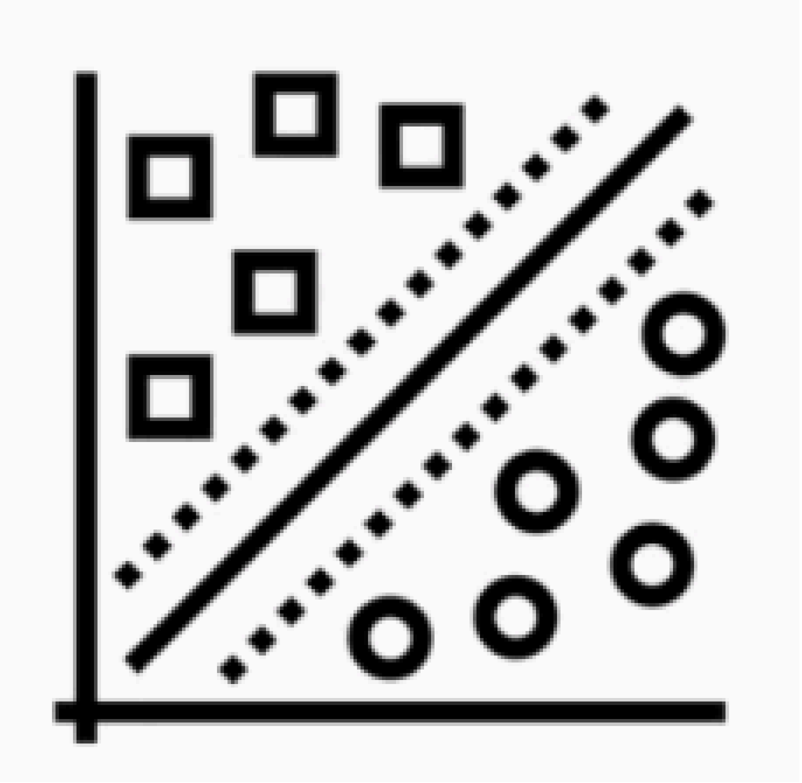
КЛАСИФІКАЦІЯ ЗА ДОПОМОГОЮ МЕТОДУ ОПОРНИХ ВЕКТОРІВ

Підготував: студент ОІ-32
Криворучко Микола

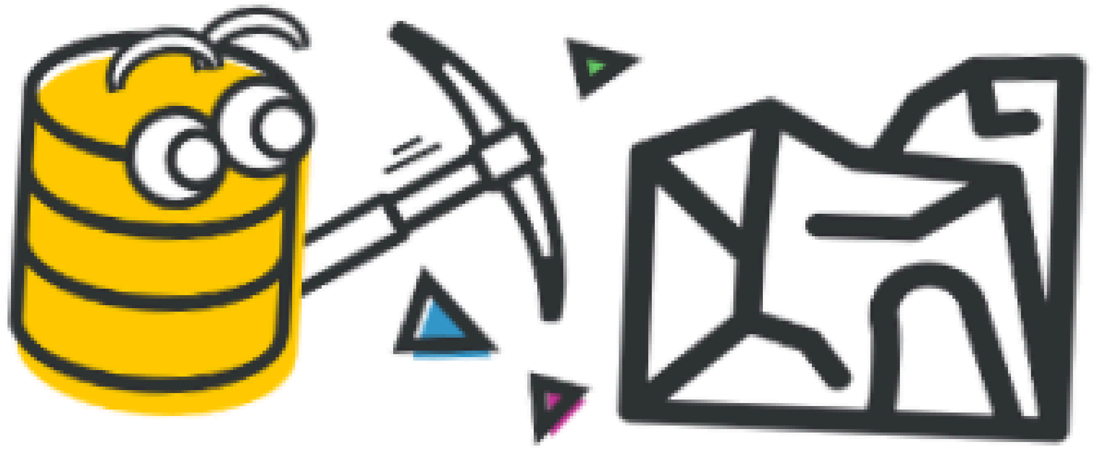
ЗМІСТ

1. Що таке Data Mining?
2. Які існують задачі Data Mining?
3. Задача класифікаї
4. Методи класифікації
5. Метод опорних векторів
6. Висновок



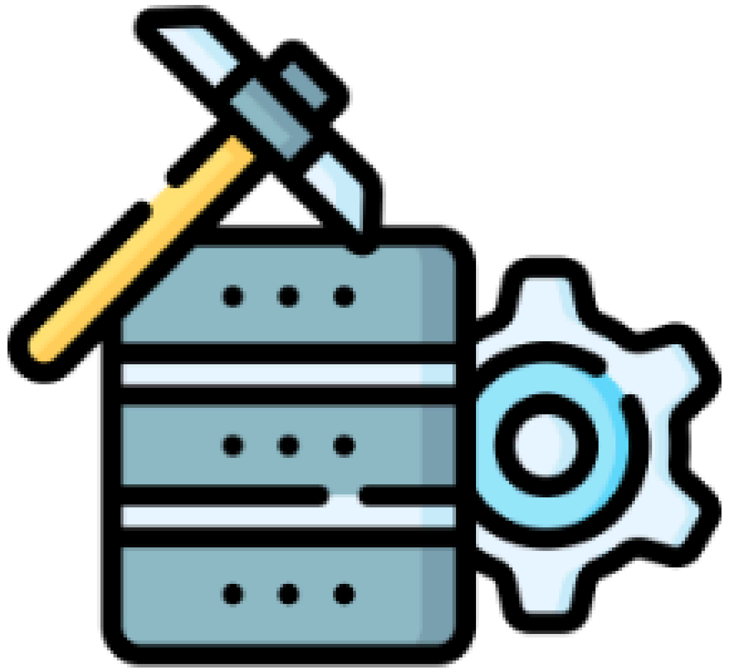
Що таке Data Mining?

Набір методів, алгоритмів та засобів опрацювання "сирих даних" із метою видобування з них необхідної інформації(знань)



Задачі Data Mining

1. Задача класифікації
2. Задача регресії
3. Задача кластеризації
4. Побудова асоціативних правил



Задача класифікації

задача класифікації зводиться до визначення класу об'єкта по його характеристикам. Необхідно зауважити, що в цьому завданні множина класів, до яких може бути віднесений об'єкт, відомо заздалегідь.

