# CSCI567 Fall16 Homework 2

Dinh Nguyen
dinhnguy@usc.edu

Oct 2nd 2016

1. **Logistic Regression**

   (a) Consider a binary logistic regression model, given $n$ training examples $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ...,$ write down the negative log likelihood (as loss function):
   For cleaner equation, I append 1 to $\mathbf{x}_i$ and $b$ to $\mathbf{w}$ and assume possible values for $y_i$ to be $\{0, 1\}$;

$$
\begin{aligned}
\mathcal{L}(\mathbf{w}) &= -\log\left(\prod_{i=1}^{n} P(Y = y_i|\mathbf{X} = \mathbf{x_i})\right) \\
&= -\log\prod_{i=1}^{n}\left(\frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)}\right)^{y_i}\left(1-\frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)}\right)^{1-y_i} \\
&= \sum_{i=1}^{n} y_i\log(1+\exp(-\mathbf{w}^\top\mathbf{x}_i)) - (1-y_i)\log\left(\frac{\exp(-\mathbf{w}^\top\mathbf{x}_i)}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)}\right) \\
&= \sum_{i=1}^{n} y_i\log(1+\exp(-\mathbf{w}^\top\mathbf{x}_i)) - (1-y_i)\log(-\exp(\mathbf{w}^\top\mathbf{x}_i)) \\
&\quad + (1-y_i)\log(1+\exp(-\mathbf{w}^\top\mathbf{x}_i)) \\
&= \sum_{i=1}^{n}\log(1+\exp(-\mathbf{w}^\top\mathbf{x}_i)) + (1-y_i)\mathbf{w}^\top\mathbf{x}_i
\end{aligned}
$$

   (b) Use Gradient Descent Method to find the update rule for $\mathbf{w}$.

$$
\begin{aligned}
\frac{\partial\mathcal{L}}{\partial\mathbf{w}} &= \sum_{i=1}^{n}\frac{-\exp(-\mathbf{w}^\top\mathbf{x}_i)\mathbf{x}_i}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)} + (1-y_i)\mathbf{x}_i \\
&= \sum_{i=1}^{n}\left(-\frac{\exp(-\mathbf{w}^\top\mathbf{x}_i)}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)} + 1 - y_i\right)\mathbf{x}_i \\
&= \sum_{i=1}^{n}\left(\frac{1}{1+\exp(-\mathbf{w}^\top\mathbf{x}_i)} - y_i\right)\mathbf{x}_i
\end{aligned}
$$

   Update rule:

$$
\mathbf{w}_{new} = \mathbf{w}_{old} - \eta\sum_{i=1}^{n}\left(\frac{1}{1+\exp(-\mathbf{w}_{old}^\top\mathbf{x}_i)} - y_i\right)\mathbf{x}_i
$$

   for some approriate $\eta$.
   This solution will not converge to a global minimum since it will only go toward the closest local minimum and stop there since the gradient at that point would be $\mathbf{0}$.

(c) The negative log likelihood $\mathcal{L}(\mathbf{w}_1, ..., \mathbf{w}_K)$

$$
\begin{aligned}
\mathcal{L}(\mathbf{w}_1, ..., \mathbf{w}_K) &= -\log\left(\prod_{i=1}^{n} P(Y = y_i|\mathbf{X} = \mathbf{x_i})\right) \\
&= -\sum_{i=1}^{n} \log\left\{\prod_{k=1}^{K-1}\left(\frac{\exp(\mathbf{w}_k^\top \mathbf{x}_i)}{1+\sum_{t=1}^{K-1}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right)^{I_{(y_i=k)}}\right. \\
&\qquad \left.\times\left(\frac{1}{1+\sum_{t=1}^{K-1}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right)^{I_{(y_i=K)}}\right\}
\end{aligned}
$$

