DSC 190/291 · Assignment 7

UCSD · Spring 2026

Released: Monday, May 18 · Due: Monday, May 25, 11:59 PM

AI policy. AI assistance is allowed and encouraged in this course. You may use AI to learn the material, explore proof structure, test examples, debug code or formalizations, and improve exposition. However, you are responsible for checking correctness and for standing behind every proof step, derivation, formalization, experiment, and explanation you submit. Use AI as a collaborator, not as an oracle: do not submit anything you cannot explain and verify. The AI usage report is a required component of the assignment.

Submission. Submit a single PDF on Gradescope containing your write-up, figures, and discussion. Also place any supporting artifacts for the assignment in your course repository under the appropriate assignment directory. This may include code, Lean files, notebooks, scripts, data, or other materials needed to inspect or reproduce your work. Your submission should make it clear how the repository artifacts relate to the write-up.

Part A: Kernel Methods (10 points)

This homework continues the scale-sensitive complexity theme from Homework 6, where generalization is controlled by norms and margins rather than by dimension.

Let $\mathcal{U}$ be a Hilbert space with inner product $\langle \cdot, \cdot \rangle$ and induced norm $\|\cdot\|$, and let $\varphi : \mathcal{X} \to \mathcal{U}$ be a feature map with associated kernel $K(x, x') = \langle \varphi(x), \varphi(x') \rangle$. For $B > 0$, let $\mathcal{H}_{K,B} = \{x \to \langle w, \varphi(x) \rangle : w \in \mathcal{U}, \|w\| \leq B\}$ be the predictors of RKHS norm at most B. For a sample $S = (x_1, ..., x_n)$ and a real-valued function class $\mathcal{F}$, write the empirical Rademacher complexity

$$\hat{\mathcal{R}}_S(\mathcal{F}) = \mathbb{E}_\sigma \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(x_i),$$

where $\sigma_1, ..., \sigma_n$ are independent uniform signs, and write $\mathcal{R}_{\mathcal{D}^n}(\mathcal{F}) = \mathbb{E}_{S \sim \mathcal{D}^n} \hat{\mathcal{R}}_S(\mathcal{F})$.

1. **(5 points) Norm-based kernel bound.**

   Prove the empirical bound

   $$\hat{\mathcal{R}}_S(\mathcal{H}_{K,B}) \leq \frac{B}{n} \sqrt{\sum_{i=1}^{n} K(x_i, x_i)},$$

   and conclude the distribution-level bound

   $$\mathcal{R}_{\mathcal{D}^n}(\mathcal{H}_{K,B}) \leq B \sqrt{\frac{\mathbb{E}[K(X, X)]}{n}}.$$

2. **(5 points) Representer theorem.**

   The space $\mathcal{U}$ may be infinite-dimensional, yet for any $\lambda > 0$ the regularized objective

$$J(w) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{loss}(\langle w, \varphi(x_i)\rangle; y_i) + \lambda\|w\|^2$$

can be minimized over finitely many coefficients. This is the representer theorem: prove that whenever J attains a minimum over $w \in \mathcal{U}$, some minimizer has the form

$$w^* = \sum_{j=1}^{n} \alpha_j \varphi(x_j), \quad \alpha \in \mathbb{R}^n.$$

Then rewrite J as a minimization over $\alpha \in \mathbb{R}^n$ in which the data enter only through the Gram matrix K_S, with entries $(K_S)_{ij} = K(x_i, x_j)$.

Part B: ℓ_1 Rademacher Complexity (25 points)

Throughout this part, $\hat{\mathcal{R}}_S$ denotes the empirical Rademacher complexity defined in Part A.

1. **(8 points) Convex hulls.**

   Let $\mathcal{F}$ be a finite class of real-valued functions. Its convex hull is

   $$\mathrm{conv}(\mathcal{F}) = \left\{ \sum_{f\in\mathcal{F}} a_f f : a_f \geq 0, \sum_{f\in\mathcal{F}} a_f = 1 \right\}.$$

   Prove that empirical Rademacher complexity is unchanged by taking the convex hull:

   $$\hat{\mathcal{R}}_S(\mathrm{conv}(\mathcal{F})) = \hat{\mathcal{R}}_S(\mathcal{F}).$$

2. **(10 points) ℓ_1 linear predictors.**

   Let $\varphi : \mathcal{X} \to \mathbb{R}^d$ satisfy $\|\varphi(x)\|_\infty \leq R$ for all x, and define

   $$\mathcal{H}_B^1 = \{x \to \langle w, \varphi(x)\rangle : \|w\|_1 \leq B\}.$$

   Derive a logarithmic-in-dimension Rademacher bound for $\mathcal{H}_B^1$. Massart's finite-class lemma bounds the Rademacher complexity of a finite class; the convex-hull identity from the previous problem is what connects $\mathcal{H}_B^1$ to such a class.

