Implementation Lab: Linear Regression & Experimental Analysis

90-Minute Session Guide

Instructor Guide

Table of contents

1 Lab Overview

Duration: 90 minutes
Format: Live coding demonstration + guided exercise

Datasets: All created via simulation code below

1.1 Learning Objectives

By the end of this lab, students will be able to:

- Implement simple and multiple linear regression in R
- Understand omitted variable bias through practical examples
- Perform log transformations for exponential relationships
- Conduct t-tests and ANOVA for experimental data
- Calculate and interpret effect sizes
- Check key statistical assumptions

1.2 Session Structure

Time	Section	Duration
0:00-0:05	Introduction & Setup	5 min
0:05-0:35	Linear Regression	30 min
0:35-0:70	Experimental Analysis	35 min
0:70-0:90	Guided Exercise	20 min

---

2 Preparation: Data Generation

Run this code before the lab to create all datasets.

```
# Load required packages
library(ggplot2)
library(dplyr)
library(broom)
library(effectsize)
library(car)

# Set seed for reproducibility
set.seed(42)
```

2.1 Dataset 1: Marketing Data (Regression)

Purpose: Demonstrate simple vs. multiple regression and omitted variable bias

```r
# Generate marketing dataset
n <- 100
ad_spend <- runif(n, 10, 100)
website_traffic <- 50 + 0.8 * ad_spend + rnorm(n, 0, 10)
sales_revenue <- 20 + 2.2 * ad_spend + 1.0 * website_traffic + rnorm(n, 0, 15)

marketing_data <- data.frame(
  ad_spend = ad_spend,
  website_traffic = website_traffic,
  sales_revenue = sales_revenue
)

# Save for students
write.csv(marketing_data, "marketing_data.csv", row.names = FALSE)

head(marketing_data)
```

```
  ad_spend website_traffic sales_revenue
1 92.33254       127.08529      349.6064
2 94.33679       117.63104      321.8988
3 35.75256        94.35932      210.5225
4 84.74029       124.22122      326.5452
5 67.75710       105.10328      267.1512
6 56.71864        98.14042      224.3476
```

2.2 Dataset 2: Firm Growth (Log Transformation)

Purpose: Demonstrate exponential growth and log transformation

```r
# Generate exponential growth data
years <- 1:20
revenue <- 50000 * exp(0.12 * years) + rnorm(20, 0, 20000)

firm_growth_data <- data.frame(
  year = years,
  revenue = revenue
)

# Save for students
write.csv(firm_growth_data, "firm_growth_data.csv", row.names = FALSE)

head(firm_growth_data)
```

```
  year   revenue
1    1  34451.72
2    2  64543.47
3    3  47696.55
4    4  84604.10
```

```
5    5 117060.06
6    6  82044.19
```

2.3 Dataset 3: Leadership Training (t-test)

Purpose: Demonstrate independent samples t-test

```r
# Generate between-subjects leadership data
n_per_group <- 30

leadership_study_between <- data.frame(
  participant_id = 1:(2 * n_per_group),
  group = rep(c("control", "training"), each = n_per_group),
  team_performance = c(
    rnorm(n_per_group, mean = 75, sd = 9),  # control
    rnorm(n_per_group, mean = 84, sd = 9)   # training
  )
)

# Save for students
write.csv(leadership_study_between, "leadership_study_between.csv", row.names = FALSE)

head(leadership_study_between)
```

```
  participant_id   group team_performance
1              1 control         80.16377
2              2 control         75.41223
3              3 control         76.41671
4              4 control         78.88409
5              5 control         71.43105
6              6 control         86.78980
```

2.4 Dataset 4: Communication Study (ANOVA)

Purpose: Demonstrate one-way ANOVA with three groups

```r
# Generate communication study data with 3 groups
n_per_group <- 30

communication_study <- data.frame(
  participant_id = 1:(3 * n_per_group),
  communication_method = rep(c("email", "video_call", "face_to_face"), each = n_per_group),
  satisfaction_score = c(
    rnorm(n_per_group, mean = 5.8, sd = 1.3),  # email
    rnorm(n_per_group, mean = 7.0, sd = 1.3),  # video_call
    rnorm(n_per_group, mean = 7.5, sd = 1.3)   # face_to_face
  )
)
```

```r
# Save for students
write.csv(communication_study, "communication_study.csv", row.names = FALSE)

head(communication_study)
```

```
  participant_id communication_method satisfaction_score
1              1                email           6.770803
2              2                email           5.246826
3              3                email           4.796293
4              4                email           5.998593
5              5                email           7.085176
6              6                email           5.704504
```

2.5 Dataset 5: Exercise Dataset

Purpose: Student guided exercise combining regression and t-test

```r
# Generate website A/B test data
n_per_design <- 50

exercise_data <- data.frame(
  user_id = 1:(2 * n_per_design),
  design = rep(c("Simple", "Complex"), each = n_per_design),
  previous_visits = rpois(2 * n_per_design, lambda = 8),
  time_on_site = c(
    rnorm(n_per_design, mean = 180, sd = 40),  # Simple
    rnorm(n_per_design, mean = 240, sd = 50)   # Complex
  )
)

# Conversion rate depends on time_on_site and previous_visits
exercise_data <- exercise_data %>%
  mutate(
    conversion_prob = plogis(-2 + 0.01 * time_on_site + 0.05 * previous_visits),
    converted = rbinom(n(), 1, conversion_prob)
  )

# Save for students
write.csv(exercise_data, "exercise_data.csv", row.names = FALSE)

head(exercise_data)
```

```
  user_id design previous_visits time_on_site conversion_prob converted
1       1 Simple              10    179.63774       0.5735567         1
2       2 Simple              10    121.67466       0.4296563         1
3       3 Simple               4    207.78119       0.5690097         1
4       4 Simple               8     81.54658       0.3133435         1
```

```
5       5 Simple                8    185.73159       0.5639764         1
6       6 Simple                5    164.35112       0.4734029         0
```

---

3 Part 1: Introduction & Setup (5 minutes)

3.1 What to Say

> "Welcome! Today we're focusing on practical implementation of two key analysis methods: linear regression and experimental data analysis. You should have read the tutorials, but don't worry - we'll work through the essential techniques together with hands-on examples."

3.2 What to Do

1. **Have students open RStudio**
2. **Share the data files** (or have them run the generation code)
3. **Load packages together:**

```
# Students run this
library(ggplot2)
library(dplyr)
library(broom)
library(effectsize)
library(car)
```

4. **Quick poll:** "How many of you have the marketing_data loaded successfully?"

---

4 Part 2: Linear Regression (30 minutes)

4.1 2.1 Simple vs. Multiple Regression (10 min)

4.1.1 Teaching Notes

Key Concept: Show how adding variables changes interpretation from "total association" to "direct effect controlling for other variables"

Common Student Misconception: Students think multiple regression just "adds more predictors" - emphasize it changes what each coefficient means!

4.1.2 Live Coding Script

```r
# STEP 1: Simple regression
model_simple <- lm(sales_revenue ~ ad_spend, data = marketing_data)
summary(model_simple)
```

```
Call:
lm(formula = sales_revenue ~ ad_spend, data = marketing_data)

Residuals:
    Min      1Q  Median      3Q     Max
-38.000 -14.155  -1.484  10.909  48.889

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 67.52803    4.06035   16.63   <2e-16 ***
ad_spend     3.03673    0.06417   47.32   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 17.35 on 98 degrees of freedom
Multiple R-squared:  0.9581,    Adjusted R-squared:  0.9576
F-statistic:  2239 on 1 and 98 DF,  p-value: < 2.2e-16
```

Pause and ask: "What does the coefficient for ad_spend mean?"

