Stochastic methods for optimization and sampling

Exam 2025

The precision and conciseness of the justifications are part of the evaluation. All exercises are independent and can be addressed in the order of your choice. Within a specific exercise, some questions may depend on the answers of the previous ones. If you did not solve a given question, you may still use its result in subsequent questions. It is not necessary to have completed all the questions to get an excellent grade.

Exercise 1: MCMC

Let p and q be two probabilities on a finite state space E, with $0 < p(x) \leq cq(x)$ for some constant $c > 0$. Let $(Y_n)_{n\geq 0}$ be independent random variables with the same distribution q. We define a sequence of random variables $(X_n)_{n\geq 0}$ as follows:

- X_0 is a random variable with distribution q, independent of the sequence $(Y_n)_{n\geq 0}$;
- X_{n+1} is defined from X_n by drawing a random variable U_n uniformly on $[0,1]$, independent of the other previously defined variables, and we define

$$X_{n+1} = Y_{n+1} \quad \text{if} \quad U_n \leq \frac{p(Y_{n+1})}{cq(Y_{n+1})}$$

$$X_{n+1} = X_n \quad \text{if} \quad U_n > \frac{p(Y_{n+1})}{cq(Y_{n+1})}.$$

1. Justify that the sequence $(X_n)_{n\geq 0}$ is an irreducible Markov chain (short answer expected).
2. Show that the transition probability $P(x,y) = \mathbb{P}(X_{n+1} = y | X_n = x)$ of $(X_n)_{n\geq 0}$ for $x, y \in E$ satisfies

$$P(x,y) = \begin{cases} \dfrac{p(y)}{c} & \text{if } x \neq y \\ 1 - \dfrac{1}{c}\displaystyle\sum_{x' \neq x} p(x') & \text{otherwise.} \end{cases}$$

3. What is the difference between this sampling procedure and the rejection method?
4. Recall the Metropolis-Hastings algorithm and discuss the differences with the algorithm studied here.
5. Show that p is the stationary probability measure of the chain, that is $\sum_{y\in E} P(y,x)p(y) = p(x)$, for any $x \in E$. Is the chain reversible?
6. Deduce an estimator $\hat{p}(x)$ of $p(x)$ for any $x \in E$ and justify its almost sure convergence toward $p(x)$.

Exercise 2: Mini-batch SGD in the interpolation setting

Let $n, d \geq 1$ be integers, and consider the optimization problem

$$\min_{\theta\in\mathbb{R}^d} F(\theta), \quad \text{where} \quad F(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta), \quad \forall \theta \in \mathbb{R}^d \tag{1}$$

for convex and differentiable functions f_i. In modern machine learning, a special case of the optimization problem above has gained considerable importance in the past few years. Specifically, it corresponds to the following restriction for the functions f_i.

Assumption A (Interpolation setting): *there exists a vector θ^* that minimizes all the functions f_i simultaneously:*

$$\exists \theta^* \in \mathbb{R}^d, \ \forall i \in \{1, \dots, n\} : \quad \nabla f_i(\theta^*) = 0_{\mathbb{R}^d}.$$

It is immediate that the vector θ^* from Assumption A is also a solution to the optimization problem (1).
Notation. For any subset $v \subseteq \{1, \dots, n\}$ of cardinality m, we write for ease

$$f_v = \frac{1}{m} \sum_{i \in v} f_i.$$

In particular, if $m = n$ and $v = \{1, \dots, n\}$, then $f_v = F$. In this exercise, we will prove the theorem below. We recall the pseudo-code below

---

Algorithm 1: Mini-batch SGD

Start from $\theta_0 \in \mathbb{R}^d$.

Until termination condition, iterate

Draw a subset $v_{k+1} \subseteq \{1, \dots, n\}$ of cardinality m u.a.r., independent of the past

$$\theta_{k+1} = \theta_k - \gamma \nabla f_{v_{k+1}}(\theta_k)$$
$$\big(= \theta_k - \frac{\gamma}{m} \sum_{i \in v_{k+1}} \nabla f_i(\theta_k) \big).$$

The exercise aims to study the performance of the mini-batch SGD algorithm in the *interpolation setting.*

Theorem 1. *Let $\beta, \lambda, \mu > 0$ and assume that*

1. *F is μ-strongly convex*
2. *∇F is λ-Lipschitz*
3. *Each ∇f_i is β-Lipschitz*
4. *Assumption A holds.*

Take any constant step size $\gamma \in \left(0, \frac{m(n-1)}{\lambda n(m-1) + \beta(n-m)}\right]$. Then the iterates of minibatch SGD satisfy

$$\mathbb{E}\|\theta_k - \theta^*\|^2 \leq (1 - \gamma\mu)^k \|\theta - \theta_0\|^2.$$

We will also use the following lemma proved in class

Lemma 1. *Let $h : \mathbb{R}^d \to \mathbb{R}$ be such that ∇h is L-Lipschitz. Then it holds that*

$$\|\nabla h(x) - \nabla h(y)\|^2 \leq 2L\big(h(x) - h(y)\big), \quad \forall x, y \in \mathbb{R}^d.$$

