# Optimisation for computer vision and data science

Lecture notes

Tuomo Valkonen
tuomov@iki.fi

Last updated May 16, 2017

# Contents

# 1 Introduction

We recall from basic optimisation courses and textbooks (e.g., [1]), that if $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable, and $\hat{x}$ is a minimiser of $f$,

$$f(\hat{x}) = \min_{x \in \mathbb{R}^n} f(x), \tag{1.1}$$

then

$$\nabla f(x) = 0. \tag{1.2}$$

If $f$ is convex, the condition (1.2) is even sufficient to ensure (1.1). But what if $f$ is non-smooth, such as when

$$f(x) = |x|, \quad (x \in \mathbb{R})?$$

It is clear that $\hat{x} = 0$ is a minimiser of this function, but at the same time $\nabla f(0)$ does not exist. In Chapter 2, we will look at ways to define a set-valued **subgradient** $\partial f(0)$, which satisfies $0 \in \partial f(0)$. In the present chapter, we look at examples that demonstrate why differentiation of non-smooth functions is important.

We begin with a very simple example problem that, while not useful by itself, forms a part of many algorithms, and sheds light on how our later examples also behave.

Example 1.1 (Soft thresholding). Consider the simple problem

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|z - x\|_2^2 + \lambda\|x\|_2,$$

where the parameter $\lambda > 0$, data $z \in \mathbb{R}^n$, and the term $\|x\|_2$ is non-smooth. As we will later learn how to derive, the solution is

$$\hat{x} = \begin{cases} 0, & \|z\|_2 \leq \lambda, \\ z(1 - \lambda/\|z\|_2), & \|z\|_2 > \lambda. \end{cases}$$

Thus the minimisation procedure can be used to remove noise from $z$: anything below amplitude $\lambda$ is considered noise.

## 1.1 Applications in image processing

Non-smooth optimisation problems can be found in various fields. An important application area is image processing. We consider images as $n_1 \times n_2$ pixels grids, mapping these grids for simplicity of overall treatment into vectors of length $n_1 n_2$. Thus the pixel at the two-dimensional index $(i, j)$ is the element $u_{i+n_1(j-1)}$ of the vector $u$, as illustrated in Figure 1.1. Here $i \in \{1, \ldots, n_1\}$ and $j \in \{1, \ldots, n_2\}$.

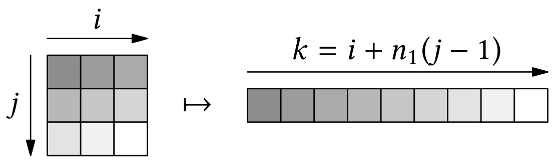
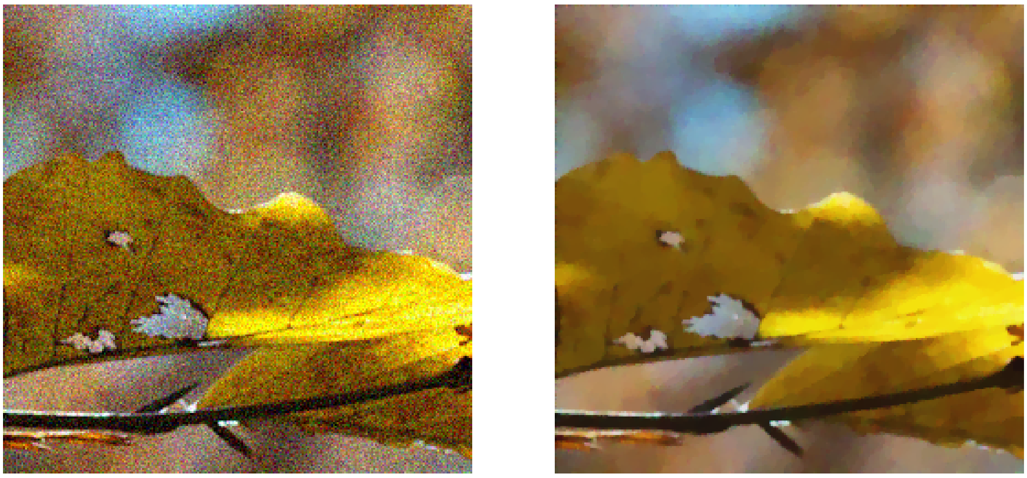


**Figure 1.1:** Mapping of an $n_1 \times n_2$ pixel grid into a vector of length $n_1 n_2$.



**(a)** Noisy image

**(b)** Denoised image

**Figure 1.2:** Demonstration of image denoising with total variation regularisation (1.3). Note how the leaf edges are preserved by the denoising procedure. This is an important feature of total variation type approaches.

The most prototypical image processing problem is **denoising**. A seminal approach to denoising is the total variation (TV) regularisation

$$\min_{x \in \mathbb{R}^{n_1 n_2}} \frac{1}{2}\|z - x\|^2 + \alpha\|\widetilde{D}x\|_{2,1}. \tag{1.3}$$

The first term in (1.3), the **fidelity term**, measures the distance of our solution $x$ to the noisy image $z$. The second **regularisation term** tells us that the solution should be pretty. The **regularisation parameter** $\alpha > 0$ balances between these two goals.

The matrix $\widetilde{D}$ transforms the image in such a way that the unwanted image features are penalised. In TV regularisation, we in particular take

$$\widetilde{D} = \begin{pmatrix} \widetilde{D}_x \\ \widetilde{D}_y \end{pmatrix} \in \mathbb{R}^{2n_1 n_2 \times n_1 n_2} \tag{1.4a}$$

as a discrete approximation of the image gradient. A common choice is forward differences with Neumann boundary conditions (roughly meaning zero gradient on the boundary). This may be written

$$[\widetilde{D}_x x]_{i+n_1(j-1)} = \begin{cases} x_{i+1+n_1(j-1)} - x_{i+n_1(j-1)}, & 1 \le i < n_1,\ 1 \le j \le n_2 \\ 0, & i = n_1,\ 1 \le j \le n_2 \end{cases}$$

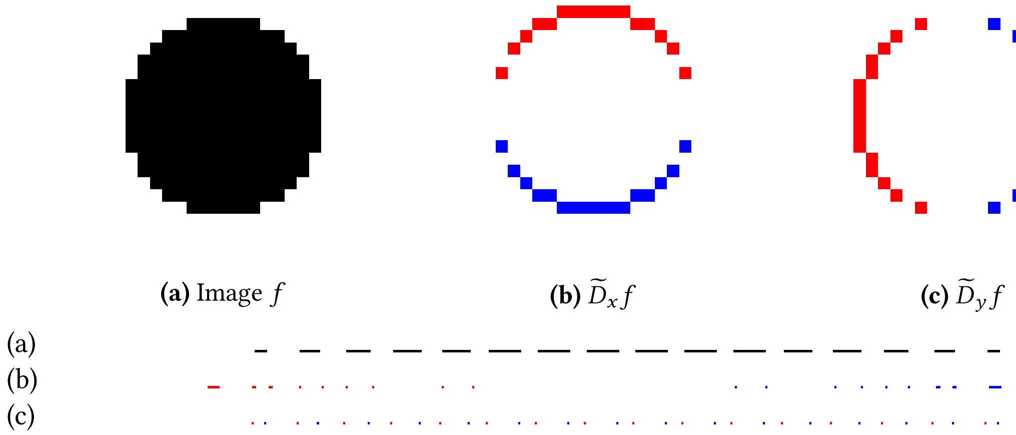
$$[\widetilde{D}_y x]_{i+n_1(j-1)} = \begin{cases} x_{i+n_1 j} - x_{i+n_1(j-1)}, & 1 \le i \le n_1,\ 1 \le j < n_2 \\ 0, & 1 \le i \le n_2,\ j = n_2. \end{cases}$$

The matrix $\widetilde{D}_x$ calculates the difference in all neighbouring pixel intensities in the $x$-direction, and $\widetilde{D}_y$ in the $y$-direction, with zero-difference extension over the image boundary. This is illustrated in Figure 1.3.

One alternative to TV-regularisation would be to replace $\widetilde{D}$ in (1.3) by a Wavelet transformation $W$. This has its own advantages and disadvantages.

We also use the 1-2 combination norm

$$\|g\|_{2,1} := \sum_{k=1}^{n_1 n_2} \sqrt{g_k^2 + g_{n_1 n_2 + k}^2},$$



**(a)** Image $f$

**(b)** $\widetilde{D}_x f$

**(c)** $\widetilde{D}_y f$

**(d)** The images (a)–(c) ordered as vectors, cf. Figure 1.1.

**Figure 1.3:** Illustration of the discrete gradient $\widetilde{D}$ on a $24 \times 24$ pixel geometric object. In the source image $f$, the values are black=1, white=0. In the gradient images, red=+1, blue=-1, and white=0. In (a)–(c) the images are displayed with the natural two-dimensional ordering of the pixels, while in (d) we plot them in the vectorised order illustrated in Figure 1.1, which is how our constructed $\widetilde{D}$ matrix expects the images.

where we take the image-wide 1-norm over the field of 2-norms of the pixelwise gradient approximations. Observe—just try to differentiate!—that this norm is non-smooth: it does not have a conventional gradient if $g_k^2 + g_{n_1 n_2 + k}^2 = 0$. If we replaced $\|\widetilde{D}x\|_{2,1}$ by the squared norm $\|\widetilde{D}x\|_{2,1}^2$, we could make the problem smooth. However, the special properties of the image-wide one-norm are important for edge preservation in image processing.

Observe now how the problem (1.3) generalises Example 1.1. In the end, TV-regularisation penalises non-zero image gradients—in fact simillarly to Example 1.1, it tries to remove any small image gradients, and prefers all non-zero gradients to be concentrated on a small number of pixels. It therefore prefers images with large flat-coloured areas.

## 1.2 Regularisation of inverse problems

Various other image processing problems besides denoising can be constructed by replacing the first term in (1.3) by one involving a matrix $T \in \mathbb{R}^{m \times n_1 n_2}$. That is, we consider the problem

$$\min_{x \in \mathbb{R}^{n_1 n_2}} \frac{1}{2}\|z - Tx\|^2 + \alpha\|\widetilde{D}x\|_{2,1}. \tag{1.5}$$

The first term models the operator equation

$$Tx + v = z,$$

for our known data $z$, noise $v$, and unknown image $x$. Trying to solve this equation for $x$ is an **inverse problem**. In general, such problems are ill-posed, and we cannot expect to have a unique solution, or a solution at all. In order to impose well-posedness, we introduce a regulariser $R$ that models our prior assumptions on a good solution $u$, as well as a fidelity functional $F$ that models the noise $v$. The choice of $R$ is specific to the problem at hand; a prototypical choice in image processing is the **total variation** $R(x) = \|\widetilde{D}x\|_{2,1}$ that we already have seen. More recent research has focused on higher-order [2] and curvature-based [3] extensions, as well as non-convex regularisers [4].

If we know a noise level $\sigma$, we may then try to solve the problem

$$\min_x R(x) \quad \text{subject to} \quad F(Tx - z) \leq \sigma. \tag{1.6}$$

Often the noise level is not known. Moreover, (1.6) can be numerically very difficult. It is therefore more common to solve the *Tikhonov regularised* problem

$$\min_u F(Tx - z) + \alpha R(x), \tag{1.7}$$

for a suitable **regularisation parameter** $\alpha$. Clearly, our image reconstruction problem (1.5) is an instance of (1.7). We refer to [5] the student interested in reading on more about inverse problems theory, and the role $\alpha$ and $\sigma$ play especially in their limit.

As we have seen, in image processing, for **denoising** $T = I$ is the identity. For **deblurring**, $T$ can be a convolution operation, $Tx = \rho * x$ for a suitable blur or convolution kernel $\rho$. For sub-sampled reconstruction from Fourier samples, as is the case with magnetic resonance imaging (MRI) reconstructions, $T = S\mathcal{F}$ for $S \in \{0,1\}^{k \times n_1 n_2}$ a sub-sampling matrix ($k \ll n_1 n_2$, and every row of $S$ sums to 1), and $\mathcal{F} \in \mathbb{C}^{n_1 n_2 \times n_1 n_2}$ the discrete Fourier transform (DFT) matrix. In two dimensions, this can be written

$$[\mathcal{F}x]_{k+n_1(\ell-1)} = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} e^{-2\pi \mathbb{i}(ki+\ell j)} x_{i+n_1(j-1)}, \quad (k = 1, \ldots, n_1;\ \ell = 1, \ldots, n_2).$$

Here the complex imaginary unit $\mathbb{i} = \sqrt{-1}$.

If simply $T = S$ for a sub-sampling matrix, then we are talking about *inpainting*. This might be used, for example, to hide hairs and scratches in old photographs or films. For a detailed treatment of various image processing tasks, see, for example [6, 7].

## 1.3 Applications in data science

Problems of similar structure as (1.7) can be found in statistics and machine learning. Various problems therein can be formulated as instances of **empirical risk minimisation**

$$\min_{x \in \mathbb{R}^m} g(x) + \frac{1}{n} \sum_{i=1}^{n} \phi_i(a_i^T x) \tag{1.8}$$

where $a_i^T x$ is a **linear predictor**, $\phi_i$ a convex **loss function**, and $g$ again a **regulariser**. Here we briefly consider a few examples, and refer to [8] a more in-depth look.[1]

Example 1.2 (Support vector machines). If $a_i \in \mathbb{R}^m$ for $i = 1, \ldots, n$ is a feature vector associated to a label $b_i = \pm 1$, and we set $\phi_i(z) = \max\{0, 1 - b_i z\}$ to be the **hinge loss**, and $g(x) = \frac{\lambda}{2}\|x\|_2^2$ for a parameter $\lambda > 0$, then (1.8) becomes a linear support vector machine (SVM)

$$\min_{x \in \mathbb{R}^m} \frac{\lambda}{2}\|x\|_2^2 + \sum_{i=1}^{n} \max\{0, 1 - b_i a_i^T x\}. \tag{1.9}$$

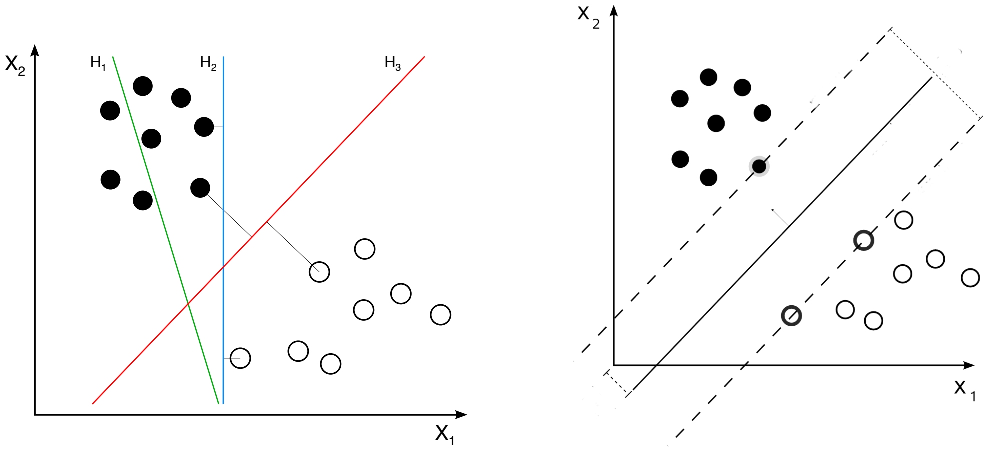
The variable $b_i = \pm 1$ is known as the **label** or **class** of the data $a_i$. The solution $x$ determines a linear classifier as

$$\text{class}_x(a) := \begin{cases} 1, & a^T x > 0, \\ -1, & a^T x < 0, \\ \text{undetermined}, & a^T x = 0. \end{cases}$$

In other words, and as illustrated in Figure 1.4a, $x$ determines the hyperplane that separates the two classes,

$$H_x := \{y \in \mathbb{R}^m \mid x^T y = 0\}.$$

[1] See also http://www.cs.cornell.edu/courses/cs4780/2015fa/web/lecturenotes/lecturenote10.html for a list of several different models.



**(a)** The hyperplane $H_1$ does not separate the two classes. $H_2$ does, but only with a small margin. The hyperplane $H_3$ separates them with the optimum margin.

**(b)** The margin of the SVM is the distance $2/\|x\|$ between the dashed lines. The data vectors touching the margin are the **support vectors**.

**Figure 1.4:** Illustrations of linear support vector machines.

*(a) is due to user ZackWeinberg on Wikipedia, licensed under Creative Commons BY-SA-3.0. It can be found at* `https://commons.wikimedia.org/wiki/File:Svm_separating_hyperplanes_(SVG).svg`.

*(b) is based on an image due to user Peter Buch on Wikipedia, and in the public domain. The original can be found at* `https://commons.wikimedia.org/wiki/File:Svm_max_sep_hyperplane_with_margin.png`.

The job of the problem (1.9) is to find the best linear classifier $x$ as determined by the regulariser and the loss function. The loss function does not penalise $x$ if $1 - b_i a_i^T x \leq 0$, which can be expanded as

$$\begin{cases} 1 \leq a_i^T x, & \text{if } b_i = 1, \\ a_i^T x \leq -1, & \text{if } b_i = -1. \end{cases}$$

Observing that the orthogonal distance of $a_i$ from $H_x$ is $|a_i^T x|/\|x\|$, we therefore see that the $i$:th loss function does not penalise $x$ if $a_i$ gets the correct classification and is further than the **margin** $1/\|x\|$ from $H_x$. This is illustrated in Figure 1.4b.

In the problem (1.9), the regulariser $g(x) = \lambda \|x\|_2^2/2$ attempts to minimise the margin, while the combination of the loss functions and linear predictors attempts to reduce mis-classifications, and do correct classification with a wide margin. Large $\lambda$ will yield small $x$ and consequently a wide margin and small penalisation of mis-classifications. The wide margin can cause correct classifications to also be penalised if they're too close to being mis-classified. Small $\lambda$ will yield large $x$ and consequently a narrow margin and large penalisation of mis-classifications. Indeed, the SVM allows mis-classification of outliers and otherwise unseparable data through the loss function approach instead of strict constraints.

Remark 1.1 (Affinely separable data and multiple classes). The basic SVM only supports the two classes ±1 separated by $H_x$. Multi-class classification has to be done by, e.g., pairwise separation of each class, and a voting mechanism between the different classifications. Also, the general approach described here only allows hyperplanes $H_x$ containing the origin—so cannot separate classes that are not separated by such hyperplanes—but if we lift the problem into a higher-dimensional space by replacing $a_i$ by $a_i' = (a_i, 1)$, it is easy to support affine separating hyperplanes.

