VERİ MADENCİLİĞİ

Fırat İsmailoğlu, PhD

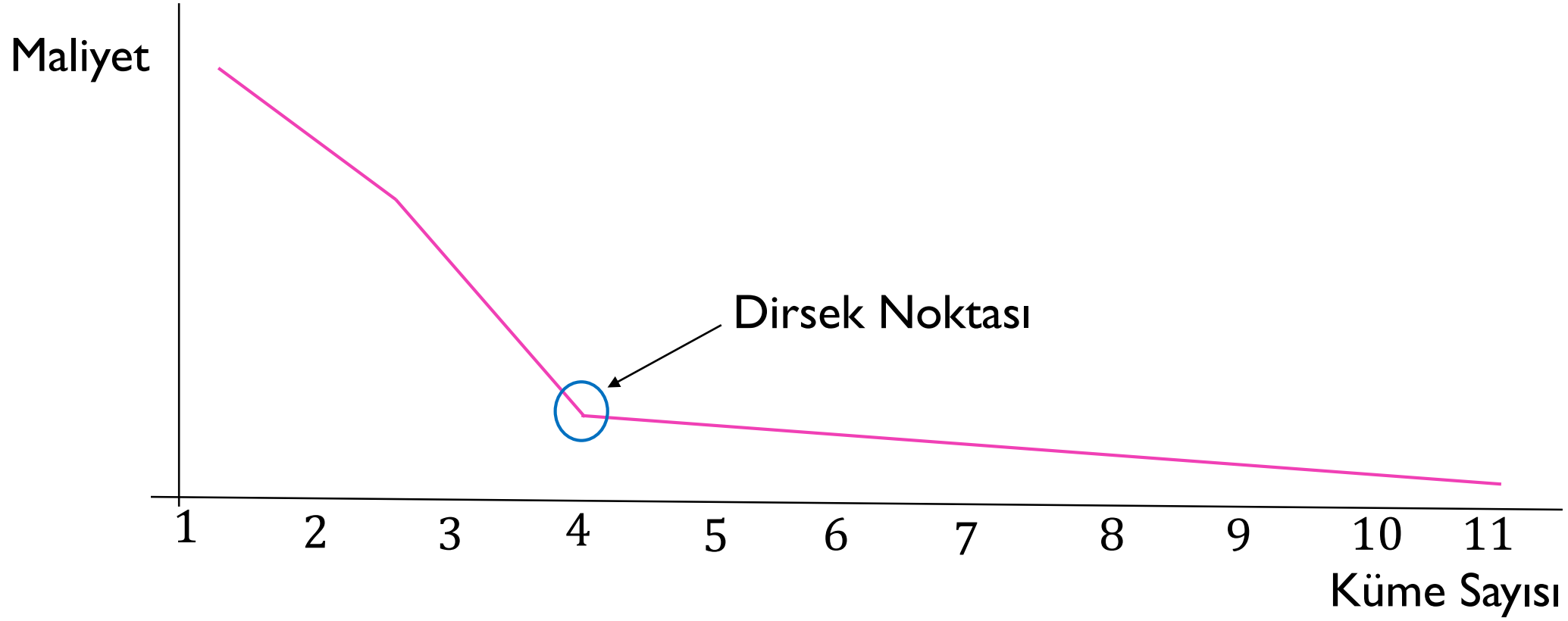
Kümeleme (Gruplama) – II

K-Ortalama Algoritmasında Küme Sayısına Nasıl Karar Veririz?

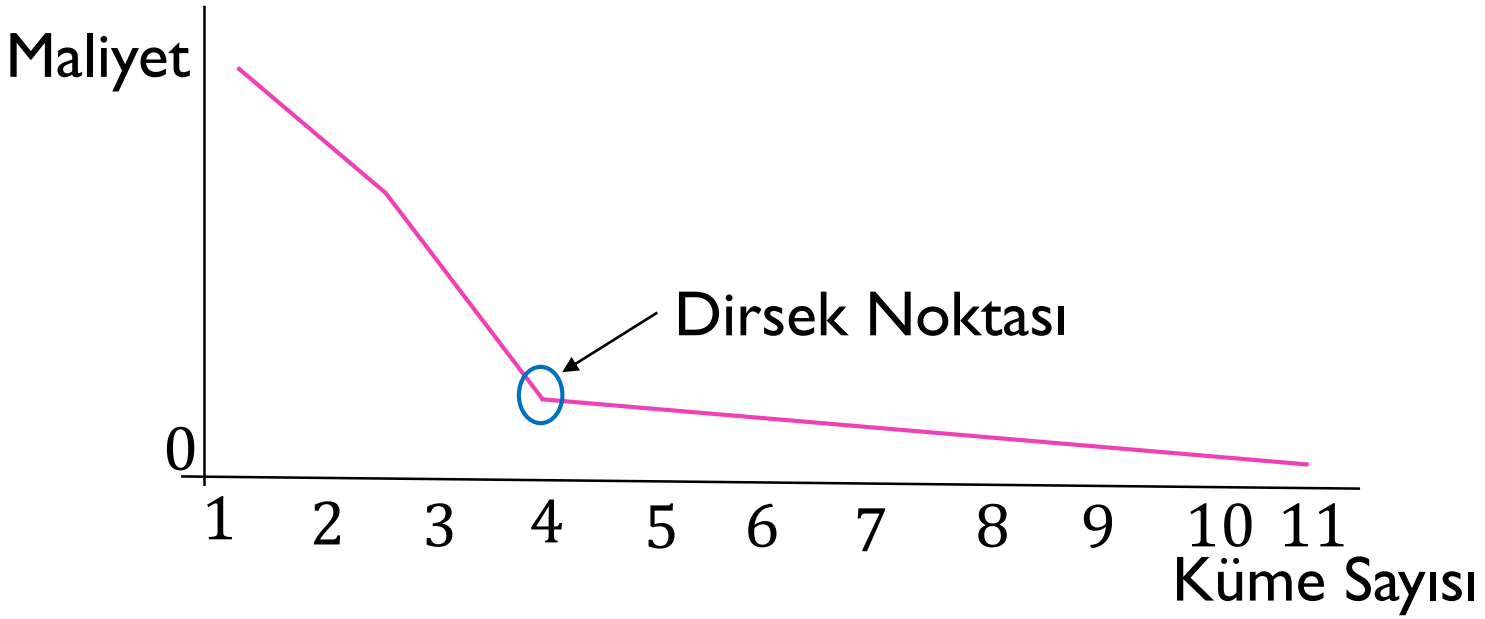
Geçen hafta k-ortalama algoritmasi ile kümeleme yaparken veri setinde kaç küme olacağını önceden bilemeyiz demiştik. Küme sayısını kullanıcıdan almıştık.

Bu hafta ise küme sayısını dirsek methodu (elbow method) ile tahmin etmeye çalışacağız.

Dirsek Methodu (Elbow Method)



Dirsek Methodu (Elbow Method)



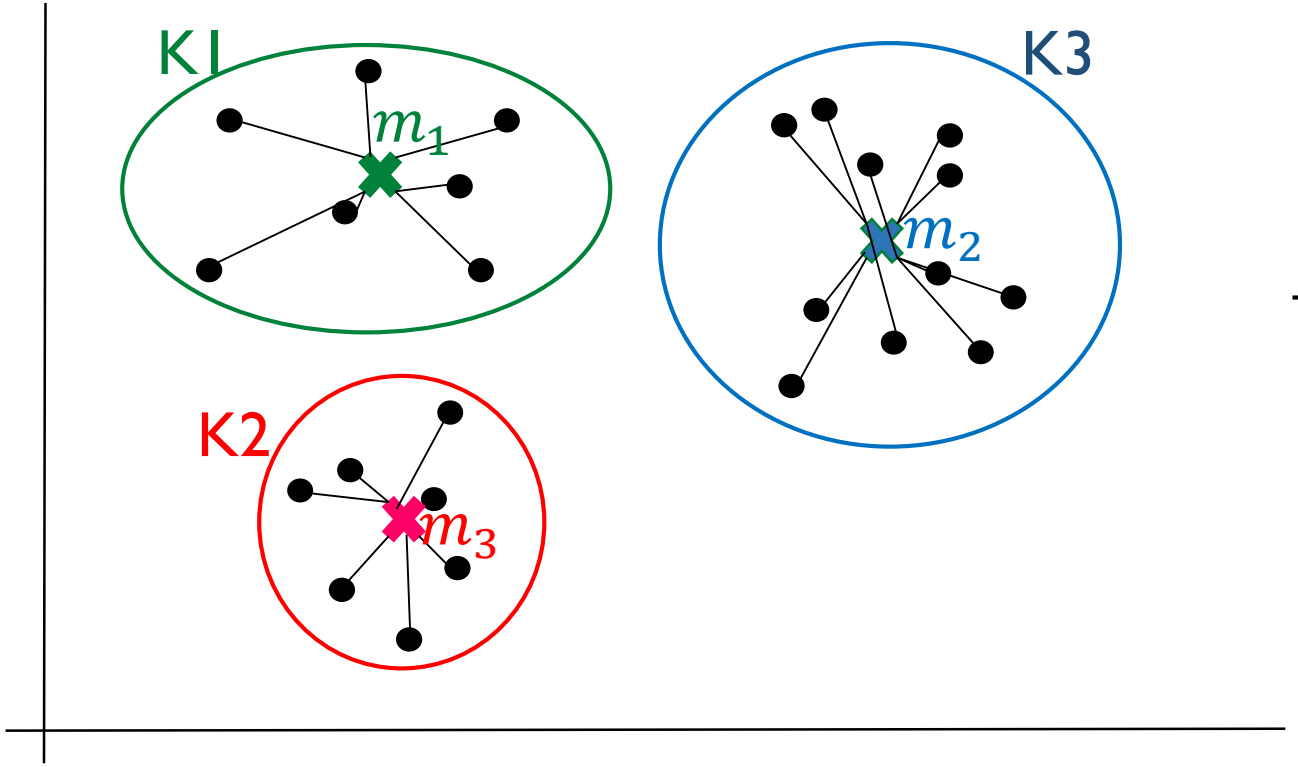
4 kümeden sonra maliyette anlamlı bir azalış gözlemlenmiyor. Bu yüzden bu veri seti için ideal küme sayısı 4'tür.

Peki maliyet k-ortalama ile kümeleme yaparken ne anlama geliyor?

Maliyet, genel olarak istemediğimiz durumların toplam değerinin sayısal ifadesidir.

Kümeleme yaparken de istemediğimiz şey her örneğin kendi küme merkezine uzak mesafede olmasıdır. Bu, cezalandırılması gereken bir durumdur.

