# CS365
# Foundations of Data Science

Charalampos E. Tsourakakis
ctsourak@bu.edu

# Two coin problem

$\Pr(H) = \theta_A$

$\Pr(H) = \theta_B$

Process

- Suppose we choose a coin (A or B) uniformly at random (uar)
- We toss the coin n times, and record the total number of heads

We repeat the process k times

# Two coin problem

- We have two unknown parameters $\theta_A$, $\theta_B$

- Suppose k=5, n=10

- The data are two vectors
  - $x=X_H = (x_1, x_2, x_3, x_4, x_5)$ #heads per round
  - $z=Z_c = (z_1, z_2, z_3, z_4, z_5)$ #coin id per round

Resue

- E.g., x=(5,9,8,4,7), z=(B, A, A, B, A)

$\frac{24}{30}$

Result

second coin

First coin

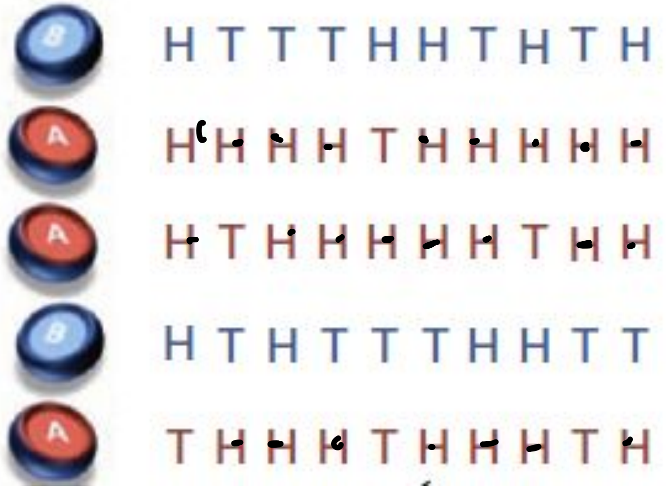
$\frac{11}{20}$

15

A: $\frac{24}{30}$ B $\frac{\overline{9}}{20}$

## Two coin problem

Comprehension question: What are the MLE for $\theta_A$, $\theta_B$ in this case?



H T T T H H T H T H

H H H H T H H H H H 9

H T H H H H H T H H 8

H T H T T T H H T T

T H H H T H H H T H 7

$$\frac{9}{20} \qquad \frac{24}{30}$$

$$L(\theta_A, \theta_B; D)^{\theta_A}$$

24 6 9 11

$$= \theta_A \, (1-\theta_A) \, \theta_B \, (1-\theta_B)$$

$$\log L(\theta_A, \theta_B) = 24 \log \theta_A + 6 \log(1-\theta_A) + 9 \log(\theta_B) + 11 \log(1-\theta_B)$$

$$= \frac{\ }{\theta_A} \quad \frac{24}{\theta_A} + \frac{6}{1-\theta_A} - 1 = 0 \qquad \frac{24}{\theta_A} - \frac{6}{1-\theta_A} = 0$$

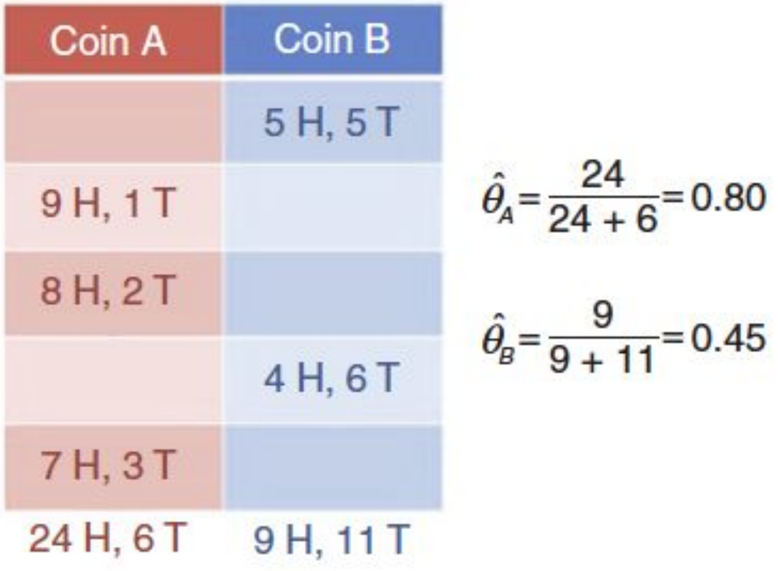
## Two coin problem

$$\theta_B = \frac{6}{1-\theta_B} - 1 + \frac{9}{\theta_B}$$

$$24 - 24\theta_A = 6\theta_A \qquad \theta = \frac{24}{30}$$

$$30\theta_A = 24$$

Maximum likelihood estimates for the unknown parameters $\theta_A$, $\theta_B$



| Coin A | Coin B |
|---|---|
| | 5 H, 5 T |
| 9 H, 1 T | |
| 8 H, 2 T | |
| | 4 H, 6 T |
| 7 H, 3 T | |
| 24 H, 6 T | 9 H, 11 T |

