The objective of this section is to review the syllabus and the `R` environment. With remaining time, I will introduce the Github repository for these sections, submit basic code puzzles, and make a shameless pitch for the `ecohack.org` conference.

Download `R`: The download of `R` will vary by operating system, but it will begin here in any event:

`cran.r-project.org`

The online documentation and installer routines are comprehensive. If you are new to `R`, then it might make sense to use the Mac or Windows distribution, along with the built-in editor to write and evaluate code. For the tech-oriented, the Linux distribution is very flexible; and I'd use Emacs with the ESS package for editing. If you are interested in using the Linux distribution and are having trouble with the setup, please see me.

In order to download specific packages that are not bundled with the base distribution of `R`, such as the `foreign` package, you'll enter the following commands to install and load the package:

```
install.packages("foreign")
library(foreign)
```

Once `foreign` is loaded, you'll have access to all of its constituent functions, including `read.dta` which will convert a Stata data file into an `R` data frame.

Github repo: I will primarily use bSpace to disseminate the section notes. You don't need to know anything about Github to productively and successfully engage in this section. The final version of the notes will be posted at least one week before section as a PDF on Blackboard. That said, if you want to review an advanced, rough copy of the notes, you can browse the Github repo for this section, and the address is found in the syllabus. Github is an immensely useful collaborative coding site. You will find a full revision history of the code and notes; and if you see any problems, you can submit a patch. This will provide a gentle but useful introduction to the type of open source project that is common in the Bay Area tech industry. Many of the facilities developed for collaborative coding are incredibly valuable for joint research projects.

Linear algebra puzzles: These notes will provide a code illustration of the Linear Algebra review in Chapter 1 of the lecture notes. Don't worry if you can't solve these puzzles. Come back to them later, once we have gone over `R` code in more detail. There are many correct ways to solve these puzzles. We will go over a few solutions in section.

1. Let $\mathbf{I}_5$ be a 5×5 identity matrix. Demonstrate that $\mathbf{I}_5$ is symmetric and idempotent using simple functions in `R`.
2. Generate a 2×2 idempotent matrix $\mathbf{X}$, where $\mathbf{X}$ is not the identity matrix. Demonstrate that $\mathbf{X} = \mathbf{XX}$.
3. Generate two random variables, $\mathbf{x}$ and $\mathbf{e}$, of dimension $n = 100$ such that $\mathbf{x}, \mathbf{e} \sim N(0, 1)$. Generate a random variable $\mathbf{y}$ according to the data generating process $y_i = x_i + e_i$. Show that if you regress $\mathbf{y}$ on $\mathbf{x}$ using the canned linear regression routine `lm()`, then you will get an estimate of the intercept β_0 and the coefficient on $\mathbf{x}$, β_1, such that $\beta_0 = 0$ and $\beta_1 = 1$.
4. Show that if $\lambda_1, \lambda_2, \ldots, \lambda_5$ are the eigenvectors of a 5×5 matrix $\mathbf{A}$, then $\text{tr}(\mathbf{A}) = \sum_{i=1}^{5} \lambda_i$.