# Expectation Propagation for Bayesian Inference

## 1 Introduction

Expectation Propagation(EP) is one of approaches for approximate inference which first formulated the way we see today at [1] though the idea has roots in many previous works in various areas. It can be considered as a variant of message-passing where each of the individual messages are approximated while being transferred. To introduce EP, it is easier to first start with a couple of approximations (projections) for any arbitrary distribution. Here we start with Assumed Density Filtering(ADF).

## 2 Assumed Density Filtering

ADF is introduced independently in several areas at different times under different names like "Moment Matching", "Weak Marginalization", etc [2, 3, 4]. The idea used is so much similar to the update equations in Kalman Filtering.
Assume that using the $\mathbf{x}$ as observations we want to make inference about the latent variables $\mathbf{y}$. Now the goal is to find a an exact posterior $p(\mathbf{y}|\mathbf{x})$ and only keep the approximation to it $q(\mathbf{y})$, using a tractable form, say exponential form, and possibly use it for future calculation. This further approximation of the posterior is can be seen as *projection* of one distribution, over another family of distributions. There are many ways to project one distribution over another, but for here, let's say we want to project any arbitrary distribution over exponential family using the KL-*divergence*[1] , or,

$$\hat{q} = \text{proj}\left(p(\mathbf{y}|\mathbf{x}) \to q(\mathbf{y})\right) \triangleq \arg\min_{q} \text{KL}\left(p(\mathbf{y}|\mathbf{x})||q(\mathbf{y})\right)$$

[1] If you don't know about the divergence measures, see this: `http://web.engr.illinois.edu/~khashab2/learn/info.pdf`

To do so, we choose the following exponential parametric form,

$$q_\theta(\mathbf{y}) = \frac{1}{Z(\theta)} \exp\left(\theta^\top \Phi(\mathbf{y})\right), \quad Z(\theta) = \int \exp\left(\theta^\top \Phi(\mathbf{y})\right) d\mathbf{y}$$

$\Phi(\mathbf{y})$ is natural statistic of $\mathbf{y}$. The most famous case is a Gaussian distribution with mean and covariance matrix. To reduce the difference between the posterior approximation, $q(.)$ and the real posterior value, again we use the KL-*divergence*.

$$f(\theta) = \mathrm{KL}(p||q) = \mathbb{E}_p \log \frac{p}{q_\theta} = \mathbb{E}_p \log(p) + \mathbb{E}_p \log\left(Z(\theta)\right) - \mathbb{E}_p \left[\theta^\top \Phi(\mathbf{y})\right]$$

$$\nabla_\theta f(\theta) = 0 \Rightarrow \nabla_\theta f(\theta) = \nabla_\theta \log Z(\theta) - \mathbb{E}_p \left[\Phi(\mathbf{y})\right] = 0 \tag{1}$$

We also have:

$$\nabla_\theta \log Z(\theta) = \frac{\nabla_\theta Z(\theta)}{Z(\theta)} = \frac{\nabla_\theta \int \exp\left(\theta^\top \Phi(\mathbf{y})\right)}{Z(\theta)} = \frac{\int \nabla_\theta \exp\left(\theta^\top \Phi(\mathbf{y})\right)}{Z(\theta)} = \mathbb{E}_q \left[\Phi(\mathbf{y})\right] \tag{2}$$

Combining the results from equations (2, 1) we have,

$$\nabla_\theta f(\theta) = 0 \Rightarrow \mathbb{E}_q \left[\Phi(\mathbf{y})\right] = \mathbb{E}_p \left[\Phi(\mathbf{y})\right] \tag{3}$$

Calculating the Hessian for $f(\theta)$ we can show that, the above solution is a *minimum* to $f(\theta)$.

$$\left[\nabla\nabla_\theta f(\theta)\right]_{ij} = \frac{\partial^2 \log Z(\theta)}{\partial\theta_j \partial\theta_i} = \frac{\partial}{\partial\theta_j} \frac{\int \Phi_i(\mathbf{y}) \exp\left(\theta^\top \Phi(\mathbf{y})\right) d\mathbf{y}}{Z(\theta)} \tag{4a}$$

$$= \mathbb{E}_q \left[\Phi_i(\mathbf{y}), \Phi_j(\mathbf{y})\right] - \mathbb{E}_q \left[\Phi_i(\mathbf{y})\right] . \mathbb{E}_q \left[\Phi_j(\mathbf{y})\right] \geq 0 \tag{4b}$$