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

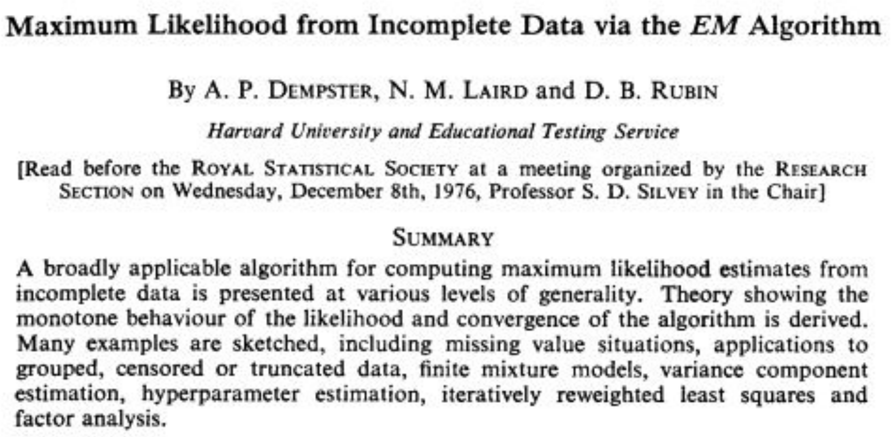
$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

# Two coin problem

- Suppose we only see the number of heads per round.
  - In other words we do not have access to z
- We refer to z as hidden variables or latent factors
- Remarks
  1. Not uncommon/common setting in data science applications ⇒ missing data!
  2. Clear that maximizing $\Pr(x|\theta)$ is much harder than $\Pr(x,z|\theta)$ in the presence of missing data z.

# The Expectation-Maximization algorithm

- Also known as the EM algorithm
  - In reality, it is an algorithm design methodology rather than a given algorithm



## Maximum Likelihood from Incomplete Data via the *EM* Algorithm

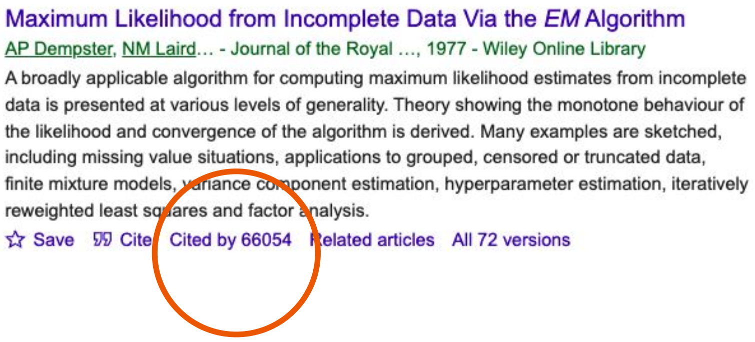
By A. P. DEMPSTER, N. M. LAIRD and D. B. RUBIN

*Harvard University and Educational Testing Service*

[Read before the ROYAL STATISTICAL SOCIETY at a meeting organized by the RESEARCH SECTION on Wednesday, December 8th, 1976, Professor S. D. SILVEY in the Chair]

SUMMARY

A broadly applicable algorithm for computing maximum likelihood estimates from incomplete data is presented at various levels of generality. Theory showing the monotone behaviour of the likelihood and convergence of the algorithm is derived. Many examples are sketched, including missing value situations, applications to grouped, censored or truncated data, finite mixture models, variance component estimation, hyperparameter estimation, iteratively reweighted least squares and factor analysis.



Maximum Likelihood from Incomplete Data Via the *EM* Algorithm

AP Dempster, NM Laird… - Journal of the Royal …, 1977 - Wiley Online Library

A broadly applicable algorithm for computing maximum likelihood estimates from incomplete data is presented at various levels of generality. Theory showing the monotone behaviour of the likelihood and convergence of the algorithm is derived. Many examples are sketched, including missing value situations, applications to grouped, censored or truncated data, finite mixture models, variance component estimation, hyperparameter estimation, iteratively reweighted least squares and factor analysis.