3. **(7 points) Comparison with sparsity bounds.**

   In Homework 4 you showed that the class of k-sparse linear classifiers in $\mathbb{R}^d$ has VC dimension $O(k\log(ed/k))$. With the VC generalization bound, with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, every k-sparse linear classifier h satisfies

   $$L_{\mathcal{D}}^{0/1}(h) - L_S^{0/1}(h) \leq C\sqrt{\frac{k\log(ed/k) + \log(1/\delta)}{n}}$$

   for a universal constant C; structural risk minimization makes the bound adapt to the unknown sparsity level k.

   Using your Rademacher bound from the previous problem and the week 7 uniform convergence theorem, derive the analogous bound for $\mathcal{H}_B^1$: a bound on $L_{\mathcal{D}}^{\ell}(w) - L_S^{\ell}(w)$, holding with probability at least $1-\delta$ over the draw of $S \sim \mathcal{D}^n$, uniformly over $\|w\|_1 \leq B$, for a fixed loss ℓ that is G-Lipschitz in the prediction and bounded in $[0,1]$.

These two bounds control different loss functionals, so they are not interchangeable guarantees. Compare them: which quantity is the complexity in each and which one is scale-sensitive; what extra step the ℓ_1 bound needs before it can say anything about 0/1 error; and exhibit a predictor for which the sparsity bound is vacuous while the ℓ_1 bound is not.

Part C: Normalized ℓ_1 Margins and FixedBoost (30 points)

In this part the sample is $S = ((x_1, y_1), ..., (x_n, y_n))$ with labels $y_i \in \{-1, +1\}$, and $\mathcal{G}$ is a finite base class of binary predictors $g : \mathcal{X} \to \{-1, +1\}$, symmetric in the sense that $-g \in \mathcal{G}$ whenever $g \in \mathcal{G}$. A coefficient vector $w = \big(w_g\big)_{g \in \mathcal{G}}$ parameterizes the real-valued predictor $f_w(x) = \sum_{g \in \mathcal{G}} w_g g(x)$, with empirical normalized ℓ_1 margin

$$\text{margin}_S(w) = \min_i \frac{y_i f_w(x_i)}{\|w\|_1},$$

defined whenever $\|w\|_1 > 0$. For a distribution D over the n training examples, the weighted error of a predictor g is $\sum_i D_i \mathbb{1}[g(x_i) \neq y_i]$.

Homework 6′s AdaBoost chooses its step size adaptively to drive training error down quickly. FixedBoost instead uses a fixed step size, so the ℓ_1 scale of the ensemble grows in a controlled way with the number of rounds. This makes the round count an explicit complexity knob and lets boosting be read as a search for a large normalized ℓ_1 margin, the theme of this part.

FixedBoost. Given a step size $\eta > 0$ and a number of rounds T, FixedBoost starts from $w_0 = 0$ and the uniform distribution D^1 over the n training examples. For each round $t = 1, ..., T$:

- pick a base predictor g_t maximizing the weighted correlation $\sum_i D_i^t y_i g_t(x_i)$;
- set $w_t = w_{t-1} + \eta e_{g_t}$, adding η to the weight of g_t;
- reweight the examples by $D_i^{t+1} \propto \exp\big(-y_i f_{w_t}(x_i)\big)$.

FixedBoost outputs the coefficient vector w_T.

1. **(10 points) Margin generalization.**

   Assume S is drawn i.i.d. from a population distribution $\mathcal{D}$ over $\mathcal{X} \times \{-1, +1\}$. Prove a population 0/1 error bound for predictors with empirical normalized margin $\text{margin}_S(w) \geq \gamma$ for some $\gamma > 0$. Build on the generalization-gap bound from B3, specialized to the ramp surrogate from the week 7 lecture, rescaled to γ. State the final bound explicitly in terms of $|\mathcal{G}|$, n, γ, and δ.

