Identifiability

We will show that the EM algorithm may behave in an unexpected way

Suppose we observe a sequence $X_1, \dots, X_n$ of coin-flips $X_i \in \{0,1\}$. The sequence is generated using three coins as follows: in each step the player first tosses Coin-1 and the result of this toss is denoted by z and we assume $z \sim \mathrm{Bern}(\frac{1}{2})$. If $z=0$, the player then tosses Coin-2 and if $z=1$, she tosses Coin-3. We have

$$p(X_i = 1|Z=0) = a_0, \qquad p(X_i=1|Z=1) = a_1$$

and for identifiability purposes, we may assume $a_1 > a_0$.
The probability of observing sequence $x_1, \dots, x_n$ is

$$\sum_{z\in\{0,1\}^n} \prod_{i=1}^{n} \tfrac{1}{2}(1-a_0)^{(1-x_i)(1-z_i)} a_0^{x_i(1-z_i)} (1-a_1)^{(1-x_i)z_i} a_1^{x_i z_i}.$$

Optimizing this directly may be hard but the EM-algorithm is simple.

The complete observed log-likelihood is

$$\sum_{i=1}^{n} \log p(x_i, z_i|\theta) = n\log\tfrac{1}{2} + n_{00}\log(1-a_0) + n_{01}\log(a_0) + n_{10}\log(1-a_1) + n_{11}\log(a_1),$$

where

$$n_{00} = \sum_{i=1}^{n}(1-z_i)(1-x_i),\ n_{01} = \sum_{i=1}^{n}(1-z_i)x_i,\ n_{10} = \sum_{i=1}^{n} z_i(1-x_i),\ n_{11} = \sum_{i=1}^{n} z_i x_i.$$

If we knew $n_{00}, n_{01}, n_{10}, n_{11}$ the estimator would be trivial $\hat{a}_0 = \frac{n_{10}}{n_{00}+n_{10}}$, $\hat{a}_1 = \frac{n_{11}}{n_{01}+n_{11}}$. The EM algorithm starts by estimating n_{ij}'s based on the observe data and the underlying parameters.

Note that

$$\begin{aligned} p(z_i=1|x_i,\theta) = \mathbb{E}[z_i|x_i,\theta] &= \frac{p(x_i|z_i=1)p(z_i=1)}{p(x_i|z_i=0)p(z_i=0)+p(x_i|z_i=1)p(z_i=1)} \\ &= \frac{(1-a_1)^{1-x_i}a_1^{x_i}}{(1-a_0)^{1-x_i}a_0^{x_i} + (1-a_1)^{1-x_i}a_1^{x_i}} \end{aligned}$$

and in particular

$$p(z_i=1|x_i=0,\theta) = \frac{1-a_1}{2-a_0-a_1} < \frac{1}{2}, \quad p(z_i=1|x_i=1,\theta) = \frac{a_1}{a_0+a_1} > \frac{1}{2}.$$

Denote $c_0 = p(z_i = 1|x_i = 0, \theta)$ and $c_1 = p(z_i = 1|x_i = 1, \theta)$ and let $n_1 = \sum x_i$. Thus

$$\mathbb{E}(n_{00}|\mathbf{X}) = \sum_{i:x_i=0} \frac{1-a_0}{2-a_0-a_1} = (n-n_1)(1-c_0),$$

$$\mathbb{E}(n_{01}|\mathbf{X}) = \sum_{i:x_i=1} \frac{a_0}{a_0+a_1} = n_1(1-c_1),$$

$$\mathbb{E}(n_{10}|\mathbf{X}) = \sum_{i:x_i=0} \frac{1-a_1}{2-a_0-a_1} = (n-n_1)c_0,$$

$$\mathbb{E}(n_{11}|\mathbf{X}) = \sum_{i:x_i=1} \frac{a_1}{a_0+a_1} = n_1c_1,$$

Now we are able to optimize the expected complete likelihood and we get

$$a_0^{\text{new}} = \frac{n_1(1-c_1)}{n_1(1-c_1)+(n-n_1)(1-c_0)}$$

and

$$a_1^{\text{new}} = \frac{n_1c_1}{n_1c_1+(n-n_1)c_0}.$$

Implementing this will give results that do not recover the true parameters even approximately. But there is an important reason for that. It can be shown that the likelihood depends on a_0,a_1 only through a_0+a_1 and so the model is not identifiable. To see this note that

$$\begin{aligned}
&\sum_{z\in\{0,1\}^n} \prod_{i=1}^n (1-a_0)^{(1-x_i)(1-z_i)} a_0^{x_i(1-z_i)} (1-a_1)^{(1-x_i)z_i} a_1^{x_i z_i} \\
&= \sum_{z\in\{0,1\}^n} \prod_{i=1}^n \left[(1-a_0)^{1-x_i} a_0^{x_i}\right]^{1-z_i} \left[(1-a_1)^{1-x_i} a_1^{x_i}\right]^{z_i} \\
&= \prod_{i=1}^n \left((1-a_0)^{1-x_i} a_0^{x_i} + (1-a_1)^{1-x_i} a_1^{x_i}\right) \\
&= \prod_{i=1}^n \left((a_0+a_1)^{x_i} + (2-(a_0+a_1))^{1-x_i}\right).
\end{aligned}$$

This means that the log-likelihood is maximized over an affine space and the only thing we can identify is the sum $a_0 + a_1$.

This is the code. Run it and verify that $a_0 + a_1$ is correctly estimated.

```
import numpy as np

# Generate some fake data
np.random.seed(42)
n=1000
```

```
latent_data = np.random.randint(2, size=n)
observed_data = np.zeros(n)

for i in range(n):
    if latent_data[i] == 0:
        observed_data[i] = np.random.binomial(1, 0.3)
    else:
        observed_data[i] = np.random.binomial(1, 0.9)

# Initialize the parameters
a0 = 0.4
a1 = 0.6

n1=np.sum(observed_data)
# Run the EM algorithm
for _ in range(100):
    c0=(1-a1)/(2-a0-a1)
    c1=a1/(a0+a1)
    a0=(n1*(1-c1))/(n1*(1-c1)+(n-n1)*(1-c0))
    a1=(n1*c1)/(n1*c1+(n-n1)*c0)



print("Estimated values:")
print(f"a = {a0}")
print(f"b = {a1}")
print(a0+a1)```

In the example code above for the true parameters $a_0+a_1=1.2$ and we get
```

Estimated values:
a = 0.5052429156943381
b = 0.696757084305662
1.202

```


# Kullback-Leibler divergence is non-negative

We will show that the Kullback-Leibler divergence is nonnegative. We have
$$
{\rm KL}(p\|q)\;=\;\mathbb E_p \log\frac{p(X)}{q(X)}\;=\;-\mathbb E_p
\log\frac{q(X)}{p(X)}.
$$
We will use the following important result
```

```
**Theorem (Jensen's inequality):** If $f:\mathbb R\to \mathbb R$ is convex
then $f(\mathbb E(Y))\leq \mathbb E f(Y)$.

Using this theorem and the fact that $-\log(y)$ is convex for $y>0$, we get
$$
{\rm KL}(p\|q)\;=\;\mathbb E_p\left(-\log\left(\frac{q(X)}
{p(X)}\right)\right)\geq -\log(\mathbb E_p \frac{q(X)}{p(X)})\;=\;-\log
(1)\;=\;0.
$$
```