

Neaten up your code: *a briefing*

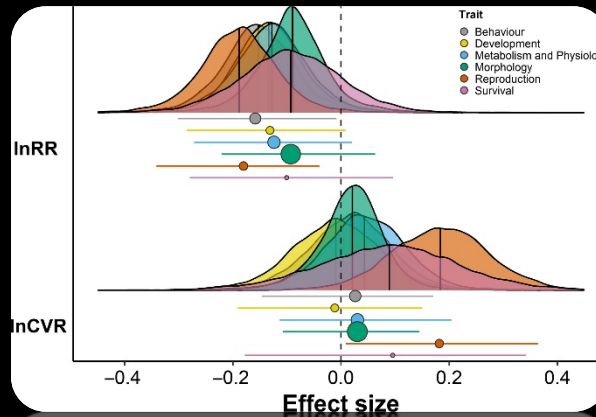
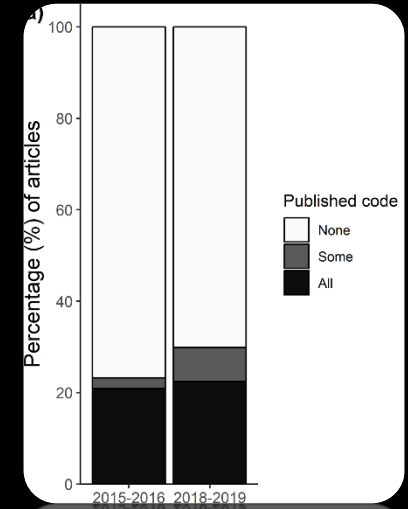
Alfredo Sánchez-Tójar

Living Norway: Ecological Data Network

16th June 2022



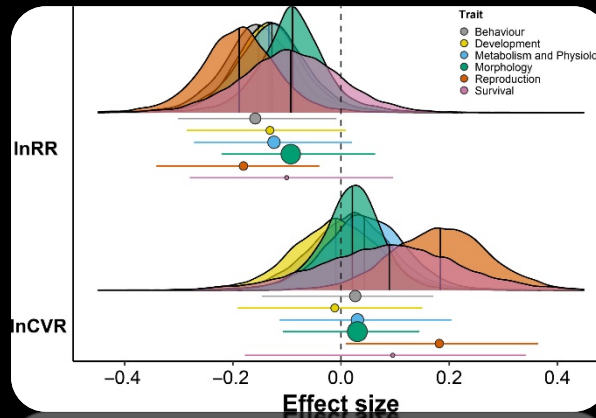
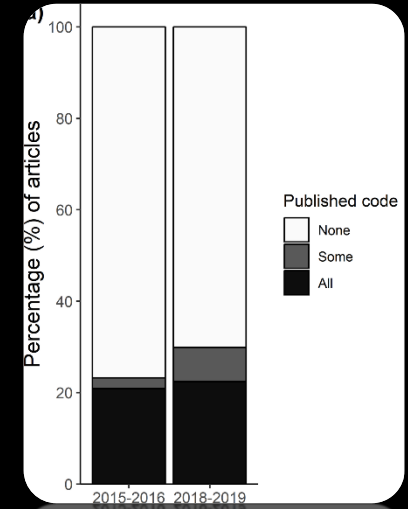
- Evolutionary Ecologist
- Ornithologist by training
- Reconverted into:
 - Evidence synthesist
 - Meta-researcher





I'm NOT a:

- Programmer
- Expert on neat code





Alfredo Sánchez-Tójar

ASanchez-Tojar

I'm a behavioural and evolutionary ecologist. I use R on a daily basis for my studies. I want to share my code with everyone. Transparency.

Edit profile

32 followers · 35 following

alfredo.tojar@gmail.com

<https://scholar.google.co.uk/citations?hl=...>

@ASanchez_Tojar

Achievements



Beta Send feedback

Overview Repositories 15 Projects Packages Stars

ASanchez-Tojar / README.md

Hi there 🙌

I'm Alfredo, an evolutionary ecologist combining empirical studies with a lot of evidence synthesis and meta-science.

I share my code here in my GitHub profile, but you can check out more about my interests in my Twitter profile @ASanchez_Tojar or my Google Scholar profile.

Pinned

Customize your pins

inferring_hierarchies_uncertainty Public

This repository contains scripts used in a manuscript aiming to provide a practical guide for inferring reliable dominance hierarchies and their uncertainty

R 1

meta-analysis_sparrows_ssh Public

It contains the script used for testing the meta-analytic mean of the relationship between bib size and dominance rank.

R

meta-analysis_of_variance Public

This repository contains all R code and data used for a project on the use of meta-analysis of both means and variance in ecology and evolution.

TeX 2

meta-analysis_transgenerational_effects_commentary Public

This project contains the code and data for a project testing the generality and robustness of adaptive "transgenerational" effects. The project consists of a re-analysis of a published meta-analyt...

TeX

code_in_ecology Public

Repository with data and code for a study on code availability in ecology

TeX 1

animal_personality_terminology Public

This repository contains all data and code needed for a project using a self-report questionnaire and a systematic review to understand how animal personality terminology is used and interpreted

TeX



PART I: The problem

PART II: The briefing

PART III: The bad example



Part I: The problem



Why neaten up code?

- For yourself: think of peer-review time or fieldwork interruptions
- For others:
 - Members of the lab: continuity
 - External researchers: open science



The Turing Way

REPRODUCIBLE



THE JOURNAL OF
**WILDLIFE
MANAGEMENT**

Note |  Open Access |  

2016-2018

Computational Reproducibility in The Wildlife Society's Flagship Journals

Althea A. Archmiller✉, Andrew D. Johnson, Jane Nolan, Margaret Edwards, Lisa H. Elliott, Jake M. Ferguson, Fabiola Iannarilli, Juliana Vélez, Kelsey Vitense, Douglas H. Johnson, John Fieberg ... [See fewer authors](#) ^

First published: 17 March 2020 | <https://doi.org/10.1002/jwmg.21855> | Citations: 1

Testable = 18% (n = 74)

Reproducible = 68% (n = 19)

**ROYAL SOCIETY
OPEN SCIENCE**

royalsocietypublishing.org/journal/rsos

Research  

Estimating the reproducibility of social learning research published between 1955 and 2018

Cite this article: Minocher R, Atmaca S, Bavero C, McElreath R, Beheim B. 2021 Estimating the reproducibility of social learning research

Riana Minocher, Silke Atmaca, Claudia Bavero, Richard McElreath and Bret Beheim

Testable = 24% (n = 560)

Reproducible = 81% (n = 40)

THE JOURNAL OF
**WILDLIFE
MANAGEMENT**

Note | Open Access | CC BY

2016-2018

Computational Reproducibility in The Wildlife Society's Flagship Journals

Althea A. Archmiller✉, Andrew D. Johnson, Jane Nolan, Margaret Edwards, Lisa H. Elliott, Jake M. Ferguson, Fabiola Iannarilli, Juliana Vélez, Kelsey Vitense, Douglas H. Johnson, John Fieberg ... See fewer authors ^

First published: 17 March 2020 | <https://doi.org/10.1002/iwmg.21855> | Citations: 1

ROYAL SOCIETY
OPEN SCIENCE

royalsocietypublishing.org/journal/rsos

Research

Check for updates

Cite this article: Minocher R, Atmaca S, Bavero C, McElreath R, Beheim B. 2021 Estimating the reproducibility of social learning research

Estimating the reproducibility of social learning research published between 1955 and 2018

Riana Minocher, Silke Atmaca, Claudia Bavero, Richard McElreath and Bret Beheim

Testable = 18% (n = 74)

Reproducible = 68% (n = 19)

Other fields: 18% to 80%

Testable = 24% (n = 560)

Reproducible = 81% (n = 40)



