CS6140: Machine Learning **Ian Magnusson - magnusson.i@husky.neu.edu**

Homework Assignment # 2

Assigned: 10/05/2019 *Due: 10/17/2019, 11:59pm, through Blackboard*

Five problems, 110 points in total. Good luck!
Prof. Predrag Radivojac, Northeastern University

Problem 1. (10 points) Suppose that the number of accidents occurring daily in a certain plant has a Poisson distribution with an unknown mean λ. Based on previous experience in similar industrial plants, suppose that our initial feelings about the possible value of λ can be expressed by an exponential distribution with parameter $\theta = \frac{1}{2}$ is, the prior density is

$$p(\lambda) = \theta e^{-\theta\lambda}$$

where $\lambda \in (0, \infty)$. If there are 88 accidents over the next 9 days, determine the Bayes estimate of λ.

If we take the loss function to be squared error:

$$\begin{aligned}\lambda_B &= E[\Lambda | D = \mathcal{D}] \\ &= \int_\Lambda \lambda \cdot p(\lambda|\mathcal{D}) d\lambda \\ &= \int_\Lambda \lambda \frac{p(\mathcal{D}|\lambda)p(\lambda)}{\int_\Lambda p(\mathcal{D}|\lambda)p(\lambda)d\lambda} d\lambda\end{aligned}$$

From the specifications of the question:

$$p(\lambda) = p(\lambda) = \theta e^{-\theta\lambda}$$

$$\mathcal{D} = \{x_i\}_{i=1}^n : x \in \mathbb{Z}^+, n = 9, \sum_{i=1}^n x_i = 88$$

$$p(\mathcal{D}|\lambda) = \prod_{i=1}^n \frac{\lambda^{x_i} e^{-\lambda}}{x_i!} = \frac{\lambda^{\sum_{i=1}^n x_i} e^{-n\lambda}}{\prod_{i=1}^n x_i!}$$

$$\int_\Lambda p(\mathcal{D}|\lambda)p(\lambda)d\lambda = \int_\Lambda \frac{\lambda^{\sum_{i=1}^n x_i} e^{-n\lambda}}{\prod_{i=1}^n x_i!} \cdot \theta e^{-\theta\lambda} d\lambda = \int_\Lambda \lambda^{\sum_{i=1}^n x_i} e^{-\lambda(n+\theta)} \cdot \frac{\theta}{\prod_{i=1}^n x_i!} d\lambda$$

Because $\int_0^\infty x^{\alpha-1} e^{-\beta x} dx = \frac{\Gamma(\alpha)}{\beta^\alpha}$:

$$\int_\Lambda p(\mathcal{D}|\lambda)p(\lambda)d\lambda = \frac{\Gamma(1 + \sum_{i=1}^n x_i)}{(n+\theta)^{1+\sum_{i=1}^n x_i}} \cdot \frac{\theta}{\prod_{i=1}^n x_i!}$$

$$\int_\Lambda \lambda \frac{p(\mathcal{D}|\lambda)p(\lambda)}{\int_\Lambda p(\mathcal{D}|\lambda)p(\lambda)d\lambda}d\lambda = \frac{(n+\theta)^{1+\sum_{i=1}^n x_i}\prod_{i=1}^n x_i!}{\theta\Gamma(1+\sum_{i=1}^n x_i)}\int_\Lambda \lambda p(\mathcal{D}|\lambda)p(\lambda)d\lambda$$
$$= \frac{(n+\theta)^{1+\sum_{i=1}^n x_i}\prod_{i=1}^n x_i!}{\theta\Gamma(1+\sum_{i=1}^n x_i)} \cdot \frac{\theta\Gamma(2+\sum_{i=1}^n x_i)}{(n+\theta)^{2+\sum_{i=1}^n x_i}\prod_{i=1}^n x_i!}$$

$$\lambda_B = \frac{1+\sum_{i=1}^n x_i}{n+\theta}$$

Plugging in the values in this specific case:

$$\lambda_B = \frac{1+88}{9+\frac{1}{2}} = \frac{178}{19}$$

Problem 2. (15 points) Let X_1, X_2, ..., X_n be i.i.d. Gaussian random variables, each having an unknown mean θ and known variance σ_0^2. If θ is itself selected from a normal population having a known mean μ and a known variance σ^2, determine

a) (5 points) the maximum a posteriori estimate of θ

Let $D = \{X_i\}_{i=1}^n$:

$$\theta_{MAP} = \arg\max_\theta\{p(\mathcal{D}|\theta)p(\theta)\} = \arg\max_\theta\{\prod_{i=1}^n p(x_i|\theta)p(\theta)\}$$