Kümelemede Maliyet Hesabı



✖ ile işaretlenen yerler küme merkezleri

—— çizgileri örneklerin küme merkezlerine olan uzaklıkları

Maliyet yukarıda gösterilen siyah çizgilerin uzunluklarının toplamıdır. Bunun sayısal ifadesi için diyelimki $S_1, S_2, \ldots, S_k$ gibi k tane kümemiz ve $m_1, m_2, \ldots, m_k$ bu kümelerin merkez noktaları olsun.

Kümelemede Maliyet Hesabı

Maliyet:

$$\sum_{i=1}^{k} \sum_{x \in S_i} (x - m_i)^2$$

Her bir küme için — Kümemenin her elamanı için

Not: Bu şekilde hesaplanmış maliyete küme içi toplam varyasyon (total wihtin-cluster variation) da denir.

Soru: Maliyet ne zaman 0 olur?

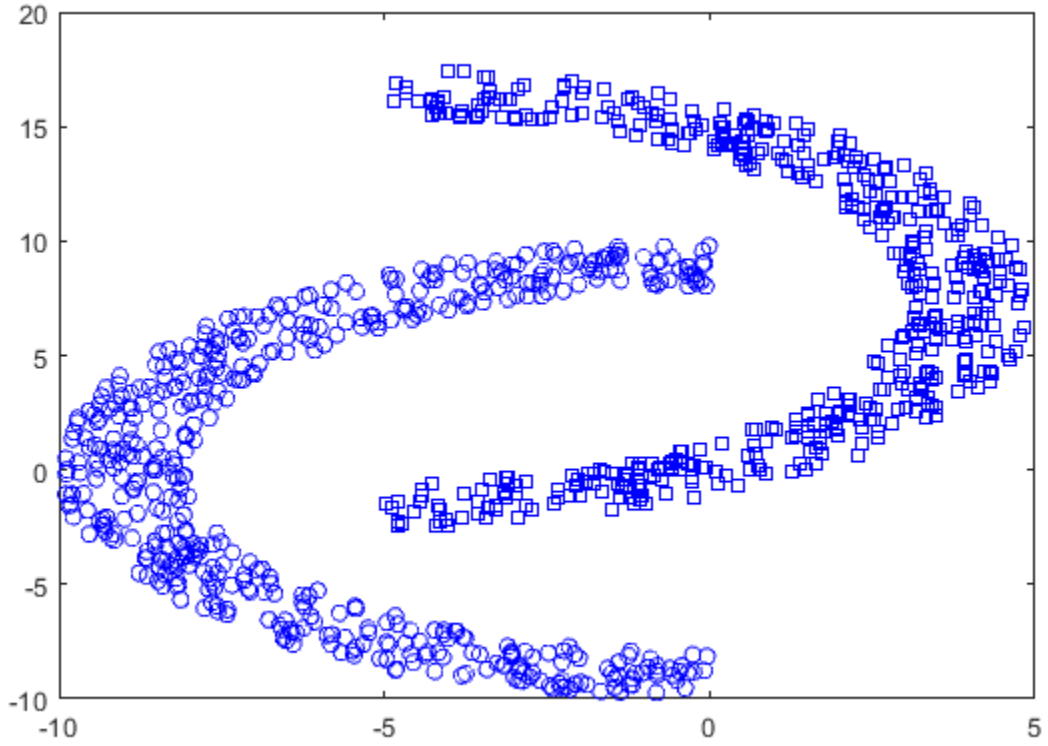
Cevap:

K-Ortalama Algoritmasının Zayıflıkları

K-ortalama algoritmasi basit fakat güçlü bir algoritmadir. Bir çok durumda verideki kümeleri bulmayı sağlar.

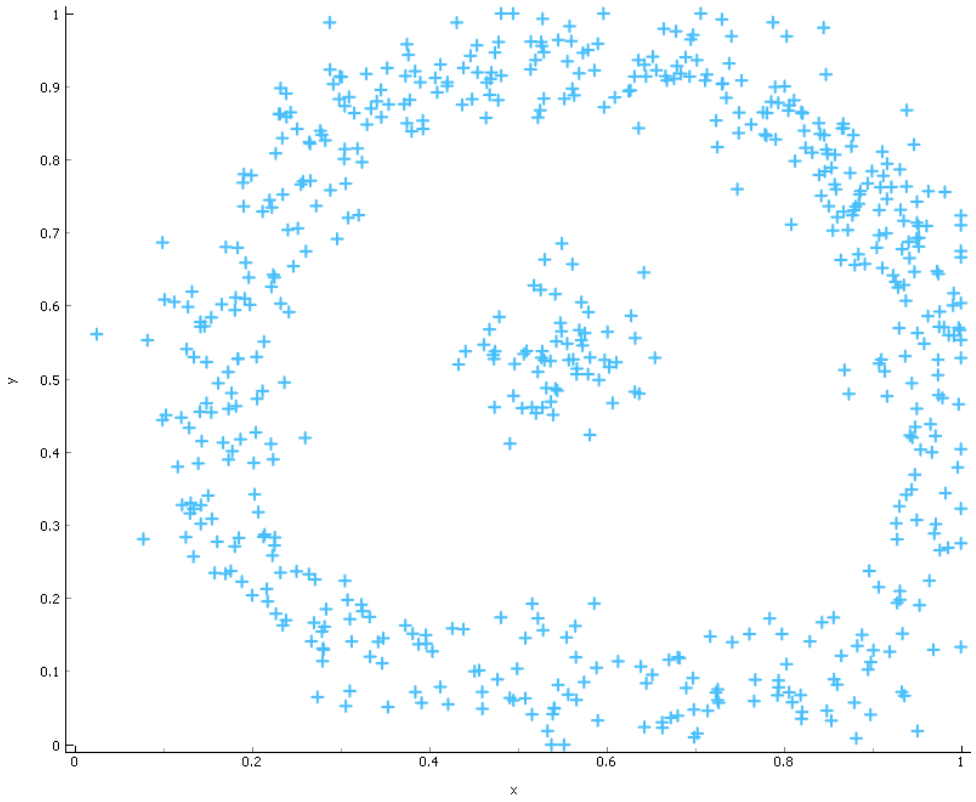
Fakat k-ortalama algoritmasi bazı durumlarda kümeleri bulamaz.

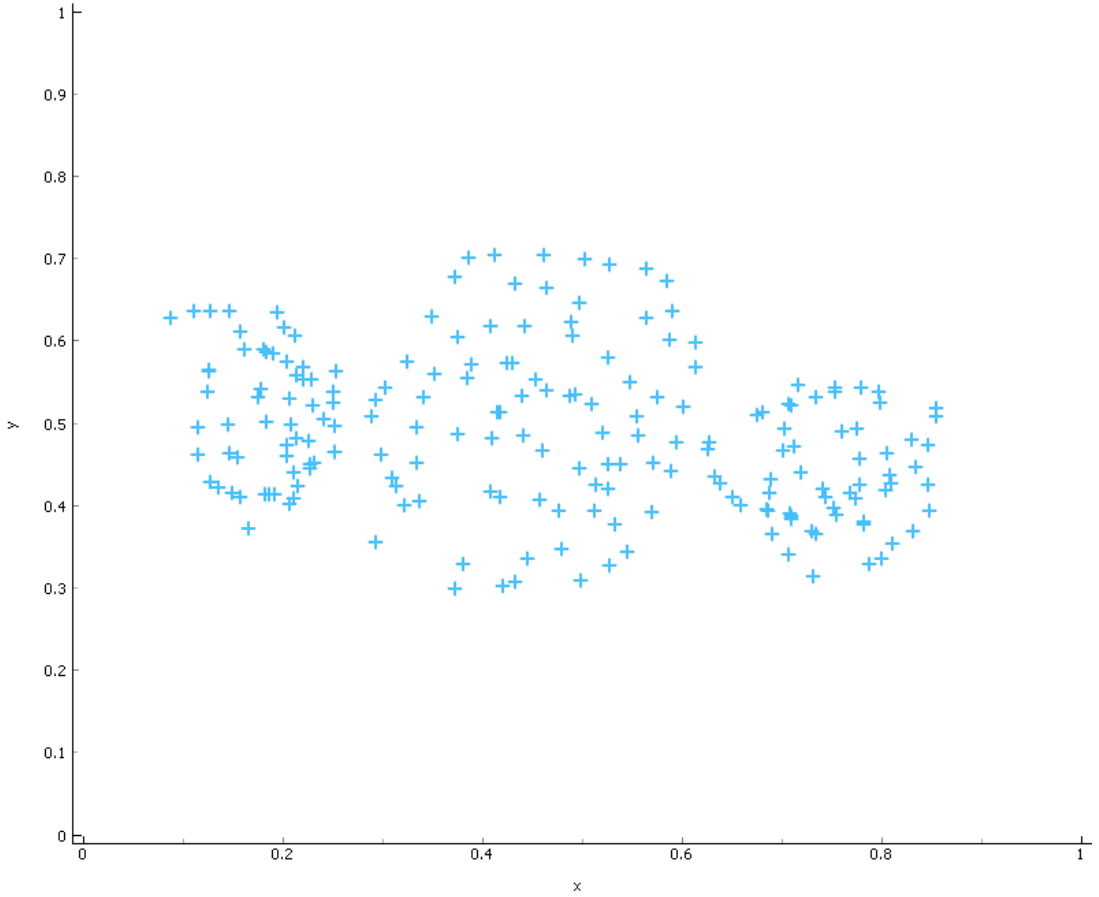
1. eğer verideki kümeler küresel (spherical) değilse:



K-Ortalama Algoritmasının Zayıflıkları

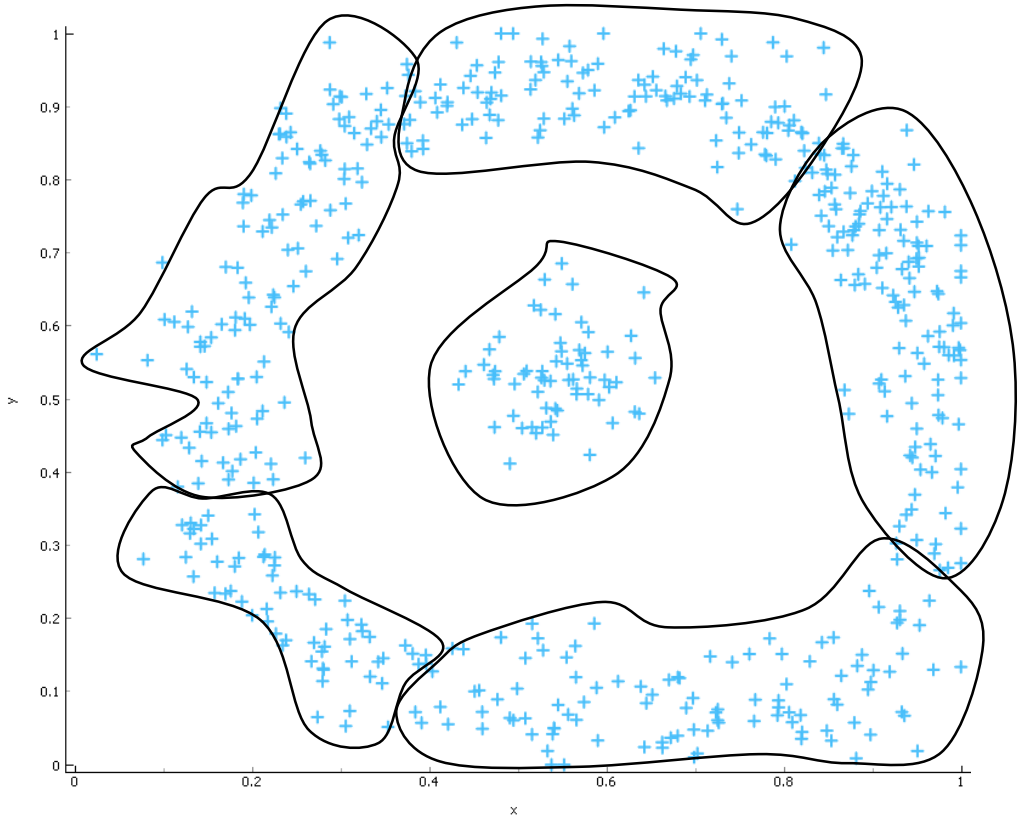
2. eğer verideki kümeler içiçe geçmişse, yada birbine çok bitişikse

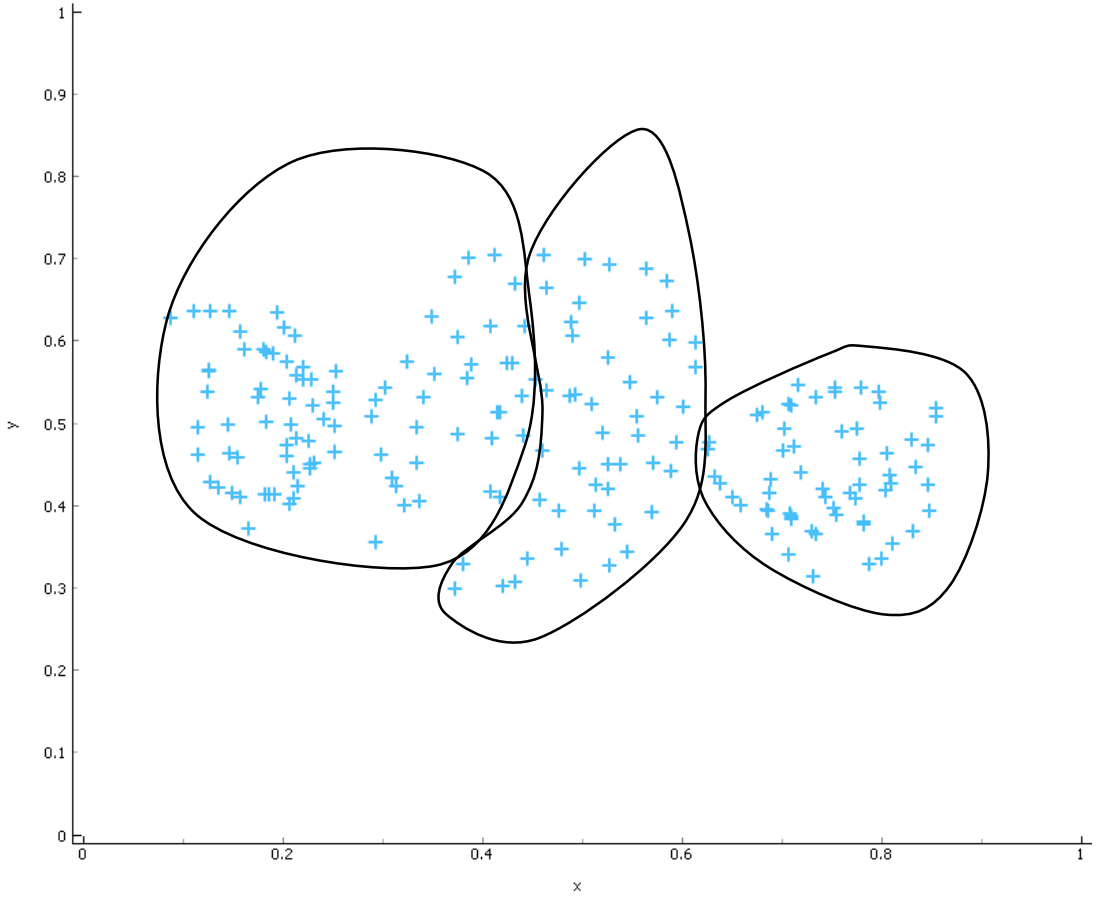




K-Ortalama Algoritmasının Zayıflıkları

K-ortalama tarafından yanlış bulunan kümeler





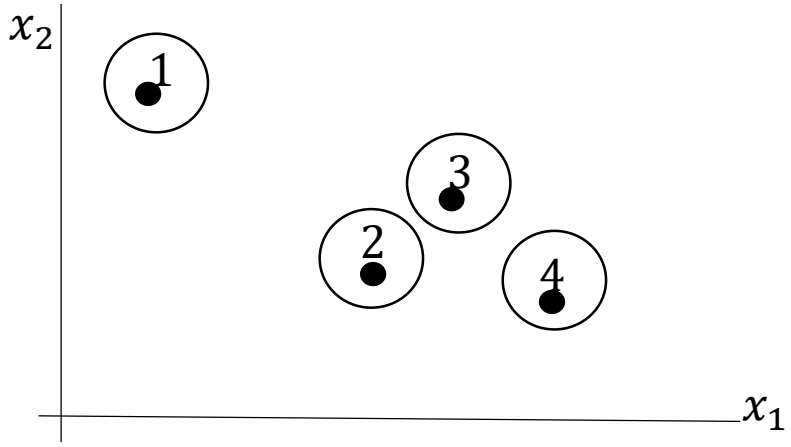
Hiyerarşik Kümeleme (Hierarchical Clustering)

K-ortalama kümeleme algoritmasindan sonraki en yaygin kullanilan kümeleme algoritmasi hiyerarşik kümelemedir.

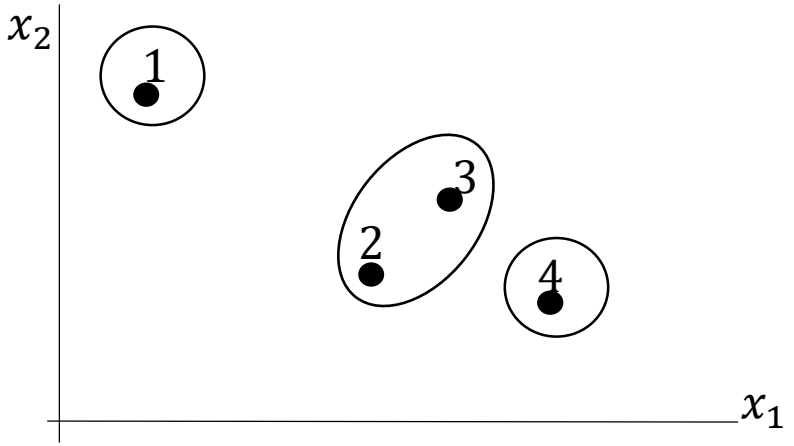
Hiyearşik kümelemede her bir örnek (nokta) bir kümeymiş gibi düşünülerek başlanır, daha sonra her bir aşamada en yakın iki küme birleştirilir. Sonuçta bütün örnekleri içeren tek bir küme elde edilir.

```
Algoritmik olarak:
her örneği bir küme kabul et.
bir küme kalana kadar tekrar et:
en yakın iki kümeyi birleştir.
```

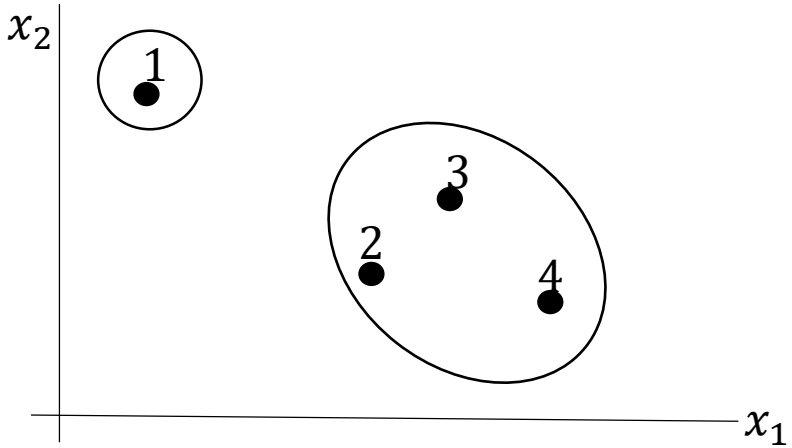
Hiyerarşik Kümeleme (Hierarchical Clustering)



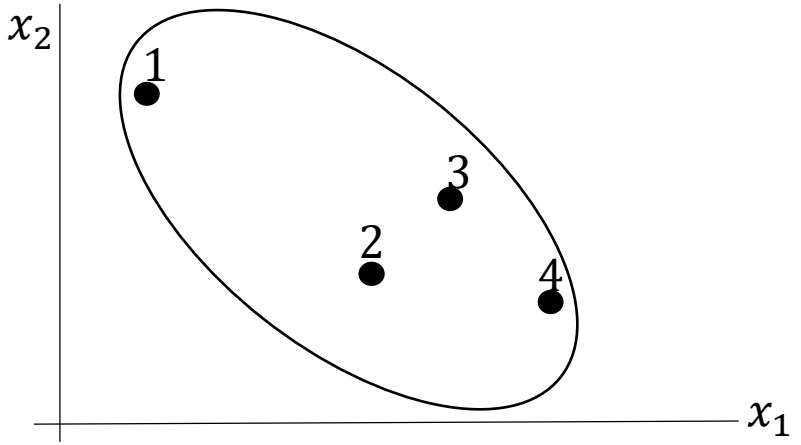
Başlangıçta her nokta kendi başına bir küme



Birbirine en yakın iki kume 2 ve 3 nolu noktaların kümesi olduğundan bu iki kümeyi birleştiriyoruz.



