Veri Madenciliği ve Bilgi Keşfi
H2. Veri , Veri Toplama ve Veri Önişleme

Dr. Öğr. Üyesi Alper Talha KARADENİZ
Samsun Üniversitesi
Yazılım Mühendisliği Bölümü

Veri nedir?

Özellik ve veri nesnelerinden oluşan koleksiyon

Özellik veriyi açıklayan veya bir karakteristiğini ortaya koyan bilgi demektir.

Veri, kayıt, nokta, durum, örnek, gözlem, girdi gibi de isimlendirilebilir.

Özellikler, karakteristik, değişken, boyut, alan olarak de isimlendirilebilir.

Özellikler

Nesneler

No	Indirim	Medeni Hal	Gelir	Sipariş
1	Evet	Bekar	125K	**Hayır**
2	Hayır	Evli	100K	**Hayır**
3	Hayır	Bekar	70K	**Hayır**
4	Evet	Evli	120K	**Hayır**
5	Hayır	Dul	95K	**Evet**
6	Hayır	Evli	60K	**Hayır**
7	Evet	Dul	220K	**Hayır**
8	Hayır	Bekar	85K	**Evet**
9	Hayır	Evli	75K	**Hayır**
10	Hayır	Bekar	90K	**Evet**

Veri Tipleri

1. **Nominal veriler:** Daha fazla ifadesi ile kullanılmazlar. ID, göz rengi vb.
2. **Sıralama Ölçeği (Ordinal):** Daha fazla ifadesi ile kullanılabilir. Uzunluk (uzun, orta boy, kısa) vb.
3. **Aralıklı (interval):** Fahrenhait, Celcius vb.
4. **Oran (Ratio):** Başlangıç noktası 0'dır ve 0 noktası yokluk ifade eder. Kelvin, uzunluk (cm), zaman gibi.

Oran ve Aralıklı Veri Tipi

- Sıcaklık gibi iki değer arasında katı olarak ifade edilemeyen durumlarda Aralıklı(interval) veri tipi tanımlaması yapılır. Örneğin 50° 100°'nin yarısıdır diyemeyiz. Yani aralarında belirli bir oran söz konusu değildir.

- Uzunluk ya da ağırlık değişkeninde ise 50 kg 100 kg'nin yarısıdır deriz. Bu nedenle aralarında bir oran söz konusudur.

	Veri Tipi	Açıklama	Örnek
Kategorik (Kalitatif) Veriler	Nominal	Eşitlik ya da eşitsizlik durumu söz konusu ($=, \neq$)	PK Kodu, göz rengi, cinsiyet vb.
	Ordinal	Aralarında büyüklük küçüklük ilişkisi var ($<, >, \leq, \geq$)	Not (AA, BA, BB), Kalite Derecesi (İyi, Orta, Kötü)
Nümerik (Kantitatif) Veriler	Interval	Toplama çıkarma gibi işlemler yapılabilir (+, -)	Takvim günleri, Santigrat ya da Fahrenayt gibi sıcaklık ölçüsü
	Ratio	İki değer arasındaki oran anlamlıdır. (*, /)	Kelvin sıcaklığı, yaş, ağırlık, uzunluk gibi.

Veri Dönüşümleri

Veri Tipi		Dönüşüm	Örnek
Kategorik	Nominal	Herbir değer başka bir değerle değiştirilir.	Öğrenci numarasının başka değerle değiştirilmesi
	Ordinal	Değerler sıralama olacak şekilde değiştirilir.	İyi, orta, kötü değerlerinin 3,2,1 ile değiştirilmesi
Nümerik	Interval	Yeni_değer=a*eski_değer+b	Fahrenayt ile celsius arasındaki dönüşüm
	Ratio	Yeni_değer = a*eski_değer	Metre ve feet arasındaki dönüşüm

Sürekli ve Kesikli Değişkenler

- Kesikli Değişken
  - Belirli sayıda değere sahiptir.
  - Genellikle tam sayılı değerler ile ifade edilir.
  - İkili (Binary) gösterim bu gösterimin bir çeşididir.
- Sürekli Değişken
  - Sınırsız sayıda değere sahiptir.
  - Genellikle küsuratlı değerler ile ifade edilir.

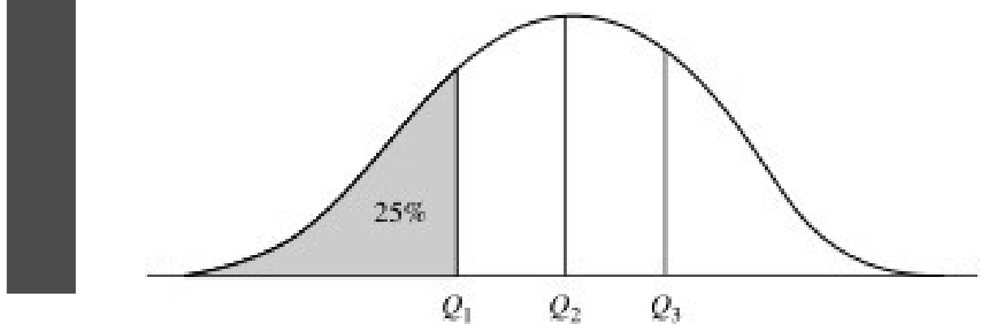
Temel İstatistik Hesaplamaları

- Merkez eğilim ölçüleri
  - Verinin ortası ya da orta noktasını ölçer (mod, medyan, ortalama)
- Dağılım ölçüleri
  - Verinin nasıl dağıldığını ölçer. (aralık, varyans, standart sapma)

