MA575 Deliverable2

Group 4

2024-02-29

Setting Up

```
# Read updated dataset
bmw_data <- read.csv("/Users/jieminyang/Documents/08.BU/Academics/MA575/Project/Proje
cts/BMWpricing_updated.csv", header=TRUE, as.is=TRUE)
# create summary
summary(bmw_data)
```

```
##   maker_key          model_key            mileage          engine_power
##  Length:4843        Length:4843        Min.   :     -64   Min.   :  0
##  Class :character   Class :character   1st Qu.: 102914   1st Qu.:100
##  Mode  :character   Mode  :character   Median : 141080   Median :120
##                                        Mean   : 140963   Mean   :129
##                                        3rd Qu.: 175196   3rd Qu.:135
##                                        Max.   :1000376   Max.   :423
##  registration_date      fuel           paint_color          car_type
##  Length:4843        Length:4843        Length:4843        Length:4843
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  feature_1       feature_2       feature_3       feature_4
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:2181      FALSE:1004      FALSE:3865      FALSE:3881
##  TRUE :2662      TRUE :3839      TRUE :978       TRUE :962
##
##
##
##  feature_5       feature_6       feature_7       feature_8
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:2613      FALSE:3674      FALSE:329       FALSE:2223
##  TRUE :2230      TRUE :1169      TRUE :4514      TRUE :2620
##
##
##
##      price          sold_at            obs_type
##  Min.   :   100   Length:4843        Length:4843
##  1st Qu.: 10800   Class :character   Class :character
##  Median : 14200   Mode  :character   Mode  :character
##  Mean   : 15828
##  3rd Qu.: 18600
##  Max.   :178500
```

Data exploration

```r
# Create the Age Variable (year sold - year register)
sold_at_split <- strsplit(bmw_data$sold_at, "/")

registration_split <- strsplit(bmw_data$registration_date, "/")

bmw_data$month_sold <- sapply(sold_at_split, function(x) as.integer(x[1]))

bmw_data$year_sold <- sapply(sold_at_split, function(x) as.integer(x[3]))

bmw_data$month_registered <- sapply(registration_split, function(x) as.integer(x[1]))

bmw_data$year_registered <- sapply(registration_split, function(x) as.integer(x[3]))

price <- bmw_data$price # our y variable
bmw_data$age <- bmw_data$year_sold-bmw_data$year_registered + (1/12)*(bmw_data$month_
sold - bmw_data$month_registered) # our x variable
age <- bmw_data$age

length(price)
```

```
## [1] 4843
```

```r
length(age)
```

```
## [1] 4843
```

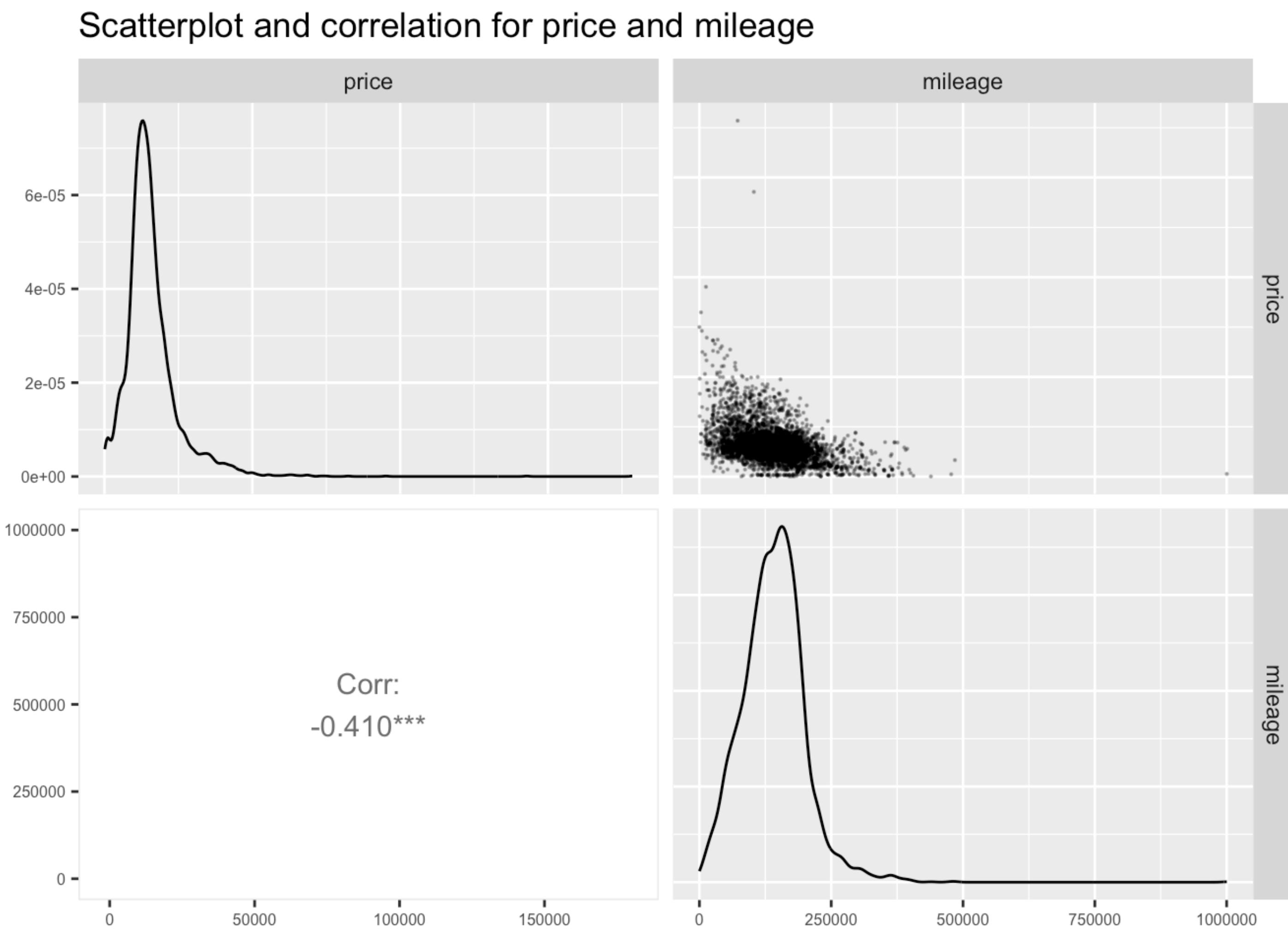
```r
#prepare predictors and response
response <- "price"
predictors <- c("mileage", "fuel", "paint_color", "car_type", "feature_1", "feature_2
", "feature_3", "feature_4", "feature_5", "feature_6", "feature_7", "feature_8", "yea
r_registered", "month_registered")
bmw_subset <- subset(bmw_data, select = c(response, predictors))
```

To get the idea of how each predictors are distributed, we are creating pairwise plots to visualize one on one correlation with car price and the distribution of the predictors.

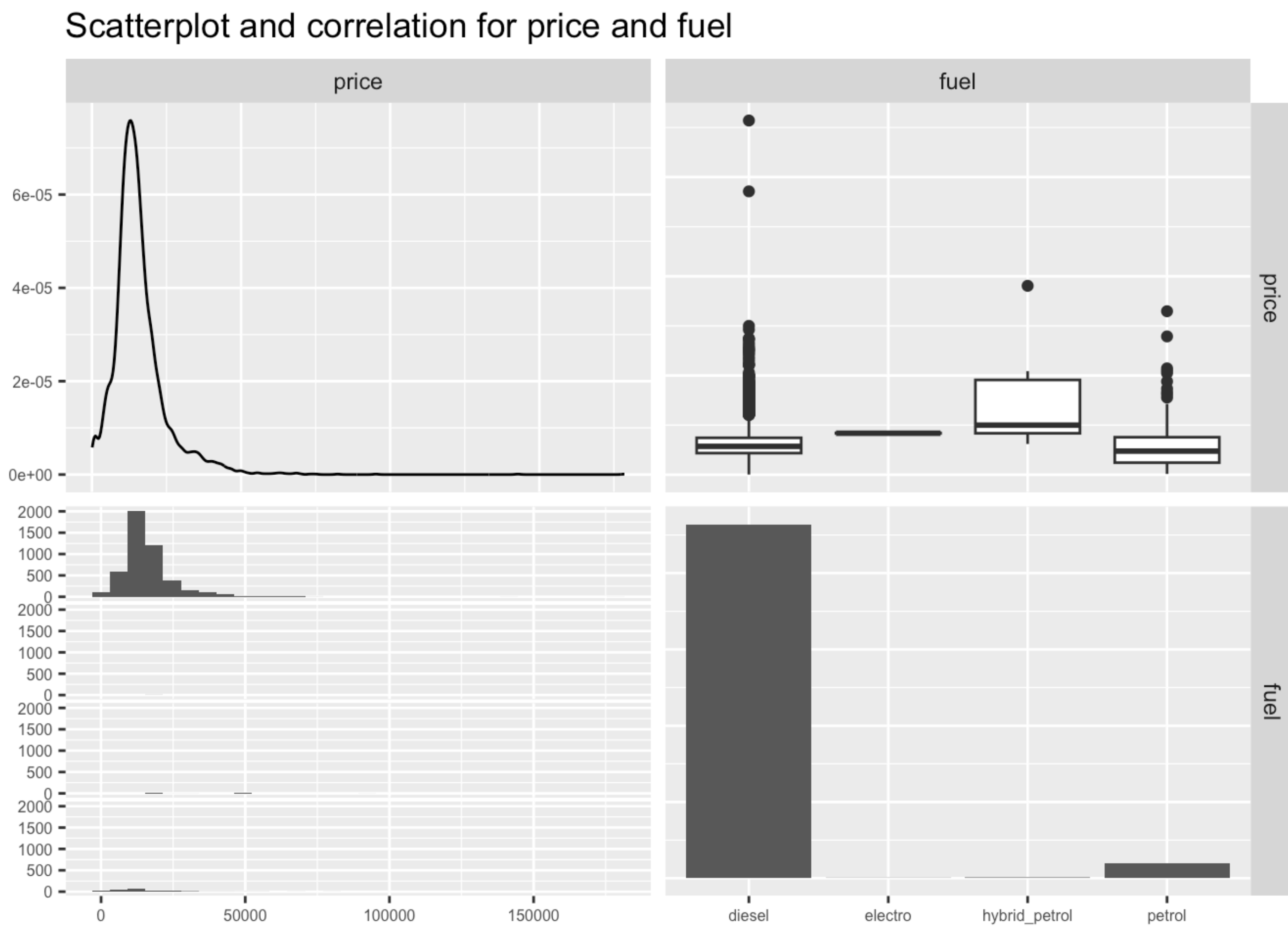
```
for (predictor in predictors) {
  # Subset the dataframe for the current predictor variable
  plot_data <- subset(bmw_subset, select = c(response, predictor))

  # Create a scatterplot/correlation graph
  g <- ggpairs(plot_data,
        title = paste("Scatterplot and correlation for price and", predictor),
        upper = list(continuous = wrap("points", alpha=0.3, size=0.1)),
        lower = list(continuous = "cor", method = "spearman")) + theme(axis.text = el
ement_text(size = 6))

  # Print the graph
  print(g)
}
```