4 nolu noktanin olduğu kümeyi, 2 ve 3 nolu noktaların kümesine birleştiriyoruz.



1 nolu noktanin olduğu kümeyi, 2,3 ve 4 nolu noktaların kümesine birleştiriyoruz ve tek bir küme elde ediyoruz.

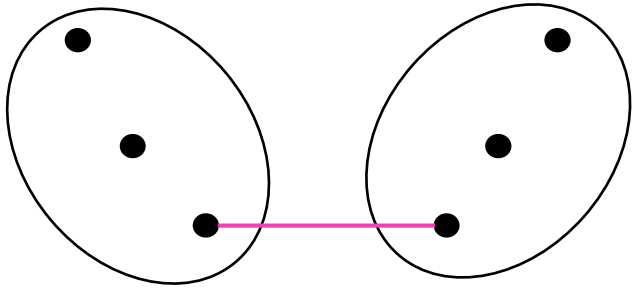
İki Kümenin Birbirine Uzaklığını Nasıl Hesaplarız?

İki kümenin birbirne olan uzaklığını hesaplamak için 3 yöntem kullanılır.

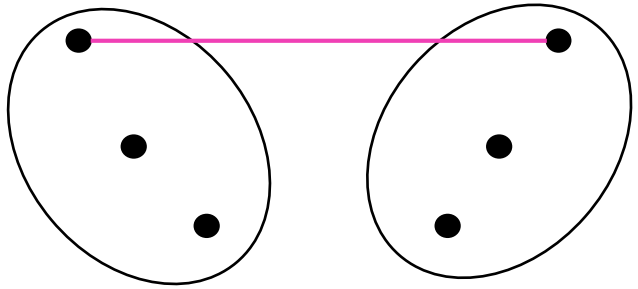
i. Tek bağlantı (single link) (min uzaklık):İki kümenin birbirine en yakın noktaları arası uzaklık.

ii. Tam bağlantı (complete link) (max uzaklık): İki kümenin birbirine en uzak noktaları arası uzaklık.

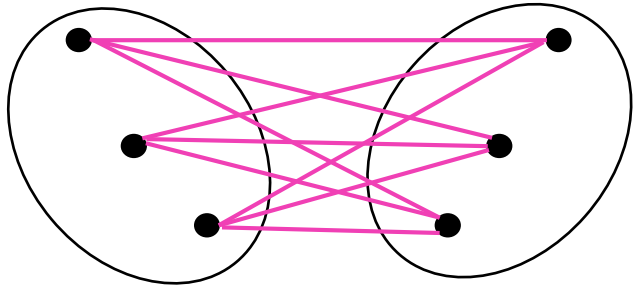
iii. Ortalama grup: İki kümedeki bütün elemanların birbirine olan uzaklıklarının ortalaması.



Tek Bağlantı
(Single link)

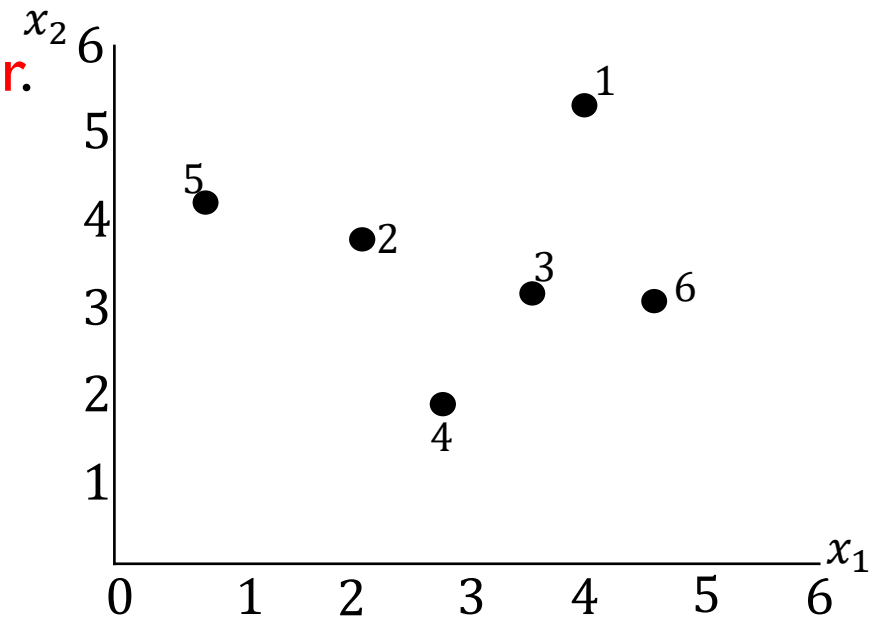


Tam Bağlantı
(Complete link)



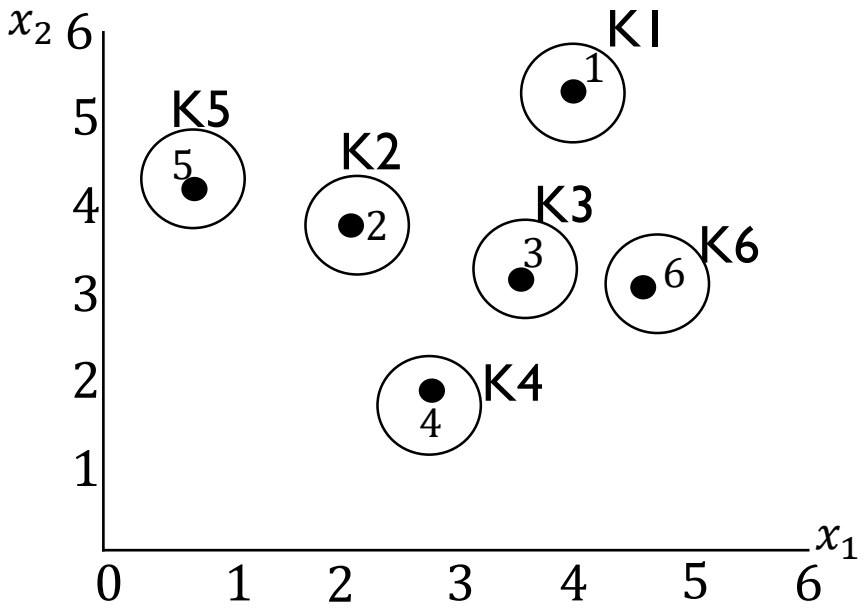
Ortalama Grup

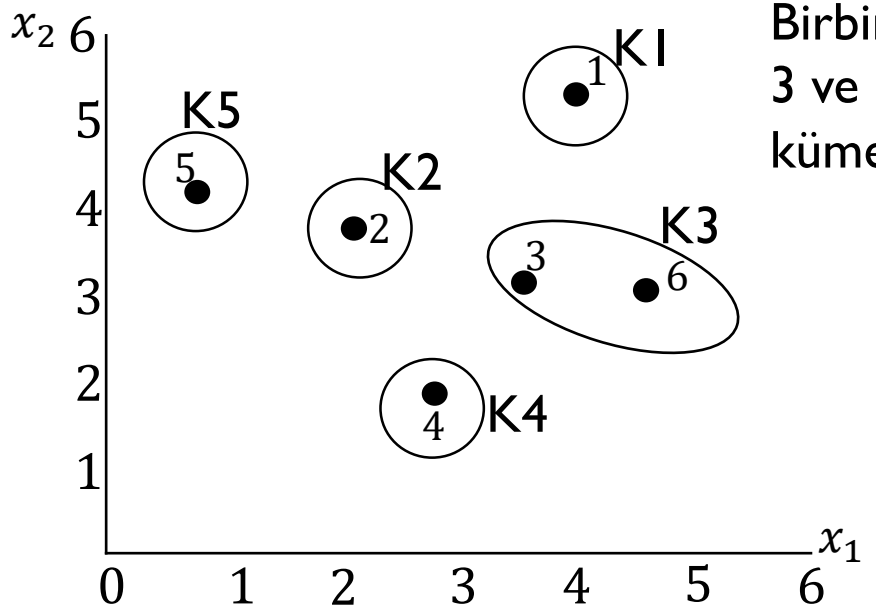
ör.



öklid	1	2	3	4	5	6
1	0	0.24	0.22	0.37	0.34	0.23
2	0.24	0	0.15	0.2	0.13	0.25
3	0.22	0.15	0	0.14	0.28	0.11
4	0.37	0.2	0.14	0	0.29	0.22
5	0.34	0.13	0.28	0.29	0	0.39
6	0.23	0.25	0.11	0.22	0.39	0

Yukarıda x_1 ve x_2 özellikleri ile ifade edilen 6 örnek (nokta) ve bu örneklerin birbirine olan öklid uzaklıkları verilmiştir. Bu örnekleri kümeler arası uzaklığı tek bağlantı ile hesaplayarak hiyerarşik gruplayalım.





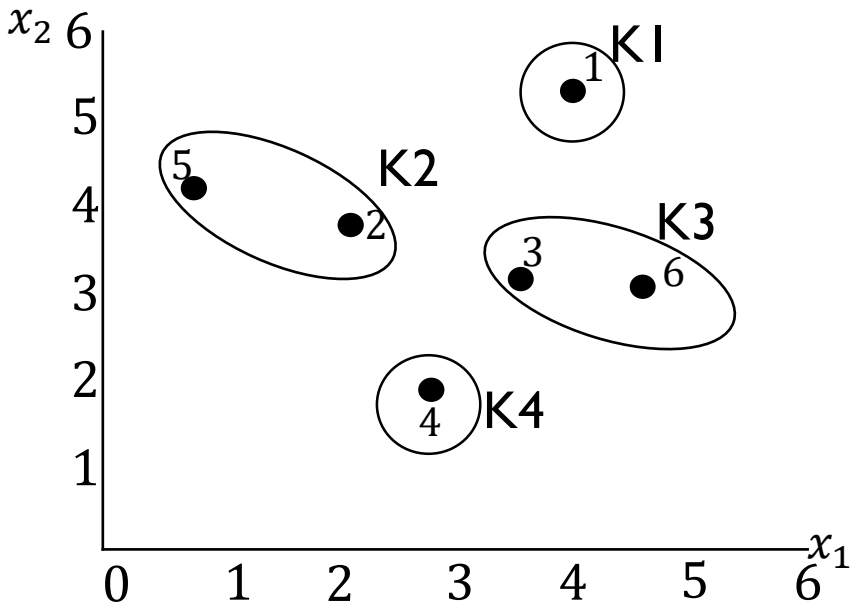
Birbirine en yakın iki nokta 3 ve 6 olduğuiçin bunların kümelerini birleştirdik.

