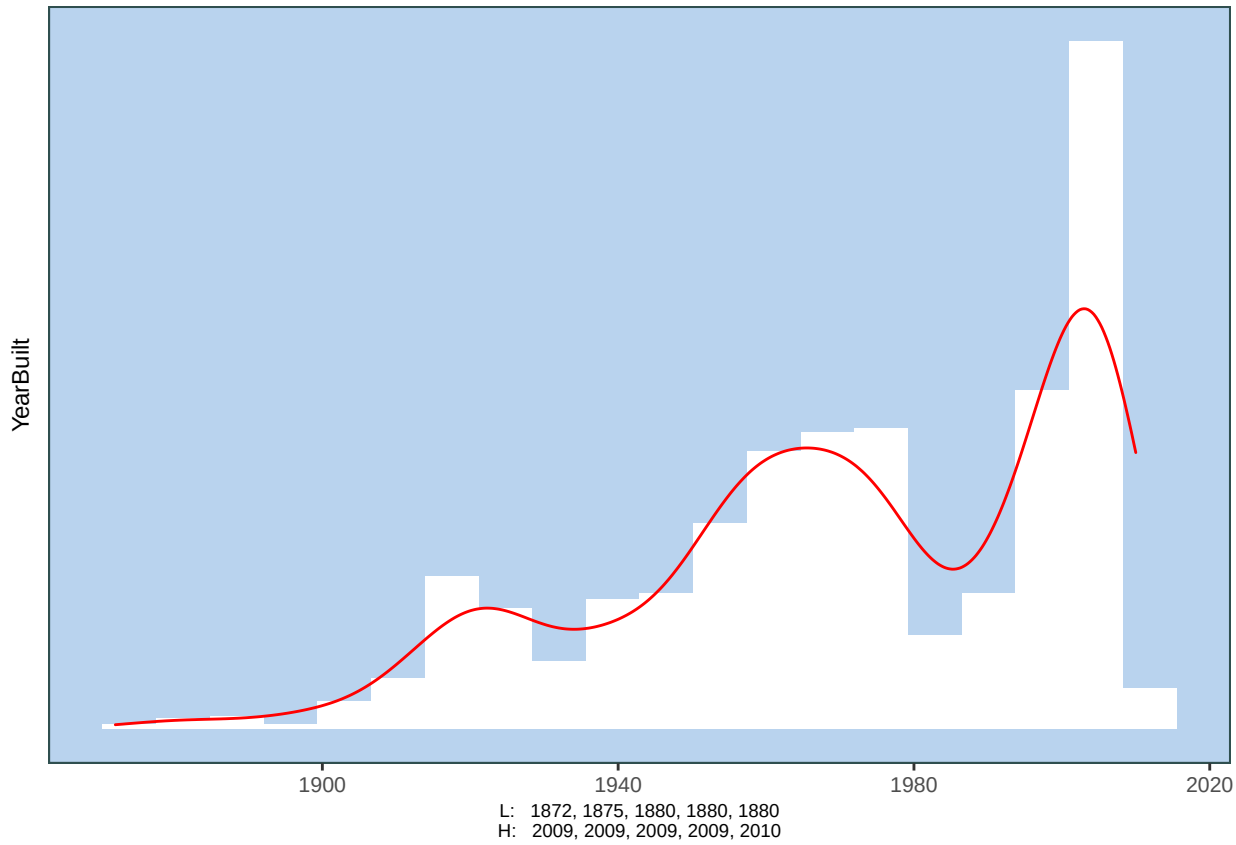
B. Other transformations A number of histograms suggest issues with some of the independent variables.

```
dfLook <- dfTrain5 %>%
  dplyr::select(LotArea, YearBuilt, YearRemodAdd, MasVnrArea, BsmtFinSF1, X1stFlrSF, GrLivArea, GarageY

EHSummarize_SingleColumn_Histograms(dfLook)
```

