Problem 1.

find the parameters that maximize

$$CL(\theta) = \frac{1}{N}\sum_{n=1}^{N} \log P(y^n | x^n, \theta)$$

where all (x^n, y^n) generated from $P(x,y|\theta^0) = P(y|x,\theta^0)P(x|\theta^0)$

Proof.

$$CL(\theta) = \frac{1}{N}\sum_{n=1}^{N} \log P(y^n | x^n, \theta)$$

$$= \langle \log P(y^n|x^n,\theta) \rangle_{p(data)} \text{ , where } p(data) = P(x^n, y^n|\theta^0)$$

$$= \langle \log P(y^n|x^n,\theta) \rangle_{P(x^n, y^n|\theta^0)} .$$

when the amount of training data is large enough.

$$CL(\theta) = \int p(x^n, y^n|\theta^0) \log P(y^n|x^n,\theta)\, dn$$

$$= \int \underbrace{P(y^n|x^n,\theta^0)}P(x^n|\theta^0)\underbrace{\log P(y^n|x^n,\theta)}\, dn$$

$$= -\int P(x^n|\theta^0)P(y^n|x^n,\theta^0)\log\frac{P(y^n|x^n,\theta^0)}{P(y^n|x^n,\theta)}\, dn$$

$$+ \int P(x^n|\theta^0)P(y^n|x^n,\theta^0)\log P(y^n|x^n,\theta^0)\, dn$$

$$= -\int P(x^n|\theta^0)\, KL(P(y^n|x^n,\theta^0)\,||\,P(y^n|x^n,\theta))\, dn + f(\theta^0)$$

Thus, to maximize $CL(\theta)$, $KL(P(y^n|x^n,\theta^0)\,||\,P(y^n|x^n,\theta)) = 0$.

which means $\theta = \theta^0$.

Therefore, $\boxed{CL(\theta) \text{ has an optimum at } \theta^0.}$ ∎

Problem 2.

Older than 60

C	F	S	B
1	0	0	0
1	0	0	1
1	1	1	1
0	0	0	1

younger than 60

C	F	S	B
0	1	1	0
1	1	1	0

New customer = 0110.

Proof. Naive Bayes assume all features to be independent

Thus, for older people: $P(C=1)=\frac{3}{4}$, $P(F=1)=\frac{1}{4}$, $P(S=1)=\frac{1}{4}$, $P(B=1)=\frac{3}{4}$ $\Big\}$ $P(old)=\frac{2}{3}$

for younger people: $P(C=1)=\frac{1}{2}$, $P(F=1)=1$, $P(S=1)=1$, $P(B=1)=0$ $\Big\}$ $P(y)=\frac{1}{3}$

Thus, the maximum likelihood will give us

$$P(new=old) \propto \frac{1}{4}\times\frac{1}{4}\times\frac{1}{4}\times\frac{1}{4}\times\frac{2}{3} = \frac{1}{16\times 8\times 3}$$

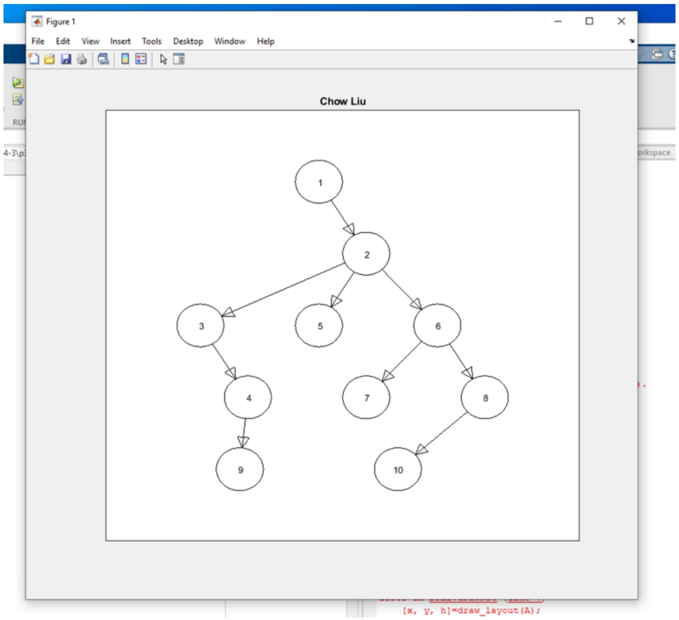
$$P(new=young) \propto \frac{1}{2}\times 1\times 1\times 1\times \frac{1}{3} = \frac{1}{6}$$

