**OCN 750 Topics in Biological Oceanography**

**Introduction to Programming and Statistics in R**

**Syllabus (subject to change)**

Instructor: Anna B. Neuheimer, office MSB 614, annabn@hawaii.edu, 956-2613

**Course Motivation and Goals**

More and more scientists are choosing to learn a programming language to handle all aspects of data analysis (exploring, summarizing, statistical analyzing, visualizing) as well as mathematical modeling tasks (e.g. modeling everything from physiology to ecosystems). Programming languages offer many benefits over “point and click” options. Programming languages allow work to be explicit & documented, promoting experimentation and exploration, and recording work-flow from start to finish. The latter aspect allows for quality control (as everything is documented, mistakes are easier to find) and the ability to simply rerun analyzes when data are updated. Moreover, the recording of processes in a programming language allows one to automate sequences of tasks that are often repeated - basic syntax and techniques can be applied to many different problems. In this way, data analysis is accelerated when a programming language is learned.

Learning one programming language helps you learn others. But where do you start? A great place is with the R programming language (http://www.r-project.org/), where benefits include:

- **R’s free and open-source:** There is no need to keep track of expensive licenses, it’s available to all (e.g. Mac, Windows, Linux), and the source code is explorable and editable.
- **The R community is large and strong:** This provides a great community for learning, and also means new analytical tools are constantly being developed and shared as packages (http://cran.r-project.org/web/packages/). Many discipline-specific packages are available (e.g. for climate research, oceanography, fisheries) and learning R means leveraging the expertise of these experts.
- **R’s great for statistics:** R’s foundation is one of statistical analysis. Designed by statisticians, the resulting language is intuitive for statistical analysis and the flexible statistical analysis toolkit is large.
- **R’s great for graphics:** Graphics are fully customizable and a number of packages are available that allow for the creation of publishable quality graphics.
- **R can interface with other languages (e.g. Python, C/C++)** when tasks require increased speed and computational power.

In this course, you will learn how to use R for effective data analysis. The course will begin with an introduction to programming languages and programming in R. You will learn basic syntax, coding grammar & “etiquette” and a range of vocabulary to aid in data analysis. Next we will walk through examples of using R for statistics. For the most part, examples will be geared towards research problems encountered in biology and oceanography, though those in other disciplines will find the methods transfer to other areas as well. This second part of the course will be tailored to the wants and needs of participants, with examples from students’ own data whenever possible. We will not have time to cover all R has to offer regarding statistical analysis, but the focus on students’ own statistical needs will ensure efforts are useful. In addition, examples will be generalized to demonstrate how similar tests can be applied to other datasets/areas.

The path to success in learning any programming language includes i) practice and ii) a community of fellow users accessible for trouble-shooting. This class will provide you with both.
Student Learning Outcomes: At the end of this course, students will be able to:

- List motivation for learning a programming language
- Access online resources for R and import new function packages into the R workspace
- Import, review, manipulate and summarize data-sets in R
- Explore data-sets to create testable hypotheses and identify appropriate statistical tests
- Perform appropriate statistical tests using R
- Create and edit visualizations with R

Requirements/Prerequisites: This course is aimed at graduate students. Students are expected to

- have no previous programming experience (but GREAT if you do!)
- have some basic statistical knowledge and a desire to learn more
- be motivated enough to work through the learning curve associated with learning any programming language.

Course Structure: The course will include 2 hours of combined lecture-computer tutorial per week. PC computers will be available but feel free to bring your own laptop (Mac, Linux or PC). Enrollment is limited to 15.

Class Times: Tuesdays, 09:30 to 11:30 in MSB 314.

Grading Scheme: Course can be taken CR/NC. Course grade will be based on weekly/bi-weekly assignments that will practice programming and statistics topics from class.

Reading/Texts: There is no required reading for this course. Programming and statistical resources will be distributed in class.

Office hours: To be determined

Tentative Schedule
Week 1  Course overview:
         Introduction to programming
         R Installation
         Basic syntax

Week 2  Introduction to R:
         Scripts – Documenting, commenting & sharing code
         Functions
         Packages
         Generating and importing data
         Data types – vectors, matrices, dataframes, lists

Week 3  Data manipulation I:
         Indexing and accessing variables
         Exploring – summarizing, sorting, sub-setting, merging

Week 4  Data manipulation II:
         Conditional functions – if, for & while loops
         Apply functions and vectorization
         Exporting – writing to files
Debugging & optimizing

Week 5  Visualization I:
          Plotting
          Editing symbols, colors and sizes
          Adding text, lines, multiple axes

Week 6  Visualization II:
          Specialized plots - pie charts, bar charts, boxplots, 3-D plots
          Plotting on multiple panels
          Map-making

Week 7  Statistics I:
          What stats do I need?
          How do I find them?

Week 8  Statistics II:
          Looking for relationships: Linear models

Week 9  Statistics III:
          Looking for relationships: GLMs, GAMs

Week 10 Time-series Analysis I:
          Discontinuity analysis
          Detrending
          Smoothing

Week 11 Time-series Analysis II:
          Spectral analysis

Weeks 12-16 Case studies:
          Depending on participant interest, topics may include:
          - Classification - estimating groups or clusters
          - Ordination - projecting variables on vector space
          - Model comparisons
          - Introduction to ecological modeling (individuals, populations, ecosystems)
          - Programming for parallel processing
          - Other open-source tools (Python, version control)
          - R-Matlab translation
          - Geostatistical analysis
          - Specific R packages for biological oceanography
          - R for fisheries management
          - Bioinformatics
          - Diversity modeling
          - etc.