Bayesian Mixtures of Bernoulli Distributions

Laurens van der Maaten
Department of Computer Science and Engineering
University of California, San Diego

1 Introduction

The mixture of Bernoulli distributions [6] is a technique that is frequently used for the modeling of binary random vectors. They differ from (restricted) Boltzmann Machines in that they do not model the marginal distribution over the binary data space $\mathcal{X}$ as a product of (conditional) Bernoulli distributions, but as a weighted sum of Bernoulli distributions. Despite the non-identifiability of the mixture of Bernoulli distributions [3], it has been successfully used to, e.g., dichotomous perceptual decision making [1], text classification [7], and word categorization [4].

Mixtures of Bernoulli distributions are typically trained using an expectation-maximization (EM) algorithm, i.e. by performing maximum likelihood estimation. In this report, we develop a Gibbs sampler for a fully Bayesian variant of the Bernoulli mixture, in which (conjugate) priors are introduced over both the mixing proportions and over the parameters of the Bernoulli distributions. We develop both a finite Bayesian Bernoulli mixture (using a Dirichlet prior over the latent class assignment variables) and an infinite Bernoulli mixture (using a Dirichlet Process prior). We perform experiments in which we compare the performance of the Bayesian Bernoulli mixtures with that of a standard Bernoulli mixture and a Restricted Boltzmann Machine on a task in which the (unobserved) bottom half of a handwritten digit needs to be predicted from the (observed) top half of that digit.

The outline of this report is as follows. Section 2 describes the generative model of the Bayesian Bernoulli mixture. Section 3 described how inference is performed in this model using a collapsed Gibbs sampler. Section 4 extends the Bayesian Bernoulli mixture to an infinite mixture model, and described the requires collapsed Gibbs sampler. Section 6 presents the setup and results of our experiments on a handwritten digit prediction task.

2 Generative model

A Bayesian mixture of Bernoulli distributions models a distribution over a D-dimensional binary space $\mathcal{X}$. The generative model of the Bayesian Bernoulli mixture is shown graphically in Figure 1.

The generative model for generating a point $\mathbf{x}$ is given below:

$$p(\boldsymbol{\pi}|\alpha) = Dirichlet(\boldsymbol{\pi}|\frac{\alpha}{K}, \ldots, \frac{\alpha}{K}), \tag{1}$$

where α is a hyperparameter of the Dirichlet prior, the so-called concentration parameter[1]. The vector $\boldsymbol{\pi}$ is a K-vector describing the mixing weights, where K is the number of mixture components (in the case of the infinite mixture model, $K \to \infty$).

$$p(\mathbf{z}|\boldsymbol{\pi}) = Discrete(\mathbf{z}|\boldsymbol{\pi}), \tag{2}$$

where $\mathbf{z}$ is a K-vector describing the class assignment (note that $z_k \in \{0, 1\}$ and $\sum_k z_k = 1$).

$$p(\mathbf{a}_k|\beta, \gamma) = Beta(\mathbf{a}_k|\beta, \gamma), \tag{3}$$

[1] For simplicity, we assume a symmetric Dirichlet prior, i.e. we assume $\forall k : \alpha_k = \alpha/K$.

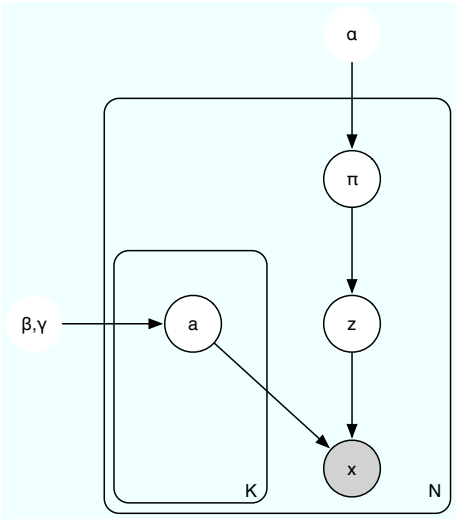


Figure 1: Generative model of infinite mixture of Bernoulli distributions.

