Chapter 14

Monte Carlo Methods

TODO plan organization of chapter (spun off from graphical models chapter)

14.1 Markov Chain Monte Carlo Methods

Drawing a sample x from the probability distribution $p(\mathrm{x})$ defined by a structured model is an important operation. The following techniques are described in (Koller and Friedman, 2009).

Sampling from an energy-based model is not straightforward. Suppose we have an EBM defining a distribution $p(\mathrm{a}, \mathrm{b})$. In order to sample a, we must draw it from $p(\mathrm{a} \mid \mathrm{b})$, and in order to sample b, we must draw it from $p(\mathrm{b} \mid \mathrm{a})$. It seems to be an intractable chicken-and-egg problem. Directed models avoid this because their $\mathcal{G}$ is directed and acyclical. In *ancestral sampling* one simply samples each of the variables in topological order, conditioning on each variable's parents, which are guaranteed to have already been sampled. This defines an efficient, single-pass method of obtaining a sample.

In an EBM, it turns out that we can get around this chicken and egg problem by sampling using a *Markov chain*. A Markov chain is defined by a *state* $\mathbf{x}$ and a transition distribution $T(\mathbf{x}' \mid \mathbf{x})$. Running the Markov chain means repeatedly updating the state $\mathbf{x}$ to a value $\mathbf{x}'$ sampled from $T(\mathbf{x}' \mid \mathbf{x})$.

Under certain distributions, a Markov chain is eventually guaranteed to draw $\mathbf{x}$ from an equilibrium distribution $\pi(\mathbf{x}')$, defined by the condition

$$\forall \boldsymbol{x}', \pi(\boldsymbol{x}') = \sum_{\boldsymbol{x}} T(rvx' \mid \boldsymbol{x})\pi(\boldsymbol{x}).$$

TODO– this vector / matrix view needs a whole lot more exposition only literally a vector / matrix when the state is discrete unpack into multiple sentences,

the parenthetical is hard to parse is the term "stochastic matrix" defined anywhere? make sure it's in the index at least whoever finishes writing this section should also finish making the math notation consistent terms in this section need to be in the index

We can think of π as a vector (with the probability for each possible value $\boldsymbol{x}$ in the element indexed by x, $\pi(x)$) and T as a corresponding stochastic matrix (with row index x' and column index x), i.e., with non-negative entries that sum to 1 over elements of a column. Then, the above equation becomes

$$T\pi = \pi$$

an eigenvector equation that says that π is the eigenvector of T with eigenvalue 1. It can be shown (Perron-Frobenius theorem) that this is the largest possible eigenvalue, and the only one with value 1 under mild conditions (for example $T(x' \mid x) > 0$). We can also see this equation as a fixed point equation for the update of the distribution associated with each step of the Markov chain. If we start a chain by picking $\mathrm{x}_0 \sim p_0$, then we get a distribution $p_1 = Tp_0$ after one step, and $p_t = Tp_{t-1} = T^t p_0$ after t steps. If this recursion converges (the chain has a so-called *stationary distribution*), then it converges to a fixed point which is precisely $p_t = \pi$ for $t \to \infty$, and the dynamical systems view meets and agrees with the eigenvector view.

This condition guarantees that repeated applications of the transition sampling procedure don't change the *distribution* over the state of the Markov chain. Running the Markov chain until it reaches its equilibrium distribution is called "burning in" the Markov chain.

Unfortunately, there is no theory to predict how many steps the Markov chain must run before reaching its equilibrium distribution[1], nor any way to tell for sure that this event has happened. Also, even though successive samples come from the same distribution, they are highly correlated with each other, so to obtain multiple samples one should run the Markov chain for many steps between collecting each sample. Markov chains tend to get stuck in a single mode of $\pi(x)$ for several steps. The speed with which a Markov chain moves from mode to mode is called its mixing rate. Since burning in a Markov chain and getting it to mix well may take several sampling steps, sampling correctly from an EBM is still a somewhat costly procedure.

TODO: mention Metropolis-Hastings

Of course, all of this depends on ensuring $\pi(x) = p(x)$. Fortunately, this is easy so long as $p(x)$ is defined by an EBM. The simplest method is to use *Gibbs sampling*, in which sampling from $T(\mathbf{x}' \mid \mathbf{x})$ is accomplished by selecting

[1] although in principle the ratio of the two leading eigenvalues of the transition operator gives us some clue, and the largest eigenvalue is 1.

