Predicting Housing Sales Prices in Ames, Iowa

Balineni Priyanka

Hanus Andrew

Waingankar Varsha

Presentation Overview

1. Problem Statement
2. Dataset Overview
3. Strategy Explained
4. Analysis and Findings
5. Best Model Identified
6. Conclusion

Problem Statement

- Can housing sales prices be accurately predicted using disparate but relevant data?
  - Can we identify a prediction model that would be most helpful in addressing this problem?
  - Can we achieve a Sales Price prediction value that is >85% accurate?

Business Case

- The 2006 sudden and immense downturn in U.S. House Prices sparked the 2007 global financial crisis and revived the interest about forecasting such imminent threats for economic stability
- Housing prices adjustments play an important role in the determination of the phase of the business cycle.
- When the economy booms, construction and employment in the housing sector expand rapidly to respond to excess demand, rapidly pushing nominal house prices upwards. During the contraction phase, the drop in private income reduces aggregate demand and nominal house prices

Purpose & Goal

- Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an Airport. But this dataset proves that many more parameters influences price negotiations than the number of bedrooms or a white-picket fence.

- With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, the goal is to predict the final price of each home with a reasonable level of accuracy.

Dataset Overview

1. Column Definitions
2. Dataset Values
3. Dataset as data.frame in R

Column Definitions

- **MSSubClass**: Identifies the type of dwelling involved in the sale.
- **MSZoning**: Identifies the general zoning classification of the sale.
- **LotFrontage**: Linear feet of street connected to property
- **LotArea**: Lot size in square feet
- **Street**: Type of road access to property
- **Alley**: Type of alley access to property
- **LotShape**: General shape of property
- **LandContour**: Flatness of the property
- **Utilities**: Type of utilities available
- **LotConfig**: Lot configuration

Column Definitions *(cont)*

- **LandSlope**: Slope of property
- **Neighborhood**: Physical locations within Ames city limits
- **Condition1**: Proximity to various conditions
- **BldgType**: Type of dwelling
- **HouseStyle**: Style of dwelling
- **OverallQual**: Rates the overall material and finish of the house
- **OverallCond**: Rates the overall condition of the house
- **Heating**: Type of heating

Column Definitions *(cont)*

- **TotalBsmtSF**: Total square feet of basement area
- **1stFlrSF**: First Floor square feet
- **2ndFlrSF**: Second floor square feet
- **GarageArea**: Size of garage in square feet
- **WoodDeckSF**: Wood deck area in square feet
- **YearBuilt**: Original construction date
- **YearRemodAdd**: Remodel date (same as construction date if no remodeling or additions)

Dataset Values

OpenPorchSI	EnclosedPor	3SsnPorch	ScreenPorch	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleConditio	SalePrice
61	0	0	0	0	NA	NA	NA	0	2	2008	WD	Normal	208500
0	0	0	0	0	NA	NA	NA	0	5	2007	WD	Normal	181500
42	0	0	0	0	NA	NA	NA	0	9	2008	WD	Normal	223500
35	272	0	0	0	NA	NA	NA	0	2	2006	WD	Abnorml	140000
84	0	0	0	0	NA	NA	NA	0	12	2008	WD	Normal	250000
30	0	320	0	0	NA	MnPrv	Shed	700	10	2009	WD	Normal	143000
57	0	0	0	0	NA	NA	NA	0	8	2007	WD	Normal	307000
204	228	0	0	0	NA	NA	Shed	350	11	2009	WD	Normal	200000

Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood	Condition1	Condition2	BldgType	HouseStyle	OverallQual	OverallCond	YearBuilt	YearRemodA	RoofStyle
1	60	RL	65	8450	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1Fam	2Story	7	5	2003	2003	Gable
2	20	RL	80	9600	Pave	NA	Reg	Lvl	AllPub	FR2	Gtl	Veenker	Feedr	Norm	1Fam	1Story	6	8	1976	1976	Gable
3	60	RL	68	11250	Pave	NA	IR1	Lvl	AllPub	Inside	Gtl	CollgCr	Norm	Norm	1Fam	2Story	7	5	2001	2002	Gable
4	70	RL	60	9550	Pave	NA	IR1	Lvl	AllPub	Corner	Gtl	Crawfor	Norm	Norm	1Fam	2Story	7	5	1915	1970	Gable
5	60	RL	84	14260	Pave	NA	IR1	Lvl	AllPub	FR2	Gtl	NoRidge	Norm	Norm	1Fam	2Story	8	5	2000	2000	Gable
6	50	RL	85	14115	Pave	NA	IR1	Lvl	AllPub	Inside	Gtl	Mitchel	Norm	Norm	1Fam	1.5Fin	5	5	1993	1995	Gable
7	20	RL	75	10084	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	Somerst	Norm	Norm	1Fam	1Story	8	5	2004	2005	Gable
8	60	RL	NA	10382	Pave	NA	IR1	Lvl	AllPub	Corner	Gtl	NWAmes	PosN	Norm	1Fam	2Story	7	6	1973	1973	Gable
9	50	RM	51	6120	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	OldTown	Artery	Norm	1Fam	1.5Fin	7	5	1931	1950	Gable
10	190	RL	50	7420	Pave	NA	Reg	Lvl	AllPub	Corner	Gtl	BrkSide	Artery	Artery	2fmCon	1.5Unf	5	6	1939	1950	Gable
11	20	RL	70	11200	Pave	NA	Reg	Lvl	AllPub	Inside	Gtl	Sawyer	Norm	Norm	1Fam	1Story	5	5	1965	1965	Hip

RoofMatl	Exterior1st	Exterior2nd	MasVnrType	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtQual	BsmtCond	BsmtExposur	BsmtFinType	BsmtFinSF1	BsmtFinType	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	Heating	HeatingQC	CentralAir	Electrical	FFlrSF
CompShg	VinylSd	VinylSd	BrkFace	196	Gd	TA	PConc	Gd	TA	No	GLQ	706	Unf	0	150	856	GasA	Ex	Y	SBrkr	856
CompShg	MetalSd	MetalSd	None	0	TA	TA	CBlock	Gd	TA	Gd	ALQ	978	Unf	0	284	1262	GasA	Ex	Y	SBrkr	1262
CompShg	VinylSd	VinylSd	BrkFace	162	Gd	TA	PConc	Gd	TA	Mn	GLQ	486	Unf	0	434	920	GasA	Ex	Y	SBrkr	920
CompShg	Wd Sdng	Wd Shng	None	0	TA	TA	BrkTil	TA	Gd	No	ALQ	216	Unf	0	540	756	GasA	Gd	Y	SBrkr	961
CompShg	VinylSd	VinylSd	BrkFace	350	Gd	TA	PConc	Gd	TA	Av	GLQ	655	Unf	0	490	1145	GasA	Ex	Y	SBrkr	1145
CompShg	VinylSd	VinylSd	None	0	TA	TA	Wood	Gd	TA	No	GLQ	732	Unf	0	64	796	GasA	Ex	Y	SBrkr	796
CompShg	VinylSd	VinylSd	Stone	186	Gd	TA	PConc	Ex	TA	Av	GLQ	1369	Unf	0	317	1686	GasA	Ex	Y	SBrkr	1694
CompShg	HdBoard	HdBoard	Stone	240	TA	TA	CBlock	Gd	TA	Mn	ALQ	859	BLQ	32	216	1107	GasA	Ex	Y	SBrkr	1107
CompShg	BrkFace	Wd Shng	None	0	TA	TA	BrkTil	TA	TA	No	Unf	0	Unf	0	952	952	GasA	Gd	Y	FuseF	1022
CompShg	MetalSd	MetalSd	None	0	TA	TA	BrkTil	TA	TA	No	GLQ	851	Unf	0	140	991	GasA	Ex	Y	SBrkr	1077

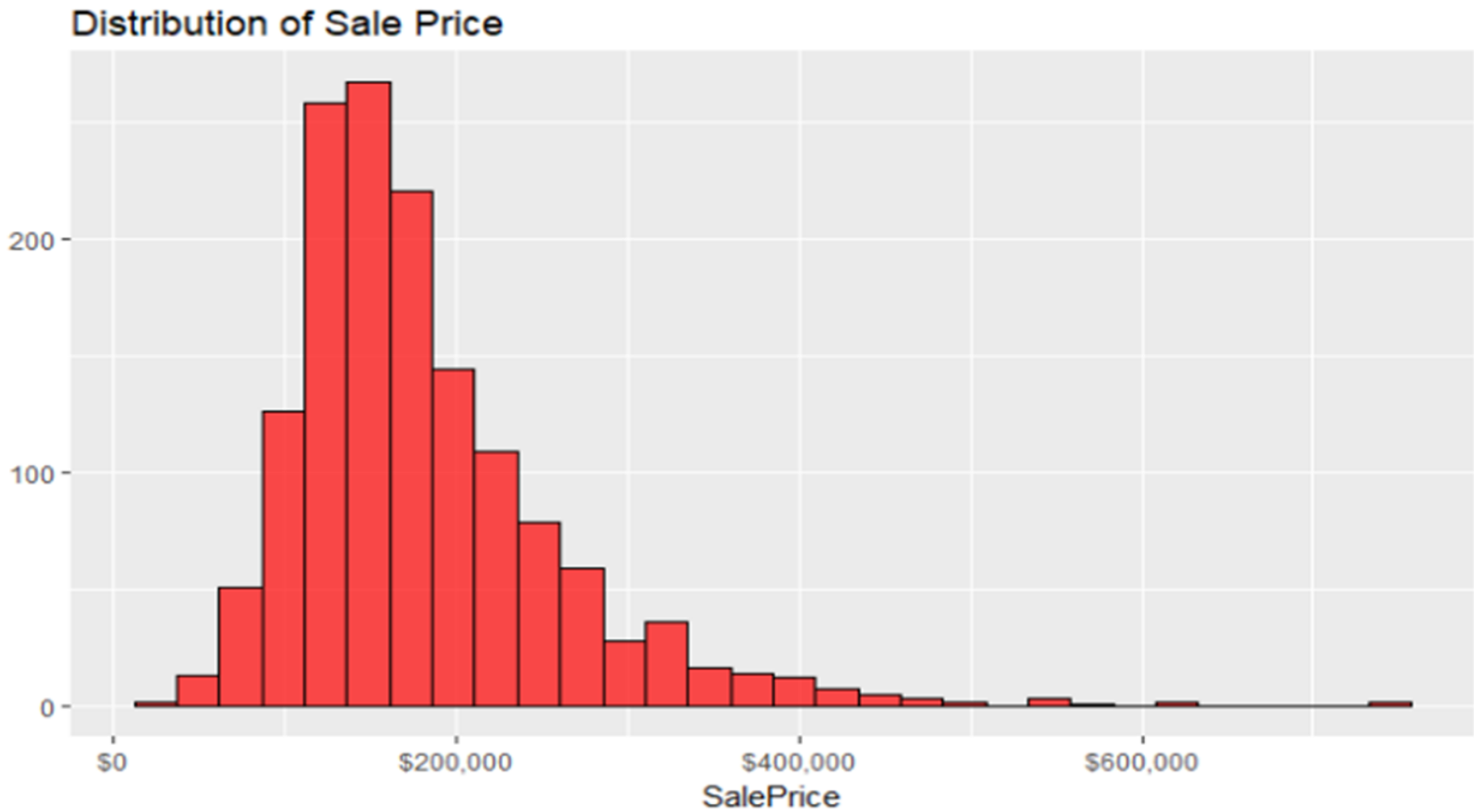
SFlrSF	LowQualFinS	GrLivArea	BsmtFullBat	BsmtHalfBat	FullBath	HalfBath	BedroomAbv	KitchenAbvG	KitchenQual	TotRmsAbvG	Functional	Fireplaces	FireplaceQu	GarageType	GarageYrBlt	GarageFinish	GarageCars	GarageArea	GarageQual	GarageCond	PavedDrive	WoodDeckSF	1040
854	0	1710	1	0	2	1	3	1	Gd	8	Typ	0	NA	Attchd	2003	RFn	2	548	TA	TA	Y	0	1182
0	0	1262	0	1	2	0	3	1	TA	6	Typ	1	TA	Attchd	1976	RFn	2	460	TA	TA	Y	298	912
866	0	1786	1	0	2	1	3	1	Gd	6	Typ	1	TA	Attchd	2001	RFn	2	608	TA	TA	Y	0	1494
756	0	1717	1	0	1	0	3	1	Gd	7	Typ	1	Gd	Detchd	1998	Unf	3	642	TA	TA	Y	0	1253
1053	0	2198	1	0	2	1	4	1	Gd	9	Typ	1	TA	Attchd	2000	RFn	3	836	TA	TA	Y	192	854
566	0	1362	1	0	1	1	1	1	TA	5	Typ	0	NA	Attchd	1993	Unf	2	480	TA	TA	Y	40	
0	0	1694	1	0	2	0	3	1	Gd	7	Typ	1	Gd	Attchd	2004	RFn	2	636	TA	TA	Y	255	
983	0	2090	1	0	2	1	3	1	TA	7	Typ	2	TA	Attchd	1973	RFn	2	484	TA	TA	Y	235	
752	0	1774	0	0	2	0	2	2	TA	8	Min1	2	TA	Detchd	1931	Unf	2	468	Fa	TA	Y	90	
0	0	1077	1	0	1	0	2	2	TA	5	Typ	2	TA	Attchd	1939	RFn	1	205	Gd	TA	Y	0	
0	0	1040	1	0	1	0	3	1	TA	5	Typ	0	NA	Detchd	1965	Unf	1	384	TA	TA	Y	0	
1142	0	2324	1	0	3	0	4	1	Ex	11	Typ	2	Gd	BuiltIn	2005	Fin	3	736	TA	TA	Y	147	
0	0	912	1	0	1	0	2	1	TA	4	Typ	0	NA	Detchd	1962	Unf	1	352	TA	TA	Y	140	
0	0	1494	0	0	2	0	3	1	Gd	7	Typ	1	Gd	Attchd	2006	RFn	3	840	TA	TA	Y	160	
0	0	1253	1	0	1	1	2	1	TA	5	Typ	1	Fa	Attchd	1960	RFn	1	352	TA	TA	Y	0	
0	0	854	0	0	1	0	2	1	TA	5	Typ	0	NA	Detchd	1991	Unf	2	576	TA	TA	Y	48	