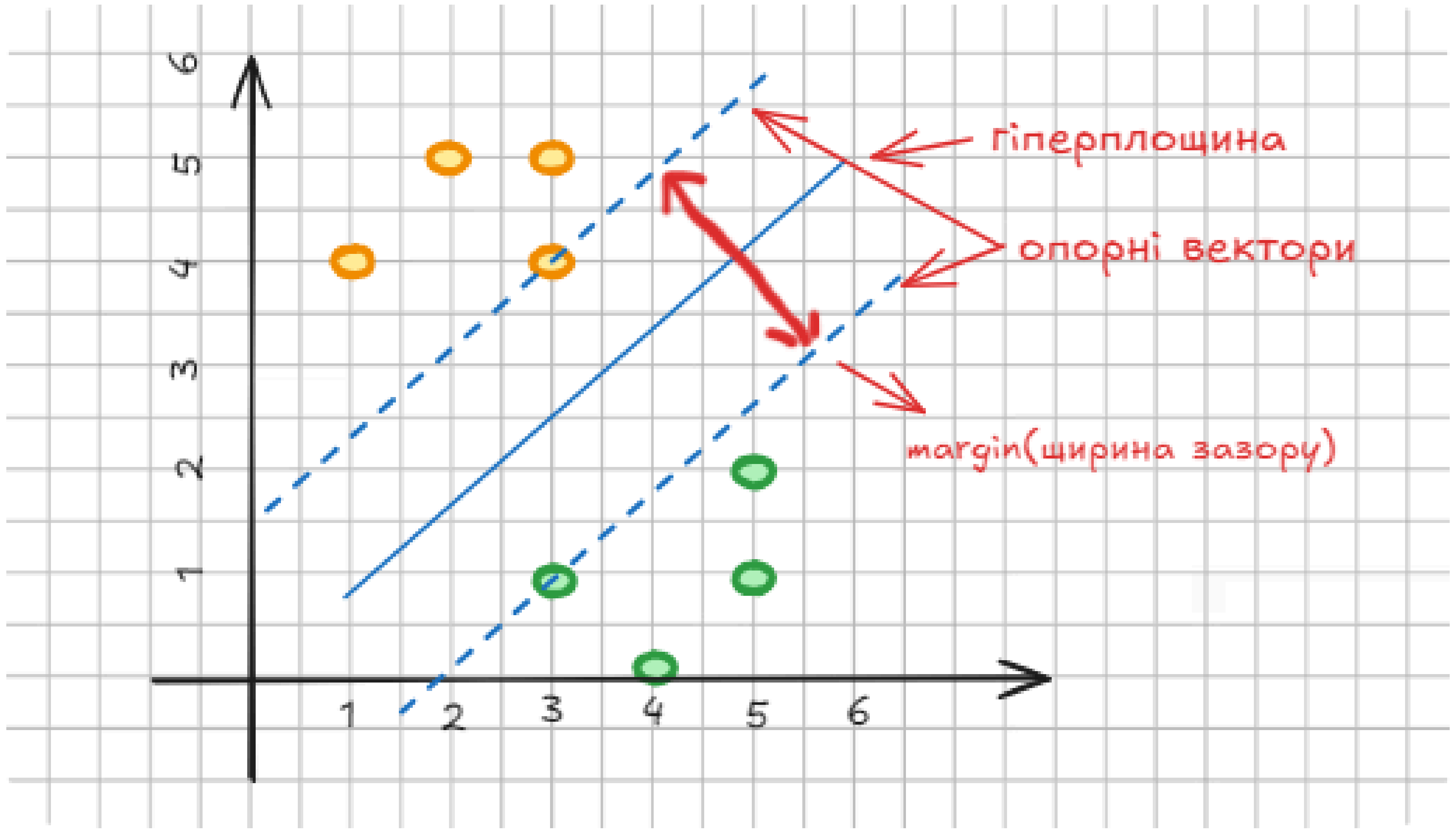
Методи класифікації

- метод найближчого сусіда
- наївна байеосва класифікація
- CBR
- метод опорних векторів
- нейронні мережі
- генетичні алгоритми
- група присвячених побудові дерева рішень

Метод опорних векторів

це алгоритм, який шукає оптимальну гіперплощину, що максимально розділяє дані різних класів, використовуючи лише "опорні вектори" — найближчі до межі точки.