Expected answer: For every €1 increase in ad spending, sales revenue increases by approximately €{coefficient} on average.

```r
# STEP 2: Multiple regression
model_multiple <- lm(sales_revenue ~ ad_spend + website_traffic,
                     data = marketing_data)
summary(model_multiple)
```

```
Call:
lm(formula = sales_revenue ~ ad_spend + website_traffic, data = marketing_data)

Residuals:
     Min       1Q   Median       3Q      Max
-31.4339  -9.9212  -0.4957   9.7412  31.2228

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      15.7527     7.9513   1.981   0.0504 .
ad_spend          2.1017     0.1407  14.940  < 2e-16 ***
website_traffic   1.1021     0.1540   7.158 1.58e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 14.11 on 97 degrees of freedom
Multiple R-squared:  0.9726,    Adjusted R-squared:  0.972
F-statistic:  1719 on 2 and 97 DF,  p-value: < 2.2e-16
```

Key teaching moment: Extract and compare coefficients

```r
# Extract coefficients for comparison
coef_simple <- coef(model_simple)["ad_spend"]
coef_multiple <- coef(model_multiple)["ad_spend"]

cat("Simple model coefficient:", round(coef_simple, 3), "\n")
```

```
Simple model coefficient: 3.037
```

```r
cat("Multiple model coefficient:", round(coef_multiple, 3), "\n")
```

```
Multiple model coefficient: 2.102
```

```r
cat("Difference (bias):", round(coef_simple - coef_multiple, 3), "\n")
```

```
Difference (bias): 0.935
```

4.1.3 What to Emphasize

> "Notice how the coefficient for ad_spend **changed** from 3.04 to 2.1. This is because the simple model was **confounding** the effects of ad spending and website traffic. The multiple regression gives us the **direct effect** (ceteris paribus effect) of ad spending, controlling for website traffic."

4.1.4 Omitted Variable Bias - Explain Conceptually

Draw on board/slides:

```
Simple model: Sales ~ Ad_spend
Problem: Ad_spend → Website_traffic → Sales
         Ad_spend → Sales

The simple model attributes BOTH effects to ad_spend!

Multiple model: Sales ~ Ad_spend + Website_traffic
Solution: Separates direct effect from indirect effect
```

4.1.5 Check for Understanding

Ask class: "If we ran a simple regression of Sales on Website Traffic only, would we overestimate or underestimate the effect of traffic?"

Answer: Overestimate - because traffic is correlated with ad spending which also affects sales.

4.2 2.2 Model Diagnostics (8 min)

4.2.1 Teaching Notes

Goal: Show students practical workflow for checking model quality **Focus:** R^2 and basic diagnostic plots

4.2.2 Live Coding Script

```
# Calculate R² manually to show what it means
y_actual <- marketing_data$sales_revenue
y_fitted <- predict(model_simple)
y_mean <- mean(y_actual)

# Components
TSS <- sum((y_actual - y_mean)^2)      # Total Sum of Squares
RSS <- sum((y_actual - y_fitted)^2)     # Residual Sum of Squares

# R²
r_squared_manual <- 1 - (RSS / TSS)
r_squared_r <- summary(model_simple)$r.squared

cat("Manual R² calculation:", round(r_squared_manual, 4), "\n")
```

```
Manual R² calculation: 0.9581
```

```
cat("R's R² calculation:", round(r_squared_r, 4), "\n")
```

```
R's R² calculation: 0.9581
```

Interpretation:

> "$R^2 = 0.96$ means our model explains 95.8% of the variation in sales revenue. The remaining 4.2% is unexplained."

4.2.3 Diagnostic Plots

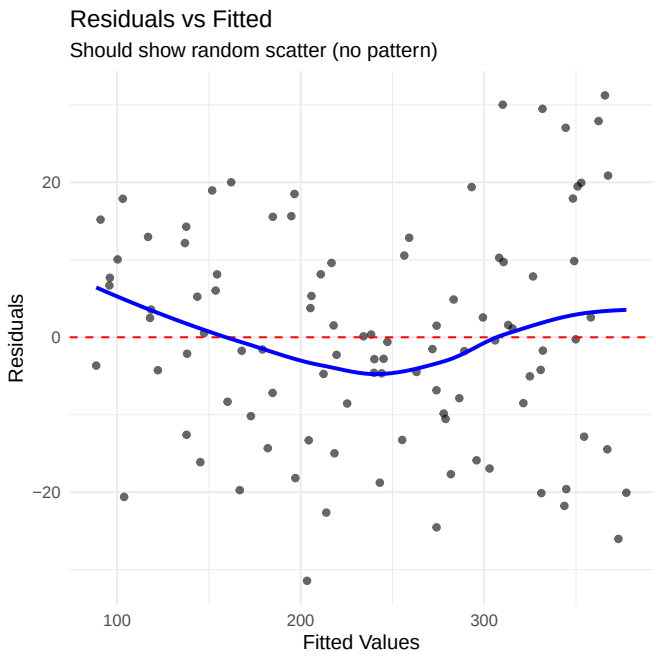
```r
# Create augmented data
model_data <- augment(model_multiple)

# Two key plots
library(gridExtra)
```