Merkezi Eğilim Ölçüleri

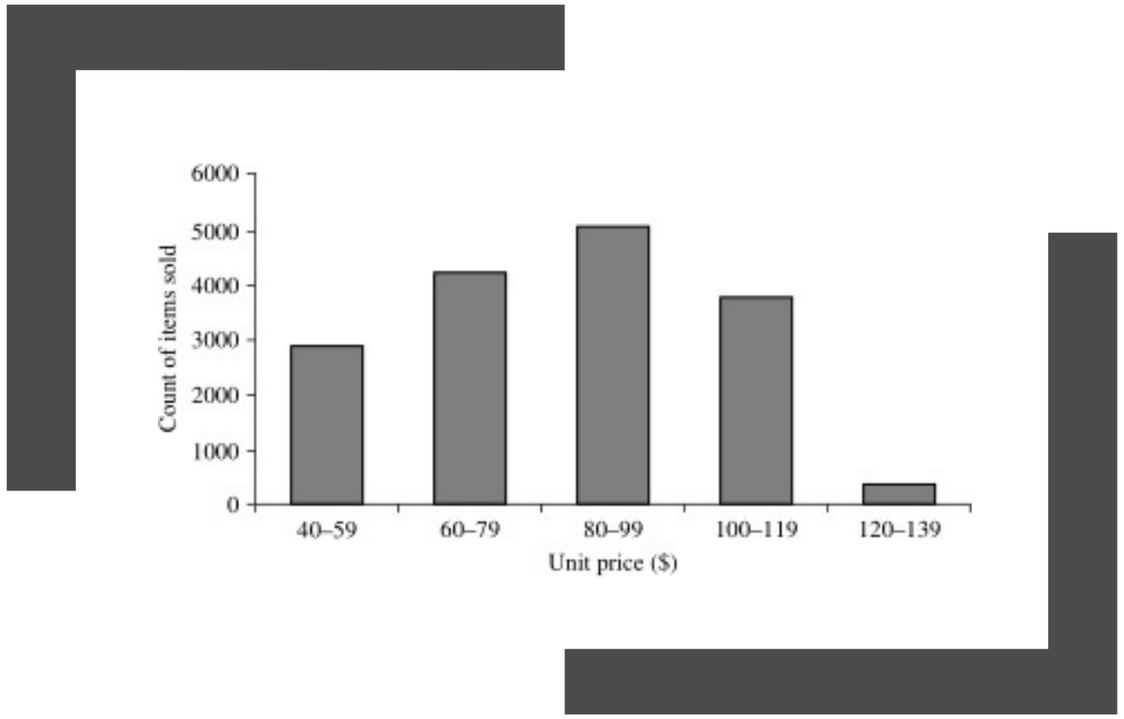
- **Ortalama**
- $\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} = \frac{x_1+x_2+\cdots+x_N}{N}$
- **Ağırlıklı Ortalama**
- $\bar{x} = \frac{\sum_{i=1}^{N} w_i x_i}{\sum_{i=1}^{N} w_i} = \frac{w_1x_1+w_2x_2+\cdots+w_Nx_N}{w_1+w_2+\cdots+w_N}$
- **Medyan:** Ortadaki Sayı
- **Mod:** En çok tekrar eden sayı

Dağılım Ölçüleri



- **Aralık(Range):** Min-Max arasında kalan değer
- **Kantiller(Quantiles):** Eşit aralığa bölünen değer
- **IQR(interquartile range)**
- $IQR = Q_3 - Q_1$

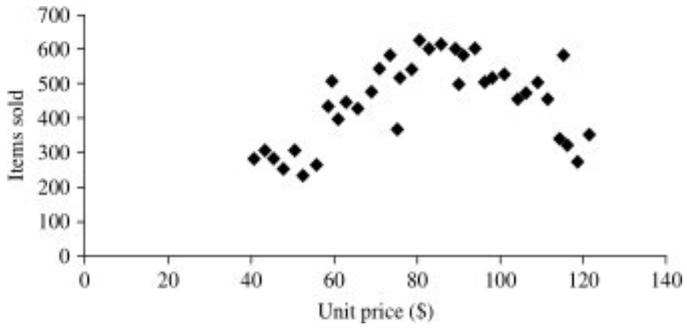
Histo-gram (Kutup Grafikleri)



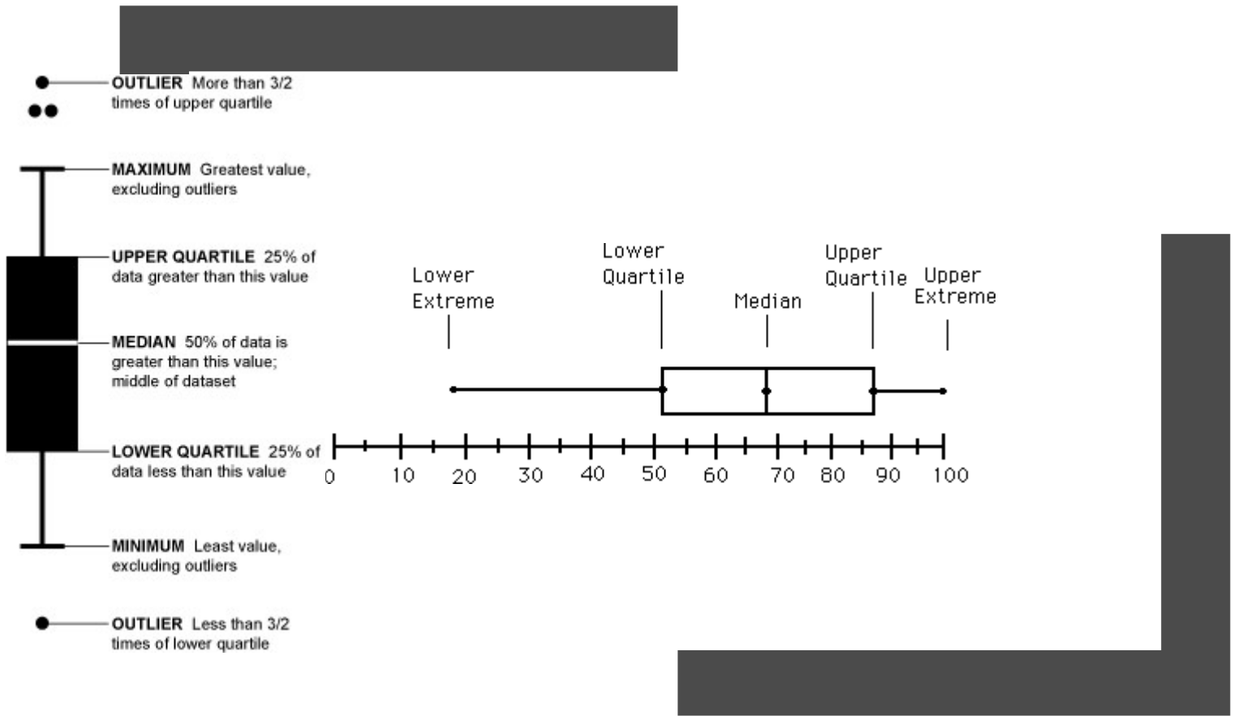
- Verinin dağılımı ile ilgili bilgi verir. Kutucuğun yüksekliği frekansı gösterir.
- Herbir kategoride ne kadar olay gerçekleştiğini gösterir.
- x ekseninde değerler ve y ekseninde değerin frekansı

