Variational Inference and Optimization I

Arto Klami

ProbAI, June 4, 2019

Introduction

- Assistant Professor at Department of Computer Science, University of Helsinki, Finland
- PhD in Computer Science from Aalto University in 2008
- Leading the *Multi-source Probabilistic Inference* group[1] with 2 postdocs and 5+ PhD students
- Statistical machine learning, approximate Bayesian inference, ML applications
- At FCAI (`fcai.fi`) working towards Agile Probabilistic AI, to make Bayesian ML practical: easier, faster, cheaper, less expertise, more reliable, ...

FCAI Finnish Center for Artificial Intelligence

REAL AI FOR REAL PEOPLE IN THE REAL WORLD

FCAI is a nation-wide competence center for Artificial Intelligence in Finland, initiated by Aalto University, University of Helsinki, and VTT Technical Research Centre of Finland.

Our mission is to create **Real AI for Real People in the Real World**—a new type of AI, which is able to operate with humans in the complex world—and to renew the Finnish industry with the new AI.

[1] `https://www.helsinki.fi/en/researchgroups/multi-source-probabilistic-inference`

Introduction

Working with variational approximations since roughly 2007

- Bayesian interpretation of canonical correlation analysis (CCA) and inter-battery factor analysis (IBFA) [Klami et al., 2013]
- Group factor analysis (GFA) generalizing CCA for multiple data sets [Klami et al., 2015]
- Collective matrix factorization (CMF) [Klami et al., 2014], topic models [Virtanen et al., 2012]
- Unsupervised object matching [Klami, 2013]
- Non-conjugate models; exponential family CCA [Klami et al., 2010], Polya-gamma augmentations for binary and count data [Klami, 2014]
- Efficient reparameterization gradients [Sakaya and Klami, 2017]
- Calibrating variational approximation for decision problems [Kusmierczyk et al., 2019]

Contents

Today: Classical variational inference

- Bayesian inference using optimization
- Mean-field approximation
- Exponential family
- Coordinate-ascent variational inference
- Stochastic variational inference

For a good reference, see Blei et al. [2017]

Tomorrow: Modern variational inference

- Stochastic estimation of gradients
- Score function estimator
- Reparameterization gradients
- Amortized inference and VAE
- Evaluating VI
- Briefly about current research

Bayesian inference

Given a probabilistic model $p(\mathbf{x}, \boldsymbol{\theta}) = p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ with

- data $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{n}$
- parameters θ
- likelihood $p(\mathbf{x}|\theta)$
- and prior $p(\theta)$

we want to solve problems like

- Prediction: $p(\tilde{\mathbf{x}}|\mathcal{D}) = \int_{\boldsymbol{\theta}} p(\tilde{\mathbf{x}}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta}$
- Decision problems: $h = \arg\max_h \int_{\boldsymbol{\theta}} u(\boldsymbol{\theta}, h)p(\boldsymbol{\theta}|\mathcal{D})d\boldsymbol{\theta}$
- Parameter inference: Interpret $\boldsymbol{\theta}$ compatible with the data $\mathcal{D}$

Bayesian inference

For all such tasks the main computational challenge is computation of the posterior distribution

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}, \boldsymbol{\theta})}{p(\mathcal{D})},$$

where the denominator marginal likelihood

$$p(\mathcal{D}) = \int_{\boldsymbol{\theta}} p(\mathcal{D}, \boldsymbol{\theta}) d\boldsymbol{\theta}$$

is difficult to compute

Once we know the posterior, everything else is (more or less) easy

Bayesian inference

Unlike majority of machine learning, Bayesian inference is not an optimization problem. Instead, we just (approximately) apply the Bayes' rule

For simple models this can be done analytically

- Discrete $\boldsymbol{\theta}$: Integral becomes a sum (though usually still slow to compute for multidimensional $\boldsymbol{\theta}$)
- Exponential family: We know that the posterior has the same functional form as the prior, so it is enough to find rules for manipulating the parameters
- Sometimes straightforward integration may be possible

Bayesian inference: Example

Consider the model

$$p(x|\theta) = \mathcal{N}(\theta, 1), \qquad p(\theta) = \mathcal{N}(0, 1)$$

Completing the square for the joint likelihood gives

$$p(x, \theta) = \frac{1}{2\pi} e^{-\frac{1}{2}[(x-\theta)^2 + \theta^2]} = \frac{1}{2\pi} e^{-[(\theta - \frac{1}{2}x)^2 + \frac{1}{4}x^2]} = \frac{1}{2\pi} e^{-(\theta-\frac{1}{2}x)^2} e^{-\frac{1}{4}x^2} = C(x) e^{-\frac{1}{2}\tau(\theta - \frac{1}{2}x)^2}$$

for $\tau = 2$ and $C(x) = \frac{1}{2\pi} e^{-\frac{1}{4}x^2}$, and consequently

$$p(x) = \int p(x, \theta) d\theta = C(x) \sqrt{\frac{2\pi}{\tau}}.$$

Now $C(x)$ cancels out when computing the posterior:

$$p(\theta|x) = \sqrt{\frac{\tau}{2\pi}} e^{-\frac{1}{2}\tau(\theta - \frac{1}{2}x)^2}$$

which is $\mathcal{N}(\mu, \tau^{-1})$ for $\mu = \frac{x}{2}$ and precision $\tau = 2$