Using the above equation, we can conclude that, to get the best estimation for an arbitrary distribution (With KL-*divergence* as the difference between two distributions.) using an exponential distribution, it is enough to match their moments. Specifically if we assume having a Gaussian distribution for the approximating distributions, $q_\theta(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ Results from equations ( 4, 3 ) are:

$$\begin{cases} \boldsymbol{\mu}^* = \mathbb{E}_p \left[\mathbf{y}\right] \\ \boldsymbol{\Sigma}^* = \mathbb{E}_p \left[\mathbf{y}\mathbf{y}^\top\right] - \mathbb{E}_p \left[\mathbf{y}\right] \mathbb{E}_p \left[\mathbf{y}\right]^\top \end{cases}$$

### 2.1 ADF for a factorized distribution

Now assume we can factorize the given distribution, $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{x})p(\mathbf{x}, \mathbf{y}) = \prod_i t_i(\mathbf{y})$. The goal is to approximate each of the factors using the approximating distribution. It is relatively better to have less components, to keep further calculations minimum, while it is better to have a simple form for each of the factors, so to have easier approximation procedure for each of them. Thus there is a trade-off between the number of factors and simplicity of each factor. The goal is to get the least error by using the minimum number of approximating factors. Assume that the approximating distribution, $q_\theta(\mathbf{y})$ has a known exponential form, e.g. Gaussian. At each step, we will consider the factor $t_i$ from our target distribution, and changed our approximation based on the added factor $t_i$ to get a better approximation. The distribution $\hat{p}(\mathbf{y})$ is an auxiliary distribution which we use during the algorithm.

$$\hat{p}(\mathbf{y}) = \frac{1}{\tilde{Z}(\theta)} t_i(\mathbf{y}) q_\theta^{old}(\mathbf{y}), \quad \tilde{Z}(\theta) = \int t_i(\mathbf{y}) q_\theta^{old}(\mathbf{y}) d\mathbf{y}. \tag{5}$$

$q_\theta^{old}(\mathbf{y})$ is the approximate-posterior distribution at the previous step. Based on the previous definition it is clear that, $\hat{p}(\mathbf{y})$ is the approximate posterior up to factor $t_i(\mathbf{y})$. Similar to what explained, by minimizing KL $(\hat{p}(\mathbf{y})||q_\theta^{new}(\mathbf{y}))$ and assuming that $q_\theta^{new}(\mathbf{y})$ has an exponential form, we use the moments of the distributions to get new approximation, $q_\theta^{new}(\mathbf{y})$. To simplify the notation, we drop *old* from $q_\theta^{old}(\mathbf{y})$. Using the equation (5),

$$\nabla_\theta q_\theta(\mathbf{y}) = \nabla_\theta \frac{1}{Z(\theta)} \exp\left(\theta^\top \Phi(\mathbf{y})\right) = \nabla_\theta \left[\frac{1}{Z(\theta)}\right] \exp\left(\theta^\top \Phi(\mathbf{y})\right) + \frac{1}{Z(\theta)}.\nabla_\theta \exp\left(\theta^\top \Phi(\mathbf{y})\right)$$

$$\Rightarrow \nabla_\theta q_\theta(\mathbf{y}) = -\frac{\nabla_\theta Z(\theta)}{Z(\theta)} q_\theta(\mathbf{y}) + \Phi(\mathbf{y}) q_\theta(\mathbf{y}) = -\mathbb{E}_q\left[\Phi(\mathbf{y})\right] + \Phi(\mathbf{y}) q_\theta(\mathbf{y}).$$

Multiplying in $\frac{1}{\tilde{Z}(\theta)}.t_i(\mathbf{y})$ and integrating with respect to $\mathbf{y}$ we have,

$$\frac{t_i(\mathbf{y})}{\tilde{Z}(\theta)} \nabla_\theta q_\theta(\mathbf{y}) = -\mathbb{E}_q\left[\Phi(\mathbf{y})\right].\frac{1}{\tilde{Z}(\theta)} t_i(\mathbf{y}) q_\theta(\mathbf{y}) + \Phi(\mathbf{y}).\frac{1}{\tilde{Z}(\theta)} t_i(\mathbf{y}) q_\theta(\mathbf{y}).$$

$$\Rightarrow \frac{1}{\tilde{Z}(\theta)} \nabla_\theta \tilde{Z}(\theta) = -\mathbb{E}_q\left[\Phi(\mathbf{y})\right] + \mathbb{E}_{\hat{p}}\left[\Phi(\mathbf{y})\right].$$

$$\Rightarrow \mathbb{E}_{\hat{p}}\left[\Phi(\mathbf{y})\right] = \nabla_\theta \log\left(\tilde{Z}(\theta)\right) + \mathbb{E}_q\left[\Phi(\mathbf{y})\right].$$