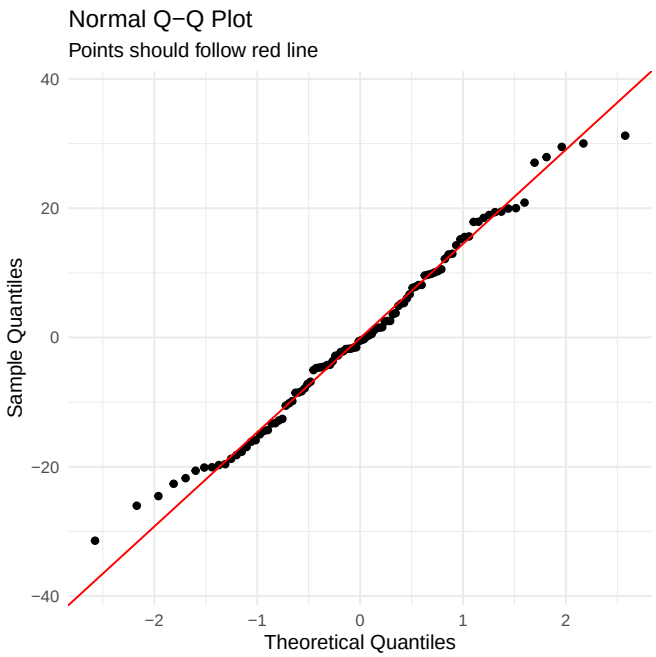
```
Attaching package: 'gridExtra'
```

```
The following object is masked from 'package:dplyr':

    combine
```

```r
p1 <- ggplot(model_data, aes(x = .fitted, y = .resid)) +
  geom_point(alpha = 0.6) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  geom_smooth(se = FALSE, color = "blue") +
  labs(title = "Residuals vs Fitted",
       subtitle = "Should show random scatter (no pattern)",
       x = "Fitted Values",
       y = "Residuals") +
  theme_minimal()

p2 <- ggplot(model_data, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line(color = "red") +
  labs(title = "Normal Q-Q Plot",
       subtitle = "Points should follow red line",
       x = "Theoretical Quantiles",
       y = "Sample Quantiles") +
  theme_minimal()

grid.arrange(p1, p2, ncol = 2)
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```





4.2.4 What to Say

"These diagnostic plots help us check our model assumptions:

- **Left plot:** Residuals vs Fitted - we want random scatter with no clear pattern. A pattern would suggest we're missing something in our model.
- **Right plot:** Q-Q plot - points should follow the red line, indicating residuals are normally distributed."

Quick check: "Do these plots look okay?" (Yes, they do)

4.3 2.3 Log Transformation (12 min)

4.3.1 Teaching Notes

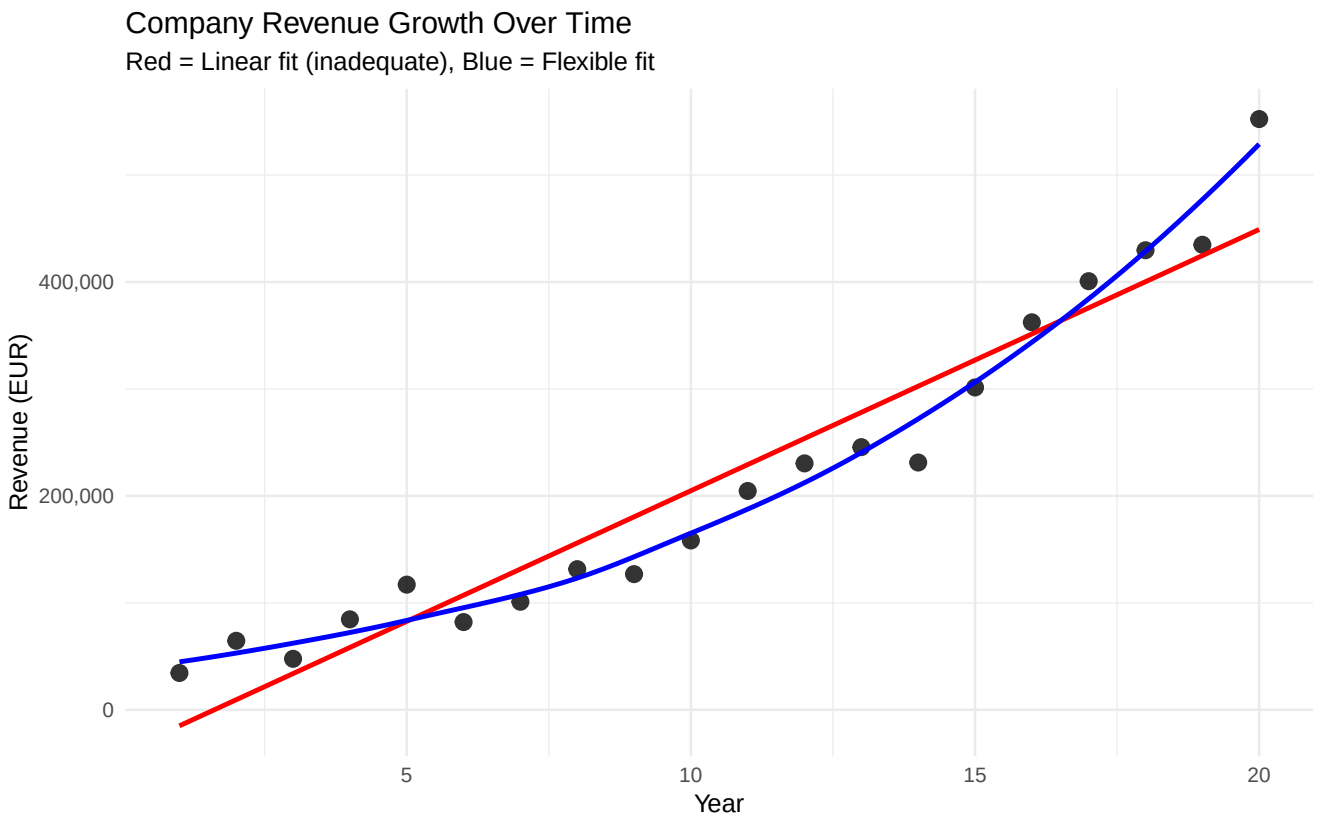
Why this matters: Real business data often shows exponential growth (revenue over time, user growth, compound returns)

Key insight: Log transformation linearizes exponential relationships

4.3.2 Setup the Problem

```
# Visualize the exponential relationship
ggplot(firm_growth_data, aes(x = year, y = revenue)) +
  geom_point(size = 3, alpha = 0.8) +
  geom_smooth(method = "lm", se = FALSE, color = "red", linewidth = 1) +
  geom_smooth(method = "loess", se = FALSE, color = "blue", linewidth = 1) +
  labs(title = "Company Revenue Growth Over Time",
       subtitle = "Red = Linear fit (inadequate), Blue = Flexible fit",
       x = "Year",
       y = "Revenue (EUR)") +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
`geom_smooth()` using formula = 'y ~ x'
```



Point out: "The red linear line completely fails to capture the exponential pattern!"

4.3.3 Apply Log Transformation

```
# Create log-transformed variable
firm_growth_data <- firm_growth_data %>%
  mutate(log_revenue = log(revenue))
```

```r
# Fit both models
model_linear <- lm(revenue ~ year, data = firm_growth_data)
model_log <- lm(log_revenue ~ year, data = firm_growth_data)

# Compare R²
r2_linear <- summary(model_linear)$r.squared
r2_log <- summary(model_log)$r.squared

cat("Linear model R²:", round(r2_linear, 4), "\n")
```

```
Linear model R²: 0.9188
```

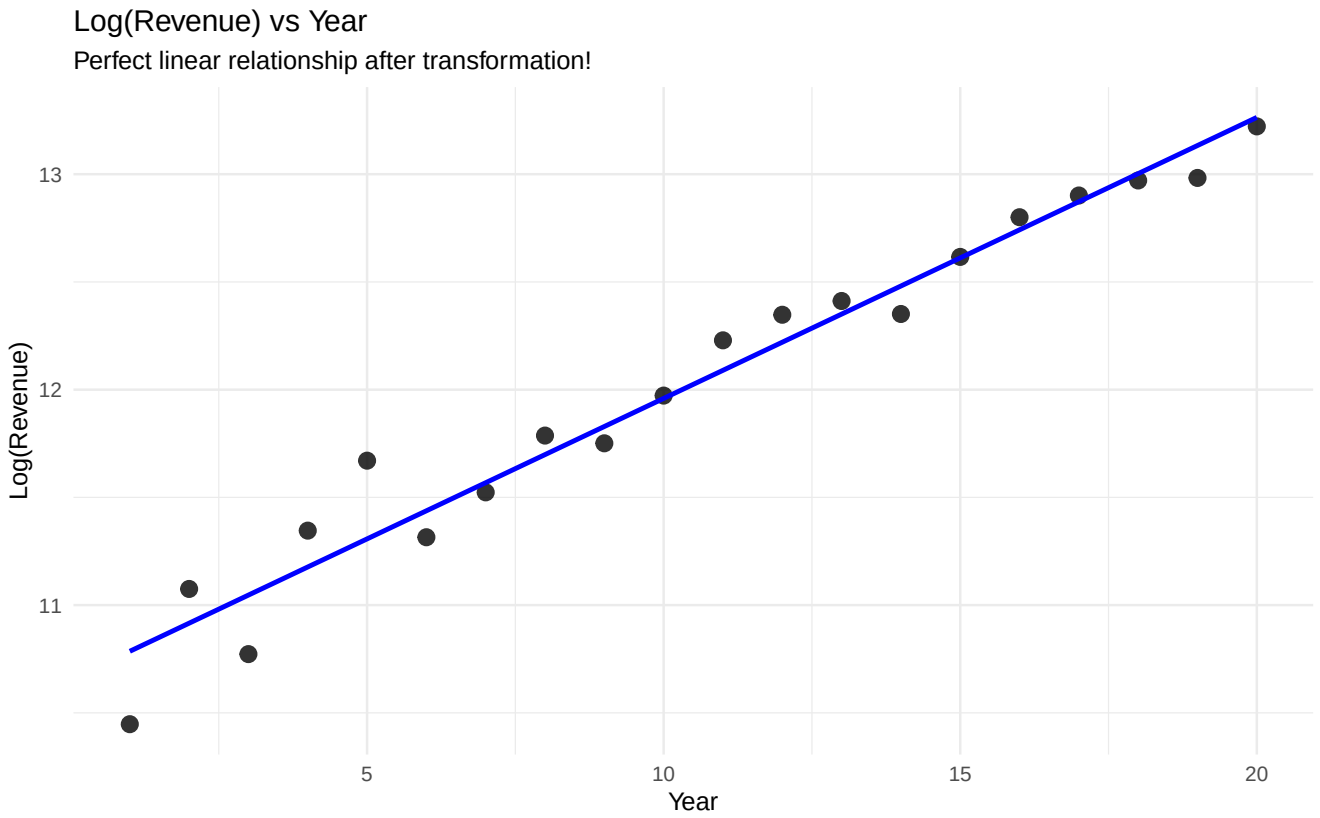
```r
cat("Log model R²:", round(r2_log, 4), "\n")
```