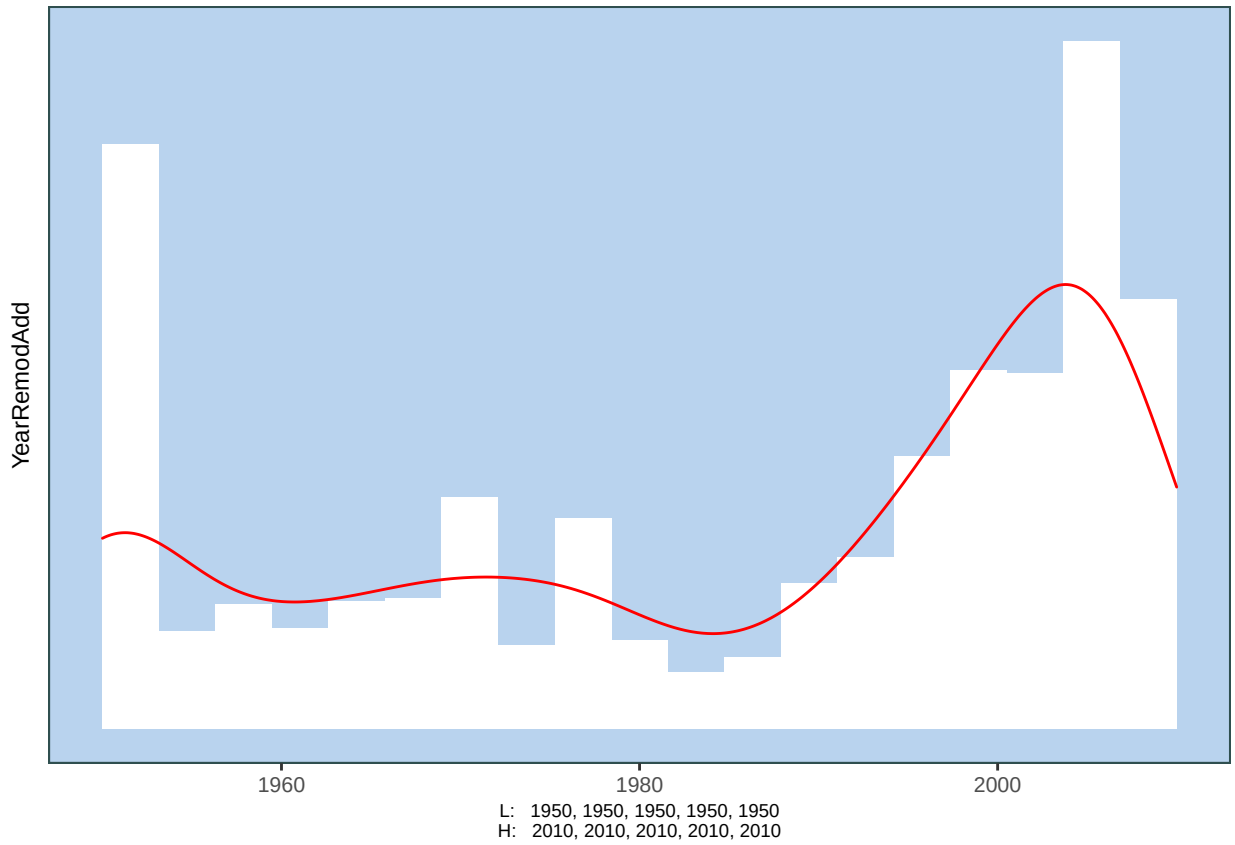
```
## [[1]]
```



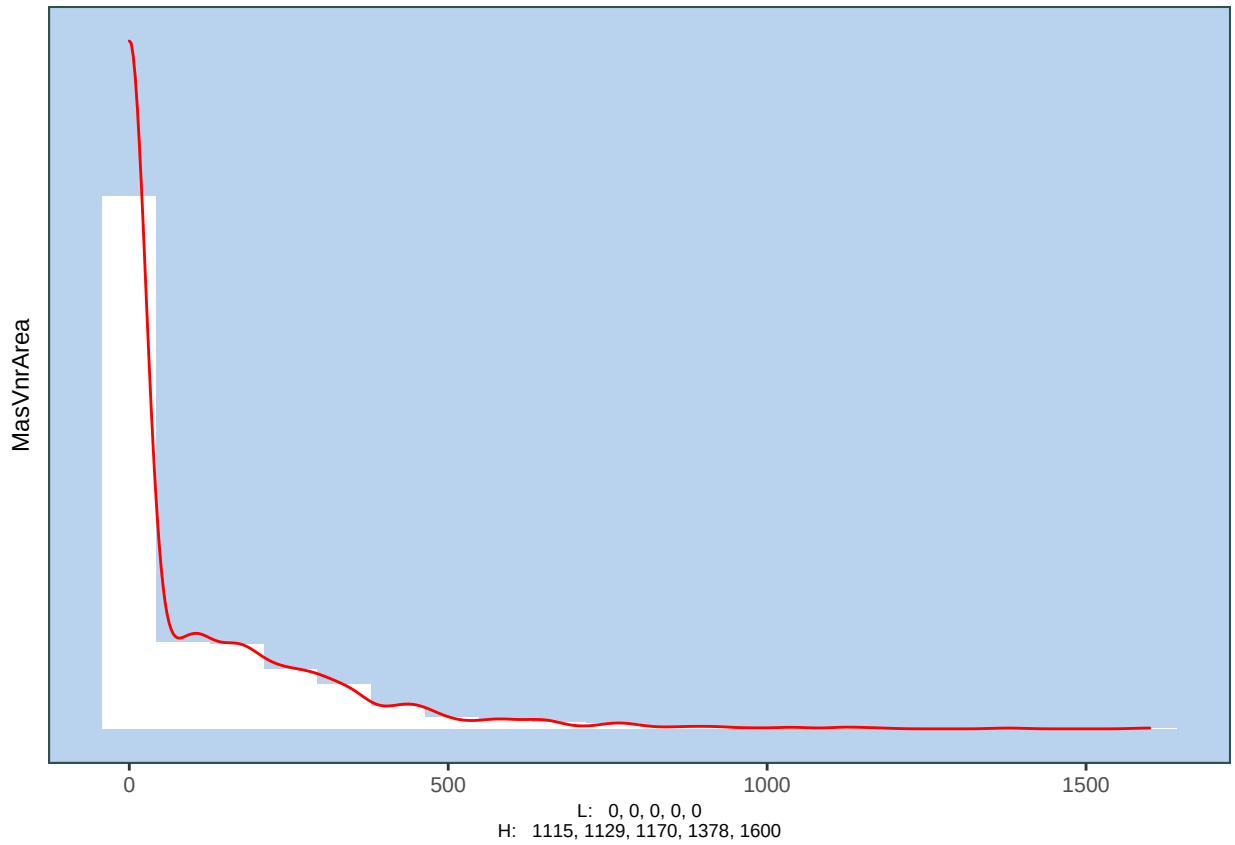

```
##
## [[2]]
```


YearBuilt
1900
1940
1980
2020
L: 1872, 1875, 1880, 1880, 1880
H: 2009, 2009, 2009, 2009, 2010