Example 1.3 (Non-linear SVM). Non-linear support vector machines basically amount to transforming the data $x$ into a higher-dimensional space, and then applying the basic linear support vector machine. We refer to [8, 9] for more details and examples of these **kernel methods**.

Here we just observe the general idea. Specifically, one takes a **kernel** $\kappa$, such as the radial basis function (RBF) kernel

$$\kappa(a_i, a_j) := \exp(-\|a_i - a_j\|^2/(2\sigma))$$

for some $\sigma > 0$. Then one constructs the matrix

$$K := \begin{pmatrix} \kappa(a_1, a_1) & \cdots & \kappa(a_1, a_n) \\ \vdots & \ddots & \vdots \\ \kappa(a_n, a_1) & \cdots & \kappa(a_n, a_n) \end{pmatrix}.$$

Decomposing $K = U^T \Lambda U$ for a diagonal matrix $\Lambda$ of eigenvectors, and an orthonormal matrix $U = (u_1, \ldots, u_n)$ of eigenvectors, one then defines

$$\widetilde{a}_i := \Lambda^{1/2} u_i \in \mathbb{R}^n.$$

If $n \gg m$ (where $a_i \in \mathbb{R}^m$), one now replaces $a_i$ in the SVM (1.9) by $\widetilde{a}_i$. Essentially this replaces $a_i$ by its similarity to the rest of the data, as measured by the kernel $\kappa$, and then decomposed into similarity features by the eigen-decomposition. In the exponential RBF, the parameter $\sigma$ controls how wide is the window (after a fashion) around $a_i$, that $a_j$ is considered similar to $a_i$.

Example 1.4 (Non-linear SVM numerical example). If $a_1 = (1, 1)^T$, $a_2 = (-1, 0)^T$, and $a_3 = (0, -1)^T$, with $\sigma = 1$ we get $\widetilde{a}_1 \approx (0, 0.7, 0.8)^T$, $\widetilde{a}_2 \approx (0.8, -0.2, -0.2)^T$, and $\widetilde{a}_3 \approx (0.2, 0.7, 0.8)^T$. If now $b_1 = 1$ and $b_2, b_3 = -1$, then clearly $\widetilde{a}_1$ is separable from $\widetilde{a}_2$ and $\widetilde{a}_3$ by the first coordinate.

Example 1.5 (Lasso). Let $a_i$ a data vector associated with a dependent variable or measurement $b_i \in \mathbb{R}$. Basic linear regression seeks the least squares solution $x$ to the typically over-determined problem

$$a_i^T x = b_i, \quad (i = 1, \ldots, n). \tag{1.10}$$

In other words, one solves the least squares problem

$$\min_{x \in \mathbb{R}^m} \frac{1}{n} \sum_{i=1}^n \frac{1}{2} \|b_i - a_i^T x\|_2^2.$$

Sometimes, one wants $x$ to be sparse—to have many zero elements, and few non-zero elements—to find the most important coordinates to describe the relationships in the data $\{(a_i, b_i)\}$. For example $a_i$ might be the attributes (genre, length, etc.) of a film, and $b_i$ its rating. The sparse $x$ would then tell the most relevant attributes for the rating, and their relative weighting. To do such sparse or **regularised regression**, let us in (1.8) set $\phi_i(z) = \frac{1}{2}\|z - b_i\|_2^2$ and $g(x) = \lambda\|x\|_1$. Then we obtain the so-called **Lasso**

$$\min_{x \in \mathbb{R}^m} \frac{1}{n} \sum_{i=1}^n \frac{1}{2} \|b_i - a_i^T x\|_2^2 + \lambda \|x\|_1$$

To explain the data, the one-norm regularisation term in Lasso causes it to automatically select more relevant features from the data, ignoring irrelevant ones.

Example 1.6 (Lasso numerical example). Suppose $a_1^T = (1, 0)$ and $b_1 = 1$, as well as $a_2^T = (0, 1)$ and $b_2 = 0.5$. For example, $b_1$ could be the rating of a meal based on flavour alone, and $b_2$ based on appearance alone.

The system (1.10) is fully determined and gives $x = (1, 0.5)^T$. This gives the weighting for flavour and appearance in the rating of the meal.

From the Lasso, if $2\lambda < 0.5$, we get $x = (1 - 2\lambda, 0.5 - 2\lambda)^T$, but if $2\lambda \in [0.5, 1]$, we get $x = (1 - 2\lambda, 0)^T$. In other words, the first component of $x$ is more important in determining the data than the second component. In our meal interpretation, flavour is more important than appearance, although its weighting gets discounted by $\lambda$. (Remember how in Example 1.1 everything gets shrunk by $\lambda$.)

## 1.4 Segmentation and computer vision

Let us return to image processing, and the Tikhonov-regularised inverse problems framework (1.7). Suppose the image pixels $\Omega = \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$ naturally divide into two subsets mutually disjoint subsets $\Omega_1$ and $\Omega_0$. That is, $\Omega_1 \cap \Omega_0 = \emptyset$, and $\Omega_1 \cup \Omega_0 = \Omega$. Specifically, we are interested in the case that $\Omega_1$ is a foreground object, and $\Omega_0$ is the image background. In fact, we want to discover $\Omega_1$. One way to do this is to set for some parameter $\theta > 0$ as the regulariser

$$R(x) = \mathrm{MS}_\theta(x) := \frac{1}{2}\|\widetilde{D}x|\Omega_0\|_2^2 + \frac{1}{2}\|\widetilde{D}x|\Omega_1\|_2^2 + \theta \cdot \mathrm{length}(\Gamma),$$

where $\Gamma$ is the boundary of $\Omega_1$, and the restriction

$$[\widetilde{D}x|\Omega_k]_j := \begin{cases} [\widetilde{D}x]_j, & \text{the calculation of } [\widetilde{D}x]_j \text{ in (1.4) only involves pixels within } \Omega_k, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, the first two terms of $R$ only penalise the image gradients within $\Omega_0$ and $\Omega_1$, ignoring any crossings over the boundary $\Gamma$. We only penalise the object boundary $\Gamma$ by its length, not how much the intensity of the image $u$ jumps over $\Gamma$.

$\mathrm{MS}_\theta$ is known as the **Mumford–Shah** regulariser. The corresponding denoising-type problem

$$\min_{x \in \mathbb{R}^{n_1 n_2}, \Gamma} \frac{1}{2}\|z - x\|^2 + \alpha \mathrm{MS}_\theta(x) \tag{1.11}$$

is known as the Mumford–Shah image **segmentation** problem. Segmentation of images forms the basis of computer vision, which might be described as the pipeline[2]

1. **Segment.** Through solution of the Mumford–Shah problem or otherwise, discover the objects in an image. In the simplest case, split the image into foreground $\Omega_1$ and background $\Omega_0$.

2. **Classify.** Through a classifier, such as the SVM that we already studied, or a neural network, classify the discovered objects. Again, in the simplest case, classify the contents or shape of the image region $\Omega_1$.

3. **Track.** When a sequence of images is available, track the movement of the discovered objects.

4. **Control.** Adapt the movement of a robot or other autonomous gadget to the discovered changes or state of surroundings.

In this course, we concentrate on the computational tools needed to realise the first two steps: optimisation algorithms for non-smooth problems, and **tricks to obtain tractable optimisation problems**. Indeed, the Mumford–Shah problem is computationally very difficult, as it is both non-convex and non-smooth. We will return to ways to deal with the non-convexity in our final Chapter 5. First we have to deal with non-smoothness.

[2] For an excellent overview, see also the annotated slides of Andrew Blake's Gibbs lecture at http://www.ams.org/meetings/lectures/BlakeGibbsLecture.pdf

## 1.5 About the course

As we have already seen, modern approaches to image processing, machine learning, and various big data applications, almost invariably involve the solution of non-smooth optimisation problems. The main part of this course studies **two (and a half) tricks** to deal with the non-smoothness. These are: splitting methods and duality, as well as saddle point problems closely related to the latter. These tricks are the topics of the respective Chapters 3 and 4. Before this, we however start in Chapter 2 with the necessary basic convex analysis, including the convex subdifferential. After this main part, in Chapter 5 we return to practical segmentation approaches based on the Mumford–Shah problem, and indeed introduce a further bag of tricks to deal non-convexity. Sections and proofs marked with a star (⋆) are additional material not covered in the lectures due to time constraints. Exercises marked with a star (⋆) are more challenging ones than the rest.

Basic convex analysis, with which we start, may be studied from [10] and [11]. The infinite-dimensional case is treated in the classic [12], and more comprehensively in [13]. For brushing up on basics of numerical optimisation of smooth functions, we point to [1]—such background is however not strictly necessary. All that is required is knowledge of undergraduate calculus and linear algebra, as well as elementary geometry. For more background on data science and machine learning, we refer to [8], while for image processing we recommend [6].

# 2 Convex subdifferentials

## 2.1 Convex sets

We know intuitively what a convex set is: one can see from any point in the set, to any other point in the set. This is illustrated in Fig. 2.1a, and is also the proper definition of a convex set.

Definition 2.1. A subset $C \subset \mathbb{R}^n$ is convex if

$$\lambda x + (1 - \lambda)y \in C, \quad \text{whenever} \quad x, y \in C,\ \lambda \in [0, 1].$$

Exercise (Light) 2.1. *Verify carefully that the following sets are convex:*

*(i) The open ball* $\operatorname{int} \mathbb{B}(x, \alpha) \in \mathbb{R}^n$ *and the closed ball* $\mathbb{B}(x, \alpha)$.

*(ii) The set* $\prod_{i=1}^n [a_i, b_i] \in \mathbb{R}^n$ *for* $a_i \le b_i$, $(i = 1, \ldots, n)$.

*(iii) The intersection* $C \cap D$ *of convex sets* $C$ *and* $D$.

*(iv) The image* $AC$ *of any convex set* $C \in \mathbb{R}^m$ *under linear transformation by matrix* $A \in \mathbb{R}^{n \times m}$.

## 2.2 Convex functions

One way to define a convex function is that the epigraph $\operatorname{epi} f$ is convex; cf. Figure 2.1b.

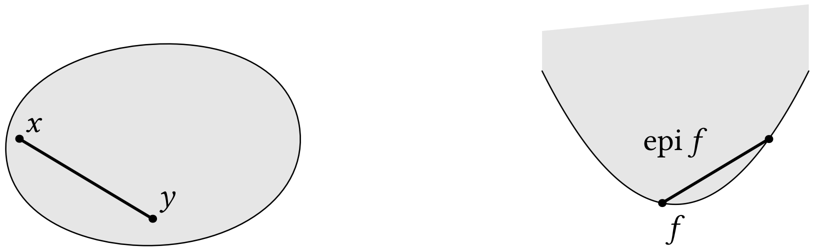
Definition 2.2. Let $\overline{\mathbb{R}} := [-\infty, \infty]$ denote the extended real numbers. The **epigraph** of a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is the set

$$\operatorname{epi} f := \{(x, t) \in \mathbb{R}^n \times \mathbb{R} \mid t \ge f(x),\ x \in \mathbb{R}^n\}.$$

We however provide a more explicit definition:

Definition 2.3. We say that $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is **convex** if

$$f(\lambda x + (1 - \lambda)y) \le \lambda f(x) + (1 - \lambda)f(y), \quad (x, y \in \mathbb{R}^n;\ \lambda \in [0, 1]).$$



**(a)** If $x, y \in C$ for a convex set $C$, the entire line segment between the points belongs to $C$.

**(b)** The line segment with start points within the epigraph of a convex function $f$, belongs completely to the epigraph.

**Figure 2.1:** Illustrations of convex sets and functions

Example 2.1. Any norm is convex, indeed $\|\lambda x + (1-\lambda)y\| \leq \lambda\|x\| + (1-\lambda)\|y\|$.

For our application purposes, the next exercise covers the most interesting types of convex functions.

Exercise 2.2. *Show that the following functions are convex:*

*(i) Any linear function $x \mapsto \langle x, a\rangle$ for some $a \in \mathbb{R}^n$.*

*(ii) Any linear combination $\sum_{i=1}^n \alpha_i f_i$ of convex functions $f_i$ with $\alpha_i \geq 0$.*

*(iii) $x \mapsto f(Ax)$, if $A \in \mathbb{R}^{n\times m}$ is a matrix, and $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ convex.*

*(iv) $t \mapsto |t|^p$ for $t \in \mathbb{R}$ is convex for $p \geq 1$.*

*(v) $t \mapsto -\log t$ if $t \geq 0$ and $\infty$ otherwise.*

Hint: *For the last two examples, try to write the epigraph as the intersection of affine half-spaces $A_x := \{(z, v) \mid v - f(z) \geq f'(x)(z-x)\}$.*

Example 2.2. For a set $C \subset \mathbb{R}^n$, we define the **indicator function**

$$\delta_C(x) := \begin{cases} 0, & x \in C, \\ \infty, & x \notin C. \end{cases}$$

Then $C$ is convex if and only if $\delta_C$ is convex.

Exercise (Light) 2.3. *For a convex function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, show that the **sub-level sets***

$$\operatorname{lev}_c f := \{x \in \mathbb{R}^n \mid f(x) \leq c\}$$

*are convex for any $c \in \overline{\mathbb{R}}$.*

Exercise 2.4. *Show that $f : \Omega \to \overline{\mathbb{R}}$ is a convex function if and only if* epi $f$ *is a convex set, cf. Figure 2.1b.*

## 2.3 Properties of (convex) functions

Frequently, we will be making some additional assumptions about our convex function $f$.

Definition 2.4. Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$. We then say that

(i) $f$ is **proper**, if $f(x) < \infty$ for some $x \in \mathbb{R}^n$, and $f(x) > -\infty$ for all $x \in \mathbb{R}^n$.

(ii) $f$ is **lower semicontinuous at** $x$ if for any sequence $\{x^i\}_{i=1}^\infty \subset \mathbb{R}^n$, with $x^i \to x$ holds

$$f(x) \leq \liminf_{i\to\infty} f(x^i).$$

(iii) $f$ is **lower semicontinuous**, if it is lower semicontinuous at every $x \in \mathbb{R}^n$.

Exercise 2.5. *Show that* $\operatorname{epi} f$ *is closed if and only if* $f$ *is lower semicontinuous, and that* $\operatorname{cl}\operatorname{epi} f$ *is convex for convex* $f$*.*

This exercise motivates the following definition.

Definition 2.5. The **closure** or **lower semicontinuous envelope** of $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is the function $\operatorname{cl} f : \mathbb{R}^n \to \overline{\mathbb{R}}$ defined by

$$\operatorname{epi}(\operatorname{cl} f) = \operatorname{cl}(\operatorname{epi} f).$$

All of these properties are important for optimisation problems, as evidence by the next proposition.

Proposition 2.1. *Let* $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ *be proper and lower semicontinuous, and* $C \subset \mathbb{R}^n$ *closed and bounded. Then there exists* $\hat{x} \in C$ *such that*

$$f(\hat{x}) = \inf_{x \in C} f(x),$$

*and this value is finite.*

*Proof.* Let

$$M := \inf_{x \in C} f(x).$$

Suppose $M = -\infty$. Then there exists a sequence $\{x^i\}_{i=1}^{\infty} \subset C$ with $f(x^i) \leq -i$ for each $i \in \mathbb{N}$. Since $C$ is closed and bounded, we can find a limit point $x \in C$ of a subsequence. By lower semicontinuity of $f$, then

$$f(x) \leq \lim_{i \to \infty} (-i) = -\infty.$$

This is in contradiction to $f$ being proper.

So $M > -\infty$. Since $f$ is proper, there exists a point $x' \in \mathbb{R}^n$ such that $f(x') < \infty$. Therefore also $M < \infty$.

So $M$ is finite. We may then take a **minimising sequence** $\{x^i\}_{i=1}^{\infty} \subset C$, such that

$$f(x^i) \leq M + 1/i.$$

Again, we may find a limit point $x$ of a subsequence, and see by lower semicontinuity that $f(x) = M$. We have found our $\hat{x} = x$. □

*Alternative proof.* The set $\widetilde{E} := \operatorname{epi} f \cap (C \times \mathbb{R})$ is closed. Since $f$ is proper, we may find a point $x$ with $f(x) < \infty$. If we let $E := \widetilde{E} \cap ([-\infty, f(x)] \times \mathbb{R})$, then $E$ is non-empty, because $f$ is proper. Now taking $z^i := (x^i, f(x^i)) \in \widetilde{E}$ for a minimising sequence (which eventually and w.log satisfies $f(x^i) \leq f(x)$, we either find that $f(x^i) \searrow -\infty$, a contradiction, or may switch to a compact subset of $E$, where a subsequence of $z^i$ converges. □

Remark 2.1. Note that we did not yet use convexity for the previous proposition.

## 2.4 Subdifferentials

Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be Fréchet-differentiable at $x \in \mathbb{R}$. That is, the gradient $\nabla f(x) := z$ exists, defined by

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - \langle z, h \rangle}{\|h\|} = 0. \tag{2.1}$$

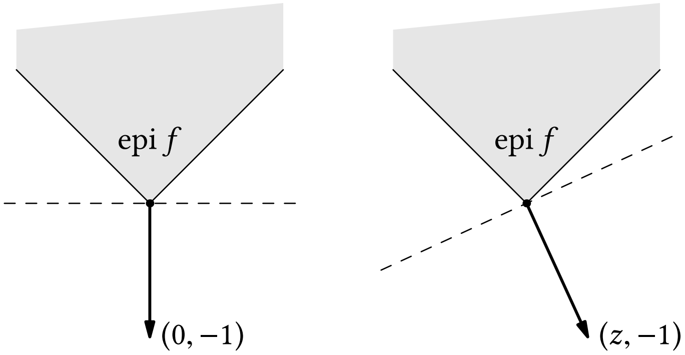
Note that this can also be written

$$\lim_{h \to 0} \frac{1}{\|h\|} \left\langle \begin{pmatrix} x+h \\ f(x+h) \end{pmatrix} - \begin{pmatrix} x \\ f(x) \end{pmatrix}, \begin{pmatrix} z \\ -1 \end{pmatrix} \right\rangle = 0.$$

The vector $(z, -1)$ is therefore a (Fréchet-)normal to $\operatorname{epi} f$; see Figure 2.2.

