Reproducible Data Processing in Team Workflows with DataPackageR
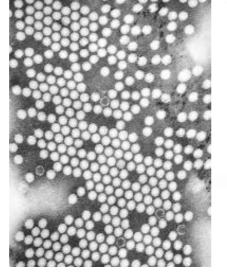
Bryan Mayer
06-08-2019

Who am I?
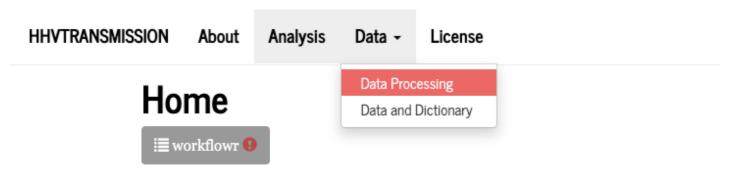
What happens when we ask for data?

The good
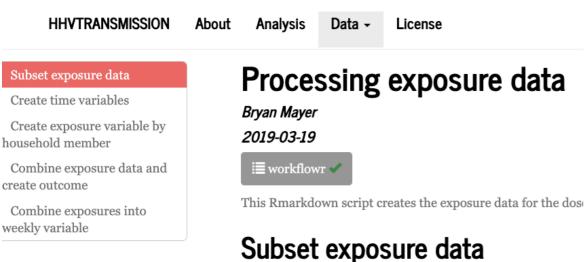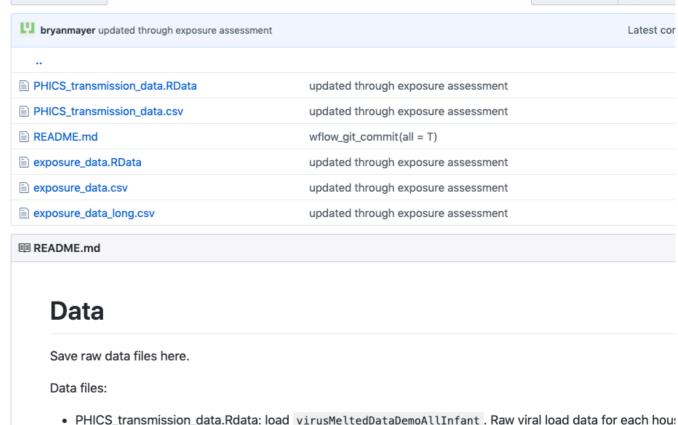
An example of a workflowr project

The bad

Notes on data processing

TBD

Uh oh…

Reproducible data practices are critical in the scientific process

- **Prepping for analysis:** Has the upstream data processing been performed correctly?

- **Reproducible research:** Verify, replicate and validate past analyses

- **Future and open research:** Pass the data on

© Sidney Harris
sciencecartoonsplus.com

"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

Problem: data may be poorly documented, inconsistently versioned, and hard to transfer

Solution: build versioned, well-documented, easily transferred dataset(s) with DataPackageR

Check for updates

SOFTWARE TOOL ARTICLE

REVISED **DataPackageR: Reproducible data preprocessing, standardization and sharing using R/Bioconductor for collaborative data analysis** [version 2; peer review: 2 approved, 1 approved with reservations]

A practical approach to wrangling multi-assay data sets for standardized and reproducible data analysis

Greg Finak[1-3], Bryan Mayer[1-3], William Fulp[1-3], Paul Obrecht[1-3], Alicia Sato[1-3], Eva Chung[1-3], Drienna Holman[1-3], Raphael Gottardo[1-3]

[1]Vaccine and Infectious Disease Division, Fred Hutchinson Cancer Research Center, Seattle, WA, 98109, USA
[2]Vaccine Immunology Statistical Center, Fred Hutchinson Cancer Research Center, Seattle, WA, 98109, USA
[3]Statistical Center For HIV AIDS Research and Prevention, Fred Hutchinson Cancer Research Center, Seattle, WA, 98109, USA

