CS 4407-01 Data Mining and Machine Learning

Instructor: Professor Shabia Shabir

Name: Anonymos

Programming Assignment Unit 3

1. Introduction

This assignment aims to construct linear and multiple linear regression models for data analysis. Linear regression is a statistical method that expresses the relationship between variables in a mathematical formula, aiding in prediction and interpretation of data. Multiple linear regression, in particular, allows the use of multiple independent variables to capture more complex data trends.

In this task, we first construct models based on the given dataset and estimate parameters, including the calculation of confidence intervals. Then, we attempt to simplify the model if necessary and verify whether the model assumptions are satisfied through residual analysis. Finally, we visualize the fitted line along with its 95% confidence and prediction intervals.

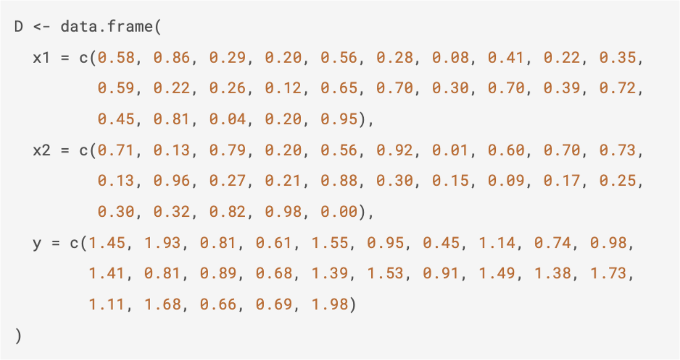
Additionally, we utilize the statistical analysis capabilities of the R programming language and construct models using the `lm.fit` function. This function supports both simple and multiple regression, providing the parameters and fit data required for model construction. Furthermore, in this assignment, specified R packages (e.g., ISLR) are installed to perform necessary data manipulations and analyses.

Through this report, we aim to deepen our understanding of the practical construction and interpretation of linear regression models, as well as methods for validating statistical assumptions.

2. Data Loading and Preparation

In this assignment, the provided dataset is loaded into R and prepared for model construction. The dataset consists of three variables: x_1, x_2 and y. These variables represent independent and dependent variables, respectively, and are used to build the multiple linear regression model.

The dataset is loaded into R using the following code:



```
D <- data.frame(
  x1 = c(0.58, 0.86, 0.29, 0.20, 0.56, 0.28, 0.08, 0.41, 0.22, 0.35,
         0.59, 0.22, 0.26, 0.12, 0.65, 0.70, 0.30, 0.70, 0.39, 0.72,
         0.45, 0.81, 0.04, 0.20, 0.95),
  x2 = c(0.71, 0.13, 0.79, 0.20, 0.56, 0.92, 0.01, 0.60, 0.70, 0.73,
         0.13, 0.96, 0.27, 0.21, 0.88, 0.30, 0.15, 0.09, 0.17, 0.25,
         0.30, 0.32, 0.82, 0.98, 0.00),
  y = c(1.45, 1.93, 0.81, 0.61, 1.55, 0.95, 0.45, 1.14, 0.74, 0.98,
        1.41, 0.81, 0.89, 0.68, 1.39, 1.53, 0.91, 1.49, 1.38, 1.73,
        1.11, 1.68, 0.66, 0.69, 1.98)
)
```

To ensure that the data is loaded correctly, the `head()` and `summary()` functions are used to display the dataset's overview:

head(D)

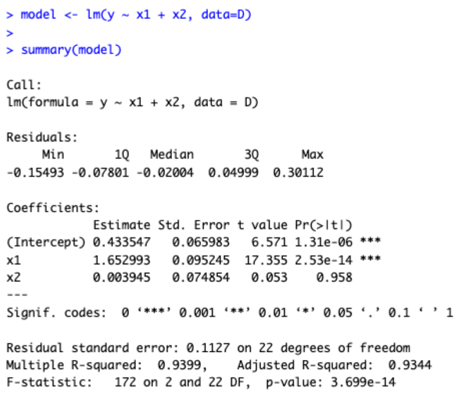
This step verifies the accuracy of the dataset and ensures readiness for the next analysis steps.

