


---

---

---

---

---



1) a)

$$\log p(\theta) = \sum_{x \in D} \sum_{z \in \{H, T\}} \frac{p(z|\theta) p(x|z, \theta) q(z|x)}{q(z|x)}$$

$$\geq \sum_{x \in D} \sum_{z \in \{H, T\}} \log \left( \frac{p(z|\theta) p(x|z, \theta)}{p(z|x)} \right) q(z|x)$$

Jensen

$$= \sum_{x \in D} \sum_{z \in \{H, T\}} [\log(p(z|\theta)) + \log(p(x|z, \theta))] q(z|x)$$

$$- \sum_{x \in D} \sum_{z \in \{H, T\}} \log(p(z|x)) q(z|x) = \sum_{x \in D} \left( \sum_{z \in \{H, T\}} [\log(p(z|\theta)) + \log(p(x|z, \theta))] q(z|x) \right) + G(x)$$

b)  $J(\theta) = J(\lambda, a, b)$

$$= \sum_{x \in D} \left( G(x) + q(z=H|x) \cdot [\log(\lambda) + \log(\binom{m}{x} a^x (1-a)^{m-x})] \right. \\ \left. + q(z=T|x) \cdot [\log(1-\lambda) + \log(\binom{m}{x} b^x (1-b)^{m-x})] \right)$$

$$\frac{dJ}{d\lambda} \stackrel{!}{=} 0 \Rightarrow \sum_{x \in D} q(z=H|x) \cdot \left[ \frac{1}{\lambda} \right] + q(z=T|x) \cdot \frac{1}{1-\lambda} \cdot (-1) = 0$$

$$\sum_{x \in D} q(z=H|x) \cdot \frac{1}{\lambda} - q(z=T|x) \cdot \frac{1}{1-\lambda} = 0$$

$$\sum_{x \in D} q(z=H|x) \cdot \frac{1-\lambda}{\lambda \cdot \cancel{\lambda^2}} - q(z=T|x) \cdot \frac{\lambda}{\cancel{\lambda} \cdot \cancel{\lambda^2}} = 0$$

$$\sum_{x \in D} q(z=H|x) - \lambda \cdot q(z=H|x) - \lambda q(z=T|x) = 0$$

$$\sum_{x \in D} q(z=H|x) - \lambda (q(z=H|x) + q(z=T|x)) = 0$$

$= 1$

$$\lambda N = \sum_{x \in \mathcal{D}} q(z=H|x)$$

$$\lambda = \frac{1}{N} \sum_{x \in \mathcal{D}} q(z=H|x)$$


---

$$\frac{dJ}{da} = \sum_{x \in \mathcal{D}} q(x=H|x) \cdot \left( \log \left( \binom{m}{x} a^x (1-a)^{m-x} \right) \right)$$

$$= \sum_{x \in \mathcal{D}} q(x=H|x) \left[ \log \left( \binom{m}{x} \right) + \log(a^x) + \log((1-a)^{m-x}) \right]$$

$$= \sum_{x \in \mathcal{D}} q(x=H|x) \left( \frac{1}{a^x} \cdot x \cdot a^{x-1} - \frac{1}{(1-a)^{m-x}} \cdot (m-x)(1-a) \right)$$

$$= \sum_{x \in \mathcal{D}} q(x=H|x) \left( \frac{x}{a} - \frac{m-x}{1-a} \right)$$

$$= \sum_{x \in \mathcal{D}} q(x=H|x) \left( \frac{x}{a} - \frac{m}{1-a} + \frac{x}{1-a} \right) = 0$$

$$= \sum_{x \in \mathcal{D}} q(z=H|x) \cdot \left( \frac{x - \cancel{x}a - \cancel{m}a + \cancel{x}a}{a - a^2} \right)$$

$$= \sum_{x \in \mathcal{D}} q(z=H|x) (x - ma) = 0$$

$$\Rightarrow \sum_{x \in \mathcal{D}} q(z=H|x) x = ma \sum_{x \in \mathcal{D}} q(z=H|x)$$

$$a = \frac{\sum_{x \in \mathcal{D}} q(z=H|x) x}{\sum_{x \in \mathcal{D}} q(z=H|x) m}$$