✓ DataPackageR is a useful tool for well defined research projects using

- Team or solo workflows utilizing git version control

- Single or multiple datasets with constrained sizes

- Well described data formats

✦ DataPackageR may be limited for

- Large datasets (hinders features and transferability)

- Exploratory data undergoing constant manipulation

– There are probably better solutions for

- **Handling distributed data or large data problems**

- **Real-time applications**

Dataset consumers: Any R user!

```r
## Install the package ##

devtools::install_github("bryanmayer/myDataPackage")
#devtools::install_local("/local_location/myDataPackage")

## Load the package and data into environment ##

library(myDataPackage)
data("dataset1")
data('dataset2')

## OR load the data without loading the package ##

dataset1 = myDataPackage::dataset1
dataset2 = myDataPackage::dataset2
```

Documentation available with the package

Pantaleo EV07 NAb Processing

Pantaleo EV07 NAb Processing

This report documents the Pantaleo EV07 NAb data processing.

Preliminaries

We first source our external functions and global variables, load libraries, and set options.

```
# Libraries
suppressPackageStartupMessages({
  library(DataPackageR)
  library(knitr)
  library(data.table)
  library(xtable)
  library(readxl)
  library(ggplot2)
  library(kableExtra)
  library(dplyr)
})


# Functions
sys.source(file.path(DataPackageR::project_path(), "data-raw", "functions.R"), envir=topenv())
sys.source(file.path(DataPackageR::project_path(), "data-raw", "ac_functions.R"), envir=topenv())
```

Vignettes created to document processing code

```
vignette("Data-Processing")
```

Help files are generated for each dataset to be accessed as data dictionaries

```
?processedData
```

bama {serumbama}   R Documentation

Serum BAMA assay data from VTN protocols 073, 076, 078, 082, 083, 085, 086, 087, 088, 090, 094, 096, 205, and 505

Description

BAMA assay data from VTN protocols 073, 076, 078, 082, 083, 085, 086, 087, 088, 090, 094, 096, 205, and 505

