431 Class 08

github.com/THOMASELOVE/2019-431

2019-09-19

Today's Agenda, Part 1 (Notes, Chapters 9, 10)

(Continuing what was posted originally as Slides Set 07)

Are these data well described by a Normal model?

1. Calibrating our understanding of visualizations
2. Numerical Approaches
3. What can we do about non-Normal data?

- Summarize it with median and IQR, not mean and SD
- Transform the data (perhaps a power transformation)?

Today's Packages for Part 1

The R packages we're using today are `NHANES`, `magrittr`, `janitor` and `tidyverse`.

```
library(NHANES); library(magrittr)
library(janitor); library(tidyverse)
```

CWRU Colors

```
cwru.blue <- '#0a304e'
cwru.gray <- '#626262'
```

Today's Agenda, Part 2 (See Notes, Chapter 11)

1. A New Data Set (!)
2. Studying Scatterplots
3. Building Linear Models
    - Making predictions with PIs and CIs
    - Fundamental Summaries of a Regression Model
    - Understanding Regression Residuals
4. Measuring Association with Correlations
    - Pearson and Spearman approaches
    - Thinking about the impact of transformations
5. Adding a categorical predictor (factor) to a model
    - Using `fct_recode` from `forcats` (tidyverse)
    - Interpreting an indicator variable regression

Part 1 (Does a Normal model fit my data?)

Our nh2 data set (for Part 1)

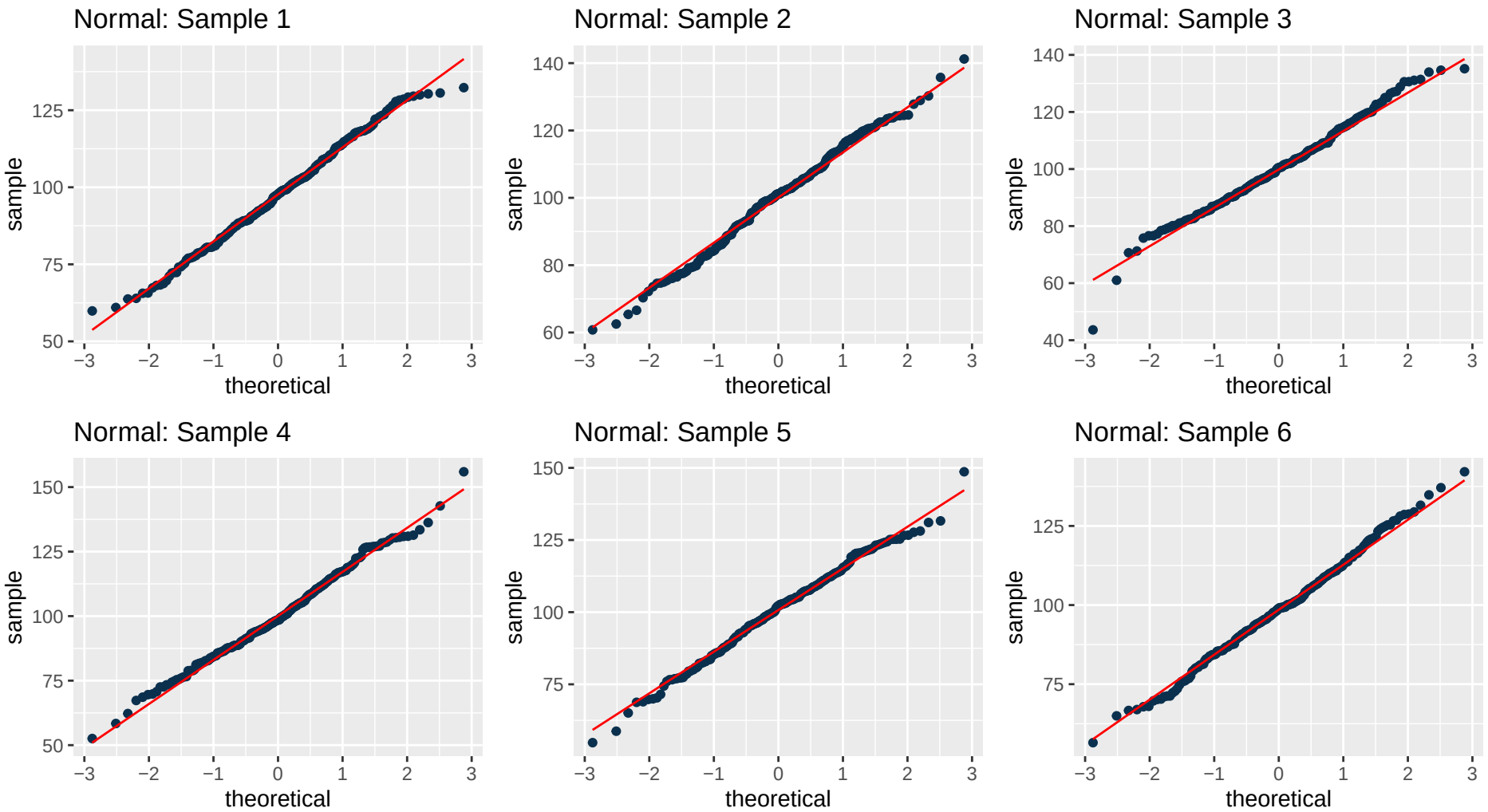
```
set.seed(20190910) # so we can get the same sample again

nh2 <- NHANES %>%
    filter(SurveyYr == "2011_12") %>%
    select(ID, SurveyYr, Age, Height, Weight, BMI, Pulse,
           SleepHrsNight, BPSysAve, BPDiaAve, Gender,
           PhysActive, SleepTrouble, Smoke100,
           Race1, HealthGen, Depressed) %>%
    rename(SleepHours = SleepHrsNight, Sex = Gender,
           SBP = BPSysAve, DBP = BPDiaAve) %>%
    filter(Age > 20 & Age < 80) %>% ## ages 21-79 only
    drop_na() %>% # removes all rows with NA
    sample_n(., size = 1000) %>% # sample 1000 rows
    clean_names() # from the janitor package (snake case)
```

Obtaining our Subset of Interest

```
nh2_GVGmales <- nh2 %>%
  filter(sex == "male" &
           health_gen %in% c("Good", "Vgood"))
```

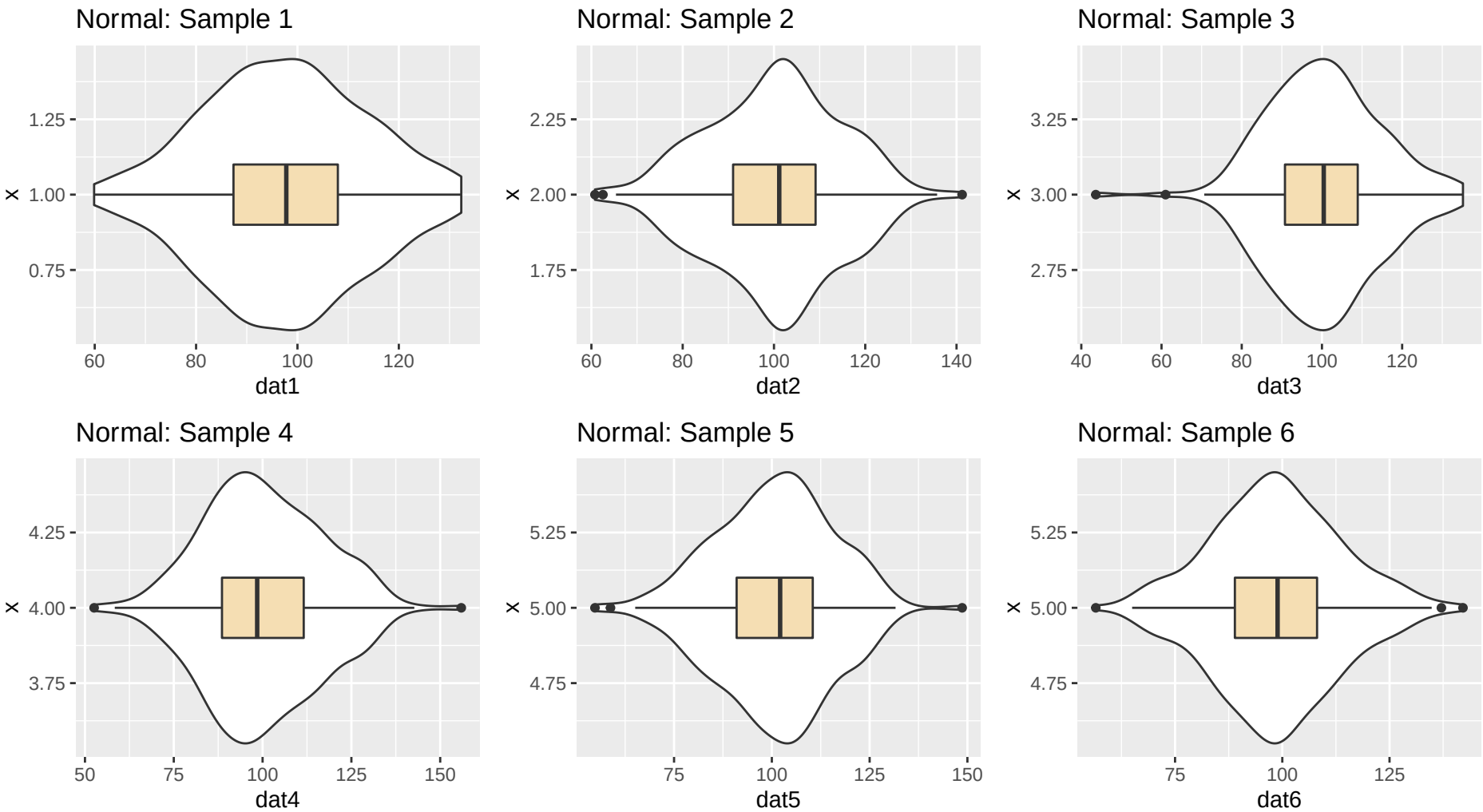
6 Normal Q-Q plots: Simulated Normal Data

Six simulations from a Normal distribution.



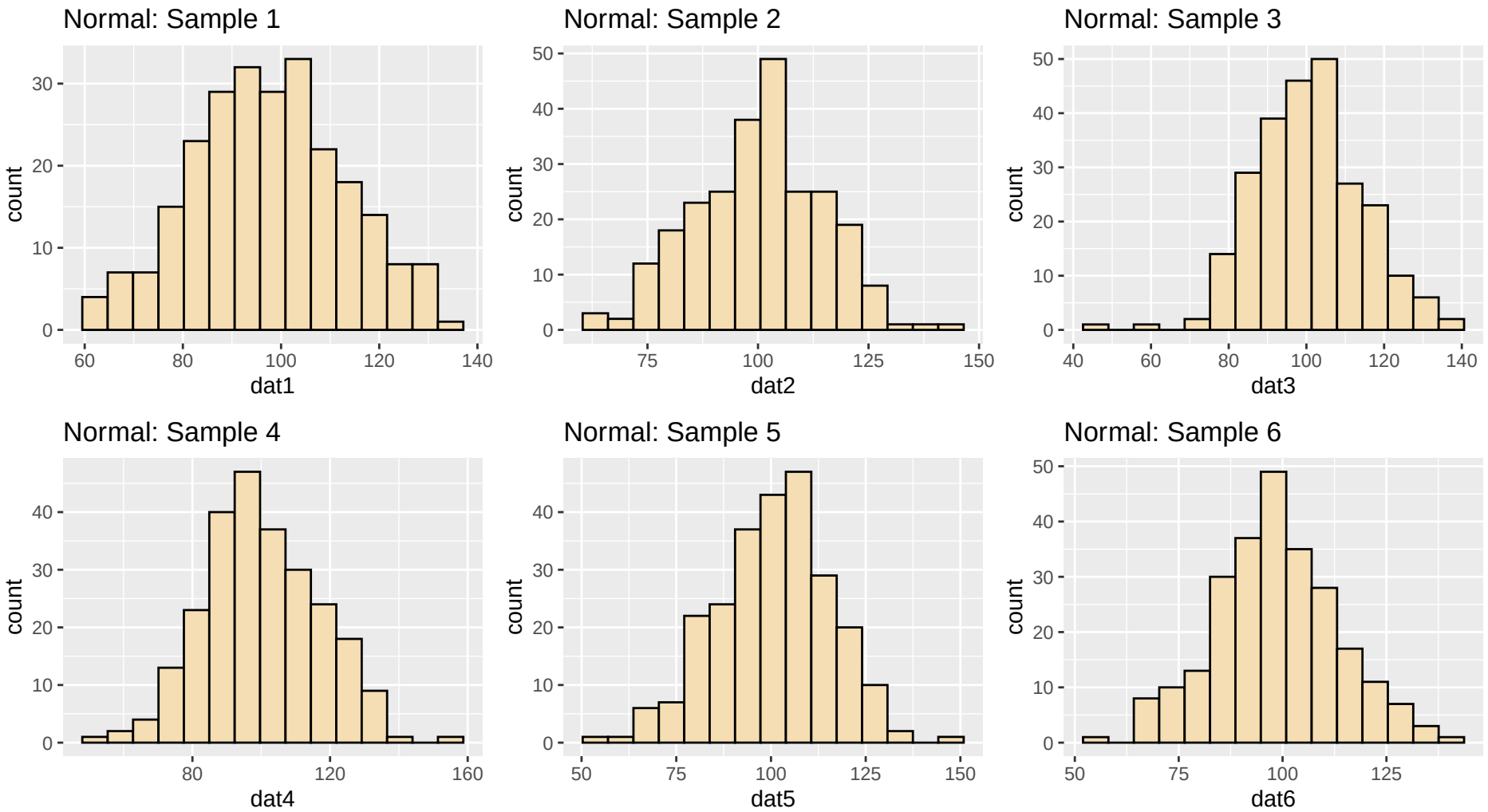
Same Six Simulations, in Box + Violin Plots

Six simulations from a Normal distribution.



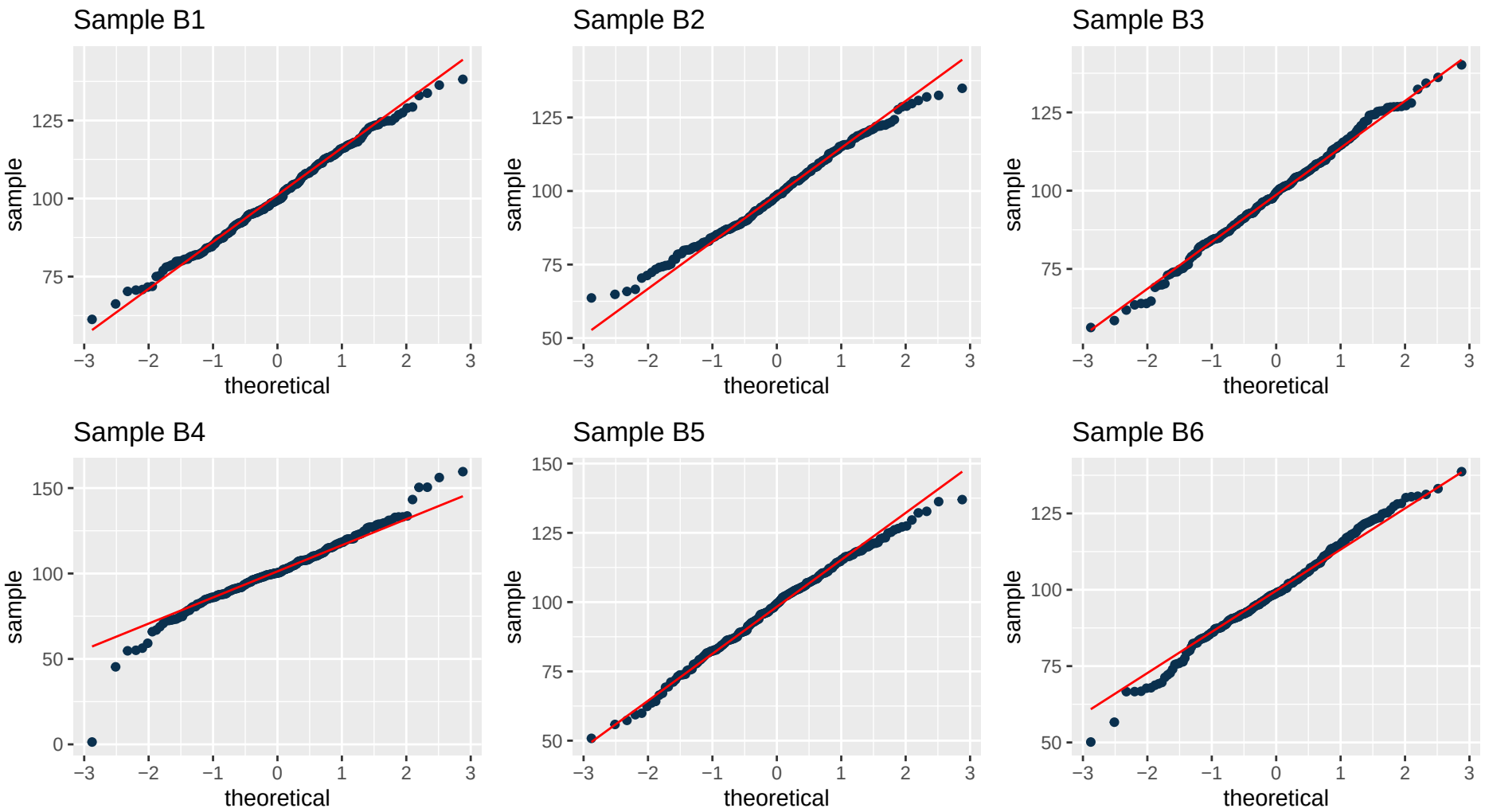
Same Six Simulations, in Histograms

Six simulations from a Normal distribution.

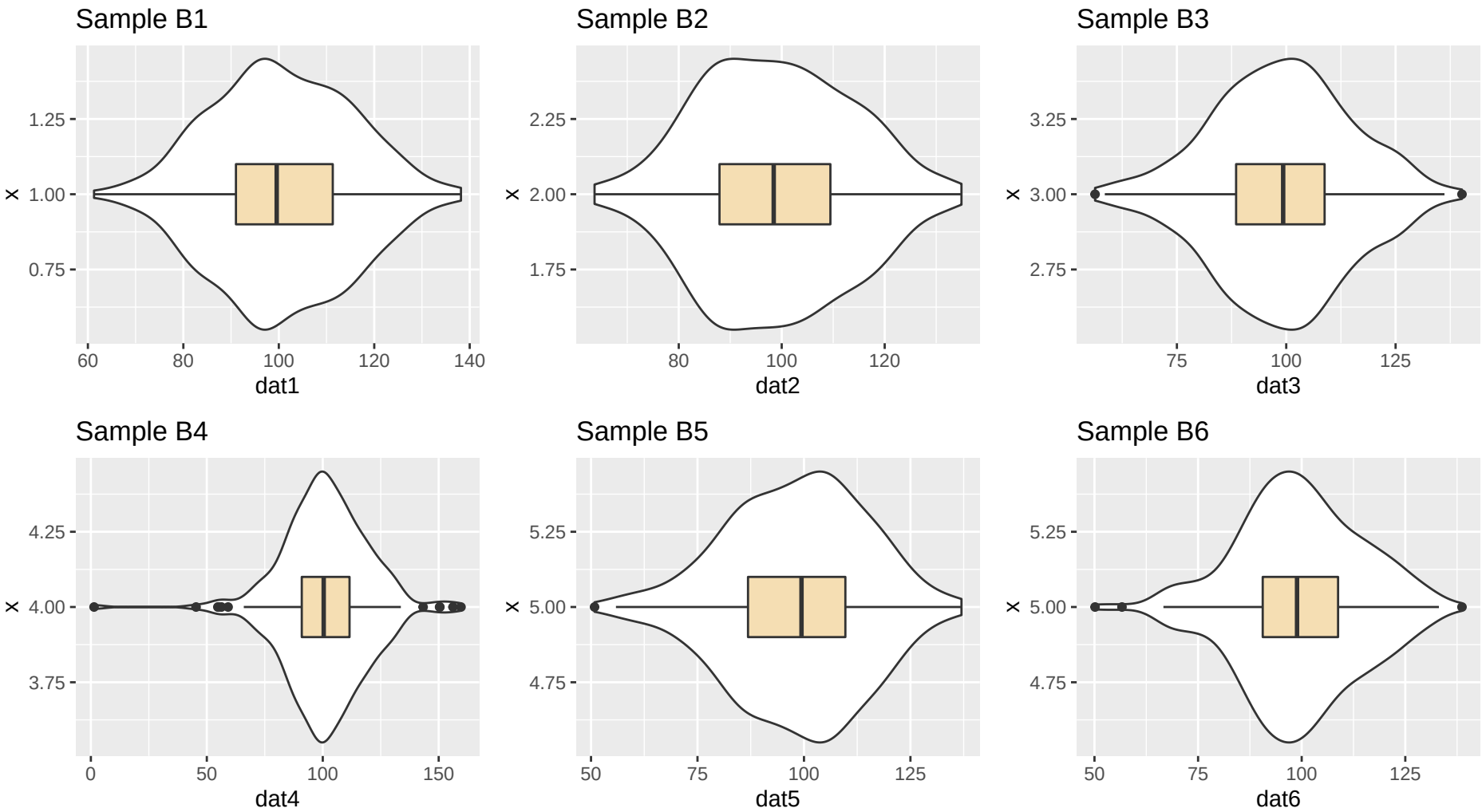


One of these things is not like the others

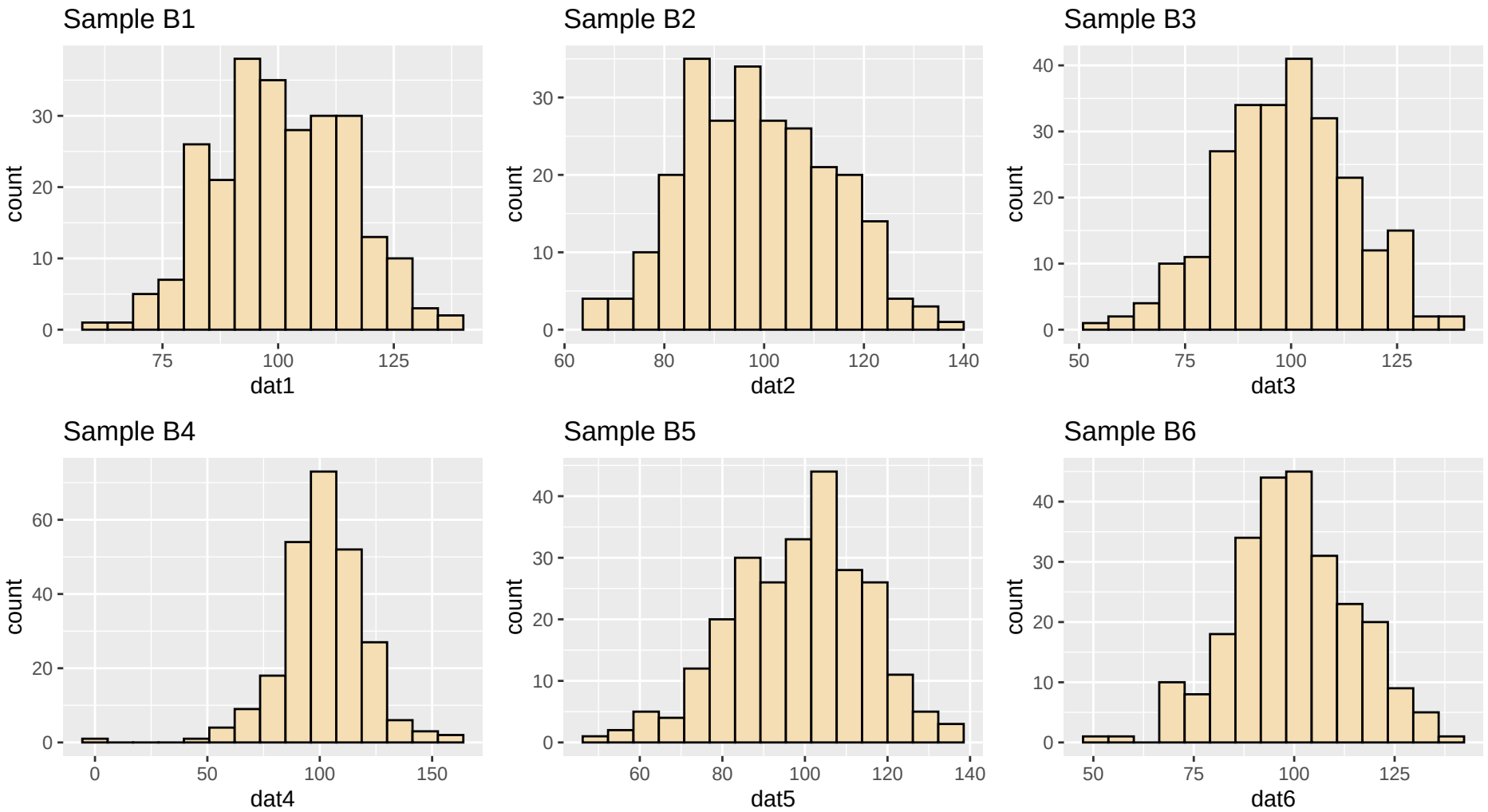
5 simulations of the Normal distribution, one of a heavy-tailed distribution.



