B6.2 Optimisation for Data Science

Lecture Notes, Lectures 1–8

Oxford Mathematical Institute, HT 2022

Prof. Raphael Hauser

March 10, 2022

Table of Contents

1 Scope of this Course

1.1 Optimisation Models

Models considered in this course can take any of the following forms.

Unconstrained model

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

where f is *smooth*, which in the context of this course means C^1 with Lipschitz continuous gradient.

Regularised model

$$\min_{x\in\mathbb{R}^n} f(x) + \lambda\Psi(x)$$

$\Psi : \mathbb{R}^n \to \mathbb{R}$ convex, possibly nonsmooth, controls the complexity and structure of the optimal solution x^* (a point where the *objective function* $f(x)+\lambda\Psi(x)$ achieves a minimum). λ is called the *regularisation parameter*.

Constrained model

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{subject to } x \in \mathcal{F}$$

e.g., $\mathcal{F} = \{x \in \mathbb{R}^n : g(x) \leq 0, x \geq 0\}$ for some $g : \mathbb{R}^n \to \mathbb{R}^q$ (all vector inequalities to be interpreted componentwise, i.e. $g(x) \leq 0$ means $g_s(x) \leq 0$ $(s = 1, \dots q)$).

1.2 Sparse Objective

In data analysis the objective function $f(x)$ often takes the form of a sum

$$f(x) = \sum_{j=1}^{m} f_j(x),$$

where each f_i is *sparse* (depends only on a few of the coordinates of x), and since $n, m \gg 1$, often only gradients $\nabla f(x)$ or $\nabla f_i(x)$ are available, but no higher derivatives.

1.3 First Order Methods

First order methods are iterative algorithms designed to produce a sequence of *solutions* (points $x \in \mathbb{R}^n$ that satisfy the constraints $x \in \mathcal{F}$) $(x^k)_{k\in\mathbb{N}} \to x^*$ that converges to an optimal solution based on updates

$$x^{k+1} = x^k + \alpha_k d^k$$

with *search direction* d^k computed from $\nabla f(x)$ or $\nabla f_i(x)$, and a step length $\alpha_k > 0$ that is either fixed or computed iteratively.

1.4 Data Analysis Problems

Data analysis problems typically combine the following elements:

Data set $D := \{(a_j, y_j) : j = 1, \dots, m\}$, where $a_j \in V$ lies in a vector space of *features*, and $y_j \in W$ in a space of *observations*.

Parametric model $\Phi(\cdot, x) : V \to W$ of a feature-observation relation, parameterised by a vector $x \in \mathbb{R}^n$.

Data fitting problem: find $x \in \mathbb{R}^n$ such that $\Phi(a_j; x) \approx y_j$ for all j by solving $\min_{x \in \mathbb{R}^n} f(x)$, with objective

$$f(x) = L_D(x) := \sum_{j=1}^{m} \ell(a_j, y_j; x).$$

$L_D(x)$ is the *loss* on the data set D associated with parameter values x. Typical example $\ell(a_j, y_j; x) = \|\Phi(a_j; x) - y_j\|^2$.

2 Motivating Examples

Example 1. Examples of observation sets:

i) $W = \mathbb{R}$ regression problems.

ii) $W = \{1, \dots, M\}$ classification problems.

iii) $W = \emptyset$ data clustering, structured dimensionality reduction.

Example 2 (Regression).

i) *Regression without intercept*: Choose $V = \mathbb{R}^n$, $W = \mathbb{R}$, $\Phi(a; x) = a^{\mathrm{T}} x$. Data fitting model

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \sum_{j=1} m(a_j^{\mathrm{T}} x - y_j)^2 = \frac{1}{2m} \|Ax - y\|^2,$$

where $A = \begin{pmatrix} a_1 & \dots & a_m \end{pmatrix}^{\mathrm{T}}$.

ii) *Regression with intercept*: Same formulation, but replace x and a_j by

$$\tilde{x} = \begin{pmatrix} x \\ \beta \end{pmatrix}, \quad \tilde{a}_j = \begin{pmatrix} a_j \\ 1 \end{pmatrix},$$

so that $\Phi(\tilde{a}; \tilde{x}) = \tilde{a}^{\mathrm{T}} x + \beta$.

iii) *Tychonov-regularised regression*: this method uses the data fitting model

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

which leads to robust regression, less sensitive to noise in the data (a_j, y_j).

iv) *Lasso-regression*: the data fitting model

$$\min_{x\in\mathbb{R}^n} \frac{1}{2m}\|Ax-y\|^2+\lambda\|x\|_1$$

tends to yield sparse optimal solution x^* (feature selection) if one exists.

v) *Dictionary learning*: Extending linear regression to learning the regressors. In dictionary learning we seek a dictionary $A \in \mathbb{R}^{m\times n}$ and sparse regressors $X \in \mathbb{R}^{n\times p}$ so that the data $Y \in \mathbb{R}^{m\times p}$ can be approximately expressed as $AX \approx Y$. We use $V = (\mathbb{R}^{m\times n}, \mathbb{R}^{n\times p})$ as data space and $W = \mathbb{R}^{1\times p}$ as observation space, with data $D = \{y_\ell\}_{\ell=1}^m$, where $y_\ell \in \mathbb{R}^{1\times p}$ and $p \gg n$. The data fitting model is designed as

$$\min_{A\in\mathbb{R}^{m\times n}, X\in\mathcal{F}_k(m,p)} \frac{1}{2mp}\|AX-Y\|_F^2,$$

where

$$\mathcal{F}_k(m,p) = \left\{ X\in\mathbb{R}^{n\times p} : \sum_{i=1}^n |x_{i,j}\neq 0| \le k \quad \forall\, j=1,\dots,p \right\},$$

indicating only k of the the learned regressors from A are used to approximate each column of data in Y.

Example 3. *Matrix completion*: Choose $V = \mathbb{R}^{n\times p}$, $W = \mathbb{R}$, $D = \{(A_j, y_j) : j = 1,\dots,m\}$. Define the *trace inner product* on V,

$$\begin{aligned}\langle\cdot,\cdot\rangle : V\times V &\to \mathbb{R}\\ (X,Y) &\mapsto \operatorname{tr}(X^{\mathrm{T}}Y).\end{aligned}$$

The data fitting problem is given by

$$\min_{X\in V} \frac{1}{2m}\sum_{j=1}^m \left(\langle A_j, X\rangle - y_j\right)^2 + \lambda\|X\|_*,$$

where $\langle A_j, X\rangle$ models the *sensing* of X, y_j the *target value*, $\lambda > 0$, and the *nuclear norm* $\|X\|_* = \sum_{i=1}^p \sigma_i$ is the sum of the singular values of X. Regularisation by the nuclear norm tends to yield a low-rank optimal solution if one exists. Typical choice $A_j = [a_{s,t}]$ with

$$a_{s,t} = \begin{cases} 1 & \text{if } s = s_j,\ t = t_j,\\ 0 & \text{otherwise}\end{cases}$$

for fixed s_j, t_j, so that $\langle A_j, X\rangle = x_{s_j,t_j}$ and y_j correspond to a few known entries of X. This problem may be seen as a sophisticated form of regression.

Example 4. *Sparse inverse covariance estimation*: $V = \mathbb{R}^n$, $W = \emptyset$, a_j i.i.d. samples of a random vector $\Omega \to \mathbb{R}^n$ with $\mathrm{E}[A] = 0$ and $\mathrm{Cov}(A) = \mathrm{E}[AA^{\mathrm{T}}] = \Sigma < \infty$ (all covariances are finite). The sample estimate of Σ is $S = \frac{1}{m}\sum_{j=1}^m a_j a_j^{\mathrm{T}}$. In many applications, the components of A are *conditionally independent*,

$$\mathscr{D}(A_j|A_i, \{A_k : k \neq i, j\}) = \mathscr{D}(A_j|\{A_k : k \neq i, j\}).$$

For example, when

$$A_i \overset{\text{causality}}{\to} A_k \overset{\text{causality}}{\to} A_j,$$

then $\mathrm{Cov}(A_i, A_j) \neq 0$ but $\mathscr{D}(A_j|A_i, A_k) = \mathscr{D}(A_j|A_k)$. *Fact:* If A_i, A_j are conditionally independent, then $(\Sigma^{-1})_{i,j} = 0$ (component (i,j) of Σ^{-1} is zero). Therefore, we correct the sample estimate S to that its inverse becomes sparse by solving the data fitting problem

$$\min_{X \in \mathbb{SR}^{n\times n}, X \succeq 0} \langle S, X\rangle - \log\det X + \lambda\|X\|_1,$$

where $SR^{n\times n}$ is the set of $n \times n$ real symmetric matrices, and $X \succeq 0$ means X is positive semidefinite. In this example we extract structure from the data without any measurement, that is, $W = \emptyset$ and

$$\begin{aligned} L_D(X) &= \langle S, X\rangle - \log\det X + \lambda\|X\|_1 \\ &= \frac{1}{m}\sum_{j=1}^m \langle a_j a_j^{\mathrm{T}}, X\rangle - \log\det X + \lambda\|X\|_1 \\ &= \frac{1}{m}\sum_{j=1}^m a_j^{\mathrm{T}} X a_j^{\mathrm{T}} - \log\det X + \lambda\|X\|_1 \\ &= -\frac{2}{m}\log\left(\prod_{j=1}^m (2\pi)^{-n/2}\det(X^{-1})^{-1/2}\exp\left\{-\frac{1}{2}a_j^{\mathrm{T}} X a_j\right\}\right) - n\log(2\pi) \\ &\quad + \lambda\|X\|_1, \end{aligned}$$

so that the first two terms of the objective represent a shift plus a negative multiple of the log-likelihood function for $X = \mathrm{Cov}(A^{-1})$ under the parametric model that the data a_j are i.i.d. samples of mean centred multivariate Gaussian vectors. Note also that the barrier term $-\log\det X$ is used to keep X away from boundary and hence invertible. The Lasso term $\lambda\|X\|_1$ incentivises a sparse solution that allows one to identify the conditional dependencies between the variables if the reconstruction is successful.

Example 5 (Principal Component Analysis).

i) *Robust PCA*: We want to render classical PCA (see prelims course on data analysis and statistics) robust to data outliers. Data space $V =$

$(\mathbb{R}^{m\times n}, \mathbb{R}^{m\times n})$, observation space $W = \mathbb{R}$, data $D = Y \in \mathbb{R}^{m\times n}$, the entire data matrix. The data fitting model is chosen as

$$\min_{L,S\in\mathbb{R}^{m\times n}} \|Y - (L+S)\|_F^2 + \lambda\|L\|_* + \gamma\|S\|_1,$$

where the nuclear norm $\|L\|_* = \sum_k \sigma_k$ encourages a low rank component L and $\|S\|_1 = \sum_{i,j} |S_{i,j}|$ encourages a sparse component S.

