Supplement for *Variational Boosting: Iteratively Refining Posterior Approximations*

A. Initializing Components

Introducing a new component requires initialization of component parameters. When our component distributions are mixtures of Gaussians, we found that the optimization procedure is sensitive to initialization. This section describes an importance-weighting scheme for initialization that produces (empirically) good initial values of component and mixing parameters.

Conceptually, a good initial component is located in a region of the target $\pi(x)$ that is underrepresented by the existing approximation $q^{(C)}$. A good initial weight is close to the proportion of mass in the unexplained region. Following this principle, we construct this component by first drawing importance-weighted samples from our existing approximation

$$x^{(\ell)} \sim q^{(C)}\,, \quad w^{(\ell)} = \frac{\pi(x^{(\ell)})}{q^{(C)}(x^{(\ell)})} \qquad \text{for } \ell = 1, \dots, L. \tag{9}$$

The samples with the largest weights $w^{(\ell)}$ tell us where regions of the target are poorly represented by our approximation. In fact, as L grows, and if $q^{(C)}$ is "close" enough to π, we can interpret $\{x^{(\ell)}, w^{(\ell)}\}$ as a weighted sample from π. Based on this interpretation, we can fit a mixture distribution (or *some components* of a mixture distribution) to this weighted sample using maximum likelihood, and recover a type of target approximation. For mixture distributions, an efficient inference procedure is Expectation-Maximization (EM) (Dempster et al., 1977).

This approach, however, presents a few complications. First, we must adapt EM to fit a *weighted* sample. Second, importance weights can suffer from extremely high variance — one or two $w^{(\ell)}$ values may be extremely large compared to all other weights. This destabilizes our new component parameters and mixing weight, particularly the variance of the component. Intuitively, if a single weight $w^{(\ell)}$ is extremely large, this would correspond to many samples being located in a single location, and maximum likelihood with EM would want to shrink the variance of the new component to zero right on that location. To combat this behavior, we use a simple method to break up the big weights using a resampling and re-weighting step before applying weighted EM. Empirically, this improves our new component initializations and subsequent ELBO convergence.

Weighted EM Expectation-maximization is typically used to perform maximum likelihood in latent variable models. Mixture distributions are easily represented with latent variables — a sample's latent variable corresponds to the mixture component that produced it. EM starts with some initialization of model parameters (e.g.,component means, variances and mixing weights). The algorithm then iterates between two steps: 1) the *E-step*, which computes the distribution over the latent variables given the current setting of parameters, and 2) the *M-step*, which maximizes the *expected complete data log-likelihood* with respect to the distributions computed in the E-step.

We suppress details of the general treatment of EM, and focus on EM for mixture models as presented in (Bishop, 2006). For mixture distributions, the E-step computes "responsibilities", or the probability that a datapoint came from one of the components. The M-step then computes a weighted maximum likelihood, where the log-likelihood of a datapoint for a particular component is weighted by the associated "responsibility". This weighted maximum likelihood is an easy entry-point for an additional set of weights — weights associated with each datapoint from the importance-weighting.

More concretely, for a sample of data, $x^{(\ell)}$, C mixture components, and current mixture component parameters and weights $\lambda = \{\rho_c, \lambda_c\}_{c=1}^{C}$, the E-step computes the following quantities

$$\begin{aligned}\gamma_c^{(\ell)} &= p(z^{(\ell)} = c | x^{(\ell)}, \lambda) \\ &\propto p(x^{(\ell)} | z^{(\ell), \lambda_c} = c) p(z^{(\ell)} = c)\end{aligned}$$

where $\gamma_c^{(\ell)}$ is the "responsibility" of cluster c for datapoint ℓ. The M-step then computes component parameters by a weighted maximum likelihood

$$\lambda_c^* = \arg\max_{\lambda} \sum_{\ell=1}^{L} \gamma_c^{(\ell)} \cdot \ln p(x^{(\ell)} | z^{(\ell)} = c, \lambda_c)\,.$$

To incorporate importance weights $w^{(\ell)}$, we only need to slightly change the M-step:

$$\lambda_c^* = \arg\max_\lambda \sum_{\ell=1}^{L} w^{(\ell)} \cdot \gamma_c^{(\ell)} \cdot \ln p(x^{(\ell)} | z^{(\ell)} = c, \lambda_c) \,.$$

Because we are adding a new component, we would like our weighted EM routine to leave the remaining components unchanged. For instance, we want $\lambda_1, \dots, \lambda_{C-1}$ to be fixed, while λ_C is free to explain the weighted sample. This can be accomplished in a straightforward manner by simply clamping the first $C-1$ parameters during the M-step.

