CS412 Office Hours

April 1, 2019

Administration Issues

- If you need an extension due to illness/travel etc. please reach out to the course staff via mcs-support@illinois.edu.
    - You will be taking the alternative exam
- We will release exam statistics and cover exam questions in next week's office hour
    - Tentative schedule is next Wednesday, please look out for notices
- Another extra credit assignment released!
- Extra credit assignment results are available on Compass. (Not Coursera as it does not support >100% points)
- Programming assignment deadlines for this module **cannot be extended**

Kernel Functions and Kernel K-means

- Motivation for using kernel k-means: we want to cluster points that are not linearly separable in the input space (by means of Euclidean distance)
- Idea: **project the points to another space** (possibly high dimensional) by a function $\Phi(x)$. Then apply k-means to the projected data.

Our objective:

$$\mathcal{D}(\{\pi_j\}_{j=1}^k) = \sum_{j=1}^{k} \sum_{\mathbf{a} \in \pi_j} w(\mathbf{a}) \|\phi(\mathbf{a}) - \mathbf{m}_j\|^2$$

$$\text{where} \quad \mathbf{m}_j = \frac{\sum_{\mathbf{b} \in \pi_j} w(\mathbf{b}) \phi(\mathbf{b})}{\sum_{\mathbf{b} \in \pi_j} w(\mathbf{b})}.$$

This is the centroid of cluster j.

Note that this formula is for weighted k-means, that is every point is associated with a weight w(a). You can think of them as all equal to 1, then you get the basic k-means.

Kernel Functions and Kernel K-means (continued)

Quick review of k-means: In every iteration, we update the assignment π for all the clusters by assigning each point to the cluster with the closest centroid.

⇒ Need to compute the distance from each point to the centroid.

The Euclidean distance:

$$\left\| \phi(\mathbf{a}) - \frac{\sum_{\mathbf{b}\in\pi_j} w(\mathbf{b})\phi(\mathbf{b})}{\sum_{\mathbf{b}\in\pi_j} w(\mathbf{b})} \right\|^2 = \boxed{\phi(\mathbf{a})\cdot\phi(\mathbf{a})} - \frac{2\sum_{\mathbf{b}\in\pi_j} w(\mathbf{b})\boxed{\phi(\mathbf{a})\cdot\phi(\mathbf{b})}}{\sum_{\mathbf{b}\in\pi_j} w(\mathbf{b})} + \frac{\sum_{\mathbf{b},\mathbf{c}\in\pi_j} w(\mathbf{b})w(\mathbf{c})\boxed{\phi(\mathbf{b})\cdot\phi(\mathbf{c})}}{(\sum_{\mathbf{b}\in\pi_j} w(\mathbf{b}))^2}.$$

Observation: All computations in this formula are in the form of a product of two projected points.

Kernel Functions and Kernel K-means (continued)

Observation: All computations in this formula are in the form of a product of two projected points.

We can define $K(x_1, x_2) = \Phi(x_1)\ \Phi(x_2)$. (We call this a kernel function).

Instead of computing the projection function and then taking the product, we can directly compute the value of the kernel function.

The advantage of using kernels: sometimes it's easier to compute the kernel function than the projection; kernels can express infinite dimension projections

Kernel Functions and Kernel K-means (continued)

Quadratic Kernel

- $\boldsymbol{\phi}\colon \mathbb{R}^d \to \mathbb{R}^{1+2d+\binom{d}{2}}$, where

$$\boldsymbol{\phi}(\boldsymbol{x}) = \Big(1,\ \sqrt{2}x_1,\ \ldots,\ \sqrt{2}x_d,\ x_1^2,\ \ldots,\ x_d^2,\ \sqrt{2}x_1x_2,\ \ldots,\ \sqrt{2}x_1x_d,\ \ldots,\ \sqrt{2}x_{d-1}x_d\Big)$$

  (Don't mind the $\sqrt{2}$'s. . .)