Box + Violin Plots of these 6 Samples

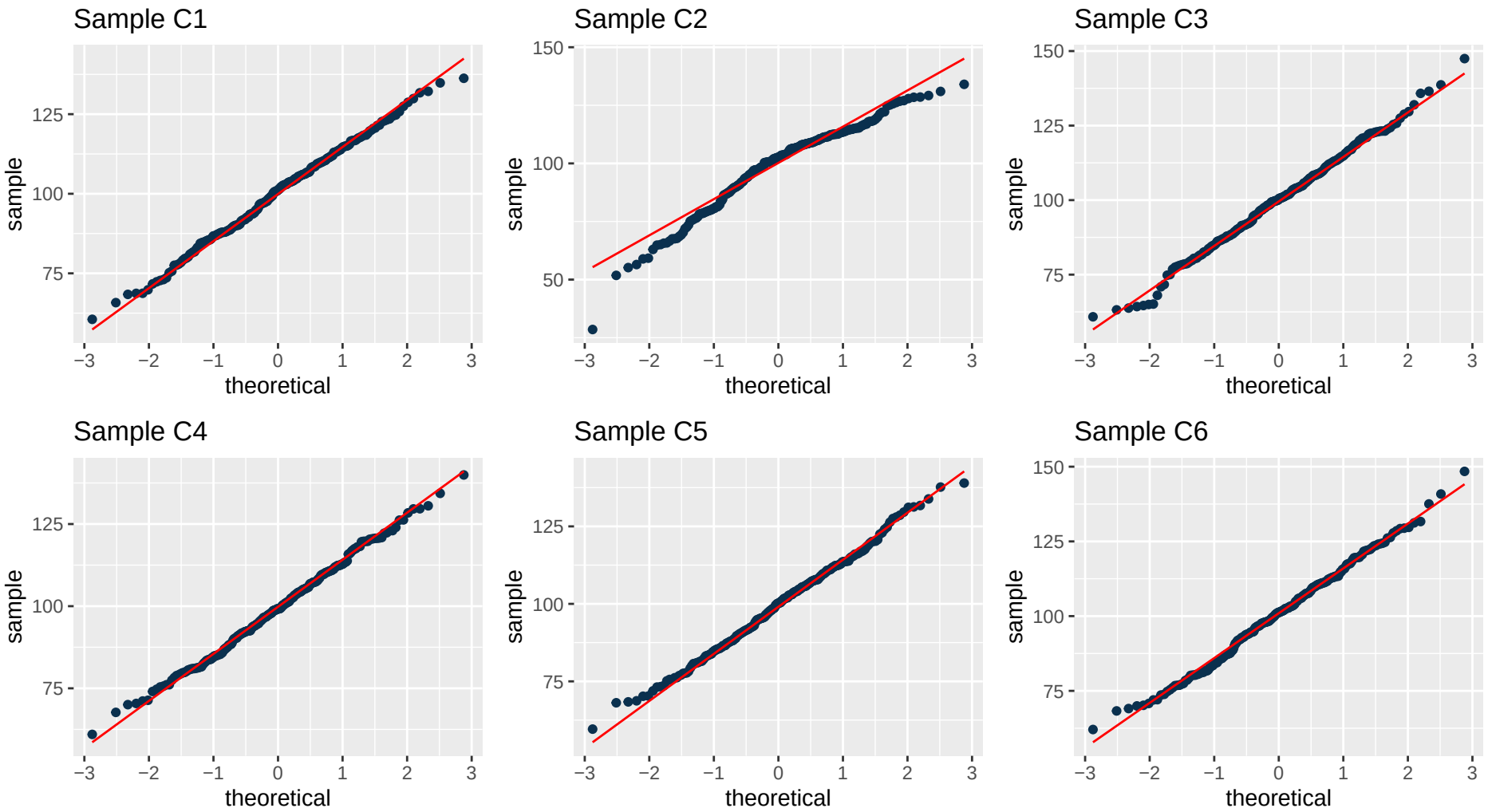


Same Six Simulations, in Histograms

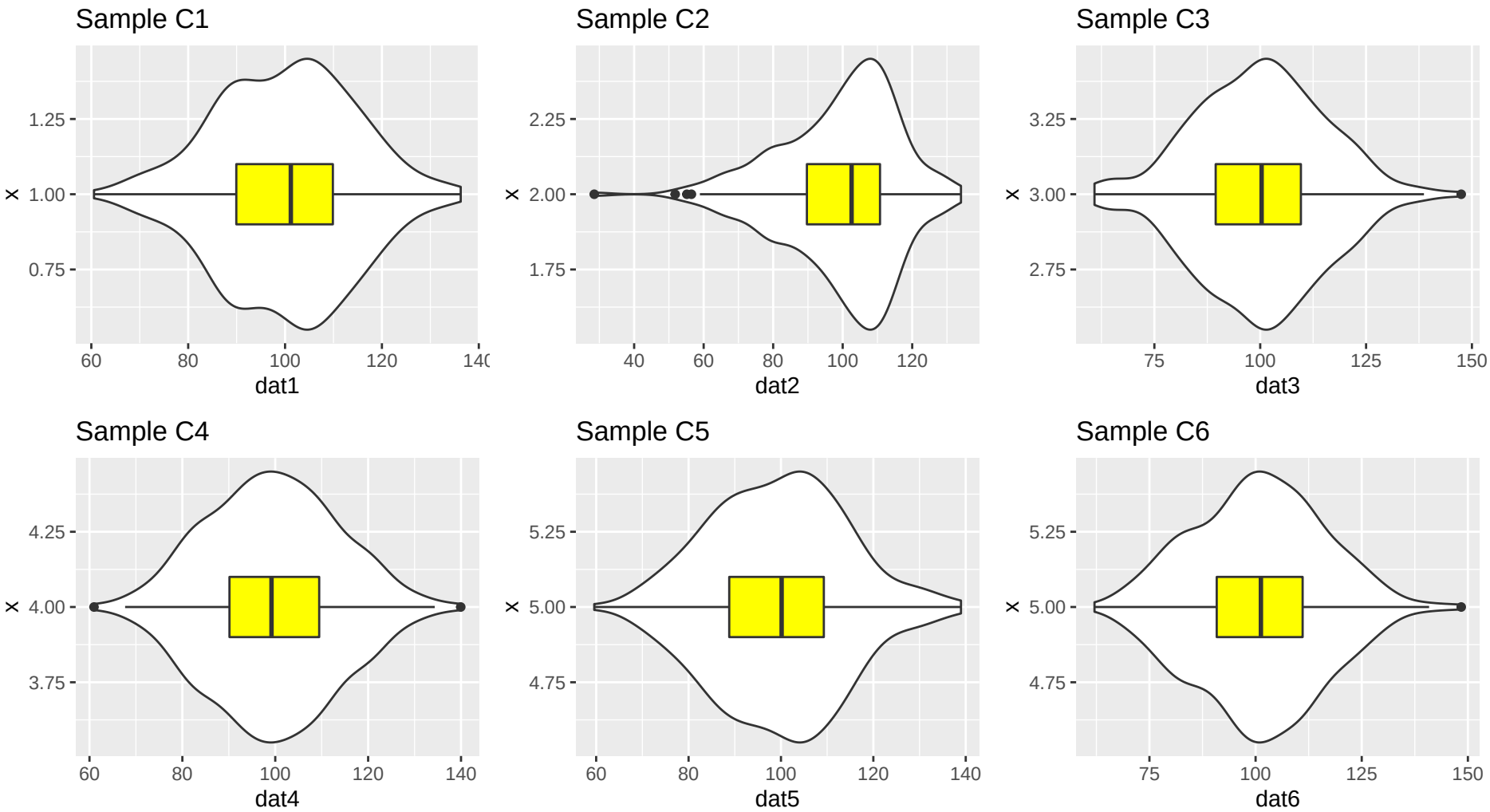


Again, one of these is not like the others

5 simulations of the Normal distribution, one of a left-skewed distribution.



Box + Violin Plots of these 6 Samples

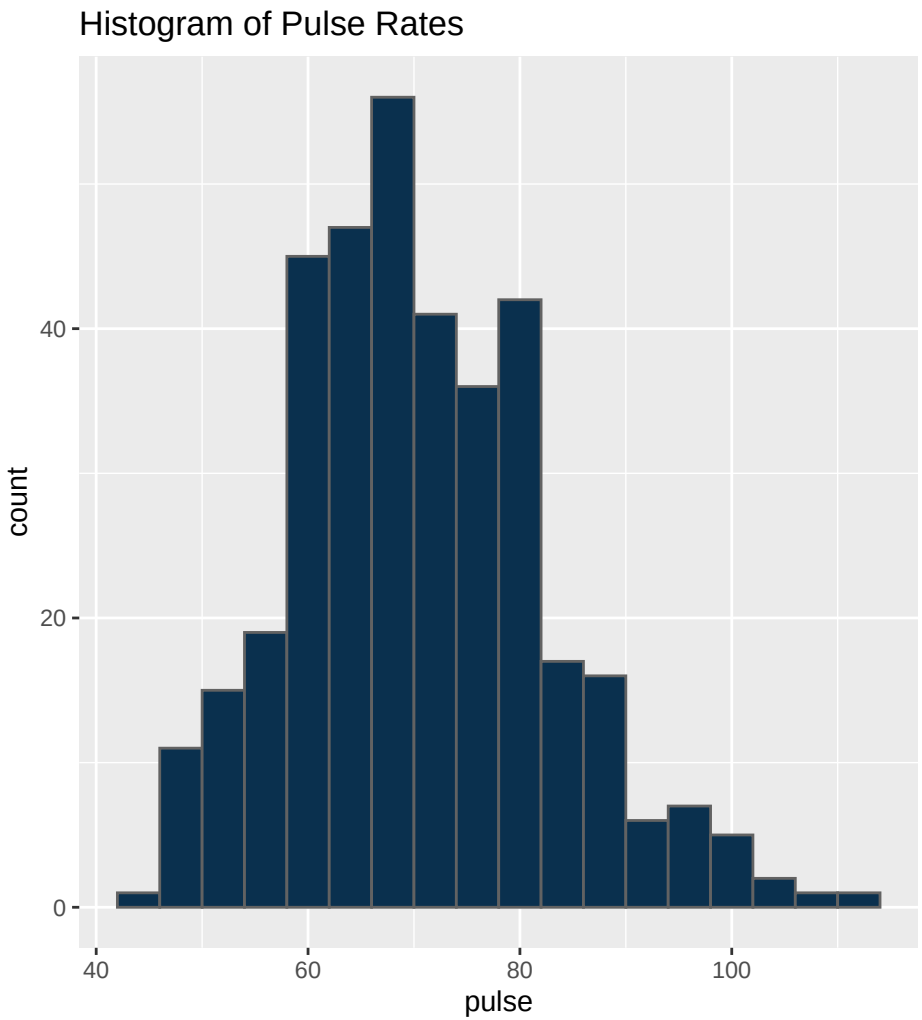


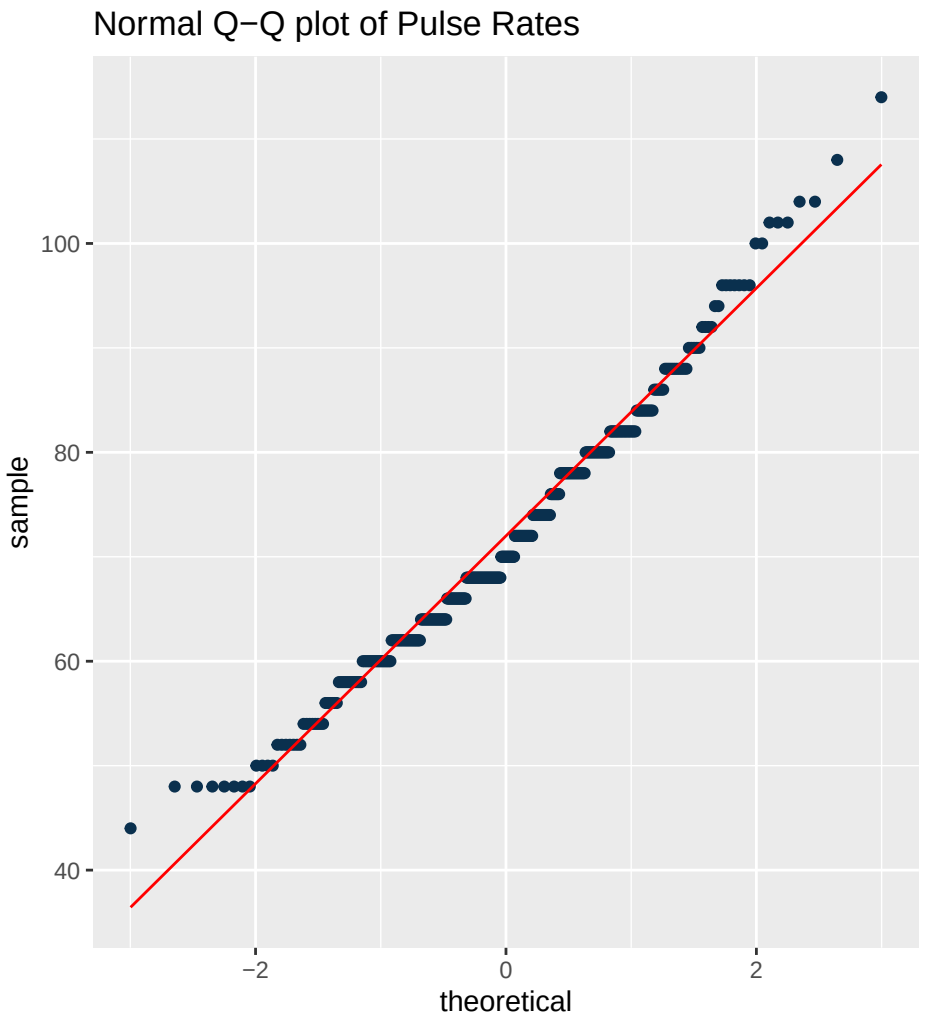
Two plots, side by side

```
plot_a <- ggplot(nh2_GVGmales, aes(x = pulse)) +
  geom_histogram(binwidth = 4,
                 fill = cwru.blue, col = cwru.gray) +
  labs(title = "Histogram of Pulse Rates")

plot_b <- ggplot(nh2_GVGmales, aes(sample = pulse)) +
  geom_qq(col = cwru.blue) + geom_qq_line(col = "red") +
  labs(title = "Normal Q-Q plot of Pulse Rates")

gridExtra::grid.arrange(plot_a, plot_b, ncol = 2)
```

Resulting plot on the next slide. . .

Would a Normal model work well here?



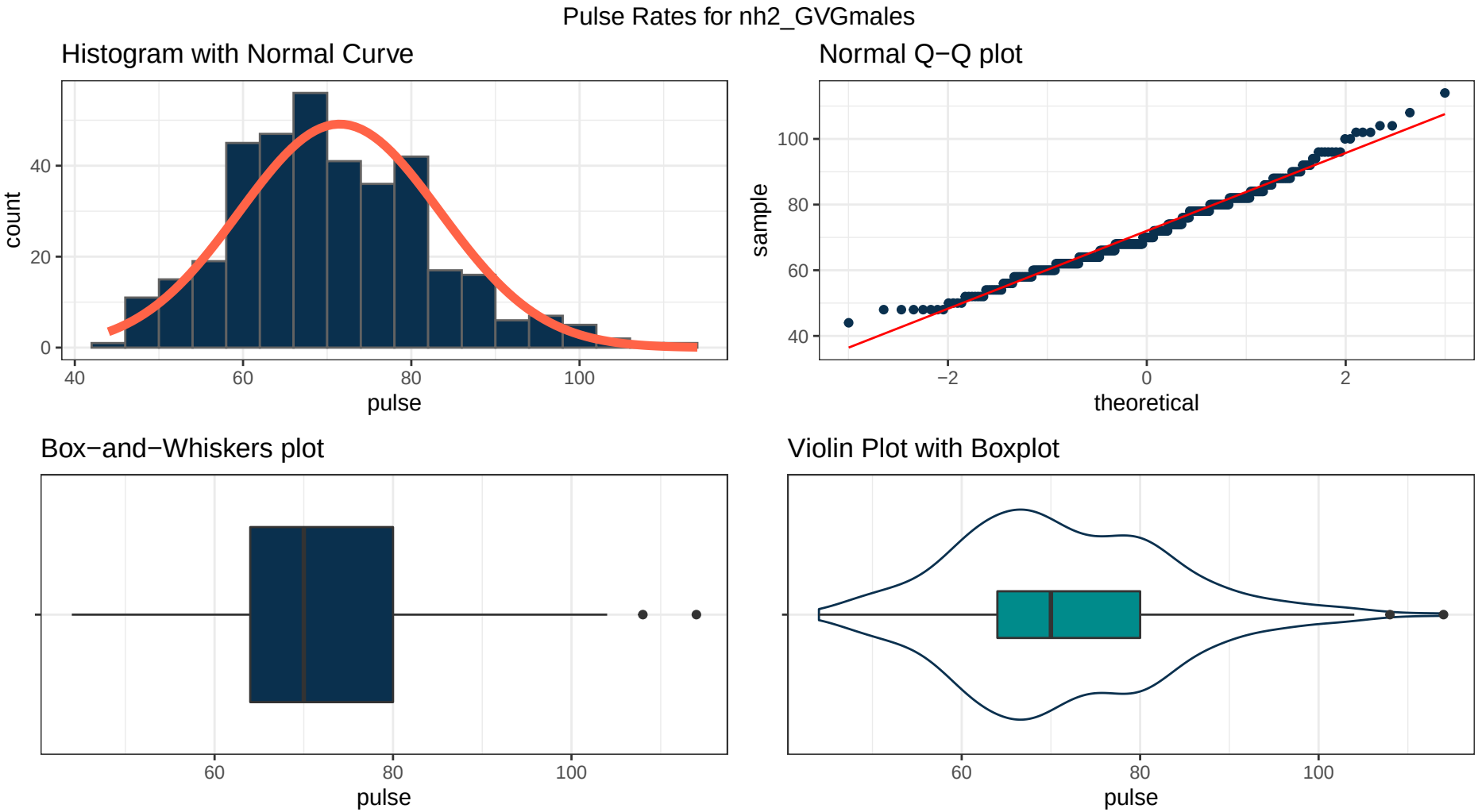


Does a Normal model fit well for my data?

1. Is a Normal Q-Q plot showing something close to a straight line, without clear signs of skew or indications of lots of outliers (heavy-tailedness)?
2. Does a boxplot, violin plot and/or histogram also show a symmetric distribution, where both the number of outliers is modest, and the distance of those outliers from the mean is modest?
3. Do numerical measures match up with the expectations of a normal model?

Let's start by looking at 1 and 2.

Four (potentially) Useful Plots



Does a Normal model fit well for my data?

3. Do numerical measures match up with the expectations of a normal model?

- Is the mean close to the median (perhaps so that $skew_1$ is less than 0.2 in absolute value)?
- In a Normal model, mean $\pm$ 1 standard deviation covers 68% of the data.
- In a Normal model, mean $\pm$ 2 standard deviations covers 95% of the data.
- In a Normal model, mean $\pm$ 3 standard deviations covers 99.7% of the data.

Normal model for pulse rates of nh2_GVGmales?

```
mosaic::favstats(~ pulse, data = nh2_GVGmales)
```

```
 min Q1 median Q3 max     mean       sd   n missing
  44 64     70 80 114 71.48913 11.94811 368       0
```

What is $skew_1$ here?

```
nh2_GVGmales %>%
  summarize(skew1 = (mean(pulse) - median(pulse))/sd(pulse))
```

```
# A tibble: 1 x 1
  skew1
  <dbl>
1 0.125
```

How many of the observations are within 1 SD of the mean?

```
nh2_GVGmales %>%
  count(pulse > mean(pulse) - sd(pulse),
        pulse < mean(pulse) + sd(pulse))
```

```
# A tibble: 3 x 3
  `pulse > mean(pulse) -~ `pulse < mean(pulse) +~     n
  <lgl>                   <lgl>                   <int>
1 FALSE                   TRUE                       46
2 TRUE                    FALSE                      55
3 TRUE                    TRUE                      267
```

So 267 of the 368 (72.6%) observations are within 1 SD of the mean. How does this compare to the expectation under a Normal model?

How about the mean ± 2 standard deviations rule?

The total sample size here is 368.

```
nh2_GVGmales %>%
  count(pulse > mean(pulse) - 2*sd(pulse),
        pulse < mean(pulse) + 2*sd(pulse))
```

```
# A tibble: 3 x 3
  `pulse > mean(pulse) -~ `pulse < mean(pulse) +~     n
  <lgl>                   <lgl>                   <int>
1 FALSE                   TRUE                        1
2 TRUE                    FALSE                      16
3 TRUE                    TRUE                      351
```

So 351 of the 368 (95.4%) observations are within 2 SD of the mean. How does this compare to the expectation under a Normal model?

Hypothesis Testing to assess Normality

Don't. Graphical approaches are **far** better than hypothesis tests.

```
shapiro.test(nh2_GVGmales$pulse)
```

```
    Shapiro-Wilk normality test

data:  nh2_GVGmales$pulse
W = 0.98244, p-value = 0.0001868
```

The very small p value indicates that the test finds some indications **against** adopting a Normal model for these data.

Why not test for Normality?

There are multiple hypothesis testing schemes (Kolmogorov-Smirnov, etc.) and each looks for one specific violation of a Normality assumption. None can capture the wide range of issues our brains can envision, and none by itself is great at its job.

- With any sort of reasonable sample size, the test is so poor at detecting non-normality compared to our eyes, that it finds problems we don't care about and ignores problems we do care about.
- And without a reasonable sample size, the test is essentially useless.

Whenever you *can* avoid hypothesis testing and instead actually plot the data, you should plot the data.

Summing Up: Does a Normal Model fit well?

If a Normal model fits our data well, then we should see the following graphical indications:

1. A histogram that is symmetric and bell-shaped.
2. A boxplot where the box is symmetric around the median, as are the whiskers, without a serious outlier problem.
3. A normal Q-Q plot that essentially falls on a straight line.

As for numerical summaries, we'd like to see

4. The mean and median within 0.2 standard deviation of each other.
5. No real evidence of too many outlier candidates (more than 5% starts to get us concerned about a Normal model)
6. No real evidence of individual outliers outside the reasonable range for the size of our data (we might expect about 3 observations in 1000 to fall more than 3 standard deviations away from the mean.)

Should our data not be well-modeled by the Normal, what can we do?

The Ladder of Power Transformations

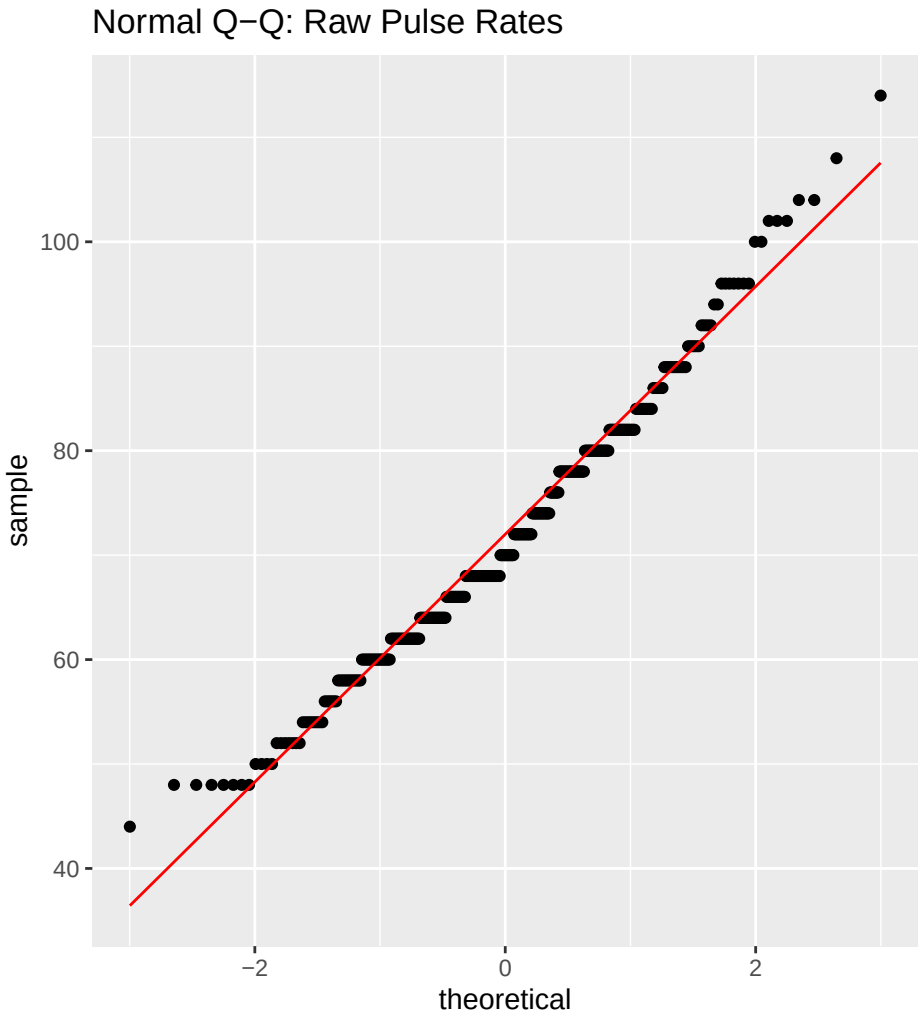
The key notion in re-expression of a single variable to obtain a better fit to a Normal model, is that of a **ladder of power transformations**, which can apply to any unimodal data.

