

다면량분석 과제 3

2015170378

정은영

[Q1] MLR 모형 구축을 위해 필요하지 않은 변수는 어떤 것들이 있는가? 왜 그렇게 생각하는가?

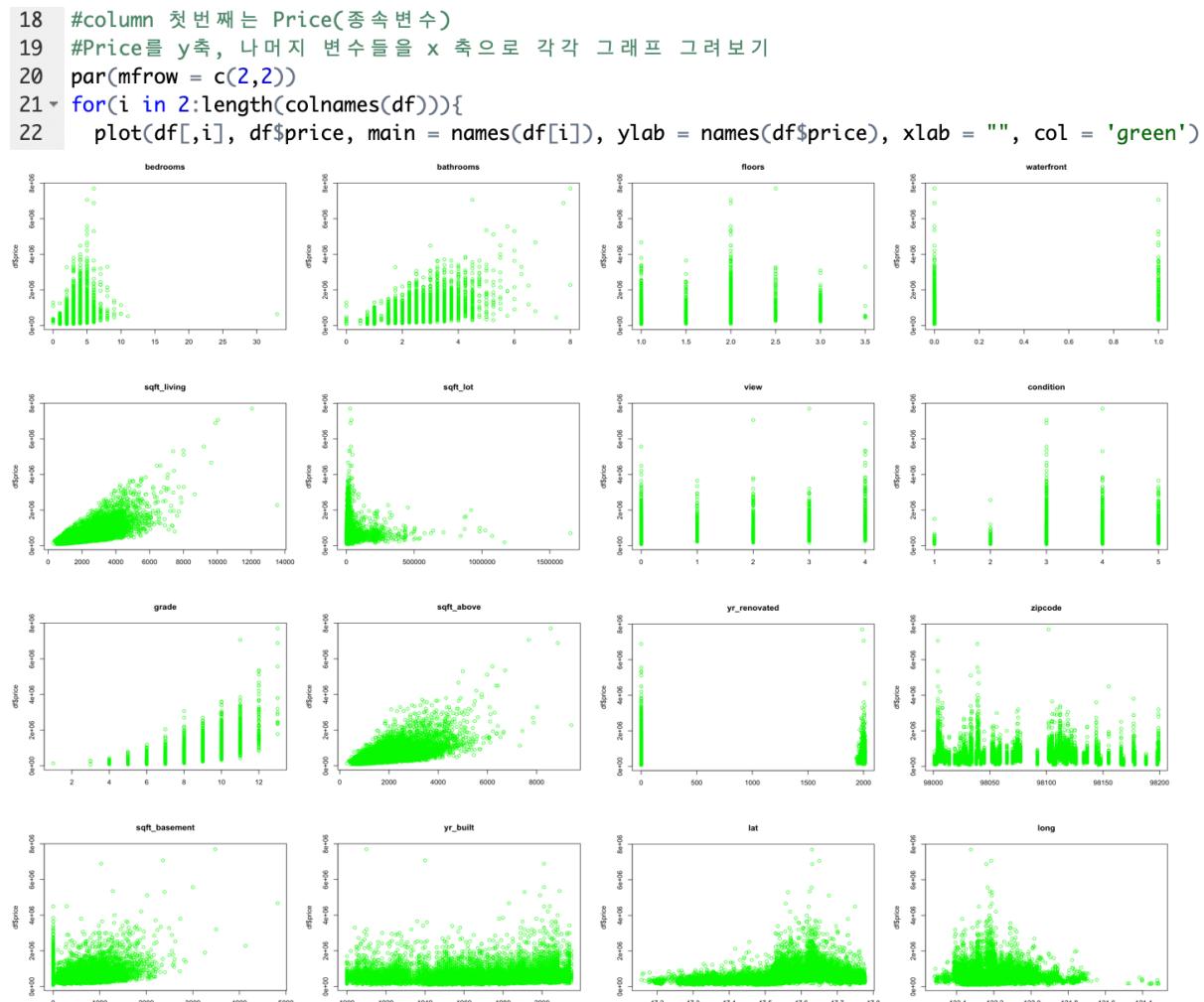
```
12 # [Q1]
13
14 df[1:2] <- NULL #id, date 제거
```

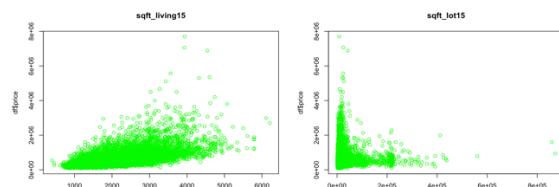
id는 Price와 완전 무관하며, 날짜도 거의 관련이 없을 것이라 생각한다.(계절적 요인이 아주 살짝 있을 수 있으나 무시)

다음 물음에 대해서는 [Q1]에서 선택한 변수들은 제외하고 답변하시오.

[Q2] 개별 입력 변수들에 대하여 각각 다음과 같은 단변량 통계량을 계산하고 Box plot 을 도시하시오: Mean, Standard deviation, Skewness, Kurtosis. 전체 변수 중에서 정규분포를 따른다고 할 수 있는 변수 들은 몇 개인가? 정규분포를 따른다고 가정한 근거는 무엇인가?

먼저, 한눈에 볼 수 있도록 Price 를 y 축으로 각각 변수들에 대한 그래프를 그려보았다.





#데이터프레임을 참고하고 그래프를 그려본 결과 변수 bedrooms, bathrooms, floors, waterfront, view, condition, grade, zipcode 가 discrete 변수라고 생각하였다.