$$P(new=old)+P(new=young)=1$$

Thus, $P(young) = \dfrac{\frac{1}{6}}{\frac{1}{6}+\frac{1}{16\times 24}} = \dfrac{4\times 16}{4\times 16+1} = \boxed{\dfrac{64}{65}}$ $\boxtimes$

Problem 3

The function is inside p3ChowLiu.m, the main code is inside p3main.m, run the p3main.m to use the data and the function. From the main code we can get the following picture:



Problem 4.

NB classifier for $x_i \in \{0,1\}$ $\leftarrow \begin{cases} \theta_i^1 = P(x_i=1 \mid class=1) \\ \theta_i^0 = P(x_i=1 \mid class=0) \end{cases}$

Proof. $P(class=0 \mid \vec{x}) \propto P(class=0) P(\vec{x} \mid class=0)$

$$= p_0 \prod_i p(x_i \mid class=0)$$

$$= p_0 \prod_i (\theta_i^0)^{x_i} (1-\theta_i^0)^{1-x_i}$$

similarly,

$\hookrightarrow P(class=1 \mid \vec{x}) \propto p_1 \prod_i (\theta_i^1)^{x_i} (1-\theta_i^1)^{1-x_i}$

decision bound $P(class=0 \mid \vec{x}) = P(class=1 \mid \vec{x})$

$$p_0 \prod_i (\theta_i^0)^{x_i} (1-\theta_i^0)^{1-x_i} = p_1 \prod_i (\theta_i^1)^{x_i} (1-\theta_i^1)^{1-x_i}$$

$$\log p_0 + \sum_i [x_i \log \theta_i^0 + (1-x_i) \log(1-\theta_i^0)]$$
$$= \log p_1 + \sum_i [x_i \log \theta_i^1 + (1-x_i) \log(1-\theta_i^1)]$$

$$\sum_i x_i (\log \theta_i^0 - \log(1-\theta_i^0) - \log \theta_i^1 + \log(1-\theta_i^1))$$
$$+ \sum_i (\log(1-\theta_i^0) - \log(1-\theta_i^1)) + \log \frac{p_0}{p_1} = 0.$$

$$\sum_i x_i \underbrace{\left(\log \frac{\theta_i^0 (1-\theta_i^1)}{\theta_i^1 (1-\theta_i^0)}\right)}_{w_i} + \underbrace{\log \frac{p_0}{p_1} + \sum_i \log \frac{1-\theta_i^0}{1-\theta_i^1}}_{b} = 0.$$

Hence, the decision holds according to $w^T x + b = 0$, while the value of w^T and b are stated as above. $\blacksquare$

Problem 5

1. derive expressions using maximum likelihood.

we know that training data $= (x^n, c^n)$, $n \in \{1, \cdots, N\}$.

Thus, $p(c=1) = \frac{\text{number of data with } c=1}{N} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\{c^i = 1\}$

where $\mathbb{I}\{c^i = 1\} = \begin{cases} 0, & \text{when } c^i \neq 1 \\ 1, & \text{when } c^i = 1 \end{cases}$

$$p(x_i = 1 | c=1) = \frac{p(x_i=1, c=1)}{p(c=1)}$$

$$= \frac{\text{number of data with } x_i = 1 \text{ and } c=1}{\text{number of data with } c=1}$$

$$= \frac{\sum_{j=1}^{N} \mathbb{I}\{c^j = 1 \text{ and } x_i^j = 1\}}{\sum_{j=1}^{N} \mathbb{I}\{c^j = 1\}}$$

similarty, $p(x_i = 1 | c=0) = \frac{p(x_i=1, c=0)}{p(c=0)}$

$$= \frac{\sum_{j=1}^{N} \mathbb{I}\{c^j = 0 \text{ and } x_i^j = 1\}}{\sum_{j=1}^{N} \mathbb{I}\{c^j = 0\}}$$

