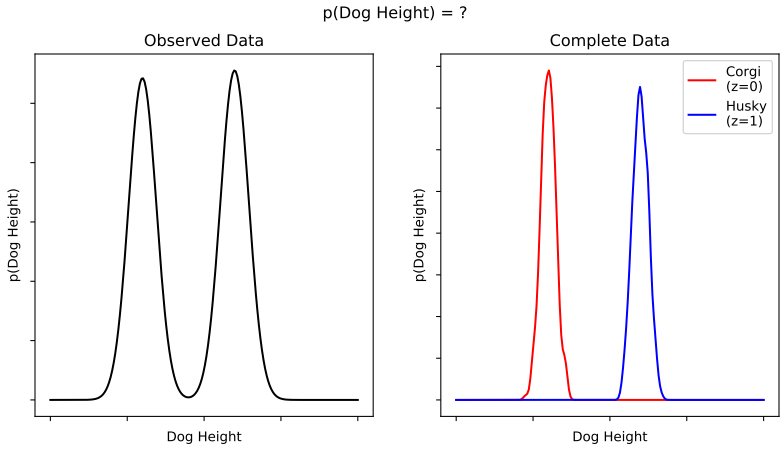
CS281 Section #4 Notes

1 Mixture Models

1.1 Latent Variables

A **latent variable** is one that is not observed at training time, but that is still part of the data generation process. Let $\{x_n\}$ be a set of observed variables and $\{z_n\}$ be a set of latent variables with joint density $p(z, x)$.



In this example, we can write the likelihood of each data point as follows:

$$p(x) = \begin{cases} N(x|\mu_0, \sigma_0), & \text{if } z = 0 \\ N(x|\mu_1, \sigma_1), & \text{if } z = 1 \end{cases} \tag{1}$$

This is really easy if we know $\{z_n\}$.
Names:

1. Observed data: $\{x_n\}$
2. Latent variables: $\{z_n\}$
3. Complete data: $\{x_n, z_n\}$

In the case of mixture models like this one, determining the values of $\{z_n\}$ can simplify the problem of computing the likelihood from estimating a very complicated distribution like the first one in the example above, to estimating a series of simple distributions like the second one in the example above.
Note: Latent variables can be used outside of the context of mixture models, but here we use this as a running example.
Note: We still don't know how to estimate $\{z_n\}$ since we never see it. The EM algorithm gives one way of doing this, but we will see several others in the next few weeks.

1.2 Mixture of Gaussians

Note: Mixtures are much more general than mixture of Gaussians. Later in the notes, we will see a mixture of Bernoullis.

Consider K Gaussian distributions with fixed means $\mu = \{\mu_1, ..., \mu_K\}$, and fixed variance $\sigma^2 = 1$.
You can draw an observation x_i from the distribution indexed by k by sampling x_i from $\mathcal{N}(\mu_k, \sigma^2)$ exactly the same as you would in a single Gaussian.
But how do you know which distribution to sample x_i from? When sampling x_i, you should first sample an indicator z_i from $\text{Cat}(\pi)$ for a fixed parameter vector π that determine the relative frequencies of the different Gaussians. z_i is a one-hot vector that corresponds to the cluster assignment that selects the proper μ_k).
We can write out the sampling procedure as follows:

$$z_i \sim \text{Cat}(\pi)$$
$$x_i | z_i, \mu, \sigma \sim \mathcal{N}(z_i^\top \mu, \sigma^2)$$

Which say that first we sample z_i, and then using the value of z_i and the value of μ, we sample x_i.
Exercise: Draw the DGM for this model.
Exercise: Use the DGM and properties of DGMs to explain why this is a harder problem than Naive Bayes.

2 The Evidence Decomposition and ELBO

2.1 The Evidence Decomposition

As we saw in class, we can re-write the log-likelihood as

$$\begin{aligned}\log p(x) &= \text{ELBO}(q) + \text{gap} \\ &= \text{ELBO}(q) + \text{KL}(q(z)||p(z|x))\end{aligned}$$

Using the fact that KL divergences are positive, we get that

$$\log p(x) \geq \text{ELBO}(q)$$

We call the ELBO the evidence lower bound because it is a lower bound on the log likelihood. You can see this from the formula above using the properties of KL divergences.
This decomposition helps us because the ELBO will be easier to work with than the full log likelihood.

