

Assignment 2: Regression Model Building

Sri Seshadri

7/1/2017

1. Introduction

This report discusses the regression models for estimating or predicting the sales price of “typical” homes in Ames, Iowa.

2. Sample definition

It is assumed that typical home buyers are those that move from apartments to single family or town homes. Also apartments are less likely to be sold to individuals as they remain holdings of owners for rental income. Single family and town homes belong to “Residential Low density” (RL) zoning classification in the city of Ames. Data belonging to only to the RL zone is considered for analysis and model development. Also, it is assumed that typical homes have paved streets for access and above grade living area greater than 800 square feet. Sales data belonging to homes that were sold in abnormal conditions such as trade in, foreclosure or short sale are not included in the analysis. Also, sales between family members, sale of adjoining lot, linked properties are omitted from the data. Homes with no basements are excluded from the analysis at this time. Table 1 shows the waterfall of the data not included in the data and the eligible samples.

Table 1: Drop waterfall

DropCondition	counts
01: Not LowDensityZone	657
02: Not Normal/Partial Sale	189
03: Street Not Paved	3
04: Less than 800 SqFt	41
05: No Basement	48
99: Eligible Sample	1992

2.1 Variables of interest for modelling

The following variables in the data were deemed to be of interest for model building. The choice of parameters was based upon initial Exploratory Data Analysis (EDA) and subject matter expertise. See appendix A.1 for data quality checks.

Table 2: Variables of interest

LotArea	TotalBsmtSF	KitchenQual
LotConfig	GrLivArea	TotRmsAbvGrd
Neighborhood	BsmtFullBath	GarageArea
BldgType	BsmtHalfBath	MoSold
HouseStyle	FullBath	YrSold
OverallCond	HalfBath	SaleCondition
YearRemodel	BedroomAbvGr	SalePrice

2.2 Training and validation samples.

From the eligible samples, 70% of the data is randomly samples to be used as the dataset on which model is developed from. This dataset would be referred to as training dataset. The remaining 30% is used as the validation set to evaluate the model performance of predicting sale price on data that is outside the training set. Table 3 shows the split of the total eligible samples.

Table 3: Training and Validation sampling

Data	Samples
Training set	1394
Validation set	597

3. Exploratory Data Analysis (EDA)

For exploratory analysis, entire sample frame is used, so any anomalies that were missed surfaces in this process. While EDA by itself doesn't become an data cleaning step, but certainly allows the opportunity to identify issues in the data. In this section we will focus only on the continuous variable. Exploratory analysis on categorical variables can be found in https://github.com/srivathses/RegressionAnalysis/blob/master/Assignment1_Seshadri.pdf.

3.1 EDA of continuous variables

It is hypothesized that bigger the house, more likely is higher occupancy and sale value. The higher occupancy means a likely higher lot area, living area, basement area, garage space for parking, total number of rooms and baths. Also it is hypothesized that newer homes are likely to be more valued than the older homes. Before exploring each of the potential predictors, it will be useful to see if there is multicollinearity amongst the predictors. Figure 1 shows, potential relationship between Above grade living area, Sale price and Total basement area. The lot area may have a steeper slope with sale price. This warrants a closer look.

Figure 2 shows the relationship between total basement area and the sale price. There are few outliers in the basement area. To explore the relationship better, data corresponding to basement area above 2500 square feet is removed from the plot. It is seen that total basement area seems very promising predictor. It may be used along with other predictors to model sales price.