The weights λ and γ trade off the rank of L, sparsity of S, and how well their sum matches the data Y. The sparse component S can be viewed as outlier errors in Y.

ii) *Sparse principal compnent analysis*: The first component of the PCA of a matrix $S \succeq 0$ is computed as follows,

$$\max_{v\in\mathbb{R}^n} v^{\mathrm{T}} S v, \quad \text{subject to } \|v\|_2 = 1.$$

A sparse 1st principal component is defined in principle by

$$\max_{v} v^{\mathrm{T}} S v, \quad \text{subject to } \|v\|_2 = 1,\ \|v\|_0 \le k,$$

where $\|v\|_0 := |\{i : v_i \neq 0\}|$ is the number of nonzero compoents of v, with $k < n$. Since the latter problem ins NP hard, we replace it by its convex relaxation,

$$\max_{M\in\mathbb{SR}^n} \langle S, M\rangle, \quad \text{subject to } M \succeq 0, \langle \mathrm{I}, M\rangle = 1,\ \|M\|_1 \le \rho,$$

where $\langle S, M\rangle$ replaces $v^{\mathrm{T}} S v = \mathrm{tr}(S, vv^{\mathrm{T}})$ and $\langle \mathrm{I}, M\rangle = 1$ replaces $1 = v^{\mathrm{T}} v = \langle \mathrm{I}, vv^{\mathrm{T}}\rangle$.

Example 6 (Data Separation).

i) *Support vector machines*: $V = \mathbb{R}^n$, $W = \{+1, -1\}$, problem of classifying data into two classes. The ansatz is to seek a separating hyperplane $\{a \in \mathbb{R}^n : a^{\mathrm{T}} x - \beta = 0\}$ such that

$$y_j \times (a_j^{\mathrm{T}} x - \beta) \ge 1. \tag{1}$$

We would like to use $\Phi(a; x, \beta) := \mathrm{sign}(a_j, x - \beta)$ and recall that we want $\Phi(a; x, \beta) = y_j$ for as many data points j as possible. To avoid having to solve an NP-hard problem and to find a robust solution, we maximise the separation (margin) between the two sets, which is obtained by solving

$$\min_{(x,\beta)} \|x\|_2^2, \quad \text{subject to (1)}. \tag{2}$$

But (2) may not have any solution (x, β) at all that satisfies (1). Therefore, solve the following data fitting model instead,

$$\begin{aligned}
&\min_{(x,\beta)} \frac{1}{m}\sum_{j=1}^{m} \max\left(1 - y_j \times (a_j^{\mathrm{T}} x - \beta), 0\right) + \frac{\lambda}{2}\|x\|_2^2 \\
&\quad \Leftrightarrow \min_{x\in\mathbb{R}^n, \beta\in\mathbb{R}, s\in\mathbb{R}^m} \frac{1}{m}\sum_{j=1}^{m} s_j + \frac{\lambda}{2}\|x\|_2^2 \\
&\qquad \text{subject to } s_j \geq 1 - y_j(a_j^{\mathrm{T}} x - \beta), \quad s_j \geq 0, \quad (j = 1, \ldots, m).
\end{aligned}$$

ii) *Nonlinear separation*: the idea is to use a nonlinear transformation $\zeta : \mathbb{R}^n \to \tilde{V}$, where $(\tilde{V}, \langle\cdot,\cdot\rangle)$ is a Hilbert space. We replace (1) by

$$y_j \times (\langle \zeta(a_j), x\rangle - \beta) \geq 1 \tag{3}$$

and solve the data fitting model

$$\begin{aligned}
&\min_{x\in\tilde{V}, \beta\in\mathbb{R}} \frac{1}{m}\sum_{j=1}^{m} \max\left(1 - y_j(\langle \zeta(a_j), x\rangle - \beta), 0\right) + \frac{\lambda}{2}\langle x, x\rangle \\
&\quad \Leftrightarrow \min_{x\in\tilde{V}, \beta\in\mathbb{R}, s\in\mathbb{R}^m} \frac{1}{m}\sum_{j=1}^{m} s_j + \frac{\lambda}{2}\|x\|_{\tilde{V}}^2 \\
&\qquad \text{subject to } s_j \geq 1 - y_j(\langle \zeta(a_j), x\rangle - \beta), \quad s_j \geq 0, \quad (j = 1, \ldots, m) \\
&\overset{\text{(convex duality)}}{\Longleftrightarrow} \min_{\alpha\in\mathbb{R}^m} \frac{1}{2}\alpha^{\mathrm{T}} Q\alpha - \sum_{j=1}^{m} \alpha_j \\
&\qquad \text{subject to } 0 \leq \alpha_j \leq \frac{1}{\lambda}, \quad (j = 1, \ldots, m) \\
&\qquad\qquad y^{\mathrm{T}}\alpha = 0,
\end{aligned}$$

where $Q = (q_{k,\ell}) \succeq 0$ in $\mathbb{SR}^{m\times m}$ is defined by $q_{k,\ell} = y_k y_\ell \langle \zeta(a_k), \zeta(a_\ell)\rangle$.

Note: to solve this convex quadratic programming problem (QP), there is no need to know ζ explicitly, only the ability to compute $K(a_k, a_\ell) := \langle \zeta(a_k), \zeta(a_\ell)\rangle$ is required, where $K : V \times V \to \mathbb{R}$ is a *kernel function*, i.e., such that $(K(a_j, a_\ell))_{j=1:m,\ell=1:m} \succeq 0$ for all choices of vectors

$$\{a_1, \ldots, a_m\} \subset V = \mathbb{R}^n.$$

For example, use a *Gaussian kernel* $K(a_j, a_\ell) = \exp\left(-\|a_j - a_\ell\|_2^2/2\sigma^2\right)$.

Example 7 (Multiclass Classification).

i) *Logistic regression*: $V = \mathbb{R}^n$, $W = \{e_1, \ldots, e_M\} \subset \mathbb{R}^M$, where $y_j = e_i$ (the i-th canonical basis vector in $\mathbb{R}^M$) indicates that a_j is in class C_i for

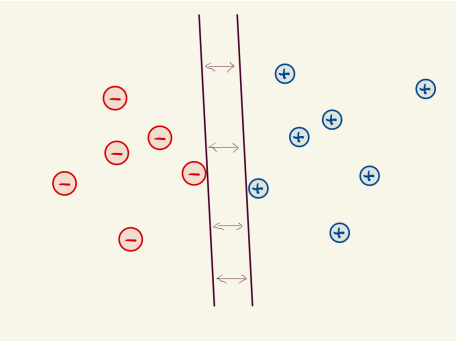

Figure 1: The support vector machine model maximises the margin between two point clouds.

classes $C_1, \ldots, C_M$. The multiclass classification problem is to output a probability map

$$\Phi(a; X) = \begin{pmatrix} p_i(a; x) \\ \vdots \\ p_M(a; X) \end{pmatrix}, \quad p_i(a; X) = \mathrm{P}(a \in C_i).$$

We wish to calibrate the parametric models

$$p_i(a; X) = \frac{\exp\left(a^{\mathrm{T}} x_{[i]}\right)}{\sum_{k=1}^{M} \exp\left(a^{\mathrm{T}} x_{[i]}\right)}, \qquad (i = 1, \ldots, M)$$

such that

$$p_i(a_j; X) \approx \begin{cases} 1 & \text{if } y_j = e_i, \\ 0 & \text{if } y_j \neq e_i, \end{cases}$$

where $x_{[i]} \in \mathbb{R}^n$ $(i = 1, \ldots, M)$ and $X = \{x_{[i]} : i = 1, \ldots, M\}$. Idea: maximise the log-likelihood function, which leads to the data fitting model

$$\max_X \frac{1}{m} \sum_{j=1}^{m} \left[y_j^{\mathrm{T}} \begin{pmatrix} x_{[1]}^{\mathrm{T}} \\ \vdots \\ x_{[M]}^{\mathrm{T}} \end{pmatrix} a_j - \log \left(\sum_{k=1}^{M} \exp(x_{[k]}^{\mathrm{T}} a_j) \right) \right].$$

The obective is convex and in summation form. Note that if $y_j = e_i$ then

$$y_j^{\mathrm{T}} \begin{pmatrix} x_{[1]}^{\mathrm{T}} \\ \vdots \\ x_{[M]}^{\mathrm{T}} \end{pmatrix} = x_i^{\mathrm{T}}.$$

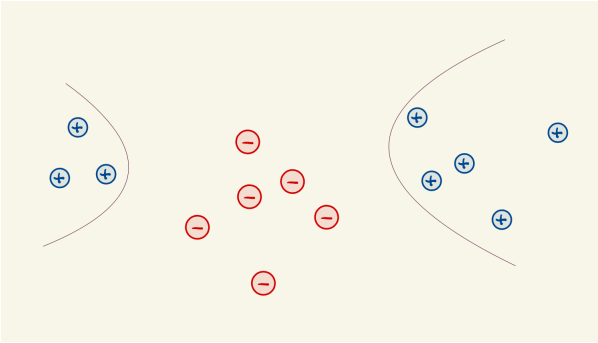

Figure 2: Nonlinear separation of point clouds.

ii) *Nonlinear multiclass logistic regression via deep learning*: Combines logistic regression with a nonlinear transformation of the inputs in $D \gg 1$ hidden layers $1, \ldots, D$ of a neural network,

$$\begin{aligned} a^0 &= a, \\ a^\ell &= \sigma(W^\ell a^{\ell-1} + g^\ell), \quad (\ell = 1, \ldots, D-1), \\ a^D &= W^D a^{D-1} + g^D, \end{aligned}$$

where $W \in \mathbb{R}^{|a^\ell| \times |a^{\ell-1}|}$, $g^\ell \in \mathbb{R}^{|a^\ell|}$, and σ is a nonlinear *activation function*, such as

- $\sigma(t) = \frac{1}{1+\exp(-t)}$ (logistic function),
- $\sigma(t) = \max(t, 0)$ (hinge loss, also called ReLU),
- $\sigma(t) = \tanh(t)$ (smooth version of ReLU),
- $\sigma(t) = 1$ with probability $1/(1+\exp(-t))$ and 0 otherwise.

Let $w = \big((W^1, g^1), \ldots, (W^D, g^D)\big)$ be the parameters of the NN, and set $a^D = a^D(w)$ as a function of the network parameters. The data fitting model becomes

$$\max_{X,w} \frac{1}{m} \sum_{j=1}^{m} \left[y_j^{\mathrm{T}} \begin{pmatrix} x_{[1]}^{\mathrm{T}} \\ \vdots \\ x_{[M]}^{\mathrm{T}} \end{pmatrix} a_j^D(w) - \log \left(\sum_{k=1}^{M} \exp(x_{[k]}^{\mathrm{T}} a_j^D(w)) \right) \right].$$

The objective is still in summation form but now non-convex.

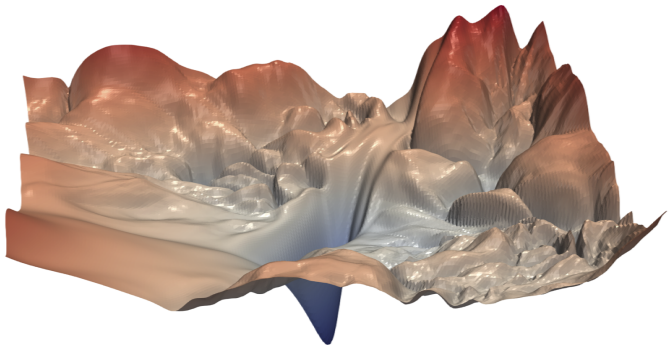

Figure 3: In this region we see one strict global minimiser, and many local minimisers. Source: http://papers.nips.cc/paper/7875-visualizing-the-loss-landscape-of-neural-nets.pdf.

