CS412: Introduction to Data Mining, Fall 2023, Midterm

Name: Teng Hou (tenghou2)

1 Q1

a. *To get the answer of* $P(S = T, M = F, G = T, V = T, A = F)$*, we could compute the answer of:*

$$P(S = T) \times P(M = F) \times P(G = T) \times P(V = T) \times P(A = F)$$

From the figure 1, we could get the answer for:

$$P(S = T) = 0.30$$

$$P(M = F) = 1 - P(M = T) = 1 - 0.40 = 0.60$$

$$P(G = T) = 0.60, (S = T, M = F)$$

$$P(V = T) = 0.75, (G = T)$$

$$P(A = F)1 - P(A = T) = 1 - 0.80 = 0.20, (G = T)$$

So the answer is

$$P = 0.30 * 0.60 * 0.60 * 0.75 * 0.20 = 0.0162$$

b. $P(G = T|S = T)$ *means that when S=T, the probability that G=T. One is that* $P(G = T) when S = T and M = T$ *One is that* $P(G = T) when S = T and M = F$ *Then we could get the final result:*

$$P(G = T|S = T) = P(G = T|S = T, M = T) + P(G = T|S = T, M = F)$$

$$P(G = T|S = T) = 0.40 * 0.80 + (1 - 0.40) * 0.60 = 0.68$$

c. $P(G = T|S = F)$ *means that when S=T, the probability that G=T. One is that* $P(G = T) when S = F and M = T$ *One is that* $P(G = F) when S = F and M = F$ *Then we could get the final result:*

$$P(G = T|S = F) = P(G = T|S = F, M = T) + P(G = F|S = F, M = F)$$

$$P(G = T|S = F) = 0.40 * 0.50 + (1 - 0.40) * 0.25 = 0.35$$

d. *For this question, the log-likelihood formula is:*

$$L(D|\theta) = \sum_{i=1}^{n} \log P(x^i|\theta)$$

And we could easily get this from the figure network:

$$P(x^i|\theta) = P(s^i) \times P(m^i) \times P(g^i|s^i, m^i) \times P(v^i|g^i) \times P(a^i|g^i)$$

Then by expanding, we could get

$$L(D|\theta) = \sum_{i=1}^{n} \left[\log P(s^i) + \log P(m^i) + \log P(g^i|s^i, m^i) + \log P(v^i|g^i) + \log P(a^i|g^i)\right]$$

2 Q2

1. *For Model M1:*

$$
\begin{aligned}
\text{a.Sensitivity (M1)} &= \frac{TP}{P} = \frac{a}{a+b} = \frac{300}{300+20} = 93.750\% \\
\text{b.Specificity (M1)} &= \frac{TN}{N} = \frac{d}{d+c} = \frac{11670}{11670+10} = 99.914\% \\
\text{c.Accuracy (M1)} &= \frac{TP+TN}{\text{ALL}} = \frac{a+d}{a+b+c+d} = \frac{300+11670}{300+20+10+11670} = 99.750\% \\
\text{d.Precision (M1)} &= \frac{TP}{TP+FP} = \frac{a}{a+c} = \frac{300}{300+10} = 96.774\% \\
\text{e.Recall (M1)} &= \frac{TP}{TP+FN} = \text{Sensitivity (M1)} = 93.750\% \\
\text{f.F1 Score (M1)} &= \frac{2 \cdot \text{Precision (M1)} \cdot \text{Recall (M1)}}{\text{Precision (M1)} + \text{Recall (M1)}} \\
&= 2 \cdot \frac{0.968 \cdot 0.938}{0.968 + 0.938} = 95.238\%
\end{aligned}
$$