It follows from (2.1) that

$$\lim_{h \to 0} \frac{f(x+h) - f(x) - \langle z, h \rangle}{\|h\|} \geq 0. \tag{2.2}$$



**(a)** Epigraph of $f(x) = |x|$ with a supporting hyperplane and normal vector at $(0,0)$.

**(b)** An alternative supporting hyperplane and normal vector at $(0,0)$.

**Figure 2.2:** Epigraphs, supporting hyperplanes, and their normal vectors $(z,-1)$. The supporting hyperplane in (a) together with the orthogonal vector, correspond to optimality conditions.

Definition 2.6. Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$. If $z$ satisfies (2.2), we say that $z$ is a **Fréchet subgradient of $f$ at $x$**. We denote the set of all Fréchet subgradients of $f$ at $x$ by $\partial_F f(x)$.

If $f$ is convex, we have the following simple characterisation.

Lemma 2.1. *If $f$ is convex,* (2.2) *is equivalent to*

$$f(x+h) - f(x) \geq \langle z, h \rangle, \quad (h \in \mathbb{R}^m). \tag{2.3}$$

*Proof.* Indeed, (2.3) implies

$$\lim_{h\to 0} f(x+h) - f(x) - \langle z, h \rangle \geq 0,$$

which implies (2.2).

On the other hand, if (2.3) does not hold, then

$$f(x+h) - f(x) \leq \langle z, h \rangle - \epsilon$$

for some $h \in \mathbb{R}^n \setminus \{0\}$ and $\epsilon > 0$. For any $i \in \mathbb{N}$ it follows

$$\begin{aligned} f(x+h/2^i) - f(x) &= f((x+h)/2^i + (1-1/2^i)x) - f(x) \\ &\leq (1/2^i)f(x+h) - (1/2^i)f(x) \leq \langle z, h/2^i \rangle - \epsilon/2^i. \end{aligned}$$

Therefore, setting $h_i := h/2^i$, we have

$$\lim_{i\to\infty} \frac{f(x+h_i) - f(x) - \langle z, h_i \rangle}{\|h_i\|} \leq \lim_{i\to\infty} \frac{-\epsilon/2^i}{\|h\|/2^i} = -\epsilon/\|h\|.$$

This violates (2.2). □

This motivates the following definition.

Definition 2.7. Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex, and $x \in \mathbb{R}^n$. If $z \in \mathbb{R}^n$ satisfies

$$f(x') - f(x) \geq \langle z, x' - x \rangle, \quad \text{for all } x' \in \mathbb{R}^n, \tag{2.4}$$

we say that $z$ is a **(convex) subgradient of $f$ at $x$**. We denote the set of all convex subgradients of $f$ at $x$ by $\partial f(x)$.

Geometrically, we already know that $(z,-1)$ for any $z \in \partial f(x)$ is normal to a supporting tangent hyperplane

$$H = \{(x', f(x) + \langle z, x' - x\rangle) \in \mathbb{R}^{n+1} \mid x' \in \mathbb{R}^n\}$$

of $\operatorname{epi} f$ at $(x, f(x))$; see Figure 2.2b. Therefore the entire set $\partial f(x)$ provides a collection of such. Moreover, each hyperplane supports the whole function globally, not just locally, in the sense that $\operatorname{epi} f$ stays on one side of $H$.

Corollary 2.1. *Suppose $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex and Fréchet-differentiable at $x$. Then $\partial f(x) = \{\nabla f(x)\}$.*

*Proof.* The inclusion $\nabla f(x) \in \partial f(x)$ is immediate from Lemma 2.1 because (2.2) holds. For the inclusion $\partial f(x) \subset \{\nabla f(x)\}$, suppose $z \neq \nabla f(x) \in \partial f(x)$. We can write $z = \nabla f(x) + d$ for $d \neq 0$. For any $h_i := d/i$, we have $h_i \to 0$ as $i \to \infty$, as well as

$$\lim_{i\to\infty} \frac{f(x+h_i) - f(x) - \langle z, h_i\rangle}{\|h_i\|} = \lim_{i\to\infty} \frac{\langle d, h_i\rangle}{\|h_i\|} = -\|d\|.$$

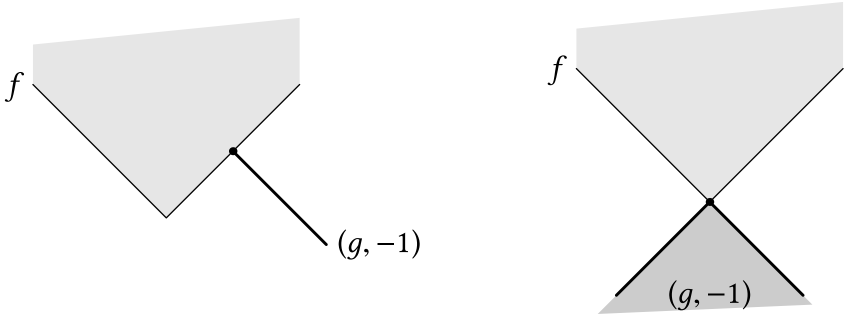
This is in contradiction to (2.2). Therefore, by Lemma 2.1, $z \notin \partial f(x)$. It follows $\partial f(x) = \{\nabla f(x)\}$. □

Example 2.3. Let $f(x) = \frac{1}{2}\|z-x\|_2^2$ for some $z \in \mathbb{R}^n$. Then $f$ is differentiable with $\nabla f(x) = x - z$. By Corollary 2.1 thus $\partial f(x) = \{x - z\}$.

Example 2.4. Let $f(x) = |x|$ for $x \in \mathbb{R}$. Then

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

This is illustrated in Figure 2.3.



**(a)** $\partial f(x) = \{\operatorname{sgn} x\}$ at $x \neq 0$ **(b)** $\partial f(x) = [-1,1]$ at $x = 0$

**Figure 2.3:** Subdifferentials of $f(x) = |x|$.

Example 2.5. Let $C \subset \mathbb{R}^n$ be a convex set. Then the subdifferential of the indicator function $\delta_C$ is the **normal cone**

$$\partial \delta_C(x) = N_C(x) := \{z \in \mathbb{R}^n \mid \langle x' - x, z\rangle \leq 0 \text{ for all } x' \in C\}.$$

We illustrate this in Figure 2.4.

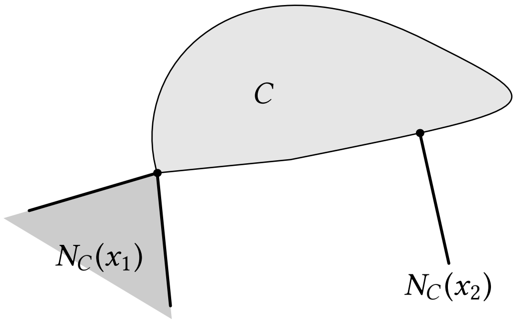


**Figure 2.4:** Normal cones of $f = \delta_C$ at two points $x_1$ and $x_2$.

Exercise 2.6. *What is the subdifferential of* $\|x\|_2$ *on* $\mathbb{R}^n$*?*

## 2.5 Effective domains and relative interiors

For convex sets, a relative definition of the interior is often useful.

Definition 2.8. For a convex set $C \subset \mathbb{R}^n$, we define the **relative interior** $\operatorname{ri} C$ as the interior of $C$ relative to the smallest affine subspace $V \supset C$.

*If the usual interior* $\operatorname{int} A \neq \emptyset$*, then* $\operatorname{ri} A = \operatorname{int} A$*.* This is the most common case, and in the applications that we consider in this course, only this case will occur. The next examples however illustrate the idea of the relative interior. The linear constraint in Example 2.7 is a very typical case when the relative interior can be needed.

Example 2.6. Let $C = \{c\}$ for some $c \in \mathbb{R}^n$. Then, as a a singleton, as 0-dimensional convex set, $C$ itself is the smallest affine subspace containing $C$. Thus $\operatorname{ri} C = C$.

Example 2.7. Let $A \in \mathbb{R}^{k \times m}$, and $b \in \mathbb{R}^m$. Set $C := \{x \in \mathbb{R}^m \mid Ax = b\}$. Then $C$ is a convex set, indeed itself an affine subspace of $\mathbb{R}^m$. Thus $\operatorname{ri} C = C$.

Example 2.8. For vectors $a, b \in \mathbb{R}^n$, define the line segment

$$C := \{\lambda a + (1 - \lambda)b \mid 0 \leq \lambda \leq 1\}.$$

This is a one-dimensional set with

$$\operatorname{ri} C = \{\lambda a + (1 - \lambda)b \mid 0 < \lambda < 1\}.$$

Definition 2.9. For a proper function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, we define the **effective domain**

$$\operatorname{dom} f := \{x \in \mathbb{R}^n \mid f(x) < \infty\}.$$

Lemma 2.2. *Let* $C \subset \mathbb{R}^n$ *be a non-empty convex set. Then* $\operatorname{ri} C$ *is non-empty.*

*Proof.* Let $V$ be the smallest affine subspace containing $C$. (Such a $V$ is unique and exists, since the intersection of two distinct affine subspace $V$ and $V'$ is still an affine subspace of lower dimension.) Let $k$ be the dimension of $V$. Then $C$ contains at least $k$ linearly independent vectors $x_k$. Otherwise the dimension $k$ of $V$ would not be minimal. As a convex set $C \supset \{\sum_{j=1}^k \lambda_k x_k \mid \sum_{j=1}^k \lambda_k = 1,\ \lambda_k \geq 0\}$. But then $\operatorname{ri} C = \operatorname{int}_V C \supset \Delta := \{\sum_{j=1}^k \lambda_k x_k \mid \sum_{j=1}^k \lambda_k = 1,\ 1 > \lambda_k > 0\}$, where $\operatorname{int}_V C$ denotes the interior of $C$ a subset of $V$. Clearly the set $\Delta$ is non-empty. □

Proposition 2.2. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex and proper. Then $\operatorname{ri} \operatorname{dom} f \neq \emptyset$.*

*Proof.* Since $f$ is proper, $\operatorname{dom} f$ is non-empty. □

Exercise (★) 2.7. *Show that a convex function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is continuous on $\operatorname{ri} \operatorname{dom} f$. Conclude that $\operatorname{cl} f = f$ on $\operatorname{ri} \operatorname{dom} f$, and that*

$$\operatorname{ri}(\operatorname{epi} f) = \{(x, t) \in \mathbb{R}^{n+1} \mid x \in \operatorname{ri}(\operatorname{dom} f),\ t > f(x)\}.$$

## 2.6 (★) Properties of the subdifferential as a set-valued map

The **subdifferential** $\partial f : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is an example of a **set-valued map**: for each $x \in \mathbb{R}^n$, the value is a subset of $\mathbb{R}^n$, $\partial f(x) \subset \mathbb{R}^n$. For general set-valued functions the equivalent concept of subdifferentiability is given by the next definition.

Definition 2.10. A set-valued function $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is **monotone** if

$$\langle A(x') - A(x), x' - x \rangle \geq 0, \quad (x', x \in \mathbb{R}^n).$$

(This inequality is to be understood in the sense

$$\langle y' - y, x -' x \rangle \geq 0, \quad (x', x \in \mathbb{R}^n;\ y' \in A(x'),\ y \in A(x))\Big).$$

Exercise 2.8. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex. Show that $\partial f$ is monotone.*

Exercise (★) 2.9. *Show that $\partial f$ is, in fact,* ***maximal monotone****. This means that there is no monotone operator $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ such that $\operatorname{Graph} A \supset \operatorname{Graph} \partial f$.* Hint: *Observe that any $z \in \mathbb{R}^n$ can be written as $z = x + y$ for $x \in \mathbb{R}^n$ and $y \in \partial f(x)$ for some convex function $f$.*

We want to build some calculus rules for the convex subdifferential. For that, we need some additional results and concepts.

Proposition 2.3. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex, proper, and lower semicontinuous. Then the sub-differential mapping $\partial f$ is* ***outer semicontinuous****, meaning that for any sequence $x^i \to x$, and $z^i \in \partial f(x^i)$, any limit $z$ of a converging subsequence of $\{z^i\}$, satisfies $z \in \partial f(x)$. We denote*

$$\limsup_{i \to \infty} \partial f(x^i) \subset \partial f(x).$$

*Moreover $\partial f(x)$ is a closed set at each $x \in \mathbb{R}^n$.*

*Proof.* Assume, without loss of generality, that $\{z^i\}$ converges to $z$. Choose arbitrary $x' \in \mathbb{R}^n$. We have by Definition 2.7 that

$$f(x') \geq f(x^i) - \langle z^i, x' - x^i \rangle, \quad (i \in \mathbb{N}).$$

The map $(\widetilde{z}, \widetilde{x}) \mapsto \langle \widetilde{z}, x' - \widetilde{x} \rangle$ is continuous, and by assumption $f$ is lower semicontinuous. Therefore

$$f(x') \geq \liminf_{i \to \infty} \Big( f(x^i) - \langle z^i, x' - x^i \rangle \Big) \geq f(x) - \langle z, x' - x \rangle.$$

Since this holds for any $x' \in \mathbb{R}^n$, we have proved that $z \in \partial f(x)$.

Finally, the closedness of $\partial f(x)$ is immediate from the definition, or choosing $x^i = x$ above. □

## 2.7 (★) Directional differentials and support functions

Definition 2.11. For $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, we define the **directional differential** at $x \in \mathbb{R}^n$ in the direction $h \in \mathbb{R}^n$ by

$$f'(x; h) := \lim_{t \searrow 0} \frac{f(x + th) - f(x)}{t}. \tag{2.5}$$

Lemma 2.3. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex and proper, and $x \in \operatorname{dom} f$. Then*

$$\partial f(x) = \{z \in \mathbb{R}^n \mid \langle z, h \rangle \leq f'(x; h)\ \forall h \in \mathbb{R}^n\}, \tag{2.6}$$

*and*

$$\operatorname{cl}(h \mapsto f'(x; h)) = \sup_{z \in \partial f(x)} \langle z, h \rangle. \tag{2.7}$$

*If $x \in \operatorname{ri} \operatorname{dom} f$, moreover*

$$f'(x; h) = \sup_{z \in \partial f(x)} \langle z, h \rangle. \tag{2.8}$$

*Proof.* Observe that

$$f'(x; h) = \inf_{t > 0} \frac{f(x + th) - f(x)}{t}. \tag{2.9}$$

Indeed, for any $0 < s < t$ by convexity

$$\frac{s}{t} f(x + th) + \frac{t - s}{t} f(x) \geq f(x + sh).$$

This gives

$$f(x + th) - f(x) \geq \frac{t}{s} \Big( f(x + sh) - f(x) \Big).$$

Therefore, the sequence $s \mapsto \frac{f(x+sh)-f(x)}{s}$ is monotonically increasing, proving (2.9).

If we define

$$A := \{z \in \mathbb{R}^n \mid \langle z, h \rangle \leq f'(x; h) \text{ for all } h \in \mathbb{R}^n\},$$

then (2.9) and (2.4) show that $A = \partial f(x)$. This proves (2.6). Observe also from the continuity of $h \mapsto \langle z, h \rangle$ that $A$ is closed (this also follows from Proposition 2.3), and that

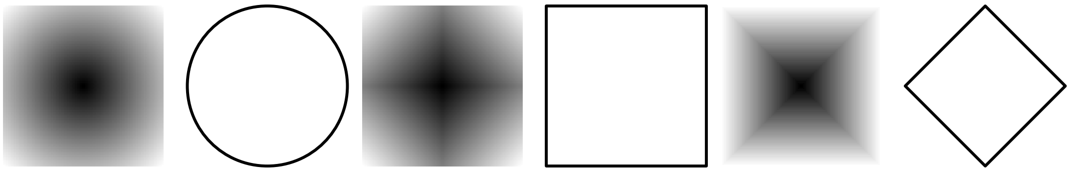
$$A = \{z \in \mathbb{R}^n \mid \langle z, h \rangle \leq \operatorname{cl}[f'(x; \cdot)](h) \text{ for all } h \in \mathbb{R}^n\}. \tag{2.10}$$

Defining the **support function** of the *closed* convex set $A$,

$$\sigma_A(h) := \sup_{z \in A} \langle z, h \rangle,$$

we find that $\sigma_A$ is proper, lower semicontinuous, and **sublinear**,

$$\sigma_A(s_1 h_1 + s_2 h_2) \leq s_1 \sigma_A(h_1) + s_2 \sigma_A(h_2), \quad (h_1, h_2 \in \mathbb{R}^n; s_1, s_2 \geq 0).$$



**(a)** For the two-norm, $\partial\|0\|_2$ is the unit circle. **(b)** For the one-norm, $\partial\|0\|_1$ is the rectangle $[-1,1]^2$. **(c)** For the $\infty$-norm, $\partial\|0\|_1$ is the diamond.

**Figure 2.5:** Some support functions on $\mathbb{R}^2$ and their corresponding convex sets.

Also $f'(x;\,\cdot\,)$ is proper and sublinear (although possibly not lower semicontinuous). Proving this is the content of Exercise 2.10. It follows easily that $\operatorname{cl}[f'(x;\,\cdot\,)]$ is proper, sublinear, and lower semicontinuous. Since by (2.10), $A$ is the maximal convex set $A'$ satisfying $\sigma_{A'} \le \operatorname{cl}[f'(x;\,\cdot\,)]$, the next therefore lemma shows that $\sigma_A = \operatorname{cl}[f'(x;\,\cdot\,)]$.

Finally, if $x \in \operatorname{ri}\operatorname{dom} f$, we have $\operatorname{cl}[f'(x;\,\cdot\,)] = f'(x;\,\cdot\,)$ by the lower semicontinuity of $f$ on $\operatorname{ri}\operatorname{dom} f$ (Exercise 2.7). This shows (2.8). □

Exercise 2.10. *For a convex proper function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, prove that $f'(x;\,\cdot\,)$ is proper and sublinear at $x \in \operatorname{dom} f$.*

Lemma 2.4. *Let $\sigma : \mathbb{R}^n \to \mathbb{R}$ be proper, lower semicontinuous, and sub-linear. Then $\sigma$ is the support function of the convex set*

$$\partial\sigma(0) = \{z \in \mathbb{R}^n \mid \langle z, h\rangle \le \sigma(h) \text{ for all } h \in \mathbb{R}^n\}. \tag{2.11}$$

*That is*

$$\sigma = \sigma_{\partial\sigma(0)},$$

*Further, $\sigma_A$ is sub-linear for any convex set $A$.*

Some very common support functions $\sigma$ and corresponding "dual balls" $\partial\sigma(0)$ are depicted in Figure 2.5.