#따라서 이 변수들의 column index 를 구해보면 2,3,6,7,8,9,10,15 이다.

```

28 cateIdx <- c(2,3,6,7,8,9,10,15) #categorical Var index
29 conIdx = c(2:19)[!(c(2:19) %in% cateIdx)] #non categorical index
30 length(conIdx)

```

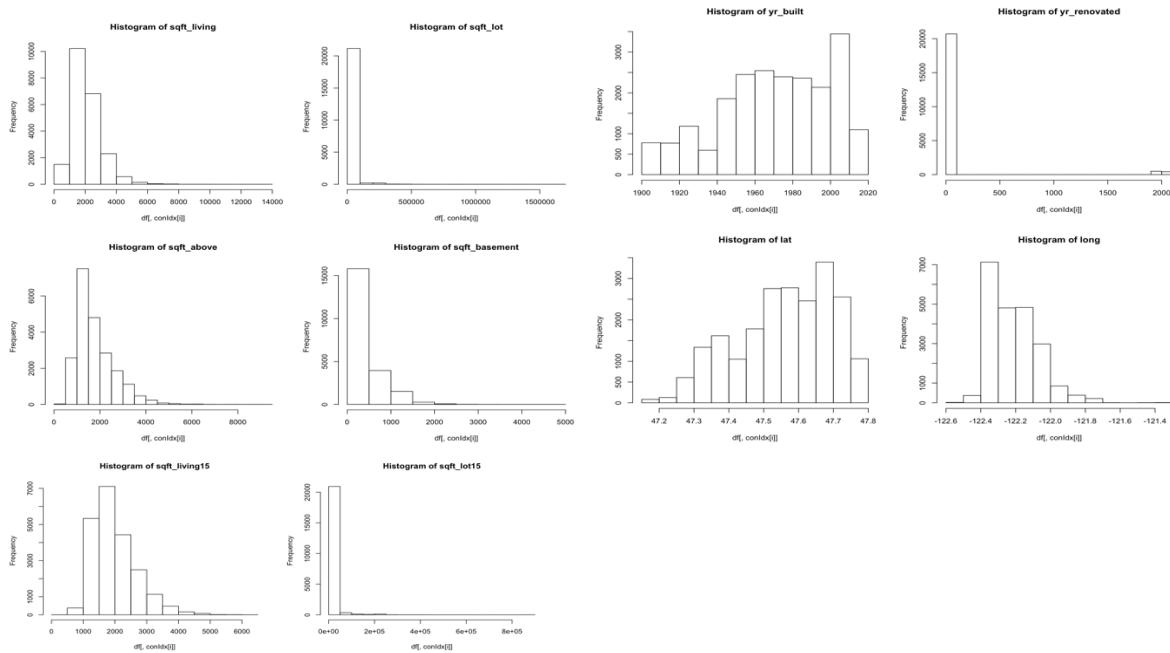
continuous 한 변수들은 mean, sqrt, skewness, kurtosis 를 구하고 histogram 을 그려보았다.

```

31 nMat <- matrix(c(1:10*4),nrow=10,ncol=4)
32 colnames(nMat)<- c("mean", "std", "skewness", "kurtosis")
33 rownames(nMat)<- colnames(df[conIdx])
34
35 #continuous value
36 par(mfrow=c(2,2))
37 for(i in 1:length(conIdx)){
38   hist(df[,conIdx[i]], main = paste("Histogram of" , colnames(df[conIdx[i]])))
39   nMat[i,1] <- mean(unlist(df[,conIdx[i]]))
40   nMat[i,2] <- sqrt(var(df[,conIdx[i]]))
41   nMat[i,3] <- skewness(df[,conIdx[i]])
42   nMat[i,4] <- kurtosis(df[,conIdx[i]])
43 }

```

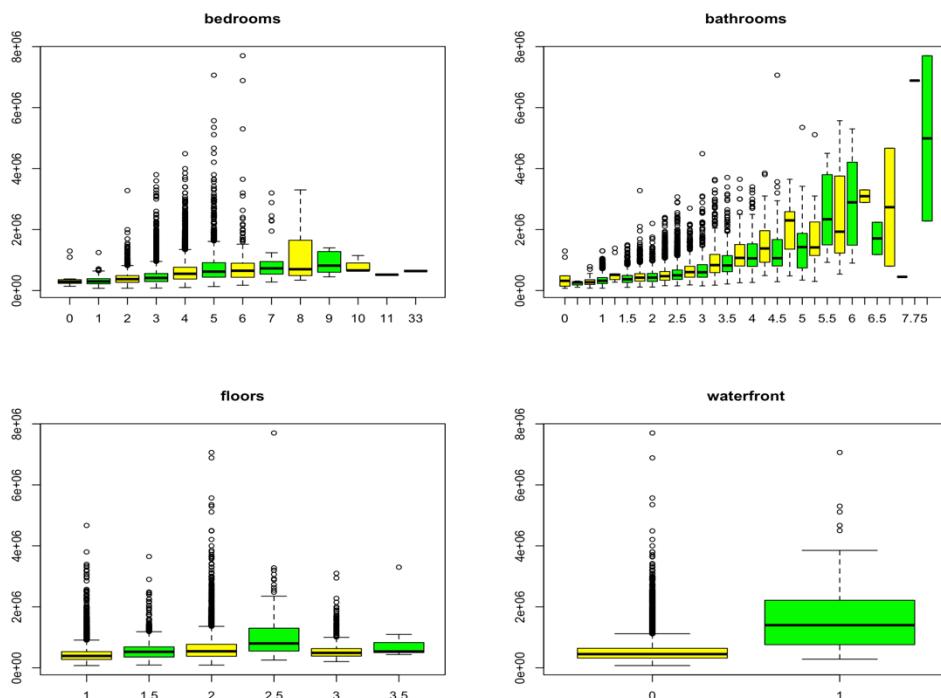
	mean	std	skewness	kurtosis
sqft_living	2079.89974	9.184409e+02	1.4714533	8.241603
sqft_lot	15106.96757	4.142051e+04	13.0591125	288.011596
sqft_above	1788.39069	8.280910e+02	1.4465641	6.401239
sqft_basement	291.50905	4.425750e+02	1.5778555	5.714668
yr_built	1971.00514	2.937341e+01	-0.4697728	2.342467
yr_renovated	84.40226	4.016792e+02	4.5491776	21.696548
lat	47.56005	1.385637e-01	-0.4852368	2.323566
long	-122.21390	1.408283e-01	0.8849916	4.048981
sqft_living15	1986.55249	6.853913e+02	1.1081044	4.596449
sqft_lot15	12768.45565	2.730418e+04	9.5060834	153.727957

