Build and Evaluate Multiple Linear Regression Models

Nithya Sarabudla

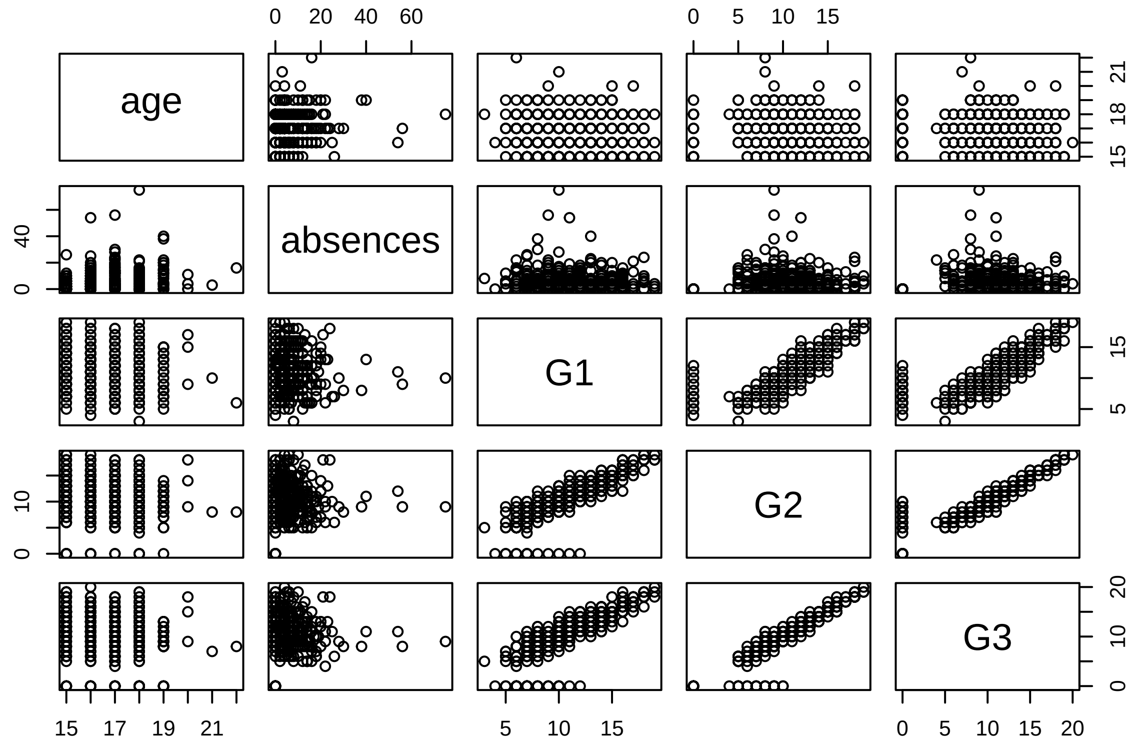
03-03-2024

```
# Load the dataset "student-mat.csv"
student_math <- read.csv("/Users/nithyasarabudla/DA5030/student-mat.csv",sep = ";", header = TRUE, stri
```