Resampling importance weights If our current approximation $q^{(C)}$ is sufficiently different in certain regions of the posterior, then some weights $w^{(\ell)}$ will end up being large compared to other weights. For instance, the objective $\text{KL}(q||p)$ tends to under-cover regions of the posterior, allowing $\pi(x)$ to be much larger than $q^{(c)}(x)$, meaning the weight associated with x will be large. This will create instability in the weighted EM approximation — likelihood maximization will want to put a zero-variance component on the single highest-weighted sample, which does not accurately reflect the local curvature of $\pi(x)$. To combat this, we construct a slightly more complicated proposal distribution. Conceptually, we first create this naïve importance-weighted sample, and then find samples with outlier weights, and break those samples up. We do this by constructing a new proposal distribution that mixes the existing proposal, $q^{(C)}$, and component means located at the outlier samples. We define this proposal to be

$$q^{(IW)}(x) \propto p_0 q^{(C)}(x) + \sum_{\ell \in \mathcal{O}} w^{(\ell)} \mathcal{N}(x | x^{(\ell)}, \Sigma^{(\ell)}) \quad (10)$$

where $\ell \in \mathcal{O}$ denote the set of outlier samples from our original sample, and $p_0 = 1 - \sum_{\ell \in \mathcal{O}} w^{(\ell)}$ is the mass not placed on outlier samples. The variance of each outlier component, $\Sigma^{(\ell)}$ is set to some heuristic value — we typically use the diagonal of the covariance of $q^{(C)}$ as a good-enough guess.

We then create a new importance-weighted sample, using $q^{(IW)}$ and $\pi(x)$ just as we did before. By placing new components (with some non-zero variance) on the outlier samples, which are known to be in a region of high target probability and low approximate probability, we assume that there is more local probability around that region that needs to be explored. This allows us to inflate the local variance of the samples in this region — the region that weighted EM will place a component. Algorithm 1 unites the components from above sections into our final initialization procedure.

B. Fitting the Rank

To specify the ELBO objective, we need to choose a rank r for the component covariance. There are a many ways to decide on the rank of the variational approximation, some more appropriate for certain settings given dimensionality and computation constraints. For instance, we can greedily incorporate new rank components. Alternatively, we can fit a sequence of components $r = 1, 2, \dots, r_{max}$, and choose the best result (in terms of KL). In the Bayesian neural network model, we report results for a fixed schedule of ranks. In the hierarchical Poisson model, we monitor the change in marginal variances to decide the appropriate rank. In both cases, we require a stopping criterion. For a single Gaussian, one such criterion is the average change in marginal variances — if the marginal variation along each dimension remains the same from rank r to $r+1$, then the new covariance component is not incorporating explanatory power, particularly if marginal variances are of interest. As the $KL(q||\pi)$ objective tends to underestimate variances when restricted to a particular model class, we observe that the marginal variances grow as new covariance rank components are added. When fitting rank $r+1$, we can monitor the average absolute change in marginal variance (or standard deviation) as more covariance structure is incorporated. Figure 5 in this supplement depicts this measurement for the $D = 37$-dimensional ‘frisk‘ posterior.

To justify sequentially adding ranks to mixture components we consider the KL-divergence between a rank-r Gaussian approximation to a full covariance Gaussian, $\text{KL}(q_r||p)$, where $q_r(\theta) = \mathcal{N}(0, \mathcal{I}(v) + \sum_{l=1}^{r} f_k f_k^\intercal)$ and $p(\theta) = \mathcal{N}(0, \Sigma)$. For simplicity, we assume both distributions have zero mean. If the true posterior is non-Gaussian we will attempt to approximate the best full-rank Gaussian with a low-rank Gaussian thus suffering an unrepresentable KL-divergence between the family of Gaussians and the true posterior. We also assume that the diagonal component, $\mathcal{I}(v)$, and the first $r-1$ columns of $F = [f_1, \dots, f_r]$ are held fixed. Then we have

$$\begin{aligned}
&\text{KL}(q_r||p) \\
&= \frac{1}{2}\Big(\text{tr}\left(\Sigma^{-1}\left(\mathcal{I}(v) + \sum_{l=1}^{r} f_l f_l^\intercal\right)\right) \\
&\quad - k + \log\det\Sigma \\
&\quad - \log\det\left(\mathcal{I}(v) + \sum_{l=1}^{r} f_l f_l^\intercal\right)\Big)
\end{aligned}$$