For calculating $q_\theta^{new}(\mathbf{y})$ from $\mathrm{KL}\left(q_\theta^{new}(\mathbf{y})||\hat{p}(\mathbf{y})\right)$ we can use equations ( 4, 3 ) to match the moments for $q_\theta^{new}(\mathbf{y})$ and $\hat{p}(\mathbf{y})$:

$$\mathbb{E}_{q^{new}}\left[\Phi(\mathbf{y})\right] = \mathbb{E}_{\hat{p}}\left[\Phi(\mathbf{y})\right].$$

$$\Rightarrow \mathbb{E}_{q^{new}}\left[\Phi(\mathbf{y})\right] = \nabla_\theta \log\left(\tilde{Z}(\theta)\right) + \mathbb{E}_q\left[\Phi(\mathbf{y})\right].$$

If we assume an exponential distribution we can find $\nabla_\theta \log\left(\tilde{Z}(\theta)\right)$ and $\mathbb{E}_q\left[\Phi(\mathbf{y})\right]$ in closed form. Assume a Gaussian distribution, $q_\theta(\mathbf{y}) = q(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \mathcal{N}\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$ and $\tilde{Z}(\theta) = \tilde{Z}(\boldsymbol{\mu},\boldsymbol{\Sigma}) = \int t(\mathbf{y}).q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)d\mathbf{y}$. We now have:

$$\nabla_\mu q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right) = \boldsymbol{\Sigma}^{-1}\left(\mathbf{y}-\boldsymbol{\mu}\right)q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$$

$$\Rightarrow \mathbf{y}.q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right) = \boldsymbol{\mu}.q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right) + \boldsymbol{\Sigma}.\nabla_{\boldsymbol{\mu}} q\left(\mathbf{y};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$$

Multiplying both sides in $\frac{1}{\tilde{Z}}t_i(\mathbf{y})$ and integrating with respect to $\mathbf{y}$,

$$\boldsymbol{\mu}^* = \boldsymbol{\mu} + \frac{1}{\tilde{Z}}\boldsymbol{\Sigma}.\nabla_{\mu}\int t(\mathbf{y})q(\mathbf{y})d\mathbf{y} \tag{6a}$$

$$= \boldsymbol{\mu} + \frac{1}{\tilde{Z}}\boldsymbol{\Sigma}.\nabla_{\boldsymbol{\mu}}\tilde{Z}\left(\boldsymbol{\mu},\boldsymbol{\Sigma}\right) \tag{6b}$$

$$= \boldsymbol{\mu} + \boldsymbol{\Sigma}.\nabla_{\boldsymbol{\mu}}\log\left(\tilde{Z}\left(\boldsymbol{\mu},\boldsymbol{\Sigma}\right)\right) \tag{6c}$$

$$= \boldsymbol{\mu} + \boldsymbol{\Sigma}.\mathbf{g}, \qquad \mathbf{g} \triangleq \nabla_{\boldsymbol{\mu}}\log\left(\tilde{Z}\left(\boldsymbol{\mu},\boldsymbol{\Sigma}\right)\right) \tag{6d}$$

Similarly we for covariance (second moment) we have,

$$\Rightarrow \mathbf{y}\mathbf{y}^\top q(\mathbf{y}) = 2\boldsymbol{\Sigma}.\left[\nabla_{\boldsymbol{\Sigma}} q(\mathbf{y})\right]$$

$$\Rightarrow \langle\mathbf{y}\mathbf{y}^\top\rangle = \boldsymbol{\Sigma}+2\boldsymbol{\Sigma}\mathbf{G}\boldsymbol{\Sigma}\langle\mathbf{y}\rangle_{\hat{p}(\mathbf{y})}\boldsymbol{\mu}^\top+\boldsymbol{\mu}\langle\mathbf{y}\rangle_{\hat{p}(\mathbf{y})}^\top-\boldsymbol{\mu}\boldsymbol{\mu}^\top, \qquad \mathbf{G} = \nabla_{\boldsymbol{\Sigma}}\log\left(\tilde{Z}(\boldsymbol{\mu},\boldsymbol{\Sigma})\right)$$

$$\Rightarrow \boldsymbol{\Sigma}^* = \langle\mathbf{y}\mathbf{y}^\top\rangle_{\hat{p}(\mathbf{y})} - \langle\mathbf{y}\rangle_{\hat{p}(\mathbf{y})}\langle\mathbf{y}\rangle_{\hat{p}(\mathbf{y})}^\top = \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\left(\mathbf{g}\mathbf{g}^\top - 2\mathbf{G}\right)\boldsymbol{\Sigma}.$$

