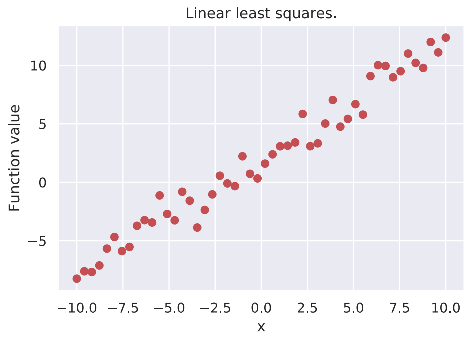
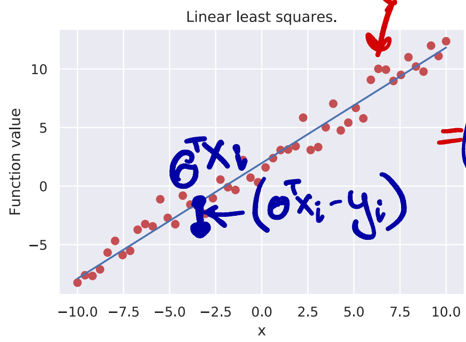
Problem





$y = kx + b$

$y = \theta^T x = \theta_0 x + \theta_1$

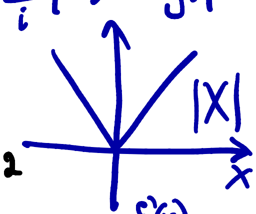
In a least-squares, or linear regression, problem, we have measurements $X \in \mathbb{R}^{m \times n}$ and $y \in \mathbb{R}^m$ and seek a vector $\theta \in \mathbb{R}^n$ such that $X\theta$ is close to y. Closeness is defined as the sum of the squared differences:

наблюдения

сильно выпуклая $f(x) = x^2$

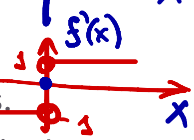
$$
\boxed{\sum_{i=1}^{m} (x_i^\top \theta - y_i)^2 \longrightarrow \min_{\theta \in \mathbb{R}^n}}
$$

$\sum_i |x_i^T\theta - y_i|$



also known as the l_2-norm squared, $\|X\theta - y\|_2^2$

$f(\theta) = \|X\theta - y\|_2^2$

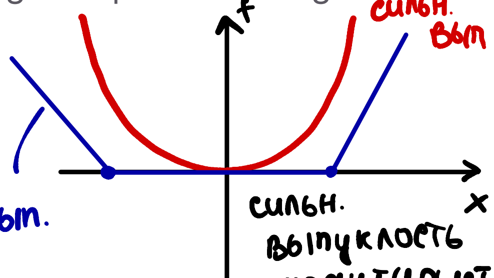


For example, we might have a dataset of m users, each represented by n features. Each row $x_i^\top$ of X is the features for user i, while the corresponding entry y_i of y is the measurement we want to predict from $x_i^\top$, such as ad spending. The prediction is given by $x_i^\top \theta$.

выпуклая $f(x) = x$

We find the optimal θ by solving the optimization problem

$$
\boxed{\|X\theta - y\|_2^2 \to \min_{\theta \in \mathbb{R}^n}}
$$



сильн. выпуклость гарантирует единств. решения

Let θ^* denote the optimal θ. The quantity $r = X\theta^* - y$ is known as the residual. If $\|r\|_2 = 0$, we have a perfect fit.

$\|x\| = \|x\|_2 = \sqrt{\sum_{i=1} x_i^2}$

$\|x\|_1, \|x\|_\infty, \|x\|_p \quad p \geq 1$

Note, that the function needn't be linear in the argument x but only in the parameters θ that are to be determined in the best fit.

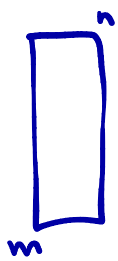
$X\theta = y \quad | \; (X^{-1} \cdot X)\theta = X^{-1} \cdot y$

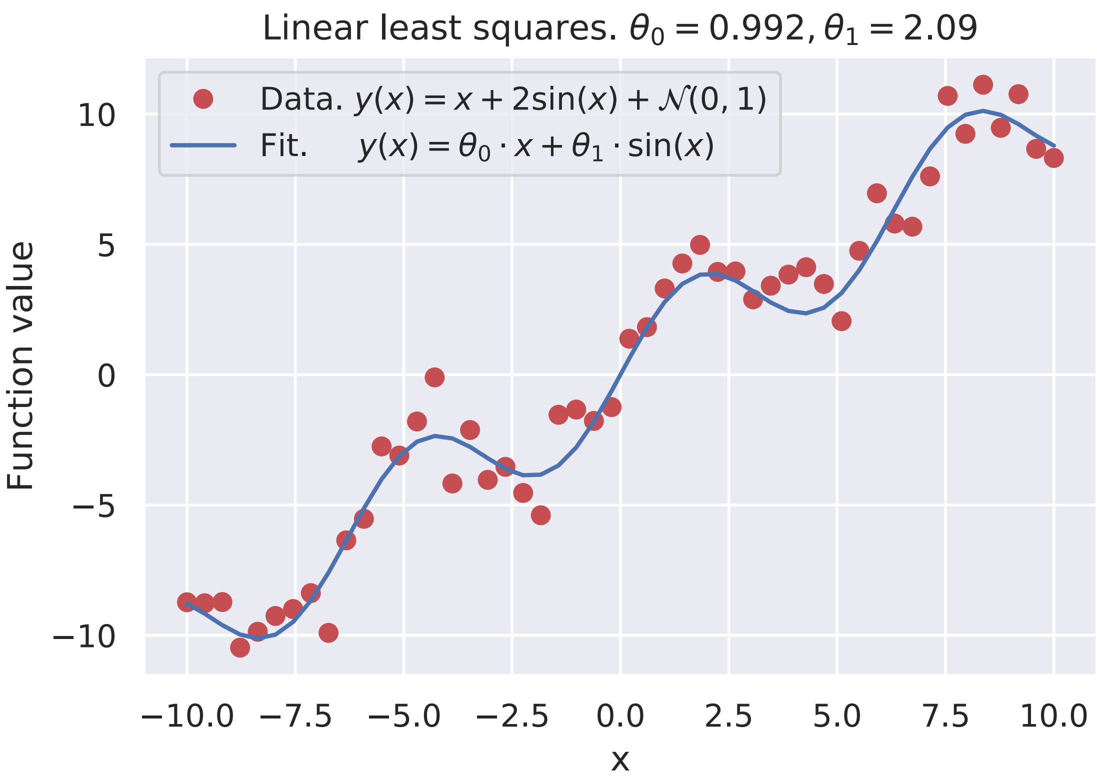
$\underset{n \times m}{X^T} \underset{m \times n}{X} \underset{n \times 1}{\theta} = \underset{n \times m}{X^T} \underset{m \times 1}{y}$

