Probabilistic Inference and Learning
Lecture 24
An Extensive Example II

Philipp Hennig

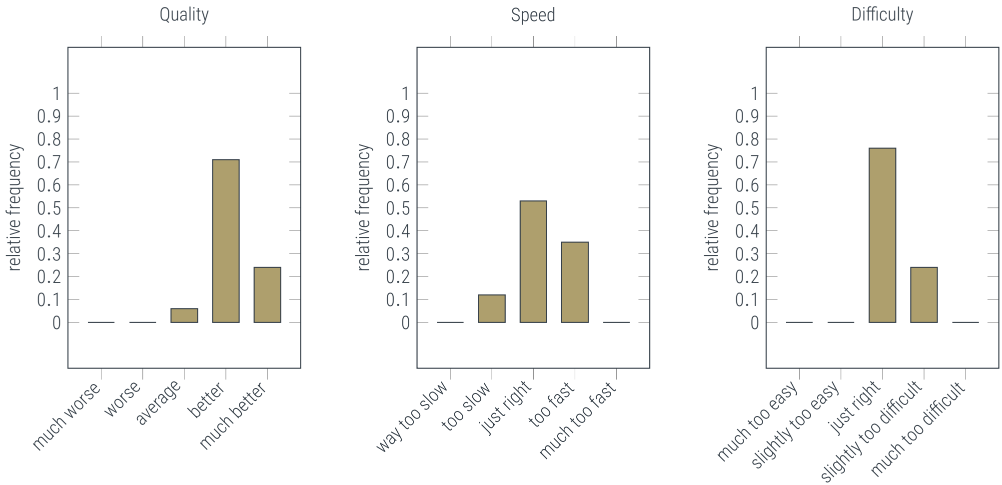
23 January 2019

Eberhard Karls Universität Tübingen

Faculty of Science
Department of Computer Science
Chair for the Methods of Machine Learning

Last Lecture: Debrief

Feedback dashboard



Last Lecture: Debrief

Detailed Feedback

Things you did not like:

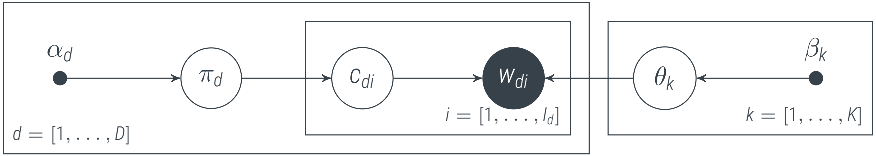
- not caring about word order
- the indices / indexing errors
- using both Π and π
- putting the equations together
- Exercise 11 was too hard, more exam-like questions, please.

Things you did not understand:

- what are π_{dk} and θ_{kv}, intuitively?
- how can we measure the performance of the approaches?

Things you enjoyed:

- **the Example**
- the “thumb of uncertainty”
- the directed graph
- trying out naïve solutions first



To draw I_d words $w_{di} \in [1, \dots, V]$ of document $d \in [1, \dots, D]$:

- Draw K topic distributions θ_k over V words from $\quad p(\Theta \mid \boldsymbol{\beta}) = \prod_{k=1}^{K} \mathcal{D}(\theta_k; \beta_k)$
- Draw D document distributions over K topics from $\quad p(\Pi \mid \boldsymbol{\alpha}) = \prod_{d=1}^{D} \mathcal{D}(\pi_d; \alpha_d)$
- Draw topic assignments c_{ik} of word w_{id} from $\quad p(C \mid \Pi) = \prod_{i,d,k} \pi_{id}^{c_{dik}}$
- Draw word w_{id} from $\quad p(w_{id} = v \mid c_{di}, \Theta) = \prod_k \theta_{kv}^{c_{dik}}$

Useful notation: $n_{dkv} = \#\{i : w_{di} = v, c_{ijk} = 1\}$. Write $n_{dk:} := [n_{dk1}, \dots, n_{dkV}]$ and $n_{dk\cdot} = \sum_v n_{dkv}$, etc.

