**Working Document on Hierarchical Dirichlet Process for Mutational Signature Discovery**

**Liu Mo and Steve Rozen**

**October 27, 2020**

This document describes the use of the R packages mSigHdp and hdpx to apply the hierarchical Dirichlet process (HDP) approach to the problem of discovering (“extracting”) mutational signatures from mutational spectra. We assume the reader is familiar with mutational signatures and the issues around extracting mutational signatures from “catalogs” of mutational spectra. The hdpx package was forked from Nicola Roberts’s hdp package. However it uses a different approach to combining “raw clusters” into “components” (the clusters of mutations due to particular mutational signatures.) The hdp and mSigHdp packages can be installed from github:

remotes::install\_github(repo = "steverozen/hdpx", ref = "0.3.0")

remotes::install\_github(repo = "steverozen/mSigHdp", ref = "1.1.1")

Broadly speaking, the hdpx package is agnostic to the application of HDP, while mSigHdp adapts HDP to mutational signature extraction. There is one important exception to this general design decision, however: parallelization is implemented in mSigHdp, even though this could be used for any application of the HDP.

**How to extract signatures**

**Important**: hdpx/mSigHdp has only been tested on Linux. Not recommended to run it on MS Windows.

The main function for extracting mutational signatures is mSigHdp::RunHdpxParallel. There is a demo script <github URL>.

R --vanilla < RunHdpxParallel.example.R > out.txt 2> err.txt &

This will create a folder containing (1) information of extracted signatures and their assignments; (2) diagnostic plots; and (3) burn-in and Gibbs sampling checkpoints.

**How to evaluate the results**

The files extracted.signatures.csv and extracted.signature.pdf contain the extracted signatures. Signatures named e.g “hdp.999” were found in > 90% of the posterior samples. Signatures named e.g. “potential hdp.999” were found in > 50% but < 90% of posterior samples, and need to be considered in light of other evidence. They are most likely generated by processes with low activity or a variant of ‘hdp.’ signatures.

‘inferred.exposure.count.pdf’ and ‘inferred.exposure.proportion.pdf’ shows the activity of each signature in each sample. This exposure information was retrieved from ‘comp\_dp\_counts’ matrix in each posterior sample. (the plots are made from inferred.exposures.csv’). We don’t think exposures (assignments) output from hdpx are very accurate, and we would recommend a separate step using other software to estimate assignments.

If ‘ground truth signatures’ are provided, there will be several plots and files generated for comparing extracted signatures with ground truth signatures.

mSigHdp generates diagnostic plots in the ‘Diagnostic\_Plots’ folder.

1.diagnostic.likelihood.pdf; 2.diagnostic.signatures.pdf; 3.diagnostics.data.assigned.pdf;

4.diagnostic.numcluster.pdf

The four plots above were inherited from Nicola Roberts’ hdp package. Please refer to the vignettes for the hdp package for details

