# OR568 - Final Report- Predicting House Prices

**Group 3 : Sriudaya Damuluri, Dilip Kumar Molugu, Navya Shivaji Mote, Nikhil Reddy Pathuri, Luyao Shen, Xin Zhang**

The main objective of our project is to determine the various aspects of a house that contribute to the sales price. We explore the various predictors and try to predict the sales price based on the available data.

1. **The sources of the data sets.**

We found this data set on Kaggle.com. It was put together from a larger real time data set known as The Ames Housing data. The original data set had 3970 observations with 113 variables. Some of the observations belonged to stand-alone garages, condos, and storage areas which was not relevant to the analysis of house prices. Therefore, these observations were not included. The house prices data set also contains recent sales data on a property when compared to the original data which had approximately 100 houses with changed ownership multiple times in a period of 4 years.

The data set is already split into a training set and a test set. Each having 1460 observations. The training set has 81 columns and the test set has 80(excluding Sale Price). The training set consists of 23 nominal, 23 ordinal, 15 discrete/interval and 20 ratio/continuous variables. Whereas the test set consists of 23 nominal, 23 ordinal, 15 discrete/interval and 19 ratio/continuous variables.

The House Prices dataset used by us for this project can be found at : https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

2. **The predictors in the raw data sets and the response(s) you are trying to predict.**

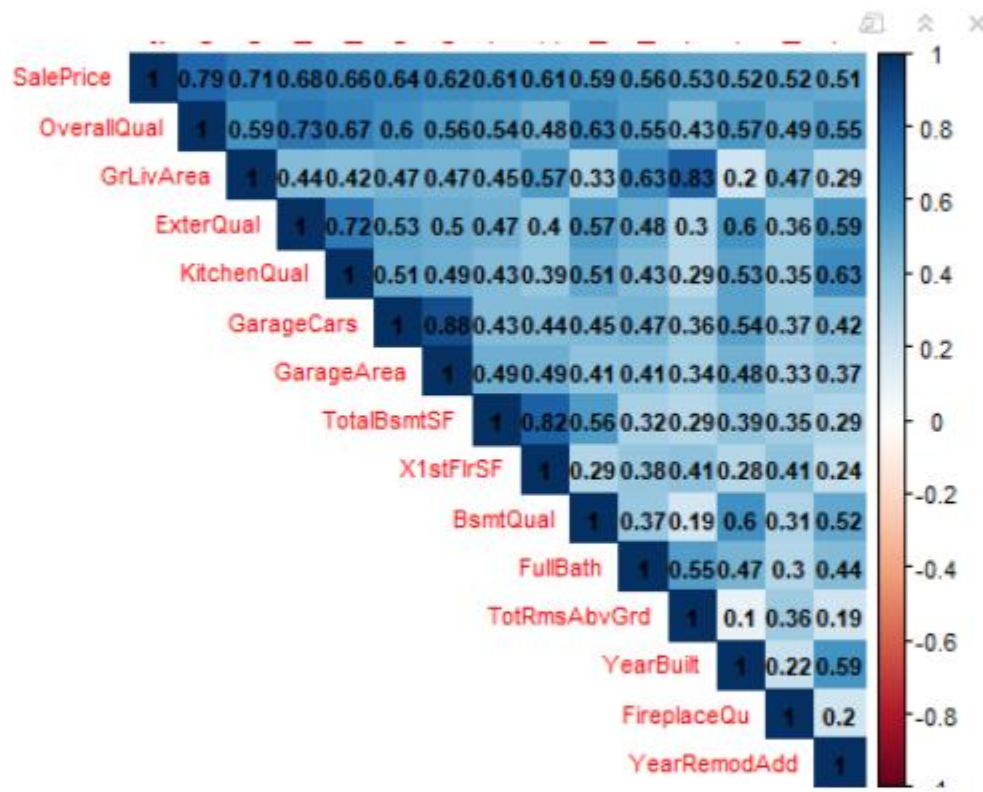
A detailed explanation of the predictors can be found in the text file named :decock.txt
The response we are trying to predict is the Sale Price of the house in USD.

