BE 434/535 Biosystems Analytics

The field of "bioinformatics" is biology plus data science. This course assumes you have some understanding of the Unix command line and some programming experience.

Ocelote (UA HPC)

Given that our class will have students on a variety of operating systems (Windows, OSX, Linux), we will use the HPC (high performance computing) cluster at the University of Arizona for our work.

To gain access to Ocelote, students must:

1) Have a terminal application ("Terminal" on OSX, Gitbash on Windows)
2) Sign up for UA's NetID+ (https://netid-plus.arizona.edu/)
3) Be sponsored (https://portal.hpc.arizona.edu/portal/)

Once you have these, open a terminal and type:

```
ssh <NetID>@hpc.arizona.edu
```

You will be prompted for additional authentication for NetID+. If all goes well, you should see something like this:

```
Last login: Sat Jan  5 07:53:30 2019 from ip72-200-123-88.tc.ph.cox.net
This is a bastion host used to access the rest of the environment.

Shortcut commands to access each resource
-----------------------------------------
Ocelote:                El Gato:
$ ocelote               $ elgato
```

Or you may see this (e.g., if you have enabled the menu with the `menuon` command):

```
Last login: Sat Jan  5 07:53:30 2019 from ip72-200-123-88.tc.ph.cox.net
This is a bastion host used to access the rest of the environment.

Shortcut commands to access each resource
-----------------------------------------
Ocelote:                El Gato:
$ ocelote               $ elgato
```

SSH Keys

If you would like to avoid the 2-factor authentication, then copy your SSH private key to the target system like so:

1) On your local machine and `cd ~/.ssh`. If you do not have one, then run `ssh-keygen`.
2) Copy the contents of the `id_rsa.pub` file (the "public" part of your key). If you do not have one, then run `ssh-keygen`.
3) Login to the target system and `cd ~/.ssh`. If you do not have one, then run `ssh-keygen`.
4) Edit the `authorized_keys` file (e.g., with `nano`) and paste in the public key.
5) Set the permissions with `chmod 600 authorized_keys`
6) Test your login from your local machine.

Git(hub)

You will use the `git` source code management program to gain access to the course materials as well as turn in your assignments. Once you are on the Ocelote (or a similar Unix platform), it's quite likely that Git is already installed. Check the version like so:

```
[hpc:login3@~]$ git --version
git version 2.2.2
```

All the code examples presented here can be found at:

```
https://github.com/hurwitzlab/biosys-analytics
```

Visit the above URL in a browser to look around. You will need to create a Github account (it's free), and then "fork" our repository into your own account by clicking the "Fork" button in the upper-right corner. Then you can "clone" the repo into your own account. I suggest you add your public SSH key (see "SSH Keys" above) into your Github "Settings/SSH and GPG Keys" so that you can more easily push and pull into your repositories. You'll need to add a key from each machine you intend to work from, i.e., both your laptop and Ocelote.

Now you can copy the course repo to your machine(s) like so:

```
[hpc:login3@~]$ git clone git@github.com:kyclark/biosys-analytics.git
Cloning into 'biosys-analytics'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 1), reused 12 (delta 0), pack-reused 0
Receiving objects: 100% (16/16), 104.71 KiB | 0 bytes/s, done.
```

```
Resolving deltas: 100% (1/1), done.
Checking connectivity... done.
```

To stay together, you will add the class Github repo as an "upstream":

```
git remote add upstream https://github.com/hurwitzlab/biosys-analytics.git
```

When you need to get new content, do this:

```
git pull upstream master
```

About The Author

I went to college (the University of North Texas, 1990) thinking I might study music and become a professional drummer. I decided against that particular career but didn't have an alternative. I changed my major a couple of times (business, communications) before deciding on an English literature degree just so I could finish Never did programming cross my mind, and biology was my most loathed science.

After college, I looked for a job that 1) would be interesting and 2) people would pay me to do. At my first job (1995), I learned Microsoft's Access database on Windows 3.1 and created my company's first website by writing HTML into Notepad and using a Windows FTP client to upload to our ISP. The die was cast. My next job was writing a technical manual for a piece of software written in VisualBasic. I was offered a position to learn VB and support the program and another Access program. I went on to learn another database (DBase IV) and language (Delphi). My next job was developing Windows desktop applications in Delphi/Interbase, but I was itching to get into web applications. So my next job (c. 1998) was writing VBScript in Microsoft ASP with SQLServer as a database. At this point, I was fed up with Microsoft and rediscovered Unix.

I discovered that my ISP offered "shell" accounts on their servers accessible by "telnet." In college I had gone to the computer labs where I used Unix programs like "pine" for email and "talk" for chatting, and so I remembered how to get around a shell. I started reading more about Unix on the Internet and how people were using the Perl programming language with CGI (common gateway interface) to create interactive web pages. The more I learned about Unix and Perl and "open source/free" software, the more I realized I'd found my tribe. At my next job at boston.com (1999) I moved into developing web apps on Linux platforms using the Apache web server with the MySQL database and Perl (the "LAMP" stack).

Around 2001, I saw that a very celebrated Perl developer named Lincoln Stein was looking to hire people. I got hired to work on a comparative plant genomics database called "Gramene." Lincoln was a very important character in a fairly new field called "bioinformatics" (cf. "How Perl Saved the Human Genome

Project") and he ran a research lab at Cold Spring Harbor Laboratory in Cold Spring Harbor, NY. Lincoln hired me to write a web-based visual comparative map application (CMap, PMID: 19648141) to augment existing web genome browsers like the UCSC browser, the Ensembl browser, and Lincoln's own Gbrowse. This was my entree into the world of biology and genomics. Around 2004, Lincoln hired Bonnie Hurwitz who left a few years later to earn her PhD from the University of Arizona. In 2014, Bonnie set up her new lab at the Universit of Arizona and hired me.

Outside of science and coding, I also enjoy biking, cooking, and playing music.