which we differentiate with respect to v_r, remove terms that

Algorithm 1 Importance-weighted initialization of new components. This algorithm takes in the target distribution, $\pi(x)$, the current approximate distribution $q^{(C)}(x)$, and a number of samples L. This returns an initial value of new component parameters, λ_{C+1} and a new mixing weight ρ_{C+1}.

```
procedure InitComp($\pi, q^{(C)}, L$)
    $x^{(\ell)} \sim q^{(C)}$ for $\ell = 1, \ldots, L$                                          ▷ sample from existing approx
    $w^{(\ell)} \leftarrow \frac{\pi(x^{(\ell)})}{q^{(C)}(x^{(\ell)})}$                                          ▷ set importance weights
    $\mathcal{O} \leftarrow$ outlier-weights($\{w^{(\ell)}\}$)
    $q^{(IW)} \leftarrow$ make-mixture($\mathcal{O}, \{w^{(\ell)}, x^{(\ell)}\}, q^{(C)}$)              ▷ break up big weights
    $x_r^{(\ell)} \sim q^{(IW)}$ for $\ell = 1, \ldots, L$                                       ▷ sample from new mixture
    $w_r^{(\ell)} \leftarrow \frac{\pi(x_r^{(\ell)})}{q^{(IW)}(x^{(\ell)})}$                                      ▷ re-sampled importance weights
    $\lambda_{C+1}, \rho_{C+1} \leftarrow$ weighted-em($\{x_r^{(\ell)}, w_r^{(\ell)}\}$)                        ▷ fit new component
    return $\lambda_{C+1}, \rho_{C+1}$
end procedure
```

do not depend on v_r, and set to zero, yielding

$$\begin{aligned}
&\frac{\partial}{\partial v_r}\mathrm{KL}(q_r||p) \\
&= \frac{1}{2}\left[\Sigma^{-1}v_r - \left(\mathcal{I}(v) + \sum_{l=1}^{r} f_l f_l^{\intercal}\right)^{-1} v_r\right] = 0 \\
\rightarrow \Sigma^{-1}v_r &= \left(\underbrace{\mathcal{I}(v) + \sum_{l=1}^{r-1} f_l f_l^{\intercal}}_{C} + f_r f_r^{\intercal}\right)^{-1} v_r \\
&= \left(C^{-1} - \frac{C^{-1}f_r f_r^{\intercal}C^{-1}}{1 + f_r^{\intercal}C^{-1}f_r}\right) f_r.
\end{aligned}$$

We can thus determine the optimal f_r from the following equation

$$\left(\Sigma^{-1} - C^{-1}\right) f_r = \left(-\frac{C^{-1}f_r f_r^{\intercal}C^{-1}}{1 + ||f_r||_C^2}\right) f_r \qquad (11)$$

where we have defined $f_r^{\intercal}C^{-1}f_r = ||f_r||_C^2$. Eq. (11) is reminiscent of an eigenvalue problem indicating that the optimal solution for f_r should maximally explain $\Sigma^{-1} - C^{-1}$, i.e. the parameter space not already explained by $C = \mathcal{I}(v) + \sum_{l=1}^{r-1} f_l f_l^{\intercal}$. This provides justification for the previously proposed stopping criterion that monitors the increase in marginal variances since incorporating a new vector into the low-rank approximation should grow the marginal variances if extra correlations are captured. This is due to minimizing $\mathrm{KL}(q_r||p)$ which underestimates the variances when dependencies between parameters are broken.

C. Experiment Figures

C.1. Frisk Model

Figures 6 and 7 depict VBoost approximations of the 'frisk' model.

C.2. Bayes Neural Network Results

Table 3 depict out of sample log probability results for the Bayesian neural network as ranks vary.

