

Supplemental Materials for
Bayesian Clustering of Transcription
Factor Binding Motifs

Shane T. Jensen
Department of Statistics
The Wharton School, University of Pennsylvania

Jun S. Liu
Department of Statistics
Harvard University

1 Implementation Details for JASPAR and Combined Applications

Our Bayesian hierarchical clustering model is implemented by a Gibbs sampling program which is available as suite of Perl scripts that can run on any platform which has Perl installed. Our suite of Perl scripts is available, along with documentation, at the following website:

<http://stat.wharton.upenn.edu/~stjensen/research/cluster.html>

Since our clustering model is implemented using a Markov-chain Monte Carlo algorithm, it is important to evaluate whether or not our Gibbs sampling iterations have converged to our desired posterior distribution. Following the recommendation of Gelman and Rubin (1992), we started separate chains of our algorithm from different initial partitions \mathbf{z}^0 . Two examples of our starting partitions which we focus on in this section are the “each-in-own” partition: $z_i^0 = i$ for $i = 1, \dots, n$ and the “all-in-one” partition: $z_i^0 = 1$ for $i = 1, \dots, n$. Several other chains were also run from “random” partitions, i.e. clustering indicators z_i^0 drawn from a discrete uniform distribution. Each Gibbs sampling chain was run for 1000 iterations in total. The computational burden of these two applications was substantial: on a Apple desktop machine, 15 hours were needed to run the JASPAR application and 125 hours needed to run the combined application. However, this burden could have been significantly reduced by the use of a grid cluster of parallel machines instead of a single desktop machine. We monitored convergence using several statistics which were calculated at each iteration of our Gibbs sampler. These statistics were:

1. the number of multiple-member clusters,
2. the average size of multiple-member clusters, and
3. the posterior density of the current partition.

Figure 1 gives time series plots of these three statistics for our JASPAR application (Section 3 of the manuscript). It should be noted that the initial values were not included because their large difference between chains meant that subsequent differences between the chains could not be visually detected. Figure 2 gives time series plots of these three statistics for our combined JASPAR + TRANSFAC application (Section 4 of the manuscript). Based on the plots given in Figure 1 and Figure 2, we decided that convergence had certainly occurred within the first 100 or so iterations for both our JASPAR and combined JASPAR+ TRANSFAC applications, and that the chains were well-mixed after that point. These first 100 iterations in each chain were then discarded as “burn-in”. Although convergence occurred relatively quickly in our applications, it is certainly possible to improve the efficiency of our algorithm through more complicated “split” and “merge” moves, as suggested in Neal (2000) and Dahl (2003).

We also evaluated the autocorrelation of these three statistics across the remaining iterations. Figure 3 gives the estimated autocorrelation function for our three clustering statistics in the JASPAR application (Section 3 of the manuscript). Figure 4 gives the estimated autocorrelation function for our three clustering statistics in the combined (JASPAR+TRANSFAC) application (Section 4 of the manuscript). The autocorrelation is generally very low, with the exception of the lag-10 correlation in the combined application, which is a consequence of our width/alignment sampling scheme that occurs every 10 iterations.

2 Comparison of Prior Specifications for Combined Application

In Section 3.4 of the manuscript, we examined the effect of prior specification on the results of our JASPAR application. In this section, we examine our larger combined (JASPAR + TRANSFAC) application for differences in the clustering results between using the Dirichlet process prior versus the uniform clustering prior. Figure 5 gives the distributions of the number of multiple-member clusters and the average cluster size between the Dirichlet process prior and the uniform clustering prior. Examining Figure 5, we see that the difference in results between the two priors is more noticeable for this combined application than our JASPAR application (Section 3.4 of the manuscript). Use of the Dirichlet process prior leads to a smaller number of clusters with larger average size than the uniform clustering prior, though this difference is small relative to the differences in our prior simulation results (Section 2.6 of the manuscript). When we examined the best partitions produced by the two different priors, we saw only small differences, especially among the strongest clusters in the partitions.

