École polytechnique fédérale de Lausanne

Mathematics of data: from theory to computation

Laboratory for information and inference systems

Fall 2021

Homework 1

Bruno Rodriguez Carrillo

Professor:
Volkan Cevher

Head TA's:
Fabian Latorre
Ali Kavis

October 31$^{\text{st}}$, 2021

1 Logistic regression

a. We have that the function is given by

$$g(u) = \log\left(1 + e^{-u}\right) \tag{1}$$

Furthermore, we are given the negative log-likelihood of $f(x)$:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \log\left(1 + e^{-b_i \mathbf{a}^T \mathbf{x}}\right) \tag{2}$$

We take the third definition of convexity given in recitation 2,that is, a function $g(x)$ is convex if and only if: a function $g \in \mathcal{C}(\mathcal{Q})$ is called convex on $\mathcal{Q}$ if for any $\mathbf{x}, \mathbf{y} \in \mathcal{Q}$:

$$\langle \nabla g(y) - \nabla g(x), y - x \rangle \geq 0 \tag{3}$$

From Eq.(1), we have:

$$\langle \nabla g(y) - \nabla g(x), y - x \rangle = \left(\nabla g(y) - \nabla g(x)\right)(y - x)$$

$$\left(-\frac{e^{-y}}{1 + e^{-y}} + \frac{-e^{-x}}{1 + e^{-x}}\right)(y - x)$$

We can divide such an equation into two cases. First, we consider $y \geq x$ and afterwards $x \geq y$. Thus, we have:

If $y \geq x$: $(y - x) \geq 0$ and $-\dfrac{e^{-y}}{1 + e^{-y}} + \dfrac{-e^{-x}}{1 + e^{-x}} = \dfrac{-e^{-y} + e^{-x}}{(1 + e^{-y})(1 + e^{-x})} \geq 0$

since $-\dfrac{1}{e^y} + \dfrac{1}{e^x} \geq 0$. Then, Eq.(3) holds.

Similarly, when $x \geq y$:

$(y - x) \leq 0$ and $-\dfrac{e^{-y}}{1 + e^{-y}} + \dfrac{-e^{-x}}{1 + e^{-x}} = \dfrac{-e^{-y} + e^{-x}}{(1 + e^{-y})(1 + e^{-x})} \leq 0$

since $-\dfrac{1}{e^y} + \dfrac{1}{e^x} \leq 0$. Then, Eq.(3) holds.

Eq.(3) equals 0 when $x = y$. As a result, Eq.(1) is convex.

Since Eq.(2) is the sum of convex functions, it is convex too.

A function can be convex but this does not mean that there is always a minimum. For instance, if we consider the 1-D exponential function $h(x) = e^x$, whose domain is $\mathbb{R}$ or $(-\infty, \infty)$; such a function is convex but it is minimum is not defined over its domain. For a minimum to exist the sufficient conditions are continuity of the function and closed and bounded domain. Convexity is neither necessary nor sufficient for the minimum to exist.

b. Let us consider an non-empty set $A \subset \mathbb{R}$. The infimum of A ($\inf A$) is the greatest lower bound for the set, not the greatest lower bound contained in the set. It is the greatest of all of the lower bounds for the set. It may or may not be a member of the set. The minimum of A ($\min A$) is the least member contained in A, assuming it exists. Mathematically, we say that $\inf A$ belongs to A if and only if $\inf A = \min A$.

   As a manner of example, once again, if we consider the exponential function $h(x) = e^x$, it is a convex function but it does not reach its infimum over its domain, which is not bounded.

c. Show that if there exists $\mathbf{x}_0$ such that

$$b_i \mathbf{a}_i^T \mathbf{x}_0 > 0, , \forall i \in \{1, \dots n\}$$

   then the function f does not attain its infimum.

   Let us first assume that such a point $\mathbf{x}_0$ is the one for which f attain its infimum. Now, we consider:

$$f(\mathbf{x}_0) = \sum_{i=1}^{n} \log\left(1 + \exp(-b_i \mathbf{a}^T \mathbf{x}_0)\right) \text{ and } f(2\mathbf{x}_0) = \sum_{i=1}^{n} \log\left(1 + \exp(-b_i \mathbf{a}^T 2\mathbf{x}_0)\right)$$

   We observe that $f(\mathbf{x}_0) > f(2\mathbf{x}_0)$ and more general, we have:

$$\lim_{\alpha \to +\infty} f(\alpha \mathbf{x}_0) = \lim_{\alpha \to +\infty} \sum_{i=1}^{n} \log\left(1 + \exp(-\alpha b_i \mathbf{a}^T \mathbf{x}_0)\right) = 0$$

   Equivalently,

$$f(\mathbf{x}_0) > f(2\mathbf{x}_0) > f(3\mathbf{x}_0) > \cdots > f(\alpha \mathbf{x}_0)$$

   Then we have a contradiction since $\mathbf{x}_0$ is supposed to be the infimum of the function. In other words, no matter what value of α one selects, one can always take the objective function upwards by increasing α towards infinity and thus the objective function has no maximum, and attempting to find one iteratively will keep increasing α.

d. We consider the function

$$f_\mu(\mathbf{x}) = f(\mathbf{x}) + \frac{\mu}{2} \|\mathbf{x}\|_2^2 \tag{4}$$

   where $f(\mathbf{x})$ is given in point a).