tek bağlantı	K1	K2	K3	K4	K5
K1	0	0.24	0.22	0.37	0.34
K2	0.24	0	0.15	0.20	0.13
K3	0.22	0.15	0	0.14	0.28
K4	0.22	0.20	0.14	0	0.29
K5	0.34	0.13	0.28	0.29	0

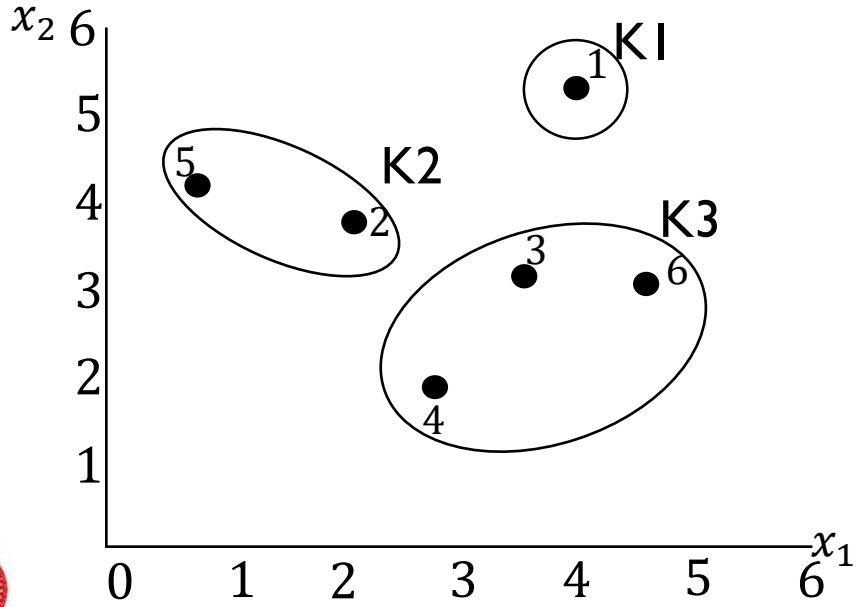
Burada kümelerin birbirine uzaklıklarını tek bağlantı ile hesapladık. Örnek olarak

$$tek\ bağlantı(K2, K3) = min(öklid(2,3), öklid(2,6))$$
$$= min(0.15, 0.25)$$
$$= 0.15$$

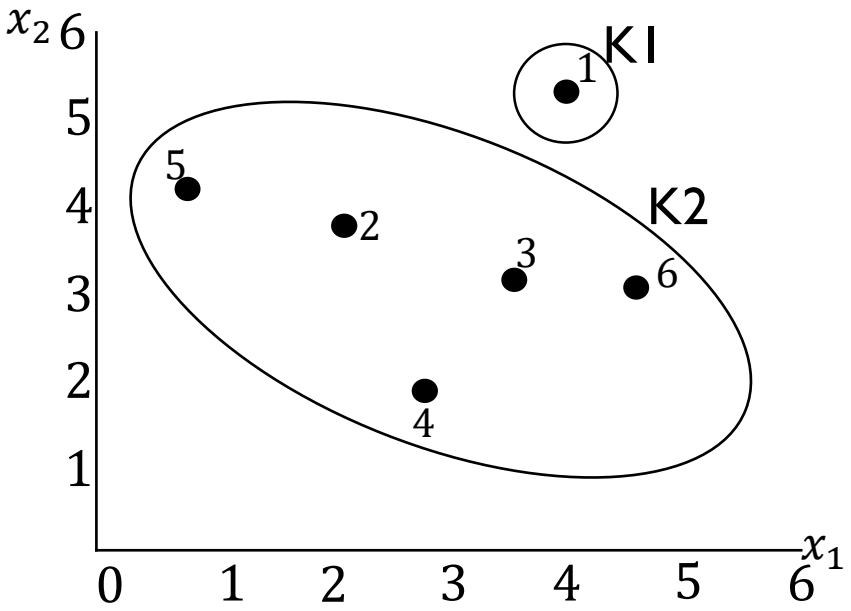
Birbirine yakın iki küme K2 ve K5, bu aşamada bu iki kümeyi birleştireceğiz.

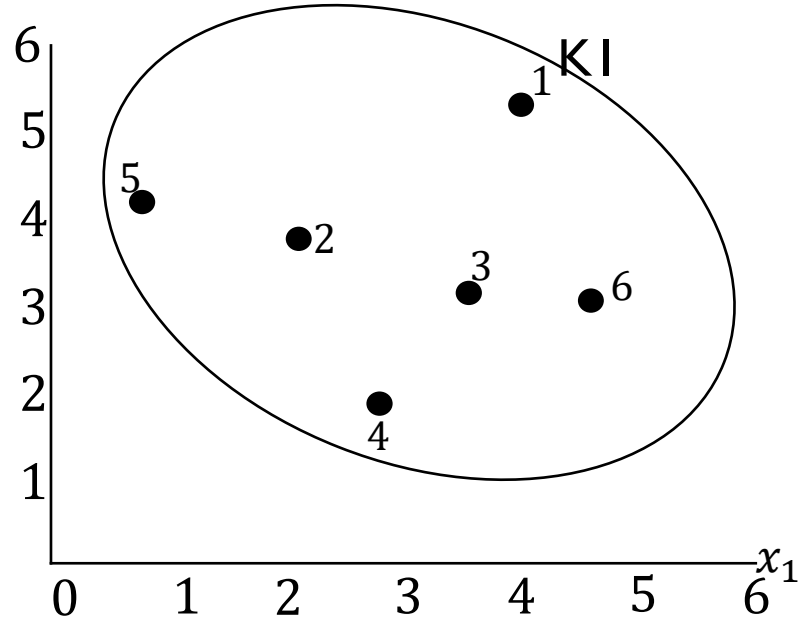


tek bağlantı	K1	K2	K3	K4
K1	0	0.24	0.22	0.37
K2	0.24	0	0.15	0.20
K3	0.22	0.15	0	0.14
K4	0.37	0.2	0.14	0



tek bağlantı	K1	K2	K3
K1	0	0.24	0.22
K2	0.24	0	0.15
K3	0.24	0.15	0





Alıştırma olarak aynı kümeleri tam bağlantı ve ortalama grup kümeler arası mesafelerini kullanarak gruplayınız. Bu şekilde farklı bir hiyerarşik kümeleme mi elde ederiz?

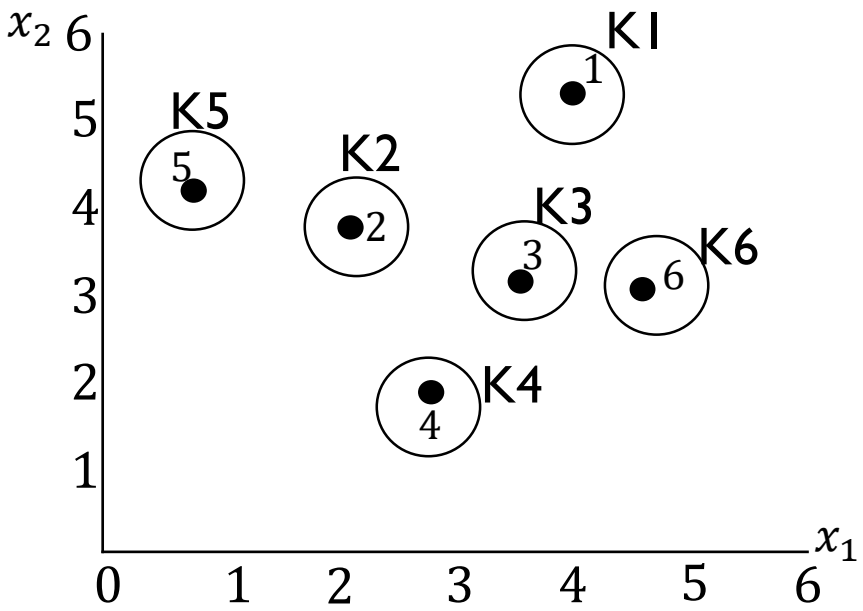
Dendogram

Hiyerarşik kümeleme *dendogram* ile gösterilebilir. Dendogram yatay bir ağaç şeklindedir.

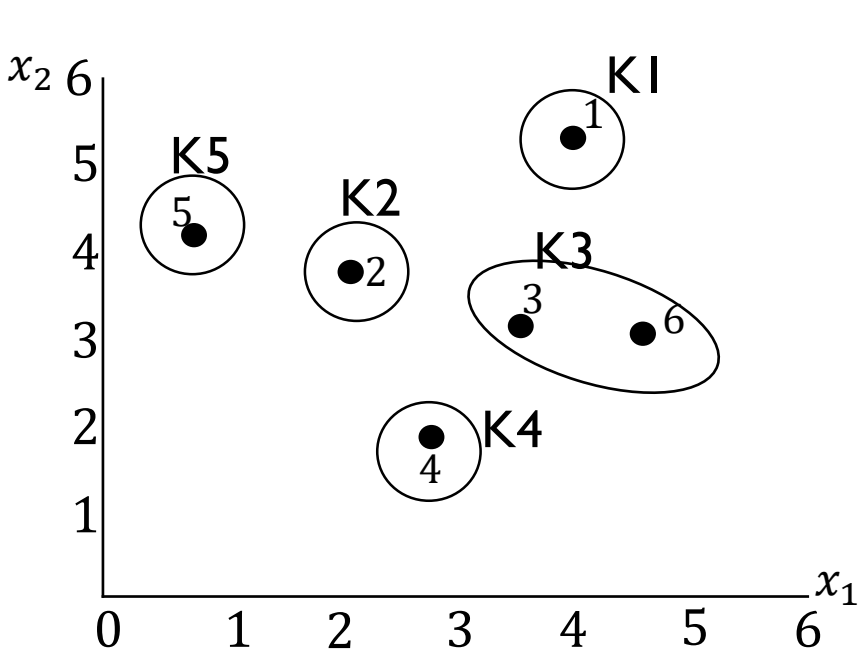
Yatay eksen kümelerin birbirine uzaklığını gösterir.