where $\mathbf{a}_k$ is a D-vector, and β and γ are hyperparameters of the Beta-prior.

$$p(\mathbf{x}|\{\mathbf{a}_1,\ldots,\mathbf{a}_K\},\mathbf{z}) = \sum_{k=1}^{K} \left(Bern(\mathbf{x}|\mathbf{a}_k)\right)^{z_k}. \tag{4}$$

Note that our model does not include weakly informative hyperpriors over the hyperparameters, as has been proposed for, e.g., the infinite mixture of Gaussians [9] or the infinite Hidden Markov Model [2]. We leave such an extension to future work.

The marginal distribution of the model over the data space is given by

$$p(\mathbf{x}) = \sum_{\mathbf{z}} \int_0^1 p(\mathbf{x}|\mathbf{z},\mathbf{a}) \left[\int_{\Delta K} p(\mathbf{z}|\boldsymbol{\pi}) p(\boldsymbol{\pi}|\alpha) d\boldsymbol{\pi}\right] p(\mathbf{a}|\beta,\gamma) d\mathbf{a}. \tag{5}$$

3 Inference

As analytically computing the posterior over cluster assignments $p(\mathbf{z}|\mathbf{x})$ is intractable, we resort to using Markov Chain Monte Carlo (MCMC) to do inference in the model. In particular, we derive the conditionals that are required to run a collapsed Gibbs sampler in the model. We first work out the integral over the mixing proportions $\boldsymbol{\pi}$, and subsequently, work out the integral over the Bernoulli parameters $\mathbf{a}$. The Gibbs sampler will sample the cluster assignments $\mathbf{z}$.

The joint distribution over the assignment variables $\mathbf{z}$ can be obtained by integrating out[2] the mixing proportions $\boldsymbol{\pi}$ as follows[3]

$$p(\mathbf{z}_1,\ldots,\mathbf{z}_N) = \int_{\Delta K} p(\mathbf{z}_1,\ldots,\mathbf{z}_n|\boldsymbol{\pi}) p(\boldsymbol{\pi}) d\boldsymbol{\pi} \tag{6}$$

$$= \frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^K} \int_{\Delta K} \prod_{k=1}^{K} \left[\pi_k^{(\alpha/K-1)} \prod_{n=1}^{N} \pi_k^{z_{nk}}\right] d\boldsymbol{\pi} \tag{7}$$

$$= \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)} \prod_{k=1}^{K} \frac{\Gamma(N_k+\alpha/K)}{\Gamma(\alpha/K)}, \tag{8}$$

where Z is a normalization constant, N_k is the number of point assigned to class k, and K is the number of classes in the model. The conditional distribution of the assignment variable z_{nk} for data point n,

[2]We use the standard Dirichlet integral in this result: $\int_{\Delta K} Dirichlet(\mathbf{x}|\boldsymbol{\alpha}) d\boldsymbol{\alpha} = \int_{\Delta K} \prod_{k=1}^{K} x_k^{(\alpha_k-1)} d\alpha_k = \frac{\prod_{k=1}^{K}\Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K}\alpha_k)}$.

[3]Throughout the report, we omit the hyperparameters from the notation where possible, in order to prevent the notation from becoming too cluttered.

given the other assignment variables for data point n and all other variables, can be obtained from the above expression by fixing all but one cluster assignment, to give

$$p(z_{nk} = 1|\mathbf{z}_{n-k}) = \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha}, \tag{9}$$

where N_{-nk} represents the number of data points assigned to class k, not counting data point n (i.e., $N_{-nk} = \sum_{i=1}^{n-1} z_{ik} + \sum_{i=n+1}^{N} z_{ik}$).