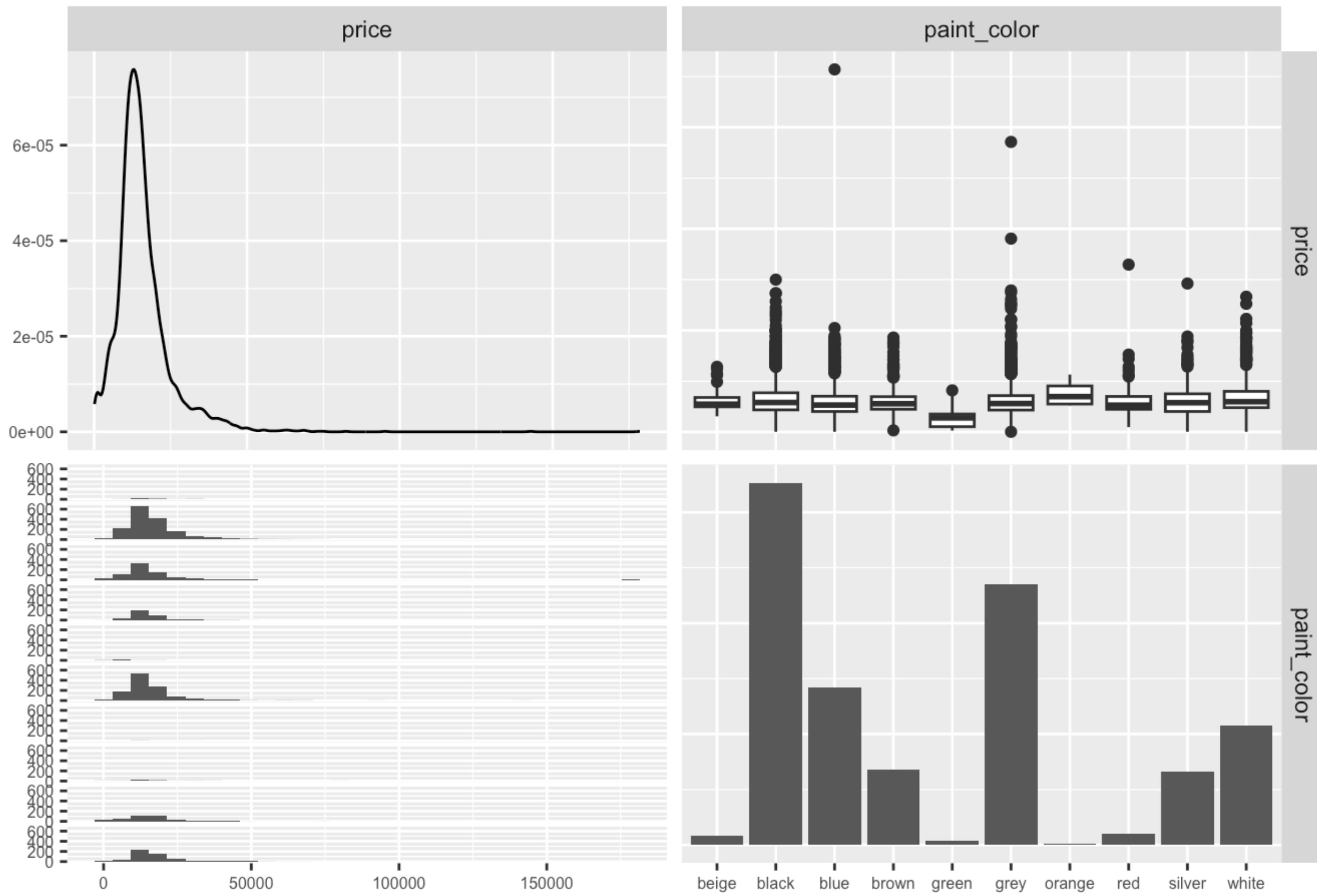



```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



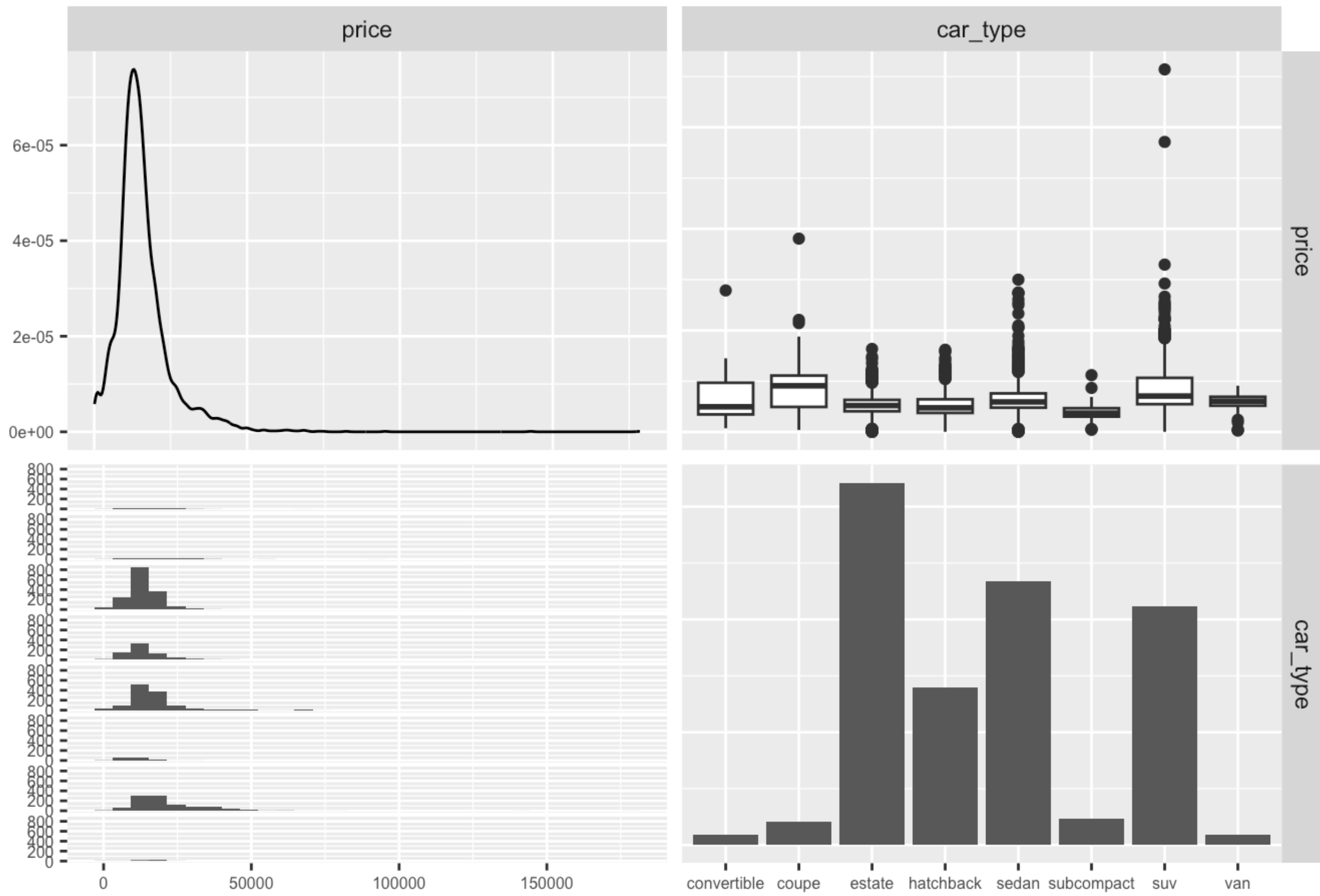

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and paint_color



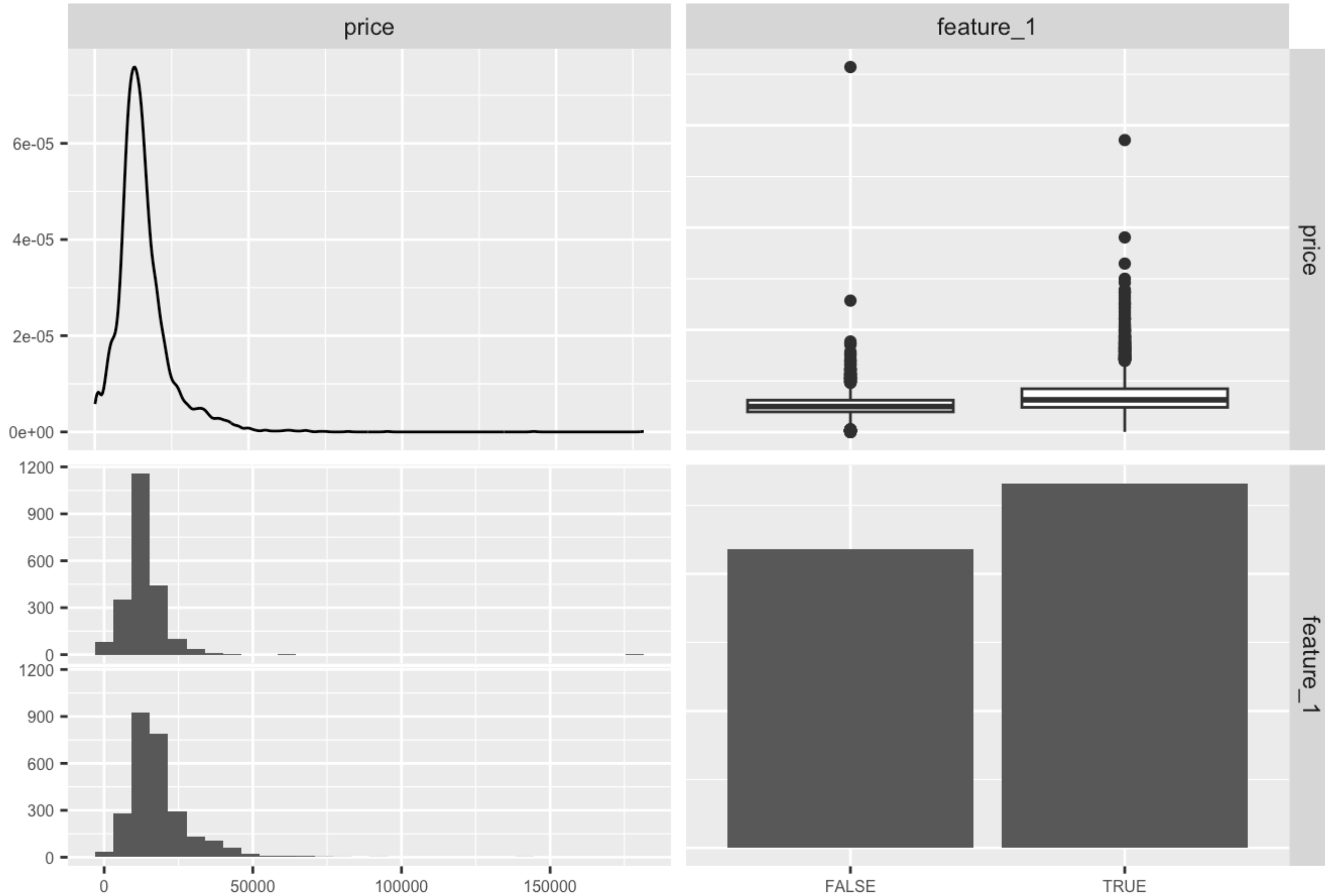

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and car_type



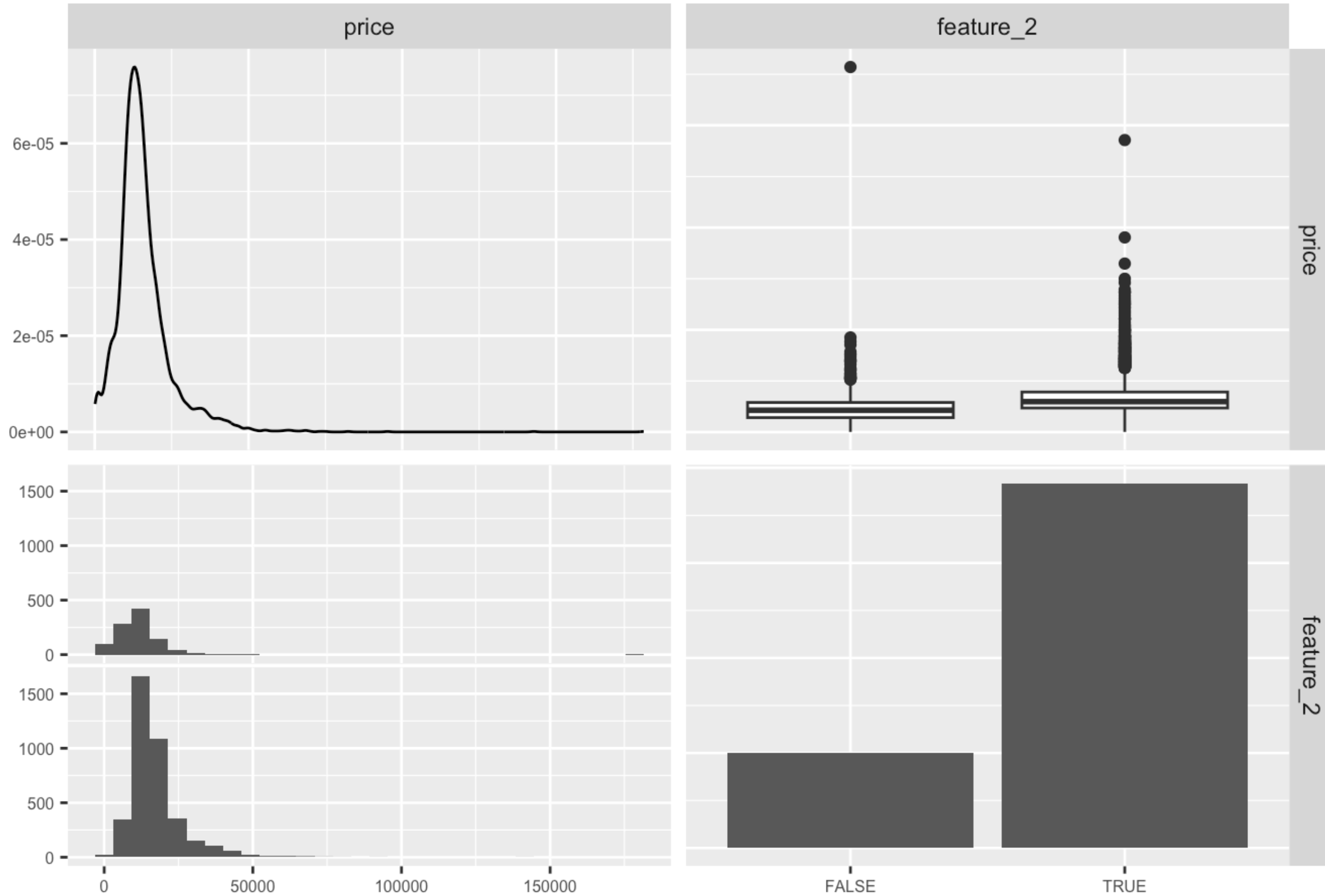

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_1



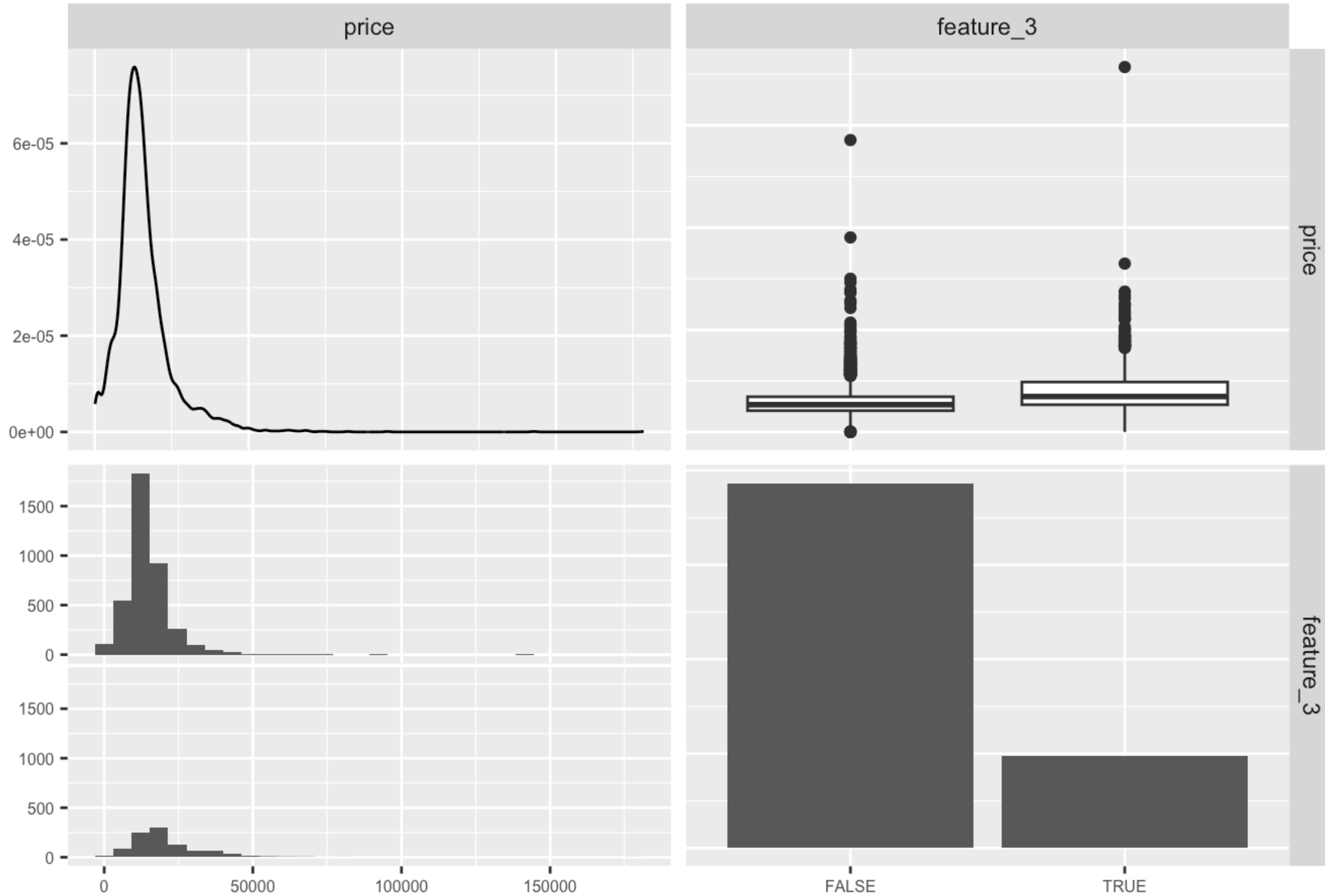

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_2



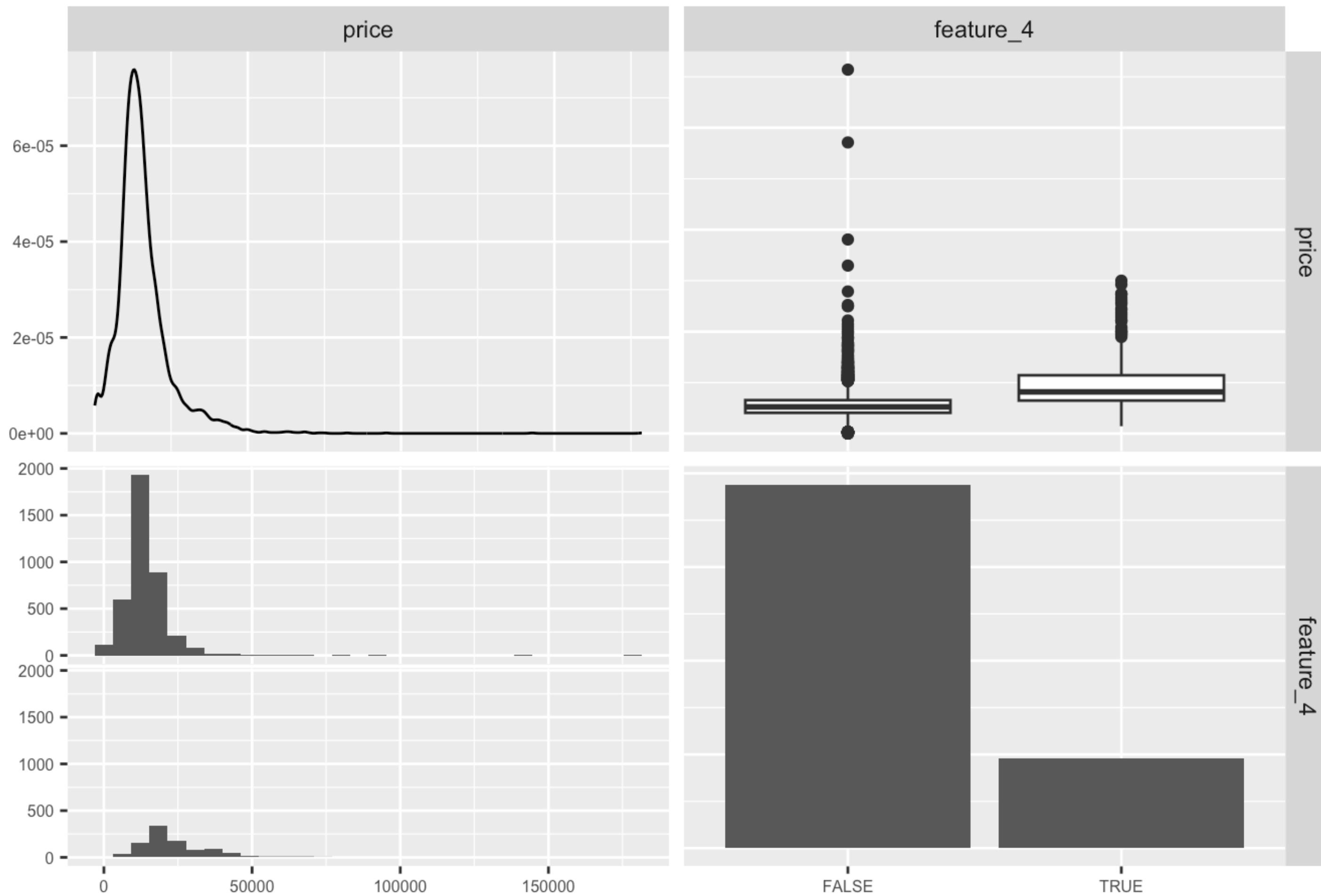

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_3



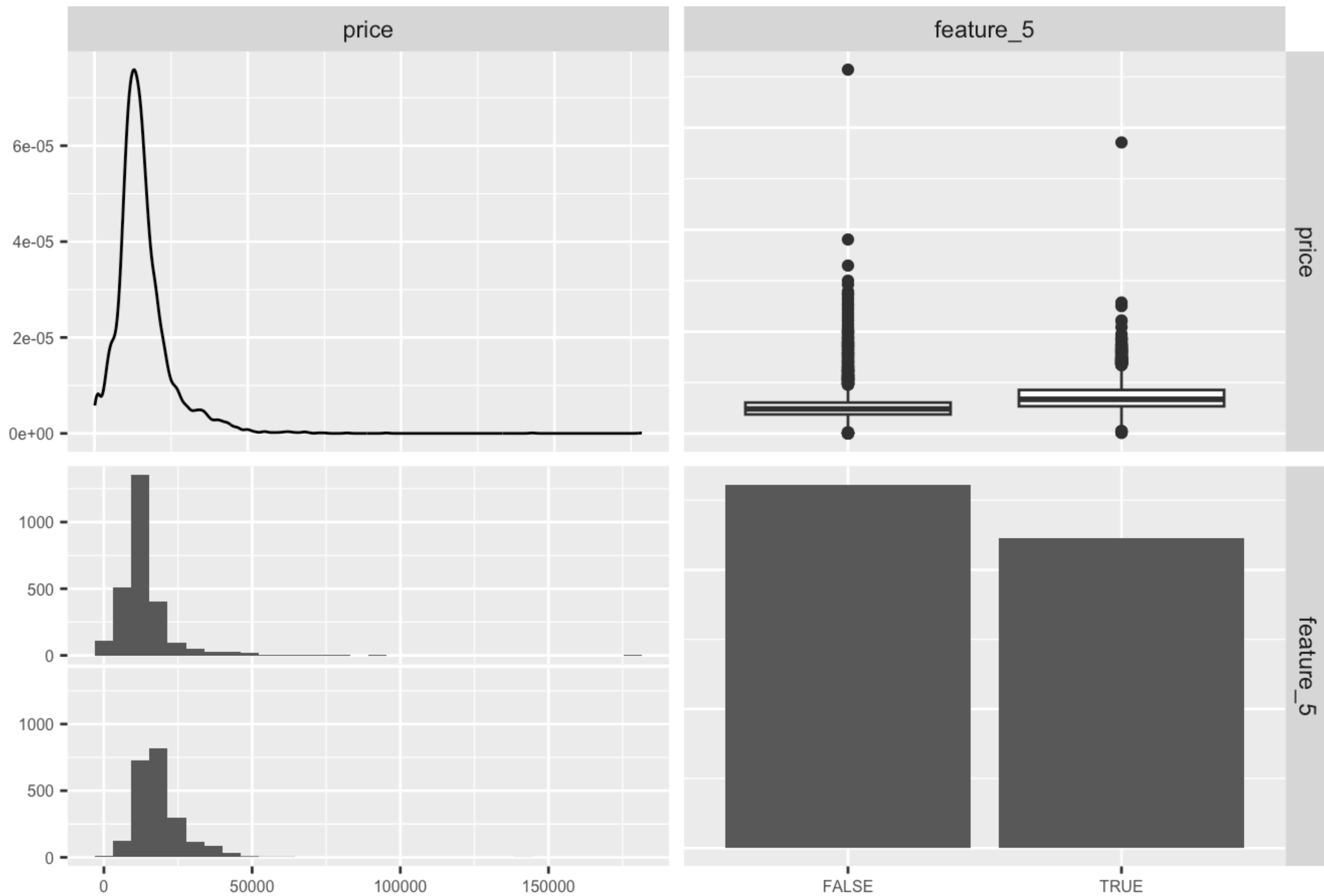

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_4



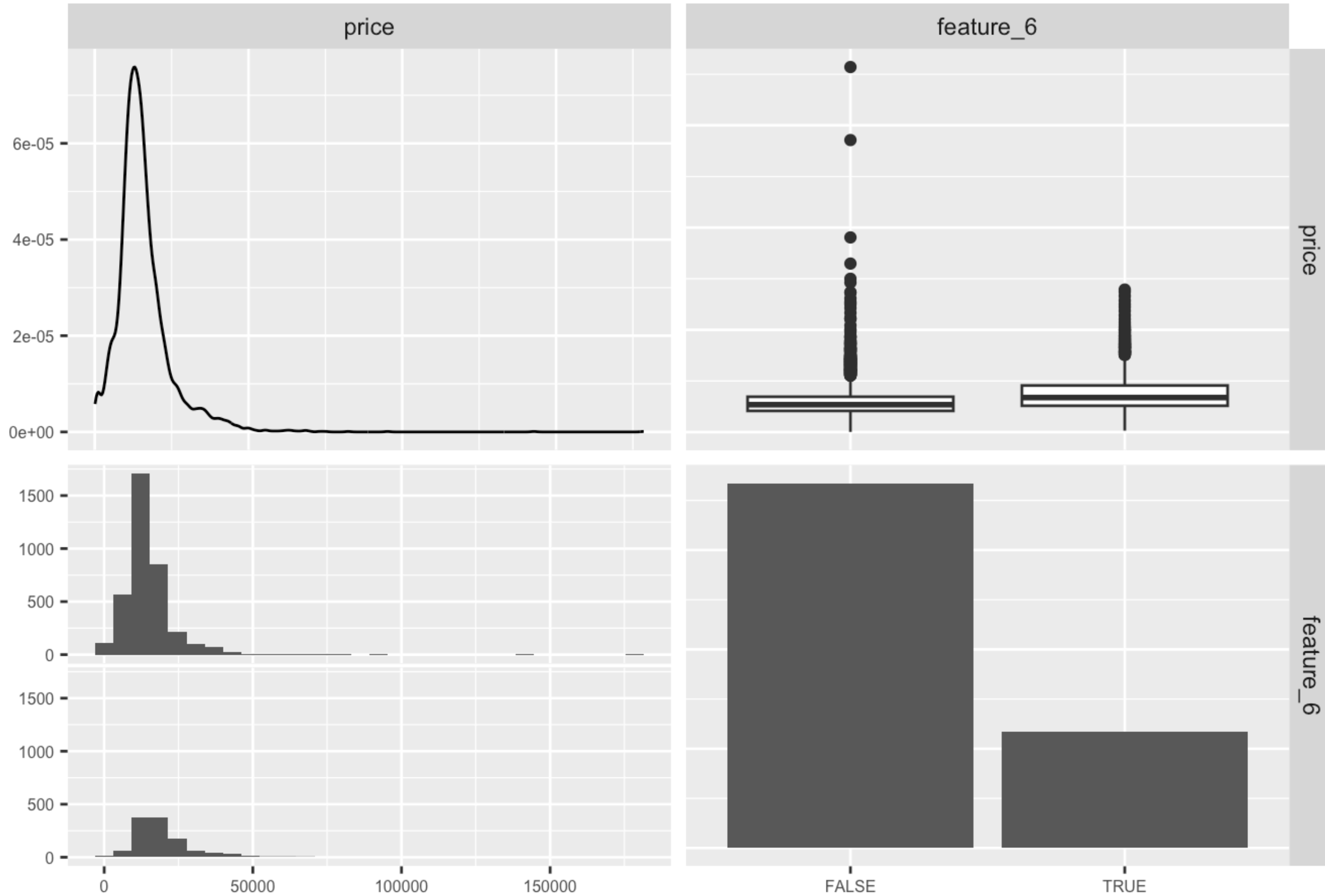

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_5



