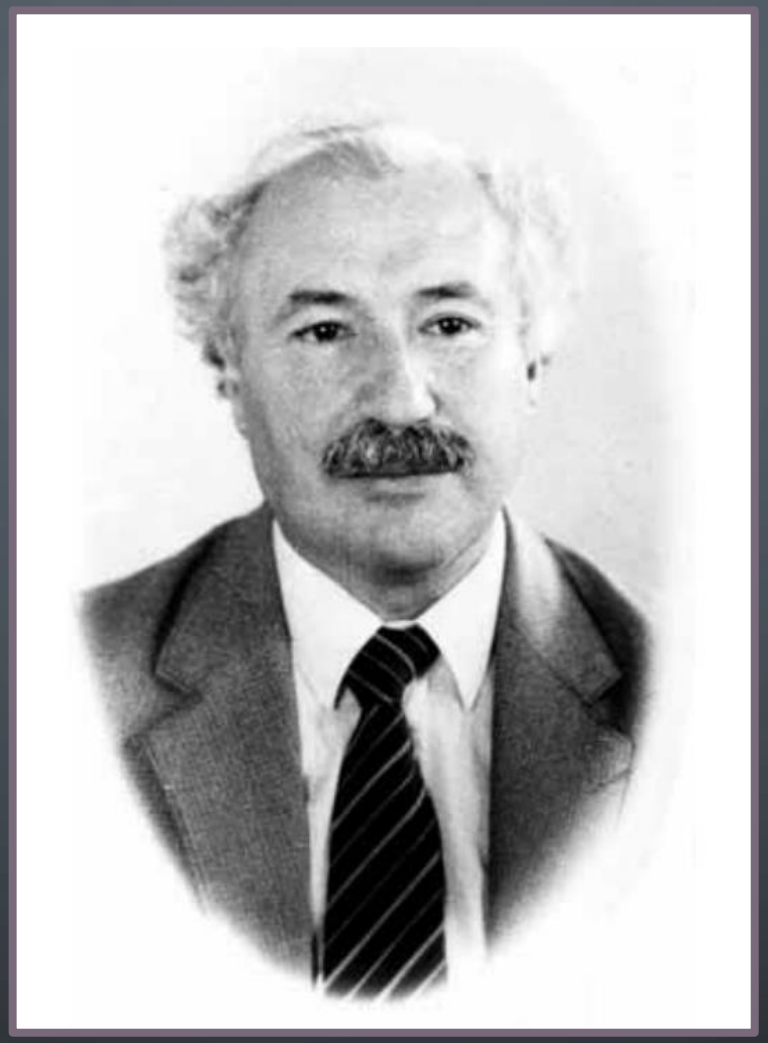
ТЕОРІЯ СУБГРАДІЄНТНИХ МЕТОДІВ Н.З. ШОРА. ПРИКЛАДИ ЗАСТОСУВАННЯ

КОРАБЛЬОВ МИКОЛА[1,2] M.M.KORABLOV@GMAIL.COM

[1]ІНСТИТУТ КІБЕРНЕТИКИ ІМЕНІ В.М. ГЛУШКОВА НАН УКРАЇНИ

[2]КИЇВСЬКИЙ АКАДЕМІЧНИЙ УНІВЕРСИТЕТ

5 ЛИПНЯ, 2024, КИЇВ, УКРАЇНА



Наум Зуселевич Шор (1937 – 2006)

Засновник Київської школи негладкої оптимізації

(автор 10 монографій та >200 статей)

ЗМІСТ:

ЗМІСТ:

- Застосування для навчання з учителем – регресія
  - Короткий огляд Методу Найменших Квадратів та чому він настільки популярний;
  - Переваги Методу Найменших Модулів;
  - Приклад: негладка регресійна модель для пошуку дефектів у регулярних 3D структурах;
- Застосування для навчання з учителем – класифікація
  - Короткий огляд Бінарної Лінійної Класифікації та Методу Опорних Векторів;
  - Інколи помилки класифікації заборонені: задача перевірки двох множин точок на лінійну роздільність;
  - Приклад: перевірка на лінійну роздільність датасету “пила” із зазором $\varepsilon \to +0$;
- Заключне слово

УЗАГАЛЬНЕННЯ ГРАДІЄНТНОГО СПУСКУ З ВИКОРИСТАННЯМ СУБГРАДІЄНТА

МОТИВАЦІЯ ДЛЯ НЕГЛАДКОЇ ОПТИМІЗАЦІЇ

- Де виникають задачі негладкої оптимізації [1]:
  - Задачі математичного програмування великої розмірності з блочною структурою
  - Міні-Максні задачі виду $\min\limits_{x \in D} \max\limits_{i \in 1,\dots,n} f_i(x)$
  - Задачі нелінійного програмування, для розв'язання яких використовується метод негладких штрафних функцій
  - Задачі оптимального керування з неперервним та дискретним часом
  - Задачі дискретного програмування чи задачі дискретно-неперервного типу
- Також, із суто обчислювальної точки зору, немає як такої чіткої різниці між гладкими та негладкими функціями: гладка функція, що має сильно осцилюючий градієнт, в певному сенсі "схожа" на негладку функцію.

ДВА НАПРЯМКИ ДОСЛІДЖЕНЬ

- Дослідження в напрямку розв'язання спеціальних класів задач, що потребують мінімізації негладкої цільової функції, спеціальна структура якої відома заздалегідь (наприклад міні-максні задачі);
- Дослідження в напрямку розробки загальних алгоритмів, які здатні розв'язувати велику кількість типів негладких задач, не потребуючи апріорного знання структури цільової функції:
  - Методи відсікаючих гіперплощин;
  - Методи узагальненого градієнтного спуску.

ГРАДІЄНТНИЙ СПУСК: СПРОБА УЗАГАЛЬНЕННЯ

- Метод градієнтного спуску для мінімізації опуклої гладкої функції $f(x)$:

$$x_{k+1} = x_k - h_k \cdot \nabla f(x_k), k = 0,1,2, \ldots$$

x_0 – початкова точка

$\nabla f(x_k)$ – градієнт цільової функції, обчислений в точці x_k

h_k – кроковий множник на ітерації k (learning rate)

- Було б добре взяти цю ідею за основу для розробки методів мінімізації опуклих функцій загального виду, як гладких, так і негладких.
- Однак виявляється, що це не так просто зробити…

ГРАДІЄНТНИЙ СПУСК: СПРОБА УЗАГАЛЬНЕННЯ

- Намагаючись узагальнити методи градієнтного спуску для функцій з розривним градієнтом, виникають дві великі проблеми [1]:
  - Визначення аналогу градієнта в тих точках, де звичайні похідні не існують, таким чином, щоб його було зручно застосовувати на практиці;
  - Визначення нових способів пошуку напрямку спуску та крокового множника, адже звичайні способи не працюють коректно при мінімізації негладких функцій.

ВИЗНАЧЕННЯ АНАЛОГУ ГРАДІЄНТА

- Таке узагальнення існує – субградієнт $g_f(x)$ функції $f(x)$;
- Однак, за означенням може існувати безліч субградієнтів функції в деякій точці $\hat{x}$, які утворюють субградієнтну множину:

$$G_f(\hat{x}) = \left\{ g_f(\hat{x}) : f(x) - f(\hat{x}) \geq \left\langle g_f(\hat{x}), x - \hat{x} \right\rangle \forall x \in \mathbb{R}^n \right\}$$

- Приклад:

$$f(x) = |x|; \quad G_f(0) = [-1; 1]$$

ВИЗНАЧЕННЯ АНАЛОГУ ГРАДІЄНТА

- Поняття субградієнту є чудовим теоретичним узагальненням поняття градієнта для негладких функцій;
- Але воно не може бути використане на практиці, в основному - саме через неєдиність визначення: якщо існує безліч субградієнтів функції в деякій точці $\hat{x}$, то який з них обрати напрямком спуску?

ВИЗНАЧЕННЯ АНАЛОГУ ГРАДІЄНТА

- Підхід Н.З. Шора [1]:
  - Зосередимо увагу на деякому класі негладких функцій, такому, що не включає "дивні" негладкі функції, що майже ніколи не зустрічаються на практиці, та визначимо аналог градієнта для цього класу функцій ⇒ майже-градієнт Шора!
  - За означенням, для всіх опуклих функцій $f(x)$ існує майже-градієнт та $\forall x \in \mathbb{R}^n$ майже-градієнт є елементом субградієнтної множини.
- Та чи буде він задавати напрямок спуску?

- **N.B.** надалі термін “субградієнт” буде означати “майже-градієнт”

ПОШУК НАПРЯМКУ СПУСКУ

- Підхід Н.З. Шора [1]:
  - Нехай $f(x) \in \mathbb{R}$ опукла функція $\forall x \in \mathbb{R}^n$, і нехай $x^* \in X^*$ є елементом множини

    $$X^* = \left\{x^* : f(x^*) = \min_{x \in \mathbb{R}^n} f(x)\right\}$$

  - За означенням субградієнта в точці $\hat{x}$:

    $$f(x) - f(\hat{x}) \geq \left\langle g_f(\hat{x}), x - \hat{x} \right\rangle \quad \forall x \in \mathbb{R}^n$$

  - Якщо $f(x) < f(\hat{x})$, тоді:

    $$\left\langle -g_f(\hat{x}), x - \hat{x} \right\rangle > 0 \quad \forall x \in \mathbb{R}^n$$

ПОШУК НАПРЯМКУ СПУСКУ

- Геометричний сенс $\langle -g_f(\hat{x}), x - \hat{x} \rangle > 0 \quad \forall x \in \mathbb{R}^n$:
  - Антисубградієнт $-g_f(\hat{x})$ в точці $\hat{x}$ утворює гострий кут з будь-яким напрямком $x - \hat{x}$ з точки $\hat{x}$ до такої точки $x \in \mathbb{R}^n$, де значення $f(\cdot)$ менше;
  - Отже, якщо $X^* \neq \emptyset$ і $\hat{x} \notin X^*$, тоді рух з точки $\hat{x}$ в напрямку $-g_f(\hat{x})$ з достатньо малим кроком зменшить відстань до X^*!
- Цей простий факт є головною ідеєю субградієнтних методів спуску для мінімізації негладких функцій!

МЕТОД СУБГРАДІЄНТНОГО СПУСКУ [2]

- a.k.a. узагальнений метод градієнтного спуску, це субградієнтна процедура для мінімізації опуклих функцій

$$x_{k+1} = x_k - h_k \cdot \frac{g_f(x_k)}{\|g_f(x_k)\|}, k = 0,1,2, \dots$$

x_0 – початкова точка

$g_f(x_k)$– суградієнт цільової функції, обчислений в точці x_k

h_k – кроковий множник на ітерації k

- Метод зовні схожий на гладкий градієнтний спуск, проте працює інакше;
- Також ще потрібно задати крокові множники h_k…

МЕТОД СУБГРАДІЄНТНОГО СПУСКУ

- Теорема:

  Нехай $f(x)$ опукла функція із обмеженою множиною X^*. Задамо послідовність $\{h_k\}_{k=0}^{\infty}$ таку, що $h_k > 0, \lim\limits_{k\to\infty} h_k = 0, \sum_{k=0}^{\infty} h_k = +\infty$;

  Тоді $\forall x_0 \in \mathbb{R}^n$ послідовність $\{x_k\}_{k=0}^{\infty}$, отримана за допомогою методу субградієнтного спуску, має одну з двох властивостей:

  - $\exists k = k^* \; s.t. \; x_{k^*} \in X^*$
  - $\lim\limits_{k\to\infty} \rho_k = 0, \lim\limits_{k\to\infty} f(x_k) = \min\limits_{x\in\mathbb{R}^n} f(x)$, де $\rho_k = \min\limits_{x\in X^*} \|x_k - x\|$

- Можна інакше задати h_k? Так! (крок Поляка, адаптивний крок Шора,...)