```
Log model R²: 0.9583
```

4.3.4 Visualize the Transformation

```r
# Plot log-transformed data
ggplot(firm_growth_data, aes(x = year, y = log_revenue)) +
  geom_point(size = 3, alpha = 0.8) +
  geom_smooth(method = "lm", se = FALSE, color = "blue", linewidth = 1) +
  labs(title = "Log(Revenue) vs Year",
       subtitle = "Perfect linear relationship after transformation!",
       x = "Year",
       y = "Log(Revenue)") +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```



What to emphasize:

> "After log transformation, the relationship becomes perfectly linear! The R^2 jumped from 0.92 to 0.96."

4.3.5 Interpretation of Log-Transformed Models

```
# Get coefficient for year
coef_log <- coef(model_log)["year"]
percentage_change <- (exp(coef_log) - 1) * 100

cat("Coefficient in log model:", round(coef_log, 4), "\n")
```

```
Coefficient in log model: 0.1304
```

```
cat("This means:", round(percentage_change, 2), "% growth per year\n")
```

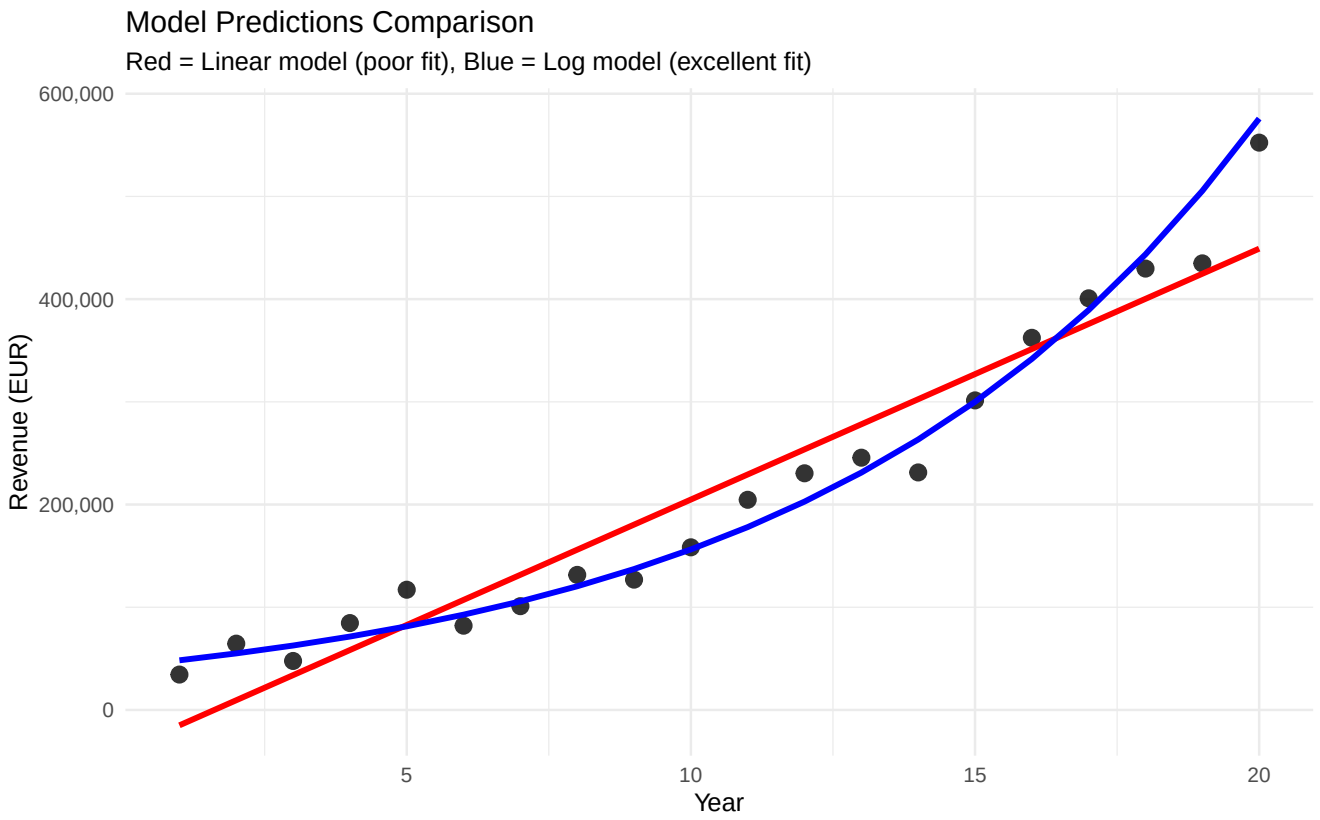
```
This means: 13.93 % growth per year
```

Key point for interpretation:

> "When we log-transform the dependent variable, coefficients represent **percentage changes**. Each year is associated with approximately 13.9% growth in revenue."

4.3.6 Visual Proof: Back to Original Scale

```
# Create predictions on original scale
predictions <- firm_growth_data %>%
  mutate(
    linear_pred = predict(model_linear),
    log_pred = exp(predict(model_log))  # Back-transform!
  )

ggplot(predictions, aes(x = year)) +
  geom_point(aes(y = revenue), size = 3, alpha = 0.8) +
  geom_line(aes(y = linear_pred), color = "red", linewidth = 1.2) +
  geom_line(aes(y = log_pred), color = "blue", linewidth = 1.2) +
  labs(title = "Model Predictions Comparison",
       subtitle = "Red = Linear model (poor fit), Blue = Log model (excellent fit)",
       x = "Year",
       y = "Revenue (EUR)") +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()
```



Final message:

> "The log model (blue) captures the exponential growth pattern perfectly, while the linear model (red) systematically misses the pattern. This is why data transformation is so important!"

