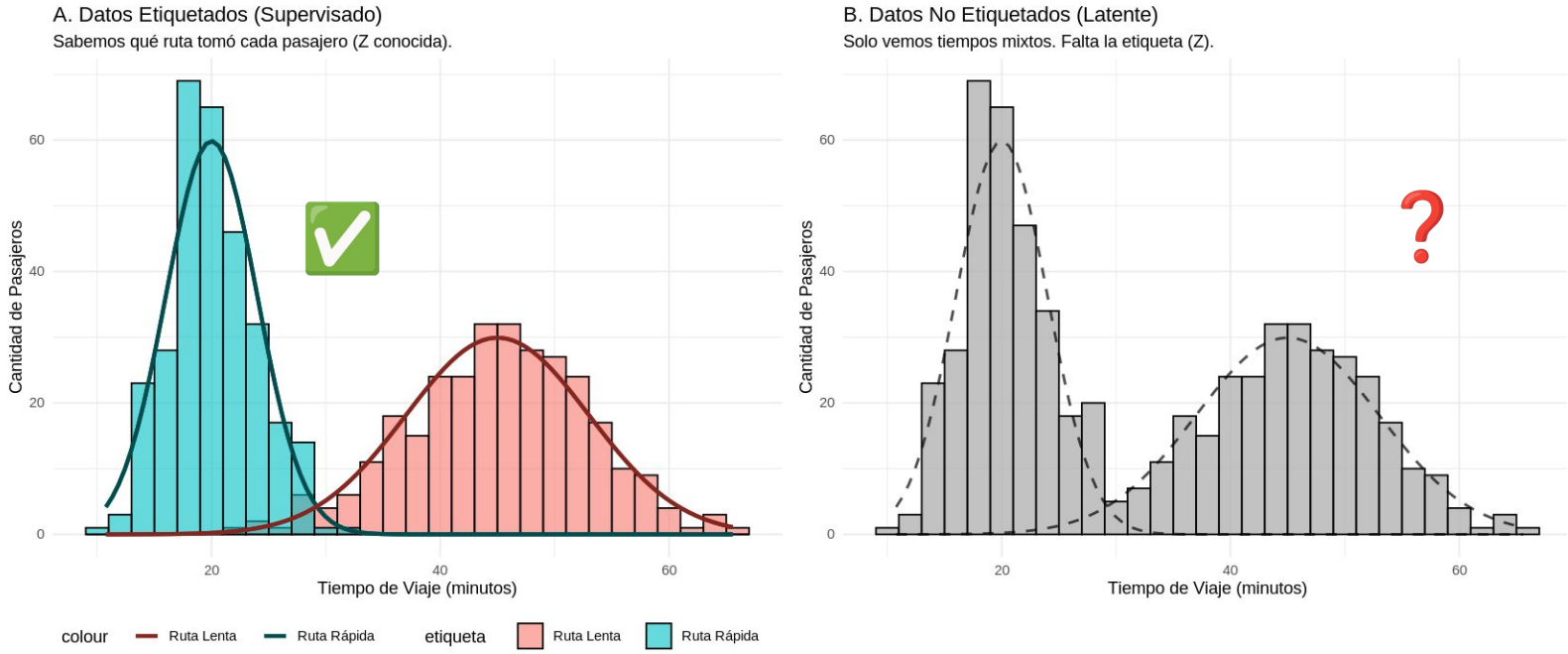
Algoritmo esperanza-maximización (Expectation–maximization algorithm)

Ezequiel Pereyra y Agustín Díaz Barquinero



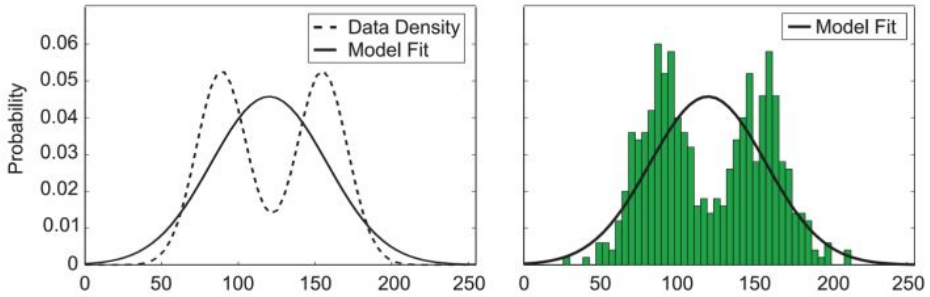
- **El Problema del "Huevo y la Gallina"**