   From recitation 2, we use the chain rule via Jacobians, which states that if $\circ$ denotes the functional composition: $g \circ f := g(f(\mathbf{x}))$. If $g \circ f$ is differentiable at $\mathbf{x}$, then the following holds:

$$\mathbf{J}_{g \circ f}(\mathbf{x}) = \mathbf{J}_g(f(\mathbf{x}))\mathbf{J}_f(\mathbf{x})$$

   As a result, taking the example of logistic loss presented in Recitation 2, we have the following:

$$h(\mathbf{x}) = \mathbf{a}^T\mathbf{x}, \text{ whose Jacobian is } \mathbf{J}_h(\mathbf{x}) = \mathbf{a}^T$$

and

$$g(u) = \log\left(1 + e^{-bu}\right), \text{ whose 1x1 Jacobian is } \mathbf{J}_g(u) = -b\frac{\exp(-bu)}{1+\exp(-bu)}$$

By the chain rule, we have:

$$\mathbf{J}_f(\mathbf{x}) = \mathbf{J}_g(h(\mathbf{x}))\mathbf{J}_h(\mathbf{x}) = -b\frac{\exp(-b\mathbf{a}^T\mathbf{x})}{1+\exp(-b\mathbf{a}^T\mathbf{x})}\mathbf{a}^T$$

Since $f(\mathbf{x})$ is real-valued, we compute its gradient by taking the transpose of $\mathbf{J}_f(\mathbf{x})$. Besides, the derivative on the right-hand of Eq.(4) is trivially computed as $\mu\mathbf{x}$ and differentiation is linear; then we have, using the definition of $\sigma(x)$:

$$\nabla f_\mu(\mathbf{x}) = \sum_{i=1}^{n} -b_i\sigma\left(-b_i\mathbf{a}_i^T\mathbf{x}\right)\mathbf{a}_i + \mu\mathbf{x} \tag{5}$$

e. From the previous computations, we observe that $\nabla f_\mu(\mathbf{x}) \in \mathbb{R}^d$, then to compute the Hessian matrix $\nabla^2 f_\mu(\mathbf{x})$ by taking the gradient of $\nabla f_\mu(\mathbf{x})^T$; thus we have:

$$\nabla^2 f_\mu(\mathbf{x}) = \nabla\left(\nabla f_\mu(\mathbf{x})^T\right) = \nabla\left(-b_i\sigma\left(-b_i\mathbf{a}_i^T\mathbf{x}\right)\mathbf{a}_i\right) + \nabla\mu\mathbf{x}^T$$

We compute

$$\nabla\sigma\left(-b_i\mathbf{a}_i^T\mathbf{x}\right) = \frac{-b_i\mathbf{a}_i\exp(b_i\mathbf{a}_i^T\mathbf{x})}{\left(1+\exp(b_i\mathbf{a}_i^T\mathbf{x})\right)^2} = -b_i\mathbf{a}_i\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\left(1-\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\right)$$

Then, we obtain

$$\nabla^2 f_\mu(\mathbf{x}) = b_i^2\mathbf{a}_i\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\left(1-\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\right)\mathbf{a}_i^T + \mu\mathbf{I}$$

where we have to use the fact that $b_i^2 = 1$ for all i.

It is important to notice that we computed $\nabla^2 f_\mu(\mathbf{x})$ for one element of the sum, since the derivative of the sum is the sum of the derivatives. Consequently, we have:

$$\nabla^2 f_\mu(\mathbf{x}) = \sum_{i=1}^{n}\mathbf{a}_i\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\left(1-\sigma(-b_i\mathbf{a}_i^T\mathbf{x})\right)\mathbf{a}_i^T + \mu\mathbf{I} \tag{6}$$

f. From point a), we have already proven that $f_\mu(\mathbf{x})$ is convex since it is the sum of two convex functions. And we have to prove that f_μ is μ-strongly convex, that is, from the lecture notes:

$$f_\mu(\mathbf{y}) \geq f_\mu(\mathbf{x}) + \langle\nabla f_\mu(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\mu}{2}\|\mathbf{y}-\mathbf{x}\|_2^2 \tag{7}$$

Then we have that the following property of convex functions holds:

$$f_\mu(\mathbf{y}) = f(\mathbf{y}) + \frac{\mu}{2}\|\mathbf{y}\|_2^2 \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y}\|_2^2$$

If we add and subtract the term $\frac{\mu}{2}\|\mathbf{x}\|_2^2$ on the left hand side, we have

$$f_\mu(\mathbf{y}) \geq f_\mu(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y}\|_2^2 - \frac{\mu}{2}\|\mathbf{x}\|_2^2$$

Now, we use the fact that $\nabla f_\mu(\mathbf{x}) = \nabla f(\mathbf{x}) + \mu\mathbf{x}$, we obtain:

$$f_\mu(\mathbf{y}) \geq f_\mu(\mathbf{x}) + \langle \nabla f_\mu(\mathbf{x}) - \mu\mathbf{x}, \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y}\|_2^2 - \frac{\mu}{2}\|\mathbf{x}\|_2^2$$

$$f_\mu(\mathbf{y}) \geq f_\mu(\mathbf{x}) + \langle \nabla f_\mu(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \langle \mu\mathbf{x}, \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y}\|_2^2 - \frac{\mu}{2}\|\mathbf{x}\|_2^2$$

Besides, we know that $\langle \mu\mathbf{x}, \mathbf{y} - \mathbf{x} \rangle = \mu\langle \mathbf{x}, \mathbf{y} \rangle - \mu\langle \mathbf{x}, \mathbf{x} \rangle = \mu\langle \mathbf{x}, \mathbf{y} \rangle - \mu\|\mathbf{x}\|_2^2$. As a result, we have

$$f_\mu(\mathbf{y}) \geq f_\mu(\mathbf{x}) + \langle \nabla f_\mu(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\left(\|\mathbf{y}\|_2^2 - 2\langle \mathbf{x}, \mathbf{y} \rangle + \|\mathbf{x}\|_2^2\right)$$

The result in Eq.(7) follows by noticing that $\|\mathbf{y}\|_2^2 - 2\langle \mathbf{x}, \mathbf{y} \rangle + \|\mathbf{x}\|_2^2 = \|\mathbf{y} - \mathbf{x}\|_2^2$.

