Statistical Ecology

Today's agenda:
Introductions,
Syllabus,
Why we need stats,
Reproducibility and Openness

Introductions

Tell us:

- Your name
- Your research interests
- Something interesting about yourself

Syllabus: basic stuff

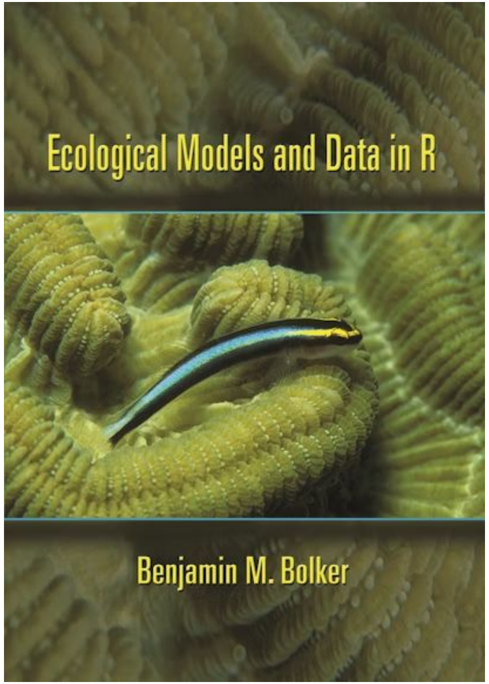
Class Meeting Days: T, Th
Class Meeting Time: 2:00 – 3:15 pm
Class Meeting Location: DAV 266
Instructor: Dr. Brian Maitner
Office Location: DAV 226 (but I'm usually in URL 106, because it's warmer)
Office Hours: TBD - ?
Email: bmaitner@usf.edu

Syllabus: Course structure

Flipped course: read before class, work during class

Assignments will focus on data of your choosing

Graded Items	Percent of Final Grade
In-class participation	20%
In-class quizzes	20%
Assignments (4x)	20%
Midterm (take-home)	20%
Final (presentation + take-home)	20%



Digital copies available through the library for free

Course Expectations

- Participate! Ask questions. Share ideas, experiences, and knowledge!
- Be respectful.

Recommendations

- Read the syllabus
- Read the book
- Take notes on paper
- Plan ahead
- Bring a laptop

Questions so far?

What will you get out of this class?

In general:

- Ability to do common analyses and visualizations
- Ability to go further on your own

What will you get out of this class?

More specifically:

Student Learning Outcomes

- Load data into R and conduct common data-wrangling tasks
- Create data visualizations using base R and ggplot2
- List the common types of statistical distribution and provide examples of when they might apply.
- Choose appropriate analyses for given ecological questions and datasets.
- Demonstrate an ability to troubleshoot and de-bug R code.
- Explain how they would approach a novel coding problem
- Apply the skills learned to their own work

Learning Stats

People learn differently and stats can be non-intuitive

- The book will expose you to verbal and mathematical descriptions of things
- In class we'll use coding and visualizations to try to understand things
- The hope is that at least one of these methods will appeal to everyone

Learning Stats

Don’t focus on memorization

DO focus on practice and understanding

A note on AI

- We’ll cover AI later in the semester
- Strongly recommend that you start out without it
- You need reasonable fundamentals to be able to effectively use AI

Questions so far?

Why do we need statistics?

Why do we need statistics?

- Human minds are imperfect (biases, fallacies, etc.)
- Quantifying effect sizes
- Hypothesis testing
- Making predictions
- Visualizations to make relationships more clear
- Lots of other stuff

What is R and why is it useful?

R is a scripted coding language (type commands, rather than click buttons)

Scripted means that:

