UCLA Samueli Computer Science

CS145 Discussion: Week 6
Midterm Review

Junheng Hao

Friday, 11/13/2020

UCLA
Engineer Change.

Roadmap

- Announcement
- Clustering
- Homework Conceptual Questions
- Midterm Review

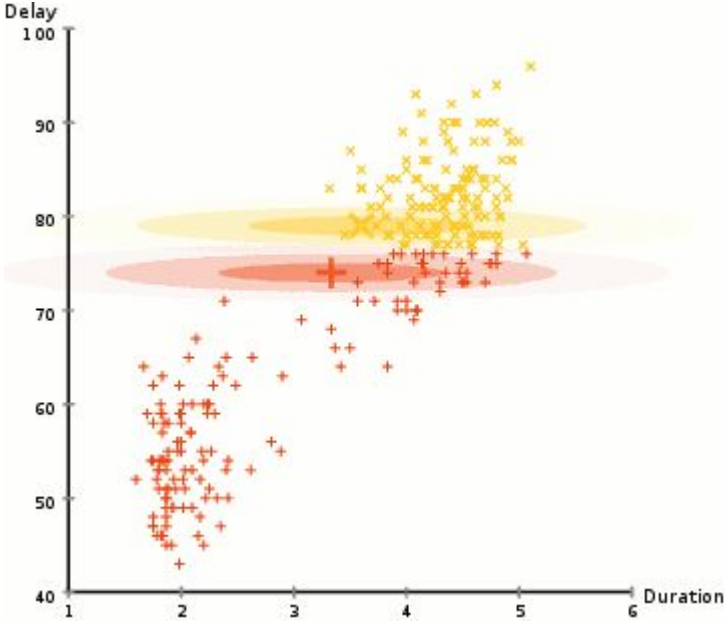
UCLA
Engineer Change.

Announcements

- Midterm exam on **Nov. 16 (Monday, Week 7)** on CCLE
  - Two time slots: 10AM-11:30AM and 6:00PM-7:30PM (PST)
  - Lockdown Browser and Respondus Required
- Homework 4 due **Nov. 20 (Friday, Week 7) 11:59 PT**
  - Submit through **GradeScope** of 1 PDF
  - Assign pages to the questions on GradeScope

Gaussian Mixture Model

- Generative model vs Discriminative model
- EM algorithm
- More details on GMM

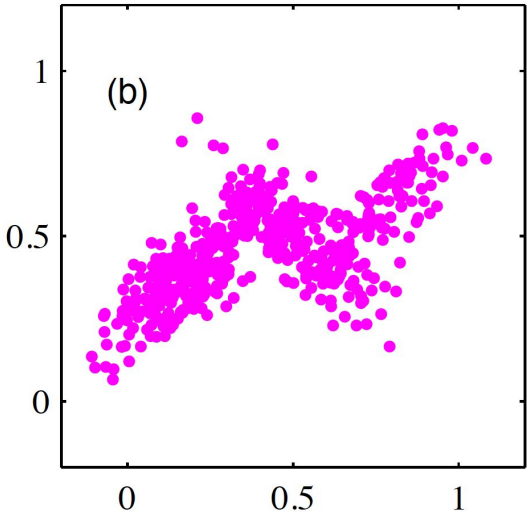


Gaussian Mixture Model

Probabilistic interpretation of clustering?

We can impose a probabilistic interpretation of our intuition that points stay close to their cluster centers

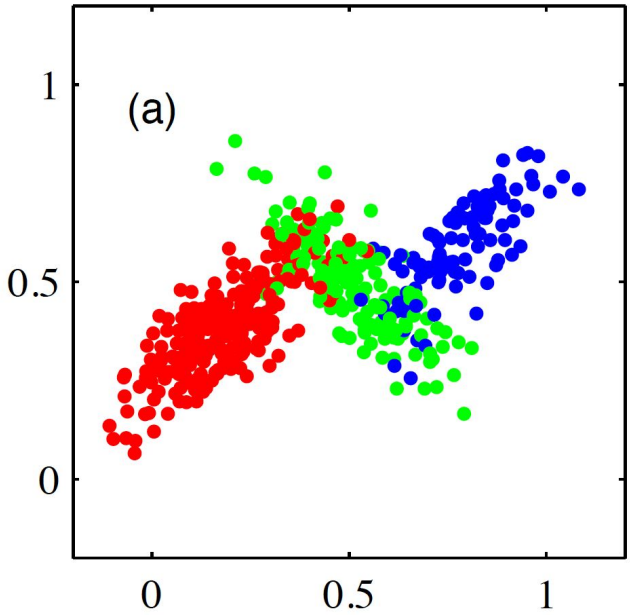
How can we model p(x) to reflect this?



Gaussian Mixture Model

Intuition

- We can model each region with a distinct distribution
- Common to use Gaussians, i.e.,
- Gaussian mixture models (GMMs) or mixture of Gaussians (MoGs).
- We don't know cluster assignments (label) or parameters of Gaussians or mixture components



Gaussian Mixture Models: Formulation

A Gaussian mixture model has the following density function for $\boldsymbol{x}$

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

- K: the number of Gaussians — they are called (mixture) components
- $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$: mean and covariance matrix of the k-th component
- ω_k: mixture weights – they represent how much each component contributes to the final distribution. It satisfies two properties:

$$\forall\ k,\ \omega_k > 0, \quad \text{and} \quad \sum_k \omega_k = 1$$

The properties ensure $p(\boldsymbol{x})$ is a properly normalized probability density function.

Gaussian Mixture Models: Formulation

Consider the following joint distribution

$$p(\boldsymbol{x}, z) = p(z)p(\boldsymbol{x}|z)$$

where z is a discrete random variable taking values between 1 and K. Denote

$$\omega_k = p(z = k)$$

Now, assume the conditional distributions are Gaussian distributions

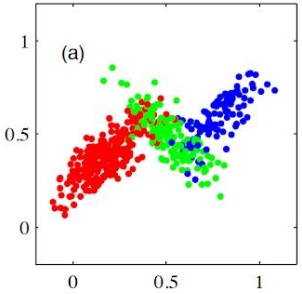
$$p(\boldsymbol{x}|z = k) = N(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Then, the marginal distribution of $\boldsymbol{x}$ is

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

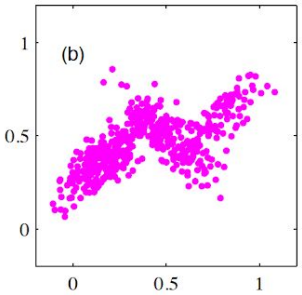
Namely, the Gaussian mixture model

Gaussian Mixture Models: Formulation



The conditional distribution between $\boldsymbol{x}$ and z (representing color) are

$$p(\boldsymbol{x}|z = red) = N(\boldsymbol{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$$
$$p(\boldsymbol{x}|z = blue) = N(\boldsymbol{x}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$$
$$p(\boldsymbol{x}|z = green) = N(\boldsymbol{x}|\boldsymbol{\mu}_3, \boldsymbol{\Sigma}_3)$$



The marginal distribution is thus

$$p(\boldsymbol{x}) = p(red)N(\boldsymbol{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) + p(blue)N(\boldsymbol{x}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) + p(green)N(\boldsymbol{x}|\boldsymbol{\mu}_3, \boldsymbol{\Sigma}_3)$$

Gaussian Mixture Models: Formulation

The parameters in GMMs are $\boldsymbol{\theta} = \{\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$. To estimate, consider the simple (and unrealistic) case first.

We have labels z If we assume z is observed for every $\boldsymbol{x}$, then our estimation problem is easier to solve. Our training data is augmented:

$$\mathcal{D}' = \{\boldsymbol{x}_n, z_n\}_{n=1}^N$$

z_n denotes the region where $\boldsymbol{x}_n$ comes from. $\mathcal{D}'$ is the *complete* data and $\mathcal{D}$ the *incomplete* data. How can we learn our parameters?

Given $\mathcal{D}'$, the maximum likelihood estimation of the $\boldsymbol{\theta}$ is given by

$$\boldsymbol{\theta} = \arg\max \log P(\mathcal{D}') = \sum_n \log p(\boldsymbol{x}_n, z_n)$$

Gaussian Mixture Models: Formulation

The *complete* likelihood is decomposable

$$\sum_n \log p(\boldsymbol{x}_n, z_n) = \sum_n \log p(z_n) p(\boldsymbol{x}_n | z_n) = \sum_k \sum_{n:z_n=k} \log p(z_n) p(\boldsymbol{x}_n | z_n)$$

where we have grouped data by its values z_n. Let us introduce a binary variable $\gamma_{nk} \in \{0, 1\}$ to indicate whether $z_n = k$. We then have

$$\sum_n \log p(\boldsymbol{x}_n, z_n) = \sum_k \sum_n \gamma_{nk} \log p(z = k) p(\boldsymbol{x}_n | z = k)$$

We use a "dummy" variable z to denote all the possible values cluster assignment values for $\boldsymbol{x}_n$

$\mathcal{D}'$ specifies this value in the complete data setting

Gaussian Mixture Models: Formulation

Parameter estimation for GMMs: complete data

From our previous discussion, we have

$$\sum_n \log p(\boldsymbol{x}_n, z_n) = \sum_k \sum_n \gamma_{nk} \left[\log \omega_k + \log N(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

Regrouping, we have

$$\sum_n \log p(\boldsymbol{x}_n, z_n) = \sum_k \sum_n \gamma_{nk} \log \omega_k + \sum_k \left\{ \sum_n \gamma_{nk} \log N(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

The term inside the braces depends on k-th component's parameters. It can be shown that the MLE is:

$$\omega_k = \frac{\sum_n \gamma_{nk}}{\sum_k \sum_n \gamma_{nk}}, \quad \boldsymbol{\mu}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} \boldsymbol{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$$

Gaussian Mixture Models: Formulation

When z_n is not given, we can guess it via the posterior probability

$$p(z_n = k|\boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n|z_n = k)p(z_n = k)}{p(\boldsymbol{x}_n)} = \frac{p(\boldsymbol{x}_n|z_n = k)p(z_n = k)}{\sum_{k'=1}^{K} p(\boldsymbol{x}_n|z_n = k')p(z_n = k')}$$

To compute the posterior probability, we need to know the parameters $\boldsymbol{\theta}$!

Let's pretend we know the value of the parameters so we can compute the posterior probability.

How is that going to help us?

Gaussian Mixture Models: Formulation

Estimation with soft γ_{nk}

We define $\gamma_{nk} = p(z_n = k|\boldsymbol{x}_n)$

- Recall that γ_{nk} should be binary
- Now it's a "soft" assignment of $\boldsymbol{x}_n$ to k-th component
- Each $\boldsymbol{x}_n$ is assigned to a component fractionally according to $p(z_n = k|\boldsymbol{x}_n)$

We now get the same expression for the MLE as before!

$$\omega_k = \frac{\sum_n \gamma_{nk}}{\sum_k \sum_n \gamma_{nk}}, \quad \boldsymbol{\mu}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} \boldsymbol{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$$

But remember, we're 'cheating' by using $\boldsymbol{\theta}$ to compute γ_{nk}!

Gaussian Mixture Models: Formulation

We can alternate between estimating γ_{nk} and using the estimated γ_{nk} to compute the parameters (same idea as with K-means!)

- Step 0: initialize $\boldsymbol{\theta}$ with some values (random or otherwise)
- Step 1: compute γ_{nk} using the current $\boldsymbol{\theta}$
- Step 2: update $\boldsymbol{\theta}$ using the just computed γ_{nk}
- Step 3: go back to Step 1

Questions:

- Is this procedure reasonable, i.e., are we optimizing a sensible criteria?
- Will this procedure converge?

The answers lie in the *EM algorithm* — a powerful procedure for model estimation with unknown data.

Gaussian Mixture Models: Formulation

No simple way to optimize the incomplete log-likelihood

Expectation-Maximization (EM) algorithm provides a strategy for iteratively optimizing this function

Two steps as they apply to GMM:

- E-step: 'guess' values of the z_n using existing values of $\boldsymbol{\theta}$
- M-step: solve for new values of $\boldsymbol{\theta}$ given imputed values for z_n (maximize complete likelihood!)

Gaussian Mixture Models: Formulation

E-step: Soft cluster assignments

We define γ_{nk} as $p(z_n = k|\boldsymbol{x}_n, \boldsymbol{\theta})$

- This is the posterior distribution of z_n given $\boldsymbol{x}_n$ and $\boldsymbol{\theta}$
- Recall that in complete data setting γ_{nk} was binary
- Now it's a "soft" assignment of $\boldsymbol{x}_n$ to k-th component, with $\boldsymbol{x}_n$ assigned to each component with some probability

Given $\boldsymbol{\theta} = \{\omega_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, we can compute γ_{nk} using Bayes theorem:

$$\begin{aligned}
\gamma_{nk} &= p(z_n = k|\boldsymbol{x}_n) \\
&= \frac{p(\boldsymbol{x}_n|z_n = k)p(z_n = k)}{p(\boldsymbol{x}_n)} \\
&= \frac{p(\boldsymbol{x}_n|z_n = k)p(z_n = k)}{\sum_{k'=1}^K p(\boldsymbol{x}_n|z_n = k')p(z_n = k')}
\end{aligned}$$

Gaussian Mixture Models: Formulation

M-step: Maximimize complete likelihood

Recall definition of complete likelihood from earlier:

$$\sum_n \log p(\boldsymbol{x}_n, z_n) = \sum_k \sum_n \gamma_{nk} \log \omega_k + \sum_k \left\{ \sum_n \gamma_{nk} \log \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

Previously γ_{nk} was binary, but now we define $\gamma_{nk} = p(z_n = k | \boldsymbol{x}_n)$ (E-step)

We get the same simple expression for the MLE as before!

$$\omega_k = \frac{\sum_n \gamma_{nk}}{\sum_k \sum_n \gamma_{nk}}, \quad \boldsymbol{\mu}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} \boldsymbol{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{\sum_n \gamma_{nk}} \sum_n \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$$

Intuition: Each point now contributes some fractional component to each of the parameters, with weights determined by γ_{nk}

Gaussian Mixture Models: Formulation

Questions to be answered next

- How does GMM relate to K-means?
- Is this procedure reasonable, i.e., are we optimizing a sensible criterion?
- Will this procedure converge?

UCLA
Engineer Change.

GMM: Examples

- Example 1: Manual calculation of the M step of GMM

  Consider clustering ID data with a mixture of 2 gaussian. You're given the 1-D data points x = [1 10 20]. Suppose the E step is the following matrix :

$$\begin{bmatrix} 1 & 0 \\ 0.4 & 0.6 \\ 0 & 1 \end{bmatrix}$$

a. What's the mixing weights after M-step?

b. What's the new values of means after M-step?

UCLA
Engineer Change.

GMM: Calculation Helper

Expectation (E) Step:

Calculate $\forall i, k$

$$\hat{\gamma}_{ik} = \frac{\hat{\phi}_k \mathcal{N}(x_i \mid \hat{\mu}_k, \hat{\sigma}_k)}{\sum_{j=1}^{K} \hat{\phi}_j \mathcal{N}(x_i \mid \hat{\mu}_j, \hat{\sigma}_j)},$$

where $\hat{\gamma}_{ik}$ is the probability that x_i is generated by component C_k. Thus, $\hat{\gamma}_{ik} = p(C_k|x_i, \hat{\phi}, \hat{\mu}, \hat{\sigma})$.

Maximization (M) Step:

Using the $\hat{\gamma}_{ik}$ calculated in the expectation step, calculate the following in that order $\forall k$:

- $\hat{\phi}_k = \sum_{i=1}^{N} \frac{\hat{\gamma}_{ik}}{N}$
- $\hat{\mu}_k = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} x_i}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$
- $\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} (x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}.$

Example: Clustering Evaluation

[HW4] Calculate:

Purity, precision, recall, F-measure, and normalized mutual information

ID	Conference Name	Ground Truth Label	Algorithm output Label
1	IJCAI	3	2
2	AAAI	3	2
3	ICDE	1	3
4	VLDB	1	3
5	SIGMOD	1	3
6	SIGIR	4	4
7	ICML	3	2
8	NIPS	3	2
9	CIKM	4	3
10	KDD	2	1
11	WWW	4	4
12	PAKDD	2	1
13	PODS	1	3
14	ICDM	2	1
15	ECML	3	2
16	PKDD	2	1
17	EDBT	1	2
18	SDM	2	1
19	ECIR	4	4
20	WSDM	4	4

UCLA
Engineer Change.

Midterm Topics

Task	Vector data
Classification	Logistic Regression; Decision Tree; KNN; SVM; Neural Networks
Clustering	K-means; Hierarchical clustering; DBSCAN; Mixture Models
Prediction	Linear Regression

UCLA
Engineer Change.

Midterm Topics: XOR Example

Task	Vector data
Classification	Logistic Regression; Decision Tree; KNN; SVM; Neural Networks
Clustering	K-means; Hierarchical clustering; DBSCAN; Mixture Models
Prediction	Linear Regression

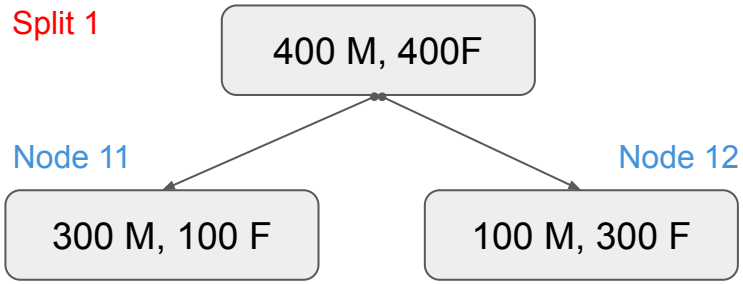
UCLA
Engineer Change.

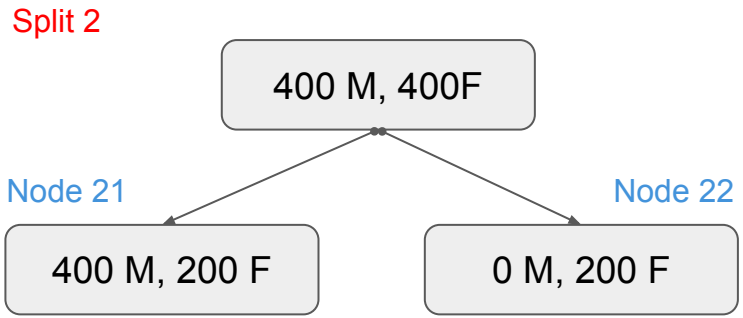
Homework 1

- Closed-form, batch gradient descent, stochastic gradient descent
- Regularization terms in linear regression and logistic regression

Homework 2: Decision Tree

- Calculate information gain and split info





UCLA
Engineer Change.

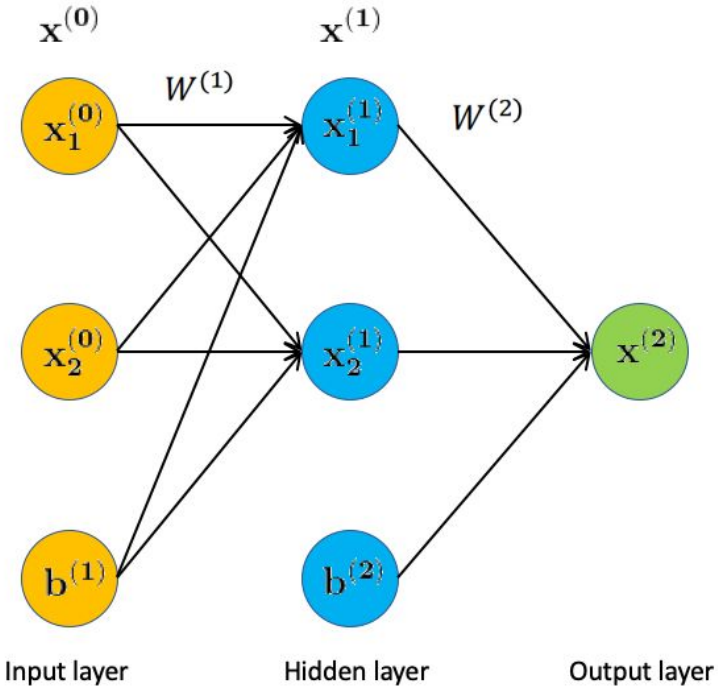
Homework 2: SVM

- Calculate SVM decision boundary and predict on new data

UCLA
Engineer Change.

Homework 3

- Forward pass and backward pass of a toy neural net



UCLA
Engineer Change.

Homework 3: Back-prop Calculation

This slides is left blank intentionally for discussion demonstration.

UCLA Samueli Computer Science

Thank you!

Q & A