Based the above formula, it should be clear that, the order in which we use the factors $\{t_i\}_i$ to create the approximation, will change the final answer, since to approximate the posterior we go through the factors linearly only once. While Expectation Propagation aims to solve this problem by finding more consistent approximation by passing the factors for several time, so that the approximation is not dependent on the order which we see the factors. One sample difference in approximation of posterior of a toy example is shown in Figure (1).

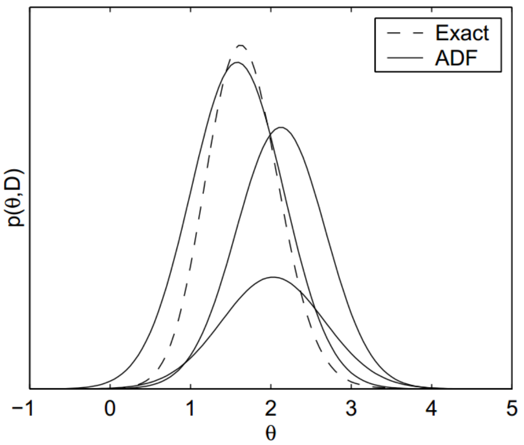

Figure 1: The effect of changing the ordering in factors of the target distribution, while approximating via ADF; [5].

# 3 Expectation Propagation (EP)

The EP approximation first introduced in [5]. The method is so much similar to Automatic Density Filtering (ADF). In fact, EP is using update rules of in an intelligent iterative way until convergence. As mentioned the main problem in ADF is that, different order in approximating the factors will result in different approximations. While EP loops over the factors until convergence. Another difference is that, instead of applying the KL-*divergence* to $q_\theta(\mathbf{y})$ as in ADF, in EP we apply it to each factors $t_i(\mathbf{y})$ and then we update the approximation $p(\mathbf{y}|\mathbf{x})$. This way, by sweep over the factors for several time, the ordering of selecting the factors $\{t_i\}_i$ doesn't make any difference. The algorithm is shown in Algorithm (1).

We want to approximate $p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\prod_i^n t_i(\mathbf{y})$ using $q_\theta(\mathbf{y}) = \dfrac{\prod_i^n \tilde{t}_i}{\int \prod_i^n \tilde{t}_i}$. We choose the approximating family, $\tilde{t}_i$ to be exponential, to make it easier to work with comparing to $t_i$ itself. Thus we want to approximate any factor $t_i$ with $\tilde{t}_i$, such that the global difference between the exact distribution and the approximating family, KL $(p(\mathbf{y}|\mathbf{x})||q(\mathbf{y})_\theta)$ is minimized. It can be shown that the global minimization could be achieved via a set of local minimizations [6]. Note that in this formulation, $\tilde{t}_i$ don't need to be normalized or a proper distribution. For example they can be Gaussians with negative variance (an improper distribution). But given the set of $\{\tilde{t}_i\}_{i=1}^n$ we can normalized their multiplication and get a proper distribution.

**Algorithm 1:** Expectation Propagation

```
Initialize $\{\tilde{t}_i\}$
$q_\theta(\mathbf{y}) = \frac{\prod_i \tilde{t}_i}{\int \prod_i \tilde{t}_i}$
repeat
    Message elimination: Choose a $\tilde{t}_i$ to do approximation with.
    Remove the factor $\tilde{t}_i$ from approximation, $q_\theta^{-i} = \frac{q_\theta}{\tilde{t}_i}$
    Belief projection: Project the approximate posterior, with $\tilde{t}_i$
    replaced with $t_i$, on the approximating family,
        $q_\theta^{new}(\mathbf{y}) = \text{proj}\left(\hat{p}_i(\mathbf{y}) \to q_\theta(\mathbf{y})\right),$
    where,
        $\hat{p}_i(\mathbf{y}) = \frac{1}{Z} q_\theta^{-i}(\mathbf{y}) t_i(\mathbf{y}), \;\; Z = \int q_\theta^{-i}(\mathbf{y}) \times t_i(\mathbf{y}) d\mathbf{y}$
    Message update: Compute the new approximating factor,
        $\tilde{t}_i = Z \frac{q_\theta^{new}(\mathbf{y})}{q_\theta^{-i}(\mathbf{y})}$
until all $\tilde{t}_i$ converge;
```

