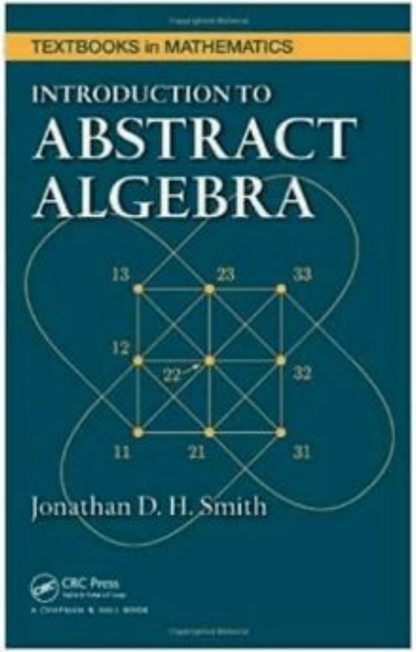
Вектори - 2 част

okay atheists

if there's no God, then explain this



and plz hurry, I have a test on it tomorrow

Защо векторите и линейните трансформации са важни за програмиста?

- Огромно значение за разработката на компютърни игри
  - векторите се използват за представяне на движения
- Огромно значение за компютърната графика и анимациите
  - Не искате да завъртите врата си, за да видите някоя снимка, нали?
- Векторите и матриците, както и операциите с тях са от изключителна важност за изкуствения интелект и машинното обучение:
  - https://machinelearningmastery.com/why-learn-linear-algebra-for-machine-learning/

Векторно (линейно) пространство

- Съвкупност от обекти (вектори), които могат да бъдат събирани и умножавани.
- Важат общо осем аксиоми, свързани с векторното пространство.
- Множеството от свободните вектори в равнината е векторно пространство.
- Векторното пространство е формирано над поле (обикновено свързано с множеството на реалните числа)

Аксиоми за векторно (линейно) пространство

- Събиране:
  - $(a + b) + c = a + (b + c)$
  - $a + 0 = a$
  - $a + (-a) = 0$
  - $a + b = b + a$
- Умножение:
  - $1a = a$
  - $(\alpha + \beta)a = \alpha a + \beta a$
  - $\alpha(a + b) = \alpha a + \alpha b$
  - $\alpha(\beta a) = (\alpha\beta)a$

Примери за векторно пространство

- Координатното пространство, например на реалните числа $\mathbb{R}^n$
    - **n**-мерни вектори (с **n** на брой координати)
- Безкрайно координатно пространство $\mathbb{R}^\infty$
    - Всеки вектор има безброй много координати
- Полиномно пространство
    - Всички полиноми на променлива **x**, които имат реални коефициенти

Линейна комбинация

- Вектори $v_1, v_2, \dots v_n$
- Числа (скалари) $\lambda_1, \lambda_2, \dots, \lambda_n$
- Линейна комбинация
    - Сумата от всеки вектор умножен по съответния скаларен коефициент

$$\lambda_1 v_1 + \lambda_2 v_2 + \cdots + \lambda_n v_n = \sum \lambda_i v_i$$

Линейно зависими и независими вектори

- Векторите $v_1, v_2, \ldots, v_n$ се наричат линейно зависими, ако съществуват числа $\lambda_1, \lambda_2, \ldots, \lambda_n$, такива че поне едно от тях е различно от 0, така че:

$$\lambda_1 \overrightarrow{\nu_1} + \lambda_2 \overrightarrow{\nu_2} + \ldots + \lambda_n \overrightarrow{\nu_n} = \vec{0}$$

- В противен случай векторите се наричат линейно независими.

Базисни вектори

- Нека да разгледаме следните векто $e_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} e_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
- А сега нека да разгледаме векто $a = \begin{bmatrix} -3 \\ 2 \end{bmatrix}$
- Можем да представим вектора като линейна комбинация с другите два вектор $a = -3e_1 + 2e_2$

