Lecture Notes on Convex 0ptimization

with Applications to Image Processing

Prof. Mohammed Hachama

hachamam@gmail.com
https://hachama.github.io/home/

University of Khemis Miliana
Laboratory of Pure and Applied Mathematics, University of Msila

September, 2020

Contents

Chapter I
Introduction

Skills to acquire

- **How to represent images as matrices and vectors.**
- **How to model some fundamental image processing problems as linear systems and optimization problems: denoising, deconvolution, inpainting, approximation, compression.**
- **Understand the PSF function.**
- **Manipulate fundamental matrix properties and operations: norms, factorization, SVD, Differentiation.**

Many practical problems in applied mathematics can be formulated as optimization problems. We can cite image reconstruction and restoration(denoising, inpainting, deconvolution), supervised/unsupervised learning, parameter estimation, statistical inference, ... In this book, we consider only **finite dimensional** optimization problems, in which images are represented as matrices or vectors.

1 Image representation

A natural representation is to consider images having m rows and n columns as matrices of the space $X \in \mathbb{R}^{m\times n}$. But, it is usually convenient to transform images into vectors by stacking the columns of the images:

$$\boldsymbol{x} = \mathbf{vec}(X), \; e.g., \quad \mathbf{vec}\left(\begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}\right) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 11 \\ 22 \end{bmatrix}.$$

With vector representation, entries of the matrix X are determined with *linear indices*. For instance, the linear index of 11 is 5.

Bases

Considered as vectors, images belong to the space $\mathbb{R}^N$, where $N = m \times n$. Thus, we can also write:

$$\boldsymbol{x} = \sum_{i=1}^{N} x_i \boldsymbol{e}_i,$$

where $(\boldsymbol{e}_i)_i$ is the canonical basis of $\mathbb{R}^N$.

Images can also be represented in more appropriate bases, frames, or dictionaries $(\boldsymbol{\psi}_i)_{i \in I}$:

$$\boldsymbol{x} = \sum_{i \in I} x_i \boldsymbol{\psi}_i.$$

For instance, the discrete 2-D Fourier atoms $(\boldsymbol{\psi}_{\boldsymbol{k}})_{\boldsymbol{k}}$ are defined as:

$$\boldsymbol{\psi}_{\boldsymbol{k}}(x) = \frac{1}{\sqrt{N}} e^{2i\pi(x_1 \frac{k_1}{m} + x_2 \frac{k_2}{n})},$$

where $\boldsymbol{k} = (k_1, k_2)$ are the frequency indexes, with $0 \leqslant k_1 < m$, $0 \leqslant k_2 < n$. Figures (I.1) and (I.2) show some basis functions of the Fourier and Wavelets bases.

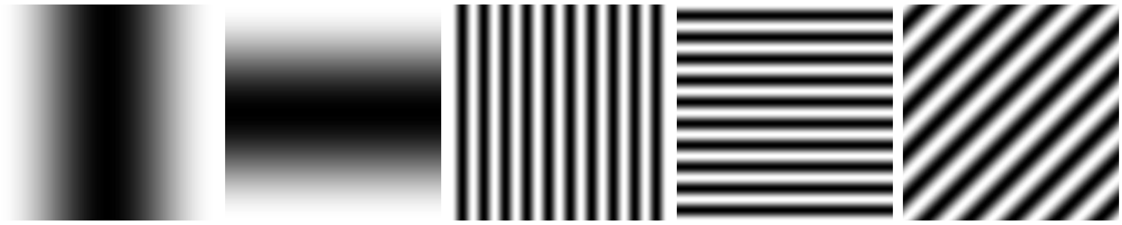


Fig. I.1: Some functions of a Fourier basis.

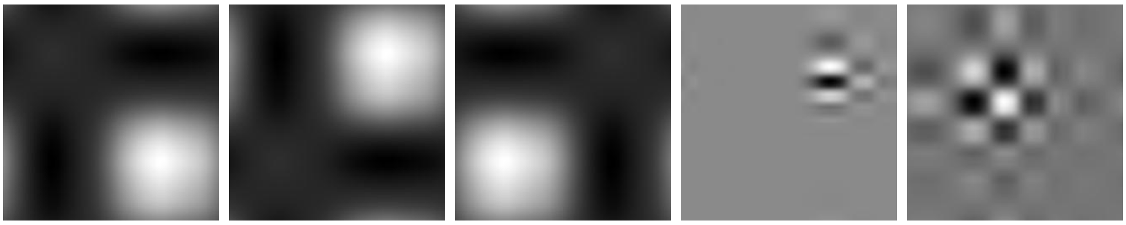


Fig. I.2: Some functions of a Wavelet (Daubechies) basis.

Figure (I.3) shows examples of reconstructions using the Fourier basis. In the Fourier decomposition,small coefficients bellow a chosen threshold are neglected. This can be considered as an **image compression** as we obtained a good image representation with 6554 coefficients only (13108 coefficients when adding also their linear indices), instead of $256 \times 256 = 65536$.

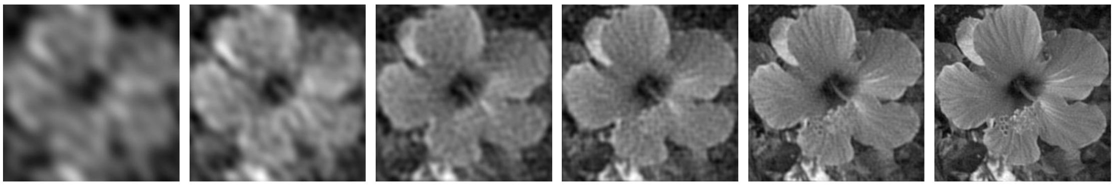


Fig. I.3: Image reconstruction using Fourier decomposition. From right to left, reconstructions with increasing number of coefficients, 109, 218, 655, 1311, 6554, chosen in descending order.

Convolution with vector representation

Convolution is an important operation frequently used in image processing. In the following, we give explicit formula of 2D Convolution computed with the 1D vector representation of images.

One-Dimensional convolution

We first consider one-dimensional convolution $\boldsymbol{b} = \boldsymbol{p} * \boldsymbol{x}$. For the sake of example, we consider $\boldsymbol{x}, \boldsymbol{b} \in \mathbb{R}^5$. It is easy to see that the vector $\boldsymbol{b}$ can be written as a matrix product:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_5 & p_4 & p_3 & p_2 & p_1 & & & & \\ & p_5 & p_4 & p_3 & p_2 & p_1 & & & \\ & & p_5 & p_4 & p_3 & p_2 & p_1 & & \\ & & & p_5 & p_4 & p_3 & p_2 & p_1 & \\ & & & & p_5 & p_4 & p_3 & p_2 & p_1 \end{bmatrix} \left[\begin{array}{c} w1 \\ w_2 \\ \hline x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \hline y_1 \\ y_2 \end{array}\right],$$

where the extra-variables (w_i and y_i) are chosen to fit the boundary conditions (**BC**). More precisely, for **zero BC**, we shall choose $w_i = y_i = 0$. Thus, we get:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \underbrace{\begin{bmatrix} p_3 & p_2 & p_1 & & \\ p_4 & p_3 & p_2 & p_1 & \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ & p_5 & p_4 & p_3 & p_2 \\ & & p_5 & p_4 & p_3 \end{bmatrix}}_{\text{Toeplitz matrix}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}.$$

For **periodic BC**, we shall choose $w_1 = x_4, w_2 = x_5, y_1 = x_1, y_2 = x_2$. We get:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \underbrace{\begin{bmatrix} p_3 & p_2 & p_1 & p_5 & p_4 \\ p_4 & p_3 & p_2 & p_1 & P_5 \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ p_1 & p_5 & p_4 & p_3 & p_2 \\ p_2 & p_1 & p_5 & p_4 & p_3 \end{bmatrix}}_{\text{circulant matrix}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}.$$

For **reflexive BC**, we have $w_1 = x_2, w_2 = x_1, y_1 = x_5, y_2 = x_4$ and

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_3 + p_4 & p_2 + p_5 & p_1 & & \\ p_4 + p_5 & p_3 & p_2 & p_1 & \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ & p_5 & p_4 & p_3 & p_2 + p_1 \\ & & p_5 & p_4 + p_1 & p_3 + p_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}.$$

Two-Dimensional convolution

The convolution operation for two-dimensional images is similar to the one-dimensional case. For example, let

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}; \quad \mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}; \quad \boldsymbol{b} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}.$$

For the two-dimensional convolution $\boldsymbol{b} = \boldsymbol{b} * \boldsymbol{b}$ and with zero boundary conditions we obtain:

$$\left[\begin{array}{c} b_{11} \\ b_{21} \\ b_{31} \\ \hline b_{12} \\ b_{22} \\ b_{32} \\ \hline b_{13} \\ b_{23} \\ b_{33} \end{array}\right] = \underbrace{\left[\begin{array}{ccc|ccc|ccc} p_{22} & p_{12} & & p_{21} & p_{11} & & & & \\ p_{32} & p_{22} & p_{12} & p_{31} & p_{21} & p_{11} & & & \\ & p_{32} & p_{22} & & p_{31} & p_{21} & & & \\ \hline p_{23} & p_{13} & & p_{22} & p_{12} & & p_{21} & p_{11} & \\ p_{33} & p_{23} & p_{13} & p_{32} & p_{22} & p_{12} & p_{31} & p_{21} & p_{11} \\ & p_{33} & p_{23} & & p_{32} & p_{22} & & p_{31} & p_{21} \\ \hline & & & p_{23} & p_{13} & & p_{22} & p_{12} & \\ & & & p_{33} & p_{23} & p_{13} & p_{32} & p_{22} & p_{12} \\ & & & & p_{33} & p_{23} & & p_{32} & p_{22} \end{array}\right]}_{\text{Toeplitz matrix with Toeplitz blocks (BTTB)}} \left[\begin{array}{c} x_{11} \\ x_{21} \\ x_{31} \\ \hline x_{12} \\ x_{22} \\ x_{32} \\ \hline x_{13} \\ x_{23} \\ x_{33} \end{array}\right],$$

2 Matrices

2.1 Norms

The standard norms on $\mathbb{R}^n$ are the ℓ_p-norms $(p \geqslant 1)$.

The *dual norm* $\|\cdot\|_*$ of a norm $\|.\|$ is defined on $\mathbb{R}^n$ as follows:

$$\|\boldsymbol{x}\|_* = \sup_{\boldsymbol{y} \neq 0} \frac{\boldsymbol{x}^T \boldsymbol{y}}{\|\boldsymbol{y}\|} = \sup_{\|\boldsymbol{y}\|=1} \boldsymbol{x}^T \boldsymbol{y}.$$

Dual norms satisfy a generalized Cauchy-Schwarz inequality

$$\left|\boldsymbol{x}^T\boldsymbol{y}\right| | \leq \|\boldsymbol{x}\|_*\|\boldsymbol{y}\|; \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n.$$

Dual norms are also defined for matrices:

$$\|A\|_{p,q} = \sup_{x\in\mathbb{R}^n, x\neq 0} \frac{\|Ax\|_p}{\|x\|_q}.$$

For example, we get the following norms.

$$\|A\|_{1,1} = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}|\,, \quad \|A\|_{\infty,\infty} = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}|\,,$$

$$\|A\|_{2,2} = \sqrt{\rho(A^*A)} = \sqrt{\rho(AA^*)} = \sigma_{max}(A),$$

where ρ denotes the spectral radius and σ_{max} the largest singular value of A.

We can also consider *mixed norms*:

$$\|A\|_{\ell_{p,q}} = \left(\sum_{j=1}^{n}\left(\sum_{i=1}^{n}|a_{ij}|^p\right)^{q/p}\right)^{1/q}.$$

For example, we have the Frobenius norm:

$$\|A\|_F = \|A\|_{\ell_{2,2}} = \left(\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2\right)^{1/2} = \sqrt{tr(A^*A)} = \sqrt{tr(AA^*)}.$$

2.2 Differentiation

Let $\boldsymbol{x}$ and $\boldsymbol{c}$ are $(n \times 1)$ vectors sand A an $(m \times n)$ matrix. Then $\boldsymbol{c}^T\boldsymbol{x}$ and $A\boldsymbol{x}$ are scalar and vector functions of $\boldsymbol{x}$, respectively, and we have

$f(\boldsymbol{x})$	$\nabla f(\boldsymbol{x})$
$\boldsymbol{c}^T x$	$\boldsymbol{c}$
$A\boldsymbol{x}$	A
$\boldsymbol{x}^T A\boldsymbol{x}$	$\boldsymbol{x}^T(A^T + A)$
$Tr(XA)$	A^T
$Tr(X^T A)$	A
$Tr(X^T AX)$	$(A + A^T)X$

2.3 Singular value decomposition

The *singular value decomposition (SVD)* is a factorization of a real or complex matrix that generalizes the eigendecomposition Any real matrix X of size $n \times p$ can be decomposed as

$$\underbrace{X}_{n\times p} = \underbrace{U}_{n\times n}\;\underbrace{\varSigma}_{n\times p}\;\underbrace{V^T}_{p\times p}, \tag{I.1}$$

- U (respect. V) is orthogonal; its columns are normalized eigenvectors of XX^T (respec. X^TX).
- The columns $\mathbf{u}_j$ and $\mathbf{v}_j$ of the matrices U and V respectively share the same eigenvalue λ_j.
- $\sigma_j = \sqrt{\lambda_j}$ are the singular values, $\sigma_1 \geqslant \cdots \geqslant \sigma_r$, $k = \min\{m, n\}$, and we have

$$\varSigma = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \\ & \bigcirc & \end{pmatrix}, \quad \text{or} \quad \varSigma = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \bigcirc \\ & & \sigma_r \end{pmatrix}.$$

- Every matrix X of rank s has exactly s nonzero singular values.

The SVD has the following geometric interpretation. The matrix X can be seen as a linear transformation which can be decomposed in three sub-transformations: 1. rotation U, 2. re-scaling D, 3. rotation V.

The decomposition (I.1) is called *Full SVD*. Truncating the full SVD till the rank s, we can write a *compact* or *reduced SVD* as

$$X = U_1 \varSigma_1 V_1^T = \sum_{i=1}^{s} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T.$$

where $U_1 = [\boldsymbol{u}_1 \cdots \boldsymbol{u}_s]$ and $V_1 = [\boldsymbol{v}_1 \cdots \boldsymbol{v}_s]$.

2.3.1 Computing the compact SVD

It is computationally inefficient to compute a full SVD. Algorithm I.1 explains the steps to compute a compact SVD.

Example 2.1. *Compute the SVD of* $A = \begin{bmatrix} 2 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & -2 & 0 \end{bmatrix}$

- a. Find the eigenvalues of A^TA and the singular values for A.

$$\det(A^TA - \lambda I) = \begin{vmatrix} 8-\lambda & 2 & 0 \\ 2 & 5-\lambda & 0 \\ 0 & 0 & -\lambda \end{vmatrix} = \lambda(\lambda-4)(\lambda-9)$$

Thus, $\lambda_1 = 9$, $\lambda_2 = 4$ and $\lambda_3 = 0$. So the singular values are 3,2 and 0.

Algorithm I.1: Computing the compact SVD

```
Data: Matrix $A$
/* Calculate the eigenvalues and eigenvectors of $A^{\mathsf{H}}A$.            */
$\boldsymbol{\lambda}, V \leftarrow \text{eig}(A^{\mathsf{H}}A)$
/* Calculate the singular values of $A$.                                          */
$\boldsymbol{\sigma} \leftarrow \sqrt{\boldsymbol{\lambda}}$
/* Sort the singular values from greatest to least.                               */
$\boldsymbol{\sigma} \leftarrow \text{sort}(\boldsymbol{\sigma})$
/* Sort the eigenvectors the same way.                                            */
$V \leftarrow \text{sort}(V)$
/* Keep only the positive singular values.                                        */
$\boldsymbol{\sigma}_1 \leftarrow \boldsymbol{\sigma}_{:r}$
/* Keep only the corresponding eigenvectors.                                      */
$V_1 \leftarrow V_{:,:r}$
/* Construct $U$ colum vy column (Can we compute full SVD?).                      */
$U_1 \leftarrow AV_1/\boldsymbol{\sigma}_1$
/* Output                                                                         */
return $U_1, \boldsymbol{\sigma}_1, V_1^{\mathsf{H}}$
```

$$\Sigma = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- b. Find $V = (e_1, e_2, e_3)$. e_j is the unit vector corresponding to the jth eigenvalue.

$$V = \begin{bmatrix} \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{5}} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Find U by using the formula $AV = U\Sigma$. We can write $\boldsymbol{u}_j = \frac{1}{\sigma_j} A\boldsymbol{v}_j$

  Finally, we get:

$$A = U\Sigma V^T = \begin{bmatrix} \frac{4}{3\sqrt{5}} & \frac{1}{\sqrt{5}} & 0 \\ \frac{\sqrt{5}}{3} & 0 & 0 \\ -\frac{2}{3\sqrt{5}} & \frac{2}{\sqrt{5}} & 0 \end{bmatrix} \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{5}} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.3.2 Applications

Solving linear systems In the next chapter, we will show how the SVD can be used to solve linear systems in various situations.

Computing Pseudoinverse For $A \in \mathbb{R}^{m \times n}$, a *pseudoinverse* of A is defined as a matrix $A^+ \in \mathbb{R}^{n \times m}$ satisfying the following Moore-Penrose conditions:

$$AA^+A = A,\ A^+AA^+ = A^+,\ (AA^+)^* = AA^+,\ (A^+A)^* = A^+A.$$

The pseudoinverse A^+ exists for any matrix A, but when the latter has full rank, A^+ can be expressed using simple algebraic formula.

When A has linearly independent columns (A^*A is invertible), A^+ constitutes a left inverse ($A^+A = I$) and we can write:

$$A^+ = (A^*A)^{-1}A^*.$$

On the other hand, when A has linearly independent rows (AA^* is invertible), A^+ is a right inverse and we have:

$$A^+ = A^*(AA^*)^{-1}.$$

Using the SVD decomposition, we get:

$$X^+ = V \begin{pmatrix} d_1^{-1} & & \\ & \ddots & \bigcirc \\ & d_p^{-1} & \end{pmatrix} U^T$$

Low-Rank Approximations: Image compression. The SVD makes easy the construction of low-rank approximations of matrices and is therefore the basis of several data compression algorithms.

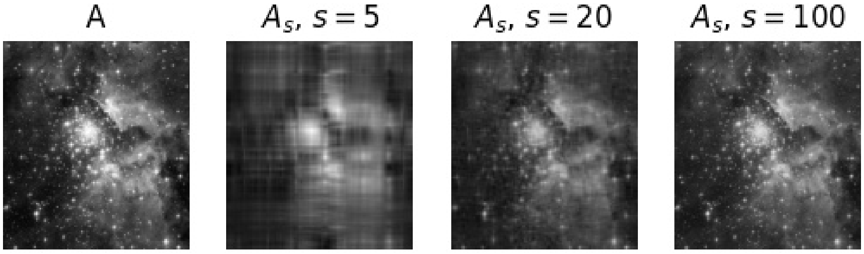
Let A be a $m \times n$ matrix of rank $r < min\{m, n\}$ and $A = U\Sigma V^T$. We can perform an image compression by storing matrices U_1, Σ_1 and V_1 instead of storing the matrix A. In the three matrices, we keep only the first k values and vectors.

Indeed, A contains mn values. But, storing the matrices U_1, Σ_1 and V_1 require $mr + r + nr$ values only. For instance, for a 100×200 matrix A of rank 20, storing A requires $20,000$ values while storing the matrices U_1, Σ_1 and V_1 requires $6,020$ entries only.

We can achieve better compression performance by using a *truncated SVD*, i.e., by keeping only the first $s < r$ singular values, plus the corresponding columns of U and V. The matrix A can be approximated by

$$A_s \approx \sum_{i=1}^{s} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^{\mathsf{H}}.$$

The resulting matrix A_s has rank s and is only an approximation to A, since $r - s$ nonzero singular values are neglected.



3 Algebraic methods for image reconstruction

3.1 Degradation model

We would like to model some linear image degradation operators. We suppose that there exists a large $N \times N$ matrix $\mathbf{A}$ such that we can write

$$\boldsymbol{x} = \mathbf{A}\boldsymbol{y} + \boldsymbol{\epsilon}, \tag{I.2}$$

where $\boldsymbol{y}$ is the observed image, $\boldsymbol{x}$ is the unknown "ideal" image, and $\boldsymbol{\epsilon}$ is a noise (usually Gaussian). Note that we distinguish between the degradation $N \times N$ matrix $\mathbf{A}$ and the $m \times n$ image arrays $X, Y \in \mathbb{R}^{n \times m}$.

The model (I.2) can describe various types of degradation.

Noise

$\mathbf{A} = \mathbb{1}$.

Bluring

$\mathbf{A} =$ convolution with a Gaussian mask.

Inpainting

$\mathbf{A} =$ Multiplication with a binary matrix.

Estimation a PSF

$\mathbf{A}$ can be defined and estimated as the *point spread function (PSF)*. Each column of

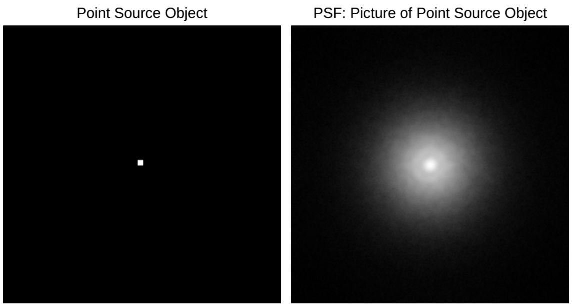


Fig. I.4

$\mathbf{A}$ can be written as $\mathbf{A}\mathbf{e}_j = \mathbf{a}_j$. One can construct $\mathbf{A}$ by generating all its columns, i.e., it can be estimated by imaging "point source" objects, for all different positions of the source point, corresponding to image pixels.

