PCA Notes

ECL243, Feb-March 2018

I am attempting to re-create the polar bear and brown bear PCA plots from Liu et al (2014) while also learning how to use the command line/gitbash shell and run things on the UCD Farm. Authors used EIGENSOFT for PCAs.

Getting to Know the Data

1) Download and Unzip Data

- data: http://gigadb.org/dataset/100008
- transform from .gz to .txt with `gzip -d filename.txt.gz`

2) Look at the Data:

```
ebola@Bolas-XPS-13-9350 MINGW64 ~/ECL243/data (master) $ less -S
polar_bear.pooled.snp.txt
```

- this method makes the data actually readable, press "q" to get out

Get SNP data into a useable format

3) What format to we need?

- This is the first road block in this analysis. One issue was that I (we?) didn't realize SNP data could come in a lot of different file types, and that each file type needs to be formatted
- Our LD and structure analyses programs needed PED files, and EIGENSOFT supports 5 file types:
  ANCESTRYMAP, EIGENSTRAT, PED, PACKEDPED, PACKEDANCESTRYMAP
- EIGENSOFT has a conversion ability: https://github.com/argriffing/eigensoft/tree/master/CONVERTF
- If using snp files:

- see example.pedsnp *** file name MUST end in .pedsnp ***
  convertf also supports .map suffix for this input file name

- The snp file contains 1 line per SNP. There are 6 columns (last 2 optional):
  1st column is chromosome. Use X for X chromosome.
  Note: SNPs with illegal chromosome values, such as 0, will be removed
  2nd column is SNP name
  3rd column is genetic position (in Morgans)
  4th column is physical position (in bases)
  Optional 5th and 6th columns are reference and variant alleles.
  For monomorphic SNPs, the variant allele can be encoded as X.

4) Formating our data, attempt 1

- Our first idea is to use the PINK package of Shaun Purcell to turn our data into .ped or .map. Plink runs in the command line (not the Bash terinal) `C:\Users\ebola\ECL243\data>plink #to open it`

- For more details on PLINK,
  see: http://zzz.bwh.harvard.edu/plink/data.shtml#bed

- Our data looks like:
  ```
  chromo position ref anc major minor #major #minor knownEM pK-EM EG01 WG01
  EG02 EG03 EG04 WG02 EG05 WG03 WG04 WG05
  ```

- Problem: we need to make a SNP name, not sure if the position we have is genetic or physical, we may be able to skip the genetic position, not sure the allele situation

- Possible solution: use vim a fancy .txt editor that will work this kind of data by selecting whole columns, etc.

5) Formatting our data, attempt 2

- Each step here took us quite a bit of time to figure out what was going wrong

- Problem: This data appears to be in HapMap format (thanks for figuring that out JRI!), and plink doesn't do HapMap, so using it for conversion isn't a good option

- Solution: Use Tassel (designed for GWAS) to do file format conversion to make ped and map files that can be used with a variety of softwares. **BONUS! We can also use Tassel for LD and PCA.**

- **About Tassel:**
  http://www.maizegenetics.net/tassel
  https://bitbucket.org/tasseladmin/tassel-5-source/wiki/Home
  https://bytebucket.org/tasseladmin/tassel-5-source/wiki/docs/Tassel5PipelineCLI.pdf?rev=eeb05c28894bf976c84bb110ec545647ab9a2f08
  https://bitbucket.org/tasseladmin/tassel-5-source/wiki/browse/UserManual

- Before opening our data in Tassel, it turns out that it's not even a working HapMap file, but instead needs to be re-formatted a little first before we can even put it in Tassel.

    - Reminder, our data looks like:
      ```
      chromo position ref anc major minor #major #minor knownEM pK-EM EG01 WG01 EG02 EG03 EG04 WG02 EG05 WG03 WG04 WG05
      ```
      But to be HapMap, it needs to look like:
      ```
      rs# alleles chrom pos strand assembly# center protLSID assayLSID panelLSID QCcode EG01 WG01 etc.
      ```

- So, in R and Vim (in command line) we do the following (please see documents folder for scripts used):

