Machine Learning

Winter 2011/12

Roland Memisevic

Lecture 9, Jan. 9, 2012

Outline

- Laplace Approximation
- Variational inference
- Sampling

Approximate Inference

- The most common operation in probabilistic modeling is to compute (marginals of) the posterior $p(\mathbf{Z}|\mathbf{X})$ over unobserved variables $\mathbf{Z}$ given observed variables $\mathbf{X}$.
- We saw in Lecture 8 that this can be done efficiently, if the dependencies in the posterior take the form of a tree.
- We also saw that, when the graph is not a tree (or the maximal cliques are too large) one can use loopy Belief Propagation to perform inference.
- Alternatively, approximate inference methods simplify inference by approximating the posterior using a simpler, tractable distribution $q(\mathbf{Z})$.

Laplace Approximation

- One approach, which applies only to continuous distributions, is the Laplace Approximation:
- Consider (for now) the one-dimensional density

$$p(z) = \frac{1}{Z} f(z)$$

  where $Z = \int_z f(z) \mathrm{d}z$ may be unkown.
- The Laplace approximation of this density is a *Gaussian at its mode* z_0 (or at one of its modes, in case there are more):

$$q(z) = \left(\frac{A}{2\pi}\right)^{\frac{1}{2}} \exp\Big(-\frac{A}{2}(z-z_0)^2\Big) = \mathcal{N}\Big(z; z_0, A^{-1}\Big)$$

- How to set the variance, A^{-1}?

Laplace Approximation

- To define the variance, A^{-1}, we Taylor-expand $\log f(z)$ about z_0:

$$\log f(z) \approx \log f(z_0) + \frac{1}{2}\Big(\frac{\mathrm{d}^2}{\mathrm{d}z^2}\log f(z)\Big)(z-z_0)^2$$

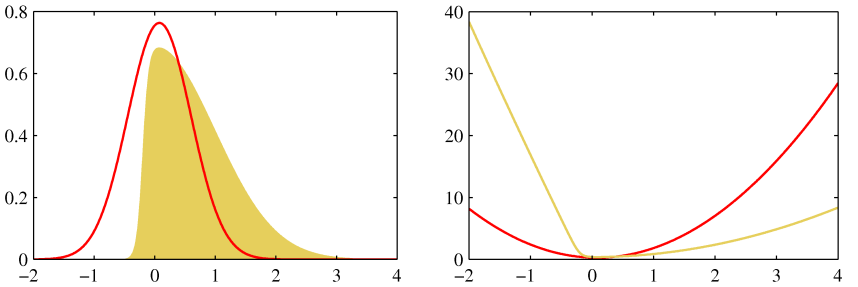
(Note that $\frac{\mathrm{d}}{\mathrm{d}z}f(z_0) = 0$)

- Exponentiating now allows us to write $f(z)$ in the form of an (unnormalized) Gaussian:

$$f(z) \approx f(z_0)\exp\Big(-\frac{-\frac{\mathrm{d}^2}{\mathrm{d}z^2}\log f(z)}{2}(z-z_0)^2\Big)$$

which suggests setting A to be the second derivative of $-\log f(z)$ at z_0.

Laplace Approximation Example



- Laplace Approximation: $p(z) \propto \exp(-\frac{z^2}{2})\sigma(20z+4)$
- Left: The density and its Laplace approximation, Right: The same in the log-domain.

Laplace Approximation in M dimensions

- For M-dimensional densities, the Taylor-expansion takes the form

$$\log f(\boldsymbol{z}) \approx \log f(\mathbf{z}_0) - \frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^{\mathrm{T}}\mathbf{A}(\mathbf{z}-\mathbf{z}_0)$$

where $\mathbf{A}$ is the Hessian of $-\log f(\mathbf{z})$ at $\mathbf{z}_0$.