2. **(8 points) FixedBoost implication.**

   Use the following theorem about FixedBoost as a black box: if every distribution D over the training sample admits a base predictor of weighted error at most $\frac{1}{2} - \frac{\gamma}{2}$, then after $T = O\big(\frac{\log(n)}{\gamma^4}\big)$ rounds FixedBoost returns coefficients w_T satisfying

   $$\min_i y_i f_{w_T}(x_i) \geq \frac{\gamma}{2} \|w_T\|_1.$$

   Derive a generalization guarantee for the classifier returned by FixedBoost.

3. **(8 points) Weak learning and normalized margin.**

Encode the sample and base class as the matrix $A \in \{-1,+1\}^{n\times|\mathcal{G}|}$ with $A_{ig} = y_i g(x_i)$. The weak-learning edge of a base predictor g under a distribution D over the sample is $\frac{1}{2}$ minus its weighted error, its advantage over random guessing under D; the sample's best weak-learning edge is the largest γ such that every distribution D over the sample admits some $g \in \mathcal{G}$ with edge at least γ.

Prove the exact relationship between the best weak-learning edge and the best normalized ℓ_1 margin $\max_w \text{margin}_S(w)$, using von Neumann's minimax theorem applied to A.

4. **(4 points) Connection to Homework 6.**

   Explain how this margin-based view refines the sparse-boosting analysis from Homework 6. Your answer should identify what changes in the comparator and what changes in the generalization argument.

Part D: FixedBoost Margin Experiment (25 points)

The base class is the $2d$ signed coordinate predictors on Boolean inputs $x \in \{-1,+1\}^d$: the maps $x \to x_j$ and $x \to -x_j$ for $j \in [d]$. This is a finite symmetric base class in the sense of Part C. For a labeled sample $(x_1, y_1), ..., (x_n, y_n)$, form the sign matrix A with $A_{ig} = y_i g(x_i)$, equal to $+1$ when g is correct on example i and -1 otherwise. FixedBoost interacts with the data only through A: its weighted correlations, ensemble margins, and reweighting are all functions of A alone.

This experiment tests whether FixedBoost drives the normalized ℓ_1 margin toward the largest value attainable on a given A. Designing the data is part of the task: choose how the labeled examples (x_i, y_i) are generated, and describe it precisely enough that someone could reproduce your runs. You may use NumPy, plotting libraries, and a linear-programming solver for the optimal-margin computation, but the FixedBoost loop must be your own.

1. **(8 points) Implementation.**

   Implement FixedBoost from scratch, running directly on the matrix A, with a fixed step size. Track $\|w_t\|_1$ and the normalized empirical margin over rounds. You may reuse a boosting harness from Homework 6; state what differs from it.

2. **(10 points) Optimal-margin comparison.**

   Formulate the optimal normalized ℓ_1 margin attainable on a matrix A as a linear program, and solve it. On the same matrix A, plot FixedBoost's normalized-margin trajectory against this optimal value. Do this for at least two instances: one with a large optimal margin, and one harder instance where the optimal margin is small. What structure in A makes the optimal margin large, and what makes it small? The relationship you proved in C3 is one way to reason about this. Construct the two instances accordingly, and report each construction.

3. **(7 points) Interpretation.**

   Does FixedBoost's normalized margin approach the optimal value, and how fast? Compare the observed behavior with the rate implied by the FixedBoost theorem in C2. Identify at least one place where that theory is loose, vacuous, or silent relative to what you observe.

Part E: Agent Skill Reflection (10 points)

A coding-agent skill is a reusable workflow or instruction package that helps an AI coding agent perform a specific kind of task. Examples include a debugging checklist, a reproducibility protocol for experiments, a plotting-and-validation workflow, or a project-specific `SKILL.md` file.

1. **(3 points) Existing workflow.**

   Describe whether you used an existing skill, checklist, prompt template, or workflow while working on this assignment.

2. **(3 points) Skill development.**

   Describe whether you developed or modified a coding-agent skill. If not, propose one concrete skill that would help with this assignment.

3. **(4 points) Verification.**

   Identify plausible coding-agent failure modes in this assignment and explain how you checked the math, code, plots, optimization outputs, and final claims.