Homework III

STAT/CS 5810/6655 - Spring semester 2024

Please upload your solutions in a single pdf file in Canvas. Any requested plots should be sufficiently labeled for full points. Include any code requested.

Unless otherwise stated, programming assignments should use built-in functions in your chosen programming language (Python, R, or Matlab). However, exercises are designed to emphasize the nuances of machine learning and deep learning algorithms - if a function exists that trivially solves an entire problem, please consult with the TA before using it.

1. **(10 pts)** Read the paper "How to avoid machine learning pitfalls: a guide for academic researchers" by Lones. You can find it on Canvas. Write a short (1-2 paragraph) summary of the paper.

2. **Convexity and Optimization (9 pts).** Let $f : \mathbb{R}^d \to \mathbb{R}$.

   (a) (3 pts) Show that if f is strictly convex, then f has at most one global minimizer. Do not assume that the function is differentiable.

   (b) (3 pts) Use the Hessian to give a simple proof that the sum of two convex functions is convex. You may assume that the two functions are twice continuously differentiable.

   (c) (3 pts) Consider the function $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x} + c$ where A is a symmetric $d \times d$ matrix. Derive the Hessian of f. Under what conditions on A is f convex? Strictly convex?

3. **Convex Losses.** We say that a loss is convex if for each fixed y, $L(y, t)$ is a convex function of t.

   (a) (7 pts) Show that the logistic loss is convex, where the logistic loss is defined as $L(y, t) = \log(1 + \exp(-yt))$.

   (b) (6655 - 7 pts) Show that if L is a general, convex loss, then

   $$\hat{R}(\boldsymbol{w}, b) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, \boldsymbol{w}^T\boldsymbol{x}_i + b)$$

   is a convex function of $\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix}$.

   Hints: For part (b), note that the fact that $L(y, t)$ is a convex function of t does NOT guarantee that $L(y, f(\boldsymbol{\theta}))$ is a convex function of $\boldsymbol{\theta}$ for all functions f. You will need to show that for this

specific choice of $f(\boldsymbol{\theta})$, we still have a convex function of $\boldsymbol{\theta}$. Also, L may not be differentiable everywhere in general and so you cannot show this using differentiation. You will need to use some other approach we covered in class.

4. **Alternative OSM hyperplane (10 pts).** An alternative way to extend the max-margin hyperplane to nonseparable data is to solve the following quadratic program (another name for an optimization problem with a quadratic objective function):

$$\begin{array}{cc} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} & \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{C}{n}\sum_{i=1}^n \xi_i^2 \\ s.t. & y_i(\boldsymbol{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i \ \forall i \\ & \xi_i \geq 0 \ \forall i \end{array}$$

The only difference with respect to the OSM hyperplane is that we are now squaring the slack variables. This assigns a stronger penalty to data points that violate the margin.

(a) (4 pts) Which loss is associated with the above quadratic program? In other words, show that learning a hyperplane by the above optimization problem is equivalent to ERM with a certain loss. **Hint**: you should not use the Lagrangian or KKT conditions to do this. Instead, look at the notes where we first talked about the OSM. In those notes, we argued that the OSM optimization problem is equivalent to regularized ERM with the hinge loss. You should follow a similar procedure.

(b) (4 pts) Argue that the second set of constraints can be dropped without changing the solution.

(c) (2 pts) Identify an advantage and a disadvantage of this loss compared to the hinge loss.

5. **Kernels.**

(a) (4 pts) To what feature map Φ does the kernel

$$k(\boldsymbol{u},\boldsymbol{v}) = (\langle \boldsymbol{u},\boldsymbol{v}\rangle + 1)^3$$

correspond? Assume the inputs have an arbitrary dimension d and the inner product is the dot product.

(b) (5810 - 24 pts, 6655 - 12 pts) Let k_1, k_2 be symmetric, positive-definite kernels over $\mathbb{R}^D \times \mathbb{R}^D$, let $a \in \mathbb{R}^+$ be a positive real number, let $f : \mathbb{R}^D \to \mathbb{R}$ be a real-valued function, and let $p : \mathbb{R} \to \mathbb{R}$ be a polynomial with positive coefficients. For each of the functions k below, state whether it is necessarily a positive-definite kernel. If you think it is, prove it. If you think it is not, give a counterexample.

i. $k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z}) + k_2(\mathbf{x},\mathbf{z})$

ii. $k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z}) - k_2(\mathbf{x},\mathbf{z})$

iii. $k(\mathbf{x},\mathbf{z}) = ak_1(\mathbf{x},\mathbf{z})$

iv. $k(\mathbf{x},\mathbf{z}) = k_1(\mathbf{x},\mathbf{z})k_2(\mathbf{x},\mathbf{z})$

v. $k(\mathbf{x},\mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$

vi. $k(\mathbf{x}, \mathbf{z}) = p\left(k_1(\mathbf{x}, \mathbf{z})\right)$

(c) (**6655 - 6 pts**) Explain why or why not the following kernels are PSD. Make heatmaps for each case, with $a = b = 1$ in part iii.

i. $cos(x - y), x, y \in \mathbb{R}$

ii. $cos(x^2 - y^2), x, y \in \mathbb{R}$

iii. $tanh(a(x^T y) + b), x, y \in \mathbb{R}^d$

6. **Support Vector Regression (6655 - 35 pts).** Support vector regression (SVR) is a method for regression analogous to the support vector classifier. Consider a regression problem for which instead of the squared error, we consider the $\epsilon-$insensitive error displayed in Figure 1.

