Lecture 7 (and Lecture 6, part 2): Convex Optimization (II). Duality principles and methods. Applications.

Coloma Ballester - UPF

November 2, 2021

Optimization and inference techniques for Computer Vision

Previously on...

Unconstrained and constrained optimization
Do we still have many open questions?

Does our problem have a solution?
(Existence) ✓

Does our problem have an unique solution?
(Uniqueness) ✓

How do we know if a point x is a solution?
(Optimality conditions) ✓

Is it possible to find the solution?
(Convexity) ✓

Does our problem still have solutions if we have restrictions on them?
(Constrained Optimization) ✓

Convex problems are easy

In the previous lectures you have worked with problems in where we can answer some of the previous questions.

In convex problems we can assure the existence of solutions.

In strictly convex problems we can assure the uniqueness of their solutions.

If the general problem is (strictly) convex and the restrictions on our solutions enclose them in convex sets then we can assure the (uniqueness and) existence of solutions.

Convex constrained minimization

Consider the **constrained minimization problem**

$$\min_{x \in C} f(x).$$

Theorem

Assume that C is a **convex subset** of $\mathbb{R}^n$. Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be a **convex function**.

Then, **a local minimum of f over C is also a global minimum over C**.

Moreover, if f is **strictly convex**, then **any global minimum in C is unique**.

Convex problems

Corollary

If C is a **closed convex subset** of $\mathbb{R}^n$ and $f : C \to \mathbb{R}$ is **convex and continuous** on C, then f **attains its infimum**.

That is, if we solve

$$\inf_{x \in C} f(x)$$

there is a point $x_0 \in C$ such that

$$f(x_0) = \min_{x \in C} f(x).$$

In other words, **convex functions on (closed) convex sets have minima.**

Convex constrained optimization

How to compute the minimum of a convex function with convex restrictions on its variables?

The solutions will satisfy the so-called the **Karush-Kuhn-Tucker (KKT) optimality conditions.**

The KKT optimality conditions are the necessary and sufficient conditions of a minimum. They allow to write equations to compute the solution to the problems.

General case: equality and inequality constraints

Consider the smooth functions $f, c_1, \cdots, c_k, d_1, \ldots, d_r : \mathbb{R}^n \to \mathbb{R}$.
Usually, f is called the **objective function**.

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} f(x) \\ &\text{s.t.} \quad c_1(x) \geq 0, \ldots, c_k(x) \geq 0 \qquad \text{(inequality constraints)} \\ &\text{and} \quad d_1(x) = 0, \ldots, d_r(x) = 0 \qquad \text{(equality constraints)} \end{aligned} \tag{1}$$

Define the Lagrange function for the problem as Lagrange multipliers λ_i, $i = 1, \ldots, k$, ν_j, $j = 1, \ldots, r$ of the problem. Setting $\lambda = (\lambda_1, \ldots, \lambda_k)$, $\nu = (\nu_1, \ldots, \nu_r)$, the Lagrange function

$$\mathscr{L}(x, \lambda, \nu) = f(x) - \sum_{i=1}^{k} \lambda_i c_i(x) - \sum_{j=1}^{r} \nu_j d_j(x)$$

Karush-Kuhn-Tucker (KKT) optimality conditions

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f(x) \\ &\text{s.t.} \quad c_1(x)\geq 0,\ldots,c_k(x)\geq 0 \qquad \text{(inequality constraints)} \\ &\text{and} \quad d_1(x)=0,\ldots,d_r(x)=0 \qquad \text{(equality constraints)} \end{aligned} \tag{2}$$

Suppose that there is a **minimum x_0 of (2)**.
It coincides with a **saddle point (x_0,λ^0,ν^0) of $\mathscr{L}(x,\lambda,\nu)$**.

The **Karush-Kuhn-Tucker (KKT) optimality conditions** are

$$\begin{aligned} \nabla_x\mathscr{L}(x_0,\lambda^0)=0 & \qquad && \text{dual feasibility} \\ c_i(x_0)\geq 0 & \quad \forall i && \text{primal feasibility} \\ d_j(x_0)=0 & \quad \forall j && \text{primal feasibility} \\ \lambda_i^0\geq 0 & \quad \forall i && \text{dual positivity} \\ \lambda_i^0 c_i(x_0)=0 & \quad \forall i && \text{complementary slackness} \end{aligned}$$

Notice that the equations $\lambda_i^0 c_i(x_0)=0$ mean that:

- either $c_i(x_0)=0$ (**active constraint**) and therefore $\lambda_i^0>0$,
- or $c_i(x_0)>0$ (x_0 **interior point**) and therefore $\lambda_i^0=0$.

Karush-Kuhn-Tucker (KKT) optimality conditions

The KKT optimality conditions are

$$\begin{aligned}
\nabla_x \mathscr{L}(x_0, \lambda^0) &= 0 && && \text{dual feasibility} \\
c_i(x_0) &\geq 0 && \forall i && \text{primal feasibility} \\
d_j(x_0) &= 0 && \forall j && \text{primal feasibility} \\
\lambda_i^0 &\geq 0 && \forall i && \text{dual positivity} \\
\lambda_i^0 c_i(x_0) &= 0 && \forall i && \text{complementary slackness}
\end{aligned}$$

The KKT optimality conditions are **necessary conditions**, that is, they hold for a minimum of (1).

If the constraints $-\mathbf{c_i(x)}$ **are convex and there is a point x such that** $\mathbf{c_i(x) > 0}$ **for all** $i = 1, \ldots, k$, then they are also **sufficient conditions**, that is, if they hold, the point $\bar{x}$ is a minimum of (1).

Karush-Kuhn-Tucker (KKT) optimality conditions

Remark the role played by considering only $\lambda \geq 0$.

In other words, while the sign of the Lagrange multipliers in case of equality constraints is not specified, in case of inequality constraints is specified.

This is because the KKT conditions come of considering Lagrange multipliers λ_i, $i = 1, \ldots, k$, ν_j, $j = 1, \ldots, r$ of the problem

$$\mathscr{L}(x, \lambda, \nu) = f(x) - \sum_{i=1}^{k} \lambda_i c_i(x) - \sum_{j=1}^{r} \nu_j d_j(x)$$

Nevertheless, the fact that they are positive is a convention. Notice that **we put a minus sign in front of the constraint terms in the Lagrangian**, so $\mathscr{L}$ is convex as a function of x.

Convex optimization: We still have a quite big open question

Does our problem have a solution?
(Existence) ✓

Does our problem have an unique solution?
(Uniqueness) ✓

How do we know if a point x is a solution?
(Optimality conditions) ✓

Is it possible to find the solution?
(Convexity) ✓

Does our problem still have solutions if we have restrictions on them?
(Constrained Optimization) ✓

Can we still find solutions for non-differentiable problems?
(Non-smooth Optimization) ✗

In general: Optimization problems

VA:

$$\min_{\mathbf{u} \in \mathbf{C}} \mathbf{J}(\mathbf{u})$$

AA:

$$\min_{\mathbf{x} \in \mathbf{C}} \mathbf{f}(\mathbf{x})$$

$\longrightarrow$ **What happens if you can't differentiate because either u belongs to an space of non-derivable functions, or if $J(u)$ is not derivable?**

Non-differentiable case

This creates a difficulty: **the two optimization strategies** you have learned so far,

- **Euler-Lagrange equation** (which is a **extremality principle**):
  $\frac{\mathbf{dJ}}{\mathbf{du}}(\mathbf{u_0}) = \mathbf{0} \iff \nabla \mathbf{J}(\mathbf{u_0}) = \mathbf{0}$ (analogy with $\nabla f(x_0) = 0$)

- **Gradient descent**:
$$\begin{cases} \mathbf{u^{k+1}} = \mathbf{u^k} - \tau \nabla \mathbf{J}(\mathbf{u^k}) \\ \mathbf{u^0} = \mathbf{u_0} \end{cases}$$

use the 'derivative of the functional J(u) with respect to the function u', $\nabla_{\mathbf{u}}\mathbf{J}$
(also denoted by $\frac{\mathbf{dJ}}{\mathbf{du}}$).

What happens if you can't because either u belongs to an space of non-derivable functions, or if J is not derivable??

Examples: Convex Problems you have already seen...

... written as finite dimensional problems (i.e. as matrices and vectors).

Image denoising

Given f a noisy image, recover u as the solution of

$$\min_u \|\nabla^h u\|^2 + \lambda \|u - f\|^2$$

Image inpainting

Given f an image and a mask M defining the region that should be preserved:

$$\min_u \|\nabla^h u\|^2 \quad \text{s.t. } M \odot u = f$$

Remember that approximating ∇ with finite differences it can be expressed as a matrix.

However, some non-smooth functionals yield sharper results...

... more similar to real world scenes which are made of well-contrasted objects that, in the image or video capturing the scene, frequently partially occlude other objects and the backgorund.

Total Variation image denoising (aka ROF denoising model)

Given f a noisy image, recover u as the solution of

$$\min_u \|\nabla^h u\| + \lambda \|u - f\|^2$$

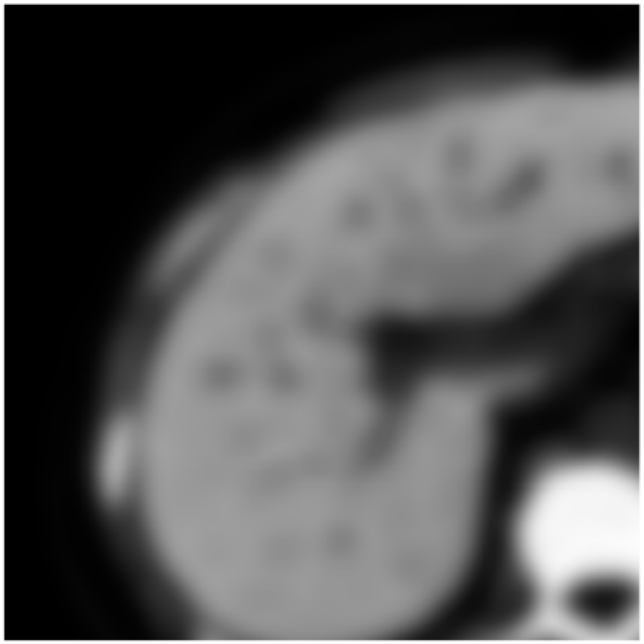
Image inpainting

Given f an image and a mask M defining the region that should be preserved:

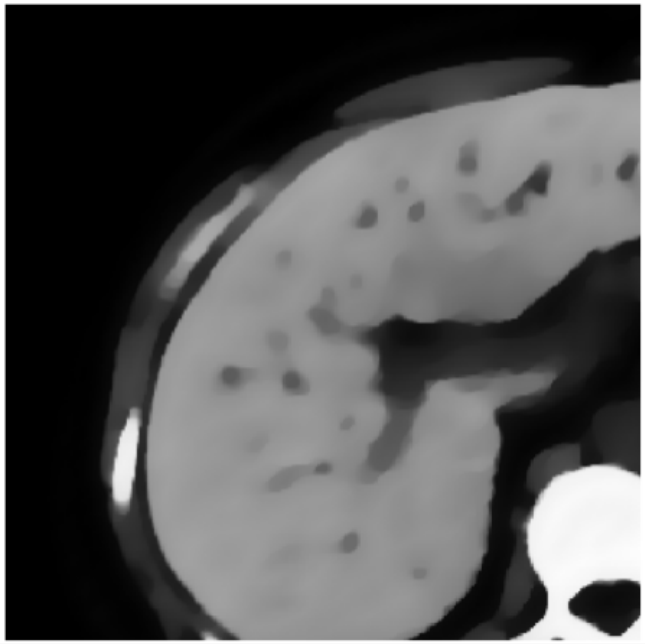
$$\min_u \|\nabla^h u\| \quad \text{s.t. } M \odot u = f$$

Image denoising problem $J(u) = \frac{1}{2\lambda} \int_\Omega |u - f|^2 d\mathbf{x} + \frac{1}{2} \int_\Omega |\nabla u|^p d\mathbf{x}$

p=2



p=1



Indeed, some non-smooth functionals yield sharper results

• **Example ($p = 1$):**, Total Variation image denoising (aka ROF denoising model):

$$J(u) = \frac{1}{2\lambda} \int_\Omega |u - f|^2 d\mathbf{x} + \frac{1}{2} \int_\Omega |\nabla u| d\mathbf{x}$$

It is also a convex function. But we cannot "differentiate" as you did with before with other differenciable functionals.

