

# Package ‘automagic’

July 22, 2025

**Type** Package

**Title** Automagically Document and Install Packages Necessary to Run R Code

**Version** 0.5.1

**Description** Parse R code in a given directory for R packages and attempt to install them from CRAN or GitHub. Optionally use a dependencies file for tighter control over which package versions to install.

**License** GPL

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Depends** R (>= 3.1.0)

**Imports** dplyr, formatR, knitr, magrittr, purrr, remotes, yaml

**URL** <https://github.com/cole-brokamp/automagic>

**BugReports** <https://github.com/cole-brokamp/automagic/issues>

**Suggests** testthat

**NeedsCompilation** no

**Author** Cole Brokamp [aut, cre],  
Steph Locke [ctb]

**Maintainer** Cole Brokamp <cole.brokamp@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-03-05 21:30:22 UTC

## Contents

automagic . . . . .	2
get_dependent_packages . . . . .	2
get_package_details . . . . .	3
install_deps_file . . . . .	3

install_package_guess . . . . .	4
make_deps_file . . . . .	4
parse_packages . . . . .	5
%>% . . . . .	6

<b>Index</b>	<b>7</b>
--------------	----------

---

automagic	<i>Automagically install all required R packages</i>
-----------	--

---

**Description**

Searches a given directory for all R and R Markdown files, parses them for required packages and attempts to install them from CRAN. More importantly, if a ‘deps.yaml’ file was made using [make\\_deps\\_file](#), automagic will use this rather than try to install based on a best guess.

**Usage**

```
automagic(directory = getwd())
```

**Arguments**

directory      folder to search for R and Rmd files

**See Also**

[install\\_package\\_guess](#), [parse\\_packages](#)

---

get_dependent_packages	<i>get packages required to run R code</i>
------------------------	--

---

**Description**

get packages required to run R code

**Usage**

```
get_dependent_packages(directory = getwd())
```

**Arguments**

directory      folder to search for R and Rmd files

**Details**

parses all R and Rmd files in a directory and uses [parse\\_packages](#) to find all R packages required for the code to run

**Value**

a vector of package names

---

get_package_details	<i>get package details</i>
---------------------	----------------------------

---

**Description**

Uses packageDescription to get details about given package from R library on local machine. Currently only supports CRAN and GitHub packages

**Usage**

```
get_package_details(pkg_name)
```

**Arguments**

pkg_name	package name
----------	--------------

**Value**

A list of package characteristics. "Package", "Repository", and "Version" for CRAN packages. "Package", "GithubUsername", "GithubRepo", "GithubRef", and "GithubSHA1" for Github packages.

---

install_deps_file	<i>Install R packages from a package dependencies (deps.yaml) file</i>
-------------------	--

---

**Description**

Installs packages from GitHub and CRAN based on Sha1 key and version number respectively, as defined in a deps.yaml file created by [make\\_deps\\_file](#)

**Usage**

```
install_deps_file(directory = getwd())
```

**Arguments**

directory	directory containing deps.yaml file
-----------	-------------------------------------

**See Also**

[make\\_deps\\_file](#), [automagic](#)

---

`install_package_guess` *Install latest version of package from CRAN*

---

### Description

If a package is not available in the R library, attempt to install it from CRAN. Unlike previous versions of `automagic`, if the package is not available on CRAN, the function will return an error (instead of trying to install from GitHub). If R is running interactively, then the user will be prompted before installing.

### Usage

```
install_package_guess(pkg, force_install = FALSE,
  prompt = interactive())
```

### Arguments

<code>pkg</code>	a character vector with the names of packages to install from CRAN
<code>force_install</code>	install even if package is in library (warning! this could install a newer or older version of an already installed package)
<code>prompt</code>	prompt the user to install a package (defaults to yes if the R session is interactive)

### Details

@details This function does not check package versions. Specify `force_install=TRUE` to force installation of the package, updating it to the latest available version. Note that this function attempts to install its packages based on a best guess and is meant for use in automatically setting up an R programming environment. Do not use for installing packages if you have the option to install from a `deps.yaml` file. See [make\\_deps\\_file](#) and [install\\_deps\\_file](#) for installing version specific packages based on a local R library.

---

`make_deps_file` *Make a package dependencies (deps.yaml) file*

---

### Description

This function parses R code for required packages using [parse\\_packages](#) and then queries the R package library to determine the exact source and version of each package to install. Currently, only CRAN and GitHub packages are supported. Install packages from the ‘`deps.yaml`’ file using `automagic{install_deps_file}`

### Usage

```
make_deps_file(directory = getwd())
```

**Arguments**

directory      directory containing R code to parse

**See Also**

[automagic](#)

---

parse_packages	<i>Parse R code for required packages</i>
----------------	---

---

**Description**

Parses an R or R Markdown file for the package names that would be required to run the code.

**Usage**

```
parse_packages(f1)
```

**Arguments**

f1      file to parse for required package names

**Details**

This function uses regular expressions to search through a file containing R code to find required package names. It extracts not only package names denoted by [library](#) and [require](#), but also packages not attached to the global namespace, but are still called with `::` or `:::`.

Because it relies on regular expressions, it assumes all packages adhere to the valid CRAN package name rules (contain only ASCII letters, numbers, and dot; have at least two characters and start with a letter and not end it a dot). Code is also tidying internally, making the code more predictable and easier to parse (removes comments, adds whitespace around operators, etc). R Markdown files are also supported by extracting only R code using [purl](#).

**Value**

a vector of package names as character strings

**See Also**

[install\\_package\\_guess](#), [automagic](#)

**Examples**

```
## Not run:
cat('library(ggplot2)\n # library(curl)\n require(leaflet)\n CB::date_print()\n',file='temp.R')
parse_packages('temp.R')
unlink('temp.R')

## End(Not run)
```

---

%>%	<i>Pipe imported from magrittr</i>
-----	------------------------------------

---

**Description**

Pipe imported from magrittr

# Index

::, [5](#)  
:::, [5](#)  
%>%, [6](#)

automagic, [2](#), [3–5](#)

get\_dependent\_packages, [2](#)  
get\_package\_details, [3](#)

install\_deps\_file, [3](#), [4](#)  
install\_package\_guess, [2](#), [4](#), [5](#)

library, [5](#)

make\_deps\_file, [2–4](#), [4](#)

parse\_packages, [2](#), [4](#), [5](#)  
purl, [5](#)

require, [5](#)