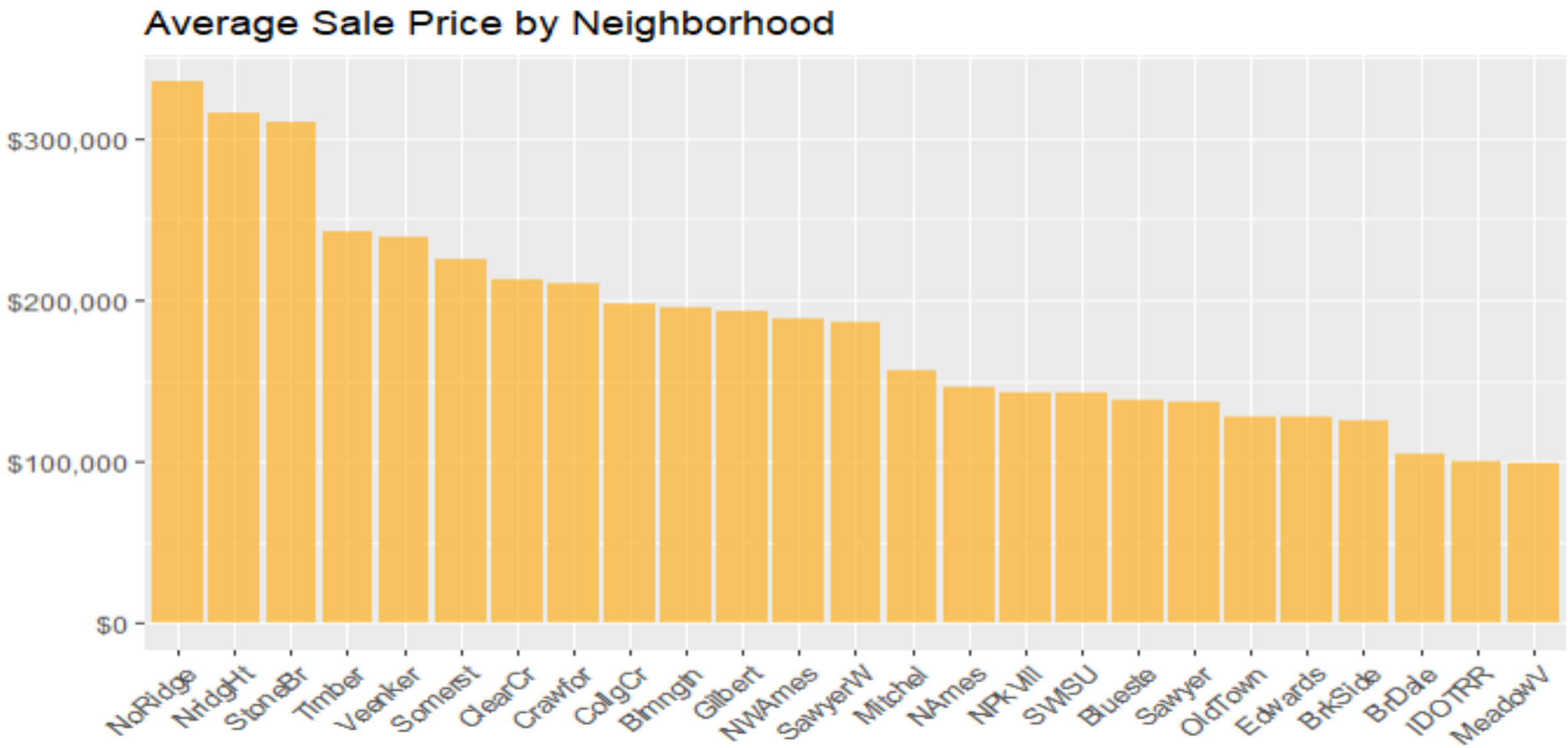
R Breakdown of Dataset as data.frame

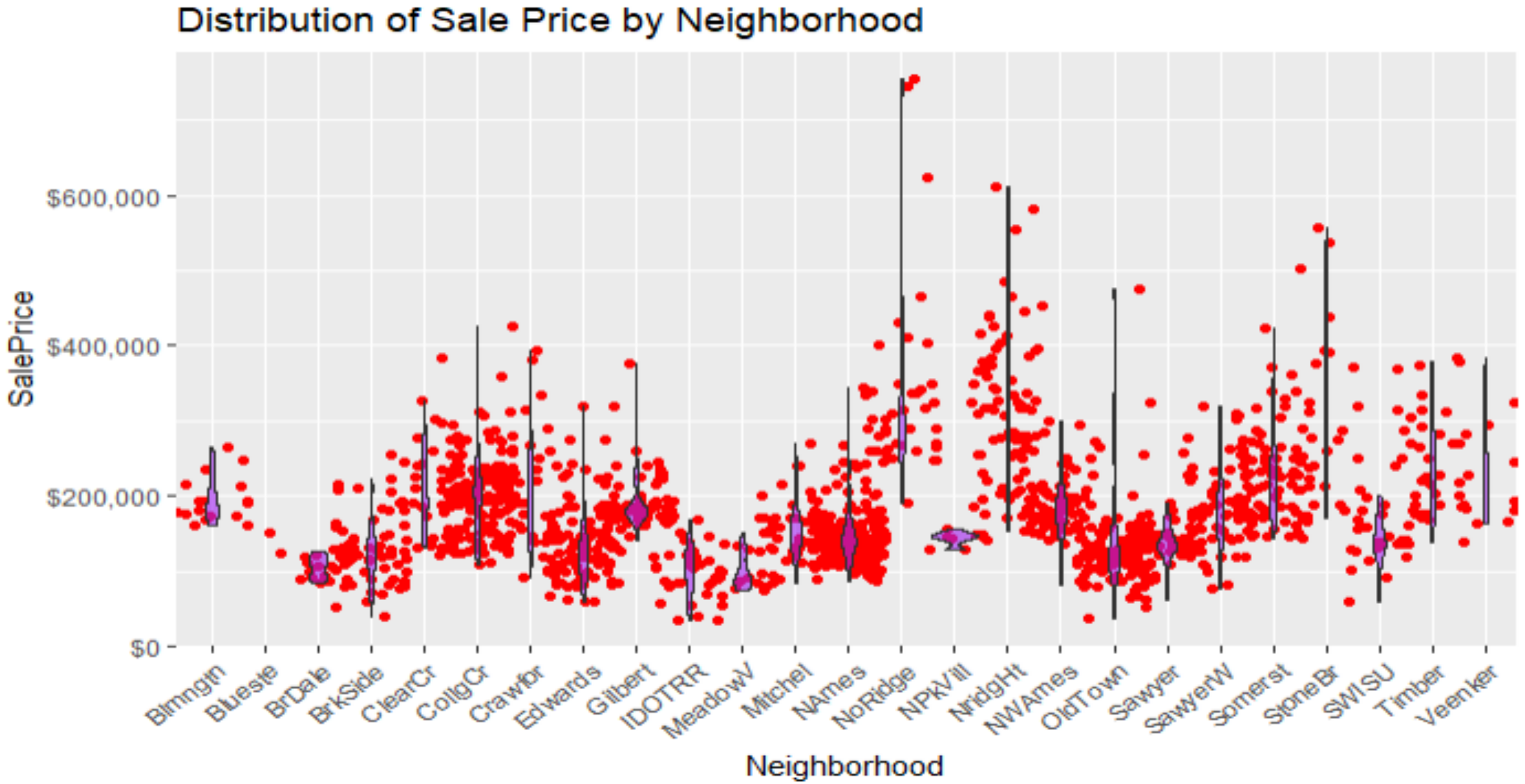
```
> str(House)
data.frame':    1459 obs. of  81 variables:
$ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
$ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
$ MSZoning     : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 4 4 4 4 5 4 ...
$ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
$ 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 6 2 6 6 ...
```

Visualizations of Dataset

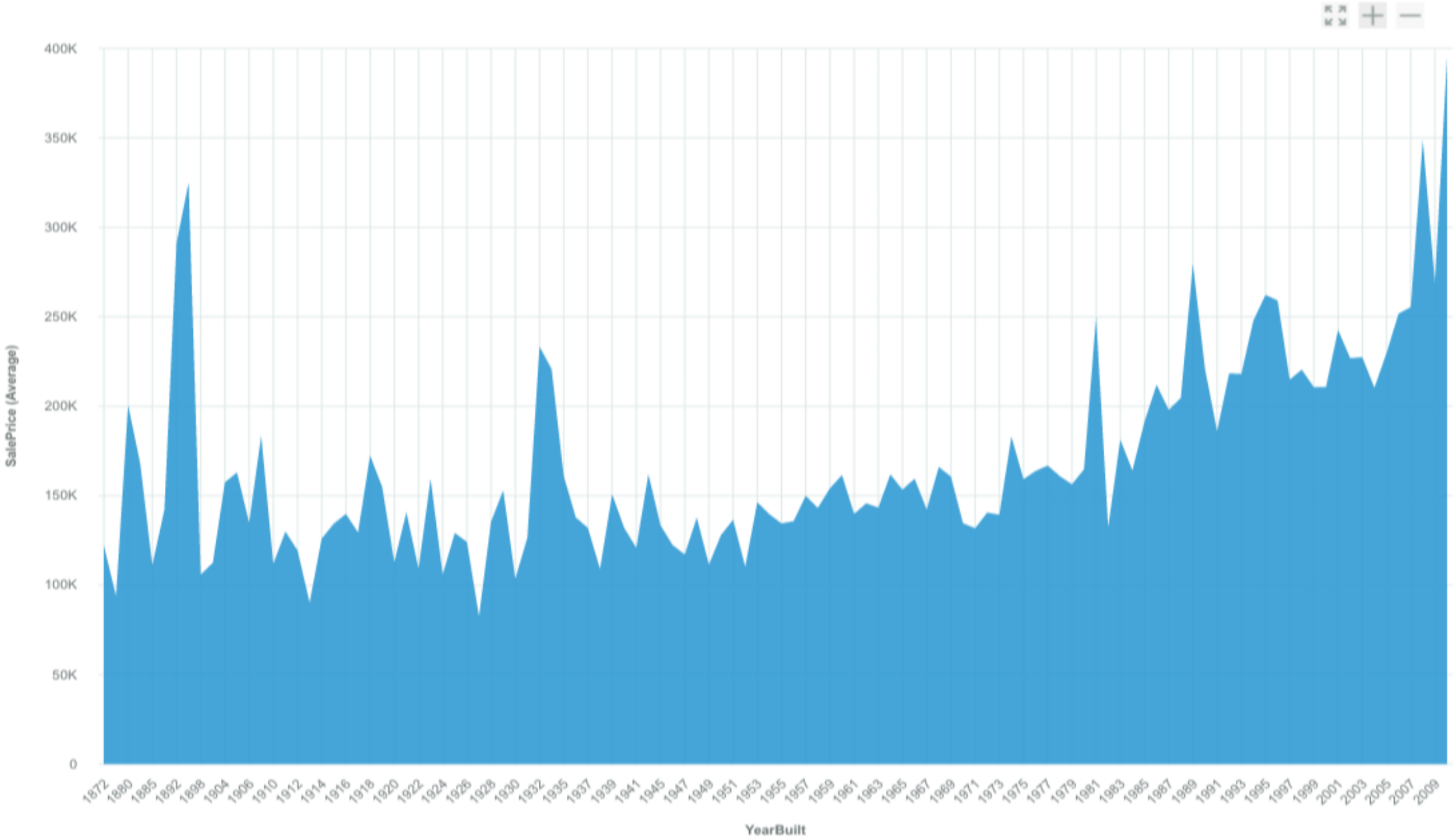
- Distribution of key variable.
- Average values of key variable based on key correlations.
- Mean values of key variable
- Highlighting some key trends associated with key variable
- Decision Tree
- Visualization of Strongest Single factor correlations
- Visualization of Strongest Multivariate correlations

