10-708: Probabilistic Graphical Models, Spring 2020

5: Parameter Estimation

Lecturer: Eric P. Xing *Scribe: Matthew Ho, Danlei Zhu, Yijie Sun, John Fang, Zhichu Lu*

In general, learning graphical models involves trying to infer the best Bayesian Network from a dataset of independent samples. Graphical model learning can be broadly separated into two categories: structural learning, wherein one might try to estimate graphical connections and the implied independences, and parameter learning, where one might seek to estimate specific conditional probabilities. This lecture covers the latter.

1 Parameter Estimation for Completely Observed GMs of Given Structure

In this section, we consider learning parameters for a Bayesian Network which has a known, fixed structure $\mathcal{G}$ and is completely observable (i.e. our data samples include observations of all variables). Stated formally, we are given a dataset of N independent, identically-distributed training cases $D = \{x_1, \ldots, x_N)$. Each training case $x_n = (x_{n,1}, \ldots, x_{n,M})$ is a vector of M values, one per node.

To address our problem of learning parameters in this context, we will describe how simple, completely-observed, structure-fixed graphical models can be generalized into the *exponential family* of distributions. This generalized reparameterization will allow us to write closed-form expressions for quantities that we are interested (e.g. conditional probabilities, means, etc.). The simple graphical models that we describe in detail are building blocks for more complex models, making the exponential family parameterization useful for learning all graphical models.

1.1 Exponential Family

1.1.1 Formulation and Examples

The *exponential family* is a parametric set of probability distributions which characterize many common examples in modern statistics, including the Bernoulli, Multinomial, Gaussian, Poisson, and Gamma distributions. For a numeric random variable X described by an exponential family distribution, the PDF can be written as:

$$\begin{aligned} p(x|\eta) &= h(x) \exp\left[\eta^T T(x) - A(\eta)\right] \\ &= \frac{1}{Z(\eta)} h(x) \exp\left[\eta^T T(x)\right] \end{aligned}$$

with natural (canonical parameter) η. The function $T(x)$ is called the *sufficient statistic* because its output is all that is required from the data to estimate η. The function $A(\eta) = \log Z(\eta)$ is the *log normalizer* and ensures the probability distribution can be integrated to unity.

We first demonstrate how a multivariate Gaussian distribution can be represented in terms of the exponential

family canonical parameters and functions. First, the classic k-dimensional Gaussian distribution:

$$\begin{aligned} p(x|\mu,\Sigma) &= \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} \exp\left\{\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\} \\ &= \frac{1}{(2\pi)^{k/2}} \exp\left\{-\frac{1}{2}\operatorname{Tr}\left(\Sigma^{-1}xx^T\right) + \mu^T\Sigma^{-1}x - \frac{1}{2}\mu^T\Sigma^{-1}\mu - \log|\Sigma|\right\} \end{aligned}$$

which is fully described by the $k+k^2$ parameters that define the first two moments of the distribution, μ and Σ. $\operatorname{Tr}(\cdot)$ is the matrix trace operation. We can represent this same distribution in the exponential family representation,

$$\begin{aligned} \eta &= \left[\Sigma^{-1}\mu; -\frac{1}{2}\operatorname{vec}\left(\Sigma^{-1}\right)\right] = \left[\eta_1, \operatorname{vec}\left(\eta_2\right)\right], \qquad \eta_1 = \Sigma^{-1}\mu \text{ and } \eta_2^2 = -\frac{1}{2}\Sigma^{-1} \\ T(x) &= \left[x; \operatorname{vec}\left(xx^T\right)\right] \\ A(\eta) &= \frac{1}{2}\mu^T\Sigma^{-1}\mu + \log|\Sigma| = -\frac{1}{2}\operatorname{Tr}\left(\eta_2\eta_1\eta_1^T\right) - \frac{1}{2}\log\left(-2\eta_2\right) \\ h(x) &= (2\pi)^{-k/2} \end{aligned}$$

where $\operatorname{vec}(\cdot)$ is an operation which flattens a matrix into a 1-D vector. The $k+k^2$ parameters in the canonical η vector fully capture the variability in the k-dimensional Gaussian previously parameterized by μ and σ.