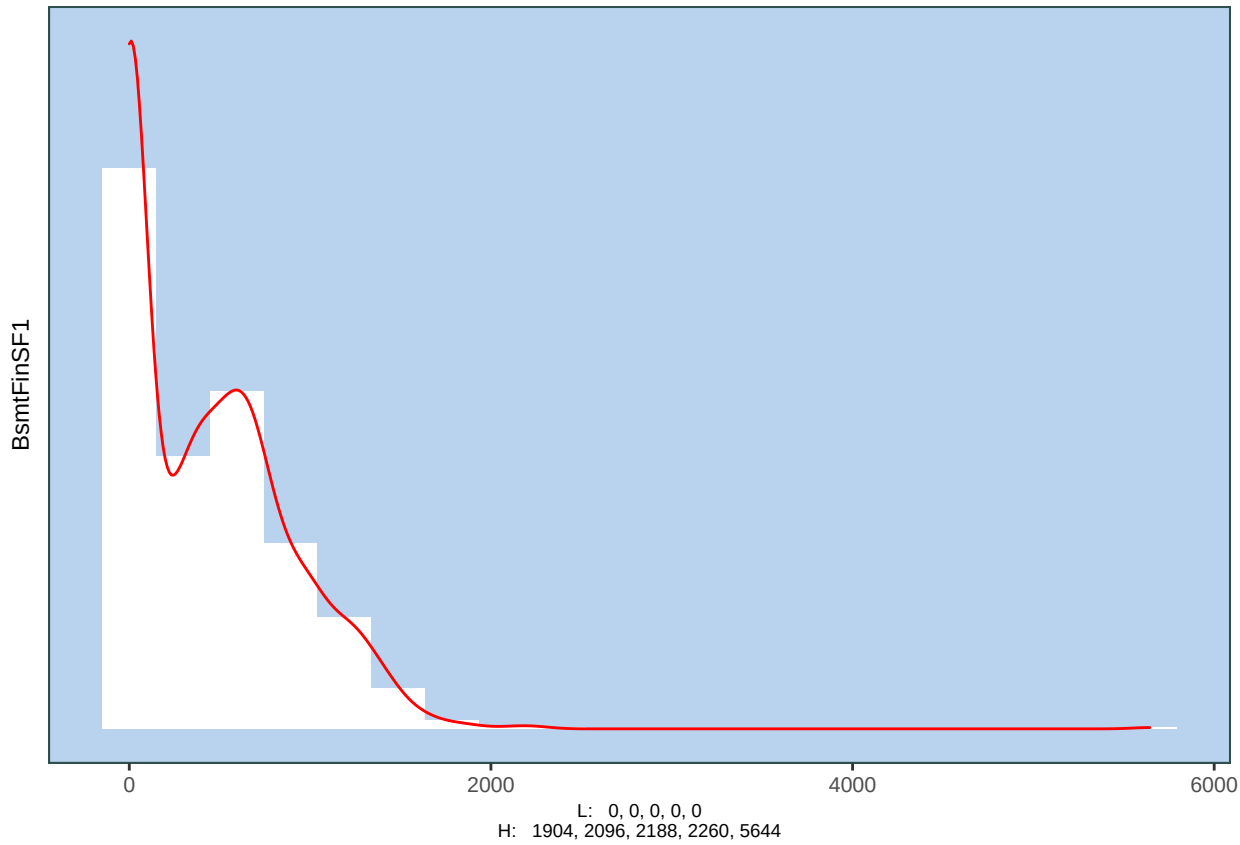
```
##
## [[3]]
```


YearRemodAdd
1960
1980
2000
L: 1950, 1950, 1950, 1950, 1950
H: 2010, 2010, 2010, 2010, 2010