3 Optimisation Terminology and Prerequisites

3.1 Terminology

Consider the optimisation model

$$\min_{x\in\mathbb{R}^n} f(x) \quad \text{subject to } x \in \mathcal{F} \tag{4}$$

Definition 1. *A solution $x^* \in \mathcal{F}$ is*

i) *a* local minimiser *if $f(x^*) \leq f(x)\ \forall x \in \mathcal{F} \cap \mathrm{B}_\varepsilon(x^*)$ for some $\varepsilon > 0$,*

ii) *a* strict local minimiser *if $f(x^*) < f(x)\ \forall x \in \mathcal{F} \cap \mathrm{B}_\varepsilon(x^*)$,*

iii) *a* global mimimiser *if $f(x^*) \leq f(x)\ \forall x \in \mathcal{F}$.*

Definition 2.

i) *$f : \mathbb{R}^n \to \mathbb{R}$ is* convex *if*

$$f\left((1-\alpha)x + \alpha y\right) \leq (1-\alpha)f(x) + \alpha f(y), \quad \forall \alpha \in [0,1], x, y \in \mathbb{R}^n. \tag{5}$$

ii) *$f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is* proper convex *if (5) holds and $f(x) < +\infty$ for at least one point x.*

iii) *For $\Omega \subset \mathbb{R}^n$, define the indicator function as follows,*

$$I_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega, \\ +\infty & \text{otherwise.} \end{cases}$$

Then $\Omega \neq \emptyset$ is a convex set if and only if $I_\Omega(x)$ is a proper convex function.

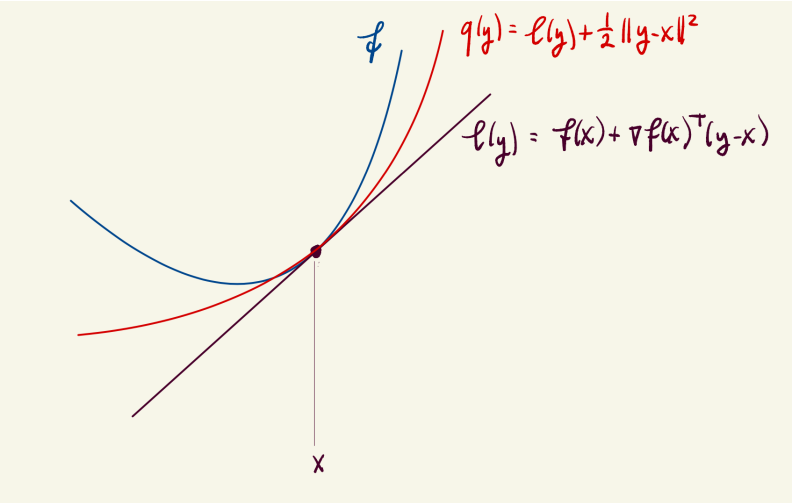

Figure 4: A strongly convex function.

iv) We say that (4) *is a* convex optimisation problem *if* f *and* $\mathcal{F} \neq \emptyset$ *are both convex. Then* (4) *is equivalent to the unconstrained problem*

$$\min_{x \in \mathbb{R}^n} f(x) + I_{\mathcal{F}}(x).$$

with proper convex objective.

v) $f : \mathbb{R}^n \to \mathbb{R}$ *is* strongly convex *with modulus of convexity* $\gamma > 0$ *if for all* $\alpha \in [0,1]$ *and* $x, y \in \mathbb{R}^n$*,*

$$f\left((1-\alpha)x + \alpha y\right) \leq (1-\alpha)f(x) + \alpha f(y) - \frac{1}{2}\gamma\alpha(1-\alpha)\|x-y\|^2.$$

vi) $f : \mathbb{R}^n \to \mathbb{R}$ *is* L-smooth *if differentiable everywhere with* L*-Lipschitz continous gradient,* $L > 0$*, that is,*

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x-y\| \quad \forall x, y \in \mathbb{R}^n.$$

vii) $v \in \mathbb{R}^n$ *is a* subgradient *of* f *at* x *if*

$$f(y) \geq f(x) + v^{\mathrm{T}}(y-x), \quad \forall x, y \in \mathbb{R}^n.$$

viii) The subdifferential $\partial f(x)$ *of* f *at* x *is the set of subgradients of* f *at* x*.*

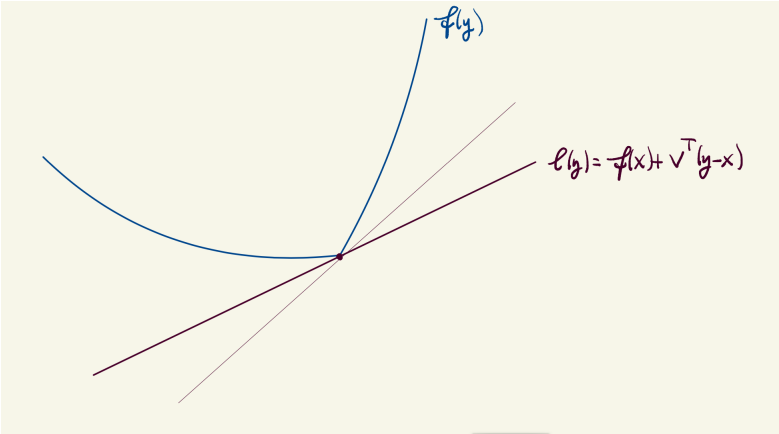

Figure 5: At points where f is not differentiable the subgradient is non-unique.

3.2 Prerequisites

Lemma 1.

i) *If $a \in \partial f(x)$ and $b \in \partial f(y)$, then $(a-b)^{\mathrm{T}}(x-y) \geq 0$.*

ii) *For any $\lambda \geq 0$, we have $\partial \lambda f(x) = \lambda \partial f(x)$.*

iii) *$\partial(f+g)(x) = \partial f(x) + \partial g(x)$, that is, for every $z \in \partial(f+g)(x)$ $\exists u \in \partial f(x)$ and $v \in \partial g(x)$ such that $z = u + v$.*

Proposition 1. *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be proper convex.*

i) *Then $x^* \in \mathbb{R}^n$ is a local minimiser if and only if it is a global minimiser.*

ii) *The set of miminisers*

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

is convex.

iii) *$x^* \in \arg\min_{x \in \mathbb{R}^n} f(x)$ if and only if[1] $0 \in \partial f(x^*)$.*

Proposition 2. *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be proper convex.*

i) *If f is differentiable at x, then $\partial f(x) = \{\nabla f(x)\}$.*

ii) *If $|\partial f(x)| = 1$, then f is differentiable at x.*

[1] To simplify notation, we write 0 instead of $\vec{0}$ for vectors of zeroes of size n or any other size

iii) If f is γ-strongly convex and differentiable at $x \in \mathbb{R}^n$, then

$$\frac{\gamma}{2}\|y-x\|^2 \leq f(y) - f(x) - \nabla f(x)^{\mathrm{T}}(y-x) \quad \forall y \in \mathbb{R}^n. \tag{6}$$

iv) If f is L-smooth, then

$$f(y) - f(x) - \nabla f(x)^{\mathrm{T}}(y-x) \leq \frac{L}{2}\|y-x\|^2 \quad \forall y \in \mathbb{R}^n. \tag{7}$$

Proposition 3. *If $f : \mathbb{R}^n \to \mathbb{R}$ is strongly convex, then there exists a unique global minimiser.*

For proofs of slightly weakened versions of Lemma 1 and Propositions 1 – 3, see problem sheets.

3.3 Characterisation of Convergence Speed

To obtain a notion of convergence for iterative algorithms, we could show that $\|\nabla f(x^k)\| \to 0$ as $k \to \infty$, or $\mathrm{dist}(0, \partial f(x^k)) \to 0$ (where $\mathrm{dist}(\cdot)$ denotes the Euclidean distance), or $\|x^k - x^*\| \to 0$, or $f(x^k) \to f(x^*)$. The rate of convergence can be characterised in several different ways:

Definition 3. *Let $(\phi_k)_{k\in\mathbb{N}} \subset \mathbb{R}_+$ be a sequence such that $\lim_{k\to\infty} \phi_k = 0$. We say that $(\phi_k)_{k\in\mathbb{N}}$ converges*

i) Q-linearly *if $\exists \sigma \in (0,1)$ such that $\frac{\phi_{k+1}}{\phi_k} \leq 1-\sigma, \quad \forall k \in \mathbb{N}$,*

ii) R-linearly *if $\exists\, \sigma \in (0,1)$ such that $\phi_k \leq C(1-\sigma)^k, \quad \forall k \in \mathbb{N}$,*

iii) Q-superlinearly *if $\lim_{k\to\infty} \frac{\phi_{k+1}}{\phi_k} = 0$,*

iv) Q-quadratically *if $\exists\, C > 0$ such that $\frac{\phi_{k+1}}{\phi_k^2} \leq C, \quad \forall k$,*

v) R-superlinearly *if $\exists\, (\nu_k)_{k\in\mathbb{N}} \subset \mathbb{R}_+$ such that $(\nu_k)_{k\in\mathbb{N}} \to 0$ Q-superlinearly and $\phi_k \leq \nu_k$ for all $k \in \mathbb{N}$.*

Lemma 2.

Q-linear convergence $\Rightarrow$ R-linear convergence $\not\Rightarrow$ Q-linear convergence,
Q-quadratic convergence $\Rightarrow$ Q-superlinear convergence $\Rightarrow$ R-superlinear convergence,
R-superlinear convergence $\not\Rightarrow$ Q-superlinear convergence $\not\Rightarrow$ Q-quadratic convergence.

See problem sheets for a proof and counterexamples.

Definition 4. *If $\phi_k \to 0$ slower than R-linearly, we say that the convergence is* sublinear.

Example 8. The following sequences all converge to zero sublinearly,

$$\left(\frac{c}{\sqrt{k}}\right)_{k\in\mathbb{N}}, \left(\frac{c}{k}\right)_{k\in\mathbb{N}}, \left(\frac{c}{\ln(k+1)}\right)_{k\in\mathbb{N}}.$$

4 Method of Steepest Descent

4.1 Specifications

Gradient methods solve unconstrained optimisation models of the form

$$\min_{x\in\mathbb{R}^n} f(x), \tag{8}$$

where $f:\mathbb{R}^n\to\mathbb{R}$ is L-smooth via iterative updates of the form

$$x^{k+1}=x^k+\alpha_k d^k. \tag{9}$$

The scalar $\alpha_k>0$ is called the *step length*, and $d^k\in\mathbb{R}^n$ is the *search direction*.

The *Method of Steepest Descent* is a gradient method based on making the following choices: $d^k=-\nabla f(x^k)$ for all k (the direction of steepest descent), and $\alpha_k\equiv L^{-1}$ for all k (constant step length). This leads to the *steepest descent updates*

$$x^{k+1}=x^k-\frac{1}{L}\nabla f(x^k). \tag{10}$$

Our main concern is whether the choice of α_k is neither too short (leading to excessively slow convergence) nor too long (leading to zig-zagging behaviour).

