STAC33 TUT-9

Uzair

3/20/2022

Introduction

This week's lecture we will be discussing topics from Ch:18. Chapter 18 is about **Simple Regression**. All the problems being discussed can be found on the PASIAS here

18.24 Running and blood sugar

Diabetic patient measures is **blood sugar** level after a run for a particular **distance**.

a. Load the data and make a sutiable plot

```
## loading the data
my_url <- "http://ritsokiguess.site/datafiles/runner.txt"
runs <- read_delim(my_url, " ")
```

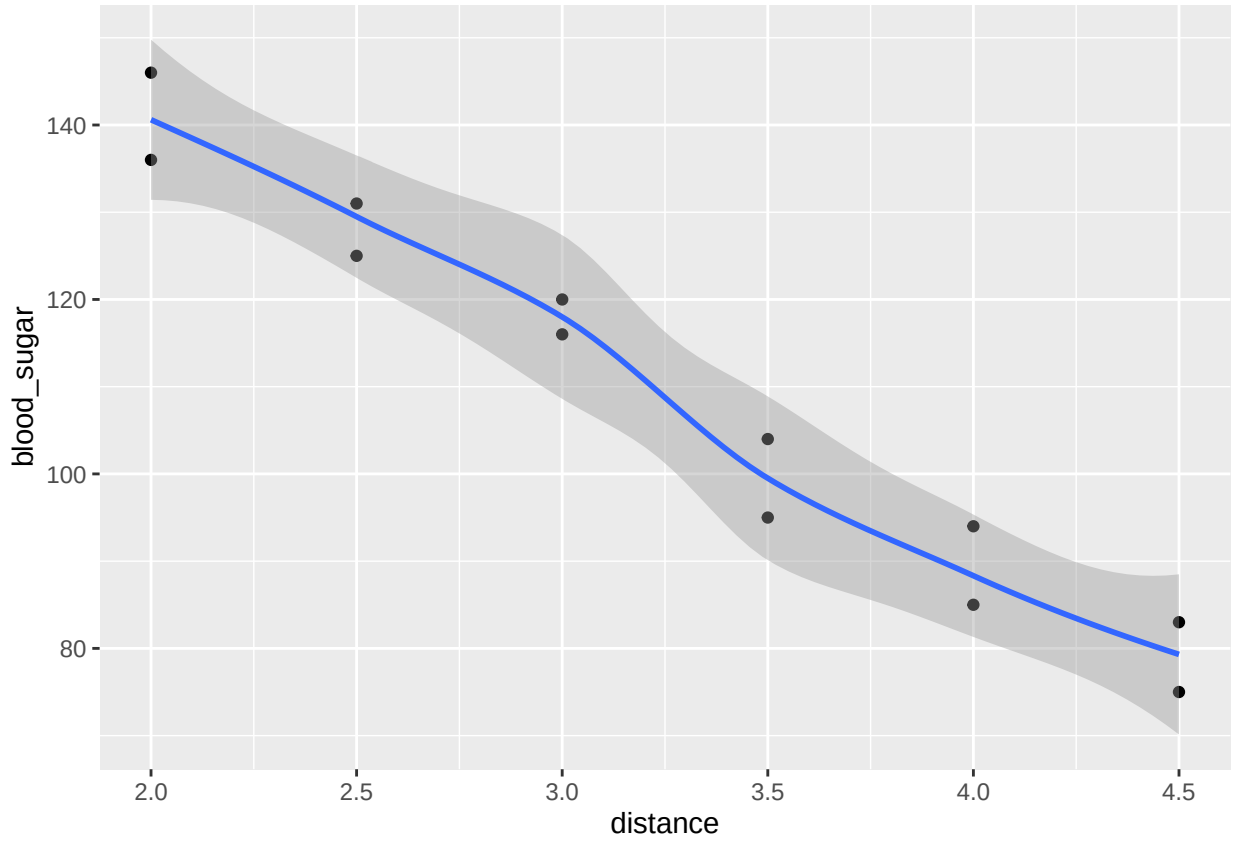
```
## Rows: 12 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: " "
## dbl (2): distance, blood_sugar
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
runs
```

```
## # A tibble: 12 x 2
##    distance blood_sugar
##       <dbl>       <dbl>
##  1      2           136
##  2      2           146
##  3      2.5         131
##  4      2.5         125
##  5      3           120
##  6      3           116
##  7      3.5         104
##  8      3.5          95
##  9      4            85
## 10      4            94
## 11      4.5          83
## 12      4.5          75
```

```
## making the scatter plot
ggplot(runs, aes(x = distance, y = blood_sugar)) + geom_point() +
  geom_smooth()
```

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



b. Based on the above plot can we say that the relationship is linear or not?

