Exercises for Stats315a

John Duchi

March 17, 2025

Contents

1 Linear Algebra Review

Question 1.1 (15 points): Let $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times k}$, $C \in \mathbb{R}^{k\times k}$, and $D \in \mathbb{R}^{k\times n}$. Assume that A and C are invertible. Define the $(n+k)\times(n+k)$ matrix

$$M := \begin{bmatrix} A & B \\ -D & C^{-1} \end{bmatrix}.$$

(a) Show that if $X \in \mathbb{R}^{n\times n}$ and $Y \in \mathbb{R}^{k\times n}$ solve

$$M \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} I \\ 0 \end{bmatrix},$$

where I is the $n \times n$ identity, then $X = (A + BCD)^{-1}$.

(b) Use this to demonstrate the Woodbury matrix identity that

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}.$$

so long as $C^{-1} + DA^{-1}B$ is invertible.

(c) In the case that $k = 1$, so that $B = b$ and $D = d^T$ for n-vectors b and d and $C = c$ is a scalar, give a (slightly) simplified form for the above. Give a sufficient condition for $(A + bcd^T)$ to be invertible.

Question 1.2 (15 points): Let A be a symmetric matrix. We say A is positive definite, denoted $A \succ 0$, if $x^TAx > 0$ for all $x \neq 0$. We say $A \succeq 0$, meaning A is positive semidefinite, if $x^TAx \geq 0$ for all x.

(a) Show that if $A \succ 0$, then A is full rank.

(b) Show that B has linearly independent columns if and only if B^TB is positive definite.

(c) We write $A \succeq B$ if $A - B \succeq 0$. Show that if U is any matrix of appropriate size, then $A + UU^T \succeq A$.

Question 1.3 (Projections, **15 points**): A symmetric matrix $P \in \mathbb{R}^{n\times n}$ is an *orthogonal projector* (or a projection matrix) if $P^2 = P$.

(a) Show that any projection matrix is positive semidefinite, i.e., $x^TPx \geq 0$ for all x.

(b) Show that for any vector v, $x^\star = Pv$ solves

$$\begin{array}{ll} \text{minimize} & \|x - v\|_2^2 \\ \text{subject to} & x \in \text{span}(P) = \{Pz \mid z \in \mathbb{R}^n\}. \end{array}$$

In particular, show the orthogonality relationship $(Pv - v)^Tw = 0$ for all vectors $w \in \text{span}(P)$, and draw a picture of this.

(c) Let $U \in \mathbb{R}^{n\times k}$ have orthonormal columns, that is, satisfy $U^TU = I_k$. Show that $P = UU^T$ is a projection matrix. What does P project onto?

Question 1.4 (Some facts about spectral decompositions, **20 points**): The spectral theorem states that if $A \in \mathbb{R}^{n \times n}$ is symmetric, then $A = U\Lambda U^T$ for a diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, and $U^T U = UU^T = I$.

(a) If $A = U\Lambda U^T$ is the spectral decomposition of A, where $U = [u_1 \ \cdots \ u_n]$ (i.e., the columns of U are $u_1, \ldots, u_n \in \mathbb{R}^n$), show directly that u_i is an eigenvector of A with eigenvalue λ_i.

(b) Show that A is positive definite if and only if its eigenvalues $\lambda_i(A) > 0$ for each i.

(c) If P is a projection matrix (i.e., $P = P^T$ and $P^2 = P$), why do its eigenvalues take only the values 0 and 1?

(d) Given part (c), argue that any projection matrix has the form $P = UU^T$ for a matrix U with orthonormal columns.

Question 1.5 (Singular value decompositions, **10 points**): In this problem, you will review spectral decompositions and demonstrate that matrices have singular value decompositions using the spectral theorem for symmetric matrices.

Let $n \geq k$ and $A \in \mathbb{R}^{n \times k}$ be an arbitrary matrix with linearly independent columns. Let $C = A^T A \in \mathbb{R}^{k \times k}$, have eigenvalue decomposition $C = V\Lambda V^T$. Using this spectral decomposition, show that A has a singular value decomposition

$$A = U\Sigma V^T,$$

where $U \in \mathbb{R}^{n \times k}$ and $V \in \mathbb{R}^{k \times k}$ have orthogonal columns, that is, $U^T U = I_k$ and $V^T V = I_k$, and

$$\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_k)$$

is the $k \times k$ diagonal matrix of singular values $\sigma_1 \geq \cdots \geq \sigma_k > 0$ of A. *Hint.* Use the results of Questions 1.2 and 1.4 to show that $\Lambda \succ 0$. Then *define* $U = AV\Lambda^{-1/2}$. Check that $U^T U = I$ and solve for A.

As a remark, it is possible to modify this argument to the case that Λ is not invertible, demonstrating that *any* matrix $A \in \mathbb{R}^{n \times k}$ (with $n \geq k$; the case that $n < k$ everything is simply transposed) has a singular value decomposition $A = U\Sigma V^T$, where Σ is a diagonal $k \times k$ matrix with $\operatorname{rank}(A)$ non-zero entries. To do this, one partitions V into blocks $V = [V_1 \ V_2]$, where $V_1 \in \mathbb{R}^{k \times m}$ corresponds to the non-zero eigenvalues of $A^T A$ and $V_2 \in \mathbb{R}^{k \times (m-k)}$ to the zero eigenvalues of A. Using the same style of argument as that in Question 1.2, one argues that $\|Av\|_2^2 = 0$ for any vector v in the span of the columns of V_2 and that $AV_1 V_1^T = A$.

Question 1.6 (**15 points**): The *operator norm* of a matrix C is defined by

$$\left\|\!\left|C\right|\!\right\|_{\text{op}} := \max_{\|u\|_2 = 1, \|v\|_2 = 1} u^T C v = \max_{\|u\|_2 \leq 1, \|v\|_2 \leq 1} u^T C v. \tag{1.1}$$

(a) If $D = \operatorname{diag}(d_1, \ldots, d_n)$, argue that $\left\|\!\left|D\right|\!\right\|_{\text{op}} = \|d\|_\infty = \max_i |d_i|$. *Hint.* Cauchy-Schwarz.

(b) Let $A = U\Sigma V^T$ be the singular value decomposition of A. Show that $\left\|\!\left|A\right|\!\right\|_{\text{op}} = \max_i |\sigma_i(A)|$, where $\sigma_i(A)$ are the singular values of A. *Hint.* First show that if Q has orthonormal columns, then $\left\|Q^T u\right\|_2 \leq 1$ for any unit vector u.

(c) If A is symmetric with eigenvalues $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$, why is $\left\|\!\left|A\right|\!\right\|_{\text{op}} = \max\{|\lambda_1(A)|, |\lambda_n(A)|\}$?

Question 1.7 (Weyl's Inequalities **15 points**): The *Courant Fisher* representation for the (real) eigenvalues of a symmetric matrix $A \in \mathbb{R}^{n\times n}$, $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$, states that

$$A = \min_{\dim(S)=n-k+1} \max_{x\in S, \|x\|_2=1} x^T A x,$$

where the minimum ranges over subspaces $S \subset \mathbb{R}^n$ of dimension $n-k+1$ and the maximum over vectors $x \in S$ with unit norm $\|x\|_2 = 1$.

(a) Show that for symmetric matrices A and B and any $k \in \{1, \ldots, n\}$,

$$\lambda_n(B) \leq \lambda_k(A+B) - \lambda_k(A) \leq \lambda_1(B).$$

(This is one of the *Weyl inequalities*.)

(b) Recall the *operator norm* (1.1) of a matrix C. Show that part (a) implies the eigenvalues of symmetric matrices are thus *Lipschitz with respect to the operator norm*, that is,

$$|\lambda_k(A+B) - \lambda_k(A)| \leq |\!|\!| B |\!|\!|_{\text{op}}.$$

(c) Show by counterexample that the eigenvectors of a symmetric matrix need not be continuous in the entries of the matrix. *Hint.* Consider the identity matrix I. What are its eigenvectors? Are they unique?

2 Probability distributions

Question 2.1 (**5 points**): Let $X \sim \mathsf{N}(\mu_1, \Sigma_1)$ and $Y \sim \mathsf{N}(\mu_2, \Sigma_2)$ be independent. Give the distribution of

$$Z = AX + BY.$$

(You do not need to prove your answer is correct.)

Question 2.2 (Conditioning and normal distributions, **10 points**):

(a) Let

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathsf{N}(\mu, \Sigma) \text{ where } \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$$

are partitioned appropriately. Give the conditional distribution of $X \mid Y = y$. You should derive this. *Hint.* You may use that a partitioned matrix has the particular inverse

$$K := \Sigma^{-1} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} = \begin{bmatrix} S_{11} & -S_{11}\Sigma_{12}\Sigma_{22}^{-1} \\ -\Sigma_{22}^{-1}\Sigma_{21}S_{11} & \Sigma_{22}^{-1} + \Sigma_{22}^{-1}\Sigma_{21}S_{11}\Sigma_{12}\Sigma_{22}^{-1} \end{bmatrix},$$

where $S_{11}^{-1} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$ is the Schur complement of Σ with respect to Σ_{22}.

(b) Is $\mathrm{Cov}(X \mid Y) \preceq \mathrm{Cov}(X)$?

Question 2.3 (**10 points**): Let $X_i \overset{\text{iid}}{\sim} \mathsf{N}(\mu, \sigma^2)$, $i = 1, \ldots, n$. Define the sample mean $\overline{X}_n = \frac{1}{n}\sum_{i=1}^n X_i$ and the leave-one-out sample means $\overline{X}_{-i} = \frac{1}{n-1}\sum_{j \neq i} X_j$. Give the distribution of the vector $Z \in \mathbb{R}^n$ with entries

$$Z_i := \overline{X}_{-i} - X_i.$$

Hint. Write $Z = AX$ for a matrix A, where $X = [X_i]_{i=1}^n$.

Question 2.4 (Distances in high dimensions, **15pts**): Let $X_i \in \{-1, 1\}^p$ be uniformly distributed on the hypercube, and let P be the uniform distribution on $\{-1, 1\}^p$, so that $X_i \overset{\text{iid}}{\sim} P$.

(a) Show that for any vector $v \in \mathbb{R}^p$,

$$\mathbb{E}[\exp(X^T v)] \leq \exp\left(\frac{1}{2}\|v\|_2^2\right).$$

It may be useful to use that $\frac{1}{2}(e^t + e^{-t}) \leq e^{t^2/2}$, valid for all $t \in \mathbb{R}$.