3. **Any pre-processing you applied to transform the raw data sets into a format that you can apply a predictive model. This includes but not restricted to Box-Cox transformation, centering, scaling, removal of near zero variance predictors, and/or removal of bad entries and missing values.**

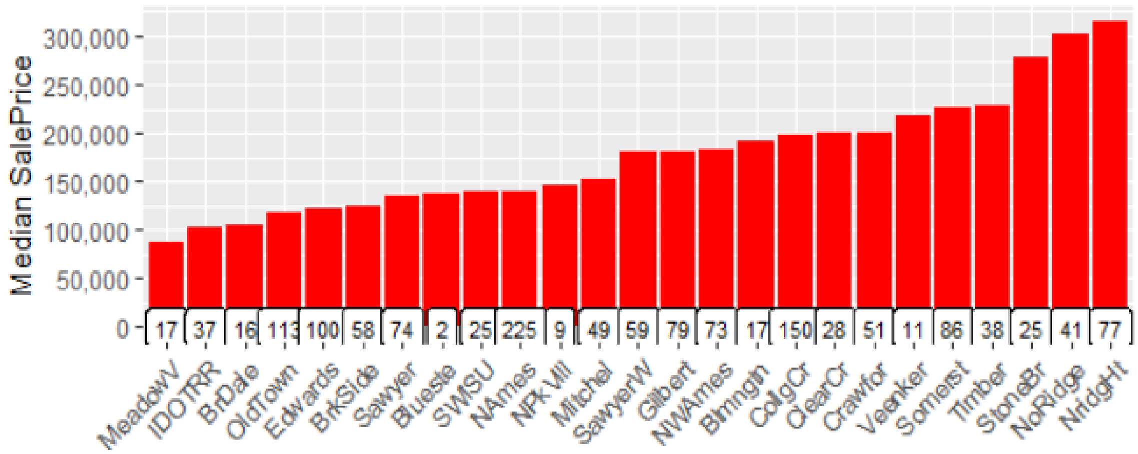
We removed columns 'FireplaceQu' and 'LotFrontage' as they had a large number of missing values. Removing these columns also improved our test set predictions. We were then left with 32 columns with missing values.

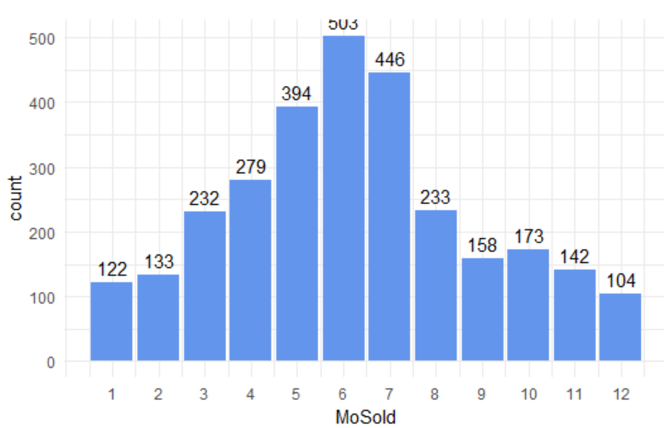
```
      PoolQC  MiscFeature        Alley        Fence
        2909         2814         2721         2348
 GarageYrBlt GarageFinish   GarageQual   GarageCond
         159          159          159          159
  GarageType     BsmtCond BsmtExposure     BsmtQual
         157           82           82           81
BsmtFinType2 BsmtFinType1   MasVnrType   MasVnrArea
          80           79           24           23
    MSZoning    Utilities BsmtFullBath BsmtHalfBath
           4            2            2            2
  Functional  Exterior1st  Exterior2nd   BsmtFinSF1
           2            1            1            1
  BsmtFinSF2    BsmtUnfSF  TotalBsmtSF   Electrical
           1            1            1            1
 KitchenQual   GarageCars   GarageArea     SaleType
           1            1            1            1
[1] "There are 32 columns with missing values"
```

- For PoolQC we replaced the missing values with ‘None’.
- For ‘GarageYrBlt’ we replaced it with the year the house was built.
- 'GarageFinish', 'GarageQual', 'GarageCond', 'GarageType' missing values were replaced with 0. For the house with GarageArea = 360 and GarageCars = 1, but NA's in the other columns, we used the most frequent values for each columns from houses with a similar area and car count.

