Last rendered: 18 Jun 2021

Supplementary Materials for "EcoEvoApps: Interactive Apps for Teaching Theoretical Models in Ecology and Evolutionary Biology"

Preprint for bioRxiv

Authors: Rosa M. McGuire[1*], Kenji T. Hayashi[1*], Xinyi Yan[2*], Madeline C. Cowen[1], Marcel C. Vaz[1], Lauren L. Sullivan[3], Gaurav S. Kandlikar[3,4]

[1] Department of Ecology and Evolutionary Biology, University of California, Los Angeles

[2] Department of Integrative Biology, University of Texas at Austin

[3] Division of Biological Sciences, University of Missouri, Columbia

[4] Division of Plant Sciences, University of Missouri, Columbia

[*] These authors contributed equally to the writing of the manuscript, and author order was decided with a random number generator.

Authors for correspondence:

Rosa M. McGuire: rmcguire1@ucla.edu

Kenji T. Hayashi: kthayashi@ucla.edu

Xinyi Yan: xinyiyan@utexas.edu

Gaurav S. Kandlikar: gkandlikar@missouri.edu

Coauthor contact information:

Madeline C. Cowen: mcowen@ucla.edu

Marcel C. Vaz: marcelcvaz@gmail.com

Lauren L. Sullivan: sullivanll@missouri.edu

Table of contents

Supplement S1: Table of apps included in the `ecoevoapps` R package

App name	Function to run app in R	Link to web app
Single population growth in continuous time	shiny_singlepop_continuous()	https://ecoevoapps.shinyapps.io/single-population-continuous/
Single population growth in discrete time	shiny_singlepop_discrete()	https://ecoevoapps.shinyapps.io/single-population-discrete/
Structured population growth	shiny_structured_pop()	https://ecoevoapps.shinyapps.io/structured_population/
Source-sink dynamics	shiny_metapopulation()	https://ecoevoapps.shinyapps.io/source-sink-dynamics/
Offspring size-fitness trade-off	shiny_smith_fretwell()	https://ecoevoapps.shinyapps.io/smith-fretwell-app/
Consumer-resource dynamics	shiny_consumer_resource()	https://ecoevoapps.shinyapps.io/consumer-resource-dynamics/
Lotka-Volterra competition	shiny_lvcompetition()	https://ecoevoapps.shinyapps.io/lotka-volterra-competition/
Biotic resource competition	shiny_biotic_comp()	https://ecoevoapps.shinyapps.io/biotic-resource-competition/
Abiotic resource competition	shiny_abiotic_comp()	https://ecoevoapps.shinyapps.io/abiotic_resource_competition/
Infectious disease	shiny_infectious_disease()	https://ecoevoapps.shinyapps.io/infectious-disease-dynamics/
Island biogeography	shiny_island_biogeo()	https://ecoevoapps.shinyapps.io/island-biogeography/

Supplement S2: Table of external apps contributed to the EcoEvoApps website

App name	Contributor	Link to web app
Stage-structured population growth	Mayda Nathan	https://gauravsk.shinyapps.io/stage-structured-growth/
Stage-structured population growth with demographic stochasticity	Mayda Nathan	https://gauravsk.shinyapps.io/demog-stochasticity/
Annual plant competition	Gaurav Kandlikar	https://gauravsk.shinyapps.io/annual-plant/
Plant-soil feedbacks	Gaurav Kandlikar	https://gauravsk.shinyapps.io/microbe-mediated-fitdiffs/

Supplement S3: Community guidelines for contributing to EcoEvoApps

EcoEvoApps has been developed with an explicit focus on enabling contributions from across the EEB community. To this end, we have established a range of mechanisms by which community members with varying levels of theoretical, computational, and pedagogical expertise can contribute to this project. Here we outline these mechanisms and provide guidelines for how community members can become involved in this project. The most up-to-date version of these guidelines can be found on the EcoEvoApps website (https://ecoevoapps.gitlab.io/contribute). Beyond the contribution guidelines described here, we expect all individuals contributing to this project in any capacity to adhere to our Contributor Covenant Code of Conduct (https://ecoevoapps.gitlab.io/codeofconduct).

Mechanism 1: Write and contribute a new app

Development of the `ecoevoapps` R package and EcoEvoApps website takes place collaboratively on GitLab (https://gitlab.com), where we write, review, and deploy new apps. If there is an app that does not already exist as part of EcoEvoApps that you would like to see implemented, you can help the project grow by writing and contributing a new app. The most direct way to do this is to develop the app directly in the `ecoevoapps` repository on GitLab (https://gitlab.com/ecoevoapps/ecoevoapps). See the section "Contributing to EcoEvoApps via GitLab" for details regarding this workflow. However, we recognize that familiarity with the git version control system and GitLab can vary. Therefore, new apps developed locally may also be contributed by emailing them to us at ecoevoapps@gmail.com. New apps that are contributed in this manner will be incorporated into the `ecoevoapps` R package (unless otherwise requested by the contributor) and uploaded to the EcoEvoApps website. In both cases, new apps should be developed so as to satisfy the basic app requirements described in the section "Review process for contributed apps." You may wish to inquire about your intent to write and contribute a new app before doing so in order to prevent any miscommunications or duplication of efforts.

Apart from contributions of fully-functioning apps, we also welcome contributions of code that simulates model dynamics but which has not yet been made into an interactive app. Such submitted code can be used by core developers or other community members as the groundwork for writing a new app, potentially in collaboration with the author of the original code. If you are interested in contributing code in this manner, please contact us at ecoevoapps@gmail.com.

Mechanism 2: Contribute an existing app

If you have already written and deployed an EEB-related web app and you are interested in having it included as part of the EcoEvoApps project, you can contribute your app by emailing us the URL for the app at ecoevoapps@gmail.com. External apps contributed in this way will be linked to from within the EcoEvoApps website, but will not be natively incorporated into the `ecoevoapps` R package. Note that we will be using an opt-in system for inclusion of external apps as part of the EcoEvoApps website. That is, we will only include on the website external apps for which the author of the app has requested or consented that their app be linked to. In some cases we may inquire from our end about the possibility of providing a link to an external app of interest on the website. In the interest of making the process of contributing external apps efficient and accessible, external apps need not be written using R and the `shiny` framework, nor do we expect these apps to adhere as strictly to the review criteria described in the section "Review process for contributed apps" as long as they fall within the scope of the EcoEvoApps project. However, in some cases we may request that some minor modifications be made to these apps in order to satisfy some basic app requirements prior to including them on the EcoEvoApps website.

Mechanism 3: Revise an existing app

Community members can also contribute to the EcoEvoApps project by revising apps that are already available in the `ecoevoapps` R package or on the EcoEvoApps website. Possible revisions to an app include, but are not limited to, fixing known issues, adding new features or functionalities, expanding or revising the description text, and translating the app into another language. The best way to revise an app is to do so directly in the `ecoevoapps` GitLab repository (https://gitlab.com/ecoevoapps/ecoevoapps). See the section "Contributing to EcoEvoApps via GitLab" for details regarding this workflow. If you are not comfortable with directly modifying an app using GitLab, you can also email us the revised code or translated text at ecoevoapps@gmail.com, or you can submit a feature request as described in the section "Mechanism 5: Provide feedback or request a feature." Note that you need not be the original contributor of an app in order to revise it.

We also welcome efforts to convert external apps made available on the EcoEvoApps website into apps that can be natively integrated into the `ecoevoapps` R package by rewriting them in the R programming language using the `shiny` R package. Doing so can help unify the functionality of the `ecoevoapps` R package that can be accessed offline from the R console and the EcoEvoApps website that can be accessed online. If you intend to undertake this effort, you should (1) obtain permission from the original contributor of the external app to reproduce and modify their app as necessary and (2) follow the instructions in the section "Mechanism 1: Write and contribute a new app" to contribute a new version of the app.

Mechanism 4: Review a contributed app

In order to ensure app quality, whenever a new app is contributed or an existing app is updated it will undergo a standardized review process to confirm that it runs as expected and that it meets the necessary requirements and community standards. Community members can sign up to be an app reviewer using this form (https://forms.gle/GCSfEvauP7RwALK67) so that we can get in touch with you when there is an app pertaining to your area of expertise that needs to be reviewed. See the section "Review process for contributed apps" for details regarding the app review workflow.

Mechanism 5: Provide feedback or request a feature

Beyond direct contributions to the EcoEvoApps project in the form of submitting, revising, or reviewing apps, we greatly value feedback and suggestions from community members. If you spot an issue with any of the existing apps, you can submit a bug report using this form (https://forms.gle/M483PXy1AVAYB9oq5) or by directly opening an issue in the `ecoevoapps` GitLab repository (https://gitlab.com/ecoevoapps/ecoevoapps/-/issues) or the GitLab repository for the EcoEvoApps website (https://gitlab.com/ecoevoapps/ecoevoapps.gitlab.io/-/issues). You can similarly use these avenues to request new features for existing apps or to request entirely new apps that you would like to see implemented. You can also directly correspond with us about any issues, suggestions, or requests by emailing us at ecoevoapps@gmail.com.

Mechanism 6: Contribute classroom use cases and activities

To facilitate the adoption of EcoEvoApps in the classroom, we intend to compile and make available to community members resources, use cases, and activities that exemplify how instructors can use these apps in an educational context. If you are an instructor and are willing to share your experience or course materials pertaining to the use of EcoEvoApps in your classes, please get in touch with us at ecoevoapps@gmail.com. We also welcome testimonials and feedback from students who have interacted with EcoEvoApps in one or more of their classes. Further details regarding this effort are forthcoming.

Review process for contributed apps

Newly contributed apps or modifications to existing apps will be reviewed following a standard procedure prior to inclusion in the `ecoevoapps` R package or EcoEvoApps website. In this section we describe the basic requirements for contributed apps and provide an overview of the app review process.

Basic requirements for all apps:

1. Title of the app
2. Short description of the model or app
3. Equations or diagrams of the model (when applicable)
4. Definitions of all variables and parameters
5. At least one user-defined input
6. At least one reactive figure complete with labels and legends
7. References (when applicable)

Beta: Newly contributed or modified apps will first be evaluated by one or more core developers for whether they satisfy all of the basic app requirements listed above. Apps that have been confirmed to satisfy these criteria will be marked as beta and will be uploaded to the EcoEvoApps website.

Reviewed: Recently contributed or modified apps in the beta stage will next be reviewed by one or more app reviewers, who will check whether the apps can be run without runtime errors (e.g. all figures update in response to user inputs, no error messages, no unexpectedly long delays, etc.). Apps that have passed this basic review process will be marked as reviewed on the EcoEvoApps website.

Stable: Additionally, reviewers who are familiar with the biological concepts underlying the apps and the expected behavior of the models at hand can review the description text, equations or diagrams, and model behavior (e.g. bifurcation points, equilibria, etc.) to confirm that the apps are not only fully functional but also biologically accurate. Apps that have passed this advanced review process will be marked as stable on the EcoEvoApps website, and will be incorporated into the `ecoevoapps` R package (unless otherwise requested by the contributor). Apps may be designated as stable without first being marked as reviewed if the reviewer sees fit.

Contributing to EcoEvoApps via GitLab

The most direct way to contribute a new app or to revise an existing app is to directly work with the `ecoevoapps` GitLab repository (https://gitlab.com/ecoevoapps/ecoevoapps). In this section we describe the general workflow for contributing to EcoEvoApps via GitLab. Note that this is not meant to be a comprehensive tutorial for git and GitLab, but rather is an outline of the general process of contributing to this project via GitLab.

1. If this is your first time contributing to EcoEvoApps via GitLab (or if there is a new version of the `ecoevoapps` R package available), you will need to first make sure that you have installed the most recent version of the `ecoevoapps` R package. Instructions for installation can be found in the repository's README file.

2. Fork the `ecoevoapps` repository on GitLab. This creates a personal copy of the repository.

3. Clone the forked repository to a desired directory on your local device. This allows you to modify your copy of the repository locally.

   ```
   git clone https://gitlab.com/YOUR-USERNAME/ecoevoapps.git
   ```

4. (Optional) It is often recommended that you keep your fork synced with the original repository while you work on it. This can be done by adding the original repository as an upstream remote and periodically obtaining any changes that have been made there. Once the upstream remote is configured, you can fetch and merge any changes in the original repository into the master branch of your fork on demand. Local updates to your fork will also have to be pushed to your remote origin. While you work on your feature branch (step 5) you may wish to periodically switch back to your master branch and merge any upstream changes to keep your fork up to date.

   ```
   git remote -v
   git remote add upstream https://gitlab.com/ecoevoapps/ecoevoapps.git
   ```

```
git checkout master
git fetch upstream
git merge upstream/master
git push origin master
```

5. Once you are ready to start working on a new app or feature, create a feature branch in your fork with a descriptive title. This allows you to make modifications to your fork without directly modifying the master branch.