We can maximize the log of the desired probability:

$$\log p(\theta) = \log(\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\theta-\mu)^2}{2\sigma^2}}) = log(\frac{1}{\sqrt{2\pi\sigma^2}}) - \frac{(\theta-\mu)^2}{2\sigma^2}$$

$$\log\prod_{i=1}^n p(x_i|\theta) = n\log(\frac{1}{\sqrt{2\pi\sigma_0^2}}) - \sum_{i=1}^n \frac{(x_i-\theta)^2}{2\sigma_0^2}$$

$$\frac{\partial p(\mathcal{D}|\theta)p(\theta)}{\partial\theta} = -\frac{2(\theta-\mu)}{2\sigma^2} - \sum_{i=1}^n \frac{-2(x_i-\theta)}{2\sigma_0^2} = -\frac{\theta}{\sigma^2} - n\frac{\theta}{\sigma_0^2} + \frac{\mu}{\sigma^2} + \sum_{i=1}^n \frac{x_i}{\sigma_0^2}$$
$$= 0$$
$$\theta_{MAP} = \frac{\mu\sigma_0^2 + \sum_{i=1}^n x_i\sigma^2}{\sigma_0^2 + n\sigma^2}$$

b) (10 points) the Bayes estimate of θ. Taking the loss function to be squared error:

$$\theta_B = \int_\Theta \theta\frac{p(\mathcal{D}|\theta)p(\theta)}{\int_\Theta p(\mathcal{D}|\theta)p(\theta)d\theta}d\theta$$

$$p(\mathcal{D}|\theta)p(\theta) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\theta-\mu)^2}{2\sigma^2}} \cdot \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma_0^2}}e^{-\frac{(x_i-\theta)^2}{2\sigma_0^2}}$$

$$\propto e^{-\frac{(\theta-\mu)^2}{2\sigma^2} - \sum_{i=1}^n \frac{(x_i-\theta)^2}{2\sigma_0^2}}$$

$$= e^{-\frac{1}{2}(\frac{1}{\sigma^2}\theta^2 - \frac{2}{\sigma^2}\theta\mu + \frac{1}{\sigma^2}\mu^2 + \frac{1}{\sigma_0^2}\sum_{i=1}^n x_i^2 - \frac{2}{\sigma_0^2}\sum_{i=1}^n \theta x_i + \frac{n}{\sigma_0^2}\theta^2)}$$

$$\propto e^{-\frac{1}{2}((\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2})\theta^2 - 2(\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2})\theta)}$$

$$= e^{-\frac{(\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2})}{2}(\theta^2 - 2(\frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}})\theta}$$

$$= e^{-\frac{(\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2})}{2}(\theta^2 - 2(\frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}})\theta + \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}}^2 - \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}}^2)}$$

$$\propto e^{-\frac{(\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2})}{2}(\theta^2 - 2(\frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}})\theta + \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}}^2)}$$

$$\propto e^{-\frac{(\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2})}{2}(\theta - \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2}+\frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2}+\frac{1}{\sigma^2}})^2}$$

Let $\bar{\Theta} \sim \mathcal{N}(\bar{\mu}, \bar{\sigma^2})$ be a new Gaussian with:

$$\bar{\mu} = \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2} + \frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2} + \frac{1}{\sigma^2}}$$

