**What this template does**

* If you provide an input CSV (relative path recommended), it will read it. If no file is given, it generates *synthetic example data* so the script can be tested immediately.
* Uses relative output path by default (plots/GO\_Terms\_Log2CombinedScore.jpeg) and creates the directory if needed.
* Produces a horizontal bar chart (flipped coordinates) where the fill reflects log2(Combined.Score) using a red gradient.

**Expected input CSV format**

* File must be comma-separated (CSV) and contain at least two columns (case-sensitive):
  + Term — the label for each ontology term (string).
  + Combined.Score — a positive numeric score for each term.
* Example (safe, synthetic):
* Term,Combined.Score
* "Term A",1234.56
* "Term B", 987.12
* "Term C", 432.00

**How to run**

1. From shell using Rscript (recommended for GitHub Actions / CI):
2. # using default sample input (if you don't pass anything)
3. Rscript plot\_go\_template.R
4. # using your own CSV and custom output location (relative paths recommended)
5. Rscript plot\_go\_template.R data/go\_enrichment.csv plots/my\_go\_plot.jpeg
6. From inside an interactive R session:
7. source("plot\_go\_template.R") # will run with defaults or synthetic data

(If you want to call the script from R interactively with arguments, use commandArgs() approach or run via system().)

**Key code sections**

* **Package check & install**: The script checks for ggplot2, attempts to install it if missing. This keeps the repository runnable on new machines.
* **Argument parsing**: commandArgs(trailingOnly = TRUE) reads optional positional args: [input\_csv] [output\_image]. Defaults are data/go\_enrichment.csv and plots/GO\_Terms\_Log2CombinedScore.jpeg.
* **Reading data**: If input\_csv exists it is read; else the script creates a synthetic example dataset.
* **Validation**:
  + Ensures Combined.Score is numeric and finite.
  + Replaces non-positive scores with a small positive value so log2() is defined (logs of non-positive numbers are invalid).
* **Transform**: log2CombinedScore <- log2(Combined.Score) — this is what you plot on the y-axis (or x-axis after flipping).
* **Plot**:
  + geom\_col() (identity bars) with reorder(Term, log2CombinedScore) to sort bars by value.
  + scale\_fill\_gradient(low="lightpink", high="red") gives the red gradient (changeable).
  + coord\_flip() makes bars horizontal (usually better for long term names).
* **Saving**:
  + ggsave() automatically chooses format by file extension and is given width (12 in), height (8 in), and dpi (150). These can be changed as needed.

**Customizations you might want**

* **Top N terms**: Uncomment the top\_n block to show only the top N terms by score.
* **Colors**: Replace scale\_fill\_gradient() with any other gradient or discrete palette.
* **Output format**: Save PNG, PDF, or JPEG by changing output filename extension:
  + "plots/GO\_Terms\_Log2CombinedScore.png"
  + "plots/GO\_Terms\_Log2CombinedScore.pdf"
* **Plot styling**: Tweak theme\_\*() or base\_size to match your lab/figure style.

**Troubleshooting**

* **"Input CSV must contain columns..."** — check header names match exactly Term and Combined.Score.
* **Non-numeric Combined.Score** — remove thousands separators or quotes; ensure the column contains numbers.
* **ggplot2 missing** — the script attempts to install, but if installation is blocked (no internet or permissions), install manually in R: install.packages("ggplot2").
* **Path issues on GitHub Actions** — use relative paths and ensure data/ file is checked into repo (or created at workflow runtime).