Final Project

Shuxin Tan 1007625447

2023-08-04

#Introduction:

#Exploratory data analysis section:

#Model development section:

#Condlusion section:

```
setwd("/Users/tanshuxin/Desktop/Second Year s/STA302/final project")

library(tidyverse)
```

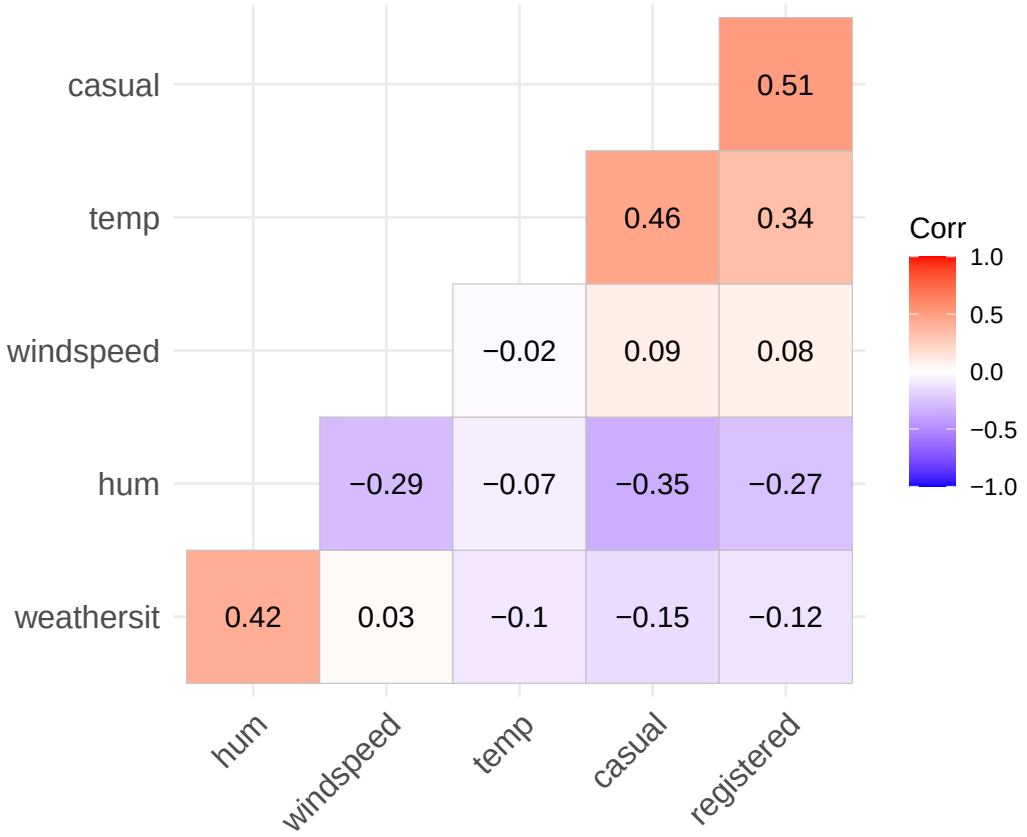
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
bike=read.csv('hour.csv', header=TRUE)

attach(bike)

summary(bike)
```

```
##     instant         dteday              season           yr
##  Min.   :    1   Length:17379       Min.   :1.000   Min.   :0.0000
##  1st Qu.: 4346   Class :character   1st Qu.:2.000   1st Qu.:0.0000
##  Median : 8690   Mode  :character   Median :3.000   Median :1.0000
##  Mean   : 8690                      Mean   :2.502   Mean   :0.5026
##  3rd Qu.:13034                      3rd Qu.:3.000   3rd Qu.:1.0000
##  Max.   :17379                      Max.   :4.000   Max.   :1.0000
##       mnth              hr           holiday           weekday
##  Min.   : 1.000   Min.   : 0.00   Min.   :0.00000   Min.   :0.000
##  1st Qu.: 4.000   1st Qu.: 6.00   1st Qu.:0.00000   1st Qu.:1.000
##  Median : 7.000   Median :12.00   Median :0.00000   Median :3.000
##  Mean   : 6.538   Mean   :11.55   Mean   :0.02877   Mean   :3.004
##  3rd Qu.:10.000   3rd Qu.:18.00   3rd Qu.:0.00000   3rd Qu.:5.000
##  Max.   :12.000   Max.   :23.00   Max.   :1.00000   Max.   :6.000
##    workingday       weathersit         temp           atemp
##  Min.   :0.0000   Min.   :1.000   Min.   :0.020   Min.   :0.0000
```

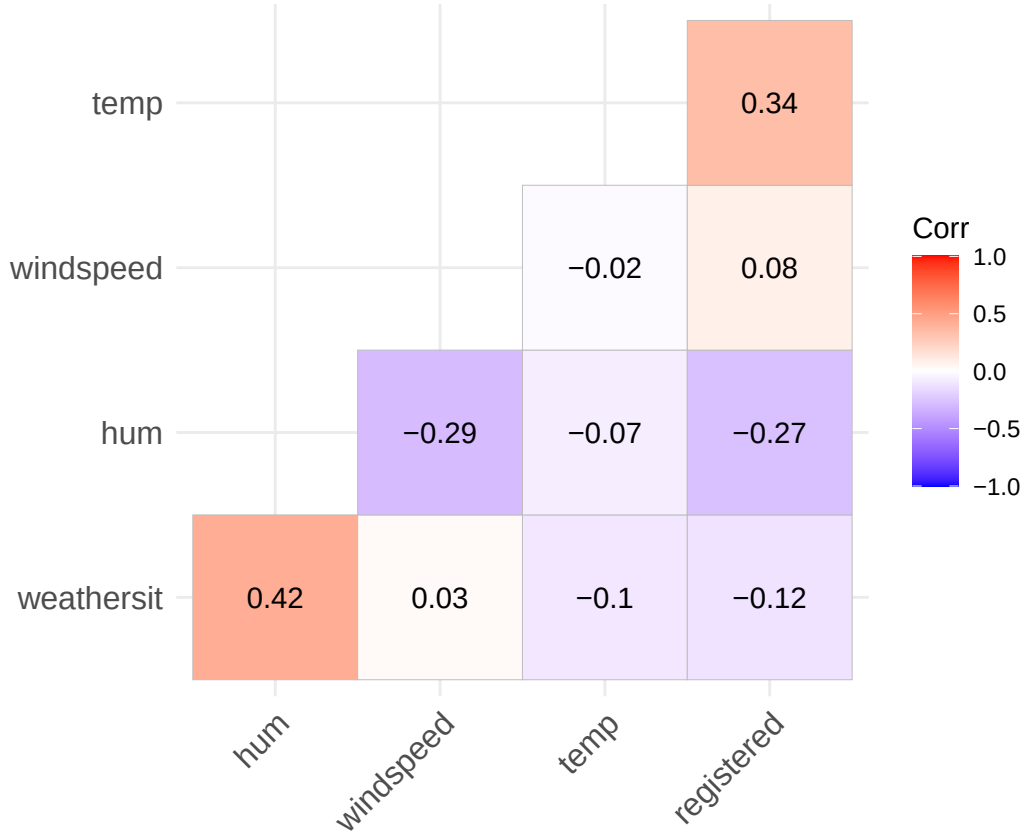
```
##  1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:0.340   1st Qu.:0.3333
##  Median :1.0000   Median :1.000   Median :0.500   Median :0.4848
##  Mean   :0.6827   Mean   :1.425   Mean   :0.497   Mean   :0.4758
##  3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:0.660   3rd Qu.:0.6212
##  Max.   :1.0000   Max.   :4.000   Max.   :1.000   Max.   :1.0000
##       hum          windspeed          casual         registered
##  Min.   :0.0000   Min.   :0.0000   Min.   :  0.00   Min.   :  0.0
##  1st Qu.:0.4800   1st Qu.:0.1045   1st Qu.:  4.00   1st Qu.: 34.0
##  Median :0.6300   Median :0.1940   Median : 17.00   Median :115.0
##  Mean   :0.6272   Mean   :0.1901   Mean   : 35.68   Mean   :153.8
##  3rd Qu.:0.7800   3rd Qu.:0.2537   3rd Qu.: 48.00   3rd Qu.:220.0
##  Max.   :1.0000   Max.   :0.8507   Max.   :367.00   Max.   :886.0
##       cnt
##  Min.   :  1.0
##  1st Qu.: 40.0
##  Median :142.0
##  Mean   :189.5
##  3rd Qu.:281.0
##  Max.   :977.0
```