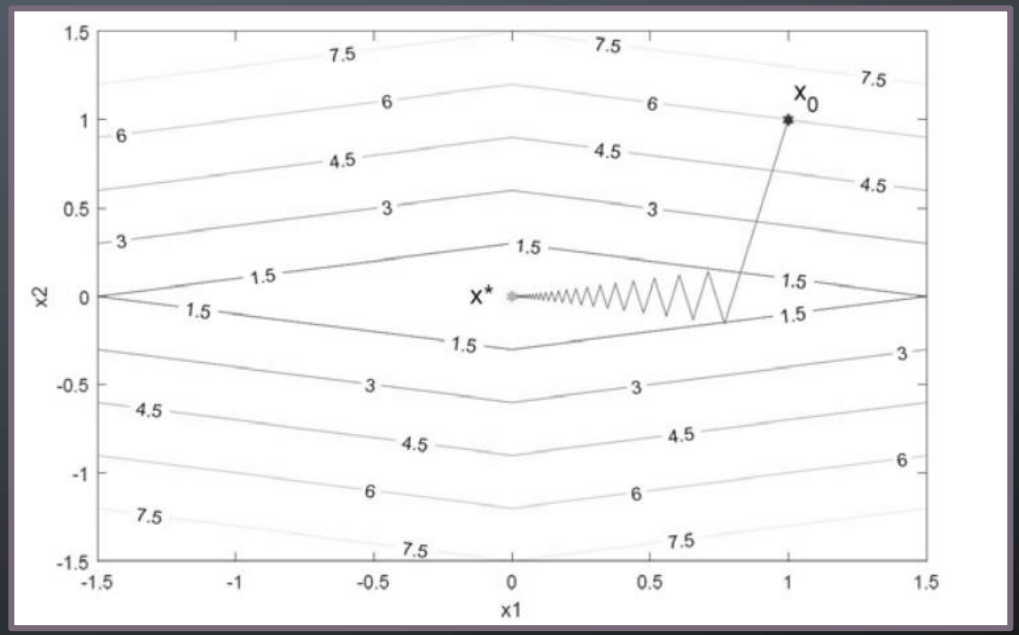
ПОКРАЩЕННЯ ЗБІЖНОСТІ ДЛЯ ЯРУЖНИХ ФУНКЦІЙ ЗА ДОПОМОГОЮ РОЗТЯГУ ПРОСТОРУ

ЯРУЖНА ФУНКЦІЯ

- Функція декількох дійсних змінних, графік якої біля точки мінімума має форму яру. Такі функції значно важче мінімізовувати.
- Приклад:

$$f(x_1, x_2) = |x_1| + 5|x_2|$$

$$x_0 = (1,1), x^* = (0,0)$$



- Добре б допомогти методу збігатися швидше… Розтягнемо простір!

РОЗТЯГ ПРОСТОРУ – ГОЛОВНА ІДЕЯ [1]

- Давайте зробимо заміну змінних під час k-ої ітерації субградієнтного методу:

$$y = A_k x \Rightarrow x = B_k y, B_k = A_k^{-1}$$

- Як відомо, для субградієнта опуклої функції $f(x)$ в точці x_k виконується наступна нерівність:

$$f(x) \geq f(x_k) + \langle g_f(x_k), x - x_k \rangle \quad \forall x \in \mathbb{R}^n$$

РОЗТЯГ ПРОСТОРУ – ГОЛОВНА ІДЕЯ

- Підставивши $x = B_k y$, отримаємо:

$$\varphi(y) \geq \varphi(y_k) + \langle B_k^T g_f(x_k), y - y_k \rangle \; \forall y \in \mathbb{R}^n$$

- Як бачимо, $g_\varphi(y_k) = B_k^T g_f(x_k)$ задовольняє нерівності:

$$\varphi(y) \geq \varphi(y_k) + \langle g_\varphi(y_k), y - y_k \rangle \; \forall y \in \mathbb{R}^n$$

що означає, що $g_\varphi(y_k)$ є субградієнтом опуклої функції $\varphi(y) = f(B_k y)$ в точці $y_k = A_k x_k$ перетвореного простору змінних $y = A_k x$

РОЗТЯГ ПРОСТОРУ – ГОЛОВНА ІДЕЯ

- Застосуємо субградієнтну процедуру для мінімізації $\varphi(y)$;
- У перетвореному просторі змінни $y = A_k x$ вона має вигляд:

$$y_{k+1} = y_k - h_k \frac{g_\varphi(y_k)}{\|g_\varphi(y_k)\|} = y_k - h_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}$$

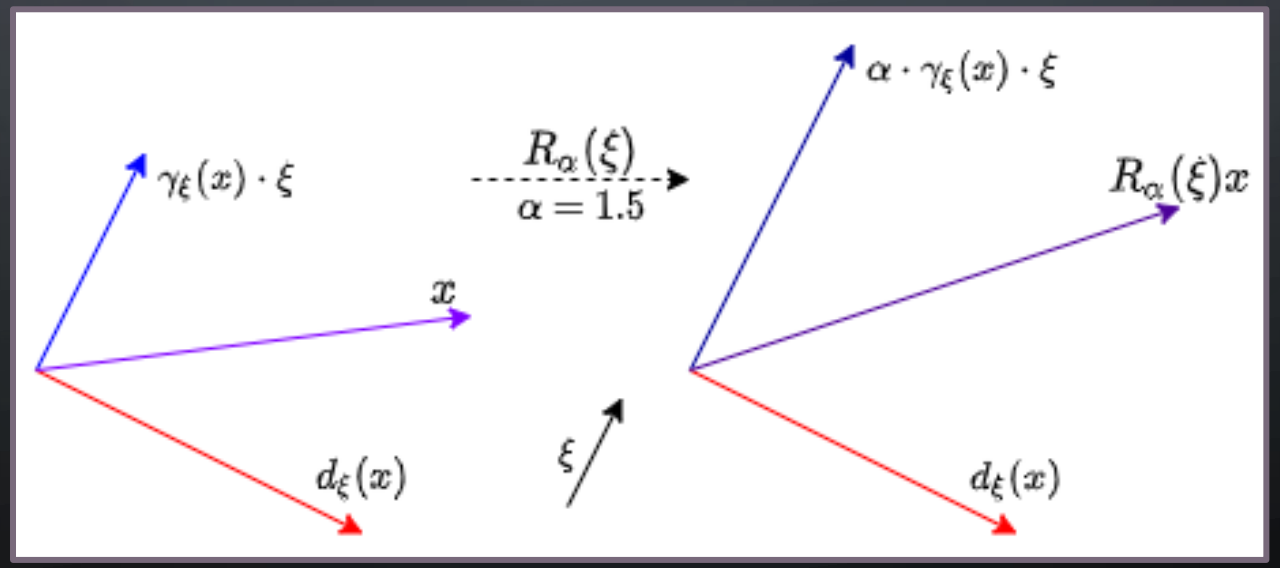
- І в початковому просторі $x = B_k y$ вона набуде вигляду:

$$x_{k+1} = B_k y_{k+1} = B_k y_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|} = x_k - h_k B_k \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}$$

- Залишилося лише визначити спосіб ітеративного оновлення B_k
- Питання в тому, які перетворення будуть корисними?

ОПЕРАТОР РОЗТЯГУ ПРОСТОРУ ШОРА [1]

- Зафіксуємо вектор $\xi \in \mathbb{R}^n, \|\xi\|_2 = 1$ та число $\alpha > 0$
- Тоді $\forall x \in \mathbb{R}^n$ маємо $x = \gamma_\xi(x) \cdot \xi + d_\xi(x)$, де $\langle \xi, d_\xi(x) \rangle = 0$
- Оператор розтягу простору $\mathbb{R}^n$ в напрямку ξ з коефіцієнтом α визначається як $R_\alpha(\xi)x = \alpha \cdot \gamma_\xi(x) \cdot \xi + d_\xi(x), \forall x \in \mathbb{R}^n$



ОПЕРАТОР РОЗТЯГУ ПРОСТОРУ ШОРА

- Деякі властивості оператора $R_\alpha(\xi)$:
  - Лінійний, симетричний;
  - Матрична форма: $R_\alpha(\xi) = I + (\alpha - 1)\xi\xi^T$;
  - $R_{\alpha\beta}(\xi) = R_\alpha(\xi)R_\beta(\xi)$;
  - $R_\alpha(\xi)R_{\frac{1}{\alpha}}(\xi) = I$;
  - $R_0(\xi)$ є проектором на $\{\xi\}^\perp$

- Тож, який напрямок ξ обрати?

МЕТОД ЕЛІПСОЇДІВ ДЛЯ ОПУКЛИХ ФУНКЦІЙ [3]

- Отриманий використовуючи розтяг простору в напрямку субградієнта із певним коефіцієнтом розтягу простору та кроковими множниками:

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad \xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k = \frac{1}{n+1} r_k, \quad k = 0,1, \dots$$

$$B_{k+1} = B_k R_{\beta_k}(\xi_k), \quad \beta_k = \sqrt{\frac{n-1}{n+1}}, \quad r_{k+1} = \frac{n}{\sqrt{n^2-1}} r_k$$

x_0 – початкова точка;

$r_0 > 0$ – радіус початкової локалізуючої кулі, $\|x_0 - x^*\| \leq r_0$;

$B_0 = I_n$ – матриця початкового оберненого перетворення простору

МЕТОД ЕЛІПСОЇДІВ ДЛЯ ОПУКЛИХ ФУНКЦІЙ

- Теорема:

  Нехай $\{x_k\}_{k=0}^{\infty}$ є послідовністю точок згенерованою методом еліпсоїдів при мінімізації опуклої функції. Тоді відношення об'ємів локалізуючих еліпсоїдів $\mathcal{E}_k$ та $\mathcal{E}_{k+1}$ отриманих на ітераціях k та $k+1$ не залежить від k та дорівнює

$$q_n = \frac{vol(\mathcal{E}_{k+1})}{vol(\mathcal{E}_k)} = \frac{n}{n+1}\left(\frac{n}{\sqrt{n^2-1}}\right)^{n-1} < e^{\frac{1}{2(n+1)}} < 1.$$

  Більш того, $x^* \in \mathcal{E}_k \ \forall k = 0,1,\dots,k^*$

МЕТОД ЕЛІПСОЇДІВ ДЛЯ ОПУКЛИХ ФУНКЦІЙ

- Цікаві факти:
  - Метод еліпсоїдів був створений незалежно Немировським та Юдіним [4], використовуючи зовсім інший підхід, в той час як Шор створив його як вид методу субградієнтного спуску із розтягом простору
  - Метод еліпсоїдів був використаний Хачияном [5] для побудови та обґрунтування першого поліноміального алгоритму для задач лінійного програмування з раціональними коефіцієнтами, тим самим спростувавши NP-складність цієї задачі!
- Однак, ви можете натрапити на статті в яких стверджується, що метод еліпсоїдів незастосовний на практиці та не працює навіть для функцій 2-5 змінних. В чому річ?
  - Цей алгоритм має дві форми: обчислювально стійку B-форму (описану вище), та нестійку H-форму. Багато людей десятиліттями використовували не ту форму!

