Data Analysis in Natural Sciences: An R-Based Approach

From Field to Figures: Essential Methods for Environmental and Life Sciences

Dr. Jimmy Moses (PhD)

2025-03-27

Table of contents

Preface

Welcome to **Data Analysis in Natural Sciences: An R-Based Approach**, a comprehensive guide designed for students, professionals, and researchers across the natural sciences. This book provides practical methods for analyzing and visualizing data using R, with applications spanning forestry, agriculture, ecology, marine biology, environmental science, geology, atmospheric science, hydrology, and more.

About the Author

This book has been developed by **Dr. Jimmy Moses (PhD)** from the School of Forestry, Faculty of Natural Resources, Papua New Guinea University of Technology. With extensive experience in ecological research and data analysis, Dr. Moses has created this resource to support students and researchers in developing essential analytical skills for natural science disciplines.

Target Audience

This book is designed for:

- **Undergraduate and postgraduate students** in natural science disciplines
- **Researchers** seeking to enhance their data analysis capabilities
- **Technicians** working in laboratories and field settings
- **Professionals** in government agencies, NGOs, and private sector
- **Hobbyists** with an interest in analyzing scientific data

The content is relevant to those working in:

- Forestry and agroforestry
- Agriculture and agronomy
- Ecology and conservation
- Environmental science
- Geography and GIS/remote sensing
- Marine biology and fisheries
- Botany and plant sciences
- Entomology and zoology
- Epidemiology and veterinary sciences
- Geology and earth sciences
- Atmospheric and climate sciences
- Hydrology and water resources
- Natural resource management
- Conservation biology

What You Will Learn

This book will guide you through:

- The fundamentals of data analysis with R
- Data preparation and management techniques
- Exploratory data analysis approaches
- Statistical hypothesis testing
- Advanced visualization methods
- Specialized analyses for environmental and scientific data
- Reproducible research practices

How to Use This Book

This book is designed to be both a learning resource and a reference guide. You can read it from start to finish to build your skills progressively, or use specific chapters as needed for particular tasks.

Code examples are provided throughout, and you can run them directly in R or RStudio. Each chapter includes practical examples using real datasets from various natural science disciplines.

Prerequisites

To get the most out of this book, you should have:

- Basic computer skills
- R and RStudio installed (instructions provided in Chapter 1)
- A basic understanding of statistics (helpful but not required)

Acknowledgments

I would like to thank all those who contributed to the development of this book, including colleagues, students, and the open-source community that makes tools like R and RStudio possible.

Let’s begin our journey into the world of data analysis for natural sciences!

About This Book

About the Author

Dr. Jimmy Moses is a Papua New Guinean entomologist and lecturer at the Papua New Guinea University of Technology's School of Forestry, specializing in ant ecology, biostatistics, and geospatial analysis. He holds a Ph.D. in Entomology from the University of South Bohemia (2021) and has extensive experience in tropical ecology research, particularly focusing on ant communities along elevational gradients.

He currently supervises four master's students and co-supervises a Ph.D. student, Dr. Moses brings significant expertise in both research and education. He has published several peer-reviewed papers, including work in prestigious journals like *Global Ecology and Biogeography* and *Proceedings of the Royal Society B*.

His technical skills span multiple areas:

- Advanced proficiency in R and Python for statistical computing and data science
- Expertise in GIS and Satellite Remote Sensing
- Strong background in biostatistics and experimental design
- Emerging skills in full-stack app development

Dr. Moses has maintained strong international collaborations, having worked with institutions in the Czech Republic, Germany, and Belgium. He has been actively involved with the New Guinea Binatang Research Center, contributing to both research and education initiatives in Papua New Guinea.

His research interests combine ecological field studies with modern analytical approaches, particularly in ant ecology, spatial ecology, macroecology, and crop protection. He spends his free time reading technical, historical, psychological and ecological books and more time tinkering with codes.

Purpose and Scope

This book is designed to serve as both a learning resource and a reference guide for data analysis in the natural sciences, with applications spanning forestry, agriculture, ecology, environmental science, marine biology, and related disciplines. Whether you're a student, researcher, technician, professional, or hobbyist in these fields, this book will help you develop the skills needed to analyze and visualize data effectively using R.