The updates (10) are motivated by the minimisation of the first order Taylor approximation

$$f(x+\alpha d)\le f(x)+\alpha\nabla f(x)^{\mathrm{T}}d+\alpha^2\frac{L}{2}\|d\|^2 \tag{11}$$

at $x=x^k$ over α for fixed $d=-\nabla f(x)$. (11) is an immediate consequence of (7). Writing $\varphi(\alpha)=f(x+\alpha d)$ and setting $d=-\nabla f(x)$, the r.h.s. of (11) is a convex quadratic

$$\varphi(\alpha)=f(x)-\alpha\|\nabla f(x)\|^2+\frac{\alpha^2}{2}L\|\nabla f(x^k)\|^2.$$

Minimising $\varphi(\alpha)$ be solving for $\varphi'(\alpha)=0$ yields $\alpha^*=1/L$.

The following will be one of our main tools of convergence analysis for the method of steepest descent:

Lemma 3 (Foundational Inequality of Steepest Descent)**.** *Starting steepest descent iterations at some arbitrary point x^0, we have*

$$f\left(x^{k+1}\right)=f\left(x^k-\frac{1}{L}\nabla f(x^k)\right)\le f(x^k)-\frac{1}{2L}\|\nabla f(x^k)\|^2\quad\forall k.$$

Proof. Substitute $\alpha=1/L$ and $d=-\nabla f(x^k)$ into (11). □

4.2 Convergence Theory

Overview of convergence results below:

- $(\nabla f(x^k))_{k\in\mathbb{N}} \to 0$ sublinearly with rate $O(1/\sqrt{k})$ for general f.
- $(f(x^k) - f(x^*))_{k\in\mathbb{N}} \to 0$ sublinearly with rate $O(1/k)$ for convex f.
- $(f(x^k) - f(x^*))_{k\in\mathbb{N}} \to 0$ Q-linearly with rate $(1 - \gamma/L)$ when f is γ-strongly convex.

Theorem 1. *For $f(x)$ L-smooth and bounded below by $\overline{f}$, the method of steepest descent with constant step size $\alpha_k = 1/L$ satisfies*

$$\min_{0\le k\le T-1} \|\nabla f(x^k)\| \le \sqrt{\frac{2L \times (f(x^0) - \overline{f})}{T}}, \quad \forall T \ge 1.$$

Proof. Lemma 3 implies

$$\sum_{k=0}^{T-1} \|\nabla f(x^k)\| \le 2L \sum_{k=0}^{T-1} \left[f(x^k) - f(x^{k+1})\right] = 2L \left[f(x^0) - f(x^T)\right].$$

Therefore,

$$\min_{0\le k\le T-1} \|\nabla f(x^k)\| \le \sqrt{\frac{1}{T}\sum_{k=0}^{T-1} \|\nabla f(x^k)\|^2} \le \sqrt{\frac{2L\left[f(x^0) - f(x^T)\right]}{T}}$$

□

Theorem 2. *If f is L-smooth and convex with minimiser[2] x^*, then the method of steepest descent with constant step size $\alpha_k = 1/L$ satisfies*

$$f(x^T) - f(x^*) \le \frac{L}{2T}\|x^0 - x^*\|^2, \quad \forall T \ge 1.$$

Proof. By convexity of f, $f(x^*) \ge f(x^k) + \nabla f(x^k)(x^* - x^k)$. Substitution into Lemma 3 yields

$$\begin{aligned} f(x^{k+1}) &\le f(x^*) + \nabla f(x^k)^{\mathrm{T}}(x^k - x^*) - \frac{1}{2L}\|\nabla f(x^k)\|^2 \quad \text{(use upper bound on } f(x^k)\text{)} \\ &= f(x^*) + \frac{L}{2}\left(\|x^k - x^*\|^2 - \left\|x^k - x^* - \frac{1}{L}\nabla f(x^k)\right\|^2\right) \quad \text{(expand to check)} \\ &= f(x^*) + \frac{L}{2}\left(\|x^k - x^*\|^2 - \|x^{k+1} - x^*\|^2\right) \quad \text{(use def of } x^{k+1}\text{)}. \end{aligned}$$

[2]Note that the minimiser x^* is not necessarily unique.

Repeated use of this bound in a telescoping sum yields

$$\sum_{k=0}^{T-1}\left[f(x^{k+1})-f(x^*)\right] \leq \frac{L}{2}\sum_{k=0}^{T-1}\left(\|x^k-x^*\|^2-\|x^{k+1}-x^*\|^2\right) \leq \frac{L}{2}\|x^0-x^*\|^2,$$

and since $f(x^{k+1}) \geq f(x^T)$ by virtue of Lemma 3, this implies

$$f(x^T)-f(x^*) \leq \frac{1}{T}\sum_{k=0}^{T-1}\left(f(x^{k+1})-f(x^*)\right) \leq \frac{L}{2T}\|x^0-x^*\|^2.$$

□

Theorem 3. *If f is L-smooth and γ-strongly convex then there exists a unique minimiser x^*, and*

$$f(x^{k+1})-f(x^*) \leq \left(1-\frac{\gamma}{L}\right)\left(f(x^k)-f(x^*)\right) \quad \forall\, k.$$

Proof. For the uniqueness of x^*, see Proposition 3. Proposition 2 implies that for all $y \in \mathbb{R}^n$,

$$f(y) \geq f(x^k)+\nabla f(x^k)^{\mathrm{T}}(y-x^k)+\frac{\gamma}{2}\|y-x^k\|^2.$$

Therefore,

$$\min_y f(y) \geq \min_y f(x^k)+\nabla f(x^k)^{\mathrm{T}}(y-x)+\frac{\gamma}{2}\|y-x\|^2. \tag{12}$$

The $\arg\min$ of the right hand side of (12) equals $y^* = x^k-(1/\gamma)\nabla f(x^k)$, and substitution back into (12) gives

$$\begin{aligned} f(x^*) &\geq f(x^k)-\nabla f(x^k)^{\mathrm{T}}\left(\frac{1}{\gamma}\nabla f(x^k)\right)+\frac{\gamma}{2}\left\|\frac{1}{\gamma}\nabla f(x^k)\right\|^2, \\ &= f(x^k)-\frac{1}{2\gamma}\left\|\nabla f(x^k)\right\|^2, \end{aligned}$$

which in turn yields

$$\left\|\nabla f(x^k)\right\|^2 \geq 2\gamma\left(f(x^k)-f(x^*)\right). \tag{13}$$

Substituting into the bound of Lemma 3, we find

$$f(x^{k+1})-f(x^*) \leq f(x^k)-f(x^*)-\frac{\gamma}{L}\left(f(x^k)-f(x^*)\right).$$

□

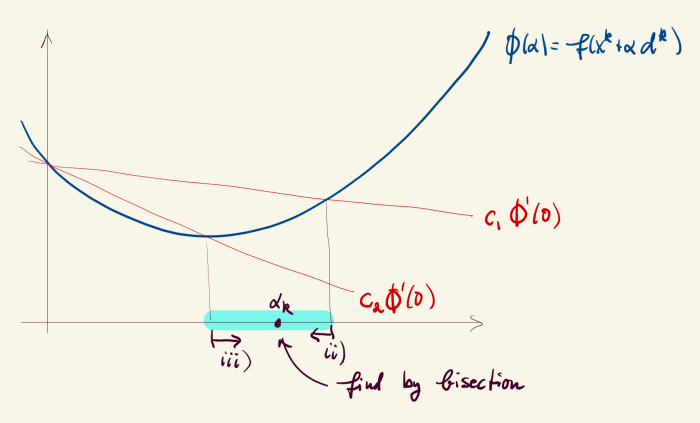

Figure 6: A step size α_k that satisfies Conditions ii) and iii) can be found by bisection.

4.3 Long-Step Descent Methods

We continue solving the unconstrained model (8) via iterative updates (9), but now with more general (and possibly much larger) step sizes α_k and search diretions d^k than before.

Throughout this section we assume $f(x)$ to be L-smooth, possibly non-convex, and bounded below $f(x) \geq \overline{f}$ for all $x \in \mathbb{R}^n$. The search directions d^k are assumed to lie within a bounded angle of the steepest descent direction. Specifically, our assumptions on d^k and α_k are as follows:

Assumption 1. There exist constants $0 < \eta \leq 1$ and $0 < c_1 < c_2 < 1$ such that for all $k \in \mathbb{N}$,

i) $\nabla f(x^k)^{\mathrm{T}} d^k \leq -\eta \|\nabla f(x^k)\| \cdot \|d^k\|$ (sufficient local descent),

ii) $f(x^k + \alpha_k d^k) \leq f(x^k) + c_1 \alpha_k \nabla f(x^k)^{\mathrm{T}} d^k$ (step not too long),

iii) $\nabla f(x^k + \alpha_k d^k))^{\mathrm{T}} d^k \geq c_2 \nabla f(x^k)^{\mathrm{T}} d^k$ (step not too short).

Note that Assumption i) is trivially met if we choose the steepest descent direction $d^k = -\nabla f(x^k)$, in which case $\eta = 1$. The condition allows us more generality in computing d^k, for example via an inexact computation of $-\nabla f(x^k)$ that is numerically cheaper to compute. Assumptions ii) and iii) look very technical at first sight, but all we need to know for the purposes of this course is that there exist very simple and numerically cheap approximate 1-D optimisation schemes called *inexact line searches* that are able to produce values α_k that satisfy both conditions. The additional generality is important in practical computations, because the Lipschitz constant L is generally not known, so

that taking constant step sizes $\alpha_k \equiv 1/L$ is largely a theoretical choice. Line searches also allow α_k to take much larger values than $1/L$ when warranted, so that the algorithm may make more rapid progress.

