

The Digital Unveiling: Boxplots

In Section 4.2, you did the heavy lifting. You manually stacked dots, physically chopped the data into quarters with a “cheese cleaver,” and drew physical brackets to represent the Range and the IQR. Today, we are going to look at how R automates the exact structures you already built.

Sally the Statypus says: Hey there, you did so great getting through all of that information in the last worksheet. That was a lot! Bill and I were a bit sneaky and actually have taught you the next two sections of the book already!

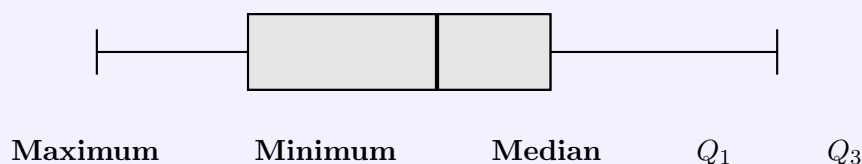
Bill the Statypus says: That’s right, Sally. We tricked them into manually drawing boxplots and calculating z -scores. Let’s take a look.

1. The Boxplot Unveiled

Open your Section 4.2 worksheet to **Page 5**. Look at the dot plot where you drew the large “Range” bracket underneath and the floating “IQR” bracket in the middle. You essentially drew a **Boxplot**.

Reflection: Labeling the Architecture

A standard boxplot is literally just a visual representation of the **Five-Number Summary** you calculated yesterday. Based on the R output on your screen, draw lines connecting the vocabulary terms below to the correct part of the abstract boxplot provided:



2. The 1.5 IQR Fences Unveiled

Bill the Statypus says: If we caught a mutant 5.0kg platypus, we wouldn’t want the “whiskers” of our boxplot to stretch all the way out to 5.0kg, because that would ruin the visual scale of where the *normal* platypuses live.

To prevent this, statisticians use the $1.5 \times IQR$ **Rule**. R automatically builds invisible “fences” that are 1.5 box-lengths away from the edges. Any data point that falls outside those invisible fences is plotted as an individual dot. We call these dots **Outliers**.

- Check the Book:** If you want to see how to build those fences by hand or make basic boxplots, read **Section 4.4** in your textbook. But for our analysis today, we will let R do the math.

3. The Sister Script: Logical Indexing & Filtering

Real data analysis is usually done using **Scripts**—pre-written files of code that act as a blueprint. Today, we are opening our first “Sister Script.”

Coding Corner: Running SpiceOfLife.r

Open `SpiceOfLife.r` from `academy.statypus.org`. This script runs a two-pass audit on the data:

1. **The Setup:** It fetches the environment from the server and creates a safe working copy dataset called `df`.
2. **Pass 1 (The Raw Data):** It splits `df` into Males and Females using logical brackets `[,]` and plots them vertically.
3. **The Filter:** It uses the `!=` (NOT equal to) operator to strip out Juveniles, creating a refined `df_adults` dataset.
4. **Pass 2 (The Refined Data):** It plots the Adults Only data directly underneath Pass 1.

Run the script line by line to generate your dual-visual!

Reflection: Synthesis: Read the Visuals

Based **only** on the stacked boxplots on your screen:

1. Look at **Pass 1**. Do the middle 50% boxes for Males and Females overlap? What does this imply about their weight differences?
2. Look at **Pass 2**. What happened to the “Whiskers” and the “Outlier Dots” when we removed the Juveniles?
3. Why is **Pass 2** a more honest representation of biological differences?