---

5 Part 3: Experimental Analysis (35 minutes)

5.1 3.1 t-Tests (15 min)

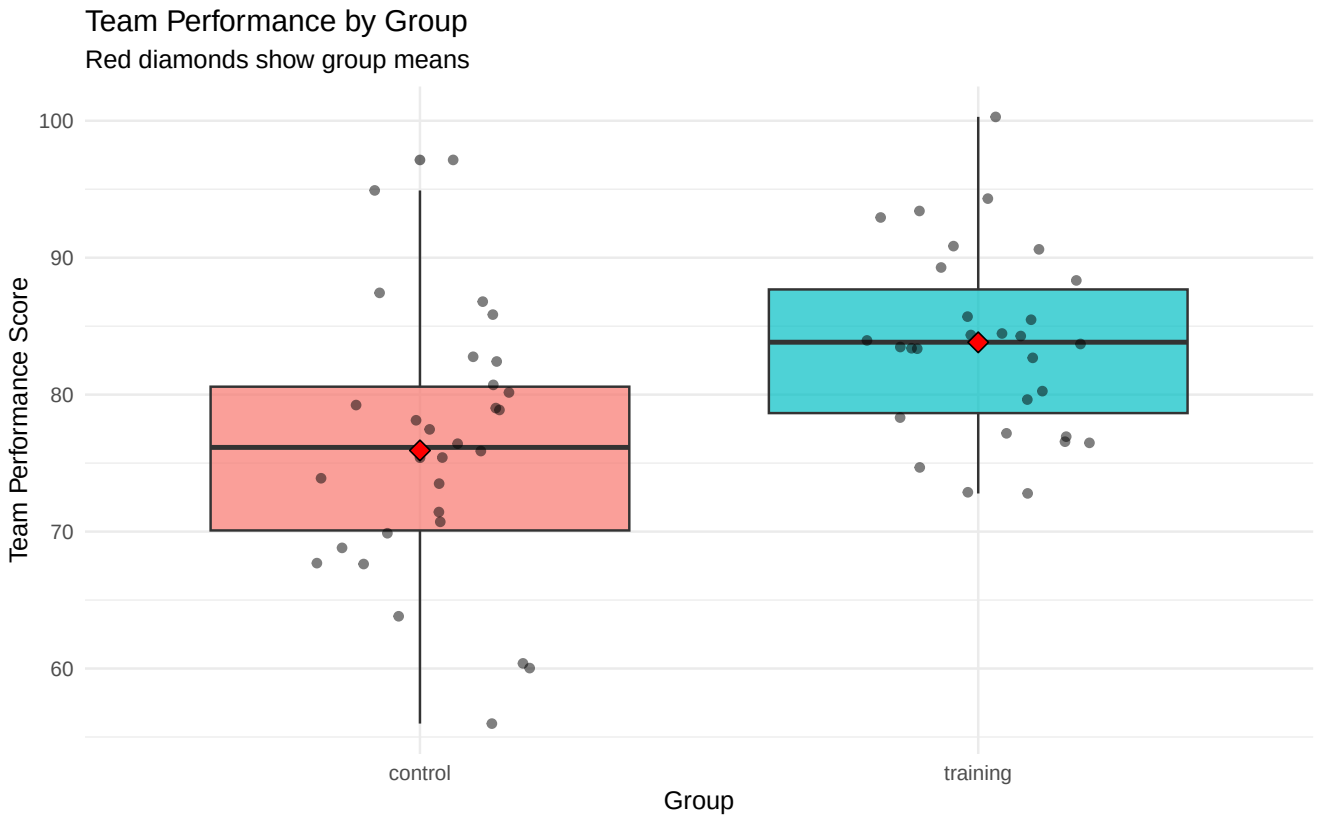
5.1.1 Teaching Notes

Research Question: Does leadership training improve team performance?

Key workflow: 1. Explore data visually 2. Check assumptions 3. Run test 4. Calculate effect size 5. Interpret for business decisions

5.1.2 Visual Exploration

```
ggplot(leadership_study_between, aes(x = group, y = team_performance, fill = group)) +
  geom_boxplot(alpha = 0.7) +
  geom_jitter(width = 0.2, alpha = 0.5) +
  stat_summary(fun = mean, geom = "point", shape = 23, size = 3, fill = "red") +
  labs(title = "Team Performance by Group",
       subtitle = "Red diamonds show group means",
       x = "Group",
       y = "Team Performance Score") +
  theme_minimal() +
  theme(legend.position = "none")
```



Ask class: "Based on this plot, do you think there's a difference?"

5.1.3 Check Assumptions

```
# Normality test for each group
shapiro_control <- shapiro.test(
  filter(leadership_study_between, group == "control")$team_performance
)
shapiro_training <- shapiro.test(
  filter(leadership_study_between, group == "training")$team_performance
)

cat("Shapiro-Wilk test for control group: p =",
    round(shapiro_control$p.value, 4), "\n")
```

```
Shapiro-Wilk test for control group: p = 0.9312
```

```
cat("Shapiro-Wilk test for training group: p =",
    round(shapiro_training$p.value, 4), "\n")
```

```
Shapiro-Wilk test for training group: p = 0.4427
```

```
# Equal variances test
levene_result <- leveneTest(team_performance ~ group,
                            data = leadership_study_between)
```

```
Warning in leveneTest.default(y = y, group = group, ...): group coerced to
factor.
```

```
cat("\nLevene's test for equal variances: p =",
    round(levene_result$`Pr(>F)`[1], 4), "\n")
```

```
Levene's test for equal variances: p = 0.1078
```

Interpretation:

> "Both p-values are > 0.05, so we cannot reject the hypotheses of normality and equal variances. We can proceed with the standard t-test."

5.1.4 Run t-test

```
t_result <- t.test(
  team_performance ~ group,
  data = leadership_study_between,
  var.equal = TRUE
)

print(t_result)
```

```
    Two Sample t-test

data:  team_performance by group
t = -3.7092, df = 58, p-value = 0.0004673
alternative hypothesis: true difference in means between group control and group training is not
95 percent confidence interval:
 -12.154597  -3.634084
sample estimates:
 mean in group control mean in group training
              75.92457                83.81891
```

Interpretation guide:

```r
mean_diff <- diff(t_result$estimate)
p_value <- t_result$p.value
ci_lower <- t_result$conf.int[1]
ci_upper <- t_result$conf.int[2]

cat("Mean difference:", round(mean_diff, 2), "\n")
```

```
Mean difference: 7.89
```

```r
cat("95% CI: [", round(ci_lower, 2), ",", round(ci_upper, 2), "]\n")
```

```
95% CI: [ -12.15 , -3.63 ]
```

```r
cat("p-value:", format.pval(p_value, digits = 3), "\n")
```

```
p-value: 0.000467
```

> "The training group scored 7.9 points higher on average. With $p < 0.001$, this difference is highly statistically significant."

5.1.5 Effect Size

```r
cohens_d_result <- cohens_d(team_performance ~ group,
                            data = leadership_study_between)
print(cohens_d_result)
```

```
Cohen's d |         95% CI
--------------------------
-0.96     | [-1.49, -0.42]

- Estimated using pooled SD.
```

Interpretation:

```
d_value <- abs(cohens_d_result$Cohens_d)

effect_label <- ifelse(d_value < 0.5, "small",
                       ifelse(d_value < 0.8, "medium", "large"))

cat("Cohen's d =", round(d_value, 2), "(", effect_label, "effect)\n")
```

```
Cohen's d = 0.96 ( large effect)
```

> "Cohen's d = 0.96 indicates a **large effect**. This means the difference is not only statistically significant but also **practically meaningful** for business decisions."

Business decision:

> "Based on these results, the leadership training shows a large, significant improvement in team performance. If the training costs less than the value of a 7.9-point performance improvement, it's worth implementing."