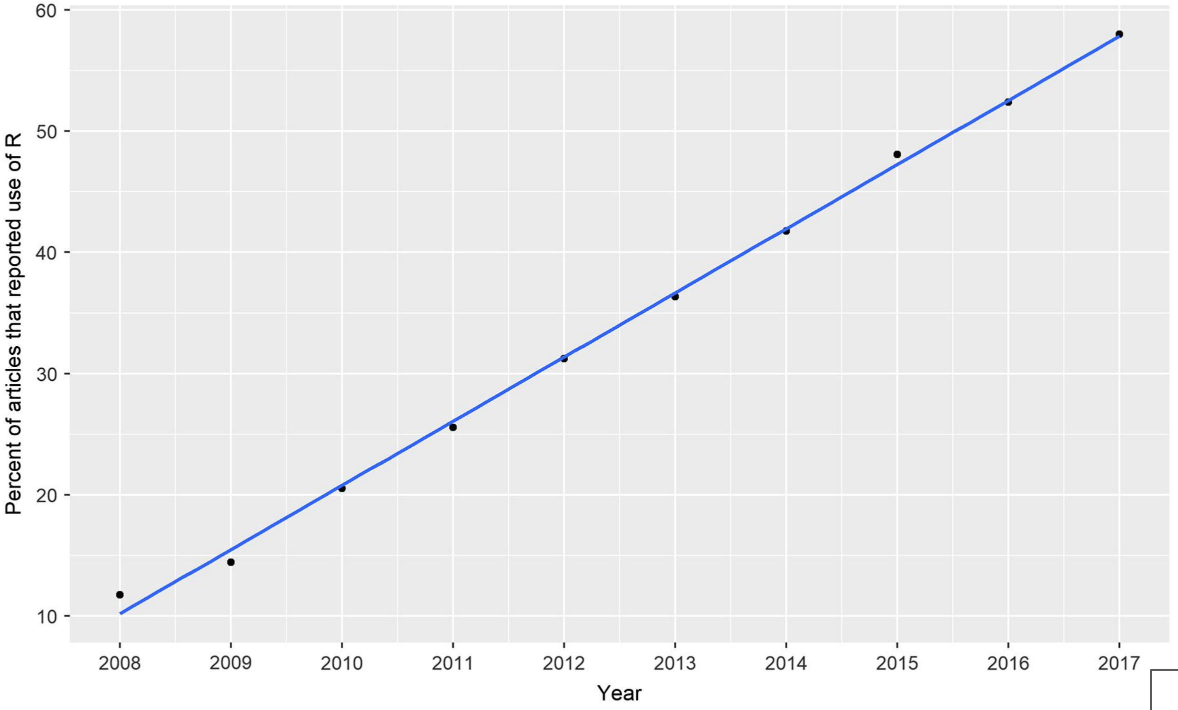
- You can easily re-run or revise analyses
- Code and be re-used for similar projects
- Your code is basically a methods section

What is R and why is it useful?

R is free!

- Users contribute packages (also for free)
- This leads to lots of flexibility in what it can do
- Popular, so lots of places to look for help

What is R and why is it useful?

