Homework 1

Yigit Memceroktay

2024-04-06

Introduction

Author: Yiğit Memceroktay

Number:2020402006

Course: IE360

The data-sets that I chose for the analysis are: CPI of Turkey (consumer price index), Unemployment rate and total house sales in Turkey. The data-sets contain data for the dates between 1/1/2015-1/12/2021. Importing packages

```
require(data.table)
require(lubridate)
require(forecast)
require(repr)
require(openxlsx)
require(ggplot2)
require(data.table)
require(skimr)
require(GGally)
require(ggcorrplot)
require(forecast)
require(tidyverse)
require(htmltools)
require(readxl)
require(corrplot)

library(reshape2)
library(corrplot)
library(readxl)
library(tidyverse)
library(data.table)
library(lubridate)
library(forecast)
library(skimr)
library(repr)
```

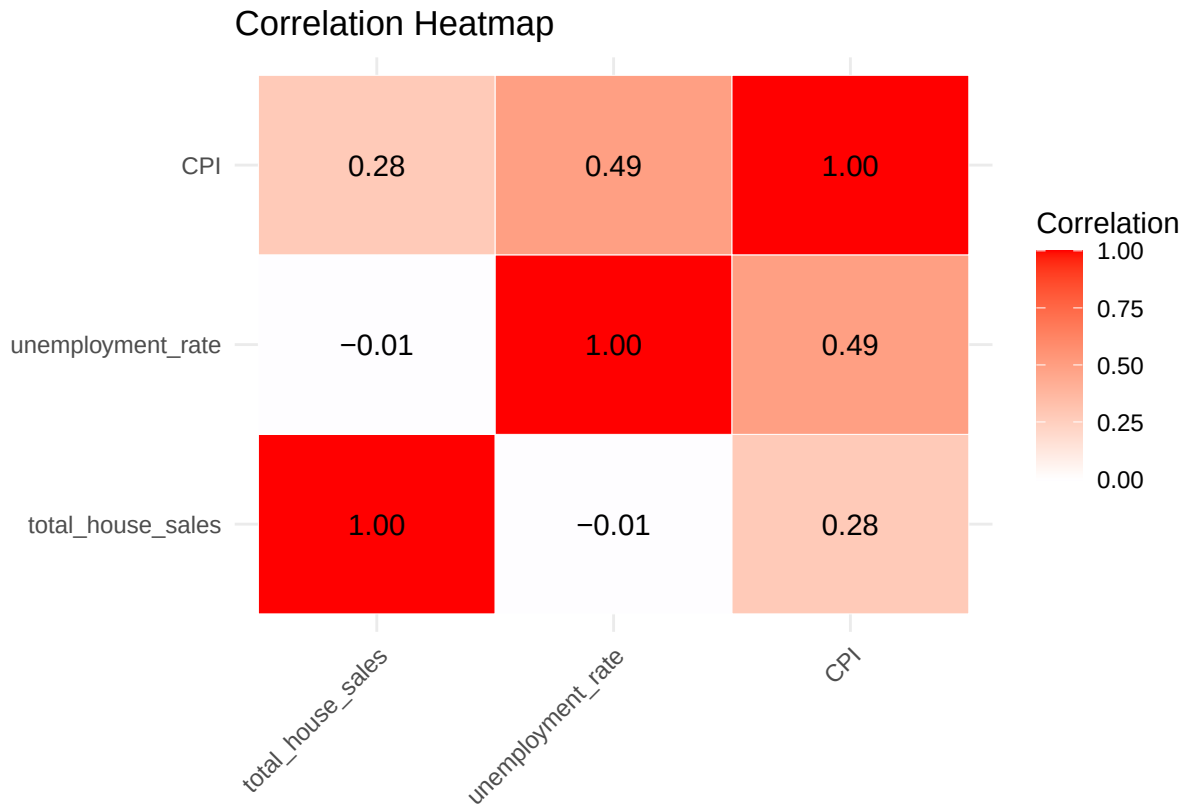
Imported the excel file that contains 3 data-sets for time-series analysis. The data-sets that I chose are: CPI (consumer price index), unemployment rate, and total amount of houses sold. The table below shows the correlation between the data-sets. As you can see, no two pair has correlation more than 0.5 therefore the data-sets are suitable for this project.

```
options(repr.plot.width=12.7, repr.plot.height=8)

data_path='C:\\Users\\Yalikent\\Downloads\\forecast.xlsx'
forecast_data = read_excel(data_path, sheet="EVDS")
forecast_data_table =  as.data.table(forecast_data)
cor_matrix = cor(forecast_data_table)
melted_cor_matrix <- melt(cor_matrix)
```

Correlation Table

```
# Use ggplot to create the heatmap
ggplot(melted_cor_matrix, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = '', y = '', title = 'Correlation Heatmap', fill = 'Correlation') +
  geom_text(aes(label = sprintf("%.2f", value)), color = "black", size = 4)
```



Importing the excel file that contains both the forecast variables (CPI, etc.) and predictor variables (number of searches on google, consumer confidence index etc.)

```
data_path='C:\\Users\\Yalikent\\Downloads\\Book2.xlsx'
all_data <- read_excel(data_path)
```

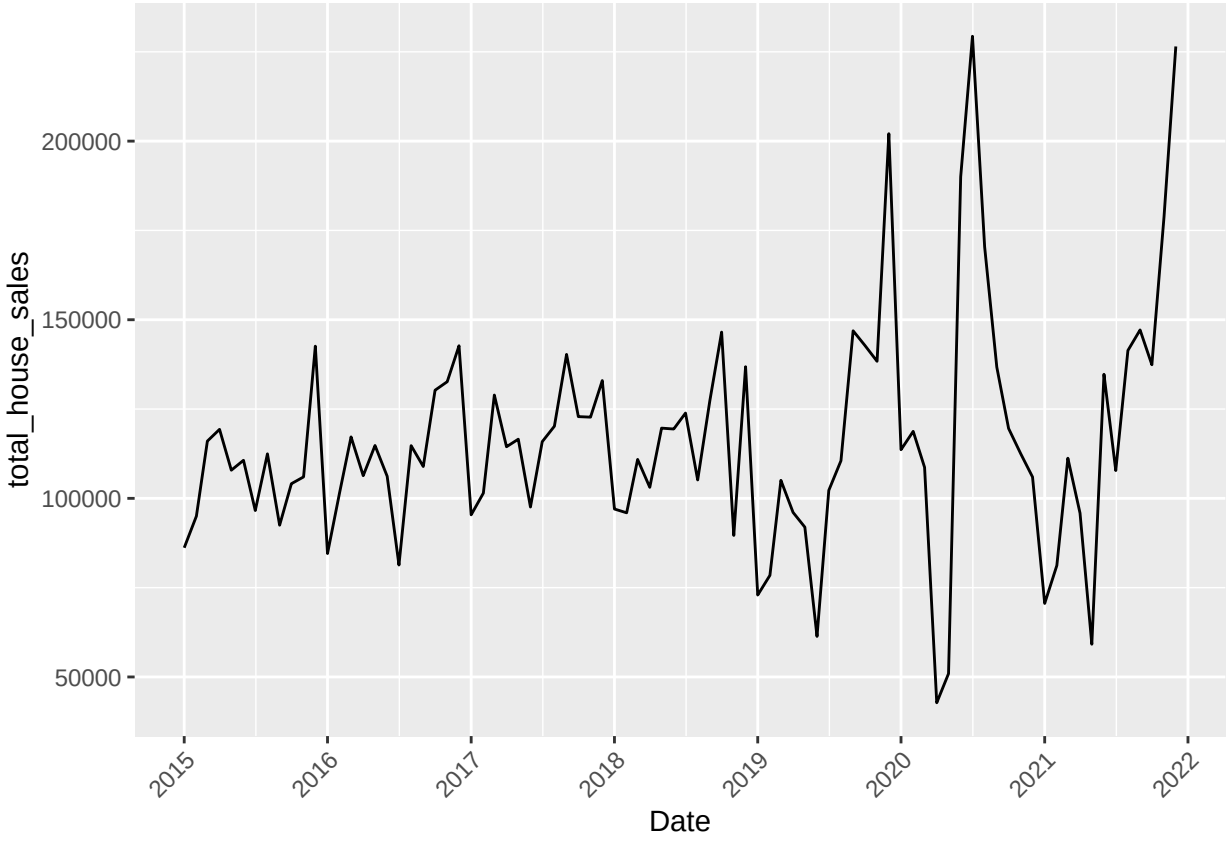
```
## New names:
## * `TP YISGUCU2 G7` -> `TP YISGUCU2 G7...2`
## * `TP FG J0` -> `TP FG J0...5`
## * `TP YISGUCU2 G7` -> `TP YISGUCU2 G7...11`
## * `TP FG J0` -> `TP FG J0...12`
```

```
all_data_table=data.table(all_data)
```

Model 1 (Total House Sales in Turkey)

In the initial model, my objective is to examine the data-set representing the total number of house sales in Turkey. The subsequent graph illustrates the overall house sales spanning from 2015 to 2021 in Turkey.

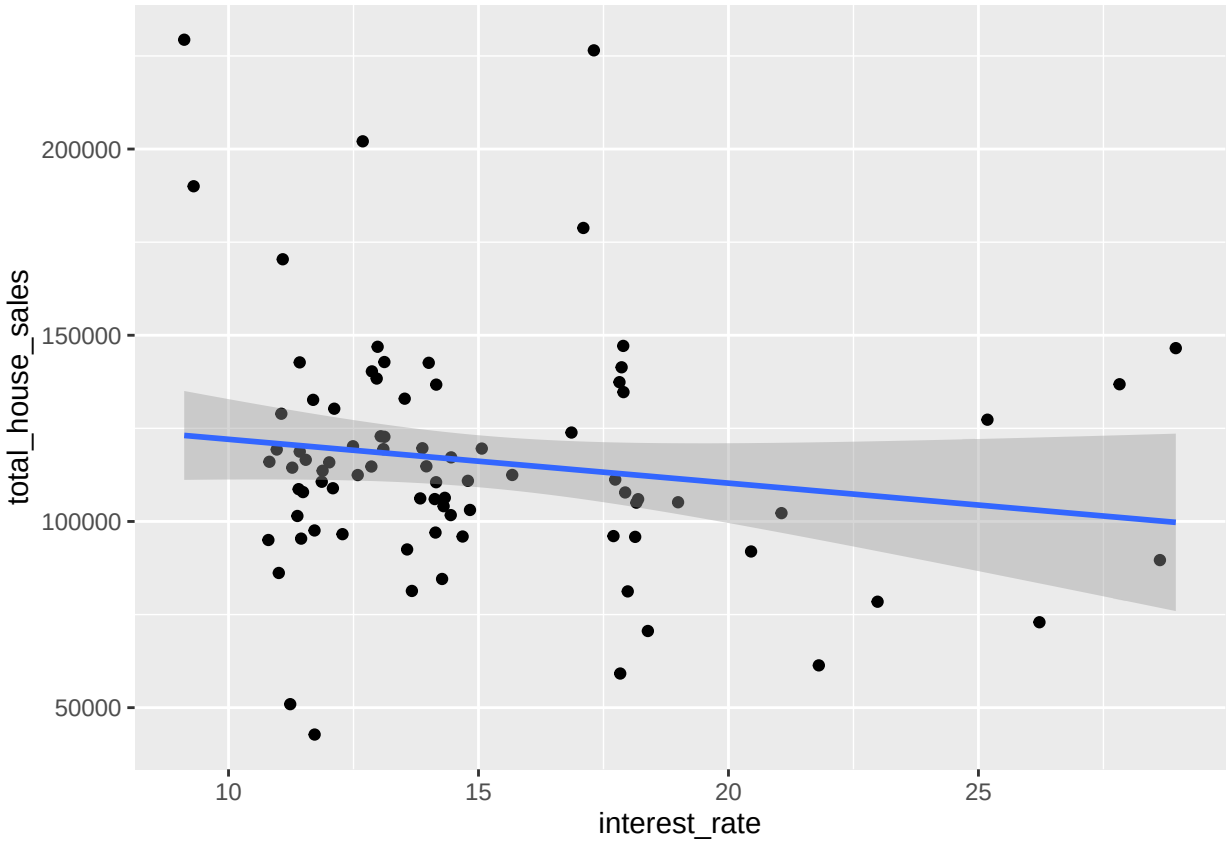
```
all_data_table[, Date := as.Date(Date, format = "%Y-%m-%d")]
ggplot(all_data_table ,aes(x=Date,y=total_house_sales, group=1)) + geom_line() + scale_x_date(date_breal
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



I decided to employ the interest rate and the volume of Google searches for houses to forecast the total house sales in Turkey. Below, the scatter plot is presented with a linear regression line depicting the relationship between interest rate and Google searches.

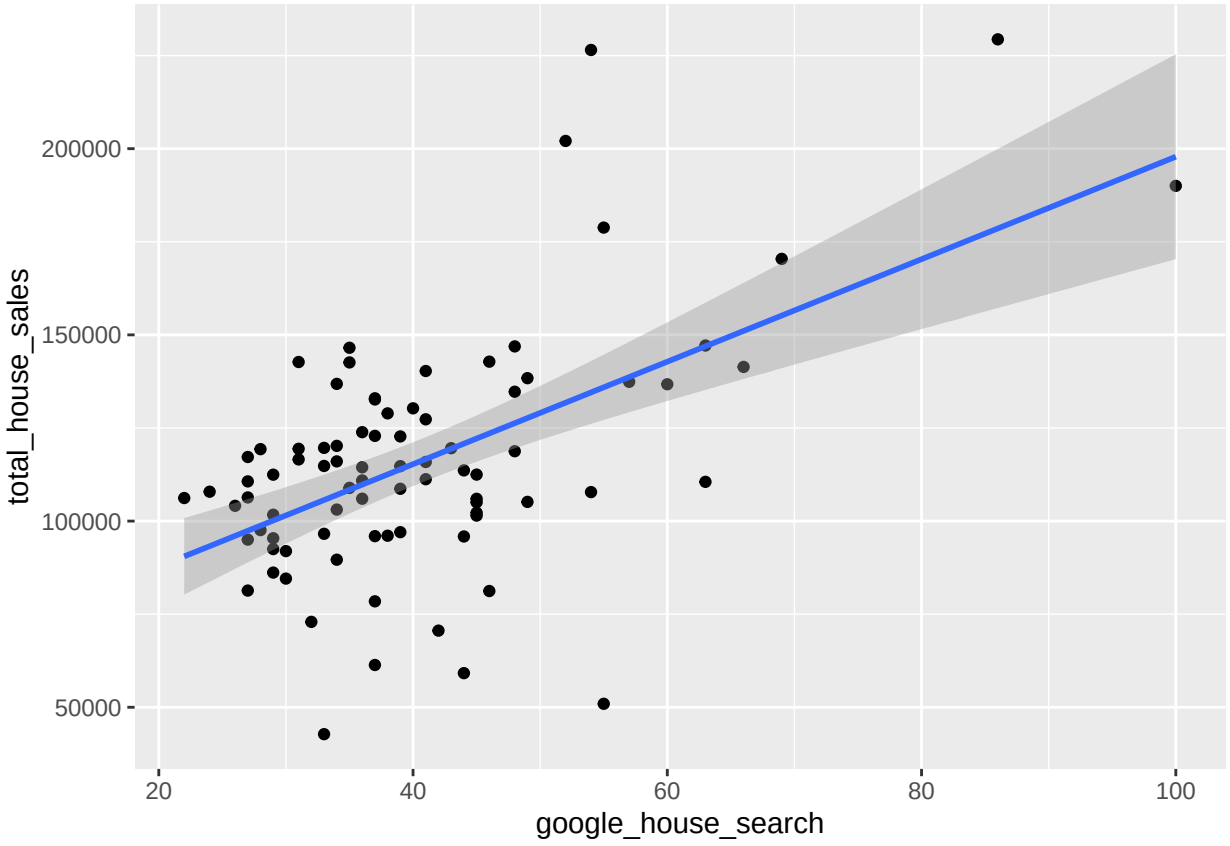
```
ggplot(all_data_table ,aes(x=interest_rate,y=total_house_sales)) + geom_point() +geom_smooth(method='lm
```

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




```
ggplot(all_data_table ,aes(x= google_house_search,y=total_house_sales)) + geom_point() +geom_smooth(metl
```

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



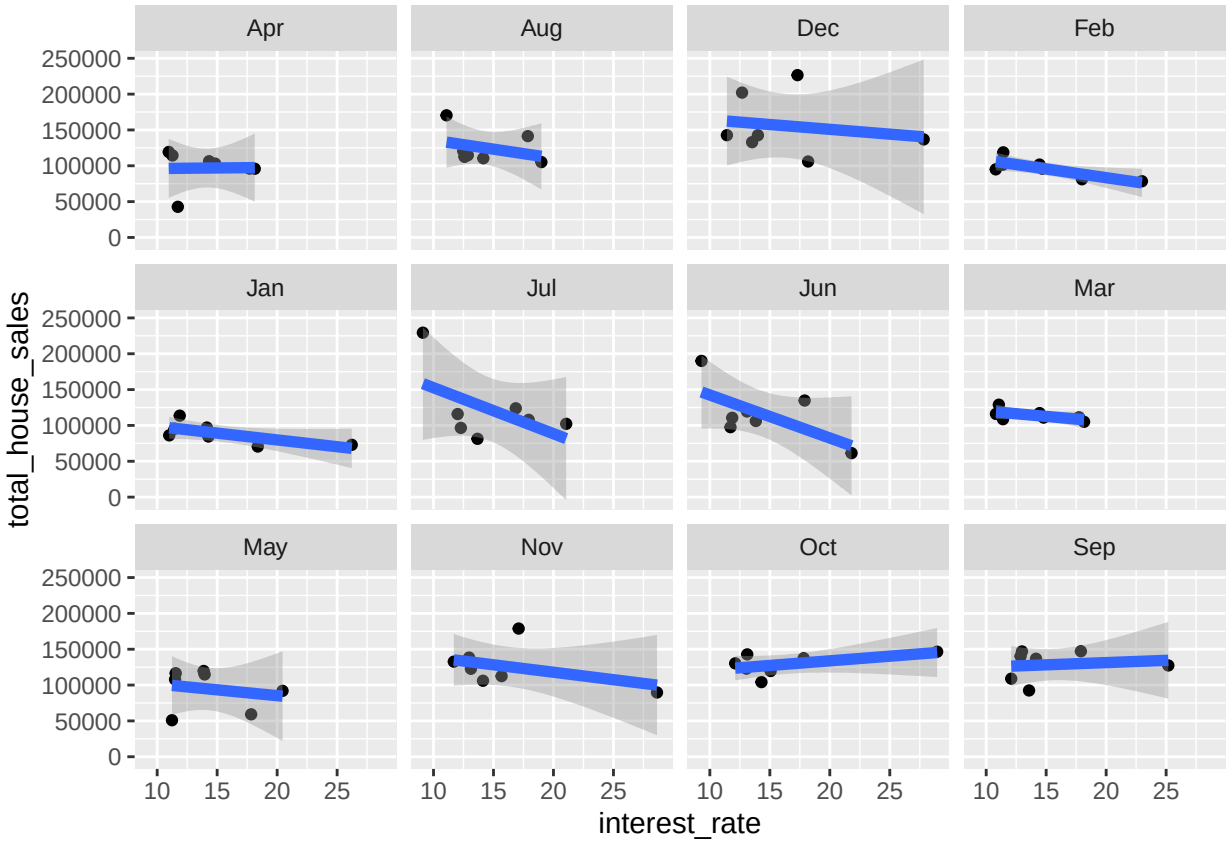
The functions add new columns to **all_data_table**: **trnd** assigns sequential numbers to rows, **month** and **year** extract and store the month and year from a **Date** column as character strings, aiding in trend and temporal analysis.

```
all_data_table[,trnd:=1:.N]
all_data_table[,month:=as.character(month(Date,label=T))]
all_data_table[,year:=as.character(year(Date))]
```

Below you can see the correlation between the data-sets for each month.

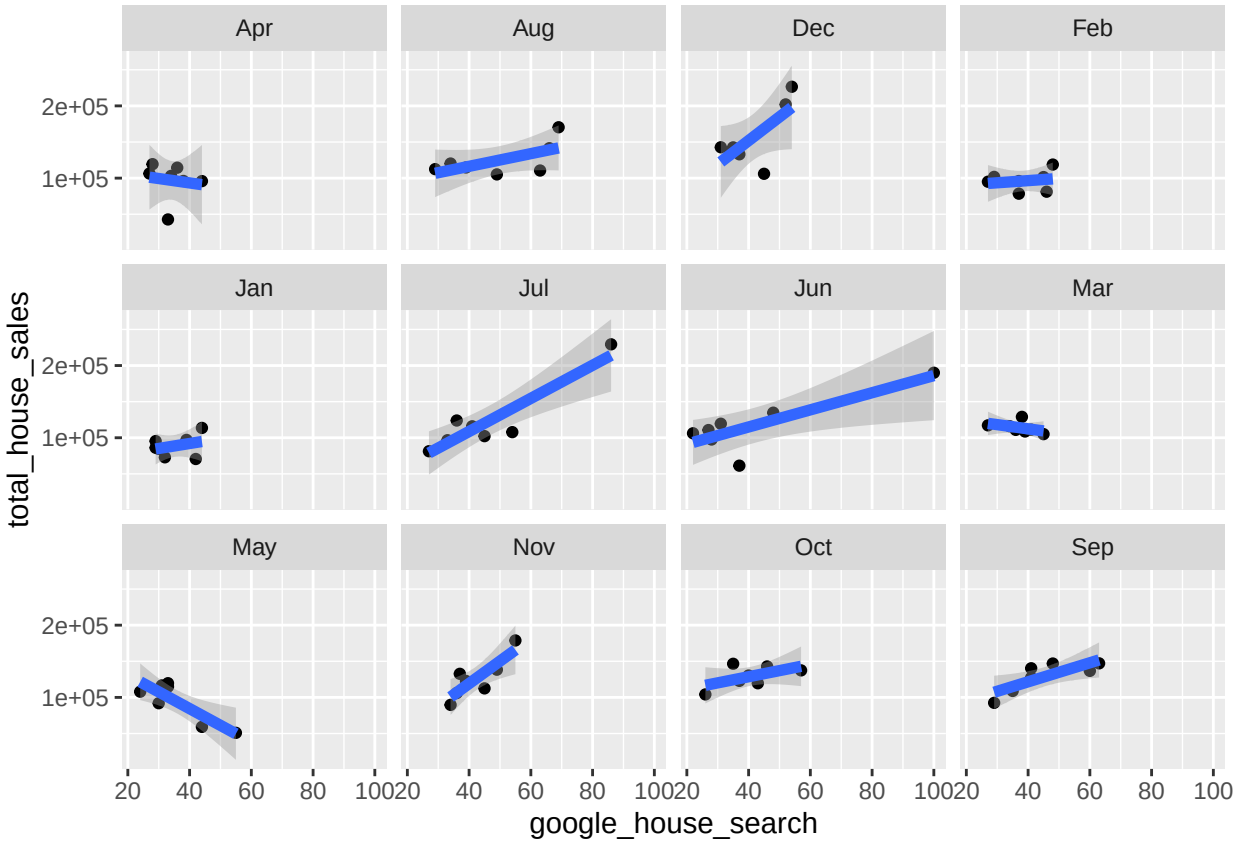
```
ggplot(all_data_table,aes(x=interest_rate,y=total_house_sales)) +
        geom_point() + geom_smooth(method=lm,linewidth=2) + facet_wrap(~month,drop = TRUE)
```

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



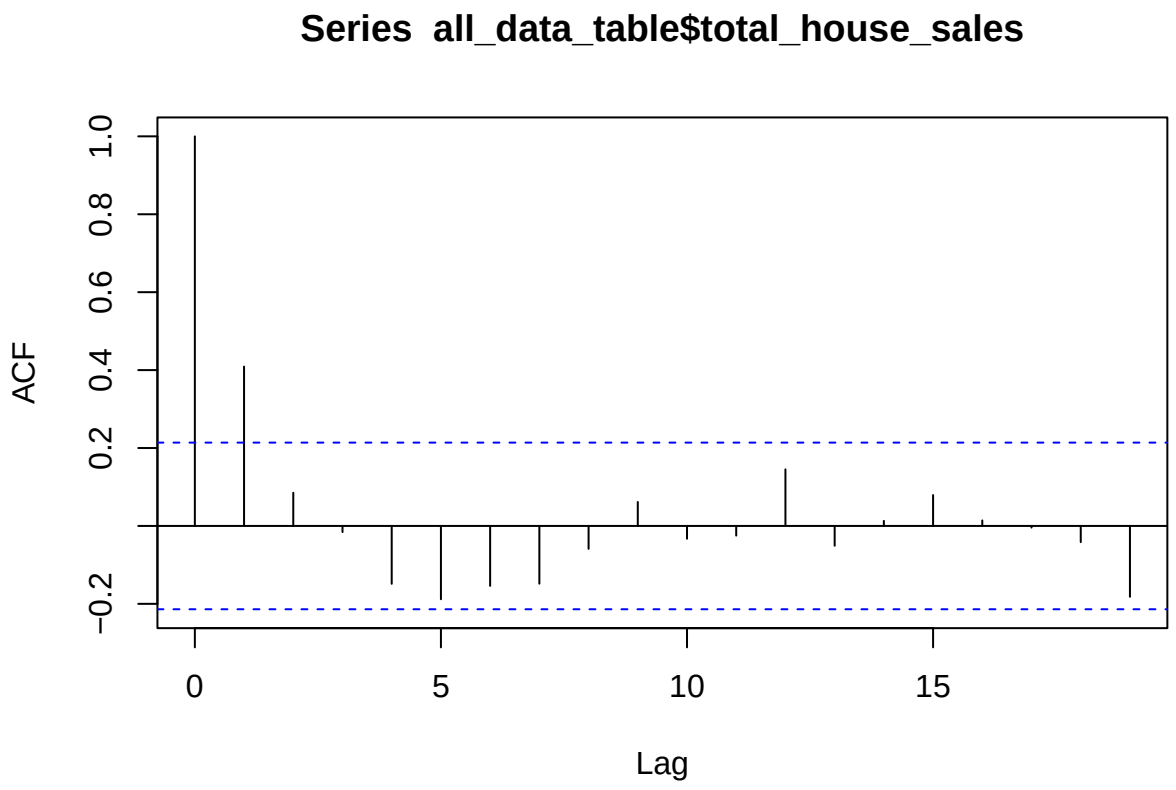

```
ggplot(all_data_table,aes(x=google_house_search,y=total_house_sales)) +
        geom_point() + geom_smooth(method=lm,linewidth=2) + facet_wrap(~month)
```

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



The graph below shows the acf plot for total_house_sales the data-set. This plot shows whether the data is correlated within itself.

