

Report Inferential Statistic Part 1

Malik Haroon

Advanced Statistical Programming (ACCA7022)

```

=====
> cat("PART 1: SAMPLING DISTRIBUTIONS AND CLT\n")
PART 1: SAMPLING DISTRIBUTIONS AND CLT
> cat("=====\\n\\n")
=====

>
> # Extract life expectancy data
> life_data <- gapminder$lifeExp
> cat("Life expectancy data extracted. Range:", min(life_data), "to", max(life_data), "years\\n")
Life expectancy data extracted. Range: 23.599 to 82.603 years
> cat("Mean:", round(mean(life_data), 2), "years\\n")
Mean: 59.47 years
> cat("Standard deviation:", round(sd(life_data), 2), "years\\n\\n")
Standard deviation: 12.92 years

>
> # Generate 1000 sample means (n = 30)
> set.seed(123) # For reproducibility
> sample_means <- replicate(1000, mean(sample(life_data, size = 30, replace = TRUE)))
>
> # Visualise sampling distribution
> hist(sample_means,
+   main = "Sampling Distribution of Sample Mean (n=30)",
+   xlab = "Mean Life Expectancy",
+   col = "lightblue",
+   border = "white",
+   probability = TRUE)
> curve(dnorm(x, mean = mean(sample_means), sd = sd(sample_means)),
+   add = TRUE, col = "red", lwd = 2)
> cat("Sampling distribution (n=30): Mean =", round(mean(sample_means), 2),
+   ", SE =", round(sd(sample_means), 2), "\\n\\n")
Sampling distribution (n=30): Mean = 59.55 , SE = 2.41

>
> # Effect of sample size - compare n = 10, 30, and 100
> means_10 <- replicate(1000, mean(sample(life_data, 10, replace = TRUE)))
> means_30 <- replicate(1000, mean(sample(life_data, 30, replace = TRUE)))
> means_100 <- replicate(1000, mean(sample(life_data, 100, replace = TRUE)))
>
> # Set up plotting area for comparison
> par(mfrow = c(1,3))
> hist(means_10, main = "n = 10", xlab = "Mean Life Expectancy",
+   col = "lightgreen", xlim = c(40, 70))

```

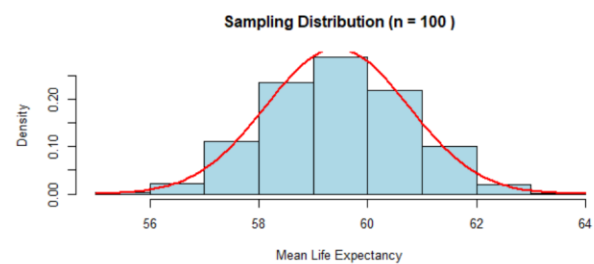
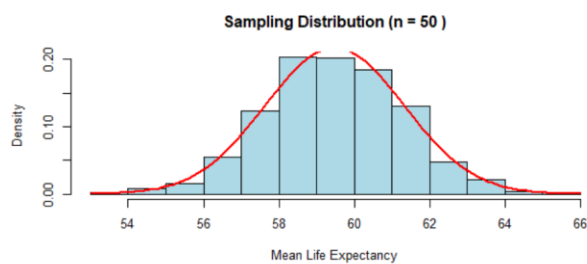
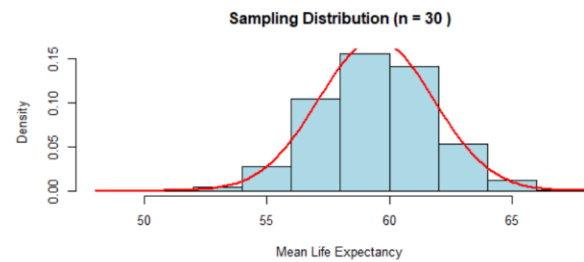
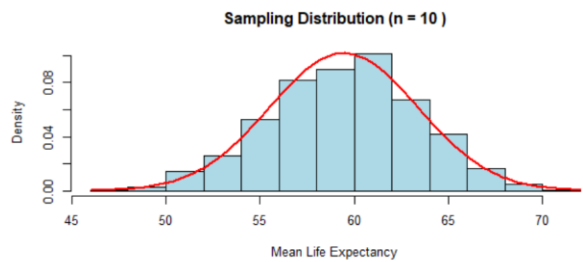
```

> hist(means_30, main = "n = 30", xlab = "Mean Life Expectancy",
+   col = "lightblue", xlim = c(40, 70))
> hist(means_100, main = "n = 100", xlab = "Mean Life Expectancy",
+   col = "lightcoral", xlim = c(40, 70))
> par(mfrow = c(1,1))
>
> cat("Effect of Sample Size on Standard Error:\n")
Effect of Sample Size on Standard Error:
> cat("n = 10: SE =", round(sd(means_10), 3), "\n")
n = 10: SE = 4.089
> cat("n = 30: SE =", round(sd(means_30), 3), "\n")
n = 30: SE = 2.44
> cat("n = 100: SE =", round(sd(means_100), 3), "\n\n")
n = 100: SE = 1.264

>
> # Activity - Sampling Distribution Exploration
> # Create function to simulate sampling means
> simulate_means <- function(n, reps = 1000) {
+   replicate(reps, mean(sample(life_data, size = n, replace = TRUE)))
+ }
>
> # Test different sample sizes
> sample_sizes <- c(10, 30, 50, 100)
> par(mfrow = c(2,2))
>
> for (n in sample_sizes) {
+   sim_means <- simulate_means(n)
+   hist(sim_means,
+     main = paste("Sampling Distribution (n =", n, ")"),
+     xlab = "Mean Life Expectancy",
+     col = "lightblue",
+     probability = TRUE)
+   curve(dnorm(x, mean = mean(sim_means), sd = sd(sim_means)),
+     add = TRUE, col = "red", lwd = 2)
+ }
>
> par(mfrow = c(1,1))
>
> # Calculate standard errors for each sample size
> se_results <- data.frame(
+   sample_size = sample_sizes,
+   standard_error = sapply(sample_sizes, function(n) sd(simulate_means(n)))
+ )