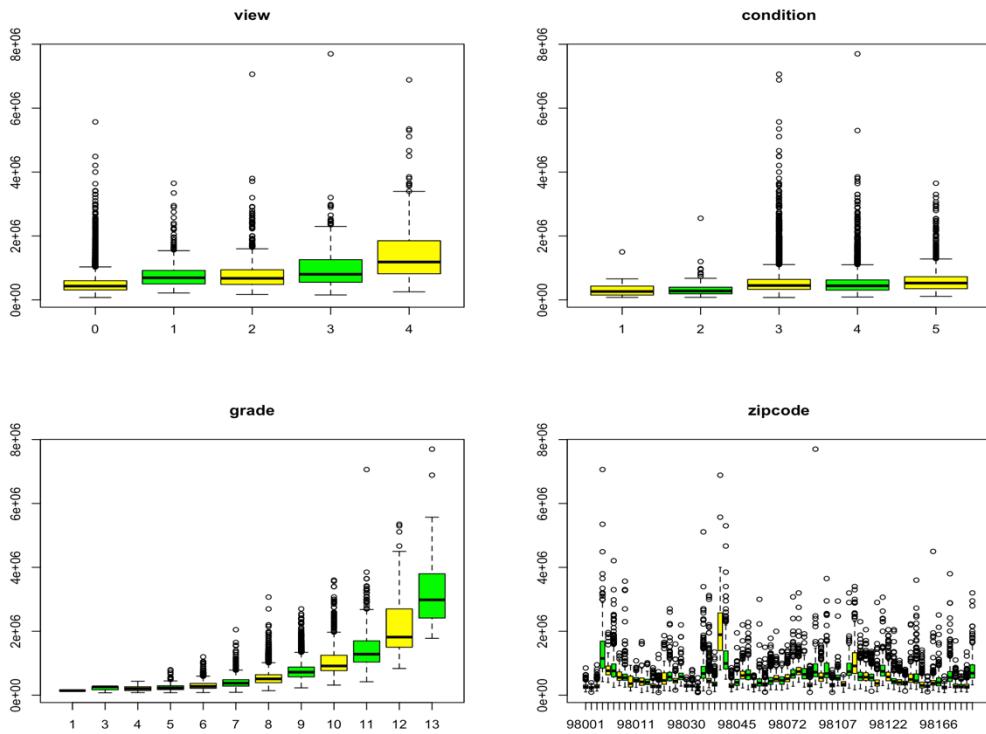


히스토그램과 표를 참고했을 때 일단 표준정규분포의 skewness 는 0, kurtosis 는 3 이어야 하기 때문에 그나마 비슷한 것은 yr_built, lat, long, sqft_living15 정도이다.

discrete 한 Input value 들에 대해서 boxplot 을 그려준다.

```
44 #discrete value
45 par(mfrow=c(2,2))
46 for(i in cateIdx){
47   boxplot(df[,1]~df[,i], xlab='', main=names(df[i]), col=c("yellow","green"))
48 }
```





box Plot 을 살펴보면, box plot 을 벗어난 값들이 많이 보이고, 이것은 skewness 가 크다는 것을 보여준다. 따라서 그나마 정규분포를 따르는 input Variables 는 위에서 구한 yr_built, lat, long, sqft_living15 정도이다.

[Q3] [Q2]의 Box plot 을 근거로 각 변수들에 대한 이상치(너무 값이 크거나 작은 객체) 조건을 정의하고, 해당하는 객체들을 데이터셋에서 제거해 보시오.

box plot 을 보았을 때, bedroom 의 경우 0-11 까지와 갑자기 33 이 등장한다. 따라서 10 보다 큰 bedroom 데이터를 삭제한다.

```

56 df$bedrooms<- ifelse(df$bedrooms> 10, NA, df$bedrooms) #outlier 제거
57
58 df<- na.omit(df) #결측치 제거

```

다음 각 물음에 대해서는 [Q3]에서 제거된 객체들을 제외시킨 데이터프레임을 사용하여 답하시오.

[Q4] 가능한 모든 두 쌍의 입력 변수 조합에 대한 산점도(scatter plot) 및 Correlation plot (hint: “corrplot” 패키지의 corrplot() 함수 사용) 상관관계를 계산해 보시오. 어떤 두 조합의 변수들이 서로 강한 상관관계가 있다고 할 수 있는가?

```
64 #[Q4]
65 corr <- cor(df) #correlation
66 corrplot(corr, method = "color", outline = T, cl.pos = 'n', rect.col = "black", tl
.col = "indianred4", addCoef.col = "black", number.digits = 2, number.cex = 0.60, tl
.cex = 0.7, cl.cex = 1, col = colorRampPalette(c("green4","white","red"))(100))
#Corrlation Plot
```