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)$$

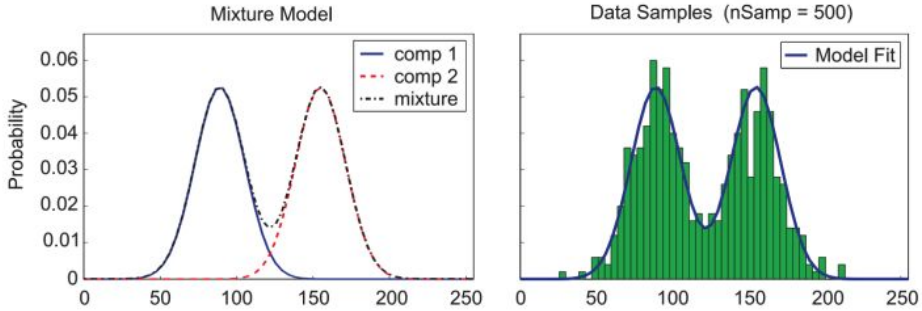
Parámetros

- La Media μ
- La Varianza σ
- El Peso Π

- If you fit a Gaussian to data:



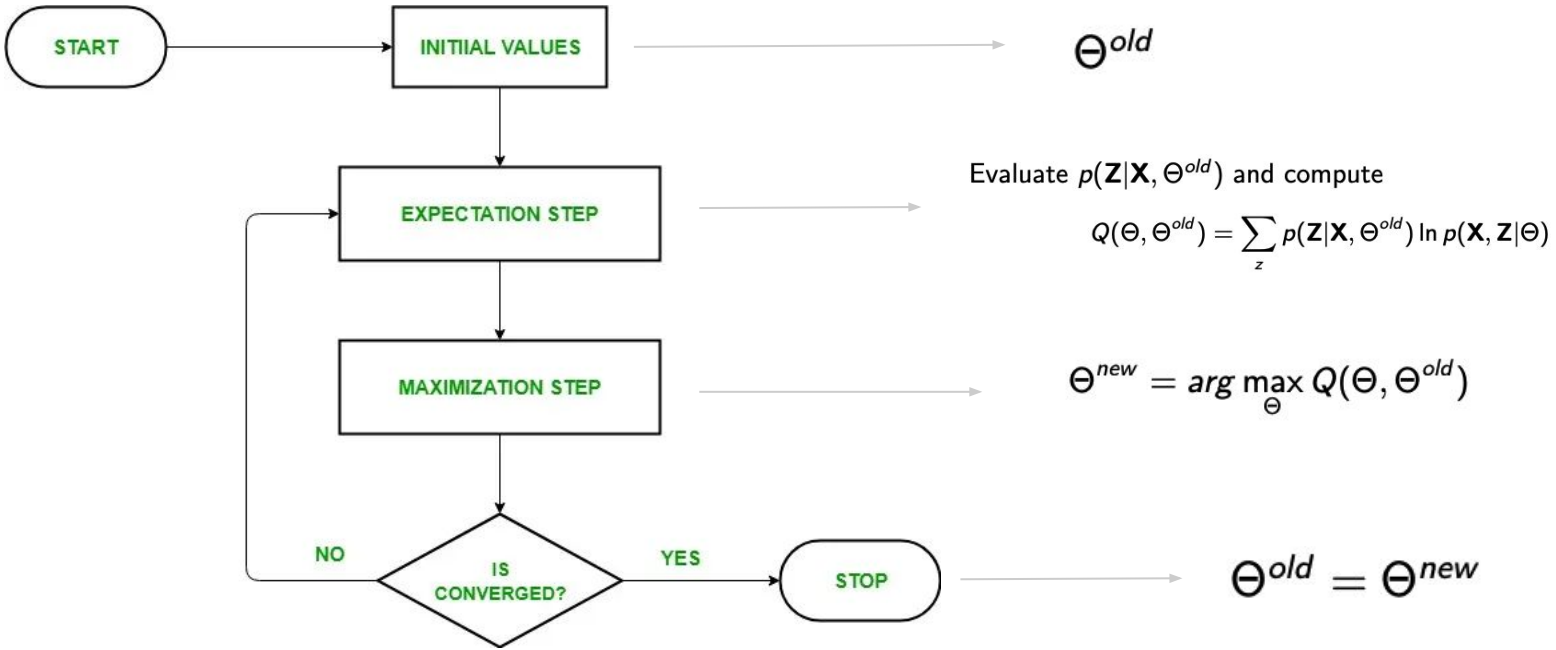
- Now, we are trying to fit a GMM (with $K = 2$ in this example):



[Slide credit: K. Kutulakos]

Rompiendo el Círculo (La Intuición)

La Solución EM: Romper el ciclo empezando con parámetros aleatorios y refinar iterativamente.