```

```
> print(se_results)
  sample_size standard_error
1      10      4.095846
2      30      2.419372
3      50      1.883793
4     100      1.290601
> cat("\n\n")
```



```
> cat("=====\n")
=====\n
> cat("PART 2: CORRELATION ANALYSIS\n")
PART 2: CORRELATION ANALYSIS
> cat("=====\n\n")
=====\n\n
```

```
>
> # Explore the dataset
> cat("First few rows of Gapminder dataset:\n")
First few rows of Gapminder dataset:
> print(head(gapminder))
# A tibble: 6 × 6
  country  continent year lifeExp  pop gdpPercap
<fct>    <fct>    <int> <dbl> <int> <dbl>
1 Afghanistan Asia    1952  28.8 8425333  779.
2 Afghanistan Asia    1957  30.3 9240934  821.
3 Afghanistan Asia    1962  32.0 10267083  853.
4 Afghanistan Asia    1967  34.0 11537966  836.
```

```

5 Afghanistan Asia 1972 36.1 13079460 740.
6 Afghanistan Asia 1977 38.4 14880372 786.
> cat("\nSummary statistics:\n")

Summary statistics:
> print(summary(gapminder))
      country continent  year  lifeExp
Afghanistan: 12 Africa :624 Min. :1952 Min. :23.60
Albania : 12 Americas:300 1st Qu.:1966 1st Qu.:48.20
Algeria : 12 Asia :396 Median :1980 Median :60.71
Angola : 12 Europe :360 Mean :1980 Mean :59.47
Argentina : 12 Oceania : 24 3rd Qu.:1993 3rd Qu.:70.85
Australia : 12 Max. :2007 Max. :82.60
(Other) :1632
      pop      gdpPercap
Min. :6.001e+04 Min. : 241.2
1st Qu.:2.794e+06 1st Qu.: 1202.1
Median :7.024e+06 Median : 3531.8
Mean :2.960e+07 Mean : 7215.3
3rd Qu.:1.959e+07 3rd Qu.: 9325.5
Max. :1.319e+09 Max. :113523.1

> cat("\n")

>
> # Visualize relationship between GDP and life expectancy
> ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
+ geom_point(alpha = 0.5, color = "darkblue") +
+ geom_smooth(method = "lm", color = "red", se = TRUE) +
+ labs(title = "Relationship between GDP per Capita and Life Expectancy",
+ x = "GDP per Capita", y = "Life Expectancy") +
+ theme_minimal()
`geom_smooth()` using formula = 'y ~ x'
>
> # Pearson correlation (linear relationship)
> pearson_cor <- cor(gapminder$lifeExp, gapminder$gdpPercap, method = "pearson")
> cat("Pearson correlation between lifeExp and gdpPercap:", round(pearson_cor, 4), "\n")
Pearson correlation between lifeExp and gdpPercap: 0.5837
> cat("Interpretation: Moderate positive linear relationship\n\n")
Interpretation: Moderate positive linear relationship

>
> # Visualize with regression line using ggpubr
> ggscatter(gapminder, x = "gdpPercap", y = "lifeExp",