```
acf(all_data_table$total_house_sales)
```

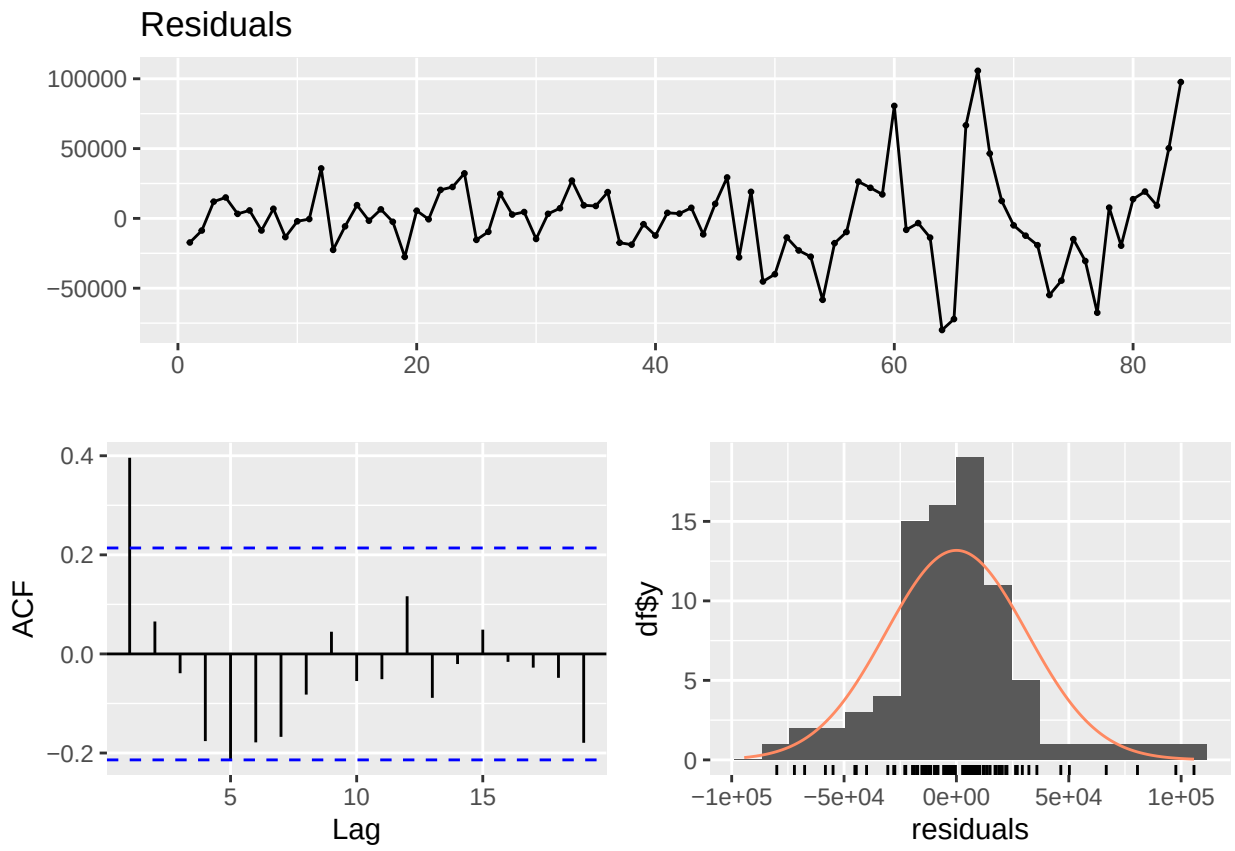


Lets's start building the model. The linear regression analysis, with total house sales as the response variable and a trend indicator as the predictor, shows a positive trend (coefficient = 306.5, p-value = 0.0345), indicating a slight increase in house sales over time. However, the model explains only a small portion of the variance (R-squared = 0.053), suggesting other factors also influence hous

```
model=lm(total_house_sales~trnd,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = total_house_sales ~ trnd, data = all_data_table)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -79942 -14987   -551  12808 105713
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 103110.6     6972.7  14.788   <2e-16 ***
## trnd           306.5      142.5   2.151   0.0345 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31670 on 82 degrees of freedom
## Multiple R-squared:  0.05339,    Adjusted R-squared:  0.04185
## F-statistic: 4.625 on 1 and 82 DF,  p-value: 0.03445
```

```
checkresiduals(model$residuals)
```

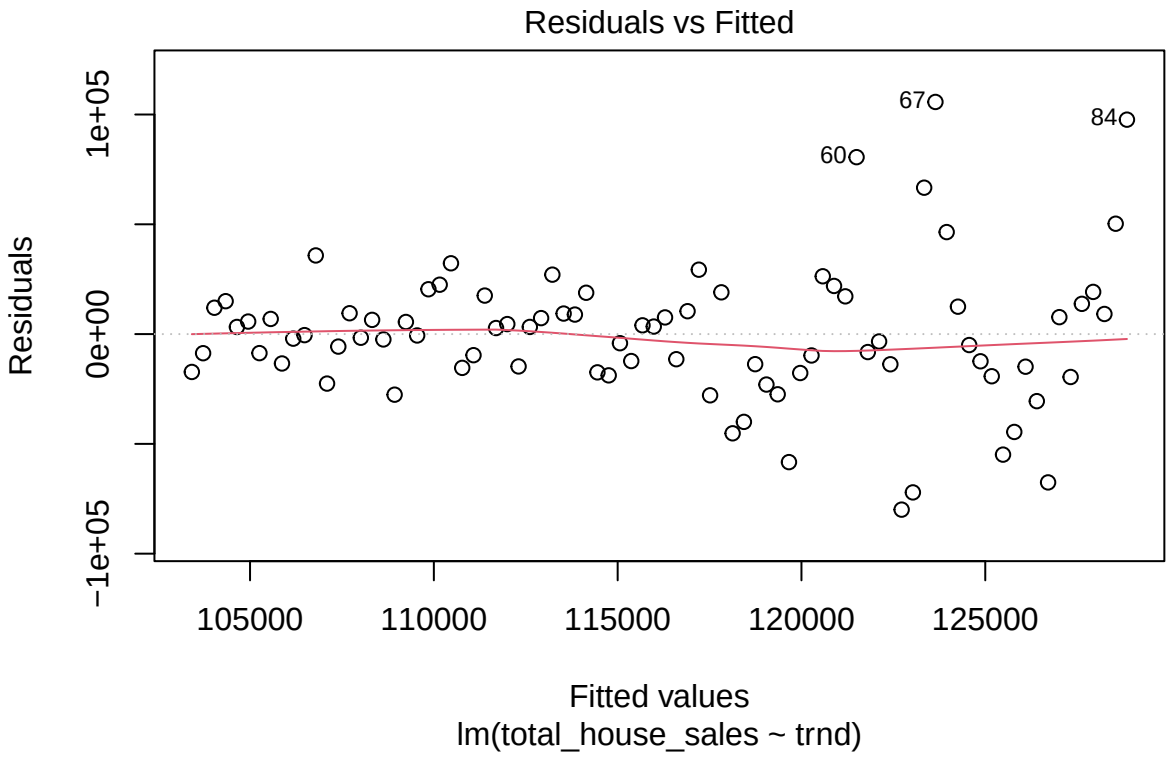


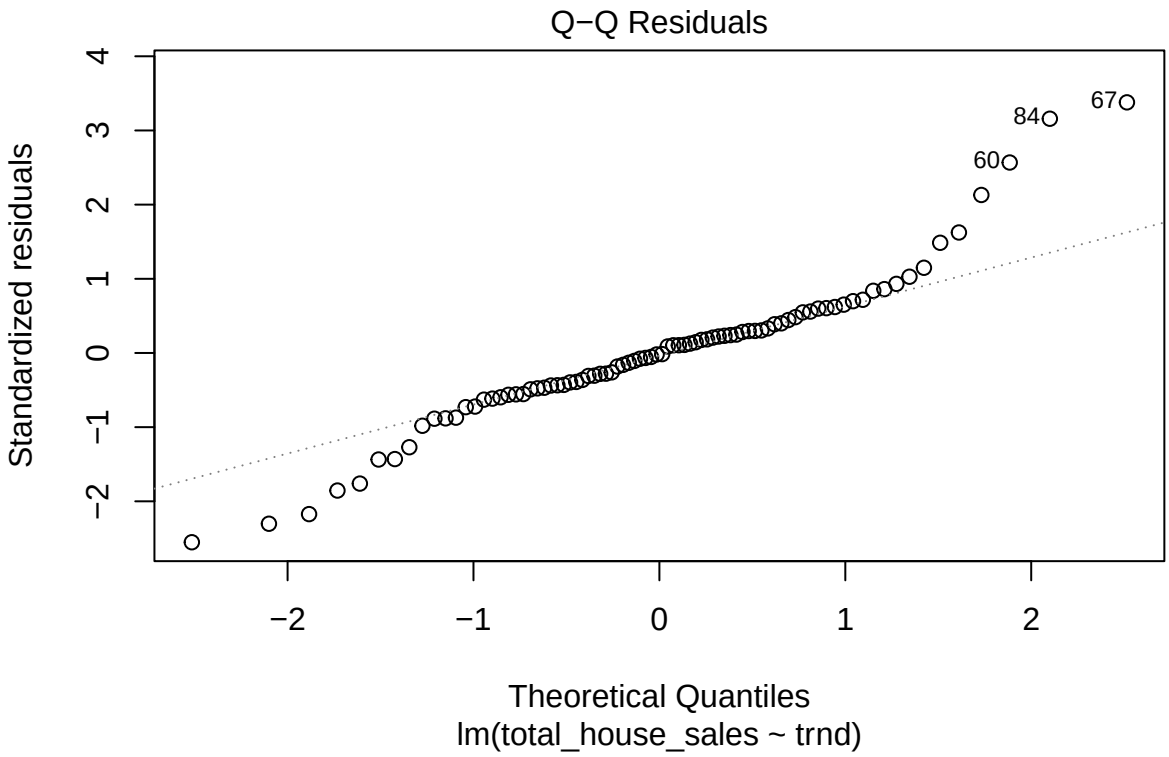

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 27.818, df = 10, p-value = 0.001931
##
## Model df: 0.   Total lags used: 10
```

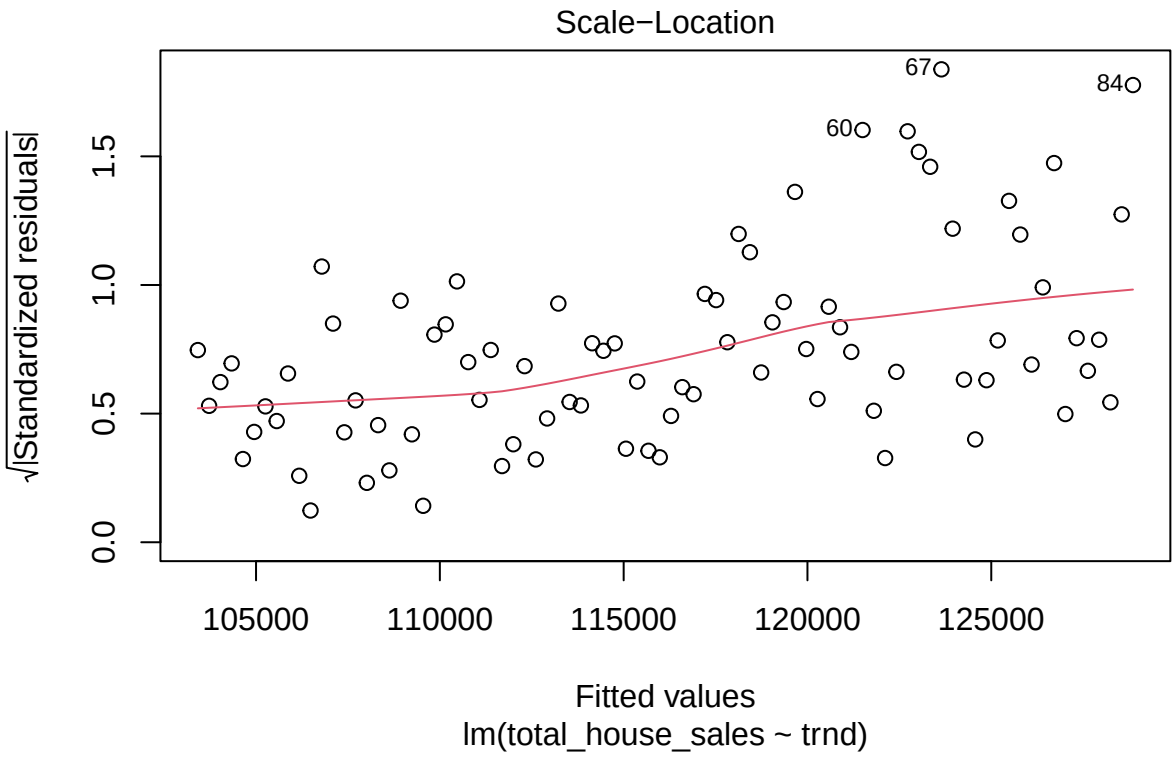
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend:=predict(model,tmp_data)]
tmp_data[,residual_trend:=actual-predicted_trend]
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend,color='predicted'))
```

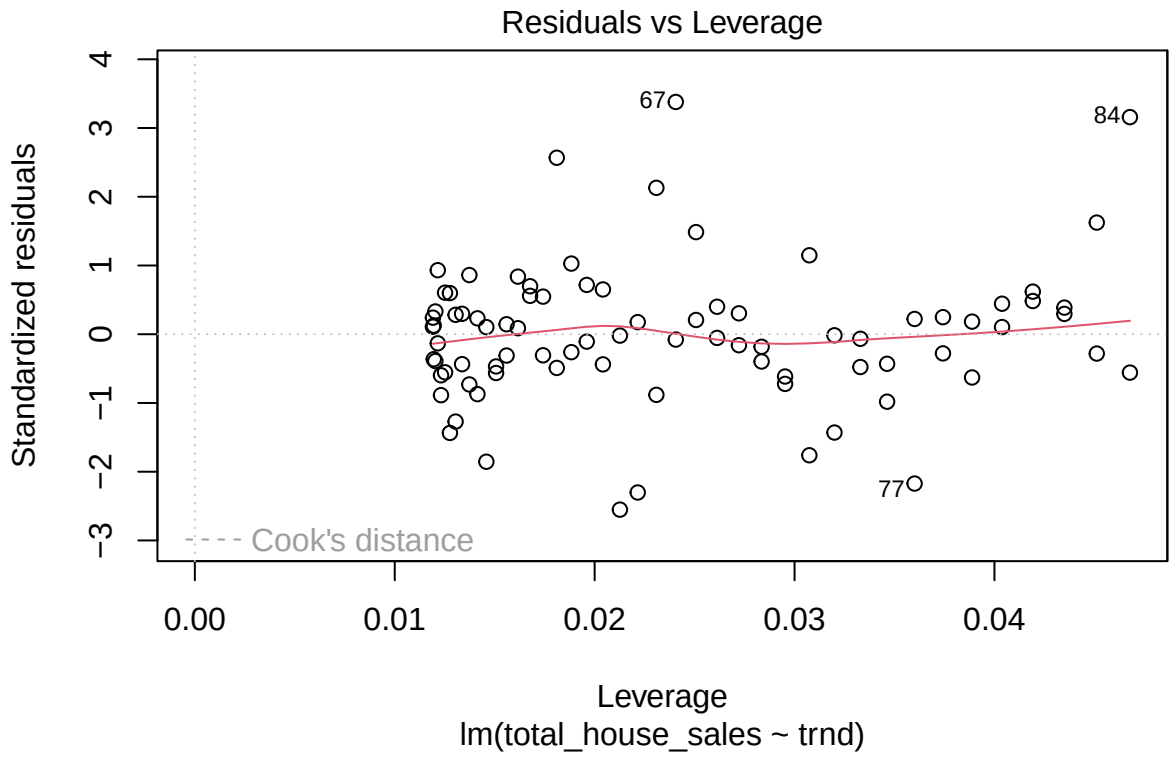

200000
150000
100000
50000
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
1e+05
0e+00
−1e+05
67
84
60
105000
110000
115000
120000
125000
Fitted values
lm(total_house_sales ~ trnd)


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(total_house_sales ~ trnd)
67
84
60
−2
−1
0
1
2
3
4


Scale−Location
67
84
60
√|Standardized residuals|
0.0
0.5
1.0
1.5
105000
110000
115000
120000
125000
Fitted values
lm(total_house_sales ~ trnd)



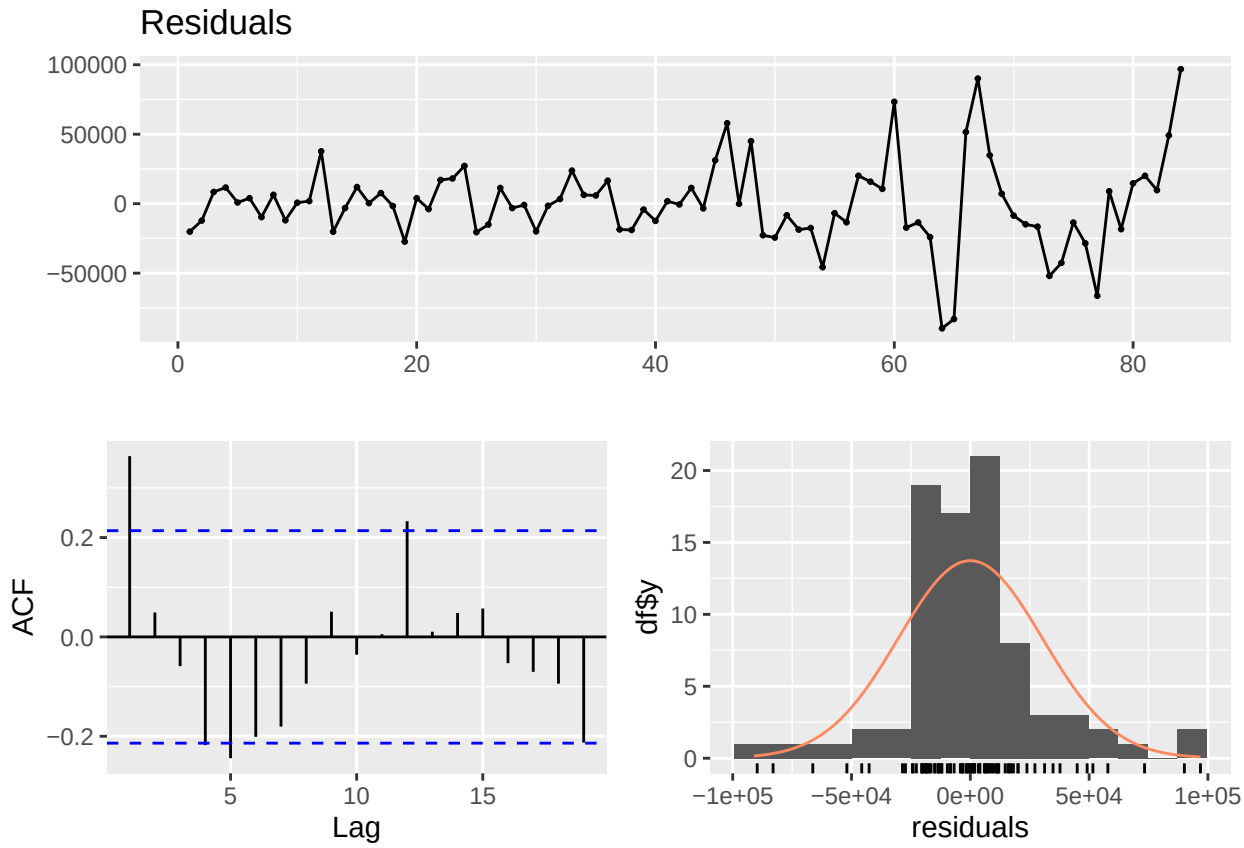
For next step I added the interest rate factor to the model. In this linear regression model, total house sales are predicted by trend and interest rate, revealing a positive association with the trend (coefficient = 439.9, p-value = 0.00399) and a negative relationship with the interest rate (coefficient = -2086.5, p-value = 0.01552). The model has an improved fit compared to the previous one, with an R-squared value of 0.1198, indicating that these factors together account for approximately 12% of the variance in house sales.

```
model=lm(total_house_sales~trnd+interest_rate,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = total_house_sales ~ trnd + interest_rate, data = all_data_table)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -89716 -16711   -727  11410  96860
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   128804.7    12401.0  10.387  < 2e-16 ***
## trnd             439.9      148.4   2.964  0.00399 **
## interest_rate  -2086.5      844.0  -2.472  0.01552 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30730 on 81 degrees of freedom
```

```
## Multiple R-squared:  0.1198, Adjusted R-squared:  0.09807
## F-statistic: 5.513 on 2 and 81 DF,  p-value: 0.005694
```

```
checkresiduals(model$residuals)
```

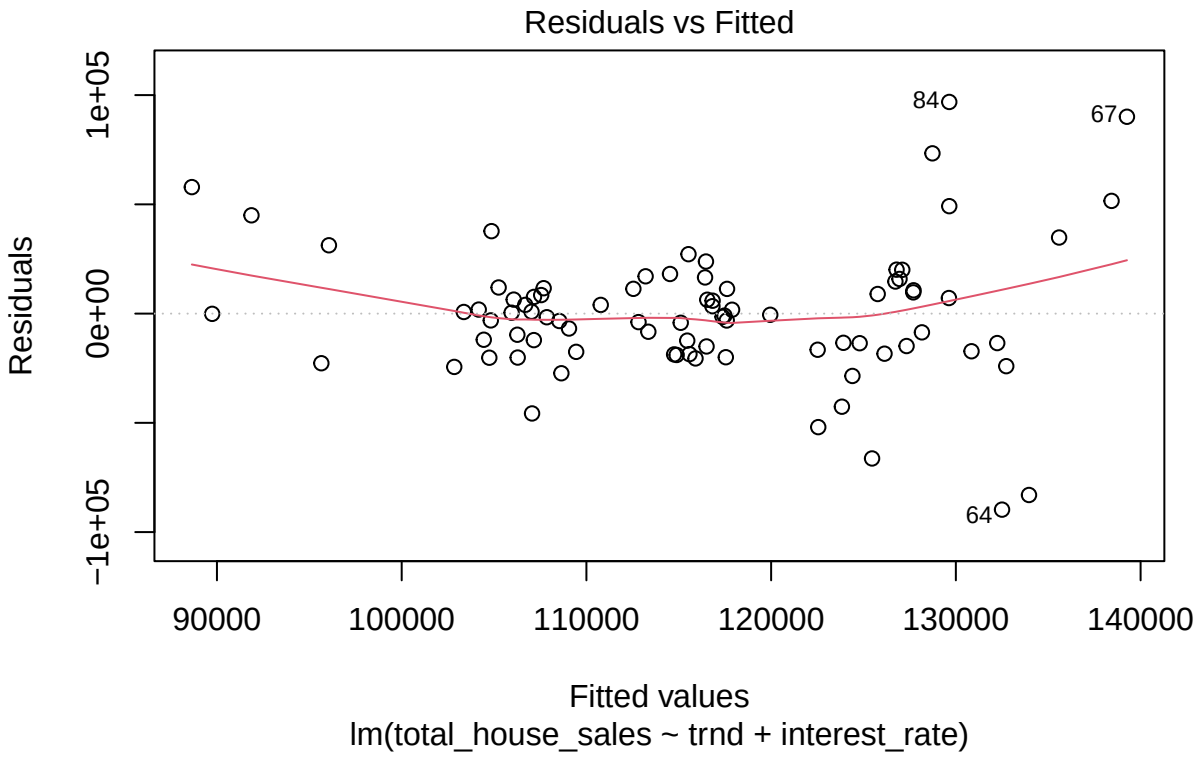


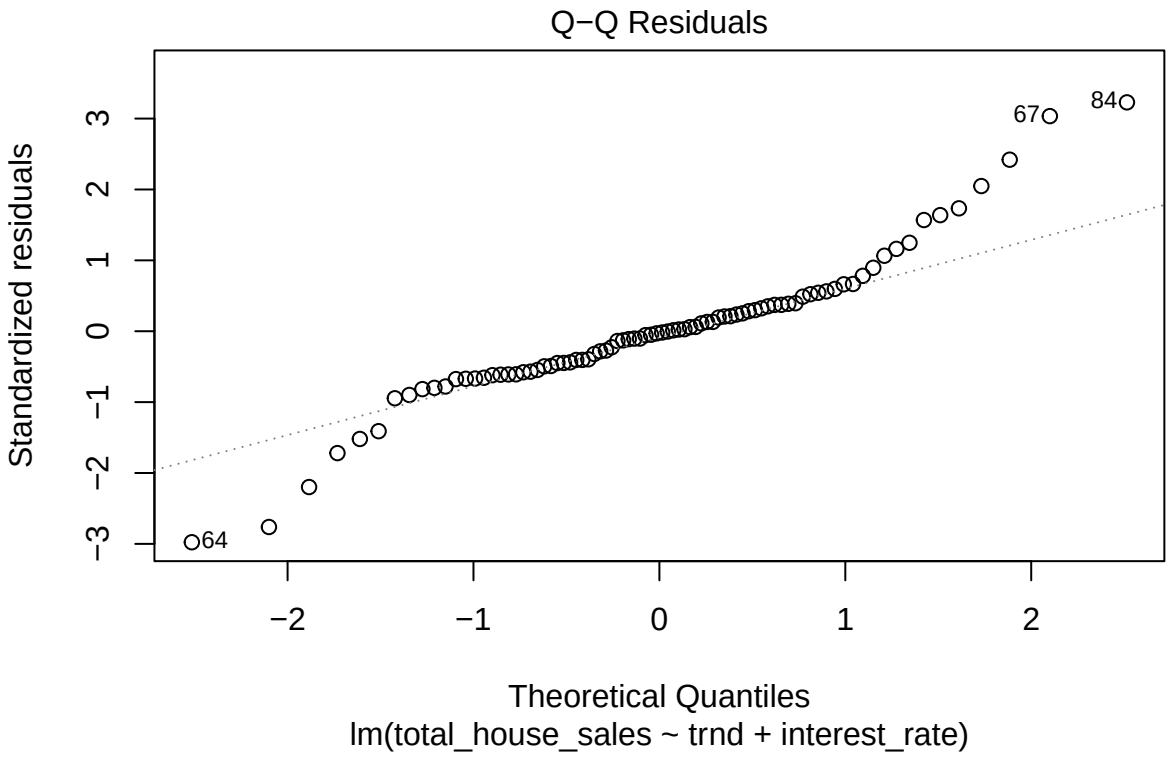

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 29.799, df = 10, p-value = 0.0009238
##
## Model df: 0.   Total lags used: 10
```

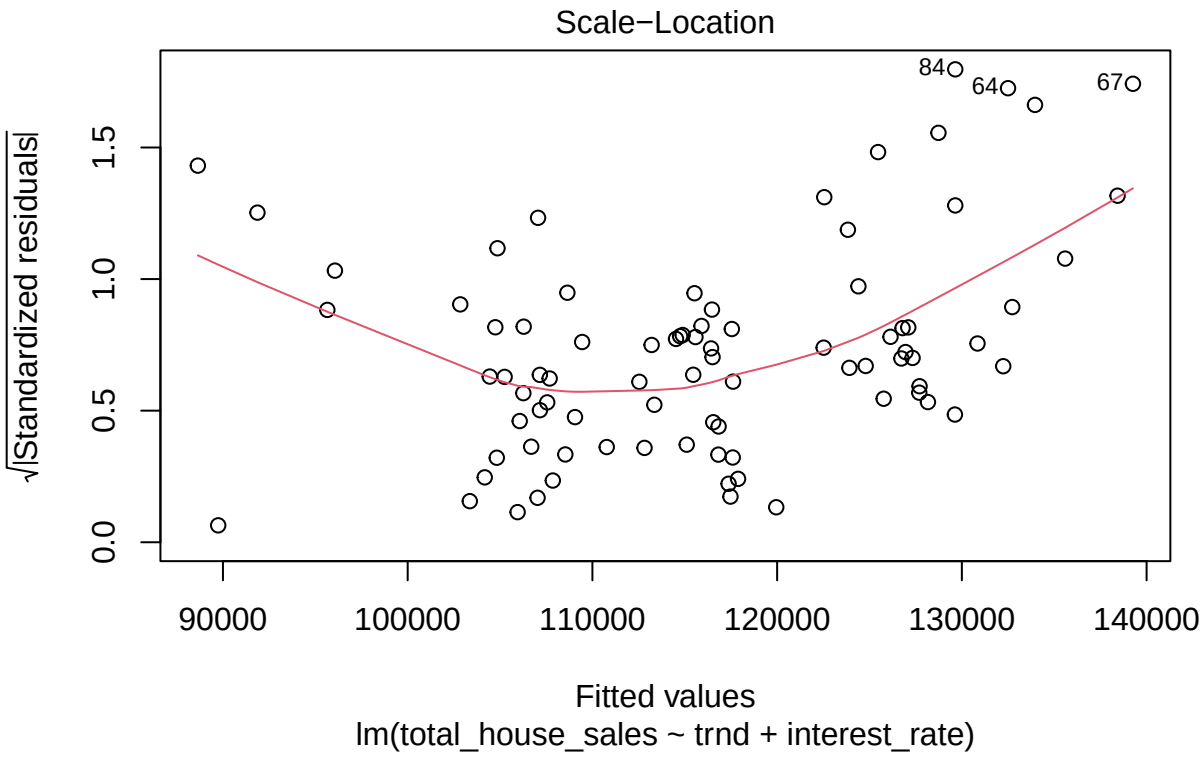
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend_house_interest_rate:=predict(model,tmp_data)]
tmp_data[,residual_trend_house_interest_rate:=actual-predicted_trend_house_interest_rate]
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_house_interest_rate,color='predicted'))
```

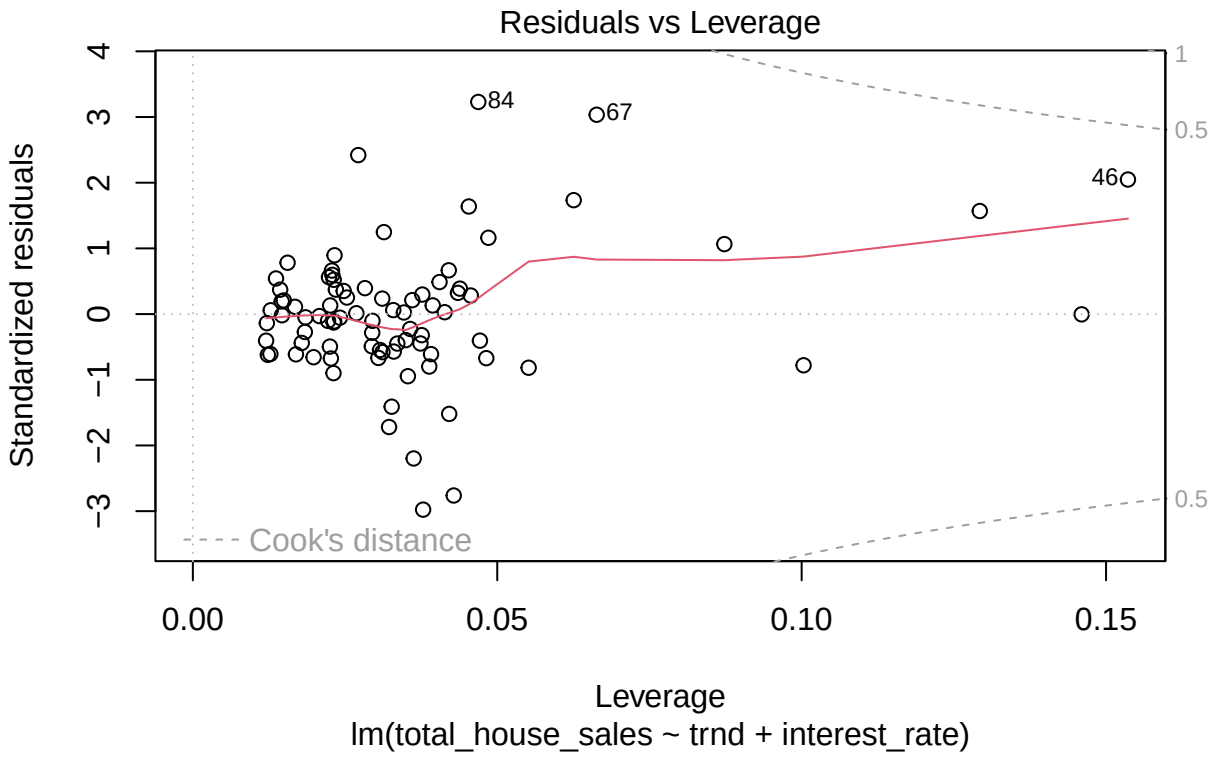

200000
150000
100000
50000
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
1e+05
0e+00
−1e+05
84
67
64
90000
100000
110000
120000
130000
140000
Fitted values
lm(total_house_sales ~ trnd + interest_rate)


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(total_house_sales ~ trnd + interest_rate)
64
67
84


Scale−Location
84
64
67
√|Standardized residuals|
0.0
0.5
1.0
1.5
90000
100000
110000
120000
130000
140000
Fitted values
lm(total_house_sales ~ trnd + interest_rate)



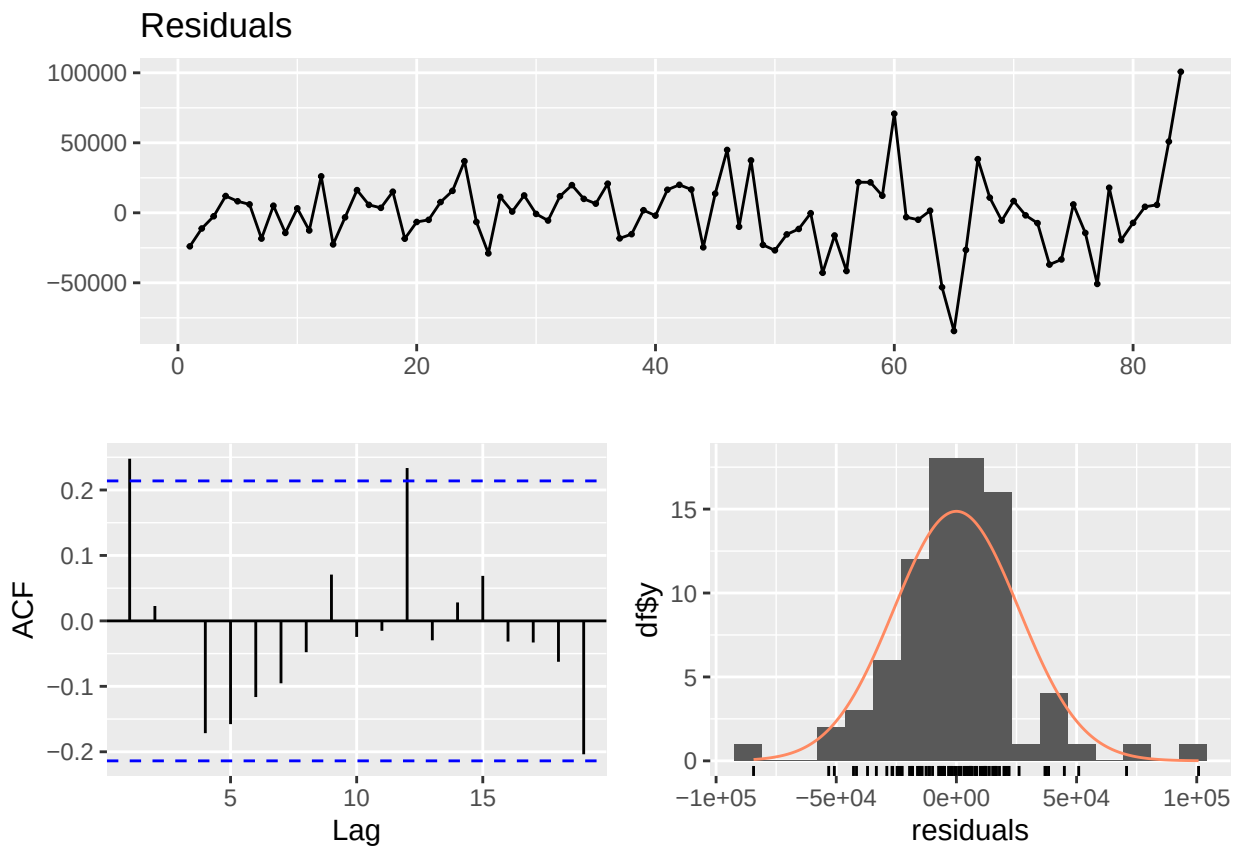
For the next step I'm also adding the number of searches for houses on google to our model. In the expanded linear regression model, total house sales are influenced by trend, interest rate, and Google searches for houses. Google searches have a strong positive effect (coefficient = 1801.9, p-value < 0.0001), while trend and interest rate have less significant impacts. The model's R-squared value of 0.35 indicates it explains 35% of the variance in total house sales, showing a considerable improvement in model fit and suggesting that Google searches are a significant predictor of house sales.