(b) Show that for any independent $X_1, X_2 \overset{\text{iid}}{\sim} P$ and $\lambda \in \mathbb{R}$,

$$\mathbb{E}[\exp(\lambda X_1^T X_2)] \leq \exp\left(\frac{\lambda^2 p}{2}\right).$$

(c) Using a Chernoff bound, show that for any $t \geq 0$,

$$\mathbb{P}\left(\|X_1 - X_2\|_2^2 \leq 2p(1-t)\right) \leq \exp\left(-\frac{pt^2}{2}\right) \text{ and } \mathbb{P}\left(\|X_1 - X_2\|_2^2 \geq 2p(1+t)\right) \leq \exp\left(-\frac{pt^2}{2}\right).$$

Question 2.5 (A curse of dimensionality, **10 pts**): Let P be the uniform distribution on $\{-1, 1\}^p$ for some $p \in \mathbb{N}$. The results of Question 2.4 will be useful for this question.

(a) Let $X_i \overset{\text{iid}}{\sim} P$ for $i = 1, \ldots, N$ and $\delta \in (0, 1)$. Show that if

$$N \leq \exp\left(\frac{pt^2}{4} - \frac{1}{2}\log\frac{1}{\delta}\right)$$

then $2p(1-t) \leq \|X_i - X_j\|_2^2 \leq 2p(1+t)$ for all $i \neq j$ with probability at least $1 - \delta$.

(b) Conclude that even if we draw a sample of size N exponential in the dimension p, we expect each pair X_i, X_j, $i \neq j$, to have ℓ_2 distance $\|X_i - X_j\|_2 \approx \sqrt{2p}$ with high probability.

3 Loss functions and decision Theory

Question 3.1 (**15 points**): Consider a k-class classification problem on a domain $\mathcal{X}$, where $f : \mathcal{X} \to \mathbb{R}^k$ is the prediction function and the loss $\ell : \mathbb{R}^k \times \mathcal{Y} \to \mathbb{R}$ measures the loss of a prediction $f(x)$ on an example (x, y). Give the function minimizing the expected loss $L(f) := \mathbb{E}[\ell(f(X), Y)]$ for the following choices of loss $\ell : \mathbb{R}^k \times \mathcal{Y} \to \mathbb{R}$.

(a) The squared error $\ell(v, y) = \frac{1}{2} \|v - e_y\|_2^2$, where e_y is the yth standard basis vector.

(b) The squared hinge loss

$$\ell(v, y) = (1 - v_y)_+^2 + \sum_{j \neq y} (1 + v_j)_+^2 ,$$

where $(t)_+ = \max\{t, 0\}$. *Hint.* If $p^\star = [P(Y = y \mid X = x)]_{y=1}^k$ is the vector of conditional probabilities of Y given X, you might use that $1 - p_y^\star = \sum_{j \neq y} p_j^\star$.

Question 3.2 (**10 points** A two parameter prediction in regression)**:** Consider a regression problem of predicting a scalar response $y \in \mathbb{R}$ from an input $x \in \mathcal{X}$. Let $f : \mathcal{X} \to \mathbb{R}$ and $s : \mathcal{X} \to \mathbb{R}_+$. Give the minimizers of $L(f, s) := \mathbb{E}[\ell(f(X), s(X), Y)]$ (in f and s) for the normalized squared loss

$$\ell(t, u, y) = \frac{(t - y)^2}{u} + u.$$

(Assume that $\mathrm{Var}(Y \mid X = x) > 0$ for each x.)

Question 3.3 (Surrogate losses in binary classification, **20 points**)**:** Consider the margin-based classification problem with data in pairs $(x, y) \in \mathcal{X} \times \{-1, 1\}$, where we seek a predictor $f : \mathcal{X} \to \mathbb{R}$ with large margin $yf(x)$. Let the loss

$$\ell(s, y) = \phi(sy)$$

for a convex $\phi : \mathbb{R} \to \mathbb{R}_+$. The loss is *infinite sample consistent* (or just consistent) for the zero-one error if for any distribution on Y, where $p = \mathbb{P}(Y = 1)$, the minimizer

$$s_\phi^\star(p) := \operatorname*{argmin}_{s \in \mathbb{R}} \{\mathbb{E}[\ell(s, Y)] = p\phi(s) + (1 - p)\phi(-s)\}$$

satisfies

$$\mathrm{sign}(s_\phi^\star(p)) = \mathrm{sign}(2p - 1)$$

whenever $p \in (0, \frac{1}{2}) \cup (\frac{1}{2}, 1)$. In the case that $p \in \{0, 1\}$, we require that

$$\inf_{s(2p-1) \leq 0} \{p\phi(s) + (1 - p)\phi(-s)\} > \inf_{s \in \mathbb{R}} \{p\phi(s) + (1 - p)\phi(-s)\} ,$$

but we will ignore that for this question.

(a) Show that if ϕ is differentiable and $\phi'(0) < 0$, then the loss is consistent.

(b) Let ϕ be differentiable with $\phi'(0) < 0$ and $\lim_{s \to \infty} \phi(s) = 0$. Give a transformation $h : \mathbb{R} \to [0, 1]$ from scores s to probabilities such that

$$s^\star = \operatorname{argmin} \{p\phi(s) + (1 - p)\phi(-s)\} \quad \text{if and only if} \quad p = h(s^\star).$$

Hint. You may use that for a convex ϕ, the derivative ϕ' is non-decreasing.

(c) Let $\phi_{\log}(s) = \log(1 + e^{-s})$ be the logistic loss. Give $s^\star_\phi(p)$ and the transformation h.

(d) Let $\phi_{\exp}(s) = \exp(-s)$ be the exponential loss. Give $s^\star_\phi(p)$ and the transformation h.

(e) Let ϕ be the hinge loss $\phi(s) = (1 - s)_+$. Give $s^\star_\phi(p)$, and show that there is no transformation of the form in part (b).

Question 3.4 (Quantile losses, **15 points**): For $\alpha \in (0, 1)$, define the loss function

$$\ell_\alpha(t) := \alpha\,(t)_+ + (1 - \alpha)\,(-t)_+\,,$$

and let Y be a random variable on $\mathbb{R}$. Let

$$\varphi_\alpha(t) = \mathbb{E}[\ell_\alpha(t - Y)]$$

and define the quantile function $Q_\beta(Y) := \inf\{t \mid \mathbb{P}(Y \le t) \ge \beta\}$.

(a) Assume that Y has a density. Show that the unique minimizer of φ_α is $t^\star = Q_{1-\alpha}(Y)$.

(b) Assume that Y has an arbitrary distribution[1]. Show that $t^\star = Q_{1-\alpha}(Y)$ is a minimizer of φ_α.

Hint. For convex functions, one can always exchange integration and differentiation [5]. For a convex function h, define the directional derivative at θ in the direction v by

$$h'(\theta; v) := \lim_{\delta \downarrow 0} \frac{h(\theta + \delta v) - h(\theta)}{\delta} = \liminf_{\delta \downarrow 0} \frac{h(\theta + \delta v) - h(\theta)}{\delta},$$

where the equality follows by convexity. You may use that if for each x, the function $h_x(\theta)$ is convex in θ, then the expected function $f(\theta) = \mathbb{E}[h_X(\theta)]$ satisfies

$$f'(\theta; v) = \mathbb{E}[h'_X(\theta; v)].$$

In particular, in one dimension, one may interchange left and right derivatives and expectation.

Question 3.5 (A loan data analysis challenge, **35 points**): A company in Chile uses crowdsourcing to fund loans to the public, as a means to offer relief from the high bank interest rates. The data in this challenge consists of historical loan records for a sample of 9000 past customers. The variables characterize some aspects of the loan, such as duration, amount, interest rate and many other more technical features of the loans. There are also a number of qualitative variables, such as reason for loan, quality rating of the borrower and others. The response variable y of interest is `default`: a 0-1 variable indicating whether or not the borrower has defaulted on their loan payments.

The company would like to build a default risk score so that they can target high-risk customers early and perhaps preempt the default event, which ends up costly for all involved. (The fraction of defaults in the entire population is around 7%.) The training data `loan-train.csv` represents a sample of 3000 defaulters, and 6000 non-defaulters, and contains 30 features and the binary outcome `default` (in the first column). The file `loan-testx.csv` consists of a random sample of 10000 other customers from the general pool. For these you are provided only the 30 features.

Your job is to build a *risk score*, that is, a model that estimates the probability of default $y = 1$. Feel free to use any of the tools discussed in the lectures of this class (or beyond). Some packages that may be useful include `pytorch`, `xgboost`, and just regular old logistic regression. Describe what you implemented, how you selected your final model. The only thing you need to submit is a text file with 10000 lines; on each line, your should have your predicted risk estimate for each test customer, in the same order as `loan-testx.csv`. Submit this as a .txt file on Gradescope.

[1] If you are worried about the case that $\mathbb{E}[|Y|] = +\infty$, replace $\ell_\alpha(t - Y)$ with $\ell_{\alpha,\mathrm{b}}(t, Y) := \ell_\alpha(t - Y) - \alpha\,(-Y)_+ - (1 - \alpha)\,(Y)_+$, which always has finite expectation as $|\ell_{\alpha,\mathrm{b}}(t, Y)| \le |t|$.

4 Linear Regression

Question 4.1 (Ridge regression risks, **25pts**): Consider the ℓ_2-regularized (ridge) regression estimator

$$\widehat{\beta}_\lambda := \operatorname*{argmin}_b \left\{ \frac{1}{2} \|Xb - y\|_2^2 + \frac{\lambda}{2} \|b\|_2^2 \right\},$$

where $X = [x_1 \ \cdots \ x_n]^T \in \mathbb{R}^{n \times p}$ is the (fixed) design matrix and $y \in \mathbb{R}^n$ is the response. Let $H_\lambda = X(X^T X + \lambda I)^{-1} X^T$, and assume that

$$y_i = f(x_i) + \varepsilon_i \tag{4.1}$$

where $\mathbb{E}[\varepsilon_i] = 0$ and $\mathbb{E}[\varepsilon_i^2] = \sigma^2$. (Note that we *do not* assume that $y_i = x^T \beta^\star + \varepsilon_i$.) Recall also that the in-sample risk of an estimate $\widehat{f}$ of f is

$$R_{\text{in}}(\widehat{f}) := \frac{1}{n} \sum_{i=1}^n \mathbb{E}[(\widehat{f}(x_i) - f(x_i))^2] = \frac{1}{n} \sum_{i=1}^n \left(\text{Bias}(\widehat{f}(x_i))^2 + \text{Var}(\widehat{f}(x_i)) \right)$$

where the expectation is taken over y_i drawn in model (4.1). Define the mean values of y by

$$\mu := \mathbb{E}[y] = [f(x_i)]_{i=1}^n.$$

Throughout the remainder of the question, let $\widehat{f}_\lambda$ be the linear function $\widehat{f}_\lambda(x) = x^T \widehat{\beta}_\lambda$ given by the ridge estimator.

