Problem Set 3, CS229(Machine Learning)

Ma Yubo

July, 25th, 2021

1 A Simple Neural Network

1.1

By the structure of Neural Network, we know that:

$$h^{(i)} = \sigma(W^{[1]}x^{(i)}) \tag{1.1}$$

$$o^{(i)} = \sigma(W^{[2]}h^{(i)}) \tag{1.2}$$

$$\begin{aligned}\frac{\partial l}{\partial w_{1,2}^{[1]}} &= \frac{2}{m}\sum_{i=1}^{m}\frac{\partial l}{\partial o^{(i)}}\frac{\partial o^{(i)}}{\partial h_2}\frac{\partial h_2}{\partial w_{1,2}^{[1]}} \\ &= \frac{2}{m}\sum_{i=1}^{m}(o^{(i)}-y^{(i)})o^{(i)}(1-o^{(i)})W_2^{[2]}h^{(i)}(1-h^{(i)})x_1^{[i]}\end{aligned} \tag{1.3}$$

1.2

Yes, it is possible. By the hint, we can get an 100% accuracy as long as we could get a triangle decision boundary within which contains all points with label 0. So we first use hidden layer to depict this triangle. (For simplicity, we ignore the subscript (i) for each sample.)

$$h_1 = f(x_1 - 0.5) \tag{1.4}$$

$$h_2 = f(x_2 - 0.5) \tag{1.5}$$

$$h_3 = f(x_1 + x_2 - 4) \tag{1.6}$$

We found a samples has label 0 if and only if $h_1 = 1, h_2 = 1, h_3 < 0$, where h_j is a binary variable. So the output layer can be:

$$o = -h_1 - h_2 + \frac{3}{2}h_3 + \frac{3}{2} \tag{1.7}$$

1.3

No, it is impossible. Because linear active function makes that the neural network is still a linear classifier. So the decision boundary is a linear hyper-plane. But the data given isn't linear-separable obviously.

2 KL Divergence and Maximum Likelihood

2.1 Non-negativity

$$
\begin{aligned}
D_{KL}(p||q) &= \sum_{x \in X} p(x) log \frac{p(x)}{q(x)} \\
&= -E_{x \sim p(x)}[log \frac{q(x)}{p(x)}] \\
&\geq -log E_{x \sim p(x)}[\frac{q(x)}{p(x)}] \\
&= -log(\sum_{x \in X} q(x)) = 0
\end{aligned}
\tag{2.1}
$$

By the definition of Jesen Inequality we know, the equality condition satisfies only when $q(x)/p(x)$ is a constant. So the following is proved:

$$
\forall p, q \quad D_{KL}(p||q) \geq 0 \tag{2.2}
$$

and $D_{KL}(p||q) = 0$ if and only if $p = q$.

2.2 Chain Rule for KL Divergence

$$
\begin{aligned}
D_{KL}(p(x,y)||q(x,y)) &= \sum_{x,y} p(x,y) log \frac{p(x,y)}{q(x,y)} \\
&= \sum_{x,y} p(y|x)p(x) log \frac{p(y|x)p(x)}{q(y|x)q(x)} \\
&= \sum_{x,y} p(y|x)p(x) log \frac{p(y|x)}{q(y|x)} + \sum_{x,y} p(x,y) log \frac{p(x)}{q(x)} \\
&= \sum_{x} p(x) \sum_{y} p(y|x) log \frac{p(y|x)}{q(y|x)} + \sum_{x} p(x) log \frac{p(x)}{q(x)} \\
&= D_{KL}(p(y|x)||q(y|x)) + D_{KL}(p(x)||q(x))
\end{aligned}
\tag{2.3}
$$

2.3 KL and maximum likelihood

Given the context we know: $\hat{P}(x) = \frac{1}{m} \sum I(x = x^{(i)})$. Thus we have,

$$
\begin{aligned}
argmin_{\theta} D_{KL}(\hat{P}||P_{\theta}) &= argmax_{\theta} \sum_{x} \hat{P}(x) log P_{\theta}(x) \\
&= argmax_{\theta} \sum_{x} \sum_{i=1}^{m} I(x = x^{(i)}) log P_{\theta}(x) \\
&= argmax_{\theta} \sum_{i=1}^{m} log P_{\theta}(x^{(i)})
\end{aligned}
\tag{2.4}
$$