```
model=lm(total_house_sales~trnd+interest_rate+google_house_search,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = total_house_sales ~ trnd + interest_rate + google_house_search,
##     data = all_data_table)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -84408 -14587    303  12263 100762
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           60727.2    16691.5   3.638 0.000485 ***
## trnd                   -336.9      194.4  -1.733 0.086870 .
## interest_rate          -230.5      808.9  -0.285 0.776359
## google_house_search    1801.9      338.6   5.322 9.13e-07 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26570 on 80 degrees of freedom
## Multiple R-squared:   0.35,  Adjusted R-squared:  0.3256
## F-statistic: 14.36 on 3 and 80 DF,  p-value: 1.434e-07
```

```
checkresiduals(model$residuals)
```

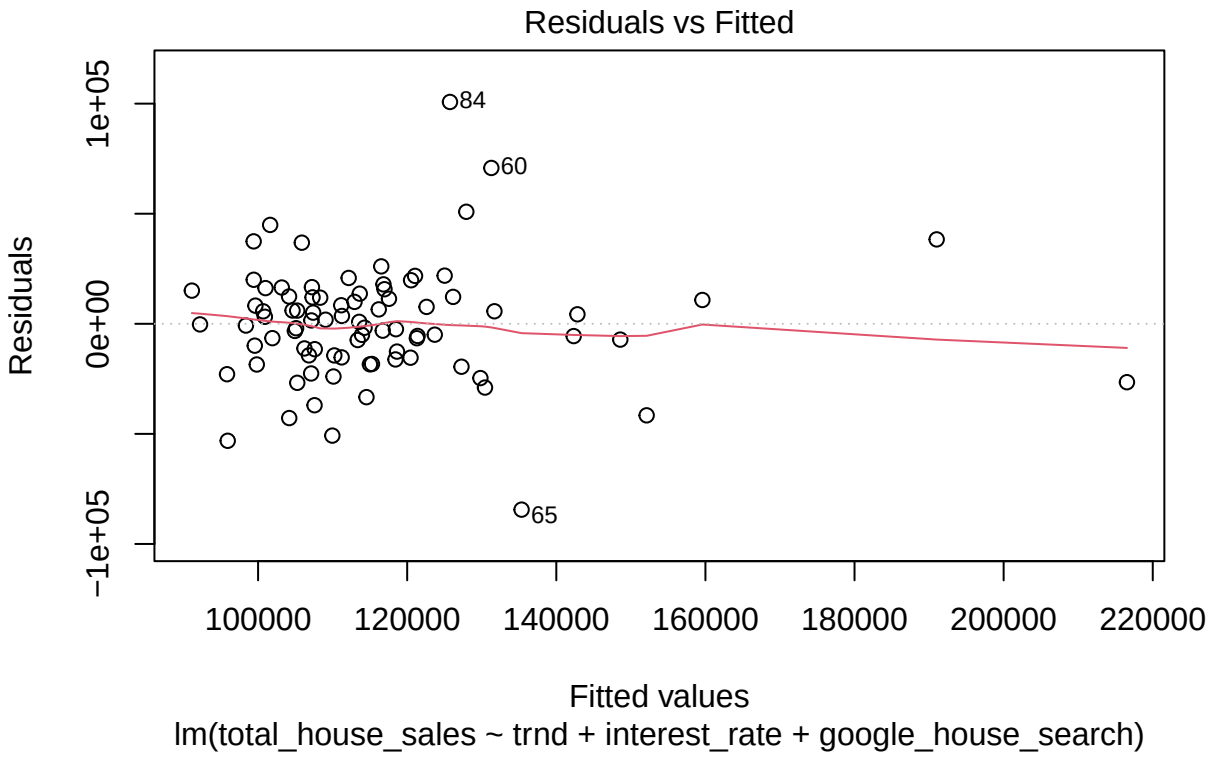


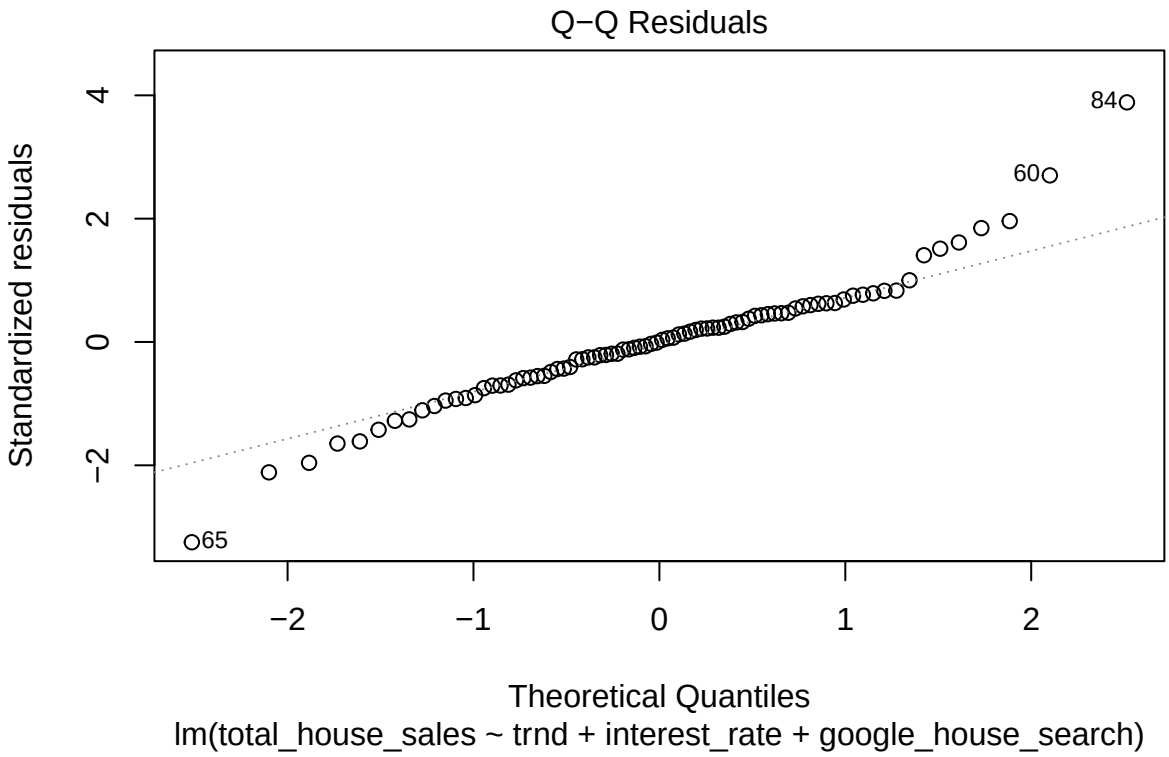

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 13.182, df = 10, p-value = 0.2137
##
## Model df: 0.   Total lags used: 10
```

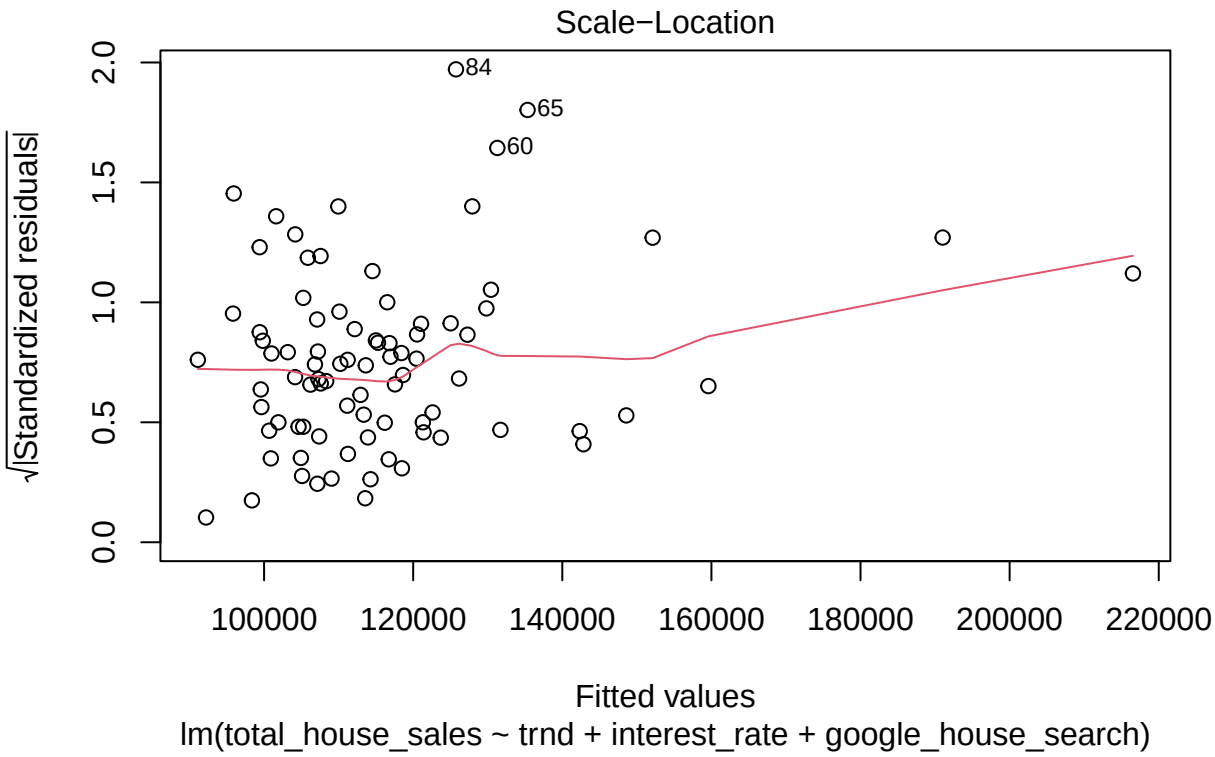
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend_house_interest_rate_search:=predict(model,tmp_data)]
tmp_data[,residual_trend_house_interest_rate_search:=actual-predicted_trend_house_interest_rate_search]
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_house_interest_rate_search,color='predicted'))
```

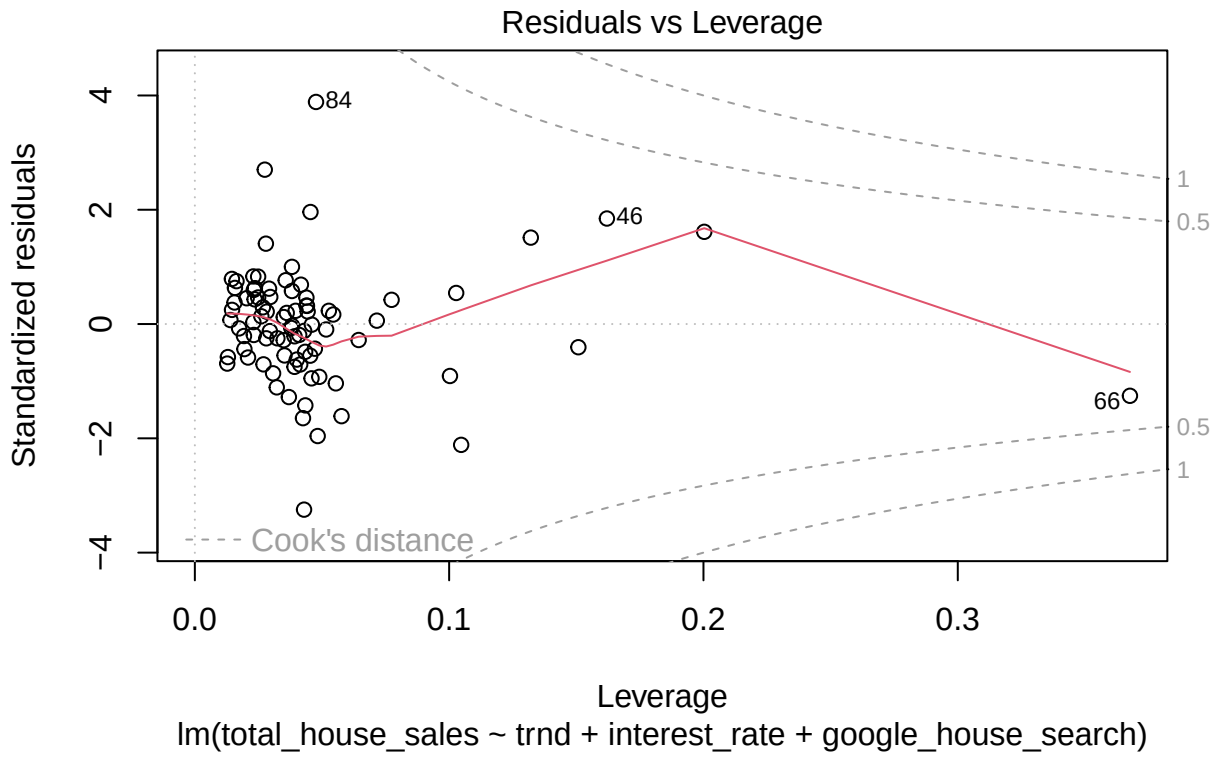

200000
150000
100000
50000
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
1e+05
0e+00
−1e+05
84
60
65
100000
120000
140000
160000
180000
200000
220000
Fitted values
lm(total_house_sales ~ trnd + interest_rate + google_house_search)


Q−Q Residuals
Standardized residuals
4
2
0
−2
84
60
65
−2
−1
0
1
2
Theoretical Quantiles
lm(total_house_sales ~ trnd + interest_rate + google_house_search)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
84
65
60
100000
120000
140000
160000
180000
200000
220000
Fitted values
lm(total_house_sales ~ trnd + interest_rate + google_house_search)



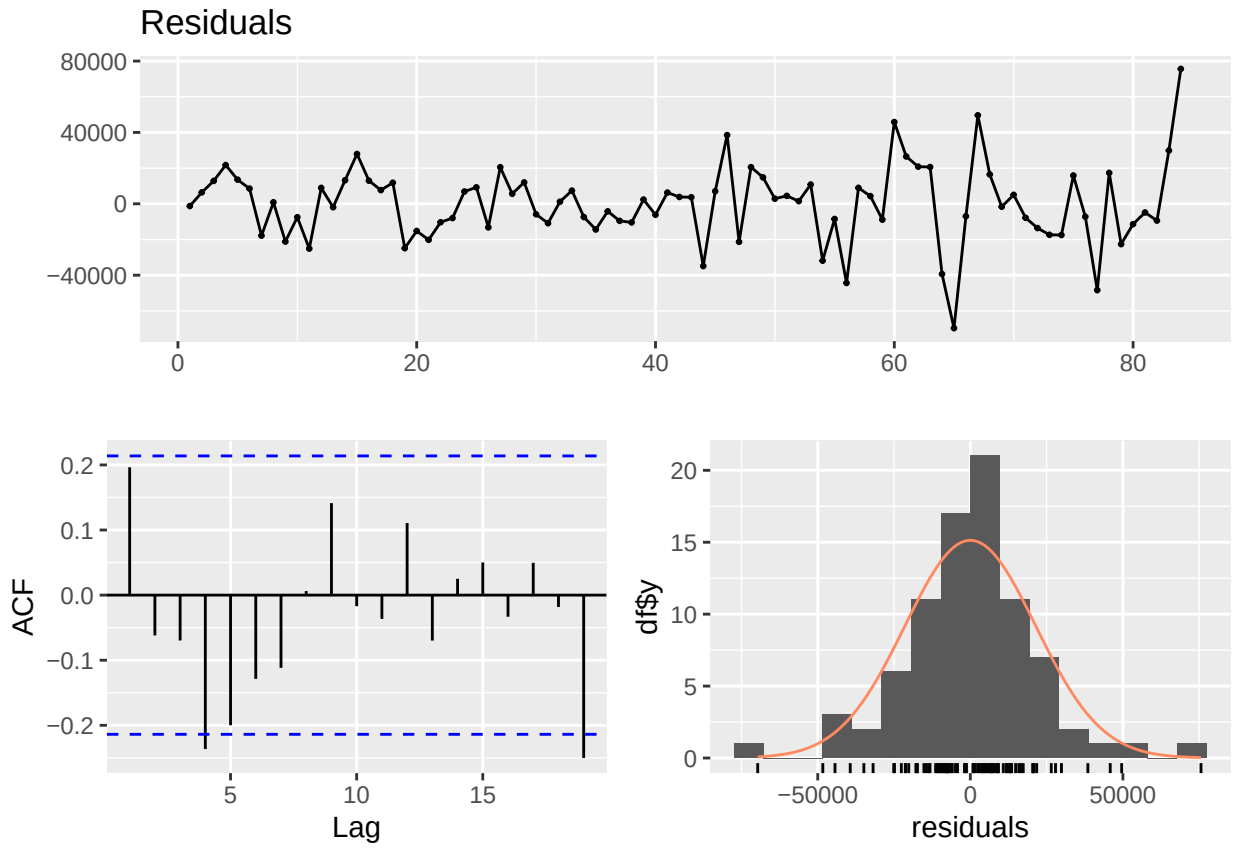
Finally, we add year data to our model and see the developments. In this regression model, total house sales are predicted using trend, interest rate, Google searches, and year. The model shows significant yearly variations with strong negative coefficients from 2016 to 2021, indicating a substantial decrease in house sales over the years. The trend and Google searches positively affect sales, with Google searches showing a particularly strong positive relationship. The model's R-squared value of 0.5604 suggests it explains approximately 56% of the variance in total house sales, significantly improving the model's explanatory power and highlighting the impact of these variables, especially the yearly changes, on house sales.

```
model=lm(total_house_sales~trnd+interest_rate+google_house_search+year,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = total_house_sales ~ trnd + interest_rate + google_house_search +
##     year, data = all_data_table)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -69678 -10534   1344  11869  75604
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           59390.8    16409.2   3.619 0.000537 ***
## trnd                   3870.2      787.0   4.918 5.13e-06 ***
## interest_rate         -1841.2      871.3  -2.113 0.037957 *
## google_house_search    1532.0      301.8   5.076 2.78e-06 ***
```

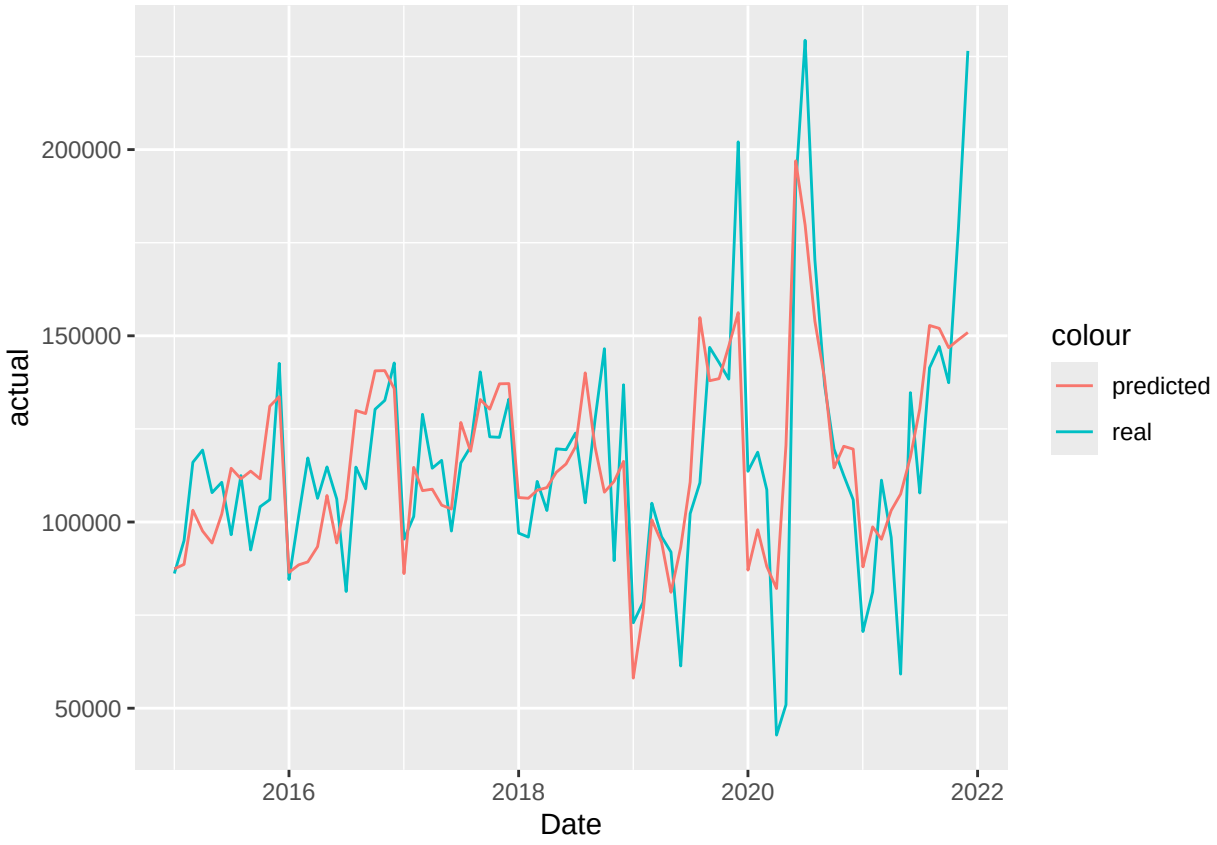
```
## year2016             -42914.0    12972.9  -3.308 0.001453 **
## year2017             -93327.6    20505.0  -4.551 2.05e-05 ***
## year2018            -129728.2    28170.1  -4.605 1.68e-05 ***
## year2019            -191684.6    36655.6  -5.229 1.52e-06 ***
## year2020            -253913.2    45811.8  -5.543 4.36e-07 ***
## year2021            -284446.7    54321.6  -5.236 1.48e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22720 on 74 degrees of freedom
## Multiple R-squared:  0.5604, Adjusted R-squared:  0.5069
## F-statistic: 10.48 on 9 and 74 DF,  p-value: 2.863e-10
```

```
checkresiduals(model$residuals)
```

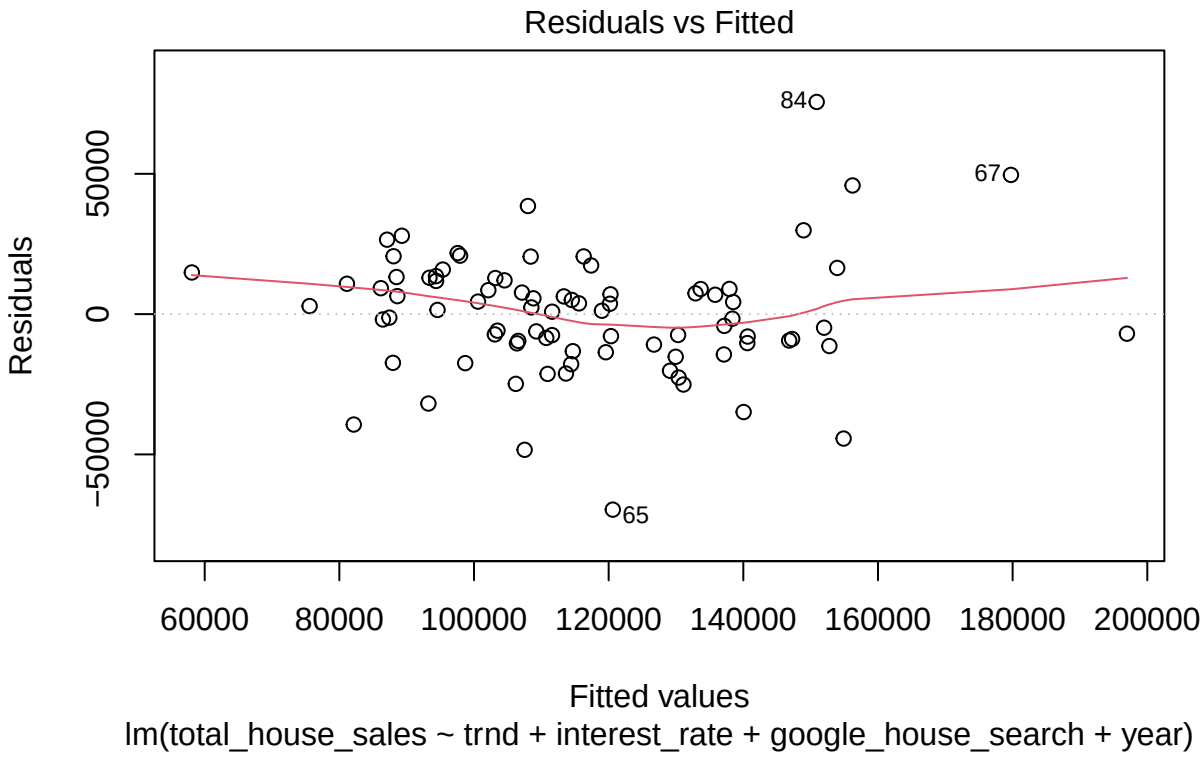


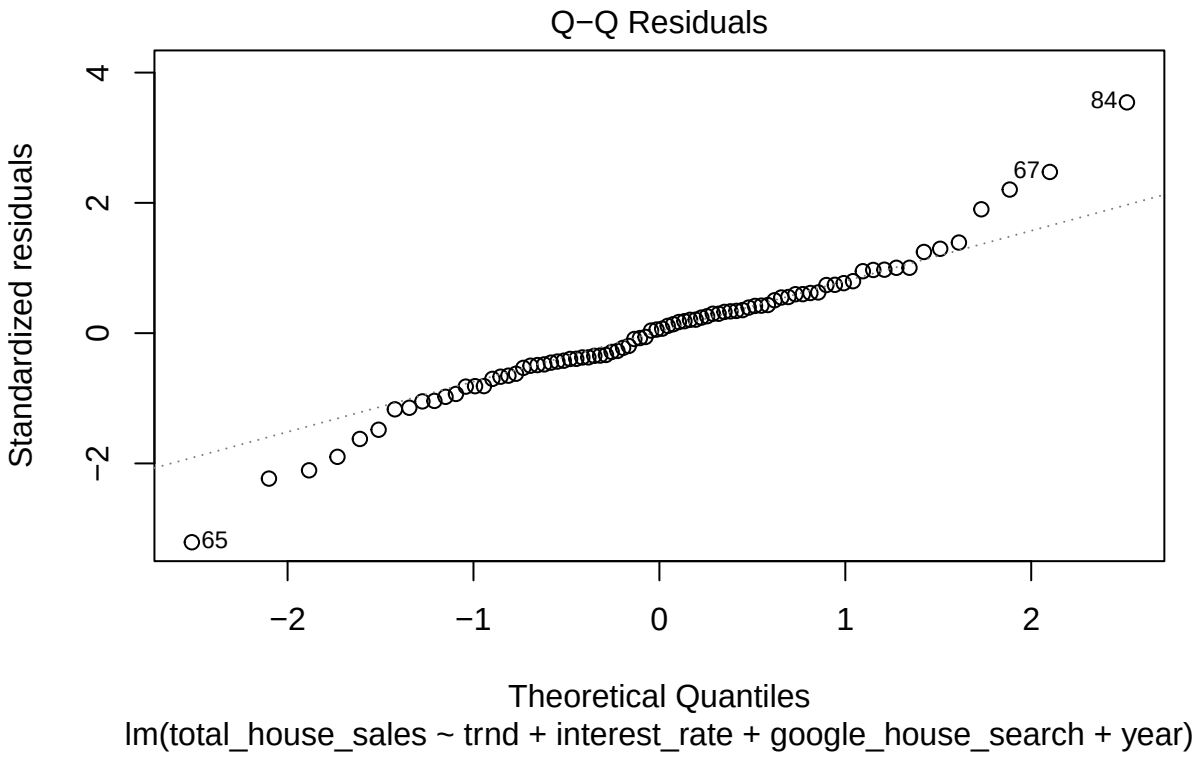

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 17.49, df = 10, p-value = 0.0642
##
## Model df: 0.   Total lags used: 10
```

