Using Structure

Users of Structure (Pritchard et al, 2000) may be familiar with the interface of the BioHPC cluster at Cornell. Unfortunately, guest access was discontinued in May, 2011. While Structure (v. 2.3.1.2) is installed (at /opt/apps/sdg/structure/), several features of the web-based BioHPC cluster interface can be replaced with a pipeline of qsub and python scripts. The scripts can be found at /opt/apps/sdg/scripts/ . Some keystrokes can be saved by setting $SDG_ROOT in the shell profile (See SDG Group Profile Setup)

Preparing to Run Structure

To run structure, three files are necessary: project_data, mainparams, and extraparams. The project_data file contains your data (microsatellites, AFLPs, SNPs). Several file formats are acceptable, and you can view sample datasets at the Structure website. The most common format is to have one individual per line, with alleles arranged in columns (two columns per locus for diploid data and codominant markers).

The easiest way to ensure your file is formatted correctly is to use the GUI version of Structure on your own computer. Choose "New Project" from the File menu and follow the setup wizard to import your file. You will be asked several questions about the format of your data, such as whether there is a column of individual IDs or a row of marker names. If the data file is accepted, a "project_data" file will be created inside your project directory.

The mainparams file contains information about your data and the admixture model you wish to run. For first-time users of Structure, it may be easiest to generate the mainparams file from the GUI version on your computer. Once you've created a new project, choose "New.." from the "Parameter Set" menu. You will choose the run length (such as 10K burn in and 1M run generations) and admixture models (refer to the Structure manual for additional details).

Next, generate a mainparams file from the "Project" menu (Generate Parameter Files...). You only need to generate one mainparams file, since you will be setting parameters such as K and the output filename from the command line. So when it asks for a "range of K" simply enter from 1 to 1. Name the parameter files and quit the GUI version!

Finally, you must create an extraparams file. For most users, this will be an empty file, but it must exist or structure will not run. Create this empty file with your text editor, and then upload project_data, mainparams, and extraparams to a directory on your cluster space.

Running Replicates of Structure

Structure is frequently run multiple times with the same parameter settings, and with multiple values of k (the number of clusters). Both of these can be achieved within one qsub script. The sample qsub script below will submit an array job (one for each value of k) and will employ a loop to run each several times.

```
#! /bin/bash


#$ -S /bin/bash -cwd
#$ -M myemail@domain.com -m e
#$ -N name_of_script
#$ -o script.out -j y
#$ -t 1-5

#10 is the number of replicates at each value of K.
for rep in {1..10}
do
structure -m mainparams -K $SGE_TASK_ID -i project_data -o outfile_k${SGE_TASK_ID}_rep${rep}
done
```

To set the values of K, change the line with **-t** This line sets the $SGE_TASK_ID variable and submits an array of jobs. The $SGE_TASK_ID variable is passed to the script so that the value of K and the output file names are changed for each job in the array.

To set the number of replicates, change the second number in "{1..10}".

The script will execute the program structure with the following parameters:

flag	name	meaning
-m	mainparams	the name of your mainparams file in the current directory
-K	number of clusters	set by the $SGE_TASK_ID in the array job
-i	project_data	the name of your project_data file in the current directory
-o	outfile name	the output file from each structure run, named for each replicate in each value of K

This script is written in bash. You may need to set your $PATH variable in your .bashrc in order to run it correctly (See SDG Group Profile Setup)

Summarizing Replicate Runs

The BioHPC cluster would return a file, simsum.txt, which would have the likelihood and Fst values for replicate runs of Structure. If you've run Structure as above, the following python script can create this file. Indicate all of the output files from Strucutre in a list. Wildcard expansion should work, so if you files are of the form outfile_k2_rep2 you can simply write outfile*. You also need to provide the maximum value of K in your dataset:

```
python $SDG_ROOT/scripts/simsum.py 5 outfile*
```

For each value of K you wish to investigate further, the Q-matricies from each replicate run of Structure will be slightly different. Use the program CLUMPP (in $SDG_ROOT/bin) to determine a single Q-matrix. First, format the input files for CLUMPP with another python script. Specify an output file name, and then list all of the replicate runs-- wildcard expansion should work here too:

```
python $SDG_ROOT/scripts/structure2clumpp.py outputfilename outfile_k2*
```

You will also have to setup the paramfile. An example CLUMPP file with default settings can be found here. You will need to change the values for the names of input and output files, and other dataset info. (If you want to fine-tune the parameters, see the CLUMPP manual for further instructions). Run CLUMPP with the command:

```
CLUMPP clumpp.paramfile
```

Depending on the size of the dataset, CLUMPP may take a few seconds to several hours. Either run from a qsub script or in an Interactive Job

Visualizing Results

Visualization of the likelihoods from replicate runs at each value of K, along with the post-hoc delta-K statistic (Evanno et al., 2003), can be done with an R script: see $SDG_ROOT/scripts/deltak.R . This script is perhaps best run on your own computer so that graphics parameters may be adjusted.

Similarly, a script scructure_barplot.R is also in the $SDG_ROOT/scripts directory. This script can output the familiar barplot: one bar per individual, different colors representing admixture/population assignment. If a list of geographic or putative populations is given, they will be marked on the graphic.