```

```

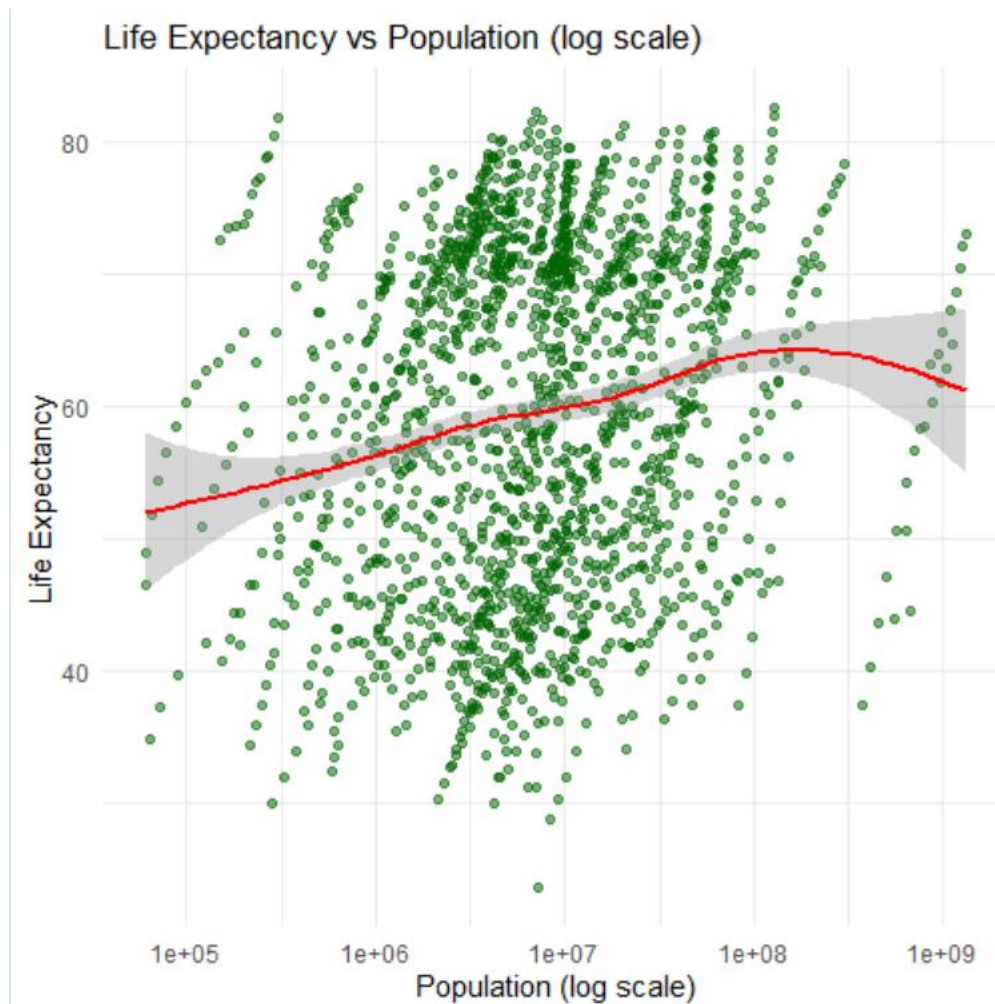
+   add = "reg.line", conf.int = TRUE,
+   cor.coef = TRUE, cor.method = "pearson",
+   title = "Life Expectancy vs GDP per Capita (Pearson)")
>
> # Spearman correlation (rank-based)
> spearman_cor <- cor(gapminder$lifeExp, gapminder$gdpPerCap, method = "spearman")
> cat("Spearman correlation between lifeExp and gdpPerCap:", round(spearman_cor, 4), "\n")
Spearman correlation between lifeExp and gdpPerCap: 0.8265
> cat("Interpretation: Strong monotonic relationship\n\n")
Interpretation: Strong monotonic relationship

>
> # Activity: Compare Pearson vs Spearman for lifeExp vs pop
> pearson_pop <- cor(gapminder$lifeExp, gapminder$pop, method = "pearson")
> spearman_pop <- cor(gapminder$lifeExp, gapminder$pop, method = "spearman")
>
> cat("\nComparing correlation methods for lifeExp vs population:\n")

Comparing correlation methods for lifeExp vs population:
> cat("Pearson correlation:", round(pearson_pop, 4), "\n")
Pearson correlation: 0.065
> cat("Spearman correlation:", round(spearman_pop, 4), "\n")
Spearman correlation: 0.1806
> cat("Spearman is more appropriate because population data is highly skewed.\n\n")
Spearman is more appropriate because population data is highly skewed.

>
> # Visualize the skewed relationship
> ggplot(gapminder, aes(x = pop, y = lifeExp)) +
+   geom_point(alpha = 0.5, color = "darkgreen") +
+   scale_x_log10() + # Log transform for better visualization
+   geom_smooth(method = "loess", color = "red") +
+   labs(title = "Life Expectancy vs Population (log scale)",
+        x = "Population (log scale)", y = "Life Expectancy") +
+   theme_minimal()
`geom_smooth()` using formula = 'y ~ x'
>
> cat("\n")