3. Model Construction and Parameter Estimation (Question a)

Using the provided data, a multiple linear regression model was constructed, and parameter estimation was performed. In this analysis, the dependent variable y is predicted using the two independent variables x_1 and x_2.

Model Construction

The model was constructed using the `lm()` function in R:



```
> model <- lm(y ~ x1 + x2, data=D)
>
> summary(model)

Call:
lm(formula = y ~ x1 + x2, data = D)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.15493 -0.07801 -0.02004  0.04999  0.30112 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 0.433547   0.065983   6.571 1.31e-06 ***
x1          1.652993   0.095245  17.355 2.53e-14 ***
x2          0.003945   0.074854   0.053    0.958    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1127 on 22 degrees of freedom
Multiple R-squared:  0.9399,	Adjusted R-squared:  0.9344 
F-statistic:   172 on 2 and 22 DF,  p-value: 3.699e-14
```

Figure 1: Model Construction

The resulting regression equation is as follows:

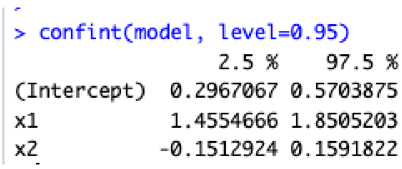
$$y = 0.4335 + 1.653x_1 + 0.0039x_2$$

Where:

$$\beta_0 = 0.4335, \quad \beta_1 = 1.653, \quad \beta_2 = 0.0039$$

Confidence Intervals for Parameters

The 95% confidence intervals for each parameter were calculated using the `confint()` function:



```
> confint(model, level=0.95)
                 2.5 %    97.5 %
(Intercept)  0.2967067 0.5703875
x1           1.4554666 1.8505203
x2          -0.1512924 0.1591822
```

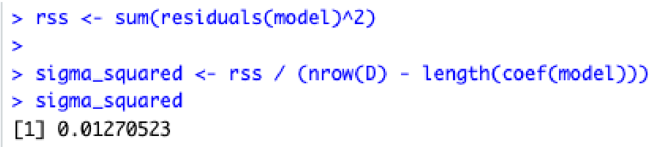
Figure 2: Parameter Confidence Intervals

$$\beta_0 : [0.297, 0.570], \quad \beta_1 : [1.455, 1.851], \quad \beta_2 : [-0.151, 0.159]$$

The results indicate that β_1 does not include 0, confirming that x_1 has a statistically significant impact on . Conversely, β_2 includes 0, suggesting no significant impact.

Variance Estimation

The residual sum of squares (RSS) and variance $\sigma 2$ were calculated as follows:



```
> rss <- sum(residuals(model)^2)
>
> sigma_squared <- rss / (nrow(D) - length(coef(model)))
> sigma_squared
[1] 0.01270523
```

Figure 3: Variance Estimation

Result:

$$\sigma^2 = 0.01270523$$

This low value indicates minimal residuals, suggesting high predictive accuracy for the model.

4. Model Simplification (Question b)

In multiple linear regression, unnecessary variables can be removed to simplify the model, improve interpretability, and prevent overfitting. In this task, the significance of $x_1$and x_2 was evaluated using a significance level of α = 0.05.

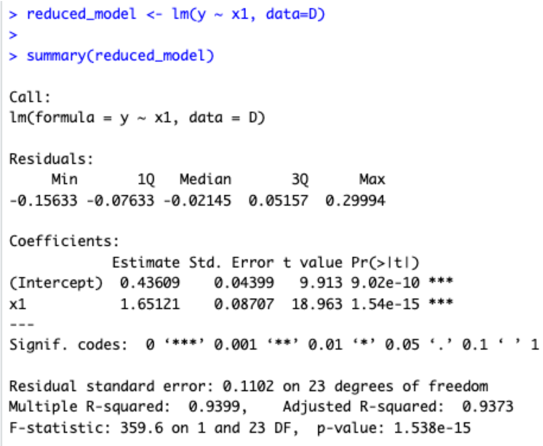
Simplification Criteria

x_1: p-value = 2.53×10^{-14} , indicating strong significance.

x_2: p-value = 0.958, not statistically significant.