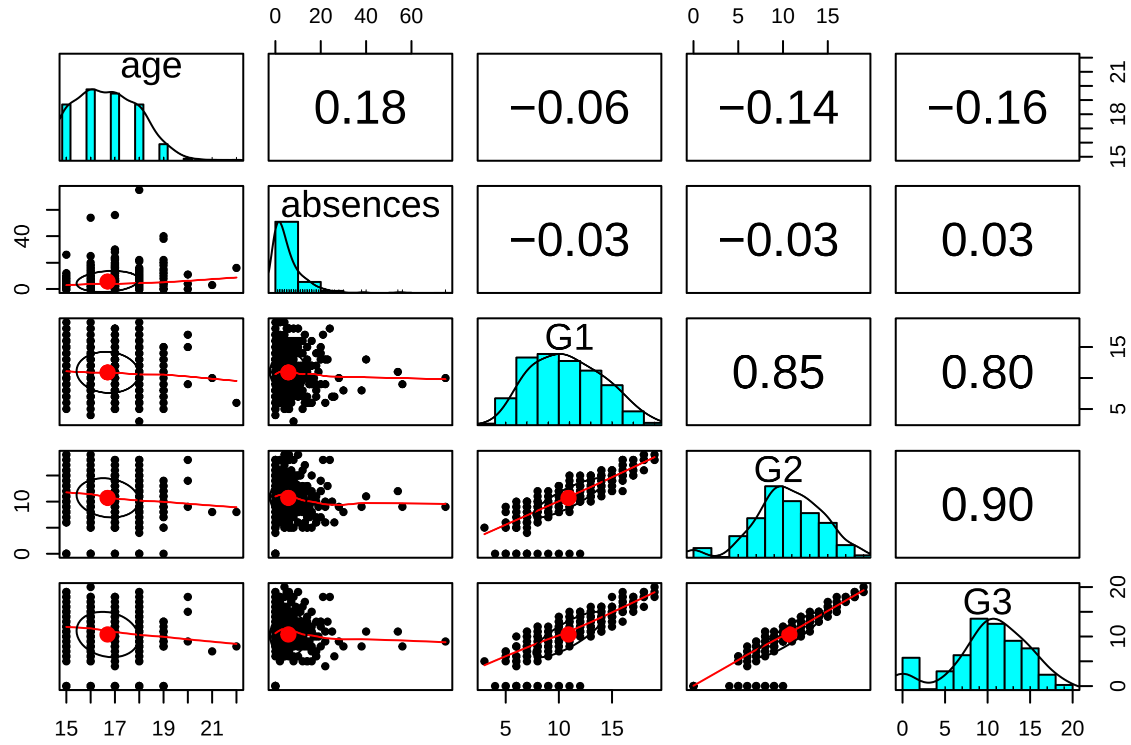
Question 1

Create scatter plots and pairwise correlations between age, absences, G1, and G2 and final grade (G3) using the pairs.panels() function in R.

```
# Load the psych library
library(psych)
# Create scatter plots for pairwise comparisons between the variables
pairs(student_math[c("age", "absences", "G1", "G2","G3")])
```



```
# Create enhanced scatter plots with pairwise correlations and distributions
pairs.panels(student_math[c("age", "absences", "G1", "G2","G3")])
```



Question 2

Build a multiple regression model predicting final math grade (G3) using as many features as you like but you must use at least four. Include at least one categorical variables and be sure to encode it properly using a method of your choice. Select the features that you believe are useful – you do not have to include all features.

```
# load the library dplyr
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Select relevant features for the regression model
# Assuming 'sex' is the categorical variable and 'G1', 'G2', 'studytime', 'absences' are the continuous
model <- lm(G3 ~ sex + G1 + G2 + studytime + absences, data = student_math)

# Evaluate the model
summary(model)
```

```
##
## Call:
## lm(formula = G3 ~ sex + G1 + G2 + studytime + absences, data = student_math)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.4443 -0.3279  0.3025  1.0077  3.7286
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.92764    0.41460  -4.649 4.57e-06 ***
## sexM         0.15293    0.20612   0.742  0.45857
## G1           0.15822    0.05590   2.830  0.00489 **
## G2           0.98733    0.04907  20.119  < 2e-16 ***
## studytime   -0.11823    0.12379  -0.955  0.34014
## absences     0.03625    0.01214   2.985  0.00301 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.917 on 389 degrees of freedom
## Multiple R-squared:  0.8272, Adjusted R-squared:  0.8249
## F-statistic: 372.3 on 5 and 389 DF,  p-value: < 2.2e-16
```