The focus is on practical applications rather than theoretical statistics, with an emphasis on techniques commonly used across natural science disciplines. By working through this book, you will:

- Master the fundamentals of data analysis in R
- Learn to import, clean, and organize various types of scientific data
- Develop skills in exploratory data analysis and visualization
- Apply appropriate statistical tests for different research questions
- Create publication-quality visualizations
- Implement reproducible research workflows
- Interpret and communicate results effectively

Features of This Book

This book includes:

- Step-by-step instructions for R with complete code examples
- Practical examples using real datasets from various natural science disciplines
- Exercises to reinforce learning and build skills
- Tips and best practices from experienced researchers
- Reproducible code that can be adapted for your own research

How to Use the Code Examples

All code examples in this book are written in R and can be executed in RStudio. To use the examples:

1. Make sure you have R and RStudio installed (see Chapter 1 for installation instructions)
2. Install the required packages mentioned at the beginning of each chapter
3. Copy and paste the code into your R console or script editor
4. Modify the code as needed for your own data

The datasets used in the examples are available in the `docs/data` directory of the book's repository and are properly cited throughout the text.

Software Requirements

This book uses:

- R (version 4.0.0 or higher)
- RStudio (latest version recommended)
- Various R packages (installation instructions provided in each chapter)

Feedback and Contributions

Your feedback is valuable for improving future editions of this book. If you find errors, have suggestions, or want to contribute examples, please submit them through the book's repository or contact the author directly.

Acknowledgments

I would like to express my gratitude to colleagues, students, and the broader R community whose insights and feedback have contributed to the development of this book. Special thanks to the creators and maintainers of the R packages used throughout this book, as well as the data providers whose datasets make the examples both practical and relevant.

Part I

Getting Started

Chapter 1

Introduction to Data Analysis

1.1 Overview

Data analysis is a critical skill in modern natural sciences research (Wickham & Grolemund, 2016; Zuur et al., 2009). This chapter introduces the fundamental concepts, tools, and approaches that form the foundation of effective data analysis across various scientific disciplines.

1.2 Why Data Analysis Matters in Natural Sciences

Data analysis plays a pivotal role in natural sciences research for several reasons:

1. **Evidence-Based Decision Making**: Data analysis transforms raw observations into actionable insights, enabling researchers and practitioners to make informed decisions about conservation strategies, resource management practices, agricultural planning, environmental interventions, and more (Bolker et al., 2009).
2. **Pattern Recognition**: Through statistical analysis, researchers can identify patterns, trends, and relationships within natural systems that might not be apparent from casual observation alone (Zuur et al., 2007). This applies to diverse fields including ecology, geology, marine biology, atmospheric science, and agriculture.
3. **Hypothesis Testing**: Data analysis provides rigorous methods to test hypotheses about natural phenomena, allowing researchers to build and refine scientific theories about how natural systems function (Gotelli & Ellison, 2004). This is fundamental across all scientific disciplines.
4. **Prediction and Modeling**: Advanced analytical techniques enable the development of predictive models that can forecast changes in natural systems, such as species distribution shifts under climate change, crop yield predictions, geological processes, weather patterns, and more (Elith et al., 2009).

1.3 Tools for Data Analysis

This book focuses on R and RStudio as the primary tools for data analysis:

1.3.1 R and RStudio

R is a powerful programming language and environment specifically designed for statistical computing and graphics. RStudio is an integrated development environment (IDE) that makes working with R more accessible and efficient.

Key advantages of R include:

- **Open-source and free**: Available to anyone without cost
- **Extensive package ecosystem**: Thousands of specialized packages for various types of analyses across all scientific disciplines
- **Reproducibility**: Code-based approach ensures analyses can be repeated and verified
- **Flexibility**: Can be adapted to virtually any analytical need in the natural sciences
- **Active community**: Large user base provides support and continuous development

```
# A simple example of R code using real-world data
# Load the Palmer penguins dataset (a subset of climate_data.csv)
penguins <- read.csv("../data/environmental/climate_data.csv")

# View the first few rows
head(penguins)
```

```
  species    island bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
1  Adelie Torgersen           39.1          18.7               181        3750
2  Adelie Torgersen           39.5          17.4               186        3800
3  Adelie Torgersen           40.3          18.0               195        3250
4  Adelie Torgersen             NA            NA                NA          NA
5  Adelie Torgersen           36.7          19.3               193        3450
6  Adelie Torgersen           39.3          20.6               190        3650
     sex year
1   male 2007
2 female 2007
3 female 2007
4   <NA> 2007
5 female 2007
6   male 2007
```

```
# Get a summary of bill length measurements
summary(penguins$bill_length_mm)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  32.10   39.23   44.45   43.92   48.50   59.60       2 
```