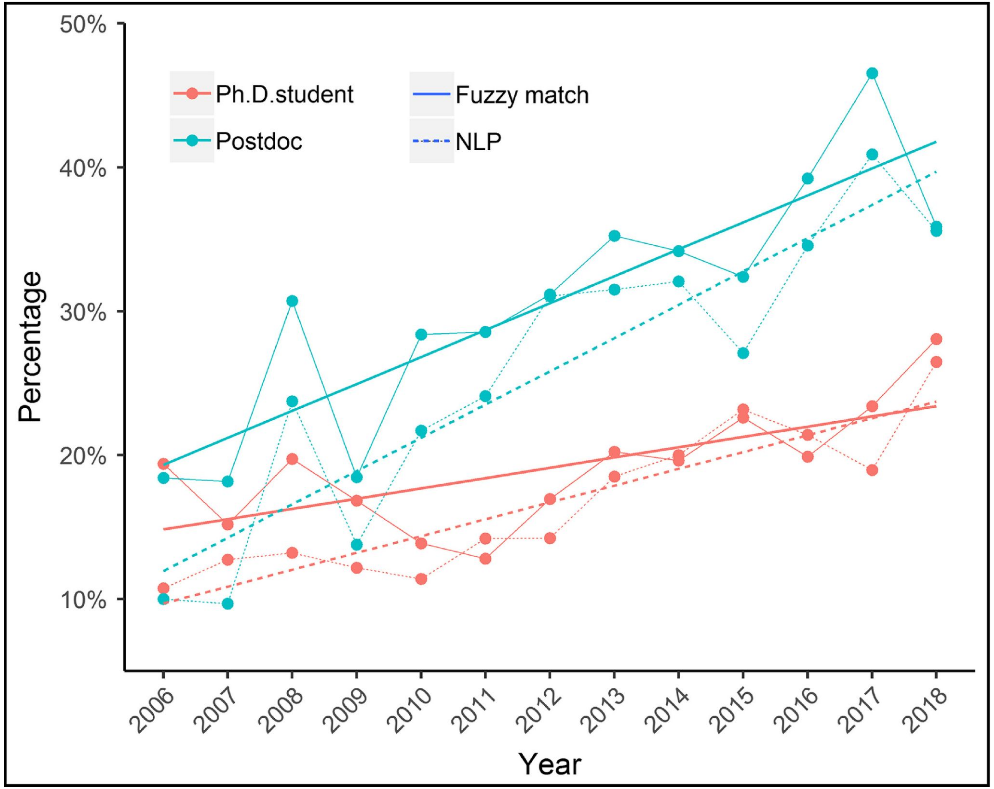


esa ECOSPHERE

Evaluating the popularity of R in ecology

Jiangshan Lai,[1,2,†] Christopher J. Lortie,[3,4] Robert A. Muenchen,[5] Jian Yang,[6] and Keping Ma[1]

What is R and why is it useful?



Front Ecol Environ 2020; 18(3): 123–124

Transparency and Reproducibility

Science needs to be transparent and reproducible:

Transparency - What was done is clear

Reproducibility - can be re-done with the same results

R helps with both of these!

Transparency and Reproducibility

Received: 14 November 2023 | Revised: 3 July 2024 | Accepted: 5 July 2024
DOI: 10.1002/ece3.70030

ACADEMIC PRACTICE IN ECOLOGY AND EVOLUTION

Ecology and Evolution Open Access WILEY

Code sharing in ecology and evolution increases citation rates but remains uncommon

Brian Maitner[1,2] | Paul Efren Santos Andrade[3] | Luna Lei[4] | Jamie Kass[5] | Hannah L. Owens[6,7,8] | George C. G. Barbosa[9] | Brad Boyle[10] | Matiss Castorena[10] | Brian J. Enquist[10,11] | Xiao Feng[12] | Daniel S. Park[13,14] | Andrea Paz[15,16] | Gonzalo Pinilla-Buitrago[17] | Cory Merow[18] | Adam Wilson[2]

2 MATERIALS AND METHODS

2.1 Data collection

2.1.1 List of ecology and evolution publications citing R

To generate a list of papers in ecology and evolution that likely made use of the R programming language (R Core Team, 2023), we performed a query on the Scopus database (https://www.scopus.com) using the *rscopus* R package (Muschelli, 2019). We searched Scopus (performed August 19, 2022) for peer reviewed journal articles that: (1) included the words "ecology" or "evolution" in an "all fields" search (which searches article titles, keywords, abstracts, and journal titles); (2) were published in journals within the subject area "agriculture and biological sciences"; (3) were published after January 1, 2010; (4) were written in English (as this is currently the dominant language of publication in ecology and evolution; Mauranen et al., 2010); and (5) included a citation of R in their reference list.

2.1.2 Checking for code and data availability

We manually evaluated a randomly chosen subset of the publications on our overall list. We selected a total of 1001 papers, evenly distributed across the time period (77 per year * 13 years). Papers that cited R but did not use it (or were unclear on whether they used it; $n = 3$) were discarded and replaced by a randomly selected paper from the same year. For each publication in this subset, we manually identified whether the publication shared any R code, either as supplementary information, or via a link (e.g., to a Github repository). For each paper, we (i) checked for the presence of code in supplemental material, (ii) skimmed publications for code and data availability statements, (iii) searched through publications for terms associated with code (i.e., "code", "supplement", "appendix", "R", "script" "Github"), and (iv) searched publications for URLs. Papers were scored with a binary variable indicating whether they shared R code or not. We did not distinguish between publications which shared sufficient code for reproduction and those which did not. We also did not attempt to rerun the code or assess its reproducibility, and only recorded the presence of any code, even if it was incomplete. Where code was included, we recorded the license the code was provider under, or lack thereof. We also assessed whether publications were open access and whether they shared open data in order to understand the importance of open code relative to these other open-access components. Open access information was provided by the *rscopus* R package (Muschelli, 2019). Open data was scored as a binary variable indicating whether the authors shared the full set of raw data underlying the analyses or not. To control for differences in citation rates among journals, we downloaded impact factor information using the *scholar* R package (Keirstead, 2016) on June 16, 2023. To estimate the proportion of publications which use R but do not properly cite it, we screened 130 randomly selected publications evenly distributed across the time period. These publications were selected using identical criteria to the publications that did cite R, except that they did not include R in their list of references.