Question 3

Using the model from (2), use stepwise backward elimination to remove all non-significant variables and then state the final model as an equation. State the backward elimination measure you applied (p-value, AIC, Adjusted R2).

```
# Perform backward elimination based on AIC
final_model <- step(model, direction="backward")
```

```
## Start:  AIC=520.02
## G3 ~ sex + G1 + G2 + studytime + absences
##
##             Df Sum of Sq    RSS    AIC
## - sex        1      2.02 1431.4 518.58
## - studytime  1      3.35 1432.8 518.94
## <none>                   1429.4 520.02
## - G1         1     29.44 1458.8 526.07
## - absences   1     32.75 1462.2 526.97
## - G2         1   1487.36 2916.8 799.74
##
## Step:  AIC=518.58
## G3 ~ G1 + G2 + studytime + absences
```

```
##
##             Df Sum of Sq    RSS    AIC
## - studytime  1      5.96 1437.4 518.22
## <none>                   1431.4 518.58
## - G1         1     30.45 1461.9 524.89
## - absences   1     31.58 1463.0 525.20
## - G2         1   1490.96 2922.4 798.50
##
## Step:  AIC=518.22
## G3 ~ G1 + G2 + absences
##
##            Df Sum of Sq    RSS    AIC
## <none>                  1437.4 518.22
## - G1        1     28.39 1465.8 523.94
## - absences  1     33.32 1470.7 525.27
## - G2        1   1491.44 2928.8 797.37
```

```
summary(final_model)
```

```
##
## Call:
## lm(formula = G3 ~ G1 + G2 + absences, data = student_math)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3616 -0.3559  0.3163  0.9642  3.9242
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.06747    0.34116  -6.060  3.2e-09 ***
## G1           0.15452    0.05561   2.779  0.00572 **
## G2           0.98838    0.04907  20.142  < 2e-16 ***
## absences     0.03635    0.01208   3.010  0.00278 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.917 on 391 degrees of freedom
## Multiple R-squared:  0.8262, Adjusted R-squared:  0.8249
## F-statistic: 619.5 on 3 and 391 DF,  p-value: < 2.2e-16
```

Question 4

Calculate the 95% confidence interval for a prediction – you may choose any data you wish for some new student.

```
# New student data
new_student <- data.frame(
  sex = factor("M", levels = c("F", "M")),  # ensure the levels are the same as in the original model
  G1 = 14,
  G2 = 12,
  studytime = 2,
  absences = 0
)

# Predict G3 for the new student with a 95% confidence interval
predict_g3 <- predict(model, newdata = new_student, interval = "confidence", level = 0.95)

# Display the prediction and the confidence interval
print(predict_g3)
```

```
##        fit    lwr      upr
## 1 12.05183 11.672 12.43166
```

The model predicts a final math grade (G3) of 12.05 for the new student, with a 95% confidence interval ranging from 11.67 to 12.43. This range represents where the actual grade is likely to fall with 95% certainty.

Question 5

```
# Calculate the residuals
residuals <- model$residuals

# Calculate RMSE
rmse <- sqrt(mean(residuals^2))
rmse
```

```
## [1] 1.902296
```

The RMSE for this model is 1.902296

Question 6

```
# Handling Missing Values
student_math_data <- na.omit(student_math)
```