```
library(ggcorrplot)
reduced_data <- subset(bike, select = c('weathersit', 'temp', 'hum', 'windspeed', 'casual', 'registered
corr_matrix = round(cor(reduced_data), 2)
ggcorrplot(corr_matrix, hc.order = TRUE, type = "lower",
           lab = TRUE)
```

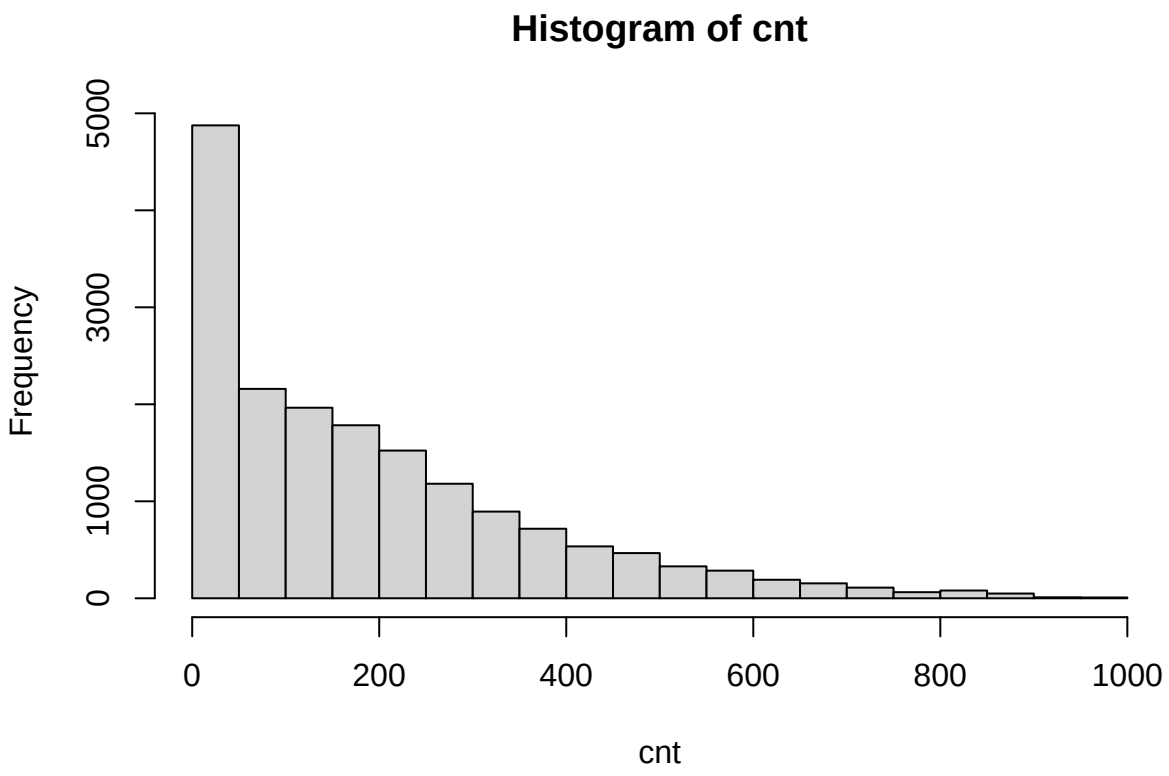



```
reduced_data0 <- subset(bike, select = c('weathersit', 'temp', 'hum', 'windspeed', 'registered'))
corr_matrix = round(cor(reduced_data0), 2)
```

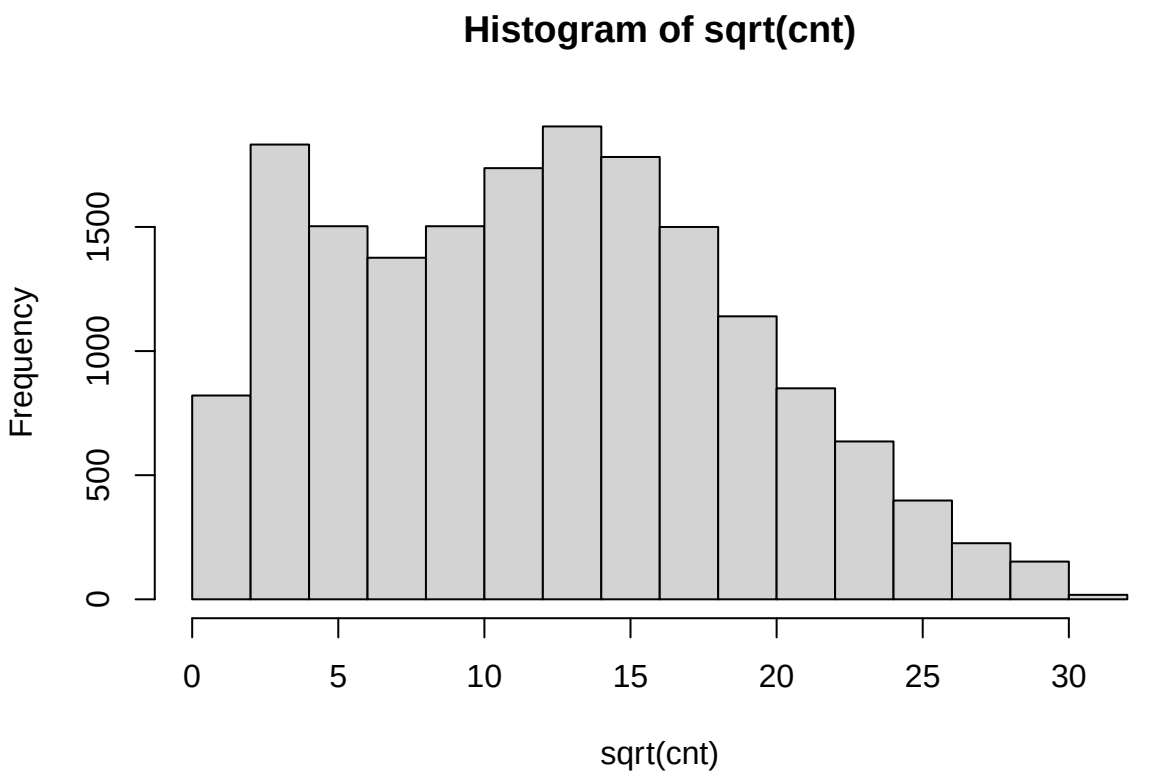
```
ggcorrplot(corr_matrix, hc.order = TRUE, type = "lower",
           lab = TRUE)
```




```
#drop the predictor variable "casual" because cor(casual, registered)=0.51 and cor(casual, temp)=0.46 i

hist(cnt)
```




```
#Since the data of cnt is more right-skewed, then we apply the squared root method to make it more cent
hist(sqrt(cnt))
```

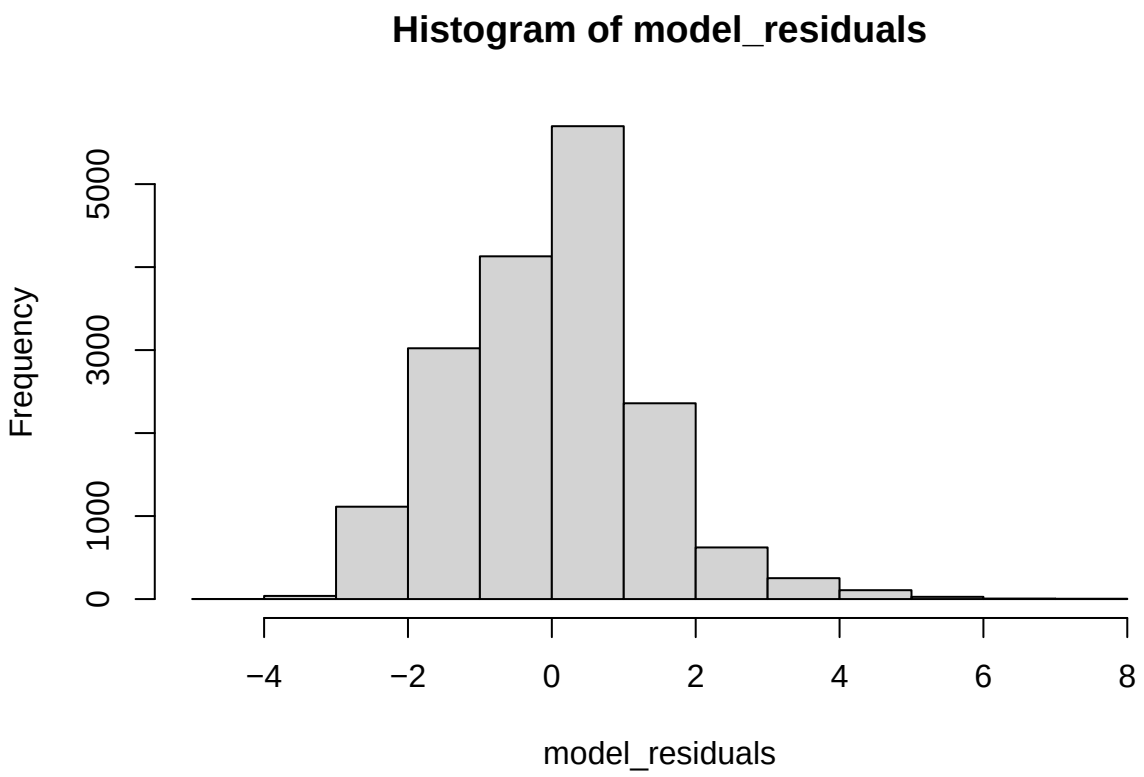



