STAT202 Lab 2: Multiple Linear Regression START

David Ewing

Due on 7 August 2024

# Introduction

This document contains the analysis for Lab 2. It explores biometric data using multiple linear regression to understand the relationships between various predictors and the response variable weight\_kg. The analysis will follow these steps:”

# Step 0: setup

loading libraries:

set.seed(82171165) #set seed   
  
knitr::opts\_chunk$set(  
 echo = TRUE, # Show all code by default  
 message = TRUE, # Include package messages  
 warning = TRUE # Include warnings if they occur  
)  
  
  
  
library(conflicted)  
library(tidyverse)

## Warning: package 'lubridate' was built under R version 4.4.2

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.4 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2

library(readxl)

## Warning: package 'readxl' was built under R version 4.4.2

library(readr)   
library(performance)

## Warning: package 'performance' was built under R version 4.4.2

library(GGally)

## Warning: package 'GGally' was built under R version 4.4.2

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(flextable)

## Warning: package 'flextable' was built under R version 4.4.2

library(broom)

## Warning: package 'broom' was built under R version 4.4.2

library(skimr)

## Warning: package 'skimr' was built under R version 4.4.2

library(data.table)

## Warning: package 'data.table' was built under R version 4.4.2

library(lmtest)

## Warning: package 'lmtest' was built under R version 4.4.2

## Loading required package: zoo  
##   
## Attaching package: 'zoo'  
##   
## The following objects are masked from 'package:data.table':  
##   
## yearmon, yearqtr  
##   
## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

conflict\_prefer("filter", "dplyr"); conflict\_prefer("select", "dplyr")

## [conflicted] Will prefer dplyr::filter over any other package.  
## [conflicted] Will prefer dplyr::select over any other package.

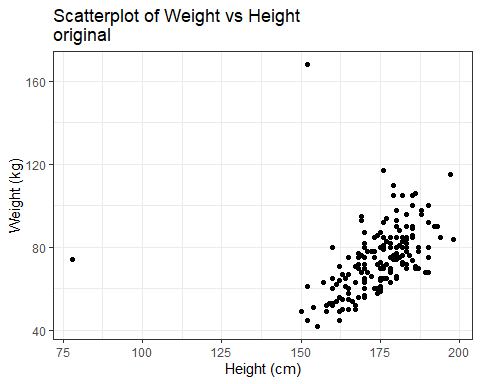
# Step 1: Read biometrics.xlsx

# Load the data  
biomet <- read\_excel("../data/biometrics.xlsx")  
  
# Summarise missing values using skimr  
skim\_biomet <- skim(biomet) |>  
 select(skim\_variable, n\_missing)  
  
skim\_biomet

## # A tibble: 6 × 2  
## skim\_variable n\_missing  
## <chr> <int>  
## 1 sex 0  
## 2 weight\_kg 0  
## 3 height\_cm 0  
## 4 handspan\_cm 0  
## 5 finger2\_cm 1  
## 6 finger4\_cm 1

# Step 2: Scatterplot of Weight vs Height

befor <- biomet |>  
 ggplot(aes(x = height\_cm, y = weight\_kg)) + # Scatterplot  
 geom\_point() +  
 labs(  
 title = "Scatterplot of Weight vs Height \noriginal",  
 x = "Height (cm)",  
 y = "Weight (kg)"  
 ) +  
 theme\_bw()  
befor