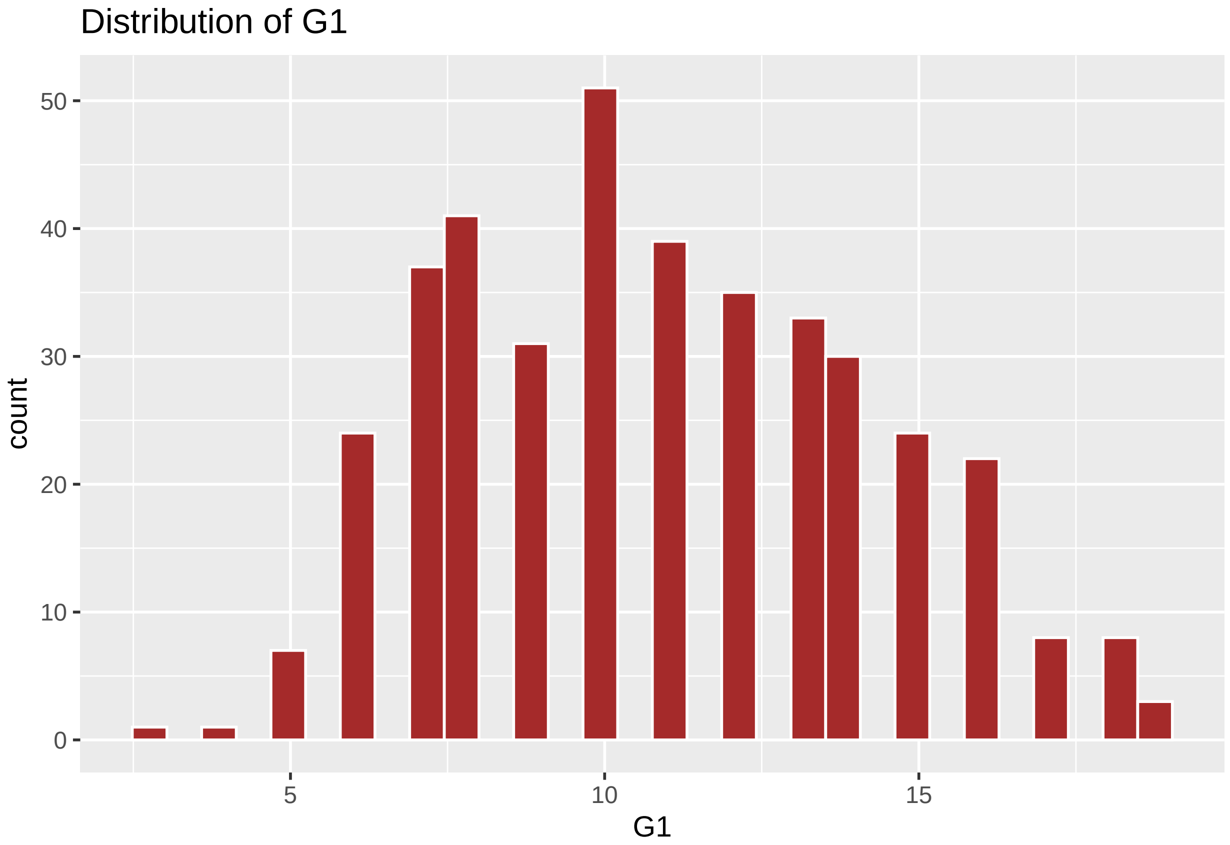
```
# Load the ggplot2 library for data visualization
library(ggplot2)
```

