Is it time to de-normalize the normalization procedures based on ratio variables? Preprint

Version: v.1.0 **; Date:** 10 June 2024

This study has not been peer reviewed

Rubén Fernández-Matías, PT, PhD Student[1,2,3]

Affiliations:

[1] Research Unit, Hospital Universitario Fundación Alcorcón, Madrid, Spain.

[2] Physical Therapy and Rehabilitation Unit, Hospital Universitario Fundación Alcorcón, Madrid, Spain

[3] Doctoral School, Department of Physical Therapy, Universitat de València, Valencia, Spain

Correspondence:

Address: Physical Therapy and Rehabilitation Unit, Hospital Universitario Fundación Alcorcón, Street Budapest 1, Postal Code 28922, Alcorcón, Madrid, Spain.

Email: ruben.fernanmat@gmail.com

Statement of financial disclosure and conflict of interest: I affirm that I have no financial affiliation (including research funding) or involvement with any commercial organization that has a direct financial interest in any matter included in this manuscript, except as disclosed and cited in the manuscript. Any other conflict of interest (i.e., personal associations or involvement as a director, officer, or expert witness) is also disclosed and cited in the manuscript.

ABSTRACT

Ratio variables are created by dividing one variable by another variable. This procedure is usually done with the aim of “normalizing” for the denominator variable, to obtain associations adjusted for that confounding variable, or to calculate ranges of normative values. Despite its widespread use, ratio variables bring with them many statistical problems. In this editorial, the reader will be introduced to the statistical assumptions of using ratio variables, the problems that arise when those assumptions are not met, and how to use proper regression models to achieve the desired objectives without the need to use ratio variables.

Keywords: ratio; normalize; normative values; regression model

INTRODUCTION

Ratio variables, which are formed by dividing one variable by another, are widely used within physical therapy and sports science research fields, and clinical practice. The rationale behind the use of ratio variables is to "normalize" for the denominator. Some examples are the scapulohumeral rhythm,[3] the between-limb normative ratio values[4] (i.e., limb strength asymmetry as a risk factor for sport injuries), the percent change of muscle thickness,[2] the percent change from baseline in clinical trials, the normalization procedures of electromyographic signals, or the Y-balance test (i.e., distance reached divided by lower limb length).

However, despite their widespread use, ratio variables bring with them many statistical problems, which is why several authors have already expressed their concern and warned about the inadequacy of their use.[1,7,9]

In this study, the reader will be introduced to the statistical assumptions of using ratio variables, and the problems that arise when those assumptions are not met.

Statistical analysis

All the statistical analysis were conducted using R software v.4.1.0 (R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/). The R code for all the simulations and figures is presented in Supplementary Material.

Statistical assumptions on the use of ratio variables

When we use ratio variables within regression models, we are making four assumptions (**FIGURE 1**):[2]

1) There is a positive linear relationship between de numerator and the denominator.
2) There is heteroskedasticity within that association, with a pattern of increased error variance as the denominator value increases.
3) The intercept of the regression model predicting the numerator by the denominator is equal to zero.
4) The ratio variable is not related in any way to the denominator.

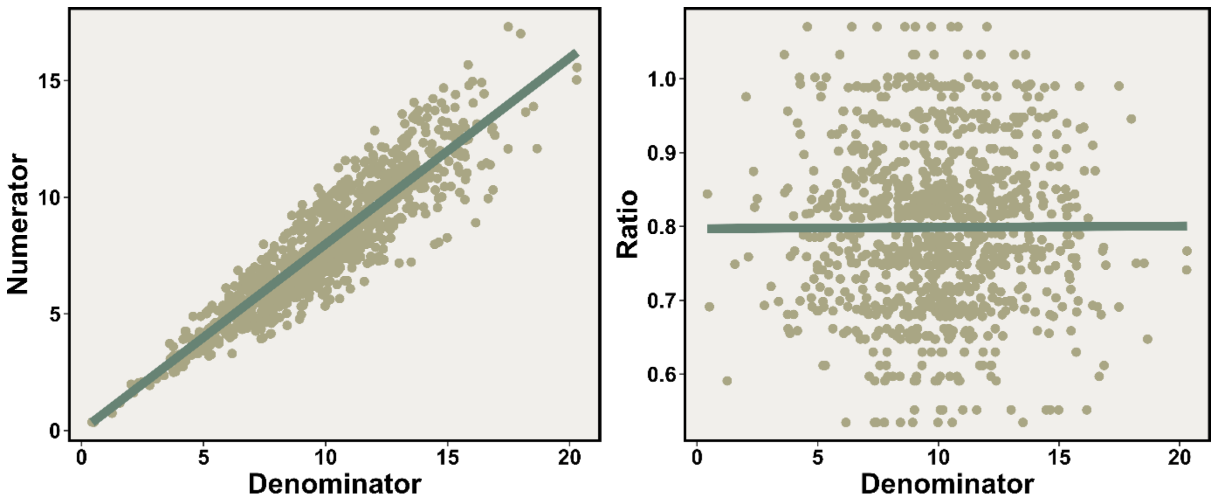

Figure 1. Scatterplot for the statistical assumptions on the use of ratio variables. Left figure shows first to third assumptions, and right figure shows fourth assumption.