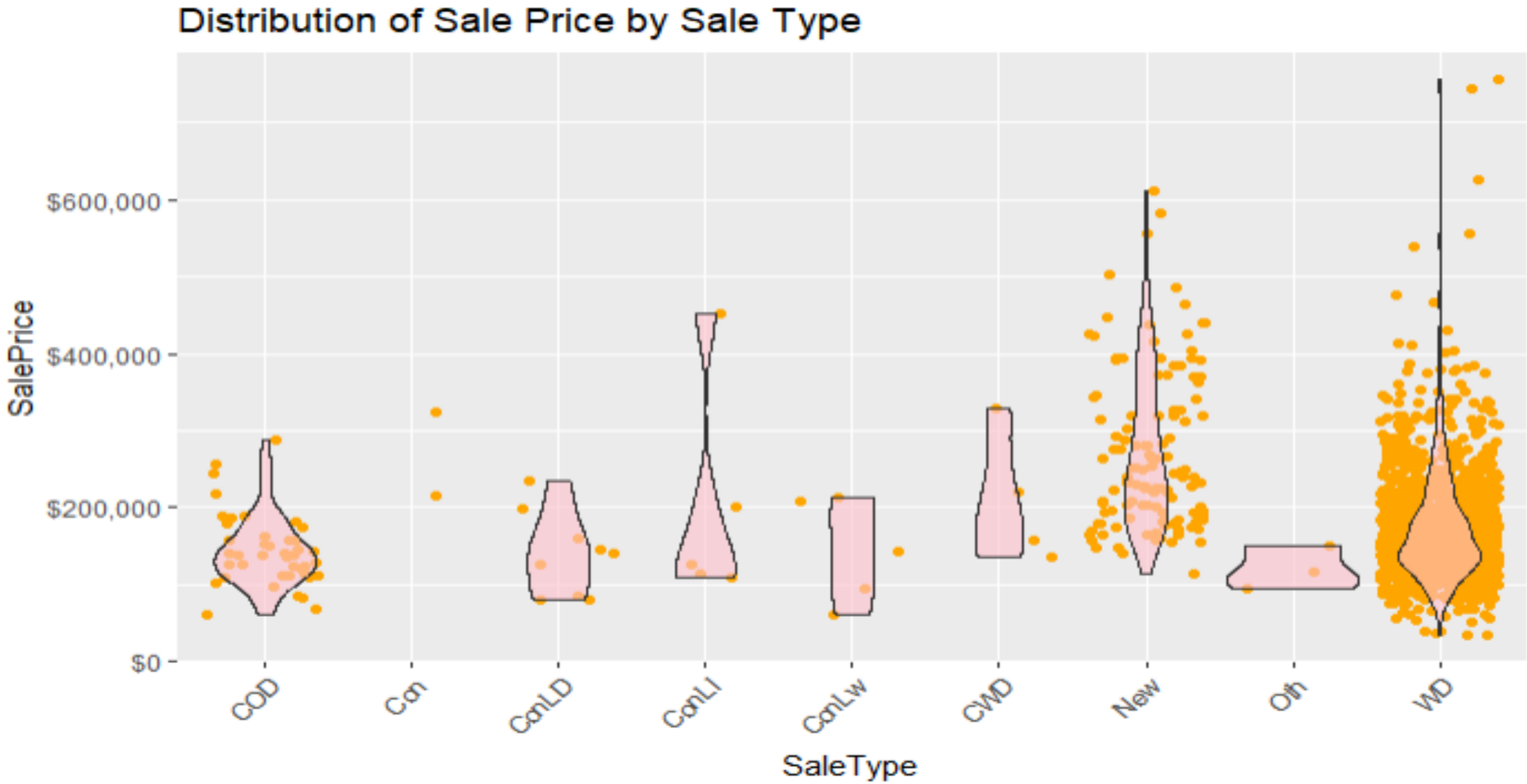

Distribution of Sale Price
200
100
0
$0
$200,000
$400,000
$600,000
SalePrice

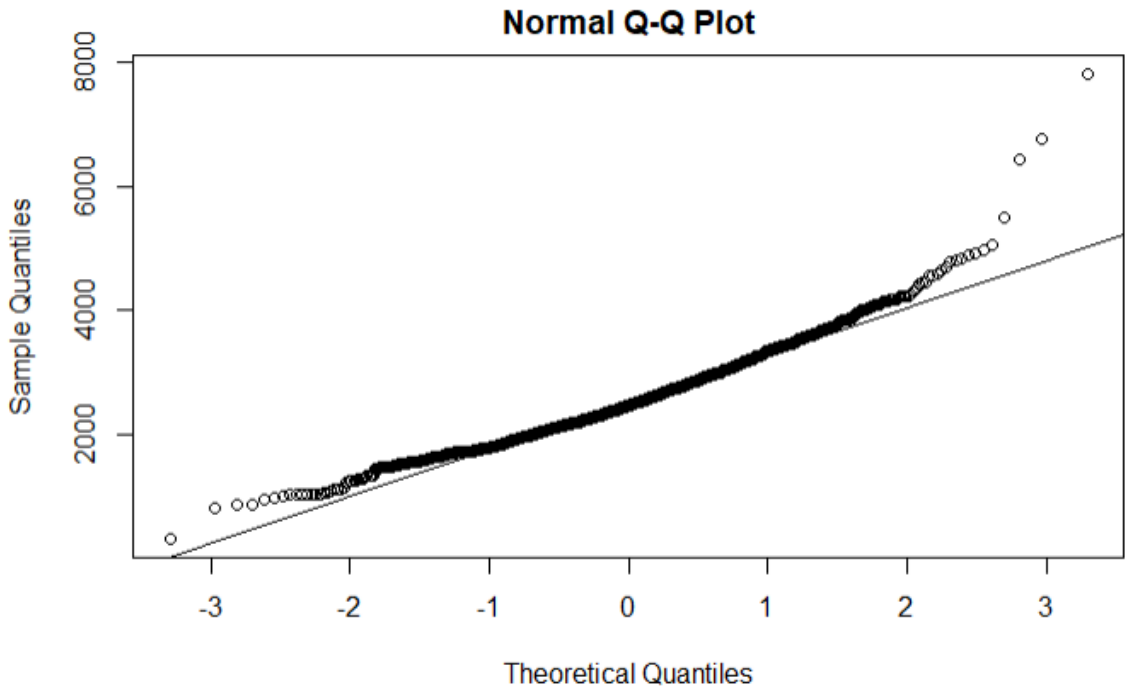

Average Sale Price by Neighborhood
$300,000
$200,000
$100,000
$0
NoRidge
NridgHt
StoneBr
Timber
Veenker
Somerst
ClearCr
Crawfor
CollgCr
Blmngtn
Gilbert
NWAmes
SawyerW
Mitchel
NAmes
NPkVill
SWISU
Blueste
Sawyer
OldTown
Edwards
BrkSide
BrDale
IDOTRR
MeadowV

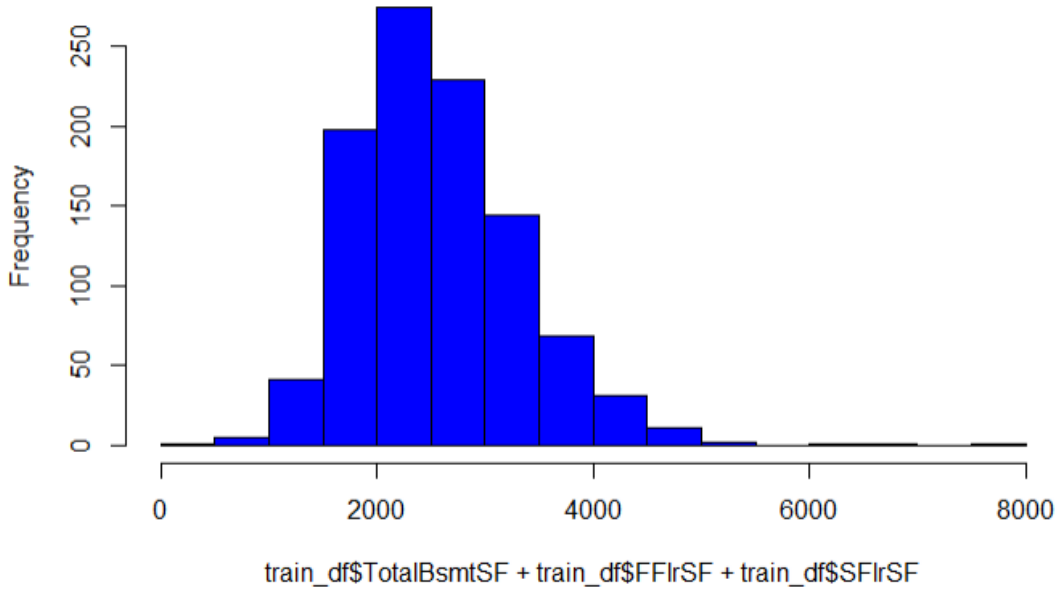

Distribution of Sale Price by Neighborhood
SalePrice
$600,000
$400,000
$200,000
$0
Blmngtn
Blueste
BrDale
BrkSide
ClearCr
CollgCr
Crawfor
Edwards
Gilbert
IDOTRR
MeadowV
Mitchel
NAmes
NoRidge
NPkVill
NridgHt
NWAmes
OldTown
Sawyer
SawyerW
Somerst
StoneBr
SWISU
Timber
Veenker
Neighborhood

What is the contribution of SalePrice ⊗ over YearBuilt ⊗ ?

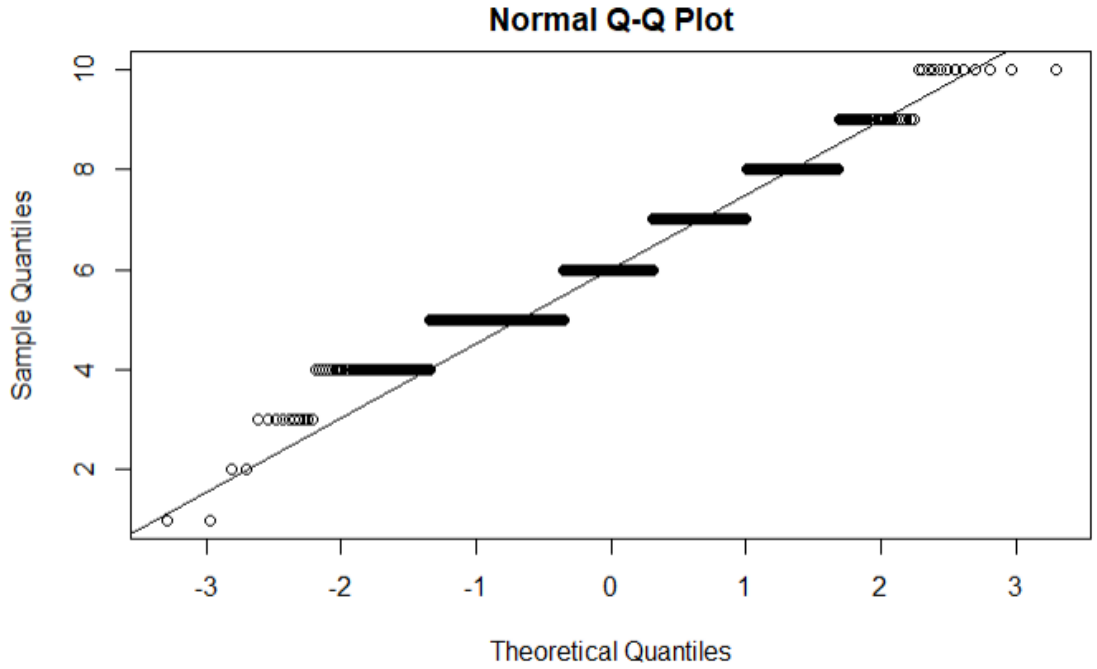


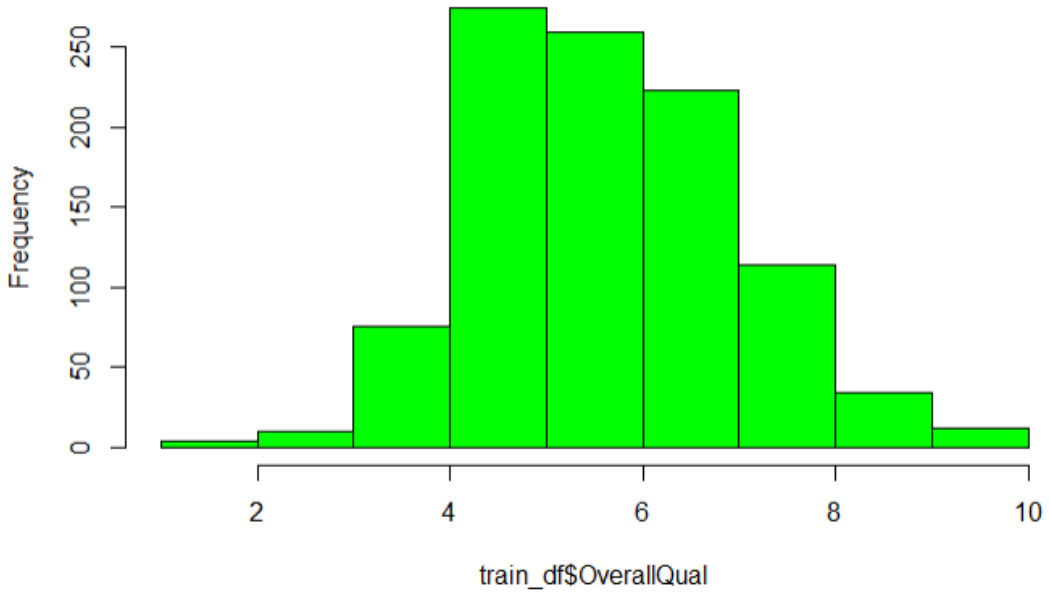

Distribution of Sale Price by Sale Type
SalePrice
$600,000
$400,000
$200,000
$0
COD
Con
ConLD
ConLI
ConLw
CWD
New
Oth
WD
SaleType