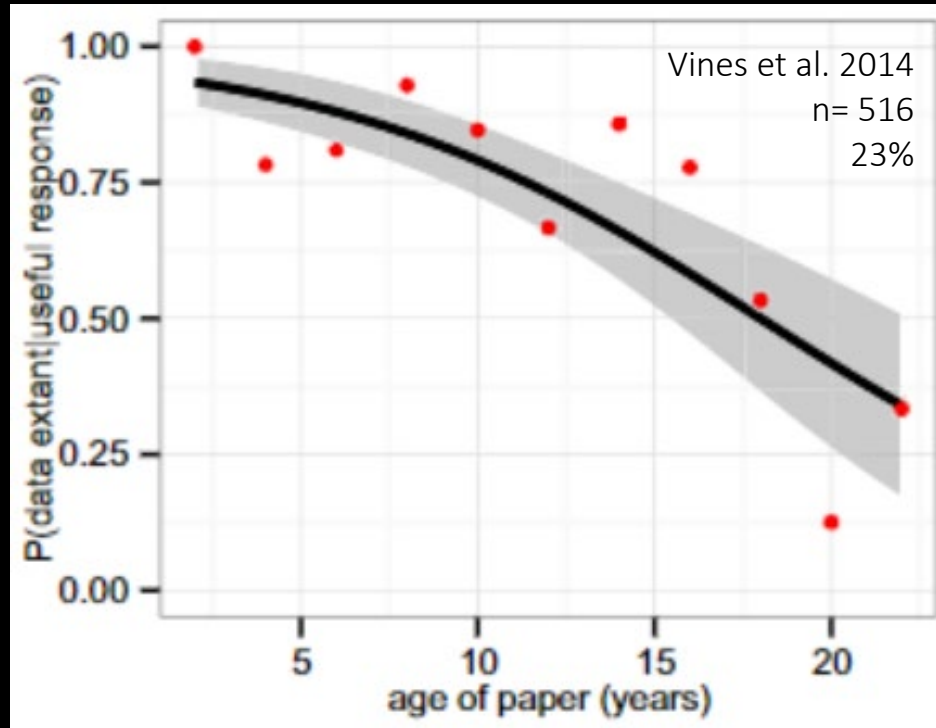
The Turing Way

REPRODUCIBLE



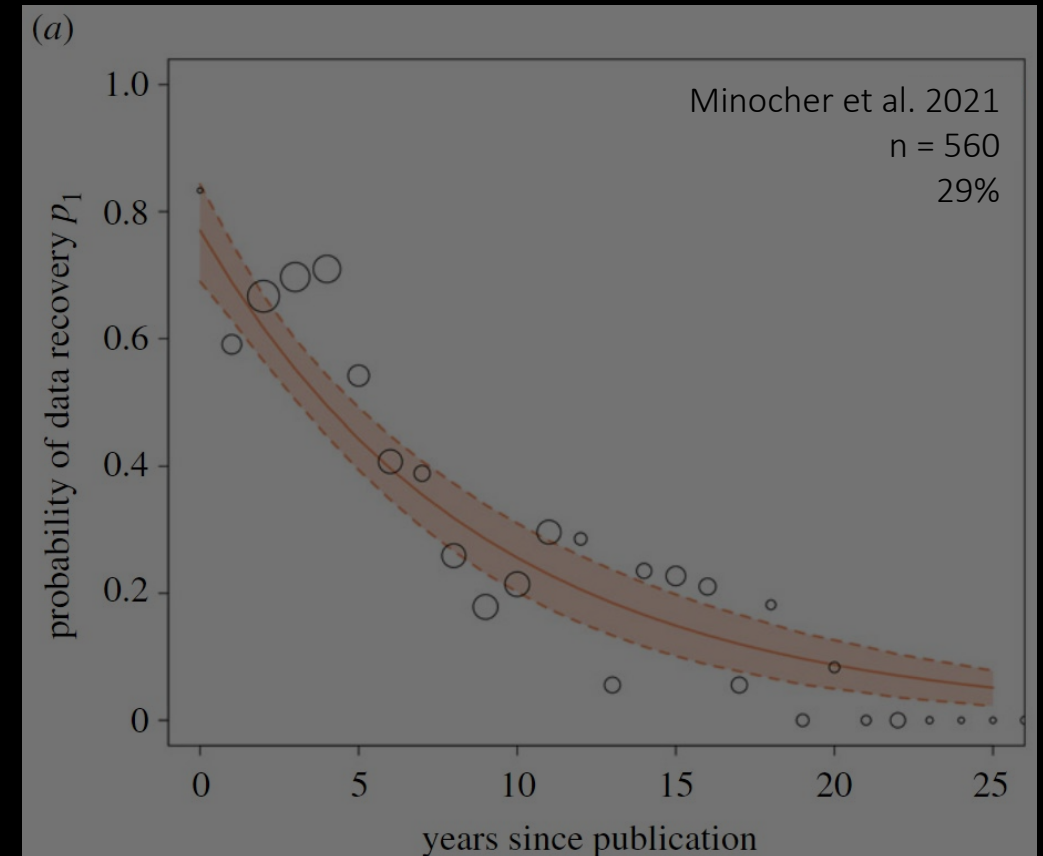
Why? Data availability

1991-2011



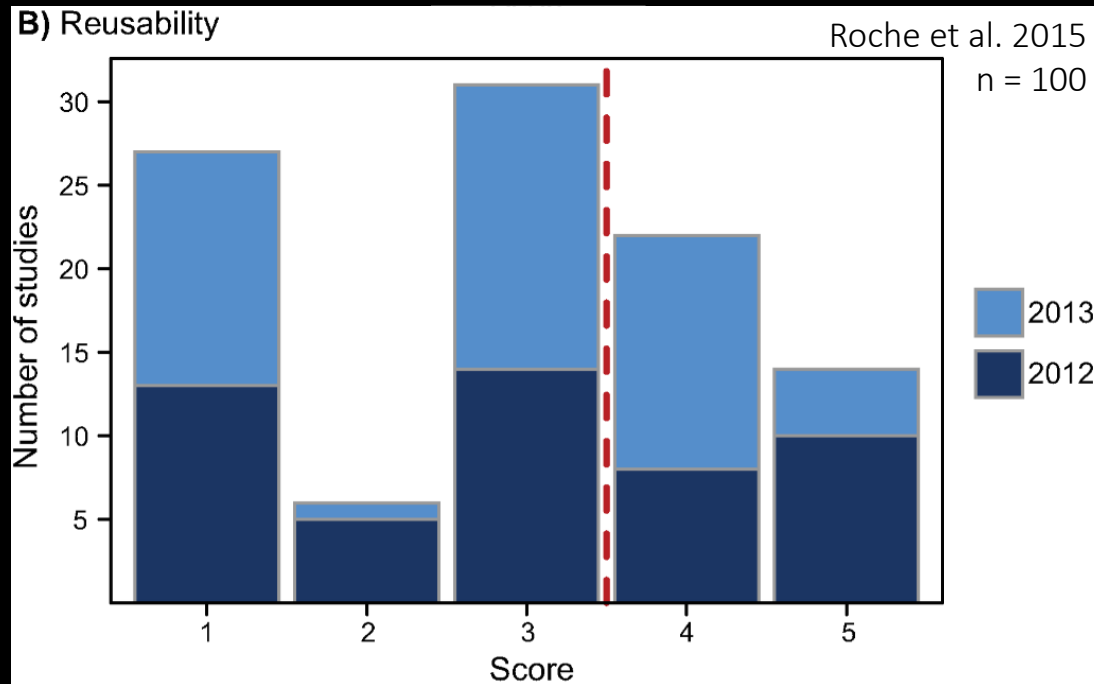
P(data available) reduces 17% every year