$\boxed{\theta = (X^T X)^{-1} X^T y}$

$\boxed{\cancel{\theta^* = \frac{y}{X}}}$

$\boxed{\cancel{\theta = X^{-1} \cdot y}}$





Approaches

Moore–Penrose inverse

If the matrix X is relatively small, we can write down and calculate exact solution:

$$\theta^* = (X^\top X)^{-1} X^\top y = X^\dagger y,$$

where $X^\dagger$ is called pseudo-inverse matrix. However, this approach squares the condition number of the problem, which could be an obstacle in case of ill-conditioned huge scale problem.

QR decomposition

For any matrix $X \in \mathbb{R}^{m \times n}$ there is exists QR decomposition:

$$X = Q \cdot R,$$

where Q is an orthogonal matrix (its columns are orthogonal unit vectors meaning $Q^\top Q = QQ^\top = I$ and R is an upper triangular matrix. It is important to notice, that since $Q^{-1} = Q^\top$, we have:

$$QR\theta = y \quad \longrightarrow \quad R\theta = Q^\top y$$

Now, process of finding theta consists of two steps:

1. Find the QR decomposition of X.
2. Solve triangular system $R\theta = Q^\top y$, which is triangular and, therefore, easy to solve.

Cholesky decomposition

For any positive definite matrix $A \in \mathbb{R}^{n \times n}$ there is exists Cholesky decomposition:

$$X^\top X = A = L^\top \cdot L,$$

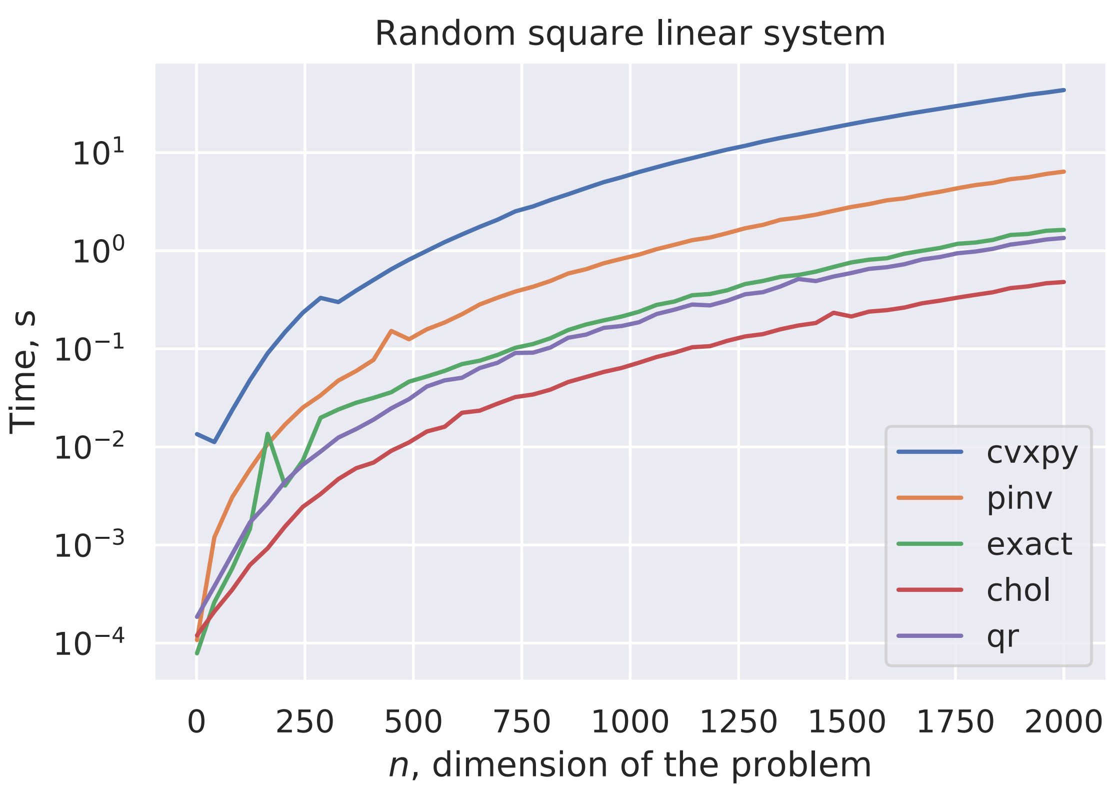
where L is an lower triangular matrix. We have:

$$L^\top L\theta = y \quad \longrightarrow \quad L^\top z_\theta = y$$

Now, process of finding theta consists of two steps:

1. Find the Cholesky decomposition of $X^\top X$.
2. Find the $z_\theta = L\theta$ by solving triangular system $L^\top z_\theta = y$
3. Find the θ by solving triangular system $L\theta = z_\theta$

Note, that in this case the error stil proportional to the squared condition number.