Theorem 4. *Under Assumption 1, we have*

$$\min_{0\le k\le T-1} \|\nabla f(x^k)\| \le \sqrt{\frac{L}{\eta^2 c_1(1-c_2)}} \times \sqrt{\frac{f(x^0)-\overline{f}}{T}}.$$

Proof. Assumption iii) implies

$$\begin{aligned} -(1-c_2)\nabla f(x^k)^{\mathrm{T}} d^k &\overset{\text{Assn iii)}}{\le} \left[\nabla f(x^k+\alpha_k d^k) - \nabla f(x^k)\right]^{\mathrm{T}} d^k \\ &\overset{L\text{-smoothness \& C.S.}}{\le} L\cdot\alpha_k \|d^k\|^2. \end{aligned}$$

Solving for α_k yields

$$\alpha_k \ge -\frac{1-c_2}{L\|d^k\|^2}\cdot \nabla f(x^k)^{\mathrm{T}} d^k \overset{\text{Assn i)}}{\le} = \frac{\eta(1-c_2)\|\nabla f(x^k)\|}{L\|d^k\|}.$$

Substitution into Assumption ii) gives

$$f(x^{k+1}) = f(x^k+\alpha_k d^k) \overset{\text{Assn ii) \& i)}}{\le} f(x^k) - \frac{c_1(1-c_2)}{L}\eta^2\|\nabla f(x^k)\|^2, \tag{14}$$

which generalises Lemma 3 that was used in Theorem 1 for the analysis of the short-step case. The rest of the proof follows the blueprint of the proof of Theorem 1: Solving (14) for $\|\nabla f(x^k)\|^2$ yields

$$\|\nabla f(x^k)\|^2 \le \frac{L}{c_1(1-c_2)\eta^2}\left(f(x^k)-f(x^{k+1})\right).$$

Nota bene this implies $f(x^k)-f(x^{k+1})\ge 0$, so that we have descent in each iteration. Summing over k, we find

$$\begin{aligned} \sum_{k=0}^{T-1}\|\nabla f(x^k)\|^2 &\le \frac{L}{c_1(1-c_2)\eta^2}\sum_{k=0}^{T-1}\left(f(x^k)-f(x^{k+1})\right) \\ &= \frac{L}{c_1(1-c_2)\eta^2}\left(f(x^0)-f(x^T)\right) \\ &\le \frac{L}{c_1(1-c_2)\eta^2}\left(f(x^0)-\overline{f}\right). \end{aligned}$$

The result now follows from

$$\min_{0\le k\le T}\|\nabla f(x^k)\| \le \sqrt{\frac{1}{T}\sum_{k=0}^{T-1}\|\nabla f(x^k)\|^2}.$$

□

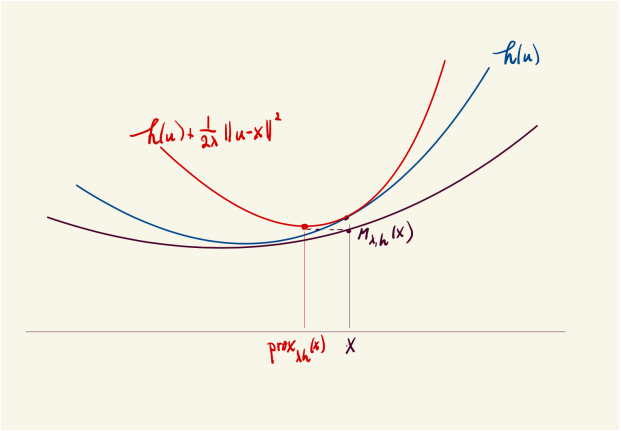

Figure 7: Construction of the Moreau envelope.

5 The Proximal Method

5.1 Proximal Operators

The following notion is useful in designing algorithms for minimising regularised problems:

Definition 5. *Let $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be proper*[3] *convex and closed*[4].

i) The Moreau envelope *of h associated with $\lambda > 0$ is the function*

$$M_{\lambda,h} : \mathbb{R}^n \to \mathbb{R},$$
$$x \mapsto \frac{1}{\lambda} \inf_u \left\{ \lambda h(u) + \frac{1}{2}\|u - x\|^2 \right\} = \inf_u \left\{ h(u) + \frac{1}{2\lambda}\|u - x\|^2 \right\}.$$

The Moreau envelope is a smoothed version of h.

ii) The proximal operator *is given by*

$$\operatorname{prox}_{\lambda h} : \mathbb{R}^n \to \mathbb{R},$$
$$x \mapsto \arg\min_u \left\{ \lambda h(u) + \frac{1}{2}\|u - x\|^2 \right\}.$$

The function is well defined by virtue of Proposition 3.

[3] By definition, $h(u)$ is finite for at least one point $u \in \mathbb{R}^n$.
[4] A function is closed if all sublevel sets $\{x : f(x) \leq \alpha\}$ are closed

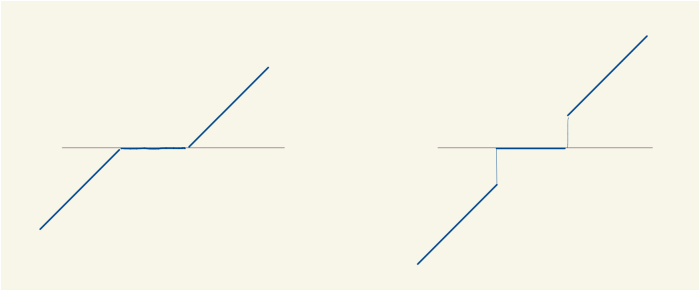

Figure 8: Soft and hard thresholding functions.

Example 9.

- $h(x) = 0$ has prox operator $\text{prox}_{\lambda h}(x) = \arg\min_u \frac{1}{2}\|u - x\|^2 = x$, that is, the identity.
- $h(x) = I_\Omega(x)$, with $\Omega \subseteq \mathbb{R}^n$ closed convex, has prox operator

$$\begin{aligned}\text{prox}_{\lambda I_\Omega}(x) &= \arg\min_u \lambda I_\Omega(u) + \frac{1}{2}\|u - x\|^2 \\ &= \arg\min_{u\in\Omega} \|u - x\|^2,\end{aligned}$$

  that is, the projection of x onto Ω.

- $h(x) = \|x\|_1$ has prox operator with i-th component

$$(\text{prox}_{\lambda\|\cdot\|_1}(x))_i = \begin{cases} x_i - \lambda & \text{if } x_i > \lambda, \\ 0 & \text{if } x_i \in [-\lambda, \lambda], \\ x_i + \lambda & \text{if } x_i < -\lambda. \end{cases}$$

  This is called the *soft thresholding* operator.

- $h(x) = \|x\|_0 := |\{i : x_i \neq 0\}|$ has prox operator with i-th component

$$(prox_{\lambda\|\cdot\|_0}(x))_i = \begin{cases} x_i & \text{if } |x_i| > \sqrt{2\lambda}, \\ 0 & \text{if } |x_i| < \sqrt{2\lambda}. \end{cases}$$

  This is called *hard thresholding*.

Proposition 4. *Let $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be proper convex and closed. For fixed $x \in \mathbb{R}^n$, let $\Phi(u) := \lambda h(u) + \frac{1}{2}\|u - x\|^2$. Then the following properties hold.*

i) $0 \in \lambda\partial h\left(\text{prox}_{\lambda h}(x)\right) + \text{prox}_{\lambda h}(x) - x.$

ii) $M_{\lambda,h}(x) < +\infty$ *for all* $x \in \mathbb{R}^n$*, even if* $h(x) = +\infty$.

iii) $M_{\lambda,h}(x)$ *is convex in* x.

iv) $M_{\lambda,h}(x)$ *is differentiable in* x*, with gradient*

$$\nabla M_{\lambda,h}(x) = \frac{1}{\lambda}\left(x - \operatorname{prox}_{\lambda h}(x)\right).$$

v) $x^* \in \arg\min_x h(x)$ *if and only if* $x^* \in \arg\min_x M_{\lambda,h}(x)$.

vi) $\|\operatorname{prox}_{\lambda h}(x) - \operatorname{prox}_{\lambda h}(y)\| \leq \|x - y\|$ *for all* $x, y \in \mathbb{R}^n$.

See problem sheets for a proof.

5.2 The Prox-Gradient Method

We now extend the scope of problems to solve and consider the regularised optimisation model

$$\min_{x\in\mathbb{R}^n} \phi(x) := f(x) + \lambda\psi(x), \tag{15}$$

where f is L-smooth and convex, and $\psi(x)$ is convex and closed. This includes the case of constrained optimisation, as

$$\min_{x\in\Omega} f(x) \Leftrightarrow \min_{x\in\mathbb{R}^n} f(x) + \lambda I_\Omega(x),$$

where the equivalence is understood in the sense that the argmin are the same in both cases.

To construct an iterative algorithm for Model (15), we use the following template:

1. $y^{k+1} = x^k - \alpha_k \nabla f(x^k)$
2. $x^{k+1} = \arg\min_z \frac{1}{2}\|z - y^{k+1}\|^2 + \alpha_k\lambda\psi(z)$,

that is, we minimise $\psi(z)$ while staying close to the steepest descent update y^{k+1}. Note the factor α_k in front of $\lambda\psi(z)$ to avoid that $x^{k+1} \to y^{k+1}$ if $\alpha_k \to \infty$, that is, both terms on the r.h.s. of the second step should be of the same order. We may also express the update of the template algorithm as

$$x^{k+1} = \operatorname{prox}_{\alpha_k\lambda\psi}\left(x^k - \alpha_k\nabla f(x^k)\right). \tag{16}$$

An alternative interpretation and motivation of the update (16) is as follows:

$$x^{k+1} = \arg\min_z f(x^k) + \nabla f(x^k)^{\mathrm{T}}(z - x^k) + \frac{1}{2\alpha_k}\|z - x^k\|^2 + \lambda\psi(z),$$

Now note that $f(x^k) + \nabla f(x^k)^{\mathrm{T}}(z - x^k)$ is the 1st order Taylor approximation of $f(z)$ around x^k, so that $(x^k) + \nabla f(x^k)^{\mathrm{T}}(z - x^k) + \frac{1}{2\alpha_k}\|z - x^k\|^2$ is a simplified 2nd order model of $f(z)$ that does not depend on 2nd order derivatives of f. The updates are thus obtained by iteratively building very simple 2nd order models by which f is replaced in (15) to obtain subproblems that are easier to solve.

Example 10. Unregularised unconstrained minimisation $\min_x f(x)$. This case reduces to $\psi(x) \equiv 0$ and the steepest descent update

$$\begin{aligned} x^{k+1} &= \operatorname{prox}_0\left(x^k - \alpha_k \nabla f(x^k)\right) = \arg\min_z \frac{1}{2}\left\|z - \left(x^k - \alpha_k \nabla f(x^k)\right)\right\|^2 \\ &= x^k - \alpha_k \nabla f(x^k). \end{aligned}$$

Example 11. Constrained convex minimisation $\min_x f(x)$ subject to $x \in \Omega$ with $\Omega \subset \mathbb{R}^n$ convex. This case reduces to $\psi(x) = I_\Omega(x)$ and prox updates

$$x^{k+1} = \operatorname{prox}_{\alpha_k \lambda I_\Omega}\left(x^k - \alpha_k \nabla f(x^k)\right) = \arg\min_{z \in \Omega} \frac{1}{2}\left\|z - \left(x^k - \alpha_k \nabla f(x^k)\right)\right\|^2,$$

which is the orthogonal projection of the steepest descent update $x^k - \alpha_k \nabla f(x^k)$ onto the feasible set Ω.

Definition 6. *The* gradient map *of model* (15) *is the map*

$$\begin{aligned} G_\alpha : \mathbb{R}^n &\to \mathbb{R}^n \\ x &\mapsto \frac{1}{\alpha}\left(x - \operatorname{prox}_{\alpha\lambda\psi}\left(x - \alpha \nabla f(x)\right)\right). \end{aligned}$$

We note that the notion of gradient map allows us to write the prox update (16) as

$$x^{k+1} = \operatorname{prox}_{\alpha_k \lambda \psi}\left(x^k - \alpha_k \nabla f(x^k)\right) = x^k - \alpha_k G_{\alpha_k}(x^k), \tag{17}$$

so that in the proximal method $G_{\alpha_k}(x^k)$ takes the role of $\nabla f(x^k)$ in comparison with the method of steepest descent.