(a) Show that

$$n \cdot R_{\text{in}}(\widehat{f}_\lambda) = \|(I - H_\lambda)\mu\|_2^2 + \sigma^2 \operatorname{tr}(H_\lambda^T H_\lambda).$$

(b) Show that the residual sum of squares $\text{RSS} = \sum_{i=1}^n (\widehat{f}_\lambda(x_i) - y_i)^2$ (this is just the training squared error) satisfies

$$\mathbb{E}[\text{RSS}] = n R_{\text{in}}(\widehat{f}_\lambda) + \sigma^2 (n - 2 \operatorname{tr}(H_\lambda)).$$

For the remainder of the question, assume that the design $X \in \mathbb{R}^{n \times p}$ has rank p, that is, it is full column rank.

(c) Let $X = U \Gamma V^T$ be the singular value decomposition (SVD) of X, where $U \in \mathbb{R}^{n \times p}$ satisfies $U^T U = I_p$ and $\Gamma = \text{diag}(\gamma_1, \ldots, \gamma_p)$ is the diagonal matrix of singular values. Using this SVD, give as explicit a formula as you can for the derivative matrix

$$\dot{H}_\lambda := \frac{\partial}{\partial \lambda} H_\lambda \in \mathbb{R}^{n \times n}.$$

(d) Let $r(\lambda) = n \cdot R_{\text{in}}(\widehat{f}_\lambda)$ be the in-sample risk as a function of $\lambda \geq 0$. Give a formula for the derivative $r'(\lambda) = \frac{\partial}{\partial \lambda} r(\lambda)$.

(e) Using your preceding two answers, show that $r'(0) < 0$, that is, there is *always* some $\lambda > 0$ so that the in-sample risk of the ridge estimator is smaller than unregularized least squares.

Question 4.2 (Limiting ridge solutions, **10 points**): Let $\widehat{\beta}_\lambda = \operatorname{argmin}_b \{\|Xb - y\|_2^2 + \lambda \|b\|_2^2\}$ be the ridge regression estimator. Let $X \in \mathbb{R}^{n \times p}$ and assume $p > n$, where X has rank n. Using the SVD of X, give a closed form for $\lim_{\lambda \downarrow 0} \widehat{\beta}_\lambda$.

Question 4.3 (Linear regression versus k-nearest neighbors, **30 points**): You compare k-nearest neighbors (knn) and linear regression in terms of their classification performance in the presence of increasing numbers of noise variables. The setup is as follows, and mimics the mixture simulation in class. The 20×2 data matrix `mixturemeans.csv` is available on the course website; the first 10 rows are for class 1, the next 10 for class 2. Let $M_1 = [\mu_1 \ \cdots \ \mu_{10}]^T \in \mathbb{R}^{10\times 2}$ and $M_2 = [\mu_{11} \ \cdots \ \mu_{20}]^T \in \mathbb{R}^{10\times 2}$ be these matrices of means, where $\mu_i \in \mathbb{R}^2$.

(a) Write a function to generate a sample of N points from a uniform mixture of Gaussians in $\mathbb{R}^2$, with each Gaussian $\mathsf{N}(\mu_i, \sigma^2 I)$ having diagonal covariance $\sigma^2 I$ for a fixed $\sigma^2 > 0$. The function takes as inputs the centroid matrix M, sample size N and σ, and outputs a matrix $X \in \mathbb{R}^{N\times p}$ whose rows are i.i.d. draws from this mixture of Gaussians.

(b) Use your function to generate a dataset of size $N_{\text{train}} = 100$ for each of the two classes, with $\sigma^2 = \frac{1}{5}$, as well as a test set of size $N_{\text{test}} = 10^4$ for each class. Create the corresponding response vectors for each. This should leave you with matrices $X_{\text{train}} \in \mathbb{R}^{2N\times 2}, X_{\text{test}} \in \mathbb{R}^{2N_{\text{test}}\times 2}$ and responses y_{train} and y_{test}.

(c) What is the Bayes (optimal) classifier for this problem? Write this in terms of the densities $f_i, i = 1, \ldots, 20$ for each of the mixture components.

(d) Write a function to compute the Bayes classifier for this setup. It should take as input the two matrices M_0, M_1 of means, variance σ^2, and an input matrix X to be classified. Your function should classifiy all the rows of X.

(e) Write an evaluation function that takes as input your training data, test data, and a vector of values for k, the knn neighborhood size parameter. Your function should

i. Estimate the Bayes error using the test data using your function from part (d).

ii. Estimate the test error of a linear classifier fit by least squares.

iii. Estimate the test errors for knn at each of the values of k (in `R`, the package `class` has a `knn` function).

Run your function using $k = 1, 3, 5, 7, 9, 11, 13, 15$.

(f) Write a new function that expands the evaluation function in the part (e) to take two extra parameters: the number noise of noise variables and a variance τ^2_{noise}. This function adds additional Gaussian noise columns to X_{train} and X_{test}, where the noise columns have i.i.d. $\mathsf{N}(0, \tau^2_{\text{noise}})$ entries. This function should produce the same outputs as that in part (e). Run your function with $p_{\text{noise}} = 1, 2, \ldots, 10$ noise variables with $\tau^2_{\text{noise}} = 1$. Summarize its outputs.

Question 4.4 (Causal estimation, **15 points**): Consider the potential outcomes framework for a real-valued response Y with randomized treatment assignments $W \in \{0, 1\}$, so that $(Y(0), Y(1)) \perp W$. Let

$$\tau^\star := \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)]$$

be the average treatment effect, which is the variable of interest. Assume there are covariates $X \in \mathbb{R}^d$, which may (or may not) be related to the responses Y, but where W is also independent of X. Let the population mean-square-error estimates be

$$(\tau_{\text{mse}}, \alpha_{\text{mse}}, \beta_{\text{mse}}) = \underset{\tau,\alpha,\beta}{\operatorname{argmin}} \, \mathbb{E}\left[(Y - \alpha - X^T\beta - \tau W)^2\right],$$

so that $\alpha_{\text{mse}} \in \mathbb{R}$ is an intercept, $\beta_{\text{mse}} \in \mathbb{R}^d$, and τ_{mse} is the coefficient of W in the model $Y_i = \alpha + X_i^T\beta + \tau W_i + \varepsilon_i$. Show that

$$\tau_{\text{mse}} = \tau^\star.$$

5 Stochastic optimization

Question 5.1 (Computing an update beyond stochastic gradient descent, **15 points**)**:** The standard stochastic gradient method iteratively makes a linear approximation to a given loss, then minimizes it plus a quadratic regularization:

$$\theta_\alpha := \operatorname*{argmin}_\theta \left\{ \ell(\theta_0) + g^T(\theta - \theta_0) + \frac{1}{2\alpha} \|\theta - \theta_0\|^2 \right\},$$

where $g \in \partial \ell(\theta_0)$, and which has the trivial solution $\theta_\alpha = \theta - \alpha g$. In this problem, we consider instead solving an update that applies when the loss ℓ is nonnegative, so we model it by a *nonnegative* function:

$$\theta_\alpha := \operatorname*{argmin}_\theta \left\{ \left(\ell(\theta_0) + g^T(\theta - \theta_0)\right)_+ + \frac{1}{2\alpha} \|\theta - \theta_0\|^2 \right\}. \tag{5.1}$$

See the papers [4, 3] for more about such more sophisticated updates.

(a) Let $b > 0$ and $a \in \mathbb{R}$. Give the minimizer in $t \in \mathbb{R}$ of

$$(a - bt)_+ + \frac{\lambda}{2} t^2.$$

(b) Demonstrate that θ_α as defined in (5.1) satisfies

$$\theta_\alpha = \theta_0 - tg$$

for some $t \geq 0$.

(c) Use your answers to the previous parts to show that (recall $\ell(\theta_0) \geq 0$)

$$\theta_\alpha = \theta_0 - \min \left\{ \frac{\ell(\theta_0)}{\|g\|^2}, \alpha \right\} g.$$

Question 5.2 (AdaGrad and Sparse Gradients (**25 points**))**:** In this question, you will implement and experiment with AdaGrad [6] on the Reuters RCV1 dataset, which investigates classifying news articles (from the Reuters news service) as one of many different subjects. We will focus on a binary version of this. The data in the Reuters corpus is represented as $n = 804414$ (scaled) word vectors, where each data vector $x \in \mathbb{R}^d$ has $d = 47236$ entries, one corresonding to each of d distinct words, where $x_j = 0$ if word j did not appear in the document corresponding to x. The data has multiple labels, where each (raw) target $y_i \in \{0, 1\}^m$ consists of an $m = 103$-vector with indices 1 for each topic the article covers.

(i) Download and store the Reuters RCV1 dataset. Construct a vector $y \in \{\pm 1\}^n$ where entry $y_i = 1$ if the document i contains the `CCAT` topic (commercial category of articles), and $y_i = -1$ otherwise. We have provided Python code that does this in `load_rcv1.py`, where the vector `y_ccat` contains the labels for the `CCAT` topic.

(ii) Implement the stochastic gradient and AdaGrad methods to compare them on binary logistic regression, using the loss

$$\ell(\theta; x, y) = \log(1 + \exp(-yx^T\theta)).$$

Your method should be able to accept an initial stepsize and a (potentially fractional) number of passes through the dataset. Each iteration of the method should consist of picking a random

example (chosen uniformly at random) from the data, then performing a single subgradient step on that example.

We have provided python starter code in the files `adagrad_starter.py` and `loss_function_starter.py`. This includes a basic implementation of the stochastic gradient method (`BaseSGDOptimizer`) and subclasses. The `BaseSGDOptimizer` class is initialized with an instance of a `LossFunction`. You might consider implementing subclasses of `LossFunction` to capture loss functions of interest.

