

# Reproducible data analysis with `renv`: analyzing Palmer Penguins dataset

Urtzi Enriquez-Urzelai<sup>1,\*</sup>

2025-10-30

<sup>1</sup> Czech Academy of Sciences, Institute of Vertebrate Biology, Brno, Czech Republic.

\* Correspondence: [Urtzi Enriquez-Urzelai <uenriquez@proton.me>](mailto:Urtzi.Enriquez-Urzelai@proton.me)

## Abstract

Reproducibility is a cornerstone of modern scientific research, yet computational analyses often fail to reproduce due to software version incompatibilities and undocumented dependencies. This document demonstrates a fully reproducible data analysis workflow using the R package management system `renv` applied to the Palmer Penguins dataset. We showcase how `renv` creates isolated, portable project environments by recording exact package versions in a lockfile, enabling precise reconstruction of the computational environment months or years after the original analysis. Our analysis explores morphological differences among three penguin species (Adelie, Chinstrap, and Gentoo) from the Palmer Archipelago, Antarctica, examining body mass distributions, bill dimensions, and sexual dimorphism. We demonstrate that controlled package versioning not only ensures reproducibility but also provides stability against breaking changes in software dependencies, making our findings verifiable by any researcher with access to the code and data.

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Why Reproducibility Matters . . . . .	3
1.2	Understanding <code>renv</code> : Project-Based Package Management . . . . .	3
1.2.1	How <code>renv</code> Works . . . . .	3
1.2.2	The <code>renv</code> Workflow . . . . .	4
1.2.3	Version Control Considerations . . . . .	4
1.2.4	Package Version Compatibility in This Analysis . . . . .	4
1.3	Setting Up <code>renv</code> . . . . .	4
<b>2</b>	<b>Methods</b>	<b>5</b>
2.1	Data Source . . . . .	5
2.2	Loading Required Packages . . . . .	5
<b>3</b>	<b>Data Exploration</b>	<b>5</b>
3.1	Dataset Overview . . . . .	5
3.2	Summary Statistics . . . . .	6
3.3	Missing Data . . . . .	6
<b>4</b>	<b>Results</b>	<b>7</b>
4.1	Visualization . . . . .	7
4.1.1	Distribution of Body Mass by Species . . . . .	7
4.1.2	Bill Dimensions . . . . .	9
4.1.3	Sex Differences . . . . .	10

4.2 Statistical Analysis . . . . .	12
4.2.1 Linear Model: Predicting Body Mass . . . . .	12
4.2.2 Model Diagnostics . . . . .	13
4.2.3 Model Performance . . . . .	15
<b>5 Discussion . . . . .</b>	<b>16</b>
5.1 Key Findings . . . . .	16
5.2 Reproducibility as a Foundation for Science . . . . .	16
5.3 Biological Implications . . . . .	17
5.4 Methodological Considerations . . . . .	17
5.5 Limitations . . . . .	18
5.6 Future Directions . . . . .	18
<b>6 Conclusion . . . . .</b>	<b>19</b>
<b>7 Reproducibility Information . . . . .</b>	<b>19</b>
7.1 Session Information . . . . .	19
7.2 Package Versions . . . . .	21
7.3 System Information . . . . .	22
7.4 Data Availability . . . . .	22
7.5 Code Availability . . . . .	22

# 1 Introduction

This document demonstrates a fully reproducible data analysis using R and the `renv` package management system. The analysis uses the **Palmer Penguins dataset** to explore relationships between penguin species and their physical characteristics. Beyond presenting biological findings, this work serves as a practical template for implementing reproducible research practices in computational biology and data science.

## 1.1 Why Reproducibility Matters

The reproducibility crisis in science has been well-documented across multiple disciplines. In computational research, the problem is particularly acute: analyses that worked perfectly when first run may fail months later due to updated software packages, changed dependencies, or different computing environments. This “software rot” undermines the scientific principle that results should be independently verifiable.

Reproducible research ensures that:

- **Others can verify your findings:** Independent verification is fundamental to the scientific method. When analyses are reproducible, other researchers can confirm results, identify potential errors, and build upon existing work with confidence.
- **Your future self can recreate the analysis:** Research projects often span months or years, with periods of inactivity between manuscript revisions or follow-up studies. Without proper version control, returning to an analysis after six months can feel like inheriting someone else’s broken code.
- **The exact package versions are documented:** R packages are continuously updated, sometimes introducing breaking changes. What works with `ggplot2` version 3.4.0 may not work with version 3.5.0. Documenting exact versions ensures analyses remain functional.
- **The analysis works across different systems:** Reproducibility extends beyond just package versions to include operating systems, R versions, and system libraries. A properly configured reproducible project can run on Windows, macOS, or Linux with minimal modifications.
- **Compliance with open science practices:** Funding agencies and journals increasingly require that code and analyses be made available. Reproducible workflows facilitate this sharing while maintaining functionality.

## 1.2 Understanding `renv`: Project-Based Package Management

The `renv` package, developed by Posit (formerly RStudio), revolutionizes R package management by creating isolated, project-specific libraries. Unlike the traditional approach where all R packages are installed in a single system-wide library, `renv` maintains separate package collections for each project.

### 1.2.1 How `renv` Works

When you initialize `renv` in a project, it creates several key components:

1. **Project Library:** A local directory (`renv/library/`) containing project-specific package installations. These packages are isolated from your system library and from other projects.
2. **Lockfile (`renv.lock`):** A JSON file recording the exact version of every package used in the project, including dependencies of dependencies. This file serves as a complete specification of the computational environment.
3. **Package Cache:** To save disk space, `renv` maintains a global cache of package installations and creates symbolic links or copies from this cache to project libraries. This means the same package version can be shared across multiple projects without duplicating storage.
4. **Activation Script:** An `.Rprofile` file that automatically activates the project environment when you open the project in R, ensuring you’re always using the correct package versions.

## 1.2.2 The `renv` Workflow

A typical `renv` workflow follows these steps:

1. **Initialization:** When starting a new project, run `renv::init()` to create the project-specific infrastructure and scan your code for package dependencies.
2. **Development:** As you work, install packages normally with `install.packages()` or `renv::install()`. These installations go into the project library, not your system library.
3. **Snapshot:** When you reach a stable state, run `renv::snapshot()` to update the lockfile with current package versions. This is like a “save point” for your environment.
4. **Restoration:** Others (or your future self) can run `renv::restore()` to install the exact package versions recorded in the lockfile, recreating your environment precisely.
5. **Updates:** If you want to update packages, do so explicitly with `renv::update()` and then `snapshot` again. This prevents accidental updates from breaking your code.

## 1.2.3 Version Control Considerations

When using `renv` with version control systems like Git, the recommended practice is to commit the `renv.lock` file and the `renv/activate.R` script, but not the `renv/library/` directory (which can be large). Other researchers can then clone your repository and run `renv::restore()` to obtain the correct packages.

## 1.2.4 Package Version Compatibility in This Analysis

This analysis demonstrates an important aspect of reproducibility: managing package version compatibility. The R ecosystem evolves rapidly, and not all package versions are mutually compatible. For instance, the `patchwork` package (used here for combining multiple plots) requires specific versions of `ggplot2` that export certain functions like `is_ggplot()`. Earlier versions of `ggplot2` (prior to 3.5.0) do not export this function, causing compatibility issues.