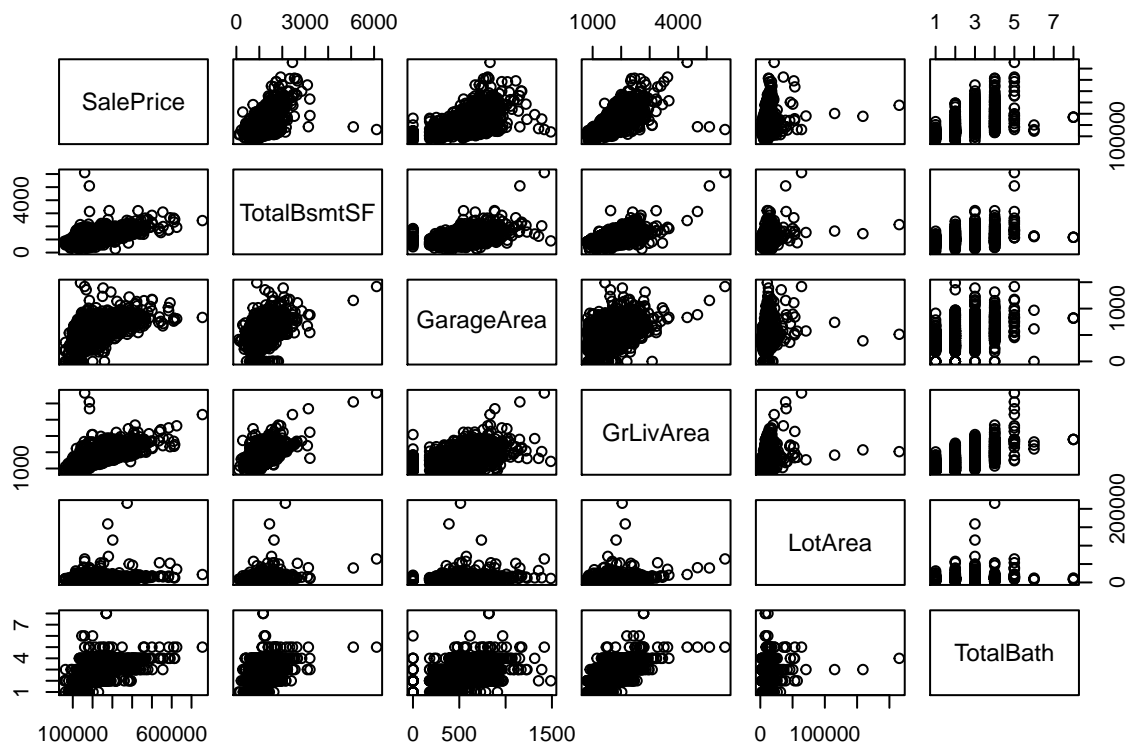


Figure 1: Multicollinearity check amongst potential predictors

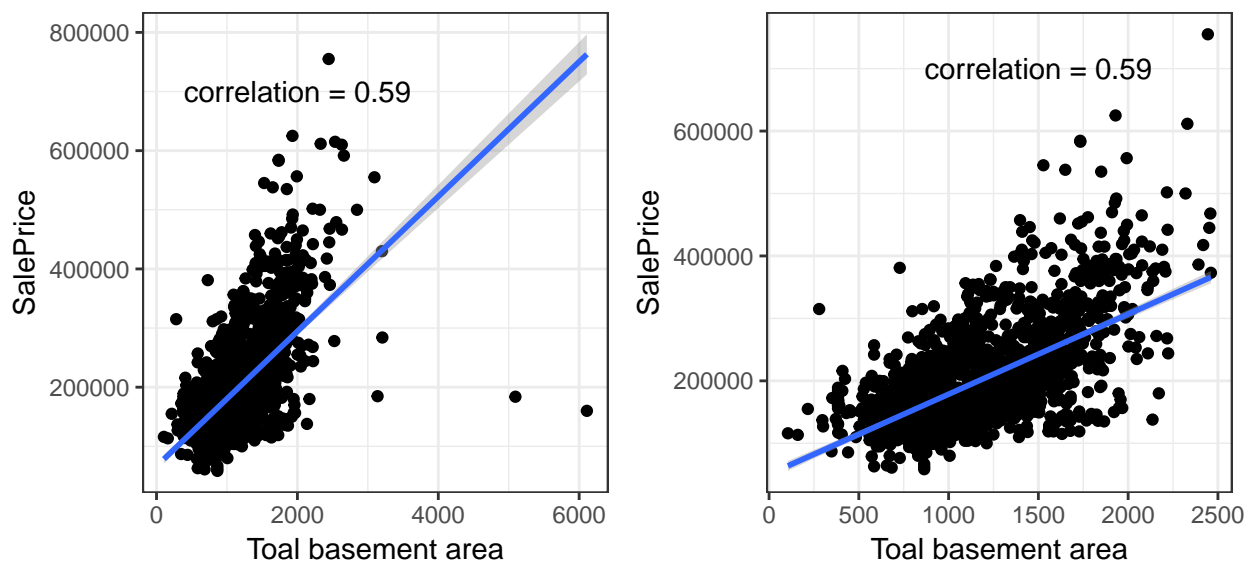


Figure 2: Sale price vs Basement area

Figure 3 shows the relationship between Living area and Total basement area as well as Sale price. There seem to be a linear relationship between living area and sale price when living area is less than 4000 square feet. Likewise with Garage area when its greater than zero and less than 1000 square feet. Figure 4 explores if there is a relationship between Garage Area, Living area and Total basement area. There is a weak correlation between living area and total basement area.