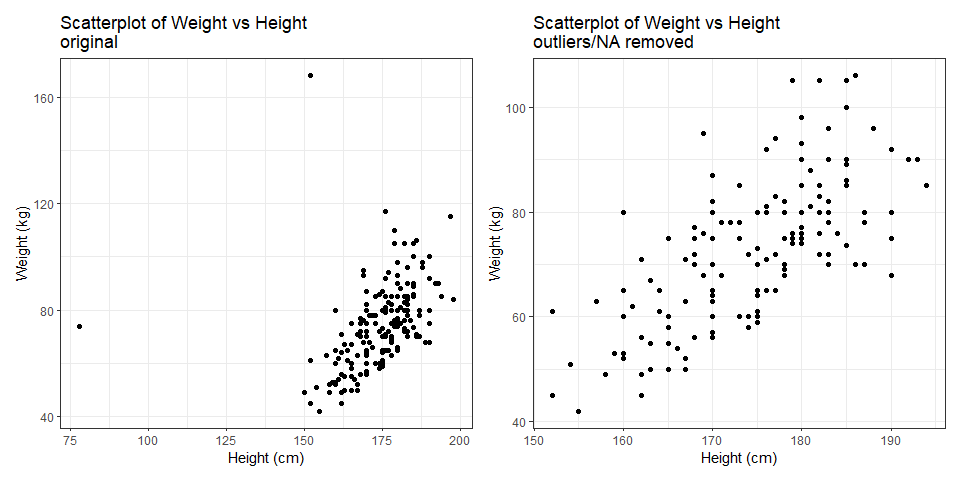


# Step 3: Drop missing and filter outliers

biomet\_0 <- biomet |>  
 drop\_na( finger2\_cm, finger4\_cm) # drop missing values identified in skim()   
  
remove\_outliers <- function(data, column) { # Function to remove outliers   
 outliers <- boxplot.stats(data[[column]])$out # identify outliers  
 data |> filter(!(data[[column]] %in% outliers)) # filter outliers  
}  
  
library(patchwork)

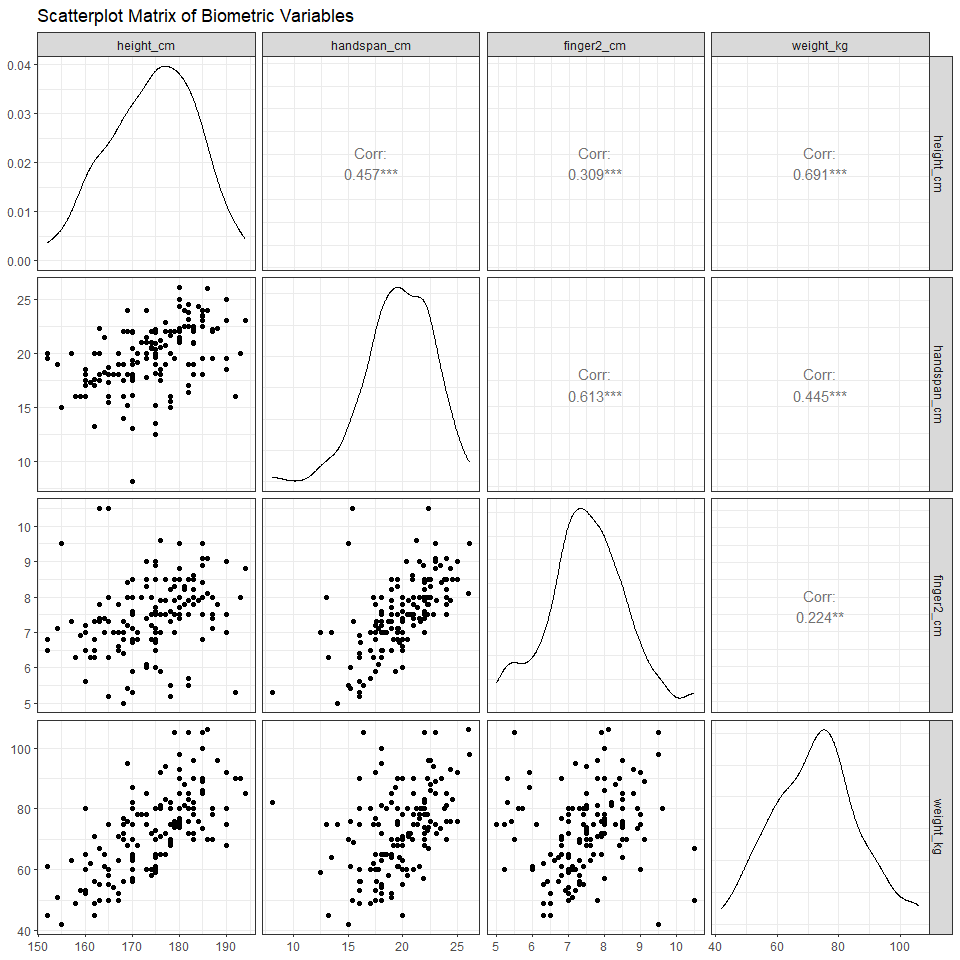
## Warning: package 'patchwork' was built under R version 4.4.2