Dağılım(Scatter) Grafiği

- Veriler arasında ilişki, trend ya da örüntü olup olmadığını kontrol için çizilir.
- Herbir ikili değer x ve y eksenindeki noktada işaretlenir.
- Noktalar ve aykırılıklar hakkında hızlı bir görüş açısı sağlar.
- Her bir nokta veri ve özellik hakkında bilgi verir.



Kutu Grafikleri (Box Plots)



- Kutu grafikleri aşğıdaki gibi özet bilgiler sağlar;
- 1. Kutu uzunluğu IQR değerini gösterir.
- 2. Medyan ortada işaretlenmiştir.
- 3. İki çizgi(whiskers) en küçük ve en büyük değerleri gösterir.
- 4. 1.5×IQR değerleri aykırı verilerdir.

Varyans, Kovaryans ve Standart Sapma

Medyan; tüm veri setindeki değerlerin küçükten büyüğe doğru sıralandığında en ortada yer alan değeri ifade etmektedir.

Mean; temelinde bir veri kümesindeki değerlerin toplamlarının veri kümesindeki değer sayısına bölünmesi ile elde edilir. İstatistikte kitle ortalaması ve örneklem ortalaması olarak iki ayrı şekilde ifade edilir.

Matematik **Fizik**

Mean :74.34 Mean :79.81
Median :75.00 Median :80.39

Standart sapma nedir? Verilerin (notların) aritmetik ortalamadan sapmalarının karelerinin aritmetik ortalamasının kare köküdür.

$$\sigma = \sqrt{\frac{\sum (X_i - \mu_x)^2}{n}}$$

σ: standart sapma

X_i: i inci öğrencinin notu

μ: ilgili dersin aritmetik ortalaması

n: öğrenci sayısı

Standart sapmanın genel ifadesi:

$$standart\ sapma\ = \sqrt{\frac{Notlarin\ ortalamadan\ farklarinin\ karelerinin\ toplami}{Ogrenci\ sayisi}}$$

	matematik	fizik
Ali	57.25465	72.31064
Ayşe	87.29831	70.30690
Aylin	64.53862	82.09290
Ahmet	92.98104	62.25533
Cemal	96.42804	78.74197
Muhittin	42.73339	79.35083
Beyza	71.68633	92.49610
Beril	93.54514	74.81282
Mehmet	73.08610	81.86234
Şaziye	67.39688	66.81048
Mehtap	97.41000	84.99986
Satılmış	67.20005	95.28662
Recep	80.65424	71.21415
Şaban	74.35800	75.93952
Melis	46.17548	90.50204
Buket	93.08050	86.76087

Showing 1 to 16 of 30 entries

$$standart\ sapma = \sqrt{\frac{(Ali - ortalama)^2 + (Ayse - ortalama)^2 + \dots.}{Ogrenci\ sayisi\ olan\ 30}}$$

$$standart\ sapma_{Matematik} = \sqrt{\frac{(57.25 - 74.34)^2 + (87.29 - 74.34)^2 + \dots.}{30}}$$

Matematik için standart sapma 17.48, fizik için 9.08

Standart sapmada kareyi her bir notun ortalamadan farkını bulduktan sonra farkını almamızın sebebi eksi değerleri düzeltmektir. Aslında notların aritmetik ortalamadan farklarının toplamı sıfırdır. Bunu önlemek için eksi değerleri artı yapacak kare alma işlemi yapılıyor.

Varyans nedir? Varyans, verilerin aritmetik ortalamadan sapmalarının karelerinin toplamıdır. Yani standart sapmanın karekök alınmamış halidir.

$$s^2 = \frac{\sum (X_i - \mu_x)^2}{n}$$

$$varyans = \frac{Bir\ ders\ icin\ her\ bir\ ogrenciye\ ait\ notun\ grup\ ortalamasindan\ farklarinin\ karelerinin\ toplami}{Ogrenci\ sayisi}$$

Peki biz niye durduk yerde standart sapma ve varyans gibi değerlerden bahsediyoruz. Ortalamalar bize bir seriyi temsil edebilecek değerlerdir. Yani bu sınıfın Matematik başarısı hakkında bir fikir edinmek istiyorsak ortalamaya bakarız. Örneğimizde 74.34, ha iyiymiş deriz. Peki ortalama tek başına bu sınıfın başarısı hakkında kanaat edinmemizi sağlayabilir mi? Hayır. Şöyle düşünelim aynı sınıftan başka bir şube olsun ve onun da ortalaması aynı olsun ancak bu sınıfın notları 30-40 ve 85-95 arasında olsun ve aralarda hiç not olmasın ancak ortalama 74.34 olsun. Şimdi bu iki sınıfın başarısı aynıdır diyebilir miyiz? Tabi ki hayır. İşte standart sapma ve varyans bu noktada ortalamaya ilave olarak bize sınıf başarısı hakkında kanaat edinmemizi sağlıyor. Bir sınıfta notlar ortalamaya yakın dağılmışken (standart sapma ve varyans düşük), diğer sınıfta ortalamadan çok uzaklara (standart sapma ve varyans büyük) dağılmış.

Kovaryans nedir? Kovaryans iki değişken arasındaki doğrusal ilişkinin değişkenliğini ölçen bir kavramdır. Başka bir tabirle, iki farklı serinin (örneğimizde seri matematik dersine ait 30 adet not ve fizik dersine ait 30 adet nottur, ya da tablo mantığı ile matematik ve fizik sütunlarını birer seri olarak düşünebiliriz) varyansıdır. **Yani iki serinin dağılımının benzerliğini analiz** ettiğimiz bir ölçüttür.