However, even when the statistical assumptions appear to be met in the scatter plots, the use of ratio variables may still be inappropriate, as it is showed elsewhere.[2]

Wrong predictions and normative values

Two of the objectives of using ratio variables are to predict the numerator as a function of the denominator, and the range of normative values of the numerator for different values of the denominator. The ratio model assumes that the regression coefficient for the denominator predicting the numerator is equal to the mean or median of the ratio variable, and that the intercept of this regression model is equal to zero. Whenever there is a difference between the mean of the ratio variable and the real regression coefficient fitted using an ordinary least squares model, as well as a deviation from zero of the intercept of this fitted model, there will be bias in the numerator predictions made by the ratio model.

I have exemplified this situation with a simulated dataset of shoulder external and internal rotation strength. As can be seen in **FIGURE 2**, the ratio model does not fit well the data, with huge bias in the predictions made about the numerator variable, as well as the range of normative values.

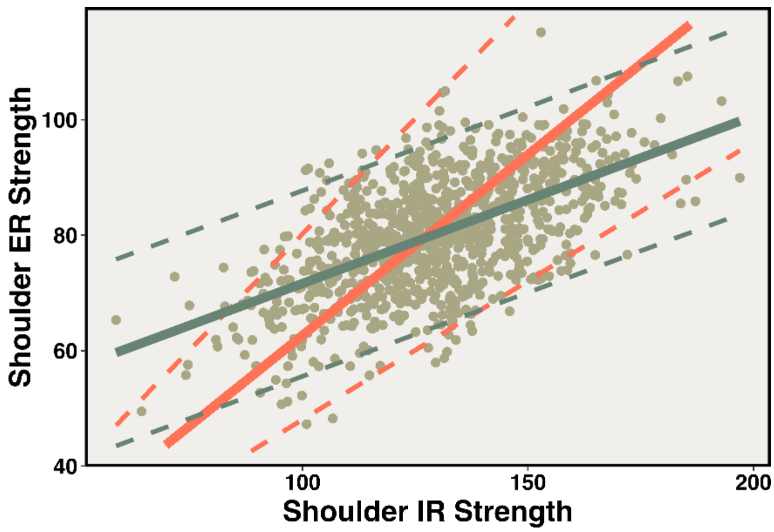

Figure 2. Scatterplot for the simulated dataset of shoulder strength. Blue lines refer to a linear regression fitted by ordinary least squares, being the dashed ones the 95% prediction interval. Red lines refer to the predictions of the mean, and percentiles 2.5 and 97.5, calculated using the ratio model.

Spurious correlations when using ratio variables

Another problem of using ratio variables is the presence of spurious correlations derived from the inappropriate adjustment for the denominator variable (i.e., the ratio is still correlated with the denominator variable).

If we assume that the denominator variable is related to an outcome measure, but the numerator is not (null hypothesis is true), and we aim to evaluate the relationship between the numerator and the outcome measure, adjusted for the denominator variable by using the ratio model, this will lead to an increase in the type 1 error rate (false positive results). This is proven by Monte Carlo simulations, which results are presented in **FIGURE 3**. As can be seen, there is a huge increase in type 1 error rates. For example, assuming a sample size of 50 subjects, with a correlation of r = 0.50 between denominator and the outcome measure, and the denominator and the numerator, the rate of false positive results is 42.6%.