R-АЛГОРИТМ ШОРА [6]

- Отриманий використовуючи розтяг простору в напрямку різниці двох послідовних субградієнтів:

$$x_{k+1} = x_k - h_k B_k \xi_k, \quad B_{k+1} = B_k R_{\beta_k}(\eta_k), k = 0,1, \dots$$

$$\xi_k = \frac{B_k^T g_f(x_k)}{\|B_k^T g_f(x_k)\|}, \quad h_k \geq h_k^* = \underset{h \geq 0}{argmin}\, f(x_k - h B_k \xi_k)$$

$$\eta_k = \frac{B_k^T r_k}{\|B_k^T r_k\|}, \quad r_k = g_f(x_{k+1}) - g_f(x_k), \quad \beta_k = \frac{1}{\alpha} < 1$$

- Дуже потужний інструмент, дозволяє розв’язувати багато складних оптимізаційних задач
- Однак його збіжність доведена лише в часткових випадках

ЗАСТОСУВАННЯ ДЛЯ НАВЧАННЯ З УЧИТЕЛЕМ – РЕГРЕСІЯ

КОРОТКИЙ ОГЛЯД: РЕГРЕСІЯ [7]

- Нехай $\left\{\left(x_1^{(j)}, \dots, x_d^{(j)}, y^{(j)}\right) \in \mathbb{R}^{d+1} : j = \overline{1,n}\right\}$ це датасет розміру n, де для кожного вимірювання $j = \overline{1,n}$ спостережні величини $y^{(j)}$ якось залежні від значень d факторів $x_1^{(j)}, \dots, x_d^{(j)}$.
- Задача регресії полягає у використанні наявних даних для побудови моделі яка:
  - Досить добре описує залежності між $y^{(j)}$ та факторами $x_i^{(j)}$;
  - Підходить для прогнозування значень y, що відповідають новим, “небаченим” факторам x_i;

КОРОТКИЙ ОГЛЯД: РЕГРЕСІЯ

- В найпростішому випадку припускається що така залежність лінійна по параметрам моделі w, тобто:

$$y = f(x_1, \dots, x_d) = \sum_{k=1}^{m} w_k \psi_k(x_1, \dots, x_d)$$

де $\psi_k, k = \overline{1, m}$ деякі базисні функції

- Щоб знайти підходящі значення параметрів w, тобто побудувати модель яка достатньо добре апроксимує залежності між y та x на основі наявних даних, потрібно:
  - Обрати базисні функції (самі фактори, поліноми, експоненти, …);
  - Мінімізувати похибки $\varepsilon^{(j)} = y^{(j)} - \hat{y}^{(j)} = y^{(j)} - f\left(x_1^{(j)}, \dots, x_d^{(j)}\right), j = \overline{1, n}$;

КОРОТКИЙ ОГЛЯД: НАЙМЕНШІ КВАДРАТИ [7]

- Найпопулярніший спосіб розв’язання задач лінійної регресії:
  - Виводиться статистично методом максимальної правдоподібності з припущень про те що фактори незалежні та похибки мають розподіл $N(0, \sigma^2 I_n)$;
  - Точний розв’язок можна отримати ортогональним проектуванням у Гільбертовому просторі і такий розв’язок має гарні властивості за теоремою Гауса-Маркова;
- Зазвичай задачу регресії розглядають з точки зору функціональної апроксимації, розв’язуючи задачу опуклої гладкої оптимізації:

$$\min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left(\varepsilon^{(j)}\right)^2 = \min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left(y^{(j)} - \hat{y}^{(j)}\right)^2$$

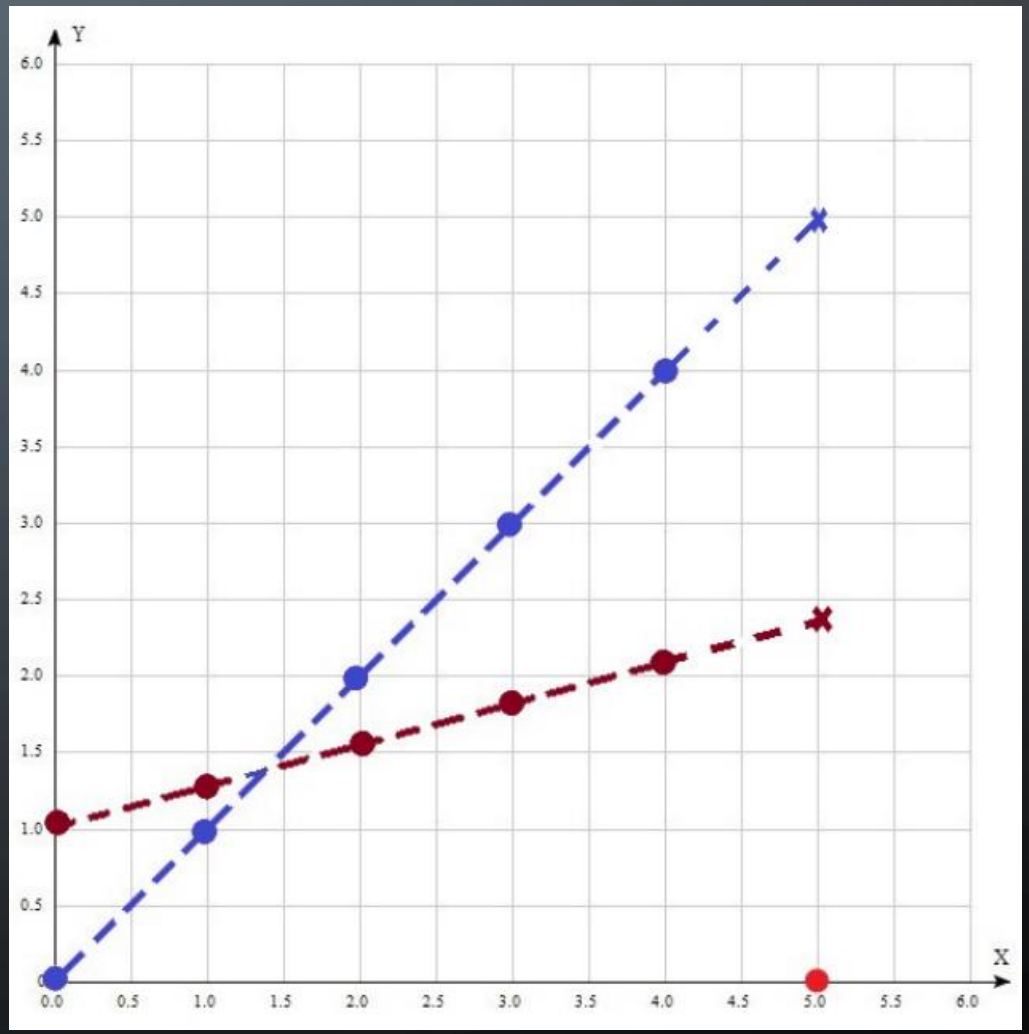
- Це стало “загальноприйнятим” підходом, проте на практиці з ним зазвичай виникає багато проблем

РЕГРЕСІЯ НА ПРАКТИЦІ – ДАНІ З ВИКИДАМИ

- “Дитячий” приклад:
  - Одновимірний, шість точок

x	0	1	2	3	4	5
y	0	1	2	3	4	0

  - Червона лінія – модель отримана Найменшими Квадратами та її прогноз $\hat{y}$
  - Синя лінія – справжня залежність



ПРОБЛЕМИ ТА ЇХ РІШЕННЯ [7]

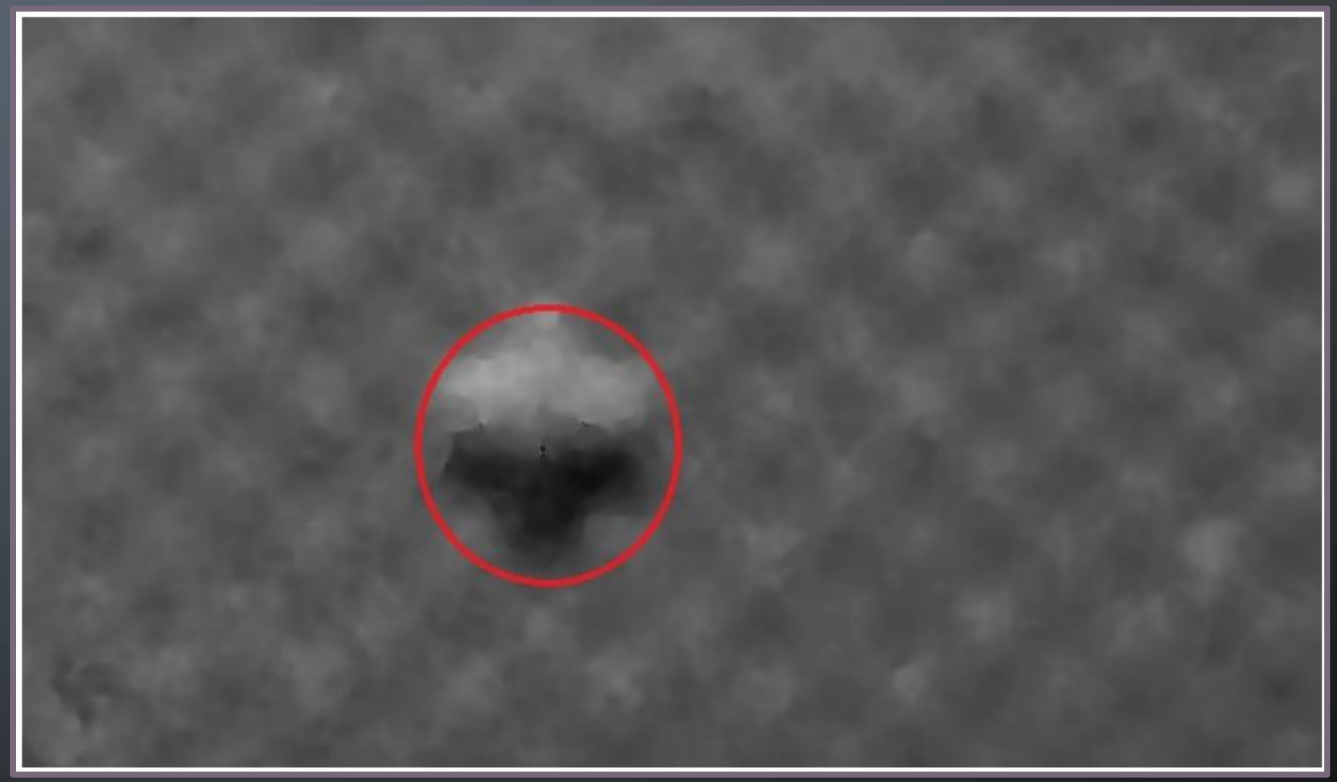
- Більшість проблем на практиці виникають коли:
  - Працюємо з датасетами малого розміру (медичні застосування);
  - Працюємо з датасетами з багатьма викидами або залежними факторами;
- Один популярний підхід - регуляризація:
  - Гребенева: $\min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left(y^{(j)} - \hat{y}^{(j)}\right)^2 + \lambda \sum_{k=1}^{m} w_k^2$
  - Лассо: $\min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left(y^{(j)} - \hat{y}^{(j)}\right)^2 + \lambda \sum_{k=1}^{m} |w_k|$
- Є ще один статистично обґрунтований спосіб – Найменші Модулі:

$$\min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left|\varepsilon^{(j)}\right| = \min_{w \in \mathbb{R}^m} \sum_{j=1}^{n} \left|y^{(j)} - \hat{y}^{(j)}\right|$$

- Проте це потребує безпосереднього розв’язання негладкої задачі…

ПРИКЛАД: ДЕФЕКТИ В РЕГУЛЯРНИХ 3D СТРУКТУРАХ

- Спільний дослідницький проект Інституту кібернетики імені В.М. Глушкова та Інституту електрозварювання імені Є.О. Патона [8];
- Короткий опис: Розробити програмне забезпечення для автоматичного неруйнуючого контролю якості тонкостінних багатошарових композитних матеріалів



РЕГУЛЯРНІ 3D СТРУКТУРИ

- Трійка $\{A;\ u;\ v\}$ називається регулярною 3D структурою, якщо $A \in \mathbb{R}^{m \times n}, u \in \mathbb{R}^m, v \in \mathbb{R}^n$ та $\forall i = \overline{1,m}, \forall j = \overline{1,n}$: $a_{ij} = u_i + v_j$;
- Елементарним дефектом в регулярній 3D структурі $\{A;\ u;\ v\}$ є така пара індексів (i,j) що $a_{ij} \neq u_i + v_j$;

- Нехай матриця A є представленням зображення тієї ділянки яка перевіряється на дефекти. Задача знайти такі параметри u та v щоб $u_i + v_j$ мали найменші відхилення від відповідних значень a_{ij}.