*Proof.* A sub-linear function is convex. For any convex function $f$, we have

$$f(x) = \sup_{x'\in\mathbb{R}^n,\ z\in\partial f(x')} f(x') + \langle z, x - x'\rangle. \tag{2.12}$$

Indeed, by the definition of the subdifferential, $\ge$ holds here, while choosing $x' = x$ gives equality. Since a sub-linear function is **positively homogeneous**, meaning

$$\sigma(\lambda x) = \lambda\sigma(x) \quad \text{for} \quad \lambda > 0,$$

we have

$$\partial\sigma(\lambda x) = \partial\sigma(x), \quad \text{for all} \quad x \ne 0,\ \lambda > 0. \tag{2.13}$$

By the outer semicontinuity of $\partial\sigma$ (Proposition 2.3), letting $\lambda \searrow 0$, we see that

$$\partial\sigma(x) \subset \partial\sigma(0), \quad \text{for any} \quad x \in \mathbb{R}^n.$$

Let $x' \in \mathbb{R}^n$, and $z \in \partial\sigma(x')$. Then, since $z \in \partial\sigma(\lambda x')$, we get

$$0 = \sigma(\lambda x') - \sigma(x') \ge \langle z, \lambda x' - x'\rangle \ge \sigma(\lambda x') - \sigma(x').$$

Thus, for any $\lambda > 0$ and $x \in \mathbb{R}^n$, we have

$$\sigma(x') + \langle z, x - x'\rangle = \sigma(\lambda x') + \langle z, x - \lambda x'\rangle.$$

Letting $\lambda \searrow 0$, we have

$$\sigma(x') + \langle z, x - x'\rangle = \langle z, x\rangle.$$

Thus by (2.13), we have

$$\begin{aligned}\sigma(x) &= \sup_{x' \in \mathbb{R}^n,\, z \in \partial\sigma(x')} (\sigma(x') + \langle z, x - x'\rangle) \\ &= \sup_{x' \in \mathbb{R}^n,\, z \in \partial\sigma(x')} \langle z, x\rangle \\ &= \sup_{z \in \partial\sigma(0)} \langle z, x\rangle.\end{aligned}$$

This proves (2.11).

Finally, that $\sigma_A$ is sub-linear for any convex set $A$, follows immediately from the definition. □

Lemma 2.5. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be a proper convex function. Then*

*(i) $\partial f(x) = \emptyset$ for every $x \notin \operatorname{dom} f$.*

*(ii) $\partial f(x) \neq \emptyset$ for every $x \in \operatorname{ri} \operatorname{dom} f$.*

*Proof.* The first claim is clear from the definition of the subdifferential: if $x \notin \operatorname{dom} f$, (2.4) gives the condition

$$f(x') - \infty \geq \langle z, x' - x\rangle, \quad (x' \in \mathbb{R}^n),$$

which cannot hold.

For the second claim, let $y \in \operatorname{dom} f$, and $x \in \operatorname{ri} \operatorname{dom} f$. If we cannot choose $y$ distinct from $x$, then $\operatorname{ri} \operatorname{dom} f = \operatorname{dom} f = \{\bar{x}\}$ for some $\bar{x} \in \mathbb{R}^n$. This by the properness of $f$ means that for some constant $c \in \mathbb{R}$ holds

$$f(x) = \begin{cases} c, & x = \bar{x}, \\ \infty & \text{otherwise.}\end{cases}$$

But, as is easily verified, $\partial f(\bar{x}) = \mathbb{R}^n$. Thus (ii) holds in this degenerate case.

For the rest, we may thus assume $y$ distinct from $x$. With $h := y - x$, writing $x$ as the convex combination

$$x = \frac{t}{1+t} y + \frac{1}{1+t}(x - th),$$

we then deduce

$$\frac{1}{1+t} f(x - th) - f(x) \geq -\frac{t}{1+t} f(y).$$

Thus

$$f(x - th) - f(x) \geq t(f(x) - f(y)).$$

In consequence

$$f'(x; -h) \geq f(x) - f(y) = C' > -\infty.$$

By Lemma 2.3 we observe that $\partial f(x)$ has to be non-empty. □

Remark 2.2. In the context of the proof, it can be that $f'(x; h) = -\infty$. Consider, for example,

$$f(x) = \begin{cases} \infty, & x < 0 \\ 1, & x = 0, \\ 0, & x > 0.\end{cases}$$

With $x = 0$, any $y > 0$, and $h = y - x > 0$, we have $f'(x; h) = -\infty$. The crucial bit is that $f'(x; -h) > -\infty$; in this example $f'(x; -h) = \infty$. This example illustrates how convex functions with non-full domain can exhibit somewhat strange behaviour.

## 2.8 Subdifferential calculus

Theorem 2.1. *Suppose $f, g : \mathbb{R}^n \to \overline{\mathbb{R}}$ are convex and proper. Then at any point $x \in \operatorname{dom}(f+g)$ one has*

$$\partial(f+g)(x) \supset \partial f(x) + \partial g(x).$$

*If $\operatorname{ri}\operatorname{dom} f \cap \operatorname{ri}\operatorname{dom} g \neq \emptyset$, then this holds as an equality.*

*Proof* ($\star$). Take first $z \in \partial f(x)$, and $w \in \partial g(x)$. Then (2.4) immediately shows that $z+w \in \partial(f+g)(x)$. This shows the claimed inclusion.

To prove the equality under the additional assumption, we note from Lemma 2.5 for each $q = f, g, f+g$ that $\partial q(x)$ is non-empty. By Lemma 2.3 this implies

$$q'(x; h) > -\infty$$

for any $h$. Since $q'(x; 0) = 0$, we find that $q'(x; \cdot)$ is proper. Hence we can sum $f'(x; \cdot)$ and $g'(x; \cdot)$. Now

$$\limsup_{t \searrow 0} \frac{(f+g)(x+th) - (f+g)(x)}{t} \leq \limsup_{t \searrow 0} \frac{f(x+th) - f(x)}{t} + \limsup_{t \searrow 0} \frac{g(x+th) - g(x)}{t}.$$

Also

$$\inf_{t>0} \frac{f(x+th) - f(x)}{t} + \inf_{t>0} \frac{g(x+th) - g(x)}{t} \leq \inf_{t>0} \frac{(f+g)(x+th) - (f+g)(x)}{t}.$$

Recalling the equivalence (2.9), and the definition (2.5), therefore

$$(f+g)'(x; h) = f'(x; h) + g'(x; h). \tag{2.14}$$

This would be enough for the application of the formulas provided by Lemma 2.3, if we had $x \in \operatorname{ri}\operatorname{dom} f \cap \operatorname{ri}\operatorname{dom} g \cap \operatorname{ri}\operatorname{dom}(f+g)$. In general, without requiring this condition, using that $q'(x; \cdot)$ is proper for $q = f, g, f+g$, (2.14) implies

$$\liminf_{h' \to h} (f+g)'(x; h') = \liminf_{h' \to h} f'(x; h') + \liminf_{h' \to h} g'(x; h').$$

That is

$$\operatorname{cl}[(f+g)'(x; \cdot)] = \operatorname{cl}[f'(x; \cdot)] + \operatorname{cl}[g'(x; \cdot)].$$

(Note that taking the closure here is only effective if $x$ is not in the relative interior of the domain of one of the functions.) Lemma 2.3 therefore gives

$$\begin{aligned} \sup_{q \in \partial(f+g)(x)} \langle h, q \rangle &= \sup_{z \in \partial f(x)} \langle h, z \rangle + \sup_{w \in \partial g(x)} \langle h, w \rangle \\ &= \sup_{q \in \partial f(x) + \partial g(x)} \langle h, q \rangle. \end{aligned} \tag{2.15}$$

Since this holds for every $h \in \mathbb{R}^n$, and both $\partial(f+g)(x)$ and $\partial f(x) + \partial g(x)$ are *closed* convex sets, we conclude equivalence. Indeed, if there was a point $z \in \big(\partial f(x) + \partial g(x)\big) \setminus \partial(f+g)(x)$, it would be at a positive distance from $\partial(f+g)(x)$, and yield a contradiction to the statement (2.15) on the support functions of these sets. □

The condition $\operatorname{ri}\operatorname{dom} f \cap \operatorname{ri}\operatorname{dom} g \neq \emptyset$, when applied to $f = \delta_C$ and $g = \delta_D$, is a form of **constraint qualification** that the reader may be familiar with from basic optimisation courses.

Example 2.9. The equality $\partial(f+g)(x) = \partial f(x) + \partial g(x)$ does not always holds without the condition $\operatorname{ri}\operatorname{dom} f \cap \operatorname{ri}\operatorname{dom} g \neq \emptyset$. Take, for example, $e = (1, 0) \in \mathbb{R}^2$ and set $f = \delta_{\mathbb{B}(-e,1)}$, and $g = \delta_{\mathbb{B}(e,1)}$. Then $f + g = \delta_{\{0\}}$; see Figure 2.6. Recalling Example 2.5, clearly $\partial(f+g)(0) = N_{\{0\}}(0) = \mathbb{R}^2$, while $\partial f(0) = N_{\mathbb{B}(-e,1)}(0) = N_{\mathbb{B}(0,1)}(e) = [0, \infty)e$, and $\partial g(0) = N_{\mathbb{B}(e,1)}(0) =$

$N_{\mathbb{B}(0,1)}(-e) = -[0,\infty)e$. Thus, in contradiction to summability, we obtain

$$\partial f(0) + \partial g(0) = \mathbb{R}e \subsetneq \mathbb{R}^2 = \partial(f+g)(0).$$

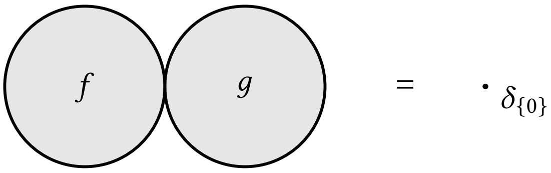


**Figure 2.6:** Non-summability of the subdifferential in Example 2.9.

Exercise 2.11. *Let $A \in \mathbb{R}^{n\times m}$, and $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex. Show that $\partial(f \circ A)(x) \supset A^T[\partial f](Ax)$ with equality if $\mathcal{R}(A) \cap \operatorname{ri}\operatorname{dom} f \neq \emptyset$.*

## 2.9 Characterisation of minima

We now concentrate on convex $f : \mathbb{R}^n \to \overline{\mathbb{R}}$. How can we characterise minima of such functions? Going back to (2.4), we see that if $z = 0$, we have

$$f(x') - f(x) \geq 0, \quad \text{for all } x' \in \mathbb{R}^n.$$

This means exactly that $x$ is a minimiser. Since this works both ways, we obtain the following.

Theorem 2.2. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ by convex. Then $x \in \mathbb{R}^n$ is a minimiser of $f$,*

$$f(x) = \min_{x' \in \mathbb{R}^n} f(x'),$$

*if and only if*

$$0 \in \partial f(x). \tag{2.16}$$

Example 2.10 (Derivation of soft thresholding solution). Recall from Example 1.1 the problem

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|z - x\|_2^2 + \lambda\|x\|_2, \tag{2.17}$$

whose solution we claimed to equal

$$\hat{x} = \begin{cases} 0, & \|z\|_2 \leq \lambda, \\ z(1 - \lambda/\|z\|_2), & \|z\|_2 > \lambda. \end{cases} \tag{2.18}$$

Setting $f(x) = \frac{1}{2}\|z - x\|_2^2$ and $g(x) = \lambda\|x\|_2$, we deduce that the conditions of the sum rule Theorem 2.1 are satisfied. Therefore, by Theorem 2.2, $\hat{x}$ is a solution to (2.17) if and only if $0 \in \partial f(\hat{x}) + \partial g(\hat{x})$. This can be also written as

$$z \in \hat{x} + \begin{cases} \mathbb{B}(0,\lambda), & \hat{x} = 0, \\ \lambda\frac{\hat{x}}{\|\hat{x}\|_2}, & \hat{x} \neq 0. \end{cases}$$

Since in the second case clearly $z \propto \hat{x}$, so that $\hat{x}/\|\hat{x}\|_2 = z = /\|z\|_2$, we easily obtain (2.18).

Example 2.11 (Karush–Kuhn–Tucker conditions). Let $C \subset \mathbb{R}^n$ be a convex set, and $f : \mathbb{R}^n \to \mathbb{R}$ convex and differentiable. Then, by Theorem 2.1 and Example 2.5, we have

$$\hat{x} \in \arg\min_{x \in C} f(x)$$

if and only if

$$0 \in \nabla f(\hat{x}) + N_C(\hat{x}).$$

In particular, let

$$C = \{x \in \mathbb{R}^n \mid g(x) \le 0\},$$

for some convex, differentiable, constraint function $g$ satisfying

$$\inf_{x \in \mathbb{R}^n} g(x) < 0. \tag{2.19}$$

Then, as we will shortly see

$$N_C(x) = \begin{cases} \emptyset, & g(x) > 0, \\ \{0\}, & g(x) < 0, \\ [0, \infty)\nabla g(x), & g(x) = 0. \end{cases} \tag{2.20}$$

Therefore, we recover the usual Karush–Kuhn–Tucker conditions

$$\nabla f(\hat{x}) + \lambda \nabla g(\hat{x}) = 0 \quad \text{with} \quad \lambda \ge 0,\ \lambda g(\hat{x}) = 0,\ g(\hat{x}) \le 0.$$

(★) To see the expression (2.20), we first of all recall that if $x \notin C = \operatorname{dom} \delta_C$, then $N_C(x) = \partial \delta_C(x)$ is empty. Otherwise, $z \in N_C(x)$ for $x \in C$ is defined by

$$0 \ge \langle z, x' - x \rangle, \quad (\text{for all } x',\ g(x') \le 0). \tag{2.21}$$

If $g(x) < 0$, we can find $\delta > 0$ such that $g(x') < 0$ for $\|x' - x\| < \delta$. Therefore, we see that the only possibility is $z = 0$, that is, $N_C(x) = \{0\}$. The case $g(x) = 0$ remains. Since $g$ is continuous and (2.19) holds, we deduce

$$C = \operatorname{cl}\{x' \in \mathbb{R}^n \mid g(x') < 0\}.$$

By convexity of $g$, if $g(x') < 0$, then $g(\lambda x' + (1 - \lambda)g(x) \le \lambda g(x') < 0$ for any $\lambda \in (0, 1)$. We now note from (2.9) that $g(x') < 0$ if and only if $x' = x + \lambda h$ for some $\lambda > 0$ and $h \in \mathbb{R}^n$ with $g'(x; h) < 0$. Therefore

$$\begin{aligned} C &= \operatorname{cl}\{x + \lambda h \mid \lambda > 0,\ h \in \mathbb{R}^n,\ g'(x; h) < 0\} \\ &= \operatorname{cl}\{x + \lambda h \mid \lambda > 0,\ h \in \mathbb{R}^n,\ \operatorname{cl}[g'(x; \cdot)](h) \le 0\}. \end{aligned}$$

Since the the normal cone of an open set agrees with the normal cone of the closure, we deduce that $z \in N_C(x)$ if and only if

$$0 \ge \langle z, h \rangle, \quad (\text{for all } h,\ \operatorname{cl}[g'(x; \cdot)](h) \le 0).$$

By Lemma 2.3, this is the same as

$$0 \ge \langle z, h \rangle, \quad (\text{for all } h, \langle \nabla g(x), h \rangle \le 0).$$

This shows that $z = \lambda \nabla g(x)$ for some $\lambda \ge 0$.

## 2.10 Strong convexity and smoothness

Definition 2.12. Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex. We say that $f$ is

(i) **strictly convex** if (2.4) holds strictly, that is

$$f(x') - f(x) > \langle z, x' - x \rangle, \quad (x' \neq x \in \mathbb{R}^n;\, z \in \partial f(x)).$$

(ii) $\gamma$-**strongly-convex** for $\gamma > 0$ if

$$f(x') - f(x) \geq \langle z, x' - x \rangle + \frac{\gamma}{2}\|x' - x\|^2, \quad (x', x \in \mathbb{R}^n;\, z \in \partial f(x)).$$

Obviously, strong convexity implies strict convexity.

Lemma 2.6. *Suppose $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is strictly convex. Then it has at most one minimiser.*

*Proof.* Let $\hat{x}$ be a minimiser. By Theorem 2.2, $0 \in \partial f(\hat{x})$. By strict convexity then

$$f(x') > f(x), \quad (x' \in \mathbb{R}^n). \qquad \square$$

Definition 2.13. Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex. We say that $f$ is $L$-**smooth** if it is differentiable and

$$f(x') \leq f(x) + \langle \nabla f(x), x' - x \rangle + \frac{L}{2}\|x' - x\|^2, \quad (x', x \in \mathbb{R}^n). \tag{2.22}$$

One could, in principle, not require differentiability in Definition 2.13, and replace $\nabla f$ by $\partial f$ in (2.22). Exercise 2.13 shows that this would lead nowhere.

For the next chapter, on optimisation methods, the following consequence is important. It introduces a stronger version of monotonicity of $\nabla f$.

Lemma 2.7. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex and $L$-smooth. Then $\nabla f$ is $L^{-1}$-**co-coercive**, that is*

$$L^{-1}\|\nabla f(x) - \nabla f(y)\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle, \quad (x, y \in \mathbb{R}^n). \tag{2.23}$$

*Proof.* We have

$$f(x') \leq f(x) + \langle \nabla f(x), x' - x \rangle + \frac{L}{2}\|x' - x\|^2. \tag{2.24}$$

Thus, adding $\langle \nabla f(y), x - x' \rangle$ on both sides, we get

$$f(x') - \langle \nabla f(y), x' \rangle \leq f(x) - \langle \nabla f(y), x \rangle - \langle \nabla f(x) - \nabla f(y), x' - x \rangle + \frac{L}{2}\|x' - x\|^2.$$

The left hand side is minimised by $x' = y$. Using $x' = x + L^{-1}(\nabla f(x) - \nabla f(y))$ on the right-hand side gives

$$f(y) - \langle \nabla f(y), y \rangle \leq f(x) - \langle \nabla f(y), x \rangle - \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2.$$

A fully analogous argument, starting from (2.24) with roles of $x$ and $y$ exchanged, gives

$$f(x) - \langle \nabla f(x), x \rangle \leq f(y) - \langle \nabla f(x), y \rangle - \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2.$$

Summing these two estimates, we obtain (2.23). $\square$

Exercise 2.12. *Show that $f(x) := \frac{1}{2}\|z - x\|^2$ is 1-smooth.*

Exercise 2.13. *Show that the following are equivalent:*

*(i) $L$-smoothness of $f$,*

*(ii) $L^{-1}$-co-coercivity of $\nabla f$.*

*(iii) Lipschitz continuity of $\nabla f$ with factor $L$.*

# 3 Convex conjugates and duality

## 3.1 Convex conjugate

Convex conjugates provide a powerful tool for transforming difficult optimisation problems into more tractable ones. We start with the definition, examples, and some basic properties.