Format

a data.frame containing standard assay fields and analysis fields

protocol

    Protocol

arm

    Treatment arm from protocol specific rx file

grp

Datasets are versioned

```
devtools::session_info()
```
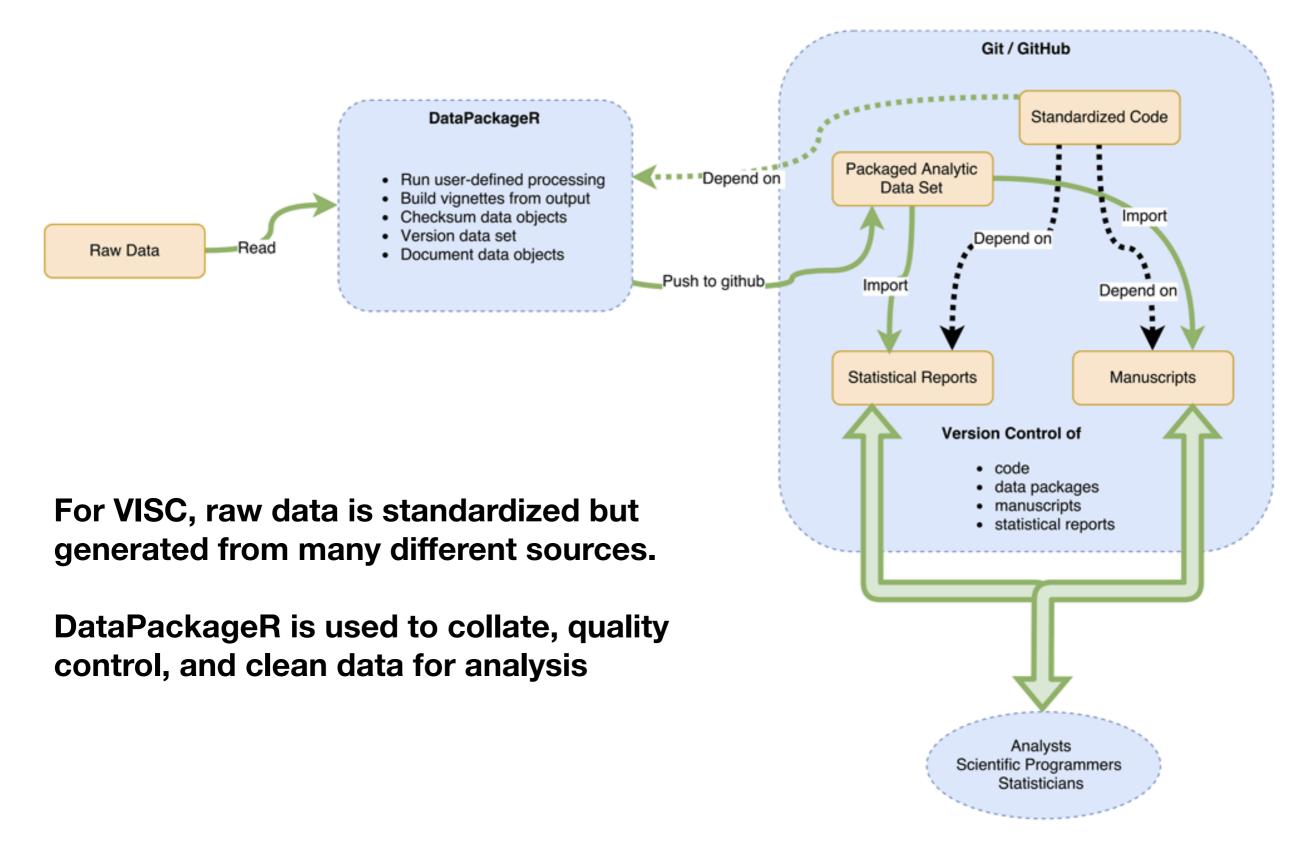
```
## — Session info ————————————————————————————————————
##   setting  value
##   version  R version 3.6.0 (2019-04-26)
##   os       macOS Mojave 10.14.5
##   system   x86_64, darwin15.6.0
##   ui       X11
##   language (EN)
##   collate  en_US.UTF-8
##   ctype    en_US.UTF-8
##   tz       America/Los_Angeles
##   date     2019-05-30
##
## — Packages ————————————————————————————————————————
##   package        * version date       lib source
##   myDataPackage    1.0     2019-05-31 [1] local
##
## [1] /Library/Frameworks/R.framework/Versions/3.6/Resources/library
```

Unprocessed, raw data available too

- Easy and often desirable for DataPackageR developer to store preprocessed data in the package (inside inst/extdata directory)

```
list.files(file.path(system.file(package = "myDataPackage"), "extdata"))
```

```
## [1] "demographics.csv" "Logfiles"          "raw-data.csv"
```

DataPackageR
Background

For VISC, raw data is standardized but generated from many different sources.

DataPackageR is used to collate, quality control, and clean data for analysis

Schematic of a reproducible workflow using DataPackageR

Finak G, Mayer B, Fulp W et al. DataPackageR: Reproducible data preprocessing, standardization and sharing using R/Bioconductor for collaborative data analysis [version 2]. Gates Open Res 2018, 2:31 (doi: 10.12688/gatesopenres.12832.2)

DataPackageR organizes R packages for housing data

`datapackage_skeleton()`

Configuration file controlling the package build process

```
package root
|--- datapackager.yml #
|--- DESCRIPTION # Adds a DataVersion
|                 # string to version the
|                 # data set.
|--- NAMESPACE
|--- DATADIGEST # Stores an MD5 hash of each
|                 # data object in the package.
|--- R
|--- Read-and-delete-me.txt    # Further instructions
|--- data          # Holds processed, analysis-ready data objects.
|--- data-raw      #
|--- documentation.R # Auto generated roxygen documentation
|                     # for data set objects.
|--- inst
| |___ extdata # (small) raw data files.
| |___ doc   # Processed vignettes are moved here.
|            # Data processing code is accessible in the
|            # final package via the vignette() API.
|--- vignettes # Scripts in data-raw
|                # are processed into vignettes.
|___ man # Autogenerated documentation is processed
         # into rd files.
```

User code for data

DataPackageR provides a suite of functions to easily navigate this directory structure.