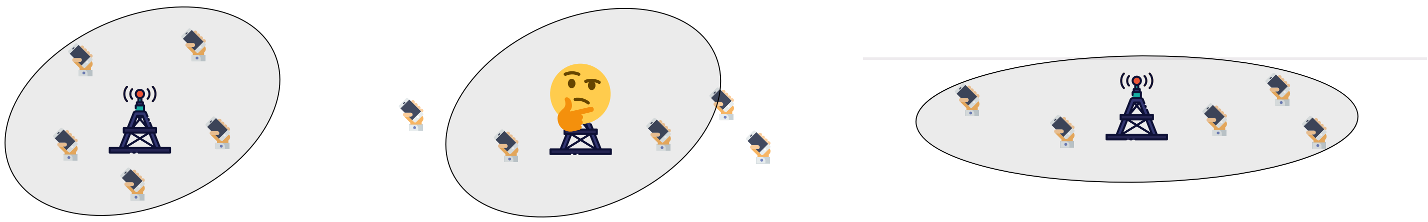
Code

Open in Colab

References

- CVXPY documentation
- Interactive example
- Jupyter notebook by A. Katrutsa

Problem



Let $x_1, \ldots, x_n$ be the points in $\mathbb{R}^2$. Given these points we need to find an ellipsoid, that contains all points with the minimum volume (in 2d case volume of an ellipsoin is just the square).

An invertible linear transformation applied to a unit sphere produces an ellipsoid with the square, that is $\det A^{-1}$ times bigger, than the unit sphere square, that's why we parametrize the interior of ellipsoid in the following way:

$$S = \{x \in \mathbb{R}^2 \mid u = Ax + b, \|u\|_2^2 \leq 1\}$$

Sadly, the determinant is the function, which is relatively hard to minimize explicitly. However, the function $\log \det A^{-1} = -\log \det A$ is actually convex, which provides a great opportunity to work with it. As soon as we need to cover all the points with ellipsoid of minimum volume, we pose an optimization problem on the convex function with convex restrictions:

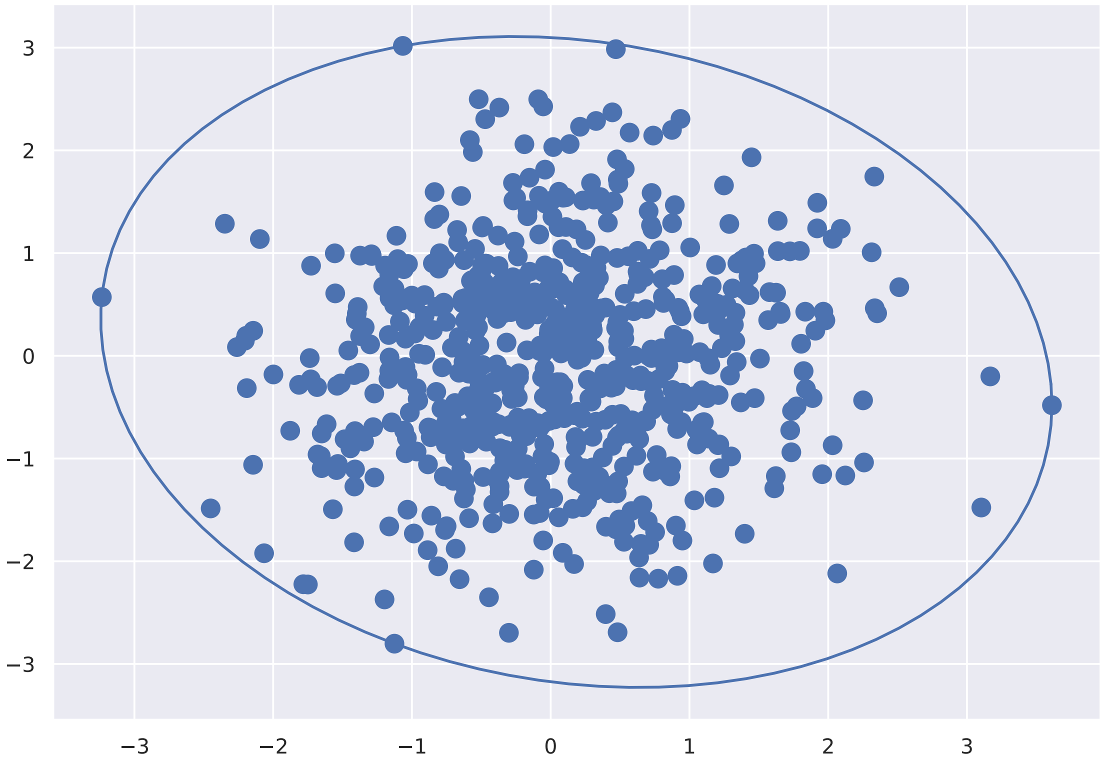
$\det A^{-1} \sim V$

$$\begin{split} & \min_{A \in \mathbb{R}^{2\times 2}, b \in \mathbb{R}^2} -\log \det(A) \\ \text{s.t. } & \|Ax_i + b\| \leq 1, i = 1, \ldots, n \\ & A \succ 0 \end{split}$$