2. *For Model M2:*

$$
\begin{aligned}
\text{a.Sensitivity (M2)} &= \frac{TP}{P} = \frac{a}{a+b} = \frac{320}{320+1} = 99.688\% \\
\text{b.Specificity (M2)} &= \frac{TN}{N} = \frac{d}{d+c} = \frac{11677}{11677+2} = 99.982\% \\
\text{c.Accuracy (M2)} &= \frac{TP+TN}{\text{ALL}} = \frac{a+d}{a+b+c+d} = \frac{320+11677}{320+1+2+11677} = 99.975\% \\
\text{d.Precision (M2)} &= \frac{TP}{TP+FP} = \frac{a}{a+c} = \frac{320}{320+2} = 99.379\% \\
\text{e.Recall (M2)} &= \frac{TP}{TP+FN} = \text{Sensitivity (M2)} = 99.688\% \\
\text{f.F1 Score (M2)} &= \frac{2 \cdot \text{Precision (M2)} \cdot \text{Recall (M2)}}{\text{Precision (M2)} + \text{Recall (M2}} \\
&= 2 \cdot \frac{0.9938 \cdot 0.9969}{0.9938 + 0.9969} = 99.533\%
\end{aligned}
$$

3 Q3

a. *As we know that to calculate the degree of freedom of k-fold cross-validation is* $k-1$. *And here we know that k=10, so the degrees of freedom is 10-1=9.*

$$DOF = 9$$

b. *We could use equation below to calculate t:*

$$t = \frac{\overline{err}(M1) - \overline{err}(M2)}{\sqrt{\frac{var(M1-M2)}{k}}}$$

and the equation below:

$$var(M1-M2) = \frac{1}{k}\sum_{i=1}^{10}[err(M1)_i - err(M2)_i - (err(M1) - err(M2))]^2$$

And from the question, we know that the two datasets are

$$err1 = [0.092, 0.038, 0.122, 0.044, 0.061, 0.045, 0.056, 0.067, 0.119, 0.051]$$

$$err2 = [0.551, 0.415, 0.619, 0.567, 0.525, 0.57, 0.48, 0.41, 0.435, 0.557]$$

And to calculate the mean value, we could use:

$$\overline{err}(M1) = \frac{\Sigma_{i=1}^{10} err1_i}{10}$$

$$\overline{err}(M2) = \frac{\Sigma_{i=1}^{10} err2_i}{10}$$

So

$$\overline{err}(M1) = \frac{\Sigma_{i=1}^{10} err1_i}{10} = 0.0695$$

$$\overline{err}(M2) = \frac{\Sigma_{i=1}^{10} err2_i}{10} = 0.5129$$

And as

$$var(M1-M2) = \frac{1}{k}\sum_{i=1}^{10}[err(M1)_i - err(M2)_i - (err(M1) - err(M2))]^2 = 0.005155$$

Then

$$t = \frac{\overline{err}(M1) - \overline{err}(M2)}{\sqrt{\frac{var(M1-M2)}{k}}} = \frac{0.0695 - 0.5129}{\sqrt{\frac{0.005155}{10}}} = -19.529$$

From Python script:

$$p = 1.120 \times 10^{-8}$$

The p-value is smaller than 0.05, so we could reject the null hypothesis, Which means that there is a significant difference between the two models.

c. *We could use equation below to calculate t:*

$$t = \frac{\overline{err}(M1) - \overline{err}(M2)}{\sqrt{\frac{var(M1-M2)}{k}}}$$

and the equation below:

$$var(M1 - M2) = \frac{1}{k}\sum_{i=1}^{10}[err(M1)_i - err(M2)_i - (err(M1) - err(M2))]^2$$

And from the question, we know that the two datasets are

$$err1 = [0.092, 0.038, 0.122, 0.044, 0.061, 0.045, 0.056, 0.067, 0.119, 0.051]$$

$$err2 = [0.032, 0, 0.05, 0.006, 0.011, 0.011, 0, 0.006, 0.034, 0.034]$$

And to calculate the mean value, we could use:

$$\overline{err}(M1) = \frac{\Sigma_{i=1}^{10} err1_i}{10}$$

$$\overline{err}(M2) = \frac{\Sigma_{i=1}^{10} err2_i}{10}$$

So

$$\overline{err}(M1) = \frac{\Sigma_{i=1}^{10} err1_i}{10} = 0.0695$$

$$\overline{err}(M2) = \frac{\Sigma_{i=1}^{10} err2_i}{10} = 0.0184$$

And as

$$var(M1 - M2) = \frac{1}{k}\sum_{i=1}^{10}[err(M1)_i - err(M2)_i - (err(M1) - err(M2))]^2 = 0.00036$$

Then

$$t = \frac{\overline{err}(M1) - \overline{err}(M2)}{\sqrt{\frac{var(M1-M2)}{k}}} = \frac{0.0695 - 0.0184}{\sqrt{\frac{0.00036}{10}}} = 8.5322$$

From Python script:

$$p = 1.3184 \times 10^{-5}$$

The p-value is smaller than 0.05, so we could reject the null hypothesis, Which means that there is a significant difference between the two models.

