Predictions with Linear Regression and R

Prince John

21/12/2020

Data Exploration

This is an example of linear regression. I am trying to predict house prices.I first load the data. First let's load the necessary libraries

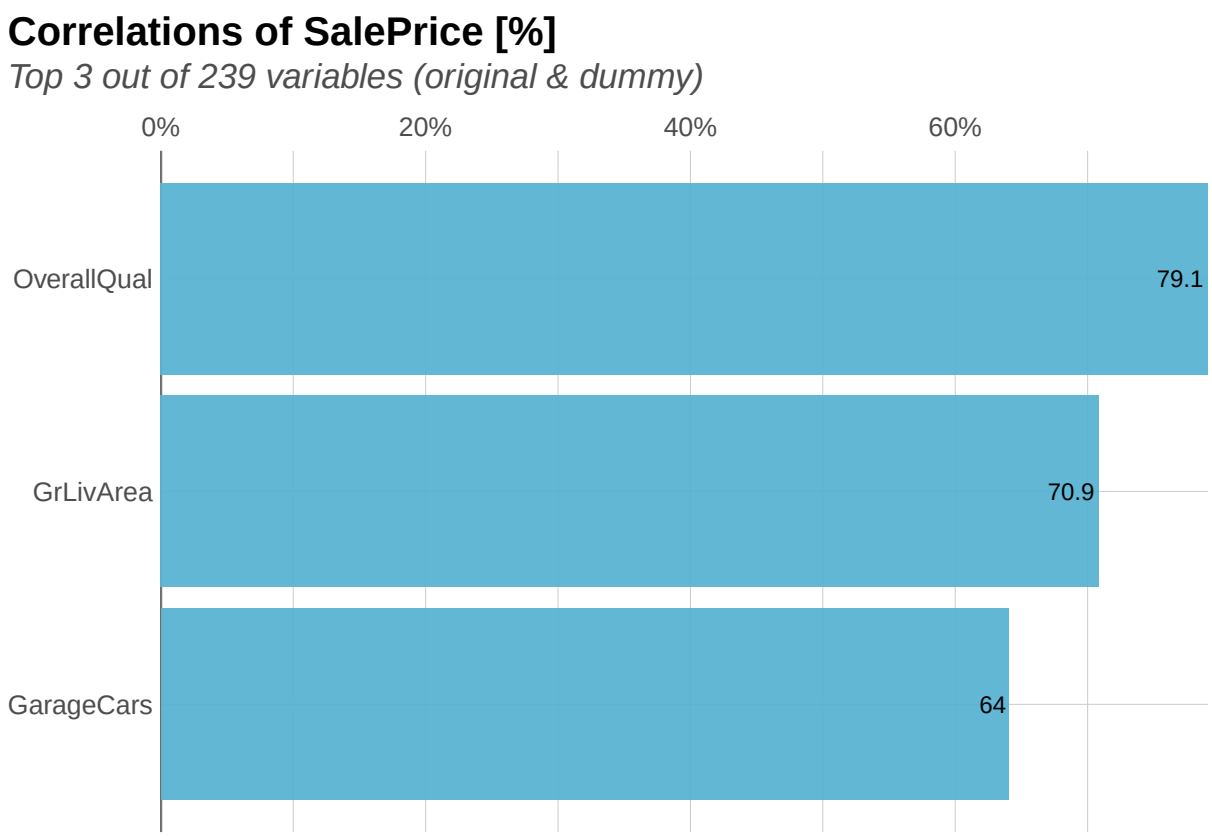
```
library(dplyr)
library(readr)
library(ggplot2)
library(lares)
library(modelr)
```

```
houseData<-read.csv('train.csv',header = T)
```

At the outset, I am interested in finding which of the variables have a positive association, or a strong positive association with the variable SalePrice. We use a package {lares} and try to find which

```
corr_var(houseData, # name of dataset
  SalePrice, # name of variable to focus on
  top = 3 # display top 5 correlations
)
```

```
## Warning in theme_lares2(pal = 2): Font Arial Narrow is not installed, has other
## name, or can't be found
```