    - make an rs# column that is a combination of chromosome and position
    - remove "scaffold" from chromosome
    - combine major and minor to make an alleles column
    - cut ref, anc, major, minor, #major, #minor, knownEM pk-EM
    - add columns: strand (+), assembly# (NA), center (NA), protLSID (NA), assayLSID (NA), panel LSID (NA), QCcode (NA)
        - Note: Normally the major allele is the ancestoral, but that isn't always correct, so derived vs. minor allele freq.are not the same thing. If the minor allele is T, the less freqent allele in your subpop. T could actually be the ancestoral state. The reference allele is the one in the reference genome. this is usually the major allele, too.
    - Save the data as tab delimited, make sure to tell it to leave out quotes
    - Use vim to add "#" since R is stressed by that

Working with the Data

6) New Work Plan

- Once the data is formatted for HapMap, we confirm that a subset of this data loads in the Tassel GUI. That means that we are now ready to do things on the cluster!
- We will do our best to replicate their qualitative findings using tassel for PCA and LD since we are already working with it
- If things look good in the GUI, we will run it in the cluster

good polar bear data: polarbearSNPcleanMar8new
good black bear data: blackbearSNPclean

7) Working with Tassel, headaches

- In the GUI, use "open as" and tell it to sort the data for it to load properly
- On the farm:
    - First, I watch JRI use tassel on the farm to try to understand how to do things
    - Then, I try to do it, and nothing works. I try doing everything in my own directories by installing tassel (two components of it) and run pipe using git clone, then putting these with my data in the same folder.
- Here I load tassel and friends:

```
bash: src/tassel-5-standalone/run_pipeline.pl: No such file or directory
ecbolas@bigmem9:~/ecl243/data$ git clone http://bitbucket.org/tasseladmin/tassel-
5-source.git
ecbolas@bigmem9:~/ecl243/data$ ls
polarbearSNPcleanMar8new.txt  polarbearSNPperfect2.txt  tassel-5-source
ecbolas@bigmem9:~/ecl243/data$ git clone http://bitbucket.org/tasseladmin/tassel-
5-standalone.git
Cloning into 'tassel-5-standalone'...
```

- But, running this doesn't work

```
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ ./run_pipeline.pl -Xmx6g  -h
polarbearSNPcleanMar8new.txt  -PrincipalComponentsPlugin -covariance true -
ncomponents 3 -endPlugin -export PCApb
./lib/avro-1.8.1.jar:./lib/jfreesvg-3.2.jar:./lib/ahocorasick-
0.2.4.jar:./lib/log4j-1.2.13.jar:./lib/biojava-alignment-4.0.0.jar:./lib/sqlite-
jdbc-3.8.5-pre1.jar:./lib/biojava-core-4.0.0.jar:./lib/ejml-
0.23.jar:./lib/postgresql-9.4-1201.jdbc41.jar:./lib/trove-3.0.3.jar:./lib/guava-
```

```
22.0.jar:./lib/commons-codec-1.10.jar:./lib/forester-1.038.jar:./lib/jhdf5-
14.12.5.jar:./lib/biojava-phylo-4.0.0.jar:./lib/snappy-java-
1.1.1.6.jar:./lib/javax.json-1.0.4.jar:./lib/commons-math3-3.4.1.jar:./lib/colt-
1.2.0.jar:./lib/json-simple-1.1.1.jar:./lib/jcommon-1.0.23.jar:./lib/itextpdf-
5.1.0.jar:./lib/slf4j-simple-1.7.10.jar:./lib/slf4j-api-1.7.10.jar:./lib/je-
6.0.11.jar:./lib/jfreechart-1.0.19.jar:./lib/mail-1.4.jar:./lib/junit-
4.10.jar:./lib/htsjdk-2.14.0.jar:./lib/jfxrt.jar:./dist/sTASSEL.jar
Memory Settings: -Xms512m -Xmx6g
Tassel Pipeline Arguments: -h polarbearSNPcleanMar8new.txt -
PrincipalComponentsPlugin -covariance true -ncomponents 3 -endPlugin -export
PCApb
Error: Could not find or load main class
net.maizegenetics.pipeline.TasselPipeline
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ run_pipeline.pl -Xmx6g  -h
polarbearSNPcleanMar8new.txt  -PrincipalComponentsPlugin -covariance true -
ncomponents 3 -endPlugin -export PCApb
```

- Problem: I have no idea why I am getting errors. Maybe because the data isn't sorted or named properly?