1.4 Setting Up Your Environment

1.4.1 Installing R and RStudio

To install R and RStudio:

1. Download and install R from CRAN
2. Download and install RStudio from RStudio's website

1.4.2 Essential R Packages

For the analyses in this book, you'll need several R packages. You can install them with the following code:

```
install.packages(c(
  "tidyverse",  # Data manipulation and visualization
  "rstatix",    # Statistical tests
  "ggplot2",    # Advanced plotting
  "knitr",      # Document generation
  "rmarkdown"   # Document formatting
))
```

1.5 The Data Analysis Workflow

Effective data analysis typically follows a structured workflow:

1. **Define the Question**: Clearly articulate what you want to learn from your data
2. **Collect Data**: Gather the necessary data through fieldwork, experiments, laboratory measurements, or existing datasets
3. **Clean and Prepare Data**: Handle missing values, correct errors, and format data appropriately
4. **Explore Data**: Conduct exploratory data analysis to understand patterns and distributions
5. **Analyze Data**: Apply appropriate statistical methods to address your research questions
6. **Interpret Results**: Draw conclusions based on your analysis
7. **Communicate Findings**: Present your results through visualizations, reports, or publications

Throughout this book, we'll follow this workflow as we explore various datasets from across the natural sciences.

1.6 Types of Data in Natural Sciences Research

Research across the natural sciences involves several types of data:

1.6.1 Categorical Data

Categorical data represent qualitative characteristics, such as: - Species names or taxonomic classifications - Habitat or ecosystem types - Rock or soil classifications - Land-use categories - Treatment groups in experiments - Genetic markers

1.6.2 Numerical Data

Numerical data involve measurements or counts: - Continuous measurements (e.g., temperature, pH, concentration, biomass, wavelength) - Discrete counts (e.g., number of individuals, species richness, occurrence frequency) - Rates (e.g., growth rates, reaction rates, decomposition rates) - Ratios and indices (e.g., diversity indices, chemical ratios)

1.6.3 Spatial Data

Spatial data describe geographical distributions: - Coordinates (latitude/longitude) - Elevation or depth - Topographic features - Land cover maps - Remote sensing data - Geological formations

1.6.4 Temporal Data

Temporal data track changes over time: - Time series of measurements - Seasonal patterns - Long-term monitoring data - Growth curves - Decay rates - Historical records

Understanding the type of data you're working with is crucial for selecting appropriate analytical methods across all natural science disciplines.

1.7 Summary

In this chapter, we've introduced the importance of data analysis in natural sciences research and the tools we'll be using throughout this book. We've also outlined the typical data analysis workflow and the types of data commonly encountered across scientific disciplines.

In the next chapter, we'll dive deeper into data basics, learning how to import, clean, and prepare data for analysis.

1.8 Exercises

1. Install R and RStudio on your computer.
2. Install the required R packages listed in this chapter.
3. Open RStudio and create a new R script. Try running a simple command like `summary(iris)`.
4. Think about a research question in your field of natural science that interests you. What type of data would you need to address this question?
5. Explore one of R's built-in datasets (e.g., `mtcars`, `iris`, or `trees`) using functions like `head()`, `summary()`, and `plot()`.

Chapter 2

Data Basics

2.1 Introduction

This chapter covers the fundamental concepts of working with data in R. You'll learn how to import, clean, and prepare data for analysis, which are essential skills for any data analysis project across all natural science disciplines.

2.2 Understanding Data Structures

Before diving into data analysis, it's important to understand the basic data structures in R:

2.2.1 Data Types

R has several basic data types:

- **Numeric**: Decimal values (e.g., measurements of temperature, pH, concentration, or distance)
- **Integer**: Whole numbers (e.g., counts of organisms, samples, or observations)
- **Character**: Text strings (e.g., species names, site descriptions, or treatment labels)
- **Logical**: TRUE/FALSE values (e.g., presence/absence data or condition met/not met)
- **Factor**: Categorical variables with levels (e.g., experimental treatments, taxonomic classifications, or soil types)
- **Date/Time**: Temporal data (e.g., sampling dates, observation times, or seasonal markers)

```r
# Examples of different data types
numeric_example <- 25.4  # Temperature in Celsius
character_example <- "Adelie"  # Penguin species
logical_example <- TRUE  # Presence/absence data
factor_example <- factor(c("Control", "Treatment", "Control"),
                         levels = c("Control", "Treatment"))
date_example <- as.Date("2020-07-15")  # Sampling date

# Print examples
print(numeric_example)
```

```
[1] 25.4
```

```r
print(character_example)
```

```
[1] "Adelie"
```

```
print(logical_example)
```

```
[1] TRUE
```

```
print(factor_example)
```

```
[1] Control   Treatment Control
Levels: Control Treatment
```

```
print(date_example)
```

```
[1] "2020-07-15"
```

2.2.2 Data Structures in R

R has several data structures for organizing information:

```
# Load real datasets
library(readr)
penguins <- read_csv("../data/environmental/climate_data.csv")
crops <- read_csv("../data/agriculture/crop_yields.csv")

# Vector example - penguin bill lengths
bill_lengths <- na.omit(penguins$bill_length_mm[1:10])
print(bill_lengths)
```

```
[1] 39.1 39.5 40.3 36.7 39.3 38.9 39.2 34.1 42.0
attr(,"na.action")
[1] 4
attr(,"class")
[1] "omit"
```

```
# Matrix example - create a matrix from penguin measurements
penguin_matrix <- as.matrix(penguins[1:5, 3:6])
print(penguin_matrix)
```

```
     bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
[1,]           39.1          18.7               181        3750
[2,]           39.5          17.4               186        3800
[3,]           40.3          18.0               195        3250
[4,]             NA            NA                NA          NA
[5,]           36.7          19.3               193        3450
```

```
# Data frame example - first few rows of penguin data
penguin_data <- penguins[1:5, ]
print(penguin_data)
```

```
# A tibble: 5 x 8
  species island    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
  <chr>   <chr>              <dbl>         <dbl>             <dbl>       <dbl>
1 Adelie  Torgersen           39.1          18.7               181        3750
2 Adelie  Torgersen           39.5          17.4               186        3800
3 Adelie  Torgersen           40.3          18                 195        3250
4 Adelie  Torgersen           NA            NA                  NA          NA
5 Adelie  Torgersen           36.7          19.3               193        3450
# i 2 more variables: sex <chr>, year <dbl>
# List example - store different aspects of the dataset
penguin_summary <- list(
  species = unique(penguins$species),
  avg_bill_length = mean(penguins$bill_length_mm, na.rm = TRUE),
  sample_size = nrow(penguins),
  years = unique(penguins$year)
)
print(penguin_summary)
```

```
$species
[1] "Adelie"    "Gentoo"    "Chinstrap"

$avg_bill_length
[1] 43.92193

$sample_size
[1] 344

$years
[1] 2007 2008 2009
```

2.3 Importing Data

2.3.1 Reading Data Files

R provides several functions for importing data from different file formats:

```
# CSV files - Palmer Penguins dataset
penguins_csv <- read.csv("../data/environmental/climate_data.csv")
head(penguins_csv, 3)
```

```
  species    island bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
1  Adelie Torgersen           39.1          18.7               181        3750
2  Adelie Torgersen           39.5          17.4               186        3800
3  Adelie Torgersen           40.3          18.0               195        3250
     sex year
1   male 2007
2 female 2007
3 female 2007
```

```
# Using the tidyverse approach for better handling
library(tidyverse)
penguins_tidy <- readr::read_csv("../data/environmental/climate_data.csv")
head(penguins_tidy, 3)
```

```
# A tibble: 3 x 8
  species island    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
  <chr>   <chr>              <dbl>         <dbl>             <dbl>       <dbl>
1 Adelie  Torgersen           39.1          18.7               181        3750
2 Adelie  Torgersen           39.5          17.4               186        3800
3 Adelie  Torgersen           40.3          18                 195        3250
# i 2 more variables: sex <chr>, year <dbl>
```

```
# Crop yields dataset
crops_csv <- read.csv("../data/agriculture/crop_yields.csv")
head(crops_csv, 3)
```

```
       Entity Code Year Wheat..tonnes.per.hectare. Rice..tonnes.per.hectare.
1 Afghanistan  AFG 1961                     1.0220                     1.519
2 Afghanistan  AFG 1962                     0.9735                     1.519
3 Afghanistan  AFG 1963                     0.8317                     1.519
  Maize..tonnes.per.hectare. Soybeans..tonnes.per.hectare.
1                      1.400                            NA
2                      1.400                            NA
3                      1.426                            NA
  Potatoes..tonnes.per.hectare. Beans..tonnes.per.hectare.
1                        8.6667                         NA
2                        7.6667                         NA
3                        8.1333                         NA
  Peas..tonnes.per.hectare. Cassava..tonnes.per.hectare.
1                        NA                           NA
2                        NA                           NA
3                        NA                           NA
  Barley..tonnes.per.hectare. Cocoa.beans..tonnes.per.hectare.
1                        1.08                               NA
2                        1.08                               NA
3                        1.08                               NA
  Bananas..tonnes.per.hectare.
1                           NA
2                           NA
3                           NA
```