3 Comparison of Posterior Width Intervals for Combined Application

In Section 3.3 of the manuscript, we examined the effect on our JASPAR application of our novel extension to allow the core motif width to vary between clusters. In this section, we replicate that examination for our larger *combined* application to see if there is also substantial variability between matrices in terms of core motif width. Figure 6 is a plot of 95% posterior intervals for the core width of each matrix in our combined collection of JASPAR and TRANSFAC matrices. Similar to the JASPAR-only results in Section 3.3 of the manuscript, the 95% posterior intervals in Figure 6 are quite different between motifs, with some motifs having a wide interval while other motifs have an interval consisting only of the minimum motif width of 6 bps, which in many cases is because the raw data matrix is only 6 bps wide. As mentioned in our main manuscript, 87% of the motifs in the combined dataset had 95% posterior intervals for the core width which covered more than a single fixed value. In many cases (20% of motifs), the posterior interval for the motif width does not even include the *a priori* expected motif width of 8 bps. Just as with the JASPAR-only application, the variation within and between these intervals for our combined application suggests that considering core widths as fixed and known would result in a substantial loss of information.

4 Full Results for JASPAR and Combined Applications

Our available results consist of the following four files:

1. `jaspar.pairwise.txt`: list of pairwise clustering probabilities p_{ij} for all motif pairs i and j with $p_{ij} > 0.001$ from our JASPAR application
2. `jaspar.partition.txt`: list of all multiple-member clusters in our best partition \hat{z} from our JASPAR application

3. `combined.pairwise.txt`: list of pairwise clustering probabilities p_{ij} for all motif pairs i and j with $p_{ij} > 0.001$ from our Combined application
4. `combined.partition.txt`: list of all multiple-member clusters in our best partition \hat{z} from our Combined application

All of these files can be downloaded from the following website:

<http://stat.wharton.upenn.edu/~stjensen/research/cluster.html>

References

- Dahl, D. (2003). An improved merge-split sampler for conjugate dirichlet process mixture models. *Department of Statistics, University of Wisconsin Technical Report 1086*.
- Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science* **7**, 457–472.
- Neal, R. (2000). Markov chain sampling methods for dirichlet process mixture models. *Journal of Computational and Graphical Statistics* **9**, 249–265.

Figure 1: Time series plots of three clustering statistics in our JASPAR application