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement	yr_built	yr_renovated	zipcode	lat	long	sqft_living15	sqft_lot15
price	1	0.31	0.53	0.7	0.09	0.26	0.27	0.4	0.04	0.67	0.61	0.32	0.05	0.13	-0.05	0.31	0.02	0.59	0.08
bedrooms	0.31	1	0.52	0.58	0.03	0.18	-0.01	0.08	0.03	0.36	0.48	0.3	0.15	0.02	-0.15	-0.01	0.13	0.39	0.03
bathrooms	0.53	0.52	1	0.75	0.09	0.5	0.06	0.19	-0.12	0.66	0.69	0.28	0.51	0.05	-0.2	0.02	0.22	0.57	0.09
sqft_living	0.7	0.58	0.75	1	0.17	0.35	0.1	0.28	-0.06	0.76	0.88	0.44	0.32	0.06	-0.2	0.05	0.24	0.76	0.18
sqft_lot	0.09	0.03	0.09	0.17	1	-0.01	0.02	0.07	-0.01	0.11	0.18	0.02	0.05	0.01	-0.13	-0.09	0.23	0.14	0.72
floors	0.26	0.18	0.5	0.35	-0.01	1	0.02	0.03	-0.26	0.46	0.52	-0.25	0.49	0.01	-0.06	0.05	0.13	0.28	-0.01
waterfront	0.27	-0.01	0.06	0.1	0.02	0.02	1	0.4	0.02	0.08	0.07	0.08	-0.03	0.09	0.03	-0.01	-0.04	0.09	0.03
view	0.4	0.08	0.19	0.28	0.07	0.03	0.4	1	0.05	0.25	0.17	0.28	-0.05	0.1	0.08	0.01	-0.08	0.28	0.07
condition	0.04	0.03	-0.12	-0.06	-0.01	-0.26	0.02	0.05	1	-0.14	-0.16	0.17	-0.36	-0.06	0	-0.01	-0.11	-0.09	0
grade	0.67	0.36	0.66	0.76	0.11	0.46	0.08	0.25	-0.14	1	0.76	0.17	0.45	0.01	-0.18	0.11	0.2	0.71	0.12
sqft_above	0.61	0.48	0.69	0.88	0.18	0.52	0.07	0.17	-0.16	0.76	1	-0.05	0.42	0.02	-0.26	0	0.34	0.73	0.19
sqft_basement	0.32	0.3	0.28	0.44	0.02	-0.25	0.08	0.28	0.17	0.17	-0.05	1	-0.13	0.07	0.07	0.11	-0.14	0.2	0.02
yr_built	0.05	0.15	0.51	0.32	0.05	0.49	-0.03	-0.05	-0.36	0.45	0.42	-0.13	1	-0.22	-0.35	-0.15	0.41	0.33	0.07
yr_renovated	0.13	0.02	0.05	0.06	0.01	0.01	0.09	0.1	-0.06	0.01	0.02	0.07	-0.22	1	0.06	0.03	-0.07	0	0.01
zipcode	-0.05	-0.15	-0.2	-0.2	-0.13	-0.06	0.03	0.08	0	-0.18	-0.26	0.07	-0.35	0.06	1	0.27	-0.56	-0.28	-0.15
lat	0.31	-0.01	0.02	0.05	-0.09	0.05	-0.01	0.01	-0.01	0.11	0	0.11	-0.15	0.03	0.27	1	-0.14	0.05	-0.09
long	0.02	0.13	0.22	0.24	0.23	0.13	-0.04	-0.08	-0.11	0.2	0.34	-0.14	0.41	-0.07	-0.56	-0.14	1	0.33	0.25
sqft_living15	0.59	0.39	0.57	0.76	0.14	0.28	0.09	0.28	-0.09	0.71	0.73	0.2	0.33	0	-0.28	0.05	0.33	1	0.18
sqft_lot15	0.08	0.03	0.09	0.18	0.72	-0.01	0.03	0.07	0	0.12	0.19	0.02	0.07	0.01	-0.15	-0.09	0.25	0.18	1

Price 와의 correlation 높은 것

sqft_living(0.7), grade(0.67), sqft_above(0.6), sqft_living15(0.59), #bathrooms(0.52)

Price 와의 correlation 낮은 것

long(0.02), yr_builtin(0.05), zipcode(-0.05), sqft_lot15(0.08), condition(0.04), sqft_lot(0.09)

#Price 제외한 두 변수의 상관관계가 높은 것들

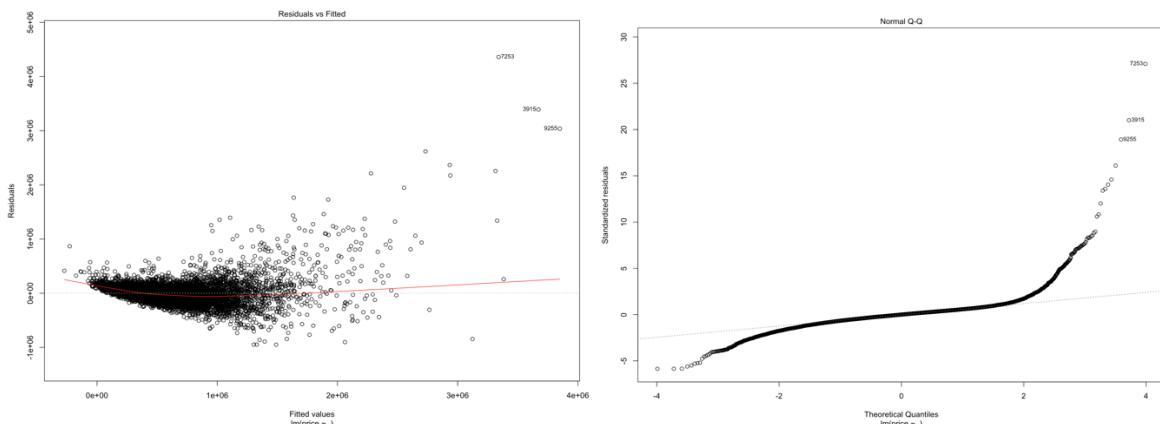
sqft_living & sqft_above(0.87), sqft_living & sqft_living15(0.76), sqft_living & grade(0.77), sqft_living & bathrooms(0.75)

[Q5] 전체 데이터셋을 70%의 학습 데이터와 30%의 테스트 데이터로 분할한 후 모든 변수를 사용하여 MLR 모델을 학습해 보시오. Adjusted R² 값을 통해 데이터의 선형성(linearity)을 판단해 보시오. Residual plot 과 Q-QPlot 을 도시하고 Ordinary Least Square 방식의 Solution 이 만족해야 하는 가정들이 만족될만한 수준인지 정성적으로 판단해 보시오.

```

75 # Split the data into the training/validation sets
76
77 nR <- nrow(df) #행의 갯수 계산하는 명령어
78 nC <- ncol(df) #열의 갯수 계산하는 명령어
79
80 set.seed(12345) #랜덤 알고리즘 지정
81 df_trn_idx <- sample(1:nR, round(0.7*nR)) #70%를 샘플링
82 df_trn_data <- df[df_trn_idx,]
83 df_val_data <- df[-df_trn_idx,] #training 아닌 것들
84
85 mlr_df <- lm(price ~ ., data = df_trn_data)
86 summary(mlr_df)
87 plot(mlr_df)

```



모델로 Residual plot 을 그렸을 때 뺄간선이 거의 수평이었다.

또한 Q-Q plot에서 데이터가 -2에서 2 정도까지 매우 잘 fitting되어 있었다. 따라서 OLS가 만족해야 하는 가정들이 만족될 만한 수준인 것을 확인할 수 있다.