4 Q4

a. The sigmoid function is defined as:

$$\sigma(z) = \frac{1}{1+\exp(-z)}$$

The 2-class cross-entropy loss for a true label y and predicted probability p is:

$$\ell(y,\hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$$

Given the model $f_\theta(x)$, which is:

$$f_\theta(x) = \sigma(\theta^T x) = \frac{1}{1+\exp(-\theta^T x)}$$

The loss for a single data point (x_i, y_i) is:

$$\ell(y_i, f_\theta(x_i)) = -y_i \log\left(\frac{1}{1+\exp(-\theta^T x_i)}\right) - (1-y_i)\log\left(1 - \frac{1}{1+\exp(-\theta^T x_i)}\right)$$

Therefore, the empirical loss $L(\theta)$ on the training set, which is the average loss over all training examples, is:

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left[-y_i \log\left(\frac{1}{1+\exp(-\theta^T x_i)}\right) - (1-y_i)\log\left(1 - \frac{1}{1+\exp(-\theta^T x_i)}\right)\right]$$

b. likelihood of label $y \in \pi^y(1-\pi)^{1-y}$ where $\pi = \sigma(x^T w)$
Maximize likelihood of training set w.r.t W:

$$max\prod_{i=1}^{n} p_w(y|x)$$

Taking log of $\prod_{i=1}^{n} p_w(y|x)$, it became to

$$\sum_{i=1}^{n}\left[y_i \log\left(\frac{1}{1+\exp(-\theta^T x_i)}\right) + (1-y_i)\log\left(1 - \frac{1}{1+\exp(-\theta^T x_i)}\right)\right]$$

So by observation, the difference between this and loss function in a). is a negative sign and $\frac{1}{n}$. So when $max\prod_{i=1}^{n} p_w(y|x)$ reached the max value, the loss function in a). will be the minimum value.
c. KL Divergence: The KL divergence for two distributions $p(x)$ and $q(x)$ is given by:

$$D_{kl}(p(x)||q(x)) = \sum_{x\in X} p(x)\ln\frac{p(x)}{q(x)}$$

For the Bernoulli distributions $Bern(p^i)$ and $Bern(\hat{p}^i)$, the divergence is:

$$KL(Bern(p_i)||Bern(\hat{p}_i)) = y^i\log\left(\frac{y^i}{\hat{y}^i}\right) + (1-y^i)\log\left(\frac{1-y^i}{1-\hat{y}^i}\right)$$

Given the 2-class cross-entropy loss from problem (a):

$$L(\theta) = \left[y^i \log(\frac{1}{1+exp(-\theta^T\mathbf{x^i})}) + (1-y^i)\log(1-\frac{1}{1+exp(-\theta^T\mathbf{x^i})}) \right]$$
$$= \left[y^i \log(\hat{y}^i) + (1-y^i)\log(1-\hat{y}^i) \right]$$

And from a). we know that

$$\ell(y,\hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$$

The difference between the KL divergence and the loss is then:

$$KL(Bern(p_i), Bern(\hat{p}_i)) - \ell(y^i,\hat{y}^i) = y^i \log y^i + (1-y^i)\log(1-y^i)$$

d. When the features are one-dimensional and the dataset is linearly separable, the optimal solution without any constraints can lead to infinite weights in a logistic regression model. This is because the model will try to push the decision boundary infinitely far from the training examples to perfectly classify them. This is because only when $|\theta^*| \to \infty$ all the x_i could be classified into two groups.(In question a. For sigmoid funciton, two groups are 0 and 1) Therefore, Professor Astral's claim that $|\theta^*| \to \infty$ is correct. The intuition is that as $|\theta|$ grows, the output of $\sigma(\theta^T x)$ gets closer to either 0 or 1, which can perfectly classify linearly separable data.