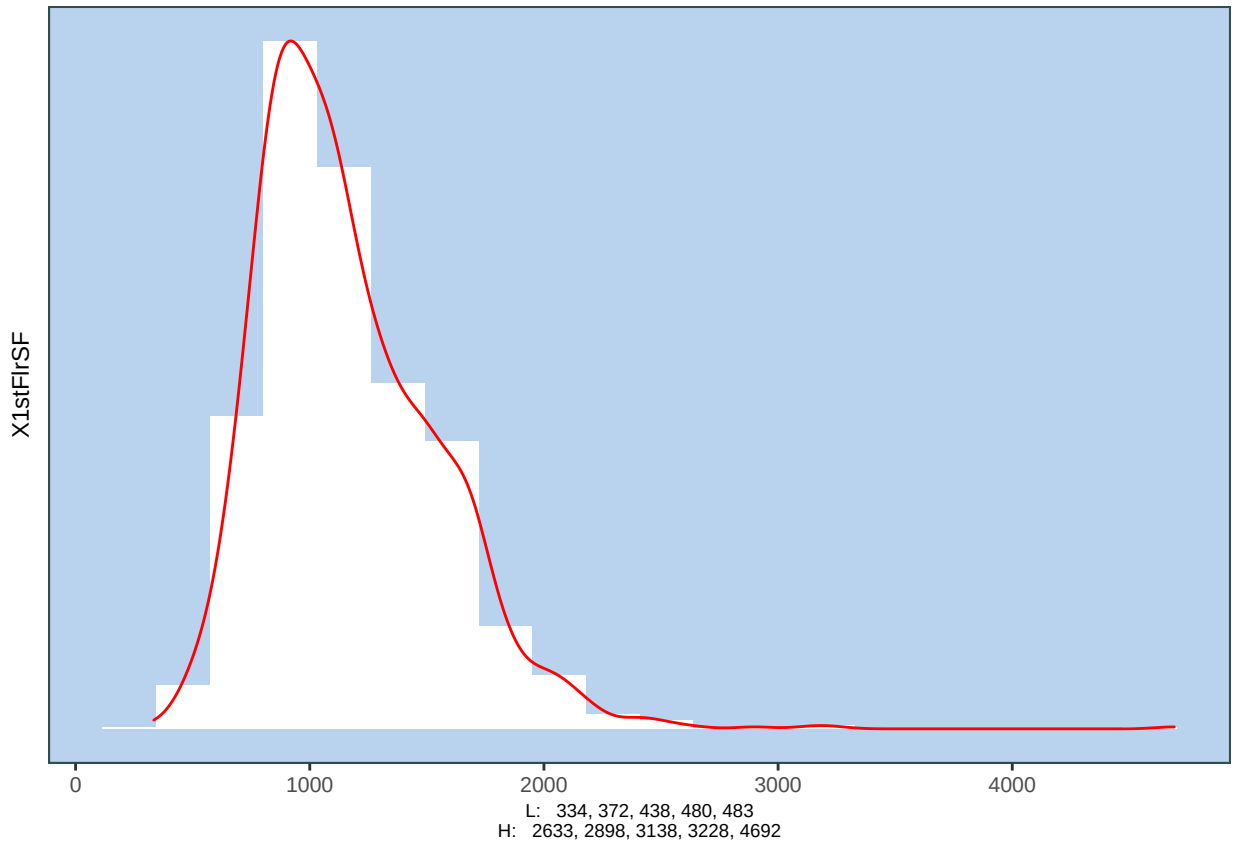
```
##
## [[4]]
```



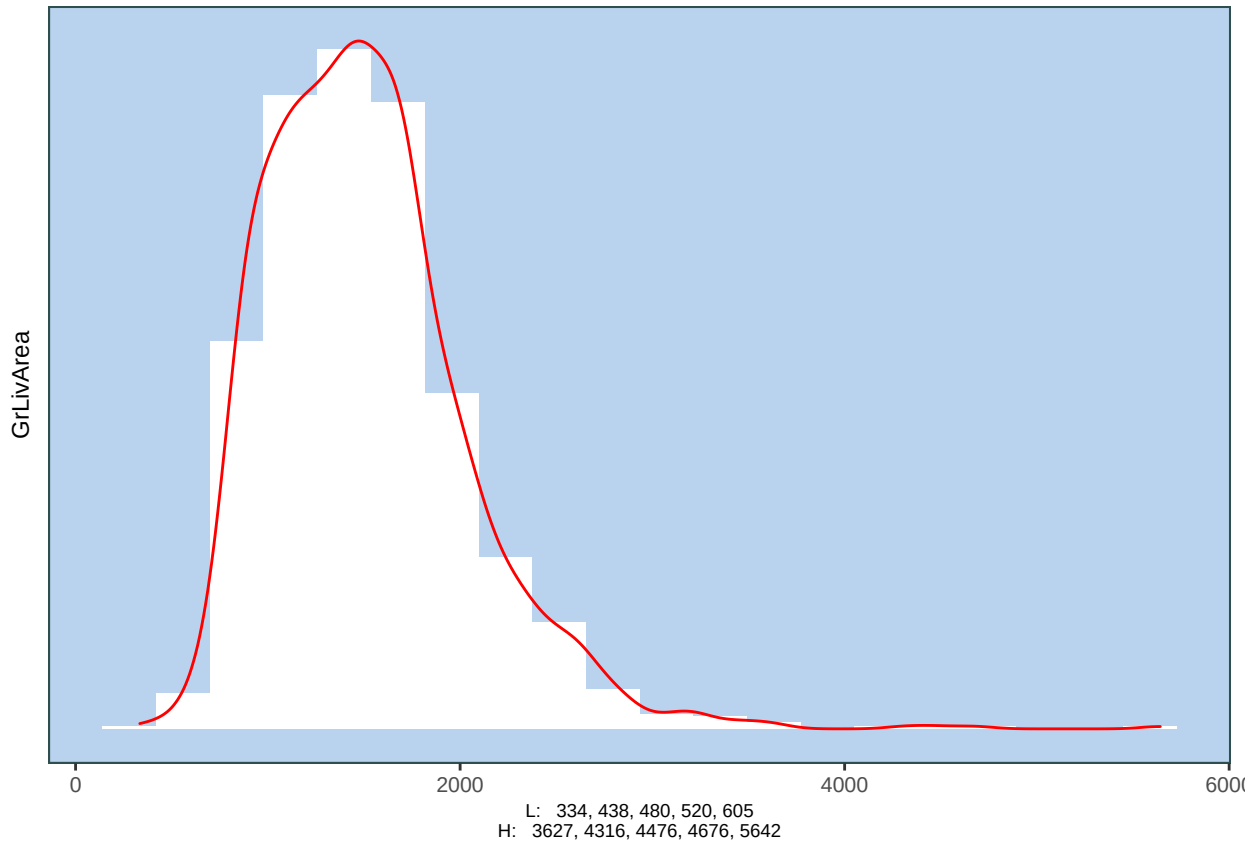

```
##
## [[5]]
```


BsmtFinSF1
0
2000
4000
6000
L: 0, 0, 0, 0, 0
H: 1904, 2096, 2188, 2260, 5644

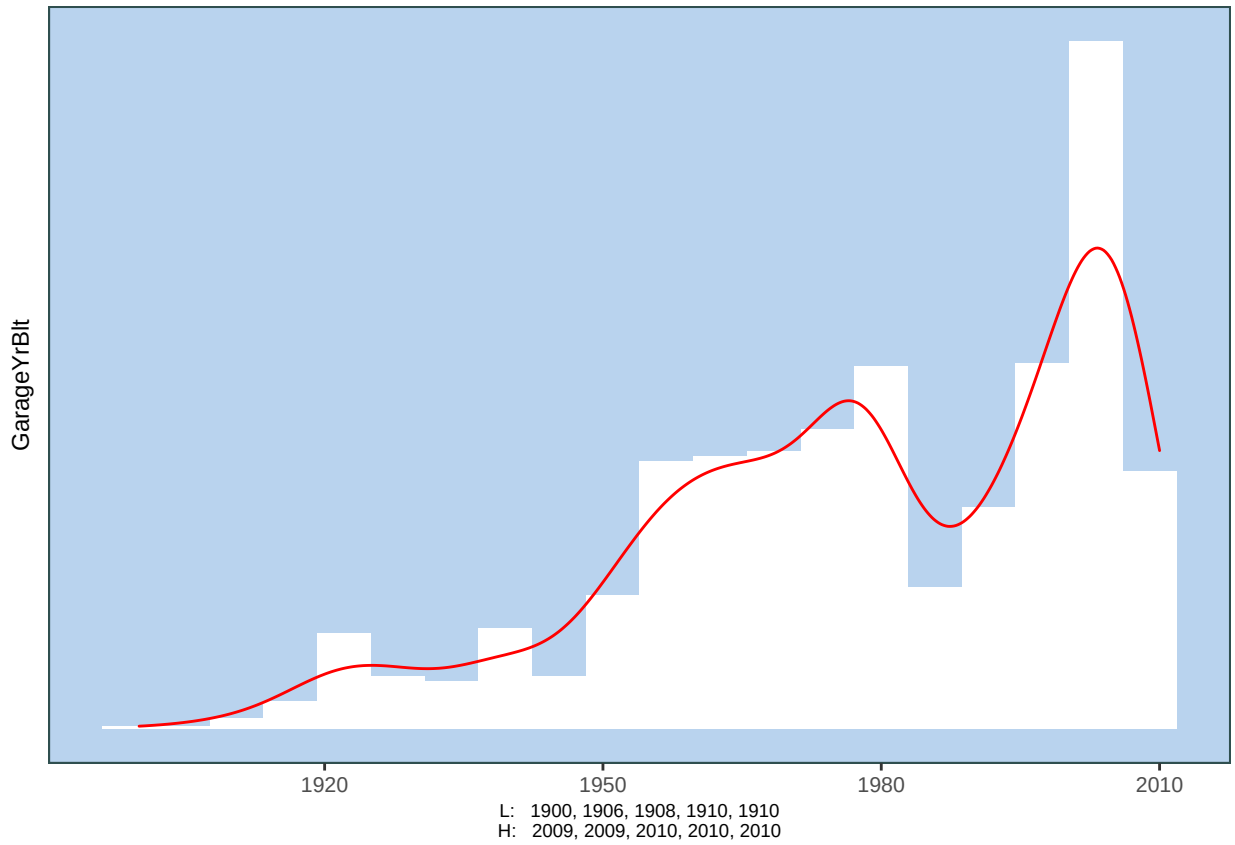
```
##
## [[6]]
```


X1stFlrSF
0
1000
2000
3000
4000
L: 334, 372, 438, 480, 483
H: 2633, 2898, 3138, 3228, 4692