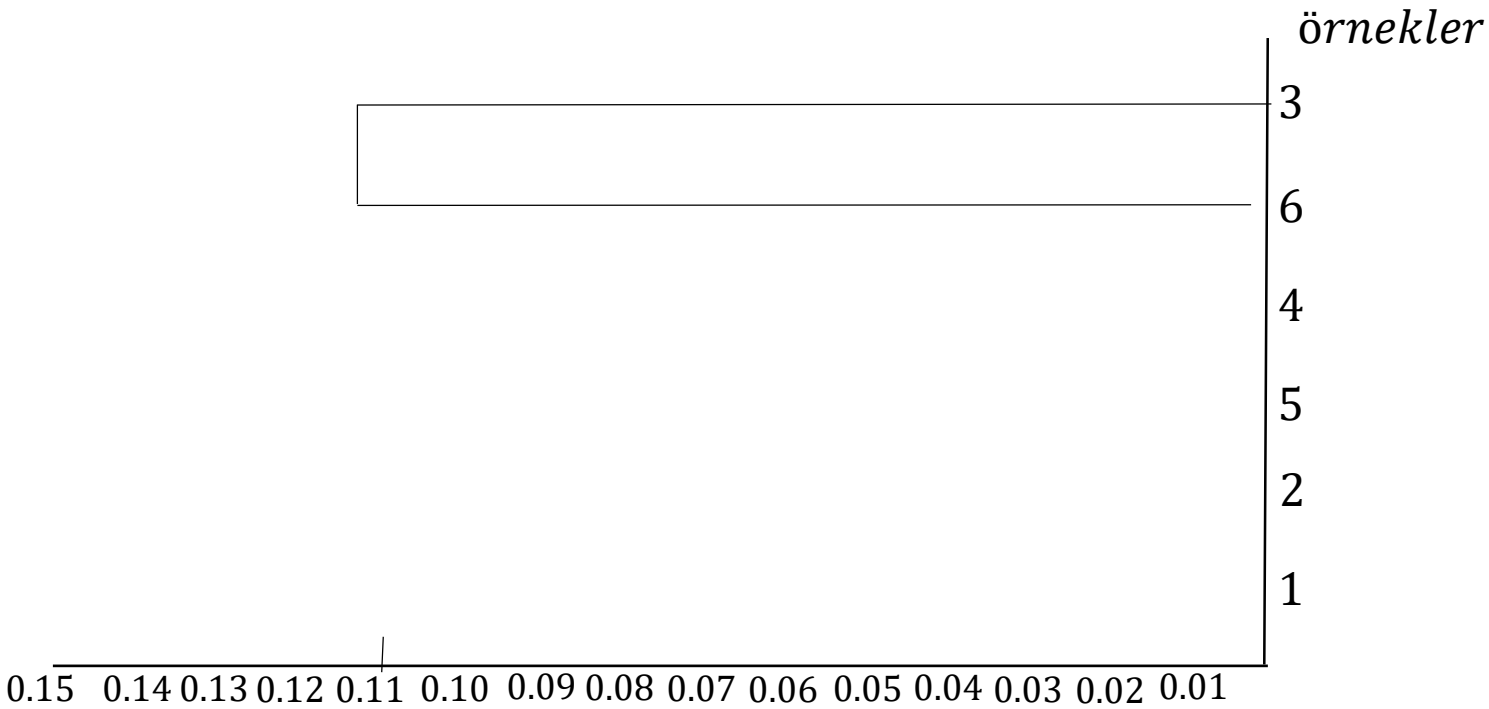
Dikey eksen örnekleri ve kümeleri gösterir.

ör.

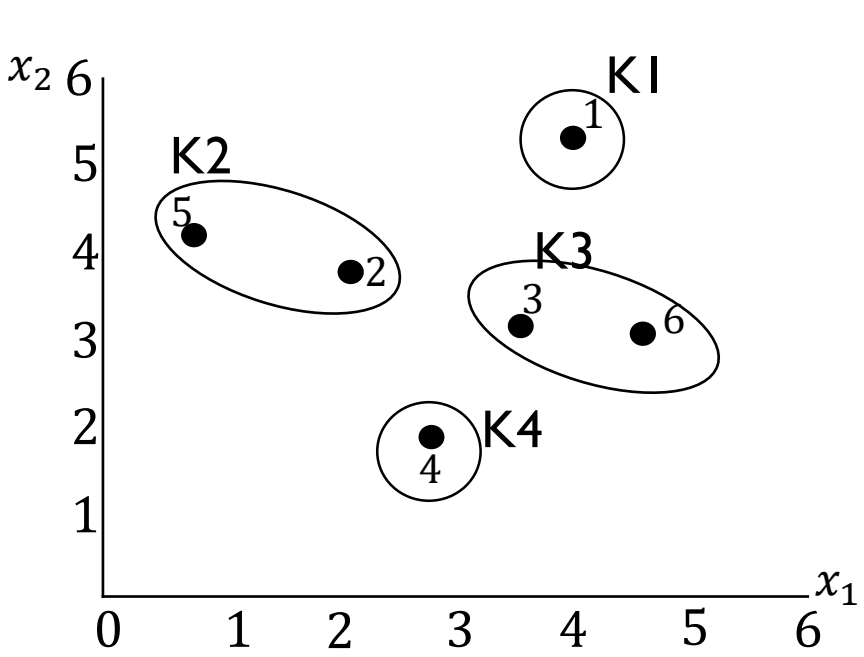


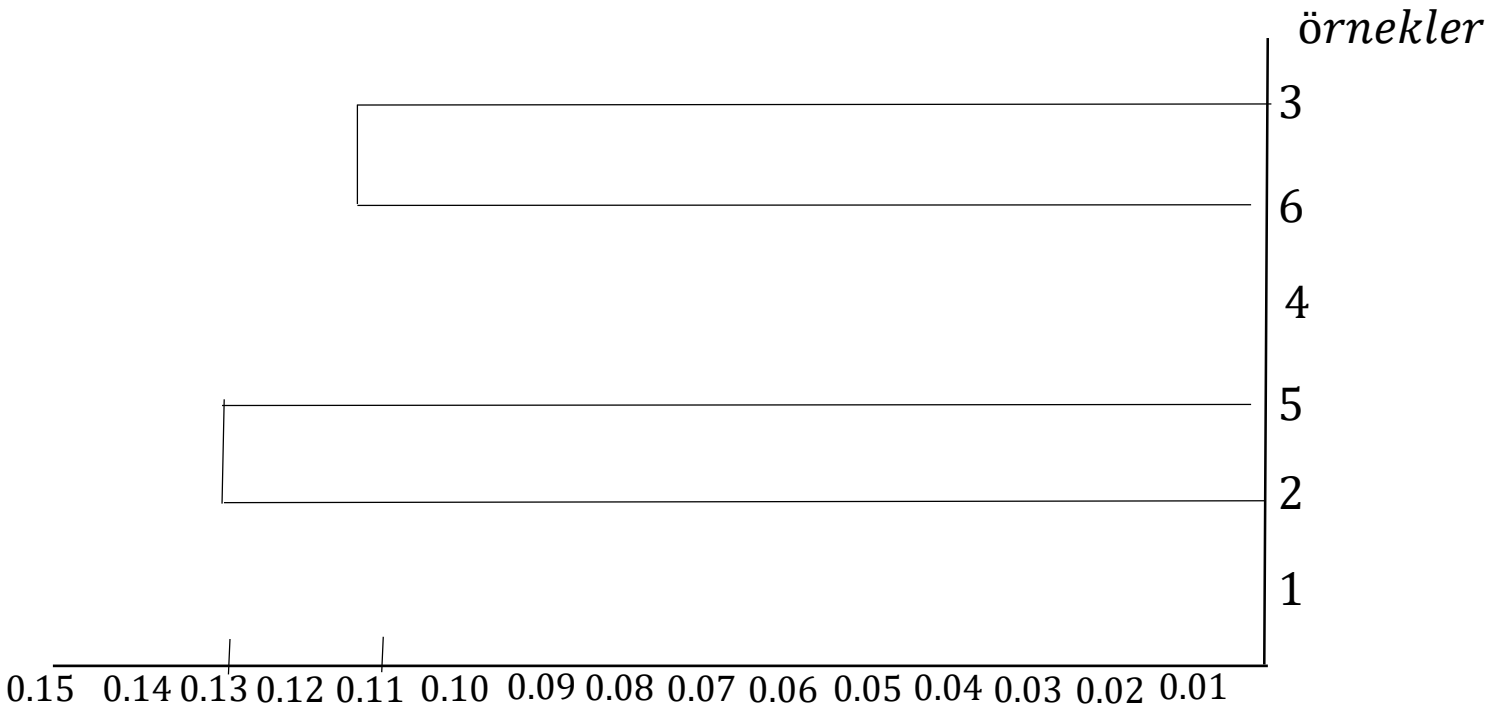
Dendogram


x_2
6
5
4
3
2
1
K1
K5
K2
K3
K4
1
5
2
3
6
4
0
1
2
3
4
5
6
x_1

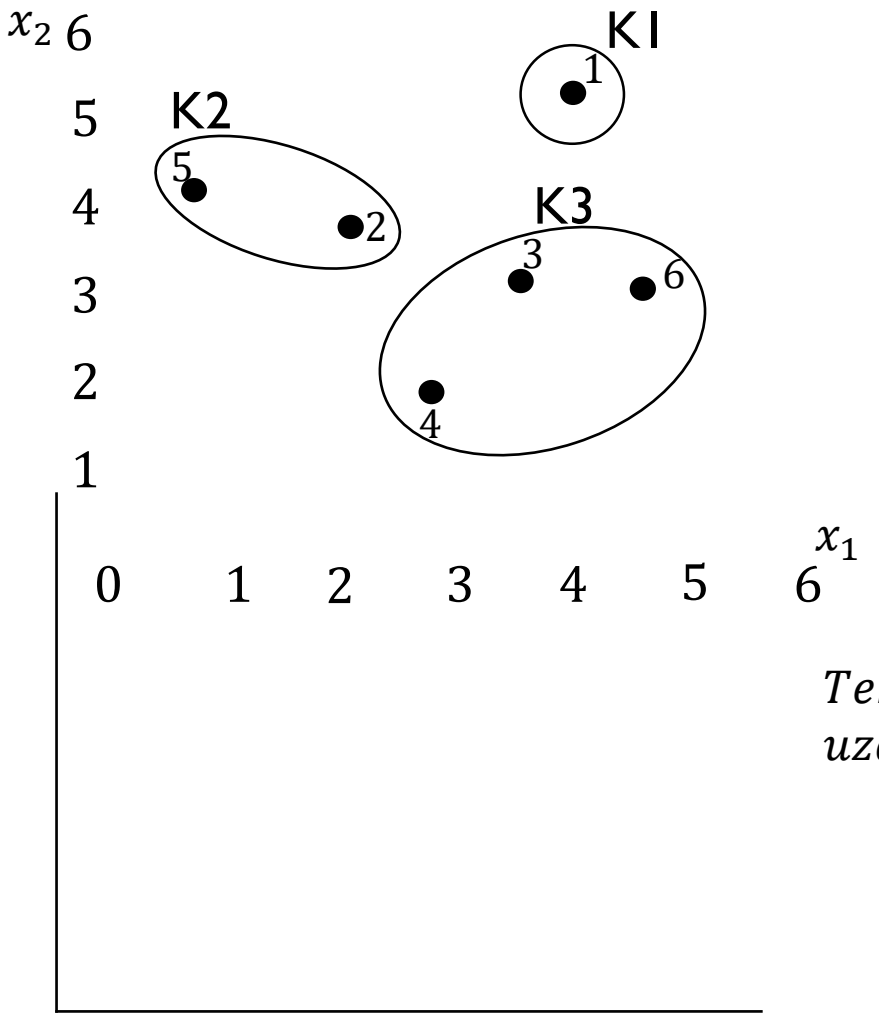

örnekler
3
6
4
5
2
1
0.15
0.14
0.13
0.12
0.11
0.10
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
Tek bağlantı uzaklığı