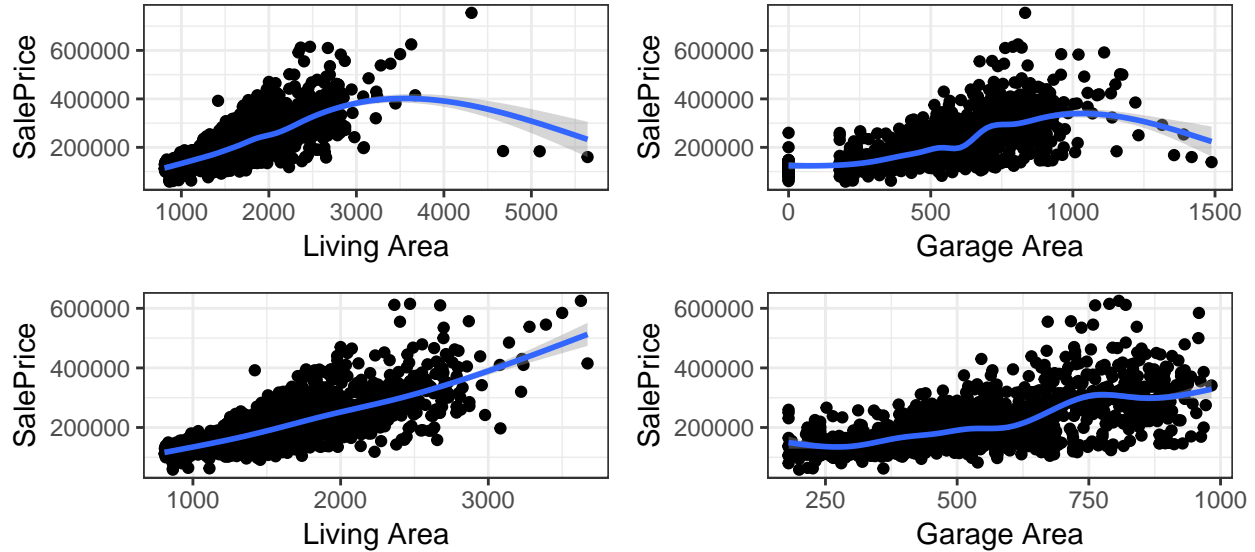


Figure 3: Sales Price vs Garage Area and LivingArea

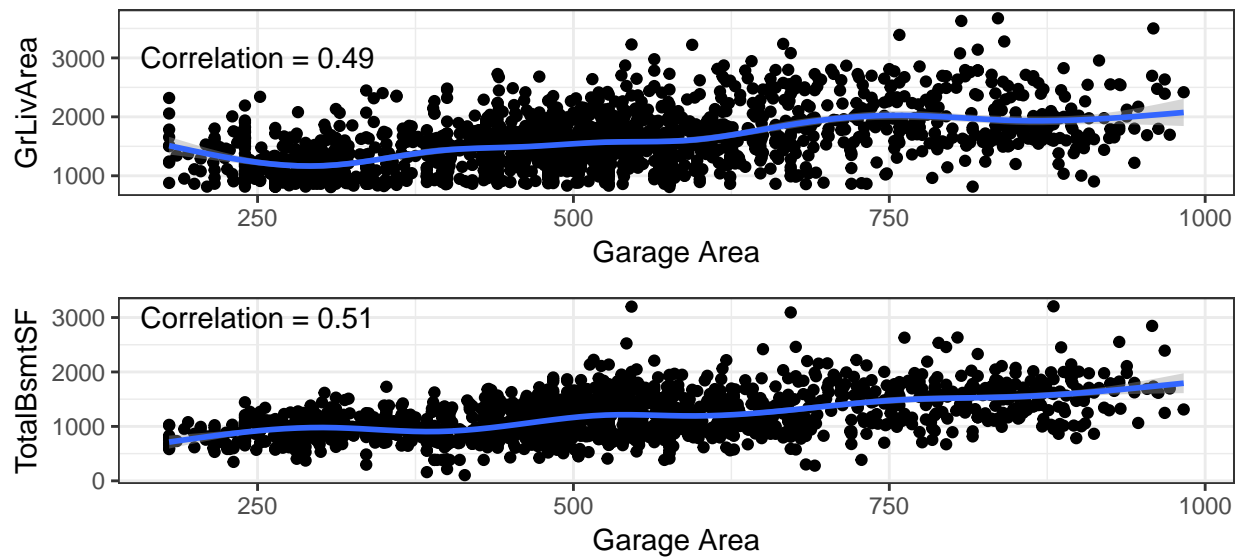


Figure 4: GarageArea vs Living and Basement area

4. Simple Linear Regression Models

The Above grade living area and total basement seem to be good two candidates for building a regression model. The regression models would built on the training data set. However, after the EDA, it seems best to remove samples that correspond to above grade living area of 4000 square feet or greater and garage area greater than 1000 square feet. While it is best to update the drop conditions in section 2, at this time, the additional drop conditions would be applied to the training set. The training set data with new drop conditions applied is called “trainFiltered” (The R output reference this as the data)"

4.1 Simple linear regression model with “above grade living area” as predictor

The model fit results for a single linear regression model with Living area above grade as predictor is seen below. The intercept = 0 hypothesis if failed to be rejected. It would be appropriate to fit a no intercept model. From a goodness of fit perspective, the residuals are of mean 0 and distributed fairly normally from a “thick pen test” of the Q-Q plot. However, the residulas are not homoscedastic, the residuals vs predictor have a bullhorn shape, i.e. the varaition in the residuals increases as the living area increases. The model is not suitable.

```
##
## Call:
## lm(formula = SalePrice ~ GrLivArea, data = trainFiltered)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -188199  -27705   -3701   19890  314519
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7200.401   4776.743   1.507   0.132
## GrLivArea    122.647     2.964  41.375 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 51660 on 1374 degrees of freedom
## Multiple R-squared:  0.5547, Adjusted R-squared:  0.5544
## F-statistic: 1712 on 1 and 1374 DF, p-value: < 2.2e-16
```

