**Bayesian NMF GWAS clustering pipeline**

GREEN = process of choosing final variant set (choose\_variants.R)

ORANGE = process of retrieving z-scores and preparing the input matrix (prep\_bNMF.R)

BLUE = bNMF clustering process and summarization (run\_bNMF.R)

**choose\_primary\_GWAS\_variants (this is not included in the code pipeline)**

* Input: primary outcome
* Output: List of variants as rsIDs with associated REF/ALT alleles
* Notes: Process: filter out studies w/ N<10k, **confirm that all variants have p < 0.05/N\_snps in the largest GWAS study**, take alleles from largest study

**count \_traits\_per\_variant**

* Input: list of summary statistic files, one per trait
* Output: vector of counts of non-missing traits for each variant
* Notes:
  1. For now, assume that summary statistics have a consistent set of headers
  2. If a single directory name is provided, loop through all files in that directory

**find\_needed\_proxies**

* Input: list of variants + alleles from GWAS, vector of trait counts per variant
* Output: list of variants that need proxies
* Notes: need proxy if AT/CG, multi-allelic, or low-count (available in <80% of traits)

**choose\_potential\_proxies**

* Input: list of variants needing proxies
* Output: dictionary linking each variant to a dataset of possible proxies (ID + r2 + alleles)
* Notes: uses 1000G LD reference (text file)
* Strategy for prioritization:
  + No AT/GC
  + Trait count >= 80% (use count\_traits\_per\_variant function above)
  + r2 >= 0.8
  + Then, choose based on highest trait count, then r2

**ld\_pruning**

* Input: list of original and proxy variants
* Output: final pruned list of variants
* Notes: LD link?

**fetch\_summary\_stats**

* Input: final list of pruned variants
* Output: z-score matrix, vector of minimum p-values, vector of median sample sizes
* Notes: align variants to the GWAS alleles

**prep\_z\_matrix**

* Input: matrix of sample size-adjusted z-scores (M\_variants x P\_traits), vector of median samples sizes per trait, vector of minimum p-values per trait
* Output: matrix ready for input to bNMF
* Notes: trait filtering by min. p-value < 0.05/N\_snps, trait pruning by |correlation| < 0.85, scaling of z-scores by round(mean(sqrt(medN))), replace NAs with 0, expanding each trait into non-negative pos and neg columns

**run\_bNMF**

* Input: matrix of preprocessed z-scores (M\_variants x 2\*P\_traits)
* Output: list containing H and W matrices as well as other useful statistics
* Notes: based on Jaegil’s code

**visualize\_bNMF**

* Input: list of bNMF run outputs (multiple iterations)
* Output: run summary table (K choices across runs), heatmaps visualizing variant and trait activity matrices