g. From the given information, we know that $\mathbf{a}_i\mathbf{a}_i^T \in \mathbb{R}^{p\times p}$, which corresponds to the outer product of $\mathbf{a}_i$ with itself. We can define a matrix $M \in \mathbb{R}^{p\times p} = \mathbf{a}_i\mathbf{a}_i^T$. Then we have:

$$M\mathbf{a}_i = \mathbf{a}_i\mathbf{a}_i^T\mathbf{a}_i = \mathbf{a}_i^T\mathbf{a}_i\mathbf{a}_i = \lambda\mathbf{a}_i$$

We now recall that by definition a eigenvalue of any matrix B satisfies $B = \lambda x$ and observe that $\mathbf{a}_i^T\mathbf{a}_i \in \mathbb{R}$, which is in turn equal to $\mathbf{a}_i^T\mathbf{a}_i = \|\mathbf{a}_i\|_2^2$. Thus, we obtain

$$\lambda(\mathbf{a}_i\mathbf{a}_i^T) = \lambda_{max}(\mathbf{a}_i\mathbf{a}_i^T) = \|\mathbf{a}_i\|_2^2$$

We should mention that since the matrix M is the result of an outer product of the same vector $\mathbf{a}_i$, it has rank 1 and all its columns are linear combinations of its first column. Then, M has one eigenvalue only.

From point e), Eq.(6), we have that

$$\nabla^2 f_\mu(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{a}_i\beta_i\mathbf{a}_i^T + \mu\mathbf{I}$$

where $\beta_i = \sigma(-b_i\mathbf{a}_i^T\mathbf{x})\left(1 - \sigma(-b_i\mathbf{a}_i^T\mathbf{x})\right) \in \mathbb{R}$.

Then we notice that $0 < \beta_i < 1$ and since $\nabla^2 f_\mu(\mathbf{x})$ is the sum of two matrices, one of which is diagonal, we can compute the eigenvalues of $\nabla^2 f_\mu(\mathbf{x})$ as the following sum:

$$\lambda\left(\nabla^2 f_\mu(\mathbf{x})\right) = \lambda\left(\beta_i \sum_{i=1}^{n} \mathbf{a}_i \mathbf{a}_i^T\right) + \mu\lambda\left(\mathbf{I}\right) \leq \lambda\left(\sum_{i=1}^{n} \mathbf{a}_i \mathbf{a}_i^T\right) + \mu\lambda\left(\mathbf{I}\right)$$

From the previous result, the following holds:

$$\lambda_{max}\left(\nabla^2 f_\mu(\mathbf{x})\right) \leq \sum_{i=1}^{n} \|\mathbf{a}_i\|_2^2 + \mu$$

From recitation 2, we have that a function $f_\mu(\mathbf{x})$ is L-smooth if:

$$\nabla^2 f_\mu(\mathbf{x}) \leq L \cdot \mathbf{I}_{p\times p}$$

with $\mathbf{I}_{p\times p}$ the $p \times p$ identity matrix. Moreover, we observe that the sum in the expression above corresponds to the definition of the Frobenius norm $\|\cdot\|_F^2$, we have that:

$$L = \|\mathbf{A}\|_F^2 + \mu$$

with $\mathbf{A}$ defined in the homework sheet, as expected.

2 Numerical methods for Logistic regression

2.1 First-order methods

After completing the missing parts in the provided code, we have the following plots:

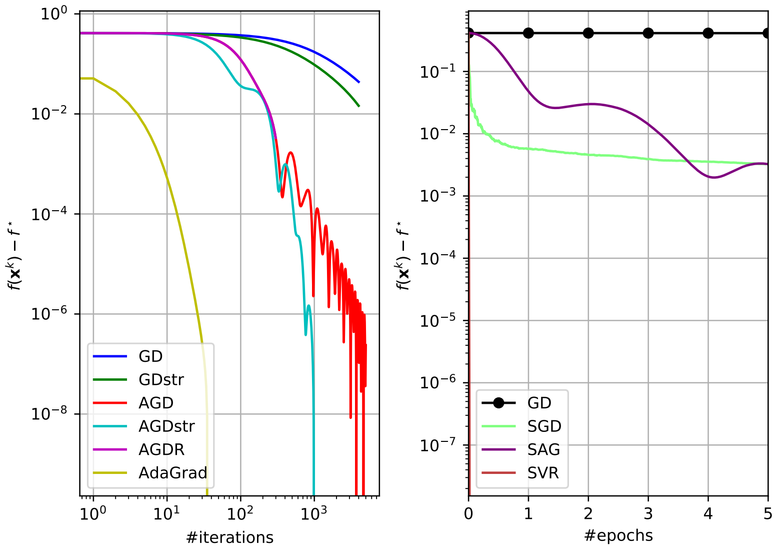

Figure 1: Convergence rates first order methods

And the corresponding errors are summarized in as follows:

Numerical results	
Method	Error
GD	0.13868
GDstr	0.09489
AGD	0.05839
AGDstr	0.05839
AGDR	0.07299
AdaGrad	0.05839
SGD	0.07299
SAG	0.05109
SVR	0.05839

Table 1: Error for plots in Fig.(1).

2.2 Stochastic gradient methods

a. For this question, we define $f(\mathbf{x})$ as

$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\Big\{\log\big(1+\exp\big(-b_i\mathbf{a}_i^T\mathbf{x}\big)\big)+\frac{\mu}{2}\|\mathbf{x}\|^2\Big\} = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})$$

And as we have explained before, its gradient is:

$$\nabla f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\Big\{-b_i\sigma\left(-b_i\mathbf{a}_i^T\mathbf{x}\right)\mathbf{a}_i+\mu\mathbf{x}\Big\} = \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(\mathbf{x})$$

where the index i is picked uniformly at random and $i \in \{1, ..., n\}$. Then, if when we compute the expected value of $\nabla f_i(\mathbf{x})$, we have:

$$\mathbb{E}\big[\nabla f_i(\mathbf{x})\big] = \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x})$$

which shows that $\nabla f_i(\mathbf{x})$ is an unbiased estimator of $\nabla f(\mathbf{x})$, as expected.