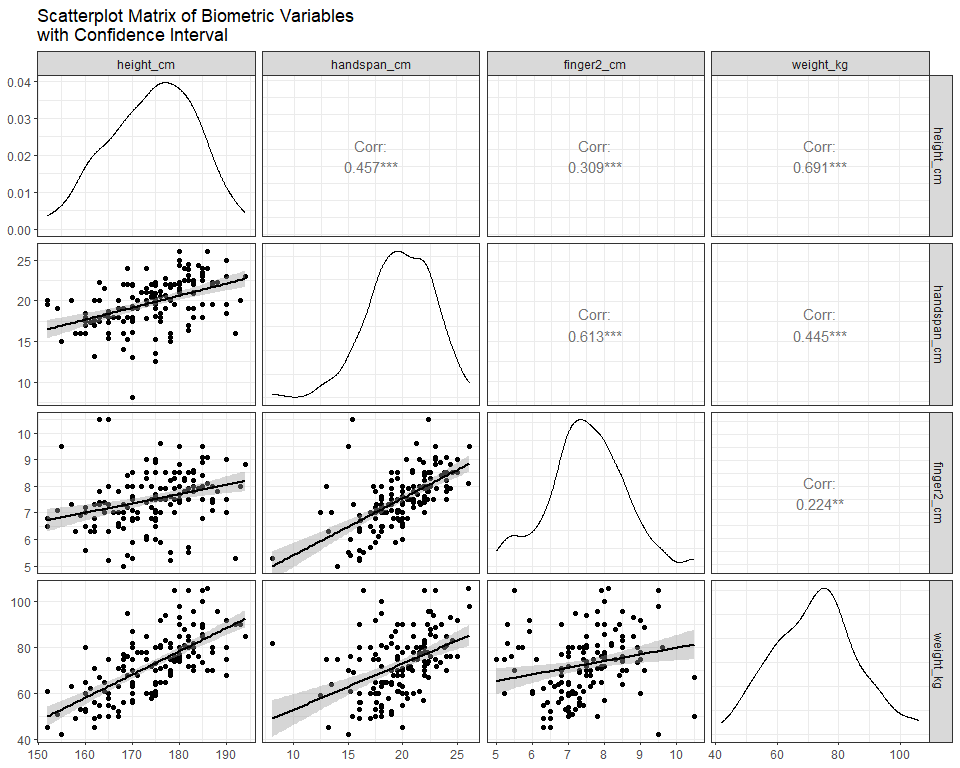
biomet\_my <- biomet |>  
 drop\_na( finger2\_cm, finger4\_cm) |> # drop missing values identified in skim()   
 remove\_outliers("height\_cm") |> # remove outliers   
 remove\_outliers("weight\_kg") |> # remove outliers   
 slice\_sample(n = 150) # take a random sample  
   
after <- biomet\_my |>  
 ggplot(aes(x = height\_cm, y = weight\_kg)) + # Scatterplot  
 geom\_point() +  
 labs(  
 title = "Scatterplot of Weight vs Height \noutliers/NA removed",  
 x = "Height (cm)",  
 y = "Weight (kg)"  
 ) +  
 theme\_bw()  
   
 befor + after



# Step 4: Scatterplot matrix

# Scatterplot matrix for selected variables  
biomet\_my |>  
 select(height\_cm, handspan\_cm, finger2\_cm, weight\_kg) |>  
 ggpairs(title = "Scatterplot Matrix of Biometric Variables") +  
 theme\_bw()





Comment:

The scatterplot matrix shows positive relationships between variables. Height and weight show the strongest correlation (0.691) with a narrow confidence interval. Handspan and finger2 have moderate correlations with weight (0.445 and 0.224, respectively). Handspan and finger2 are also moderately correlated (0.613), indicating some relationship.