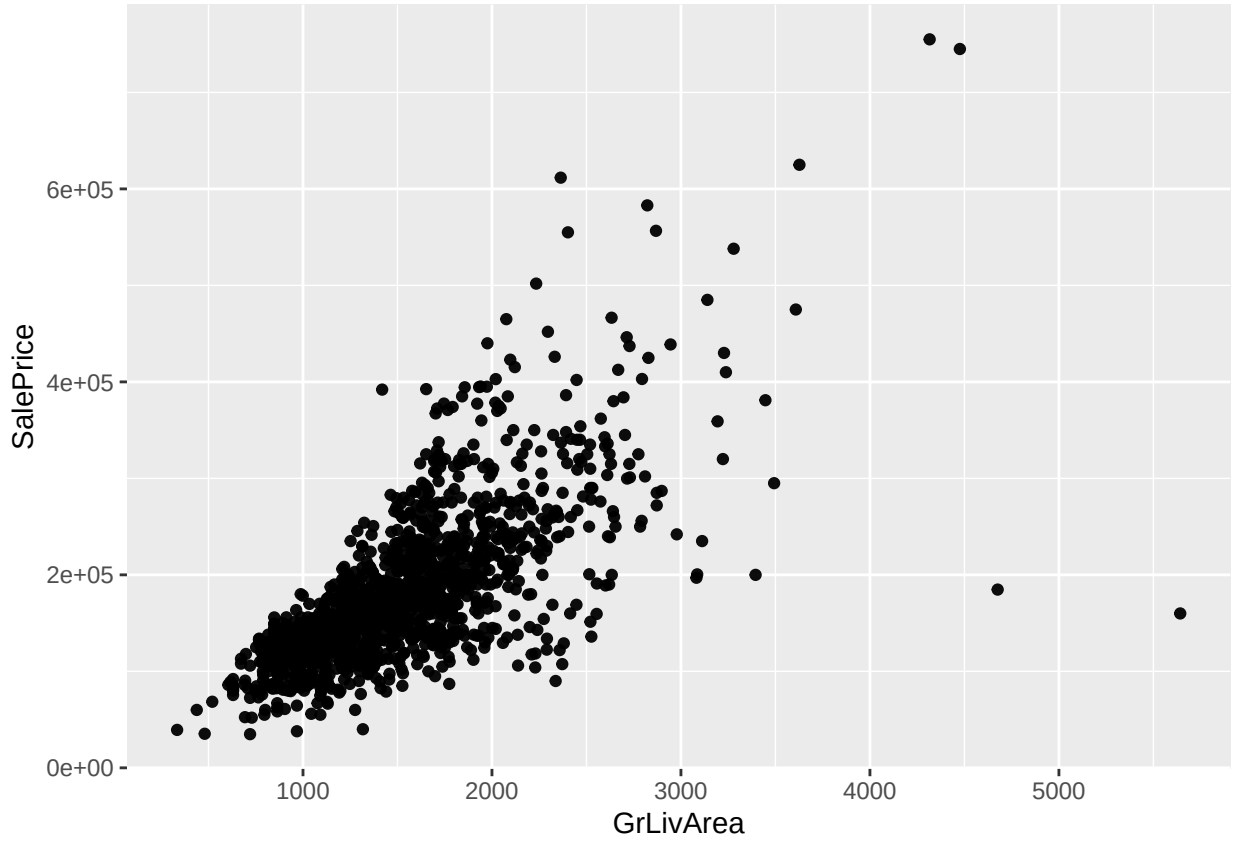


From the Correlogram, it seems that the OverallQual is most correlated to the SalePrice. However, since the OverallQual is discrete, we will not take that value, but rather the next one, which is GrLivArea.

Visualizing the data

We can visualize the data as follows to see how the data is correlated.

```
ggplot(houseData,aes(x=GrLivArea,y=SalePrice))+geom_point()
```




```
### Test and training data
```

```
data_sample<-sample(c(TRUE,FALSE),nrow(houseData),replace=T, prob=c(0.6,0.4))
train<-houseData[data_sample,]
test<-houseData[!data_sample,]
```

Linear regression

We are trying to fit our data to $y = a + bx$, where b is the slope of the regression line.

```
model=lm(SalePrice~GrLivArea,data=train)
```

Let us check the residual standard error (RSE) and the R^2 value. The RSE is an estimate of the standard deviation of the error of the model (error in our mathematical definition of linear regression). Roughly speaking, it is the average amount that the response will deviate from the true regression line.

```
sigma(model)
```

```
## [1] 58254.05
```