The conditional distribution over the Bernoulli parameters a_{kd} (a_{kd} represents the d-th variable of $\mathbf{a}_k$), given all other parameters, is given by

$$\begin{aligned}
p(\mathbf{a}_k|\mathbf{z}, \mathcal{D}) &\propto p(\mathbf{a}_k|\beta, \gamma)p(\mathcal{D}|\mathbf{a}_k, \mathbf{z}) && (10)\\
&= Beta(\mathbf{a}_k|\beta, \gamma) \prod_{n=1}^{N} (Bern(\mathbf{x}_n|\mathbf{a}_k))^{z_k} && (11)\\
&= \frac{1}{Z} \prod_{d=1}^{D} \left[a_{kd}^{(\beta-1)}(1 - a_{kd})^{(\gamma-1)} \prod_{n\in\mathcal{C}_k} a_{kd}^{x_{nd}}(1 - a_{kd})^{(1-x_{nd})} \right] && (12)\\
&= \frac{1}{Z} \prod_{d=1}^{D} \left[a_{kd}^{(\beta-1+\sum_{n\in\mathcal{C}_k} x_{nd})}(1 - a_{kd})^{(\gamma-1+N_k-\sum_{n\in\mathcal{C}_k} x_{nd})} \right] && (13)\\
&= \prod_{d=1}^{D} Beta(a_{kd}|\beta + \sum_{n\in\mathcal{C}_k} x_{nd}, \gamma + N_k - \sum_{n\in\mathcal{C}_k} x_{nd}), && (14)
\end{aligned}$$

where $\mathcal{C}_k$ denotes the set of points that were assigned to class k (i.e., all points i for which $z_{ik} = 1$), and N_k denotes the cardinality of this set.

The conditional distribution is obtained by combining our results from Equation 14 and 9, and integrating out[4] the Bernoulli parameters $\mathbf{a}$ as follows

$$\begin{aligned}
p(z_{nk} = 1|\mathbf{z}_{n-k}, \mathcal{D}) &= \int_0^1 p(z_{nk} = 1|\mathbf{z}_{n-k})p(\mathbf{a}_k|\mathbf{z}_{n-k}, \mathcal{D})d\mathbf{a}_k && (15)\\
&= \int_0^1 \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha} \left[\prod_{d=1}^{D} Bern(x_{nd}|a_{kd})Beta(a_{kd}|\beta + \sum_{i\in\mathcal{C}_k} x_{id}, \gamma + N_k - \sum_{i\in\mathcal{C}_k} x_{id}) \right] d\mathbf{a}_k && (16)\\
&= \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha} \prod_{d=1}^{D} \left[\int_0^1 Bern(x_{nd}|a_{kd})Beta(a_{kd}|\beta + \sum_{i\in\mathcal{C}_k} x_{id}, \gamma + N_k - \sum_{i\in\mathcal{C}_k} x_{id})da_{kd} \right] && (17)\\
&= \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha} \prod_{d=1}^{D} \frac{1}{Z_{nd}} \left[\int_0^1 a_{kd}^{(x_{nd}+\beta+\sum_{i\in\mathcal{C}_k} x_{id}-1)}(1 - a_{kd})^{(-x_{nd}+\gamma+N_k-\sum_{i\in\mathcal{C}_k} x_{id})} da_{kd} \right] && (18)\\
&= \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha} \prod_{d=1}^{D} \frac{1}{Z_{nd}} B(x_{nd} + \beta + \sum_{i\in\mathcal{C}_k} x_{id}, -x_{nd} + \gamma + N_k - \sum_{i\in\mathcal{C}_k} x_{id} + 1), && (19)
\end{aligned}$$

where $\mathcal{C}_k$ and N_k do not include the current data point n, where Z_{nd} represents a normalization constant, and where $B(x, y)$ represents the beta function, i.e., $B(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x)+\Gamma(y)}$. The normalization term Z_{nd} requires some additional attention, as its presence allows us to sample from the above conditional