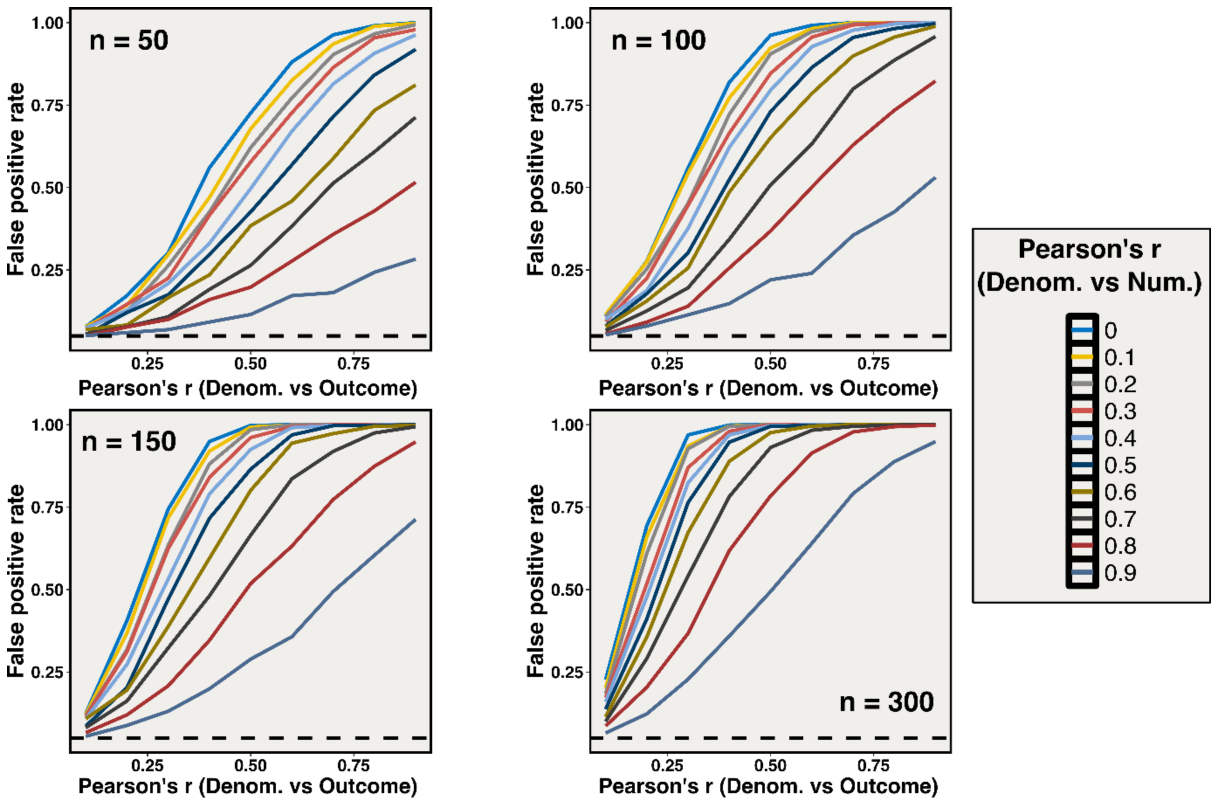

Figure 3. Type 1 error rate (false positive rate) when using the ratio model to adjust for a confounding variable, as a function of the correlation between the denominator and the outcome measure, and the numerator, as well as the sample size. The Monte Carlo simulations are based on 1,000 samples per combination of correlations and sample size, with the three variables having the same coefficient of variation.

Another scenario is when the denominator is present in both sides of the regression equation (i.e., outcome divided by the denominator predicted by the numerator divided by the denominator), leading to what is called mathematical coupling.[5] An example of this kind of analysis is presented in the study of Martins et al.,[8] who predicted hip external rotator strength by hip extensor strength, normalizing both measures by body mass. In 1999, Ji-Hyun Kim[5] provided and exact formula for estimating the spurious correlation between ratios with a common divisor, when all three variables are uncorrelated, as a function of their coefficients of variation. For example, when all three variables have the same coefficient of variation, the spurious correlation is expected to be $r = 0.51$.[5] In **FIGURE 4** it is presented a full simulation of the expected spurious correlations for different combinations of coefficients of variation.