```



```

> cat("=====\n")
=====
> cat("PART 3: T-TESTS\n")
PART 3: T-TESTS
> cat("=====\n\n")
=====

>
> # Create boxplot comparing lifeExp in Europe vs Africa
> p <- ggplot(gapminder %>% filter(continent %in% c("Europe", "Africa")),
+   aes(x = continent, y = lifeExp, fill = continent)) +
+   geom_boxplot() +
+   labs(title = "Life Expectancy: Europe vs Africa",
+     y = "Life Expectancy (years)") +
+   theme_minimal() +
+   scale_fill_manual(values = c("Africa" = "orange", "Europe" = "blue"))

```

```

> print(p)
>
> # Extract data for t-test
> europe <- gapminder %>% filter(continent == "Europe") %>% pull(lifeExp)
> africa <- gapminder %>% filter(continent == "Africa") %>% pull(lifeExp)
>
> # Perform two-sample t-test
> t_test_result <- t.test(europe, africa)
> print(t_test_result)

      Welch Two Sample t-test

data:  europe and africa
t = 49.551, df = 981.2, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 22.12595 23.95076
sample estimates:
mean of x mean of y
 71.90369 48.86533

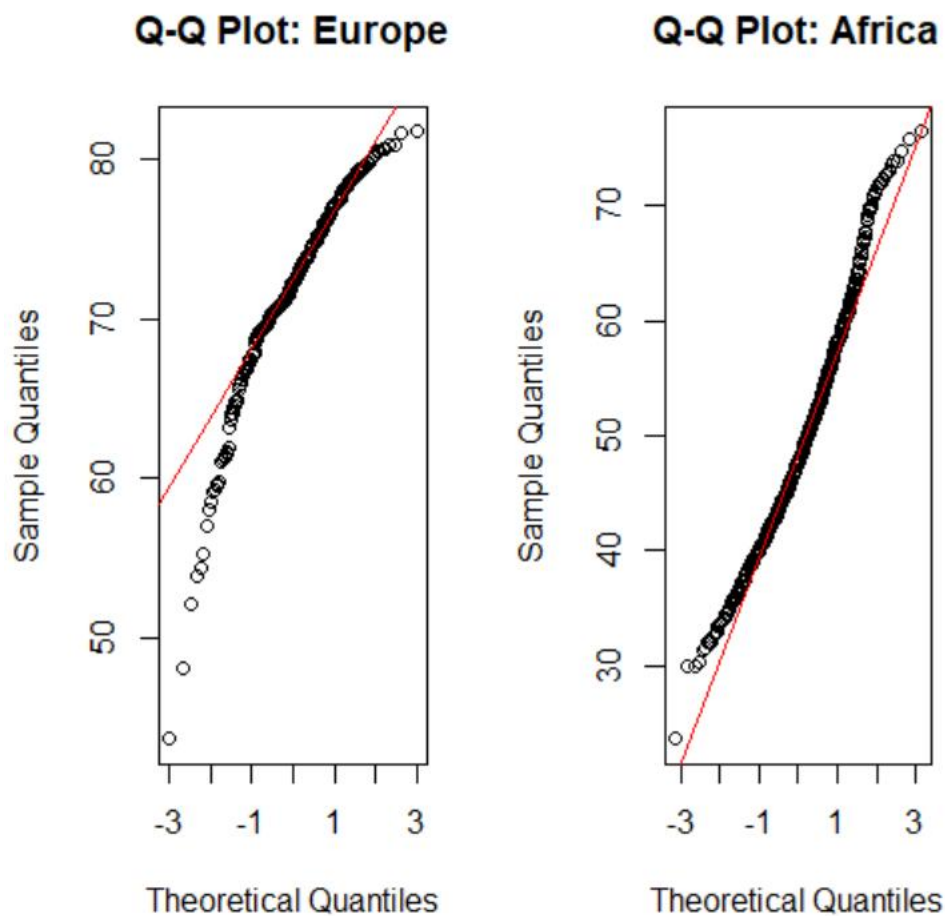
>
> # Interpretation summary
> cat("\n===== T-TEST INTERPRETATION =====\n")

===== T-TEST INTERPRETATION =====
> cat("Test Statistic (t):", round(t_test_result$statistic, 3), "\n")
Test Statistic (t): 49.551
> cat("Degrees of Freedom (df):", round(t_test_result$parameter, 3), "\n")
Degrees of Freedom (df): 981.202
> cat("p-value:", format(t_test_result$p.value, scientific = TRUE), "\n")
p-value: 2.622716e-269
> cat("95% Confidence Interval:", paste(round(t_test_result$conf.int, 2), collapse = " to "),
"\n")
95% Confidence Interval: 22.13 to 23.95
> cat("Europe mean:", round(t_test_result$estimate[1], 2), "years\n")
Europe mean: 71.9 years
> cat("Africa mean:", round(t_test_result$estimate[2], 2), "years\n")
Africa mean: 48.87 years
> cat("Mean difference:", round(t_test_result$estimate[1] - t_test_result$estimate[2], 2),
"years\n")
Mean difference: 23.04 years
> cat("Conclusion: p < 0.05 → Reject H0. Europe's mean life expectancy is significantly
higher.\n\n")