Let’s see the key idea in the context of the two-coins problem before full description

# Two coin problem

- We will proceed iteratively by updating
- Each iteration starts with a guess of the unknown parameters

$$\hat{\theta}^{(t)} = \left(\hat{\theta}_A^{(t)}, \hat{\theta}_B^{(t)}\right)$$

- E-step: a probability distribution over possible completions is computed using the current parameters $\hat{\theta}^{(t)}$

- M-step: the new parameters are determined using the current completions.

## Two coin problem

Suppose our initial guess for the unknown variables are 0.6, and 0.5.
$\theta_A$ $\theta_B$

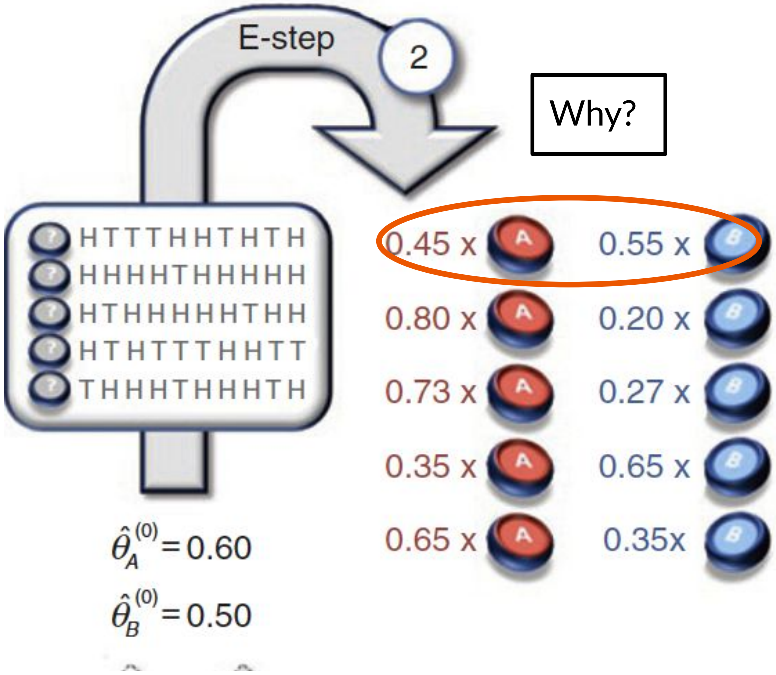
**E-step**: what is the probability that round i comes from coin A/coin B?

Likelihoods

Coin A: $\propto 0.6^5 \cdot 0.4^5 \approx 0.000796$

Coin B $\propto 0.5^5 \, 0.5^5 = \left(\frac{1}{2}\right)^{10} \approx 0.000976$

$$Pr(Z_1=A) = \frac{0.6^5 \cdot 0.4^5}{0.6^5 0.4^5 + (0.5)^{10}} = 0.45 , \quad Pr(Z_1=B) = 0.55$$



Why?

$\theta_A = 0.6$

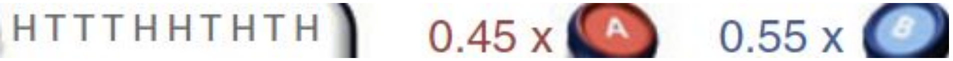
## Two coin problem

M-step: In order to learn $\hat{\theta}^{(t+1)} = \left(\hat{\theta}_A^{(t+1)}, \hat{\theta}_B^{(t+1)}\right)$ we first need to estimate the number of heads/tails from coins A/B given our estimates of the latent variables.

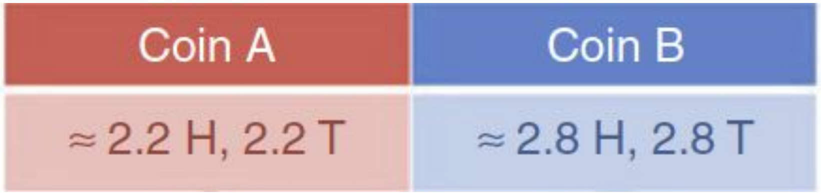
- Notice that instead of being 100% certain whether a round was due to coin A or B, we have a probability distribution.