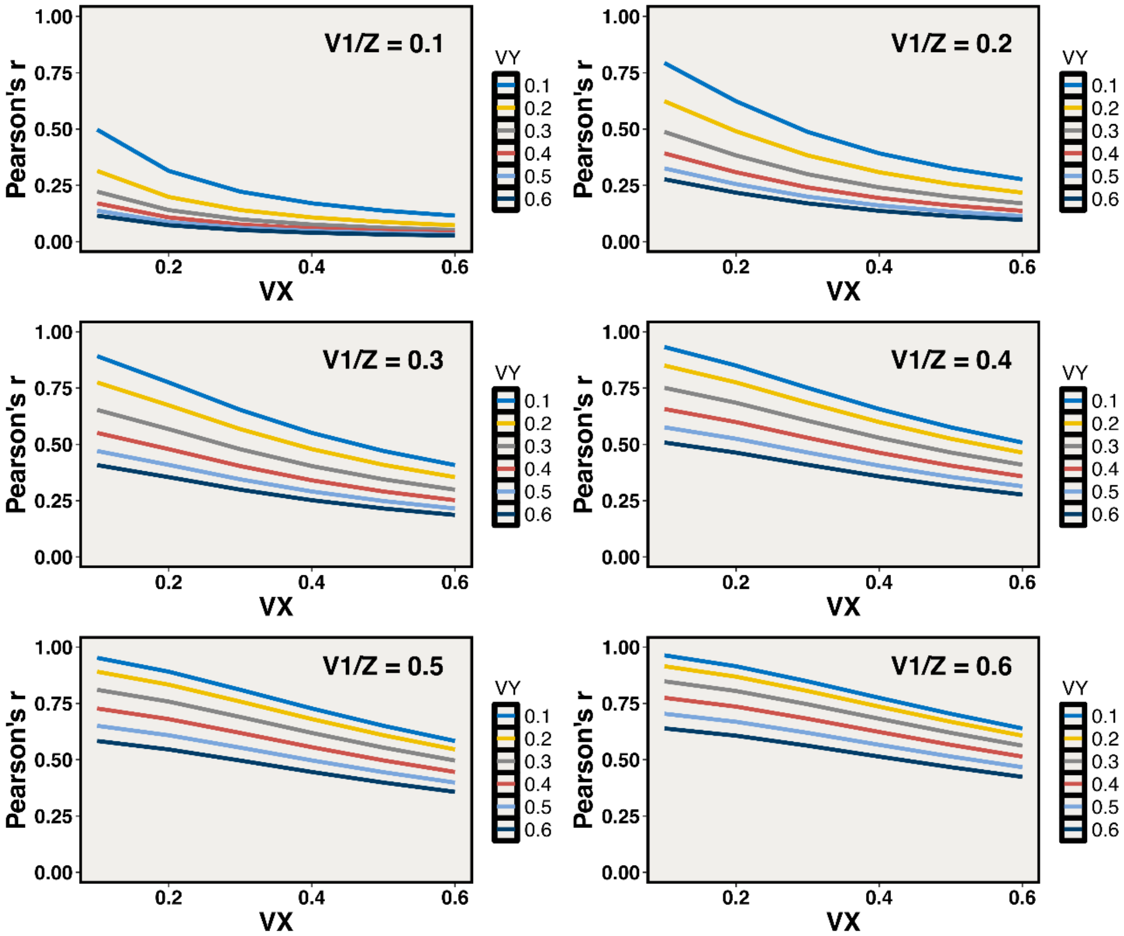

Figure 4. Spurious correlation between variables with common divisor. VX, VY and V1/Z refer to the coefficient of variation of the numerator variables X, Y and the inverse of Z (denominator variable).

An example of how to calculate normative values using regression analyses

There are many ways to model data properly using regression analyses, even when there is heteroskedasticity, in aim to estimate normative values for an outcome measure as a function of one or more predictors, without the need of using ratio variables.

A simulated dataset of 400 subjects was created for shoulder isometric internal rotation and external rotation strength measurements in Newtons, assuming a positive linear heteroskedastic relationship. The selected statistical model was quantile regression, using the package *‘quantreg’* by Roger Koenker.[6] This kind of regression analysis allows to directly estimate desired quantiles of the outcome measure, allowing for the presence of heteroskedasticity, and even non-linear relationships. Furthermore, this model has the advantage of not making distributional assumptions of the outcome measure. A scatterplot of the predictions made by the two models is presented in **FIGURE 5**.

The 10th, 50th (median), and 90th percentiles were estimated by three quantile regression models. The predictions equations are:

Quantile regression

$$Median\ (ShER) = 36.77 + 0.53 * ShIR$$
$$10^{th}\ percentile\ (ShER) = 35.55 + 0.43 * ShIR$$
$$90^{th}\ percentile\ (ShER) = 35.43 + 0.65 * ShIR$$

Ratio model

$$Median\ (ShER) = 0 + 0.82 * ShIR$$
$$10^{th}\ percentile\ (ShER) = 0 + 0.70 * ShIR$$
$$90^{th}\ percentile\ (ShER) = 0 + 0.95 * ShIR$$

ShER = Predicted shoulder external rotation strength

ShIR = Observed shoulder internal rotation strength

Based on these equations, we can estimate if a given subject is within “normative” values (i.e., expected 10th and 90th population values). For example, let's imagine that we have an athlete who has suffered a shoulder injury a couple of months ago. The patient presents with 105N of shoulder internal rotation strength, and 75N of shoulder external rotation strength. The estimated range of normative values is 80.70N to 102.78N. Based on this information, we could state that the patient has too low external rotation strength for its amount of internal rotation one, so we could prescribe some exercise to improve it in aim to prevent another injury. We could have defined the normative range with other percentiles, such as 25th and 75th or 2.5th and 97.5th, that could be estimated also with quantile regression models.