3.2 Algebraic solutions

A *naive approach* fo image reconstruction consists in inverting the matrix $\mathbf{A}$:

$$\hat{\boldsymbol{x}} = \mathbf{A}^{-1}\boldsymbol{y}.$$

This approach fails because of the noise. We explain here an *improved SVD-based method.*

We have

$$\begin{aligned} A &= U\Sigma V^T, \\ &= u_1\sigma_1 v_1^T + \cdots + u_N\sigma_N v_k^T, \\ &= \textstyle\sum_{i=1}^k \sigma_i u_i v_i^T. \end{aligned}$$

Similarly,

$$A^{-1} = V\Sigma^{-1}U^T = \sum_{i=1}^k \frac{1}{\sigma_i} v_i u_i^T$$

The error term of the naive solution becomes

$$A^{-1}\boldsymbol{\epsilon} = V\Sigma^{-1}U^T e = \sum_{i=1}^k \frac{u_i^T\boldsymbol{\epsilon}}{\sigma_i} v_i$$

For all i, the error components $|u_i^T e|$ are small and roughly of the same order of magnitude. Dividing by small singular values increases the corresponding error component. In addition, singular vectors corresponding to small singular values represent high frequency information. Thus, we discard the terms corresponding to small singular values to get better reconstruction results.

4 Optimization problems

Most image processing mathematical models considered here in this book are based on energy minimization. In such approaches, given an observed image $\boldsymbol{y}$, we "estimate" $\boldsymbol{x}$ using the following criterion:

$$\begin{aligned} \hat{\boldsymbol{x}} &= \underset{\boldsymbol{x}}{\operatorname{argmin}} \;\; J(\boldsymbol{x}, \boldsymbol{y}), \\ &= \underset{\boldsymbol{x}}{\operatorname{argmin}} \;\; h(\boldsymbol{x}, \boldsymbol{y}) + \tau\varphi(\boldsymbol{x}), \\ &= \underset{\boldsymbol{x}}{\operatorname{argmin}} \;\; f(\boldsymbol{x}) + \tau\varphi(\boldsymbol{x}). \end{aligned}$$

- The **data term** $h(\boldsymbol{x}, \boldsymbol{y})$ describes how well $\boldsymbol{x}$ "fits"/"explains" the data $\boldsymbol{y}$. A typical example is when there exists a theoretical functional relation between $\boldsymbol{x}$ and $\boldsymbol{y}$, of the form: $\boldsymbol{y} = F(\boldsymbol{x})$. In this case, one can choose $h(\boldsymbol{x}, \boldsymbol{y}) = \|F(\boldsymbol{x}) - \boldsymbol{y}\|$.

- The **regularization term** $\varphi(\boldsymbol{x})$ encodes some knowledge/constraints/structure. Common regularizers impose/encourage one, or a combination of the following characteristics: small norm (vector or matrix), sparsity (few nonzeros), low-rank (matrix), smoothness or piece-wise smoothness, ... Adding a regularization term to the data term to from one single criterion is called *Tikhonov regularization*. We shall also point out other forms or regularization, "equivalent" under mild conditions, which are:

- Morozov regularization: $\underset{\boldsymbol{x}}{\operatorname{argmin}}\ \varphi(\boldsymbol{x}),\ s.t.,\ f(\boldsymbol{x}) \leq \varepsilon$
- Ivanov regularization: $\underset{\boldsymbol{x}}{\operatorname{argmin}}\ f(\boldsymbol{x}),\ s.t.,\ \varphi(\boldsymbol{x}) \leq \delta.$

Morozov and Ivanov can be written as Tikhonov using indicator functions.

- The **regularization weight** $\tau \geq 0$ controls the importance given to each term of the function J.

For instance, let $Y \in \mathbb{R}^{n\times m}$ be a given noisy image. A model for image denoising consists in solving the following optimization problem:

$$\min_{X} \ \|Y - X\|_{2,2}^2 + \lambda\varphi(X) \tag{I.3}$$

where $\varphi(X) = \sum\limits_{i=1}^{n-1}\sum\limits_{j=1}^{m-1} ((x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2)$. We should mention here that we used the matrix representation of images.

Départ. Maths-Info. 1^e Master Maths Traitement d'images

Série d'exercices 1

Mars 2020

Introduction

Exercice 1

Derive the matrices associated to the following differential operators and for a 3 × 3 image: D_x, D_y, $\mathit{\Delta}$.

Exercice 2

Perform a SVD decomposition of the following matrices :

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ -1 & 1 \end{pmatrix}.$$

Exercice 3

- Does the matrix $A = \begin{pmatrix} 4 & 3 \\ 8 & 6 \end{pmatrix}$ have a left inverse? A right inverse? A pseudoinverse? If the answer to any of these questions is "yes", find the appropriate inverse.
- Prove that the left singular vectors of A are the right singular vectors of A^T.

Exercice 4

Let $\boldsymbol{x}$ and $\boldsymbol{c}$ are $(n \times 1)$ vectors sand A an $(m \times n)$ matrix. Prove the following relations.

$f(\boldsymbol{x})$	$\nabla f(\boldsymbol{x})$
$\boldsymbol{c}^T x$	$\boldsymbol{c}$
$A\boldsymbol{x}$	A
$\boldsymbol{x}^T A\boldsymbol{x}$	$\boldsymbol{x}^T(A^T + A)$
$Tr(XA)$	A^T
$Tr(X^T A)$	A
$Tr(X^T AX)$	$(A + A^T)X$

Chapter II
Unconstrained and constrained optimization

Skills to acquire

- **Fundamental results about existence of minimizers.**
- **Lagrange multipliers for constrained optimization problems**.
- **Least squares problems**.

1 Generalities

Let U be an open subset of $\mathbb{R}^n$ and $f : U \subset \mathbb{R}^n \to \mathbb{R} \cup \{-\infty, +\infty\}$. We aim at minimizing f, i.e., finding some $v \in U$ such that

$$v = \underset{u \in U}{\operatorname{argmin}} \ f(x). \tag{II.1}$$

Definition 1.1 (Sublevel and epigraph)**.** *A c-sublevel set of f is:*

$$S_c f = \{x \in \operatorname{dom}(f) : f(x) \leq c\}.$$

The epigraph of f is defined by

$$\operatorname{epi} f = \{(x, y) \in \mathbb{R}^{n+1} : f(x) \leq y\}.$$

f is said to be closed if its epigraph is closed.

Proposition 1.1. *f is closed if and only if all its sublevel sets are closed.*

Proposition 1.2. *If f is closed with bounded sublevel sets then it has a minimizer.*

Semi-continuity is a property of extended real-valued functions that is weaker than continuity. A function f is upper (respectively, lower) semi-continuous at a point x_0 if, roughly speaking, the function values for arguments near x_0 are not much higher (respectively, lower) than $f(x_0)$.

Definition 1.2 (Semi-continuity). *f is lower semi-continuous (l.s.c.) at $x_0 \in \operatorname{dom}(f)$ if*

$$\forall \epsilon > 0, \exists \delta > 0 : f(x) \geq f(x_0) - \epsilon, \forall x \in B_\delta(x_0)$$

or, for metric spaces

$$f(x_0) \leq \lim \inf_{x \to x_0} f(x).$$

The function f is l.s.c. if it is for every $x_0 \in domf$.

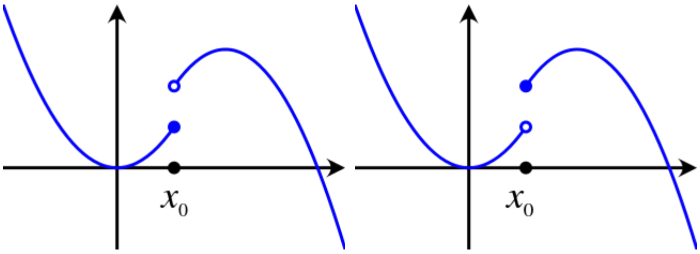


Fig. II.1: Lower and upper semi-continuous

Theorem 1.1. *f is closed $\iff$ f is l.s.c.*

Theorem 1.2 (Weierstrass theorem for closed functions). *Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper closed function and assume that C is a compact set satisfying $C \cap \operatorname{dom}(f) \neq \emptyset$. Then f is bounded below over C and attains its minimal value over C.*

The compactness of C is replaced by closedness if the function has a property called coerciveness.

Definition 1.3 (coercivity). *A proper function $f : \mathbb{R}^n \to (-\infty, +\infty]$ is called coercive if*

$$\lim_{\|x\| \to +\infty} f(x) = +\infty.$$

Theorem 1.3 (attainment under coercivity). *Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper closed and coercive function and let $S \subseteq \mathbb{R}^n$ be a nonempty closed set satisfying $S \cap \operatorname{dom}(()f) \neq \emptyset$. Then f attains its minimal value over S.*

2 Unconstrained problem

Consider the following optimization problem:

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}) \tag{II.2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is sufficiently differentiable.

The next two theorems give fundamental results of existence and characterization of minimizers.

Theorem 2.4 (First-order necessary conditions). *If $\bar{\boldsymbol{x}}$ is a local minimizer of f and f is continuously differentiable in an open neighborhood of $\bar{\boldsymbol{x}}$, then $\nabla f(\bar{\boldsymbol{x}}) = 0$.*

Theorem 2.5 (Second-order conditions). *Suppose that $\nabla^2 f$ is continuous in an open neighborhood of $\bar{\boldsymbol{x}}$. If $\bar{\boldsymbol{x}}$ is a local minimizer of f then $\nabla f(\bar{\boldsymbol{x}}) = 0$ and $\nabla^2 f$ is positive semi-definite. Conversely, if $f(\bar{\boldsymbol{x}}) = 0$ and $\nabla^2 f$ is positive definite, then $\bar{\boldsymbol{x}}$ is a strict local minimizer of f.*

Example 2.1 (Mean and median). *Consider $(x_i)_{i=1,\dots,n} \in \mathbb{R}$. We would like to compute the best value $\bar{x} \in \mathbb{R}$ which approximates the whole set, i.e., which minimizes the following function:*

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} (x - x_i)^2.$$

This criterion is called the Mean Squared Error *(MSE). From the necessary and sufficient conditions, one can easily show that the solution is the arithmetic mean*

$$\boxed{\bar{x} = \frac{x_i}{n}}.$$

If we consider the Mean Absolute Error *(MAE):*

$$f(x) = \sum_{i=1}^{n} |x - x_i|,$$

and restrict the search space to the set $\{x_i, ..., x_i\}$, we can prove that the solution is the median value:

$$\boxed{\bar{x} = \text{median}\{x_1, \cdots, x_n\}}.$$

Indeed, let i such that $x \in [x_i, x_{i+1}]$. We have:

$$f(x) = \sum_{j \leqslant i} (x - x_j) + \sum_{j > i} (x_j - x)$$

which is a linear function (when restricted to the interval $[x_i, x_{i+1}]$; with gradient equal to the number of j such that $j \leqslant i$ minus the the number of j such that $j > i$. Thus the gradient is zero if it is less than a median, it is positive if greater than a medium, and zero precisely when there are as many data points to the left and right. In other words, x decreases as we approach a medium from below, stays constant on the set of mediums, and then decreases afterwards. So minimums occurs exactly at mediums.

Example 2.2 (Linear regression). *Consider $(x_i, y_i)_{i=1,\dots,n} \in \mathbb{R}^2$. We would like to approximate this point cloud by an affine function $y = ax + b$, i.e., we seek $(a, b) \in \mathbb{R}^2$ which minimizes the following function:*

$$J(a, b) = \sum_{i=1}^{n} (a x_i + b - y_i)^2.$$

First order conditions

$$\begin{cases} \frac{\partial J}{\partial a} & = 2\sum_{i=1}^{n} (a x_i + b - y_i) x_i = 0, \\ \frac{\partial J}{\partial b} & = 2\sum_{i=1}^{n} (a x_i + b - y_i) = 0. \end{cases}$$

We can write

$$\begin{cases} a\sum_{i=1}^n x_i^2 \,+\, b\sum_{i=1}^n x_i \,-\, \sum_{i=1}^n x_i y_i & = 0 \\ a\sum_{i=1}^n x_i \,+\, nb \,-\, \sum_{i=1}^n y_i & = 0 \end{cases}$$

We use the following notations:

$$\bar{x} = \sum_{i=1}^n \frac{x_i}{n}, \quad s_x = \sum_{i=1}^n x_i, \quad s_{xx} = \sum_{i=1}^n x_i^2, \quad s_{xy} = \sum_{i=1}^n x_i y_i, \quad s_y = \sum_{i=1}^n y_i,$$

So, we obtain

$$\begin{cases} s_{xx}\, a \,+\, s_x\, b \,-\, s_{xy} & = 0 \\ s_x\, a \,+\, n\, b \,-\, s_y & = 0 \end{cases}$$

$$\boxed{b = \frac{s_x\, s_{xy} - s_{xx}\, s_y}{s_x^2 - n s_{xy}}, \quad a = \frac{s_y}{s_x} - \frac{n}{s_x}\, b.}$$

Second order conditions

$$J''(a,b) = \begin{bmatrix} s_{xx} & s_x \\ s_x & n \end{bmatrix}$$

We note that $s_{xx} > 0$ *and, thanks to the Cauchy-Schwarz inequality*

$$\left(\sum_{i=1}^n u_i v_i\right)^2 \leqslant \left(\sum_{i=1}^n u_i^2\right)\left(\sum_{i=1}^n v_i^2\right),$$

with $u = (x_1, ..., x_n)$ *and* $v = (1, ..., 1)$, *we get:*

$$\left(\sum_{i=1}^n x_i\right)^2 \leqslant \left(\sum_{i=1}^n x_i^2\right) n, \text{ i.e., } s_x^2 \le n\, s_{xx}.$$

From the previous equations, we can deduce that $J''(x)$ *is positive definite.*

Example 2.3 (Linear regression)**.** *We consider the linear regression problem*

$$z = \beta_0 + \beta_1 x + \beta_2 x^2,$$

i.e., we have a dataset $(x_i, y_i)_{i=1,\dots,n} \in \mathbb{R}^2$ *and we want to estimate the parameters* β_0, β_1 *and* β_2. *We put*

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 \\ \vdots & \vdots & \\ 1 & x_n & x_n^2 \end{bmatrix}; \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} \; ; \; \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}.$$

The loss criterion J *can be written as*

$$\begin{aligned} J(\beta) &= \sum_{i=1}^n (y_i - \beta_2 x_i^2 - \beta_1 x_i - \beta_0)^2, \\ &= \|X\beta - \mathbf{y}\|^2. \end{aligned}$$

We write the first and second order conditions:

$$\frac{\partial J}{\partial \beta} = 2(X^T X)\beta - 2X^T\mathbf{y} = 0, \quad \text{and} \quad J''(\beta) = 2(X^T X).$$

So, the minimizer is the solution of the linear system

$$\boxed{X^T X\, \beta \;=\; X^T\mathbf{y}}.$$

3 Equality constrained problem

Consider the following equality constrained problem:

$$\begin{aligned} \underset{\boldsymbol{x} \in \mathbb{R}^n}{\text{minimize}} \quad & f(\boldsymbol{x}), \\ \text{subject to} \quad & h_i(\boldsymbol{x}) = 0, \, i = 1, \dots, m. \end{aligned} \tag{II.3}$$

where $h_i : \mathbb{R}^n \to \mathbb{R}$ are m constraints. We assume that h_i once or twice differentiable and put $\boldsymbol{h} = (h_1, \cdots, h_m)$.

To solve this problem, we can use the method of *Lagrange Multipliers*, which puts the cost function as well as the constraints in a single minimization problem:

$$\boldsymbol{x}^\star = \underset{\boldsymbol{x}}{\text{argmin}} \underbrace{f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x})}_{\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})} . \tag{II.4}$$

where $\mathcal{L}(\mathbf{x}, \lambda)$ is called the *Lagrangian* and $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m) \in \mathbb{R}^m$ is a vector of *multipliers*.

Definition 3.4. *A point $\bar{\boldsymbol{x}} \in \mathbb{R}^n$ is called feasible when it satisfies all the constraints, i.e. $\boldsymbol{h}(\bar{\boldsymbol{x}}) = 0$. We say that a feasible point $\boldsymbol{x}$ is regular when $\nabla h_1(\boldsymbol{x})$, ..., $\nabla h_m(\boldsymbol{x})$ are linearly independent.*

The next theorem gives necessary conditions for the minimization problem (II.3).

Theorem 3.6 (First-Order Necessary Conditions)**.** *If a **regular** point $\boldsymbol{x}$ is a local minimizer of* (II.3)*, then there exists $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that*

$$\begin{cases} \nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \lambda) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) + \sum_i \lambda_i \nabla_{\boldsymbol{x}} h_i(\boldsymbol{x}) = 0, \\ \frac{\partial \mathcal{L}(\boldsymbol{x}, \lambda)}{\partial \lambda_i} \quad = \quad h_i(\boldsymbol{x}) \quad = 0, \, i = 1, \dots, m. \end{cases} \tag{II.5}$$

Solutions of the optimality conditions are called *stationary or critical points*. Solving the optimality conditions gives also the Lagrange multipliers associated with the critical points.

> **Remark:** a local minimizer which is not a regular point might not fulfill the above optimality conditions.

Theorem 3.7 (Second-Order Conditions)**.** *Suppose that $\boldsymbol{x}$ is a stationary point of* (II.3)*. The, if $\boldsymbol{x}$ is a local minimizer, then Hessian of the Lagrangian, $H = \nabla^2_{\boldsymbol{xx}}(\mathcal{L}(\boldsymbol{x}, \lambda))$, is positive semidefinite on the tangent space:*

$$T = \left\{ \boldsymbol{y} \;\middle|\; \nabla h_j(\boldsymbol{x}) \boldsymbol{y} = 0, j = 1, \cdots, m \right\} .$$

Conversely, it H is positive definite on T and the first order conditions are satisfied, the $\boldsymbol{x}$ is a local minimizer.

Example 3.1. *Consider the problem*

$$\begin{aligned} \underset{(x,y) \in \mathbb{R}^2}{\text{minimize}} \quad & f(x, y) = 5x - 3y \\ \text{subject to} \quad & x^2 + y^2 = 136 \end{aligned}$$

The optimality conditions yield:

$$\begin{cases} 5 & = 2\lambda x, \\ -3 & = 2\lambda y, \\ x^2 + y^2 & = 136. \end{cases}$$

This system has two solutions: $(x_1, y_1, \lambda_1) = (-10, 6, -\frac{1}{4})$ *and* $(x_2, y_2, \lambda_2) = (10, -6, \frac{1}{4})$. *Now, the Hessian of the Lagrangian is*

$$H = \begin{pmatrix} 2\lambda & 0 \\ 0 & 2\lambda \end{pmatrix}.$$

So, (x_1, y_1, λ_1) *is a minimizer while* (x_2, y_2, λ_2) *is a maximizer.*

Example 3.2. *Consider the following problem:*

$$\begin{aligned} \underset{(x_1,x_2)\in\mathbb{R}^2}{\text{minimize}} \quad & f(x_1, x_2) = x_1 + x_2 \\ \text{subject to} \quad & h_1(x_1, x_2) = (x_1 - 1)^2 + x_2^2 - 1 = 0, \\ & h_2(x_1, x_2) = (x_1 - 2)^2 + x_2^2 - 4 = 0 \end{aligned}$$

The solution is $(x_1, x_2) = (0, 0)$. *The optimality condition writes:*

$$\nabla_x f(x) + \lambda_1 \nabla_x h_1(x) + \lambda_2 \nabla_x h_2(x) = 0.$$

However, the gradients at the origin are $\nabla_x f(0, 0) = (1, 1)$, $\nabla_x h_1(0, 0) = (-2, 0)$, $\nabla_x h_2(0, 0) = (-4, 0)$. *Therefore, there are no Lagrange multipliers that enforce the optimality condition.*

4 Inequality constrained problem

Consider the following problem:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^n}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & h_i(\boldsymbol{x}) = 0, \forall\, i = 1, \dots, m, \\ & g_j(\boldsymbol{x}) \le 0, \forall\, j = 1, \dots, p. \end{aligned} \tag{II.6}$$

$h_i, g_j : \mathbb{R}^n \to \mathbb{R}$ supposed to be differentiable. We put $\boldsymbol{h} = (h_1, \cdots, h_m)$ and $\boldsymbol{g} = (g_1, \cdots, g_p)$.

When the constraints also have inequalities, the Lagrange Multipliers method is extended to the *KKT conditions*. The expression for the Lagrangian becomes:

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ \mathcal{L}(\boldsymbol{x}, \lambda, \mu) = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x}) + \sum_{i=1}^{\ell} \mu_i g_i(\boldsymbol{x}), \tag{II.7}$$

where $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)$ and $\boldsymbol{\mu} = (\mu_1, \dots, \mu_\ell)$ are vectors of the multipliers.

