ECON 413 Setup & Install How–To (with detailed Windows Rtools step)

Clone the GitHub project, install build tools, and install `gEcon` reliably.

What you will do (in order)

1. Install **R** and **RStudio**.
2. Install **Git** and OS build tools (**Rtools** on Windows).
3. Create a **GitHub Personal Access Token (PAT)** and store it for RStudio.
4. Get the assignment into RStudio (clone via Git or Download ZIP).
5. Install **gEcon** from **R-Forge** and verify.

A. Install the fundamentals

A1. R and RStudio

- Install **R** (latest release) and **RStudio Desktop**.
- Check your R version inside RStudio: `R.version.string`.

A2. Git

- Windows: install "Git for Windows" (includes Git Credential Manager).
- macOS: Terminal → `xcode-select -install` (provides `git`) or install via Homebrew.
- Linux: use your package manager, e.g. `sudo apt install git`.
- Verify: `git -version` prints a version (✔).

A3. Build tools for compiling R packages

Windows users: follow the step-by-step guide below (Rtools). macOS/Linux users can skip to Section D.

Windows: Rtools (step-by-step with URLs)

1) Find your R version

```
R.version.string
# Example output: "R version 4.4.1 (2025-08-10)"
```

2) Download the matching Rtools installer

- Go to the official Rtools page (index with version mapping): https://cran.r-project.org/bin/windows/Rtools/
- Choose the installer that matches your R version:

- R 4.5.x → **Rtools 4.5**
- R 4.4.x → **Rtools 4.4**
- R 4.3.x → **Rtools 4.3**
- R 4.2.x → **Rtools 4.2**
- R 4.0–4.1.x → **Rtools 4.0**

- Direct page for Rtools 4.4 (example): `https://cran.r-project.org/bin/windows/Rtools/rtools44/rtools.html`

3) Run the installer

- Close R/RStudio first. Double-click the downloaded `rtoolsXX-*.exe`.
- Accept defaults. Note the install folder (usually `C:\rtools44` or similar).

4) Restart RStudio and verify the toolchain

Listing 1: Check that build tools are available

```
# Install helper if needed
if (!requireNamespace("pkgbuild", quietly = TRUE)) install.packages("pkgbuild")
pkgbuild::has_build_tools(debug = TRUE)  # should return TRUE
```

Important: On **R 4.2+**, R automatically finds the correct **Rtools**—you *do not* need to edit your PATH. (*If you are on R 4.0–4.1, see the legacy PATH note below.*)

Legacy PATH note (only R 4.0–4.1 with Rtools40). If you use R 4.0 or 4.1, add Rtools40 to PATH via `.Renviron`:

```
# Create/open your ~/.Renviron file, then add this line and save:
# (This is for Rtools40 only; not needed for R >= 4.2.)
# PATH="${RTOOLS40_HOME}\usr\bin;${PATH}"
```

Then restart RStudio and run `pkgbuild::has_build_tools(debug=TRUE)` again. See Rtools40 notes here: `https://cran.r-project.org/bin/windows/Rtools/rtools40.html`

Quick sanity checks.

```
# Should show a path to 'make' (Windows)
Sys.which("make")
# Try a tiny from-source build (forces toolchain use if no binary available)
install.packages("PKI", type = "source")
```

B. GitHub Personal Access Token (PAT)

Create & store your PAT

1. In GitHub: Settings → Developer settings → Personal access tokens.
2. Create a token (*classic:* scope `repo`; *fine-grained:* repo read/write for your Classroom org).
3. In RStudio, store it using:

   ```
   if (!requireNamespace("gitcreds", quietly = TRUE)) install.packages("gitcreds")
   gitcreds::gitcreds_set()    # paste token at the prompt
   ```

C. Get the assignment into RStudio

Option 1 (preferred): Clone via RStudio

1. File → New Project → Version Control → Git.
2. **Repository URL:** paste the *HTTPS clone URL* ending with `.git` (example shape: `https://github.com/<org>/<repo>.git`).
3. Choose a local folder → Create Project.

Option 2 (fallback): Download ZIP

Green **Code** button → Download ZIP → unzip → double-click the `.Rproj` file.

Common URL & PAT pitfalls

- ✘ “Repository not found”: you pasted a page URL, not the **clone URL**. Use the one ending with `.git`.
- ✘ Repeated password prompts: run `gitcreds::gitcreds_set()` and paste your PAT.

