

Name:

EPFL
EE-556: Mathematics of Data
Mock Exam

Instructions:

- You are not allowed to use calculators, computers, phones, the internet, or any other computing device.
- You are not allowed to use your textbook or course notes.
- You are allowed to use handwritten notes (not typed, printed, or photocopied) on **BOTH** sides of **ONE** sheet of A4 paper.
- Please show and **EXPLAIN** all of your work. For example, it is important to write down any formula you are using so that we can give you partial credit if you make a small mistake along the way.
- Problems marked with * are difficult; attempt them last.
- In some questions, the answer to one part may depend on the answers to previous parts. You can get full credit for the latter part even if the answer to the first part is wrong. In these cases, it is especially important to show your working for all parts.
- If you have any questions about the English vocabulary, please do not hesitate to ask us.
- Once you begin the exam, please write your name on all pages.
- If you run out of space, you may write on the extra blank pages at the end of the exam sheets.

Problem	Possible Points	Score
1	20	
2	20	
3	20	
4	20	
5	20	
Total:	100	

Name:

PROBLEM 1: DIFFUSION MODELS

Let $(x_1, \dots, x_n)$ be a dataset of n images. Let X_0 be a random clean image sampled uniformly from $x_1, \dots, x_n$. A random noisy image at time t ($t > 0$), denoted by X_t , is defined as

$$X_t = e^{-t}X_0 + \sqrt{1 - e^{-2t}}Z,$$

where Z is a standard Gaussian variable. Clearly, given this setup, the higher the time t , the noisier the image X_t will be.

Diffusion models learn how to obtain X_0 from observing X_t . In other words, a denoising network net can be obtained by optimizing the following loss, given X_0 and X_t from some unknown distribution $\mathbb{P}$:

$$\min_{\text{net}} \mathcal{L}(\text{net}) := \mathbb{E}_{(X_0, X_t) \sim \mathbb{P}} [\| \text{net}(X_t) - X_0 \|_2^2].$$

As a result, the optimized denoising network learns to approximate the conditional expectation

$$\text{net}(X_t) \approx \mathbb{E}[X_0|X_t].$$

(a) (5 points) Let us assume that $X_0 \sim p(x_0)$ is sampled from the uniform distribution over $(x^1, \dots, x^n)$. Show that the distribution of X_t is $p(x_t) = \frac{1}{n} \sum_{i=1}^n p(x_t|x_i)$.

By definition, X_0 is sampled uniformly from the dataset $\{x_1, \dots, x_n\}$, so

$$p(X_0 = x_i) = \frac{1}{n}, \quad i = 1, \dots, n.$$

The distribution of X_t is obtained by marginalizing over all possible values of X_0 :

$$p(x_t) = \sum_{i=1}^n p(x_t|x_0 = x_i)p(x_0 = x_i).$$

Since $p(x_0 = x_i) = \frac{1}{n}$, this becomes:

$$p(x_t) = \sum_{i=1}^n \frac{1}{n} p(x_t|x_0 = x_i) = \frac{1}{n} \sum_{i=1}^n p(x_t|x_i).$$

Name:

(b) (5 points) Compute the gradient of $\log p(x_t)$ and show that

$$\nabla \log p(x_t) = \sum_{i=1}^n \left(\frac{e^{-t} x_i - x_t}{1 - e^{-2t}} \right) p(x_i | x_t).$$

Hint: From the definition of $p(x_t | x_i)$, X_t is a Gaussian with mean $e^{-t} x_i$ and variance $1 - e^{-2t}$.

We know:

$$\log p(x_t) = \log \left(\frac{1}{n} \sum_{i=1}^n p(x_t | x_i) \right).$$

Taking the gradient:

$$\nabla \log p(x_t) = \frac{\nabla p(x_t)}{p(x_t)}.$$

Now, $p(x_t) = \frac{1}{n} \sum_{i=1}^n p(x_t | x_i)$, so:

$$\nabla p(x_t) = \frac{1}{n} \sum_{i=1}^n \nabla p(x_t | x_i).$$

From the definition of $p(x_t | x_i)$, X_t is a Gaussian with mean $e^{-t} x_i$ and variance $1 - e^{-2t}$, i.e.,

$$p(x_t | x_i) = \frac{1}{\sqrt{2\pi(1 - e^{-2t})}} \exp \left(-\frac{(x_t - e^{-t} x_i)^2}{2(1 - e^{-2t})} \right).$$

Taking the gradient of $p(x_t | x_i)$ w.r.t. x_t :

$$\nabla p(x_t | x_i) = p(x_t | x_i) \cdot \frac{x_i e^{-t} - x_t}{1 - e^{-2t}}.$$

Thus:

$$\nabla p(x_t) = \frac{1}{n} \sum_{i=1}^n p(x_t | x_i) \cdot \frac{e^{-t} x_i - x_t}{1 - e^{-2t}}.$$

Finally, dividing by $p(x_t) = \frac{1}{n} \sum_{i=1}^n p(x_t | x_i)$, we get:

$$\nabla \log p(x_t) = \sum_{i=1}^n \frac{p(x_t | x_i)}{p(x_t)} \cdot \frac{e^{-t} x_i - x_t}{1 - e^{-2t}} = \sum_{i=1}^n \frac{e^{-t} x_i - x_t}{1 - e^{-2t}} p(x_i | x_t),$$

where $p(x_i | x_t) = \frac{p(x_t | x_i)}{p(x_t)}$.

Name:

(c) (5 points) One can show that the Hessian is given by

$$\nabla^2 \log p(x_t) = \frac{e^{-2t}}{(1 - e^{-2t})^2} \text{Cov}[X_0|X_t = x_t] - \frac{1}{1 - e^{-2t}} I,$$

where $\text{Cov}[X_0|X_t]$ is the covariance matrix of $X_0|X_t = x_t$, and I is the identity matrix. Recalling that a covariance matrix is always positive semidefinite (psd), what can you say about the Hessian of $\log p_t$ as $t \rightarrow \infty$? Does it become psd, negative definite, or non-definite?

$$\nabla^2 \log p(x_t) = \frac{e^{-2t}}{1 - e^{-2t}} \text{Cov}[X_0|X_t = x_t] - \frac{1}{1 - e^{-2t}} I.$$

As $t \rightarrow \infty$, we observe:

- $e^{-2t} \rightarrow 0$, so the first term involving the covariance matrix vanishes.
- The Hessian becomes:

$$\nabla^2 \log p(x_t) \approx -\frac{1}{1 - e^{-2t}} I \rightarrow -I.$$

Thus, the Hessian becomes *negative definite* as $t \rightarrow \infty$.

