University of Illinois Fall 2018

ECE 544NA: Pattern Recognition
Lecture 18: October 25

Lecturer: Alexander Schwing Scribe: Yuan Shen

1 Overview

In the previous lectures, Professor introduced how we can use kMeans/Gaussian mixture model algorithm to model the distribution of the data x^i. For this class, Expectation Maximization (EM),a more generalized algorithm, is introduced, which gives us more freedom to model distribution by alternating optimization process.

1.1 Goal of this lecture

- Generalizing the kMeans/Gaussian mixture model algorithm
- Getting to know the Concave-convex procedure (CCCP)

1.2 Reading material

- C. Bishop; Pattern Recognition and Machine Learning; Chapter 9.3, 9.4
- Yuille and Rangarajan; Concave Convex Procedure (CCCP); NIPS 2001
- K. Murphy; Machine Learning: A Probabilistic Perspective; Chapter 11

1.3 Recap from previous lecture

We derived the E-step and M-step for the case of Gaussian Mixture Model in the last lecture. To solve the inflexibility issue of single Gaussian, we introduced the following constraints:

$$\sum_{k=1}^{K} \pi_k = 1, \pi_k \geq 0 \tag{1}$$

, where K is the number of cluster centers and the mixing weights $\pi_k \in [0, 1]$ And then we apply the standard procedure, minimize negative log-likelihood:

$$\min_{\pi,\mu,\sigma} -\log \prod_{i\in\mathcal{D}} p(x^{(i)}|\pi, \mu, \sigma) = -\sum_{i\in\mathcal{D}} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(x^{(i)}|\mu_k, \sigma_k) \tag{2}$$

More generally, we are minimize negative log-likelihood of the following:

$$\begin{aligned}\sum_{\mathbf{z_i}} \log p_\theta(x^{(i)}|\mathbf{z_i}) p_\theta(\mathbf{z_i}) &:= \sum_{\mathbf{z_i}} \log p_\theta(x^{(i)}, \mathbf{z_i}) \\ &:= \log p_\theta(x^{(i)})\end{aligned} \tag{3}$$

, where θ are the parameters for the distribution of x. However, there is no closed form solution for the negative log likelihood of this function.

2 Expectation and Maximization algorithm

In order to optimize θ for the above objective function without a closed-form solution, people introduced two alternating optimization processes, which end up being identical.

- Empirical Lower Bound
- Concave-Convex Procedure/Majorize-Minimize (CCCP)

2.1 Empirical Lower Bound

2.1.1 Goal

Using minimize negative log-likelihood method to optimize θ such that the combinations of latent distributions is close to the real distribution of x.

$$NLL(\theta) := -\sum_{i \in \mathcal{D}} \log p_\theta(x^{(i)}) \tag{4}$$

We will focus on optimizing the inner elements inside the Σ for later section.

2.1.2 A bit about KL divergence

D_{KL} is the expectation of the log difference between the probability of data in the original distribution with the approximating distribution. It calculates how many information is lost when we approximate one distribution with another.

$$\begin{aligned}D_{KL}(p,q) &= \sum_{i=1}^{N} p(x_i) \log \frac{p(x_i)}{q(x_i)} \\ &= E[\log p(x) - \log q(x)]\end{aligned} \tag{5}$$

you can think of $\log_2$ as ”how many bits of information we need to represent a probability”. Find details of KL divergence at this tutorial.

2.1.3 Jensen's inequality

- When function f is concave:

$$f(\sum_{\mathbf{z}} q(\mathbf{z})g(\mathbf{z})) \leq \sum_{\mathbf{z}} q(\mathbf{z})f(g(\mathbf{z})) \tag{6}$$

- When function f is convex:

$$f(\sum_{\mathbf{z}} q(\mathbf{z})g(\mathbf{z})) \geq \sum_{\mathbf{z}} q(\mathbf{z})f(g(\mathbf{z})) \tag{7}$$

2.1.4 Method

Let's start by introducing distribution q($\mathbf{z_i}$) and rewrite

$$\begin{aligned}
\log p_\theta(x^{(i)}) &= \log \sum_{\mathbf{z_i}} p_\theta(x^{(i)}, \mathbf{z_i}) \\
&= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(x^{(i)}, \mathbf{z_i})}{q(\mathbf{z_i})} + \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{q(\mathbf{z_i})}{p_\theta(\mathbf{z_i}|x^{(i)})} \\
&= \mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) + D_{KL}(q(\mathbf{z_i}), p_\theta(\mathbf{z_i}|x^{(i)}))
\end{aligned} \tag{8}$$

, where D_{KL} is Kullback-Leibler divergence. Pay attention to the difference of domain for $\mathcal{L}$ and D_{KL}.

