14. Clustering, Kmeans, GMM

- clustering
- kmeans
- gaussian mixture models

K-means and Clustering

Supervised, semi-supervised, unsupervised

Supervised = there exists a training set

- Teacher gives students some stuff to learn, test is on stuff learned
- MNIST classifier is trained on 60,000 labels, tested on 10,000 labels

Supervised, semi-supervised, unsupervised

Semi-supervised = there is a training set, but it's pretty small and unrepresentative

- Teacher gives some lessons, but test could branch into new subjects
- Doctor is medically trained, but may diagnose a disease never seen before
- Self-driving car sees most scenarios, but may face something new on road

Supervised, semi-supervised, unsupervised

Unsupervised = there is no training set

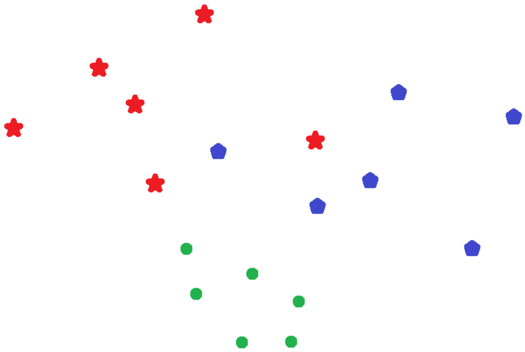
- Student finds patterns in observations, starts to form theories and models
- Amy Adams talks to aliens by identifying structure in language
- I clean my room by putting things in piles, and decide my own labels

Clustering

I've got gadgets and gizmos aplenty
I've got whosits and whatsits galore
You want thingamabobs? I got twenty!

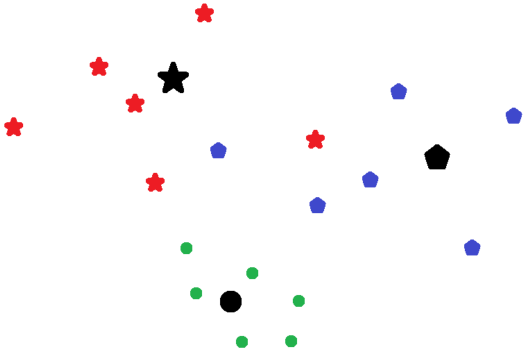
I don't know what they're called, so I'll just categorize them and label them later

Clustering



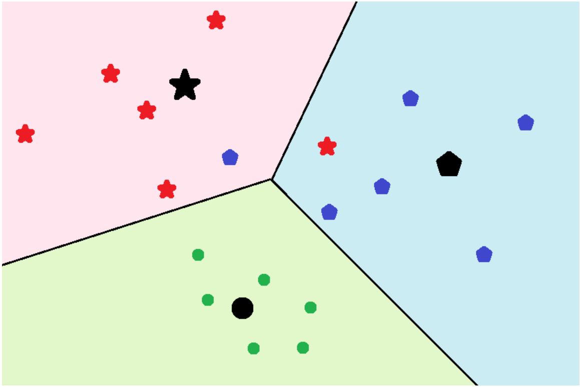
Distance = dissimilarity

Cluster centers



- No labels!
- Ideally: representative center

Clustering into Voronoi cells



Cluster based on closest representative (centers)

K-means algorithm

$$\underset{\mathcal{S}_k,\mu_k}{\text{minimize}} \quad \sum_{i=1}^{m}\sum_{k=1}^{K} \|x_i - \mu_k\|_2$$

Data: $x_1, ..., x_m \in \mathbb{R}^n$, K clusters

- **Init:** Pick some centers $\mu_1^{(0)}, ..., \mu_K^{(0)} \in \mathbb{R}^n$
- **Iterate:** $t = 1, ...$
  - Classify each point based on closest center (e.g. KNN)

    $$i \in \mathcal{S}_k^{(t)} \text{ if } k = \underset{k=1,\ldots,K}{\text{argmin}} \|x_i - \mu_k^{(t-1)}\|_2, \quad k = 1, ..., K$$

  - Recompute centers $\mu_k^{(t)} = \frac{1}{|\mathcal{S}_k^{(t)}|}\sum_{i\in\mathcal{S}_k^{(t)}} x_i$
- **Until** Convergence $\mathcal{S}_k^{(t)} = \mathcal{S}_k^{(t-1)},\ k = 1, ..., K$

Optimality

$$\underset{\mathcal{S}_k,\mu_k}{\text{minimize}} \quad \sum_{k=1}^{K} \sum_{i \in \mathcal{S}_k} \|x_i - \mu_k\|_2 \qquad (\star)$$

- Does it converge?
- Does it always converge to the same point?

Optimality

$$\underset{\mathcal{S}_k,\mu_k}{\text{minimize}} \quad \sum_{k=1}^{K} \sum_{i \in \mathcal{S}_k} \|x_i - \mu_k\|_2 \qquad (\star)$$

- Does it converge?

  Ans: Yes, objective value decreases each step, bounded below by 0.

- Does it always converge to the same point?

Optimality

$$\underset{\mathcal{S}_k, \mu_k}{\text{minimize}} \quad \sum_{k=1}^{K} \sum_{i \in \mathcal{S}_k} \|x_i - \mu_k\|_2 \qquad (\star)$$

- Does it converge?

  Ans: Yes, objective value decreases each step, bounded below by 0.

- Does it always converge to the same point?

  Ans: No. What happens if we initialize

  $$\mu_1^{(0)} = \mu_2^{(0)} = \cdots = \mu_K^{(0)} = \frac{1}{m} \sum_i x_i ?$$

  Usually start with random initialization.

Optimality

$$\underset{\mathcal{S}_k,\mu_k}{\text{minimize}} \quad \sum_{k=1}^{K} \sum_{i \in \mathcal{S}_k} \|x_i - \mu_k\|_2 \qquad (\star)$$

- Does it converge?

  Ans: Yes, objective value decreases each step, bounded below by 0.

- Does it always converge to the same point?

  Ans: No. What happens if we initialize

  $$\mu_1^{(0)} = \mu_2^{(0)} = \cdots = \mu_K^{(0)} = \frac{1}{m}\sum_i x_i?$$

  Usually start with random initialization.

- $(\star)$ is <u>nonconvex</u>, may have multiple (not that great) local optima

Extensions

$$\underset{\mathcal{S}_k,\mu_k}{\text{minimize}} \quad \sum_{k=1}^{K}\sum_{i\in\mathcal{S}_k} d(x_i - \mu_k)$$

- If $d(x) = \|x\|_2$, we are solving K-means.
- If $d(x) = \|x\|_1$, we are solving K-median. Specifically,

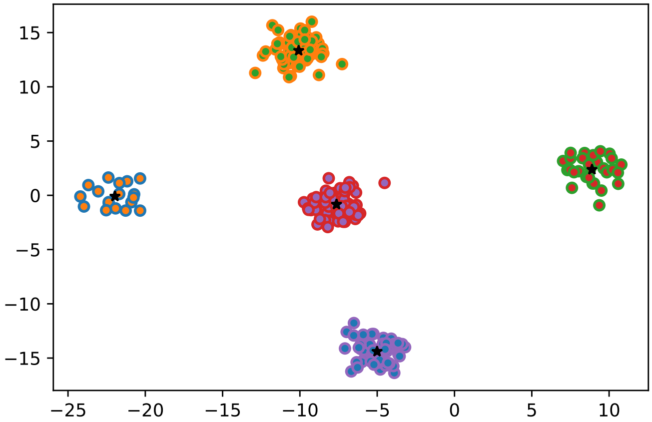
$$\mu = \underset{\mu}{\operatorname{argmin}} \sum_{i\in\mathcal{S}} \|x_i - \mu\|_1$$

  recovers $\mu =$ the median of x_i, $i \in \mathcal{S}$.

  This formulation is more robust to outliers.

- We can choose d as we want, but optimization may be harder.

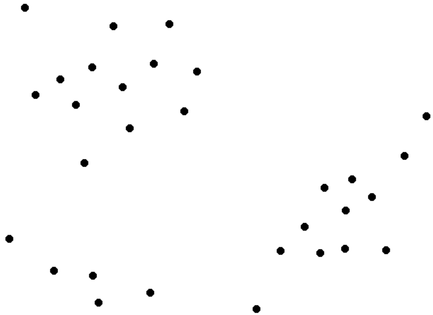
Go to demo



Gaussian mixture models and data modeling

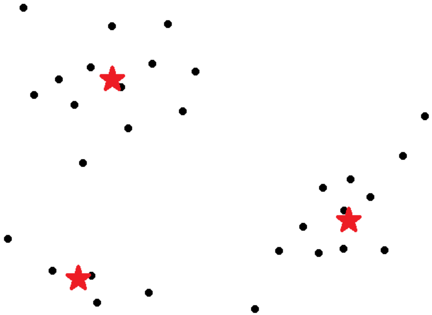
Soft cluster assignments

- How to quantify uncertainty of an assignment $i \in \mathcal{S}_k$?
- Clusters may have different shapes, eccentricites
- I want a generative data model $\Pr(x_i)$, not just a cluster assignment



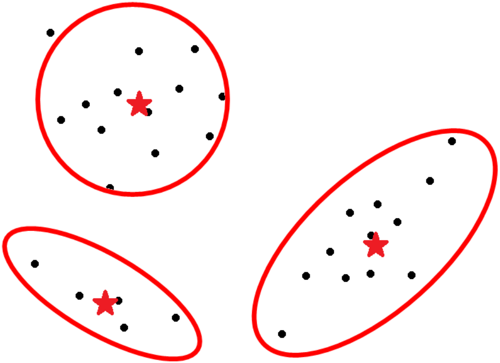
Soft cluster assignments

- How to quantify uncertainty of an assignment $i \in \mathcal{S}_k$?
- Clusters may have different shapes, eccentricites
- I want a generative data model $\Pr(x_i)$, not just a cluster assignment



Soft cluster assignments

- How to quantify uncertainty of an assignment $i \in \mathcal{S}_k$?
- Clusters may have different shapes, eccentricites
- I want a generative data model $\Pr(x_i)$, not just a cluster assignment



K-means with indicator variables

Data: $x_1, ..., x_m \in \mathbb{R}^n$, K clusters

- **Init:** Pick some centers $\mu_1^{(0)}, ..., \mu_K^{(0)} \in \mathbb{R}^n$
- **Iterate:** $t = 1, ...$
  - Classify each point based on closest center (assume unique)

$$z_{i,k}^{(t)} = \begin{cases} 1 & \text{if } k = \underset{k=1,\ldots,K}{\operatorname{argmin}} \|x_i - \mu_k^{(t-1)}\|_2, \\ 0 & \text{else.} \end{cases}$$

  - Recompute centers $\mu_k^{(t)} = \frac{\sum_{i=1}^m z_{i,k} x_i}{\sum_{i=1}^m z_{i,k}}$
- **Until** Convergence $z_i^{(t)} = z_i^{(t-1)}$, $i = 1, ..., m$

Gaussian mixture models (GMM)

Gaussian mixture model

$$\Pr(x_i|z_{i,k}=1,\theta_k) = \underbrace{\frac{1}{\sqrt{2\pi|C_k|}}\exp\left(-\frac{1}{2}(x_i-\mu_k)^T C_k^{-1}(x_i-\mu_k)\right)}_{=:f_{\mathcal{N}}(x_i;\mu_k,C_k)}$$

Mixture coefficient

$$\Pr(x_i, z_{i,k}=1|\theta) = \Pr(x_i|z_{i,k}=1,\theta)\underbrace{\Pr(z_{i,k}=1|\theta)}_{=:\alpha_k}$$

Distribution parameters: $\theta = (\alpha, \mu, C)$

$$\underbrace{\alpha \in \Delta_{K-1}}_{\text{mixture coeffs}}, \quad \underbrace{\mu_k \in \mathbb{R}^n}_{\text{mean}}, \quad \underbrace{C_k \in \mathbb{R}^{n\times n} \text{ PSD}}_{\text{covariance}}, \quad k = 1, ..., K$$

where $\Delta_{K-1} := \{0 \le \alpha \in \mathbb{R}^K : \sum_k \alpha_k = 1\}$ is the unit simplex

Gaussian mixture models (GMM)

The assumption

$$\begin{aligned}
\Pr(z_{i,k}=1|x_i,\theta) &\overset{\text{Bayes' formula}}{=} \frac{\Pr(z_{i,k}=1)\Pr(x_i|z_{i,k}=1)}{\sum_{l=1}^{K}\Pr(z_{i,l}=1)\Pr(x_i|z_{i,l}=1)} \\
&= \frac{\alpha_k f_{\mathcal{N}}(x_i;\mu_k,C_k)}{\sum_{l=1}^{K}\alpha_l f_{\mathcal{N}}(x_i;\mu_l,C_l)}
\end{aligned}$$

Reminder

$$f_{\mathcal{N}}(x;\mu,C)=\frac{1}{(2\pi)^{n/2}\sqrt{|C|}}\exp\left(-\frac{1}{2}(x-\mu)^T C^{-1}(x-\mu)\right)$$

Here,

- $\theta=(\mu,C)$ (model parameters)
- $\alpha_k=\Pr(z_{i,k}=1)\in\Delta_{K-1}$
- $|C|=$ determinant of C

Log likelihood objective function

$$
\begin{aligned}
\log(\Pr(z|x,\theta)) &= \log\left(\prod_i \prod_k \Pr(z_{i,k}|x_i,\theta)^{z_{i,k}}\right) \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k}\log(\Pr(z_{i,k}=1|x_i,\theta) \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k}\log(\alpha_k f_{\mathcal{N}}(x_i;\mu_k,C_k)) - \overbrace{\sum_{i=1}^{n}\underbrace{\sum_{k=1}^{K} z_{i,k}}_{=1} B}^{\text{constant}}
\end{aligned}
$$

where $B = \log\left(\sum_{l=1}^{K}\alpha_l f_{\mathcal{N}}(x_i;\mu_l,C_l)\right)$

Log likelihood objective function

$$
\begin{aligned}
& \max_{\theta, z_{i,k}} \log(\Pr(z|x,\theta)) \\
\iff \quad & \max_{\alpha_k \in \Delta_{K-1}, \mu_k, C_k, z_{i,k}} \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k} \log(\alpha_k f_{\mathcal{N}}(x_i; \mu_k, C_k)) \\
\iff \quad & \max_{\alpha_k \in \Delta_{K-1}, \mu_k, C_k, z_{i,k}} \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k} \log(\alpha_k) \\
& \qquad -\frac{1}{2} \left(\log(|C|) + \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k}((x_i - \mu_k)^T C_k^{-1} (x_i - \mu_k) \right)
\end{aligned}
$$

3 maximization problems: Mixing coefficients α_k

$$\max_{\alpha \in \Delta_{K-1}} \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k} \log(\alpha_k), \quad z_{i,k} \in \{0,1\}, \quad \sum_k z_{i,k} = 1, \ \forall i$$

- Denote $s_k = \frac{1}{n} \sum_{i=1}^{n} z_{i,k}$. Note that

$$\sum_{k=1}^{K} s_k = \frac{1}{n} \sum_{i=1}^{n} \underbrace{\sum_{k=1}^{K} z_{i,k}}_{=1} = 1$$

- Therefore we can reduce the problem to

$$\max_{\alpha \in \Delta_{K-1}} \sum_{k=1}^{K} s_k \log(\alpha_k), \quad 0 \leq s_k \leq 1, \quad \sum_k s_k = 1$$

- I would like to say $\alpha_k = s_k$, but how to prove?

3 maximization problems: Mixing coefficients α_k

$$\max_{\alpha \in \Delta_{K-1}} \sum_{k=1}^{K} s_k \log(\alpha_k), \quad 0 \le s_k \le 1, \quad \sum_k s_k = 1 \qquad (\star)$$

- Consider $K = 2$. Then $s_1 + s_2 = \alpha_1 + \alpha_2 = 1$ and

$$\max_{\alpha_1} \; s_1 \log(\alpha_1) + (1 - s_1) \log(1 - \alpha_1)$$

has optimum $\alpha_1 = s_1$, $\alpha_2 = 1 - \alpha_1 = s_2$.

3 maximization problems: Mixing coefficients α_k

$$\max_{\alpha \in \Delta_{K-1}} \sum_{k=1}^{K} s_k \log(\alpha_k), \quad 0 \le s_k \le 1, \quad \sum_k s_k = 1 \qquad (\star)$$

- Recursively, suppose $\alpha_i = s_i$ for $i = 1, ..., K-1$. Then

$$\begin{aligned}
(\star) \quad &= \quad \max_{0 \le \alpha_K \le 1} \left(\max_{\alpha \in (1-\alpha_K)\Delta_{K-2}} \sum_{k=1}^{K-1} s_k \log(\alpha_k) \right) + s_K \log(\alpha_K) \\
&= \quad \max_{0 \le \alpha_K \le 1} \sum_{k=1}^{K-1} s_k \log(s_k \cdot (1-\alpha_K)) + s_K \log(\alpha_K) \\
&\iff \quad \max_{0 \le \alpha_K \le 1} \left(\sum_{k=1}^{K-1} s_k \right) \log((1-\alpha_K)) + s_K \log(\alpha_K)
\end{aligned}$$

which reduces to the $K=2$ case with optimum $\alpha_K = s_K$

3 maximization problems: Mixing coefficients α_k

- Overall,

$$\alpha_k^* = s_i = \frac{1}{n}\sum_{i=1}^{n} z_{i,k}, \qquad k = 1, ..., K$$

3 maximization problems: Mean μ

$$\min_{\mu_k} \sum_{i=1}^{n} \frac{z_{i,k}}{2} (x_i - \mu_k)^T C_k^{-1} (x_i - \mu_k)$$

- Given C, x, z, the minimization is convex in μ
- Set $\nabla = 0$ to find stationary point:

$$C_k^{-1} \sum_{i=1}^{n} z_{i,k}(x_i - \mu_k) = 0 \iff \mu_k^* = \frac{1}{\sum_{i=1}^{n} z_{i,k}} \sum_{i=1}^{n} z_{i,k} x_i$$

3 maximization problems: Inverse covariance $S_k = C_k^{-1}$

$$\min_{S_k := C_k^{-1}} \left(\sum_{i=1}^{n} \underbrace{-z_{i,k} \log(|S_k|)}_{\text{convex in } S} + \underbrace{z_{i,k}(x_i - \mu_k)^T S_k (x_i - \mu_k)}_{\text{linear in } S} \right)$$

- Setting $\nabla = 0$ gives

$$\sum_{i=1}^{n} -z_{i,k} S^{-1} = \sum_{i=1}^{n} z_{i,k}(x_i - \mu_k^*)(x_i - \mu_k^*)^T$$

- Simplify, plug back in $C_k = S_k^{-1}$

$$C_k^* = \frac{1}{\sum_{i=1}^{n} z_{i,k}} \sum_{i=1}^{n} z_{i,k}(x_i - \mu_k^*)(x_i - \mu_k^*)^T$$

- This is a weighted covariance matrix

What about the hidden variables?

- I don't actually know $z_{i,k} \in \{0, 1\}$!
- Training GMMs: use soft weights

$$\pi_{i,k} := \Pr(z_{i,k} = 1 | x_i, \theta) = \frac{\alpha_k f_{\mathcal{N}}(x_i; \mu_k, C_k)}{\sum_{l=1}^{K} \alpha_l f_{\mathcal{N}}(x_i; \mu_l, C_l)}$$

- Training for $z_{i,k}$ rather than $\pi_{i,k}$ causes

$$\max_\theta \log(\Pr(x, z|\theta)) \rightarrow \max_\theta \mathbb{E}_{z|x,\bar{\theta}}[\log(\Pr(x, z|\theta))]$$

hence the name "Expectation maximization"

Training a Gaussian Mixture Model

Data: $x_1, ..., x_m \in \mathbb{R}^n$, K clusters

- **Init:** $\mu_k^{(0)}$ somewhere, $\Sigma_k^{(0)} = I$, $\alpha_k^{(0)} = 1/K$
- **Iterate:** $t = 1, ...$
    - **(E)** Update soft indicator π given α, μ, C

$$\pi_{i,k}^{(t)} = \frac{\alpha_k p_{\mu_k^{(t-1)}, C_k^{(t-1)}}(x_i)}{\sum_{j=1}^{K} \alpha_j p_{\mu_j^{(t-1)}, C_j^{(t-1)}}(x_i)}$$

    - **(M)** Update α, μ, C given z

$$\alpha_k^{(t)} = \frac{1}{m} \sum_{i=1}^{m} \pi_{i,k}^{(t)}, \qquad \mu_k^{(t)} = \frac{1}{\sum_i \pi_{i,k}^{(t)}} \sum_{i=1}^{m} \pi_{i,k}^{(t)} x_i,$$

$$C_k^{(t)} = \frac{1}{\sum_i \pi_{i,k}^{(t)}} \sum_{i=1}^{m} \pi_{i,k}^{(t)} (x_i - \mu_k^{(t)})(x_i - \mu_k^{(t)})^T$$

- **Until** convergence

Summary

- Clustering: our first unsupervised learning task
- K-means: hacky-but-still-principled way of finding clusters
- Gaussian mixture models: a generative model that allows for "soft weights" π on cluster identities z