[4] We also use the Beta integral $\int_0^1 x^{(m-1)}(1 - x)^{(n-1)}dx = \frac{\Gamma(m)\Gamma(n)}{\Gamma(m+n)}$.

distribution without performing the (expensive) computation of beta functions. The normalization term Z_{nd} has the form

$$Z_{nd} = B(\beta + \sum_{i \in \mathcal{C}_k} x_{id}, \gamma + N_k - \sum_{i \in \mathcal{C}_k} x_{id}). \tag{20}$$

As a result, the value of the normalized beta evaluation when is $\frac{\beta + \sum_{i \in \mathcal{C}_k} x_{id}}{\beta + \gamma + N_k}$ when $x_{nd} = 1$, and $\frac{\gamma + N_k - \sum_{i \in \mathcal{C}_k} x_{id}}{\beta + \gamma + N_k}$ when $x_{nd} = 0$.

We can thus set up a collapsed Gibbs sampler that only samples the cluster assignments by sampling from the following conditional distribution

$$p(z_{nk} = 1 | \mathbf{z}_{n-k}, \mathcal{D}) = \frac{N_{-nk} + \frac{\alpha}{K}}{N - 1 + \alpha} \prod_{d=1}^{D} \left[\frac{(\beta + \sum_{i \in \mathcal{C}_k} x_{id})^{x_{nd}} (\gamma + N_k - \sum_{i \in \mathcal{C}_k} x_{id})^{(1 - x_{nd})}}{\beta + \gamma + N_k} \right]. \tag{21}$$

To prevent numerical problems, it is better to compute the product over the D dimensions in the log-domain (when D is large).

4 Infinite Bernoulli mixture

In the infinite Bernoulli mixture, the Dirchlet prior in Equation 1 is replaced by a Dirichlet Process

$$p(\boldsymbol{\pi} | \alpha) = DP(\boldsymbol{\pi} | \alpha), \tag{22}$$

where α is the concentration parameter of the Dirichlet Process. Most of the derivation remains similar, but the conditional distribution over the class assignment variables (given all other variables) changes as to reflect the distribution over unrepresented classes. Specifically, the probability of an assignment variable z_{nk} being on [8, 5] is given by

$$p(z_{nk} = 1 | \mathbf{z}_{n-k}, \mathcal{D}) = \begin{cases} \frac{N_{-nk}}{N - 1 + \alpha} \prod_{d=1}^{D} \left[\frac{(\beta + \sum_{i \in \mathcal{C}_k} x_{id})^{x_{nd}} (\gamma + N_k - \sum_{i \in \mathcal{C}_k} x_{id})^{(1 - x_{nd})}}{\beta + \gamma + N_k} \right], \text{iff } k \leq K_+ \\ \frac{\alpha}{N - 1 + \alpha} \left[\prod_{d=1}^{D} \int_0^1 Bern(x_{nd} | a_{kd}) Beta(a_{kd} | \beta, \gamma) da_{kd} \right], \text{iff } k = K_+ + 1 \\ 0, \text{iff } k > K_+ + 1, \end{cases} \tag{23}$$

where K_+ indicates the number of represented classes. The integral in the equation for the unrepresented class can be worked out as follows:

$$\int_0^1 Bern(x_{nd} | a_{kd}) Beta(a_{kd} | \beta, \gamma) da_{kd} = \frac{1}{B(\beta, \gamma)} \int_0^1 a_{kd}^{(x_{nd} + \beta - 1)} (1 - a_{kd})^{(x_{nd} + \gamma)} da_{kd} \tag{24}$$

$$= \frac{B(x_{nd} + \beta, \gamma - x_{nd} + 1)}{B(\beta, \gamma)}. \tag{25}$$

The integral thus has value $\frac{\beta}{\beta + \gamma}$ for $x_{nd} = 1$, which means that the prior belief that an observed variable is on under an unrepresented class is $\frac{\beta}{\beta + \gamma}$. The value for $x_{nd} = 0$ is $\frac{\gamma}{\beta + \gamma}$.