Name:

(d) (2 points) Deduce from your previous answer if $\log p(x_t)$ becomes more convex, more concave or neither as a function of x as $t \rightarrow \infty$? If you could not solve part (c), assume its solution (psd, negative definite, non-definite) and provide an answer here.

From the Hessian:

$$\nabla^2 \log p(x_t) \approx -I \quad (\text{negative definite as } t \rightarrow \infty).$$

This implies that $\log p(x_t)$ becomes *more concave* as $t \rightarrow \infty$.

(e) (3 points) The conditional expectation $\mathbb{E}[X|X_t]$ can be thought of an average of all the possible candidates X_i that could have generated X . If the noise is large then the conditional expectation is an average of many candidates from $1, \dots, n$. The next image illustrates this idea. Recalling that given a denoiser, at test time, the basic recipe for generating a new sample is the following:

1. Generate a noisy sample $X_t = e^{-t}X + \sqrt{1 - e^{-2t}}Z$
2. Denoise to obtain X

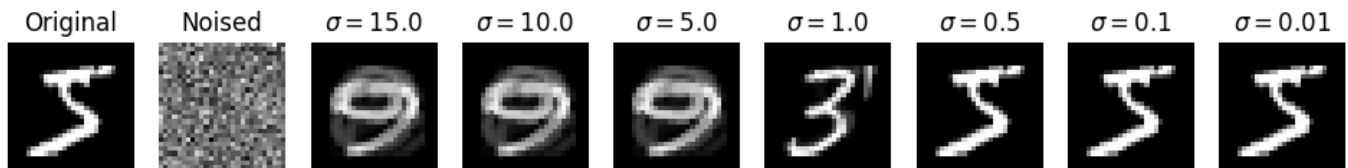


Figure 1: Denoising MNIST from various noise levels, here $\sigma = \sqrt{1 - e^{-2t}}$

With this in mind, choose the correct sentences and justify your choice:

- ☐ If the noise $t \rightarrow \infty$, the distribution of X_t is very close to being a Gaussian
- ☐ If the noise $t \rightarrow \infty$, step 1 of creating a noisy sample becomes tractable but denoising from this high noise level using a conditional expectation becomes inaccurate.
- ☐ If the noise $t \rightarrow \infty$, the distribution of X_t has exactly two modes.

The first two are correct.

Step 1 becomes easier because of the concavity we showed w.r.t. the distribution of X_t (slide 33 of lecture 12). Concavity is not about optimization but more about number of modes (which makes it easy or hard). Remember, we do not have X we want to train the model with a high noise so that the term $\sqrt{1 - e^{-2t}}Z$ dominates that term in order to be easier to generate X_t at test time. Nonetheless, if the noise is very large, a lot of candidates (specifically n in this case) could have generated that noise when we do the conditional expectation $\mathbb{E}[X_0|X_t]$ (slide 34 of lecture 12). If all my samples $1, \dots, n$ could have generated X_t , when I condition on X_t , then I get an average of them all, given that condition on X_t I could create all of those samples.

Name:

PROBLEM 2: SHARPNESS-AWARE MINIMIZATION (SAM)

A machine learning team is designing a binary classifier using a linear function $h_{\mathbf{x}}$ with parameter $\mathbf{x} \in \mathbb{R}^d$. They train a classifier using empirical risk minimization on a labeled dataset containing only one sample $\{(\mathbf{a}^{(1)}, b^{(1)})\}$ where $\mathbf{a} \in \mathbb{R}^d$ and $b \in \{\pm 1\}$. Their SAM framework incorporates an ℓ_2 -norm constraint on the parameter perturbations. As a result,

- The SAM objective involves finding an adversarial perturbation δ that maximizes the loss, subject to the constraint $\|\delta\|_2 \leq \epsilon$.
 - The team's goal is to determine the parameter $\mathbf{x}$ that minimizes the sharpness-aware loss over the dataset.
- (a) (2 points) For a given loss function of the form $L(\mathbf{x}, \mathbf{a}, b)$, Fill in the blanks below to write the corresponding SAM optimization problem.

$$\min_{\mathbf{x}} \max_{\|\delta\|_2 \leq \epsilon} L(\mathbf{x} + \delta, \mathbf{a}, b).$$

Solution

$$\min_{\mathbf{x}} \max_{\|\delta\|_2 \leq \epsilon} L(\mathbf{x} + \delta, \mathbf{a}, b).$$

- (b) (3 points) Recall that we assume a linear function for the classification problem above

$$h_{\mathbf{x}}(\mathbf{a}) = \langle \mathbf{x}, \mathbf{a} \rangle,$$

and now we assume a loss function as the **logistic loss**, defined as:

$$L(\mathbf{x}, \mathbf{a}, b) = \log(1 + \exp(-b \langle \mathbf{x}, \mathbf{a} \rangle)).$$

Using the inner maximization problem from part (a), reformulate the SAM problem considering the logistic loss. Given this loss function, is the minimization problem in the minmax problem above convex?

Solution

$$\min_{\mathbf{x}} \max_{\|\delta\|_2 \leq \epsilon} L(\mathbf{x} + \delta, \mathbf{a}, b) = \min_{\mathbf{x}} \max_{\|\delta\|_2 \leq \epsilon} \log(1 + \exp(-b \langle \mathbf{x} + \delta, \mathbf{a} \rangle)).$$

- The logistic loss function is convex with respect to $\mathbf{x}$.
- The constraint set $\{\delta : \|\delta\|_2 \leq \epsilon\}$ preserves convexity.

Name:

- (c) (4 points) Suppose that the data given is $\mathbf{a}^{(1)} = (1, -1)$, $b^{(1)} = 1$, the initial iterate is $\mathbf{x}_1 = (1, 0)$, and $\epsilon = 0.5$. Find the set of possible perturbations that can maximize the inner SAM objective, i.e., solve the maximization problem corresponding to the perturbation. Show that the solution is unique.