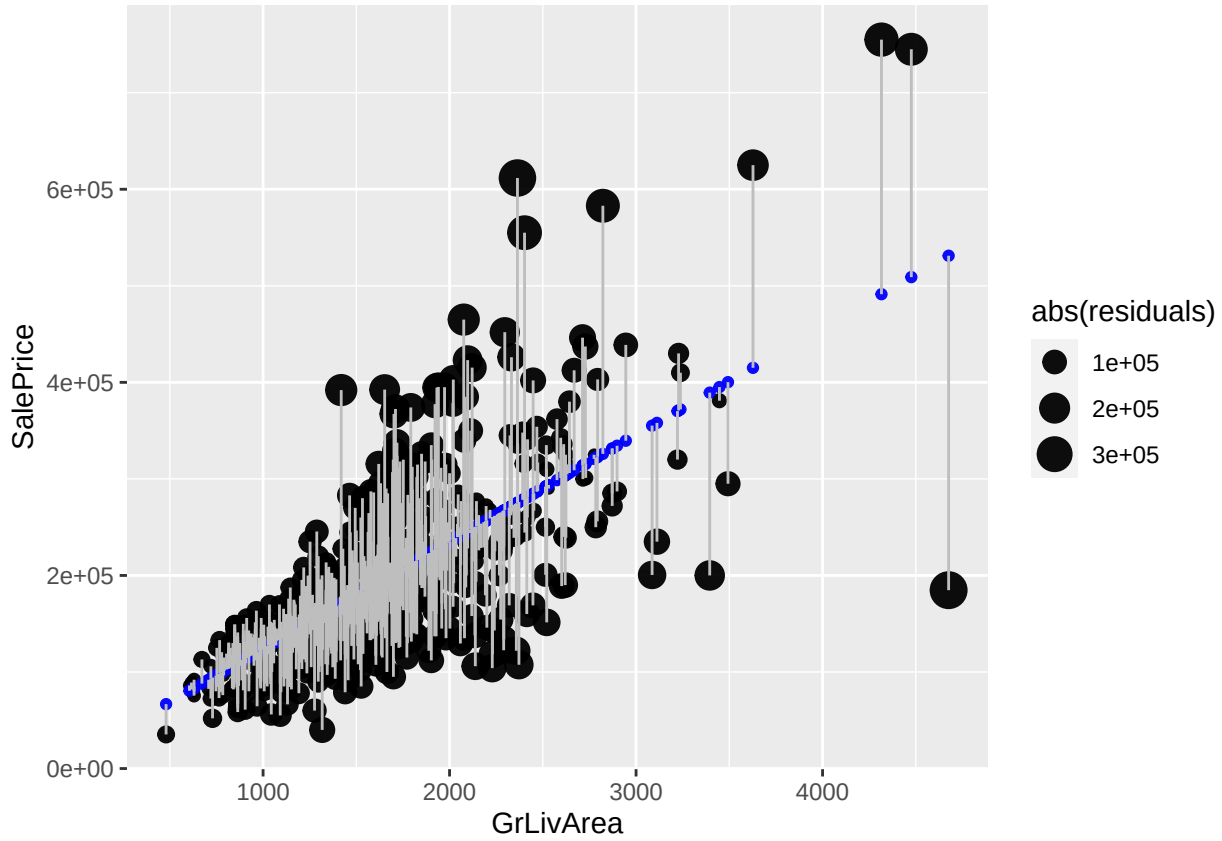
```
summary(model)$r.squared
```

```
## [1] 0.5135529
```

Now, its time to calculate the estimate and the residuals. They are as follows.

```
train$estimate<-predict(model)
train$residuals<-residuals(model)
ggplot(train,aes(GrLivArea,SalePrice))+
```

```
  geom_point(aes(size=abs(residuals)))+
  geom_point(aes(y=estimate),color='blue')+
  geom_segment(aes(xend=GrLivArea,yend=estimate),color='gray')
```