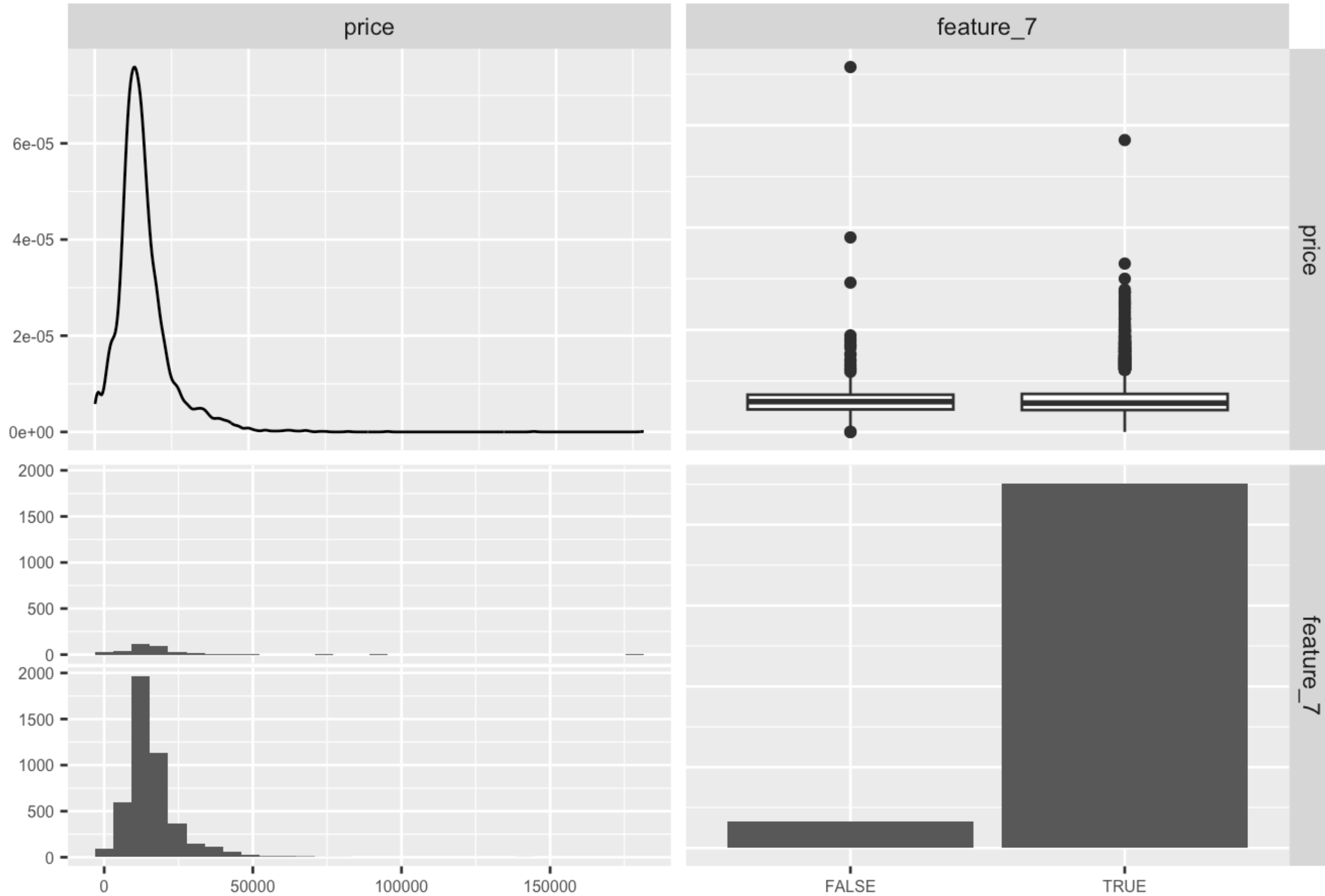

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_6



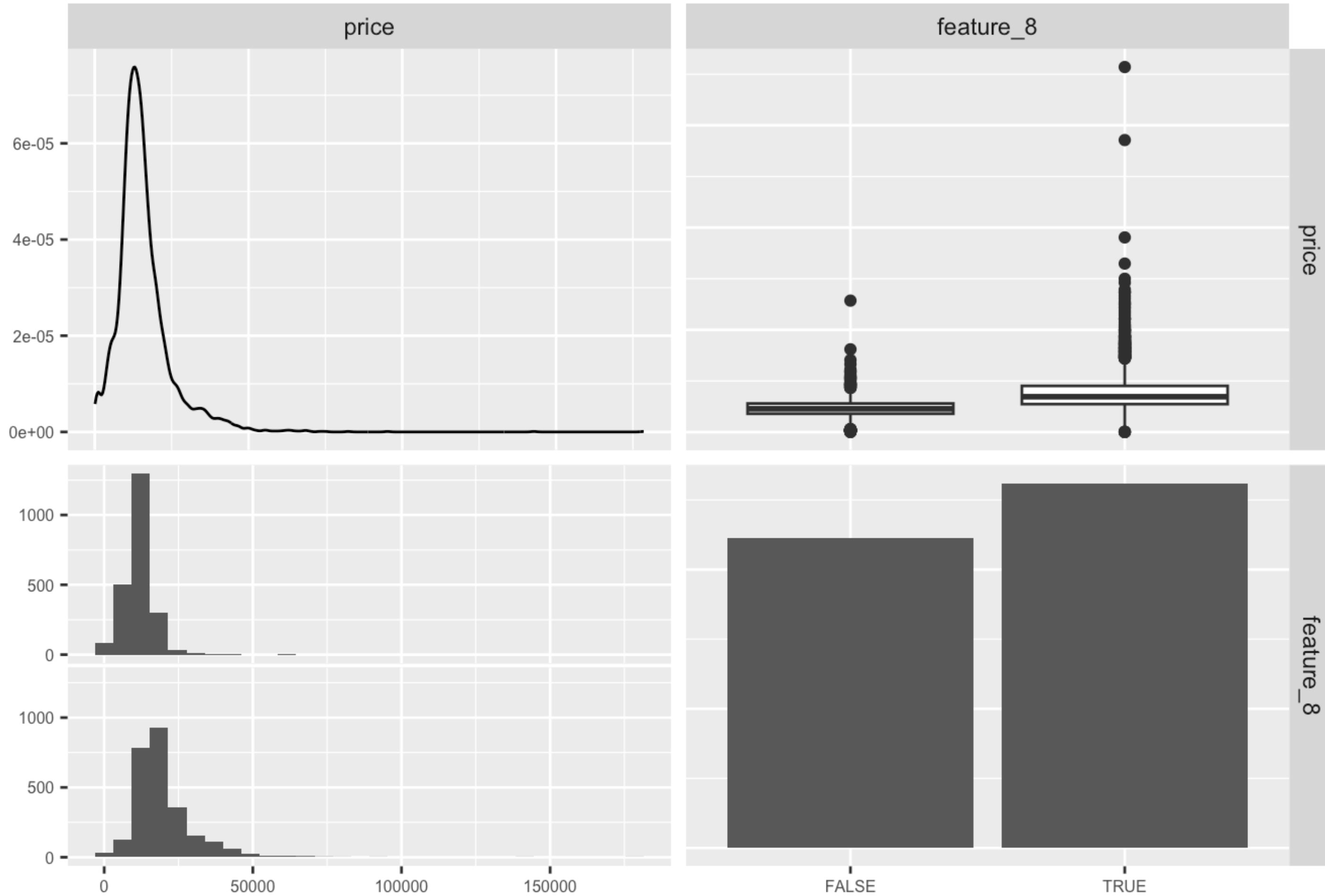

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_7

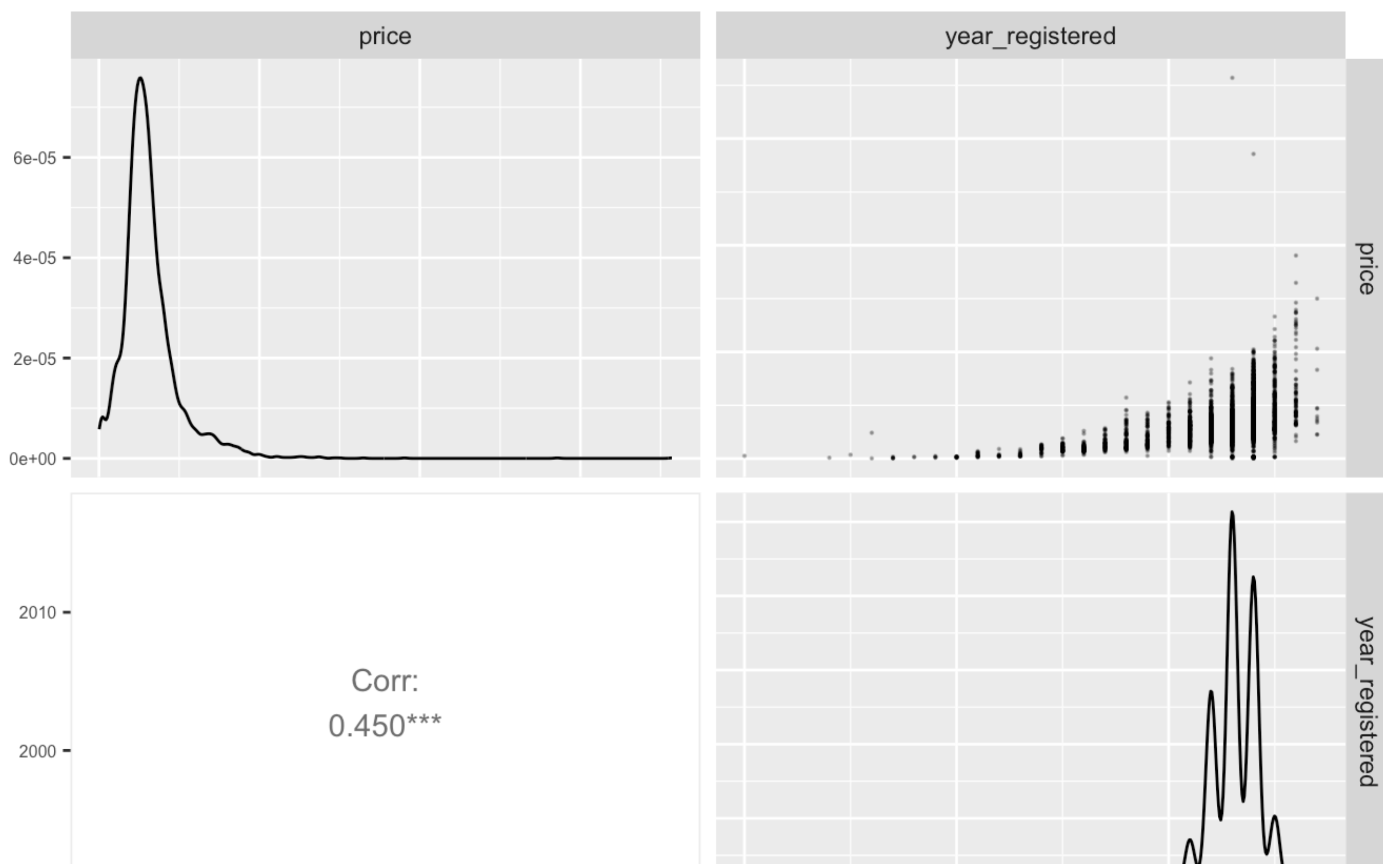


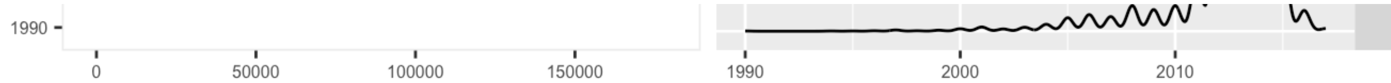

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Scatterplot and correlation for price and feature_8

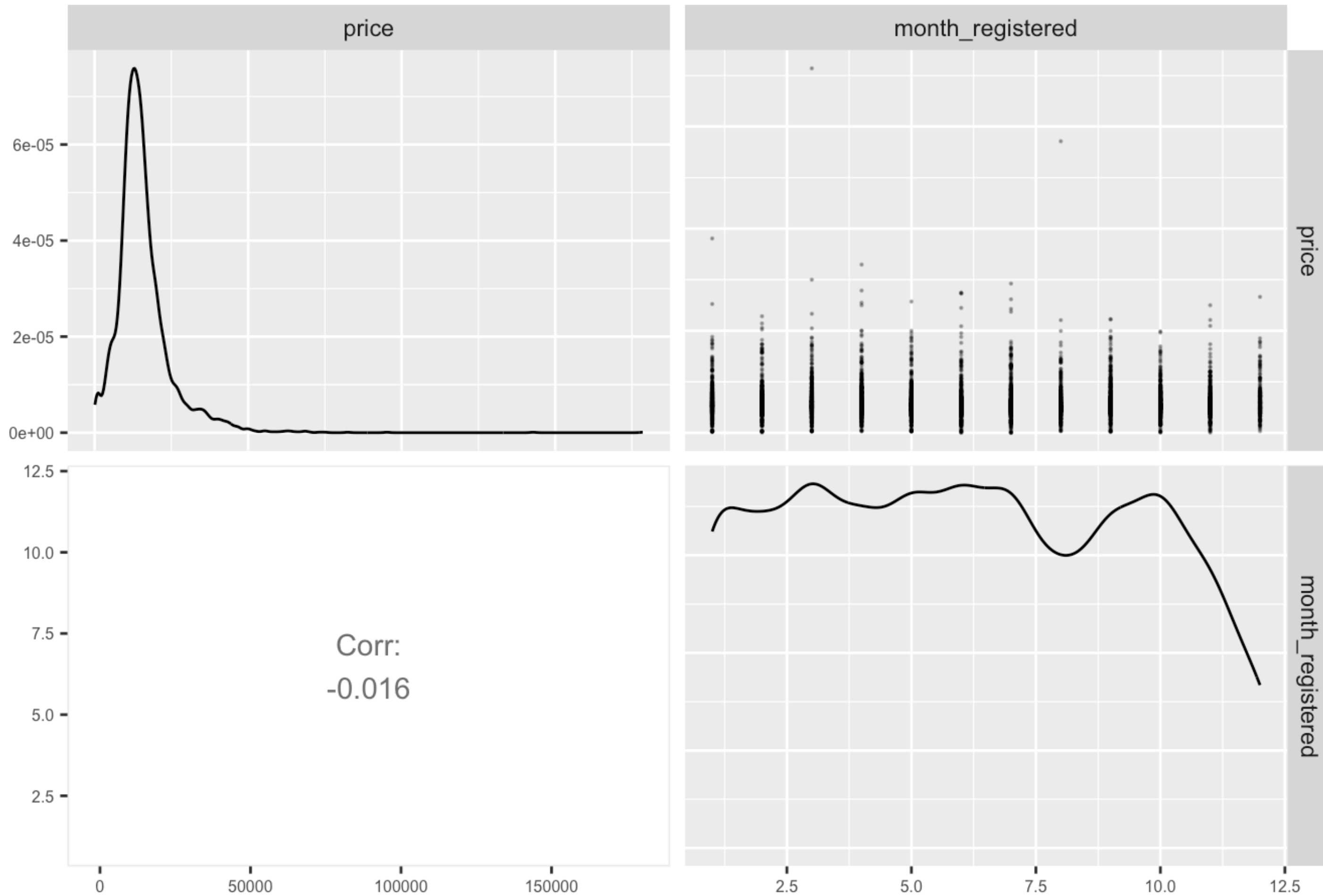


Scatterplot and correlation for price and year_registered



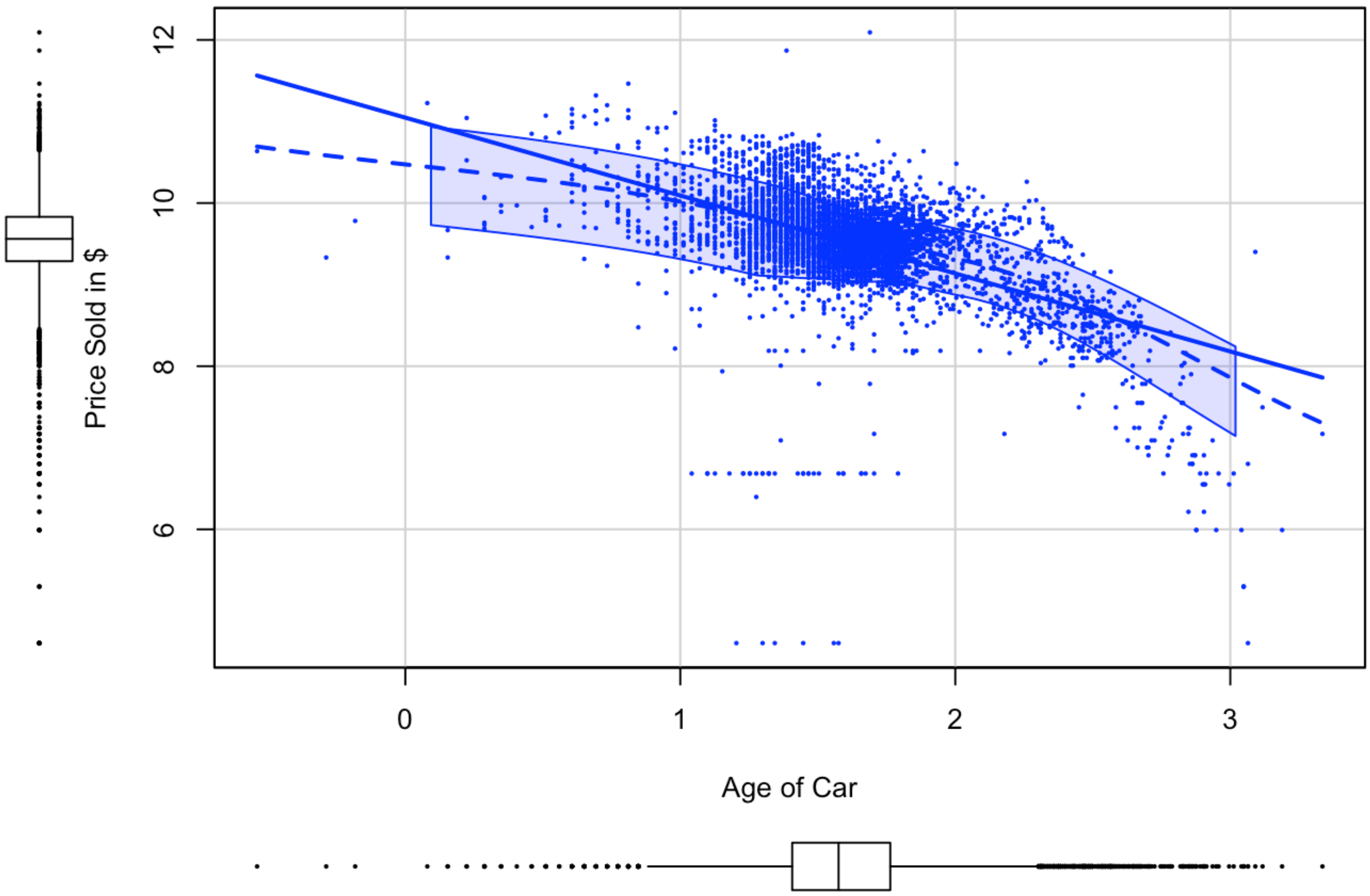


Scatterplot and correlation for price and month_registered

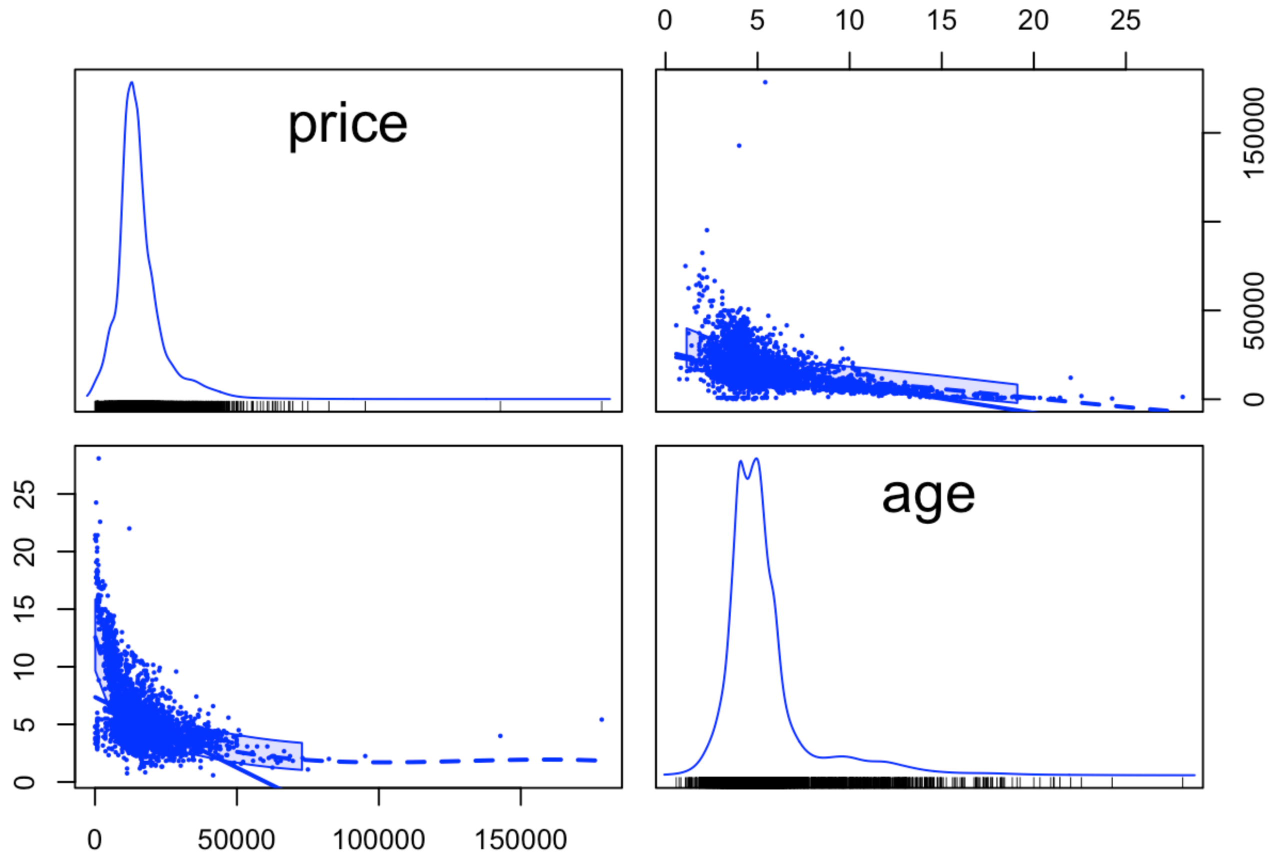


We decide to use Age as the predictor of our simple linear regression. It shows car price and car age are right skewed, with few extreme values at the tail.