Kovaryans formülü:

$$\sigma_{xy} = \frac{1}{N}\sum_{i=1}^{N}((X_i - \mu_x)*(Y_i - \mu_y)$$

$$\sigma_{mat,fiz} = \frac{1}{30}((mat_{Ahmet} - ort_{mat})*(fiz_{Ahmet} - ort_{fiz}) + \ldots)$$

R kodu ile kovaryans hesaplayalım:

```
> cov(df\$matematik,df\$fizik)
[1] 16.52833
```

Kayıt Veriseti

- Bir gözleme ait verilerin tutulduğu verilerden oluşan veri seti.

No	Indirim	Medeni Hal	Gelir	Sipariş
1	Evet	Bekar	125K	**Hayır**
2	Hayır	Evli	100K	**Hayır**
3	Hayır	Bekar	70K	**Hayır**
4	Evet	Evli	120K	**Hayır**
5	Hayır	Dul	95K	**Evet**
6	Hayır	Evli	60K	**Hayır**
7	Evet	Dul	220K	**Hayır**
8	Hayır	Bekar	85K	**Evet**
9	Hayır	Evli	75K	**Hayır**
10	Hayır	Bekar	90K	**Evet**

VeriMatrisi

- Eğer veri objesi sabit sayıda nümerik verilerden oluşuyorsa bu veri setine veri matrisi denir ve nxm boyutuna sahiptir.

Projection of x Load	Projection of y load	Distance	Load	Thickness
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

Belge Veriseti

- Bu veri setinde her belgeye ait bir özellik vardır.

	team	coach	play	ball	score	game	win	lost	time	sea
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

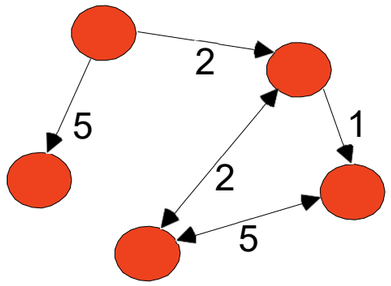
İşlemler Veriseti

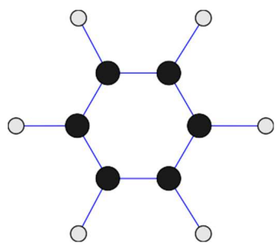
- Özel bir kayıt verisidir. Her kayıt birkaç elemandan oluşan setleri tutar. Örneğin bakkal dükkanının satılan ürünleri sıraladığı liste.

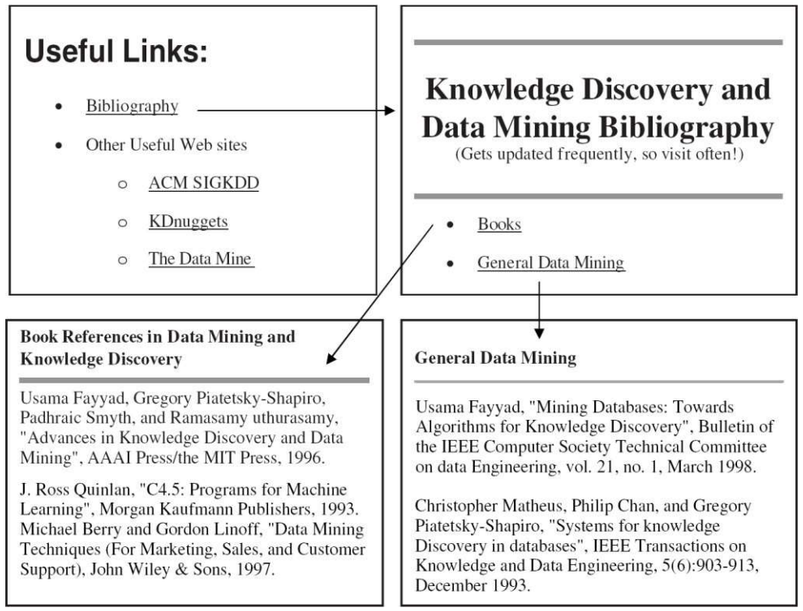
TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Grafik Veriseti

- Molekül yapısı ya da internet sayfalarının gösterilmesi







Benzerlik Ölçütleri

- Benzerlik ölçüsü
  - İki verinin benzeyip benzemediğinin nümerik ölçüsü
  - [0-1] Arasında bir değerle ölçülür ve 1'e yakın olması daha fazla benzediği anlamına gelir.
- Uzaklık (Benzemezlik) ölçüsü
  - İki verinin birbirinden ne kadar farklı olduğunun ölçülmesi durumu.
  - Eğer benzemez ise değer 0'a yakın çıkar.

Kümelemede Benzerlik Ölçütleri

- Küme bir koleksiyondaki verilerin birbirine benzemesi ve diğer koleksiyondaki verilere benzememesi demektir. Örneğin bir bakkal dükkanı müşterilerini kümelere ayırmak istediğinde müşterilerinin gelir durumları, yaşları gibi özelliklerine bakarak ayrım yapabilir. Bu özellikler arasındaki ilişki ve benzerlik durumu da kişilerin kümeye alınıp alınmayacağını belirler.

Veri ve Uzaklık Matrisi

- Veri Matrisi:

$$\begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix}$$

- Uzaklık Matrisi:

$$\begin{bmatrix} 0 & d(1,2) & \cdots & d(1,p) \\ d(2,1) & 0 & \cdots & d(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ d(n,1) & d(n,2) & \cdots & 0 \end{bmatrix}$$

- Benzerlik ölçüsü: $s(i,j) = 1 - d(i,j)$

Nominal Verilerin Uzaklık Hesabı

- $d(i,j) = \frac{p-m}{p}$
- m : toplam eşleşme sayısı
- p : nesneyi tanımlayan toplam özellik sayısı

No	Özellik-1	Özellik-2	Özellik-3
1	A	Çok İyi	45
2	B	Kötü	22
3	C	İyi	64
4	A	Çok iyi	28

Uzaklık Matrisi : $\begin{bmatrix} 0 & & & \\ d(2,1) & 0 & & \\ d(3,1) & d(3,2) & 0 & \\ d(4,1) & d(4,2) & d(4,3) & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ 1 & 1 & 0 & \\ 0 & 1 & 1 & 0 \end{bmatrix}$

