# cs229 problem set 1

Bhavit Sharma

4/23/23

## Table of contents

## 1 Problem 1

### 1.1 (a)

Let's compute Hessian of $J(\theta)$ for one training sample. We have

$$J(\theta) = y \log \sigma(\theta^T x) + (1 - y) \log(1 - \sigma(\theta^T x))$$

Now, we compute the first derivate of $J(\theta)$ with respect to $\theta_i$:

$$\frac{\partial J(\theta)}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \left[ y \log \sigma(\theta^T x) + (1 - y) \log(1 - \sigma(\theta^T x)) \right]$$

We need to use the fact that derivative of $\sigma(\theta^T x)$ is $\frac{\partial}{\partial \theta_i} = \sigma(\theta^T x)(1 - \sigma(\theta^T x))(x[i])$ i.e. the derivative of $\sigma(\theta^T x)$ is $\sigma(\theta^T x)(1 - \sigma(\theta^T x))x$.

Using chain rule, we have

$$\frac{\partial J(\theta)}{\partial \theta_i} = y * (1 - \sigma(\theta^T x)) * x[i] + (1 - y) * (-\sigma(\theta^T x)) * x[i]$$

Simplifying, we have

$$\frac{\partial J(\theta)}{\partial \theta_i} = (y - \sigma(\theta^T x)) * x[i]$$

So for $n$ training samples, we have

$$\frac{\partial J(\theta)}{\partial \theta_i} = \sum_{j=1}^{n} (y_j - \sigma(\theta^T x_j)) * x_{ij}$$

Writing this in vector form, we have

$$\frac{\partial J(\theta)}{\partial \theta} = -\frac{1}{n} \sum_{j=1}^{n} (y_j - \sigma(\theta^T x_j)) * x_j$$

Now let us compute the Hessian of $J(\theta)$ with respect to $\theta_i$ and $\theta_j$: We know that the derivate with respect to $j$ is

$$\frac{\partial J(\theta)}{\partial \theta_j} = -\frac{1}{n} \sum_{j=1}^{n} (y_j - \sigma(\theta^T x_j)) * x_{ij}$$

So $H_{ij}$ is

$$\begin{aligned} H_{ij} = \frac{\partial^2 J(\theta)}{\partial \theta_i \partial \theta_j} &= \frac{\partial}{\partial \theta_i} \left[ -\frac{1}{n} \sum_{k=1}^{n} (y_k - \sigma(\theta^T x_k)) * x_{kj} \right] \\ &= -\frac{1}{n} \sum_{k=1}^{n} \frac{\partial}{\partial \theta_i} (y_k - \sigma(\theta^T x_k)) * x_{kj} \\ &= -\frac{1}{n} \sum_{k=1}^{n} (\sigma(\theta^T x_k)) * (\sigma(\theta^T x_k) - 1) * x_{ki} x_{kj} \end{aligned}$$

Writing it in matrix form, we have

$$H = \frac{1}{n} \sum_{k=1}^{n} (\sigma(\theta^T x_k)) * (1 - \sigma(\theta^T x_k)) * x_k x_k^T$$

**Now we want to show that the Hessian is positive semi-definite which implies that $J$ has a local minima and it's a convex function** The way it's done is by showing that for any vector $v$, we have

$$v^T H v \geq 0$$

Note: TODO(Bhavit): Why is this true?

$$v^T H v = \frac{1}{n}\sum_{k=1}^{n}(\sigma(\theta^T x_k)) * (1 - \sigma(\theta^T x_k)) v^T x_k x_k^T v$$

Now we can see that $V^T x_k x_k^T v$ can be written as

$$v^T x x^T v = \sum_{i=1}^{d}\sum_{j=1}^{d} v[i]x[i]x[j]v[j]$$

where $d$ is the dimension of $x$. Try to write this in matrix form and you can see. Now using the hint, we can easily see that the above form is equivalent to $v^T x x^T v = (v^T x)(v^T x) > 0$.

Since $\sigma(\theta^T x_k) \in [0, 1]$, we have $H \geq 0$ always.