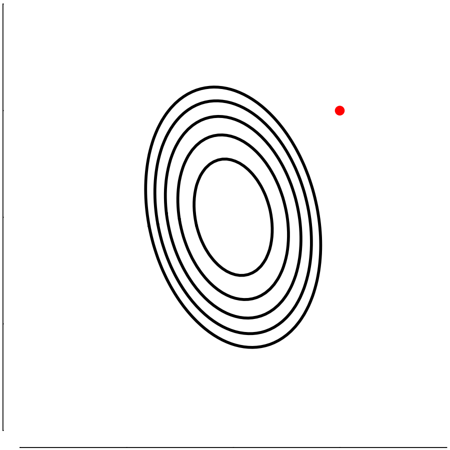
Bayesian inference

In the end we mostly care about expectations of some function over the posterior $\mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}\left[f(\boldsymbol{\theta})\right]$, so Monte Carlo approximation can be used instead

Draw $\boldsymbol{\theta}_m$ from $p(\boldsymbol{\theta}|\mathcal{D})$ using some algorithm that hopefully gives good enough samples and compute

$$\mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D})}\left[f(\mathbf{x})\right] \approx \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{\theta}_m)$$

Usually $\boldsymbol{\theta}_m$ are drawn using Markov Chain Monte Carlo (MCMC) algorithms, which provide the samples sequentially

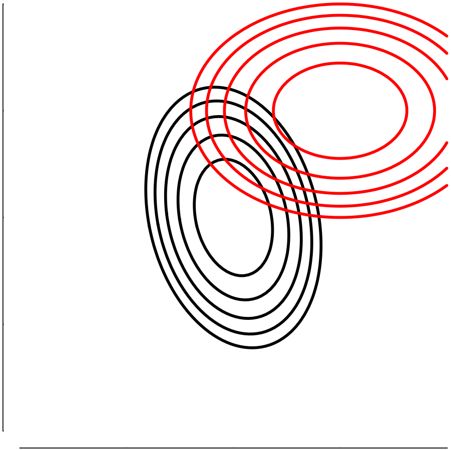


Bayesian inference using optimization

Even though MCMC algorithms provide samples from the posterior, the quality is quantified only implicitly. There is no clear learning objective, we just rely on the sampler being good enough

We can convert the search for the posterior distribution into an optimization problem as well

- Choose some parametric family of distributions $q(\boldsymbol{\theta}|\boldsymbol{\lambda})$
- Find $q(\boldsymbol{\theta}|\boldsymbol{\lambda})$ that is close to $p(\boldsymbol{\theta}|\mathcal{D})$, by minimizing some dissimilarity measure wrt $\boldsymbol{\lambda}$



Bayesian inference using optimization

Why?

- Computational efficiency
- Well-behaving objective
- Deterministic solution
- Sometimes easier to integrate into existing pipelines

Variational inference

Variational inference is one solution to handling Bayesian inference with optimization, but before we get there we need to take a look at some preliminaries

- Measuring the dissimilarity between distributions
- Exponential family of distributions

Necessary mathematical concepts

Kullback-Leibler (KL) divergence

$$D_{KL}(q,p) = \int_{\boldsymbol{\theta}} q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})} d\boldsymbol{\theta}$$

is a dissimilarity measure between probability distributions $q(\cdot)$ and $p(\cdot)$, with

- $D_{KL}(p,q) \geq 0$
- $D_{KL}(p,q) = 0$ if and only if $p(\boldsymbol{\theta}) = q(\boldsymbol{\theta}) \; \forall \; \boldsymbol{\theta}$
- It is not symmetric: $D_{KL}(q,p) \neq D_{KL}(p,q)$

(Note: There are other divergences as well, and practical algorithms using those, but KL divergence has a lot of nice properties for VI)

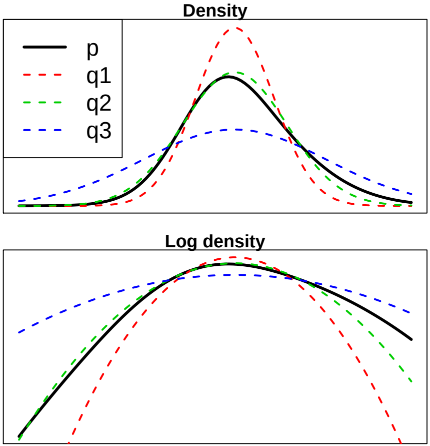
KL illustration

For $q(x) = 0$, we have $q(x) \log \frac{q(x)}{p(x)} = 0$, but for $p(x) \to 0$ when $q(x) > 0$ the value tends to ∞

Hence, the support of $q(\cdot)$ has to be within the support of $p(\cdot)$

Consequently, $q(\cdot)$ minimizing the divergence to $p(\cdot)$ will have less fat tails, and in general smaller variance

Here $D(q_2|p) < d(q_1|p) \ll d(q_3|p)$



Necessary mathematical concepts: Exponential family

Normal distribution typically parameterized by mean μ and variance σ^2 as

$$p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

Alternative parameterization using natural parameter $\boldsymbol{\eta} = [\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}]$ and density

$$p(x|\boldsymbol{\theta}) = h(x) e^{\boldsymbol{\eta}(\boldsymbol{\theta})^T \mathbf{t}(x) - a(\boldsymbol{\eta})},$$

where $\mathbf{t}(x) = [x, x^2]$ are the sufficient statistics, $a(\boldsymbol{\eta}) = -\frac{\eta_1^2}{4\eta_2} - \frac{1}{2}\log|2\eta_2|$ is the log-partition function, and $h(x) = \frac{1}{\sqrt{2\pi}}$

Simple algebraic manipulation reveals they are the same

Necessary mathematical concepts: Exponential family

This exponential family parameterization makes manipulation easier and identical for wide range of distributions: Bernoulli, binomial, Poisson, negative binomial, exponential, chi-squared, normal, lognormal, gamma, inverse-gamma, beta, multivariate normal, Dirichlet, Wishart, ...