3 KL Divergence, Fisher Information, Natural Gradient

3.1 Score function

$$
\begin{aligned}
&E_{y\sim p(y;\theta)}[\nabla_{\theta'} logp(y;\theta^{'})|_{\theta'=\theta}] \\
&= \int_y p(y;\theta)\nabla_\theta logp(y;\theta)dy \\
&= \int_y p(y;\theta)\frac{1}{p(y;\theta)}\nabla_\theta p(y;\theta)dy \\
&= \nabla_\theta \int_y p(y;\theta)dy \\
&= \nabla_\theta 1 = 0
\end{aligned}
\tag{3.1}
$$

3.2 Fisher Information

$$
\begin{aligned}
I(\theta) &= Cov_{y\sim p(y;\theta)}[\nabla_{\theta'} logp(y;\theta^{'})|_{\theta'=\theta}] \\
&= E_{y\sim p(y;\theta)}[(\nabla_{\theta'} logp(y;\theta^{'}) - \mu)(\nabla_{\theta'} logp(y;\theta^{'}) - \mu)^T|_{\theta'=\theta}]
\end{aligned}
\tag{3.2}
$$

where $\mu = E_{y\sim p(y;\theta)}[\nabla_{\theta'} logp(y;\theta^{'})|_{\theta'=\theta}] = 0$. Therefore, we have:

$$
I(\theta) = E_{y\sim p(y;\theta)}[\nabla_{\theta'} logp(y;\theta^{'})\nabla_{\theta'} logp(y;\theta^{'})^T|_{\theta'=\theta}]
\tag{3.3}
$$

3.3 Fisher Information(alternate form)

$$
\begin{aligned}
&E_{y\sim p(y;\theta)}[-\nabla^2_{\theta'} logp(y;\theta^{'})|_{\theta'=\theta}] \\
&= E_{y\sim p(y;\theta)}[-\nabla_{\theta'}(\nabla_{\theta'} logp(y;\theta^{'}))|_{\theta'=\theta}] \\
&= E_{y\sim p(y;\theta)}[-\nabla_{\theta'}(\frac{1}{p(y;\theta')}\nabla_{\theta'} p(y;\theta^{'}))|_{\theta'=\theta}] \\
&= E_{y\sim p(y;\theta)}[\frac{1}{p(y;\theta')^2}\nabla_{\theta'} p(y;\theta^{'})\nabla_{\theta'} p(y;\theta^{'})^T - \frac{1}{p(y;\theta')}\nabla^2_{\theta'} p(y;\theta^{'})|_{\theta'=\theta}] \\
&= \int_y p(y;\theta)(\frac{1}{p(y;\theta)}\nabla_\theta p(y;\theta))(\frac{1}{p(y;\theta)}\nabla_\theta p(y;\theta))^T dy + \int_y \nabla^2_\theta p(y;\theta)dy \\
&= \int_y p(y;\theta)\nabla_\theta logP(y;\theta)\nabla_\theta logP(y;\theta)^T dy + \nabla^2_\theta \int_y p(y;\theta)dy \\
&= E_{y\sim p(y;\theta)}[\nabla_{\theta'} logp(y;\theta^{'})\nabla_{\theta'} logp(y;\theta^{'})^T|_{\theta'=\theta}] \\
&= I(\theta)
\end{aligned}
\tag{3.4}
$$

3.4 Approximate D_{KL} with Fisher Information

Define functional: $f(\beta) = D_{KL}(P_\theta || P_\beta)$.
Take derivative and we will get:

$$\begin{aligned}
&\nabla_{\beta|\beta=\theta} D_{KL}(P_\theta||P_\beta) = -\int_y p(y;\theta)\nabla_{\beta|\beta=\theta} logp(y;\beta)dy \\
&= -\int_y p(y;\theta)\frac{1}{p(y;\theta)}\nabla_{\beta|\beta=\theta} p(y;\beta)dy \\
&= 0
\end{aligned} \tag{3.5}$$

Similarly, take second derivative on $f(\beta)$ and we get:

$$\begin{aligned}
&\nabla^2_\beta D_{KL}(P_\theta||P_\beta) \\
&= \int_y p(y;\theta)[\frac{1}{p(y;\beta)^2}(\nabla_\beta p(y;\beta))^2 - \frac{1}{p(y;\beta)}\nabla^2_\beta p(y;\beta)]dy
\end{aligned} \tag{3.6}$$

$$\Rightarrow \nabla^2_{\beta|\beta=\theta} D_{KL}(P_\theta||P_\beta) = I(\theta) \tag{3.7}$$