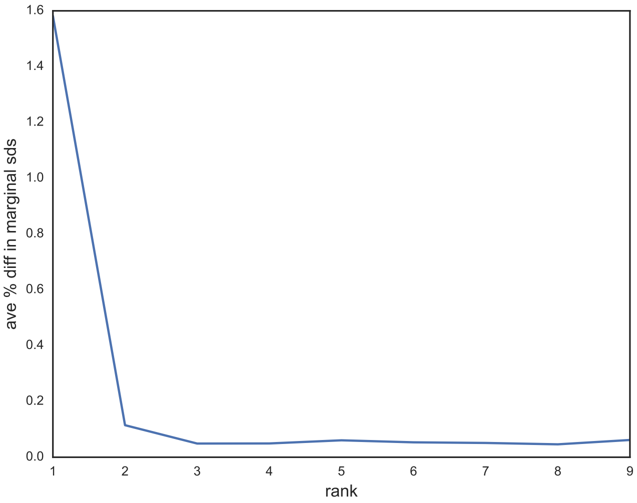
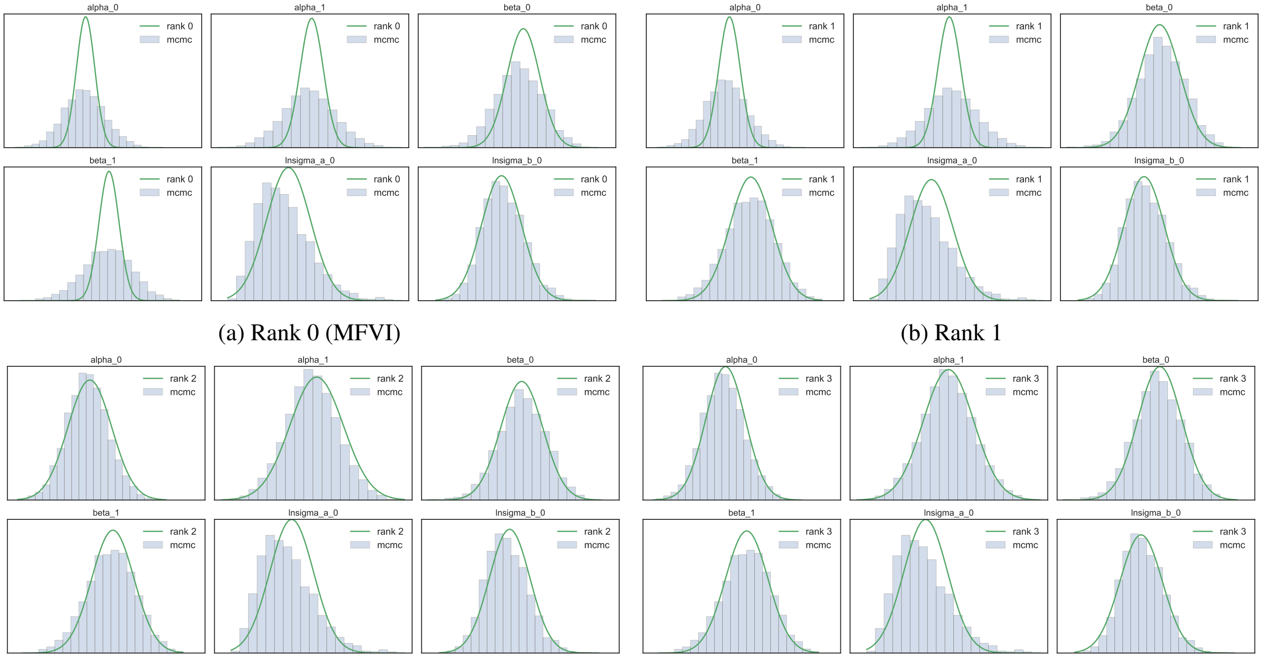


Figure 5. Mean percent change in marginal variances for the Poisson GLM. After rank 5, the average percent change is less than 5% — this estimate is slightly noisy due to the stochastic optimization procedure.