РЕГУЛЯРНІ 3D СТРУКТУРИ

- Гладка задача мінімізації (Найменші Квадрати):

$$\min_{u\in\mathbb{R}^m, v\in\mathbb{R}^n} \sum_{i=1}^{m}\sum_{j=1}^{n}\left(a_{ij}-u_i-v_j\right)^2$$

- Негладка задача мінімізації (Найменші Модулі):

$$\min_{u\in\mathbb{R}^m, v\in\mathbb{R}^n} \sum_{i=1}^{m}\sum_{j=1}^{n}\left|a_{ij}-u_i-v_j\right|$$

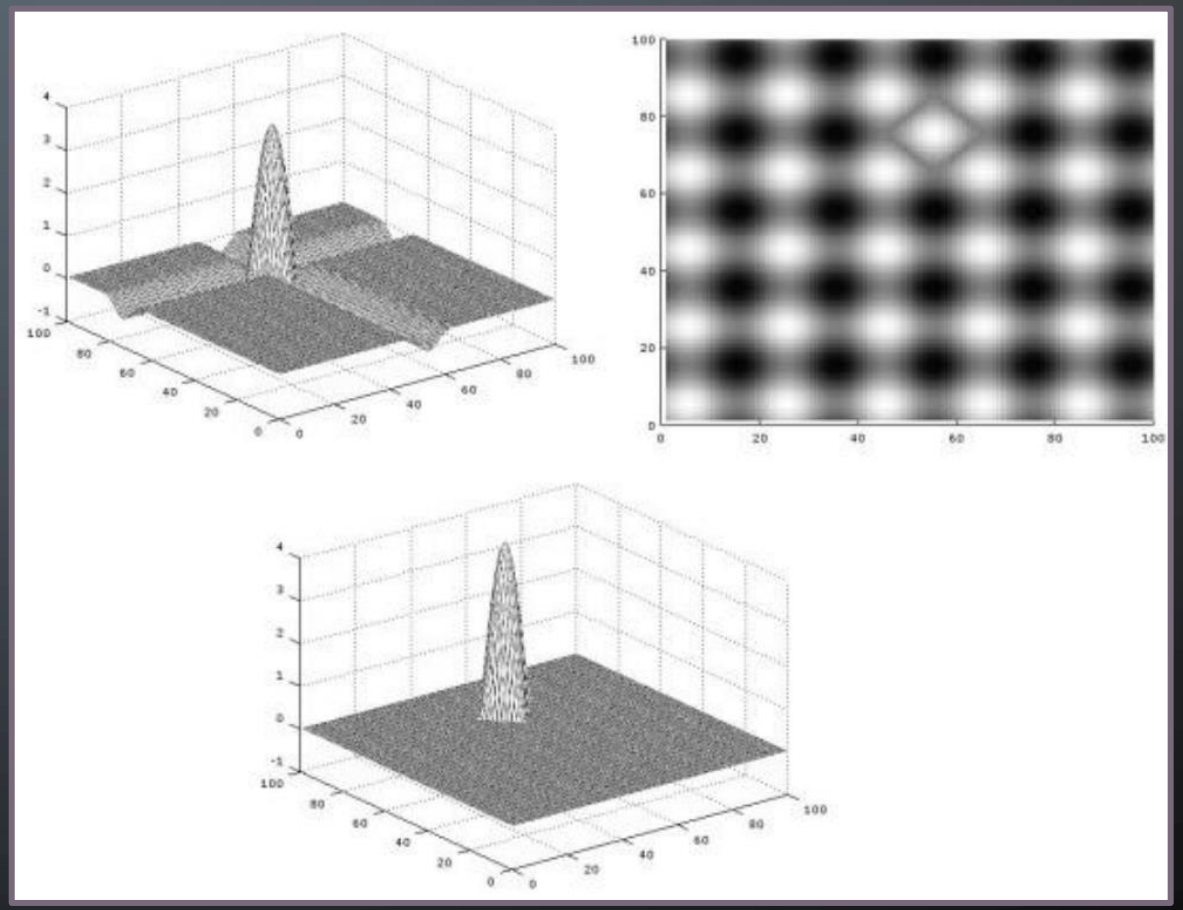
- Обидві отримані моделі були порівняні на різних зображеннях, відповідні задачі оптимізації були розв’язані r-алгоритмом

3D СТРУКТУРА З 1 ДЕФЕКТОМ

Визначення дефекту Найменшими Квадратами (згори зліва)

Саме зображення (згори справа)

Визначення дефекту Найменшими Модулями (знизу)

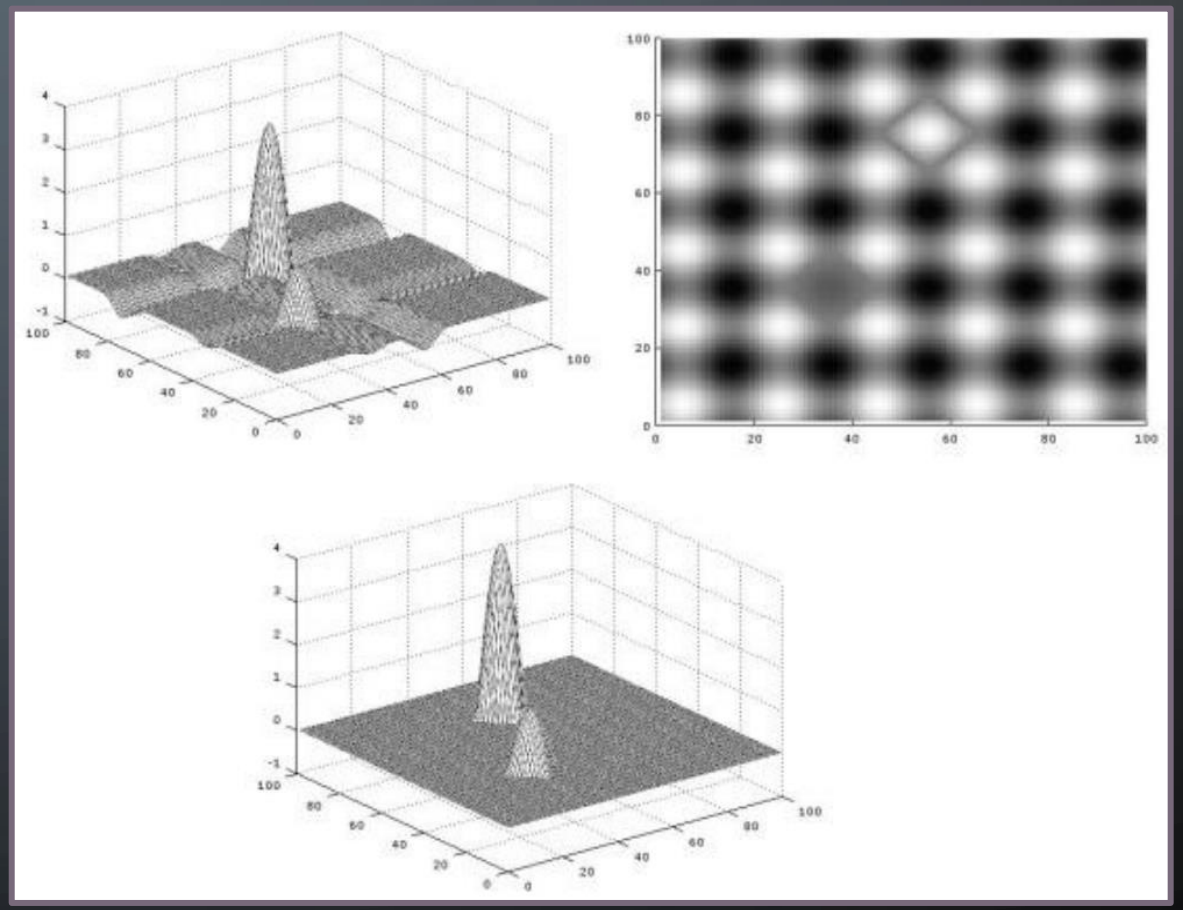


3D СТРУКТУРА З 2 ДЕФЕКТАМИ

Визначення дефектів Найменшими Квадратами (згори зліва)

Саме зображення (згори справа)

Визначення дефектів Найменшими Модулями (знизу)



ЗАСТОСУВАННЯ ДЛЯ НАВЧАННЯ З УЧИТЕЛЕМ – КЛАСИФІКАЦІЯ

КОРОТКИЙ ОГЛЯД: БІНАРНА КЛАСИФІКАЦІЯ [9]

- Нехай $\{(x_i, y_i), x_i \in \mathbb{R}^m, y_i \in \{\pm 1\}: i = \overline{1,n}\}$ це датасет розміру n, де для кожного вимірювання $i = \overline{1,n}$ елементи x_i належать до одного з двох класів, що позначаються значенням y_i.
- Задача лінійної класифікації полягає у використанні наявних даних для пошуку такої розділяючої два класи гіперплощини $\langle w, x \rangle + b = 0$, яка має найбільший зазор:

$$\max_{w \in \mathbb{R}^m, b \in \mathbb{R}, r > 0} r$$

$$y_i \cdot (\langle w, x_i \rangle + b) \geq r, \ i = \overline{1,n}$$

$$\|w\| = 1$$

КОРОТКИЙ ОГЛЯД: МЕТОД ОПОРНИХ ВЕКТОРІВ

- Без втрати загальності, можна прийняти $r = 1$ та розглядати такі моделі [9]:
- Hard SVM (забороняє помилку класифікації):

$$\min_{w \in \mathbb{R}^m, b \in \mathbb{R}} \frac{1}{2} \|w\|^2$$

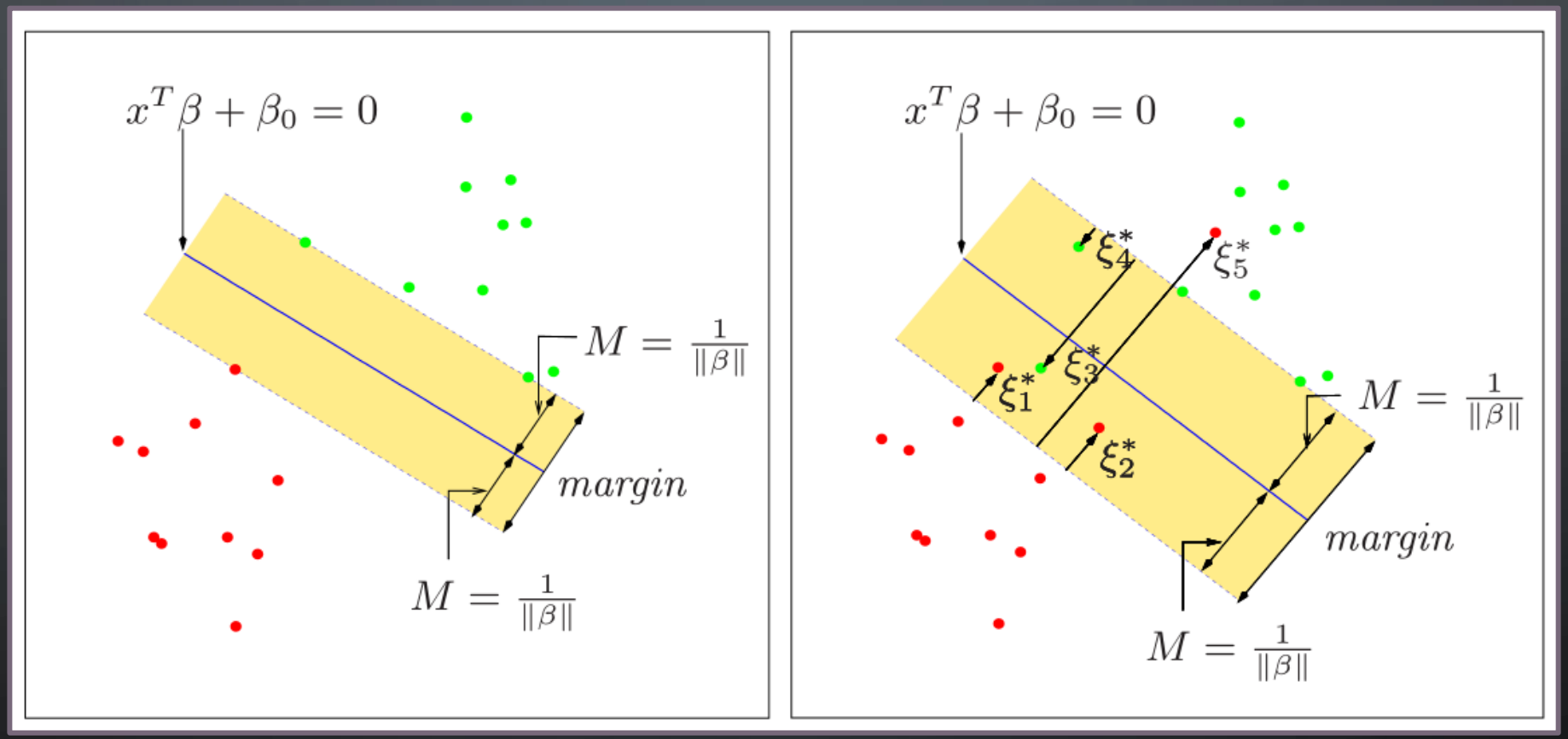
$$y_i \cdot (\langle w, x_i \rangle + b) \geq 1, \quad i = \overline{1, n}$$