Finak G, Mayer B, Fulp W et al. DataPackageR: Reproducible data preprocessing, standardization and sharing using R/Bioconductor for collaborative data analysis [version 2]. Gates Open Res 2018, 2:31 (doi: 10.12688/gatesopenres.12832.2)

Configuration yaml: specifying processing scripts and output data

```yaml
configuration:
  files:                              # files property lists
    process_dataset_one.Rmd:          # R or Rmd code files
      name: process_dataset_one.Rmd   # Each file has a name
      enabled: yes                    # The enabled property
specifies
                                      # if the file should be
processed
    process_dataset_two.Rmd:
      name: process_dataset_two.Rmd
      enabled: yes
  objects:                            # A list of the data objects
created
  - dataset_one                       # by processing the files.
  - dataset_two
  - dataset_three
render_root:                          # root directory where scripts
are
  tmp: '298918'                       # render()ed for multi-script
                                      # pipelines
```

DataPackageR provides the API for configuring the yaml file.

Once the code and data is ready: build_package() builds the package

Let's build a data package with DataPackageR

An example: HIV vaccine meta-analysis

- AIDSVAX B/E was one of the vaccines used in a partially-efficacious HIV trial

- **A diligent researcher wants to look at participant responses in three early clinical studies of AIDSVAX**

  - Specifically want datasets with responses measured by antibody levels

  - But… datasets currently exist in multiple places

All example data is publicly available:

DataSpaceR API

https://dataspace.cavd.org

Getting (false) started

```r
library(DataPackageR)
datapackage_skeleton("AIDSVAX-data")
```

```
## Error: 'AIDSVAX-data' is not a valid package name. It should:
## * Contain only ASCII letters, numbers, and '.'
## * Have at least two characters
## * Start with a letter
## * Not end with '.'
```

PSA: Remember R packages have rules around names

Initializing the data package

```
datapackage_skeleton("AIDSVAXdata",
                     r_object_names = "meta_aidsvax")
```
← Optional: specify output data name here

```
## ✔ Setting active project to '/BAMA/AIDSVAXdata'
## ✔ Leaving 'DESCRIPTION' unchanged
## Package: AIDSVAXdata
## Title: What the Package Does (One Line, Title Case)
## Version: 0.0.0.9000
## Authors@R (parsed):
##     * First Last <first.last@example.com> [aut, cre] (<https://orcid.org/YOUR-ORCID-ID>)
## Description: What the package does (one paragraph).
## License: What license it uses
## Encoding: UTF-8
## LazyData: true
## ✔ Setting active project to '<no active project>'
## ✔ Setting active project to '/BAMA/AIDSVAXdata'
## ✔ Added DataVersion string to 'DESCRIPTION'
## ✔ configured 'datapackager.yml' file
```

```
datapackager.yml

1  configuration:
2    files: {}
3    objects: meta_aidsvax
4    render_root:
5      tmp: '262198'
6
```

initial yaml file

Storing raw data in the package

- **In this example: we'll put the raw data into inst/extdata folder in the package**

```
.
├── DESCRIPTION
├── R
├── Read-and-delete-me
├── data
├── data-raw
├── datapackager.yml
└── inst
    └── extdata
        ├── CAVD526.xlsx
        ├── HVTN097.xlsx
        └── HVTN105.xlsx
```
(manually added these)