Thus for some constant c that does not involve θ:

$$\bar{\Theta} \sim \frac{1}{c} \cdot p(\mathcal{D}|\theta)p(\theta)$$

$$\int_\Theta \theta \frac{p(\mathcal{D}|\theta)p(\theta)}{\int_\Theta p(\mathcal{D}|\theta)p(\theta)d\theta}d\theta = \int_\Theta \theta \frac{P(\bar{\Theta}=\theta)}{\int_\Theta P(\bar{\Theta}=\theta)d\theta}d\theta$$

$$= E_{\bar{\Theta}}[\frac{\theta}{E_{\bar{\Theta}}[1]}] = E_{\bar{\Theta}}[\theta] = \bar{\mu}$$

$$\theta_B = \frac{\frac{\sum_{i=1}^n x_i}{\sigma_0^2} + \frac{\mu}{\sigma^2}}{\frac{n}{\sigma_0^2} + \frac{1}{\sigma^2}}$$

Problem 3. (40 points) Expectation-Maximization (EM) algorithm. Let X be a random variable distributed according to $p_X(x)$ and Y be a random variable distributed according to $p_Y(y)$. Let $\mathcal{D}_X = \{x_i\}_{i=1}^m$ be an i.i.d sample from $p_X(x)$ and $\mathcal{D}_Y = \{y_i\}_{i=1}^n$ be an i.i.d. sample from $p_Y(y)$. Let $\mathcal{D} = \mathcal{D}_X \cup \mathcal{D}_Y$. Finally, let $p_X(x)$ and $p_Y(y)$ be defined as follows

$$p_X(x) = \alpha\mathcal{N}(\mu_1, \sigma_1^2) + (1-\alpha)\mathcal{N}(\mu_2, \sigma_2^2)$$

and

$$p_Y(y) = \beta\mathcal{N}(\mu_1, \sigma_1^2) + (1-\beta)\mathcal{N}(\mu_2, \sigma_2^2),$$

where $\mathcal{N}(\mu, \sigma^2)$ is a univariate Gaussian distribution with mean μ and variance σ^2, $\alpha \in (0,1)$, $\beta \in (0,1)$, $\mu_1 \in \mathbb{R}$, $\mu_2 \in \mathbb{R}$, $\sigma_1 \in \mathbb{R}^+$ and $\sigma_2 \in \mathbb{R}^+$ are unknown parameters.

a) (5 points) Derive update rules of the EM algorithm for estimating β, μ_1, μ_2, σ_1, and σ_2 based only on data set $\mathcal{D}_Y$.

Let $\mathcal{V} = \{V_i \in \{0,1\}\}_{i=1}^m$ be the labels of the source Gaussian of the points in $\mathcal{D}_Y$.

$$\begin{aligned} p(y_i, v_i) &= p(y_i|v_i) \cdot p(v_i) \\ &= \mathcal{N}(\mu_1, \sigma_1^2)^{v_i}\mathcal{N}(\mu_2, \sigma_2^2)^{1-v_i} \cdot (\beta)^{v_i}(1-\beta)^{v_i} \\ &= (\beta)\mathcal{N}(\mu_1, \sigma_1^2)^{v_i}(1-\beta)\mathcal{N}(\mu_2, \sigma_2^2)^{1-v_i} \end{aligned}$$

$$\log p(D_y, \mathbf{V}) = \sum_{i=1}^{n} v_i(\log(\beta) + \log\mathcal{N}(\mu_1, \sigma_1^2)) + (1-v_i)(\log(1-\beta) + \log\mathcal{N}(\mu_2, \sigma_2^2))$$

Following the derivation on page 60 of the lecture notes, with $\bar{v_1} = p_v(1|y_i, \theta^{(t)})$ and $\bar{v_2} = p_v(0|y_i, \theta^{(t)})$:

$$E[\log p(D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}] = \sum_{i=1}^{n} \bar{v_1}(\log(\beta) + \log\mathcal{N}(\mu_1, \sigma_1^2)) + \bar{v_2}(\log(1-\beta) + \log\mathcal{N}(\mu_2, \sigma_2^2))$$

