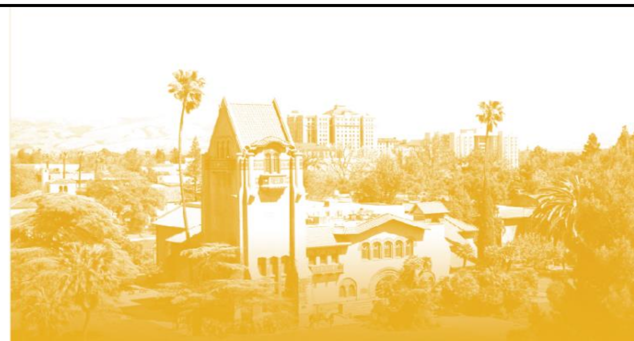


SAN JOSÉ STATE UNIVERSITY

Vector Data – Clustering
Gaussian Mixture Models

1

SAN JOSÉ STATE UNIVERSITY

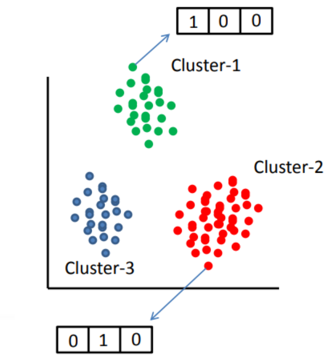
Agenda

- Recap of K-Means
- Mixture Models
- Gaussian Mixture Models (GMM)

2

K-Means Clustering

- K-Means is one of the most popular clustering algorithms.
- Input:
  - Observations/data points (N): $x_i \quad \forall\, i \in \{1, \dots, N\}$
  - # of clusters: k
- Output:
  - Cluster Assignments: w_{ij}
  - Cluster Centroids: $c_j \quad \forall\, j \in \{1, \dots, k\}$



1-of-k representation for cluster assignment.

- Objective Function: It aims to minimize the within-cluster sum of squares (WCSS):

3

K-Means Clustering

- K-Means is a minimization problem with the objective function:

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} (x_i - c_j)^2 = \sum_{j=1}^{k} \sum_{i=1}^{N} w_{ij} (x_i - c_j)^2 \qquad w_{ij} = \begin{cases} 1 & if\ x_i \in C_{ij} \\ 0 & if\ x_i \notin C_j \end{cases}$$

- We need to find w_{ij} and c_j ➔ minimize J

$$c_k = \frac{\sum_{i=1}^{N} w_{ik} x_i}{\sum_{i=1}^{N} w_{ij}}$$

4

SJSU SAN JOSÉ STATE UNIVERSITY

K-Means Example & Algorithm

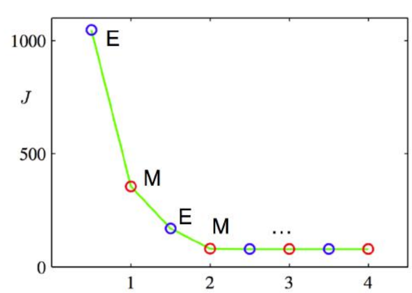
each object is randomly assigned to a cluster | cluster centroids are computed | each object is assigned to the nearest centroid | new cluster centroids are computed | results after 10 iterations

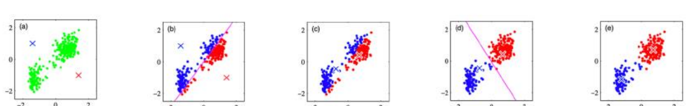
Iterative Algorithm For K-Means
- Initialize k centroids
- Repeat till convergence
  - Calculate w_{ij} ← Expectation Step (E-Step)
  - Update $c_j = \frac{\sum_{i=1}^{N} w_{ij} x_i}{\sum_{i=1}^{N} w_{ij}}$ ← Maximization Step (M-Step)

5

SJSU SAN JOSÉ STATE UNIVERSITY

K Means as an E-M Algorithm





Initialize | E-step | M-step | E-step | M-step | ...