analog für b

$$c) \quad q(z|x) = p(z|x, \theta) = \frac{p(z, x, \theta)}{p(x, \theta)} = \frac{p(x|z, \theta) \cdot p(z, \theta)}{p(x, \theta)}$$

$$p(x, \theta) = \frac{p(x|\theta) \cdot p(\theta)}{1}$$

$$\downarrow \quad \quad \quad \downarrow$$
$$p(x|\theta, z=H) + p(x|\theta, z=T) \quad \left\{ \frac{1}{2} \right.$$

$$\Rightarrow q(z|x) = \gamma \cdot \binom{m}{x} a^x (1-a)^{m-x} \quad \lambda$$

analog für  $z=T$

$$c) \mathcal{Q}(z|x) = P(z|x, \theta) = \frac{P(x|z, \theta) P(z|\theta)}{P(x|\theta)}$$

$$\mathcal{Q}(z=H|x) = \binom{m}{x} a^x (1-a)^{m-x} \cdot \lambda \cdot \frac{1}{P(x|\theta)} \quad \mathcal{Q}(z=T|x) \text{ analog.}$$

$$= \gamma \cdot a^x (1-a)^{m-x} \cdot \lambda$$

$$\mathcal{Q}(z=H|x) + \mathcal{Q}(z=T|x) = \binom{m}{x} a^x (1-a)^{m-x} \cdot \lambda \frac{1}{P(x|\theta)} + \binom{m}{x} b^x (1-b)^{m-x} \cdot (1-\lambda) \frac{1}{P(x|\theta)}$$

$$= \frac{1}{P(x|\theta)} \binom{m}{x} \left[ a^x (1-a)^{m-x} \cdot \lambda + b^x (1-b)^{m-x} \cdot (1-\lambda) \right]$$

$$= \frac{1}{P(x|\theta)} \binom{m}{x} \cdot \left[ \frac{1}{\binom{m}{x}} \cdot P(x|z=H, \theta) P(z=H|\theta) + \frac{1}{\binom{m}{x}} P(x|z=T, \theta) \cdot P(z=T|\theta) \right]$$

$$= \frac{1}{P(x|\theta)} \cdot (P(x|\theta)) = 1$$

# sheet12

February 8, 2021

## 1 Expectation-Maximization

In this programming exercise we will apply the Expectation-Maximization method to estimate the parameters of the latent variable model presented in the theoretical part. The analysis will be performed on one dimension of the Boston Housing dataset, where input values are integers between 0 and 24.

```
[92]: import sklearn, sklearn.datasets

X = sklearn.datasets.load_boston()['data'][:,8]
m = 24
```

We restate here the model presented in the theory part. The first distribution  $p(z|\theta)$  produces the latent state  $z \in \{H, T\}$  which can be interpreted as the head/tail outcome of an (unfair) coin toss, and the following two distributions  $p(x | z = H, \theta)$ ,  $p(x | z = T, \theta)$  produce the observed data  $x \in \{0, 1, \dots, m\}$  conditioned on outcome of the coin toss  $z$ . The probability distributions are defined as:

$$\begin{aligned} p(z = H | \theta) &= \lambda \\ p(z = T | \theta) &= 1 - \lambda \\ p(x | z = H, \theta) &= \binom{m}{x} a^x (1 - a)^{m-x} \\ p(x | z = T, \theta) &= \binom{m}{x} b^x (1 - b)^{m-x} \end{aligned}$$

The vector  $\theta = (\lambda, a, b)$  contains the parameters of the model and these parameters need to be learned.

### 1.1 Computing the Probability Mass Function (15 P)

We first would like to implement a function that computes the probability mass function  $p(x|\theta)$  for every value of  $x$  between 0 and  $m$ .

**Task:**

- Implement the function `pmf(lambd,a,b,m)`

```
[93]: import numpy
from scipy.special import binom

def pmf(lambd,a,b,m):
    # -----
    # TODO: Replace by your code
    # -----
    x = numpy.arange(m+1)
    pH = binom(m,x) * a ** x * (1-a)**(m-x)
    pT = binom(m,x) * b ** x * (1-b)**(m-x)
    ret = lambd*pH + (1-lambd)*pT
    # -----
    return ret
```

## 1.2 Computing the Log-Likelihood (15 P)

Additionally, we would like to measure the quality of our latent variable model, specifically, the log-likelihood of the data according to the model:

$$\log p(\mathcal{D}|\theta) = \sum_{x \in \mathcal{D}} \log p(x|\theta)$$

Task:

- Implement the function below that takes the data and the model parameters as input and returns the log-likelihood.