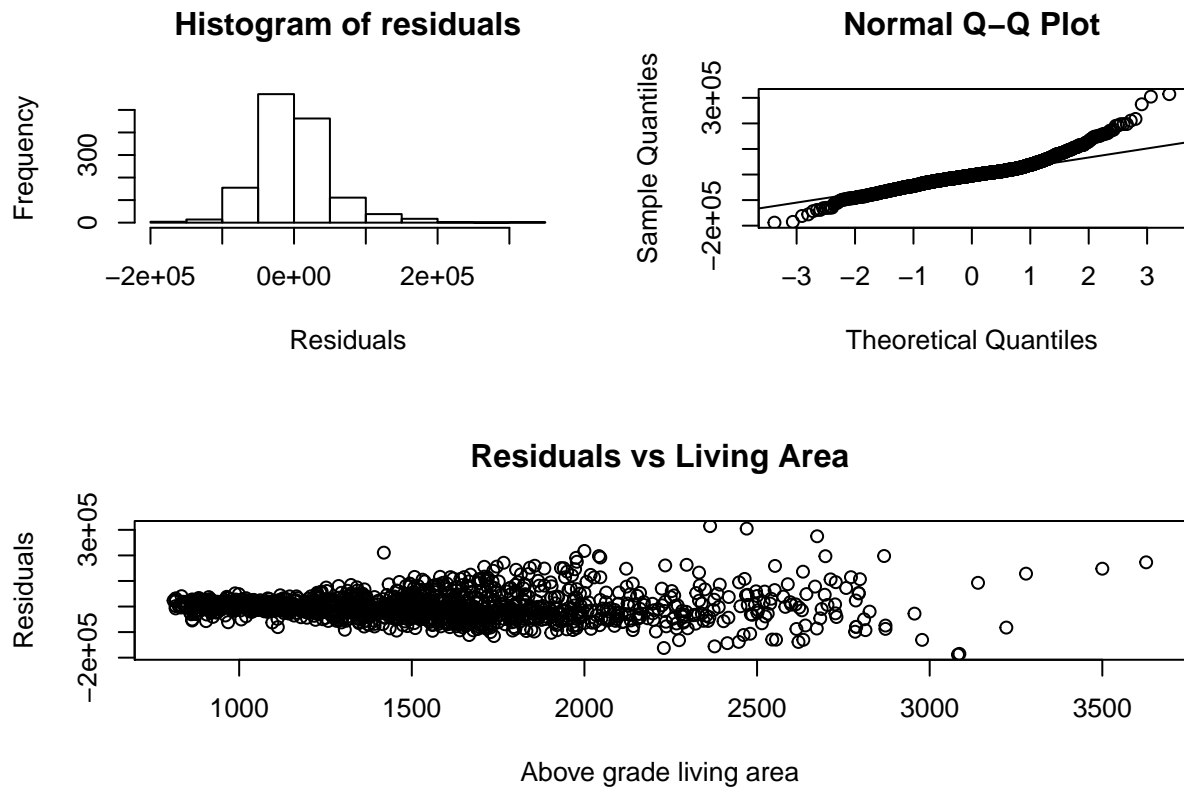


Figure 5: Model diagnostics Sale Price ~ GrLivArea

4.1.1 Non - intercept model with Above grade living area as predictor

APPENDIX

A.1 Data quality check

Tables below shows the summary statistics of the numeric variables and it is noted that statistics are within reasonable bounds and appear to be in the units of measure as described in the data dictionary with only 2 rows missing. Also shown are the number of levels or categories in the nominal variables and the number of missing data (0 missing). The data is deemed usable.

Table 4: Data sanity check for numeric variables

	min	Q1	median	Q3	max	mean	sd	n	missing
SID	1	675.75	1467.5	2171.25	2930	1e+03	8e+02	1992	0
LotArea	1700	8400.00	10030.5	12227.25	215245	1e+04	8e+03	1992	0
OverallCond	1	5.00	5.0	6.00	9	6e+00	1e+00	1992	0
YearRemodel	1950	1969.75	1995.0	2004.00	2010	2e+03	2e+01	1992	0
TotalBsmtSF	105	864.00	1076.0	1393.25	6110	1e+03	4e+02	1992	0
GrLivArea	808	1196.00	1499.5	1800.25	5642	2e+03	5e+02	1992	0
BsmtFullBath	0	0.00	0.0	1.00	2	5e-01	5e-01	1992	0
BsmtHalfBath	0	0.00	0.0	0.00	2	6e-02	3e-01	1992	0
FullBath	0	1.00	2.0	2.00	4	2e+00	5e-01	1992	0
HalfBath	0	0.00	0.0	1.00	2	4e-01	5e-01	1992	0
BedroomAbvGr	0	3.00	3.0	3.00	6	3e+00	8e-01	1992	0
TotRmsAbvGrd	3	6.00	6.0	7.00	15	7e+00	1e+00	1992	0