As another example, we will show how the K-outcome multinomial distribution can be written in exponential family form. We start by stating the familiar probability distribution in a form more representative of the exponential family,

$$\begin{aligned} p(x|\pi) = \pi_1^{x_1}\pi_2^{x_1}\cdots\pi_K^{x_K} &= \exp\left\{\sum_k x_k \ln \pi_k\right\} \\ &= \exp\left\{\sum_{k=1}^{K-1} x_k \ln \pi_k + \left(1 - \sum_{k=1}^{K-1} x_K\right)\ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right)\right\} \\ &= \exp\left\{\sum_{k=1}^{K-1} x_k \ln\left(\frac{\pi_k}{1-\sum_{k=1}^{K-1}\pi_k}\right) + \ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right)\right\}. \end{aligned}$$

Note, there are only $K-1$ parameters to fit, as $\sum_{k=1}^{K}\pi_k = 1$. We follow by stating the explicit exponential family representation,

$$\begin{aligned} \eta &= \left[\ln\left(\frac{\pi_k}{\pi_K}\right); 0\right] \\ T(x) &= [x] \\ A(\eta) &= -\ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right) = \ln\left(\sum_{k=1}^{K} e^{\eta_k}\right) \\ h(x) &= 1 \end{aligned}$$

1.1.2 Moments

One particularly useful property of exponential family distributions is that we can easily compute their q-th central moments through the q-th derivatives of the log normalizer $A(\eta)$:

$$\begin{aligned}\frac{dA(\eta)}{d\eta} &= \mathbb{E}\left[T(x)\right] \equiv \mu \\ \frac{d^2A(\eta)}{d\eta^2} &= \mathrm{Var}\left[T(x)\right] \\ &\vdots\end{aligned}$$

where the expectation value of $T(x)$ is defined as the moment parameter μ. Since the log normalizer's first derivative is μ and its second derivative must be positive, then there exists some function ψ which defines a 1-to-1 relationship between canonical and moment parameters,

$$\eta \equiv \psi(\mu)$$

This property of the exponential family is particularly significant in inferring η. When performing MLE on an exponential family distribution, we can maximize the log-likelihood, ℓ, with respect to η, estimate μ directly, and then infer η using the ψ function.

$$\begin{aligned}\ell(\eta; D) &= \sum_n \log h(x_n) + \left(\eta^T \sum_n T(x_n)\right) - NA(\eta) \\ &\downarrow \\ 0 = \frac{\partial \ell}{\partial \eta} &= \sum_n T(x_n) - N\frac{\partial A(\eta)}{\partial \eta} \\ &\downarrow \\ \frac{\partial A(\eta)}{\partial \eta} = \hat{\mu}_{\mathrm{MLE}} &= \frac{1}{N}\sum_n T(x_n)\end{aligned}$$

Subsequently, we can find an estimate for the canonical parameter via $\hat{\eta}_{\mathrm{MLE}} = \psi(\hat{\mu}_{\mathrm{MLE}})$. This procedure is called *moment matching*.

1.1.3 Sufficiency

In previous sections, we have seen that most of the distributions we encounter can be expressed in the form of exponential family with appropriate $T(x)$ and η.

However, why is this interesting and practically useful? It turns out that for $p(x|\theta), T(x)$ is sufficient for θ if there is no information in X regarding θ beyond that in $T(x)$. For instance, if your boss wants you to do inference w.r.t. θ, you do not need to save all data X but $T(x)$, the sufficient statistics.

To define this property more rigorously, we need the following.

In the Bayesian view, the posterior distribution of the parameter θ is dependent of the data X. However, the posterior of the parameter is independent of data X given sufficient statistics $T(x)$, i.e. $p(\theta|T(x), x) = p(\theta|T(x))(A)$. In the Frequentist view, our data is generated from some true parameter. Yet, the distribution of our data x is dependent of the parameter θ if given the sufficient statistics $T(x)$, i.e. $p(x|T(x), \theta) = p(x|T(x))(B)$.

If we combine these two views, we obtain the Neyman factorization theorem: The statistics $T(x)$ is sufficient for the parameter θ if both (A) and B hold.

1.1.4 Examples

Here are some common distributions written in the general expoential family form.