Histogram of Square Footage

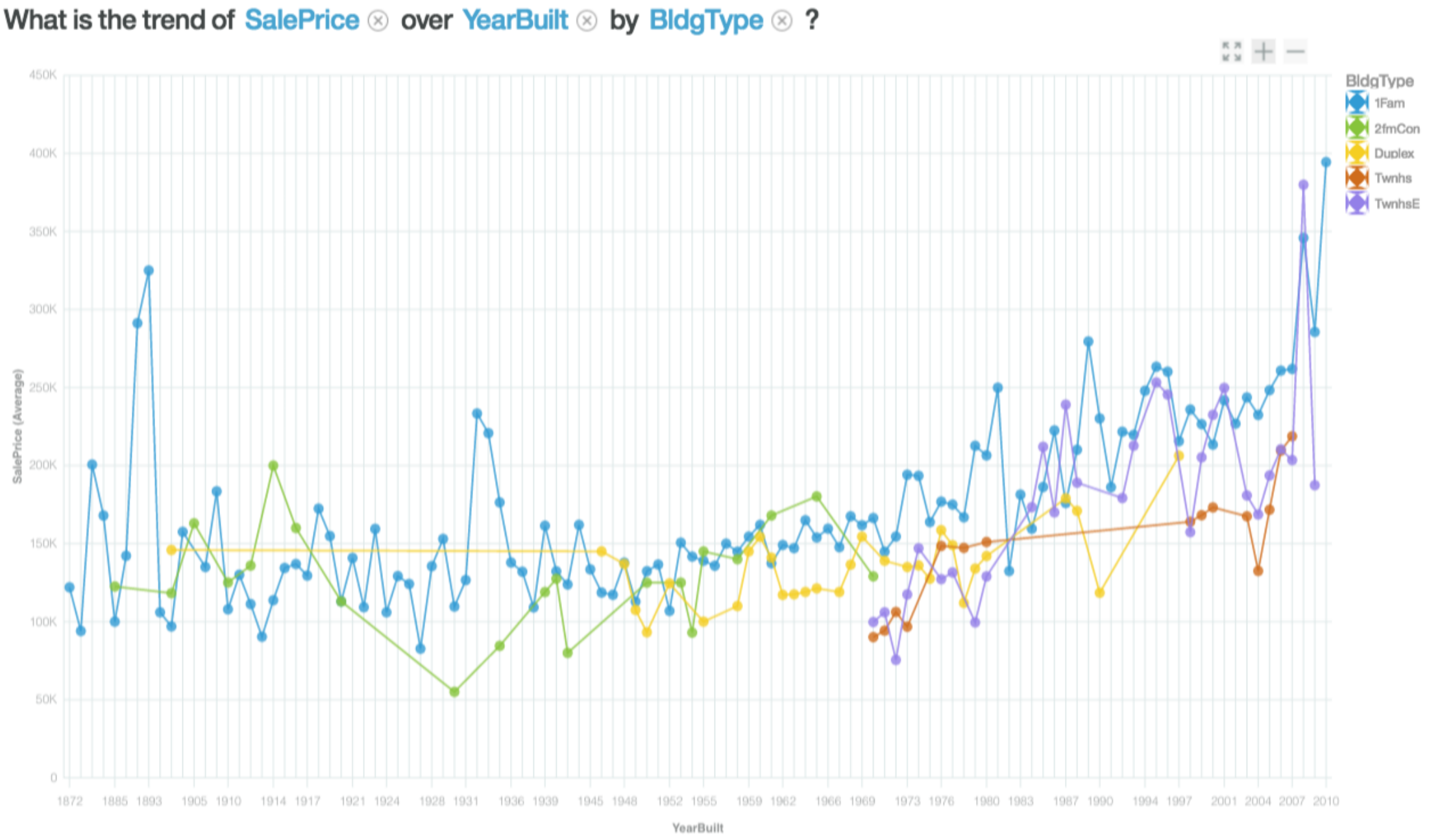




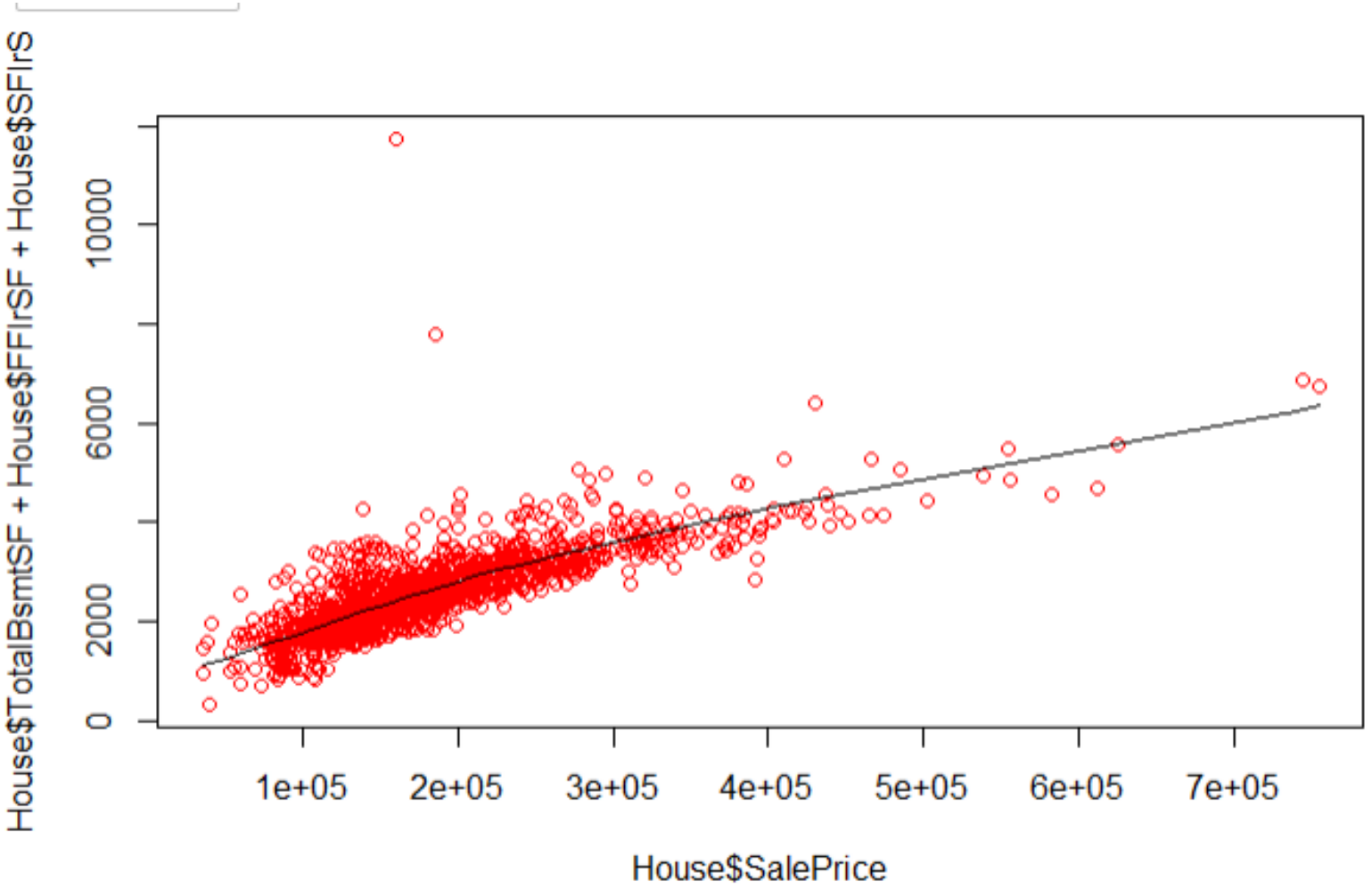
Histogram of Overall Quality


Mean Sale Price
125,000
100,000
75,000
50,000
25,000
2006
2007
2008
2009
2010

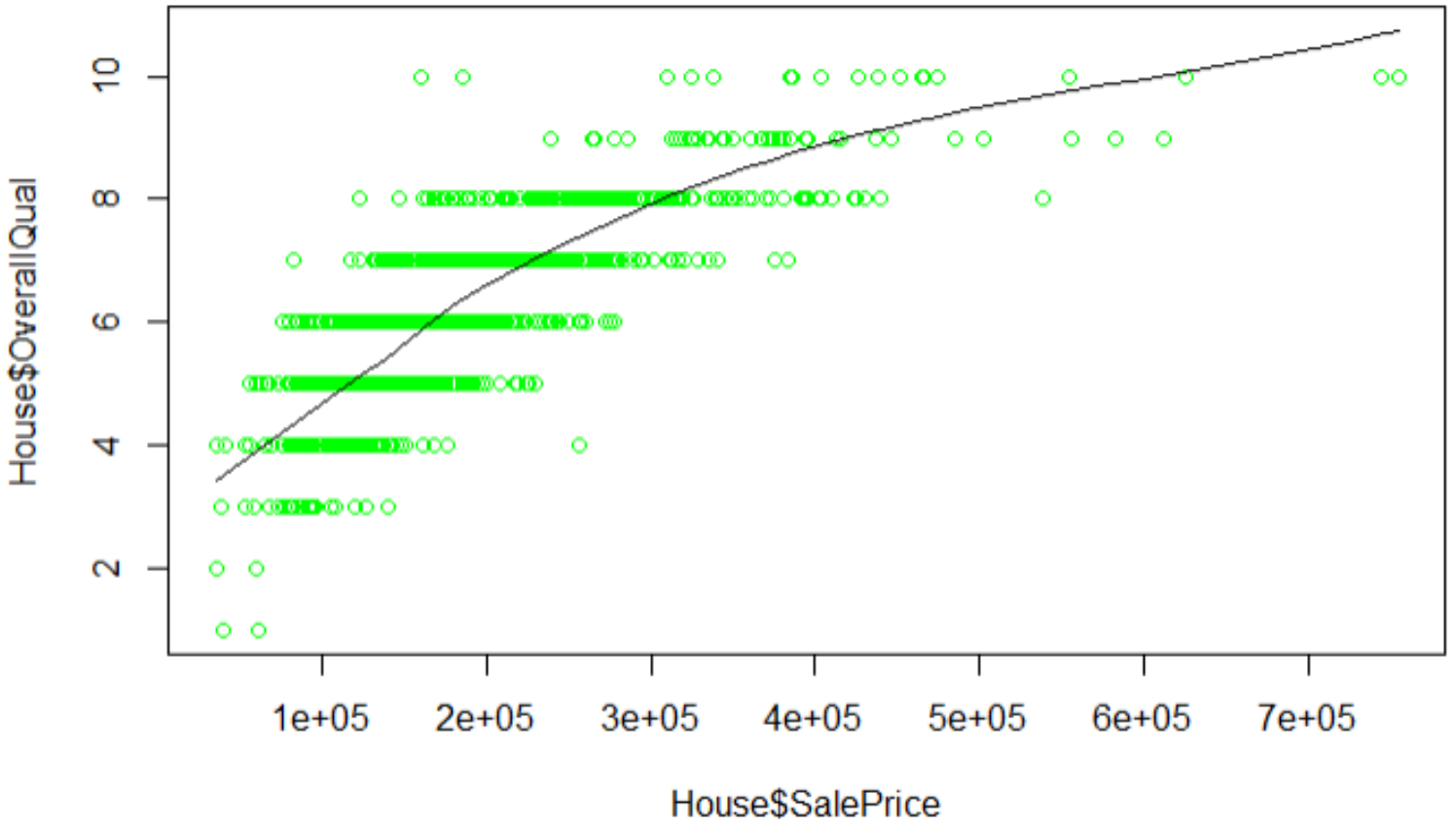

Sales Volume
volume
60
40
20
0
2006
2007
2008
2009
2010


What is the trend of SalePrice over YearBuilt by BldgType ?
BldgType
1Fam
2fmCon
Duplex
Twnhs
TwnhsE
SalePrice (Average)
450K
400K
350K
300K
250K
200K
150K
100K
50K
0
1872 1885 1893 1905 1910 1914 1917 1921 1924 1928 1931 1936 1939 1945 1948 1952 1955 1959 1962 1966 1969 1973 1976 1980 1983 1987 1990 1994 1997 2001 2004 2007 2010
YearBuilt