   ```
   git checkout -b FEATURE-BRANCH
   ```

6. Now you can work on developing your new app or feature on the feature branch, staging, committing, and pushing any notable changes as you go. Remember to keep in mind the review criteria described in the section "Review process for contributed apps" when writing or revising an app. Note: If you have configured the upstream remote in step 4, we generally recommend against merging any upstream changes into your feature branch in order to avoid duplication of commits when merging the feature branch to the original repository (step 7). We will reach out to you upon receiving your merge request if your feature branch needs to be updated to match new changes to the original repository.

   ```
   git stage -A
   git commit -m "COMMIT MESSAGE"
   git push origin FEATURE-BRANCH
   ```

7. Once you are ready to submit your new app or feature, navigate to your forked repository on GitLab and create a merge request. Make sure that you select your feature branch as the source branch and the master branch of the original `ecoevoapps` repository as the target branch. Add a descriptive title to your merge request and describe the additions or changes you have made in the description box (description templates forthcoming). When submitting your merge request, you can select to have your source branch deleted upon approval of the merge request (this will not automatically delete the feature branch on your local device). If your merge request contains an excessive number of commits, you can also select to have your commits squashed into one upon merging.

8. Your merge request will be reviewed by one or more of the core developers and app reviewers (when applicable). Once the merge request and the submitted app or feature has been satisfactorily reviewed (see section "Review process for contributed apps"), the request will be accepted and your contributions will be incorporated into the original `ecoevoapps` repository and the EcoEvoApps website. We will communicate with you about any issues that need to be resolved prior to merging, should they arise.

Credit for contributions

We strive to give appropriate credit to community members who have contributed to the growth and maintenance of this community-based project. To this end, all community members who have contributed to the EcoEvoApps project through the mechanisms 1-4 and 6 described above will be credited accordingly on the EcoEvoApps website (https://ecoevoapps.gitlab.io/people) unless requested otherwise. App contributors will additionally be displayed on the landing page for the corresponding app, and can also embed their name within the app if they wish to do so. If a contributed app is natively incorporated into the `ecoevoapps` R package, the app contributor will also be listed as an author of the R package. If you are interested in becoming further involved in this project (e.g. as a core developer), please contact us at ecoevoapps@gmail.com.

Usage of EcoEvoApps

EcoEvoApps is a free, open-source project. The `ecoevoapps` R package is developed under the GNU General Public License v3.0 (https://gitlab.com/ecoevoapps/ecoevoapps/-/blob/master/LICENSE.md) and the EcoEvoApps website is developed under the MIT License (https://gitlab.com/ecoevoapps/ecoevoapps.gitlab.io/-/blob/master/LICENSE.md). Users are free to use, modify, and distribute this software as long as conditions of the respective licenses are satisfied. If you are using the apps for teaching, research, or self learning, we would love to hear from you, but there is no need to obtain our permission for use. If using EcoEvoApps in your work, please cite the published manuscript.

Supplement S4: Teaching materials used at MU and UCLA

Teaching materials included in this supplement:

Supplement S4-1: MU Lotka-Volterra competition worksheet

At Home Lab – Lotka Volterra

This week for your at-home lab we will be using simulations of the Lotka-Volterra equations to better understand how these dynamics work, and the consequences of altering parameters in the equations for competitive dynamics. You will be using this web-based simulation of Lotka-Volterra equations created by our scientist spotlight Dr. Gaurav Kandlikar.

https://ecoevoapps.shinyapps.io/lvcomp_simplified/ (Links to an external site.)

For this week, you will need to go through the steps outlined below, and write answers to all of the numbered questions. You will need to turn in a document that answers all the questions below. You can complete this assignment alone, or you can work in your groups. If you work in your group, you must submit your own assignment in your own words. If you submit the exact answers as someone else, you both will receive a zero.

Part 1 - competition coefficients alpha and beta

First, let's remind ourselves that Lotka-Volterra is very similar to population growth equations but has the added influence of the other competitor in the equation.

1. What are alpha and beta in the equations? What do they represent?

Leave all of the parameters the same as when you opened the website, and change the alpha and beta parameters to 0.01 (as small as they go).

2. Describe the population dynamics of the two species in this figure? Does this look familiar? What does it look like? Are these two species coexisting under these parameters? What is your evidence?

Now systematically increase and decrease the alpha and beta parameters independently, but leave all other parameters the same.

3. Take a screenshot of the following cases: A) a set of alpha and beta values where species 1 competitively excludes species 2. B) a set of alpha and beta values where species 2 competitively excludes species 1. C) a set of alpha and beta values where both species coexist, but species 2 has a higher population size than species 1 by the end of the simulation (at time 100), and alpha and beta are each greater than 0.4. D) a set of alpha and beta values where both species coexist, but species 1 has a higher population size than species 2 by the end of the simulation (at time 100), and alpha and beta are each greater than 0.4. Include these screenshots in your response, and add figure legends that include your alpha and beta values for each case. What does this tell you about alpha and beta?

Part 2 - population growth rate

Hit refresh on the website so the parameter values return to their initial conditions. Now let's explore how growth rates affect the outcome of Lotka-Volterra competition. To isolate growth rate let's set all other parameters to be equal and change growth rates systematically to see their effect.

4. What happens when you set the alpha and beta values to both be 0.3, the carrying capacities for both species to be 200, and the initial population sizes to be 50 for both species? Take a screenshot, put this in your write up, and include a figure legend. What explains the differences in these growth curves?

5. If the carrying capacity for both species is 200, why does the population size for either species never get higher than ~170?

6. Find a situation where one species reaches its carrying capacity. How is this species able to reach this population size?

7. Now, set both K's to be 200, both initial population sizes to be 50, and both competition coefficients (alpha and beta) to be 0.9. What do you see when your r1 (growth rate for species 1) = 0.75, and r2 (growth rate for species 2) = 0.1? Take a screenshot, add a legend, and include it in your writeup. Are these species coexisting? Why do you think this? **Without changing any parameter values before you do this**, make a hypothesis about what will happen to the population dynamics of both species if the simulation runs for 1000 time steps given the parameter values above. Explain why you made this hypothesis.

8. To test your hypothesis, change the length of your simulation to 1000. Take a screenshot of your result, add a figure legend and include it in your writeup. Does this support or refute your hypothesis. Why or why not? If this figure refutes your hypothesis, explain why this outcome occurred.

Part 3 - Initial Conditions and Carrying Capacity

Now we will explore how carrying capacity and initial population size alter competition under Lotka-Volterra dynamics. Refresh your website so your parameters return to their initial values.

9. Change the carrying capacity of species 1 (K1) to 50. What population outcome do you see? Take a screenshot, add a legend, and include this in your report.

10. Leaving your carrying capacities at K1 = 50 and K2 = 200, can you find a set of parameter values where Species 1 does not go extinct? Take a screenshot, add a legend, and include it in your write up. Provide all of your parameter values (e.g. alpha = 5). What does it take for species 1 to be able to coexist when it has a very low carrying capacity? How much does initial population size matter relative to the species' growth rates, relative to their competition coefficients alpha and beta?

Supplement S4-2: MU SIR worksheet

At Home Lab 2 SIR Simulations

This week we will be exploring SIR models with simulations. While we are off in our units (SIR models were in Unit 2), I believe working through these models will help you understand the concepts better. **You can work with your groups for this assignment if you want, but you need to turn in your own work, in your own words.**

We will be using another simulation created by Dr. Kandlikar, which you can find at this link. https://ecoevoapps.shinyapps.io/sir-simlified/ (Links to an external site.) We will be focusing on the SIR and SEIR models. Please answer all of the following questions below on a separate word document and submit the answers.

Part 1 - SIR Models

Go to the SIR page, and refresh so your parameters are all at their starting values.

Question 1: What do the S, I and R parameters stand for in this model? (1 point)

Question 2: When your vaccination rate is zero (as it is when you refresh the website), interpret the graph of population size with time. What does this figure tell you?? What does this mean for the potential for the virus to spread in these populations? Why do you think this? (2 points)

Question 3: What happens when you set your vaccination rate = 1? Take a screen shot of this new figure, and interpret it (what is it telling you?). What does this mean for the potential for the virus to spread in these new populations? (2 points)

Question 4: With the parameters set as they were for question 3, is there any value of the vaccination rate that completely eliminates the potential for the virus to spread in the population? How did you come to this answer? (1 points)

Question 5: We have seen that at a low infection rate (0.01), it is difficult to eliminate the potential for virus spread in the population unless 100% of individuals are vaccinated. Refresh the website, and then increase your infection rate to 0.2. Take a screen shot of this new figure, and interpret it (what is it telling you?). Can the virus still spread in the population? Why or why not? How does the population size of the infected individuals explain your answer to if the virus can still spread in the population? (2 points)

Question 6: Keeping the parameters the same as for question 5, try to find a vaccination rate that completely eliminates the potential for virus spread. What is it? (1 point)

Question 7: Based on your answers and exploration of the SIR models so far, how does increasing infectivity of a virus influence its spread and ease of elimination from a population? (2 points)

Part 2 – SEIR Models

Go to the SEIR page, and refresh so your parameters are all at their starting values. This SEIR model is very similar to COVID-19, and how it affects our population.

Question 8: What do the S, E, I and R parameters stand for in this model? (1 point)

Question 9: Explore the SEIR model by changing parameters and examining outcomes. Based on your exploration, provide 2 things you have learned about the SEIR model. For each, provide your parameter values, and a corresponding figure. Explain what you were investigating, and what insight you gained from the graph (what it is telling you based on the data available). (4 points)

Supplement S4-3: UCLA extra credit worksheet

I. Pre-Activity Survey (Optional/Anonymous)

II. Part 1: Lotka-Volterra Competition Model

- Watch the video on CCLE
- Use Lotka-Volterra competition app: https://ecoevoapps.shinyapps.io/lotka-volterra-competition/
- Using any parameters of your choice, explore these 4 scenarios:
  - Competitive exclusion:
    - K2 < K1/alpha and K1 > K2/beta
    - K2 > K1/alpha and K1 < K2/beta
  - Priority effects/coexistence:
    - K2 > K1/alpha and K1 > K2/beta
    - K2 < K1/alpha and K1 < K2/beta
- For each scenario:
  - Look at the time series graph-What happens to species 1 and species 2 in the long run? Does one of them go extinct? Why?
  - Look at the isoclines and the arrow that shows the combined population trajectory-what happens in each case? What is driving this pattern? Do the outcomes change by using different initial population sizes (while keeping everything else the same)?
- Case Study 1: Honeybees (Species 1) compete for nectar with bumblebees (Species 2) - At a certain habitat patch, the initial density of honeybees is 120 while the initial density of bumblebees is 40. The parameters are listed as follows: r_1=0.2, r_2= 0.3, K_1= 150, K_2= 200, alpha=1.2, beta=0.7. Given the initial conditions, what is the expected outcome?
- Case Study 2: Think of two migratory bird populations: species 1 and species 2. The parameters are listed as follows: r_1= 0.4, r_2 = 0.3, K_1= 500, K_2= 400, alpha=1.6, beta=1.2. These two species compete every year for nesting sites on cliffsides on the coast of Peru. In 2018, the number of individuals of species 1 that arrived at the beginning of the breeding season was 90, while 150 individuals of species 2 arrived. The following year, 140 individuals of species 1 arrived and 150 of species 2 arrived. Given the initial conditions, what is the expected outcome? What is happening?

III. Part 2: Island Biogeography

- Watch the video on CCLE
- Read the introductory text: (optional) https://web.stanford.edu/group/stanfordbirds/text/essays/Island_Biogeography.html
- Use Island Biogeography app: https://ecoevoapps.shinyapps.io/island-biogeography/
- Using any parameters of your choice, explore the 4 cases:
  - Larger island is more diverse than smaller island;
  - Both larger and smaller island are similarly diverse;
  - Smaller island is more diverse than larger island;
  - Both islands are less diverse compared to the initial settings (species pool size).

- Study case:
  - Imagine two similar sized islands, A and B (1 kmˆ2 each), that are 2 and 6 km away from the mainland, respectively; now imagine island B will expand its area at a rate of 1 kmˆ2 every 1,000 years.

IV. Take the quiz (required for extra credit points)- feel free to use the competition/island biogeography app as you solve the quiz (if needed).