1955-2018

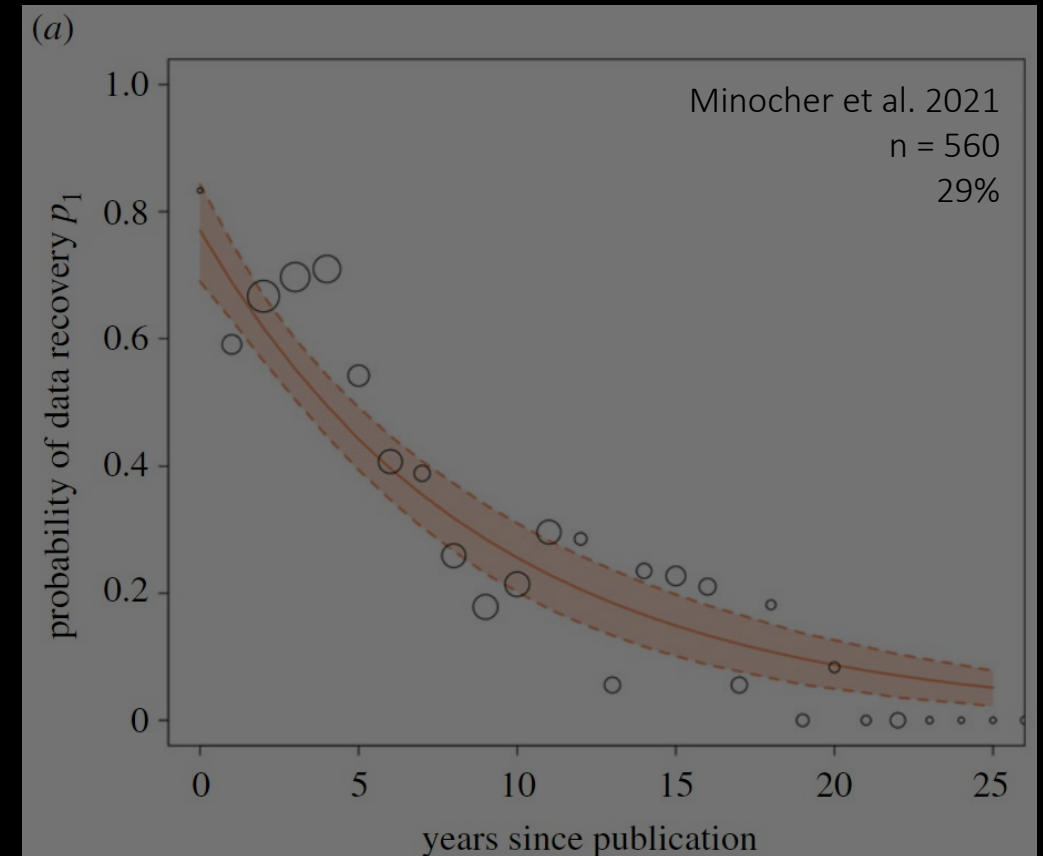


P(data available) halves every 6 years

Why? Data reusability



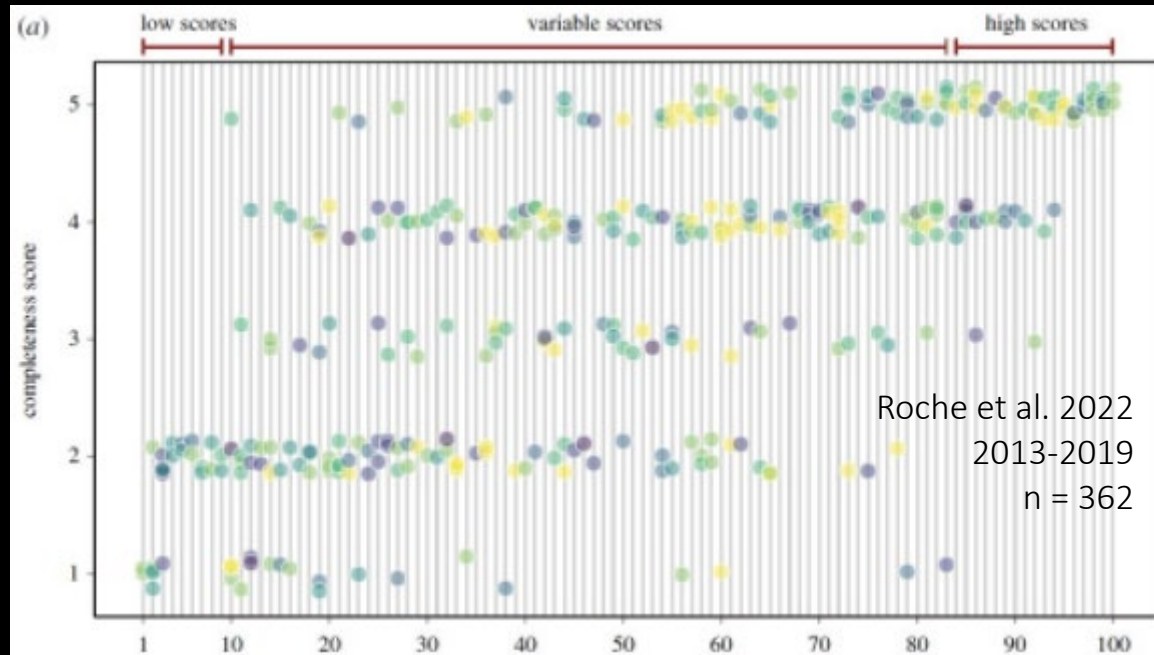
56% datasets are incomplete



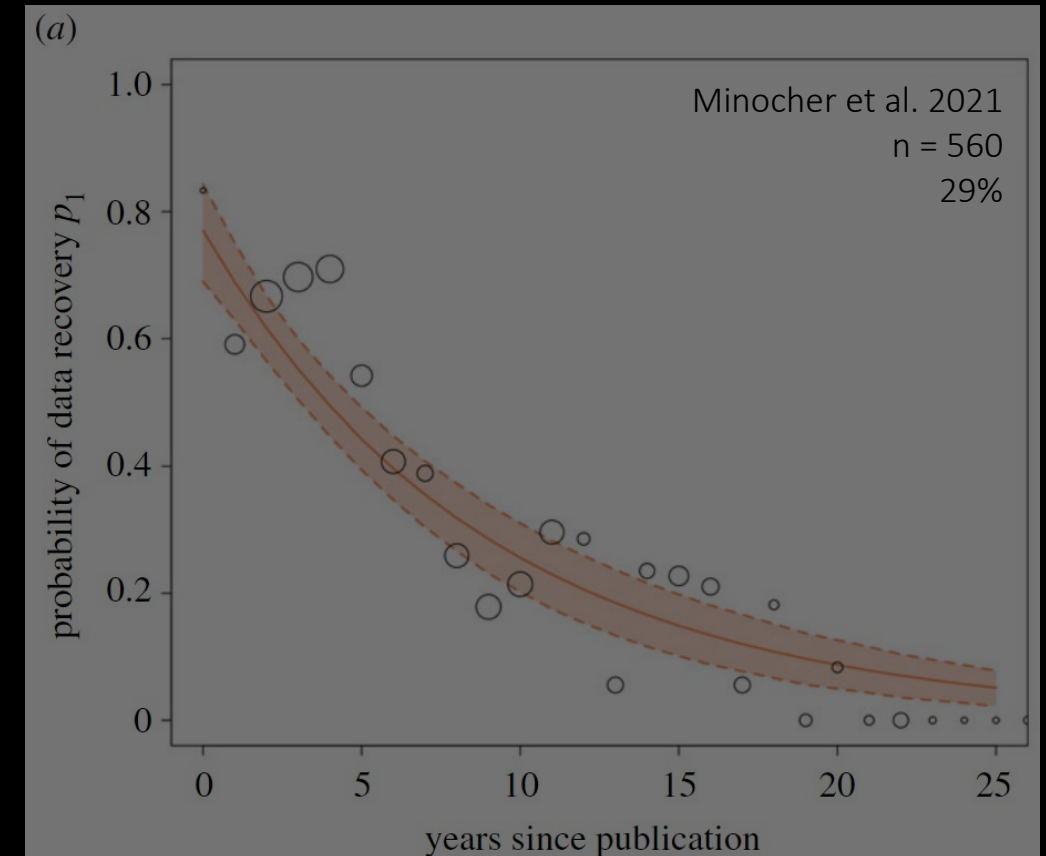
$P(\text{data reusable}) = 87\%$

Why? Data reusability

1955-2018



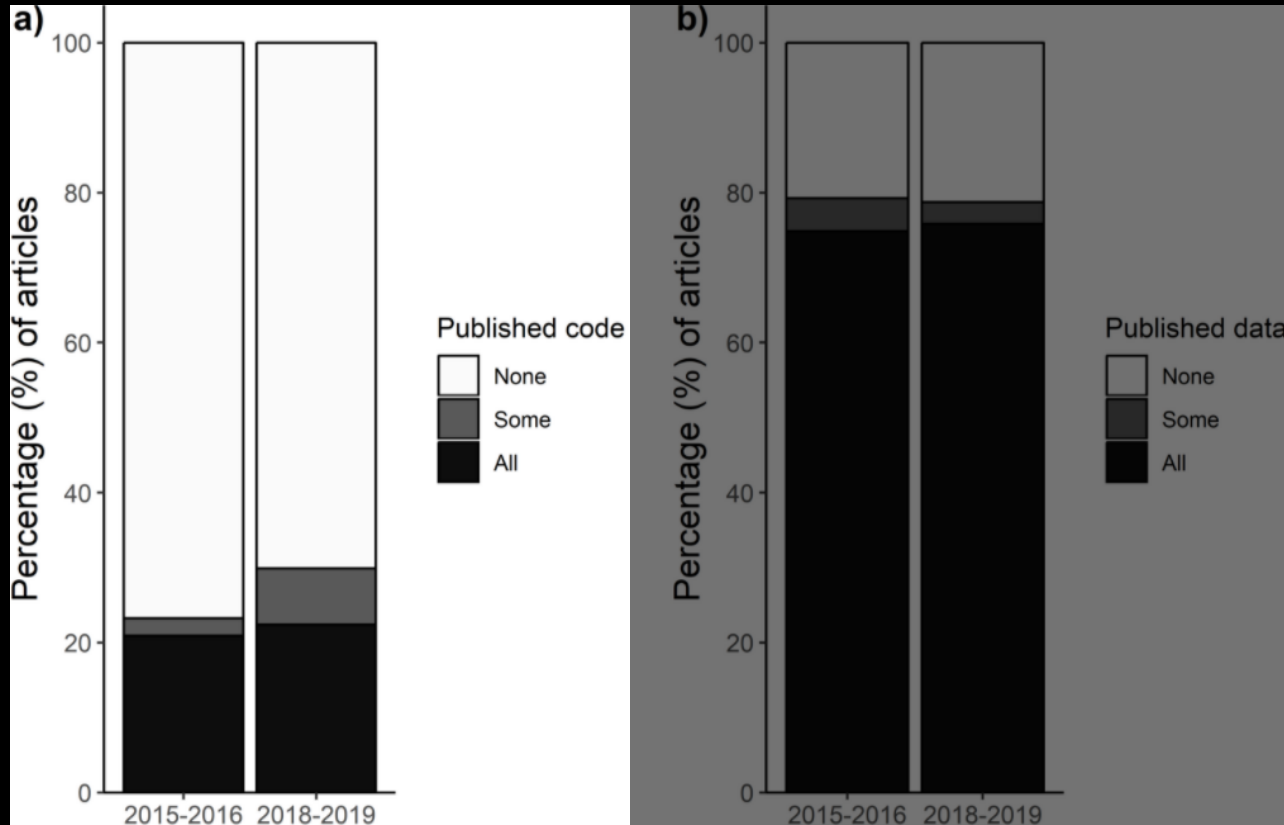
56% datasets are STILL incomplete



$P(\text{data reusable}) = 87\%$

Why? Code availability

Culina et al. 2020
2015-2019
n = 346



27% code-sharing

79% data-sharing

Why? Code usability

IOP Publishing

Environ. Res. Lett. 17 (2022) 041005

<https://doi.org/10.1088/1748-9326/ac5cf8>

ENVIRONMENTAL RESEARCH
LETTERS



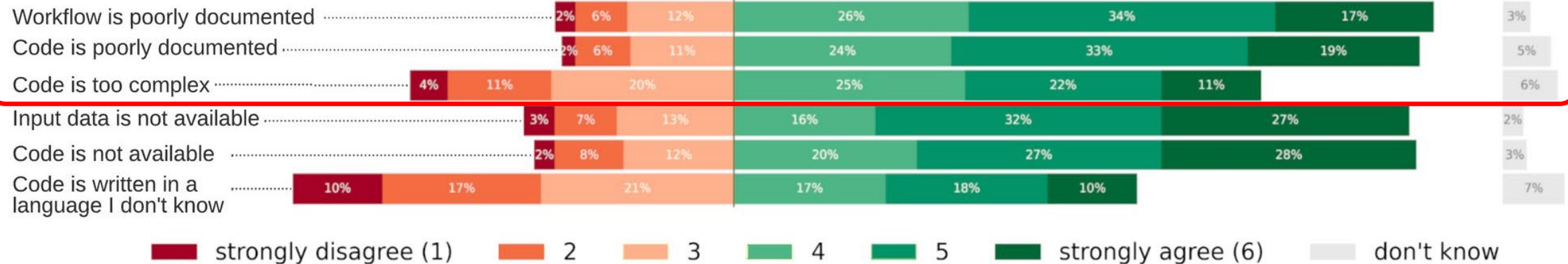
PERSPECTIVE

The critical need to foster computational reproducibility

OPEN ACCESS

Robert Reinecke^{1,2,*}, Tim Trautmann³, Thorsten Wagener¹ and Katja Schüler⁴

Reasons for a lack of reproducibility



N = 265 scientists earth sciences

Part II: The briefing



Reproducible Code

A guide to best practice

Laura Graham

19 May 2022

▶ ⏮ 🔊 1:40 / 1:00:25



To recapitulate:

1. Consistency:

- Choose your coding style (e.g. Google's R Style Guide, Tidyverse Style Guide, or whatever you like)
- BUT most importantly, KEEP IT!



upper CamelCase



snake_case



Alfredo Sánchez-Tójar
@ASanchez_Tojar

Conferences > 2009 IEEE 17th International ... ?

To camelcase or under_score

Publisher: IEEE

Cite This

PDF

Dave Binkley ; Marcia Davis ; Dawn Lawrie ; Christopher Morrell

<https://doi.org/10.1109/ICPC.2009.5090039>

Abstract:

Naming conventions are generally adopted in an effort to improve program comprehension. Two of the most popular conventions are alternatives for composing multi-word identifiers: the use of underscores and the use of camel casing. While most programmers have a personal opinion as to which style is better, empirical study forms a more appropriate basis for choosing between them. The central hypothesis considered herein is that identifier style affects the speed and accuracy of manipulating programs. An empirical study of 135 programmers and non-programmers was conducted to better understand the impact of identifier style on code readability. The experiment builds on past work of others who study how readers of natural language perform such tasks. Results indicate that camel casing leads to higher accuracy among all subjects regardless of training, and those trained in camel casing are able to recognize identifiers in the camel case style faster than identifiers in the underscore style.



@ASanchez_Tojar



Alfredo Sánchez-Tójar

@ASanchez_Tojar



What's your preferred option when naming variables and objects in datasets and/or scripts? Feel free to add your reasoning in the comments below. Thank you!

#RStats #programming #DataScience #OpenScience

Underline ("blind_data")

74.4%

Camel case ("BlindData")

23.1%

Other (add below please)

2.6%

156 votes · Final results

11:44 AM · Mar 27, 2022 · Twitter Web App

➤ Objects: snake_case



➤ Functions: lower camelCase

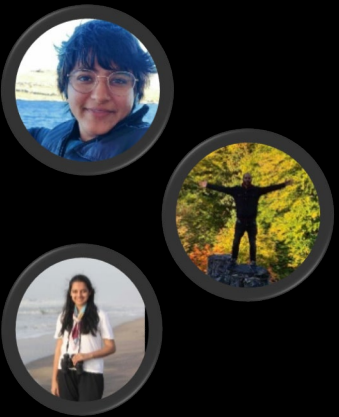


To recapitulate:

1. Consistency:

- Choose your coding style (e.g. Google's R Style Guide, Tidyverse Style Guide)