GarageArea	0	390.00	490.0	605.25	1488	5e+02	2e+02	1992	0
MoSold	1	4.00	6.0	8.00	12	6e+00	3e+00	1992	0
YrSold	2006	2007.00	2008.0	2009.00	2010	2e+03	1e+00	1992	0
SalePrice	58500	143000.00	177547.0	230000.00	755000	2e+05	8e+04	1992	0
TotalBath	1	2.00	3.0	3.00	8	3e+00	9e-01	1992	0

Table 5: Data sanity check for nominal variables

	# Unique	n	missing
LotConfig	5	1992	0
Neighborhood	22	1992	0
BldgType	5	1992	0
HouseStyle	8	1992	0
KitchenQual	5	1992	0
SaleCondition	2	1992	0

A.2 R code

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 60),
  tidy = TRUE)
ames <- readr::read_delim(file = "ames_housing_data.csv", delim = ",")
# change from scientific notations, to restore to default
# options(scipen = 0)
options(scipen = 999)
library(magrittr)
LivingAreaCutoff <- 800
# Adding drop conditions variable insert dummy variable to
# code SaleCondition being either equal to 'Normal' or
# 'Partial'
ames$Sale_NrmPar <- ifelse(ames$SaleCondition == "Normal" | ames$SaleCondition ==
  "Partial", 1, 0)
ames$DropCondition <- ifelse(ames$Zoning != "RL", "01: Not LowDensityZone",
  ifelse(ames$Sale_NrmPar == 0, "02: Not Normal/Partial Sale",
    ifelse(ames$Street != "Pave", "03: Street Not Paved",
      ifelse(ames$GrLivArea < LivingAreaCutoff, "04: Less than 800 SqFt",
        ifelse(ames$TotalBsmtSF < 1, "05: No Basement",
          "99: Eligible Sample"))))))