$f(A) = \det A$

$\boxed{f(A) = -\ln \det A}$

выпуклая



Code

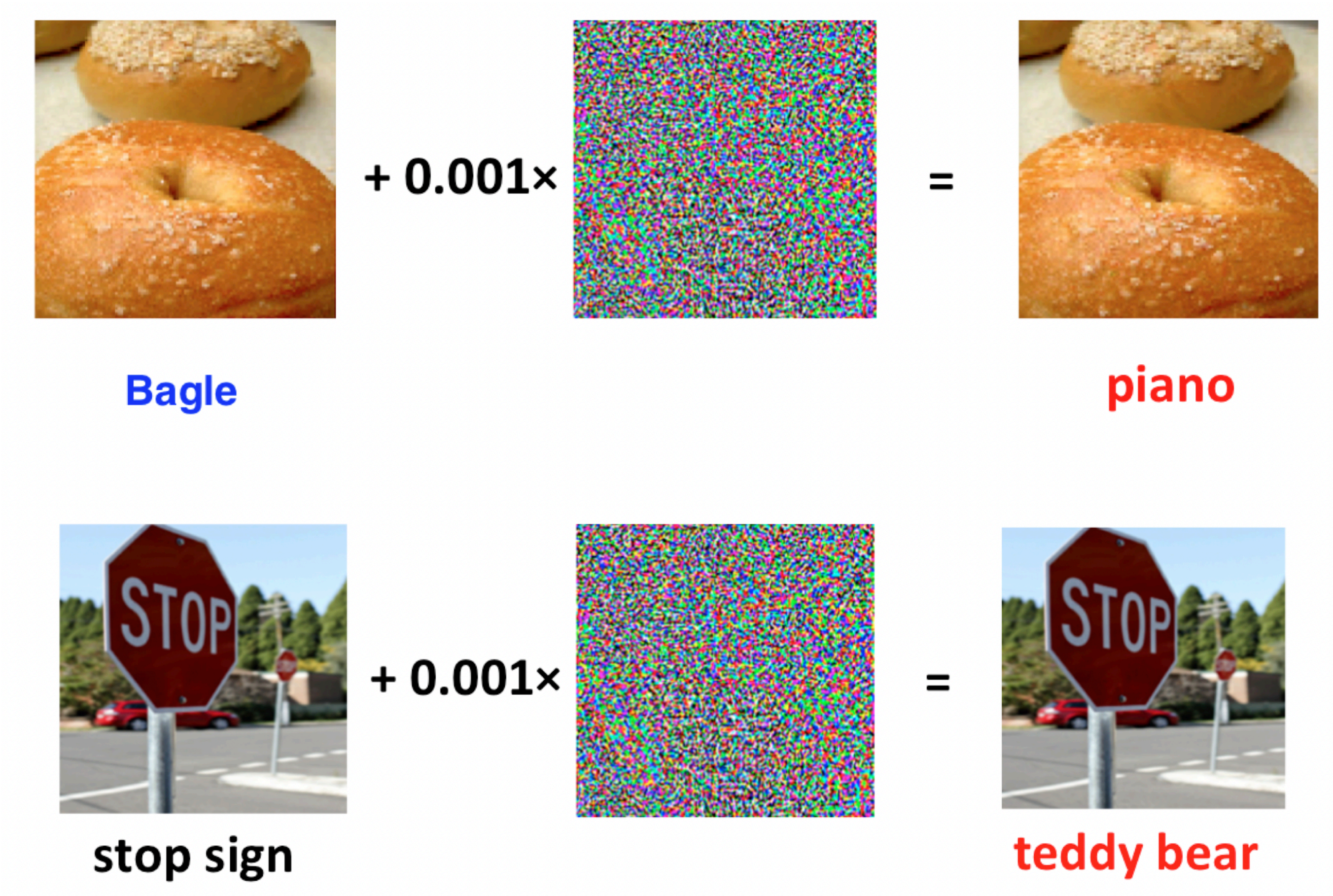
Open in Colab

References

- Jupyter notebook by A. Katrutsa
- https://cvxopt.org/examples/book/ellipsoids.html

Lipschitz constant of a convolutional layer in neural network

It was observed, that small perturbation in Neural Network input could lead to significant errors, i.e. misclassifications.



Lipschitz constant bounds the magnitude of the output of a function, so it cannot change drastically with a slight change in the input

$$\|NN(image) - NN(image + \varepsilon)\| \leq L\|\varepsilon\|$$

если L – мало, то выход меняется не сильно при малом изм. входа

In this notebook we will try to estimate Lipschitz constant of some convolutional layer of a Neural Network.