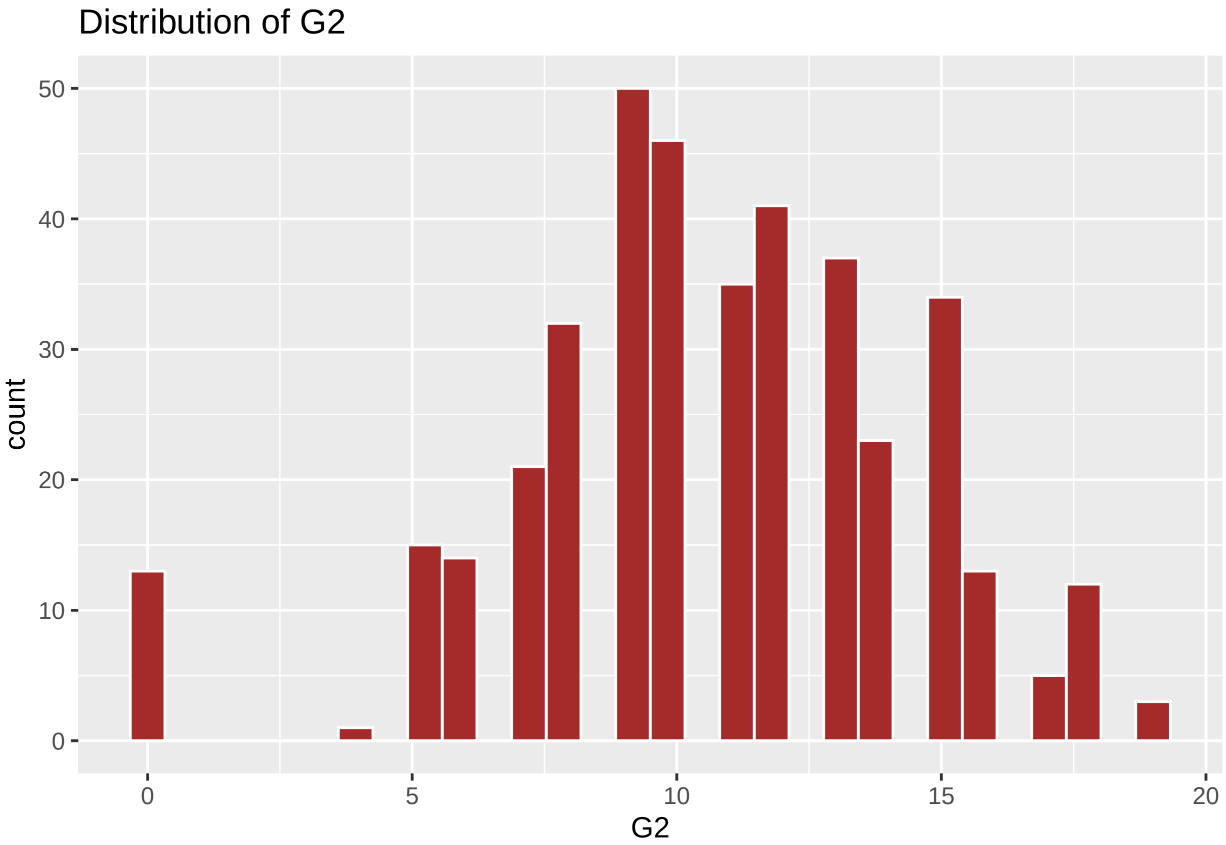
```
##
## Attaching package: 'ggplot2'
```