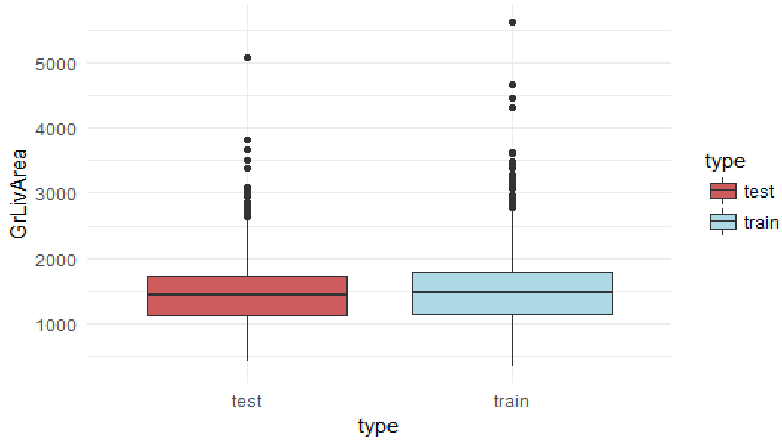



- ‘KitchenQual’ and ‘Electrical’ both have 1 missing values each. We filled in the missing value with the most frequent value from each column.

- 

- Created bins with similar median sale prices from the above graph. We have created four bins, categorizing them according to the median sale price of the houses in the neighborhood.
- ‘BsmtQual’,‘BsmtCond’,‘BsmtExposure’,’BsmtFinType1’,’BsmtFinSF1’,’BsmtFinType2’,’BsmtFinSF2’, ‘BsmtUnfSF’,’TotalBsmtSF’,’BsmtFullBath’,’BsmtHalfBath’.Almost all of the missing values for each basement feature comes from houses with 0 corresponding to area. We filled in these values with 'None'. Rows 949, 1488 and 2349 are the only missing values from BsmtExposure, we can fill this with ‘No’ as that is the most frequent value.
- Similarly for the rest of the predictors, we imputed the missing values based on the corresponding area and the most frequent value. If the area was 0 we replaced it with None. We also made use of the contingency table to find a missing value based on another variable.
- We then converted the categorical features into numeric values. We also created new variables to add more value to the prediction(eg: Age, sold in summer, new houses, Recently Remodeled).
- 

- We observed some suspicious houses with abnormally large GrLivArea's, 2 of which had very low SalePrices