Лин. слой нейронной сети

x_{in}

$$f(x) = y = W \cdot x_{in}$$

$$\|f(x+\varepsilon) - f(x)\| \leq L\underbrace{\|x+\varepsilon - x\|}$$

$$\|W(x+\varepsilon) - Wx\| \leq L\|\varepsilon\|$$

$$\|W\cdot\varepsilon\| \leq L\cdot\|\varepsilon\|$$

Code

Open in Colab

$\|W\| = \sigma_{MAX}(W)$

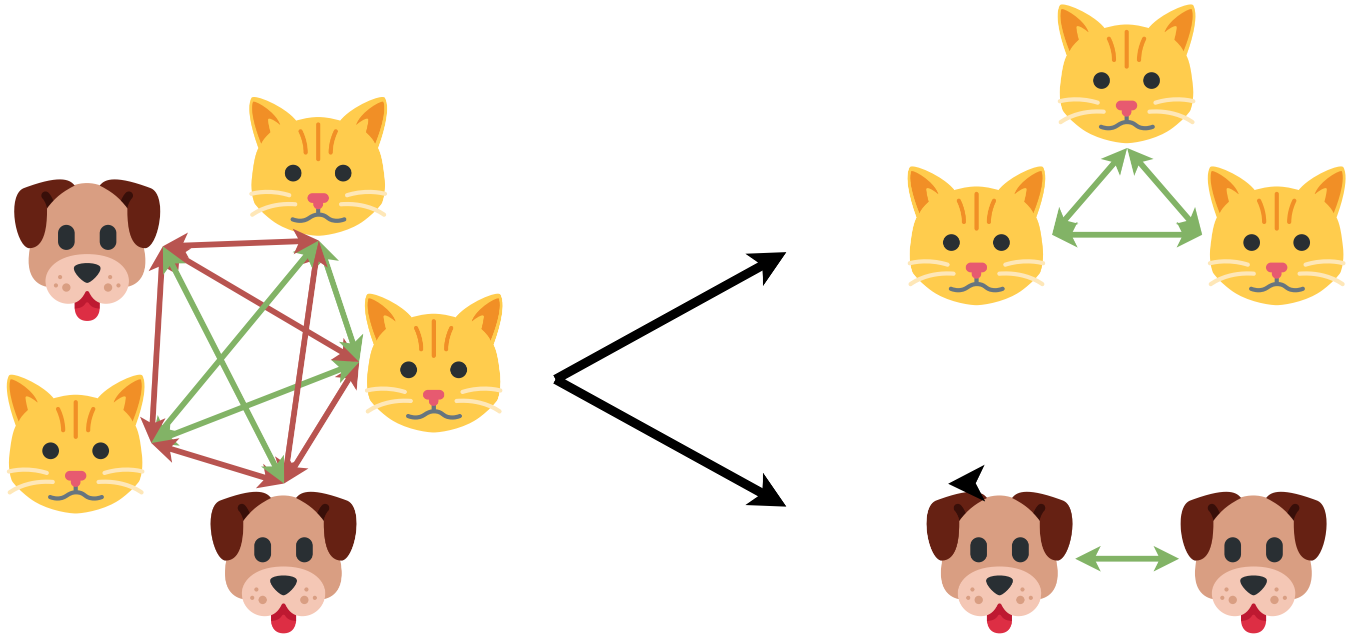
чтобы найти

$W(W(W \cdot x))$ power method

$L \sim \|W\|$ L

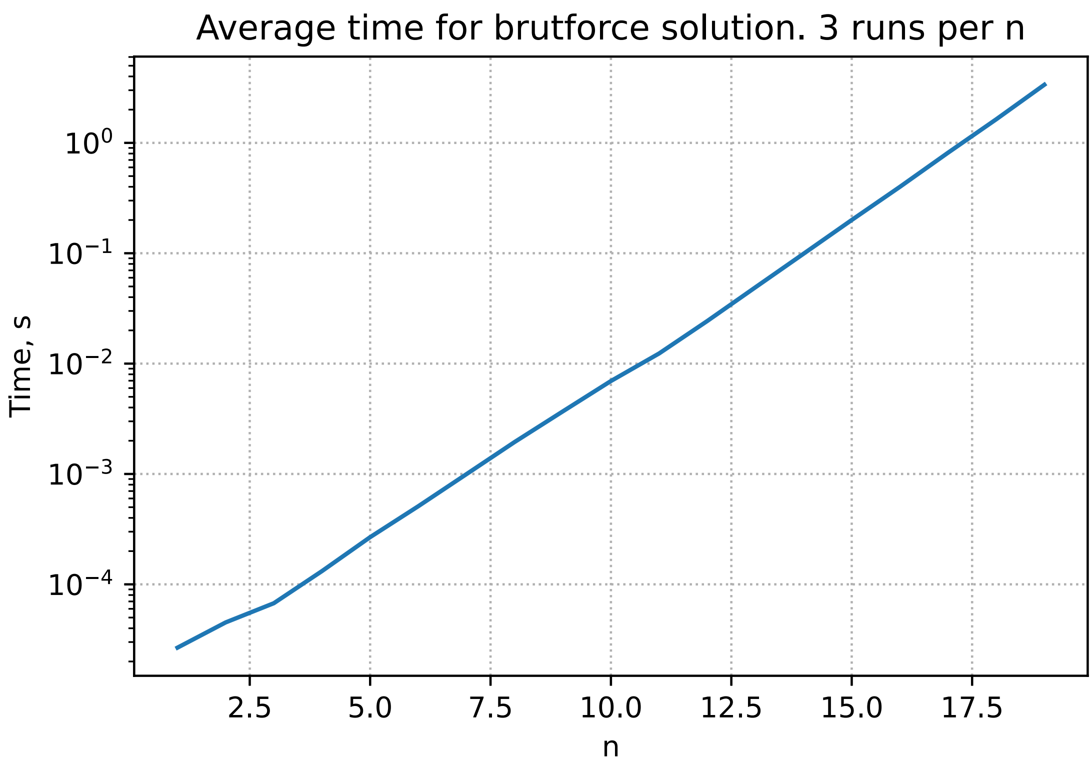
$\|W \cdot \varepsilon\| \leq \|W\| \|\varepsilon\|$

Intuition



Suppose, we have a set of n objects, which are needed to be splitted into two groups. Moreover, we have information about the preferences of all possible pairs of objects to be in the same group. this information could be presented in the matrix form: $W \in \mathbb{R}^{n \times n}$, where $\{w_{ij}\}$ is the cost of having i-th and j-th object in the same partitions. It is easy to see, that the total number of partitions is finite and eqauls to 2^n. So this problem can in principle be solved by simply checking the objective value of each feasible point. Since the number of feasible points grows exponentially, however, this is possible only for small problems (say, with $n \leq 30$). In general (and for n larger than, say, 50) the problem is very difficult to solve.

For example, bruteforce solution on MacBook Air with M1 processor without any explicit parallelization will take more, than a universe lifetime for $n = 62$.



Despite the hardness of the problems, there are several ways to approach it.

Problem

We consider the (nonconvex) problem

$$x \quad \begin{array}{c} 1\\ 2\\ 3\\ 4\\ 5 \end{array} \begin{pmatrix} +1 \\ -1 \\ -1 \\ -1 \\ +1 \end{pmatrix}$$

$$\min_{x \in \mathbb{R}^n} x^\top W x,$$
$$\text{s.t. } x_i^2 = 1, \; i = 1, \ldots, n$$

where $W \in \mathbb{R}^n$ is the symetric matrix. The constraints restrict the values of x_i to 1 or -1, so the problem is equivalent to finding the vector with components ± 1 that minimizes $x^\top W x$. The feasible set here is finite (it contains 2^n points), thus, is non-convex.

The objective is the total cost, over all pairs of elements, and the problem is to find the partition with least total cost.

