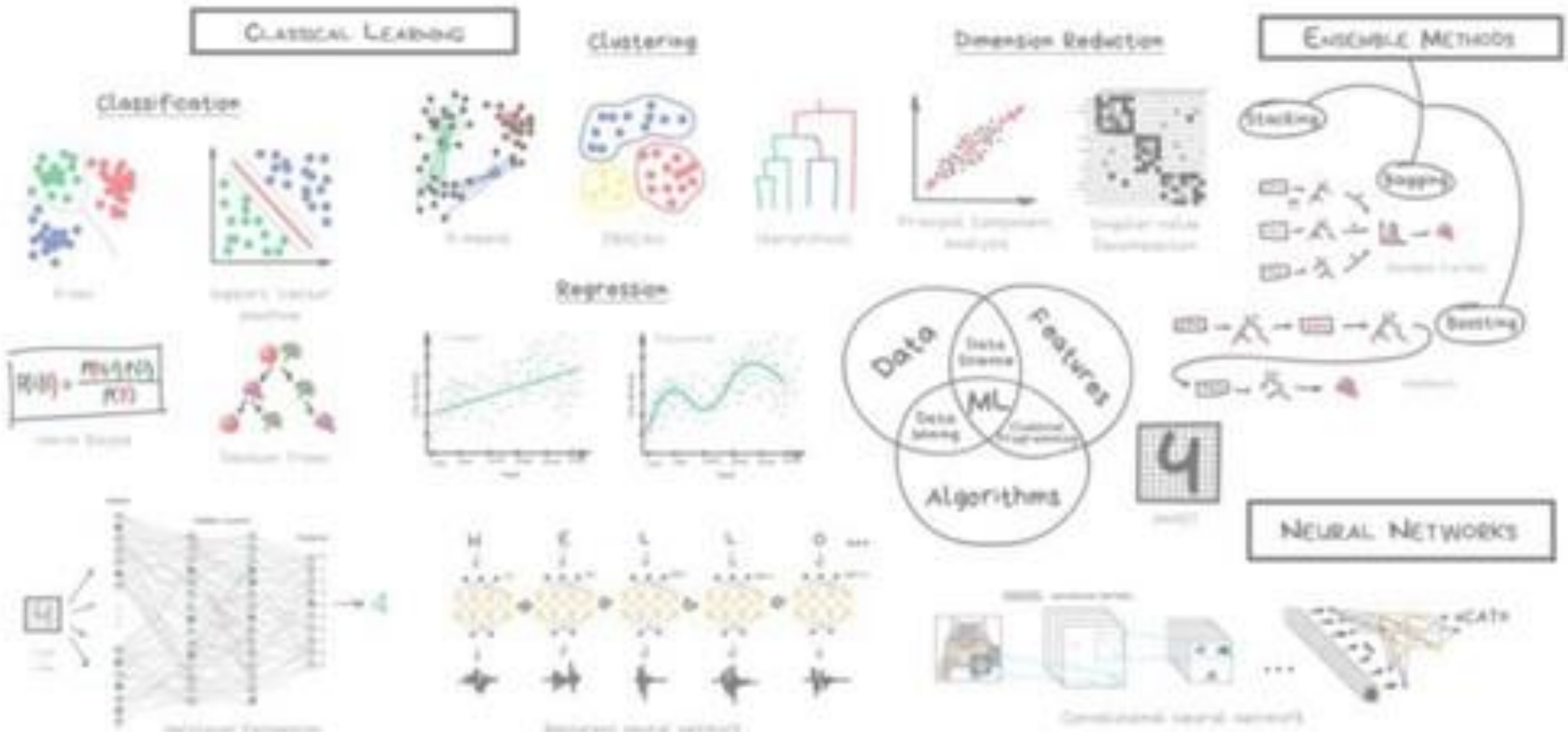
# МАШИНЕ НАВЧАННЯ

## Класичне навчання. Навчання без вчителя




Лекція №8

# Навчання без учителя

Навчання без вчителя (Unsupervised Learning) було створено в 90-і. На практиці використовується рідше ніж навчання з учителем.

Проте бувають задачі, де просто немає выбору:

реальні розмічені дані — рідкість.

Приклад: написати класифікатор автобусів:

йти на вулицю й фотографувати мільйон різних автобусів.