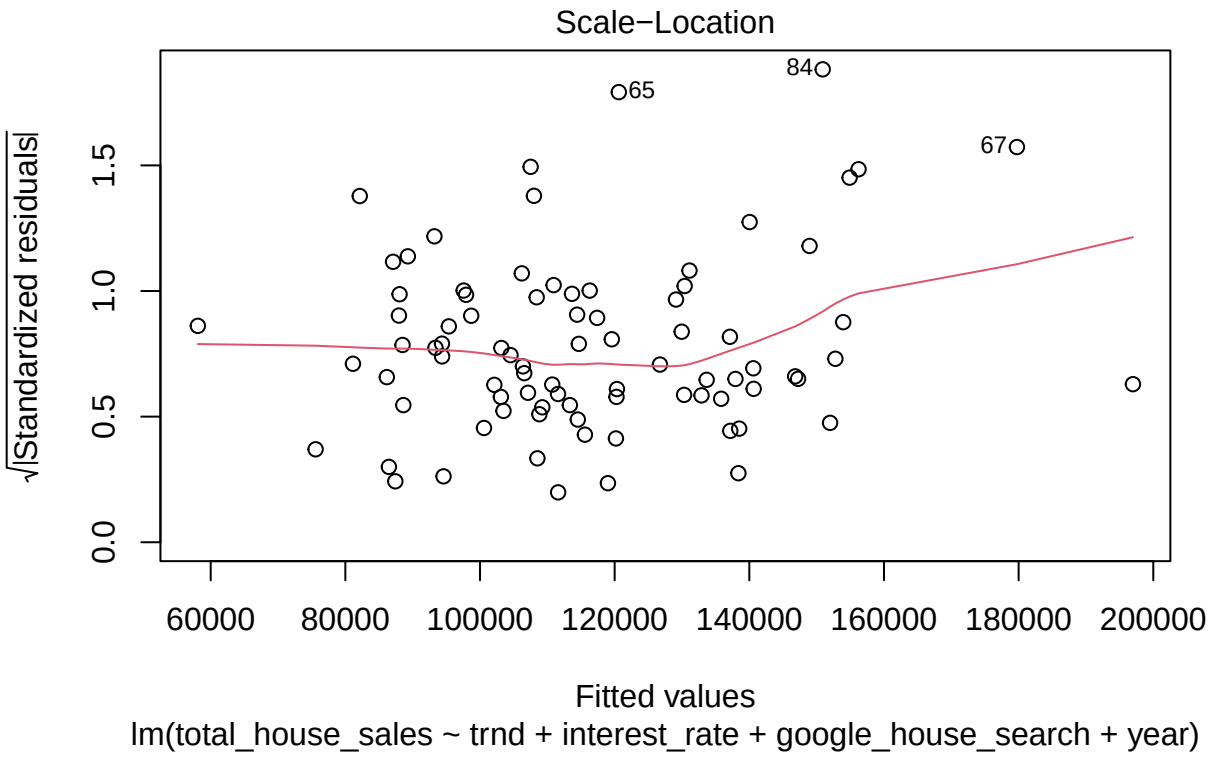
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend_house_interest_rate_search:=predict(model,tmp_data)]
tmp_data[,residual_trend_house_interest_rate_search:=actual-predicted_trend_house_interest_rate_search]
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_house_interest_rate_search,color='predicted'))
```

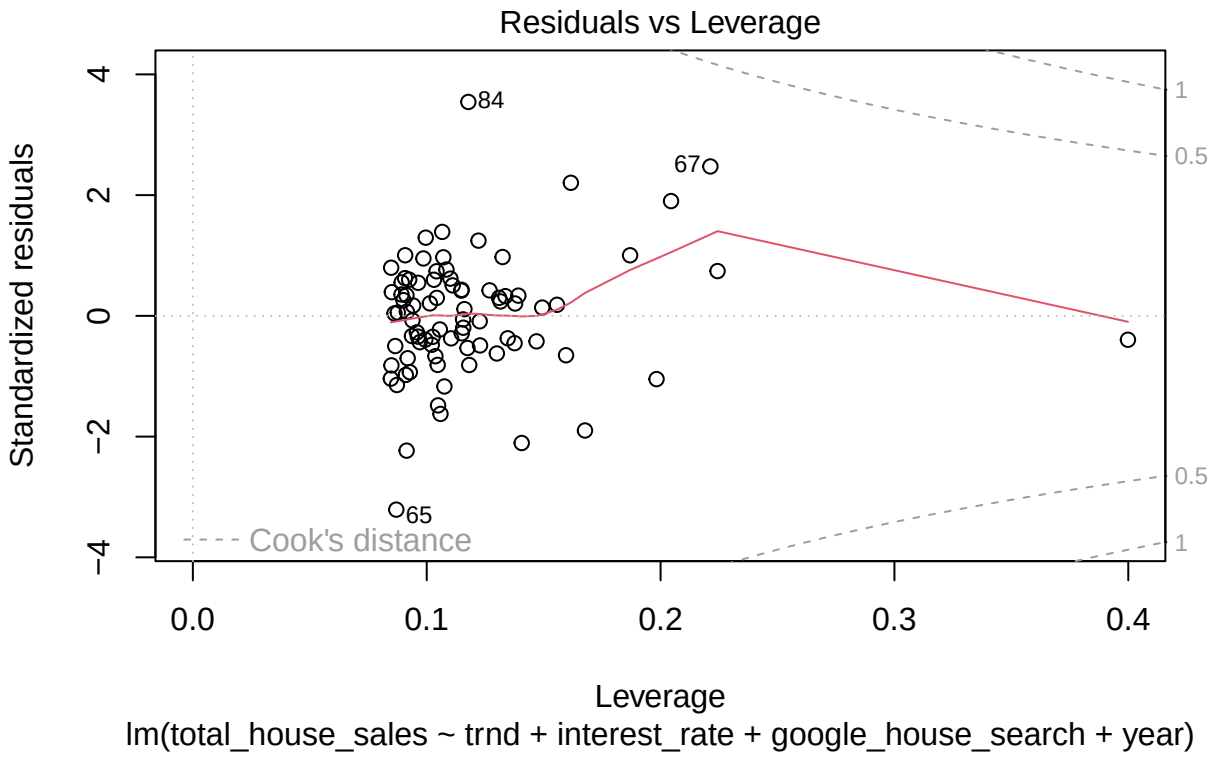



```
plot(model)
```


Residuals vs Fitted
Residuals
50000
0
−50000
84
67
65
60000
80000
100000
120000
140000
160000
180000
200000
Fitted values
lm(total_house_sales ~ trnd + interest_rate + google_house_search + year)


Q−Q Residuals
Standardized residuals
4
2
0
−2
84
67
65
−2
−1
0
1
2
Theoretical Quantiles
lm(total_house_sales ~ trnd + interest_rate + google_house_search + year)

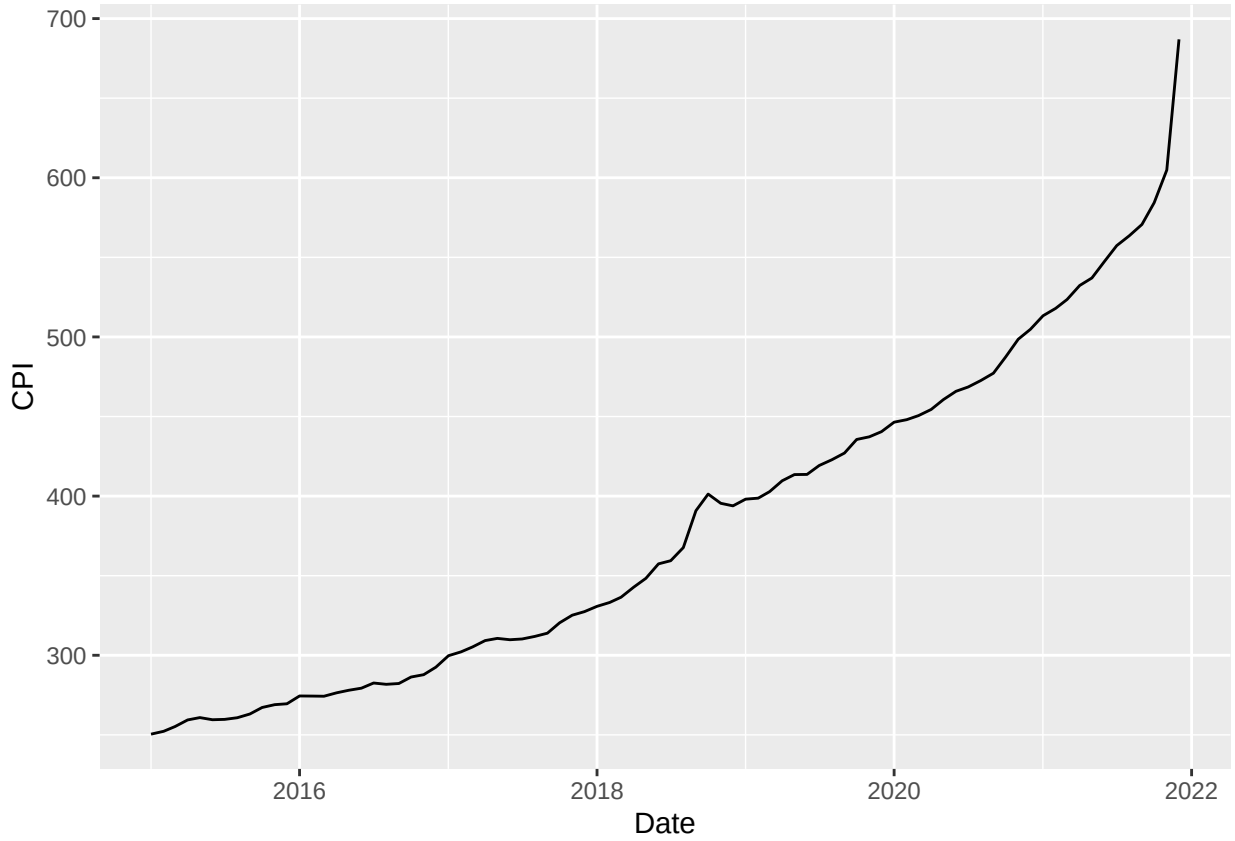

Scale−Location
84
65
67
√|Standardized residuals|
0.0
0.5
1.0
1.5
60000
80000
100000
120000
140000
160000
180000
200000
Fitted values
lm(total_house_sales ~ trnd + interest_rate + google_house_search + year)



MODEL 2 (CPI)

For the second model, I chose to examine the Consumer Price Index (CPI) data-set in Turkey covering the period from 2015 to 2023.

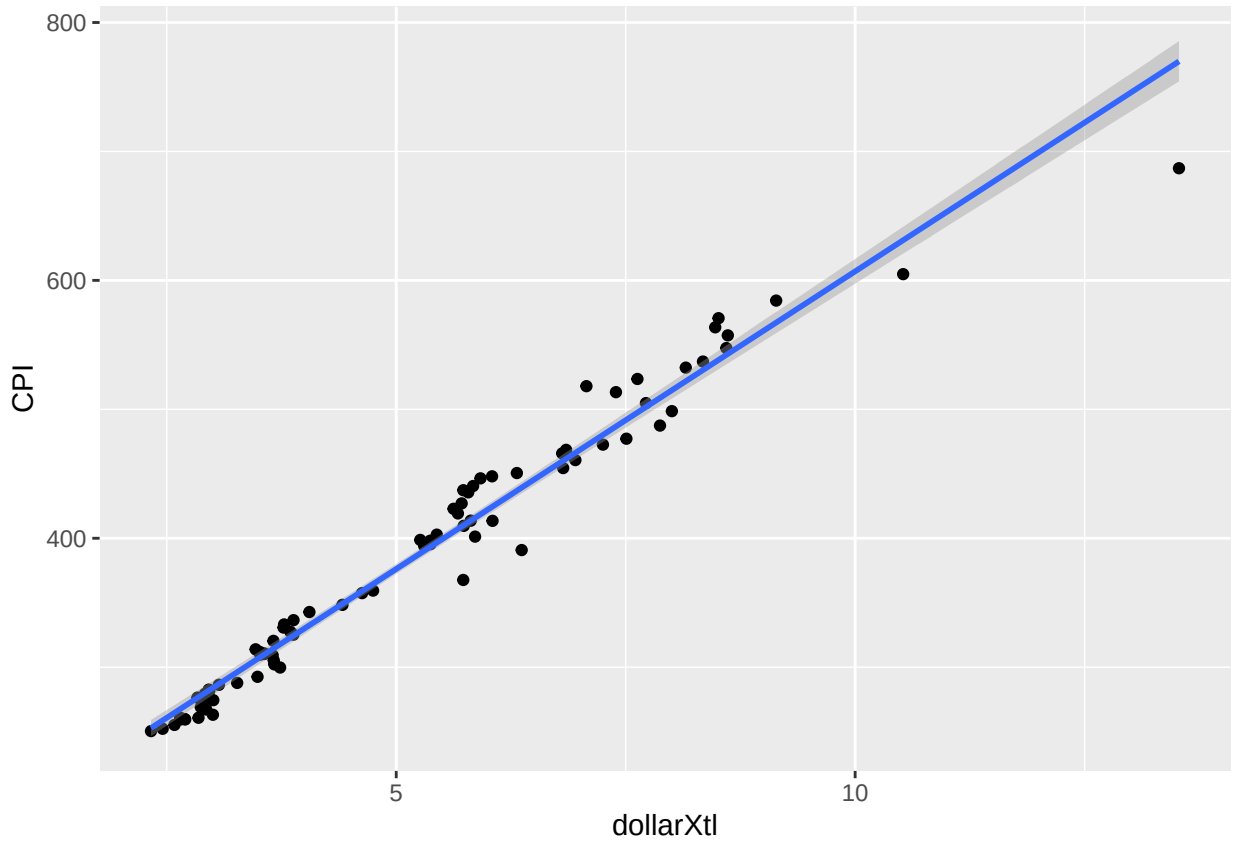
```
ggplot(all_data_table ,aes(x=Date,y=CPI, group=1)) + geom_line()
```



I have decided to use dollar/tl exchange rate to predict CPI. The graph below is the scatter plot and regression line for dollar/tl exchange rate and CPI.

```
ggplot(all_data_table ,aes(x=dollarXtl,y=CPI)) + geom_point() +geom_smooth(method='lm')
```

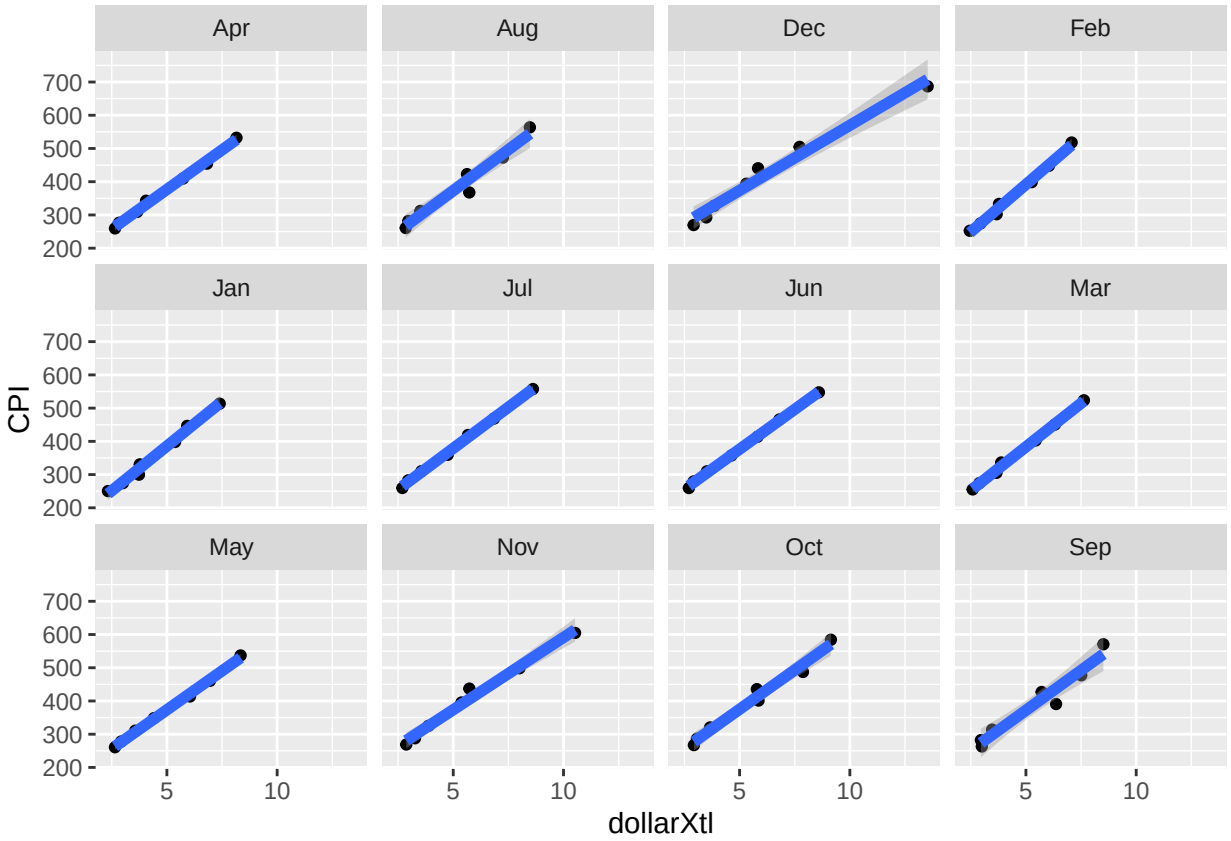
```
## `geom_smooth()` using formula = 'y ~ x'
```



The graphs below shows the correlation of the CPI and dollar/tl exchange rate for each month.

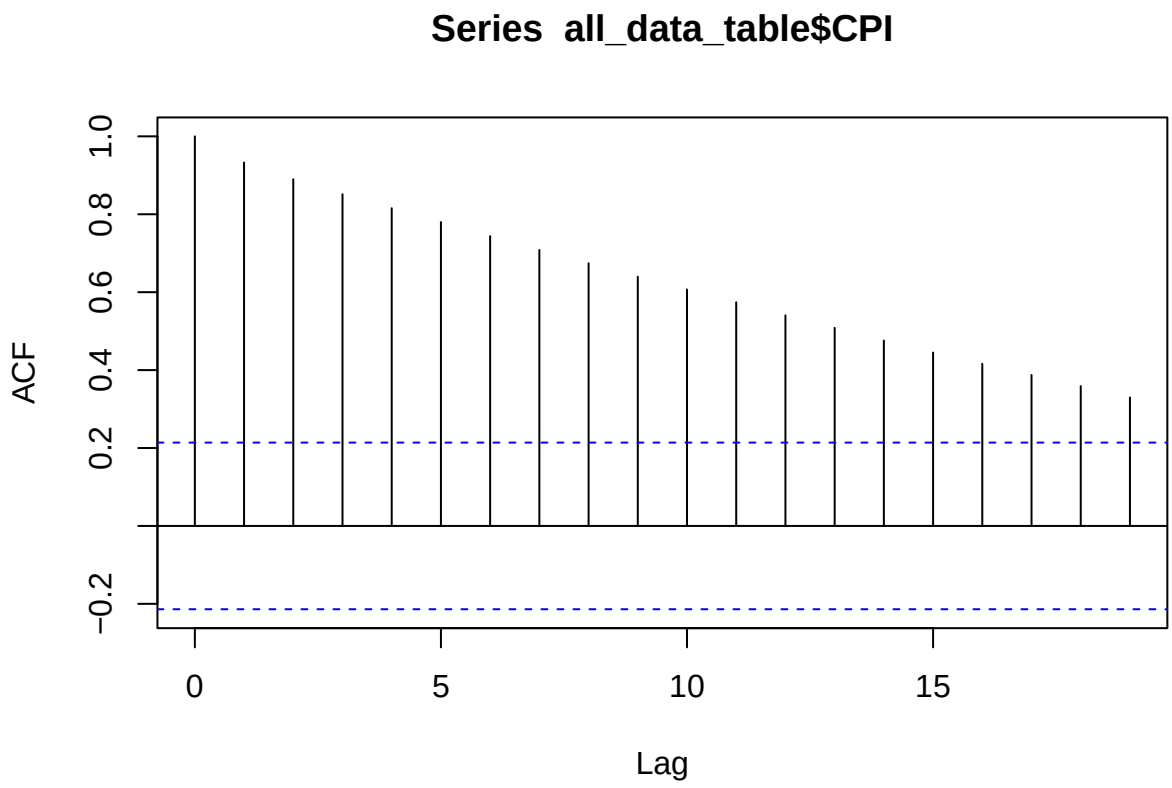
```
ggplot(all_data_table,aes(x=dollarXtl,y=CPI)) +
        geom_point() + geom_smooth(method=lm,linewidth=2) + facet_wrap(~month)
```

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



AFC plot to see if the data correlates within itself.

```
acf(all_data_table$CPI)
```



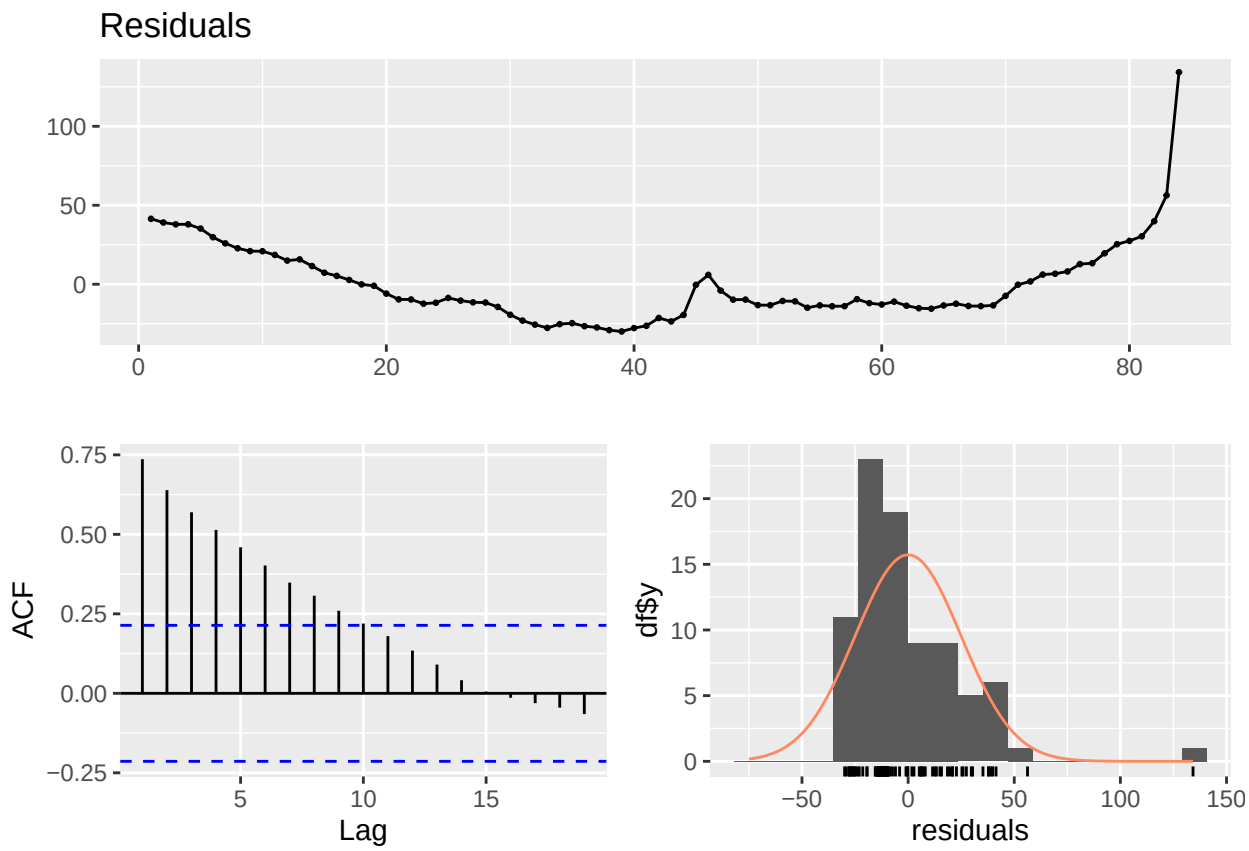
First step is to start building the model. In this linear regression model, the Consumer Price Index (CPI) is analyzed using only the trend variable. The model shows a strong positive trend (coefficient = 4.1409, p-value < 2e-16), indicating a significant increase in CPI over time. The high R-squared value of 0.9423 suggests that the model explains approximately 94% of the variance in CPI, demonstrating a strong linear relationship between time and CPI in the dataset.

