Training Products of Experts by Minimizing Contrastive Divergence

Geoffrey E. Hinton

presented by Frank Wood
Goal

- Learn parameters for probability distribution models of high dimensional data
  - (Images, Population Firing Rates, Securities Data, NLP data, etc)

Mixture Model

\[ p(\tilde{d} | \theta_1, \ldots, \theta_n) = \sum_m \alpha_m f_m(\tilde{d} | \theta_m) \]

Use EM to learn parameters

Product of Experts

\[ p(\tilde{d} | \theta_1, \ldots, \theta_n) = \frac{\prod_m f_m(\tilde{d} | \theta_m)}{\sum_{\tilde{c}} \prod_m f_m(\tilde{c} | \theta_m)} \]

Use Contrastive Divergence to learn parameters.
Take Home

• Contrastive divergence is a general MCMC gradient ascent learning algorithm particularly well suited to learning Product of Experts (PoE) and energy-based (Gibbs distributions, etc.) model parameters.

• The general algorithm:
  - Repeat Until “Convergence”
    • Draw samples from the current model starting from the training data.
    • Compute the expected gradient of the log probability w.r.t. all model parameters over both samples and the training data.
    • Update the model parameters according to the gradient.
Sampling - Critical to Understanding

- **Uniform**
  - \( \text{rand()} \)
    - Linear Congruential Generator
      - \( x(n) = a \times x(n-1) + b \mod M \)
      - 0.2311  0.6068  0.4860  0.8913  0.7621  0.4565  0.0185

- **Normal**
  - \( \text{randn()} \)
    - Box-Mueller
      - \( x_1, x_2 \sim \text{U}(0,1) \rightarrow y_1, y_2 \sim \text{N}(0,1) \)
        - \( y_1 = \sqrt{-2 \ln(x_1)} \cos(2 \pi x_2) \)
        - \( y_2 = \sqrt{-2 \ln(x_1)} \sin(2 \pi x_2) \)

- **Binomial(p)**
  - if(rand()<p)

- **More Complicated Distributions**
  - Mixture Model
    - Sample from a Gaussian
    - Sample from a multinomial (CDF + uniform)
  - Product of Experts
    - Metropolis and/or Gibbs
The Flavor of Metropolis Sampling

- Given some distribution \( p(d | \theta) \), a random starting point \( d_{t-1} \), and a symmetric proposal distribution \( J(d_t | d_{t-1}) \).
- Calculate the ratio of densities where \( d_t \) is sampled from the proposal distribution.
- With probability \( \min(r, 1) \) accept \( d_t \).
- Given sufficiently many iterations

\[
\{d_n, d_{n+1}, d_{n+2}, \ldots\} \sim p(d | \theta)
\]

Only need to know the distribution up to a proportionality!
Contrastive Divergence (Final Result!)

\[ \Delta \theta_m \propto \langle \frac{\partial \log f_{\theta_m}}{\partial \theta_m} \rangle_{p^0} - \langle \frac{\partial \log f_{\theta_m}}{\partial \theta_m} \rangle_{p^1} \]

- Law of Large Numbers, compute expectations using samples.

**Model parameters.**

**Training data (empirical distribution).**

**Samples from model.**

Now you know how to do it, let's see why this works!
But First: The last vestige of concreteness.

- Looking towards the future:
  - Take $f$ to be a Student-t.

  $$f_{\theta_m}(\vec{d}) = f_{\alpha_m;j_m}(\vec{d}) = \frac{1}{\left(1 + \frac{1}{2}(j_m^T \vec{d})\right)^{\alpha_m}}$$

  - Then (for instance)

  $$\frac{\partial \log f_{\alpha_m;j_m}(\vec{d})}{\partial \alpha_m} = -\frac{\partial \alpha_m \log \left(1 + \frac{1}{2}(j_m^T \vec{d})\right)}{\partial \alpha_m} = -\log \left(1 + \frac{1}{2}(j_m^T \vec{d})\right)$$
Maximizing the training data log likelihood

- We want maximizing parameters
  \[
  \arg\max_{\theta_1, \ldots, \theta_n} \log p(D | \theta_1, \ldots, \theta_n) = \arg\max_{\theta_1, \ldots, \theta_n} \log \prod_{d \in D} \frac{\prod_m f_m(d | \theta_m)}{\sum_{c} \prod_m f_m(c | \theta_m)}
  \]

  Assuming \(d\)'s drawn independently from \(p()\)

- Differentiate w.r.t. to all parameters and perform gradient ascent to find optimal parameters.