In the case when the approximating family is from the exponential family, $q_\theta(\mathbf{y}) \propto \exp\left(\eta^\top \phi(\mathbf{y})\right)$, the projection in algorithm 1, $q_\theta^{new}(\mathbf{y}) = \text{proj}\left(\hat{p}_i(\mathbf{y}) \to q_\theta(\mathbf{y})\right)$ is equivalent to the following moment matching,

$$\mathbb{E}_{q_\theta(\mathbf{y})}\left[\phi(\mathbf{y})\right] = \mathbb{E}_{\hat{p}_i(\mathbf{y})}\left[\phi(\mathbf{y})\right] \tag{7}$$

Note that this moment matching is distributed over each factor.

To find the marginal likelihood, it is enough to find the following expression:

$$p(\mathbf{x}) \approx \int p(\mathbf{x}|\mathbf{y}) p(\mathbf{y}) d\mathbf{y} = \int \prod_i \tilde{t}_i(\mathbf{y}) d\mathbf{y}.$$

## 4 Problems with the standard EP

EP, though giving nice approximations in many applications is sensitive to outliers, and the cases when the approximating family is not close to the target distribution. This issue is addressed in [7] and a relaxed form of EP is introduced which has better convergence properties. Another problem with EP is that, there is no known convergence proof for it.

## 5 EP for inference in graphical models

In a graphical model we can interpret each $\tilde{t}_i$ as an approximate message to the original message $t_i$. There is a nice discussion of using EP for general factor graphs in [8]. [6] also provides a unifying view of different message passing algorithms on graphs. Here we give some examples on how to use EP for inference in graphical models.

We can represent the distribution in a factor-graph as following[2],

$$p(\mathbf{x}) = \frac{1}{\mathcal{Z}} \prod_i f_i(\mathbf{x})$$

where $\mathcal{Z}$ is the partition function. To compute $\mathcal{Z}$, since we need to sum over all of the variables $\mathbf{x}$, this could be quite computationally expensive. Given this product of factors, in which it is hard to marginalize over, we can create an approximate product of factors using EP. In other words, we create another graphical model, in which the marginals are as close as possible to the marginals in the original graph. But in this graph, the it is easier to find the marginals.
Note that it is always possible to approximate any factor with several variables with several approximating functions. For example if a factor $f(x_a, x_b, x_c)$ is a function of three variables, it could be approximated with three approximating factors(approximate it with a function of three variables is computationally intensive, and approximating with one single-variable function is insufficient). This is easy to visualize for pairwise factors, in a graphical model. Consider Figure 2. In the lest-side there is a pairwise factor which is being approximated by two single-variable factors in the right.

In Figure 2 (left) an HMM is represented in factor-graph notation. Each of the factor could decomposed into two disjoint factors. In practice, when creating $\hat{p}_i$ (in the Algorithm 1) we decompose only one of the factors and use its approximation in our updates.

[2] If you have problem with factor graphs, see this: `web.engr.illinois.edu/~khashab2/learn/graph.pdf`

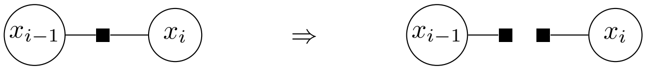

Figure 2: Decomposition of the pairs edges in factor graph

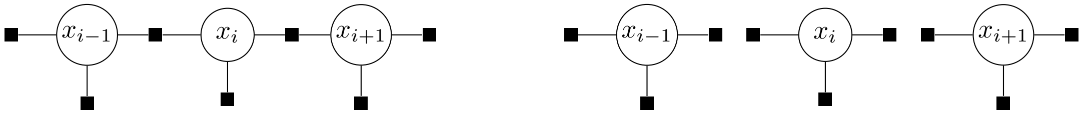

Figure 3: Decomposition of the factors in an HMM

## 6 EP energy function

There is a discussion of primal/dual energy minimization for EP in [9]. The primal energy is given in the following lemma.