Helps at least with writing generic equations, but may also help with software implementation

We also have several identities/tricks that help with derivations. For example,

$$\begin{aligned} a(\boldsymbol{\eta}) &= \log Z \text{ (the normalizing constant)} \\ \mathbb{E}[t(x)] &= \nabla_{\boldsymbol{\eta}} a(\boldsymbol{\eta}) \\ \mathrm{Cov}(t(x)_i, t(x)_j) &= \frac{\partial a(\boldsymbol{\eta})}{\partial \eta_i \partial \eta_j} \\ D_{KL}(p, q) &= B(\boldsymbol{\eta}_p, \boldsymbol{\eta}_q) \text{ (a Bregman divergence)} \end{aligned}$$

Exponential family

The most important property for us relates to conjugacy. For all exponential family distributions

$$p(x|\boldsymbol{\eta}) = h(x)e^{\boldsymbol{\eta}^T \mathbf{t}(x) - a(\boldsymbol{\eta})},$$

there exists a conjugate prior

$$p(\boldsymbol{\eta}|\boldsymbol{\xi}, \nu) = f(\boldsymbol{\xi}, \nu)e^{\boldsymbol{\eta}^T \boldsymbol{\xi} - \nu a(\boldsymbol{\eta})}$$

such that the posterior $p(\boldsymbol{\theta}|\mathcal{D})$ is of the same distribtution as the prior and it is easy to express for $\mathcal{D} = \{x_1, \ldots, x_n\}$ as

$$p(\boldsymbol{\eta}|\mathcal{D}) \propto e^{\boldsymbol{\eta}^T (\boldsymbol{\xi} + \sum_i \mathbf{t}(x_i)) - (\nu + n)a(\boldsymbol{\eta})}$$

In other words: Just add sufficient statistics to $\boldsymbol{\xi}$ and the number of samples to ν

Exponential family

Re-visit the example of $p(x|\theta) = \mathcal{N}(\theta, 1)$. In exponential family it is

$$p(x|\eta) = h(x)e^{\eta x - \frac{\eta^2}{2}}$$

where only one parameter is needed because the variance is known. Any distribution of the form

$$p(\eta|\xi, \nu) = f(\xi, \nu)e^{\xi\eta - \nu\frac{\eta^2}{2}}$$

is a conjugate prior. For some $p(\eta|\boldsymbol{\xi}, \nu) = \mathcal{N}(\mu, \tau^{-1})$ we can write the density over η as

$$Ce^{\tau[\mu, -\frac{1}{2}]^T \mathbf{t}(\eta)} = Ce^{\tau\mu\eta - \tau\frac{\eta^2}{2}}$$

and hence we see that the earlier $\mathcal{N}(0, 1)$ corresponds to $\xi = 0$ and $\nu = 1$, giving the posterior with $\hat{\xi} = x$ and $\hat{\nu} = 2$, which is $\mathcal{N}(\frac{x}{2}, \frac{1}{2})$ as previously

Variational approximation

Now we can finally talk about variational approximation. The goal is to find $q(\boldsymbol{\theta}) \approx p(\boldsymbol{\theta}|\mathcal{D})$ (dropping $\boldsymbol{\lambda}$ to simplify notation)

As dissimilarity measure we use $D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}|\mathcal{D}))$, where the order is important: We can integrate over $q(\boldsymbol{\theta})$ (because we choose it), but not over the unknown posterior

This gives the learning objective (to be minimized)

$$\mathcal{L}_{KL}(\boldsymbol{\lambda}) = \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathcal{D})} d\boldsymbol{\theta}$$

which we cannot compute since the posterior is indeed unknown

Variational approximation

To proceed, re-write it as

$$\mathcal{L}_{KL}(\boldsymbol{\lambda}) = \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta}) p(\mathcal{D})}{p(\boldsymbol{\theta}|\mathcal{D}) p(\mathcal{D})} d\boldsymbol{\theta} = \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta}) p(\mathcal{D})}{p(\boldsymbol{\theta}, \mathcal{D})} d\boldsymbol{\theta}$$

to convert it into a function of joint likelihood (easy to compute) and marginal likelihood (hard to compute, but not a function of $\boldsymbol{\theta}$)

Variational approximation

Since $p(\mathcal{D})$ does not depend on the variational parameters we can take it out to get

$$\begin{aligned}\mathcal{L}_{KL}(\boldsymbol{\lambda}) &= \int q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}, \mathcal{D})} d\boldsymbol{\theta} + \log p(\mathcal{D}) = D_{KL}(q(\boldsymbol{\theta}), p(\boldsymbol{\theta}, \mathcal{D})) + \log p(\mathcal{D}) \\ &= -\mathbb{E}_{q(\boldsymbol{\theta})}\left[\log p(\boldsymbol{\theta}, \mathcal{D})\right] - \mathcal{H}(q(\boldsymbol{\theta})) + \log p(\mathcal{D})\end{aligned}$$

When minimizing this we can ignore the constant term, and hence we have a learning objective consisting of two computable terms

- Negative expected log-density over the approximation, pushing probability mass for parameters maximizing the likelihood
- Negative entropy of the approximation, preventing collapsing the distribution to a point distribution

However, we cannot actually compute the objective value!