Коли немає розмітки використовують навчання без учителя.

Навчання без учителя, все ж таки, частіше використовують як метод аналізу даних, а не як основний алгоритм навчання.

Основний алгоритм:

Вкидуємо до алгоритму дані та спостерігаємо:

- Чи є кластери?
- Чи є залежності?
- Чи є асоціації?

Задачі навчання без учителя:

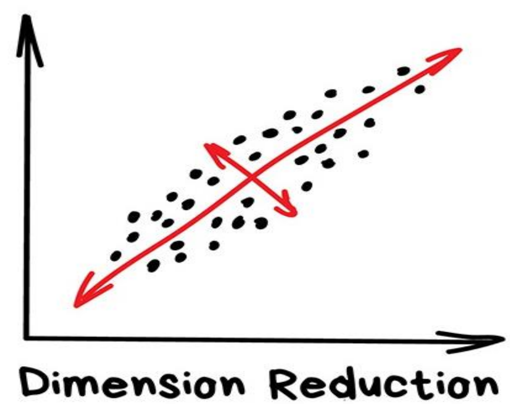
- **зменшення ромірності (узагальнення) — збирає конкретні ознаки в абстракції більш високого рівня**
- **кластеризація — розділяє об’єкты за незалежною ознакою**
- **асоціація — шукає закономірності у потоці замовлень**

# Зменшення розмірності
## (Dimension Reduction или Feature Learning)

Збирає конкретні ознаки в абстракції вищого рівня

Практична користь їх методів полягає в тому, що ми можемо об'єднати кілька ознак в одну і отримати абстракцію. Наприклад, собаки з трикутними вухами, довгими носами та великими хвостами поєднуються у корисну абстракцію «вівчарки». Так, ми втрачаємо інформацію про конкретні вівчарки, але нова абстракція набагато корисніша за ці зайві деталі. Плюс, навчання на меншій кількості розмірностей йде сильно швидше.Сьогодні використовують для:

- Визначення тематики та пошуку схожих документів
- Аналіз фейкових зображень
- Ризик-менеджмент

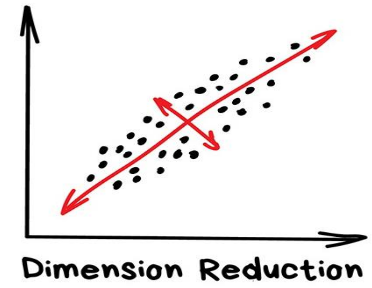



**Популярні алгоритми:**

Метод головних компонентів (PCA), Сингулярне розкладання (SVD), Латентне розміщення Діріхле (LDA), Латентно-семантичний аналіз (LSA)

# Зменшення розмірності

## Метод головних компонент (PCA)

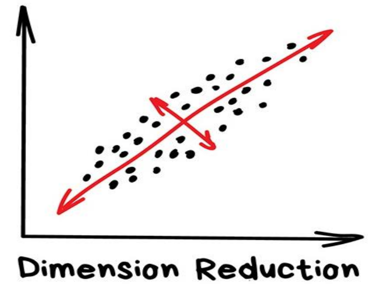



**Ме́тод головни́х компоне́нт** (МГК, англ. *principal component analysis, PCA*) — метод факторного аналізу в статистиці, який використовує ортогональне перетворення множини спостережень з можливо пов'язаними змінними (сутностями, кожна з яких набуває різних числових значень) у множину змінних без лінійної кореляції, які називаються **головними компонентами**.

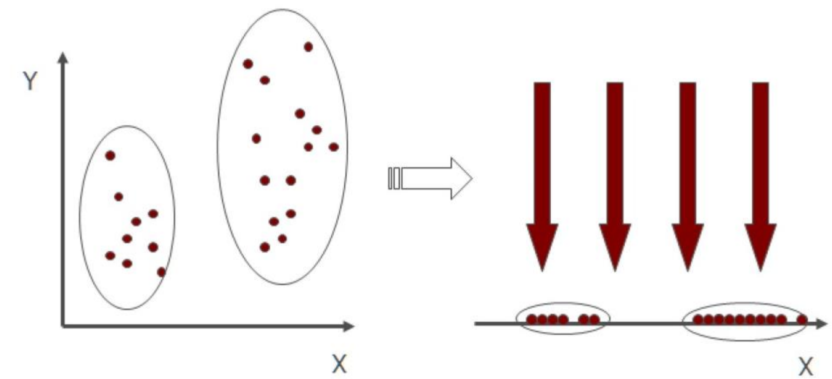
Метод головних компонент — один з основних способів зменшити розмірність даних, втративши найменшу кількість інформації. Винайдений Карлом Пірсоном у 1901 році та доповнений і розширений Гарольдом Готелінґом в 1933 р. Застосовується в багатьох галузях, зокрема, в економетриці, біоінформатиці, обробці зображень, для стиснення даних, у суспільних науках.