(iii) For various values of the initial stepsize α_0, perform a single pass over the entire RCV1 dataset with the basic stochastic gradient method with stepsize $\alpha_t = \frac{\alpha_0}{\sqrt{t}}$ and AdaGrad.

a. For SGD, you should use initial stepsizes $\alpha_0 \in \{2^k \text{ for } k = 2, 3, \ldots, 9, 10\}$

b. For AdaGrad, you should use initial stepsize $\alpha_0 \in \{2^k \text{ for } k = -5, -4, \ldots, 2, 3\}$.

Give a plot of the average loss for the final fit parameter for each initial stepsize, aligning the final results for each (i.e., the horizontal axis should be logarithmically spaced with $\{4, 8, 16, \ldots\}$ for SGD and $\{2^{-5}, 2^{-4}, \ldots\}$ for AdaGrad). We recommend performing this experiment 5–10 times per initial stepsize so that you can obtain average results, though there should be little variance in the outputs. What do you observe?

Your submission should include both the figure and any new code you have written.

Question 5.3 (5 points): Let $H \in \mathbb{R}^{d \times d}$ be a positive definite matrix, $\ell : \mathbb{R}^d \to \mathbb{R}_+$ be a nonnegative loss function, and $g \in \partial \ell(\theta_0)$ be an element of the subdifferential of ℓ at θ_0. Let $\alpha > 0$. Give

$$\theta_\alpha := \operatorname*{argmin}_\theta \left\{ \left(\ell(\theta_0) + g^\top (\theta - \theta_0) \right)_+ + \frac{1}{2\alpha} (\theta - \theta_0)^\top H (\theta - \theta_0) \right\}.$$

Hint. See Question 5.1.

Question 5.4 (Implementing truncated gradient methods **25 points**)**:** In this question, you will implement two versions of truncated stochastic gradient methods. We wish to solve the stochastic optimization problem

$$\operatorname*{minimize}_\theta \; L(\theta) := \mathbb{E}_P[\ell(\theta, Z)] = \int \ell(\theta, z) dP(z),$$

where Z is a random variable and ℓ is nonnegative (with minimal value $0 = \inf_\theta \ell(\theta, z)$ for all z). For these methods, we iteratively draw $Z_k \sim P$, i.i.d., set $g_k \in \partial \ell(\theta_k, Z_k)$, then update

$$\theta_{k+1} = \operatorname*{argmin}_\theta \left\{ \left(\ell(\theta_k) + \langle g_k, \theta - \theta_k \rangle \right)_+ + \frac{1}{2\alpha_k} (\theta - \theta_k)^\top H_k (\theta - \theta_k) \right\}, \tag{T-UP}$$

where $H_k \succeq 0$ is a diagonal matrix.

i. For the truncated stochastic gradient method, $H_k = I$ is the identity matrix.

ii. For truncated AdaGrad, $H_k = \operatorname{diag}(\sum_{i=1}^k g_i g_i^\top)^{1/2} + \epsilon I$ has jth diagonal entry equal to the square root of the sum of squared jth gradient component including iteration k, where $\epsilon \geq 0$ (typically, $\epsilon \approx 10^{-10}$) avoids `NaN`s.

The file `beyond_gd_starter.py` provides starter code for implementing the update (T-UP) in `PyTorch`, with code you should fill in marked `Your code here` in the two classes `TruncatedSGD` and `TruncatedAdagrad`. These classes subclass the `PyTorch` optimizer class, meaning that they need only implement the `step` method. We provide wrapper code in the file `mnist_experiments.py` to load and fit either a small multi-layer perceptron or a small convolutional neural network on the MNIST digit recognition dataset. This starter code will automatically leverage GPUs if you have them,[2] though this only makes a difference for convolutional networks.

(a) Implement the `step` update for `TruncatedSGD`.

(b) Implement the `step` update for `TruncatedAdagrad`. Be careful about exactly which square roots you take in implementing the updates (T-UP).

(c) Using the method `FitNN` with the default learning rate (i.e., stepsize), run the following 8 experiments: fit a multi-layer perceptron or a convolutional network with the optimization methods Adam, Adagrad, TruncatedSGD, and TruncatedAdagrad. Include print-outs of your results.

For reference, on a 2021 MacBook Pro, our code fits the MLP in roughly 2–4 seconds per epoch, for a total of approximately 20 seconds; the ConvNet architecture requires approximately 15 seconds per epoch using the GPUs available through `MPS`, for a total of approximately 90 seconds. It is twice as slow using only the CPU. You should achieve roughly 99% test accuracy using the Convolutional network and any of Adam, AdaGrad, and TruncatedAdagrad.

Question 5.5 (25 points): You are the technical lead of a team at a company employing machine learning, and one of your colleagues comes to you and claims that they have developed a new optimization method (say, the truncated AdaGrad method (T-UP) with $H_k = \operatorname{diag}(\sum_{i \le k} g_i g_i^\top)^{1/2})$ that "outperforms" other models. How might you evaluate this claim? Give a "procedure" (broadly construed) to help elucidate ways in which one method might be preferred to others. This question is deliberately open ended. To obtain full credit, you should

i. Describe (precisely) what it is you are comparing, and have a criterion for making the decision

ii. Have something like a null hypothesis, a set of hypotheses between which you would like to decide, or a set of expected results that your experiments might contradict

iii. Describe a data gathering procedure with which you can evaluate your hypothesis or hypotheses, and give (asymptotically) valid p-values, confidence intervals/sets, or otherwise.

iv. Implement your procedure on the truncated AdaGrad method to decide whether we ought to use it over other methods.

As an example, Asi and Duchi [3, 4] argue that truncated updates (T-UP) and related methods are more robust to initial stepsize choices than classical optimization methods. They also argue that in problems where achieving zero error is possible, such stepping may be preferred. How might you rigorously test these claims? (It's fine to use simulated or real data here.) Often, in optimization, one more or less simply points at loss plots over time and shrugs. Give something more substantial than that.

[2] If you are running experiments on a Mac, you will need to use the nightly build of `PyTorch` to have access to Apple's `MPS`.

6 Parameter Inference

Question 6.1 (Resampling methods and normal inference, **25 points**): In this problem, we compare three parameter inference methods: (i) model-based inference, which assumes the model is true, (ii) the "sandwich estimator" we derived in class, and (iii) the bootstrap resampling estimator of variance. We will do this both for linear and (binary) logistic regression, repeating the following experimental protocol many times and providing summary results. We first describe the protocol for the Gaussian linear model case; we then describe modifications for the other cases.

i. For the data model
$$y_i = x_i^T \beta^\star + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{iid}}{\sim} \mathsf{N}(0, \sigma^2),$$
generate a sample of size n (to be specified) in dimension $d = 10$, where $x_i \stackrel{\text{iid}}{\sim} \mathsf{N}(0, I_d)$ and $\sigma^2 = 1$, and draw $\beta^\star \sim \mathsf{Uni}(\mathbb{S}^{d-1})$, the sphere in $\mathbb{R}^d$.

ii. Compute $\widehat{\beta}_n$ minimizing
$$L_n(\beta) = \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \beta)^2.$$

iii. Consider the following three covariance estimates:
$$\Sigma_n := \widehat{\sigma}^2 (X^T X)^{-1}, \quad \widehat{\sigma}^2 = \frac{1}{n-d} \sum_{i=1}^n (y_i - x_i^T \widehat{\beta}_n)^2 \tag{Fisher}$$
$$\Sigma_n := (X^T X)^{-1} \bigg(\sum_{i=1}^n (y_i - x_i^T \widehat{\beta}_n)^2 x_i x_i^T \bigg) (X^T X)^{-1} \tag{Sandwich}$$
$$\Sigma_n := \frac{1}{B} \sum_{b=1}^B (\widehat{\beta}_n^b - \widehat{\beta}_n)(\widehat{\beta}_n^b - \widehat{\beta}_n)^T \tag{Bootstrap}$$
where $\widehat{\beta}_n^b$ is a bootstrap resampled least-squares estimate, and $B = 200$ is the number of bootstrap replicates.

The first covariance is the classical Fisher information matrix, the second the covariance as per the standard asymptotic theory we have developed, and the third that of the boostrap. We have $\widehat{\beta}_n \stackrel{\cdot}{\sim} \mathsf{N}(\beta^\star, \Sigma_n)$ for each of these (so long as the data model remains true!), and therefore
$$\mathcal{C}_n := \left\{ \beta \in \mathbb{R}^d \mid (\beta - \widehat{\beta}_n)^T \Sigma_n^{-1} (\beta - \widehat{\beta}_n) \le \chi^2_{d,1-\alpha} \right\}$$
as an asymptotically valid $1 - \alpha$ confidence set, that is, $\mathbb{P}(\beta^\star \in \mathcal{C}_n) \to 1 - \alpha$, where $\chi^2_{d,1-\alpha}$ is the $1 - \alpha$ quantile of a χ^2 random variable with d degrees of freedom. Use $\alpha = .1$ for the remainder.

(a) Repeat the experiment i–iii for $n = 50, 100, 200, 400$ for $T = 200$ times, and track the fraction of times that $\beta^\star \in \mathcal{C}_n$ for each of the covariances (Fisher), (Sandwich), and (Bootstrap). For each covariance approximation, plot your coverage against sample size n.

(b) Repeat part (a) except for logistic regression. Thus, make the following changes to the procedure i–iii. Instead of the linear regression model, generate data from the logistic regression model
$$\mathbb{P}_\beta(Y = y \mid X = x) = \frac{e^{y \beta^T x}}{1 + e^{x^T \beta}}, \quad y \in \{0, 1\},$$

where $x_i, \beta^\star$ are generated identically. In part ii, choose $\widehat{\beta}_n$ to minimize the negative log likelihood, that is, for $\ell(\beta, x, y) = -\log p_\beta(y \mid x) = \log(1 + e^{x^T\beta}) - yx^T\beta$, let $L_n(\beta) = \frac{1}{n}\sum_{i=1}^n \ell(\beta, x_i, y_i)$. In part iii, replace the Fisher information and Sandwich covariances with their counterparts

$$\Sigma_n := (X^T W X)^{-1}, \quad W = \mathrm{diag}([\widehat{p}_i(1-\widehat{p}_i)]_{i=1}^n)$$
$$\Sigma_n := \frac{1}{n}\nabla^2 L_n(\widehat{\beta}_n)^{-1}\widehat{\mathrm{Cov}}(\nabla\ell(\widehat{\beta}_n(X,Y)))\nabla^2 L_n(\widehat{\beta}_n)^{-1},$$

respectively (the bootstrap covariance does not change).