Definition 4.5. *A point $\bar{\boldsymbol{x}} \in \mathbb{R}^n$ is called feasible when it satisfies all the constraints, i.e.* $\boldsymbol{h}(\bar{\boldsymbol{x}}) = 0$ *and* $\boldsymbol{g}(\bar{\boldsymbol{x}}) \leqslant 0$. *We say that a feasible point* $\boldsymbol{x}$ *is regular when* $\nabla h_1(\boldsymbol{x})$, $\ldots$, $\nabla h_m(\boldsymbol{x})$ *are linearly independent.*

Definition 4.6. *An inequality constraint* $g_j(\boldsymbol{x}) \leqslant 0$ *is said to be active at* $\boldsymbol{x}^*$ *if* $g_j(\boldsymbol{x}^*) = 0$ *and inactive if* $g_j(\boldsymbol{x}^*) < 0$.

> **Remark:** Equality constraint $h_i(\boldsymbol{x}) = 0$ is considered to be always active.

Definition 4.7. *A feasible point* $\boldsymbol{x}$ *is said to be regular if gradients of all active constraints are linearly independent.*

As previously, we give necessary and sufficient conditions for optimality of problem (II.6).

Theorem 4.8 (Karush-Kuhn-Tucker (KKT) theorem)**.** *If* $\boldsymbol{x}$ *is a regular point and a local minimizer for the problem* (II.6)*, then there exists* $\boldsymbol{\lambda} \in \mathbb{R}^m$, $\boldsymbol{\mu} \in \mathbb{R}^p$ *such that*

- *Stationarity:* $\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \lambda, \mu) = 0$.
- *Dual feasibility:* $\boldsymbol{\mu} \geq 0$.
- *Complementary slackness:* $\mu_j g_j(\boldsymbol{x}) = 0, \forall j$*, or, equivently,* $\boldsymbol{\mu}^T g(\boldsymbol{x}) = 0$.

> **Remark:** The fact that $\boldsymbol{x}$ is supposed to be a regular point adds the constraints:
>
> $$\boldsymbol{h}(\boldsymbol{x}) = 0; \quad \boldsymbol{g}(\boldsymbol{x}) \leqslant 0.$$

Theorem 4.9 (Second order conditions)**.** *If* $\boldsymbol{x}$ *is a regular point and a local minimizer for the problem* (II.6)*, then the Hessian of the Lagrangian,* $H = \nabla^2_{\boldsymbol{xx}}(\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$*, is positive semidefinite on the tangent space:*

$$T(\boldsymbol{x}) = \{\boldsymbol{y} \in \mathbb{R}^n \ : \ D\boldsymbol{h}(\boldsymbol{x})\boldsymbol{y} = 0, \ Dg_j(\boldsymbol{x})\boldsymbol{y} = 0 \ \textit{when} \ g_j(\boldsymbol{x}) = 0.\}$$

Conversely, if H *is positive definite on the space:*

$$\bar{T}(\boldsymbol{x}) = \{\boldsymbol{y} \in \mathbb{R}^n \ : \ D\boldsymbol{h}(\boldsymbol{x})\boldsymbol{y} = 0, \ Dg_j(\boldsymbol{x})\boldsymbol{y} = 0 \ \textit{when} \ g_j(\boldsymbol{x}) = 0, \mu_j > 0\},$$

and the first order conditions are satisfied, the $\boldsymbol{x}$ *is a local minimizer.*

Example 4.1. *Consider the following minimization problem*

$$\begin{array}{cl} \underset{\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2}{\text{minimize}} & \|\boldsymbol{x}\|_2 \\ \text{subject to} & x_1 + x_2 + 1 \ \leq 0 \end{array}$$

We find the solution of the KKT conditions.

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \sqrt{x_1^2 + x_2^2} + \mu(x_1 + x_2 + 1)$$

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial x_1} = \frac{x_1}{\|\boldsymbol{x}\|_2} + \mu = 0 \\ \frac{\partial \mathcal{L}}{\partial x_2} = \frac{x_2}{\|\boldsymbol{x}\|_2} + \mu = 0 \\ \mu(x_1 + x_2 + 1) = 0 \\ x_1 + x_2 + 1 \quad \leq 0 \\ \mu \quad \geq 0 \end{cases}$$

So, the unique stationary point is $\boldsymbol{x}^* = -\begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}$. *It is easy to check the second order sufficient condition (first, determine* $\bar{T}(\boldsymbol{x})$*).* x^* *is a minimizer.*

Example 4.2. *Solve the following minimization problem*

$$\begin{array}{cc} \underset{\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2}{\text{minimize}} & 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2 \\ \text{subject to} & x_1^2 + x_2^2 \leqslant 5, \\ & 3x_1 + x_2 \leqslant 6 \end{array}$$

The solution is $x_1 = 1$, $x_2 = 2$, $\lambda_1 = 1$, $\lambda_2 = 0$.

> **Remark:** Strict inequalities cannot be taken into account in KKT conditions. To solve the problem, one need to compute stationary points of the problem without these constraints, then throw away all of them that violate these conditions.

5 Least squares

Consider the system of linear equations

$$A\mathbf{x} = \mathbf{y}, \tag{II.8}$$

where $A \in \mathbb{R}^{n \times m}$, $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^n$, and $n \neq m$, i.e., the problem is ill-posed. We give in the following some techniques to solve this problem.

5.1 Overdetermined linear systems

Assume that $n > m$ (more observations than variables). The problem has no solution if the data are not redundant. In this case, it is common to seek a solution $\mathbf{x}$ minimizing a Least squares loss:

$$J(\mathbf{x}) = \|\mathbf{y} - A\mathbf{x}\|_2^2.$$

Setting the derivative to zero, we obtain the **normal equation**:

$$\frac{\partial}{\partial \mathbf{x}} J(\mathbf{x}) = 0 \implies A^T A\mathbf{x} = A^T\mathbf{y}$$

When $A^T A$ is invertible, then the solution:

$$\mathbf{x} = \left(A^T A\right)^{-1} A^T,$$

$$\mathbf{x} = A^+ \mathbf{y}, \tag{II.9}$$

where A^+ is the pseudinverse. This solution is also valid when $A^T A$ is not invertible (using the SVD decomposition, we can compute A^+).

5.2 Under-constrained linear systems

Assume that $n < m$ (more variables than observations). The problem may have infinite solutions and needs to be constrained.

For instance, we can be interested in finding the one with least ℓ_2-norm, i.e., solution of the following problem:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \|\boldsymbol{x}\|_2^2, \\ \text{subject to} \quad & \boldsymbol{y} - A\boldsymbol{x} = 0. \end{aligned}$$

Setting the problem in Lagrangian form:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \|\boldsymbol{x}\|_2^2 + \lambda^T(\boldsymbol{y} - A\boldsymbol{x}) = \|\boldsymbol{x}\|_2^2 + \lambda^T \boldsymbol{y} - (A^T \lambda)^T \boldsymbol{x}.$$

The optimality conditions yield:

$$\begin{cases} 2\boldsymbol{x} - A^T \boldsymbol{\lambda} = 0 \\ \boldsymbol{y} \qquad\quad\; = A\boldsymbol{x} \end{cases}$$

If AA^T is invertible, we get: $\boldsymbol{\lambda} = 2(AA^T)^{-1}\boldsymbol{y}$ and

$$\boldsymbol{x} = A^T (AA^T)^{-1} \boldsymbol{y},$$

i.e.,

$$\boldsymbol{x} = A^+ \boldsymbol{y}.$$

Again, this solution is also valid when $A^T A$ is not invertible.

Tikhonov regularization

To cope with the ill-posedness of the problem II.8, another approach is to minimize the weighted sum

$$J(\boldsymbol{x}) = \|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_2^2,$$

with $\lambda > 0$. Setting the derivative to zero,

$$\boldsymbol{x} = \left(A^T A + \lambda I\right)^{-1} A^T \boldsymbol{y}.$$

This is referred to as "diagonal loading" because a constant, λ, is added to the diagonal elements of $A^T A$. The matrix $A^T A + \lambda I$ is invertible even if $A^T A$ is not. Indeed, it is easy to show that if α is a eigenvalue of a matrix M, the $\alpha + \lambda$ is an eigenvalue of $M + \lambda I$. As $A^T A$ is symmetric, all its eigenvalues are real. So, the eigenvalues of $A^T A + \lambda I$ are positive. Hence, this matrix is invertible.

6 Applications

6.1 Data fitting

We want to fit a line $y = ax + b$ to a given point cloud $(x_1, y_1), \dots, (x_n, y_n)$ of $\mathbb{R}^2$.

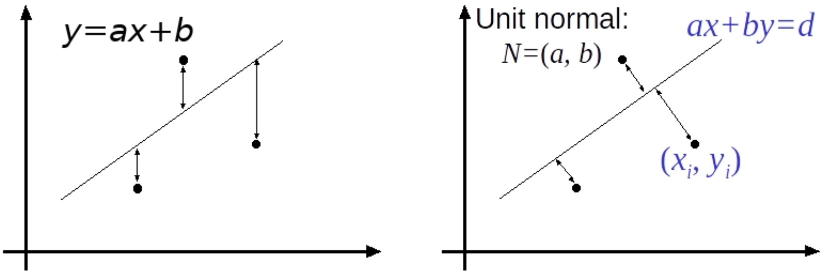


Fig. II.2: Least squares fitting: orthogonal (left) and normal (right).

So, we define a data term

$$E(\boldsymbol{c}) = \sum_{i=1}^{n} (y_i - ax_i - b)^2 = \|A\boldsymbol{c} - \mathrm{d}\|_2^2,$$

where

$$A = \begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} ; \mathbf{c} = \begin{bmatrix} a \\ b \end{bmatrix} \; ; \; \mathbf{d} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} .$$

This model is called *Orthogonal Least Squares.* The minimization of E leads to an optimal estimation of the fitting line. Nevertheless, such a model fails for vertical lines and is not rotation-invariant. To cope with that, we can slightly modify the distance between the points and the estimated line (See Figure II.2), where the line is represented by an equation of the form:

$$ax + by = d.$$

We define the associated data term

$$J(a, b, d) = \sum_{i=1}^{n} (ax_i + by_i - d)^2.$$

This model is called *Total least squares.*

We can eliminate d:

$$\frac{\partial J}{\partial d} = \sum_{i=1}^{n} -2(ax_i + by_i - d) = 0.$$

Put

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \; , \; \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i.$$

We get,

$$\begin{aligned}
E(a,b) &= \sum_{i=1}^{n}(a(x_i-\bar{x})+b(y_i-\bar{y}))^2 \\
&= \left\| \begin{bmatrix} x_1-\bar{x} \; y_1-\bar{y} \\ \vdots \qquad \vdots \\ x_n-\bar{x} \; y_n-\bar{y} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right\|_2^2 \\
&= (U\boldsymbol{c})^T(U\boldsymbol{c}) \\
&= \boldsymbol{c}^T(U^TU)\boldsymbol{c}
\end{aligned}$$

To ensure uniqueness of the solution, we formulate the problem:

$$\begin{array}{cl}
\underset{\boldsymbol{c}=(a,b)\in\mathbb{R}^2}{\text{minimize}} & E(a,b), \\
\text{subject to} & \|\boldsymbol{c}\|_2^2 = 1.
\end{array}$$

We can show that the solution of the latter problem is the eigenvector of U^TU associated with the smallest eigenvalue.

6.2 Signal denoising

Suppose that a noisy measurement of a signal $x \in \mathbb{R}^n$ is given:

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\epsilon}.$$

where $\boldsymbol{x}$ is an unknown signal, $\boldsymbol{\epsilon}$ is an unknown noise vector, and $\boldsymbol{y}$ is the known measurements vector. To denoise the signal $\boldsymbol{b}$ and so find a "good" estimate of $\boldsymbol{x}$, we consider a least squares problem, for wich we add a regularization term R:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{x}-\boldsymbol{y}\|^2 \; + \; \lambda \, R(\boldsymbol{x}), \tag{II.10}$$

where λ is a given regularization para- meter and $R(\boldsymbol{x})$ is sum of the squares of the differences of consecutive components of $\boldsymbol{x}$:

$$R(\boldsymbol{x}) = \sum_{i=1}^{n-1}(x_i - x_{i+1})^2.$$

This quadratic function can also be written as $R(\boldsymbol{x}) = \|\boldsymbol{L}\boldsymbol{x}\|^2$, where $\boldsymbol{L} \in \mathbb{R}^{(n-1)\times n}$ is given by

$$\boldsymbol{L} = \begin{pmatrix}
1 & -1 & 0 & 0 & \cdots & 0 & 0 \\
0 & 1 & -1 & 0 & \cdots & 0 & 0 \\
0 & 0 & 1 & -1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\
1 & -1 & 0 & 0 & \cdots & 1 & -1
\end{pmatrix}$$

The optimal solution of (II.10) is given by

$$\bar{\boldsymbol{x}} = (\boldsymbol{I} + +\lambda\boldsymbol{L}^T\boldsymbol{L})^{-1}\boldsymbol{y}.$$

6.3 Image denoising

Consider a noisy image $Y \in \mathbb{R}^{n\times n}$ and its associated vectorial represntation $\boldsymbol{y}$. As in the previous section, we express the denoising of $\boldsymbol{y}$ as the following least squares problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \|\boldsymbol{x}-\boldsymbol{y}\|^2 \ + \ \lambda \, R(\boldsymbol{x}), \tag{II.11}$$

where λ is a given regularization para- meter, $N = n^2$, and $R(\boldsymbol{x})$ is a regularization term

$$R(\boldsymbol{x}) = \sum_{j=1}^{n}\left(\sum_{i=1}^{n-1}(x_{i,j}-x_{i+1,j})^2 + (x_{n,j}-x_{i-1,j})^2\right).$$

The regularization term R corresponds to the $x-$derivative only. It can also be written as $R(\boldsymbol{x}) = \|\boldsymbol{L}\boldsymbol{x}\|^2$, where $\boldsymbol{L}\in\mathbb{R}^{N\times N}$ is given by

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{I} & 0 & \cdots & 0 \\ 0 & \boldsymbol{I} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{I} \end{bmatrix}, \quad \text{where} \quad \boldsymbol{I} = \begin{pmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \\ 0 & 0 & 0 & \cdots & 1 & -1 \end{pmatrix}$$

Again, the optimal solution of (II.13) is given by

$$\bar{\boldsymbol{x}} = (\boldsymbol{I} + \lambda\boldsymbol{L}^T\boldsymbol{L})^{-1}\boldsymbol{y}.$$

Départ. Maths-Info. 1^e Master Maths Traitement d'images

Série d'exercices 2

Mars 2020

Unconstrained and constrained optimization

Exercice 1

Find the maximum and minimum values of f in the following cases:

1. $f(x,y) = 81\,x^2 + y^2$ s.t. $4x^2 + y^2 = 9$.
2. $f(x,y) = 8x^2 - 2y$ s.t. $x^2 + y^2 = 1$.
3. $f(x,y,z) = y^2 - 10z$ s.t. $x^2 + y^2 + z^2 = 36$.
4. $f(x,y,z) = xyz$ s.t. $x + 9y^2 + z^2 = 4$. Assume that $x \geqslant$ for this problem. Why is this assumption needed?
5. $f(x,y,z) = 3x^2 + y$ s.t. $4x - 3y = 9$ and $x^2 + z^2 = 9$.
6. $f(x,y,z) = 4y - 2z$ s.t. $2x - y - z = 2$ and $x^2 + y^2 = 1$.

Exercice 2

Find the maximum and minimum values of $f(x,y) = 4x^2 + 10y^2$ on the disk $x^2 + y^2 \leq 4$.

Exercice 3

Given $(n_i)_{i=1,\dots,N} \in \mathbb{N}^*$ and $P > 0$, solve the following problem

$$\begin{aligned} \underset{(p_1,\dots,p_n)}{\text{maximize}} \quad & \sum_{i=1}^{N} \ln\left(1 + \frac{p_i}{n_i}\right) \\ \text{subject to} \quad & \sum_{i=1}^{N} p_i \leq P, \\ & \quad p_i > 0, \forall\, i = 1, ..N \end{aligned}$$

Exercice 4

Given A, b, and m, use the SVD algorithm to find a vector x with $\|x\|_2 < m$ minimizing $\|Ax - b\|_2$.

Exercice 5

We want to find a normal $n \times 1$ vector h satisfying $Ah = 0$, where A is $m \times n$ matrix, $m \geqslant n$, and $rank(A) = n$. We consider the following problem:

$$\begin{aligned} \underset{h \in \mathbb{R}^n}{\text{minimize}} \quad & \|Ah\| \\ \text{subject to} \quad & 1 - h^T h = 0, \end{aligned} \tag{II.12}$$

1. Using Lagrange multipliers, derive a characteristic equation.Derive h in terms of eigenvector of $(A^T A)$.

2. Let $A = USV^T$ the SVD decomposition.
   a. Show that $\|USV^T h\| = \|SV^T h\|$ and $\|Vh\| = \|h\|$.
   b. Express the minimization problem in terms of $y = V^T h$.
   c. Deduce that $y = [0, 0, ..., 1]^T$ and the corresponding vector h.

Exercice 6

Find the solution x to the least squares problem:

$$\underset{x}{\operatorname{argmin}} \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda\|\mathbf{b} - \mathbf{x}\|_2^2$$

Exercice 7 Problem

Find a circle that approximates given m points $\mathbf{a}_1, \cdots, \mathbf{a}_m$ of $\mathbb{R}^n$.

Exercice 8 Problem

Consider a noisy image $Y \in \mathbb{R}^{n\times n}$ and its associated vectorial represntation $\boldsymbol{y}$. As in the previous section, we express the denoising of $\boldsymbol{y}$ as the following least squares problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \; + \; \lambda \, R(\boldsymbol{x}), \tag{II.13}$$

where λ is a given regularization para- meter, $N = n^2$, and $R(\boldsymbol{x})$ is a regularization term.

1. Express R in terms of the Frobenius nor mof the gradient norm matrix.
2. Write the denoising problem as a least squres problem.
3. Grive the explicit solution of the minimization problem.

Chapter III
Convex Analysis

Skills to acquire

- **How to recognize convexity of sets, functions, and problems.**
- **How to characterize convexity of sets, functions, and problems.**

1 Convex sets

1.1 Basic notions

Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{R}^n$ and $C \subset \mathbb{R}^n$.

Definition 1.1 (Points combination). *A a linear combination of* $(\boldsymbol{x}_i)_i$ *is any sum*

$$\boldsymbol{x} = \theta_1 \boldsymbol{x}_1 + \ldots + \theta_k \boldsymbol{x}_k.$$

where $\theta_i \in \mathbb{R}$*. This combination is called convex if* $0 \leqslant \theta_i \leqslant 1, \forall\, i$ *and* $\sum\limits_{i=1}^{k} \theta_i = 1$ *; affine if* $\sum\limits_{i=1}^{k} \theta_i = 1$*; and conic if* $\theta_i \geqslant 0,\ \forall\, i$*.*

Definition 1.2. *The set* $\mathcal{C}$ *is called convex if it contains all of its convex combinations:*

$$\forall\, (x, y, t) \in \mathcal{C}^2 \times [0,1], \quad (1-t)x + ty \in \mathcal{C}.$$

Similarly, $\mathcal{C}$ *is called affine subspace if it contains all its affine combinations.*

Example 1.1 (Convex sets). *Let* $A \in \mathbb{R}^{m \times n}$*,* $\boldsymbol{a} \in \mathbb{R}^n$*,* $\mathbf{c} \in \mathbb{R}^m$*, and* $b \in \mathbb{R}$*. The following sets are convex.*

- *Euclidean balls (for any norm).*
- *Positive Orthant:* $\{\boldsymbol{x} \in \mathbb{R} : x_i \geqslant 0, \forall\, i\}$.
- *Hyperplane:* $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^T \boldsymbol{x} = b\}$*, Halfspace:* $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^T \boldsymbol{x} \leqslant b\}$.
- *Affine space:*$\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \mathbf{c}\}$.

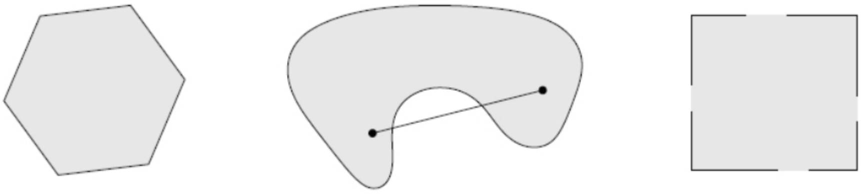


Fig. III.1: Examples of convex (first) and non-convex sets.