By letting $\mathbf{w}_K = \mathbf{0}$, we have $\exp(\mathbf{w}_K^\top \mathbf{x}_i) = 1$ so we can combine the $K$ term into to product as well as the sum in the denominator:

$$
\begin{aligned}
\mathcal{L}(\mathbf{w}_1, ..., \mathbf{w}_K) &= -\sum_{i=1}^{n} \log\left\{\prod_{k=1}^{K}\left(\frac{\exp(\mathbf{w}_k^\top \mathbf{x}_i)}{\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right)^{I_{(y_i=k)}}\right\} \\
&= -\sum_{i=1}^{n}\sum_{k=1}^{K} I_{(y_i=k)} \log\left(\frac{\exp(\mathbf{w}_k^\top \mathbf{x}_i)}{\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right) \\
&= -\sum_{i=1}^{n}\sum_{k=1}^{K} I_{(y_i=k)}\left(\log(\exp(\mathbf{w}_k^\top \mathbf{x}_i)) - \log(\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i))\right) \\
&= -\sum_{i=1}^{n}\sum_{k=1}^{K} I_{(y_i=k)}\left(\mathbf{w}_k^\top \mathbf{x}_i - \log(\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i))\right)
\end{aligned}
$$

(d) Compute the gradient of the negative log likelihood:

$$
\begin{aligned}
\frac{\partial \mathcal{L}(\mathbf{w}_1, ..., \mathbf{w}_K)}{\partial \mathbf{w}_k} &= -\sum_{i=1}^{n} I_{(y_i=k)}\left(\mathbf{x}_i - \frac{\exp(\mathbf{w}_k^\top \mathbf{x}_i)\mathbf{x}_i}{\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right) \\
&= -\sum_{i=1}^{n} I_{(y_i=k)}\left(1 - \frac{\exp(\mathbf{w}_k^\top \mathbf{x}_i)}{\sum_{t=1}^{K}\exp(\mathbf{w}_t^\top \mathbf{x}_i)}\right)\mathbf{x}_i
\end{aligned}
$$

Update rule:

$$
\mathbf{w}_k^{new} = \mathbf{w}_k^{old} + \eta\sum_{i=1}^{n} I_{(y_i=k)}\left(1 - \frac{\exp(\mathbf{w}_k^{(old)\top} \mathbf{x}_i)}{\sum_{t=1}^{K}\exp(\mathbf{w}_t^{(old)\top} \mathbf{x}_i)}\right)\mathbf{x}_i
$$

2. **Linear/ Gaussian Discriminant**

(a) Write the log likelihood function $\mathcal{L}(\mathcal{D})$
Since $y_n \in \{1,2\}$,

$$\mathcal{L}(\mathcal{D}) = \sum_{n=1}^{N} I(y_n = 1)\left( \log(p_1) - (1/2)\log(2\pi\sigma_1^2) - \frac{(x_n - \mu_1)^2}{2\sigma_1^2}\right)$$
$$+ I(y_n = 2)\left( \log(p_2) - (1/2)\log(2\pi\sigma_2^2) - \frac{(x_n - \mu_2)^2}{2\sigma_2^2}\right)$$

Use MLE to find $(p_1^*, p_2^*, \mu_1^*, \mu_2^*, \sigma_1^*, \sigma_2^*)$
Since $p_1 + p_2 = 1$, we can rewrite $p_2$ as $1 - p_1$ in the likelihood function before taking the derivative with respect to $p_1$:

$$\frac{\partial L}{\partial p_1} = \sum_{n=1}^{N} I(y_n = 1)/p_1 - I(y_n = 2)/(1 - p_1) = 0$$

Solve for $p_1$: we have

$$p_1^* = \frac{\sum_{n=1}^{N} I(y_n = 1)}{N}$$

since $\sum_{n=1}^{N} I(y_n = 1) + I(y_n = 2) = N$. And $p_2^* = 1 - p_1* = 1 - \frac{\sum_{n=1}^{N} I(y_n=1)}{N} = \frac{\sum_{n=1}^{N} I(y_n=2)}{N}$
For $\mu_i^*$, we have:

$$\frac{\partial L}{\partial \mu_i} = \sum_{n=1}^{N} I(y_n = i)\frac{x_n - \mu_i}{\sigma_i^2} = 0$$

Solve for $\mu_i^*$ we have

$$\mu_i^* = \frac{\sum_{n=1}^{N} I(y_n = i)x_n}{\sum_{n=1}^{N} I(y_n = i)}$$

i.e we take the average of all the $x_n$ where $y_n = i$.
Lastly, for $\sigma_i^*$:

$$\frac{\partial L}{\partial \sigma_i} = \sum_{n=1}^{N} I(y_n = i)\frac{(x_n - \mu_i)^2 - \sigma_i^2}{\sigma_i^2} = 0$$

Solve for $\sigma_i$:

$$\sigma_i^* = \sqrt{\frac{\sum_{n=1}^{N} I(y_n = i)(x_n - \mu_i^*)^2}{\sum_{n=1}^{N} I(y_n = i)}}$$

(b) Since the samples are iid, we know the covariance matrix for both classes is diagonal with $\sigma^2$ entries. For the mean vector, $\mu_1 = \mathbf{0}$ and $\mu_2 = (0, ..0, \delta, ..., \delta)$.