```
#After applying the squared root method, the plot of the sqrt(cnt) is more like a normal distribution s

model <- lm(sqrt(cnt) ~ weathersit+poly(temp,2)+hum+poly(registered,2), data=bike)
```

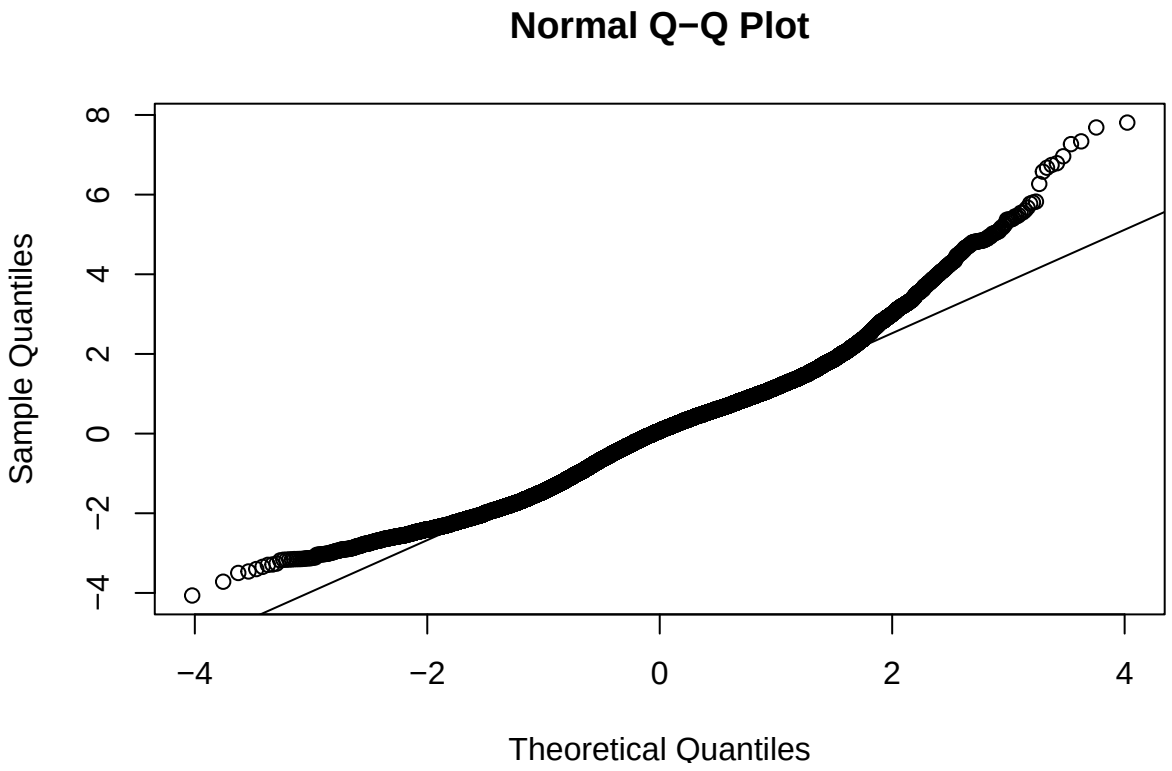
```r
summary(model)
```