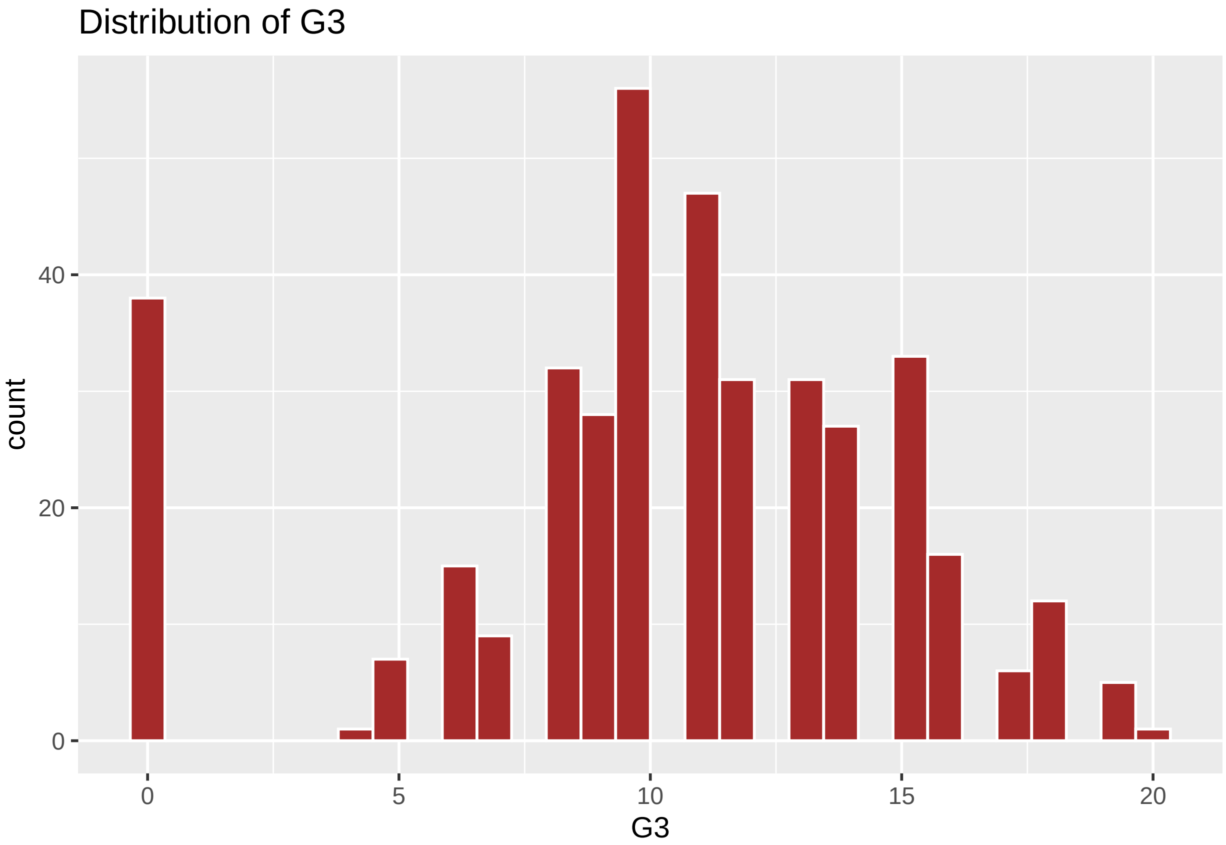
```
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

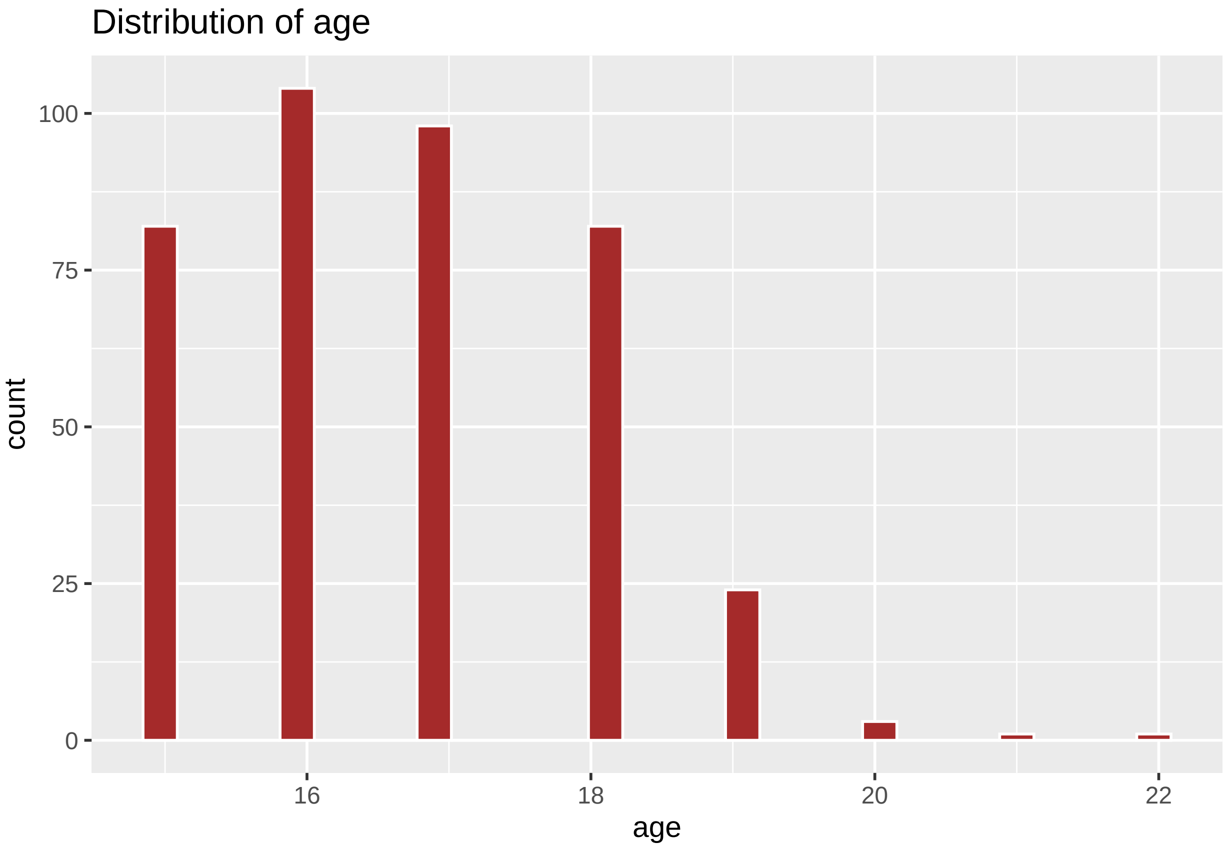
```
# Define the features to check for normality
features <- c("G1", "G2", "G3", "age", "absences")