```
model=lm(CPI~trnd,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = CPI ~ trnd, data = all_data_table)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.904 -13.865  -9.691  12.870 134.226
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 204.8895     5.5349   37.02   <2e-16 ***
## trnd          4.1409     0.1131   36.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.14 on 82 degrees of freedom
## Multiple R-squared:  0.9423, Adjusted R-squared:  0.9416
```

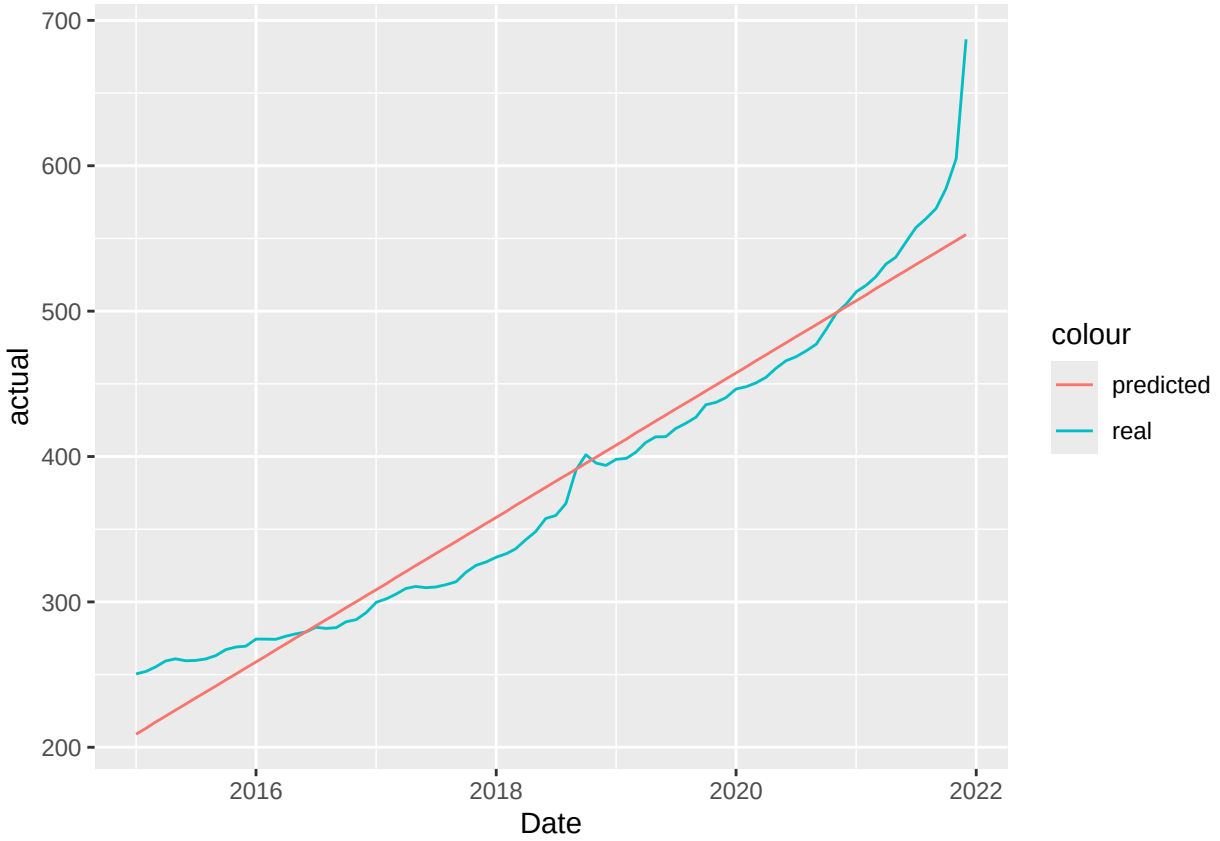
```
## F-statistic:  1340 on 1 and 82 DF,  p-value: < 2.2e-16
```

```
checkresiduals(model$residuals)
```

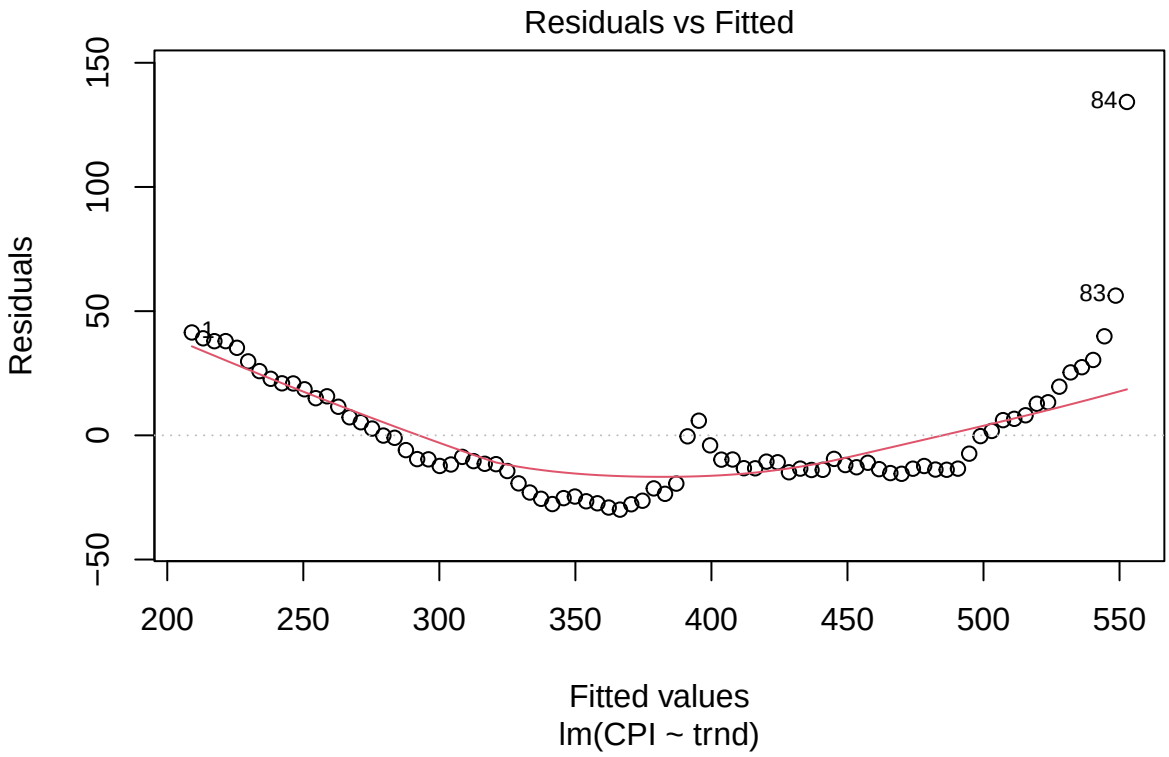


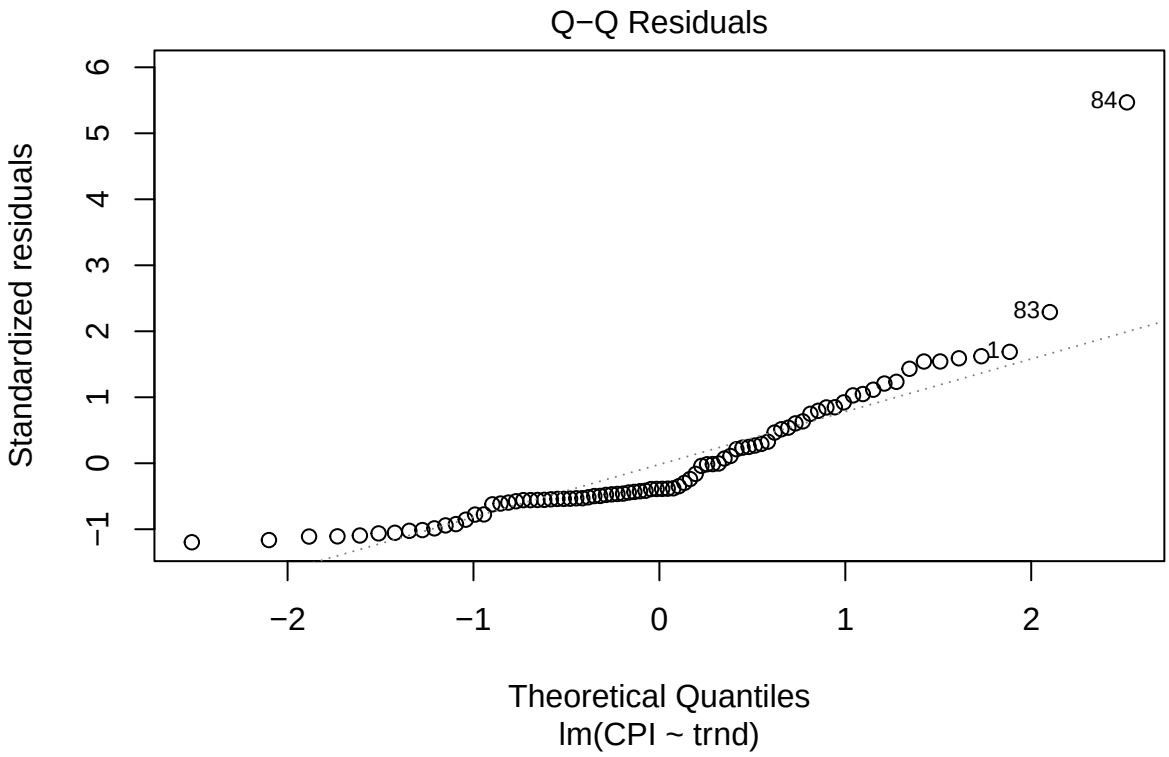

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 201.71, df = 10, p-value < 2.2e-16
##
## Model df: 0.   Total lags used: 10
```

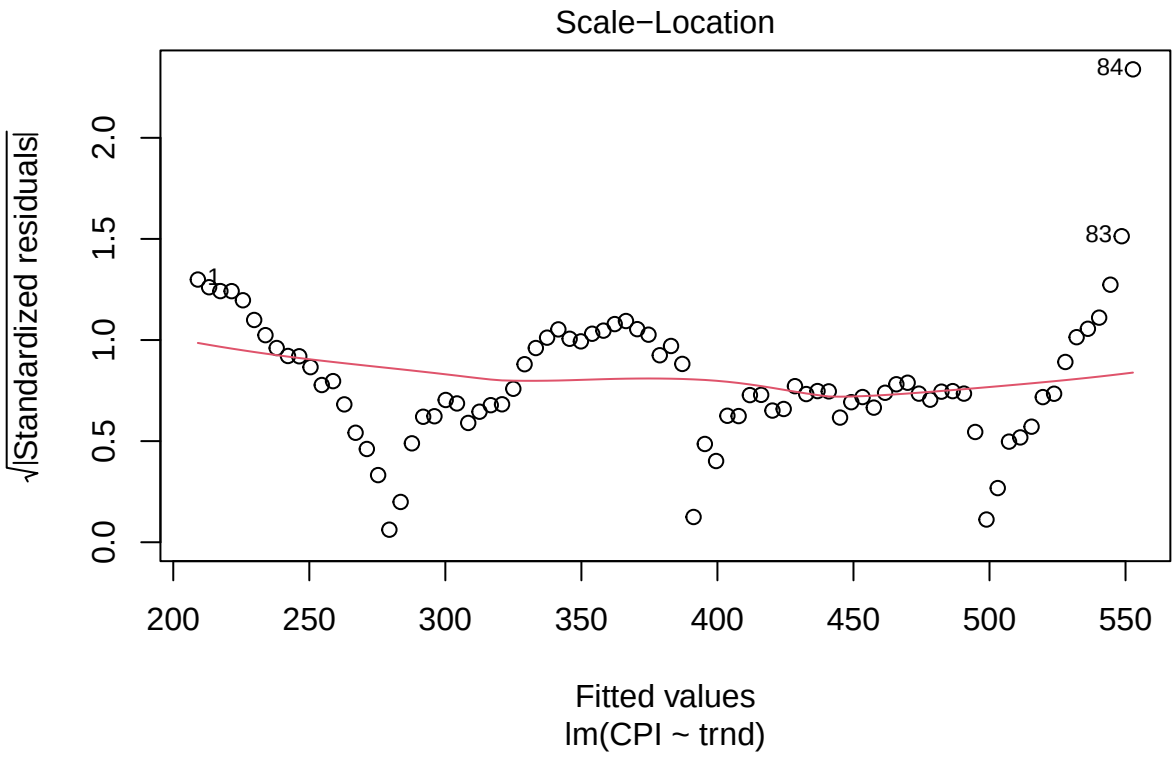
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=CPI]
tmp_data[,predicted_trend:=predict(model,tmp_data)]
tmp_data[,residual_trend:=actual-predicted_trend]
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend,color='predicted'))
```

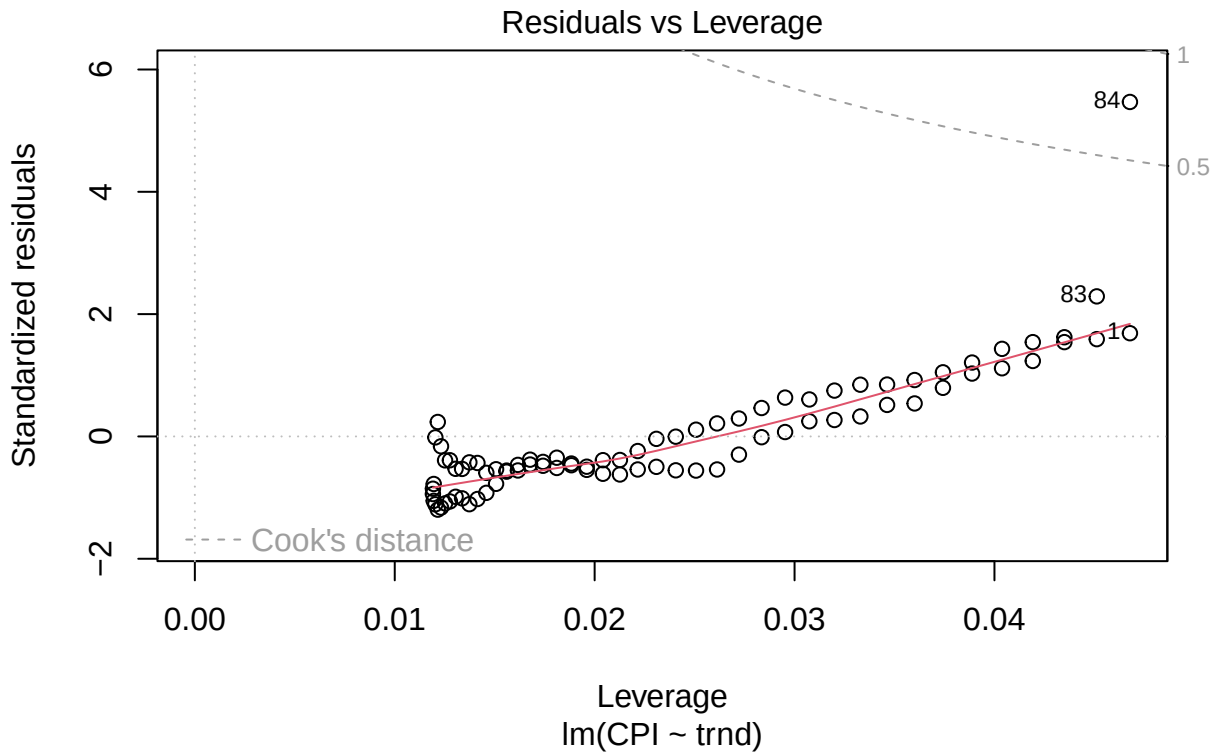

700
600
500
400
300
200
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
150
100
50
0
−50
84
83
1
200
250
300
350
400
450
500
550
Fitted values
lm(CPI ~ trnd)


Q−Q Residuals
Standardized residuals
−1
0
1
2
3
4
5
6
84
83
1
−2
−1
0
1
2
Theoretical Quantiles
lm(CPI ~ trnd)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
84
83
1
200
250
300
350
400
450
500
550
Fitted values
lm(CPI ~ trnd)



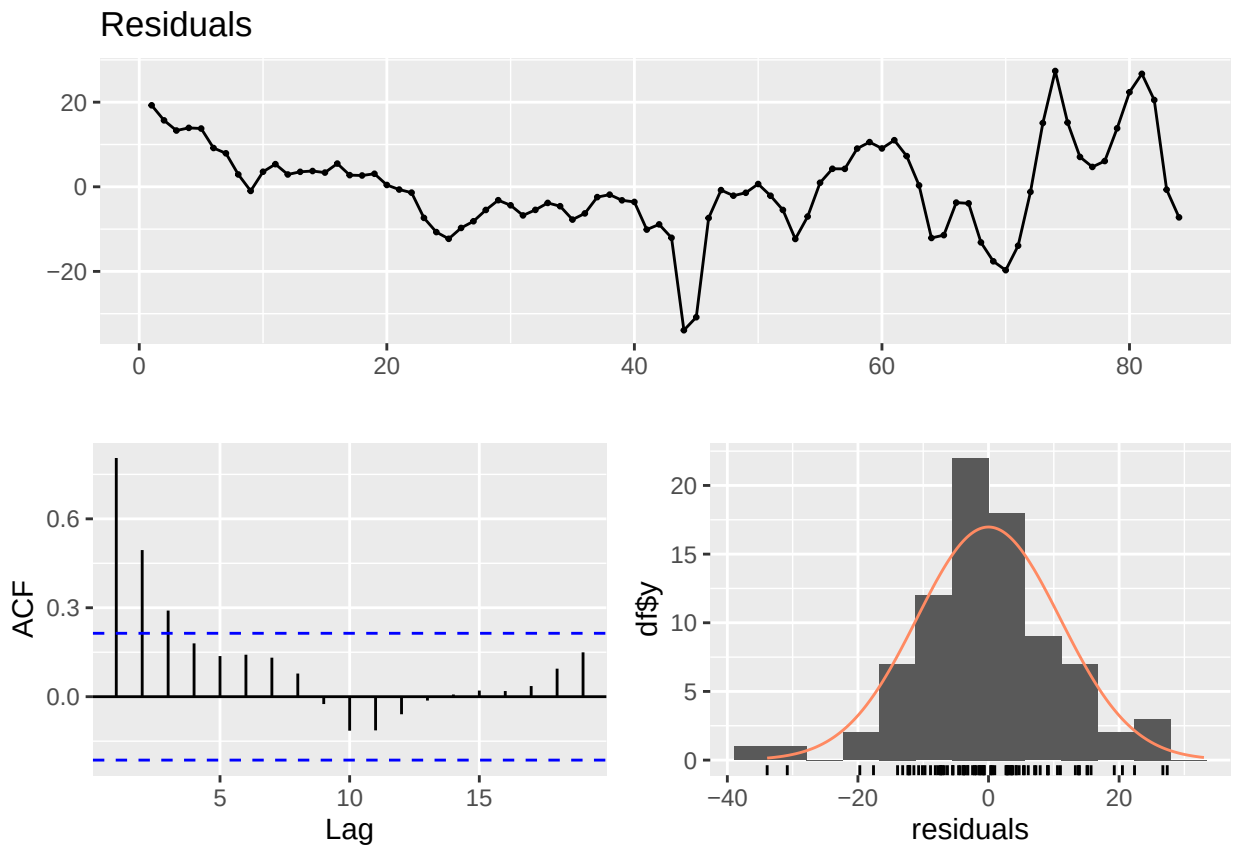
For the next step we are adding dollar/tl exchange rate data-set into our model. In this regression analysis, the CPI is modeled using the trend and the exchange rate (dollar to Turkish lira). The model indicates a significant positive relationship with both the trend and the exchange rate, with notably high coefficients (trnd = 1.6725, dollarXtl = 28.9434, both with p-values < 2e-16). The model has an extremely high R-squared value of 0.9888, meaning it explains around 99% of the variance in CPI, showing a very strong linear relationship with these predictors.

```
model=lm(CPI~trnd+dollarXtl,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = CPI ~ trnd + dollarXtl, data = all_data_table)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.905  -6.831  -0.877   5.638  27.375
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 162.1254     3.3816   47.94   <2e-16 ***
## trnd          1.6725     0.1435   11.66   <2e-16 ***
## dollarXtl    28.9434     1.5766   18.36   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

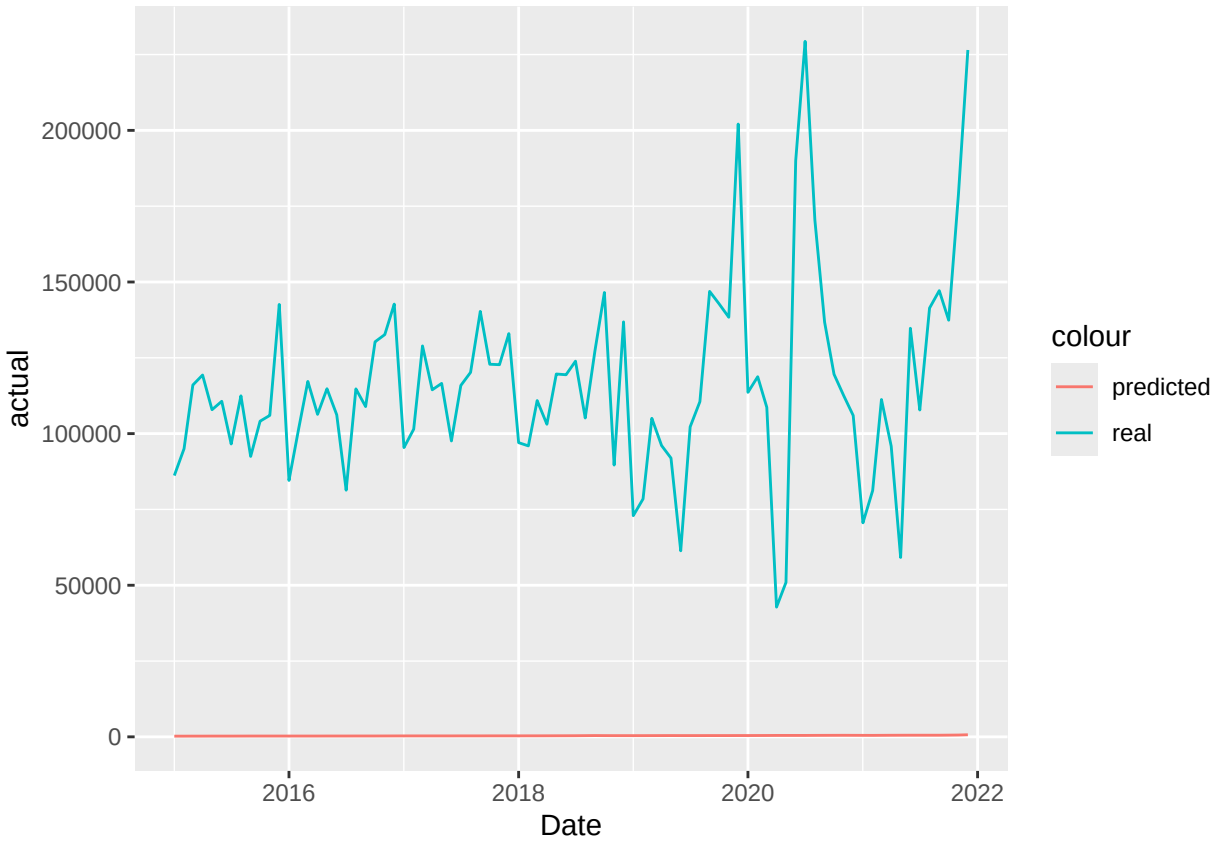
```
## Residual standard error: 11.13 on 81 degrees of freedom
## Multiple R-squared:  0.9888, Adjusted R-squared:  0.9886
## F-statistic:  3584 on 2 and 81 DF,  p-value: < 2.2e-16
```

```
checkresiduals(model)
```

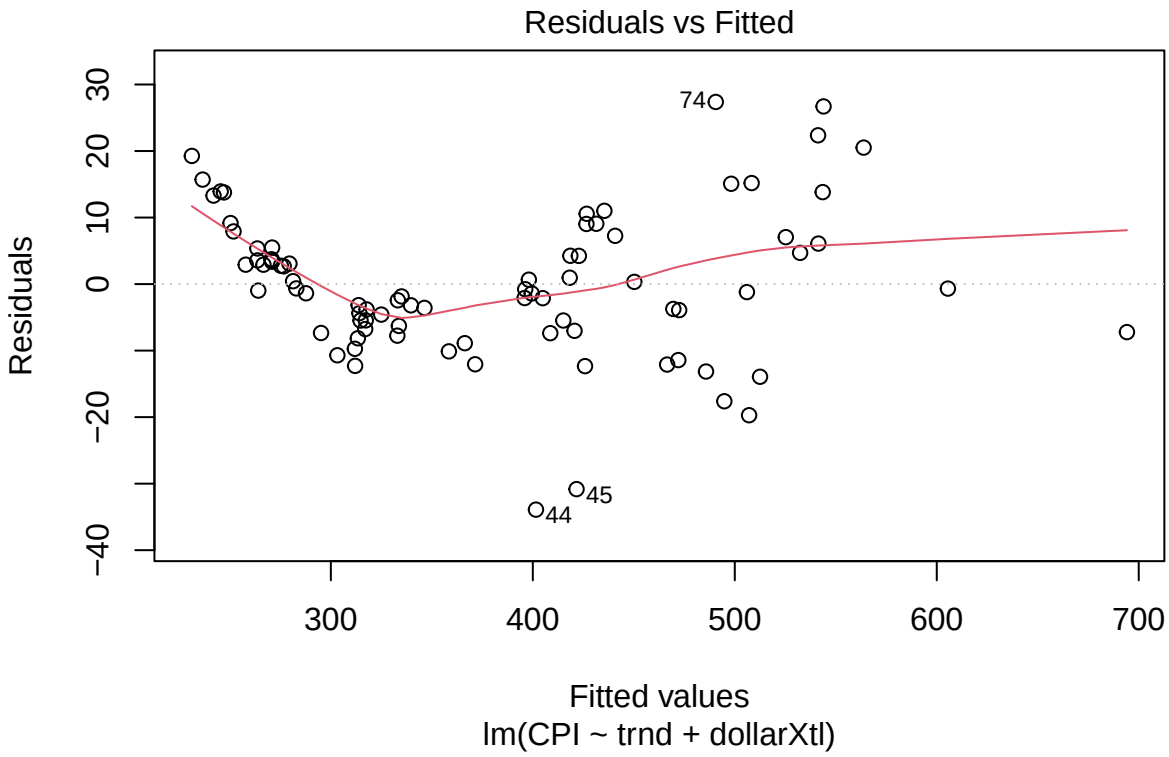


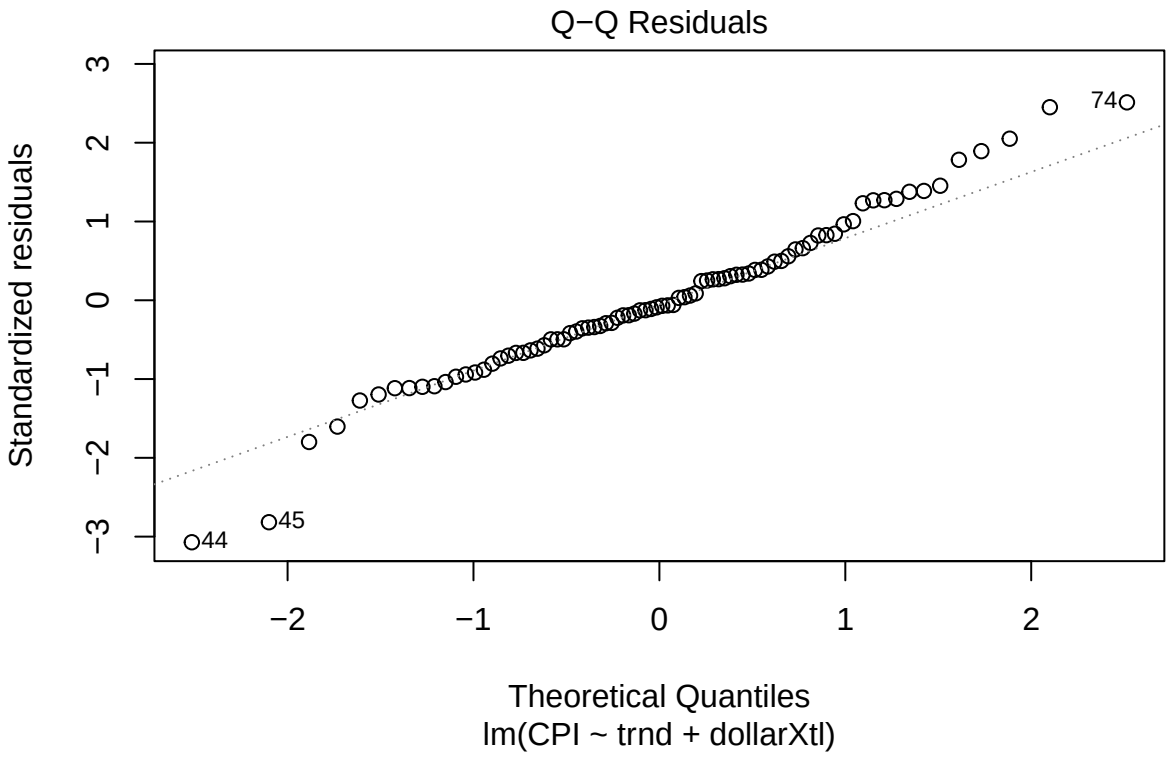

```
##
##  Breusch-Godfrey test for serial correlation of order up to 10
##
## data:  Residuals
## LM test = 63.678, df = 10, p-value = 7.25e-10
```