V. Post-Activity Survey (Optional/Anonymous)

Supplement S4-4: UCLA quiz

1. What is the outcome of Case Study #1?

    a. Priority effects
    b. **Competitive exclusion of honeybees by bumblebees**
    c. Competitive exclusion of bumblebees by honeybees
    d. Stable coexistence of honeybees and bumblebees

2. What is the outcome of Case Study #2?

    a. Species 1 wins
    b. **Priority effects**
    c. Stable coexistence of species 1 and species 2
    d. Species 2 wins

3. What is the coexistence criterion?

    a. K2 < K1/alpha and K1 > K2/beta
    b. K2 > K1/alpha and K1 < K2/beta
    c. K2 > K1/alpha and K1 > K2/beta
    d. **K2 < K1/alpha and K1 < K2/beta**

4. In other words, for two species to coexist, **intraspecific** competition needs to be greater than **interspecific** competition.

5. What happens in the priority effects scenario? (select all that apply)

    a. Species 1 always wins
    b. **There will be no coexistence**
    c. Species 2 always wins
    d. Competing species coexist
    e. **The outcome of competition depends on the initial conditions**

6. The Lotka-Volterra competition model is useful to show (Select all that apply):

    a. **Under what circumstances can two species coexist**
    b. Predict the outcome of predator-prey interactions
    c. **Predict the outcome of interspecific competition**
    d. Quantify niche overlap

7. Regardless of its size, every island that is farther from the mainland will have:

    a. Larger extinction rates;
    b. Larger immigration rates;
    c. Smaller extinction rates;
    d. **Smaller immigration rates.**

8. For two islands that are equally far from the mainland, the **smaller** one will have a **lower** species richness than the **larger** one, as **extinction** rates decrease with increasing island size, whereas **immigration** rates depend only on the distance from the mainland.

9. Can a large island be less diverse than a small one?

   a. No, the larger island will always have smaller extinction rates, thus higher species richness.
   b. No, the larger island will always have larger immigration rates, thus higher species richness.
   c. **Yes, the larger island will always have smaller extinction rates, but this may be offset by a greater distance from the mainland.**
   d. Yes, the larger island may have greater extinction rates if it is too far from the mainland.

10. What are the species richness of islands A and B, respectively, in the beginning of the study-case scenario (1 kmˆ2 each, 2 and 6 km away from the mainland, respectively)?

    a. $S_A = 33$; $S_B = 14$

11. How long will it take for island B to have as many species as island A?

    a. t = 2,000 years

12. The IB model is useful to show (select all that apply):

    a. **How immigration can be an important driver of species diversity;**
    b. **That species richness may be constant over time, although individual species are coming and going all the time;**
    c. That species whose niche overlaps too much end up being excluded competitively.

Supplement S5: Surveys for assessing teaching outcomes at MU and UCLA

Surveys included in this supplement:

Supplement S5-1: MU survey

Questions after specific webapps:

From a scale of 1 (not interested) to 7 (very interested), please rate your interest in the following topics:

1. Population ecology
2. Disease ecology
3. Community ecology
4. Global change ecology
5. Ecosystem ecology
6. Conservation ecology

From a scale of 1 (not confident) to 7 (very confident), please rate your confidence in the following subjects:

1. Exponential vs. Logistic growth
2. Carrying capacity
3. Sampling populations in the field
4. Analyzing population data
5. structured population dynamics
6. population growth rate
7. Fecundity effects on population dynamics

From a scale of 1 (not confident) to 7 (very confident), please rate your confidence in the following subjects:

1. Exponential vs. Logistic growth
2. population growth
3. two species interactions
4. competition coefficients
5. positive or negative species interactions
6. interspecific interactions vs intraspecific interactions
7. competitive exclusion
8. coexistence
9. Lotka-Volterra competition model

From a scale of 1 (not confident) to 7 (very confident), please rate your confidence in the following subjects:

1. SIR disease dynamics models
2. disease resistance
3. population dynamics
4. disease transmission

End of class survey questions

From a scale of 1 (not helpful) to 7 (very helpful), how helpful were the interactive apps as a way to generally learn mathematical models in this course?

From a scale of 1 (not helpful) to 7 (very helpful), please rate how much the interactive apps helped you learn the following models:

1. Structured population growth
2. SIR disease dynamics
3. Lotka-Volterra competition

What concepts did these model simulations help you understand better? [Free response] [Optional]

Do you suggest any improvements that would make the apps more valuable as a tool to learn these models? [Free response] [Optional]

Supplement S5-2: UCLA survey

Using interactive apps to teach mathematical models in a remote-learning ecology course

Pre-activity Survey:

I. Select the math classes you have taken:

- Calculus (Math 3A/B/C or LS 30A/B)
- Differential Equations
- Linear Algebra
- Other (write in)

II. From a scale of 1 (not interested) to 7 (very interested), please rate your interest in the following topics:

- Population ecology
- Disease ecology
- Community ecology
- Global change ecology
- Ecosystem ecology
- Conservation ecology

III. From a scale of 1 (not confident) to 7 (very confident), please rate your confidence in the following subjects:

- Exponential vs. Logistic growth
- Carrying capacity
- Population dynamics and time series
- Molecular ecology
- Population growth rates
- Competitive interactions
- Competition coefficients
- Competitive exclusion
- Biogeochemical cycles
- Coexistence
- Lotka-Volterra competition model
- Island Biogeography
- Immigration rate
- Extinction rate
- Mainland/Island Dynamics

Post-activity Survey:

I. From a scale of 1 (not helpful) to 7 (very helpful), how helpful were the interactive apps as a way to generally learn mathematical models in this course?

- Lotka-Volterra Competition
- Island Biogeography

II. From a scale of 1 (not helpful) to 7 (very helpful), please rate how much the interactive apps helped you learn the following concepts (Will also add a not applicable choice):

- Exponential vs. Logistic growth
- Carrying capacity
- Population dynamics and time series
- Molecular ecology
- Population growth rates
- Competitive interactions
- Competition coefficients
- Competitive exclusion
- Biogeochemical cycles
- Coexistence
- Lotka-Volterra competition model
- Island Biogeography
- Immigration rate
- Extinction rate
- Mainland/Island Dynamics

III. What concepts did these model simulations help you understand better? [Free response]

IV. Do you suggest any improvements that would make the apps more valuable as a tool to learn these models? [Free response]

Supplement S6: Student comments on their learning experience with EcoEvoApps at UCLA

What concepts did these model simulations help you understand better?

- It was nice to have visual representation of the data. I could physically see how different input values were affecting the results, which was really helpful in making connections.
- Helped clarify the lot Volterra model a lot more
- i liked when there were graphs and each part was explained but i would like an example problem to solidify it
- Conditions that lead to coexistence or competitive exclusion
- Carrying capacity of island biogeography
- Lotka Volterra Model
- I liked the island biogeography model because the equations are difficult to visualize.
- Understanding how to use variables to compare the strength of inter- and intra- specific competition and make predictions with that information. Visualizing how island size and distance affect immigration and extinction rates.
- The stimulations helped to understand both the models because I was able to apply to real life settings
- larger islands and the distance of the island having an impact on extinction rates.
- Both Lotka-Volterra and island biogeography.
- Logistic growth and carrying capacity
- The Lotka-Volterra model was super helpful in understanding the mathematics behind competitive interactions. Similarly, the island biogeography model helped me to better understand immigration, emigration, and interspecific competition
- Population dynamics
- they really helped put all the concepts together because you were able to see how tweaking one thing affects another
- How population dynamics occur through time and then how two populations interact with each other differently through time.
- How changing certain parameters lead to changes in population size
- It helped me understand competitive interactions as well as the impact that initial population numbers have
- Both concepts of Island Biogeography and Lotka-Volterra were better understood after using both apps. I personally feel like I learned more from the island biogeography app more but that could just be a testament to the concept being easier for me. Specifically, I understand the immigration and extinction rates for island biogeography very well. I think more information may be needed for Lotka-Volterra is needed on the app because I learned more from Rosa's video than the app itself.
- Lotka Voltera
- The model simulations helped me best understand the island biogeography theory and the Lotka-Volterra model.
- Lotka Volterra dynamics
- These mode simulations helped me better understand population dynamics.

- I now understand the impact of the competition coefficient on the stability or unstable equilibriums
- The Lotka- Volterra app helped me visualize the concepts learned in lecture, and adjusting the settings and tinkering on my own helped me understand the model better
- The Lotka Volterra model simulation was helpful in putting the difficult math in context for me, although it didn't help me understand the meaning of the math and how to apply it very well. This might just be my fault for having a harder time with mathematical models. The island biogeography model was interesting and I particularly enjoyed the animations. That simulation helped me understand the concept a lot better
- Definitely competitive exclusion/coexistence since it was easier to visualize
- Lotka-Volterra and Island Biogeography theory and how various variable manipulations affected two species and their ecological interaction
- the island biogeography model was very beneficial to understand the connection between size and distance
- These simulations helped myself better understand Lotka-Volterra Competition Models and competitive exclusion.
- Understanding under which conditions populations coexist and when one overcomplete the other
- They helped me visualize the relationships between each variable using graphs.
- These simulations helped clearly show how Lotka-Volterra is utilized to predict species competition as well as island biogeography theory is used to predict species richness for island- mainland dynamics.
- The Lokta Volterra Model explanation was very helpful!! The Island biogeopgraphy model helped outline the main concepts.
- Island biogeography (specifically how island size and distance each contributes to population dynamics) and Lotka-Volterra equations (specifically what the variables meant)
- Effect of distance versus area for island biography in a way easy to visualize
- competitive interactions between different species and island biogeography concepts
- Population dynamics using island biogeography model
- It really helped with Island biogeography theory! It was really helpful seeing how island sizes and distance affect the model
- These models helped me to better understand competitive interactions between 2 species as simulated by the Lotka-Volterra model, as well as how immigration and extinction rates are affected by the size and distance of islands.
- LV model = Competitive competition, Coexistence, and unstable equilibrium Island model = island biogeography model
- growth rates
- The lotka-Voltera simulation helped me understand what the lotka-Voltera predicts because it was more hands on than listening to its explanation during lecture.The island simulation also made it easy to understand how different variables and values of size/distance affect island populations. Visualizing these concepts made the model very clear.
- These models helped me to better understand competitive interactions and changes in population dynamics over time
- extinction rate
- The model simulations really helped me understand the relationships between the extinction

rates and immigration rates and distance in the island biogeography model, but the Lotka-Volterra didn't help me as much, but it still helped me understand the way Lotka-Volterra functions and the different outcomes that it can have.
- How size and distance from the island can impact extinction/immigration rates
- population dynamics and growth rates
- Interspecific competition, immigration, and extinction rates.

Do you suggest any improvements that would make the apps more valuable as a tool to learn these models?

- It would be nice to have a sample example given, so that students are able to correctly interpret the results shown on the graphs
- NA, thanks!
- showing more examples and step by step
- not right now
- Not really, both were very specific already
- not that I can think of!
- The lotka voltera interactive is already very helpful, but a blurb in the app about what each of the values means could drive the point home even better. I had to check a few times to make sure my understanding of each variable was correct.The apps were very easy to use. The instructions that Rosa and Marcel provided were excellent for doing the extra credit. I really enjoyed how I can apply real life concepts to the apps
- None. Overall great experience.
- No. It just constantly had to be reloaded if you went off the website for a short while, but that's not really an issue.
- It should be a bit more clearing with its instructions on how to use
- I think that the island biogeography model was almost perfect as is. It was very user friendly and easy to use. The Lotka-Volterra model could be simplified somewhat by reducing the number of variables that can be changed at once, or perhaps by designing sub modules to teach individual concepts
- I enjoyed it!
- sometimes the website would crash and i wasn't able to use itt on the laptop
- The coefficients and all the input values just become numbers after you throw so many at once. It would be helpful to accompany a slow-pace conceptual backing when tweaking each value individually while looking at the model.
- none at this time
- No suggestions!
- I don't have improvements for the island biogeography app; I found it very helpful. An improvement I have for the lotka-Volterra app is addressing what the open and closed dots mean as well as increasing the size of the graphs when the isoclines are intercepted in a closer range. It got really hard to see some effects when the parameters made the isoclines fairly close.
- no, i thought they were good as they were
- The apps were great and truly helped with improving my understanding of certain concepts. The only issue I had was that the website wouldn't work/load at times. I am not exactly sure

why, but I would have to refresh the page or visit it at a different time for it to completely load.