- Soft SVM (враховує і допускає можливу помилку класифікації):

$$\min_{w \in \mathbb{R}^m, b \in \mathbb{R}} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i$$

$$y_i \cdot (\langle w, x_i \rangle + b) \geq 1 - \xi_i, \quad i = \overline{1, n}$$
$$\xi_i \geq 0, \qquad i = \overline{1, n}$$

ПОРІВНЯННЯ HARD SVM ТА SOFT SVM

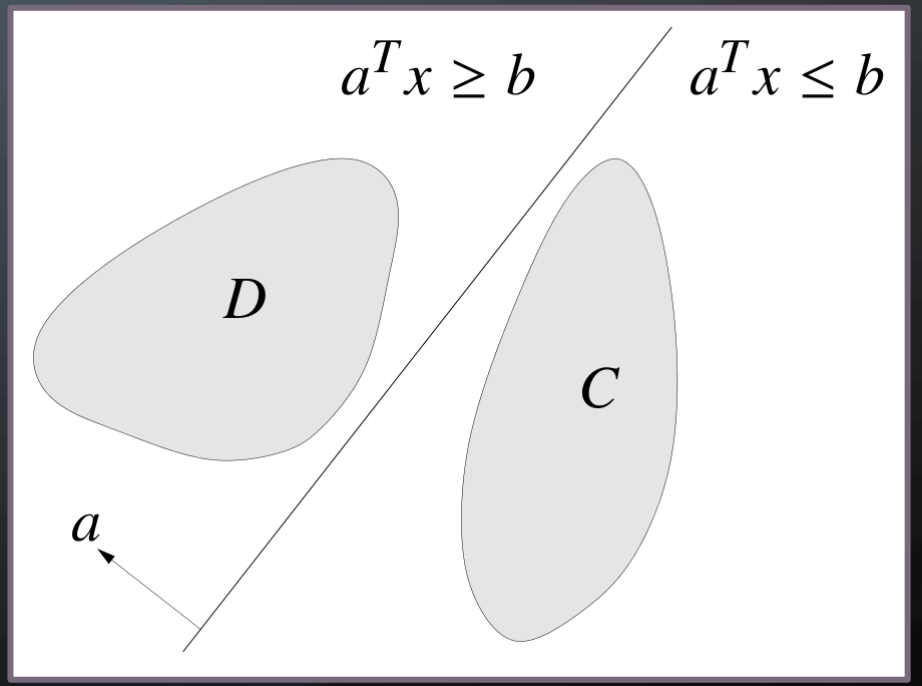


Hard SVM класифікація (зліва), Soft SVM класифікація (справа) [7]

(в наших позначеннях $M = r, \beta = w, \beta_0 = b$)

ЛІНІЙНА РОЗДІЛЬНІСТЬ МНОЖИН

- Щоб застосовувати лінійну класифікацію та очікувати коректний результат – спочатку потрібно впевнитися, що класи лінійно роздільні.
- Це відома теоретична задача з математичного програмування [10], і для опуклих множин існують гарні теореми.
- Приклад:
  - Гіперплощина $\{x: \langle a, x \rangle = b\}$ розділяє множини C та D



ЛІНІЙНА РОЗДІЛЬНІСТЬ МНОЖИН

- Проте на практиці ми не знаємо, чи дві «хмари точок» справді утворюють опуклі множини, що не перетинаються – і тому є лінійно роздільними.
- Тому виникає потреба у чисельному алгоритмі який буде здатний перевіряти лінійну роздільність заданого розміченого датасету.
- В багатьох ситуаціях, Hard SVM справиться із цією задачею:
  - Якщо вдається розв'язати задачу із нульовою похибкою класифікації навчальних даних, тоді лінійне розділення існує
  - Якщо ж Hard SVM не дає розв'язку, то вважають що лінійного розділення не існує
- Проте, такий підхід не надійний!

ЛІНІЙНИЙ КЛАСИФІКАТОР З МАКС. ЗАЗОРОМ [11]

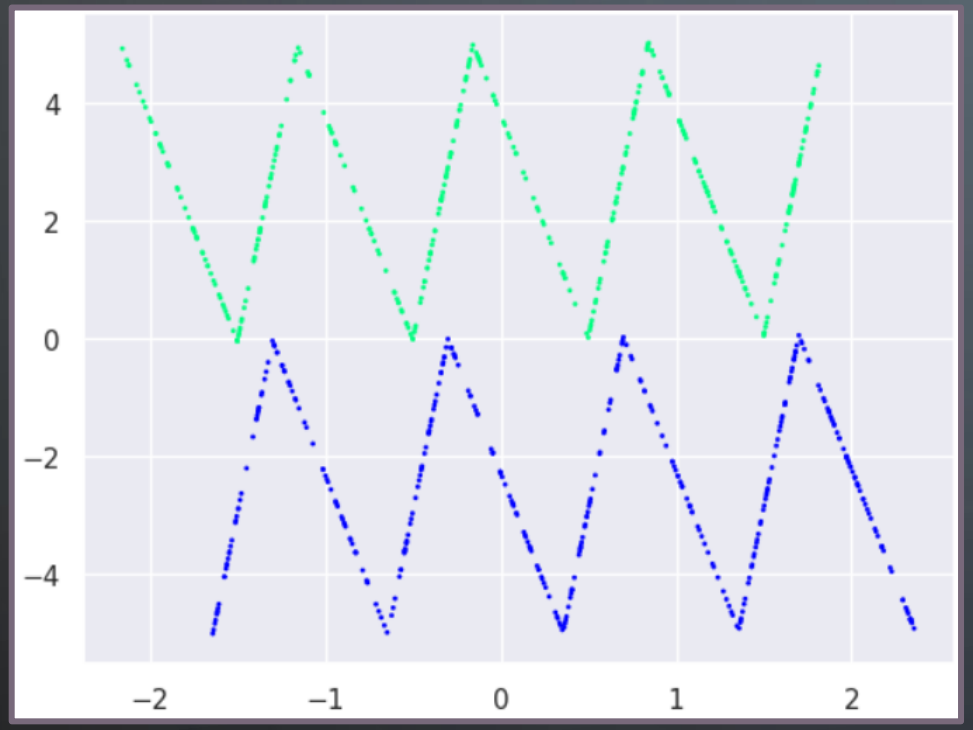
- Задача: знайти таку гіперплощину $\langle w, x \rangle + b = 0$, щоб жодної точки не містилося між $\langle w, x \rangle + b = 1$ та $\langle w, x \rangle + b = -1$:

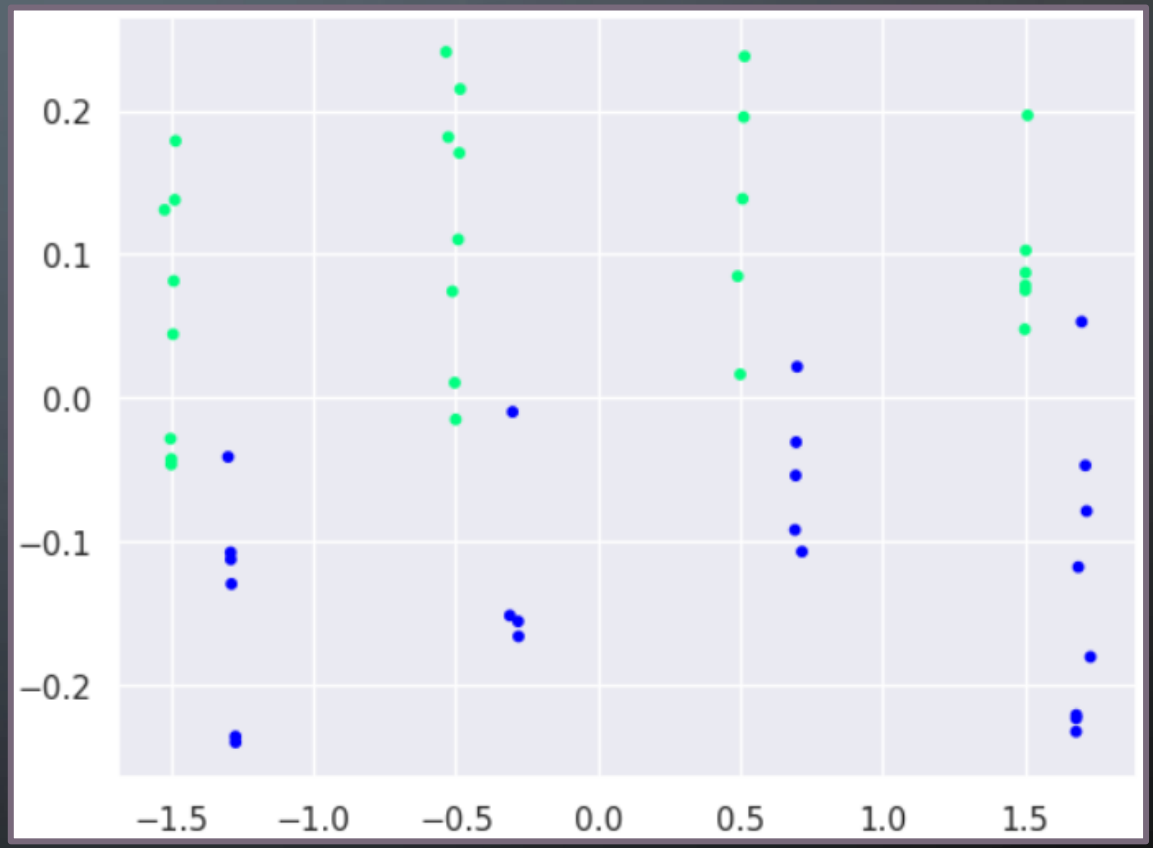
$$\min_{w \in \mathbb{R}^m, b \in \mathbb{R}} \left\{ \max_{i \in 1,n} -y_i(\langle w, x_i \rangle + b) \right\}$$
$$\|w\|^2 \leq 1$$

- Ця задача негладкої оптимізації з обмеженнями може бути розв'язана використовуючи метод еліпсоїдів та негладку штрафну функцію [12]:

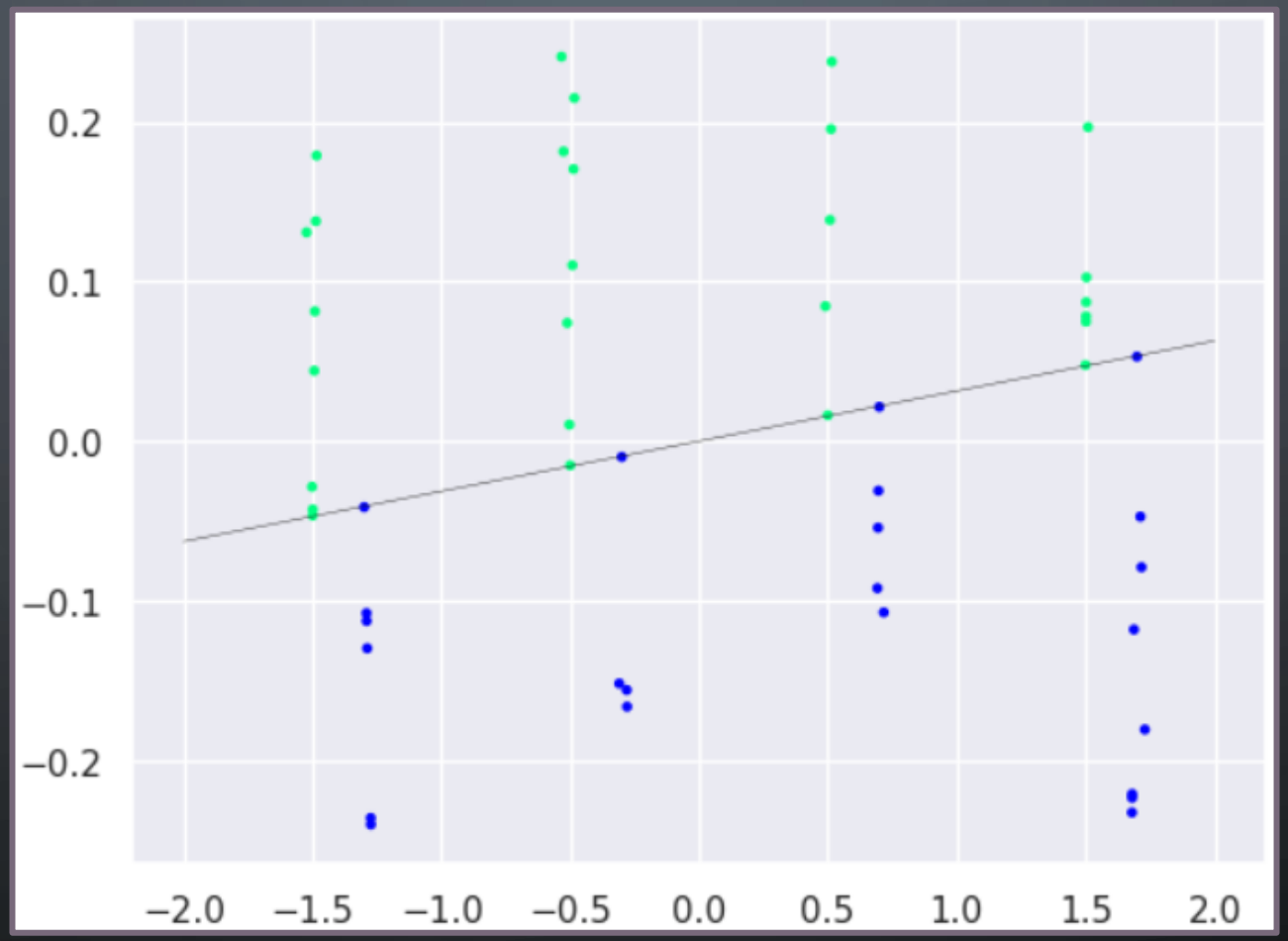
$$\min_{w \in \mathbb{R}^m, b \in \mathbb{R}} \left\{ \max_{i=1,..,n} \{-y_i(\langle w, x_i \rangle + b)\} + P \cdot max\left\{0, \sum_{j=1}^{m} w_j^2 - 1\right\} \right\}$$

ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-3}$



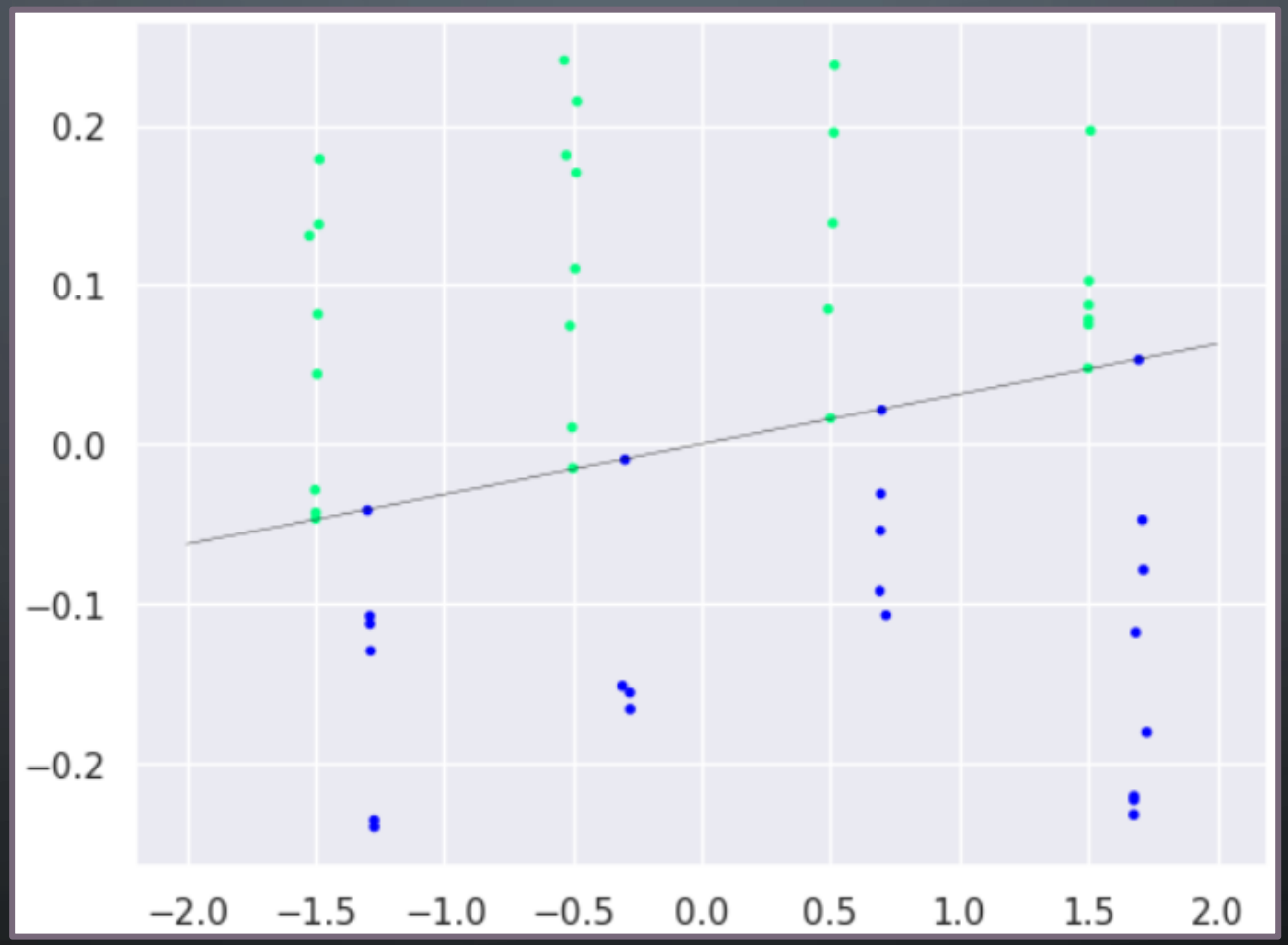


ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-3}$



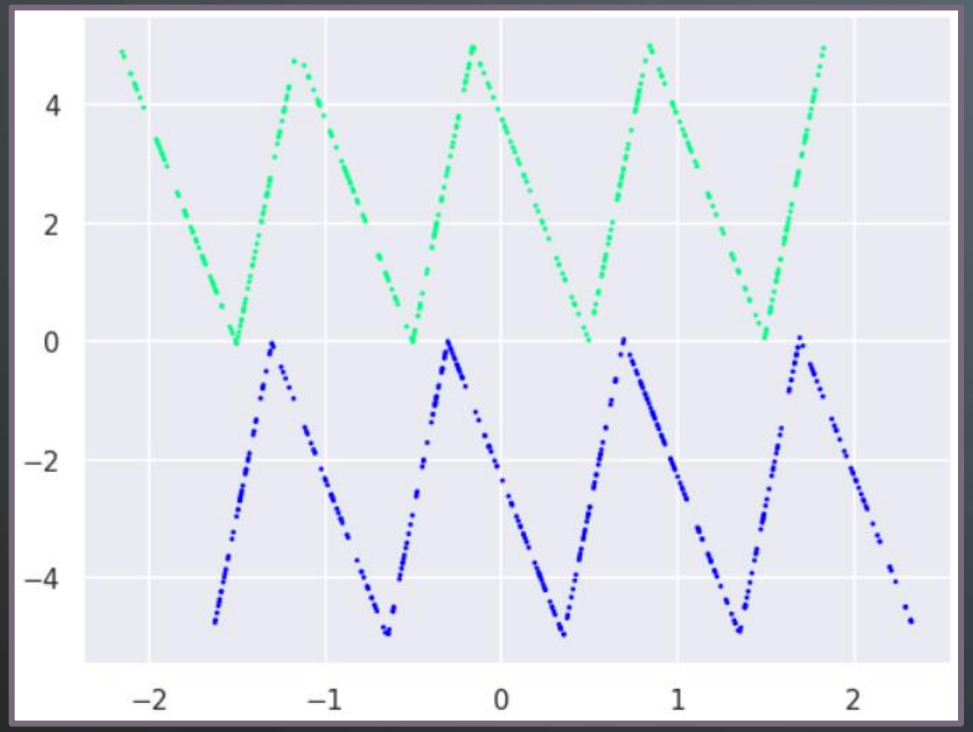
Розв’язок з допомогою методу еліпсоїдів та негладкої штрафної функції

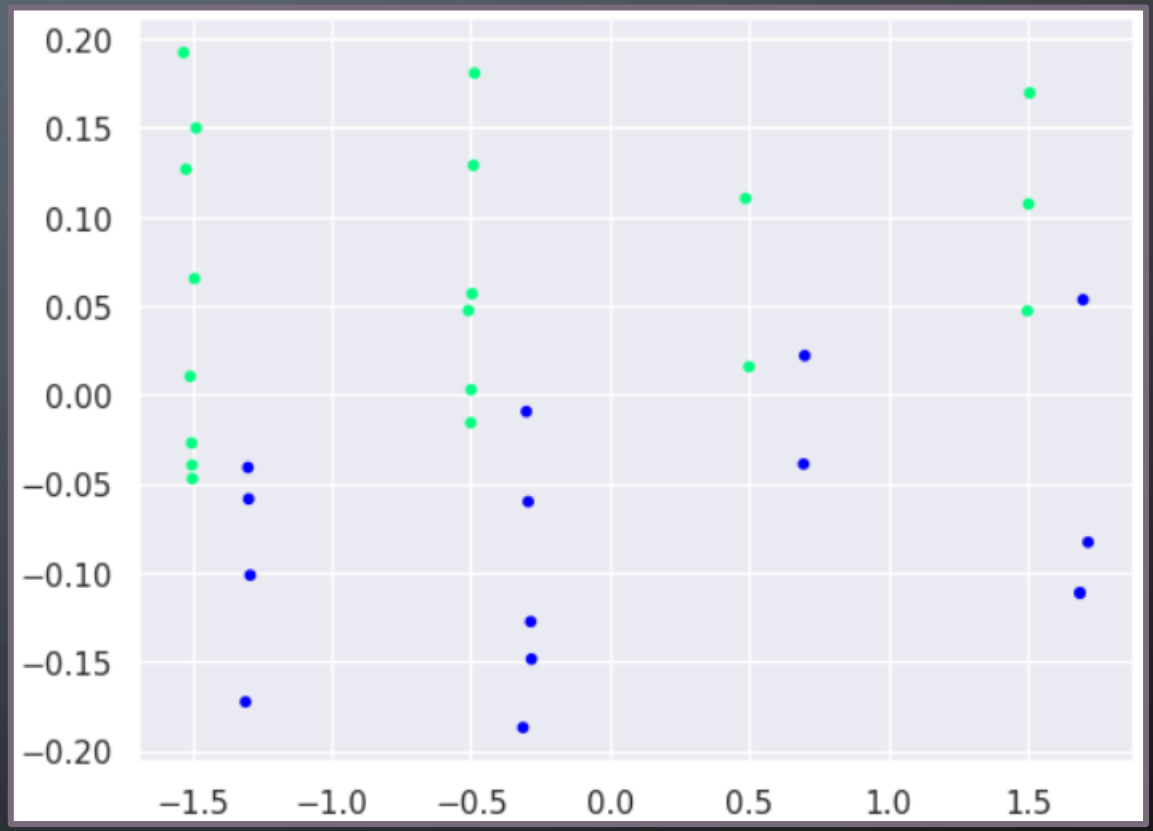
ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-3}$