```

Conclusion: $p < 0.05 \rightarrow$ Reject H_0 . Europe's mean life expectancy is significantly higher.

```
>
> # Check normality assumption (for t-test)
> par(mfrow = c(1,2))
> qqnorm(europe, main = "Q-Q Plot: Europe")
> qqline(europe, col = "red")
> qqnorm(africa, main = "Q-Q Plot: Africa")
> qqline(africa, col = "red")
> par(mfrow = c(1,1))
>
> cat("\n")
```



```
> cat("=====\n")
====
> cat("PART 4: SIMPLE LINEAR REGRESSION\n")
PART 4: SIMPLE LINEAR REGRESSION
```

```

> cat("=====\n\n")
=====

>
> # Simple linear regression: lifeExp ~ gdpPercap
> model_simple <- lm(lifeExp ~ gdpPercap, data = gapminder)
> cat("SIMPLE LINEAR REGRESSION MODEL:\n")
SIMPLE LINEAR REGRESSION MODEL:
> cat("lifeExp =  $\beta_0$  +  $\beta_1$  * gdpPercap\n\n")
lifeExp =  $\beta_0$  +  $\beta_1$  * gdpPercap

> print(summary(model_simple))

Call:
lm(formula = lifeExp ~ gdpPercap, data = gapminder)

Residuals:
    Min     1Q   Median     3Q      Max
-82.754 -7.758  2.176  8.225 18.426

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.396e+01  3.150e-01  171.29  <2e-16 ***
gdpPercap   7.649e-04  2.579e-05   29.66  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.49 on 1702 degrees of freedom
Multiple R-squared:  0.3407,    Adjusted R-squared:  0.3403