5.2 3.2 One-Way ANOVA (12 min)

5.2.1 Teaching Notes

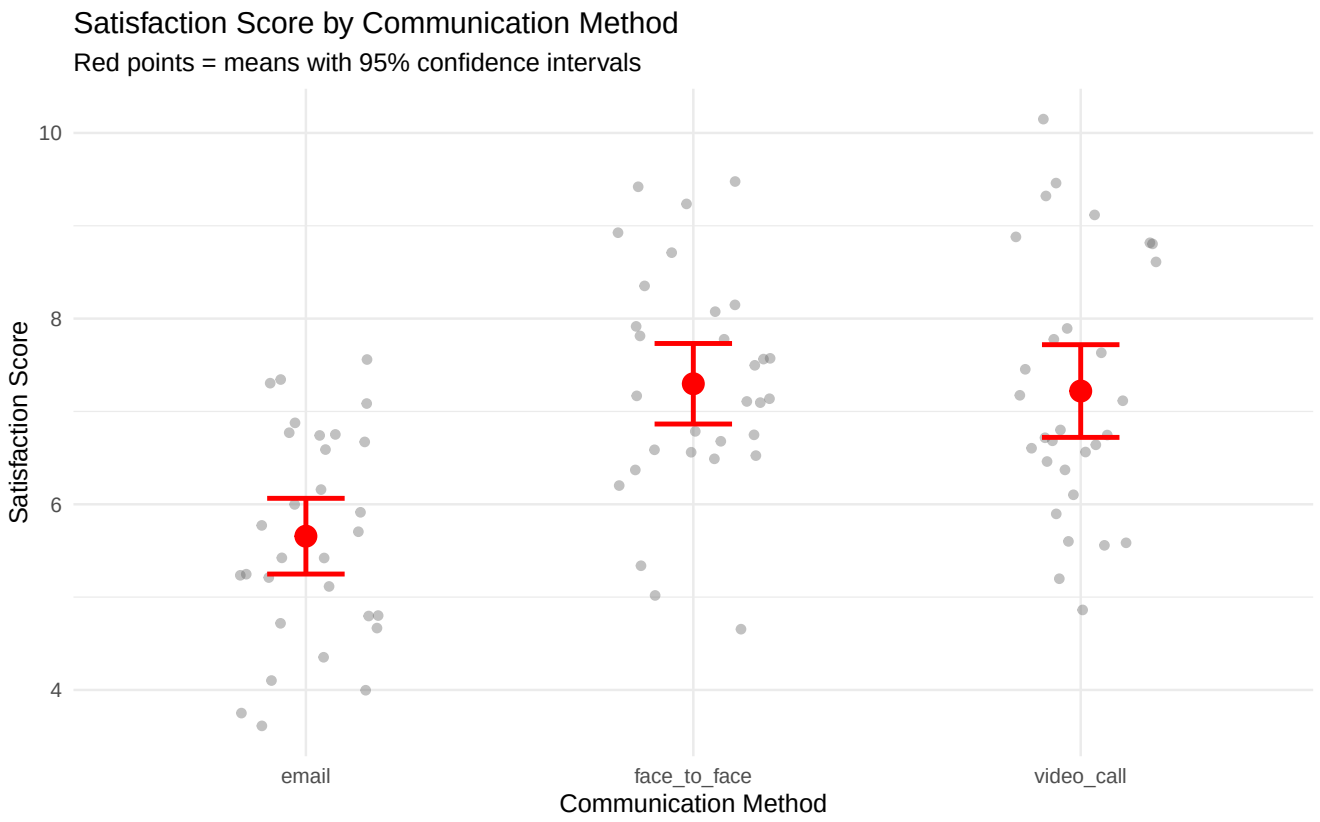
Research Question: Which communication method leads to highest satisfaction?

Key concept: ANOVA tests whether at least one group differs from others (not which specific groups differ - that requires post-hoc tests)

5.2.2 Visual Exploration

```
# Calculate means for plotting
comm_means <- communication_study %>%
  group_by(communication_method) %>%
  summarise(
    mean_satisfaction = mean(satisfaction_score),
    se = sd(satisfaction_score) / sqrt(n())
  )

ggplot(communication_study, aes(x = communication_method, y = satisfaction_score)) +
  geom_jitter(width = 0.2, alpha = 0.4, color = "gray40") +
  geom_point(data = comm_means, aes(y = mean_satisfaction),
             size = 4, color = "red") +
  geom_errorbar(data = comm_means,
                aes(y = mean_satisfaction,
                    ymin = mean_satisfaction - 1.96*se,
```

```
                    ymax = mean_satisfaction + 1.96*se),
                width = 0.2, color = "red", linewidth = 1) +
  labs(title = "Satisfaction Score by Communication Method",
       subtitle = "Red points = means with 95% confidence intervals",
       x = "Communication Method",
       y = "Satisfaction Score") +
  theme_minimal()
```



5.2.3 Fit ANOVA

```
# Fit model
anova_model <- aov(satisfaction_score ~ communication_method,
                   data = communication_study)
summary(anova_model)
```

```
                     Df Sum Sq Mean Sq F value   Pr(>F)
communication_method  2  51.46  25.731   16.37 9.21e-07 ***
Residuals            87 136.72   1.571
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation:

```r
anova_summary <- summary(anova_model)
f_value <- anova_summary[[1]]$`F value`[1]
p_value_anova <- anova_summary[[1]]$`Pr(>F)`[1]

cat("F-statistic =", round(f_value, 2), "\n")
```

```
F-statistic = 16.37
```

```r
cat("p-value:", format.pval(p_value_anova, digits = 3), "\n")
```

```
p-value: 9.21e-07
```

> "The very small p-value ($p < 0.001$) tells us that **at least one** communication method produces significantly different satisfaction scores than the others. But it doesn't tell us *which* methods differ."

5.2.4 Connection to Regression

Key teaching point:

```r
# Show that lm() gives identical results
lm_model <- lm(satisfaction_score ~ communication_method,
               data = communication_study)
anova(lm_model)
```

```
Analysis of Variance Table

Response: satisfaction_score
                     Df  Sum Sq Mean Sq F value    Pr(>F)
communication_method  2  51.462 25.7312  16.374 9.212e-07 ***
Residuals            87 136.716  1.5714
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

> "ANOVA is just a special case of regression! When we use categorical predictors, R creates dummy variables automatically."

```r
summary(lm_model)$coefficients
```

```
                                 Estimate Std. Error   t value     Pr(>|t|)
(Intercept)                      5.656295  0.2288701 24.713991 4.231093e-41
communication_methodface_to_face 1.641875  0.3236713  5.072663 2.195718e-06
communication_methodvideo_call   1.563436  0.3236713  4.830320 5.797655e-06
```

Explain:

"Since 'email' comes first alphabetically, it's the reference group. The coefficients show:

- Intercept = mean satisfaction for email group
- face_to_face coefficient = difference between face_to_face and email
- video_call coefficient = difference between video_call and email"

5.2.5 Effect Size

```
eta_sq <- eta_squared(anova_model)
```

```
For one-way between subjects designs, partial eta squared is equivalent
  to eta squared. Returning eta squared.
```

```
print(eta_sq)
```

```
# Effect Size for ANOVA

Parameter            | Eta2 |       95% CI
-----------------------------------------
communication_method | 0.27 | [0.14, 1.00]

- One-sided CIs: upper bound fixed at [1.00].
```

Interpretation:

```
eta_value <- eta_sq$Eta2[1]
cat(" ² =", round(eta_value, 3), "\n")
```

```
 ² = 0.273
```

```
cat("Communication method explains", round(eta_value * 100, 1),
    "% of variance in satisfaction\n")
```

```
Communication method explains 27.3 % of variance in satisfaction
```