모델의 요약은 다음과 같다.

```
> summary(mlr_df)

Call:
lm(formula = price ~ ., data = df_trn_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-1082826 -99400   -9106   77249  4374824 

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 4.731e+06 3.442e+06  1.375   0.1693  
bedrooms   -3.697e+04 2.336e+03 -15.831  < 2e-16 ***
bathrooms   4.218e+04 3.824e+03  11.028  < 2e-16 *** 
sqft_living  1.485e+02 5.177e+00  28.685  < 2e-16 *** 
sqft_lot     1.391e-01 5.440e-02   2.557   0.0106 *  
floors       5.106e+03 4.236e+03   1.206   0.2280  
waterfront   6.143e+05 2.035e+04  30.183  < 2e-16 *** 
view         5.199e+04 2.507e+03  20.737  < 2e-16 *** 
condition   2.452e+04 2.794e+03   8.775  < 2e-16 *** 
grade        9.770e+04 2.528e+03  38.646  < 2e-16 *** 
sqft_above   2.728e+01 5.123e+00   5.326  1.02e-07 *** 
sqft_basement NA       NA       NA       NA      
yr_built     -2.585e+03 8.540e+01 -30.264  < 2e-16 *** 
yr_renovated 1.882e+01 4.289e+00   4.387  1.16e-05 *** 
zipcode     -5.557e+02 3.888e+01 -14.294  < 2e-16 *** 
lat          5.933e+05 1.266e+04  46.858  < 2e-16 *** 
long         -2.125e+05 1.549e+04 -13.718  < 2e-16 *** 
sqft_living15 2.126e+01 4.063e+00   5.233  1.69e-07 *** 
sqft_lot15   -3.931e-01 8.636e-02  -4.552  5.37e-06 *** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 198700 on 15110 degrees of freedom
Multiple R-squared:  0.7038,    Adjusted R-squared:  0.7035 
F-statistic:  2112 on 17 and 15110 DF,  p-value: < 2.2e-16
```

Adjusted R²는 0.7035로 모델이 y의 평균보다 70% 잘 설명해준다는 것을 의미한다.

[Q6] 유의수준 0.01에서 모형 구축에 통계적으로 유의미한 변수들은 어떤 것들이 있는가? 해당 변수들은 Price와 양/음 중에서 어떤 상관관계를 갖고 있는가?

```
-- 
89 uColNames <- colnames(df)[-12] #sqft_basement는 NA 이므로 제외
90 for (i in 2:(length(summary(mlr_df)$coefficient[,1]))){
91   if(summary(mlr_df)$coefficient[(i),4] <= 0.01){
92     print(paste(uColNames[i], "//", summary(mlr_df)$coefficient[(i),1]))
93   }
94 }
```

Summary의 **을 가지고 확인할 수도 있지만 코드로 해결해보았다.

```
[1] "bedrooms // -36974.7568266955"
[1] "bathrooms // 42175.0758454522"
[1] "sqft_living // 148.488960211389"
[1] "waterfront // 614347.785782467"
[1] "view // 51993.4993809033"
[1] "condition // 24519.126774517"
[1] "grade // 97704.3653419738"
[1] "sqft_above // 27.2822143302475"
[1] "yr_built // -2584.59879726132"
[1] "yr_renovated // 18.8150075690582"
[1] "zipcode // -555.737332048751"
[1] "lat // 593297.064523243"
[1] "long // -212502.077193067"
[1] "sqft_living15 // 21.2595214097243"
[1] "sqft_lot15 // -0.39309108637175"
```

왼쪽은 통계적으로 MLR 에 유의한 변수이름이며 오른쪽은 coefficient 이다. 각 coefficient 의 부호가 Price 와의 상관관계를 나타낸다.

[Q7] Test 데이터셋에 대하여 MAE, MAPE, RMSE 를 계산하고 그에 대한 해석을 해 보시오.

```
98 perf_eval_reg <- function(tgt_y, pre_y){ #tgt=정답, pre=예측 제공
99
100   # RMSE
101   rmse <- sqrt(mean((tgt_y - pre_y)^2))
102   # MAE
103   mae <- mean(abs(tgt_y - pre_y))
104   # MAPE
105   mape <- 100*mean(abs((tgt_y - pre_y)/tgt_y))
106
107   return(c(rmse, mae, mape))
108
109 }
110
111 perf_mat <- matrix(0, nrow = 2, ncol = 3)
112
113 # Initialize a performance summary
114 rownames(perf_mat) <- c("House Data(full model)", "House Data(7-Var model)")
115 colnames(perf_mat) <- c("RMSE", "MAE", "MAPE")
116
117 # 함수 사용하여 MAE, MAPE, RMSE 계산
118 mlr_df_haty <- predict(mlr_df, newdata = df_val_data)
119 perf_mat[1,] <- perf_eval_reg(df_val_data$price, mlr_df_haty)
120 perf_mat #MAE는 가격 오차, MAPE는 % 오차
```

	RMSE	MAE	MAPE
House Data(full model)	206748.6	125510	25.42285

절대오차는 125510 이며 MAPE 가 $(\text{예측}-\text{실제})/\text{예측}$ 이기 때문에 예측값에 대한 오차의 비율이라고 할 수 있다. 이 경우 25.42%나 된다.

[Q8] 만약 7 개의 입력 변수만을 사용하여 모델을 구축해야 할 경우 어떤 변수들을 선택하겠는가?

[Q4]와 [Q6]의 답변을 바탕으로 본인이 선택한 변수들에 대한 근거를 제시하시오.