- So we get the Laplace Approximation

$$\begin{aligned} q(\mathbf{z}) &= \frac{|\mathbf{A}|^{\frac{1}{2}}}{(2\pi)^{\frac{M}{2}}}\exp\Big(-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^{\mathrm{T}}\mathbf{A}(\mathbf{z}-\mathbf{z}_0)\Big) \\ &= \mathcal{N}\Big(\mathbf{z}|\mathbf{z}_0, \mathbf{A}^{-1}\Big) \end{aligned}$$

Variational Inference

- A different approach to performing approximate inference is *variational inference*:
- Like before, we pick a class of simpler distributions $q(\mathbf{Z})$, and find among this class the distribution $q^*(\mathbf{Z})$ that is as "close" as possible to the true $p(\mathbf{Z}|\mathbf{X})$.
- A common way to find the closest is by minimizing the KL-divergence:

$$q^*(\mathbf{Z}) = \arg\min_q \ \mathrm{KL}(q||p) = -\int q(\mathbf{Z})\log\Big(\frac{p(\mathbf{Z}|\mathbf{X})}{q(\mathbf{Z})}\Big)\mathrm{d}\mathbf{Z}$$

- (We could replace the integrals by sums here.)

Variational Inference

- This turns inference into an optimization problem.
- And solving that problem can be tractable in cases where computing $p(\mathbf{Z}|\mathbf{X})$ is not.
- A common choice for $q(\mathbf{Z})$ is the factorized distribution

$$q(\mathbf{Z}) = \prod_i q_i(z_i)$$

- This is also known as "mean-field" approximation.
- (One may also minimize $\mathrm{KL}(p||q)$ instead of $\mathrm{KL}(q||p)$, which is the basis for an approach known as "expectation propagation")

Variational EM

- In Lecture 5 we saw that we can decompose the log-likelihood of a latent variable model, $L = \log p(\mathbf{X}) = \log \int p(\mathbf{X}, \mathbf{Z}) \mathrm{d}\mathbf{Z}$ as

$$L = \mathcal{L}(q(\mathbf{Z})) + \mathrm{KL}(q(\mathbf{Z})||p(\mathbf{Z}|\mathbf{X}))$$

  where

$$\mathcal{L}(q(\mathbf{Z})) = \int q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} \mathrm{d}\mathbf{Z}$$

  and q is some auxiliary distribution over $\mathbf{Z}$.
- The EM-algorithm amounts to alternating between optimizing $q(\mathbf{Z})$ and optimizing model parameters.
- We also saw that optimizing $q(\mathbf{Z})$ amounts to setting it to $p(\mathbf{Z}|\mathbf{X})$.

Variational EM

- If $p(\mathbf{Z}|\mathbf{X})$ is not tractable, we can perform approximate EM, by restricting the search space to a tractable family of distributions $q(\mathbf{Z})$ (for example, mean-field).
- Restricting $q(\mathbf{Z})$ then amounts to increasing the lower bound on the log-likelihood in the E-step, without making it tight.
- This approach is commonly used in Bayesian models, where we have to infer not only distributions over latent variables but also over model parameters, which typically gives rise to intractable posteriors.

Sampling

- Sampling is a third way to perfom inference with intractable distributions.
- Sampling means drawing examples ("samples") from a distribution.
- Estimating quantities by resorting to sampling is also known as *Monte Carlo* approach.
- The most famous toy application of a Monte Carlo method is estimating π by uniformly sampling the unit square and counting the ratio of points that land in the unit circle vs. those that do not. (This will estimate $\frac{\pi}{4}$).
- In general, the most common use of sampling is to compute an expectation of some function $f(\mathbf{z})$ with respect to some distribution $p(\mathbf{z})$.

Sampling

- Consider the expectation of some arbitrary function $f(\mathbf{z})$ under the distribution $p(\mathbf{z})$:

$$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})\mathrm{d}\mathbf{z}$$

- If we have L iid. samples $\mathbf{z}^{(l)}$ from $p(\mathbf{z})$, we can approximate this expectation using

$$\hat{f} = \frac{1}{L}\sum_{l} f(\mathbf{z}^{(l)})$$

- Note that

$$\mathbb{E}[\hat{f}] = \mathbb{E}[f]$$

Sampling from simple densities

- There are many different sampling techniques, with different benefits and drawbacks.
- The most basic sampling method is *sampling by coordinate transform*:
- Assume that we have a function $\mathrm{rand}()$ that provides samples z from the uniform density in the interval $[0, 1]$.
- (This is a task for which fairly good solutions exist. Note that most programming environments provide a $\mathrm{rand}()$-function for sampling from the uniform density.)
- Sampling from a different density $p(y)$, by transforming the uniform density, then works as follows:

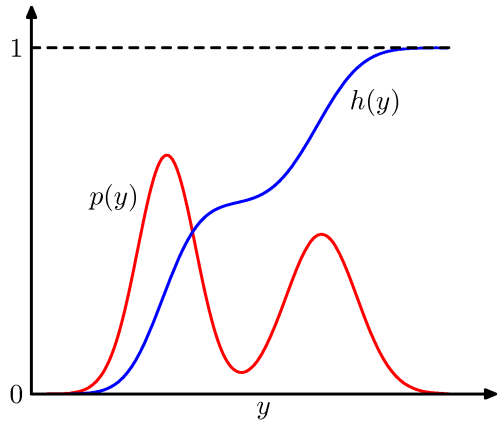
Sampling from simple densities

Drawing from a density $p(y)$ by coordinate transform

1. Find the cumulative density $h(y) = \int_{-\infty}^{y} p(\hat{y})\mathrm{d}\hat{y}$.
2. Draw samples z from the uniform density on $[0, 1]$ (for example, using $\mathrm{rand}()$).
3. Get samples y from $p(y)$ by computing

$$y = h^{-1}(z)$$

Sampling by coordinate transform, intuition



Sampling by coordinate transform, comments

- One can generalize this idea to multivariate densities by using conditional cumulatives.
- In general, sampling by coordinate transform is applicable to only a restricted set of densities: those, for which we are able to compute the inverse of the cumulative density.
- Two common, more generally applicable methods, are *rejection sampling* and *importance sampling*.

Rejection Sampling

- Assumption: We can *evaluate* the density $p(\mathbf{z})$.
- Remark: For rejection sampling, it is sufficient to be able to evaluate $p(\mathbf{z})$ up to a normalizing constant, so we may write:

$$p(\mathbf{z}) = \frac{1}{Z_p}\tilde{p}(\mathbf{z})$$

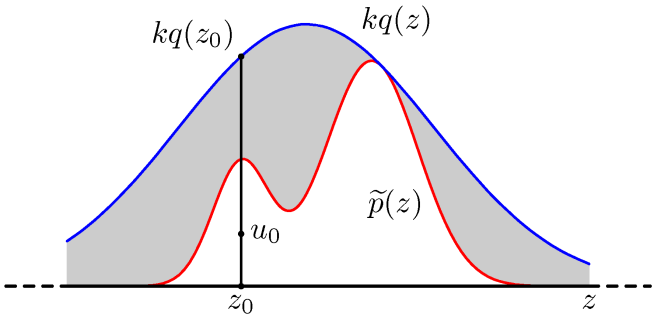
  where Z_p may be unknown.
- Rejection sampling requires two additional ingredients:
  1. Another distribution, $q(\mathbf{z})$, known as *proposal distribution*, from which we *can* sample.
  2. A constant k, such that $kq(\mathbf{z}) > \tilde{p}(\mathbf{z}) \;\; \forall \mathbf{z}$.
- Using these, rejection sampling works as follows:

Rejection Sampling Algorithm

Rejection Sampling

- Generate pairs $(\mathbf{z}_0, \mathbf{u}_0)$, where
  - $\mathbf{z}_0$ is sampled from $q(\mathbf{z})$ and
  - $\mathbf{u}_0$ is sampled from the uniform distribution in the interval $[0, kq(\mathbf{z}_0)]$
- Keep all those $\mathbf{z}_0$ for which $\mathbf{u}_0 < \tilde{p}(\mathbf{z}_0)$, and discard ("reject") the rest.

Rejection Sampling



- The remaining pairs will be distributed uniformly in the white area, thus the $\mathbf{z}$-components will be distributed according to $p(\mathbf{z})$.

Rejection Sampling Comments

- Rejection sampling will be the most efficient for proposal distributions that match $p(\mathbf{z})$ well, because these will incur fewer rejections.
- Because this may be hard to achieve in practice, rejection sampling can be inefficient.
- This is true in particular in high-dimensional spaces.