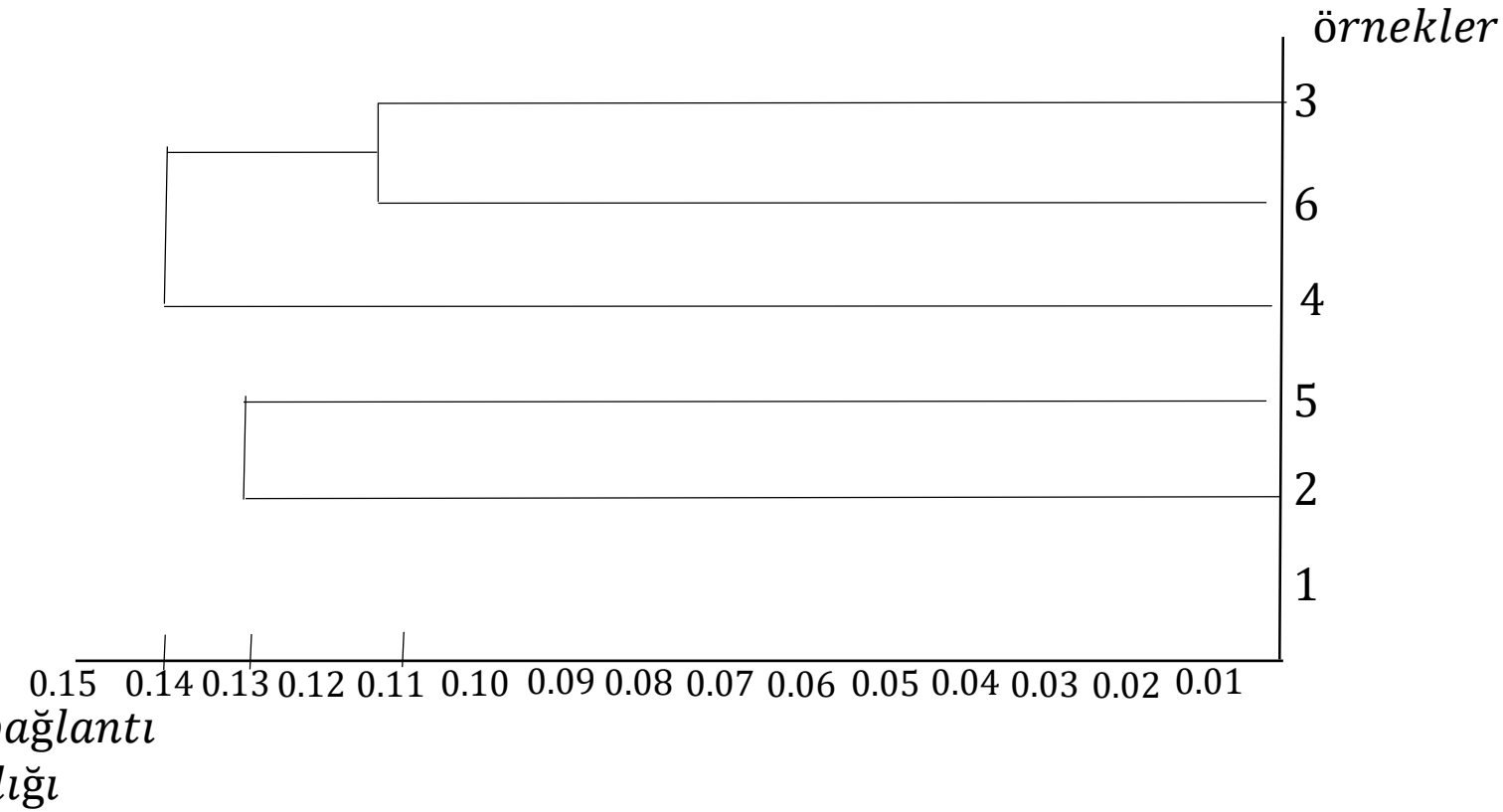
Dendogram


x_2
6
5
4
3
2
1
K1
K2
K3
K4
1
2
3
4
5
6
0
1
2
3
4
5
6
x_1

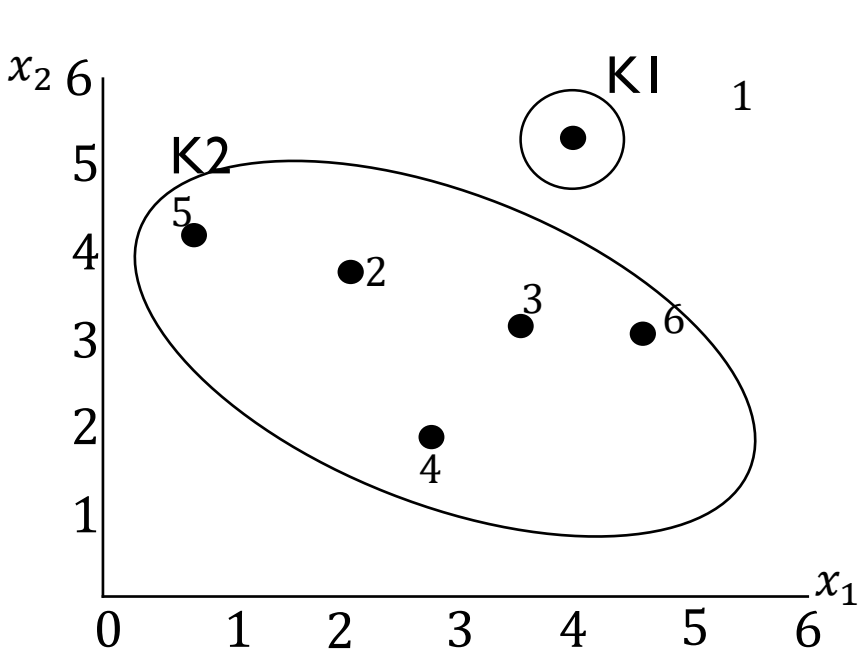

örnekler
3
6
4
5
2
1
0.15
0.14
0.13
0.12
0.11
0.10
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
Tek bağlantı uzaklığı

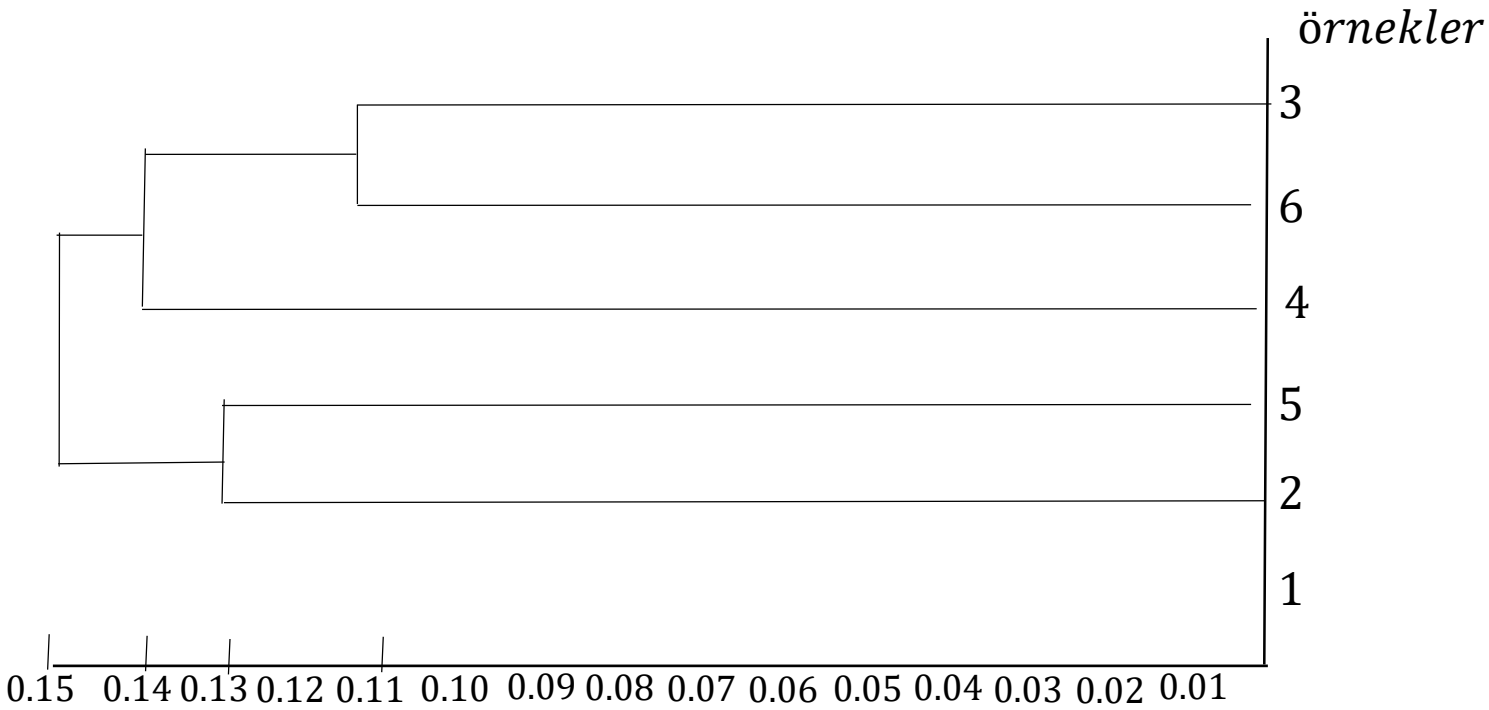
Dendogram


x_2
6
5
4
3
2
1
K1
K2
K3
1
2
3
4
5
6
0
1
2
3
4
5
6
x_1


örnekler
3
6
4
5
2
1
0.15
0.14
0.13
0.12
0.11
0.10
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
Tek bağlantı uzaklığı