6

SJSU SAN JOSÉ STATE UNIVERSITY

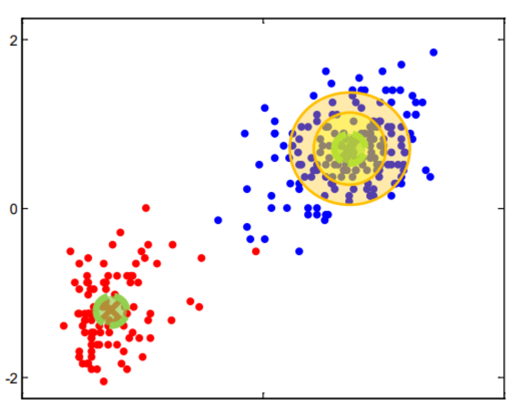
Pros & Cons

- Good
  - Simple to implement
  - Fast
- Bad
  - Local minima
  - Model only "spherical" clusters
  - Sensitive to the features scale
  - Number of clusters K to be chosen in advance
  - Cluster assignments are "hard", not probabilistic => Gaussian Mixture Model

7

SJSU SAN JOSÉ STATE UNIVERSITY

Problem with K-Means

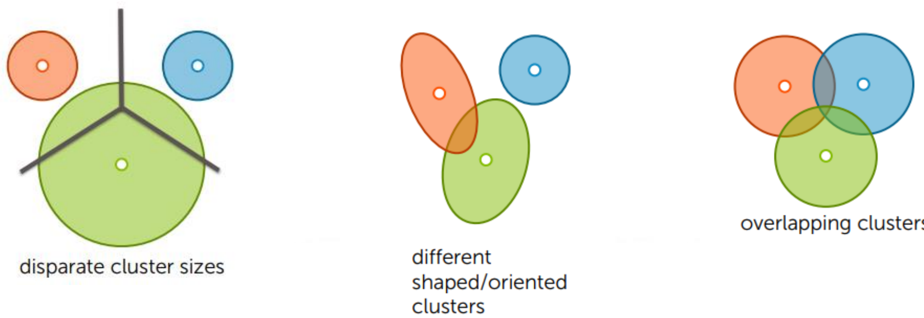


- K-Means makes hard guesses for cluster assignment.
- For some cases our model may not be sure about exact cluster assignment.
- Can we make this probabilistic (probability that the n^{th} observation belongs to the k^{th} cluster) ?

8

SJSU SAN JOSÉ STATE UNIVERSITY

Different Failure Modes of K-Means



9

SJSU SAN JOSÉ STATE UNIVERSITY

Mixture Models

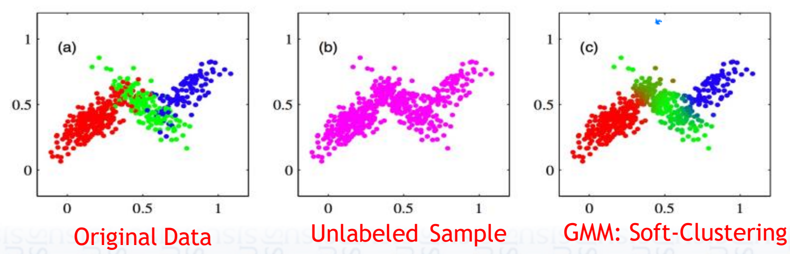
10

Hard Clustering vs. Soft Clustering

- Hard Clustering
  - Every object i is assigned to one cluster j (e.g., k-means)
  - w_i = {0, 1} and $\sum_j w_{ij}$ = 1

$$z_i = \arg\min_j \|x_i - \mu_j\|^2$$

- Soft Clustering
  - Every object i is assigned with a probability to different clusters
  - w_i = [0, 1] and $\sum_j w_{ij}$ = 1



11

Latent Variables

- Latent variables are "hidden" or unobserved variables that influenced the observed data.
- In clustering, latent variables z often represent the cluster to which an observation or data point belongs:
  - cluster assignment for K-Means
  - probability of belonging to a cluster for GMMs and other probabilistic models

12

SJSU SAN JOSÉ STATE UNIVERSITY

Mixture Models

$$p(z, \boldsymbol{x}) = \underbrace{p(z)}_{\text{prior probability of assignment}} \overbrace{p(\boldsymbol{x} \mid z)}^{\text{likelihood}}$$