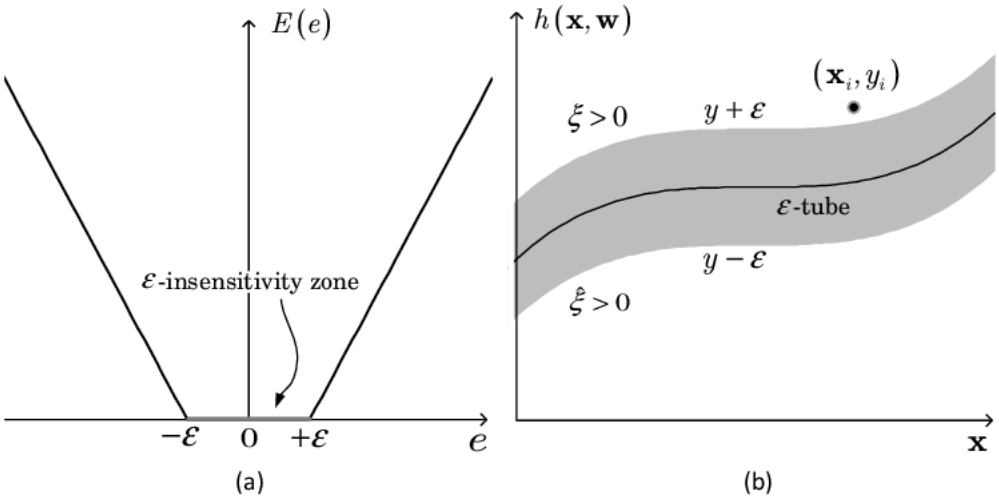

Figure 1: $\epsilon-$insensitive loss

Let $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \ldots, n$ be training data for a regression problem. In the case of linear regression, SVR solves

$$\begin{aligned} \min_{\mathbf{w}, b, \xi^+, \xi^-} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n}\left(\xi_i^+ + \xi_i^-\right) \\ \text{s.t.} \quad & y_i - \mathbf{w}^T\mathbf{x}_i - b \leq \epsilon + \xi_i^+ \ \forall i \\ & \mathbf{w}^T\mathbf{x}_i + b - y_i \leq \epsilon + \xi_i^- \ \forall i \\ & \xi_i^+ \geq 0 \ \forall i \\ & \xi_i^- \geq 0 \ \forall i \end{aligned}$$

where $\mathbf{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$, $\xi^+ = \left(\xi_1^+, \ldots, \xi_n^+\right)^T$, and $\xi^- = \left(\xi_1^-, \ldots, \xi_n^-\right)^T$. Here ϵ is fixed.

(a) (6655 - 8 pts) Show that for an appropriate choice of λ, SVR solves

$$\min_{\mathbf{w}, b} \frac{1}{n}\sum_{i=1}^{n} \ell_\epsilon\left(y_i, \mathbf{w}^T\mathbf{x}_i + b\right) + \lambda\|\mathbf{w}\|^2$$

where $\ell_\epsilon(y,t) = \max\{0, |y-t| - \epsilon\}$is the ϵ-insensitive loss, which does not penalize prediction errors below a level of ϵ.

(b) (6655 - 15 pts) The optimization problem is convex with affine contraints and therefore strong duality holds. Use the KKT conditions to derive the dual optimization problem in a manner analogous to the support vector classifier (SVC). As in the SVC, you should eliminate the dual variables corresponding to the constraints $\xi_i^+ \geq 0$, $\xi_i^- \geq 0$.

(c) (6655 - 8 pts) Explain how to kernelize SVR. Be sure to explain how to recover $\mathbf{w}^*$ and b^* and write the final kernelized regression function $f(\boldsymbol{x})$ in terms of the optimal dual variables and b^*.

(d) (6655 - 4 pts) Argue that the final predictor will only depend on a subset of training examples (i.e. support vectors) and characterize those training examples.

7. **Support Vector Regression (5810 - 36 pts).** In the software that you prefer (R, Matlab, Python), apply SVR to the Crowdeness data to predict the people count in the gym. You may use any build-in methods. You may remove the date column and the timestamp, and use the rest as the features. Try two different kernels and three different values of C. Create a table or plot of the cross-validation results for all combinations above. Briefly comment on the results. The data can be found in `Canvas - Files/Data/HW3/Crowdeness.csv`.