Principal Components Analysis:

Today we will contuniue to work with the genotype data produced yesterday, and we will start with performing a principal components analysis. We will be using the Eigensoft "smartPCA" package – it is located in the scripts folder that you were using yesterday. It is also possible to do PCAs in R and Excel, but for most it is required to first convert genotype data into a distance matrix. smartPCA can perform a PCA directly on genotype input files, avoiding this extra step. Unfortunately, though, smartPCA requires specially formatted input files. Our first goal will be to create these input files, using a custom-made python script.

a. Create the input files for the smartpca program.

Script to use: vcf2smartpca.py
Input file: VQSR_PASS_SNPS.vcf
Output files: *PREFIX*_Indiv, *PREFIX*_Geno, *PREFIX*_SNP, par.*PREFIX*

This script will convert the genotype data from the .vcf file into the proper format for analysis with the program smartpca. This includes four different files: the Indiv file, the Geno file, the SNP file and the parameter or 'par' file. Added to the names of all of the output files will be a prefix specified on the command line when the program is run. In order to focus in on only high-quality portions of the dataset, this script will only utilize data from SNPs that have been genotyped in all individuals.

Make sure that you are in the same directory as your data files, then execute the script with the following command:

```
../scripts/vcf2smartpca.py VQSR_PASS_SNPS.vcf passing_snps_q20 20
```

In this case, the prefix used is 'passing_snps_q20.' This argument can be any arbitrary string of characters, but it is nice to try to make it as informative as possible.

The third argument (20) is an integer specifying a minimum genotype quality cutoff. This is a cut-off for the quality of the individual-level genotypes, as opposed to the SNP quality as a whole. Individuals with genotype qualities below the minimum threshold will be treated as missing data. Therefore, you should try preparing input files at a couple different quality cut-offs. Choose a cut-off that provides a good number of SNPs (1000s, at least), while cutting out some portion of the worst quality genotypes. After creating the input files, the script will print to the screen the number of SNPs that were included in the input files for the smartpca analysis.

Following the last argument, you can also (if you want) specify the names of individuals that are in your .vcf file but which you do not want to include in the PCA. These names should be identical to the names in the .vcf file and should be separated by whitespace. If no names are specified then all individuals will be included in the PCA.

b. Run the principal components analysis using smartpca:

Program : smartpca
Iinput files: *PREFIX*_Indiv, *PREFIX*_Geno, *PREFIX*_SNP, par.*PREFIX*
Output files: *PREFIX*_#LOCI.eval, *PREFIX*_#LOCI.eval, *PREFIX*_snpweights, *PREFIX*_logfile.txt

Execute the script with the following command:

```
../scripts/smartpca -p par.passing_snps_q20 > passing_snps_q20_logfile.txt
```

There are several runtime parameters that are specified in the 'par' file. The above script automatically sets these parameters to levels that we have found to be useful in our data analysis. However, these parameters can easily be changed prior to running the analysis. All parameters are described in the help pages for smartpca.

This analysis will generate a number of output files. The one that we are the most interested in is the one with the extension '.evec', This file contains each individual's loadings on each of the principle components, and it is these loadings that we will plot to visualize the data. Open this eigenvector file in a Text editor after copying it to your computer. The top row specifies the amount of variance explained by each of the eigenvectors. This is followed by one row per individual containing that individual's loadings for each principle component.

To easily work with the data in Excel, we need to replace all of the whitespace in the file with tabs. We can do this with a simple find and replace within TextWrangler. To do so, press command + f, search for ' +' (that is a single space character followed by a plus sign) and then replace with '\t' (this is the regular expression for a tab character, make sure that the 'grep' box is checked in the 'find' dialog box). This search and replace, therefore, will replace any continuous stretch of spaces with a single tab character.

We can then copy and paste the contents of this file into Excel and create scatterplots to visualize the data. It is best to start by comparing principle components 1 & 2 because these will explain the largest variance in the data. Plot different 'series' for each population in the analysis so that differences can be easily visualized. If populations are significantly differentiated then we expect to see clustering of individuals by their respective populations. This clustering will not necessarily occur on the first several principle components though, so go ahead and visualize at least the first five or six, to see the major trends in the data.

The number of individuals that are included in the analysis determines the number of possible principal components for a data set. However, the default for this script is to only report the top 8 (this can be changed in the 'par' file).

PCA is most useful for getting a general idea of the major trends in the data set. However, it is also possible to look at the loci that explain the most of the variance in the data along a particular axis (i.e. likely to be under selection). This is done by looking at the PREFIX_snpweights output file. This file contains weights for each SNP along all reported principle component axes. The higher the absolute value of the weight, the stronger the correlation of that SNP with that principle component. Open the snpweight file in TextWrangler, and replace all white space with tabs (as above). Then copy and paste the data into Excel and find the most strongly weighted SNPs for the first few principal components.

c. Identify the number of significant eigenvectors using twstats:

Program: twstats
Input file: *prefix*_#loci.eval
Output file: *prefix*_twstats.txt

twstats is a second program that comes in the Eigensoft package. It takes the '.eval' file output during the smartpca analysis and calculates the significance of each principal component. This is a useful tool for determining the number of principal components exhibiting meaningful variation. However, the Tracy-Widom statistics used to determine significance can break down under certain conditions. See Patterson et al. (2006) for more details.

Execute the script with the following command:

```
../scripts/twstats -t ../scripts/twtable -i passing_snps_q20_245loci.eval
-o passing_snps_q20_twstats.txt
```

The file called “twtable” is necessary for these calculations. It is distributed along with the Eigensoft package and should be located in your *scripts* folder. Now, open the _twstats.txt file in nano and examine it. Are any of the axes significant?