- BUT most importantly, KEEP IT!

2. Consistent naming: also for files, folders, scripts...



- ✓ code
- ✓ data
- ✓ figures
- ✓ manuscript
- ✓ pre-registration
- ✓ promotion
- ✓ protocol
- ✓ tables

- ✓ 001_title-and-abstract_screening
- ✓ 002_fulltext_screening
- ✓ 003_data_cleaning_and_standardization
- ✓ 004_plotting
- ✓ 005_supporting_information
- ✓ 006_twitter_promotion
- ✓ code_practices_in_ecology

To recapitulate:

1. Consistency:

- Choose your coding style (e.g. Google's R Style Guide, Tidyverse Style Guide)
- BUT most importantly, KEEP IT!

2. Consistent naming: also for files, folders, scripts...

3. Self-explanatory names

```

49 ▾ #####
50 # WEB OF SCIENCE: general search
51
52 # importing the .bib files
53 vos <- read_bibliography("literature_review/systematic_search/WoS_IGEs_search.bib")
54
55
56 # reducing fields to the minimum number of fields
57 # so that all databases have the same columns. Also, these fields
58 # are the important ones for the screening (see below).
59 reducing.fields.vos <- c("label","title","author","journal","issn","volume",
60                        "number","pages","year","publisher","doi","abstract") #number to issue,
61
62 vos.red <- vos[,reducing.fields.vos]
63
64
65 ▾ #####
66 # SCOPUS: general search
67
68 # importing the .bib files
69 scopus <- read_bibliography("literature_review/systematic_search/Scopus_IGEs_search.bib")
70
71 scopus$publisher <- "" #simply adding an empty column for the publisher since we did not extracted.
72
73
74 # reducing fields to the minimum number of fields
75 # so that all databases have the same columns. Also, these fields
76 # are the important ones for the screening (see below).
77 reducing.fields.scopus <- c("label","title","author","journal","issn","volume",
78                           "number","pages","year","publisher","doi","abstract") #number to issue,
79
80 scopus.red <- scopus[,reducing.fields.scopus]

```

```

49 ▾ #####
50 # WEB OF SCIENCE: general search
51
52 # importing the .bib files
53 wos <- read_bibliography("literature_review/systematic_search/WoS_IGEs_search.bib")
54
55
56 # reducing fields to the minimum number of fields
57 # so that all databases have the same columns. Also, these fields
58 # are the important ones for the screening (see below).
59 reducing.fields.wos <- c("label","title","author","journal","issn","volume",
60                          "number","pages","year","publisher","doi","abstract") #number to issue,
61
62 wos.red <- wos[,reducing.fields.wos]
63
64
65 ▾ #####
66 # SCOPUS: general search
67
68 # importing the .bib files
69 scopus <- read_bibliography("literature_review/systematic_search/Scopus_IGEs_search.bib")
70
71 scopus$publisher <- "" #simply adding an empty column for the publisher since we did not extracted.
72
73
74 # reducing fields to the minimum number of fields
75 # so that all databases have the same columns. Also, these fields
76 # are the important ones for the screening (see below).
77 reducing.fields.scopus <- c("label","title","author","journal","issn","volume",
78                             "number","pages","year","publisher","doi","abstract") #number to issue,
79
80 scopus.red <- scopus[,reducing.fields.scopus]

