SHENDUO ZHANG
2176112379 **Statistics-71**

Problem

Suppose now we have two biased coin A and B, each one with probability θ_A and θ_B to come up with heads. In one experiment, we choose one coin to throw for 10 times independently, with probability π to choose coin A instead of B. We run the experiment n times and collect the numbers of heads in each experiment, denote it as $X = (x_1, x_2, \cdots, x_n)$.

We ask when $\Theta \ni \theta = (\theta_A, \theta_B, \pi)$ is unknown, how can we recover those parameter from the data we have.

We call the coin A as the one with a larger probability to come up with heads.

How the algorithm work

Introduced a latent variable $Z = (z_1, z_2, \cdots, z_n)$, where $z_i \in \{A, B\}$, for which coin was chosen for flipping in each experienment. And on each step we denote the parameter comes from the last step as $\theta^o = (\theta_A^o, \theta_B^o, \pi^o)$.(in the first step, it stands for the initial values which we need to specify manually when we start the algorithm.)

One approach is to try maximum log-likelihood estimator. In another word,

$$\theta = \arg\max_{\theta\in\Theta} \mathcal{L}(\theta) = \arg\max_{\theta\in\Theta} \log \mathbb{P}\{X|\theta\} \tag{1}$$

After introducing Z into the model, we have

$$\begin{aligned}
\log \mathbb{P}\{X|\theta\} &= \log \sum_Z \mathbb{P}\{X, Z|\theta\} = \log \sum_Z \mathbb{P}\{X|Z,\theta\}\mathbb{P}\{Z|\theta\} \\
&= \log \sum_Z \mathbb{P}\{Z|X,\theta^o\} \frac{\mathbb{P}\{X|Z,\theta\}\mathbb{P}\{Z|\theta\}}{\mathbb{P}\{Z|X,\theta^o\}} \\
&= \log \mathbb{E}_{Z|X,\theta^o} \frac{\mathbb{P}\{X|Z,\theta\}\mathbb{P}\{Z|\theta\}}{\mathbb{P}\{Z|X,\theta^o\}} \\
&\geq \mathbb{E}_{Z|X,\theta^o} \log \frac{\mathbb{P}\{X|Z,\theta\}\mathbb{P}\{Z|\theta\}}{\mathbb{P}\{Z|X,\theta^o\}}
\end{aligned} \tag{2}$$

The inequality follows from the concaveness of logarithm and Jensen's inequality. Instead of choose a parameter that maximize the log-likelihood of X, we now find its lower bound as a candidate for it.

$$\theta = \arg\max_{\theta\in\Theta} \mathbb{E}_{Z|X,\theta^o} \log \frac{\mathbb{P}\{X|Z,\theta\}\mathbb{P}\{Z|\theta\}}{\mathbb{P}\{Z|X,\theta^o\}} = \arg\max_{\theta\in\Theta} \mathbb{E}_{Z|X,\theta^o} \log \mathbb{P}\{X, Z|\theta\} \tag{3}$$

And each time we will improve this lower bound a little bit until we reach it maximum.

2.1 Expectation

In the $\mathbb{E}$ step, we need to compute $\mathbb{E}_{Z|X,\theta} \log \mathbb{P}\{X|Z,\theta\}$. It follows that,

$$\begin{aligned}
\Omega(\theta,\theta^o) &= \mathbb{E}_{Z|X,\theta^o} \log \mathbb{P}\{X,Z|\theta\} \\
&= \mathbb{E}_{Z|X,\theta^o} \sum_{i=1}^{n} \log \mathbb{P}\{X=x_i,Z|\theta\} \\
&= \sum_{z\in\{A,B\}} \sum_{i=1}^{n} \log(\mathbb{P}\{X=x_i,Z=z|\theta\})\mathbb{P}\{Z=z|X=x_i,\theta^o\} \\
&= \sum_{z\in\{A,B\}} \sum_{i=1}^{n} \log(\mathbb{P}\{X=x_i|Z=z,\theta\}\mathbb{P}\{Z=z|\theta\})\mathbb{P}\{Z=z|X=x_i,\theta^o\} \\
&= \sum_{z\in\{A,B\}} \sum_{i=1}^{n} \log(\mathbb{P}\{X=x_i|Z=z,\theta\}\mathbb{P}\{Z=z|\theta\})\frac{\mathbb{P}\{X=x_i|Z=z,\theta^o\}\mathbb{P}\{Z=z|\theta^o\}}{\mathbb{P}\{X=x_i|\theta^o\}}
\end{aligned} \tag{4}$$

Denote the posterior of Z as $\gamma_i(z,\theta^o) = \mathbb{P}\{Z=z|X=x_i,\theta^o\}$, we have

$$\gamma_i(A,\theta^o) = \frac{{\theta_A^o}^{x_i}(1-\theta_A^o)^{10-x_i}}{{\theta_A^o}^{x_i}(1-\theta_A^o)^{10-x_i} + {\theta_B^o}^{x_i}(1-\theta_B^o)^{10-x_i}}\pi^o \tag{5}$$

$$\gamma_i(B,\theta^o) = \frac{{\theta_B^o}^{x_i}(1-\theta_B^o)^{10-x_i}}{{\theta_A^o}^{x_i}(1-\theta_A^o)^{10-x_i} + {\theta_B^o}^{x_i}(1-\theta_B^o)^{10-x_i}}(1-\pi^o) \tag{6}$$

