OutFLANK

Finding F_{ST} outliers with an inferred neutral distribution

Michael C. Whitlock and Katie E. Lotterhos

whitlock@zoology.ubc.ca; lotterke@wfu.edu

OutFLANK is an R package that implements the method developed by Whitlock and Lotterhos (in prep.) to use likelihood on a trimmed distribution of F_{ST} values to infer the distribution of F_{ST} for neutral markers. This distribution is then used to assign *q*-values to each locus to detect outliers that may be due to spatially heterogeneous selection.

If you use OutFLANK, please cite:

Whitlock, M. C., and K. J. Lotterhos. Reliable detection of loci responsible for local adaptation: Inference of a neutral model through trimming the distribution of F_{ST}. To be submitted to *The American Naturalist*.

Please see this paper for a discussion of the limitations and strengths of this approach.

To load OutFLANK

OutFLANK is distributed through github. To install the OutFLANK package, first install the devtools package on your session of R, followed by the installation of OutFLANK from github.

```
install.packages(“devtools”)
install_github(“whitlock/OutFLANK)
```

Next, load the package into your R session. In later uses of the package, you can start with this library step.

```
library(OutFLANK)
```

You may also need the bioclite with its qvalue routines:

```
source("http://bioconductor.org/biocLite.R")
biocLite("qvalue")
```

Preparing input files for OutFLANK

The input for OutFLANK is a data frame, with a row for each locus. OutFLANK requires that the input data include a large number of loci not strongly affected by spatially heterogeneous selection, although these need not be previously identified. It will work best with at least thousands of loci in the data set.

There are several required columns in the input data frame. The input can also include other columns, which will be ignored by OutFLANK as long as their names do not include $GoodH, $q, $OutlierFlag , or $indexOrder.

Some of the columns in the input are somewhat unusual, and functions to create the values in these columns are also included in the package.

Here are the columns required for the input data frame. They should have these names, but order does not matter:

`$LocusName`: a character string that uniquely names each locus.

`$FST`: F_{ST} calculated for this locus. (Kept here to report the unbiased F_{ST} of the results)

`$T1`: The numerator of the estimator for F_{ST} (necessary, with $T2, to calculate mean F_{ST})

`$T2`: The denominator of the estimator of F_{ST}

`$FSTNoCorr`: F_{ST} calculated for this locus without sample size correction. (See below for ways to calculate this and the next two columns)

`$T1NoCorr`: The numerator of the estimator for F_{ST} without sample size correction (necessary, with `$T2`, to find mean F_{ST})

`$T2NoCorr`: The denominator of the estimator of F_{ST} without sample size correction

`$He`: The heterozygosity of the locus (used to screen out low heterozygosity loci that have a different distribution)

`$indexOrder`: integer index giving the original order of rows in the input file.

For diploid data, the appropriate input data frame can be calculated using the function `MakeDiploidFSTMat(SNPmat,locusNames,popNames)`. The output of this function is a data frame that can be given to the function `OutFLANK`() as the input parameter `FstDataFrame` (see below). This function requires three data objects as input. `SNPmat` is an array with a row for each individual in the data set and a column for each locus. This function assumes biallelic data, and the value in each column is either 0, 1 or 2, showing the number of the focal alleles that the individual carries at that locus. (I.e., 2 means that the individual is a homozygote for the focal allele, 0, means that the individual has no copies of that focal allele, and 1 indicates a heterozygote.) For any locus that is unknown for an individual, `SNPmat` should have a 9 for that locus on that row.

The other two parameters for `MakeDiploidFSTMat()` are `locusNames` and `popNames`. The vector `locusNames` gives a list of identifying names for each locus. (The length of the `locusNames` vector should be the same as the number of columns in `SNPmat`.) The vector `popNames` has an entry for each individual, in the same order as the rows in `SNPmat`, which gives the population that that individual came from. OutFLANK assumes that the individuals are grouped into relatively discrete populations, so there must be multiple individuals per population for the function to work properly.

If you use `MakeDiploidFSTMat()` to create the input data frame, you don't nee d to use the F_{ST} functions described in the next subsection.

Calculating F_{ST} and FSTNoCorr

OutFLANK finds outliers for F_{ST}, but it needs to use estimates of F_{ST} that have not been corrected for sample size adjustments. This is because these sample size adjustments can sometimes cause the estimate of the F_{ST} of a locus to be negative, which is not a possible value of the χ^2 distribution used by OutFLANK. These uncorrected values are labeled with "NoCorr", e.g. FSTNoCorr is the F_{ST} of a locus without this correction. (In order to properly average F_{ST} over loci, OutFLANK also needs the numerator and denominator of the F_{ST} calculation for FSTNoCorr; these are called T1NoCorr and T2NoCorr respectively.

