COMP0083: Advanced Topics in Machine Learning 2022/2023
Introduction to Convex Optimization

Coursework

Due Date: **9 January 2022**
Instructor: Massimiliano Pontil
TA: Isak Texas Falk
Coursework based on the notes by Saverio Salzo

Instructions There are 4 sections. Two assess the knowledge of the theory and two require implementing algorithms. Provide sufficient explanations for all solutions. For the coding parts the following libraries are required: `numpy, sklearn, matplotlib`.

Deadlines Submit your report by January 9 2022.

Policies Late submissions: past the online due date, late submissions will be penalized by 20% of the original total score per late day (e.g., 40% for 2 days). Excused extensions will be given only for significant issues and if requested well in advance the due date.

---

1 Questions with multiple answers [20%]

Question 1 [4%] Which one of these is a convex function?

a) $\max\{ax + b, x^4 - 5, e^{x^2}\}$

b) $\min\{ax + b, x^4 - 5, e^{x^2}\}$

c) $ax + b + x^4 - 5 - e^{x^2}$.

Question 2 [4%] Consider the function

$$f(x) = \begin{cases} -x & \text{if } x \in]-1, 0] \\ x^2 & \text{if } x \geq 0. \end{cases}$$

Indicate which one of the two graphs below represents the subdifferential of f.

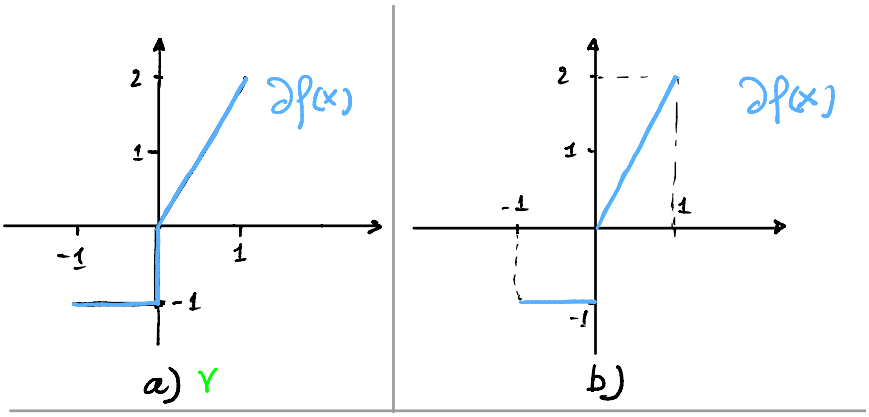

Figure 1: Question 2

Question 3 [4%] The gradient of $f(x) = \langle Ax, x\rangle + \langle x, b\rangle + c$, where A is a square matrix, not necessarily symmetric is

a) $A^*x + Ax + b$

b) $2Ax + b$

c) $A^* + b$, where A^* denotes the transpose of A.

Question 4 [4%] Let $g \in \Gamma_0(\mathbb{R})$. The Fenchel conjugate of $f(x) = g(2x)$ is

a) $f^*(u) = g^*(u/2)$

b) $f^*(u) = g^*(2u)$

c) $f^*(u) = g^*(u)$.

Question 5 [4%] Referring to the ridge regression problem

$$\min_{w\in H} \frac{1}{2n}\sum_{i=1}^{n}(y_i - \langle w, \Lambda(x_i)\rangle)^2 + \frac{\lambda}{2}\|w\|^2,$$

and denoting by K the Gram matrix of the data, indicate the solution of the dual problem

a) $\bar{u} = (\mathsf{K} + \lambda n \mathsf{Id})y$

b) $\bar{u} = (\mathsf{K}^2 + \lambda n \mathsf{Id})^{-1}y$

c) $\bar{u} = (\mathsf{K} + \lambda n \mathsf{Id})^{-1}y$.

2 Theory on convex analysis and optimization [30%]

Problem 1. [4%] Compute (showing a complete derivation) the Fenchel conjugates of the following functions.

1. $f\colon \mathbb{R} \to]-\infty, +\infty]$, with $f(x) = \begin{cases} +\infty & \text{if } x \leq 0 \\ -\log x & \text{if } x > 0 \end{cases}$.
2. $f(x) = x^2$.
3. $\iota_{[0,1]}$ (the indicator function of the interval $[0, 1]$).

Problem 2. [7%] Let $f\colon X \to]-\infty, +\infty]$ be a proper convex function.