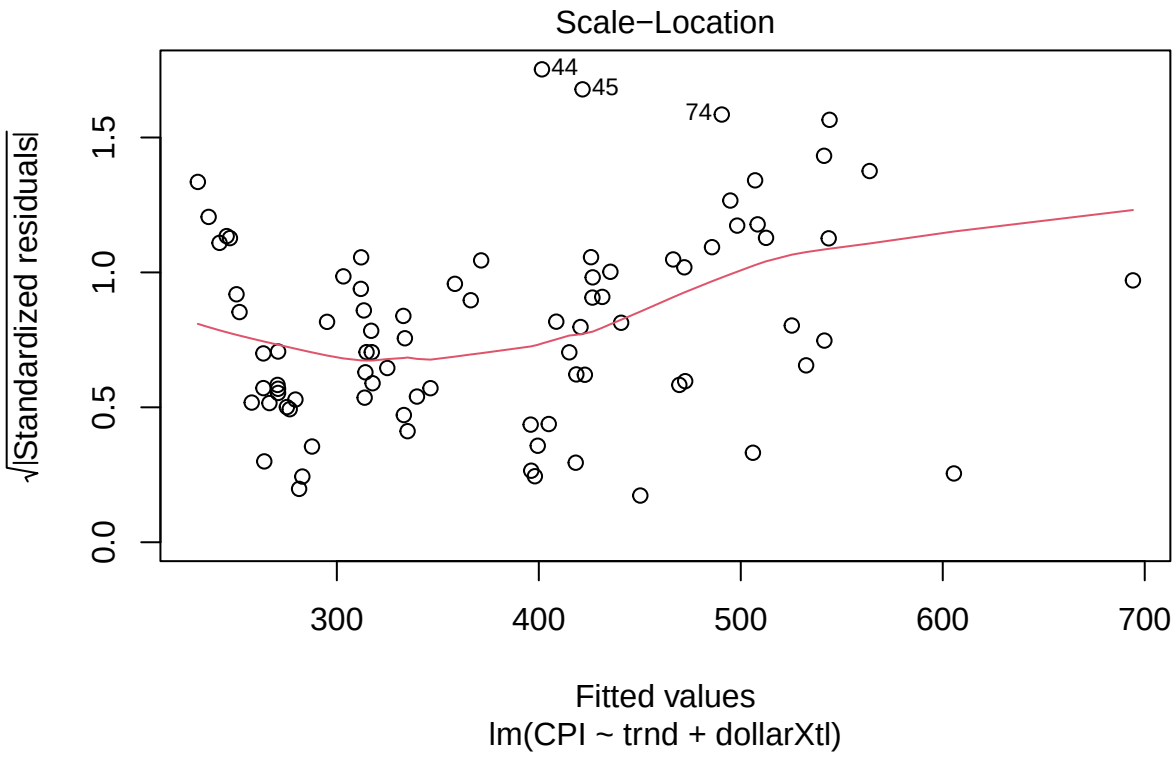
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend_dollar:=predict(model,tmp_data)]
tmp_data[,residual_trend_dollar:=actual-predicted_trend_dollar]
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_dollar,color='predicted'))
```

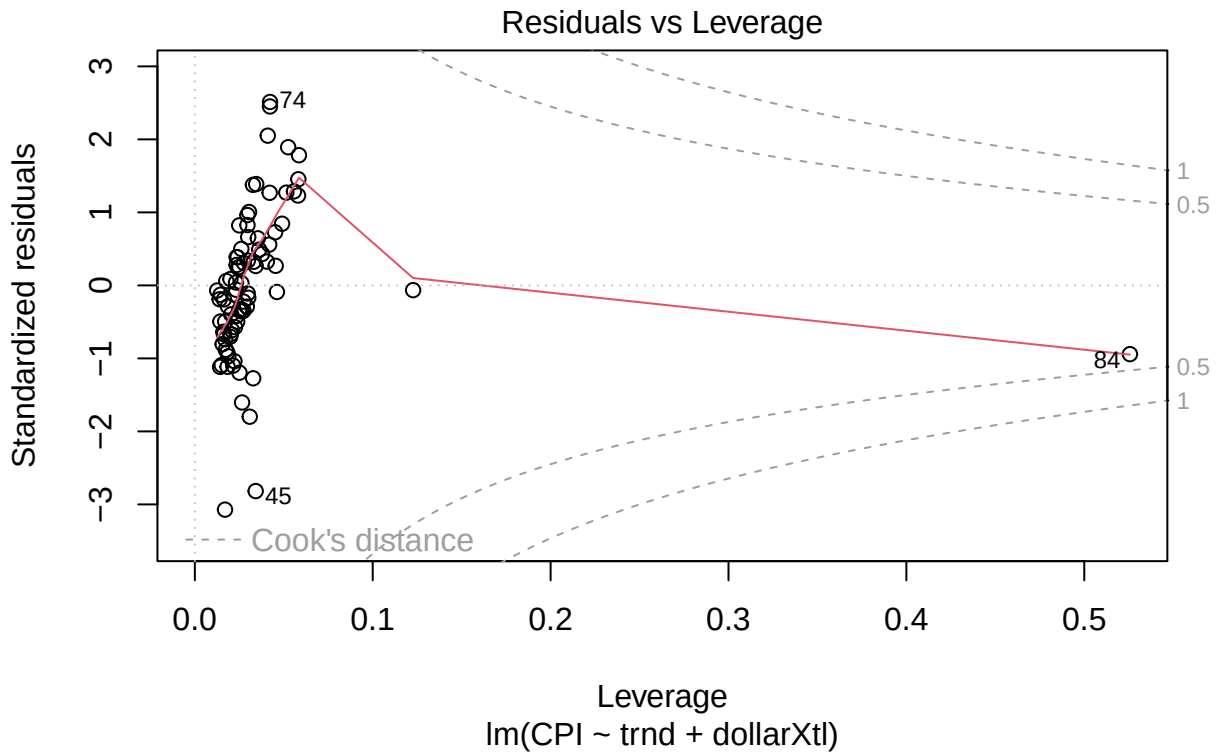

200000
150000
100000
50000
0
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
30
20
10
0
−20
−40
74
45
44
300
400
500
600
700
Fitted values
lm(CPI ~ trnd + dollarXtl)


Q−Q Residuals
Standardized residuals
3
2
1
0
−1
−2
−3
74
45
44
−2
−1
0
1
2
Theoretical Quantiles
lm(CPI ~ trnd + dollarXtl)


Scale−Location
44
45
74
√|Standardized residuals|
0.0
0.5
1.0
1.5
300
400
500
600
700
Fitted values
lm(CPI ~ trnd + dollarXtl)



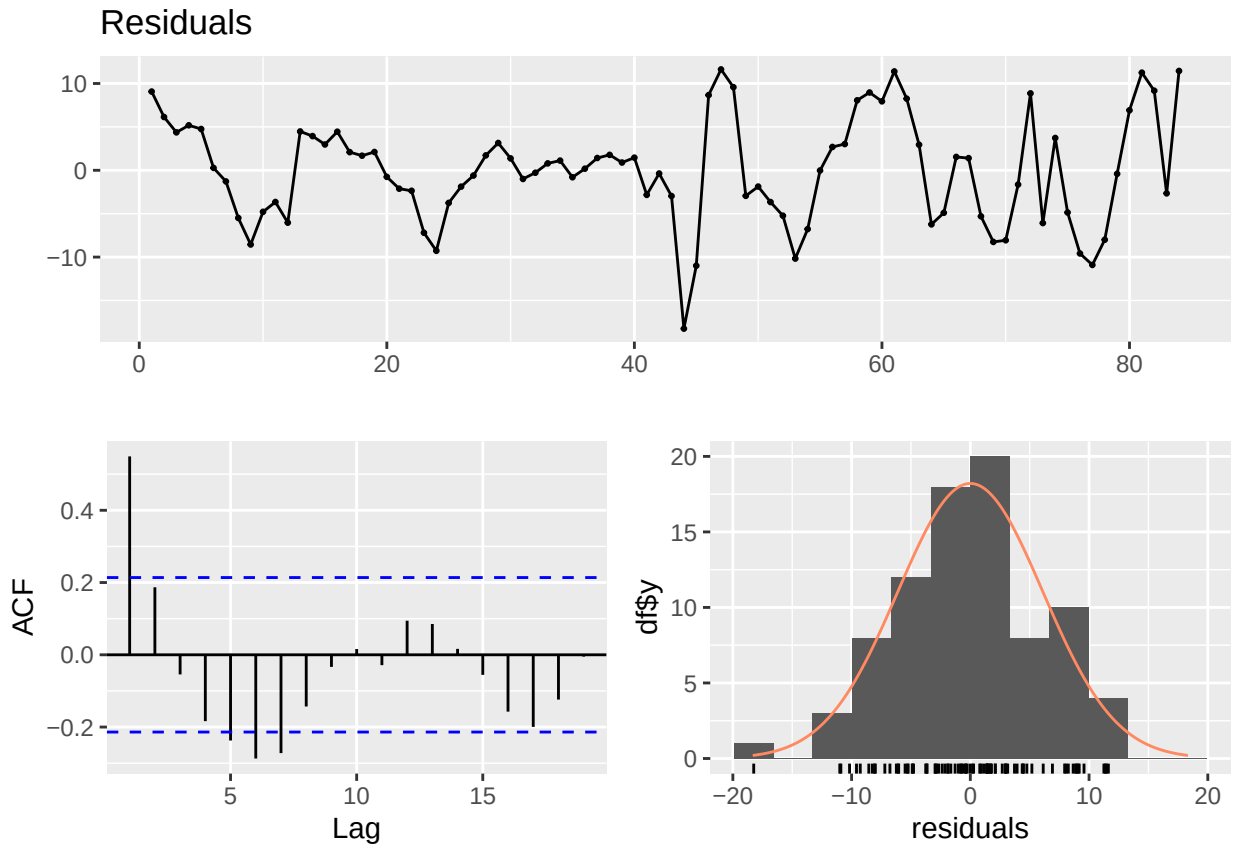
For the next step I am adding year data to our model. In this model, CPI is predicted using the trend, the dollar to Turkish lira exchange rate, and the year. The results show significant positive effects for the trend and exchange rate on CPI. Yearly coefficients vary, with some years like 2016 and 2017 showing significant negative effects. The model has an extremely high R-squared value of 0.9966, indicating it accounts for nearly 100% of the variance in CPI, highlighting the strong influence of these factors on the CPI over time.

```
model=lm((CPI)~trnd+(dollarXtl)+year,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = (CPI) ~ trnd + (dollarXtl) + year, data = all_data_table)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.2432  -4.0144   0.0815   3.7750  11.6248
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 188.25568    3.13538  60.042  < 2e-16 ***
## trnd          1.93157    0.25326   7.627  6.1e-11 ***
## dollarXtl    21.98668    1.15471  19.041  < 2e-16 ***
## year2016     -9.51160    3.87042  -2.458   0.0163 *
## year2017    -15.16451    6.09939  -2.486   0.0151 *
## year2018    -13.32951    8.33189  -1.600   0.1138
## year2019      0.05671   10.80645   0.005   0.9958
```

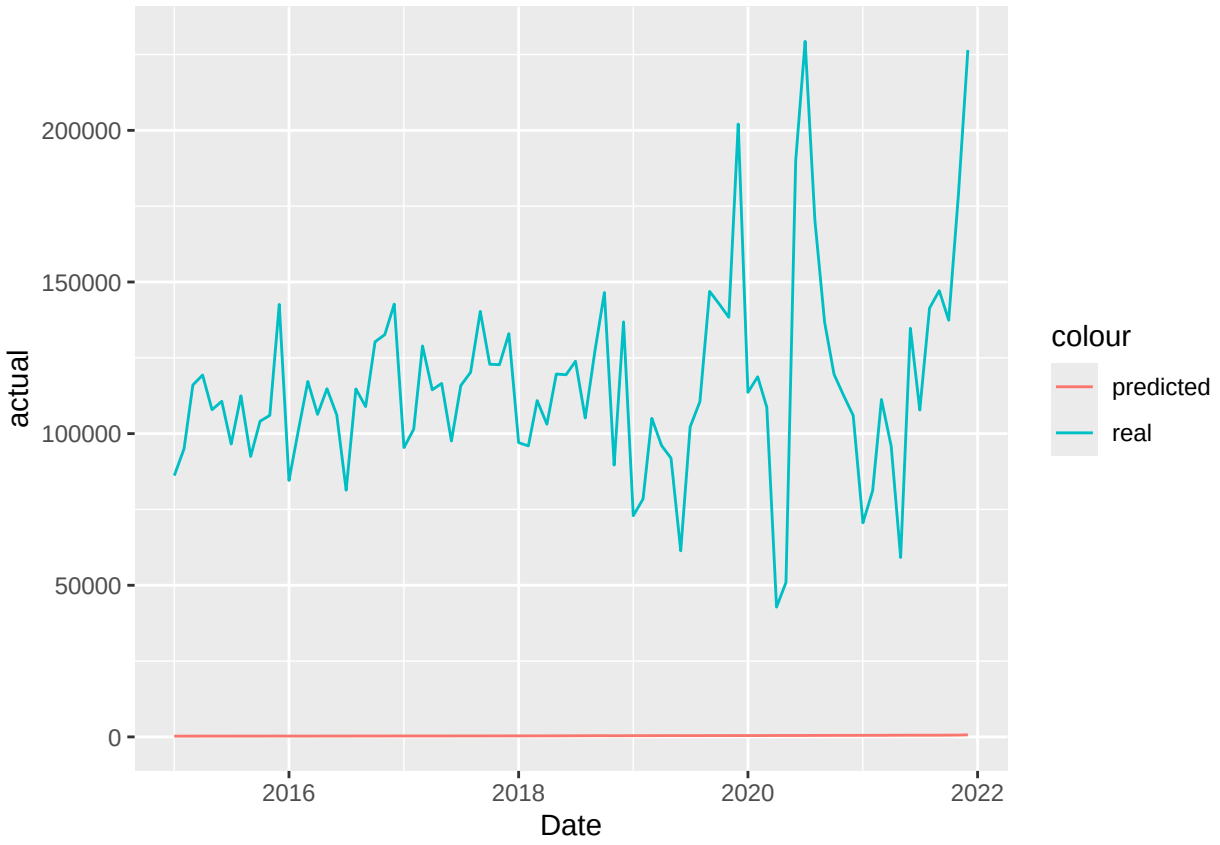
```
## year2020     -1.14213   13.13069  -0.087   0.9309
## year2021     27.54986   15.34330   1.796   0.0766 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.426 on 75 degrees of freedom
## Multiple R-squared:  0.9966, Adjusted R-squared:  0.9962
## F-statistic:  2711 on 8 and 75 DF,  p-value: < 2.2e-16
```

```
checkresiduals(model)
```

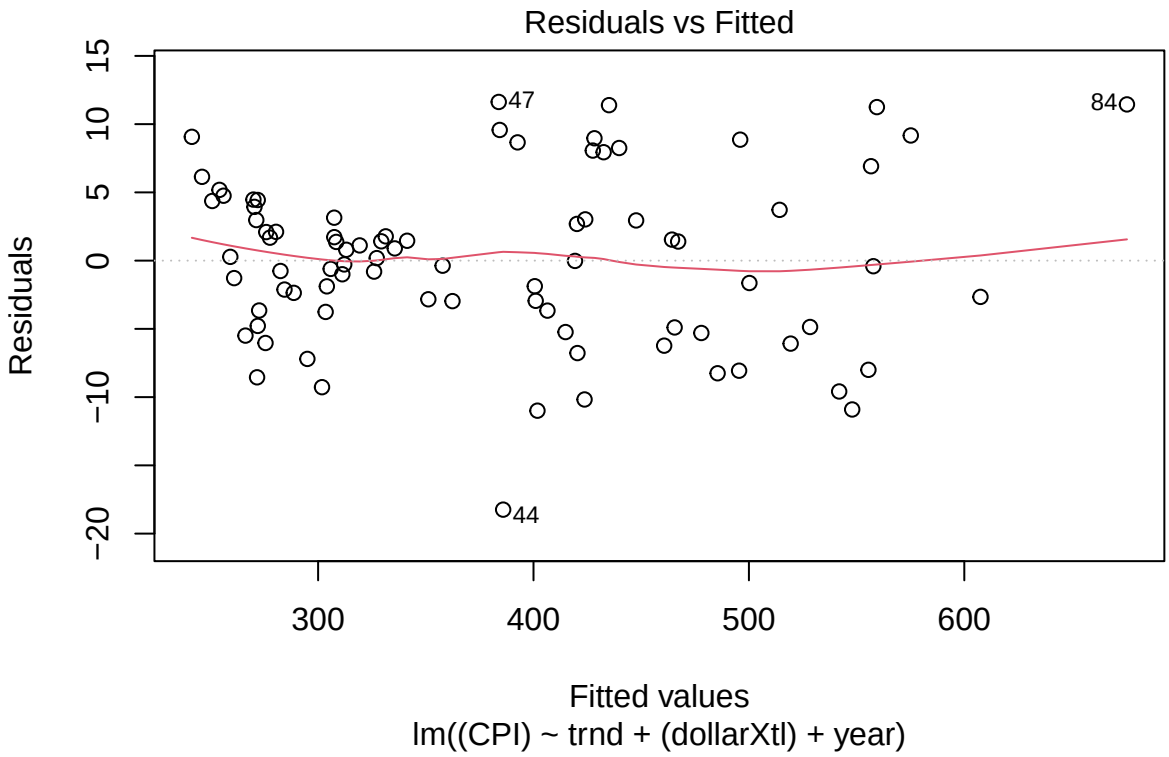


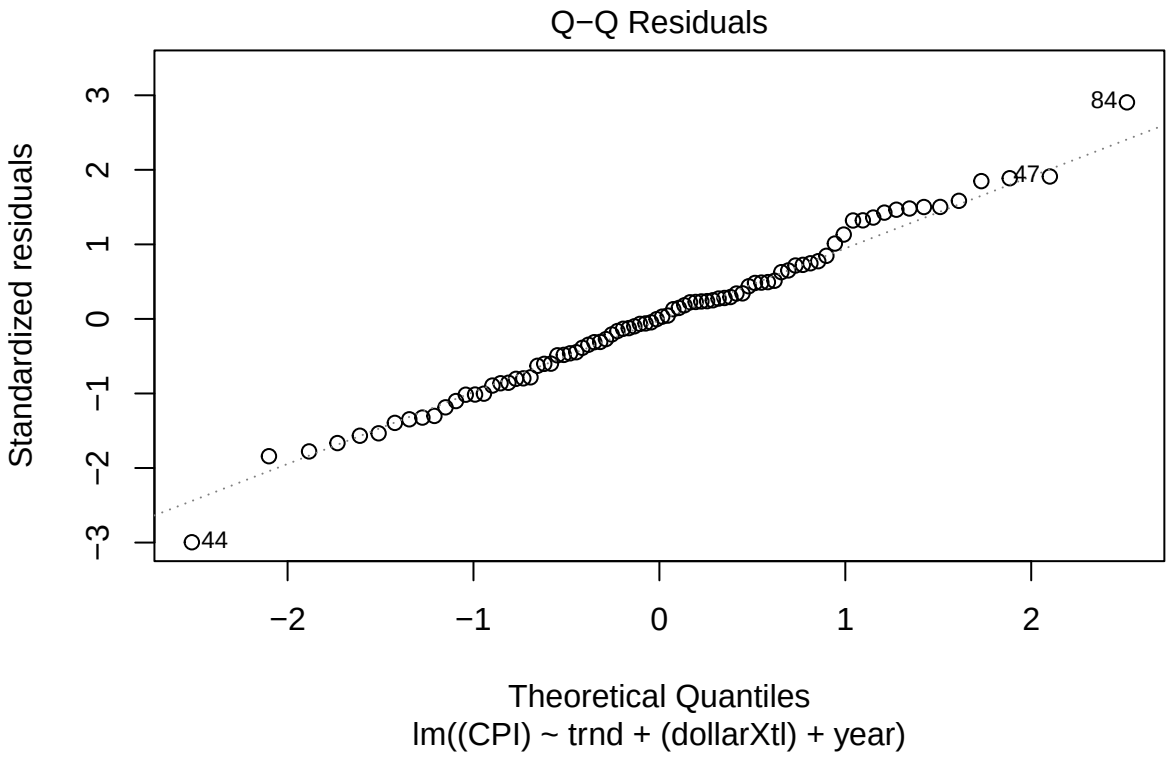

```
##
##  Breusch-Godfrey test for serial correlation of order up to 12
##
## data:  Residuals
## LM test = 43.399, df = 12, p-value = 1.932e-05
```

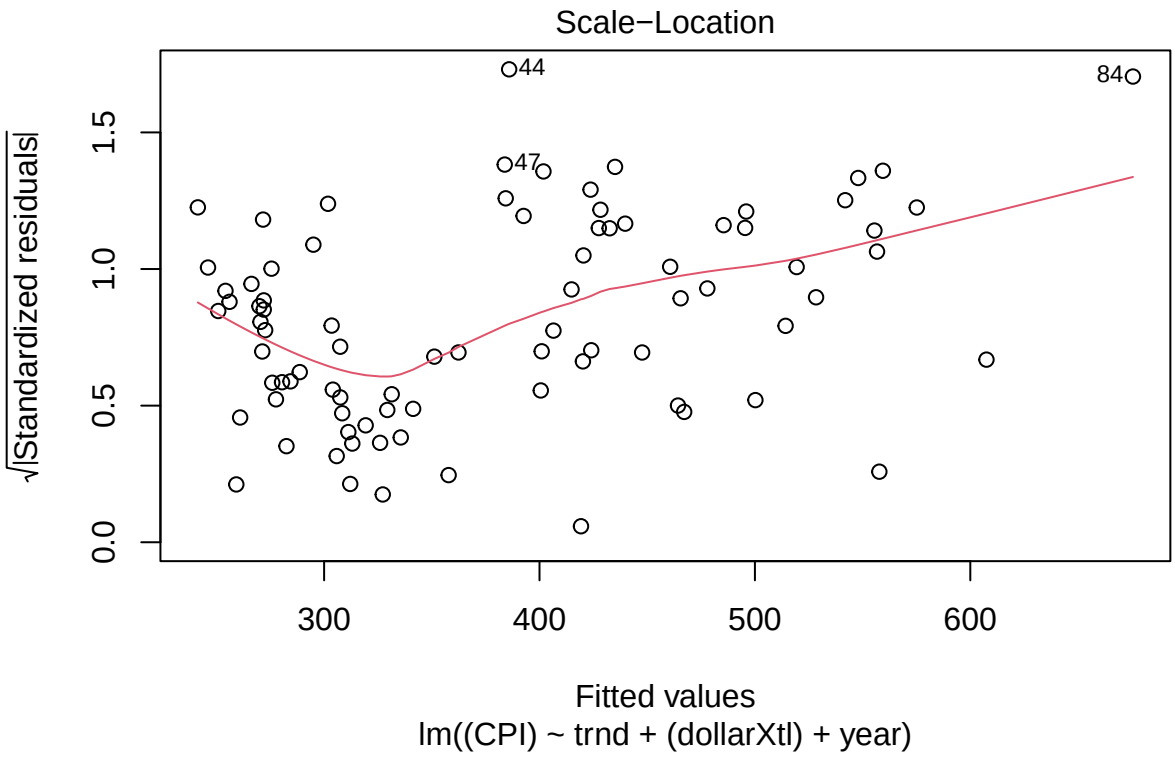
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=total_house_sales]
tmp_data[,predicted_trend_dollar_year:=predict(model,tmp_data)]
tmp_data[,residual_trend_dollar_year:=actual-predicted_trend_dollar_year]
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_dollar_year,color='predicted'))
```

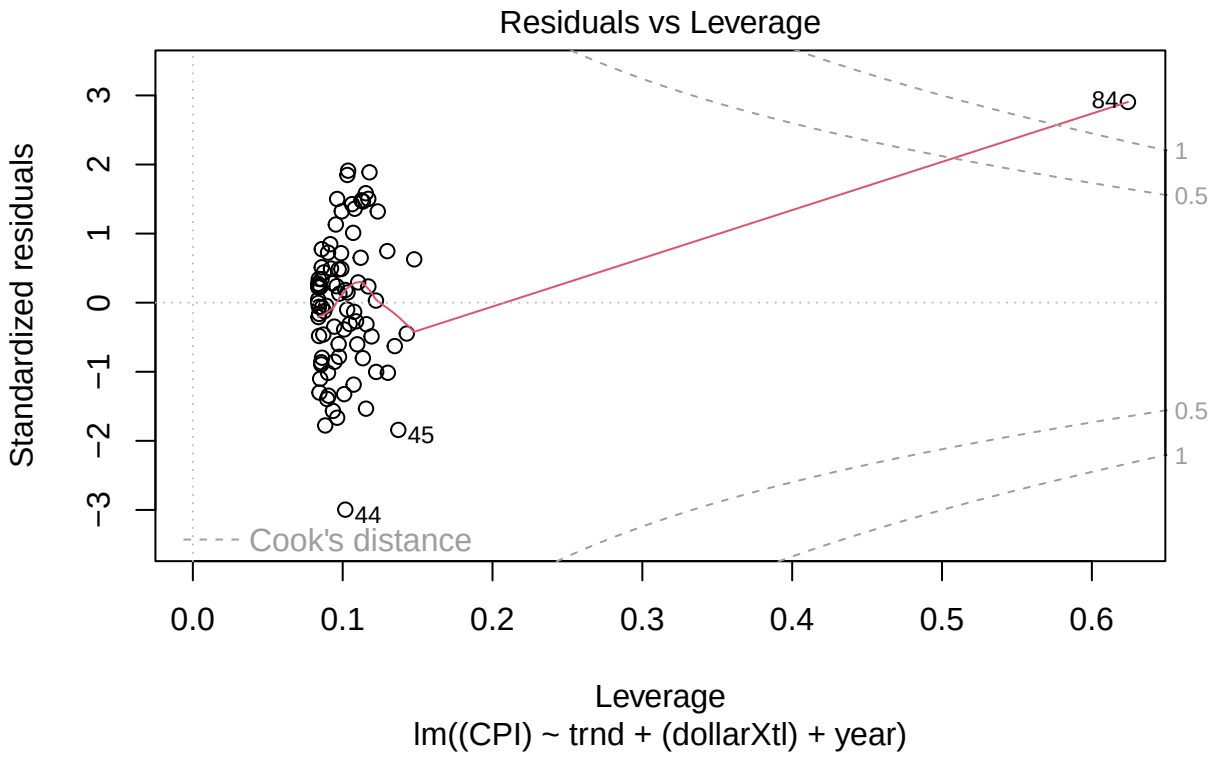

200000
150000
100000
50000
0
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
15
10
5
0
−10
−20
47
84
44
300
400
500
600
Fitted values
lm((CPI) ~ trnd + (dollarXtl) + year)


Q−Q Residuals
Standardized residuals
−3
−2
−1
0
1
2
3
84
47
44
−2
−1
0
1
2
Theoretical Quantiles
lm((CPI) ~ trnd + (dollarXtl) + year)

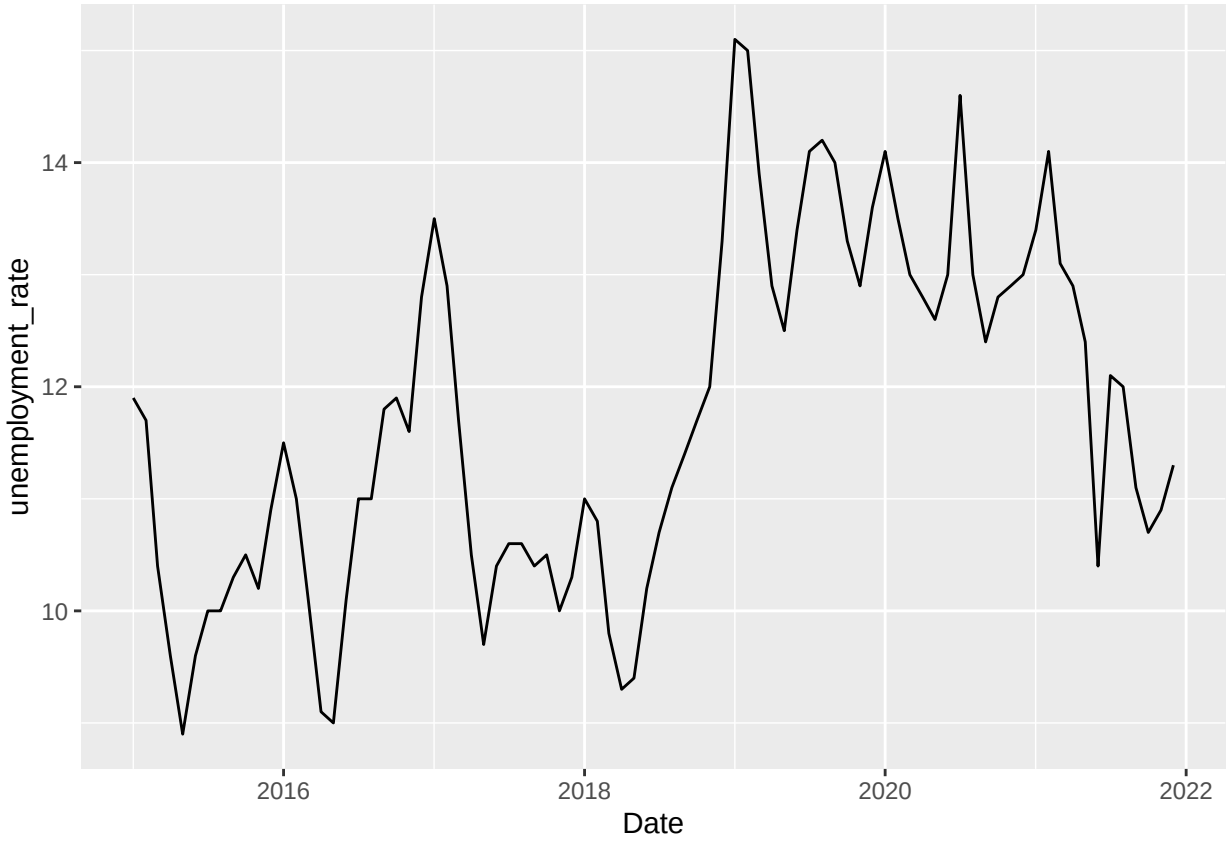

Scale−Location
44
84
47
√|Standardized residuals|
0.0
0.5
1.0
1.5
300
400
500
600
Fitted values
lm((CPI) ~ trnd + (dollarXtl) + year)



MODEL 3 (Unemployment Rate)

For the final model, I aimed to forecast the unemployment rate, with the following graph depicting the temporal trend of unemployment in Turkey.