$$a = -3 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

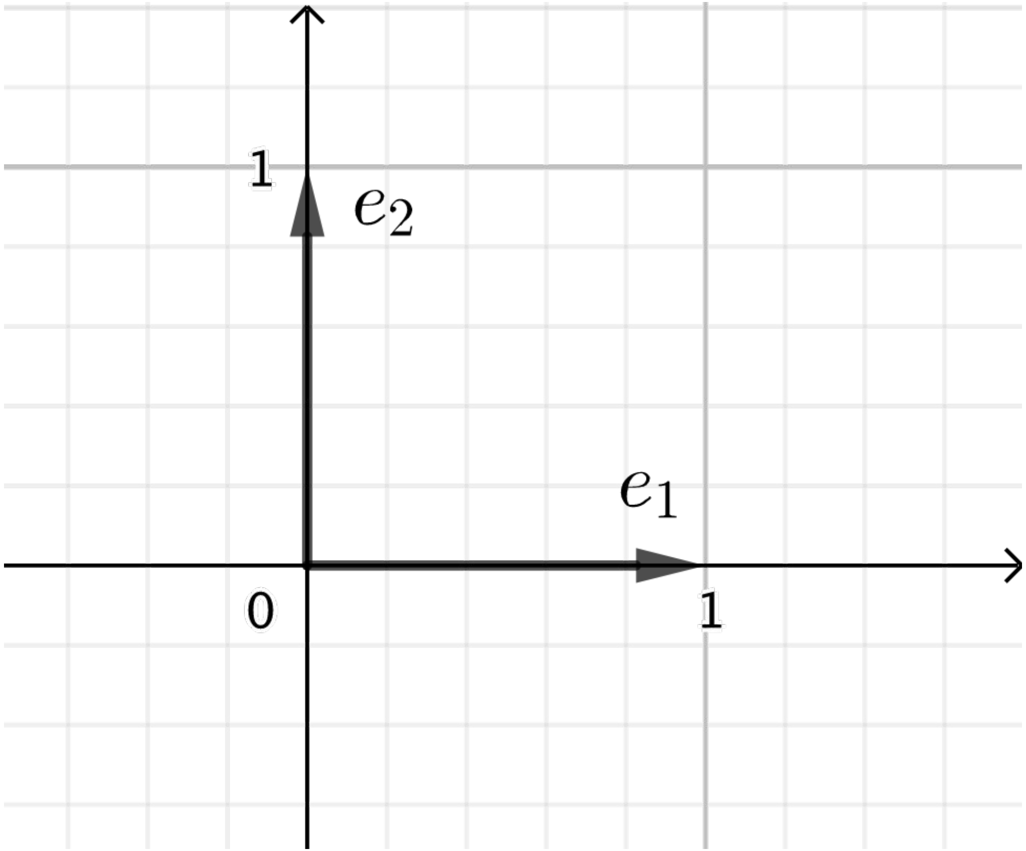
- Базисните вектори са линейно независими
- Всеки останал вектор в дадено векторно пространство се представя чрез тяхна линейна комбинация
    - Всяка линейна комбинация е уникална, т.е. Няма два различни вектора с една и съща линейна комбинация с базисните вектори

Базисни вектори

- Всяка двойка линейно независими вектори формира базис в 2D пространството.
    - За 3D тройка линейно независими вектори формират базис и т. н.
- За практически цели често векторите, които съответстват на координатните ос .
- Например за 2D: $e_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} e_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

- Понякога базисните вектори се транспонират, за да се превърнат в стълбое ината, а на 2-рия е y координата и т.н.
- За 3D: $e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

Базисни вектори за 2D



Линейни трансформации

- Трансформация
  - **Съпоставяне** (функция) между две **векторни пространства** V -> W
  - Частен случай: Съпоставяне на пространство върху себе си V -> V
    - Тогава говорим за **линеен оператор**
  - На всеки вектор от V се съпоставя друг вектор от W
- Линейна
  - Допускат се само линейни комбинации
  - Началната точка остава фиксирана
  - Всички линии остават линии (не се превръщат в криви)
  - Всички линии остават на еднакво разстояние (равноотдалеченост)
- Препоръчително видео:
https://www.youtube.com/watch?v=kYB8IZa5AuE

Някои трансформации

- Изобразяване на вектор сам в себе си
- Мащабиране (уголемяване/умаляване)
- Симетрия (отражение)
- Ротация
- Транслация (успоредно преместване)

Как да приложим линейна трансформация?

- Линейната трансформация често се задава с матрица на трансформацията.
- Всеки стълб от матрицата на трансформацията отговаря на базиса.
- Пример: В 2D с базисните вектори може да направим т... която изобразява вектора в себе си, нейната м... а така:

$$A = \begin{array}{c} \begin{pmatrix} \boxed{\begin{matrix} 1 \\ 0 \end{matrix}} & \boxed{\begin{matrix} 0 \\ 1 \end{matrix}} \end{pmatrix} \\ \quad\;\; \mathbf{e_1} \quad \mathbf{e_2} \end{array}$$

- За 3D тази матрица би била 3x3

Как да приложим линейна трансформация?