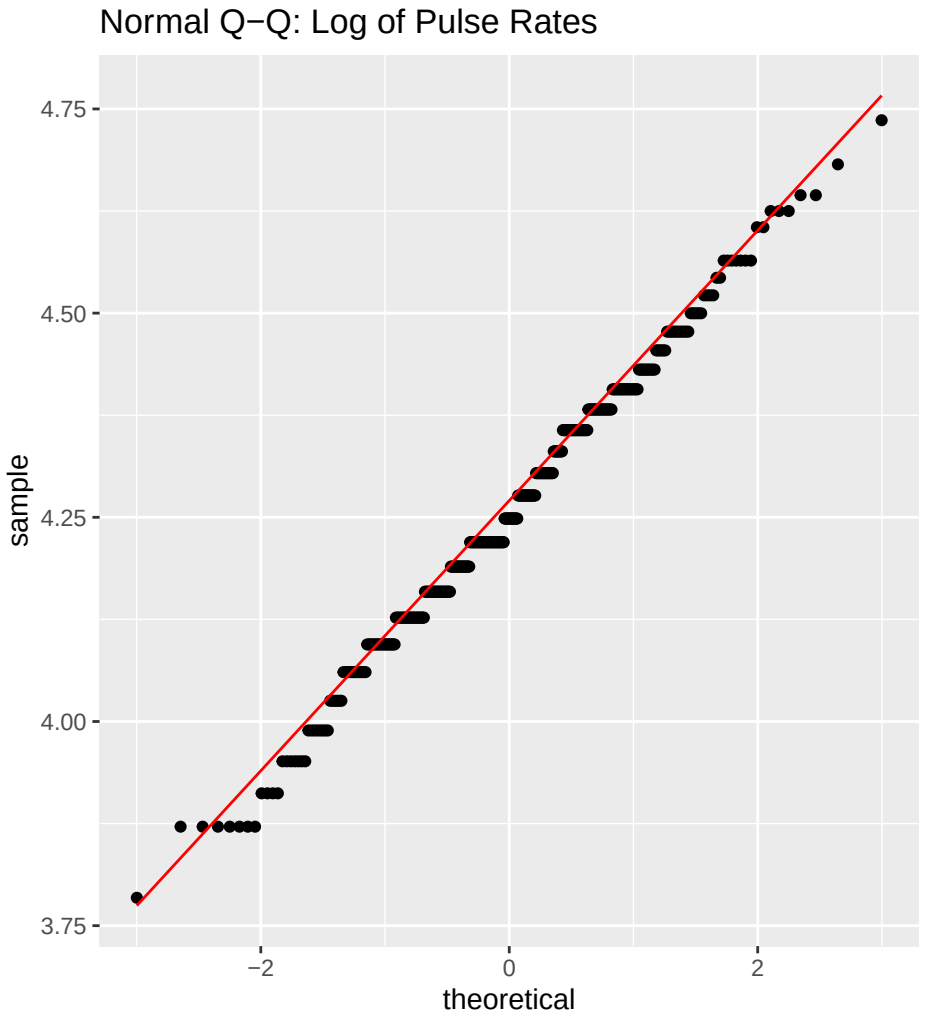
Power	Transformation
3	x^3
2	x^2
1	x (unchanged)
0.5	$x^{0.5} = \sqrt{x}$
0	ln x
-0.5	$x^{-0.5} = 1/\sqrt{x}$
-1	$x^{-1} = 1/x$
-2	$x^{-2} = 1/x^2$

nh2_GVGmales Pulse Rates, and their Natural Logarithms

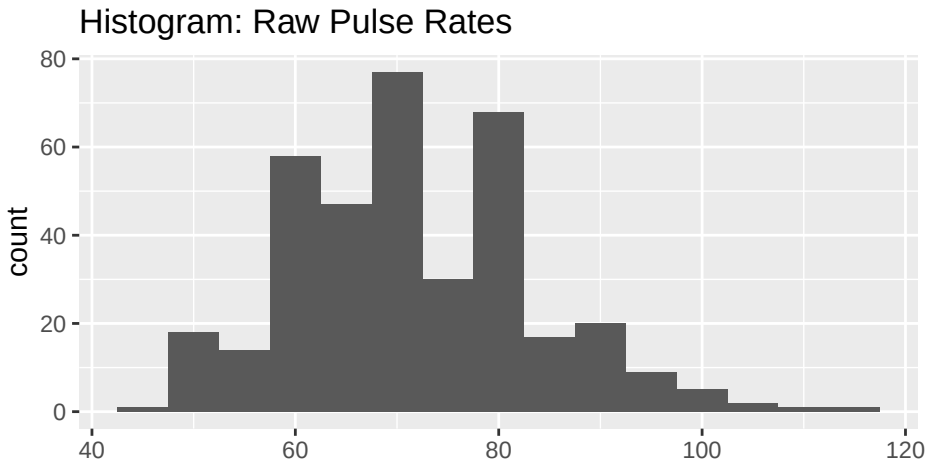
```
p1 <- ggplot(data = nh2_GVGmales, aes(sample = pulse)) +
  geom_qq() + geom_qq_line(col = "red") +
  labs(title = "Normal Q-Q: Raw Pulse Rates")

p2 <-  ggplot(data = nh2_GVGmales, aes(sample = log(pulse))) +
  geom_qq() + geom_qq_line(col = "red") +
  labs(title = "Normal Q-Q: Logarithm of Pulse Rates")

gridExtra::grid.arrange(p1, p2, ncol = 2)
```

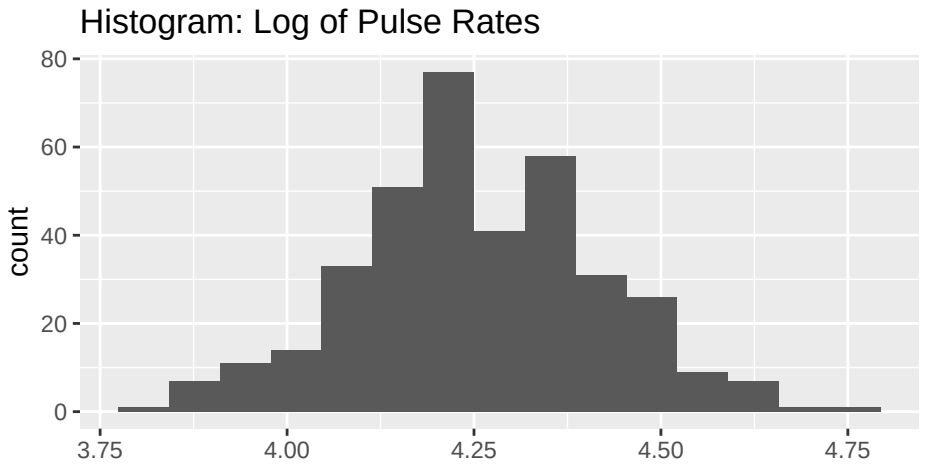
nh2_GVGmales Pulse Rates, and their Natural Logarithms

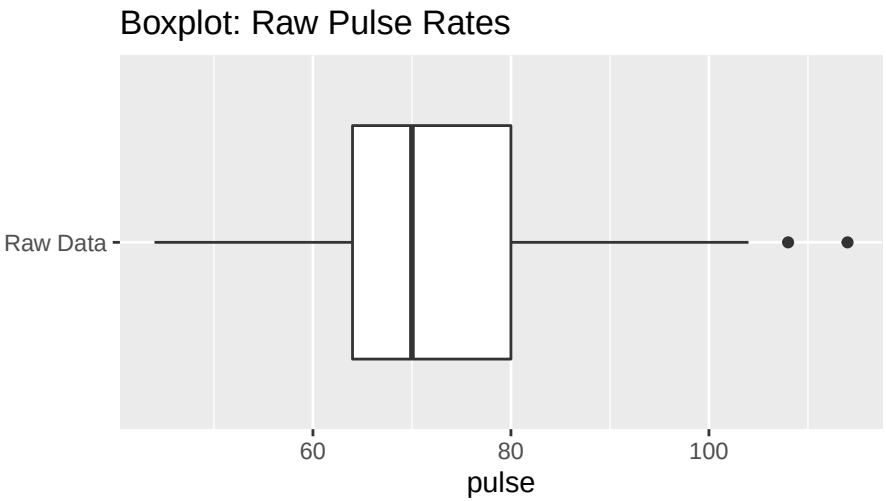


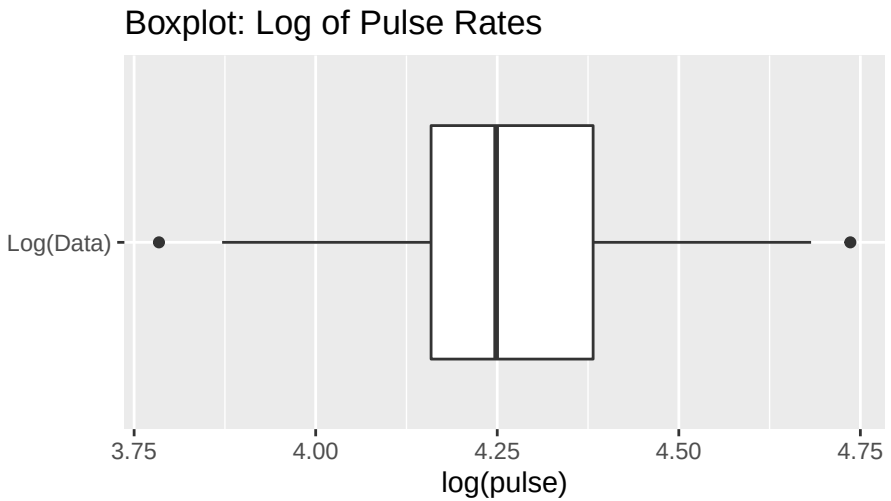


`nh2_GVGmales` Pulse Rates, and their Natural Logarithms







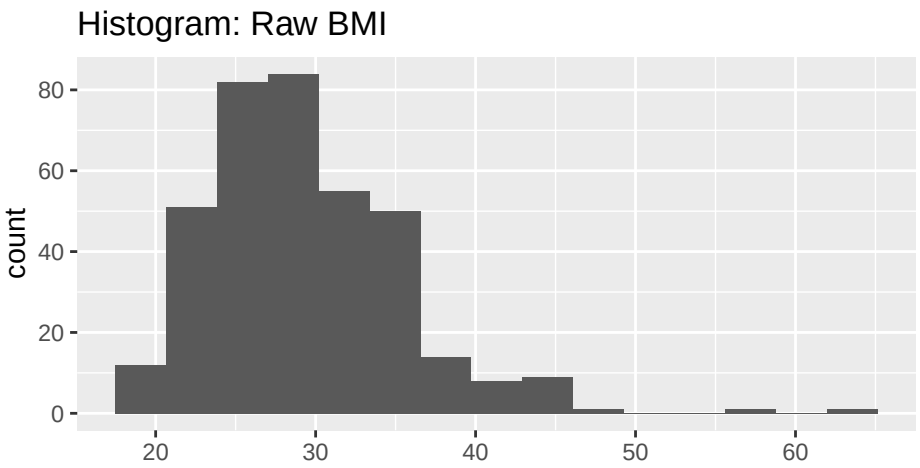


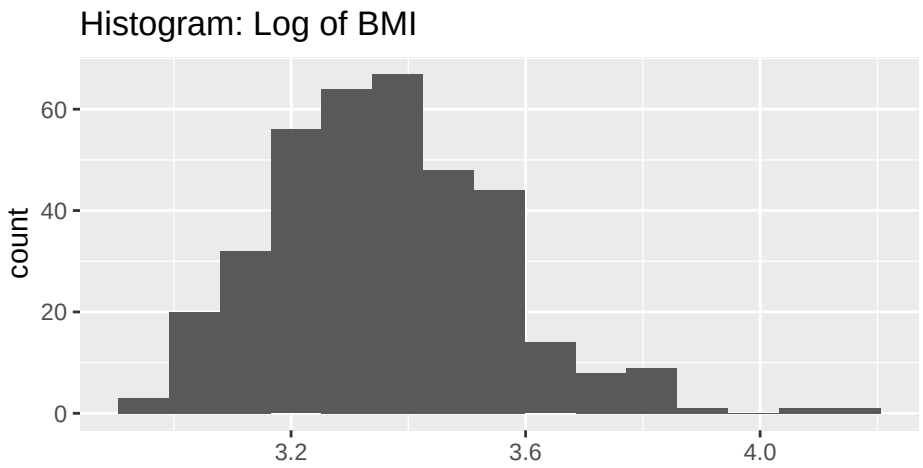
Using the Ladder

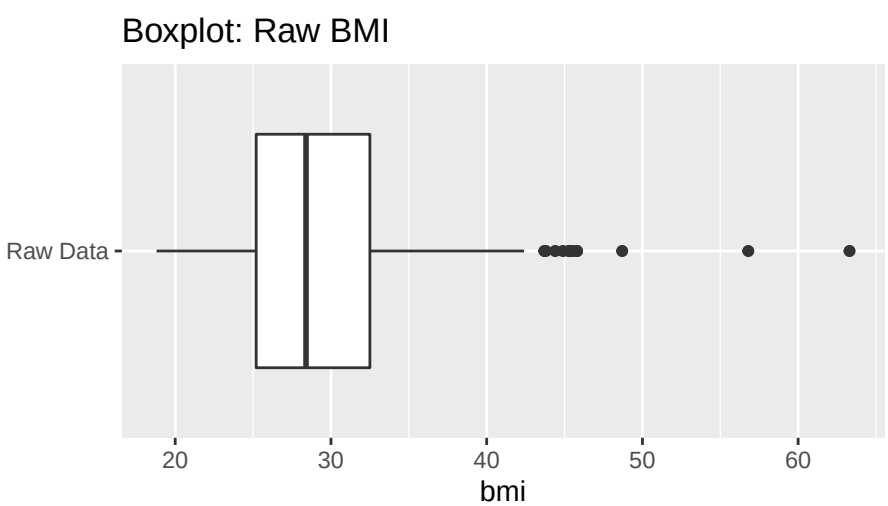
- The ladder is most useful for strictly positive, ratio variables.
- Sometimes, if 0 is a value in the data set, we will add 1 to each value before applying a transformation like the logarithm.
- Interpretability is often an important criterion, although back-transformation at the end of an analysis is usually a sensible strategy.

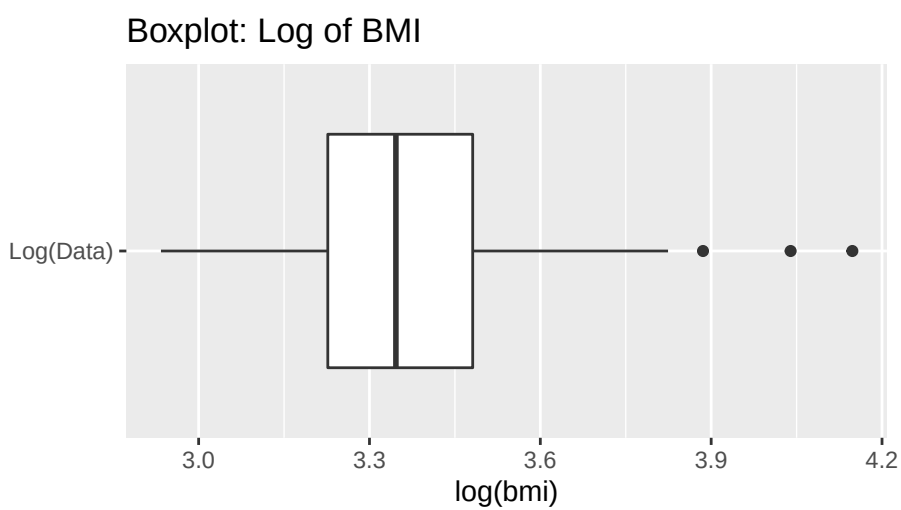
Power	-2	-1	-0.5	0	0.5	1	2	3
Transformation	$1/x^2$	$1/x$	$1/\sqrt{x}$	ln x	$\sqrt{x}$	x	x^2	x^3

nh2_GVGmales BMI Data (Raw data and Log)

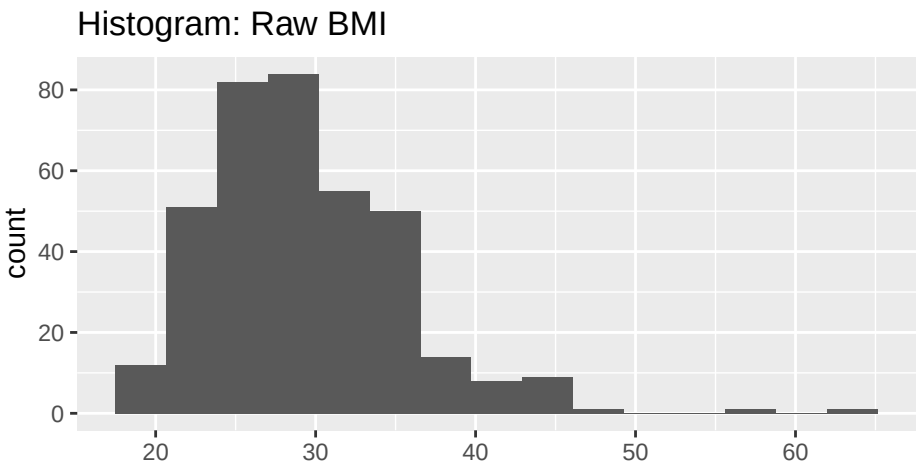


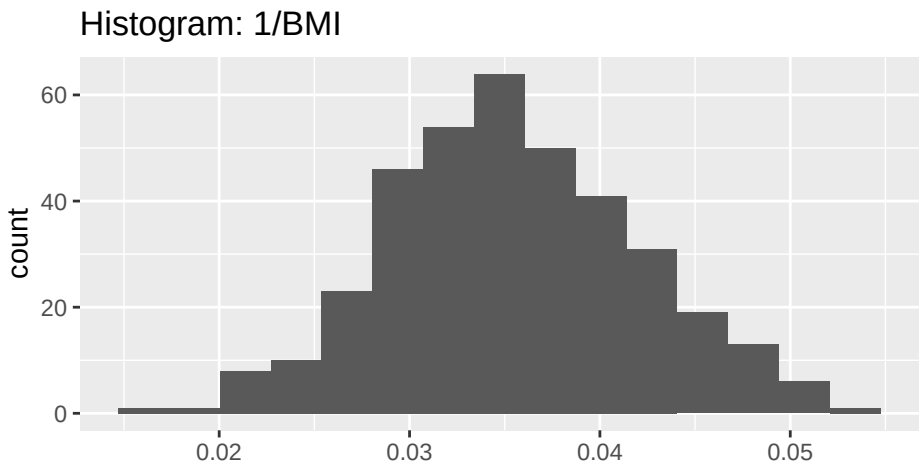


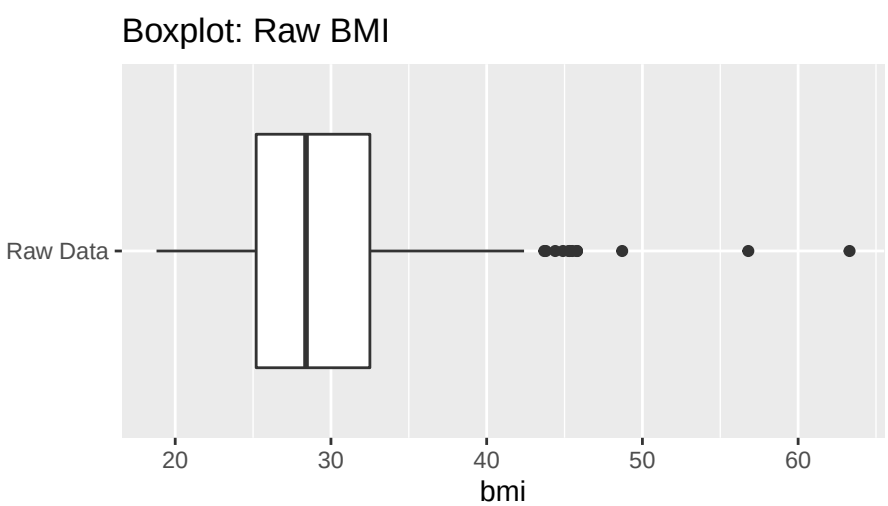


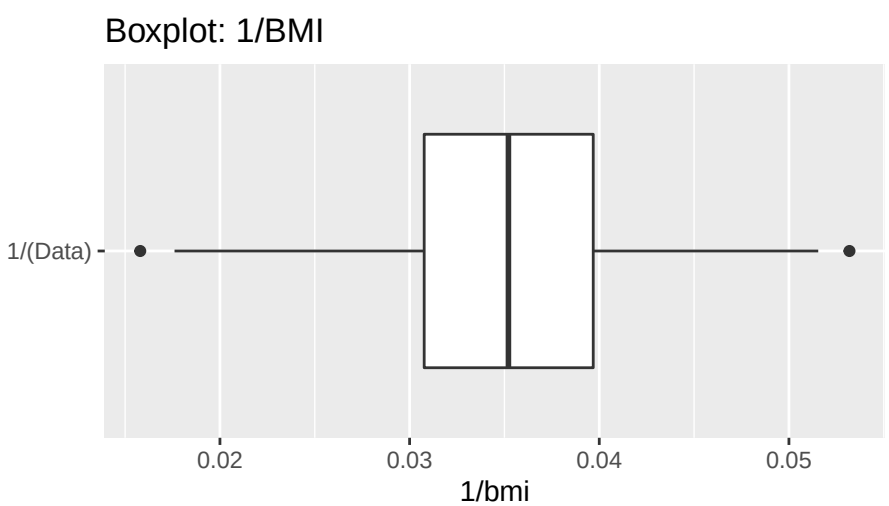


nh2_GVGmales BMI - down the ladder to 1/BMI?

