Exercises for Stats305a

Prof. John Duchi

October 21, 2022

Note: All data files available at https://web.stanford.edu/class/stats305a/Data/.

0 Matrix review questions

Question 0.1: Let R be a square right-triangular (upper triangular) matrix, that is,

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1n} \\ 0 & r_{22} & r_{23} & \cdots & r_{2n} \\ 0 & 0 & r_{33} & \cdots & r_{3n} \\ 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & r_{nn} \end{bmatrix}$$

Assume $\mathrm{diag}(R)$ is all non-zero. Give a (short!) algorithm to solve for x in $Rx = b$.

Question 0.2 (Projections)**:** Let $A \in \mathbb{R}^{m \times n}$ where $m < n$ and A has full row rank (that is, $\mathrm{rank}(A) = m$), and let $b \in \mathbb{R}^m$. Consider the affine space (subspace plus an offset)

$$\mathcal{S} := \{x \in \mathbb{R}^n \mid Ax = b\}.$$

(a) Let $y \in \mathbb{R}^n$. Give the (Euclidean) projection

$$\pi_{\mathcal{S}}(y) := \operatorname*{argmin}_{x \in \mathcal{S}} \left\{ \|y - x\|^2 \right\}.$$

(b) Draw a picture of your result above.

Question 0.3: A *Householder* reflection (or transformation) is $H_u = I - 2uu^\top$ where u is a unit vector and I is the identity.

(a) Show that H_u is symmetric and unitary, meaning $H_u^\top = H_u$ and $H_u^\top H_u = I$.

(b) Draw a picture of the mapping $x \mapsto H_u x$ exhibiting why this is called a reflection.

(c) We show how to reflect a vector x about the line between the direction of x and the first standard basis vector e_1, so that the transform $H_u x$ is on the line $\{te_1 \mid t \in \mathbb{R}\}$. Let $x \in \mathbb{R}^n$ be an arbitrary vector, and define

$$v := \frac{x/\|x\| + e_1}{\|x/\|x\| + e_1\|}$$

(taking $v = 0$ if $x = -te_1$ for some $t > 0$). Show that for $H_v x = -\|x\| e_1$.

(d) Let $u_1, \dots, u_k$ be arbitrary unit vectors. Show that the following matrix is unitary:

$$H := \prod_{i=1}^{k} H_{u_i}.$$

(e) Let $A \in \mathbb{R}^{n \times n}$ be full rank with first column a_1. Give a Householder transformation (by specifying the unit vector u) so that

$$H_u A = \begin{bmatrix} -\|a_1\| & * & \cdots & * \\ \mathbf{0}_{n-1} & & B & \end{bmatrix}$$

where $*$s are arbitrary numbers, $\mathbf{0}_{n-1} \in \mathbb{R}^{n-1}$ is all zeros, and $B \in \mathbb{R}^{n-1 \times n-1}$ is a matrix.

(f) Given a matrix A with the block structure

$$A = \begin{bmatrix} R & * \\ \mathbf{0} & B \end{bmatrix},$$

where R is square and upper triangular, $B \in \mathbb{R}^{k \times k}$ is square and has first column b_1, and $\mathbf{0}$ is an all-zeros matrix of appropriate size, give a symmetric unitary matrix S so that

$$SA = \begin{bmatrix} R & * & * \\ \mathbf{0} & -\|b_1\| & * \\ \mathbf{0} & \mathbf{0}_{k-1} & C \end{bmatrix}.$$

(g) Describe (in one or two sentences) how one may use Householder transformation to construct a QR factorization of any full rank matrix A.

Question 0.4 (The power method)**:** Let $A \in \mathbb{R}^{n \times n}$ be diagonalizable, meaning $A = S \Lambda S^{-1}$ for a diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_n)$, and let S have columns $S = [s_1 \ \cdots \ s_n]$. Assume the largest eigenvalue is unique, with $|\lambda_1| > |\lambda_j|$ for each $j \neq 1$, and consider the iteration

$$x^{k+1} \leftarrow \frac{1}{\|x^k\|} A x^k, \quad k = 0, 1, 2, \dots$$

where $\|\cdot\|$ is the usual ℓ_2-norm.

Recall that a diagonalizable matrix has left and right eigenvectors, where left eigenvectors satisfy $v^\top A = \lambda v^\top$, that is, $A^\top v = \lambda v$, while right eigenvectors satisfy $Av = \lambda v$. (Note that the left and right eigenvalues are necessarily identical.) Let $v_1, \dots, v_n$ be the left eigenvectors of A, and assume that $v_1^\top x^0 \neq 0$, that is, the initial iterate x^0 is *not* orthogonal to the left eigenvector v_1 corresponding to the largest eigenvalue λ_1.

(a) To what does x^k converge?

(b) Prove it.

Hint. Convince yourself that the left eigenvectors are the rows of the matrix S^{-1}. Because S is full rank, we can write any $x \in \mathbb{R}^n$ as $x = S\alpha$ for some $\alpha \in \mathbb{R}^n$. Show that if $x^0 = S\alpha$ then $\alpha_1 \neq 0$ as $v_1^\top x^0 \neq 0$, then develop the iteration for x^k.