Simple lower bound with duality

We now derive the dual function for this problem. The Lagrangian is

$$L(x,\nu) = x^\top W x + \sum_{i=1}^n \nu_i (x_i^2 - 1) = x^\top (W + \text{diag}(\nu)) x - \mathbf{1}^\top \nu.$$

We obtain the Lagrange dual function by minimizing over x:

$$\begin{aligned} g(\nu) &= \inf_{x \in \mathbb{R}^n} x^\top (W + diag(\nu)) x - \mathbf{1}^\top \nu = \\ &= \begin{cases} \mathbf{1}^\top \nu, & W + \text{diag}(\nu) \succeq 0 \\ -\infty, & \text{otherwise} \end{cases} \end{aligned}$$

sa

This dual function provides lower bounds on the optimal value of the difficult problem. For example, we can take any specific value of the dual variable

$$\nu = -\lambda_{min}(W)\mathbf{1},$$

This yields the bound on the optimal value p^*:

$$p^* \geq g(\nu) \geq -\mathbf{1}^\top \nu = n\lambda_{min}(W)$$

Question Can you obtain the same lower bound without knowledge of duality, but using the iddea of eigenvalues?

Code

Open in Colab

References

- Convex Optimization book by Stephen Boyd and Lieven Vandenberghe.

$$L(k) = \sum_{i=1}^{n} (k x_i - y_i)^2$$

$$\frac{\partial L}{\partial k} = \sum_{i=1}^{n} \frac{\partial}{\partial k}(k x_i - y_i)^2 = \sum_{i=1}^{n} 2(k x_i - y_i) \cdot \frac{\partial}{\partial k}(k x_i - y_i) =$$

$$= \boxed{\sum_{i=1}^{n} 2(k x_i - y_i) \cdot x_i} = 0 \quad \rightarrow k = \dots$$

$$\sum_{i=1}^{n} (k x_i - y_i) x_i = 0$$

$$\sum_{i=1}^{n} (k x_i^2 - x_i y_i) = 0$$

$$\sum k x_i^2 - \sum x_i y_i = 0$$

$$k \cdot \sum_i x_i^2 = \sum x_i y_i \rightarrow \boxed{k = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sum_{i=1}^{n} x_i^2}}$$

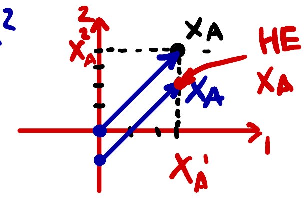
Useful definitions and notations

We will treat all vectors as column vectors by default. The space of real vectors of length n is denoted by $\mathbb{R}^n$, while the space of real-valued $m \times n$ matrices is denoted by $\mathbb{R}^{m \times n}$.

$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n \in \mathbb{C}^n$

Basic linear algebra background

$x_A \in \mathbb{R}^2$



$a = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ $b = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$

The standard **inner product** between vectors x and y from $\mathbb{R}^n$ is given by

СКАЛЯРНОЕ произведение

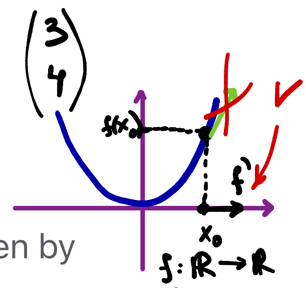
$\begin{pmatrix} x_A^1 \\ x_A^2 \end{pmatrix}$ $\begin{pmatrix} 3 \\ 4 \end{pmatrix}$

$$\boxed{\langle x, y \rangle = x^\top y = \sum_{i=1}^n x_i y_i = y^\top x = \langle y, x \rangle}$$

$a^T b = \langle a, b \rangle =$
$= 1 \cdot (-1) + 2 \cdot 0 + 3 \cdot 1 = 2$

$\underset{n\times 1}{x} \cdot \underset{n\times 1}{y}$ $\underset{n\times 1}{x}\,\underset{1\times n}{y^T}$ ← $n \times n$ outer product

Here x_i and y_i are the scalar i-th components of corresponding vectors.



$f: \mathbb{R} \to \mathbb{R}$
f' - число

The standard **inner product** between matrices X and Y from $\mathbb{R}^{m \times n}$ is given by

$$\boxed{\langle X, Y \rangle = \text{tr}(X^\top Y) = \sum_{i=1}^m \sum_{j=1}^n X_{ij} Y_{ij} = \text{tr}(Y^\top X) = \langle Y, X \rangle}$$

$\langle X, X \rangle = \|X\|_F^2$

The determinant and trace can be expressed in terms of the eigenvalues

A - квадратная

$A - \lambda_1, \dots, \lambda_n$

$\boxed{Av = \lambda v}$

$$\boxed{\det A = \prod_{i=1}^n \lambda_i,} \qquad \boxed{\text{tr} A = \sum_{i=1}^n \lambda_i}$$

$\text{tr} \begin{pmatrix} 1 & 4 \\ 3 & 2 \end{pmatrix} = \boxed{\lambda_1 + \lambda_2 = 3}$

$\det \begin{pmatrix} 1 & 4 \\ 2 & 3 \end{pmatrix} = 1 \cdot 3 - 2 \cdot 4 = -5$

$\boxed{\lambda_1 \cdot \lambda_2 = -5}$

Don't forget about the cyclic property of a trace for a square matrices A, B, C, D:

$$\text{tr}(ABCD) = \text{tr}(DABC) = \text{tr}(CDAB) = \text{tr}(BCDA)$$

Про λ в следующей серии.