Price 와의 correlation 높은 것

```
sqft_living(0.7), grade(0.67), sqft_above(0.6), sqft_living15(0.59), #bathrooms(0.52)
```

Price 와의 correlation 낮은 것

```
long(0.02), yr_built(0.05), zipcode(-0.05), sqft_lot15(0.08), condition(0.04), sqft_lot(0.09)
```

#Price 제외한 두 변수의 상관관계가 높은 것들

```
sqft_living & sqft_above(0.87), sqft_living & sqft_living15(0.76), sqft_living & grade(0.77),  
sqft_living & bathrooms(0.75)
```

#통계적 유의미한 것들

```
bedrooms + bathrooms + sqft_living + waterfront + view + condition + grade + sqft_above +  
yr_built + yr_renovated + zipcode + lat + long + sqft_living15 + sqft_lot15
```

-> 통계적으로 유의미한 변수는 15 개이며, 그 중 8 개를 제거해야한다. Price 와의 correlation 이 낮은 변수들을 제거해준다.

```
125 install.packages("car")
126 library(car)
127
128 mlr2_df <- lm(price ~ bedrooms + bathrooms + sqft_living + waterfront + view + grade +
+ sqft_above + yr_renovated + lat + sqft_living15, data = df_trn_data)
129 summary(mlr2_df)
130 plot(mlr2_df)
131 vif(mlr2_df)
```

```
bedrooms + bathrooms + sqft_living + waterfront + view + grade + sqft_above + yr_renovated +
lat + sqft_living15 #10 개
```

```
lat           6.505e+05  1.263e+04  51.487  < 2e-16 ***
sqft_living15 1.047e+01  4.131e+00   2.534  0.011292 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 211100 on 15117 degrees of freedom
Multiple R-squared:  0.6655,    Adjusted R-squared:  0.6653
F-statistic:  3008 on 10 and 15117 DF,  p-value: < 2.2e-16
```

```
> vif(mlr3_df)
      bedrooms   bathrooms   waterfront   sqft_living      view      grade   sqft_above   yr_renovated
1.674407     2.528315     1.195847     6.911765     1.362994    2.896089     5.074156     1.017178
      lat
1.041646
```

-> 10 개로 모델 수립한 경우 sqft_living15 이 통계적으로 유의미하지 않기 때문에 제거해준다. 이 중 Price 제외 두 변수의 상관관계가 제일높은 두개의 변수가 들어가있다. 실제로 확인해볼 때 VIF 를 계산한 경우 5 가 넘는 sqft_living, sqft_above 중 price 와 상관관계가 상대적으로 낮은 sqft_above 를 제거해준다.

```
133 mlr3_df <- lm(price ~ bedrooms + bathrooms + waterfront + sqft_living + view +
grade + yr_renovated + lat , data = df_trn_data)
134 summary(mlr3_df)
135 plot(mlr3_df)
136 vif(mlr3_df)
```

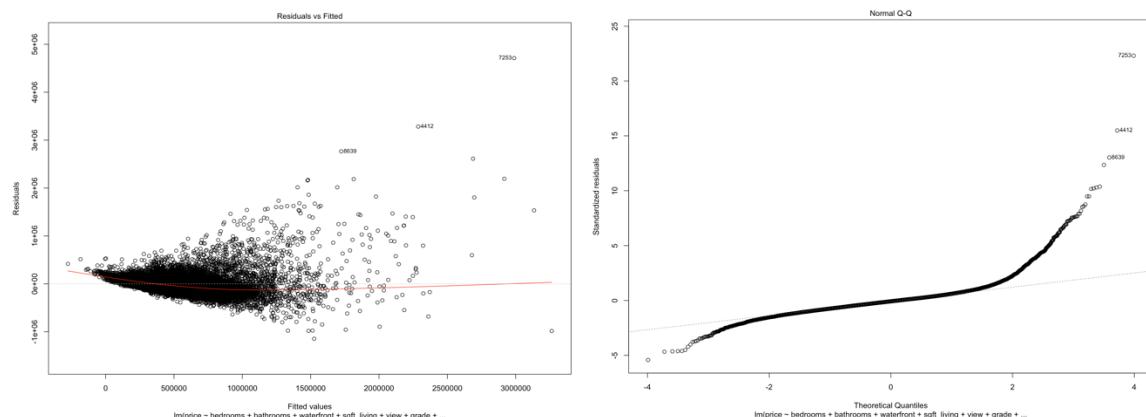
bedrooms + bathrooms + waterfront + sqft_living + view + grade + yr_renovated + lat #8 개

-> 이 중 yr_renovated 가 상관관계가 0.13 으로 가장 작으므로 제거해준다.

bedrooms + bathrooms + waterfront + sqft_living + view + grade + lat #7 개

[Q9] [Q8]에서 선택한 변수들만을 사용하여 MLR 모형을 다시 학습하고 Adjusted R2, Test 데이터셋에 대한 MAE, MAPE, RMSE 를 산출한 뒤, 두 모형(모든 변수 사용 vs. 7 개 변수 선택)을 비교해 보시오.

```
140 mlr4_df <- lm(price ~ bedrooms + bathrooms + waterfront + sqft_living + view + grade + lat ,
, data = df_trn_data)
141 summary(mlr4_df)
142 plot(mlr4_df)
143 vif(mlr4_df)
144
145 #함수 사용하여 MAE, MAPE, RMSE 계산
146 mlr4_df_haty <- predict(mlr4_df, newdata = df_val_data)
147 perf_mat[2,] <- perf_eval_reg(df_val_data$price, mlr4_df_haty)
148 perf_mat #MAE는 절대적 오차, MAPE는 % 오차
```



```
> summary(mlr4_df)

Call:
lm(formula = price ~ bedrooms + bathrooms + waterfront + sqft_living +
    view + grade + lat, data = df_trn_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-1145852 -109262 -15842   76007  4711529 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -3.188e+07  5.970e+05 -53.404 < 2e-16 ***
bedrooms    -2.476e+04  2.462e+03 -10.058 < 2e-16 ***
bathrooms   -1.240e+04  3.557e+03 -3.488 0.000489 *** 
waterfront   6.333e+05  2.171e+04  29.170 < 2e-16 *** 
sqft_living  1.934e+02  3.732e+00  51.831 < 2e-16 *** 
view        6.744e+04  2.546e+03  26.484 < 2e-16 *** 
grade       8.080e+04  2.380e+03  33.950 < 2e-16 *** 
lat         6.620e+05  1.258e+04  52.631 < 2e-16 *** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 212500 on 15120 degrees of freedom
Multiple R-squared:  0.6612, Adjusted R-squared:  0.661 
F-statistic: 4215 on 7 and 15120 DF, p-value: < 2.2e-16
```

```
> vif(mlr4_df)
bedrooms    bathrooms  waterfront sqft_living      view      grade      lat
1.668427    2.525350    1.191929    3.971257    1.293900    2.646302  1.019471
```