```

```

49 ▾ #####
50 # WEB OF SCIENCE: general search
51
52 # importing the .bib files
53 wos <- read_bibliography("literature_review/systematic_search/WoS_IGEs_search.bib")
54
55
56 # reducing fields to the minimum number of fields
57 # so that all databases have the same columns. Also, these fields
58 # are the important ones for the screening (see below).
59 reducing.fields.wos <- c("label","title","author","journal","issn","volume",
60                          "number","pages","year","publisher","doi","abstract") #number to issue,
61
62 wos.red <- wos[,reducing.fields.wos]
63
64
65 ▾ #####
66 # SCOPUS: general search
67
68 # importing the .bib files
69 scopus <- read_bibliography("literature_review/systematic_search/Scopus_IGEs_search.bib")
70
71 scopus$publisher <- "" #simply adding an empty column for the publisher since we did not extracted.
72
73
74 # reducing fields to the minimum number of fields
75 # so that all databases have the same columns. Also, these fields
76 # are the important ones for the screening (see below).
77 reducing.fields.scopus <- c("label","title","author","journal","issn","volume",
78                             "number","pages","year","publisher","doi","abstract") #number to issue,
79
80 scopus.red <- scopus[,reducing.fields.scopus]

```

To recapitulate:

1. Consistency:

- Choose your coding style (e.g. Google's R Style Guide, Tidyverse Style Guide)
- BUT most importantly, KEEP IT!

2. Consistent naming: also for files, folders, scripts...

3. Self-explanatory names

4. Redability: spacing and indentation

```

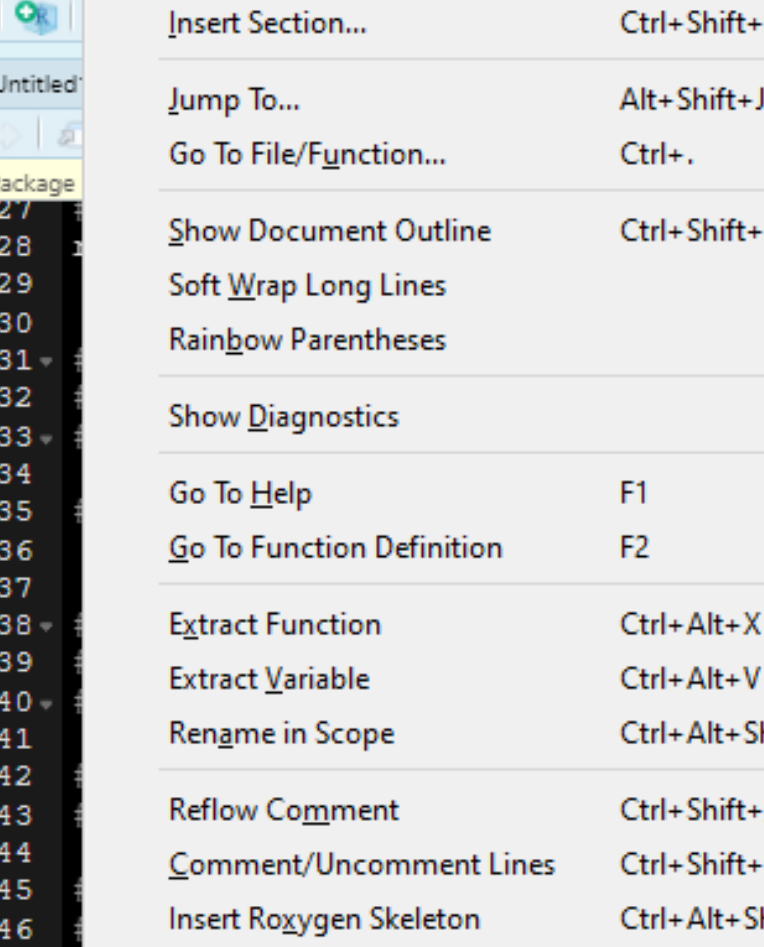
1 #####
2 # Authors:
3 # Alfredo Sanchez-Tojar (with support from Endika Blanco)
4
5 # Script first created in Nov 2018
6
7 #####
8 # Description of script and Instructions
9 #####
10
11 # This script is for a meta-analysis on the effect of urbanization on
12 # biodiversity.
13
14 #####
15 # Packages needed
16 #####
17
18 # Clear memory
19 rm(list=ls())
20
21 ##install.packages("pacman")
22 # devtools::install_github("itchyshin/orchard_plot", subdir = "orchard",
23 # force = TRUE, build_vignettes = TRUE)
24 # devtools::install_github("daniellnoble/orchard", force = TRUE)
25 pacman::p_load(devtools, tidyverse, metafor, patchwork, R.rsp, orchard, emmeans,
26 ape, phytools, report, car) #dplyr, ggplot2
27
28 #####
29 # Importing the data (from script 'data_preparation_urban.R')
30 #####
31
32 meta.final <- read.table("meta_final_reduced.csv",header=T, sep=",")
33
34 # modifying variables type
35 meta.final$studyID <- as.factor(meta.final$studyID)
36 meta.final$es_type <- as.factor(meta.final$es_type)
37
38 # creating a new variable to categorize effects as biserrial correlations,
39 # Pearson's correlation coefficients and effects from inferential statistics
40 meta.final$es_type_2 <- as.factor(ifelse(meta.final$es_type %in%
41 c("tgfu","tgri","tgsh","tgsi"),
42 "biserrial_correlation",
43 ifelse(meta.final$es_type %in%
44 c("pearsons","spearmans"),
45 "correlation_coefficient",
46 "inferential_statistic"))))

```

```

1 #####
2 # Authors:
3 # Alfredo Sanchez-Tojar (with support from Endika Blanco)
4
5 # Script first created in Nov 2018
6
7 #####
8 # Description of script and Instructions
9 #####
10
11 #####
12 # Packages needed
13 #####
14
15 # Importing the data (from script 'data_preparation_urban.R')
16 #####
17
18 # Sampling variance-covariance matrix
19 #####
20
21 # We specified sampling variance as variance-covariance matrix, with the
22 # sampling variance for each effect size on the diagonal, and the covariance
23 # between these measures as off-diagonal elements. The conservative model
24 # assumed a 0.5 correlation between the effect size sample variances with the
25 # same studyID. For a similar approach, see O'Dea et al., 2019
26 # (https://doi.org/10.1111/faf.12394) or Kim et al. 2022
27 # (https://doi.org/10.1111/1365-2656.13554), from which this code is taken from.
28
29 ### Creating a var-covar matrix assuming a 0.5 correlation between effect sizes
30 # from the same studyID covariance = (0.5 * sqrt(vi.1) * sqrt(vi.2))
31
32 # creates a matrix (called 'VCV_ESVar') with the dimensions
33 # n(effect_sizes) x n(effect_sizes)
34 VCV_ESVar <- matrix(0, nrow = nrow(meta.final), ncol = nrow(meta.final))
35
36 # names rows and columns for each obsID
37 rownames(VCV_ESVar) <- meta.final[, "obsID"]
38 colnames(VCV_ESVar) <- meta.final[, "obsID"]
39
40 # finds effect sizes that come from the same study
41 shared_coord <- which(meta.final[, "studyID"] %in%
42 meta.final[duplicated(meta.final[, "studyID"]),
43 "studyID"] == TRUE)
44
45 combinations <- do.call("rbind", tapply(shared_coord,
46 meta.final[shared_coord, "studyID"],
47 function(x) t(utils::combn(x, 2))))

```



The screenshot shows the RStudio interface with the 'Code' menu open. The menu items and their shortcuts are as follows:

Action	Shortcut
Insert Section...	Ctrl+Shift+R
Jump To...	Alt+Shift+J
Go To File/Function...	Ctrl+.
Show Document Outline	Ctrl+Shift+O
Soft Wrap Long Lines	
Rainbow Parentheses	
Show Diagnostics	
Go To Help	F1
Go To Function Definition	F2
Extract Function	Ctrl+Alt+X
Extract Variable	Ctrl+Alt+V
Rename in Scope	Ctrl+Alt+Shift+M
Reflow Comment	Ctrl+Shift+ /
Comment/Uncomment Lines	Ctrl+Shift+C
Insert Roxygen Skeleton	Ctrl+Alt+Shift+R
Reindent Lines	Ctrl+I
Reformat Code	Ctrl+Shift+A



@ASanchez_Tojar



To recapitulate:

1. Consistency:

- Choose your coding style (e.g. Google's R Style Guide, Tidyverse Style Guide)
- BUT most importantly, KEEP IT!