- N/A, these models were very helpful. However, just general stability could be improved (the models frequently crashed and needed to be reloaded)
- No
- Provide many questions but also add the answers so you can see why you got a problem incorrect
- It would be helpful if the lines in the island biogeography app are thicker, so they are easier to see with the graphs
- The Lotka Volterra model simulation would benefit from some explanation of how changes to the parameters affected the model. I think there were just a lot of parameters to focus on that it confused me
- It can explicitly say which species won
- n/a; very helpful tools!
- i think it would be beneficial to have the lotka voltera model explained with examples
- I took me a while to understand the results so if the app states "the population will coexist" "population 1 will outcompete population 2", etc. that would be helpful
- The site is a little buggy/laggy.
- Allowing a wider range of values for the parameters to see extreme results.
- The apps worked great. The lotka volterra model simulation is not as user friendly. It is hard to exactly understand which K1>k2/alpha you are inputting. I didn't know how to explicitly tell what I was doing.
- Apps are already super easy to use! Pairing them with the scenarios made it even clearer what the app was trying to demonstrate
- The Lotka-Volterra model would not load for the longest time. It took me around 45 minutes of opening it on different and devices to get it to open and/or stop freezing. I would suggest fixing this app before using it as an educational tool.
- Yes, the Lotka-Volterra app got disconnected a lot. It sometimes stops working and that's discouraging, but it was simple to use overall
- One aspect of the model that can be better explained is the model for coexistence for Lotka-voltera model.
- Maybe you could give some examples at the bottom for people to test before putting their own data in
- I had trouble using the Lotka-Volterra model app as it would freeze periodically and I would be unable to reload the page which made taking the extra credit quiz a little difficult.
- LV model = include things like K1, K2, etc. to label the lines;Island model = nothing – it's really good
- showing how to use it
- I would suggest ability to hover with the mouse over the variable you are changing, and a pop up box appears explaining clearly how adjusting that variable will affect the results.
- I would suggest adding a key that explains each of the variables that can be manipulated and expanding on how they affect the overall model.
- I think it is helpful
- The Lotka-Volterra models could be a bit more clear in how these different variables affect the outcomes of the models, but the island biogeography one was very clear.

- None, both apps were clearly explained well by both TAs and there was nothing too confusing about them. If anything, they helped me better understand the four possible outcomes of the Lotka-Volterra model better than I did before.
- It looks great
- I think they are pretty good overall. Maybe it'd be better if the instructions on how to use the apps could be clearer

Supplement S7: R code to recreate all analyses

Code to analyse classroom data for EcoEvoApps

Gaurav Kandlikar

4/23/2021

Analyze survey data from Univ. Missouri

Before analyzing the data, we have to scrub the data to remove student names and exclude some redundancies. These raw data files are not made available in the supplement to avoid exposing students' names.

```
### THIS CHUNK IS NOT EVALUATED, AS IT WOULD EXPOSE
### STUDENT NAMES
library(readr)
library(dplyr)
library(stringr)
library(tidyr)
library(tidyverse)
library(patchwork)
library(readxl)

# There are some students we need to exclude from analyses,
# because they submitted Quiz 1 after they might have used the apps.
exclude <- read_csv("teaching-survey/data/names-to-exclude.csv")
survey <- read_csv("teaching-survey/data/missouri-round2-all.csv") %>%
  select(-date)
survey <-
  survey %>%
  filter(!(name %in% exclude$name))

only_one_quiz <-
  survey %>%
  group_by(name) %>%
  summarize(how_many = n()) %>%
  filter(how_many == 31)

survey <-
  survey %>%
  filter(!(name %in% only_one_quiz$name)) %>%
  filter(!(is.na(response))) %>%
  # after thinking more about the controls, the following three questions
  # don't seem like true controls bc they were part of the curriculum throughout
  # the semester and are generally reflected in the apps themselves too.
  # this may muddy the results a little bit, so I am excluding them here.
  filter(!(topic == "structured_popgrowth" & question == "Population growth rate")) %>%
  filter(!(topic == "structured_popgrowth" & question == "Exponential vs. Logistic growth")) %>%
  filter(!(topic == "structured_popgrowth" & question == "Fecundity effects on population dynamics"))
```

```
name_id <- bind_cols(
  real_name = unique(survey$name),
  random_name = stringi::stri_rand_strings(length = 5, n = length(unique(survey$name)))
)

to_export <-
  left_join(survey, name_id, by = c("name" = "real_name")) %>%
  select(-name) %>%
  select(random_name, everything())

write_csv(to_export, file = "teaching-survey/data/missouri_scrubbed_data.csv")
```

The code above exports a new file called `missouri_scrubbed_data.csv`, which we can now analyze.

```
# Load libraries and read in data -----------------------------------------
library(tidyverse)
```

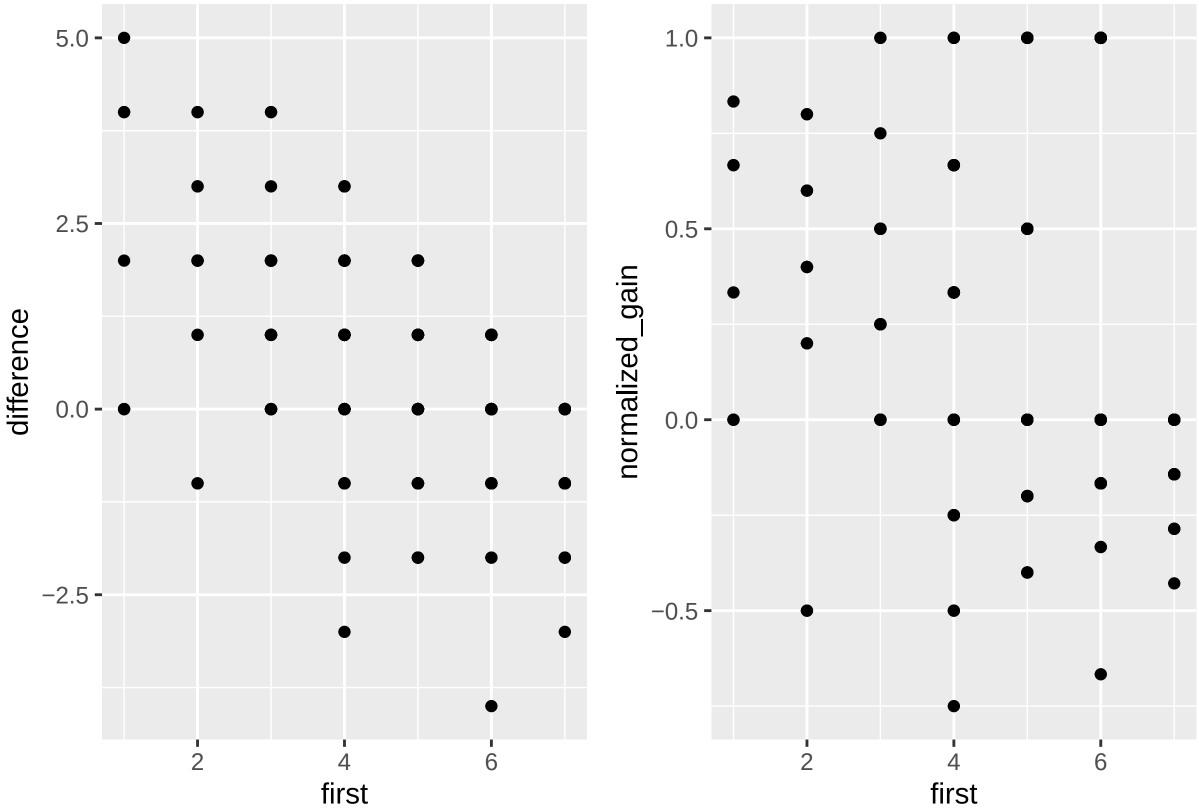
```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(patchwork)
library(readxl)
# missouri_data <-
missouri_data <-
  read_xls("teaching-survey/data/Survey_data.xls", sheet = 1)
```

```
# for Lotka-Volterra analysis ---------------------------------------------

for_lv_analysis <-
  missouri_data %>%
  filter(quiz_no %in% c("first", "second"),
         topic %in% c("lotka_volterra", "structured_popgrowth")) %>%
  pivot_wider(names_from = quiz_no, values_from = response) %>%
  mutate(difference = second - first) %>%
  filter(!(is.na(difference))) %>%
  mutate(normalized_gain = ifelse(second > first,
                                  difference/(7-first),
                                  ifelse(second == first,
                                         0,
                                         (second-first)/first)))
```

Now that we have normalized change scores, check whether normalized scores are less correlated with response to pre-quiz than the raw differences (as we expect them to be).

```
cor_pre_diff_lv <- ggplot(for_lv_analysis) +
  geom_point(aes(x = first, y = difference))
cor_pre_ndiff_lv <- ggplot(for_lv_analysis) +
  geom_point(aes(x = first, y =  normalized_gain))
cor_pre_diff_lv + cor_pre_ndiff_lv
```

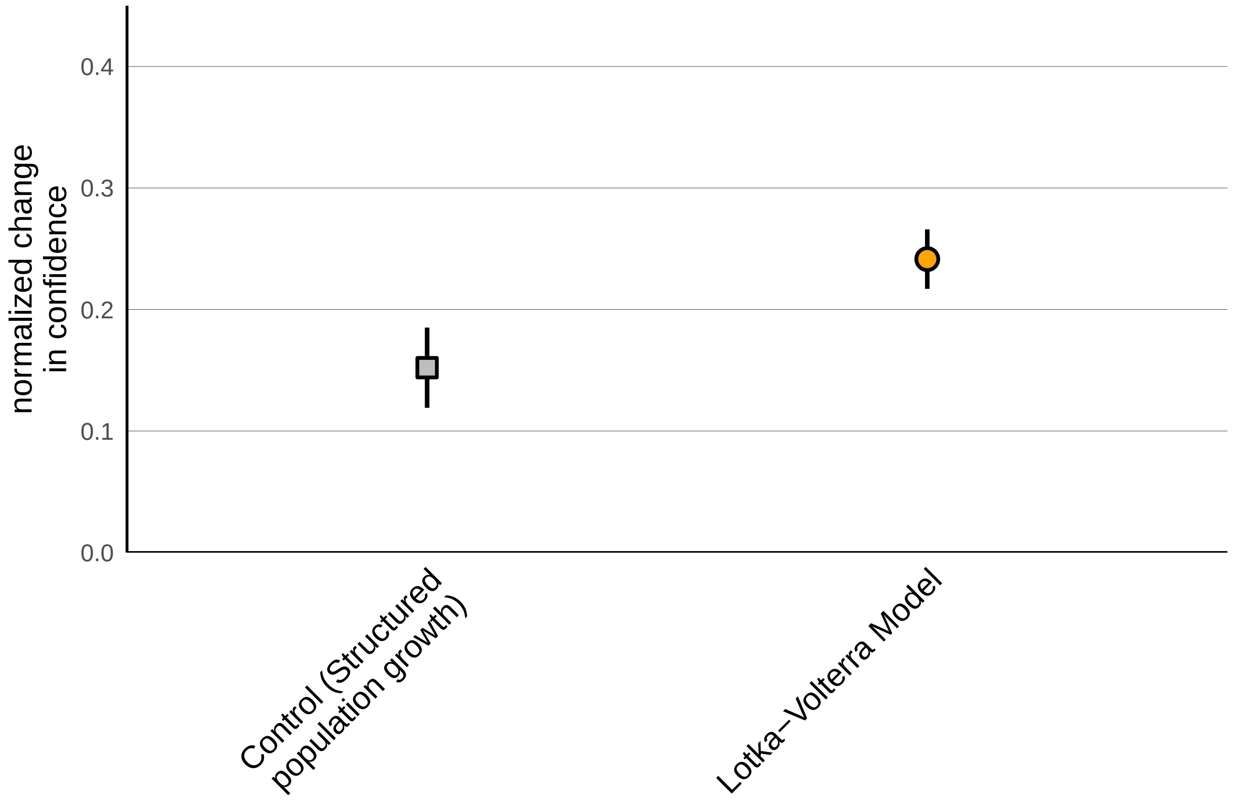


That looks better– less correlation between normalized change and pre-survey score.

