EJAM 2.32.0

Installing the EJAM R package

Source: vignettes/1_installing.Rmd

EJAM is not only a web app, it is also an R package.

To use the R package, you first need to have R and RStudio.

The EJAM package can be installed on your computer as an R package. This is useful if you want to use the full set of EJAM R functions outside of the web app interface, to do customized analysis, to explore the data in R, or to reuse data or code in other applications.

How to install

Try this first (if you just want to use the package, not clone & edit source code)

If you just install the package without cloning the repository, you can use the EJAM package, but cannot edit the source code, for example. To just install it, try this:

```
options(timeout = 300) # Just in case the download takes lor
if (!require(remotes)) {install.packages("remotes")}

remotes::install_github("USEPA/EJAM-open", dependencies = TR
```

If install_github does not work, you can try something like this:

```
x <- "https://github.com/USEPA/EJAM-open/archive/refs/tags/v
remotes::install_url(url = x, dependencies = TRUE, auth_toke
```

load/attach the package

To use EJAM in the RStudio console, start by loading / attaching the package:

EJAM 2.32.0

That should be all you need to do.

Cloning: if you need the full source code or want to build/install in RStudio on your own

You can use the package just by installing without cloning. However, if you want to explore and edit source code, etc., you may want to clone the github repository that has the EJAM source code. Note that to do so, you have to know how to use R/RStudio with git and github, create a project, build from source, etc.

The EJAM package is available at the USEPA/EJAM-open github repository (not on CRAN).

Options for getting the full source package:

a. One way to get the full source package is to use a browser to go to the repository page, USEPA/EJAM-open github repo and then click the green "Code" button, and download and unzip the zip file that contains the package.

b. Another way to get the source package is that in RStudio you can click New Project, Version Control, Git, and enter the repository URL.

c. A third way is to Clone or Fork the package via GitHub Desktop.

Regardless of how you got the full source code, you would then need to build/install the package from source on your computer using RStudio (using the Build menu) or using the devtools package, for example.

Technical details

Just in case you need more details on how installing and attaching the package works, the following describes the package dependencies and code used by the package to get the data and build an index.

to handle package dependencies. EJAM needs a number of other packages to be installed that are (almost all) available from CRAN. Installing the EJAM package as explained above will handle obtaining those other packages. Cloning and building/installing and then trying to load/attach EJAM will also alert you to those other packages you need to install if you don't already have them. In case it is of interest, the list of CRAN packages needed is in the `DESCRIPTION` file in the R package source code root folder (as can be found in the code repository). Note some are in Suggests and you might want to install those as well – using dependencies=T in `remotes::install_github()` or `remotes::install_url()` will make sure of that.

Details on the automatic data downloads

To work in the RStudio console, EJAM needs some datasets not stored as part of the package. However, they already should be downloaded and loaded into memory automatically as soon as you use `require(EJAM)` .

On first use, it should automatically download some data files. Each time it is attached, it will check for updates and also will build a spatial index of Census block points.

Typically you would not need to download any datasets yourself, because EJAM just downloads these when the app starts (technically, when the R package is attached) (or only as needed in the case of certain datasets that are not always needed). Some datasets are installed along with the package, such as the `blockgroupstats` data. But large files like `blockpoints` are stored on a "pins board" server and EJAM downloads them from there. You might want your own local copies, though, for these reasons:

Attaching the package actually checks for copies in memory first (e.g., `exists("quaddata", envir = globalenv())`), then local disk (using `dataload_from_local()` looking in the data folder of the (source or installed) package, as defined by `EJAM:::app_sys()` which is just a wrapper for `system.file()`), then finally tries to download any still needed, using internal functions.

Details on the indexing of blocks

EJAM 2.32.0

EJAM does this automatically when attached via `library()` or `require()` e.g., require(EJAM), by creating an object called localtree based on the quaddata object obtained as mentioned above. Some functions check for it and try to build the index on the fly if it is missing. You can also (re)build it manually:

```
indexblocks()
```

On this page

- How to install
  - Try this first (if you just want to use the package, not clone & edit source code)
- load/attach the package
- Cloning: if you need the full source code or want to build/install in RStudio on your own
- Technical details