2.2 The Evidence Lower Bound (ELBO)

The **evidence lower bound** (ELBO) is defined as

$$\begin{aligned}\text{ELBO}(q) &= \mathbb{E}_q[\log p(z)] + \mathbb{E}_q[\log p(x|z)] - \mathbb{E}_q[\log q(z)] \\ &= \mathbb{E}_q[\log p(x|z)] - \text{KL}(q(z)||p(z))\end{aligned}$$

We have seen the expectation maximization algorithm as an approach to maximizing the ELBO with coordinate ascent. In the next few lectures that variational inference works in general by maximizing the ELBO.
Exercise: Derive

$$\log p(x) = \text{ELBO}(q) + \text{KL}(q(z)||p(z|x))$$

Solution: We start from the right hand side of the equation and expand both KL terms using the definition of KL divergence.

$$\begin{aligned}\text{ELBO}(q) + \text{KL}(q(z)||p(z|x)) &= \mathbb{E}_q[\log p(x|z)] - \text{KL}(q(z)||p(z)) + \text{KL}(q(z)||p(z|x)) \\ &= \mathbb{E}_q[\log p(x|z) - \log \frac{q(z)}{p(z)} + \log \frac{q(z)}{p(z|x)}]\end{aligned}$$

Using the properties of logs, canceling terms then using the properties of probability distributions, we get that this is equal to the expectation of $p(x)$ under q. Since there is no q in x, we can remove the expectation and get $p(x)$.

$$\text{ELBO}(q) + \text{KL}(q(z)||p(z|x)) = \mathbb{E}_q[\log \frac{p(x|z)p(z)q(z)}{q(z)p(z|x)}] = \mathbb{E}_q[\log \frac{p(x,z)}{p(z|x)}] = \mathbb{E}_q[\log p(x)] = \log p(x)$$

3 Expectation Maximization

The expectation maximization algorithm performs coordinate ascent on the ELBO. It iteratively updates q in the expectation step, and θ in the maximization step until convergence.
Expectation:

$$\begin{aligned}q^+ &= \arg\max_q \text{ELBO}(q, \theta) \\ &= \arg\min_q \sum_{n=1}^{N} \text{KL}(q(z)||p(z|x_n)) \\ &= p(z|x_n; \theta)\end{aligned}$$

From the evidence decomposition, we can see that maximizing the ELBO is the same as minimizing $\text{KL}(q(z)||p(z|x))$. In this case, we have a closed for solution for this. The expectation step updates z, and has a closed form.
Maximization:

$$\begin{aligned}\theta^+ &= \arg\max_\theta \text{ELBO}(q^+, \theta) \\ &= \sum_{n=1}^{N} \mathbb{E}_{z\sim q_n}[log(p(x_n|z_n; \theta)]\end{aligned}$$

Here, we use the posterior probabilities of z computed in the last step to update the parameters of each mixture. The $\text{KL}(q(z)||p(z))$ disappears from the maximization since it doesn't depend on q.
The expectation step always has this form:

$$q_{nk} = p(z_n = k|x_n, \pi_k, \mu_k) \leftarrow \frac{\pi_k p(x_n|\mu_k)}{\sum_{k'} \pi_{k'} p(x_n|\mu_{k'})}$$

The maximization step for the class prior π always has this form::

$$\pi_k \leftarrow \frac{\sum_n q_{nk}}{\sum_n \sum_{k'} q_{nk'}}$$

The updates for the mixture parameters will depend on the distributions in the mixture.
Exercise: What is the μ_{kd} update for a D-dimensional mixture of Bernoullis?
Solution:

The likelihood of a mixture of Bernoullis is

$$p(x|\mu,\pi) = \sum_{k=1}^{K} \pi_k p(x|\mu_k)$$
$$p(x_i|\mu_k) = \prod_{d=1}^{D} \mu_{kd}^{x_{id}}(1-\mu_{kd})^(1-x_{id})$$

The mean and covariance of this distribution are:

$$\mathbb{E}[x] = \sum_{k=1}^{K} \pi_k \mu_k$$
$$cov[x] = \sum_{k=1}^{K} \pi_k(\Sigma_k + \mu_k \mu_k^T) - \mathbb{E}[x]\mathbb{E}[x]^T$$
$$\Sigma_k = \texttt{diag}(\mu_{kd}(1-\mu_k d))$$

Note: the draws in each dimension are no longer independent because the covariance is no longer diagonal. An example of a distribution that could be modeled effectively with a mixture of Bernoullis is 2 flips of a weighted coin, where before flipping the coin, you first choose between 2 coins with different weights. z in this example is the coin you chose, and x is the results of the 2 independent flips from that coin.