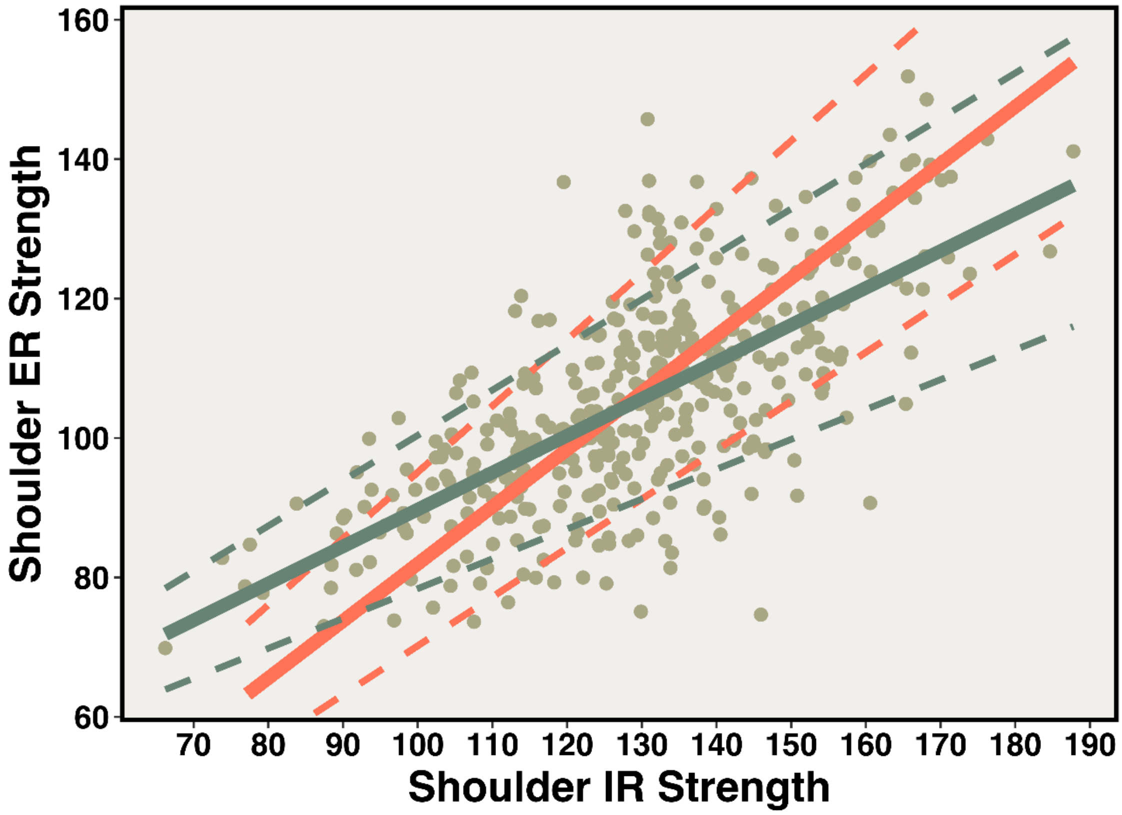

Figure 5. Scatterplot for the simulated dataset (n = 400) of shoulder internal (IR) and external (ER) rotation strength measurements in Newtons. The blue lines refer to the predictions made by the quantile regression models, and the red lines refer to the predictions made by the ratio model. Solid lines reflect predicted median external rotation strength, and dashed lines refer to the 10th and 90th percentiles.

The purpose of this example is not to recommend quantile regression as the best option, but to show the reader the benefits of using regression models to obtain normative values instead of using ratio variables.

Conclusion

The use of ratio variables in aim to “normalize” data is problematic and presents with multiple statistical assumptions that, if not met, lead to wrong predictions and estimation of normative values, and the presence of spurious associations. Researcher should implement appropriate regression models to avoid these issues when analyzing their data.

References

1. Curran-Everett D. Explorations in statistics: The analysis of ratios and normalized data. *Am J Physiol - Adv Physiol Educ*. 2013;37(3):213-219. doi:10.1152/ADVAN.00053.2013

2. Fernández-Matías R, Tu Y-K, Koppenhaver S. On the use of percent change within rehabilitative ultrasound imaging research: a systematic review with Monte Carlo simulations. *OSF Prepr*. Published online August 23, 2022:1-82. doi:10.31219/OSF.IO/K9QG4

3. Habechian FAP, Fornasari GG, Sacramento LS, Camargo PR. Differences in scapular kinematics and scapulohumeral rhythm during elevation and lowering of the arm between typical children and healthy adults. *J Electromyogr Kinesiol*. 2014;24(1):78-83. doi:10.1016/j.jelekin.2013.10.013

4. Johansson F, Asker M, Malmberg A, Fernández-Fernández J, Warnqvist A, Cools A. Eccentric and Isometric Shoulder Rotation Strength and Range of Motion: Normative Values for Adolescent Competitive Tennis Players. *Front Sport Act living*. 2022;4:798255.

5. Kim JH. Spurious correlation between ratios with a common divisor. *Stat Probab Lett*. 1999;44(4):383-386. doi:10.1016/S0167-7152(99)00030-9

6. Koenker R, Hallock KF. Quantile Regression. *J Econ Perspect*. 2001;15(4):143-156.

7. Kronmal RA. Spurious Correlation and the Fallacy of the Ratio Standard Revisited. *J R Stat Soc*. 1993;156(3):379-392. doi:10.2307/2983064

8. Martins SCS, Resende RA, Pinheiro LSP, et al. Prediction equation of hip external rotators maximum torque in healthy adults and older adults using the measure of hip extensors maximum torque. *Brazilian J Phys Ther*. 2021;25(4):415-420. doi:10.1016/J.BJPT.2020.08.005

9. Vickers AJ. The use of percentage change from baseline as an outcome in a controlled trial is statistically inefficient: A simulation study. *BMC Med Res Methodol*. 2001;1(1):1-4. doi:10.1186/1471-2288-1-6