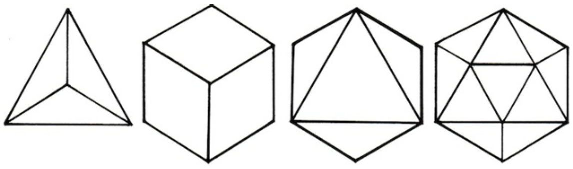


Fig. III.2: Polyhedra

- *Polyhedron:* $\{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \preceq \mathbf{c}\}$*, where inequality is component-wise.*

- *Simplex:* $conv\{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_k\}$*, where these points are affinely independent*[1]*. It is a special case of polyhedron and a generalization of the notion of a triangle to arbitrary dimensions.*

- *Ellipsoid: for a symmetric* $Q \succ 0$ *(positive definite)*

$$\left\{\boldsymbol{x} \in \mathbb{R}^n : (\boldsymbol{x} - \mathbf{c})^T Q(\boldsymbol{x} - \mathbf{c}) \leqslant r^2\right\}.$$

Definition 1.3. $\mathcal{C}$ *is called a cone if:*

$$x \in \mathcal{C} \implies tx \in \mathcal{C},\ \forall\, t \geqslant 0.$$

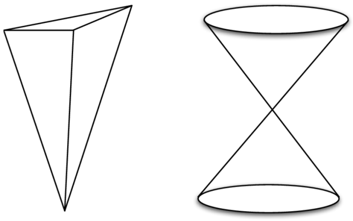


Fig. III.3: Examples of convex and non-convex cones

It is easy to show that $\mathcal{C}$ is a convex cone if:

$$x_1, x_2 \in \mathcal{C} \implies t_1x_1 + t_2x_2 \in \mathcal{C},\ \forall\, t_1, t_2 \geqslant 0.$$

Example 1.2 (Convex cones)**.** *The following sets are convex cones.*

- *Norm cone (for any norm* $\|\cdot\|$*):*

$$\{(\boldsymbol{x}, t) \in \mathbb{R}^n \times \mathbb{R}\ :\ \|\boldsymbol{x}\| \leqslant t\}.$$

For ℓ_2*-norm, it is called second-order cone.*

[1] $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_k$ are affinely independent means $\boldsymbol{x}_1 - \boldsymbol{x}_0, \ldots, \boldsymbol{x}_k - \boldsymbol{x}_0$ are linear independent.

- *Normal cone (given a closed convex $\mathcal{C}$ and point $\boldsymbol{x} \in \mathcal{C}$):*

$$\mathcal{N}_{\mathcal{C}}(\boldsymbol{x}) = \{\boldsymbol{g} \in \mathbb{R}^n | \boldsymbol{g}^T(\boldsymbol{z} - \boldsymbol{x}) \leqslant 0, \ \forall \, \boldsymbol{z} \in \mathcal{C}\}.$$

  By convention, we let $\mathcal{N}_C(\boldsymbol{x}) := \emptyset$ when $\boldsymbol{x} \notin \mathcal{C}$.

  1. $\mathcal{C} = [0,1]$*:* $\mathcal{N}_{\mathcal{C}}(x) = \begin{cases} \mathbb{R}_- & \textit{if}\, x = 0 \\ \mathbb{R}_+ & \textit{if}\, x = 1 \\ \{0\} & \textit{if}\, x \in]0,1[\\ \emptyset & \textit{otherwise} \end{cases}.$
  2. $\mathcal{C} = \{\boldsymbol{x} \in \mathbb{R}^n \, \big| \, \|\boldsymbol{x}\| \leqslant 1\}$*:* $\mathcal{N}_{\mathcal{C}}(\boldsymbol{x}) = \begin{cases} \mathbb{R}_+ \boldsymbol{x}, \ \textit{if} \ \|\boldsymbol{x}\| = 1 \\ \{0\}, \quad \textit{if} \ \|\boldsymbol{x}\| < 1 \\ \emptyset, \qquad \textit{otherwise} \end{cases}.$
  3. *The normal cone of a triangle at some points is depicted in Figure III.4.*

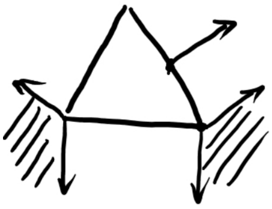


Fig. III.4: Some normal cones.

- *Positive semidefinite matrices*

The convexification of a nonconvex set X is achieved by defining the convex hull, which is the smallest convex set containing X. It is also the intersection of all convex sets containing X.

Definition 1.4. *Convex (resp. conic, affine) hull of S, denoted by conv(S) (resp. where* aff(S)*), is the set of all convex (resp. conic, affine) combinations of elements of S.*

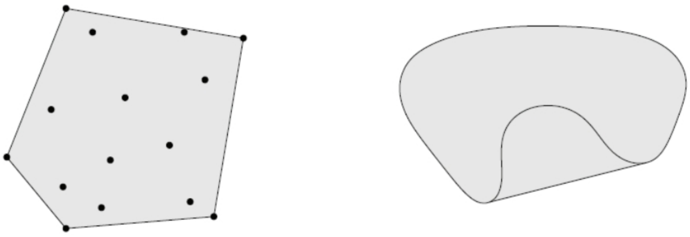


Fig. III.5: Examples of convex hulls

> **Remark:** A convex hull is always convex even when $\mathcal{C}$ is not convex.

We now consider some generic topological properties of convex sets.

Proposition 1.1. *The closure and the interior of a convex set are convex.*

The relative interior of a set S, denoted by $\operatorname{relint}(S)$, is a refinement of the concept of the interior, which is often more useful when dealing with low-dimensional sets placed in higher-dimensional spaces.

Definition 1.5. *The relative interior of a set S (denoted $\operatorname{relint}(S)$) is defined as its interior within the affine hull of S, i.e.,*

$$\operatorname{relint}(S) := \{x \in S : \exists \epsilon > 0, \mathbb{B}(x, \epsilon) \cap \operatorname{aff}(S) \subseteq S\}.$$

All metrics define the same $\operatorname{relint}(S)$.

Example 1.3. *Consider the closed unit square $S := \left\{(x, y, 0) \in \mathbb{R}^3 \,\middle|\, 0 \leq x, y \leq 1\right\}$. We have $\operatorname{int}(S) = \emptyset$ but $\operatorname{relint}(S) = \left\{(x, y, 0) \in \mathbb{R}^3 \,\middle|\, 0 < x, y < 1\right\}$.*

1.2 Key properties of convex sets

Theorem 1.1 (Separating hyperplane theorem)**.** *If C, D are nonempty convex sets with $C \cap D = \emptyset$, then there exists $\boldsymbol{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$ such that*

$$C \subset \{\boldsymbol{x} : \boldsymbol{a}^T\boldsymbol{x} \leqslant b\} \ \ \textit{and} \ \ D \subset \{\boldsymbol{x} : \boldsymbol{a}^T\boldsymbol{x} \geqslant b\}.$$

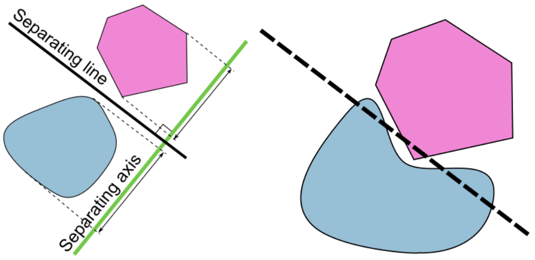


Fig. III.6: Separating hyperplane theorem

Theorem 1.2 (Supporting hyperplane theorem)**.**
If C is a nonempty convex set, and $\boldsymbol{x}_0 \in \partial(C)$, then there exists $\boldsymbol{a}$ such that

$$C \subset \{\boldsymbol{x} : \boldsymbol{a}^T\boldsymbol{x} \leqslant \boldsymbol{a}^T\boldsymbol{x}_0\}.$$

1.3 Operations preserving sets convexity

To prove the convexity of sets, one can use the definition. Nevertheless, it is usually easier and more effective to use some properties to prove convexity of sets that can be obtained from other sets, for which convexity is easier to establish.

Theorem 1.3 (Operations preserving sets convexity)**.** *The following operations preserve convexity.*

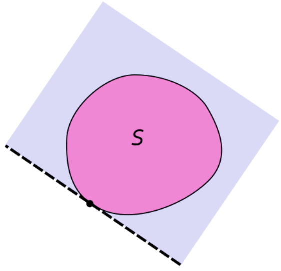


Fig. III.7: Supporting hyperplane

- *The intersection of convex sets is convex.*
- *The vector sum $C_1 + C_2$ of two convex sets C_1 and C_2 is convex.*
- *Scaling and translation: if C is convex, then $\lambda C + \boldsymbol{a}$ is convex for any a and λ. Furthermore, if C is a convex set and λ_1, λ_2 are positive scalars,*
$$(\lambda_1 + \lambda_2)C = \lambda_1 C + \lambda_2 C.$$
- *Affine images and preimages: Let $f : \mathbb{R}^n \to \mathbb{R}^m$ with $f(\boldsymbol{x}) = A\boldsymbol{x} + \boldsymbol{b}$, $A \in \mathbb{R}^{n\times m}$ and $\boldsymbol{b} \in \mathbb{R}^m$. If $\mathcal{C}$ is convex then $f(\mathcal{C})$ and $f^{-1}(\mathcal{C})$ are convex.*

Example 1.4. *content...*

2 Convex functions

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

2.1 Basic notions

Definition 2.6. *The function f is called* convex *if* $\mathrm{dom}(f)$ *convex and*

$$\forall \boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom}(f), \forall t \in [0,1],\ f((1-t)\boldsymbol{x} + t\boldsymbol{y}) \leqslant (1-t)f(\boldsymbol{x}) + tf(\boldsymbol{y}) \qquad \text{(III.1)}$$

f is strictly convex if f is convex and if equality only holds for $t \in]0,1[$. If $-f$ is convex, f is called concave.

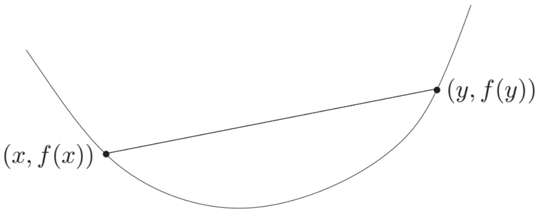


Fig. III.8: Convex function

Definition 2.7. *The function f is called strongly convex with parameter $m > 0$ if* $\mathrm{dom}(f)$ *is convex and*

$$\forall\,(\boldsymbol{x},\boldsymbol{y},t) \in \mathrm{dom}(f)^2\times[0,1]:\ f(t\boldsymbol{x}+(1-t)\boldsymbol{y}) \le tf(\boldsymbol{x})+(1-t)f(\boldsymbol{y})-\frac{\mu}{2}t(1-t)\|\boldsymbol{x}-\boldsymbol{y}\|^2.$$

f is quasi-convex if

$$\forall\,(\boldsymbol{x},\boldsymbol{y},t) \in \mathrm{dom}(f)^2 \times [0,1], \quad f((1-t)\boldsymbol{x}+t\boldsymbol{y}) \leqslant \max\{f(\boldsymbol{x}), f(\boldsymbol{y})\}$$

> **Remark:** It is often convenient to extend a convex function to all of $\mathbb{R}^n$ as the following:
> $$\tilde{f}(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}), \boldsymbol{x} \in \mathrm{dom}(f), \\ \infty, \ \ \boldsymbol{x} \notin \mathrm{dom}(f). \end{cases}$$

> **Remark:** Note that strongly convex $\Longrightarrow$ strictly convex $\Longrightarrow$ convex. For example, function $f(x) = \frac{1}{x}$ is strictly convex but not strongly convex.

It is often easy to draw a function defined on $\mathbb{R}$ to check geometrically whether it is convex or not. Here are some basic examples.

Example 2.1. *Functions on $\mathbb{R}$.*

- *Exponential functions e^{ax} are convex for any $a \in \mathbb{R}$.*
- *Even powers x^p (p is even) and powers of absolute vaue $|x|^p$ for $p \geqslant 1$ are convex.*
- *Power function x^a is convex for $a \geqslant 1$ or $a \leqslant 0$, and concave for $0 \leqslant a \leqslant 1$.*
- *Logarithmic function $\log x$ is concave over $\mathbb{R}_{++}$ and $x \log x$ is convex.*

Example 2.2. *For any norm $\|\cdot\|$, the following functions are convex:*
$f(\boldsymbol{x}) = \|\boldsymbol{x}\|$, $f(\boldsymbol{x}) = \|\boldsymbol{x}\|^2$, and the least squares loss $\|\boldsymbol{y} - A\boldsymbol{x}\|_2^2$ for any matrix A.

Example 2.3. *For a convex set C, the indicator function $\mathbb{1}_C(\boldsymbol{x})$ convex.*

2.2 Characterizations of convex functions

The convexity of functions defined on $\mathbb{R}$ is easy to check geometrically. The following theorem links between convexity on $\mathbb{R}$ and convexity on $\mathbb{R}^n$.

Theorem 2.4. *A function is convex iff its restriction to any line is convex.*

The next theorem relates function convexity with sets convexity.

Theorem 2.5. *A function f is convex if its epigraph is convex. Furthermore, if f is convex, then every level set is convex.*

> **Remark:** The converse is not true. For example, $f(x) = \sqrt{|x|}$ is not a convex function but each of its sublevel sets are convex sets.

When the function is differentiable, convexity can be characterized using its derivatives.

Theorem 2.6 (First-order characterization)**.** *Assume that f is differentiable and* $\mathrm{dom}(f)$ *is convex. Then, f is convex if and only if f completely lies above each of its tangent hyperplanes, i.e.,*

$$\forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom}(f): \quad f(\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}).$$

Furthermore, if f is convex then ∇f is a monotone mapping:

$$\left(\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\right)^T(\boldsymbol{x} - \boldsymbol{y}) \geq 0, \quad \forall\, \boldsymbol{x}, \boldsymbol{y} \in dom(f).$$

Theorem 2.7. *The following conditions are all equivalent to the condition that a differentiable function f is strongly-convex with constant $\mu > 0$.*

(i) $f(y) \geq f(x) + \nabla f(x)^T(y - x) + \frac{\mu}{2}\|y - x\|^2, \ \forall x, y.$

(ii) $g(x) = f(x) - \frac{\mu}{2}\|x\|^2$ *is convex,* $\forall x.$

(iii) $(\nabla f(x) - \nabla f(y))^T(x - y) \geq \mu\|x - y\|^2, \ \forall x, y.$

(iv) $f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y) - \frac{\alpha(1-\alpha)\mu}{2}\|x - y\|^2, \ \alpha \in [0, 1].$

Theorem 2.8 (Second-order characterization)**.** *If f is twice differentiable and* $\mathrm{dom}(f)$ *is convex then*

$$f \ \textit{ is convex} \quad \Longleftrightarrow \quad \forall\, x \in \mathrm{dom}(f): \ \nabla^2 f(x) \succeq 0.$$

Example 2.4. *Functions on $\mathbb{R}^n$*

- *Any affine function $f(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{x} + b$ is both convex and concave.*
- *Quadratic function $\frac{1}{2}\boldsymbol{x}^TQ\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x} + c$ is convex provided that $Q \succeq 0$.*
- *Affine function on $\mathbb{R}^{m\times n}$: $f(X) = \mathrm{tr}(A^TX) + b = \sum_{i,j} A_{i,j}X_{i,j} + b$, is convex.*

2.3 Operations preserving functions convexity

The following propositions gives useful techniques to establish function convexity from convexity of simpler functions.

Proposition 2.2. *Operations preserving functions convexity.*

- *Any conic combination of convex functions is convex.*
- *Affine composition: if f is convex, then $g(\boldsymbol{x}) = f(A\boldsymbol{x} + \boldsymbol{b})$ is convex.*

- *Point-wise maximization: if* $(f_i)_{i\in I}$ *are convex then* $f(\boldsymbol{x}) = \max_{i\in I} f_i(\boldsymbol{x})$ *is convex (I here can be infinite).*
- *Partial minimization: if* $g(\boldsymbol{x},\boldsymbol{y})$ *is convex in and* C *is convex, then* $f(\boldsymbol{x}) = \min_{\boldsymbol{y}\in C} g(\boldsymbol{x},\boldsymbol{y})$ *is convex.*

Example 2.5. *The following functions are convex.*

- $g(\boldsymbol{x}) = \max(\mathbf{a}_1^T\boldsymbol{x} + b_1, \cdots + \mathbf{a}_k^T x + b_k)$*, e.g.,* $f(\boldsymbol{x}) = \max\{x_1, \ldots, x_n\}$.
- $d(x, C)$*, where* C *is convex.*
- *For any set* C *(convex or not), the support function for is convex:* $\mathbb{1}_C^*(\boldsymbol{x}) = \max_{\boldsymbol{y}\in C} \boldsymbol{x}^T\boldsymbol{y}$.

Proposition 2.3. *Consider the composition* $f(\boldsymbol{x}) = h(g(\boldsymbol{x}))$*, where* $h : \mathbb{R} \to \mathbb{R}$ *and* $g : \mathbb{R}^n \to \mathbb{R}$*. If* h *is convex then* f *is convex if* $\Big($*h is nondecreasing and g is convex* $\Big)$ *or* $\Big($*h is nonincreasing and g is concave* $\Big)$

> **Remark:** Trick to remember the rule: $f''(x) = h''(g(x))\, g'(x)^2 + h'(g(x))\, g''(x)$.

Example 2.6. *The following functions are convex:* $\exp f(x)$ *with* f *convex and* $-\log f(x)$ *with* f *concave.*

3 Convex optimization

A convex optimization problem is of the form

$$\begin{array}{ll} \underset{\boldsymbol{x}\,\in\,\mathbb{R}^n}{\text{minimize}} & f(\boldsymbol{x}) \\ \text{subject to} & g_i(\boldsymbol{x}) \leqslant 0,\, i = 1, \ldots, m, \\ & h_j(\boldsymbol{x}) = 0,\, j = 1, \ldots, \ell \end{array} \tag{III.2}$$

- $f : \mathbb{R}^n \longrightarrow \overline{\mathbb{R}}$ and $g_i : \mathbb{R}^n \longrightarrow \mathbb{R}$ are all convex,
- h_j are affine; the equality constraints can be written as $A\boldsymbol{x} = \mathbf{b}$.

For an optimization problem, every local minimizer is a global one.

Proposition 3.4. *A local minimizer for a convex optimization is a global minimizer and the solution set* X_{opt} *is convex.*

Example 3.1 (Basis pursuit). *Given* $\boldsymbol{y} \in \mathbb{R}^n$ *and* $X \in \mathbb{R}^{n\times p}$*, where* $p > n$*, we seek the sparsest solution to the under-determined linear system* $X\beta = \boldsymbol{y}$*. A straightforward nonconvex formulation can be expressed as:*

$$\begin{array}{ll} \underset{\beta}{\text{minimize}} & \|\beta\|_0 \\ \text{subject to} & X\beta = \boldsymbol{y} \end{array} \tag{PB0}$$

where $\|\beta\|_0$ corresponds to the total number of nonzero elements in a vector. It is actually not a norm. So, we approximate $\|\cdot\|_0$ by a ℓ_1 norm to obtain the basis pursuit:

$$\begin{aligned} \underset{\beta}{\text{minimize}} \quad & \|\beta\|_1 \\ \text{subject to} \quad & X\beta = \boldsymbol{y} \end{aligned} \tag{PB}$$

Example 3.2 (**Lasso**)**.** *Given $\boldsymbol{y} \in \mathbb{R}^n$ and $X \in \mathbb{R}^{n\times p}$, a lasso problem can be formulated as follows.*

$$\begin{aligned} \underset{\beta\,\in\,\mathbb{R}^p}{\text{minimize}} \quad & \|\boldsymbol{y} - X\beta\|_2^2 \\ \text{subject to} \quad & \|\beta\|_1 \leq s \end{aligned}$$

Lasso consists in finding sparse approximate solution to the system $\boldsymbol{y} = X\beta$.

The optimization problem III.2 can be rewritten as

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \tilde{f}(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{x} \in \mathcal{C} \end{aligned} \tag{III.3}$$

where $\mathcal{C}$ represents the constraints set and $\tilde{f}(\boldsymbol{x}) = f(\boldsymbol{x}) + \mathbb{1}_{\mathcal{C}}(\boldsymbol{x})$, where $\mathbb{1}_{\mathcal{C}}(\boldsymbol{x}) = +\infty$ if $\boldsymbol{x} \notin \mathcal{C}$.

Theorem 3.9 (First order condition for optimality)**.** *Given a differentiable function f and convex set $\mathcal{C}$, consider the problem* (III.3)*. Then, a feasible point $\boldsymbol{x}$ is optimal if and only if:*

$$\nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) \geqslant 0, \quad \forall\, \boldsymbol{y} \in \mathcal{C}.$$

If $\mathcal{C} = \mathbb{R}^n$ (unconstrained optimization), then optimality condition reduces to familiar $\nabla f(\boldsymbol{x}) = 0$.

Example 3.3 (Equality-constrained minimization)**.** *Consider the equality-constrained problem:*

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & A\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

with $A \in \mathbb{R}^{n\times p}$ and f differentiable. Using the first-order optimality condition, solution $\boldsymbol{x}$ satisfies $A\boldsymbol{x} = \boldsymbol{b}$ (so as to be a feasible point) and

$$\nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) \geqslant 0, \quad \forall\, \boldsymbol{y} \textit{ such that } A\boldsymbol{y} = \boldsymbol{b}.$$

This is equivalent to

$$\nabla f(\boldsymbol{x})^T\boldsymbol{v} = 0, \quad \forall\ \boldsymbol{v}\ \in\ \mathrm{Ker}(A).$$

On the other hand, $\ker(A) = \mathrm{Im}(A^T)^{\perp}$. *Indeed,*

$$
\begin{aligned}
\boldsymbol{v} \ \in \mathrm{Ker}(A) &\iff A\boldsymbol{v} = 0 \\
&\iff \forall \boldsymbol{w}\, \langle A\boldsymbol{v}, \boldsymbol{w}\rangle = 0 \\
&\iff \forall \boldsymbol{w}\, \langle \boldsymbol{v}, A^T\boldsymbol{w}\rangle = 0 \\
&\iff \boldsymbol{v} \in \mathrm{Im}(A^T)^{\perp}.
\end{aligned}
$$

So, we have

$$\langle \boldsymbol{v}, \nabla f(\boldsymbol{x})\rangle = 0, \quad \forall\ \boldsymbol{v} \in \mathrm{Im}(A^T)^{\perp}.$$

This implies that $\nabla f(\boldsymbol{x}) \in (\mathrm{Im}(A^T)^{\perp})^{\perp} = \mathrm{Im}(A^T)$ *and hence,*

$$\exists \lambda \in \mathbb{R}^p : \nabla f(\boldsymbol{x}) = -\lambda^T A,$$

which are Lagrange multipliers.