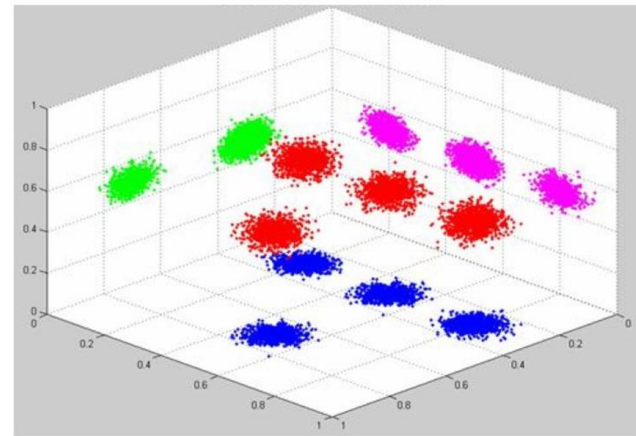
Обчислення головних компонент може бути зведене до обчислення сингулярного розкладу матриці даних або до обчислення власних векторів і власних чисел коваріаційної матриці початкових даних. Іноді метод головних компонент називають *перетворенням Кархунена —Лоева* або *перетворенням Хотеллінга* (англ. *Hotelling transform*).

# Зменшення розмірності

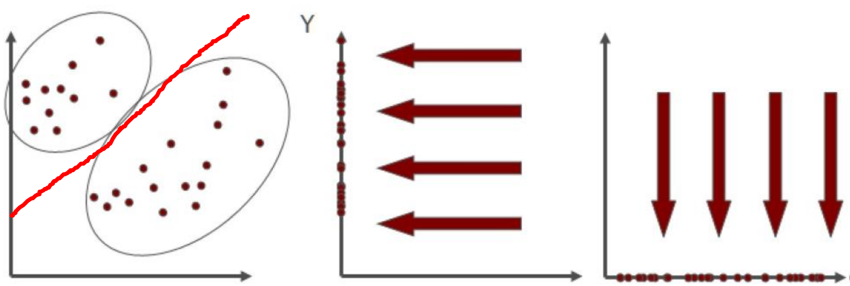



## **Метод головних компонент (PCA)**

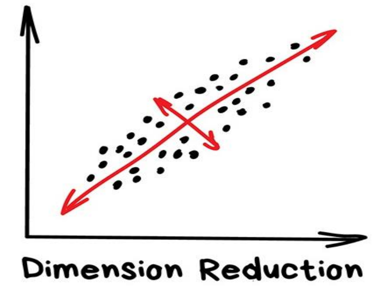


Y є надлишковим.
Для кластеризації достатньо X

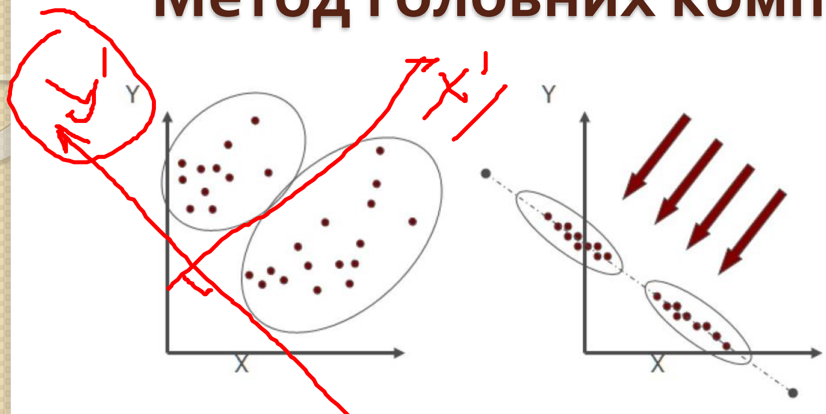


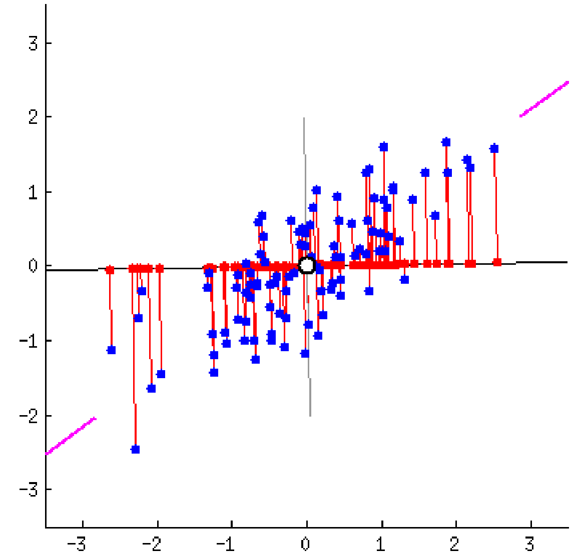

Не існує проекції яка б дозволила розділити дані




# Зменшення розмірності

## Метод головних компонент (PCA)

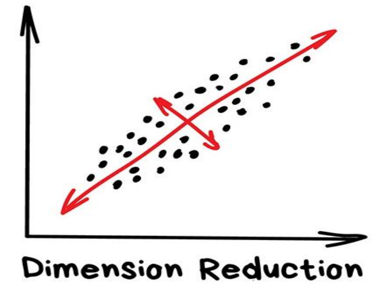


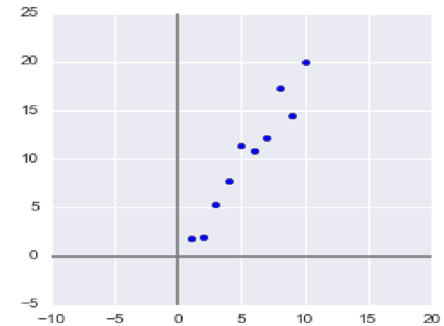
Однак, якщо проеціювати дані на лінійну комбінацію двох осей то можна легко кластеризувати дані