As you can clearly see, probably a linear relationship is not the best relationship. Neveertheless, we will continue with this. Let's look at it closely to see where it deviates from a linear relationship using the LOESS function.

```
ggplot(train,aes(GrLivArea,SalePrice))+geom_point()+
  geom_smooth(method='lm')+
  geom_smooth(se=FALSE,color='red')
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

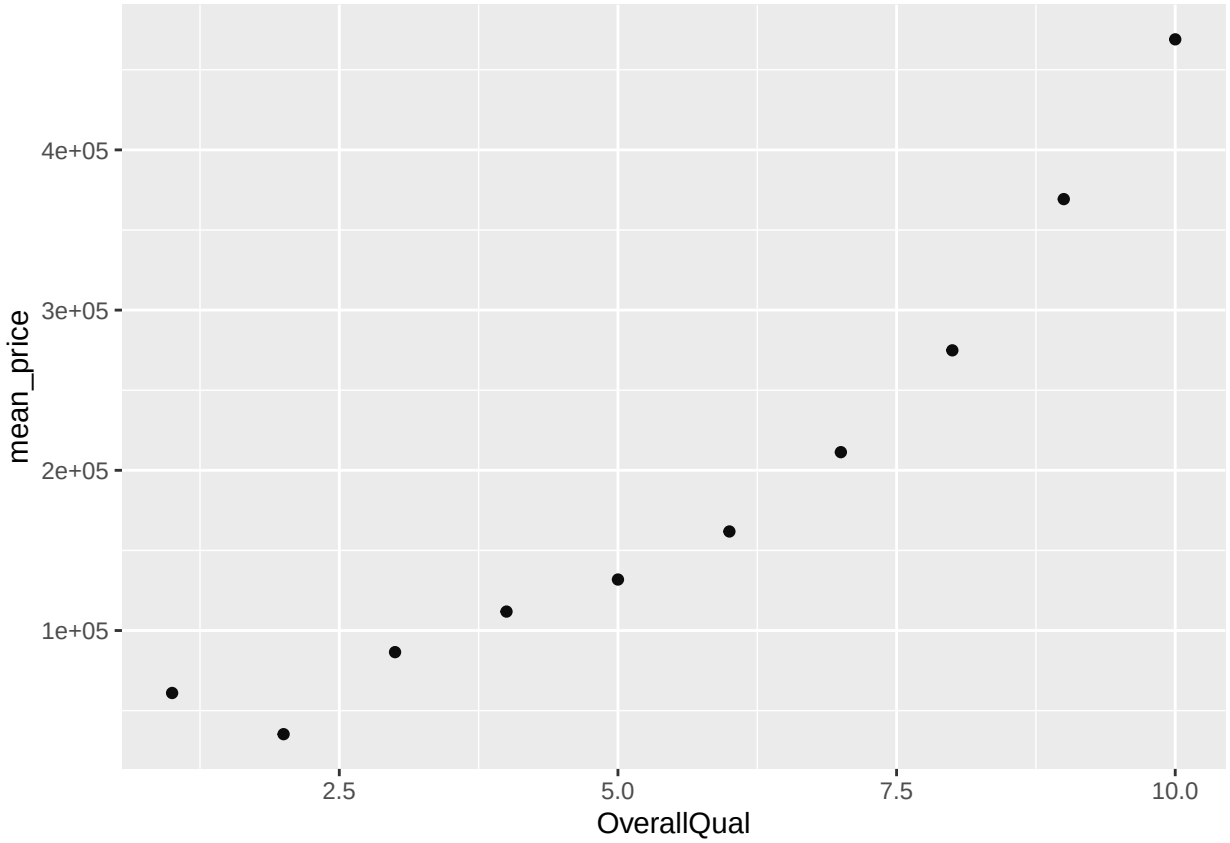
Clearly, it deviates from a relationship past GrLivArea value above 3000.

Mean SalePrice by OverallQual

Assessing the Model Results

```
train%>%group_by(OverallQual)%>%summarise(mean_price=mean(SalePrice))%>%ggplot(aes(OverallQual,mean_pri
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```




```
summary(model)
```

```
##
## Call:
## lm(formula = SalePrice ~ GrLivArea, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -346403  -30648    -413   22658  336412
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13581.459   5973.790   2.274   0.0232 *
## GrLivArea     110.687      3.691  29.991   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58250 on 852 degrees of freedom
## Multiple R-squared:  0.5136, Adjusted R-squared:  0.513
## F-statistic: 899.5 on 1 and 852 DF,  p-value: < 2.2e-16
```

```
sprintf("The results show that the intercept is %s and slope is %s. This means that when the GrLivArea
```

```
## [1] "The results show that the intercept is 13581.4589040586 and slope is 110.686774965761. This mea
```

The column adjacent to Estimate is called Std. Error; the standard error of each coefficient is the estimate of the standard deviation of the coefficient. The t value and $P(>|t|)$ inherently answer the same question-- given the value of our variable's regression

coefficient and its' standard error, does the variable explain a significant part of the change in our outcome variable? However, the $P(>|t|)$ column purposely provides a more concise response to this question, using the asterisk notation that corresponds with the Signif. codes legend at the bottom of the Coefficients results section. In R model output, one asterisk means $(p < .05)$. Two asterisks mean $(p < .01)$; and three asterisks mean $(p < .001)$. These values are referred to as p-values in scientific literature. How can we use p-values to answer our question around model significance?

Asterisks in a regression table indicate the level of the statistical significance of a regression coefficient. Our understanding of statistical significance is based off of the idea of a random sample.

Making predictions

```
test %>%
  add_predictions(model) %>%
  ggplot(aes(GrLivArea, SalePrice)) +
  geom_point() +
geom_point(aes(y = pred), color = "blue")
```