The Generative Model

Joint Distribution over Latents and Data

Recall definitions

$$\mathcal{D}(\boldsymbol{\pi};\boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \pi_k^{\alpha_k - 1} \qquad \text{and} \qquad n_{dkv} = \#\{i : w_{di} = v, c_{ijk} = 1\}, n_{dk\cdot} = \sum_v n_{dkv}$$

Thus

$$\begin{aligned}
p(C,\Pi,\Theta,W) &= \left(\prod_{d=1}^{D} p(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha}_d)\right) \cdot \left(\prod_{d=1}^{D}\prod_{i=1}^{I_d} p(c_{di} \mid \pi_d)\right) \cdot \left(\prod_{d=1}^{D}\prod_{i=1}^{I_d} p(w_{di} \mid c_{di},\Theta)\right) \cdot \left(\prod_{k=1}^{K} p(\boldsymbol{\theta}_k \mid \boldsymbol{\beta}_k)\right) \\
&= \left(\prod_{d=1}^{D} \mathcal{D}(\boldsymbol{\pi}_d;\boldsymbol{\alpha}_d)\right) \cdot \left(\prod_{d=1}^{D}\prod_{i=1}^{I_d} \left(\prod_{k=1}^{K} \pi_{dk}^{c_{dik}}\right)\right) \cdot \left(\prod_{d=1}^{D}\prod_{i=1}^{I_d} \left(\prod_{k=1}^{K} \theta_{kw_{di}}^{c_{dik}}\right)\right) \cdot \left(\prod_{k=1}^{K} \mathcal{D}(\boldsymbol{\theta}_k;\boldsymbol{\beta}_k)\right) \\
&= \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}}\right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}}\right)
\end{aligned}$$

Designing a Probabilistic Machine Learning Model

1. Take a close look at the **Data**
2. Think about modelling goals, decide on data types
3. Design the **Model**
4. Design the **Algorithm**
   - For conditionally independent parts, conjugate priors may give analytic inference
   - for linearly related variables, consider Gaussians
   - consider revisiting the model to simplify as much as possible
   - if there is still no analytic answer, use **The Toolbox**!

The Toolbox

Five principal methods for dealing with computational complexity in probabilistic inference

1. **Maximum Likelihood (ML) / Maximum A-Posteriori (MAP)** estimation:
   To estimate θ in $p(D \mid \theta)$ or $p(\theta \mid D)$, set $\hat{\theta} = \arg\max_\theta p$.
2. **Laplace** Approximation: $p(\theta \mid D) \approx \mathcal{N}\left(\theta; \hat{\theta}, -(\nabla\nabla^\intercal \log p(\theta \mid D))^{-1}\right)$
3. **Variational Inference:**
   To approximate $p(\theta \mid D)$, impose structure on $q(\theta)$, then minimize $D_{\mathrm{KL}}(q \| p)$
4. **Monte Carlo:** $\int f(x)p(x)\,\mathrm{d}x \approx \sum_i f(x_i)$ where $x_i \sim p$
5. **Numerical Quadrature:** $\int p(f \mid \theta)p(\theta)\,d\theta \approx \sum_i w_i \cdot p(f \mid \theta_i)$

Disclaimer: The listed items are neither mutually exclusive nor collectively exhaustive. Some of the methods are intricately interrelated.

A Variational Bound

Find the *best simple* way to describe a *complex* thing

reminder: $n_{dkv} = \#\{i : w_{di} = v, c_{ijk} = 1\}$. $n_{dk\cdot} = \sum_v n_{dkv}$, etc.

$$p(C, \Pi, \Theta, W) = \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}} \right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}} \right)$$

- The posterior $p(\Pi, \Theta, C \mid W)$ is intractable. We want an approximation q that *factorises*

$$q(\Pi, \Theta, C) = \left(\prod_{d,i} q(\boldsymbol{c}_{di}) \right) \left(\prod_{d} q(\boldsymbol{\pi}_d) \right) \left(\prod_{k} q(\boldsymbol{\theta}_k) \right)$$

- To find the *best* such approximation – the one that *minimizes* $D_{\mathrm{KL}}(q \| p(\Pi, \Theta, C \mid W))$, we *maximize* the **ELBO** (minimize variational free energy)

$$\mathcal{L}(q) = \int q(C, \Theta, \Pi) \log \left(\frac{p(C, \Pi, \Theta, W)}{q(C, \Theta, \Pi)} \right) dC \, d\Theta \, d\Pi$$

$$p(C, \Pi, \Theta, W) = \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}} \right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}} \right)$$