2.2 Checking for code citations

Where code was shared in a citable location such as a DOI or URL ($n = 33$), we assessed whether the code itself was cited by querying the Scopus database for the URL (and DOI, where appropriate) using the *rscopus* R package (Muschelli, 2019). Publications where code was shared in appendices or supplementary information ($n = 22$) were excluded, as there was no way of distinguishing citations of the code with citations of the publication itself.

2.3 Analyses

All analyses were conducted in R version 4.3.0 (R Core Team, 2023). All R scripts underlying these analyses are available at: https://github.com/bmaitner/R_citations and via Zenodo (Maitner & Lei, 2024). For data processing, we used the R packages *stringdist* version 0.9.12 (van der Loo, 2014), *tidyverse* version 2.0.0 (Wickham et al., 2019), *googledrive* version 2.1.1 (D'Agostino McGowan & Bryan, 2023), and *googlesheets4* version 1.1.1 (Bryan, 2023); for analyses, the packages *bbmle* version 1.0.25.1 (Bolker & R Development Core Team, 2022), *DHARMa* version 0.4.6 (Hartig, 2022), *MuMIn* version 1.47.5 (Barton, 2023), *rscopus* version 0.6.6 (Muschelli, 2019), *rsq* version 2.6 (Zhang, 2018), *scholar* version 0.2.4 (Keirstead, 2016), and *stats* version 4.3.0 (R Core Team, 2023); for plotting, the packages *ggplot2* version 3.4.4 (Wickham, 2016), *ggpmisc* version 0.5.5 (Aphalo, 2022), and *questionr* version 0.7.8 to identify all R packages used.

2.3.1 Proportion of papers sharing code over time

We tested for a trend in code-sharing over time by modeling code sharing (binary, yes/no) as a function of the year (relative to 2010) using a generalized linear model. Modeling was performed using the function *glm* in the *stats* R package (R Core Team, 2023) with a binomial error distribution. We similarly tested for temporal trends in two other open-science components, open-access publication (binary) and open data (binary). We also tested whether open-access or open-data papers were disproportionately likely to share code using chi-square tests via the *chisq.test* function in the *stats* R package (R Core Team, 2023).

2.3.2 Impact of code sharing on citations

We additionally modeled the relationship between code sharing and citation count using generalized linear models in R. We modeled the dependent variable (cumulative number of citations of each article by 2022) using a Poisson distribution, which models the number of independent events occurring within a period of time (Bolker, 2008). In addition to the predictor variable for code sharing (binary, yes/no), we included other variables that were hypothesized to influence citation count. Data sharing (binary, yes/no) may increase citation counts as readers may cite papers as data sources (Christensen et al., 2019; Piwowar et al., 2007). Open access (binary, yes/no) may also increase citation counts by reaching a broader set of readers (Tang et al., 2017). Publications accumulate citations over time, and so citation count should increase with publication age (continuous, 1–13 years). Finally, publications in higher impact journals may be more likely to be read and cited, and hence, journal impact factor (continuous, 0–11.633) may be positively associated with citation count. In addition to main effects, we considered two classes of interactions: (1) interactions between publication age and other main effects, which are appropriate if a main effect modifies the rate at which a publication accumulates citations over time; and (2) interactions between open-science criteria (i.e., open access, open code, and open data), which are appropriate if there are synergistic effects of meeting multiple open-access criteria. We compared 11 models (including one null model) that differed in complexity and that represented different hypotheses regarding the factors that influence citations (Table 1). Continuous variables were scaled and centered. Overall model pseudo-R^2 for the best-performing model was calculated using the function *r.squaredGLMM* in the *rsq* package (Zhang, 2018).

TABLE 1. Candidate models of citation count.

ID	Models	df	ΔAIC
1	Citations ~ Impact factor × Age + R code shared × Age + Open access × Age + Data shared × Age + Data shared × R code shared + R code shared × Open Access + Open access × Data shared	13	0.0
2	Citations ~ Impact factor × Age + R code shared × Age + Data shared × Age + Data shared × R code shared	9	834.0

Transparency and Reproducibility

Received: 14 November 2023 | Revised: 3 July 2024 | Accepted: 5 July 2024
DOI: 10.1002/ece3.70030

ACADEMIC PRACTICE IN ECOLOGY AND EVOLUTION

Ecology and Evolution Open Access WILEY

Code sharing in ecology and evolution increases citation rates but remains uncommon