```
##
## Call:
## lm(formula = sqrt(cnt) ~ weathersit + poly(temp, 2) + hum + poly(registered,
##     2), data = bike)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0642 -0.9525  0.0708  0.7995  7.8082
##
## Coefficients:
##                          Estimate Std. Error  t value Pr(>|t|)
## (Intercept)              13.13015    0.03907  336.050  < 2e-16 ***
## weathersit                0.09937    0.01756    5.658 1.56e-08 ***
## poly(temp, 2)1           66.67839    1.44997   45.986  < 2e-16 ***
## poly(temp, 2)2           -7.10015    1.39407   -5.093 3.56e-07 ***
## hum                      -2.00235    0.06320  -31.682  < 2e-16 ***
## poly(registered, 2)1  787.46744    1.48895  528.876  < 2e-16 ***
## poly(registered, 2)2 -244.80894    1.39145 -175.938  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.338 on 17372 degrees of freedom
## Multiple R-squared:  0.9603, Adjusted R-squared:  0.9603
## F-statistic: 7.003e+04 on 6 and 17372 DF,  p-value: < 2.2e-16
```

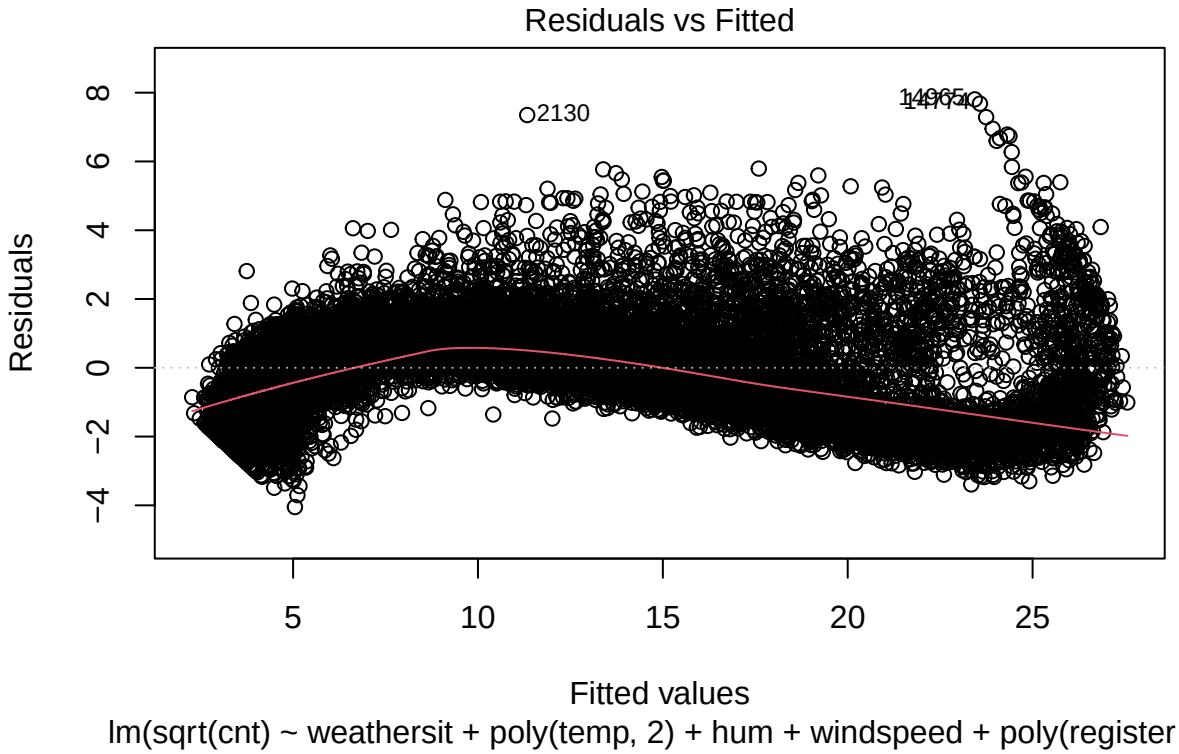
```r
model_residuals <- model$residuals
hist(model_residuals)
```

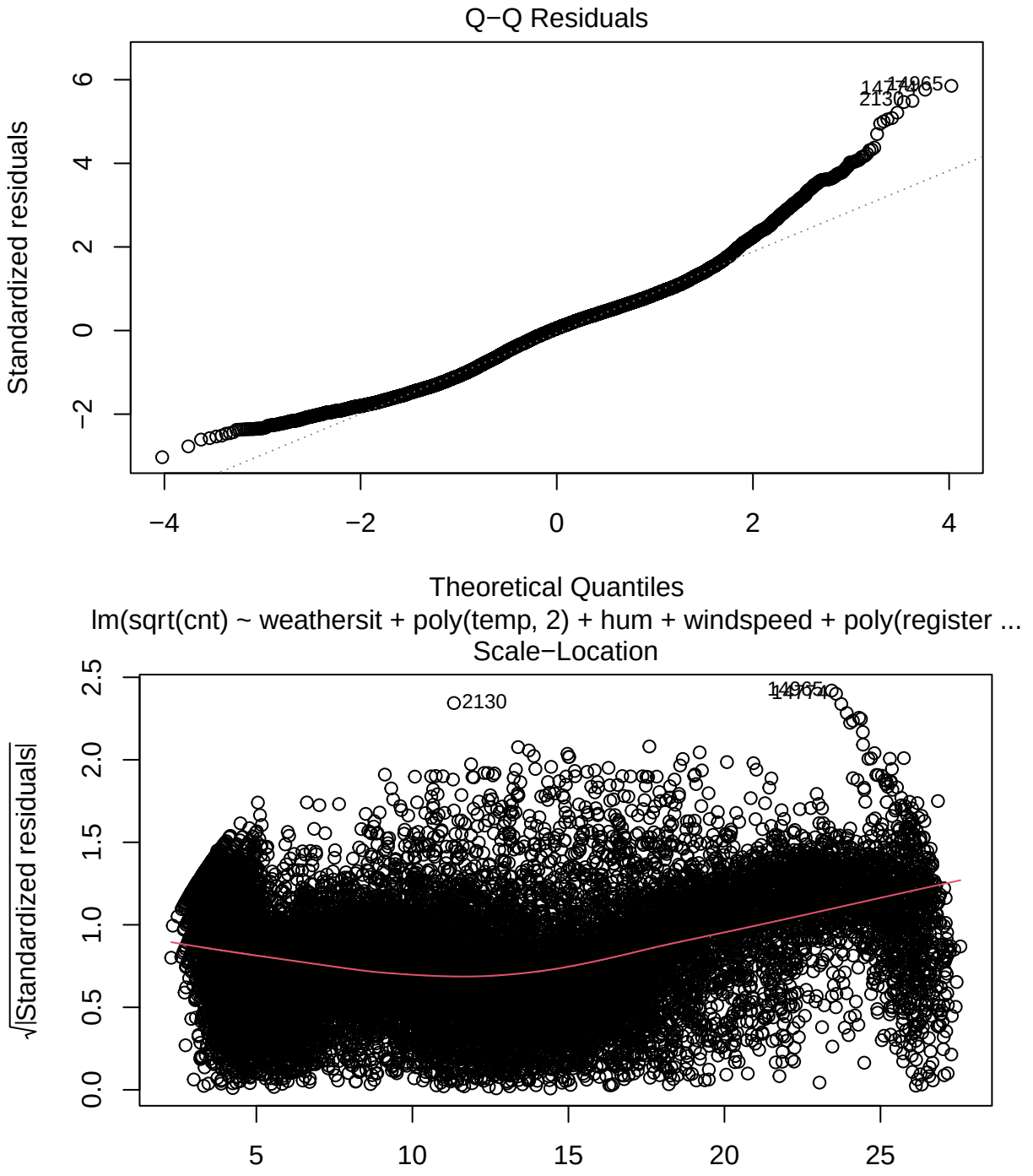


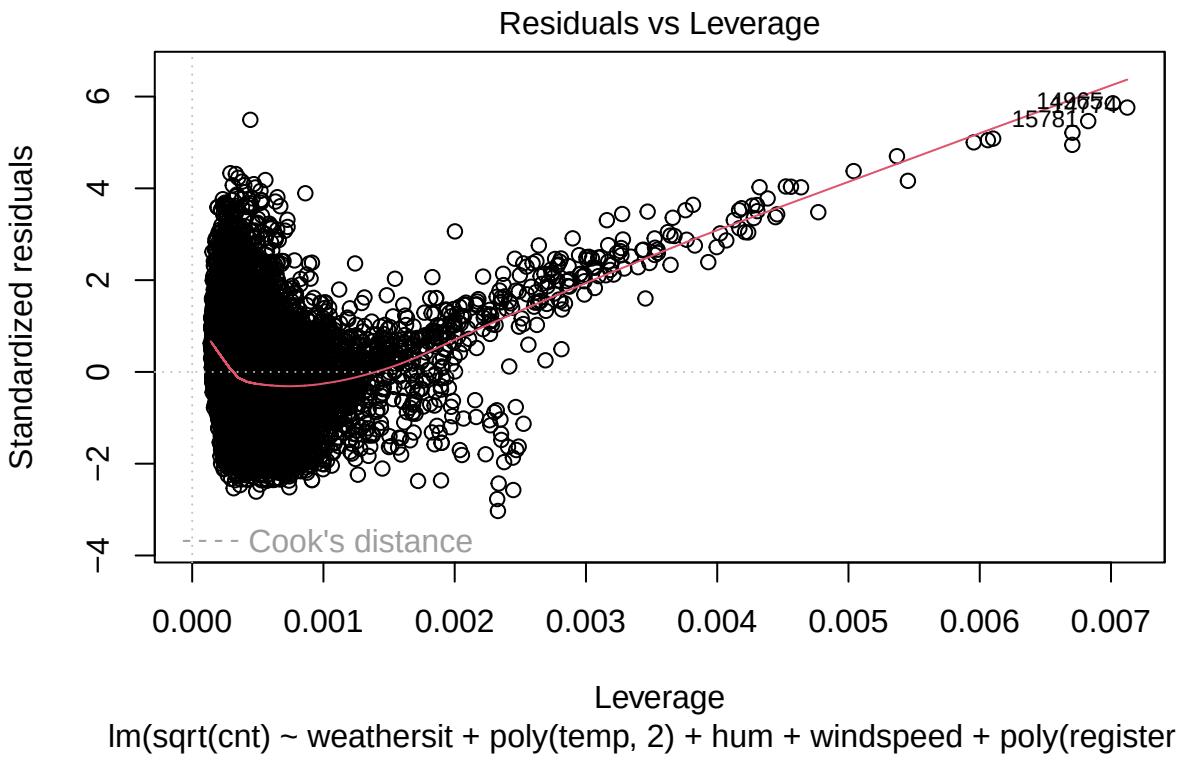
```
qqnorm(model_residuals)
qqline(model_residuals)
```