Question 0.5 (Block matrix inversion, linear algebra, and the Sherman-Morrison-Woodbury formula)**:** In this question, we will work out a few formulae for inverting block-structured matrices, using them to develop various inversion formulas for structured matrices.

(a) Consider the matrix equation

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

where A and D are square matrices and B, C have appropriate sizes (this is not important for this question). Give a formula for x in terms of A, B, C, D, a, and b; your formula, if correct, will involve inverses of some of these. You may assume that A and D are invertible and that $A - BD^{-1}C$ is invertible. (Note: B and C may be rectangular, so don't try to invert them alone.)

(b) We now consider inverting a matrix plus a (typically) low rank matrix. We wish to solve

$$(A + UCV^\top)x = z$$

for x, where $A \in \mathbb{R}^{n\times n}$, $U \in \mathbb{R}^{n\times k}$, $C \in \mathbb{R}^{k\times k}$, and $V \in \mathbb{R}^{n\times k}$. Introducing the variable $y = CV^\top x$, or $V^\top x - C^{-1}y = 0$, solve

$$\begin{bmatrix} A & U \\ -V^\top & C^{-1} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} z \\ 0 \end{bmatrix}$$

for x. Using this, show that

$$x = \left(A^{-1} - A^{-1}U(C^{-1} + V^\top A^{-1}U)^{-1}V^\top A^{-1}\right) z,$$

i.e.,

$$(A + UCV^\top)^{-1} = A^{-1} - A^{-1}U(C^{-1} + V^\top A^{-1}U)^{-1}V^\top A^{-1}.$$

(As an aside, if you know A^{-1} and C^{-1} already, the largest matrix you must invert to compute $(A + UCV^\top)^{-1}$ is then $k \times k$, which is smaller.)

Question 0.6 (Majorization inequalities)**:** A matrix $P \in \mathbb{R}^{n\times n}$ is a *permutation matrix* if its entries are all $\{0,1\}$-valued and $P^\top P = I$, that is, P has a single 1 in each row and column. Let $\mathcal{P}_n$ be the collection of $n \times n$ permutation matrices. The Birkhoff polytope is the convex hull of the permutation matrices and coincides with the *doubly stochastic matrices*, where we recall a matrix $S \in \mathbb{R}^{n\times n}$ is doubly stochastic if $S\mathbf{1} = S^\top\mathbf{1} = \mathbf{1}$ and its entries are nonnegative. That is, letting $\mathcal{S}_n$ be the doubly stochastic matrices, we have

$$\mathcal{S}_n = \operatorname{Conv}(\mathcal{P}_n) = \left\{ S = \sum_{l=1}^k \lambda_l P_l \mid P_l \in \mathcal{P}_n \text{ and } \lambda_l \ge 0, \sum_{l=1}^k \lambda_l = 1 \right\},$$

so each $S \in \mathcal{S}_n$ is a convex combination of permutation matrices.

(a) Let $a_1 \ge a_2$ and $b_1 \ge b_2$. Show that

$$a_1b_1 + a_2b_2 \ge a_2b_1 + a_1b_2.$$

(b) Let $u, v \in \mathbb{R}^n$ and assume $v_1 \ge v_2 \ge \cdots \ge v_n$ and $u_1 \ge u_2 \ge \cdots \ge u_n$. Show that

$$u^\top v \ge u^\top Pv \text{ for all } P \in \mathcal{P}_n.$$

You may use that if $\sigma : [n] \to [n]$ is any permutation, there is a sequence of transpositions (i.e., swaps of two elements, so that if σ' and σ are identical except that $\sigma(i) = \sigma'(i+1)$ and $\sigma(i+1) = \sigma'(i)$ for some index i, then they are transpositions) that transform σ into the identity permutation.

(c) Assume u and v are as in part (a). Show that

$$u^\top v = \max_{S \in \mathcal{S}_n} u^\top S v.$$

Question 0.7: Let $A, B \in \mathbb{R}^{n \times n}$. The *von Neumann trace inequality* states that

$$\operatorname{tr}(AB) \le \sum_{i=1}^{n} \sigma_i(A)\sigma_i(B) \tag{vNTI}$$

where $\sigma_1(A) \ge \cdots \ge \sigma_n(A) \ge 0$ denote the singular values of A (and similarly for $\sigma_i(B)$). In this question, you will demonstrate this inequality.

(a) Show that it is no loss of generality to assume that A is diagonal, that is, to show that inequality (vNTI) holds when $A = \operatorname{diag}(a_1, \ldots, a_n)$ with $a_1 \ge \cdots \ge a_n \ge 0$.