Benzerlik : $s(i,j) = 1 - d(i,j) = \frac{m}{p}$

İkili Verilerin Uzaklık Ölçüsü

- 0 : Olmama Durumu Simetrik : 2 durum aynı
- 1 : Var Olma Durumu Asimetrik : 2 durum farklı
- $p = q + r + s + t$

		j nesnesi		
		1	0	Toplam
i nesnesi	1	q	r	q+r
	0	s	t	s+t
	toplam	q+s	r+t	p

İkili Verilerin Uzaklığını Hesaplama

- Simetrik Durum
- $d(i,j) = \frac{r+s}{q+r+s+t}$
- Asimetrik Durum
- $d(i,j) = \frac{r+s}{q+r+s}$
- Benzerlik Oranı (Asimetrik)
- $s(d,j) = 1 - d(i,j) = \frac{q}{q+r+s}$ → Jaccard Katsayısı

İkili Veriler için Örnek

- Aşağıdaki hastalık tablosunu ele alalım.

İsim	Cinsiyet	Öksürük	Ateş	Test1	Test2	Test3	Test4
Ali	E	E	H	P	N	N	N
Faruk	E	E	E	N	N	N	N
Fatma	K	E	H	P	N	P	N
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Hesaplamalar

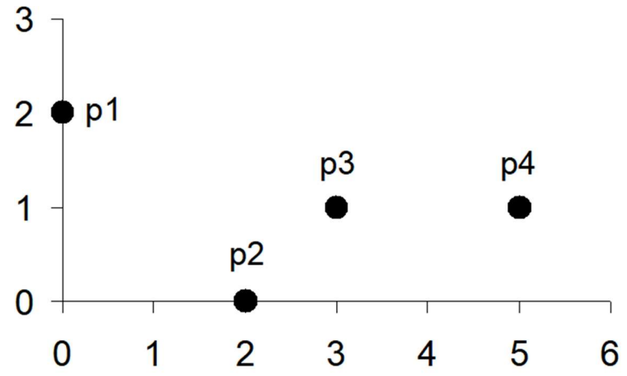
- $d\ (Ali, Faruk) = \frac{1+1}{1+1+1} = 0{,}67$
- $d\ (Ali, Fatma) = \frac{0+1}{2+0+1} = 0{,}33$
- $d\ (Faruk, Fatma) = \frac{1+2}{1+1+2} = 0{,}75$

- Bu ölçümler Faruk ve Fatma'nın birbirinden daha uzak olduğunu en yakın ikilinin Ali ve Fatma olduğunu göstermektedir.

Nümerik Verilerin Uzaklığı

- Öklid: 2 nokta arasındaki uzaklık.
- $d(i,j) = \sqrt{\left(x_{i1} - x_{j1}\right)^2 + \left(x_{i2} - x_{j2}\right)^2 + \cdots + \left(x_{ip} - x_{jp}\right)^2}$
- Eğer farklı ölçeklerde ise standardizasyon gereklidir.
- p boyut sayısı,i $= x_{i1}, x_{i2} \ldots x_{ip}$ve j $= x_{j1}, x_{j2} \ldots x_{jp}$

Öklid Uzaklığı Hesaplama



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

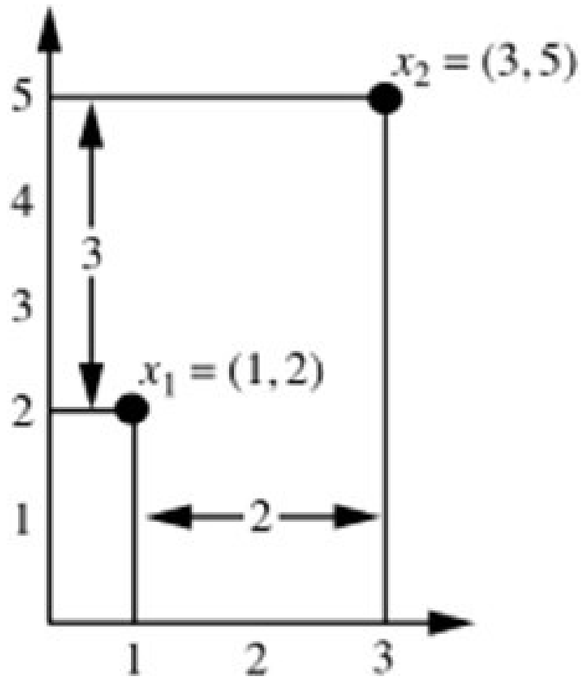
	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Manhattan Uzaklık Hesabı

- $d(i,j) = \sigma_{k=1}^{n} |i_k - y_k|$
- Özellikleri
- 1. Negatif olmama koşulu : $d(i,j) \geq 0$
- 2. Aynı şehirde olma : $d(i,i) = 0$
- 3. Simetrik olma koşulu : $d(i,j) = d(j,i)$
- 4. Üçgen eşitsizliği : $d(i,j) \leq d(i,k) + d(k,j)$
- Eğer bu özellikler sağlanırsa ölçüt **metriktir.**

Manhattan Uzaklığı Örnek

- Örneğin $x_1 = \ (1,2)$ ve $x_2 = (3,5)$ olsun.
- Öklid uzaklığı : $\sqrt{2^2 + 3^2} = 3{,}61$
- Manhattan uzaklığı : $2 + 3 = 5$



Ordinal Veriler için Uzaklık Hesabı

- M → ordinal listenin alabileceği değerler
- Adımlar
- Özellikler arasında {1,2,…M} şeklinde sıralama yap.
- Normalizasyon yap.
- Nümerik veriler için yapılan hesaplamaları burada da yap.

Ordinal veriler için örnek

- M=3 {Çok iyi: 3, İyi: 2, Kötü: 1}
- Atamayı yaparsak özellik-2 → {3,1,2,3} olur.
- Normalizasyon yaparsak [0-1] Çok İyi: 1, İyi: 0.5, Kötü: 0 olur.