From point 1. g), we have that $L = \sum_{i=1}^{n}\|\mathbf{a}_i\|_2^2 + \mu$; however, in this case we have one sample out of n only, which is selected at random and then the sum is over a single element. As a consequence:

$$L(f_i) = \max_{i\in\{1...n\}}\|\mathbf{a}_i\|_2^2 + \mu$$

2.3 Proximal methods

In a similar fashion, we have the following plots:

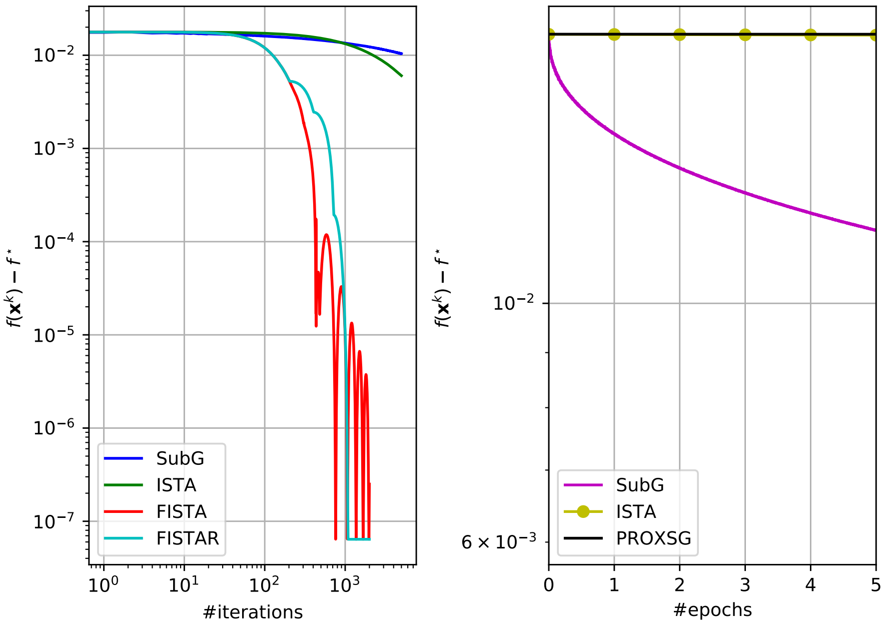

Figure 2: Convergence rates proximal, L-1 norm

with errors summarized as follows:

Numerical results	
Method	Error
SubG	0.12408
ISTA	0.14598
FISTA	0.14598
FISTAR	0.14598
PROXSG	0.16058

Table 2: Error for plots in Fig.(2).

And for L-2 norm, we have:

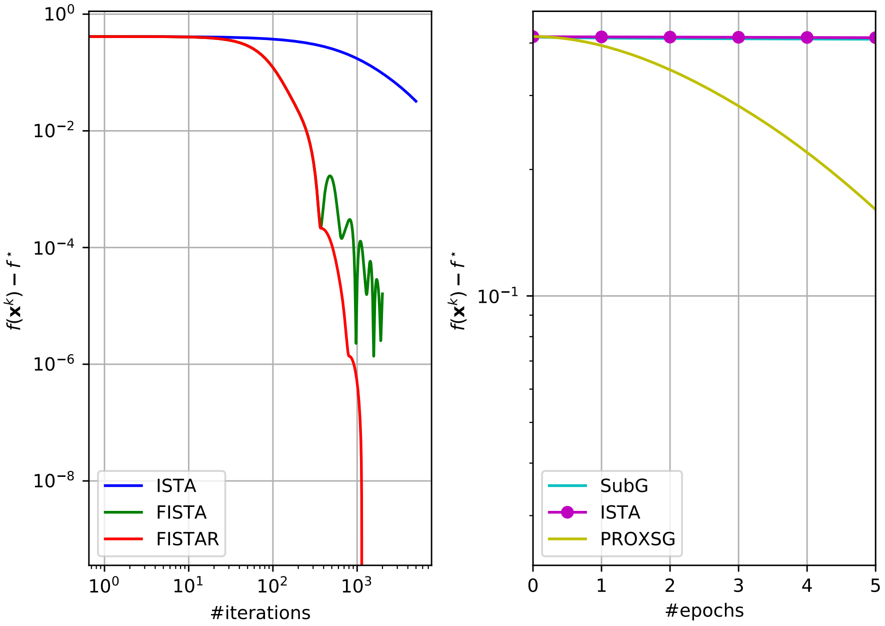

Figure 3: Convergence rates proximal, L-2 norm

and errors given as:

Numerical results	
Method	Error
ISTA	0.11678
FISTA	0.06569
FISTAR	0.05839
PROXSG	0.12408

Table 3: Error for plots in Fig.(3).