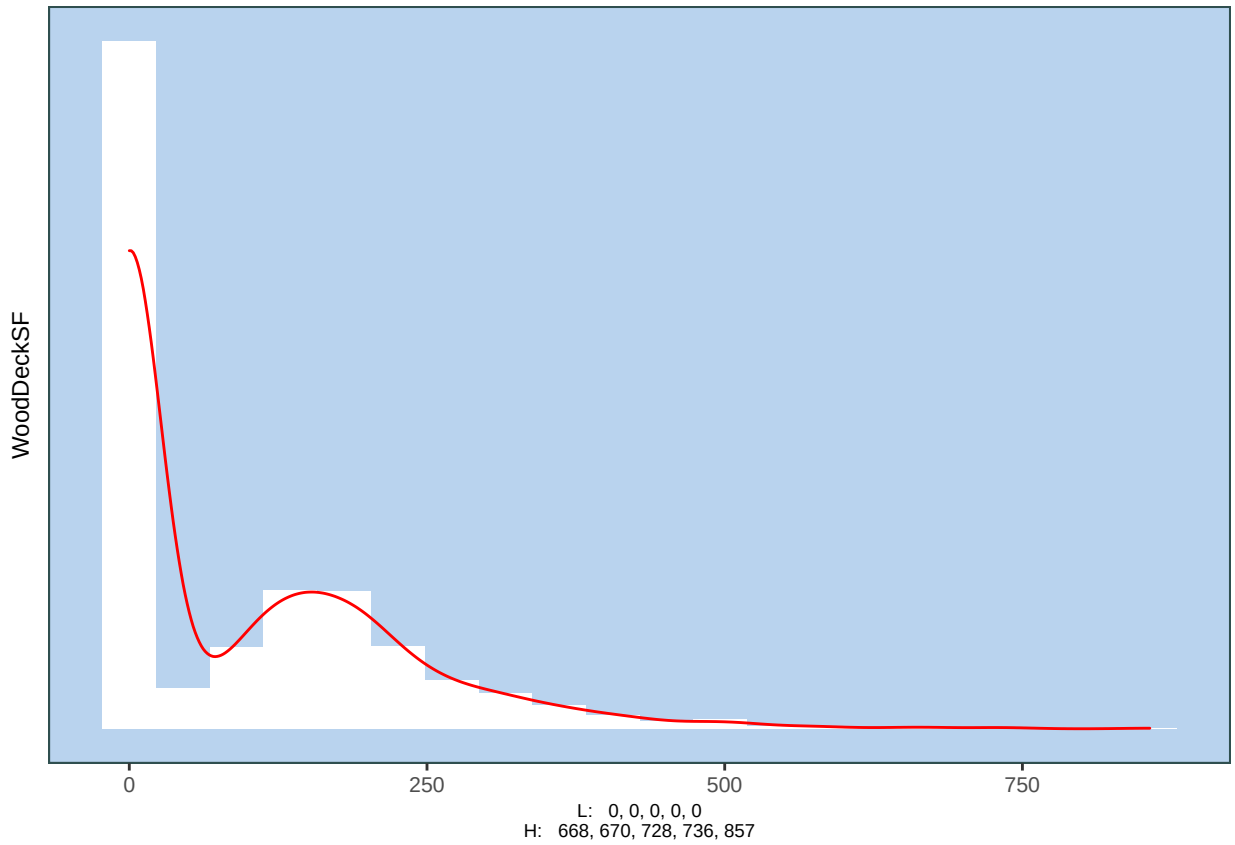
```
##
## [[7]]
```


GrLivArea
0
2000
4000
6000
L: 334, 438, 480, 520, 605
H: 3627, 4316, 4476, 4676, 5642

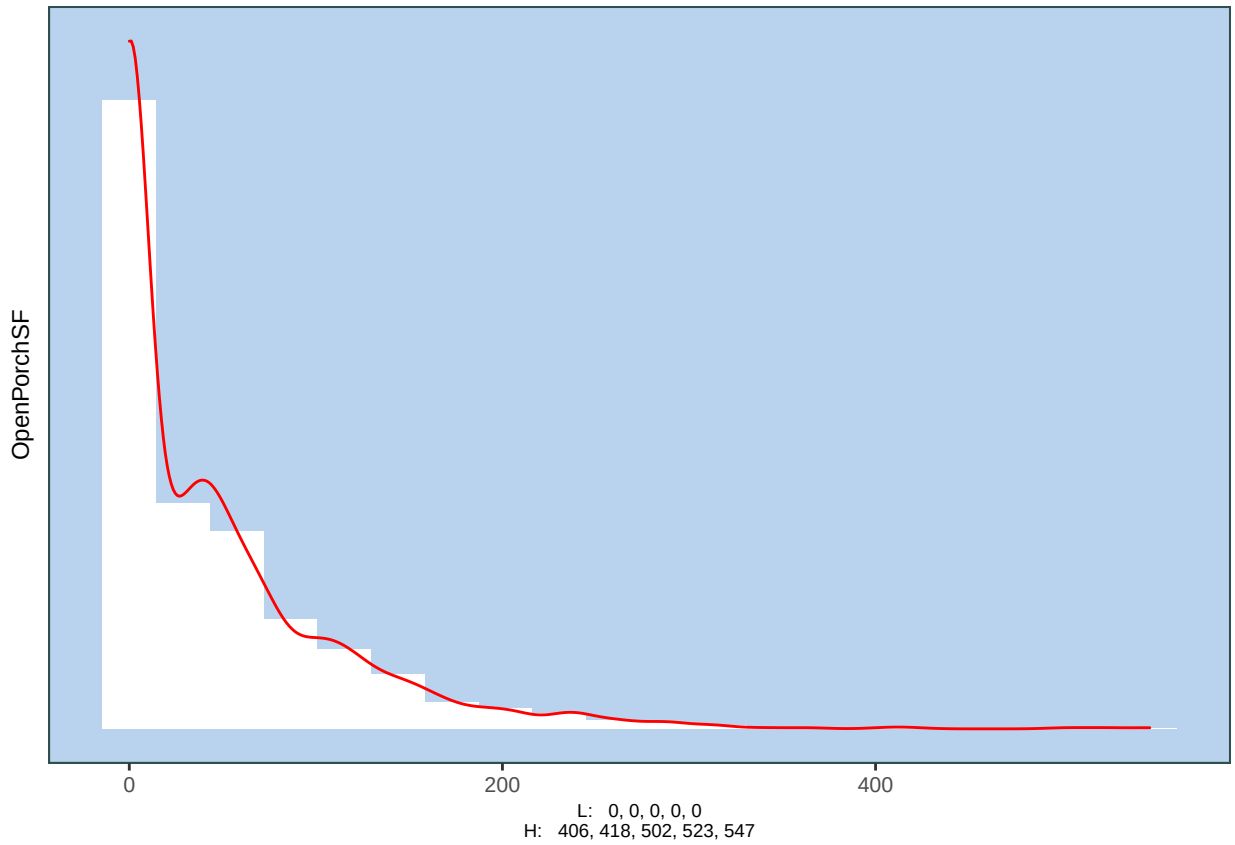
```
##
## [[8]]
```


GarageYrBlt
1920
1950
1980
2010
L: 1900, 1906, 1908, 1910, 1910
H: 2009, 2009, 2010, 2010, 2010

```
##
## [[9]]
```




```
##
## [[10]]
```



We can see some transformations might be useful. We: 1. Add a dummy variable to mark YearBuilt before and after 1920 2. We set YearRemodAdd = 1950 to 0, and create a dummy variable YearRemodUnknown to track it 3. We add dummies for NoFinBsmt, HasDeck, and HasPorch 4. We eliminate outliers by setting LotArea<35000, GrLivArea3500 and BsmtFinSF1<4000

```
dfTrain6 <- dfTrain5 %>%
  dplyr::mutate(BuiltAfter1920 = ifelse(YearBuilt>1920,1,0), YearRemodUnknown = ifelse(YearRemodAdd==195
  dplyr::filter(LotArea<35000, GrLivArea<3500, BsmtFinSF1<4000)

dfTest6 <- dfTest5 %>%
  dplyr::mutate(BuiltAfter1920 = ifelse(YearBuilt>1920,1,0), YearRemodUnknown = ifelse(YearRemodAdd==195
```