(c) Repeat part (a) with faulty linear modeling assumptions, so that we evaluate coverage of $\widehat{\beta}_n$ and $\mathcal{C}_n$ for the best linear predictor in mean-squared error, $\beta_{\mathrm{mse}} = \mathrm{argmin}_\beta \mathbb{E}[(y - x^T\beta)^2]$. To do so, replace the model in part i with

$$y_i = x_i^T\beta^\star + (x_i^T\theta^\star)^2 + \varepsilon_i, \quad \varepsilon_i \stackrel{\mathrm{iid}}{\sim} \mathsf{N}(0, \sigma^2),$$

where $\theta^\star \sim \mathsf{Uni}(\mathbb{S}^{d-1})$ as well. Note that

$$\begin{aligned}\mathbb{E}[(y - x^T\beta)^2] &= \mathbb{E}[(\varepsilon + x^T(\beta - \beta^\star) + (x^T\theta^\star)^2)^2] \\ &= \sigma^2 + \|\beta - \beta^\star\|_2^2 + \mathbb{E}[(x^T\theta^\star)^4],\end{aligned}$$

because $\mathbb{E}[v^T x(u^T x)^2] = 0$ for any vectors u, v,[3] so $\beta_{\mathrm{mse}} = \mathrm{argmin}_\beta \mathbb{E}[(y - x^T\beta)^2] = \beta^\star$ as well.

Question 6.2 (Asymptotics of causal inference, **10 points**): As in Question 4.4, consider the potential outcomes framework for a real-valued response Y with randomized treatment assignments $W \in \{0, 1\}$, so that $(Y(0), Y(1)) \perp W$. Let the population mean-square-error estimates be

$$(\tau_{\mathrm{mse}}, \alpha_{\mathrm{mse}}, \beta_{\mathrm{mse}}) = \operatorname*{argmin}_{\tau,\alpha,\beta} \mathbb{E}\left[(Y - \alpha - X^T\beta - \tau W)^2\right],$$

so that $\alpha_{\mathrm{mse}} \in \mathbb{R}$ is an intercept, $\beta_{\mathrm{mse}} \in \mathbb{R}^p$, and τ_{mse} is the coefficient of W in the model $Y_i = \alpha + X_i^T\beta + \tau W_i + \varepsilon_i$. Given a sample of size n, where each individual is chosen to be in treatment ($W = 1$) or control ($W = 0$) independently of (X, Y) with $\mathbb{P}(W = 1) = p \in (0, 1)$, let $\widehat{\tau}_n, \widehat{\alpha}_n, \widehat{\beta}_n$ be the empirical squared error minimizers. Give the limiting distribution of $\widehat{\tau}_n$. That is, give the value of the asymptotic variance $\sigma^2(\tau)$ in the limiting normal

$$\sqrt{n}(\widehat{\tau}_n - \tau_{\mathrm{mse}}) \stackrel{d}{\to} \mathsf{N}\left(0, \sigma^2(\tau)\right).$$

Hint. Reparameterize the problem slightly: let $\mu_X = \mathbb{E}[X]$ be the mean of X and $\widetilde{\alpha} = \alpha - \mu_X^T\beta - \tau p$. Then use that if $T \sim \mathsf{N}(0, \Sigma)$, then $v^T T \sim \mathsf{N}(0, v^T\Sigma v)$.

Question 6.3 (**10 points**): Consider the setting of Question 6.2.

(a) Let $\mathbb{P}(W = 1) = p$ be the treatment probability in a randomized treatment assignment. Show that for the treatment/control counts $N(w) = |\{i \in [n] \mid W_i = w\}|$, $w = 0, 1$, the naive estimator

$$\widehat{\tau}_{\mathrm{naive}} := \frac{1}{N(1)}\sum_{i=1}^n W_i Y_i - \frac{1}{N(0)}\sum_{i=1}^n (1 - W_i)Y_i$$

satisfies

$$\sqrt{n}\left(\widehat{\tau}_{\mathrm{naive}} - \tau_{\mathrm{mse}}\right) \dot{\sim} \mathsf{N}(0, \sigma^2_{\mathrm{naive}}) \quad \text{for } \sigma^2_{\mathrm{naive}} := \frac{1}{p}\mathrm{Var}(Y(1)) + \frac{1}{1-p}\mathrm{Var}(Y(0)).$$

[3] We have $\mathbb{E}[v^T x(u^T x)^2] = \sum_{i=1}^d v_i \mathbb{E}[x_i(u^T x)^2]$, and (w.l.o.g. taking $i = 1$) we observe $\mathbb{E}[x_1(u^T x)^2] = \sum_{i,j=1}^d u_i u_j \mathbb{E}[x_1 x_i x_j]$. Then note that $\mathbb{E}[x_1 x_i x_j] = 0$ for any coordinates i, j when $x \sim \mathsf{N}(0, I)$.

(b) Show (perhaps using a reparameterization as in the hint for Q. 6.2) that for $\mu_X = \mathbb{E}[X]$, the population minimizers β_{mse} satisfies

$$\beta_{\text{mse}} = \operatorname*{argmin}_{\beta} \mathbb{E}\left[(Y(W) - \mathbb{E}[Y(W) \mid W] - (X - \mu_X)^T\beta)^2\right]$$

(c) Now assume the treatment probability $p = \frac{1}{2}$. Show that for the variance $\sigma^2(\tau)$ from Question 6.2,

$$\sigma^2(\tau) \le \sigma^2_{\text{naive}}.$$

When is the inequality strict? What does this mean for a randomized trial?

Question 6.4 (Asymptotics of PPI and causal inference)**:** Let the setting of Questions 4.4 and 6.2 hold, but consider the prediction-powered inference (PPI) setting [2, 1]. We assume we have a very large potential treatment population of size $N + n$, of which we sample n individuals to participate in an RCT. Mathematizing this, we have

i. A labeled sample $(X_i, W_i, Y_i(W_i))_{i=1}^n$ of treated and control individuals (assume $W_i \overset{\text{iid}}{\sim} \mathsf{Uni}\{0,1\}$)

ii. A large unlabeled sample $(\widetilde{X}_i)_{i=1}^N$ of individuals

iii. That the X_i and $\widetilde{X}_i$ are i.i.d.

iv. A black box prediction function $f : \mathbb{R}^d \to \mathbb{R}$, which predicts outcomes Y_i for individuals.

Let P_n and $\widetilde{P}_N$ denote the empirical distributions of the observations, respectively. Let $\widetilde{W}_i$ also be i.i.d. $\mathsf{Uni}\{0,1\}$, but note that these are "fake" treatments—individuals from the study population whom we would have assigned to treatment had we included them in the labeled/measured sample. Define the objectives

$$L_n(\alpha, \tau, \beta) = \mathbb{E}_{P_n}\left[(Y - \alpha - W\tau - X^\top\beta)^2\right], \quad L^f_N(\alpha, \tau, \beta) = \mathbb{E}_{\widetilde{P}_N}\left[(f(\widetilde{X}) - \alpha - \widetilde{W}\tau - \widetilde{X}^\top\beta)^2\right]$$
$$L^f_n(\alpha, \tau, \beta) = \mathbb{E}_{P_n}\left[(f(X) - \alpha - W\tau - X^\top\beta)^2\right].$$

Let

$$(\alpha_{\text{mse}}, \tau_{\text{mse}}, \beta_{\text{mse}}) = \operatorname*{argmin}_{\alpha,\tau,\beta} \mathbb{E}[L_n(\alpha, \tau, \beta)]$$

be the population minimizers. Now consider the PPI++ estimator of the average treatment effect

$$\widehat{\tau}_\lambda = \operatorname*{argmin}_{\tau} \inf_{\alpha,\beta} \left\{ L_n(\alpha, \tau, \beta) + \lambda\left(L^f_N(\alpha, \tau, \beta) - L^f_n(\alpha, \tau, \beta)\right)\right\}.$$

(a) Assume the sample sizes satisfy $N/n \to r \in (1, \infty)$. Give the asymptotic distribution of $\widehat{\tau}_\lambda$. *Hint.* Reparameterize as in Q. 6.2, and write your answer in terms of the variance of W and $\varepsilon := Y(W) - \alpha_{\text{mse}} - \tau_{\text{mse}}(W - \mathbb{E}[W]) - (X - \mu_X)^\top\beta_{\text{mse}}$ and $\varepsilon^f := f(X) - \alpha_{\text{mse}} - \tau_{\text{mse}}(W - \mathbb{E}[W]) - \beta^\top_{\text{mse}}(X - \mu_X)$.

(b) Now assume the "black box" f is correct in that given $X = x$, for treatment $W = w$,

$$Y(W) = f(X) + \tau_{\text{mse}}(W - \mathbb{E}[W]) + \xi, \quad \text{where } \xi \overset{\text{iid}}{\sim} (0, \sigma^2)$$

is independent of X, W, and everything else. Assume also that $\mathbb{P}(W = 0) = \mathbb{P}(W = 1) = \frac{1}{2}$. Define $h(X) = f(X) - (X - \mu_X)^\top\beta_{\text{mse}} - \alpha_{\text{mse}}$ to be the "nonlinear" components of f. Show that

$$\sqrt{n}\left(\widehat{\tau}_\lambda - \tau_{\text{mse}}\right) \overset{\cdot}{\sim} \mathsf{N}\left(0, \frac{\text{Var}(\xi)}{\text{Var}(W)} + \frac{(1-\lambda)^2\text{Var}(h(X))}{\text{Var}(W)} + \frac{\lambda^2}{r}\frac{\text{Var}(\xi + h(X))}{\text{Var}(W)}\right).$$

(c) The choice $\lambda = 0$ corresponds to the "classical" treatment effect estimate. Show that if the model $Y(w) = \alpha + \tau w + \beta^\top x + \xi$ is well-specified (i.e., the linear model and treatment effect is correct), then PPI++ has no effect and the minimal variance of $\widehat{\tau}_\lambda$ is attained at $\lambda = 0$.

(d) When does PPI++ (i.e., choosing λ to minimize the asymptotic variance of $\widehat{\tau}_\lambda$) lead to an improvement over $\lambda = 0$? How much improvement is possible when $r = \infty$ (i.e., $N/n \to \infty$)? (You may assume the limiting variance is still sensible.)

