

# Helpful tools for efficient and reproducible research

Hansen Johnson

PhD Student  
Oceanography Department, Dalhousie University  
hansen.johnson@dal.ca

MEOPAR Annual Training Meeting  
Victoria, BC  
June 11, 2019

Presentation online at:  
[https://hansenjohnson.org/talk/2019\\_meopar\\_atm/](https://hansenjohnson.org/talk/2019_meopar_atm/)

## Tools of the trade

- Science education teaches research concepts, but expects technical skills
- Using the tools can make a HUGE difference for efficiency and reproducibility
- Little time or resources are allocated to developing technical skills and best practices
- Students must spend their limited time learning for themselves

# My background

- Biology major in undergrad
- No training in computer programming or technical aspects of research
- Started grad school at Dalhousie in 2015
- Given a project that is **impossible** without technical chops

# My background

- Luckily I have interest and supportive advisers
- Developed many helpful skills with help from my peers (especially Christoph Renkl) and the internet
- Hope to help others acquire these skills more efficiently

Some examples:

Methods in Ten Minutes:

<https://christophrenkl.github.io/mtm/>

R/Python Programming Tutorials:

[https://christophrenkl.github.io/programming\\_tutorials/](https://christophrenkl.github.io/programming_tutorials/)

# Today's Goal

**Goal:** Provide some tools and concepts that I find essential for research

- Imagine we've been given some data on sea ice coverage and asked to characterize how it has changed over time
- Approach this simple project in 3 steps:
  - Analyse the data
  - Write a report
  - Document the workflow
- We'll pause briefly after each section for questions and/or discussion

# Analysis

**Goal:** Process and plot some data

- 1 Structure the project
- 2 Read, process, and save data
- 3 Make and save plots

You will need

**R** ([www.r-project.org](http://www.r-project.org))  
**Rstudio** ([www.rstudio.com](http://www.rstudio.com))

# A good project structure

A well-structured project allows you or someone else to easily understand and even reproduce the workflow

Organizing a project helps you:

- Expand, revisit and update efficiently
- Have confidence in the results
- Collaborate easily

# Project structure

Projects vary and organizing them is hard. Some tips:

- Keep an untouchable ‘sacred data directory’ for raw data
- Dedicated directories for outputs (processed data and plots)
- Use simple file/folder names (ideally without spaces)
- Try to be consistent among projects
- Document prolifically (more later)

Check out CookieCutter Data Science for more details



## Example [simple] project structure

```

example.....Project directory
├── data.....All data
│   ├── processed.....Processed data by code in src
│   └── raw.....Raw data - never touch!
├── figures.....Plots produced by code in src
├── reports.....Any reports or presentations
├── src.....All source code
├── wrk.....Development sandbox
├── readme.md.....Project description
└── master.R.....Master script
  
```

# R and Rstudio

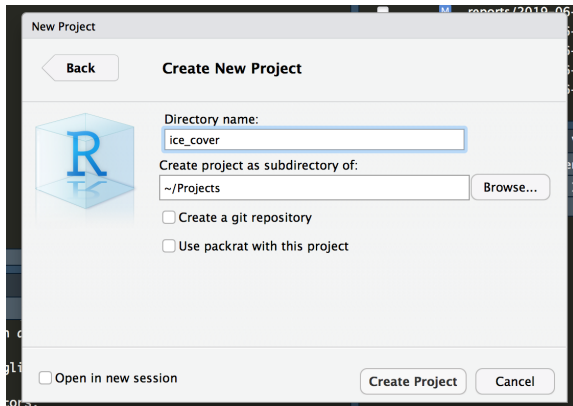


*The basics of R and Rstudio are outside the scope of this session. See the tutorial here for more information:*

[https://christophrenkl.github.io/programming\\_tutorials/](https://christophrenkl.github.io/programming_tutorials/)

# R and Rstudio

- 1 Open Rstudio
- 2 Create new project in a logical place with a short, descriptive name (e.g., `~/Projects/ice_cover`)



## Get the data

Download data from:

`https://www.canada.ca/en/environment-climate-change/  
services/environmental-indicators/sea-ice.html`

Save the file in `data/raw/`

## Process the data

Create a script called `src/process_data.R` to:

- 1 Read in data from `data/raw/`
- 2 Clean and format
- 3 Save output in `data/processed/`

# src/process\_data.R

```
## process_data ##
# Read, process, and save ice cover timeseries data

# input -----

# choose data file
infile = "data/raw/1.SeaIce-NCW-EN.csv"

# choose output file
outfile = "data/processed/ice_cover.rda"

# process -----

# read in data and rename columns
df = read.csv(infile, skip = 2, col.names = c("year", "ice_cover"))

# remove missing values
df = df[complete.cases(df),]

# format year
df$year = as.numeric(as.character(df$year))

# save
save(df, file = outfile)
```

## Plot the data

Create a script called `src/plot_timeseries.R` to:

- 1 Read in data from `data/processed/`
- 2 Make plot
- 3 Save output in `figures/timeseries.png`

## src/plot\_timeseries.R

```
## plot_timeseries ##
# Make and save an ice cover timeseries plot

# input -----

# data file
infile = "data/processed/ice_cover.rda"

# plot file
outfile = "figures/timeseries.png"

# setup -----

# external libraries
library(ggplot2)

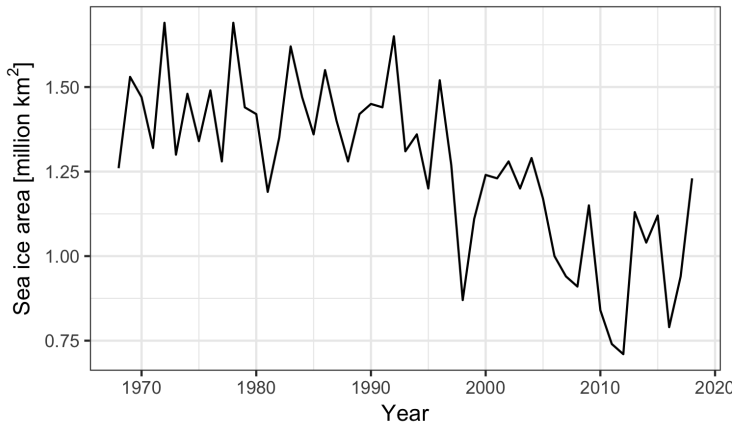
# process -----

# plot
plt = ggplot(df)+
  geom_path(aes(x=year, y=ice_cover))+
  labs(x="Year", y=expression(paste("Sea ice area [million"," km"^2,"]")))+
  theme_bw()

# save
ggsave(plt, filename = outfile, height = 3, width = 5, units = "in", dpi = 300)
```



figures/timeseries.png



# Simple project orchestration with a master script

Create a master file to execute all the analysis steps in the correct order. This should:

- 1 Run `src/process_data.R`
- 2 Run `src/plot_timeseries.R`

# master.R

```
## master ##  
# Process and plot example ice cover timeseries  
  
# process raw data  
source("src/process_data.R")  
  
# plot timeseries  
source("src/plot_timeseries.R")
```

# Play around

The project is totally reproducible from raw data! Now you can:

- Make changes to either the plotting or the processing script
- Delete anything in data/processed or figures

And simply run `master.R` to re-build the entire project!

## Key concepts

- Never edit raw data!
- All processed data and figures should be reproducible from raw data
- Use a master script (or other means) to orchestrate data processing
- Take time to improve code readability (use comments, indent, consolidate inputs, etc.)

### Possible next steps

- Use `Make` instead of a master script to orchestrate the project more efficiently
- Use `symlinks` to link to large datasets that are stored remotely
- Use functions for repeated tasks

# BREAK

Questions?

How do you keep your projects organized?