3. Model and Predict:

A. Base Model We run a regression using the stepAIC algorithm to minimize AIC.

```
#abc <- EHModel_Regression_StandardLM(dfTrain6, "SalePrice", splitRatio = 1, returnLM=TRUE)
abc <- lm(SalePrice ~ GrLivArea, dfTrain6)
```

Now we make predictions

```
makePredictions2 <- function(df)
{
predictions <- predict(df,newdata=dfTest6)
predictions <- data.frame(as.vector(predictions))
```

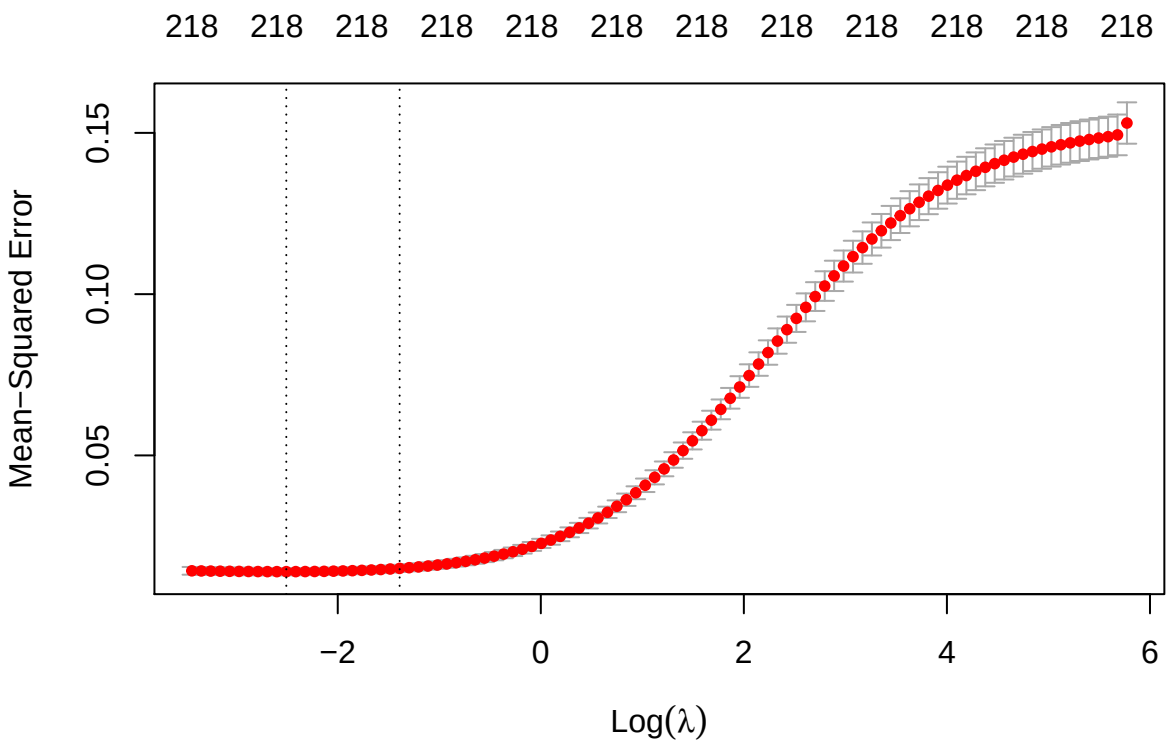
```
predictions$Id <- dfTest6$Id
predictions[,c(1,2)] <- predictions[,c(2,1)]
colnames(predictions) <- c("Id", "SalePrice")
predictions[is.na(predictions)] <- log(mean(dfTrain$SalePrice))
predictions$SalePrice <- exp(predictions$SalePrice)
#write_csv(predictions, "C:\\Users\\eric.hirsch\\Desktop\\predictionsABCLess.csv")
write_csv(predictions, "D:\\RStudio\\CUNY_621\\Final\\predictionsABC.csv")
}

makePredictions2(abc)
```

We achieve a score of .14586 on kaggle.

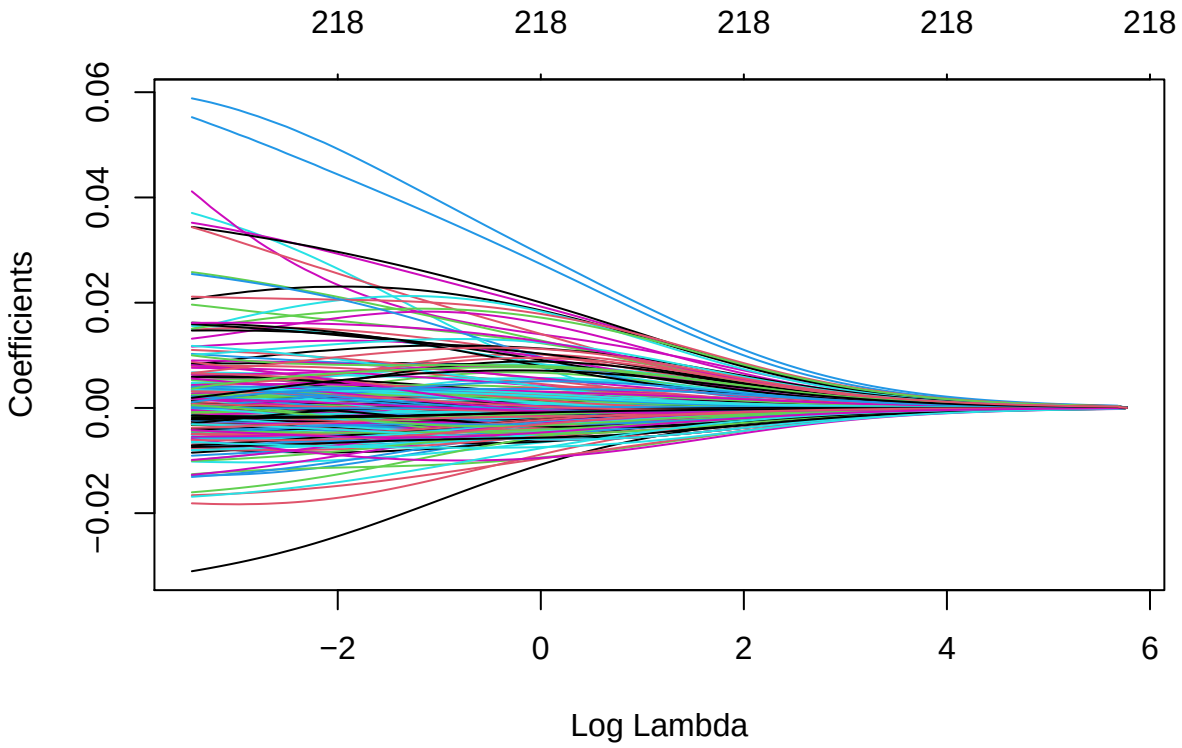
```
library(glmnet)

df7a <- EHData::EHPrepare_ScaleAllButTarget(dfTrain6, "SalePrice")
dfSubmit7a <- as.data.frame(scale(dfTest6))

df7b <- df7a %>%
    dplyr::select(-SalePrice)

y <- df7a$SalePrice
x <- data.matrix(df7b)
xSub <- data.matrix(dfSubmit7a)

model <- glmnet(x, y, alpha = 0)
```