**Lemma 1** (EP primal)**.** *The primal energy function is the following,*

$$\min_{\hat{p}_i} \max_{q} \left[ \sum_i \int_{\mathbf{y}} \hat{p}_i(\mathbf{y}) \log \frac{\hat{p}_i(\mathbf{y})}{t_i(\mathbf{y}) p(\mathbf{y})} d\mathbf{y} - (n-1) \int_{\mathbf{y}} q_\theta(\mathbf{y}) \log \frac{q_\theta(\mathbf{y})}{p(\mathbf{y})} d\mathbf{y} \right] \quad (8)$$

*with the constraints that*

$$\mathbb{E}_{q_\theta(\mathbf{y})}\left[\phi(\mathbf{y})\right] = \mathbb{E}_{\hat{p}_i(\mathbf{y})}\left[\phi(\mathbf{y})\right], \forall i \qquad \textit{(The local moment matching, in Equation 7)} \quad (9)$$

Basically by assigning energy function to a graphical model, we assign a global function to it, that is being optimized in a distributed fashion. This approach has been previously used to model the behaviour of other distributed inference algorithms like Belief Propagation. For more details on energy minimization schemas see [10].

Justification of the above energy function is not hard. Basically we are minimizing a linear combination of several objectives, using an arbitrary divergence measure $D(.||.)$:

$$\begin{cases} \min_{\{\hat{p}_i\}} \min_q \sum_i \alpha_i D(\hat{p}_i || q) + \beta D(q||p) \\ \mathbb{E}_{q_\theta(\mathbf{y})}\left[\phi(\mathbf{y})\right] = \mathbb{E}_{\hat{p}_i(\mathbf{y})}\left[\phi(\mathbf{y})\right], \forall i \end{cases}$$

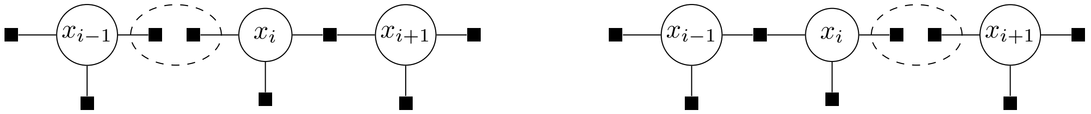

Figure 4: Decomposing each pairwise edge at each step of EP.

Choosing $D(.||.)$ to be the *KL*-divergence, $\alpha = 1$ and $\beta = (n-1)$ will give the desired result. Note that we chose these values intentionally in a way that would result in our distributed updates. Note that this is not a proof, this is a justification[3]. The proof of this is by finding the model evidence. To see the details, see Equation (66) at [6], when $\alpha = 1$.

Using the primal form, one can obtain the dual form, and by setting the gradient of the dual form to zero, one can find the fixed-point updates introduced in the previous sections. In the rest, we show how to obtain this function.

**Lemma 2** (Variational lower bound on KL divergence)**.** *If $p(.)$ and $q(.)$ are proper distributions defined on $\mathcal{X}$, we can show that,*

$$KL(p||q) = \int_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)} dx = \max_{\nu} \left[ \int_{x \in \mathcal{X}} p(x) v(x) dx - \log \int_{x \in \mathcal{X}} q(x) e^{v(x)} \right]$$

*Proof.* Easy enough to take functional derivatives with respect to $\nu(.)$ and observe that $\nu(x) = \log \frac{p(x)}{q(x)} + c$ is indeed a global maximizer for the terms inside the max[.] function. □

Now we derive the dual EP energy.

**Lemma 3** (The dual EP energy)**.** *The dual enegy function for EP is,*

$$\min_{\nu} \max_{\lambda} \left[ (n-1) \log \int_{\mathbf{y}} p(\mathbf{y}) \exp\left(\nu^\top \phi(\mathbf{y})\right) d\mathbf{y} - \sum_{i=1}^{n} \log \int_{\mathbf{y}} \hat{t}_i(\mathbf{y}) p(\mathbf{y}) \exp\left(\lambda_i{}^\top \phi(\mathbf{y})\right) d\mathbf{y} \right], \tag{10}$$

*with the constraint that,*

$$(n-1)\nu = \sum_i \lambda_i.$$

[3] In fact, a very bad justification!

