CPSC 532D, Fall 2023: Assignment 4
due Wednesday December 20th, **11:59 pm**

Use LaTeX, like usual.

You can do this with a partner if you'd like (there's a "find a group" post on Piazza). If so, **do not just split the questions up**; if you hand in an assignment with your name on it, you're pledging that you participated in and understand all of the solutions. (If you work with a partner on some problems and then end up doing some of them separately, hand in separate answers and put a note in each question saying whether you did it with a partner or not.)

*If you look stuff up anywhere other than in SSBD or MRT, **cite your sources**: just say in the answer to that question where you looked. If you ask anyone else for help, **cite that too**. Please do not look at solution manuals / search for people proving the things we're trying to prove / etc. Also, please do* not *ask ChatGPT or similar models. It's okay to talk to others outside your group about general strategies – if so, just say who and for which questions – but **not** to sit down and do the assignment together.*

Submit your answers as a single PDF on Gradescope: here's the link. Make sure to use the Gradescope group feature if you're working in a group. *You'll be prompted to mark where each question is in your PDF; make sure you mark all relevant pages for each part (which saves a surprising amount of grading time).*

Please **put your name(s) on the first page** as a backup, just in case. If something goes wrong, you can also email your assignment to me directly (`dsuth@cs.ubc.ca`).

1 Expectation bounds imply PAC-learning [10 points]

Our SGD bound, as well as the stability bound that we actually proved (not the one relying on appealing to a complicated proof we didn't cover), only showed learning in expectation. This problem establishes that this is equivalent to PAC learning, albeit maybe with a bad rate.

Let $\mathcal{A}$ be a learning algorithm, $\mathcal{D}$ a probability distribution, and ℓ a loss function bounded in $[0, 1]$. For brevity's sake, let L be the random variable $L_{\mathcal{D}}(\mathcal{A}(S))$.

Prove that the following two statements are equivalent:

1. There is some $m(\varepsilon, \delta)$ such that for every $\varepsilon, \delta \in (0, 1)$, for all $m \geq m(\varepsilon, \delta)$, $\Pr_{S\sim\mathcal{D}^m}(L > \varepsilon) < \delta$.
2. $\mathcal{A}$'s expected loss is asymptotically zero: $\lim_{m\to\infty} \mathbb{E}_{S\sim\mathcal{D}^m} L = 0$.

Answer: TODO

2 A really hard convex-Lipschitz-bounded problem [15 points]

Recall that we showed in class that regularized loss minimization can learn any convex-Lipschitz-bounded problem: if $h \mapsto \ell(h, z)$ is convex and ρ-Lipschitz for each $z \in \mathcal{Z}$, $\mathcal{H}$ is convex, and there is some strongly convex function $R(h)$ – e.g. $R(h) = \frac{1}{2}\|h\|^2$ – such that $R(h^*) \leq \frac{1}{2}B^2$, then regularized loss minimization with the right choice of regularization weight can find $\hat{h}$ such that $L_{\mathcal{D}}(\hat{h}) \leq L_{\mathcal{D}}(h^*) + \mathcal{O}(1/\sqrt{m})$, either appealing to the complicated paper or by our expectation bound plus Question 1. We also showed that in this setting, gradient descent can implement regularized loss minimization up to ε accuracy using $\mathcal{O}(1/\varepsilon^2)$ gradient steps.[1] *Thus, any convex-Lipschitz-bounded problem can be PAC-learned in polynomially many gradient steps.*

This doesn't *guarantee that convex-Lipschitz-bounded problems can be efficiently learned.*

Let $\mathcal{H} = [0, 1]$ – nice and simple – but let the example domain $\mathcal{Z}$ be the class of all pairs of Turing machines T and input strings s. Define

$$\ell(h, (T, s)) = \begin{cases} \mathbb{1}(T \text{ halts on the input } s) & \text{if } h = 0 \\ \mathbb{1}(T \text{ does not halt on the input } s) & \text{if } h = 1 \\ (1-h)\ell(0, (T, s)) + h\ell(1, (T, s)) & \text{if } 0 < h < 1. \end{cases}$$

Prove that this problem is convex-Lipschitz-bounded, but no computable algorithm can PAC-learn it.

Hint: Think about what the loss minimizer h^*, or the ERM, represents with this loss.

Hint: If you have no idea what I'm talking about: look up the "halting problem."

Answer: TODO

[1] You can actually show $\mathcal{O}(1/\varepsilon)$; we didn't assume strong convexity in our bound.

3 Learning without concentration [25 points]

We're going to do an unsupervised learning task, where we try to estimate the mean of a distribution, but we do it with some *missing* observations. Specifically, let $\mathcal{B}$ be the closed unit ball $\mathcal{B} = \{w \in \mathbb{R}^d : \|w\| \leq 1\}$, and let the samples be in $\mathcal{Z} = \mathcal{B} \times \{0,1\}^d$, where an entry $z = (x, \alpha)$ with α is a binary "mask" vector indicating whether the given entry is missing. We want to estimate the mean ignoring the missing entries, i.e. $\mathcal{H} = \mathcal{B}$ and

$$\ell(w, (x, \alpha)) = \sum_{i=1}^{d} \begin{cases} 0 & \text{if } \alpha_i = 1 \\ (x_i - w_i)^2 & \text{if } \alpha_i = 0. \end{cases}$$

[3.1] [10 points] Show that regularized loss minimization can PAC-learn this problem with a sample complexity independent of d.