- Often raw data cannot be stored in the package:
  the processing script can access it in remote locations
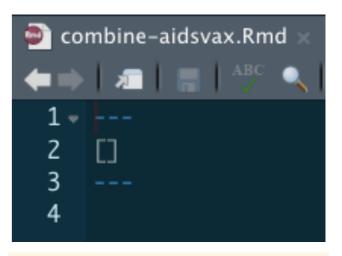
Initializing the processing script

If the processing script exists it will be copied to data-raw/ otherwise a blank script is generated

```
## specify data objects (we already did this)
# use_data_object("meta_aidsvax")

## create/specify processing scripts
use_processing_script("combine-aidsvax.Rmd")
```

```
## ✔ Setting active project to '/BAMA/AIDSVAXdata'
```

```
## Warning in normalizePath(file.path(proj_path, "data-raw",
## basename(raw_file)), : path[1]="/BAMA/AIDSVAXdata/data-raw/combine-
## aidsvax.Rmd": No such file or directory
```

```
## Attempting to create  combine-aidsvax.Rmdconfiguration:
##    files:
##      combine-aidsvax.Rmd:
##        enabled: yes
##    objects: meta_aidsvax
##    render_root:
##      tmp: '262198'
```

Output confirms updated yaml

```
.
├── DESCRIPTION
├── R
├── Read-and-delete-me
├── data
├── data-raw
│   └── combine-aidsvax.Rmd
├── datapackager.yml
└── inst
    └── extdata
        ├── CAVD526.xlsx
        ├── HVTN097.xlsx
        └── HVTN105.xlsx
```

Write the code

```
combine-aidsvax.Rmd ×
```

```
 1  ---
 2  title: "Combining AIDSVAX data"
 3  author: "Bryan Mayer"
 4  date: "2019-06-08"
 5  ---
 6
 7  ```{r setup}
 8  library(tidyverse)
 9  library(DataPackageR)
10  ```
11
12  ```{r processing}
13
14  data_to_process = c("HVTN097.xlsx", "HVTN105.xlsx")
15
16  meta_aidsvax = map_df(data_to_process,
17                        ~readxl::read_excel(project_extdata_path(.x), sheet = "NAb"))
18
19  ```
20
21  ```{r error-checking}
22
23  # VERY IMPORTANT TESTS
24
25  ```
```

Useful DataPackageR function for accessing raw data
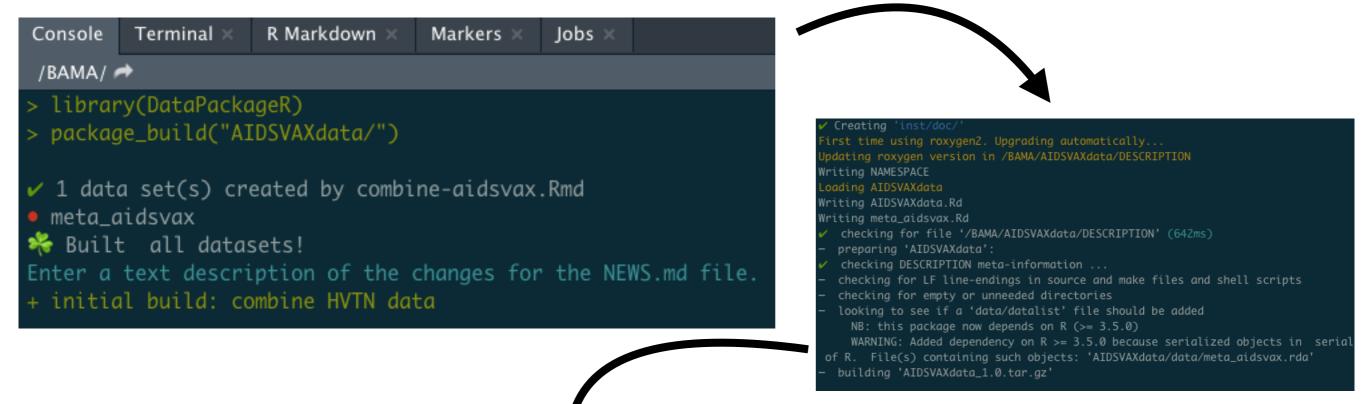
```
DataPackageR::project_extdata_path()
```

```
## [1] "/BAMA/AIDSVAXdata/inst/extdata"
```