• Moreover, $u \in C$ can be non-differentiable (for instance, if u is a piecewise constant image).

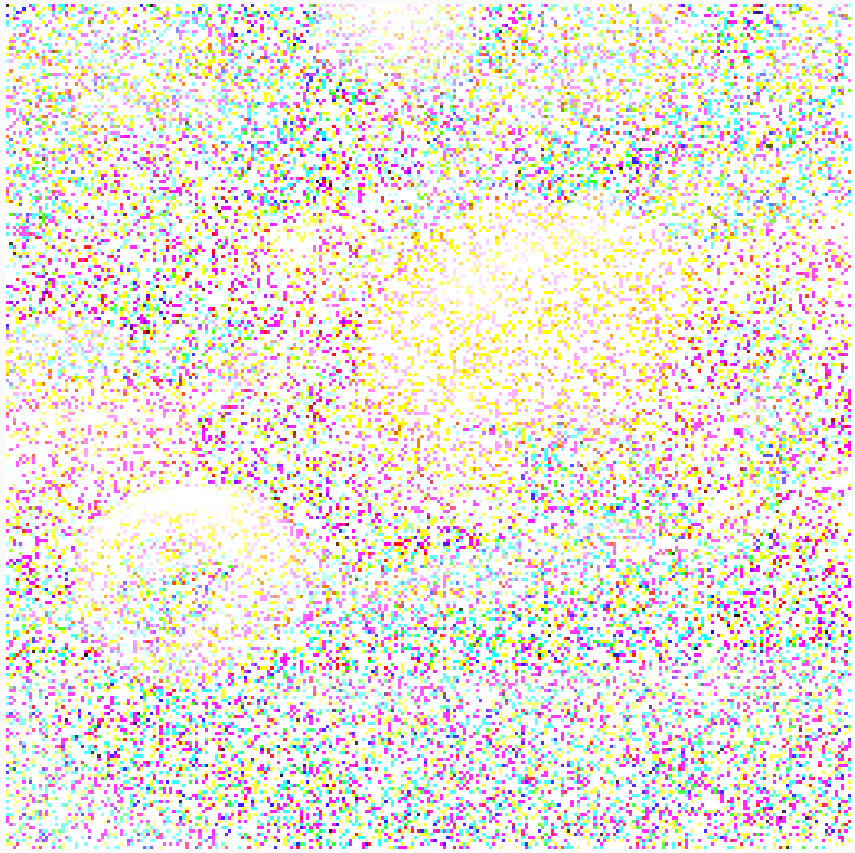
• **Another example:** optical flow estimation in video by minimization of the **TV**-L^1 optical flow functional

$$\min_{\mathbf{v}} \lambda \int_\Omega |(I(\mathbf{x} + \mathbf{v}, t+1) - I(\mathbf{x}, t)| d\mathbf{x} + \int_\Omega |\nabla \mathbf{v}| d\mathbf{x}$$

where $\mathbf{v}$ is the (unknown) optical flow, that is, the vector field that recovers the apparent motion of two consecutive frames of the input video.

Similar example: Inpainting with $\mathbf{p = 1}$

Recovering/interpolating an image in regions where the original information is missing. Given the color image $f \in L^\infty(\Omega)$, the image domain Ω an the region $D \subset \Omega$ where the image will be inpainted.



$$\begin{cases} \inf_u \int_D |\nabla u| dx \\ u_{|\partial\Omega_I} = f \end{cases}$$

Reformulated as:

$$\begin{cases} \inf_{u,v} \int_\Omega |\nabla u| dx + \dfrac{1}{2\lambda} \int_\Omega |u - v|^2 dx \\ v_{|\Omega \setminus D} = f \end{cases}$$

How to solve these problems?

• In this lecture we will study a trick to deal with some non-differentiable functions: **Dual and Primal-Dual methods**

• It is based in augmenting the objective function, including new variables, the (auxiliary) dual variables.

Duality. The dual problem

We will first describe how to compute the **dual problem of a given constrained optimization problem.**

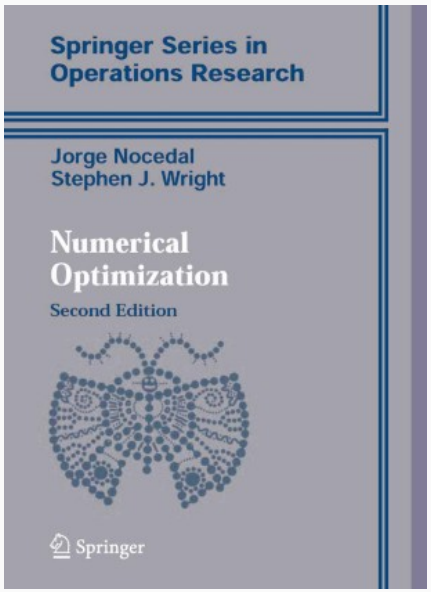
We will study primal, dual and primal-dual formulation of the problems and numerical algorithms that use those formulations to solve them.

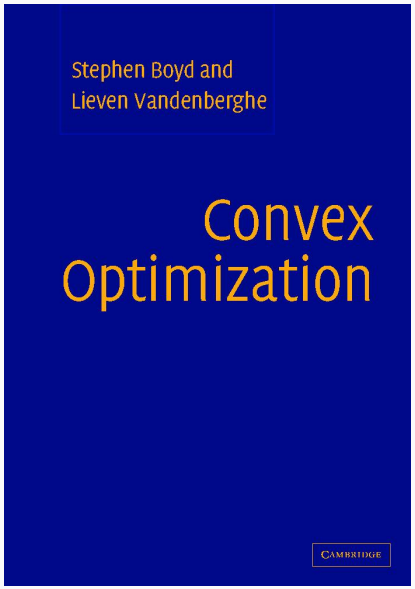
Then, we will solve **non-differentiable problems with dual and primal-dual methods.**

Primal dual methods are an example of interior point methods, that is methods that look for a solution from the interior of the set determined by the constraints.

Bibliography

- Nocedal, J., Wright, S.J., "Numerical Optimization", Springer.
- Boyd, S., Vandenberghe, L., "Convex Optimization", Cambridge University Press.
  http://www.stanford.edu/~boyd/cvxbook/
- Stamford course on Convex Optimization II
  https://web.stanford.edu/class/ee364b/lectures.html





Outline

1. Duality: Min-max Theorem.
2. Lagrangian duality.
3. Primal-dual and dual approaches.
    - Solving the primal/original problem via solving its dual problem.
4. Applications.
5. Non-convex problems and convex relaxation.

Outline

1. **Duality: Min-max Theorem.**
2. Lagrangian duality.
3. Primal-dual and dual approaches.
4. Applications.
5. Non-convex problems and convex relaxation.

Min-max theorem

Min-max theorem. Let $L : X \times Y \to \mathbb{R}$ any function of two variables, $x \in X \subset \mathbb{R}^n$, $y \in Y \subset \mathbb{R}^m$. Always we have

$$\max_y \min_x L(x,y) \leq \min_x \max_y L(x,y),$$

assuming that the minima and maxima exist, otherwise we replace them by inf and sup.

Indeed, Observe that for any x, y we have

$$\min_{\tilde{x}} L(\tilde{x},y) \leq L(x,y)$$

Take max in y to get

$$\max_y \min_{\tilde{x}} L(\tilde{x},y) \leq \max_y L(x,y).$$

The left hand side does not depend on x. Take min in x.

Duality gap

The difference

$$DG := \min_x \max_y L(x,y) - \max_y \min_x L(x,y) \geq 0$$

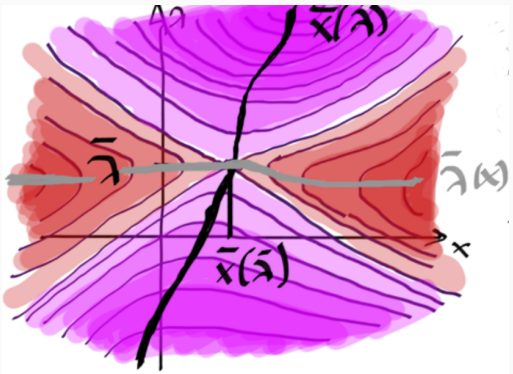
is called the **duality gap**.

If $DG = 0$ and (x_0, y_0) is such that

$$L(x_0,y_0) = \min_{x \in X} L(x,y_0) \quad \text{and} \quad L(x_0,y_0) = \max_{y \in Y} L(x_0,y)$$

then (x_0, y_0) is called a **saddle point**. It satisfies

$$L(x_0,y) \leq L(x_0,y_0) \leq L(x,y_0) \qquad \forall x, y.$$



If there exists a saddle point, then the dual gap is $DG = 0$. (**necessary condition**)

Sufficient condition for a saddle point

Theorem

Assume that X, Y are closed convex sets,

$$\mathbf{x} \in X \to L(\mathbf{x}, y) \qquad \text{is } \mathbf{convex} \quad \text{for} \quad \text{all} \quad y \in Y,$$

$$\mathbf{y} \in Y \to L(x, \mathbf{y}) \qquad \text{is } \mathbf{concave} \quad \text{for} \quad \text{all} \quad x \in X,$$

and $\quad$ either X is bounded or $\exists \bar{y} \in Y$ such that $L(x, \bar{y}) \to \infty$ as $x \to \infty$,

either Y is bounded or $\exists \bar{x} \in X$ such that $L(\bar{x}, y) \to -\infty$ as $x \to \infty$.

Then $DG = 0$ and L has a **saddle point** (x_0, y_0) in $X \times Y$.

This result is the **basis of the duality theory**.

Outline

1. Duality: Min-max Theorem.
2. **Lagrangian duality (and saddle points).**
3. Primal-dual and dual approaches.
4. Applications.
5. Non-convex problems and convex relaxation.

Lagrangian Duality

Given differentiable functions $f, c_1, \cdots, c_k, d_1, \ldots, d_r : \mathbb{R}^n \to \mathbb{R}$, let us consider the general optimization problem with inequality and equality constraints

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} f(x) \\ &\text{subject to} \quad c_1(x) \geq 0, \ldots, c_k(x) \geq 0 \quad \text{(inequality constraints)} \\ &\text{and} \qquad d_1(x) = 0, \ldots, d_r(x) = 0 \qquad \text{(equality constraints)} \end{aligned} \tag{3}$$

Assume that the set determined by the constraints, that is
$C = \{x \in \mathbb{R}^n \text{ such that } c_1(x) \geq 0, \ldots, c_k(x) \geq 0 \text{ and } d_1(x) = 0, \ldots, d_r(x) = 0\}$,
is non-empty, and suppose that there is a minimum $x_0 \in C$ of (3).

Consider **Lagrange multipliers** λ_i, $i = 1, \ldots, k$, ν_j, $j = 1, \ldots, r$ of the problem, where $\lambda \geq 0$ (meaning $\lambda_i \geq 0, \forall i$). The Lagrange multipliers are also called the **dual variables**.

