Makine Ögrenmesi

2. hafta

- Uzaklığa dayalı gruplandırma
  - K-means kümeleme
  - K-NN sınıflayıcı

Uzaklığa dayalı gruplandırma

Makine öğrenmesinde amaç birbirine en çok benzeyen veri noktalarını aynı grup içerisinde birarada tutmaktır. Benzerlik kavramı çoğu kez uzaklığın tersiyle ifade edilir. Farklı hesaplara dayanan birçok sınıflandırma veya kümeleme algoritmalarından bazıları uzaklık ölçütü ile gruplandırma yapar.

Benzerlik

3

Benzerlik vs. Uzaklık

Uzaklık için en çok kullanılan ölçüt Öklit (Euclidean distance) uzaklığıdır. İki nokta (*n* boyutlu) arasındaki uzaklık şöyle hesaplanır:

$$d_{ab} = \|a-b\| = \sqrt{(a_1-b_1)^2+(a_2-b_2)^2+\ldots+(a_n-b_n)^2}$$

Benzerlik ise genellikle uzaklığın tersidir.

$$s_{ab} = \frac{1}{1+\|a-b\|}$$

4

Benzerlik vs. Uzaklık

Fakat uzaklık hesabı için Öklit dışında önerilmiş Mahalanobis, Manhattan ve Chebyshev gibi farklı birçok ölçüt bulunmaktadır. Benzerlik ölçütü de her zaman uzaklık ile ilişkili olmak zorunda değildir. Literatürde uzaklığa dayalı, olasılıksal ve özellik tabanlı gibi farklı temellere dayanan birçok benzerlik ölçütü önerilmiştir.

5

Uzaklığa dayalı kümeleme

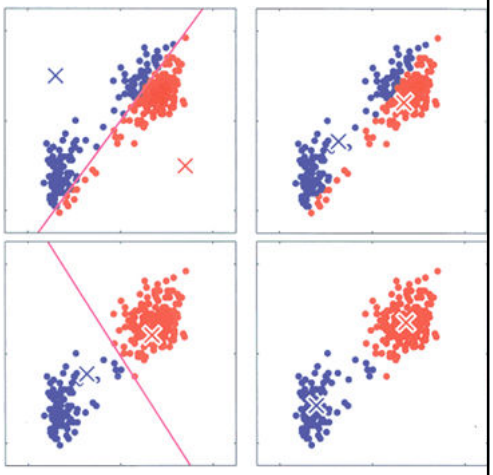
Kümeleme, sınıf bilgisi içermeyen veri içerisinde benzerliği yüksek noktaları gruplama işlemidir. Bu tanımdaki "benzerliği yüksek" terimini "benzersizliği düşük" şeklinde değiştirebiliriz. Böylece benzersizlik için uzaklık ölçütü kullanılabilir.

Uzaklığı kullanarak kümeleme yapan en popüler ve çoğunlukla en başarılı yöntem K-means algoritmasıdır.

6

K Ortalamalar (K-means)

Veri içerisinden rasgele seçilen K adet noktaya küme merkezi gözüyle bakılır. Tüm veri noktaları bu küme merkezlerine uzaklıklarına göre gruplanır. Her gruplamadan sonra küme merkezleri tekrar hesaplanır.



7

K Ortalamalar (K-means)

Her gruplama ve küme merkezinin tekrar hesabı işlemi bir iterasyon içerisinde gerçekleştirilir. Algoritmanın kaç iterasyon yapacağı çoğu kez *J* maliyet fonksiyonu ile tespit edilir.

$$J = \sum_{i=1}^{K} \left(\sum_{k} \| x_k - c_i \|^2 \right)$$

Maliyet fonksiyonu tüm küme merkezlerinin en uygun konumda olduğu durumda minimum değerde olmalıdır.

8

K-means Algoritması

1. Rasgele *K* adet küme merkezi belirle.
2. Her veri noktasını en yakın küme merkezine göre bir gruba yerleştir.
3. Her gruba atanan tüm veri noktalarının ortalamasını alarak küme merkezini güncelle.
4. Grup üyelerinde değişim olmayana kadar 2. adıma git. Bazen değişim, sıfıra yakın bir değerde uzun zaman sürebilir. O zaman maliyet fonksiyonu belirli bir değere düşünce veya toplam iterasyon sayısı belirli bir değere ulaşınca algoritma durdurulur.

9

K-means Dezavantajları

- Rasgele noktalar ile başladığından elde edilen her sonuç kararlı değildir.
- Küme sayısı birçok kümeleme algoritmasında olduğu gibi dışarıdan istenilir.
- Konkav (konvex olmayan) küme şekillerinde başarılı değildir.
- Keskin kümeleme yapar. Ama uygulamada gruplar genelde içiçedir (overlapping).
- Aykırı değerlere (outlier) karşı hassastır.