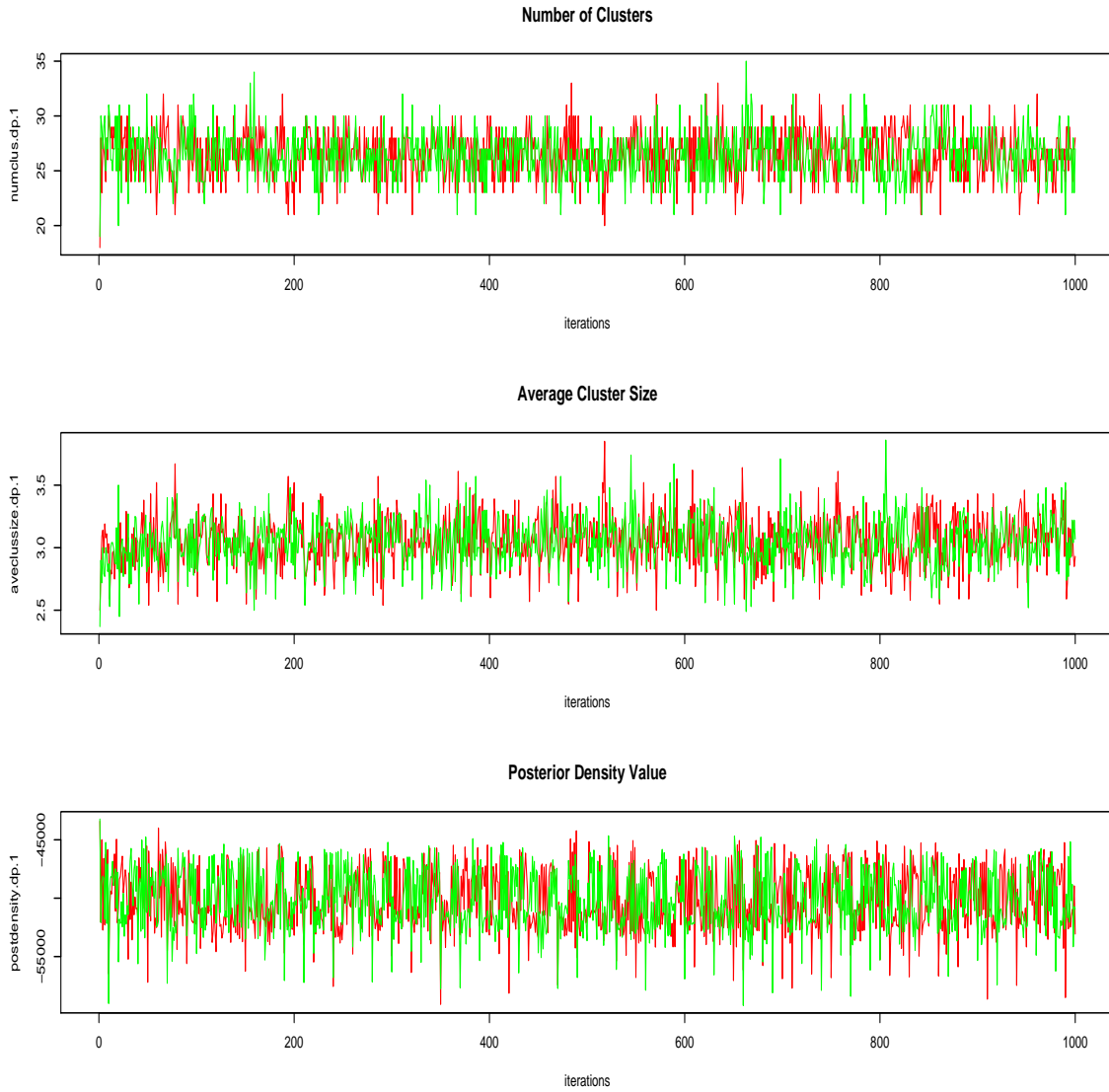


Figure 2: Time series plots of three clustering statistics in our Combined application