```
# check the distribution of 2 variables
scatterplot(log(age), log(price),
     ylab="Price Sold in $", xlab="Age of Car",
     pch=19, cex=0.2)
```




```
# boxplot shows both variable is not normally distributed, scatterplot detects extrem
e outliers
scatterplotMatrix( ~ price + age, pch=19, cex=0.2)
```



Model Fitting and Diagnostics

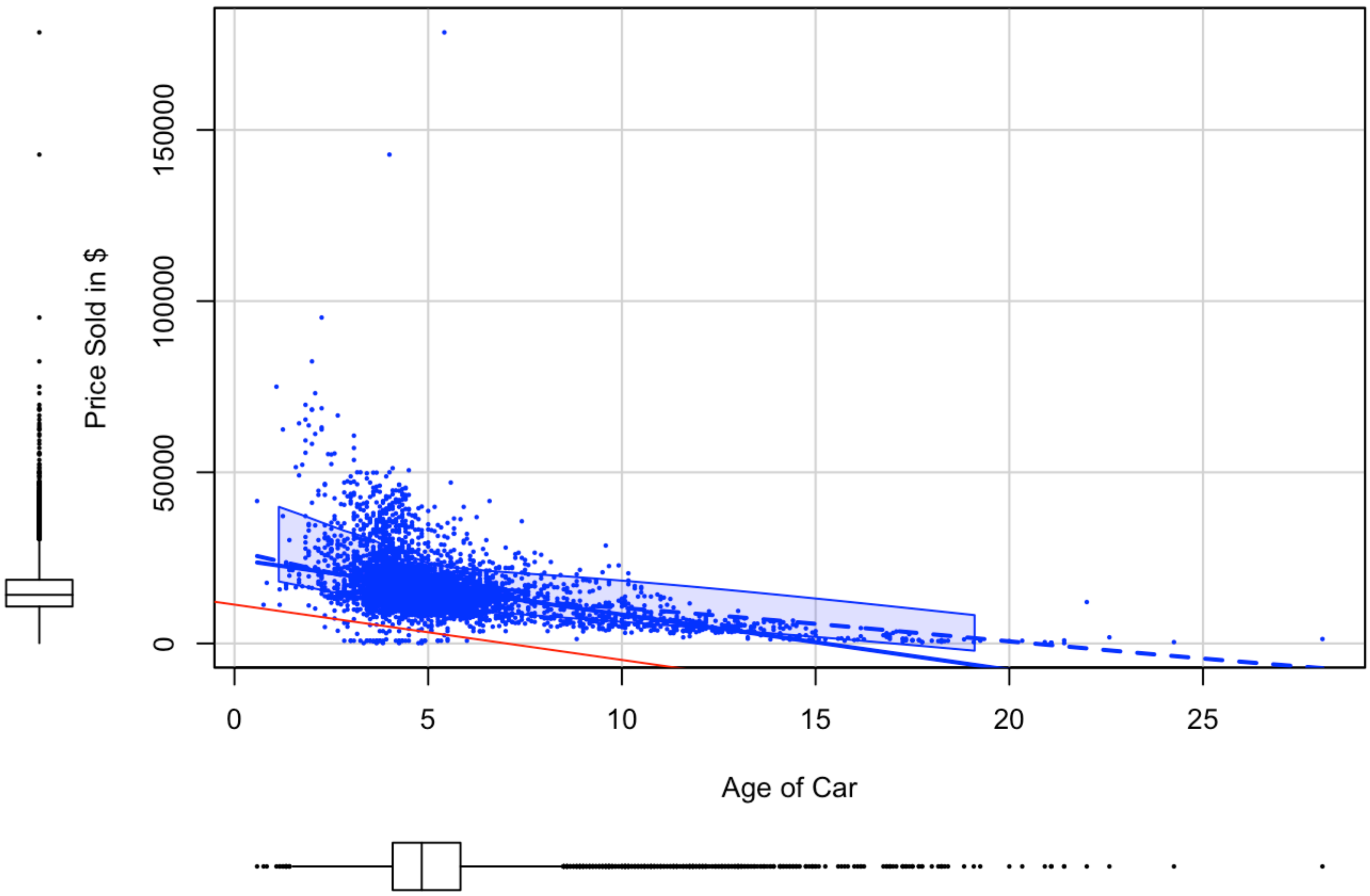
Model 1

Our first model is simply using age as x variable and price as y variable. We set this as the benchmark predicability. R^2 is 0.1987 which means approximately 19.87% of variations in price is explained by the model. The NQQ plot shows great deviation from normal quantile, indicating data on the upper tail is highly skewed. A U-shape pattern is identified in the SR plot, indicating non-constant variance. Several points with high leverage is observed, but no potential bad leverage point detected, since all leverage points lies in the Cook distance. The model need improvements on the above mentioned issues.