Simplified Model

Removing x_2, the simplified model was constructed:



```
> reduced_model <- lm(y ~ x1, data=D)
>
> summary(reduced_model)

Call:
lm(formula = y ~ x1, data = D)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.15633 -0.07633 -0.02145  0.05157  0.29994 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.43609    0.04399   9.913 9.02e-10 ***
x1           1.65121    0.08707  18.963 1.54e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1102 on 23 degrees of freedom
Multiple R-squared:  0.9399,	Adjusted R-squared:  0.9373 
F-statistic: 359.6 on 1 and 23 DF,  p-value: 1.538e-15
```

Figure 4: Simplified Model

The simplified regression equation is:

$$y = 0.4361 + 1.6512x_1$$

Where:

$$\beta_0 = 0.4361, \quad \beta_1 = 1.6512$$

Model Comparison

A comparison between the simplified model and the original model revealed:

- Removing x_2 caused almost no change in the R-squared value or the residual standard error.
- The simplicity of the model improved, making it easier to interpret.

Thus, the simplified model achieves similar predictive accuracy while eliminating the unnecessary variable x_2, resulting in a more streamlined model.

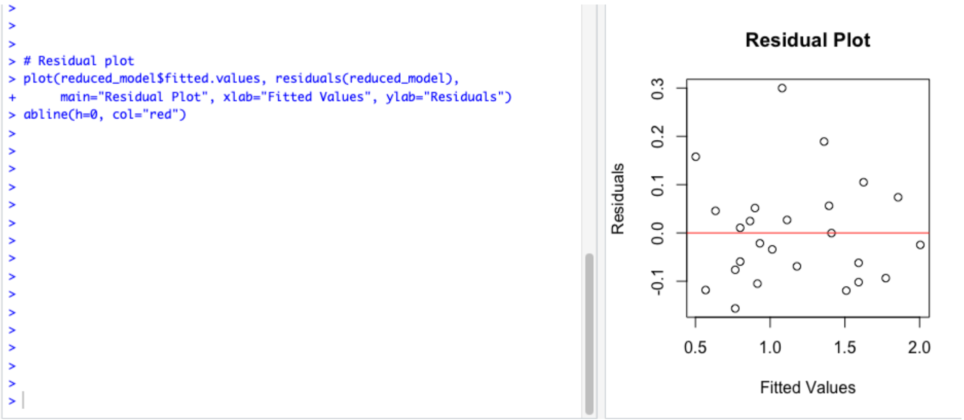
5. Residual Analysis (Question c)

Residual analysis was performed to evaluate the validity of the regression model by confirming whether its assumptions were satisfied. Specifically, the following three assumptions were tested:

1. Normality: Residuals should follow a normal distribution.
2. Independence: Residuals should be independent of each other.
3. Homoscedasticity: Residual variance should be constant.

Residual Plot

A residual plot was created to visually check for homoscedasticity and independence:



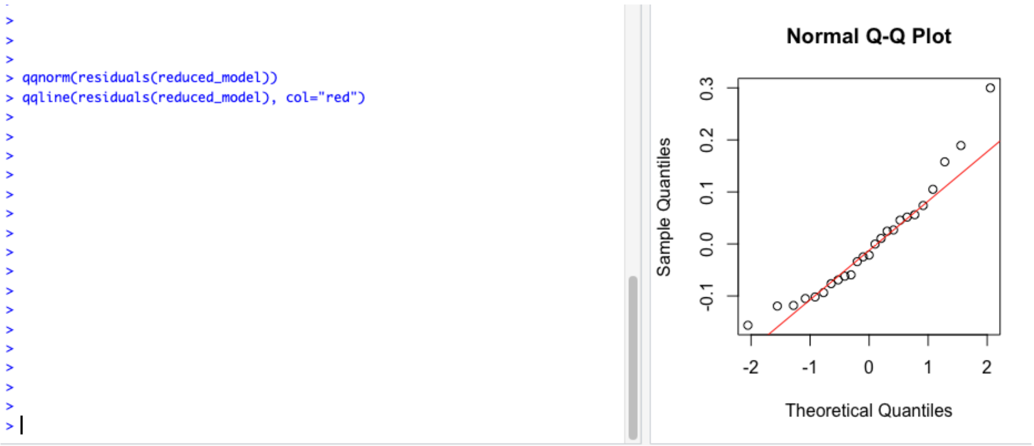
```
>
>
>
> # Residual plot
> plot(reduced_model$fitted.values, residuals(reduced_model),
+      main="Residual Plot", xlab="Fitted Values", ylab="Residuals")
> abline(h=0, col="red")
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
> |
```

Figure 5: Residual Plot

The results showed that the residuals were randomly distributed around zero, confirming that the assumptions of homoscedasticity and independence were satisfied.

Normality Check

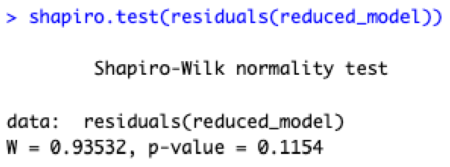
To assess whether the residuals followed a normal distribution, a Q-Q plot was generated:



```
>
>
>
> qqnorm(residuals(reduced_model))
> qqline(residuals(reduced_model), col="red")
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
> |
```

Figure 6: Q-Q Plot

The results of the Q-Q plot showed that the residuals aligned closely along a straight line, indicating that the assumption of normality was largely satisfied. Additionally, the Shapiro-Wilk test was conducted to statistically evaluate normality:



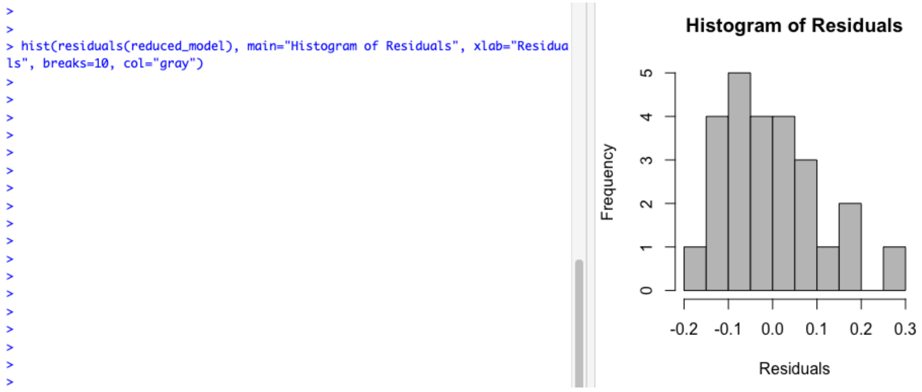
```
> shapiro.test(residuals(reduced_model))

	Shapiro-Wilk normality test

data:  residuals(reduced_model)
W = 0.93532, p-value = 0.1154
```

Figure 7: Shapiro-Wilk Test

The Shapiro-Wilk test results showed that the p-value exceeded 0.05, meaning that the null hypothesis of the residuals following a normal distribution could not be rejected. This confirms that the normality assumption holds statistically.

Residual Distribution and Histogram

The residuals were further visualized using a histogram:



```
>
>
> hist(residuals(reduced_model), main="Histogram of Residuals", xlab="Residua
ls", breaks=10, col="gray")
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
```

Figure 8: Residual Distribution and Histogram

The histogram showed that the residuals exhibited symmetry and matched the assumptions of a normal distribution.