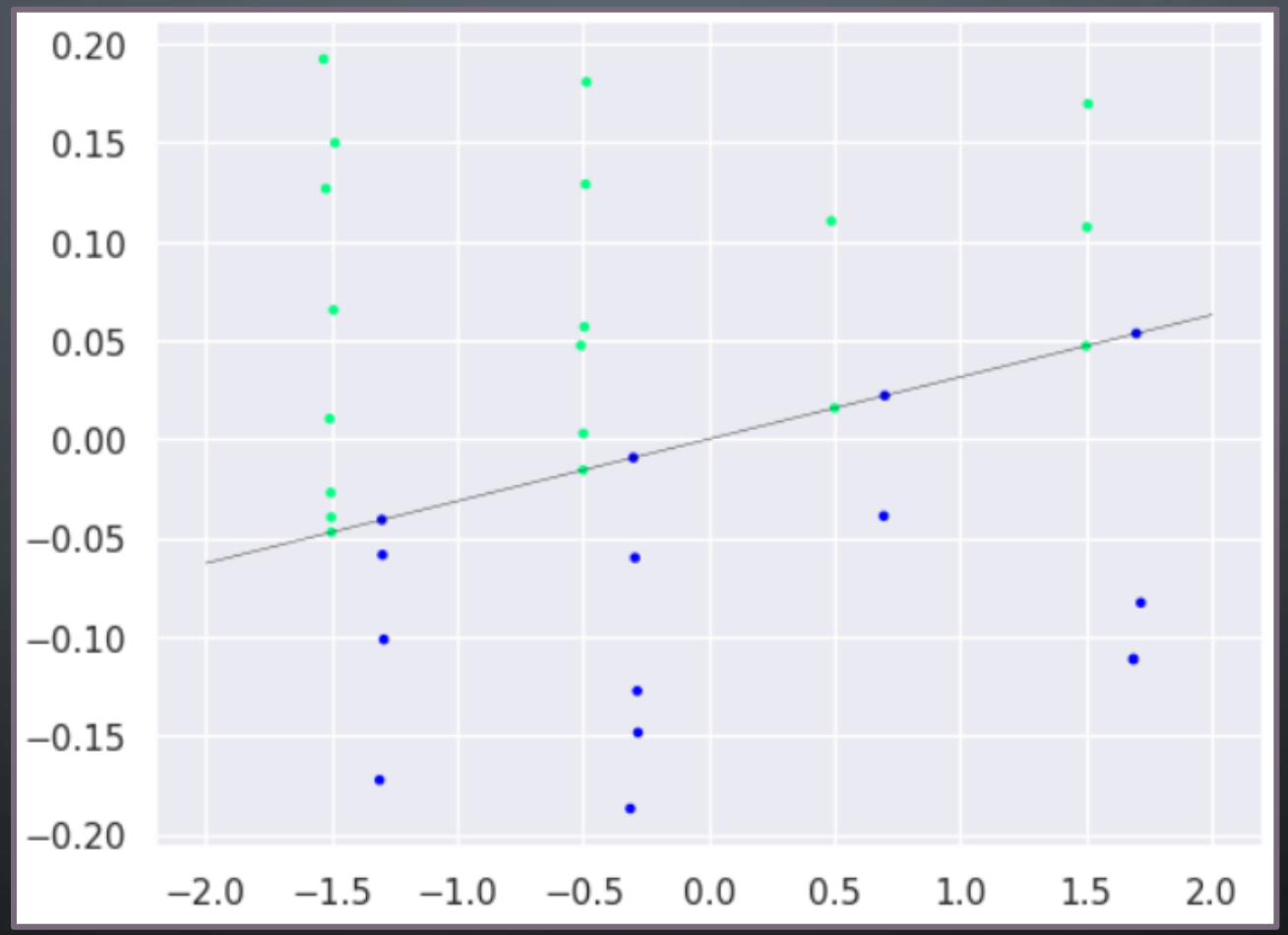
Розв’язок з допомогою sklearn.svm.SVC(C=1e9, kernel=‘linear’)

ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-6}$



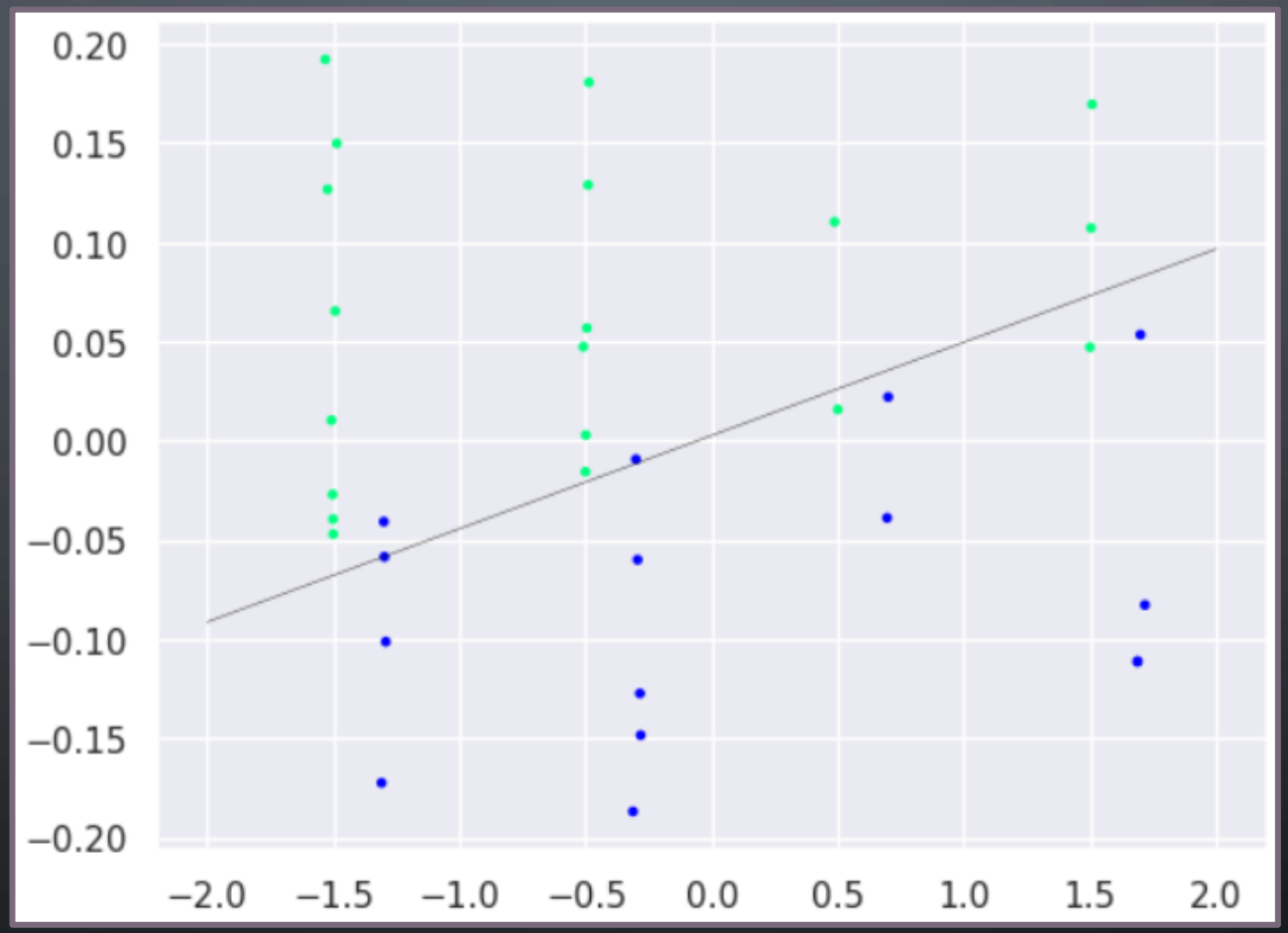


ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-6}$



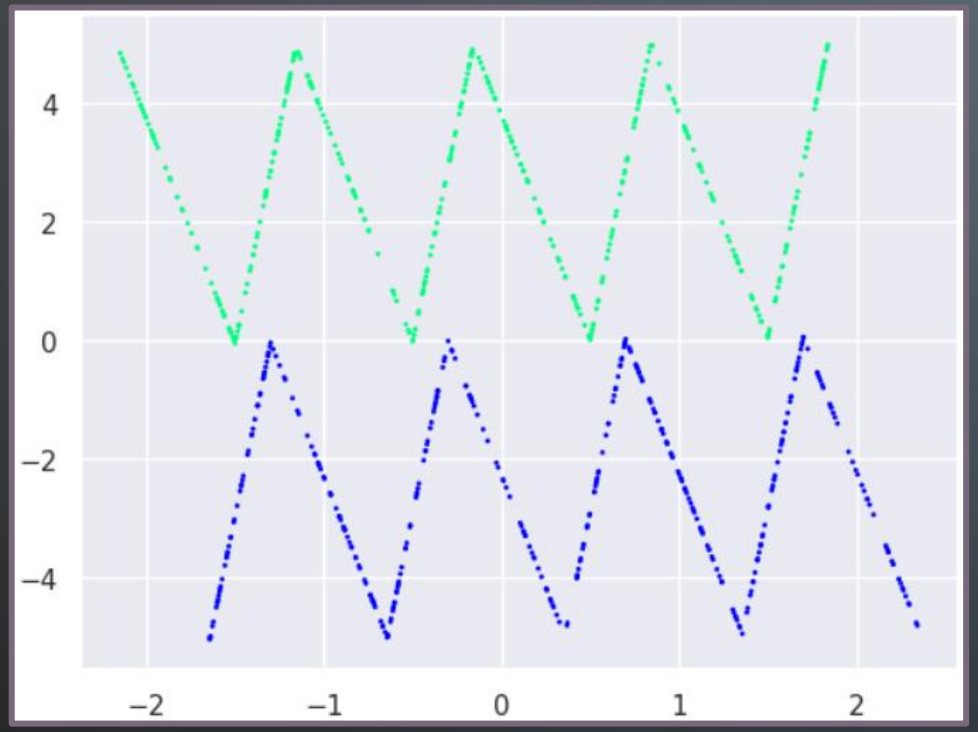
Розв’язок з допомогою методу еліпсоїдів та негладкої штрафної функції

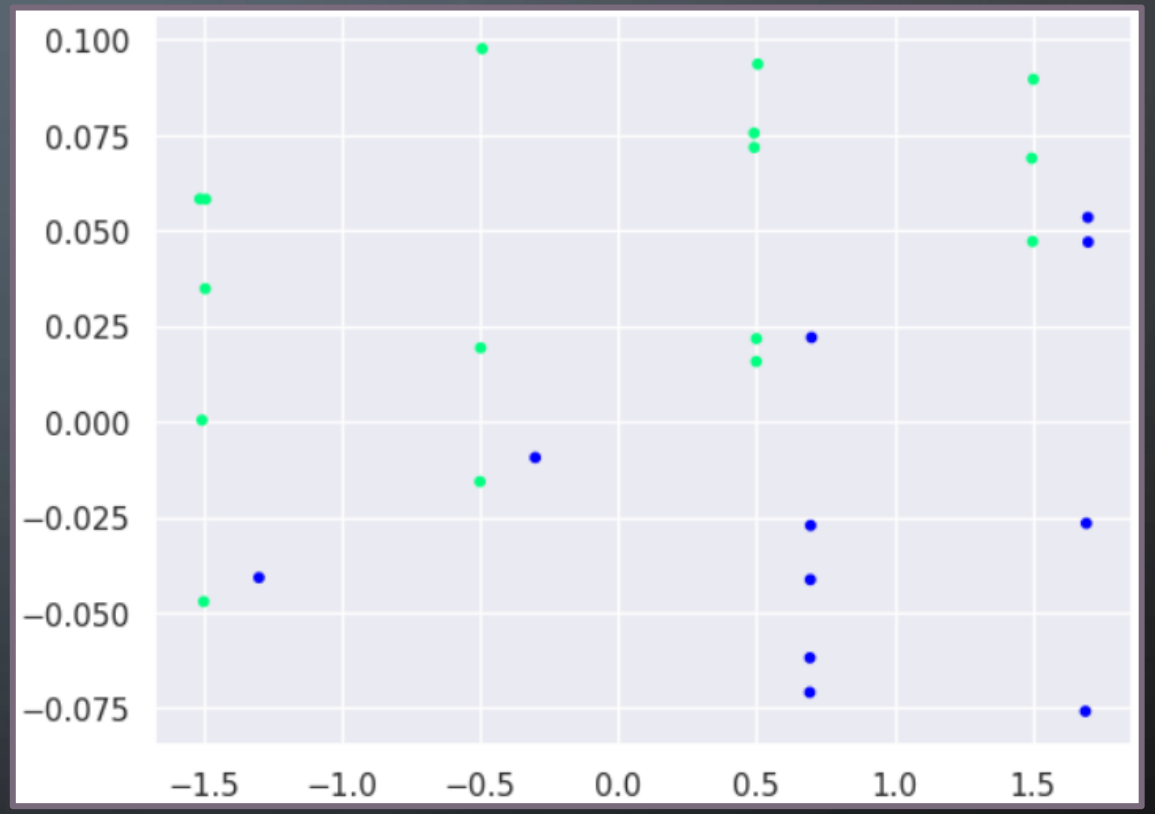
ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-6}$