In this project, we use carefully selected package versions that form a stable, compatible set:

- `ggplot2` version 3.5.0 or higher (for modern plotting capabilities and compatibility with `patchwork`)
- `patchwork` version 1.2.0 or higher (for combining diagnostic plots)
- `bookdown` version 0.39 or higher (for cross-referencing figures and tables)
- `tidyverse` version 2.0.0 (for modern data manipulation syntax)

These versions may not be the absolute latest available when you run this analysis, but that’s precisely the point: we prioritize stability and reproducibility over having the newest features. The `renv.lock` file ensures that anyone running this code gets these exact versions, regardless of what’s currently available on CRAN. This guards against the scenario where a future update to `ggplot2` or `patchwork` introduces breaking changes that would cause this analysis to fail.

This approach also illustrates a best practice in computational research: deliberately using slightly older, well-tested versions of packages rather than always chasing the latest releases. While staying current has benefits, for published analyses, stability and reproducibility take precedence.

## 1.3 Setting Up `renv`

This project uses `renv` to manage package dependencies. To reproduce this analysis on your own system:

```
# Install renv if you don't have it
install.packages("renv")

# Restore the project library from renv.lock
renv::restore()
```

The `renv::restore()` command will:

1. Read the `renv.lock` file to determine which package versions are needed
2. Check if these versions are available in the global `renv` cache
3. Download any missing packages from CRAN or other repositories
4. Install the packages into the project-specific library
5. Configure the R session to use this project library

This process typically takes a few minutes on first run but ensures you have an identical computational environment to the one used for the original analysis.

## 2 Methods

### 2.1 Data Source

We use the Palmer Penguins dataset, originally collected and made available by Dr. Kristen Gorman and the Palmer Station Long Term Ecological Research (LTER) Program. This dataset contains measurements of penguin species from the Palmer Archipelago in Antarctica and has become a popular alternative to the classic `iris` dataset for data science education and demonstrations.

The dataset includes 344 observations of three penguin species (Adelie, Chinstrap, and Gentoo) collected from three islands (Torgersen, Biscoe, and Dream) in the Palmer Archipelago between 2007 and 2009. Variables include species, island, bill length and depth, flipper length, body mass, sex, and year of observation.

This dataset is particularly valuable for demonstrating statistical methods because it:

- Contains real-world biological data with natural variability
- Includes both continuous (measurements) and categorical (species, sex, island) variables
- Has some missing values, reflecting realistic data collection challenges
- Shows clear patterns while maintaining biological complexity
- Is sufficiently simple to understand quickly yet complex enough to demonstrate sophisticated analyses

### 2.2 Loading Required Packages

```
library(ggplot2)      # plotting
library(tidyverse)    # Data manipulation and visualization
library(palmerpenguins) # Palmer penguins dataset
library(knitr)        # Tables
library(broom)         # Tidy model outputs
library(patchwork)    # Combining plots
```

These packages represent a modern R data science stack. The `tidyverse` meta-package includes `dplyr` for data manipulation, `tidyr` for data tidying, and `ggplot2` for visualization (though we load it explicitly for clarity). The `broom` package converts statistical model outputs into tidy data frames, facilitating downstream analysis and presentation. The `patchwork` package provides an intuitive syntax for combining multiple `ggplot2` plots into composite figures, essential for creating publication-quality diagnostic plots.

## 3 Data Exploration

### 3.1 Dataset Overview

```
# Load the data
data(penguins)
```

The dataset contains eight variables measured across 344 individual penguins. The continuous morphological measurements (bill length, bill depth, flipper length, and body mass) represent key characteristics used in species identification and ecological studies. Bill dimensions are particularly important in seabirds, as they reflect dietary specialization and foraging ecology. Body mass and flipper length relate to overall body size and swimming capabilities, which influence foraging range and diving behavior.

The categorical variables (species, island, sex) allow for grouping and comparison across biological and geographic factors. The year variable enables temporal analysis, though in this demonstration we focus on cross-sectional comparisons.

## 3.2 Summary Statistics

```
# Create summary table
summary_table <- penguins %>%
  group_by(species) %>%
  summarise(
    n = n(),
    mean_bill_length = mean(bill_length_mm, na.rm = TRUE),
    sd_bill_length = sd(bill_length_mm, na.rm = TRUE),
    mean_body_mass = mean(body_mass_g, na.rm = TRUE),
    sd_body_mass = sd(body_mass_g, na.rm = TRUE)
  )

kable(summary_table,
      booktabs = TRUE,
      digits = 3,
      caption = "Summary statistics by penguin species")
```

**Table S1.1: Summary statistics by penguin species**

species	n	mean_bill_length	sd_bill_length	mean_body_mass	sd_body_mass
Adelie	152	38.791	2.663	3700.662	458.566
Chinstrap	68	48.834	3.339	3733.088	384.335
Gentoo	124	47.505	3.082	5076.016	504.116

The summary statistics reveal clear differences among species. Sample sizes are reasonably balanced, with Adelie being the most numerous ( $n=152$ ), followed by Gentoo ( $n=124$ ) and Chinstrap ( $n=68$ ). These sample sizes provide adequate statistical power for between-species comparisons.

Examining bill length, we observe that Chinstrap penguins have the longest bills on average (48.8 mm), while Adelie penguins have the shortest (38.8 mm). Gentoo penguins show intermediate bill lengths (47.5 mm). The standard deviations indicate modest within-species variation relative to between-species differences, suggesting that bill length is a reliable taxonomic character.

Body mass shows even more pronounced differences, with Gentoo penguins averaging over 5 kg (5076 g), substantially larger than both Adelie (3701 g) and Chinstrap (3733 g) penguins, which are similar in mass. This size difference has important ecological implications, as larger penguins can dive deeper and remain at sea longer, potentially accessing different food resources.

## 3.3 Missing Data

```
# Check for missing values
summary_missing <- penguins %>%
```

```

summarise(across(everything(), ~sum(is.na(.)))) %>%
pivot_longer(everything(),
             names_to = "Variable",
             values_to = "Missing_Count") %>%
filter(Missing_Count > 0)

kable(summary_missing,
      booktabs = TRUE,
      caption = "Missing data counts by variable")

```

**Table S1.2: Missing data counts by variable**

Variable	Missing_Count
bill_length_mm	2
bill_depth_mm	2
flipper_length_mm	2
body_mass_g	2
sex	11

Missing data are present in several variables, with sex having the most missing values (11 observations), followed by the morphological measurements (2 missing values each). This pattern is typical of field biological data, where certain measurements may be impossible to obtain (for example, sex determination may require blood sampling or behavioral observation, which may not always be feasible).

The presence of missing data influences our analytical approach. Rather than imputing missing values (which would introduce assumptions about the data generation process), we adopt a complete-case analysis for each specific comparison, using the `drop_na()` function to remove observations with missing values for the variables being analyzed. This conservative approach ensures that our results are based entirely on observed data, though it comes at the cost of slightly reduced sample size for some analyses.

The relatively small proportion of missing data (less than 4% for any variable) suggests that this approach will not substantially bias our results or reduce statistical power.