# Step 5: Fit Multiple Linear Regression Models

# Fit the models  
m1 <- lm(weight\_kg ~ height\_cm, data = biomet\_my)  
m2 <- lm(weight\_kg ~ handspan\_cm, data = biomet\_my)  
m3 <- lm(weight\_kg ~ finger2\_cm, data = biomet\_my)  
m4 <- lm(weight\_kg ~ height\_cm + handspan\_cm, data = biomet\_my)  
m5 <- lm(weight\_kg ~ height\_cm + handspan\_cm + finger2\_cm, data = biomet\_my)  
  
# Summarise the model performance  
library(broom)  
models\_summary <- list(m1, m2, m3, m4, m5) |>  
 purrr::map\_dfr(glance, .id = "model") |>  
 select(model, r.squared, adj.r.squared, sigma)  
  
# Mutate to include rounded values and an observation column  
models\_summary\_display <- models\_summary |>  
 mutate(  
 r.squared = round(r.squared, 3),  
 adj.r.squared = round(adj.r.squared, 3),  
 sigma = round(sigma, 3),  
 adj.R.Square.observation = case\_when(  
 model == "1" ~ "highest for single predictors.",  
 model == "2" ~ "moderate for single predictors.",  
 model == "3" ~ "the lowest for single predictors.",  
 model == "4" ~ " handspan increases slightly.",  
 model == "5" ~ " minimal effect for 3-predictors."  
 )  
 )

# Display the updated summary  
models\_summary\_display |>  
 flextable() |>  
 set\_caption("Model Comparison: Adjusted Metrics and Observations") |>  
 autofit() # Automatically adjust table size

Model Comparison: Adjusted Metrics and Observations

| model | r.squared | adj.r.squared | sigma | adj.R.Square.observation |
| --- | --- | --- | --- | --- |
| 1 | 0.477 | 0.473 | 9.783 | highest for single predictors. |
| 2 | 0.198 | 0.193 | 12.111 | moderate for single predictors. |
| 3 | 0.050 | 0.044 | 13.184 | the lowest for single predictors. |
| 4 | 0.498 | 0.491 | 9.614 | handspan increases slightly. |
| 5 | 0.505 | 0.495 | 9.577 | minimal effect for 3-predictors. |

Comment:

Adjusted R-squared improves from 0.473 in Model 1 to 0.495 in Model 5. Adding handspan in Model 4 increases Adjusted R-squared modestly (0.491), while adding finger2 in Model 5 results in a minimal improvement (0.495). This suggests to me that height and handspan are the most influential predictors of weight.

# Step 6: Compare Models m4 and m5

# Perform ANOVA to compare m4 and m5  
anova\_results <- anova(m4, m5)  
  
# Display the results  
as.data.frame(summary(m4)$coefficients) |> flextable() |>autofit()

| Estimate | Std. Error | t value | Pr(>|t|) |
| --- | --- | --- | --- |
| -99.7475921 | 15.08360262 | -6.612982 | 0.0000000006528637696201 |
| 0.9042911 | 0.09648207 | 9.372634 | 0.0000000000000001179095 |
| 0.7468058 | 0.29895241 | 2.498076 | 0.0135881399888256090142 |

as.data.frame(summary(m5)$coefficients) |> flextable() |>autofit()

| Estimate | Std. Error | t value | Pr(>|t|) |
| --- | --- | --- | --- |
| -95.9962862 | 15.24124402 | -6.298455 | 0.00000000333788688050796 |
| 0.9100533 | 0.09618729 | 9.461264 | 0.00000000000000007324034 |
| 1.0405786 | 0.35892171 | 2.899180 | 0.00431945035992133824498 |
| -1.4088226 | 0.96087068 | -1.466194 | 0.14474602018354193466720 |

anova\_results |> flextable() |>autofit()

| Res.Df | RSS | Df | Sum of Sq | F | Pr(>F) |
| --- | --- | --- | --- | --- | --- |
| 147 | 13,587.86 |  |  |  |  |
| 146 | 13,390.69 | 1 | 197.1664 | 2.149724 | 0.144746 |

Which model fits best?