```
m1 <- lm(price~age)
summary(m1)
```

```
##
## Call:
## lm(formula = price ~ age)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19229  -4763  -1643   2408 162647
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24608.65     280.26   87.81   <2e-16 ***
## age         -1616.40      46.74  -34.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8258 on 4841 degrees of freedom
## Multiple R-squared:  0.1981, Adjusted R-squared:  0.1979
## F-statistic:  1196 on 1 and 4841 DF,  p-value: < 2.2e-16
```

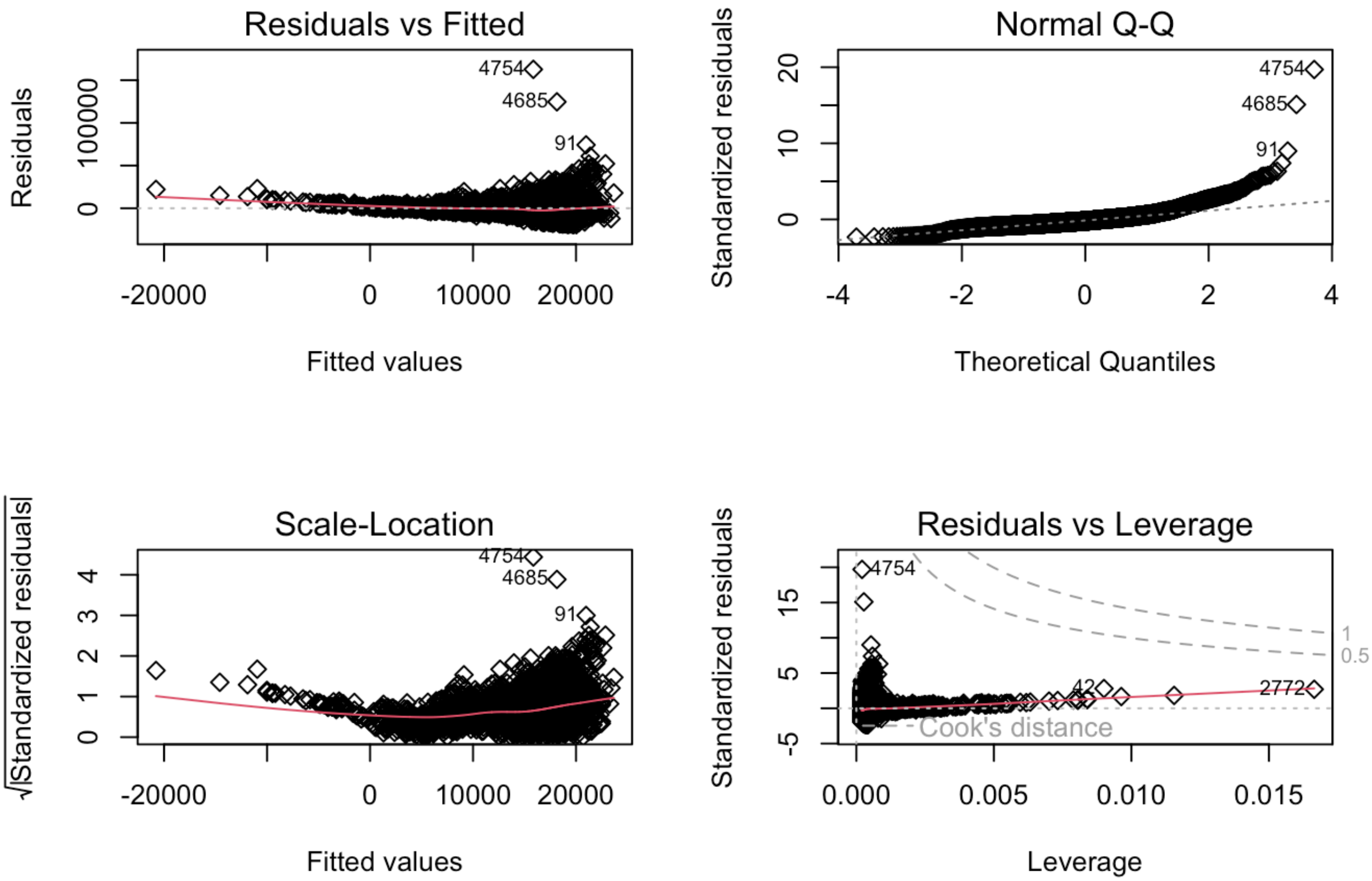
```
scatterplot(age, price,
     ylab="Price Sold in $", xlab="Age of Car",
     pch=19, cex=0.2)
abline(m1, col = 'red')
```




```
summary(m1)$r.squared
```

```
## [1] 0.1981027
```

```
par(mfrow = c(2,2))
plot(m1, cex = 1, pch = 5)
```



Cheking the outlier Here we use leverage score and z-score of residual to find potential outliers and bad leverage points, and filter them out. After cleaning, the clustering problem in the data is solved (shown on the next graph), and the not skewed distributed variables become nearly normal. We lost 411 data points through cleaning.

```
# make a new dataframe for cleaning outlier
model_data <- data.frame(age = age, price = bmw_data$price)
# Use leverage to check the outlier on age of cars
lev <- hatvalues(m1)
model_data$filter1 <- lev <= (4/length(age))
# use z-score for residuals to check the outliers for age of cars
resid = residuals(m1)
z_resid = (resid - mean(resid))/sd(resid)
model_data$filter2 <- z_resid > (-3) & z_resid < 3
cleaned_data <- model_data[model_data$filter1 != FALSE & model_data$filter2 != FALSE,
]
cleaned_data <- cleaned_data[, -3:-4]
```

```
summary(cleaned_data) # almost normally distributed
```

```
##       age            price      
##  Min.   :1.167   Min.   :  100  
##  1st Qu.:4.000   1st Qu.:11400  
##  Median :4.750   Median :14500  
##  Mean   :4.909   Mean   :15951  
##  3rd Qu.:5.583   3rd Qu.:18800  
##  Max.   :9.750   Max.   :44600
```

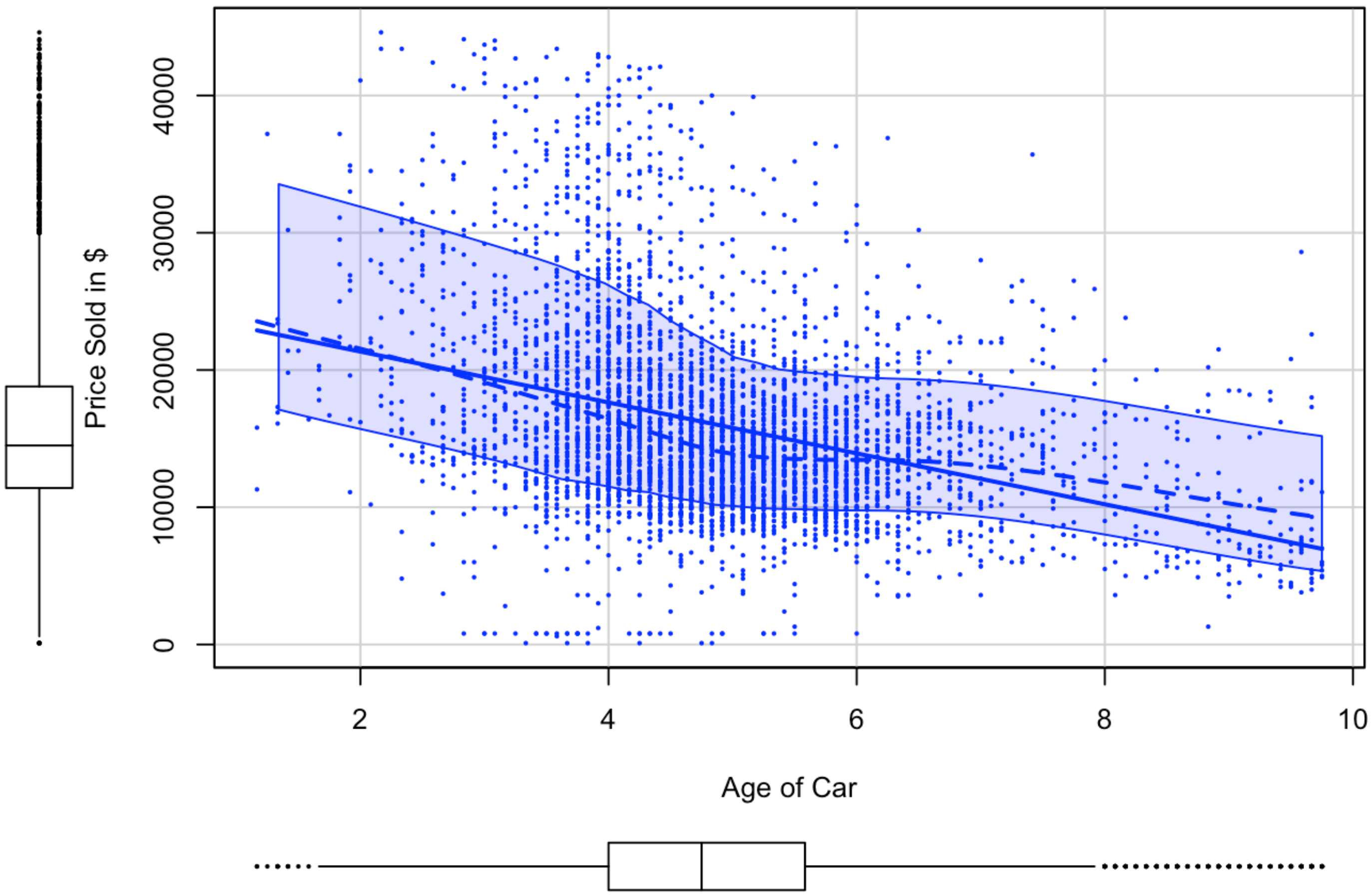
```
summary(cleaned_data$age)# almost normally distributed
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.167   4.000   4.750   4.909   5.583   9.750
```

```
summary(cleaned_data$price)# almost normally distributed
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     100   11400   14500   15951   18800   44600
```

```
# plot the cleaned data again
scatterplot(cleaned_data$age, cleaned_data$price,
     ylab="Price Sold in $", xlab="Age of Car",
     pch=19, cex=0.2)
```



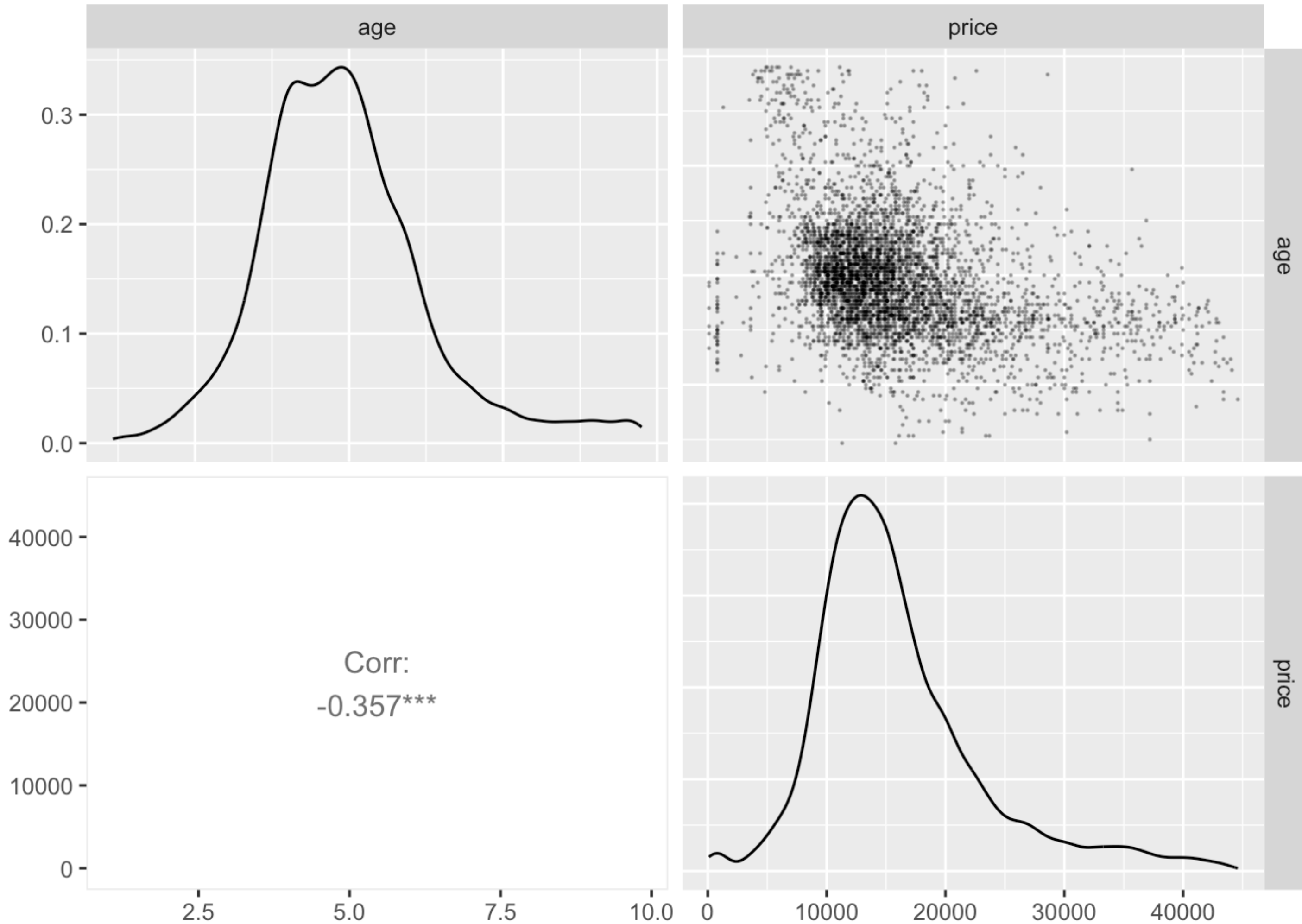