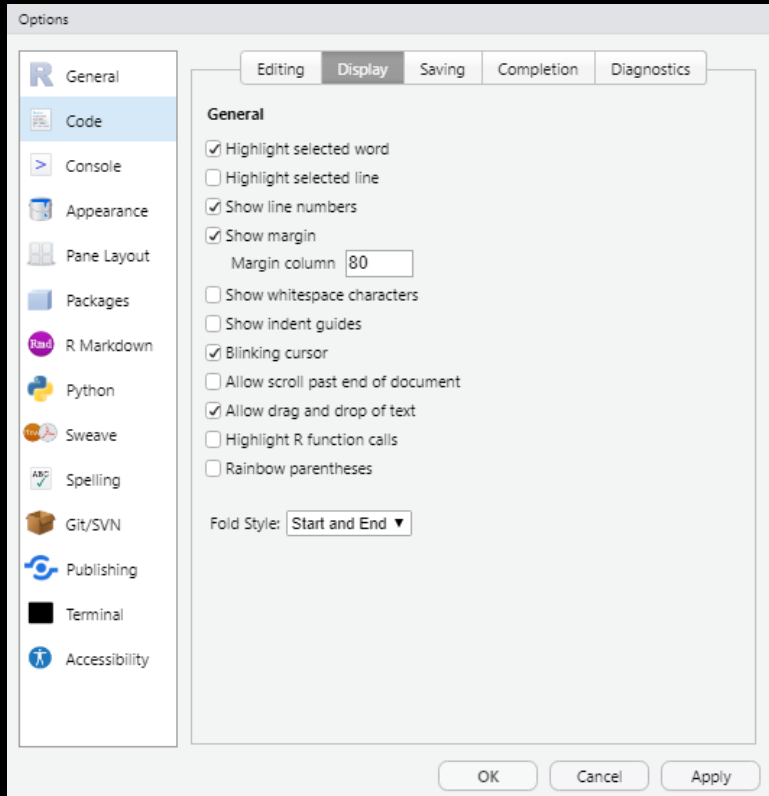
2. Consistent naming: also for files, folders, scripts...

3. Self-explanatory names

4. Redability: spacing and indentation

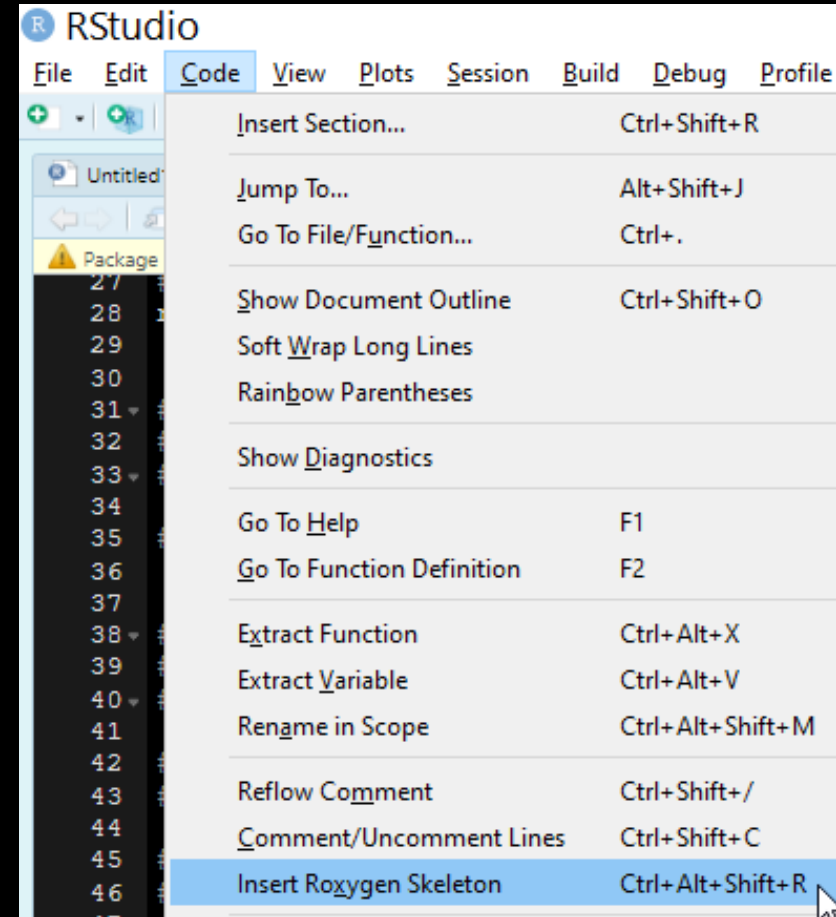
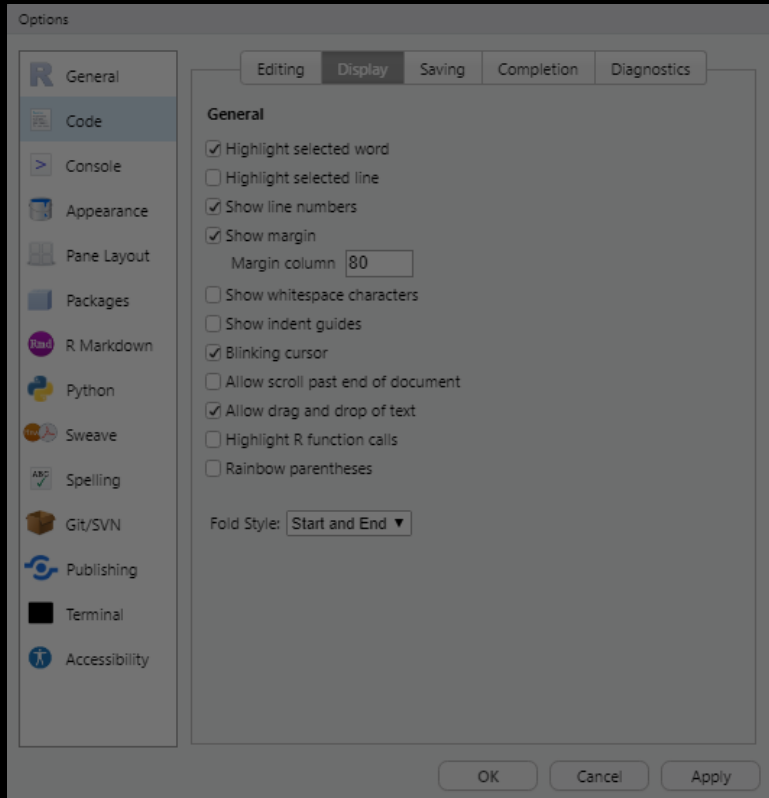
5. Commenting: *why* rather than *what*, although...

Bonus tricks



```
1 #####
2 # Authors:
3 # Alfredo Sanchez-Tojar (with support from Endika Blanco)
4
5 # Script first created in Nov 2018
6
7 #####
8 # Description of script and Instructions
9 #####
14 #####
15 Packages needed
16 #####
28 #####
29 # Importing the data (from script 'data_preparation_urban.R')
30 #####
133 #####
134 # Sampling variance-covariance matrix
135 #####
136
137 # We specified sampling variance as variance-covariance matrix, with the
138 # sampling variance for each effect size on the diagonal, and the covariance
139 # between these measures as off-diagonal elements. The conservative model
140 # assumed a 0.5 correlation between the effect size sample variances with the
141 # same studyID. For a similar approach, see O'Dea et al., 2019
142 # (https://doi.org/10.1111/faf.12394) or Kim et al. 2022
143 # (https://doi.org/10.1111/1365-2656.13554), from which this code is taken from.
144
145 ### Creating a var-covar matrix assuming a 0.5 correlation between effect sizes
146 # from the same studyID covariance = (0.5 * sqrt(vi.1) * sqrt(vi.2))
147
148 # creates a matrix (called 'VCV_ESVar') with the dimensions
149 # n(effect_sizes) x n(effect_sizes)
150 VCV_ESVar <- matrix(0, nrow = nrow(meta.final), ncol = nrow(meta.final))
151
152 # names rows and columns for each obsID
153 rownames(VCV_ESVar) <- meta.final[, "obsID"]
154 colnames(VCV_ESVar) <- meta.final[, "obsID"]
155
156 # finds effect sizes that come from the same study
157 shared_coord <- which(meta.final[, "studyID"] %in%
158                       meta.final[duplicated(meta.final[, "studyID"]),
159                                   "studyID"] == TRUE)
160
161 combinations <- do.call("rbind", tapply(shared_coord,
162                                         meta.final[shared_coord, "studyID"],
163                                         function(x) t(utils::combn(x, 2))))
```

Bonus tricks



Part III:

The bad example



Journal of Avian Biology 48: 407–416, 2017

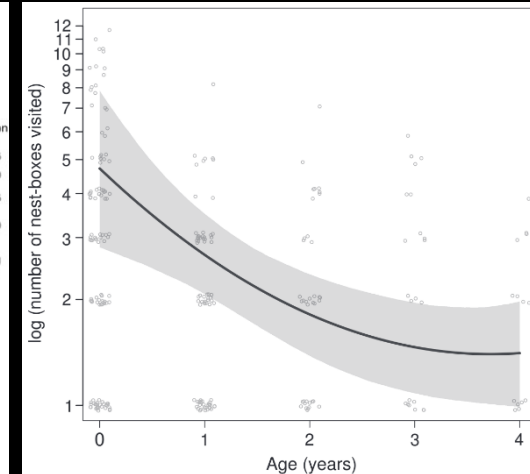
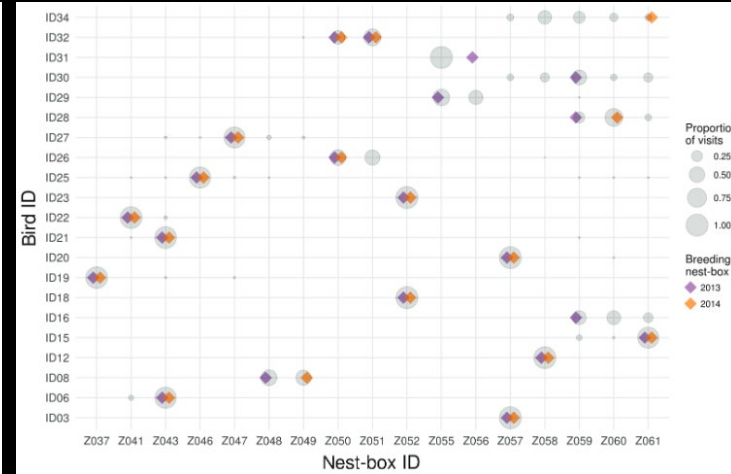
doi: 10.1111/jav.01055

© 2016 The Authors. This is an Online Open article

Subject Editor: Ronald Ydenberg. Editor-in-Chief: Jan-Åke Nilsson. Accepted 11 July 2016

Winter territory prospecting is associated with life-history stage but not activity in a passerine

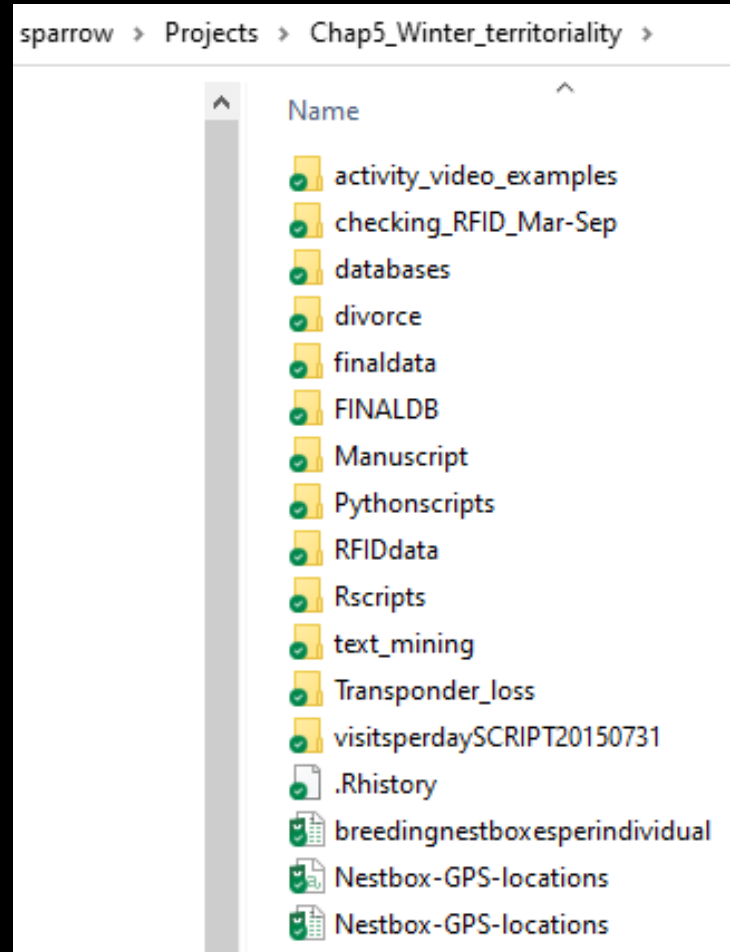
Alfredo Sánchez-Tójar, Isabel Winney, Antje Girndt, Mirre J. P. Simons, Shinichi Nakagawa, Terry Burke and Julia Schroeder





Winter territory prospecting is associated with life-history stage but not activity in a passerine

Alfredo Sánchez-Tójar, Isabel Winney, Antje Girdt, Mirre J. P. Simons, Shinichi Nakagawa, Terry Burke and Julia Schroeder



sparrow > Projects > Chap5_Winter_territoriality > finaldata





Winter territory prospecting is associated with life-history stage but not activity in a passerine

Alfredo Sánchez-Tójar, Isabel Winney, Anlje Girndt, Mirre J. P. Simons, Shinichi Nakagawa, Terry Burke and Julia Schroeder

```
1 #visitspernestbox script
2 #Last update: 8th.July.2015
3
4 rm(list=ls())
5
6 getwd()
7 setwd("/home/alfredo/Desktop/visitsperdaySCRIPT20150703")
8 getwd()
9
10 #visitspernestbox<-read.table("003FYandWSnestboxesperBirdperdayonepernestboxperday-playing-numdate.csv",header=TRUE,sep='\t')
11
12 #data on winter 2011 nest box use, subset containing only birds that we know
13 #that kept the transponder during the whole winter
14 visitspernestboxRFID<-read.table("003nestboxesperBirdperdayonepernestboxperday-numdate-accountingforpittaglost.csv",header=TRUE,sep='\t')
15
16 visitspernestboxRFID$BirdID<-as.factor(visitspernestboxRFID$BirdID)
17 head(visitspernestboxRFID)
18 summary(visitspernestboxRFID)
19
20 #data on winter 2011 nest box use, subset containing only birds that we know
21 #that kept the transponder during the whole winter and only birds that breed in that
22 #nestbox
23 visitspernestboxRFID.subset<-read.table("003nestboxesperBirdperdayonepernestboxperday-numdate-accountingforpittaglost-onlybreeder-sexandyearofbreedingincluded.csv",header=TRUE,sep='\t')
24 visitspernestboxRFID.subset<-read.table("003nestboxesperBirdperdayonepernestboxperday-numdate-accountingforpittaglost-onlybreeder-sexandyearandnestboxofbreedingincluded.csv",header=TRUE,sep='\t')
25
26 visitspernestboxRFID.subset$BirdID<-as.factor(visitspernestboxRFID.subset$BirdID)
27 visitspernestboxRFID.subset$sex<-as.factor(visitspernestboxRFID.subset$sex)
28 visitspernestboxRFID.subset$year<-as.factor(visitspernestboxRFID.subset$year)
29 head(visitspernestboxRFID.subset)
30 summary(visitspernestboxRFID.subset)
31
32 #from 2011 to 2015 (1st of July) data on birds breeding in each of the
33 #RFID equipped nestboxes
34 visitspernestbox<-read.table("RFIDnestboxesandbreedingpairs2011-2015-TOTAL-separatedbyindividual-v1-20150701.csv",header=TRUE,sep=',')
35
36 visitspernestbox$BirdID<-as.factor(visitspernestbox$BirdID)
37 visitspernestbox$sex<-as.factor(visitspernestbox$sex)
38 visitspernestbox$year<-as.factor(visitspernestbox$year)
```



```

1 # #####
2 # Authors:
3 # Alfredo Sanchez-Tojar (with support from Endika Blanco)
4
5 # Script first created in Nov 2018
6
7 #####
8 # Description of script and Instructions
9 #####
10
11 # This script is for a meta-analysis on the effect of urbanization on
12 # biodiversity.
13
14 #####
15 # Packages needed
16 #####
17
18 # Clear memory
19 rm(list=ls())
20
21 ##install.packages("pacman")
22 # devtools::install_github("itchyshin/orchard_plot", subdir = "orchardR",
23 # force = TRUE, build_vignettes = TRUE)
24 # devtools::install_github("daniellnoble/orchardR", force = TRUE)
25 pacman::p_load(devtools, tidyverse, metafor, patchwork, R.rsp, orchardR, emmeans,
26 ape, phytools, report, car) #dplyr, ggplot2
27
28 #####
29 # Importing the data (from script 'data_preparation_urban.R')
30 #####
31
32 meta.final <- read.table("meta_final_reduced.csv",header=T, sep=",")
33
34 # modifying variables type
35 meta.final$studyID <- as.factor(meta.final$studyID)
36 meta.final$es_type <- as.factor(meta.final$es_type)
37
38 # creating a new variable to categorize effects as biserrial correlations,
39 # Pearson's correlation coefficients and effects from inferential statistics
40 meta.final$es_type_2 <- as.factor(ifelse(meta.final$es_type %in%
41 c("tgfu","tgri","tgsh","tgsi"),
42 "biserrial_correlation",
43 ifelse(meta.final$es_type %in%
44 c("pearsons","spearmans"),
45 "correlation_coefficient",
46 "inferential_statistic"))))