```
plot(lm(sqrt(cnt) ~ weathersit+poly(temp,2)+hum+windspeed+poly(registered,2), data=bike))
```

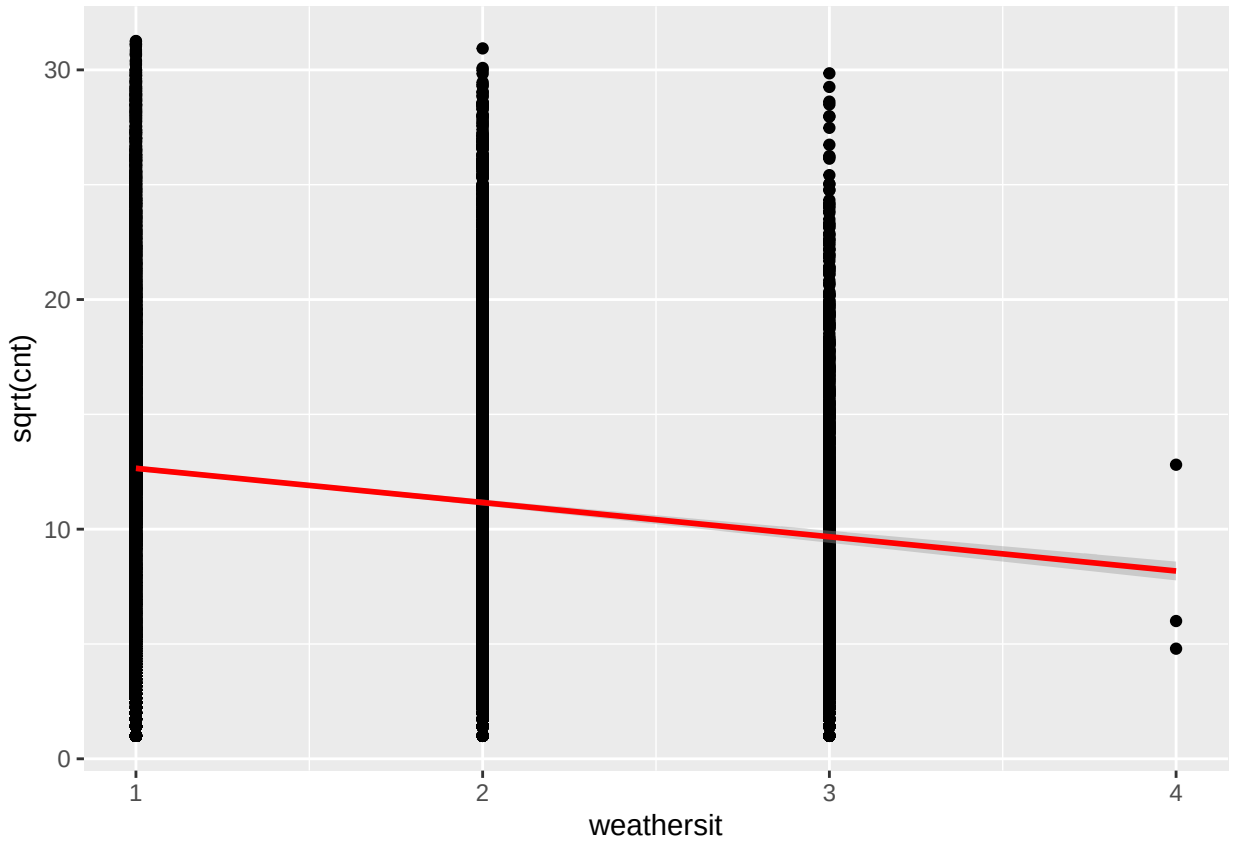



Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(sqrt(cnt) ~ weathersit + poly(temp, 2) + hum + windspeed + poly(register ...
Scale−Location
√|Standardized residuals|
2130
Fitted values
lm(sqrt(cnt) ~ weathersit + poly(temp, 2) + hum + windspeed + poly(register ...



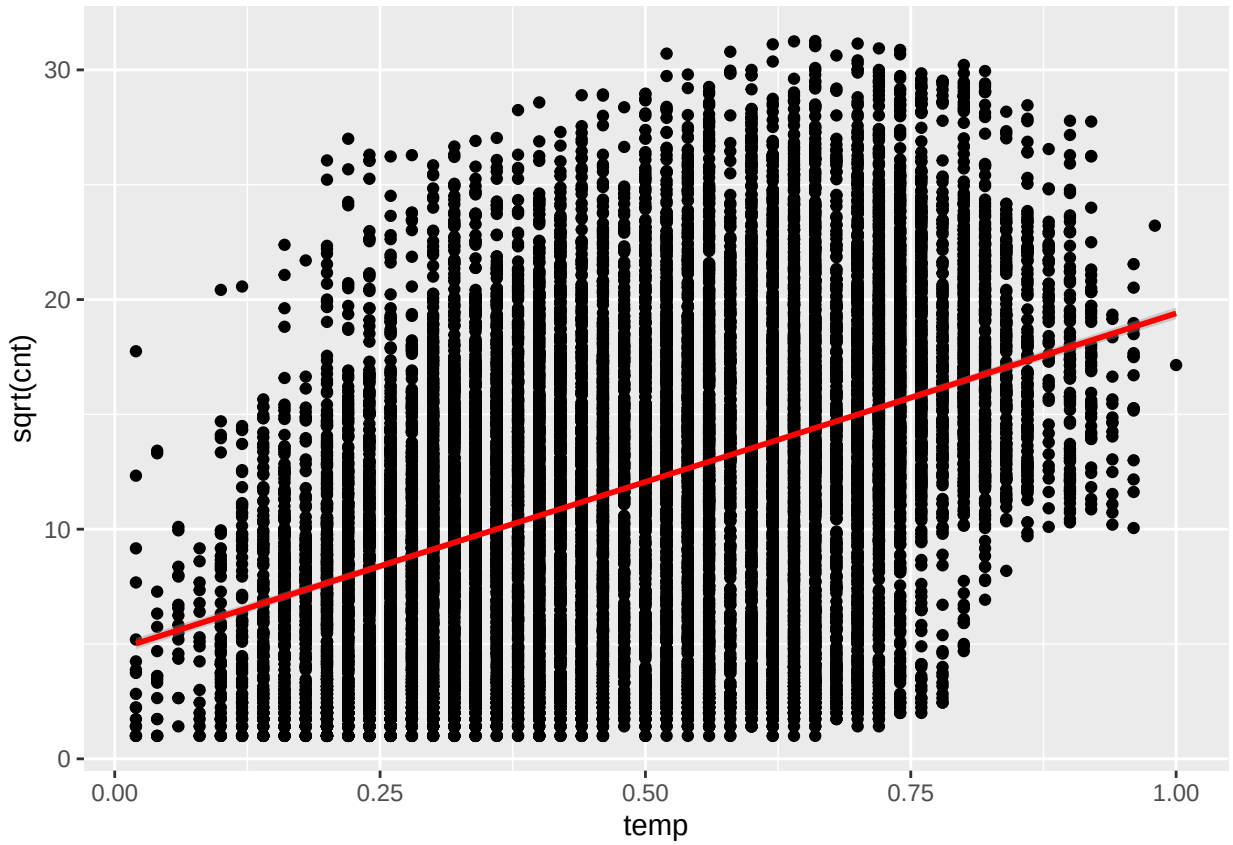

```
ggplot(model, aes(x = weathersit, y = sqrt(cnt)))+
  geom_point()+
  stat_smooth(method = "lm", col = "red")
```

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



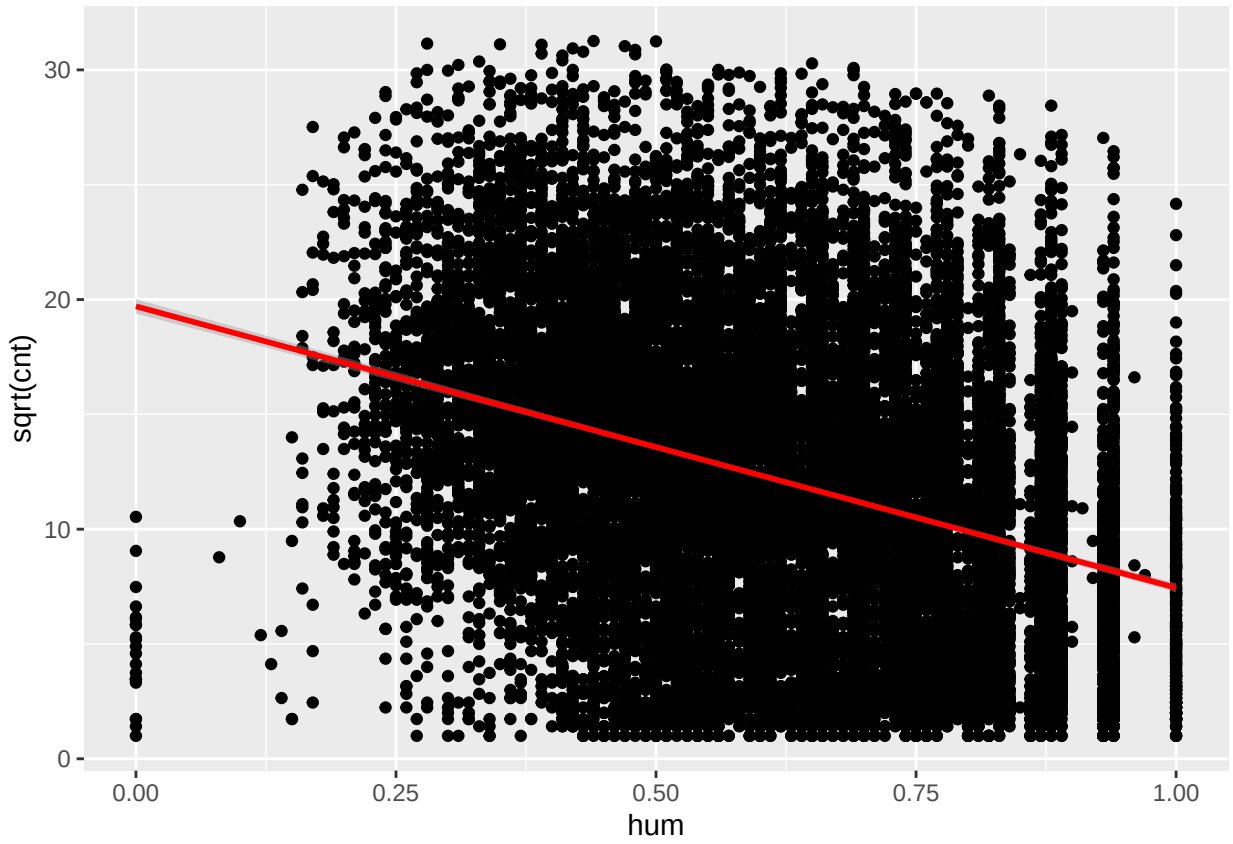