Therefore, by the definition of Taylor Expansion, let $\beta = \theta + d$, then we have:

$$\begin{aligned}
&D_{KL}(P_\theta||P_\theta + d) \\
&= D_{KL}(P_\theta||P_\theta) + d^T\nabla_{\beta|\beta=\theta+d} D_{KL}(P_\theta||P_\beta) + \frac{1}{2}d^T\nabla_{\beta|\beta=\theta+d} D_{KL}(P_\theta||P_\beta)d \\
&= \frac{1}{2}d^T I(\theta)d
\end{aligned} \tag{3.8}$$

3.5 Natural Gradient

Denote $l(\theta) = logp(y;\theta)$.Write down the constrained optimization problem as below:

$$max_d \quad l(\theta + d) = l(\theta) + d^T\nabla_\theta l(\theta) + o(d^2) \tag{3.9}$$

$$s.t. \quad D_{KL}(P_\theta||P_{\theta+d}) = c \tag{3.10}$$

We solve this problem by **Lagrangian Multiplier**.

$$L(d,\lambda) = l(\theta) + d^T\nabla_\theta l(\theta) - \lambda(\frac{1}{2}d^T I(\theta)d - c) \tag{3.11}$$

$$\Rightarrow \nabla_d L = \nabla_\theta l(\theta) - \lambda I(\theta)d = 0 \tag{3.12}$$

$$\Rightarrow \nabla_\lambda L = \frac{1}{2}d^T I(\theta)d - c = 0 \tag{3.13}$$

By (3.12), we have $d = \frac{1}{\lambda}I^{-1}(\theta)\nabla_\theta l(\theta)$. Substitute this term into (3.13) and then we have:

$$\frac{1}{2\lambda^2}\nabla_\theta l(\theta)^T I^{-1}(\theta)\nabla_\theta l(\theta) = c \tag{3.14}$$

$$\Rightarrow \lambda = \sqrt{\frac{1}{2c} \nabla_\theta l(\theta)^T I^{-1}(\theta) \nabla_\theta l(\theta)} \tag{3.15}$$

Then we get optimal d^*:

$$\begin{aligned} d^* &= \frac{1}{\lambda} I^{-1}(\theta) \nabla_\theta l(\theta) \\ &= \frac{\sqrt{2c}}{[\nabla_\theta l(\theta)^T I^{-1}(\theta) \nabla_\theta l(\theta)]^{1/2}} I^{-1}(\theta) \nabla_\theta l(\theta) \end{aligned} \tag{3.16}$$

3.6 Relation to Newton's method

The update rule following natural gradient is:

$$\theta := \theta + \frac{\sqrt{2c}}{[\nabla_\theta l(\theta)^T I^{-1}(\theta) \nabla_\theta l(\theta)]^{1/2}} I^{-1}(\theta) \nabla_\theta l(\theta) \tag{3.17}$$

And the update rule following method is:

$$\theta := theta - \alpha H^{-1}(\theta) \nabla_\theta l(\theta) \tag{3.18}$$

By the definition of Hessian matrix and Fisher Informative matrix, we have:

$$I(\theta) = -E_{y \sim p(y;\theta)}[\nabla_\theta^2 l(\theta)] = -E_{y \sim p(y;\theta)}[H(\theta)] \tag{3.19}$$

We had derived in **Question 4, Problem set 1** that for GLM, the Hessian matrix about NLL function is unrelated to the y. Thus,

$$I(\theta) = -E_{y \sim p(y;\theta)}[H(\theta)] = -H(\theta) \tag{3.20}$$

By adjusting the learning rate of Newton's method as $\alpha = \frac{\sqrt{2c}}{[\nabla_\theta l(\theta)^T I^{-1}(\theta) \nabla_\theta l(\theta)]^{1/2}}$, the direction of update of Newton's method is equivalent to that of natural gradient. It is an interesting property for GLM.