For Fig.(1), Fig.(2) and Fig.(3), along with their corresponding charts, the error is with respect to the 0-1 loss.

b. Given $g : \mathbb{R}^d \Rightarrow \mathbb{R}, g(\mathbf{x}) = \|\mathbf{x}\|_1$, we have to show that its proximal operator can be written as:

$$\text{prox}_{\lambda g}(\mathbf{x}) = \max\left(|\mathbf{x}| - \lambda, 0\right) \circ \text{sign}\left(\mathbf{x}\right), \mathbf{x} \in \mathbb{R}^d \tag{8}$$

where $\circ$ represents $(\mathbf{x} \circ \mathbf{y})_i = x_i y_i$ and the max, sign and $|\cdot|$ operators are applied element-wise. To proceed with the proof, we take the definition of the proximal operator of g given in the homework 1 sheet:

$$\text{prox}_{\lambda g}(\mathbf{x}) = \arg\min_{\mathbf{y} \in \mathbb{R}^d} \left\{\lambda g(\mathbf{x}) + \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_2^2\right\}$$

The function g is convex but not differentiable, thus we compute its subdifferential $\partial g(\mathbf{x})$. Moreover, the function to minimize is convex since it is the sum of two convex functions and a point $\mathbf{x}^* \in \mathbb{R}^d$ is a minimizer if and only if f is subdifferentiable at $\mathbf{x}^*$ and $\mathbf{0} \in \partial f(\mathbf{x}^*)$, which is known as the first order optimality condition and calling $f(\mathbf{x}) = \lambda g(\mathbf{x}) + \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_2^2$.

First, we compute the subdifferential of $g(\mathbf{x})$. The g function can be seen as a generalization of the $|\cdot|$ function; for such a function, we obtain:

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

Taking this as a starting point, we can thus write the subdifferential operator of the g function as:

$$\partial g(\mathbf{x}) = \left\{ \mathbf{p} = (p_i, \ldots p_d)^T : p_i \in \text{sign}(x_i) \text{ if } x_i \neq 0 \text{ and } p_i \in [-1, 1] \text{ if } x_i = 0 \right\}$$

Afterwards, we have that $\mathbf{0} \in \partial g(\mathbf{x}) + \frac{1}{\lambda}(\mathbf{x} - \mathbf{y})$; as a result we can state that $\partial g(\mathbf{x}) = \text{sign}(x_i)$ if $x_i \neq 0$ and $\partial g(\mathbf{x}) \in [-1, 1]$ if $x_i = 0$. Thus an optimal solution $\mathbf{x}^* = \mathbf{y} - \lambda \partial g(\mathbf{x})$ can be computed by:

$$(\mathbf{x}^*)_i = \begin{cases} y_i - \lambda & , y_i > \lambda \\ 0 & , \lambda \geq y_i \leq \lambda \\ y_i + \lambda & , y_i < \lambda \end{cases}$$

since we apply such criterion element-wise, we obtain Eq.(8), as requested.

2.4 Convergence rates

a. From the given information, we know that our problem is L-smooth and μ-strongly convex, then for the gradient method, we implement:

$$\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \leq \left(\frac{L - \mu}{L + \mu}\right)^{k/2} \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2$$

and for the stochastic gradient method we have:

$$\|\mathbf{x}^k - \mathbf{x}^*\|_2^2 \leq \left(\frac{L - \mu}{L + \mu}\right)^{k} \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2$$

as a result, we have:

$$\text{theoretical rate for GD } = \left(\frac{L - \mu}{L + \mu}\right)^{k/2} \text{ and for SGD } = \left(\frac{L - \mu}{L + \mu}\right)^{k}$$

After filling in the code, we have the following plots:

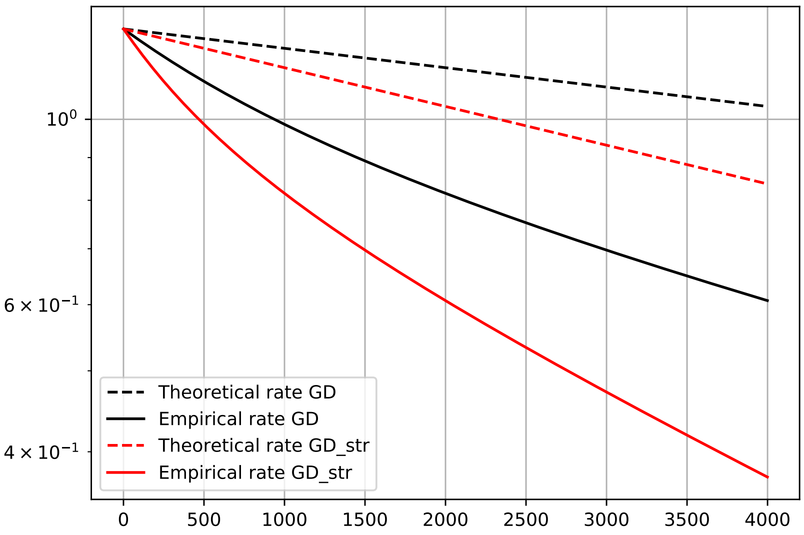

Figure 4: Convergence rates GD and GDstr

b. We begin by reporting the values of the constants during the simulations: $L = 1872.441$ and $\mu = 0.1$. In the plot above, as the number of iterations increases, the change in the slope is quite small, that is, in 4000 iterations, there is a reduction in the slope of $4 \cdot 10^{-1}$. This means that practically the line is an horizontal line. As a result, the convergence rate for both methods is sub-linear.

Let us consider the ratio $\left(\frac{L-\mu}{L+\mu}\right) \approx 1$. Based on recitation 2, given an numerical method, if the sequence $\mathbf{x}^k$ generated by the algorithm satisfies:

$$\lim_{k\to\infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} = 1$$

then the method is said to converge sub-linearly. In our case, we have that:

$$\text{GD: } \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} \leq \left(\frac{L-\mu}{L+\mu}\right) \approx 1 \text{ and GDstr: } \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} \leq \left(\frac{L-\mu}{L+\mu}\right)^{1/2} \approx 1$$

Consequently, we confirm that the rates are sublinear and then the numerical results are consistent with the theoretical rates.