- Prior probability encodes our belief about the latent variable z (cluster assignment) before observing any data.

$$p(z = k) = \pi_k \qquad 0 \le \pi_k \le 1 \qquad \sum_{k=1}^{K} \pi_k = 1$$

- Likelihood captures the probability of the data $\boldsymbol{x}$ given a cluster assignment z.
- Recall that the law of total probability ➔ the marginal probability of the data $\boldsymbol{x}$:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} p(z = k)\, p(\boldsymbol{x} \mid z = k) = \sum_{k=1}^{K} \pi_k\, p(\boldsymbol{x} \mid z = k)$$

13

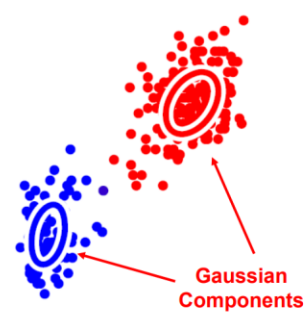
SJSU SAN JOSÉ STATE UNIVERSITY

Gaussian Mixture Model

- One of the most common mixtures are over Gaussians:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k\, p(\boldsymbol{x} \mid z = k) = \sum_{k=1}^{K} \pi_k\, \mathcal{N}(\boldsymbol{x}; \mu_k, \boldsymbol{\Sigma}_k)$$

- Unlike K-Means, assignment (probabilities) aren't just distance.



14

SJSU SAN JOSÉ STATE UNIVERSITY

Soft GMM Assignments (Responsibilities)

- Using Bayes' rule, we have the posterior probability/inference for a given data point $\boldsymbol{x}_i$:

$$p(z_i = k \mid \boldsymbol{x}_i) = \frac{p(z_i = k) \cdot p(\boldsymbol{x}_i \mid z_i = k)}{p(\boldsymbol{x})}$$

- For Gaussian mixtures, this is:

$$p(z_i = k \mid \boldsymbol{x}_i) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k)}{p(\boldsymbol{x})}$$

- In mixture modeling, this is called the responsibility (it is how "responsible" cluster k is for data point $\boldsymbol{x}_i$):

$$\gamma(z_i = k) = p(z_i = k \mid \boldsymbol{x}_i) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_i; \mu_j, \boldsymbol{\Sigma}_j)}$$

15

SJSU SAN JOSÉ STATE UNIVERSITY

Gaussian Mixture Model Summary

- Mixture model is a weighted combination of component distributions:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \, p(\boldsymbol{x} \mid z = k)$$

- Bayes' rule gives the posterior probability/inference of assignment (responsibility):

$$p(z_i = k \mid \boldsymbol{x}_i) = \frac{p(z_i = k) \cdot p(\boldsymbol{x}_i \mid z_i = k)}{p(\boldsymbol{x})}$$

- A GMM uses Gaussian component distributions with responsibilities:

$$p(z_i = k \mid \boldsymbol{x}_i) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_i; \mu_j, \boldsymbol{\Sigma}_j)}$$

16

SJSU SAN JOSÉ STATE UNIVERSITY

Parameter Estimation for GMMs

- For data $\boldsymbol{x}$ that's N-dimensional:

mixture probabilities or proportions

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

N-dimensional vector of mean parameters

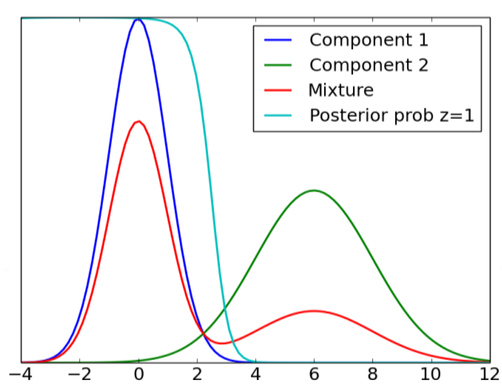
$N \times N$-dimensional matrix of covariance parameters

- For K components, we need to estimate (or train): $O(K + KN + KN^2)$ parameters!!!