Lemma 4 (Foundational Inequality of Prox-Method)**.** *The gradient map has the following properties:*

i) $G_\alpha(x) \in \nabla f(x) + \lambda \partial \psi\left(x - \alpha G_\alpha(x)\right)$.

ii) For all $\alpha \in \left(0, \frac{1}{L}\right]$ *and* $z \in \mathbb{R}^n$,

$$\phi\left(x - \alpha G_\alpha(x)\right) \leq \phi(z) + G_\alpha(x)^{\mathrm{T}}(x - z) - \frac{\alpha}{2}\|G_\alpha(x)\|^2. \tag{18}$$

Notes:

- Property i) shows that (17) has an interpretation of a generalised subgradient step, with the gradient map replacing the gradient with the following subtlety: $-\alpha_k G_{\alpha_k}(x^k)$ consists of an *explicit* forward descent step for f and an *implicit* backward ascent step for $\lambda\psi$.

- Property ii) holds true for all $z \in \mathbb{R}^n$. The term $\phi(z) + G_\alpha(x)^{\mathrm{T}}(x - z)$ can be seen as a backward first order model of $\phi(x)$. Using (17) and setting $x = x^k$ and $\alpha = \alpha_k$, (18) reads

$$\phi\left(x^{k+1}\right) \leq \phi(z) + G_{\alpha_k}(x^k)^{\mathrm{T}}(x^k - z) - \frac{\alpha_k}{2}\|G_{\alpha_k}(x^k)\|^2.$$

  In particular, setting $z = x^k$ we obtain

$$\phi\left(x^{k+1}\right) \leq \phi(x^k) - \frac{\alpha_k}{2}\|G_{\alpha_k}(x^k)\|^2. \tag{19}$$

  Compare this result to the foundational inequality of the short-step steepest descent, i.e., Lemma 3, which reads

$$f\left(x^{k+1}\right) \leq f(x^k) - \frac{1}{2L}\|\nabla f(x^k)\|^2. \tag{20}$$

  Note that when $\alpha_k = 1/L$, (19) becomes (20) with $f(x)$ replaced by the merit function $\phi(x)$ and $\nabla f(x)$ by the gradient map. Inequality (18) thus generalises the foundational inequality and can be used the same way to analyse the convergence of the prox-gradient method as Lemma 3 was used in the analysis of the Method of Steepest Descent.

Proof. Using $x - \alpha G_\alpha(x) = \mathrm{prox}_{\alpha\lambda\psi}(x - \alpha\nabla f(x)) = \arg\min_u \Xi(u)$, where $\Xi(u) := \alpha\lambda\psi(u) + \frac{1}{2}\|u - (x - \alpha\nabla f(x))\|^2$ is convex, it must be true that

$$\begin{aligned} 0 \in \partial\,\Xi\left(\mathrm{prox}_{\alpha\lambda\psi}(x - \alpha\nabla f(x))\right) &= \partial\,\Xi(x - \alpha G_\alpha(x)) \\ &= \alpha\lambda\partial\psi(x - \alpha G_\alpha(x)) + (x - \alpha G_\alpha(x)) - (x - \alpha\nabla f(x)) \end{aligned}$$

Therefore, $\alpha G_\alpha(x) \in \alpha\lambda\partial\psi(x - \alpha G_\alpha(x)) + \alpha\nabla f(x)$, as claimed in i).

Further, since f is L-smooth, (7) implies

$$f(y) \leq f(x) + \nabla f(x)^{\mathrm{T}}(y - x) + \frac{L}{2}\|y - x\|^2, \quad \forall\, y \in \mathbb{R}^n.$$

Substituting $y = x - \alpha G_\alpha(x)$ yields

$$\begin{aligned} f(x - \alpha G_\alpha(x)) &\leq f(x) - \alpha G_\alpha(x)^{\mathrm{T}}\nabla f(x) + \frac{L\alpha^2}{2}\|G_\alpha(x)\|^2 \\ &\leq f(x) - \alpha G_\alpha(x)^{\mathrm{T}}\nabla f(x) + \frac{\alpha}{2}\|G_\alpha(x)\|^2, \end{aligned} \tag{21}$$

where the last inequality follows from the assumption that $\alpha \in (0, 1/L]$. On the other hand, the convexity of $f(x)$ implies (see Definition 2 and properties of the subdifferential)

$$f(z) \geq f(x) + \nabla f(x)^{\mathrm{T}}(z - x) \tag{22}$$

By virtue part i) we also have $G_\alpha(x) - \nabla f(x) \in \partial\lambda\psi\left(x - \alpha G_\alpha(x)\right)$, so that by the definition of subgradients we have

$$\lambda\psi(z) \geq \lambda\psi\left(x - \alpha G_\alpha(x)\right) + \left[G_\alpha(x) - \nabla f(x)\right]^{\mathrm{T}}\left(z - (x - \alpha G_\alpha(x))\right). \tag{23}$$

It follows that

$$\begin{aligned}
\phi\left(x - \alpha G_\alpha(x)\right) &\overset{\text{def of } \phi}{=} f\left(x - \alpha G_\alpha(x)\right) + \lambda\psi\left(x - \alpha G_\alpha(x)\right) \\
&\overset{(21)}{\leq} f(x) - \alpha G_\alpha(x)^{\mathrm{T}}\nabla f(x) + \frac{\alpha}{2}\|G_\alpha(x)\|^2 + \lambda\psi\left(x - \alpha G_\alpha(x)\right) \\
&\overset{(22)}{\leq} f(z) - \nabla f(x)^{\mathrm{T}}\left(z - x + \alpha G_\alpha(x)\right) + \frac{\alpha}{2}\|G_\alpha(x)\|^2 \\
&\quad + \lambda\psi\left(x - \alpha G_\alpha(x)\right) \\
&\overset{(23)}{\leq} f(z) - \nabla f(x)^{\mathrm{T}}\left(z - x + \alpha G_\alpha(x)\right) + \frac{\alpha}{2}\|G_\alpha(x)\|^2 \\
&\quad + \lambda\psi\left(z\right) - \left[G_\alpha(x) - \nabla f(x)\right]^{\mathrm{T}}\left(z - (x - \alpha G_\alpha(x))\right) \\
&= f(z) + \lambda\psi(z) + G_\alpha(x)^{\mathrm{T}}(x - z) - \frac{\alpha}{2}\|G_\alpha(x)\|^2 \\
&= \phi(z) + G_\alpha(x)^{\mathrm{T}}(x - z) - \frac{\alpha}{2}\|G_\alpha(x)\|^2,
\end{aligned}$$

as claimed in part ii). □

5.3 Convergence Theory

Theorem 5. *If Model (15) has a minimiser x^* and the prox-gradient algorithm is run with constant step length $\alpha_k \equiv 1/L$ starting from an initial point x^0, then*

$$\phi(x^k) - \phi(x^*) \leq \frac{L}{2k}\|x^0 - x^*\|^2, \quad \forall\, k \geq 1.$$

Proof. Equation (19) establishes that $\left(\phi(x^k)\right)_{k\in\mathbb{N}}$ is monotone decreasing,

$$\phi(x^{k+1}) \leq \phi(x^k), \quad \forall\, k. \tag{24}$$

Further, using $z = x^*$ in Lemma 4 ii), we have

$$\begin{aligned}
0 \leq \phi(x^{k+1}) - \phi(x^*) &\leq G_{\alpha_k}(x^k)^{\mathrm{T}}(x^k - x^*) - \frac{\alpha_k}{2}\|G_{\alpha_k}(x^k)\|^2 \\
&= \frac{1}{2\alpha_k}\left(\left\|x^k - x^*\right\|^2 - \left\|x^k - x^* - \alpha_k G_{\alpha_k}(x^k)\right\|^2\right) \\
&= \frac{1}{2\alpha_k}\left(\left\|x^k - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2\right).
\end{aligned} \tag{25}$$

Therefore, $\left(\|x^k - x^*\|\right)_{k\in\mathbb{N}}$ is also montone decreasing.

Combining both insights, we find

$$\begin{aligned} K \times \left(\phi(x^K) - \phi(x^*)\right) &\overset{(24)}{\leq} \sum_{k=0}^{K-1} \left(\phi(x^{k+1}) - \phi(x^*)\right) \\ &\overset{(25),\alpha_k = L^{-1}}{\leq} \frac{L}{2} \sum_{k=0}^{K-1} \left(\left\|x^k - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2\right) \\ &\overset{\text{telescoping sum}}{\leq} \frac{L}{2} \times \left\|x^0 - x^*\right\|^2. \end{aligned}$$

□

6 Acceleration of Gradient Methods

6.1 Summary of Complexity Results Seen so Far

- The Method of Steepest Descent solves

$$\min_x f(x)$$

  - in sublinear $O\left(\frac{1}{\sqrt{k}}\right)$ time when f is smooth (but nonconvex) and bounded below, both in short and long step variants,
  - in sublinear $O\left(\frac{1}{k}\right)$ time when f is smooth, convex and bounded below,
  - in linear time with rate $\left(1 - \frac{\gamma}{L}\right)$ when f is L-smooth and γ-strongly convex.

- The Prox-Gradient Method solves

$$\min_x f(x) + \lambda\psi(x)$$

  in sublinear $O\left(\frac{1}{k}\right)$ time when f is smooth and convex and ψ is convex and closed.

Accelerated gradient methods are designed to obtain faster, in some cases provably optimal, convergence rates.

6.2 The Heavy Ball Method

This algorithm is the easiest to understand among accelerated gradient methods, but it is designed for the somewhat restricted scope of minimising convex quadratic functions,

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} x^{\mathrm{T}} A x - b^{\mathrm{T}} x, \tag{26}$$

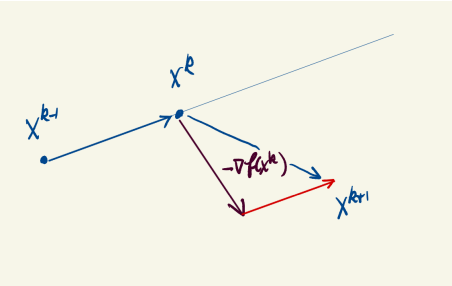

Figure 9: Heavy ball updates with momentum term.

where A is symmetric with eigenvalues in $[\gamma, L]$ for some $0 < \gamma < L$, so that $f(x)$ is L-smooth and γ-strongly convex.

The iterative updates are defined as

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

consisting of a steepest descent step $x^k - \alpha_k \nabla f(x^k)$ and an additional *momentum term* $\beta_k(x^k - x^{k-1})$ for some $\beta_k > 0$.

To avoid the requirement of two starting points x^{-1}, x^0, the first iteration takes a steepest descent update. The following theorem is given without proof:

Theorem 6. *There exists a constant $C > 0$ such that when the Heavy Ball Method is applied to Model (26) with constant step lengths*

$$\alpha_k \equiv \alpha = \frac{4}{(\sqrt{L} + \sqrt{\gamma})^2}, \quad \beta_k \equiv \beta = \frac{\sqrt{L} - \sqrt{\gamma}}{\sqrt{L} + \sqrt{\gamma}}$$

satisifies $\|x^k - x^*\| \leq C\beta^k$ for all k.