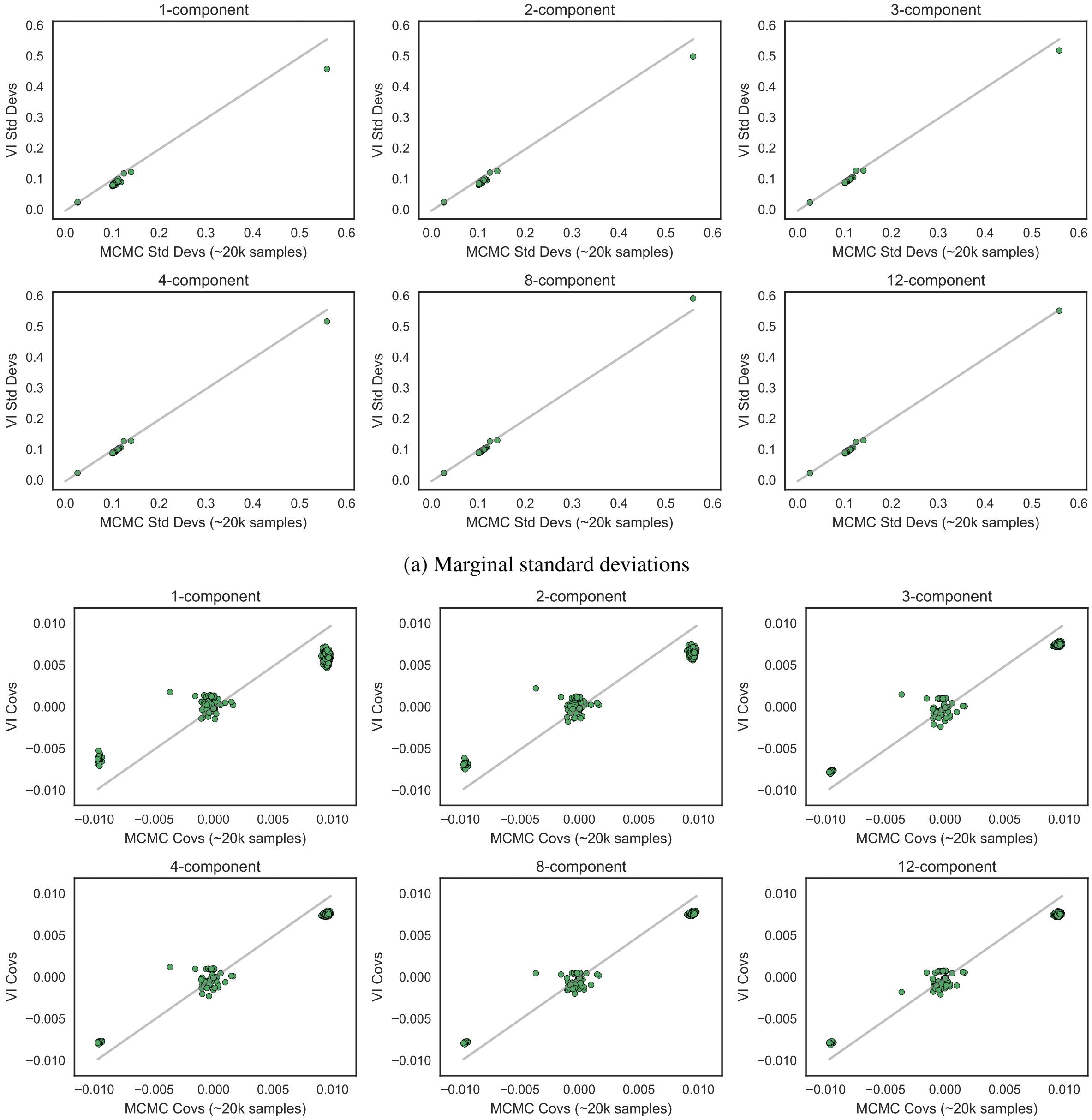
(a) Rank 0 (MFVI)

(b) Rank 1

(c) Rank 2

(d) Rank 3

Figure 6. Comparison of single Gaussian component marginals by rank for a 37-dimensional Poisson GLM posterior. The top left plot is a diagonal Gaussian approximation. The next plots show the how the marginal variances inflate as the covariance is allotted more capacity.



(a) Marginal standard deviations

(b) Pairwise covariances

Figure 7. A comparison of standard deviations and covariances for the `frisk` model. The MCMC-inferred values are along the horizontal axis, with the variational boosting values along the vertical axis. While the rank 3 plus diagonal covariance structure is able to account for most of the marginal variances, the largest one is still underestimated. Incorporating more rank 3 components allows the approximation to account for this variance. Similarly, the non-zero covariance measurements improve as more components are added.

	pbp	mfvi	rank 5	rank 10	rank 15
wine	-0.990 (± 0.08)	-0.973 (± 0.05)	-0.972 (± 0.05)	**-0.972** (± 0.05)	-0.973 (± 0.05)
boston	-2.902 (± 0.64)	**-2.658** (± 0.18)	-2.670 (± 0.16)	-2.696 (± 0.14)	-2.743 (± 0.12)
concrete	**-3.162** (± 0.15)	-3.248 (± 0.07)	-3.247 (± 0.06)	-3.261 (± 0.06)	-3.286 (± 0.05)
power-plant	**-2.798** (± 0.04)	-2.812 (± 0.03)	-2.814 (± 0.03)	-2.838 (± 0.03)	-2.867 (± 0.02)
yacht	-0.990 (± 0.08)	-0.973 (± 0.05)	-0.972 (± 0.05)	**-0.972** (± 0.05)	-0.973 (± 0.05)
energy-efficiency	**-1.971** (± 0.11)	-2.451 (± 0.12)	-2.452 (± 0.12)	-2.469 (± 0.11)	-2.502 (± 0.09)

Table 3. Comparison of test log probability for PBP (Hernández-Lobato & Adams, 2015) to Variational Inference with various ranks. Each entry shows the average predictive performance of the model on a specific dataset and the standard deviation across the 20 trials — bold indicates the best average (though not necessarily statistical significance).