Complete Setup Guide for datalab Package & Workshop

This guide explains how to set up the complete datalab package and workshop materials from scratch.

Table of Contents

- 1. Quick Start for Package Developers
- 2. <u>Directory Structure</u>
- 3. Building the Package
- 4. Running the Workshop
- 5. Customization Guide

Quick Start for Package Developers

Prerequisites

```
install.packages(c("devtools", "roxygen2", "testthat", "knitr", "rmarkdown", "usethis"))
```

Create Package Structure

```
bash

# Create main package directory

mkdir -p datalab/R

mkdir -p datalab/data

mkdir -p datalab/data-raw

mkdir -p datalab/man

mkdir -p datalab/tests/testthat

mkdir -p datalab/vignettes

mkdir -p datalab/workshop
```

Add Files

- 1. Copy (functions.R) to (R/functions.R)
- 2. Copy DESCRIPTION to package root
- 3. Copy NAMESPACE to package root

- 4. Copy test file to (tests/testthat/test-functions.R)
- 5. Copy vignette to (vignettes/datalab-intro.Rmd)
- 6. Copy workshop materials to (workshop/)
- 7. Add (LICENSE) file
- 8. Add (README.md)

Generate Sample Data

```
# Run in R
setwd("datalab")
source("data-raw/create_sample_data.R")
```

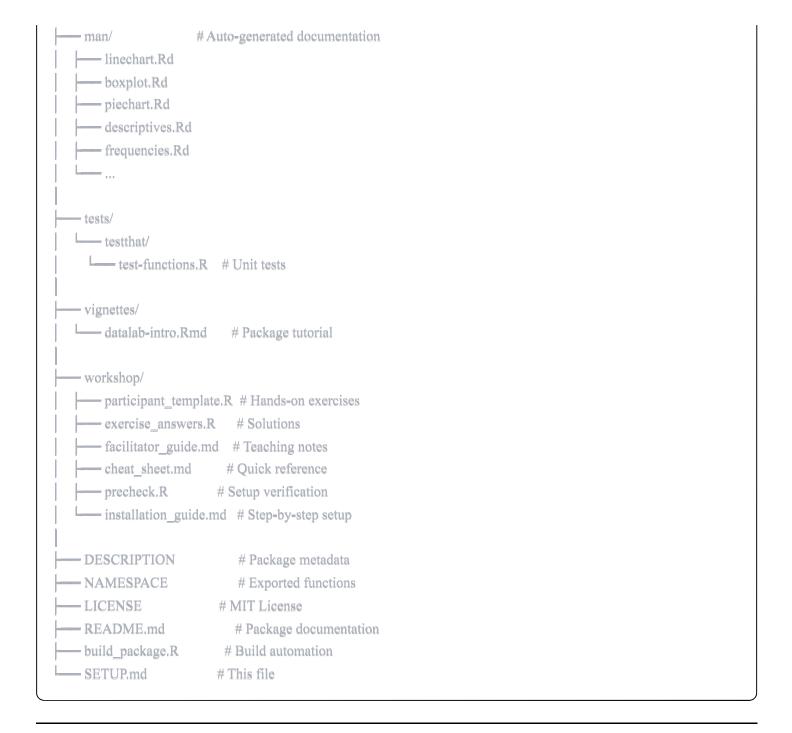
Build Package

```
setwd("datalab")
source("build_package.R")
```

This creates:

- datalab_0.1.0.tar.gz (package file for distribution)
- (datalab_workshop_bundle.zip) (workshop materials)

Directory Structure



Building the Package

Method 1: Using build_package.R (Recommended)

```
# From package root directory
source("build_package.R")
```

This automatically:

• Generates documentation

- Runs tests
- Checks package
- Builds .tar.gz file
- Creates workshop bundle

Method 2: Manual Build

```
library(devtools)

# Step-by-step
document() # Generate documentation
test() # Run tests
check() # Check package
build() # Create .tar.gz
build_vignettes() # Build vignettes
```

Verification

```
# Install locally
install.packages("datalab_0.1.0.tar.gz", repos = NULL, type = "source")

# Test
library(datalab)
data(heart_data)
data <- heart_data
piechart(chest_pain_type)
```

Running the Workshop

Pre-Workshop (1-2 Weeks Before)

1. Distribute Materials:

- Send datalab_0.1.0.tar.gz to participants
- Send (installation_guide.md)
- Send precheck.R

2. Participant Instructions:

Please complete before the workshop:

- 1. Install R and RStudio (see installation_guide.md)
- 2. Install required packages
- 3. Install datalab package
- 4. Run precheck.R to verify setup

3. Set Up Support:

- Schedule office hours for installation help
- Create FAQ document
- Set up communication channel (email/Slack)