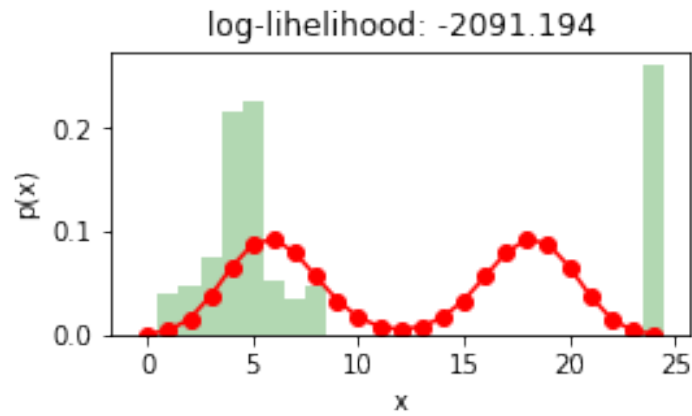
```
[94]: def ll(X,lambd,a,b,m):
    ll = [pmf(lambd,a,b,m)[int(x)] for x in X]
    ll = numpy.array(ll)
    summ = numpy.log(ll).sum()
    return summ
```

```
[95]: import matplotlib
%matplotlib inline
from matplotlib import pyplot as plt
import numpy

def plot(data,pdf,ll):
    f = plt.figure(figsize=(4,2))
    plt.hist(data,bins=numpy.arange(m+2)-0.5,alpha=0.3,color='g',density=True)
    plt.plot(range(int(m+1)),pdf,'-o',c='r')
    plt.ylabel('p(x)')
    plt.xlabel('x')
    plt.title('log-likelihood: %.3f'%ll)

lambd,a,b = 0.5,0.25,0.75
```

```
plot(X,pmf(lambd,a,b,m),ll(X,lambd,a,b,m))
```



We observe that the model is not very accurate and the log-likelihood of the data is quite low.

### 1.3 Implementing and Running the EM Algorithm (20 P)

We now would like to implement the EM algorithm that iteratively searches for optimal parameters  $\lambda$ ,  $a$  and  $b$ . The function starts with some initial parameters that we set at hand. In each iteration, the E-step and the M-step are applied. According to our derivations in the theory part, the E-step and M-step are given by:

*E-Step:*

$$q(z = H|x) = \gamma \cdot a^x (1 - a)^{m-x} \cdot \lambda$$

$$q(z = T|x) = \gamma \cdot b^x (1 - b)^{m-x} \cdot (1 - \lambda)$$

with  $\gamma$  a normalization constant set to ensure  $q(z = H|x) + q(z = T|x) = 1$ .

*M-Step:*

$$\lambda = \frac{1}{N} \sum_{x \in \mathcal{D}} q(z = H|x) \quad a = \frac{\sum_{x \in \mathcal{D}} q(z = H|x) \cdot x}{\sum_{x \in \mathcal{D}} q(z = H|x) \cdot m} \quad b = \frac{\sum_{x \in \mathcal{D}} q(z = T|x) \cdot x}{\sum_{x \in \mathcal{D}} q(z = T|x) \cdot m}$$

**Task:**

- Implement the E-Step and the M-Step.

```
[96]: def Estep(lambd,a,b,X,m):
```

```
    # -----
```

```
    # TODO: Replace by your code
```



```

# -----
qH = a**X*(1-a)**(m-X)*lambd
qT = b**X*(1-b)**(m-X)*(1-lambd)
gamma = (qH + qT)
ret = (qH/gamma, qT/gamma)
# -----
return ret

def Mstep(qH,qT,X,m):
    # -----
    # TODO: Replace by your code
    # -----
    lambd = qH.sum()/len(X)
    a = (qH*X).sum()/(qH*m).sum()
    b = (qT*X).sum()/(qT*m).sum()
    ret = (lambd, a, b)
    # -----
    return ret