Denoting by $\lambda = (\lambda_1, \ldots, \lambda_k)$, $\nu = (\nu_1, \ldots, \nu_r)$, the **Lagrange function** is

$$\mathscr{L}(x, \lambda, \nu) = f(x) - \sum_{i=1}^{k} \lambda_i c_i(x) - \sum_{j=1}^{r} \nu_j d_j(x)$$

Lagrangian Duality

Given

$$\mathscr{L}(x,\lambda,\nu) = f(x) - \sum_{i=1}^{k} \lambda_i c_i(x) - \sum_{j=1}^{r} \nu_j d_j(x)$$

we have

(1) $\mathscr{L}(x,\lambda,\nu) \leq f(x) \qquad \forall \lambda \geq 0\ (\lambda_i \geq 0\ \forall i),\ \nu \in \mathbb{R}^r,\ \forall x \in C.$

(2) $$\max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \mathscr{L}(x,\lambda,\nu) = \tilde{f}(x) = \begin{cases} f(x) & \text{if } x \in C \\ +\infty & \text{if } x \notin C \end{cases}$$

(3) $$\min_{x \in C} f(x) = \min_{x \in \mathbb{R}^n} \bar{f}(x) = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \mathscr{L}(x,\lambda,\nu).$$

That is

$$\min_{x \in C} = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} f(x) - \sum_{i=1}^{k} \lambda_i c_i(x) - \sum_{j=1}^{r} \nu_j d_j(x),$$

and **the dual variables satisfy $\lambda \geq 0$ and there is no restriction on ν, i.e., $\nu \in \mathbb{R}^r$.**

The dual problem

We have seen that solving our mininimization problem is equivalent to solve the min-max problem:

$$\min_{x \in C} f(x) = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \mathscr{L}(x, \lambda, \nu).$$

By the Min-max theorem, we know that by changing min-max by max-min, we have

$$\max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \min_{x \in \mathbb{R}^n} \mathscr{L}(x, \lambda, \nu) \leq \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \mathscr{L}(x, \lambda, \nu)$$

The duality gap is

$$DG = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \mathscr{L}(x, \lambda, \nu) - \max_{\lambda \geq 0, \nu \in \mathbb{R}^r} \min_{x \in \mathbb{R}^n} \mathscr{L}(x, \lambda, \nu) \geq 0.$$

The dual problem

Let's assume that $DG = 0$ (in other words, it exists a saddle point).

This is guaranteed if f is convex, $-c_i(x)$ are convex constraints, and d_i are linear constraints: It is a consequence of the Theorem giving the **sufficient conditions**. Obviously, we need the remaining mild assumptions to guarantee the rest of assumptions of the Theorem, but they are usually satisfied in practice. Therefore,

$$\min_{x \in C} f(x) = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, v \in \mathbb{R}^r} \mathscr{L}(x, \lambda, v) = \max_{\lambda \geq 0, v \in \mathbb{R}^r} \min_{x \in \mathbb{R}^n} \mathscr{L}(x, \lambda, v).$$

The function

$$\mathbf{g_D}(\lambda, v) = \min_{x \in \mathbb{R}^n} \mathscr{L}(x, \lambda, v)$$

is called the Lagrange dual function, or simply the **dual function**.

As $0 = DG = \min_{x \in C} f(x) - \max_{\lambda \geq 0, v \in \mathbb{R}^r} g_D(\lambda, v)$, the original problem can be re-stated as

$$\min_{x \in C} f(x) = \min_{x \in \mathbb{R}^n} \max_{\lambda \geq 0, v \in \mathbb{R}^r} \mathscr{L}(x, \lambda, v) = \max_{\lambda \geq 0, v \in \mathbb{R}^r} \min_{x \in \mathbb{R}^n} \mathscr{L}(x, \lambda, v) = \max_{\lambda \geq 0, v \in \mathbb{R}^r} g_D(\lambda, v). \quad (4)$$

That is,

$$\min_{x \in C} f(x) = \max_{\lambda \geq 0, v \in \mathbb{R}^r} g_D(\lambda, v).$$

The (right-hand) problem in (4) is the **dual problem** of (3) (i.e., the left-hand problem in (4), which is called the **primal problem**, $f(x)$ the **primal function** and x the **primal variable**. *Sometimes it is much easier to solve the dual problem than the primal one.*

Summary

By the *theorem with the sufficient condition*, if f is convex, $-c_i$ convex (c_i concave) and d_i linear (plus the mild assumptions), then $DG = 0$ and there is a saddle point $(x_0, \lambda_0, \nu_0) \in \mathbb{R}^n \times \mathbb{R}^{k+r}$.

In this case, the following three problems are equivalent:

(1) Primal problem: $\min_{x \in C} f(x)$.

(2) Dual problem: $\max_{\lambda \geq 0, \nu \in \mathbb{R}^r} g_D(\lambda, \nu)$.

(3) Primal-Dual problem: find a sadle point of $\mathscr{L}(x, \lambda, \nu)$.

Example: Computing the dual problem

Let A be an $m \times n$ matrix, $b \in \mathbb{R}^m$. For $x \in \mathbb{R}^n$, consider the problem

$$\begin{aligned} &\min \|x\|^2 \qquad\qquad (\mathrm{P}) \\ &\text{subject to } Ax = b \end{aligned}$$

Let's compute (and solve) its dual problem.

- Let's write problem (P) as a min-max problem and define the duality gap.

$Ax = b$ gives m equality constraints on x: $(Ax)_i = b_i$, $i = 1, \cdots, m$. Therefore, we introduce m Lagrange multipliers (or dual variables), $\nu_1, \ldots, \nu_m \in \mathbb{R}$, and we construct the Lagrangian function, depending on $n+m$ variables

$$\mathscr{L}(x, \nu) = f(x) - \sum_{i=1}^{m} \nu_i((Ax)_i - b_i) = \langle x, x\rangle - \langle \nu, Ax - b\rangle = \langle x, x\rangle - \langle A^t\nu, x\rangle + \langle \nu, b\rangle,$$

where $\nu = (\nu_1, \ldots, \nu_m)^t \in R^m$. Therefore

$$\min_{\text{subject to } Ax=b} \langle x, x\rangle = \min_{x \in R^n} \max_{\nu \in R^m} \mathscr{L}(x, \nu)$$

The duality gap is the difference

$$DG = \min_{x \in R^n} \max_{\nu \in R^m} \mathscr{L}(x, \nu) - \max_{\nu \in R^m} \min_{x \in R^n} \mathscr{L}(x, \nu).$$

which is always ≥ 0.

Example: Computing the dual problem

- Let's now define and compute the dual function of problem (P).

Remember

$$\mathscr{L}(x,\nu) = f(x) - \sum_{i=1}^{m} \nu_i((Ax)_i - b_i) = \langle x,x\rangle - \langle \nu, Ax-b\rangle = \langle x,x\rangle - \langle A^t\nu, x\rangle + \langle \nu, b\rangle,$$

In our case, $DG = 0$ because

- $\mathscr{L}(x,\nu)$ is convex with respect to x (for each ν fixed). Indeed,
  - f is convex because it is a quadratic function which Hessian is equal to $2I$, a strictly positive definite matrix,
  - and $d_i(x) = (Ax)_i - b_i$ are linear constraints, thus $-d_i(x)$ is convex with respect to x.
- $\mathscr{L}(x,\nu)$ is concave with respect to ν (for each x fixed) because it is a linear function on each of the variables ν_j, thus concave with respect to ν.

Therefore, $DG = 0$, there exists a saddle point (x^*, ν^*) and we can change min-max by max-min:

Example: Computing the dual problem

Therefore, $DG = 0$, there exists a saddle point (x^*, v^*) and we can change min-max by max-min:

$$\min_{\text{subject to } Ax=b} \langle x, x \rangle = \min_{x \in R^n} \max_{v \in R^m} \mathscr{L}(x, v) = \max_{v \in R^m} \min_{x \in R^n} \mathscr{L}(x, v) = \max_{v \in R^m} g_D(v),$$

where

$$g_D(v) = \mathscr{L}(x^*(v), v) \qquad \text{with} \qquad x^*(v) = \arg\min_{x \in R^n} \mathscr{L}(x, v)$$

is the dual function. We compute the dual function by solving $\min_{x \in R^n} \mathscr{L}(x, v)$. As $\mathscr{L}(x, v)$ is strictly convex on $x \in \mathbb{R}^n$, a necessary and sufficient condition of (the unique) minimum is $\nabla_x \mathscr{L}(x^*, v) = 0$.

In our case, $2x - A^t v = 0$, which gives $x^*(v) = \frac{1}{2} A^t v$. Then,

$$g_D(v) = \mathscr{L}(x^*(v), v) = -\frac{1}{4} \langle A^t v, A^t v \rangle + \langle v, b \rangle$$

Example: Computing the dual problem

- Let's write down the dual problem and solve it.

$$\max_{v\in R^m}\left(-\frac{1}{4}\langle A^t v, A^t v\rangle + \langle v, b\rangle\right) = \max_{v\in R^m}\left(-\frac{1}{4}\langle AA^t v, v\rangle + \langle v, b\rangle\right) =$$
$$= \min_{v\in R^m}\left(\frac{1}{4}\langle AA^t v, v\rangle - \langle v, b\rangle\right).$$

The last function is convex with respect to v as its Hessian is equal to $\frac{1}{2}AA^t$ which is strictly positive definite. Thus, there exists an only minimum which is found by imposing that the gradient is equal to 0. Doing the computations, we obtain the **solution of the dual problem**

$$v^* = 2(AA^t)^{-1}b.$$

Here we have assumed that AA^t is invertible, which might well not to be the case.

Finally, the **solution of the primal problem** is

$$x^* = x^*(v^*) = A^t(AA^t)^{-1}b.$$

Let's verify that satisfies the constraint of the primal problem:

$$Ax^* = AA^t(AA^t)^{-1}b = b.$$

Constrained optimization

Exercise: Let $c \in \mathbb{R}^n$ be a given vector/point of $\mathbb{R}^n$, A be a given $m \times n$ matrix, and $b \in \mathbb{R}^m$. For $x \in \mathbb{R}^n$, consider the problem

$$\begin{aligned} &\min \|x - c\|^2 \\ &\text{subject to } Ax = b \end{aligned}$$

Compute its dual problem. Solve it.

More examples: Exercises in exams of past years

Outline

1. Duality: Min-max Theorem.
2. Lagrangian duality (and saddle points).
3. **Primal-dual and dual approaches (for some non-differentiable problems).**
4. Applications.
5. Non-convex problems and convex relaxation.

Duality principles for non-smooth problems

Let's first analyze it on an example:

Goal: minimize the function $f : \mathbb{R}^n \to \mathbb{R}$ defined by

$$f(x) = \|Ax\|_{\mathbb{R}^m} + \frac{1}{2\lambda}\|x - b\|^2_{\mathbb{R}^n}$$

for $x \in \mathbb{R}^n$, where A is a $m \times n$ matrix, b is a given vector in $\mathbb{R}^n$, and $\lambda > 0$.

This function is not differentiable when $Ax = 0$. That is, on the points of $KerA$, the function f is not differentiable.

The minimization algorithms we have seen so far use ∇f in a way or another. Here we cannot use it, since $\nabla f(x)$ is not defined in the case of $Ax = 0$.

Using duality we can remove the non-differentiability introducing an additional variable and formulating an equivalent problem.

Removing the non-differentiability with an auxiliary variable

We can remove the non-differentiability by formulating an equivalent problem with an additional variable.

It is based on the following observation:

$$\|y\|_{\mathbb{R}^m} = \max_{\|\xi\|_{\mathbb{R}^m} \leq 1} \langle y, \xi \rangle_{\mathbb{R}^m}$$

Exercise: Prove it using that for $x, y \in \mathbb{R}^m$, $\langle x, y \rangle = \|x\| \|y\| cos\theta$.