# Two coin problem

**Round 1**: 5H, 5T



|  | Coin A | Coin B |
|---|---|---|
| Heads | 0.45x5 | 0.55x5 |
| Tails | 0.45x5 | 0.55x5 |



| Coin A | Coin B |
|---|---|
| ≈ 2.2 H, 2.2 T | ≈ 2.8 H, 2.8 T |

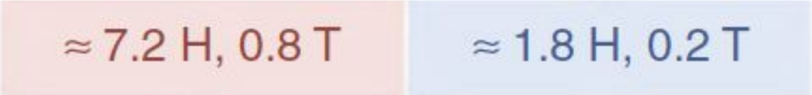
We repeat this for all five rounds

## One more round

**Round 2**: 9H, 1T
From the E-round we have

0.80 x
A
0.20 x
B

|  | Coin A | Coin B |
|---|---|---|
| Heads | 0.8x9 | 0.2x9 |
| Tails | 0.8x1 | 0.2x1 |


≈ 7.2 H, 0.8 T

≈ 1.8 H, 0.2 T

Likelihoods Round 2

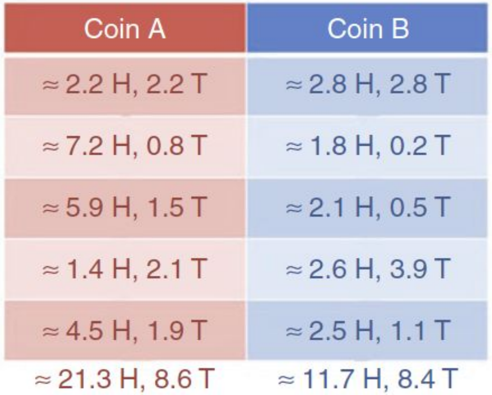
Coin A: $\propto 0.6^{9} \cdot 0.4^{1} \approx 0.00403$

Coin B $\propto 0.5^{9}\ 0.5^{1} = \left(\frac{1}{2}\right)^{10} \approx 0.000976$

$$Pr(Z_2 = A) = \frac{0.6^{9} \cdot 0.4^{1}}{0.6^{9}\, 0.4^{1} + (0.5)^{10}} = 0.8049 \approx 0.8,\ Pr(Z_2 = B) = 0.2$$

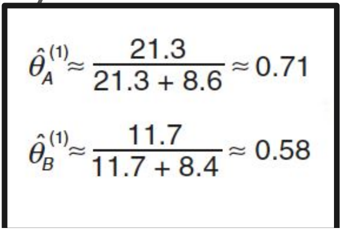
## Two coin problem

- Having done this for all 5 rounds we obtain the following



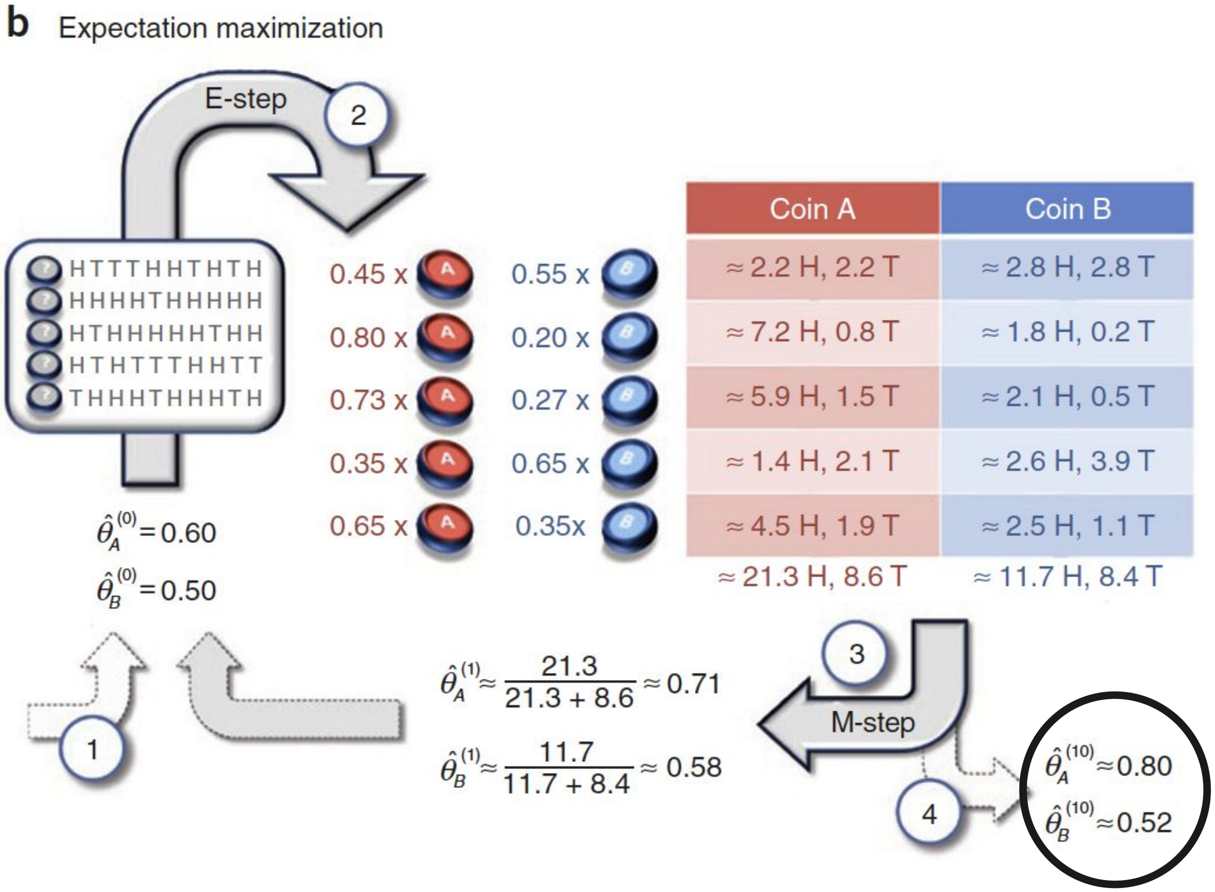
| Coin A | Coin B |
|---|---|
| ≈ 2.2 H, 2.2 T | ≈ 2.8 H, 2.8 T |
| ≈ 7.2 H, 0.8 T | ≈ 1.8 H, 0.2 T |
| ≈ 5.9 H, 1.5 T | ≈ 2.1 H, 0.5 T |
| ≈ 1.4 H, 2.1 T | ≈ 2.6 H, 3.9 T |
| ≈ 4.5 H, 1.9 T | ≈ 2.5 H, 1.1 T |
| ≈ 21.3 H, 8.6 T | ≈ 11.7 H, 8.4 T |