Solution

We are solving:

$$\delta^* = \arg \max_{\|\delta\|_2 \leq \epsilon} L(\mathbf{x} + \delta, \mathbf{a}, b) = \arg \max_{\|\delta\|_2 \leq \epsilon} \log(1 + \exp(-b\langle \mathbf{x} + \delta, \mathbf{a} \rangle)).$$

Substituting $b = 1$:

$$L(\mathbf{x} + \delta, \mathbf{a}, b) = \log(1 + \exp(-\langle \mathbf{x}, \mathbf{a} \rangle - \langle \delta, \mathbf{a} \rangle)).$$

The key term driving the maximization is $-\langle \delta, \mathbf{a} \rangle$, because the rest of the loss depends on $\langle \mathbf{x}, \mathbf{a} \rangle$, which is fixed with respect to δ . The function $\log(1 + \exp(z))$ is increasing with respect to z . So the optimization becomes:

$$\delta^* = \arg \max_{\|\delta\|_2 \leq \epsilon} -\langle \delta, \mathbf{a} \rangle.$$

It is equalvalant to:

$$\delta^* = \arg \min_{\|\delta\|_2 \leq \epsilon} \langle \delta, \mathbf{a} \rangle.$$

1. To minimize $\langle \delta, \mathbf{a} \rangle$, δ must align in the direction opposite to $\mathbf{a}$, because minimizing a dot product involves moving in the opposite direction.
2. The solution lies on the boundary of the L2 ball with radius $\epsilon = 0.5$. Normalize $\mathbf{a} = (1, -1)$:

$$\frac{\mathbf{a}}{\|\mathbf{a}\|_2} = \frac{(1, -1)}{\sqrt{1^2 + (-1)^2}} = \frac{(1, -1)}{\sqrt{2}}.$$

3. Scale by $-\epsilon$ to move in the opposite direction:

$$\delta^* = -0.5 \cdot \frac{(1, -1)}{\sqrt{2}} = \left(-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}} \right).$$

The solution is **unique**.

Name:

Consider the same setting as in part (c), where we have

$$\mathbf{a}^{(1)} = (1, -1), \quad b^{(1)} = 1, \quad \mathbf{x}_1 = (1, 0), \quad \epsilon = 0.5.$$

- (d) (2 points) Based on the formulation and results from part (b), does the convex variant of Danskin's Theorem applies to this problem? If yes, check the conditions for the convex variant. If not, check the conditions of the general variant. (See the end of the question for their definitions.)

Does the convex variant of Danskin's Theorem apply? ☐ Yes ☐ No

Check the conditions:

Solution

Convex variant. Checking Conditions

- According to (b), the problem is convex. Thus, the convex variant of Danskin's Theorem applies.
- According to (c), the solution is unique.

- (e) (4 points) Using the appropriate Danskin's theorem, is the SAM's objective function differentiable at the point $\mathbf{x} = \mathbf{x}_1$? If the function is differentiable, compute the gradient at $\mathbf{x} = \mathbf{x}_1$. If the function is not differentiable, find a subgradient and provide a justification for your answer.

Is the function differentiable? ☐ Yes ☐ No

gradient or subgradient =

Solution

Since Danskin's Theorem applies, $f(x)$ is differentiable at $x = (1, 0)$ if and only if the maximizer δ^* is unique, which is clarified in (c).

The gradient of the loss with respect to x is:

$$\nabla_x L(x + \delta^*, a, b) = -\sigma(-\langle x + \delta^*, a \rangle)ba,$$

where $\sigma(z) = \frac{1}{1 + \exp(-z)}$.

Name:

3. Substitute $a(1) = (1, -1)$, $b(1) = 1$, $x(1) = (1, 0)$, $\delta^* = \left(-\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}\right)$, and the gradient is:

$$-\frac{1}{1 + \exp(1 - 0.5\sqrt{2})} \cdot (1, -1).$$

Name:

- (f) (5 points) Write a single iteration of the stochastic SAM training update in pseudo-code. Then, using the values obtained in part (c) and the computed gradient from part (e), calculate the updated weights $\mathbf{x}_2$ for the adversarial training objective. Assume a learning rate of $\eta = 1$. Show your work.

- Model parameters: $\mathbf{x}$
- Learning rate: η
- Input data: $(\mathbf{a}, b)$
- Loss function: $L(\mathbf{x}, \mathbf{a}, b)$
- Perturbation size: ϵ

Input: Initial weights $\mathbf{x}(1)$, data point $\mathbf{a}(1)$, label $b(1)$,
perturbation budget ϵ , learning rate η ,
loss function L , gradient function $\text{grad}_{\mathbf{x}}$.
Output: Updated weights $\mathbf{x}(2)$.

1. Initialize $\mathbf{x}(1)$ and define loss function $L(\mathbf{x}, \mathbf{a}, b)$.
2. Compute perturbation δ : [Answer 1](#)
3. Compute gradient: $\text{grad} = \text{grad}_{\mathbf{x}} L(\text{Answer 2})$
4. Update weights using gradient descent: $\mathbf{x}(2) = \text{Answer 3}$

Solution

Step 1: Pseudo-code for One Loop of Stochastic SAM Training.

Input: Initial weights $\mathbf{x}(1)$, data point $\mathbf{a}(1)$, label $b(1)$,
perturbation budget ϵ , learning rate η ,
loss function L , gradient function $\text{grad}_{\mathbf{x}}$.
Output: Updated weights $\mathbf{x}(2)$.

1. Initialize $\mathbf{x}(1)$ and define loss function $L(\mathbf{x}(\mathbf{a}), b)$.
2. Find adversarial perturbation δ^* :
 $\delta^* = \underset{\delta}{\text{argmax}} \|\delta\|_2 \leq \epsilon L(\mathbf{x} + \delta, \mathbf{a}, b)$.
3. Compute gradient of L with respect to $\mathbf{x}$:
 $\text{grad} = \text{grad}_{\mathbf{x}} L(\mathbf{x} + \delta, \mathbf{a}, b)$.
4. Update weights using gradient descent:
 $\mathbf{x}(2) = \mathbf{x}(1) - \eta * \text{grad}$.