Correlation between Square footage and sale price 0.7822

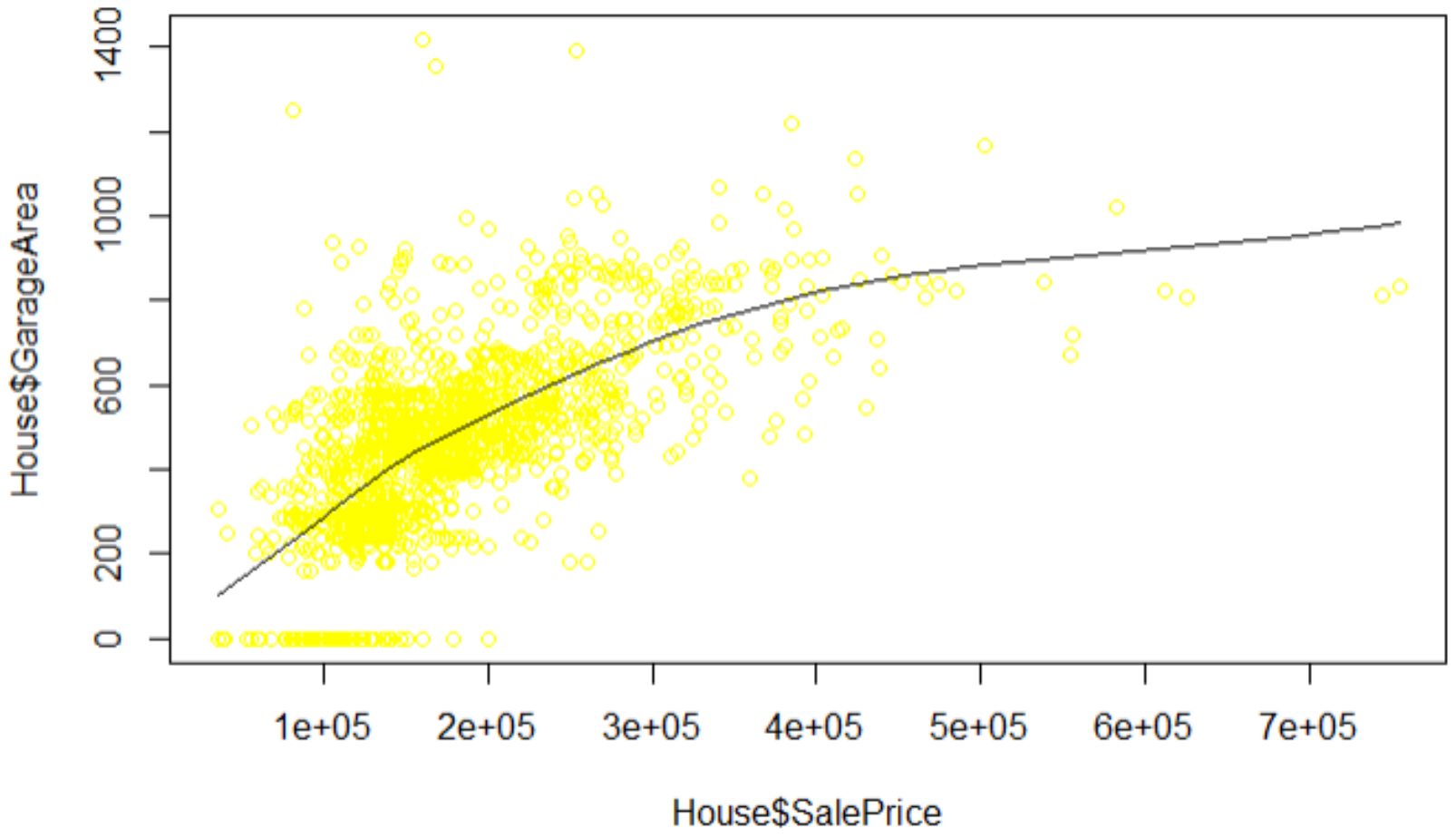


Correlation between sale price and overall quality 0.79



Correlation between sale price and year built 0.52287

Correlation between sale price and garage Area is 0.623



SalePrice

- Zscore - 1.959964
- Confidence Interval - 97.5%
- # of observations - 1459
- Standard Error - 9284.636
- Lower Limit - 171659.50
- Upper Limit - 190228.70

First Floor Sq Ft

- Zscore - 1.959964
- Confidence Interval - 97.5%
- # of observations - 1459
- Standard Error – 19.84306
- Lower Limit - 1142.72
- Upper Limit - 1182.406

Second Floor

- Zscore - 1.959964
- Confidence Interval - 97.5%
- # of observations - 1459
- Standard Error - 22.40205
- Lower Limit - 324.8282
- Upper Limit - 369.6323

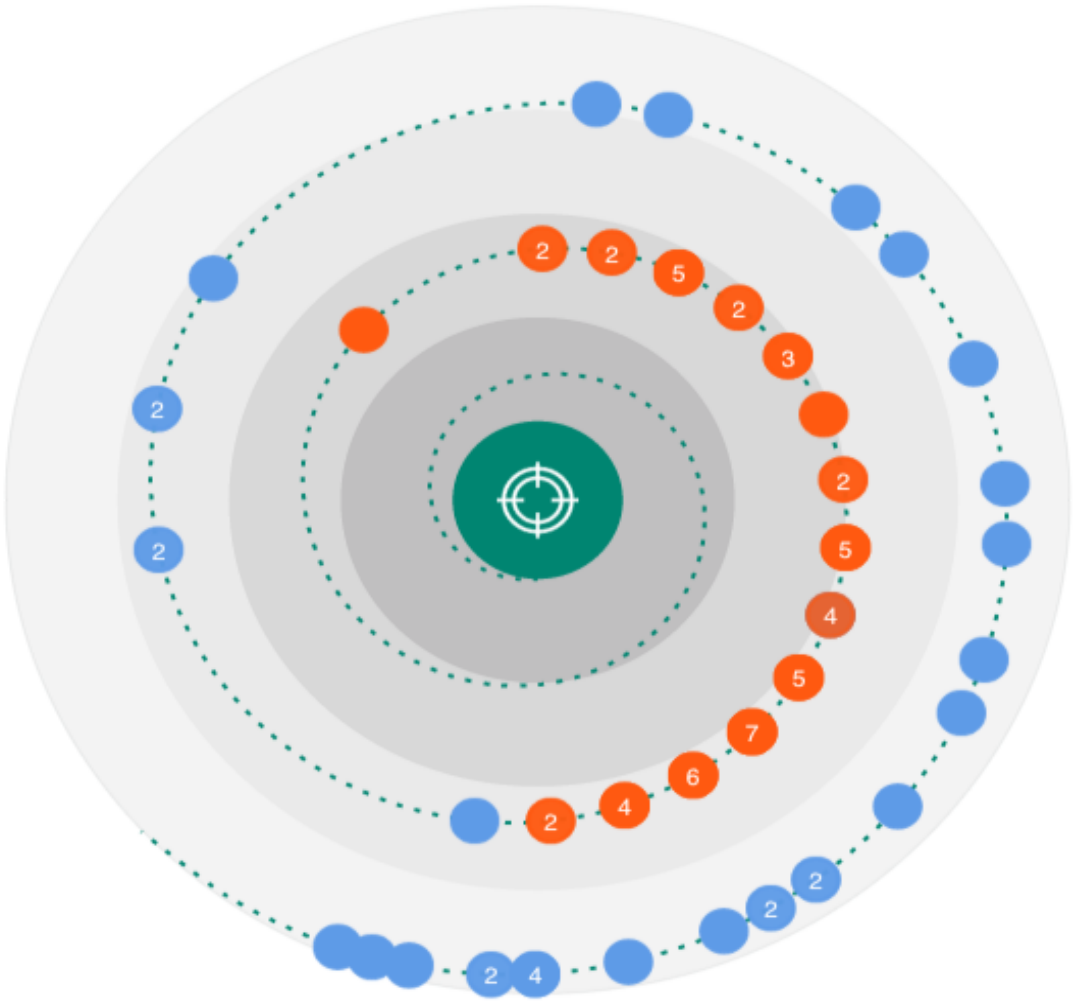
Basement Sq Footage

- Zscore - 1.9599
- Confidence Interval - 97.5%
- # of observations - 1459
- Standard Error - 22. 51706
- Lower Limit - 1034.776
- Upper Limit - 1079.81

Statistical Analysis

- Mean sale price training set is 194597.8
- Mean sale price test set is 177510.6

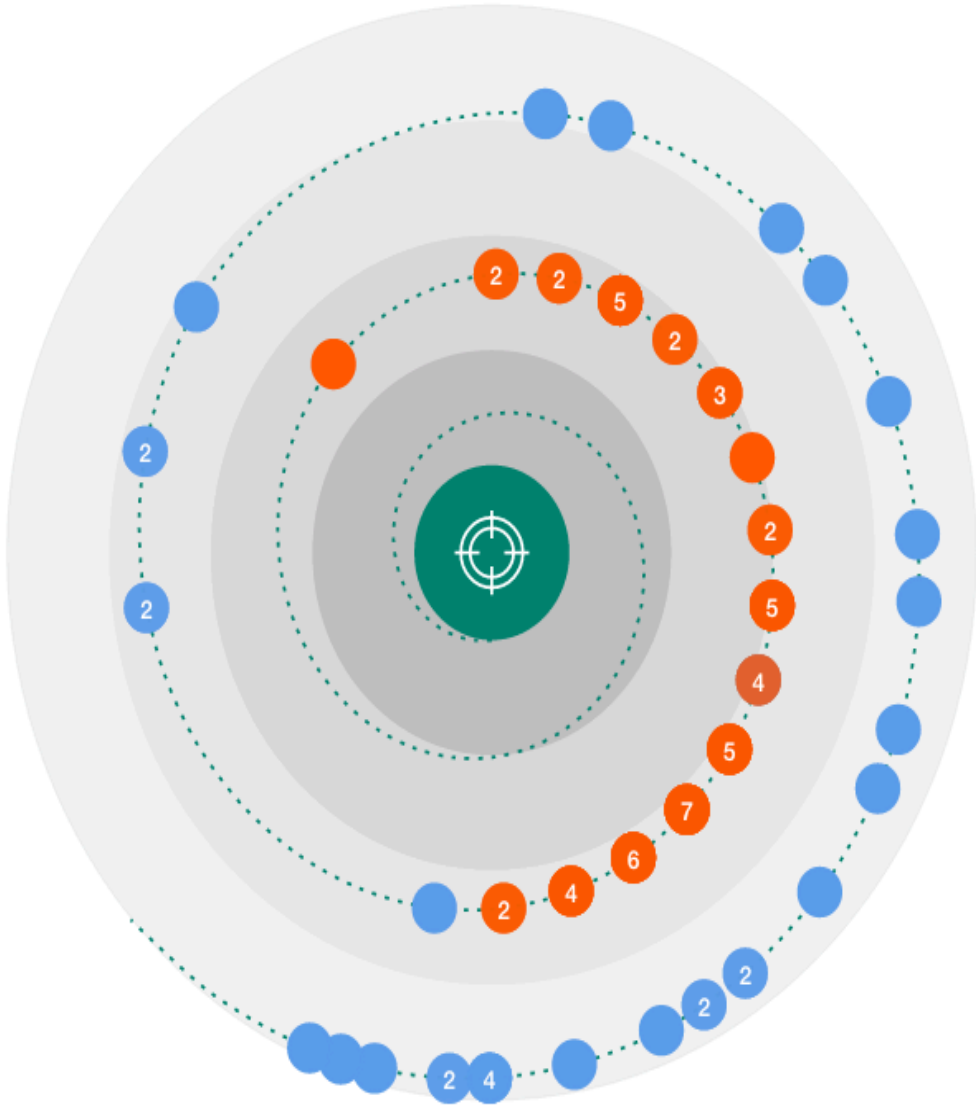
What drives SalePrice ⊗ ?