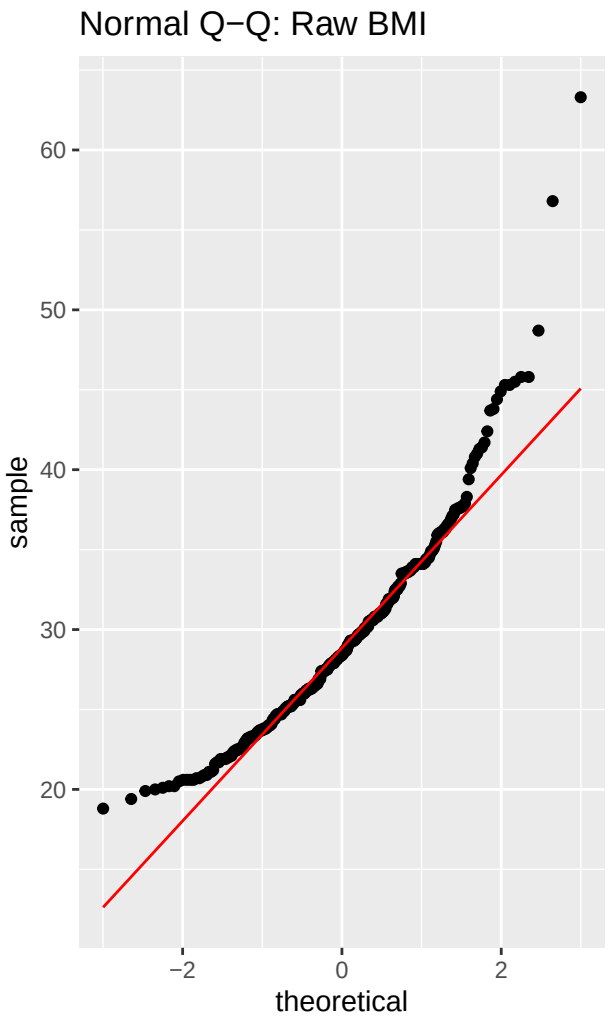


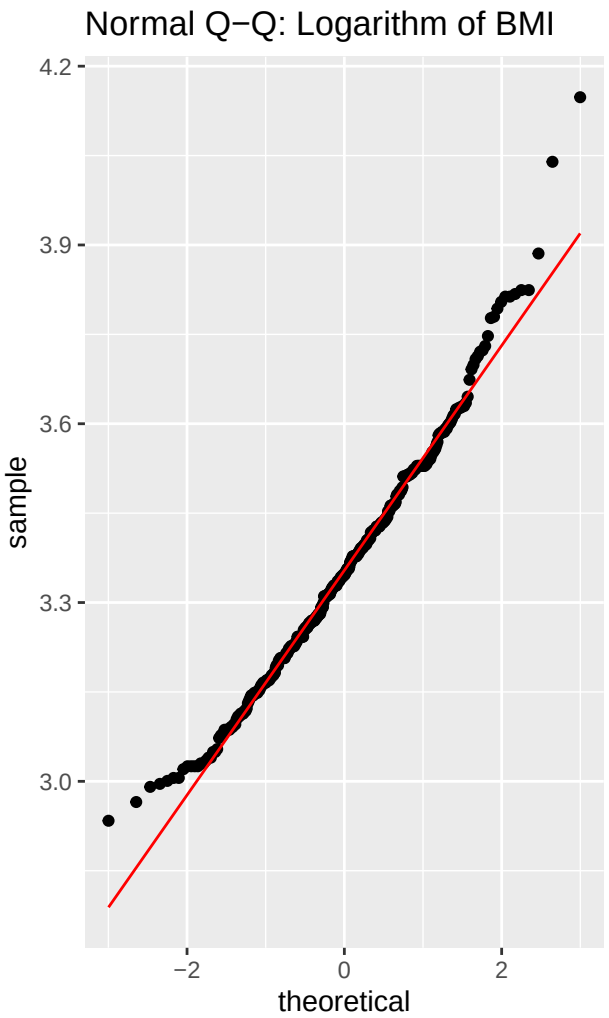


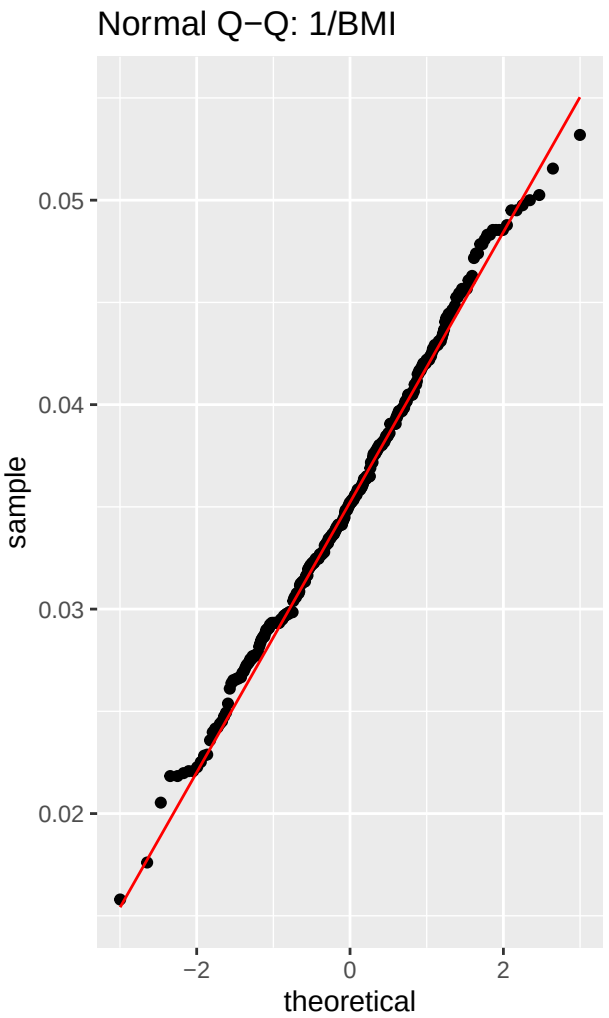




Normal Q-Q plots for BMI







Again, does a Normal Model fit our data?

If a Normal model fits our data well, then we should see the following graphical indications:

1. A histogram that is symmetric and bell-shaped.
2. A boxplot where the box is symmetric around the median, as are the whiskers, without a serious outlier problem.
3. A normal Q-Q plot that essentially falls on a straight line.

As for numerical summaries, we'd like to see

4. The mean and median within 0.2 standard deviation of each other.
5. No real evidence of too many outlier candidates (more than 5% starts to get us concerned about a Normal model)
6. No real evidence of individual outliers outside the reasonable range for the size of our data (we might expect about 3 observations in 1000 to fall more than 3 standard deviations away from the mean.)

Part 2 (Scatterplots, Linear Models, Correlation)

Today's Agenda, Part 2 (See Notes, Chapter 11)

Repeating . . .

1. A New Data Set (!)
2. Studying Scatterplots
3. Building Linear Models
    - Making predictions with PIs and CIs
    - Fundamental Summaries of a Regression Model
    - Understanding Regression Residuals
4. Measuring Association with Correlations
    - Pearson and Spearman approaches
    - Thinking about the impact of transformations
5. Adding a categorical predictor (factor) to a model
    - Using `fct_recode` from `forcats` (tidyverse)
    - Interpreting an indicator variable regression

Part 2 Data Load

```
VHL <- read_csv("vonHippel-Lindau.csv")

dim(VHL)
```

```
[1] 37  4
```

Von Hippel - Lindau study Codebook

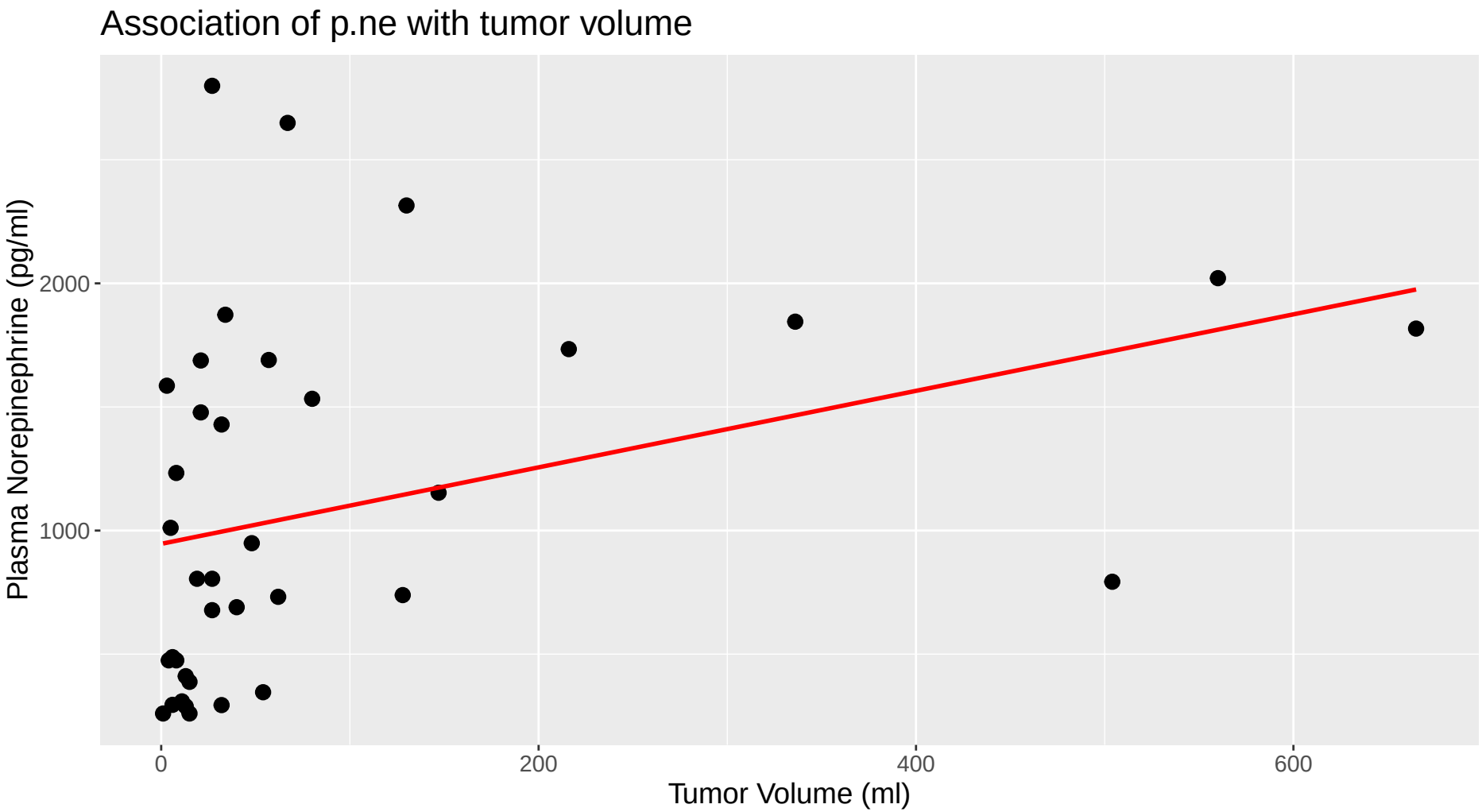
- `p.ne` = plasma norepinephrine (pg/ml)
- `tumorvol` = tumor volume (ml)
- `disease` = 1 for patients with multiple endocrine neoplasia type 2
- `disease` = 0 for patients with von Hippel-Lindau disease

```
head(VHL, 3)
```

```
# A tibble: 3 x 4
     id disease  p.ne tumorvol
  <dbl>   <dbl> <dbl>    <dbl>
1   101       0   289       13
2   102       1   294       32
3   103       0  2799       27
```

First, we want to describe the association of `p.ne` and `tumorvol`.

Scatterplot predicting `tumorvol` from `p.ne`



The Linear Model

```
model1 <- lm(p.ne ~ tumorvol, data = VHL)
model1
```

```
Call:
lm(formula = p.ne ~ tumorvol, data = VHL)

Coefficients:
(Intercept)     tumorvol
    946.185        1.547
```

The (simple regression / prediction / ordinary least squares) model is

- p.ne = 946.2 + 1.55 * tumorvol.

Using the model to make predictions (PI)

To predict the `p.ne` for a subject with tumor volume 200 ml, we have

- `p.ne` = 946.2 + 1.55 * 200

A 95% **prediction interval** for a single subject with volume 200 ml. . .

```
predict(model1, newdata = tibble(tumorvol = 200),
        interval = "prediction", level = 0.95)
```

```
       fit        lwr      upr
1 1255.666 -162.3308 2673.662
```

Using the model to make predictions (CI)

To predict the `p.ne` for the average of many subjects each with tumor volume 200 ml, we have

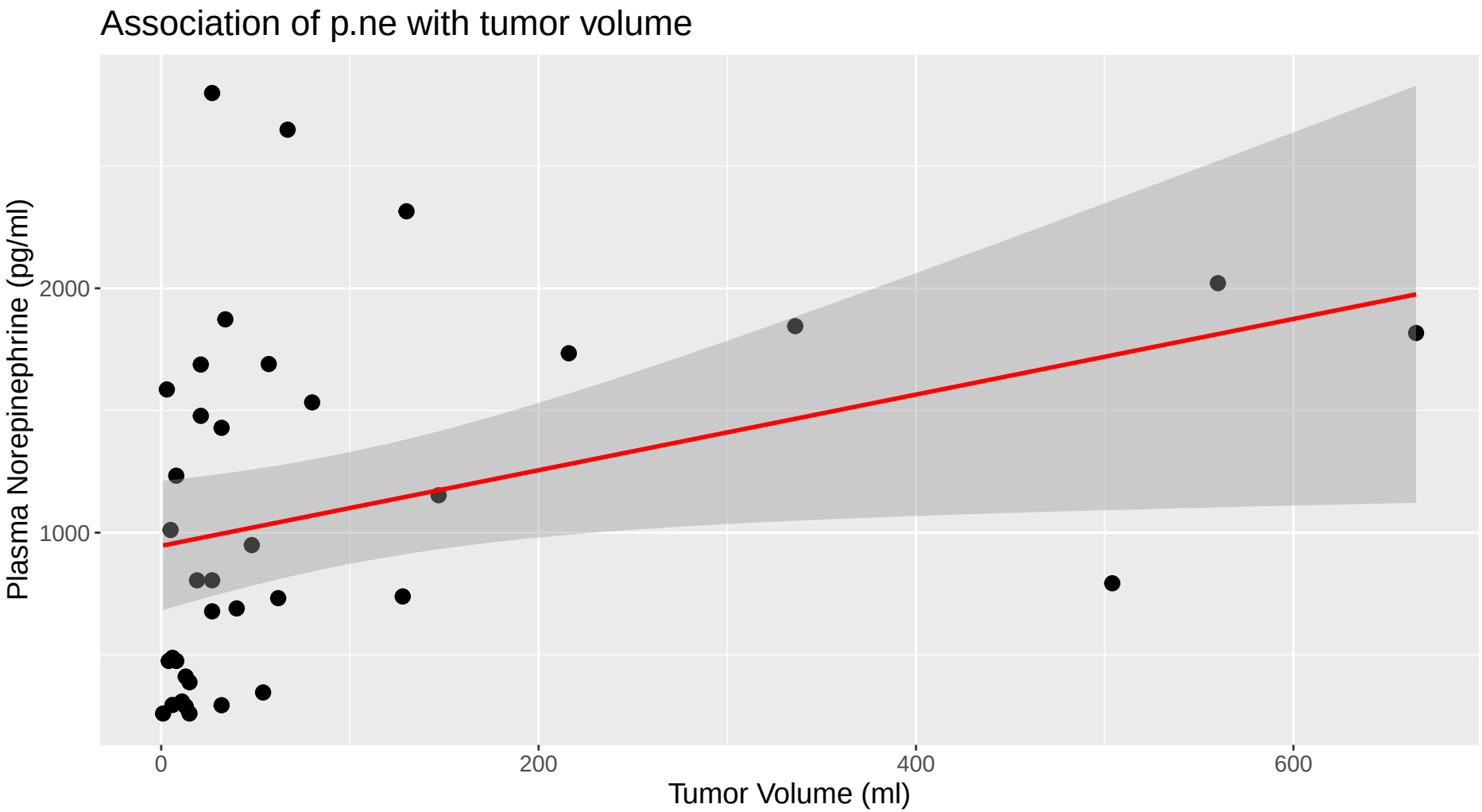
- `p.ne` = 946.2 + 1.55 * 200

A 95% **confidence interval** for the population average of all subjects with volume 200 ml...

```
predict(model1, newdata = tibble(tumorvol = 200),
        interval = "confidence", level = 0.95)
```

```
       fit      lwr      upr
1 1255.666 980.1149 1531.217
```

Adding a Confidence Interval to the Scatterplot



Summary of our Linear (OLS) Model

```
> summary(model1)

Call:
lm(formula = p.ne ~ tumorvol, data = VHL)

Residuals:
   Min     1Q Median     3Q    Max
-933.1 -555.3 -170.6  453.6 1811.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 946.1846   130.4810   7.252 1.81e-08 ***
tumorvol      1.5474     0.7079   2.186   0.0356 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 685.2 on 35 degrees of freedom
Multiple R-squared:  0.1201,    Adjusted R-squared:  0.09497
F-statistic: 4.778 on 1 and 35 DF,  p-value: 0.03561
```

Key Elements of the Summary (1)

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 946.1846   130.4810   7.252 1.81e-08 ***
tumorvol      1.5474     0.7079   2.186   0.0356 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

- The straight line model for these data fitted by ordinary least squares is p.ne = 946 + 1.55 `tumorvol`.
- The slope of `tumorvol` is positive, which indicates that as `tumorvol` increases, we expect that `p.ne` will also increase.
- Specifically, we expect that for every additional ml of `tumorvol`, the `p.ne` is increased by 1.55 pg/ml.

Tidying the Model Coefficients

```
model1 <- lm(p.ne ~ tumorvol, data = VHL)

broom::tidy(model1, conf.int = TRUE) %>%
  knitr::kable(digits = 2)
```

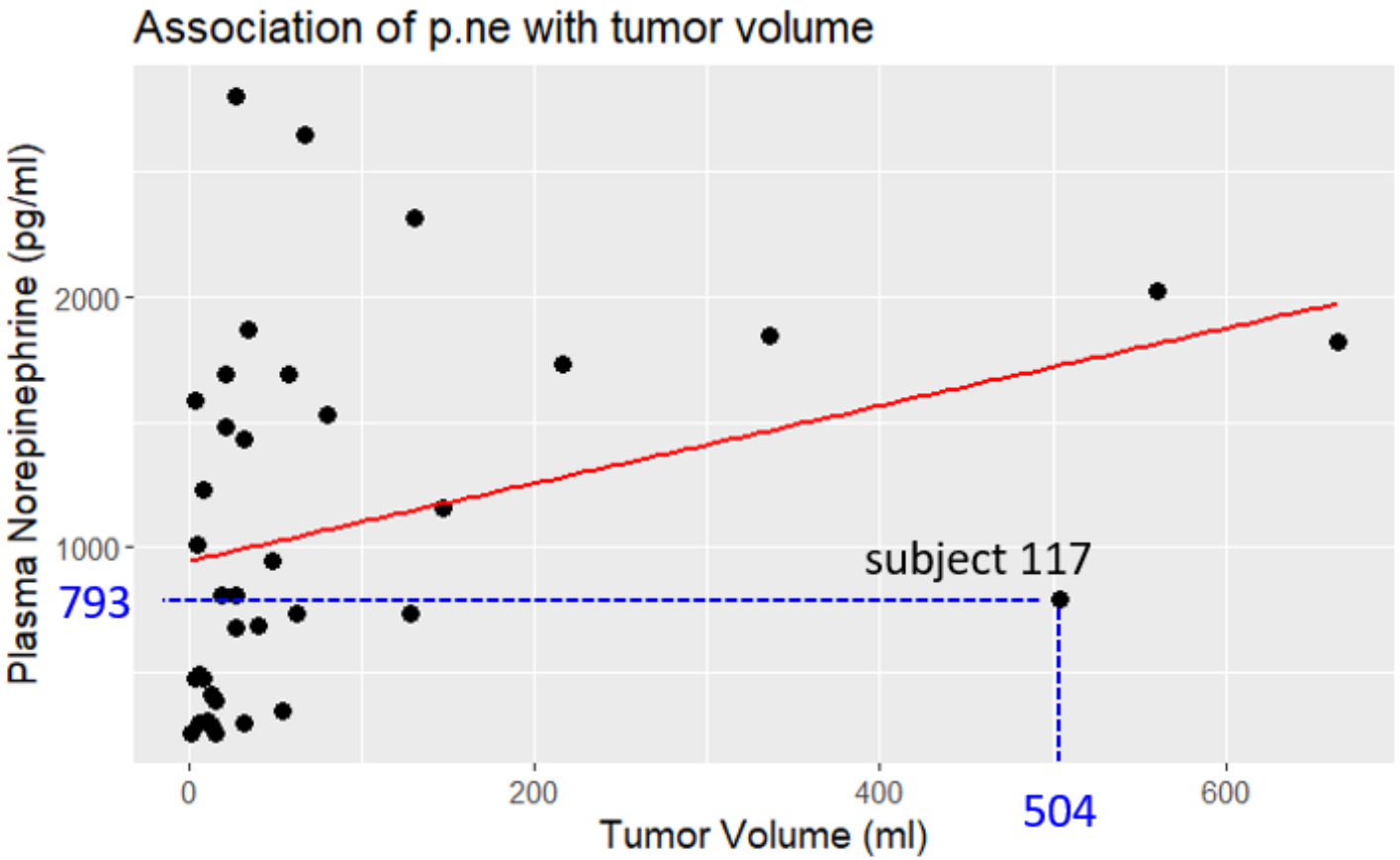
term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	946.18	130.48	7.25	0.00	681.29	1211.08
tumorvol	1.55	0.71	2.19	0.04	0.11	2.98

Key Elements of the Summary (2)

```
Call:
lm(formula = p.ne ~ tumorvol, data = VHL)

Residuals:
    Min      1Q  Median      3Q     Max
-933.1 -555.3 -170.6  453.6 1811.0
```

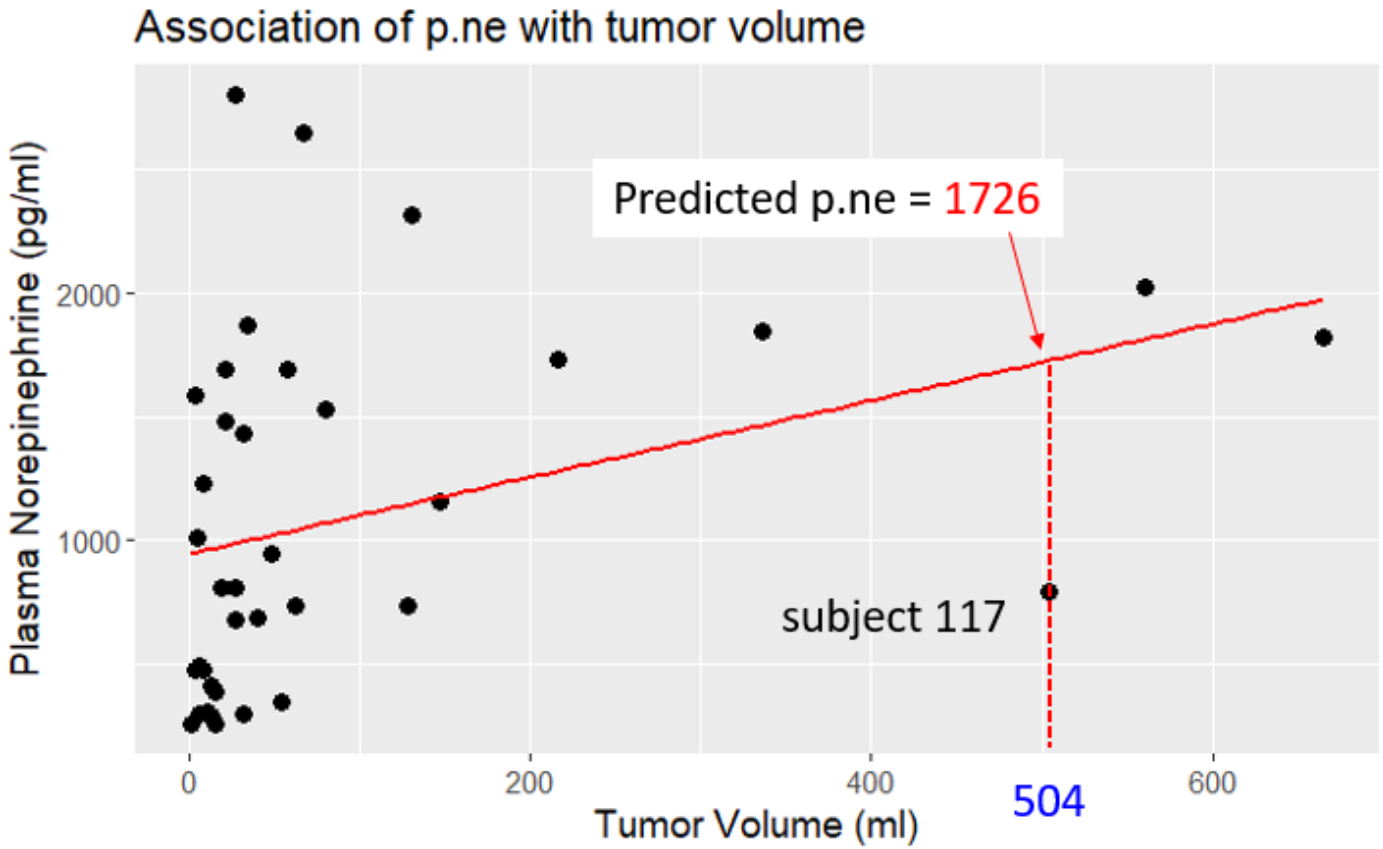
- Here, the **outcome** is `p.ne`, and the **predictor** is `tumorvol`.
- The **residuals** are the observed `p.ne` values minus the model's predicted `p.ne`. The sample residuals are the prediction errors.
  - The biggest miss is for a subject whose observed `p.ne` was 1,811 pg/nl higher than the model predicts based on the subject's tumor volume.
  - The mean residual will always be zero in an OLS model.

Understanding Regression Residuals (A)



Subject 117 has tumorvol = 504, and observed p.ne = 793 pg/nl.