There seems to be a linear relationship, as there seems to be a lack of curvature/variability in the overall trend.

c. Fit a regression model and obtain the model details

`lm` -> fit the model
`summary` -> gets the model diagnostics and details

```
# fitting model for blood sugar level against the distance run
runs.1 <- lm(blood_sugar ~ distance, data = runs)
# getting the summary of the model
summary(runs.1)
```

```
##
## Call:
## lm(formula = blood_sugar ~ distance, data = runs)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -7.8238 -3.6167  0.8333  4.0190  5.5476
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  191.624      5.439   35.23 8.05e-12 ***
## distance     -25.371      1.618  -15.68 2.29e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.788 on 10 degrees of freedom
## Multiple R-squared:  0.9609, Adjusted R-squared:  0.957
## F-statistic: 245.7 on 1 and 10 DF,  p-value: 2.287e-08
```

d. What does the slope tell you about this model wrt. our data?

The slope is -25.37. This means that for each additional mile run, the runner's blood sugar will decrease on average by about 25 units.

e. Is there a (statistically) significant relationship between running distance and blood sugar? How do you know? Do you find this surprising, given what you have seen so far? Explain briefly.

Recall the t-test being conducted here: Null = $\beta_{i0} = 0$ that is the relationship is not significant.

Look at the P-value either on the distance line (for its t-test) or for the F-statistic on the bottom line. These are the same: 0.000000023. (They will be the same any time there is one x-variable.) This P-value is way smaller than 0.05, so there is a significant relationship between running distance and blood sugar.

f. This diabetic is planning to go for a 3-mile run tomorrow and a 5-mile run the day after. Obtain suitable 95% intervals that say what his blood sugar might be after each of these runs.

`predict` -> used to get the prediction interval

```
# declaring the two values of interest in a tibble
dists <- c(3, 5)
dist.new <- tibble(distance = dists)
dist.new
```

```
## # A tibble: 2 x 1
##   distance
##      <dbl>
## 1        3
## 2        5
```

```
# getting their prediction intervals
pp <- predict(runs.1, dist.new, interval = "p")
pp
```

```
##         fit       lwr       upr
## 1 115.50952 104.37000 126.64905
## 2  64.76667  51.99545  77.53788
```

```r
# combining the results
cbind(dist.new, pp)
```

```
##   distance       fit       lwr       upr
## 1        3 115.50952 104.37000 126.64905
## 2        5  64.76667  51.99545  77.53788
```

Blood sugar after a 3-mile run is predicted to be between 104 and 127; after a 5-mile run it is predicted to be between 52 and 77.5.

g. Which one of the prediction interval is longer? Explain

Prediction interval closer to the observed values will have a narrower range than prediction intervals far from the observed values.

```r
cbind(dist.new, pp) %>% mutate(int.length = -lwr + upr)
```

```
##   distance       fit       lwr       upr int.length
## 1        3 115.50952 104.37000 126.64905   22.27905
## 2        5  64.76667  51.99545  77.53788   25.54243
```

The intervals are about 22.25 and 25.5 units long. The one for a 5-mile run is a bit longer. I think this makes sense because 3 miles is close to the average run distance, so there is a lot of "nearby" data. 5 miles is actually longer than any of the runs that were actually done (and therefore we are actually extrapolating), but the important point for the prediction interval is that there is less nearby data: those 2-mile runs don't help so much in predicting blood sugar after a 5-mile run.

18.28 Predicting height from foot length

Goal is to predict the height from the footlength of the people.
We have 33 male entries in this dataset.