Definition 3.1. Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be a general, possibly non-convex function. We then define the **(convex) conjugate**

$$f^*(y) := \sup_{x \in \mathbb{R}^n} (\langle x, y \rangle - f(x)).$$

We also denote the second conjugate $f^{**} := (f^*)^*$.

Example 3.1. Let $z \in \mathbb{R}$, and set $f(z) = |z|$. Then

$$f^*(y) = \max_{y \in \mathbb{R}} (zy - |z|) = \delta_{[-1,1]}(y).$$

We can also write

$$f(z) = \max_{y \in [-1,1]} zy = \max_{y \in \mathbb{R}} zy - f^*(y) \quad \text{for} \quad f^*(y) = \delta_{[-1,1]}.$$

Therefore $f^{**} = f$. This is a general property of convex, proper, and lower semicontinuous, as we will see in Theorem 3.1 below.

Example 3.2. Let $z \in \mathbb{R}^{2M}$ for some $M$, and recall our 2-1 norm

$$f(z) = \|z\|_{2,1} = \sum_{k=1}^{M} \sqrt{z_k^2 + z_{M+k}^2},$$

from the total variation denoising (1.3). Similarly to Example 3.1, we can then compute

$$f^*(y) = \begin{cases} 0, & \max_{k=1,\ldots,M} |y_k^2 + y_{M+k}^2| \leq 1, \\ \infty, & \text{otherwise} \end{cases}$$

This is the indicator function of the product of pixelwise (index $k$) two-dimensional unit balls.

We may again verify $f^{**} = f$, as

$$f(z) = \sum_{k=1}^{M} \max\{z_k y_k + z_{M+k} y_{M+k} \mid |y_k^2 + y_{M+k}^2| \leq 1\} = \max_{y \in \mathbb{R}^{2M}} \langle y, z \rangle - f^*(y).$$

This is almost the most important example of conjugacy for our needs.

Example 3.3 (★). The support function $\sigma_A$ equals $\delta_A^*$ for any set $A \subset \mathbb{R}^n$. In Theorem 3.1 below we will see that if $A \neq \emptyset$ is convex and closed, then the opposite also holds, $\delta_A = \sigma_A^*$. In particular, the norms in Figure 2.5 are in one-to-one correspondence with the corresponding unit balls

$B_q = \partial\|\cdot\|_p(0)$ also through $\delta_{B_q} = (\|\cdot\|_p)^*$ for $q$ the conjugate exponent of $p$. This is defined through $1/p + 1/q = 1$.

Exercise 3.1. *What are the conjugate functions of*

*(i)* $g(x) = \|z - x\|_2^2/2$, *(*$x \in \mathbb{R}^n$*)?*

*(ii)* $\phi(t) = \max\{0, 1 - bt\}$, *(*$t \in \mathbb{R}$*)?*

*Do* $g^{**} = g$ *and* $\phi^{**} = \phi$ *hold?*

The next exercise and proposition list some basic properties of $f^*$ for arbitrary $f$.

Exercise 3.2. *Show that the function* $f^*$ *is convex and lower semicontinuous for any* $f : \mathbb{R}^n \to \overline{\mathbb{R}}$*. Also show that* $f^*$ *is proper if* $f$ *is proper, lower semicontinuous, and* ***level-bounded****. The latter means that all of the level sets* $\operatorname{lev}_c f$ *are bounded.*

Lemma 3.1. *Let* $f : \mathbb{R}^n \to \overline{\mathbb{R}}$*. Then*

*(i)* $f \geq f^{**}$*.*

*(ii) (Fenchel–Young)* $f(x) + f^*(y) \geq \langle x, y\rangle$ *for all* $x, y \in \mathbb{R}^n$*.*

*Proof.* We first of all note that by definition of $f^*$ holds

$$f^*(y) \geq \langle y, x\rangle - f(x), \quad (y, x \in \mathbb{R}^n). \tag{3.1}$$

Since $f$ is proper, we cannot have $f(x) = -\infty$, so simple rearrangements quickly yield (ii).

To prove (i), we note that if $f^{**}(x) < \infty$, then for every $\epsilon > 0$ we can find $y$ with

$$f^{**}(x) \leq \langle x, y\rangle - f^*(y) + \epsilon.$$

Combining this with (3.1) yields

$$f^{**}(x) \leq f(x) + \epsilon.$$

Since $\epsilon > 0$ was arbitrary, we get $f^{**} \leq f$.

If $f^{**}(x) = \infty$, we can likewise for any $k \geq 0$ find $y$ such that

$$f^{**}(x) \geq \langle x, y\rangle - f^*(y) \geq k.$$

This shows for any $x' \in \mathbb{R}^n$ that

$$\langle x, y\rangle - \big(\langle x', y\rangle - f(x')\big) \geq k.$$

Choosing $x' = x$ shows that $f(x) \geq k$. Since $k \geq 0$ was arbitrary, $f(x) = \infty$. This finishes the proof of (i). □

For convex $f$, we have the following stronger relationships.

Theorem 3.1. *Let* $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ *be convex, proper, and lower semicontinuous. Then*

*(i) (Fenchel–Moreau)* $f = f^{**}$*.*

*(ii)* $f(x) + f^*(y) = \langle x, y\rangle$ *if and only if* $y \in \partial f(x)$*.*

*(iii)* $y \in \partial f(x)$ *if and only if* $x \in \partial f^*(y)$*.*

*Proof.* We already know from Lemma 3.1(i) that $f \ge f^{**}$. If $f^{**}(x) = \infty$, then this already shows that $f(x) = f^{**}(x)$. We may therefore suppose that $f^{**}(x) < \infty$. By Exercise 3.2, we know that $f^{**}$ is proper, so also $f^{**}(x) > -\infty$. If there exists some $y \in \partial f(x) \neq \emptyset$, then by Theorem 2.2, $f^*(y) = \langle y, x\rangle - f(x)$. This shows that

$$f^{**}(x) \ge \langle x, y\rangle - f^*(y) \ge f(x).$$

This establishes that $f^{**} = f$ on $\operatorname{dom}\partial f := \{x \in \mathbb{R}^n \mid \partial f(x) \neq \emptyset\}$.

We then observe that

$$y \in \partial f(x) \implies x \in \partial f^*(y). \tag{3.2}$$

Indeed, suppose $y \in \partial f(x)$. By Theorem 2.2, this holds if and only if

$$f^*(y) = \langle y, x\rangle - f(x). \tag{3.3}$$

In particular, (ii) holds. By Lemma 3.1(ii), moreover

$$f(x) + f^*(y') \ge \langle x, y'\rangle. \tag{3.4}$$

The inequality (3.4) and equality (3.3) imply

$$f^*(y') - f^*(y) \ge \langle y' - y, x\rangle.$$

Thus $x \in \partial f^*(y)$, so (3.2) holds.

The same argument naturally also establishes

$$x \in \partial f^*(y) \implies y \in \partial f^{**}(x). \tag{3.5}$$

Thus $\partial f^{**}(x) \supset \partial f(x)$ for all $x \in \mathbb{R}^n$. With this, (2.12), and the fact that $f = f^{**}$ on $\operatorname{dom}\partial f$ that we have proven above, we deduce

$$\begin{aligned} f(x) &= \sup_{x' \in \mathbb{R}^n,\, y \in \partial f(x')} f(x') + \langle y, x - x'\rangle \\ &= \sup_{x' \in \operatorname{dom}\partial f,\, y \in \partial f(x')} f(x') + \langle y, x - x'\rangle \\ &\le \sup_{x' \in \operatorname{dom}\partial f,\, y \in \partial f^{**}(x')} f^{**}(x') + \langle y, x - x'\rangle = f^{**}(x). \end{aligned}$$

This proves (i). To prove (iii), we simply use (i) in (3.5), and combine this with (3.2). □

## 3.2 Fenchel–Rockafellar duality

The next theorem provides a very useful duality correspondence.

**Theorem 3.2 (Fenchel–Rockafellar "lite").** *Let $f : \mathbb{R}^m \to \overline{\mathbb{R}}$ and $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex, proper, and lower semicontinuous, and $K \in \mathbb{R}^{m\times n}$. Then we have* ***weak duality***

$$\inf_{x\in\mathbb{R}^n} (g(x) + f(Kx)) + \inf_{y\in\mathbb{R}^m} \left(g^*(-K^T y) + f^*(y)\right) \ge 0. \tag{3.6}$$

*Suppose*

$$K(\operatorname{ri}\operatorname{dom} g) \cap \operatorname{int}\operatorname{dom} f \neq \emptyset, \tag{3.7}$$

*and that $x \mapsto g(x) + f(Kx)$ has a minimiser $\hat{x}$. Then we have* ***strong duality***

$$\min_{x\in\mathbb{R}^n} (g(x) + f(Kx)) + \min_{y\in\mathbb{R}^m} \left(g^*(-K^T y) + f^*(y)\right) = 0. \tag{3.8}$$

*Proof.* For any $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, we have by Lemma 3.1(ii) that

$$g(x) + g^*(-K^T y) \geq -\langle x, K^T y\rangle \quad \text{and} \quad f(Kx) + f^*(y) \geq \langle Kx, y\rangle. \tag{3.9}$$

Summing these expressions shows (3.6).

To show (3.8), it suffices that both inequalities in (3.9) hold as equalities for some $x = \hat{x}$ and $y = \hat{y}$.By Theorem 3.1(ii) this is the case if and only if

$$-K^T\hat{y} \in \partial g(\hat{x}), \quad \text{and} \quad \hat{y} \in \partial f(K\hat{x}). \tag{3.10}$$

To prove (3.10). Here we use our assumption of the existence of a minimiser $\hat{x}$ of $x \mapsto g(x) + f(Kx)$. By Theorem 2.2 and Theorem 2.1, whose conditions are verified by (3.7), this satisfies

$$0 \in \partial g(\hat{x}) + \partial(f \circ K)(\hat{x}).$$

The condition (3.7) also implies $\mathcal{R}(K) \cap \operatorname{ri}\operatorname{dom} f \neq \emptyset$ by (3.7). Exercise 2.11 therefore shows that $\partial(f \circ K)(\hat{x}) = K^T\partial f(K\hat{x})$. Thus there exists $\hat{y} \in \partial f(K\hat{x})$ such that $0 \in \partial g(\hat{x}) + K^T\hat{y}$. In other words, (3.10) holds. This finishes the proof of (3.8) and strong duality. □

Remark 3.1. The condition $K(\operatorname{ri}\operatorname{dom} g) \cap \operatorname{int}\operatorname{dom} f \neq \emptyset$ is enough for strong duality, without requiring the existence of a minimiser, albeit with the first "min" remaining an "inf" in (3.8). Even more relaxed conditions exist [14]. We stick to our stronger requirements, as the relaxed ones demand a little bit more machinery than we have time for, and in practise we are interested in the case when (3.10) is satisfied.

Remark 3.2. Note that (3.10) holding implies that $\hat{x}$ is the minimiser required for the theorem. Moreover, under (3.10), it is not necessary to assume (3.7), which was only used to prove (3.10).

Due to the relationships (3.6) and (3.8), we call

$$\min_{y \in \mathbb{R}^m} \; g^*(-K^T y) + f^*(y) \tag{D}$$

the **dual problem** of the **primal problem**

$$\min_{x \in \mathbb{R}^n} \; g(x) + f(Kx). \tag{P}$$

We denote by

$$\mathcal{G}(x, y) := g(x) + f(Kx) + g^*(-K^T y) + f^*(y) \geq 0$$

the **duality gap**. It is only zero when $x$ solves (P), and $y$ solves (D). Hence $\mathcal{G}(x, y) \leq \epsilon$ for a suitable solution quality $\epsilon > 0$ forms a good stopping criterion, independent of any knowledge of the optimal solution, for **primal-dual algorithms** that simultaneously look for primal and dual solutions $\hat{x}$ and $\hat{y}$.

Corollary 3.1 (Primal–dual optimality conditions). *Suppose* (3.7) *holds. Then the next conditions are equivalent:*

*(i)* $\hat{x} \in \mathbb{R}^n$ *and* $\hat{y} \in \mathbb{R}^m$ *achieve the minima in* (3.8),

*(ii)* $\mathcal{G}(\hat{x}, \hat{y}) = 0$, *and*

*(iii)* (3.10) *holds, i.e.,* $-K^T\hat{y} \in g(\hat{x})$ *and* $\hat{y} \in f^*(K\hat{x})$,

*(iv)* $-K^T\hat{y} \in g(\hat{x})$ *and* $K\hat{x} \in \partial f^*(\hat{y})$.

The conditions (iv) will in particular be practical for the algorithms that we develop in Chapter 4.

*Proof.* The equivalence of (i) and (ii) is clear from Theorem 3.2, while the equivalence of (iv) to (iii) is immediate from Theorem 3.1(iii).

To finish the proof, it is thus enough to show that (i) is equivalent to (iii). We have already seen in the proof of Theorem 3.2 that of a solution $(\hat{x}, \hat{y})$ of (3.10), the primal solution $\hat{x}$ is exactly a minimiser of the primal problem. We now note that (3.10) can using Theorem 3.1(iii) be rewritten

$$\hat{x} \in g^*(-K^T\hat{y}) \quad \text{and} \quad K\hat{x} \in \partial f^*(\hat{y}). \tag{3.11}$$

This is to say that $0 \in -K\partial g^*(-K^T\hat{y}) + \partial f^*(\hat{y})$. In other words, from Exercise 2.11, we have $0 \in \partial(g^* \circ (-K^T))(\hat{y}) + \partial f^*(\hat{y})$. By Theorem 2.2 and Theorem 2.1, $\hat{y}$ is therefore a minimiser of the dual problem. Thus (i) is equivalent to (iii). □

Remark 3.3. To recover a solution $\hat{x}$ to the primal problem (P) from a solution $\hat{y}$ the dual problem (D), observe from (3.10) that $-K^T\hat{y} \in \partial g(\hat{x})$. If, for example, $g(x) = \frac{1}{2}\|z - x\|^2$, then we can solve $\hat{x} = z - K^T\hat{y}$. Alternatively, if we can efficiently compute $g^*$, we can use (3.11) to recover $\hat{x}$ from $\hat{y}$.

Example 3.4 (Dual of soft thresholding). Recall the soft-thresholding problem of Example 1.1. There $K = I$, $g(x) = \frac{1}{2}\|z - x\|^2$, and $f(x) = \|x\|_2$. We can derive $g^*(y) = \frac{1}{2}\|z + y\|^2 - \frac{1}{2}\|z\|^2$, and $f^*(y) = \delta_{\mathbb{B}(0,\lambda)}$. Therefore, we obtain the dual problem

$$\min_{y \in \mathbb{R}^m} \frac{1}{2}\|z + y\|^2 + \delta_{\mathbb{B}(0,\lambda)}(y) - \frac{1}{2}\|z\|^2.$$

For the purposes of computing a minimiser to this dual problem, we can ignore the constant term $\frac{1}{2}\|z\|^2$. Following Remark 3.3, if we have a solution $\hat{y}$ to the dual problem, we obtain a primal solution $\hat{x} = z - \hat{y}$.

Example 3.5 (Duals of empirical risk minimisation problems). Consider the empirical risk minimisation problem (1.8), that is

$$\min_{x \in \mathbb{R}^m} g(x) + \frac{1}{n}\sum_{i=1}^n \phi_i(a_i^T x). \tag{3.12}$$

We can also write this as

$$\min_{x \in \mathbb{R}^m} g(x) + \phi(A^T x) \quad \text{for} \quad A := \begin{pmatrix} a_1 & \dots & a_n \end{pmatrix} \in \mathbb{R}^{m \times n} \quad \text{and} \quad \phi(z) := \frac{1}{n}\sum_{i=1}^n \phi_i(z_i).$$

The dual problem is

$$\min_{y \in \mathbb{R}^n} g^*(-Ay) + \phi^*(y).$$

Observing that $(ng)^*(z) = ng^*(z/n)$, we can also write this as

$$\min_{y \in \mathbb{R}^n} ng^*(-n^{-1}Ay) + \sum_{i=1}^n \phi_i^*(y_i).$$

(You can easily observe that since each $\phi_i$ only depends on $z_i$, the conjugate of $\sum_i \phi_i$ is the sum of the conjugates $\phi_i^*$ acting on $y_i$.)

Example 3.6 (Dual of linear SVM). Continuing from Example 3.5, for the linear SVM,

$$g(x) = \frac{\lambda}{2}\|x\|^2, \quad \text{and} \quad \phi_i(t) := \max\{0, 1 - b_i t\}.$$

These have the conjugates

$$g^*(z) = \frac{1}{2\lambda}\|z\|^2, \quad \text{and} \quad \phi_i^*(y_i) := \begin{cases} y_i/b, & y_i \in [-b_i, 0], \\ \infty, & \text{otherwise}, \end{cases}$$

where we denote $[-b_i, 0] := [0, -b_i]$ if $b_i < 0$. Expanded, the dual form of the SVM therefore is

$$\min_{y \in \prod_{i=1}^n [-b_i, 0]} \frac{1}{2\lambda n} y^T A^T A y + \sum_{i=1}^n y_i/b_i. \tag{3.13}$$

In this dual formulation, the non-smooth function $\phi^*$ therefore nicely splits into componentwise functions, with the "mixing" of the different coordinates of $y_i$ by $A$ moved into the smooth part $g^*(-Ay)$. This dual form of the problem will be easy to solve with the forward–backward splitting method that we introduce in the next section, while the original form is less trivial.

Example 3.7 (Dual of nonlinear SVM). Continuing from Example 3.6, for the nonlinear SVM of Example 1.3, $A^T A$ is a matrix of entries $\widetilde{a}_i^T \widetilde{a}_j = \kappa(a_i, a_j)$. The high dimensionality of the transformed problem therefore disappears in the dual formulation. In this way, the dual form (3.13) forms the computationally tractable basis of non-linear support vector machines.