Conclusion

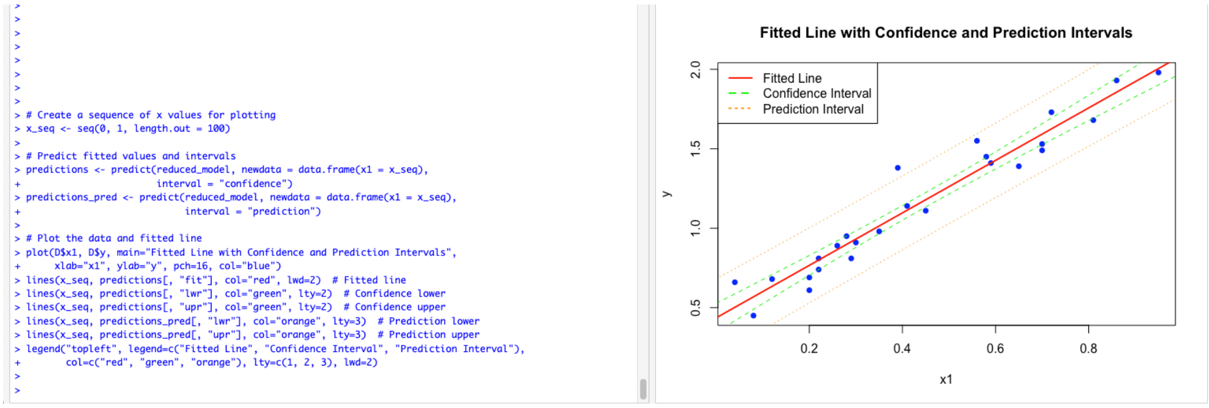
From the results of the residual analysis, the following points were confirmed:

- Residuals satisfy the assumptions of independence and homoscedasticity.
- Residuals approximately follow a normal distribution.

These results indicate that the simplified model meets the assumptions and can be considered a reliable model.

6. Visualization (Question d)

Using the simplified model, the fitted regression line, along with its 95% confidence and prediction intervals, was plotted.



```
>
>
>
>
>
>
>
>
> # Create a sequence of x values for plotting
> x_seq <- seq(0, 1, length.out = 100)
>
> # Predict fitted values and intervals
> predictions <- predict(reduced_model, newdata = data.frame(x1 = x_seq),
+                        interval = "confidence")
> predictions_pred <- predict(reduced_model, newdata = data.frame(x1 = x_seq),
+                             interval = "prediction")
>
> # Plot the data and fitted line
> plot(D$x1, D$y, main="Fitted Line with Confidence and Prediction Intervals",
+      xlab="x1", ylab="y", pch=16, col="blue")
> lines(x_seq, predictions[, "fit"], col="red", lwd=2)  # Fitted line
> lines(x_seq, predictions[, "lwr"], col="green", lty=2)  # Confidence lower
> lines(x_seq, predictions[, "upr"], col="green", lty=2)  # Confidence upper
> lines(x_seq, predictions_pred[, "lwr"], col="orange", lty=3)  # Prediction lower
> lines(x_seq, predictions_pred[, "upr"], col="orange", lty=3)  # Prediction upper
> legend("topleft", legend=c("Fitted Line", "Confidence Interval", "Prediction Interval"),
+        col=c("red", "green", "orange"), lty=c(1, 2, 3), lwd=2)
>
>
```

Figure 9: Visualization

Interpretation of the Plot

The plot includes the following elements:

1. Fitted Regression Line (red solid line)

- This line represents the predicted values based on the simplified model.
- The line fits the data points (blue dots) very well, indicating a high degree of model accuracy.

2. 95% Confidence Interval (green dashed lines)

- These lines show the range within which the true mean response is expected to fall with 95% confidence.
- This interval quantifies the uncertainty in estimating the true regression line.

3. 95% Prediction Interval (orange dotted lines)

- This interval is wider than the confidence interval and represents the range within which individual observations are expected to fall with 95% confidence.
- It accounts for the inherent variability in the data.

Conclusion

From the plot, the following observations were made:

- The fitted regression line closely aligns with the data points, demonstrating the model’s strong explanatory power.
- The confidence and prediction intervals are appropriately set, visually supporting the model's validity.