Then it follows that,

$$\begin{aligned}
\Omega(\theta,\theta^o) &= \sum_{i=1}^{n} \gamma_i(A,\theta^o)\log(\theta_A^{x_i}(1-\theta_A)^{10-x_i}\pi) + \gamma_i(B,\theta^o)\log\{\theta_B^{x_i}(1-\theta_B)^{10-x_i}(1-\pi)\} \\
&= \log\{\theta_A\}\sum_{i=1}^{n} x_i\gamma_i(A,\theta^o) + \log\{1-\theta_A\}\sum_{i=1}^{n}(10-x_i)\gamma_i(A,\theta^o) + \log\{\pi\}\sum_{i=1}^{n}\gamma_i(A,\theta^o) \\
&+ \log\{\theta_B\}\sum_{i=1}^{n} x_i\gamma_i(B,\theta^o) + \log\{1-\theta_B\}\sum_{i=1}^{n}(10-x_i)\gamma_i(B,\theta^o) + \log\{1-\pi\}\sum_{i=1}^{n}\gamma_i(B,\theta^o)
\end{aligned} \tag{7}$$

With some extra notation

$$\begin{aligned}
\alpha_A &= \sum_{i=1}^{n} x_i\gamma_i(A,\theta^o), \quad \beta_A = \sum_{i=1}^{n}(10-x_i)\gamma_i(A,\theta^o) \\
\alpha_B &= \sum_{i=1}^{n} x_i\gamma_i(B,\theta^o), \quad \beta_B = \sum_{i=1}^{n}(10-x_i)\gamma_i(B,\theta^o) \\
\Gamma_A &= \sum_{i=1}^{n}\gamma_i(A,\theta^o), \quad \Gamma_B = \sum_{i=1}^{n}\gamma_i(B,\theta^o),
\end{aligned} \tag{8}$$

we can simplify the function into a nice form

$$\Omega(\theta,\theta^o) = \alpha_A\log\theta_A + \beta_A\log\{1-\theta_A\} + \Gamma_A\log\pi + \alpha_B\log\theta_B + \beta_B\log\{1-\theta_B\} + \Gamma_B\log\{1-\pi\} \tag{9}$$

2.2 Maximation

To maximize $\Omega(\theta, \theta^o)$ over all the θ, take partial derivative and find the zero of it.

$$\begin{aligned}\frac{\partial \Omega}{\partial \theta_A} &= \frac{\alpha_A}{\theta_A} - \frac{\beta_A}{1-\theta_A} = 0 \\ \frac{\partial \Omega}{\partial \theta_B} &= \frac{\alpha_B}{\theta_B} - \frac{\beta_B}{1-\theta_B} = 0 \\ \frac{\partial \Omega}{\partial \pi} &= \frac{\Gamma_A}{\pi} - \frac{\Gamma_B}{1-\pi} = 0\end{aligned} \tag{10}$$

Which gives us

$$\begin{aligned}\theta_A &= \frac{\alpha_A}{\alpha_A + \beta_A} \\ \theta_B &= \frac{\alpha_B}{\alpha_B + \beta_B} \\ \pi &= \frac{\Gamma_A}{\Gamma_A + \Gamma_B}\end{aligned} \tag{11}$$

Result And Analysis

```
Data = Generate_data(0.8,0.2,0.7,10,1000)#Generate data for coin

#Test for EM algorithm on some initial values
print(EM(Data,0.4,0.4,0.5,10))
print(EM(Data,0.5,0.6,0.5,10))
print(EM(Data,0.6,0.5,0.5,10))
print(EM(Data,0.6,0.5,0.4,10))
```

We generate the coin data with parameter $\theta_A = 0.8, \theta_B = 0.2, \pi = 0.7$, and we do this for a thousand times. Then we run the EM algorithm with different initial values, each through 1000 steps.

And we make extra assumption that $\theta_A^o > \theta_B^o$, because there will be no way for a computer to distinguish which coin was given the name A by us using information other than this parameter. Which is also shown in the result that reverse the position of two initial value, the outcome will also reverse in position. And when the $\theta_A^o = \theta_B^o$, the algorithm would just be not able to distinguish two coin which makes perfectly sense.

```
[0.6239, 0.6239, 0.5]
[0.20846739321853716, 0.8069183048197647, 0.0009769037295276348]
[0.8069183048197647, 0.20846739321853539, 0.9990230962704724]
[0.8069183048197652, 0.208467393218538, 0.9985353598119396]
```

The result shows that the EM algorithm can easily recover the parameter θ_A, θ_B. However the parameter π can not be recovered.

It happens that alone with the iteration goes, the parameter π just becomes either 1 or 0. And the value of θ_A, θ_B is reversed in position in the two cases, which means there is simply not enough

information for π to be recovered. There is no way to tell the difference between two pair of distinct parameters, because they will result in a same distribution on our data.