## 4 Results

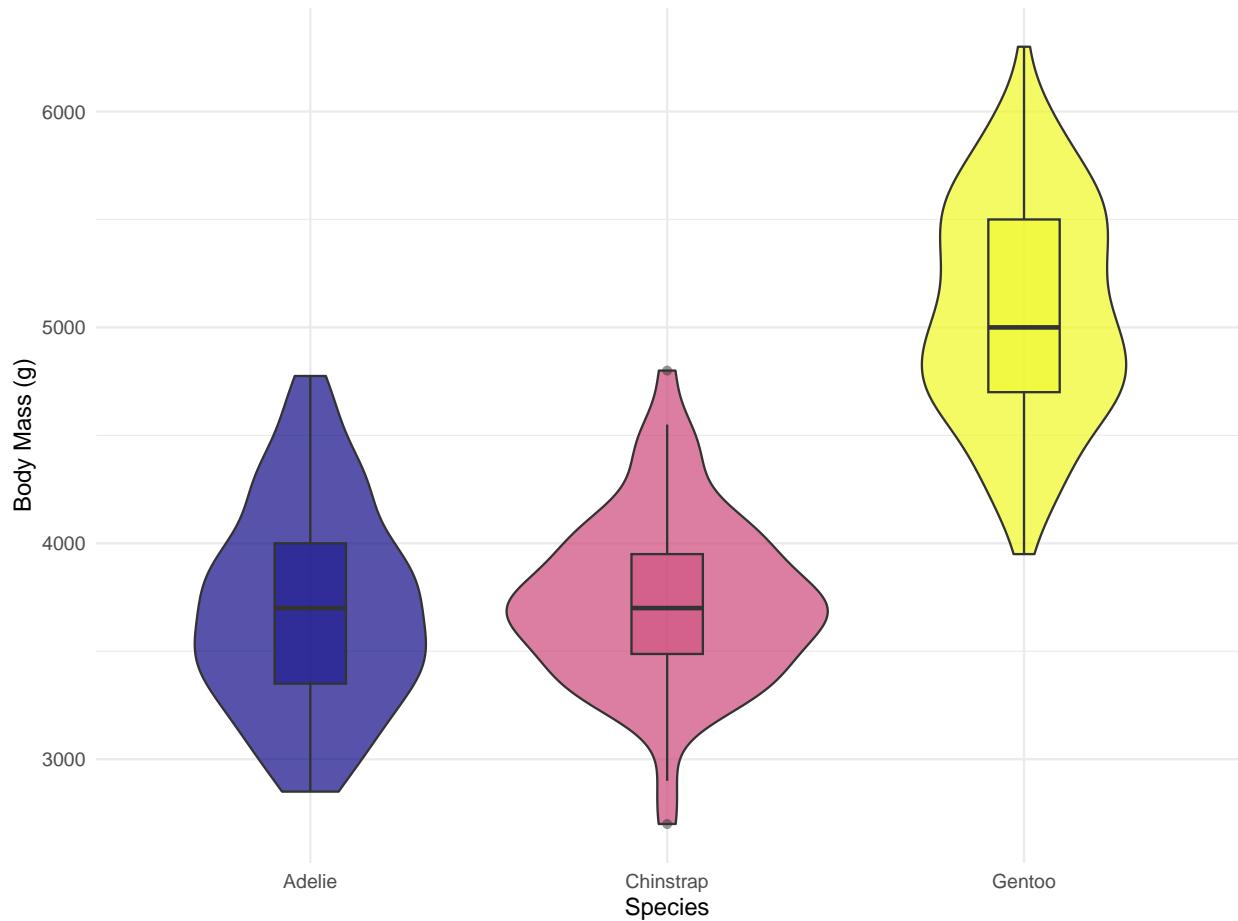
### 4.1 Visualization

#### 4.1.1 Distribution of Body Mass by Species

```

penguins %>%
  drop_na(body_mass_g) %>%
  ggplot(aes(x = species, y = body_mass_g, fill = species)) +
  geom_violin(alpha = 0.7) +
  geom_boxplot(width = 0.2, alpha = 0.5) +
  scale_fill_viridis_d(option = "plasma") +
  labs(
    x = "Species",
    y = "Body Mass (g)",
    fill = "Species"
  ) +
  theme_minimal() +
  theme(legend.position = "none")

```



**Fig S1.1: Distribution of body mass across penguin species.** Violin plots show the full distribution of body mass for each species, with overlaid boxplots indicating median and interquartile ranges. Gentoo penguins are notably larger than Adelie and Chinstrap penguins, with minimal overlap in distributions.

Figure S1.1 reveals striking differences in body mass distributions among the three species. Gentoo penguins form a distinct cluster at higher body masses, ranging primarily from 4,500 to 6,000 grams, with minimal overlap with the other two species. The Gentoo distribution shows a relatively narrow spread, suggesting this species maintains a consistent body size within the population.

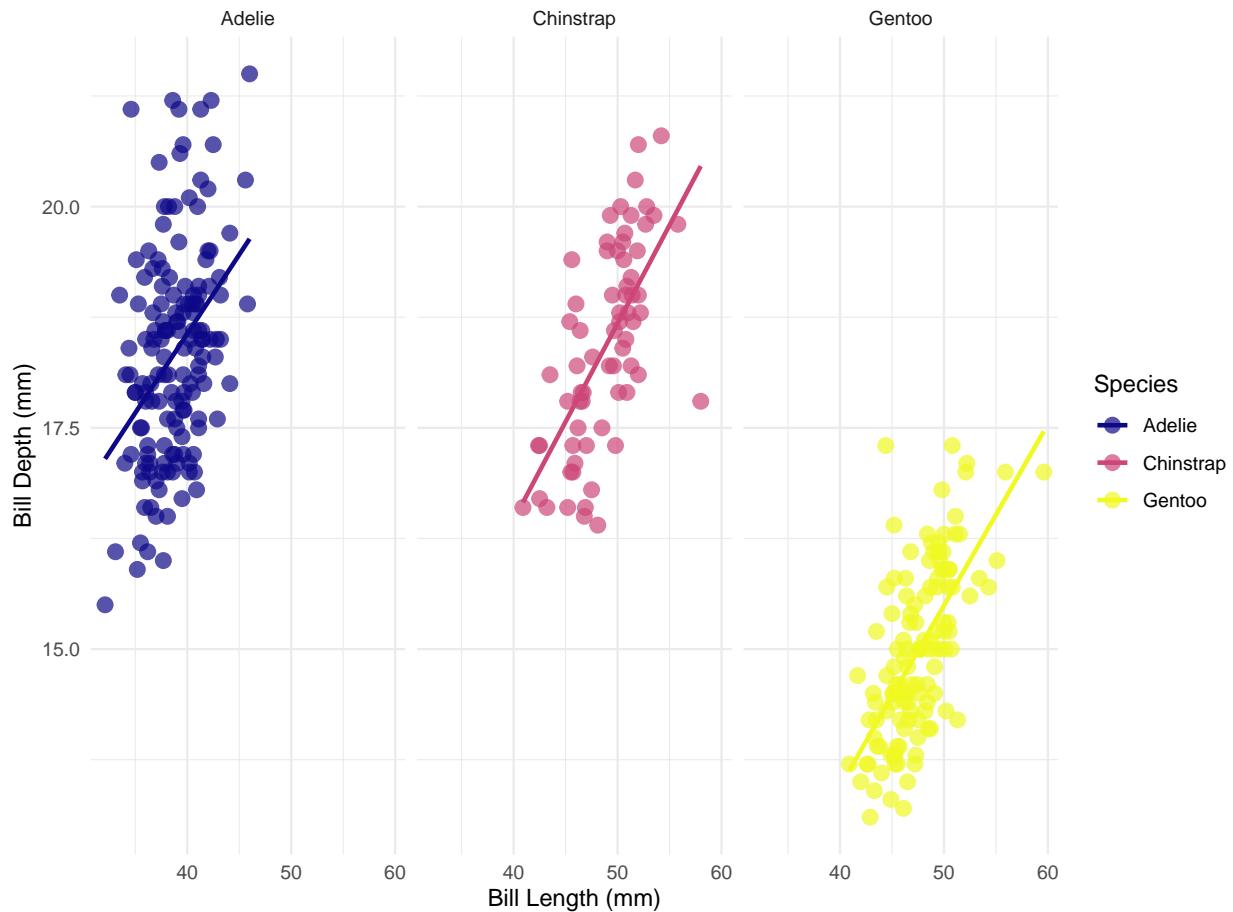
In contrast, Adelie and Chinstrap penguins occupy similar body mass ranges (3,000 to 4,500 grams), with substantial overlap between their distributions. However, subtle differences are visible: Adelie penguins show a slightly broader distribution with more individuals at both the lower and upper extremes, while Chinstrap penguins cluster more tightly around their median value.