5 Predictive distribution

At each state of the Gibbs sampler, the predictive distribution comprises two parts: a part corresponding to the represented classes and a part corresponding to the unrepresented classes (in the finite mixture the second part is empty). Denoting the query-part of $\mathbf{x}$ by $\mathbf{x}_q$ and the non-query part by $\mathbf{x}_{-q}$, the predictive distribution we are interested in is given by

$$p(\mathbf{x}_{-q} | \mathbf{x}_q) = \sum_{\mathbf{z}} \int \frac{p(\mathbf{x}_{-q} | \mathbf{z}, \mathbf{a})}{p(\mathbf{x}_q | \mathbf{z}, \mathbf{a})} \left[\int p(\mathbf{z} | \boldsymbol{\pi}) p(\boldsymbol{\pi} | \alpha) d\boldsymbol{\pi} \right] p(\mathbf{a} | \beta, \gamma) d\mathbf{a}. \tag{26}$$

After every iteration of Gibbs sampling, we obtain a sample $\mathbf{z}^{(s)}$ (and corresponding counts N_k), which can be used to infer the Bernoulli parameters $\mathbf{a}_k^{(s)}$ as follows

$$\mathbf{a}_k^{(s)} = \frac{\sum_{n=1}^{N} \mathbf{x}_n^{z_k^{(s)}}}{N_k}. \tag{27}$$

Using the samples $\mathbf{z}^{(s)}$ and the corresponding Bernoulli parameters $\mathbf{a}_k^{(s)}$, the predictive distribution can be estimated by a factorized approximation

$$p(\mathbf{x}_{-q}|\mathbf{x}_q) = \prod_{i \in -q} \frac{1}{S} \sum_{s=1}^{S} \left[\frac{1}{N} \sum_{k=1}^{K_+} N_k \frac{p(x_i|\mathbf{z}_k^{(s)}, \mathbf{a}_k^{(s)})}{p(\mathbf{x}_q|\mathbf{z}_k^{(s)}, \mathbf{a}_k^{(s)})} \right]. \tag{28}$$

6 Experiments

To evaluate the performance of Bayesian Bernoulli mixtures, we performed experiments on the USPS handwritten digit dataset. The USPS dataset consist of $11,000$ binary digit images ($1,100$ digits per class) of size $16 \times 16 = 256$ pixels. We defined a prediction task in which the model has to predict the bottom half of a digit, given the top half of that digit. In our experiments, we compare the performance of Bayesian Bernoulli mixtures with those made by (1) Bernoulli mixtures trained using the EM algorithm and (2) Restricted Boltzmann Machines (RBMs) trained using contrastive divergence.

We performed experiments on each digit class separately. In each experiment, we randomly select $1,000$ images (of the same class) as training data, and used the remaining 100 images as test data. The bottom 8 rows of each test image are not presented to the models; these $8 \times 16 = 128$ pixels have to be predicted, given the top 128 pixels. As the models output a probability that a pixel is on, a natural way to measure the quality of the prediction is using receiver-operating curves (ROC curves). We use the area under the ROC curve (AUC) as evaluation criterion for the predictions. We repeat each experiment 10 times (for each digit class) to reduce the variance in our AUC estimates.

The standard (non-Bayesian) Bernoulli mixtures were trained by running the EM algorithm for 50. The predictions from the standard Bernoulli mixture were obtained by (1) computing the responsibilities of the test image under each of the mixture components based on the top half of the image and (2) computing a weighted sum of the bottom half of each of the mixture components, using the responsibilities as weights.

The Restricted Boltzmann Machines (RBMs) were trained using 30 iterations of contrastive divergence. At the start of the training, we use a single Gibbs sweep for contrastive divergence (CD-1). During the training, the number of Gibbs sweeps is slowly increased to 9 (CD-9). We also experimented with sparsity priors on the hidden unit states (which makes RBMs behave more like mixture models), but we did not find this to improve the performance of the RBMs. To predict the bottom half of each digit while observing only the top half of the digit, we employed the exact factorized predictive distribution that is described in [10].