	RMSE	MAE	MAPE
House Data(full model)	206748.6	125510.0	25.42285
House Data(7-Var model)	221516.6	134723.8	26.19063

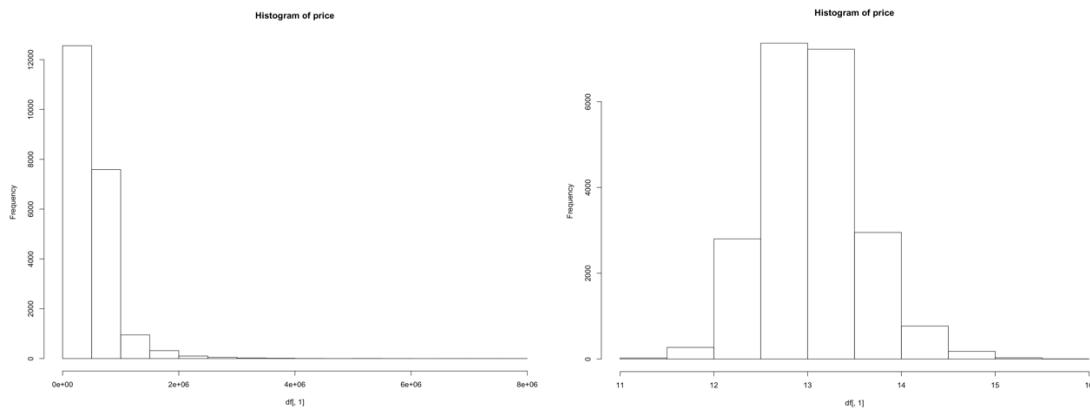
full 모델에 비해 7-Var 모델의 절대적 오차와 MAPE 가 모두 증가하였다.

adjusted R 도 0.7035->0.661로 줄었다.

[Extra Question] 이 외 해당 데이터셋을 통해 MLR 관점에서 가능한 추가적인 분석을 웹에서 검색해서 수행하고 그 결과를 해석해 보시오.

```
151 hist(df[,1], main = paste("Histogram of" , colnames(df[1])))
152 df$price <- log(df$price)
153 hist(df[,1], main = paste("Histogram of" , colnames(df[1])))
154
155 skewness(df[,1])
156 kurtosis(df[,1])
---
```

Price 의 그래프를 그려보면 한쪽에 치우쳐져 있기 때문에 log 변환을 해준다.



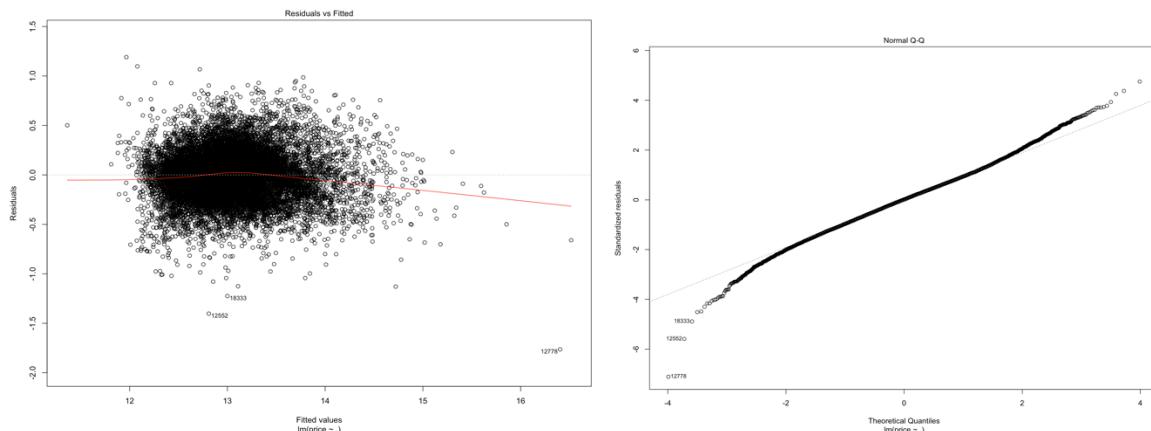
전 후의 다이어그램

```
> skewness(df[,1])
[1] 0.4281391
> kurtosis(df[,1])
[1] 3.691277
```

--> 정규분포에 가깝다.

$\log(\text{Price})$ 를 종속변수로 하여 모델을 구성해보았다.

```
162 # Split the data into the training/validation sets
163
164 nR <- nrow(df) #행의 갯수 계산하는 명령어
165 nC <- ncol(df) #열의 갯수 계산하는 명령어
166
167 set.seed(12345) #랜덤 알고리즘 지정
168 df_trn_idx <- sample(1:nR, round(0.7*nR)) #70%를 샘플링
169 df_trn_data <- df[df_trn_idx,]
170 df_val_data <- df[-df_trn_idx,] #training 아닌 것들
171
172 mlr5_df <- lm(price ~ ., data = df_trn_data)
173 summary(mlr5_df)
174 plot(mlr5_df)
```