Example 3.4 (Projection onto a convex set)**.** *Consider a convex set* $\mathcal{C}$ *and the optimization problem:*

$$
\begin{aligned}
\underset{\boldsymbol{x}}{\text{minimize}} \quad & \|\boldsymbol{a} - x\|_2^2 \\
\text{subject to} \quad & \boldsymbol{x} \in \mathcal{C}
\end{aligned}
$$

First-order optimality condition says that the solution $\boldsymbol{x}$ *satisfies*

$$\nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) = 2(\boldsymbol{x} - \boldsymbol{a})^T(\boldsymbol{y} - \boldsymbol{x}) \geqslant 0, \quad \textit{for all } \ \boldsymbol{y} \in \mathcal{C}$$

Equivalently, this says that

$$\boldsymbol{a} - \boldsymbol{x} \in \mathcal{N}_{\mathcal{C}}(\boldsymbol{x})$$

where $\mathcal{N}_{\mathcal{C}}$ *is the normal cone.*

Départ. Maths-Info. 1^e Master Maths Traitement d'images

Série d'exercices 3

Mars 2020

Convexity

Convex sets

Exercice 1 Hyperplans et demi-plans

- Quelle est la distance entre deux hyperplans paralleles $\{\boldsymbol{x}\in\mathbb{R}^n|\mathbf{a}^T\boldsymbol{x}=b_1\}$ et $\{\boldsymbol{x}\in\mathbb{R}^n|\mathbf{a}^T\boldsymbol{x}=b_2\}$?
- Montrer que l'ensemble des points qui sont plus proches de **a** que de **b** (au sens de la norme euclidienne) est un demi-plan (Description de Voronoi).

Exercice 2 polyèdres

Dites lesquels parmis les ensembles suivants sont des polyèdres. Le cas échéant, exprimer S sous la forme $S=\{\boldsymbol{x}|A\boldsymbol{x}\leqslant\mathbf{b},C\boldsymbol{x}=\mathbf{d}\}$.

- $S_1=\left\{\, y_1\mathbf{a}_1+y_2\mathbf{a}_2 \;\middle|\; -1\leqslant y_1,y_2\leqslant 1\right\}$, où $\mathbf{a}_1,\ \mathbf{a}_2\in\mathbb{R}^n$.
- $S_2=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \boldsymbol{x}\succeq 0, \mathbf{1}^Tx=1,\ \mathbf{a}^T\boldsymbol{x}=b_1,\ \sum_{i=1}^n x_ia_i^2=b_2\right\}$, où $\mathbf{a}\in\mathbb{R}^n$ et $b_1,b_2\in\mathbb{R}$.
- $S_3=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \boldsymbol{x}\succeq 0,\ \boldsymbol{x}^T\mathbf{y}\leqslant 1,\forall\,\mathbf{y}\in\mathbb{R}^n,\|\mathbf{y}\|_2=1\right\}$.
- $S_4=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \boldsymbol{x}\succeq 0,\ \boldsymbol{x}^T\mathbf{y}\leqslant 1,\forall\,\mathbf{y}\in\mathbb{R}^n,\|y\|_1=1\right\}$.

Exercice 3

Examiner la convexité des ensembles suivants.

- $S_1=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \alpha\leq\mathbf{a}^T\boldsymbol{x}\leq\beta\right\}$.
- $S_2=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \alpha_i\leqslant x_i\leqslant\beta_i, i=1,\dots,n\right\}$ (Rectangle).
- $S_3=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \mathbf{a}_1^T\boldsymbol{x}\leq b_1,\mathbf{a}_2^T\boldsymbol{x}\leq b_2\right\}$.
- $S_4=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \|\boldsymbol{x}-\boldsymbol{x}_0\|\leq\|\boldsymbol{x}-\mathbf{y}\|,\forall\,\mathbf{y}\in S\right\}$, avec $S\subset\mathbb{R}^n$.
- $S_5=\left\{\boldsymbol{x}\in\mathbb{R}^n \;\middle|\; \text{dist }(\boldsymbol{x},S)\leq\text{dist }(\boldsymbol{x},T)\right\}$, avec $S,T\subset\mathbb{R}^n$.

Exercice 4

Montrer que que si S_1 et S_2 sont deux ensembles convexes de $\mathbb{R}^{m\times n}$, alors il en est de même pour leurs sommes partielles :

$$S=\left\{(x,y_1+y_2)\;\middle|\; x\in\mathbb{R}^m,y_1,y_2\in\mathbb{R}^n,(x,y_1)\in S_1,(x,y_2)\in S_2\right\}.$$

Exercice 5

- On suppose que C et D sont deux parties differentes de $\mathbb{R}^n$. On considère l'ensemble $A = \{(\mathbf{a}, b) \in \mathbb{R}^{n+1},\ \mathbf{a}^T x \leq b,\ \forall \boldsymbol{x} \in C,\ \text{ et } \mathbf{a}^T x \geqslant b,\ \forall \boldsymbol{x} \in D\}$. Montrer que A est un cone convexe.
- Donner un exemple de deux ensembles convexes fermés disjoints qui ne peuvent pas etre séparés strictement.
- Exprimer l'ensemble convexe fermé $\{\boldsymbol{x} \in \mathbb{R}^2_+ | x_1 x_2 \geqslant 1\}$ comme l'intersection de demi-plans.

Exercice 6 Fonction support

La fonction support d'un ensemble $C \subset \mathbb{R}^n$ est définie par

$$S_C(\mathbf{y}) = \sup \left\{ \mathbf{y}^T \boldsymbol{x} \;\middle|\; x \in C \right\}.$$

($S_C(y)$ peut prendre $+\infty$). On suppose que C et D sont deux convexes fermés de $\mathbb{R}^n$. Montrer que

$$C = D \iff S_C = S_D.$$

Exercice 7

Soit K^* le cone dual d'un cone convexe K, i.e.,

$$K^* := \left\{ \mathbf{y} \in \mathbb{R}^n \;\middle|\; \forall \boldsymbol{x} \in K : \ \langle \mathbf{y}, \boldsymbol{x} \rangle \geqslant 0 \right\}.$$

Montrer les relations suivantes.

- K^* est un cone convexe.
- $K_1 \subset K_2 \implies K_2^* \subset K_1^*$.
- K^* est fermé.

Convex functions

Exercice 8 Examples of convex functions

Pour chacune des fonctions suivantes, dites si elle est convexe, concave, quasiconvexe[2], ou quasiconcave:

$$f_1(x) = e^x - 1 \text{ on } \mathbb{R}, \quad f_2(\boldsymbol{x}) = x_1 x_2 \text{ on } \mathbb{R}^2_{++}, \quad f_3(\boldsymbol{x}) = 1/(x_1 x_2) \text{ on } \mathbb{R}^2_{++},$$

$$f_4(\boldsymbol{x}) = x_1/x_2 \text{ on } \mathbb{R}^2_{++}, \quad f_5(\boldsymbol{x}) = x_1^2/x_2 \text{ on } \mathbb{R} \times \mathbb{R}_{++},$$

$$f_6(\boldsymbol{x}) = x_1^{\alpha} x_2^{1-\alpha} \text{ on } \mathbb{R}^2_{++}, 0 \leq \alpha \leq 1, \quad f_7(\boldsymbol{x}) = \left(\sum_{i=1}^{n} x_i^p \right)^{1/p}, p > 1, p \neq 0.$$

[2] l'image inverse de chaque ensemble de la forme $(-\infty, a)$ est convexe, or; $\forall\, x, y \in Set\lambda \in [0,1]$: $f(\lambda x + (1-\lambda) y) \leqslant \max \left\{ f(x), f(y) \right\}$.

Exercice 9 Norms and Dual Norms

Show that The negative log-determinant function

$$f(X) = -\log\left(det(X)\right)$$

is convex on $\mathbb{S}^n_{++}$.

Exercice 10 Inégalité de Jensen

Soit $f(x)$ une fonction convexe et $\lambda_1, \ldots, \lambda_n \in \mathbb{R}_+$ des poids vérifiants $\sum\limits_{j=1}^{k} w_j = 1$. Montrer que pour $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{R}^n$:

$$f(\lambda_1 \boldsymbol{x}_1 + \ldots + \lambda_k \boldsymbol{x}_k) \geq \lambda_1 f(\boldsymbol{x}_1) + \ldots + \lambda_k f(\boldsymbol{x}_k).$$

Exercice 11

Soit $f : \mathbb{R} \to \mathbb{R}$ est convexe, et $a, b \in \mathrm{dom}(()f)$, avec $a < b$.

- Montrer que: $\forall\, x \in [a, b] :\ f(x) \leq \frac{b-x}{b-a} f(a) + \frac{x-a}{b-a} f(b)$.
- Montrer que: $\frac{f(x)-f(a)}{x-a} \leq \frac{f(b)-f(a)}{b-a} \leq \frac{f(b)-f(x)}{b-x}$.
- Supposons que f est différentiable. Montrer que

$$f'(a) \leq \frac{f(b) - f(a)}{b - a} \leq f'(b)$$

- Supposons que f est deux fois différentiable. Montrer que $f''(a) \geq 0$ et $f''(b) \geqslant 0$.

Exercice 12

Quand est-ce que l'épigraphe d'une fonction f est un demi-plan ? cône convexe ? polyèdre ?

Exercice 13

On suppose que $f : \mathbb{R}^n \to \mathbb{R}$ est convexe avec $\mathrm{dom}(f) = \mathbb{R}^n$, et majorée sur $\mathbb{R}^n$. Montrer que f est constante.

Convex optimization

Exercice 14

Montrer que $\boldsymbol{x}^* = (1, 1/2, -1)$ est optimal pour le problème

$$\begin{array}{ll} \underset{\boldsymbol{x} \in \mathbb{R}^3}{\text{minimize}} & f(\boldsymbol{x}) = (1/2)\, \boldsymbol{x}^T P \boldsymbol{x} + \mathbf{q}^T \boldsymbol{x} + r \\ \text{subject to} & -1 \leq x_i \leq 1,\ i = 1, 2, 3 \end{array}$$

où

$$P = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, \quad q = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, \quad r = 1.$$

Chapter IV
First-order methods

Skills to acquire

- Compute subdifferentials and proximals operators
- Proximal gradient algorithms
- Applications to well known optimization problems.

We consider a general convex optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) \tag{IV.1}$$

This chapter presents first order algorithms, i.e., which make use of the function gradient. Such algorithms have low computational cost per iterations.

1 Gradient descent algorithms

1.1 Description

We suppose that f is differentiable with $\mathrm{dom}(f) = \mathbb{R}^n$. The gradient descent algorithm solves the problem (VI.1) using the core iteration k:

- **Choice of the step**: Choose $\tau^{(k)}$
- **Main iteration**: $\boxed{\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \tau^{(k)}\nabla f(\boldsymbol{x}^{(k)})}$

The basic algorithm have some variants which depend mainly on the choice of the step $\tau^{(k)}$. A simple *constant step*

$$\tau^{(k)} = \tau, \forall k,$$

can diverge if τ is too big and slow if τ is too small. On the other hand, an optimal *exact line search*

$$\tau^{(k)} = \min_{\tau} f(x^k - \tau \nabla f(x^l))$$

can be hard to compute, computationally demanding, or even does not exist.

Another popular approach is *Backtracking line search*, which consists in the following. At each iteration, start with $\tau^{(k)} = \tau_0$ (e.g., $\tau_0 = 1$). Then, take

$$\tau^{(k)} = \beta \tau^{(k)},$$

til it satifies the Armijo condition

$$f(x^{(k)} - t^{(k)} \nabla f(x^{(k)})) \leq \left(f(x^{(k)}) - \alpha t^{(k)} \|\nabla f(x)\|_2^2 \right),$$

where $0 < \beta < 1$ and $0 < \alpha \leqslant 1/2$ are some fixed parameters. One can simply use $\alpha = \beta = 0.5$. These simple τ updates tend to work well in practice.

1.2 Convergence

Gradient descent algorithms are inexpensive as they do not require second derivatives.

Theorem 1.1. *Suppose that f is convex and $\nabla f(x)$ is L-Lipschitz continuous*

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\|_2 \leqslant L \|\boldsymbol{x} - \boldsymbol{y}\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{y}.$$

If the optimal value is finite and attained at x^* then the gradient descent converges to x^*, with a rate $O(1/\epsilon)$[1] for constant step size and $O(\log(1/\epsilon))$ for backtracking line search.

For unconstrained problems, gradient descent is robust and less tuning. Thus, it is still widely used in practice. But, for a non-convex function f, the convergence is often slow and depends on the starting point and the scaling. In addition, it cannot handle nondifferentiable functions. There are many other better algorithms that deal with its shortcomings.

1.3 Accelerated gradient method

Heavy Ball Method In this technique, we add a momentum to gradient descent:

$$\boxed{x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}) + \beta_k (x^{(k)} - x^{(k-1)}),}$$

where momentum prevents oscillation.

[1] i.e., to get $f(x^{(k)}) - f(x^*) \leqslant \epsilon$, we need $O(1/\epsilon)$ iterations.

Nesterov Extrapolation Nesterov's exploration alternates between gradient updates and proper extrapolation

Choose $x^{(0)} = x^{(-1)} = y^0 \in \mathbb{R}^n, \tau \leqslant 1/L, t_0 = 0.$

$$\begin{cases} t_{k+1} &= \frac{1+\sqrt{1+4t_k^2}}{2}, \\ y &= x^{(k)} + \frac{k-1}{k+2}(x^{(k)} - x^{(k-1)}), \\ x^{(k+1)} &= y - \tau \nabla f(y). \end{cases}$$

Each iteration takes nearly same cost as GD but converge faster. Note that it is not a descent method.

1.4 Quasi-Newton methods

By successive measurements of the gradient, Quasi-Newton methods build a quadratic model of the objective function that can be used to estimate the inverse of the Hessian. The algorithm is the following.

Algorithm IV.1: Quasi-Newton algorithm

```
/* Initialization                                                  */
Choose initial point $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$ and a matrix $H_0 \succ 0$
/* Main loop                                                       */
for $k = 0, 1, 2, 3, \ldots, N$ do
    /* compute quasi-Newton direction                              */
    $\Delta x_k = -H_k^{-1} \nabla f x_k$
    /* determine step size (e.g., by backtracking line search)    */
    Compute $t_k$
    /* compute $x_{k+1}$                                           */
    $x_{k+1} = x_k + t_k + \Delta x_k$
    /* Update $H$                                                  */
    Compute $H_{k+1}$.
end
```

There are different update rules exist for H_{k+1}. One can also estimate H_k^{-1} or a factorization of H^k to simplify the computation of Δx_k. The BFGS method (Broyden-Fletcher-Goldfarb-Shanno) is the most popular quasi-Newton method.

BFGS update

$$H_{k+1} = H_{k+1} + \frac{yy^T}{y^T s} - \frac{H_k s s^T H_k}{s^T H_k s},$$

where

$$s = x_{k+1} - x_k, \quad y = \nabla f(x_{k+1}) - \nabla f(x_k).$$

Inverse update

$$H_{k+1}^{-1} = \left(I - \frac{sy^T}{y^T s}\right) H_k^{-1} \left(I - \frac{ys^T}{y^T s}\right) + \frac{ss^T}{y^T s}.$$

1.5 Nondifferentiable or constrained problems

For nondifferentiable or constrained problems, there are many methods that can be used, e.g., subgradient and proximal gradient methods.

2 Sub-gradient methods

Subgradients generalize gradients for optimizing convex functions that are not necessarily differentiable. In many situations, we deal with common operations that preserve convexity but not differentiability (e.g., the max-operation).

2.1 Sub-gradient

Recall that for differentiable f, that there is a single unique linear tangent $\nabla f(\boldsymbol{x})$ that under estimates the function (Figure IV.1):

$$f(\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y}-\boldsymbol{x}) \quad \forall \boldsymbol{x}, \boldsymbol{y}.$$

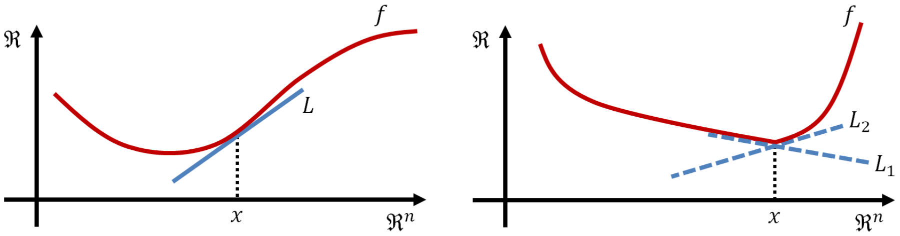


Fig. IV.1: Linear tangents under estimating a function.

Definition 2.1. *A **subgradient** of a function f at $\boldsymbol{x}$ is any $g \in \mathbb{R}^n$ such that*

$$f(\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y}-\boldsymbol{x}), \quad \forall \boldsymbol{y}$$

*The set of all subgradients of convex f is called the **subdifferential**:*

$$\partial f(\boldsymbol{x}) \overset{\text{def.}}{=} \left\{\boldsymbol{g} \in \mathbb{R}^n \ ; \ \forall \boldsymbol{y}, f(\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y}-\boldsymbol{x})\right\}.$$

> **Remark:** Informally, the “size” of $\partial f(x)$ controls how smooth f is at x. For nonconvex f, subgradients need not exist.

Example 2.1 (absolute value)**.** *For $f(x) = |x|$, it is easy to show that for $x \neq 0$, there is a unique subgradient $g = \text{sign}(x)$. For $x = 0$, $\partial f(x) = [-1, 1]$.*

Example 2.2 (Subdifferential of norms at 0)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(\boldsymbol{x}) = \|\boldsymbol{x}\|$. We can show that:*

$$\partial f(0) = \mathbb{B}_{\|.\|_*}(0,1) = \left\{ \boldsymbol{g} \in \mathbb{R}^n \;\middle|\; \|g\|_* \leq 1 \right\},$$

where $\|.\|_*$ denotes the dual norm. In particular, for $g(x) = \|\boldsymbol{x}\|_1$, $\partial g(0) = [-1,1]^n$.

Proposition 2.1 (Differentiability)**.** *f is differentiable at $x \iff \partial f(x) = \{\nabla f(x)\}$.*

Example 2.3. *Let $f(x) = |\cdot|, x \in \mathbb{R}$.*

$$\partial f(x) = \begin{cases} -1 & \mathit{if} \quad x < 0, \\ +1 & \mathit{if} \quad x > 0, \\ [-1,1] & \mathit{if} \quad x = 0. \end{cases}$$

Example 2.4 (Subdifferential of the ℓ_2-norm)**.** *Let $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_2$.*

$$\partial f(\boldsymbol{x}) = \begin{cases} \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} & \boldsymbol{x} \neq 0, \\ \mathbb{B}_2(0,1) & \boldsymbol{x} = 0. \end{cases}$$

Theorem 2.2. *A subdifferential $\partial f(x)$ is closed and convex, even for nonconvex f.*

Theorem 2.3 (Existence)**.** *Le f be convex with a nonempty* dom(f)*. Then,*

(a) $x \in \operatorname{relint}(\operatorname{dom}(f)) \implies \partial f(\boldsymbol{x}) \neq \emptyset$.
(b) $\partial f(\boldsymbol{x}) \neq \emptyset$ and bounded $\iff x \in \operatorname{int}(\operatorname{dom}(f))$.

Example 2.5.

$$f(x) = \begin{cases} -\sqrt{x}, x \geqslant 0 \\ +\infty, \;\; x < 0 \end{cases} ; \quad \partial f(x) = \begin{cases} -\frac{1}{2\sqrt{x}}, x > 0 \\ \emptyset, \qquad x = 0 \end{cases}$$

Note that 0 is not in the interior of dom(f)*.*

Theorem 2.4 (Directional Derivatives)**.** *Let $\boldsymbol{x} \in \operatorname{int}(\operatorname{dom}(f))$. Then, the directional derivative $f'(\boldsymbol{x};\boldsymbol{d})$ is finite for all $\boldsymbol{d} \in \mathbb{R}^n$. In particular, we have*

$$f'(\boldsymbol{x};\boldsymbol{d}) = \max_{s \in \partial f(x)} s^T \boldsymbol{d}$$

Example 2.6 (connection to convex geometry)**.** *For a non empty set $C \subseteq \mathbb{R}^n$, $\partial \mathbb{1}_C(\boldsymbol{x}) = \mathcal{N}_C(\boldsymbol{x})$. Recall that for $\boldsymbol{x} \notin C, \partial \mathbb{1}_C(\boldsymbol{x}) = \mathcal{N}_C(\boldsymbol{x}) = \emptyset$ by convention. As a special case,*

$$\partial \mathbb{1}_{\mathbb{B}(0,1)}(\boldsymbol{x}) = \mathcal{N}_{\mathbb{B}(0,1)}(\boldsymbol{x}) = \begin{cases} \left\{ \boldsymbol{y} \in \mathbb{R}^n \;\middle|\; \|\boldsymbol{y}\|_* \leq \boldsymbol{y}^T \boldsymbol{x} \right\}, & \|\boldsymbol{x}\| \leq 1 \\ \emptyset, & \|\boldsymbol{x}\| > 1. \end{cases}$$

2.2 Subdifferential calculus

In this section, we are present some practical rules for calculating subgradients. We can distinguish two types of rules giving weak (one subgradient only) or strong (all subgradients) results.