No	Özellik-1	Özellik-2	Özellik-3
1	A	Çok İyi	45
2	B	Kötü	22
3	C	İyi	64
4	A	Çok iyi	28

Ordinal Veriler Örnek

- Sonuç Matrisi→ $\begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ 0{,}5 & 0{,}5 & 0 & \\ 0 & 1 & 0{,}5 & 0 \end{bmatrix}$
- $d(1{,}2)\ ve\ d(4{,}2) = 1$ olduğu için en uzak veriler 1-2 ve 4-2 sonucu çıkarılır.
- Benzerlik ölçüsü için $s(\mathrm{i}, \mathrm{j}) = 1 - d(i, j)$ formülü kullanılır.

Kosinüs Benzerliği

	Takım	Koç	Futbol	Oyun	Penaltı	Gol	Sezon
Doküman1	5	3	0	0	3	2	0
Doküman2	3	0	2	1	0	1	1
Doküman3	0	7	0	0	1	3	0

$$s(x,y) = \frac{x.y}{\text{Ъ}x\|y\|}$$

$\text{Ъ}x\|y\|$: x ve y vektörlerinin Öklid formu

$$\boldsymbol{x}^t \cdot \boldsymbol{y} = 5\times 3+0\times 0+3\times 2+0\times 0+2\times 1+0\times 1+0\times 0+2\times 1 +0\times 0+0\times 1=25$$

$$||\boldsymbol{x}|| = \sqrt{5^2+0^2+3^2+0^2+2^2+0^2+0^2+2^2+0^2+0^2} = 6.48$$

$$||\boldsymbol{y}|| = \sqrt{3^2+0^2+2^2+0^2+1^2+1^2+0^2+1^2+0^2+1^2} = 4.12$$

$$sim(\boldsymbol{x},\boldsymbol{y}) = 0.94$$

Veri Önişleme

- Birleştirme
- Örneklem Alma
- Özellik İndirgeme
- Kesikli ya da İkili Hale Getirme
- Özellik Dönüşümleri
- Veri Görselleştirme

Veri Kalitesi

- Veri Kalitesini Etkileyen Faktörler
    - Doğruluk
    - Tam olma durumu
    - Tutarlılık
    - Güvenirlik
    - Tahmin edilebilirlik

Veri Kalitesi

- Zayıf veri kümesi verilerin analizi aşamasında sorunlar doğurabilir.
    - Gürültü
    - Eksik Veriler
    - Tekrarlı Veriler
    - Yanlış Veriler

Eksik Veriler

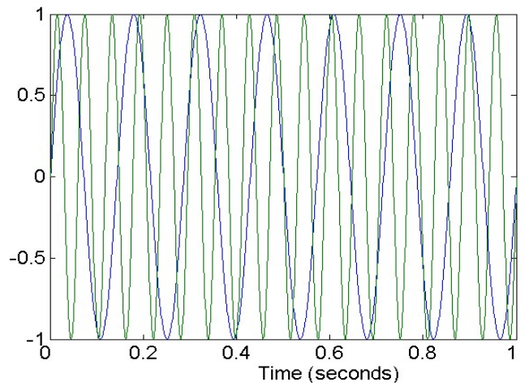
- Tamamen rastgele eksik olan veriler
  - Tüm değişkenlere ait verilerden herhangi birisinde eksik veri yer alabilir.
  - Bu veriler rastgele bir değer ile doldurulabilir.

- Diğer Doldurma Metotları
  - Ortalama ile doldurma
  - Mod, Medyan ile doldurma
  - Akıllı bir algoritma kullanarak doldurma

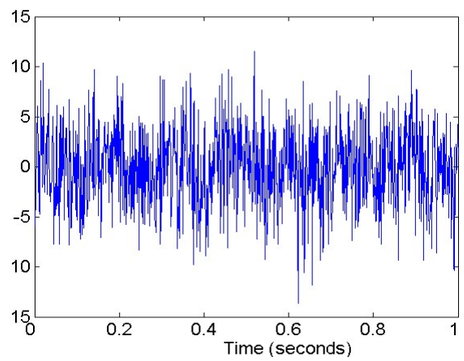
- 1. Eksik verileri gözardı et.
- 2. Elle doldur.
- 3. Global bir değişken kullan.
- 4. Merkezi bir ölçü kullanarak değiştir.
- 5. Aynı sınıfa ait merkezi bir ölçü ile değiştir.
- 6. Olası bir değer ile değiştir

Gürültülü Veri

- Uç verilerin veri setine girmesi. Örneğin ses kalitesi düşük olan bir telefondan alınan insan sesi frekansları.



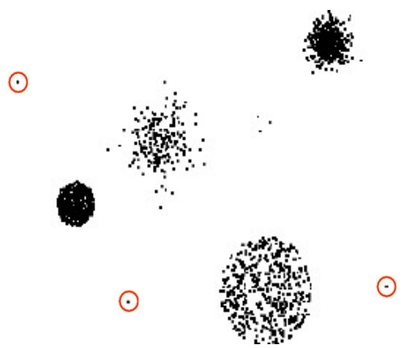
Two Sine Waves



Two Sine Waves + Noise

Aykırı (Uç) Veriler

- Veri setinin karakteristiğini yansıtmayan veriler.
- 1.5×IQR



Tekrarlı Veriler

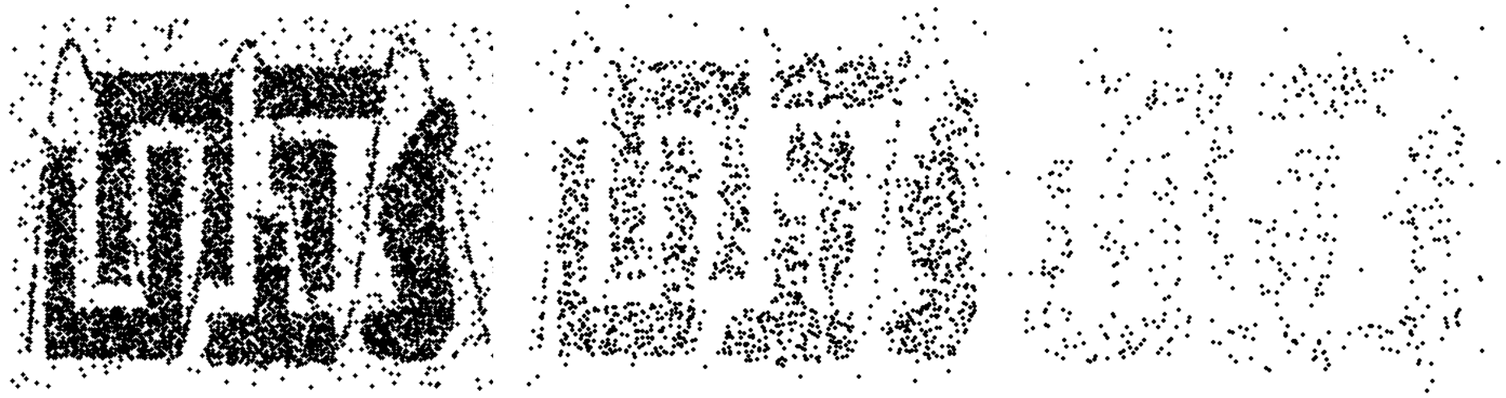
- Veri setinde aynı olarak geçen birden fazla verinin olması durumu. Örneğin bir kişinin birden fazla mail adresi girmesi.
- Bu tip verilerin temizlenmesi gereklidir.