The largest and smallest eigenvalues satisfy

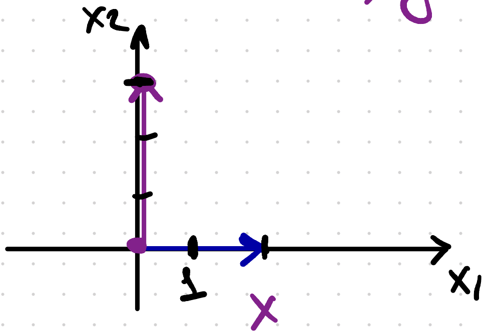
$$\lambda_{\min}(A) = \inf_{x \neq 0} \frac{x^\top Ax}{x^\top x}, \qquad \lambda_{\max}(A) = \sup_{x \neq 0} \frac{x^\top Ax}{x^\top x}$$

and consequently $\forall x \in \mathbb{R}^n$ (Rayleigh quotient):

$$\lambda_{\min}(A) x^\top x \leq x^\top Ax \leq \lambda_{\max}(A) x^\top x$$

A matrix $A \in \mathbb{S}^n$ (set of square symmetric matrices of dimension n) is called **positive (semi)definite** if for all $x \neq 0 (\text{for all } x) : x^\top Ax > (\geq) 0$. We denote this as

Если $\langle x, y\rangle = 0 \iff x \perp y$ ортогональность



$x = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$ $\quad \langle x, y\rangle =$

$y = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$ $\quad = 2\cdot 0 + 0\cdot 3 =$

$= 0$

Пусть $x_1 \in \mathbb{R}^n$, $x_2 \in \mathbb{R}^n$

Насколько x_1 далеко от x_2?

$\rho(x_1, x_2) = \|x_1 - x_2\| = 0 \iff x_1 = x_2$

НОРМА
↳ $\|x\|$

$\|x\| \geqslant 0$

$\|x\| = 0 \iff x = 0_n$ $\quad \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}_n$

$\|\alpha x\| = |\alpha| \cdot \|x\|$

$\|x+y\| \leq \|x\| + \|y\|$

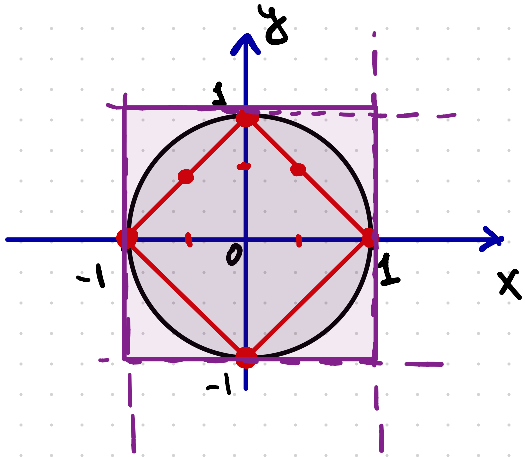
Евклидова норма

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$\left\|\begin{pmatrix} 3 \\ 4 \end{pmatrix}\right\|_2 = \sqrt{3^2+4^2} = \sqrt{5^2} = 5$

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$$

$$\|X\|_1 = \sum_{i=1}^{n} |X_i| .$$ ↲ ℓ_1 регуляр.



$\|V\|_2 \leq 1$

$\sqrt{x^2+y^2} \leq 1$

$x^2+y^2 \leq 1$

$\|V\|_1 \leq 1$

$|x|+|y| \leq 1$

$$\boxed{\langle x, x \rangle = \|X\|_2^2}$$

$\langle x, x \rangle = \|X\|^2$

МАТРИЦЫ

$$A \in \mathbb{R}^{m \times n}$$

строки

столбцы

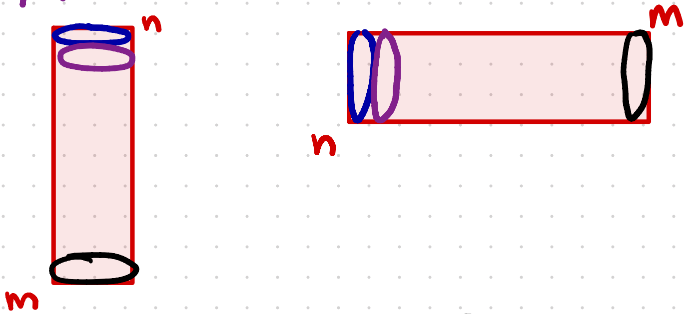
$$\begin{array}{ccc} & 1\text{ стол} & & n\text{ стл} \\ 1\text{ стр} & \left(\begin{matrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{matrix}\right. & & \left.\right) \\ m\text{ стр} & & & \end{array}$$

покомпонентно

$$A + B = C$$

транспонирование:

$A \in \mathbb{R}^{m \times n}$ $A^T \in \mathbb{R}^{n \times m}$



$$A \cdot B = C$$
$m \times n$ $n \times k$ $m \times k$

для c_{ij} ← элемент C на i-ой строке в j-ом столбце

взять i-ую строку из A
умн. скал. на j-ый столбец B

c_{ij}

$A \succ (\succeq) 0.$

The **condition number** of a nonsingular matrix is defined as

$$\kappa(A) = \|A\|\|A^{-1}\|$$

Matrix and vector multiplication

Let A be a matrix of size $m \times n$, and B be a matrix of size $n \times p$, and let the product AB be:

Если матрица имеет структуру, то операции можно делать быстрее.

$$C = AB$$

$O(n^3)$ ← наивн. алгоритм

∃ более быстрые алгоритмы $O(n^{\log_2 7})$ ← Штрассен

then C is a $m \times p$ matrix, with element (i, j) given by:

$$A_1 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix} \cdot \begin{pmatrix} -1 & & \\ & 2 & \\ & & 3 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 9 \end{pmatrix} \quad O(n)$$

$$\boxed{c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}}$$

n умн. и слож.