- For the training data we can see 4 houses whose GrLivArea is greater than 4000. These houses in the training set are obnoxiously large and ultimately do not add much value and are causing heavy skewness in both the SalePrice and GrLivArea. Therefore we removed these houses from the training dataset.
- For skewed data we took a log transform and to normalize the data we applied Caret's preProcess function.
- We removed all of the following near-zero-variance variables from our dataframe

```
  [1] "BsmtFinSF2"             "LowQualFinSF"
  [3] "KitchenAbvGr"           "OpenPorchSF"
  [5] "EnclosedPorch"          "X3SsnPorch"
  [7] "ScreenPorch"            "PoolArea"
  [9] "MiscVal"                "BsmtFinType2"
 [11] "Functional"             "LandSlopeGentle"
 [13] "HasShed"                "NewHouse"
 [15] "HasX3SsnPorch"          "MSZoningC..all."
 [17] "MSZoningFV"             "MSZoningRH"
 [19] "StreetGrvl"             "StreetPave"
 [21] "AlleyGrvl"              "AlleyPave"
 [23] "LotShapeIR2"            "LotShapeIR3"
 [25] "LandContourBnk"         "LandContourHLS"
 [27] "LandContourLow"         "LotConfigFR2"
 [29] "LotConfigFR3"           "LandSlopeGtl"
 [31] "LandSlopeMod"           "LandSlopeSev"
 [33] "NeighborhoodBlmngtn"    "NeighborhoodBlueste"
 [35] "NeighborhoodBrDale"     "NeighborhoodBrkSide"
 [37] "NeighborhoodClearCr"    "NeighborhoodCrawfor"
 [39] "NeighborhoodIDOTRR"     "NeighborhoodMeadowV"
 [41] "NeighborhoodMitchel"    "NeighborhoodNoRidge"
 [43] "NeighborhoodNPkVill"    "NeighborhoodNWAmes"
 [45] "NeighborhoodSawyerW"    "NeighborhoodStoneBr"
 [47] "NeighborhoodSWISU"      "NeighborhoodTimber"
 [49] "NeighborhoodVeenker"    "Condition1Artery"
 [51] "Condition1PosA"         "Condition1PosN"
 [53] "Condition1RRAe"         "Condition1RRAn"
 [55] "Condition1RRNe"         "Condition1RRNn"
 [57] "Condition2Artery"       "Condition2Feedr"
 [59] "Condition2Norm"         "Condition2PosA"
 [61] "Condition2PosN"         "Condition2RRAe"
 [63] "Condition2RRAn"         "Condition2RRNn"
 [65] "BldgType2fmCon"         "BldgTypeDuplex"
 [67] "BldgTypeTwnhs"          "HouseStyle1.5Unf"
 [69] "HouseStyle2.5Fin"       "HouseStyle2.5Unf"
 [71] "HouseStyleSFoyer"       "HouseStyleSLvl"
 [73] "RoofStyleFlat"          "RoofStyleGambrel"
 [75] "RoofStyleMansard"       "RoofStyleShed"
 [77] "RoofMatlCompShg"        "RoofMatlMembran"
 [79] "RoofMatlMetal"          "RoofMatlRoll"
 [81] "RoofMatlTar.Grv"        "RoofMatlWdShake"
 [83] "RoofMatlWdShngl"        "Exterior1stAsbShng"
 [85] "Exterior1stAsphShn"     "Exterior1stBrkComm"
 [87] "Exterior1stBrkFace"     "Exterior1stCBlock"
 [89] "Exterior1stCemntBd"     "Exterior1stImStucc"
 [91] "Exterior1stOther"       "Exterior1stStone"
 [93] "Exterior1stStucco"      "Exterior1stWdShing"
 [95] "Exterior2ndAsbShng"     "Exterior2ndAsphShn"
 [97] "Exterior2ndBrk.Cmn"     "Exterior2ndBrkFace"
 [99] "Exterior2ndCBlock"      "Exterior2ndCmentBd"
[101] "Exterior2ndImStucc"     "Exterior2ndOther"
```

```
[103] "Exterior2ndStone"      "Exterior2ndStucco"
[105] "Exterior2ndWd.Shng"    "MasVnrTypeBrkCmn"
[107] "ExterQualEx"           "ExterQualFa"
[109] "ExterCondEx"           "ExterCondFa"
[111] "ExterCondPo"           "FoundationSlab"
[113] "FoundationStone"       "FoundationWood"
[115] "BsmtQualFa"            "BsmtQualNone"
[117] "BsmtCondFa"            "BsmtCondGd"
[119] "BsmtCondNone"          "BsmtCondPo"
[121] "BsmtExposureNone"      "BsmtFinType1None"
[123] "BsmtFinType2ALQ"       "BsmtFinType2BLQ"
[125] "BsmtFinType2GLQ"       "BsmtFinType2LwQ"
[127] "BsmtFinType2None"      "BsmtFinType2Rec"
[129] "HeatingFloor"          "HeatingGasA"
[131] "HeatingGasW"           "HeatingGrav"
[133] "HeatingOthW"           "HeatingWall"
[135] "HeatingQCFa"           "HeatingQCPo"
[137] "ElectricalFuseF"       "ElectricalFuseP"
[139] "ElectricalMix"         "KitchenQualFa"
[141] "FunctionalMaj1"        "FunctionalMaj2"
[143] "FunctionalMin1"        "FunctionalMin2"
[145] "FunctionalMod"         "FunctionalSev"
[147] "FireplaceQuEx"         "FireplaceQuFa"
[149] "FireplaceQuPo"         "GarageType2Types"
[151] "GarageTypeBasment"     "GarageTypeCarPort"
[153] "GarageQualEx"          "GarageQualFa"
[155] "GarageQualGd"          "GarageQualPo"
[157] "GarageCondEx"          "GarageCondFa"
[159] "GarageCondGd"          "GarageCondPo"
[161] "PavedDriveP"           "PoolQCEx"
[163] "PoolQCFa"              "PoolQCGd"
[165] "PoolQCNone"            "FenceGdPrv"
[167] "FenceGdWo"             "FenceMnWw"
[169] "MiscFeatureGar2"       "MiscFeatureNone"
[171] "MiscFeatureOthr"       "MiscFeatureShed"
[173] "MiscFeatureTenC"       "SaleTypeCOD"
[175] "SaleTypeCon"           "SaleTypeConLD"
[177] "SaleTypeConLI"         "SaleTypeConLw"
[179] "SaleTypeCWD"           "SaleTypeOth"
[181] "SaleConditionAdjLand"  "SaleConditionAlloca"
[183] "SaleConditionFamily"
```