It is also important to say that both convergence rates are not quadratic since there does exist a number $r \in (0, 1)$ such that:

$$\lim_{k \to \infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|^2} = r$$

Furthermore, the GD μ-strongly method converges twice as fast as the GD method, as expected.

c. As stated in the homework sheet, we assume that the observed rates will be of the form $\|\mathbf{x}^k - \mathbf{x}^*\|_2 = a \cdot b^k$, then we have:

Now, if we are working on a $\left(k, \|\mathbf{x}^k - \mathbf{x}^*\|_2\right)$, we can compute the values of a and b taking into account the following:

$$\log\left(\|\mathbf{x}^k - \mathbf{x}^*\|_2\right) = \log a + k \log b = \text{ intercept} + k \cdot \text{slope} \tag{9}$$

where the first term on the right hand side corresponds to the vertical intercept of a line in the above scale and the second term on the right hand side represents the slope of the line multiplied by k. We should comment on Eq.(9), which represents a line in the mentioned scale and k is the iteration of our algorithm. Consequently, the values of a and b can be computed from a linear fit. Then we have the following linear regression coefficients:

$$\log a = \text{intercept} \to a = \exp\left(\text{intercept}\right) \text{ and } \log b = \text{slope} \to b = \exp\left(\text{slope}\right)$$

As a result, in the attached python code, we compute such parameters for the GD and GDstr using the above equations.

d. We have the following results regarding theoretical and empirical speed of convergence:

GD method:

(a) numerical results: *intercept* = 1.196692, *slope* = 0.999820

(b) theoretical results: *intercept* = 1.282909, *slope* = 0.999893

GDstr method:

(a) numerical results: *intercept* = 1.119548, *slope* = 0.999711

(b) theoretical results: *intercept* = 1.2829, *slope* = 0.999893

And we have the following plot when we add the linear fit:

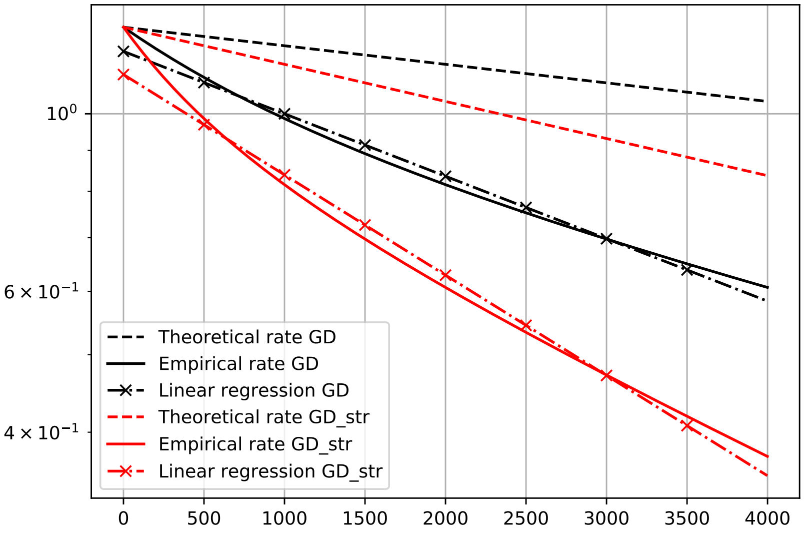

Figure 5: Convergence rates GD and GDstr, linear fit

We observe that as iteration increases, the empirical rates better approximate the theoretical ones. In order to observe better similarities, we would need to run our codes for more iterations.

3 Image reconstruction

See codes if applies.

3.1 Wavelets

See codes if applies.

3.2 Total variation

See codes if applies.

3.3 Image in-painting

a. We have that the following functions from which we will compute their gradients:

$$f_{l_1}(\boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{b} - \mathbf{P}_{\boldsymbol{\Omega}}\mathbf{W}^T\boldsymbol{\alpha}\|_2^2 \tag{10}$$

$$f_{TV}(\mathbf{x}) = \frac{1}{2}\|\mathbf{b} - \mathbf{P_\Omega x}\|_2^2 \tag{11}$$

with $\mathbf{b} = \mathbf{P_\Omega x}, \mathbf{P_\Omega} \in \mathbb{R}^{n\times p}, \mathbf{x} \in \mathbb{R}^p$.

We can apply the same procedure as the one used before and compute the gradient of both functions given above using the chain rule for Jacobians. Hence, we have for Eq.(10).

We define a couple of functions as:

$$f(\boldsymbol{\alpha}) = \mathbf{b} - \mathbf{P_\Omega W}^T\boldsymbol{\alpha} \Rightarrow \mathbf{J}_f(\boldsymbol{\alpha}) = -\mathbf{P_\Omega W}^T$$

and

$$g(\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{\alpha}\|_2^2 \Rightarrow \nabla g(\boldsymbol{\alpha}) = \boldsymbol{\alpha} \Rightarrow \mathbf{J}_g(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T$$

Combining both results, we have:

$$\mathbf{J}_{g\circ f}(\boldsymbol{\alpha}) = \mathbf{J}_g(f(\boldsymbol{\alpha}))\mathbf{J}_f(\boldsymbol{\alpha}) = \left(\mathbf{b} - \mathbf{P_\Omega W}^T\boldsymbol{\alpha}\right)^T\left(-\mathbf{P_\Omega W}^T\right)$$

To compute the gradient, we take the transpose of the above expression; thus:

$$\nabla f_{l_1}(\boldsymbol{\alpha}) = -\mathbf{W P_\Omega}^T\left(\mathbf{b} - \mathbf{P_\Omega W}^T\boldsymbol{\alpha}\right) \tag{12}$$