Rule 1: Conic combination

Theorem 2.5. *Let $f, g : \mathbb{R}^n \to (-\infty, \infty]$ be proper functions and let $\alpha > 0$.*

- *Multiplication by a Positive Scalar*

$$\forall\, \boldsymbol{x} \in \operatorname{dom}(f) : \quad \partial(\alpha f(\boldsymbol{x})) = \alpha\, \partial f(\boldsymbol{x}).$$

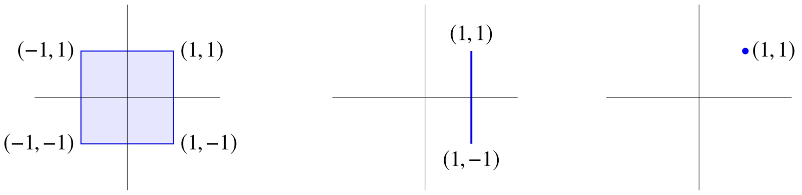
- *Summation*

$$\forall\, \boldsymbol{x} \in \operatorname{dom}(f) \cap \operatorname{dom}(g) : \quad \partial f(\boldsymbol{x}) + \partial g(\boldsymbol{x}) \subseteq \partial(f+g)(\boldsymbol{x}),$$
$$\forall\, \boldsymbol{x} \in \operatorname{int}(\operatorname{dom}(f) \cap \operatorname{dom}(g)) : \quad \partial(f+g)(\boldsymbol{x}) = \partial f(\boldsymbol{x}) + \partial g(\boldsymbol{x}).$$

The first summation rule is weak and the second is strong.

Example 2.7 (Subdifferentials of the ℓ_1-norm). *Consider $f : \mathbb{R}^n \to \mathbb{R}$ given by $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1 = \sum_{i=1}^n |x_i|$. Then, $f = \sum_{i=1}^n f_i$, where $f_i(\boldsymbol{x}) = |x_i|$. Hence,*

$$\partial f(\boldsymbol{x}) = \left\{ \boldsymbol{z} \in \mathbb{R}^n \;\middle|\; z_i = sign(x_i) \text{ if } x_i \neq 0,\ |z_j| \leq 1 \text{ if } x_j = 0 \right\}.$$



$\partial f(0,0) = [-1,1] \times [-1,1]$ $\quad$ $\partial f(1,0) = \{1\} \times [-1,1]$ $\quad$ $\partial f(1,1) = \{(1,1)\}$

Rule 2: Affine composition

Theorem 2.6. *Let $f : \mathbb{R}^n \to (-\infty, \infty]$ be a proper function and $h(\boldsymbol{x}) = f(A\boldsymbol{x} + \boldsymbol{b})$. Assume h is proper, and let $\alpha > 0$.*

- *Weak result*

$$\forall\, \boldsymbol{x} \in \operatorname{dom}(f) : \quad A^T(\partial f(A\boldsymbol{x} + \boldsymbol{b})) \subseteq \partial h(\boldsymbol{x}).$$

- *Strong result*

$$\forall\, \boldsymbol{x} \in \operatorname{int}(\operatorname{dom}(f)) \;/\; A(\boldsymbol{x}) + \boldsymbol{b} \in \operatorname{int}(\operatorname{dom}(f)) : \quad \partial h(\boldsymbol{x}) = A^T(\partial f(A\boldsymbol{x} + \boldsymbol{b})).$$

Example 2.8. *Compute the subdifferential of $f(\boldsymbol{x}) = \|A\boldsymbol{x} + \boldsymbol{b}\|_1$.*

$$\partial f(\boldsymbol{x}) = \sum_{i \in I_1} \operatorname{sign}(\mathbf{a}_i^T \boldsymbol{x} + b_i)\mathbf{a}_i + \sum_{i \in I_2} [-\mathbf{a}_i, \mathbf{a}_i],$$

where $\mathbf{a}_i$ is the i^{th} column of A, $I_1 = \{i : \mathbf{a}_i^T \boldsymbol{x} + b_i \neq 0\}$, $I_2 = \{i : \mathbf{a}_i^T \boldsymbol{x} + b_i = 0\}$. A weak result indicating one possible subgradient is $A^T \operatorname{sign}(A\boldsymbol{x} + \boldsymbol{b}) \in \partial f(\boldsymbol{x})$.

Example 2.9. *Compute the subdifferential of $\|A\boldsymbol{x} + \boldsymbol{b}\|_2$.*

$$\partial f(\boldsymbol{x}) = \begin{cases} \frac{A^T(A\boldsymbol{x}+\boldsymbol{b})}{\|A\boldsymbol{x}+\boldsymbol{b}\|_2}, & A\boldsymbol{x} + \boldsymbol{b} \neq 0, \\ A^T \mathbb{B}_{\|.\|_2}(0,1), & A\boldsymbol{x} + \boldsymbol{b} = 0. \end{cases}$$

If a weak result is required, then $0 \in \partial f(\boldsymbol{x})$ for any $\boldsymbol{x}$ satisfying $A\boldsymbol{x} + \boldsymbol{b} = 0$.

Rule 3: Chain rule

Theorem 2.7 (Sub-differential calculus). *. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function and $g : \mathbb{R} \to \mathbb{R}$ be a nondecreasing convex function. Let $\boldsymbol{x} \in \mathbb{R}^n$, and suppose that g is differentiable at the point $f(\boldsymbol{x})$. Let $h = g \circ f$. Then*

$$\partial h(\boldsymbol{x}) = g'(f(\boldsymbol{x}))\partial f(\boldsymbol{x}).$$

Example 2.10 (subdifferential of $\|.\|_1^2$). *Consider the function $h : \mathbb{R}^n \to \mathbb{R}$ given by $h(\boldsymbol{x}) = \|\boldsymbol{x}\|_1^2$, which can be written as the composition $h = g \circ f$, where $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$ and $g(t) = \max\{t, 0\}^2$. Both f and g are real-valued convex functions, and g is nondecreasing and differentiable over $\mathbb{R}$ with derivative $g'(t) = 2\max\{t, 0\}$. Therefore, by the chain rule of subdifferential calculus, for any $x \in \mathbb{R}^n$,*

$$\partial h(\boldsymbol{x}) = g'(f(\boldsymbol{x}))\partial f(\boldsymbol{x}) = 2\max\{0, \|x\|_1\}\partial f(x),$$

$$\partial h(x) = 2\|x\|_1 \left\{z \in \mathbb{R}^n : z_i = sign(x_i), i \in I_1, |z_j| \leq 1, j \in I_2\right\},$$

where $I_1 = \{i : x_i \neq 0\}$ and $I_2 = \{i : x_i = 0\}$.

Example 2.11 (Distance to a convex set). *Recall the distance function to a closed and convex set C:*

$$d_C(\boldsymbol{x}) = \min_{\boldsymbol{y} \in C} \ \|\boldsymbol{y} - \boldsymbol{x}\|_2 = \|\boldsymbol{x} - P_C(\boldsymbol{x})\|_2$$

where $P_C(\boldsymbol{x})$ is the projection of $\boldsymbol{x}$ onto C. We will show that

$$\partial d_C(\boldsymbol{x}) = \begin{cases} \left\{\frac{\boldsymbol{x} - P_C(\boldsymbol{x})}{\|\boldsymbol{x} - P_C(\boldsymbol{x})\|_2}\right\}, & \boldsymbol{x} \notin C, \\ \\ \mathcal{N}_C(\boldsymbol{x}) \cap \mathbb{B}(0,1), \, \boldsymbol{x} \in C. \end{cases}$$

We know that the function $\varphi_C(\boldsymbol{x}) = \frac{1}{2}d_C^2(\boldsymbol{x})$ is differentiable and

$$\forall\, \boldsymbol{x}, \quad \partial\varphi_C(\boldsymbol{x}) = \{\boldsymbol{x} - P_C(\boldsymbol{x})\}.$$

Note that $\varphi_C = g \circ d_C$, where $g(t) = \frac{1}{2}\max\{0, t\}^2$ is a nonincreasing real-valued convex differentiable function. Then by the chain rule:

$$\varphi_C(\boldsymbol{x}) = d_C(\boldsymbol{x}) \ \partial d_C(\boldsymbol{x}).$$

If $\boldsymbol{x} \notin C$, then $d_C(\boldsymbol{x}) = 0$, and thus $\partial d_C(\boldsymbol{x}) = \left\{\frac{\boldsymbol{x} - P_C(\boldsymbol{x})}{d_C(\boldsymbol{x})}\right\}$. If $\boldsymbol{x} \in C$, we show the double inclusion.

Rule 4: Maximization

Theorem 2.8 (max rule of subdifferential calculus). *Let $f_1, f_2, \ldots, f_m : \mathbb{R}^n \to (-\infty, \infty]$ be proper convex functions. Define $f(\boldsymbol{x}) = \max\{f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})\}$. Then,*

$$\forall\, \boldsymbol{x} \in \bigcap_{i=1,..,m} \int \mathrm{dom}(f_i) : \quad \partial f(\boldsymbol{x}) = conv\left(\bigcup_{i \in I(\boldsymbol{x})} \partial f_i(\boldsymbol{x})\right),$$

where $I(\boldsymbol{x}) = \left\{i \,\middle|\, f_i(\boldsymbol{x}) = f(\boldsymbol{x})\right\}$, the "active" functions at $\boldsymbol{x}$. As a weak result; to compute a subgradient at $\boldsymbol{x}$, choose any $k \in I(\boldsymbol{x})$, any subgradient of f_k at $\boldsymbol{x}$.

Example 2.12 (max of two functions)**.** *Let $f_1, f_2 : \mathbb{R}^N \to \mathbb{R}$ convex, differentiable, and $f(\boldsymbol{x}) = \max\{f_1(\boldsymbol{x}), f_2(\boldsymbol{x})\}$. Then,*

- *For $f_1(\boldsymbol{x}) > f_2(\boldsymbol{x})$, unique subgradient $g = \nabla f_1(\boldsymbol{x})$*
- *For $f_2(\boldsymbol{x}) > f_1(\boldsymbol{x})$, unique subgradient $g = \nabla f_2(\boldsymbol{x})$*
- *For $f_1(\boldsymbol{x}) = f_2(\boldsymbol{x})$, $\partial f(\boldsymbol{x}) =$ the line segment joining $\nabla f_1(\boldsymbol{x})$ and $\nabla f_2(\boldsymbol{x})$.*

Example 2.13 (subdifferential of the max function)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(\boldsymbol{x}) = \max\{x_1, x_2, \ldots, x_n\}$. Then we get:*

$$\partial f(\boldsymbol{x}) = \operatorname{conv}\left(\bigcup_{i\in I(\boldsymbol{x})} \partial f_i(\boldsymbol{x})\right) = \operatorname{conv}\left(\bigcup_{i\in I(\boldsymbol{x})} \{\boldsymbol{e}_i\}\right),$$

$$\partial f(\boldsymbol{x}) = \left\{\sum_{i\in I(\boldsymbol{x})} \lambda_i \boldsymbol{e}_i : \sum_{i\in I(\boldsymbol{x})} \lambda_i = 1, \lambda_{\geq} 0\right\}.$$

Example 2.14 (subdifferential of the ℓ_∞-norm)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_\infty$.*

$$\partial f(\boldsymbol{x}) = \begin{cases} \mathbb{B}_{\|.\|_1}(0,1), & \boldsymbol{x} = 0, \\ \left\{\sum\limits_{i\in I(\boldsymbol{x})} \lambda_i \operatorname{sign}(x_i)\boldsymbol{e}_i : \sum\limits_{i\in I(\boldsymbol{x})} \lambda_i = 1, \lambda_j \geq 0, j \in I(\boldsymbol{x})\right\}, & \boldsymbol{x} \neq 0. \end{cases}$$

Example 2.15. ***piecewise-linear function***

$$f(\boldsymbol{x}) = \max_{i=1,\ldots,m} \; a_i^T \boldsymbol{x} + b$$

the subdifferential at $\boldsymbol{x}$ is a polyhedron

$$\partial f(\boldsymbol{x}) = \operatorname{conv}\{a_i | i \in I(\boldsymbol{x})\}$$

Theorem 2.9 (weak maximum rule of subdifferential calculus)**.** *Let I be an arbitrary set, and suppose that any $i \in I$ is associated with a proper convex function $f_i : \mathbb{R}^n \to (-\infty, \infty]$. Let $f(\boldsymbol{x}) = \max\limits_{i\in I} f_i(\boldsymbol{x})$. Then for any $\boldsymbol{x} \in \operatorname{dom}(f)$:*

$$\operatorname{conv}\left(\bigcup_{i\in I(\boldsymbol{x})} \partial f_i(\boldsymbol{x})\right) \subseteq \partial f(\boldsymbol{x}),$$

where $I(\boldsymbol{x}) = \left\{i \in I \;\middle|\; f(\boldsymbol{x}) = f_i(\boldsymbol{x})\right\}$. Usually we get equality, but requires some technical conditions to hold.

Rule 5: Minimization

Theorem 2.10 (Sub-differential calculus)**.** *$f(x) = \inf_{y} \; h(x, y)$, h convex. To find a subgradient at $\hat{x}$ (weak result), find $\hat{y}$ that minimizes $h(\hat{x}, y)$ (assuming minimum is attained) and then find subgradient $(g, 0) \in \partial h(\hat{x}, \hat{y})$.*

Example 2.16. *Find a subgradient of*

$$f(x) = \inf_{y \in \mathcal{C}} \|x - y\|_2$$

where $\mathcal{C}$ is a closed convex set.

- *if $f(\hat{x}) = 0$ (that is, $\hat{x} \in \mathcal{C}$), take $g = 0$.*
- *if $f(\hat{x}) > 0$, find projection $\hat{y} = P(\hat{x})$ on $\mathcal{C}$ and take*

$$g = \frac{\hat{x} - \hat{y}}{\|\hat{x} - \hat{y}\|_2}$$

2.3 Optimality conditions

Subdifferential sets are extremely useful in characterizing minima points. The following theorem is simple but extremely important.

Theorem 2.11 (Fermat's optimality condition)**.** *Let $f : \mathbb{R}^n \to (-\infty, \infty]$ be a proper convex function. Then,*

$$\boldsymbol{x}^\star \in \operatorname*{argmin}_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}) \iff 0 \in \partial f(\boldsymbol{x}^\star).$$

Proof. One has

$$\boldsymbol{x}^\star \in \operatorname{argmin} f \quad \Leftrightarrow \quad \Big(\forall \boldsymbol{y}, f(\boldsymbol{x}^\star) \leqslant f(\boldsymbol{y}) + \boldsymbol{0}^T(\boldsymbol{x}^\star - \boldsymbol{y})\Big) \quad \Leftrightarrow \quad 0 \in \partial f(\boldsymbol{x}^\star).$$

□

Example 2.17 (medians)**.** *Suppose that we are given n different 20 and ordered numbers $a_1 < a_2 < \cdots < a_n$. Denote $A = \{a_1, a_2, \ldots, a_n\} \subset \mathbb{R}$. The set of possible medians is the optimal solution set of the problem*

$$\min_{x \in \mathbb{R}} f(x),$$

where $f(x) = \sum_{i=1}^{n} |x - a_i| = f_i(x)$.

$$\partial f(x) = \begin{cases} 2i - n, & x \in (a_i, a_{i+}), \\ 2i - 1 - n + [-1, 1], & x = a_i, \\ -n, & x < a_1, \\ n, & x > a_n. \end{cases}$$

$0 \in \partial f(a_i)$ if and only if $|2i - 1 - n| \leqslant 1$, which is equivalent to $\frac{n}{2} \leqslant i \leqslant \frac{n}{2} + 1$ and $0 \in \partial f(x)$ for some $x \in (a_i, a_{i+1})$ if and only if $i = \frac{n}{2}$. We can thus conclude that if n is odd, then the only optimal point is $\frac{a_{n+1}}{2}$ and when n is even, the optimal set is the interval $[a_{\frac{n}{2}}, a_{\frac{n}{2}+1}]$, establishing the fact that the optimal set is exactly the set of medians.

Theorem 2.12 (optimality conditions for composite problems)**.** *Let* $f, g : \mathbb{R}^n \to (-\infty, \infty]$ *be proper functions such that* $\mathrm{dom}(g) \subset \mathrm{int}(\mathrm{dom}(f))$*,* g *is convex. Consider the problem*

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) + g(\boldsymbol{x})$$

If $x^* \in dom(g)$ *is a local optimal solution for which* f *is differentiable, then*

$$-\nabla f(\boldsymbol{x}^*) \in \partial g(\boldsymbol{x}^*).$$

Such points are called ***stationary points****. This condition is sufficient when* f *is convex. In addition, the stationary points are global optimal solutions.*

Example 2.18 (Convex constraints)**.** *For* f *convex and differentiable and convex set* C*:*

$$x^* \in \operatorname*{argmin}_{x\in C}\ f(x) \iff \exists g \in \partial f(x^*)/ -g \in \mathcal{N}_C(x^*).$$

Example 2.19 (Lasso)**.** *Given* $y \in \mathbb{R}^n, X \in \mathbb{R}^{n\times p}$*:*

$$\operatorname*{argmin}_{\beta}\ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1,$$

where $\lambda \geqslant 0$*. Subgradient optimality:*

$$0 \in \partial\left(\frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1\right) \iff 0 \in -X^T(y - X\beta) + \lambda\partial\|\beta\|_1$$

$$0 \in \partial\left(\frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1\right) \iff \exists v \in \partial\|\beta\|_1 :\ X^T(y - X\beta) = \lambda v,$$

$$v_i = \begin{cases} \{1\} & if\, \beta_i > 0 \\ \{-1\} & if\, \beta_i < 0, \\ = [-1, 1] & if\, \beta_i = 0 \end{cases} \quad i = 1, ..., p$$

Write X_1*, ...,* X_p *for columns of* X*. Then our condition reads:*

$$\begin{cases} X_i^T(y - X\beta) \ \ = \lambda\ \mathrm{sign}(\beta_i) & if\, \beta_i \neq 0 \\ |X_i^T(y - X\beta)| \leqslant \lambda & if\, \beta_i = 0 \end{cases}$$

Note: subgradient optimality conditions don't lead to closed-form expression for a lasso solution. However, they do provide a way to check lasso optimality.

Example 2.20 (Soft-thresholding)**.** *Simplified lasso problem with* $X = I$*:*

$$\operatorname*{argmin}_{\beta}\ \frac{1}{2}\|y - \beta\|_2^2 + \lambda\|\beta\|_1,$$

$$\beta = S_\lambda(y); \quad [S_\lambda(y)]_i = \begin{cases} y_i - \lambda & if\, y_i > 0 \\ 0 & if\, -\lambda \leqslant y_i < \lambda, \\ y_i + \lambda & if\, y_i < -\lambda \end{cases} \quad i = 1, ..., n$$

Example 2.21. *Consider the problem*

$$\underset{x \in \mathbb{R}}{\text{minimize}} \quad f(x) + \lambda \|x\|_1 \tag{IV.2}$$

where $f : \mathbb{R}^n \to (-\infty, \infty]$ is an extended real-valued function. A point $x^* \in$ int(dom(f)) in which f is differentiable is a stationary point if $- \bigtriangledown f(x^*) \in \lambda \partial g(x^*)$, where $g(.) = \|.\|_1$. So, we obtain that x^* is a stationary point if

$$\frac{\partial f(x^*)}{\partial x_i} \begin{cases} = -\lambda & \textit{if } x_i^* > 0 \\ = \lambda & \textit{if } x_i^* < 0 \\ \in [-\lambda, \lambda] & \textit{if } x_i^* = 0 \end{cases}$$

This is a necessary condition for x^* to be a local minimum. If f is also convex, then it is a necessary and sufficient condition for x^* to be a global optimal solution.

2.4 Sub-gradient descent

When f is non differentiable, one can use a sub-gradient instead of the gradient to define a minimization.

Algorithm IV.2: Sub-gradient descent

```
/* Main loop                                                    */
for $k = 0, 1, 2, 3, \dots, N$ do
    /* Main iteration                                           */
    $\boxed{\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \tau^{(k)} g^{(\ell)} \quad \text{where} \quad g^{(\ell)} \in \partial f(\boldsymbol{x}^{(\ell)}).}$
    /* Stopping criterion                                       */
    if $\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k+1)}\|_2 < \epsilon$ then
        Stop
```

But, subgradients are not necessarily descent directions asshown on the example below. For instance, for $f(\boldsymbol{x}) = |x_1| + 3\,|x_2|$ at $\boldsymbol{x} = (1, 0)$, we have:

- $\mathbf{g}_1 = (1, 0) \in \partial f(x)$, and $-\mathbf{g}_1$ is a descent direction
- $\mathbf{g}_2 = (1, 3) \in \partial f(x)$, but $-\mathbf{g}_g$ is not a descent direction

So we keep track of best iterate $x_{best}^{(k)}$ among $x^{(0)}$, ..., $x^{(k)}$ so far, i.e., $f(x_{best}^{(k)}) = \min\limits_{i=0,\dots,k} f(x^{(i)})$.

Step size rules

- fixed step: τ_k constant
- fixed length: $\tau^{(k)} g^{(\ell)} = \|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k+1)}\|_2$ is constant
- diminishing: $\tau_k \to 0$ and $\sum\limits_{k=0}^{\infty} \tau_k = 0$.