Основні поняття



Формула гіперплощини

$$w^T x + b = 0$$

набір коефіцієнтів:
$[b1, b2, ... bn]^T$

$$x = (x_1, x_2, \ldots, x_n)$$

Відповідає за "зсув" гіперплощини від початку координат

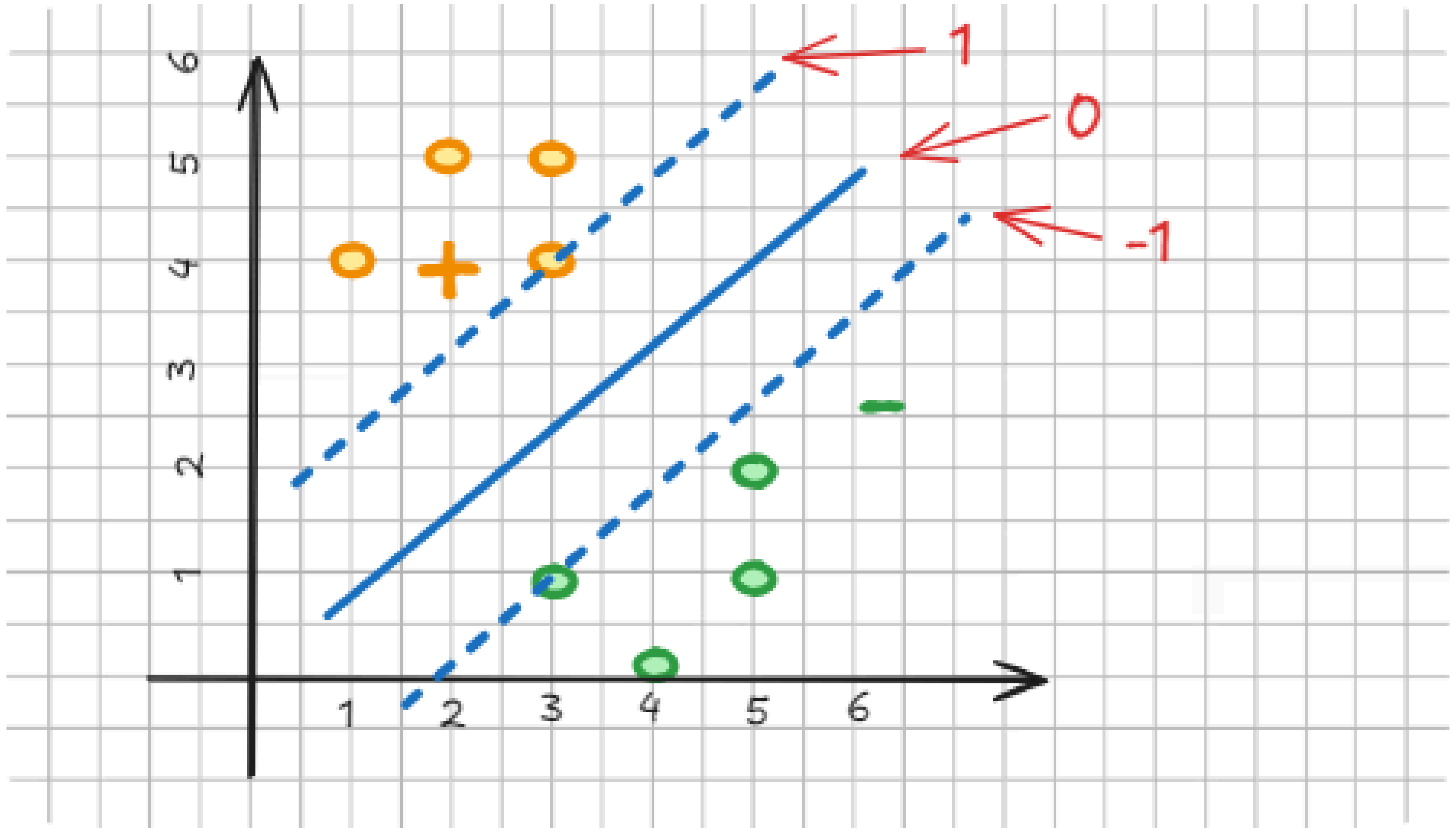
Розуміння класифікації в SVM

Group (Y_i)	*x*	*y*
A (+1)	1	4
A (+1)	2	5
A (+1)	3	5
A (+1)	3	4
B (-1)	6	1
B (-1)	4	0
B (-1)	5	2
B (-1)	5	1

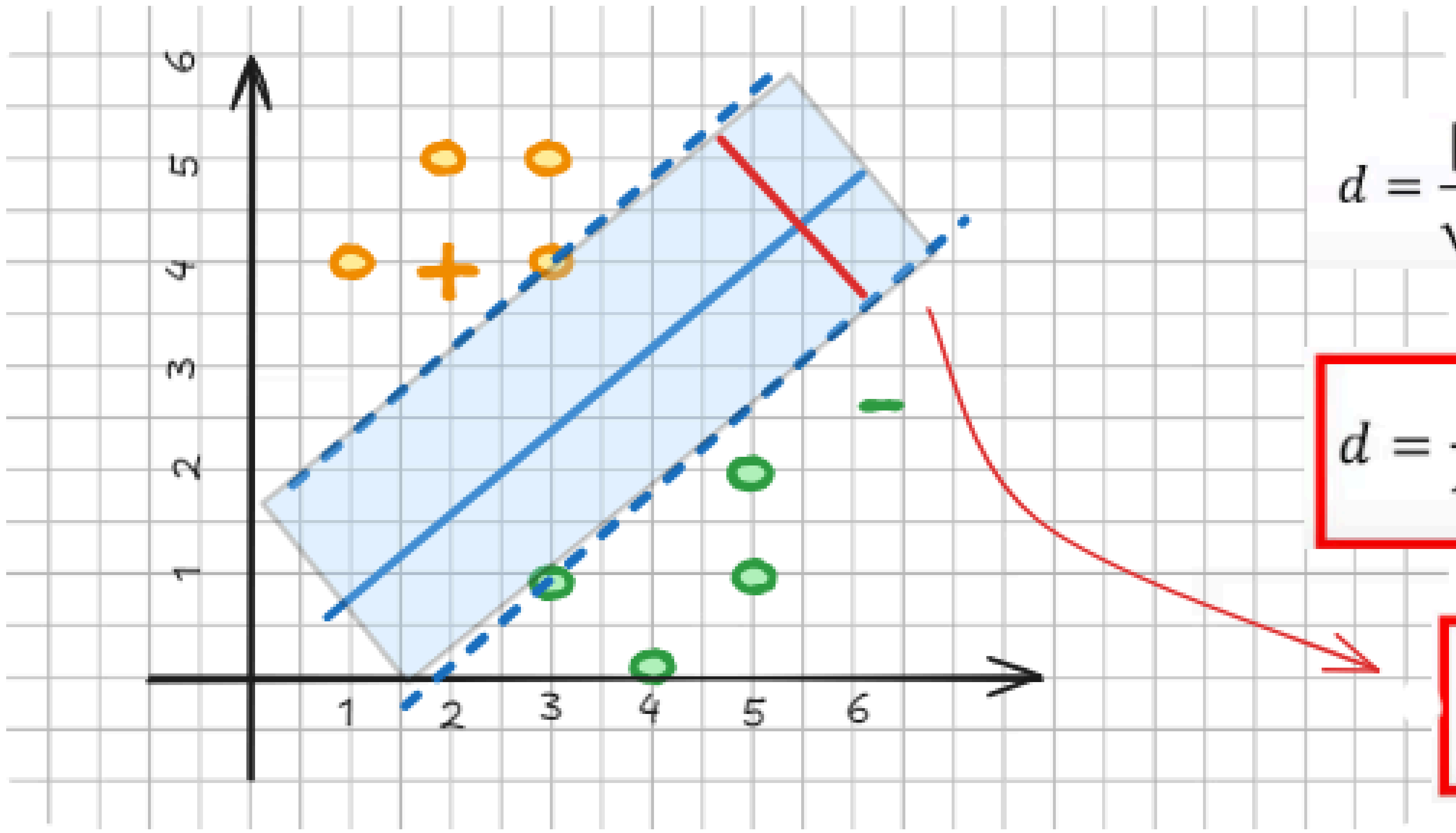
$$Y_i = [\,1, 1, 1, 1, -1, -1, -1, -1]$$

$$Y \begin{cases} +1 \text{ if } w^T x + b \geq 0 \\ -1 \text{ if } w^T x + b < 0 \end{cases}$$