## 1.2 (b)

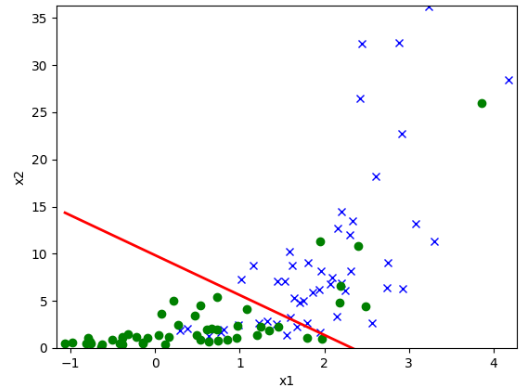

Figure 1: first

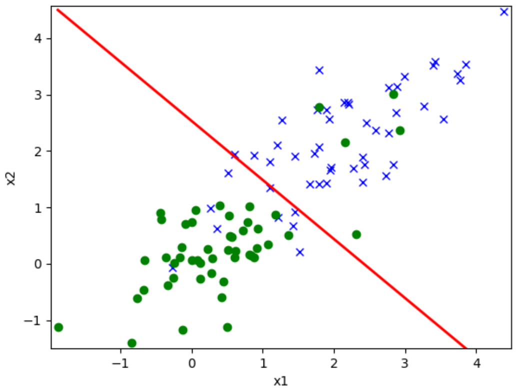

Figure 2: second

## 1.3 (c)

We need to show that GDA results in a classifier that has linear decision boundary i.e.

$$P(y=1|x;\phi,\mu_0,\mu_1,\Sigma)=\frac{1}{1+\exp(-(\theta^T x+\theta_0))}$$

Using bayes rule, we have

$$P(y=1|x;\phi,\mu_0,\mu_1,\Sigma)=\frac{P(x|y=1;\mu_1,\Sigma)P(y=1;\phi)}{P(x;\mu_0,\mu_1,\Sigma)}$$

Plugging formulas, we end up with an expression that looks like

$$P(y=1|x;\phi,\mu_0,\mu_1,\Sigma)=\frac{1}{1+\exp(-(f_1-f_0))}$$

and $f_0=\frac{-1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)+\log(1-\phi)$ and $f_1=\frac{-1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)+\log(\phi)$ and $f_1-f_0=\frac{-1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)+\log(\phi)-\frac{-1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)+\log(1-\phi)$

If we expand the above expression by opening the transpose, we get

$$f_1-f_0=\frac{-1}{2}(k)+\log(\phi)-\log(1-\phi)$$

$$\begin{aligned}k&=(x^T-\mu_1^T)\Sigma^{-1}(x-\mu_1)-(x^T-\mu_0^T)\Sigma^{-1}(x-\mu_0)\\&=\theta^T x+\theta_0\end{aligned}$$

We also use the fact $(X.Y)^T=Y^T.X^T$ and $(X+Y)^T=X^T+Y^T$.

where $\theta_0=(\mu_1^T\Sigma^{-1}\mu_1-\mu_0^T\Sigma^{-1}\mu_0)+\log(\phi)-\log(1-\phi)$ and $(\mu_0-\mu_1)^T\Sigma^{-1}=\theta$

## 1.4 (d)

We want to derive the parameters for $\phi,\mu_0,\mu_1,\Sigma$. We derive this using the log likelihood function. We know that the log likelihood function is

$$\ell(\phi,\mu_0,\mu_1,\Sigma)=\prod_{i=1}^{n}\log P(x^{(i)},y^{(i)};\phi,\mu_0,\mu_1,\Sigma)$$

$$=\sum_{i=1}^{n}\log P(x^{(i)}|y^{(i)};\mu_0,\mu_1,\Sigma)+\log P(y^{(i)};\phi)$$

$$\log P(x^{(i)}|y^{(i)};\mu_0,\mu_1,\Sigma)=\log\left(\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(x^{(i)}-\mu_{y^{(i)}})^T\Sigma^{-1}(x^{(i)}-\mu_{y^{(i)}})\right)\right)$$

$$= -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x^{(i)} - \mu_{y^{(i)}})^T\Sigma^{-1}(x^{(i)} - \mu_{y^{(i)}})$$

and

$$\log P(y^{(i)};\phi) = y^{(i)}\log\phi + (1 - y^{(i)})\log(1-\phi)$$

Now differentiating the above expression with respect to $\phi, \mu_0, \mu_1, \Sigma$ and setting it to zero, we get the following expressions.

$$\frac{\partial \ell}{\partial \phi} = \frac{1}{\phi}\sum_{i=1}^{n} y^{(i)} - \frac{1}{1-\phi}\sum_{i=1}^{n}(1 - y^{(i)}) = 0$$