Exercise 3.3. *What is the dual problem of the Lasso? Is this likely to be useful? How about TV denoising?*

## 3.3 Saddle-point problems

One way to derive primal-dual algorithms—algorithms that simultaneously look for a primal solution $\hat{x}$ and a dual solution $\hat{y}$—is to work **saddle point problems**.

Generally, for arbitrary $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ holds

$$\sup_y \inf_x L(x, y) \leq \inf_x \sup_y L(x, y). \tag{3.14}$$

Indeed, clearly $\sup_y L(\widetilde{x}, y) \geq \sup_y \inf_x L(x, y)$ for any $\widetilde{x}$. Taking the infimum over $\widetilde{x}$ proves (3.14). A **saddle point** $(\hat{x}, \hat{y})$ satisfies $L(\hat{x}, y) \leq L(\hat{x}, \hat{y}) \leq L(x, \hat{y})$ for all $x$ and $y$. Then

$$\inf_x \sup_y L(x, y) \leq \sup_y L(\hat{x}, y) = L(\hat{x}, \hat{y}), \quad \text{and} \sup_y \inf_x L(x, y) \quad \geq \inf_x L(x, \hat{y}) = L(\hat{x}, \hat{y}).$$

Therefore, if a saddle point exists, clearly

$$\inf_x \sup_y L(x, y) = \sup_y \inf_x L(x, y) = L(\hat{x}, \hat{y}).$$

In this case, since a point achieving the inf-sup value exists, we can write

$$\inf_x \sup_y L(x, y) = \min_x \max_y L(x, y).$$

Proposition 3.1. *Let $f : \mathbb{R}^m \to \overline{\mathbb{R}}$ and $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex, proper, and lower semicontinuous, and $K \in \mathbb{R}^{m\times n}$. Let $L$ be the* ***Lagrangian***

$$L(x, y) := g(x) + \langle y, Kx\rangle - f^*(y).$$

*Then a solution $(\hat{x}, \hat{y})$ to the primal–dual optimality conditions (3.10) is a saddle point of $L$ and vice versa.*

*Proof.* By Corollary 3.1(iv), (3.10) hold if and only if $\min_x L(x, \hat{y})$ is solved by $x = \hat{x}$, and $\max_y L(\hat{x}, y)$ is solved by $y = \hat{y}$. This says exactly that $L(\hat{x}, y) \le L(\hat{x}, \hat{y})$ and $L(\hat{x}, \hat{y}) \le L(x, \hat{y})$ for all $x$ and $y$. But this again is equivalent to saying that $(\hat{x}, \hat{y})$ is a saddle point of $L$. □

These considerations lead us to consider the specific saddle point problem

$$\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^m} g(x) + \langle y, Kx\rangle - f^*(y). \tag{S}$$

Under the conditions of Theorem 3.2, this problem can be derived from (P) by writing $f(Kx) = \sup_y \big(\langle y, Kx\rangle - f^*(y)\big)$.

Example 3.8. Continuing from Example 3.6, using the conjugate expression from Exercise 3.1, we can rewrite (3.12) for the SVM as

$$\min_{x\in\mathbb{R}^m} \max_{y\in\mathbb{R}^n} \frac{\lambda}{2}\|x\|_2^2 + \sum_{i=1}^{n} \frac{1}{n} y_i a_i^T x - \frac{1}{n}\phi_i^*(y^i).$$

With

$$g(x) := \frac{\lambda}{2}\|x\|_2^2, \quad \widetilde{f}^*(y) := \frac{1}{n}\sum_{i=1}^{n}\Big(\delta_{[-b_i,0]}(y_i) + y_i/b_i\Big), \quad \text{and} \quad K := A^T/n,$$

we obtain the standard-form saddle-point problem

$$\min_{x\in\mathbb{R}^m} \max_{y\in\mathbb{R}^n} g(x) + \langle Kx, y\rangle - \widetilde{f}^*(y).$$

Exercise 3.4. *What is the saddle point problem of TV denoising?*

# 4 Non-smooth optimisation methods

## 4.1 Surrogate objectives and gradient descent

Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex and differentiable. We want to find a point $\hat{x}$ such that

$$f(\hat{x}) = \min_{x \in \mathbb{R}^n} f(x). \tag{P}$$

As we have learned, this is of course characterised by

$$\nabla f(\hat{x}) = 0.$$

This system is, however, in most interesting cases difficult to solve analytically. So let us try to derive numerical methods. One way of deriving numerical methods is to replace the original difficult objective with a simpler one whose minimisation provides improvement to the original objective.

Definition 4.1. A function $\widetilde{f_{\bar{x}}} : \mathbb{R}^n \to \overline{\mathbb{R}}$ is a **surrogate objective** for $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ at $\bar{x}$ if $\widetilde{f_{\bar{x}}} \geq f$, and $\widetilde{f_{\bar{x}}}(\bar{x}) = f(\bar{x})$.

Starting with a point $x^0$, we would then minimise $\widetilde{f_{x^0}}$ to obtain a new point $x^{i+1}$. Through the properties of the surrogate objective, this will not increase the value of $f$. Hopefully it will provide significant improvement! Then we repeat the process, minimising $\widetilde{f_{x^1}}$, and so on.

What options are there for surrogate objectives, and which would be a good one? If $f$ is differentiable, one possibility is

$$\min_{x \in \mathbb{R}^n} \widetilde{f_{\bar{x}}}(x) := f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle + \frac{1}{2\tau} \|x - \bar{x}\|^2. \tag{4.1}$$

Here $\tau > 0$ is a suitable factor. In general $f(\bar{x}) = \widetilde{f_{\bar{x}}}(\bar{x})$. If $f$ is $L$-smooth per Definition 2.13, and $L\tau \leq 1$, then also $f \leq \widetilde{f_i}$. Therefore, in this case, $\widetilde{f_{\bar{x}}}$ is a valid surrogate objective, and minimising $\widetilde{f_{\bar{x}}}$ will provide improvement to $f$ as well.

The optimality condition $0 \in \partial \widetilde{f_{x^i}}(x)$ becomes

$$\nabla f(x^i) + \tau^{-1}(x - x^i) = 0. \tag{4.2}$$

This holds if $x^i = \hat{x}$ by taking also $x = \hat{x}$. Therefore, there is a direct correspondence between the solutions of the surrogate objective and the original. If $x^i \neq \hat{x}$, solving (4.2) for $x = x^{i+1}$, we get the rule

$$x^{i+1} = x^i - \tau \nabla f(x^i). \tag{GD}$$

This is known as the **gradient descent method**. In this context the quadratic term in (4.1) can be seen as a step length condition.

Will sequentially minimising $\widetilde{f_{x^i}}$ provide sufficient decrease in $f$ such that we obtain convergence of $\{x^i\}$ to a minimiser $\hat{x}$ of $f$? This is what we study next.

## 4.2 Fixed point theorems

Convergence of optimisation methods can often by proved through various fixed point theorems applied to the operator $T : x^i \mapsto x^{i+1}$, mapping one iterate to the next one. We will in particular use the following result from [15].

Theorem 4.1 (Browder fixed point theorem, version 1). *Let $T : \mathbb{R}^n \to \mathbb{R}^n$ be* ***firmly non-expansive****, that is*

$$\|T(x) - T(y)\|^2 \le \langle T(x) - T(y), x - y\rangle, \quad (x, y \in \mathbb{R}^n).$$

*Suppose $T$ admits some fixed point $x^* = T(x^*)$. Then, for any starting point $x^0 \in \mathbb{R}^n$, the iteration sequence $x^{i+1} := T(x^i)$ satisfies $x^i \to \widetilde{x}$ for some fixed point $\widetilde{x} = T(\widetilde{x})$.*

Remark 4.1. Firm non-expansivity is the co-coercivity of (2.23) with constant $L = 1$.

Th above variant of Browder's fixed point theorem follows from a more general one for averaging operators.

Definition 4.2. A map $T : \mathbb{R}^n \to \mathbb{R}^n$ is **non-expansive**, if

$$\|T(x) - T(y)\| \le \|x - y\|, \quad (x, y \in \mathbb{R}^n).$$

It is $\alpha$-**averaging**, if $T = (1 - \alpha)I + \alpha J$ for some non-expansive $J : \mathbb{R}^n \to \mathbb{R}^n$, and $\alpha \in (0, 1)$.

Theorem 4.2 (Browder fixed point theorem, version 2). *Let $T : \mathbb{R}^n \to \mathbb{R}^n$ be averaging, and suppose $T$ admits some fixed point $x^* = T(x^*)$. Then, for any starting point $x^0 \in \mathbb{R}^n$, the iteration sequence $x^{i+1} := T(x^i)$ satisfies $x^i \to \widetilde{x}$ for some fixed point $\widetilde{x} = T(\widetilde{x})$.*

Theorem 4.1 now follows from Theorem 4.2 and the following lemma.

Lemma 4.1. *$T : \mathbb{R}^n \to \mathbb{R}^n$ is firmly non-expansive if and only if it is $(1/2)$-averaging.*

*Proof.* Suppose $T$ is $(1/2)$-averaging. Then $T = (I + J)/2$ for some non-expansive $J$. We compute

$$\begin{aligned}\|T(x) - T(y)\|^2 &= \frac{1}{4}\left(\|J(x) - J(y)\|^2 + 2\langle J(x) - J(y), x - y\rangle + \|x - y\|^2\right)\\ &\le \frac{1}{2}\left(\langle J(x) - J(y), x - y\rangle + \|x - y\|^2\right)\\ &= \langle T(x) - T(y), x - y\rangle.\end{aligned}$$

Thus $T$ is firmly non-expansive.

Suppose then that $T$ is firmly non-expansive. If we show that $J := 2T - I$ is non-expansive, it follows that $T$ is $(1/2)$-averaging. This is established by the simple calculations

$$\begin{aligned}\|J(x) - J(y)\|^2 &= 4\|T(x) - T(y)\|^2 - 4\langle T(x) - T(y), x - y\rangle + \|x - y\|^2\\ &\le \|x - y\|^2.\end{aligned}$$

This completes the proof. □

Browder's fixed point theorem is a practical improvement over the classical **Banach fixed point theorem**.

Theorem 4.3 (Banach fixed point theorem). *Let $T : \mathbb{R}^n \to \mathbb{R}^n$ be a* ***contraction mapping****, that is for some $\kappa \in [0, 1)$ holds*

$$\|T(x) - T(y)\| \le \kappa\|x - y\|, \quad (x, y \in \mathbb{R}^n). \tag{4.3}$$

*Then $T$ admits a unique fixed point $x^* = T(x^*)$. This can be moreover discovered as the limit of the iteration sequence $x^{i+1} := T(x^i)$ for any starting point $x^0$.*

Note that firm non-expansivity implies non-expansivity, that is (4.3) with $\kappa = 1$, motivating the choice of the term. While non-expansivity is enough to show the existence of a fixed point of $T$ in some cases ($T$ maps a bounded convex set $C$ into itself [16]), it is not enough to show the convergence of the sequence $x^{i+1} := T(x^i)$ to a fixed point. So we need one of the stronger conditions: firm non-expansivity, the averaging property, or contractivity with $\kappa < 1$.

Theorem 4.4. *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is convex and $L$-smooth. If the step length $\tau \leq L^{-1}$, then, for any starting point $x^0 \in \mathbb{R}^n$, the iterates $\{x^i\}_{i=0}^\infty$ of the gradient descent method* (GD) *converge to a minimiser $\hat{x}$ of $f$.*

*Proof.* By Lemma 2.7, we have

$$L^{-1}\|\nabla f(x) - \nabla f(y)\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y\rangle, \quad (x, y \in \mathbb{R}^n). \tag{4.4}$$

The iteration (GD) may be written in terms of the operator

$$T(x) := x - \tau \nabla f(x).$$

Now

$$\begin{aligned}\|T(x) - T(y)\|^2 &= \langle T(x) - T(y), x - y\rangle - \tau\langle T(x) - T(y), \nabla f(x) - \nabla f(y)\rangle \\ &= \langle T(x) - T(y), x - y\rangle + \tau^2\|\nabla f(x) - \nabla f(y)\|^2 - \tau\langle \nabla f(x) - \nabla f(y), x - y\rangle \\ &\leq \langle T(x) - T(y), x - y\rangle.\end{aligned}$$

In the final step we have used (4.4) and $\tau \leq L^{-1}$. Thus $T$ is firmly non-expansive. Theorem 4.1 now proves the claim. □

## 4.3 Variational inclusions and the proximal point method

The gradient descent method is very basic, but often not very good. In particular, subgradient extensions of (GD) can have very slow convergence. Therefore we need alternative methods.

We now allow for general (possibly non-differentiable) convex functions $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, and replace the surrogate objective in (4.1) by another surrogate

$$\min_{x \in \mathbb{R}^n} \bar{f}_{\bar{x}}(x) := f(x) + \frac{1}{2\tau}\|x - \bar{x}\|^2. \tag{4.5}$$

In other words, we remove the linearisation, and try to minimise $f$ directly with a step length condition. Again $\bar{f}_{\bar{x}}(\bar{x}) = f(\bar{x})$, and clearly $f_{\bar{x}} \geq f$. Therefore $\bar{f}_{\bar{x}}$ is a valid surrogate objective for $f$ at $\bar{x}$. This time the optimality conditions for $x$ minimising $\bar{f}_{x^i}$ are

$$0 \in \partial f(x) + \tau^{-1}(x - x^i). \tag{4.6}$$

If $x^i = \hat{x}$ for $\hat{x}$ a minimiser of the original objective $f$, then (4.6) is solved by $x = \hat{x}$, so again there is a direct correspondence between the solutions of the surrogate objective and the original.

The method based on solving (4.6) resp. (4.5) is known as the **proximal point method**. The step is the **backward step**, or the **implicit step**, since we cannot in general derive an explicit solution $x = x^{i+1}$, and try to go "back to $x^i$ from $x^{i+1}$". However *often, and especially in context of splitting algorithms,* (4.6) *is easy to solve.* We will get back to this. By contrast, the gradient descent step (GD) is also known as the **forward step** or the **explicit step**, because we calculate $\nabla f(x^i)$ already at the current iterate, going "forward" from it.

Re-ordering as

$$x^i \in \tau\partial f(x^{i+1}) + x^{i+1},$$

the iteration resulting from the condition (4.6) may also be written as

$$x^{i+1} := (I + \tau\partial f)^{-1}(x^i), \tag{PP}$$

where the **proximal mapping**

$$\operatorname{prox}_{\tau\partial f} := (I + \tau\partial f)^{-1}$$

is the inverse of the set-valued map $A := I + \tau\partial f$, defined simply as

$$A^{-1}y := \{x \mid y \in Ax\}.$$

(Thus $y \in Ax \iff x \in A^{-1}y$.) As is evident from the expression

$$\operatorname{prox}_{\tau\partial f}(x) = \arg\min_{x'} f(x') + \frac{1}{2\tau}\|x' - x\|^2, \tag{4.7}$$

the proximal mapping is, in fact, single-valued.

Remark 4.2. Let $f_\tau := \min_{x'} f(x') + \frac{1}{2\tau}\|x' - x\|^2$. This is known as the **Moreau–Yosida regularisation** of $f$—a type of smoothing. In this way, the proximal step also corresponds to solving a sequence of smoothed problems.

Example 4.1. Let $f(x) = \|z - x\|_2^2/2$ for some $z \in \mathbb{R}^n$. By (4.7), we have $x' = \operatorname{prox}_{\tau\partial f}(x)$ if and only if $x \in \tau\partial f(x') + x'$. This gives the requirement $x = \tau(x' - z) + x'$. Consequently

$$\operatorname{prox}_{\tau\partial f}(x) = \frac{x + \tau z}{1 + \tau}.$$

Example 4.2. Let $f(x) = \delta_{[-1,1]}$ on $\mathbb{R}$. Then by (4.7), $x' = \operatorname{prox}_{\tau\partial f}(x)$ if and only if $x \in \tau N_{[-1,1]}(x') + x'$. Since $z \in N_C(x')$ implies $\tau z \in N_C(x')$ for any convex set $C$ and $\tau > 0$, this is to say $x \in x' + N_{[-1,1]}(x')$. Since

$$N_{[-1,1]}(x') = \begin{cases} [0,\infty), & x' = 1, \\ \{0\}, & x' \in (-1,1), \\ (-\infty,0], & x' = 1, \\ \emptyset, & \text{otherwise}, \end{cases}$$

it is not difficult to verify that

$$x' = x \cdot \min\{1, 1/|x|\} = \begin{cases} 1, & x > 1, \\ x, & x \in [-1,1], \\ -1, & x < -1. \end{cases}$$

In other words, the proximal mapping is the (Euclidean) projection of $x$ to $[-1,1]$. This is true in the general case $f(x) = \delta_C$, as is already evident from (4.7).

Exercise 4.1. *Calculate* $\operatorname{prox}_{\tau\partial f}$ *on* $\mathbb{R}^n$ *for*

*(i)* $f(x) = \delta_{\alpha B}(x)$, *where* $B$ *is the unit ball and* $\alpha > 0$.

*(ii)* $f(x) = \alpha\|x\|_2$.

Hint: *For (ii) you may find the next Exercise 4.2 useful.*

Exercise 4.2. *Suppose the convex function* $f(x) = \sup_{y\in\mathbb{R}^m} (\langle y, x\rangle - f^*(y))$ *for another proper convex lower semicontinuous function* $f^*$. *Prove* ***Moreau's identity***

$$y = \operatorname{prox}_{\tau\partial f^*}(y) + \tau \operatorname{prox}_{\tau^{-1}\partial f}(\tau^{-1}y). \tag{4.8}$$

Hint: *Use Theorem 3.1.*

The proximal point method (PP) readily generalises to solving for monotone operators $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ the monotone **variational inclusion**

$$0 \in A(x). \tag{MVI}$$

The method is simply

$$x^{i+1} := \operatorname{prox}_{\tau A}(x^i) = (I + \tau A)^{-1}(x^i). \tag{MPP}$$

Theorem 4.5. *Let $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be monotone, and suppose there exists a solution $\hat{x}$ to* (MVI)*. Then for any starting point $x^0 \in \mathbb{R}^n$, and any $\tau > 0$, the iterates $\{x^i\}_{i=0}^\infty$ of the proximal point method* (MPP) *converge to a solution of* (MVI).