Variational approximation

Instead of minimizing the KL divergence we can alternatively re-write the objective as maximizing a lower bound for the marginal likelihood

$$\log p(\mathcal{D}) = \mathcal{L}_{KL}(\boldsymbol{\lambda}) - D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}, \mathcal{D}))$$

where writing $p(\boldsymbol{\theta}, \mathcal{D}) = p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})$ leads to

$$\log p(\mathcal{D}) = \mathbb{E}_{q(\boldsymbol{\theta})}\left[\log p(\mathcal{D}|\boldsymbol{\theta})\right] - D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta})) + \mathcal{L}_{KL}(\boldsymbol{\lambda})$$

Since KL-divergence is always positive, we get an alternative objective (to be maximized)

$$\mathcal{L}_{ELBO}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\boldsymbol{\theta})}\left[\log p(\mathcal{D}|\boldsymbol{\theta})\right] - D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}))$$

that combines (a) expected log-likelihood and (b) divergence between the approximation and the prior

Variational approximation

The evidence lower bound

$$\mathcal{L}_{ELBO}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\boldsymbol{\theta})}\left[\log p(\mathcal{D}|\boldsymbol{\theta})\right] - D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}))$$

only contains expectations of known parts of the model over the approximation. It is a well-defined learning objective, but we still need to be able to both evaluate and optimize for it

Note, however, that comparing $\mathcal{L}_{ELBO}$ for different approximations does not really make sense – it differs from $\log p(\mathcal{D})$ by a term whose magnitude can vary a lot

Evaluation: To compute the expectations, we need to assume $q(\boldsymbol{\theta})$ is sufficiently simple

Optimization:

- Today: Coordinate ascent and closed-form analytic updates
- Tomorrow: Stochastic gradient descent and Monte Carlo approximations

Evaluating the bound

Even though the optimization algorithm we will later develop does not require evaluating the bound, it is always a very good idea to do so for debugging and for monitoring optimization

Evaluating

$$\mathcal{L}_{ELBO}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\boldsymbol{\theta})}\left[\log p(\mathcal{D}|\boldsymbol{\theta})\right] - D_{KL}(q(\boldsymbol{\theta})|p(\boldsymbol{\theta}))$$

requires computing an integral over the approximating family – when is this easy?

If $\boldsymbol{\theta} \in \mathbb{R}^D$, we have a D-dimensional integral, and we need to compute the expectation of $\log p(\mathcal{D}|\boldsymbol{\theta})$ – it is a scalar function, but depends on all D parameters and may be slow to compute (for large n)

Mean-field approximation

The most common approach for ensuring we can compute various expectations over $q(\boldsymbol{\theta})$ is to assume it factorizes into a product of lower-dimensional terms

This is called the mean-field approximation, which in fully factorized case is

$$q(\boldsymbol{\theta}|\boldsymbol{\lambda}) = \prod_{d=1}^{D} q_d(\theta_d|\boldsymbol{\lambda}_d)$$

where $\boldsymbol{\lambda}_d$ can still have multiple dimensions (e.g. μ and σ^2 for univariate normal distribution)

Now $\mathbb{E}_{q(\boldsymbol{\theta})}\left[f(\boldsymbol{\theta})\right]$ becomes a nested collection of D one-dimensional integrals

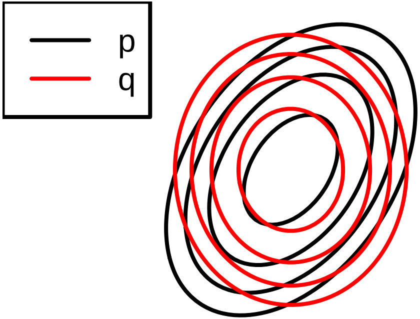
$$\mathbb{E}_{q(\boldsymbol{\theta})}\left[f(\boldsymbol{\theta})\right] = \int \dots \int \prod_{d=1}^{D} q_d(\theta_d|\boldsymbol{\lambda}_d) f(\boldsymbol{\theta}) d\theta_1 \dots d\theta_D,$$

which is typically considerably easier

Mean-field approximation

Reminder: For small $D_{KL}(q|p)$ the support of $q(\cdot)$ needs to be within the support of $p(\cdot)$

VI fits the approximation "inside" the posterior, to avoid putting probability mass for $p(\mathcal{D}) \approx 0$, and hence always underestimates variance. Mean-field makes the issue worse



Mean-field approximation

For the mean-field approximation the objective can be re-written as

$$\mathcal{L}_{ELBO}(\boldsymbol{\lambda}) = \int \dots \int q(\boldsymbol{\theta}) \log p(\mathcal{D}|\boldsymbol{\theta}) d\theta_1 \dots d\theta_D - \sum_{d=1}^{D} D_{KL}(q(\theta_d|\lambda_d)|p(\theta_d))$$

where the KL divergences separate because they are independent, but the expected log-likelihood still involves the whole joint distribution

To optimize this we can use coordinate ascent, which allows maximizing the objective wrt to any of $q(\theta_d|\lambda_d)$ at a time

However, since we did not define $q(\theta_d|\lambda_d)$, there is no obvious parameterization. Instead, we need to perform variational calculus to optimize over distributions

Detour: Coordinate ascent

Coordinate ascent is one of the simplest iterative optimization algorithms

To maximize some objective $\mathcal{L}(\boldsymbol{\theta})$ for $\boldsymbol{\theta} \in \mathbb{R}^D$, we iteratively

1. Select one dimension $d \in [1, \ldots, D]$
2. Maximize $\mathcal{L}(\boldsymbol{\theta})$ by modifying only θ_d, keeping all other dimensions of $\boldsymbol{\theta}$ fixed

Solving the one-dimensional optimization problems is often quite easy; closed-form updates, line search, ...