- Знаючи залежності ми можемо виразити кілька ознак через одну і працювати з більш простою моделлю. Звичайно, уникнути втрат інформації не вдасться, але мінімізувати її допоможе метод PCA.
- Висловлюючись суворіше, даний метод апроксимує n-розмірну хмару спостережень до еліпсоїда (теж n-вимірного), півосі якого і будуть майбутніми головними компонентами. І про проекції на такі вісі (зниження розмірності) зберігається найбільше інформації.

# Зменшення розмірності




## Метод головних компонент (PCA)

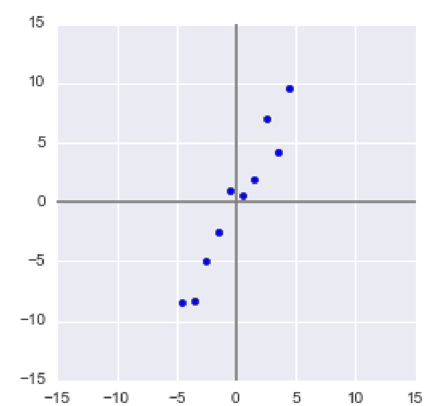


У цій вибірці у нас є дві ознаки, що сильно корелюють одна з одною. За допомогою алгоритму PCA ми зможемо легко знайти ознаку-комбінацію і виразити обидві ці ознаки однією новою.

Для опису випадкової величини використовуються моменти:

- мат. очікування – центр тяжіння» (положення) випадкової величини
- дисперсія – "розмір" (розкид) випадкової величини.

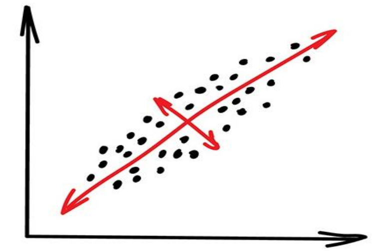
Сам процес проектування на вектор ніяк не впливає на значення середніх, оскільки для мінімізації втрат інформації наш вектор має проходити через центр нашої вибірки.



