We propose a Bayesian deep learning method that does expressive inference over a carefully chosen subnetwork within a neural network, and show that this works better than doing crude inference over the full network.

1. MAP Estimation
   Use SGD to obtain a point estimate over the weights: 
   \[ W_{MAP} = \arg \max_W \left[ \log p(y | X, W) + \log p(W) \right] \]

2. Subnetwork Selection
   Find the subnetwork whose posterior is closest to the full network posterior in terms of Wasserstein distance:
   1) Estimate a factorized Gaussian posterior over all weights
   2) Subnetwork = weights with largest marginal variances

3. Bayesian Inference
   Use the linearized Laplace approximation to infer a full-covariance Gaussian posterior over the subnet. All other weights are fixed to their MAP estimates.
   \[ p(W | y, X) \approx \mathcal{N}(W; W_{MAP}^-, \tilde{H}^{-1}) \prod_r \delta(w_r - w_r^*) \]

4. Prediction
   Make predictions using the full network of mixed Bayesian/deterministic weights
   \[ p(y^* | X^*, y, X) \approx \int_W p(y^* | X^*, W) \mathcal{N}(W; W_{MAP}^-, \tilde{H}^{-1}) \prod_r \delta(w_r - w_r^*) \, dW \]