Vektörler (temel kavramlar)

Dr. Öğr. Üyesi Işık İlber Sırmatel

T.C. Trakya Üniversitesi
Mühendislik Fakültesi
Elektrik - Elektronik Mühendisliği Bölümü
Kontrol Anabilim Dalı

Kaynak (source)

Lecture Slides for Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares.
Stephen Boyd, Lieven Vandenberghe

Konu listesi

1. Tanım ve notasyon
2. Örnekler
3. Toplama ve skaler çarpım
4. İç çarpım
5. Karmaşıklık

Bölüm 1

Tanım ve notasyon

Vektörler

- **vektör** sıralı bir sayı listesidir
- bir vektör

$$\begin{bmatrix} -1.1 \\ 0.0 \\ 3.6 \\ -7.2 \end{bmatrix} \quad \text{veya} \quad \begin{pmatrix} -1.1 \\ 0.0 \\ 3.6 \\ -7.2 \end{pmatrix}$$

  veya $\left(-1.1, 0.0, 3.6, -7.2\right)$ şeklinde yazılır
- listedeki sayılara vektörün **eleman**ları denir
- elemanların sayısına vektörün **boyut**u denir
- yukarıdaki vektörün boyutu 4; üçüncü elemanı 3.6
- n boyutlu bir vektöre n-**vektör** denir
- sayılara **skaler** denir

Sembollerle vektörler

- vektörleri belirtmek için semboller kullanılır, örneğin: a, X, p, β, E^{aut}
- diğer gösterimler: $\mathbf{g}$, $\vec{a}$
- a isimli bir n-vektörün i. elemanı a_i olarak gösterilir
- a_i'deki i'ye **indis** denir
- bir n-vektörün indisleri $i = 1$'den $i = n$'e kadar gider
- **uyarı:** bazen a_i bir vektör listesindeki i. vektörü belirtir
- a ve b isimli aynı boyutlu iki vektör bütün i'ler için $a_i = b_i$ ise **eşit**tir. bu durum, $=$ işaretine ek anlam yüklenerek $a = b$ şeklinde yazılır

Blok vektörler

- b, c ve d (boyutları m, n ve p) vektörlerini ele alalım
- **istiflenmiş vektör** veya b, c ve d'nin **zincirlenmesi** şu şekildedir:

$$a = \begin{bmatrix} b \\ c \\ d \end{bmatrix}$$

- buna **blok vektör** de denir (blok elemanları b, c ve d)
- a'nın boyutu $m + n + p$'dir:

$$a = \begin{pmatrix} b_1 & b_2 & \dots & b_m & c_1 & c_2 & \dots c_n & d_1 & d_2 & \dots d_p \end{pmatrix}$$

Sıfır, birler ve birim vektörler

- bütün elemanları 0 olan n-vektör 0_n veya 0 ile gösterilir
- bütün elemanları 1 olan n-vektör $\mathbf{1}_n$ veya $\mathbf{1}$ ile gösterilir
- bir elemanı 1 ve diğer elemanları 0 olan vektöre **birim vektör** denir
- i. elemanı 1 olan birim vektör e_i ile gösterilir
- 3 boyutlu birim vektörler:

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Seyreklik

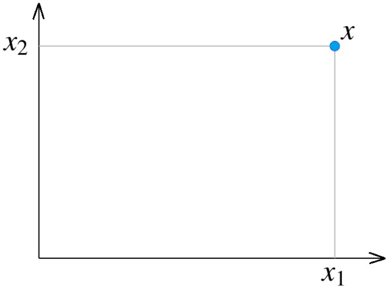
- elemanlarının çoğu 0 olan vektöre **seyrek** denir
- bir bilgisayarda verimli şekilde saklanabilir ve işlenebilirler
- x isimli bir vektörün sıfır olmayan elemanlarının sayısı $\mathbf{nnz}(x)$ ile gösterilir
- örnekler: sıfır vektörler, birim vektörler

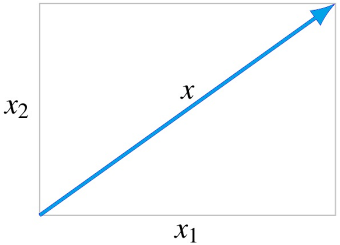
Bölüm 2

Örnekler

2 boyutta konum veya yerdeğiştirme

(x_1, x_2) şeklindeki bir 2-vektör 2 boyutta konumu veya yerdeğiştirmeyi temsil edebilir