For Gaussian, $\eta = [\Sigma^{-1}\mu; -\frac{1}{2}\vec{\Sigma^{-1}}]$, $T(x) = [x, x\vec{x^T}]$, $A(\eta) = \frac{1}{2}\mu^T\Sigma^{-1}\mu + \frac{1}{2}\log|\Sigma|$, $h(x) = (2\pi)^{-k/2}$,

$$\mu_{MLE} = \frac{1}{N}\Sigma_n T_1(x_n) = \frac{1}{N}\Sigma_n x_n$$

.

For Multinomial, $\eta = [\ln\frac{\pi_k}{\pi_K}; 0]$, $T(x) = [x]$, $A(\eta) = -\ln(1 - \Sigma_{k=1}^{K-1}\pi_k) = \ln(\Sigma_{k=1}^{K}e^{\eta_k})$, $h(x) = 1$,

$$\mu_{MLE} = \frac{1}{N}\Sigma_n x_n$$

For Possion, $\eta = \log\lambda$, $T(x) = x$, $A(\eta) = \lambda = e^{\eta}$, $h(x) = \frac{1}{x!}$,

$$\mu_{MLE} = \frac{1}{N}\Sigma_n x_n$$

.

1.2 Generalized Linear Models (GLIMs)

With the definition of Exponential Family Distribution, we can begin analyzing Generalized Linear Models(GLIMs). Suggested by its name, the definition of GLIM is very general.

1. The observed input x is assumed to enter into the model via a linear combination of its elements ξ, where $\xi = \theta^T x$.
2. The conditional mean μ is represented as a function $f(\xi)$ of ξ, where f is known as the response function.
3. The observed output y is assumed to be characterized by an exponential family distribution p with conditional mean μ.

Then the model $E_p(y) = \mu = f(\theta^T x)$ is called a GLIM. Some basic examples of GLIM include:

1. Linear Regression

   Assume the target variable y and the inputs are related by the equation: $y_i = \theta^T X_i + \epsilon_i$,where $\epsilon \sim N(0, \sigma)$. Then we have $p(y_i|x_i, \theta) = \frac{1}{\sqrt{2\pi}\sigma}\exp -\frac{(y_i - \theta^T x_i)^2}{2\sigma^2}$. To estimate θ, we can apply LMS algorithm (a gradient ascent/descent).

2. Logistic Regression(sigmoid classifier, perceptron,etc.)

   In logistic regression, the condition distribution is $p(y|x) = \mu(x)^y(1 - \mu(x))^{1-y}$, where $\mu(x) = \frac{1}{1+e^{-\theta^T x}}$ and $y \in \{0, 1\}$. To estimate parameter θ, we can either directly apply brute-force gradient method or generic laws by observing that $p(y|x)$ is a GLIM.

More advanced examples of GLIM include:

1. Markov Random Fields, where $p(X) = \frac{1}{Z}\exp(\Sigma_{i,j\in N_i}\theta_{ij}X_iX_j + \Sigma_i\theta_{i0}X_i)$

2. Restricted Boltzmann Machines, where

$$p(x, h|\theta) = \exp\{\Sigma_i \theta_i \phi_i(x_i) + \Sigma_j \theta_j \phi_j(h_j) + \Sigma_{i,j} \theta_{i,j} \phi_{i,j}(x_i, h_j) - A(\theta)\}$$

3. Conditional Random Fields, where

$$p_\theta(y|x) = \frac{1}{Z}(\theta, x) \exp\{\Sigma_c \theta_c f_c(x_c y_c)\}$$

and all X_i are assumed as features that are inter-dependent.

A more formal view of GLIMs is the following:

parameter θ and data $x \to \xi \overset{f}{\to} \mu \overset{\Psi}{\to} \eta \overset{\text{EXP}}{\to} y$, where f is some response function, Ψ is some reversible transformer corresponding to the T operator before, and EXP is some exponential family distribution we use.

Notice that the choice of exp family distribution is constrained by the nature of the data y. For example, if y is a continuous vector, then multivariate Gaussian is a reasonable choice. However, if y is a class label, using Bernouli or multinomial is more favorable.

We also have some mild constraints for the choice of response function, such as positivity. There also exists some canonical response functions for different models.