10

MATLAB Uygulaması

```
X = [rand(20,2)+0.3*ones(20,2); rand(20,2)-0.3*ones(20,2)];
opts = statset('maxiter',1);
C = X(1:2,:);
plot(X(:,1), X(:,2), 'k.', 'MarkerSize', 12);
hold on;
plot(C(:,1), C(:,2), 'kx', 'MarkerSize', 12, 'LineWidth', 2);
axis([-1.1 2.1 -1.1 2.1]);

for i=1:5
   pause;
   hold off;
   [idx,C] = kmeans(X,2,'Options',opts,'start',C);
   plot(X(idx==1,1), X(idx==1,2), 'r.', 'MarkerSize', 12);
   hold on;
   plot(X(idx==2,1), X(idx==2,2), 'b.', 'MarkerSize', 12);
   plot(C(:,1), C(:,2), 'kx', 'MarkerSize', 12, 'LineWidth', 2);
   axis([-1.1 2.1 -1.1 2.1]);
   legend('Cluster 1', 'Cluster 2', 'Centroids', 'Location', 'NW');
end
```

11

ÖDEVLER

- Agglomerative
- Divisive
- BIRCH
- STING
- Chameleon
- OPTICS
- CURE
- DBSCAN
- CLARANS

12

Uzaklığa dayalı sınıflandırma

Sınıflandırma, veri içerisindeki noktaları hem özellik uzayındaki hem de sınıf bilgisindeki benzerliklere göre gruplama işlemidir. Daha önce bahsedildiği gibi yüksek benzerlik terimi düşük uzaklık olarak kullanılmaktadır. Veri noktaları arasındaki uzaklıkları kullanarak sınıflandırma yapan en popüler yöntem K en yakın komşu (K-NN) yöntemidir.

13

K en yakın komşu (K-NN)

Bir *a* noktasının sınıfını belirlemek için veride bulunan tüm x_i noktaları kullanılır. K değeri, bilinmeyen noktaya en yakın kaç adet sınıflı noktanın kullanılacağını belirler. Örneğin 1-NN için hedef sınıf, bilinmeyen noktaya en yakın sınıflı veri noktası tarafından belirlenir. 3-NN için hedef sınıf ise bilinmeyen noktaya en yakın 3 adet sınıflı noktadan çoğunluğu sağlayan sınıf olmalıdır.

14

K en yakın komşu (K-NN)

Farklı K değerleri için

15

K-NN Algoritması

- K değeri dışarıdan istenir.
- Sınıfı bilinmeyen noktaya sınıflı veri noktalarının uzaklıkları hesaplanır. Genelde Öklit uzaklığı kullanılır.
- Hesaplanan uzaklıklar sıralanır. En küçük uzaklığa sahip (en yakın) K adet sınıflı veri noktası tespit edilir.
- Seçilen K adet sınıflı veri noktası içerisinde çoğunluğa sahip sınıf belirlenir. Bu sınıf, bilinmeyen nokta için kestirim sonucudur.

16

K-NN Dezavatajları

- Bellek tabanlı bir sınıflayıcıdır. Sınıflı veri noktaları sürekli hafızada tutulmalıdır. Veri kümesi çok büyük olduğunda hesaplama süresi de kötüleşir.
- Hesaplamaya tüm özellikler katıldığı için gereksiz veya ilgisiz özellikler sınıflandırmayı kötü yönde etkileyebilir.
- Sınıflandırma başarısı açısından genellikle yapay sinir ağları gibi gelişmiş sınıflandırma tekniklerinin gerisinde kalır.

17

MATLAB Uygulaması

Rasgele üretilen *X*, *D* eğitim matrisleri yardımıyla rasgele oluşturulan *A* matrisindeki örneklerin sınıflarını (*Y*) bulmak istiyoruz.

```
X=rand(10,2);
D=fix(2*rand(10,1));
A=rand(5,2);
```

18

MATLAB Uygulaması

%%K=1 için uygulama sonucu

```
Y=knnclassify(A,X,D,1);
figure; plot(X(D==1,1),X(D==1,2),'r.');
hold on; plot(X(D==0,1),X(D==0,2),'b.');
plot(A(Y==1,1),A(Y==1,2),'ro');
plot(A(Y==0,1),A(Y==0,2),'bo');
axis([-0.1 1.1, -0.1 1.1]);
```

19

MATLAB Uygulaması

%%K=3 için uygulama sonucu

```
Y=knnclassify(A,X,D,3);
figure; plot(X(D==1,1),X(D==1,2),'r.');
hold on; plot(X(D==0,1),X(D==0,2),'b.');
plot(A(Y==1,1),A(Y==1,2),'ro');
plot(A(Y==0,1),A(Y==0,2),'bo');
axis([-0.1 1.1, -0.1 1.1]);
```

20

ÖDEV

Herhangi bir sınıflandırma örneğini K-NN ile sınıflandırmada aşağıdaki uzaklık ölçütlerini karşılaştırınız.

- Euclid
- Manhattan
- Chebyshev
- Mahalanobis

21