Workshop Day

Setup (30 min before):

- 1. Test your own setup
- 2. Open (participant template.R) in RStudio
- 3. Have (exercise_answers.R) open in second window
- 4. Load sample data
- 5. Test projector/screen sharing

Materials Needed:

- Projector/screen share working
- (facilitator_guide.md) printed or on second screen
- (participant_template.R) ready to share
- (cheat_sheet.md) ready to distribute

Backup Plans:

- Participants without working setup: pair them with neighbors
- Network issues: have offline copy of vignette
- Time runs short: skip Exercise 3

Post-Workshop

1. Send Follow-Up:

- Thank you email
- Link to resources
- Feedback survey

2. Share:

- (cheat_sheet.md)
- Vignette link
- Exercise solutions

Customization Guide

Custom Sample Datasets

Edit (data-raw/create_sample_data.R):

```
# Your custom dataset

my_data <- data.frame(
    var1 = ...,
    var2 = ...
)

usethis::use_data(my_data, overwrite = TRUE)
```

Then document in (R/data.R):

```
#' My Custom Dataset

#'

#' Description of your data

#'

#' @format A data frame with X rows and Y variables:

#' \describe{

#' \item{var1}{Description}

#' \item{var2}{Description}

#' }

"my_data"
```

Modify Functions

Edit (R/functions.R) to:

- Change default colors
- Adjust plot themes
- Add new statistics
- Create new visualizations

After changes:

```
devtools::document()
devtools::test()
devtools::build()
```

Customize Workshop Materials

For Different Durations:

- 60 min: Skip Section 4 (line charts) and Exercise 3
- 120 min: Add case study and group activities

For Different Audiences:

- Healthcare: Use only heart_data, add medical interpretation
- Agriculture: Use only agriculture_data, focus on yield optimization
- Business: Create sales/marketing datasets

To Modify:

- 1. Edit (workshop/participant template.R)
- 2. Update (workshop/facilitator guide.md)
- 3. Adjust (workshop/cheat_sheet.md)

Add New Functions

1. Write function in (R/functions.R):

```
#' My New Function
#'
#' @param var Variable name
#' @param data Data frame
#' @return Plot object
#' @export
my_function <- function(var, data = NULL) {
# Implementation
}
```

- 2. Export in NAMESPACE (or use roxygen2)
- 3. Add tests in tests/testthat/test-functions.R
- 4. Update documentation
- 5. Rebuild package

Testing Checklist

Before distributing to participants:

Package installs without errors

All functions work with sample data

Pre-check script runs successfully

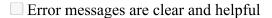
Vignette builds and displays correctly

Workshop materials are complete

Test on fresh R installation

Test on Windows, Mac, or Linux

■ All links in documentation work



Troubleshooting

"Error in roxygen2"

- Update roxygen2: (install.packages("roxygen2"))
- Run (devtools::document()) again

"Tests failed"

- Check (testthat) is installed
- Review test output for specific failures
- Verify sample data loaded correctly

"Check shows warnings"

- Review R CMD check output
- Common issues: missing dependencies, documentation mismatch
- Fix and rebuild

"Vignette doesn't build"

- Check knitr and rmarkdown installed
- Verify vignette YAML header correct
- Test vignette code chunks manually

Deployment Workflow

For a New Workshop

1. Customize

- Update datasets for your domain
- Modify examples in vignette
- Adjust workshop timing in facilitator guide

2. Build

r
source("build_package.R")

3. Test

- Install on clean R installation
- Run through participant_template.R
- Verify all exercises work

4. Distribute

- Email .tar.gz to participants
- Share installation guide.md
- Schedule setup office hours

5. Run Workshop

- Follow facilitator_guide.md
- Be flexible with timing
- Encourage questions

6. Follow Up

- Collect feedback
- Share resources
- Iterate for next time

Version Control

Recommended Git Workflow

bash

```
git init
git add .
git commit -m "Initial datalab package"

# For updates
git add R/functions.R
git commit -m "Added new visualization function"
git tag -a v0.2.0 -m "Version 0.2.0 with new features"
```

Version Numbering

- 0.1.0 Initial release
- 0.1.1 Bug fixes
- 0.2.0 New features
- 1.0.0 Production ready

Update version in (DESCRIPTION) file.

Additional Resources

- R Packages Book: https://r-pkgs.org/
- roxygen2 Documentation: https://roxygen2.r-lib.org/
- testthat Documentation: https://testthat.r-lib.org/
- ggplot2 Documentation: https://ggplot2.tidyverse.org/

Support

For questions or issues:

- 1. Check this guide
- 2. Review facilitator_guide.md
- 3. Check R Packages book
- 4. Contact package maintainer

Ready to start? Follow the Quick Start section above to build your first version of the package!