```
ggplot(model, aes(x = temp, y = sqrt(cnt)))+
  geom_point()+
  stat_smooth(method = "lm", col = "red")
```

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



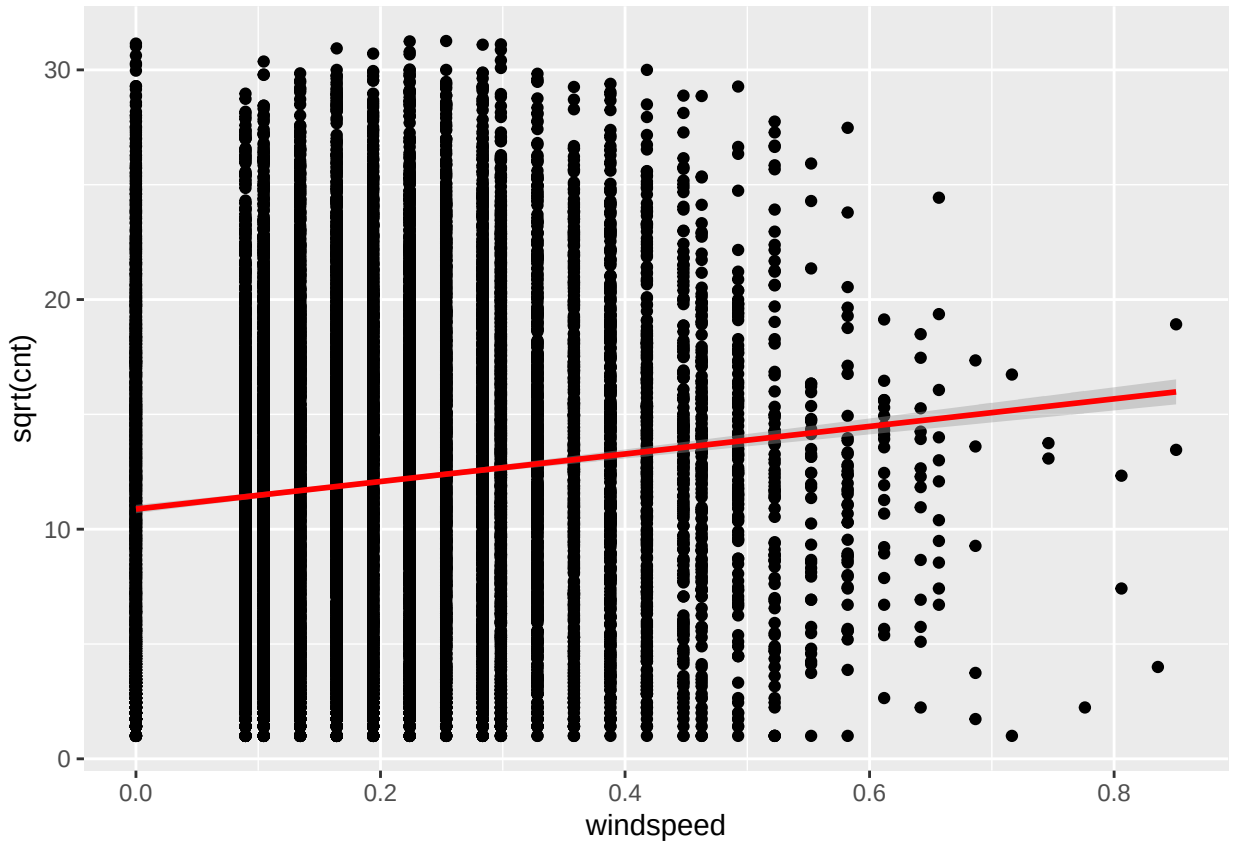

```
ggplot(model, aes(x = hum, y = sqrt(cnt)))+
  geom_point()+
  stat_smooth(method = "lm", col = "red")
```

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

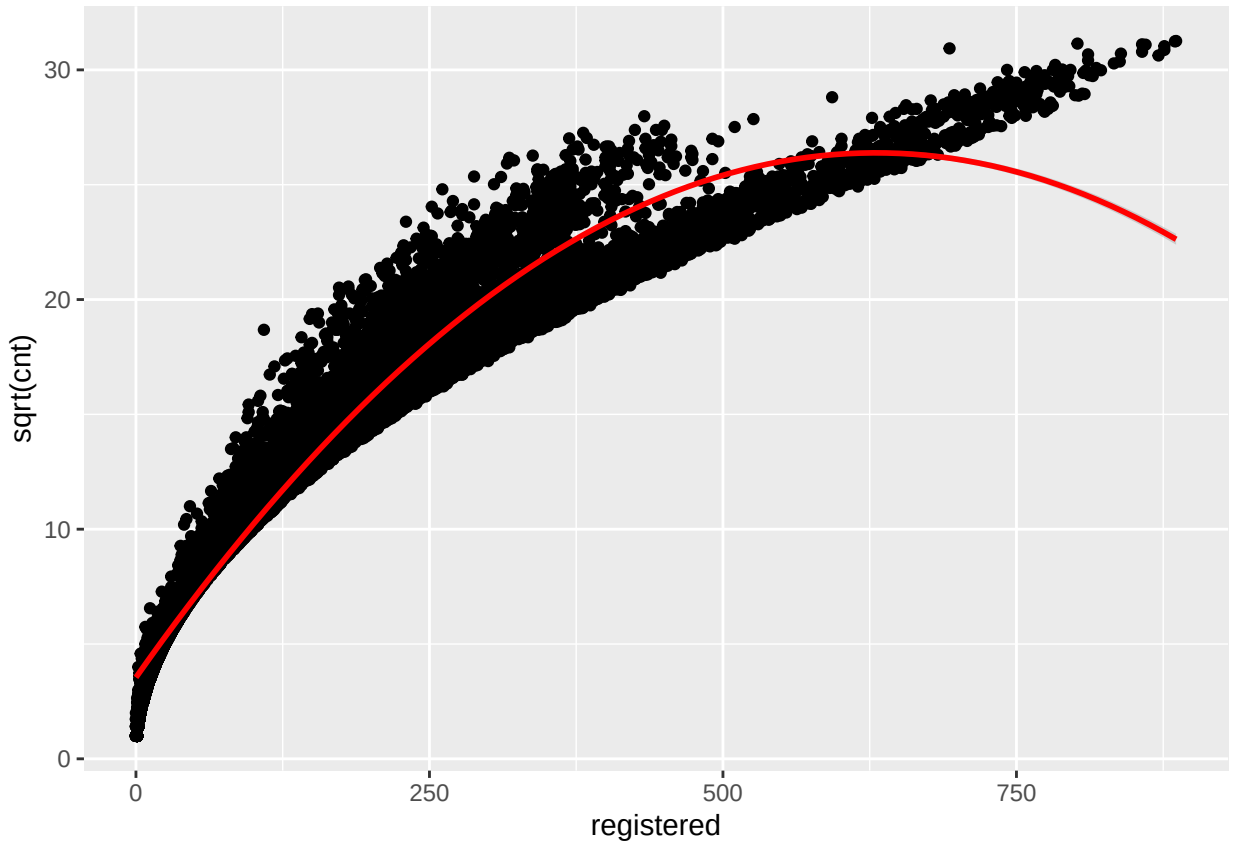