B. Now we try Ridge regression: R makes it easy to find the best lambda by using kfold validation:

```
#We find the optimal lambda by performing k-fold cross validation:

mcv <- cv.glmnet(x, y, alpha = 0)
plot(mcv)
```




```
lambda1 <- mcv$lambda.min

plot(model, xvar = "lambda")
```




```
m10 <- glmnet(x, y, alpha = 0, lambda = lambda1)
coef(m10)
```

```
## 219 x 1 sparse Matrix of class "dgCMatrix"
##                                    s0
## (Intercept)            1.201515e+01
## Id                    -3.133506e-03
## MSSubClass             2.541243e-04
## LotArea                1.762322e-02
## OverallQual            5.341424e-02
## OverallCond            3.082037e-02
## YearBuilt              2.816658e-02
## YearRemodAdd           8.398431e-03
## MasVnrArea             4.576357e-03
## BsmtFinSF1             2.282256e-02
## BsmtFinSF2             2.587535e-03
## BsmtUnfSF              6.371615e-03
## TotalBsmtSF            3.148372e-02
## X1stFlrSF              3.147916e-02
## X2ndFlrSF              2.844491e-02
## LowQualFinSF          -5.550169e-04
## GrLivArea              4.846274e-02
## BsmtFullBath           1.484837e-02
## BsmtHalfBath           9.328698e-04
## FullBath               2.275671e-02
## HalfBath               1.526141e-02
```

```
## BedroomAbvGr           2.347086e-03
## KitchenAbvGr          -1.197557e-02
## TotRmsAbvGrd           1.900976e-02
## Fireplaces             1.572615e-02
## GarageYrBlt            1.027337e-02
## GarageCars             2.072197e-02
## GarageArea             1.740993e-02
## WoodDeckSF             9.148706e-03
## OpenPorchSF            5.435481e-03
## EnclosedPorch          5.103035e-03
## X3SsnPorch             5.662279e-03
## ScreenPorch            1.011398e-02
## PoolArea               5.255949e-03
## MiscVal               -1.647101e-03
## MoSold                 4.296459e-05
## YrSold                -1.341752e-03
## MSZoning_C..all.      -2.725191e-02
## MSZoning_FV            7.886800e-03
## MSZoning_RM           -1.163201e-02
## Street_Grvl           -3.095042e-03
## LotShape_IR1           7.229968e-04
## LotShape_IR2           3.245097e-03
## LotShape_IR3           4.836080e-04
## LandContour_Bnk       -1.573942e-03
## LandContour_HLS        3.711035e-03
## LandContour_Low       -3.178889e-03
## LotConfig_Corner       3.218985e-03
## LotConfig_CulDSac      7.242164e-03
## LotConfig_FR2         -5.425615e-03
## LotConfig_FR3         -1.586681e-03
## LandSlope_Mod          2.255921e-03
## LandSlope_Sev         -6.750588e-03
## Neighborhood_Blmngtn   5.925355e-04
## Neighborhood_Blueste  -2.524816e-03
## Neighborhood_BrDale   -8.398179e-03
## Neighborhood_BrkSide   6.377692e-03
## Neighborhood_ClearCr   3.556907e-03
## Neighborhood_Crawfor   2.272735e-02
## Neighborhood_Edwards  -1.025742e-02
## Neighborhood_Gilbert  -4.173445e-04
## Neighborhood_IDOTRR   -3.104836e-03
## Neighborhood_MeadowV  -1.803383e-02
## Neighborhood_Mitchel  -5.100591e-03
## Neighborhood_NPkVill  -1.565946e-03
## Neighborhood_NWAmes   -5.136261e-03
## Neighborhood_NoRidge   1.262470e-02
## Neighborhood_NridgHt   1.468311e-02
## Neighborhood_OldTown  -6.496458e-03
## Neighborhood_SWISU     3.070593e-03
## Neighborhood_Sawyer   -4.569020e-03
## Neighborhood_SawyerW   3.254197e-03
## Neighborhood_Somerst   8.733120e-03
## Neighborhood_StoneBr   1.527004e-02
## Neighborhood_Timber    1.988751e-03
```

```
## Neighborhood_Veenker   3.880426e-03
## Condition1_Artery     -1.147914e-02
## Condition1_PosA       -1.311204e-03
## Condition1_PosN       -4.684259e-04
## Condition1_RRAe       -7.258901e-03
## Condition1_RRAn       -4.388201e-03
## Condition1_RRNe       -8.489757e-04
## Condition1_RRNn        4.669209e-04
## Condition2_Artery     -2.744204e-03
## Condition2_Feedr       1.094076e-03
## Condition2_PosA        1.855663e-03
## Condition2_PosN       -2.093885e-03
## BldgType_2fmCon       -2.106806e-04
## BldgType_Duplex       -8.115042e-03
## BldgType_Twnhs        -8.302200e-03
## BldgType_TwnhsE       -4.582917e-03
## HouseStyle_1.5Fin      5.499030e-03
## HouseStyle_1.5Unf      2.747528e-03
## HouseStyle_2.5Unf      4.420802e-03
## HouseStyle_SFoyer     -2.330722e-04
## HouseStyle_SLvl        3.674931e-04
## RoofStyle_Flat         2.903013e-03
## RoofStyle_Gambrel      1.603602e-03
## RoofStyle_Hip          1.094832e-03
## RoofStyle_Mansard      3.223311e-03
## RoofStyle_Shed         3.340161e-03
## RoofMatl_Tar.Grv      -2.895901e-03
## RoofMatl_WdShake       1.268145e-03
## RoofMatl_WdShngl      -1.894941e-03
## Exterior1st_AsbShng   -2.610116e-04
## Exterior1st_AsphShn    6.384863e-06
## Exterior1st_BrkComm   -6.537100e-03
## Exterior1st_BrkFace    1.031998e-02
## Exterior1st_CBlock    -1.696402e-04
## Exterior1st_CemntBd   -7.526228e-04
## Exterior1st_HdBoard   -8.010106e-03
## Exterior1st_MetalSd   -1.949174e-03
## Exterior1st_Plywood   -4.298283e-03
## Exterior1st_Stucco     1.647143e-03
## Exterior1st_Wd.Sdng   -1.059012e-02
## Exterior1st_WdShing   -3.824942e-03
## Exterior2nd_AsbShng   -2.996355e-03
## Exterior2nd_AsphShn    5.603697e-04
## Exterior2nd_Brk.Cmn   -1.910185e-03
## Exterior2nd_BrkFace   -4.647412e-03
## Exterior2nd_CBlock    -1.746664e-04
## Exterior2nd_CmentBd    1.113080e-03
## Exterior2nd_HdBoard   -6.990352e-03
## Exterior2nd_ImStucc   -6.287353e-04
## Exterior2nd_MetalSd   -1.909483e-03
## Exterior2nd_Plywood   -7.471830e-03
## Exterior2nd_Stone     -1.597835e-03
## Exterior2nd_Stucco    -7.664655e-04
## Exterior2nd_Wd.Sdng   -4.522464e-04
```

