Generative models

Coin with some intrinsic probability p_H

and $p_T = 1 - p_H$

H T H H T H T H H T

Document classification

Categories – Spam / NotSpam

p_S $1 - p_S$

Set of words $P(w_i \mid \text{spam})$ $P(w_i \mid \text{not spam})$

Generate a document

1. Choose spam / not spam p_S
2. Depending on choice, for each w_i, include w_i with prob. $P(w_i \mid c)$

Richer model – "bag of words"

Bag = multiset = set with a count for each value

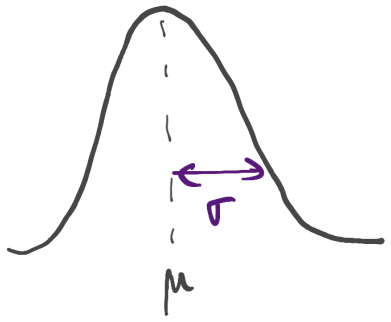
Generalize Spam/Not Spam to Topic Modelling

Google News – automatically assign new stories to categories

What if some of the generating parameters are hidden?

Gaussian

$$\frac{1}{2\sigma^2} e^{-(x-\mu)^2}$$



Given samples $x_1, x_2, \ldots, x_N$

estimate σ, μ

Simpler: σ fixed, estimate μ

Given a hypothesis for μ — $\hat{\mu}$

$P(x_1)$ — prob $\frac{1}{2\sigma^2} e^{-(x_1-\hat{\mu})^2}$

$P(x_2)$ - - -

⋮

$P(x_n)$

Choose $\hat{\mu}$ to maximize $\prod_i P(x_i|\hat{\mu})$

Likelihood: $L(\theta, \mu)$

Maximizing $L(\theta, \mu)$ is messy

Instead take logs

$$\ln\left(L(\theta,\mu)\right) = \ln \prod_{i=1}^{k} P(x_i|\mu)$$

$$= \sum_{i=1}^{k} \ln P(x_i|\mu)$$

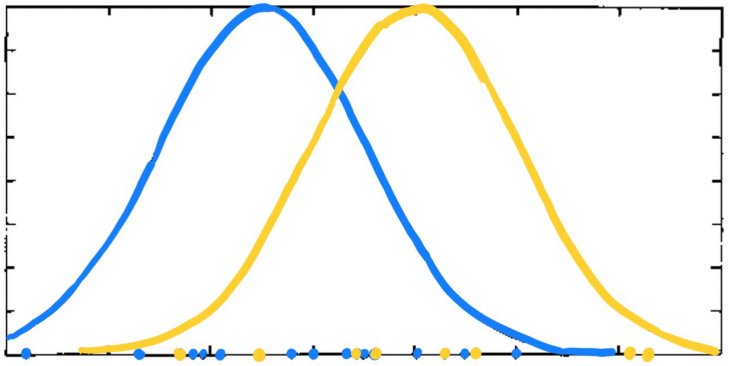
Log likelihood

$\ln \frac{1}{2\sigma^2} e^{-(x_i-\mu)^2}$

Maximize $-(x_i-\mu)^2$

Best choice for μ is sample mean $= \frac{1}{k}\sum_{i=1}^{k} x_i$

Mixture of 2 Gaussians



Know colour of points - apply sample mean separately to blue/yellow points

If we don't know ?

Generative model

Pick blue or yellow distribution (equally likely – uniform)

Generate x_i according to chosen distribution

H T T T H H T H T H

H H H H T H H H H H

H T H H H H H T H H

H T H T T T H H T T

T H H H T H H H T H

2 biased coins

Experiment

Choose one coin (uniform)

Toss it M times

] Repeat K times

	Coin A	Coin B
H T T T H H T H T H		5 H, 5 T
H H H H T H H H H H	9 H, 1 T	
H T H H H H H T H H	8 H, 2 T	
H T H T T T H H T T		4 H, 6 T
T H H H T H H H T H	7 H, 3 T	
	24H 6T	9H 11T
	$P_H = 0.8$	$P_H = 0.45$