F-statistic: 879.6 on 1 and 1702 DF,  p-value: < 2.2e-16

>
> # Extract coefficients
> coef_simple <- coef(model_simple)
> cat("\n===== SIMPLE REGRESSION RESULTS =====\n")

===== SIMPLE REGRESSION RESULTS =====
> cat("Intercept ( $\beta_0$ ):", round(coef_simple[1], 4), "\n")
Intercept ( $\beta_0$ ): 53.9556
> cat("Interpretation: When gdpPercap is $0, predicted life expectancy is",
round(coef_simple[1], 2), "years\n")
Interpretation: When gdpPercap is $0, predicted life expectancy is 53.96 years
> cat("(Note: Extrapolation beyond data range - GDP = $0 is unrealistic)\n\n")
(Note: Extrapolation beyond data range - GDP = $0 is unrealistic)

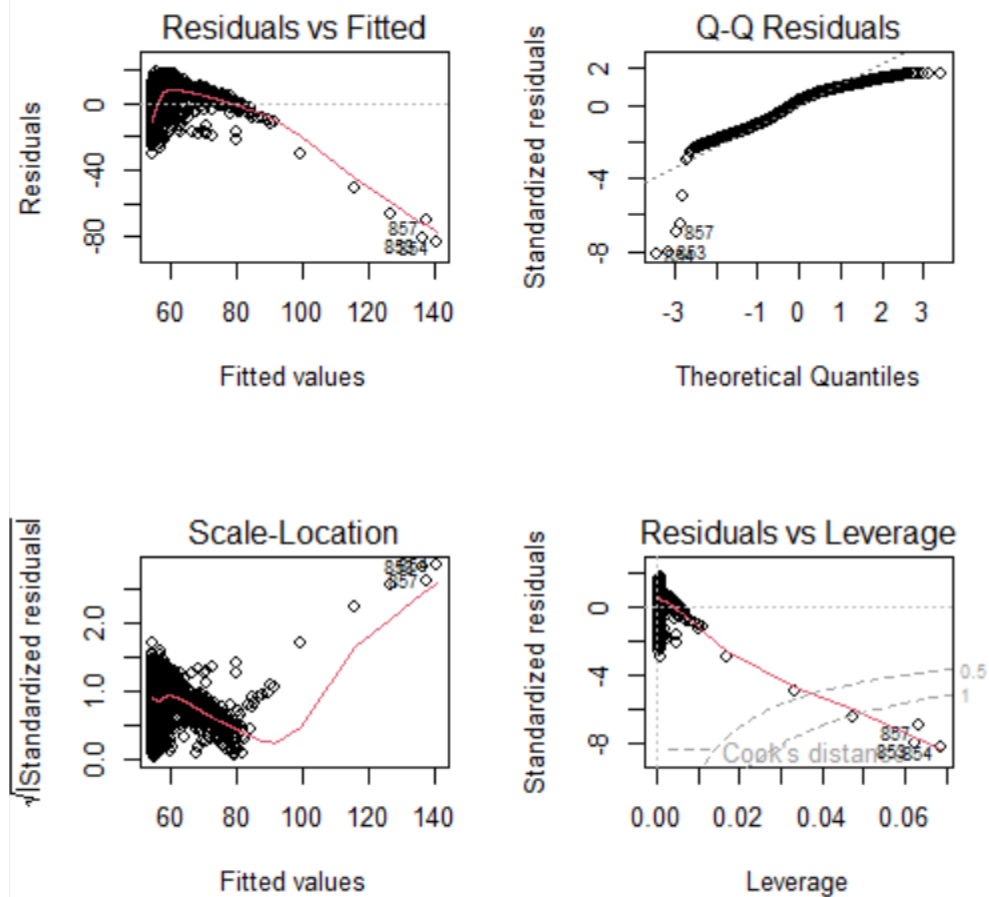
```

```

> cat("Slope ( $\beta_1$ ):", round(coef_simple[2], 8), "\n")
Slope ( $\beta_1$ ): 0.00076488
> cat("Interpretation: For every $1,000 increase in GDP per capita,\n")
Interpretation: For every $1,000 increase in GDP per capita,
> cat("life expectancy increases by ~", round(coef_simple[2] * 1000, 2), "years\n")
life expectancy increases by ~ 0.76 years
> cat("Statistical significance: p < 2.2e-16 ***\n\n")
Statistical significance: p < 2.2e-16 ***

>
> # Diagnostic plots for simple regression
> par(mfrow = c(2,2))
> plot(model_simple)
> par(mfrow = c(1,1))
>
> cat("\n")