Розуміння класифікації в SVM



Відстань між опорними векторами



$$d = \frac{|C_2 - C_1|}{\sqrt{A^2 + B^2}}$$

$$\downarrow$$

$$\boxed{d = \frac{2}{\sqrt{A^2 + B^2}}}$$

$$\downarrow$$

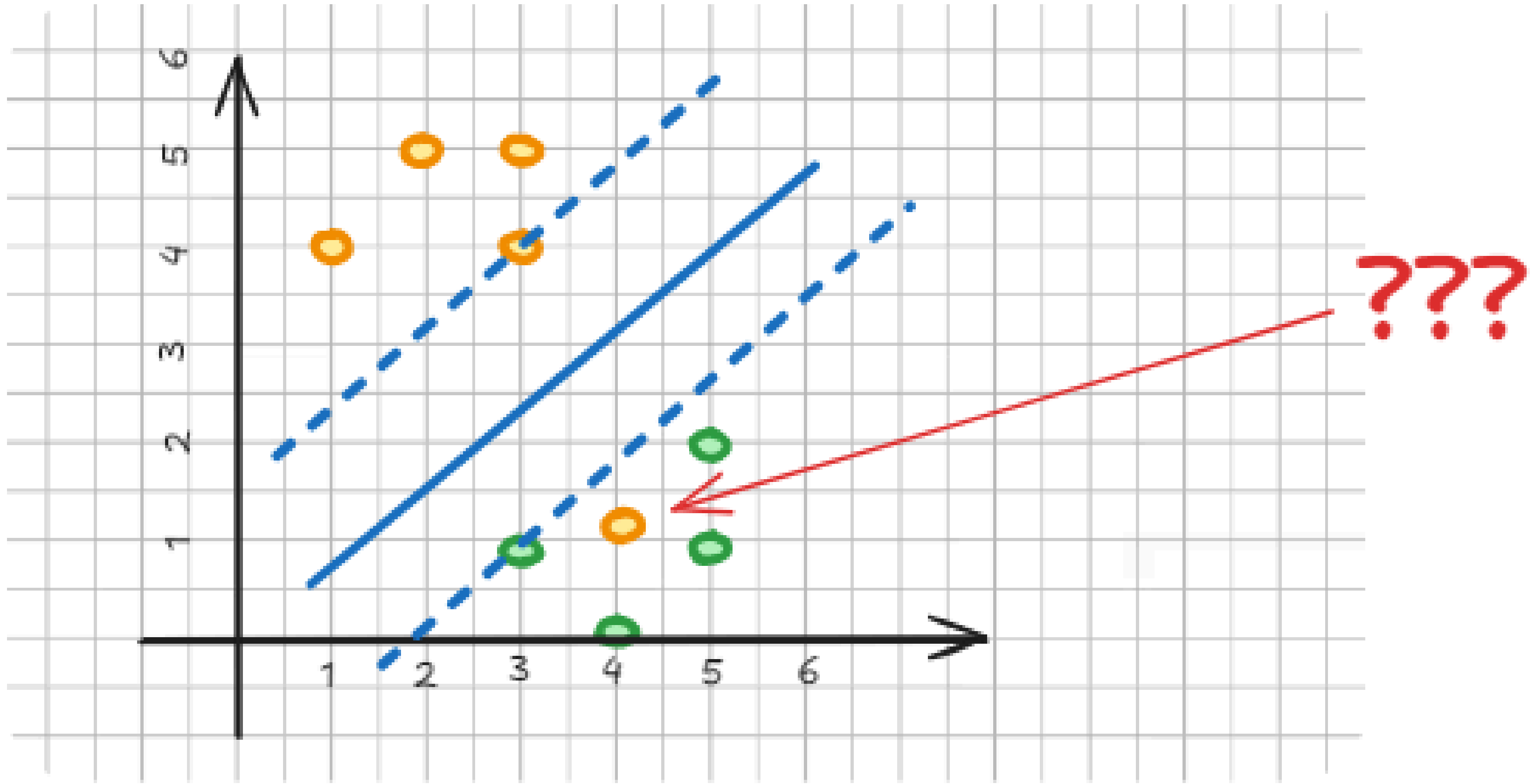
$$\boxed{\frac{2}{\|w\|}}$$

Задача методу

$\frac{2}{\|w\|} \longrightarrow$ макисмізувати відстань $\longrightarrow$ мінімізувати $\|w\|$

$$\max \frac{2}{\|w\|} \text{ such that } w^T x_i + b \quad \begin{array}{l} \geq 1 \;\; \text{if } Y_i = +1 \\ \leq -1 \text{ if } Y_i = -1 \end{array}$$

Проблеми



Функція втрат

C (параметр регуляризації)

- Контролює баланс між **шириною зазору** та **кількістю/ступенем помилок**.
- Великий C: алгоритм "карає" за помилки сильніше → вузький зазор, але менше помилок.
- Малий C: алгоритм дозволяє більше помилок, але зазор стає ширшим (краща узагальненість).

$$\boxed{\min \frac{1}{2}\|w\|^2 + C\sum_{i}^{N} \varepsilon_i}$$

- Якщо точка правильно класифікована та лежить поза margin → $\varepsilon_i = 0$.
- Якщо точка **усередині margin** → $\varepsilon_i > 0$.
- Якщо точка навіть на "неправильній стороні" площини → $\varepsilon_i > 1$.