Now, we can start evaluating how these normalized differences vary across different questions, or categories of questions.

```
# Summarize across the categories (LV and Structured pop growth)
lv_popg_cat_comparison <-
  for_lv_analysis %>%
  group_by(topic) %>%
  summarize(mean_ndiff = mean(normalized_gain),
            sd_ndiff = sd(normalized_gain),
            sem_ndiff = sd_ndiff/sqrt(n())) %>%
  arrange(topic)

lv_popg_cat_comparison$topic <-
  factor(lv_popg_cat_comparison$topic,
         levels = c("structured_popgrowth", "lotka_volterra"))

# Make a plot to compare normalized gains across categories
(lv_popg_cat_comparison_plot <-
```

```
    ggplot(lv_popg_cat_comparison) +
    geom_pointrange(aes(x = topic, y = mean_ndiff,
                        ymin = mean_ndiff-sem_ndiff,
                        ymax = mean_ndiff+sem_ndiff,
                        fill = topic, shape = topic),
                    size = 0.8, stroke = 1) +
    scale_fill_manual(values = c("grey", "orange")) +
    scale_shape_manual(values = c(22,21)) +
    scale_y_continuous(limits = c(0, 0.45), expand = c(0,0)) +
    scale_x_discrete(labels = c("Control (Structured\npopulation growth)",
                                "Lotka-Volterra Model")) +
    geom_hline(yintercept = 0, linetype = 1) +
    ecoevoapps::theme_apps() +
    ylab("normalized change\nin confidence") + xlab("") +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          axis.line.x = element_blank(),
          panel.grid.major.y = element_line(size = .05, color = "grey25")))
```



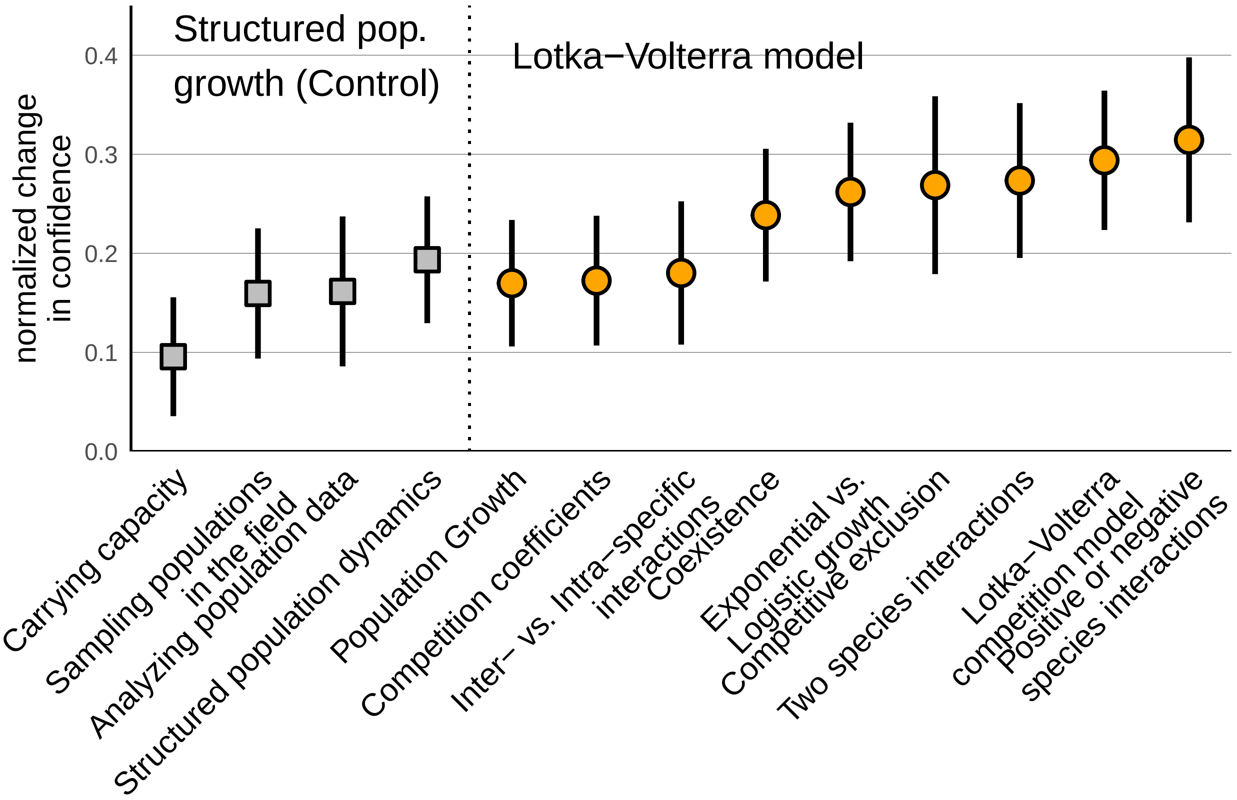
The graph above shows differences across the whole category; let's now split it up by each question within the category.

```
# Summarize across the questions within each category
lv_popg_q_comparison <-
  for_lv_analysis %>%
```

```
  mutate(topicq = paste0(topic,"_", question)) %>%
  group_by(topic, topicq) %>%
  summarize(mean_ndiff = mean(normalized_gain),
            sd_ndiff = sd(normalized_gain),
            sem_ndiff = sd_ndiff/sqrt(n())) %>%
  arrange(desc(topic), mean_ndiff) %>% ungroup %>%
  mutate(x_val = c(1:nrow(.)))
```

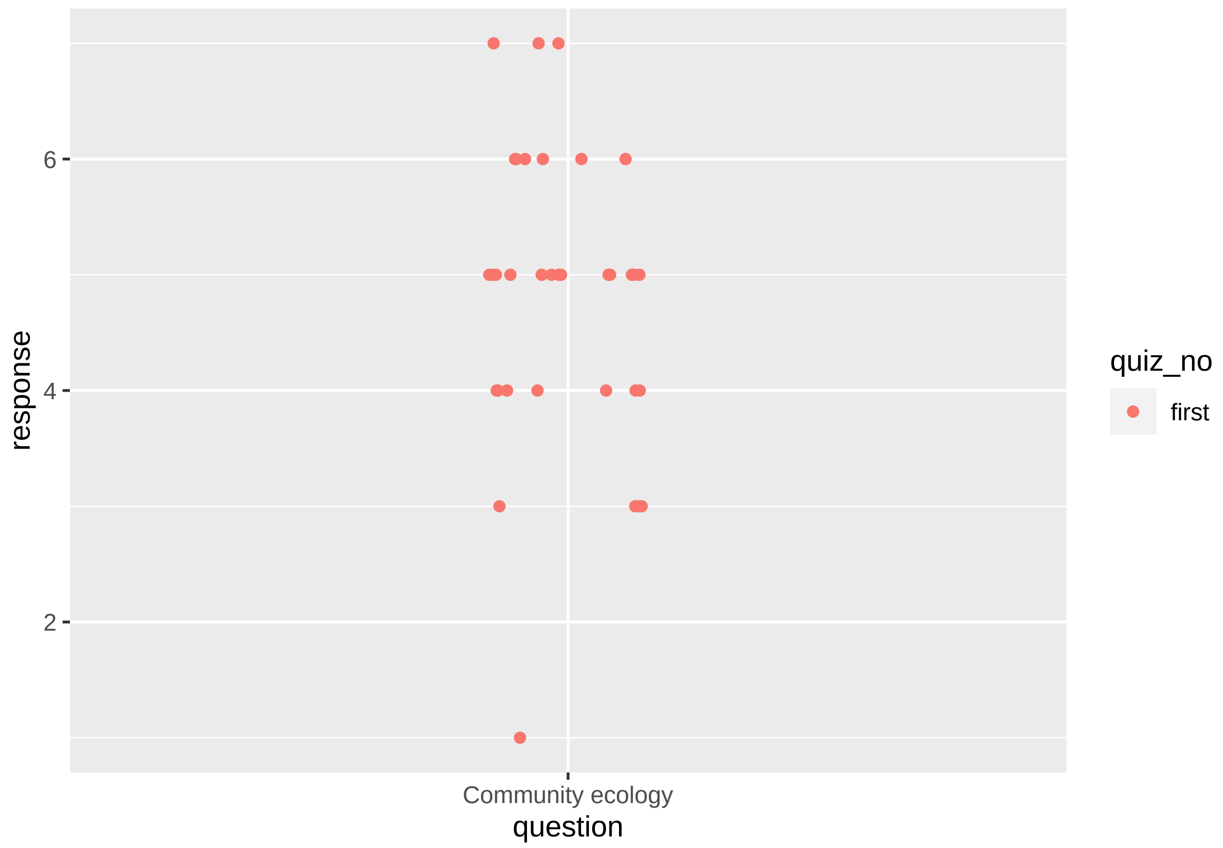
```
## ‘summarise()‘ has grouped output by ’topic’. You can override using the ‘.groups‘ argument.
```

```
# Extract the question labels and reformat
x_labels <- sub( ".*_(.*)", "\\1", lv_popg_q_comparison$topicq)
x_labels[2] <- "Sampling populations\nin the field"
x_labels[7] <- "Inter- vs. Intra-specific\ninteractions"
x_labels[9] <- "Exponential vs.\nLogistic growth"
x_labels[12] <- "Lotka-Volterra\ncompetition model"
x_labels[13] <- "Positive or negative\nspecies interactions"

(lv_popg_q_comparison_plot <-
    ggplot(lv_popg_q_comparison) +
    geom_pointrange(aes(x = x_val, y = mean_ndiff,
                        ymin = mean_ndiff - sem_ndiff,
                        max = mean_ndiff + sem_ndiff,
                        fill = topic, shape = topic), size = 1, stroke = 1) +
    scale_fill_manual(values = c("orange", "grey")) +
    scale_shape_manual(values = c(21,22)) +
    ylab("normalized change\nin confidence") + xlab("") +
    scale_y_continuous(limits = c(0, 0.45), expand = c(0,0)) +
    scale_x_continuous(expand = c(0,0),
                       limits = c(0.5,nrow(lv_popg_q_comparison)+0.5),
                       breaks = 1:nrow(lv_popg_q_comparison), labels = x_labels)+
    geom_hline(yintercept = 0, linetype = 1) +
    geom_segment(x = 4.5, xend = 4.5, y = 0, yend = Inf, linetype = 3) +
    annotate(geom = "text", x = 1,  y = 0.4, hjust = 0, size = 5,
             label =  "Structured pop.\ngrowth (Control)") +
    annotate(geom = "text", x = 5, y = 0.4, hjust = 0, size = 5,
             label =  "Lotka-Volterra model") +
    ecoevoapps::theme_apps() +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          axis.line.x = element_blank(),
          panel.grid.major.y = element_line(size = .05, color = "grey25")))
```



Now, let's evaluate whether the change in student confidence is related at all to how interested students were when they took the initial survey, i.e. before doing the activity:

```r
# Interest in community ecology BEFORE the app (quiz_no = 1)
interest_commEcol <- missouri_data %>%
  filter(topic == "interest", quiz_no == "first", question == "Community ecology")
# Take a look at the distribution of interest levels
ggplot(interest_commEcol) +
  geom_jitter(aes(x = question, y = response, color = quiz_no), height = 0, width = .1)
```



```
# From this, I feel like we can reasonably categorize an interest of 1,2,3 as "low",
# interest of 4 or 5 as "medium", and interest of 6 or 7 as "high":
interest_commEcol <- interest_commEcol %>%
  mutate(response2 = response,
         response2 = ifelse(response2 < 4, "AA_low",
                            ifelse(response2 < 6, "AB_medium", "AC_high"))) %>%
  # rename some columns for easier merging in the future, and remove some
  # columns that are no longer necessary
  select(-quiz_no, -topic, question2 = question)

# Figure out how many students indicated different levels of interest...
num_interest_level_lv <-
  interest_commEcol %>%
  group_by(response2) %>%
  summarize(num = n())

# Join the interest in with the bigger LV question confidence dataset
for_lv_interest <-
  left_join(for_lv_analysis, interest_commEcol) %>%
  filter(!(is.na(response2)))
```