- The derivation is somewhat nasty.
Maximizing the training data log likelihood

\[
\frac{\partial \log p(D | \theta_1, \ldots, \theta_n)}{\partial \theta_m} = \frac{\partial \log \prod_{d \in D} p(\tilde{d} | \theta_1, \ldots, \theta_n)}{\partial \theta_m}
\]

\[
= \sum_{d \in D} \frac{\partial}{\partial \theta_m} \log p(\tilde{d} | \theta_1, \ldots, \theta_n)
\]

\[
= N \left\langle \frac{\partial \log p(\tilde{d} | \theta_1, \ldots, \theta_n)}{\partial \theta_m} \right\rangle_{p_\theta^\infty}
\]

Remember this equivalence!
Maximizing the training data log likelihood

\[
\frac{1}{N} \sum_{d \in D} \frac{\partial \log f_m(d \mid \theta_m)}{\partial \theta_m} = \frac{1}{N} \sum_{d \in D} \frac{\partial \log \sum_{\tilde{c}} \prod_{m} f_m(\tilde{c} \mid \theta_m)}{\partial \theta_m} - \frac{1}{N} \sum_{d \in D} \frac{\partial \log \prod_{m} f_m(d \mid \theta_m)}{\partial \theta_m}
\]
Maximizing the training data log likelihood

\[
\begin{align*}
&= \frac{1}{N} \sum_{d \in D} \frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m} - \frac{\partial \log \sum \prod f_m(\tilde{c} | \theta_m)}{\partial \theta_m} \\
&= \left\langle \frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m} \right\rangle p^0 - \frac{\partial \log \sum \prod f_m(\tilde{c} | \theta_m)}{\partial \theta_m} \\
&= \left\langle \frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m} \right\rangle p^0 - \frac{1}{\sum \prod f_m(\tilde{c} | \theta_m)} \frac{\partial \sum \prod f_m(\tilde{c} | \theta_m)}{\partial \theta_m}
\end{align*}
\]

\[\log(x)' = x'/x\]
Maximizing the training data log likelihood

\[
\frac{\partial}{\partial \theta_m} \log f_m \left( \bar{d} \mid \theta_m \right) = \frac{\partial}{\partial \theta_m} \sum_{\tilde{c}} \prod_{m} f_m \left( \tilde{c} \mid \theta_m \right) - \sum_{\tilde{c}} \prod_{m} f_m \left( \tilde{c} \mid \theta_m \right) \frac{\partial}{\partial \theta_m} \frac{\sum_{j \neq m} f_j \left( \tilde{c} \mid \theta_j \right) f_m \left( \tilde{c} \mid \theta_m \right)}{\partial \theta_m}
\]

\[
\frac{\partial}{\partial \theta_m} \log f_m \left( \bar{d} \mid \theta_m \right) = \frac{\partial}{\partial \theta_m} \sum_{\tilde{c}} \prod_{m} f_m \left( \tilde{c} \mid \theta_m \right) \log f_m \left( \tilde{c} \mid \theta_m \right)
\]
Maximizing the training data log likelihood

\[
\begin{align*}
\mathbb{E}_{\pi^0}\left[\frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m}\right] & = -\sum_{\tilde{c}} \prod_m f_m(\tilde{c} | \theta_m) \sum_{\tilde{c}} \prod_m f_m(\tilde{c} | \theta_m) \frac{\partial \log f_m(\tilde{c} | \theta_m)}{\partial \theta_m} \\
= \mathbb{E}_{\pi^0}\left[\frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m}\right] & - \sum_{\tilde{c}} \left( \frac{\prod_m f_m(\tilde{c} | \theta_m)}{\sum_{\tilde{c}} \prod_m f_m(\tilde{c} | \theta_m)} \frac{\partial \log f_m(\tilde{c} | \theta_m)}{\partial \theta_m} \right) \\
= \mathbb{E}_{\pi^0}\left[\frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m}\right] & - \sum_{\tilde{c}} p(\tilde{c} | \theta_1, \ldots, \theta_n) \frac{\partial \log f_m(\tilde{c} | \theta_m)}{\partial \theta_m} \end{align*}
\]
Maximizing the training data log likelihood

\[
\mathbb{E}_{p^0} \left( \frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m} \right) - \sum_c p(c | \theta_1, \ldots, \theta_n) \frac{\partial \log f_m(c | \theta_m)}{\partial \theta_m} = \int \mathbb{E}_{p^0} \left( \frac{\partial \log f_m(\tilde{d} | \theta_m)}{\partial \theta_m} \right) - \mathbb{E}_{p^\infty} \left( \frac{\partial \log f_m(c | \theta_m)}{\partial \theta_m} \right)
\]