Hint: Feel free to use the result of Question 1 and results from class.

Answer: TODO

[3.2] [10 points] Let $\mathcal{D}$ be a distribution where x is always the fixed vector 0, and α has its entries i.i.d. $\text{Unif}(\{0,1\}) = \text{Bernoulli}(1/2)$. Let $m_{\mathcal{D}}(\varepsilon, \delta)$ denote the sample complexity of uniform convergence for this $\mathcal{D}$, so that if $m \geq m_{\mathcal{D}}(\varepsilon, \delta)$, then

$$\Pr_{S \sim \mathcal{D}^m} \left(\sup_{w \in \mathcal{H}} L_{\mathcal{D}}(w) - L_S(w) \leq \varepsilon \right) \geq 1 - \delta.$$

Show that for some particular value of $\varepsilon > 0$ and $\delta > 0$, $m_{\mathcal{D}}(\varepsilon, \delta)$ increases with d.

Hint: Show that if d is large enough relative to m, you're likely to get at least one dimension j where $(\alpha_i)_j = 1$ for all *your m samples $x_i \in S_x$.*

Answer: TODO

[3.3] [5 points] Describe a problem where RLM is a PAC learner, but uniform convergence doesn't hold. Why doesn't this contradict the fundamental theorem of statistical learning?

Answer: TODO

4 Maximizing differences [40 + 5 challenge points]

Let's consider learning a kernel classifier with the somewhat unusual *linear loss*, $\ell(h,(x,y)) = -yh(x)$, where $y \in \{-1,1\}$. Assume a continuous kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with associated RKHS $\mathcal{F}$ and canonical feature map $\varphi : \mathcal{X} \to \mathcal{F}$ with $k(x,x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}}$.

[4.1] [10 points] Find the regularized loss minimizer

$$\hat{h}_\lambda = \operatorname*{arg\,min}_{h \in \mathcal{F}} L_S(h) + \tfrac{1}{2}\lambda \|h\|_{\mathcal{F}}^2, \tag{RLM}$$

for a training sample $S = ((x_1,y_1),\dots,(x_n,y_n))$ and $\lambda > 0$.

Answer: TODO

[4.2] [5 points] Show that $L_S(\hat{h}_\lambda) = -\frac{1}{\lambda}\left\| \frac{1}{n}\sum_{i:y_i=1}\varphi(x_i) - \frac{1}{n}\sum_{i:y_i=-1}\varphi(x_i) \right\|_{\mathcal{F}}^2$.

Answer: TODO

[4.3] [10 points] Find a (data-dependent) value of λ, call it $\hat{\lambda}$, such that $\|\hat{h}_{\hat{\lambda}}\|_{\mathcal{F}} = 1$, and simplify the expression for $L_S(\hat{h}_{\hat{\lambda}})$.

Answer: TODO

[4.4] [5 points] Argue that $\hat{h}_{\hat{\lambda}}$ is a solution to

$$\min_{h \in \mathcal{F} : \|h\|_{\mathcal{F}} \le 1} L_S(h). \tag{ERM}$$

Further argue that solving (ERM) is equivalent to solving

$$\max_{h \in \mathcal{F} : \|h\|_{\mathcal{F}} \le 1} \sum_{i:y_i=1} h(x_i) - \sum_{i:y_i=-1} h(x_i), \tag{MAX}$$

i.e. finding a function high on the positively-labeled points and low on the negatively-labeled ones.

Answer: TODO

Let $\mathcal{P}$ and $\mathcal{Q}$ be probability distributions. A distribution-level version of (MAX) is known as the *maximum mean discrepancy*,

$$\mathrm{MMD}(\mathcal{P},\mathcal{Q}) = \sup_{f \in \mathcal{F} : \|f\|_{\mathcal{F}} \le 1} \mathop{\mathbb{E}}_{X \sim \mathcal{P}} f(X) - \mathop{\mathbb{E}}_{Y \sim \mathcal{Q}} f(Y).$$

Let $\varphi : \mathcal{X} \to \mathcal{F}$ be the canonical feature map $k(x,x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}}$, and assume for simplicity that $\sup_{x \in \mathcal{X}} \|\varphi(x)\| \le \kappa < \infty$. Define the *kernel mean embedding* of a distribution $\mathcal{P}$ as $\mu_{\mathcal{P}} = \mathbb{E}_{X \sim \mathcal{P}} \varphi(X)$; for bounded kernels, this is guaranteed to exist.[2] Moreover, you can move the expectation inside or outside of inner products: for any $f \in \mathcal{F}$,

$$\langle \mu_{\mathcal{P}}, f \rangle_{\mathcal{F}} = \left\langle \mathop{\mathbb{E}}_{X \sim \mathcal{P}} \varphi(X), f \right\rangle_{\mathcal{F}} = \mathop{\mathbb{E}}_{X \sim \mathcal{P}} \langle \varphi(X), f \rangle_{\mathcal{F}} = \mathop{\mathbb{E}}_{X \sim \mathcal{P}} f(X).$$