- Gaussian, $\mu = \eta$
- Bernoulli, $\mu = \frac{1}{1+e^{-\eta}}$
- multinomial, $\mu_i = \frac{\eta_i}{\Sigma_j e^{\eta_j}}$
- Poisson, $\mu = e^\eta$
- gamma, $\mu = -\eta^{-1}$.

1.3 Learning GLIMs

1.3.1 MLE for GLIMs with natural response

For example for log-likelihood,

$$l = \sum_n \log h(y_n) + \sum_n (\theta^T x_n y_n - A(\eta_n))$$

note that here $\theta^T x_n$ acts as the natural parameters for the exponential family distribution. Take derivative of the log-likelihood,

$$\begin{aligned}
\frac{dl}{d\theta} &= \sum_n (x_n y_n - \frac{dA(\eta_n)}{d\eta_n} \frac{d\eta_n}{d\theta}) \\
&= \sum_n (y_n - \mu_n) x_n \\
&= X^T (y - \mu)
\end{aligned}$$

Note that this is a fixed point function since μ is a function of θ. This can be used as a generic learning rule in online learning for canonical GLIMs. The stochastic gradient ascent is given by

$$\theta^{t+1} = \theta^t + \rho(y_n - \mu_n^t)x_n$$
$$\text{where } \mu_n^t = (\theta^t)^T x_n, \rho \text{ is a step size}$$

Alternative to stochastic gradient descent to speed up is batch learning algorithm:

$$\begin{aligned} H = \frac{dl^2}{d\theta d\theta^T} &= \frac{d}{d\theta^T}\sum_n (y_n - \mu_n)x_n \\ &= \sum_n x_n \frac{d\mu_n}{d\theta^T} \\ &= -\sum_n x_n \frac{d\mu_n}{d\eta_n}\frac{d\eta_n}{d\theta^T} \\ &= -\sum_n x_n \frac{d\mu_n}{d\eta_n} x_n^T \\ &= -X^T W X \end{aligned}$$

where the second but last equality is due to $\eta_n = \theta^T x_n$. $X = [x_n^T]$ is the design matrix and $W = diag(\frac{d\mu_i}{d\eta_i})_{i=1}^N$ can be computed via the second derivative of $A(\eta_n)$.

After obtaining Hessian $H = -X^T W X$ together with jacobian, we can apply Iteratively Reweighted Least Squares (IRLS).

Recall Newton-Raphson methods

$$\begin{aligned} \theta^{t+1} &= \theta^t + H^{-1}\nabla\theta l \\ &= H^{-1}(H\theta^t + \nabla\theta l) \\ &= (X^T W^t X)^{-1} X^T W^t z^t \end{aligned}$$

where $z^t = X\theta^t + (W^t)^{-1}(y - \mu^t)$ is the adjusted response. This can be understood as solving the the iteratively reweighted least squares problem

$$\theta^t = \arg\min_\theta (x - X\theta)^T W (z - X\theta)$$

1.3.2 Examples: Logistic and Linear regression

For logistic regression, conditional distribution is given by a Bernoulli

$$p(y|x) = \mu(x)^y (1 - \mu(x))^{1-y}$$

where $\mu(x) = \frac{1}{1+e^{-\eta(x)}}$. We know $p(y|x)$ is an exponential family function with mean $E(y|x) = \mu = \frac{1}{1+e^{-\eta(x)}}$ and canonical response function $\eta = \theta^T x$. Hence from previous section we know IRLS updates with

$$\frac{d\mu}{d\eta} = \mu(1 - \mu)$$

$$W = diag(\mu_i(1 - \mu_i)_{i=1}^N)$$

For linear regression, condition distribution is a Gaussian

$$\begin{aligned} p(y|x,\theta,\Sigma) &= \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} \exp\{-\frac{1}{2}(y-\mu(x))^T\Sigma^{-1}(y-\mu(x))\} \\ &= h(x)\exp\{-\frac{1}{2}\Sigma^{-1}(\eta^T(x)y - A(\eta))\} \end{aligned}$$

where $\mu(x) = \theta^T x = \eta(x)$. Thus $p(y|x)$ is an exponential family function with $E(y|x) = \mu = \theta^T x$ and canonical response function $\eta_1 = \theta^T x$. Since $\frac{d\mu}{d\eta} = 1$ thus in IRLS $W = I$ hence

$$\theta^{t+1} = \theta^t + (X^T X)^{-1} X^T (y - \mu^t)$$

which is reduced to steepest descent. If further take $t \to \infty$ we get the normal equation

$$\theta = (X^T X)^{-1} X^T Y$$

Remember that simple GMs (with one or two nodes) are the building blocks of complex GMs.

