[Paper review 7]

Expecation Propagation for Approximate Bayesian Inference

(Thomas P Minka, 2001)

[Contents]

0. Abstract

Expectation Propagation : (1) + (2)

- (1) ADF (Assumed-Density Filtering)
- (2) Loopy belief propagation

approximates the belief states by only retaining expectations (such as mean, variance)

and iterates until these expectations are consistent!

Lower computational cost than ...

- Laplace's method, variational bayes, MC

1. introduction

Bayesian Inference : require large computational expense

Expectation Propagation (EP):

- "one-pass, sequential" method for computing an approximate posterior distribution
- weakness of ADF : information discarded previously may turn out to be important later!

2. ADF (Assumed-Density Filtering)

goal : to compute "approximate posterior"

also called..."online Bayesian Learning", "moment matching", "weak marginalization"...

applicable when we have postulated a joint pdf $p(D, x)$

(D : observed , x : hidden)

find out posterior over x (= $P(x \mid D)$) and evidence (= $P(D)$)

Example

have observation from Gaussian distribution (embedded in a sea of unrelated clutter)

(w : ratio of clutter)

$$p(\mathbf{y} \mid \mathbf{x}) = (1-w)\mathcal{N}(\mathbf{y}; \mathbf{x}, \mathbf{I}) + w\mathcal{N}(\mathbf{y}; \mathbf{0}, 10\mathbf{I})$$
$$\mathcal{N}(\mathbf{y}; \mathbf{m}, \mathbf{V}) = \frac{\exp\left(-\frac{1}{2}(\mathbf{y}-\mathbf{m})^{\mathrm{T}}\mathbf{V}^{-1}(\mathbf{y}-\mathbf{m})\right)}{|2\pi\mathbf{V}|^{1/2}}$$

d dimensional vector x has Gaussian prior :

- $p(\mathbf{x}) \sim \mathcal{N}\left(\mathbf{0}, 10\mathbf{I}_d\right)$

joint pdf of x and D (where $D = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$) :

- $p(D, \mathbf{x}) = p(\mathbf{x}) \prod_i p\left(\mathbf{y}_i \mid \mathbf{x}\right)$

How does it work?

[STEP 1] to apply ADF, re-express the joint-pdf as below

- $p(D, \mathbf{x}) = \prod_i t_i(\mathbf{x})$, where $t_0(\mathbf{x}) = p(\mathbf{x})$ and $t_i(\mathbf{x}) = p\left(\mathbf{y}_i \mid \mathbf{x}\right)$.

[STEP 2] choose an approximating family

- $q(\mathbf{x}) \sim \mathcal{N}\left(\mathbf{m}_x, v_x \mathbf{I}_d\right)$
- (choose a "spherical Gaussian")

[STEP 3] incorporate the terms t_i into the approximate posterior

- initial $q(x) = 1$
- at each step, move from old $q^{\backslash i}(\mathbf{x})$ to a new $q(\mathbf{x})$
- Incorporating the prior term is trivial ! $\hat{p}(\mathbf{x}) = \frac{t_i(\mathbf{x}) q^{\backslash i}(\mathbf{x})}{\int_{\mathbf{x}} t_i(\mathbf{x}) q^{\backslash i}(\mathbf{x}) d\mathbf{x}}$
- (each step produces normalizing factor.
  in this case, $Z_i = (1-w)\mathcal{N}\left(\mathbf{y}_i; \mathbf{m}_x^{\backslash i}, \left(v_x^{\backslash i} + 1\right)\mathbf{I}\right) + w\mathcal{N}\left(\mathbf{y}_i; \mathbf{0}, 10\mathbf{I}\right)$)

[STEP 4] minimize KL-Divergence

- $D(\hat{p}(\mathbf{x}) \| q(\mathbf{x}))$
- subject to the constraint that $q(x)$ is in the approximating family

ADF Algorithm

1. Initialize $\mathbf{m}_x = \mathbf{0}$, $v_x = 100$ (the prior). Initialize $s = 1$ (the scale factor).

2. For each data point $\mathbf{y}_i$, update $(\mathbf{m}_x, v_x, s)$ according to

$$
\begin{aligned}
s &= s^{\setminus i} \times Z_i \\
r_i &= 1 - \frac{1}{Z_i} w \mathcal{N}(\mathbf{y}_i; \mathbf{0}, 10\mathbf{I}) \\
\mathbf{m}_x &= \mathbf{m}_x^{\setminus i} + v_x^{\setminus i} r_i \frac{\mathbf{y}_i - \mathbf{m}_x^{\setminus i}}{v_x^{\setminus i} + 1} \\
v_x &= v_x^{\setminus i} - r_i \frac{(v_x^{\setminus i})^2}{v_x^{\setminus i} + 1} + r_i (1 - r_i) \frac{(v_x^{\setminus i})^2 \|\mathbf{y}_i - \mathbf{m}_x^{\setminus i}\|^2}{d(v_x^{\setminus i} + 1)^2}
\end{aligned}
$$

Intuitive Interpretation

- for each data point, compute r (= probability of not being clutter)
- make a update to $x(m_z)$
- change our confidence ! (in the estimate v_x)

BUT, depends on the order in which data is processed (because the clutter probability r depends on the current estimate of x)

3. Expectation Propagation

novel interpretation of ADF

(original ADF) treat each observation term t_i exactly $\rightarrow$ approximate posterior that includes t_i

(new interpretation) approximate t_i with $\tilde{t}_i$ $\rightarrow$ using an exact posterior with $\tilde{t}_i$

define approximation term $\tilde{t}_i$ as...

- ratio of NEW & OLD posterior
- $\tilde{t}_i(\mathbf{x}) = Z_i \frac{q(\mathbf{x})}{q^{\setminus i}(\mathbf{x})}$ (multiplying this with OLD posterior gives $q(x)$)
- (still in the same family! (exponential family))

EP Algorithm

1. Initialize the term approximations $\tilde{t}_i$
2. Compute the posterior for $\mathbf{x}$ from the product of $\tilde{t}_i$:

$$q(\mathbf{x}) = \frac{\prod_i \tilde{t}_i(\mathbf{x})}{\int \prod_i \tilde{t}_i(\mathbf{x}) d\mathbf{x}} \tag{9}$$

3. Until all $\tilde{t}_i$ converge:
   (a) Choose a $\tilde{t}_i$ to refine
   (b) Remove $\tilde{t}_i$ from the posterior to get an 'old' posterior $q^{\backslash i}(\mathbf{x})$, by dividing and normalizing:

$$q^{\backslash i}(\mathbf{x}) \propto \frac{q(\mathbf{x})}{\tilde{t}_i(\mathbf{x})} \tag{10}$$

   (c) Combine $q^{\backslash i}(\mathbf{x})$ and $t_i(\mathbf{x})$ and minimize KL-divergence to get a new posterior $q(\mathbf{x})$ with normalizer Z_i.
   (d) Update $\tilde{t}_i = Z_i q(\mathbf{x}) / q^{\backslash i}(\mathbf{x})$.
4. Use the normalizing constant of $q(\mathbf{x})$ as an approximation to $p(D)$:

$$p(D) \approx \int \prod_i \tilde{t}_i(\mathbf{x}) d\mathbf{x} \tag{11}$$