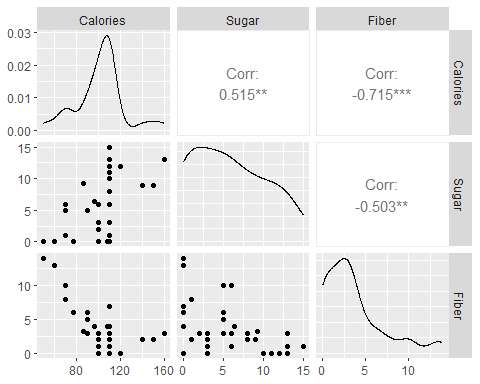
Homework 4

library(here) ## File Path Management  
library(statcalpolypackage) ## Data Extraction  
library(gato365dsh2024) ## Data Extraction  
library(dplyr) ## Data Transformation  
library(ggplot2) ## Data Visualization  
library(GGally) ## Data Visualization  
library(broom) ## Data Analysis  
source(here("R","assessment\_regression.R"))

## Cereal dataset

The Cereal dataset contains nutritional information about different breakfast cereals, including variables like Calories, Sugar content, and Fiber. It allows us to explore how certain ingredients, like Sugar, are related to the overall calorie count of cereals. This dataset is useful for practicing regression, diagnostics, and understanding when transformations might be needed to better model relationships.

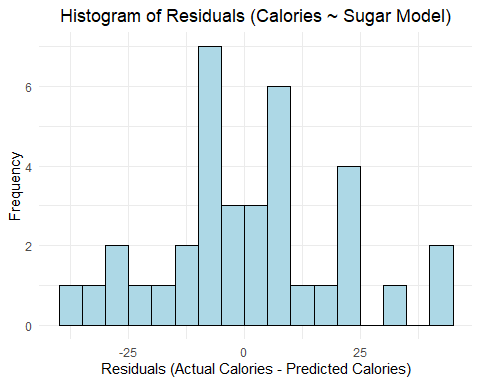
Cereal %>%   
 select(-Cereal) %>%   
 ggpairs()



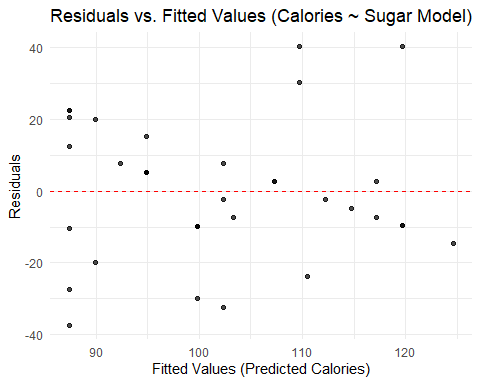
# Fit the linear model using the lm() function  
lm\_cereal <- lm(Calories ~ Sugar, data = Cereal)  
summary(lm\_cereal)

Call:  
lm(formula = Calories ~ Sugar, data = Cereal)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-37.428 -9.832 0.245 8.909 40.322   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 87.4277 5.1627 16.935 <2e-16 \*\*\*  
Sugar 2.4808 0.7074 3.507 0.0013 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.27 on 34 degrees of freedom  
Multiple R-squared: 0.2656, Adjusted R-squared: 0.244   
F-statistic: 12.3 on 1 and 34 DF, p-value: 0.001296

augmented\_cereal <- augment(lm\_cereal)  
  
  
ggplot(augmented\_cereal, aes(x =.resid)) +  
 geom\_histogram(binwidth = 5, fill = "lightblue", color = "black", boundary = 0) + # Adjusted binning  
 labs(title = "Histogram of Residuals (Calories ~ Sugar Model)",  
 x = "Residuals (Actual Calories - Predicted Calories)",  
 y = "Frequency") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))



ggplot(augmented\_cereal, aes(x =.fitted, y =.resid)) +  
 geom\_point(alpha = 0.7) +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "red") + # Reference line at zero residual  
 labs(title = "Residuals vs. Fitted Values (Calories ~ Sugar Model)",  
 x = "Fitted Values (Predicted Calories)",  
 y = "Residuals") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))