Naturally only converges to a local optimum for non-convex objectives

Detour: Langrange multipliers

Optimization problems of type

$$\max_{\boldsymbol{\lambda}} \; f(\boldsymbol{\lambda})$$
$$\text{s.t. } g(\boldsymbol{\lambda}) = 0$$

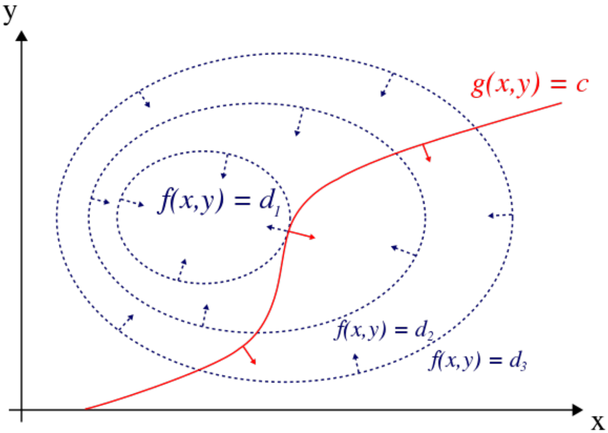
can be solved by forming the lagrangian

$$\mathcal{L}_L(\boldsymbol{\lambda}) = f(\boldsymbol{\lambda}) - \alpha g(\boldsymbol{\lambda}),$$

where $\alpha \geq 0$ is a langrange multiplier, and solving for $\nabla \mathcal{L}_L(\boldsymbol{\lambda}) = 0$

"Proof": We can always move along the constraint towards optimal solution, unless $\nabla f(\boldsymbol{\lambda})$ is orthogonal to $g(\boldsymbol{\lambda})$ (that is, $\nabla f(\boldsymbol{\lambda}) = \alpha \nabla g(\boldsymbol{\lambda})$ for some α).

Detour: Lagrange multipliers



Mean-field approximation

So, let's optimize for $q(\theta_d|\lambda_d)$ such that $\int q(\theta_d|\lambda_d) = 1$ (it is a distribution)
The lagrangian of the optimization problem is then

$$\int \dots \int q(\boldsymbol{\theta}) \log p(\mathcal{D}, \boldsymbol{\theta}) d\theta_1 \dots d\theta_D - \int q(\theta_d) \log q(\theta_d) d\theta_d - \alpha \left(\int q(\theta_d) d\theta_d - 1 \right)$$

Start by re-writing the first term as

$$\int_{q_d} q(\theta_d) \mathbb{E}_{q_{-d}(\boldsymbol{\theta}_{-d}|\boldsymbol{\lambda}_{-d})} \left[\log p(\mathcal{D}, \boldsymbol{\theta}) \right] d\theta_d$$

where the inner expectation is computed over *all other terms* in the approximation

Mean-field approximation

Now we have the objective

$$\int q(\theta_d) \left[\mathbb{E}_{q_{-d}} [\log p(\mathcal{D}, \boldsymbol{\theta})] - \log q(\theta_d) - \alpha\right] d\theta_d + \alpha$$

which can be solved by differentiating w.r.t. to $q(\theta_d)$ (yes, we can do that) and solving for $\nabla q(\theta_d) = 0$

The derivative is

$$\mathbb{E}_{q_{-d}} [\log p(\mathcal{D}, \boldsymbol{\theta})] - \log q(\theta_d) - \alpha + \frac{q(\theta_d)}{q(\theta_d)}$$

and solving for $\nabla q(\theta_d) = 0$ gives

$$\log q(\theta_d) = \mathbb{E}_{q_{-d}} [\log p(\mathcal{D}, \boldsymbol{\theta})] + C$$

and hence

$$q(\theta_d) \propto e^{\mathbb{E}_{q_{-d}} [\log p(\mathcal{D}, \boldsymbol{\theta})]}$$

Mean-field approximation

We just derived a coordinate ascent update rule

$$q(\theta_d) \propto e^{\mathbb{E}_{q_{-d}}[\log p(\mathcal{D},\boldsymbol{\theta})]}$$

by assuming *only* that the approximating family factorizes over the dimensions, making no assumptions about the distributional family

If we know all other terms $q_j(\theta_j)$ then the update rule actually determines the distributional family as well

The only remaining problem concerns computing that expectation

Mean-field approximation and exponential family

Exponential family parameterization simplified direct posterior calculation for models with conjugate priors

It helps similarly for variational approximation: For models with conjugate priors the optimal approximating distribution is of the same form as the prior

We can use this knowledge in two ways:

- Just take this for granted and use it to simplify the calculations in ”standard parameterization”
- Directly write the models and updates in exponential family parameterization

