**Expectation Maximization with Coin Flips**

AYUSH TRIPATHI

12/03/2022

Algorithm Summary

In the above example, a coin among two coin 1 and coin 2 is chosen 30 times and tossed 20 times. The results of the 20 coin tosses is provided via the API, however, the identity of which coin tossed against these values is not provided.

The probability is to either get heads or tails, when a coin is tossed. In this case, we are taking the probability of getting heads while tossing for coin 1 and coin 2 as θ1 and θ2, respectively. In the code:

Θ1 = theta\_coin\_1 and θ2 = theta\_coin\_2

For developing the EM algorithm the following steps were taken:

Step 1: Data Preparation

The coin was selected randomly and then flipped for 20 times and the number of heads and tails were recorded. This procedure was repeated for 30 times to record 600 coin flips.

The data can be represented as below:

The toss results (H/T) with the chosen coin X = (X1, X2……X20) where Xn ε (0, 1)

The coin chosen for the toss = Z (Z1, Z2…………….Z30) where Zn ε (A, B)

A the value of X is provided and the value of Z isn’t, the Z values were considered as hidden variables.

Step 2: Estimation

The first assumption made was the probability of getting heads for Coin 1 is 60% and for Coin 2 is 50% as below:

Θ1 (0) = 0.6

Θ2 (0) = 0.5

The 20 coin tossed with the chosen coin X = (X1, X2……X20) where Xn ε (0, 1)

Step 2: Maximization

The probability of getting heads and tails can be derived using binomial distribution and normalizing the values for each toss.

P(K) = θK (1-θ)n-k

The total value for each coin was derived, and this gives new theta values.

Θ1(n) = Y ; θ2(n) = Z

This process (refer figures a, b) was repeated until the best result was derived.

Table

Description automatically generated

Diagram

Description automatically generated with medium confidence

Figure a,b: Example illustrating the process followed for deriving the algorithm finding maximum likelihood and expectation maximization (2 coins, 5 times and 10 coin flips); source[1]

The following assumptions were made while writing this code:

1. the initialized values for the first coin was taken as 0.6 and an OR between any random value in (0, 1) and the bias for the second coin was initialized to 0.5 and an OR between any random value in (0, 1).
2. An AND operation was introduced between these two choices (define a max number of iterations the algorithm iterates, or stopping the iteration as soon as the convergence is achieved) so that the convergence could be reached in at most 100 steps of iterations else the program would stop iterating over the values of θ1 and θ2.

**Results:**

The figure c and d demonstrates how the value of theta for Coin 1 and 2 has changed over the iterations until the convergence is achieved. Using Matplotlib the likelihood of the values for θ1 and θ2 on a contour plot is illustrated (figure e). The estimates for θ1 and θ2 for this algorithm are θ1= 0.737 θ2 = 0.283 which was achieved at 22nd iteration.

|  |  |
| --- | --- |
|  |  |
|  |  |
| Figure c: Bias calculation for Coin 1 across the number of iterations | Figure d: Bias calculation for Coin 2 across the number of iterations |

A picture containing text, electronics

Description automatically generated

Figure e: Log Likelihood for θ1 and θ2 with respect to each other

**References:**

1. Do, C., Batzoglou, S. What is the expectation maximization algorithm?. *Nat Biotechnol* 26, 897–899 (2008). <https://doi.org/10.1038/nbt1406>
2. <http://karlrosaen.com/ml/notebooks/em-coin-flips/>
3. <http://karlrosaen.com/ml/notebooks/em-coin-flips/>
4. <https://www.youtube.com/watch?v=7e65vXZEv5Q&t=535s>