Diğer örnekler

- renk: $(\mathrm{R}, \mathrm{G}, \mathrm{B})$
- n farklı hammadde veya kaynağın miktarı (örneğin, malzeme listesi)
- portföy: elemanlar n farklı varlığın her birindeki hisseyi ($\$$ cinsinden değer veya kesir olarak) belirtir (negatif elemanlar açık pozisyonu ifade eder)
- nakit akışı: x_i i. dönemde alınan ödeme
- ses: x_i i. örnekleme zamanındaki akustik basınç
- öznitelikler: x_i bir varlığın i. öznitelik veya özelliğinin değeri
- müşteri alımı: x_i bir müşterinin ürün i için bir dönemde yaptığı satınalmanın toplam değeri
- sözcük sayısı: x_i sözcük i'nin bir belgede geçme sayısı

Sözcük sayısı vektörleri

- kısa bir belge:

> **Bilgisayar** tabanlı **belge** analizinde **sözcük sayısı vektör**leri kullanılır. **Sözcük sayısı vektör**ünün her elemanı, ilgili **sözcüğün belge**de kaç defa geçtiğini gösterir.

- küçük bir sözlük ve sözcük sayısı vektörü

$$\begin{array}{c} \text{bilgisayar} \\ \text{belge} \\ \text{sözcük} \\ \text{kalem} \\ \text{sayı} \\ \text{vektör} \end{array} \qquad \begin{bmatrix} 1 \\ 2 \\ 3 \\ 0 \\ 2 \\ 2 \end{bmatrix}$$

- uygulamadaki sözlükler çok daha büyüktür

Bölüm 3

Toplama ve skaler çarpım

Vektör toplama

- n-vektörler a ve b toplanabilir; toplam $a + b$ ile gösterilir
- toplamı hesaplamak için elemanlar toplanır:

$$\begin{bmatrix} 0 \\ 7 \\ 3 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 \\ 9 \\ 3 \end{bmatrix}$$

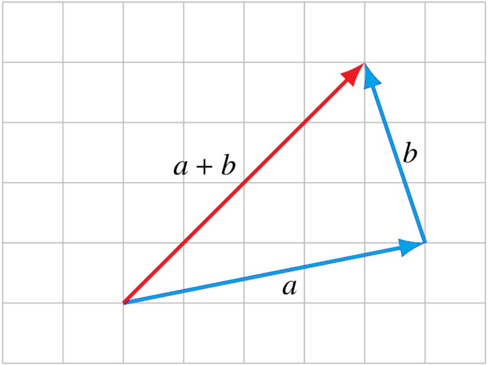
- çıkarma benzer şekilde yapılır

Vektör toplamanın özellikleri

- değişmeli: $a + b = b + a$
- birleşmeli: $(a + b) + c = a + (b + c)$ (dolayısıyla iki taraf da $a + b + c$ olarak yazılabilir)
- $a + 0 = 0 + a = a$ (0 etkisiz eleman)
- $a - a = 0$

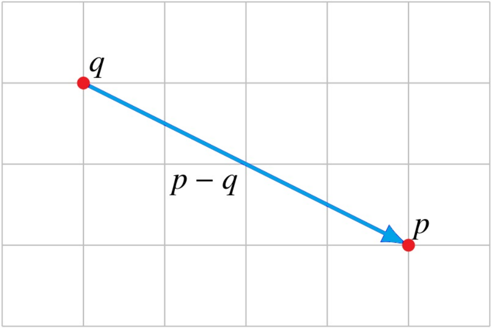
Yerdeğiştirmelerin toplanması

2-vektörler a ve b yerdeğiştirmeleri ifade ediyorsa, $a + b$ toplam yerdeğiştirmeyi verir



Bir noktadan diğerine yerdeğiştirme

q noktasından p noktasına olan yerdeğiştirme $p - q$ ile verilir



Skaler-vektör çarpım

- skaler β ve n-vektör a çarpılabilir

$$\beta a = \left(\beta a_1,\, \beta a_2,\, \ldots,\, \beta a_n\right)$$

- $a\beta$ olarak da gösterilebilir
- örnek

$$-2\begin{bmatrix}1\\9\\6\end{bmatrix} = \begin{bmatrix}-2\\-18\\-12\end{bmatrix}$$

Skaler-vektör çarpımın özellikleri

- birleşmeli: $(\beta\gamma)a = \beta(\gamma a)$
- soldan dağılmalı: $(\beta + \gamma)a = \beta a + \gamma a$
- sağdan dağılmalı: $\beta(a + b) = \beta a + \beta b$
- not: β ve γ skaler, a ve b vektör

bu özellikler basit görünebilirler ancak bunları kusursuz olarak anladığınızdan emin olun