# Loop through each feature to check for normality
for (feature in features) {
```

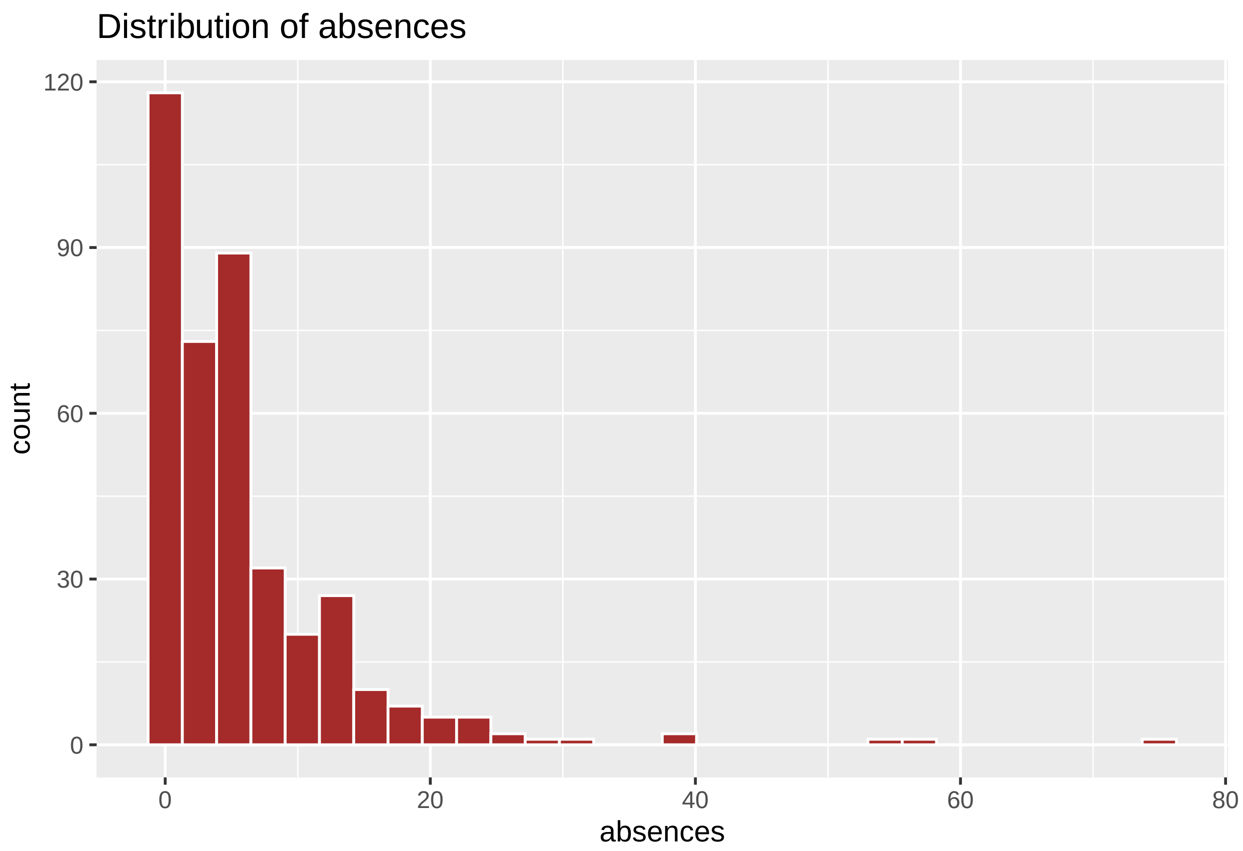
```r
  # Plot the distribution of the feature using a histogram
  plot_histogram <- ggplot(student_math, aes(x = .data[[feature]])) +
  geom_histogram(bins = 30, fill = "brown", color = "white") +
  labs(title = paste("Distribution of", feature))