Birleştirme

- 2 veya daha fazla özelliği bir araya getirme.
- Örneğin gün ve saat olarak verilen özelliklerin tek özellik olan tarih özelliğinde birleştirilmesi
- Ölçek değiştirilmesi
    - Şehirlerin bölgede birleştirilmesi
    - Günlerin ay olarak birleştirilmesi
- Daha stabil veri
    - Birleştirme sayesinde daha az değişken veriler elde edilir.

Örneklem Alma

- Veri azaltımı için gerekli olabilir.
    - Analiz öncesinde veya sonrasında yapılabilir.
- İstatistikçiler evrensel kümeye ulaşmak imkansız olduğundan örneklem üzerinden işlem yaparlar.
- Veri madenciler ise evrensel küme üzerinde işlem yapmak çok yorucu ve maliyetli olduğundan dolayı örneklem alırlar.
- Örneklemin evrensel kümeyi temsil etmesi gerekir. Yani evrensel küme içerisinde yer alan tüm özellikleri barındırması gerekir.



Örneklem çeşitleri

Herhangi bir elemanın seçilmesinin eşit olasılıklı olması

Seçerek ayırma

Rastgele seçilen verilerin örneklemden çıkarılması

Verileri küçük parçalara bölmek

Özellik İndirgemesi

- Amaç
    - Fazla özelliklerden kurtulmak
    - Gereksiz zaman kaybının önüne geçmek
    - Verilerin daha iyi görselleştirilmesini sağlamak
    - Gereksiz özellik ve gürültülerin elenmesini sağlamak
- Teknikler
    - Birincil Etken Analizi (PCA=Principle Component Analysis)

Özellik İndirgeme

- Gereksiz özellikler
    - Birbirine çok benzeyen iki verinin elenmesi
    - Örneğin araba satış fiyatı ile araba vergi tutarı
- Alakasız veriler
    - Veri madenciliği için kullanılmayacak olan özellikler.
    - Örneğin öğrenci numarası

Kesikli Hale Getirme

- Sürekli verilerin kesikli veriler haline dönüştürülmesi işlemi
    - Sürekli veriler için bir takım kategoriler oluşturarak kesikli veriler kullanılabilir.
    - Özellikle sınıflandırma görevinde kullanılmalıdır.
    - Birçok sınıflandırma algoritması kesikli veriler ile daha iyi sonuçlar vermektedir.

İkili Sisteme Çevirme

- Sürekli ya da kategorik değişkenin ikili(0-1) sistemine dönüştürülmesi işlemidir.
- Genellikle birliktelik kuralları için gerçekleştirilir.
- Sürekli veriler önce kategorik verilere çevrilir ardından ikili değerlere çevrilir.
- Örneğin uzunluğun kısa, orta, uzun diye kategorize edilmesi.

Özellik Dönüşümleri

- Normalizasyon
    - Özellik verileri arasındaki farklılıkları azaltarak bir aralık içerisine alma işlemi
    - Sezonsal etki gibi istenmeyen özellikler ortadan kalkar.
- Standardizasyon
    - Verinin ortalamadan çıkarılarak standart sapmasına bölünmesi ile elde edilen sonuçlardır.

Veri Önişleme Uygulamaları Kod Örnekleri

```
import warnings
warnings.filterwarnings("ignore")

import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.datasets import load_iris
from sklearn.preprocessing import KBinsDiscretizer, Binarizer, MinMaxScaler, StandardScaler
from sklearn.decomposition import PCA
```

Veri Setinin Yüklenmesi

Iris veri seti, çiçek yaprak uzunlukları ve genişliklerini (4 adet sayısal özellik) ve bu çiçeklerin türlerini (Setosa, Versicolour, Virginica) içeren bir veri setidir.

```
iris = load_iris(as_frame=True)
df_iris = iris.frame  # iris verisini pandas DataFrame olarak alır
df_iris.head()
```

- **sepal length (cm)**, **sepal width (cm)**, **petal length (cm)**, **petal width (cm)** sütunları sayısal verileri,
- **target** sütunu ise sınıf etiketlerini (0,1,2) tutar.

Birleştirme (Merging/Joining)

Veri bilimi projelerinde farklı kaynaklardan gelen veri tablolarını birleştirme ihtiyacı sıkça ortaya çıkar. Örneğin, elinizde bir tablodaki özelliklerle (feature) başka bir tablodaki hedef değerler (label) ya da yan özellikler farklı tablolarda tutuluyor olabilir.

Örnek Senaryo: Elimizde `df_iris_features` ve `df_iris_target` adında iki farklı tablo olsun ve bunları birleştirelim.

```
# Özellik veri çerçevesi
```

```
df_iris_features = df_iris.drop("target", axis=1).copy()
df_iris_features["id"] = range(len(df_iris_features))  # id sütunu ekleyelim

# Hedef (target) veri çerçevesi
df_iris_target = df_iris[["target"]].copy()
df_iris_target["id"] = range(len(df_iris_target))

# İki tabloyu 'id' sütunu üzerinden birleştirelim
df_merged = pd.merge(df_iris_features, df_iris_target, on="id")
df_merged.head()
```

Bu örnekte, `id` sütununu ekleyerek iki tabloyu **Inner Join** mantığıyla birleştirdik. Gerçek projelerde ise birincil anahtar (primary key) olan sütunlar üzerinden birleştirme yapılabilir.