How RHS = LHS?

$$\begin{aligned}
\text{RHS of (8)} := \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log[\frac{p_\theta(x^{(i)}, \mathbf{z_i})}{q(\mathbf{z_i})} \times \frac{q(\mathbf{z_i})}{p_\theta(\mathbf{z_i}|x^{(i)})}] \\
&= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(x^{(i)}, \mathbf{z_i})}{p_\theta(\mathbf{z_i}|x^{(i)})} \\
&= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(\mathbf{z_i}|x^{(i)}) \times p_\theta(x^{(i)})}{p_\theta(\mathbf{z_i}|x^{(i)})} \\
&= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log p_\theta(x^{(i)}) \\
&= \log p_\theta(x^{(i)})
\end{aligned} \tag{9}$$

Apply Jensen's inequality to D_{KL}, treat $\log(x)$ function as f. Notice that $\log(x)$ is concave, so we can use the concave case of Jensen's inequality from right to left :

$$\begin{aligned}
-D_{KL}(q(\mathbf{z_i}), p_\theta(\mathbf{z_i}|x^{(i)})) &= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(\mathbf{z_i}|x^{(i)})}{q(\mathbf{z_i})} \\
&\leq \log \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \times \frac{p_\theta(\mathbf{z_i}|x^{(i)})}{q(\mathbf{z_i})} \\
&\leq \log \sum_{\mathbf{z_i}} p_\theta(\mathbf{z_i}|x^{(i)}) \\
&\leq \log 1 \\
&\leq 0
\end{aligned} \tag{10}$$

Therefore, KL divergence is non-negative. If you interpret KL divergence as the information loss, then it's quite an obvious conclusion. You can never loss more than you have if you use another distribution to approximate current distribution.

Then,

$$\log p_\theta(x^{(i)}) \geq \mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) \tag{11}$$

Instead of maximizing log-likelihood, let's maximize lower bound:

$$\max_{q,\theta} \mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) \tag{12}$$

Alternating optimization:

- Maximize w.r.t. q:

  $$q(\mathbf{z_i}) = p_\theta(\mathbf{z_i}|x^{(i)}) \tag{13}$$

  From equation (10), we know if and only if (13) is true, then KL divergence = 0, and therefore RHS of (12) reaches the upper bound. Recall that KL divergence = 0 means there is no information loss, namely two distributions are exactly identical.

  With that, let's plug (13) into (12), and expand it in Gaussian case:

  $$\begin{aligned} \mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) &= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(x^{(i)}, \mathbf{z_i})}{q(\mathbf{z_i})} \\ &= \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{\prod_{k=1}^{K} \pi_k^{z_{ik}} \mathcal{N}(x^{(i)}|\mu_k, \sigma_k)^{z_{ik}}}{q(\mathbf{z_i})} \\ &= \sum_{\mathbf{z_i},k} q(\mathbf{z_i}) \log \pi_k^{z_{ik}} \mathcal{N}(x^{(i)}|\mu_k, \sigma_k)^{z_{ik}} + H(q(\mathbf{z_i})) \\ &= \sum_{k} r_{ik} \log \pi_k \mathcal{N}(x^{(i)}|\mu_k, \sigma_k) + H(q(\mathbf{z_i})) \end{aligned} \tag{14}$$

  ,where r_{ik} is the responsibility that cluster k takes for data point i.

  For general case:
  We represent the joint distribution between $x^{(i)}$ and $\mathbf{z_i}$ given θ as the following form:

  $$p_\theta(x^{(i)}, \mathbf{z_i}) = \frac{1}{Z(\theta)} \exp F(x^{(i)}, \mathbf{z_i}, \theta) \tag{15}$$

  , where $Z(\theta)$ is called partition function.