Step 2: Using Values from (e). From part (e), the gradient of the perturbation objective at $\mathbf{x}(1) = (1, 1)$ is:

$$\mathbf{g} = -\frac{1}{1 + \exp(1 - 0.5\sqrt{2})} \cdot (1, -1)$$

Step 3: Updating the Weights. Using the gradient descent update rule:

$$\mathbf{x}(2) = \mathbf{x}(1) - \eta \cdot \mathbf{g} = \left(1 + \frac{1}{1 + \exp(1 - 0.5\sqrt{2})}, -\frac{1}{1 + \exp(1 - 0.5\sqrt{2})} \right)$$

Name:

Danskin's Theorem. (Convex variant) Let $\Phi(\mathbf{x}, \mathbf{y}) : \mathbb{R}^p \times \mathcal{Y} \rightarrow \mathbb{R}$ be a continuous function, where $\mathcal{Y} \subset \mathbb{R}^m$ is a compact set and define $f(\mathbf{x}) := \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$. Suppose that for each $\mathbf{y}$ in the compact set $\mathcal{Y}$, $\Phi(\mathbf{x}, \mathbf{y})$ (as a function of $\mathbf{x}$) is an extended real-valued closed proper convex function.

Define $\mathcal{Y}^*(\mathbf{x}) := \arg \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$ as the set of maximizers (for a fixed value of $\mathbf{x}$ and $\mathbf{y}_x^* \in \mathcal{Y}^*$ as an element of this set. We have

1. $f(\mathbf{x})$ is a convex function.
2. If $\mathbf{y}_x^* = \arg \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$ is unique, then the function $f(\mathbf{x}) = \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$ is differentiable at $\mathbf{x}$:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \nabla_{\mathbf{x}} \left(\max_{\mathbf{y} \in \mathcal{Y}} \phi(\mathbf{x}, \mathbf{y}) \right) = \nabla_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y}_x^*).$$

3. If $\mathbf{y}_x^* = \arg \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$ is not unique, then the subdifferential $\partial_{\mathbf{x}} f(\mathbf{x})$ of f is given by

$$\partial_{\mathbf{x}} f(\mathbf{x}) = \text{conv} \left\{ \partial_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y}_x^*) : \mathbf{y}_x^* \in \mathcal{Y}^*(\mathbf{x}) \right\}.$$

Danskin's Theorem. (General) Let $\Phi(\mathbf{x}, \mathbf{y}) : \mathbb{R}^p \times \mathcal{Y} \rightarrow \mathbb{R}$ be a continuous function, where $\mathcal{Y} \subset \mathbb{R}^m$ is a compact set and define $f(\mathbf{x}) := \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$ and let $\mathcal{Y}^*(\mathbf{x}) = \arg \max_{\mathbf{y} \in \mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y})$.

Suppose that for each $\mathbf{y}$ in the compact set $\mathcal{Y}$, $\Phi(\mathbf{x}, \mathbf{y})$ (as a function of $\mathbf{x}$) is differentiable and that $\nabla_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y})$ exists and is continuous. Then f is continuous, directionally differentiable and its directional derivatives satisfy

$$f'(\mathbf{x}, \mathbf{u}) = \sup_{\mathbf{y}_x^* \in \mathcal{Y}^*(\mathbf{x})} \mathbf{u}^T \nabla_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y}_x^*)$$

In particular, if for some $\mathbf{x} \in \mathbb{R}^p$ the set $\mathcal{Y}^*(\mathbf{x}) = \mathbf{y}_x^*$ is a singleton (contains only one element) then f is differentiable and we have

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \nabla_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y}_x^*)$$

Corollary to Danskin's Theorem Suppose the conditions of Danskin's Theorem (General) hold.

If $\mathbf{y}_x^* \in \mathcal{Y}^*(\mathbf{x})$, then as long as $-\nabla_{\mathbf{x}} \Phi(\mathbf{x}, \mathbf{y}_x^*)$ is nonzero then it is a descent direction for $f(\mathbf{x})$.

Name:

PROBLEM 3: CONVERGENCE RATE AND PER-ITERATION COMPLEXITY TRADEOFFS (20 points)

Fourier and András are trying to solve some optimization problems and are in disagreement on who is correct. In problems (a)–(d) below,

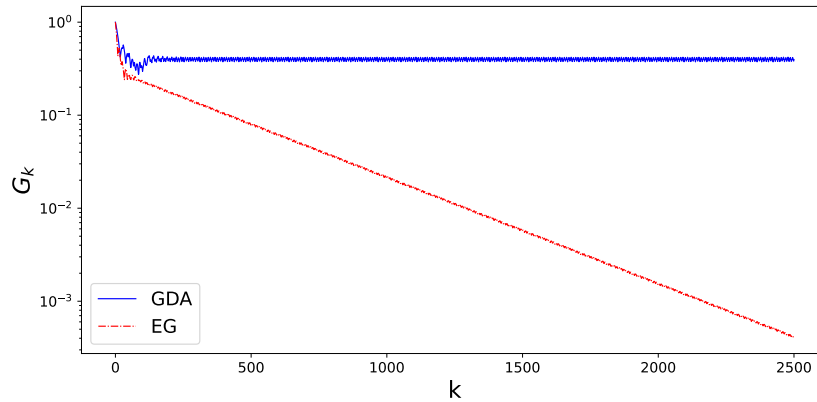
If the function f is smooth, we assume that the Lipschitz constant L of $\nabla f(\mathbf{x})$ is known.

If the function f is strongly convex, we assume that the strong convexity parameter μ is known.

(a) (5 points) András was comparing two optimization algorithms w.r.t. how they solve the following min-max optimization problem:

$$\min_{\mathbf{x} \in \Delta_5} \max_{\mathbf{y} \in \Delta_5} f(\mathbf{x}, \mathbf{y}) := \mathbf{x}^\top \mathbf{I} \mathbf{y},$$

where Δ_5 is the simplex in 5 dimensions and $\mathbf{I}$ is the 5×5 identity matrix. He obtained the linear-linear scale convergence plots below after running Gradient Descent Ascent (GDA) and ExtraGradient (EG). On the y-axis in the plots, the duality gap G over time is shown, $G_k := G(\mathbf{x}_k, \mathbf{y}_k) = \max_{\mathbf{y} \in \Delta_5} f(\mathbf{x}_k, \mathbf{y}) - \min_{\mathbf{x} \in \Delta_5} f(\mathbf{x}, \mathbf{y}_k)$.



