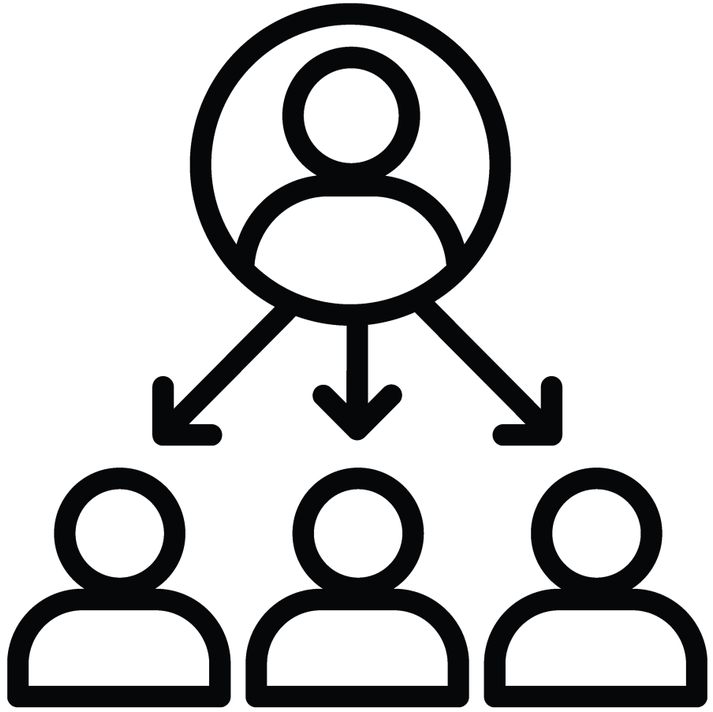
DEVELOPING R PACKAGES

Introduction to Package Building

Aimée Gott and Nic Crane
Data Science Consultants, Mango Solutions



The Structure of an R Package

An R package can include:

- Functions
- Data
- Documentation
- Vignettes
- Tests

Mandatory Components

As a minimum your package must include:

- R directory
- man directory
- NAMESPACE file
- DESCRIPTION file

Packages for easily creating packages!

- `devtools`
- `roxygen2`

Setting Up an R Package

Core devtools functions:

- `create()`
- `document()`
- `check()`
- `build()`
- `test()`

create()

```
library(devtools)
create("simutils")
```

Note: Avoid names already taken on CRAN.

DEVELOPING R PACKAGES

Let's practice!

DEVELOPING R PACKAGES

Description and Namespace files

Aimée Gott
Education Practice Lead, Mango Solutions

DESCRIPTION file

- General package information
- Author and maintainer details
- Package dependencies
- License

Example DESCRIPTION file

```
Package: simutils
Title: Simulation Analysis Tools
Version: 1.0.0.0
Authors@R: c(
    person("Nic", "Crane", email = "ncrane@mango-solutions.com", role = c("aut", "cre")),
    person("Aimee", "Gott", email = "agott@mango-solutions.com", role = c("aut", "ctb")))
Description: A series of tools for simulation analysis used for learning about
    distributions.
Depends:
    R (>= 3.4.2)
License: GPL-2
LazyData: true
RoxygenNote: 6.0.1
Imports: dplyr,
    purrr
Suggests: testthat,
    knitr,
    rmarkdown
VignetteBuilder: knitr
```

Example DESCRIPTION file

```
Package: simutils
Title: Simulation Analysis Tools
Version: 1.0.0.0
Authors@R: c(
    person("Nic", "Crane", email = "ncrane@mango-solutions.com", role = c("aut", "cre")),
    person("Aimee", "Gott", email = "agott@mango-solutions.com", role = c("aut", "ctb")))
Description: A series of tools for simulation analysis used for learning about
    distributions.
Depends:
    R (>= 3.4.2)
License: GPL-2
LazyData: true
RoxygenNote: 6.0.1
Imports: dplyr,
    purrr
Suggests: testthat,
    knitr,
    rmarkdown
VignetteBuilder: knitr
```

Example DESCRIPTION file

```
Package: simutils
Title: Simulation Analysis Tools
Version: 1.0.0.0
Authors@R: c(
    person("Nic", "Crane", email = "ncrane@mango-solutions.com", role = c("aut", "cre")),
    person("Aimee", "Gott", email = "agott@mango-solutions.com", role = c("aut", "ctb")))
Description: A series of tools for simulation analysis used for learning about
    distributions.
Depends:
    R (>= 3.4.2)
License: GPL-2
LazyData: true
RoxygenNote: 6.0.1
Imports: dplyr,
    purrr
Suggests: testthat,
    knitr,
    rmarkdown
VignetteBuilder: knitr
```

Example DESCRIPTION file

```
Package: simutils
Title: Simulation Analysis Tools
Version: 1.0.0.0
Authors@R: c(
    person("Nic", "Crane", email = "ncrane@mango-solutions.com", role = c("aut", "cre")),
    person("Aimee", "Gott", email = "agott@mango-solutions.com", role = c("aut", "ctb")))
Description: A series of tools for simulation analysis used for learning about
    distributions.
Depends:
    R (>= 3.4.2)
License: GPL-2
LazyData: true
RoxygenNote: 6.0.1
Imports: dplyr,
    purrr
Suggests: testthat,
    knitr,
    rmarkdown
VignetteBuilder: knitr
```

NAMESPACE file

The NAMESPACE file controls:

- Functions or packages imported for use by our package
- Functions exported by our package

We do NOT edit by hand!

Example NAMESPACE file

```
# Generated by roxygen2: do not edit by hand

export(na_counter)
export(sample_from_data)
import(dplyr)
import(purrr)
```

DEVELOPING R PACKAGES

Let's practice!

DEVELOPING R PACKAGES

Optional Directories

Nic Crane
Data Science Consultant, Mango Solutions

Optional Directories

We can also include:

- Data
- Vignettes
- Tests
- Compiled code
- Translations
- Demos

Data

```
sim_dat <- data.frame(
    ID = 1:10,
    Value = sample(1:11, 10),
    Apples = sample(c(TRUE, FALSE), 10, replace = TRUE)
)
```

```
use_data(sim_dat, pkg = "simutils")
```

Vignettes

```
use_vignette("my_first_vignette", pkg = "simutils")
```

Best Practice for Structuring Code

Guidelines for the R directory:

- No subdirectories
- Don't have everything in one script
- Don't have a large number of small files
- Group similar functions together

DEVELOPING R PACKAGES

Let's practice!