(Answer Key)

n\_cereal <- nrow(Cereal)  
p\_cereal <- 1   
leverage\_threshold\_cereal <- 2 \* p\_cereal / n\_cereal  
cooks\_threshold\_cereal <- 4 / n\_cereal  
resid\_sd\_cereal <- sd(augmented\_cereal$.resid)

**Leverage** (Answer Key)

leverage\_threshold\_cereal

[1] 0.05555556

**Cooks Distance** (Answer Key)

cooks\_threshold\_cereal

[1] 0.1111111

augmented\_cereal %>%   
 arrange(desc(.cooksd)) %>%   
 head(10) %>%   
 round(3)

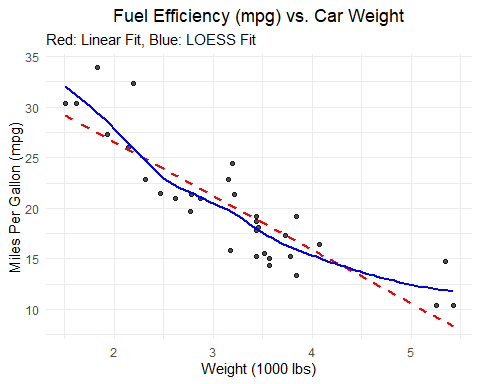
# A tibble: 10 × 8  
 Calories Sugar .fitted .resid .hat .sigma .cooksd .std.resid  
 <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 160 13 120. 40.3 0.099 18.1 0.268 2.20   
 2 50 0 87.4 -37.4 0.072 18.4 0.157 -2.02   
 3 150 9 110. 40.2 0.042 18.2 0.101 2.13   
 4 60 0 87.4 -27.4 0.072 18.9 0.084 -1.48   
 5 110 0 87.4 22.6 0.072 19.1 0.057 1.22   
 6 110 0 87.4 22.6 0.072 19.1 0.057 1.22   
 7 140 9 110. 30.2 0.042 18.8 0.057 1.60   
 8 110 15 125. -14.6 0.144 19.4 0.057 -0.821  
 9 108 0 87.4 20.6 0.072 19.2 0.047 1.11   
10 70 6 102. -32.3 0.028 18.7 0.041 -1.70

## mtcars dataset

The mtcars dataset is a classic and widely used dataset in R that contains specifications and performance data for 32 different car models from the 1974 Motor Trend magazine. Each row represents a unique vehicle, and each column records a specific attribute related to engine performance, design, or efficiency. Some key variables include mpg (miles per gallon), hp (gross horsepower), wt (weight in 1000 lbs), drat (rear axle ratio), and qsec (quarter-mile time). Additionally, the dataset includes categorical variables encoded as numeric values, such as cyl (number of cylinders), am (transmission type), and gear (number of forward gears).

ggplot(mtcars, aes(x = wt, y = mpg)) +  
 geom\_point(alpha = 0.7) +  
 geom\_smooth(method = "lm", se = FALSE, color = "red", linetype = "dashed") + # Add linear trend  
 geom\_smooth(method = "loess", se = FALSE, color = "blue") + # Add non-linear trend (LOESS)  
 labs(title = "Fuel Efficiency (mpg) vs. Car Weight",  
 subtitle = "Red: Linear Fit, Blue: LOESS Fit",  
 x = "Weight (1000 lbs)",  
 y = "Miles Per Gallon (mpg)") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))

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



**LOESS (Locally Estimated Scatterplot Smoothing)** fits many small, simple models to localized sections of the data instead of assuming one global line.  
It creates a smooth curve that captures bends and changes in the relationship between variables.  
LOESS is important because it helps reveal patterns that a simple linear model might miss, especially when the true relationship is nonlinear.