$\frac{1}{2}\|w\|^2$

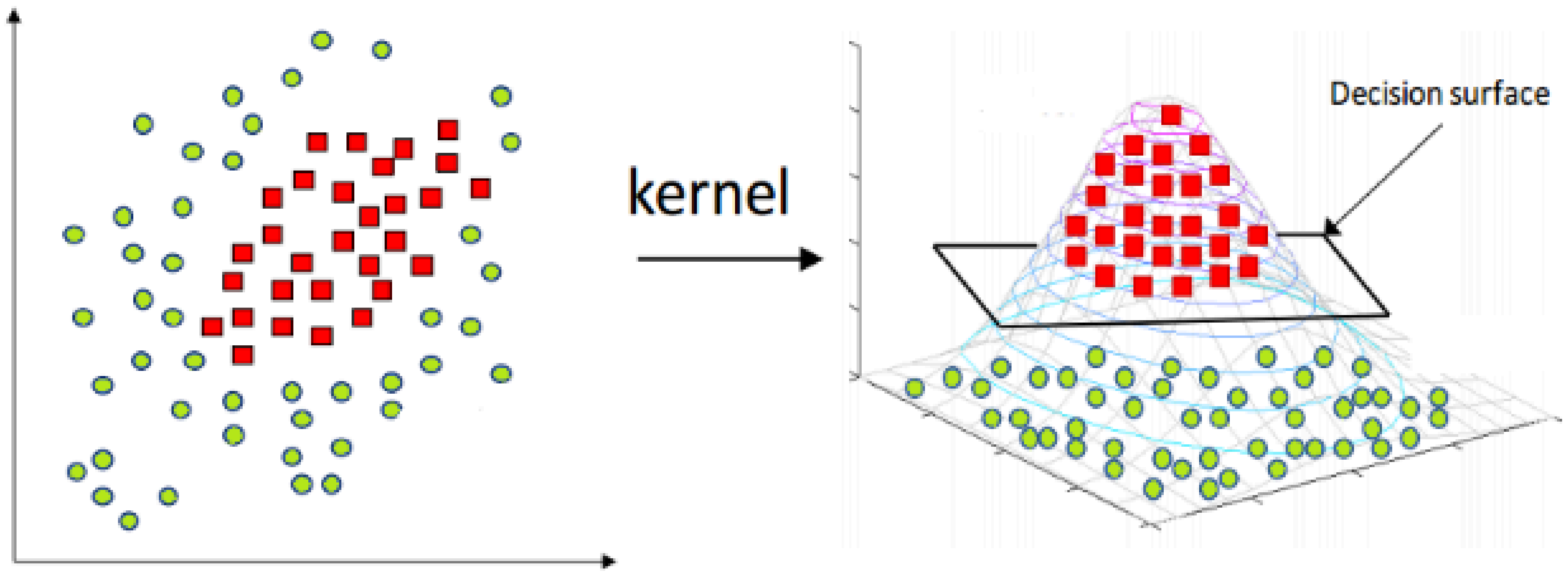
- Це **регуляризаційний член**.
- $\|w\|^2$ — квадрат норми вектора ваг (довжина вектора).
- Мінімізація цього терміну означає:
  - ми хочемо, щоб w було якнайменше,
  - що еквівалентно **максимізації ширини зазору (margin)** між класами.

Функція втрат

$$\min \frac{1}{2}\|w\|^2 + C \sum_i^N \varepsilon_i \quad \boxed{\text{such that } Y_i(w^T x_i + b) \geq 1} - \varepsilon_i$$

Чим більший ε, тим **далі від правильного класу** точка може бути.

Kernel trick

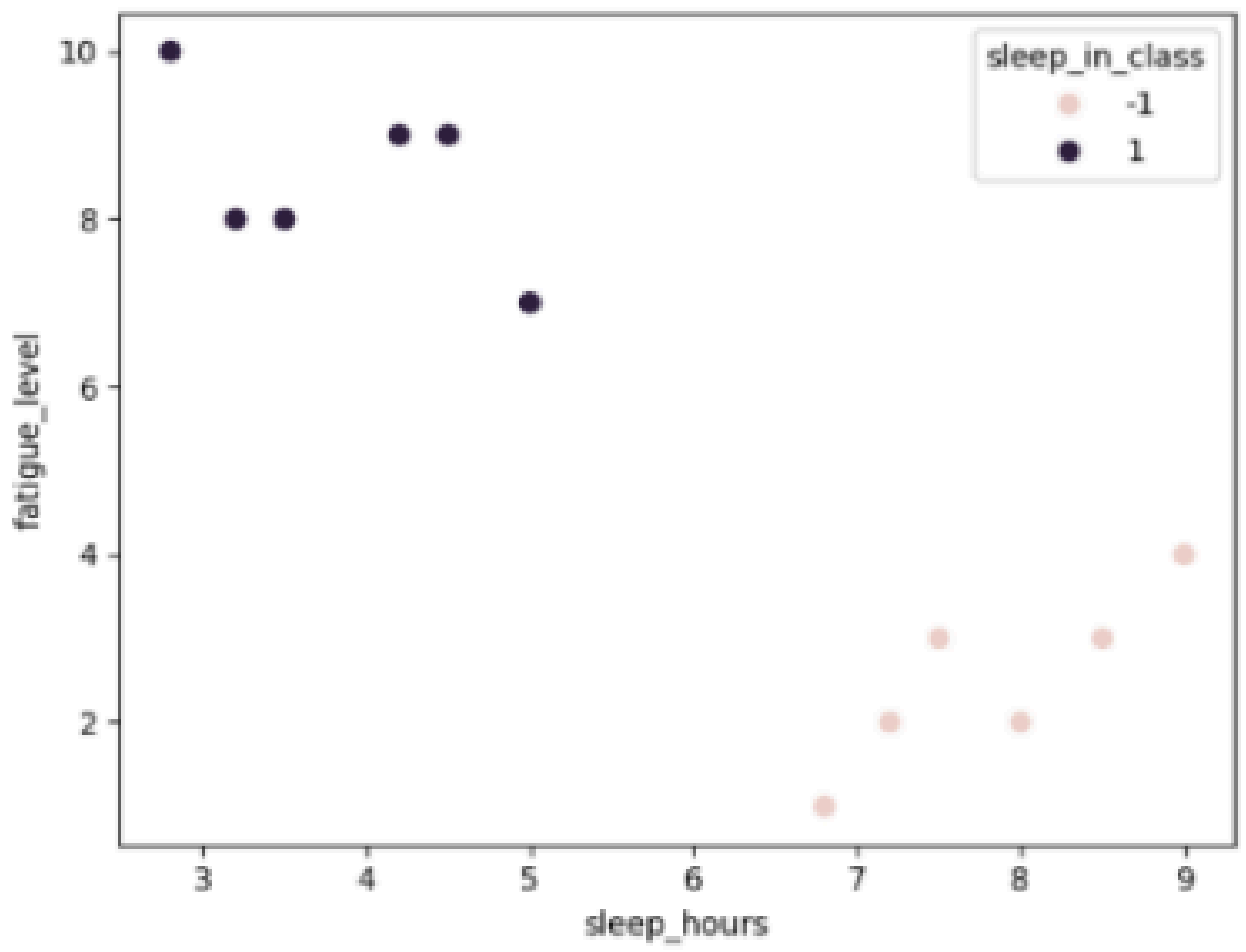


Завдання

Вхідні дані:

	student_name	sleep_hours	fatigue_level	sleep_in_class
0	Alex	8.0	2	-1
1	John	7.5	3	-1
2	Mary	6.8	1	-1
3	Anna	9.0	4	-1
4	Dmytro	7.2	2	-1
5	Kate	8.5	3	-1
6	Olga	3.5	8	1
7	Andrew	4.2	9	1
8	Marta	5.0	7	1
9	Denys	2.8	10	1
10	Iryna	4.5	9	1
11	Sofia	3.2	8	1

Візуалізція:

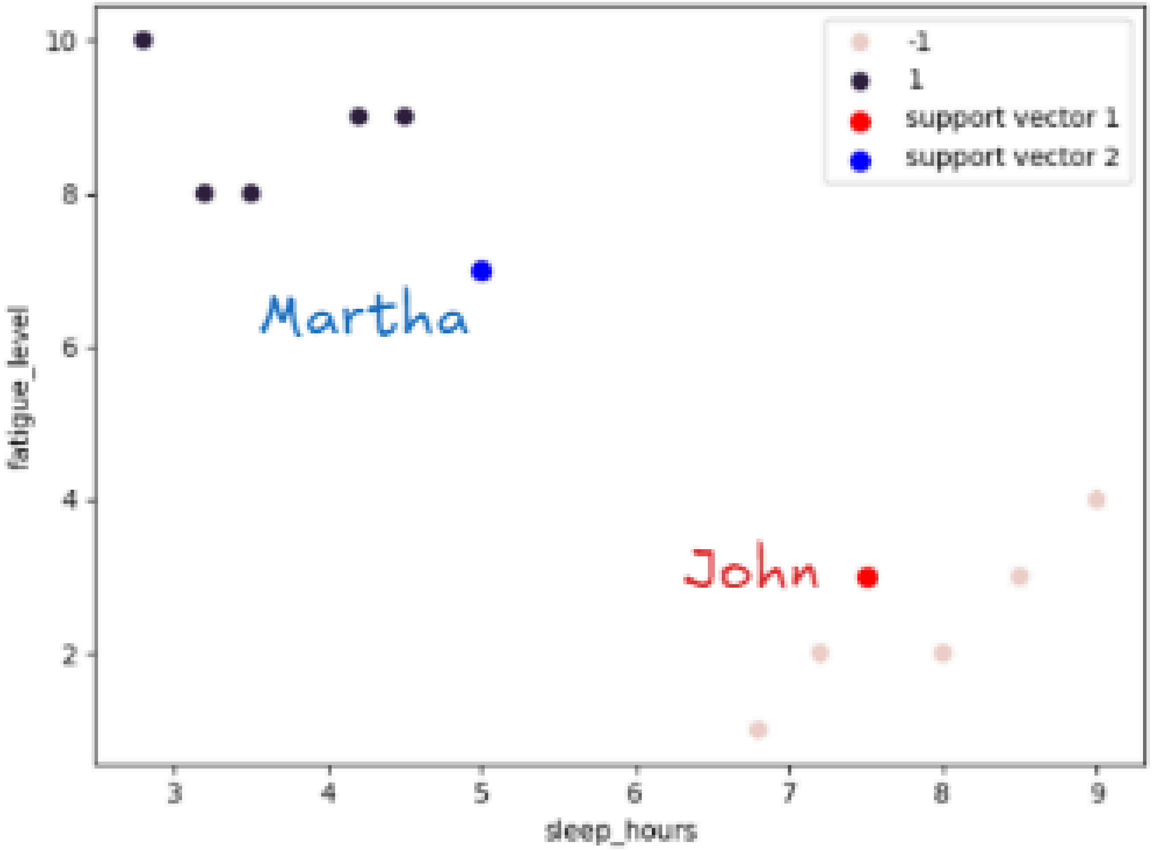


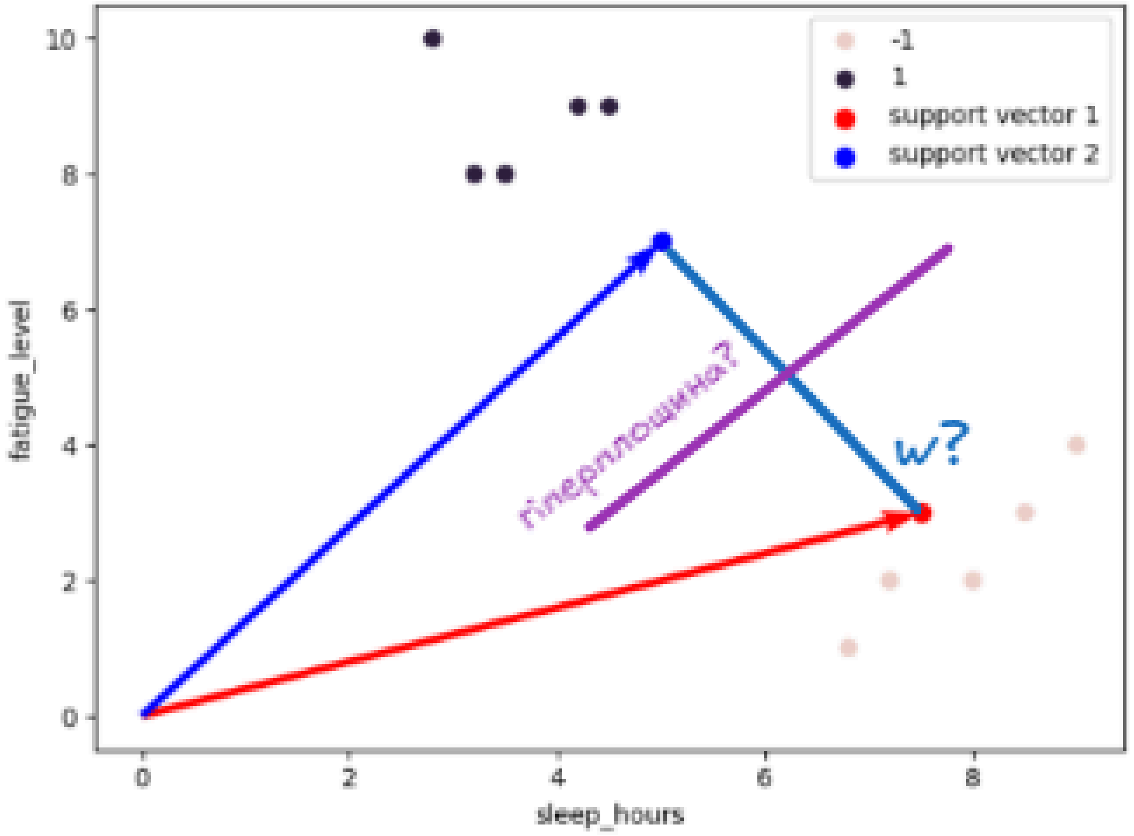
Знаходження опорних векторів

student_name	Olga	Andrew	Marta	Denys	Iryna	Sofia
Alex	7.5	7.964923	5.830952	9.541488	7.826238	7.683749
John	6.403124	6.847627	4.716991	8.421489	6.708204	6.594695
Mary	7.738863	8.411896	6.264184	9.848858	8.324062	7.871467
Anna	6.800735	6.931089	5.0	8.627862	6.726812	7.045566
Dmytro	7.049113	7.615773	5.4626	9.13017	7.502666	7.211103
Kate	7.071068	7.381734	5.315073	9.027181	7.211103	7.286288