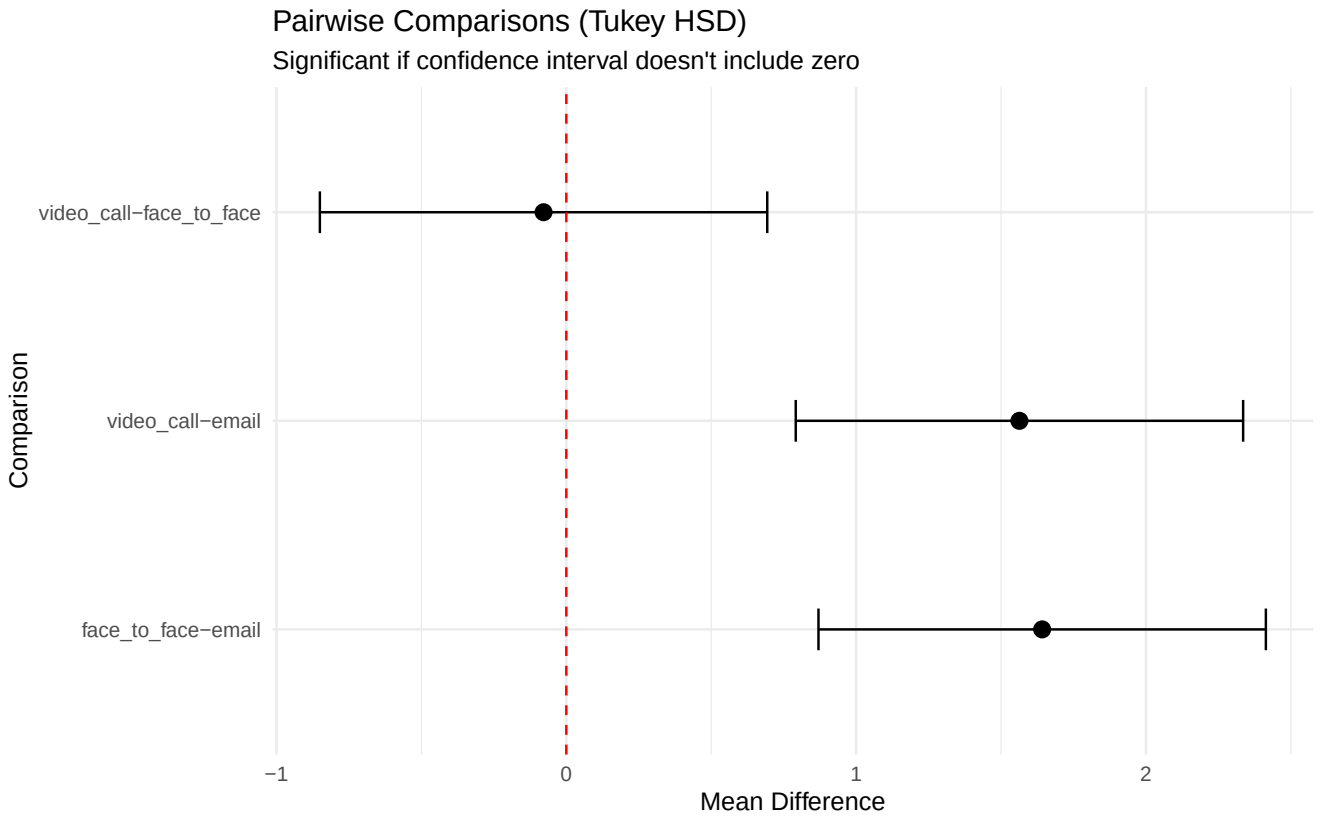
5.2.6 Post-Hoc Tests

```
tukey_result <- TukeyHSD(anova_model)
print(tukey_result)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = satisfaction_score ~ communication_method, data = communication_study)

$communication_method
                               diff        lwr       upr     p adj
face_to_face-email       1.64187523  0.8700877 2.4136627 0.0000065
video_call-email         1.56343582  0.7916483 2.3352233 0.0000172
video_call-face_to_face -0.07843941 -0.8502269 0.6933481 0.9681547
```

Create visualization:

```r
# Convert to data frame for plotting
tukey_df <- as.data.frame(tukey_result$communication_method)
tukey_df$comparison <- rownames(tukey_df)

ggplot(tukey_df, aes(x = comparison, y = diff)) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = lwr, ymax = upr), width = 0.2) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  coord_flip() +
  labs(title = "Pairwise Comparisons (Tukey HSD)",
       subtitle = "Significant if confidence interval doesn't include zero",
       x = "Comparison",
       y = "Mean Difference") +
  theme_minimal()
```



Interpretation:

"The post-hoc tests reveal:

- Face-to-face vs email: Significant difference (CI doesn't include 0)
- Video_call vs email: Significant difference
- Video_call vs face_to_face: NOT significant (CI includes 0)

Conclusion: Both face-to-face and video calls produce higher satisfaction than email, but face-to-face and video don't differ significantly from each other."

5.3 3.3 Brief Preview: Factorial Designs (8 min)

5.3.1 Teaching Notes

Note: Students haven't learned about interactions formally in the tutorials. Keep this light and conceptual.

Goal: Plant the seed that effects can depend on other variables

5.3.2 Conceptual Introduction

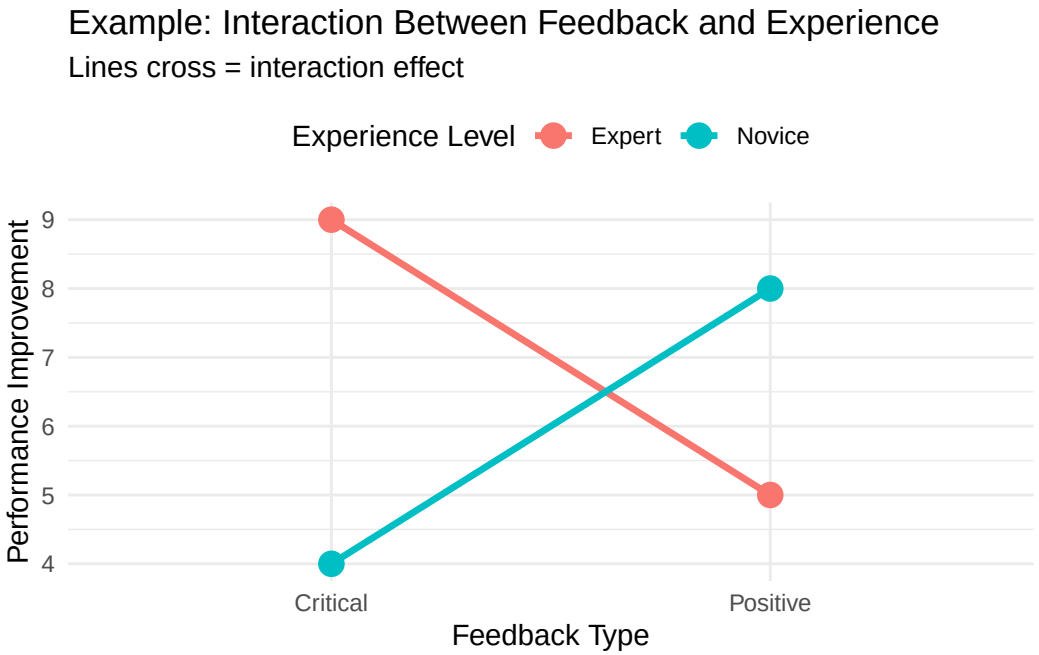
"So far we've looked at one factor at a time. But what if the effect of one variable **depends on** another variable? That's what factorial designs let us explore."