4 Semi-supervised EM

4.1 Convergence

$$
\begin{aligned}
l(\theta^{(t+1)}) &= \sum_{i=1}^{m} logp(x^{(i)};\theta^{(t+1)}) \\
&= \sum_{i=1}^{m} log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta^{(t+1)}) \\
&\geq \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) log \frac{p(x^{(i)}, z^{(i)};\theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})} \\
&= \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) log \frac{p(x^{(i)}, z^{(i)};\theta^{(t)})}{Q_i^{(t)}(z^{(i)})}
\end{aligned}
\tag{4.1}
$$

By the definition of E-step, we know that $Q_i^{(t)}(z^{(i)}) = p(z^{(i)}|x^{(i)};\theta^{(t)})$. Thus

$$
\frac{p(x^{(i)}, z^{(i)};\theta^{(t)})}{Q_i^{(t)}(z^{(i)})} = p(x^{(i)};\theta^{(t)}) = const \quad (w.r.t \quad z^{(i)})
\tag{4.2}
$$

By the property of Jesen Inequality, the following equation holds:

$$
\begin{aligned}
&\sum_{i=1}^{m} \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) log \frac{p(x^{(i)}, z^{(i)};\theta^{(t)})}{Q_i^{(t)}(z^{(i)})} \\
&= \sum_{i=1}^{m} log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta^{(t+1)}) \\
&= \sum_{i=1}^{m} logp(x^{(i)};\theta^{(t)}) \\
&= l(\theta^{(t)})
\end{aligned}
\tag{4.3}
$$

Combining (4.1) and (4.3) we know that $l(\theta^{(t+1)}) \geq l(\theta^{(t)})$ holds strictly.

4.2 Semi-supervised E-step

$$
\begin{aligned}
w_j^{(i)} &= Q_i(z^{(i)} = j) \\
&= p(z^{(i)} = j|x^{(i)};\theta) \\
&= \frac{p(x^{(i)}|z^{(i)} = j;\theta)p(z^{(i)} = j;\theta)}{\sum_k p(x^{(i)}|z^{(i)} = k;\theta)p(z^{(i)} = k;\theta)} \\
&= \frac{\frac{\phi_j}{|\Sigma_j|^{1/2}} exp(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1}(x^{(i)} - \mu_j))}{\sum_k \frac{\phi_k}{|\Sigma_k|^{1/2}} exp(-\frac{1}{2}(x^{(i)} - \mu_k)^T \Sigma_k^{-1}(x^{(i)} - \mu_k))}
\end{aligned}
\tag{4.4}
$$

4.3 Semi-supervised M-step

$$
\begin{aligned}
l(\theta) &= \sum_{i=1}^{m}\sum_{j=1}^{k} Q_i(z^{(i)}=j) log \frac{p(x^{(i)}, z^{(i)}=j;\theta)}{Q_i(z^{(i)}=j)} + \alpha \sum_{i=1}^{\tilde{m}} logp(\tilde{x}^{(i)}, \tilde{z}^{(i)};\theta) \\
&= \sum_{i=1}^{m}\sum_{j=1}^{k} w_j^{(i)} log \frac{\frac{\phi_j}{(2\pi)^{d/2}|\Sigma_j|^{1/2}} exp(-\frac{1}{2}(x^{(i)}-\mu_j)^T \Sigma_j^{-1}(x^{(i)}-\mu_j))}{w_j^{(i)}} \\
&+ \alpha \sum_{i=1}^{\tilde{m}} log[\frac{\phi_{\tilde{z}^{(i)}}}{(2\pi)^{d/2}|\Sigma_{\tilde{z}^{(i)}}|^{1/2}} exp(-\frac{1}{2}(x^{(i)}-\mu_{\tilde{z}^{(i)}})^T \Sigma_{\tilde{z}^{(i)}}^{-1}(x^{(i)}-\mu_{\tilde{z}^{(i)}})]
\end{aligned} \tag{4.5}
$$

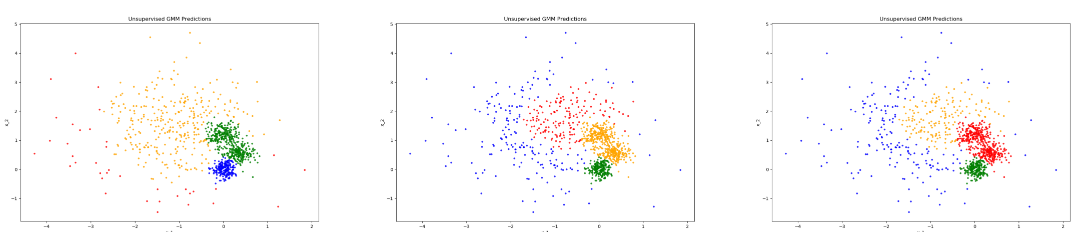
we need to maximize, with respect to our parameters ϕ, μ and Σ.

$$
\begin{aligned}
\nabla_{\phi_l} l(\theta) &= 0 \\
\nabla_{\mu_l} l(\theta) &= 0 \\
\nabla_{\Sigma_l} l(\theta) &= 0
\end{aligned} \tag{4.6}
$$