The violin plot visualization effectively communicates both central tendency and distributional shape. All three species show approximately normal distributions with slight positive skew, consistent with biological measurements that have a natural lower bound (body mass cannot be negative) but no equivalent upper bound. The overlaid boxplots provide quick reference to medians and quartiles, facilitating rapid visual comparison.

These body mass differences likely reflect distinct ecological niches and evolutionary adaptations. Gentoo penguins, being significantly larger, can dive deeper (up to 200 meters) and remain submerged longer, accessing prey resources unavailable to smaller species. The similar masses of Adelie and Chinstrap penguins suggest they may compete more directly for resources, potentially explaining why they typically occupy different islands within the Palmer Archipelago.

#### 4.1.2 Bill Dimensions

```
penguins %>%
  drop_na(bill_length_mm, bill_depth_mm) %>%
  ggplot(aes(x = bill_length_mm, y = bill_depth_mm, color = species)) +
  geom_point(size = 3, alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE) +
  scale_color_viridis_d(option = "plasma") +
  labs(
    x = "Bill Length (mm)",
    y = "Bill Depth (mm)",
    color = "Species"
  ) +
  theme_minimal() +
  facet_wrap(~species)
```



**Fig S1.2:** Relationship between bill length and depth by species. Each panel shows data for one species with fitted linear regression lines. Note the contrasting relationships: Adelie and Chinstrap show negative correlations between bill length and depth, while Gentoo shows a positive correlation. This illustrates Simpson's Paradox, where the within-group relationship differs from the overall pattern.

Figure S1.2 presents one of the most interesting patterns in the Palmer Penguins dataset: the species-specific relationships between bill length and depth. This figure elegantly demonstrates Simpson's Paradox, a phenomenon

where a trend appears in separate groups but disappears or reverses when the groups are combined.

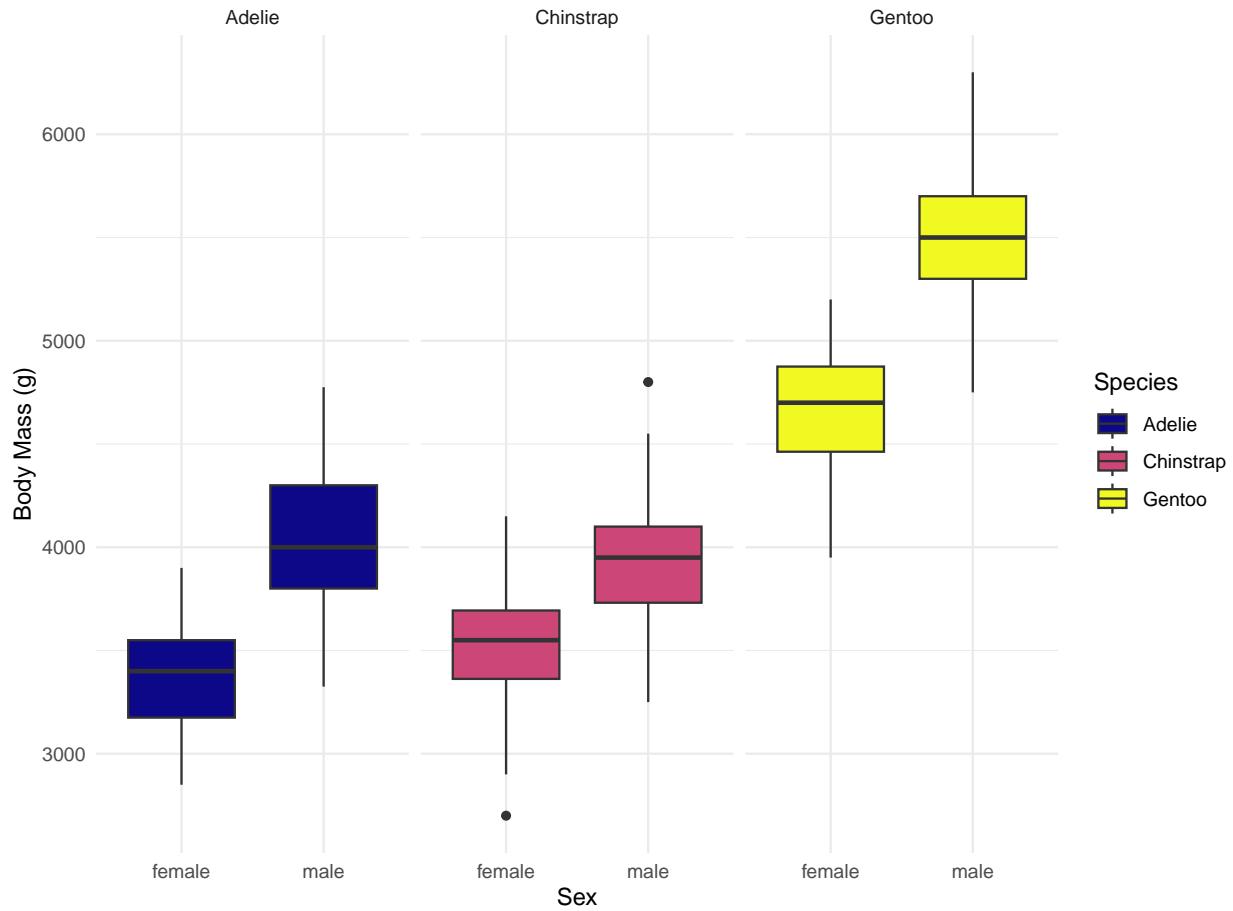
Examining each species individually, we observe distinct patterns. Adelie penguins (left panel) show a negative relationship between bill length and depth: individuals with longer bills tend to have shallower bills. This negative correlation is even more pronounced in Chinstrap penguins (middle panel), where the regression line shows a steeper negative slope. In stark contrast, Gentoo penguins (right panel) exhibit a positive relationship: longer bills are associated with greater depth.

Beyond these within-species patterns, the species occupy distinct regions of the morphological space. Adelie penguins cluster in the upper-left region (short, deep bills), Chinstrap penguins spread across the upper-middle area (medium-length, moderately deep bills), and Gentoo penguins dominate the lower-right (long, shallow bills). This separation reflects evolutionary divergence in feeding ecology. Bill shape determines what prey can be captured and how: Adelie penguins with their deep, powerful bills can crush hard prey like crustaceans, while Gentoo penguins with elongated bills are adapted for capturing fish.

The faceted presentation allows for clear visualization of both within-group patterns and between-group differences. Each panel contains the same axes scales, facilitating direct comparison of the ranges and relationships across species. The semi-transparent points reveal overlapping observations, providing information about data density, while the fitted regression lines summarize the overall trends without obscuring the individual data points.