SUPPLEMENTARY MATERIAL

R code used for simulations

```r
# Required packages
require(ggplot2)
require(MonteCarlo)
require(ggpubr)
require(MASS)
require(dplyr)
require(ggpubr)
require(ggsci)
require(quantreg)
```

```r
# Assumptions ratio variables
set.seed(1232)
Denominator <- rnorm(1000, 10, 3)
Ratio <- rnorm(100, 0.8, 0.1)
Numerator <- Denominator*Ratio
DF_assumptions <- as.data.frame(cbind(Denominator,Numerator,Ratio))
p1 <- ggplot(data = DF_assumptions, aes(x = Denominator, y = Numerator)) +
  geom_point(size = 2.5, colour = "#A9A684") +
  geom_smooth(method = "lm", colour = "#678374", size = 3, se = FALSE) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
```

```
  theme(panel.background = element_rect(colour = "black", size = 2,
                                         fill = "#F1EFEB"))
p2 <- ggplot(data = DF_assumptions, aes(x = Denominator, y = Ratio)) +
  geom_point(size = 2.5, colour = "#A9A684") +
  geom_smooth(method = "lm", colour = "#678374", size = 3, se = FALSE) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                         fill = "#F1EFEB"))
ggarrange(p1,p2) # Export size: 12x5
```

```
# Wrong predictions and normative values. Example with simulated shoulder ER/IR strength ratio
set.seed(1232)
sd1 <- 20
sd2 <- 10
cor <- 0.6
Shoulder_Strength <- mvrnorm(n = 1000, mu = c(130, 80),
                             Sigma = matrix(ncol=2,nrow=2,
c(sd1^2,sd1*sd2*cor,sd1*sd2*cor,sd2^2)))
colnames(Shoulder_Strength) <- c("Shoulder_IR", "Shoulder_ER")
Shoulder_Strength <- as.data.frame(Shoulder_Strength)
```

```
Shoulder_Strength$Ratio <-
Shoulder_Strength$Shoulder_ER/Shoulder_Strength$Shoulder_IR

Mean_Ratio <- mean(Shoulder_Strength$Ratio)

reg_model <- lm(Shoulder_ER ~ Shoulder_IR, data = Shoulder_Strength)

predictions <- predict(reg_model, interval = "prediction")

predictions <- as.data.frame(predictions)

predictions$Shoulder_IR <- Shoulder_Strength$Shoulder_IR

pctl_2.5 <- quantile(Shoulder_Strength$Ratio, probs = 0.025)

pctl_97.5 <- quantile(Shoulder_Strength$Ratio, probs = 0.975)

ggplot(data = Shoulder_Strength, aes(x = Shoulder_IR, y = Shoulder_ER)) +

  geom_point(size = 2.5, colour = "#A9A684") +

  geom_segment(x = min(Shoulder_Strength$Shoulder_IR+11),

               y = Mean_Ratio*min(Shoulder_Strength$Shoulder_IR+11),

               xend = max(Shoulder_Strength$Shoulder_IR-11),

               yend = Mean_Ratio*max(Shoulder_Strength$Shoulder_IR-11),

               colour = "coral1", size = 3) +

  geom_segment(x = min(Shoulder_Strength$Shoulder_IR+30),

               y = pctl_2.5*min(Shoulder_Strength$Shoulder_IR+30),

               xend = max(Shoulder_Strength$Shoulder_IR),

               yend = pctl_2.5*max(Shoulder_Strength$Shoulder_IR),

               colour = "coral1", size = 1.5, linetype = "dashed") +

  geom_segment(x = min(Shoulder_Strength$Shoulder_IR),

               y = pctl_97.5*min(Shoulder_Strength$Shoulder_IR),

               xend = max(Shoulder_Strength$Shoulder_IR-50),

               yend = pctl_97.5*max(Shoulder_Strength$Shoulder_IR-50),
```

```
             colour = "coral1", size = 1.5, linetype = "dashed") +
  geom_smooth(method = "lm", colour = "#678374", size = 3, se = FALSE) +
  geom_line(data = predictions, aes(x = Shoulder_IR, y = lwr), colour = "#678374",
          size = 1.5, linetype = "dashed") +
  geom_line(data = predictions, aes(x = Shoulder_IR, y = upr), colour = "#678374",
          size = 1.5, linetype = "dashed") +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  xlab("Shoulder IR Strength") + ylab("Shoulder ER Strength")
```

```
# Spurious correlations when using ratio variables. Simulation number one.
simulate_spurious <- function(n, cor1, cor2){
  Denominator <- rnorm(n)
  Numerator <- cor1*Denominator + rnorm(n,0,sd = sqrt(1-cor1^2))
  Outcome <- cor2*Denominator + rnorm(n,0,sd = sqrt(1-cor2^2))
  Denominator <- Denominator + 10
  Numerator <- Numerator + 10
  Outcome <- Outcome + 10
  Ratio <- Numerator/Denominator
```

```
  pvalue_ratio <- cor.test(Ratio, Outcome)$p.value
  pvalue_regression <- summary(lm(Outcome ~ Denominator +
Numerator))$coefficients[3,4]
  return(list("pvalue_ratio"=pvalue_ratio, "pvalue_regression"=pvalue_regression))
}
cor1 <- c(seq(0,0.9, by = 0.1))
cor2 <- c(seq(0.1,0.9, by = 0.1))
n <- c(50, 100, 150, 300)
param_list <- list("n"=n, "cor1"=cor1, "cor2"=cor2)