Importance Sampling

- Importance sampling is an approach to evaluating *expectations* wrt. $p(\mathbf{z})$ using sampling, rather than providing samples themselves.
- Again, we assume there is a distributions $q(\mathbf{z})$ from which we can sample, and we assume that we can *evaluate* $p(\mathbf{z})$.
- We can rewrite the expectation of some function f with respect to $p(\mathbf{z})$ as follows:

$$\begin{aligned}\mathbb{E}[f] &= \int f(\mathbf{z})p(\mathbf{z})\,\mathrm{d}\mathbf{z} \\ &= \int f(\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z})}q(\mathbf{z})\,\mathrm{d}\mathbf{z} \\ &\approx \frac{1}{L}\sum_{l=1}^{L}\frac{p(\mathbf{z}^{(l)})}{q(\mathbf{z}^{(l)})}f(\mathbf{z}^{(l)})\end{aligned}$$

Importance Sampling

- Thus, to compute the expectation with respect ot $p(\mathbf{z})$ we can sample from $q(\mathbf{z})$ instead, and then weight the function values using the so-called "*importance weights*"

$$r_l = \frac{p(\mathbf{z}^{(l)})}{q(\mathbf{z}^{(l)})}$$

- Importance Sampling can be easily extended to the case where we can evaluate $p()$ and $q()$ only up to normalizing constants.
- As with rejection sampling, importance sampling may not work well, if $q(\mathbf{z})$ does not match $p(\mathbf{z})$ very well.
- The problem here is slightly different: We may "miss" modes of $p(\mathbf{z})$ and get wrong results.

Markov Chain Monte Carlo

- A class of sampling methods that often work better, in particular, in higher dimensions, are *Markov Chain Monte Carlo* (MCMC)-methods.
- Idea: Construct a Markov chain in "$\mathbf{z}$"−space which, in the long term, will take on states $\mathbf{z}$ according to $p(\mathbf{z})$.
- Recall that a Markov chain is a distribution $p(\mathbf{z}_1, \ldots, \mathbf{z}_K)$ where $p(\mathbf{z}_i|\mathbf{z}_1, \ldots, \mathbf{z}_{i-1}) = p(\mathbf{z}_i|\mathbf{z}_{i-1})$
- MCMC methods will typically *not* generate iid samples, but samples that are highly correlated.
- One way then to get close-to-uncorrelated samples is to keep only every M^{th} sample.

Examples of Markov Chains that sample from rand

Three simple, degenerate ways of using a Markov chain to sample from the uniform distribution on $[0, 1]$:

1. Define z_i using $\mathrm{rand}()$
2. Define $z_0 = 0$ and $z_{i+1} = \mathrm{mod}(z_i + \sqrt{2}, 1)$
3. Define $z_0 = 0$ and $z_{i+1} = \mathrm{mod}(z_i + \frac{1}{100}\mathrm{rand}(), 1)$

Evaluating marginals

- To define a Markov chain for a specific distribution $p(\mathbf{z})$, one typically makes use of the following properties of (*homogeneous*) Markov chains:
- A homogeneous Markov chain is given by the **start-probabilities** $p(\mathbf{z}_0)$ and the **transition probabilities** $T(\mathbf{z}^{(m)}, \mathbf{z}^{(m+1)})$.
- The **marginal probability** at time step $m + 1$ may be written

$$p(\mathbf{z}^{(m+1)}) = \sum_{\mathbf{z}^{(m)}} p(\mathbf{z}^{(m+1)}|\mathbf{z}^{(m)}) p(\mathbf{z}^{(m)})$$

$$\left(= \sum_{\mathbf{z}^{(m)}} T(\mathbf{z}^{(m)}, \mathbf{z}^{(m+1)}) p(\mathbf{z}^{(m)}) \right)$$

Invariant distribution

- A distribution $p^*(\mathbf{z})$ is known as an **invariant distribution** with respect to a homogeneous Markov chain with transition probabilities $T(\mathbf{z}', \mathbf{z})$, if

$$p^*(\mathbf{z}) = \sum_{\mathbf{z}'} T(\mathbf{z}', \mathbf{z}) p(\mathbf{z}')$$

- A Markov chain may have several invariant distributions.
- When thinking about invariant distributions, it may help build intuition to imagine running many copies of the same chain.