print(plot_histogram)

  # Perform Shapiro-Wilk test for normality on the feature
  # If the p-value is less than 0.05, indicating non-normality, apply a log transformation
  if (shapiro.test(student_math[[feature]])$p.value < 0.05) {
    student_math[[feature]] <- log(student_math[[feature]])
  }
}
```




Distribution of G2
count
50
40
30
20
10
0
0
5
10
15
20
G2


Distribution of G3
count
40
20
0
0
5
10
15
20
G3


Distribution of age
count
100
75
50
25
0
16
18
20
22
age



```r
# Loop through each feature to identify and handle outliers
for (feature in features) {
  # Calculate basic statistics for the feature to identify outliers
  stats <- boxplot.stats(student_math_data[[feature]])

  detected_outliers <- stats$out

  # Print the number of detected outliers for each feature
  print(paste("Detected outliers for", feature, ":", length(detected_outliers)))


  if (length(detected_outliers) > 0) {

    student_math_data <- student_math_data[!student_math_data[[feature]] %in% detected_outliers, ]
  }
}
```

```
## [1] "Detected outliers for G1 : 0"
## [1] "Detected outliers for G2 : 13"
## [1] "Detected outliers for G3 : 25"
## [1] "Detected outliers for age : 1"
## [1] "Detected outliers for absences : 25"
```

```r
library(caret)
```

```
## Loading required package: lattice
```

```r
# Generate dummy variables for categorical features in the cleaned and preprocessed dataset
dummy_vars <- dummyVars(" ~ .", data = student_math_data)
transformed_student_math_data <- data.frame(predict(dummy_vars, newdata = student_math_data))

# Train the final linear regression model using the transformed dataset
final_linear_model <- lm(G3 ~ age + absences + G1 + G2 + sexF + addressU, data = transformed_student_ma

# Display the summary of the final model to evaluate its performance
summary(final_linear_model)
```

```
##
## Call:
## lm(formula = G3 ~ age + absences + G1 + G2 + sexF + addressU,
##     data = transformed_student_math_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2327 -0.3540 -0.1173  0.6891  2.7549
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.225941   0.717480  -0.315  0.75303
## age          0.025300   0.039879   0.634  0.52625
## absences    -0.007893   0.010833  -0.729  0.46676
## G1           0.108736   0.034533   3.149  0.00179 **
## G2           0.881924   0.036784  23.976  < 2e-16 ***
## sexF         0.024730   0.091882   0.269  0.78799
## addressU     0.166303   0.111374   1.493  0.13636
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8275 on 324 degrees of freedom
## Multiple R-squared:  0.9339, Adjusted R-squared:  0.9327
## F-statistic: 762.9 on 6 and 324 DF,  p-value: < 2.2e-16
```