MC_simulations <- MonteCarlo(func = simulate_spurious, nrep = 1000, param_list =
param_list)
Frame_simulations <- MakeFrame(MC_simulations)
Frame_false_positives <- Frame_simulations %>%
  group_by(n, cor1, cor2) %>%
  summarize(
    false_positives = sum(pvalue_ratio < 0.05),
    false_positive_rate = false_positives / 1000
  )
Frame_false_positives$cor1 <- as.factor(Frame_false_positives$cor1)
plot_50 <- ggplot(data = Frame_false_positives[Frame_false_positives$n == "50",],
       aes(x = cor2, y = false_positive_rate, colour = cor1)) + geom_line(size = 1.5) +
  geom_hline(yintercept = 0.05, size = 1.5, colour = "black", linetype = "dashed") +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
```

```
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("Correlation between denominator and outcome") +
  ylab("False positive rate") + labs(color = "Correlation D vs. N")


plot_100 <- ggplot(data = Frame_false_positives[Frame_false_positives$n ==
"100",],
                   aes(x = cor2, y = false_positive_rate, colour = cor1)) + geom_line(size =
1.5) +
  geom_hline(yintercept = 0.05, size = 1.5, colour = "black", linetype = "dashed") +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
```

```
  xlab("Correlation between denominator and outcome") +
  ylab("False positive rate") + labs(color = "Correlation D vs. N")
plot_150 <- ggplot(data = Frame_false_positives[Frame_false_positives$n ==
"150",],
                  aes(x = cor2, y = false_positive_rate, colour = cor1)) + geom_line(size =
1.5) +
  geom_hline(yintercept = 0.05, size = 1.5, colour = "black", linetype = "dashed") +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("Correlation between denominator and outcome") +
  ylab("False positive rate") + labs(color = "Correlation D vs. N")
plot_300 <- ggplot(data = Frame_false_positives[Frame_false_positives$n ==
"300",],
                  aes(x = cor2, y = false_positive_rate, colour = cor1)) + geom_line(size =
1.5) +
  geom_hline(yintercept = 0.05, size = 1.5, colour = "black", linetype = "dashed") +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
```

```
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("Correlation between denominator and outcome") +
  ylab("False positive rate") + labs(color = "Correlation D vs. N")
ggarrange(plot_50, plot_100, plot_150, plot_300,nrow = 2, ncol = 2)
```

```
# Spurious correlations when correlating two variables with a common divisor.
# Based on the formula of Ji-Hyun Kim (1999).
# Spurious Correlation between Ratios with a Common Divisor.
# https://www.sciencedirect.com/science/article/abs/pii/S0167715299000309
cor_common_divisor <- function(V1Z, VX, VY){
  r <- (V1Z^2) / (
    sqrt(
      ((VX^2)*(1 + V1Z^2) + V1Z^2)*
        ((VY^2)*(1 + V1Z^2) + V1Z^2)
    )
  )
  return(list("r"=r))
}
```

```
V1Z <- c(seq(0.1,0.6,by = 0.1))

VX <- c(seq(0.1,0.6,by = 0.1))

VY <- c(seq(0.1,0.6,by = 0.1))

param_list <- list("V1Z"=V1Z, "VX"=VX, "VY"=VY)

MC_cor_common_divisor <- MonteCarlo(func = cor_common_divisor,

                                    nrep = 1, param_list = param_list)

Frame_cor_common_divisor <- MakeFrame(MC_cor_common_divisor)

Frame_cor_common_divisor$r <- as.numeric(Frame_cor_common_divisor$r)

Frame_cor_common_divisor$VY <- as.factor(Frame_cor_common_divisor$VY)

plot_01 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.1",],

       aes(x = VX, y = r, color = VY)) +

  geom_line(size = 1.5) + ylim(0,1) +

  scale_color_manual(values = pal_jco()(10)) +

  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +

  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +

  theme(panel.grid.major = element_blank()) +

  theme(panel.grid.minor = element_blank()) +

  theme(panel.border = element_blank()) +

  theme(panel.background = element_rect(colour = "black", size = 2,

                                        fill = "#F1EFEB")) +

  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+

  xlab("VX") +

  ylab("Pearson's r") + labs(color = "VY")
```

```
plot_02 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.2",],
                  aes(x = VX, y = r, color = VY)) +
  geom_line(size = 1.5) + ylim(0,1) +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("VX") +
  ylab("Pearson's r") + labs(color = "VY")
plot_03 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.3",],
                  aes(x = VX, y = r, color = VY)) +