Mean-field approximation: Example

Consider a simplified linear regression model (see Drugowitsch [2013] for a proper model)

$$
\begin{aligned}
p(y|\mathbf{w}, \mathbf{x}, \tau_y) &= \mathcal{N}(\mathbf{w}^T\mathbf{x}, \tau_y), \\
p(\mathbf{w}|\tau_{w0}) &= \mathcal{N}(0, \tau_{w0}), \\
p(\tau_y|\alpha_0, \beta_0) &= \mathcal{G}(\alpha_0, \beta_0)
\end{aligned}
$$

and assume a factorization

$$q(\boldsymbol{\theta}|\boldsymbol{\lambda}) = q_w(\mathbf{w}|\boldsymbol{\mu}_w, \boldsymbol{\Lambda}_w)q_\tau(\tau_y|\alpha, \beta)$$

where the first term is a multivariate normal distribution with precision $\boldsymbol{\Lambda}_w$, and the second term is a gamma distribution

To fit the approximation we need

- Update rule for q_w
- Update rule for q_τ
- Computation of $\mathcal{L}_{ELBO}$

Mean-field approximation: Example

Start with $q_w()$ and write the update in log-domain as

$$\log q_w = \mathbb{E}_{q_\tau}\left[\log p(y, \mathbf{x}, \mathbf{w}, \tau)\right] + C = \mathbb{E}_{q_\tau}\left[\log p(y|\mathbf{x}, \mathbf{w}) p(\mathbf{w}) p(\tau_y)\right] + C$$

Plugging the log-densities results in

$$\log q_w = \mathbb{E}_{q_\tau}\Big[-\frac{1}{2}\log 2\pi + \frac{1}{2}\log \tau_y - \frac{1}{2}\tau_y(y - \mathbf{w}^T\mathbf{x})^2 - \frac{1}{2}\log 2\pi + \frac{1}{2}\log \tau_{w0} - \frac{1}{2}\tau_{w0}\mathbf{w}^T\mathbf{w}$$
$$+ \alpha_0 \log \beta_0 - \log \Gamma(\alpha_0) + (\alpha_0 - 1)\log \tau_y - \beta \tau_y\Big] + C$$

which simplifies dramatically by ignoring all terms that do not depend on $\mathbf{w}$ (collected in C) and moving terms independent of τ_y outside the expectation

$$\begin{aligned} & -\frac{1}{2}\mathbb{E}_{q_\tau}\left[\tau_y\right](y - \mathbf{w}^T\mathbf{x})^2 - \frac{1}{2}\tau_{w0}\mathbf{w}^t\mathbf{w} + C \\ = & -\frac{1}{2}\left[\mathbb{E}_{q_\tau}\left[\tau_y\right](y^2 - 2y\mathbf{x}^T\mathbf{w} + \mathbf{w}^T\mathbf{x}\mathbf{x}^T\mathbf{w}) + \tau_{w0}\mathbf{w}^T\mathbf{w}\right] + C \end{aligned}$$

Mean-field approximation: Example

For multiple observations $\mathcal{D}$ stored as $\mathbf{X} \in \mathbb{R}^{D \times N}$ and $\mathbf{y} \in \mathbb{R}^{1 \times N}$, the log-density is

$$-\frac{1}{2}\left[\mathbb{E}_{q_\tau}\left[\tau_y\right]\left(\mathbf{y}\mathbf{y}^T - 2\mathbf{y}\mathbf{X}^T\mathbf{w} + \mathbf{w}^T\mathbf{X}\mathbf{X}^T\mathbf{w}\right) + \tau_{w0}\mathbf{w}^T\mathbf{w}\right]$$

and after quite some algebraic manipulation we see that it corresponds to a normal distribution with

$$\begin{aligned}\boldsymbol{\Lambda}_w &= \mathbb{E}_{q_\tau}\left[\tau_y\right]\mathbf{X}\mathbf{X}^T + \tau_{w0}\mathbf{I}\\ \boldsymbol{\mu}_w &= \boldsymbol{\Lambda}_w^{-1}\mathbf{X}\mathbf{y}^T\end{aligned}$$

The derivation is actually identical to that of computing $p(\mathbf{w}|\mathcal{D}, \tau_y)$ for fixed τ_y in standard Bayesian linear regression, and the only difference is that we now use the expected precision

$$\mathbb{E}_{q_\tau}\left[\tau_y\right] = \frac{\alpha}{\beta}$$

in place of τ_y that would appear when conditioning on the exact value

Mean-field approximation: Example

For $q(\tau)$ we start with the same expression but now integrate over $q(\mathbf{w})$, and hence can drop terms that do not depend on τ_y and take out of expectation all terms independent of $\mathbf{w}$

$$\log q_\tau = \mathbb{E}_{q_w}\left[-\frac{1}{2}\log 2\pi + \frac{1}{2}\log\tau_y - \frac{1}{2}\tau_y(y - \mathbf{w}^T\mathbf{x})^2 - \frac{1}{2}\log 2\pi + \frac{1}{2}\log\tau_{w0} - \frac{1}{2}\tau_{w0}\mathbf{w}^T\mathbf{w}\right.$$
$$\left. + \alpha_0\log\beta_0 - \log\Gamma(\alpha_0) + (\alpha_0 - 1)\log\tau_y - \beta\tau_y\right] + C$$

becomes

$$\frac{1}{2}\log\tau_y - \frac{1}{2}\tau_y\mathbb{E}_{q_w}\left[(y - \mathbf{w}^T\mathbf{x})^2\right] + (\alpha_0 - 1)\log\tau_y - \beta\tau_y + C$$

which immediately gives the updates

$$\alpha = \alpha_0 + \frac{1}{2}, \quad \beta = \beta_0 + \frac{1}{2}\mathbb{E}_{q_w}\left[(y - \mathbf{w}^T\mathbf{x})^2\right],$$