Тому можна відцентрувати вибірку - лінійно змістити її так, щоб середні значення ознак дорівнювали нулеві. Це дуже спростить подальші обчислення (хоча можна обійтися і без центрування).

# Зменшення розмірності

## Метод головних компонент (PCA)




**Для опису форми випадкового вектора потрібна коваріаційна матриця:** матриця, у якої $(i, j)$-елемент є кореляцією ознак $(x_i, x_j)$.

Коваріація двох випадкових величин:

$$Cov(x_i, x_j) = E\left[\left(x_i - E(x_i)\right) \cdot \left(x_j - E(x_j)\right)\right] = E(x_i, x_j) - E(x_i)E(x_j)$$

У даному випадку $E(x_i) = E(x_j) = 0$

$$Cov(x_i, x_j) = E(x_i, x_j); \quad Cov(x_i, x_i) = Var(x_i)$$

Таким чином, в нашій матриці по діагоналі будуть дисперсії, а в інших комірках – коваріації відповідних пар ознак. Через симетричність коваріації матриця теж буде симетричною

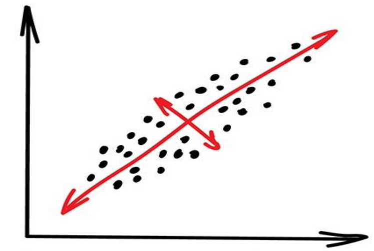
$$\Sigma = \begin{bmatrix} Var(x_i) & Cov(x_i, x_j) \\ Cov(x_i, x_j) & Var(x_j) \end{bmatrix}$$

Тепер треба знайти такий вектор $u$ (у нашому випадку тільки один, $u^T u = 1$), при якому буде максимальною дисперсія проекції нашої вибірки $u^T X$.

Дисперсія проекції на вектор

$$Var(u^T X) = E\left((u^T X) \cdot (u^T X)^T\right) = E(u^T X \cdot X^T u) = u^T E(X \cdot X^T) u = u^T \Sigma u$$

# Зменшення розмірності
## Метод головних компонент (PCA)




Необхідно максимізувати $u^T \Sigma u$ при обмеженні $u^T u = 1$
Використовуємо метод множників Лагранжа:

$$\mathcal{L} = u^T \Sigma u + \lambda(1 - u^T u)$$

Максимум досягається при $\frac{\partial \mathcal{L}}{\partial u} = 0$: $\Sigma u = \lambda u \; \rightarrow \lambda = u^T \Sigma u$

**Отже:** $u$ – власний вектор матриці $\Sigma$ з максимальним власним значенням $\lambda$

**Таким чином, напрямок максимальної дисперсії у проекції завжди збігається з власним вектором, що має максимальне значення, що дорівнює величині цієї дисперсії.**
І це справедливо також для проекцій на більшу кількість вимірювань – дисперсія (коваріаційна матриця) проекції на m-мірний простір буде максимальною у напрямку m власних векторів, що мають максимальні власні значення.

З проекцією зручно працювати, будувати на її основі гіпотези та розробляти моделі. Але іноді корисно розкодувати, наприклад, виявлені викиди, щоб подивитися, що за ними стоїть.

# Власні значення та власні вектори

**Означення.** Нехай $A$ - квадратна матриця розміру $n \times n$ над полем $F$. Число $\lambda \in F$ називається власним значенням матриці $A$, якщо існує ненульовий вектор $v$ такий, що

$$Av = \lambda v.$$

В цьому випадку вектор $v$ називається власним вектором матриці $A$ з власним значенням $\lambda$.

Аналогічно можна дати означення власного вектора лінійного оператора.

**Означення.** Нехай $f : V \to V$ - лінійний оператор. Число $\lambda \in F$ називається власним значенням $f$, якщо існує ненульовий вектор $v \in V$ такий, що

$$f(v) = \lambda v.$$

В цьому випадку вектор $v$ називається власним вектором лінійного оператора $f$ з власним значенням $\lambda$.