Brian Maitner[1,2] | Paul Efren Santos Andrade[3] | Luna Lei[4] | Jamie Kass[5] | Hannah L. Owens[6,7,8] | George C. G. Barbosa[9] | Brad Boyle[10] | Matiss Castorena[10] | Brian J. Enquist[10,11] | Xiao Feng[12] | Daniel S. Park[13,14] | Andrea Paz[15,16] | Gonzalo Pinilla-Buitrago[17] | Cory Merow[18] | Adam Wilson[2]

2 MATERIALS AND METHODS

2.1 Data collection

2.1.1 List of ecology and evolution publications citing R

To generate a list of papers in ecology and evolution that likely made use of the R programming language (R Core Team, 2023), we performed a query on the Scopus database (https://www.scopus.com) using the *rscopus* R package (Muschelli, 2019). We searched Scopus (performed August 19, 2022) for peer reviewed journal articles that: (1) included the words "ecology" or "evolution" in an "all fields" search (which searches article titles, keywords, abstracts, and journal titles); (2) were published in journals within the subject area "agriculture and biological sciences"; (3) were published after January 1, 2010; (4) were written in English (as this is currently the dominant language of publication in ecology and evolution; Mauranen et al., 2010); and (5) included a citation of R in their reference list.

2.1.2 Checking for code and data availability

We manually evaluated a randomly chosen subset of the publications on our overall list. We selected a total of 1001 papers, evenly distributed across the time period (77 per year * 13 years). Papers that cited R but did not use it (or were unclear on whether they used it; $n = 3$) were discarded and replaced by a randomly selected paper from the same year. For each publication in this subset, we manually identified whether the publication shared any R code, either as supplementary information, or via a link (e.g., to a Github repository). For each paper, we (i) checked for the presence of code in supplemental material, (ii) skimmed publications for code and data availability statements, (iii) searched through publications for terms associated with code (i.e., "code", "supplement", "appendix", "R", "script" "Github"), and (iv) searched publications for URLs. Papers were scored with a binary variable indicating whether they shared R code or not. We did not distinguish between publications which shared sufficient code for reproduction and those which did not. We also did not attempt to rerun the code or assess its reproducibility, and only recorded the presence of any code, even if it was incomplete. Where code was included, we recorded the license the code was provider under, or lack thereof. We also assessed whether publications were open access and whether they shared open data in order to understand the importance of open code relative to these other open-access components. Open access information was provided by the *rscopus* R package (Muschelli, 2019). Open data was scored as a binary variable indicating whether the authors shared the full set of raw data underlying the analyses or not. To control for differences in citation rates among journals, we downloaded impact factor information using the *scholar* R package (Keirstead, 2016) on June 16, 2023. To estimate the proportion of publications which use R but do not properly cite it, we screened 130 randomly selected publications evenly distributed across the time period. These publications were selected using identical criteria to the publications that did cite R, except that they did not include R in their list of references.

2.2 Checking for code citations

Where code was shared in a citable location such as a DOI or URL ($n = 33$), we assessed whether the code itself was cited by querying the Scopus database for the URL (and DOI, where appropriate) using the *rscopus* R package (Muschelli, 2019). Publications where code was shared in appendices or supplementary information ($n = 22$) were excluded, as there was no way of distinguishing citations of the code with citations of the publication itself.

2.3 Analyses

All analyses were conducted in R version 4.3.0 (R Core Team, 2023). All R scripts underlying these analyses are available at: https://github.com/bmaitner/R_citations and via Zenodo (Maitner & Lei, 2024). For data processing, we used the R packages *stringdist* version 0.9.12 (van der Loo, 2014), *tidyverse* version 2.0.0 (Wickham et al., 2019), *googledrive* version 2.1.1 (D'Agostino McGowan & Bryan, 2023), and *googlesheets4* version 1.1.1 (Bryan, 2023); for analyses, the packages *bbmle* version 1.0.25.1 (Bolker & R Development Core Team, 2022), *DHARMa* version 0.4.6 (Hartig, 2022), *MuMIn* version 1.47.5 (Barton, 2023), *rscopus* version 0.6.6 (Muschelli, 2019), *rsq* version 2.6 (Zhang, 2018), *scholar* version 0.2.4 (Keirstead, 2016), and *stats* version 4.3.0 (R Core Team, 2023); for plotting, the packages *ggplot2* version 3.4.4 (Wickham, 2016), *ggpmisc* version 0.5.5 (Aphalo, 2022), and *questionr* version 0.7.8 to identify all R packages used.

