**Bi592 Theoretical Ecology**  
**Course Syllabus - Fall 2015**

**Course Times & Location**  
Tuesday & Thursday 10-11:50 am  
Cordley 3030

**Instructor:**  
Dr. Mark Novak  
mark.novak@oregonstate.edu

**Course web site:**  
Blackboard ([http://my.oregonstate.edu](http://my.oregonstate.edu))

**Office Hours:** Wednesday 10 am, or by appointment

**Course Credits:**  
4

**Prerequisites:** Motivation & graduate standing  
or by instructor approval

**Description:** This course entails a quantitative treatment of the central concepts of theoretical ecology. Emphasis is on models of single populations and multi-species interactions, and the integration of models with data. Topics include discrete- and continuous-time models of population growth, stochastic and deterministic processes in unstructured populations, the sustainability of harvested populations, numerical and analytical investigations of population and community invasibility and stability, and an introduction to model-fitting and comparison in an information-theoretic framework.

**Learning Outcomes:** By the end of this course you will be able to:  
(i) independently interpret the classic and modern literature of theoretical ecology,  
(ii) apply analytic and simulation-based approaches in evaluating ecological questions of your own interest and communicate them to an audience of your peers, and  
(iii) have a working knowledge of R and an introductory knowledge of Mathematica.

**Readings:** There is no required textbook. Readings are posted on the course website. Assigned readings are expected to have been read before the topic is covered in class. (In particular, you are required to have read all the discussion papers so that you can participate in the discussion.) I have also posted additional recommended readings. These are optional and may be of historic relevance, provide a different view, or just be good additional material.  
Errata for Ted Case's «Illustrated Guide to Theoretical Ecology» book (which we will use frequently) may be found at: [http://www.nceas.ucsb.edu/BookCase/known-typos.html](http://www.nceas.ucsb.edu/BookCase/known-typos.html)

**Grading:** Grades (because I have to) will be derived as follows...

<table>
<thead>
<tr>
<th>Attendance &amp; Participation</th>
<th>70%</th>
<th>Leading a paper discussion</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem set completion</td>
<td>10%</td>
<td>Online quizzes</td>
<td>5%</td>
</tr>
<tr>
<td>Problem set reviews</td>
<td>5%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Attendance & Participation:** Showing up, staying awake, participating in discussions, asking questions, pointing out all the mistakes I make when writing equations on the whiteboard, etc., is all you have to do. What you get out of the class is up to you.

I recognize that field-work and other conflicts will prevent some of you from being able to attend all classes. If you tell me ahead of time what weeks you expect to be gone (i.e. have field-work conflicts), then we’ll work out a solution regarding the assignments.

**Problem sets:** Problem sets will be posted to Blackboard over the course of the quarter (in the Assignments section). We will be starting these problem sets together in class. Completed problem
sets will be due by the Friday of the following week (i.e. you will typically have a 1 to 1-1/2 weeks to complete each). Typically, each problem set will consist of two or more parts. You should be able to tackle Part A the same day that the homework is assigned; Part B, however, might require attending and having done the readings for a subsequent class. Problem sets will not be graded. Rather, you will receive credit simply for completing them. However, each of your codes will be “reviewed” by your peers (see below).

**Submitting your problem sets:** Problem set answers are to be uploaded on Blackboard. **You will have to upload the R code for your problem set twice.** (Once to get credit for completing the assignment, and once to make your code available to a reviewer.)

Step 1. Name your R code file in the following format: “ProblemSet#-LastName.R” (e.g., 2-Novak.R)

Step 2. Upload your file to the relevant **Assignment** on Blackboard (to receive credit).

(a) Click on **Browse My Computer** between the two text boxes to select your file. You can leave the two text boxes blank.

(b) Click **Submit** (an important step that is forgotten all too often).

Step 3. Upload your file to the **File Exchange** (to make it available to review):

(a) Click **Tools** on the course menu.

(b) Click **Groups.** Sign-up for the relevant problem set.

(c) Click **File Exchange, click Add File** (top-left corner), and upload the file.

**Collaborating on problem sets:** I strongly encourage you to work on the problem sets with others. My only request is that the scripts you hand in actually reflect your level of understanding, even if you have to add commentary to indicate what you didn’t understand. That is, don’t just copy blindly from others, but rather figure out what you do and don’t understand.

