**Supplemental Methods**

Detailed Description of the PopNet pipeline

PopNet is a set of Python scripts that together work as a pipeline to produce network graph files, readable by the Cytoscape network visualization tool, from either a table of SNPs or a collection of SNPS files as generated by MUMMER3. The input SNPs are used to construct a series of similarity matrices, each representing one fixed-size section of the genome. The matrices are clustered in a ‘Primary Clustering’ step to generate a series of clustering patterns representing the local ancestry of each sample within that section. These patterns are used to define the subpopulations within the samples during a ‘Secondary Clustering’ step, which are referred to as ‘groups’. Combining the results from the two clustering steps, a ring shaped (annulus) depiction of shared ancestry is generated through a chromosome painting step. These annuli are subsequently used as nodes in a network visualization of population structure.

*Configuration file*

PopNet is evoked via ‘python path\_to\_popnet/FullRunner.py path\_to\_config’, where the argument is a configuration file in plain text. The format and options in the configuration file are shown in the example configuration file provided with the package. All options need to be present.

* Base\_directory: Location of the input file(s)
* Output\_directory: Location of the output files. The lowest level directory will be created if not present.
* Organism: Specifies the organism bring analyzed. Current options are toxoplasma, yeast, and plasmodium. Only affects the parsing of the chromosome names.
* Input\_type: Specifies the type of input used. Options are tabular or nucmer.
* File\_name: If tabular type is used, specifies the file name of the SNP table.
* Reference: Specify only if the first column of the SNP table is the reference genome. Otherwise use: None.
* Section\_length: Length of each section used for primary clustering, in base pairs.
* S1\_iVal: Inflation parameter for Primary Clustering
* S1\_piVal: Pre-Inflation parameter for Primary Clustering
* S2\_iVal: Inflation parameter for Secondary Clustering
* S2\_piVal: Pre-Inflation parameter for Secondary Clustering

The S1 values should generally not be changed. The S2 values should be set according to the heatmaps generated with each run of PopNet, located in the output folder as heatmaps.pdf. We therefore recommend that each dataset be analyzed twice: first with default S2 parameter values to identify optimal dataset-specific S2 values, and then to generate the final visualization.

*Steps in the PopNet pipeline*

1. Loading Data

PopNet starts by reading either the input tabular file, or the FASTA files provided by the configuration file, and storing the information as an internal data structure (nested dictionary). This consists of a set of chromosomes, each of which is associated with a list of positions which indicates the nucleotide at that position for each sample. During this step, singletons (where only one sample is different from the rest) are filtered out. This data structure is recorded as a text file as results.txt (a brief example of output is provided below).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Chromosome | RefPosition | 273614N\_S288C | 322134S\_S288C | 378604X\_S288C | AWRI1631\_S288C |
| SC\_CHRI | 56 | - | - | - | - |
| SC\_CHRI | 63 | - | - | - | - |
| SC\_CHRI | 2796 | - | - | - | - |
| SC\_CHRI | 2797 | - | - | - | - |
| SC\_CHRI | 2823 | - | G | G | - |
| SC\_CHRI | 2825 | - | C | C | - |
| SC\_CHRI | 2842 | - | - | - | - |
| …. |  |  |  |  |  |

1. Generating Matrices

The data structure is used to generate one matrix for each defined section of sequence (as defined by the Section\_Length parameter in the configuration file). Starting at position 0 of each chromosome, the program generates an NxN (where N is the number of individuals being analysed) matrix where each element in the matrix represents the SNP similarity between a specific pair of samples. Each element is initialized to 0. The program then iterates through each successive SNP position until the end of the defined section. At each position, a pairwise comparison is performed between all strains. If two strains bear the same nucleotide at that position, then the element that corresponds to the two strains is incremented by 1. At the end of the block, the matrix is normalized by dividing each cell by the total number of positions considered for that matrix (i.e. number of SNP positions), and recorded as ready for ‘Primary Clustering’. A new matrix is then initialized and the process repeated for the next block of sequence until the end of the chromosome is reached. Pseudo-code is provided for clarification.

##

List positions = List containing each row of the data structure from Step 1

List samples = Ordered list of all samples

Array Matrix = NxN matrix of zeroes

For position in positions:

If not current block:

#normalize current matrix

#record current matrix

#create zero matrix

For sample\_1 in samples:

For sample\_2 in samples:

If position[sample\_1] == position[sample\_2]:

Matrix[sample\_1][sample\_2] += 1

###

1. Primary Clustering

Each matrix, which represents a similarity matrix between all individuals within a given section, is clustered using the MCL algorithm immediately after being generated. The parameters specified for this include the inflation and pre-inflation values. Together with the matrix, these two values determine the number and size of the clusters formed.

In the clustering process, MCL attempts to place sample pairs that have high similarity values in the matrix into the same cluster. The output from MCL is a nested list (i.e. a list of lists) where each top level list represents one cluster, and contains the names of all individuals within that cluster.

1. Secondary Clustering

The results of the Primary Clustering are collected, and a second NxN matrix is created where each element in the matrix represents the number of times a specific pair of samples appear in the same cluster. This new matrix is then again clustered using MCL, and these resultant clusters define the Clades associated with that population. Each Clade is assigned a unique color based on the HSL spectrum.