Recall from Lecture 20: To maximize the ELBO of a factorized approximation, compute the **mean field**

$$\log q^*(z_i) = \mathbb{E}_{z_j, j\neq i}(\log p(x, z)) + \text{const.}$$

$$\log q^*(\boldsymbol{c}_{di}) = \mathbb{E}_{\prod_{d,j\neq i} q(\boldsymbol{c}_{dj}) q(\boldsymbol{\pi}_d) \prod_k q(\boldsymbol{\theta}_k)} \left(\sum_{d,k} (\alpha_{dk} - 1 + n_{dk\cdot}) \log \pi_{dk} + \sum_{k,v} (\beta_{kv} - 1 + n_{\cdot kv}) \log \theta_{kv} \right) + \text{const.}$$

$$= \sum_{k=1}^{K} c_{dik} \underbrace{\left(\mathbb{E}_{q(\pi_{dk})}(\log \pi_{dk}) + \mathbb{E}_{q(\theta_{di})}(\log \theta_{kw_{di}}) \right)}_{=:\log \gamma_{dik}} + \text{const.}$$

Thus, $q(\boldsymbol{c}_{di}) = \prod_k \tilde{\gamma}_{dik}^{c_{dik}}$, where $\tilde{\gamma}_{dik} = \gamma_{dik} / \sum_k \gamma_{dik}$ (Note: Thus, $\mathbb{E}_q(c_{dik}) = \tilde{\gamma}_{dik}$)

$$p(C, \Pi, \Theta, W) = \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}}\right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}}\right)$$

Recall from Lecture 20: To maximize the ELBO of a factorized approximation, compute the **mean field**

$$\log q^*(z_i) = \mathbb{E}_{z_j, j\neq i}(\log p(x, z)) + \text{const.}$$

$$\log q^*(\boldsymbol{\pi}_d) = \mathbb{E}_{\prod_{e\neq d,i} q(\boldsymbol{c}_{ei}) q(\boldsymbol{\pi}_e) \prod_k q(\boldsymbol{\theta}_k)} \left(\sum_{d,k} (\alpha_{dk} - 1 + n_{dk\cdot}) \log \pi_{dk} + \sum_{k,v} (\beta_{kv} - 1 + n_{\cdot kv}) \log \theta_{kv}\right) + \text{const.}$$

$$= \sum_{k=1}^{K} (\alpha_{dk} - 1 + \mathbb{E}_{q(C)}(n_{dk\cdot})) \log \pi_{dk} + \text{const.}$$

Thus, $q(\boldsymbol{\pi}_d) = \mathcal{D}(\boldsymbol{\pi}_d; \tilde{\alpha}_{dk} := [\alpha_{dk} + \sum_{i=1}^{I_d} \tilde{\gamma}_{dik}]_{k=1,\dots,K})$.

Constructing the Bound

Mean-Field Theory: Putting Lectures 18–20 to use

reminder: $n_{dkv} = \#\{i : w_{di} = v, c_{ijk} = 1\}$. $n_{dk\cdot} = \sum_v n_{dkv}$, etc.

$$p(C, \Pi, \Theta, W) = \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}} \right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}} \right)$$

Recall from Lecture 20: To maximize the ELBO of a factorized approximation, compute the **mean field**

$$\log q^*(z_i) = \mathbb{E}_{z_j, j \neq i}(\log p(x, z)) + \text{const.}$$

$$\log q^*(\boldsymbol{\theta}_k) = \mathbb{E}_{\prod_{d,i} q(\boldsymbol{c}_{di}) q(\boldsymbol{\pi}_d) \prod_{\ell \neq k} q(\boldsymbol{\theta}_\ell)} \left(\sum_{d,k} (\alpha_{dk} - 1 + n_{dk\cdot}) \log \pi_{dk} + \sum_{k,v} (\beta_{kv} - 1 + n_{\cdot kv}) \log \theta_{kv} \right) + \text{const.}$$

$$= \sum_{v=1}^{V} (\beta_{kv} - 1 + \mathbb{E}_{q(C)}(n_{\cdot kv})) \log \theta_{kv} + \text{const.}$$

Thus, $q(\boldsymbol{\theta}_k) = \mathcal{D}(\boldsymbol{\theta}_k; \tilde{\beta}_{kv} := [\beta_{kv} + \sum_d^D \sum_{i=1}^{I_d} \tilde{\gamma}_{dik} \mathbb{I}(w_{di} = v)]_{v=1,\dots,V})$.