To make group work easier, I will set up a blog for you to communicate on our website. (Blogs are like Discussion boards, but are easier to work with.) If you want to receive an email when someone responds to your blog post (so you don’t have to keep checking Blackboard needlessly), follow these instructions:

Go to the **Notifications Dashboard** (upper left corner on the home page of Blackboard (My Oregon State tab). Find the **Needs Attention** module, and click **Edit Notification Settings.** Select a course, or choose **Bulk Notifications settings > Courses I am Taking.** In the Notification column, make sure %60Unread Blog Posts%60 is clicked On and that the Email box is checked.

**Problem set reviews:** The objective of these is to provide an additional opportunity for you to learn tips, tricks, and coding best practices from each other. They should not take more than 15-30 minutes for you to complete. Your reviews will not be graded, but you will receive credit for completing them.

The day after each problem set is due, you will be assigned another student’s R code. (Who you are assigned will be completely randomized, so you may end up reviewing the same person’s code more than once over the quarter.) Your “review” can be short or long, but must consist of at least one of the following:

(a) A suggestion for an alternative approach (with explanation or an example) for implementing a step in the code (e.g., how to avoid looping, the use of a built-in function, etc.).
(b) A suggestion regarding coding best practices (e.g., more informative annotation, simpler code layout, etc.)

(c) A description of something new that you learned from the code.

**Submitting your reviews:** Again, you’ll have to upload your review **twice**. (Once for the credit, and once for the code author to see your comments.)

Step 1. Save your review in a *.txt format and name it as follows:

“Review-ProblemSet#-AuthorLastName-YourLastName.txt”

For example,

If the original file was: 2-Smith.R,

then name your review file: Review-2-Smith-Novak.txt

Step 2. Upload your file to the relevant Assignment on Blackboard (to receive credit).

Step 3. Upload your file to the File Exchange (to make your review available to the author).

**Paper discussions:** In the first week you will each have the opportunity to choose a paper on which to lead a discussion. We’ll typically reserved 45min to 1hr of our class time for the discussion. All of us will be required to have read the papers. Each of the discussion leader will be responsible for providing a 1 to 2 page summary of the paper, in bullet-point form, to the rest of us. (Our code for the office photocopier is: ). Discussion leaders should also make themselves a general outline of what they want to discuss and how they might see the discussion going; a loose game plan so to speak.

**Online quizzes & quiz evaluations:** I will post or handout a number of “quizzes” to take at various points in the course. They’re meant to be no-pressure! They’re not graded, should take less than 10-15 minutes and are simply meant for you to self-assess your grasp of the material and help me to ensure that we’re all up to speed and on the same page. Hand in a paper copy of your quiz a week after its assigned. To receive credit for taking a quiz fill out a “Quiz Evaluation” survey. I’ll post an answer key at the end of the week.

**Computers:** Computers are a necessary part of this course and you will need one on which R is installed. Let me know if you do not have a laptop that you can bring to class. You’ll need one occasionally (of which I’ll warn you a class ahead).

**Programming languages & script editors:**

*Simulations:* We’ll be using R for most of the simulations in this class. See the “Getting started with Rrrrr.pdf” document on Blackboard for information on installation, or go to [http://www.r-project.org/](http://www.r-project.org/) where you can find many more "Starting R" manuals.

**Symbolic mathematics:** We’ll be doing a bit of single-variable calculus and a fair bit more algebra. For all but the simplest situations this can get tedious. We’ll be using Mathematica to help us out with this, or you can use some open source software called Sage.

OSU has an unlimited site license for Mathematica. (Great, but you can only use it while you’re here at OSU.) There’s a few steps to follow, the first of which is to sign-up and request an activation key: [https://user.wolfram.com/portal/requestAK/be753b9afbb7c78e4bcdefbbaad8dfba4](https://user.wolfram.com/portal/requestAK/be753b9afbb7c78e4bcdefbbaad8dfba4)
**Script editors:** You can type commands into any of the above software and save everything (i.e. your code and output) as a file. Frequently, however, it is far simpler and more convenient to write your code in a separate text editor and save it as a text file. This aids in organizing your codes when you start accumulating them. It also means you can easily run your code again when your input data changes. You can simply copy and paste your code from the text file into the command prompts to perform the calculations again. Any text editor will suffice (e.g., *TextEdit, Notepad*, etc.).