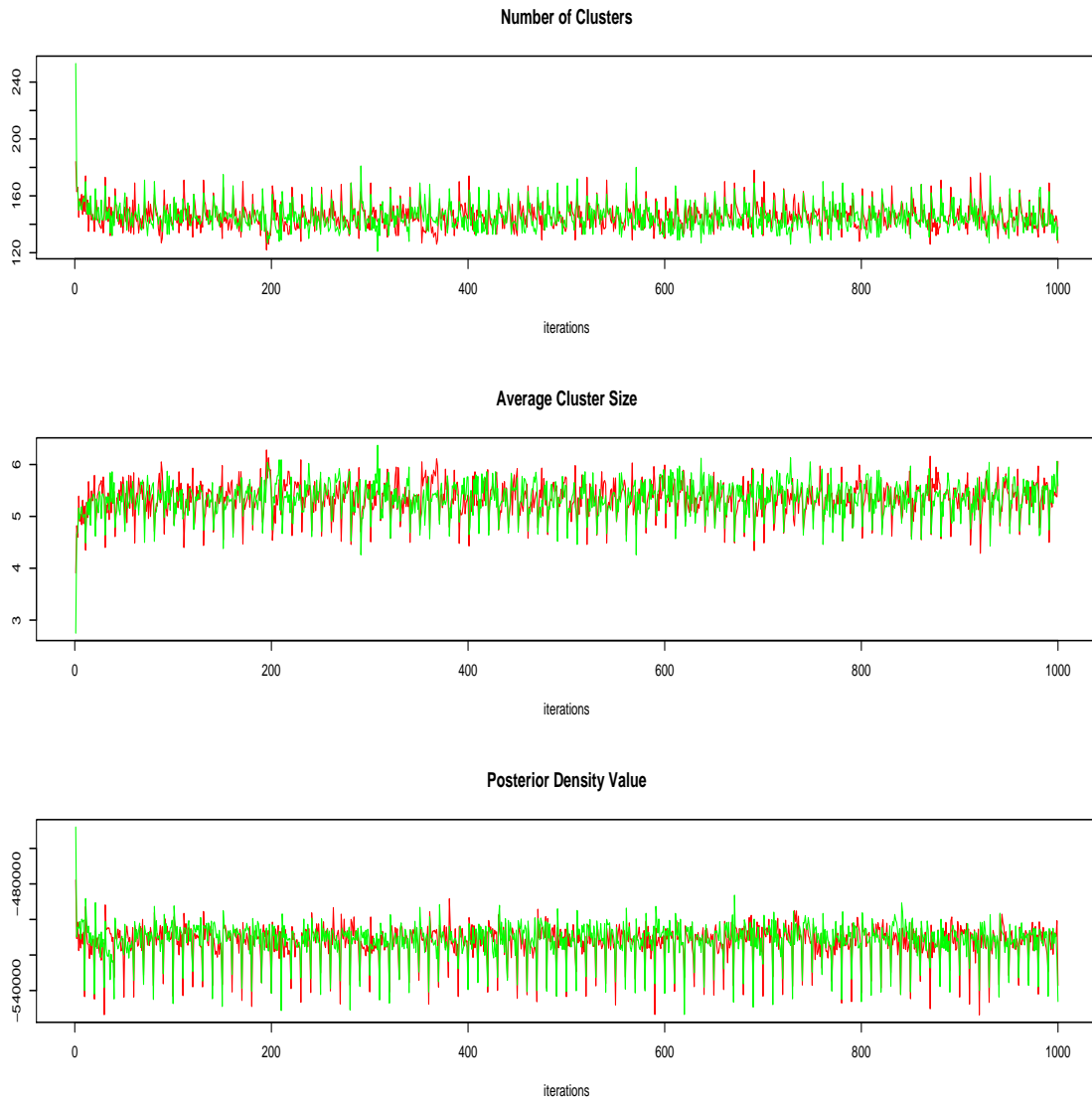


Figure 3: Autocorrelation function for the three clustering statistics in our JASPAR application