(b) Let $B = U\Sigma V^\top$ where $U = [u_1 \ \cdots u_n]$ and $V = [v_1 \ \cdots \ v_n]$ are unitary, and $A = \operatorname{diag}(a_1, \ldots, a_n)$ where $a_i \ge 0$. Show that

$$\operatorname{tr}(AB) = \sum_{i=1}^{n} \sigma_i v_i^\top A u_i = \sum_{i,j \le n} \sigma_i a_j v_{ij} u_{ij}$$

(c) Under the same conditions as in part (b), show that

$$\operatorname{tr}(AB) \le \frac{1}{2} \sum_{i,j \le n} \sigma_i v_{ij}^2 a_j + \frac{1}{2} \sum_{i,j \le n} \sigma_i u_{ij}^2 a_j.$$

Hint. For any $x, y \in \mathbb{R}$, we have $xy \le \frac{1}{2}x^2 + \frac{1}{2}y^2$ because $0 \le \frac{1}{2}(x-y)^2 = \frac{1}{2}x^2 - xy + \frac{1}{2}y^2$.

(d) Using the preceding parts and Question 0.6, show the trace inequality (vNTI).

1 A few preliminary questions

Question 1.1 (A rank-one update to a linear regression solution)**:** Consider a least-squares problem with data $X \in \mathbb{R}^{n \times d}$, $Y \in \mathbb{R}^n$, where $n \geq d$ and X has rank d, and let

$$\widehat{\beta}_n = \operatorname*{argmin}_{\beta} \|X\beta - Y\|^2 .$$

Assume we get a new observation $(x_{n+1}, y_{n+1}) \in \mathbb{R}^d \times \mathbb{R}$ and wish to update

$$\widehat{\beta}_{n+1} = \operatorname*{argmin}_{\beta} \left\{ \|X\beta - Y\|^2 + (x_{n+1}^T \beta - y_{n+1})^2 \right\} .$$

Give a formula for $\widehat{\beta}_{n+1}$ in terms of $\widehat{\beta}_n$, $(X^T X)^{-1}$, x_{n+1}, and y_{n+1}.

Assuming you have already computed $H = (X^T X)^{-1} \in \mathbb{R}^{d \times d}$ and that multiplying H by a vector, i.e., computing Hv, takes time d^2, roughly how much time does computing your update take? (Note: you can simply say "A few multiples of d," or "A few multiples of d^2," or "A few multiples of d^3," depending on which is true.)

Question 1.2 (The most negatively correlated distribution)**:** A financial analyst tells you that he has a great stock tip that will allow you to short stocks based on others that are doing well. He assures you that he has found a correlation matrix between the n stocks, $C \in \mathbb{R}^{n \times n}$, with entries

$$C_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1/2 & \text{otherwise.} \end{cases}$$

(a) Is his correlation matrix possible? Would you trust him with your graduate stipend?

(b) More generally, consider a correlation matrix C_ρ with entries of the following form:

$$[C_\rho]_{ij} = \begin{cases} 1 & \text{if } i = j \\ -\rho & \text{otherwise} \end{cases}$$

where $\rho \geq 0$. How large can ρ be while C_ρ remains a potentially valid correlation matrix? *Hint:* A correlation matrix for a random vector $X \in \mathbb{R}^n$ has entries $C_{ij} = \mathrm{Cov}(X_i, X_j)/\sqrt{\mathrm{Var}(X_i)\mathrm{Var}(X_j)}$, and so may be written

$$C = \mathrm{diag}(v)^{-1/2} \mathrm{Cov}(X)\, \mathrm{diag}(v)^{-1/2},$$

where v is a vector with entries $v_i = \mathrm{Var}(X_i)$ and $\mathrm{diag}(v)$ is the diagonal matrix with diagonal v.

Question 1.3 (Some basic plotting and data processing)**:** The UCI Machine Learning repository has a collection of useful datasets for experiments and data exploration. This is a question that simply serves as a forcing function for you to pick a computer language, read in data, and plot it appropriately. Using the data in the UCI Wine quality dataset (`https://archive.ics.uci.edu/ml/datasets/Wine+Quality`, and see the `winequality-red.csv` file in the data folder there), plot a scatterplot matrix showing the five variables `density`, `alcohol`, `pH`, `volatile.acidity`, and the target variable `quality`, which is a measure of wine quality. Such pairwise scatterplots can be a useful tool for data exploration and summarization (see, e.g., Fig. 1.1 of [5]).

In your plots, what do you notice about density, alcohol, and quality? (Just a sentence suffices here.)

Question 1.4 (Predicting high temperatures at SFO)**:** In this question, we use linear regression to predict the high temperature at San Francisco International Airport (SFO). A natural model of the temperature is the following: let x be the day of the year (that is, $x = 1$ corresponds to January 1, while $x = 365$ corresponds to December 31, except in leap years); we assume that the temperature

$$y = \beta_0 + \beta_1 \sin\left(\frac{2\pi}{365}(x-1)\right) + \beta_2 \cos\left(\frac{2\pi}{365}(x-1)\right). \tag{1.1}$$

(Admittedly this ignores the issue of leap years, but we will punt on that.) Let $\phi(x) = [1 \;\; \sin(\frac{2\pi}{365}(x-1)) \;\; \cos(\frac{2\pi}{365}(x-1))]^T$ be the vector feature representation above.