Örneklem Alma (Sampling)

Büyük veri setlerinden belirli bir oranda örnek almak gerekebilir (performans testleri, prototip modelleme vb. için).

```
# Veri setinden %30 oranında örnek alalım
df_sampled = df_merged.sample(frac=0.3, random_state=42)
print("Orijinal veri boyutu:", df_merged.shape)
print("Örneklem (sample) boyutu:", df_sampled.shape)
df_sampled.head()
```

- **`frac=0.3`**: Yüzde 30’luk bir kısım çekilmesi.
- **`random_state=42`**: Rastgelelik tekrar üretilebilir olması için sabit tohum.

Özellik İndirgeme (Dimensionality Reduction)

Örnek olarak **PCA (Principal Component Analysis)** yöntemini kullanalım. PCA, yüksek boyutlu verilerdeki varyansı en iyi açıklayan temel bileşenleri (principal components) bulur ve boyutu küçültür.

PCA ile 2 Bileşene İndirgeme

```
# Hedef sütununu (target) ayıralım
X = df_merged.drop(["id", "target"], axis=1)
y = df_merged["target"]

# PCA modelini oluşturalım (2 bileşene düşüreceğiz)
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)

print("Orijinal özellik boyutu:", X.shape[1])
print("Yeni PCA özellik boyutu:", X_pca.shape[1])
```

```
# PCA sonrası verileri bir DataFrame'e koyup görselleştirelim
df_pca = pd.DataFrame(X_pca, columns=["PC1", "PC2"])
df_pca["target"] = y

plt.figure(figsize=(6,5))
sns.scatterplot(x="PC1", y="PC2", hue="target", data=df_pca, palette="Set2")
plt.title("Iris Veri Seti (PCA Sonrası 2 Bileşen)")
plt.show()
```

Bu sayede 4 boyutlu Iris verisini 2 boyuta indirgemiş olduk ve grafik üzerinde farklı çiçek türlerinin nasıl kümelendiğini gözlemleyebiliyoruz.

Kesikli ya da İkili Hale Getirme (Binarization & Discretization)

Bazı modellerde **kategorik** veya **kesikli** verilere ihtiyaç olabilir. Örneğin, sürekli (continuous) bir özelliği belirli aralıklara bölmek (discretization) ya da ikili (binary) biçime dönüştürmek isteyebiliriz.

6.1 Kesikli Hale Getirme (Discretization)

KBinsDiscretizer kullanarak, örneğin `sepal length (cm)` sütununu **3** eşit aralığa bölelim:

```
disc = KBinsDiscretizer(n_bins=3, encode='ordinal', strategy='uniform')
sepal_length = df_merged["sepal length (cm)"].values.reshape(-1, 1)

sepal_length_binned = disc.fit_transform(sepal_length)
df_merged["sepal_length_binned"] = sepal_length_binned.astype(int)

df_merged[["sepal length (cm)", "sepal_length_binned"]].head(10)
```

- **`n_bins=3`**: 3 kesik aralığı,
- **`encode='ordinal'`**: aralıkları 0,1,2 gibi ordinal sayılara dönüştürür,
- **`strategy='uniform'`**: verileri aralıklara eşit genişlikte böler.

İkili Hale Getirme (Binarization)

Binarizer, belirli bir eşik değerinin üzerindeki değerleri 1, altındakileri 0 yapar. Örneğin, `petal length (cm)` için 2.0 eşik değeri kullanalım:

```
binarizer = Binarizer(threshold=2.0)
petal_length = df_merged["petal length (cm)"].values.reshape(-1, 1)

petal_length_binary = binarizer.fit_transform(petal_length)
df_merged["petal_length_binary"] = petal_length_binary.astype(int)

df_merged[["petal length (cm)", "petal_length_binary"]].head(10)
```

Bu sayede `petal_length_binary` sütunu ya 0 (2.0 cm'den küçük değerler) ya da 1 (2.0 cm'den büyük veya eşit değerler) olacak şekilde düzenlenir.

Özellik Dönüşümleri (Feature Transformations)

Özellik dönüşümleri, veriyi belli bir dağılıma yaklaştırmak veya ölçüm aralığını değiştirmek için yapılır. En yaygınları **MinMaxScaling, StandardScaling** (Z-Skoru), **Log** veya **Power** dönüşümleridir.

Min-Max Ölçeklendirme

[0, 1] aralığına dönüştürür:

```
minmax_scaler = MinMaxScaler()
X_minmax = minmax_scaler.fit_transform(X)

print("MinMax ile ölçeklendirilmiş ilk 5 satır:")
print(X_minmax[:5])
```

Standardizasyon (Z-Skoru)

Ortalaması 0, standart sapması 1 olacak şekilde dönüştürür:

```
std_scaler = StandardScaler()
X_std = std_scaler.fit_transform(X)

print("StandardScaler ile ölçeklendirilmiş ilk 5 satır:")
print(X_std[:5])
```

Bu yöntem, örneğin lineer regresyon veya lojistik regresyon gibi modellerin performansını arttırabilir.

---

8. Veri Görselleştirme

Veri görselleştirme, verinin içindeki desenleri ve aykırı değerleri anlamamızı kolaylaştırır. **Seaborn** ve **Matplotlib** bu konuda oldukça sık kullanılır.

Dağılım Grafiği (Pairplot)

Iris veri setinde sayısal değişkenlerin dağılımını ve aralarındaki ilişkileri göstermek için **pairplot** sıklıkla kullanılır:

```
sns.pairplot(df_iris, hue="target", height=2, diag_kind="hist", palette="Set2")
plt.show()
```

- **`hue="target"`** ile farklı çiçek türlerini farklı renklerde görebiliriz.
- **`diag_kind="hist"`** ile diyagonal eksende histogram çizdirir.

Kutu Grafiği (Box Plot)

Sayısal sütunlardaki olası aykırı değerleri hızlıca görmek için:

```
plt.figure(figsize=(8, 5))
sns.boxplot(data=df_iris.drop("target", axis=1), palette="Set2")
plt.title("Iris Özellikleri için Kutu Grafiği")
plt.show()
```