# Writing

**Goal:** Find and organize references and draft a research report

- 1 Find references
- 2 Organize and review references with Zotero
- 3 Write and cite document with Word / LibreOffice

You will need

**Zotero** ([www.zotero.org](http://www.zotero.org))

**LibreOffice** ([www.libreoffice.org](http://www.libreoffice.org))

OR

**Microsoft Office [paid]** (<https://products.office.com/>)

# Introducing Zotero



An open-source, one stop shop for acquiring, organizing,  
reviewing, and citing references



# Acquiring

- 1 Install Zotero plugin for web browser
- 2 Find a reference (usually w/ Google Scholar)
- 3 Navigate to the journal page
- 4 Right click anywhere on the page and select Save to Zotero (Embedded Metadata)

SHARE

SPECIAL REVIEWS



## Perspectives on the Arctic's Shrinking Sea-Ice Cover

Mark C. Serreze<sup>1,\*</sup>, Marika M. Holland<sup>2</sup>, Julianne Stroeve<sup>1</sup>

+ See all authors and affiliations

Science, 16 Mar 2007

3-1536  
26

Back

Forward

Reload

Save As...

Print...

Cast...

Translate to English

Save to Zotero

View Page Source

Inspect

Speech

Figures &amp; Data

Info &amp; Metrics

eLetters

PDF

Save to Zotero (HighWire 2.0)

Save to Zotero (Embedded Metadata)

Save to Zotero (DOI)

Save to Zotero (Web Page with Snapshot)

Save to Zotero (Web Page without Snapshot)

...are negative in every month.

...ability in the coupled ice-

...ed with rising

...atmospheric greenhouse gases, the latter supported by evidence of

# Organizing

Open Zotero application and browse references. You can:

- Search / sort by author, year, journal, etc.
- Organize into project folders / collections / tags
- Add items from scratch

The screenshot shows the Zotero application window. The top toolbar includes icons for adding items, collections, tags, and notes. The search bar contains the text "sea ice cover". The main list displays a table of references with columns for Title, Publication, Creator, and Year. The first item, "Thinning of the Arctic sea-ice cover" by Rothrock et al. (1999), is selected. The right-hand pane shows the details for this item, including its title, author (Rothrock, D. A.), and publication information (Geophysical Research Letters, 2010).

Title	Publication	Creator	Year
Thinning of the Arctic sea-ice cover	Geophysic...	Rothroc...	1999
Subarctic cetaceans in the southern Chu...	Oceanogra...	Clarke e...	2013
Age and growth estimates of bowhead w...	Canadian J...	George ...	1998
Comparing marine mammal acoustic ha...	Polar Biology	Moore e...	2011
Updated 1978-2001 abundance estimat...	Journal of ...	Zeh and...	2004
Satellite Tracking of Western Arctic Bow...	Alaska Dep...	Quaken...	2010
Abundance and Population Trend (1978-...	Marine Ma...	George ...	2003
Assessing the potential of autonomous s...	Methods in...	Suberg ...	2014
An Overview of Fixed Passive Acoustic ...	Oceanogra...	Mellinge...	2006
Acoustically Detected Year-Round Prese...	Conservati...	Morano ...	2012
Relationship between the distribution of ...	Polar Biology	Murase ...	2002

**Info** Notes Tags Related

Item Type Journal Article

Title Thinning of the Arctic sea-ice cover

Author Rothrock, D. A.

Author Yu, Y.

Author Maykut, G. A.

Abstract

Publication Geophysical Research Letters

Volume 26

Issue 23

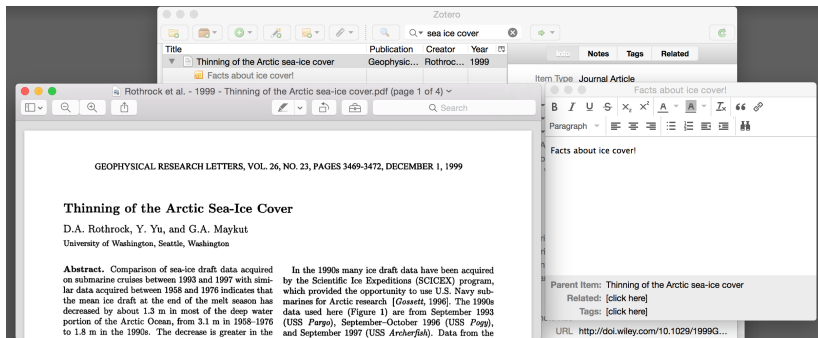
Pages 3469-3472

Date 1999-12-01

# Reviewing

You can:

- View PDFs (with default viewer)
- Add notes / other files / etc
- Update / edit metadata
- Click and drag to share reference



# Write and cite

## In Word / Libre:

- 1 Install Zotero plugin
- 2 Click Zotero tab
- 3 Add references and bibliography with desired style

### The past 30 years of sea ice cover in Canada

June 11, 2019

Hansen Johnson

Sea ice has been in decline for many years (Rothrock et al. 1999). Stroeve et al., (2008) suggest it declined sharply in 2007. This has been confirmed by modeling efforts (Saucier et al. 2003, 2004). Figure 1 shows the timeseries. Here's another citation from {Citation}

**Z** george

#### My Library

**Abundance and Population Trend (1978-2001) of Western Arctic Bowhead Whales Surveyed Near Barrow, Alaska**  
George et al. (2003), *Marine Mammal Science*, 20(4), 755-773.

**Brief overview of the 2010 and 2011 bowhead whale abundance surveys near Point Barrow, Alaska**  
George et al. (2011), *Paper SC/64/AWMP7 presented to the IWC Scientific Committee*.

**Age and growth estimates of bowhead whales (*Balaena mysticetus*) via aspartic acid racemization**  
George et al. (1998), *Canadian Journal of Zoology*, 77(4), 571-580.

**Observations on the ice-breaking and ice navigation behavior of migrating bowhead whales (*Balaena mysticetus*) near Point Barrow, Alaska...**  
George et al. (1988), *Arctic*, 42, 24-30.

# Key concepts

- Use Zotero to acquire, organize, review, and cite references

## Possible next steps

- Use  $\text{\LaTeX}$  for writing reports
- Use  $\text{\LaTeX}$  beamer for making presentations
- Combine text, code and output into documents (html, pdf, word) and presentations (pdf, ppt, html) with Rmarkdown

# BREAK

Questions?

What other tools do you rely on for writing?

# Documentation

**Goal:** Document your work so that you can easily revisit, revert, and share

- 1 Add a readme file
- 2 Tracking changes with `git` and Rstudio
- 3 Remote backups and hosting with GitHub

## You will need

**git** ([www.git-scm.com](http://www.git-scm.com))

**GitHub account** ([www.github.com](http://www.github.com))

## Readme files

What is a readme file?

- Usually simple text (\*.txt) or markdown (\*.md) file
- Includes any information required to implement or interpret the project workflow

Common things to include:

- Brief project background (goals, motivation etc.)
- Description of contents
- System requirements (code, software, etc.)
- Any caveats or known errors / bugs
- To do list
- Links for more information



# readme.md

## # README

Simple project to provide examples of helpful tools and concepts for efficient and reproducible research

## ## Goal

Review recent trends in Canadian sea ice cover

## ## Dataset

Sea ice cover data were downloaded here:

<https://www.canada.ca/en/environment-climate-change/services/environmental-indicators>

## ## Contents

- 'data' - all data

  - 'processed' - cleaned and formatted data ready

  - 'raw' - only raw data \*never touch\*

- 'src' - R code

- 'wrk' - development sandbox

- 'reports' - all presentations, reports, etc

- 'figures' - all figures

- 'master.R' - master script to reproduce full analysis

- 'readme.md' - this file

# What is git?



- Git is a hugely popular version control system (VCS)
- Open source software designed to help you track and document changes to projects
- Originally designed to be run on command line, but many more convenient interfaces now (e.g., Rstudio)

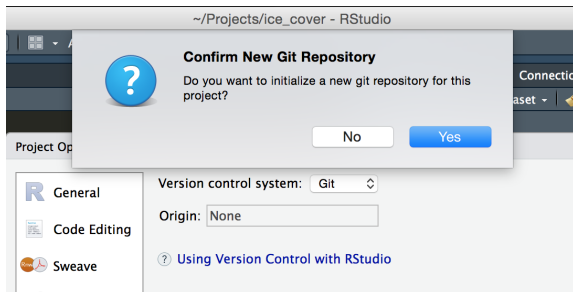
# How does git work?