быстрее $O(n^2)$ невозможно

Let A be a matrix of shape $m \times n$, and x be $n \times 1$ vector, then the i-th component of the product:

$$\underset{m\times 1}{z} = \underset{m\times n}{A}\,\underset{n\times 1}{x}$$

is given by:

$$z_i = \sum_{k=1}^n a_{ik} x_k$$

Finally, just to remind:

n×k m×n n×k k×n k×n n×m

- $\boxed{C = AB \quad C^\top = B^\top A^\top}$
- $\boxed{AB \neq BA}$
- $\boxed{e^A = \sum\limits_{k=0}^{\infty} \frac{1}{k!} A^k}$ матр. эксп.
- $e^{A+B} \neq e^A e^B$ (but if A and B are commuting matrices, which means that $AB = BA$, $e^{A+B} = e^A e^B$)
- $\boxed{\langle x, Ay \rangle = \langle A^\top x, y \rangle}$ $\langle x, yA \rangle = \langle xA^T, y \rangle$

$(x^T A y) = (A^T x)^T y = \langle A^T x, y \rangle$

Gradient

Let $f(x) : \mathbb{R}^n \to \mathbb{R}$, then vector, which contains all first order partial derivatives:

$A, B \qquad \|A - B\|$

$\|A\|_F$

норма Фробениуса

для простоты

$$\left\| \begin{pmatrix} | & | & | \\ | & | & | \end{pmatrix} \right\| = \left\| \begin{pmatrix} | \\ | \\ | \end{pmatrix} \right\|_2 = \sqrt{\sum_{i,j} a_{ij}^2}$$

спектральная норма $\|A\|_2 = \sigma_{MAX}(A)$.

скалярное произведение матриц

$$\boxed{\langle \underset{m\times n}{A}, \underset{m\times n}{B} \rangle = tr(A^T B)}$$

$$tr(A) = \sum_{i=1}^{n} a_{ii}$$

$$tr\begin{pmatrix} 1 & & \\ & 2 & \\ & & 3 \end{pmatrix} = 1+2+3 = 6$$

Градиент

$$\nabla f(x) = \frac{df}{dx} = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

$f: \mathbb{R}^n \to \mathbb{R}$

$f: \mathbb{R}^2 \to \mathbb{R}$ $\quad v = \begin{pmatrix} x \\ y \end{pmatrix}$

$f = x^2 + y^2 = \|v\|^2$

$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} = \begin{pmatrix} 2x \\ 2y \end{pmatrix}$

$= 2v$

named gradient of $f(x)$. This vector indicates the direction of steepest ascent. Thus, vector $-\nabla f(x)$ means the direction of the steepest descent of the function in the point. Moreover, the gradient vector is always orthogonal to the contour line in the point.

$f'(x^*) = 0 \qquad x^*$ – не точка минимума

$(x^*)^3 = 0 \qquad x^* = 0$

$6x^* = 0$

Hessian

Let $f(x) : \mathbb{R}^n \to \mathbb{R}$, then matrix, containing all the second order partial derivatives:

$$\boxed{f''(x) = \frac{\partial^2 f}{\partial x_i \partial x_j} = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{pmatrix}}$$

In fact, Hessian could be a tensor in such a way: $(f(x) : \mathbb{R}^n \to \mathbb{R}^m)$ is just 3d tensor, every slice is just hessian of corresponding scalar function $(H(f_1(x)), H(f_2(x)), \dots, H(f_m(x)))$.

40 Gb = VRAM

10р $\quad$ 100р²

1000 коп $\quad 10^6$ коп² $= 10^4$ руб²

оценим n

$n^2 \cdot 32$ бит $= 40$ Гбайт $= 320$ Гбит

$n^2 \cdot \cancel{32}$ бит $= \cancel{320}$ Гбит

$n^2 = 10$ Гбит $= 10 \cdot 10^3$ Мбит $= 10 \cdot 10^6$ Кбит $= 10 \cdot 10^9$ бит

$n^2 = 10^{10}$

$\boxed{n \approx 10^5}$

$n = 100к$

Jacobian

The extension of the gradient of multidimensional $f(x) : \mathbb{R}^n \to \mathbb{R}^m$ is the following matrix:

$f: \mathbb{R}^n \to \mathbb{R}^m$

$x \odot x$

ЯКОБИАН

$$f'(x) = \frac{df}{dx^T} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \dots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}$$

LLAMA – 7b

– 13b

– 22b

10^{10} f...

Summary

$$f(x) : X \to Y; \qquad \frac{\partial f(x)}{\partial x} \in G$$

X	Y	G	Name
$\mathbb{R}$	$\mathbb{R}$	$\mathbb{R}$	$f'(x)$ (derivative)
$\mathbb{R}^n$	$\mathbb{R}$	$\mathbb{R}^n$	$\dfrac{\partial f}{\partial x_i}$ (gradient)
$\mathbb{R}^n$	$\mathbb{R}^m$	$\mathbb{R}^{m \times n}$	$\dfrac{\partial f_i}{\partial x_j}$ (jacobian)
$\mathbb{R}^{m \times n}$	$\mathbb{R}$	$\mathbb{R}^{m \times n}$	$\dfrac{\partial f}{\partial x_{ij}}$

В следующей серии:
* матрично-векторное диффер.
* SVD, PCA, спектр матрицы $\lambda(A)$, $\sigma(A)$

General concept

Naive approach

The basic idea of naive approach is to reduce matrix/vector derivatives to the well-known scalar derivatives.

Matrix notation of a function

$$f(x) = c^\top x$$

Scalar notation of a function

$$f(x) = \sum_{i=1}^n c_i x_i$$

Matrix notation of a gradient

$$\nabla f(x) = c$$

$$\frac{\partial f(x)}{\partial x_k} = c_k$$

Simple derivative

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial \left(\sum_{i=1}^n c_i x_i \right)}{\partial x_k}$$

One of the most important practical tricks here is to separate indices of sum (i) and

- $d(X^{-1}) = -X^{-1}(dX)X^{-1}$

References

- Convex Optimization book by S. Boyd and L. Vandenberghe - Appendix A. Mathematical background.
- Numerical Optimization by J. Nocedal and S. J. Wright. - Background Material.
- Matrix decompositions Cheat Sheet.
- Good introduction
- The Matrix Cookbook
- MSU seminars (Rus.)
- Online tool for analytic expression of a derivative.
- Determinant derivative