Removing the non-differentiability with an auxiliary variable

Applying this to our problem, we have that

$$\begin{aligned} f(x) &= \|Ax\|_{\mathbb{R}^m} + \frac{1}{2\lambda}\|x-b\|^2_{\mathbb{R}^n} \\ &= \max_{\|\xi\|_{\mathbb{R}^m}\leq 1} \left(\langle Ax, \xi \rangle_{\mathbb{R}^m} + \frac{1}{2\lambda}\|x-b\|^2_{\mathbb{R}^n} \right) \end{aligned}$$

Then if we denote the previous function of two variables by

$$H(x,\xi) = \langle Ax, \xi \rangle_{\mathbb{R}^m} + \frac{1}{2\lambda}\|x-b\|^2_{\mathbb{R}^n},$$

and the **feasible set** for ξ as $C = \{\xi \in \mathbb{R}^m : \|\xi\|_{\mathbb{R}^m} \leq 1\}$, we have that

$$f(x) = \max_{\xi \in C} H(x,\xi)$$

Since we want to minimize f, we are interested in finding

$$\min_x f(x) = \min_x \max_{\xi \in C} H(x,\xi).$$

We changed our original non-differentiable problem by another one, a min-max problem, which objective is now differentiable with respect to both variables.

A min-max problem

The duality gap is the difference
$DG = \min_{x \in R^n} \max_{\xi \in C} H(x, \xi) - \max_{\xi \in C} \min_{x \in R^n} H(x, \xi)$ which is always ≥ 0.

Let us study the properties of the saddle-point problem:

$$\min_x \max_{\xi \in C} H(x, \xi).$$

depending on the primal variables x and on the dual variables ξ. Is the primal problem equivalent to this saddle-point problem?

• On the one hand $H(x, \xi)$ **is (strictly) convex on** x**, for any fixed** ξ (i.e., $H(\cdot, \xi)$ is a convex function, for any fixed ξ). Why?

It is a quadratic function which Hessian is equal to $\frac{1}{\lambda} I$, a strictly positive definite matrix as $\lambda > 0$.

• On the other hand, **for any fixed** x**,** $H(x, \xi)$ **is a concave function on** ξ (that is, for any fixed x, $H(x, \cdot)$ is a concave function). Why?

It is a linear function on each variable ξ_j, thus concave with respect to $\xi = (\xi_j)_j$.

A min-max problem

This implies that $DG = 0$ and H has a saddle point (x_0, ξ_0). Thus, we can exchange the min with the max, resulting a max-min problem:

$$\min_x \max_{\xi \in C} H(x, \xi) = H(x_0, \xi_0) = \max_{\xi \in C} \min_x H(x, \xi).$$

where (x_0, ξ_0) **is the solution of both problems, which is a saddle point of** H:

$$H(x_0, \xi) \leq H(x_0, \xi_0) \leq H(x, \xi_0)$$

that is, a maximum with respect the dual variable ξ, and a minimum with respect the primal variable x.

Thus, the three problems are equivalent

$$\min_{x\in R^n}\max_{\xi\in C}\left(\langle Ax,\xi\rangle+\frac{1}{2\lambda}\|x-b\|\right)=\max_{\xi\in C}\min_{x\in R^n}\left(\langle Ax,\xi\rangle+\frac{1}{2\lambda}\|x-b\|\right)=\max_{\xi\in C}g_D(\xi)$$

where

$$g_D(\xi)=\mathscr{H}(x_0(\xi),\xi)\qquad\text{and}\qquad x_0(\xi)=\arg\min_{x\in R^n}\mathscr{H}(x,\xi)$$

is the dual function.

Solving a max-min problem

Two approaches for solving the max-min problem:

1. **Dual approach:**
   - Eliminating x by solving first the min part as a function of ξ.
   - Then, solve the max problem of a function that only depends on ξ. (This function is the **dual function**, and ξ is the **dual variable**. By analogy, f is called the **primal function** and x is the **primal variable**).

2. **Primal dual approach:** Solving for x and ξ simultaneously, finding a saddle-point of H.

2. Primal-dual approach: finding a saddle point

A saddle point is a maximum of H with respect to ξ and a minimum with respect to x.
The proposed Algorithm consists in alternating:

- a **gradient descent step** for the variable x, and
- a **gradient ascent step** for the variable ξ.

This idea was introduced by K. Arrow and L. Hurwicz for solving problems in economy.

Since the maximization on ξ is **constrained**, we use a **projected gradient ascent step**.

In our case, the 'partial' gradients of H, $\nabla_x H$ and $\nabla_\xi H$ with respect to x and ξ, respectively, are given by

$$\nabla_x H(x,\xi) = A^t\xi + \frac{1}{\lambda}(x-b),$$
$$\nabla_\xi H(x,\xi) = Ax.$$

Exercise: Compute them using the directional derivative.

Dual update

Starting from an initial (x^0, ξ^0), and given a current iteration (x^k, ξ^k) with $k \geq 0$, we update the dual variable ξ by solving:

$$\max_{\xi \in C} H(x^k, \xi). \tag{5}$$

We do not solve this problem but we just update the variable ξ.

The **gradient ascent** direction is $\nabla_\xi H(x^k, \xi)$ and we evaluate it at $\xi = \xi^k$.

If there were no constraints, the equations for the gradient ascent update of ξ^k would be

$$\frac{\xi^{k+1} - \xi^k}{\tau} = \nabla_\xi H(x^k, \xi^k).$$

The parameter $\tau > 0$ is the time step for the variable ξ.

That is,

$$\xi^{k+1} = \xi^k + \tau \nabla_\xi H(x^k, \xi^k). \tag{6}$$

Dual update

We **enforce the constraint** $\xi \in C$ by projecting our variable onto it.

Let P_C be the projection on the set $C = \{\xi \in \mathbb{R}^m : \|\xi\|_{\mathbb{R}^m} \leq 1\}$.

We replace the previous equation (6) by

$$\xi^{k+1} = P_C(\xi^k + \tau \nabla_\xi H(x^k, \xi^k)). \tag{7}$$

The projection operator on the set C is given by

$$P_C(\xi) = \frac{\xi}{\max(\|\xi\|_{\mathbb{R}^m}, 1)},$$

for any $\xi \in \mathbb{R}^m$. Observe that $P_C(\xi)$ is a vector of $\mathbb{R}^m$ of norm less than 1.

Primal update

Then we proceed to the update the primal variable x^k by solving

$$\min_{x \in \mathbb{R}^n} H(x, \xi^{k+1}). \tag{8}$$

The **gradient descent** direction is $\nabla_x H(x, \xi^{k+1})$ evaluated at $x = x^k$ and the update equation is

$$x^{k+1} = x^k - \theta \nabla_x H(x^k, \xi^{k+1}). \tag{9}$$

The parameter $\theta > 0$ is the time step (or step length) for the variable x.

Summary: A primal-dual algorithm

Given our saddle point problem

$$\min_x \max_{\xi \in C} H(x,\xi) = \min_x \max_{\xi \in C} \langle Ax, \xi \rangle_{\mathbb{R}^m} + \frac{1}{2\lambda} \|x - b\|^2_{\mathbb{R}^n},$$

we find a solution by iterating the following update equations:

$$\xi^{k+1} = P_C(\xi^k + \tau A x^k),$$

$$x^{k+1} = x^k - \theta \left(A^t \xi^{k+1} + \frac{1}{\lambda}(x^k - b) \right),$$

where θ, τ are time steps that have to be specified, and P_C is a projector over C. Given any vector $v \in \mathbb{R}^n$, then

$$P_C(v) = \frac{v}{\max\{1, \|v\|_{\mathbb{R}^m}\}}.$$

In fact this approach can be applied for more general convex functions using the conjugate function to transform the original problem in a saddle point problem.

1. Dual approach for constrained minimization

The dual problem is obtained from the max-min problem by solving first the min part:

$$\min_x H(x,\xi) = \min_x \left(\langle Ax, \xi \rangle_{\mathbb{R}^m} + \frac{1}{2\lambda} \|x - b\|^2_{\mathbb{R}^n} \right).$$

This is an unconstrained minimization problem with a differentiable objective function.

It is in fact, a quadratic function of x which Hessian is a positive definite matrix, thus **convex with respect to** x, and for each ξ there is a unique minimizer $x_0(\xi)$ (this minimizer depends on ξ).

This minimizer is the solution of $\nabla_x H(x,\xi) = 0$, which is

$$x_0(\xi) = b - \lambda A^t \xi.$$

Substituting $x_0(\xi)$ one obtains the **dual function**:

$$g_D(\xi) = \min_x H(x,\xi) = H(x_0(\xi),\xi) = \langle Ab, \xi \rangle_{\mathbb{R}^m} - \frac{\lambda}{2} \|A^t \xi\|^2_{\mathbb{R}^m}.$$

Dual approach for constrained minimization

Once the min part is solved, we only have to solve the max part. This is the **dual problem**:

$$\max_{\xi \in C} g_D(\xi) = \max_{\xi \in C} \left(\langle Ab, \xi \rangle_{\mathbb{R}^m} - \frac{\lambda}{2} \| A^t \xi \|^2_{\mathbb{R}^m} \right).$$

This is a quadratic problem with constraints where we have eliminated the primal variable. Its Hessian is a negative definite matrix, thus H is **concave with respect to** ξ, thus there is a maximizer ξ^0.

We can solve it with a **projected gradient ascent:**

$$\nabla g_D(\xi) = Ab - \lambda AA^t \xi = A(b - \lambda A^t \xi) = AAb - \lambda AA\xi$$

and use the following maximization scheme.
We start from $\xi^0 \in C$ (for instance, $\xi = 0$) and we use a time step (or step lenght) τ.

Then iterate: $\xi^{k+1} = P_C(\xi^k + \tau A x^0(\xi^k))$.

Once we compute the dual optimum ξ^0, we can recover the primal optimum by substituting ξ^0 into $x^0 = x^0(\xi^0)$.

Dual approach for constrained minimization

Remark: The duality gap $(\min_x \max_\xi H(x,\xi) - \max_\xi \min_x H(x,\xi))$ can be used as a stopping criterion for the iterative process.

Indeed, at any (x,ξ)

$$\min_x \max_{\xi\in C} H(x,\xi) - \max_{\xi\in C} \min_x H(x,\xi) \geq 0$$

That is,

$$\min_x f(x) - \max_{\xi\in C} g_D(\xi) \geq 0$$

and at the saddle point $x^*,\xi^*)$

$$f(x^*) - g_D(\xi^*) = 0$$

Thus, at any (x^k,ξ^k), the difference

$$f(x^k) - g_D(\xi^k) \geq 0$$

can be used as a stopping criterion: stop the iterative algorithm when

$$f(x^k) - g_D(\xi^k) < \varepsilon$$

for $\varepsilon > 0$ small.

Summarizing

Under the conditions of the previous duality theory theorem, two algorithms for computing a minimum of the primal problem $\min_x f(x)$, are

(1) Dual algorithm: $\max_{\xi \in C} g_D(\xi)$.

(2) Primal-Dual problem: find a saddle point of $\min_x \max_{\xi \in C} H(x, \xi)$.

Outline

1. Duality: Min-max Theorem.
2. Lagrangian duality (and saddle points).
3. Primal-dual and dual approaches (for some non-differentiable problems).
4. **Applications.**
   - Total Variation restoration
   - disparity computation
   - optical flow estimation
   - minimization of non-local functionals
5. Non-convex problems and convex relaxation.

Application 1: Denoising with the Total Variation

In the context of a discrete image $u = (u_{i,j})_{i=1,j=1}^{M,N}$, with $M \times N$ pixels $(1 \leq i \leq M, 1 \leq j \leq N)$ and its discrete gradient $(\nabla u)_{i,j}$,

let us denote $n = M \times N$, $X = \mathbb{R}^n$, and the discrete gradient operator $A = \nabla : X \to Y = X \times X$, $g \in X$.