1. Prove by induction the Jensen's inequality, that is
$$\sum_{i=1}^{n} f\left(\sum_{i=1}^{n} \lambda_i x_i\right) \leq \sum_{i=1}^{n} \lambda_i f(x_i),$$
for all $x_1, \ldots, x_n \in X$ and for all $\lambda_1, \ldots, \lambda_n \in \mathbb{R}_+$ with $\sum_{i=1}^{n} \lambda_i = 1$.
2. Using the characterization for differentiable functions prove that the function $-\log$ is convex
3. Applying Jensen's inequality to $-\log$, prove the following arithmetic-geometric inequality
$$\sqrt[n]{x_1 \cdots x_n} \leq \frac{1}{n} \sum_{i=1}^{n} x_i,$$
for all $x_1, \ldots, x_n \in \mathbb{R}^+$.

Problem 3. [4%] Given a polytope $C = \mathrm{co}(a_1, \ldots, a_m)$ in X. Prove that the maximum of a convex function f on C is attained at one of the vertices $a_1, \ldots, a_m$.

Problem 4. [2%] Prove that the function $f(x, y) = \|x - 2y\|^2$ is (jointly) convex

Problem 5. [4%] Provide minimal sufficient conditions for the existence and uniqueness of minimizers for a convex function $f\colon X \to]-\infty, +\infty]$.

Problem 6. [9%] Consider the optimization problem.
$$\min_{\|Ax-b\|_\infty \leq \varepsilon} \frac{1}{2} \|x\|^2,$$
where $\varepsilon > 0$, $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$. Solve the following points.

1. Compute the dual problem (using the Fenchel-Rockafellar duality theory). [Hint: put the problem in the form $f(x) + g(Ax)$]
2. Does strong duality hold? Justify the answer. [Hint: use the qualification condition]
3. Write the KKT conditions.
4. Derive a rate of convergence on the primal iterates from the application of FISTA on the dual problem. [Hint: recall the it is possible to bound the square of the distance to the primal solution by the dual objective values]

3 Solving the Lasso problem [25%]

Consider the following problem

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

where $A \in \mathbb{R}^{n\times d}$. Let us denote by $a^i \in \mathbb{R}^d$ and $a_j \in \mathbb{R}^n$ the i-th row and j-column of A respectively. Then the objective function can be written in the form

$$\frac{1}{2n} \sum_{i=1}^{n} (\langle a^i, x\rangle - y_i)^2 + \lambda \|x\|_1 , \tag{2}$$

Then the *proximal stochastic gradient algorithm* is

$$x^{k+1} = \mathrm{prox}_{\gamma_k \lambda \|\cdot\|_1}\big(x^k - \gamma_k(\langle a^{i_k}, x\rangle - y_{i_k})a^{i_k}\big), \tag{PSGA}$$

where $\gamma_k = n/(\|A\|^2 \sqrt{k+1})$ and $(i_k)_{k\in\mathbb{N}}$ is a sequence of i.i.d. random variables uniformly distributed on $\{1, \dots, n\}$.

Alternatively, problem (1) can be approached via a *randomized coordinate proximal gradient algorithm*. Indeed, recalling that $\nabla(1/2)\|Ax - y\|^2 = A^*(Ax - y) = (\langle a_j, Ax - y\rangle)_{1\le j\le d}$ one can consider the following algorithm

$$x_j^{k+1} = \begin{cases} \mathrm{soft}_{\gamma_j\lambda}\big(x_j^k - \frac{\gamma_j}{n}\langle a_j, Ax^k - y\rangle\big) & \text{if } j = j_k \\ x_j^k & \text{otherwise,} \end{cases} \tag{RCPGA}$$

where $(j_k)_{k\in\mathbb{N}}$ is a sequence of i.i.d. random variables uniformly distributed on $\{1, \dots, d\}$ and $\gamma_j = n/\|a_j\|^2$.

Generate the data according to the following python code with $n = 1000$, $d = 500$, $s = 50$, $\sigma = 0.06$.

```
def generate_problem(n, d, s, std=0.06):
    # Generate xs
```

```
# vectors with entries in [0.5, 1] and [-1, -0.5]
# repectively
assert s % 2 == 0, "s needs to be divisible by 2"
xsp = 0.5 * (np.random.rand(s // 2) + 1)
xsn = - 0.5 * (np.random.rand(s // 2) + 1)
xsparse = np.hstack([xsp, xsn, np.zeros(d - s)])
random.shuffle(xsparse)

# Generate A
A = np.random.randn(n, d)

# Generate eps
y = A @ xsparse + std * np.random.randn(n)

return xsparse, A, y
```

1. Implement algorithm (PSGA), recalling that $\operatorname{prox}_{\gamma_k \lambda \|\cdot\|_1}$ acts component-wise as a soft-thresholding operator with threshold $\gamma_k \lambda$.
2. Implement algorithm (RCPGA).
3. Choose an appropriate regularization parameter λ and plot the objective function values vs the number of iterations for both the algorithms described above. For algorithm (PSGA) consider also the behavior of the objective values over the sequence of ergodic means, that is, $\bar{x}^k = (\sum_{i=0}^k \gamma_i)^{-1} \sum_{i=0}^k \gamma_i x_i$.