Theorem 2.13. *If $\sum_\ell \tau_\ell = +\infty$ and $\sum_\ell \tau_\ell^2 < +\infty$, then $x^{(\ell)}$ converges to a minimizer of f.*

Analysis The subgradient method is not a descent method but the distance to the optimal goes to zero. As for the gradient method, the convergence is ensured when f is L-Lipschitz near the optimum. But, in general, this method performs poorly and convergence can be very slow. However, it handles general nondifferentiable convex problem, especially when ∂f is easy to compute, and often leads to very simple algorithms. Another option could be trying a splitting strategy as we will show later.

3 Proximal Algorithms

Proximal mapping are fundamental tools in convex optimization. Let's start with a proximal view of gradient descent.

3.1 Proximal Mapping

Consider the minimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}).$$

We perform quadratic approximation, replacing usual Hessian $\nabla^2 f(x)$ by $\frac{1}{t}I$,

$$f(y) \approx f(x) + \nabla f(x)^T(y-x) + \frac{1}{2t}\|y-x\|.$$

This can be seen as a linear approximation of f for which we add a proximity term to x, with weight $1/(2t)$. So, the update rule can be written as

$$\boldsymbol{x}^{(k+1)} = \operatorname*{argmin}_{\boldsymbol{x}} \left\{ f(\boldsymbol{x}^{(k)}) + \nabla f(\boldsymbol{x}^{(k)})^T(\boldsymbol{x}-\boldsymbol{x}^{(k)}) + \frac{1}{2t^{(k)}}\|\boldsymbol{x}-\boldsymbol{x}^{(k)}\|_2^2 \right\}.$$

Definition 3.2 (proximal mapping)**.** *Given a function $f:\mathbb{R}^n \to (-\infty,\infty]$, the proximal mapping of f is the operator given by*

$$\operatorname{prox}_f(\boldsymbol{x}) \overset{\text{def.}}{=} \operatorname*{argmin}_{\boldsymbol{z}} \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{z}\|_2^2 + f(\boldsymbol{z}). \tag{IV.3}$$

Example 3.1. $f(x) = \lambda\,|x|$.

$$\operatorname{prox}_f(x) = \operatorname{soft}(x,\lambda) \overset{\text{def.}}{=} \begin{cases} x-\lambda,\, x>\lambda \\ x+\lambda,\, x<-\lambda \\ 0, \qquad |x|\le\lambda. \end{cases} = \operatorname{sign}(x).\max\{|x|-\lambda, 0\},$$

i.e., $\operatorname{prox}_f = S_\lambda$ *the soft-thresholding operator.*

Example 3.2. *Consider the following functions from $\mathbb{R}$ to $\mathbb{R}$,$\lambda>0$ and $\mu\in\mathbb{R}$,*

$$g_1(x)=0;\; g_2(x)=\begin{cases}0, \quad x\neq 0\\ -\lambda,\, x=0\end{cases};\; g_3(x)=\begin{cases}0,\, x\neq 0\\ \lambda,\, x=0\end{cases};\; g_4(x)=\begin{cases}\mu x,\, x\geq 0,\\ \infty,\, x<0,\end{cases};$$

$$\operatorname{prox}_{g_1}(x) = \{x\}; \quad \operatorname{prox}_{g_2}(x) = \begin{cases} \{0\}, & |x| < \sqrt{2\lambda} \\ \{x\}, & |x| > \sqrt{2\lambda}, \\ \{0, x\}, & |x| = \sqrt{2\lambda}, \end{cases}; \quad \operatorname{prox}_{g_3}(x) = \begin{cases} \{x\}, & x \neq 0 \\ \emptyset, & x = 0 \end{cases}$$

$\operatorname{prox}_{g_4}(x) = 0.$

Example 3.3 (Convex Quadratic)**.** *Let* $f : \mathbb{R}^n \to \mathbb{R}$ *be given by* $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x} + c$*, where* $A \in \mathbb{S}^n_+$*,* $\boldsymbol{b} \in \mathbb{R}^n$*, and* $c \in \mathbb{R}$*. Then,*

$$\operatorname{prox}_f(\boldsymbol{x}) = (A + I)^{-1}(\boldsymbol{x} - \boldsymbol{b}).$$

Theorem 3.14. *If f is proper closed and convex, then* $\operatorname{prox}_f(x)$ *is a singleton. It can be evaluated efficiently for many widely used functions (in particular, regularizers).*

Example 3.4 (Projection Operator)**.**

$$g(\boldsymbol{x}) = \mathbb{1}_C; \quad \operatorname{prox}_g(\boldsymbol{x}) = \min_{\boldsymbol{z} \in C} \|\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

Example 3.5 (Multivariate Shrinkage)**.**

$$g(\boldsymbol{x}) = \lambda\|\boldsymbol{x}\|_2; \quad \operatorname{prox}_g(\boldsymbol{x}) = \begin{cases} (1 - \frac{\lambda}{\|u\|}, & \|u\| > \lambda, \\ 0, & \|u\| \leqslant \lambda. \end{cases}$$

Theorem 3.15. *Properties*

- ***Contraction:***

$$\|\operatorname{prox}_f(x) - \operatorname{prox}_f(y)\| \leqslant \|x - y\|.$$

Iteration of a general nonexpansive operator need not converge to a fixed point.

- ***Subgradient characterization:***

$$\boldsymbol{y} = \operatorname{prox}_f(\boldsymbol{x}) \iff \boldsymbol{x} - \boldsymbol{y} \in \partial f(\boldsymbol{y})$$

Thus we can write

$$\boldsymbol{y} = \operatorname{prox}_f(\boldsymbol{x}) \iff \boldsymbol{y} = \underbrace{(\mathrm{Id} + \partial f)^{-1}}_{\textit{Resolvent of operator of } \partial f}(\boldsymbol{x})$$

- ***Moreau decomposition***

$$\boldsymbol{x} = \operatorname{prox}_f(\boldsymbol{x}) + \operatorname{prox}_{f^*}(\boldsymbol{x})$$

where $f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}} \boldsymbol{y}^T\boldsymbol{x} - f(\boldsymbol{x})$ *is the convex conjugate of f. Tis is the main relationship between proximal operators and duality. It can be viewed as a generalization of orthogonal decomposition induced by a subspace:* $\boldsymbol{z} = \Pi_L(\boldsymbol{z}) + \Pi_{L^\perp}L(\boldsymbol{z})$*.*

3.2 Prox Calculus Rules

Rule 1: Separable functions

Theorem 3.16. *Suppose that $f : \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_m} \to (-\infty, \infty]$ is given by*

$$f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) = \sum_{i=1}^{m} f_i(\boldsymbol{x}_i), \quad x_i \in \mathbb{R}^{n_i}$$

Then for any $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_m}$,

$$\operatorname{prox}_f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) = \operatorname{prox}_{f_1}(\boldsymbol{x}_1) \times \cdots \times \operatorname{prox}_{f_m}(\boldsymbol{x}_m).$$

Example 3.6 (ℓ_1-norm)**.** *Suppose that $g : \mathbb{R}^n \to \mathbb{R}$ is given by $g(x) = \lambda \|\boldsymbol{x}\|_1$, where $\lambda > 0$. Then, $g(\boldsymbol{x}) = \sum_{i=1}^{n} \varphi(x_i)$, and* prox_g *is the soft thresholding applied for each component.*

Rule 2: Scaling and translation

Theorem 3.17. *Let $g : \mathbb{R}^n \to (-\infty, \infty]$ be a proper function, $\lambda \neq 0$, and $\mathbf{a} \in \mathbb{R}^n$. Define $f(\boldsymbol{x}) = g(\lambda \boldsymbol{x} + \mathbf{a})$ and $h(\boldsymbol{x}) = \lambda g(\boldsymbol{x}/\lambda)$. Then*

$$\operatorname{prox}_f(\boldsymbol{x}) = \frac{1}{\lambda}(\operatorname{prox}_{\lambda^2 g}(\lambda \boldsymbol{x} + \mathbf{a}) - \mathbf{a}),$$

$$\operatorname{prox}_h(\boldsymbol{x}) = \lambda(\operatorname{prox}_{g/\lambda}(\boldsymbol{x}/\lambda).$$

Example 3.7. *Consider the function $f : \mathbb{R}^n \to (-\infty, \infty]$ given for any*

$$f(x) = \begin{cases} \mu x, 0 \leq x \leq \alpha, \\ \\ \infty \quad \textit{else}, \end{cases}$$

where $\mu \in \mathbb{R}$ and $\alpha \geq 0$.

$$\operatorname{prox}_f(x) = \min\{\max\{x - \mu, 0\}, \alpha\}.$$

Rule 3: Norm composition

Theorem 3.18. *Let $f : \mathbb{R}^m \to (-\infty, \infty]$ be given by $f(\boldsymbol{x}) = g(\|\boldsymbol{x}\|)$ where $g : \mathbb{R}^m \to (-\infty, \infty]$ is a proper closed convex function satisfying* $\operatorname{dom}(g) \subseteq [0, \infty)$*. Then*

$$\operatorname{prox}_f(\boldsymbol{x}) = \begin{cases} \operatorname{prox}_g(\|\boldsymbol{x}\|)\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}, & \boldsymbol{x} \neq 0, \\ \\ \{u \in \mathbb{R}^n : \|u\| = \operatorname{prox}_g(0)\}, & \boldsymbol{x} = 0. \end{cases}$$

Example 3.8 (prox of Euclidean norm)**.** *Let $f(\boldsymbol{x}) = \lambda \|x\|$.*

$$\operatorname{prox}_f(\boldsymbol{x}) = \left(1 - \frac{\lambda}{\max\{\|x\|, \lambda\}}\, \boldsymbol{x}\right)$$

Rule 4: Quadratic addition

Theorem 3.19 (Quadratic addition)**.** *If $f(x) = g(x) + \frac{\rho}{2}\|x - a\|_2^2$, then*

$$\operatorname{prox}_f(x) = \operatorname{prox}_{\frac{1}{1+\rho}g}\left(\frac{1}{1+\rho}\boldsymbol{x} + \frac{\rho}{1+\rho}\mathbf{a}\right)$$

3.3 Algorithms

3.3.1 Proximal Point Algorithm

Consider a closed proper convex function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$. One has the following equivalence

$$\begin{aligned} x^\star \in \operatorname{argmin} f \quad &\Leftrightarrow \quad 0 \in \partial f(x^\star) \quad \Leftrightarrow \quad x^\star \in (\mathrm{Id} + \tau\partial f)(x^\star) \\ &\Leftrightarrow \quad x^\star = (\mathrm{Id} + \tau\partial f)^{-1}(x^\star) = \operatorname{Prox}_{\tau f}(x^\star). \end{aligned}$$

The *proximal iteration* or *proximal point algorithm* is

$$\boxed{x^{(k+1)} = \operatorname{Prox}_{\tau_k f}(x^{(k)}).}$$

On contrast to the gradient descent fixed point scheme, the proximal point method converges to the set of minimizers of f for any sequence of steps $\tau_k > 0$ such that $\sum\limits_{k=1}^{\infty} \tau_k = \infty$.

This basic proximal method has not found many applications because each iteration requires us to minimize the function f plus a quadratic. Nevertheless, the proximal algorithm would be useful in a situation where it is easier to minimize f plus a quadratic than minimizing f.

Example 3.9. *Consider the problem of minimizing the quadratic function*

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{b}\boldsymbol{x},$$

where $A \in \mathcal{S}_+^n$. This problem is, of course, equivalent to solving the system of linear equations $A\boldsymbol{x} = \boldsymbol{b}$, and when A is nonsingular, the unique solution is $\boldsymbol{x} = A^{-1}\boldsymbol{b}$. The proximal minimization method is then

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + (A + \epsilon I)^{-1}(\boldsymbol{b} - A\boldsymbol{x}^k),$$

where $\epsilon = \frac{1}{\tau}$ (we can choose $\tau_s = \tau$). This algorithm is a standard algorithm, called iterative refinement for solving $A\boldsymbol{x} = \boldsymbol{b}$ using only the regularized inverse $(A + \epsilon I)^{-1}$.

3.3.2 Proximal gradient descent (Forward-Backward)

It is not always possible to compute the proximal operator of the objective function. We will consider here a class of problems by imposing some structure on the function to be minimized. We consider functions F of the form

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \; F(\mathbf{x}) := f(\mathbf{x}) + g(\mathbf{x}),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R} \to (-\infty, +\infty]$ are closed proper convex and f is differentiable.

$$\begin{aligned} x^\star \in \operatorname{argmin} f + g \quad &\Leftrightarrow \quad 0 \in \nabla f(x^\star) + \partial g(x^\star) \\ &\Leftrightarrow \quad x^\star - \tau \nabla f(x^\star) \in (\mathrm{Id} + \tau \partial g)(x^\star) \\ &\Leftrightarrow \quad x^\star = (\mathrm{Id} + \tau\partial g)^{-1} \circ (\mathrm{Id} - \tau \nabla f)(x^\star). \end{aligned}$$

This fixed point suggests the following *Forward-Backward algorithm.*

$$\boxed{\mathbf{x}^{(k+1)} = \mathrm{prox}_{t_k g}(\mathbf{x}^{(k)} - t_k \nabla f(\mathbf{x}^{(k)})).}$$

When ∇f is Lipschitz continuous with constant L, this method can be shown to converge with rate $O(1/k)$ when a fixed step size $\lambda_k = \lambda \in (0, 1/L]$ is used. If L is not known, the step sizes can be found by a line search.

Example 3.10 (Lasso)**.** *Given $y \in \mathbb{R}^n, X \in \mathbb{R}^{n\times p}$:*

$$\min_\beta \; \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{f(\beta)} + \underbrace{\lambda\|\beta\|_1}_{g(\beta)},$$

$$\mathrm{prox}_{tg}(x) = \underset{z}{\operatorname{argmin}} \; \frac{1}{2t}\|\beta - z\|_2^2 + \lambda\|z\|_1 = S_{\lambda t}(\beta)$$

Hence proximal gradient update is:

$$\beta^+ = S_{\lambda t}\left(\beta + tX^T(y - X\beta)\right) \tag{IV.4}$$

Often called the ***iterative soft-thresholding algorithm*** *(**ISTA**).*

3.3.3 Accelerated proximal gradient descent

As for the gradient descent, we can apply a Nesterov acceleration to define the *Accelerated proximal gradient method*

$$\boxed{\begin{aligned} \boldsymbol{y} \quad &= \boldsymbol{x}^{(k)} + \tfrac{k-1}{k+2}\left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right), \\ x^{(k+1)} &= \mathrm{Prox}_{t_k g}\left(\boldsymbol{y} - t_k \nabla f(\boldsymbol{y})\right). \end{aligned}}$$

Example 3.11 (Lasso)**.** *Given $y \in \mathbb{R}^n, X \in \mathbb{R}^{n\times p}$:*

$$\min_\beta \; \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{f(\beta)} + \underbrace{\lambda\|\beta\|_1}_{g(\beta)},$$

Applying acceleration on the ISTA algorithm gives us ***FISTA*** *(F is for Fast):*

$$v = \beta^{(k-1)} + \frac{k-2}{k+1}\left(\beta^{(k-1)} - \beta^{(k-2)}\right)$$

$$\beta^{(k)} = \text{prox}_{\lambda t_k}\left(v + t_k X^T(y - Xv)\right), \quad k = 1, 2, 3, \ldots$$

or

$$x^{(k+1)} = \text{prox}_{\eta_k h}(y^k - \eta_k \nabla f(y^k))$$
$$y^{(k+1)} = x^{(k+1)} + \frac{\theta_k - 1}{\theta_{k+1}}(x^{(k+1)} - x^{(k)})$$
$$y^0 = x^0, \theta_0 = 1, \theta_{k+1} = \frac{1 + \sqrt{1 + 4\theta_k^2}}{2}$$

Example 3.12 (Projected gradient descent)**.** *Given closed, convex set $C \in \mathbb{R}$,*

$$\min_{x \in C} g(x) \iff \min_x g(x) + \mathbb{1}_C(x) \tag{IV.5}$$

$$x^+ = P_C\left(x - t\nabla g(x)\right) \tag{IV.6}$$

3.3.4 Douglas–Rachford splitting algorithm

We consider here the structured minimization problem

$$\underset{\boldsymbol{x} \in \mathbb{R}^n}{\text{minimize}} \quad F(\mathbf{x}) := f(\mathbf{x}) + g(\mathbf{x}), \tag{IV.7}$$

where $f, g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are closed proper convex functions. On contrary to the Forward-Backward, no smoothness is imposed on f. We here suppose that we can compute easily the proximal map of f and g.

The Douglas–Rachford iteration can also be written as

$$\boxed{y_{k+1} = F(y_k) = y_k + \text{prox}_g(2\,\text{prox}_f(y_k) - y_k) - \text{prox}_f(y_k).}$$

The operator $R_f(x) = 2\,\text{prox}_f(x) - x$ is called the Reflection operator.

The advantage of ADMM is that the objective terms (which can both include constraints, since they can take on infinite values) are handled completely separately, and indeed, the functions are accessed only through their proximal operators. ADMM is most useful when the proximal operators of f and g can be efficiently evaluated but the proximal operator for $f + g$ is not easy to evaluate.

Equivalent forms

- Introduce x_{k+1}

$$\begin{aligned} x_{k+1} &= \text{prox}_f(y_k), \\ y_{k+1} &= y_k + \text{prox}_g(2x_{k+1} - y_k) - x_{k+1} \end{aligned}$$

- Start iteration at y-update (introduce x_k)

$$\begin{aligned} y_{k+1} &= y_k + \text{prox}_g(2x_k - y_k) - x_k, \\ x_{k+1} &= \text{prox}_f(y_{k+1}). \end{aligned}$$

- switch y- and x-updates (introduce u_k)

$$\begin{aligned} u_{k+1} &= \operatorname{prox}_g(2x_k - y_k), \\ x_{k+1} &= \operatorname{prox}_f(y_k + u_{k+1} - x_k), \\ y_{k+1} &= y_k + u_{k+1} - x_k. \end{aligned}$$

- Make change of variables $w_k = x_k - y_k$

$$\begin{aligned} u_{k+1} &= \operatorname{prox}_g(x_k + w_k), \\ x_{k+1} &= \operatorname{prox}_f(u_{k+1} - w_k), \\ w_{k+1} &= w_k + x_{k+1} - u_{k+1}. \end{aligned}$$

This form is also called the alternating direction method of multipliers (ADMM).

Remarque 3.1. *In DF algorithm, it is possible to inter-change the roles of f and g, which defines another set of iterations.*

3.3.5 Alternating Direction Methods of Multipliers (ADMM)

We consider the following optimization problem.

$$\begin{aligned} &\text{minimize} \quad f(\boldsymbol{x}) + g(\boldsymbol{z}) \\ &\text{subject to} \quad A\boldsymbol{x} + B\boldsymbol{z} = \boldsymbol{c} \end{aligned}$$

Let $\rho > 0$, the augmented Lagrangian for this optimization is given as follows.

$$\mathcal{L}_\rho(x, z, \lambda) = f(x) + g(z) + \lambda^T(Ax + Bz - c) + \|Ax + Bz - c\|_2^2.$$

ADMM consists of the iterations.

$$\begin{aligned} z_{k+1} &= \underset{z}{\operatorname{argmin}} \left\{ g(z) + \tfrac{\rho}{2}\|Ax_k + Bz - c + u_k\|_2^2 \right\}, \\ x_{k+1} &= \underset{x}{\operatorname{argmin}} \left\{ f(x) + \tfrac{\rho}{2}\|Ax + Bz_{k+1} - c + u_k\|_2^2 \right\}, \\ u_{k+1} &= u_k + \rho(Ax_{k+1} + Bz_{k+1} - c). \end{aligned}$$

A general convex optimization problem such as $\min\limits_{x} f(x) + g(x)$ can be converted as the form that ADMM can be applied to:

$$\begin{aligned} &\underset{x,\, z}{\text{minimize}} \quad f(x) + g(z) \\ &\text{subject to} \quad x - z = 0 \end{aligned}$$

Example 3.13. *Consider ADMM for generic problem*

$$\min_{\boldsymbol{x} \in \mathcal{C}} f(\boldsymbol{x})$$

which can be written as

$$\min_{\boldsymbol{x} \in \mathcal{C}} f(\boldsymbol{x}) + \mathbb{1}_{\mathcal{C}}(\boldsymbol{x})$$

The ADMM algorithm gives

$$\begin{aligned} \boldsymbol{u}^{(k+1)} &= \Pi_{\mathcal{C}}(x^{(k)} + w^{(k)}), \\ \boldsymbol{x}^{(k+1)} &= \operatorname{argmin}(f(x) + \tfrac{\lambda}{2}\|u^{(k+1)} - w^{(k)}\|_2^2), \\ w^{(k+1)} &= w^{(k)} + (x^{(k+1)} - u^{(k+1)}). \end{aligned}$$

Série d'exercices 4

2020

First order methods

Exercice 1 Subgradients

Compute the subgradients of the following functions.

$$f(x) = \|x\|; \quad g(x) = \begin{cases} 0 & \text{if } x \in C \\ \infty & \text{if } x \notin C \end{cases}$$

where C is a nonempty convex subset of $\mathbb{R}^n$.

Exercice 2

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function. Show that a vector $d \in \mathbb{R}^n$ is a subgradient of f at x if and only if the function $d'y - f(y)$ attains its maximum at $y = x$.