The predictions from the Bayesian Bernoulli mixture were obtained by performing Gibbs sampling with 100 sweeps, and computing a prediction for the model at then end of the Gibbs chain (using the same approach as for the non-Bayesian Bernoulli mixtures[5]). This process is repeated 30 times, and the 30 resulting predictions are averaged to obtain the final prediction. We set the hyperparameters to $\alpha = 50$, $\beta = \frac{1}{2}$, and $\gamma = \frac{1}{2}$. For the infinite Bernoulli mixture, we used the same hyperparameters.

The mean AUCs that were recorded during our experiments are presented in Table 1. The table presents results for all digit classes, and for different values of the number of components K. The results presented in the table reveal the merits of using a fully Bayesian approach to Bernoulli mixture modeling: the Bayesian mixtures outperform their non-Bayesian counterparts in all experiments. Also, somewhat surprisingly, the Bernoulli mixtures appear to outperform the RBMs.

Some examples of the predictions produced by a standard Bernoulli mixture, an RBM, and a Bayesian Bernoulli mixture (all with $K = 50$) are shown in Figure 2. In the figure, a brighter pixel

[5]For the infinite mixture model, we ignore the contribution of the non-represented classes when making a prediction.

	K	Digit 1	Digit 2	Digit 3	Digit 4	Digit 5	Digit 6	Digit 7	Digit 8	Digit 9	Digit 0
BM	$K = 10$	0.9682	0.7725	0.8242	0.8193	0.8413	0.7988	0.8965	0.8059	0.8513	0.9069
BM	$K = 20$	0.9436	0.7728	0.7989	0.8316	0.8414	0.8007	0.8941	0.8100	0.8682	0.9179
BM	$K = 30$	0.9559	0.7748	0.7961	0.8246	0.8344	0.7929	0.9015	0.8068	0.8732	0.9113
BM	$K = 40$	0.9503	0.7473	0.7857	0.8144	0.8320	0.7936	0.8819	0.8153	0.8572	0.9073
BM	$K = 50$	0.9602	0.7636	0.8064	0.8232	0.8252	0.7980	0.8950	0.8047	0.8752	0.9093
BBM	$K = 10$	0.9727	0.7847	0.8585	0.8423	0.8622	0.7797	0.8999	0.8196	0.8739	0.9300
BBM	$K = 20$	0.9741	0.7893	0.8650	0.8632	0.8624	0.7960	0.9142	0.8293	0.8896	0.9350
BBM	$K = 30$	0.9743	0.7875	0.8653	0.8658	0.8643	0.7995	0.9167	0.8332	0.8965	0.9347
BBM	$K = 40$	0.9742	0.7938	0.8655	**0.8699**	0.8656	**0.8009**	**0.9188**	0.8356	0.8955	0.9385
BBM	$K = 50$	**0.9747**	0.7905	**0.8695**	0.8647	**0.8681**	0.7999	0.9181	**0.8379**	**0.8973**	**0.9387**
BBM	$K \to \infty$	0.9737	**0.8030**	0.8313	0.8412	0.8425	0.7937	0.8762	0.8162	0.8409	0.9087
RBM	$K = 10$	0.9623	0.7390	0.7764	0.7775	0.7810	0.7767	0.8469	0.7923	0.8263	0.8872
RBM	$K = 20$	0.9541	0.7503	0.7891	0.7829	0.7707	0.7663	0.8693	0.7885	0.8401	0.9010
RBM	$K = 30$	0.9609	0.7371	0.7801	0.7823	0.7653	0.7757	0.8515	0.7949	0.8334	0.9051
RBM	$K = 40$	0.9583	0.7411	0.7787	0.7781	0.7730	0.7524	0.8640	0.7899	0.8419	0.8763
RBM	$K = 50$	0.9559	0.7283	0.7828	0.7807	0.7785	0.7619	0.8686	0.7842	0.8242	0.8966

Table 1: Area under the ROC curve (AUC) for the fill-in task on the USPS digits.

corresponds to a higher probability of that pixel being on. The plots reveal that the Bayesian Bernoulli mixture produce *smoother* predictions, resulting in higher AUCs on the prediction task.