Build the package!


```
Console    Terminal ×    R Markdown ×    Markers ×    Jobs ×

/BAMA/ ➜

> library(DataPackageR)
> package_build("AIDSVAXdata/")

✔ 1 data set(s) created by combine-aidsvax.Rmd
● meta_aidsvax
🍀 Built  all datasets!
Enter a text description of the changes for the NEWS.md file.
+ initial build: combine HVTN data
```

```
✔ Creating 'inst/doc/'
First time using roxygen2. Upgrading automatically...
Updating roxygen version in /BAMA/AIDSVAXdata/DESCRIPTION
Writing NAMESPACE
Loading AIDSVAXdata
Writing AIDSVAXdata.Rd
Writing meta_aidsvax.Rd
✔  checking for file '/BAMA/AIDSVAXdata/DESCRIPTION' (642ms)
─  preparing 'AIDSVAXdata':
✔  checking DESCRIPTION meta-information ...
─  checking for LF line-endings in source and make files and shell scripts
─  checking for empty or unneeded directories
─  looking to see if a 'data/datalist' file should be added
    NB: this package now depends on R (>= 3.5.0)
    WARNING: Added dependency on R >= 3.5.0 because serialized objects in  serial
of R.  File(s) containing such objects: 'AIDSVAXdata/data/meta_aidsvax.rda'
─  building 'AIDSVAXdata_1.0.tar.gz'
```

```
Next Steps
1. Update your package documentation.
    - Edit the documentation.R file in the package source data-raw subdirectory and update the roxygen mar
kup.
    - Rebuild the package documentation with  document() .
2. Add your package to source control.
    - Call  git init .  in the package source root directory.
    -  git add  the package files.
    -  git commit  your new package.
    - Set up a github repository for your pacakge.
    - Add the github repository as a remote of your local package repository.
    -  git push  your local repository to gitub.
[1] "/BAMA/AIDSVAXdata_1.0.tar.gz"
```

Time to fill in documentation!

Then run document() to build help files for your packaged data

```r
documentation.R ✕

Source on Save

1   #' AIDSVAXdata
2   #' A data package for AIDSVAXdata.
3   #' @docType package
4   #' @aliases AIDSVAXdata-package
5   #' @title Package Title
6   #' @name AIDSVAXdata
7   #' @description A description of the dat
8   #' @details Use \code{data(package='AIDS
    data sets in this data package
9   #'        and/or DataPackageR::load_all
10  #' _datasets() to load them.
11  #' @seealso
12  #' \link{meta_aidsvax}
13  NULL
14
15
16
17  #' Detailed description of the data
18  #' @name meta_aidsvax
19  #' @docType data
20  #' @title Descriptive data title
21  #' @format a \code{tbl_df} containing th
22  #' \describe{
23  #' \item{Subject Id}{}
24  #' \item{Study}{}
25  #' \item{Treatment Summary}{}

1:51   (Top Level) ⬍
```