From this we can derive the following update rules:

$$\begin{aligned} \frac{\partial E[\log p(D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \mu_i} &= \sum_{i=1}^{n} \bar{v_i}(\frac{-2}{2\sigma_i^2}(y_i - \mu_i)) \\ &= 0 \\ \bar{\mu_i} &= \frac{\sum_{i=1}^n \bar{v_i} y_i}{\sum_{i=1}^n \bar{v_i}} \end{aligned}$$

$$\begin{aligned} \frac{\partial E[\log p(D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \sigma_i} &= \sum_{i=1}^{n} \bar{v_i}(-\frac{1}{2}(\sigma_i^2)^{-1} + \frac{1}{2}(y_i - \mu_i)^2(\sigma_i^2)^{-2}) \\ &= 0 \\ \bar{\sigma_i^2} &= \frac{\sum_{i=1}^n \bar{v_i}(y_i - \bar{\mu_i})^2}{\sum_{i=1}^n \bar{v_i}} \end{aligned}$$

$$\begin{aligned} \frac{\partial E[\log p(D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \beta} &= \sum_{i=1}^{n} \{\bar{v_1}\frac{1}{\beta} - (1-\bar{v_1})\frac{1}{1-\beta}\} \\ &= 0 \end{aligned}$$

$$\begin{aligned} \frac{1-\beta}{\beta} &= \frac{\sum_{i=1}^n (1-\bar{v_1})}{\sum_{i=1}^n \bar{v_1}} \\ \frac{1}{\beta} - 1 &= \frac{n}{\sum_{i=1}^n \bar{v_1}} - 1 \\ \bar{\beta} &= \frac{\sum_{i=1}^n \bar{v_1}}{n} \end{aligned}$$

b) (15 points) Derive update rules of an EM algorithm for estimating α, β, μ_1, μ_2, σ_1, and σ_2 based only on data set $\mathcal{D}$.

Let $\mathcal{U} = \{u_i \in \{0,1\}\}_{i=1}^m$ and $\mathcal{V} = \{v_i \in \{0,1\}\}_{i=1}^n$ be the labels of the source Gaussian of the points in $\mathcal{D}_X$ and $\mathcal{D}_Y$ respectively.

Because the samples in D_x, D_y are i.i.d.:

$$p(D_x, \mathbf{U}, D_y, \mathbf{V}) = p(D_x, \mathbf{U})p(D_y, \mathbf{V})$$

With $\bar{v_i}$ as before and $\bar{u_1} = p_u(1|x_i, \theta^{(t)})$ and $\bar{u_2} = p_u(0|x_i, \theta^{(t)})$:, We get that

$$E[\log p(D_x, \mathbf{U}, D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}] = E[\log p(D_x, \mathbf{U}) + \log P(D_y, \mathbf{V})|\theta)|\mathcal{D}, \theta^{(t)}]$$

$$= \sum_{i=1}^m \bar{u_1}(\log(\alpha\mathcal{N}(\mu_1, \sigma_1^2))) + \bar{u_2}(\log((1-\alpha)\mathcal{N}(\mu_2, \sigma_2^2))) + \sum_{i=1}^n \bar{v_1}(\log(\beta\mathcal{N}(\mu_1, \sigma_1^2))) + \bar{v_2}(\log((1-\beta)\mathcal{N}(\mu_2, \sigma_2^2)))$$

From this we can first see that the update rule for β will remain the same because is not involved in any new terms. The rest of update rules can be derived as follows:

$$\frac{\partial E[\log p(D_x, \mathbf{U}, D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \alpha} = \sum_{i=1}^m \{\bar{u_1}\frac{1}{\alpha} - (1-\bar{u_1})\frac{1}{1-\alpha}\}$$

$$= 0$$

$$\frac{1-\alpha}{\alpha} = \frac{\sum_{i=1}^m (1-\bar{u_1})}{\sum_{i=1}^m \bar{u_1}}$$