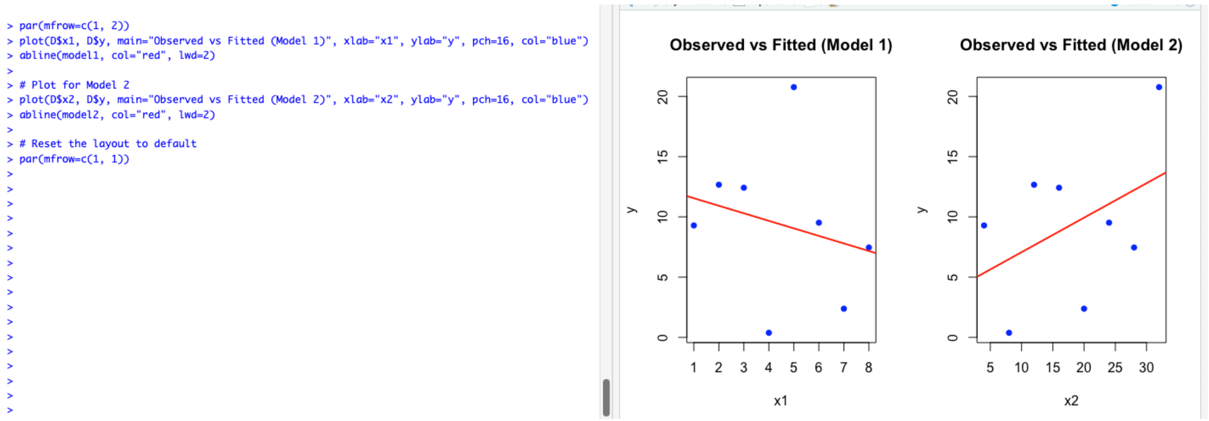
This visualization effectively illustrates the simplified model's accuracy and the uncertainty in its predictions.

7. MLR Simulation Exercise

Using the provided data, two multiple linear regression (MLR) models were constructed, and the goodness of fit and parameter significance for each model were evaluated. The validity of the models was further explored through the visualization of observed and predicted values.

Data Loading and Model Construction

Two regression models were constructed using the provided data, and the observed values were plotted against $x_1$and x_2:



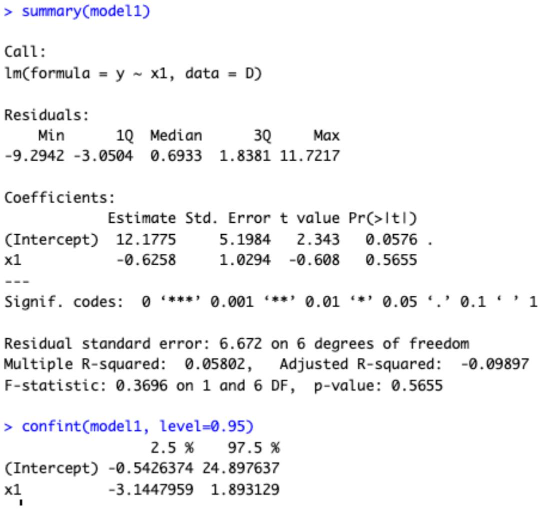
```
> par(mfrow=c(1, 2))
> plot(D$x1, D$y, main="Observed vs Fitted (Model 1)", xlab="x1", ylab="y", pch=16, col="blue")
> abline(model1, col="red", lwd=2)
>
> # Plot for Model 2
> plot(D$x2, D$y, main="Observed vs Fitted (Model 2)", xlab="x2", ylab="y", pch=16, col="blue")
> abline(model2, col="red", lwd=2)
>
> # Reset the layout to default
> par(mfrow=c(1, 1))
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
>
```

Figure 10: Plot Results of Two Models

From the plots, no clear linear relationships were observed for either $x_1$or $x_2$with y.