Following a similar procedure to compute the gradient of Eq.(11), we notice that in this case the operator $\mathbf{W} = \mathbf{1}$, then we have the following:

$$\nabla f_{TV}(\boldsymbol{x}) = -\mathbf{P_\Omega}^T\left(\mathbf{b} - \mathbf{P_\Omega}\boldsymbol{\alpha}\right) \tag{13}$$

b. To compute the Lipschitz constant of Eq.(12), we make use of the definition given in the lecture notes:

$$\|\nabla f_{l_1}(\boldsymbol{\alpha}) - \nabla f_{l_1}(\boldsymbol{\beta})\|_2 \le L_{l_1}\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2 \tag{14}$$

for $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{R}^n$. From the above expression for $\nabla f_{l_1}(\boldsymbol{\alpha})$, we obtain:

$$\|\nabla f_{l_1}(\boldsymbol{\alpha}) - \nabla f_{l_1}(\boldsymbol{\beta})\|_2 = \|-\mathbf{W P_\Omega}^T\left(\mathbf{b} - \mathbf{P_\Omega W}^T\boldsymbol{\alpha}\right) + \mathbf{W P_\Omega}^T\left(\mathbf{b} - \mathbf{P_\Omega W}^T\boldsymbol{\beta}\right)\|_2$$

$$\|\nabla f_{l_1}(\boldsymbol{\alpha}) - \nabla f_{l_1}(\boldsymbol{\beta})\|_2 = \|\mathbf{W P_\Omega}^T\mathbf{P_\Omega W}^T\boldsymbol{\alpha} - \mathbf{W P_\Omega}^T\mathbf{P_\Omega W}^T\boldsymbol{\beta}\|_2$$

We can bound that as:

$$\|\mathbf{W P_\Omega}^T\mathbf{P_\Omega W}^T\boldsymbol{\alpha} - \mathbf{W P_\Omega}^T\mathbf{P_\Omega W}^T\boldsymbol{\beta}\|_2 \le \|\mathbf{W P_\Omega}^T\mathbf{P_\Omega W}^T\|_2\|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2$$

Similar result can be computed for Eq.(13); consequently, we have the following Lipschitz constants, where we have taken into account the fact that the matrix $\mathbf{W}$ is orthonormal:

$$L_{l_1} = \|\mathbf{W_\Omega P_\Omega}^T \mathbf{P_\Omega W_\Omega}^T\|_2 \text{ and } L_{TV} = \|\mathbf{P_\Omega}^T \mathbf{P_\Omega}\|_2 \tag{15}$$

We recall now the definition of the operator norm. We restrict ourselves to the real numbers and to the L-2 norm. Any $m \times n$ matrix $\mathbf{A}$ induces a linear operator from $\mathbb{R}^n$ to $\mathbb{R}^m$ with respect to the standard basis and then one defines the induced norm or norm operator on the space $\mathbb{R}^{m\times n}$ of all $m \times n$ matrices as follows.

$$\|\mathbf{A}\|_2 = \sup\left\{\|\mathbf{Ax}\|_2 : \mathbf{x} \in \mathbb{R}^n \text{ with } \|\mathbf{x}\|_2 = 1\right\}$$

From the above definition, we observe that the induced norm is smaller or equal to 1 since it is a projection of a unitary vector. As a result, both of our constants in Eq.(15) are equal to 1, that is:

$$L_{l_1} = L_{TV} = 1 \tag{16}$$

It is worth mentioning that such a value of L is the smallest one that one can have and larger values also satisfy Eq.(14).

c. To estimate the optimal values of the parameters λ_{ℓ_1} and λ_{TV}, we run our codes with for 20 equally spaced intervals I_i in log scale for both axes between -4 and -0.10 and create one plot for each norm ℓ_l-norm and TV-norm and each of the three provided images. That is:

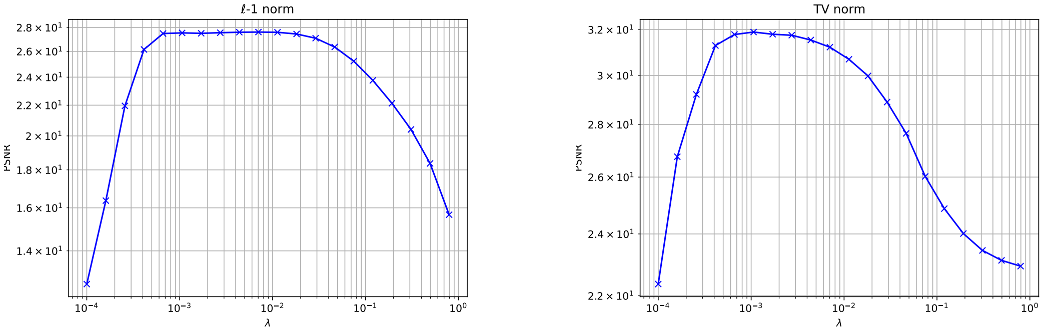

Figure 6: PSNR plots for gandalf picture.

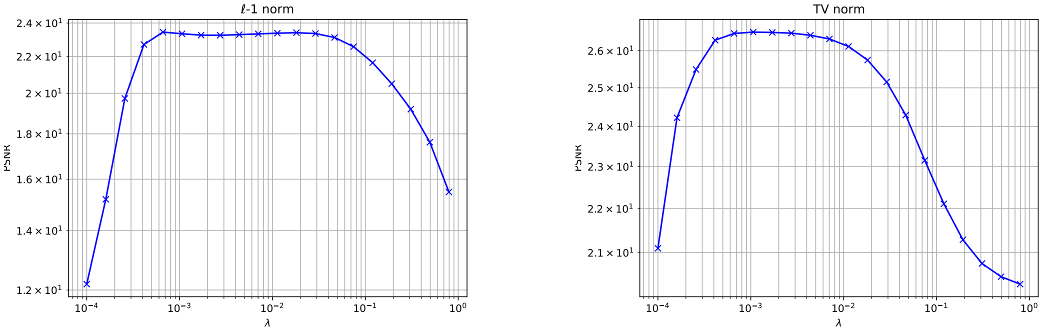

Figure 7: PSNR plots for Lauterbrunnen picture.