$$\frac{1}{\alpha} - 1 = \frac{n}{\sum_{i=1}^m \bar{u_1}} - 1$$

$$\bar{\alpha} = \frac{\sum_{i=1}^m \bar{u_1}}{n}$$

$$\frac{\partial E[\log p(D_x, \mathbf{U}, D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \mu_i} = \sum_{i=1}^m \bar{u_i}(\frac{-2}{2\sigma_i^2}(x_i - \mu_i)) + \sum_{i=1}^n \bar{v_i}(\frac{-2}{2\sigma_i^2}(y_i - \mu_i))$$

$$= 0$$

$$\bar{\mu_i} = \frac{\sum_{i=1}^m \bar{u_i}x_i + \sum_{i=1}^n \bar{v_i}y_i}{\sum_{i=1}^m \bar{u_i} + \sum_{i=1}^n \bar{v_i}}$$

$$\frac{\partial E[\log p(D_x, \mathbf{U}, D_y, \mathbf{V}|\theta)|\mathcal{D}, \theta^{(t)}]}{\partial \sigma_i} = \sum_{i=1}^m \bar{u_i}(-\frac{1}{2}(\sigma_i^2)^{-1} + \frac{1}{2}(x_i - \mu_i)^2(\sigma_i^2)^{-2}) + \sum_{i=1}^n \bar{v_i}(-\frac{1}{2}(\sigma_i^2)^{-1} + \frac{1}{2}(y_i - \mu_i)^2(\sigma_i^2)^{-2})$$

$$= 0$$

$$\bar{\sigma_i^2} = \frac{\sum_{i=1}^m \bar{u_i}(x_i - \bar{\mu_i})^2 + \sum_{i=1}^n \bar{v_i}(y_i - \bar{\mu_i})^2}{\sum_{i=1}^m \bar{u_i} + \sum_{i=1}^n \bar{v_i}}$$

c) (20 points) Implement both learning algorithms from above and evaluate them on simulated data when $m, n = 100$ and $m, n = 1000$. However, in each case repeat the experiment $B = 1000$ times to estimate the expectation and variance of all parameters. To do so, repeatedly draw samples $\mathcal{D}_X$ and $\mathcal{D}_Y$ and then estimate the parameters based on these samples. Finally, average those B estimates and calculate their mean and variance. Document all experiments and discuss your findings.

```
## Evaluation 1: m, n = 100

Notes: The full data noticeably improves the accuracy and
stability of estimates compared to the partial data.
This makes intuitive sense because there is effectively
twice as much data. Also note the reduced variance for the
second component in the full data compared to the partial.
This may be due to the greater amount of data or also
because the partial is only relying on information with a
0.8 weight towards component with the high variance estimates.

MEANS = [4.5, -3.5]
STANDARD DEVIATIONS = [2.5, 0.25]
ALPHA = 0.4
BETA = 0.8
B = 1000
M, n = 100

****** partial data results *******
mu means: [4.518687636103337, -3.1753173486756516]
mu variances: [0.09759593158588642, 1.5160403709240224]
sigma means: [2.4167123651965134, 0.5129393002691209]
sigma variances: [0.07882953834030851, 1.0431727833632138]
beta means: 0.7759519944542738
beta variances: 0.008465485893130162


****** full data results *******
mu means: [4.494356789973491, -3.4998645619710405]
mu variances: [0.054388004387335084, 0.0007931919354386998]
sigma means: [2.4870322537326452, 0.24662409526749804]
sigma variances: [0.026189999757353682, 0.0004046661196388699]
alpha means: 0.399340781916612
alpha variances: 0.0022573046382196546
beta means: 0.7996477367430143
beta variances: 0.0014888736600144903


## Evaluation 2: m, n = 1000
Notes: This larger trial further supports the conclusions
from the previous evaluation. It also shows greater accuracy
and stability for both the partial and full data compared
to the previous trial. This shows that larger datasets are
```

```
are also favorable for the partial training algorithm. This
does not appear to make up for the issue with the heavily
weighted component in the partial data having higher variance
in its estimates.

MEANS = [4.5, -3.5]
STANDARD DEVIATIONS = [2.5, 0.25]
ALPHA = 0.4
BETA = 0.8
B = 1000
M, n = 100

****** partial data results *******
mu means: [4.505557136097958, -3.4636827797896963]
mu variances: [0.011362691783113754, 0.13912564762522836]
sigma means: [2.494169253688083, 0.2841100952422203]
sigma variances: [0.006578007033815644, 0.11939745619322767]
beta means: 0.7977877224521351
beta variances: 0.0008904898126709888


****** full data results *******
mu means: [4.498472861668103, -3.499702236813232]
mu variances: [0.005068899391565506, 8.677535465332804e-05]
sigma means: [2.498883502715743, 0.24987747960622902]
sigma variances: [0.0030393496190142507, 3.9106211298193916e-05]
alpha means: 0.3995532322863421
alpha variances: 0.0002470342794184324
beta means: 0.800458343648992
beta variances: 0.0001555979483667243
```