**Означення.** Характеристичним многочленом матриці $A$ називається $\chi_A(\lambda) = det(A - \lambda E)$, або докладніше

$$\chi_A(\lambda) = det \begin{pmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} - \lambda & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{pmatrix}.$$

# Власні значення та власні вектори

**Означення.** Характеристичним многочленом матриці $A$ називається $\chi_A(\lambda) = det(A - \lambda E)$, або докладніше

$$\chi_A(\lambda) = det \begin{pmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} - \lambda & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{pmatrix}.$$

**Теорема.** Власні значенням матриці $A$ є в точності коренями її характеристичного многочлена $\chi_A(\lambda)$

**Доведення.** Якщо вектор $v$ є власним вектором матриці $A$ з власним числом $\lambda$, то

$$Av = \lambda v,$$

це фактично система лінійних рівнянь, яку можна переписати таким чином

$$Av - \lambda Ev = 0,$$

$$(A - \lambda E)v = 0.$$

Це однорідна система лінійних рівнянь, яка має ненульовий розв'язок. Це буває тоді і лише тоді, коли матриця $A - \lambda E$ вироджена, тобто, $det(A - \lambda E) = 0$. 😊 **Успішно доведено.**

# Власні значення та власні вектори

**Приклад**

$$A = \begin{pmatrix} 1 & 4 & 1 \\ 0 & 2 & 2 \\ 0 & 0 & 3 \end{pmatrix},$$

$$|A - \lambda E| = (1-\lambda) \cdot \begin{vmatrix} 2-\lambda & 2 \\ 0 & 3-\lambda \end{vmatrix} - 4 \cdot \begin{vmatrix} 0 & 2 \\ 0 & 3-\lambda \end{vmatrix} + 1 \cdot \begin{vmatrix} 0 & 2-\lambda \\ 0 & 0 \end{vmatrix} =$$

$$= (1-\lambda) \cdot (2-\lambda) \cdot (3-\lambda).$$

**Розв'язання**

$$(1-\lambda) \cdot (2-\lambda) \cdot (3-\lambda) = 0,$$

$$\lambda_1 = 1,$$

$$\lambda_2 = 2,$$

$$\lambda_3 = 3.$$

**Python** with **Numpy**

```
>>> A = np.matrix("1 4 1; 0 2 2 ; 0 0 3")
>>> A
matrix([[1, 4, 1],
        [0, 2, 2],
        [0, 0, 3]])
>>> w, v = np.linalg.eig(A)
>>> w
array([1., 2., 3.])
```

# Власні значення та власні вектори

**Приклад** $A = \begin{pmatrix} 7 & 3 \\ -5 & -1 \end{pmatrix}.$

$$|A - \lambda E| = \begin{vmatrix} 7-\lambda & 3 \\ -5 & -1-\lambda \end{vmatrix} = (7-\lambda)\cdot(-1-\lambda) - (-15) = \lambda^2 - 6\lambda + 8.$$

$$\lambda^2 - 6\lambda + 8 = 0,$$

$$\lambda_{1,2} = \frac{6 \pm \sqrt{36-32}}{2} = \frac{6 \pm 2}{2},$$

$$\lambda_1 = 4,$$

$$\lambda_2 = 2.$$

$$\begin{pmatrix} 7 & 3 \\ -5 & -1 \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 4 \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

$$\begin{cases} 7\cdot x_1 + 3\cdot x_2 = 4\cdot x_1 \\ -5\cdot x_1 - 1\cdot x_2 = 4\cdot x_2 \end{cases}$$

$$\begin{cases} 3\cdot x_1 + 3\cdot x_2 = 0 \\ -5\cdot x_1 - 5\cdot x_2 = 0 \end{cases}$$

$$x_2 = -x_1.$$

$$\begin{pmatrix} 7 & 3 \\ -5 & -1 \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 2 \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

$$\begin{cases} 7\cdot x_1 + 3\cdot x_2 = 2\cdot x_1 \\ -5\cdot x_1 - 1\cdot x_2 = 2\cdot x_2 \end{cases}$$

$$\begin{cases} 5\cdot x_1 + 3\cdot x_2 = 0 \\ -5\cdot x_1 - 3\cdot x_2 = 0 \end{cases}$$

$$x_2 = -\frac{5}{3}\cdot x_1.$$

$$\lambda_1(x_1, -x_1);$$

$$\lambda_2\left(x_1, -\frac{5}{3}x_1\right)$$

# Дякую за увагу