Script and example coding for phylogenetic tree

17/3/19 J Receveur

* Installing VCFtools

git clone https://github.com/vcftools/vcftools.git

cd vcftools

./autogen.sh

./configure

make

make install

* Potential issues
  + If mac development tools are not installed it will tell you, if autoconfig is not installed you will get a not found error when running autogen.sh but make sure you are in the right directory because it will return the same error message if you try it in the wrong directory.
  + You may get an error if bgzip or tabix are not installed with some of the commands, if you do, see http://www.htslib.org/doc/tabix.html
* Navigate to folder with data using cd
  + Type ls to see what files are in the current folder(for windows the command is dir and linux !dir works as well)
* Type in vcftools and hit enter
  + If installed correctly, you should see a couple lines with general info about vcftools appear
* Commands in vcftools
  + Start each command with vcftools to specify that you want to use that program
  + next specify the data you want to use
    - This can either in a .vcf format where you would write --vcf filename.vcf
    - Or in a vcf.gz format which is a compressed version where you would write --vcf.gz filename.vcf.gz
  + The next part of the file is where you specify the action you want the program to do ( a full list of options is below)
    - For example if you want to filter out chromosome 3L you would now input --chr 3L
    - Or if you wanted to output the relatedness between samples using Yang’s method from 2010 you would type --relatedness
    - Other commands may require additional inputs but the more complicated inputs are listed below
  + The next optional part is the --recode option
    - This will take the header/metadata and only include the metadata for samples that passed filtering
      * For example if you give it the command to filter chromosome 3L, adding the recode option will remove all the header/metadata associated with anything that is not on chromosome 3L
      * For most of the filtering options, always add the --recode option, otherwise it interferes with analysis. There are exceptions but don’t worry about them for now
  + The next part of the command is the output file, where you specify what you want to the output file name to be
    - Most of the commands will add their own file extension to the end regardless of what you write
      * For example if you name a file with -o filteredSamples.vcf the command will return a file name of filteredSamples.vcf.vcf
      * Some general formatting tips for naming files
        + Do not include any spaces, special characters (like $ % !)
        + Keep them as short as possible while still being informative

It reduces the chance of making a spelling mistake while writing and saves time

* + - * + Keep your use of capital letters consistent

If you have a file named filteredSamples.vcf and wanted to use that in an input function if you inputted filteredsamples.vcf it wouldn’t work and it would return an error message

* An example with all the parts put together, if you wanted to filter out the sequences for chromosome 3L and save it in a new file, while recoding the metadata you would write the following command
  + vcftools --vcf merged.vcf --chr 3L --recode --out chr3L
  + the command would create a new file named chr3L.vcf in the file directory that you are currently in.

Example of taking the mosquito genome data, filtering before testing for relatedness between samples

* Filtering
  + 1st step remove any SNPs that have a filter other than PASS from the previous filtering steps that were conducted by the *A.* gambiae genome project
    - Some examples of sites that would have a flag other than pass are SNPs of low quality or locations that were otherwise unsuitable for SNP analysis, for example SNPs in areas of long tandem repeats that the reference file has low mapping quality for
    - vcftools --vcf merged.vcf --remove-filtered-all --recode --out Pass
  + 2nd step, remove any SNPs that do not meet the quality that you want ( in this example 40 but replace it with whatever number you want)
    - Sequencing Depth
      * vcftools vcf PASS.vcf --min-meanDP 40 --recode --out depth40
  + 3rd step Genotype filtering, removes all locations where the genotype quality is below a user defined number
    - vcftools --depth40.vcf --minGQ 30 --out filtered
  + Separating out the data by chromosomes
    - vcftools --vcf filtered.vcf --chr 1L --recode --out chr1L
    - the recode option is important for this step
    - For the rest of the chromosomes, just replace 1L with whatever chromosome you want (the data from this project is named according to the NCBIs naming for mosquito chromosomes)
    - The files at this step will have to be divided into L and R arms but they can be put back together later
  + Separating out data by base pair
    - if you would like to subset a section of a .vcf file
    - vcftools --vcf filtered.vcf --from-bp 1 --to-bp 10000 --recode --out bp1.10000
    - The NCBI database will give the locations for genes using the same reference strain as used to create the vcf file.
* Calculate the Fst for each SNP in a group of samples
  + For this command, you will need to create two or more text files, each of them containing all the sample names for one population with no header, for each of the populations you would like to compare.
  + For example to compare the differences in SNPs between coluzzi and gambiae

* + vcftools --vcf InGene.vcf --weir-fst-pop AnColuzzi.txt --weir-fst-pop AnGambiae.txt out InGeneBySpecies