Detailed balance

- A sufficient (not necessary) condition for $p^*(\mathbf{z})$ to be an invariant distribution with respect to a Markov chain with transition probabilities $T(\mathbf{z}', \mathbf{z})$ is that

$$p^*(\mathbf{z}) T(\mathbf{z}, \mathbf{z}') = p^*(\mathbf{z}') T(\mathbf{z}', \mathbf{z})$$

- This condition is known as **detailed balance** and a Markov chain that respects it is called *reversible*.
- Invariance of $p^*(\mathbf{z}')$ follows immediately:

$$\sum_{\mathbf{z}'} p^*(\mathbf{z}') T(\mathbf{z}', \mathbf{z}) = \sum_{\mathbf{z}'} p^*(\mathbf{z}) T(\mathbf{z}, \mathbf{z}') = p^*(\mathbf{z}) \sum_{\mathbf{z}'} T(\mathbf{z}, \mathbf{z}') = p^*(\mathbf{z})$$

- To construct a Markov chain that samples from some desired distribution, we can set up the chain such that the desired distribution is an invariant distribution.

Ergodicity

- A further condition that the chain needs to meet for it to sample from the desired distribution is that it also *converges* to the invariant distribution as $m \to \infty$
- This property is known as **ergodicity**, and a chain that satisfies it is known as **ergodic**.
- An ergodic Markov chain can have only one invariant distribution, which is called **equilibrium distribution**.
- A requirement for ergodicity is that any point can be reached from any other point in a finite number of steps.

Constructing Markov chains

- From these considerations, it follows that we can draw samples from a distribution by constructing an ergodic Markov chain that satisfies detailed balance with respect to this distribution.
- The two main techniques for doing this are
  - **Gibbs Sampling** and
  - the **Metropolis-Hastings algorithm**
- (Gibbs sampling may be viewed as a special case of the Metropolis-Hastings algorithm.)

Gibbs Sampling

- We wish to sample from $p(\mathbf{z}) = p(z_1, \ldots, z_M)$
- Gibbs sampling works under the assumption that we *can* sample from each conditional over z_i, given all the other variables.
- It amounts to cycling through all variables (randomly or in some specific order) and updating $\mathbf{z}$ by sampling the conditionals:

Gibbs Sampling

Gibbs Sampling

- Initialize all z_i
- For $\tau = 1, \ldots, T$:
  - Sample $z_1^{(\tau+1)}$ from $p(z_1 | z_2^{(\tau)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$
  - Sample $z_2^{(\tau+1)}$ from $p(z_2 | z_1^{(\tau+1)}, z_3^{(\tau)}, \ldots, z_M^{(\tau)})$
  - $\vdots$
  - Sample $z_j^{(\tau+1)}$ from $p(z_j | z_1^{(\tau+1)}, \ldots, z_{j-1}^{(\tau+1)}, z_{j+1}^{(\tau)}, \ldots, z_M^{(\tau)})$
  - Sample $z_M^{(\tau+1)}$ from $p(z_M | z_1^{(\tau+1)}, z_2^{(\tau+1)}, \ldots, z_{M-1}^{(\tau+1)})$

Gibbs Sampling comments

- It is intuitively clear that $p(\mathbf{z})$ is an invariant distribution:
- In each step, we sample from the correct conditional, and the marginals over the variables we condition on stays unchanged.
- One can also show that Gibbs sampling satisfies detailed balance.
- Similarly for ergodicity, *given* that all conditional distributions are non-zero everywhere.
- So the Markov chain will converge to producing samples from $p(\mathbf{z})$. The samples will be highly correlated, so to get iid samples one needs to discard many of the samples.

Metropolis-Hastings

- The Metropolis-Hastings algorithm, like rejection and importance sampling, makes use of a proposal distribution.
- In contrast to rejection and importance sampling, here the proposal distribution is a conditional distribution $q(\mathbf{z}^{(\tau)}|\mathbf{z}^{(\tau-1)})$, so it defines a Markov chain.
- The proposal distribution is used in combination with an acceptance probability which ensures that detailed balance wrt. to the desired distribution holds.