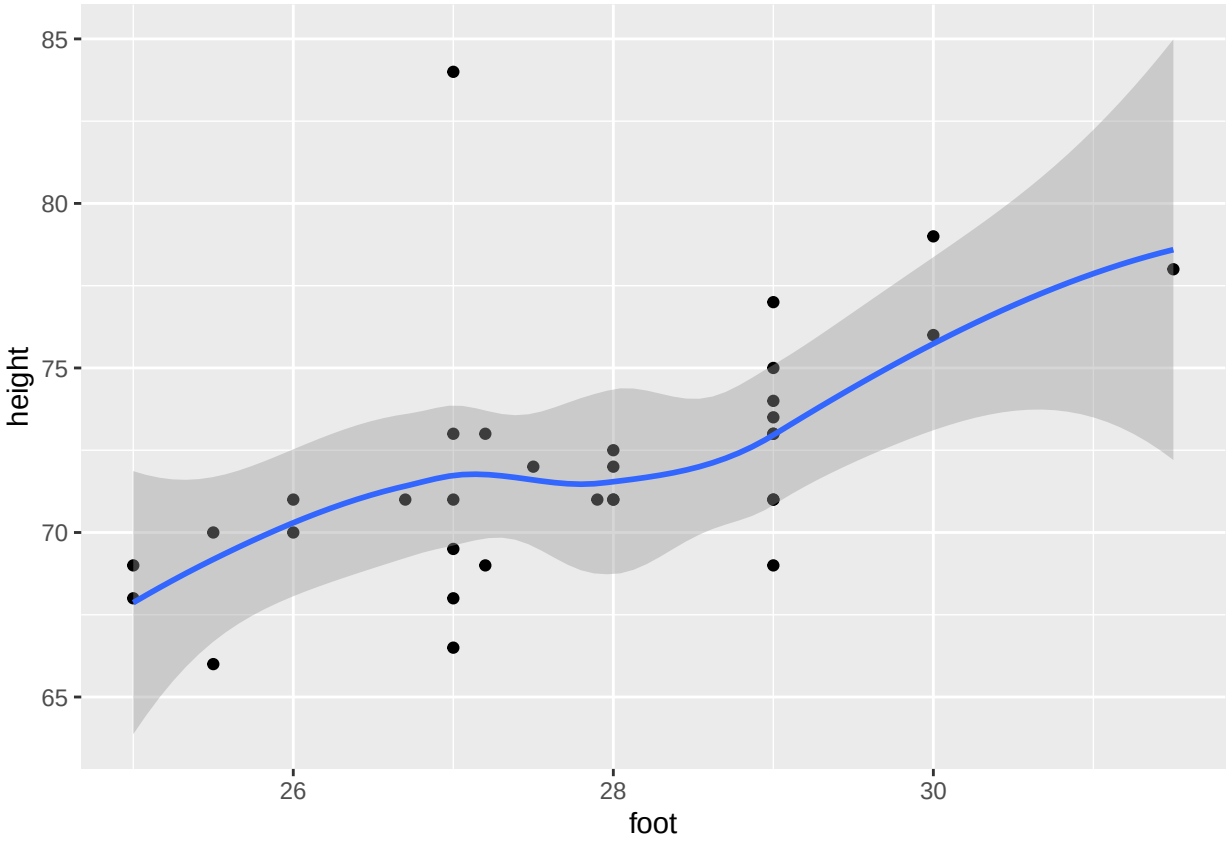
a. Read data and make sutiable plots

```r
# read in the data
my_url <- "http://ritsokiguess.site/datafiles/heightfoot.csv"
hf <- read_csv(my_url)
```

```
## Rows: 33 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (2): height, foot
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# make a scatterplot with trend line along with CI-bands
ggplot(hf, aes(y=height, x=foot)) + geom_point() + geom_smooth()
```

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



b. Are there any observations not on the trend of the other points? What is unusual about those observations? The observation with height greater than 80 at the top of the graph looks like an outlier and does not follow the trend of the rest of the points. Or, this individual is much taller than you would expect for someone with a foot length of 27 inches. Or, this person is over 7 feet tall, which makes little sense as a height. Say something about what makes this person be off the trend.

c. Fit a regression predicting height from foot length, including any observations that you identified in the previous part. For that regression, plot the residuals against the fitted values and make a normal quantile plot of the residuals.