1.4 MLE for General BNs

1.4.1 Example

Assume the parameters for each CPD are globally independent and all nodes are fully observed then the log-likelihood function decomposes into a sum of local terms, one per node.

$$\begin{aligned} l(\theta, D) &= \log p(D|\theta) \\ &= \log \Pi_n (\Pi_i p(x_{n,i}|\mathbf{x}_{n,\pi_i}, \theta_i)) \\ &= \sum_i (\sum_n \log p(x_{n,i}|\mathbf{x}_{n,\pi_i}, \theta_i)) \end{aligned}$$

which allows us to utilize what we have learned from the small GM to instantiate each term in the above equation and get the result for general GM.

1.4.2 Decomposable Likelihood of a BN

Distribution defined by the DAG GM

$$p(x|\theta) = p(x_1|\theta_1)p(x_2|x_1,\theta_2)p(x_3|x_1,\theta_3)p(x_4|x_2,x_3,\theta_4)$$

leads us to learn four separate small BNs, each of which consists of a node and its parents.

Suppose now each CPD is represented as a table (multinomial) where

$$\theta_{ijk} := p(X_i = j | X_{\pi_i} = k)$$

and the sufficient statistics are counts of family configurations

$$n_{ijk} = \sum_n x_{n,i}^j x_{n,\pi_i}^k$$

and the log likelihood is

$$l(\theta, D) = \log \Pi_{ijk} \theta_{ijk}^{n_{ijk}} = \sum_{i,j,k} n_{ijk} \log \theta_{ijk}.$$

Using a Lagrange multiplier to enforce $\sum_j \theta_{ijk} = 1$ we get

$$\theta_{ijk}^{ML} = \frac{n_{ijk}}{\sum_{j'} n_{ij'k}}$$

In summary, learning BN with more nodes rely on local operations which build off that.

2 Parameter Estimation for Partially Observed GMs: EM Algorithm

2.1 Unobserved Variables

In previous sections we have seen parameter estimation for fully observed graphical models. However, often in practice some random variables in a graphical model may be unobserved, and for various reasons. Some random variables are imaginary quantities designed to capture the abstract data generation process and thus are not physically measurable (e.g. the latent variables in speech recognition models and mixture models); some others may be unobserved because of faulty sensors. As we will see, the fact that there are unobserved random variables makes parameter estimation trickier. Nonetheless, partially observed graphical models models remain useful in practice, and we will see how to use Expectation-Maximization to estimate their parameters.

2.1.1 Why is Learning Harder for Partially Observed GMs?

Let's consider the case of Gaussian Mixture Models, where the data is sampled from a mixture of Gaussian distributions by first sampling Z, the class indicator vector, and then sampling X from a Gaussian distribution with a class specific mean and co-variance matrix.

We can estimate the parameters of the graphical model using MLE. In a fully observed setting where Z is observed, we maximize the log likelihood function

$$\ell_c(\theta; x, z) = \log p(x, z|\theta) = \log p(x, z|\theta_z) + \log p(x|z, \theta_x)$$

which factors nicely into two terms with decoupled parameters θ_z and θ_x which can be optimized individually.

If we do not observe Z, then the likelihood function is the following,

$$\ell_c(\theta; x) = \log p(x|\theta) = \log \sum_z p(x, z|\theta)$$

where parameters θ_x and θ_z become coupled via marginalization. This objective is much harder to optimize than the objective in the fully observed setting. In the following sections, we will see the EM algorithm and how it can be seen as optimizing a surrogate of this objective.