  With that, let's plug (15) into (12):

  $$\begin{aligned} -\mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) &= -\sum_{\mathbf{z_i}} q(\mathbf{z_i}) \log \frac{p_\theta(x^{(i)}, \mathbf{z_i})}{q(\mathbf{z_i})} \\ &= \log Z(\theta) - \sum_{\mathbf{z_i}} q(\mathbf{z_i}) F(x^{(i)}, \mathbf{z_i}, \theta) - H(q(\mathbf{z_i})) \end{aligned} \tag{16}$$

- Maximize the lower bound $\mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i}))$ w.r.t. θ

  Entropy term $H(q(\mathbf{z}))$ isn't important when we are optimizing parameters for M-steps, because it is constant w.r.t. θ

  For further reading for this section, please take a look at page 363, Machine Learning A Probabilistic Perspective [2]. It offers a pretty clear explanation.

2.1.5 Alternative approach to show that $q(\mathbf{z_i}) = p_\theta(\mathbf{z_i}|x^{(i)})$ for general case

$$\begin{aligned}
&\max_q \mathcal{L}(p_\theta(x^{(i)}, \mathbf{z_i}), q(\mathbf{z_i})) \\
&= \max_q \sum_{\mathbf{z_i}} q(\mathbf{z_i}) \ln p_\theta(x^{(i)}, \mathbf{z_i}) + H(q(\mathbf{z_i})), \text{ s.t.} q(\mathbf{z_i}) \geq 0, \text{ and } \sum_{\mathbf{z_i}} q(\mathbf{z_i}) = 1
\end{aligned} \tag{17}$$

We can use stationary of Lagrangian to solve.

Solution:

$$\begin{aligned}
q(\mathbf{z_i}) &= \frac{p_\theta(x^{(i)}, \mathbf{z_i})}{\sum_{\mathbf{z_i}} p_\theta(x^{(i)}, \mathbf{z_i})} \\
&= p_\theta(\mathbf{z_i}|x^{(i)}) \\
&= r_i
\end{aligned} \tag{18}$$

2.1.6 The observed data log likelihood monotonically increase

- Definition
  The observed data log likelihood:

$$\begin{aligned}
l(\theta) &= \sum_{i=1}^{N} \log \sum_{\mathbf{z_i}} p(x_i, z_i|\theta) \\
&= \sum_{i=1}^{N} \log \sum_{\mathbf{z_i}} q(\mathbf{z_i} \frac{p(x_i, z_i|\theta)}{q(\mathbf{z_i})}
\end{aligned} \tag{19}$$

  ,where $q(z_i)$ is an arbitrary distribution over the hidden variables.

  Since log(u) is a concave function, so we can apply Jensen's inequality equation to get the lower bound:

$$l(\theta) \geq \sum_i \sum_{\mathbf{z_i}} q_i(z_i) \log \frac{p(x_i, z_i|\theta)}{q_i(z_i)} \tag{20}$$

  Let's define the lower bound as $Q(\theta, q)$

- Inequality

  Then the following inequality equations will have hold as iteration t increases:

$$l(\theta) \geq \sum_i Q(\theta^{t+1}, \theta^t) \geq Q(\theta^t, \theta^t) = l(\theta^t) \tag{21}$$

  This is a very nice feature for debug purpose. If you find the log likelihood doesn't increase, then there are probably some bugs in your code.

  For further reading for this section, please take a look at page 365, Machine Learning A Probabilistic Perspective. It offers a pretty clear explanation. [2]

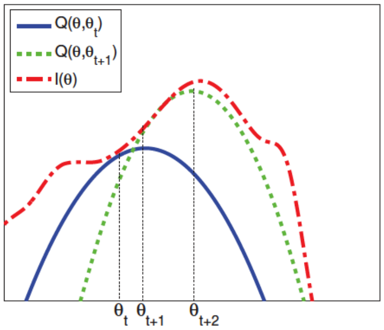


Figure 11.16 Illustration of EM as a bound optimization algorithm. Based on Figure 9.14 of (Bishop 2006a). Figure generated by **`emLogLikelihoodMax`**.