- Before applying the modelling technique we applied the findCorrelation function to remove the highly correlated predictors.

```
 [1] "Has2ndFlr"             "HasX2ndFlrSF"
 [3] "PartialPlan"           "TotalArea"
 [5] "AreaInside"            "YearSinceRemodel"
 [7] "LotShapeReg"           "RoofStyleHip"
 [9] "MasVnrTypeNone"        "ExterQualTA"
[11] "HeatingQCEx"           "GarageFinishNone"
[13] "GarageQualNone"        "GarageCondNone"
[15] "FenceNone"             "SaleTypeNew"
[17] "SaleConditionPartial"  "GarageQual"
[19] "WoodDeckSF"            "MasVnrArea"
[21] "HasMasVnr"             "HasWoodDeck"
[23] "HeatingQC"             "YearBuilt"
[25] "YrSold"                "RegularLotShape"
[27] "LandLeveled"           "Exterior1stMetalSd"
[29] "Exterior1stVinylSd"    "BsmtFinSF1"
[31] "CentralAirN"           "ElectricalSB"
[33] "GarageDetchd"          "GarageCond"
[35] "HasPavedDrive"
```

- We also centered and scaled the data while applying the modelling techniques

**4. The final cleaned-up data set that are used in predictive analysis: predictors and responses.**

The final cleaned data has 122 predictors out of which we have 43 numeric variables and 79 dummy variables:

MSSubClass
LotArea
OverallQual
OverallCond
YearRemodAdd
BsmtUnfSF
TotalBsmtSF
X1stFlrSF
X2ndFlrSF
GrLivArea
BsmtFullBath
BsmtHalfBath
FullBath
HalfBath
BedroomAbvGr
TotRmsAbvGrd
Fireplaces
GarageYrBlt
GarageCars
GarageArea
MoSold
ExterQual
ExterCond
KitchenQual
BsmtQual
BsmtExposure
BsmtFinType1
GarageFinish
Fence
NewerDwelling
Remodeled
RecentRemodel
HasMasVnrArea
HasWoodDeckSF
HasOpenPorchSF
HasEnclosedPorch
HasScreenPorch
HighSeason
NbrhRich
NeighborhoodBin
HeatingScale
Age
TimeSinceSold
MSZoningRL
MSZoningRM
AlleyNone
LotShapeIR1
LandContourLvl
LotConfigCorner
LotConfigCulDSac
LotConfigInside
NeighborhoodCollgCr
NeighborhoodEdwards
NeighborhoodGilbert
NeighborhoodNAmes
NeighborhoodNridgHt
NeighborhoodOldTown
NeighborhoodSawyer
NeighborhoodSomerst
Condition1Feedr
Condition1Norm
BldgType1Fam
BldgTypeTwnhsE
HouseStyle1.5Fin
HouseStyle1Story
HouseStyle2Story
RoofStyleGable
Exterior1stHdBoard
Exterior1stPlywood
Exterior1stWd.Sdng
Exterior2ndHdBoard
Exterior2ndMetalSd
Exterior2ndPlywood
Exterior2ndVinylSd
Exterior2ndWd.Sdng
MasVnrTypeBrkFace
MasVnrTypeStone
ExterQualGd
ExterCondGd
ExterCondTA
FoundationBrkTil
FoundationCBlock
FoundationPConc
BsmtQualEx
BsmtQualGd
BsmtQualTA
BsmtCondTA
BsmtExposureAv
BsmtExposureGd
BsmtExposureMn
BsmtExposureNo
BsmtFinType1ALQ
BsmtFinType1BLQ
BsmtFinType1GLQ
BsmtFinType1LwQ
BsmtFinType1Rec
BsmtFinType1Unf
BsmtFinType2Unf
HeatingQCGd
HeatingQCTA
CentralAirY
ElectricalFuseA
ElectricalSBrkr
KitchenQualEx
KitchenQualGd
KitchenQualTA
FunctionalTyp
GarageTypeAttchd
GarageTypeBuiltIn
GarageTypeDetchd
GarageTypeNone
GarageFinishFin
GarageFinishRFn
GarageFinishUnf
GarageQualTA
GarageCondTA
PavedDriveN
PavedDriveY
FenceMnPrv
SaleTypeWD
SaleConditionAbnorml
SaleConditionNormal