NOTE: For further results and discussion see EM.ipynb

To implement the EM algorithm you must make some decisions regarding stopping the estimation. You may decide to impose the maximum number of steps or stop the algorithm when the updated parameters stabilize or when the log-likelihood stabilizes. In either case, experiment first with the stoppage criterion and once it is fixed carry out the experiment.

Problem 4. (20 points) Naive Bayes classifier. Consider a binary classification problem where there are only four data points in the training set. That is $\mathcal{D} = \{(0, 0, -), (0, 1, +), (1, 0, +), (1, 1, -)\}$, where each tuple (x_1, x_2, y) represents a training example with input vector (x_1, x_2) and class label y.

a) (10 points) Construct a naive Bayes classifier for this problem and evaluate its accuracy on the training set.

$$\hat{y} = \arg\max_{y \in Y} p(\mathbf{x}|y)p(y) = \arg\max_{y \in Y} p(y) \prod_{i=1}^{d} p(x_i|y)$$

$$p_Y(-) = p_Y(+) = \frac{1}{2}$$

$$p((0,0)|+) = \frac{count(x_1 = 0 \wedge y = +)}{count(y = +)} \frac{count(x_2 = 0 \wedge y = +)}{count(y = +)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((0,0)|-) = \frac{count(x_1 = 0 \wedge y = -)}{count(y = -)} \frac{count(x_2 = 0 \wedge y = -)}{count(y = -)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

Thus for this data point either classification is equally likely and we must resort to arbitrary selection. For the next 3 data points we see the same pattern:

$$p((0,1)|+) = \frac{count(x_1 = 0 \wedge y = +)}{count(y = +)} \frac{count(x_2 = 1 \wedge y = +)}{count(y = +)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((0,1)|-) = \frac{count(x_1 = 0 \wedge y = -)}{count(y = -)} \frac{count(x_2 = 1 \wedge y = -)}{count(y = -)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((1,0)|+) = \frac{count(x_1 = 1 \wedge y = +)}{count(y = +)} \frac{count(x_2 = 0 \wedge y = +)}{count(y = +)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((1,0)|-) = \frac{count(x_1 = 1 \wedge y = -)}{count(y = -)} \frac{count(x_2 = 0 \wedge y = -)}{count(y = -)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((1,1)|+) = \frac{count(x_1 = 1 \wedge y = +)}{count(y = +)} \frac{count(x_2 = 1 \wedge y = +)}{count(y = +)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

$$p((1,1)|-) = \frac{count(x_1 = 1 \wedge y = -)}{count(y = -)} \frac{count(x_2 = 1 \wedge y = -)}{count(y = -)} = \frac{1}{2}\frac{1}{2} = \frac{1}{4}$$

So to conclude this classifier must choose its predictions arbitrarily for each of the data points. So following a default to one label or another in case of ties, this will always label two points correctly and two points incorrectly.

b) (10 points) Transform the input space into a six-dimensional space $(1, x_1, x_2, x_1x_2, x_1^2, x_2^2)$ and repeat the previous step.

$$\mathcal{D} = \{(1,0,0,0,0,0,-), (1,0,1,0,0,1,+), (1,1,0,0,1,0,+), (1,1,1,1,1,1,-)\}$$

let each tuple be $(z_1, z_2, z_3, z_4, z_5, z_6, y)$

$$p_{z_1}(1|+) = p_{z_1}(1|-) = 1$$

z_2 and z_3 are the same as x_1 and x_2 before and thus their class conditional probabilities are also still $\frac{1}{2}$.