### 1.4.1 Solve for $\phi$

$$\begin{aligned}
\frac{1}{\phi}\sum_{i=1}^{n} y^{(i)} &= \frac{1}{1-\phi}\sum_{i=1}^{n}(1 - y^{(i)}) \\
\sum_{i=1}^{n} y^{(i)} &= \frac{\phi}{1-\phi}\sum_{i=1}^{n}(1 - y^{(i)}) \\
\phi &= \frac{\sum_{i=1}^{n} y^{(i)}}{n}
\end{aligned}$$

### 1.4.2 For $\mu_0, \mu_1$

The proof follows similar for $\mu_0$ and $\mu_1$. We'll just prove for $\mu_0$.

$$\frac{\partial \ell}{\partial \mu_0} = \sum_{i=1}^{n} \frac{\partial}{\partial \mu_0} y_i \left(-\frac{1}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x^{(i)} - \mu_0)^T\Sigma^{-1}(x^{(i)} - \mu_0)\right)$$

We need to calculate this quantity: i.e. for $x = x^T U x$ where $x$ is a vector, $U$ is a matrix and $u$ is a vector, we need to calculate

$$\frac{\partial}{\partial x^T U x} x = 2Ux$$

if $U$ is symmetric.

So this means

$$\sum_{i=1}^{n} 2\Sigma^{-1}(x^{(i)} - \mu_0)y^{(i)} = 0$$

Since $\Sigma$ is symmetric and positive definite, we can write

$$\Sigma^{-1}\sum_{i=1}^{n}(x^{(i)} - \mu_0)y^{(i)} = 0$$

This gives the results we want.

### 1.4.3 For $\Sigma$

$$\frac{\partial \ell}{\partial \Sigma} = \sum_{i=1}^{n} \frac{\partial}{\partial \Sigma} y_i \left( -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x^{(i)} - \mu_0)^T \Sigma^{-1} (x^{(i)} - \mu_0) \right)$$

I have added the solution in my my-notes file.

## 1.5 (e)

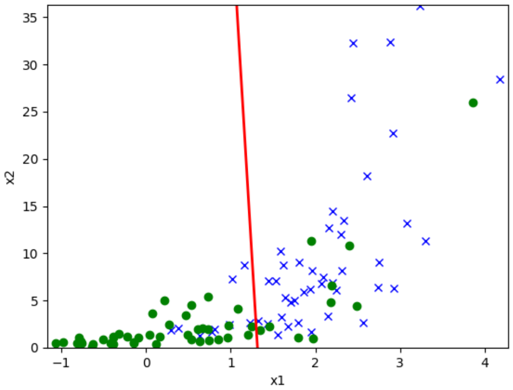

Figure 3: first

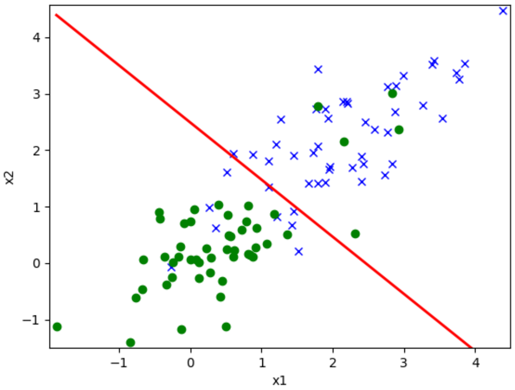

Figure 4: second

## 1.6 (f)

The decision boundary of logistic regression is much better. As we can see there are some misclassifications of both green and blue points in the GDA case.

## 1.7 (g)

similar to (f)

## 1.8 (h)

As suggested by chatGPT, we can use the following transformations o get a better decision boundary.

1. Standardization: Standardize the input features by subtracting the mean and dividing by the standard deviation. This can make the Gaussian distributions more comparable and easier for GDA to model.
2. Log transformation: If the input data is highly skewed, applying a log transformation can help reduce the skewness and make the distribution more symmetric.
3. Polynomial transformations: You can introduce higher-order polynomial features or interaction terms to capture non-linear relationships between input features.
4. Principal Component Analysis (PCA): Perform PCA to transform the input data into a new set of orthogonal features that capture most of the variance in the data. This can help reduce the dimensionality and make it easier for GDA to model the underlying distributions.
5. Whitening: Transform the input data to have zero mean and identity covariance matrix. This can be achieved by first standardizing the data and then applying PCA followed by scaling the principal components to have unit variance.