In order to denoise an input image g, the goal is to minimize

$$\min_{u \in X} \sum_{i=1}^{M} \sum_{j=1}^{N} \|(\nabla u)_{i,j}\| + \frac{1}{2}\|u - g\|^2,$$

where $g = (g_{i,j})_{i=1,j=1}^{M,N}$ and

$$\|u - g\|^2 = \sum_{i,j=1}^{N} (u_{i,j} - g_{i,j})^2.$$

Application 1: Denoising with the Total Variation

The matrix (operator) ∇^+ is defined by stacking the two matrices ∇_i^+ and ∇_j^+. This matrix computes pixel differences along both i and j direction

$$\nabla^+ u = \begin{bmatrix} \nabla_i^+ \\ \hline \\ \nabla_j^+ \end{bmatrix} \cdot u = \left.\begin{bmatrix} \nabla_i^+ u \\ \hline \\ \nabla_j^+ u \end{bmatrix}\right\} (\text{vector with } 2MN \text{ components}).$$

Indeed: we are multiplying the matrix ∇^+ of size $2MN \times MN$ by the vector u of size $MN \times 1$. Thus, we obtain a vector $\nabla^+ u$ of size $2MN \times 1$

AA: Image restoration (Remember, from JF Garamendi's lecture)

As example we will use this 4x5 image

$$u = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} & u_{15} \\ u_{21} & u_{22} & u_{23} & u_{24} & u_{25} \\ u_{31} & u_{32} & u_{33} & u_{34} & u_{35} \\ u_{41} & u_{42} & u_{43} & u_{44} & u_{45} \end{bmatrix}$$

Then,

Application 1: Denoising with the Total Variation

Firstly, We define the matrix ∇_i^+. This matrix computes pixel differences along i direction (i.e vertical direction)

$$\nabla_i^+ u = \left[\begin{array}{cccc:cccc:cccc:cccc}
-1 & 1 & & & & & & & & & & & & & & \\
 & -1 & 1 & & & & & & & & & & & & & \\
 & & -1 & 1 & & & & & & & & & & & & \\
 & & & 0 & & & & & & & & & & & & \\
\hdashline
 & & & & -1 & 1 & & & & & & & & & & \\
 & & & & & -1 & 1 & & & & & & & & & \\
 & & & & & & -1 & 1 & & & & & & & & \\
 & & & & & & & 0 & & & & & & & & \\
\hdashline
 & & & & & & & & -1 & 1 & & & & & & \\
 & & & & & & & & & -1 & 1 & & & & & \\
 & & & & & & & & & & -1 & 1 & & & & \\
 & & & & & & & & & & & 0 & & & & \\
\hdashline
 & & & & & & & & & & & & -1 & 1 & & \\
 & & & & & & & & & & & & & -1 & 1 & \\
 & & & & & & & & & & & & & & -1 & 1 \\
 & & & & & & & & & & & & & & & 0
\end{array}\right] \cdot \begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \\ \hdashline u_{12} \\ u_{22} \\ u_{32} \\ u_{42} \\ \hdashline u_{13} \\ u_{23} \\ u_{33} \\ u_{43} \\ \hdashline u_{14} \\ u_{24} \\ u_{34} \\ u_{44} \\ \hdashline u_{15} \\ u_{25} \\ u_{35} \\ u_{45} \end{bmatrix} = \begin{bmatrix} u_{21} - u_{11} \\ u_{31} - u_{21} \\ u_{41} - u_{31} \\ 0 \\ \hdashline u_{22} - u_{12} \\ u_{32} - u_{22} \\ u_{42} - u_{32} \\ 0 \\ \hdashline u_{23} - u_{13} \\ u_{33} - u_{23} \\ u_{43} - u_{33} \\ 0 \\ \hdashline u_{24} - u_{14} \\ u_{34} - u_{24} \\ u_{44} - u_{34} \\ 0 \\ \hdashline u_{25} - u_{15} \\ u_{35} - u_{25} \\ u_{45} - u_{35} \\ 0 \end{bmatrix}$$

Application 1: Denoising with the Total Variation

Secondly, We define the matrix ∇_j^+. This matrix computes pixel differences along j direction (i.e horizontall direction)

$$\nabla_j^+ u = \begin{bmatrix}
-1 & 1 & & & & & & & & & & & & & & & \\
 & -1 & 1 & & & & & & & & & & & & & & \\
 & & -1 & 1 & & & & & & & & & & & & & \\
 & & & -1 & 1 & & & & & & & & & & & & \\
\hdashline
 & & & & -1 & 1 & & & & & & & & & & & \\
 & & & & & -1 & 1 & & & & & & & & & & \\
 & & & & & & -1 & 1 & & & & & & & & & \\
 & & & & & & & -1 & 1 & & & & & & & & \\
\hdashline
 & & & & & & & & -1 & 1 & & & & & & & \\
 & & & & & & & & & -1 & 1 & & & & & & \\
 & & & & & & & & & & -1 & 1 & & & & & \\
 & & & & & & & & & & & -1 & 1 & & & & \\
\hdashline
 & & & & & & & & & & & & -1 & 1 & & & \\
 & & & & & & & & & & & & & -1 & 1 & & \\
 & & & & & & & & & & & & & & -1 & 1 & \\
 & & & & & & & & & & & & & & & -1 & 1 \\
\hdashline
 & & & & & & & & & & & & 0 & & & & \\
 & & & & & & & & & & & & & 0 & & & \\
 & & & & & & & & & & & & & & 0 & & \\
 & & & & & & & & & & & & & & & 0 &
\end{bmatrix} \cdot \begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \\ u_{12} \\ u_{22} \\ u_{32} \\ u_{42} \\ u_{13} \\ u_{23} \\ u_{33} \\ u_{43} \\ u_{14} \\ u_{24} \\ u_{34} \\ u_{44} \\ u_{15} \\ u_{25} \\ u_{35} \\ u_{45} \end{bmatrix} = \begin{bmatrix} u_{12} - u_{11} \\ u_{22} - u_{21} \\ u_{32} - u_{31} \\ u_{42} - u_{41} \\ u_{13} - u_{12} \\ u_{23} - u_{22} \\ u_{33} - u_{32} \\ u_{43} - u_{42} \\ u_{14} - u_{13} \\ u_{24} - u_{23} \\ u_{34} - u_{33} \\ u_{44} - u_{43} \\ u_{15} - u_{14} \\ u_{25} - u_{24} \\ u_{35} - u_{34} \\ u_{45} - u_{44} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Application 1: Denoising with the Total Variation

In the context of a discrete image $\{u\}_{i,j}$ and its discrete gradient operator $(\nabla u)_{i,j}$, let $n = N \times N$, $X = \mathbb{R}^n$, and $A = \nabla : X \to Y = X \times X$, $f \in X$.
We can apply previous Algorithm to minimize

$$\min_{u \in X} \sum_{i,j=1}^{N} \|(\nabla^+ u)_{i,j}\|_2 + \frac{1}{2}\|u - f\|_2^2,$$

where $f = (f_{i,j})_{i,j=1}^N$ and

$$\|u - f\|_2^2 = \sum_{i,j=1}^{N} (u_{i,j} - f_{i,j})^2.$$

Which are the ascent and descent equations for this problem?

Extra: Variational approach for the equivalent denoising functional $\mathbf{p=1}$.

$$\min_u J(u) = \min_u \left\{ \frac{1}{2\lambda} \int_\Omega |u-f|^2 dx + \int_\Omega |\nabla u| dx \right\}$$

The TV term is not differentiable. Moreover, $u \in C$ can be non-differentiable (for instance, if u is a piecewise constant image).

A remedy is given by introducing a **dual variable** $\xi \in \mathbb{R}^2$ and use the general fact of norms: $\|y\|_{\mathbb{R}^m} = \max_{\|\xi\|_{\mathbb{R}^m} \leq 1} \langle y, \xi \rangle_{\mathbb{R}^m}$. Then:

$$|\nabla u| = \sup_{|\xi| \leq 1} \langle \nabla u, \xi \rangle$$

where the supremum is attained (if u is differentiable and $\nabla u \neq 0$) at $\xi = \frac{\nabla u}{|\nabla u|}$.

It allows to give an intuition of the following definition of **Total Variation** $TV(u)$ of u in the case of a differentiable u but which generalizes to discontinuous functions u:

$$TV(u) := \sup_{\xi \in C} \int u \operatorname{div} \xi \, dx = (\text{if } u \text{ diff}) = \sup_{\xi \in C} \int \langle \nabla u, \xi \rangle dx = \int |\nabla u| dx$$

with the dual variable ξ being a differentiable vector field with compact support (i.e., $\xi = 0$ at the boundary) constrained to the unit disc at every point $x \in \Omega$:

$$C = \{ \xi \in C_c^1(\Omega, \mathbb{R}^2) : |\xi(x)| \leq 1 \, \forall x \in \Omega \}.$$

Variational approach for the equivalent denoising functional $\mathbf{p = 1}$. Formal computations

$$\begin{aligned}
\min_u J(u) = & \quad \min_u \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx + \int |\nabla u| dx \right\} \\
= & \quad \min_u \sup_{|\xi| \leq 1|} \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx + \int \langle \nabla u, \xi \rangle \, dx \right\} \\
= & \quad \min_u \sup_{|\xi| \leq 1|} \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx - \int u \, \mathrm{div} \xi \, dx \right\}
\end{aligned}$$

In other words, the minimization problem $\min_u J(u)$ has become a

saddle point problem $\min_u \max_{|\xi| \leq 1|} \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx + \int \langle \nabla u, \xi \rangle \, dx \right\}$

$$\min_u J(u) = \min_u \max_{|\xi| \leq 1|} \mathscr{L}(u, \xi).$$

More variables, but easier to solve! (under some conditions, of course).

This is an unconstrained minimization problem with a differentiable objective function. It is in fact, a quadratic function with positive definite Hessian (plus linear) of x, and for each ξ there is a unique minimizer $x_0(\xi)$ (this minimizer depends on ξ). Let's compute it.

We have $\min_u J(u) = \min_u \max_{|\xi|\leq 1|} \mathscr{L}(u,\xi)$.

That is,

$$\min_u \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx + \int |\nabla u| dx \right\} = \min_u \max_{|\vec{\xi}|\leq 1|} \left\{ \frac{1}{2\lambda} \int |u-f|^2 dx + \int \langle \nabla u, \vec{\xi} \rangle \, dx \right\}$$

Exercices:

- Can we can exchange min-max (or inf-sup) by max-min (or sup-inf)? Why?
- Solve the inner min proble with respect to u and verify that

$$u = f - \lambda \, div \vec{\xi}$$

- Write down the dual problem.