```

```

1 #####
2 # Authors:
3 # Alfredo Sanchez-Tojar (with support from Endika Blanco)
4
5 # Script first created in Nov 2018
6
7 #####
8 # Description of script and Instructions
9 #####
10
11 #####
12 # Packages needed
13 #####
14
15 # Importing the data (from script 'data_preparation_urban.R')
16 #####
17
18 # Sampling variance-covariance matrix
19 #####
20
21 # We specified sampling variance as variance-covariance matrix, with the
22 # sampling variance for each effect size on the diagonal, and the covariance
23 # between these measures as off-diagonal elements. The conservative model
24 # assumed a 0.5 correlation between the effect size sample variances with the
25 # same studyID. For a similar approach, see O'Dea et al., 2019
26 # (https://doi.org/10.1111/faf.12394) or Kim et al. 2022
27 # (https://doi.org/10.1111/1365-2656.13554), from which this code is taken from.
28
29 ### Creating a var-covar matrix assuming a 0.5 correlation between effect sizes
30 # from the same studyID covariance = (0.5 * sqrt(vi.1) * sqrt(vi.2))
31
32 # creates a matrix (called 'VCV_ESVar') with the dimensions
33 # n(effect_sizes) x n(effect_sizes)
34 VCV_ESVar <- matrix(0, nrow = nrow(meta.final), ncol = nrow(meta.final))
35
36 # names rows and columns for each obsID
37 rownames(VCV_ESVar) <- meta.final[, "obsID"]
38 colnames(VCV_ESVar) <- meta.final[, "obsID"]
39
40 # finds effect sizes that come from the same study
41 shared_coord <- which(meta.final[, "studyID"] %in%
42 meta.final[duplicated(meta.final[, "studyID"]),
43 "studyID"] == TRUE)
44
45 combinations <- do.call("rbind", tapply(shared_coord,
46 meta.final[shared_coord, "studyID"],
47 function(x) t(utils::combn(x, 2))))

```

DOI 10.5281/zenodo.3549539

Meta-analysis of adaptive "transgenerational" effects: a commentary on Yin et al. (2019), Ecology Letters.

This project contains the materials for a project testing the generality and robustness of adaptive "transgenerational" effects across species, environments... It re-analyzes a published meta-analytic data set to show that adaptive "transgenerational" effects - more specifically, anticipatory parental effects, despite existing in some species, are not as widespread as generally stated.

This repository contains the data and Rmarkdown code used in the following study:

Alfredo Sánchez-Tójar, Malgorzata Lagisz, Nicholas P. Moran, Shinichi Nakagawa, Daniel W. A. Noble, Klaus Reinhold. 2020. *The jury is still out regarding the generality of adaptive "transgenerational" effects*. Ecology Letters, 23(11):1715-1718. DOI: [10.1111/ele.13479](https://doi.org/10.1111/ele.13479)

More information and materials available at the [OSF project](#). For any further information, please contact: [Alfredo Sánchez-Tójar](#), email: alfredo.tojar@gmail.com

Folders:

code: contains an Rmarkdown file (`supplementary_information.biblio.rmd`) that corresponds with the supplementary information file. This Rmarkdown file contains all the code used in this project, except some functions, which are available in the R script `functions.R` and imported from within the Rmarkdown file. In addition, the file `Sanchez-Tojar_et_al_commentary.bib` corresponds to all the references used in the study, both the main text and the supplementary information. The folder **docs** contains the .html version of the supplementary information.

data: contains the original data set published by Yin et al. (2019) (i.e. `Yin et al raw data.csv`) together with the final data set used in our study (i.e. `transgenerational.csv`). Additionally, the remaining files are related to the phylogenetic tree and relationships among the species included in the models. For more information, see the



- 1 - Systematic search
- 2 - Data preparation
- 3 - Non-independence
- 3.1 - Phylogenetic relationships**
- 3.2 - Shared control
- 4 - Calculating effect sizes
- 5 - Analyses
- R Session Information
- References

3 - Non-independence

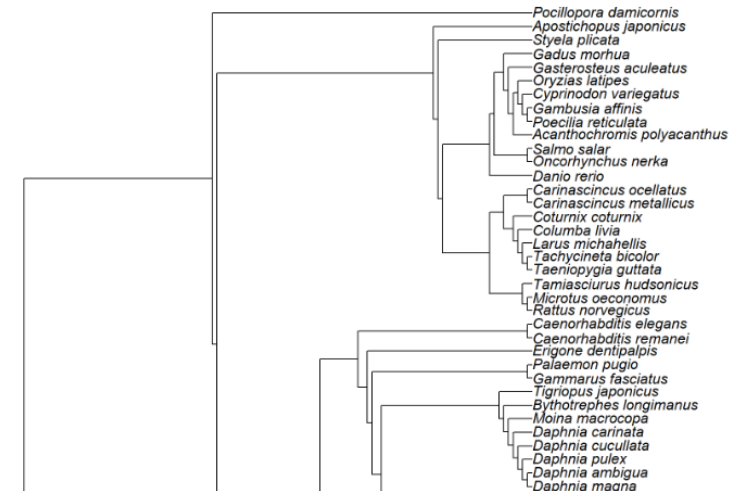
3.1 - Phylogenetic relationships

Using the corrected species information (see section 2.8), we built a phylogeny by retrieving the phylogenetic relationships from the Open Tree of Life (Hinchliff et al. 2016) using the R package `rot1` v.3.0.5 (Michonneau, Brown, and Winter 2016). We estimated branch lengths following Grafen (1989) as implemented in the function `compute.br1en()` of the R package `ape` v.5.2 (Paradis and Schliep 2019). We dealt with polytomies using randomization as implemented in the function `multizdi()` from the R package `ape` v.5.2 (Paradis and Schliep 2019). We then constructed a phylogenetic relatedness correlation matrix for the statistical models (see below).

Code

Hide

```
# we can then plot the tree, but notice that the species names shown in the tree are not exactly the same as the
# species names that we had in our List. This is because those names had synonyms in the tree of life databas
# e, and we are using those names for the plot. This tree uses the data once polytomies have been dealt with.
plot(tree_random, cex=1, label.offset = .1, no.margin = TRUE)
```



Part IV:

The commercial break

Why?



Hackathon: #SORTEE2022



Antica Culina



Simon Evans

2022 SORTEE CONFERENCE NEWS

Registrations now open!

<https://www.sortee.org/upcoming/>

[READ MORE](#)



@sortecoevo

- 11-13th July
- Worldwide (online)
- 30-60 USD (10 or 0!!!)



@ASanchez_Tojar

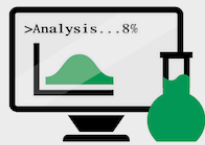
Time to neaten up some CODE!!!

Extras:

> WHY OPEN CODE IN RESEARCH?



RESEARCHERS STUDENTS



- **Learning** new computational methods
- **Sharing** verified resources
- **Collaborating** efficiently
- **Reproducing** statistical analyses
- **Improving employability**

FUNDERS



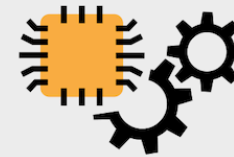
- **Reducing costs**
- **Improving research efficiency**
- **Increasing transparency and integrity**
- **Facilitating project evaluations**
- **Showcasing** best projects

JOURNALS PUBLISHERS



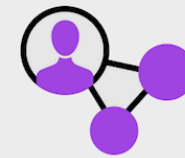
- **Increasing impact** of papers
- **Improving journal's metrics/ranking**
- **Improving peer-review** process
- **Ensuring integrity**
- **Becoming role models** in open science

PRACTITIONERS AGENCIES



- **Facilitating industry applications**
- **Encouraging collaborations**
- **Leading change** to best practice
- **Supporting R&D** information transfer
- **Repurposing** code

GENERAL PUBLIC



- **Building trust** in science
- **Encouraging public access**
- **Delivering robust, science-based solutions**
- **Promoting public computational literacy**
- **Reusing code** in citizen-science

© by Society for Open Reliable and Transparent Ecology and Evolutionary Biology sortee.org 2021 CC-BY-SA

COMMENT

Open Access

Five selfish reasons to work reproducibly



Florian Markowetz

1. Avoid disaster;
2. Writing papers;
3. Peer-review;
4. Continuity;
5. Reputation

Dr Florian Markowetz | Five selfish reasons to working reproducibly | RIOTS Exeter



5 selfish reasons to work reproducibly

Florian Markowetz
CRUK Cambridge Institute
www.markowetzlab.org
[@markowetzlab](https://twitter.com/markowetzlab)