Questions

1. For $n = 2$, give an example of an optimization problem (1) where Assumption A holds, and one where it does not (short answer expected).
2. Justify that, for any integer $k \geq 0$,

$$\|\theta_{k+1} - \theta^*\|^2 = \|\theta_k - \theta^*\|^2 - 2\gamma \left\langle \theta_k - \theta^*, \nabla f_{v_{k+1}}(\theta_k) \right\rangle + \gamma^2 \left\|\nabla f_{v_{k+1}}(\theta_k)\right\|^2.$$

3. Writing $\mathbb{E}_k$ for the expectation conditional on θ_k, prove that:
$$\mathbb{E}_k\left\|\theta_{k+1}-\theta^*\right\|^2 \leq \left(1-\gamma\mu\right)\left\|\theta_k-\theta^*\right\|^2 - 2\gamma\left[F\left(\theta_k\right)-F\left(\theta^*\right)\right] + \gamma^2\mathbb{E}_k\left\|\nabla f_{v_{k+1}}\left(\theta_k\right)\right\|^2.$$
(One can use the strong convexity of F: $\forall x, y \in \mathbb{R}^d : F(x) - F(y) \geq \langle \nabla F(y), x-y\rangle + \frac{\mu}{2}\|x-y\|^2$).

4. For any $m \in \{1,\dots,n\}$, we denote by $\mathcal{D}_m$ the uniform distribution over subsets of $\{1,\dots,n\}$ whose cardinality is m, and for any $\theta \in \mathbb{R}^d$, we define $\sigma_m^2(\theta) = \mathbb{E}_{v\sim\mathcal{D}_m}\left\|\nabla f_v(\theta)\right\|^2$. How can one rewrite $\sigma_1^2(\theta)$ and $\sigma_n^2(\theta)$ without expectations and only using sums?
Remark: Note that here we are considering $\sigma_m^2(\theta)$, not $\sigma_m^2(\theta^*)$ as in the first exercise session.

5. Justify that, for any $\theta \in \mathbb{R}^d$ and any $m \in \{1,\dots,n\}$, we have
$$\sigma_m^2(\theta) = \frac{1}{m^2}\sum_{i,j=1}^{n}\left\langle \nabla f_i(\theta), \nabla f_j(\theta)\right\rangle \mathbb{P}_{v\sim\mathcal{D}_m}\big(i\in v \text{ and } j\in v\big).$$

6. Justify that for any integers $i, j \in \{1,\dots,n\}$, we have
$$\mathbb{P}_{v\sim\mathcal{D}_m}\big(i\in v \text{ and } j\in v\big) = \begin{cases} \frac{m}{n} & \text{if } i=j \\ \frac{m(m-1)}{n(n-1)} & \text{if } i\neq j. \end{cases}$$

7. Deduce that
$$\sigma_m^2(\theta) = \frac{1}{n-1}\Big(\frac{n}{m}-1\Big)\cdot\frac{1}{n}\sum_{i=1}^{n}\left\|\nabla f_i(\theta)\right\|^2 + \frac{n(m-1)}{m(n-1)}\left\|\nabla F(\theta)\right\|^2.$$

8. Using Assumption A and Lemma 1, deduce that
$$\sigma_m^2(\theta) \leq 2\left(F(\theta)-F(\theta^*)\right)\frac{\beta(n-m)+\lambda n(m-1)}{m(n-1)}.$$

9. Using Question 3, conclude that if γ is chosen as specified in the theorem, then
$$\mathbb{E}_k\left\|\theta_{k+1}-\theta^*\right\|^2 \leq \left(1-\gamma\mu\right)\left\|\theta_k-\theta^*\right\|^2.$$

10. Conclude the proof of the theorem.

11. This theorem shows that minibatch SGD achieves an exponential rate of convergence in the interpolation regime even when $m=1$, where the algorithm reduces to SGD. In comparison, what is the classical rate of SGD for Problem (1) if we only assume that F is μ-strongly convex and L Lipschitz, but do not assume that Assumption A holds? How should the step size be chosen in this case?

12. Among all favorable values $\gamma \in \left(0, \frac{m(n-1)}{\lambda n(m-1)+\beta(n-m)}\right]$, which value γ^* yields the fastest convergence?

13. Assume that $\beta = \lambda$. How does γ^* simplify? What optimal batch size m^* should one choose in this case to minimize the runtime of the algorithm? How does γ^* compare with the optimal step size of the gradient descent algorithm?

14. We are interested in computing the runtime of mini-batch SGD with mini-batches of size m and step size γ^* to reach precision ε in the interpolation setting. Assuming that computing one gradient takes 1 second, show that the runtime is proportional to
$$\frac{m\log\left(\frac{\|\theta_0-\theta^*\|^2}{\varepsilon}\right)}{\log\left(1-\gamma^*\mu\right)}.$$

15. Assuming that μ is small enough to use the approximation $\log(1-\gamma^*\mu) \approx -\gamma^*\mu$ for any m, simplify this expression and minimize it as a function of $m \in \{1,\dots,n\}$. Note that the result should be different depending on whether $\lambda n > \beta$ or $\lambda n \leq \beta$. What do you notice? How does the optimal batch size m^* differ from the classical optimal batch size if the interpolation setting is not in force?