Plug those in the formula of $P(y = c_1|x, \mu, \Sigma)$ we have from last hw (problem 2) and set it equal to $1/2$. We have

$$P(y = c_1|\mathbf{x}, \mu, \Sigma) = \frac{1}{1 + p_2/p_1 \exp\left((\sum_{n=D}^{2D} \delta\sigma^2 x_n) - D\delta^2\sigma^2/2\right)} = \frac{1}{2}$$

So

$$p_1 = p_2 \exp\left((\sum_{n=D}^{2D} \delta\sigma^2 x_n) - D\delta^2\sigma^2/2\right)$$

Take log of both sides and we have the linear discriminant:

$$\sum_{n=D}^{2D} \delta\sigma^2 x_n = \log(p_1) - \log(p_2) + D\delta^2\sigma^2/2$$

which clearly depends on $\delta$ so it changes when $\delta$ changes.

(c) Let $P(y = c_i) = p_i$:

$$\begin{aligned} P(y = 1|\mathbf{x}) &= \frac{p_1 P(\mathbf{x}|y = 1)}{p_1 P(\mathbf{x}|y = 1) + p_2 P(\mathbf{x}|y = 2)} \\ &= \frac{1}{1 + \frac{p_2 P(\mathbf{x}|y=2)}{p_1 P(\mathbf{x}|y=1)}} \end{aligned}$$

Since $p_i P(\mathbf{x}|y = c_i)$ follows a multivariate Gaussian distribution with the same covariance matrix, we can cancel out the $(2\pi)^{D/2}|\Sigma|^{-1/2}$. So we have:

$$\begin{aligned} \frac{P(\mathbf{x}|y = 2)}{P(\mathbf{x}|y = 1)} &= \frac{\exp(-1/2(\mathbf{x} - \mu_2)^\top \Sigma^{-1}(\mathbf{x} - \mu_2))}{\exp(-1/2(\mathbf{x} - \mu_1)^\top \Sigma^{-1}(\mathbf{x} - \mu_1))} \\ &= \exp(-1/2(\mu_2^\top \Sigma^{-1}\mu_2 - \mu_1^\top \Sigma^{-1}\mu_1) - (\Sigma^{-1}(\mu_1 - \mu_2))^\top \mathbf{x}) \end{aligned}$$

So by setting $\theta = \left(\log(p_1) - \log(p_2) + 1/2(\mu_2^\top \Sigma^{-1}\mu_2 - \mu_1^\top \Sigma^{-1}\mu_1), \Sigma^{-1}(\mu_1 - \mu_2)\right)$ and append 1 to $\mathbf{x}$ we have the form of a logistic function.

3. **Perceptron and Online Learning**

By setting $\mathbf{w}_{i+1} = \mathbf{w}_i + (y_i - \mathbf{w}_i^\top \mathbf{x}_i)\frac{\mathbf{x}_i}{|\mathbf{x}_i|^2}$. Since we used $\mathbf{w}_i^\top \mathbf{x}_i$ to evaluate $y_i$ and it didn't work (otherwise we would not update ), by adding the difference to $w_i$ with a normalized $\mathbf{x}_i$, we guarantee to have get $y_i$ when using $w_{i+1}$:

$$\mathbf{w}_{i+1}^\top \mathbf{x}_i = \mathbf{w}_i^\top \mathbf{x}_i + y_i - \mathbf{w}_i^\top \mathbf{x}_i \frac{\mathbf{x}_i^\top \mathbf{x}_i}{|\mathbf{x}_i|^2} = y_i$$

since $\mathbf{x}_i^\top \mathbf{x}_i = |\mathbf{x}_i|^2$. Since we add the exact distance to $\mathbf{w}_i$, $\mathbf{w}_{i+1}$ is the closest vector to $\mathbf{w}_i$ that classifies correctly.

4. **Programming** All the answers are in the output of the scripts. Thanks.