2.3.2 Exploring Real-World Datasets

Let's explore some of the real-world datasets we have available:

```r
# Palmer Penguins dataset
penguins <- read_csv("../data/environmental/climate_data.csv")
glimpse(penguins)
```

```
Rows: 344
Columns: 8
$ species           <chr> "Adelie", "Adelie", "Adelie", "Adelie", "Adelie", "A~
$ island            <chr> "Torgersen", "Torgersen", "Torgersen", "Torgersen", ~
$ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34.1, ~
$ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18.1, ~
$ flipper_length_mm <dbl> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, 186~
$ body_mass_g       <dbl> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 3475, ~
$ sex               <chr> "male", "female", "female", NA, "female", "male", "f~
$ year              <dbl> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007~
```

```r
# Basic summary statistics
summary(penguins$bill_length_mm)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  32.10   39.23   44.45   43.92   48.50   59.60       2 
```

```r
summary(penguins$flipper_length_mm)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  172.0   190.0   197.0   200.9   213.0   231.0       2 
```

```
# Crop yields dataset
crops <- read_csv("../data/agriculture/crop_yields.csv")
glimpse(crops)
```

```
Rows: 13,075
Columns: 14
$ Entity                             <chr> "Afghanistan", "Afghanistan", "Afgh~
$ Code                               <chr> "AFG", "AFG", "AFG", "AFG", "AFG", ~
$ Year                               <dbl> 1961, 1962, 1963, 1964, 1965, 1966,~
$ `Wheat (tonnes per hectare)`       <dbl> 1.0220, 0.9735, 0.8317, 0.9510, 0.9~
$ `Rice (tonnes per hectare)`        <dbl> 1.5190, 1.5190, 1.5190, 1.7273, 1.7~
$ `Maize (tonnes per hectare)`       <dbl> 1.4000, 1.4000, 1.4260, 1.4257, 1.4~
$ `Soybeans (tonnes per hectare)`    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Potatoes (tonnes per hectare)`    <dbl> 8.6667, 7.6667, 8.1333, 8.6000, 8.8~
$ `Beans (tonnes per hectare)`       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Peas (tonnes per hectare)`        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Cassava (tonnes per hectare)`     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Barley (tonnes per hectare)`      <dbl> 1.0800, 1.0800, 1.0800, 1.0857, 1.0~
$ `Cocoa beans (tonnes per hectare)` <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Bananas (tonnes per hectare)`     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA,~
```

2.4 Data Cleaning and Preparation

2.4.1 Handling Missing Values

Missing values are common in scientific datasets and need to be addressed before analysis:

```
# Check for missing values in the penguins dataset
sum(is.na(penguins))
```

```
[1] 19
```

```
colSums(is.na(penguins))
```

```
          species            island    bill_length_mm     bill_depth_mm
                0                 0                 2                 2
flipper_length_mm       body_mass_g               sex              year
                2                 2                11                 0
```

```
# Create a complete cases dataset
penguins_complete <- na.omit(penguins)
print(paste("Original dataset rows:", nrow(penguins)))
```

```
[1] "Original dataset rows: 344"
```

```
print(paste("Complete cases rows:", nrow(penguins_complete)))
```

```
[1] "Complete cases rows: 333"
```

```
# Replace missing values with the mean for numeric columns
penguins_imputed <- penguins
penguins_imputed$bill_length_mm[is.na(penguins_imputed$bill_length_mm)] <-
  mean(penguins_imputed$bill_length_mm, na.rm = TRUE)
penguins_imputed$bill_depth_mm[is.na(penguins_imputed$bill_depth_mm)] <-
  mean(penguins_imputed$bill_depth_mm, na.rm = TRUE)
```

```
# Check if missing values were replaced
sum(is.na(penguins_imputed$bill_length_mm))
```

```
[1] 0
```