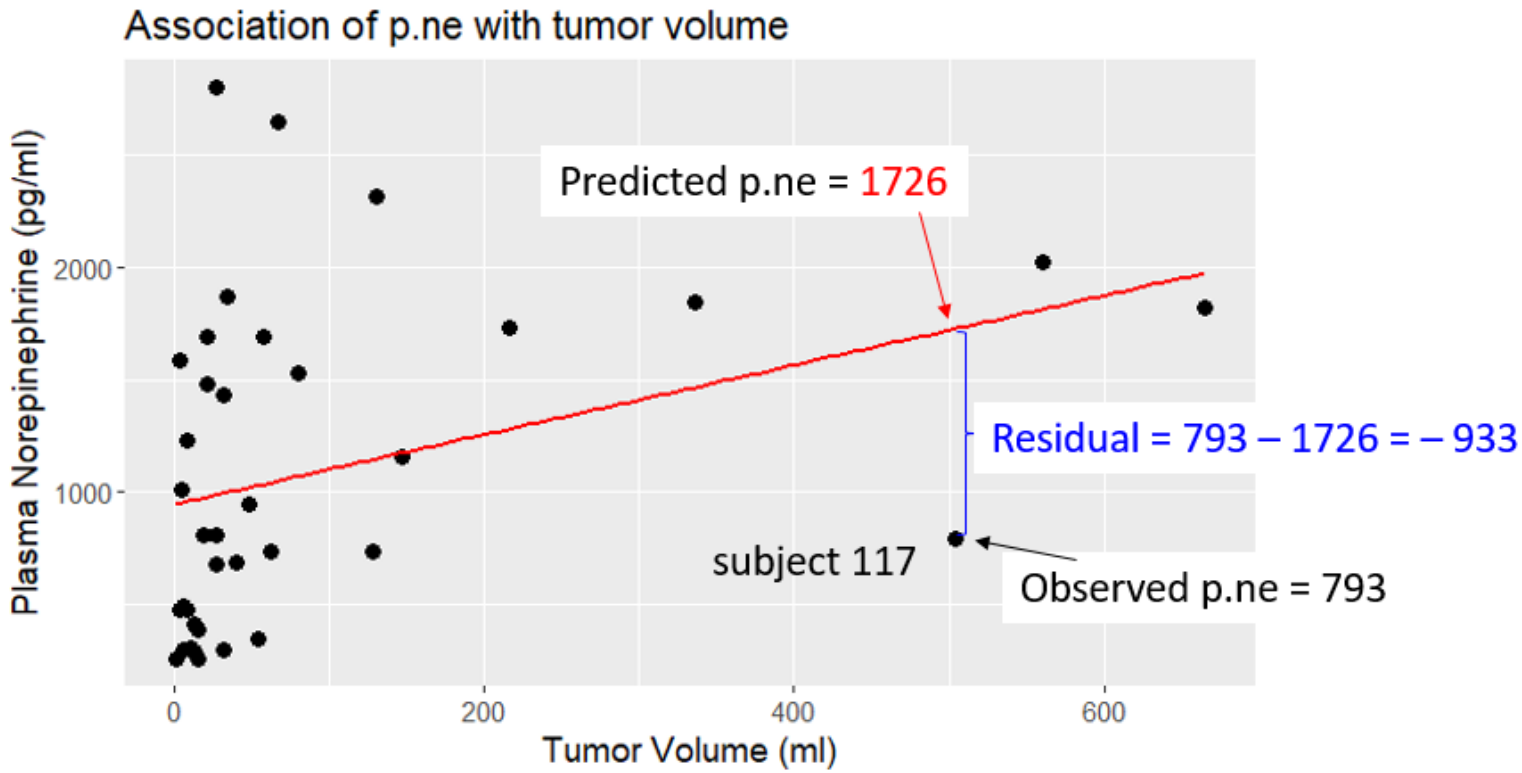
Understanding Regression Residuals (B)



Subject 117 has tumorvol = 504, and observed p.ne = 793 pg/nl.

Model predicts p.ne is 946.2 + 1.55(504) = 1726 pg/nl.

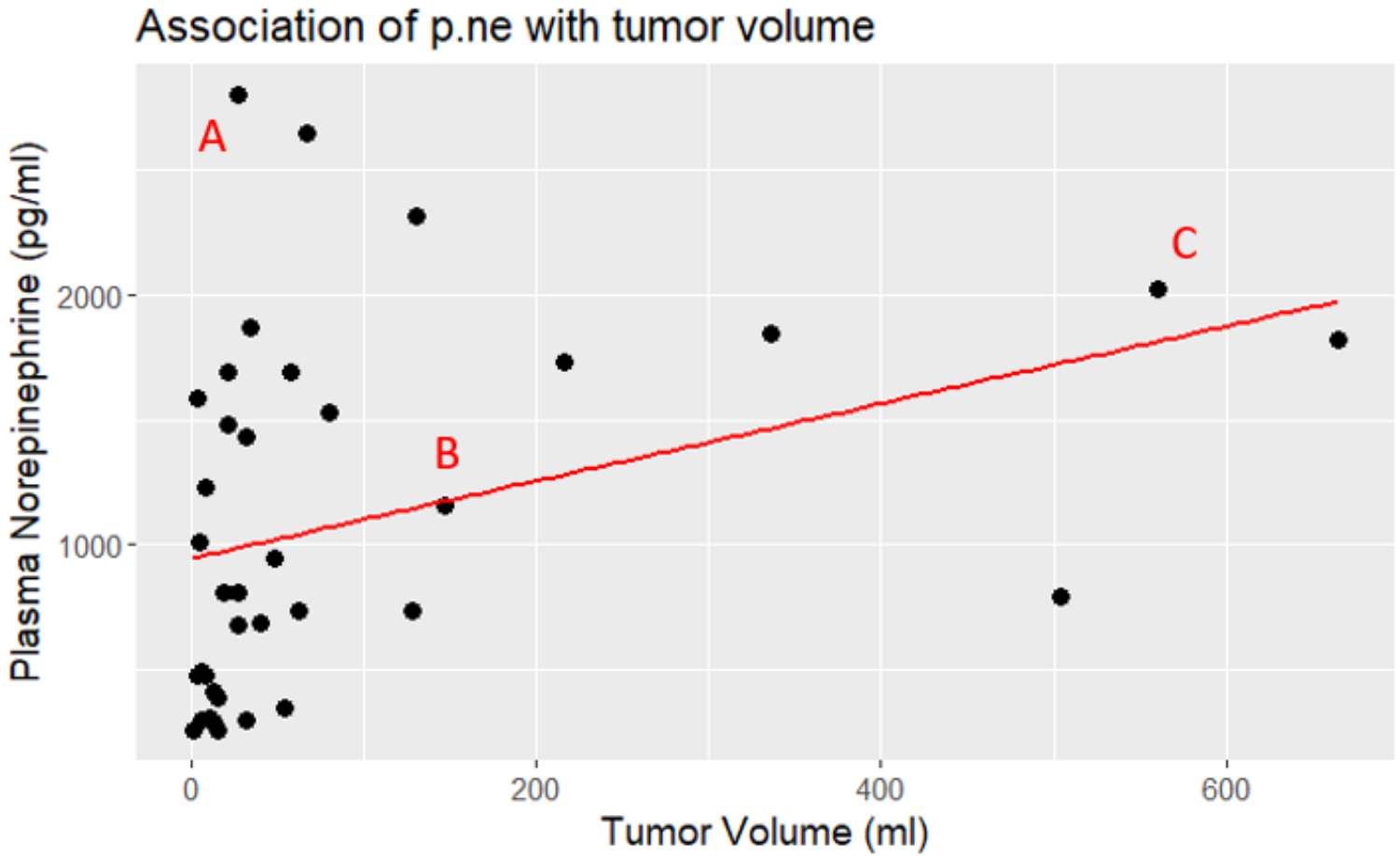
Understanding Regression Residuals (C)



Subject 117 has tumorvol = 504, and observed p.ne = 793 pg/nl.

Model predicts p.ne is 946.2 + 1.55(504) = 1726. So, residual = 793 – 1726 = **-933**

Understanding Regression Residuals (D)



Which point (A, B or C) has the largest positive residual?

Key Elements of the Summary (3)

```
Residual standard error: 685.2 on 35 degrees of freedom
Multiple R-squared:  0.1201,    Adjusted R-squared:  0.09497
F-statistic: 4.778 on 1 and 35 DF,  p-value: 0.03561
```

- The multiple R-squared (squared correlation coefficient) is 0.12, which implies that 12% of the variation in `p.ne` is explained using this linear model with `tumorvol`.
- It also implies that the Pearson correlation between `p.ne` and `tumorvol` is the square root of 0.12, or 0.347.

```
cor(VHL$p.ne, VHL$tumorvol)
```

```
[1] 0.3465646
```

Model 1, summarized at a glance, with broom

```
broom::glance(model1)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic p.value    df
      <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>
1     0.120        0.0950  685.      4.78  0.0356     2
# ... with 5 more variables: logLik <dbl>, AIC <dbl>,
#   BIC <dbl>, deviance <dbl>, df.residual <int>
```

Let's look at the elements of this. . .

Key Elements of `glance` for us now...

```
broom::glance(model1) %>%
  select(r.squared, adj.r.squared, sigma) %>%
  knitr::kable(digits = 3)
```

r.squared	adj.r.squared	sigma
0.12	0.095	685.168

Correlation Coefficients

Two key types of correlation coefficient to describe an association between quantities.

- The one most often used is called the *Pearson* correlation coefficient, symbolized r or sometimes rho (ρ).
- Another is the Spearman rank correlation coefficient, also symbolized by ρ, or sometimes ρ_s.

```
cor(VHL$p.ne, VHL$tumorvol)
```

```
[1] 0.3465646
```

```
cor(VHL$p.ne, VHL$tumorvol, method = "spearman")
```

```
[1] 0.5414319
```

Meaning of Pearson Correlation

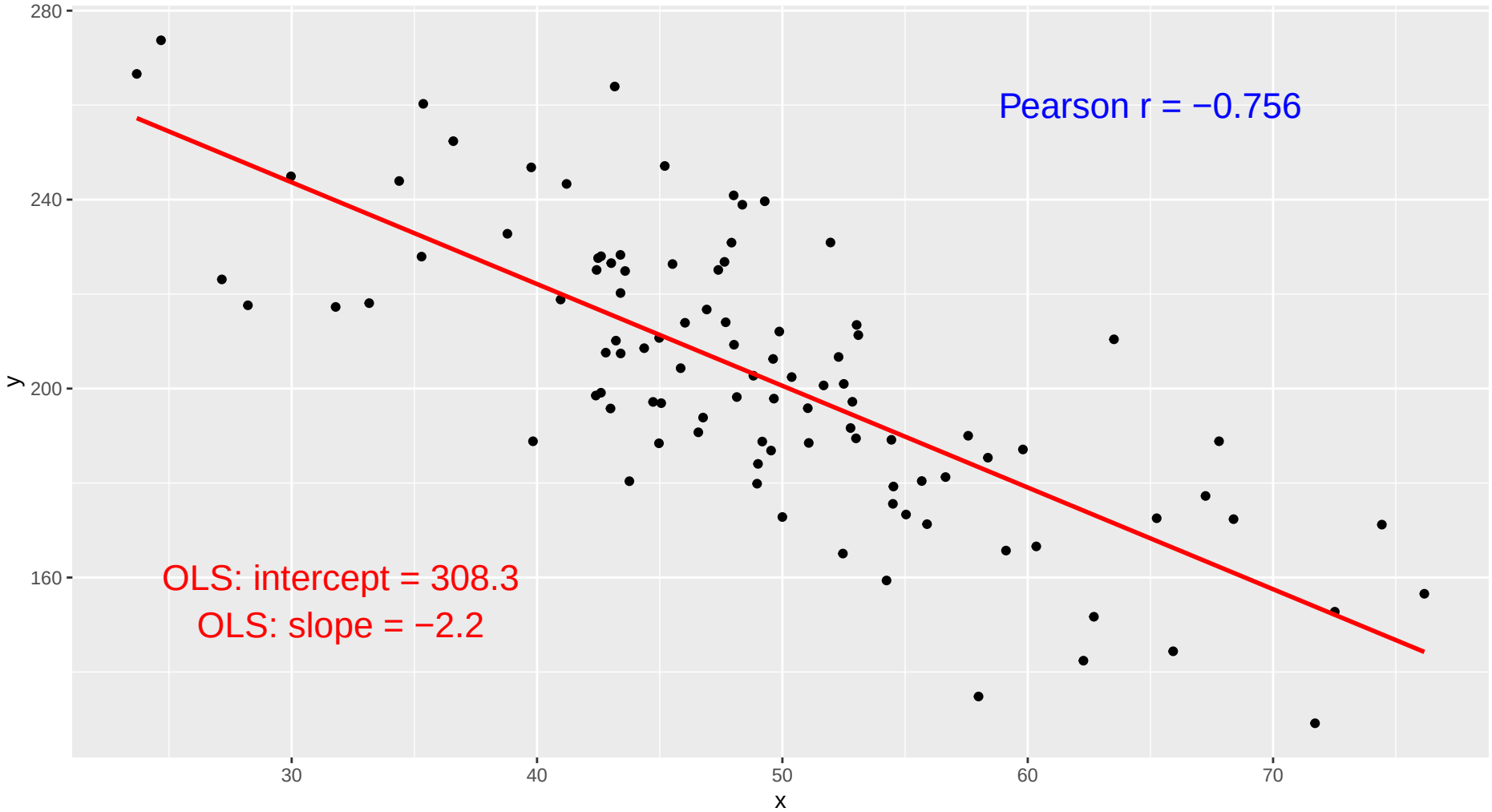
The Pearson correlation coefficient assesses how well the relationship between X and Y can be described using a linear function.

- The Pearson correlation is dimension-free.
- It falls between -1 and +1, with the extremes corresponding to situations where all the points in a scatterplot fall exactly on a straight line with negative and positive slopes, respectively.
- A Pearson correlation of zero corresponds to the situation where there is no linear association.
- Unlike the estimated slope in a regression line, the sample correlation coefficient is symmetric in x and y, so it does not depend on labeling one of them (y) the response variable, and one of them (x) the predictor.

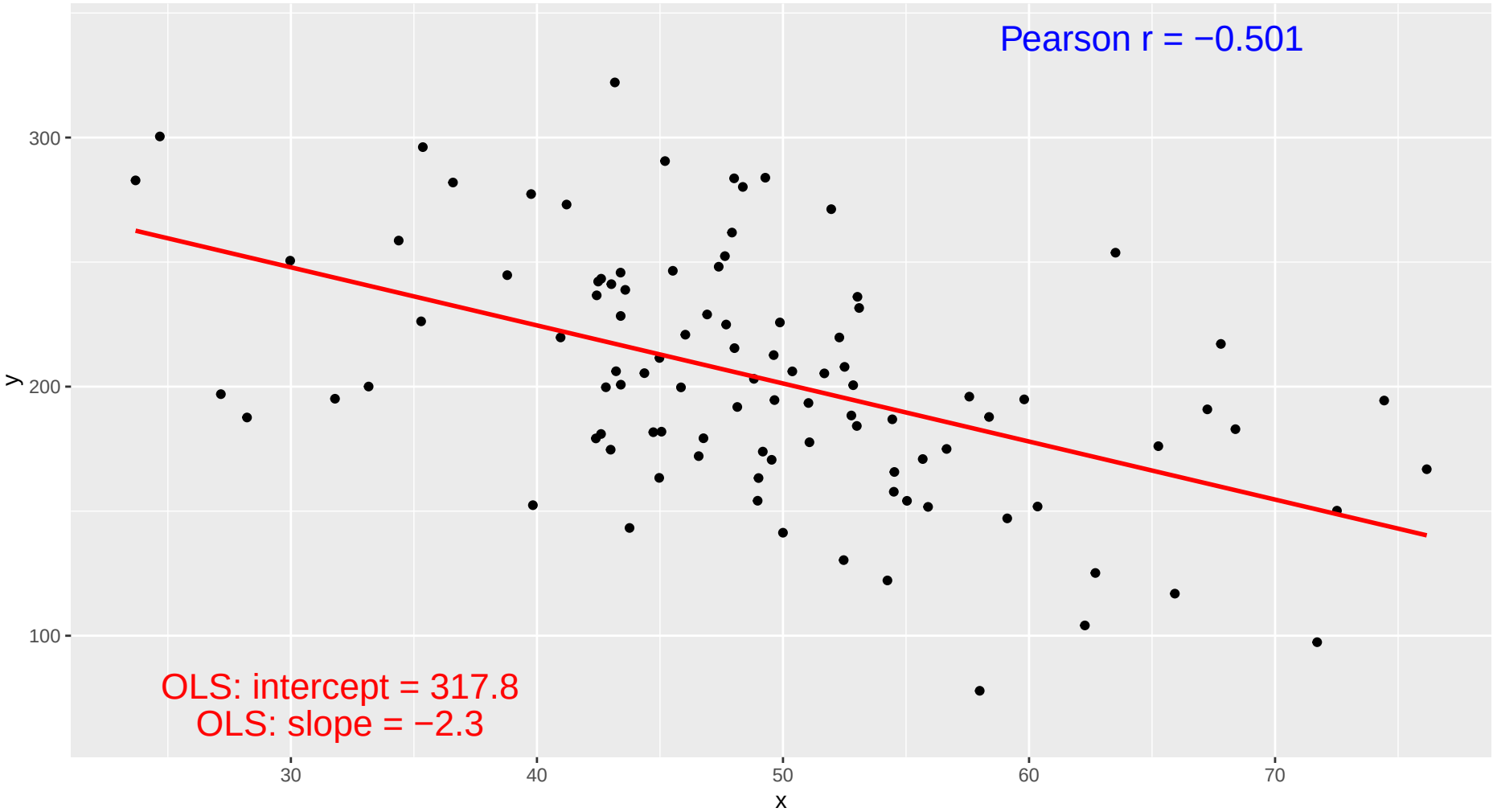
$$r_{XY} = \frac{1}{n-1} \Sigma_{i=1}^{n} \left(\frac{x_i - \bar{x}}{s_x} \right) \left(\frac{y_i - \bar{y}}{s_y} \right)$$

Simulated Example 1

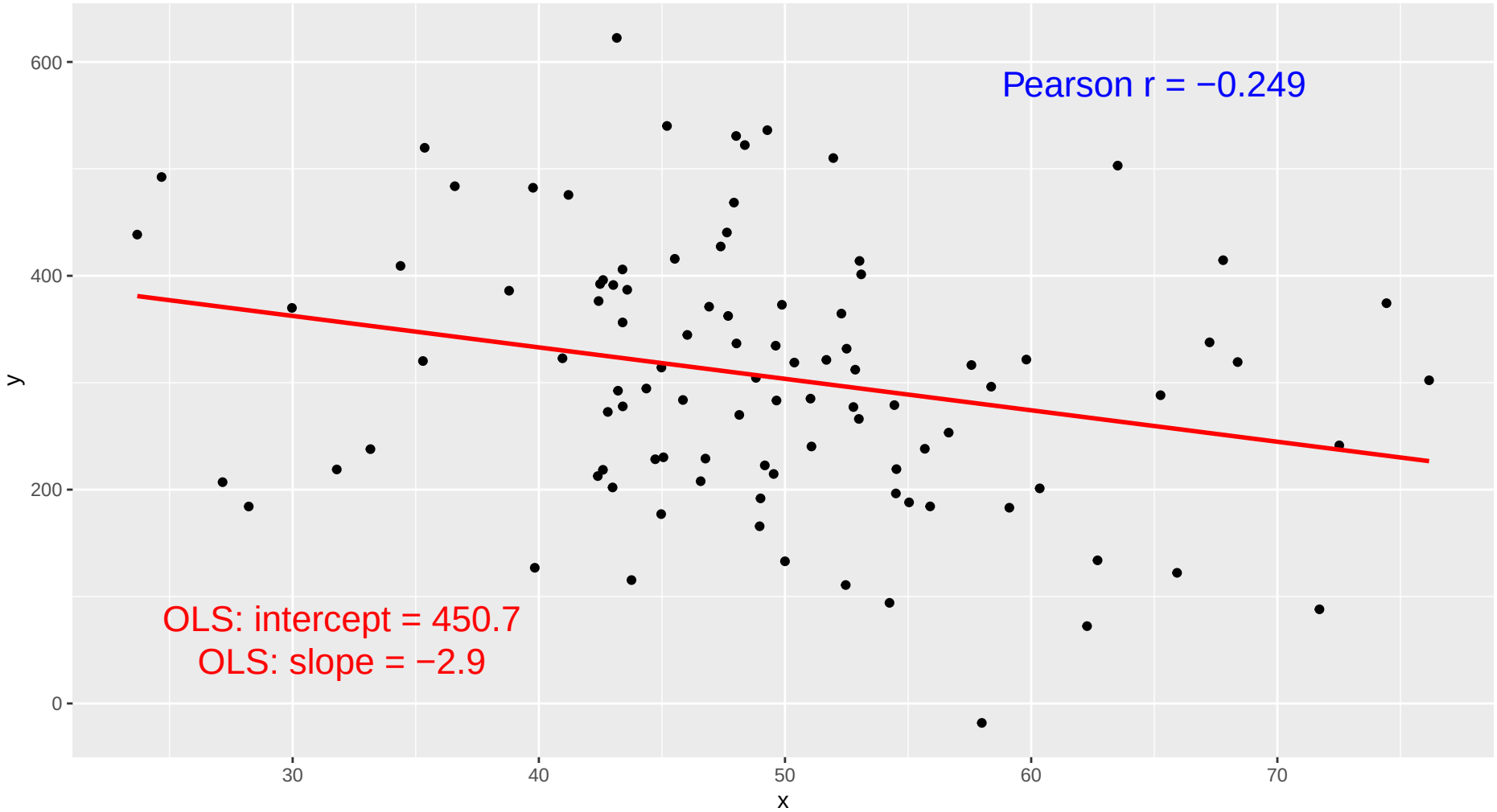
```
Warning: `data_frame()` is deprecated, use `tibble()`.
This warning is displayed once per session.
```