Chambolle's Projection Algorithm (extra)

Extra: Chambolle's Projection Algorithm

$$\inf_{u} \sup_{|\vec{\xi}|\leq 1} \left\{ \int_{\Omega} < u, \nabla \cdot \vec{\xi} > dx + \frac{1}{2\lambda} \int_{\Omega} |u-f|^2 dx \right\}$$

Using the min-max theorem (see slides 79- , as the functional is convex in u and concave in ξ, and $C = \{|\xi| \leq 1\}$ is convex), we can change inf-sup by sup-inf and solve, first, with respect to u. In this case, the optimal condition given by Euler-Lagrange technique is:

$$u = f - \lambda \nabla \cdot \vec{\xi}$$

where $\vec{\xi}$ is computed as solution of

$$\boxed{\nabla(\lambda \nabla \cdot \vec{\xi} - f) - \left| \nabla(\lambda \nabla \cdot \vec{\xi} - f) \right| \vec{\xi} = \vec{0}}$$

Numerical resolution by a semi-implicit gradient descent scheme:

$$\frac{\vec{\xi}^{n+1} - \vec{\xi}^{n}}{\tau} = \nabla(\lambda \nabla \cdot \vec{\xi}^{n} - f) - \left| \nabla(\lambda \nabla \cdot \vec{\xi}^{n} - f) \right| \vec{\xi}^{n}$$

$$\vec{\xi}^{n+1} = \frac{\vec{\xi}^{n} + \tau \nabla(\lambda \nabla \cdot \vec{\xi}^{n} - f)}{1 + \left| \nabla(\lambda \nabla \cdot \vec{\xi}^{n} - f) \right|}$$

Some remarks on the Euler-Lagrange equations and the Gradient Descent method of a friend problem (extra)

Extra: Euler-Lagrange equations of a friend problem and some remarks

$$J(u) = \int_\Omega \mathscr{F}(x, u(x), \nabla u(x))\, dx = \int_\Omega \frac{1}{2}|\nabla u|^p + \frac{1}{2\lambda}|u - f|^2 dx$$

with $p = 1$ or $p = 2$.

Or, better, let's simplify and consider

$$J(u) = \int_\Omega \mathscr{F}(x, u(x), \nabla u(x))\, dx = \int_\Omega \frac{1}{2}|\nabla u|^p dx$$

Remember: For a given energy functional

$$J(u) = \int_\Omega \mathscr{F}\left(x, u(x), \nabla u(x)\right) dx$$

(Example: $\mathscr{F}\left(x, u(x), \nabla u(x)\right) = \frac{1}{2}|\nabla u|^2 + \frac{1}{2\lambda}|u - f|^2$)

How does one minimize a given functional w.r.t. the function u?

The necessary condition for extremality of the functional J states that the derivative w.r.t u must be 0.

$$\frac{d\,J}{d\,u} = 0 \qquad \text{extremality principle}$$

Remember the notion of "functional derivative" $\frac{d\,J}{d\,u}$:

The Gâteaux derivative extends the concept of directional derivative (in differential calculus) to infinite-dimensional spaces.

The derivative w.r.t. u of the functional J(u) in the direction $V(x)$ is defined as

$$\mathbf{D_V J(u)} = \frac{d}{d\varepsilon} J(u+\varepsilon V)|_{\varepsilon=0} = \lim_{\varepsilon \to 0} \frac{J(u+\varepsilon V) - J(u)}{\varepsilon} = \langle \nabla_u J(u), V \rangle$$

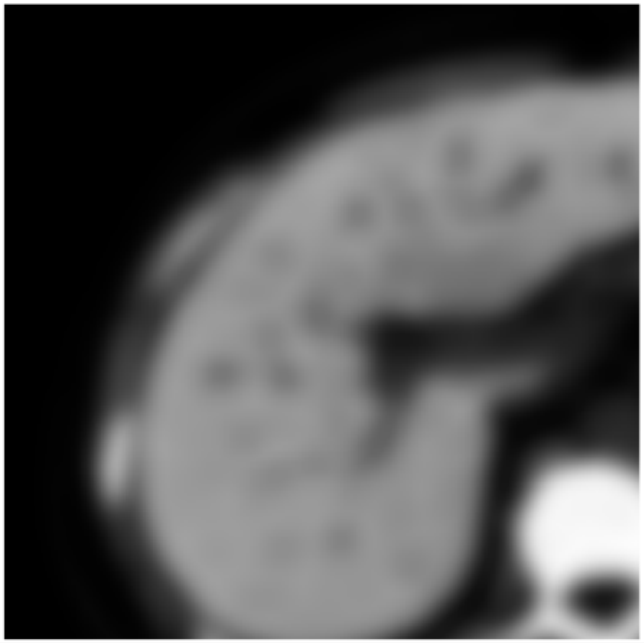
• As in finite dimensions, this directional derivative can be interpreted as the projection of the functional gradient on the respective direction.

• Keep in mind that there are infinite many directions (the value at each point x of the function domain).

Why, depending on p are the solutions so different?

Image denoising problem $\mathscr{F}(u) = \int_\Omega \frac{1}{2}|\nabla u|^p + \frac{1}{2\lambda}|u-f|^2 dx$

p=2



p=1

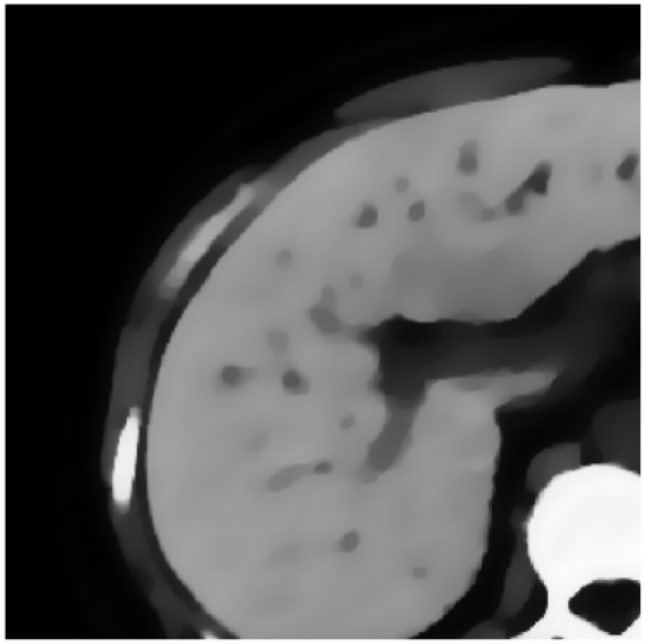


Image denoising problem

We can denoise f by **minimizing** the energy or cost

$$J(u) = \frac{1}{2\lambda} \int_\Omega |u - f|^2 dx + \frac{1}{2} \int_\Omega |\nabla u|^p dx$$

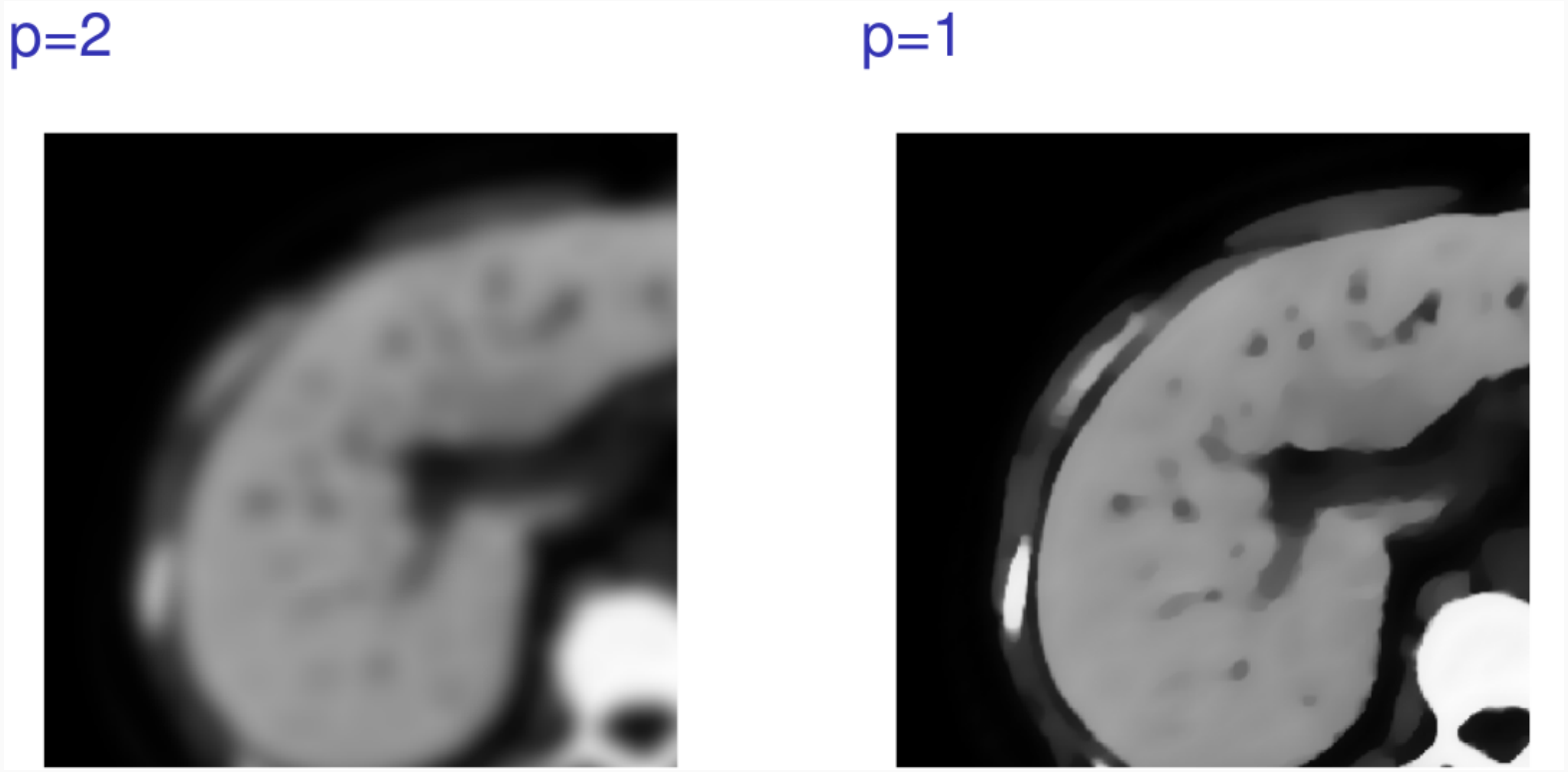
where f is a **known noisy** image and u is the **unknown** image.
$p = 2$ or $= 1$, and λ is a given parameter that controls the trade-off between:

- the **data fidelity term** $J_{\text{data fidelity}}(u) = \int_\Omega |u - f|^2 dx$
  (measures the similarity between u and f)
- the **smoothness term** $J_{\text{regularity}}(u) = \int_\Omega |\nabla u|^p dx$)
  Minimizing this term avoids the irregularities or pixel changes due to high frequency noise.
  (it is a prior (*): asks for a-chosen/the-more-likely degree of regularity)

(*) Common on **inverse problems**: add a (regularity) **prior on the optimum**.

In this example, the power p indicates the prior you expect.

Image denoising problem



Before, some additional comments on the TV term

• In Lecture 2, you did some formal computations in order to formally obtain the Euler-Lagrange equation of the previous $J(u)$ You obtained (for x such that $\nabla u(x) \neq 0$):

$$\frac{1}{\lambda}(u-f)-\Delta u=0\,(p=2) \qquad \frac{1}{\lambda}(u-f)-\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)=0\,(p=1)$$

• If we **forget about the data term and use the gradient descent method**, we obtain

$$(p=2)\begin{cases}\frac{\partial u}{\partial t}(t,x)=\Delta u\\ u(0,x)=u_0(x) \quad x\in\Omega\end{cases} \qquad (p=1)\begin{cases}\frac{\partial u}{\partial t}(t,x)=\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)\\ u(0,x)=u_0(x) \quad x\in\Omega\end{cases}$$

The right hand PDE $(p=1)$ is the **popular steepest descent method to minimize the total variation introduced by L. Rudin and S. Osher in 1994**.

• **What is this equation?** **TV diffusion**. In other words, regularizing or filtering the initial datum u_0. It is very popular (among the convex regularisers).