2.2 Expectation-Maximization

Again let's consider Gaussian Mixture Models. Our goal is to estimate parameters of the GMM given partially observed data. We have just seen that the fully observed objective for GMM is much easier to optimize than the partially observed objective. What we are missing in the partially observed setting is the value for the latent variable Z. Hence, the goal of the E-Step step is to compute the expectation of Z so that the M-step can perform parameter estimation similar to how it is done in the fully observed setting, but using the expected value of Z (and in general its sufficient statistics) rather than the value of Z.

This can be formulated as follows, in the **E-step**, we compute the expected value of the the hidden variables (i.e. z_n^k) given the current estimate of the parameters,

$$\tau_n^{k(t)} = \langle z_n^k \rangle_{q(t)} = p(z_n^k = 1 | x, \mu^{(t)}, \Sigma^{(t)}) = \frac{\pi_k^{(t)} N(x_n | \mu^{(t)}, \Sigma^{(t)})}{\sum_i \pi_i^{(t)} N(x_n | \mu_i^{(t)}, \Sigma_i^{(t)})}$$

In the **M-step**, we perform parameter estimation using the current expected values for the hidden variables. We are optimizing the *expected complete log likelihood* which is discussed with more detail in Section 2.3.

$$\begin{aligned}
\pi_k^* &= \frac{\sum_n \langle z_n^k \rangle_{q(t)}}{N} = \frac{\sum_n \tau_n^{k(t)}}{N} = \frac{\langle n_k \rangle}{N} \\
\mu_k^{(t+1)} &= \frac{\sum_n \tau_n^{k(t)} x_n}{\sum_n \tau_n^{k(t)}} \\
\Sigma_k^{(t+1)} &= \frac{\sum_n \tau_n^{k(t)} (x_n - \mu_k^{(t+1)})(x_n - \mu_k^{(t+1)})^T}{\sum_n \tau_n^{k(t)}}
\end{aligned}$$

2.3 Complete and Incomplete Log Likelihoods

Using MLE, we want to learn the model parameters that maximize the likelihood of the data. In other words, we want to maximize the *complete log likelihood*, defined as

$$\ell_c(\theta; x, z) := \log p(x, z | \theta)$$

Maximizing this would be easy if all the variables were observed. However, when z is not observed, we instead have an *incomplete log likelihood*,

$$\ell_c(\theta; x) = \log p(x | \theta) = \log \sum_z p(x, z | \theta)$$

This objective doesn't decouple, so we cannot maximize it directly. Instead, we try to maximize a surrogate that lower bounds the objective we want. For any distribution $q(z)$ we define the *expected complete log likelihood* as

$$\langle \ell_c(\theta; x, z) \rangle_q := \sum_z q(z | x, \theta) \log p(x, z | \theta)$$

We can see that this is a lower bound using Jensen's inequality. The proof is as follows

$$\begin{aligned}
\ell(\theta; x) &= \log p(x|\theta) \\
&= \log \sum_z p(x, z|\theta) \\
&= \log \sum_z q(z|x) \frac{p(x, z|\theta)}{q(z|x} \\
&\geq \sum_z q(z|x) \log \frac{p(x, z|\theta)}{q(z|x)} \\
&= \langle \ell_c(\theta; x, z) \rangle_q + H_q
\end{aligned}$$

2.4 EM as Maximizing Free Free Energy

We define the *free energy* as follows:

$$F(q, \theta) := \sum_z q(z|x) \log \frac{p(x, z|\theta)}{q(z|x)} = \langle \ell_c(\theta; x, z) \rangle_q + H_q$$

Note that this is the second-to-last expression in the previous proof. Then we can view the EM algorithm as coordinate ascent on F.

