

Funk et al -- Arlequin 3.5 Fst Values

For this part of the project, I was responsible for replicating the Fst values that the project found between the various fox populations.

The instructions in the paper are as follows: "Pairwise FST estimates and their significance were calculated using 1000 permutations in ARLEQUIN 3.5 (Excoffier et al. 2005) and Jost's D estimates were calculated in GENALEX 6.5 (Peakall & Smouse 2006, 2012)."

Step 1: Setting up Arlequin

The first step in replicating the Fst values was downloading and setting up Arlequin version 3.5.

Arlequin 3.5 is downloadable from the developer website for Windows, as there is only a GUI for Windows. Downloading and using Arlequin 3.5 on a Mac involves using command lines.

The developer website is: cmpg.unibe.ch/software/arlequin35/Arl35Downloads.html.

Install is straightforward and involves unzipping the file and opening the Windows application.

After opening the application and before use, specify the helper program in the box at the bottom of the 'Arlequin Configuration' tab. I used TextPad 8. I specified no R packages and kept all the boxes above the Helper programs checked on their default.

These are the checked boxes: Use associated settings. Append results. Prompt for handling unphased multi-locus data. XML Output. Use 64bit external arlecore program for computations.

Change no other settings.

Step 2: Conversion

The second step in getting the Fst values involved converting the original data files to Arlequin format. Arlequin has a very specific format that the input file must be for the software to work.

The paper's original files were in GenePop format, which is a common format for genetic data. However, Arlequin does not read GenePop format.

Arlequin has a conversion program built-in, and it can be found under the Import Data tab. I found this to be unhelpful, as it was not able to truly convert my data. I uploaded the original data with grey fox data, set the

format type as GenePop for the source, and then set the target format to Arlequin. After pressing the Translate button, there was no change and no conversion.

After attempting the translation in Arlequin and finding it to be unsuccessful, conversion in R Studio was attempted. The code for this is found in the `Fox_Conversion.R` file in this repository. It utilizes PopGenKit package, the HWxtest package, and the adegenet package. These were also unsuccessful.

The final converter utilized was PGDSpyder. This was successful in converting the Genepop format to Arlequin format. PGDSpyder 2.1.1.0 was the version used. This software asks for Input File Format, which should be Genepop, and to select an input file. Unlike the Arlequin converter, it requires an output file to be made before the conversion can take place.

To do this, the Arlequin Project Wizard tab is useful. In this tab, you can create a new Arlequin project that has the necessary .arp extension. After clicking Browse, upload the original text file, set the Data Type to STANDARD, and leave the three boxes unchecked, as well as the number of samples as 1, the Locus separator as WHITESPACE and Missing Data as ?. Leave all the Optional sections unchecked. When converted, these will be overwritten. Clicking CREATE PROJECT will create a file with an .arp extension PGDSpyder can use to store output data.

Back in PDGSpYder, select the newly created project file as the output file, with the format of Arlequin. After clicking convert, it will ask if you want to overwrite the data. Click yes. After this, the SPID editor will pop up. Set the data type to SNPs, and the microsatellite alleles coded as random numbers. You can choose to save it or just apply it.

Step 3: Making the data usable

Despite all the work that goes into conversion, I was still unable to get the converted file type to work in Arlequin. This was most likely due to a formatting issue and it would give a strange error message that read: "Phenotypes have uneven number of loci."

I consulted with Dr. John Nason, and he was able to get my data into a more usable format. He opened the Arlequin data file in a text editor, and changed the DNA type to STANDARD from DNA. He then ran the data through a program that got rid of the zeros in front of the allele numbers, and then aligned all the columns. My assumption is that the program was written in Python and the alignment was done in Unix.

In this step, I also went into the Arlequin file and changed the pop_ names to the actual population names, as they were lost in the conversion.

Step 4: Pulling Data into Arlequin

This step is the most straight forward. In Arlequin, go to Open Project and open the .arp file with the converted and processed data. When the file is opened in Arlequin, you will be prompted with a 'Handling unphased genotypes' box. It was unclear if the original paper had this problem, so I choose the second option in the box: 'Consider unphased data as multi-loci data with unknown gametic phase: Arlequin will handle unphased multi-locus genotypes the old way, and some analyses requiring explicit phase information won't be possible.'

Under the Settings tab in Arlequin, go to Genetic Structure, and click on Population comparisons. Click on Compute pairwise FST. Make sure the number of permutations in 1000, and the significance level is 0.05. The two boxes underneath should read Computer distance matrix and Number of different alleles. Click on the Start button in the upper corner. A black text box will appear to show the progress of the project analysis. It will disappear when the analysis is over. After it disappears, click on the View Results box in the top bar. This will take you to an HTML page with all the results.

The Fst results will be found on this page under Pairwise comparisons.

Afterward

Once this is done, the results can be compared. I found that despite all of my processing, I could only get a portion of the SNP loci to read in correctly. The rest the program disregarded as too incomplete to be used. The paper reported using all of the SNPs, so there is possible there is another processing step that I was unable to figure out.

More accurate information on how Arlequin was used and how the data was converted would have been helpful.

All results from this project are stored on GitHub.