1 Driver
2 Drivers

Strength	Search drivers
71%	GrLivArea and Neighborhood
68%	GrLivArea and BsmtQual
68%	TotRmsAbvGrd and Neighborhood
67%	GarageArea and OverallQual
67%	GarageCars and Neighborhood
66%	LotArea and Neighborhood
66%	FFlrSF and Neighborhood
66%	GrLivArea and KitchenQual
66%	TotalBsmtSF and Neighborhood
66%	GarageArea and Neighborhood
65%	FullBath and Neighborhood
65%	GrLivArea and GarageCars
64%	KitchenQual and GarageCars
64%	TotalBsmtSF and GrLivArea
64%	Fireplaces and Neighborhood
63%	MSSubClass and Neighborhood

What drives SalePrice ?



1 Driver
2 Drivers

Strength	Search drivers
54%	Neighborhood
48%	GarageCars
48%	ExterQual
46%	BsmtQual
46%	KitchenQual
44%	GrLivArea
38%	GarageArea
37%	TotalBsmtSF
34%	FullBath
33%	FFlrSF
31%	GarageFinish
29%	FireplaceQu
28%	TotRmsAbvGrd
26%	Foundation
25%	GarageType
23%	Fireplaces

Decision Tree – Showing Predictive model for SalePrice (1/3)

Decision tree

OverallQual and 8 other inputs predict **SalePrice**.

Statistical details

SalePrice is a continuous target, so a CHAID regression tree is used.

Predictive strength (1 - Relative error): 75%

Records summary

Records included	Records excluded
1,459(100%)	0(0%)

Predictor importance

Input	Value
OverallQual	0.75
TotalBsmtSF	0.16
GarageArea	0.03
BedroomAbvGr	0.03
Fireplaces	0.03

Untitled 1

What is a predictive model for SalePrice ? (Predictive strength: 75%)

Decision Rules | Tree

Decision rules show that OverallQual and 8 other inputs predict SalePrice.

△▼	Predicted value	Rules	Records
	379038.60	OverallQual > 7 GarageCars > 2 GrLivArea > 1,869	81
	287014.06	OverallQual > 7 GarageCars > 2 GrLivArea = 1,339 to 1,869	50
	253065.21	OverallQual > 7 GarageCars <= 2	98
	242561.06	OverallQual = 6 to 7 GrLivArea > 1,869	93
	223201.64	OverallQual = 6 to 7 GrLivArea <= 1,869 TotalBsmtSF > 1,392	53
	203600.67	OverallQual = 5 to 6 GrLivArea > 1,578 LotArea > 10,200	66
	194093.77	OverallQual = 6 to 7 GrLivArea <= 1,869 TotalBsmtSF <= 1,392 Exterior2nd = BrkFace; VinylSd; CmentBd; ImStucc; Other	101

SalePrice

The predictor importance value is less than 0.01 for the following inputs:
GrLivArea
LotArea
Exterior2nd
GarageCars

Test results for high/low analy

Type	Leaf node	t[a]
	(OverallQual> 7) & (GarageCars> 2) & (GrLivArea> 1,869)	44.77
	(OverallQual> 7) & (GarageCars> 2) & (GrLivArea = 1,339 to	18.84

[a]Each t value has df=1,439.
Leaf nodes in the high group have high means compared to the root node.
Leaf nodes in the low group have low means compared to the root node.

Medium leaf nodes, if any, are not listed.

st results for unusually gh/low analysis

e	Leaf node	Modified z

Untitled 1

What is a predictive model for SalePrice ⊗ ? (Predictive strength: 75%)

Decision Rules | Tree

Decision rules show that OverallQual and 8 other inputs predict SalePrice.

△▼	Predicted value	Rules	Records
	170419.47	OverallQual = 6 to 7 GrLivArea <= 1,869 TotalBsmtSF <= 1,392 more...	72
	167787.30	OverallQual = 5 to 6 GrLivArea > 1,578 LotArea <= 10,200	63
	167067.00	OverallQual <= 5 TotalBsmtSF > 1,088 Fireplaces > 0	50
	161037.10	OverallQual = 5 to 6 GrLivArea = 1,339 to 1,578	98
	148131.31	OverallQual = 5 to 6 GrLivArea = 1,065 to 1,339	88
	140255.71	OverallQual <= 5 TotalBsmtSF = 910 to 1,088 GarageArea > 440	52
	138778.05	OverallQual <= 5 TotalBsmtSF > 1,088 Fireplaces <= 0	62

SalePrice

Decision Tree – Showing Predictive model for SalePrice (3/3)

Leaf nodes in the high group have high means compared to the root node.
Leaf nodes in the low group have low means compared to the root node.

Medium leaf nodes, if any, are not listed.

Test results for unusually high/low analysis

Type	Leaf node	Modi z
Unusually high	(OverallQual> 7) & (GarageCars> 2) & (GrLivArea> 1,869)	

Field transformations

OverallQual
- Equal frequency binning was performed.

TotalBsmtSF
- Equal frequency binning was performed.

GarageArea
- Equal frequency binning was performed.

BedroomAbvGr
- Equal frequency binning was performed.

Fireplaces
- Equal frequency binning was performed.

Exterior2nd
- Supervised merge was performed.

LotArea
- Equal frequency binning was performed.

Untitled 1

What is a predictive model for SalePrice ⊗ ? (Predictive strength: 75%)

Decision Rules | Tree

Decision rules show that OverallQual and 8 other inputs predict SalePrice.

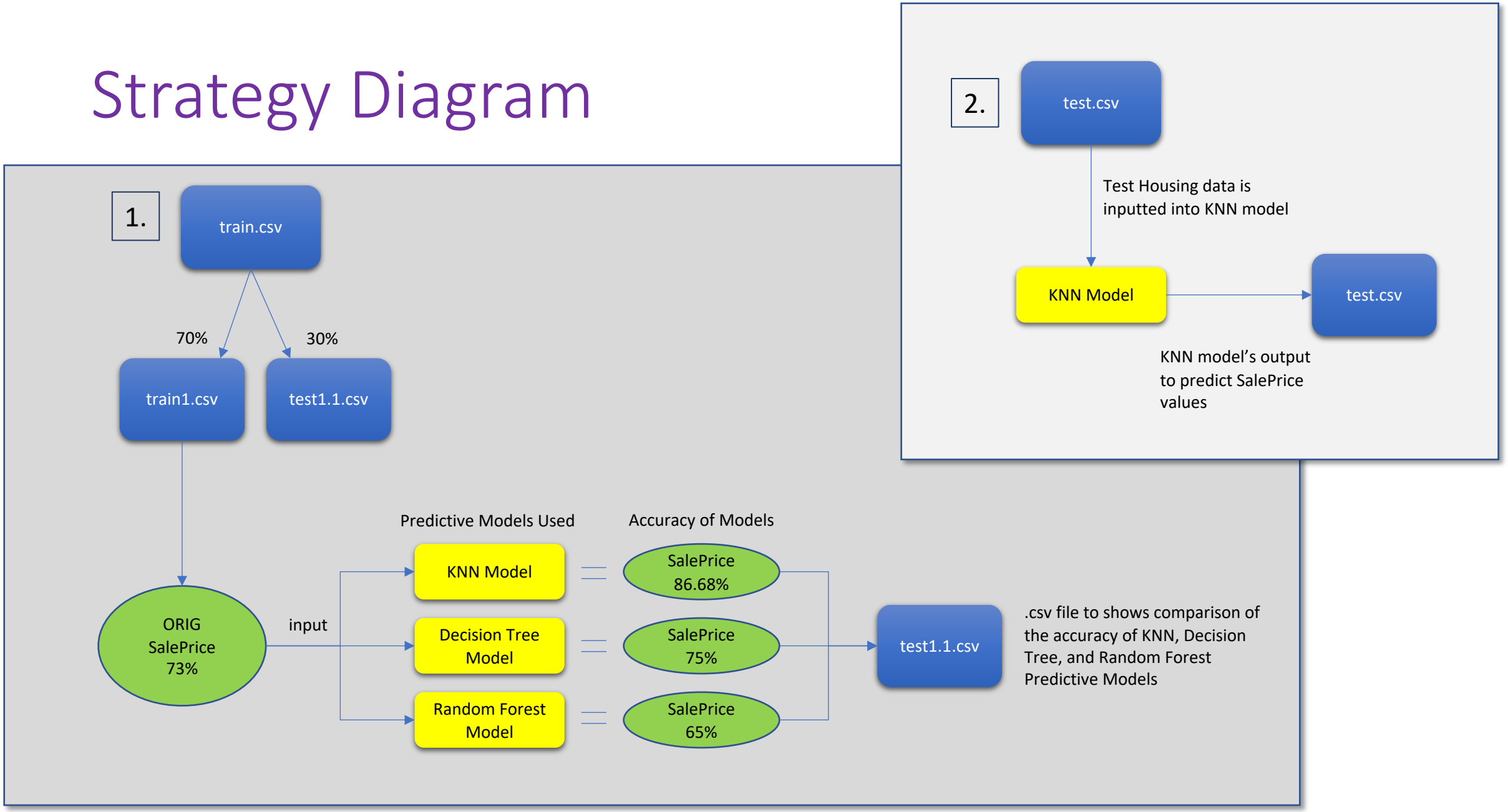
△▼	Predicted value	Rules	Records
	138778.05	OverallQual <= 5 TotalBsmtSF > 1,088 Fireplaces <= 0	62
	129052.54	OverallQual = 5 to 6 GrLivArea <= 1,065	59
	127463.14	OverallQual <= 5 TotalBsmtSF = 755 to 910 BedroomAbvGr > 2	78
	124326.76	OverallQual <= 5 TotalBsmtSF = 910 to 1,088 GarageArea <= 440	88
	119597.57	OverallQual <= 5 TotalBsmtSF <= 755 GrLivArea > 1,065	89
	113742.67	OverallQual <= 5 TotalBsmtSF = 755 to 910 BedroomAbvGr <= 2	55
	85862.76	OverallQual <= 5 TotalBsmtSF <= 755 GrLivArea <= 1,065	63

SalePrice

Strategy Overview

- Divide the training data set into subset of train and test
- (70 percent train and 30 percent test)
- Try KNN, Decision tree and random forest , to predict the SalePrice, using train dataset as reference.
- Once the SalePrice values are predicted , compare it with the values, present in the test set
- Find the accuracy.
- Used R- squared as an estimate of accuracy
- Depending on accuracy, use the method with highest adjusted R square value, to predict the final SalePrice on testing data set.