Mean-field approximation: Example

Again the updates for a single data point

$$\alpha = \alpha_0 + \frac{1}{2}, \quad \beta = \beta_0 + \frac{1}{2}\mathbb{E}_{q_w}\left[(y - \mathbf{w}^T\mathbf{x})^2\right],$$

look rather familiar for people familiar with the exact posteriors, where instead of the residual error we have the expectation

$$\mathbb{E}_{q_w}\left[(\mathbf{y} - \mathbf{w}^T\mathbf{X})(\mathbf{y} - \mathbf{w}^T\mathbf{X})^T\right] = \mathbf{y}\mathbf{y}^T - 2\mathbf{X}^T\mathbb{E}_{q_w}[\mathbf{w}]\,\mathbf{y}^T + \mathbf{X}^T\mathbb{E}_{q_w}\left[\mathbf{w}\mathbf{w}^T\right]\mathbf{X}.$$

This can be computed by plugging in known results for multivariate normal distribution

$$\mathbf{y}\mathbf{y}^T - 2\mathbf{X}^T\boldsymbol{\mu}_w\mathbf{y}^T + \mathbf{X}^T(\boldsymbol{\Lambda}_w^{-1} + \boldsymbol{\mu}_w\boldsymbol{\mu}_w^T)\mathbf{X},$$

and for multiple data points we also have $\alpha = \alpha_0 + \frac{n}{2}$

Mean-field approximation: Example

Finally, in order to monitor progress and to verify the updates we need to compute $\mathcal{L}_{ELBO}$, which includes the expected joint likelihood

$$\mathbb{E}_{q_w q_\tau}\left[\log p(y, \mathbf{x}, \mathbf{w}, \tau_y)\right] = -\log 2\pi - \frac{1}{2}\mathbb{E}_{q_\tau}\left[\tau_y\right]\mathbb{E}_{q_w}\left[(y - \mathbf{w}^t\mathbf{x})^2\right]$$
$$-\frac{1}{2}\log \tau_{w0} + \frac{1}{2}\tau_{w0}\mathbb{E}_{q_w}\left[\mathbf{w}^T\mathbf{w}\right] - \alpha_0 \log \beta_0 + \log \Gamma(\alpha_0) - (\alpha_0 - 1)\mathbb{E}_{q_\tau}\left[\log \tau_y\right] + \beta\mathbb{E}_{q(\tau)}\left[\tau_y\right]$$

and additionally the entropies $\mathcal{H}(q_\tau)$ and $\mathcal{H}(q_w)$

The former only involves terms we have already computed, except for

$$\mathbb{E}_{q_\tau}\left[\log \tau_y\right] = \psi(\alpha) - \log(\beta)$$
$$\mathbb{E}_{q_w}\left[\mathbf{w}^T\mathbf{w}\right] = \mathrm{Tr}(\mathbf{\Lambda}_w^{-1}),$$

where $\psi(\cdot)$ is the digamma function. The entropies, in turn, are available in literature

Mean-field approximation

In the end we have

- Closed-form updates that only depend on expectations of the other approximation factors, and that look a lot like conditional distributions of the model
- Closed-form ELBO that only depends on expectations
- Coordinate ascent algorithm that converges to local optimum of ELBO by non-decreasing updates

The derivation, however, was quite involved, requiring some non-trivial algebraic manipulation and knowledge on non-trivial expectations (log of a gamma variable, second-order terms for normal distribution)

Mean-field approximation

The fact that the updates look like conditional distributions is not a coincidence. Instead of

$$q(\theta_d) \propto e^{\mathbb{E}_{q_{-d}}[\log p(\mathcal{D},\boldsymbol{\theta})]}$$

we can just as well use

$$q(\theta_d) \propto e^{\mathbb{E}_{q_{-d}}[\log p(\theta_d|\boldsymbol{\theta}_{-d},\mathcal{D})]}$$

since

$$p(\mathcal{D},\boldsymbol{\theta}) = p(\boldsymbol{\theta}_d|\mathcal{D},\boldsymbol{\theta}_{-d})p(\mathcal{D},\boldsymbol{\theta}_{-d}),$$

where the second term is constant wrt to θ_d and does not matter

Note that these are the same conditional distributions we would need to derive for Gibbs sampling, so in case one already did this it is easy to write mean-field updates as well

Mean-field approximation and exponential family

If the conditional is in exponential family, it can be written as

$$p(\theta_d | \dots) = h(\theta_d) e^{-\eta_d(\boldsymbol{\theta}_{-d}, \mathcal{D})^T \theta_d - a(\eta_d(\boldsymbol{\theta}_{-d}, \mathcal{D}))},$$

where the parameter η depends on some subset of the other parameters

Now we can write the update as

$$\begin{aligned} q(\theta_d) &\propto e^{\log h(\theta_d) + \mathbb{E}_{q_{-d}}[\eta_d(\boldsymbol{\theta}_{-d}, \mathcal{D})]^T \theta_d - \mathbb{E}_{q_{-d}}[a(\eta_d(\boldsymbol{\theta}_{-d}, \mathcal{D}))]} \\ &\propto h(\theta_d) e^{\mathbb{E}_{q_{-d}}[\eta_d(\boldsymbol{\theta}_{-d}, \mathcal{D})]^T \theta_d}, \end{aligned}$$

only requiring the expected natural parameter

Allows for compact notation, but does not always help with practical derivations; you still need to solve for the conditional distributions and compute the expectations