Exercice 3

Let $C \subset \mathbb{R}^n$ be nonempty closed and convex set. Consider the function $\varphi_C : \mathbb{R}^n \to \mathbb{R}$ given by $\varphi_C(x) = \frac{1}{2}d_C^2(x) = \frac{1}{2}\|x - P_C(x)\|_2$, where P_C is the orthogonal projection mapping.

- Show that: $\bigtriangledown\varphi_C(x) = x - P_C(x)$.
- Compute $\partial d_C(x)$

Exercice 4 Proximal operators

Compute the prox of the following functions.

- Negative sum of Logs ($\lambda > 0$):

$$f(\boldsymbol{x}) = \begin{cases} -\lambda \sum_{j=1}^{n} \log x_j, \ \boldsymbol{x} > 0 \\ \infty \qquad\qquad\qquad \text{else,} \end{cases}$$

- ℓ_0-norm: $g(x) = \lambda\|\boldsymbol{x}\|_0$ where $\lambda > 0$.
- Norms: $h_1(\boldsymbol{x}) = \frac{\tau}{2}\|\boldsymbol{x}\|^2$, $h_2(\boldsymbol{x}) = \tau\|\boldsymbol{x}\|_1$.
- Loss function: $h_4(\boldsymbol{x}) = \frac{\tau}{2}\|A \cdot -y\|^2$, for $A \in \mathbb{R}^{p\times n}$.

Exercice 5

Prove the four proximal calculus rules.

Exercice 6

Apply the ADMM algorithm to Lasso problem.

Chapter V
Optimality and duality

Skills to acquire

- Laagrange Duality
- Conjugate function and duality
- Algorithms on dual problems

We consider a general convex optimization problem

$$\min_{x\in\mathbb{R}^n} f(x) \tag{V.1}$$

Duality is associated to a particular formulation of the optimization problem, so that for instance making change of variables results in a different duality.

1 Lagrange Duality

We consider the convex optimization problem:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & A\,\boldsymbol{x} = y, \\ & g(\boldsymbol{x}) \leq 0. \end{aligned}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $A \in \mathbb{R}^{p\times n}$, and $g : \mathbb{R}^n \to \mathbb{R}^q$. This problem will be referred to as the *primal problem*. We write the associated Lagrangian.

$$\mathcal{L}(\boldsymbol{x}, \lambda, \beta) \;=\; f(\boldsymbol{x}) + \lambda^T(A\boldsymbol{x} - \boldsymbol{y}) + \mu^T(g(\boldsymbol{x}))$$

The dual objective function $q : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R} \cup \{-\infty\}$ is defined to be

$$q(\lambda, \mu) \overset{\text{def.}}{=} \min_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \lambda, \mu). \tag{V.2}$$

The dual problem is given by

$$q^\star = \max_{(\lambda,\mu)\in\text{dom}(q)} q(\lambda, \mu). \tag{V.3}$$

Theorem 1.1 (convexity of the dual problem)**.** dom(q) *is a convex set and q is a concave function over* dom(q)*.*

The following proposition is the so-called weak duality, which assert that values of the dual problems always lower bounds values of the primal one

Theorem 1.2 (Weak duality)**.** *Consider the primal problem its dual problem. Then*

$$q^* \leq f^*,$$

where q^*, f^* are the optimal dual and primal values respectively.

Example 1.1.

$$\begin{aligned} \underset{x_1,\, x_2}{\text{minimize}} \quad & x_1^2 - 3x_2^2 \\ \text{subject to} \quad & x_1 = x_2^3 \end{aligned} \tag{V.4}$$

It is not difficult to show that the optimal solutions of the problem are $(1, 1)$, $(-1, -1)$ *with an optimal value of $f^* = -2$. The Lagrangian function is*

$$\mathcal{L}(x_1, x_2, \mu) = x_1^2 + \mu x_1 - 3x_2^2 - \mu x_2^3.$$

Obviously, for any $\mu \in \mathbb{R}$,

$$\min_{x_1, x_2} \mathcal{L}(x1, x_2, \mu) = -\infty,$$

and hence the dual optimal value is $q^* = -\infty$, which is an extremely poor lower bound on the primal optimal value $f^* = -2$.

The following fundamental theorem gives a sufficient condition (so-called qualification of the constraints) such that one actually has equality.

Theorem 1.3 (Strong duality)**.** *If the primal problem is convex and $\exists x_0 \in \mathbb{R}^N, \quad Ax_0 = y \quad \text{and} \quad g(x_0) < 0$, then $q^\star = f^\star$ (strong duality holds). Furthermore, $x^\star$ and $(u^\star, v^\star)$ are solutions of the dual problem verifying strong duality iff and only if*

$$Ax^\star = y, \quad g(x^\star) \leqslant 0, \quad u^\star \geqslant 0 \tag{V.5}$$

$$0 \in \partial f(x^\star) + A^* u^\star + \sum_i v_i^\star \partial g_i(x^\star) \tag{V.6}$$

$$\forall\, i, \quad u_i^\star g_i(x^\star) = 0 \tag{V.7}$$

2 Conjugate function and duality

To simplify and accelerate computation involving Lagrange duality, we introduce the *Legendre-Fenchel transform* which plays a similar role for convex function as the Fourier transform for signal or images.

2.1 Definitions

Definition 2.1. *Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper function. The conjugate function (or Legendre-Fenchel transformation) is a function $f^* : \mathbb{R}^n \to \overline{\mathbb{R}}$, defined by*

$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x} \in \mathrm{dom}(f)} \left\{ \boldsymbol{y}^T \boldsymbol{x} - f(\boldsymbol{x}) \right\}.$$

Proposition 2.1. *f^* is closed and convex (even when f is not).*

The conjugacy operation can be invoked twice resulting in the second conjugate:

$$f^{**}(\boldsymbol{x}) = \sup_{\boldsymbol{x} \in \mathbb{R}^n} (\boldsymbol{x}^T \boldsymbol{y} - f^*(y))$$

Theorem 2.4. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be a function. Then*

$$f^{**}(\boldsymbol{x}) \leq f(\boldsymbol{x}), \quad \forall\, \boldsymbol{x} \in \mathbb{R}^n.$$

If $f : \mathbb{R}^n \to (-\infty, +\infty]$ is closed and convex, then $f^{**} = f$.

Remarque 2.1. *f^{**} is the convex envelop of f (i.e. the largest convex function smaller than f).*

Example 2.1 (Indicator function)**.** *$f = \mathbb{1}_S$, where $S \subset \mathbb{R}^n$ is nonempty.*

$$f^*(\boldsymbol{y}) = \sup_{\boldsymbol{x} \in S} \boldsymbol{y}^T \boldsymbol{x}.$$

Example 2.2.

$$f(x) = \alpha x + \beta; \quad g(\boldsymbol{x}) = \|\boldsymbol{x}\|; \quad h(x) = \frac{c}{2} x^2.$$

$$g^*(y) = \begin{cases} \beta, & y = \alpha \\ +\infty, & y \neq \alpha \end{cases}; \quad g^*(\boldsymbol{y}) = \begin{cases} 0, & \|\boldsymbol{y}\|_* \leq 1 \\ \\ +\infty, & \|\boldsymbol{y}\|_* > 1 \end{cases}; \quad h^*(y) = \frac{1}{2c} y^2.$$

Example 2.3 (Quadratic)**.** $Q \succ 0$

$$f(\boldsymbol{x}) = \frac{1}{2} \boldsymbol{x}^T Q \boldsymbol{x} + \mathbf{b}^T \boldsymbol{x} + c; \quad f^*(\boldsymbol{y}) = \frac{1}{2} (\boldsymbol{y} - \boldsymbol{b})^T Q^{-1} (\boldsymbol{y} - \boldsymbol{b}) - c$$

2.2 Calculus rules

Theorem 2.5 (Calculus rules)**.**

- *Separable sum: If $g(x_1, x_2) = f_1(x_1) + f_2(x_2)$, then*

$$g^*(y_1, y_2) = f_1^*(y_1) + f_2^*(y_2)$$

- *Scaling ($\alpha > 0$): If $g(x) = \alpha f(x)$, then*
$$g^*(y) = \alpha f^*(y/\alpha),$$
and if $g(\boldsymbol{x}) = \alpha f(\boldsymbol{x}/\alpha)$ then $g^*(\boldsymbol{y}) = \alpha g^*(\boldsymbol{y})$.
- *Summation: If $g(x) = f_1(x) + f_2(x)$, then*
$$g^*(y) = \inf_z \{f_1^*(z) + f_2^*(y-z)\}$$
- *Addition to affine function*
$$f(x) = g(x) + a^T x + b; \quad f^*(y) = g^*(y-a) - b$$
- *Infimal convolution*
$$(f \otimes g)(\boldsymbol{x}) \overset{\text{def.}}{=} \sup_{\boldsymbol{y}+\boldsymbol{y}'=\boldsymbol{x}} f(\boldsymbol{y}) + g(\boldsymbol{y}').$$
$$(f+g)^* = f \otimes g \quad \textit{and} \quad (f \otimes g)^* = f + g.$$

2.3 Properties

Theorem 2.6 (Fenchel's inequality)**.** *the definition implies that*
$$f(x) + f^*(y) \geqslant x^T y; \quad \forall\, x, y$$
Theorem 2.7 (Conjugate Subgradient Theorem)**.** *Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper convex function.*
$$\forall\, (\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{x}^T \boldsymbol{y} = f(\boldsymbol{x}) + f^*(\boldsymbol{y}) \quad \Leftrightarrow \quad \boldsymbol{y} \in \partial f(\boldsymbol{x}).$$
If, in addition f is closed, then
$$\boldsymbol{y} \in \partial f(\boldsymbol{x}) \iff \boldsymbol{x} \in \partial f^*(\boldsymbol{y})$$

The conjugate subgradient theorem can be written as the following.

Corollary 2.1. *If f is proper closed and convex then*
$$\partial f^*(\boldsymbol{y}) = \operatorname*{argmax}_{\boldsymbol{x}} \{\boldsymbol{y}^T \boldsymbol{x} - f(\boldsymbol{x})\}$$
$$\partial f(\boldsymbol{x}) = \operatorname*{argmax}_{\boldsymbol{y}} \{\boldsymbol{x}^T \boldsymbol{y} - f^*(\boldsymbol{y})\}$$
Theorem 2.8 (Fenchel's duality theorem)**.** *Let $f, g : \mathbb{R}^n \to (-\infty, +\infty]$ be proper convex functions. If $\operatorname{relint}(\operatorname{dom}(f)) \cap \operatorname{relint}(\operatorname{dom}(g)) \neq \emptyset$, then*
$$\min_{\boldsymbol{x}} \{f(\boldsymbol{x}) + g(\boldsymbol{x})\} = \max_{\boldsymbol{y}} \{-f^*(\boldsymbol{y}) - g^*(-\boldsymbol{y})\},$$
and the maximum in the right-hand problem is attained whenever it is finite.

Proposition 2.2 (Legendre transform and smoothness)**.** *One has*
$$\nabla f \textit{ is L-Lipschitz} \quad \iff \quad \nabla f^* \textit{ is } \mu\textit{-strongly convex.}$$
Assume f is closed and strongly convex with parameter $\mu > 0$. Then, f^* is defined for all y and is differentiable everywhere, with gradient
$$\nabla f^*(\boldsymbol{y}) = \operatorname*{argmax}_{\boldsymbol{x}} \left(y^T x - f(x)\right).$$

2.4 Fenchel-Rockafellar Duality

Very often the Lagrange dual can be expressed using the conjugate of the function f. We give here a particularly important example, which is often called Fenchel-Rockafellar Duality.

Consider the generic primal problem

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + g(A\boldsymbol{x})$$

We rewrite the primal as:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) + g(\boldsymbol{y}), \\ \text{subject to} \quad & A\boldsymbol{x} = \boldsymbol{y}. \end{aligned} \tag{V.8}$$

From Lagrange duality:

$$\inf_{x,y} \left(f(x) + z^T Ax + g(y) - z^T y\right) = -f^*(-A^T z) - g^*(z) = q(\boldsymbol{z}).$$

We get the dual problem

$$\max_{\boldsymbol{z}} \; q(\boldsymbol{z}).$$

The next theorem gives primal-dual optimality conditions.

Theorem 2.9 (Fenchel-Rockafellar)**.** *Let* $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, $g : \mathbb{R}^m \to \overline{\mathbb{R}}$ *two convex functions and* $A : \mathbb{R}^n \to \mathbb{R}^m$. *If*

$$0 \in \operatorname{relint}(\operatorname{dom}(g) - A \operatorname{dom}(f))$$

the one has the following strong duality

$$\inf_{\boldsymbol{x}} \; (f(\boldsymbol{x}) + g(A\boldsymbol{x})) = \sup_{\boldsymbol{z}} \; - f^*(-A^T\boldsymbol{z}) - g^*(\boldsymbol{z})$$

Furthermore, (x, z) *is a pair of optimal primal-dual solutions if and only if*

$$-A^T z \in \partial f(x) \quad \textit{and} \quad Ax \in \partial g^*(z) \; (\textit{or } \boldsymbol{z} \in \partial g(Ax)). \tag{V.9}$$

Example 2.4 (Norm regularization)**.**

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + \|A\boldsymbol{x} - \boldsymbol{b}\|$$

The dual problem can be written as

$$\begin{aligned} \underset{\boldsymbol{z}}{\text{maximize}} \quad & - \boldsymbol{b}^T\boldsymbol{z} - f^*(-A^T\boldsymbol{z}) \\ \text{subject to} \quad & \|\boldsymbol{z}\|_* \leqslant 1 \end{aligned} \tag{V.10}$$

3 Algorithms

3.1 Dual Subgradient method

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & A\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

Dual Problem

$$\max_{\boldsymbol{z}} h(\boldsymbol{z}) = -f^*(-A^T\boldsymbol{z}) - \boldsymbol{b}^T\boldsymbol{z}$$

Subgradient of the Dual

- $\partial h(\boldsymbol{z}) = A\partial f^*(-A^T\boldsymbol{z}) - \boldsymbol{b}$
- $u \in \partial f^*(-A^T\boldsymbol{z}) \iff \boldsymbol{z} \in \underset{\boldsymbol{x}}{\operatorname{argmin}} \left\{ f(\boldsymbol{x}) + \boldsymbol{z}^T A\boldsymbol{x} \right\}$

If f is strictly convex, f^* is differentiable. We get dual gradient ascent.

$$\begin{aligned} x^{k+1} &\in \underset{\boldsymbol{x}}{\operatorname{argmin}} \left\{ f(\boldsymbol{x}) + (\boldsymbol{z}^k)^T A\boldsymbol{x} \right\}, \\ \boldsymbol{z}^{k+1} &= \boldsymbol{z}^k + \beta_k(A\boldsymbol{x}^{k+1} - \boldsymbol{b}) \end{aligned}$$

Dual Decomposition

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \sum_{i=1}^{n} f_i(\boldsymbol{x}_i) \\ \text{subject to} \quad & A\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

Algorithm

$$\begin{aligned} x_i^{k+1} &\in \underset{\boldsymbol{x}_i}{\operatorname{argmin}} \left\{ f_i(\boldsymbol{x}_i) + (\boldsymbol{z}^k)^T A_i\boldsymbol{x}_i \right\}, \\ \boldsymbol{z}^{k+1} &= \boldsymbol{z}^k + \beta_k(\sum_{i=1}^{n} A_i\boldsymbol{x}_i^{k+1} - \boldsymbol{b}) \end{aligned}$$

3.2 Forward-backward on the Dual

Problem 1

$$\begin{aligned} &\text{primal: } \min_{\boldsymbol{x}} f(\boldsymbol{x}) + g(\boldsymbol{x}), \\ &\text{dual: } \quad \max_{\boldsymbol{z}} \; - g^*(\boldsymbol{z}) - f^*(-\boldsymbol{z}) \end{aligned}$$

Use Moreau decomposition to simplify DR iteration

$$\begin{aligned} x_{k+1} &= \operatorname{prox}_f(y_k), \\ y_{k+1} &= x_{k+1} + \operatorname{prox}_{g^*}(2x_{k+1} - y_k) \end{aligned}$$

Make change of variables $z_k = x_k - y_k$:

$$\begin{aligned} x_{k+1} &= \operatorname{prox}_f(x_k - z_k), \\ z_{k+1} &= \operatorname{prox}_{g^*}(z_k + 2x_{k+1} - x_k) \end{aligned}$$

Problem 2: constrained convex problem

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x \in V \end{aligned} \tag{V.11}$$

f closed and convex; V a subspace.

- Douglas–Rachford splitting with $g = \delta_V$

$$\begin{aligned} x_{k+1} &= \text{prox}_f(y_k), \\ y_{k+1} &= y_k + P_V(2x_{k+1} - y_k) - x_{k+1} \end{aligned}$$

- Primal-dual form

$$\begin{aligned} x_{k+1} &= \text{prox}_f(x_k - z_k), \\ z_{k+1} &= P_{V^\perp}(z_k + 2x_{k+1} - x_k) \end{aligned}$$

Problem 3: Equality constraints optimization problem

Consider the "primal" proble

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & Ax = b \end{aligned} \tag{V.12}$$

The dual problem is

$$\max_{\boldsymbol{z}} \; -b^T\boldsymbol{z} - f^*(-A^T\boldsymbol{z})$$

Dual gradient ascent algorithm

$$\begin{aligned} \hat{x} \; &= \underset{\boldsymbol{x}}{\text{argmin}} \; \left(f(\boldsymbol{x}) + \boldsymbol{z}^T A\boldsymbol{x}\right), \\ z^+ &= z + t(A\hat{x} - b) \end{aligned}$$

- Step one: compute a subgradient $\hat{x} \in \partial f^*(-A^T\boldsymbol{z})$
- Step two: compute a subgradient $b - A\hat{x}$ of $\boldsymbol{b}^T\boldsymbol{z} + f^*(-A^T\boldsymbol{z})$ at $\boldsymbol{z}$
- Step 3: Update $\boldsymbol{z}$: $\boldsymbol{z}^+ = \boldsymbol{z} + t(A\hat{x} - \boldsymbol{b})$.

This algorithm is of interest if calculation of $\hat{x}$ is inexpensive (for example, f is separable).

Problem 4: composite optimization problem

Consider the "primal" problem

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) + g(A\boldsymbol{x})$$

f_1 and f_2 have simple prox-operators.

The associated dual problem

$$\max_{\boldsymbol{z}} \; - f^*(-A^T z) - g^*(z)$$

or

$$\min_{\boldsymbol{z}} g^*(z) + f^*(-A^T z)$$

- proximal gradient update:

$$z^+ = \operatorname{prox}_{tg^*}\Big(\boldsymbol{z} + tA\nabla f^*(-A^T\boldsymbol{z})\Big)$$

where

$$\nabla f^*(-A^Tz) = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ \Big(f(x) + z^TAx\Big)$$

Total variation image denoisng

A typical example, which was the one used by Antonin Chambolle to develop this class of method, is the total varation denoising. Consider that images $\boldsymbol{x} \in \mathbb{R}^N$ are represented by arrays X of size (n, n) and gradient vector fields $\nabla\boldsymbol{x} = \boldsymbol{u} \in \mathbb{R}^P$ are represented by arrays U size $(n, n, 2)$, where $P = N \times 2$. The total variation denoising model writes as

$$\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2 + \lambda\|\nabla\boldsymbol{x}\|_{1,2}$$

where $\|\cdot\|_{1,2}$ is the vectorial-ℓ^1 norm (also called $\ell^1 - \ell^2$ norm):

$$\|\boldsymbol{u}\|_{1,2} \overset{\text{def.}}{=} \sum_{i,j}(U_{i,j,1})^2 + (U_{i,j,2})^2.$$

The primal problem corresponds to minimizing $E(x) = f(x) + g(A(x))$ where

$$f(x) = \frac{1}{2}\|x - y\|^2 \quad \text{and} \quad g(u) = \lambda\|u\|_{1,2}.$$

The dual problem corresponds to minimzing $F(u) + G(u)$ where

$$F(u) = \frac{1}{2}\|y - A^*u\|^2 - \frac{1}{2}\|y\|^2 \quad \text{and} \quad G(u) = \iota_{\mathcal{C}}(u) \quad \text{where} \quad \mathcal{C} = \{u\ ;\ \|u\|_{\infty,2} \leqslant \lambda\},$$

and

$$\|u\|_{\infty,2} = \max_{i,j}\ \|u_{i,j}\|.$$

One can thus solves the ROF problem by computing

$$x^\star = y - A^*u^\star$$

where

$$u^\star \in \underset{\|u\|_{1,2}\leqslant\lambda}{\operatorname{argmin}}\ \|y - A^*u\|.$$

One can compute explicitly the gradient of F:

$$\nabla F(u) = A(A^*u - y).$$

The proximal operator of G is the orthogonal projection on $\mathcal{C}$, which is obtained as

$$\operatorname{prox}_{\gamma G}(u)_{i,j} = \frac{u_{i,j}}{\max(1, \|u_{i,j}\|/\lambda)}.$$

Note that it does not depends on γ. The gradient step size of the FB should satisfy

$$\gamma < \frac{2}{\|A^*A\|} = \frac{1}{4}.$$

4 Exercises

Chapter VI

Applications to image processing

Skills to acquire

- **Theoretical**:
- **Practical**: Compute subdifferential, Compute conjugate, Compute

We consider a general convex optimization problem

$$\min_{x\in\mathbb{R}^n} f(x) \tag{VI.1}$$

1 Sparse solutions

2 Compressed Sensing