The response is the Sale Price of the house in USD

5. **Predictive models that you have used. Describe the tuning procedure if the model has tunable parameters.**

**Linear Models:**

- Multiple Linear Regression with all predictors
- Multiple Linear Regression with filtered predictors and 10 fold CV
- Robust Linear Regression with filtered predictors, 10 fold CV and preprocess using PCA
- Partial Least Squares with filtered predictors, 10 fold CV, tune length of 20
- Principal Component Regression with filtered predictors, 10 fold CV and ncomp 35
- Ridge Regression with filtered predictors, 10 fold CV and lambda 0.014285714
- Elastic Net with filtered predictors, 10 fold CV, lambda(0.10) and fraction(0.60)
- An ensemble of PLS and Elastic Net is created using equal weights as both the models have similar RMSE for train and test predictions.

**Non Linear Models:**

- MARS using 10 fold CV and degree=1:2, .nprune=2:38
- SVM using 10 fold CV and tuneLength=20
- Neural Network with filtered predictors, decay 0.1, size=5
- K-NN with filtered predictors, 10 fold CV and tune length of 10. Optimum k =7

**Tree Models:**

- Bagged tree
- Random Forest with ntree=500
- Boosted tree using gaussian distribution, ntree=100,interaction.depth=7, shrinkage=0.1
- CART using 10 fold CV and tuneLength = 20

6. **Present the performance of the predictive model. Include results from resampling (i.e., 10-fold cross-validation with 5 repeats) and/or a testing data set. For regression problems, report RMSE and R^2. Do not forget about the SDs for these two metrics from resampling procedures. For classification problems, show ROC curves and other metrics that you believe are important for the specific predictive exercise.**

**Linear Models:**

| **Model** | **Train RMSE** | **R^2** | **RMSE SD** | **R^2 SD** | **Test RMSE** |
|---|---|---|---|---|---|
| Multiple LR with all predictors | 0.1051 | Multiple R-squared: 0.9361 Adjusted R-squared: 0.9296 | - | - | 0.12194 |
| Linear regression model with 10 folds CV | 0.1068 | Multiple R-squared: 0.933, Adjusted R-squared: 0.9273 | 0.01434768 | 0.01812518 | 0.12213 |
| Robust linear regression | 0.1209279 | 0.9076801 | 0.01556433 | 0.02063341 | 0.13200 |
| PLS | 0.1132457 | 0.9184889 | 0.01414828 | 0.01785560 | 0.11948 |
| PCR | 0.1214349 | 0.9067512 | 0.01359602 | 0.01822849 | 0.13939 |
| Ridge regression | 0.1129433 | 0.9192039 | 0.01403201 | 0.01789303 | 0.12104 |
| Enet | 0.1170720 | 0.9145279 | 0.01412760 | 0.01714102 | 0.12006 |
| **Ensemble (PLS Tune and Enet)** | - | - | - | - | **0.11849** |