Fourier says that there must be a bug in the code (i.e., the results are not reasonable, so there is something wrong). Do you agree or disagree with Fourier? Explain your answer. (No credits will be given unless you provide a clear explanation, and we will accept any reasonable answers.)

Solution

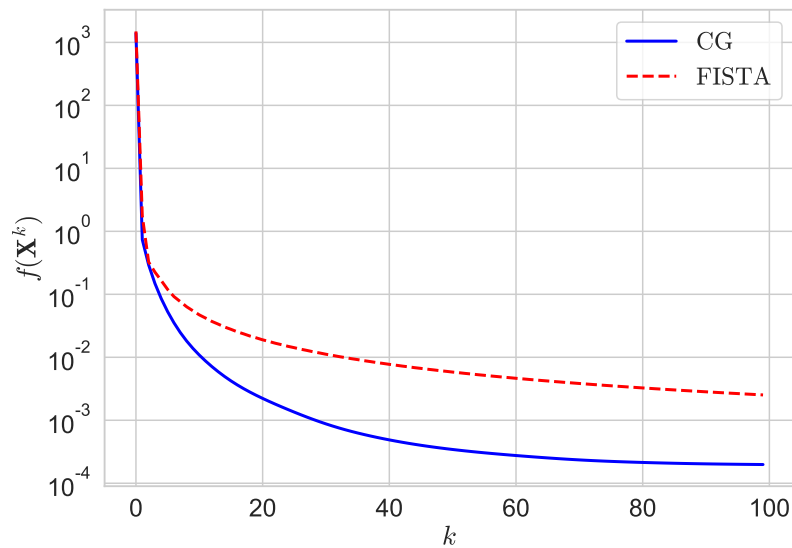
Fourier is wrong, there is no bug here.

Name:

(b) (5 points) András was comparing two algorithms on how they solve a semidefinite programming (SDP) problem, namely:

$$\min_{\mathbf{X}, \|\mathbf{X}\|_F \leq 1, \mathbf{X} \succeq 0} f(\mathbf{X}) := \|\mathbf{A}(\mathbf{X}) - \mathbf{b}\|_F^2,$$

where $\mathbf{A}$ is a smooth and convex operator. He obtained the linear-log scale convergence plots of the **objective values** below after running Fast Proximal Gradient Descent (FISTA) and Frank-Wolfe / Conditional Gradient (CG).



Fourier says that there must be a bug in the code (i.e., the results are not reasonable, so there is something wrong). Do you agree or disagree with Fourier? Explain your answer. (No credits will be given unless you provide a clear explanation, and we will accept any reasonable answers.)

Solution

Fourier is right. From the perspective of the number of iterations, for this problem FISTA should not converge with a slower rate than CG.

Name:

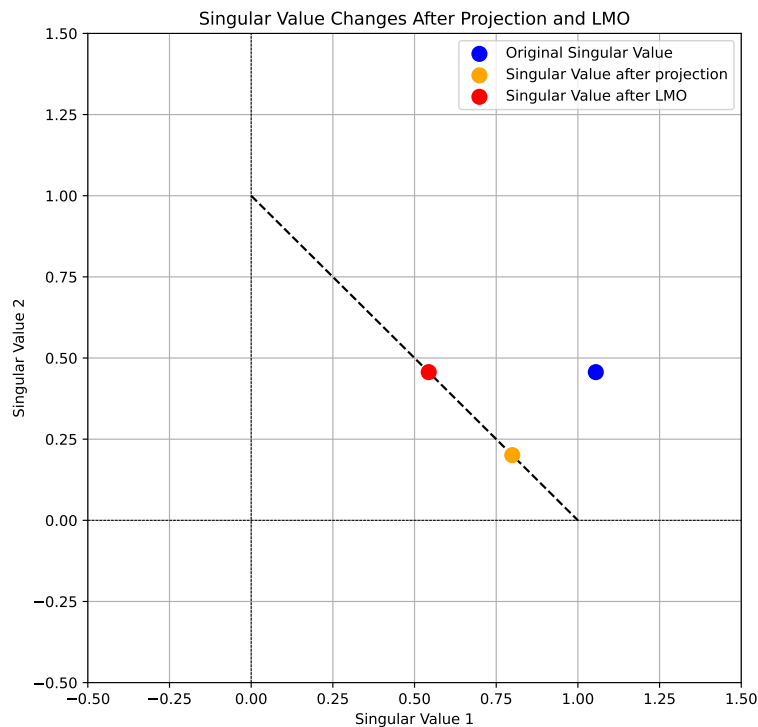
(c) (5 points) András and Fourier were comparing the output of two oracles for a nuclear-norm-constrained optimization problem, specifically:

$$\min_{\mathbf{X}, \|\mathbf{X}\|_* \leq 1} f(\mathbf{X}) := (\langle \mathbf{A}, \mathbf{X} \rangle - b)^2,$$

where $\mathbf{A}$ is dimensionality-reducing. They implemented the proximal / projection operator and the Linear Minimization Oracle (LMO) for the nuclear norm ball to test their results.

Hint: The definitions of these operators are provided on the next page.

For a simple 2×2 example matrix, they visualized the singular values before and after applying each oracle in the x-axis and y-axis of a scatter plot, shown below:



Fourier says that there must be a bug in the code. Do you agree or disagree with Fourier? Explain your answer. (No credit will be awarded without a clear and reasonable explanation. Both correct and incorrect answers are accepted as long as they are well justified.) You can check the hints in the next page.

Solution

Fourier is right, because the LMO output must have 1 and 0 as eigenvalues, while the projected eigenvalues are also incorrect.

Name:

Recall: The definitions of the nuclear norm ball projection and LMO operators are as follows:

1. Projection: The projection of a matrix $\mathbf{X}$ onto the nuclear norm ball $\{\mathbf{X} \mid \|\mathbf{X}\|_* \leq 1\}$ is performed by truncating its singular values to ensure the nuclear norm constraint holds. If the singular value decomposition (SVD) of $\mathbf{X}$ is $\mathbf{X} = \mathbf{U} \text{diag}(\sigma) \mathbf{V}^\top$, then the projection is computed as:

$$\text{proj}(\mathbf{X}) = \mathbf{U} \text{diag}(\sigma') \mathbf{V}^\top,$$

where σ' is the vector of singular values thresholded such that $\sum \sigma' \leq 1$ and $\sigma' \geq 0$.