- The M-step now simply becomes the MLEs according to this data



$$\hat{\theta}_A^{(1)} \approx \frac{21.3}{21.3 + 8.6} \approx 0.71$$

$$\hat{\theta}_B^{(1)} \approx \frac{11.7}{11.7 + 8.4} \approx 0.58$$



**b** Expectation maximization

| Coin A | Coin B |
|---|---|
| ≈ 2.2 H, 2.2 T | ≈ 2.8 H, 2.8 T |
| ≈ 7.2 H, 0.8 T | ≈ 1.8 H, 0.2 T |
| ≈ 5.9 H, 1.5 T | ≈ 2.1 H, 0.5 T |
| ≈ 1.4 H, 2.1 T | ≈ 2.6 H, 3.9 T |
| ≈ 4.5 H, 1.9 T | ≈ 2.5 H, 1.1 T |
| ≈ 21.3 H, 8.6 T | ≈ 11.7 H, 8.4 T |

$\hat{\theta}_A^{(0)} = 0.60$

$\hat{\theta}_B^{(0)} = 0.50$

$$\hat{\theta}_A^{(1)} \approx \frac{21.3}{21.3 + 8.6} \approx 0.71$$

$$\hat{\theta}_B^{(1)} \approx \frac{11.7}{11.7 + 8.4} \approx 0.58$$

$\hat{\theta}_A^{(10)} \approx 0.80$

$\hat{\theta}_B^{(10)} \approx 0.52$

# EM algorithm

- Suppose maximizing Pr(x|θ) has no closed form solution/intractable.

- Key idea: latent variables z that make likelihood computations tractable

- Intuition: Pr(x,z|θ), Pr(z|x,θ) should be easy to compute after introducing the “right” latent variables.

- EM guaranteed to **converge** to a *local* maximum!

# EM algorithm

Define the “expected log” Q(θ|θ’) where θ’ is the current estimate of θ:

$$Q(\theta \mid \theta') = \sum_z \Pr\left(z \mid x, \theta'\right) \log \Pr(x, z \mid \theta)$$

The EM algorithm is an iterative method consisting of two steps.

1. E-step: Find  Q(θ|θ’) in terms of the latent variables z

2. M-step: Find θ* maximizing Q(θ|θ’)

# EM-algorithm

Some remarks

1. The quantity Q typically is not computed, but is useful for proving the convergence of the EM algorithm.

2. What we really need to know is the distribution of z given x, our guess θ’, and frequently knowing the expectation of z over its distribution suffices as we saw in our example.

3. The M-step maximizes Q over all possible θ values. Ideally, knowing x and z allows an easy maximization over θ.

## Readings

1. What is the EM algorithm?
2. Wikipedia article
3. Original EM paper
4. Andrew Ng's notes