7 Predictive Inference

Question 7.1 (Constructions of conformal confidence sets, **15 points**): Suppose we have set-valued mappings $C_\tau : \mathcal{X} \rightrightarrows \mathcal{Y}$, meaning that $C_\tau(x) \subset \mathcal{Y}$, indexed by $\tau \in \mathbb{R}_+$, where

$$C_\tau(x) \subset C_{\tau+\delta}(x)$$

for all $\delta \geq 0$, where $\lim_{\tau\to\infty} C_\tau(x) = \mathcal{Y}$ (that is, for large enough τ the confidence set $C_\tau(x)$ includes all of $\mathcal{Y}$). Define

$$s(x, y) := \inf \{\tau \in \mathbb{R} \mid y \in C_\tau(x)\}. \tag{7.1}$$

You are given a sample $(X_i, Y_i)_{i=1}^n \stackrel{\text{iid}}{\sim} P$ of size n and define $S_i = s(X_i, Y_i)$ for each i, then set

$$\widehat{\tau}_n := \text{the } (1 + 1/n)(1 - \alpha) \text{ quantile of } \{S_i\}_{i=1}^n.$$

Let $\widehat{C} = C_{\widehat{\tau}_n}$ be the associated confidence set.

(a) Using the results from class, show that $\widehat{C}$ is a valid $(1 - \alpha)$ prediction set, that is, on a new example (X_{n+1}, Y_{n+1}) from P,

$$P(Y_{n+1} \in \widehat{C}(X_{n+1})) \geq 1 - \alpha.$$

We now explore different constructions of such confidence sets. Each of these will leverage an already constructed predictor f taking inputs in $\mathcal{X}$.

(b) Let ℓ be a loss function and $\ell(f(x), y)$ be the loss for predicting $f(x)$ on response y. Set

$$C_\tau(x) = \{y \in \mathcal{Y} \mid \ell(f(x), y) \leq \tau\}.$$

Give the value $s(x, y)$ the definition (7.1) yields.

(c) For binary logistic regression, $f : \mathcal{X} \to \mathbb{R}$, $y \in \{\pm 1\}$, and $\ell(f(x), y) = \log(1 + e^{-yf(x)})$. Give the value $s(x, y)$ the definition (7.1) yields for the confidence set in part (b). For a given threshold τ, when is $C_\tau(x)$ a singleton?

(d) For k-class logistic regression, $f : \mathcal{X} \to \mathbb{R}^k$, $y \in \{1, \ldots, k\}$, and $\ell(f(x), y) = \log(1 + \sum_{l=1}^k e^{f_l(x) - f_y(x)})$. Give the value $s(x, y)$ the definition (7.1) yields for the confidence set in part (b). For a given threshold τ, when is $C_\tau(x)$ a singleton?

(e) Let $\mathcal{Y} = \mathbb{R}$ (so we have real-valued responses as in regression), and let $l, u : \mathcal{X} \to \mathbb{R}$ model lower and upper quantiles of Y given X, respectively. (That is, we wish to have $Y \in [l(x), u(x)]$ with a given probability.) Let

$$C_\tau(x) = [l(x) - \tau, u(x) + \tau]$$

where $C_\tau(x) = \emptyset$ if $l(x) - \tau > u(x) + \tau$, i.e., the lower end of the interval is greater than the upper. Give the value $s(x, y)$ the definition (7.1) yields for this confidence set.

Question 7.2 (Conformal sets, **20 points**): Consider the following heteroskedastic linear model:

$$y = x^T \beta^\star + \|x\|_2 \varepsilon, \quad \varepsilon \stackrel{\text{iid}}{\sim} \mathsf{N}(0, 1), \quad x \stackrel{\text{iid}}{\sim} \begin{cases} \mathsf{N}(e_1, I_d) & \text{w.p. } \frac{1}{2} \\ \mathsf{N}(-e_1, 3I_d) & \text{w.p. } \frac{1}{2} \end{cases} \tag{7.2}$$

where e_1 is the first standard basis vector. Given a sample of size n from this model, let $X \in \mathbb{R}^{n \times d}$ and $Y \in \mathbb{R}^n$ be the usual design matrix and responses. Consider fitting the following two models to this data: first, the standard linear regression model

$$\widehat{\beta}^{\text{mse}} := \operatorname*{argmin}_{\beta} \|X\beta - Y\|_2^2 .$$

This gives predictor $\widehat{f}(x) = x^T \widehat{\beta}^{\text{mse}}$. Second, a quantile model, which attempts to predict the lower and upper α quantiles of the responses y_i. To do this, define the feature mapping $\phi(x) = (x, \|x\|_2) \in \mathbb{R}^{d+1}$, the quantile loss function

$$\ell_\alpha(t, y) := \alpha \, (y - t)_+ + (1 - \alpha) \, (t - y)_+ ,$$

and then find the $(d+1)$-dimensional vectors $\widehat{\theta}_\alpha$ and $\widehat{\theta}_{1-\alpha}$ solving

$$\widehat{\theta}_\alpha = \operatorname*{argmin}_{\theta} \sum_{i=1}^n \ell_\alpha(\theta^T \phi(x_i), y_i) \quad \text{and} \quad \widehat{\theta}_{1-\alpha} = \operatorname*{argmin}_{\theta} \sum_{i=1}^n \ell_{1-\alpha}(\theta^T \phi(x_i), y_i).$$

This gives lower and upper predictors $\widehat{l}(x) = \phi(x)^T \widehat{\theta}_\alpha$ and $\widehat{u}(x) = \phi(x)^T \widehat{\theta}_{1-\alpha}$.

(a) What is the population counterpart of $\widehat{\beta}^{\text{mse}}$? That is, give $\beta^\star = \operatorname{argmin}_\beta \mathbb{E}[(y - x^T\beta)^2]$.

(b) What are the population counterparts of $\widehat{\theta}_\alpha$ and $\widehat{\theta}_{1-\alpha}$? That is, give

$$\theta^\star_\alpha = \operatorname*{argmin}_{\theta} \mathbb{E}[\ell_\alpha(\theta^T \phi(x), y)] \quad \text{and} \quad \theta^\star_{1-\alpha} = \operatorname*{argmin}_{\theta} \mathbb{E}[\ell_{1-\alpha}(\theta^T \phi(x), y)].$$

(c) Generate three datasets from the model (7.2), each in dimension $d = 5$ with sample size $n = 400$: a training set, a validation set, and a test set, where $\beta^\star \sim \mathsf{Uni}(\mathbb{S}^{d-1})$. Now, fit the linear predictor $\widehat{f}$ and lower/upper predictors $\widehat{l}, \widehat{u}$ on the training data. Consider the confidence sets

$$\widehat{C}^{\text{mse}}_\tau(x) := \left[\widehat{f}(x) - \tau, \widehat{f}(x) + \tau\right] \quad \text{and} \quad \widehat{C}^{\text{q}}_\tau := \left[\widehat{l}(x) - \tau, \widehat{u}(x) + \tau\right].$$

Using the validation data, use conformal inference to choose τ so that $\mathbb{P}(Y^* \in \widehat{C}_\tau(X^*)) \geq 1 - 2\alpha$ for each of these confidence sets, where $\alpha = .025$. Repeat this experiment $T = 100$ times and give the (empirical) coverage you obtain on the test set for each method.

(d) As in part (c) (with $T = 100$ experiments), give average the empirical coverage on the following two subsets of the test set:

$$S_{\text{left}} := \{i \mid e_1^T x_i \leq 0\} \quad \text{and} \quad S_{\text{right}} := \{i \mid e_1^T x_i > 0\}.$$

Explain your result in a few words.

Question 7.3 (Conditional validity on CIFAR-100, **40 points**): The CIFAR-100 dataset is a dataset consisting of small images from 100 distinct classes. In this question, you will compare "static" split-conformal methods for predictive inference to "conditional" methods. Assuming you have a predictor $f : \mathcal{X} \to \mathbb{R}^k$, which assigns a score $f_y(x)$ to each class $y \in \{1, \ldots, k\}$ (where $k = 100$ in this case), the static conformal methodology constructs a confidence set of the form

$$\widehat{C}(x) = \{y \in [k] \mid f_y(x) \geq \widehat{\tau}\} ,$$

where $\widehat{\tau}$ is a threshold, so that $\widehat{C}(x)$ contains classes assigned a high-enough score by the predictive model f. The conditional type method uses

$$\widehat{C}(x) = \left\{ y \in [k] \mid f_y(x) \geq \phi(x)^T \widehat{\theta} \right\},$$

where $\widehat{\theta}$ is a fit vector and $\phi : \mathcal{X} \to \mathbb{R}^d$ is a feature function. For each, we assume existence of a validation dataset $Z_{\text{val}} = \{(x_i, y_i)\}_{i=1}^{n_{\text{val}}}$, and for the loss $\ell_\alpha(t) = \alpha\,(t)_+ + (1-\alpha)\,(-t)_+$, choose $\widehat{\tau}$ and $\widehat{\theta}$, respectively, by fitting

$$\widehat{\tau} = \operatorname*{argmin}_\tau \frac{1}{n_{\text{val}}} \sum_{x,y \in Z_{\text{val}}} \ell_\alpha(f_y(x) - \tau) \quad \text{and} \quad \widehat{\theta} = \operatorname*{argmin}_\theta \frac{1}{n_{\text{val}}} \sum_{x,y \in Z_{\text{val}}} \ell_\alpha(f_y(x) - \theta^T \phi(x)).$$

Note that these correspond to using the scoring function $s(x, y) = -f_y(x)$ and prediction set

$$C(x) = \{ y \in [k] \mid s(x, y) \leq \tau(x) \}$$

for a threshold $\tau(x)$ in the "standard" conformal prediction setup. In this question, for covariates $x \in \mathbb{R}^p$, we will use random feature functions of the form

$$\phi(x) = Wx, \quad \text{where } W \in \mathbb{R}^{d \times p}, \quad W_{ij} \stackrel{\text{iid}}{\sim} \mathsf{N}(0, 1)$$

to get a sense of coverage.