2. Linear Minimization Oracle (LMO): The LMO for the nuclear norm ball solves the linear problem:

$$\min_{\|\mathbf{Y}\|_* \leq 1} \langle \mathbf{C}, \mathbf{Y} \rangle,$$

where $\mathbf{C}$ is any matrix. The solution is given by $\mathbf{Y} = -\mathbf{u}\mathbf{v}^\top$, where $\mathbf{u}$ and $\mathbf{v}$ are the left and right singular vectors corresponding to the largest singular value of $\mathbf{C}$.

Name:

(d)* (5 points) Neural Tangent Kernel: András wants to derive the NTK matrix for a neural network with squared loss:

$$L(\mathbf{w}(t)) = \frac{1}{2} \sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i)^2,$$

under gradient flow:

$$\frac{d\mathbf{w}(t)}{dt} = -\nabla_{\mathbf{w}} L(\mathbf{w}(t)).$$

András claims that the NTK matrix $\mathbf{H}(t) \in \mathbb{R}^{n \times n}$ can be derived as follows. For the time evolution of the output $\mathbf{f}(t) = (f(\mathbf{x}_i; \mathbf{w}(t)))_{i=1}^n$, he writes:

1. Differentiating $\mathbf{f}(t)$ with respect to time:

$$\frac{d\mathbf{f}(t)}{dt} = \frac{\partial \mathbf{f}(t)}{\partial \mathbf{w}(t)} \cdot \frac{d\mathbf{w}(t)}{dt}.$$

2. The gradient of the loss with respect to $\mathbf{w}(t)$ is:

$$\nabla_{\mathbf{w}} L(\mathbf{w}(t)) = \sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i) \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}}.$$

3. Substituting the gradient flow equation:

$$\frac{d\mathbf{w}(t)}{dt} = - \sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i) \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}}.$$

4. Substituting this into $\frac{d\mathbf{f}(t)}{dt}$, András writes:

$$\frac{d\mathbf{f}(t)}{dt} = - \frac{\partial \mathbf{f}(t)}{\partial \mathbf{w}(t)} \sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i) \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)}.$$

5. Using the definition of the NTK matrix, he concludes:

$$\frac{d\mathbf{f}(t)}{dt} = -\mathbf{H}(t) [\mathbf{f}(t) - \mathbf{y}], \quad \text{where } H_{ij}(t) = \left\langle \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)}, \sum_{i=1}^n \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)} \right\rangle.$$

Fourier says that there is a mistake in the step 5. Do you agree or disagree with Fourier? Explain your answer. (No credit will be awarded without a clear and reasonable explanation. Both correct and incorrect answers are accepted as long as they are well justified.)

Hint: You can check the dimensions after each step of the derivation.

Solution

Fourier is right. Write $\sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i) \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)}$ in vector form:

$$\sum_{i=1}^n (f(\mathbf{x}_i; \mathbf{w}(t)) - y_i) \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)} = \frac{\partial \mathbf{f}(t)}{\partial \mathbf{w}(t)}^\top [\mathbf{f}(t) - \mathbf{y}].$$

So:

$$\frac{d\mathbf{f}(t)}{dt} = - \frac{\partial \mathbf{f}(t)}{\partial \mathbf{w}(t)} \frac{\partial \mathbf{f}(t)}{\partial \mathbf{w}(t)}^\top [\mathbf{f}(t) - \mathbf{y}] = -\mathbf{H}(t) [\mathbf{f}(t) - \mathbf{y}], \quad \text{where } H_{ij}(t) = \left\langle \frac{\partial f(\mathbf{x}_i; \mathbf{w}(t))}{\partial \mathbf{w}(t)}, \frac{\partial f(\mathbf{x}_j; \mathbf{w}(t))}{\partial \mathbf{w}(t)} \right\rangle.$$

Name:

PROBLEM 4: LIPSCHITZ CONSTANT AND PROXIMAL OPERATOR

In this question, we consider Reinforcement Learning from Human Feedback (RLHF) and reward modeling. We will see how the Lipschitz constant can be related to RLHF.

Particularly, we consider the Lipschitz constant with respect to the ℓ_2 -norm. Given a prompt a , associated with a preferred answer $b^{(1)}$ and a non-preferred answer $b^{(2)}$, the loss in reward modelling is given by:

$$f(a, b^{(1)}, b^{(2)}, \mathbf{x}) = -\log \sigma \left(r(a, b^{(1)}, \mathbf{x}) - r(a, b^{(2)}, \mathbf{x}) \right), \quad (1)$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function and r is the reward model with learnable parameters $\mathbf{x}$.

(a) (5 points) Assume r is an L -Lipschitz function with respect to $\mathbf{x}$, define $h(\mathbf{x}) := r(a, b^{(1)}, \mathbf{x}) - r(a, b^{(2)}, \mathbf{x})$, then show that h is $2L$ -Lipschitz continuous with respect to $\mathbf{x}$.

Hint: Let $f: Q \rightarrow \mathbb{R}$ where $Q \subseteq \mathbb{R}^p$. Then, f is L -Lipschitz continuous if there exists a constant value $L \geq 0$ such that:

$$|f(\mathbf{y}) - f(\mathbf{x})| \leq L \|\mathbf{y} - \mathbf{x}\|_2, \quad \forall \mathbf{x}, \mathbf{y} \in Q.$$

Solution

$$\begin{aligned} |h(\mathbf{x}) - h(\hat{\mathbf{x}})| &= |r(a, b^{(1)}, \mathbf{x}) - r(a, b^{(1)}, \hat{\mathbf{x}}) + r(a, b^{(2)}, \hat{\mathbf{x}}) - r(a, b^{(2)}, \mathbf{x})| \\ &\leq |r(a, b^{(1)}, \mathbf{x}) - r(a, b^{(1)}, \hat{\mathbf{x}})| + |r(a, b^{(2)}, \hat{\mathbf{x}}) - r(a, b^{(2)}, \mathbf{x})| \\ &\leq L \|\mathbf{x} - \hat{\mathbf{x}}\|_2 + L \|\mathbf{x} - \hat{\mathbf{x}}\|_2 = 2L \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \end{aligned}$$

(b) (5 points) Based on the condition of (a), calculate an informative upper bound of the Lipschitz constant of the function f (defined in Eq.(1)) with respect to $\mathbf{x}$.