Ventajas sobre K-means

- ✔ Captura **clusters elípticos**, no solo esféricos
- ✔ Clusterización por **probabilidad**
- ✔ Cada cluster tiene su propia **covarianza**
- ✔ Mejor para datos con **diferentes escalas o densidades**

Desventajas

- Sensibilidad a la inicialización
- Velocidad de Convergencia

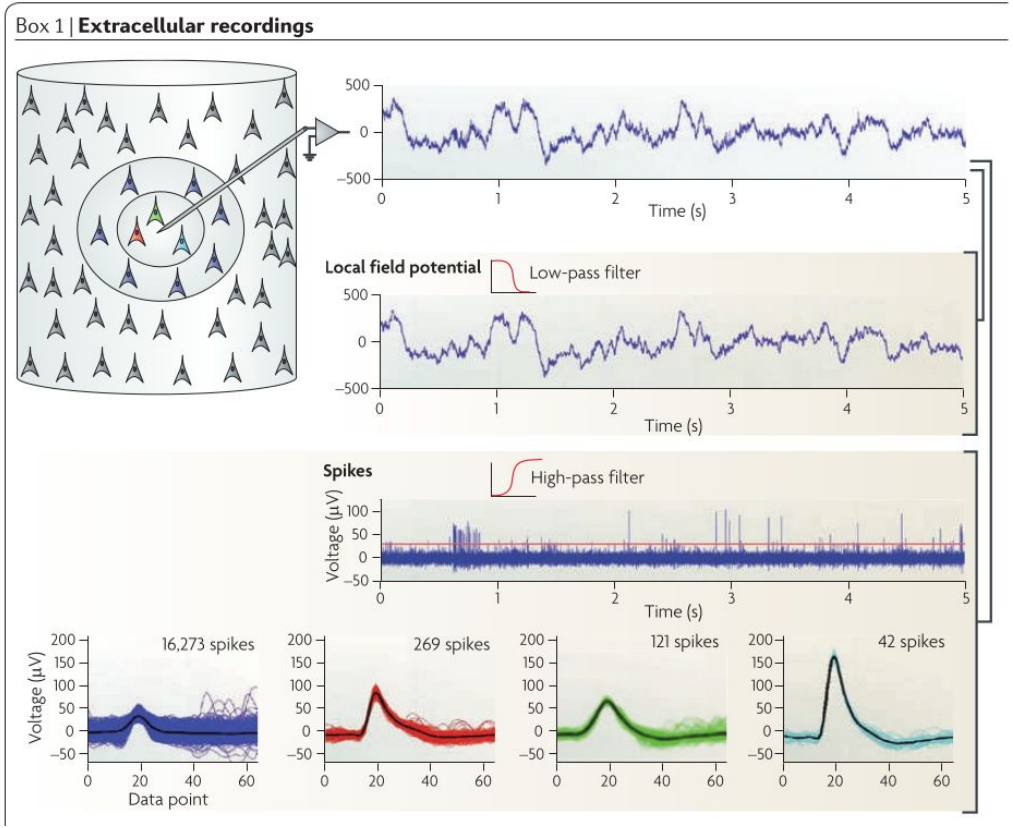
Cuando usarlo

- Cuando tus clusters **no son bien circulares**
- Cuando esperás **superposición** entre grupos
- Cuando querés obtener **probabilidades** de pertenencia
- Para **detectar anomalías** (componentes de baja probabilidad)

Cuando no usarlo

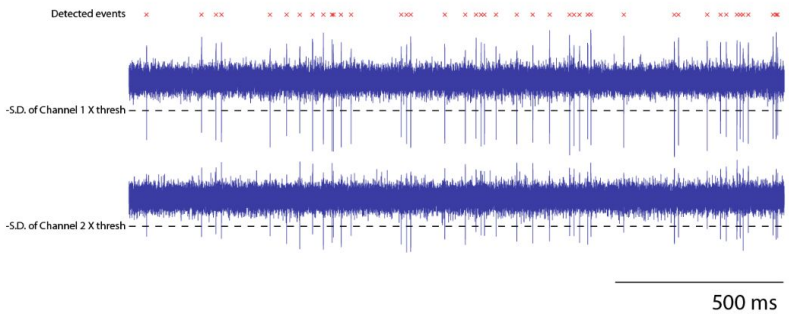
- Cuando la dimensionalidad es muy alta (covarianzas se vuelven difíciles de estimar)
- Cuando los clusters están muy separados y son esféricos → K-means suele ser más simple y rápido
- Cuando tenés pocos datos por componente

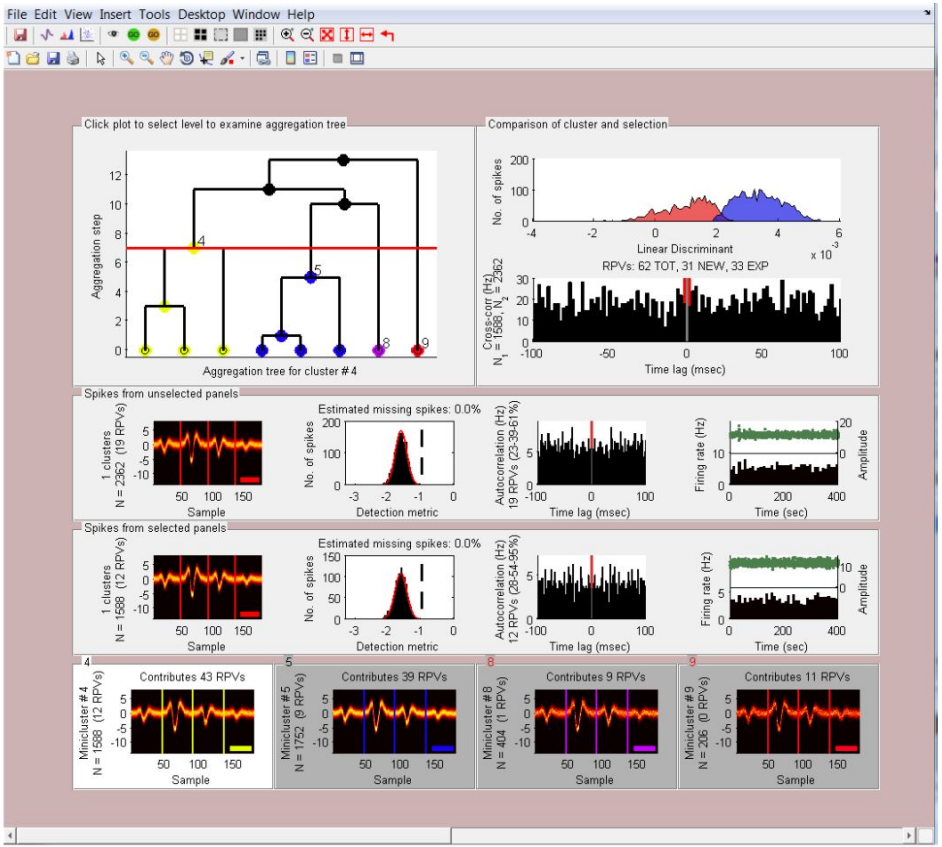
REGISTROS EXTRACELULARES DE ACTIVIDAD NEURONAL



Quian Quiroga y Panzeri, 2009

REGISTROS EXTRACELULARES DE ACTIVIDAD NEURONAL





UltraMegaSort2000
Hill, Mehta y Kleinfeld, 2012

En un **modelo de mezcla de gaussianas**, cada dato x_i se asume generado por uno de los K componentes:

$$x_i \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)$$

Pero **no sabemos** qué componente generó cada punto → esa variable oculta o latente se llama z_{ik} :

$$z_{ik} = \begin{cases} 1, & \text{si el dato } x_i \text{ fue generado por el cluster } k \\ 0, & \text{en caso contrario} \end{cases}$$

E-step (Expectation):

El algoritmo EM calcula para cada dato de entrada la probabilidad de pertenecer a cada componente: $\gamma_{ik} = P(z_{ik} = 1|x_i, \theta)$

Estas γ_{ik} son **responsabilidades**, y son lo que se considera *datos faltantes estimados*. No existían antes → EM los estima.

M-step (Maximization): Usa esas responsabilidades para actualizar los parámetros:

- medias:
  $\mu_k = \frac{\sum_i \gamma_{ik} x_i}{\sum_i \gamma_{ik}}$
- covarianza:
  $\Sigma_k = \frac{\sum_i \gamma_{ik} (x_i - \mu_k)(x_i - \mu_k)^\top}{\sum_i \gamma_{ik}}$
- pesos:
  $\pi_k = \frac{1}{n} \sum_i \gamma_{ik}$

cantidad de parámetros:

$$\underbrace{(K-1)}_{\text{proporciones}} + \underbrace{Kd}_{\text{medias}} + \underbrace{K\frac{d(d+1)}{2}}_{\text{covarianzas}}$$