The ANOVA p-value for comparing m4 and m5 is above 0.05 (0.144746). The addition of finger2\_cm in m5 does not significantly improve the model fit.

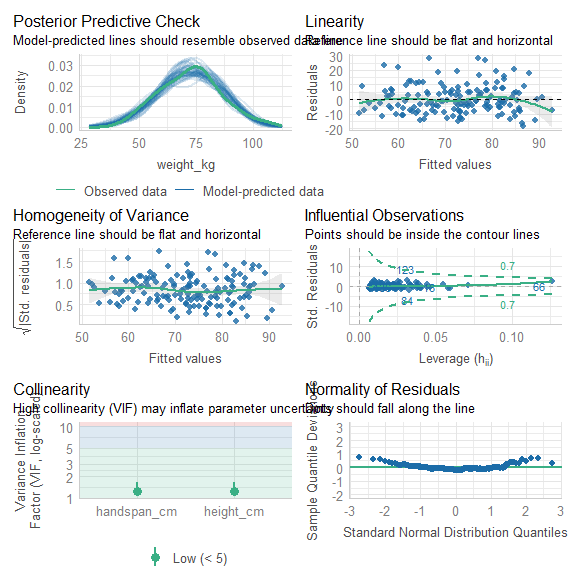
M4 is best as it is also simpler.

# Step 7:

Validate the best model (m4) based on Step 6) by usinge check\_model().

# Step 7: Validate Model Assumptions for m4

# Load necessary library  
library(performance)  
  
cm <- check\_model(m4)  
print(cm)



chedk\_model() Observation:

* Predictive Check: Model predictions align with observed data
* Linearity: Residuals scatter around zero, indicating linearity.
* Homoscedasticity: Variance is consistent across fitted values.
* Influential Observations: Points largely within Cook’s distance contour.
* Collinearity: VIF values are acceptable (< 5).
* Normality: Residuals follow the Q-Q plot line.

# Step 8: Read tricarpa.csv file

# Read the tricarpa.csv file  
tricarpa <- read\_csv("../data/tricarpa.csv")

## Rows: 962 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (3): MOE, dry\_density, acoustic\_velocity  
##   
## 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.

tri\_skim0 <- skim(tricarpa) |>  
 select(skim\_variable, n\_missing)  
  
tri\_skim0

## # A tibble: 3 × 2  
## skim\_variable n\_missing  
## <chr> <int>  
## 1 MOE 0  
## 2 dry\_density 0  
## 3 acoustic\_velocity 0

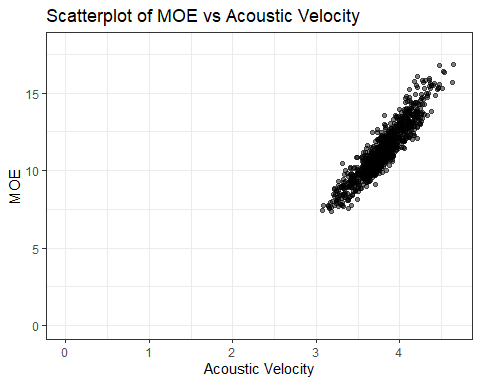
# Step 9: Random Sampling of 900 Observations

# Reset the seed for reproducibility  
set.seed(82171165) # Replace with your student ID number  
  
  
tri\_sample <- tricarpa |> slice\_sample(n = 900)  
my\_tri <- tri\_sample  
tri\_skim1 <- skim(tri\_sample) |>  
 select(skim\_variable, n\_missing)

# Step 10: Scatterplot of MOE vs Acoustic Velocity

# Scatterplot with axes starting at zero  
my\_tri |>   
 ggplot(aes(x = acoustic\_velocity, y = MOE)) +  
 geom\_point(alpha = 0.5) +  
 labs(  
 title = "Scatterplot of MOE vs Acoustic Velocity",  
 x = "Acoustic Velocity",  
 y = "MOE"  
 ) +  
 lims(x = c(0, max(my\_tri$acoustic\_velocity, na.rm = TRUE)), y = c(0, 18)) +  
 theme\_bw()

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_point()`).