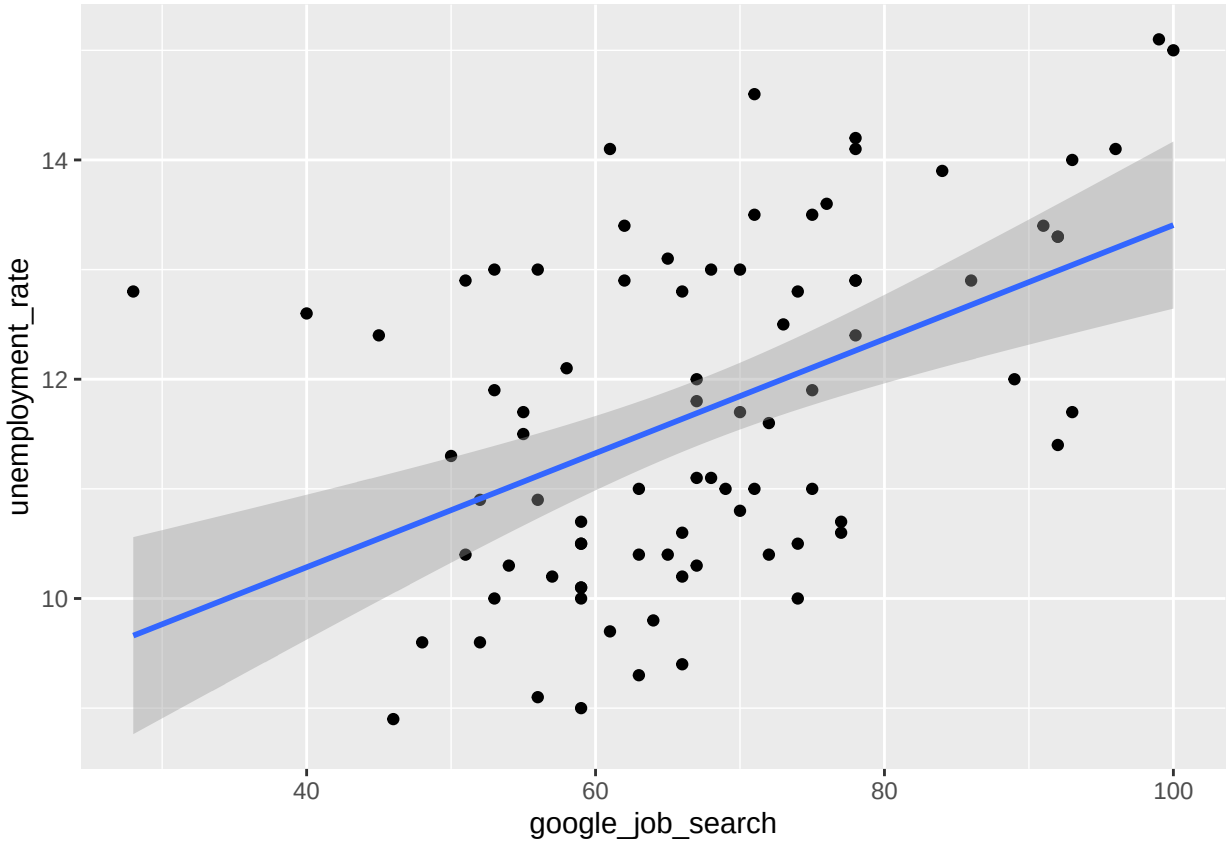
```
ggplot(all_data_table ,aes(x=Date,y=unemployment_rate, group=1)) + geom_line()
```



I concluded that the volume of job-related searches on Google could be a reliable predictor for the unemployment rate. Below is a scatter plot with a regression line that illustrates the relationship between the unemployment rate and the frequency of Google job searches.

```
ggplot(all_data_table ,aes(x=google_job_search,y=unemployment_rate)) + geom_point() +geom_smooth(method=
```

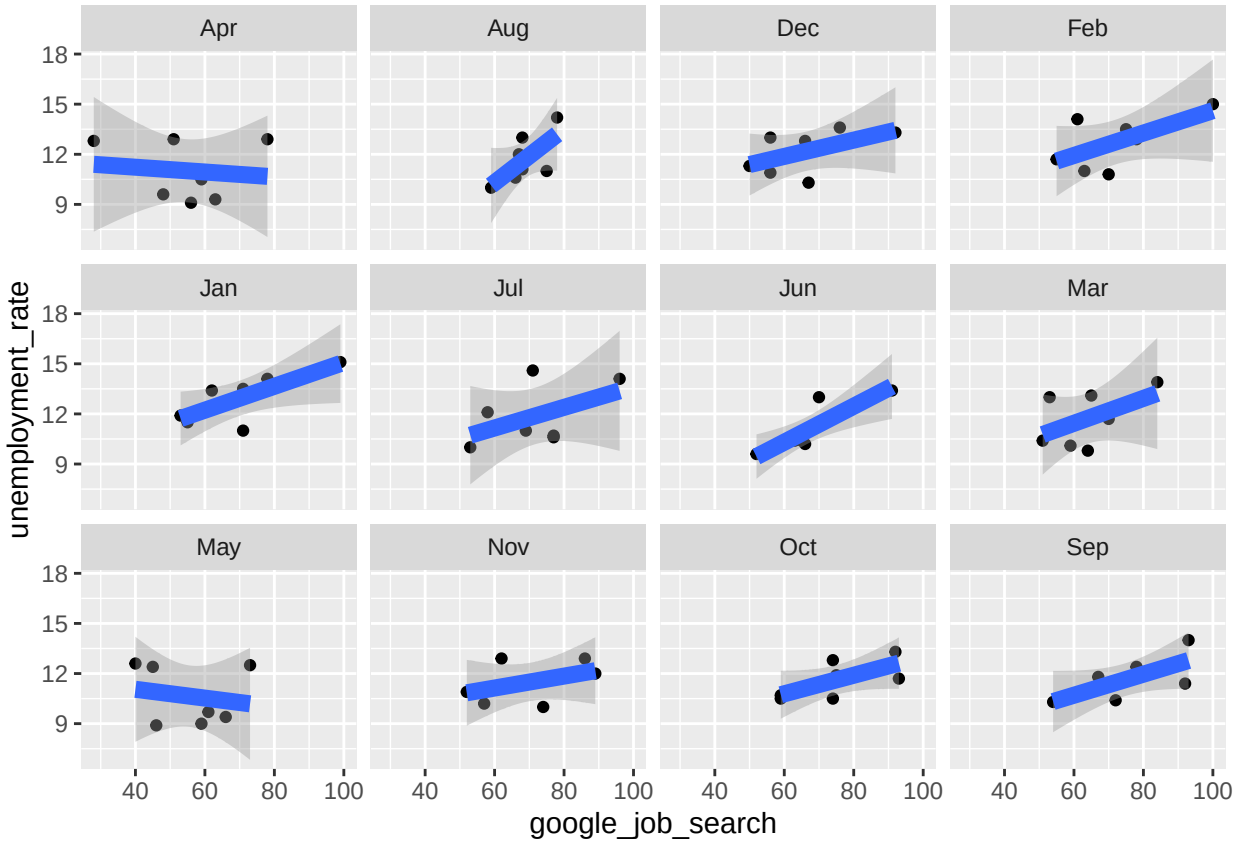
```
## `geom_smooth()` using formula = 'y ~ x'
```



The graphs below shows the correlation between number of searches in google and unemployment rate for each month.

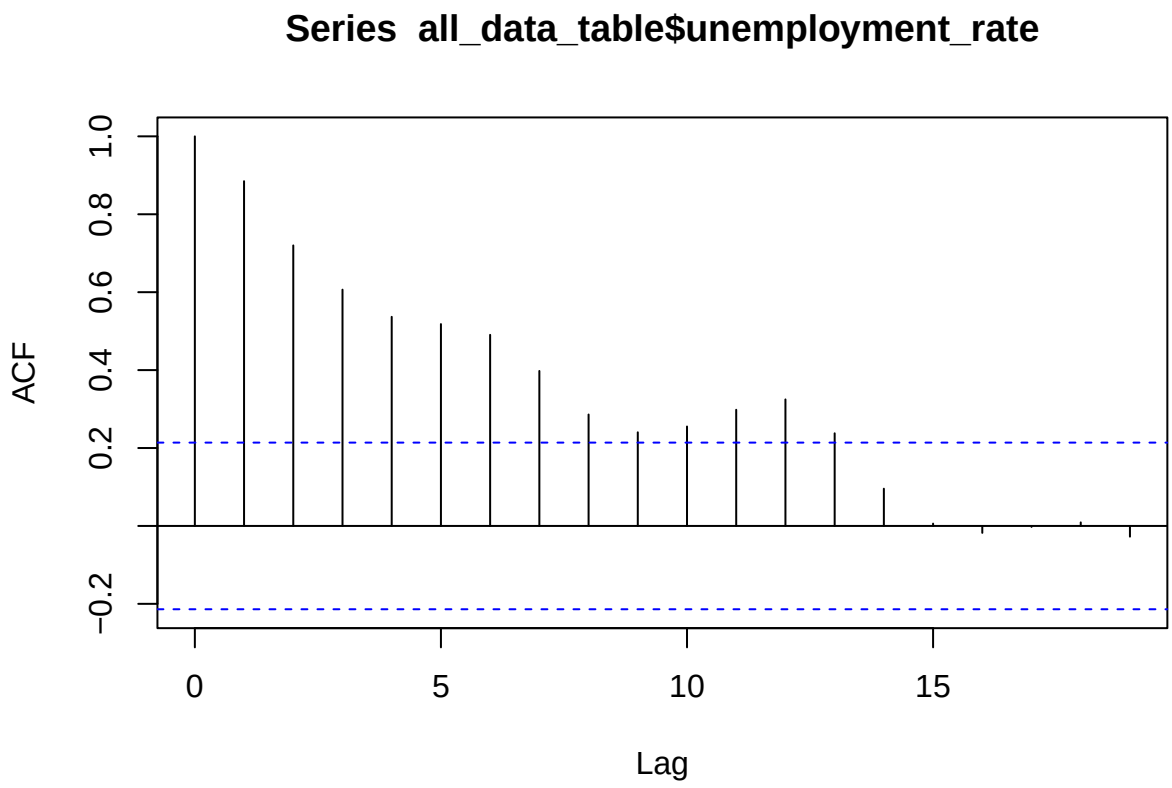
```
ggplot(all_data_table,aes(x=google_job_search,y=unemployment_rate)) +
        geom_point() + geom_smooth(method=lm,linewidth=3) + facet_wrap(~month)
```

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



ACF to see whether there is a correlation within the data itself.

```
acf(all_data_table$unemployment_rate)
```

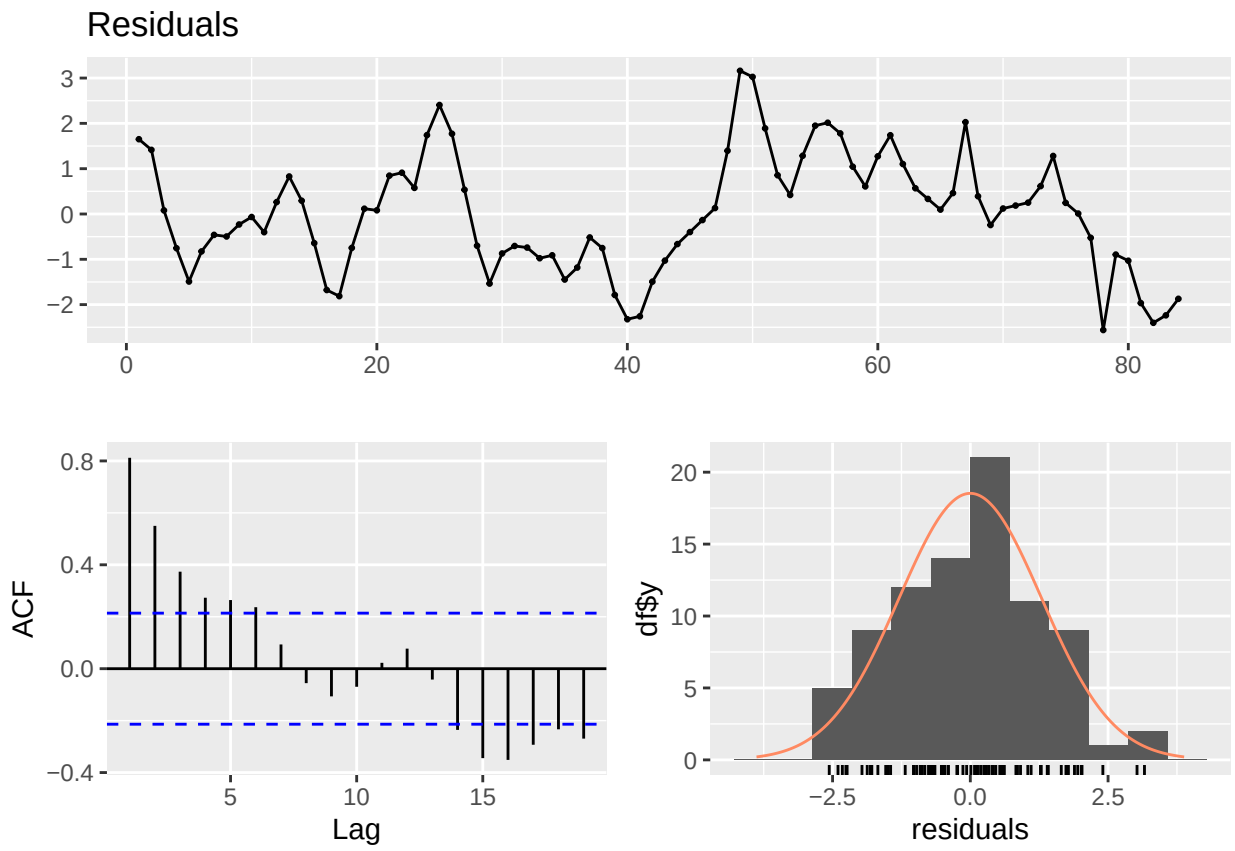


Let's start building the model. In this linear model, the unemployment rate is regressed on a time trend. The positive trend coefficient (0.0352, p-value < 2e-16) suggests a gradual increase in the unemployment rate over time. The model, with an R-squared of 0.3059, indicates that the trend explains about 30% of the variation in the unemployment rate.

```
model=lm(unemployment_rate~trnd,all_data_table)
summary(model)
```

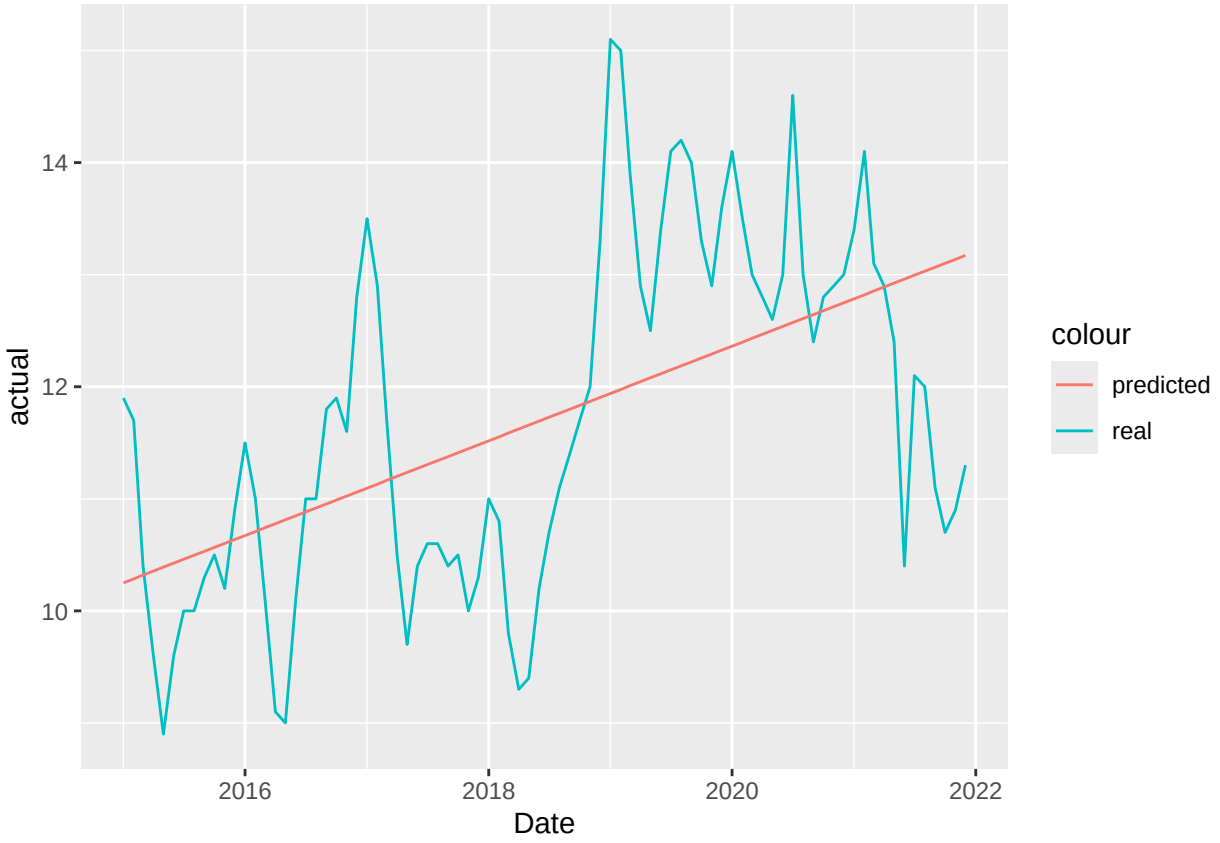
```
##
## Call:
## lm(formula = unemployment_rate ~ trnd, data = all_data_table)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.56046 -0.83699  0.08061  0.84834  3.16046
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.214544   0.286521  35.650  < 2e-16 ***
## trnd         0.035204   0.005856   6.012 4.87e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.301 on 82 degrees of freedom
## Multiple R-squared:  0.3059, Adjusted R-squared:  0.2975
## F-statistic: 36.14 on 1 and 82 DF,  p-value: 4.869e-08
```

```
checkresiduals(model$residuals)
```

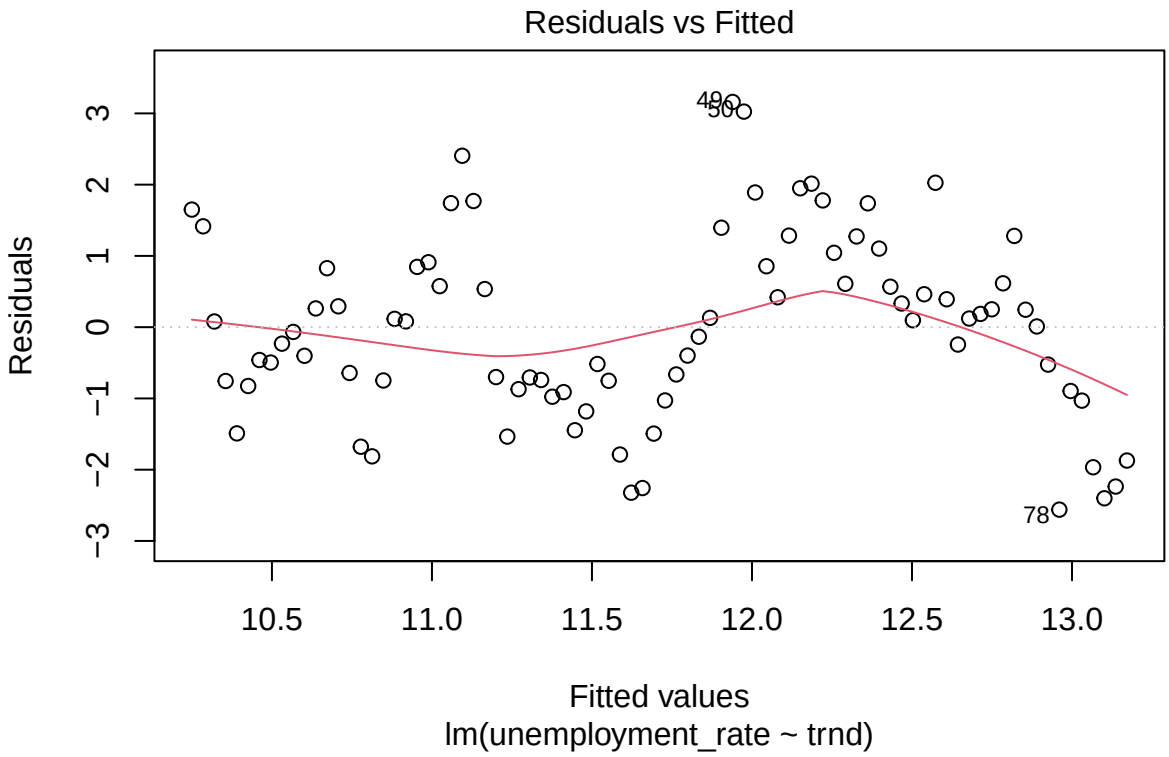


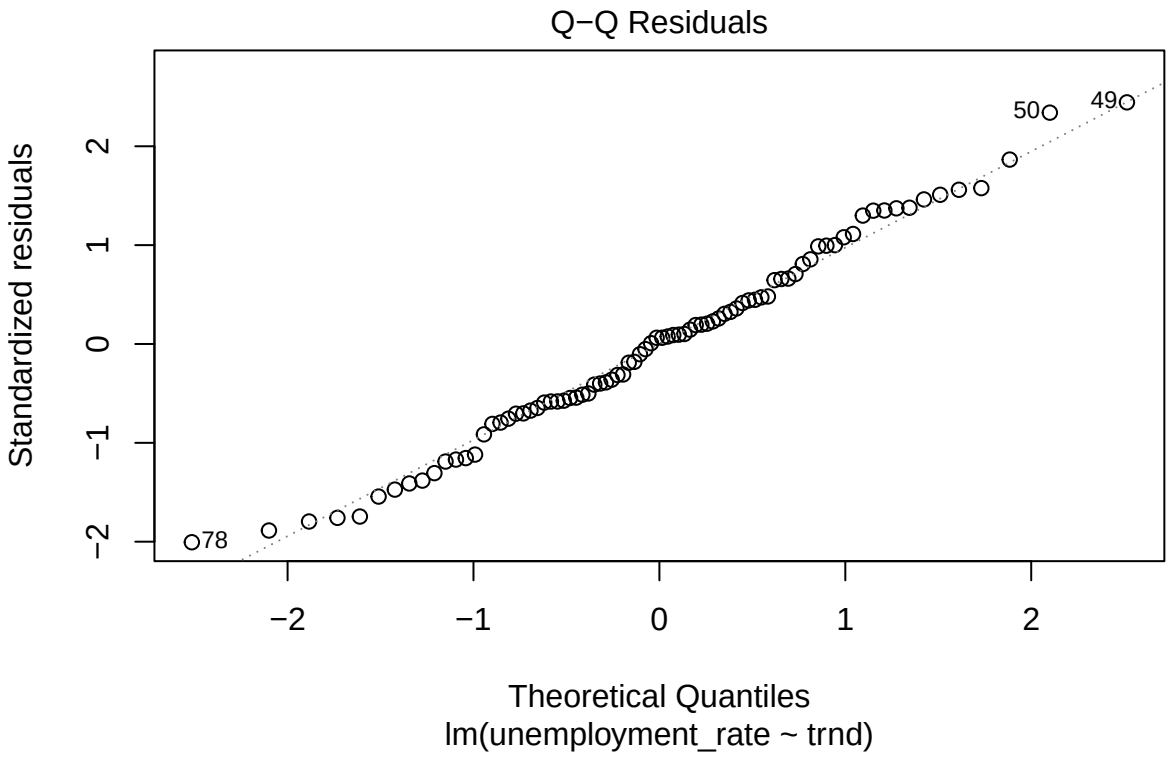

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 117.48, df = 10, p-value < 2.2e-16
##
## Model df: 0.   Total lags used: 10
```

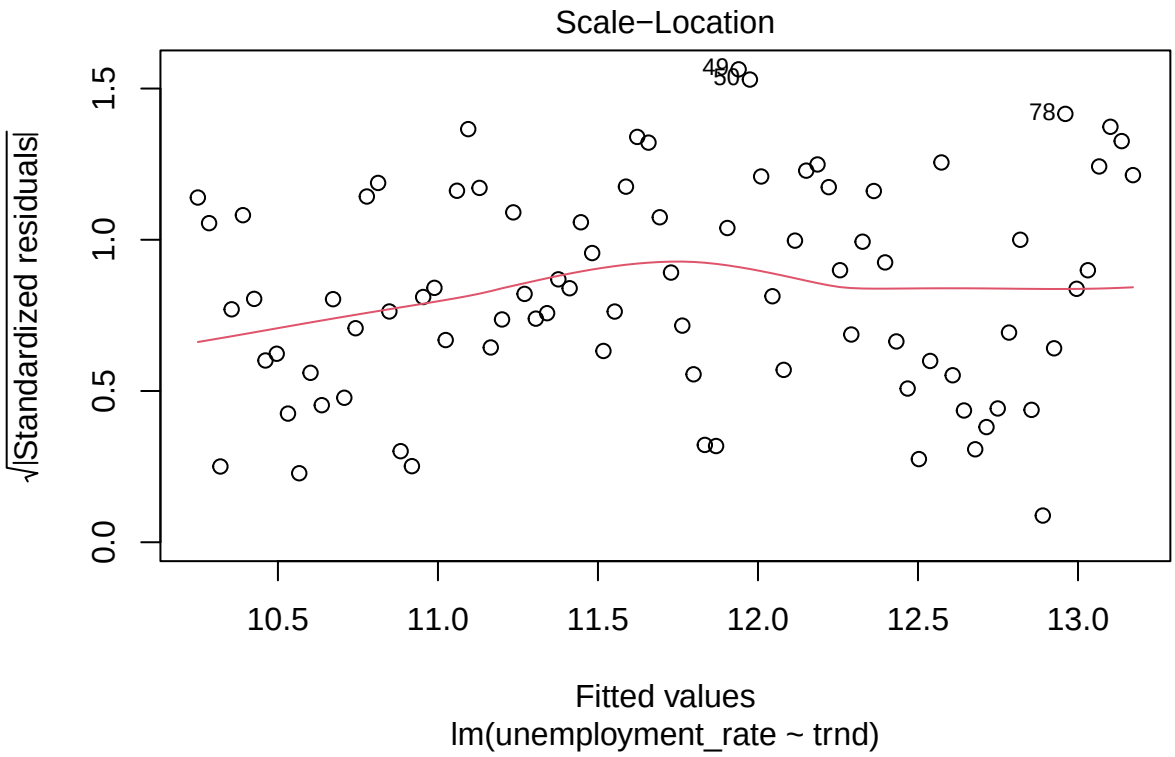
```
tmp_data=copy(all_data_table)
tmp_data[,actual:=unemployment_rate]
tmp_data[,predicted_trend:=predict(model,tmp_data)]
tmp_data[,residual_trend:=actual-predicted_trend]
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend,color='predicted'))
```

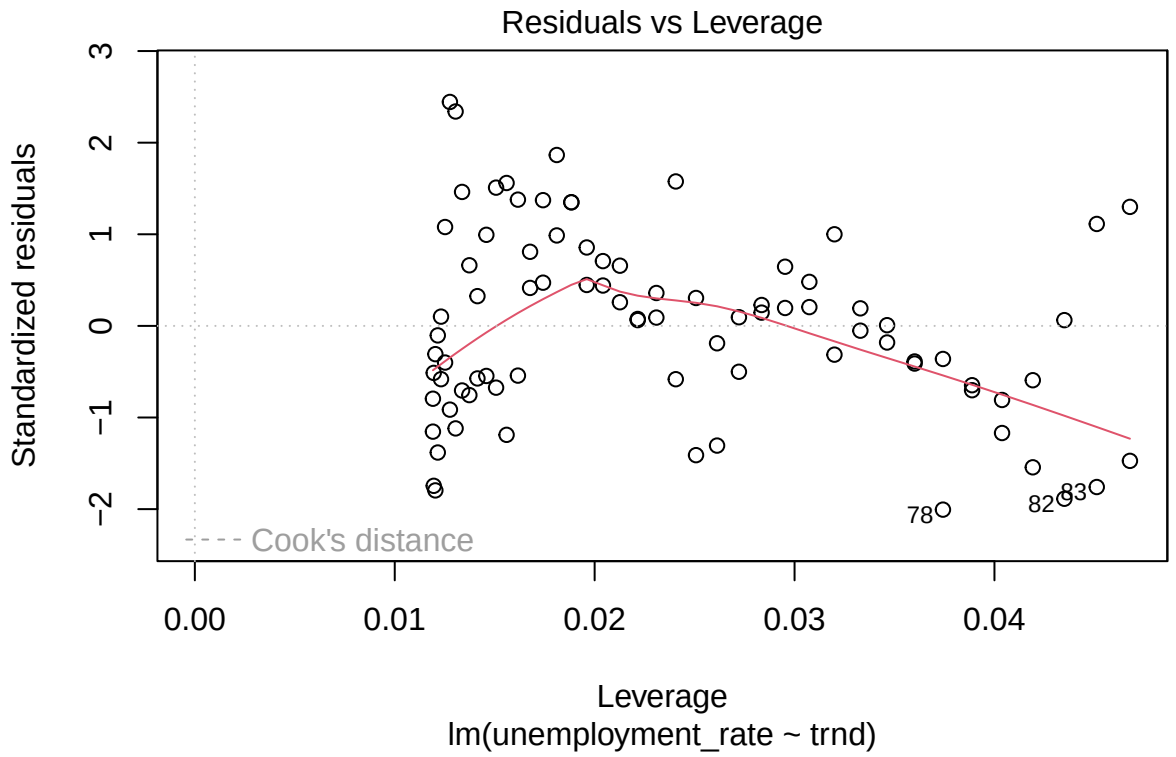

14
12
10
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(unemployment_rate ~ trnd)
49
50
78
10.5
11.0
11.5
12.0
12.5
13.0
−3
−2
−1
0
1
2
3


Q−Q Residuals
Standardized residuals
−2
−1
0
1
2
78
50
49
−2
−1
0
1
2
Theoretical Quantiles
lm(unemployment_rate ~ trnd)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
49
50
78
10.5
11.0
11.5
12.0
12.5
13.0
Fitted values
lm(unemployment_rate ~ trnd)



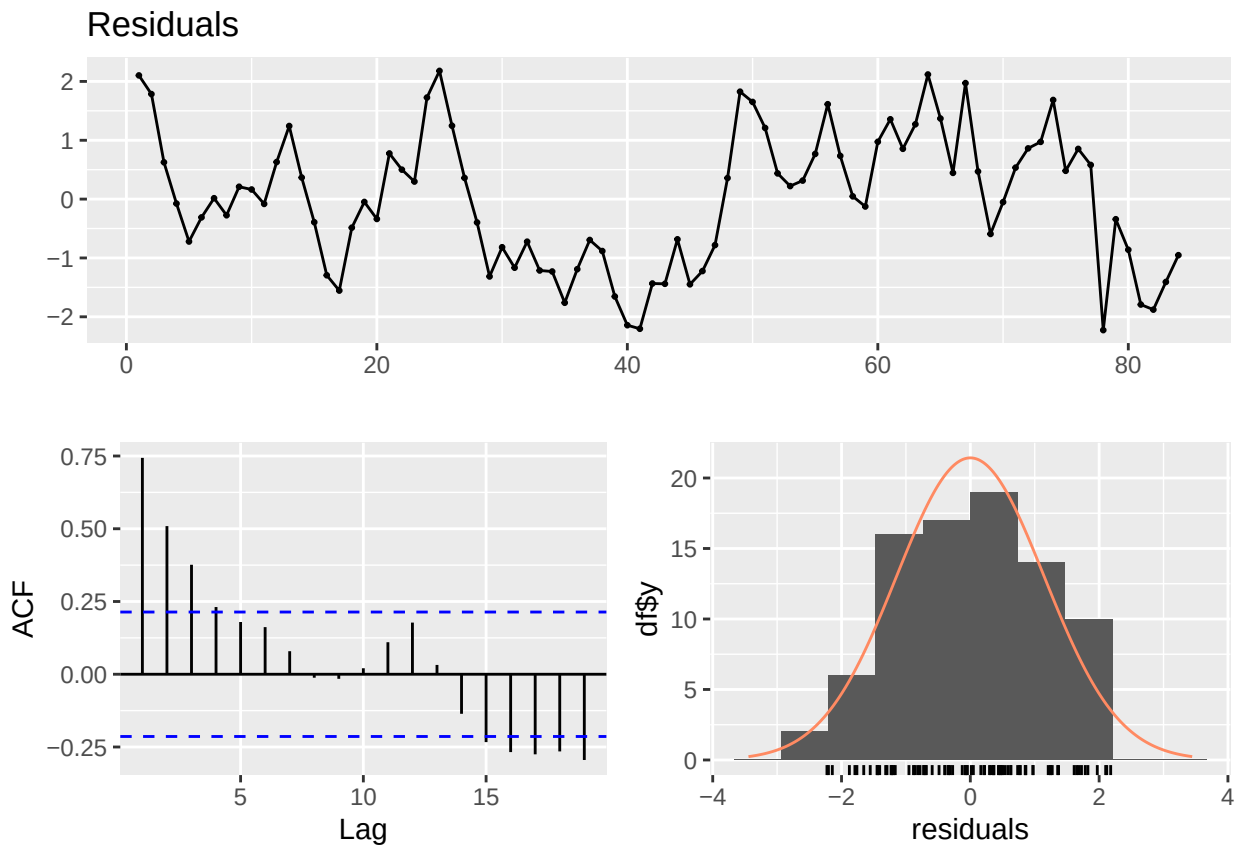
For next step add number of searches for jobs on google to the model. In this model, the unemployment rate is predicted using both the trend and Google job search data. The results show positive correlations for both trend (coefficient = 0.0311, p-value < 2e-16) and Google job search volume (coefficient = 0.0431, p-value = 1.20e-05), indicating that both factors are associated with an increase in unemployment rate. The model has an R-squared of 0.453, suggesting that these variables together explain approximately 45% of the variance in the unemployment rate.