Finally we get:

$$
\begin{aligned}
\phi_l &= \frac{\sum_{i=1}^{m} w_l^{(i)} + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=l)}{\sum_k (\sum_{i=1}^{m} w_k^{(i)} + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=k))} \\
\mu_l &= \frac{\sum_{i=1}^{m} w_l^{(i)} x^{(i)} + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=l) x^{(i)}}{\sum_{i=1}^{m} w_l^{(i)} + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=l)} \\
\Sigma_l &= \frac{\sum_{i=1}^{m} w_l^{(i)} (x^{(i)}-\mu_l)(x^{(i)}-\mu_l)^T + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=l)(x^{(i)}-\mu_l)(x^{(i)}-\mu_l)^T}{\sum_{i=1}^{m} w_l^{(i)} + \alpha \sum_{i=1}^{\tilde{m}} I(\tilde{z}^{(i)}=l)}
\end{aligned} \tag{4.7}
$$

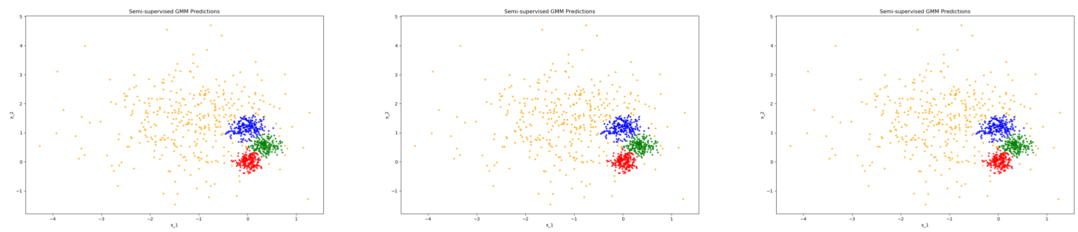
4.4 Unsupervised EM Implementation



(a) Trial 1 (b) Trial 2 (c) Trial 3

Figure 4.1: Unsupervised EM Implementation

4.5 Semi-supervised EM Implementation



(a) Trial 1 (b) Trial 2 (c) Trial 3

Figure 4.2: Semi-supervised EM Implementation

4.6 Comparison of Unsupervised and Semi-supervised EM

4.6.1 Number of iterations taken to converge

Under this setting, Unsupervised EM often takes several hundreds steps, sometimes even more than 1000 steps to converge. Semi-supervised EM converges much faster, which only takes less than 100 steps.

4.6.2 Stability

By the figures shown above, the results of Semi-supervised EM is more stable than that of unsupervised EM.

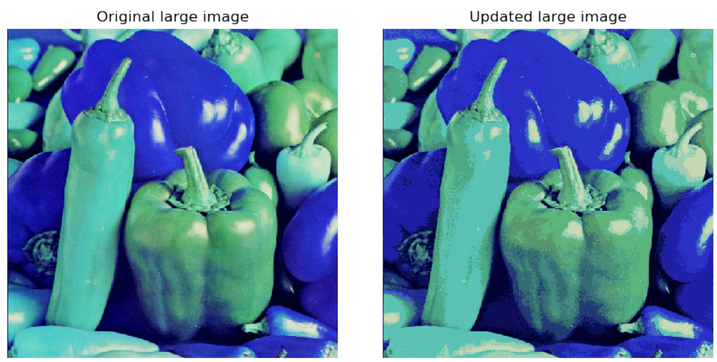
4.6.3 Overall quality of assignment

Prior knowledge tells us that **the dataset was sampled from a mixture of three low-variance Gaussian distributions, and a fourth, high-variance Gaussian distribution.**
We could see three clusters of low-variance Gaussian distributions (color in blue, green and red) and one cluster of high-variance Gaussian distribution (color in yellow) from the results of Semi-supervised EM.
It's hard to find similar clusters from the results of unsupervised EM. So the overall quality of semi-supervised EM is better.

5 K-means for compression

5.1 K-Means Compression Implementation

The source code is written in *p05_kmeans.py*. And here are results of image compression.



(a) Original image (b) Compressed image

Figure 5.1: Peppers image compression

5.2 Compression Factor

- Before compression, each pixel is represented with 24 bits: (r, g, b) three components. Each component with 8 bits).
- After compression, each pixel is represented as 4 bits: 16 candidate (r,g,b) choices.

In summary, the compression factor is approximately 6.