Hint: The Lipschitz constant of the composition of two or more functions can be bounded by the product of the Lipschitz constants of the individual functions

Solution

First, we calculate an informative upper bound of the Lipschitz constant of the function f with respect to h . We express $f = \log(1 + e^{-h})$ and calculate the gradient:

$$\nabla_h f = \frac{-1}{1 + e^h}.$$

Since $\|\nabla_h f\|_2 \leq 1$, we can upper bound the Lipschitz constant of f with respect to h by 1. Then, the upper bound of the Lipschitz constant of f with respect to $\mathbf{x}$ can be upper bounded by $2L \times 1 = 2L$.

Name:

(c) (5 points) Suppose the answers $b^{(1)}$ and $b^{(2)}$ are real number, we define $\mathbf{b} = [b^{(1)}, b^{(2)}]^\top \in \mathbb{R}^2$. Assume that we use a trivial reward model, i.e., $r(a, b^{(i)}, \mathbf{x}) = b^{(i)}$, for $i = 1, 2$. Is f an $3/2$ Lipschitz-continuous function with respect to $\mathbf{b}$?

Solution

Yes. In this case, $f(a, b^{(1)}, b^{(2)}, \mathbf{x}) = -\log \sigma(b^{(1)} - b^{(2)})$, then

$$\nabla_{\mathbf{b}} f = \begin{bmatrix} -\sigma(b^{(2)} - b^{(1)}) \\ \sigma(b^{(2)} - b^{(1)}) \end{bmatrix}$$

Next, denote by $\tau := \sigma(b^{(2)} - b^{(1)})$, we have $\|\nabla_{\mathbf{b}} f\|_2 = \sqrt{2}\tau \leq 3/2$.

(d) (3 points) Gradient descent is an optimization method used for learning the reward model. Next, we will explore how the proximal operator can be interpreted as a gradient descent step. We have learnt that:

Definition 1. Let $f \in \mathcal{F}(\mathbb{R}^d)$, $\mathbf{x} \in \mathbb{R}^d$ and $\lambda > 0$. The proximal operator of f is defined as:

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

Next, we introduce Bregman Proximal operator which is based on the Bregman divergence:

$$\text{prox}_{\lambda f}^{D_h}(\mathbf{y}) \equiv \arg \min_{\mathbf{x} \in \mathbb{R}^d} \left\{ f(\mathbf{x}) + \frac{1}{\lambda} D_h(\mathbf{y}, \mathbf{x}) \right\}, \quad (3)$$

where the Bregman divergence is defined as follows: Let h be a continuously-differentiable and strictly convex function. The **Bregman divergence** associated with h for points $\mathbf{x}$ and $\mathbf{y}$ is:

$$D_h(\mathbf{y}, \mathbf{x}) = h(\mathbf{y}) - h(\mathbf{x}) - \langle \nabla h(\mathbf{x}), (\mathbf{y} - \mathbf{x}) \rangle$$

Show that when $h(\mathbf{x}) = \frac{1}{2} \|\mathbf{x}\|_2^2$, the Bregman Proximal operator reduces to the proximal operator as seen in the lecture.

Solution

We have:

$$D_h(\mathbf{y}, \mathbf{x}) = \frac{1}{2} \|\mathbf{y}\|_2^2 - \frac{1}{2} \|\mathbf{x}\|_2^2 - \langle \mathbf{x}, \mathbf{y} - \mathbf{x} \rangle = \frac{1}{2} \|\mathbf{y} - \mathbf{x}\|_2^2.$$

Therefore,

$$\text{prox}_{\lambda f}(\mathbf{y}) = \arg \min_{\mathbf{x} \in \mathbb{R}^p} \left\{ f(\mathbf{x}) + \frac{1}{\lambda} D_h(\mathbf{x}, \mathbf{y}) \right\}$$

Name:

(e) (2 points) Next, when $f(\mathbf{x})$ is convex and we replace $f(\mathbf{x})$ with its lower bound $f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle$ inside the proximal operator, prove that the new optimization problem results in a gradient descent step with step size λ .

Solution

$$\text{prox}_{\lambda f^{\text{Lower bound}}}(\mathbf{x}_t) = \arg \min_{\mathbf{x} \in \mathbb{R}^d} \left\{ f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle + \frac{1}{2\lambda} \|\mathbf{x} - \mathbf{x}_t\|_2^2 \right\}.$$

By setting the gradient of the right-hand side to zero, we get

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \lambda \nabla f(\mathbf{x}_t).$$

Name:

Problem 5: Constrained Convex Optimization (20 points)

Consider the following convex optimization problem

$$\min_{X \in \mathbb{R}^{p \times n}} f(X) := \frac{1}{2} \|AX - b\|_F^2 \quad \text{subject to} \quad \|X\|_{S_\infty} \leq 1,$$

where $A \in \mathbb{R}^{m \times p}$ and $b \in \mathbb{R}^{m \times n}$ are known and $X \in \mathbb{R}^{p \times n}$ and $\|\cdot\|_{S_\infty}$ is the Schatten infinity norm or equivalently the $\|\cdot\|_{2 \rightarrow 2}$ operator norm:

$$\|X\|_{2 \rightarrow 2} = \max_{u \in \mathbb{R}^n: \|u\|_2 \leq 1} \|Xu\|_2. \quad (4)$$

Recall that Frank-Wolfe's method applies to this problem as follows:

Algorithm 1 Frank-Wolfe Method for Convex Optimization

```

1: Input:  $x^0 \in \mathbb{R}^p$ 
2: for  $k = 1, 2, \dots$  do
3:    $\hat{X}^k \in \arg \min_X (\nabla f(X^k), X)$  subject to  $\|X\|_{S_\infty} \leq 1$ 
4:    $X^{k+1} = (1 - \gamma_k)X^k + \gamma_k \hat{X}^k$ , where  $\gamma_k := \frac{2}{k+1}$ 
5: end for

```

(a) (5 points) Show that f is a smooth function with respect to the S_∞ norm, in the sense that its gradient is Lipschitz continuous, with respect to this norm, i.e., $|f(X) - f(Y)| \leq L\|X - Y\|_{2 \rightarrow 2}, \forall X, Y$. Find L in this equation.