- **Computing $\boldsymbol{\phi}(\boldsymbol{x})^\top\boldsymbol{\phi}(\boldsymbol{x}')$ in $O(d)$ time**:

$$\boldsymbol{\phi}(\boldsymbol{x})^\top\boldsymbol{\phi}(\boldsymbol{x}') = (1 + \boldsymbol{x}^\top\boldsymbol{x}')^2.$$

- Much better than d^2 time.

Kernel Functions and Kernel K-means (continued)

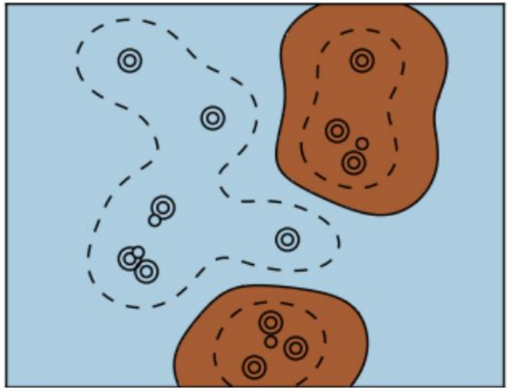
Gaussian Kernel

We will see more of Kernel functions in the Support Vector Machine lession.

For any $\sigma > 0$, there is an infinite feature expansion $\phi\colon \mathbb{R}^d \to \mathbb{R}^\infty$ such that

$$\phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2^2}{2\sigma^2}\right),$$

which can be computed in $O(d)$ time.



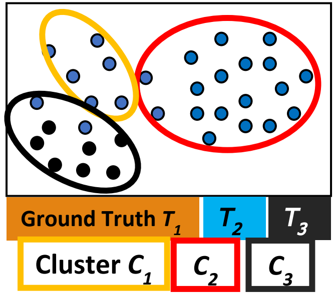
(This is called the *Gaussian kernel* with bandwidth σ.)

Clustering Pairwise Measures

- **Four possibilities** based on the agreement between cluster label and partition label
  - *TP*: true positive—Two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ belong to the same partition *T*, and they also in the same cluster *C*

$$TP = |\{(\mathbf{x}_i, \mathbf{x}_j) : y_i = y_j \text{ and } \hat{y}_i = \hat{y}_j\}|$$

  where y_i: the true partition label , and $\hat{y}_i$: the cluster label for point $\boldsymbol{x}_i$



  - *FN*: false negative: $FN = |\{(\mathbf{x}_i, \mathbf{x}_j) : y_i = y_j \text{ and } \hat{y}_i \neq \hat{y}_j\}|$
  - *FP: false positive* $FP = |\{(\mathbf{x}_i, \mathbf{x}_j) : y_i \neq y_j \text{ and } \hat{y}_i = \hat{y}_j\}|$
  - *TN*: true negative $TN = |\{(\mathbf{x}_i, \mathbf{x}_j) : y_i \neq y_j \text{ and } \hat{y}_i \neq \hat{y}_j\}|$
- Calculate the four measures:

$$N = \binom{n}{2}$$

Total # of pairs of points

$$TP = \sum_{i=1}^{r}\sum_{j=1}^{k}\binom{n_{ij}}{2} = \frac{1}{2}\left(\left(\sum_{i=1}^{r}\sum_{j=1}^{k} n_{ij}^{2}\right) - n\right) \quad FN = \sum_{j=1}^{k}\binom{m_j}{2} - TP$$

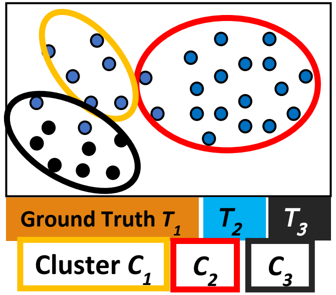
$$FP = \sum_{i=1}^{r}\binom{n_i}{2} - TP \quad TN = N - (TP + FN + FP) = \frac{1}{2}\left(n^2 - \sum_{i=1}^{r} n_i^{2} - \sum_{j=1}^{k} m_j^{2} + \sum_{i=1}^{r}\sum_{j=1}^{k} n_{ij}^{2}\right)$$