Phew! We’re done! So:

\[
\frac{\partial \log p(D | \theta_1, \ldots, \theta_n)}{\partial \theta_m} \propto \mathbb{E}_{p^0} \left( \frac{\partial \log p^\infty(\tilde{d} | \theta_m)}{\partial \theta_m} \right)
\]
Equilibrium Is Hard to Achieve

• With:

$$\frac{\partial \log p(D | \theta_1, \ldots, \theta_n)}{\partial \theta_m} \propto \left\langle \frac{\partial \log f_m(d | \theta_m)}{\partial \theta_m} \right\rangle_{P^0} - \left\langle \frac{\partial \log f_m(c | \theta_m)}{\partial \theta_m} \right\rangle_{P_\theta^\infty}$$

we can now train our PoE model.

• But... there’s a problem:
  - $P_\theta^\infty$ is computationally infeasible to obtain (esp. in an inner gradient ascent loop).
  - Sampling Markov Chain must converge to target distribution. Often this takes a very long time!
Solution: Contrastive Divergence!

\[ \frac{\partial \log p(D | \theta_1, \ldots, \theta_n)}{\partial \theta_m} \propto \left( \frac{\partial \log f_m(d | \theta_m)}{\partial \theta_m} \right)_{p^0} - \left( \frac{\partial \log f_m(c | \theta_m)}{\partial \theta_m} \right)_{p_\theta} \]

- Now we don’t have to run the sampling Markov Chain to convergence, instead we can stop after 1 iteration (or perhaps a few iterations more typically)

- Why does this work?
  - Attempts to minimize the ways that the model distorts the data.
Equivalence of argmax log \( P() \) and argmax KL()

\[
P^0\|_{P_\theta^\infty} = \sum_{\tilde{d}} P^0(\tilde{d}) \log \frac{P^0(\tilde{d})}{P_\theta^\infty(\tilde{d})}
= \sum_{\tilde{d}} P^0(\tilde{d}) \log P^0(\tilde{d}) - \sum_{\tilde{d}} P^0(\tilde{d}) \log P_\theta^\infty(\tilde{d})
= H(P^0) - \left< \log P_\theta^\infty(\tilde{d}) \right>_{P^0}
\]

\[
\frac{\partial P^0\|_{P_\theta^\infty}}{\partial \theta_m} = -\left< \frac{\partial \log P_\theta^\infty(\tilde{d})}{\partial \theta_m} \right>_{P^0}
\]
Contrastive Divergence

- We want to “update the parameters to reduce the tendency of the chain to wander away from the initial distribution on the first step”.

$$\frac{\partial}{\partial \theta_m} \left( P^0 \beta P^\infty - P^l \beta P^\infty \right) = -\left\langle \frac{\partial \log P^\infty_{\theta_m} (\tilde{d})}{\partial \theta_m} \right\rangle_{P^0} - \left\langle \frac{\partial \log P^\infty_{\theta_m} (\tilde{d})}{\partial \theta_m} \right\rangle_{P^l_{\theta}}$$

$$\propto \left\langle \frac{\partial \log f_m (\tilde{d} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^0} - \left\langle \frac{\partial \log f_m (\tilde{c} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^\infty_{\theta}} - \left\langle \frac{\partial \log f_m (\tilde{d} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^l_{\theta}} + \left\langle \frac{\partial \log f_m (\tilde{c} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^\infty_{\theta}}$$

$$\propto \left\langle \frac{\partial \log f_m (\tilde{d} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^0} - \left\langle \frac{\partial \log f_m (\tilde{d} \mid \theta_m)}{\partial \theta_m} \right\rangle_{P^l_{\theta}}$$
Contrastive Divergence (Final Result!)

\[ \Delta \theta_m \propto \left\langle \frac{\partial \log f_{\theta_m}}{\partial \theta_m} \right\rangle_{p^0} - \left\langle \frac{\partial \log f_{\theta_m}}{\partial \theta_m} \right\rangle_{p_{\theta}^1} \]

Law of Large Numbers, compute expectations using samples.

\[ \Delta \theta_m \propto \frac{1}{N} \sum_{d \in D} \frac{\partial \log f_{\theta_m}(d)}{\partial \theta_m} - \frac{1}{N} \sum_{c \sim P_{\theta}^1} \frac{\partial \log f_{\theta_m}(c)}{\partial \theta_m} \]

Now you know how to do it and why it works!