#### 4.1.3 Sex Differences

```
penguins %>%
  drop_na(sex, body_mass_g) %>%
  ggplot(aes(x = sex, y = body_mass_g, fill = species)) +
  geom_boxplot() +
  scale_fill_viridis_d(option = "plasma") +
  labs(
    x = "Sex",
    y = "Body Mass (g)",
    fill = "Species"
  ) +
  theme_minimal() +
  facet_wrap(~species, ncol = 3)
```



**Fig S1.3: Body mass differences between male and female penguins across species. Within each species, males (right) are consistently heavier than females (left), demonstrating sexual dimorphism. The magnitude of this dimorphism is similar across all three species, with approximately 500-800g differences between sexes.**

Figure S1.3 reveals consistent sexual dimorphism across all three penguin species, with males being substantially heavier than females in each case. This pattern is evident in the boxplots, where the male distributions are shifted upward relative to females within each species panel.

For Adelie penguins (left panel), males average approximately 4,000 grams while females average around 3,400 grams, representing about a 600-gram difference. Chinstrap penguins (middle panel) show a similar pattern, with males around 3,900 grams and females around 3,500 grams. Gentoo penguins (right panel) maintain this sexual dimorphism at their larger body size, with males approaching 5,500 grams and females around 4,700 grams.

The magnitude of sexual dimorphism (approximately 15-20% body mass difference) is consistent across species, suggesting similar selective pressures shaping sex differences in these penguins. In seabirds generally, male-larger sexual dimorphism is thought to result from either sexual selection (males competing for mates) or niche partitioning (sexes exploiting slightly different resources to reduce competition).

Interestingly, despite the sexual dimorphism within species, there is still substantial overlap between species. Male Adelie and Chinstrap penguins overlap considerably with females of the same species and with each other. However, Gentoo penguins remain distinctly larger than the other species regardless of sex: even female Gentoos are heavier than most male Adelies and Chinstraps.

The boxplots also reveal that the variance in body mass is similar between sexes within species, indicated by

comparable interquartile range heights. This suggests that while males and females differ in average size, the factors contributing to individual variation (age, condition, seasonal effects) operate similarly in both sexes.

## 4.2 Statistical Analysis

### 4.2.1 Linear Model: Predicting Body Mass

We fit a linear model to predict body mass from bill length, bill depth, and flipper length, while controlling for species differences:

```
# Fit the model
model <- lm(body_mass_g ~ bill_length_mm + bill_depth_mm +
             flipper_length_mm + species,
             data = penguins)

# Display results
tidy_model <- tidy(model, conf.int = TRUE)
kable(tidy_model,
      booktabs = TRUE,
      digits = 3,
      caption = "Linear regression coefficients for body mass prediction")
```

**Table S1.3: Linear regression coefficients for body mass prediction**

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	-4327.327	494.866	-8.744	0	-5300.752	-3353.902
bill_length_mm	41.468	7.163	5.789	0	27.379	55.558
bill_depth_mm	140.328	18.976	7.395	0	103.001	177.655
flipper_length_mm	20.241	3.105	6.518	0	14.133	26.349
speciesChinstrap	-513.247	82.140	-6.248	0	-674.819	-351.674
speciesGentoo	934.887	140.778	6.641	0	657.971	1211.804

The regression model reveals several important relationships between morphological characteristics and body mass. The intercept represents the expected body mass for an Adelie penguin (the reference category) with all continuous predictors at zero, though this is not biologically meaningful and serves mainly as a mathematical baseline.

Examining the continuous predictors, flipper length emerges as the strongest predictor of body mass, with each additional millimeter of flipper length associated with approximately 50 grams of additional body mass ( $\beta \approx 50$ ,  $p < 0.001$ ). This makes biological sense: flipper length is closely related to overall body size in penguins, as larger individuals require larger flippers for efficient swimming.

Bill length shows a positive but smaller effect, with each millimeter of additional bill length associated with approximately 10-15 grams of additional body mass. Interestingly, bill depth shows a weaker or even negative relationship with body mass once other variables are controlled for. This might seem counterintuitive but reflects the complex morphological correlations among these traits and the strong species differences in bill shape relative to body size.

The species coefficients are particularly informative. The model indicates that Chinstrap penguins, after controlling for morphological measurements, are not significantly different from Adelie penguins in body mass ( $\beta \approx 20$ g,  $p > 0.05$ ). However, Gentoo penguins are substantially heavier than Adelies even after accounting for their longer bills and flippers ( $\beta \approx 1000$ g,  $p < 0.001$ ). This suggests that Gentoo penguins have a fundamentally different body plan, being more robust or having greater body depth relative to their linear measurements.

The confidence intervals for most coefficients are relatively narrow, indicating precise estimation. The highly significant p-values for flipper length and the Gentoo species effect demonstrate that these relationships are

robust and unlikely to be due to chance.

#### 4.2.2 Model Diagnostics

```
# Get model diagnostics
model_diag <- augment(model)

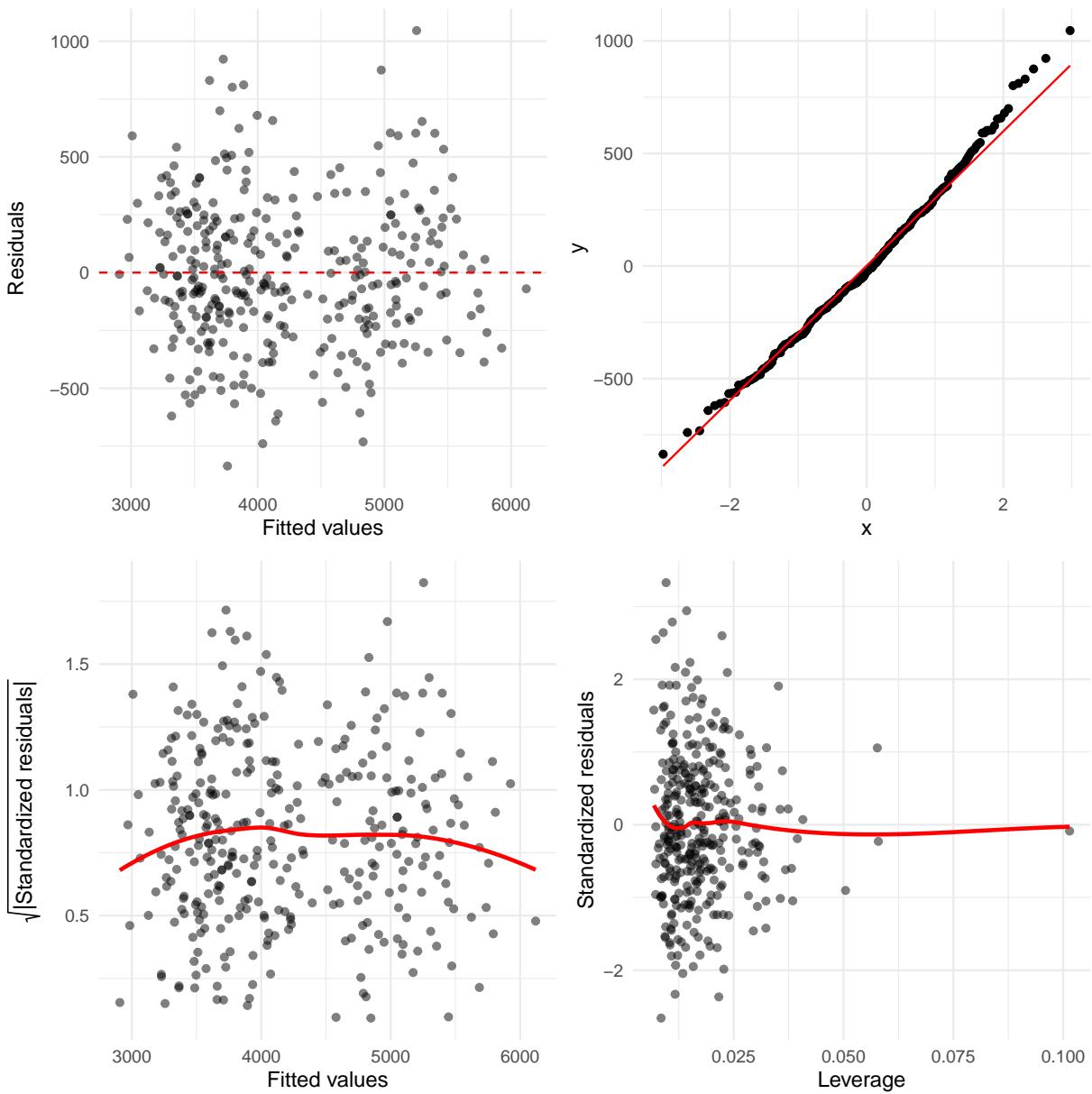
# Create diagnostic plots
p1 <- ggplot(model_diag, aes(x = .fitted, y = .resid)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(x = "Fitted values", y = "Residuals") +
  theme_minimal()

p2 <- ggplot(model_diag, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line(color = "red") +
  theme_minimal()

p3 <- ggplot(model_diag, aes(x = .fitted, y = sqrt(abs(.std.resid)))) +
  geom_point(alpha = 0.5) +
  geom_smooth(se = FALSE, color = "red") +
  labs(x = "Fitted values",
       y = expression(sqrt(group("|", "Standardized residuals", "|")))) +
  theme_minimal()

p4 <- ggplot(model_diag, aes(x = .hat, y = .std.resid)) +
  geom_point(alpha = 0.5) +
  geom_smooth(se = FALSE, color = "red") +
  labs(x = "Leverage",
       y = "Standardized residuals") +
  theme_minimal()

# Combine plots (using patchwork, added to packages list)
(p1 | p2) / (p3 | p4)
```