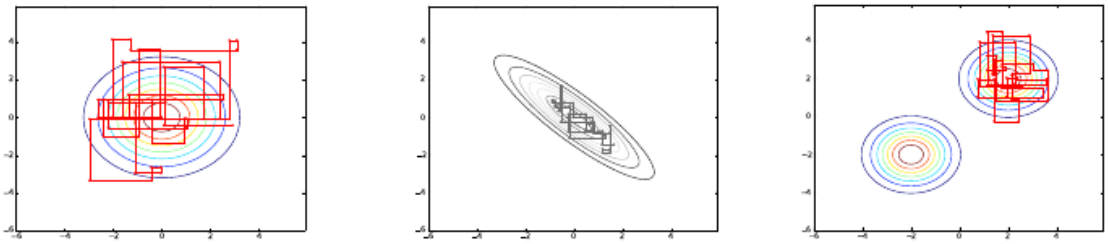

Figure 14.1: Paths followed by Gibbs sampling for three distributions, with the Markov chain initialized at the mode in both cases. Left) A multivariate normal distribution with two independent variables. Gibbs sampling *mixes* well because the variables are independent. Center) A multivariate normal distribution with highly correlated variables. The correlation between variables makes it difficult for the Markov chain to mix. Because each variable must be updated conditioned on the other, the correlation reduces the rate at which the Markov chain can move away from the starting point. Right) A mixture of Gaussians with widely separated modes that are not axis-aligned. Gibbs sampling mixes very slowly because it is difficult to change modes while altering only one variable at a time.

one variable x_i and sampling it from p conditioned on its neighbors in $\mathcal{G}$. It is also possible to sample several variables at the same time so long as they are conditionally independent given all of their neighbors.

TODO: discussion of mixing example with 2 binary variables that prefer to both have the same state IG's graphic from lecture on adversarial nets

TODO: refer to this figure in the text:

TODO: refer to this figure in the text

14.1.1 Markov Chain Theory

TODO

State Perron's theorem

DEFINE detailed balance

14.1.2 Importance Sampling

TODO write this section

14.2 The Difficulty of Mixing between Well-Separated Modes

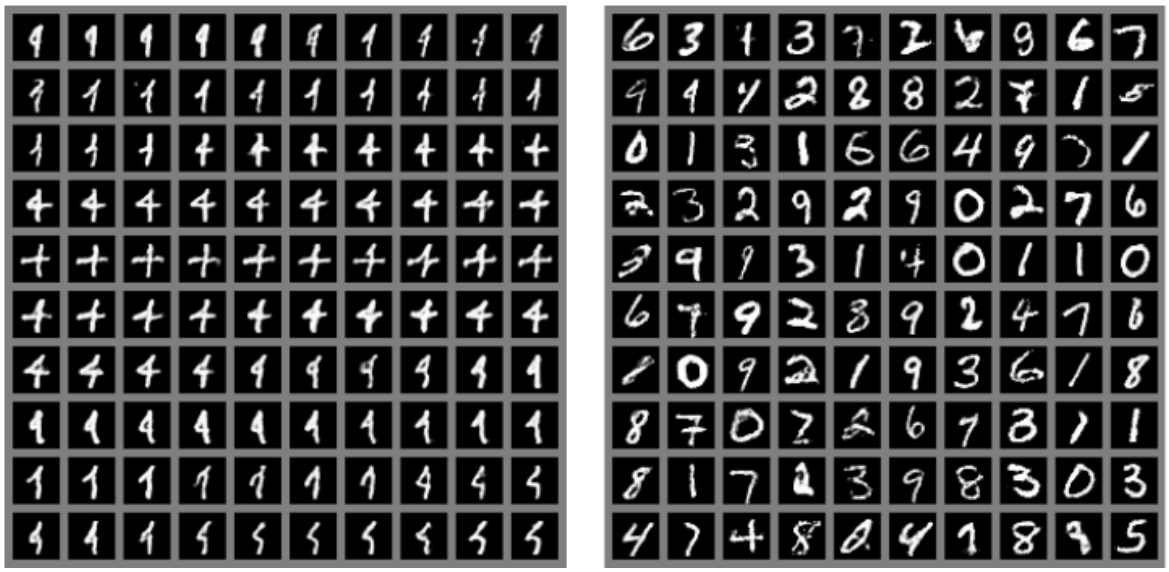

Figure 14.2: An illustration of the slow mixing problem in deep probabilistic models. Each panel should be read left to right, top to bottom. Left) Consecutive samples from Gibbs sampling applied to a deep Boltzmann machine trained on the MNIST dataset. Consecutive samples are similar to each other. Because the Gibbs sampling is performed in a deep graphical model, this similarity is based more on semantic rather than raw visual features, but it is still difficult for the Gibbs chain to transition from one mode of the distribution to another, for example by changing the digit identity. Right) Consecutive ancestral samples from a generative adversarial network. Because ancestral sampling generates each sample independently from the others, there is no mixing problem.