Pairwise Measures: Jaccard Coefficient and Rand Statistic

- **Jaccard coefficient:** Fraction of true positive point pairs, but after ignoring the true negatives (thus asymmetric)
  - $Jaccard = TP/(TP + FN + FP)$ [i.e., denominator ignores TN]
  - Perfect clustering: $Jaccard = 1$
- **Rand Statistic**:
  - $Rand = (TP + TN)/N$
  - Symmetric; perfect clustering: $Rand = 1$
- **Fowlkes-Mallow Measure**:
  - Geometric mean of precision and recall

$$FM = \sqrt{prec \times recall} = \frac{TP}{\sqrt{(TP + FN)(TP + FP)}}$$

- Using the above formulas, one can calculate all the measures for the green table (leave as an exercise)

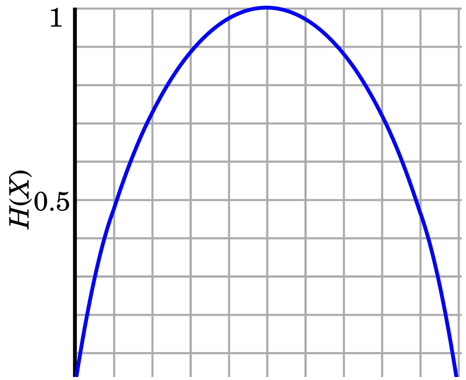


$C \backslash T$	T_1	T_2	T_3	Sum
C_1	0	20	30	50
C_2	0	20	5	25
C_3	25	0	0	25
m_j	25	40	35	100

From Entropy to Info Gain: A Brief Review of Entropy

- Entropy (Information Theory)
  - A measure of uncertainty associated with a random number
  - Calculation: For a discrete random variable Y taking m distinct values $\{y_1, y_2, \ldots, y_m\}$

$$H(Y) = -\sum_{i=1}^{m} p_i \log(p_i) \quad where\ p_i = P(Y = y_i)$$



  - Interpretation
    - Higher entropy → higher uncertainty
    - Lower entropy → lower uncertainty
- Conditional entropy

$$H(Y|X) = \sum_{x} p(x) H(Y|X = x)$$

$$\mathrm{H}(X|Y) = -\sum_{i,j} p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(y_j)}$$

Information Gain: An Attribute Selection Measure

- Select the attribute with the highest information gain (used in typical decision tree induction algorithm: ID3)

Suppose we have A, B, … N attributes and one response variable Y which are all random variables

- Let p_i be the probability that an arbitrary tuple in D belongs to class C_i, estimated by $|C_{i, D}|/|D|$
- Expected information (entropy) needed to classify a tuple in D:

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

At the root of the decision tree, expected info is H(Y).

- Information needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

After we know the value of A, expected info is H(Y|A).

- Information gained by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

Example: Attribute Selection with Information Gain

- Class P: buys_computer = "yes"
- Class N: buys_computer = "no"

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

$\frac{5}{14}I(2,3)$ means "age <=30" has 5 out of 14 samples, with 2 yes'es and 3 no's.

Hence

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

Similarly, we can get

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

Course Structure

- Lesson 0: Classification in Data Mining: An Introduction
- Lesson 1: Decision Tree Induction
- Lesson 2: Bayes Classifier and Bayesian Networks
- Lesson 3: Model Evaluation, Selection and Improvements
- Lesson 4: Linear Classifier and Support Vector Machines
- Lesson 5: Neural Networks and Deep Learning
- Lesson 6: Pattern-Based Classification and K-Nearest Neighbors Algorithm