데이터가 가격을 종속변수로 했을 때보다 \log 를 취했을 때 훨씬 잘 fitting 된 모습을 보인다.

```

Call:
lm(formula = price ~ ., data = df_trn_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.76384 -0.16170  0.00377  0.15965  1.19066 

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -5.066e+00  4.348e+00 -1.165 0.243958  
bedrooms     -1.131e-02  2.951e-03 -3.834 0.000127 *** 
bathrooms     7.299e-02  4.831e-03 15.108 < 2e-16 *** 
sqft_living   1.485e-04  6.540e-06 22.701 < 2e-16 *** 
sqft_lot      4.705e-07  6.873e-08  6.847 7.86e-12 *** 
floors        7.146e-02  5.351e-03 13.354 < 2e-16 *** 
waterfront    3.836e-01  2.571e-02 14.917 < 2e-16 *** 
view          6.087e-02  3.167e-03 19.217 < 2e-16 *** 
condition     5.983e-02  3.530e-03 16.947 < 2e-16 *** 
grade         1.590e-01  3.194e-03 49.769 < 2e-16 *** 
sqft_above     -1.610e-05  6.472e-06 -2.488 0.012868 *  
sqft_basement NA       NA       NA       NA      
yr_built      -3.427e-03  1.079e-04 -31.765 < 2e-16 *** 
yr_renovated  3.705e-05  5.418e-06  6.839 8.30e-12 *** 
zipcode       -6.503e-04  4.911e-05 -13.241 < 2e-16 *** 
lat           1.388e+00  1.600e-02  86.768 < 2e-16 *** 
long          -1.677e-01  1.957e-02 -8.571 < 2e-16 *** 
sqft_living15 9.802e-05  5.132e-06 19.100 < 2e-16 *** 
sqft_lot15    -2.735e-07  1.091e-07 -2.507 0.012189 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.251 on 15110 degrees of freedom
Multiple R-squared:  0.7723,    Adjusted R-squared:  0.772 
F-statistic:  3014 on 17 and 15110 DF,  p-value: < 2.2e-16

```

Adjusted R² 값도 0.772로 가격을 종속변수로 했을 때의 값(0.705)보다 높게 나왔다.

	RMSE	MAE	MAPE
House Data(full model)	0.2557568	0.196759	1.506994

**모델선택

#통계적 유의미한 것들

bedrooms + bathrooms + sqft_living + sqft_lot + floors + waterfront + view + condition + grade + yr_built + yr_renovated + zipcode + lat + long + sqft_living15 #15 개

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement	yr_built	yr_renovate	zipcode	lat	long	sqft_living15	sqft_lot15
price	1	0.35	0.55	0.7	0.1	0.31	0.17	0.35	0.04	0.7	0.6	0.32	0.08	0.11	-0.04	0.45	0.05	0.62	0.09
bedrooms	0.35	1	0.53	0.59	0.03	0.18	-0.01	0.08	0.03	0.37	0.49	0.31	0.16	0.02	-0.16	-0.01	0.13	0.4	0.03
bathrooms	0.55	0.53	1	0.75	0.09	0.5	0.06	0.19	-0.12	0.67	0.69	0.28	0.51	0.05	-0.2	0.02	0.22	0.57	0.09
sqft_living	0.7	0.59	0.75	1	0.17	0.35	0.1	0.28	-0.06	0.76	0.88	0.44	0.32	0.06	-0.2	0.05	0.24	0.76	0.18

-> price 와 상관관계가 적은 condition, zipcode, long, yr_built 을 제거한다.

bedrooms + bathrooms + sqft_living + sqft_lot + floors + waterfront + view + grade + yr_renovated + lat + sqft_living15 #11 개

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-5.915e+01	7.611e-01	-77.717	< 2e-16 ***
bedrooms	4.562e-03	3.140e-03	1.453	0.146318
bathrooms	1.812e-02	4.806e-03	3.771	0.000163 ***
sqft_living	1.684e-04	5.258e-06	32.017	< 2e-16 ***
sqft_lot	2.599e-07	5.348e-08	4.860	1.18e-06 ***
floors	-3.340e-03	4.893e-03	-0.683	0.494800
waterfront	3.905e-01	2.756e-02	14.171	< 2e-16 ***
view	8.029e-02	3.258e-03	24.648	< 2e-16 ***
grade	1.376e-01	3.294e-03	41.778	< 2e-16 ***
yr_renovated	8.438e-05	5.487e-06	15.378	< 2e-16 ***
lat	1.484e+00	1.604e-02	92.515	< 2e-16 ***
sqft_living15	7.488e-05	5.216e-06	14.356	< 2e-16 ***

-> bathrooms, floors 제거

bathrooms + sqft_living + sqft_lot + waterfront + view + grade + yr_renovated + lat + sqft_living15 #9 개

```
lm(formula = price ~ bathrooms + sqft_living + sqft_lot + waterfront +
   view + grade + yr_renovated + lat + sqft_living15, data = df_trn_data)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.81329	-0.17052	-0.00866	0.16391	1.12722

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-5.910e+01	7.603e-01	-77.725	< 2e-16 ***
bathrooms	1.830e-02	4.448e-03	4.114	3.92e-05 ***
sqft_living	1.716e-04	4.848e-06	35.398	< 2e-16 ***
sqft_lot	2.556e-07	5.323e-08	4.802	1.59e-06 ***
waterfront	3.881e-01	2.752e-02	14.103	< 2e-16 ***
view	8.020e-02	3.240e-03	24.752	< 2e-16 ***
grade	1.362e-01	3.135e-03	43.435	< 2e-16 ***
yr_renovated	8.417e-05	5.485e-06	15.344	< 2e-16 ***
lat	1.483e+00	1.603e-02	92.512	< 2e-16 ***
sqft_living15	7.505e-05	5.205e-06	14.418	< 2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2693 on 15118 degrees of freedom

Multiple R-squared: 0.7377, Adjusted R-squared: 0.7376

F-statistic: 4725 on 9 and 15118 DF, p-value: < 2.2e-16

```
> vif(mlr7_df)
   bathrooms    sqft_living     sqft_lot     waterfront       view       grade
      2.457616     4.170581     1.048224     1.191523     1.303759     2.857448
  yr_renovated      lat sqft_living15
      1.019897     1.030012     2.682460
```

Adjusted R-squared 값은 0.7376, VIF 도 모두 5 보다 작다.

```
#함수 사용하여 MAE, MAPE, RMSE 계산
mlr7_df_haty <- predict(mlr7_df, newdata = df_val_data)
perf_mat2[2,] <- perf_eval_reg(df_val_data$price, mlr7_df_haty)
perf_mat2
```

	RMSE	MAE	MAPE
House Data(full model)	0.2557568	0.1967590	1.506994
House Data(9-Var model)	0.2739279	0.2101528	1.606216

full 모델이 MAE, MAPE 가 더 적다.

추가 search 를 통하여 변수가 정규분포가 아닐 때 함수를 취해주어 정규분포의 형태를 띄게 만드는 것을 배웠다.