The data file `simplified-sfo-weather.csv` contains weather data for SFO since 1960, including precipitation (in inches), low, and high temperatures (in degrees Fahrenheit). Note that in May 2018, the temperature sensor broke and consequently a few days report `NA` as the high and low temperatures. You should simply omit those from any averages or model fitting.[1] Using this data, fit the model (1.1) to predict high temperature (this is column `"temphigh"` in the file) from the date x of the year for years prior to 1990. For each decade 1961–1970, 1971–1980, 1981–1990, 1991–2000, 2001–2010, and 2011–2020, print the mean of the actual high temperature minus the predicted high temperature for the decade. That is, if $\widehat{\beta} = [\widehat{\beta}_0 \; \widehat{\beta}_1 \; \widehat{\beta}_2]^T$ denotes your fit model, compute the average difference

$$y - \widehat{y} = y - \phi(x)^T \widehat{\beta}$$

over each of those decades. Include your code and a printout of the results.

Question 1.5: We consider monitoring changes in rainfall/precipitation over the years at San Francisco International Airport (SFO) using the data in `simplified-sfo-weather.csv`. To do so, we will set up a standard linear model with $d = 3$ dimensions, where for dates (times) $t \in \{1, 2, 3, \ldots, 366\}$ (we have 366 for leap years) we set

$$x = \left[1 \quad \sin\left(\frac{2\pi}{365}(t-1)\right) \quad \cos\left(\frac{2\pi}{365}(t-1)\right)\right]^T \in \mathbb{R}^d \tag{1.2}$$

where $d = 3$. Under the standard linear model

$$y_i = x_i^T \beta + \varepsilon_i, \quad \varepsilon_i \overset{\text{iid}}{\sim} \mathsf{N}(0, \sigma^2), \quad i = 1, 2, 3, \ldots,$$

we would like to test whether future data follows a similar distribution to the past data.

We begin with a few mathematical generalities. Consider two datasets modeled by

$$Y = X\beta + \varepsilon, \qquad Y_{\text{new}} = Z\beta + \varepsilon_{\text{new}},$$

where $X \in \mathbb{R}^{m \times d}$ and $Z \in \mathbb{R}^{n \times d}$ are the given covariates, and we model $\varepsilon \sim \mathsf{N}(0, \sigma^2 I_m)$ and $\varepsilon_{\text{new}} \sim \mathsf{N}(0, \sigma^2 I_n)$ independently; we will think of (X, Y) as the initial data pair and (Z, Y_{new}) as the new data. (Their particular form is immaterial; we assume both X and Z are rank d.) Let $\widehat{\beta} = (X^T X)^{-1} X^T Y$, be the usual least-squares estimate on the "initial" data pair (X, Y), let $H_X = X(X^T X)^{-1} X^T \in \mathbb{R}^{m \times m}$ be the usual hat matrix, and define the predicted values

$$\widehat{Y} := X\widehat{\beta} = H_X Y \;\; \text{and} \;\; \widehat{Y}_{\text{new}} := Z\widehat{\beta}.$$

[1] In `R`, you may do this automatically in the `lm` methods using the keyword `na.action = na.omit`, and in computing a `mean` using `na.rm = TRUE`.

(a) Show that the residuals $Y - \widehat{Y}$ and $Y_{\text{new}} - \widehat{Y}_{\text{new}}$ are independent.

(b) Give a (symmetric, positive definite) matrix $M \in \mathbb{R}^{n \times n}$ so that

$$M(Y_{\text{new}} - \widehat{Y}_{\text{new}}) \sim \mathsf{N}(0, \sigma^2 I_n),$$

where I_n denotes the $n \times n$ identity matrix.

(c) Give the distribution of the ratio

$$A := \frac{\frac{1}{n}\left\|M(Y_{\text{new}} - \widehat{Y}_{\text{new}})\right\|_2^2}{\frac{1}{m-d}\left\|Y - \widehat{Y}\right\|_2^2} \tag{1.3}$$

under the null hypothesis

$$H_0 : \begin{cases} Y = X\beta + \varepsilon, & Y_{\text{new}} = Z\beta + \varepsilon_{\text{new}}, \\ \varepsilon \sim \mathsf{N}(0, \sigma^2 I_n), & \varepsilon_{\text{new}} \sim \mathsf{N}(0, \sigma^2 I_n), \quad \varepsilon \perp\!\!\!\perp \varepsilon_{\text{new}}. \end{cases}$$

We now consider implementing a series of hypothesis tests about whether the precipitation at SFO is remaining consistent over the years or whether it is changing in some meaningful way.

(d) For each of the years 1966, 1967, . . ., 2020, repeat the following. Define a data matrix X using the featurization (1.2) consisting of *all* dates prior to that year (so that for 1966, X will be a data matrix for the years 1960–1965, for 1967, X will be the data for years 1960–1966, and so on). Define the responses Y to consist of precipitation (column `precip` in the `csv` file) for the given years. Define the new data matrix $Z \in \mathbb{R}^{n \times d}$ to consist of the $n = 365$ (or 366 in a leap year) rows in the given year and the responses Y_{new} to be the precipitation in the given year. For this data, compute the statistic A in Eq. (1.2) and its p-value, that is, conditional on $A = a$, report

$$p := \mathbb{P}(A \geq a)$$

where A follows the null above.

Give a printed list of your p values for each of the years, and provide a plot of the p values for each of the years as well. Provide a few sentences of discussion about the observed p-values.