Mean field approximation and exponential family

For latent variable models of the form

$$p(\beta, \mathbf{z}, \mathbf{x}) = p(\beta) \prod_{i=1}^{n} p(z_i, x_i | \beta)$$

with n local latent variables z_i and some global parameters β, conjugate conditionals and $q(\cdot) = q(\beta|\boldsymbol{\lambda}) \prod_i q(z_i|\psi_i)$, the updates become:

- For local variables: $\psi_i = \mathbb{E}_{q(\beta)}\left[\eta(\beta, x_i)\right]$
- For parameters: $\boldsymbol{\lambda} = \left[\alpha_1 + \sum_{i=1}^{n} \mathbb{E}_{q(z_i)}\left[t(z_i, x_i)\right], \alpha_2 + n\right]$ for some prior parameters α

Between all global updates we need to update all local parameters, with complexity $\mathcal{O}(n)$. This can be remedied with stochastic VI, computing the expectation based on $m \ll n$ samples [Hoffman et al., 2013]

Stochastic VI

Instead of closed-form update for $\boldsymbol{\lambda}$, we perform gradient-based optimization wrt to $\boldsymbol{\lambda}$ (still updating ψ_i with closed-form updates)

It turns out that the natural gradient (gradient multiplied with the inverse Fisher information matrix) of the bound wrt to $\boldsymbol{\lambda}$ is

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}_{ELBO} = \hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}$$

where $\boldsymbol{\lambda}$ is the current value of the parameter and $\hat{\boldsymbol{\lambda}}$ is the result of coordinate ascent step, and a gradient step with length ϵ then becomes

$$\lambda_{t+1} = (1 - \epsilon)\boldsymbol{\lambda}_t + \epsilon \hat{\boldsymbol{\lambda}}_t$$

Here we can plug in any stochastic estimate for $\hat{\boldsymbol{\lambda}}_t$, e.g. $\hat{\boldsymbol{\lambda}}_t \approx \left[\alpha_1 + n\mathbb{E}_\psi\left[t(z_i, x_i)\right], \alpha_2 + n\right]$ using $m = 1$ samples

Beyond conjugacy

Everything above holds only for mean-field approximations of conditionally conjugate models

For non-conjugate models we need to introduce additional bounds or augment the model to make it conditionally conjugate. This is often hard and practical solutions exist only for some simple cases

For example, for logistic regression we can use:

- Jaakkola bound: Introduce additional variational parameter for each likelihood term, with log-quadratic form [Jordan et al., 1999]
- Bohning bound: Bound negative log-likelihood using 2nd order Taylor expansion [Seeger and Bouchard, 2012]
- Polya-gamma (PG) augmentation: Introduce a latent variable that follows the PG distribution and provides Gaussian and PG conditionals [Polson et al., 2013, Klami, 2014]

Variational inference, classical

Re-cap

- VI fits approximation by minimizing it's KL divergence to the real posterior
- Converted into practical learning problem as maximizing a lower bound for the marginal likelihood
- Mean-field approximation leads to elegant closed-form update rule for coordinate ascent
- For conditionally conjugate models the updates only depend on expectations over other approximating terms
- Exponential family helps writing nice equations, but not necesssarily in implementation
- Stochastic optimization over latent variables is possible, but needs to be done right

Variational inference, classical

Core limitations

- Underestimates uncertainty
- ELBO is useful for monitoring convergence, but not directly for comparing models or approximations
- Moving outside of mean-field is difficult
- Moving outside of conditionally conjugate models is difficult (but not impossible)
- The derivations are slow, error-prone and typically waste of time

References I

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518), 2017.

Jan Drugowitsch. Variational bayesian inference for linear and logistic regression. In *arXiv:1310.5438*, 2013.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

Arto Klami. Bayesian object matching. *Machine Learning*, 92(2):225–250, 2013.

Arto Klami. Polya-gamma augmentations for factor models. In *Proceedings of the Asian Conference on Machine Learning*, 2014.

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian exponential family projections for coupled data sources. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, 2010.

References II

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian canonical correlation analysis. *Journal of Machine Learning Research*, 14:965–1003, 2013.

Arto Klami, Guillaume Bouchard, and Abhishek Tripathi. Group-sparse embeddings in collective matrix factorization. In *Proceedings of the International Conference on Learning Representations*, 2014.

Arto Klami, Seppo Virtanen, Eemeli Leppäaho, and Samuel Kaski. Group factor analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 26(9):2136–2147, 2015.

Tomasz Kusmierczyk, Joseph Sakaya, and Arto Klami. Variational Bayesian decision-making for continuous utitilies. In *arXiv:1902.00792*, 2019.

Nicholas G. Polson, James G. Scott, and Jess Windle. Bayesian inference for logistic models using polya-gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

Joseph Sakaya and Arto Klami. Importance sampled stochastic optimization for variational inference. In *Proceedings of Uncertainty in Artificial Intelligence*, 2017.

Matthias Seeger and Guillaume Bouchard. Fast variational Bayesian inference for non-conjugate matrix factorization models. In *Proceedings of AISTATS*, 2012.

References III

Seppo Virtanen, Yangqing Jia, Arto Klami, and Trevor Darrell. Factorized multi-modal topic model. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence*, 2012.