**Non Linear Models:**

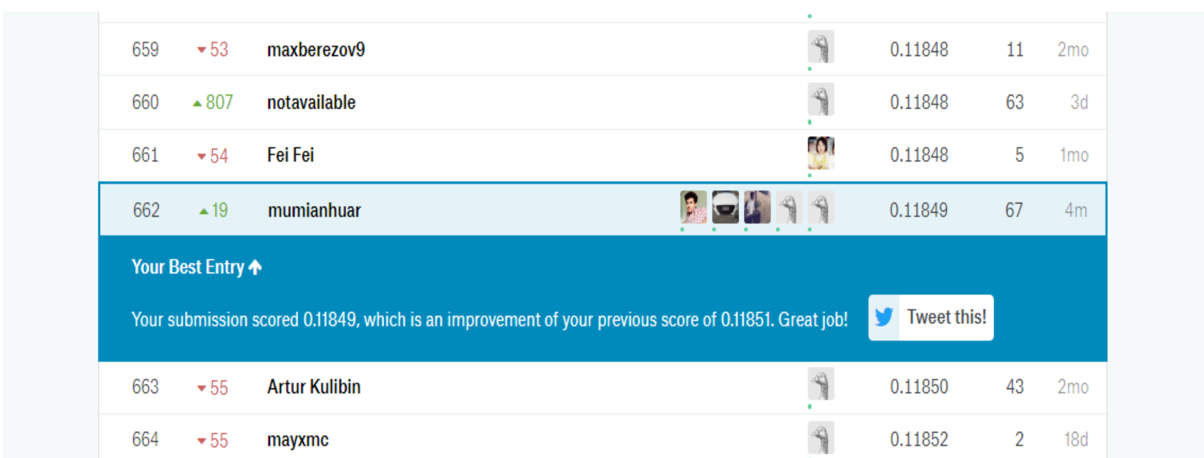
| Model | Train RMSE | R^2 | RMSE SD | R^2 SD | Test RMSE |
|---|---|---|---|---|---|
| MARS | 0.1211108 | 0.9068691 | 0.01679566 | 0.02203292 | 0.12590 |
| SVM | 0.1141004 | 0.9170376 | 0.01986244 | 0.02363894 | **0.12514** |
| Neural Net | 0.2240647 | 0.7217428 | 0.06384209 | 0.11379937 | 0.17189 |
| KNN | 0.1737524 | 0.8143997 | 0.01731130 | 0.03058243 | 0.17713 |

**Tree Models:**

| Model | Test RMSE |
|---|---|
| Bagged tree | 0.18729 |
| Random Forest | 0.13971 |
| Boosted tree | 0.12952 |
| CART | 0.22134 |

Ensemble performs the best among all the models that we performed with an RMSE of **0.11849** for the test set.

**Kaggle Competition Ranking: (Placed in Top 12%)**



7. **Discussion of the predictors that are found to be important and whether these predictors agree with what a human expert would believe as important (if this is possible to discuss).**

   The top predictors that are found to be important are listed below:

```
                     Overall
OverallQual           100.00
GrLivArea              94.75
NeighborhoodBin        82.94
GarageCars             78.70
TotalBsmtSF            78.47
X1stFlrSF              77.11
ExterQual              76.53
GarageArea             76.49
KitchenQual            76.05
Age                    71.49
TotRmsAbvGrd           68.70
FullBath               68.62
GarageYrBlt            68.51
YearRemodAdd           64.75
FoundationPConc        62.95
KitchenQualTA          61.49
Fireplaces             61.20
ExterQualGd            60.60
BsmtQualTA             56.56
GarageFinish           55.72
```

   The top three important predictors for determining the house price for this data set are the overall quality, size of the living area and the neighborhood.

   The predictors that were found to be important by the models agree with what we believed to be important (OverallQual, GrLivArea, Neighborhood). Also there are few additional predictors that the model found to be more important than we thought (TotalBsmtSF, Garage Cars, GarageYrBlt). This has given has more insights about dependence of these additional predictors on the sale price of the house.

8. **Detailed step-by-step instructions on how to run your codes with the data sets to reproduce your results. If your data sets are too large to upload, detailed instructions on where the data sets can be downloaded.**
   - **Step 1 :** To open the 'Damuluri_Molugu_Mote_Reddy_Shen_Zhang_House Price.RMD' file.
   - **Step 2** : Replace the file path of the training and test sets (train and test set attached in the folder given. It can also be downloaded from : https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data)at line no 33&34
   - **Step 3 :** All chunks can be run until line no 1000(Data cleaning). From 1000 onwards is the application of Linear Models
   - **Step 4 :** Each of the chuck can be run to see the respective model results
   - **Step 5** : From line no 1196 are the Tree Models. All the tree models are contained in one chunk
   - **Step 6** : From line no 1268 are the non linear models. Each chuck can be run separately to see the results
   - **Step 7 :** The predictions CSV file needs to be in the same format as the 'sample_submission.csv' in order to be uploaded to Kaggle to see the test results.

**References :**


1. Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project, Dean De Cock, Truman State University, Journal of Statistics Education Volume 19, Number 3(2011), www.amstat.org/publications/jse/v19n3/decock.pdf

2. We took reference for the cleaning part from : Detailed Data Analysis & Ensemble Modeling, TannerCarbonati, April 5 2017, https://www.kaggle.com/tannercarbonati/detailed-data-analysis-ensemble-modeling