- git provides a convenient way to save a 'snapshot' of your project at a point in time
- Allows you to review project history and revert one or more files to a previous version
- You must add ('commit') changes to one or more files to the project timeline, and provide a description of your changes



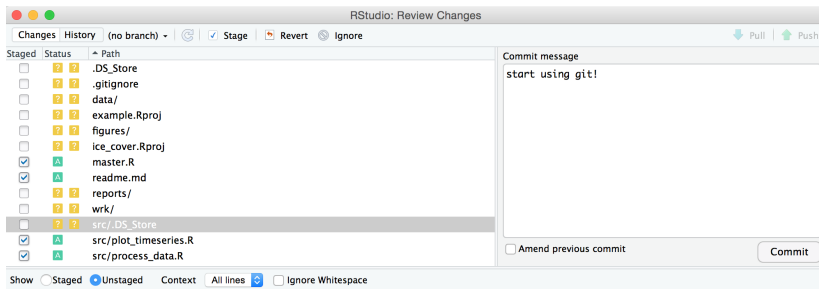
# Using git in Rstudio

- 1 Navigate to Tools -> Version Control -> Project Options -> Git/SVN and switch Version Control System to Git
- 2 Restart Rstudio



# Using git in Rstudio

- 1 Navigate to the **Git** tab and click **Commit**
- 2 Check the boxes next to all `*.R`, and `*.md` files
- 3 Write 'initial commit' in the box and click **Commit**



## Tracking changes with git

- Edit various files and commit the changes
- Click on the `Git` tab, then on the clock icon to view your commit history (project timeline)
- You can view the full project history, or review changes to a particular file
- You can continue working in this self-contained way (i.e., not putting anything online) and track the entire history of your project

*Avoid tracking any large datasets or private info. These can be ignored by listing them by name in a `.gitignore` file*

# Tracking changes with git

RStudio: Review Changes

Changes History master (all commits) Search Pull

Subject	Author	Date	SHA
<b>HEAD -&gt; refs/heads/master</b> update comment for year conversion	Hansen Johnson <hansen.johnson@dal.ca>	2019-06-11	a446dadb
add new section	Hansen Johnson <hansen.johnson@dal.ca>	2019-06-11	dccf276e
change description	Hansen Johnson <hansen.johnson@dal.ca>	2019-06-11	96e03fe1
start using git!	Hansen Johnson <hansen.johnson@dal.ca>	2019-06-11	da5bc69b

Commits 1-4 of 4

SHA a446dadb

Author Hansen Johnson <hansen.johnson@dal.ca>

Date 2019-06-11 07:59

Subject update comment for year conversion

Parent dccf276e

src/process\_data.R

src/process\_data.R View file @ a446dadb

```
@@ -17,7 +17,7 @@ df = read.csv(infile, skip = 2, col.names = c("year", "ice_cover"))
17 17 # remove missing values
18 18 df = df[complete.cases(df),]
19 19
20 20 # format year
20 20 # convert year to numeric
21 21 df$year = as.numeric(as.character(df$year))
22 22
23 23 # save
```

# What is GitHub?



- GitHub is NOT git
- GitHub is a massive hosting service for git repositories
- Provides convenient tools for reviewing and collaborating on code (and free backups!)
- Unlimited free public and private\* repositories


\* Only with  $\leq 3$  collaborators (student accounts are unlimited)



# Creating and linking with GitHub repository

- 1 Go to GitHub user page
- 2 Create a new repository with the same name as our example project (e.g., `ice_cover`)
- 3 Choose to initialize without a readme

Owner Repository name \*

 **hansenjohnson** /  ✓

Great repository names are short and memorable. Need inspiration? How about [upgraded-spoon?](#)