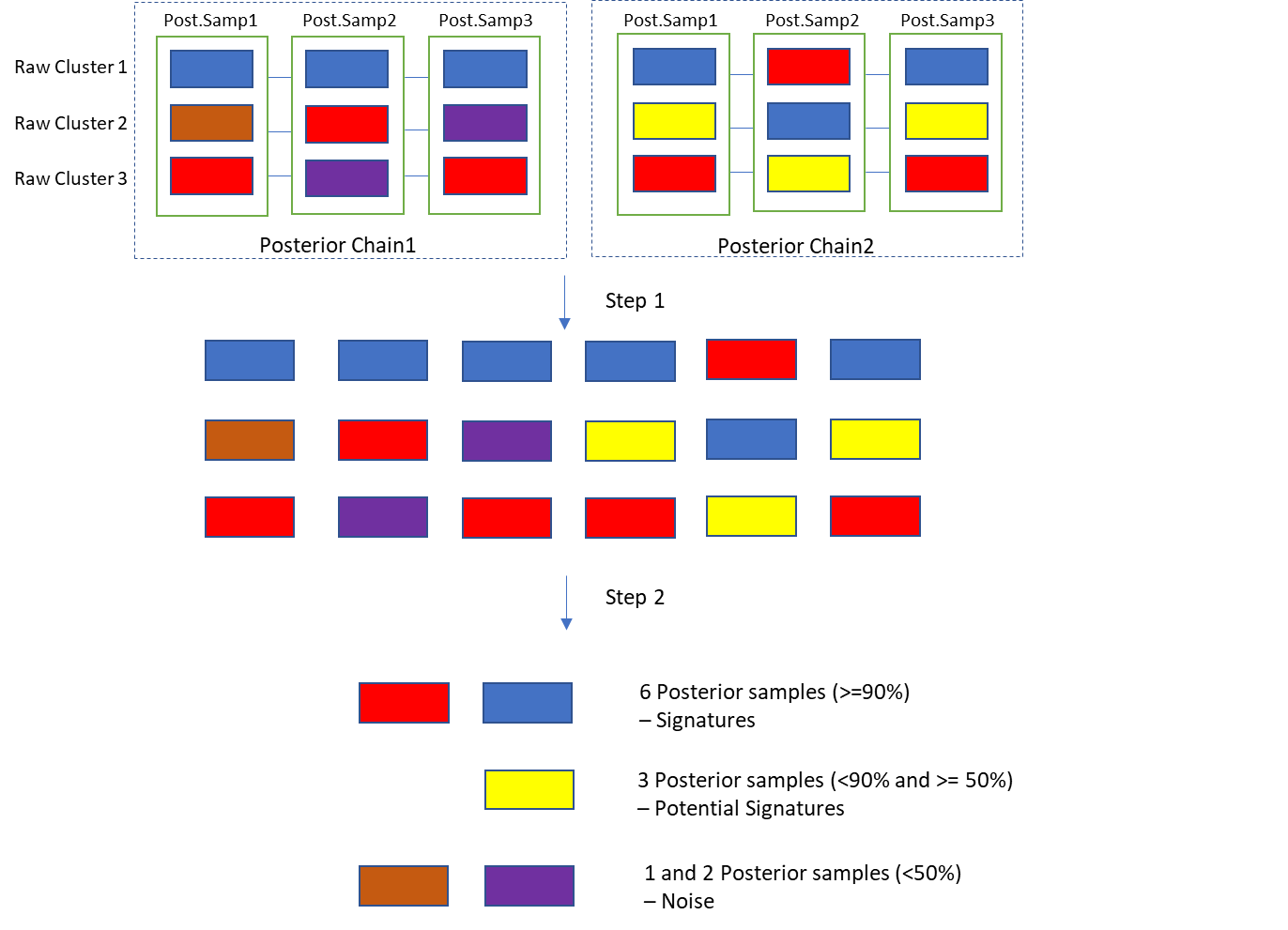
5. ‘diagnostic.hdp.signature.exposure.each.sample.pdf’: For each hdp signature, this file plots the five tumors with highest exposure proportion of the signature. This is useful in understanding which tumors contribute to the signature extraction most and a preliminary decision on if a ‘potential hdp’ signature should be included.

6. ‘component.distribution.in.posterior.samples.pdf’: this plotting is still under development. Its purpose is to show that for each signature, the presence of raw clusters that contributing to this signature across the posterior chain. For example, some signatures are contributed by raw clusters from all posterior chains while some are only contributed by half of or one posterior chain.

**Background**

mSigHdp::RunHdpxParallel incorporates the burn-in and Gibbs sampling processes, followed up by extracting signatures from raw clusters collected by Gibbs sampling, and improved diagnostic plotting with visualization on signature and exposures. One key parameter to prevent HDP from over-splitting or under-splitting is the proper setting on prior gamma distribution of concentration parameters. We recommended the gamma distribution with alpha (shape) = 1 and beta (rate) = 20 for as the concentration hyperparameter for single base substitution signature extraction; and alpha = 1 and beta = 50 for doublet base substitution and indel signature extraction.

This diagram shows the new approach to combining “raw clusters” into “components”.



We assume a very simple case with two independent MCMC chains 3 posterior samples per chain. There are 3 raw clusters in each posterior sample. Each color represents a raw cluster of mutations generated by a mutational process. Step 1 includes removal of highly identical clusters in the same posterior samples (not shown in the diagram) and put all clusters into a matrix. Step 2 includes a hierarchical clustering to combine raw clusters with high similarity (highly similar clusters are most likely generated by the same mutational process) into components. We then summarized the number of posterior samples that contribute to the components. In the diagram, the ‘red’ component were contributed by ‘red’ raw clusters from all 6 posterior samples, which means that ‘red’ component was consistently found in all posterior samples and it is very likely to be generated by a mutational process; in contrast, ‘brown’ component was contributed by ‘brown’ raw clusters from only 1 posterior sample, which means that the component was very likely to be found by randomness.

As the diagram shows, we defined 90% as a confidence level and 50% as a noise level. This is the recommended setting, however, can be changed according to prior-knowledge of the dataset. For example, if it is known that there is a lowly active mutational process in the dataset, confidence or noise level can be set to a lower value.