In the data files `cifar100_train_features.npy` and `cifar100_train_labels.npy` (and the similarly named `test` files), we have NumPy matrices of the CIFAR-100 train and test datasets processed through a 50 layer Residual network [9, 8], yielding data vectors $x \in \mathbb{R}^p$ with $p = 2048$. We have also included a file `cifar_processing.py`, which provides methods `load_numpy_into_data` and `split_into_train_and_validation` to help with data processing. Perform the following experiment with this data:

i. Split the non-test data (randomly) into a training dataset of $4 \cdot 10^4$ examples and validation dataset of $n_{\text{val}} = 10^4$ examples, and fit a classifier using multiclass logistic regression (the method `train_prediction_model` from `conformal_multiclass_starter.py` may be helpful). Your classifier should achieve roughly 70+% accuracy on the validation data.

ii. Fit the standard (static) split-conformal predictor $\widehat{\tau}$ using your classifier f using the validation data and $\alpha = .1$.

iii. For a random matrix $W \in \mathbb{R}^{d \times p}$ with $d = 10$ and rows $w_1, \ldots, w_d \in \mathbb{R}^p$, find the best linear predictor of the quantiles $\widehat{\theta}$ as above, with $\alpha = .1$.

iv. Let Z_{test} denote the test data. On this dataset, evaluate the coverage of the resulting confidence sets on "extreme" subsets of the test set defined by inner products $w_i^T x$, that is, the subsets of the test defined by

$$Z_{\text{test},i} := \{(x, y) \in Z_{\text{test}} \mid w_i^T x \geq \text{QUANT}_{.9}(x^T w_i) \quad \text{or} \quad w_i^T x \leq \text{QUANT}_{.1}(x^T w_i)\},$$

the smallest and largest 10% of data as defined by $x^T w_i$, for each $i = 1, \ldots, d$. Record both the coverage of $\widehat{C}$ on $Z_{\text{test},i}$ as well as the average confidence set size. Also record the marginal coverage on Z_{test}.

Repeat this experiment 10 times (i.e., over 10 random splits of the training data into train and validation data), and provide a box plot of the coverage and confidence set sizes across the random splits defined by W; provide also the marginal coverage and marginal confidence set sizes on Z_{test}. Describe your observations.

8 Forecasting and Modeling

Question 8.1 (COVID Forecasting, **40 points**): The data file `covid_data.csv` contains data on the top 100 counties in the USA in terms of overall COVID case counts, between June and November 2020, with counties identified by a five-digit FIPS (Federal Information Processing Standard) code. In addition to the response variable `response`, which is the case incidence counts per 100,000 people, we include 31 signals that may be useful in predicting COVID-19 cases. The goal is to predict COVID counts 14 days ahead; we have cleaned the data somewhat by smoothing and performing a bit of simple imputation for missing values.

Your goal is to evaluate predictive models on the dates from November 1–30, 2020, on each of the 100 counties, using models fit for prediction on the preceding dates. In particular, fit a predictive model that for each county $c \in \{1, 2, \ldots, 100\}$, that at day t outputs a predictor $\widehat{y}_{c,t}$ of the expected case counts per 100,000 people on day t in county c. For your model, you will evaluate its performance in constructing predictive intervals for the true value $y_{c,t}$, but *only using data through day* $t - 14$, that is, preceding 14 days prior to date t. For each county c, you will construct a predictive interval $\widehat{C}_{c,t}$ for each date t in November 2020 (again, using only data through date $t - 14$) and each county c. To construct these intervals, consider two methods:

i. Use Adaptive Conformal Inference (ACI) [7], where we set

$$\widehat{C}_{c,t} = [\widehat{y}_{c,t} - \widehat{\tau}_{c,t}, \widehat{y}_{c,t} + \widehat{\tau}_{c,t}] \, .$$

Adjust the scalars $\widehat{\tau}_{c,t}$ via the updates

$$\widehat{\tau}_{c,t+1} = \widehat{\tau}_{c,t} - \eta \left(\mathbf{1} \left\{ y_{c,t} \in \widehat{C}_{c,t} \right\} - (1 - \alpha) \right) \tag{8.1}$$

for each t, c.

ii. Use a more sophisticated quantile tracking procedure. Define $s_{c,t} = |\log(1 + y_{c,t}) - \log(1 + \widehat{y}_{c,t})|$ for each t to be the error in predicted log case counts, and let $\boldsymbol{s}_t = [s_{c,t}]_{c=1}^{100}$ be the vector of county-wise scores (errors). You should use a predictive model for these scores of the form

$$\widehat{\boldsymbol{s}}_t = W_0 \boldsymbol{s}_{t-14} + \cdots + W_k \boldsymbol{s}_{t-14-k}$$

where k is a fixed number (say, $k = 5$) of time steps in the past, and $W_i \in \mathbb{R}^{100 \times 100}$ are matrices you fit. Then define

$$\widehat{C}_{c,t} = \{ y \in \mathbb{R} \text{ s.t. } |\log(1 + y) - \log(1 + \widehat{y}_{c,t})| \le \widehat{s}_{c,t} + \widehat{\tau}_{c,t} \} \, ,$$

where $\widehat{\tau}_{c,t}$ follows the update scheme (8.1).

Your submission should include the code you use to solve this problem, and should include a paragraph describing each of the following:

(a) Your model for predicting the case counts $y_{c,t}$ for each county based on the past data.

(b) How you fit the matrices W_i in part ii. Note that you *may* fit these matrices online; there is no requirement that you fit them on the offline/training data. (You don't need to fit them online, of course, either.) It is probably a good idea to regularize your matrices so that the diagonals are typically larger than the off-diagonals, as we would typically expect case counts in a county c to be more predictive for the county than other counties.

You should also include a table, with a row for each date in November (i.e., 30 entries), showing

(c) the fraction of counties covered for each of the confidence sets $\widehat{C}$ above, and the average confidence set width for each of the methods above.

JCD Comment: We ought to write an exercise with ERCOT data as well.

9 Regularization and advanced modeling

Question 9.1 (A semiparametric least squares model, 30 points)**:** Consider the model that predicts $\widehat{y}$ via

$$\widehat{y}_i = x_i^T \beta + f(x_i)$$

where $f : \mathbb{R}^p \to \mathbb{R}$ belongs to an RKHS with reproducing kernel $\mathsf{k} : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$. We have a sample $(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}$ of size n, and solve the least-squares estimation problem

$$(\widehat{\beta}, \widehat{f}) = \operatorname*{argmin}_{\beta, f} \left\{ \frac{1}{2} \left\| X\beta - f(X) - y \right\|_2^2 + \frac{\lambda_0}{2} \left\| \beta \right\|_2^2 + \frac{\lambda_1}{2} \left\| f \right\|^2 \right\}, \tag{9.1}$$

where $f(X) = [f(x_1) \; \cdots \; f(x_n)]^T \in \mathbb{R}^n$ denotes the vector of predictions of f and $\|f\|^2$ is the squared RKHS norm of f.

(a) If $K = [\mathsf{k}(x_i, x_j)]_{i,j \le n}$ is the Gram (Kernel) matrix, describe with a few words (literally) why problem (9.1) is equivalent to the problem

$$\operatorname*{minimize}_{\beta \in \mathbb{R}^p, \alpha \in \mathbb{R}^n} \; \frac{1}{2} \left\| X\beta - K\alpha - y \right\|_2^2 + \frac{\lambda_0}{2} \left\| \beta \right\|_2^2 + \frac{\lambda_1}{2} \alpha^T K \alpha. \tag{9.2}$$

(b) Show that the minimizers for problem (9.2) satisfy the consistency conditions

$$H_{\lambda_0} \widehat{\beta} = X^T (y - \widehat{f})$$
$$S_{\lambda_1} \widehat{f} = y - X\widehat{\beta}$$

where $\widehat{f} = [\widehat{f}(x_1) \; \cdots \; \widehat{f}(x_n)]^T = K\widehat{\alpha}$ is the semiparametric part of the model. Give the matrices H_{λ_0} and S_λ. (You may assume that K is invertible.)

(c) Show that we may solve problem (9.2) via the block matrix inversion problem

$$\begin{bmatrix} H_{\lambda_0} & X^T K \\ X & K + \lambda_1 I \end{bmatrix} \begin{bmatrix} \beta \\ \alpha \end{bmatrix} = \begin{bmatrix} X^T y \\ y \end{bmatrix}.$$

Question 9.2 (Fitting a semiparametric model, 40 points)**:** The datasets `adult_train.csv`, `adult_val.csv`, and `adult_test.csv` in the data directory contain random subsets of 2000 datapoints (each) from the Folktables package, with a full description available at https://github.com/socialfoundations/folktables. This consists of data with covariates for several categorical and numerical characteristics, including hours-per-week of work, educational attainment, and income. Treating income as the response, you will fit a semiparametric model as in Question 9.1.

For the non-income covariates, you should standardize the numerical covariates to have mean-zero and variance 1 across the data; for the non-numerical covariates, use a 1-hot encoding. (So if a categorical covariate has k distinct values, which may include missing, expand it into k positions in your vectors x with 1 in the position corresponding to the present category.) Note that this dataset has a few idiosyncrasies of which you ought to be aware: first, it is part of the census data from 1990 (updated through 1994), and so incomes were lower; it censors the highest income at 99999. You may ignore that censoring in your modeling. Second, we consider the following covariates in the model:

i. `hours_per_week`, a numerical covariate of the number of hours worked

ii. `age`, numerical, the age of the individual
iii. `workclass`, a binary variable of whether someone works in the private or public sector
iv. `education_num`, which is (related to) the number of years of education an individual has, with modifications, as 13 corresponds to completing a Bachelors, 10 some college, 9 finishing high school, among other strata.
v. `marital_status`, which is categorical
vi. `relationship`, which is categorical
vii. `race`, which includes mostly "White" and "Black" but three less common categories (which you may wish to group into "non-white-black")
viii. `sex`, which in this dataset is binary.

Use the Gaussian kernel function $\mathsf{k}(x, z) = \exp(-\frac{1}{2\tau^2}\|x - z\|_2^2)$, for $\tau > 0$ to be chosen, and regularization $\lambda_0 = 0$ to fit the model as in (9.2). Use the `adult_val.csv` data to perform held-out validation to choose the regularizer λ_1 and τ for the kernel, selecting values for each in the exponentially spaced range $\{2^{-2.5}, 2^{-2}, \ldots, 2^2, 2^{2.5}\} = \{2^{i/2}\}_{i=-5}^5$.

(a) What is the root-mean-square error on the data in `adult_test.csv` for the model you have selected?

(b) Assume that the estimate $\widehat{f}$ is sufficiently consistent that solving

$$\widehat{\beta} = \operatorname*{argmin}_\beta \frac{1}{n}\sum_{i=1}^n (y_i - \widehat{f}(x_i) - x_i^T\beta)^2$$

is equivalent to the "oracle" solution

$$\widehat{\beta}^{\text{oracle}} = \operatorname*{argmin}_\beta \frac{1}{n}\sum_{i=1}^n (y_i - f^\star(x_i) - x_i^T\beta)^2,$$

where $(\beta^\star, f^\star) = \operatorname{argmin}_{\beta,f} \mathbb{E}[(y - f(x) - x^T\beta)^2] + \lambda\|f\|^2$, using the notation of problem 9.1. Using this, give a sandwich covariance estimate, computable from the data, for the covariance in the approximation

$$\widehat{\beta} - \beta^\star \stackrel{\cdot}{\sim} \mathsf{N}(0, \widehat{\Sigma}). \tag{9.3}$$

(c) For the preceding covariance, give a 95% confidence interval for the component $\beta_j^\star$ associated to the `sex` variable.