Now the package is ready for installation and loading

```r
library(knitr)
library(tidyverse)
devtools::install_local("AIDSVAXdata/")
```

```r
DataPackageR::data_version("AIDSVAXdata")
```

```
## [1] '0.1.0'
```

```r
library(AIDSVAXdata)
data("meta_aidsvax")

meta_aidsvax %>%
  glimpse()
```

```
## Observations: 2,899
## Variables: 7
## $ `Subject Id`        <chr> "vtn097 006", "vtn097 006", "vtn097 006", "v…
## $ Study               <chr> "HVTN 097", "HVTN 097", "HVTN 097", "HVTN 09…
## $ `Treatment Summary` <chr> "Group 2 Arm T2 Vaccine", "Group 2 Arm T2 Va…
## $ `Study days`        <dbl> 28, 28, 28, 28, 28, 28, 210, 210, 210, 210, …
## $ `Target cell`       <chr> "TZM-bl", "TZM-bl", "TZM-bl", "TZM-bl", "TZM…
## $ `Titer ID50`        <dbl> 5, 5, 5, 5, 5, 5, 54, 55, 5, 5, 5, 5, 37, 5,…
## $ `Virus name`        <chr> "MW965.26", "TH023.6", "MN.3", "BaL.26", "SF…
```

Updating the data

Small edit to processing code

```
data_to_process = c("HVTN097.xlsx", "HVTN105.xlsx", "CAVD526.xlsx")
```

Rebuild the package

```
> package_build("AIDSVAXdata/")

✔ Setting active project to '/BAMA/AIDSVAXdata'
✔ 1 data set(s) created by combine-aidsvax.Rmd
● meta_aidsvax
🍀 Built  all datasets!
WARN [2019-05-31 15:27:46] meta_aidsvax has changed.
WARN [2019-05-31 15:27:46] meta_aidsvax has changed.
WARN [2019-05-31 15:27:46] meta_aidsvax has changed.
Enter a text description of the changes for the NEWS.md file.
+ Add CAVD data
```
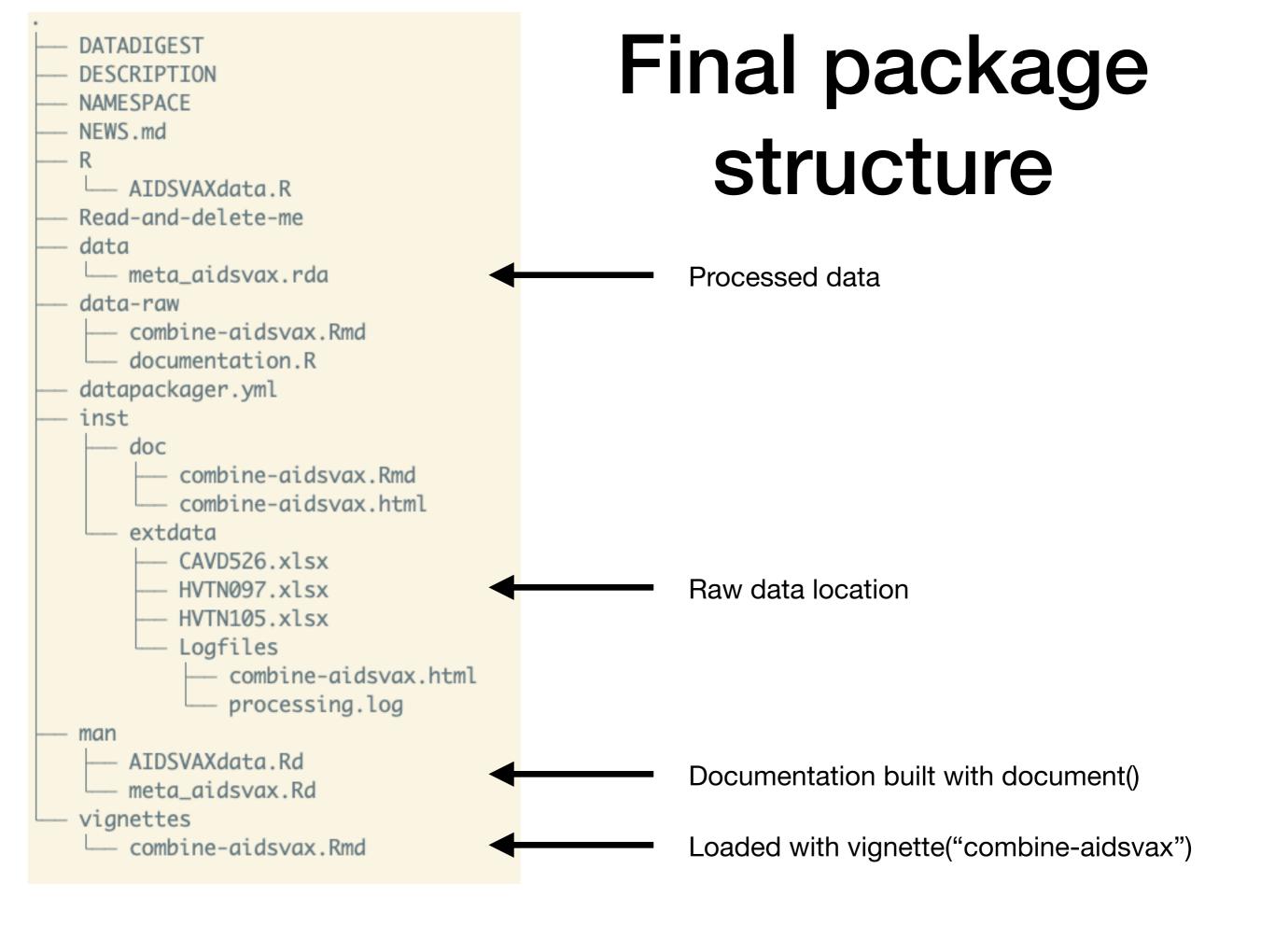
The data and data version are now updated

```r
library(knitr)
library(tidyverse)
devtools::install_local("AIDSVAXdata/")