*Proof.* We again use the Browder fixed point theorem, writing the iteration (MPP) in terms of the mapping $T := \operatorname{prox}_{\tau A}$. We have

$$Tx \in x - \tau A(Tx).$$

Thus

$$\|Tx - Ty\|^2 \in \langle Tx - Ty, x - y\rangle - \tau\langle Tx - Ty, A(Tx) - A(Ty)\rangle \leq \langle Tx - Ty, x - y\rangle.$$

In the latter step we have used the Cauchy–Schwarz inequality and the monotonicity of $A$. Thus $T$ is non-expansive, and the rest follows from Theorem 4.1. □

Corollary 4.1. *Suppose $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex and proper, and there exists a solution $\hat{x}$ to* (P)*. Then for any starting point $x^0 \in \mathbb{R}^n$, and any $\tau > 0$, the iterates $\{x^i\}_{i=0}^\infty$ of the proximal point method* (PP) *converge to a solution $\hat{x}$ of* (P).

## 4.4 Forward–backward splitting

Let us consider the minimisation of the composite objective

$$\min_{x \in \mathbb{R}^n} h(x) := g(x) + f(x), \tag{4.9}$$

where $g$ is smooth, but $f$ possibly non-smooth. By Theorem 2.1, we may write the optimality conditions as

$$0 \in \nabla g(x) + \partial f(x).$$

We can rewrite this as

$$\tau^{-1}x - \nabla g(x) \in \tau^{-1}x + \partial f(x),$$

or

$$x = (I + \tau\partial f)^{-1}(x - \tau\nabla g(x)).$$

This gives the iteration

$$x^{i+1} = \operatorname{prox}_{\tau\partial f}(x^i - \tau\nabla g(x^i)). \tag{FB}$$

In other words, we do a gradient/forward step with respect to $g$, and a proximal/backward step with respect to $f$. The resulting method is known as **forward–backward splitting**. Particular instances include the so-called **iterative soft-thresholding (IST)** algorithm for Lasso, with $\operatorname{prox}_{\tau\partial|\cdot|}$ known as the iterative soft-thresholding operator.

Exercise 4.3. *When does the method* (FB) *converge to a solution of* (4.9)*?* Hint: *You will need to use the second version of Browder's fixed point theorem.*

Example 4.3 (Forward–backward splitting for the SVM). We recall from Example 3.6 the dual form of the (linear) SVM, namely

$$\min_{y\in\mathbb{R}^n} g(y)+f(y),\quad g(y):=\frac{1}{2\lambda n}y^TA^TAy,\quad f(y):=\sum_{j=1}^n f_j(y_j),\quad f_j(y_j):=\delta_{[-b_j,0]}(y_j)+y_j/b_j,$$

where we recall that $[-b_j,0]:=[0,b_j]$ if $b_j<0$. To use the forward–backward splitting algorithm, we need to compute $y':=\operatorname{prox}_{\tau\partial f}(\widetilde{y})$. Clearly this splits as $y_j'=\operatorname{prox}_{\tau\partial f_j}(\widetilde{y}_j)$. From (4.7), we deduce that $y_j'\in[-b_j,0]$ has to satisfy

$$0\in y_j'-\widetilde{y}_j+\tau b_j^{-1}+\begin{cases}\{0\}, & y_j'\in(-b_j,0),\\ [0,\infty), & y_j'=\max\{0,-b_j\},\\ (-\infty,0], & y_j'=\min\{0,-b_j\}.\end{cases}$$

Proceeding as in Example 4.2, we see that $y_j'$ is the projection of $\widetilde{y}_j-\tau b_j^{-1}$ to $[-b_j,0]$. This can be written

$$y_j'=\operatorname{prox}_{\tau\partial f_j}(\widetilde{y}_j)=\begin{cases}\max\{-b_j,\min\{\widetilde{y}_j-\tau b_j^{-1},0\}\}, & b_j>0,\\ \max\{0,\min\{\widetilde{y}_j-\tau b_j^{-1},-b_j\}\}, & b_j<0\end{cases}$$

Consequently the forward–backward splitting algorithm (FB), namely $y^{i+1}:=\operatorname{prox}_{\tau\partial f}(y^i-\tau\nabla g(y^i))$, can with $b^{-1}=(b_1^{-1},\dots,b_n^{-1})$ be written

$$\begin{aligned}\bar{y}^{i+1}&:=y^i-\tau\Big(\frac{1}{\lambda n}A^TAy^i+b^{-1}\Big),\\ y_j^{i+1}&:=\begin{cases}\max\{-b_j,\min\{\bar{y}_j^{i+1},0\}\}, & b_j>0,\\ \max\{0,\min\{\bar{y}_j^{i+1},-b_j\}\}, & b_j<0,\end{cases}\quad\text{for each } j=1,\dots,n.\end{aligned}$$

Recall that in a non-linear SVM, the matrix $A^TA$ is replaced by the matrix $K$ with entries $\kappa(a_i,a_j)$.

Exercise 4.4. *Implement (FB) for the Lasso problem of Example 1.5. With your implementation, find the two most relevant physicochemical attributes for the quality of Portuguese* vinho verde, *according to the Wine Quality data set from the UCI machine learning repository at* http://archive.ics.uci.edu/ml/datasets/Wine+Quality. Note: *you will need to choose a stopping criterion for the algorithm. For the purposes of this exercise, it is sufficient to take a fixed number of iterations, let's say 1000.*

Exercise (Light) 4.5. *Express forward–backward splitting in terms of a surrogate objective.*

Exercise 4.6. *The total variation denoising problem (1.3) may be written in a dual form (cf. Chapter 3)*

$$\min_{\phi\in\mathbb{R}^{2n_1n_2}}\frac{1}{2}\|z-\widetilde{D}^T\phi\|^2,\quad \text{s.t.}\quad \sqrt{\phi_k^2+\phi_{n_1n_2+k}^2}\le\alpha\ \forall k=1,\dots,n_1n_2.$$

*Implement (FB) for this problem. Recall from Remark 3.3 that the solution of the original primal problem, the desired image, is $\hat{x}=z-\widetilde{D}^T\hat{\phi}$ for $\hat{\phi}$ the solution of the dual problem.*

## 4.5 (★) Douglas–Rachford splitting

Let $A : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be a general (set-valued) monotone operator, and $B : \mathbb{R}^n \to \mathbb{R}^n$ a single-valued monotone operator. Completely analogously to (4.9) and (FB), we can derive for the inclusion

$$B(x) + A(x) \ni 0 \tag{4.10}$$

the iteration

$$x^{i+1} \in \operatorname{prox}_{\tau A}(x^i - \tau B(x^i)).$$

However, this is not a very exactly defined method, as $B(x^i)$ can be set-valued, and therefore there can be many possibilities for $x^{i+1}$.

So, let us try to derive an improved method for (4.10). This will of course give an algorithm for (4.9) as well, through the choice $A = \partial f$ and $B = \partial g$. Picking $\lambda > 0$, let us set $z \in (I + \lambda B)(x)$. Then $\operatorname{prox}_{\lambda B}(z) = x$. Multiplying (4.10) by $\lambda$, and inserting this, we obtain

$$z + \lambda A(\operatorname{prox}_{\lambda B}(z)) \ni \operatorname{prox}_{\lambda B}(z).$$

This reorganises into

$$\operatorname{prox}_{\lambda B}(z) + \lambda A(\operatorname{prox}_{\lambda B}(z)) \in (2 \operatorname{prox}_{\lambda B} - I)(z),$$

and further into

$$\operatorname{prox}_{\lambda B}(z) = \operatorname{prox}_{\lambda A}((2 \operatorname{prox}_{\lambda B} - I)(z)).$$

This gives the fixed point equation

$$z = \operatorname{prox}_{\lambda A}((2 \operatorname{prox}_{\lambda B} - I)(z)) + (I - \operatorname{prox}_{\lambda B})(z).$$

Consequently, we derive the algorithm

$$z^{i+1} := \operatorname{prox}_{\lambda A}((2 \operatorname{prox}_{\lambda B} - I)(z^i)) + (I - \operatorname{prox}_{\lambda B})(z^i). \tag{4.11}$$

Note that this is for the transformed variable $z$, not our variable of interest $x$. To get a useful result, after the final step $i$, we therefore need to set

$$x^{i+1} := \operatorname{prox}_{\lambda B}(z^i). \tag{4.12}$$

Performing this at each step, and employing the result in (4.11), we may divide the algorithm into two distinct steps that are called the **Douglas–Rachford splitting algorithm**

$$x^{i+1} := \operatorname{prox}_{\lambda B}(z^i), \tag{DRS-0}$$
$$z^{i+1} := z^i + \operatorname{prox}_{\lambda A}(2x^{i+1} - z^i) - x^{i+1}. \tag{DRS-1}$$

Theorem 4.6 ([17, 18]). *Let $A, B : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be maximal monotone operators, and suppose there exists a solution $\hat{x}$ to $0 \in A(\hat{x}) + B(\hat{x})$. Then, for any $\lambda > 0$, and any starting point $z^0$, the iterates $\{x^i\}_{i=1}^\infty$ of the method (DRS-0)–(DRS-1) converge to a point $\widetilde{x}$ satisfying $0 \in A(\widetilde{x}) + B(\widetilde{x})$.*

In particular, since the convex subdifferential can be shown to be a maximal monotone operator, we have the following.

Corollary 4.2. *Let $f, g : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex, and suppose there exists a solution to the composite minimisation problem (4.9). Then, for any $\lambda > 0$, and any starting point $z^0$, the iterates $\{x^i\}_{i=1}^\infty$ of the method*

$$x^{i+1} := \operatorname{prox}_{\lambda \partial g}(z^i), \tag{DRS'-0}$$
$$z^{i+1} := z^i + \operatorname{prox}_{\lambda \partial f}(2x^{i+1} - z^i) - x^{i+1} \tag{DRS'-1}$$

*converge to a solution of (4.9).*

Exercise 4.7. *Implement the Douglas–Rachford splitting algorithm for dual form of total variation denoising, described in Exercise 4.6. How does the performance compare to basic forward–backward splitting?*
Note: *You will need to invert $I + \widetilde{D}^T\widetilde{D}$. For small images, you can simply employ sparse matrices and the slash operator in Matlab, but for bigger images it is beneficial use Fourier transform techniques, familiar from basic numerical analysis courses.*

Remark 4.3. The Douglas–Rachford splitting method (DRS-0)–(DRS-1), when applied to the operators $A := \partial[g^*(-K^T\,\boldsymbol{\cdot}\,)]$, and $B := \partial f^*$, is also known as the **Alternating Direction Method of Multipliers** (**ADMM**) for the solution of

$$\min_{x\in\mathbb{R}^n} g(x) + f(Kx), \tag{4.13}$$

In Exercise 4.7 we have, in fact, already implemented the ADMM for the TV denoising problem (1.3). Since a solution of (4.13) corresponds the condition $0 \in H(x, y)$ for $H$ as in (4.15), we have therefore finally, through splitting, found a practical variant for solving the latter problem.

## 4.6 The Chambolle–Pock method

Let us return to the saddle point problems of Chapter 3. That is, let us try to solve

$$\min_x \max_y \; g(x) + \langle Kx, y\rangle - f^*(y), \tag{4.14}$$

for some convex and proper $g : \mathbb{R}^n \to \overline{\mathbb{R}}$, and $f^* : \mathbb{R}^m \to \overline{\mathbb{R}}$, and some matrix $K \in \mathbb{R}^{m\times n}$. As we have seen in Chapter 3, the optimality conditions for this system are

$$-K^T\hat{y} \in \partial g(\hat{x}), \quad \text{and} \quad K\hat{x} \in \partial f^*(\hat{y}).$$

This may be encoded as $0 \in H(x, y)$ in terms of the monotone operator

$$H(x, y) := \begin{pmatrix} \partial g(x) + K^T y \\ \partial f^*(y) - Kx \end{pmatrix}. \tag{4.15}$$

In principle, we may therefore apply (MPP) to solve the saddle point problem (4.14). In practise we however need to work a little bit more, as the step (MPP) can rarely be given an explicit, easily solvable form.

However, there is a very effective primal–dual method for (4.14), that can be obtained from (MPP) with a small change. Let us first write out the algorithm, known as the **Chambolle–Pock method**, in explicit form. For parameters $\tau, \sigma > 0$, the primal variable $x$, and the dual variable $y$, we specifically iterate

$$x^{i+1} := (I + \tau\partial g)^{-1}(x^i - \tau K^T y^i), \tag{CP-0}$$
$$\bar{x}^{i+1} := 2x^{i+1} - x^i, \tag{CP-1}$$
$$y^{i+1} := (I + \sigma\partial f^*)^{-1}(y^i + \sigma K\bar{x}^{i+1}). \tag{CP-2}$$

The step (CP-0) is simply a proximal step for $x$ in (4.14), keeping $y = y^i$ fixed. The step (CP-2) is likewise a proximal step for $y$ in (4.14), keeping $x$ fixed, not to $x^i$ or $x^{i+1}$ but to the **inertial variable** $\bar{x}^{i+1}$ defined in (CP-1). This may be visualised as a “heavy ball” version of $x^{i+1}$ that has enough inertia to not get stuck in small bumps in the landscape.

With the general notation

$$u = (x, y),$$

the steps (CP-0)–(CP-2) may also be written in the **preconditioned** proximal point form

$$H(u^{i+1}) + M(u^{i+1} - u^i) \ni 0, \tag{4.16}$$

for the monotone operator $H$ as in (4.15), and the preconditioning matrix

$$M := \begin{pmatrix} I/\tau & -K^T \\ -K & I/\sigma \end{pmatrix}.$$

Through the replacement of $I$ by $M$ in the basic proximal point iteration $u^{i+1} := (I+H)^{-1}(u^i)$, we thus have in (CP-0)–(CP-1) a proximal point method for which the steps can often be solved explicitly.

Theorem 4.7. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ and $g : \mathbb{R}^m \to \overline{\mathbb{R}}$ be convex, proper, and lower semicontinuous, and $K \in \mathbb{R}^{m\times n}$. Choose $\tau, \sigma > 0$ such that $\tau\sigma\|K\|^2 < 1$. Let $u^* = (x^*, y^*)$ be a cluster point of the sequence of iterates $\{u^i\}$ generated by (CP-0)–(CP-2) for any starting point $u^0 = (x^0, y^0)$. Then $u^*$ is a saddle point of (4.14).*

*Proof.* A saddle point $\widehat{u}$ satisfies $0 \in H(\widehat{u})$. Therefore

$$\langle H(u^{i+1}) - H(\widehat{u}), u^{i+1} - \widehat{u}\rangle \geq 0.$$

Thus (4.16) gives

$$\langle M(u^{i+1} - u^i), u^{i+1} - \widehat{u}\rangle \leq 0. \tag{4.17}$$

With the notation $\|x\|_M := \sqrt{\langle Mx, x\rangle}$, we have

$$\langle M(u^{i+1} - u^i), u^{i+1} - \widehat{u}\rangle = \frac{1}{2}\|u^{i+1} - u^i\|_M^2 - \frac{1}{2}\|u^i - \widehat{u}\|_M^2 + \frac{1}{2}\|u^{i+1} - \widehat{u}\|_M^2.$$

Now (4.17) shows that

$$\frac{1}{2}\|u^{i+1} - \widehat{u}\|_M^2 + \frac{1}{2}\|u^{i+1} - u^i\|_M^2 \leq \frac{1}{2}\|u^i - \widehat{u}\|_M^2. \tag{4.18}$$

Summing (4.18) over $i = 0, \dots, N-1$ shows that

$$\frac{1}{2}\|u^N - \widehat{u}\|_M^2 + \sum_{i=0}^{N-1} \frac{1}{2}\|u^{i+1} - u^i\|_M^2 \leq \frac{1}{2}\|u^0 - \widehat{u}\|_M^2. \tag{4.19}$$

Now, the condition $\tau\sigma\|K\|^2 < 1$ ensures that $\|u\|_M^2 \geq \theta\|u\|^2$ for some $\theta > 0$. Therefore (4.19) shows that $\|u^{i+1} - u^i\| \to 0$, and that $\{u^i\}_{i\in\mathbb{N}}$ is bounded. Therefore, every subsequence $\{u^{i_j}\}_{j\in\mathbb{N}}$ has a further subsequence that converges to some point $u^*$ satisfying $0 \in H(u^*)$. In particular, every cluster point is a saddle point. □

Exercise 4.8. *Using Opial's lemma below, show that there is, in fact, only one cluster point. Show, therefore, that the whole sequence of iterates converges to a saddle point.*

Opial's lemma: *Let $C \subset \mathbb{R}^n$ be closed and convex, and $\{u^i\}_{i\in\mathbb{N}} \subset \mathbb{R}^n$. If the following conditions hold, then $u^i \rightharpoonup u^*$ for some $u^* \in C$:*

*(i) $i \mapsto \|u^i - u^*\|_M$ is non-increasing for all $u^* \in C$.*

*(ii) All limit points of $\{u^i\}_{i\in\mathbb{N}}$ belong to $C$.*

Example 4.4 (Dualisation trick for hard-to-invert forward operators). As we have seen in Example 4.1, the proximal mapping of $g(x) = \|z - x\|_2^2/2$ is easy to calculate. But what about $g(x) = \|z - Ax\|_2^2/2$ for some $A \in \mathbb{R}^{k \times n}$ and $z \in \mathbb{R}^k$? Unless $A$ is unitary (i.e., $A^T A = I$, such as a Fourier transform), the computation of $\operatorname{prox}_{\tau\partial f}$ will generally require a costly matrix inversion. However, we can also use the *dualisation trick*

$$g(x) = \sup_{\lambda \in \mathbb{R}^k} \langle Ax - z, \lambda \rangle - \frac{1}{2}\|\lambda\|^2,$$

and replace the saddle point problem

$$\min_x \max_y \; g(x) + \langle Kx, y \rangle - f^*(y)$$

by

$$\min_x \max_{\widetilde{y}} \; \widetilde{g}(x) + \langle \widetilde{K}x, \widetilde{y} \rangle - \widetilde{f}^*(\widetilde{y}),$$

where $\widetilde{y} = (y, \lambda)$ and the mappings

$$\widetilde{g}(x) = 0, \quad \widetilde{f}^*(\widetilde{y}) = f^*(y) + \frac{1}{2}\|\lambda\|^2 + \langle z, \lambda \rangle, \quad \text{and} \quad \widetilde{K}x = (Kx, Ax).$$

Exercise 4.9. *Implement the Chambolle–Pock method for total variation denoising, described in Exercise 4.6. What is the effect of the choice of $\tau$ and $\sigma$? How does the performance compare to forward–backward splitting?*

Remark 4.4. Various further splitting algorithms exist in the literature, many of which are closely linked to each other. The Chambolle–Pock method and forward–backward splitting can also be accelerated, to obtain fast convergence rates on strongly convex problems [19–21]. There also exist stochastic variants of all our algorithms, which allow very large problems—Big Data problems—to be split on computing clusters with reduced communication needs; see, e.g., [22, 23].