```

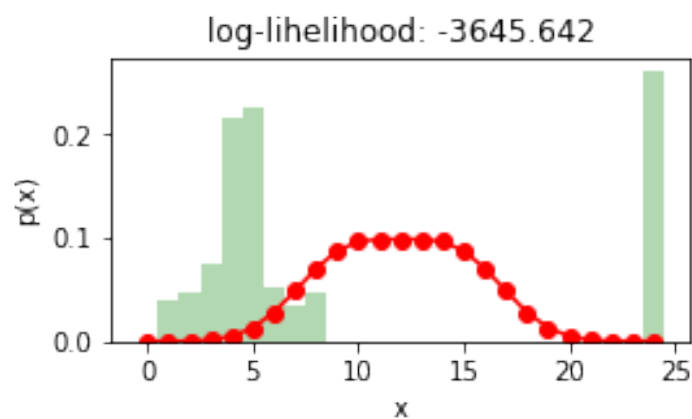
The code below runs the EM procedure for 4 iterations, starting from some initial parameters chosen at hand, and shows the pdf and the log-likelihood at each iteration.

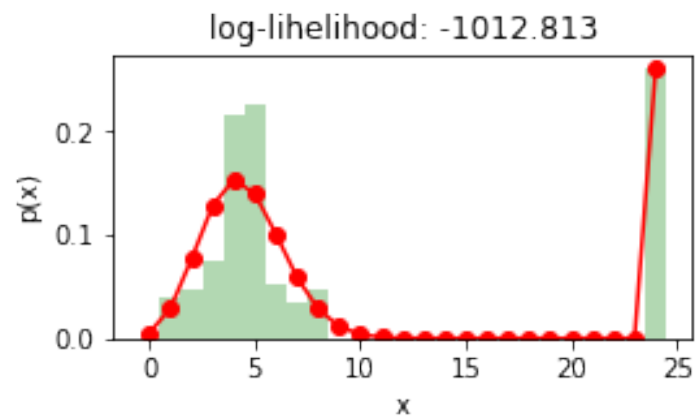
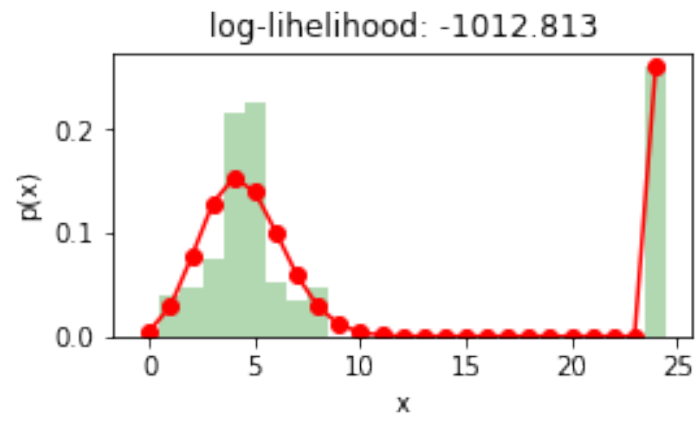
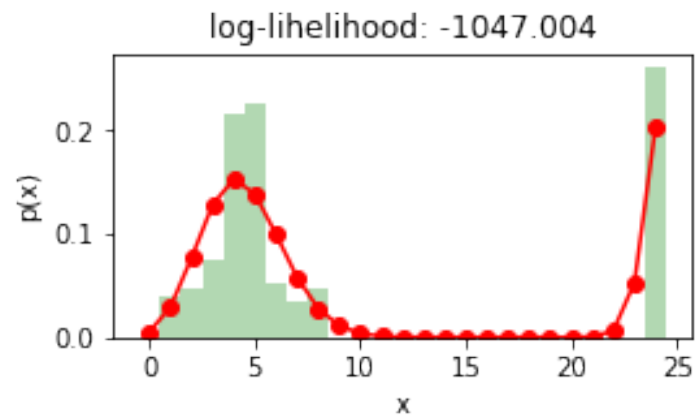
```

[97]: lambd,a,b = 0.5,0.4,0.6

for t in range(4):
    plot(X,pmf(lambd,a,b,m),ll(X,lambd,a,b,m))
    qH,qT = Estep(lambd,a,b,X,m)
    lambd,a,b = Mstep(qH,qT,X,m)

```





Although the initial choice of parameters is not good, the procedure converges very quickly, and

the improvement becomes marginal after only two iterations.