2.4.1 E-Step

In the E-step, we maximize over q, and we can show that the solution is

$$q^{t+1} = \text{argmax}_q F(q, \theta^t) = p(z|x, \theta^t)$$

We can prove this by showing that this choice of q^{t+1} achieves the upper bound on F that we derived in the previous section

$$\begin{aligned}
F(p(z|x, \theta^t), \theta^t) &= \sum_z p(z|x, \theta^t) \log \frac{p(x, z|\theta^t}{p(z|x, \theta^t} \\
&= \sum_z p(z|x, \theta^t) \log p(x|\theta^t) \\
&= \log p(x|\theta^t) \\
&= \ell(\theta^t; x)
\end{aligned}$$

Given this result, we assume WLOG that $p(x, z|\theta)$ is a generalized exponential family distribution. Then the expected complete log likelihood is

$$\begin{aligned}
\langle \ell_c(\theta^t; x, z) \rangle_{q^{t+1}} &= \sum_z q(z|x, \theta^t) \log p(x, z|\theta^t) - A(\theta) \\
&= \sum_i \theta_i^t \langle f_i(x, z) \rangle_{q(z|x, \theta^t)} - A(\theta) \\
&\overset{p \sim \text{GLIM}}{=} \sum_i \theta_i^t \langle \eta_i(z) \rangle_{q(z|x, \theta^t)} \xi_i(x) - A(\theta)
\end{aligned}$$

2.4.2 M-Step

In the M-step, we now maximize over θ. We note that in the definition of free energy, the H_q term does not depend on θ, so we are just optimizing the first term.

$$\theta^{t+1} = \text{argmax}_\theta \langle \ell_c(\theta; x, z) \rangle_{q^{t+1}} = \text{argmax}_\theta \sum_z q(z|x) \log p(x, z|\theta)$$

Under optimal q^{t+1}, this is equivalent to solving a standard MLE of the fully observed model $p(x, z|\theta)$ with the sufficient statistics involving z replaced by their expectations w.r.t $p(z|x, \theta)$.

2.5 EM Algorithm for K-Means

In K-means, we assume there are K clusters and we want to find the parameters of the model (i.e. means of the K clusters) and the hidden variable (i.e. cluster assignments of data points). We can estimate the means iteratively by alternating between 1) computing cluster assignments at time t using the means at time t, and 2) using cluster assignments at time t to recompute the mean for the next iteration. This can be formalized as follows, in E-step

$$z_n^{(t)} = \text{argmin}_k (x_n - \mu_k^{(t)})^T \Sigma_k^{-1(t)} (x_n - \mu_k^{(t)})$$

and in M-step

$$\mu_k^{(t+1)} = \frac{\sum_n \delta(z_n^{(t)}, k) x_n}{\sum_n \delta(z_n^{(t)}, k)}$$

.

2.6 EM Algorithm for Gaussian Mixture Models (GMMs)

In GMMs, we assume that we have some data that is sampled from a mixture of k Gaussian distributions. The data $\{x_n\}$ are observed, but we do not know the parameters for the Gaussian distributions $\{\mu_k, \Sigma_k\}$. Let z_n be a latent class indicator vector and suppose we have a prior $\pi_k = p(z_n^k = 1)$ on the class labels. Then we can write its likelihood as

$$p(z_n) = \prod_k (\pi_k)^{z_n^k}$$

If we knew the class label for a data point, then the likelihood is

$$\begin{aligned} p(x_n|z_n^k = 1, \mu, \Sigma) &= N(x_n; \mu_k \Sigma_k) \\ &= \frac{1}{(2\pi)^{m/2} |\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2} (x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) \right) \end{aligned}$$

Thus, we can combine the previous two equations and utilize the fact that z_n is a binary indicator vector to write the likelihood of x_n

$$\begin{aligned} p(x_n|\mu, \Sigma) &= \sum_k p(x_n, z_n^k = 1|\mu, \Sigma) \\ &= \sum_k p(z_n^k = 1|\pi) p(x_n|z_n^k = 1, \mu, \Sigma) \\ &= \sum_k \pi_k \cdot N(x_n; \mu_k \Sigma_k)) \end{aligned}$$

The complete log-likelihood is thus

$$\begin{aligned}\langle \ell_c(\theta; x, z)\rangle &= \sum_n \langle \log p(z_n|\pi)\rangle_{p(z|x)} + \sum_n \langle \log p(x_n|z_n, \mu, \Sigma)\rangle_{p(z|x)} \\ &= \sum_n \sum_k \langle z_n^k\rangle \log \pi_k - \frac{1}{2}\sum_n \sum_k \langle z_n^k\rangle \left((x_n - \mu_k)^T \Sigma_k^{-1}(x_n - \mu_k) + \log|\Sigma_k| + C\right)\end{aligned}$$