Please include your code in your solution.

(e) For a fixed year, suppose we perform the procedure in part (d) for only that year, getting a single p-value. Suppose that this p-value is very small, for example, something like $p = 10^{-5}$ or $p = 10^{-6}$. In one or two sentences, does rejecting the null hypothesis necessarily mean that the distribution of precipitation is changing over time?

2 Sampling error and testing

Question 2.1 (Regression with random samples and best linear approximations)**:** Say that $(X, Y) \in \mathbb{R}^d \times \mathbb{R}$ come from a joint probability distribution P, where $\mathbb{E}[XX^T] = C \succ 0$ and X, Y both have finite second moments. Assume that the first coordinate of X is constant, with $X_1 = 1$. For $x \in \mathbb{R}^d$, define the regression function

$$f^\star(x) := \mathbb{E}[Y \mid X = x].$$

(a) Show that f minimizes $\mathbb{E}[(Y - f(X))^2]$ over all functions $f : \mathbb{R}^d \to \mathbb{R}$.

(b) Instead of fitting a model of $Y \mid X$ over the space of all functions, consider fitting one over all linear predictors, and choosing $\beta^\star$ to minimize the expected squared loss

$$L(b) := \frac{1}{2}\mathbb{E}[(Y - X^T b)^2]$$

over $b \in \mathbb{R}^d$. Characterize the solution $\beta^\star$ (i.e., give a formula for it), and show that the linear function $\varphi(x) = \beta^{\star T} x$ is the best linear approximation to f (in mean-squared distance). *Note:* in case you are worried about it, it is fine to exchange expectation and differentiation in this case; you definitely don't need to show that, though it *is* true (see, for example, Bertsekas [2]).

(c) Say that we have an i.i.d. sample (x_i, y_i), $i = 1, \ldots, n$ from P, with $y_i = f(x_i) + \varepsilon_i$, and

$$\widehat{\beta} = \operatorname*{argmin}_b L_n(b) := \frac{1}{2n}\sum_{i=1}^n (x_i^T b - y_i)^2$$

is the ordinary least-squares estimator, and assume $n \ge d$. Is $\widehat{\beta}$ unbiased for $\beta^\star$?

Question 2.2 (T statistics, F statistics, and linear algebra)**:** Consider the model $y = X\beta + \varepsilon$, $\varepsilon \sim \mathsf{N}(0, \sigma^2 I)$, $X \in \mathbb{R}^{n \times d}$ with rank d. The t-statistic for a coordinate j is

$$t_j = \frac{\widehat{\beta}_j}{\widehat{\text{se}}(\widehat{\beta}_j)},$$

where $\widehat{\text{se}} = \widehat{\sigma}\sqrt{e_j^T (X^T X)^{-1} e_j}$ is the usual standard error estimate for $\widehat{\beta}_j$. For example, `R` reports p-values for these t-statistics when using `lm` and `summary`. Let X have columns $x^{(j)}$, $j = 1, \ldots, d$, and $X_{\setminus j}$ be X with column j removed, i.e.

$$X_{\setminus j} = \left[x^{(1)} \ \cdots \ x^{(j-1)} \ x^{(j+1)} \ \cdots \ x^{(d)}\right],$$

which is a submodel as we have discussed in class. The F-statistic for coordinate j is then

$$F_j = \frac{\|(H - H_j)y\|_2^2}{\frac{1}{n-d}\|(I - H)y\|_2^2},$$

where $H = X(X^T X)^{-1} X^T$ is the usual hat matrix (projection onto range(X)) and H_j is the projection matrix onto range$(X_{\setminus j})$. Show that $t_j^2 = F_j$.

Hint. Assume without loss of generality that $j = d$, the dth component. (One can do so by permutating the columns of X.) Consider the QR factorization of X, i.e., $X = QR$ where $Q \in \mathbb{R}^{n\times n}$ is orthogonal and $R \in \mathbb{R}^{n\times d}$ has the form

$$R = \begin{bmatrix} T \\ \mathbf{0}_{n-d\times d} \end{bmatrix}$$

for an upper triangular (invertible) matrix T with entries $T_{ij} = R_{ij}$ for all $1 \le i, j \le d$.

Question 2.3 (Non-independent noise and testing challenges)**:** A subtle but problematic situation occurs in linear models when noise is correlated instead of independent—indeed, this is often *much* worse than non-normality of noise, which the central limit theorem more or less addresses. To make this a bit more concrete, we consider a 2-group ANOVA model,

$$y_{1j} = \mu + \alpha_1 + \varepsilon_{1j}, \quad y_{2j} = \mu + \alpha_2 + \varepsilon_{2j}, \tag{2.1}$$

where we assume we observe a sample of size n for each group (i.e. $j = 1, \ldots, n$). The standard assumption is that $\varepsilon_i \sim \mathsf{N}(0, \sigma^2 I_n)$, where we use $\varepsilon_i = [\varepsilon_{i1} \ \cdots \ \varepsilon_{in}]^T \in \mathbb{R}^n$ for shorthand, and we have the null model

$$H_0 : \alpha_1 = \alpha_2, \quad \varepsilon_{ij} \overset{\text{iid}}{\sim} \mathsf{N}(0, \sigma^2).$$