Doğrusal bileşimler

- vektörler $a_1, \ldots, a_m$ ve skalerler $\beta_1, \ldots, \beta_m$ için

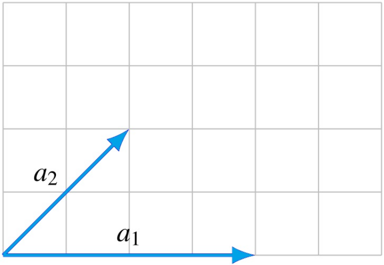
$$\beta_1 a_1 + \cdots + \beta_m a_m$$

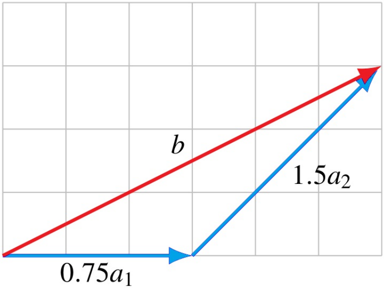
  ifadesi vektörlerin bir **doğrusal bileşim**idir
- $\beta_1, \ldots, \beta_m$ **katsayı**lardır
- doğrusal bileşim **çok** önemli bir kavramdır
- basit bir özdeşlik: herhangi bir n-vektör b için

$$b = b_1 e_1 + b_2 e_2 + \cdots + b_n e_n$$

Doğrusal bileşimler

örnek: a_1 ve a_2 olarak verilen iki vektör ve bunların $b = 0.75a_1 + 1.5a_2$ şeklindeki doğrusal bileşimi





Bölüm 4

İç çarpım

İç çarpım

- n-vektörler a ve b'nin iç çarpımı

$$a^T b = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

  şeklindedir
- iç çarpım için şu notasyonlar da kullanılır: $\langle a, b \rangle$, $\langle a | b \rangle$, (a, b), $a \cdot b$
- örnek:

$$\begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}^T \begin{bmatrix} 1 \\ 0 \\ -3 \end{bmatrix} = (-1)(1) + (2)(0) + (2)(-3) = -7$$

İç çarpımın özellikleri

- $a^T b = b^T a$
- $(\gamma a)^T b = \gamma(a^T b)$ (γ skaler)
- $(a+b)^T c = a^T c + b^T c$

bu özellikler birleştirilerek örneğin

$$(a+b)^T(c+d) = a^T c + a^T d + b^T c + b^T d$$

şeklinde özellikler türetilebilir

Genel örnekler

- $e_i^T a = a_i$ (a'nın 1. elemanını seçer)
- $\mathbf{1}^T a = a_1 + \ldots + a_n$ (a'nın elemanlarının toplamı)
- $a^T a = a_1^2 + \ldots + a_n^2$ (a'nın elemanlarının karelerinin toplamı)

Örnekler

- w ağırlık vektörü, f öznitelik vektörü; $w^T f$ skor
- p fiyat vektörü, q nicelik vektörü; $p^T q$ toplam maliyet
- s varlık hisse miktarları, p varlık fiyatları; $p^T s$ portföyün toplam değeri

Bölüm 5

Karmaşıklık

Flop sayısı

- bilgisayarlar (reel) sayıları *kayan virgüllü sayı* (floating-point number) formatında saklarlar
- bu formattaki temel aritmetik işlemlere (toplama, çarpım, ...) *kayan virgüllü işlem* (floating-point operation veya flop) denir
- bir algoritmanın veya işlemin karmaşıklığı (giriş boyutunun fonksiyonu olarak) gereken toplam flop sayısı ile ölçülür
- karmaşıklık kabaca yaklaşık olarak hesaplanabilir
- algoritma/işlem yürütme (execution) süresinin kabaca yaklaşık hesabı: gereken flop sayısı/bilgisayar hızı
- günümüzdeki bilgisayarların hızı 1 Gflop/saniye (10^9 flop/saniye) civarındadır
- ancak bu 100 kata kadar değişkenlik gösterebilir

Vektör toplama ve iç çarpımın karmaşıklığı

- $x + y$ işlemi için n adet toplama işlemi gerekir, dolayısıyla bu işlemin karmaşıklığı n floptur
- $x^T y$ işlemi için n adet çarpım ve $n - 1$ adet toplama işlemi gerekir, dolayısıyla bu işlemin karmaşıklığı $2n - 1$ floptur
- $x^T y$ için bunu $2n$ (hatta n) olarak alıp basitleştiririz
- x veya y seyrek ise karmaşıklık çok daha az olur