Solution

$\forall X, X' \in \mathbb{R}^{p \times n}$, we have

$$\|\nabla f(X) - \nabla f(X')\|_{S_\infty} = \|A^T A(X - X')\|_{S_\infty} \leq \|A^T A\|_{S_\infty} \|X - X'\|_{S_\infty}$$

where $\|A^T A\|_{S_\infty}$ denotes the spectral norm of $A^T A$. Hence, f is smooth with smoothness constant $\|A^T A\|_{S_\infty}$.

(b) (5 points) Let r the rank of $\nabla f(X^k)$ and $\sigma_1, \dots, \sigma_r$ and $u_1, \dots, u_r$ and $v_1, \dots, v_r$ the singular values, the left and right singular vectors corresponding to the r non zero singular values of $\nabla f(X^k)$. Show that we can choose an output of the linear minimization oracle, $\text{lmo}(\nabla f(X^k))$, as the following matrix:

$$\hat{X}^k = - \sum_{i=1}^r u_i v_i^\top$$

Hint: The dual of the S_∞ norm is the nuclear norm of the matrix.

Solution

By Hölder's inequality, we have:

$$\langle \nabla f(X^k), X \rangle \geq -\|\nabla f(X^k)\|_* \|X\|_{S_\infty} \quad \forall X \in \mathbb{R}^{p \times n}, \|X\|_{S_\infty} \leq 1.$$

Let's consider the suggested choice of $\hat{X}^k$. This choice is feasible, since $\|\hat{X}^k\|_{S_\infty} = 1$, and

$$\langle \nabla f(X^k), \hat{X}^k \rangle = -\text{Tr}(\nabla f(X^k)^\top \sum_{i=1}^r u_i v_i^\top) = -\text{Tr}\left(\sum_{i=1}^r \sigma_i v_i u_i^\top \sum_{i=1}^r u_i v_i^\top\right) = -\sum_{i=1}^r \sigma_i = -\|\nabla f(X^k)\|_*.$$

Hence this choice of $\hat{X}^k$ is optimal.

Consider the following convex optimization problem:

$$\min_{x \in \mathbb{R}^p} f(x) \quad \text{subject to} \quad Ax = b, \quad (15)$$

Name:

where $A \in \mathbb{R}^{n \times p}$ and $b \in \mathbb{R}^n$ are known.

(c) (3 points) Write down the Lagrangian function corresponding to formulation (15) with the dual variable $\lambda \in \mathbb{R}^n$. Write down an equivalent formulation of (15) using the Lagrangian function.

Solution

$$L(x, \lambda) = f(x) + \langle \lambda, Ax - b \rangle$$
$$f(x^*) = \min_{x \in \mathbb{R}^p} f(x) \quad \text{subject to } \{Ax = b\} = \min_{\lambda \in \mathbb{R}^n} \max_{x \in \mathbb{R}^p} f(x) + \langle \lambda, Ax - b \rangle$$

(d) (5 points) Write down the augmented Lagrangian (AL) of formulation (15) with quadratic penalty. Use μ as the parameter for the penalty function. Given the definition of the Lagrange dual function as:

Solution

$$L_\mu(x, \lambda) = f(x) + \langle \lambda, Ax - b \rangle + \frac{\mu}{2} \|Ax - b\|^2$$
$$d_\mu^* = L_\mu(x^*, \lambda^*) = \max_{\lambda \in \mathbb{R}^n} \min_{x \in \mathbb{R}^p} L_\mu(x, \lambda) = \max_{\lambda \in \mathbb{R}^n} \min_{x \in \mathbb{R}^p} \left\{ f(x) + \langle \lambda, Ax - b \rangle + \frac{\mu}{2} \|Ax - b\|^2 \right\}$$
$$d^* = L(x^*, \lambda^*) = \max_{\lambda \in \mathbb{R}^n} \min_{x \in \mathbb{R}^p} L(x, \lambda) = \max_{\lambda \in \mathbb{R}^n} \min_{x \in \mathbb{R}^p} \{ f(x) + \langle \lambda, Ax - b \rangle \}$$

Under the Slater's condition: $\text{relint}(\text{dom}(f)) \cap \{x : Ax = b\} \neq \emptyset$, the KKT condition is

$$\begin{cases} 0 \in \partial_x L(x^*, \lambda^*) = A^T \lambda^* + \partial f(x^*), \\ 0 = \nabla_\lambda L(x^*, \lambda^*) = Ax^* - b. \end{cases} \quad \Rightarrow \quad d^* = d_\mu^*$$

Name:

Consider the following convex optimization problem:

$$\min_{x \in \mathbb{R}^p} c^T x \quad \text{subject to} \quad Ax = b, \|x\|_1 \leq 1, \quad (16)$$

where $A \in \mathbb{R}^{n \times p}$, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}^p$ are known.

(e)* (2 points) Derive the dual function of formulation (16). Simplify the expression of $d(\lambda)$ so that it does not involve a min or max. (Hint: You may use Hölder's inequality)

Solution

$$d(\lambda) := \min_{\|x\|_1 \leq 1} L(x, \lambda)$$

The Lagrangian function $L(x, \lambda)$ is:

$$L(x, \lambda) = \langle x, c \rangle + \langle \lambda, Ax - b \rangle$$

Thus,

$$d(\lambda) = \min_{\|x\|_1 \leq 1} (\langle x, c \rangle + \langle \lambda, Ax - b \rangle)$$

We can simplify this as:

$$d(\lambda) = \min_{\|x\|_1 \leq 1} (\langle x, c \rangle + \langle A^T \lambda, x \rangle - \langle \lambda, b \rangle)$$

$$d(\lambda) = \min_{\|x\|_1 \leq 1} (\langle x, c + A^T \lambda \rangle - \langle \lambda, b \rangle)$$

By Hölder's inequality, we can bound the term $\langle x, c + A^T \lambda \rangle$ as:

$$d(\lambda) = -\|c + A^T \lambda\|_\infty - \langle \lambda, b \rangle$$

Name:

This page is extra space to write. Please make sure to refer to the problem on which you are working.

Name:

This page is extra space to write. Please make sure to refer to the problem on which you are working.

Name:

This page is extra space to write. Please make sure to refer to the problem on which you are working.

Name:

This page is extra space to write. Please make sure to refer to the problem on which you are working.

Name:

This page is extra space to write. Please make sure to refer to the problem on which you are working.