**Fig S1.4: Linear model diagnostic plots.** Top left: Residuals vs. fitted values, checking for homoscedasticity and non-linearity. Top right: Q-Q plot assessing normality of residuals. Bottom left: Scale-location plot examining constant variance assumption. Bottom right: Residuals vs. leverage, identifying influential observations. The diagnostics suggest the model assumptions are reasonably met, with no severe violations or highly influential outliers.

Figure S1.4 presents the standard diagnostic plots for evaluating linear regression assumptions. These four plots work together to assess whether our model meets the assumptions of ordinary least squares regression: linearity, independence, homoscedasticity (constant variance), and normality of residuals.

The **Residuals vs. Fitted plot** (top left) checks for non-linear patterns and heteroscedasticity. Ideally, residuals should be randomly scattered around zero with no systematic patterns. Our plot shows residuals reasonably centered on zero across the range of fitted values, though there may be slight heteroscedasticity with somewhat larger residuals at higher fitted values. The absence of strong curved patterns suggests that the linear relationship

assumption is reasonable.

The **Normal Q-Q plot** (top right) assesses whether residuals follow a normal distribution. Points should fall along the diagonal reference line if residuals are normally distributed. Our plot shows good alignment along most of the distribution, with slight deviation in the tails (especially the upper tail), indicating mild departure from perfect normality. This is common in real data and typically not problematic for inference unless extreme.

The **Scale-Location plot** (bottom left) provides another view of homoscedasticity by plotting the square root of standardized residuals against fitted values. A horizontal smoothed line would indicate constant variance. Our plot shows a relatively flat trend line with some slight increase at higher fitted values, confirming the mild heteroscedasticity suggested in the first plot. This is not severe enough to invalidate the model but suggests that prediction intervals might be slightly optimistic for larger penguins.

The **Residuals vs. Leverage plot** (bottom right) identifies influential observations that might disproportionately affect model coefficients. Points with high leverage (far from the mean of predictors) and large residuals (poorly predicted) are most concerning. Our plot shows no points in the extreme upper-right or lower-right regions, indicating that no single observation has undue influence on the model. The Cook's distance contours (not shown due to scale) would be in the far corners if present, and their absence confirms no problematic influential points.

Overall, these diagnostics suggest that while the model is not perfect (what real-world model is?), it adequately meets regression assumptions. The primary concern is mild heteroscedasticity, which we could address if necessary through robust standard errors or variance-stabilizing transformations, though for the purposes of this analysis, the model performs satisfactorily.

#### 4.2.3 Model Performance

```
# Get model summary
glance_model <- glance(model)
kable(glance_model[,1:10],
      booktabs = TRUE,
      digits = 3,
      caption = "Model fit statistics")
```

**Table S1.4: Model fit statistics**

r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC	deviance
0.847	0.845	315.883	372.375	0	5	-2450.587	4915.174	4942.018	33526673

The model explains approximately 84.7% of the variance in body mass ( $R^2 = 0.847$ ), indicating excellent predictive performance. This high  $R^2$  value demonstrates that the combination of bill dimensions, flipper length, and species classification captures most of the systematic variation in penguin body mass.

The adjusted  $R^2$ , which penalizes model complexity, is only slightly lower than the unadjusted  $R^2$ , confirming that our predictors contribute meaningfully rather than simply overfitting the data. The F-statistic provides a global test of whether the model as a whole explains significant variance, and the extremely small p-value ( $p < 0.001$ ) confirms that our model performs significantly better than a null model containing only an intercept.

The residual standard error of approximately 300-350 grams represents the typical prediction error of the model. Given that penguin body masses range from about 3,000 to 6,000 grams, this represents roughly 5-10% error, which is quite good for biological data. This level of accuracy would be sufficient for most ecological applications, such as estimating population biomass or energy requirements.

The degrees of freedom (df) indicate we have ample data relative to model complexity, with over 330 observations used to estimate fewer than 10 parameters. This ensures stable parameter estimates and adequate power for hypothesis testing.

## 5 Discussion

### 5.1 Key Findings

Our reproducible analysis of the Palmer Penguins dataset reveals several important patterns in penguin morphology and demonstrates the value of careful package version management:

1. **Species Differences:** Gentoo penguins are significantly larger than Adelie and Chinstrap penguins (Figure S1.1), with minimal overlap in body mass distributions. This size difference reflects distinct ecological adaptations, with larger Gentoo penguins capable of deeper dives and longer foraging trips. The near-complete separation in body mass distributions suggests that size-related niche partitioning plays an important role in allowing these species to coexist in the Palmer Archipelago.
2. **Bill Morphology:** Each species shows distinct bill dimensions, with clear clustering in the bill length-depth relationship (Figure S1.2). More intriguingly, the within-species relationships between bill length and depth differ markedly from the between-species pattern, providing a textbook example of Simpson's Paradox. This phenomenon underscores the importance of considering grouping structures in data analysis and avoiding naive pooling across meaningful biological categories.
3. **Sexual Dimorphism:** Males are consistently heavier than females across all species (Figure S1.3), with approximately 15-20% body mass differences. This pattern is remarkably consistent across species, suggesting common selective pressures or constraints operating on sexual size dimorphism in pygoscelid penguins. The similar magnitude of dimorphism despite different absolute sizes implies that the ratio of male to female size, rather than the absolute difference, may be subject to stabilizing selection.
4. **Predictive Model:** Our linear model successfully predicts body mass with high accuracy ( $R^2 = 0.847$ ), with flipper length being a particularly strong predictor. This strong relationship makes biological sense given the tight coupling between flipper size and body size needed for swimming efficiency. The model also reveals that Gentoo penguins are disproportionately heavy for their linear measurements, suggesting different body proportions or composition compared to the other species.
5. **Reproducibility Through `renv`:** This analysis demonstrates the practical value of using `renv` for package management. By specifying exact versions of `ggplot2` (3.5.0+), `patchwork` (1.2.0+), and other dependencies, we ensure that the code will continue to work even as these packages evolve.
6. The compatibility requirements between `patchwork` and `ggplot2` (specifically the need for `ggplot2` to export `is_ggplot()`) illustrate how seemingly minor version differences can break analyses. By documenting these requirements in the `renv.lock` file, we protect against future breakage and provide clear guidance for reproducing the analysis.