These NoCorr statistics can be calculated for biallelic loci using two other functions included in the OutFLANK package:

```
WC_FST_FiniteSample_Haploids_2AllelesB_NoSamplingCorrection(AllCounts)
```

and

```
WC_FST_FiniteSample_Diploids_2Alleles_NoCorr(Sample_Ma
t)
```

calculate FSTNoCorr, T1NoCorr, and T2NoCorr for haploid or diploid data, respectively. For both of these function, the input is an array with data about a single locus, with a row for each population and a column for each genotype. For the haploid function, there are two columns which contain counts of alleles of each of two types found in each population. For diploid data, there are three columns, with data giving counts in each population for the first homozygote, the heterozygotes, and the other homozygote, respectively.

These functions are based on Weir and Cockerham (1985) and Weir (*Genetic Data Analysis II,* 1996, Sinauer).

Similar functions are also included that can be used to populate the columns in the OutFLANK input data frame for the corrected values of FST,T1, and T2. These functions are derived from the package hierfstat by Jérôme Goudet:

```
WC_FST_FiniteSample_Haploids_2AllelesB_MCW(AllCounts)
```

and

```
WC_FST_FiniteSample_Diploids_2Alleles(Sample_Mat)
```

Each of these functions will also return the expected heterozygosity (`He`) for each locus.

Running OutFLANK

Once the appropriate input dataframe is available (see above), the procedure can be run with one command:

```
OutFLANK(FstDataFrame, LeftTrimFraction=0.05,
RightTrimFraction=0.05, Hmin=0.1, NumberOfSamples,
qthreshold=0.05)
```

FstDataFrame is the input package prepared above. (If your dataframe for OutFLANK is called "myFs", then the argument would read FstDataFrame = myFs.)

LeftTrimFraction and RightTrimFraction are arguments, each set to 5% by default, to tell OutFLANK how many of the lowest and highest F_{ST} values to remove before estimating the shape of the F_{ST} distribution through likelihood. (This removes the loci most affected by selection, and the likelihood procedure in OutFLANK accounts for the fact that data below and above these thresholds have been removed.) In some cases, when there are potentially a large number of loci affected by spatially heterogeneous selection, it can be worth trying a higher RightTrimFraction.

Hmin is the threshold for expected heterozygosity. Loci with low H_e have distributions of F_{ST} very different from loci with higher H_e, and it is important to screen these out before proceeding. By default, OutFLANK removes from consideration all loci with expected heterozygosity less than 10%.

NumberOfSamples is the number of populations sampled. This is the only other argument without a default, aside form the data frame.

qthreshold sets the threshold for whether a locus is deemed an "outlier", i.e. if the *q* value for that locus is below this threshold. This is set to 5% by default (aiming at a 5% or lower false discovery rate.) These *q*-values are calculated based on the right-tail *P*-values for each locus.

Plotting neutral distributions of F_{ST}

We recommend that you plot the distribution of FSTNoCorr for your loci against the inferred distribution. The included function OutFLANKResultsPlotter uses the output of OutFLANK to plot the distribution of FSTNoCorr:

```
OutFLANKResultsPlotter(OFoutput, withOutliers = TRUE,
NoCorr = TRUE, Hmin = 0.1, binwidth = 0.005, Zoom =
FALSE, RightZoomFraction = 0.05, titletext = NULL)
```

`OFoutput` is the output of the function OutFLANK(). Run OutFLANK first, storing the results to a variable.

`withOutliers` determines whether to include loci flagged by OutFLANK as outliers in the histogram.

`NoCorr = TRUE` tells the function to plot FSTNoCOrr instead of FST. This is recommended, because the distribution curve plotted will be predicted for FSTNoCorr, not F_{ST} itself.

`Hmin` tells the function the minimum expected heterozygosity that a locus should have before it is included in the plots.

`binwidth` describes the width of the histogram bins.

`Zoom` provides the option of "zooming in" on the right hand tail of the distribution. When Zoom = TRUE, only loci in the highest quantiles of F_{ST} will be plotted (with the proportion included determined by `RightZoomFraction`.)

`titletext` provides an opportunity to provide a title for the graph. Put the phrase you want as the title in "quotes".