As an immediate consequence of Theorem 6 we find that

$$\begin{aligned} f(x^k) - f(x^*) &\leq \nabla f(x^*)(x^k - x^*) + \frac{L}{2}\|x^k - x^*\|^2 \quad \text{(by (6))} \\ &\leq \frac{LC^2}{2}\beta^{2k} \quad \text{(since } \nabla f(x^*) = 0\text{, and by Theorem 6)} \\ &\approx \frac{LC^2}{2}\left(1 - 2\sqrt{\frac{\gamma}{L}}\right)^{2k} \quad \text{(approximation for } L \gg \gamma\text{)} \end{aligned}$$

The case where $L \gg \gamma$ is especially interesting, as this corresponds to ill-conditioned A for which Model (26) is numerically difficult to solve. In this case the Heavy Ball Method has a much faster convergence rate than the Method of Steepest Descent, which as we saw in Theorem 3 satisfies

$$f(x^k) - f(x^*) \leq \left[f(x^0) - x(x^*)\right] \times \left(1 - \frac{\gamma}{L}\right)^k.$$

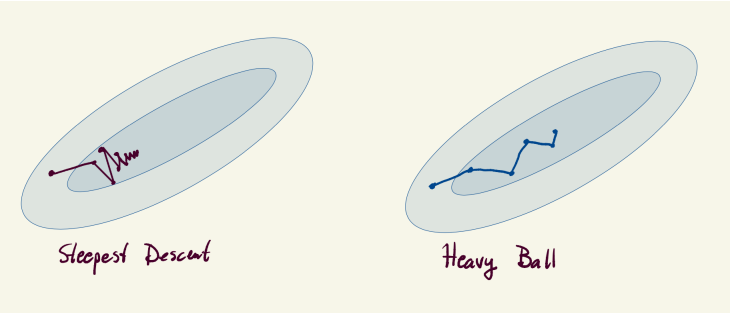

Figure 10: Heavy ball updates converge faster due to reduced zig zagging.

We conclude that the Heavy Ball Method also converges linearly, but at a faster rate than the Method of Steepest Descent.

Example 12. If $\gamma = 0.01L$, which corresponds to a matrix A with moderately sized condition number of 100, we have $(1 - 2\sqrt{\gamma/L})^2 = 0.64$. Hence, the Heavy Ball Method shrinks $f(x^k) - f(x^*)$ by at least a third in each iteration, while the steepest descent method shrinks $f(x^k) - f(x^*)$ only by 1%, since $(1 - \gamma/L) = 0.99$.

6.3 Nesterov Acceleration

Nesterov's *Accelerated Gradient Method* is a modification of the Heavy Ball Method that works for general strongly convex objectives to solve

$$\min_{x \in \mathbb{R}^n} f(x) \tag{27}$$

via iterative updates

$$x^{k+1} = x^k - \alpha_k \nabla f\left(x^k + \beta_k(x^k - x^{k-1})\right) + \beta_k(x^k - x^{k-1}), \tag{28}$$

which, compared to the heavy ball updates

$$x^{k+1} = x^k - \alpha_k \nabla f\left(x^k\right) + \beta_k(x^k - x^{k-1}), \tag{29}$$

only differ in the point where the gradient ∇f is evaluated: in the case of Nesterov's method this point contains the momentum term also. To avoid the requirement of two starting points, the first update is again taken as the steepest descent step. The next result is given without proof:

Theorem 7. *For f L-smooth γ-strongly convex, Nesterov's Accelerated Gradient Method applied to Model (27) from a given starting point $x^0 \in \mathbb{R}^n$ and with constant step lengths*

$$\alpha_k \equiv \alpha = \frac{1}{L}, \quad \beta_k \equiv \beta = \frac{\sqrt{L} - \sqrt{\gamma}}{\sqrt{L} + \sqrt{\gamma}}$$

satisifies

$$f(x^k) - f(x^*) \leq \frac{L+\gamma}{2}\|x^0 - x^*\|^2 \times \left(1 - \sqrt{\frac{\gamma}{L}}\right)^k \quad \forall\, k \geq 1.$$

Theorem 7 shows that the Accelerated Gradient Method with constant step size again converges linearly with a faster rate than the Method of Steepest Descent, as

$$1 - \sqrt{\frac{\gamma}{L}} < 1 - \frac{\gamma}{L}.$$

Example 13. For $\gamma = 0.01L$ we have $(1-\sqrt{\gamma/L}) = 0.9$ which compares favourably with the decrease rate of $(1 - \gamma/L) = 0.99$ for steepest descent. It takes more than 10 steepest descent iterations to guarantee the same relative decrease in $f(x^k) - f(x^*)$ as for Nesterov's Method.

6.4 Nesterov Acceleration of L-Smooth Convex Functions

We will now discuss a variant of Nesterov's Method for the unconstrained minimisation of an L-smooth convex function $f(x)$. Therefore, $f(x)$ is bounded from above by simple quadratic models:

$$f(y) \leq f(x) + \langle \nabla f(x), y - x\rangle + \frac{L}{2}\|y - x\|^2 \tag{30}$$

The Lipschitz constant is only assumed to exist but is not necessarily known. In contrast to the case characterised by Theorem 7, the step size varies in each iteration.

This method has an *iteration complexity* of $\mathrm{O}(\varepsilon^{-1/2})$, that is, there exists a constant $C > 0$ (which depends of course on the starting point) such that for any given error tolerance $\varepsilon > 0$ it takes only $C/\sqrt{\varepsilon}$ iterations to guarantee that $f(x^k) - f(x^*) < \varepsilon$. It is known theoretically that this iteration complexity cannot be improved, thus Nesterov's Method is *optimal* from an iteration complexity perspective.

The algorithm proceeds via two sequences of iterates, x^k and y^k, whose computation is interlaced. The points x^k are obtained as the minimisers of a quadratic upper bound function, as discussed earlier in the course, while the iterates y^k are obtained as approximate second order corrections to the points x^k.

Algorithm 1 (Nesterov Acceleration for L-Smooth Convex Minimisation)**.**

```
// initialisation
```
find $z \neq y^0 \in \mathbb{R}^n$ *and set* $\alpha_{-1} := \frac{\|y^0 - z\|}{\|\nabla f(y^0) - \nabla f(z)\|}$;
$k := 0$, $\lambda_0 := 1$, $x^{-1} := y^0$;
```
// main body
```
while $\|x^k - x^{k-1}\| > tol$ **do**
  find minimal $i \in \mathbb{N}$ *such that*

$$f\left(y^k - 2^{-i}\alpha_{k-1}\nabla f(y^k)\right) \leq f(y^k) - 2^{-(i+1)}\alpha_{k-1}\|\nabla f(y^k)\|^2 \qquad (31)$$

  $\alpha_k := 2^{-i}\alpha_{k-1}$;
  $x^k := y^k - \alpha_k \nabla f(y^k)$;
  $\lambda_{k+1} := \frac{1}{2}\left(1 + \sqrt{4\lambda_k^2 + 1}\right)$;
  $y^{k+1} := x^k + \frac{\lambda_k - 1}{\lambda_{k+1}}\left(x^k - x^{k-1}\right)$;
  $k \leftarrow k + 1$;
end

A few remarks are in order:

- Since $y^0 \neq z$, we have $\|\nabla f(y^0) - \nabla f(z)\| \leq L\|y^0 - z\|$, and hence $\alpha_{-1} \geq L^{-1}$ is an overestimate of the optimal step length. This guarantees that the steps are not unnecessarily short initially.
- Subsequently, $\alpha_{k+1} \leq \alpha_k$ for all k. The step length is approaching the unkown optimal step length L^{-1} from above until some iteration k_0 when $\alpha_{k_0} \leq L^{-1}$ occurs for the first time. At that point in all subsequent iterations $k > k_0$ Equation (30) implies that

$$\begin{aligned} f\left(y^k - \alpha_{k-1}\nabla f(y^k)\right) &\leq f(y^k) + \langle \nabla f(y^k), -\alpha_{k-1}\nabla f(y^k)\rangle + \frac{L}{2}\|\alpha_{k-1}\nabla f(y^k)\|^2 \\ &\leq f(y^k) - \frac{\alpha_{k-1}}{2}\|\nabla f(y^k)\|^2. \end{aligned}$$

  The sufficient decrease condition (31) is therefore satisfied with $i = 0$ so that $\alpha_k = \alpha_{k-1} = \cdots = \alpha_{k_0}$ ceases to decrease any further, and we have

$$\frac{L^{-1}}{2} \leq \alpha_k \leq L^{-1}, \quad \forall\, k \geq k_0, \qquad (32)$$

  and where the lower bound inequality holds for all $k \in \mathbb{N}$. Note that $\alpha \leq L^{-1}$ is exactly the condition we need to ensure that

$$m^u_{k,\alpha}(y) = f(y^k) + \langle \nabla f(y^k), y - y^k\rangle + \frac{\alpha^{-1}}{2}\|y - y^k\|^2$$

  is an upper bound function on $f(y)$. If we had a priori knowledge of L, we could simply set $\alpha = L^{-1}$ at the outset, but Algorithm 1 is designed

for the case where L is not known explicitly and learns an appropriate value of α from recycled information, that is from quantities that have to be computed anyway as the iterations proceed. The back tracking decreases the value of α at most

$$\lfloor \log_2(2L\alpha_{-1}) \rfloor$$

times before the final value is attained.

- The point x^k is simply the global minimiser of

$$m^u_{k,\alpha}(x) = f(y^k) + \langle \nabla f(y^k), x - y^k \rangle + \frac{\alpha^{-1}}{2}\|x - y^k\|^2.$$

- The update λ_{k+1} is obtained by solving the quadratic equation

$$\lambda^2_{k+1} - \lambda_{k+1} = \lambda^2_k. \tag{33}$$

It follows by induction that

$$\lambda_{k+1} \geq \lambda_k + \frac{1}{2} \geq 1 + \frac{k+1}{2}, \tag{34}$$

and furthermore,

$$\begin{aligned} \frac{\lambda_0 - 1}{\lambda_{k+1}} &= 0, \\ \frac{\lambda_k - 1}{\lambda_{k+1}} &< 1, \quad \forall k \in \mathbb{N}, \\ \lim_{k\to\infty} \frac{\lambda_k - 1}{\lambda_{k+1}} &= 1 \end{aligned}$$

- The step from x^k to y^{k+1} is designed to go along the "ravine" in ill-conditioned problems that have a long, narrow valley of near-optimal points. Pure gradient descent methods tend to stall in such situations and jump back an forth across the ravine. To speed up the convergence, a step along the ravine is required. The vector $x^k - x^{k-1}$ asymptotically points in the right direction, and the step size $\frac{\lambda_k - 1}{\lambda_{k+1}}$ is designed to start out timidly and to become asymptotically bolder.