4 Support Vector Machines [25%]

The problem of SVM's for classification is

$$\min_{w \in H} \frac{1}{n} \sum_{i=1}^{n} \left(1 - y_i \langle w, \Lambda(x_i) \rangle\right)_+ + \frac{\lambda}{2} \|w\|^2, \tag{3}$$

where $(x_i, y_i)_{1 \le i \le n}$ is the training set, $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ and $\Lambda \colon \mathbb{R}^d \to H$ is the feature map corresponding to the Gaussian kernel, that is,

$$K(x, x') = \langle \Lambda(x), \Lambda(x') \rangle = \exp\left(- \frac{\|x - x'\|^2}{2\sigma^2} \right). \tag{4}$$

The dual problem is

$$\min_{u \in \mathbb{R}^n} \frac{1}{2} \langle \mathsf{K} u, u \rangle - \langle y, u \rangle + \sum_{i=1}^{n} \iota_{[0, \frac{1}{\lambda n}]}(y_i u_i), \tag{5}$$

where $\mathsf{K}_{i,j} = K(x_i, x_j)$ and $\iota_{[0, \frac{1}{\lambda n}]}$ is the indicator function of the interval $[0, \frac{1}{\lambda n}]$. The problem can be equivalently restated as

$$\min_{\alpha \in \mathbb{R}^n} \frac{1}{2} \langle \mathsf{K}_y \alpha, \alpha \rangle - \langle \mathbf{1}_n, \alpha \rangle + \sum_{i=1}^{n} \iota_{[0, \frac{1}{\lambda n}]}(\alpha_i), \tag{6}$$

where $(\mathsf{K}_y)_{i,j} = y_i \mathsf{K}_{i,j} y_j$. Note that the primal solution can be recovered via $w = \sum_{i=1}^n y_i \alpha_i \Lambda(x_i)$ and hence the decision function is

$$h_w(x) = \operatorname{sign}(\langle w, \Lambda(x) \rangle) = \operatorname{sign}\left(\sum_{i=1}^n y_i \alpha_i K(x_i, x) \right). \tag{7}$$

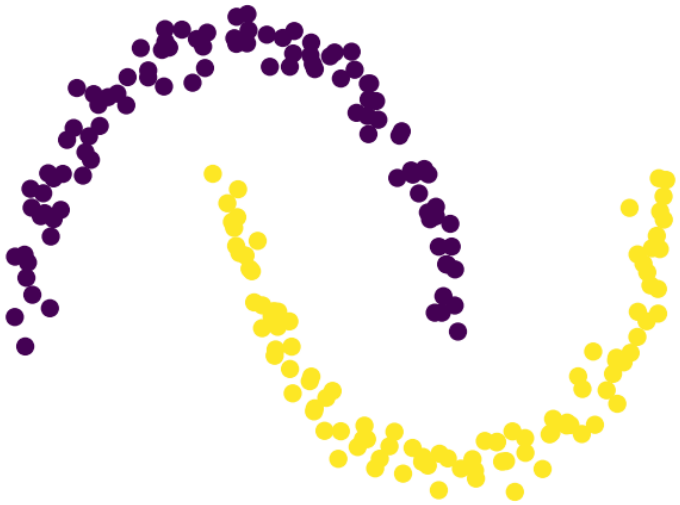

Figure 2: A realization of the two-moons dataset.

Generate the data (2D) according to the following python code:

```
from sklearn.datasets import make_moons

X, y = make_moons(n_samples=200, noise=0.05, random_state=0)
y = 2*y - 1
```

Choose appropriate values of λ and σ and address the following points.

1. Implement FISTA for solving the dual problem and plot the dual objective function.
2. Implement the randomized coordinate projected gradient algorithm on the dual problem (6) and plot the dual objective function.
3. For each algorithm above, using a contour plot command, plot the decision boundary as well as the two classes.
4. compare the performance of the two approaches in terms of speed and accuracy.