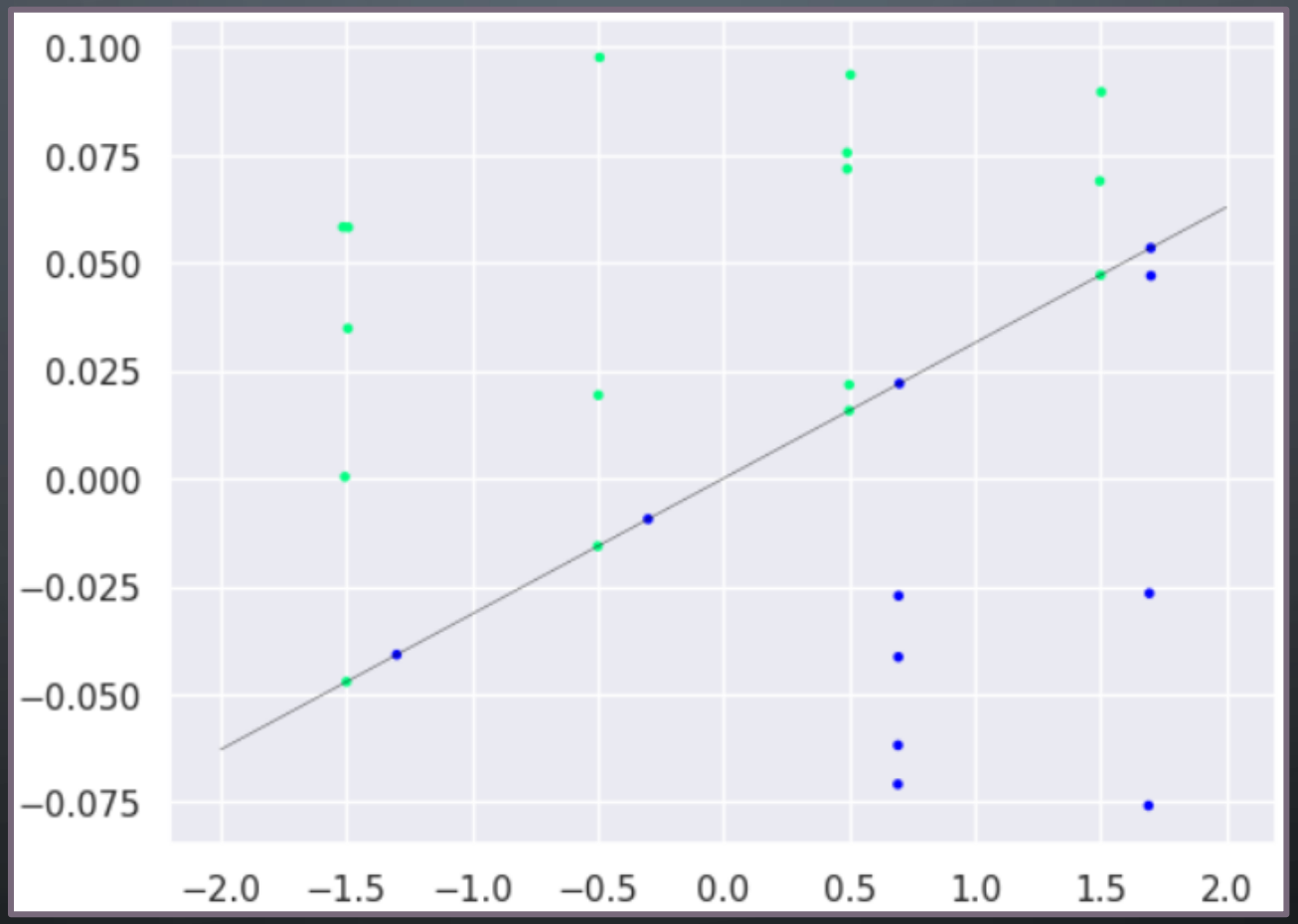
Розв’язок з допомогою sklearn.svm.SVC(C=1e9, kernel=‘linear’)

ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-12}$



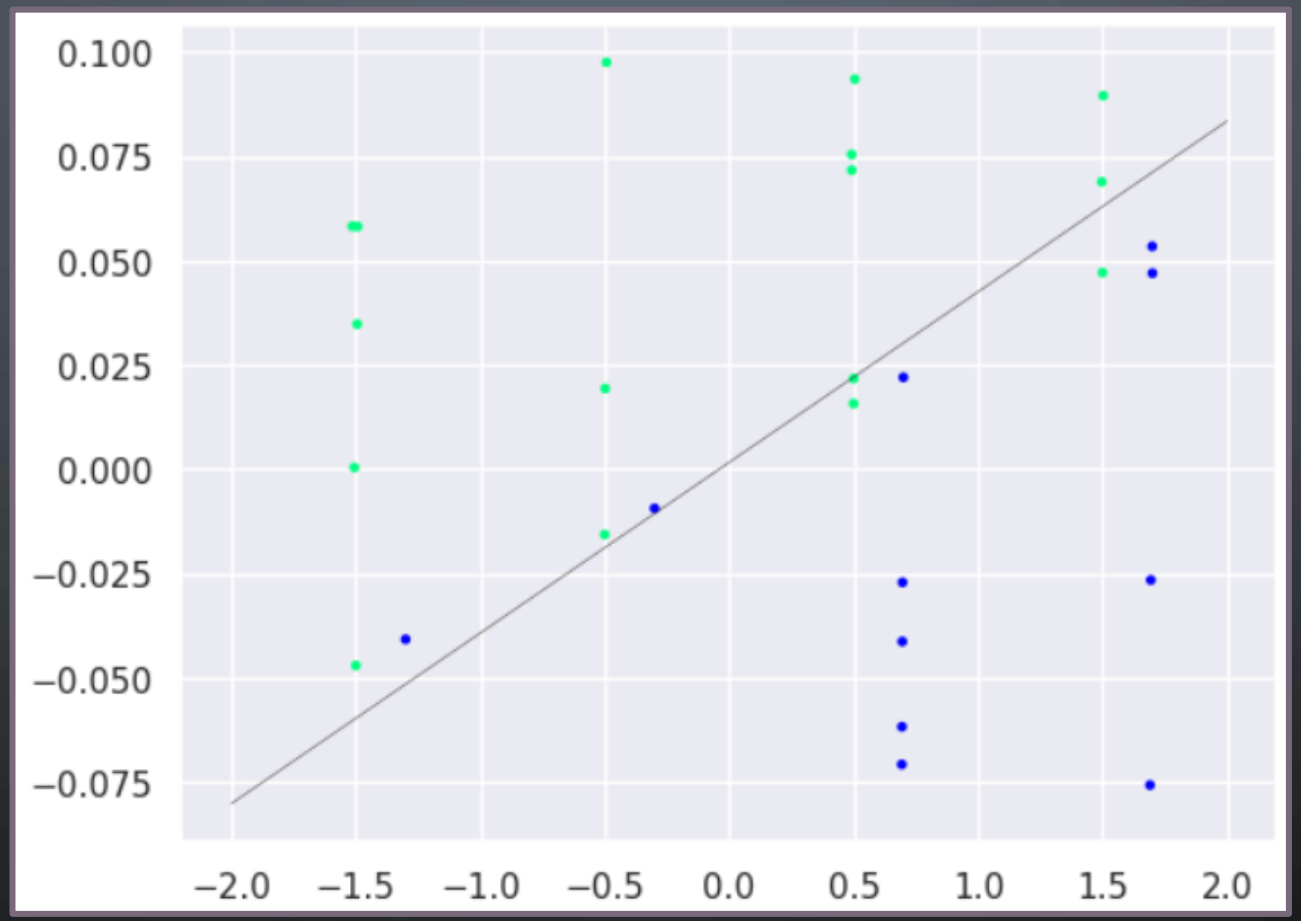


ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-12}$



Розв’язок з допомогою методу еліпсоїдів та негладкої штрафної функції

ДАТАСЕТ “ПИЛА” ІЗ ЗАЗОРОМ $\varepsilon = 10^{-12}$



Розв’язок з допомогою sklearn.svm.SVC(C=1e9, kernel=‘linear’)

ЗАКЛЮЧНЕ СЛОВО

April 15, 2005

Dear Professor Shor,

We have never met, but your work has very much influenced me for many years now. I started with your small 1985 Springer book on subgradient methods, which I read as a PhD student. I recently read your newer book on nondifferentiable optimization (1998), which I enjoyed very much.

I'm enclosing copies of the three books I've written. The first concerns the design of linear controllers via convex optimization; the second is on linear matrix inequalities; and the third one is a basic textbook on convex optimization. [. . .] I hope you can see your strong influence in all of these books.

With the best regards,
Stephen P. Boyd

“Теорія всього класу алгоритмів з розтягом простору ще далека від досконалості. Нам здається достатньо реальною метою побудова такого алгоритму, який би не поступався r-алгоритму за своєю практичною ефективністю і був би настільки добре обґрунтованим, як метод еліпсоїдів.”

Н.З. Шор

ПОСИЛАННЯ

1. N.Z. Shor Minimization Methods for Non-Differentiable Functions, 1985, 164 p. (translated monograph)
2. N.Z. Shor Application of the gradient descent method for solving the network transport problem, Materials of a scientific seminar on theoretical and applied issues of cybernetics and operations research 1, pp. 9-17 (1962) (in russian)
3. N.Z. Shor Cut-off method with space extension in convex programming problems, Cybernetics 13, pp. 94-96 (1977)
4. D.B. Yudin, A.S. Nemirovskii, Informational complexity and efficient methods for the solution of convex extremal problems, Matekon 13, pp. 25-45 (1976)
5. L.G. Khachiyan, Polynomial algorithms in linear programming, USSR Comput. Math. Math. Phys. 20 (1), pp. 53-72 (1980)
6. N.Z. Shor, M.G. Zhurbenko, A minimization method using the operation of extension of the space in the direction of the difference of two successive gradients, Cybernetics 7 (3), pp. 450-459 (1971)

ПОСИЛАННЯ

7. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Springer Series in Statistics): textbook. Springer, 2nd Edition. 2016. 767 p.
8. P.I. Stetsyuk, V.V. Savitsky, On Defects Searching in Regular 3D-Structures, Journal of Automation and Information Sciences 50 (3), pp. 21-37 (2018)
9. Deisenroth M., Faisal A., Soon Ong C. Mathematics for Machine Learning: textbook. Cambridge, 1st Edition. 2020. 398 p.
10. Boyd S., Vandenberghe L., Convex Optimization: textbook, Cambridge University Press, 2004 https://web.stanford.edu/~boyd/cvxbook/
11. Stetsyuk P.I., Berezovskyi O.A., Zhurbenko M.G., Kropotov D.O. Non-smooth optimization methods in special classification problems, preprint, V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, 2009-1, 28 p. (in Ukrainian)
12. N.Z. Shor, Nondifferentiable optimization and polynomial problems: Amsterdam, Kluwer, 1998, 396 p.

ПОДЯКИ

Ця доповідь була підтримана:

- Проєктом "Нові субградієнтні та екстраградієнтні методи для негладких задач регресії", 0124U002162
- Грантом Volkswagen Foundation №97775
- Проєктом дослідних робіт молодих вчених №07-02/03-2023

ДЯКУЮ ЗА УВАГУ!

Якщо Вас зацікавив представлений напрямок і Ви б хотіли долучитися до досліджень чи обрати його тематикою Вашої дипломної роботи, звертайтеся за адресами:

Стецюк Петро Іванович	stetsyukp@gmail.com
Стовба Віктор Олександрович	vik.stovba@gmail.com
Корабльов Микола Миколайович	m.m.korablov@gmail.com