```
model=lm((unemployment_rate)~trnd+(google_job_search),all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = (unemployment_rate) ~ trnd + (google_job_search),
##     data = all_data_table)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.22617 -0.86622  0.03106  0.79539  2.17921
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7.482880   0.638877  11.713  < 2e-16 ***
## trnd              0.031134   0.005303   5.871 9.13e-08 ***
## google_job_search 0.043092   0.009234   4.667 1.20e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.162 on 81 degrees of freedom
## Multiple R-squared:  0.453,  Adjusted R-squared:  0.4395
## F-statistic: 33.54 on 2 and 81 DF,  p-value: 2.448e-11
```

```
checkresiduals(model)
```

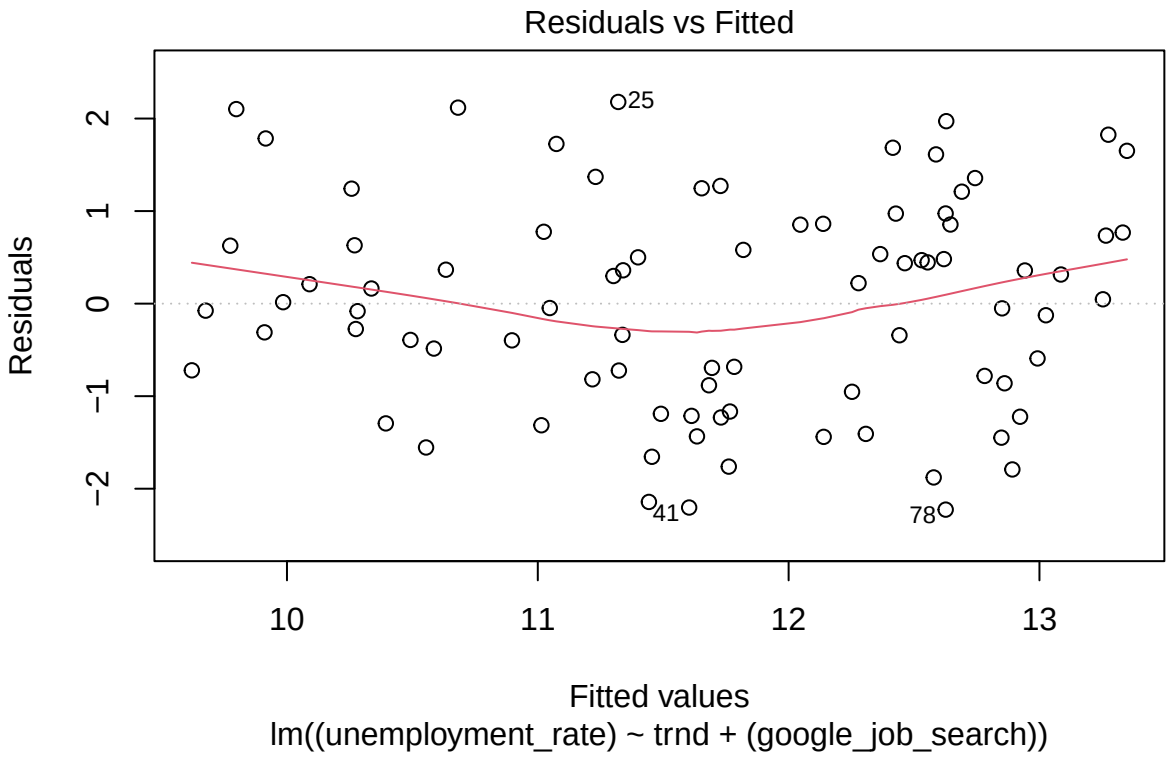


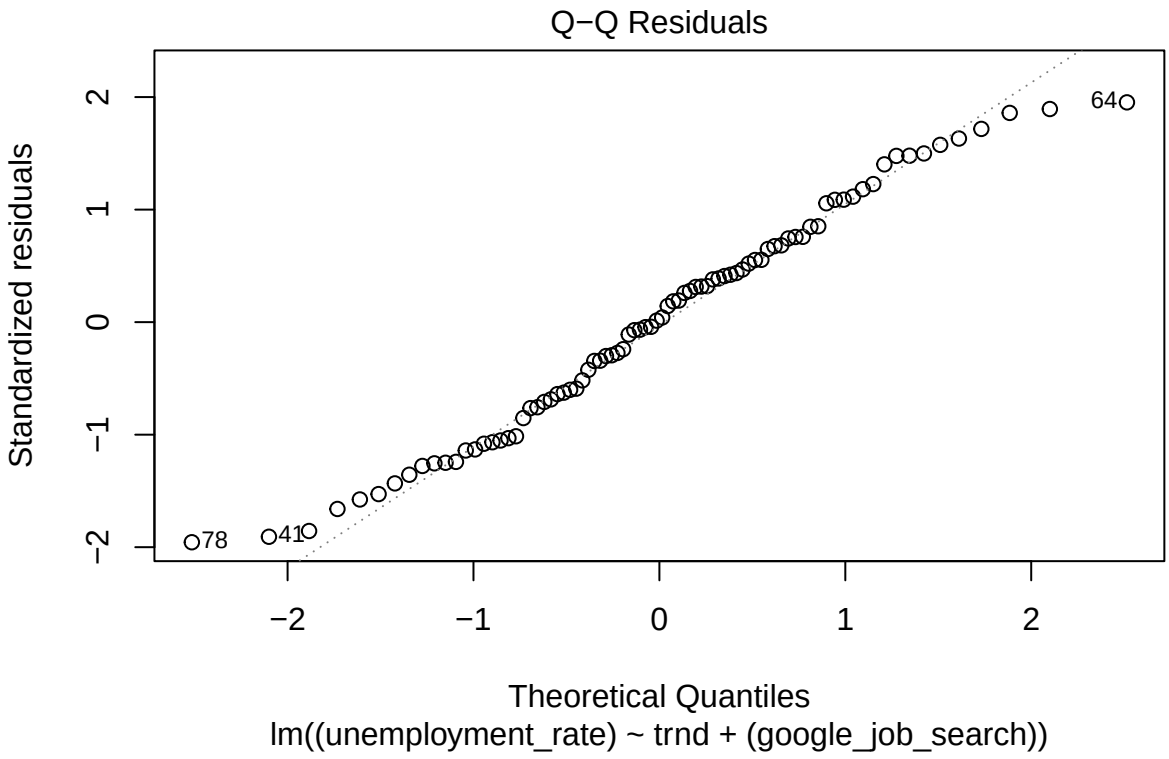

```
##
##  Breusch-Godfrey test for serial correlation of order up to 10
##
## data:  Residuals
## LM test = 50.463, df = 10, p-value = 2.193e-07
```

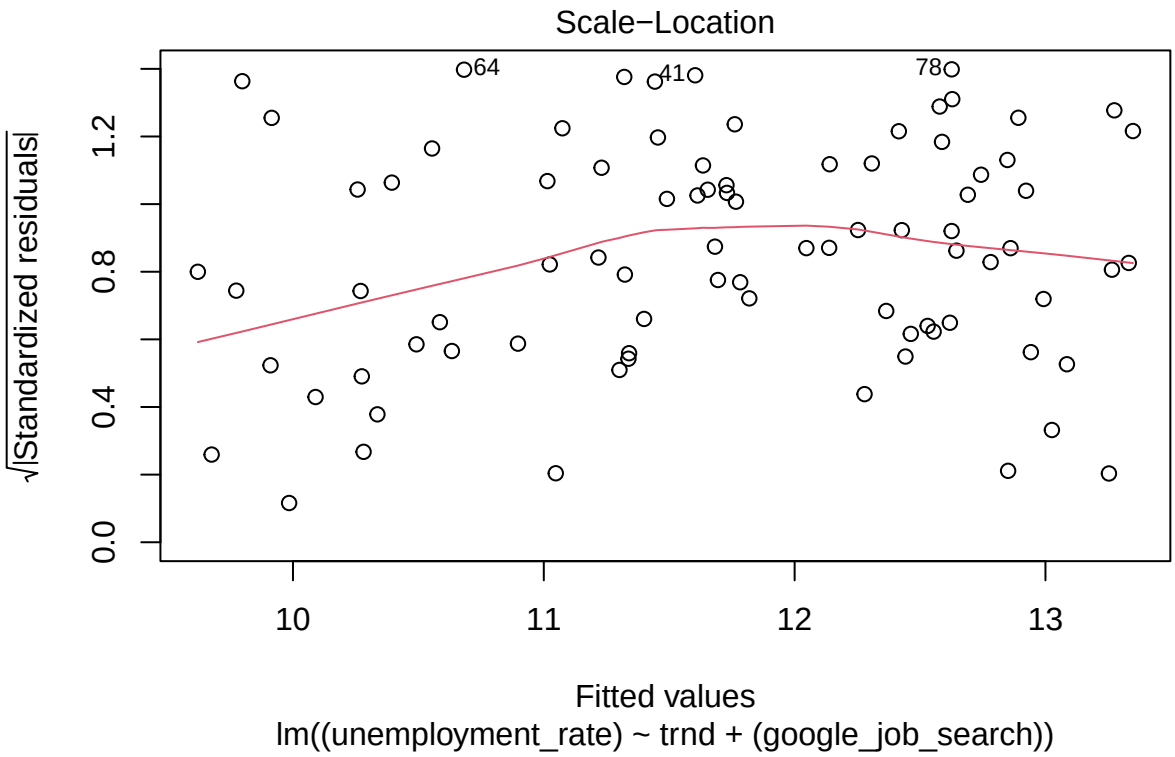
```
tmp_data = copy(all_data_table)
tmp_data[,actual:=unemployment_rate]
tmp_data[,predicted_trend_job_search_year:=predict(model,tmp_data)]
tmp_data[,residual_trend_job_search_year:=actual-predicted_trend_job_search_year]
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_job_search_year,color='predicted'))
```

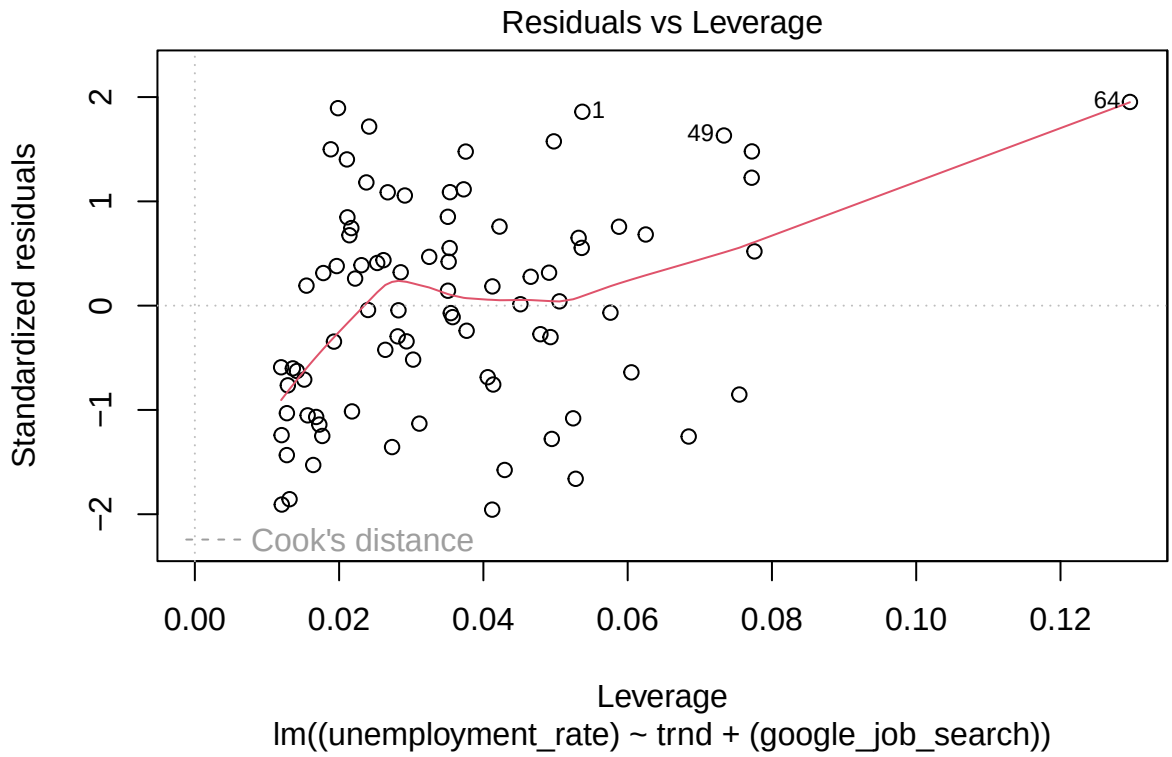

14
12
10
actual
2016
2018
2020
2022
Date
colour
predicted
real

```
plot(model)
```


Residuals vs Fitted
Residuals
Fitted values
lm((unemployment_rate) ~ trnd + (google_job_search))
25
41
78
10
11
12
13
−2
−1
0
1
2


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm((unemployment_rate) ~ trnd + (google_job_search))
78
41
64
−2
−1
0
1
2


Scale−Location
64
41
78
√|Standardized residuals|
0.0
0.4
0.8
1.2
10
11
12
13
Fitted values
lm((unemployment_rate) ~ trnd + (google_job_search))



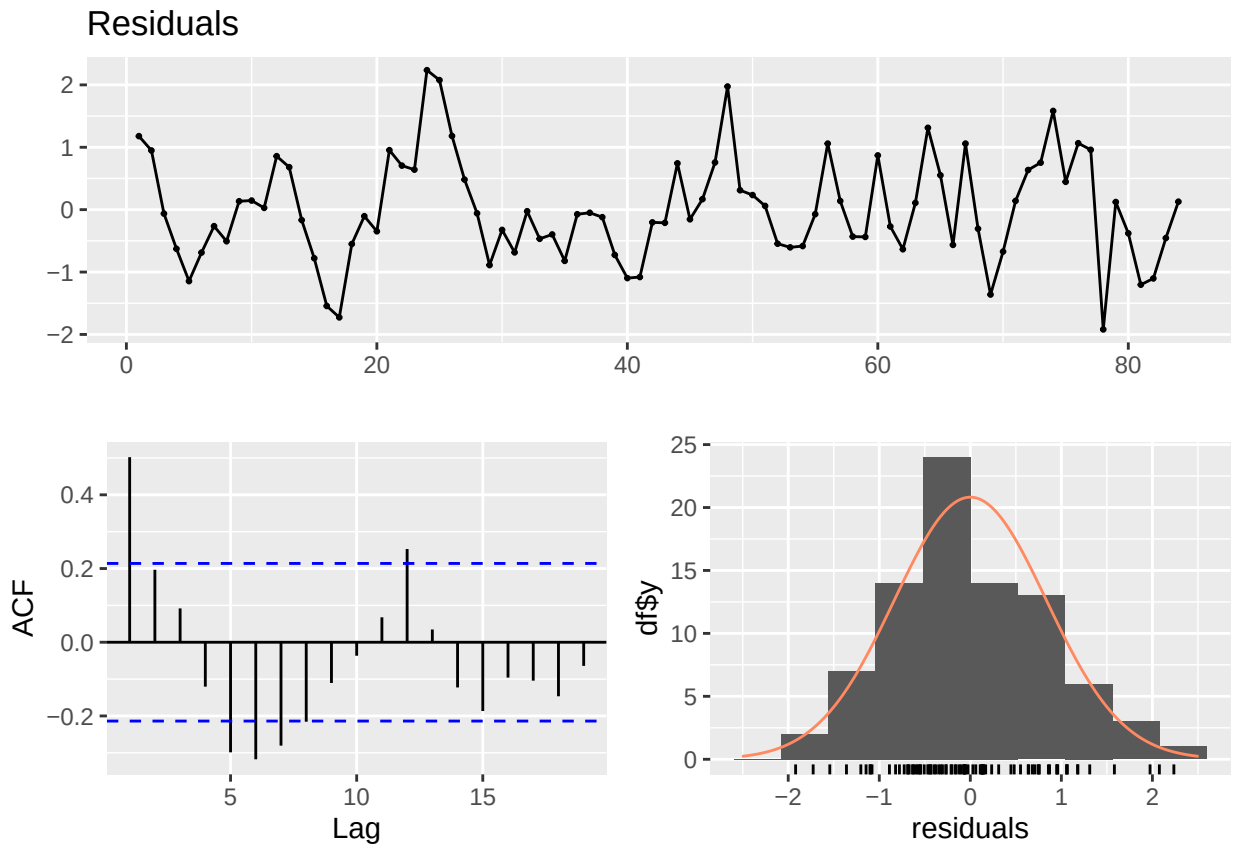
For the last step we are adding years to our model. This model predicts the unemployment rate using trend, Google job search data, and year. The negative trend coefficient suggests a slight decrease over time in the unemployment rate, but Google job searches positively correlate with unemployment, indicating that more searches are associated with higher unemployment rates. Yearly effects are significant, especially in later years, showing increasing unemployment rates. The model's R-squared value of 0.7102 means it explains 71% of the variance in unemployment, demonstrating a strong fit with the data.

```
model=lm(unemployment_rate~trnd+google_job_search+year,all_data_table)
summary(model)
```

```
##
## Call:
## lm(formula = unemployment_rate ~ trnd + google_job_search + year,
##     data = all_data_table)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.91905 -0.55374 -0.07277  0.63695  2.23559
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.98540    0.61872  12.906  < 2e-16 ***
## trnd              -0.07620    0.02852  -2.672 0.009245 **
## google_job_search  0.05306    0.01075   4.935 4.71e-06 ***
## year2016           0.90570    0.49169   1.842 0.069427 .
## year2017           1.57587    0.75761   2.080 0.040937 *
```

```
## year2018           2.11644    1.06291   1.991 0.050105 .
## year2019           5.28388    1.38240   3.822 0.000271 ***
## year2020           6.89385    1.72932   3.986 0.000154 ***
## year2021           6.93426    2.07371   3.344 0.001292 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8792 on 75 degrees of freedom
## Multiple R-squared:  0.7102, Adjusted R-squared:  0.6793
## F-statistic: 22.97 on 8 and 75 DF,  p-value: < 2.2e-16
```

```
checkresiduals(model$residuals)
```

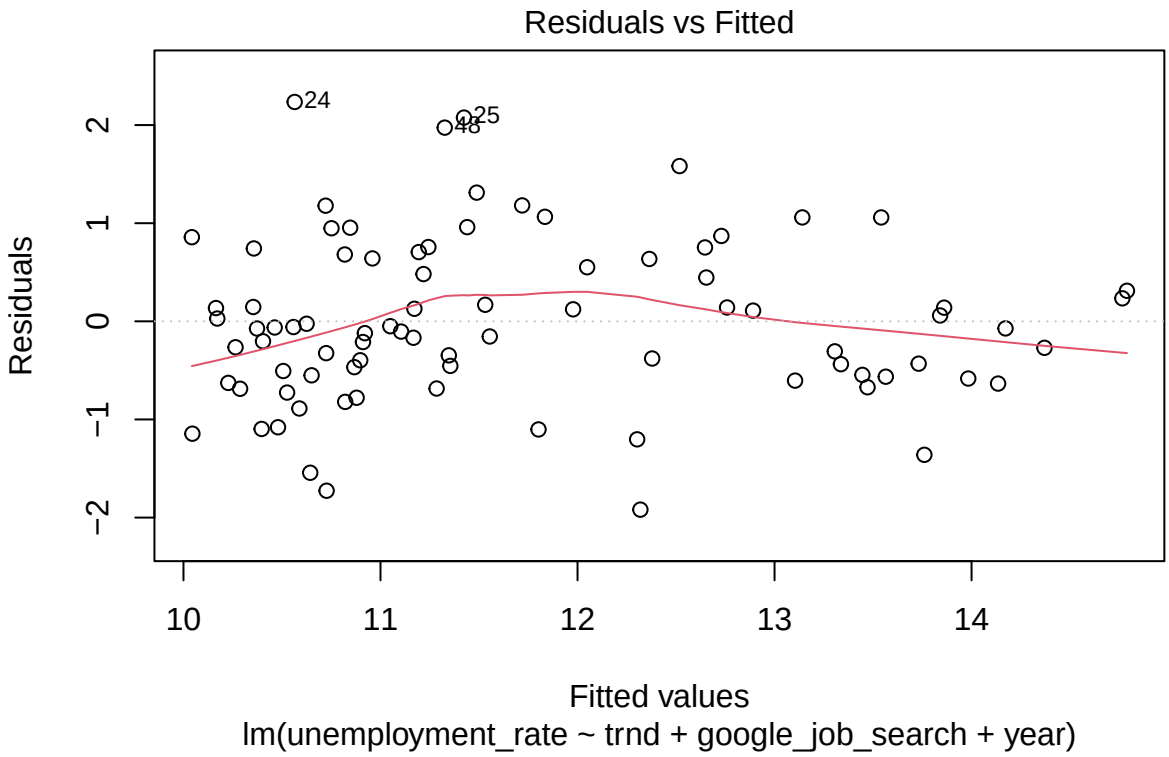


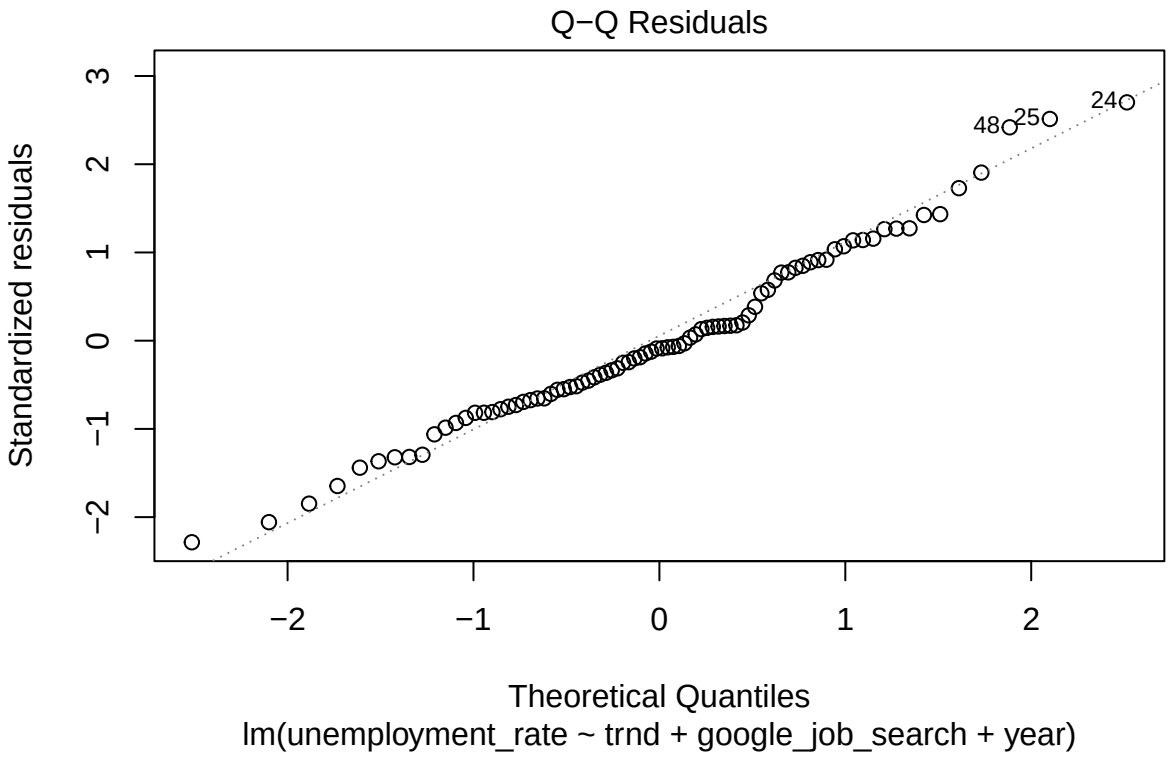

```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 58.014, df = 10, p-value = 8.591e-09
##
## Model df: 0.   Total lags used: 10
```

```
tmp_data=copy(all_data_table)
tmp_data[,actual:=unemployment_rate]
tmp_data[,predicted_trend_unemployment_rate_job_search_year:=predict(model,tmp_data)]
tmp_data[,residual_trend_unemployment_rate_job_search_year:=actual-predicted_trend_unemployment_rate_jo
```

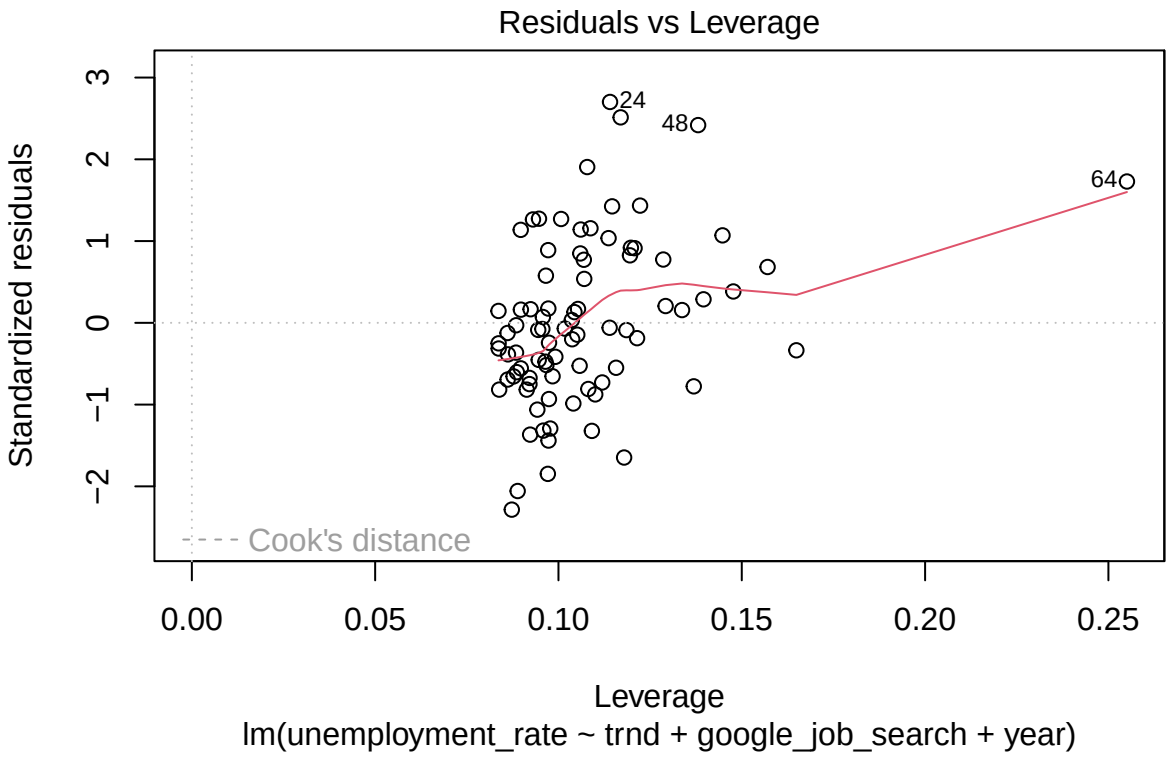
```
#head(tmp)
ggplot(tmp_data ,aes(x=Date)) +
        geom_line(aes(y=actual,color='real')) +
        geom_line(aes(y=predicted_trend_unemployment_rate_job_search_year,color='predicted'))
```


```
plot(model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(unemployment_rate ~ trnd + google_job_search + year)
24
25
48
2
1
0
−1
−2
10
11
12
13
14


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(unemployment_rate ~ trnd + google_job_search + year)
48
25
24


Scale−Location
24
25
48
√|Standardized residuals|
0.0
0.5
1.0
1.5
10
11
12
13
14
Fitted values
lm(unemployment_rate ~ trnd + google_job_search + year)



Conclusion

In the first model, focusing on total house sales in Turkey, we found a slight upward trend, suggesting a gradual increase in sales over time. However, the model's accuracy, as indicated by an R-squared value of around 0.05, was relatively low, pointing to other influential factors not captured in the analysis. The residuals, with a wide range, hinted at significant variability unexplained by the model. Autocorrelation was also evident, suggesting that past sales figures have some influence on future ones, a detail not fully accounted for in the model.

The second analysis delved into the Consumer Price Index (CPI) and demonstrated a substantial improvement in modeling accuracy, with an R-squared value close to 0.99. This high R-squared value indicated that the model, which included the trend and exchange rate variables, could account for most of the variability in the CPI. The residuals were considerably smaller, implying a better fit of the model to the data. The precise estimation of CPI changes highlighted the strong impact of the included variables, although the exact autocorrelation values were not specified, the close fit suggests that the model captured the underlying pattern effectively.

The third model aimed to predict the unemployment rate, incorporating variables such as trend, Google job search data, and annual changes. This model achieved an R-squared value of about 0.71, indicating a good level of fit and a substantial improvement over the first model. It managed to capture significant year-to-year fluctuations and showed that Google search trends for jobs had a notable effect on the unemployment rate. The model's residuals were smaller and more consistent than in the first model, indicating less unexplained variance. However, like the other models, it exhibited some degree of autocorrelation, suggesting that incorporating additional lagged variables or adjusting the model could potentially improve its predictive power.