The scatterplot shows a strong positive linear relationship. The data points are clustered around the trend, suggesting a high degree of correlation . A warning indicates that one row was removed due to missing values presuably outside the scale range. acoustic\_velocity is likely a significant predictor of MOE.

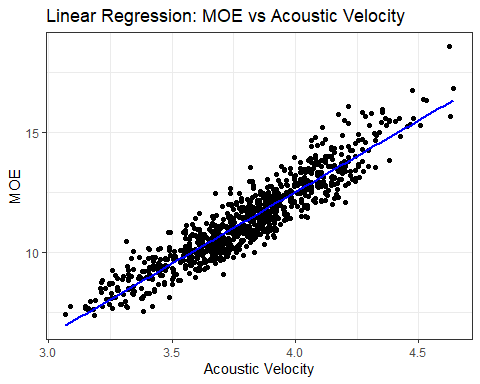
# Step 11: Fit Linear Regression Model

# Fit the linear regression model  
moe\_model <- lm(MOE ~ acoustic\_velocity, data = my\_tri)  
  
# Display model summary  
step11\_summary <- summary(moe\_model)  
step11\_summary

##   
## Call:  
## lm(formula = MOE ~ acoustic\_velocity, data = my\_tri)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.69620 -0.46304 -0.01932 0.41874 2.33507   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -11.4116 0.3051 -37.40 <2e-16 \*\*\*  
## acoustic\_velocity 5.9806 0.0800 74.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6588 on 898 degrees of freedom  
## Multiple R-squared: 0.8616, Adjusted R-squared: 0.8614   
## F-statistic: 5589 on 1 and 898 DF, p-value: < 2.2e-16

# Visualise the regression line  
my\_tri |>   
 ggplot(aes(x = acoustic\_velocity, y = MOE)) +  
 geom\_point() +  
 geom\_smooth(method = "lm", se = FALSE, colour = "blue") +  
 labs(  
 title = "Linear Regression: MOE vs Acoustic Velocity",  
 x = "Acoustic Velocity",  
 y = "MOE"  
 ) +  
 theme\_bw()

## `geom\_smooth()` using formula = 'y ~ x'



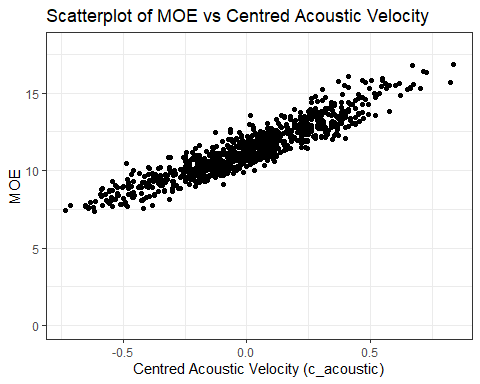
For every one-unit increase in acoustic\_velocity, MOE is predicted to increase by approximately 5.98, while the intercept of -11.41 lacks practical significance in this context but is required to properly position the regression line. As the p-value for the slope is <2e-16, the model demonstrates a highly statistically significant relation. The adjusted R-squared value (0.8614) suggests that acoustic\_velocity explains approximately 86.14% of the variation in MOE, reiterating a strong model for predicting MOE.

# Step 12: Create Centred Acoustic Velocity and Scatterplot

# Create centred variable  
my\_tri <- my\_tri |>  
 mutate(c\_acoustic = acoustic\_velocity - mean(acoustic\_velocity))  
  
# Scatterplot with centred acoustic velocity  
tri\_splot <- my\_tri |>   
 ggplot(aes(x = c\_acoustic, y = MOE)) +  
 geom\_point() +  
 labs(  
 title = "Scatterplot of MOE vs Centred Acoustic Velocity",  
 x = "Centred Acoustic Velocity (c\_acoustic)",  
 y = "MOE"  
 ) +  
 lims(y = c(0, 18)) +  
 theme\_bw()

tri\_splot

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_point()`).