[4.5] [10 points] Prove that

$$\mathrm{MMD}(\mathcal{P},\mathcal{Q}) = \|\mu_{\mathcal{P}} - \mu_{\mathcal{Q}}\|_{\mathcal{F}}$$

and

$$\mathrm{MMD}^2(\mathcal{P},\mathcal{Q}) = \mathop{\mathbb{E}}_{\substack{X,X' \sim \mathcal{P} \\ Y,Y' \sim \mathcal{Q}}} \Big[k(X,X') - 2k(X,Y) + k(Y,Y') \Big].$$

Answer: TODO

[2] As long as $\mathcal{P}$ is a Borel measure, which is the kind of very mild assumption we don't worry about in this class.

[4.6] [2 challenge points] Let $\mathcal{X}$ be a compact metric space. Prove that if k is universal, then $\mathrm{MMD}(\mathcal{P}, \mathcal{Q}) = 0$ implies $\mathcal{P} = \mathcal{Q}$.

Hint: You can use the following helpful result, where $C(\mathcal{X})$ is as usual the space of all bounded continuous functions $\mathcal{X} \to \mathbb{R}$.

Lemma 4.1. *Two Borel probability measures $\mathcal{P}$ and $\mathcal{Q}$ on a metric space $\mathcal{X}$ are equal if and only if for all $f \in C(\mathcal{X})$, $\mathbb{E}_{X\sim\mathcal{P}} f(X) = \mathbb{E}_{Y\sim\mathcal{Q}} f(Y)$.*

Answer: TODO

[4.7] [3 challenge points] Prove that $k(x, y) = \|x\| + \|y\| - \|x - y\|$, where $\|\cdot\|$ is the norm of any Hilbert space, is a valid kernel. Further show that the MMD with this kernel is exactly the *energy distance*, whose square is

$$\rho(\mathcal{P}, \mathcal{Q})^2 = 2 \mathop{\mathbb{E}}_{X\sim\mathcal{P}, Y\sim\mathcal{Q}} \|X - Y\| - \mathop{\mathbb{E}}_{X,X'\sim\mathcal{P}} \|X - X'\| - \mathop{\mathbb{E}}_{Y,Y'\sim\mathcal{Q}} \|Y - Y'\|.$$

Hint: You can use without proof that for all $n \geq 1$, for all $x_1, \ldots, x_n$ and $c_1, \ldots, c_n$ such that $\sum_{i=1}^n c_i = 0$, it holds that

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i \|x_i - x_j\| c_j \leq 0.$$

You'll need to fiddle a bit from this inequality to get the desired result: how to get the $\|x\|$ in k?

Answer: TODO

We won't prove this, but it turns out that the energy distance is positive for any $\mathcal{P} \neq \mathcal{Q}$, but this k actually isn't *universal.*

5 Lasso and stability [5 challenge points]

The Lasso algorithm uses linear predictors $h_w(x) = w \cdot x$, the square loss $\ell(h, (x, y)) = (h(x) - y)^2$, and a $\|w\|_1 = \sum_{j=1}^{d} |w_j|$ regularizer:

$$A_\lambda(S) \in \operatorname*{arg\,min}_{w \in \mathbb{R}^d} L_S(h_w) + \lambda \|w\|_1.$$

(If there are multiple minimizers, let's have A_λ return one uniformly at random from the set of possible minimizers.) The Lasso algorithm is nice because it often returns sparse solutions, i.e. w with many $w_j = 0$.

Let's use $\mathcal{Z} = \mathcal{X} \times \mathcal{Y} = \{x \in \mathbb{R}^d : \|x\| \le C\} \times [-M, M]$ for simplicity.

[5.1] [5 points] Show that the Lasso algorithm is not uniformly stable. That is, there is no $\beta(m)$ satisfying Definition 3 of the stability notes such that $\beta(m) \to 0$ as $m \to \infty$.

Hint: There's a reason I mentioned multiple minimizers above.

Answer: TODO

I think *the Lasso algorithm for any* $\lambda > 0$ is *actually on-average-replace-one stable under these assumptions on* $\mathcal{Z}$*, because any algorithm that on-average learns* $\mathcal{D}$ *is on-average-replace-one-stable. We can show this under these assumptions for the Lagrange dual problem to the Lasso, ERM with* $\mathcal{H} = \{h_w : \|w\|_1 \le B\}$*, with Rademacher bounds (depending on* B*,* C*, and* M*). But the relationship of* B *to* λ *is complicated, and I don't even know how to get a worst-case upper bound on it, though something might be possible.*[3]

[3] In fact, I'm not 100% sure that Lasso even *does* learn without any distributional assumptions. For typical analyses with some distributional assumptions, see Chapter 8 of the Bach book; e.g. Exercise 8.5 is pretty close.