$\blacksquare$

2. We know that $p(x,c) = p(x|c)p(c) = p(c|x)p(x)$

Thus, $p(c|\vec{x}) = \frac{p(\vec{x}, c)}{p(\vec{x})} = \frac{p(\vec{x}, c)}{p(\vec{x}, c=0) + p(\vec{x}, c=1)}$

$$= \frac{p(c) \prod_{i=1}^{N} p(x_i | c)}{\sum_{c=\{0,1\}} p(c) \prod_{i=1}^{N} p(x_i | c)}$$

$\blacksquare$

3、

① what effect: since this word never appear in training dataset

$\Rightarrow p(\text{"viagra"}=1|c) = 0.$

when doing classification, $p(c|x, \text{"viagra"}=1)$

$$= \frac{p(c)\prod_{i=1}^{N} p(x_i|c)}{\sum_{c=\{0,1\}} p(c)\prod_{i=1}^{N} p(x_i|c)} = \frac{0}{0+0}$$

which is impossible to deduce.

Also, in this case, $\underbrace{p(c=1|x,\text{"viagra"}=1) = p(c=0|x,\text{"viagra"}=1)}$

☆ Therefore, the appearance of a new word will mess up the performance of the classifier, and make the classifier ignore all other words and give an arbitrary prediction.

② how to counter this effect: $\boxed{\text{Laplace smoothing}}$

change the original $p(x_i|c) = \frac{\sum_{i=1}^{N} \mathbb{I}\{x_i \text{ and } c\}}{\sum_{i=1}^{N} \mathbb{I}\{c\}}$

into new $p_N(x_i|c) = \frac{\sum_{i=1}^{N} \mathbb{I}\{x_i \text{ and } c\}+1}{\sum_{i=1}^{N} \mathbb{I}\{c\}+2}$

Thus, when doing the training, the $p(x_i|c)$ for unseen words would be 50%, which will not effect the performance.

Also, another trick would be: replace all infrequent words into a mark "RARE", and calculate all new unseen words according to the possibility of "RARE".

③ How spammer fool NB filter:

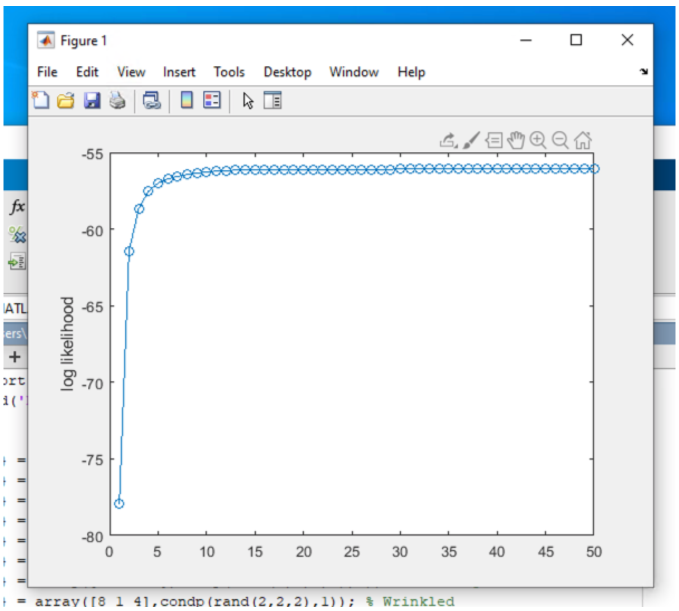
add a lot of non-spam-like words in the end of the spam email will reduce its chance of being recognized as spam.

Problem 6

The code is inside the file p6.m, we can get the result of

3.810283e-01
6.189717e-01

Is the probability of drum unit problem, where 0.619 is the possibility of there is a drum unit problem, and 0.381 is the possibility of there is no drum unit problem. From the plotprogress variable of the function EMbeliefnet, we can have the following plot which monitors when the function converges.



It converges well.