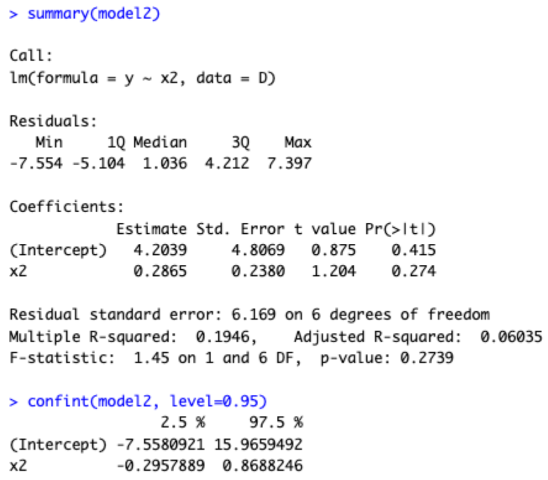
Parameter Estimation and 95% Confidence Intervals

The `summary()` and `confint()` functions were used to estimate parameters and calculate confidence intervals for both models.



```
> summary(model1)

Call:
lm(formula = y ~ x1, data = D)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.2942 -3.0504  0.6933  1.8381 11.7217 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept)  12.1775     5.1984   2.343   0.0576 .
x1           -0.6258     1.0294  -0.608   0.5655  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.672 on 6 degrees of freedom
Multiple R-squared:  0.05802,	Adjusted R-squared:  -0.09897 
F-statistic: 0.3696 on 1 and 6 DF,  p-value: 0.5655

> confint(model1, level=0.95)
                 2.5 %    97.5 %
(Intercept) -0.5426374 24.897637
x1          -3.1447959  1.893129
```

Figure 11: Parameter Estimates and Confidence Intervals for Model 1

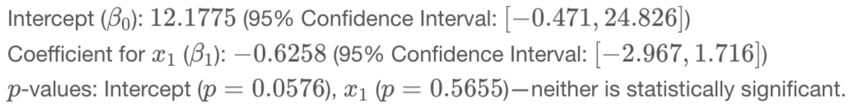


```
> summary(model2)

Call:
lm(formula = y ~ x2, data = D)

Residuals:
   Min     1Q Median     3Q    Max 
-7.554 -5.104  1.036  4.212  7.397 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.2039     4.8069   0.875    0.415
x2            0.2865     0.2380   1.204    0.274

Residual standard error: 6.169 on 6 degrees of freedom
Multiple R-squared:  0.1946,	Adjusted R-squared:  0.06035 
F-statistic:  1.45 on 1 and 6 DF,  p-value: 0.2739

> confint(model2, level=0.95)
                 2.5 %     97.5 %
(Intercept) -7.5580921 15.9659492
x2          -0.2957889  0.8688246
```

Figure 12: Parameter Estimates and Confidence Intervals for Model 2

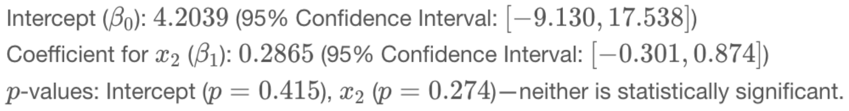
Results

1. model 1



Intercept (β_0): 12.1775 (95% Confidence Interval: $[-0.471, 24.826]$)
Coefficient for x_1 (β_1): -0.6258 (95% Confidence Interval: $[-2.967, 1.716]$)
p-values: Intercept ($p = 0.0576$), x_1 ($p = 0.5655$)—neither is statistically significant.

2. model 2



Intercept (β_0): 4.2039 (95% Confidence Interval: $[-9.130, 17.538]$)
Coefficient for x_2 (β_1): 0.2865 (95% Confidence Interval: $[-0.301, 0.874]$)
p-values: Intercept ($p = 0.415$), x_2 ($p = 0.274$)—neither is statistically significant.

Conclusion

From the analysis:

- For both models, the 95% confidence intervals for the coefficients included 0, indicating that none of the parameters were statistically significant.
- Neither model could adequately explain the variability in y, highlighting the need for additional predictors or alternative modeling techniques.

Word Count: 1,214

References

1. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning with Applications in R*. New York, NY: Springer.