Some Tabulated Identities

To be able to compute the required terms for the algorithm

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Properties of the Dirichlet

(let $\hat{\alpha} := \sum_d \alpha_d$)

$$p(x \mid \alpha) = \mathcal{D}(x; \alpha) = \frac{\Gamma(\hat{\alpha})}{\prod_d \Gamma(\alpha_d)} \prod_d x^{\alpha_d - 1} = \frac{1}{B(\alpha)} \prod_d x^{\alpha_d - 1}$$

- $\mathbb{E}_p(x_d) = \frac{\alpha_d}{\hat{\alpha}}$
- $\operatorname{var}_p(x_d) = \frac{\alpha_d(\hat{\alpha} - \alpha_d)}{\hat{\alpha}^2(\hat{\alpha}+1)}$
- $\operatorname{cov}(x_d, x_i) = -\frac{\alpha_d \alpha_i}{\hat{\alpha}^2(\hat{\alpha}+1)}$
- $\operatorname{mode}(x_d) = \frac{\alpha_d - 1}{\hat{\alpha} - D}$
- $\mathbb{E}_p(\log x_d) = \digamma(\alpha_d) - \digamma(\hat{\alpha})$
- $\mathbb{H}(p) = -\int p(x) \log p(x)\, dx = -\sum_d (\alpha_d - 1)(\digamma(\alpha_d) - \digamma(\hat{\alpha})) + \log B(\alpha)$

Where $\digamma(z) = \frac{d}{dz} \log \Gamma(z)$ (the “digamma-function”). `scipy.special.digamma(z)` https://dlmf.nist.gov/5

The Variational Approximation

closing the loop

$$
\begin{aligned}
q(\boldsymbol{\pi}_d) &= \mathcal{D}\left(\boldsymbol{\pi}_d; \tilde{\alpha}_{dk} := \left[\alpha_{dk} + \sum_{i=1}^{I_d} \tilde{\gamma}_{dik}\right]_{k=1,\dots,K}\right) \\
q(\boldsymbol{\theta}_k) &= \mathcal{D}\left(\boldsymbol{\theta}_k; \tilde{\beta}_{kv} := \left[\beta_{kv} + \sum_{d}^{D} \sum_{i=1}^{I_d} \tilde{\gamma}_{dik} \mathbb{I}(w_{di} = v)\right]_{v=1,\dots,V}\right) \\
q(\boldsymbol{c}_{di}) &= \prod_k \tilde{\gamma}_{dik}^{c_{dik}}, \quad \text{where} \quad \tilde{\gamma}_{dik} = \gamma_{dik} / \sum_k \gamma_{dik} \quad \text{and (note that } \sum_k \tilde{\alpha}_{dk} = \text{const.)} \\
\gamma_{dik} &= \exp\left(\mathbb{E}_{q(\pi_{dk})}(\log \pi_{dk}) + \mathbb{E}_{q(\theta_{di})}(\log \theta_{kw_{di}})\right) \\
&= \exp\left(F(\tilde{\alpha}_{jk}) + F(\tilde{\beta}_{kw_{di}}) - F\left(\sum_v \tilde{\beta}_{kv}\right)\right)
\end{aligned}
$$

The ELBO

useful for monitoring / bug-fixing

$$p(C,\Pi,\Theta,W) = \left(\prod_{d=1}^{D} \frac{\Gamma(\sum_k \alpha_{dk})}{\prod_k \Gamma(\alpha_{dk})} \prod_{k=1}^{K} \pi_{dk}^{\alpha_{dk}-1+n_{dk\cdot}}\right) \cdot \left(\prod_{k=1}^{K} \frac{\Gamma(\sum_v \beta_{kv})}{\prod_v \Gamma(\beta_{kv})} \prod_{v=1}^{V} \theta_{kv}^{\beta_{kv}-1+n_{\cdot kv}}\right)$$