  geom_line(size = 1.5) + ylim(0,1) +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
```

```
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("VX") +
  ylab("Pearson's r") + labs(color = "VY")
plot_04 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.4",],
                  aes(x = VX, y = r, color = VY)) +
  geom_line(size = 1.5) + ylim(0,1) +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("VX") +
  ylab("Pearson's r") + labs(color = "VY")
plot_05 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.5",],
```

```
                 aes(x = VX, y = r, color = VY)) +
  geom_line(size = 1.5) + ylim(0,1) +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                         fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("VX") +
  ylab("Pearson's r") + labs(color = "VY")
plot_06 <- ggplot(data =
Frame_cor_common_divisor[Frame_cor_common_divisor$V1Z == "0.6",],
                 aes(x = VX, y = r, color = VY)) +
  geom_line(size = 1.5) + ylim(0,1) +
  scale_color_manual(values = pal_jco()(10)) +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
```

```
                                    fill = "#F1EFEB")) +
  theme(legend.title = element_text(size = 15), legend.text = element_text(size = 15))
+
  xlab("VX") +
  ylab("Pearson's r") + labs(color = "VY")
ggarrange(plot_01, plot_02, plot_03, plot_04,
         plot_05, plot_06, nrow = 3, ncol = 2)
```

```
# Example of good predictions and normative values using quantile regression.
set.seed(1232)
Shoulder_IR <- rnorm(n = 400, mean = 130, sd = 20)
Shoulder_ER <- 40 + Shoulder_IR*(0.5 + 0.3*rnorm(n = 400, 0, 0.3))
Ratio_Shoulder <- Shoulder_ER/Shoulder_IR
Data_Shoulder <- as.data.frame(cbind(Shoulder_ER, Shoulder_IR, Ratio_Shoulder))
reg_quantile_low <- rq(Shoulder_ER ~ Shoulder_IR, data = Data_Shoulder, tau =
0.10)
reg_quantile_median <- rq(Shoulder_ER ~ Shoulder_IR, data = Data_Shoulder, tau
= 0.50)
reg_quantile_high <- rq(Shoulder_ER ~ Shoulder_IR, data = Data_Shoulder, tau =
0.90)
Data_Shoulder$Predicted_median <- reg_quantile_median$fitted.values
Data_Shoulder$Predicted_qlow <- reg_quantile_low$fitted.values
Data_Shoulder$Predicted_qhigh <- reg_quantile_high$fitted.values
Median_Ratio <- median(Data_Shoulder$Ratio_Shoulder)
q10_Ratio <- quantile(Data_Shoulder$Ratio_Shoulder, probs = 0.10)
```

```
q90_Ratio <- quantile(Data_Shoulder$Ratio_Shoulder, probs = 0.90)

ggplot(data = Data_Shoulder, aes(x = Shoulder_IR, y = Shoulder_ER)) +

  geom_point(size = 2.5, colour = "#A9A684") +

  geom_segment(x = min(Data_Shoulder$Shoulder_IR+11),

               y = Median_Ratio*min(Data_Shoulder$Shoulder_IR+11),

               xend = max(Data_Shoulder$Shoulder_IR),

               yend = Median_Ratio*max(Data_Shoulder$Shoulder_IR),

               colour = "coral1", size = 3) +

  geom_segment(x = min(Data_Shoulder$Shoulder_IR+11),

               y = q90_Ratio*min(Data_Shoulder$Shoulder_IR+11),

               xend = max(Data_Shoulder$Shoulder_IR - 20),

               yend = q90_Ratio*max(Data_Shoulder$Shoulder_IR - 20),

               colour = "coral1", size = 1.5, linetype = "dashed") +

  geom_segment(x = min(Data_Shoulder$Shoulder_IR+20),

               y = q10_Ratio*min(Data_Shoulder$Shoulder_IR+20),

               xend = max(Data_Shoulder$Shoulder_IR),

               yend = q10_Ratio*max(Data_Shoulder$Shoulder_IR),

               colour = "coral1", size = 1.5, linetype = "dashed") +

  geom_line(data = Data_Shoulder, aes(x = Shoulder_IR, y = Predicted_median),
colour = "#678374",

            size = 3) +

  geom_line(data = Data_Shoulder, aes(x = Shoulder_IR, y = Predicted_qlow), colour
= "#678374",

            size = 1.5, linetype = "dashed") +
```

```
  geom_line(data = Data_Shoulder, aes(x = Shoulder_IR, y = Predicted_qhigh),
colour = "#678374",
            size = 1.5, linetype = "dashed") +
  theme(axis.text = element_text(size = 15, face = "bold", colour = "black")) +
  theme(axis.title = element_text(size = 20, face = "bold", colour = "black")) +
  theme(panel.grid.major = element_blank()) +
  theme(panel.grid.minor = element_blank()) +
  theme(panel.border = element_blank()) +
  theme(panel.background = element_rect(colour = "black", size = 2,
                                        fill = "#F1EFEB")) +
  scale_x_continuous(breaks = seq(70, 190, by = 10)) +
  xlab("Shoulder IR Strength") + ylab("Shoulder ER Strength")
```