In this case, for $\overline{y}_i = \frac{1}{n}\sum_{j=1}^n y_{ij}$ and standard error estimate

$$S_n^2 := \frac{1}{2n-2}\left[\sum_{j=1}^n (y_{1j} - \overline{y}_1)^2 + \sum_{j=1}^n (y_{2j} - \overline{y}_2)^2)\right]$$

the usual t-statistic is

$$t := \frac{\overline{y}_1 - \overline{y}_2}{S_n\sqrt{2/n}} \sim T_{2n-2},$$

the t-distribution with $2n - 2$ degrees of freedom, or equivalently,

$$\widehat{F} := \frac{\frac{n}{2}(\overline{y}_1 - \overline{y}_2)^2}{S_n^2} \sim F_{1,2n-2}.$$

We will show that a test that rejects when F is large (i.e., the standard ANOVA) may reject unrealistically frequently when the errors are correlated.

To that end, consider the situation that $\varepsilon_1, \varepsilon_2$ are independent, but

$$\varepsilon_i \sim \mathsf{N}\left(0, \sigma^2(1-\rho)I_n + \sigma^2\rho\mathbf{1}\mathbf{1}^T\right), \tag{2.2}$$

where $\rho \in [0, 1]$ indicates correlation within the group. Such correlation may be reasonable, e.g., when (hidden) confounding relates members of a group. Through the remainder of this question, let $C_\rho = (1-\rho)I_n + \rho\mathbf{1}\mathbf{1}^T$ be a shorthand for the covariance.

(a) Show that if $Z \sim \mathsf{N}(0, C_\rho)$, then $(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)Z = Z - \mathbf{1}\overline{Z}_n \sim \mathsf{N}(0, (1-\rho)(I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T))$.

(b) Show that $\frac{1}{1-\rho}S_n^2 \sim \frac{\sigma^2}{2n-2} \cdot \chi^2_{2n-2}$ under the correlation structure (2.2).

(c) Show that if $\alpha_1 = \alpha_2$ and the correlation (2.2) holds,

$$\overline{y}_1 - \overline{y}_2 \sim \mathsf{N}\left(0, \sigma^2\left(2\frac{1-\rho}{n} + 2\rho\right)\right).$$

(d) Argue that $\overline{y}_1 - \overline{y}_2$ is independent of S_n^2 even with correlation (2.2).

(e) Show that under correlation (2.2),

$$\widehat{F}_\rho := \frac{1-\rho}{1-\rho+\rho n} \cdot \widehat{F} = \frac{\frac{n}{2(1-\rho)+2\rho n}(\overline{y}_1 - \overline{y}_2)^2}{\frac{1}{1-\rho} S_n^2} \sim F_{1,2n-2},$$

so that the *valid* test is to reject when $\widehat{F}_\rho$ is large.

(f) Argue that as n grows large, the standard ANOVA will falsely reject the null hypothesis H_0 too frequently when the correlation (2.2) holds.

Question 2.4 (Intuition for correlated rejections via simulation)**:** Here we revisit question 2.3, except that we perform some simulations and corrections. First, we describe a general strategy for eliminating correlations in the noise, making the *prima facie* ridiculous assumption that we know the noise covariance.

(a) Let $y = X\beta + \varepsilon$ where $\varepsilon \sim (0, \Sigma)$. Show that $\Sigma^{-1/2} y = \Sigma^{-1/2} X\beta + \xi$, where $\xi \sim (0, I_n)$.

By part (a), if we knew Σ, we could make the substitutions

$$\widetilde{y} = \Sigma^{-1/2} y, \quad \widetilde{X} = \Sigma^{-1/2} X$$

and perform ordinary least squares on $(\widetilde{X}, \widetilde{y})$; all of the distributional results and tests we have developed would then work.

(b) Repeat the following experiment several (say, 100) times for values of n = 2, 4, 8, 16, 32, 64, 128, 256, 512. Generate data from the ANOVA model (2.1), except that the noise is correlated (2.2) with $\rho = .1$, with $\mu = \alpha_1 = \alpha_2 = 0$. Perform an F-test of significance at level $\alpha = .05$, rejecting when $\widehat{F}$ is large. (As the null $\alpha_1 = \alpha_2$ holds, any rejections of equality of means is false, though a rejection of the ANOVA model with independent noise is sensible.) Plot the frequency of false rejections against sample size n.

It is of interest to *correct* an estimate for possible correlations, thereby achieving a test whose nominal level is closer to accurate. In general, one never has enough data to estimate $\mathrm{Cov}(\varepsilon)$ in a linear regression model except under assumptions on the noise model. In the ANOVA model (2.1), it may be reasonable to assume that *within* a group, the noises are all equally correlated, that is, the noise model (2.2) holds, and we can approximate $\mathrm{Cov}(\varepsilon_i)$. Note that $\mathbb{E}[(y_{1j} - y_{2l})^2] = 2\sigma^2$ and that $y_1 \perp\!\!\!\perp y_2$ under model (2.2) and the null $\alpha_1 = \alpha_2$. Define the estimates

$$\widehat{\sigma}^2 := \frac{1}{2n^2} \sum_{j,l} (y_{1j} - y_{2l})^2 \quad \text{and} \quad \widehat{\rho} = 1 - \frac{S_n^2}{\widehat{\sigma}^2},$$

so that $\widehat{\rho} \to \rho$ as $n \to \infty$ (you do not need to show this! We are simply asserting it). Then the plug-in test uses the statistic $\widehat{F}_\rho$ from Question 2.3, except we replace ρ with $\widehat{\rho}$.