```
length(cleaned_data$age)-length(age)
```

```
## [1] -411
```

Model 2

Our second experiment model is age vs price after cleaning. The r^2 is 0.1273, which is significantly lower than the model without cleaning. There is still large deviation on both tails from quantile of normal distribution on the NQQ plot. SR and leverage points has much improved.

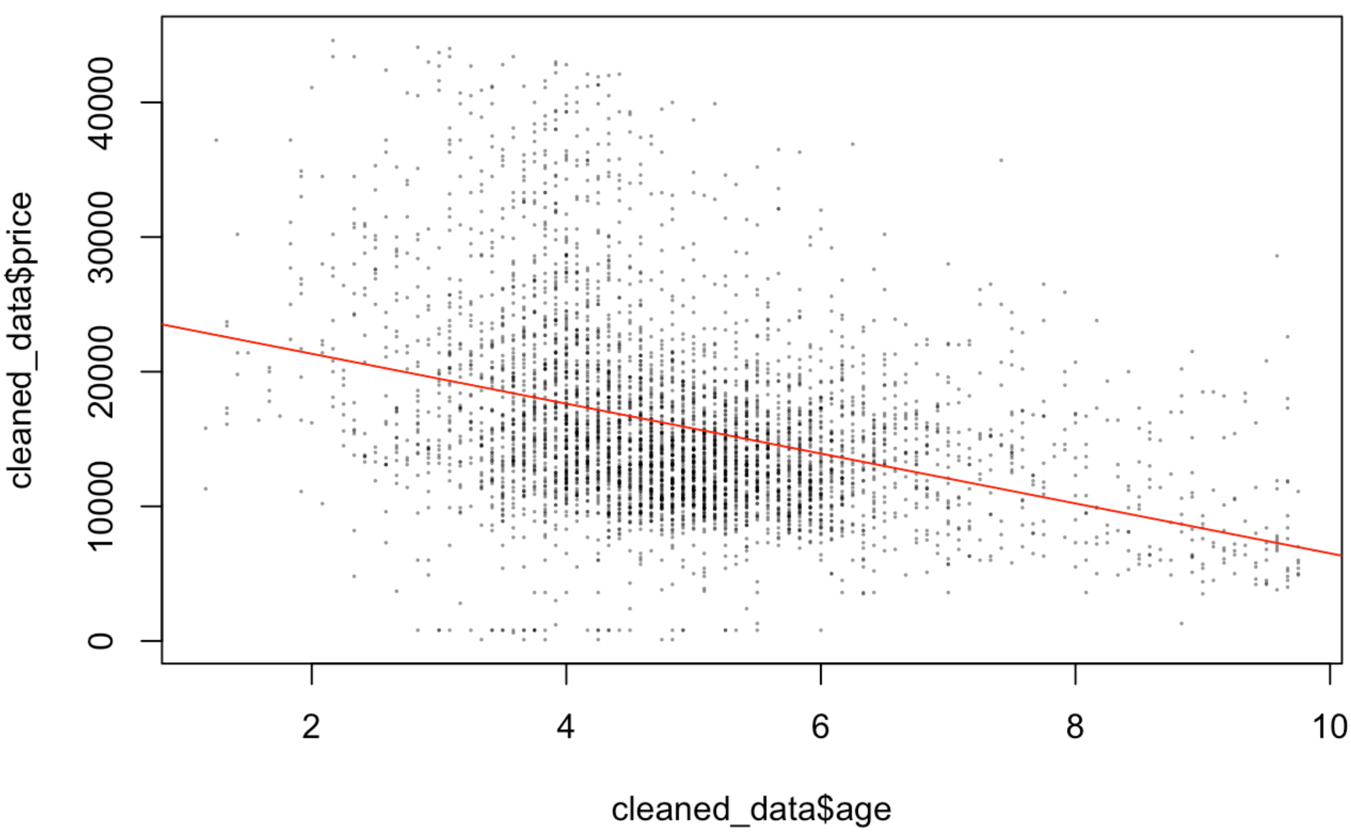
```
# use the correlation matrix plot to check linear association and distribution
ggpairs(cleaned_data,
        upper=list(continuous=wrap("points", alpha=0.3, size=0.1)),
        lower=list(continuous=wrap('cor', size=4)))
```




```
# age normally distributed
# price right skewed, need log transformation
# some negative linear association as r = -0.357
m2 <- lm(cleaned_data$price~cleaned_data$age)
summary(m2)
```

```
##
## Call:
## lm(formula = cleaned_data$price ~ cleaned_data$age)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19000  -4303  -1309   2882  25236
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      25055.89     371.61   67.43   <2e-16 ***
## cleaned_data$age -1854.83      72.96  -25.42   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6596 on 4430 degrees of freedom
## Multiple R-squared:  0.1273, Adjusted R-squared:  0.1271
## F-statistic: 646.3 on 1 and 4430 DF,  p-value: < 2.2e-16
```

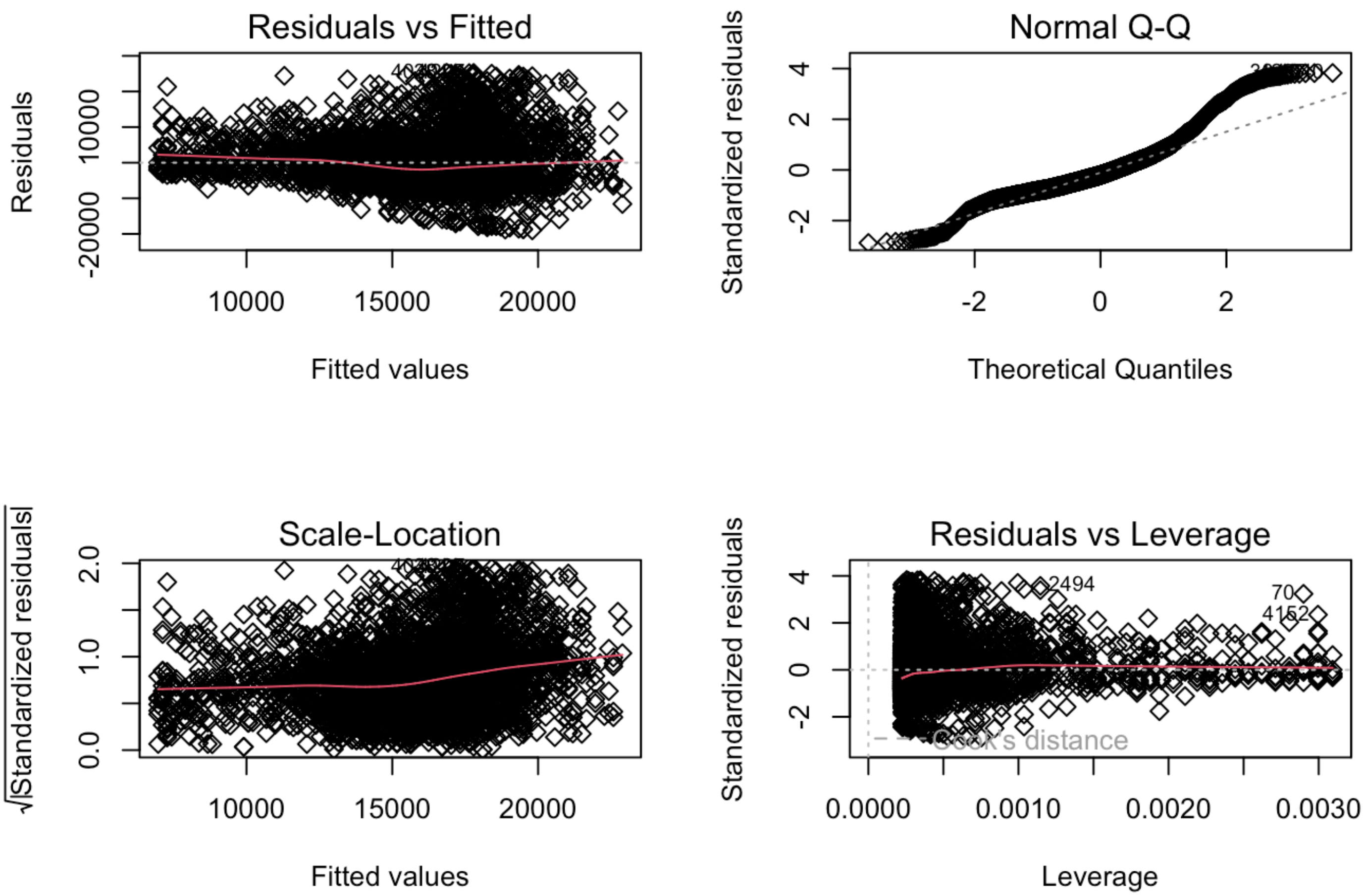
```
plot(cleaned_data$age, cleaned_data$price, col = rgb(0,0,0, alpha = 0.5), cex = 0.1)
abline(m2, col = 'red')
```



```
summary(m2)$r.squared
```

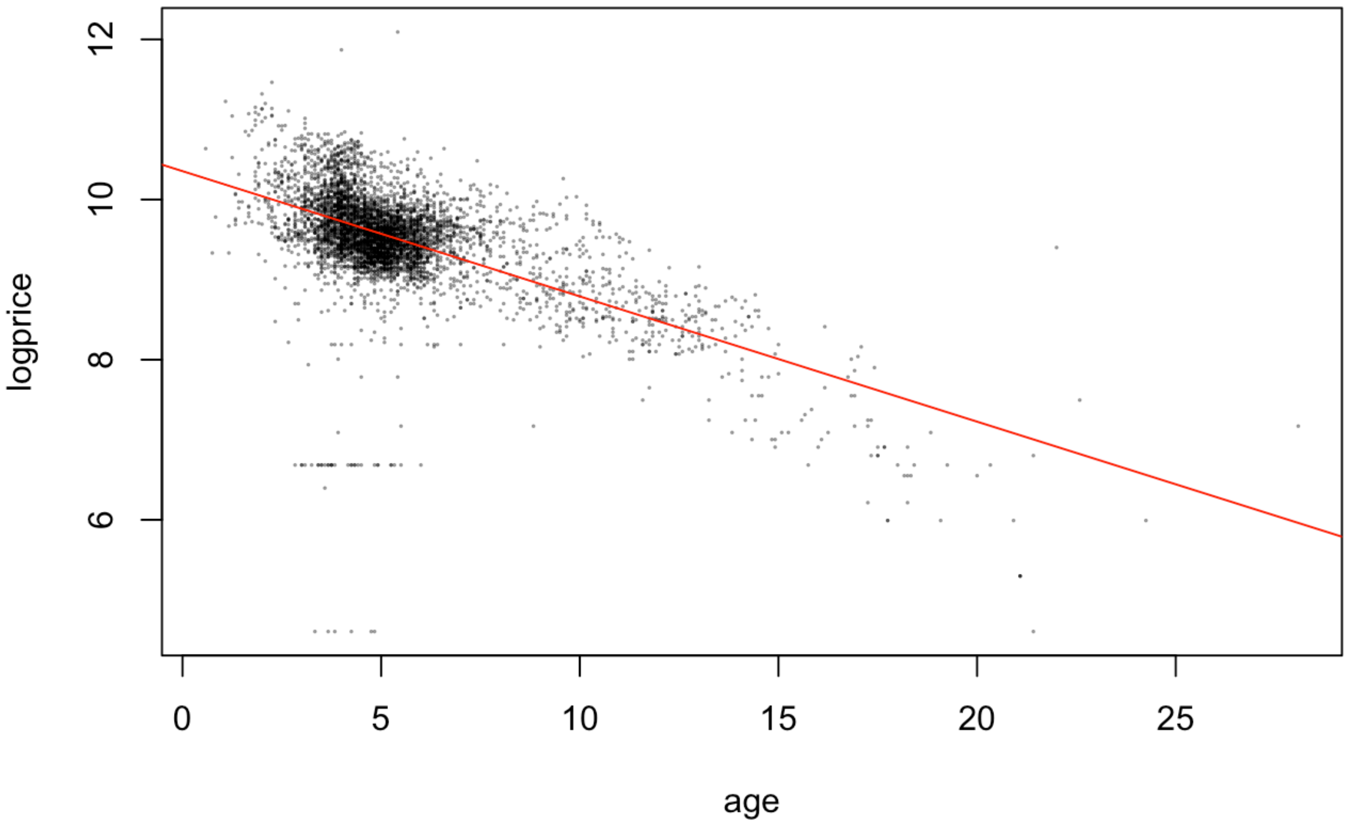
```
## [1] 0.1273159
```

```
par(mfrow = c(2,2))
plot(m2, cex = 1, pch = 5)
```



Model 3 After taking log on the y variable, price, deviation from normal became worse on the NQQ plot. Patterns in the scatterplot has much improved, with clear linearity observed. Patterns in the SR plot has improved as well, and number of high leverage point is greatly reduced. R^2 at 0.3735 shows improvements in the predicability.

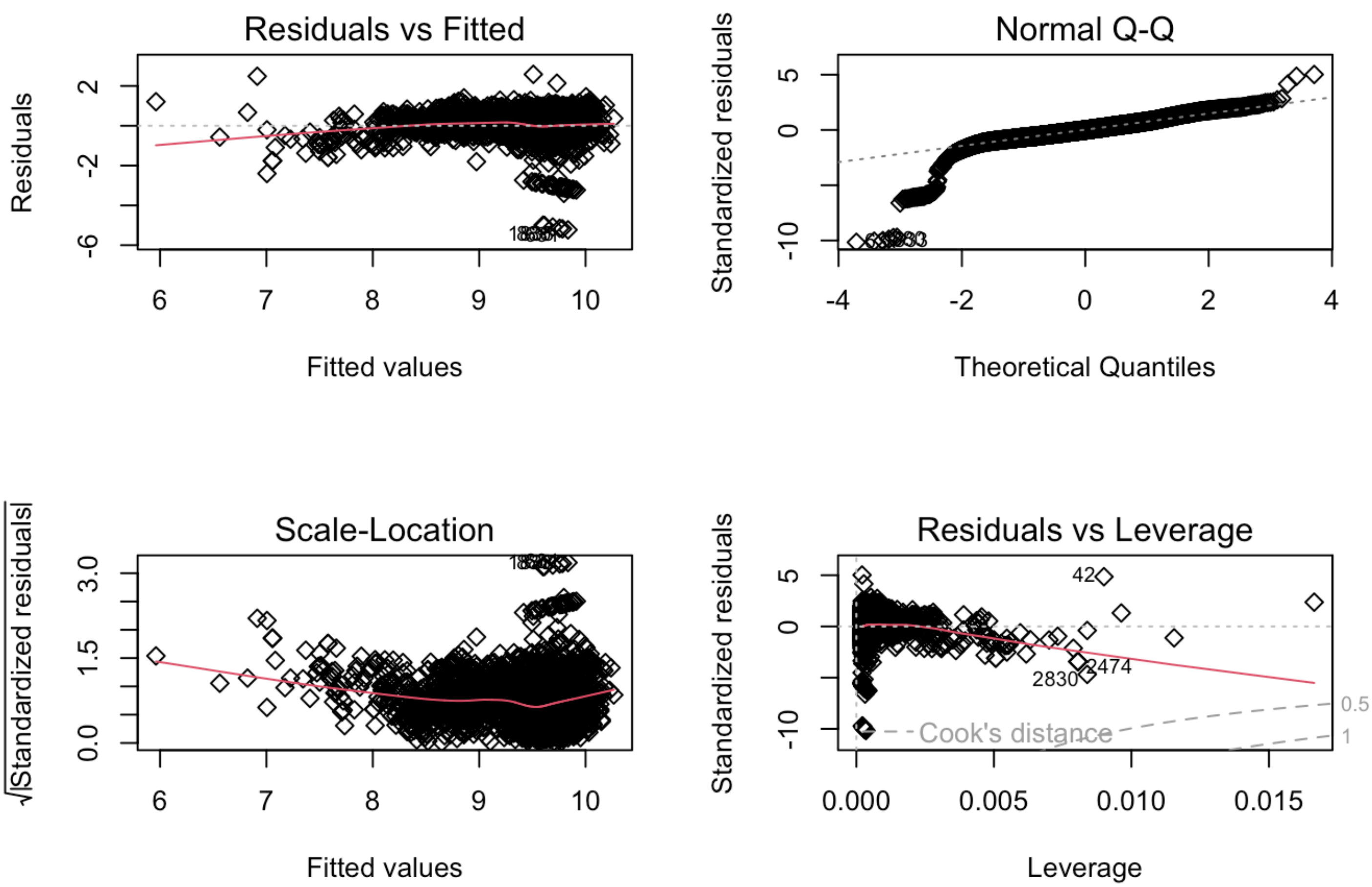
```
# now apply log transformation to y and check the predictability of the model
logprice = log(price)
m3 <- lm(logprice~age)
plot(age, logprice, col = rgb(0,0,0, alpha = 0.5), cex = 0.1)
abline(m3, col = 'red')
```