The rest of the class conditional probabilities are as follows:

$$p_{z_4}(1|+) = \frac{0}{2}$$

$$p_{z_4}(0|+) = \frac{2}{2}$$

$$p_{z_4}(1|-) = \frac{1}{2}$$

$$p_{z_4}(0|-) = \frac{1}{2}$$

$$p_{z_5}(1|+) = \frac{1}{2}$$

$$p_{z_5}(0|+) = \frac{1}{2}$$

$$p_{z_5}(1|-) = \frac{1}{2}$$

$$p_{z_5}(0|-) = \frac{1}{2}$$

$$p_{z_6}(1|+) = \frac{1}{2}$$

$$p_{z_6}(0|+) = \frac{1}{2}$$

$$p_{z_6}(1|-) = \frac{1}{2}$$

$$p_{z_6}(0|-) = \frac{1}{2}$$

From this we can get find the likelihoods for each data point, and the maximum of these will also maximize the posterior because of the equal priors.

$$\mathcal{D} = \{(1,0,0,0,0,0,-), (1,0,1,0,0,1,+), (1,1,0,0,1,0,+), (1,1,1,1,1,1,-)\}$$

$p((1,0,0,0,0,0)|+) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{4}{64}$
$p((1,0,0,0,0,0)|-) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{2}{64}$
$p((1,0,1,0,0,1)|+) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{4}{64}$
$p((1,0,1,0,0,1)|-) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{2}{64}$
$p((1,1,0,0,1,0)|+) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{4}{64}$
$p((1,1,0,0,1,0)|-) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{2}{64}$
$p((1,1,1,1,1,1)|+) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{0}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{0}{64}$
$p((1,1,1,1,1,1)|-) = \frac{2}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot = \frac{2}{64}$

We can see that now the model no longer has to resort to choosing arbitrarily. it chooses the correct label for the last 3 data points but in miss labels the first one. So while this is an improvement, the model is still unable to learn an XOR operation and instead has learned a negated AND operation.

Carry out all steps manually and show all your calculations.

Problem 5. (25 points) Consider a binary classification problem in which we want to determine the optimal decision surface. A point $\mathbf{x}$ is on the decision surface if $P(Y = 1|\mathbf{x}) = P(Y = 0|\mathbf{x})$.

a) (10 points) Find the optimal decision surface assuming that each class-conditional distribution is defined as a two-dimensional Gaussian distribution:

$$p(\mathbf{x}|Y=i) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_i|^{1/2}} \cdot e^{-\frac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T\mathbf{\Sigma}_i^{-1}(\mathbf{x}-\mathbf{m}_i)}$$

where $i \in \{0,1\}$, $\mathbf{m}_0 = (1,2)$, $\mathbf{m}_1 = (6,3)$, $\mathbf{\Sigma}_0 = \mathbf{\Sigma}_1 = \mathbf{I}_2$, $P(Y=0) = P(Y=1) = 1/2$, $\mathbf{I}_d$ is the d-dimensional identity matrix, and $|\mathbf{\Sigma}_i|$ is the determinant of $\mathbf{\Sigma}_i$.

$$\frac{p(x|Y=1)p(Y=1)}{p(x|Y=1)p(Y=1)+p(x|Y=0)p(Y=0)} = \frac{p(x|Y=0)p(Y=0)}{p(x|Y=1)p(Y=1)+p(x|Y=0)p(Y=0)}$$

$$p(x|Y=1) = p(x|Y=0)$$

$$\frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_1|^{1/2}} \cdot e^{-\frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\mathbf{\Sigma}_1^{-1}(\mathbf{x}-\mathbf{m}_1)} = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_0|^{1/2}} \cdot e^{-\frac{1}{2}(\mathbf{x}-\mathbf{m}_0)^T\mathbf{\Sigma}_0^{-1}(\mathbf{x}-\mathbf{m}_0)}$$