Let us now analyse the convergence of Algorithm 1. Note that, asymptotically we have $x^k - y^k \to 0$, so that it doesn't matter whether we analyse the algorithm in terms of the points y^k or x^k. It turns out that it is easiest to carry out the analysis in terms of the vectors

$$p^k = (\lambda_k - 1)(x^{k-1} - x^k). \tag{35}$$

Theorem 8. *Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a convex L-smooth function that has at least one finite minimiser $x^* \in \arg\min f(x)$ and a finite minimum $f^* = \min f(x)$. Then the sequence $(x^k)_{k\in\mathbb{N}}$ of iterates produced by Algorithm 1 satisfies*

$$f(x^k) \leq f^* + \frac{4L\|x^0 - x^*\|^2}{(k+2)^2}$$

for all $k \in \mathbb{N}$. Hence, x^k is ε-optimal for all

$$k \geq N(\varepsilon) = \left\lfloor \frac{2\sqrt{L} \times \|x^0 - x^*\|}{\sqrt{\varepsilon}} \right\rfloor - 1.$$

Proof. Recall that we have

$$\begin{aligned} y^{k+1} &= x^k + \frac{\lambda_k - 1}{\lambda_{k+1}}(x^k - x^{k-1}) = x^k - \frac{1}{\lambda_{k+1}}(\lambda_k - 1)\left(x^{k-1} - x^k\right) \\ &= x^k - \frac{1}{\lambda_{k+1}} p^k, \end{aligned} \tag{36}$$

and

$$\begin{aligned} x^{k+1} &= y^{k+1} - \alpha_{k+1}\nabla f(y^{k+1}) \\ &= x^k + \frac{\lambda_k - 1}{\lambda_{k+1}}(x^k - x^{k-1}) - \alpha_{k+1}\nabla f(y^{k+1}), \end{aligned}$$

which implies

$$0 = \lambda_{k+1}\left(x^k - x^{k+1}\right) + (\lambda_k - 1)\left(x^k - x^{k-1}\right) - \lambda_{k+1}\alpha_{k+1}\nabla f(y^{k+1}). \tag{37}$$

Using the auxiliary vectors p^k defined in (35), we have

$$\begin{aligned} p^{k+1} - x^{k+1} &= (\lambda_{k+1} - 1)(x^k - x^{k+1}) - x^{k+1} \\ &= \lambda_{k+1}(x^k - x^{k+1}) - x^k \\ &\overset{(37)}{=} (\lambda_k - 1)\left(x^{k-1} - x^k\right) - x^k + \lambda_{k+1}\alpha_{k+1}\nabla f(y^{k+1}) \\ &= p^k - x^k + \lambda_{k+1}\alpha_{k+1}\nabla f(y^{k+1}). \end{aligned}$$

It follows that

$$\begin{aligned} &\|p^{k+1} - x^{k+1} + x^*\|^2 \\ &\quad = \|p^k - x^k + x^*\|^2 + \lambda_{k+1}^2\alpha_{k+1}^2\|\nabla f(y^{k+1})\|^2 \\ &\qquad + 2\lambda_{k+1}\alpha_{k+1}\langle \nabla f(y^{k+1}), p^k - x^k + x^*\rangle \\ &\quad = \|p^k - x^k + x^*\|^2 + \lambda_{k+1}^2\alpha_{k+1}^2\|\nabla f(y^{k+1})\|^2 \\ &\qquad + 2(\lambda_{k+1} - 1)\alpha_{k+1}\langle \nabla f(y^{k+1}), p^k\rangle \\ &\qquad\quad + 2\lambda_{k+1}\alpha_{k+1}\langle \nabla f(y^{k+1}), x^* - x^k + \lambda_{k+1}^{-1}p^k\rangle \\ &\quad \overset{(36)}{=} \|p^k - x^k + x^*\|^2 + \lambda_{k+1}^2\alpha_{k+1}^2\|\nabla f(y^{k+1})\|^2 \\ &\qquad + 2(\lambda_{k+1} - 1)\alpha_{k+1}\langle \nabla f(y^{k+1}), p^k\rangle \\ &\qquad\quad + 2\lambda_{k+1}\alpha_{k+1}\langle \nabla f(y^{k+1}), x^* - y^{k+1}\rangle, \end{aligned} \tag{38}$$

Next, we will derive bounds on the last two terms in the right hand side of (38). To bound the second term, we use convexity of f, which implies $f(x^*) \geq f(y^{k+1}) + \langle \nabla f(y^{k+1}), x^* - y^{k+1} \rangle$, and hence

$$f(y^{k+1}) - f^* \leq -\langle \nabla f(y^{k+1}), x^* - y^{k+1} \rangle. \tag{39}$$

On the other hand, the sufficient decrease condition (31) yields

$$f\left(x^k\right) = f\left(y^k - \alpha_k \nabla f(y^k)\right) \leq f(y^k) - \frac{\alpha_k}{2} \|\nabla f(y^k)\|^2$$

and at iteration $k+1$ this implies

$$f(y^{k+1}) \geq f(x^{k+1}) + \frac{\alpha_{k+1}}{2} \|\nabla f(y^{k+1})\|^2. \tag{40}$$

Substitution into (39) yields

$$\langle \nabla f(y^{k+1}), x^* - y^{k+1} \rangle \leq -\left(f(x^{k+1}) - f^*\right) - \frac{\alpha_{k+1}}{2} \|\nabla f(y^{k+1})\|^2 \tag{41}$$

To bound the first term, we use (36), $x^k = y^{k+1} + \lambda_{k+1}^{-1} p^k$. Convexity now implies $f(y^{k+1}) + \lambda_{k+1}^{-1} \langle \nabla f(y^{k+1}), p^k \rangle \leq f(x^k)$, so (40) yields

$$\frac{\alpha_{k+1}}{2} \|\nabla f(y^{k+1})\|^2 \leq f(x^k) - f(x^{k+1}) - \lambda_{k+1}^{-1} \langle \nabla f(y^{k+1}), p^k \rangle,$$

and hence,

$$\begin{aligned}(\lambda_{k+1}^2 - \lambda_{k+1}) \alpha_{k+1}^2 \|\nabla f(y^{k+1})\|^2 \leq\ & 2(\lambda_{k+1}^2 - \lambda_{k+1}) \alpha_{k+1} \left(f(x^k) - f(x^{k+1})\right) \\ & - 2(\lambda_{k+1} - 1) \alpha_{k+1} \langle \nabla f(y^{k+1}), p^k \rangle, \end{aligned} \tag{42}$$

Let us combine what we derived so far:

$$\begin{aligned}&\|p^{k+1} - x^{k+1} + x^*\|^2 - \|p^k - x^k + x^*\|^2 \\ &\quad \overset{(38),(41)}{\leq} 2(\lambda_{k+1} - 1) \alpha_{k+1} \langle \nabla f(y^{k+1}), p^k \rangle \\ &\qquad - 2\lambda_{k+1} \alpha_{k+1} (f(x^{k+1}) - f^*) + (\lambda_{k+1}^2 - \lambda_{k+1}) \alpha_{k+1}^2 \|\nabla f(y^{k+1})\|^2 \\ &\quad \overset{(42)}{\leq} -2\lambda_{k+1} \alpha_{k+1} (f(x^{k+1}) - f^*) + 2(\lambda_{k+1}^2 - \lambda_{k+1}) \alpha_{k+1} (f(x^k) - f(x^{k+1})) \\ &\quad = 2\alpha_{k+1} \lambda_k^2 (f(x^k) - f^*) - 2\alpha_{k+1} \lambda_{k+1}^2 (f(x^{k+1}) - f^*) \\ &\quad \leq 2\alpha_k \lambda_k^2 (f(x^k) - f^*) - 2\alpha_{k+1} \lambda_{k+1}^2 (f(x^{k+1}) - f^*),\end{aligned}$$

where the last inequality follows from the fact that the α_k are montonically decreasing, and the penultimate equality is a consequence of $\lambda_{k+1}^2 - \lambda_{k+1} = \lambda_k^2$.

Applying this inequality iteratively, we find

$$\begin{aligned}2\alpha_{k+1} \lambda_{k+1}^2 (f(x^{k+1}) - f^*) &\leq 2\alpha_{k+1} \lambda_{k+1}^2 (f(x^{k+1}) - f^*) + \|p^{k+1} - x^{k+1} + x^*\|^2 \\ &\leq 2\alpha_k \lambda_k^2 (f(x^k) - f^*) + \|p_k - x^k + x^*\|^2 \\ &\leq \dots \\ &\leq 2\alpha_0 \lambda_0^2 (f(x^0) - f^*) + \|p_0 - x^0 + x^*\|^2 \\ &\leq \|y^0 - x^*\|^2, \end{aligned} \tag{43}$$

where the last inequality follows from $\lambda_0 = 1$, $p^0 = (\lambda_0 - 1)(x^{-1} - x^0) = 0$, and

$$\begin{aligned}\|p^0 - x^0 + x^*\|^2 &= \|x^0 - x^*\|^2 \\ &= \|y^0 - \alpha_0 \nabla f(y^0) - x^*\|^2 \\ &= \|y^0 - x^*\|^2 + \alpha_0^2 \|\nabla f(y^0)\|^2 - 2\alpha_0 \langle \nabla f(y^0), y^0 - x^* \rangle \\ &\overset{(41)}{\leq} \|y^0 - x^*\|^2 + \alpha_0^2 \|\nabla f(y^0)\|^2 - 2\alpha_0 \left(f(x^0) - f^* + \frac{\alpha_0}{2} \|\nabla f(y^0)\|^2 \right) \\ &= \|y^0 - x^*\|^2 - 2\alpha_0 \lambda_0^2 \left(f(x^0) - f^* \right)\end{aligned}$$

To recap, (43), (34) and (32) revealed that

$$\begin{aligned}2\alpha_{k+1}\lambda_{k+1}^2 (f(x^{k+1}) - f^*) &\leq \|y^0 - x^*\|^2, \\ \lambda_{k+1} &\geq 1 + \frac{k+1}{2}, \\ \frac{1}{2L} &\leq \alpha_{k+1}.\end{aligned}$$

In combination this yields

$$L^{-1} \left(\frac{k+3}{2} \right)^2 \left(f(x^{k+1}) - f^* \right) \leq 2\alpha_{k+1} \left(1 + \frac{k+1}{2} \right)^2 \left(f(x^{k+1}) - f^* \right) \leq \|y^0 - x^*\|^2,$$

so that

$$f(x^{k+1}) - f^* \leq \frac{4L\|y^0 - x^*\|^2}{(k+3)^2},$$

as claimed by the theorem. □

References

[1] A. Beck. First Order Methods in Optimization. MOS-SIAM Series on Optimization, 2017.

[2] S.J. Wright. Optimization Algorithms for Data Analysis. http://www.optimization-online.org/DB FILE/2016/12/5748.pdf

[3] S.J. Wright. Coordinate descent algorithms. Mathematical Programming, 151:334, 2015. https://arxiv.org/abs/1502.04759

[4] D.P. Woodruff. Sketching as a Tool for Numerical Linear Algebra https://arxiv.org/abs/1411.4357

[5] L. Bottou, F.E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. SIAM Review, 59(1): 65-98, 2017.

[6] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. The Journal of Machine Learning Research, 18(1): 8194-8244, 2017.

[7] J. Wright and Y. Ma. High-dimensional data analysis with low-dimensional models. CUP, 2021.

[8] A.S. Nemirovskii and D.B. Yudin. Complexity of problems and efficiency of optimization methods. "Nauka" Moskow, 1979, (Russian).

[9] N. Shor. Minimization methods for Non-Differentiable Functions. (Springer-Verlag, Berlin, 1985)

[10] B. Polyak. Introduction to Optimization. (Optimization Software Inc., Publications Division, New York, 1987)

[11] Y. Nesterov. A Method of Solving a Convex Programming Problem with Convergence Rate $O(1/k^2)$. Soviet Math. Dokl., Vol 27 (1983), No 2.

[12] Y. Nesterov. Smooth Minimization of Non-Smooth Functions. Math. Program., Ser. A 103, 127-152 (2005).