[2]

2.2 Concave-Convex Procedure, (CCCP)

2.2.1 General model

We represent the joint distribution between $x^{(i)}$ and $\mathbf{z}$ given θ as the following form:

$$p_\theta(x^{(i)}, \mathbf{z}) = \frac{1}{Z(\theta)} \exp F(x^{(i)}, \mathbf{z}, \theta) \tag{22}$$

, where $Z(\theta)$ is called partition function.

And then we optimize the parameter θ using negative log likelihood estimation:

$$\begin{aligned} &\min_\theta - \ln \sum_{\mathbf{z}} \frac{1}{Z(\theta)} \exp F(x^{(i)}, \mathbf{z}, \theta) \\ = &\min_\theta (\ln Z(\theta) - \ln \sum_{\mathbf{z}} p_\theta(x^{(i)}, \mathbf{z}, \theta)) \end{aligned} \tag{23}$$

2.2.2 Basic idea

- Decompose functions into convex and concave parts.
- Linearalize the concave parts
- Update the parameters using Jensen's inequality
- Repeating the above steps

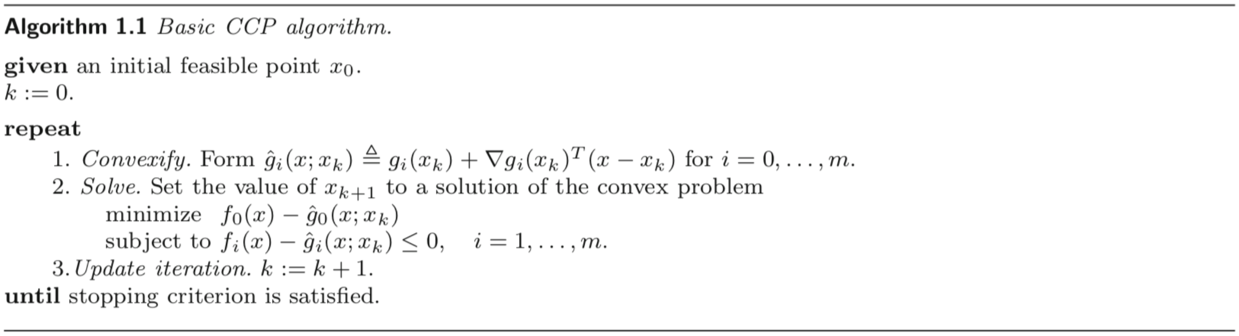
2.2.3 Procedure

- Initialize θ
- Repeat:

  Decompose concave part into "concave + convex" at current θ

  Solve convex program

2.2.4 Pseudo code



```
Algorithm 1.1 Basic CCP algorithm.
given an initial feasible point $x_0$.
$k := 0$.
repeat
    1. Convexify. Form $\hat{g}_i(x; x_k) \triangleq g_i(x_k) + \nabla g_i(x_k)^T(x - x_k)$ for $i = 0, \ldots, m$.
    2. Solve. Set the value of $x_{k+1}$ to a solution of the convex problem
        minimize $f_0(x) - \hat{g}_0(x; x_k)$
        subject to $f_i(x) - \hat{g}_i(x; x_k) \leq 0, \quad i = 1, \ldots, m$.
    3. Update iteration. $k := k + 1$.
until stopping criterion is satisfied.
```

This pseudo code is from a paper on the extension of concave-convex procedure from Stanford. [1]