References

[1] B.T. Backus. The mixture of Bernoulli experts: A theory to quantify reliance on cues in dichotomous perceptual decisions. *Journal of Vision*, 9(1):1–19, 2009.

[2] M.J. Beal, Z. Ghahramani, and C.E. Rasmussen. The infinite Hidden Markov Model. In *Advances of Neural Information Processing Systems*, volume 14, 2002.

[3] M.Á. Carreira-Perpiñán and S. Renals. Practical identifiability of finite mixtures of multivariate Bernoulli distributions. *Neural Computation*, 12(1):141–152, 2000.

[4] J. Gonzalez, A. Juan, P. Dupont, E. Vidal, and F. Casacuberta. A Bernoulli mixture model for word categorisation. In *Symposium Nacional de Reconocimiento de Formas y Analises de Imagenes*, 2001.

[5] T.L. Griffiths and Z. Ghahramani. Infinite latent feature models and the indian buffet process. Technical Report GCNU TR 2005–001, Gatsby Computational Neuroscience Unit, 2005.

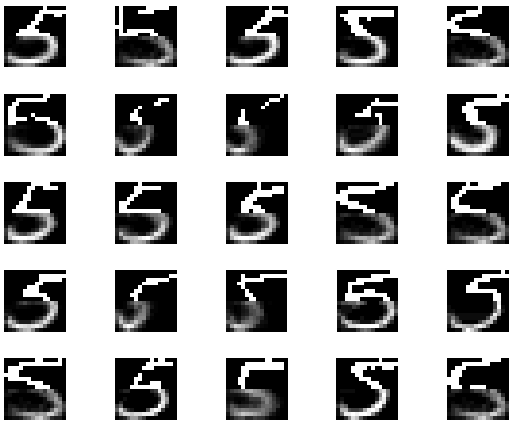
[6] M.I. Jordan and R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

[7] A. Juan and E. Vidal. On the use of Bernoulli mixture models for text classification. *Pattern Recognition*, 35(12):2705–2710, 2002.

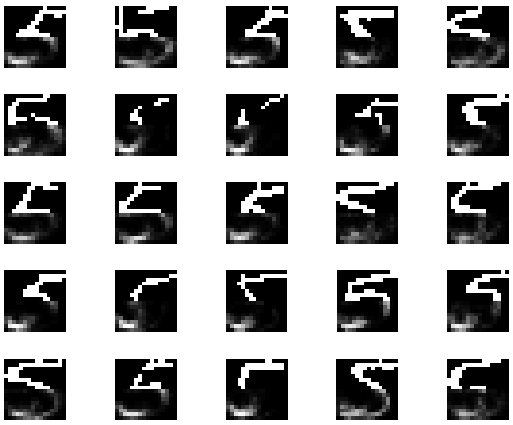
[8] R.M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9:249–265, 2000.

[9] C.E. Rasmussen. The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*, volume 12, pages 554–560, 2000.

[10] R.R. Salakhutdinov, A. Mnih, and G.E. Hinton. Restricted Boltzmann Machines for collaborative filtering. In *Proceedings of the* 24^{th} *International Conference on Machine Learning*, pages 791–798, 2007.



(a) Standard Bernoulli mixture.



(b) Restricted Boltzmann Machine.

(c) Bayesian Bernoulli mixture.

Figure 2: Examples of predictions constructed by a standard Bernoulli mixture, a Restricted Boltzmann Machine, and a Bayesian Bernoulli mixture (all with $K = 50$).