```

```r
DataPackageR::data_version("AIDSVAXdata")
```

```
## [1] '0.1.1'
```

```r
library(AIDSVAXdata)
data("meta_aidsvax")

meta_aidsvax %>%
  glimpse()
```

```
## Observations: 6,632
## Variables: 7
## $ `Subject Id`        <chr> "vtn097 006", "vtn097 006", "vtn097 006", "v…
## $ Study               <chr> "HVTN 097", "HVTN 097", "HVTN 097", "HVTN 09…
## $ `Treatment Summary` <chr> "Group 2 Arm T2 Vaccine", "Group 2 Arm T2 Va…
## $ `Study days`        <dbl> 28, 28, 28, 28, 28, 28, 210, 210, 210, 210, …
## $ `Target cell`       <chr> "TZM-bl", "TZM-bl", "TZM-bl", "TZM-bl", "TZM…
## $ `Titer ID50`        <dbl> 5, 5, 5, 5, 5, 5, 54, 55, 5, 5, 5, 5, 37, 5,…
## $ `Virus name`        <chr> "MW965.26", "TH023.6", "MN.3", "BaL.26", "SF…
```

Final package structure

```
.
├── DATADIGEST
├── DESCRIPTION
├── NAMESPACE
├── NEWS.md
├── R
│   └── AIDSVAXdata.R
├── Read-and-delete-me
├── data
│   └── meta_aidsvax.rda          ◄─────────── Processed data
├── data-raw
│   ├── combine-aidsvax.Rmd
│   └── documentation.R
├── datapackager.yml
├── inst
│   ├── doc
│   │   ├── combine-aidsvax.Rmd
│   │   └── combine-aidsvax.html
│   └── extdata
│       ├── CAVD526.xlsx
│       ├── HVTN097.xlsx          ◄─────────── Raw data location
│       ├── HVTN105.xlsx
│       └── Logfiles
│           ├── combine-aidsvax.html
│           └── processing.log
├── man
│   ├── AIDSVAXdata.Rd            ◄─────────── Documentation built with document()
│   └── meta_aidsvax.Rd
└── vignettes
    └── combine-aidsvax.Rmd       ◄─────────── Loaded with vignette("combine-aidsvax")
```

Thanks!

Questions?