*Proof.* The dual can be found by applying Lemma 2 for several times. Consider the first part of equation 8, which can be lower-bounded using Lemma 2.

$$\int_{\mathbf{y}} \hat{p}_i(\mathbf{y}) \log \frac{\hat{p}_i(\mathbf{y})}{t_i(\mathbf{y})p(\mathbf{y})} d\mathbf{y} = \max_{\lambda} \left[ \int_{\mathbf{y}} \hat{p}(\mathbf{y})\lambda_i(\mathbf{y}) d\mathbf{y} - \log \int_{\mathbf{y}} t_i(\mathbf{y})p(\mathbf{y}) \exp \lambda_i(\mathbf{y}) d\mathbf{y} \right]$$

Define $\lambda_i(\mathbf{y}) = {\lambda_i}^\top \phi(\mathbf{y})$, also using the constraint in equation 9,

$$\int_{\mathbf{y}} p_i(\mathbf{y}) \log \frac{\hat{p}_i(\mathbf{y})}{t_i(\mathbf{y})p(\mathbf{y})} d\mathbf{y} = \max_{\lambda} \left[ \sum_i \lambda_i \int_{\mathbf{y}} \hat{q}_\theta(\mathbf{y})\phi_i(\mathbf{y}) d\mathbf{y} - \log \int_{\mathbf{y}} t_i(\mathbf{y})p(\mathbf{y}) \exp \left( {\lambda_i}^\top \phi(\mathbf{y}) \right) d\mathbf{y} \right],$$

where $\phi_i(\mathbf{y})$ is the $i$-th element in the $\phi(\mathbf{y})$. Now consider the second part of equation 8 which can also be lower-bounded using Lemma 2,

$$\int_{\mathbf{y}} q_\theta(\mathbf{y}) \log \frac{q_\theta(\mathbf{y})}{p(\mathbf{y})} d\mathbf{y} = \max_{\nu} \left[ \int_{\mathbf{y}} q_\theta(\mathbf{y})\nu(\mathbf{y}) d\mathbf{y} - \log \int_{\mathbf{y}} p(\mathbf{y}) \exp \nu(\mathbf{y}) d\mathbf{y} \right]$$

$$\Rightarrow - \int_{\mathbf{y}} q_\theta(\mathbf{y}) \log \frac{q_\theta(\mathbf{y})}{p(\mathbf{y})} d\mathbf{y} = \min_{\nu} \left[ - \int_{\mathbf{y}} q_\theta(\mathbf{y})\nu(\mathbf{y}) d\mathbf{y} + \log \int_{\mathbf{y}} p(\mathbf{y}) \exp \nu(\mathbf{y}) d\mathbf{y} \right].$$

Similarly define $\nu_i(\mathbf{y}) = {\nu_i}^\top \phi(\mathbf{y})$,

$$- \int_{\mathbf{y}} q_\theta(\mathbf{y}) \log \frac{q_\theta(\mathbf{y})}{p(\mathbf{y})} d\mathbf{y} = \min_{\nu} \left[ - \sum_i \nu_i \int_{\mathbf{y}} q_\theta(\mathbf{y})\phi_i(\mathbf{y}) d\mathbf{y} + \log \int_{\mathbf{y}} p(\mathbf{y}) \exp \left( \nu^\top \phi(\mathbf{y}) \right) d\mathbf{y} \right].$$

Combining these two results we get,

$$\begin{aligned} \text{Primal} = \max_{\hat{p}_i, q, \lambda} \min_{\nu} \{ \sum_j \sum_i \lambda_i \int_{\mathbf{y}} q_\theta(\mathbf{y})\phi_i(\mathbf{y}) d\mathbf{y} - \log \int_{\mathbf{y}} t_i(\mathbf{y})p(\mathbf{y}) \exp \left( {\lambda_i}^\top \phi(\mathbf{y}) \right) d\mathbf{y} \\ - (n-1) \sum_i \nu_i \int_{\mathbf{y}} q_\theta(\mathbf{y})\phi_i(\mathbf{y}) d\mathbf{y} + (n-1) \log \int_{\mathbf{y}} p(\mathbf{y}) \exp \left( \nu^\top \phi(\mathbf{y}) \right) d\mathbf{y} \}. \end{aligned}$$

By defining $\sum_i \lambda_i = (n-1)\nu$, and dropping $q_\theta(\mathbf{y})$ as a result, we will end up with the dual form in Equation 10. □

Stable fixed points of EP correspond to the *local* minima of its energy function. By taking derivatives of the dual EP energy function, we will end-up with the updates in Algorithm 1. We show this fact in the following.