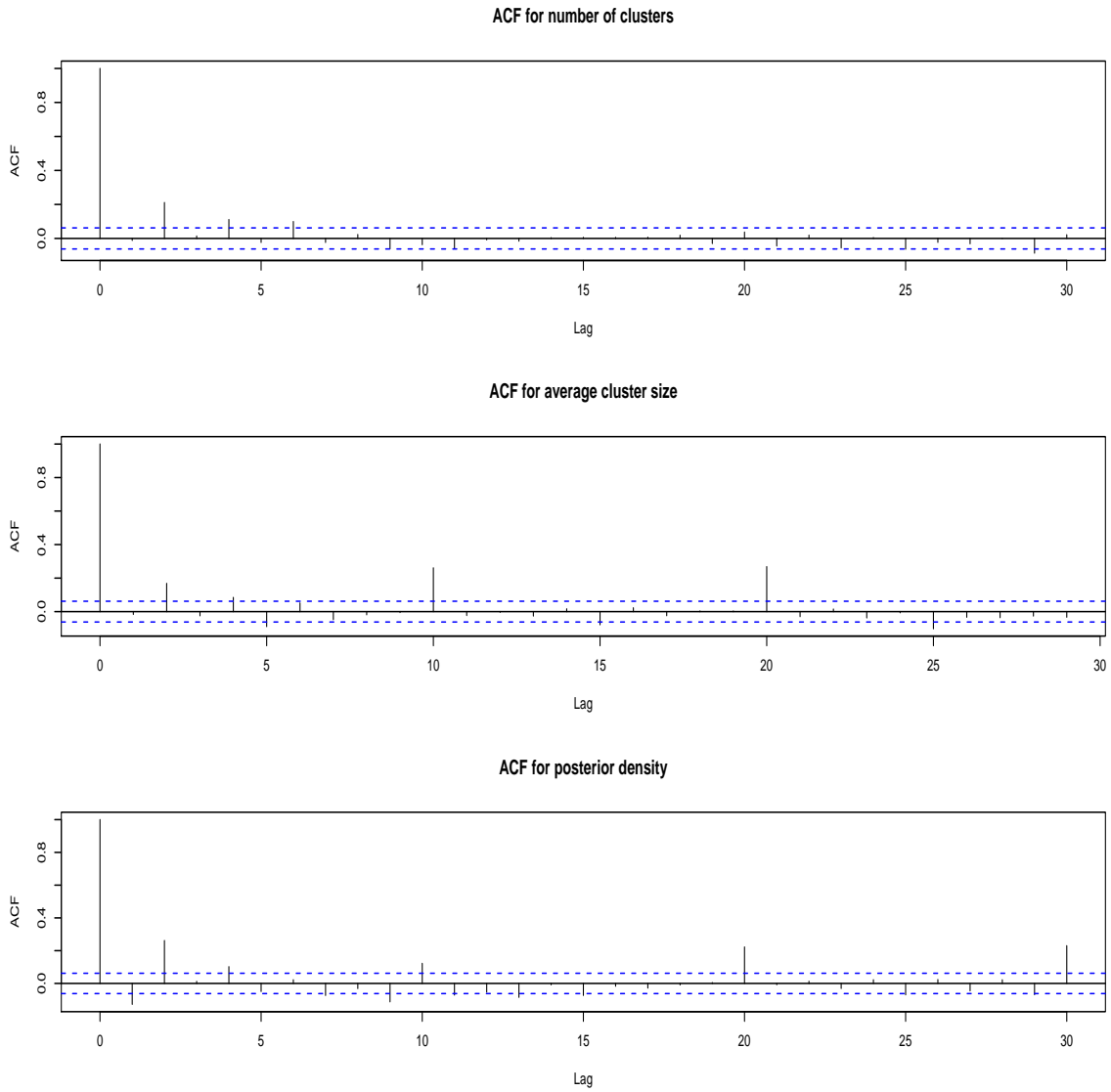


Figure 4: Autocorrelation function for the three clustering statistics in our Combined application