мінімальна відстань між точками

Візуалізація опорних векторів

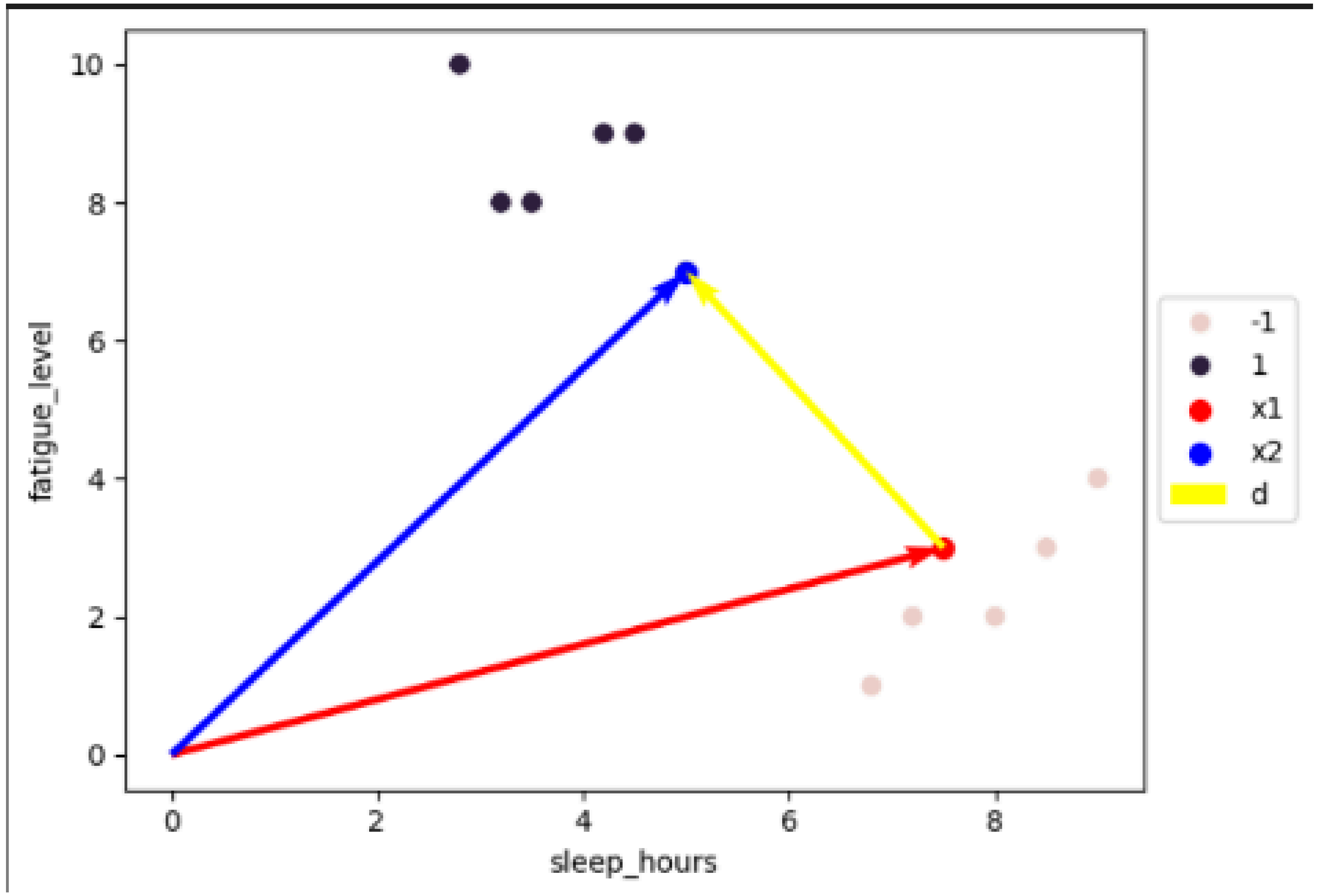




Розрахунок коефіцієнтів рівняння гіперплощини

$x1$ = [7.5, 3] - John
$x2$ = [5, 7] - Martha

$d = x2 - x1 = [2.5, -4]$



Розрахунок коефіцієнтів рівняння гіперплощини

a = 2 / ‖d‖ = 0.08988

Це масштабуючий коефіцієнт, який потрібен, щоб отримати правильний вектор нормалі

w

w для SVM.

$\boxed{w}$ = a * d = [-0.2247191   0.35955056]

Вектор w визначає нормаль до гіперплощини (лінії), яка розділяє класи.

