Algorithms other than SGD

CS6787 Lecture 10 — Fall 2017

Machine learning is not just SGD

- Once a model is trained, we need to use it to classify new examples
  - This **inference task** is not computed with SGD

- There are other algorithms for optimizing objectives besides SGD
  - **Stochastic coordinate descent**
  - **Derivative-free optimization**

- There are other common tasks, such as sampling from a distribution
  - **Gibbs sampling** and other Markov chain Monte Carlo methods
  - And we sometimes use this together with SGD → called **contrastive divergence**

Why understand these algorithms?

- They represent a significant fraction of machine learning computations
  - **Inference** in particular is huge
- You may want to use them **instead of SGD**
  - But you don't want to suddenly pay a computational penalty for doing so because you don't know how to make them fast
- **Intuition from SGD** can be used to make these algorithms faster too
  - And vice-versa

Inference

Inference

- Suppose that our training loss function looks like

$$f(w) = \frac{1}{N} \sum_{i=1}^{n} l(\hat{y}(w; x_i), y_i)$$

- Inference is the problem of computing the prediction

$$\hat{y}(w; x_i)$$

How important is inference?

- **Train once, infer many times**
  - Many production machine learning systems just do inference
- Image recognition, voice recognition, translation
  - All are just applications of inference once they're trained
- Need to get **responses to users quickly**
  - On the web, users won't wait more than a second

Inference on linear models

- Computational cost: relatively **low**
  - Just a matrix-vector multiply
- But still can be more costly in some settings
  - For example, if we need to compute a random kernel feature map
  - **What is the cost of this?**
- **Which methods can we use to speed up inference in this setting?**

Inference on neural networks

- Computational cost: **relatively high**
  - Several matrix-vector multiplies and non-linear elements
- **Which methods can we use to speed up inference in this setting?**
- **Compression**
  - Find an easier-to-compute network with similar accuracy by fine-tuning
  - The subject of this week's paper

Other techniques for speeding up inference

- Train a fast model, and run it most of the time
  - If it's **uncertain**, then run a more accurate, slower model

- For video and time-series data, **re-use some of the computation** from previous frames
  - For example, only update some of the activations in the network at each frame
  - Or have a more-heavyweight network run less frequently
  - Rests on the notion that the **objects in the scene do not change frequently** in most video streams

Other Techniques for Training, Besides SGD

Coordinate Descent

- Start with objective

$$\text{minimize:} f(x_1, x_2, \ldots, x_n)$$

- Choose a random index **i**, and update

$$x_i = \arg\min_{\hat{x}_i} f(x_1, x_2, \ldots, x_i, \ldots, x_n)$$

- And repeat in a loop

Variants

- Coordinate descent with derivative and step size

- Stochastic coordinate descent

- **How do these compare to SGD?**

Derivative Free Optimization (DFO)

- Optimization methods that don't require differentiation
- Basic coordinate descent is actually an example of this
- Another example: for normally distributed ε

$$x_{t+1} = x_t - \alpha \frac{f(x_t + \sigma\epsilon) - f(x_t - \sigma\epsilon)}{2\sigma} \epsilon$$

- **Applications?**

Another Task: Sampling

Focus problem for this setting:

Statistical Inference

- Major class of machine learning applications
  - Goal: **draw conclusions from data** using a statistical model
  - Formally: find marginal distribution of unobserved variables given observations
- Example: decide whether a coin is biased from a series of flips
- Applications: LDA, recommender systems, text extraction, etc.
- De facto algorithm used for inference at scale: **Gibbs sampling**

Graphical models

- A graphical way to describe a probability distribution

- Common in machine learning applications
  - Especially for applications that deal with uncertainty

What types of inference exist here?

- Maximum-a-posteriori (MAP) inference
  - Find the state with the highest probability
  - Often reduces to an optimization problem
  - **What is the most likely state of the world?**

- Marginal inference
  - Compute the marginal distributions of some variables
  - **What does our model of the world tell us about this object or event?**

What is Gibbs Sampling?

Algorithm 1 Gibbs sampling

Require: Variables x_i for $1 \le$ … π.

l…

…pling uniformly … n $\{1, \ldots, n\}$.

Re…mple x_s uniformly from $\mathbf{P}_\pi(x_s | x_{\{1,\ldots,n\}\setminus\{s\}})$.

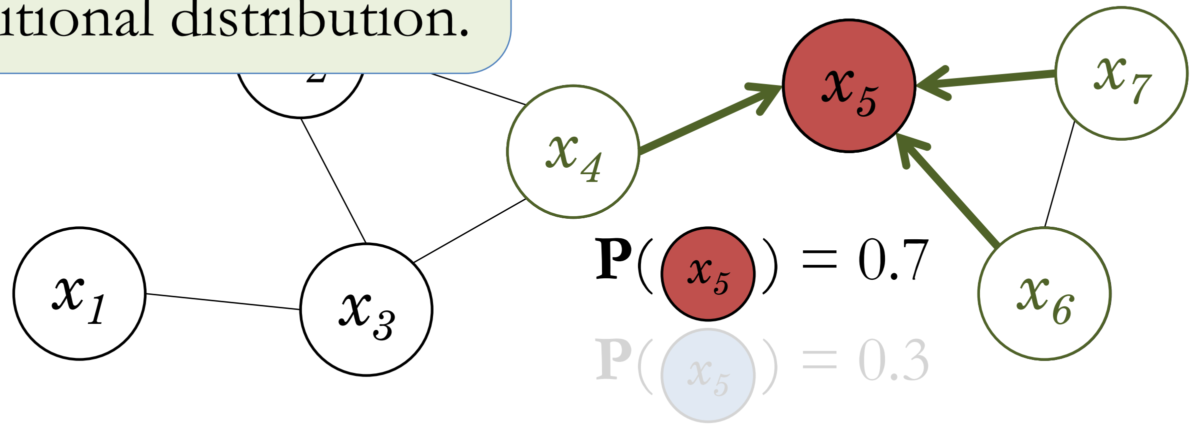
out… x

e…

Output the current state as a sample.

Compute its conditional distribution given the other variables.

Update the variable by sampling from its conditional distribution.



Learning graphical models

- Contrastive divergence
  - SGD on top of Gibbs sampling

- The de facto way of training
  - Restricted boltzmann machines (RBM)
  - Deep belief networks (DBN)
  - Knowledge-base construction (KBC) applications

What do all these algorithms look like?

Stochastic Iterative Algorithms

Given an immutable input dataset and a model we want to output.

Repeat:

1. **Pick a data point at random**
2. **Update the model**
3. **Iterate**

same structure

↓

same systems properties

↓

same techniques

Questions?

- Upcoming things
  - Paper Review #9 — **due today**
  - Paper Presentation #10 **on Wednesday**
  - **Project proposals due the following Monday**