(d) For the preceding covariance, give a 95% confidence interval for the variable corresponding to being `married`.

Question 9.3 (Reproducing Kernel Hilbert Spaces, 15 points)**:** In this question we explicate some of the conditions required for a symmetric $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ to be a valid kernel function. Recall that K is a valid kernel if for all sets of points $\{x_i\}_{i=1}^n \subset \mathcal{X}$, the Gram matrix

$$G := [K(x_i, x_j)]_{i,j=1}^n \in \mathbb{R}^{n\times n}$$

is positive semidefinite, that is, $G \succeq 0$. An equivalent statement is that $K(x, z) = \langle\phi(x), \phi(z)\rangle$ for some feature mapping ϕ and inner product $\langle\cdot,\cdot\rangle$.

(a) Let K_1, K_2 be valid kernel functions. Show that $K_1 + K_2$ is a valid kernel.

(b) Let K_1 be a kernel on $\mathbb{R} \times \mathbb{R}$ and let K_2 be a kernel on $\mathbb{R} \times \mathbb{R}$. Define the "direct sum" kernel $K : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ by

$$K((x_1, x_2), (z_1, z_2)) = K_1(x_1, z_1) + K_2(x_2, z_2).$$

Show that K is a valid kernel.

(c) Let K_1, K_2 be valid kernel functions. Show that $K_1 \cdot K_2$, that is, the function $K(x, z) = K_1(x, z)K_2(x, z)$ is a valid kernel.

Question 9.4 (A direct sum Hilbert space, 20 points)**:** Let $\mathcal{X}_1, \ldots, \mathcal{X}_d$ be arbitrary spaces (for example, each could be just a copy of $\mathbb{R}$), and let $\mathcal{X}^d = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ be their Cartesian product. (So $x \in \mathcal{X}^d$ has the form $x = (x_1, \ldots, x_d)$ for $x_j \in \mathcal{X}_j$.) Suppose that K_i is the reproducing kernel for a Hilbert space $\mathcal{H}_i$ of functions from spaces $\mathcal{X}_i \to \mathbb{R}$, where $\mathcal{H}_i$ has inner product $\langle \cdot, \cdot \rangle_i$. That is, $\langle K(x, \cdot), f \rangle_i = f(x)$ for any $f \in \mathcal{H}_i$ and $x \in \mathcal{X}_i$. Let $\mathcal{F}$ be the space of functions mapping $\mathcal{X}^d \to \mathbb{R}$ of the form

$$f(x) = \sum_{j=1}^{d} f_j(x_j),$$

where $f_j \in \mathcal{H}_j$. Define the direct sum inner product for $f, g \in \mathcal{F}$ by

$$\langle f, g \rangle = \sum_{j=1}^{d} \langle f_j, g_j \rangle_j,$$

noting that if $f \in \mathcal{F}$, then the reproducing property becomes $\langle f, K_j(x_j, \cdot) \rangle = \langle f_j, K_j(x_j, \cdot) \rangle_j = f_j(x_j)$, and for $K = \sum_{j=1}^{d} K_j$ we have the coordinate-wise reproducing inner product

$$\langle f, K(x, \cdot) \rangle = \sum_{j=1}^{d} \langle f_j, K_j(x_j, \cdot) \rangle = \sum_{j=1}^{d} f_j(x_j) = f(x).$$

(a) Write $\|f\|^2 = \langle f, f \rangle$ in terms of the norms $\|h\|_{\mathcal{H}_i}^2 := \langle h, h \rangle_i$, defined for $h \in \mathcal{H}_i$.

(b) Now you will demonstrate a variant of the representer theorem specialized to such direct sums. Consider the problem

$$\underset{f \in \mathcal{F}}{\text{minimize}} \; \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i), y_i) + \frac{\lambda}{2} \|f\|^2, \tag{9.4}$$

where $\lambda > 0$, $\|\cdot\|$ is the norm from part (a), and $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ is some loss function. Show that it is no loss of generality to assume that the minimizer of this problem takes the form

$$f(x) = \sum_{j=1}^{d} \sum_{i=1}^{n} \alpha_{ij} K_j(x_i, x),$$

and rewrite the problem (9.4) as an nd-dimensional optimization problem.

(c) Consider an extension of the previous part in which we model predictions of a response $y \in \mathbb{R}$ given $x \in \mathbb{R}^d$ as

$$\widehat{y}_{\theta, f}(x) = \theta_0 + x^T \theta + \sum_{j=1}^{d} f_j(x_j).$$

Show that for $\lambda_0 \geq 0, \lambda_1 > 0$, it is no loss of generality assume that the minimizers (in f) of the problem

$$\underset{\theta \in \mathbb{R}^{d+1}, f \in \mathcal{F}}{\text{minimize}} \frac{1}{n} \sum_{i=1}^{n} \ell(\widehat{y}_{\theta,f}(x_i), y_i) + \lambda_0 \cdot \text{reg}(\theta) + \lambda_1 \|f\|^2 \tag{9.5}$$

take the form $f(x) = \sum_{j=1}^{d} \sum_{i=1}^{n} \alpha_{ij} K_j(x_i, x)$.

Question 9.5 (ℓ_1-regularization and forward-selection, 20 points)**:** Consider a forward-selection- or boosting-type procedure for predicting targets y from $x \in \mathcal{X}$, where at iteration k we have a feature mapping $\phi^k : \mathcal{X} \to \{-1, 1\}^k$, $\phi^k(x) = (\phi_1(x), \ldots, \phi_k(x))$, and we wish to add a new feature $\phi_{k+1} : \mathcal{X} \to \{-1, 1\}$. At iteration k, our predictive model is thus

$$f_k(x) = \langle \theta^k, \phi^k(x) \rangle = \sum_{j=1}^{k} \theta_j \phi_j(x).$$

We assume we are minimizing a loss $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$, convex in its first argument, so that this new feature should (approximately) minimize

$$\frac{1}{n} \sum_{i=1}^{n} \ell(f_k(x_i) + \theta_{k+1} \phi_{k+1}(x_i), y_i)$$

jointly in $\theta_{k+1} \in \mathbb{R}$ and ϕ_{k+1}.

At each iteration, we conduct a hypothesis test to assess whether to add a prospective new feature ϕ_{k+1}. Say that the null at iteration $k+1$ is that

$$H_{0,k+1} : \operatorname*{argmin}_{\theta} \{\mathbb{E}[\ell(f_k(x) + \theta \phi_{k+1}(x), y)]\} = 0$$

(where the expectation is over (x, y) drawn from the population being sampled).

(a) Show that the null $H_{0,k+1}$ equivalent to the equality

$$\mathbb{E}[\ell'(f_k(x), y) \phi_{k+1}(x)] = 0,$$

where $\ell'(t, y) = \frac{\partial}{\partial t} \ell(t, y)$.

(b) Ignoring the issue that f_k depends on the sample, an approximation to the preceding condition is that

$$\frac{1}{n} \sum_{i=1}^{n} \ell'(f_k(x_i), y_i) \phi_{k+1}(x_i) \stackrel{\cdot}{\sim} \mathsf{N}\left(0, \frac{1}{n}\left(\frac{1}{n} \sum_{i=1}^{n} \ell'(f_k(x_i), y_i)^2\right)\right) \tag{9.6}$$

(because $\phi_{k+1}(x_i)^2 = 1$ for each x_i). Give an (approximate) level $1 - \alpha$ test of $H_{0,k+1}$ using the approximation (9.6), that is, test whether $\theta^\star_{k+1} = 0$.

(c) Suppose we are given the potential new feature mapping ϕ_{k+1} and choose the value θ_{k+1} as

$$\theta_{k+1} = \operatorname*{argmin}_{\theta} \left\{ \frac{1}{n} \sum_{i=1}^{n} \ell(f_k(x_i) + \theta \phi_{k+1}(x_i), y_i) + \lambda |\theta| \right\}.$$

Give the value $\lambda > 0$ such that $\theta_{k+1} \neq 0$ if and only if your test from part (b) rejects that $\theta^\star_{k+1} = 0$.

(d) Assume now that $\ell(t, y)$ has M-Lipschitz continuous derivative in t, or, equivalently, that $\ell''(t, y) \le M$ for all t. Show that with your value λ from part (c), the alternative update

$$\theta_{k+1} = \operatorname*{argmin}_{\theta} \left\{ \left(\frac{1}{n} \sum_{i=1}^{n} \ell'(f_k(x_i), y_i) \phi_{k+1}(x_i) \right) \cdot \theta + \frac{M}{2} \theta^2 + \lambda |\theta| \right\},$$

which arises by upper bounding ℓ with a quadratic, satisfies $\theta_{k+1} \neq 0$ if and only if your test from part (b) rejects that $\theta^\star_{k+1} = 0$.

(e) Let $\ell(t, y) = \log(1 + e^{t-y}) + \log(1 + e^{y-t})$ be a smooth robust regression loss. Give $M = \sup_{t \in \mathbb{R}} \ell''(t, y)$.

References

[1] A. Angelopoulos, J. C. Duchi, and T. Zrnic. PPI++: Efficient prediction-powered inference. *arXiv:2311.01453 [stat.ML]*, 2023.

[2] A. N. Angelopoulos, S. Bates, C. Fannjiang, M. I. Jordan, and T. Zrnic. Prediction-powered inference. *Science*, 382:669–674, 2023.

[3] H. Asi and J. C. Duchi. The importance of better models in stochastic optimization. *Proceedings of the National Academy of Sciences*, 116(46):22924–22930, 2019. URL https://doi.org/10.1073/pnas.1908018116.

[4] H. Asi and J. C. Duchi. Stochastic (approximate) proximal point methods: Convergence, optimality, and adaptivity. *SIAM Journal on Optimization*, 29(3):2257–2290, 2019.

[5] D. P. Bertsekas. Stochastic optimization problems with nondifferentiable cost functionals. *Journal of Optimization Theory and Applications*, 12(2):218–231, 1973.

[6] J. C. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[7] I. Gibbs and E. J. Candès. Conformal inference for online prediction with arbitrary distribution shifts. *Journal of Machine Learning Research*, 25(162):1–36, 2024.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645, 2016.