```



```
> cat("=====\n")
=====
> cat("PART 5: MULTIPLE REGRESSION\n")
PART 5: MULTIPLE REGRESSION
> cat("=====\n\n")
=====

>
> # Multiple regression with additional predictors
> model_multi <- lm(lifeExp ~ gdpPercap + year + continent, data = gapminder)
> cat("MULTIPLE LINEAR REGRESSION MODEL:\n")
MULTIPLE LINEAR REGRESSION MODEL:
> cat("lifeExp =  $\beta_0 + \beta_1 \cdot \text{gdpPercap} + \beta_2 \cdot \text{year} + \beta_3 \cdot \text{continent} + \epsilon$ \n\n")
lifeExp =  $\beta_0 + \beta_1 \cdot \text{gdpPercap} + \beta_2 \cdot \text{year} + \beta_3 \cdot \text{continent} + \epsilon$ 

> print(summary(model_multi))
```

Call:

```
lm(formula = lifeExp ~ gdpPercap + year + continent, data = gapminder)
```

Residuals:

Min	1Q	Median	3Q	Max
-28.4264	-4.0725	0.2154	4.4853	19.9977

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-5.207e+02	1.979e+01	-26.31	<2e-16 ***
gdpPercap	2.967e-04	1.996e-05	14.87	<2e-16 ***
year	2.874e-01	1.000e-02	28.73	<2e-16 ***
continentAmericas	1.433e+01	4.936e-01	29.03	<2e-16 ***
continentAsia	9.506e+00	4.567e-01	20.81	<2e-16 ***
continentEurope	1.940e+01	5.173e-01	37.49	<2e-16 ***
continentOceania	2.059e+01	1.469e+00	14.01	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.884 on 1697 degrees of freedom

Multiple R-squared: 0.717, Adjusted R-squared: 0.716

F-statistic: 716.6 on 6 and 1697 DF, p-value: < 2.2e-16

>

> # Compare models

```
> cat("\n===== MODEL COMPARISON =====\n")
```

===== MODEL COMPARISON =====

```
> cat("Simple Model R-squared:", round(summary(model_simple)$r.squared, 4), "\n")
```

Simple Model R-squared: 0.3407

```
> cat("Multiple Model R-squared:", round(summary(model_multi)$r.squared, 4), "\n")
```

Multiple Model R-squared: 0.717

```
> cat("Improvement:", round(summary(model_multi)$r.squared -  
summary(model_simple)$r.squared, 4), "\n")
```

Improvement: 0.3763

```
> cat("Multiple model explains", round(summary(model_multi)$r.squared * 100, 1),  
+ "% of variance in life expectancy\n\n")
```

Multiple model explains 71.7 % of variance in life expectancy

>

> # Add predictions to dataframe

```
> gapminder_with_pred <- gapminder
```

```
> gapminder_with_pred$predicted <- predict(model_multi)
```

```

>
> # Plot actual vs predicted
> ggplot(gapminder_with_pred, aes(x = predicted, y = lifeExp, color = continent)) +
+   geom_point(alpha = 0.5) +
+   geom_abline(intercept = 0, slope = 1, color = "red", linetype = "dashed") +
+   labs(title = "Actual vs Predicted Life Expectancy",
+         x = "Predicted Values", y = "Actual Values") +
+   theme_minimal() +
+   coord_fixed()
>
> cat("\n")

```