Description (optional)

☒ **Public**  
Anyone can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

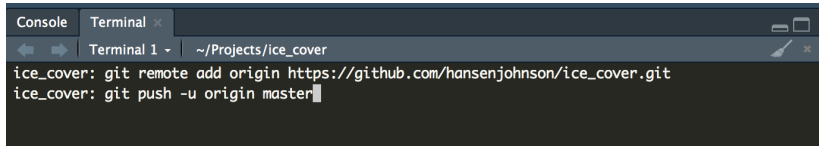
☐ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** / Add a license: **None** ⓘ

**Create repository**

# Creating and linking with GitHub repository

- 1 Copy code listed in "...or push an existing repository from the command line"
- 2 Move to Rstudio and open Tools -> Terminal -> New Terminal
- 3 Paste the lines into the terminal
- 4 Refresh your browser and check out your project online!



The screenshot shows the RStudio interface with the 'Terminal' pane active. The terminal title bar indicates the current directory is ~/Projects/ice\_cover. The terminal output shows two git commands being executed: 'git remote add origin https://github.com/hansenjohnson/ice\_cover.git' and 'git push -u origin master'. The prompt 'ice\_cover:' is visible before each command.

```
ice_cover: git remote add origin https://github.com/hansenjohnson/ice_cover.git
ice_cover: git push -u origin master
```

# Using GitHub

- 1 Make commits on your computer
- 2 When ready, push commits to GitHub by clicking on Push arrow on the git tab in Rstudio
- 3 Check out new code online

The screenshot shows the GitHub repository page for `hansenjohnson / ice_cover`. At the top, there are buttons for Watch (0), Star (0), and Fork (0). Below this is a navigation bar with links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Security, Insights, and Settings. The main content area shows a simple example project to review some helpful tools / concepts for research. Below this, there are statistics: 4 commits, 1 branch, 0 releases, and 1 contributor. A blue bar separates these statistics from the repository content. Below the bar, there are buttons for Branch: master, New pull request, Create new file, Upload files, Find File, and Clone or download. The repository content is displayed in a table with columns for the file name, the commit message, and the time since the last commit. The files listed are `src`, `master.R`, and `readme.md`.

hansenjohnson / ice\_cover

Watch 0 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

Simple example project to review some helpful tools / concepts for research Edit

Manage topics

4 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

hansenjohnson	update comment for year conversion	Latest commit a446dac 13 minutes ago
src	update comment for year conversion	13 minutes ago
master.R	start using git!	17 minutes ago
readme.md	add new section	14 minutes ago

# Using GitHub

- Project contributors (collaborators, or you working on another computer) can `clone` the project onto their computer, `commit` changes, then `push` back to GitHub
- `git` and GitHub have many, many features for organization and collaboration including:
  - Branching
  - Merging / pull requests
  - Issue tracking
  - Website hosting

Check out fantastic GitHub documentation:  
[guides.github.com](https://guides.github.com)

# Key concepts

- 1 Use readme files to describe your project, even if just to yourself
- 2 Use git in Rstudio to track changes
- 3 Use GitHub for backups, sharing, and collaboration

## Possible next steps

- Dig deeper into `git` features (branching, pull requests, merging, etc)
- Use `git` and GitHub for collaboration
- Use Jekyll or Hugo to build project websites and host on GitHub

# Questions?

## **Thanks to:**

Christoph Renkl, Dalhousie Oceanography Student Association (DOSA), Methods in Ten Minutes (MTM), MEOPAR-WHaLE, and more!

## **Link to presentation:**

[https://hansenjohnson.org/talk/2019\\_meopar\\_atm/](https://hansenjohnson.org/talk/2019_meopar_atm/)

## **Link to example project:**

[https://github.com/hansenjohnson/ice\\_cover\\_example/](https://github.com/hansenjohnson/ice_cover_example/)

## **Get in touch:**

[hansen.johnson@dal.ca](mailto:hansen.johnson@dal.ca)