```
summary(m3)$r.squared
```

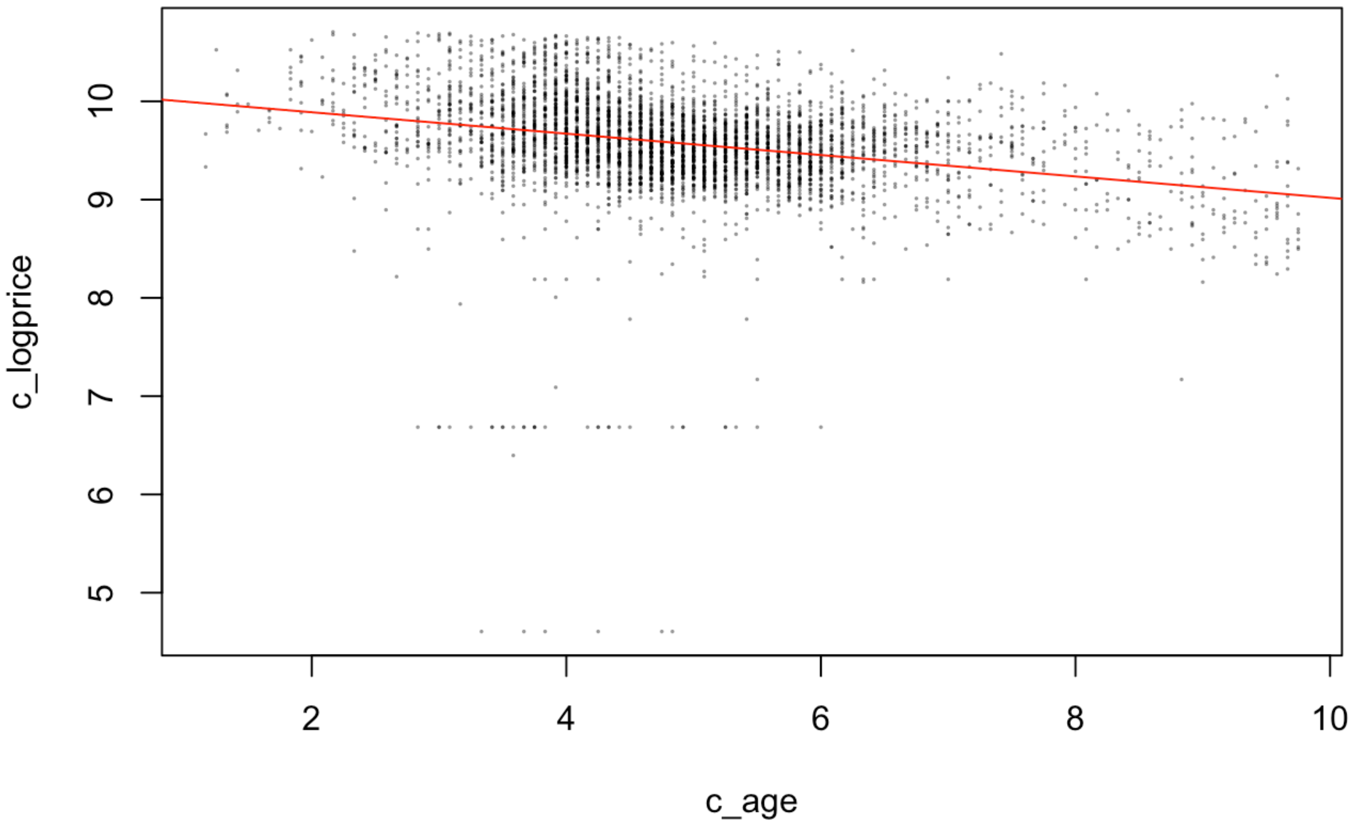
```
## [1] 0.373509
```

```
par(mfrow = c(2,2))
plot(m3, cex = 1, pch = 5)
```



Model 4 After doing log transform on age on the cleaned data, the above identified issue didn’t improved much. Improvements shows in NQQ plot, with deviation only shown in the lower tail, and upper tail become approximately normal. Patterns in the SR plot has become worse, few points with extreme negative SR shows in the bottom half, and most point has positive SR, which is a position we don’t want.The R^2 at 0.0831 comfirm our observation that the predicability of model has been reduced. We would potentially drop the cleaned data.

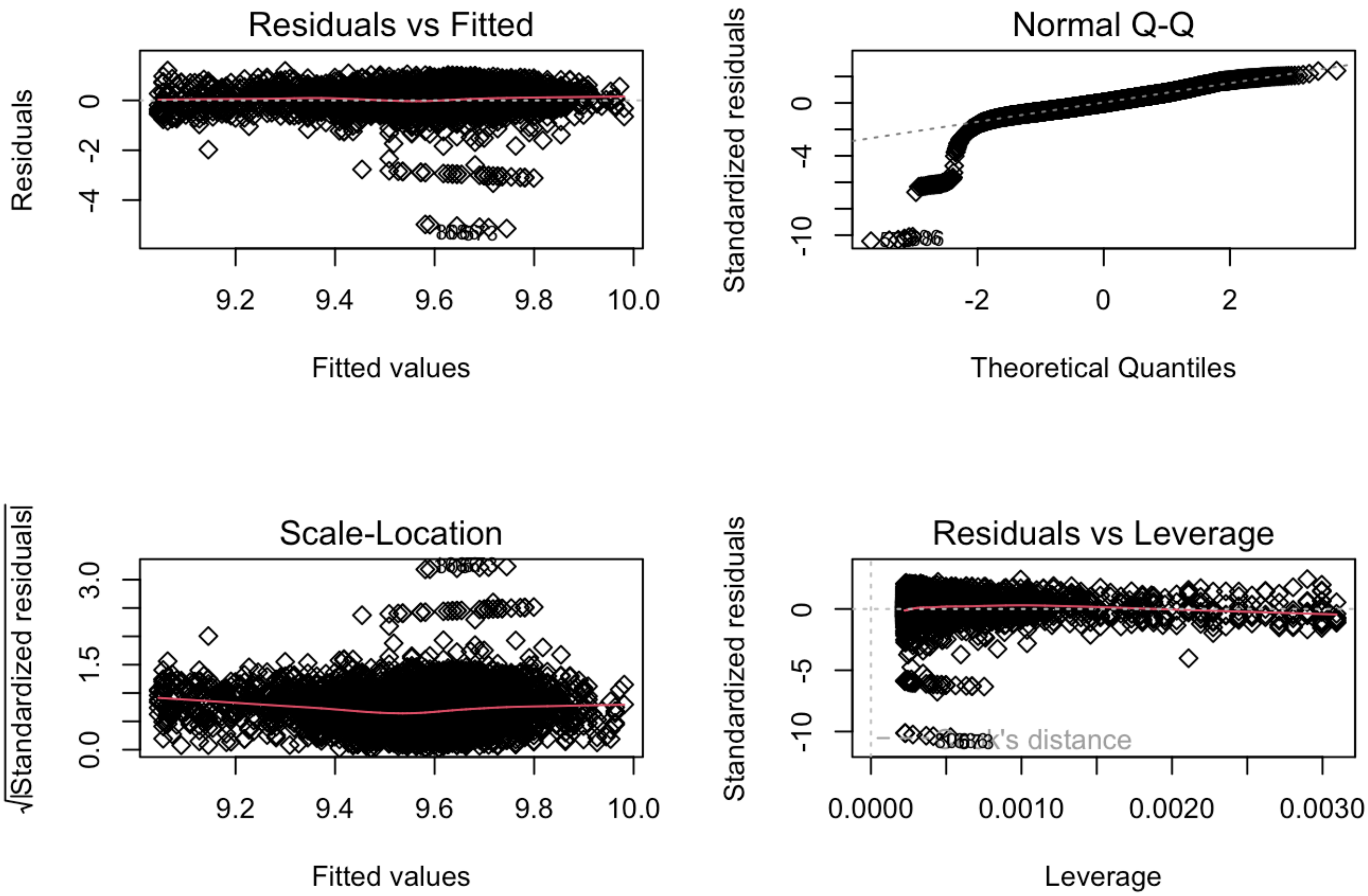
```
# now apply log transformation to cleaned dataset on y and check the predictability o
f the model
c_logprice = log(cleaned_data$price)
c_age = cleaned_data$age
m4 <- lm(c_logprice~c_age)
plot(c_age, c_logprice, col = rgb(0,0,0, alpha = 0.5), cex = 0.1)
abline(m4, col = 'red')
```




```
summary(m4)$r.squared
```

```
## [1] 0.08310191
```

```
par(mfrow = c(2,2))
plot(m4, cex = 1, pch = 5)
```

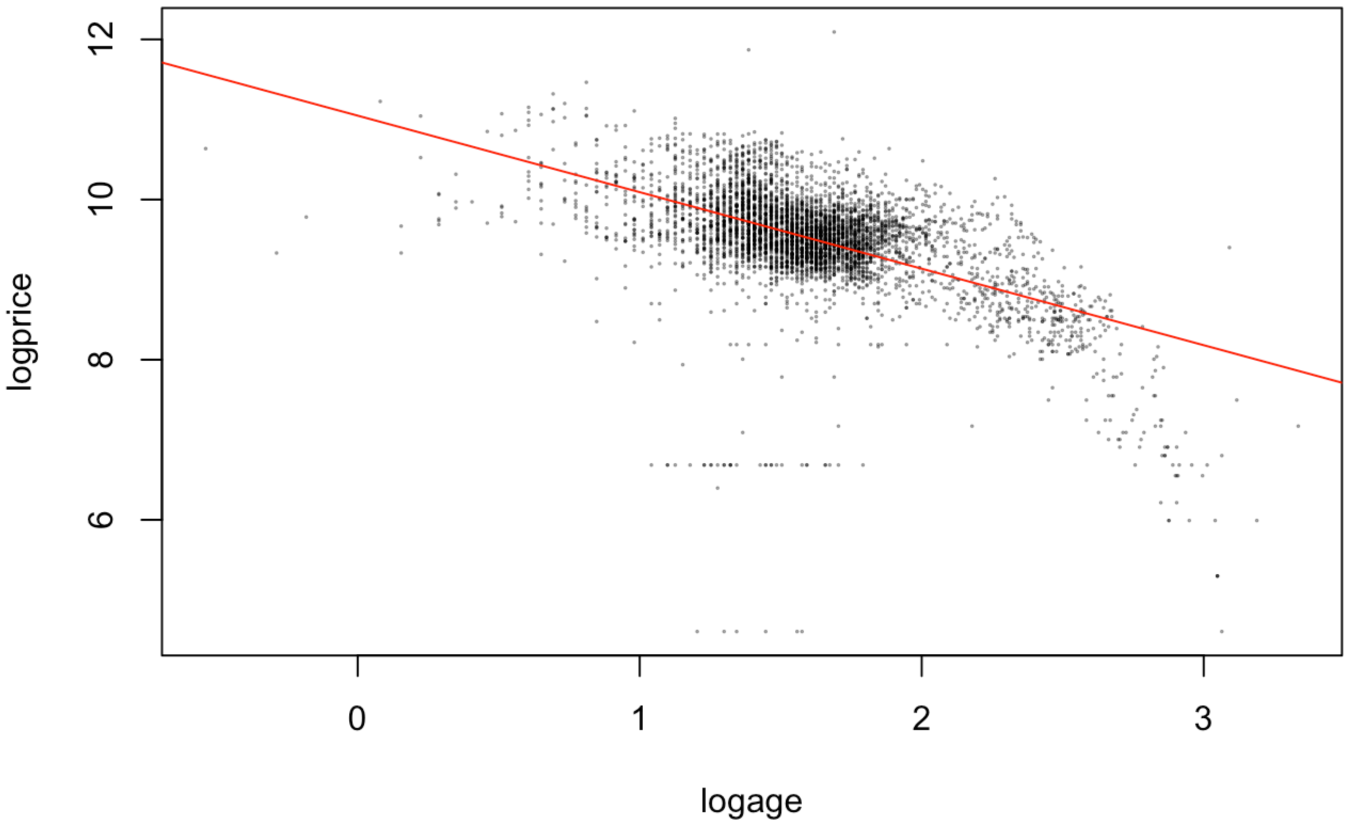



```
# cleaned data has significanly lower r-squared value, considering not using the clea
ned data
```

Model 5

Some patterns shows in the residual plot, but overall is pretty good (mean residual near 0, and little pattern is observed). Non-linear pattern oberved on the scatterplot. Normal QQ plot shows improvement in the middle part of the data, and the tail and bottoms has more deviations than before. Maybe because of the increase deviation on the tails, R^2 has been reduced to 0.3122 compared to the model with log transformation only on y.

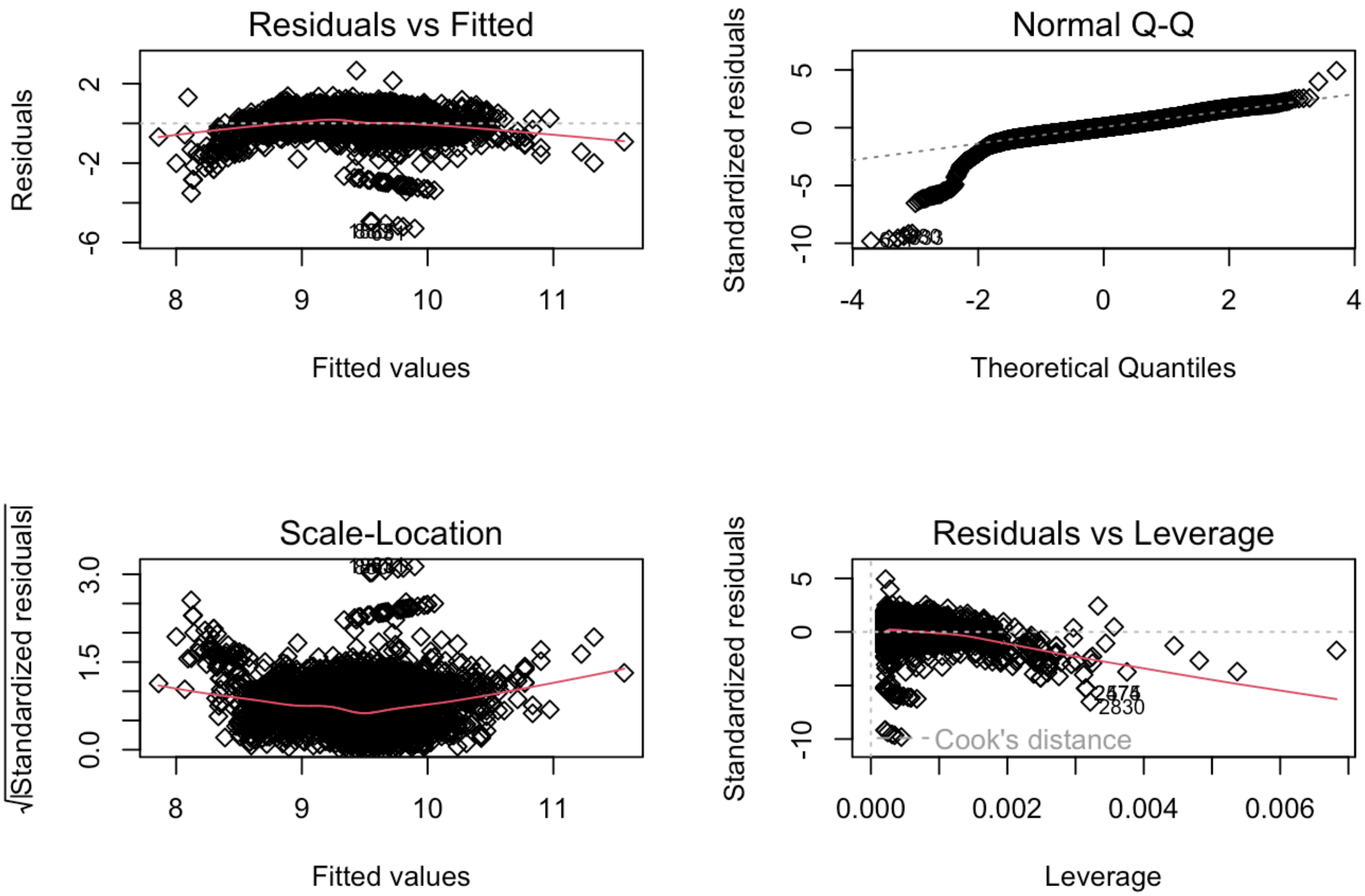
```
# has seen worsen prediction power in the cleaned data... now get back to the origina
l data
# apply log transformation to both x and y and test the model
logprice = log(price)
logage = log(age)
m5 <- lm(logprice~logage)
plot(logage, logprice, col = rgb(0,0,0, alpha = 0.5), cex = 0.1)
abline(m5, col = 'red')
```