5.3.3 Simple Example (Conceptual)

On board/slide:

```
Research Question: Does feedback type (positive vs. critical)
affect performance improvement?

But wait... does this depend on experience level?

Maybe:
- Novices benefit from positive feedback (encouragement)
- Experts benefit from critical feedback (growth mindset)

This is called an INTERACTION EFFECT.
```

5.3.4 Show the Pattern Visually



Point out:

"Notice how the lines cross? This tells us the effect of feedback type is **different** for novices vs. experts. That's an interaction!

- For novices: Positive feedback works better
- For experts: Critical feedback works better

If there was no interaction, the lines would be parallel."

5.3.5 Why This Matters

"Interactions are everywhere in business:

- Does advertising effectiveness depend on customer segment?
- Does training effectiveness depend on prior knowledge?
- Does pricing strategy depend on market conditions?

The tutorials cover how to test for interactions with two-way ANOVA. For now, just remember: **effects can depend on context**."

Transition to exercise:

"In your exercise, you'll work with simpler designs - just regression and t-tests. But keep interactions in mind for future analyses!"

---

6 Part 4: Guided Exercise (20 minutes)

6.1 Exercise Setup (2 min)

6.1.1 Scenario

You have data from an A/B test of two website designs:

- **Simple Design:** Minimalist layout, fewer options
- **Complex Design:** Feature-rich layout, many options

Variables measured:

- `design`: Which design the user saw (Simple vs. Complex)
- `time_on_site`: Time spent on site (seconds)
- `previous_visits`: Number of previous visits to the site
- `converted`: Whether user made a purchase (1 = yes, 0 = no)

Dataset: `exercise_data.csv`

6.1.2 Tasks for Students

Task 1: Regression Analysis

a. Create a scatter plot of `time_on_site` vs `converted`
b. Fit a linear regression: `converted ~ time_on_site`
c. Interpret the coefficient for `time_on_site`
d. Calculate and report R^2

Task 2: Compare Designs

a. Use a t-test to compare `time_on_site` between the two designs
b. Calculate Cohen's d effect size

c. Which design keeps users on the site longer?

Task 3: Omitted Variable Bias (Challenge)

a. Add `previous_visits` to your regression model
b. How does the coefficient for `time_on_site` change?
c. Does this suggest omitted variable bias? Why or why not?

6.2 Solution Guide (For Discussion)

6.2.1 Task 1 Solution

```
# Load data
exercise_data <- read.csv("exercise_data.csv")

# a. Scatter plot
ggplot(exercise_data, aes(x = time_on_site, y = converted)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(title = "Conversion vs Time on Site",
       x = "Time on Site (seconds)",
       y = "Converted (0/1)") +
  theme_minimal()

# b. Simple regression
model_exercise_simple <- lm(converted ~ time_on_site, data = exercise_data)
summary(model_exercise_simple)

# c. Interpretation
coef_time <- coef(model_exercise_simple)["time_on_site"]
cat("For every additional second on site, conversion probability increases by",
    round(coef_time, 4), "\n")

# d. R²
r2 <- summary(model_exercise_simple)$r.squared
cat("R² =", round(r2, 4), "meaning the model explains",
    round(r2*100, 1), "% of variance\n")
```

6.2.2 Task 2 Solution

```
# a. t-test
t_test_design <- t.test(time_on_site ~ design, data = exercise_data)
print(t_test_design)
```

```r
# b. Effect size
cohens_d_design <- cohens_d(time_on_site ~ design, data = exercise_data)
print(cohens_d_design)

# c. Interpretation
cat("Complex design keeps users", abs(round(diff(t_test_design$estimate), 1)),
    "seconds longer on average\n")
```

6.2.3 Task 3 Solution

```r
# a. Add previous_visits
model_exercise_multiple <- lm(converted ~ time_on_site + previous_visits,
                              data = exercise_data)
summary(model_exercise_multiple)

# b. Compare coefficients
coef_simple <- coef(model_exercise_simple)["time_on_site"]
coef_multiple <- coef(model_exercise_multiple)["time_on_site"]

cat("Simple model coefficient:", round(coef_simple, 5), "\n")
cat("Multiple model coefficient:", round(coef_multiple, 5), "\n")
cat("Change:", round(coef_multiple - coef_simple, 5), "\n")

# c. Interpretation
# If coefficient changed substantially, there was omitted variable bias
# This happens when previous_visits correlates with both time_on_site
# and converted
```

6.3 Discussion Points (3 min)

Call on students to share:

1. "What did you find in Task 1? Is time on site a good predictor?"
2. "In Task 2, which design won? Was the effect large?"
3. "In Task 3, did the coefficient change much? What does that tell us?"

Key takeaways to emphasize:

- Time on site predicts conversion, but R^2 might be modest
- One design likely keeps users longer (check which one!)
- If previous_visits affected the coefficient, it was a confounder
- This is why we need multiple regression - to control for confounders!

---

7 Part 5: Wrap-up (5 minutes)

7.1 Key Concepts Review

7.1.1 Linear Regression

Multiple regression controls for confounders and gives *ceteris paribus* effects
Omitted variable bias happens when we leave out relevant variables
Log transformation linearizes exponential relationships
R^2 measures explained variance, but doesn't tell the whole story
Diagnostic plots check model assumptions

7.1.2 Experimental Analysis

Always check assumptions before running tests
t-tests compare two groups
ANOVA compares three or more groups
Effect sizes (Cohen's d, ^{2}) tell us about practical significance
Post-hoc tests identify which specific groups differ
ANOVA is regression with categorical predictors

7.2 Resources

- **Tutorials:** Full details on all methods we covered today
- **Power analysis:** See experiments tutorial for sample size planning
- **Office hours:** For questions on your own data/projects

7.3 Final Message

"Statistics is not just about getting significant p-values. It's about:

1. Understanding what your data can and cannot tell you
2. Making valid comparisons by controlling for confounders
3. Assessing practical significance, not just statistical significance
4. Checking assumptions so your results are trustworthy

Practice these workflows, and you'll be able to analyze real data effectively!"

---

8 Appendix: Common Student Questions

8.1 "When should I use log transformation?"

- When you see exponential growth/decay patterns
- When plotting shows a curve that gets steeper over time
- When residuals show a pattern (heteroscedasticity)
- Common in: revenue over time, population growth, compound effects

8.2 "What if my assumptions are violated?"

- **Non-normal data:** Use non-parametric tests (Wilcoxon, Kruskal-Wallis)
- **Unequal variances:** Use Welch's t-test or robust ANOVA
- **Small samples:** Bootstrap methods
- **Always visualize first** - sometimes transformations help!

8.3 "How do I know if an effect size is 'big enough'?"

- Cohen's guidelines are just rules of thumb
- Consider the **context**:
  - A 2% improvement in click-through rate might be huge for online advertising
  - A 10-point IQ difference might be small for educational interventions
- Ask: "Is this large enough to matter for decisions?"

8.4 "Simple vs. multiple regression - which should I use?"

- **Simple:** When you have one clear predictor and no obvious confounders
- **Multiple:** Almost always in real research!
  - Controls for confounders
  - More accurate effect estimates
  - Allows you to compare importance of different predictors

8.5 "What's the difference between aov() and lm()?"

- They're the same! ANOVA is regression with categorical predictors
- Use `aov()` for traditional ANOVA output
- Use `lm()` when you want to see coefficients or add continuous predictors
- Both give identical F-statistics and p-values