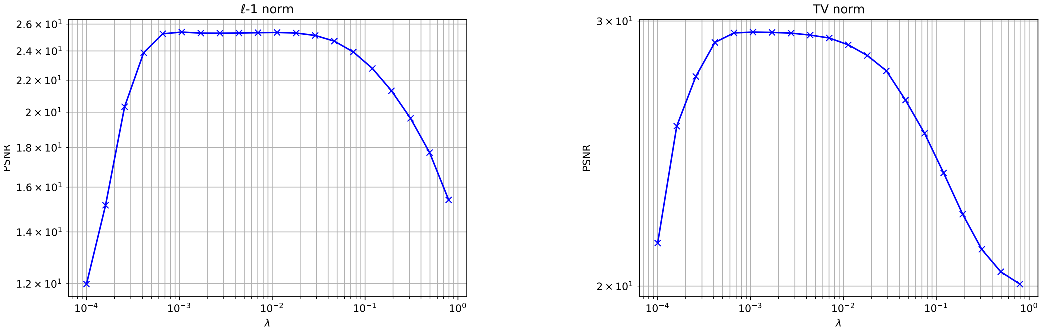

Figure 8: PSNR plots for Lena picture.

The values for λ for each norm are as follows.

(a) Gandalf $\lambda_{L1} = 0.00703669$, Lauterbrunnen $\lambda_{L1} = 0.00066229$ and Lena $\lambda_{L1} = 0.00106246$

(b) Gandalf $\lambda_{TV} = 0.00106246$, Lauterbrunnen $\lambda_{TV} = 0.00066229$ and Lena $\lambda_{TV} = 0.00170442$

The ordering follows the same as the one presented next. As a consequence, we have the following reconstructed images, which correspond to the the ones using the optimized λ_{ℓ_1} and λ_{TV}.

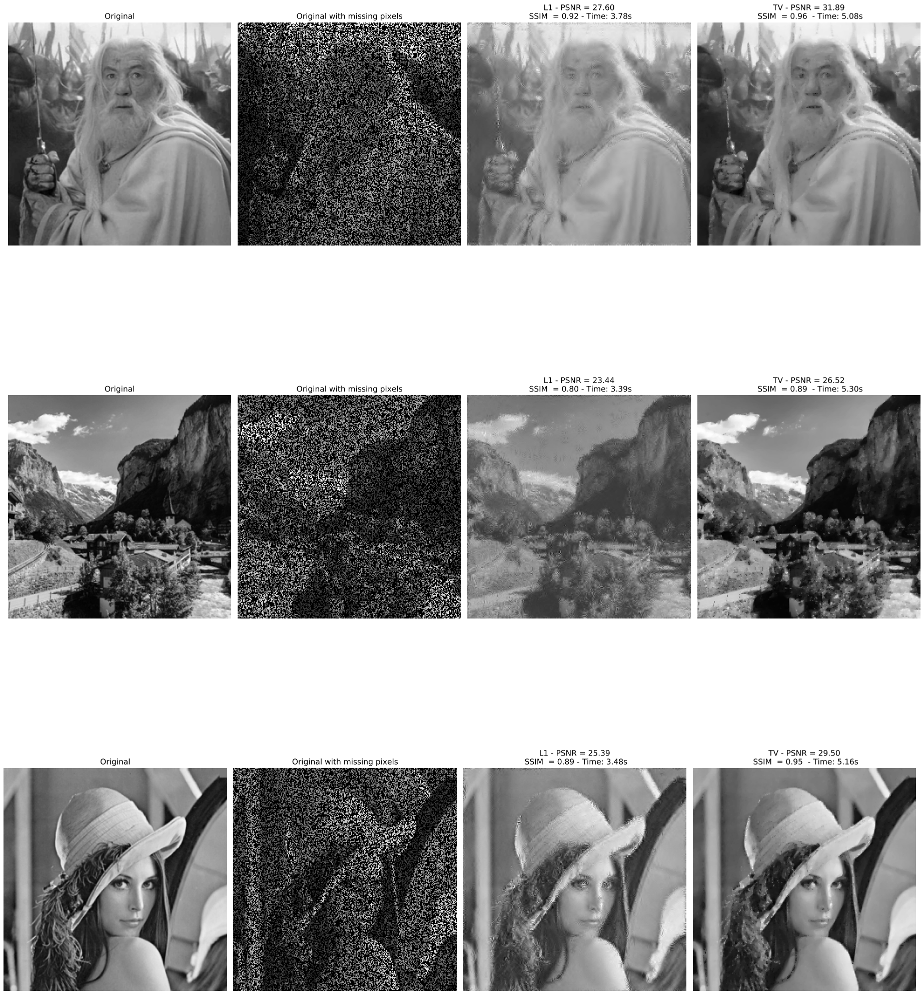
Original
Original with missing pixels
L1 - PSNR = 27.60
SSIM = 0.92 - Time: 3.78s
TV - PSNR = 31.89
SSIM = 0.96 - Time: 5.08s
Original
Original with missing pixels
L1 - PSNR = 23.44
SSIM = 0.80 - Time: 3.39s
TV - PSNR = 26.52
SSIM = 0.89 - Time: 5.30s
Original
Original with missing pixels
L1 - PSNR = 25.39
SSIM = 0.89 - Time: 3.48s
TV - PSNR = 29.50
SSIM = 0.95 - Time: 5.16s

Figure 9: Reconstructed pictures: Gandalf, Lauterbrunnen and Lena.

For all previous images, the value of PNSR and the norm are indicated in each, and the method of reconstruction was FISTA.