```
summary(m5)$r.squared
```

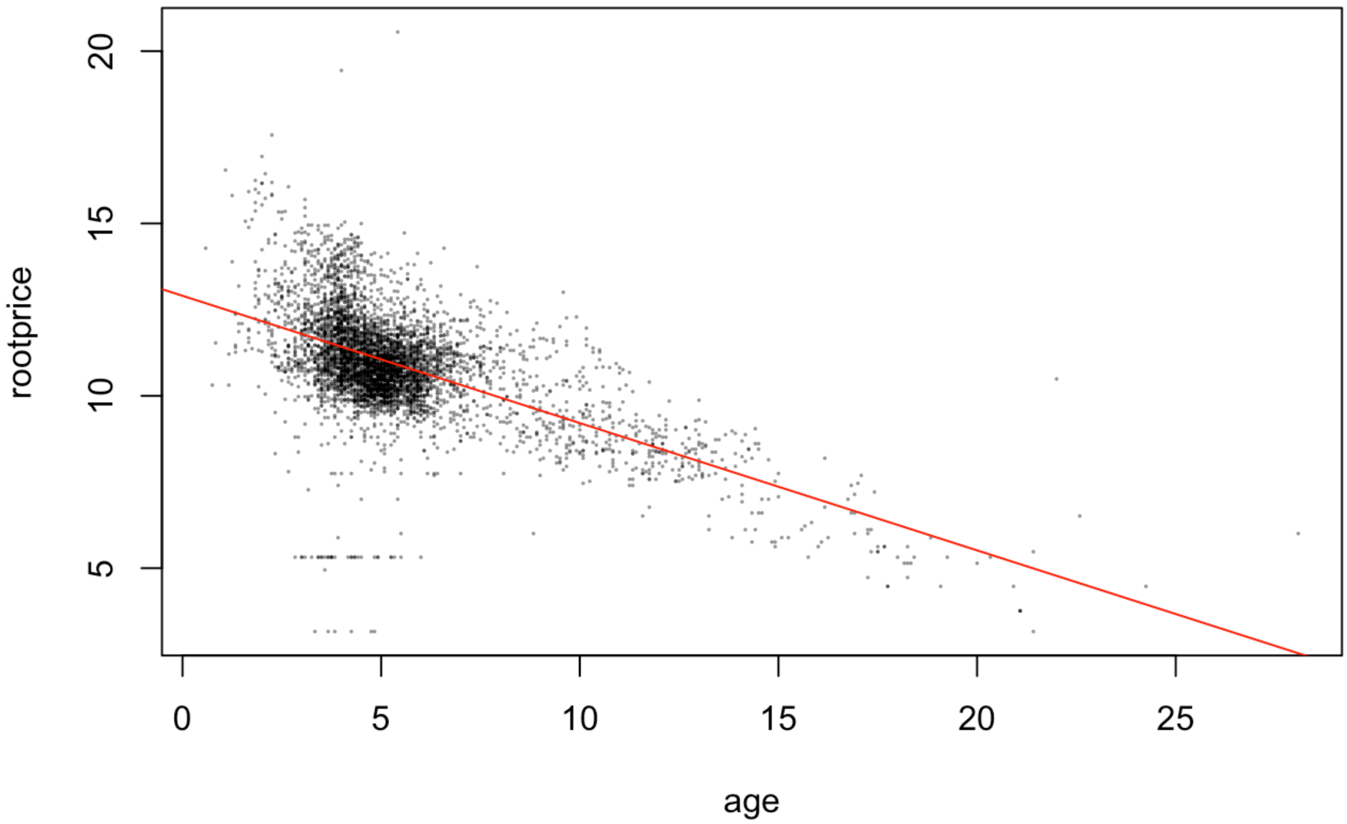
```
## [1] 0.3122484
```

```
par(mfrow = c(2,2))
plot(m5, cex = 1, pch = 5)
```



Model 6 Then we tried take 1/4 root of price, this model is okay with R^2 at 0.3545. But there is pattern in the SR plot, which is not an ideal model because constant variance was violated.

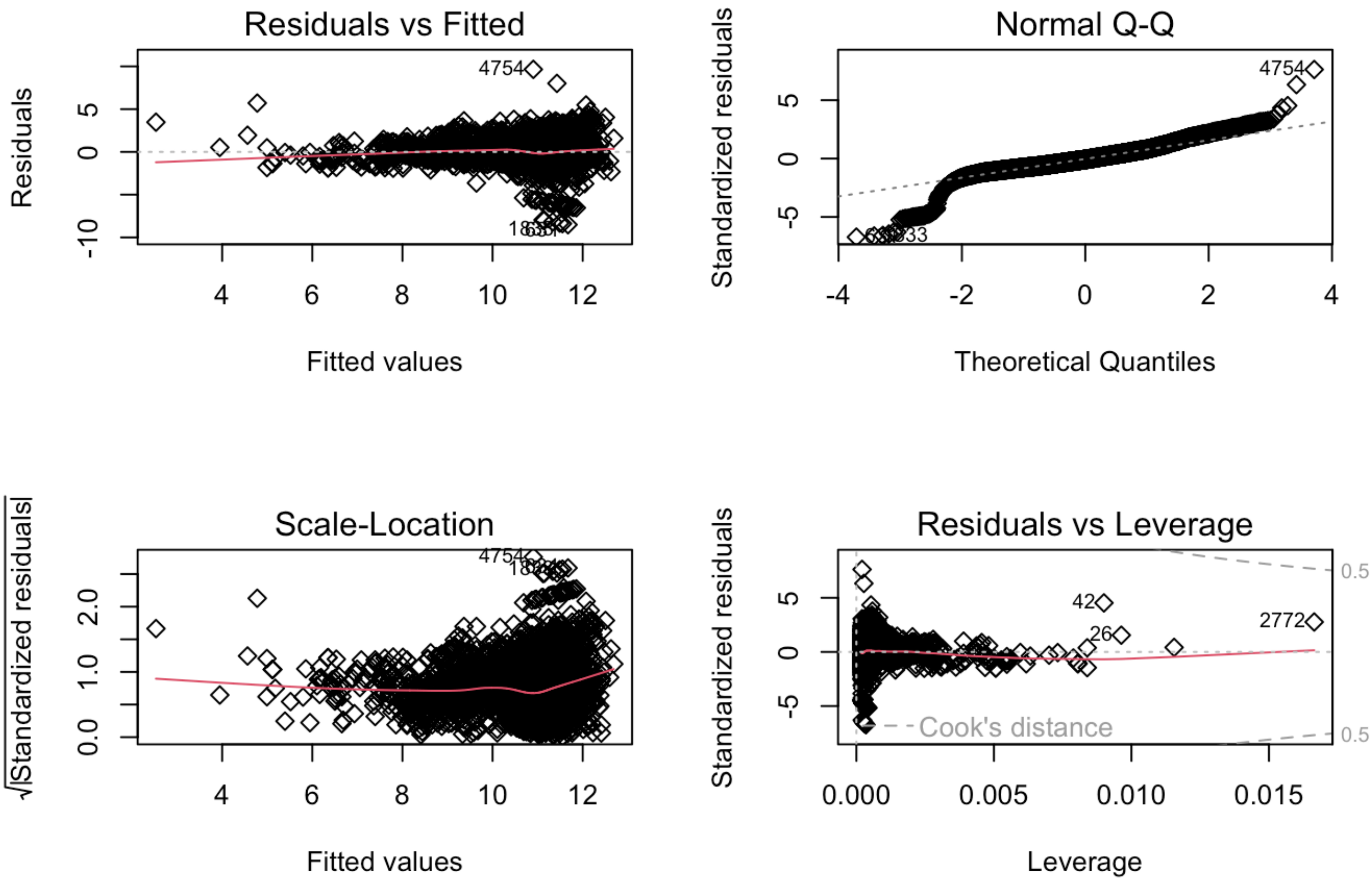
```
rootprice = price^0.25
age = age
m6 <- lm(rootprice~age)
plot(age, rootprice, col = rgb(0,0,0, alpha = 0.5), cex = 0.1)
abline(m6, col = 'red')
```




```
summary(m6)$r.squared
```

```
## [1] 0.3544822
```

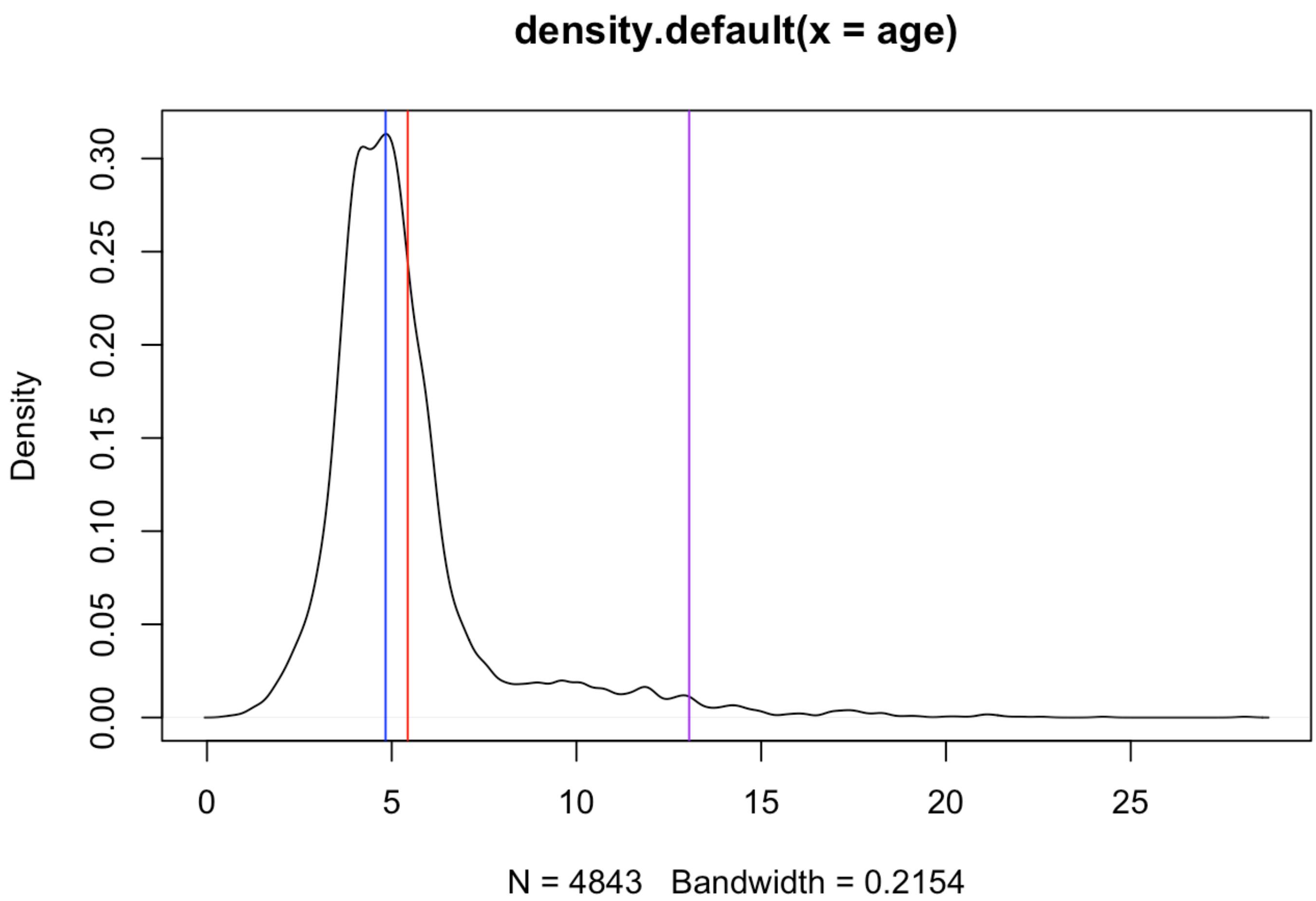
```
par(mfrow = c(2,2))
plot(m6, cex = 1, pch = 5)
```



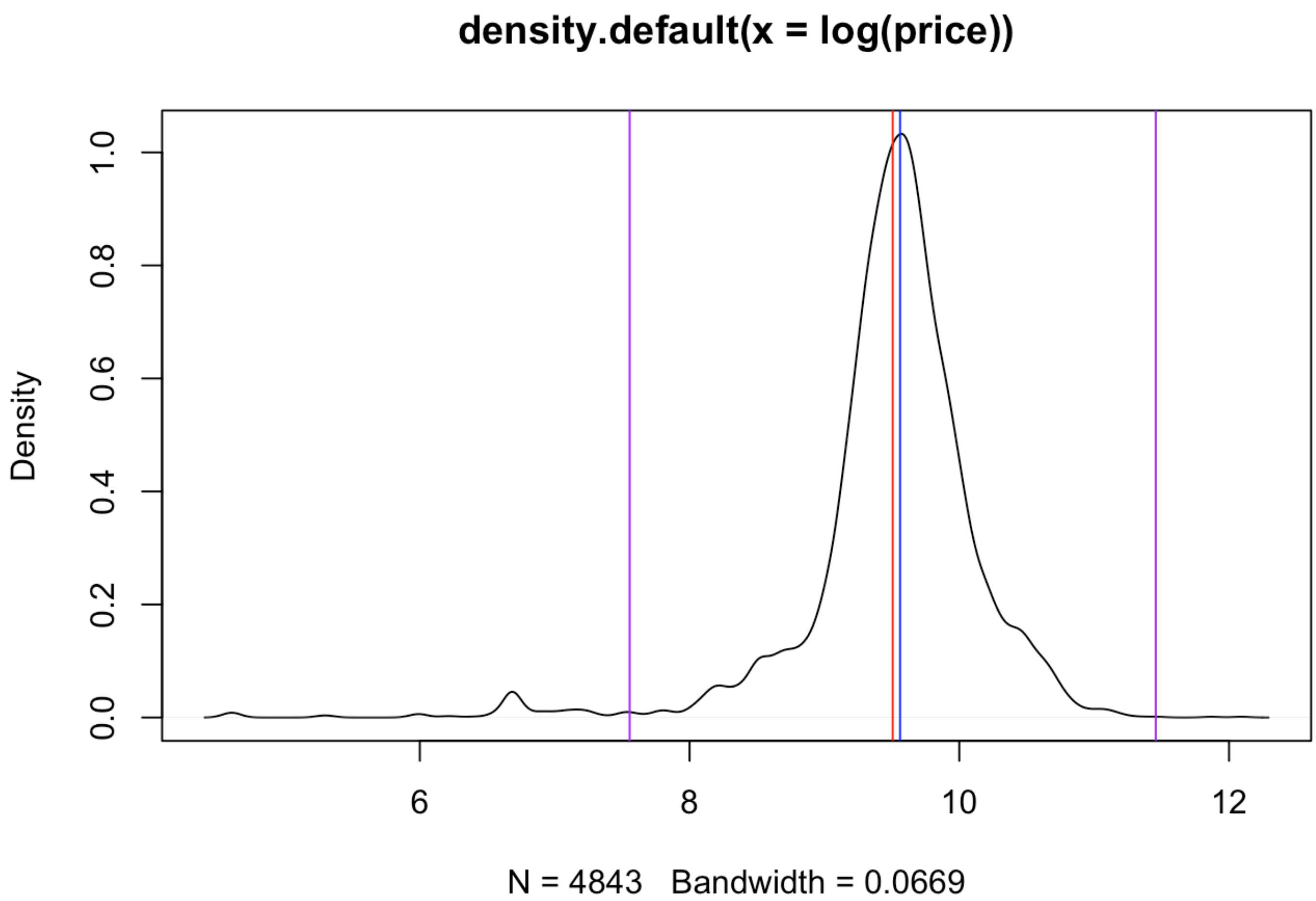
Final Model As discussed above, doing log transformation only on y is the best choice for our simple linear regression. Choose this as our regression model for further analysis.

both x is highly right skewed and logy has slightly trend of left skewness. We may address it in further analysis in the next deliverable.

```
bmw_model = data.frame(age, logprice)
plot(density(age))
abline(v = mean(age), col ='red')
abline(v = median(age), col = 'blue')
abline(v = mean(age)-3*sd(age), col ='purple')
abline(v = mean(age)+3*sd(age), col ='purple')
```




```
plot(density(log(price)))
abline(v = mean(logprice), col ='red')
abline(v = median(logprice), col = 'blue')
abline(v = mean(logprice)-3*sd(logprice), col ='purple')
abline(v = mean(logprice)+3*sd(logprice), col ='purple')
```




```
summary(m3)
```

```
##
## Call:
## lm(formula = logprice ~ age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2298 -0.2420 -0.0005  0.2664  2.5834
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.356672   0.017468  592.91   <2e-16 ***
## age         -0.156505   0.002913  -53.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5147 on 4841 degrees of freedom
## Multiple R-squared:  0.3735, Adjusted R-squared:  0.3734
## F-statistic:  2886 on 1 and 4841 DF,  p-value: < 2.2e-16
```

```
anova(m3)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
	<int>	<dbl>	<dbl>	<dbl>	<dbl>
age	1	764.4691	764.4690859	2886.166	0
Residuals	4841	1282.2531	0.2648736	NA	NA

2 rows

The coefficient is: T value is: P value is: