Lec 11

Thursday, October 3, 2019 10:58

Recap: Clustering

First clustering algo: k-means

Soft clustering / Gaussian Mixture Models

GMM: $Y \sim \text{Bernoulli}(\pi)$

(p=2 clusters, p=1 univariate X) $(X|Y=y) \sim N(\mu_y, \sigma_y^2)$

This is a generative model for the data w/ parameters $\theta = (\pi, \mu_0, \sigma_0^2, \mu_1, \sigma_1^2)$

$$\mathbb{P}(Y=1|X=x) = \frac{\pi \varphi\left(\frac{x-\mu_1}{\sigma_1}\right)}{\pi \varphi\left(\frac{x-\mu_1}{\sigma_1}\right) + (1-\pi)\varphi\left(\frac{x-\mu_0}{\sigma_0}\right)}$$

= cluster responsibility

Estimate θ by MLE using EM algo

Given observations $X_1, \dots, X_n$, the log-lik is

$$\ell(\theta; \mathbf{X}) = \sum_i \log\left((1-\pi)\varphi\left(\frac{x_i-\mu_0}{\sigma_0}\right) + \pi\varphi\left(\frac{x_i-\mu_1}{\sigma_1}\right)\right)$$

actually hard to optimize

Instead:

Let's pretend that we also observe

$y_1, \dots, y_n$ (the true cluster memberships)

$$\ell(\theta; X, Y) = \sum_{i=1}^n \left((1-y_i) \log \varphi\left(\frac{x_i-\mu_0}{\sigma_0}\right) + y_i \log \varphi\left(\frac{x_i-\mu_1}{\sigma_1}\right) \right) \quad (*)$$

$$+ \sum_{i=1}^n \left((1-y_i) \log(1-\pi) + y_i \log \pi \right)$$

max of this over π gives $\hat{\pi} = \frac{1}{n}\sum_i y_i$

only involves $\mu_1, \sigma_1, \mu_0, \sigma_0$

only involves π

This is an easy convex opt problem

$$(*) = \sum_{i: y_i=1} \log \varphi\left(\frac{x_i-\mu_1}{\sigma_1}\right) + \sum_{i: y_i=0} \log \varphi\left(\frac{x_i-\mu_0}{\sigma_0}\right)$$

max of this over μ_1, σ_1

Same but for the $y_i=0$ group

$$\hat{\mu}_1 = \frac{\sum_{i: y_i=1} x_i}{\sum_{i: y_i=1} 1}$$

$$\hat{\sigma}_1^2 = \frac{\sum_{i: y_i=1} (x_i - \hat{\mu}_1)^2}{\sum_{i: y_i=1} 1}$$

E-M trick: replace y_i w/ currently estimated cluster responsibilities

Expectation-Maximization (EM) alg:

1. Take an initial guess $\hat{\theta} = (\hat{\pi}, \hat{\mu}_0, \hat{\mu}_1, \hat{\sigma}_1, \hat{\sigma}_0)$

2. E-step:

Compute the cluster responsibilities implied by $\hat{\theta}$:

$$\hat{\gamma}_i = \frac{\hat{\pi}\, \varphi\left(\frac{x_i-\hat{\mu}_1}{\hat{\sigma}_1}\right)}{\hat{\pi}\, \varphi\left(\frac{x_i-\hat{\mu}_1}{\hat{\sigma}_1}\right) + (1-\hat{\pi})\, \varphi\left(\frac{x_i-\hat{\mu}_0}{\hat{\sigma}_0}\right)}$$

Impute $Y_i \leftarrow \hat{\gamma}_i \in [0,1]$

(impute unknown Y_i by its conditional expectation)

3. M-step

Maximize the likelihood of $\hat{\theta}$ given a complete set of observations

X (real obs)

Y (imputed obs)

$$\hat{\mu}_1 = \frac{\sum_{i=1}^n \hat{\gamma}_i x_i}{\sum_{i=1}^n \hat{\gamma}_i} \qquad \hat{\mu}_0 = \frac{\sum_i (1-\hat{\gamma}_i) x_i}{\sum_i (1-\hat{\gamma}_i)}$$

$$\hat{\sigma}_1^2 = \frac{\sum_i \hat{\gamma}_i (x_i - \hat{\mu}_1)^2}{\sum_i \hat{\gamma}_i} \qquad \hat{\sigma}_0^2 = \frac{\sum_i (1-\hat{\gamma}_i)(x_i - \hat{\mu}_0)}{\sum_i (1-\hat{\gamma}_i)}$$

$$\hat{\pi} = \frac{1}{n} \sum_i \hat{\gamma}_i$$

4. Repeat from step 2

Unsupervised learning w/ abstract distances

So far in unsupervised learning:

$X \in \mathbb{R}^p \rightsquigarrow Z$ lower dim

either: $G \in \mathbb{R}^2$ PCA

$G \in \{1, \dots, K\}$ cluster

but we always started w/ vector data $X \in \mathbb{R}^p$

Next: use more abstract descriptions of the data.

(Dis) Similarity Measures

1. Euclidean dist

$x \in \mathbb{R}^p \quad x' \in \mathbb{R}^p$

$\|x - x'\|_2^2 = \sum_i (x_i - x'_i)^2$

2. Chi-squared dist

$$\frac{1}{2} \sum_{k=1}^{p} \frac{(x_k - x'_k)^2}{x_k + x'_k}$$

nonnegative vectors representing a histogram distribution or a list of frequencies

3. Cosine-similarity

measure the alignment b/w two vectors

Let θ be the angle b/w two vectors, x & x'.

$$\text{CosineSim} = \cos(\theta) = \frac{x^T x'}{\|x\|_2 \|x'\|_2}$$

$\cos\theta = 1 \Rightarrow$ perfectly aligned

$= -1 \Rightarrow$ perfectly opposite

Cos dissim $= 1 - \cos\theta$

often used in conjunction w/ BoW.

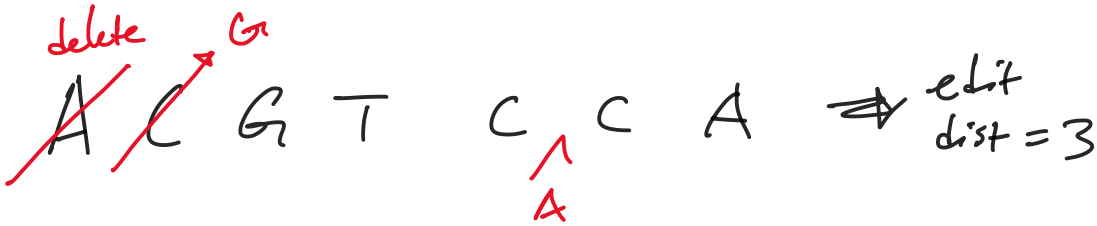
4. Edit distance

Given two strings

$X = A C G T C C A$

$X' = G G T C A C A$

edit dist = how many insertions, deletions, & replacements minimally needed to go from X to X'



Multidimensional Scaling (MDS)

MDS: Given distance/dissimilarity matrix

$D \in \mathbb{R}^{n \times n}$

Q: What vectors $X \in \mathbb{R}^{n \times p}$ recover D in the closest way via $D_{ij} \approx \|X_i - X_j\|_2^2$

I.e. X is embedding of the n abstract datapts in p real dimensions that maintains dissimilarities as vector distances.

If indeed $D_{ij} = \|X_i - X_j\|_2^2$ then

$$D_{ij} = \|x_i\|_2^2 + \|x_j\|_2^2 - 2 x_i^T x_j$$

$$= T_{ii} + T_{jj} - 2 T_{ij}$$

Where $T = X X^T \in \mathbb{R}^{n \times n}$

(nxn matrix whose $(i,j)^{th}$ entry contains inner prod of x_i & x_j)

Solve for T, then eigendecompose it to get X

MDS: Eigendecompose T to get X s.t. $T = X X^T$

Specifically: $T = U \Lambda U^T$ ✓ a soln up to rotation & translation

set $X = U_p \Lambda_p^{1/2}$

Fact: if D really came from Euclidean distances, then MDS is just PCA