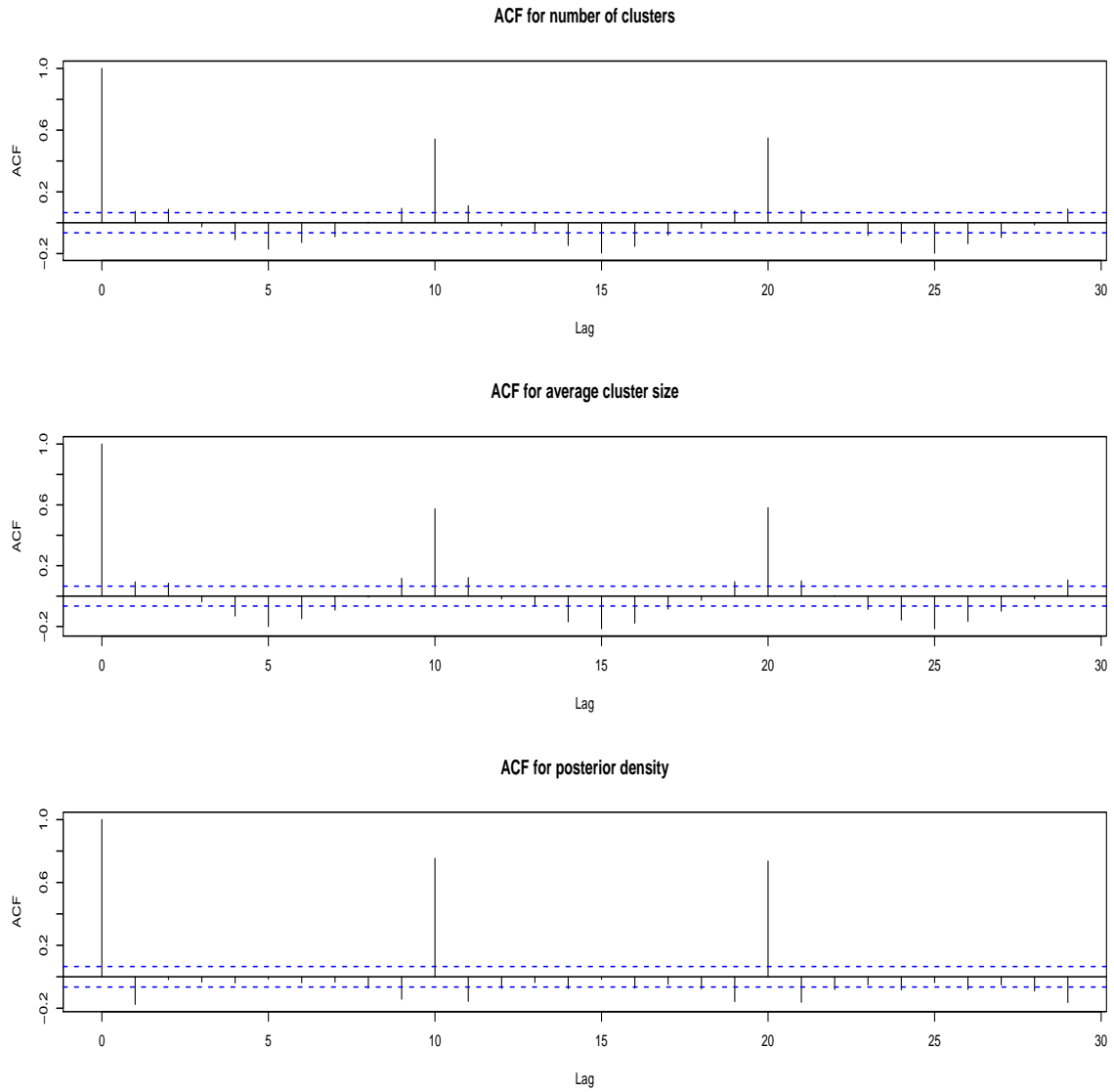


Figure 5: Comparing the uniform and DP models for Combined application

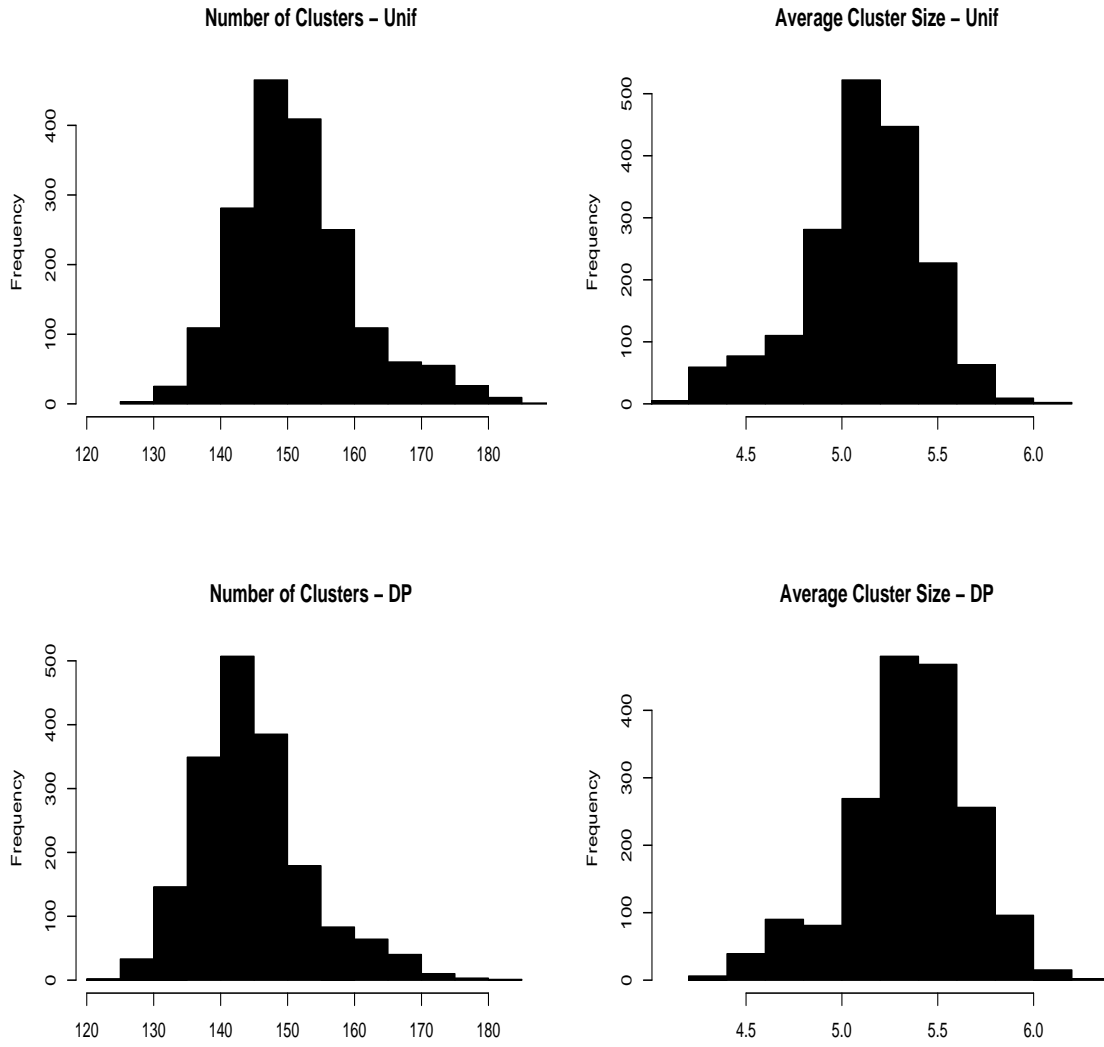


Figure 6: 95% posterior intervals of each motif width in Combined dataset