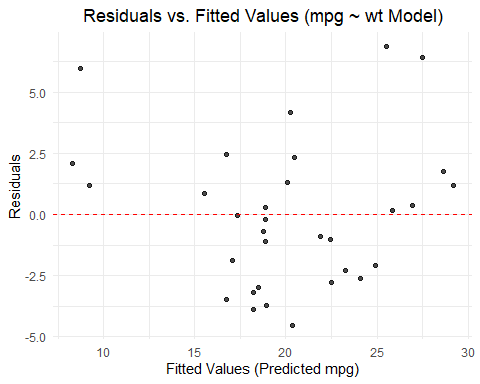
# Fit the initial linear model  
lm\_mtcars\_orig <- lm(mpg ~ wt, data = mtcars\_data)  
  
# Display the model summary  
summary(lm\_mtcars\_orig)

Call:  
lm(formula = mpg ~ wt, data = mtcars\_data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-4.5432 -2.3647 -0.1252 1.4096 6.8727   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 37.2851 1.8776 19.858 < 2e-16 \*\*\*  
wt -5.3445 0.5591 -9.559 1.29e-10 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 3.046 on 30 degrees of freedom  
Multiple R-squared: 0.7528, Adjusted R-squared: 0.7446   
F-statistic: 91.38 on 1 and 30 DF, p-value: 1.294e-10

summarize\_reg\_model(lm\_mtcars\_orig,"mpg ~ wt Model")

type RSS RSE R2 Adj\_R2 AIC BIC  
1 mpg ~ wt Model 278.32 3.05 0.75 0.74 71.22 72.68

augmented\_mtcars <- augment(lm\_mtcars\_orig)  
  
ggplot(augmented\_mtcars, aes(x =.fitted, y =.resid)) +  
 geom\_point(alpha = 0.7) +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "red") +   
 labs(title = "Residuals vs. Fitted Values (mpg ~ wt Model)",  
 x = "Fitted Values (Predicted mpg)",  
 y = "Residuals") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))



(Answer Key)

n\_mtcars <- nrow(mtcars)  
p\_mtcars <- 1   
leverage\_threshold\_mtcars <- 2 \* p\_mtcars / n\_mtcars  
cooks\_threshold\_mtcars <- 4 / n\_mtcars

**Leverage** (Answer Key)

leverage\_threshold\_mtcars

[1] 0.0625

**Cooks Distance** (Answer Key)

cooks\_threshold\_mtcars

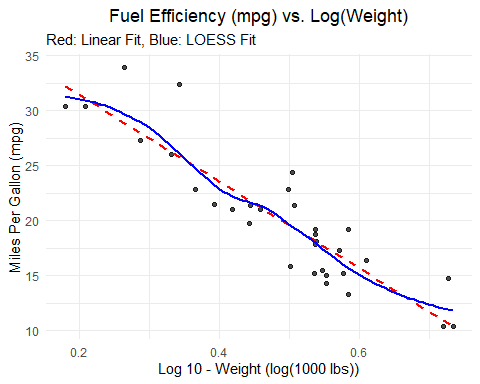
[1] 0.125

augmented\_mtcars %>%   
 arrange(desc(.cooksd)) %>%   
 head(10) %>%   
 round(3)

# A tibble: 10 × 8  
 mpg wt .fitted .resid .hat .sigma .cooksd .std.resid  
 <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 14.7 5.34 8.72 5.98 0.184 2.84 0.532 2.17  
 2 33.9 1.84 27.5 6.42 0.096 2.83 0.26 2.22  
 3 32.4 2.2 25.5 6.87 0.066 2.80 0.193 2.34  
 4 10.4 5.42 8.30 2.10 0.195 3.07 0.072 0.77  
 5 15.8 3.17 20.3 -4.54 0.031 2.98 0.037 -1.52  
 6 13.3 3.84 16.8 -3.46 0.044 3.03 0.031 -1.16  
 7 14.3 3.57 18.2 -3.90 0.035 3.01 0.031 -1.31  
 8 24.4 3.19 20.2 4.16 0.031 3.00 0.031 1.39  
 9 15.2 3.44 18.9 -3.73 0.033 3.02 0.026 -1.24  
10 30.4 1.62 28.7 1.75 0.118 3.08 0.025 0.61

mtcars\_data <- mtcars %>%  
 mutate(log10\_wt = log10(wt))  
  