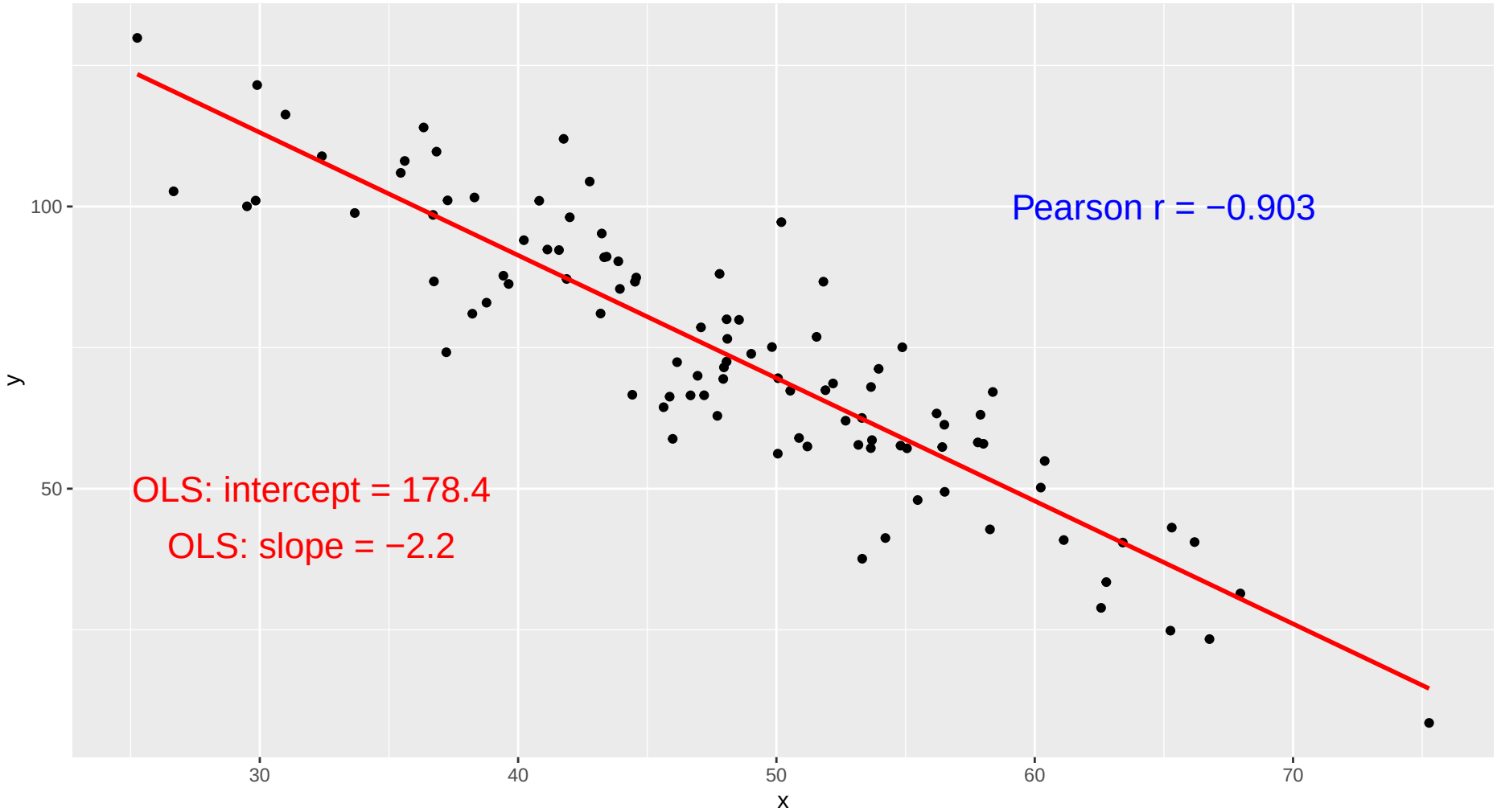
Simulated Example 2



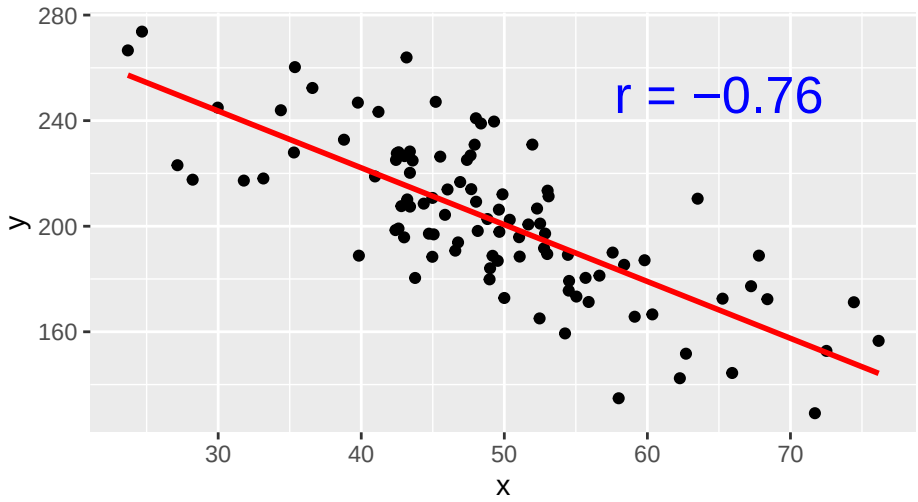
Simulated Example 3

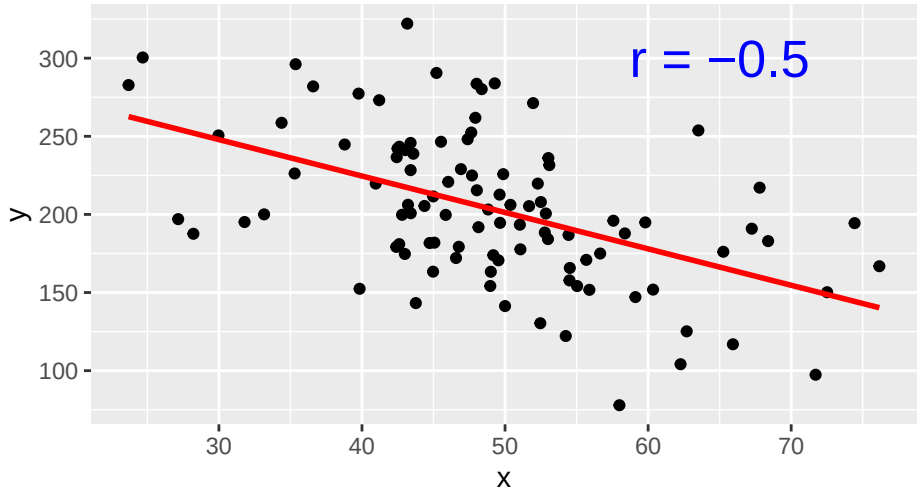


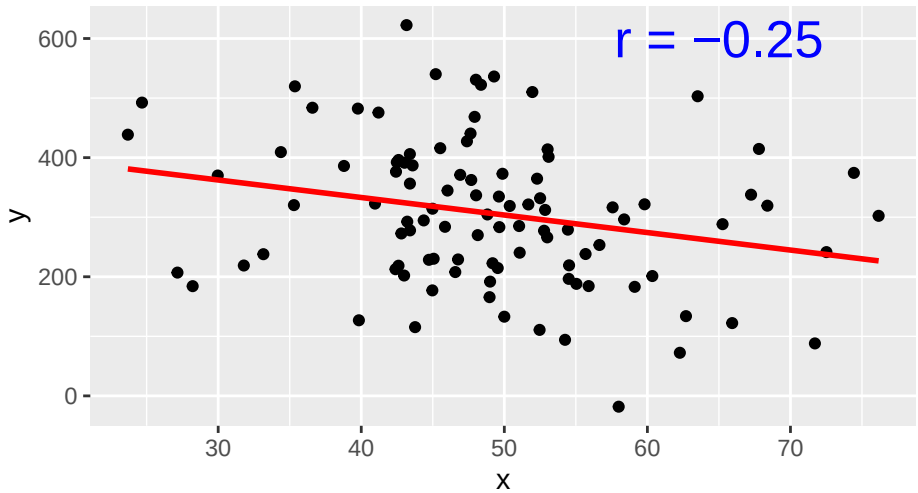
Simulated Example 4

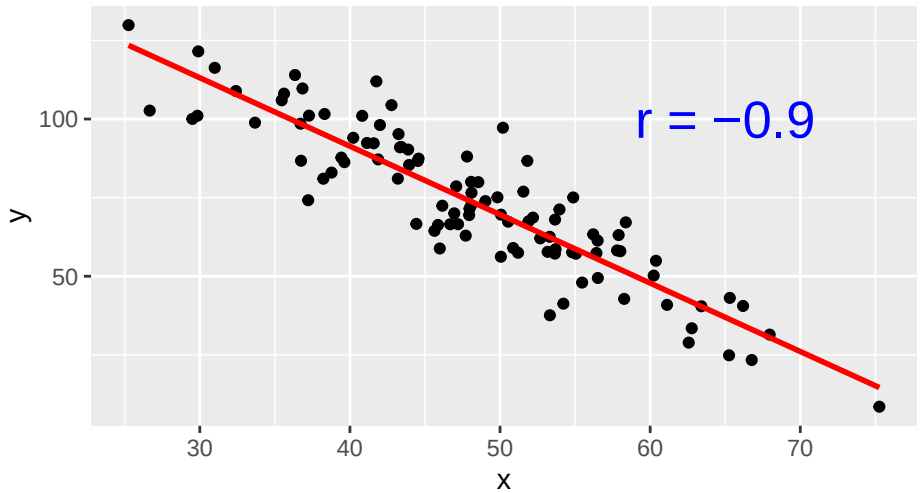


Calibrate Yourself on Correlation Coefficients

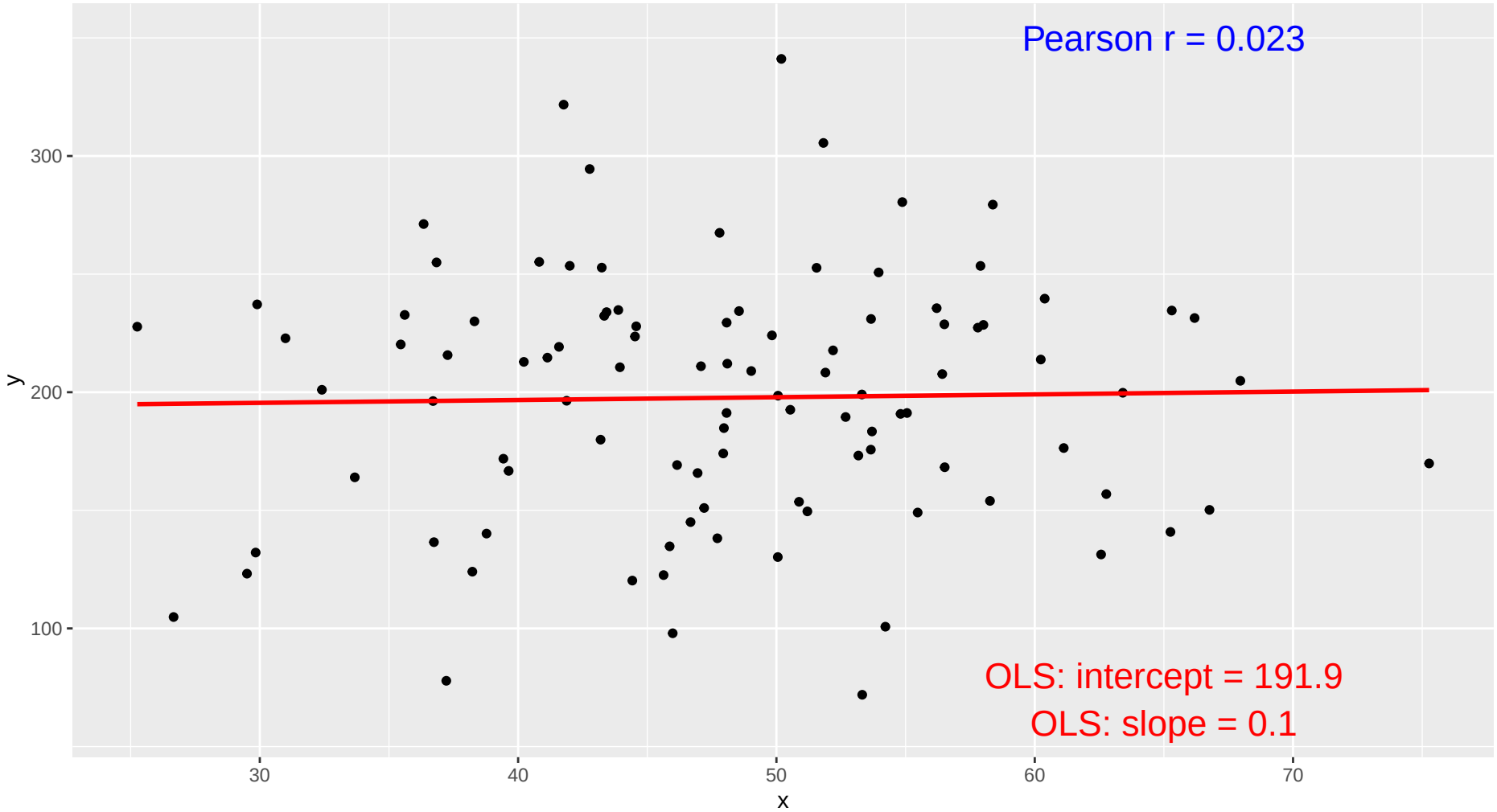




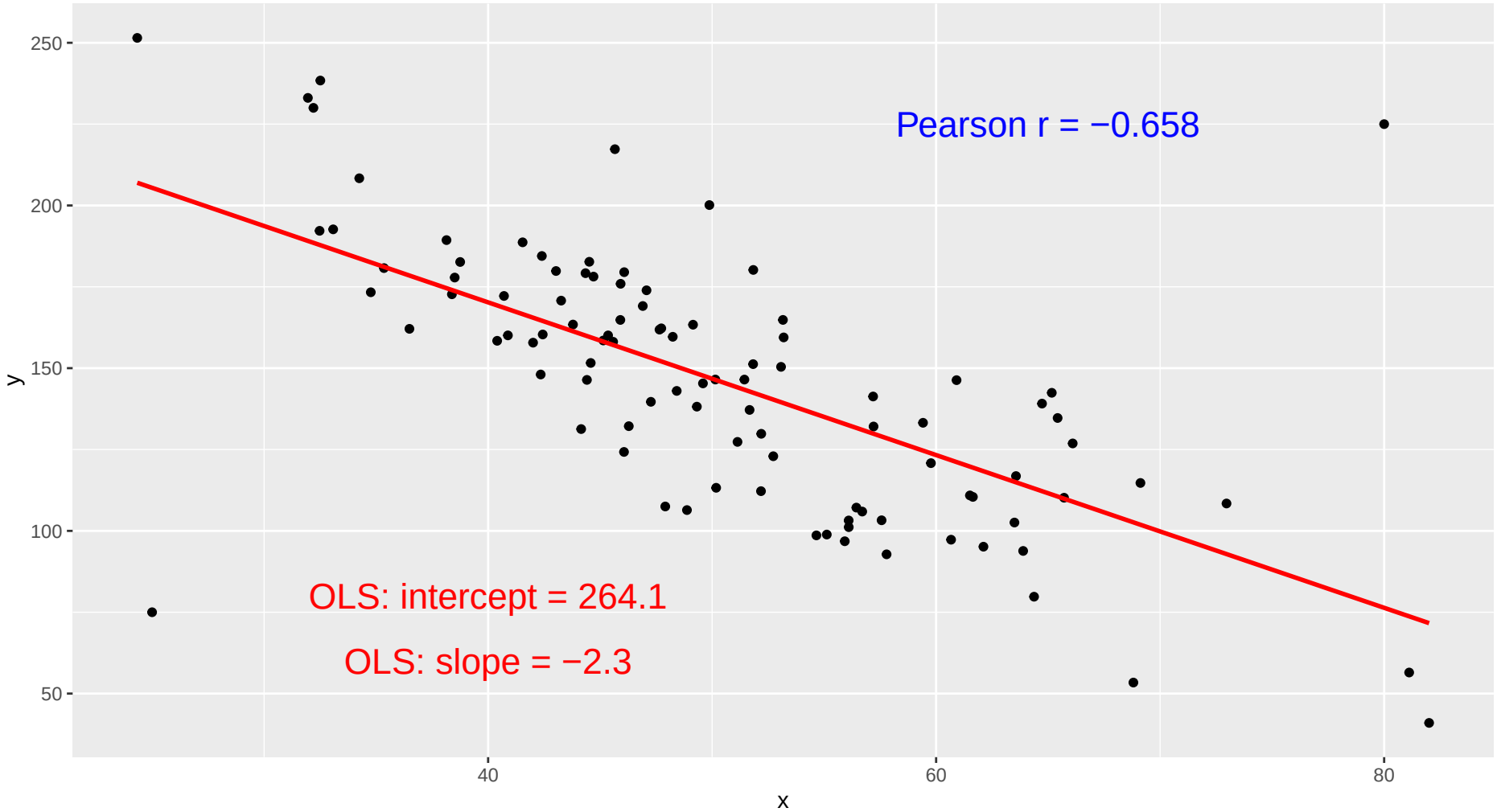




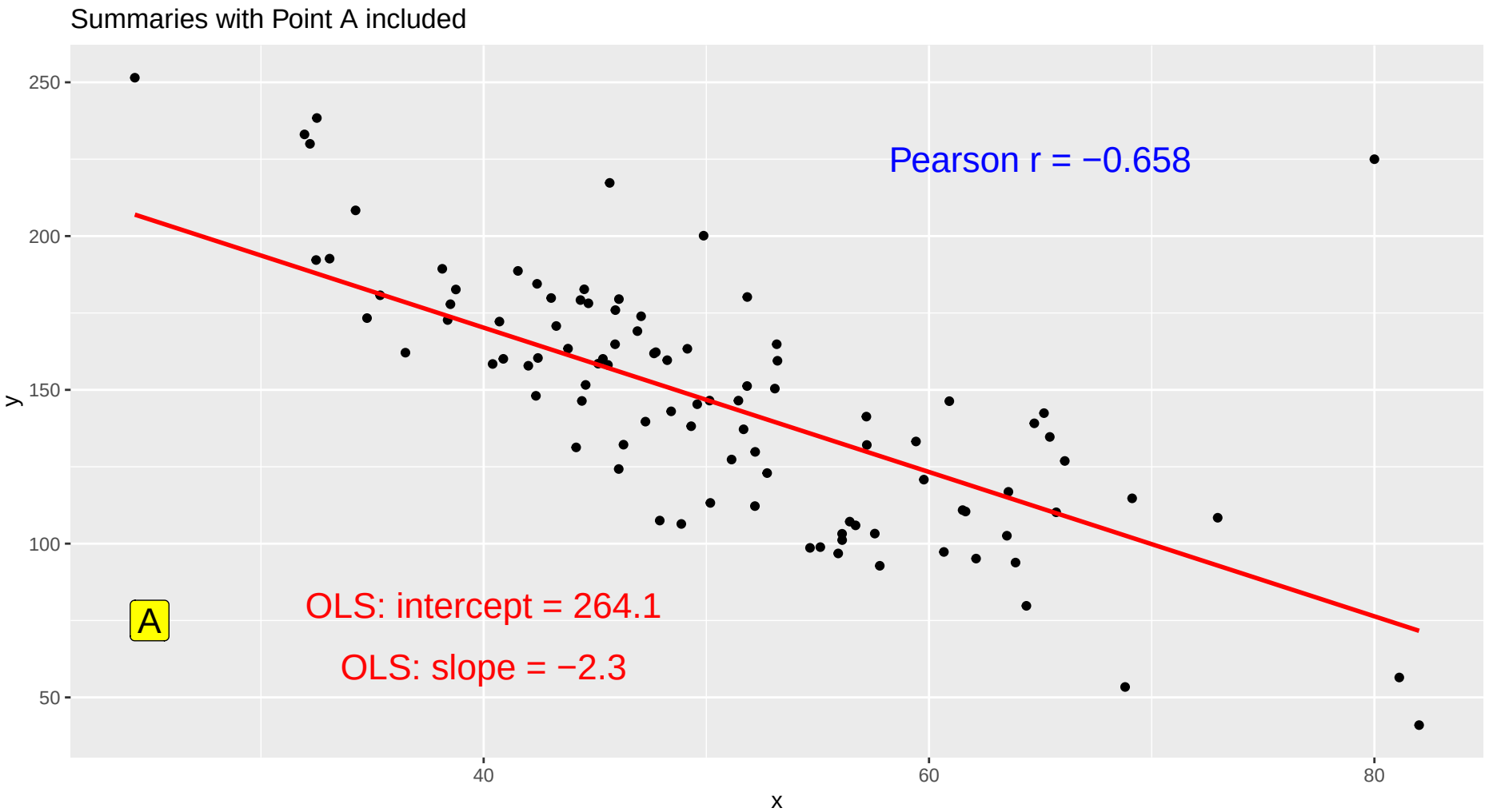
Simulated Example 5



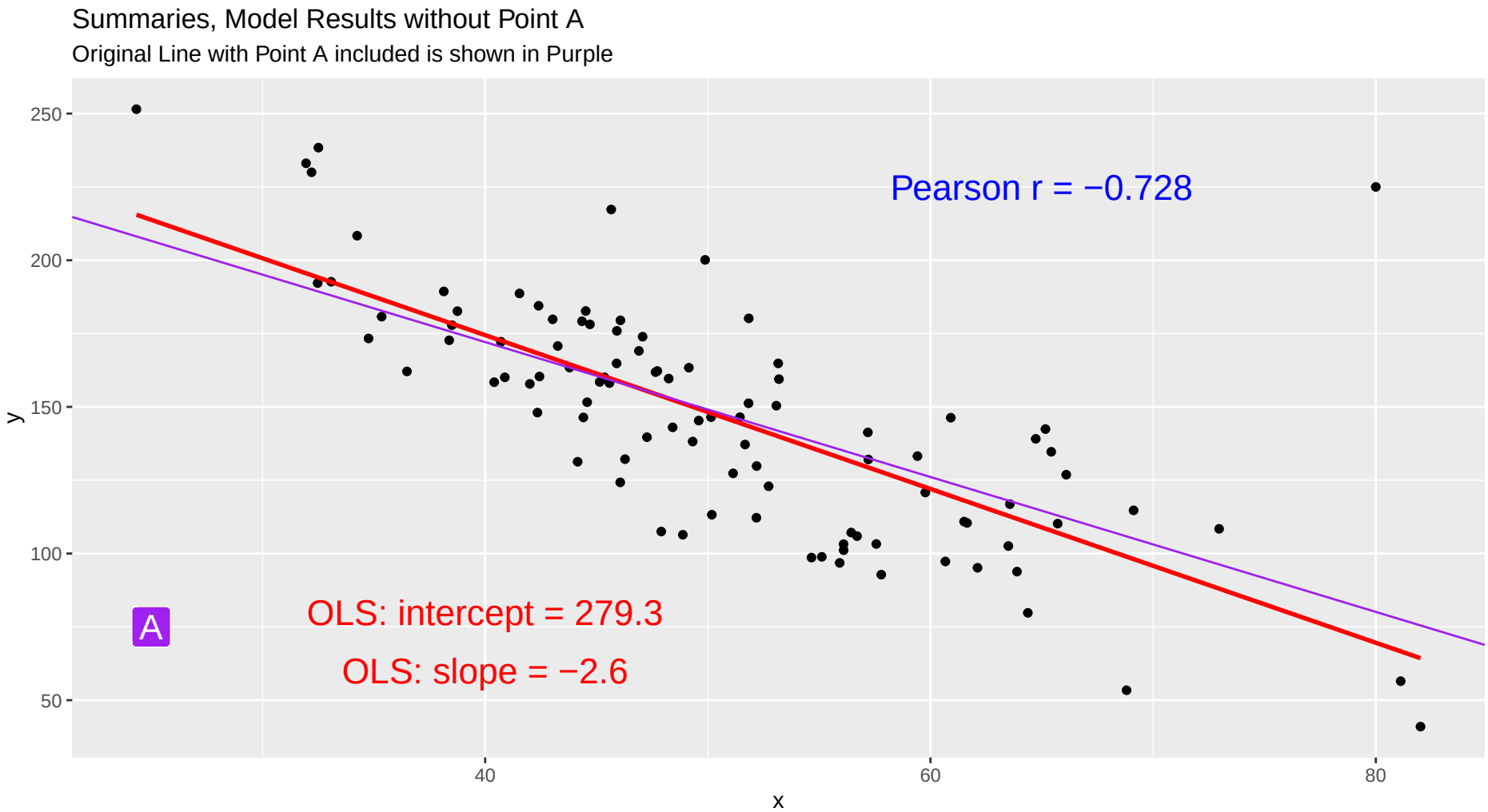
Simulated Example 6



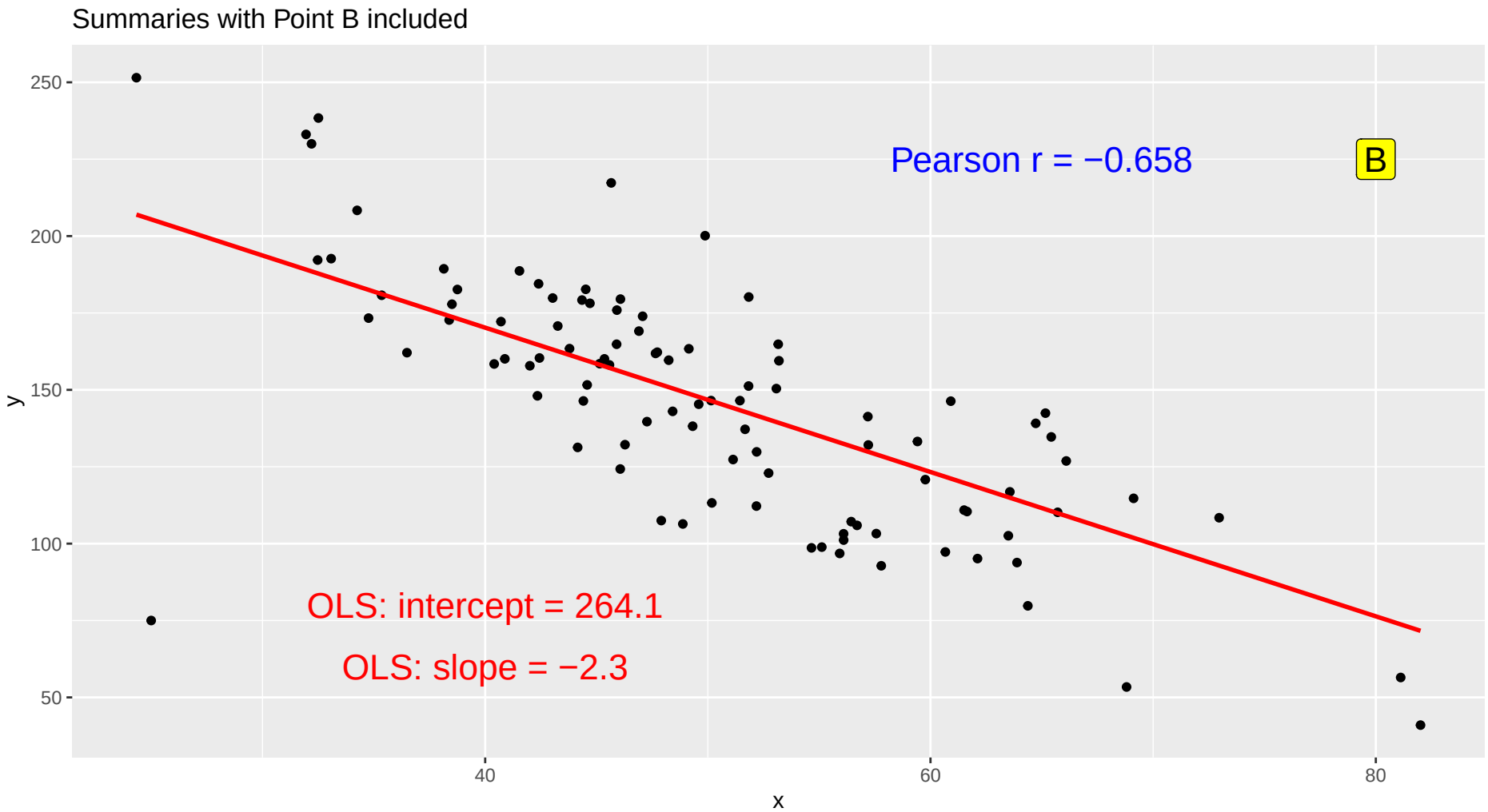
Example 6: What would happen if we omit Point A?