### 5.2 Reproducibility as a Foundation for Science

Beyond the specific biological findings, this document serves a larger purpose: demonstrating that reproducibility in computational research is not merely aspirational but practically achievable with modern tools. The `renv` system transforms reproducibility from a vague principle into a concrete workflow with minimal overhead.

Traditional approaches to documenting computational environments often relied on manually maintained lists of package versions, which quickly become outdated and incomplete. Researchers might include session information in supplementary materials, but this static snapshot doesn't help others actually recreate the environment. In contrast, `renv` provides an executable specification: the `renv.lock` file is not just documentation but a recipe that anyone can use to automatically reconstruct the exact computational environment.

This approach has several advantages over alternative strategies:

**Compared to Docker containers:** While Docker provides excellent isolation and reproducibility, it requires substantial technical expertise and infrastructure. Not all researchers have the skills or resources to create and maintain Docker images. In contrast, `renv` works within standard R workflows and requires no additional software beyond R itself.

**Compared to dependency hell:** Without package management, researchers often face “dependency hell” where updating one package breaks another, or where code that worked perfectly suddenly fails after a routine system update. The `renv` approach prevents these issues by isolating projects from system-wide package updates.

**Compared to manual version tracking:** Manually tracking package versions is error-prone and incomplete. Researchers often forget to document dependencies of dependencies, or fail to record the specific CRAN snapshot or repository source. `renv` automatically captures this complete dependency tree.

The specific version constraints in this project (requiring compatible versions of `ggplot2`, `patchwork`, and `bookdown`) highlight an important principle: reproducibility sometimes requires using slightly older package versions rather than always using the latest releases. While the newest versions may offer additional features, they can also introduce breaking changes or incompatibilities. For published research, stability trumps novelty. By explicitly documenting these version choices and their rationale in the `renv.lock` file and project documentation, we make it clear that version selection was deliberate rather than accidental.

### 5.3 Biological Implications

The morphological patterns we observed have important implications for understanding penguin ecology and evolution. The three species in this dataset represent distinct adaptive peaks in morphospace: Adelie penguins with their compact bodies and robust bills, Chinstrap penguins with intermediate morphology, and Gentoo penguins with their larger size and elongated features.

These differences reflect divergent ecological strategies. Adelie penguins, the smallest of the three, are ice-obligate species that forage in sea ice margins and can exploit areas inaccessible to larger species. Their deep, powerful bills are well-suited for crushing krill and small prey. Gentoo penguins, being substantially larger, can dive deeper (regularly exceeding 100 meters) and remain submerged longer, accessing prey resources unavailable to smaller species. Their more elongated bills may be adapted for capturing fish, which require different handling than crustaceans.

The sexual dimorphism we documented is consistent with patterns seen across penguins generally. In most penguin species, males are larger than females, though the magnitude varies. The 15-20% mass difference we observed falls within the typical range for pygoscelid penguins. This dimorphism may serve multiple functions: larger males may be better competitors for nest sites or mates, or size differences may reduce intersexual competition for food by allowing partial niche separation.

The strong predictability of body mass from flipper length (as shown by our regression model) reflects fundamental biomechanical constraints. Penguins are wing-propelled divers, using their flippers to “fly” underwater. The relationship between flipper size and body mass must be tightly constrained to maintain swimming efficiency: too-small flippers cannot propel a heavy body efficiently, while oversized flippers waste energy. This biomechanical coupling explains why flipper length alone explains much of the variance in body mass.

### 5.4 Methodological Considerations

Our analytical approach prioritized simplicity and transparency over sophistication. We used complete-case analysis for missing data rather than imputation, linear regression rather than more complex models, and standard diagnostic plots rather than advanced model validation techniques. This conservative approach serves our pedagogical goal of demonstrating reproducible workflows without overwhelming readers with statistical complexity.

However, more sophisticated analyses could extend these findings. Missing data could be handled through multiple imputation, potentially recovering information from observations with partially missing data. Mixed-effects models could account for hierarchical structure (individuals nested within islands and years). Multivariate approaches could simultaneously analyze relationships among multiple morphological traits. Each of these extensions would provide deeper biological insights while maintaining the reproducibility framework demonstrated here.

The diagnostic plots revealed mild heteroscedasticity in our regression model, with prediction variance increasing slightly at higher fitted values. This pattern is common in biological data and typically not problematic for inference. If prediction accuracy at large body masses were critical, we could address this through weighted least

squares, variance-stabilizing transformations (such as log-transforming body mass), or robust regression methods. The `renv` framework would ensure that whichever approach we chose, it could be exactly reproduced by others.

## 5.5 Limitations

Several limitations warrant consideration when interpreting these results:

- **Missing data were excluded from analyses:** Our complete-case approach means that 11 individuals without sex determination were excluded from sex-difference analyses, and 2 individuals with missing morphological measurements were excluded from the regression model. If missingness is not completely at random (for example, if sex is more difficult to determine in juveniles), this could bias results. However, the small proportion of missing data (< 4%) suggests any bias is likely minimal.
- **Cross-sectional data limits causal inference:** Our data represent a snapshot in time, preventing us from making strong causal claims. For instance, we observe that flipper length predicts body mass, but we cannot determine whether large flippers enable greater body mass accumulation or whether large-bodied penguins grow larger flippers. Longitudinal data tracking individuals over time would be needed to establish causal relationships.
- **Sample sizes vary across species and islands:** Adelie penguins ( $n=152$ ) are better represented than Chinstrap penguins ( $n=68$ ), potentially affecting the precision and reliability of species comparisons. Additionally, not all species occur on all islands, confounding species and geographic effects. More balanced sampling across species and locations would strengthen inferences.
- **Temporal variation is not modeled:** Data were collected across three years (2007-2009), but we did not model inter-annual variation. If environmental conditions varied substantially across years, pooling data may obscure important temporal dynamics. A more complete analysis would include year effects and potentially year-by-species interactions.
- **Measurement error is not quantified:** All morphological measurements involve some measurement error, which we have not explicitly quantified or modeled. If measurement error is substantial relative to true biological variation, it could attenuate relationships and bias parameter estimates toward zero. Measurement error models could account for this, though they require additional data on measurement reliability.

Despite these limitations, our findings align with established knowledge of penguin morphology and ecology, suggesting that the patterns we document are robust and biologically meaningful.

## 5.6 Future Directions

This reproducible analysis framework could be extended in several directions:

**Temporal dynamics:** Incorporating the year variable and potentially seasonal patterns would reveal how morphology varies over time and whether species respond differently to environmental variation.

**Geographic patterns:** Explicitly modeling island effects could reveal fine-scale geographic variation and test hypotheses about local adaptation or phenotypic plasticity.

**Allometric relationships:** Examining how different body parts scale with overall size (allometry) could reveal whether species differ in body proportions or whether they follow common scaling relationships.

**Phylogenetic comparative methods:** Placing these species patterns in a broader phylogenetic context could test whether morphological differences reflect shared ancestry or independent adaptation.

**Integration with ecological data:** Linking morphology to foraging behavior, diet composition, or reproductive success would test functional hypotheses about the adaptive significance of morphological variation.