# 5 Practical segmentation

We recall the Mumford–Shah image segmentation problem (1.11), written as

$$\min_{x \in \mathbb{R}^{n_1 n_2}, \Gamma} \frac{1}{2}\|z - x\|^2 + \alpha \mathrm{MS}_\theta(x, \Gamma) \tag{5.1}$$

for the regulariser

$$\mathrm{MS}_\theta(x, \Gamma) := \frac{1}{2}\|\widetilde{D}x|\Omega_0\|_2^2 + \frac{1}{2}\|\widetilde{D}x|\Omega_1\|_2^2 + \theta \cdot \mathrm{length}(\Gamma),$$

with $\Gamma$ the boundary of foreground image region $\Omega_1 \subset \Omega$, and $\Omega_0 = \Omega \setminus \Omega_1$ the background image region within our image domain

$$\Omega = \{1, \ldots, n_1\} \times \{1, \ldots, n_2\} \sim \{1, \ldots, n_1 n_2\}.$$

(We equate these two and one-dimensional ways to index the image domain, cf. Figure 1.1.) This problem is difficult to solve. Especially the length-term is highly non-convex.

The length of the boundary is, of course, not clearly defined for discrete pixelised images. For our purposes, we set

$$\mathrm{length}(\Gamma) := \|\widetilde{D}\phi_{\Omega_1}\|_{2,1},$$

where the vector presentation $\phi_{\Omega_1} \in \mathbb{R}^{n_1 n_2}$ of $\Omega_1$ is given by the components

$$\phi_{\Omega_1, j} := \begin{cases} 1, & \text{pixel } j \text{ is contained in } \Omega_1, \\ 0, & \text{pixel } j \text{ is not contained in } \Omega_1. \end{cases} \tag{5.2}$$

Recalling Figure 1.3, we see that this gives a reasonable definition of the length. Indeed, through limiting arguments, we can see our definition to be better than counting the length of the geometric boundary of the pixels as squares, cf. [24].

## 5.1 Convex relaxation of the Mumford–Shah problem

An idea to simplify (5.1) is to first of all replace the foreground region $\Omega_1$ by a "lifting" $\phi \in \{0, 1\}^{n_1 n_2}$. We then define

$$\Omega_1 := \{j \in \Omega \mid \phi_j = 1\}, \quad \text{and} \quad \Omega_0 := \{j \in \Omega \mid \phi_j = 0\}.$$

Because $[\widetilde{D}x|\Omega_1]_j$ only depends on pixels in $\Omega_1$, and $[\widetilde{D}x|\Omega_0]_j$ only depends on pixels in $\Omega_0$, we may then write

$$\frac{1}{2}\|\widetilde{D}x|\Omega_1\|_2^2 = \frac{1}{2}\sum_{j \in \Omega} \phi_j \|[\widetilde{D}x|\Omega_1]_j\|_2^2, \quad \text{and} \quad \frac{1}{2}\|\widetilde{D}x|\Omega_0\|_2^2 = \frac{1}{2}\sum_{j \in \Omega}(1 - \phi_j)\|[\widetilde{D}x|\Omega_0]_j\|_2^2.$$

Equivalently to (5.1), we may therefore solve

$$\min_{x \in \mathbb{R}^{n_1 n_2}, \phi \in \{0,1\}^{n_1 n_2}} \frac{1}{2}\|z - x\|^2 + \alpha \widetilde{\mathrm{MS}}_\theta(x, \phi) \tag{5.3}$$

for the regulariser

$$\widetilde{\mathrm{MS}}_\theta(x, \phi) := \frac{1}{2}\sum_{j \in \Omega} \phi_j \|[\widetilde{D}x|\Omega_0]_j\|_2^2 + \sum_{j \in \Omega}(1 - \phi_j)\|[\widetilde{D}x|\Omega_1]_j\|_2^2 + \theta\|\widetilde{D}\phi\|_{2,1},$$

If we assume $x$ to be a constant $c_1$ on the foreground $\Omega_1$, and a constant $c_0$ on the background $\Omega_0 := \Omega \setminus \Omega_1$, we can write

$$x = x_\phi := c_0 + (c_1 - c_0)\phi$$

as well as

$$\widetilde{\mathrm{MS}}_\theta(x_\phi, \phi) = \theta\|\widetilde{D}\phi\|_{2,1}.$$

In fact, let us define regulariser

$$\mathrm{CV}_\theta(\phi) = \theta\|\widetilde{D}\phi\|_{2,1} + \sum_{j\in\Omega} \phi_j. \tag{5.4}$$

The additional last term here is simply area$(\Omega_1)$, so penalises large foreground objects. All of these changes applied to (5.3) yields the problem

$$\min_{c_0, c_1 \in \mathbb{R}; \phi\in\{0,1\}^{n_1 n_2}} \frac{1}{2}\|z - x_\phi\|^2 + \alpha \mathrm{CV}_\theta(\phi). \tag{5.5}$$

We can also expand

$$\begin{aligned}\frac{1}{2}\|z - x_\phi\|^2 &= \frac{1}{2}\sum_{j\in\Omega}(z_j - c_0)^2\phi_j + \frac{1}{2}\sum_{j\in\Omega}(z_j - c_1)^2(1 - \phi_j)\\ &= \frac{1}{2}\sum_{j\in\Omega}\left((z_j - c_0)^2 - (z_j - c_1)^2\right)\phi_j + \frac{1}{2}\sum_{j\in\Omega}(z_j - c_1)^2\\ &= \langle r, \phi\rangle + \frac{1}{2}\sum_{j\in\Omega}(z_j - c_1)^2,\end{aligned} \tag{5.6}$$

where $r \in \mathbb{R}^{n_1 n_2}$ has components defined by

$$r_j := \frac{1}{2}\left((z_j - c_0)^2 - (z_j - c_1)^2\right).$$

Thus the segmentation problem (5.5) can with (5.6) be equivalently presented as

$$\min_{c_0, c_1 \in \mathbb{R}; \phi\in\{0,1\}^{n_1 n_2}} \langle r, \phi\rangle + \alpha\theta\|\widetilde{D}\phi\|_{2,1} + \alpha\sum_{j\in\Omega}\phi_j + \frac{1}{2}\sum_{j\in\Omega}(z_j - c_1)^2. \tag{5.7}$$

The problem (5.7) is still non-convex and non-smooth. To simplify it further, we first of all observe that by taking the first-order optimality conditions of (5.8) with respect to $c_0$ and $c_1$, we can solve explicitly

$$c_1 = \frac{\sum_{j\in\Omega} z_j\phi_j}{\sum_{j\in\Omega}\phi_j} \quad\text{and}\quad c_0 = \frac{\sum_{j\in\Omega} z_j(1 - \phi_j)}{\sum_{j\in\Omega}(1 - \phi_j)}$$

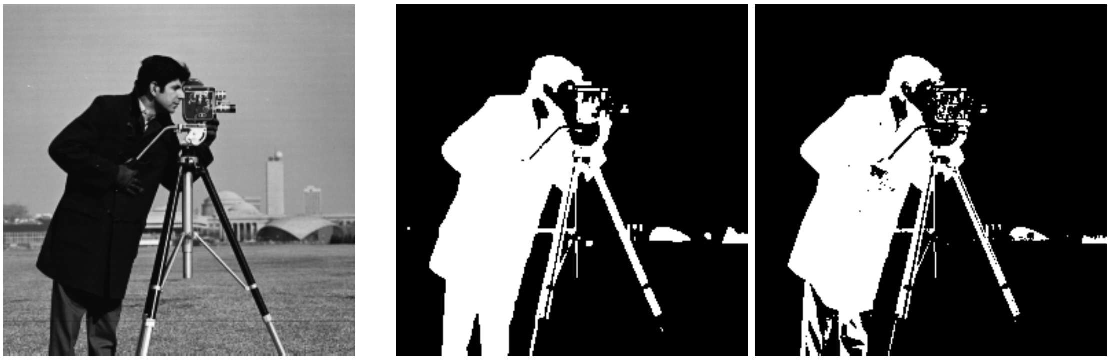
Thus $c_1$ is the average of $z$ on $\Omega_0$, and $c_0$ is the average of $z$ on $\Omega_1$. Making fixed a priori guesses about the average intensities $c_0$ and $c_1$ fixed, we can convert (5.7) to

$$\min_{\phi\in\{0,1\}^{n_1 n_2}} \langle r + \alpha, \phi\rangle + \alpha\theta\|\widetilde{D}\phi\|_{2,1}. \tag{5.8}$$

This is still highly non-convex due to the constraint $\phi \in \{0,1\}^{n_1 n_2}$. Observe that so far, the only real transformative change we have made to (1.11) is the assumption of the constant known intensities $c_0$ and $c_1$. We now make the bigger change of relaxing $\phi \in [0,1]^{n_1 n_2}$. We then obtain the convex problem

$$\min_{\phi\in\mathbb{R}^{n_1 n_2}} \langle r + \alpha, \phi\rangle + \delta_{[0,1]^{n_1 n_2}}(\phi) + \alpha\theta\|\widetilde{D}\phi\|_{2,1}. \tag{5.9}$$

This is the improved formulation of **Chan–Vese segmentation** [25] derived in [26]. Generally, this type of approaches that lift a set to an indicator vector, and then relax the zero–one constraints on the vector, are known as **level-set methods**.



**(a)** Image **(b)** Chan–Vese segmentation **(c)** Intensity thresholding

**Figure 5.1:** Demonstration of Chan–Vese segmentation versus simple intensity thresholding (select as $\Omega_1$ those pixels with high enough intensity). The Chan–Vese segmentation has much smoother boundaries and is lacking speckle-like artefacts.

Our algorithms from Chapter 4, in particular Chambolle–Pock, are applicable to (5.9). As a solution foreground object we can take

$$\Omega_1 := \{j \in \Omega \mid \phi_j > 1/2\}.$$

This is compared against simple intensity thresholding in Figure 5.1. Although not perfect, it provides a much smoother uniform segmentation than intensity thresholding. It does, however select also objects other than the camera man of interest. This is because (5.9) is a **global segmentation** model. Only single objects of interest demand **local segmentation**. Models based on (5.8) are discussed, for example, in [27].

Remark 5.1 (Alternative relaxation approaches). There are other, more general, convex relaxation approaches for segmentation approaches that work slightly differently to the one presented here. The fundamental idea is, however, the same: lift the problem to a higher-dimensional space. Here, we replace the set $\Omega_1$, which can be seen as an element of $\{0,1\}^{n_1 n_2}$ by the vector $\phi \in \mathbb{R}^{n_1 n_2}$. In the alternative approaches, we work with $u$ a **function of bounded variation** (see [28]). In that space, $u$ can encode $\Gamma$ as a jump in the function. Then we lift $u$ to a **measure**, and discretise this measure in such a way that any non-convexity is hidden into pre-computable values in the discretisation [29].

Exercise 5.1. *Implement the Chambolle–Pock algorithm for* (5.9). *Test it on some images with clear foreground objects, adjusting the foreground and background intensities* $c_0$ *and* $c_1$ *as well as the segmentation parameters* $\theta$ *and* $\alpha$ *by educated guesses to yield a good result.*
Hint: *Similarly to TV denoising, convert the term* $\alpha\theta\|\widetilde{D}v\|_{2,1}$ *into* $\max_y \langle v, \widetilde{D}y\rangle - F^*(y)$ *for a dual variable* $y$ *and a suitable constraint modelled by* $F^*$.

## 5.2 Dictionary learning

Regularisers do not have to be entirely analytically constructed. They model prior information of a good solution, and therefore constructing them from data provides a good alternative. In particular, for segmentation, we might want to construct the regularisers from known poses of a known object that we want to detect. Thus the task of deciding a regularisers becomes a task of **machine learning**. One simple and for segmentation proven approach is principal component analysis, PCA [30, 31].

So let $\phi_1, \ldots, \phi_N \in [0,1]^{n_1 n_2}$ be a our sample or “training” shapes on the domain $\Omega$, defined using the level set approach (5.2). Let $\mu := \frac{1}{N}\sum_{i=1}^N \phi_N$ be the average shape. We can also arrange the shapes into a matrix

$$M = (\phi_1 - \mu, \ldots, \phi_n - \mu).$$

Singular value decomposition then gives $A = (a_1, \dots, a_N)$, where the $a_n$ are orthonormal column vectors, and $D = \operatorname{diag}(d_1, \dots, d_N)$ such that $ADA^T = \frac{1}{N} MM^T$. We assume without loss of generality that $d_1 \geq d_2 \geq \dots \geq d_N$. We then denote the first $n \ll N$ columns of $A$ by $A_n = (a_1, \dots, a_n)$, and set $D_n := \operatorname{diag}(d_1, \dots, d_n)$. These will be our "dictionary" of "shape variations".

We then replace (5.4) by

$$R(\phi) = \frac{1}{2} \| D_n^{1/2} A_n^T (\phi - \mu) \|_2^2 + \delta_{\mathcal{R}(A_n)+\mu}(\phi) = \sum_{j=1}^{n} d_j \langle a_j, \phi - \mu \rangle^2 + \delta_{\mathcal{R}(A_n)+\mu}(\phi),$$

The indicator function merely means that

$$\phi \in \left\{ \phi_\beta := \mu + A_n \beta = \mu + \sum_{i=1}^{n} \beta_i a_i \,\middle|\, \beta \in \mathbb{R}^n \right\}. \tag{5.10}$$

The vector $\beta$ is our coefficient vector for constructing $\phi$ out of the dictionary of shape variations. Given the presentation (5.10), we get

$$R(\phi) = \sum_{j=1}^{n} d_j \|a_j\|^2 \beta_j = \frac{1}{2} \langle D_n \beta, \beta \rangle.$$

Thus it makes sense to look for $\beta$ instead of $\phi$, and solve

$$\min_{\beta \in \mathbb{R}^n} \frac{1}{2} \| z - x_{\phi_\beta} \|^2 + \frac{1}{2} \langle D_n \beta, \beta \rangle. \tag{5.11}$$

Following Section 5.1, we convert this into

$$\min_{\beta \in \mathbb{R}^n} \langle r, \mu + A_n \beta \rangle + \frac{1}{2} \langle D_n \beta, \beta \rangle.$$

This is again a very simple convex problem.

# Bibliography

[1] J. Nocedal & S. Wright, *Numerical Optimization*, Springer, 2006.

[2] K. Bredies, K. Kunisch & T. Pock, *Total Generalized Variation*, SIAM Journal on Imaging Sciences **3** (2011).

[3] J. Shen, S. Kang & T. Chan, *Euler's Elastica and Curvature-Based Inpainting*, SIAM Journal on Applied Mathematics **63** (2003).

[4] J. Huang & D. Mumford, *Statistics of natural images and models*, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999.

[5] H. Engl, M. Hanke & A. Neubauer, *Regularization of Inverse Problems*, Springer, 2000.

[6] T. Chan & J. Shen, *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*, Society for Industrial & Applied Mathematics (SIAM), 2005.

[7] G. Aubert & P. Kornprobst, *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Springer, 2006.

[8] K. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012.

[9] B. Schölkopf & A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2002.

[10] J.-B. Hiriart-Urruty & C. Lemaréchal, *Convex analysis and minimization algorithms I-II*, Springer, 1993.

[11] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, 1972.

[12] I. Ekeland & R. Temam, *Convex analysis and variational problems*, SIAM, 1999.

[13] H. Bauschke & P. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, 2011.

[14] H. Attouch & H. Brezis, *Duality for the Sum of Convex Functions in General Banach Spaces*, in: *Aspects of Mathematics and its Applications*, vol. 34, North-Holland, Elsevier, 1986, 125–133.

[15] F. Browder & W. Petryshyn, *Construction of fixed points of nonlinear mappings in Hilbert space*, Journal of Mathematical Analysis and Applications **20** (1967).

[16] F. E. Browder, *Nonexpansive nonlinear operators in a Banach space*, Proceedings of the National Academy of Sciences of the United States of America **54** (1965).

[17] P. L. Lions & B. Mercier, *Splitting Algorithms for the Sum of Two Nonlinear Operators*, SIAM Journal on Numerical Analysis **16** (1979).

[18] J. Eckstein & D. Bertsekas, *On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators*, Mathematical Programming **55** (1992).

[19] A. Chambolle & T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of Mathematical Imaging and Vision **40** (2011).

[20] A. Beck & M. Teboulle, *A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems*, SIAM Journal on Imaging Sciences **2** (2009).

[21] T. Valkonen & T. Pock, *Acceleration of the PDHGM on partially strongly convex functions*, Journal of Mathematical Imaging and Vision (2016).

[22] S. Wright, *Coordinate descent algorithms*, Mathematical Programming **151** (2015).

[23] T. Valkonen, *Block-proximal methods with spatially adapted acceleration*, Submitted, 2016, arXiv:1609.07373.

[24] E. Casas, K. Kunisch & C. Pola, *Regularization by Functions of Bounded Variation and Applications to Image Enhancement*, Applied Mathematics & Optimization **40** (1999).

[25] T. F. Chan & L. A. Vese, *Active contours without edges*, IEEE Transactions on Image Processing **10** (2001).

[26] T. F. Chan, S. Esedoglu & M. Nikolova, *Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models*, SIAM Journal on Applied Mathematics **66** (2006).

[27] J. Spencer & K. Chen, *Global and Local Segmentation of Images by Geometry Preserving Variational Models and Their Algorithms*, in: *Forging Connections between Computational Mathematics and Computational Geometry: Papers from the 3rd International Conference on Computational Mathematics and Computational Geometry*, Springer International Publishing, 2016, 87–105.

[28] T. Valkonen, *Measure and Image*, Lecture Notes, 2013.

[29] T. Pock et al., *A convex relaxation approach for computing minimal partitions*, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, 810–817.

[30] M. E. Leventon, W. E. L. Grimson & O. Faugeras, *Statistical shape influence in geodesic active contours*, in: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, 2000, 316–323.

[31] D. Cremers, *Dynamical statistical shape priors for level set-based tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence **28** (2006).