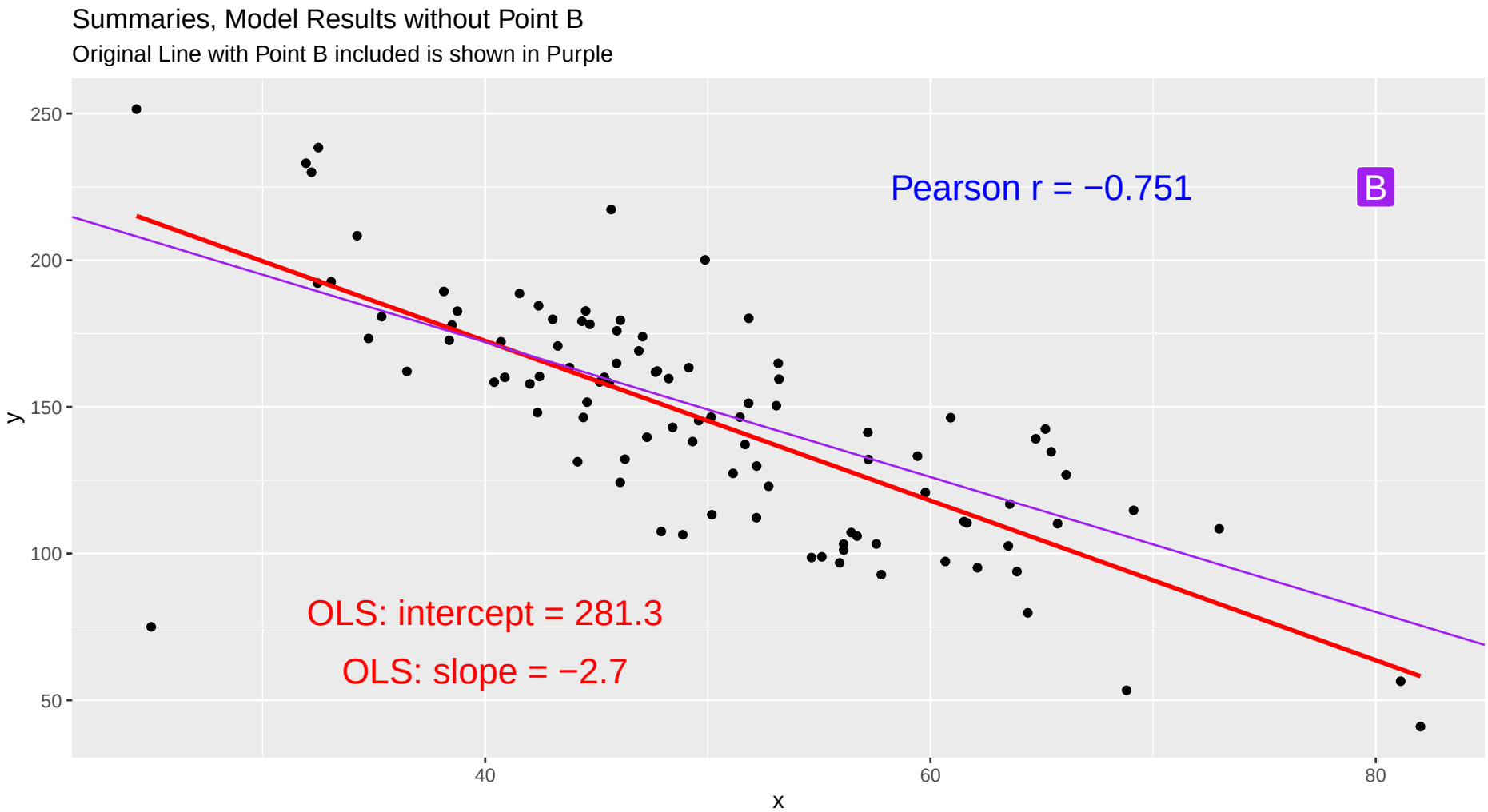
Example 6: Result if we omit Point A



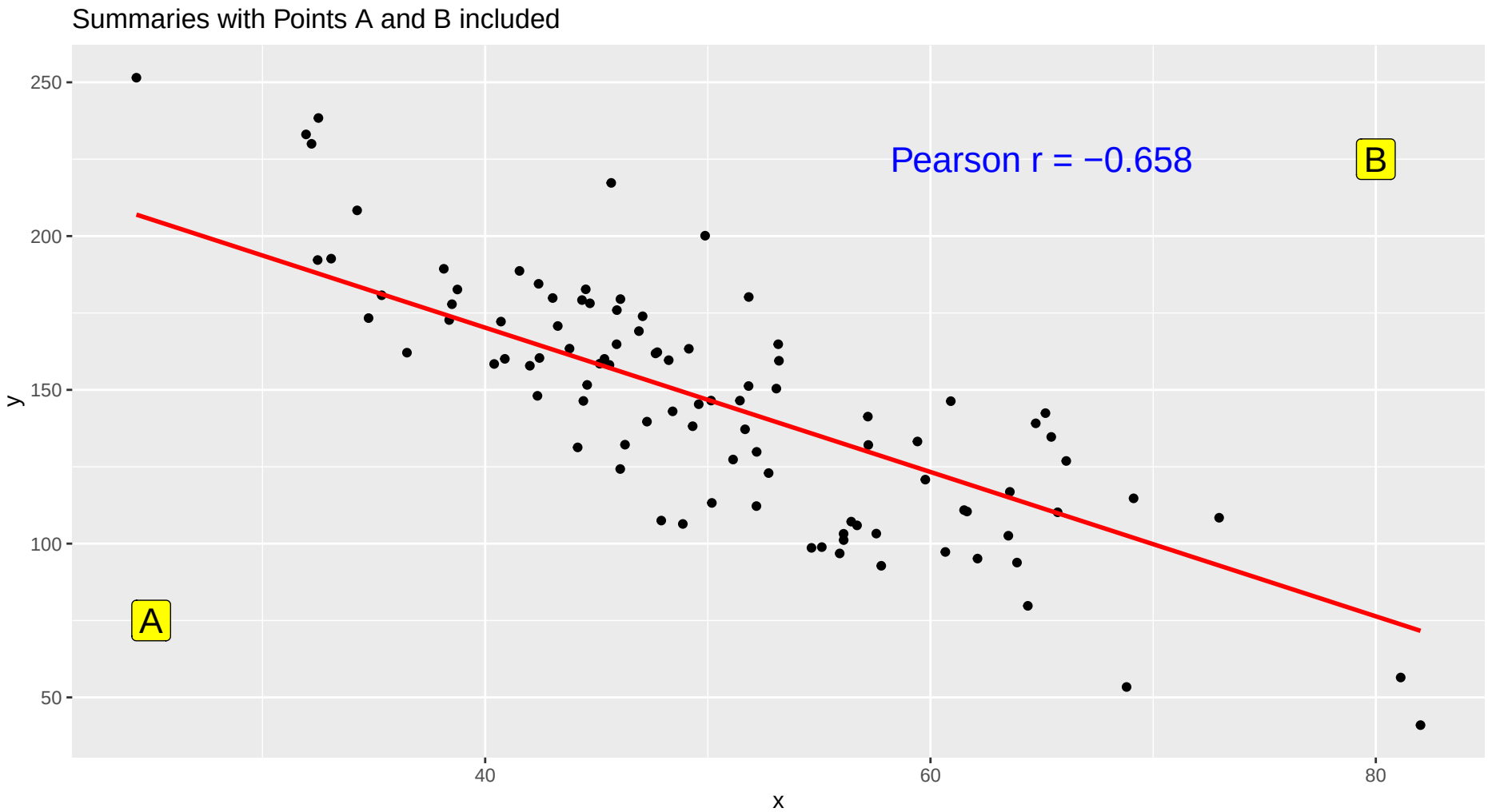
Example 6: What would happen if we omit Point B?



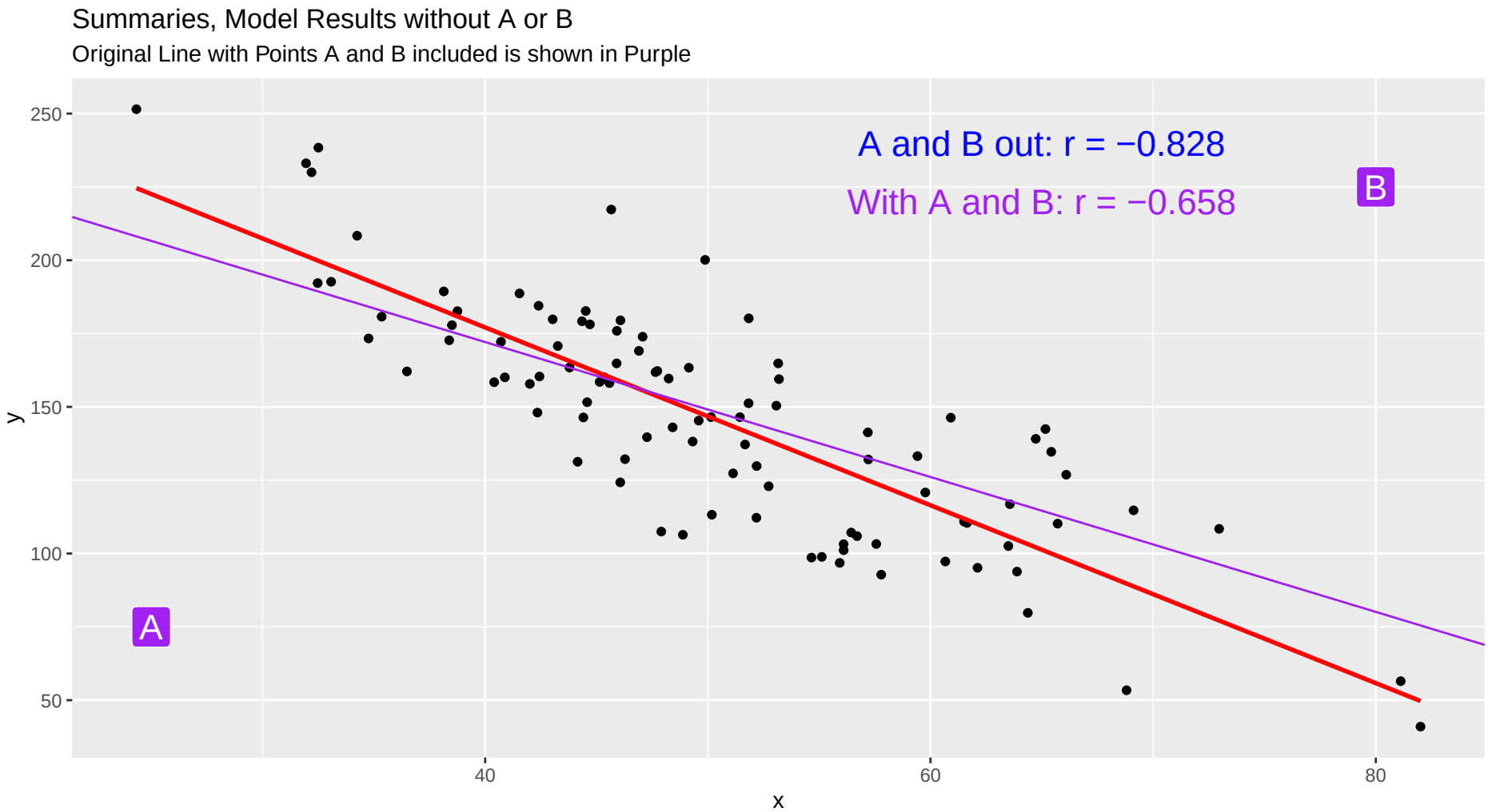
Example 6: Result if we omit Point B



Example 6: What if we omit Point A AND Point B?



Example 6: Result if we omit Points A and B

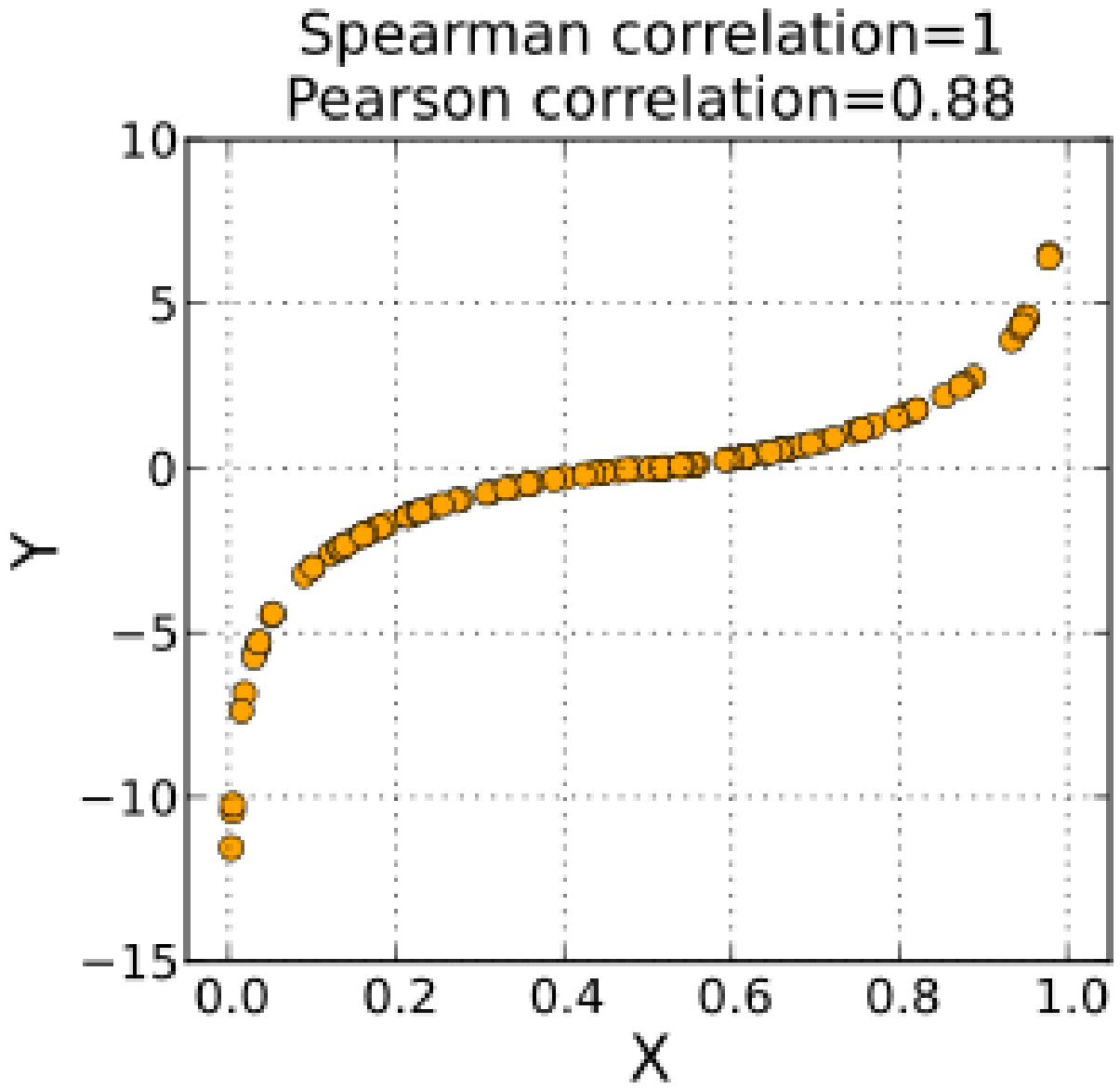


The Spearman Rank Correlation

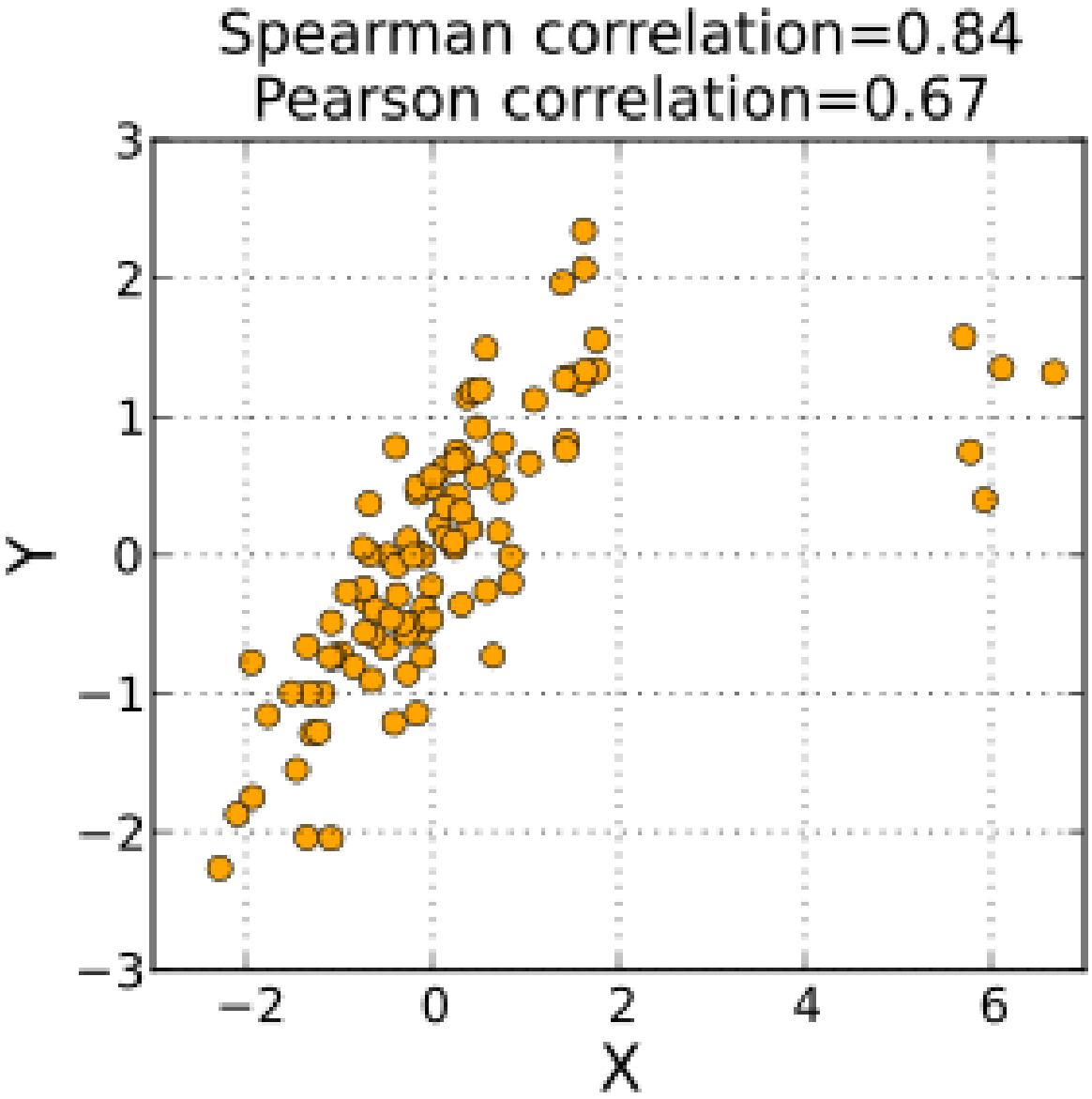
The Spearman rank correlation coefficient assesses how well the association between X and Y can be described using a **monotone function** even if that relationship is not linear.

- A monotone function preserves order - that is, Y must either be strictly increasing as X increases, or strictly decreasing as X increases.
- A Spearman correlation of 1.0 indicates simply that as X increases, Y always increases.
- Like the Pearson correlation, the Spearman correlation is dimension-free, and falls between -1 and $+1$.
- A positive Spearman correlation corresponds to an increasing (but not necessarily linear) association between X and Y, while a negative Spearman correlation corresponds to a decreasing (but again not necessarily linear) association.

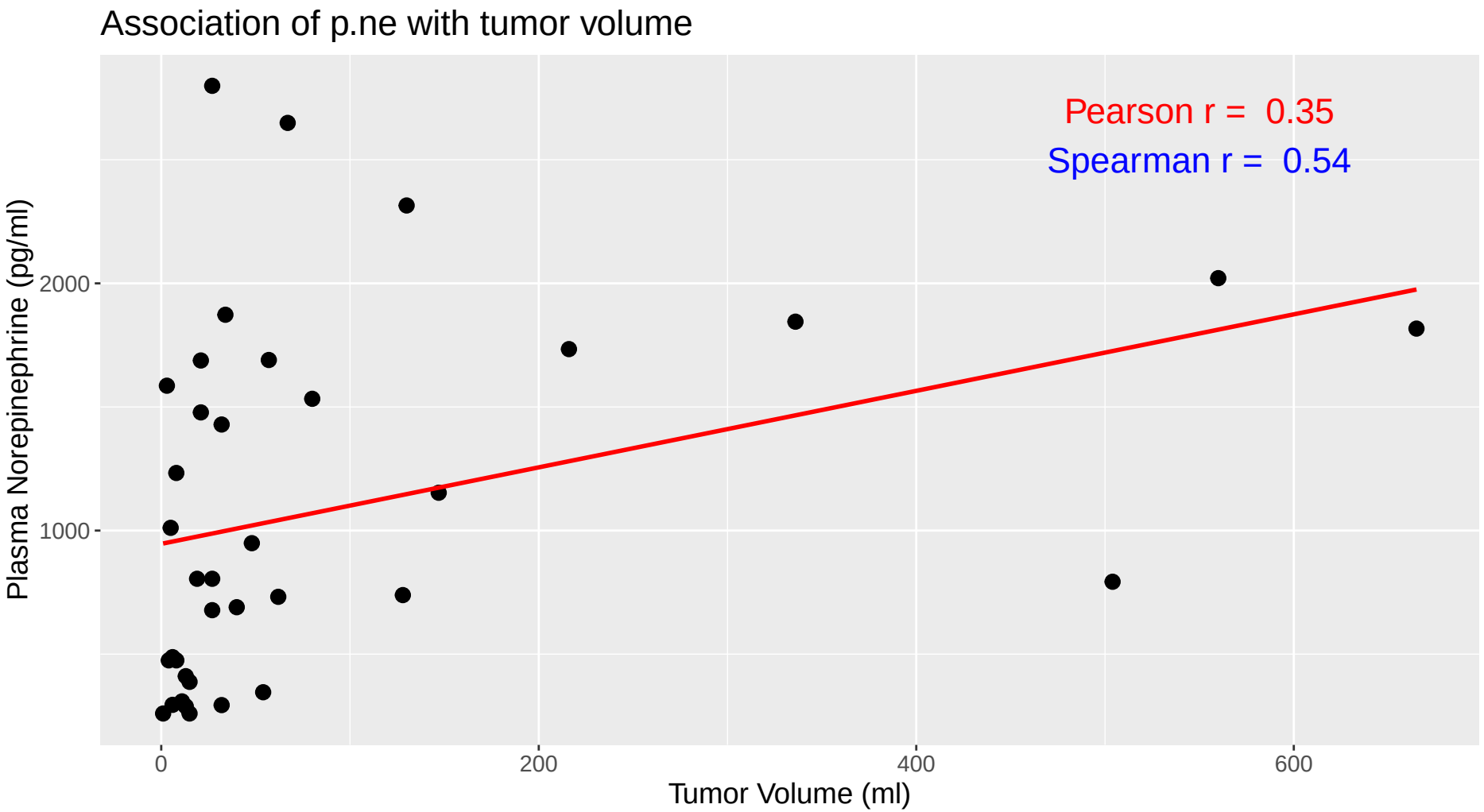
Monotone Association (Source: Wikipedia)



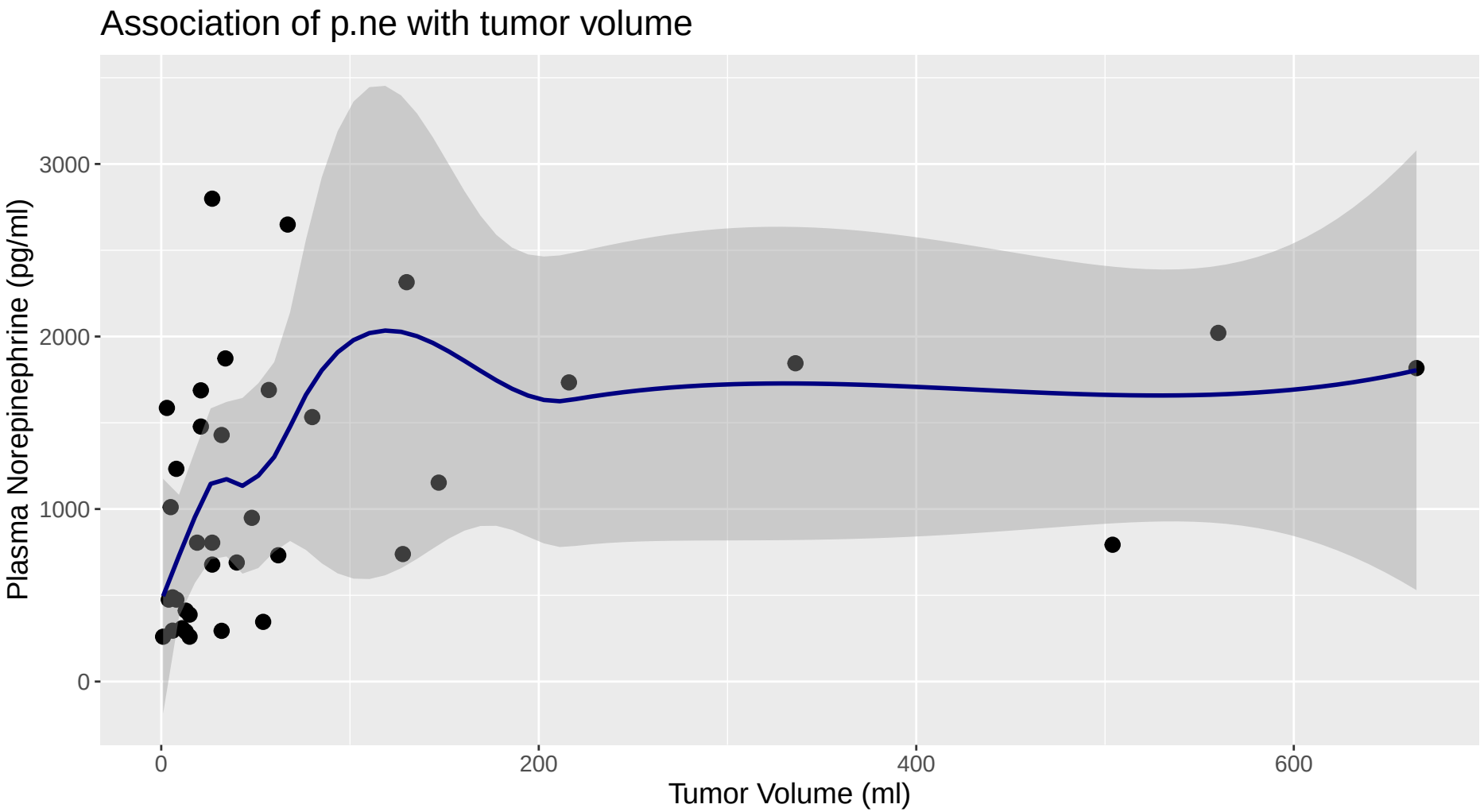
Spearman correlation reacts less to outliers