How to proceed if info about coins is missing

Initially Guess p_1, p_2

p_H red p_H blue

For each row of 10 tosses – 3H, 7T

$p_1^3 (1-p_1)^7$ – red coin p_R

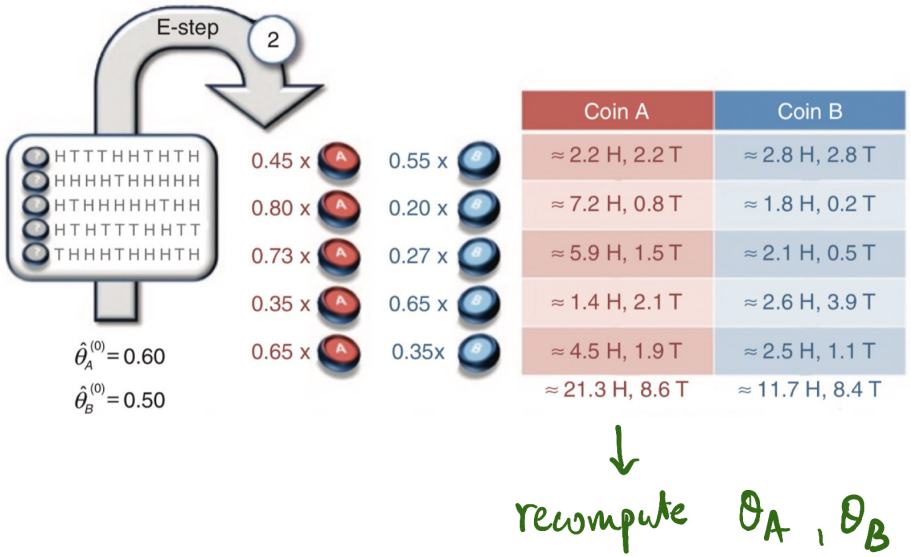
$p_2^3 (1-p_2)^7$ – blue coin p_B

$$\frac{p_R}{p_R+p_B} \sim 0.7 \qquad \frac{p_B}{p_R+p_B} \sim 0.3$$

Split the outcome in this ratio

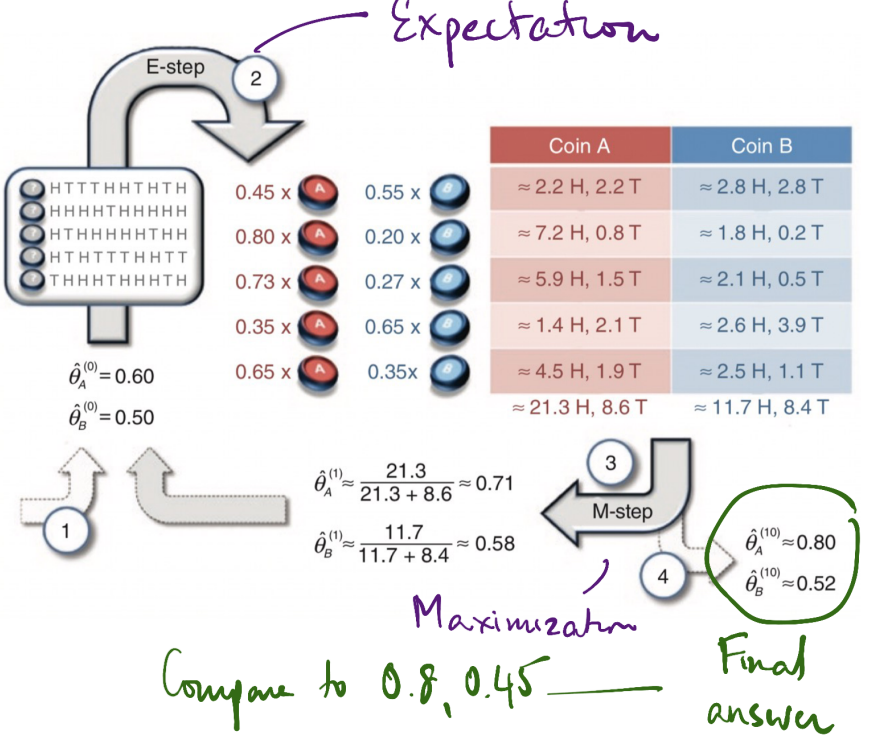
3H, 7T

- Red get 0.7×3H, 0.7×7T
- Blue get 0.3×3H, 0.7×7T



↓ recompute θ_A, θ_B

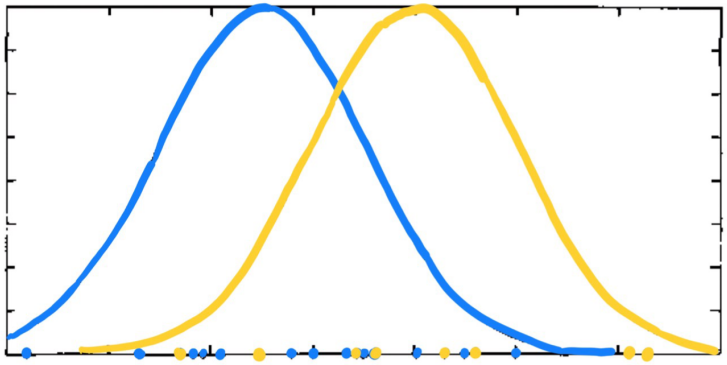
Expectation



Maximization

Final answer

Compare to 0.8, 0.45



Can prove that EM converges if underlying function is convex

Otherwise – local maximum

Choose μ_1, μ_2

For each x_i $\dfrac{P(x_i|\mu_1)}{\sum_{j=1}^{2} P(x_i|\mu_j)}$, $\dfrac{P(x_i|\mu_2)}{\sum_{j=1}^{2} P(x_i|\mu_j)}$

For each x_i – $w_i^1 = \dfrac{P(x_i|\mu_1)}{\Sigma}$

$w_i^2 = 1 - w_i^1 = \dfrac{P(x_i|\mu_2)}{\Sigma}$

Sample mean $\frac{1}{n} \sum x_i$

Weighted sample mean $\sum_{i=1}^{n} \frac{w_i^1 x_i}{w_i^1} = \mu_1$

$\sum_{i=1}^{n} \frac{w_i^2 x_i}{w_i^2} = \mu_2$

Back to topic modelling

What happens if we don't know the topics?

- I like to eat broccoli and bananas.] T_1
- I ate a banana and spinach smoothie for breakfast.] T_1
- Chinchillas and kittens are cute.] T_2
- My sister adopted a kitten yesterday.] T_2
- Look at this cute hamster munching on a piece of broccoli.

Told that there are 2 topics $\alpha T_1 + (1-\alpha) T_2$

K topics : $p_1, p_2 \dots p_k$, $\sum p_i = 1$

For each word w_i & topic t_j $P(w_i | t_j)$

Latent Dirichlet Analysis = LDA

Step 1 For each word in the overall set of docs, randomly assign a topic

Step 2 Assign a mixture of topics to each document

For each word w_i overall

For each topic t_j

Estimate $P(w_i | t_j)$ globally

Reassign a topic to w_i according to these probabilities

Re-estimate topic allocations for each doc

Iterate till convergence

Semi - Supervised learning

Biggest bottleneck of Supervised learning is providing labelled training data