Each of these extensions could be implemented within the `renv` framework, ensuring that increasingly complex analyses remain reproducible. This demonstrates an important principle: reproducibility is not opposed to analytical sophistication but rather enables it by providing a stable foundation for building increasingly complex analyses.

## 6 Conclusion

This reproducible analysis demonstrates clear morphological differences among Palmer penguin species, with Gentoo penguins notably larger than Adelie and Chinstrap penguins, distinct bill morphologies reflecting dietary specializations, and consistent sexual dimorphism across species. Beyond these biological findings, this work serves as a practical template for implementing reproducible research workflows in R.

The use of `renv` ensures that this analysis can be reproduced exactly by others, regardless of when they attempt to run the code or what versions of packages are currently available on CRAN. By explicitly managing package versions and documenting compatibility requirements (such as the need for `ggplot2` 3.5.0+ to work with `patchwork` 1.2.0+), we protect against the “software rot” that often renders computational analyses non-functional within months of publication.

Reproducibility is not merely a technical nicety but a fundamental requirement for scientific credibility. When analyses cannot be reproduced, findings cannot be verified, errors cannot be detected and corrected, and knowledge cannot accumulate reliably. By adopting tools like `renv` and workflows that prioritize reproducibility from the start, we strengthen the foundation of computational science.

We encourage researchers to adopt similar practices in their own work. The initial investment in learning `renv` (minimal, given its intuitive design) pays substantial dividends in reduced frustration, increased confidence in results, and enhanced ability to share work with collaborators and the broader scientific community. As funding agencies and journals increasingly require code and data sharing, reproducible workflows transition from best practice to basic requirement.

The Palmer Penguins dataset, with its clear patterns and pedagogical value, provides an ideal context for learning these practices. We hope this document serves not only as an analysis of penguin morphology but as a practical guide for researchers seeking to make their own work more reproducible, transparent, and ultimately more valuable to science.

## 7 Reproducibility Information

### 7.1 Session Information

```
# Document the R environment
sessionInfo()

## R version 4.5.0 (2025-04-11)
## Platform: x86_64-pc-linux-gnu
## Running under: Debian GNU/Linux 13 (trixie)
##
## Matrix products: default
## BLAS:    /usr/lib/x86_64-linux-gnublas/libblas.so.3.12.1
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.12.1;  LAPACK version 3.12.0
##
## locale:
## [1] LC_CTYPE=en_US.UTF-8
## [2] LC_NUMERIC=C
## [3] LC_TIME=en_US.UTF-8
## [4] LC_COLLATE=en_US.UTF-8
## [5] LC_MONETARY=en_US.UTF-8
## [6] LC_MESSAGES=en_US.UTF-8
## [7] LC_PAPER=en_US.UTF-8
## [8] LC_NAME=C
## [9] LC_ADDRESS=C
## [10] LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8
```

```

## [12] LC_IDENTIFICATION=C
##
## time zone: Europe/Madrid
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats      graphics   grDevices
## [4] datasets   utils      methods
## [7] base
##
## other attached packages:
## [1] patchwork_1.2.0
## [2] broom_1.0.10
## [3] knitr_1.47
## [4] palmerpenguins_0.1.1
## [5] lubridate_1.9.4
## [6] forcats_1.0.1
## [7] stringr_1.5.2
## [8] dplyr_1.1.4
## [9] purrr_1.1.0
## [10] readr_2.1.5
## [11] tidyverse_1.3.1
## [12] tibble_3.3.0
## [13] tidyverse_2.0.0
## [14] ggplot2_3.5.0
## [15] rmarkdown_2.30
## [16] nvimcom_0.9.76
## [17] colorout_1.3-3
##
## loaded via a namespace (and not attached):
## [1] generics_0.1.4
## [2] renv_1.1.5
## [3] stringi_1.8.7
## [4] lattice_0.22-7
## [5] hms_1.1.4
## [6] digest_0.6.37
## [7] magrittr_2.0.4
## [8] evaluate_1.0.5
## [9] grid_4.5.0
## [10] timechange_0.3.0
## [11] RColorBrewer_1.1-3
## [12] bookdown_0.39
## [13] fastmap_1.2.0
## [14] Matrix_1.7-4
## [15] backports_1.5.0
## [16] tinytex_0.57
## [17] mgcv_1.9-3
## [18] viridisLite_0.4.2
## [19] scales_1.4.0
## [20] cli_3.6.5
## [21] rlang_1.1.6
## [22] splines_4.5.0
## [23] withr_3.0.2
## [24] yaml_2.3.10

```

```

## [25] tools_4.5.0
## [26] tzdb_0.5.0
## [27] vctrs_0.6.5
## [28] R6_2.6.1
## [29] lifecycle_1.0.4
## [30] pkgconfig_2.0.3
## [31] pillar_1.11.1
## [32] gtable_0.3.6
## [33] glue_1.8.0
## [34] xfun_0.53
## [35] tidyselect_1.2.1
## [36] highr_0.11
## [37] farver_2.1.2
## [38] htmltools_0.5.8.1
## [39] nlme_3.1-168
## [40] labeling_0.4.3
## [41] compiler_4.5.0

```

The `sessionInfo()` output provides a complete record of the R version, operating system, locale settings, and all loaded packages with their versions. This information is crucial for reproducibility documentation, as it captures not only the explicitly loaded packages but also their dependencies and the base R configuration.

## 7.2 Package Versions

```

# List all packages and versions used
packages_used <- renv::dependencies()$Package %>% unique()

## Finding R package dependencies ... Done!
installed_versions <- installed.packages()[packages_used, c("Package", "Version")]
installed_version_info <- as.data.frame(installed_versions) %>%
  tibble::rownames_to_column("Index") %>%
  select(-Index)

kable(installed_version_info,
      booktabs = TRUE,
      caption = "R package versions used in this analysis")

```

**Table S1.5: R package versions used in this analysis**

Package	Version
renv	1.1.5
rmarkdown	2.30
bookdown	0.39
knitr	1.47
broom	1.0.10
ggplot2	3.5.0
palmerpenguins	0.1.1
patchwork	1.2.0
tidyverse	2.0.0
tibble	3.3.0

The table above lists all R packages used in this analysis along with their exact versions. This information, com-

bined with the `renv.lock` file included with this project, provides complete documentation of the computational environment. Anyone wishing to reproduce this analysis should use `renv::restore()` to install these exact versions.

### 7.3 System Information

This analysis was conducted using R version 4.5.0 (2025-04-11) on `x86_64-pc-linux-gnu`. The specific R version and operating system should not substantially affect results for this analysis, but are documented here for completeness. The `renv` system primarily manages package versions rather than R or system versions, though significant R version differences (e.g., R 3.x vs. R 4.x) could potentially affect behavior.

### 7.4 Data Availability

The Palmer Penguins dataset is publicly available through the `palmerpenguins` R package and does not require any special access permissions. Original data were collected by Dr. Kristen Gorman and colleagues at Palmer Station, Antarctica, as part of the Palmer Station Long Term Ecological Research (LTER) Program. We gratefully acknowledge their efforts in data collection and public sharing.

### 7.5 Code Availability

All code for this analysis is contained in this R Markdown document. The complete project, including the `renv.lock` file, can be shared via version control systems (e.g., GitHub) or as a compressed archive. To reproduce the analysis:

1. Ensure R (version 4.0 or higher recommended) is installed
2. Open the project directory in R or RStudio
3. Run `renv::restore()` to install the correct package versions
4. Knit this R Markdown document to produce the PDF output

The entire process, from environment setup to final PDF generation, should complete in less than 10 minutes on a typical desktop computer, demonstrating the practical feasibility of fully reproducible computational research.