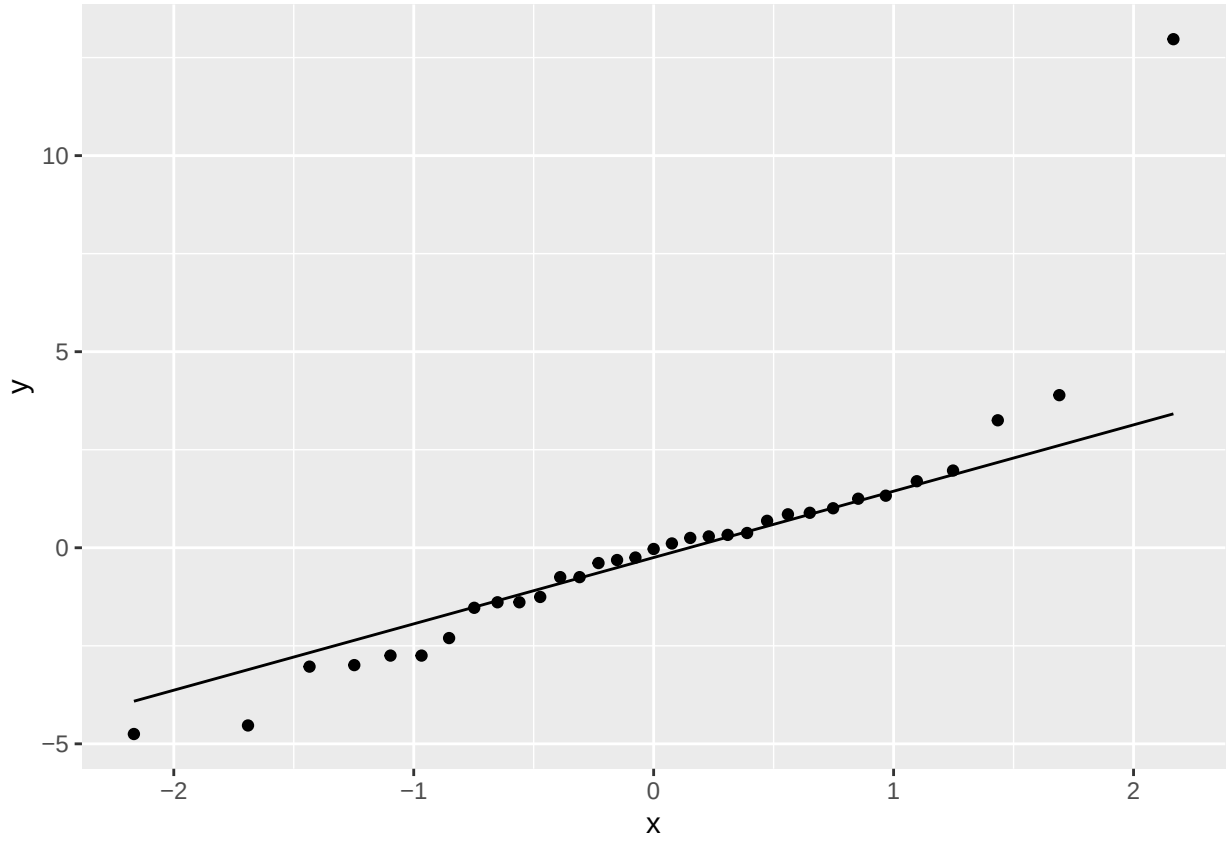
```
hf.1 <- lm(height~foot, data=hf)
summary(hf.1)
```

```
##
## Call:
## lm(formula = height ~ foot, data = hf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7491 -1.3901 -0.0310  0.8918 12.9690
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  34.3363     9.9541   3.449 0.001640 **
## foot          1.3591     0.3581   3.795 0.000643 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.102 on 31 degrees of freedom
## Multiple R-squared:  0.3173, Adjusted R-squared:  0.2952
## F-statistic: 14.41 on 1 and 31 DF,  p-value: 0.0006428
```

Cool we know about the relationship and significance tests.

```
# to check the assumption about normality of the residuals
ggplot(hf.1, aes(sample=.resid)) + stat_qq() + stat_qq_line()
```



That one point is still waay off.