w * x1 + b = -1

$\boxed{b}$ = -1 - w * x1 =

= -1 - [-0.2247191   0.35955056] * [7.5, 3] = -0.3932

Розрахунок гіперплощини

$$w * x + b = 0$$

$$[-0.2247191 \quad 0.35955056] * [x, y] - 0.3932 = 0$$

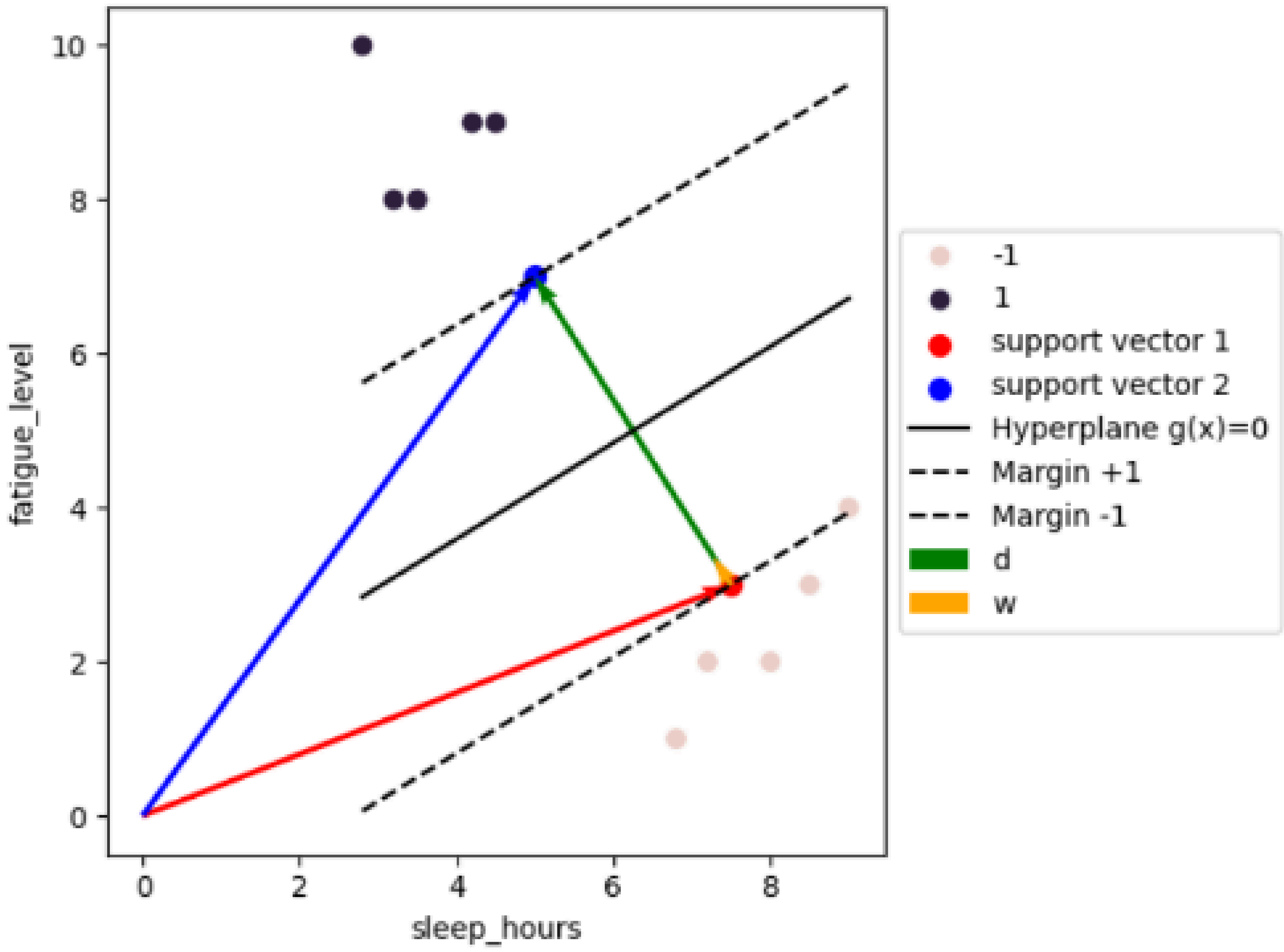
$$-0.2247191 * x + 0.35955056 * y - 0.3932 = 0$$

$$y = 0.3932/0.35955056 + 0.2247191/0.35955056 * x$$

$$g(x) = 1.094 + 0.625 * x$$

рівняння гіперплощини

Фінальна візуалізація розрахунків

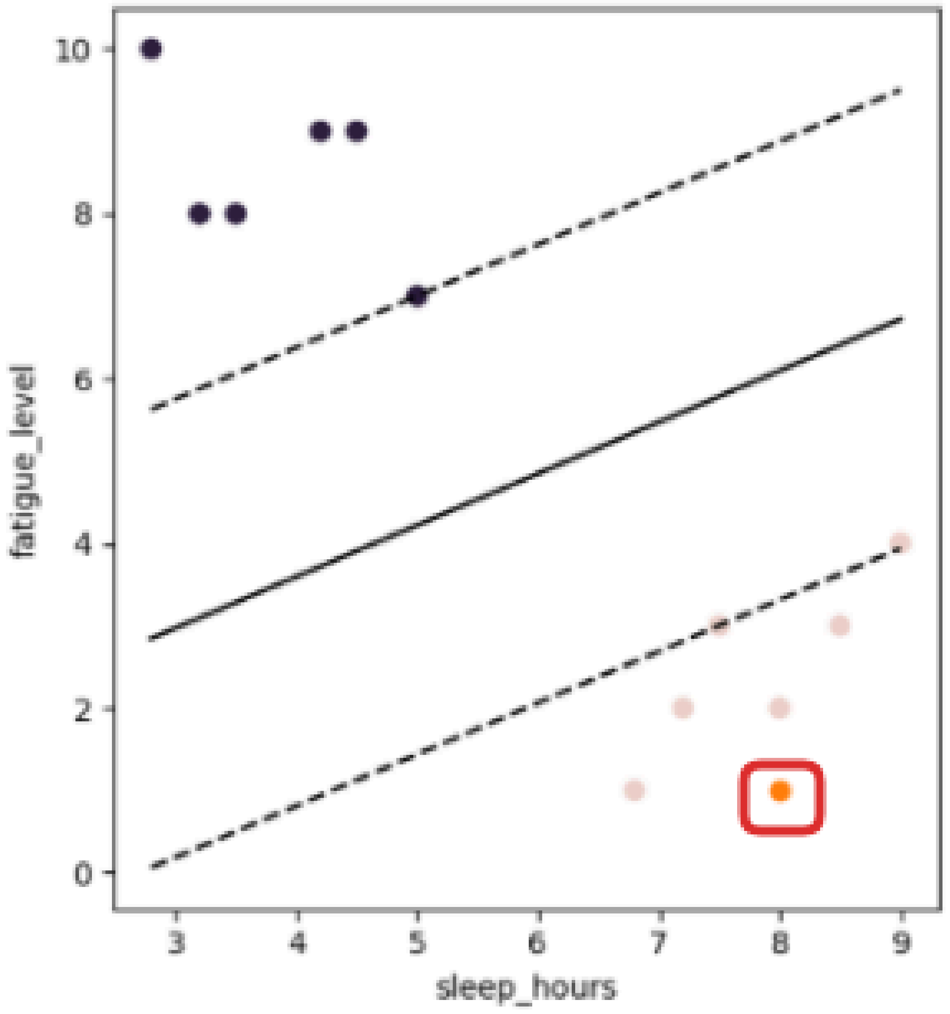


Тестова вибірка

x = [x, y] = [8, 1]

res = w * x + b =
= [-0.2247191   0.35955056] *
[8, 1] - 0.3932 =
= -1.83140224

res < 0 => Y = -1



Висновок

У ході виконання лабораторної роботи я ознайомився з методом опорних векторів та принципом побудови гіперплощини для класифікації. Я навчився визначати опорні вектори, знаходити рівняння гіперплощини та візуалізувати результати роботи алгоритму. Отримані знання дозволили мені краще зрозуміти, як SVM застосовується для розв'язання задач класифікації.