```
ggplot(model, aes(x = windspeed, y = sqrt(cnt)))+
  geom_point()+
  stat_smooth(method = "lm", col = "red")
```

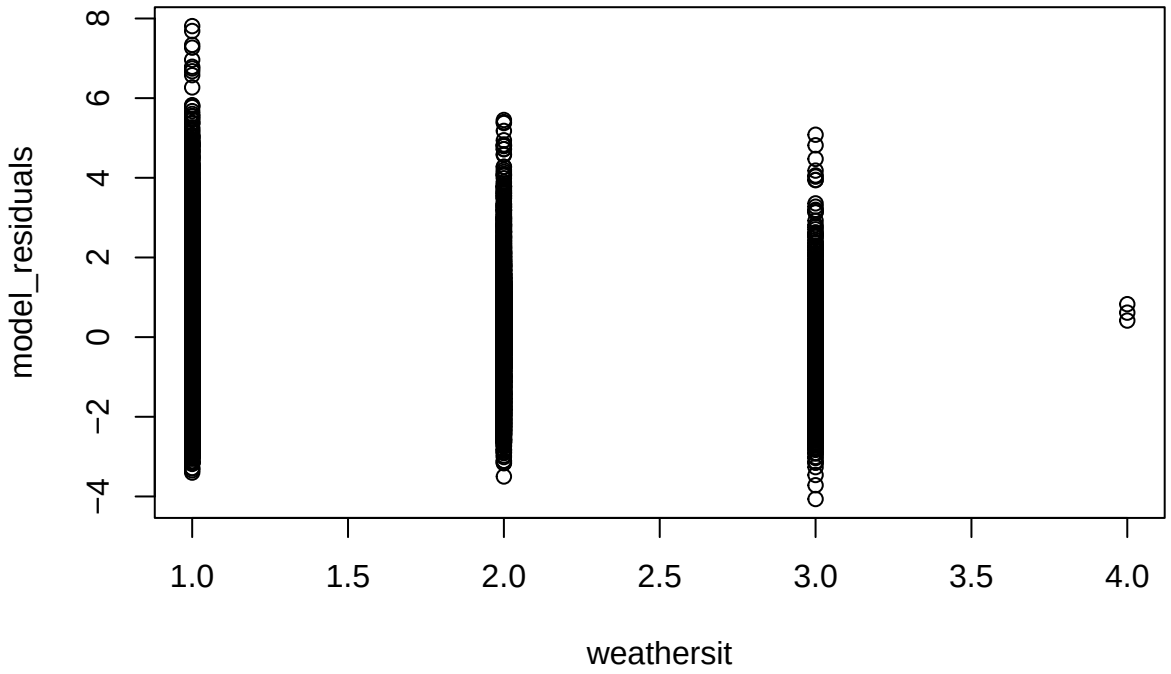
```
## `geom_smooth()` using formula = 'y ~ x'
```



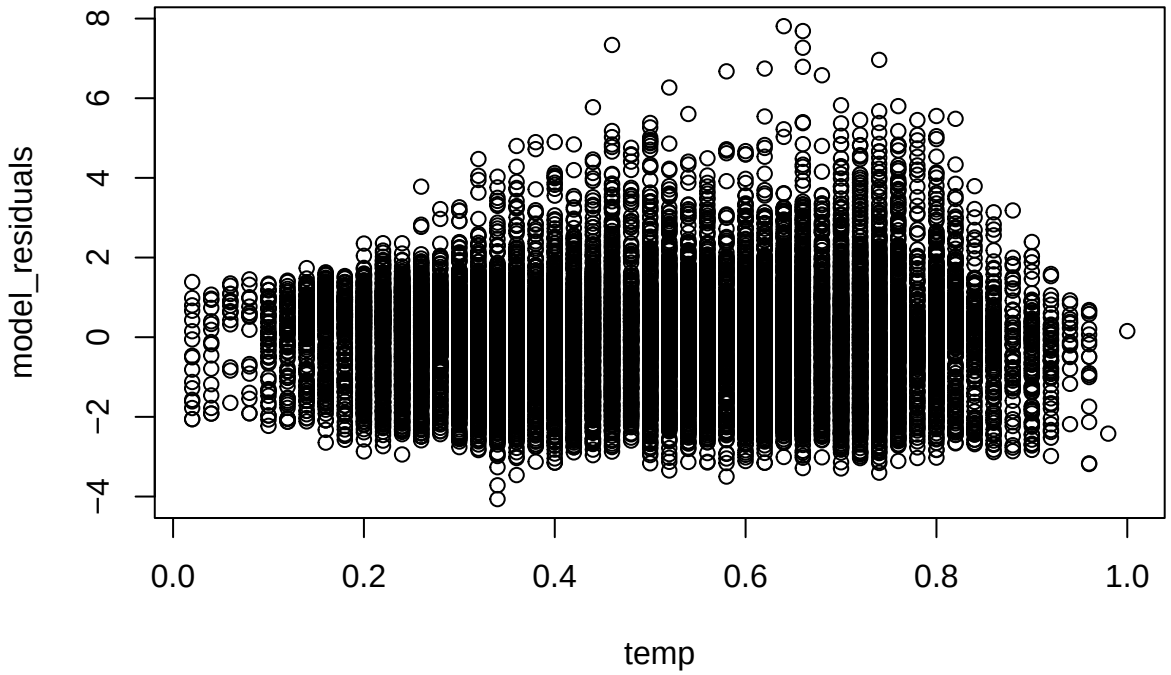

```
#ggplot(model, aes(x = casual, y = sqrt(cnt))) + geom_point()+ stat_smooth(method = "lm", col = "red")
ggplot(model, aes(x = registered, y = sqrt(cnt)))+
  geom_point()+
  stat_smooth(method = "lm", formula = y ~ x + I(x^2), col = "red")
```



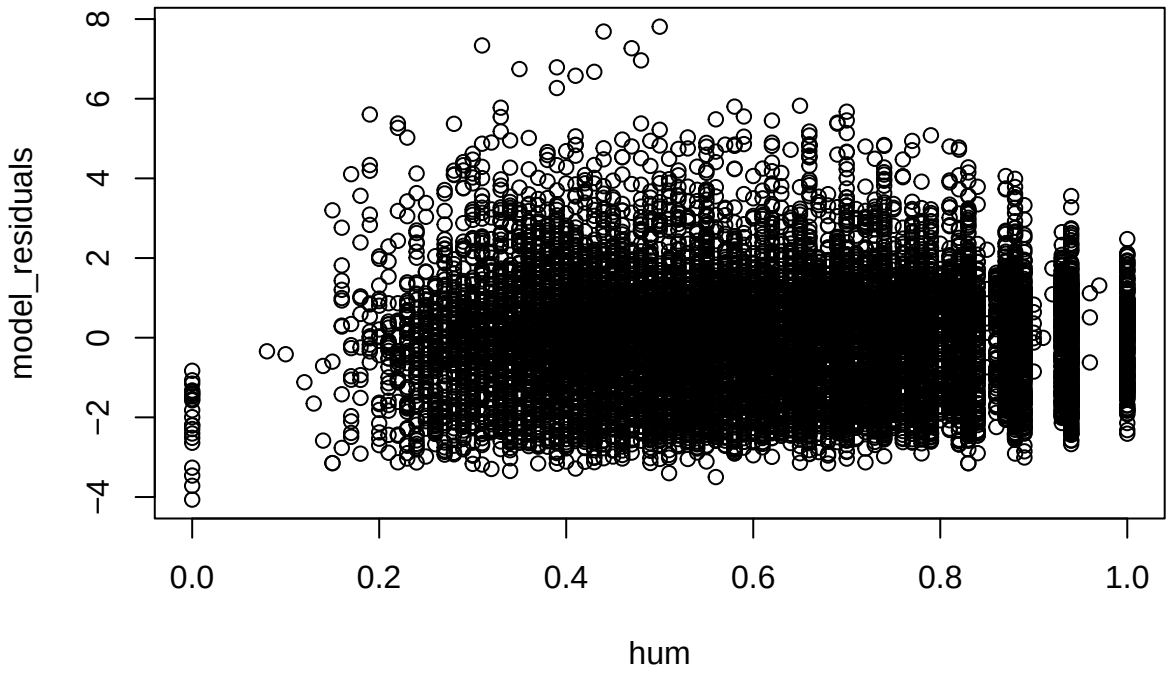