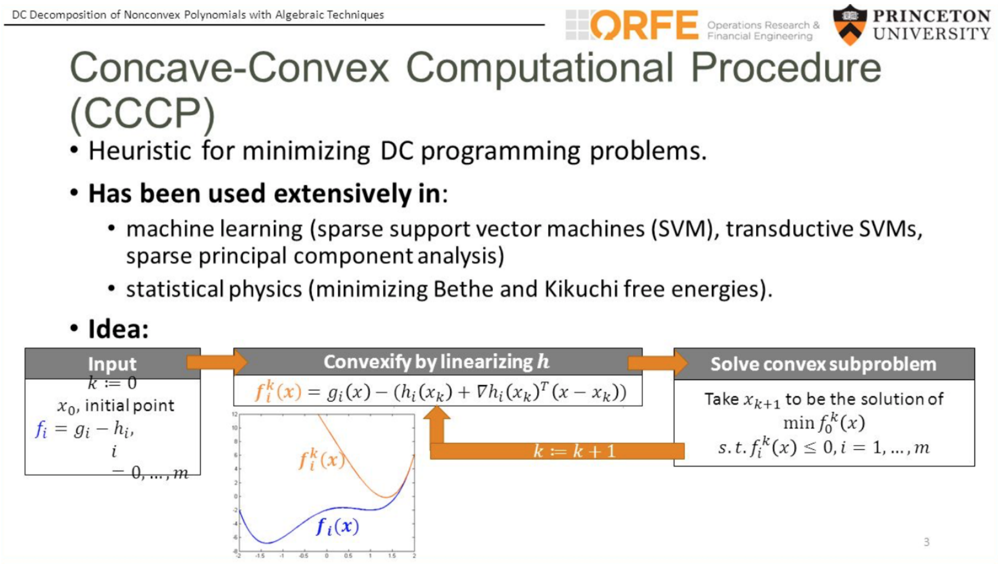
2.3 How to decompose

Using Jensen's inequality:

$$\begin{aligned} &\ln \sum_{\mathbf{z}} \exp F(x^{(i)}, \mathbf{z}, \theta) \\ &= \ln \sum_{\mathbf{z}} q(\mathbf{z}) \frac{\exp F(x^{(i)}, \mathbf{z}, \theta}{q(\mathbf{z})} \\ &= \max_{q(\mathbf{z})} (\sum_{\mathbf{z}} q(\mathbf{z}) F(x^{(i)}, \mathbf{z}, \theta) + H(q(\mathbf{z}))) \end{aligned} \tag{24}$$

, where we apply the convex case of Jensen's inequality to find the lower bound. And to maximize the lower bound will give us the estimated distribution that close to the actual distribution.

Notice that $\ln \sum_{\mathbf{z}} \exp F(x^{(i)}, \mathbf{z}, \theta)$ are called partition function.
And the part $F(x^{(i)}, \mathbf{z}, \theta)$ is called variational function.



DC Decomposition of Nonconvex Polynomials with Algebraic Techniques

ORFE Operations Research & Financial Engineering

PRINCETON UNIVERSITY

Concave-Convex Computational Procedure (CCCP)

- Heuristic for minimizing DC programming problems.
- **Has been used extensively in**:
  - machine learning (sparse support vector machines (SVM), transductive SVMs, sparse principal component analysis)
  - statistical physics (minimizing Bethe and Kikuchi free energies).
- **Idea:**

Input

$k := 0$

x_0, initial point

$f_i = g_i - h_i,$ $i = 0, \dots, m$

Convexify by linearizing h

$f_i^k(x) = g_i(x) - (h_i(x_k) + \nabla h_i(x_k)^T(x - x_k))$

Solve convex subproblem

Take x_{k+1} to be the solution of

$\min f_0^k(x)$

$s.t.\ f_i^k(x) \leq 0, i = 1, \dots, m$

3 Summary

Important topics of this lecture

- Generalizing EM algorithm
- Getting to know its relationship with CCCP
- Seeing the variational form of the partition function
- Observing its similarity to inference

4 Quiz

- Jensens inequality?
  - When function f is concave:

$$f(\sum_{\mathbf{z}} q(\mathbf{z})g(\mathbf{z})) \leq \sum_{\mathbf{z}} q(\mathbf{z})f(g(\mathbf{z})) \tag{25}$$

  - When function f is convex:

$$f(\sum_{\mathbf{z}} q(\mathbf{z})g(\mathbf{z})) \geq \sum_{\mathbf{z}} q(\mathbf{z})f(g(\mathbf{z})) \tag{26}$$

- General idea of CCCP?

  Decompose objective function into concave and convex part, optimize concave part using Jensen's inequality. And alternating the above procedure until terminating conditions are met.

References

[1] B. S. Lipp T. *Variations and extension of the convexconcave procedure.* Springer Science+Business, New York, US, 2015.

[2] K. P. Murphy. *Machine Learning, A Probabilistic Perspective.* The MIT Press, London, England, 2012.