2.4.2 Data Transformation

Often, you'll need to transform variables to meet statistical assumptions or for better visualization:

```
# Load the biodiversity dataset
biodiversity <- read_csv("../data/ecology/biodiversity.csv")
glimpse(biodiversity)
```

```
Rows: 500
Columns: 24
$ binomial_name     <chr> "Abutilon pitcairnense", "Acaena exigua", "Acalypha ~
$ country           <chr> "Pitcairn", "United States", "Congo", "Saint Helena,~
$ continent         <chr> "Oceania", "North America", "Africa", "Africa", "Oce~
$ group             <chr> "Flowering Plant", "Flowering Plant", "Flowering Pla~
$ year_last_seen    <chr> "2000-2020", "1980-1999", "1940-1959", "Before 1900"~
$ threat_AA         <dbl> 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1~
$ threat_BRU        <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0~
$ threat_RCD        <dbl> 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0~
$ threat_ISGD       <dbl> 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_EPM        <dbl> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_CC         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_HID        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_P          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_TS         <dbl> 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_NSM        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_GE         <dbl> 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ threat_NA         <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0~
$ action_LWP        <dbl> 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0~
$ action_SM         <dbl> 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ action_LP         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ action_RM         <dbl> 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ action_EA         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ action_NA         <dbl> 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1~
$ red_list_category <chr> "Extinct in the Wild", "Extinct", "Extinct", "Extinc~
```

```
# Log transformation of a skewed variable (if available)
if("n" %in% colnames(biodiversity)) {
  biodiversity$log_n <- log(biodiversity$n + 1)  # Add 1 to handle zeros

  # Compare original and transformed
  summary(biodiversity$n)
  summary(biodiversity$log_n)
}

# Standardization (z-score) of penguin measurements
penguins_std <- penguins %>%
  mutate(
    bill_length_std = scale(bill_length_mm),
    flipper_length_std = scale(flipper_length_mm),
    body_mass_std = scale(body_mass_g)
  )

# View the first few rows of the transformed data
head(select(penguins_std, species, bill_length_mm, bill_length_std,
             flipper_length_mm, flipper_length_std), 5)
```

```
# A tibble: 5 x 5
  species bill_length_mm bill_length_std[,1] flipper_length_mm
  <chr>            <dbl>               <dbl>             <dbl>
1 Adelie            39.1              -0.883               181
2 Adelie            39.5              -0.810               186
3 Adelie            40.3              -0.663               195
4 Adelie            NA                NA                    NA
5 Adelie            36.7              -1.32                193
# i 1 more variable: flipper_length_std <dbl[,1]>
```

2.4.3 Creating New Variables

Creating new variables from existing ones is a common data preparation task:

```
# Create new variables in the penguins dataset
penguins_derived <- penguins %>%
  filter(!is.na(bill_length_mm) & !is.na(bill_depth_mm)) %>%
  mutate(
    bill_ratio = bill_length_mm / bill_depth_mm,
    size_category = case_when(
      body_mass_g < 3500 ~ "Small",
      body_mass_g < 4500 ~ "Medium",
      TRUE ~ "Large"
    )
  )

# View the new variables
head(select(penguins_derived, species, bill_length_mm, bill_depth_mm,
             bill_ratio, body_mass_g, size_category), 5)
```

```
# A tibble: 5 x 6
  species bill_length_mm bill_depth_mm bill_ratio body_mass_g size_category
  <chr>            <dbl>         <dbl>      <dbl>       <dbl> <chr>
1 Adelie            39.1          18.7       2.09        3750 Medium
2 Adelie            39.5          17.4       2.27        3800 Medium
3 Adelie            40.3          18         2.24        3250 Small
4 Adelie            36.7          19.3       1.90        3450 Small
5 Adelie            39.3          20.6       1.91        3650 Medium
```