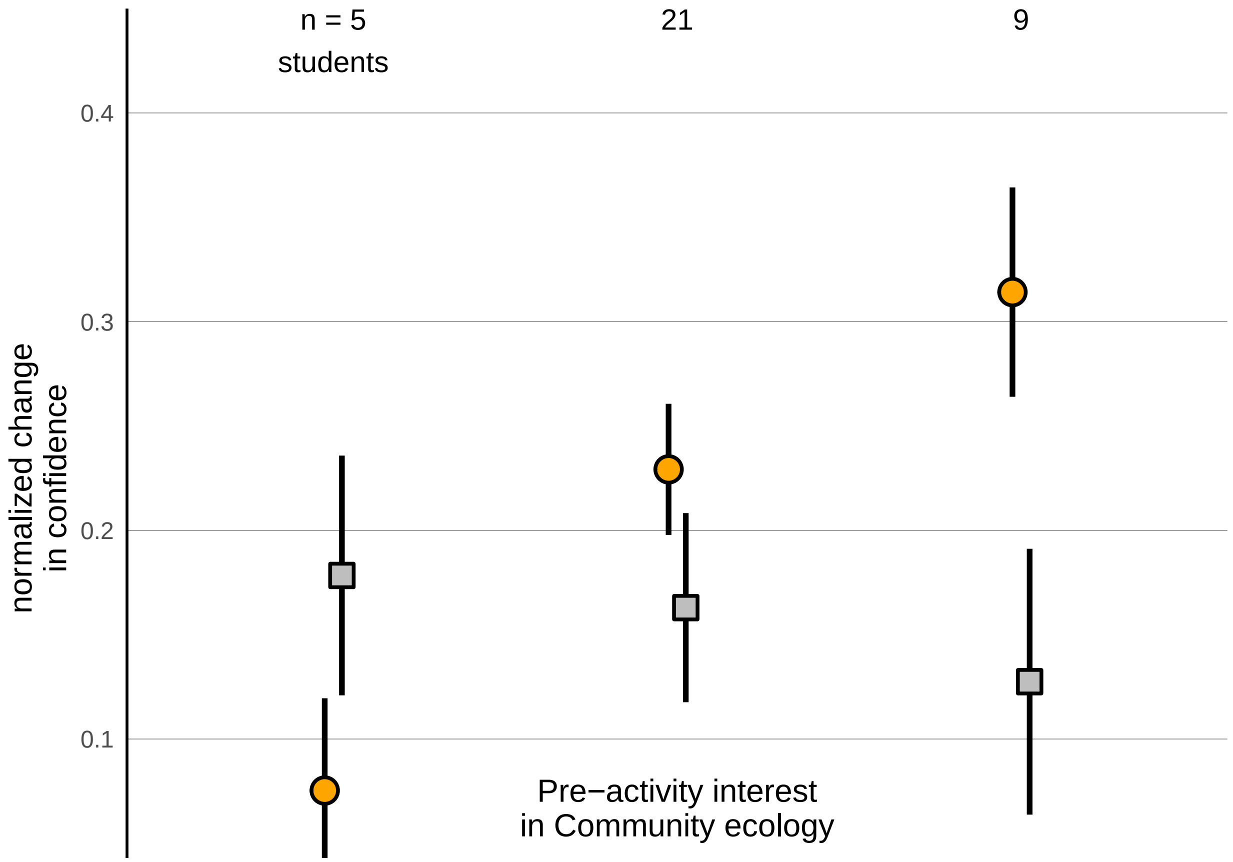
```
## Joining, by = "random_name"
```

```
for_lv_interest_sum <- for_lv_interest %>%
  group_by(topic, response2) %>%
```

```
  summarize(mean_diff = mean(normalized_gain),
            sem_diff = sd(normalized_gain)/sqrt(n()),
            total = n())
```

```
## `summarise()` has grouped output by 'topic'. You can override using the `.groups` argument.
```

```
(lv_int <-
    ggplot(for_lv_interest_sum) +
    geom_pointrange(aes(x = response2, y = mean_diff,
                        ymin = mean_diff - sem_diff, ymax = mean_diff + sem_diff,
                        fill = topic, shape = topic),
                    position = position_dodge(width = .1), size = 1) +
    scale_fill_manual(values = c("orange", "grey")) +
    scale_shape_manual(values = c(21,22)) +
    scale_x_discrete(labels = c("low\n(1, 2, or 3)", "med\n(4 or 5)", "high\n(6 or 7)")) +
    ylab("normalized change\nin confidence") +
    xlab("Pre-activity interest\nin Community ecology") +
    scale_y_continuous(expand = c(0,0), limits = c(0,.45)) +
    annotate("text", x = 1, y = Inf,
             label = paste0("n = ", num_interest_level_lv$num[1], "\nstudents"), vjust = 1) +
    annotate("text", x = 2, y = Inf,
             label = paste0(num_interest_level_lv$num[2]), vjust = 1) +
    annotate("text", x = 3, y = Inf,
             label = paste0(num_interest_level_lv$num[3]), vjust = 1) +
    ecoevoapps::theme_apps() +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          axis.title.x = element_text(margin=margin(-125,0,0,0)),
          panel.grid.major.y = element_line(size = .05, color = "grey25"))
)
```

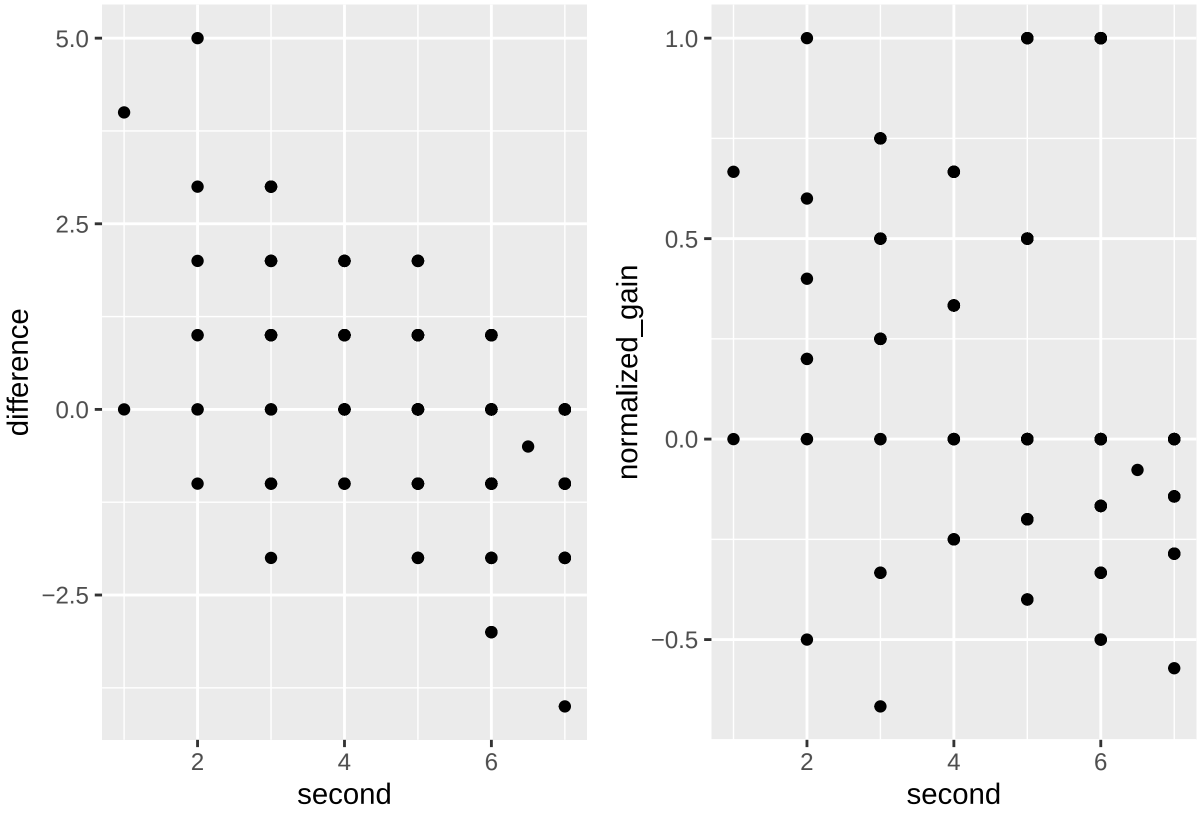


For SIR model

Now we repeat everything for the SIR model.

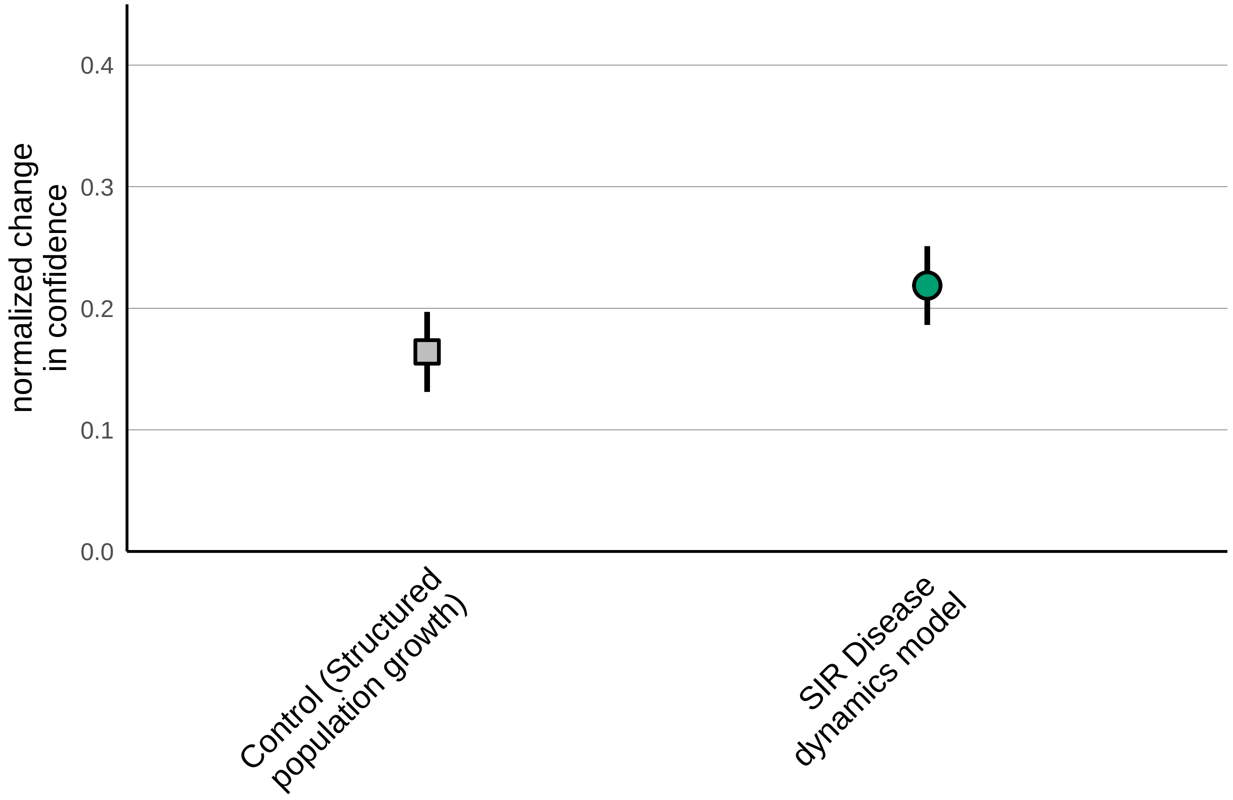
```
# For SIR analysis -------------------------------------------------------
for_sir_analysis <-
  missouri_data %>%
  filter(quiz_no %in% c("third", "second"),
         topic %in% c("structured_popgrowth", "SIR")) %>%
  pivot_wider(names_from = quiz_no, values_from = response) %>%
  mutate(difference = third - second) %>%
  filter(!(is.na(difference))) %>%
  mutate(normalized_gain = ifelse(third > second,
                                  difference/(7-second),
                                  ifelse(third == second,
                                         0,
                                         (third-second)/second)))


cor_pre_diff_si <- ggplot(for_sir_analysis) +
  geom_point(aes(x = second, y = difference))

cor_pre_ndiff_si <-
  ggplot(for_sir_analysis) +
  geom_point(aes(x = second, y =  normalized_gain))
```

```
cor_pre_diff_si + cor_pre_ndiff_si
```



```
sir_popg_cat_comparison <-
  for_sir_analysis %>%
  group_by(topic) %>%
  summarize(mean_ndiff = mean(normalized_gain),
            sd_ndiff = sd(normalized_gain),
            sem_ndiff = sd_ndiff/sqrt(n())) %>%
  arrange(topic)

sir_popg_cat_comparison$topic <- factor(sir_popg_cat_comparison$topic,
                                        levels = c("structured_popgrowth", "SIR"))

# Make a plot to compare normalized gains across categories
(sir_popg_cat_comparison_plot <-
    ggplot(sir_popg_cat_comparison) +
    geom_pointrange(aes(x = topic, y = mean_ndiff,
                        ymin = mean_ndiff-sem_ndiff,
                        ymax = mean_ndiff+sem_ndiff,
                        fill = topic, shape = topic), size = 1, stroke = 1) +
    scale_fill_manual(values = c("grey", "#009E73")) +
    scale_shape_manual(values = c(22,21)) +
    scale_x_discrete(labels = c("Control (Structured\npopulation growth)",
                                "SIR Disease \n dynamics model")) +
    ecoevoapps::theme_apps() +
```

```
    ylab("normalized change\nin confidence") + xlab("") +
    scale_y_continuous(limits = c(0, 0.45), expand = c(0,0)) +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          panel.grid.major.y = element_line(size = .05, color = "grey25")))
```