Question: We found closed form solutions for the mean and covariance. Why aren't we done?
Answer: μ depends on π and π depends on μ, so unlike in previous examples where you can estimate the mean and then use the mean to get to covariance, you can't decouple them in these equations. On the other hand, the E.M. algorithm gives us a good way of estimating μ with a fixed π and then π with a fixed μ iteratively.

We can find the update for μ_{kd} by taking the derivative of the ELBO with the current expected values of z with respect to mu_{kd} and setting it to 0. When we do this, we see that $KL(q(z)||p(z))$ does not depend on θ, so we can ignore it. We end up with the following term:

$$\text{ELBO}(q,\theta) = \mathbb{E}_q[\log p(x|z)] - \text{KL}(q(z)||p(z))$$
$$\frac{d}{d\mu_{kd}}\text{ELBO}(q,\theta) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(z_{nk})(\sum_{d=1}^{D}[x_{nd}\log(\mu_{kd}) + (1-x_{nd})\log(1-\mu_{kd}))])$$

At this point, we can also see that the terms in the sum where $k' \neq k$ and and $d' \neq d$ do not depend on μ_{kd}, so we can drop them as well. We now have

$$\frac{d}{d\mu_{kd}}\text{ELBO}(q,\theta) = \frac{d}{d\mu_{kd}}\sum_{n=1}^{N}\gamma(z_{nk})(x_{nd}\log(\mu_{kd}) + (1-x_{nd})\log(1-\mu_{kd}))$$

Using the properties of derivatives of natural logs, we have

$$\frac{d}{d\mu_{kd}}\text{ELBO}(q,\theta) = \sum_{n=1}^{N}\gamma(z_{nk})(\frac{x_{nd}}{\mu_{kd}} - \frac{1-x_{nd}}{1-\mu_{kd}})$$
$$\frac{d}{d\mu_{kd}}\text{ELBO}(q,\theta) = \sum_{n=1}^{N}\gamma(z_{nk})\frac{x_{nd}(1-\mu_{kd}) - \mu_{kd}(1-x_{nd})}{\mu_{kd}(1-\mu_{kd})}$$
$$\frac{d}{d\mu_{kd}}\text{ELBO}(q,\theta) = \sum_{n=1}^{N}\gamma(z_{nk})\frac{x_{nd}-\mu_{kd}}{\mu_{kd}(1-\mu_{kd})}$$

Setting the derivative to zero and solving for μ_{kd}, we get

$$\frac{d}{d\mu_{kd}} \text{ELBO}(q, \theta) = 0$$

$$\sum_{n=1}^{N} \gamma(z_{nk}) \frac{x_{nd} - \mu_{kd}}{\mu_{kd}(1 - \mu_{kd})} = 0$$

$$\sum_{n=1}^{N} \gamma(z_{nk})(x_{nd} - \mu_{kd}) = 0$$

$$\sum_{n=1}^{N} \gamma(z_{nk}) x_{nd} = \sum_{n=1}^{N} \gamma(z_{nk}) \mu_{kd}$$

Since μ_{kd} doesn't depend on n, we can pull it out of the sum and get:

$$\mu_{kd} = \frac{1}{\sum_{n=1}^{N} \gamma(z_{nk})} \sum_{n=1}^{N} \gamma(z_{nk}) x_{nd}$$

The denominator of the first term is the effective number of data points associated with component K. We can write it as:

$$N_k = \sum_{n=1}^{N}$$

$$\mu_{kd} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_{nd}$$

Which sets the mean equal to the mean of the data weighted by responsibility the cluster takes for each point.

4 Additional Resources

1. CS281 Lectures on Info Theory and Mixture Models (2017), Sasha Rush
2. Bayesian Mixture Models and the Gibbs Sampler (2015), David M. Blei*
3. Variational Inference: A Review for Statisticians (2017), David M. Blei, Alp Kucukelbir, Jon D. McAuliffe*