- Като използваме **матрицата на трансформацията**, просто можем да разгледаме произведението на **матрицата** и стария **вектор** и така ще получим като резултат **трансформирания ве** $v' = Av$
- **ВАЖНО: Редът е важен - матрицата по вектора**
- Пример (уголемяване на вектора 2 пъти във всяко направление в 2D):

$$v = \begin{pmatrix} 3 & 5 \end{pmatrix}$$

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

$$v' = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \begin{bmatrix} 3 \\ 5 \end{bmatrix} =$$

$$= \begin{pmatrix} 2 * 3 + 0 * 5 & 0 * 3 + 2 * 5 \end{pmatrix}$$

$$v' = \begin{pmatrix} 6 & 10 \end{pmatrix}$$

Как да приложим линейна трансформация?

- В Python:
  - Въведете вектора в np.array
  - Въведете матрицата в np.array
  - Използвайте .dot метода с два параметъра - матрицата и транспонирания вектор
- Пример:

```
x = np.array([3, 5])
A = np.array([[2, 0], [0, 2]])
transformedX = np.dot(A, x.T)

print(transformedX)
```

Линейна трансформация: Симетрия (отражение)

- Хоризонтална
  - Векторът се отразява огледално по отношение на ординатата.
  - Примерна матрица

$$A_{hor} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

- Вертикална
  - Векторът се отразява огледално по отношение на абсцисата
  - Примерна матрица

$$A_{vert} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Линейна трансформация: Ротация

- Може да завъртите вектора по часовниковата стрелка по ъгъл α.
- Примерна матрица:

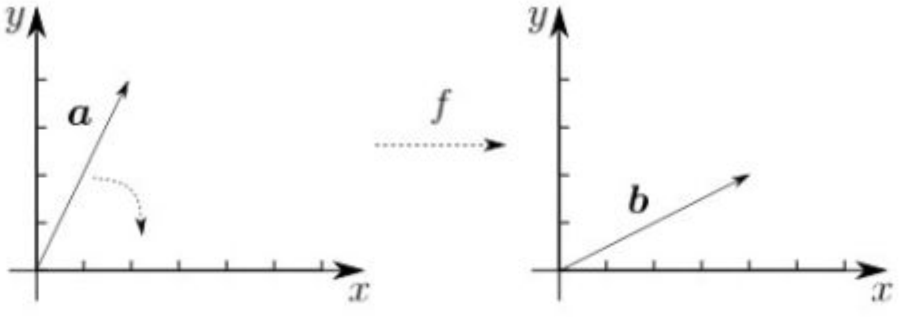
$$A_{rot\_cl} = \begin{pmatrix} \cos\ \alpha & \sin\ \alpha \\ -\sin\ \alpha & \cos\ \alpha \end{pmatrix}$$

- Може да завъртите вектора по посока обратна на часовниковата стрелка по ъгъл α. Пример

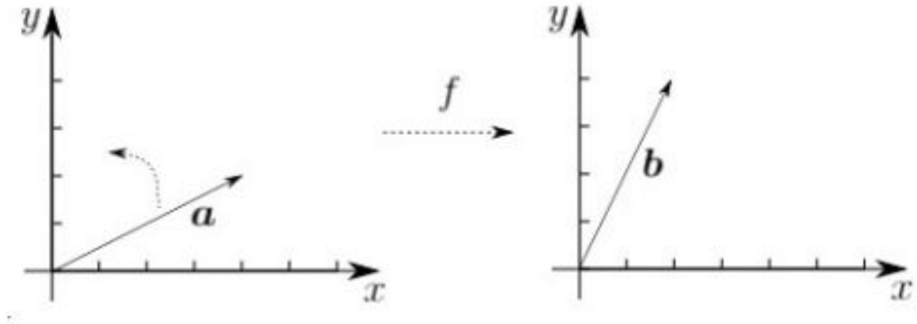
$$A_{rot\_not\_cl} = \begin{pmatrix} \cos\ \alpha & -\sin\ \alpha \\ \sin\ \alpha & \cos\ \alpha \end{pmatrix}$$

Линейна трансформация: Ротация

- По часовникова стрелка:



- Обратно на часовниковата стрелка:



Благодаря за вниманието!

Автор: Петър Р. Петров, учител по програмиране, ПГЕЕ „Константин Фотинов”, гр. Бургас