2.3.1 Proportion of papers sharing code over time

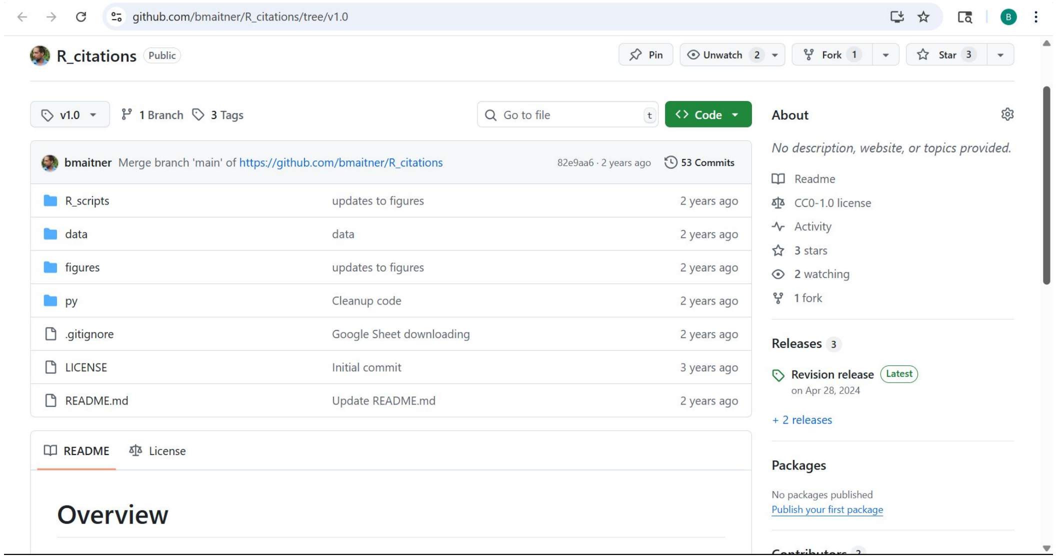
We tested for a trend in code-sharing over time by modeling code sharing (binary, yes/no) as a function of the year (relative to 2010) using a generalized linear model. Modeling was performed using the function *glm* in the *stats* R package (R Core Team, 2023) with a binomial error distribution. We similarly tested for temporal trends in two other open-science components, open-access publication (binary) and open data (binary). We also tested whether open-access or open-data papers were disproportionately likely to share code using chi-square tests via the *chisq.test* function in the *stats* R package (R Core Team, 2023).

2.3.2 Impact of code sharing on citations

We additionally modeled the relationship between code sharing and citation count using generalized linear models in R. We modeled the dependent variable (cumulative number of citations of each article by 2022) using a Poisson distribution, which models the number of independent events occurring within a period of time (Bolker, 2008). In addition to the predictor variable for code sharing (binary, yes/no), we included other variables that were hypothesized to influence citation count. Data sharing (binary, yes/no) may increase citation counts as readers may cite papers as data sources (Christensen et al., 2019; Piwowar et al., 2007). Open access (binary, yes/no) may also increase citation counts by reaching a broader set of readers (Tang et al., 2017). Publications accumulate citations over time, and so citation count should increase with publication age (continuous, 1–13 years). Finally, publications in higher impact journals may be more likely to be read and cited, and hence, journal impact factor (continuous, 0–11.633) may be positively associated with citation count. In addition to main effects, we considered two classes of interactions: (1) interactions between publication age and other main effects, which are appropriate if a main effect modifies the rate at which a publication accumulates citations over time; and (2) interactions between open-science criteria (i.e., open access, open code, and open data), which are appropriate if there are synergistic effects of meeting multiple open-access criteria. We compared 11 models (including one null model) that differed in complexity and that represented different hypotheses regarding the factors that influence citations (Table 1). Continuous variables were scaled and centered. Overall model pseudo-R^2 for the best-performing model was calculated using the function *r.squaredGLMM* in the *rsq* package (Zhang, 2018).

TABLE 1. Candidate models of citation count.

ID	Models	df	ΔAIC
1	Citations ~ Impact factor × Age + R code shared × Age + Open access × Age + Data shared × Age + Data shared × R code shared + R code shared × Open Access + Open access × Data shared	13	0.0
2	Citations ~ Impact factor × Age + R code shared × Age + Data shared × Age + Data shared × R code shared	9	834.0



Transparency and Reproducibility

Note: many journals and funders are now requiring code to be published

Questions on any of this?

Next time:

Before class: read 1.1 - 1.3

During class:

- Discuss 1.1-1.3
- Installing and setting up R, RStudio, and Github