• This filtering process has less destrictive effect on the edges of u_0 than filtering with a Gaussian, that is, than solving the heat equation $\frac{\partial u}{\partial t}=\Delta u$ with initial condition u_0. **Why?**

Before, some additional comments on the TV term

$$\frac{\partial u}{\partial t}(t,x) = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)$$
$$u(0,x) = u_0(x) \qquad x \in \Omega$$

• **Why** this filtering process has less destrictive effect on the edges of u_0 than filtering with a Gaussian, that is, than solving the heat equation $\frac{\partial u}{\partial t} = \Delta u$ with initial condition u_0 **?**

• **Intuition:** Write the previous equation as

$$\frac{\partial u}{\partial t} = \operatorname{div}(g\nabla u)$$

where the "weight" g is $g(x) = \frac{1}{|\nabla u|}$. It produces an **anisotropic diffusion** such that, on points x on the edges of u_0, the modulus of the gradient is hight and thus there is no (or little) diffussion. Therefore, **the edges of u_0 are better preserved** when filtering it.

• **Thus, it is crucial to have at our disposal correct algorithms to solve the minimization of functionals including the Total Variation term!**

And we have them!! WHICH ONES?

Application 2: Optical flow estimation (extra)

Example: Two frames of a video

Frame 1

Optical flow estimation

Example: Two frames of a video

Frame 2

Optical flow estimation

Example: Optical flow between two frames of a video

Optical flow

Optical flow estimation (M6)

• The computation of the **motion field or optical flow between two consecutive frames of a video sequence** is one of the fundamental problems in image processing and computer vision.

• Let us consider a video $I : \Omega \times \mathbb{T} \to \mathbb{R}$, where $\Omega \subset \mathbb{R}^2$ is the discrete rectangular domain of each frame, and $\mathbb{T} = \{0, 1, \ldots, T\}$ represents **time**. Thus, $I(\mathbf{x}, t) \in \mathbb{R}$ is the value of the video at time $t \in \mathbb{T}$ in position $\mathbf{x} = (x, y) \in \Omega$.

• The forward **optical flow** is a vector field $\mathbf{v} : \Omega \times \mathbb{T} \to \mathbb{R}^2$, which **captures the displacement of pixels from one frame to the following**:

pixel $\mathbf{x}$ at frame t "moves" to $\mathbf{x} + \mathbf{v}(\mathbf{x}, t)$ at frame $t + 1$.

Optical flow estimation

- Let's consider only two frames $I(\mathbf{x}, t)$ and $I(\mathbf{x}, t+1)$.

- Let $\mathbf{v} = (v_1, v_2) : \Omega \to \mathbb{R}^2$ be the optical flow between them.
Pixel $\mathbf{x} = (x, y)$ at frame t "moves" to $(x, y) + \mathbf{v}(\mathbf{x}, t)$ at frame $t+1$.

- Most methods for computing the optical flow are based on the basic assumption that the gray level of objects remains constant along the video. In particular, **the gray value of a moving pixel stays constant over time**:

$$I(x, y, t) = I((x, y) + \mathbf{v}(\mathbf{x}, t), t+1).$$

This is the so-called **brightness (gray level) constancy assumption**, or **optical flow constraint**.



Optical flow estimation

First Attempt **Basic assumption:** Gray level constancy

$$I(x,y,t) = I((x,y)+(v_1,v_2),t+1). \tag{10}$$

Problem: Find $\mathbf{v} = (v_1, v_2)$ satisfying (10) for all $(x,y) \in \Omega$.

*$I(\mathbf{x}+\mathbf{v}, t+1)$ is often linearized using a 1st order Taylor approximation.

The optical flow constraint has an inherent problem: it is under-determined. Indeed, it yields only one constraint to solve for two variables v_1, v_2: infinite number of solutions.

On the other hand, there is noise and illumination changes. Therefore (10) **cannot be exactly satisfied**.

Optical flow estimation

Second Attempt Relax the gray level constancy assumption allowing *small perturbations* due to noise or illumination changes.

Problem: Find $\mathbf{v} = (v_1, v_2)$ satisfying

$$\min_{(v_1,v_2)} \int_{\Omega} (I(x+v_1, y+v_2, t+1) - I(x,y,t))^2 \tag{11}$$

But, again, this problem has infinite minima.

Moreover, they may be chaotic (not real!).

Optical flow estimation

Third Attempt If a point (x, y) is following a motion given by (v_1, v_2) and a point $(\tilde{x}, \tilde{y})$ is following a motion given by $(\tilde{v}_1, \tilde{v}_2)$ and $(\tilde{x}, \tilde{y})$ is *near* (x, y), then probably $(\tilde{v}_1, \tilde{v}_2)$ is *similar* to (v_1, v_2).

In other words, $|\nabla v_1|$ and $|\nabla v_2|$ have to be **small**.

Problem: Find $\mathbf{v} = (v_1, v_2)$ satisfying

$$\min_{(v_1,v_2)} \int_\Omega \left(|\nabla v_1|^2 + |\nabla v_2|^2 \right) + \lambda \int_\Omega \left(I(x + v_1, y + v_2, t + 1) - I(x, y, t) \right)^2$$

This (or its variant

$\min_{(v^1,v^2)} \int_\Omega \left(|\nabla v_1|^2 + |\nabla v_2|^2 \right) + \lambda \int_\Omega \left(I(x + v_1, y + v_2, t + 1) - I(x, y, t) \right)^2$)

is the so-called **Horn-Schunk model** for optical flow estimation.

Remark that, again, our estimated optical flow will be the minimum of an energy of the form

$$J(v) = J_S(v) + \lambda J_D(v).$$

TV-L^1 Optical flow model

TV$-\mathrm{L}^1$ optical flow functional for optical flow estimation in video:

$$\min_{\mathbf{v}} \lambda \int_{\Omega} |I(\mathbf{x}+\mathbf{v}, t+1) - I(\mathbf{x}, t)| d\mathbf{x} + \int_{\Omega} |\nabla \mathbf{v}| d\mathbf{x}$$

where $\mathbf{v}$ is the (unknown) optical flow, that is, the vector field that recovers the apparent motion of two consecutive frames of the input video.

Optical Flow

An Improved Algorithm for TV-L^1 Optical Flow

Andreas Wedel[1,3], Thomas Pock[2], Christopher Zach[4], Horst Bischof[2], and Daniel Cremers[1]

[1] Computer Vision Group, University of Bonn
[2] Institute for Computer Graphics and Vision, TU Graz
[3] Daimler Group Research and Advanced Engineering, Sindelfingen
[4] Department of Computer Science, University of North Carolina at Chapel Hill

In their seminal work [18], Horn and Schunck studied a variational formulation of the optical flow problem.

$$\min_{\boldsymbol{u}} \left\{ \int_{\Omega} |\nabla u_1|^2 + |\nabla u_2|^2 \, \mathrm{d}\Omega + \lambda \int_{\Omega} \left(I_1(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) - I_0(\boldsymbol{x}) \right)^2 \mathrm{d}\Omega \right\} . \quad (1)$$

Here, I_0 and I_1 is the image pair, $\boldsymbol{u} = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}))^T$ is the two-dimensional displacement field and λ is a free parameter. The first term (regularization term) penalizes high variations in $\boldsymbol{u}$ to obtain smooth displacement fields. The second term (data term) is also known as the optical flow constraint. It assumes, that the intensity values of $I_0(\boldsymbol{x})$

Optical Flow

In the basic setting two image frames I_0 and $I_1 : (\Omega \subseteq \mathbb{R}^2) \rightarrow \mathbb{R}$ are given. The objective is to find the disparity map $\boldsymbol{u} : \Omega \rightarrow \mathbb{R}^2$, which minimizes an image-based error criterion together with a regularization force. In this work we focus on the plain intensity difference between pixels as the image similarity score. Hence, the target disparity map $\boldsymbol{u}$ is the minimizer of

$$\int_{\Omega} \left\{ \lambda \phi \left(I_0(\boldsymbol{x}) - I_1(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) \right) + \psi(\boldsymbol{u}, \nabla \boldsymbol{u}, \ldots) \right\} d\boldsymbol{x}, \tag{2}$$

where $\phi \left(I_0(\boldsymbol{x}) - I_1(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})) \right)$ is the image data fidelity, and $\psi(\boldsymbol{u}, \nabla \boldsymbol{u}, \ldots)$ depicts the regularization term. The parameter λ weighs between the data fidelity and the regularization force. Selecting $\phi(x) = x^2$ and $\psi(\nabla \boldsymbol{u}) = |\nabla \boldsymbol{u}|^2$ results in the Horn-Schunck model [18].

The choice of $\phi(x) = |x|$ and $\psi(\nabla \boldsymbol{u}) = |\nabla \boldsymbol{u}|$ yields to the following functional consisting of an L^1 data penalty term and total variation regularization:

$$E = \int_{\Omega} \left\{ \lambda |I_0(\boldsymbol{x}) - I_1(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x}))| + |\nabla \boldsymbol{u}| \right\} d\boldsymbol{x}. \tag{3}$$

Optical Flow (extra)

Some drawbacks to both previous models:

- Can you find the corresponding point of a point x at time t that will be occluded in time $t+1$?
- On the other hand, a point that is disoccluded at time $t+1$ has not correspondig point at time t.

That is, mainly occlusion and disocclusion regions.

Optical Flow

A TV-L1 Optical Flow Method with Occlusion Detection

Coloma Ballester[1], Lluis Garrido[2], Vanel Lazcano[1], and Vicent Caselles[1]

[1] Dept Information and Communication Technologies, University Pompeu Fabra
[2] Dept Applied Mathematics and Analysis, University Barcelona
{coloma.ballester,vanel.lazcano,vicent.caselles}@upf.edu,
lluis.garrido@ub.edu

optical flow and occlusions. Let $\chi : \Omega \to [0,1]$ be the function modeling the occlusion mask, so that $\chi = 1$ identifies the occluded pixels, i.e. pixels that are visible in I_0 but not in I_1. Our model is based on the assumption that pixels that are not visible in frame I_1 are visible in the previous frame of I_0. Let $I_{-1} : \Omega \to \mathbb{R}$ be that frame. Thus, if $\chi(\mathbf{x}) = 0$, then we compare $I_0(\mathbf{x})$ and $I_1(\mathbf{x} + \mathbf{u}(\mathbf{x}))$. If $\chi(\mathbf{x}) = 1$, we compare $I_0(\mathbf{x})$ and $I_{-1}(\mathbf{x} - \mathbf{u}(\mathbf{x}))$. On the other hand, the occluded region given by $\chi = 1$ should be correlated with the region where $\mathrm{div}(\mathbf{u})$ is negative. Thus we propose to compute the optical flow by minimizing the energy

$$E(\mathbf{u}, \chi) = E_d(\mathbf{u}, \chi) + E_r(\mathbf{u}, \chi) + \frac{\alpha}{2} \int_\Omega \chi |\mathbf{u}|^2 \, d\mathbf{x} + \beta \int_\Omega \chi \mathrm{div}(\mathbf{u}) \, d\mathbf{x}, \qquad (2)$$

Optical Flow (extra)

where

$$E_d(\mathbf{u},\chi) = \lambda \int_\Omega ((1-\chi)|I_0(\mathbf{x})-I_1(\mathbf{x}+\mathbf{u}(\mathbf{x}))|+\chi|I_0(\mathbf{x})-I_{-1}(\mathbf{x}-\mathbf{u}(\mathbf{x}))|)\,d\mathbf{x},$$

$$E_r(\mathbf{u},\chi) = \int_\Omega g(\mathbf{x})(|\nabla u_1| + |\nabla u_2| + |\nabla \chi|)\,d\mathbf{x},$$

As in [19], in order to cope with the nonlinearities of both $E_d(\mathbf{u},\chi)$ and $E_r(\mathbf{u},\chi)$ we introduce an auxiliary variable $\mathbf{v}$ representing the optical flow and we penalize its deviation form $\mathbf{u}$. Thus, we minimize the energy

$$E_\theta = E_d(\mathbf{v},\chi)+E_r(\mathbf{u},\chi)+\frac{\alpha}{2}\int_\Omega \chi|\mathbf{v}|^2\,d\mathbf{x}+\beta\int_\Omega \chi\mathrm{div}(\mathbf{u})\,d\mathbf{x}+\frac{1}{2\theta}\int_\Omega |\mathbf{u}-\mathbf{v}|^2, \quad (5)$$

depending on the three variables $(\mathbf{u},\mathbf{v},\chi)$, where $\theta > 0$. This energy can be minimized by alternatively fixing two variables and minimizing with respect to the third one.