# Waterfall
waterfall <- ames %>% dplyr::group_by(DropCondition) %>% dplyr::summarise(counts = n())

# Print waterfall table
knitr::kable(waterfall, align = c("l", "r"), caption = "Drop waterfall")
# Define training portion of the data
trainPercent <- round(0.7, 1)
# Columns of interest
colsofinterest <- c("SID", "LotArea", "LotConfig", "Neighborhood",
  "BldgType", "HouseStyle", "OverallCond", "YearRemodel", "TotalBsmtSF",
  "GrLivArea", "BsmtFullBath", "BsmtHalfBath", "FullBath",
  "HalfBath", "BedroomAbvGr", "KitchenQual", "TotRmsAbvGrd",
  "GarageArea", "MoSold", "YrSold", "SaleCondition", "SalePrice")

# Cleanly show the columns of interest in pdf. Making the
# colsofinterest as matrix for easy printing.
colsmatrix <- matrix(colsofinterest[2:length(colsofinterest)],
  ncol = 3)
# printing on pdf
knitr::kable(colsmatrix, caption = "Variables of interest")
# Get sample frame.
SampleFrame <- ames %>% dplyr::filter(DropCondition == "99: Eligible Sample") %>%
  dplyr::select_(.dots = colsofinterest)
SampleFrame <- SampleFrame %>% dplyr::mutate(TotalBath = BsmtFullBath +
  BsmtHalfBath + FullBath + HalfBath)
# training set
train <- dplyr::sample_n(SampleFrame, size = trainPercent * nrow(SampleFrame),
  replace = F, set.seed(2000))
train <- train %>% dplyr::arrange(SID)
# Validation set
Validation <- dplyr::sample_n(SampleFrame, size = (1 - trainPercent) *
```



```

    nrow(SampleFrame), replace = F, set.seed(2000))
Validation <- Validation %>% dplyr::arrange(SID)
# Check row counts
df <- cbind(Data = c("Training set", "Validation set"), Samples = c(nrow(train),
    nrow(Validation)))
knitr::kable(df, align = c("l", "r"), caption = "Training and Validation sampling")
# Multicollinearity exploration
test <- SampleFrame[, c("SalePrice", "TotalBsmtSF", "GarageArea",
    "GrLivArea", "LotArea", "TotalBath")]
pairs(test)
# Plot of basement area vs sale price
library(ggplot2)
library(gridExtra)
BasementArea <- ggplot(data = SampleFrame, mapping = aes(x = TotalBsmtSF,
    y = SalePrice)) + geom_point() + geom_smooth(method = "lm",
    se = T) + xlab("Toal basement area") + theme_bw()
# annotation addition
BasementArea <- BasementArea + annotate("text", x = 2000, y = 7e+05,
    label = paste0("correlation = ", round(cor(SampleFrame$TotalBsmtSF,
    SampleFrame$SalePrice), 2)))
# Restrict basment to < 2500
RestrictedBsmtSF <- SampleFrame %>% dplyr::filter(TotalBsmtSF >
    0 & TotalBsmtSF < 2500)
# Basement < 2500 vs SalePrice
RestrictedBasement <- ggplot(data = RestrictedBsmtSF, mapping = aes(x = TotalBsmtSF,
    y = SalePrice)) + geom_point() + geom_smooth(method = "lm",
    se = T) + xlab("Toal basement area") + theme_bw()
# Annotations
RestrictedBasement <- RestrictedBasement + annotate("text", x = 1500,
    y = 7e+05, label = paste0("correlation = ", round(cor(SampleFrame$TotalBsmtSF,
    SampleFrame$SalePrice), 2)))
# Print plot
grid.arrange(BasementArea, RestrictedBasement, ncol = 2)
LivingArea <- ggplot(SampleFrame) + geom_point(mapping = aes(x = GrLivArea,
    y = SalePrice)) + xlab("Living Area") + geom_smooth(mapping = aes(x = GrLivArea,
    y = SalePrice), se = T) + theme_bw()

GargeArea <- ggplot(SampleFrame) + geom_point(mapping = aes(x = GarageArea,
    y = SalePrice)) + xlab("Garage Area") + geom_smooth(mapping = aes(x = GarageArea,
    y = SalePrice), se = T) + theme_bw()

RestrictedLivingGarageArea <- SampleFrame %>% dplyr::filter(GrLivArea <
    4000 & GarageArea > 0 & GarageArea < 1000)