Strategy Diagram



```r
#Finding the highest correlation between the variables, affecting the saleprice
cor((House$TotalBsmtSF+House$FFlrSF+House$SFlrSF+House$YearRemodAdd+House$YearBuilt+House$YrSold+House$WoodDeckSF+House$GarageArea+House$PoolArea), House$SalePrice , use="complete")

model1 = lm(House$SalePrice~House$TotalBsmtSF+House$FFlrSF+House$SFlrSF+House$YearRemodAdd+House$YearBuilt+House$YrSold+House$WoodDeckSF+House$GarageArea+House$PoolArea)
plot(model1)
summary(model1)
```

```r
#Linear model to check for significance of paramaters affecting sale price

model_one = lm(train_df$SalePrice~train_df$OverallQual+train_df$OverallCond)
plot(model_one)
summary(model_one)
```

```
> summary(model_one)

Call:
lm(formula = train_df$SalePrice ~ train_df$OverallQual + train_df$OverallCond)

Residuals:
    Min      1Q  Median      3Q     Max
-177762  -29724   -1806   21556  393707

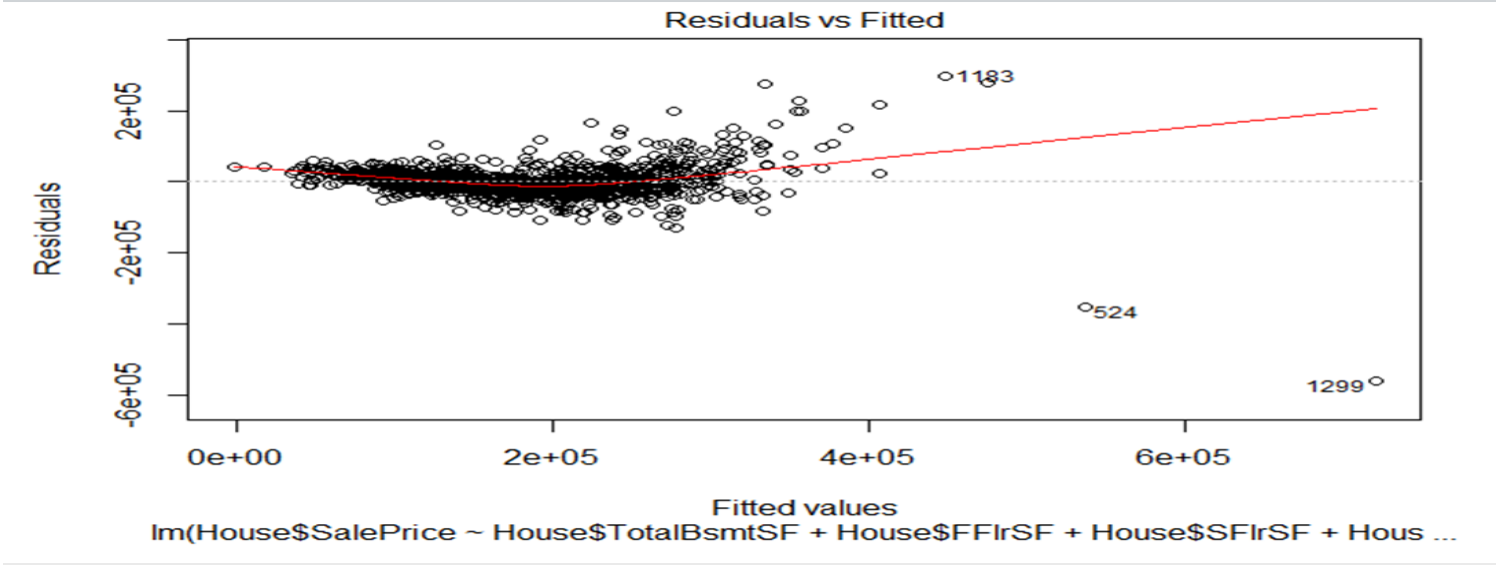
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          -94578      10858  -8.710   <2e-16 ***
train_df$OverallQual  46319       1110  41.738   <2e-16 ***
train_df$OverallCond  -1219       1378  -0.885    0.377
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 48310 on 1005 degrees of freedom
Multiple R-squared:  0.6373,    Adjusted R-squared:  0.6366
F-statistic: 883.1 on 2 and 1005 DF,  p-value: < 2.2e-16
```

- Relationship between data points Predictor variables vs response
- Finding the variables that are correlated to the sale price
- The correlation coefficient between sale price and predictors is 0.83 , which shows a strong relationship

Regression Analysis

```
model1 = lm(House$SalePrice~House$TotalBsmtSF+House$FFlrSF+House$SFlrSF+House$YearRemodAdd+House$Yea
rBuilt+House$YrSold+House$WoodDeckSF+House$GarageArea+House$PoolArea)
plot(model1)
summary(model1)
```



Linear Regression between response and predictors

Linear Regression

Original Output % of Variance

```
Residuals:
    Min      1Q  Median      3Q     Max
-561034  -18947   -3130   14615  295965

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -5.522e+05  1.646e+06  -0.335   0.7373
House$TotalBsmtSF  3.549e+01  4.529e+00   7.835 9.01e-15 ***
House$FFlrSF       7.228e+01  5.143e+00  14.055  < 2e-16 ***
House$SFlrSF       6.871e+01  2.727e+00  25.199  < 2e-16 ***
House$YearRemodAdd  5.425e+02  6.706e+01   8.090 1.25e-15 ***
House$YearBuilt    4.484e+02  4.919e+01   9.117  < 2e-16 ***
House$YrSold      -6.993e+02  8.199e+02  -0.853   0.3938
House$WoodDeckSF   3.929e+01  9.242e+00   4.251 2.27e-05 ***
House$GarageArea   5.820e+01  6.650e+00   8.752  < 2e-16 ***
House$PoolArea    -5.105e+01  2.752e+01  -1.855   0.0638 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 41390 on 1449 degrees of freedom
Multiple R-squared:  0.7303,    Adjusted R-squared:  0.7287
F-statistic:   436 on 9 and 1449 DF,  p-value: < 2.2e-16
```

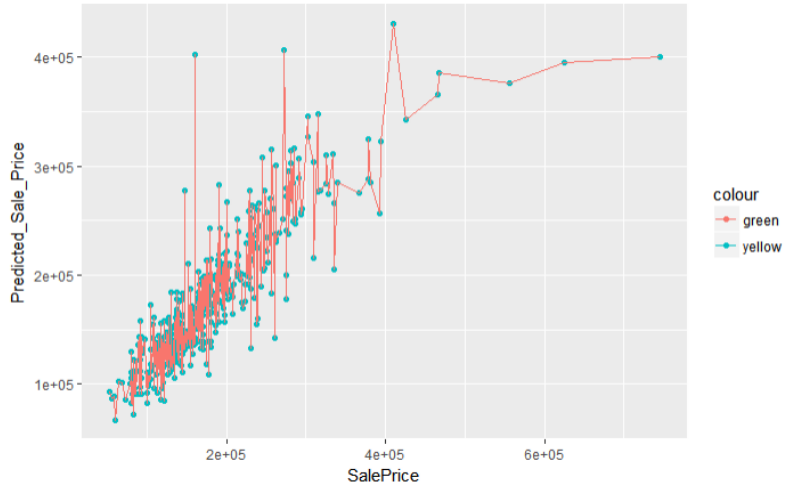
KNN Predictive Modeling

- Train control function is of the Caret package
- We choose the repeated cross validation
- Repeated = 3 , repeating the procedure thrice for
- consistency
- Default we have used is 10 folds
- Predict the value for test based on performance of train
- data.
- **Accuracy is 86 %**

```
## Using K- Nearest Neighbors to predict the value of sale price.  Accuracy is 86 percent
library(caret)
library(ggplot2)
trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
set.seed(3333)
knn_fit <- train(SalePrice ~
TotalBsmtSF+FFlrSF+SFlrSF+YearRemodAdd+YearBuilt+YrSold+WoodDeckSF+GarageArea+PoolArea, data =
train_df, method = "knn", trControl=trctrl, preProcess = c("center", "scale"), tuneLength = 10)

 test_pred <- predict(knn_fit, newdata = test_df)
 summary(test_pred)
 #Printing the cost prediction on the test data file to a file named test_pred.txt

write.table(test_pred,file="tpredict1.txt", sep="\n",row.names= TRUE,col.names=NA)
```



Resampling results across tuning parameters

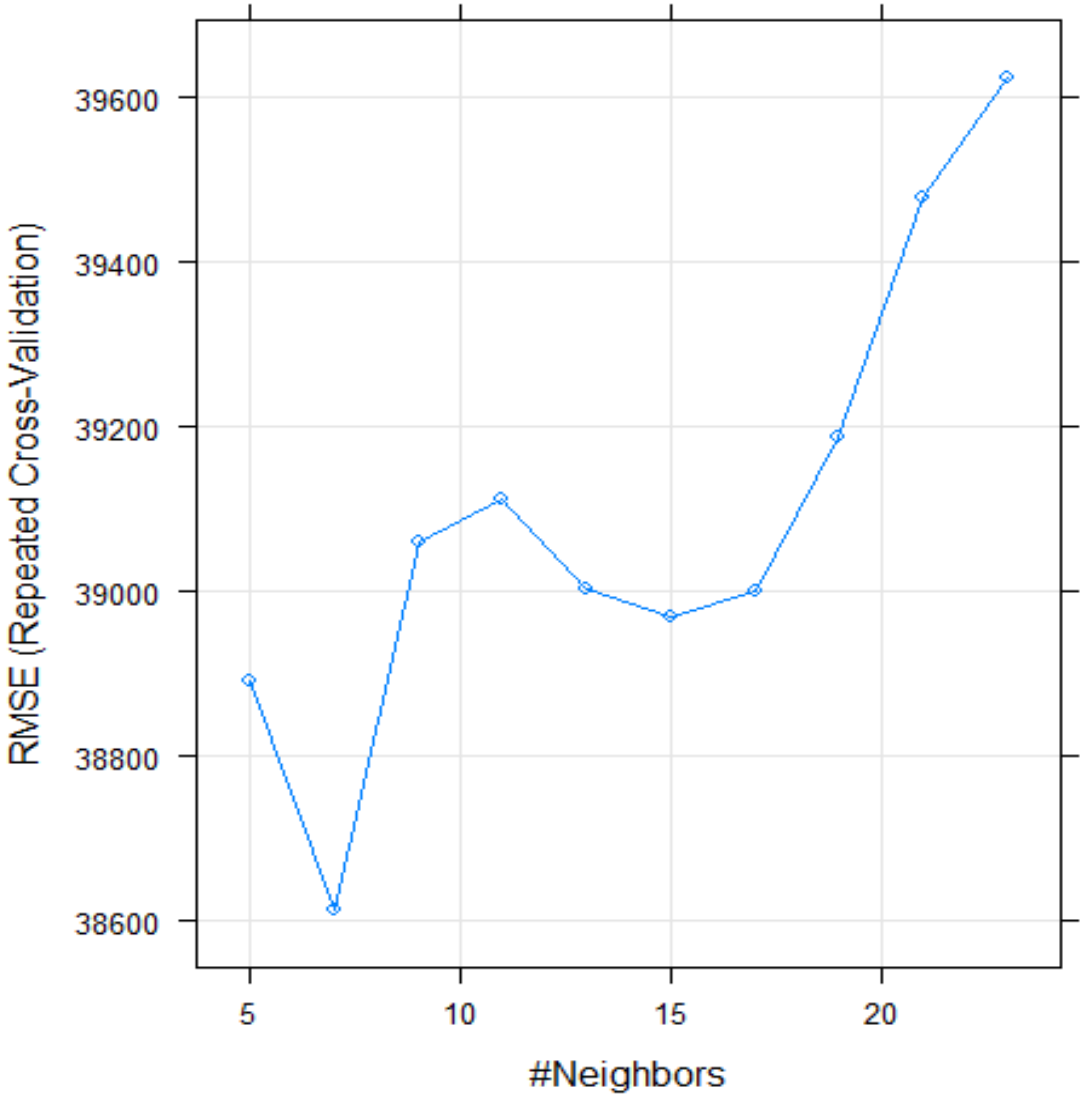
```
> knn_fit
k-Nearest Neighbors

1008 samples
   9 predictor

Pre-processing: centered (9), scaled (9)
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 908, 908, 906, 906, 907, 906, ...
Resampling results across tuning parameters:

  k   RMSE      Rsquared   MAE
   5  38891.08  0.7670992  25031.36
   7  38614.32  0.7736677  24603.77
   9  39059.63  0.7715043  24515.15
  11  39113.34  0.7730315  24353.28
  13  39002.03  0.7768252  24163.21
  15  38969.91  0.7789462  24043.92
  17  38999.60  0.7814577  24088.65
  19  39186.17  0.7811951  24142.84
  21  39478.43  0.7794527  24310.19
  23  39622.78  0.7790840  24382.80

RMSE was used to select the optimal model using  the smallest value.
The final value used for the model was k = 7.
>
```



KNN - Output % of Variance

```
Call:
lm(formula = Predicted_Sale_Price ~ TotalBsmtSF + FFlrSF + SFlrSF +
    YearRemodAdd + YearBuilt + YrSold + WoodDeckSF + GarageArea +
    PoolArea, data = test_df)

Residuals:
    Min      1Q  Median      3Q     Max
-180171  -12618   -1490   10955  138814

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.142e+06  1.656e+06   0.690   0.4906
TotalBsmtSF  3.257e+01  4.113e+00   7.917 1.98e-14 ***
FFlrSF       5.733e+01  4.775e+00  12.005  < 2e-16 ***
SFlrSF       5.428e+01  2.728e+00  19.897  < 2e-16 ***
YearRemodAdd 5.605e+02  6.779e+01   8.268 1.61e-15 ***
YearBuilt    5.398e+02  5.066e+01  10.655  < 2e-16 ***
YrSold      -1.630e+03  8.225e+02  -1.981   0.0482 *
WoodDeckSF   5.638e+01  9.466e+00   5.956 5.28e-09 ***
GarageArea   1.217e+01  6.366e+00   1.911   0.0566 .
PoolArea    -9.493e+01  1.908e+01  -4.975 9.36e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23040 on 442 degrees of freedom
Multiple R-squared:  0.8665,    Adjusted R-squared:  0.8637
F-statistic: 318.7 on 9 and 442 DF,  p-value: < 2.2e-16
```