  
  
ggplot(mtcars\_data, aes(x = log10\_wt, y = mpg)) +  
 geom\_point(alpha = 0.7) +  
 geom\_smooth(method = "lm", se = FALSE, color = "red", linetype = "dashed") +  
 geom\_smooth(method = "loess", se = FALSE, color = "blue") +  
 labs(title = "Fuel Efficiency (mpg) vs. Log(Weight)",  
 subtitle = "Red: Linear Fit, Blue: LOESS Fit",  
 x = "Log 10 - Weight (log(1000 lbs))",  
 y = "Miles Per Gallon (mpg)") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))

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



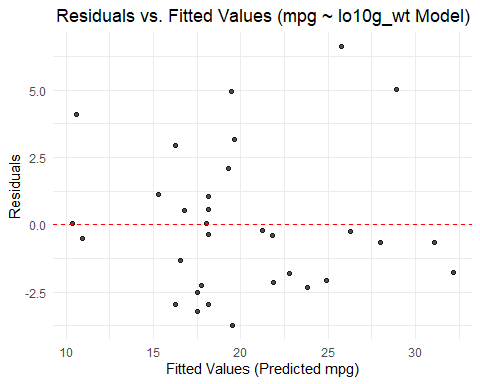
lm\_mtcars\_transformed <- lm(mpg ~ log10\_wt, data = mtcars\_data)  
  
summary(lm\_mtcars\_transformed)

Call:  
lm(formula = mpg ~ log10\_wt, data = mtcars\_data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-3.7440 -2.0954 -0.3672 1.0709 6.6150   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 39.257 1.758 22.32 < 2e-16 \*\*\*  
log10\_wt -39.342 3.477 -11.31 2.39e-12 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 2.669 on 30 degrees of freedom  
Multiple R-squared: 0.8101, Adjusted R-squared: 0.8038   
F-statistic: 128 on 1 and 30 DF, p-value: 2.391e-12

summarize\_reg\_model(lm\_mtcars\_transformed, "mpg ~ log10\_wt Model")

type RSS RSE R2 Adj\_R2 AIC BIC  
1 mpg ~ log10\_wt Model 213.78 2.67 0.81 0.8 62.77 64.24

augmented\_mtcars\_transformed <- augment(lm\_mtcars\_transformed)  
  
ggplot(augmented\_mtcars\_transformed, aes(x =.fitted, y =.resid)) +  
 geom\_point(alpha = 0.7) +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "red") +   
 labs(title = "Residuals vs. Fitted Values (mpg ~ lo10g\_wt Model)",  
 x = "Fitted Values (Predicted mpg)",  
 y = "Residuals") +  
 theme\_minimal() +   
 theme(plot.title = element\_text(hjust = 0.5))



(Answer Key)

n\_mtcars <- nrow(mtcars)  
p\_mtcars <- 1   
leverage\_threshold\_mtcars <- 2 \* p\_mtcars / n\_mtcars  
cooks\_threshold\_mtcars <- 4 / n\_mtcars

**Leverage** (Answer Key)

leverage\_threshold\_mtcars

[1] 0.0625

**Cooks Distance** (Answer Key)

cooks\_threshold\_mtcars

[1] 0.125

augmented\_mtcars\_transformed %>%   
 select(-.rownames) %>%   
 arrange(desc(.cooksd)) %>%   
 head(10) %>%   
 round(3)

# A tibble: 10 × 8  
 mpg log10\_wt .fitted .resid .hat .sigma .cooksd .std.resid  
 <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 33.9 0.264 28.9 5.01 0.116 2.53 0.262 2.00   
 2 32.4 0.342 25.8 6.62 0.067 2.40 0.235 2.56   
 3 14.7 0.728 10.6 4.08 0.13 2.59 0.2 1.64   
 4 30.4 0.18 32.2 -1.78 0.191 2.69 0.065 -0.742  
 5 24.4 0.504 19.4 4.96 0.032 2.55 0.058 1.89   
 6 15.8 0.501 19.5 -3.74 0.032 2.62 0.033 -1.42   
 7 13.3 0.584 16.3 -2.97 0.047 2.66 0.032 -1.14   
 8 19.2 0.585 16.2 2.95 0.047 2.66 0.032 1.13   
 9 14.3 0.553 17.5 -3.21 0.039 2.65 0.03 -1.23   
10 15.2 0.536 18.2 -2.97 0.035 2.66 0.023 -1.13