```
## Exterior2nd_Wd.Shng   -3.087904e-03
## MasVnrType_BrkCmn     -6.326883e-03
## MasVnrType_NA         -2.018904e-03
## MasVnrType_Stone       6.578705e-03
## ExterQual_Ex           3.364147e-03
## ExterQual_Fa          -1.919526e-03
## ExterCond_Ex           2.633701e-03
## ExterCond_Fa          -5.735560e-03
## ExterCond_Gd          -3.681410e-03
## ExterCond_Po          -2.623128e-03
## Foundation_BrkTil     -4.049352e-03
## Foundation_Slab       -1.000576e-03
## Foundation_Stone       4.217615e-03
## Foundation_Wood       -3.867307e-03
## BsmtQual_Ex            1.143789e-02
## BsmtQual_Fa            4.754575e-04
## BsmtQual_NA           -6.047714e-04
## BsmtCond_Fa           -5.421374e-03
## BsmtCond_Gd            2.254375e-03
## BsmtCond_NA           -7.898301e-04
## BsmtCond_Po            2.395445e-03
## BsmtExposure_Av        4.558670e-03
## BsmtExposure_Gd        1.457335e-02
## BsmtExposure_Mn        3.639242e-03
## BsmtExposure_NA       -1.236478e-03
## BsmtFinType1_ALQ      -3.100295e-03
## BsmtFinType1_BLQ      -7.301036e-03
## BsmtFinType1_LwQ      -5.133744e-03
## BsmtFinType1_NA       -6.631989e-04
## BsmtFinType1_Unf      -4.897788e-03
## BsmtFinType2_ALQ       2.135341e-03
## BsmtFinType2_BLQ      -5.635419e-03
## BsmtFinType2_GLQ       3.630114e-03
## BsmtFinType2_NA       -7.760969e-04
## BsmtFinType2_Rec      -2.526781e-03
## Heating_GasW           5.915704e-03
## Heating_Grav          -9.063935e-03
## Heating_Wall           2.530404e-03
## HeatingQC_Fa          -2.864430e-03
## HeatingQC_Gd          -2.983698e-03
## HeatingQC_Po          -1.804511e-03
## CentralAir_N          -1.568817e-02
## Electrical_FuseA      -5.987276e-04
## Electrical_FuseF       9.526663e-04
## Electrical_FuseP      -1.340982e-03
## KitchenQual_Ex         1.616981e-02
## KitchenQual_Fa         1.169909e-04
## Functional_Maj1       -6.038288e-03
## Functional_Maj2       -1.413922e-02
## Functional_Min1       -5.637884e-03
## Functional_Min2       -5.893809e-03
## Functional_Mod        -8.190862e-03
## Functional_Sev        -6.027016e-03
## GarageType_2Types     -4.936727e-03
```

```
## GarageType_Basment    -1.838910e-03
## GarageType_BuiltIn     1.683662e-03
## GarageType_CarPort    -1.314438e-03
## GarageType_Detchd     -8.038252e-03
## GarageType_NA         -3.573268e-03
## GarageFinish_Fin       5.462489e-03
## GarageFinish_NA       -3.662982e-03
## GarageQual_Fa         -3.711385e-03
## GarageQual_Gd          3.732808e-03
## GarageQual_NA         -3.640180e-03
## GarageQual_Po         -1.107423e-03
## GarageCond_Ex          3.650520e-04
## GarageCond_Fa         -4.626332e-03
## GarageCond_Gd         -5.508078e-05
## GarageCond_NA         -3.586673e-03
## GarageCond_Po          3.852822e-03
## PavedDrive_N          -6.966399e-03
## PavedDrive_P          -3.141862e-03
## SaleType_COD          -8.787954e-04
## SaleType_CWD           3.984195e-03
## SaleType_Con           3.401284e-03
## SaleType_ConLD         6.474170e-03
## SaleType_ConLI        -1.620913e-03
## SaleType_ConLw         2.724617e-03
## SaleType_New           8.536291e-03
## SaleType_Oth           2.910276e-03
## SaleCondition_Abnorml -1.533326e-02
## SaleCondition_AdjLand  1.241752e-03
## SaleCondition_Alloca  -1.886021e-03
## SaleCondition_Family  -6.130207e-03
## SaleCondition_Partial  5.886652e-03
## BuiltAfter1920         3.135186e-03
## YearRemodUnknown      -7.210878e-03
## NoFinBsmt             -5.055903e-03
## HasDeck                4.108807e-03
## HasPorch               8.663457e-03
```

```
x2 <- tidy(coef(m10))

y_predicted <- predict(m10, s = lambda1, newx = xSub)
```

We predict values based on our Ridge regressions.

```
mSubmit7 = as.matrix(dfSubmit7a)
mod=m10

predictions <- predict(mod, s = lambda1, newx = xSub)
predictions <- data.frame(as.vector(predictions))
predictions$Id <- dfTest6$Id
predictions[,c(1,2)] <- predictions[,c(2,1)]
colnames(predictions) <- c("Id", "SalePrice")
predictions[is.na(predictions)] <- log(mean(dfTrain$SalePrice))
predictions$SalePrice <- exp(predictions$SalePrice)
write_csv(predictions, "D:\\RStudio\\CUNY_621\\Final\\predictionsRidge2.csv")
```

Ridge regression performs the best, with a score of .14047. This puts us at 1690 out of 4216 individuals.

New code goes here: THE CLEAN DATASET IS dfTrain6, not df7! The test set (for submission) is dfTest6.