Centring acoustic\_velocity by introducing c\_acoustic (mean of zero) improves the interpretability of the model by making the intercept represent the predicted MOE for the average value of acoustic\_velocity, which has a more practical meaning. Additionally, centring helps to reduce multicollinearity when the interaction or polynomial terms are included in the model, Overall, centring enhances both the interpretability and numerical stability of the model.

# Step 13: Fit Regression Model Using Centred Variable

# Fit the regression model with centred acoustic velocity  
moe\_model\_centre <- lm(MOE ~ c\_acoustic, data = my\_tri)  
  
  
step13\_summary <- summary(moe\_model\_centre)  
step13\_summary

##   
## Call:  
## lm(formula = MOE ~ c\_acoustic, data = my\_tri)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.69620 -0.46304 -0.01932 0.41874 2.33507   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 11.33703 0.02196 516.28 <2e-16 \*\*\*  
## c\_acoustic 5.98057 0.08000 74.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6588 on 898 degrees of freedom  
## Multiple R-squared: 0.8616, Adjusted R-squared: 0.8614   
## F-statistic: 5589 on 1 and 898 DF, p-value: < 2.2e-16

# Create variables dynamically  
step11\_values <- c(  
 coef(step11\_summary)[1, 1], # Intercept  
 coef(step11\_summary)[2, 1], # Slope  
 step11\_summary$r.squared, # R-squared  
 step11\_summary$adj.r.squared, # Adjusted R-squared  
 step11\_summary$sigma # Residual Std. Error  
)  
  
step13\_values <- c(  
 coef(step13\_summary)[1, 1], # Intercept  
 coef(step13\_summary)[2, 1], # Slope  
 step13\_summary$r.squared, # R-squared  
 step13\_summary$adj.r.squared, # Adjusted R-squared  
 step13\_summary$sigma # Residual Std. Error  
)  
  
# Create a comparison data frame  
comparison\_df <- data.frame(  
 Metric = c(  
 "Intercept", "Slope", "R-squared",   
 "Adj R-squared", "Residual Std. Error"  
 ),  
 `Step 11` = step11\_values,  
 `Step 13` = step13\_values,  
 Observation = c(  
 "Intercept is the MOE when acoustic\_velocity = 0",  
 "Slope is identical; centering does not change relationship strength",  
 "R-squared remains the same; centering does not affect fit",  
 "Adjusted R-squared remains the same",  
 "Residual standard error remains the same"  
 )  
)  
  
# Create a rounded version of the data for improved readability  
comparison\_df\_display <- comparison\_df  
comparison\_df\_display[ ,2:3] <- round(comparison\_df\_display[ ,2:3], 3)  
  
# Use flextable to format the table  
library(flextable)  
ft <- flextable(comparison\_df\_display) %>%  
 set\_caption("Comparison of Regression Models: Step 11 vs Step 13") %>%  
 bg(part = "header", bg = "#D3D3D3") %>% # Grey background for header  
 theme\_box() %>% # Simplified border styling  
 align(j = 1, align = "left", part = "all") %>% # Left-align Metric column  
 align(j = 2:3, align = "center", part = "all") %>% # Center-align model columns  
 align(j = 4, align = "left", part = "all") %>% # Left-align Observation column  
 border\_inner\_v(part = "all") %>% # Add vertical borders  
 border\_inner\_h(part = "all") %>% # Add horizontal borders  
 border\_outer(part = "all") %>% # Add outer borders  
 autofit() # Automatically adjust table size

ft

Comparison of Regression Models: Step 11 vs Step 13

| **Metric** | **Step.11** | **Step.13** | **Observation** |
| --- | --- | --- | --- |
| Intercept | -11.412 | 11.337 | Intercept is the MOE when acoustic\_velocity = 0 |
| Slope | 5.981 | 5.981 | Slope is identical; centering does not change relationship strength |
| R-squared | 0.862 | 0.862 | R-squared remains the same; centering does not affect fit |
| Adj R-squared | 0.861 | 0.861 | Adjusted R-squared remains the same |
| Residual Std. Error | 0.659 | 0.659 | Residual standard error remains the same |

The identical values for the slope, R-squared, adjusted R-squared, and residual standard error in both models (except the intercept) suggest that centring does not alter the underlying relationships between the predictor and the response.