2.5 Data Manipulation with dplyr

The dplyr package provides a powerful grammar for data manipulation:

```
library(dplyr)

# Filter rows - only Adelie penguins
adelie_penguins <- penguins %>%
  filter(species == "Adelie")
```

```r
head(adelie_penguins, 3)
```

```
# A tibble: 3 x 8
  species island    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
  <chr>   <chr>              <dbl>         <dbl>             <dbl>       <dbl>
1 Adelie  Torgersen           39.1          18.7               181        3750
2 Adelie  Torgersen           39.5          17.4               186        3800
3 Adelie  Torgersen           40.3          18                 195        3250
# i 2 more variables: sex <chr>, year <dbl>
```

```r
# Select columns - focus on measurements
penguin_measurements <- penguins %>%
  select(species, island, bill_length_mm, bill_depth_mm, flipper_length_mm, body_mass_g)
head(penguin_measurements, 3)
```

```
# A tibble: 3 x 6
  species island    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
  <chr>   <chr>              <dbl>         <dbl>             <dbl>       <dbl>
1 Adelie  Torgersen           39.1          18.7               181        3750
2 Adelie  Torgersen           39.5          17.4               186        3800
3 Adelie  Torgersen           40.3          18                 195        3250
```

```r
# Create new variables
penguins_analyzed <- penguins %>%
  mutate(
    bill_ratio = bill_length_mm / bill_depth_mm,
    body_mass_kg = body_mass_g / 1000
  )
head(select(penguins_analyzed, species, bill_ratio, body_mass_kg), 3)
```

```
# A tibble: 3 x 3
  species bill_ratio body_mass_kg
  <chr>        <dbl>        <dbl>
1 Adelie        2.09         3.75
2 Adelie        2.27         3.8
3 Adelie        2.24         3.25
```

```r
# Summarize data by species
penguin_summary <- penguins %>%
  group_by(species) %>%
  summarize(
    count = n(),
    avg_bill_length = mean(bill_length_mm, na.rm = TRUE),
    avg_bill_depth = mean(bill_depth_mm, na.rm = TRUE),
    avg_body_mass = mean(body_mass_g, na.rm = TRUE)
  ) %>%
  arrange(desc(avg_body_mass))
print(penguin_summary)
```

```
# A tibble: 3 x 5
  species   count avg_bill_length avg_bill_depth avg_body_mass
  <chr>     <int>           <dbl>          <dbl>         <dbl>
1 Gentoo      124            47.5           15.0         5076.
2 Chinstrap    68            48.8           18.4         3733.
3 Adelie      152            38.8           18.3         3701.
```

```r
# Analyze crop yields data
crop_summary <- crops %>%
  filter(!is.na(`Wheat (tonnes per hectare)`)) %>%
  group_by(Entity) %>%
  summarize(
    years_recorded = n(),
    avg_wheat_yield = mean(`Wheat (tonnes per hectare)`, na.rm = TRUE),
    max_wheat_yield = max(`Wheat (tonnes per hectare)`, na.rm = TRUE)
  ) %>%
  arrange(desc(avg_wheat_yield)) %>%
  head(10)  # Top 10 countries by average wheat yield

print(crop_summary)
```

```
# A tibble: 10 x 4
   Entity           years_recorded avg_wheat_yield max_wheat_yield
   <chr>                     <int>           <dbl>           <dbl>
 1 Belgium                      19            8.54           10.0 
 2 Netherlands                  58            7.03            9.29
 3 Ireland                      58            6.83           10.7 
 4 United Kingdom               58            6.37            8.98
 5 Denmark                      58            6.18            8.24
 6 Luxembourg                   19            5.98            6.82
 7 Germany                      58            5.89            8.63
 8 Europe, Western              58            5.72            7.88
 9 France                       58            5.65            7.80
10 Northern Europe              58            5.59            7.21
```

2.6 Exploratory Data Analysis

Before diving into formal statistical tests, it's essential to explore your data:

```r
# Basic summary statistics
summary(penguins$bill_length_mm)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  32.10   39.23   44.45   43.92   48.50   59.60       2 
```

```r
summary(penguins$flipper_length_mm)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  172.0   190.0   197.0   200.9   213.0   231.0       2 
```

```r
summary(penguins$body_mass_g)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   2700    3550    4050    4202    4750    6300       2 
```

```r
# Correlation between variables
cor_matrix <- cor(
  penguins %>%
    select(bill_length_mm, bill_depth_mm, flipper_length_mm, body_mass_g),
  use = "complete.obs"
)
print(cor_matrix)
```