D. Install gEcon from R-Forge (NOT on CRAN) & verify

What this means. **Do not** use `install.packages("gEcon")` against CRAN. Instead, **download the package file from the official R-Forge page** and install from that file. *This is the supported method.*

D1. Prerequisites (install these first)

- **R version:** gEcon 1.2.3 is compatible with R 4.4.3 and R 4.5.0.[1]
- **R packages (from CRAN):** `Matrix`, `MASS`, `nleqslv`, `Rcpp`, and base `methods`.[2]

Install the dependencies now (copy into the R Console):

```
install.packages(c("Matrix","MASS","nleqslv","Rcpp"))
```

D2. Download the correct gEcon file

Go to the official gEcon download page: `https://gecon.r-forge.r-project.org/download.html`

- **Windows:** Download the `.zip` that matches your R version (e.g., `gEcon_1.2.3.zip` for R 4.5.0 or R 4.4.3).
- **macOS / Linux:** Download the `.tar.gz` source file (e.g., `gEcon_1.2.3.tar.gz`). You need build tools (macOS: Command Line Tools; Linux: build-essential).

[1] See the gEcon Download page “gEcon 1.2.3 is fully compatible with R-4.4.3 and R-4.5.0.” `https://gecon.r-forge.r-project.org/download.html`

[2] Listed under “Prerequisites” on the gEcon Documentation & FAQ page. `https://gecon.r-forge.r-project.org/doc.html`

D3. Install from the downloaded file (two simple ways)

Option A — RStudio GUI (easiest).

1. RStudio → **Tools** → **Install Packages**.
2. **Install from:** choose **Package Archive File (`*.zip` / `*.tar.gz`)**.
3. Click **Browse**, select the file you just downloaded (`.zip` on Windows, `.tar.gz` on macOS/Linux).
4. Check **Install dependencies**. Click **Install**.

Option B — R Console (explicit). *Replace the path with your actual download location.*

```
# Windows (binary .zip):
install.packages("C:/Users/<you>/Downloads/gEcon_1.2.3.zip",
                 repos = NULL, type = "win.binary")

# macOS / Linux (source .tar.gz):
install.packages("~/Downloads/gEcon_1.2.3.tar.gz",
                 repos = NULL, type = "source")
```

D4. Verify the install (quick smoke test)

```
library(gEcon)
ex <- system.file("examples", package = "gEcon")
stopifnot(file.exists(file.path(ex, "rbc.gcn")))
file.copy(file.path(ex, "rbc.gcn"), getwd(), overwrite = TRUE)
m <- make_model("rbc.gcn")   # should succeed if install is OK
cat("gEcon OK: parsed the example RBC model.\n")
```

D5. If you hit errors, use this checklist

Symptom	Fix
"there is no package called 'gEcon' "	You installed the wrong thing (e.g., tried `library("gEcon_1.2.3.zip")`). You must **install** from the file first (GUI or `install.packages(..., repos=NULL)`), then run `library(gEcon)`.
Windows: source install fails / toolchain error	Install **Rtools** matching your R version, then retry. (See our Windows Rtools section.)
macOS: compilation error / toolchain missing	Run `xcode-select -install` (Command Line Tools), then re-run the `.tar.gz` install.
"object 'make_model' not found"	The package didn't load. Run `library(gEcon)` again; if it fails, reinstall from the correct file for your OS / R version.
GUI install didn't fetch dependencies	Pre-install the listed deps (`Matrix, MASS, nleqslv, Rcpp`) from CRAN, then redo the file install.

Note. The *gEcon* docs explicitly endorse installing from the downloaded file (either via command line `R CMD INSTALL` or directly from the R GUI). If you prefer a one-liner and R-Forge provides a current build for your platform, `install.packages("gEcon", repos="https://R-Forge.R-project.org")` can work, but the file-based method above is the official path.

E. Troubleshooting checklist

Symptom	**Fix**
“`gEcon is not a package`”	Install via `install.packages("gEcon")` after adding R-Forge; then `library(gEcon)`. Do *not* call `library("gEcon_*.tar.gz")`.
Windows: “Rtools not found”	Install matching Rtools version (see mapping & URLs above), restart RStudio, run `pkgbuild::has_build_tools(debug=TRUE)`.
R 4.0–4.1 on Windows	Use Rtools40 and add PATH in `.Renviron` as shown; restart RStudio.
“Repository not found” when cloning	Use the **HTTPS clone URL** ending with `.git` and ensure you are logged into the invited GitHub account.

Reference pages: Rtools index and version mapping: `https://cran.r-project.org/bin/windows/Rtools/`. R 4.4 Windows build notes (auto-detection of Rtools, no PATH edits on R 4.2+): `https://cran.r-project.org/bin/windows/base/howto-R-4.4.html`.