Decision Tree Predictive Modeling

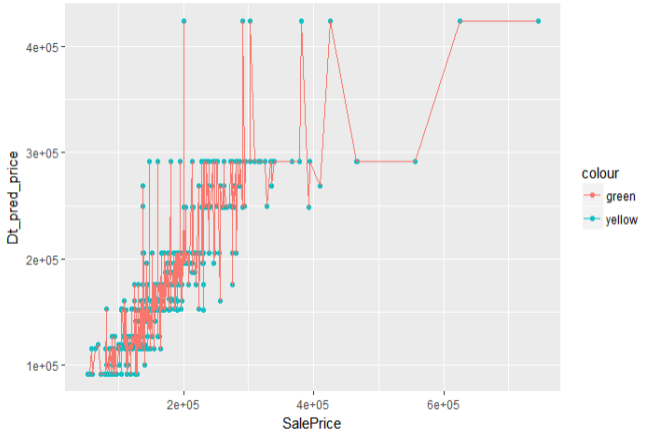
- Regression tree is used, cause of the continuous, dependent variable
- Load the r part package
- Trees divide the predictor space (independent variables) into distinct and non-overlapping regions
- Use the rpart function, specifying the model formula, data and parameters
- Split is based on the values of cut points.
- **Accuracy is 75%**

```
#Decision trees to predict the sale price
target = train_df$SalePrice ~train_df$TotalBsmtSF +train_df$FFlrSF+
train_df$SFlrSF+train_df$YearRemodAdd+train_df$YearBuilt+train_df$YrSold+train_df$WoodDeckSF+tra
in_df$GarageArea+train_df$PoolArea

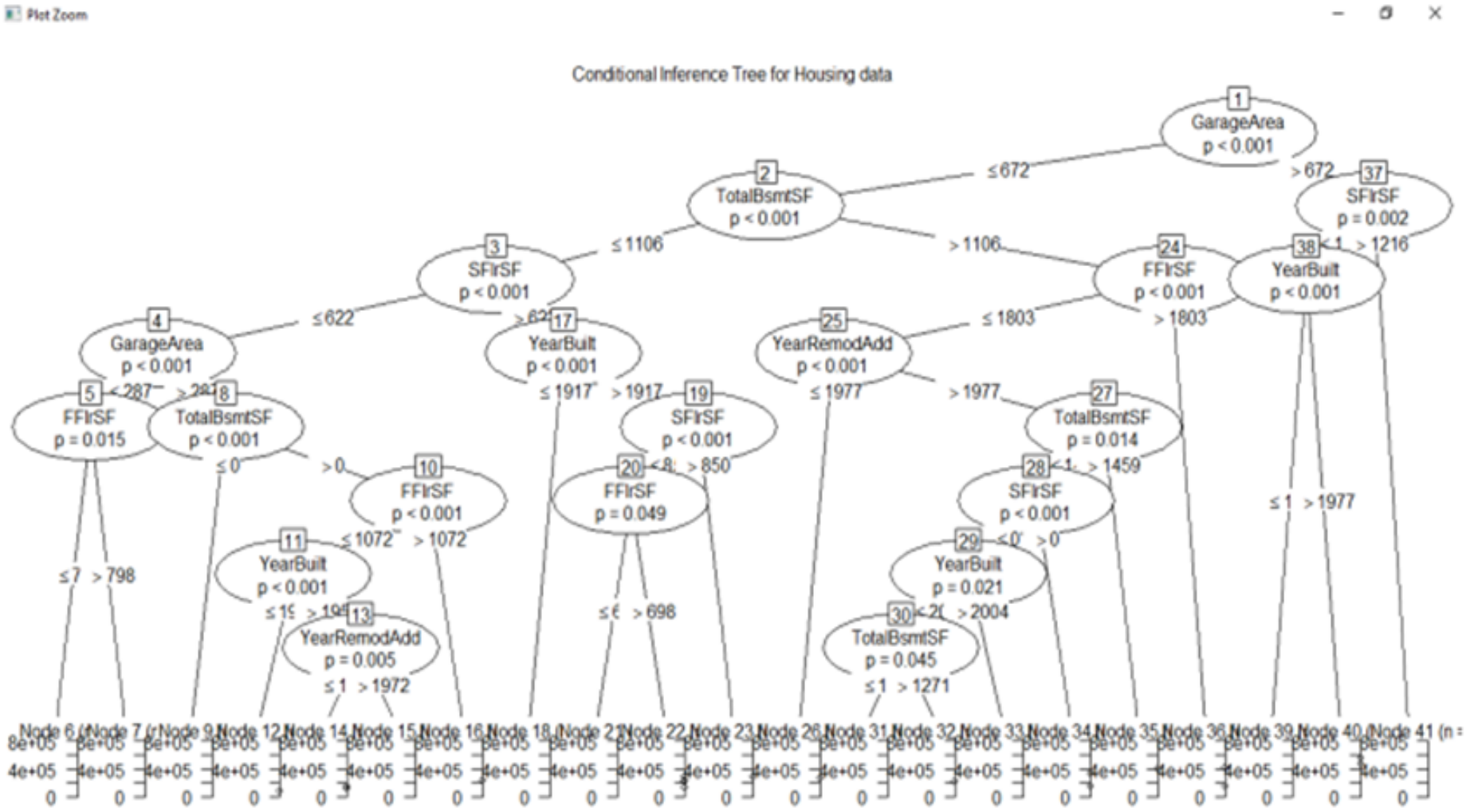

tree = rpart(target, data = train_df, method = "class")
rpart.plot(tree)
#install.packages("party")
library(party)
tree = ctree(SalePrice ~ TotalBsmtSF
+FFlrSF+SFlrSF+YearRemodAdd+YearBuilt+YrSold+WoodDeckSF+GarageArea+PoolArea, data = test_df)
plot(tree, main="Conditional Inference Tree for Housing data")
conf = table(predict(tree), test_df$SalePrice)

 treeop = predict(tree)


write.table(treeop, file="output_price.csv")
```



Decision Tree



Decision Tree - Output % of Variance

```
Call:
lm(formula = Dt_pred_price ~ TotalBsmtSF + FFlrSF + SFlrSF +
    YearRemodAdd + YearBuilt + YrSold + WoodDeckSF + GarageArea +
    PoolArea, data = test_df)

Residuals:
    Min      1Q  Median      3Q     Max
-263747  -17445   -3248   16458  152152

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.444e+06  2.369e+06   0.610 0.542272
TotalBsmtSF  2.302e+01  5.885e+00   3.912 0.000106 ***
FFlrSF       5.151e+01  6.832e+00   7.539 2.71e-13 ***
SFlrSF       5.385e+01  3.903e+00  13.798  < 2e-16 ***
YearRemodAdd 1.877e+02  9.698e+01   1.935 0.053632 .
YearBuilt    5.745e+02  7.248e+01   7.927 1.85e-14 ***
YrSold      -1.449e+03  1.177e+03  -1.232 0.218789
WoodDeckSF   1.779e+01  1.354e+01   1.314 0.189615
GarageArea   7.268e+01  9.107e+00   7.980 1.27e-14 ***
PoolArea    -1.049e+02  2.730e+01  -3.843 0.000139 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32960 on 442 degrees of freedom
Multiple R-squared:  0.7517,    Adjusted R-squared:  0.7466
F-statistic: 148.7 on 9 and 442 DF,  p-value: < 2.2e-16
```

Random Forest Predictive Modeling

- The random forest regression is used to predict values
- The idea is that real data points that are similar to one another will frequently end up in the same terminal node of a tree
- Exactly ,what is measured by the proximity matrix that can be returned using the proximity=TRUE option of randomForest.
- Thus the proximity matrix can be taken as a similarity measure
- **Accuracy is 65%**

```
## RandomForest method to predict the values. Accuracy is 65 percent

You can also embed plots, for example:

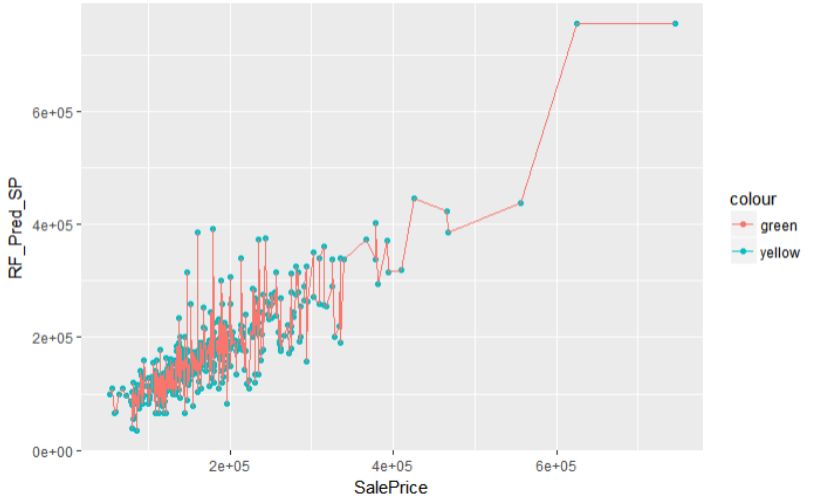
```{r}
library(party)
library(randomForest)
test_df = read.csv("test1.1.csv")


fit_model <- randomForest(as.factor(SalePrice) ~
TotalBsmtSF+FFlrSF+SFlrSF+YearRemodAdd+YearBuilt+YrSold+WoodDeckSF+GarageArea+PoolArea,
                      data=train_df,
                      importance=TRUE,
                      ntree=200)

Prediction <- predict(fit_model, test_df)
 submit <- data.frame(Id = test_df$Id, SalePrice = Prediction)
 write.csv(submit, file = "firstforest.csv", row.names = FALSE)

 modacc = lm(RF_Pred_SP~TotalBsmtSF+FFlrSF+SFlrSF+YearRemodAdd+YearBuilt+GarageArea,
data=test_df)

 plot(modacc)
 summary(modacc)
```


```

Random Forest – Output % of Variance

```
Call:
lm(formula = RF_Pred_SP ~ TotalBsmtSF + FFlrSF + SFlrSF + YearRemodAdd +
    YearBuilt + GarageArea, data = test_df)

Residuals:
    Min      1Q  Median      3Q     Max
-283133  -19759   -4621   15400  378231

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.087e+06  2.250e+05  -9.274  < 2e-16 ***
TotalBsmtSF   3.914e+01  8.268e+00   4.734 2.96e-06 ***
FFlrSF        4.867e+01  9.515e+00   5.115 4.66e-07 ***
SFlrSF        6.770e+01  5.290e+00  12.799  < 2e-16 ***
YearRemodAdd  4.649e+02  1.363e+02   3.411 0.000707 ***
YearBuilt     6.050e+02  1.013e+02   5.974 4.74e-09 ***
GarageArea    5.290e+01  1.281e+01   4.131 4.32e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 46570 on 445 degrees of freedom
Multiple R-squared:  0.6619,    Adjusted R-squared:  0.6573
F-statistic: 145.2 on 6 and 445 DF,  p-value: < 2.2e-16
```

Predictive Model Comparison

Dataset	R Squared Value	Adjusted R Squared Value
Train (Original SalePrice)	77.3	77.1
Test (**KNN**)	86.65	86.37
Test (Decision Tree)	75.17	74.66
Test (Random Forest)	66.19	65.73

- KNN's accuracy was ~9% higher than original SalePrice accuracy
- KNN's accuracy was >10% higher than Decision Tree's accuracy
- KNN's accuracy was >20% higher than Random Forest's accuracy

- Since KNN Predictive Model had the highest Adjusted R Squared Value and provided an accuracy % that we considered within the range of reasonable, we decided to proceed with the KNN Predictive Model as the method to predict the final SalePrice on the Test data set

Conclusion – Addressing S.M.A.R.T. Question

- If a realtor or home buyer/seller in Ames, Iowa were to ask us if to provide them with predicted home sales prices, based on this research we are confident that we would be able to provide them a predicted Sales Price using a KNN model that would provide them a Sales Price ~86% accurate.
- We would also be able to provide them with a predicted min and max Sales Prices. For Example:
    - Clients provide us with the identified key home-related variables to feed KNN model
    - We would provide them with a predicted Sales Price (i.e. $265,000) with 86% accuracy
    - We would also provide predicted min Sales Price ($227,425) and max Sales price ($287,200)

```
> head(test_val$SalePrice)    #Actual sale Price
[1]  91000 206000  82000  86000 232000 136905
> head(test_val$Knn_predicted_price)  #KNN predicted sale price
[1] 115557.0 195459.6 100244.4  91300.0 291458.2 160693.1
>
```

Conclusion

- We, could effectively predict the sales price of the houses, given the train and test data set
- With the help of variables, that had the highest significance in relation to the Sale Price.
- The train set had adjusted R square of 72 percent , using sale price as response variable and the same combination of predictors
- Out of all the methods employed to predict the sale price, KNN outperformed the rest like decision trees and random forest, yielding an adjusted R squared rate of 86 %, even higher than the train data.

Background/Reference Slides

References

- The train and test data set is retrieved from *https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data*.