- Rename the data in the right file format:

```
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ cp polarbearSNPcleanMar8new.txt
polarbearSNPclean.hmp.txt
```

- Load stuff (both types of java):

```
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ module load jdk
Module JAVA 1.8.0.31 Loaded.
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ module load tassel
Module tassel/5.2.14 loaded
```

- then sort the data (just like the data was sorted when imported in the tassel GUI):

```
ecbolas@bigmem9:~/ecl243/data/tassel-5-source$ ./run_pipeline.pl -
SortGenotypeFilePlugin -inputFile polarbearSNPclean.hmp.txt -outputFile
pbsort.hmp.txt 1>/dev/null
Error: Could not find or load main class
net.maizegenetics.pipeline.TasselPipeline
```

- Same error! But…
    - I try this in the Tassel GUI on my laptop with sortex.hmp.txt, which is the sorted hapmap file for a subset of data and it works.
    - Also, Lauren is able to run stuff on her own cluster (using the codes below), so a lot of the analysis is done with the data she prepares.

8) Working with Tassel, good code but more headaches

- Solution: Run this in the tassel that's on the farm (rather than the one in my directory) and load one more java component. Here's the working scripts (and don't use ./ here for run pipeline):

```
module load java
module load tassel
```

- sort code (use "1>/dev/null" so that it doesn't output onto the screen but just makes the file):

```
run_pipeline.pl -Xmx24g -SortGenotypeFilePlugininputFile
blackbearSNPclean.hmp.txt -outputFile bbsort.hmp.txt 1>/dev/null
```

- PCA code for polar bears with 3 PCs

```
ecbolas@bigmem9:~/ecl243/data/$ run_pipeline.pl -Xmx32g  -h sortex.hmp.txt  -
PrincipalComponentsPlugin -covariance true -ncomponents 3 -endPlugin -export
PCApbsub
```

here, -Xmx32g means use 32 gigs, -h means use a hapmap file, then the plugin to use, if you want covariance or not, number of components, stop the plugin, then what you want exported.

- Brown Bear PCA with 2 PCs:

```
ecbolas@bigmem9:~/ecl243/data$ run_pipeline.pl -Xmx44g  -h bbsort.hmp.txt  -
PrincipalComponentsPlugin -covariance true -ncomponents 2 -endPlugin -export
bbpcafull2
```

- This code all seems fine, but it doesn't work b/c there some kind of error with this data set (I think)
- Polar Bear Subset with 2 PCs:

```
ecbolas@bigmem9:~/ecl243/data$ run_pipeline.pl -Xmx24g  -h sortex.hmp.txt  -
PrincipalComponentsPlugin -covariance true -ncomponents 2 -endPlugin -export
PCApbsub2
```

And this worked! Although I couldn't figure out how to get it off the farm.

PCA Plots

9) Plotting PCAs

- Once Lauren ran PCA on both data sets, I plotted those in R.

- Problem: the plots don't look the same as what the authors got. the PB PCA plot looks pretty similar to Liu et al (2014), but the BB PCA plot looks flipped on a diagonal.
- Possible explanations:
    - Values for PC1 and PC2 are very different then for us and the authors, is this part of the problem?
    - Did they choose more or less PCs than us (we used 3, maybe they only used 2, or used more?)
    - Did they use a log plot to make the data look better?

Take Homes

- SNP (and other genetic data) comes in a lot of crazy formats that may not work with the software that authors (or you) want to use
- Authors don't put enough information into Supplementary Methods. Under the PCA section, they have a lot of detail about how they called SNPS and created a covariance matrix and PCA for the low-coverage samples using EIGENSOFT for the latter two. For high coverage, they have a single sentence: "We also performed PCA on samples sequenced at 22X." I don't even know if that means they used Eigensoft for the high coverage or not, although I assume so. As stated above I also have no idea how many PCs they included, or if they played with the axes to make things look better.
- I still feel really unsure about how to run modules or scripts on the cluster. It's clear I have a lot to learn about programming language, and learning more about java will probably help, since so many softwares were built in java and use it's language
- Most "help"information or readme documents have been written for folks who already have a working knowledge of how to do bioinformatics. It's not clear where the beginner's guides are.