1. Chromosome Painting

For chromosome painting, each individual is considered in sequence. For each individual, PopNet iterates through all the primary clustering results to determine the parent group for the considered individual at each segment. In general, the Clade with the highest proportion of members present in the same cluster as the individual is defined as the parent Clade. To account for genetic drift and potential sequencing errors and also to resolve situations where multiple possibilities exist, a ‘chain extension’ method is employed. A key assumption here is that meaningful patterns in ancestry likely extend beyond a single section. Moreover, the Clade sharing the most recent ancestor should show the longest stretch of similarity out of all possible Clades.

PopNet begins each chain by considering all Clades as possible ancestors. It then iterates through each segment, and determines whether each Clade is a candidate as sharing an ancestor with that individual, referred to as ‘matching’, for that segment. Matches are defined when at least 30% of the Clade members clusters with that individual (note the 30% is a tunable parameter that can be adjusted to provide greater resolution or minimize noise). This continues for each Clade until they are no longer deemed candidates for this chain by exceeding a certain threshold of non-matching segments. This threshold can be exceeded either by the ratio of matching:non-matching segments (Gap Penalty), or by having too many consecutive non-matching segments (Max Gap Length). The last remaining candidate Clade to exceed the threshold is assigned as sharing a common ancestor for all segments within this chain, and those segments are assigned that Clade’s colour. Finally, a backtrace is performed to remove all segments that do not match the assigned ancestor from the end of the chain, terminating that chain. The next chain then starts at the position the previous chain terminated. After chromosome painting has completed, any section that contained missing data will have the assigned color changed to grey.

Pseudo-code for the chain extension algorithm is presented for clarification:

###

#Function finds the length and most likely Clade sharing an ancestor (candidate) for the current chain.

List initial\_candidates = all candidates (A1…Ax)

List tape = [1:{Object containing value for all candidates}, 2:{Object contain…}, 3{Obj…}…]

Int Total\_score = 0

Int Current\_mismatch\_score = 0

Int Mismatch\_penalty = P

Int Max\_mismatch\_value = M

Int Minimum\_Score = S

For section in chain:

#record current position

For candidate in initial\_candidates:

If candidate’s value at this section > X:

Current\_mismatch\_score = 0

Total\_score += 1

Else:

Total\_score -= P

Current\_mismatch\_score += 1

If Total\_score < S or Current\_mismatch\_score > M:

#remove candidate from candidate list

If candidate list length == 1:

#Iterate backwards until a segment that matches the last candidate is encountered

#returns the last element in candidate list and the current section after backtrace

###

1. Visualization

Visualization uses the Cytoscape network visualization software, with the *enhanced graphics* plugin. PopNet generates an xgmml file including all nodes, edges, and chromosome painting data, to be imported into Cytoscape. PopNet does not define the location of each node. Instead, network layout can be generated using any of the many layout algorithms included in the Cytoscape package (typically we use Force Directed layouts).

Nodes in the visualization represent individuals, and each node depicts the chromosome painting of the individual as an annulus. We recommend that the border colour of each node is mapped to ‘color’ column of the Node table. Edge weights connecting nodes are derived from the matrix used in the Secondary Clustering steps, but are rescaled according to

where x is the normalized value of the corresponding element in the matrix (i.e. represented as a fraction of the highest value in the matrix), a is a scaling factor (by default set to 10), and Vmax is the value of the highest edge weight allowed. This rescaling emphasizes the highest values while limiting the visual representation of lower values. This is performed since it is expected that there will be a significant degree of noise in this matrix. A trimming algorithm is then applied to delete edges with low weights, typically resulting in removal of at least 60% of the edges.

Pseudo-code pertaining to the trimming algorithm is provided below:

###

List anchors = []

List groups = []

List numbers = [sorted list of integers]

Int stringency = S

Int percentage = P

For number in numbers:

Bool matched = False

For anchor in anchors:

If abs(anchor – number / anchor) < S:

Matched = True

#Add to groups.index(anchor)

Break

If not matched:

#Add number to anchors

#Add empty list to end of groups

While true:

Remove first group (smallest numbers)

If total length of all groups < P \* length of numbers or

Length of group is 1:

Return groups

###

1. Output

PopNet outputs an XGMML as primary output, as well as a number of other files for diagnostic purposes. The most commonly ones are listed below:

**/path/to/results/cytoscape/cytoscapeGenome.xgmml**: The primary output – the graph file to be loaded into cytoscape.

**Heatmaps.pdf**: Metrics to help you decide on the Inflation (I) and Pre-inflation (pI) parameters for secondary clustering

**log.txt**: A log file that includes all the run parameters

**Genome\_nexus.nex**: A neighbor-net of the population

**groups.txt.mci**: The raw matrix used in secondary clustering. You can quickly try out the effects of different I and pI values.

**persistentResult.txt**: The clustering results of each individual chromosome segment during primary clustering

**results.txt**: A tab-delimited file containing all the SNPs used

**/cytoscape/tabNetwork.tsv**: A tab-delimited file containing all chromosome paintings in the network. Useful for focusing on specific locations on the genome.