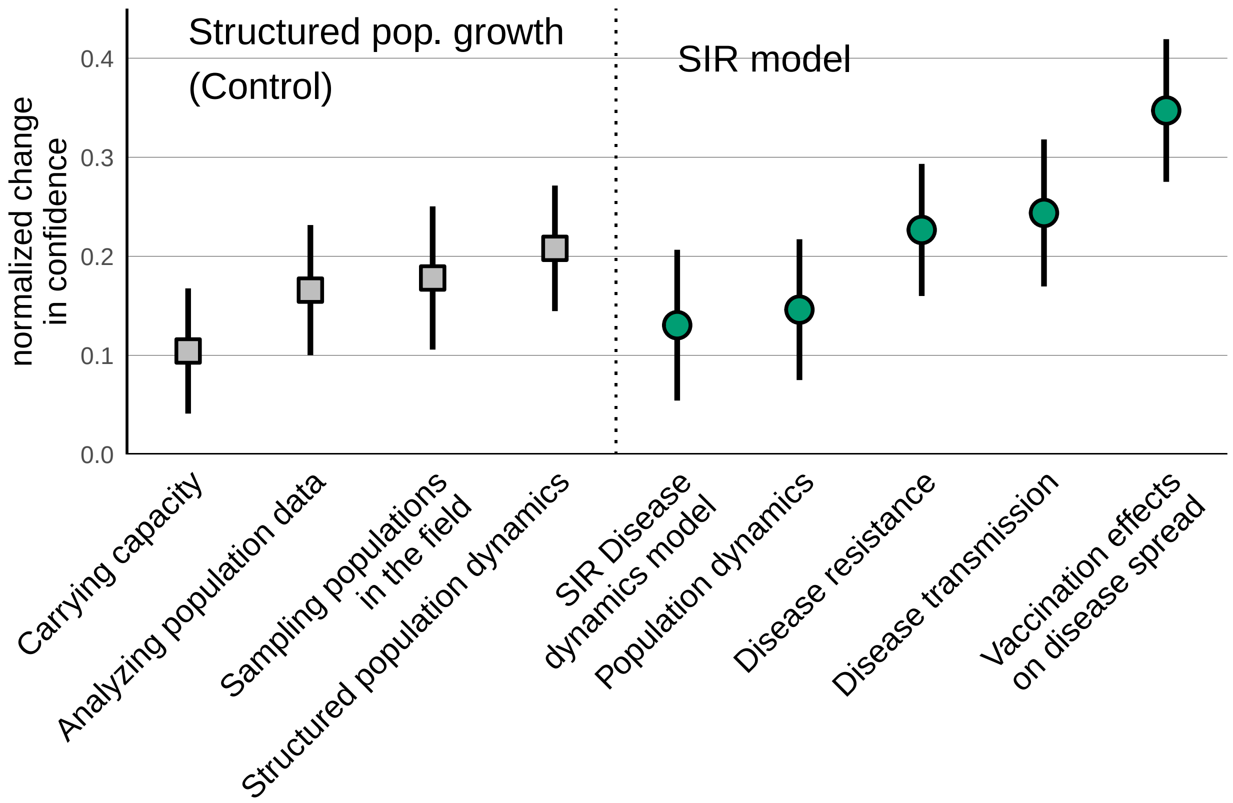


```
sir_popg_q_comparison <-
  for_sir_analysis %>%
  mutate(topicq = paste0(topic,"_", question)) %>%
  group_by(topic, topicq) %>%
  summarize(mean_ndiff = mean(normalized_gain),
            sd_ndiff = sd(normalized_gain),
            sem_ndiff = sd_ndiff/sqrt(n())) %>%
  arrange(desc(topic), mean_ndiff) %>% ungroup %>%
  mutate(x_val = 1:nrow(.))
```

```
## ‘summarise()‘ has grouped output by ’topic’. You can override using the ‘.groups‘ argument.
```

```
x_labels2 <- sub( ".*_(.*)", "\\1", sir_popg_q_comparison$topicq)
x_labels2[3] <- "Sampling populations\nin the field"
x_labels2[5] <- "SIR Disease\ndynamics model"
x_labels2[9] <- "Vaccination effects\non disease spread"
```

```r
# Make a plot to compare normalized gains across questions
(sir_popg_q_comparison_plot <- ggplot(sir_popg_q_comparison) +
    geom_pointrange(aes(x = x_val, y = mean_ndiff,
                        ymin = mean_ndiff - sem_ndiff,
                        max = mean_ndiff + sem_ndiff,
                        fill = topic, shape = topic),
                    size = 1, stroke = 1) +
    scale_shape_manual(values = c(21,22)) +
    scale_fill_manual(values = c("#009E73", "grey")) +
    ylab("normalized change\nin confidence") + xlab("") +
    scale_y_continuous(limits = c(0, 0.45), expand = c(0,0)) +
    scale_x_continuous(expand = c(0,0),
                       limits =  c(0.5,nrow(sir_popg_q_comparison)+0.5),
                       breaks = 1:nrow(sir_popg_q_comparison),
                       labels = x_labels2)+
    geom_hline(yintercept = 0, linetype = 1) +
    geom_segment(x = 4.5, xend = 4.5, y = 0, yend = Inf, linetype = 3) +
    annotate(geom = "text", x = 1,  y = 0.4, hjust = 0,
             size = 5, label =  "Structured pop. growth\n(Control)") +
    annotate(geom = "text", x = 5, y = 0.4, hjust = 0,
             size = 5, label =  "SIR model") +
    ecoevoapps::theme_apps() +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          axis.line.x = element_blank(),
          panel.grid.major.y = element_line(size = .05, color = "grey25")))
```



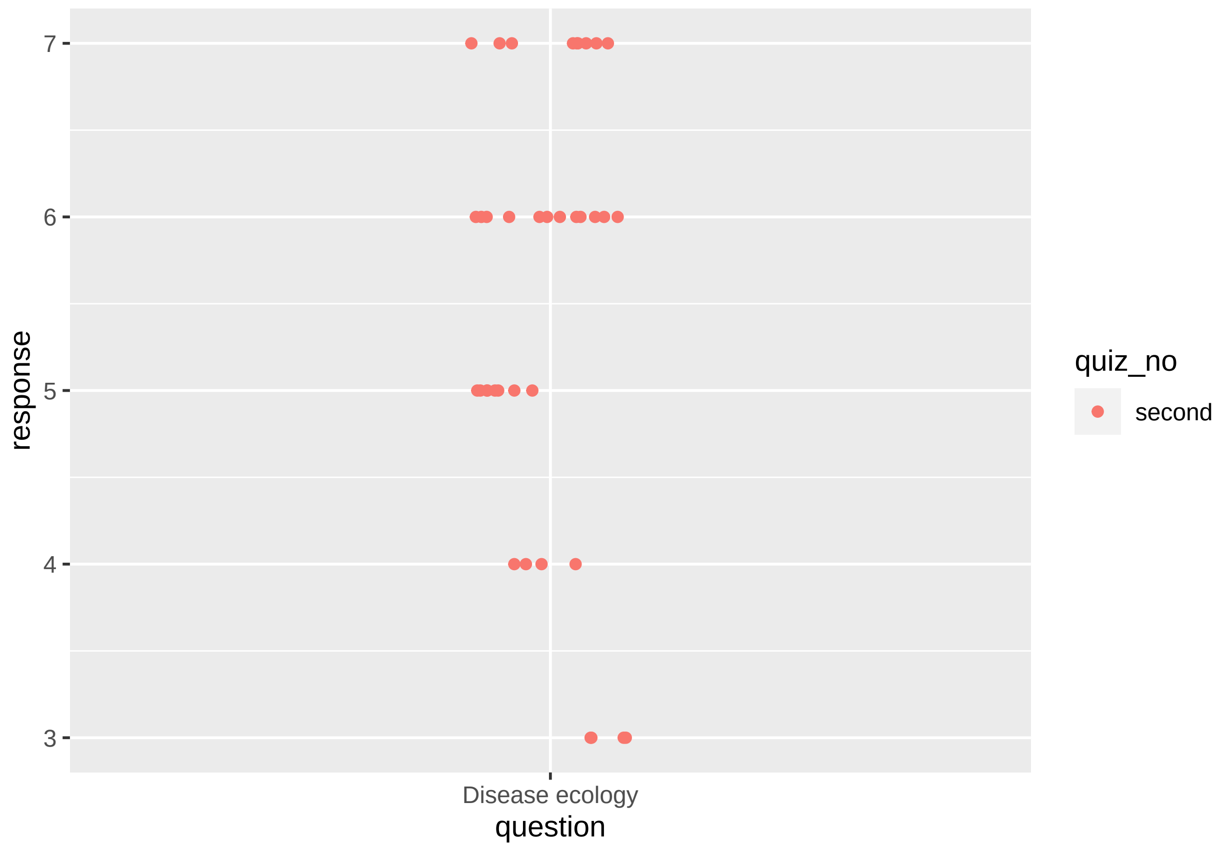
```r
# Join the interest in with the bigger LV question confidence dataset
for_lv_interest <-
  left_join(for_lv_analysis, interest_commEcol) %>%
  filter(!(is.na(response2)))
```

```
## Joining, by = "random_name"
```

```r
for_lv_interest_sum <- for_lv_interest %>%
  group_by(topic, response2) %>%
  summarize(mean_diff = mean(normalized_gain),
            sem_diff = sd(normalized_gain)/sqrt(n()),
            total = n())
```

```
## 'summarise()' has grouped output by 'topic'. You can override using the '.groups' argument.
```

```r
interest_diseaseEcol <- missouri_data %>%
  filter(topic == "interest", quiz_no == "second", question == "Disease ecology")
# Take a look at the distribution of interest levels
ggplot(interest_diseaseEcol) +
  geom_jitter(aes(x = question, y = response, color = quiz_no), height = 0, width = .1)
```



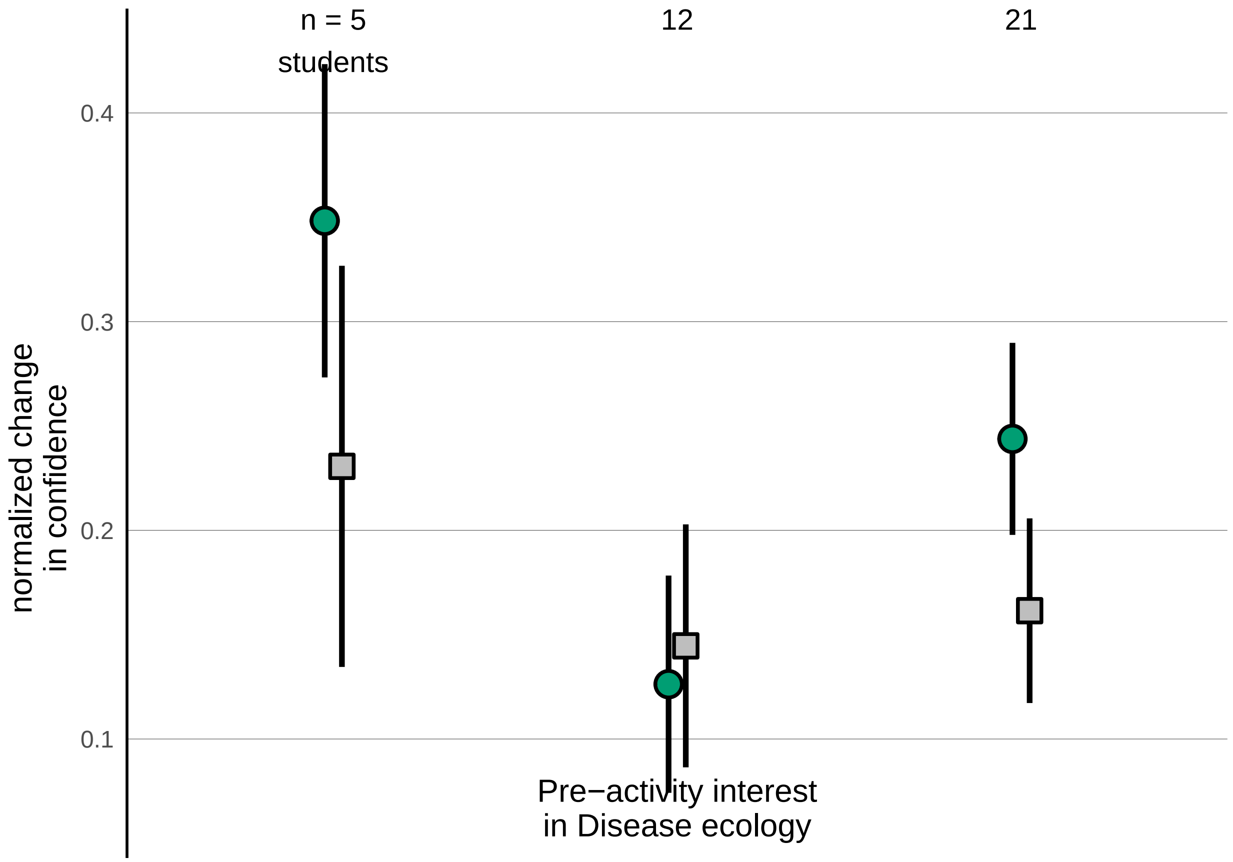
```
# From this, I feel like we can reasonably categorize an interest of 1,2,3 as "low",
# interest of 4 or 5 as "medium", and interest of 6 or 7 as "high":
interest_diseaseEcol <-
  interest_diseaseEcol %>%
  mutate(response2 = response,
         response2 = ifelse(response2 < 4, "AA_low",
                            ifelse(response2 < 6, "AB_medium", "AC_high"))) %>%
  # rename some columns for easier merging in the future, and remove some
  # columns that are no longer necessary
  select(-quiz_no, -topic, question2 = question)

# Figure out how many students indicated different levels of interest...
num_interest_level_si <-
  interest_diseaseEcol %>%
  group_by(response2) %>%
  summarize(num = n())

# Join the interest in with the bigger LV question confidence dataset
for_si_interest <-
  left_join(for_sir_analysis, interest_diseaseEcol) %>%
  filter(!(is.na(response2)))
```

```
## Joining, by = "random_name"
```

```r
for_si_interest_sum <-
  for_si_interest %>%
  group_by(topic, response2) %>%
  summarize(mean_diff = mean(normalized_gain),
            sem_diff = sd(normalized_gain)/sqrt(n()),
            total = n())
```