We need

$$\begin{aligned}
\mathcal{L}(q,W) &= \mathbb{E}_q(\log p(W,C,\Theta,\Pi)) + \mathbb{H}(q) \\
&= \int q(C,\Theta,\Pi) \log p(W,C,\Theta,\Pi)\, dC\, d\Theta\, d\Pi - \int q(C,\Theta,\Pi) \log q(C,\Theta,\Pi)\, dC\, d\Theta\, d\Pi \\
&= \int q(C,\Theta,\Pi) \log p(W,C,\Theta,\Pi)\, dC\, d\Theta\, d\Pi + \sum_k \mathbb{H}(\mathcal{D}(\theta_k\ \tilde{\beta}_k)) + \sum_d \mathbb{H}(\mathcal{D}(\pi_d\ \tilde{\alpha}_d)) + \sum_{di} \mathbb{H}(\tilde{\gamma}_{di})
\end{aligned}$$

The entropies can be computed from the tabulated values. For the expectation, we use $\mathbb{E}_{q(C)}(n_{dkv}) = \sum_i \gamma_{dik}\mathbb{I}(w_{di} = v)$ and use $\mathbb{E}_{\mathcal{D}(\pi_d;\tilde{\alpha})}(\log \pi_d) = F(\tilde{\alpha}_d) - F(\hat{\tilde{\alpha}})$ from above.

Dirty secret: In practice, the ELBO itself isn't strictly necessary.

Building the Algorithm

updating and evaluating the bound

procedure LDA(W, α, β)
  $\tilde{\gamma}_{dik} \leftarrow$ DIRICHLET_RAND(α) // initialize
  $\mathcal{L} \leftarrow -\infty$
  while $\mathcal{L}$ not converged **do**
    for $d = 1, \dots, D; k = 1, \dots, K$ **do**
      $\tilde{\alpha}_{dk} \leftarrow \alpha_{dk} + \sum_i \tilde{\gamma}_{dik}$ // update document-topics distributions
    end for
    for $k = 1, \dots, K; v = 1, \dots, V$ **do**
      $\tilde{\beta}_{kv} \leftarrow \beta_{kv} + \sum_{d,i} \tilde{\gamma}_{dik} \mathbb{I}(w_{di} = v)$ // update topic-word distributions
    end for
    for $d = 1, \dots, D; k = 1, \dots, K; i = 1, \dots, I_d$ **do**
      $\tilde{\gamma}_{dik} \leftarrow \exp(\digamma(\tilde{\alpha}_{dk}) + \digamma(\tilde{\beta}_{kw_{di}}) - \digamma(\sum_v \tilde{\beta}_{kv}))$ // update word-topic assignments
      $\tilde{\gamma}_{dik} \leftarrow \tilde{\gamma}_{dik} / \tilde{\gamma}_{di\cdot}$
    end for
    $\mathcal{L} \leftarrow$ BOUND($\tilde{\gamma}, w, \tilde{\alpha}, \tilde{\beta}$) // update bound
  end while
end procedure

$$\mathcal{L}(q) = \int q(C, \Theta, \Pi) \log \left(\frac{p(C, \Pi, \Theta, W)}{q(C, \Theta, \Pi)} \right) dC \, d\Theta \, d\Pi$$

Variational Inference is a powerful mathematical tool to construct efficient approximations to intractable *probability distributions* (not just point estimates, but entire distributions). Often, just imposing factorization is enough to make things tractable. The downside of variational inference is that constructing the bound can take significant ELBOw grease. However, the resulting algorithms are often highly efficient compared to tools that require less derivation work, like Monte Carlo. Lectures 18–20.

"Derive your variational bound in the time it takes for your Monte Carlo sampler to converge."

Designing a Probabilistic Machine Learning Model

1. Take a close look at the **Data**
2. Think about modelling goals, decide on data types
3. Design the **Model**
4. Design the **Algorithm**
   - For conditionally independent parts, conjugate priors may give analytic inference
   - for linearly related variables, consider Gaussians
   - consider revisiting the model to simplify as much as possible
   - if there is still no analytic answer, use **The Toolbox**!