(c) Repeat your experiment from part (b), except that you use the statistic $\widehat{F}_{\widehat{\rho}}$ in place of $\widehat{F}$. Plot your frequency of false rejections against sample size n. *Hint.* You should truncate $\widehat{\rho}$ so that $\widehat{\rho} \geq 0$, as this will keep the problem better-conditioned.

Question 2.5 (Clumpy testing errors)**:** In the data file `abalone.data` we have data on abalone (a type of mollusc) age, where the dataset is explained in file `abalone.names`. The goal is to predict the age of an abalone (given by the count of rings in its shell) from other characteristics. Here we use this dataset to investigate false discoveries and whether they come alone or in groups by adding additional complete noise variables to the data, then regressing a linear model including these noise variables.

Write code to perform the following: first, load the abalone data. Then

i. Add two columns (call them $x^{(1)}$ and $x^{(2)}$, say) to the data, where their entries are i.i.d.

$$\begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \end{bmatrix} \sim \mathsf{N}\left(0, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right),$$

that is, i.i.d. normal random variables with correlation $\rho \in (-1, 1)$.

ii. Fit a linear model for the response $y = \texttt{Rings}$ against all other variables (including the noise variables $x^{(1)}$ and $x^{(2)}$)

iii. Perform a t-test for association of variable $x^{(1)}$ (adjusting for all other variables) and $x^{(2)}$ (again, adjusting for all other variables) with y, rejecting at the level $\alpha = .05$.

For the values $\rho \in \{-.9, -.8, -.4, 0, .4, .8, .9\}$, repeat the experiment in steps i–iii $N = 1000$ times. Across the experiments, record the number of times there is a false discovery of $x^{(1)}$, a false discovery for $x^{(2)}$, and a false discovery of both simultaneously. Report your false discovery counts and describe them. (Include your code in your solution.)

Hints and pointers. You will want to represent the abalone's sex as a factor, that is, instead of the raw character `M`, `F`, or `I` (infant), represent it in a 0-1 encoding over 3 levels. That is, if $S \in \{\texttt{M}, \texttt{F}, \texttt{I}\}$ represents the sex of the abolone, transform it into

$$\phi(S) = \begin{bmatrix} 1\{S = \texttt{M}\} \\ 1\{S = \texttt{F}\} \\ 1\{S = \texttt{I}\} \end{bmatrix} \in \{0, 1\}^3.$$

In `R` this is achieved by using the method `factor`. Also, the t-test in step iii is simply the standard t-test we have developed in class and is that performed by `R`'s `summary` method of a linear model.

Question 2.6: A frequent question in optimization, machine learning, and other computational disciplines that develop methodology is to decide when a method is better than other methods. We consider this as follows. We have m algorithms, $i = 1, \ldots, m$, whose performance we wish to evaluate on solving one of p different problems—we have a test suite we believe covers enough problem space—where we can generate random instances of each of our problems.

As a motivating example, consider empirical risk minimization algorithms, which solve

$$\underset{\theta}{\text{minimize}} \ \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; x_i, y_i)$$

where $\theta \in \mathbb{R}^d$ is a parameter of interest, ℓ is a loss, and (x_i, y_i) are observations. Given p different datasets and loss functions—say, for regression—one can generate realizations by taking random subsamples of each of the datasets, performing the optimization algorithm, and tracking the time it takes to find a solution. In Figure 1, we copy a figure from a typical (and highly cited) machine learning paper developing a new optimization method. In the figure, the authors compare the performance of $m = 5$ methods, citing performance on one problem as evidence for their method, by plotting what are typically called “training curves” (these show the error of an optimization method versus time). Such curves often form the dubious basis for claims that an algorithm is “good” or “better than others.”

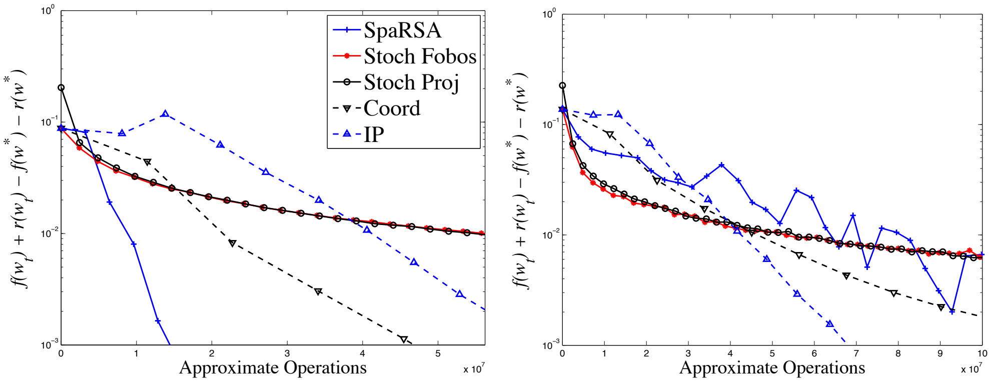

Figure 1. The performance of five methods (Sparse Reconstruction by Separable Approximation, Stochastic Forward/backward splitting (Fobos), Stochastic Projected gradient descent, Coordinate descent, and an Interior Point method) on a regularized regression optimization problem. Left: uncorrelated data. Right: highly correlated data.