17

SJSU SAN JOSÉ STATE UNIVERSITY

Example: Mixture of 2 Gaussians

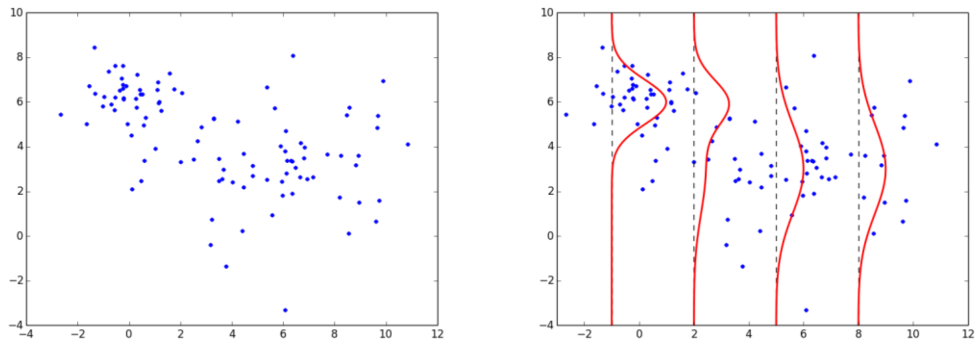


$z \sim \text{Multinomial}(0.7, 0.3)$
$\boldsymbol{x} \mid z = 1 \sim \text{Gaussian}(0, 1)$
$\boldsymbol{x} \mid z = 2 \sim \text{Gaussian}(6, 2)$

18

SJSU SAN JOSÉ STATE UNIVERSITY

Example: Mixture of Gaussians in 2D



$$
\begin{aligned}
z &\sim \text{Multinomial}(0.4, 0.6)\\
x_1 \mid z = 1 &\sim \text{Gaussian}(0, 1)\\
x_1 \mid z = 1 &\sim \text{Gaussian}(6, 1)\\
x_2 \mid z = 2 &\sim \text{Gaussian}(6, 2)\\
x_2 \mid z = 2 &\sim \text{Gaussian}(3, 2)
\end{aligned}
$$

SJSU SAN JOSÉ STATE UNIVERSITY

Parameter Estimation Using MLE

- We need to fit two sets of parameters:
  - The mixture probabilities π_k
  - The mean μ_k and standard deviation $\boldsymbol{\Sigma}_k$for each component
- Recall that the likelihood of a single data point $\boldsymbol{x}_i$ is given by:

$$p(\boldsymbol{x}_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k)$$

- For N points, the log likelihood is:

$$\mathcal{L}(\mu, \boldsymbol{\Sigma}) = \log \prod_{i=1}^{N} p(\boldsymbol{x}_i) = \sum_{i=1}^{N} \log \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k) \right\}$$

Parameter Estimation Using MLE

- We need to maximize $\mathcal{L}(\mu, \boldsymbol{\Sigma})$ wrt π_k, μ_k and $\boldsymbol{\Sigma}_k$

$$\mathcal{L}(\mu, \boldsymbol{\Sigma}) = \log \prod_{i=1}^{N} p(\boldsymbol{x}_i) = \sum_{i=1}^{N} \log \left\{ \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\boldsymbol{x}_i; \mu_k, \boldsymbol{\Sigma}_k) \right\}$$

- This is highly non-convex ➔ very difficult to optimize!!!
- Instead, we try to find the local minima by using an iterative algorithm ➔ EM method (Expectation-Maximization)

21

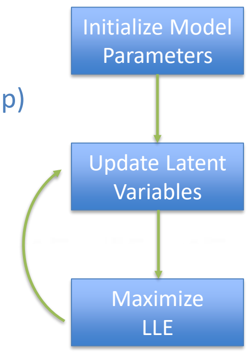
Expectation Maximization (EM) Method

- A very powerful method for dealing with probabilistic models that involve latent/missing variables.
- Each iteration of the EM is guaranteed to maximize the data log likelihood.
- Guaranteed to converge to a local maxima.
- Sensitive to starting points.
- We have applied it to Gaussian Mixture Models, which can model any arbitrary shaped densities. Can be used for data density estimation aside from clustering

22

Expectation Maximization Method

- Here's the overview:
  - Start with initial guesses for the model parameters: $\pi_k, \mu_k, \boldsymbol{\Sigma}_k$
  - Update latent variables based on our expectations (E-Step)
  - Update model parameters to maximize log likelihood estimaes (M-Step)
  - Keep repeating steps 2 and 3 until changes in $\mathcal{L}(\mu, \boldsymbol{\Sigma})$ is small



23

Expectation Step (E-Step)

- This is similar to the cluster assignment step in K-Means, except that we update the (fractional) responsibilities here:

$$\gamma(z_i = k) = p(z_i = k \mid \boldsymbol{x}_i) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

- This is the posterior inference that was discussed earlier.

24

SJSU SAN JOSÉ STATE UNIVERSITY

Maximization Step

- Define the effective number of points in cluster k by:

$$N_k = \sum_{j=1}^{N} \gamma(z_j = k)$$

- Update model parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ with updated responsibilities $\gamma(z_i = k)$:

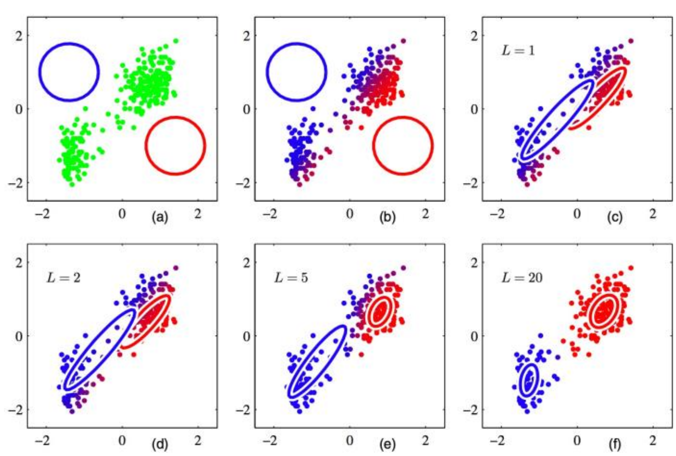
$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{j=1}^{N} \gamma(z_j = k)\boldsymbol{x}_j \qquad \pi_k = \frac{N_k}{N}$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{j=1}^{N} \gamma(z_j = k)(\boldsymbol{x}_j - \boldsymbol{\mu}_k)(\boldsymbol{x}_j - \boldsymbol{\mu}_k)^T$$

25

SJSU SAN JOSÉ STATE UNIVERSITY

Example: GMM with EM



26

SJSU SAN JOSÉ STATE UNIVERSITY

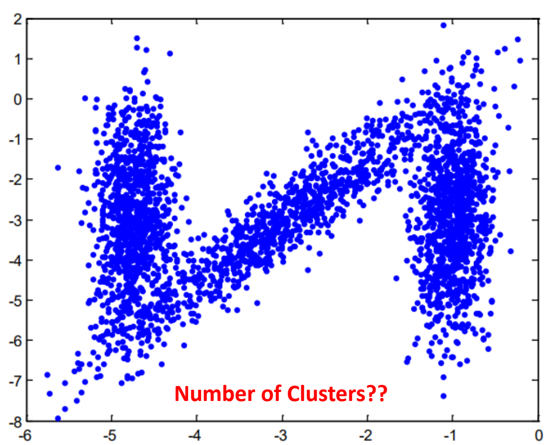
Summary of GMM Components

- Observations or Data: $\boldsymbol{x}_i \in \mathbb{R}^N$, $i = \{1, 2, \dots, N\}$
- Hidden Cluster Labels $z_i \in \{1, 2, \dots, N\}$, $i = \{1, 2, \dots, N\}$
- Hidden Mixture Means $\boldsymbol{\mu}_k \in \mathbb{R}^N$, $k = \{1, 2, \dots, \boldsymbol{K}\}$
- Hidden Mixture Covariances $\boldsymbol{\Sigma}_k \in \mathbb{R}^{N \times N}$, $k = \{1, 2, \dots, \boldsymbol{K}\}$
- Hidden Mixture Probabilities $\pi_k \in \mathbb{R}^N$, $\sum_{j=1}^{N} \pi_k = 1$

27

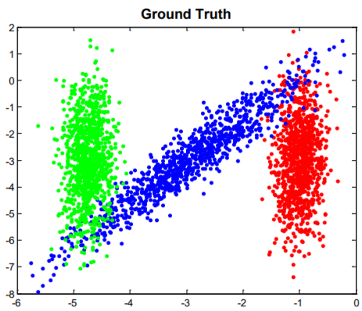
SJSU SAN JOSÉ STATE UNIVERSITY

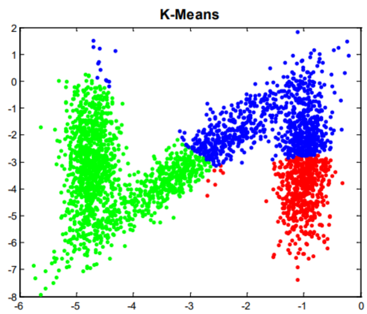
Example: GMM vs K-Means

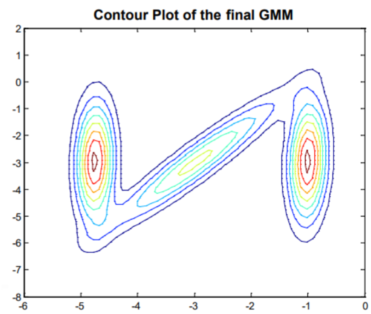


28

Example







[-1,-3]
[-3,-3]
[-4.75,-3]

K-Means with 50 random starting points

[-1.0335,-4.057]
[-1.5821, -1.6458]
[-4.3681, -3.4009]

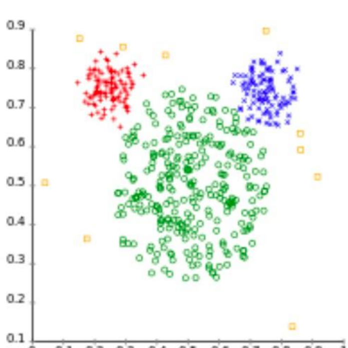
Soft Clustering using a 3 component Gaussian Mixture Model with random starting point

[-1.0006, -2.9663]
[-2.9747, -2.9921]
[-4.7488, -2.9717]

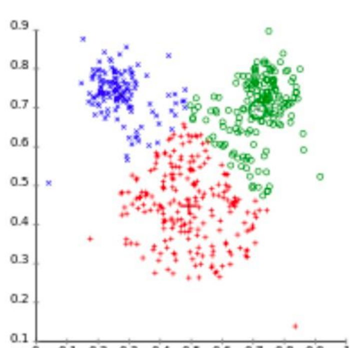
29

Example

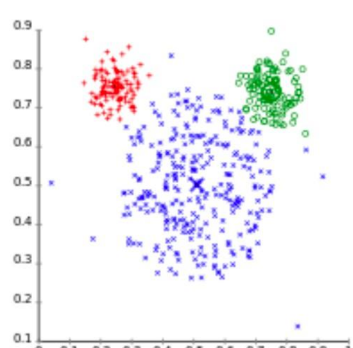
- Mouse Clusters…



Original Clusters



K-means



GMM

30

SJSU SAN JOSÉ STATE UNIVERSITY

Summary

- Mixture models are probabilistic models that represent data as a combination of multiple underlying distributions. Each underlying distribution models a specific cluster (or component) within the data.
- GMM is one type of mixture models by using Gaussian as the probability distribution.
- Model parameters for GMM include means, covariances, and mixing probability for each Gaussian component.
- GMMs are more flexible as it can model different mean/covariance for each cluster
- However, GMMs have (local) convergence issues, just like K-Menas
- GMMs are also slower than K-Means due to the higher computation effort needed by the EM algorithm. K-Means is sometimes used to initialize the cluster means.

31