

R Cheat Sheet: Regression & Experimental Analysis

Quick Reference for Lab Session

Linear Regression

Fitting Models

```
# Simple regression
model <- lm(dependent_var ~ independent_var, data = mydata)

# Multiple regression
model <- lm(dependent_var ~ var1 + var2, data = mydata)

# With log transformation
model <- lm(log(dependent_var) ~ independent_var, data = mydata)
```

Examining Results

```
summary(model)           # Full model summary
coef(model)              # Extract coefficients
confint(model)           # Confidence intervals
summary(model)$r.squared  # R2
```

Model Diagnostics

```
# Extract values
fitted_values <- fitted(model)      # Fitted values (predictions)
residuals_values <- residuals(model) # Residuals

# Diagnostic plots
plot(fitted_values, residuals_values) # Residuals vs Fitted
qqnorm(residuals_values)             # Q-Q plot
qqline(residuals_values)             # Add reference line
```

t-Tests

Independent Samples t-test

```
# Basic syntax
t.test(outcome ~ group, data = mydata)

# With equal variances assumed
t.test(outcome ~ group, data = mydata, var.equal = TRUE)

# With unequal variances (Welch's test)
t.test(outcome ~ group, data = mydata, var.equal = FALSE)
```

Paired t-test

```
t.test(after ~ before, data = mydata, paired = TRUE)
```

Effect Size (Cohen's d)

```
library(effectsize)
cohens_d(outcome ~ group, data = mydata)
```

ANOVA

One-Way ANOVA

```
# Using aov()
model <- aov(outcome ~ group, data = mydata)
summary(model)

# Using lm() (gives same results)
model <- lm(outcome ~ group, data = mydata)
anova(model)
```

Effect Size (η^2)

```
library(effectsize)
eta_squared(model)
```

Post-Hoc Tests

```
# Tukey HSD (for aov models)
TukeyHSD(model)

# Alternative using emmeans (works with lm too)
library(emmeans)
emmeans(model, pairwise ~ group)
```

Assumption Checking

Normality Tests

```
# Shapiro-Wilk test (H0: data is normally distributed)
shapiro.test(mydata$variable)

# For residuals
shapiro.test(residuals(model))

# Visual check
qqnorm(mydata$variable)
qqline(mydata$variable)
```

Equal Variances Test

```
library(car)
# Levene's test (H0: variances are equal)
leveneTest(outcome ~ group, data = mydata)
```

Data Transformation

Log Transformation

```
# Create new variable
mydata <- mydata %>%
  mutate(log_var = log(variable))

# Interpretation: coefficients = % change
# Formula: percentage_change = (exp(coefficient) - 1) * 100
```

Square/Quadratic

```
# For U-shaped or inverted-U relationships
mydata <- mydata %>%
  mutate(var_squared = variable^2)

model <- lm(outcome ~ variable + var_squared, data = mydata)
```

Visualization (ggplot2)

Scatter Plot with Regression Line

```
ggplot(data, aes(x = predictor, y = outcome)) +
  geom_point() +
  geom_smooth(method = "lm", se = TRUE)
```

Boxplot for Group Comparisons

```
ggplot(data, aes(x = group, y = outcome, fill = group)) +
  geom_boxplot() +
  geom_jitter(width = 0.2, alpha = 0.5) +
  stat_summary(fun = mean, geom = "point", shape = 23, size = 3, fill = "red")
```

Bar Plot with Error Bars

```
# First calculate means and SE
summary_data <- data %>%
  group_by(group) %>%
  summarise(
    mean = mean(outcome),
    se = sd(outcome) / sqrt(n())
  )

# Then plot
ggplot(summary_data, aes(x = group, y = mean)) +
  geom_col(fill = "steelblue") +
  geom_errorbar(aes(ymin = mean - 1.96*se, ymax = mean + 1.96*se),
    width = 0.2)
```

Common dplyr Operations

```
library(dplyr)

# Filter rows
data %>% filter(variable > 10)

# Select columns
data %>% select(var1, var2)

# Create new variables
data %>% mutate(new_var = var1 + var2)

# Group operations
data %>%
  group_by(group) %>%
  summarise(
    mean = mean(outcome),
    sd = sd(outcome),
    n = n()
  )
```

Interpretation Guidelines

p-values

- **p < 0.05:** Statistically significant (reject null hypothesis)
- **p = 0.05:** Not statistically significant (fail to reject null)
- Report exact p-values when possible (not just “p < 0.05”)

Effect Sizes

Cohen's d (for t-tests)

- **Small:** d = 0.2
- **Medium:** d = 0.5
- **Large:** d = 0.8

η^2 (eta-squared, for ANOVA)

- **Small:** η^2 = 0.01 (1% of variance)
- **Medium:** η^2 = 0.06 (6% of variance)
- **Large:** η^2 = 0.14 (14% of variance)

R² (for regression)

- Proportion of variance explained
- Range: 0 to 1 (or 0% to 100%)
- Higher is better, but context matters!

Quick Troubleshooting

Common Errors

Error	Likely Cause	Solution
object not found	Variable name typo or data not loaded	Check spelling, use <code>head(data)</code>
non-numeric argument	Using categorical where numeric expected	Check variable type with <code>str(data)</code>
missing values	NA values in data	Use <code>na.omit()</code> or check with <code>sum(is.na(data))</code>
Package not found	Library not installed	Run <code>install.packages("packagename")</code>

Getting Help in R

```
?function_name      # Help for specific function
??search_term        # Search help files
example(lm)          # See examples
str(data)            # Structure of data
head(data)           # First 6 rows
summary(data)        # Summary statistics
```

Essential Packages to Load

```
library(ggplot2)    # Visualization
library(dplyr)      # Data manipulation
library(broom)       # Model tidying (optional, but useful)
library(effectsize) # Effect size calculations
library(car)         # Assumption tests (Levene's test)
```

Statistical Decision Tree

Do you have...

1 continuous outcome + 1 continuous predictor?

→ Linear Regression (lm)

1 continuous outcome + 2+ continuous predictors?

→ Multiple Regression (lm)

1 continuous outcome + 1 categorical predictor (2 groups)?

→ Independent t-test (t.test)

1 continuous outcome + 1 categorical predictor (3+ groups)?

→ One-Way ANOVA (aov or lm)

Non-linear relationship?

→ Consider transformation (log, quadratic)

Binary outcome (0/1)?

→ Logistic Regression (glm, family = binomial)

Tip: Keep this sheet handy during the exercise portion. Most questions can be answered by adapting these templates to your specific variable names!