However, many other editors offer syntax highlighting whereby the different commands of your code (e.g., functions, annotation comments, printed text) will be highlighted in different colours. This aids in programming. They also offer keyboard shortcuts for sending your code to *R* (rather than having to copy-paste). Check out: [http://www.sciviews.org/_rgui/projects/Editors.html](http://www.sciviews.org/_rgui/projects/Editors.html). That site also has information for installing the syntax highlighting. I use a simple one called *TextWrangler* (on a Mac), but many more exist: [http://en.wikipedia.org/wiki/Comparison_of_text_editors](http://en.wikipedia.org/wiki/Comparison_of_text_editors). There are also all-built-in apps like *RStudio* that some people like.

**Academic Honesty and Student Conduct:** Students are expected to adhere to the OSU Student Conduct Regulations described at [http://studentlife.oregonstate.edu/studentconduct/offenses-0](http://studentlife.oregonstate.edu/studentconduct/offenses-0). Cheating or plagiarizing by students is subject to the disciplinary process outlined in the Student Conduct Regulations. Behaviors disruptive to the learning environment of others will not be tolerated.

**Statement Regarding Students with Disabilities:** Oregon State University is committed to student success; however, we do not require students to use accommodations nor will we provide them unless they are requested by the student. The student, as a legal adult, is responsible to request appropriate accommodations. The student must take the lead in applying to Disability Access Services (DAS) and submit requests for accommodations each term through DAS Online. OSU students apply to DAS and request accommodations at our [http://ds.oregonstate.edu/gettingstarted](http://ds.oregonstate.edu/gettingstarted) page.

**Disclaimer:** I reserve the right to change the schedule, policies, and assignment in this course due to extenuating circumstances or by mutual agreement between the instructors and students.
<table>
<thead>
<tr>
<th>Date</th>
<th>Class</th>
<th>Topic</th>
<th>Due</th>
<th>Discussion Papers*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td>10.1</td>
<td>1 Philosophy of modeling</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.3</td>
<td>2 Density-independent deterministic growth</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 2</td>
<td>10.8</td>
<td>3 Intro to R &amp; Problem set #1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.10</td>
<td>4 Density-independent stochastic growth</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 3</td>
<td>10.15</td>
<td>5 Paper discussion #1 &amp; Problem set #2</td>
<td>Levins ’66, Hilborn &amp; Mangel Chp2, May ’04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.17</td>
<td>6 Density-dependent growth</td>
<td>PS #1</td>
<td></td>
</tr>
<tr>
<td>Week 4</td>
<td>10.22</td>
<td>7 Model-fitting</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.24</td>
<td>8 Maximum likelihood &amp; AIC, Paper discussion #2 &amp; PS #2 cont.</td>
<td>Review #1</td>
<td>Anderson et. al ’00, Stephen et al. ’05</td>
</tr>
<tr>
<td>Week 5</td>
<td>10.29</td>
<td>9 Stability analysis - single species</td>
<td>PS #2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10.31</td>
<td>10 Paper discussion #3 &amp; Problem set #3</td>
<td>May ’74, Hassell et al. ’76</td>
<td></td>
</tr>
<tr>
<td>Week 6</td>
<td>11.5</td>
<td>11 Competition</td>
<td>Review #2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11.7</td>
<td>12 Consumer-resource</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 7</td>
<td>11.12</td>
<td>13 Network modules</td>
<td>PS #3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11.14</td>
<td>14 Pulse perturbations 1 &amp; PS #4/5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 8</td>
<td>11.19</td>
<td>15 Perturbations 2 &amp; Paper discussion #4</td>
<td>Review #3</td>
<td>Ives &amp; Carpenter ’07, May ’72, Allesina &amp; Tang ’13</td>
</tr>
<tr>
<td></td>
<td>11.21</td>
<td>16 Press perturbations</td>
<td>PS #4/5</td>
<td></td>
</tr>
<tr>
<td>Week 9</td>
<td>11.26</td>
<td>17 Tipping points &amp; PS #6</td>
<td>Review #4/5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11.28</td>
<td>&quot;Thanksgiving break&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 10</td>
<td>12.3</td>
<td>18 Estimating &quot;interaction strengths&quot;</td>
<td>PS #6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12.5</td>
<td>19 Paper discussion #5 &amp; Parting thoughts</td>
<td>Review #6</td>
<td>Aber ’97, Ellner ’06, Sand-Jensen ’07, Fawcett ’12</td>
</tr>
</tbody>
</table>

* See “Z599-TheoreticalEcology-Readings” in the Course Documents folder on Blackboard for PDF’s of the weekly reading assignments and additional recommended readings.