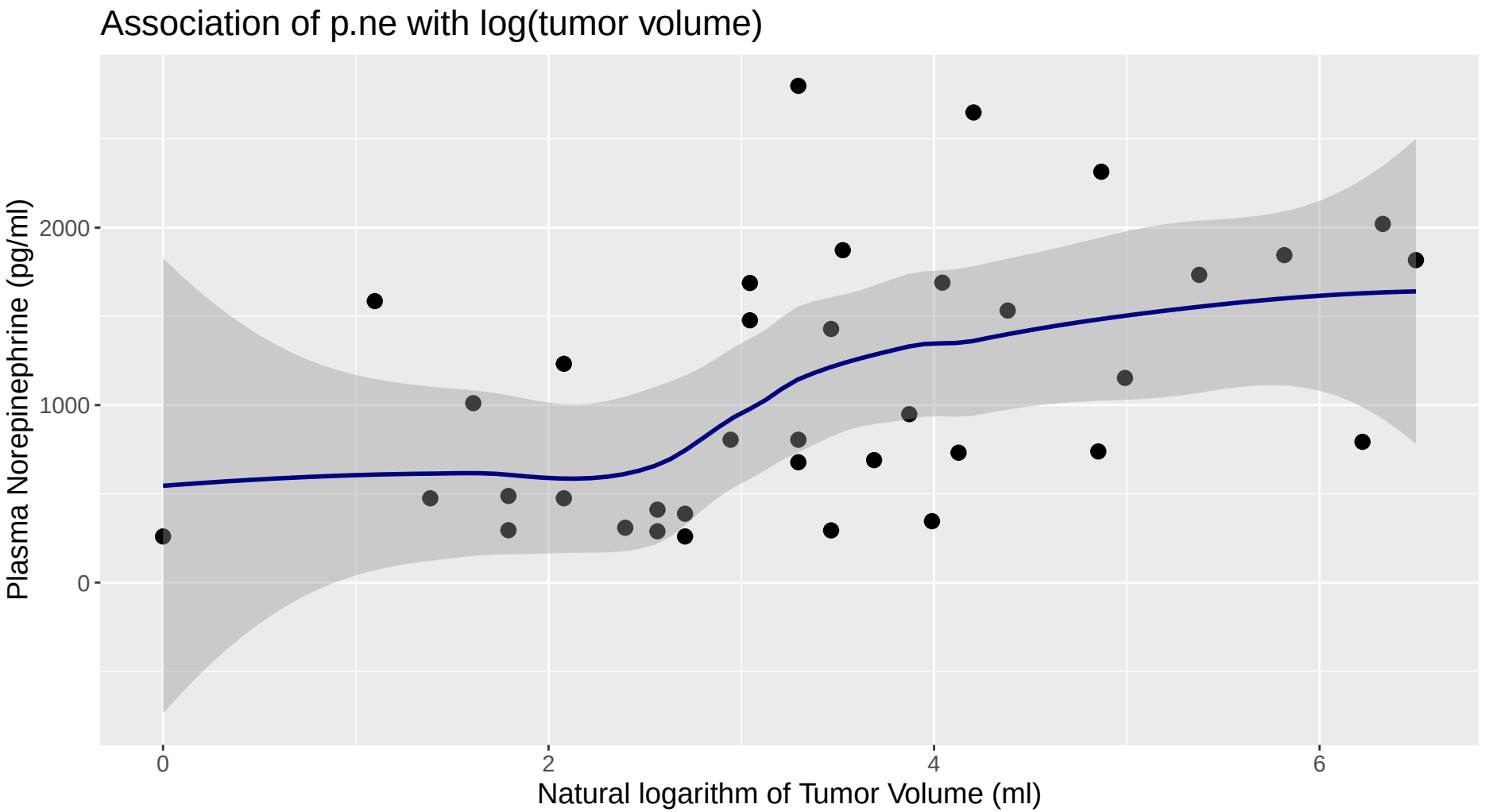
Our Key Scatterplot again



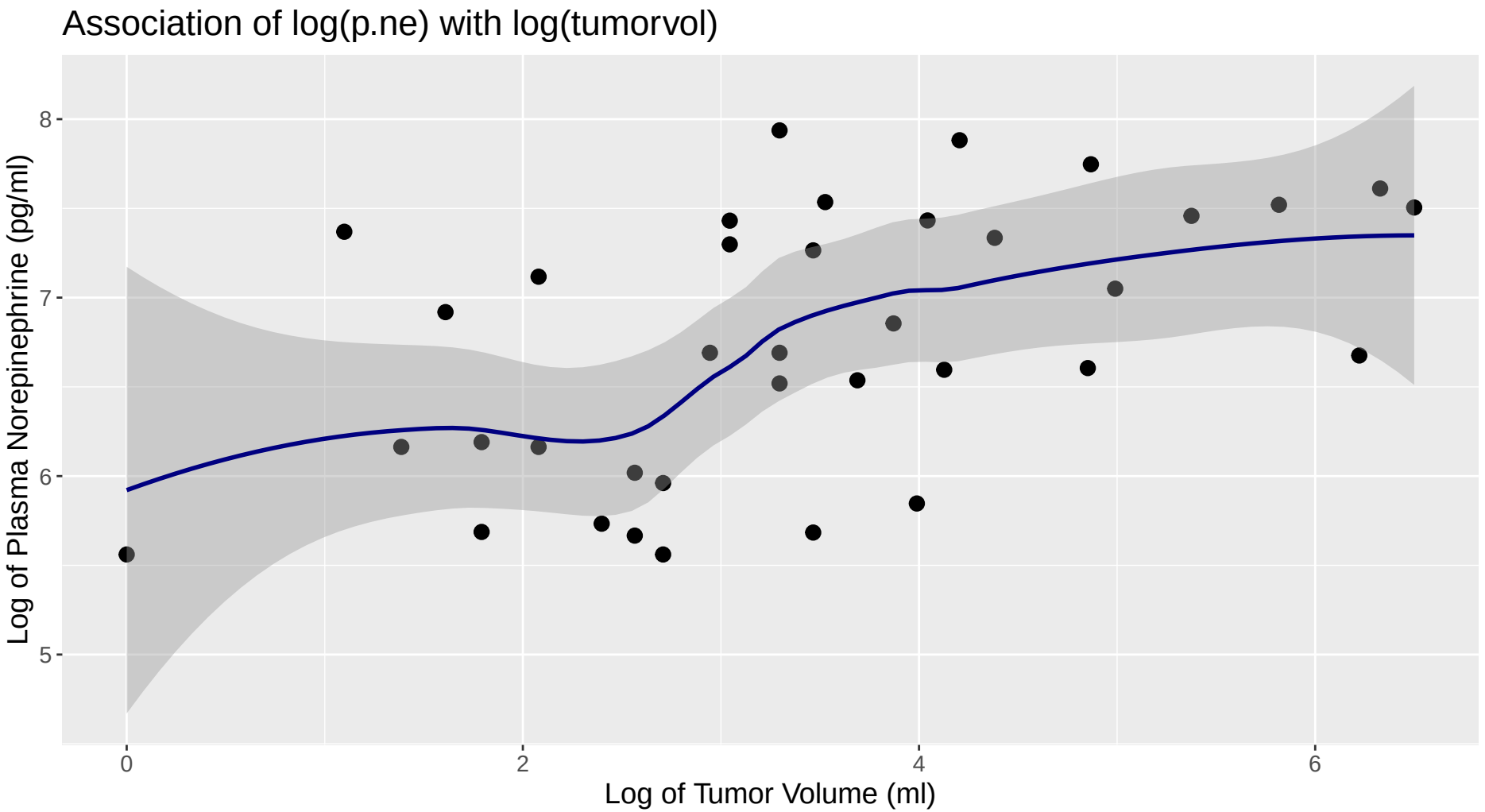
Smoothing using loess, instead



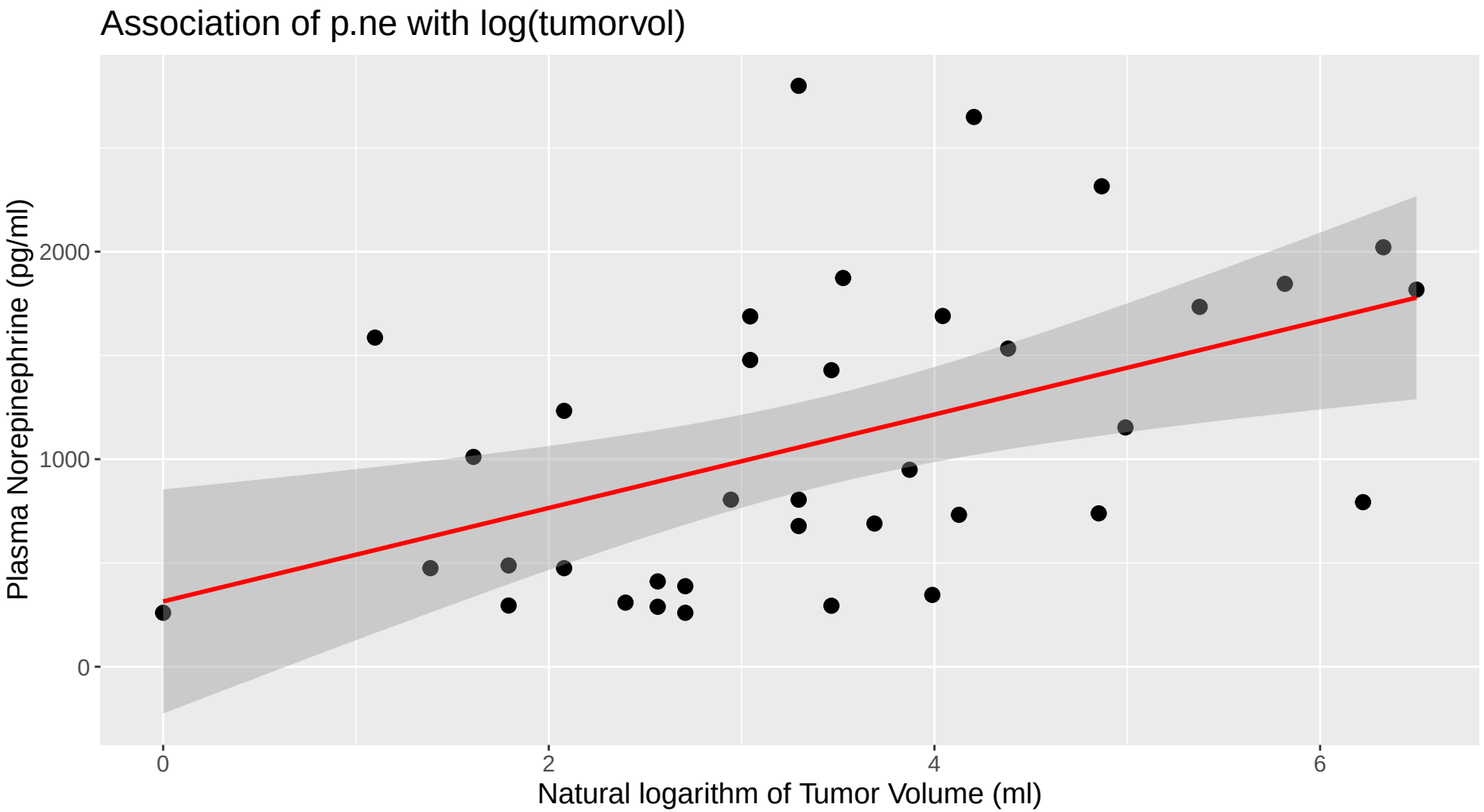
Using the Log transform to spread out the Volumes



Does a Log-Log model seem like a good choice?



Linear Model for p.ne using log(tumor volume)



Creating a Factor to represent disease diagnosis

We want to add a new variable, specifically a factor, called `diagnosis`, which will take the values `von H-L` or `neoplasia`.

- Recall `disease` is a numeric 1/0 variable (0 = von H-L, 1 = neoplasia)
- Use `fct_recode` from the `forcats` package. . .

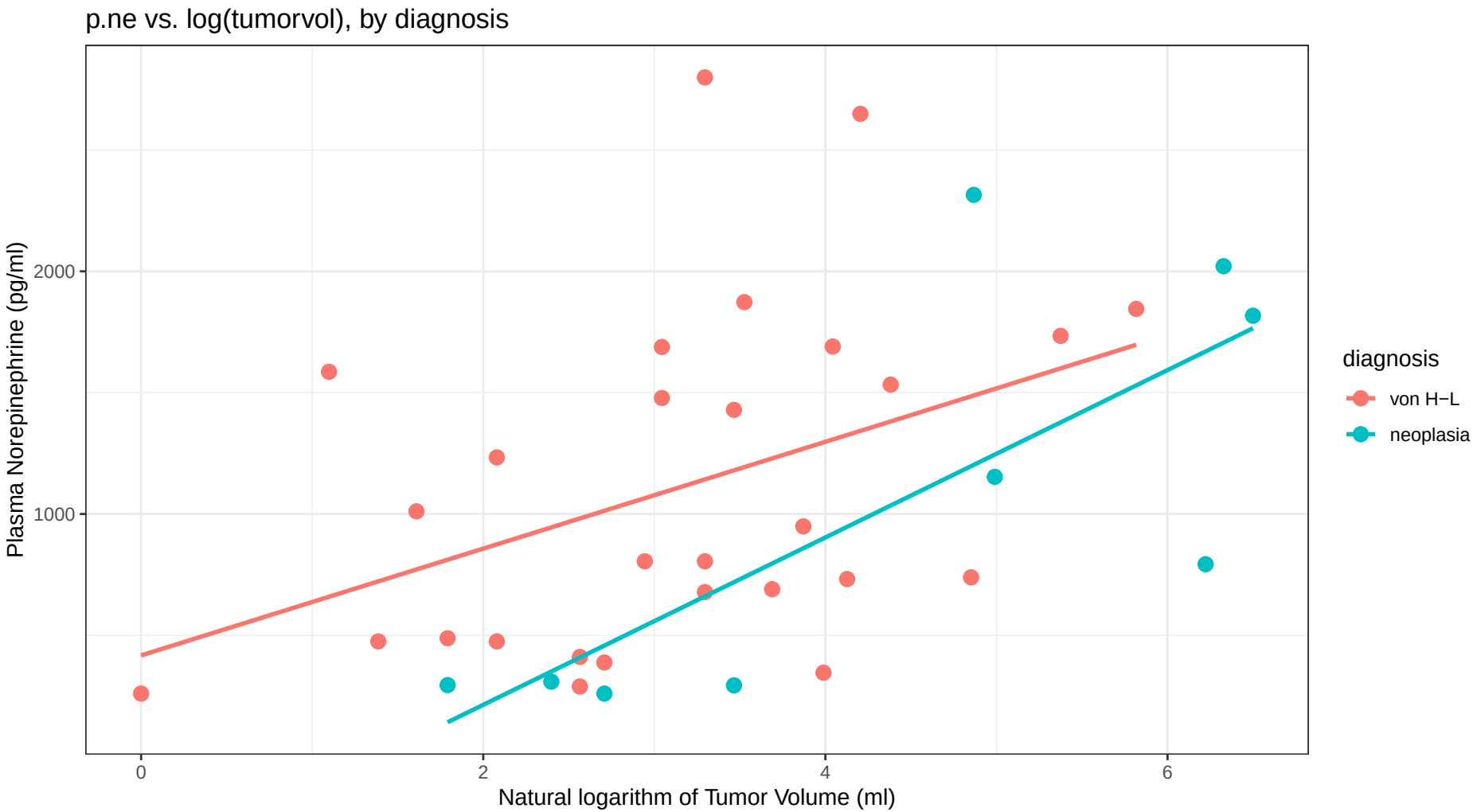
```
VHL <- VHL %>%
  mutate(diagnosis = fct_recode(factor(disease),
                                "neoplasia" = "1",
                                "von H-L" = "0")
  )
```

Now, what does VHL look like?

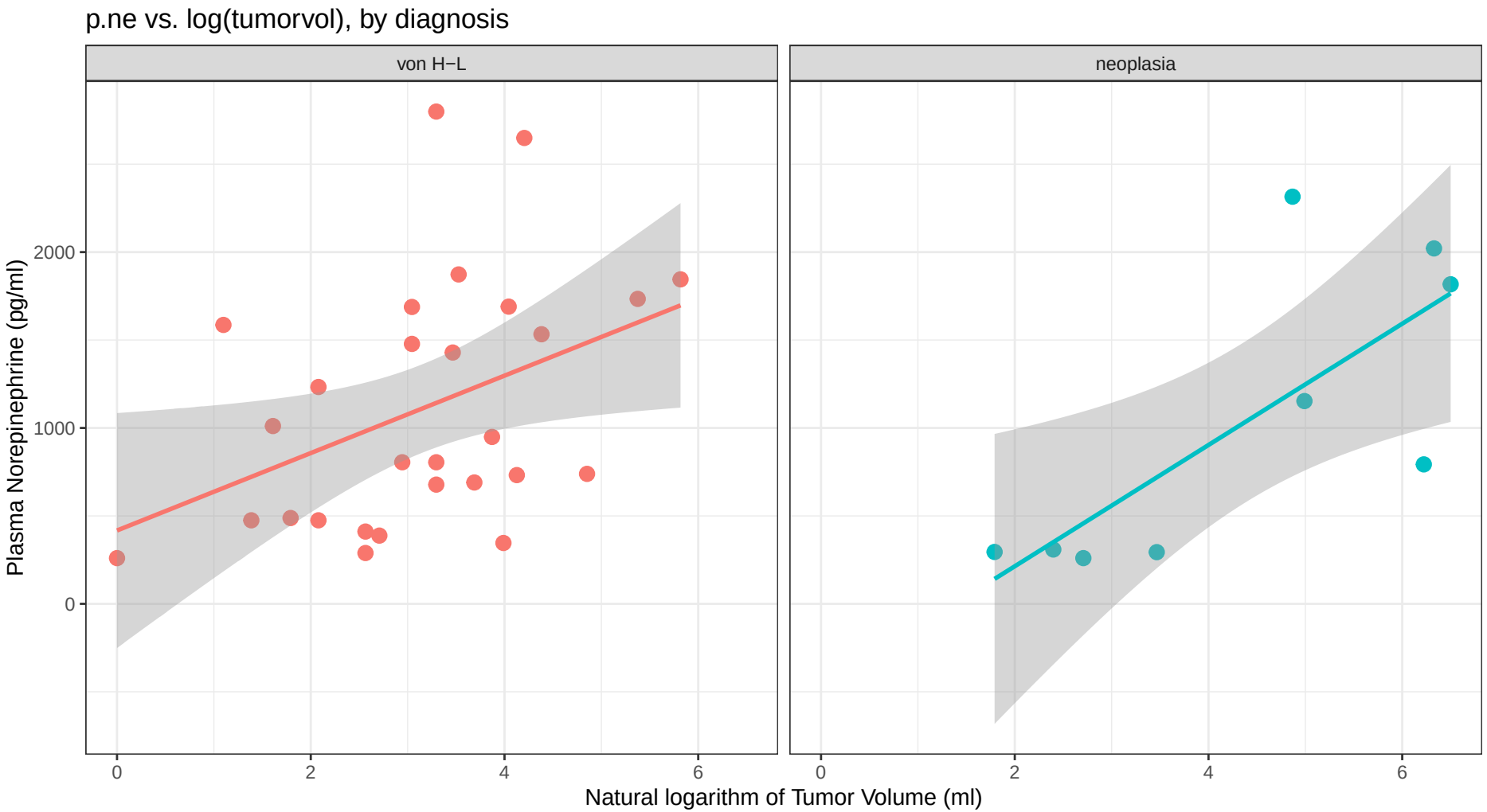
```
VHL
```

```
# A tibble: 37 x 5
      id disease  p.ne tumorvol diagnosis
   <dbl>   <dbl> <dbl>    <dbl> <fct>
 1   101       0   289       13 von H-L
 2   102       1   294       32 neoplasia
 3   103       0  2799       27 von H-L
 4   104       0  2649       67 von H-L
 5   105       0   346       54 von H-L
 6   106       0  1690       57 von H-L
 7   107       0   805       19 von H-L
 8   108       1  1153      147 neoplasia
 9   109       0   678       27 von H-L
10   110       1  1817      665 neoplasia
# ... with 27 more rows
```

Compare the patients by diagnosis



Facetted Scatterplots by diagnosis



Model accounting for different slopes and intercepts

```
model2 <- lm(p.ne ~ log(tumorvol) * diagnosis, data = VHL)
model2
```

```
Call:
lm(formula = p.ne ~ log(tumorvol) * diagnosis, data = VHL)

Coefficients:
                      (Intercept)
                            417.2
                    log(tumorvol)
                            220.0
               diagnosisneoplasia
                           -893.3
log(tumorvol):diagnosisneoplasia
                            124.8
```

Model 2 results

`p.ne` = 417 + 220 log(`tumorvol`) - 893 (`diagnosis = neoplasia`) + 125 (`diagnosis = neoplasia`)*log(`tumorvol`)

where the indicator variable (`diagnosis = neoplasia`) = 1 for neoplasia subjects, and 0 for other subjects. . .

- Model for `p.ne` in von H-L patients:
  - 417 + 220 log(`tumorvol`)
- Model for `p.ne` in neoplasia patients:
  - (417 - 893) + (220 + 125) log(`tumorvol`)
  - -476 + 345 log(`tumorvol`)

Model 2 Predictions

What is the predicted `p.ne` for a single new subject with `tumorvol` = 200 ml (so log(tumorvol) = 5.3) in each diagnosis category?

```
predict(model2, newdata = data_frame(tumorvol = 200,
        diagnosis = "neoplasia"), interval = "prediction")
```

```
       fit      lwr     upr
1 1350.896 -28.0571 2729.85
```

```
predict(model2, newdata = data_frame(tumorvol = 200,
        diagnosis = "von H-L"), interval = "prediction")
```

```
       fit      lwr      upr
1 1583.079 208.6489 2957.509
```

Tidying the Model 2 coefficients, with broom

```
broom::tidy(model2)
```

```
# A tibble: 4 x 5
  term                 estimate std.error statistic p.value
  <chr>                   <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)              417.     318.       1.31   0.199 
2 log(tumorvol)            220.      93.6      2.35   0.0248
3 diagnosisneopla~        -893.     659.      -1.36   0.184 
4 log(tumorvol):d~         125.     155.       0.807  0.425 
```

Model 2, summarized at a glance, with broom

```
broom::glance(model2)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic p.value    df
      <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>
1     0.290         0.226  634.      4.50 0.00937     4
# ... with 5 more variables: logLik <dbl>, AIC <dbl>,
#   BIC <dbl>, deviance <dbl>, df.residual <int>
```

Compare this to model 1...

```
broom::glance(model1)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic p.value    df
      <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>
1     0.120        0.0950  685.      4.78  0.0356     2
# ... with 5 more variables: logLik <dbl>, AIC <dbl>,
#   BIC <dbl>, deviance <dbl>, df.residual <int>
```

Conclusions about VHL data

- The second model, accounting for the interaction of diagnosis with the log of tumor volume, was able to account for about 29% of the variation in the plasma norepinephrine levels.
- Model 1, our original linear model, which didn't account for diagnosis at all, showed that tumor volume accounted for about 12% of the variation we observed in plasma norepinephrine levels.

Can we draw a lot more from this yet?

So what did we hear about today?

- The central role of linear regression in understanding associations between quantitative variables.
- The interpretation of a regression model as a prediction model.
- The meaning of key regression summaries, including residuals.
- Using tidy and glance from the broom package to help with summaries.
- Measuring association through correlation coefficients.
- How we might think about "adjusting" for the effect of a categorical predictor on a relationship between two quantitative ones.
- How a transformation might help us "linearize" the relationship shown in a scatterplot.