d. Remove that problem point and carry out similar steps as you did previously

We notice the max height is 80 so remove the point that has height greater than 80.
we use `filter()` -> to filter based on a logical statement.

```
hf %>% filter(height<80) -> hfx
hfx
```

```
## # A tibble: 32 x 2
##    height  foot
##     <dbl> <dbl>
##  1   66.5  27
```

```
##  2   73.5  29
##  3   70    25.5
##  4   71    27.9
##  5   73    27
##  6   71    26
##  7   71    29
##  8   69.5  27
##  9   73    29
## 10   71    27
## # ... with 22 more rows
```

Running the model on the new dataset.

```
hf.2 <- lm(height~foot, data=hfx)
summary(hf.2)
```

```
##
## Call:
## lm(formula = height ~ foot, data = hfx)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5097 -1.0158  0.4757  1.1141  3.9951
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.1502     6.5411   4.609 7.00e-05 ***
## foot          1.4952     0.2351   6.360 5.12e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.029 on 30 degrees of freedom
## Multiple R-squared:  0.5741, Adjusted R-squared:  0.5599
## F-statistic: 40.45 on 1 and 30 DF,  p-value: 5.124e-07
```

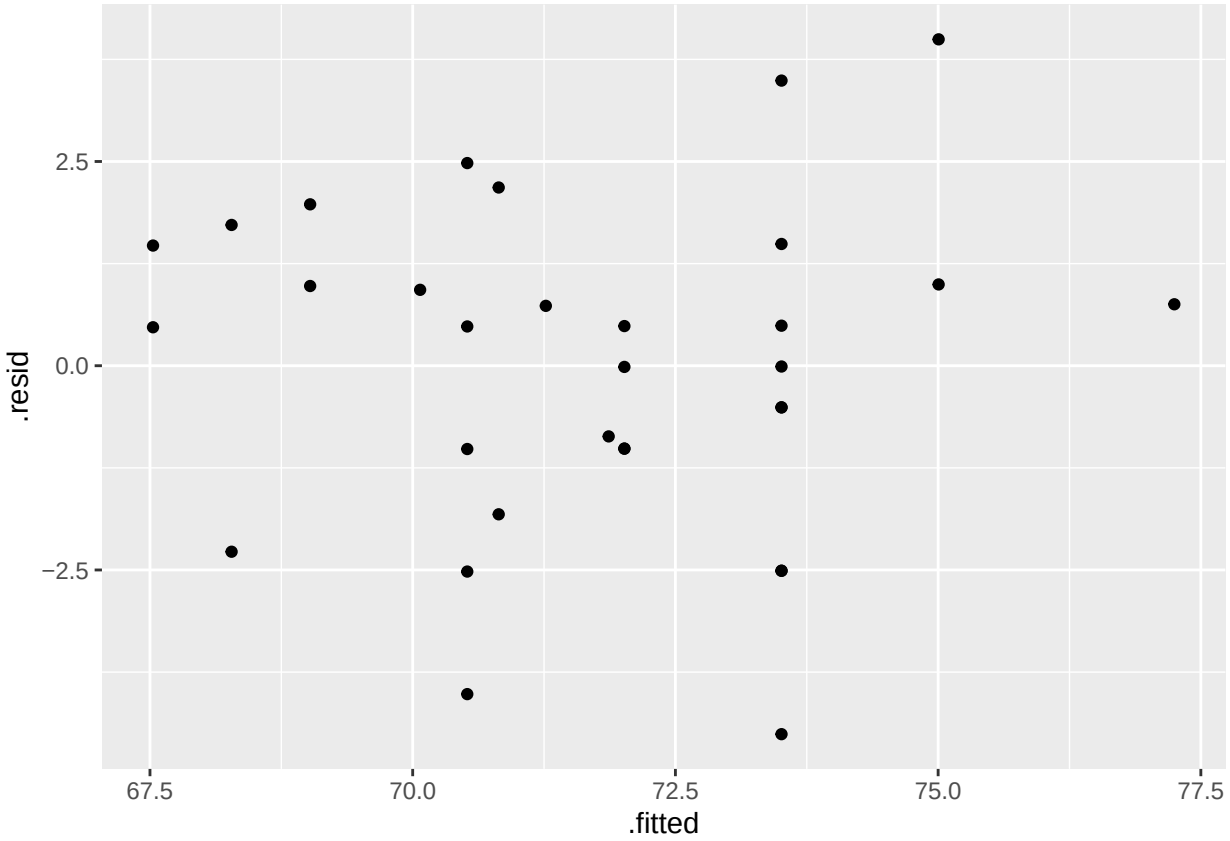
Improved R^2

Model Diagnostics time

Checking normality for our residuals

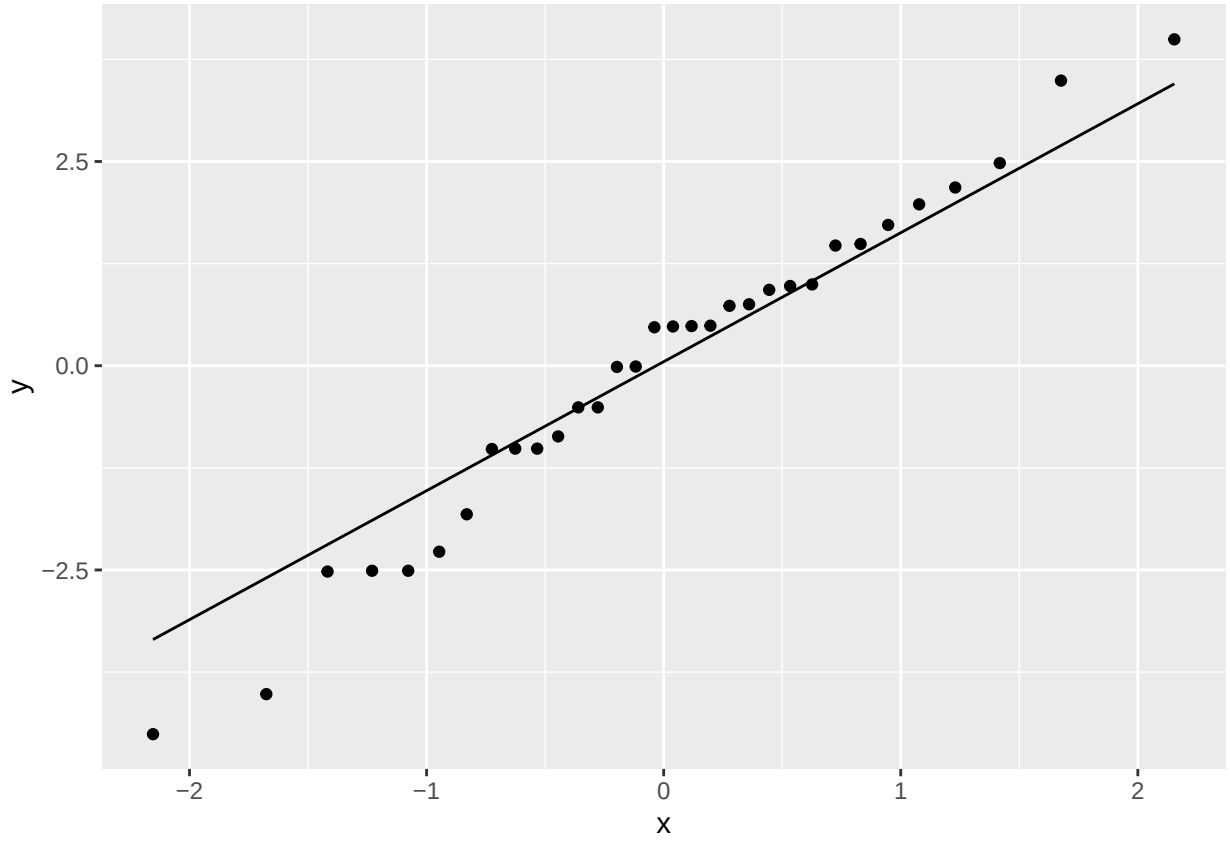
Fitted x Residual plot

```
ggplot(hf.2, aes(x=.fitted, y=.resid)) + geom_point()
```



Making QQ-plot

```
ggplot(hf.2, aes(sample=.resid)) + stat_qq() + stat_qq_line()
```



Looks fine as points are close to the line.
Hence we can be satisfied with the results we have.

Questions you can expect

Interpretation of model coefficients
Statistical Significance of the variables
Which fit was better based on R^2 this ties to model selection(which variable to choose)
Assumptions about regression model are satisfied or not?
Normality of the residuals, outliers effect on the slope and intercept