Gyujin Park

CSE 6740 HOMEWORK #1

1. Questions

Question 1. 1.

$$P(I=n)=\frac{1}{2^n}$$

If $I=n$, $H:e^{-n},T:1-e^{-n}$

$$\begin{aligned}P(H)&=\sum_{n=1}^{\infty}\frac{1}{2^n}e^{-n}\\&=\sum_{n=1}^{\infty}(\frac{1}{2e})^n\\&=\frac{1}{2e-1}\end{aligned}$$

2.

$$P=\frac{15\%*95\%}{15\%*95\%+85\%*10\%}=62.6\%$$

3.(a) Let $P(walks\ on\ nth\ day)=w_n$

$P(walk\ n)=P(walk\ n-1)*\frac{1}{2}+P(bus\ n-1)*\frac{1}{6}$

$$w_n=\frac{1}{2}w_{n-1}+\frac{1}{6}(1-w_{n-1}),w_1=p$$

We can easily solve it.

$$w_n=\frac{1}{4}3^{-n}\left(3^n+12p-3\right)$$

3.(b) Let $P(late\ on\ nth\ day)=l_n$

$$\begin{aligned}P(late\ n)&=P(walk\ n)*\frac{1}{2}+P(bus\ n)*\frac{1}{6}\\&=\frac{1}{2}*w_n+\frac{1}{6}*(1-w_n)\\&=\frac{1}{12}3^{-n}\left(3^{n+1}+12p-3\right)\end{aligned}$$

Question 2. (a) Exponential distribution

$$L = \frac{1}{\beta^n} e^{-\frac{1}{\beta}\sum x_i}$$

$$l = \log L = -n \log \beta - \frac{1}{\beta} \sum x_i$$

$$\frac{dl}{d\beta} = -\frac{n}{\beta} + \frac{1}{\beta^2} \sum x_i = \frac{1}{\beta^2} \left(-n\beta + \sum x_i\right)$$

$$\therefore \beta = \frac{\sum x_i}{n}$$

(b) Pareto distribution

$$L = \theta^n x_0^{n\theta} \left(\Pi x_i\right)^{-\theta-1}$$

$$l = \log L = n \log \theta + n\theta \log x_0 - (\theta + 1) \sum \log x_i$$

$$\frac{dl}{d\theta} = \frac{n}{\theta} + n \log x_0 - \sum \log x_i$$

$$\frac{n}{\theta} = \sum \log x_i - n \log x_0$$

$$\frac{\theta}{n} = \frac{1}{\sum \log x_i - n \log x_0}$$

$$\therefore \theta = \frac{n}{\sum \log x_i - n \log x_0}$$

(c) Normal linear regression model

$$l = \log L = -\frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum (y_i - x_i\beta)^2$$

$$\frac{dl}{d\beta_j} = -\frac{1}{2\sigma^2} \sum 2(-x_{ij})(y_i - x_i\beta)$$

$$= \frac{1}{\sigma^2} \sum x_{ij}(y_i - x_i\beta)$$

$$\frac{dl}{d\beta} = \frac{1}{\sigma^2} \sum x_i^T (y_i - x_i\beta) = 0$$

$$\sum x_i^T y_i - \left(\sum x_i^T x_i\right) \beta = 0$$

In other words,

$$X^T y - X^T X \beta = 0$$

$$\hat{\beta_N} = (X^T X)^{-1} X^T y$$

$$l = \log L = -\frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum (y_i - x_i\beta)^2$$

$$\frac{dl}{d\sigma^2} = -\frac{N}{2\sigma^2} + \frac{1}{2\sigma^4}\sum(y_i - x_i\beta)^2$$
$$= \frac{1}{2\sigma^2}\left[\frac{1}{\sigma^2}\sum(y_i - x_i\beta)^2 - N\right]$$
$$\therefore \hat{\sigma_N}^2 = \frac{1}{N}\sum(y_i - x_i\hat{\beta_N})^2$$

Question 3. 1.

$$w = (w_1|w_2|\cdots|w_q)$$
$$(w^T w)_{ij} = w_i^T w_j$$

Since w_i is orthonormal set, $w_i^T w_i = 1$ and $w_i^T w_j = 0$, and we can conclude that

$$w^T w = I_q$$

2. We are given n datas x_i, each $x_i \in \mathbb{R}^p$.

$$\hat{x_i} = \sum_{j=1}^{q} < x_i, w_j > w_j$$

Let M be the matrix of q-dimensional approximations.

$$M_{ij} = < x_i, w_j >$$

3.

$$x = \sum_{j=1}^{q} < x, w_j > w_j + residual$$
$$MSE = \left\| x - \sum_{j=1}^{q} < x, w_j > w_j \right\|^2$$

By Pythagorean theorem,

$$\left\| x - \sum_{j=1}^{q} < x, w_j > w_j \right\|^2 + \left\| \sum_{j=1}^{q} < x, w_j > w_j \right\|^2 = \|x\|^2$$
$$\therefore MSE = \|x\|^2 - \left\| \sum_{j=1}^{q} < x, w_j > w_j \right\|^2,$$

where the first term on the right hand side depends only on x and the second term on the right side depends only on the scores along those directions.

4. By the above equation, we can see that minimizing projection residuals is equivalent to maximizing the sum of variances along the different directions.

Question 4. 1. Optimal clustering for this data is $1, 3$, and 6.5. (The object value is $\frac{1}{2}$).
2. If we initialize centers as $3, 6$, and 7, then the algorithm gives us $2, 6$, and 7 as a result. But the object value is 2 in this case, which is sub-optimal.
3. Each iteration of K-means algorithm decrease the objective, because both cluster assignment step and center adjustment step decrease the objective. Therefore, K-means algorithm converges to a local optimum in finite steps.
4. If data is distributed as several concentric circles, K-means doesn't work well. We can use other metrics for estimating distance, or use spectral clustering.

Report for Programming

K-medoids framwork

I used l^{\infty} norm for estimating distance.
And I initialized centers randomly.
For center adjustment step, I calculated mean of the members, and I let closest member to the mean to be a new representative.
I stopped iteration when the centers don't change.

K-medoids with K = 2

K-medoids with K = 3

K-medoids with K = 16

K-medoids with K = 32

K-medoids summary

As K get bigger, the result gets closer to the original picture.

Running time

```
/mnt/c/U/gyujin/Desktop/6740hw  python homework1.py wallpaper.jpg 2
wallpaper.jpg 2
--- 7.741184949874878 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 3
wallpaper.jpg 3
--- 8.411914110183716 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 16
wallpaper.jpg 16
--- 26.9620201587677 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 32
wallpaper.jpg 32
--- 48.40027379989624 seconds ---
/mnt/c/U/gyujin/De/6740hw
```

K-means with K = 2

K-means with K = 3

K-means with K = 16

K-means with K = 32

K-means summary

As K get bigger, the result gets closer to the original picture.

Running time

```
/mnt/c/U/gyujin/Desktop/6740hw  python homework1.py wallpaper.jpg 2
wallpaper.jpg 2
--- 4.464332580566406 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 3
wallpaper.jpg 3
--- 6.537437438964844 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 16
wallpaper.jpg 16
--- 21.669082403182983 seconds ---
/mnt/c/U/gyujin/De/6740hw  python homework1.py wallpaper.jpg 32
wallpaper.jpg 32
--- 41.67862153053284 seconds ---
```

Comparison between K-medoids and K-means

When I used the given picture, football.bmp, two algorithms gave me different result pictures.
But for what I used, there isn't much difference.
For quality and robustness, it was similar.
For running time, it was similar.