```
#since the plot between the predictor variable "registered" and the response variable is more like a qu

plot(x = weathersit, y = model_residuals)
```



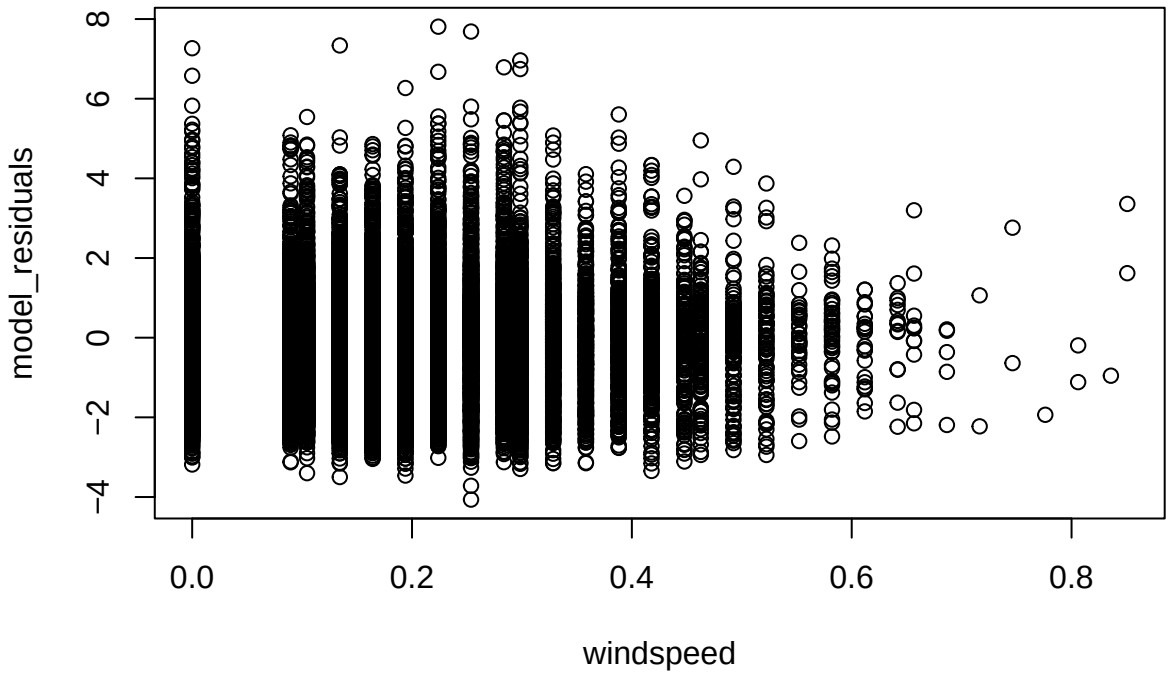
```
plot(x = temp, y = model_residuals)
```



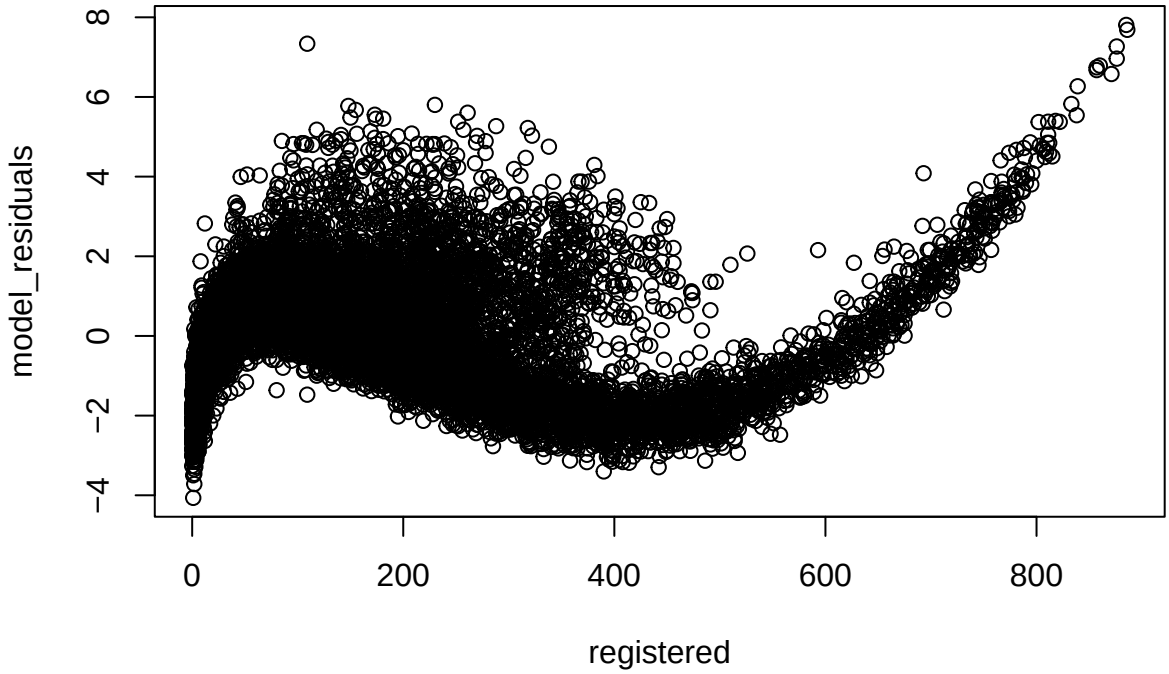

```
plot(x = hum, y = model_residuals)
```




```
plot(x = windspeed, y = model_residuals)
```




```
#plot(x = casual, y = model_residuals)
#We drop the predictor variable "casual" also because the relationship between it and our model's resid
plot(x = registered, y = model_residuals)
```




```
quadraticModel <- lm(sqrt(cnt) ~ registered + registered^2, data=bike)
summary(quadraticModel)
```

```
##
## Call:
## lm(formula = sqrt(cnt) ~ registered + registered^2, data = bike)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9649  -1.6206   0.4242   1.5836   8.5126
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.6741197  0.0267715   211.9   <2e-16 ***
## registered  0.0412371  0.0001241   332.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.476 on 17377 degrees of freedom
## Multiple R-squared:  0.8641, Adjusted R-squared:  0.8641
## F-statistic: 1.105e+05 on 1 and 17377 DF,  p-value: < 2.2e-16
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

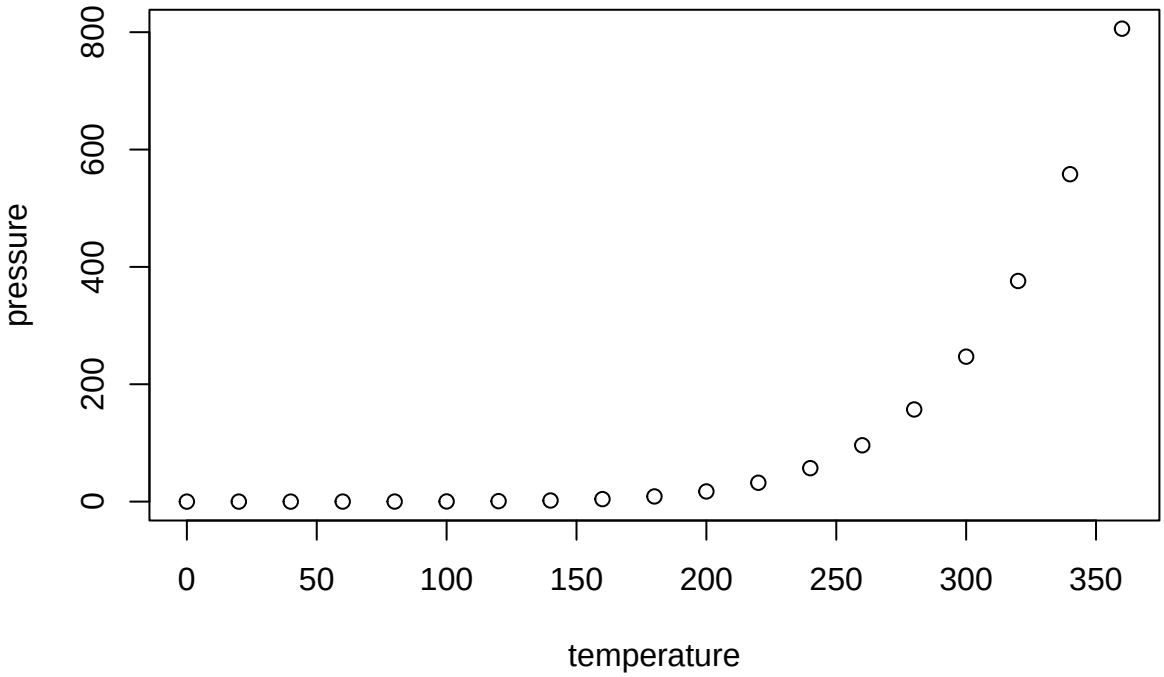
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.