$$(\mathbf{x}-\mathbf{m}_1)^T(\mathbf{x}-\mathbf{m}_1) = (\mathbf{x}-\mathbf{m}_0)^T(\mathbf{x}-\mathbf{m}_0)$$

$$(x_1-6)^2+(x_2-3)^2 = (x_1-1)^2+(x_2-2)^2$$

$$-10x_1 \cdot -2x_2 = -40$$

$$5x_1 \cdot x_2 = 20$$

b) (5 points) Generalize the solution from part (a) using $\mathbf{m}_0 = (m_{01}, m_{02})$, $\mathbf{m}_1 = (m_{11}, m_{12})$, $\mathbf{\Sigma}_0 = \mathbf{\Sigma}_1 = \sigma^2\mathbf{I}_2$ and $P(Y=0) \neq P(Y=1)$.

$$p(x|Y=1)p(Y=1) = p(x|Y=0)p(Y=0)$$

$$|\sigma^2\mathbf{I}_2| = (\sigma^2)^2$$

$$(\sigma^2\mathbf{I}_2)^{-1} = \frac{1}{\sigma^2}\mathbf{I}_2$$

$$\log(p(Y=1)) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\frac{1}{\sigma^2}\mathbf{I}_2(\mathbf{x}-\mathbf{m}_1) = \log(p(Y=0)) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_0)^T\frac{1}{\sigma^2}\mathbf{I}_2(\mathbf{x}-\mathbf{m}_0)$$

$$-2\sigma^2(\log(p(Y=1)) - \log(p(Y=0))) - ((x_1-m_{11})^2+(x_2-m_{12})^2) = (x_1-m_{01})^2+(x_2-m_{02})^2$$

$$x_1m_{01} - x_1m_{11} + x_2m_{02} - x_2m_{12} = \sigma^2(\log(p(Y=1)) - \log(p(Y=0))) + 1/2(m_{01}^2+m_{02}^2-m_{11}^2-m_{12}^2)$$

c) (10 points) Generalize the solution from part (b) to arbitrary covariance matrices $\mathbf{\Sigma}_0$ and $\mathbf{\Sigma}_1$. Discuss the shape of the optimal decision surface.

Let $|\Sigma_i|$ be some scalar z_i.

$$\log(p(Y=1)) - \frac{1}{2}\log(z_1) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\Sigma_i^{-1}(\mathbf{x}-\mathbf{m}_1) = \log(p(Y=0)) - \frac{1}{2}\log(z_0) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_0)^T\Sigma_i^{-1}(\mathbf{x}-\mathbf{m}_0)$$

$$\frac{1}{2}(\mathbf{x}-\mathbf{m}_0)^T\Sigma_0^{-1}(\mathbf{x}-\mathbf{m}_0) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\Sigma_1^{-1}(\mathbf{x}-\mathbf{m}_1) = \log(p(Y=0)) - \frac{1}{2}\log(z_0) - \log(p(Y=1)) + \frac{1}{2}\log(z_1)$$

Where $\frac{1}{2}(\mathbf{x}-\mathbf{m}_0)^T\Sigma_0^{-1}(\mathbf{x}-\mathbf{m}_0) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_1)^T\Sigma_1^{-1}(\mathbf{x}-\mathbf{m}_1)$ will produce terms with x_i^2 that may not cancel out. This is because, firstly, $\Sigma_i^{-1}(\mathbf{x}-\mathbf{m}_i)$ will produce a vector $\mathbf{v}$ where each v_k is a linear combination of all $x_j - m_i$ for dimension j of $\mathbf{x}$. And Secondly, $(\mathbf{x}-\mathbf{m}_1)^T\mathbf{v}$ is a sum of terms $v_k(x_j - m_i)$ where each term will contain some x_j^2 scaled by a value from Σ_1^{-i}. The rest of the terms where x_j values do not match will be linear. Finally, unlike the previous two problems, the quadratic terms will not necessarily cancel because because they are scaled by different Σ_1^{-i}. Thus the decision surface will be some quadratic equation.