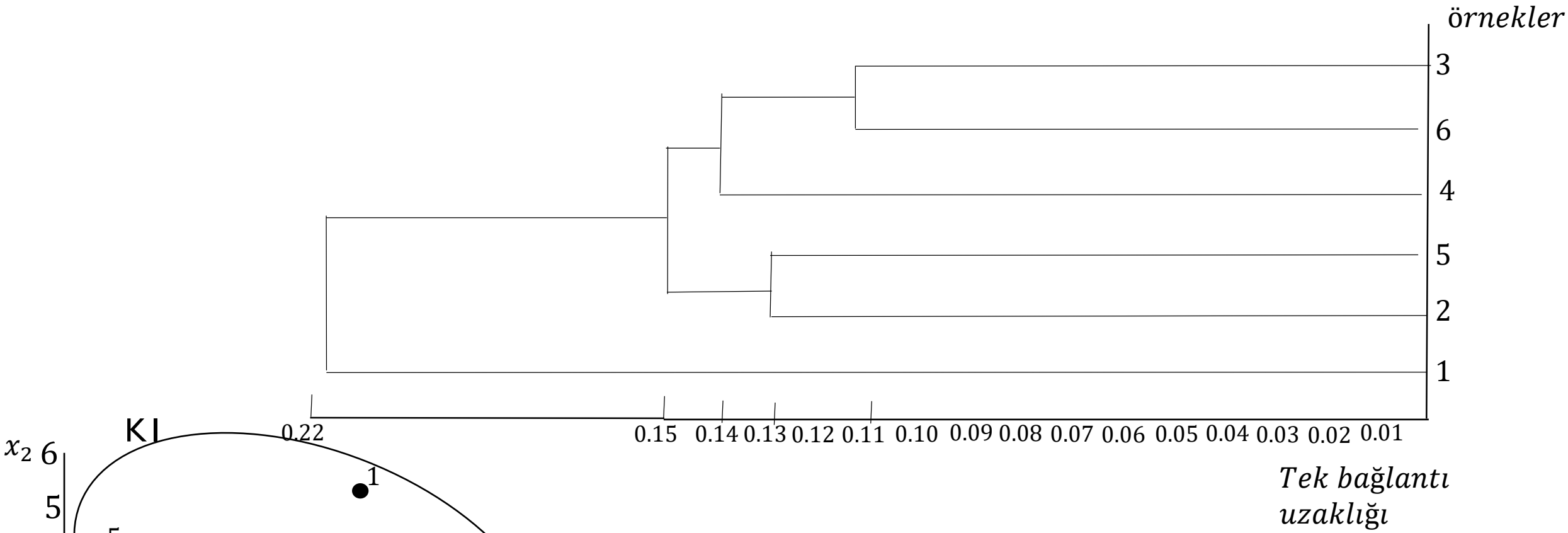
Dendogram

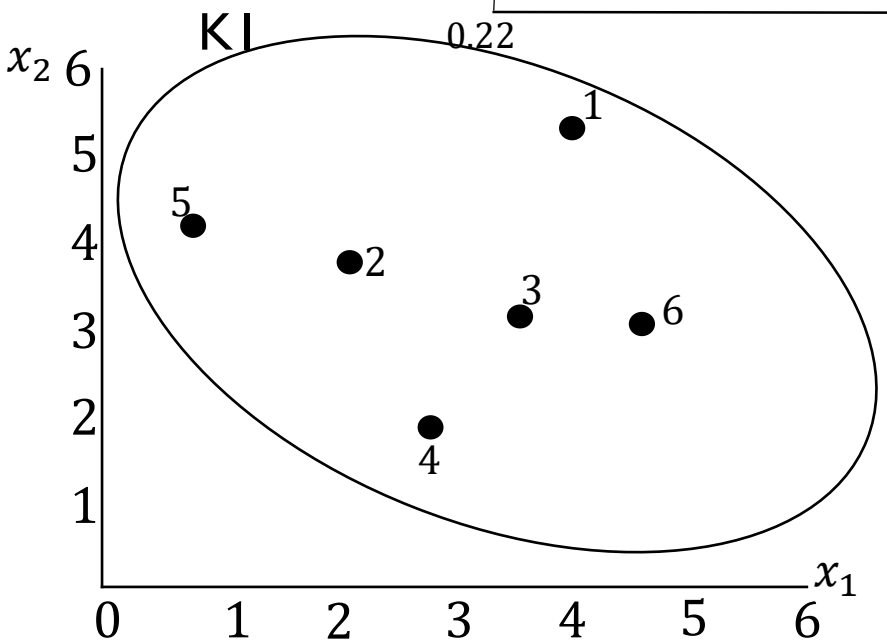




Tek bağlantı uzaklığı

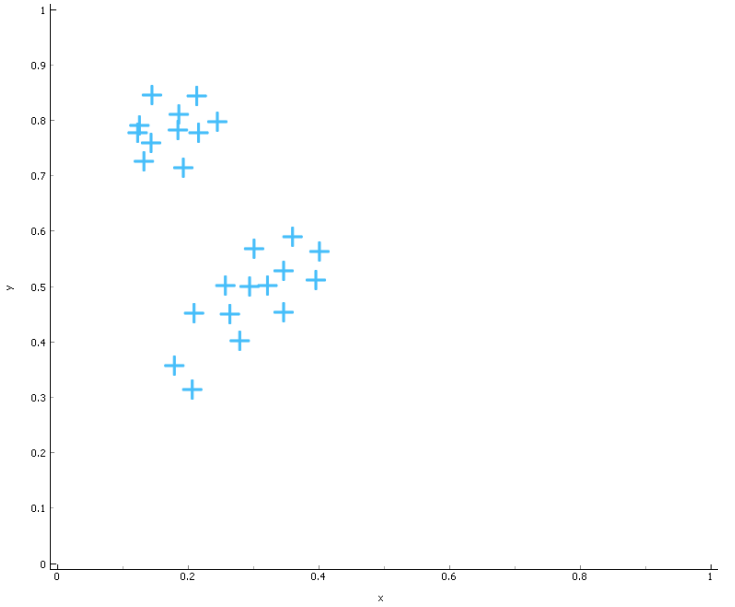
Dendogram

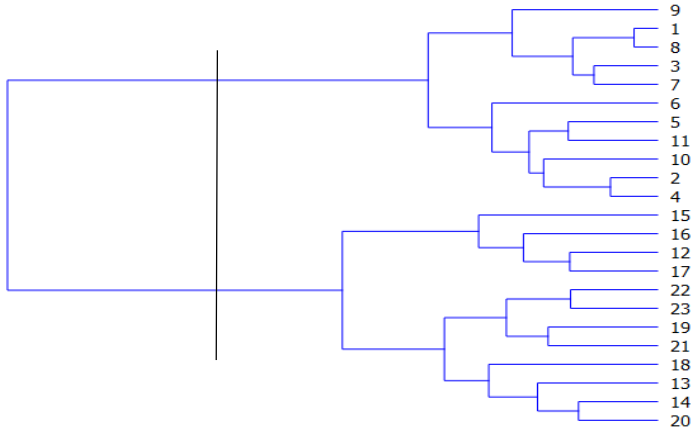

örnekler
3
6
4
5
2
1
0.22
0.15 0.14 0.13 0.12 0.11 0.10 0.09 0.08 0.07 0.06 0.05 0.04 0.03 0.02 0.01
Tek bağlantı uzaklığı


KI
x_2
6
5
4
3
2
1
0 1 2 3 4 5 6
x_1
1
2
3
4
5
6

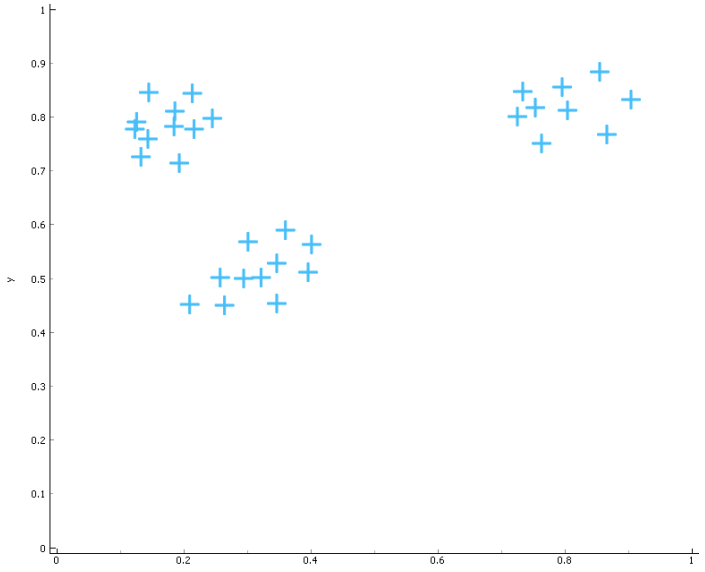
Dendogram ile Küme Sayısına Karar Verme

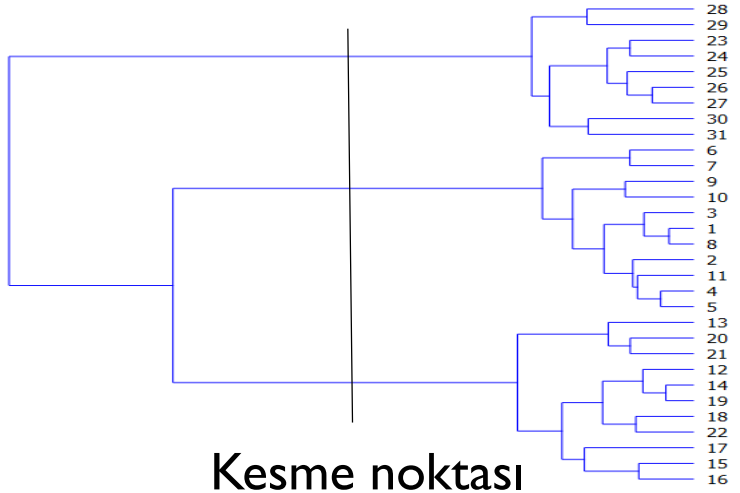
Dendogram ile çoğu kez veri setimizde kaç küme olduğuna karar verebiliriz. Bunun için dendogramda sıçrayış yapılan yere bakılır. Bu, eklenen kümenin çok uzak mesafede olduğunu gösterir.





Kesme noktası





Kesme noktası