Optical Flow

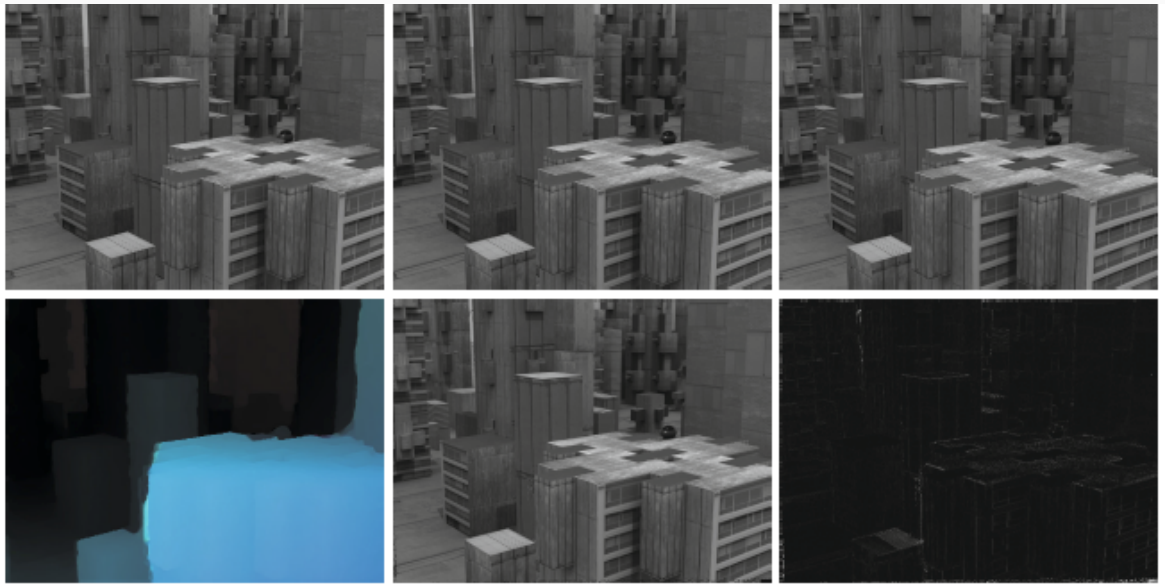


Fig. 1. *First row*: three consecutive frames I_{-1}, I_0, I_1 of the Urban2 Middlebury video sequence. *Second row*: the optical flow $\mathbf{u}$, the backwards motion compensated image using I_1 and I_{-1}, and the normalized absolute value of the difference between the motion compensated image and the original image in the *first row*.

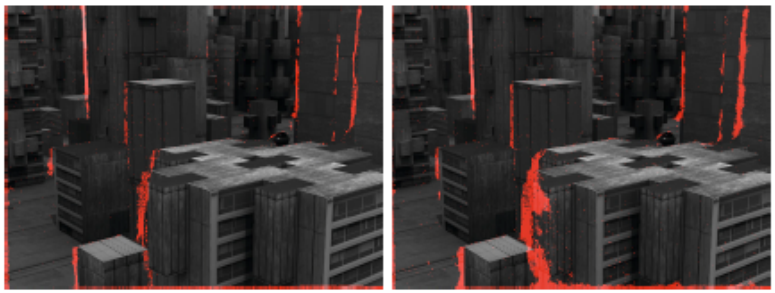


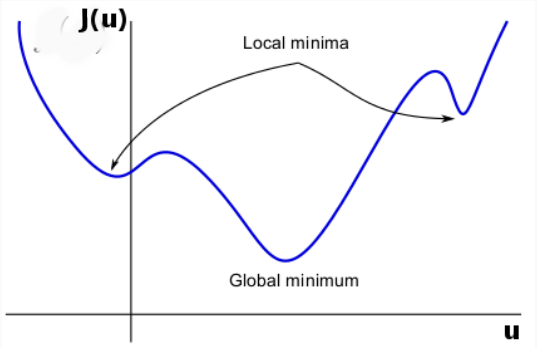
Fig. 2. *Left*: occlusion layer χ (*red*) associated to $\alpha > 0$ superimposed on I_0. *Right*: occlusion layer χ associated to $\alpha = 0$ superimposed on I_0

Outline

1. Duality: Min-max Theorem.
2. Lagrangian duality (and saddle points).
3. Primal-dual and dual approaches (for some non-differentiable problems).
4. Applications.
5. **Non-convex problems and convex relaxation.**
   - Why we need them? More on motivating. Problem geometry
   - Several possibilities.

(4.) Convex Relaxation Methods (extra)

If you have a cost function that it is not convex



then a gradient descent approach will (probably) only lead you to a local minimum, not to a global minimum of your energy.

• Most of the well-known and useful energy functionals are non-convex (for instance, the TV-L1 functional for optical flow estimation).

• This is a big drawback and then the challenge is to find global minima or, at least, a "good" local-minima.

• The **idea** is to convexify non-convex formulations: replace the original non-convex formulation by a convex one.

• Typically, one looses global optimality with respect the original energy but, it turns out that this is not always the case. There are methods and examples (we will see some of them) where you have global optimal solutions of both formulations.

(4.) Convex Relaxation Methods (extra)

• For **non-convex energies**, the corresponding Gradient descent method (for instance) only provides **local optima**.

• While the computed solutions are often good, we generally do not have a performance guarantee, i.e., we do not know how far we are from the optimal solution.

• Some variational approaches have been proposed which are aimed at approximating the original energies with **convex functionals**.

Rather than minimizing the original energy locally, they minimize an approximation of the original energy globally.

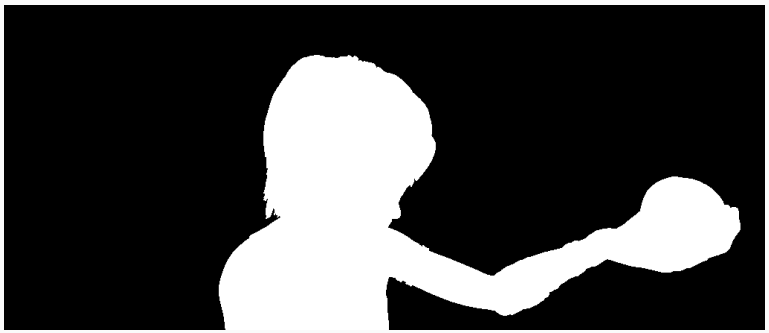
How far this framework can be extended to the kinds of energies arising in computer vision is among the major challenges in current research.

(4.) Convex Relaxation Methods (extra)

Relaxation often denotes the technique of simply dropping certain constraints from the overall optimization problems. **Convex relaxation** means that upon relaxation the problem becomes convex.

For instance, if you are minimizing over a set of binary functions

Example: **Disocclusion of moving objects or shapes:**



(4.) Convex Relaxation Methods (extra)

SPATIO-TEMPORAL BINARY VIDEO INPAINTING VIA THRESHOLD DYNAMICS

M. Oliver *R.P. Palomares* *C. Ballester* *G. Haro**

DTIC – Universitat Pompeu Fabra

ABSTRACT

We propose a new variational method for the completion of moving shapes through binary video inpainting that works by smoothly recovering the objects into an inpainting hole. We solve it by a simple dynamic shape analysis algorithm based on threshold dynamics. The model takes into account the optical flow and motion occlusions. The resulting inpainting algorithm diffuses the available information along the space and the visible trajectories of the pixels in time. We show its performance with examples from the Sintel dataset, which contains complex object motion and occlusions.

Index Terms— Shape completion, binary video inpainting, threshold dynamics.

1. INTRODUCTION

Video inpainting stands for the completion of missing, damaged or occluded information in a video sequence in such a way that this restoration is as unnoticeable (visually plausible) as possible. The applications include tools for cinema post-production to remove, [illegible] ensuring spatio-temporal consistency in other video editing applications [6, 5, 17, 31].

Trajectories and the convective derivative are defined by the optical flow (the vector field that recovers the apparent motion of two consecutive frames) which is previously estimated and completed (inside the inpainting mask or hole) also through variational methods. and we present qualitative and quantitative results showing that our binary video inpainting method obtains similar results using as input either the ground truth optical flow or an estimated one. On the other hand, the optical flow is unknown inside the hole and it is interpolated with a motion inpainting method.

One of the main difficulties that has to be tackled in video completion is due to occlusion effects. Object occlusions and disocclusions generate artifacts which are specially visible at moving occlusion boundaries. Moreover, optical flow methods may fail in occlusion areas due to unreliable shape or point matching. In general, points visible at time t that are occluded at time $t+1$ should not have a corresponding point at frame $t+1$. Thus video completion algorithms have to detect such occlusions in order to correctly decide how to interpolate. Our method keeps track of the motion occlusions, which are estimated from the optical flow, and incorporates

In: Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on, pp. 1822-1826. 2017.

(4.) Convex Relaxation Methods (extra)

In order to inpaint a binary video (defined on a spatio-temporal domain $\mathscr{V}$) inside a spatio-temporal inpainting mask $\mathscr{M} \subset \mathscr{V}$, we proposed to solve the following optimization problem

$$\min_{u:\mathscr{V}\to\{0,1\}} \int_{\mathscr{M}} \|\mathscr{L}(u)\|^2, \quad \text{s.t. } u = u_0 \text{ in } \mathscr{V} \setminus \mathscr{M}$$

where $\mathscr{L}(u)$ is the following differential operator defined taking into account both spatial and temporal regularity as well as the occlusion areas produced by the motion of objects in the scene:

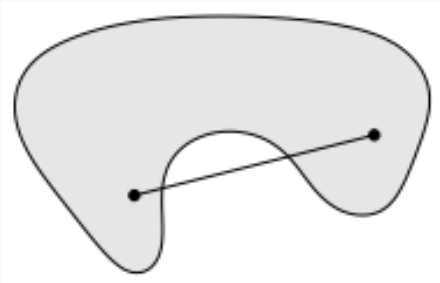
$$\mathscr{L}(u) = (u_x, u_y, \gamma\chi\partial_{\mathbf{v}} u)\,.$$

Here, $\partial_{\mathbf{v}} u$ denotes the convective derivative, $\gamma > 0$ is a parameter and $\chi : \mathscr{V} \to [0,1]$ is a function modelling the occlusion areas so that $\chi(\mathbf{x}, t) = 0$ identifies the occluded pixels, i.e. pixels that are visible at time t but not at time $t+1$.

(4.) Convex Relaxation Methods (extra)

In this context, the goal is to minimize over a set of binary functions $u \in \mathbf{BV}(\mathscr{V}; \{\mathbf{0}, \mathbf{1}\})$, the **space of functions of bounded variation**, i.e., functions u for which the **total variation**, **TV(u)**" $= \int_{\mathscr{V}} |\nabla u| dx$" (the definition will be precised afterwards), is finite.

But the set of binary functions is not convex. Convex combinations of binary functions are tipically no longer binary.



Thus, we convexify by simply dropping the constraint that u must be binary. This is quite popular: to allow u to take on values in the entire interval $[0, 1]$, which is the **convex hull** of the original domain

$$u \in \mathbf{BV}(\mathscr{V}; [\mathbf{0}, \mathbf{1}])$$

By construction, this is a **convex optimization problem**, and a global optimum can be computed.

(4.) Convex Relaxation Methods (extra)