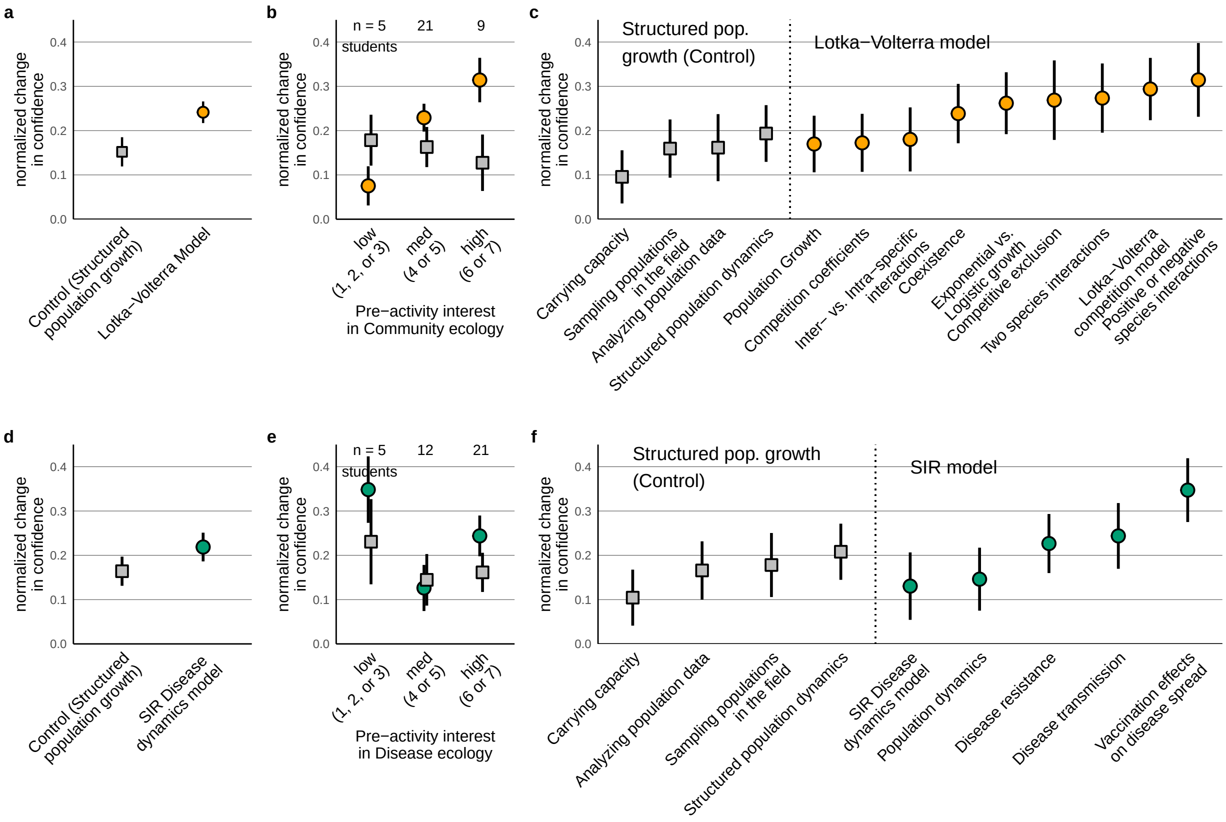
```
## ‘summarise()‘ has grouped output by ’topic’. You can override using the ‘.groups‘ argument.
```

```r
(si_int <-
    ggplot(for_si_interest_sum) +
    geom_pointrange(aes(x = response2, y = mean_diff,
                        ymin = mean_diff - sem_diff, ymax = mean_diff + sem_diff,
                        fill = topic, shape = topic),
                    position = position_dodge(width = .1), size = 1) +
    scale_fill_manual(values = c("#009E73", "grey")) +
    scale_shape_manual(values = c(21,22)) +
    scale_x_discrete(labels = c("low\n(1, 2, or 3)", "med\n(4 or 5)", "high\n(6 or 7)")) +
    ylab("normalized change\nin confidence") + xlab("Pre-activity interest\nin Disease ecology") +
    scale_y_continuous(expand = c(0,0), limits = c(0,.45)) +
    annotate("text", x = 1, y = Inf,
             label = paste0("n = ", num_interest_level_lv$num[1], "\nstudents"), vjust = 1) +
    annotate("text", x = 2, y = Inf,
             label = paste0(num_interest_level_si$num[2]), vjust = 1) +
    annotate("text", x = 3, y = Inf,
             label = paste0(num_interest_level_si$num[3]), vjust = 1) +
    ecoevoapps::theme_apps() +
    theme(legend.position = "none",
          axis.text.x = element_text(size = 12, color = "black", angle = 45, hjust = 1),
          axis.title = element_text(size = 12),
          axis.title.x = element_text(margin=margin(-125,0,0,0)),
          panel.grid.major.y = element_line(size = .05, color = "grey25"))
)
```



All the individual plots above are collated together for figure 4:

```
fig4 <- lv_popg_cat_comparison_plot +  lv_int + lv_popg_q_comparison_plot +
  sir_popg_cat_comparison_plot +   si_int + sir_popg_q_comparison_plot +
  plot_layout(widths = c(2,2,7)) + plot_annotation(tag_levels = "a") & theme(legend.position = 'none')
fig4
```



```r
ggsave("manuscript/fig4.png", fig4, width = 13, height = 9)
```

Analyze Univ. California Los Angeles data

```r
# Read in the data and reshape it to be more useful
ucla_post <- read_xls("teaching-survey/data/Survey_data.xls", sheet = 3)

ucla_post_long <- ucla_post %>%
  pivot_longer(help_lv:conf_island_dyn)
ucla_post_long <- ucla_post_long %>%
  # replace values of < 1 with 1 (some students entered 0 though they were asked not to)
  mutate(value = ifelse(value < 1, 1, value))

# Rename some values for better plotting
ucla_post_long_h <- ucla_post_long %>%
  filter(str_detect(name, "help")) %>%
  mutate(name = ifelse(name == "help_lv", "Lotka Volterra model",name),
         name = ifelse(name == "help_ib", "Island Biogeography",name))

# Change the levels for better plotting
ucla_post_long_h$name <- factor(ucla_post_long_h$name,
                                levels = c("Lotka Volterra model",
                                           "Island Biogeography"))
panels_name <- data.frame(name = as.factor(levels(ucla_post_long_h$name)))
```

Now that we have a workable data frame, we can start plotting.

```
plot_overall <-
  ggplot(ucla_post_long_h) +
  geom_histogram(aes(x = value, fill = name)) +
  scale_fill_manual(values = c("#009E73",  "darkmagenta")) +
  facet_wrap(.~name, scales = "free", nrow = 2) +
  labs(subtitle = "From a scale of 1 (not helpful) to 7 (very helpful)\nhow helpful were the interactiv
  theme_bw() +
  geom_text(data = panels_name,
            mapping = aes(x = .52, y = Inf, label = name),
            hjust   = 0, vjust   = 1.5) +
  scale_x_continuous(breaks = 1:7, labels = 1:7, limits = c(0.5,7.5)) +
  theme(panel.grid = element_blank(),
        strip.background = element_blank(),
        strip.text.x = element_blank(),
        legend.position = "none",
        plot.subtitle = element_text(size = 9.5))
```

Next, the goal is to make a similar histogram of responses to questions about individual concepts. This takes some more thinking...

```
ucla_post_long_q <- ucla_post_long %>%
  filter(!(str_detect(name, "help")))
ucla_post_long_q$name <- str_replace(ucla_post_long_q$name, pattern = "conf_", replacement = "")
ucla_post_long_q$name <- str_replace(ucla_post_long_q$name, pattern = "con_", replacement = "")
ucla_post_long_q <-
  ucla_post_long_q %>%
  mutate(name = ifelse(name == "growth", "Exponential vs. Logistic\ngrowth (LV)",name),
         name = ifelse(name == "K", "Carrying capacity\n(LV)",name),
         name = ifelse(name == "popdyn", "Population dynamics\nand time series (LV)",name),
         name = ifelse(name == "mol_eco", "Molecular ecology\n(control)",name),
         name = ifelse(name == "popgrowth", "Population growth rates\n(LV)",name),
         name = ifelse(name == "comp_int", "Competitive interactions\n(LV)",name),
         name = ifelse(name == "comp_coeff", "Competition coefficients\n(LV)",name),
         name = ifelse(name == "comp_ex", "Competitive exclusion\n(LV)",name),
         name = ifelse(name == "biogeochem", "Biogeochemical cycles\n(control)",name),
         name = ifelse(name == "coexistence", "Coexistence\n(LV)",name),
         name = ifelse(name == "lvcomp", "Lotka-Volterra\ncompetition model (LV)",name),
         name = ifelse(name == "islandbio", "Island Biogeography\n(Biogeo)",name),
         name = ifelse(name == "im_rate", "Immigration rate\n(Biogeo)",name),
         name = ifelse(name == "ext_rate", "Extinction rate\n(Biogeo)",name),
         name = ifelse(name == "island_dyn", "Mainland/Island\nDynamics (Biogeo)", name)
  )

ucla_post_long_q <-
  ucla_post_long_q %>%
  mutate(category = ifelse(str_detect(name, "LV"), "Lotka Volterra Model",
                          ifelse(str_detect(name, "control"), "Control",
                                            "Island Biogeography")))

ucla_post_long_q$name <- factor(ucla_post_long_q$name,
                                levels = c("Molecular ecology\n(control)",
                                           "Biogeochemical cycles\n(control)",
```

```
                                           "Exponential vs. Logistic\ngrowth (LV)",
                                           "Carrying capacity\n(LV)",
                                           "Population dynamics\nand time series (LV)",
                                           "Population growth rates\n(LV)",
                                           "Competitive interactions\n(LV)",
                                           "Competitive exclusion\n(LV)",
                                           "Competition coefficients\n(LV)",
                                           "Coexistence\n(LV)",
                                           "Lotka-Volterra\ncompetition model (LV)",
                                           "Island Biogeography\n(Biogeo)",
                                           "Immigration rate\n(Biogeo)",
                                           "Extinction rate\n(Biogeo)",
                                           "Mainland/Island\nDynamics (Biogeo)"))
panels_name <- data.frame(name = as.character(as.factor(levels(ucla_post_long_q$name))))
# panels_name$name[11] <- "Lotka-Volterra\ncompetition model\n(LV)"
plot_specifics <-
  ggplot(ucla_post_long_q) +
  geom_histogram(aes(x = value, fill = category)) +
  facet_wrap(.~name,  nrow = 3) +
  theme_bw() +
  # geom_text(data = panels_name,
  #   mapping = aes(x = .52, y = Inf, label = name),
  #   hjust = 0, vjust   = 1.5, size = 3.25) +
  scale_fill_manual(values = c("grey", "darkmagenta", "#009E73")) +
  theme(panel.grid = element_blank(),
        strip.background = element_blank(),
        #strip.text.x = element_blank(),
        legend.position = "none",
        plot.subtitle = element_text(size = 9.5)) +
  scale_x_continuous(breaks = 1:7, labels = 1:7, limits = c(0.5,7.5)) +
  labs(subtitle = "From a scale of 1 (not helpful) to 7 (very helpful), please rate how much\nthe intera
```

And now we can put these two sets of histograms together:

```
fig5 <- plot_overall + plot_specifics + plot_layout(widths = c(2,5)) +
  plot_annotation(tag_level = "a")
# fig5
ggsave("manuscript/fig5.png", fig5, width = 12, height = 6)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 4 rows containing missing values (geom_bar).

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 30 rows containing missing values (geom_bar).
```