### 6.1 EP updates from dual energy

[TBW]

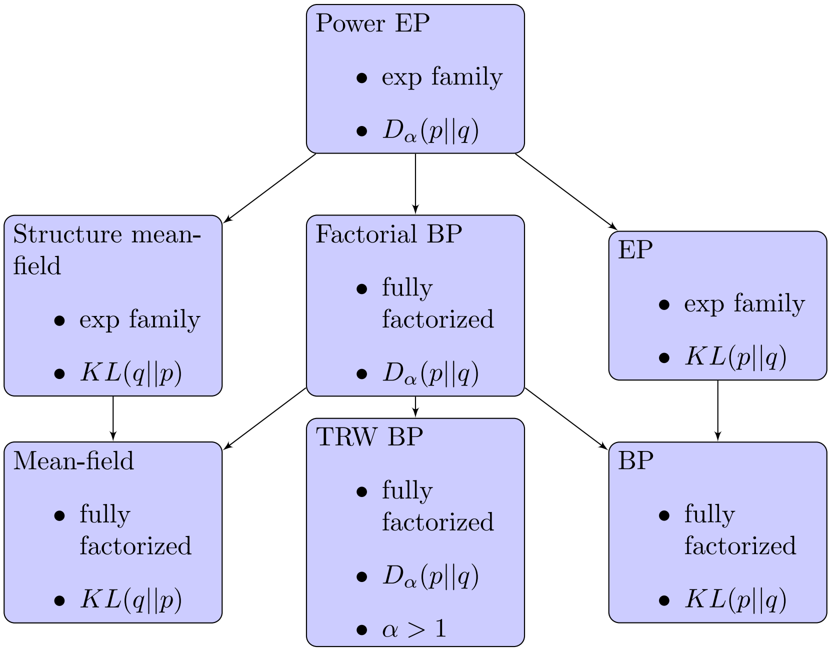

Figure 5: Hierarchies of different inference methods (From: `http://research.microsoft.com/en-us/um/people/minka/papers/message-passing/minka-message-passing-slides.pdf`. )

## 7 BP vs. EP

[More later]

## 8 Further improvements on EP

Another future work could be in making the inference more efficient. In [11] a faster way with convergence guarantees is proposed. Also at [12, 13] a faster scheme is presented. [14] is using the $\alpha$-divergence for mapping the messages. It also showed that, it includes Fractional Belief Propagation [15], EP and variational Bayes as special cases. In [16] EP has successfully used for Gaussian Process classification.

## 9 Bibliographical notes

Thanks to Doan Thanh Nam for reporting mistakes.

## References

[1] D. Koller, U. Lerner, and D. Angelov. A general algorithm for approximate inference and its application to hybrid bayes nets. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 324–333. Morgan Kaufmann Publishers Inc., 1999.

[2] Malte Kuss and Carl Edward Rasmussen. Assessing approximate inference for binary gaussian process classification. *The Journal of Machine Learning Research*, 6:1679–1704, 2005.

[3] S.L. Lauritzen. Propagation of probabilities, means, and variances in mixed graphical association models. *Journal of the American Statistical Association*, pages 1098–1108, 1992.

[4] P.S. Maybeck. *Stochastic models, estimation and control*, volume 1. Academic Pr, 1979.

[5] Thomas Minka. Power ep. *Dep. Statistics, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep*, 2004.

[6] Thomas P Minka. The ep energy function and minimization schemes. *See www. stat. cmu. edu/˜ minka/papers/learning. html*, 2001.

[7] Thomas P Minka. Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc., 2001.

[8] Tom Minka et al. Divergence measures and message passing. *Microsoft Research Cambridge, Tech. Rep. MSR-TR-2005-173*, 2005.

[9] T.P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

[10] Manfred Opper, Ulrich Paquet, and Ole Winther. Improving on expectation propagation. *Advances in Neural Information Processing Systems (NIPS)*, 2008:8–13, 2008.

[11] Jason L Pacheco and Erik B Sudderth. Improved variational inference for tracking in clutter. In *Statistical Signal Processing Workshop (SSP), 2012 IEEE*, pages 852–855. IEEE, 2012.

[12] Yuan Qi and Yandong Guo. Message passing with l1 penalized kl minimization. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, volume 28, pages 262–270. JMLR Workshop and Conference Proceedings, May 2013.

[13] Matthias W Seeger and Hannes Nickisch. Fast convergent algorithms for expectation propagation approximate bayesian inference. *arXiv preprint arXiv:1012.3584*, 2010.

[14] Charles Sutton. Expectation propagation in factor graphs:a tutorial. 2005.

[15] Wim Wiegerinck, Tom Heskes, et al. Fractional belief propagation. *Advances in Neural Information Processing Systems*, pages 455–462, 2003.

[16] Jonathan S Yedidia, William T Freeman, and Yair Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *Information Theory, IEEE Transactions on*, 51(7):2282–2312, 2005.