LivingAreaRestricted <- ggplot(RestrictedLivingGarageArea) +
    geom_point(mapping = aes(x = GrLivArea, y = SalePrice)) +
    xlab("Living Area") + geom_smooth(mapping = aes(x = GrLivArea,
    y = SalePrice), se = T) + theme_bw()

GargeAreaRestricted <- ggplot(RestrictedLivingGarageArea) + geom_point(mapping = aes(x = GarageArea,
    y = SalePrice)) + xlab("Garage Area") + geom_smooth(mapping = aes(x = GarageArea,
    y = SalePrice), se = T) + theme_bw()

```

```

grid.arrange(LivingArea, GargeArea, LivingAreaRestricted, GargeAreaRestricted,
  ncol = 2)

RestGarageVsLivArea <- ggplot(RestrictedLivingGarageArea) + geom_point(mapping = aes(x = GarageArea,
  y = GrLivArea)) + xlab("Garage Area") + geom_smooth(mapping = aes(x = GarageArea,
  y = GrLivArea), se = T) + theme_bw()

RestGarageVsLivArea <- RestGarageVsLivArea + annotate("text",
  x = 250, y = 3000, label = paste0("Correlation = ", round(cor(RestrictedLivingGarageArea$GarageArea,
  RestrictedLivingGarageArea$GrLivArea), 2)))

RestGarageVsBsmt <- ggplot(RestrictedLivingGarageArea) + geom_point(mapping = aes(x = GarageArea,
  y = TotalBsmtSF)) + xlab("Garage Area") + geom_smooth(mapping = aes(x = GarageArea,
  y = TotalBsmtSF), se = T) + theme_bw()

RestGarageVsBsmt <- RestGarageVsBsmt + annotate("text", x = 250,
  y = 3000, label = paste0("Correlation = ", round(cor(RestrictedLivingGarageArea$GarageArea,
  RestrictedLivingGarageArea$TotalBsmtSF), 2)))

grid.arrange(RestGarageVsLivArea, RestGarageVsBsmt, nrow = 2)

options(scipen = 0)
trainFiltered <- train %>% dplyr::filter(GrLivArea < 4000 & GarageArea <
  1000)

SLR_LivingArea <- lm(data = trainFiltered, SalePrice ~ GrLivArea)
print(summary(SLR_LivingArea), caption = "ANOVA Simple Linear Regression Above grade living area")

# Model diagnostics Tidy store of model results
RedDf <- broom::augment(SLR_LivingArea)
# Plot of residuals
layout(matrix(c(1, 2, 3, 3), 2, 2, byrow = TRUE))
hist(RedDf$.resid, main = "Histogram of residuals", xlab = "Residuals")
qqnorm(RedDf$.resid, title = "Normal Q-Q plot of residuals (Sale Price)")
qqline(RedDf$.resid)
plot(RedDf$GrLivArea, RedDf$.resid, main = "Residuals vs Living Area",
  xlab = "Above grade living area", ylab = "Residuals")
SLR_LivingArea_NoIntercept <- lm(data = trainFiltered, SalePrice ~
  GrLivArea + 0)
library(mosaic)
sanitycheck <- do.call(rbind, dfapply(SampleFrame, favstats,
  select = is.numeric))
sanitycheck$mean <- as.numeric(format(sanitycheck$mean, digits = 1,
  nsmall = 2))
sanitycheck$sd <- as.numeric(format(sanitycheck$sd, digits = 1,
  nsmall = 2))
knitr::kable(sanitycheck, caption = "Data sanity check for numeric variables")
sanitycheckcharacter <- select(SampleFrame, colnames(SampleFrame[1,
  sapply(SampleFrame, class) == "character"])))

library(purrr)
UniqueVals <- sanitycheckcharacter %>% map(unique)

```

```

# s <-
# data.frame(names(tst), sapply(tst, function(x){paste(x, collapse
# = ', ')}), row.names = NULL)
Counts <- data.frame(sapply(UniqueVals, length), do.call(rbind,
  dfapply(sanitycheckcharacter, length, select = is.character)),
  do.call(rbind, dfapply(sanitycheckcharacter, n_missing, select = is.character)),
  row.names = names(UniqueVals))
colnames(Counts) <- c("# Unique", "n", "missing")

knitr::kable(Counts, caption = "Data sanity check for nominal variables",
  align = c("l", "r", "r", "r"))

```