(a) Give a few reasons why Figure 1 is unsatisfactory for declaring a method, e.g., SpaRSA, good. What does it actually show?

As alternatives to such training curves, we consider a few models for comparing algorithmic performance. The first is a two way ANOVA model without interaction. Let R_{ijk} denote the *runtime* of algorithm i in solving problem j for replication k, where we replicate each experiment n times (i.e. $k \in \{1, \ldots, n\}$). The basic ANOVA model without interaction is

$$R_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}, \tag{2.3}$$

for $i \in [m]$, $j \in [p]$, $k \in [n]$. Let $N = npm$ be the total number of measurements. Here $\varepsilon_{ijk} \sim (0, \sigma^2)$ independently, α_i represents algorithm i's quality, and β_j represents the difficulty of problem j.

(b) Give a reason that model (2.3) may be too naive when R_{ijk} represents runtime.

(c) Give the transformation $\phi : \mathbb{R}_+ \to \mathbb{R}$ so that if we set $Y_{ijk} = \phi(R_{ijk})$, the ANOVA model

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk} \tag{2.4}$$

has the semantics that adding 1 to α_i or to β_j doubles the runtime on instance (i, j, k).

Consider fitting model (2.4) by least squares. Define the usual sample mean shorthands

$$\overline{Y}_{\cdots} = \frac{1}{N} \sum_{i,j,k} Y_{ijk}, \quad \overline{Y}_{i\cdot\cdot} = \frac{1}{np} \sum_{j=1}^{p} \sum_{k=1}^{n} Y_{ijk}, \quad \overline{Y}_{\cdot j \cdot} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{k=1}^{n} Y_{ijk},$$

and consider the problem to

$$\underset{\mu,\alpha,\beta}{\text{minimize}} \sum_{i,j,k} (Y_{ijk} - \mu - \alpha_i - \beta_j)^2. \tag{2.5}$$

(d) Show that a solution to problem (2.5) is

$$\widehat{\mu} = \overline{Y}_{...}, \quad \widehat{\alpha}_i = \overline{Y}_{i\cdot\cdot} - \widehat{\mu}, \quad \widehat{\beta}_j = \overline{Y}_{\cdot j \cdot} - \widehat{\mu},$$

noting that this guarantees $\mathbf{1}^T\widehat{\alpha} = \mathbf{1}^T\widehat{\beta} = 0$.

Let us call algorithm $i = 1$ the algorithm of interest (e.g., the one we have invented in our ML paper). We would like to test the following $m - 1$ null hypotheses:

$$H_{0,i}: \quad \alpha_1 \geq \alpha_i, \tag{2.6}$$

where $\varepsilon_{ijk} \overset{\text{iid}}{\sim} \mathsf{N}(0, \sigma^2)$ for β, σ^2 unknown.

(e) Fix a level $a \in (0, 1)$. Give a level a test for $H_{0,i}$ by giving a test statistic and when to reject it. *Note.* If the inequality in the null (2.6) throws you off—meaning this is a composite null—then feel free to solve this problem by making the null $H_{0,i} : \alpha_1 = \alpha_i$. *Hint.* In our solution, we develop a t-test, using that $\widehat{\alpha}_1 - \widehat{\alpha}_i = \overline{Y}_{1\cdot\cdot} - \overline{Y}_{i\cdot\cdot} \sim \mathsf{N}(\alpha_1 - \alpha_i, \frac{2\sigma^2}{np})$ and $S^2 := \frac{1}{N-mp}\sum_{i,j,k}(Y_{ijk} - \overline{Y}_{ij\cdot})^2 \sim \frac{\sigma^2}{N-mp} \cdot \chi^2_{(N-mp)}$ are independent under the null.

We now turn to an alternative parameterization: we may assume we know nothing about the model of runtimes of the algorithms, but a natural null distribution is that for a pair of algorithms 1 and i, on an instance of a given problem each is equally likely to be faster than the other. Thus, we identify the null hypotheses that algorithm 1 is unlikely to be faster than algorithm i:

$$H'_{0,i}: \quad \mathbb{P}(R_{1jk} \leq R_{ijk}) \leq \frac{1}{2} \quad \text{for all } j \in [p], k \in [n], \tag{2.7}$$

and the events $\{R_{1jk} \leq R_{ijk}\}$ are independent.

(f) Show that if the null $H_{0,i}$ holds then $H'_{0,i}$ holds, so that $H'_{0,i}$ is a more general null.

(g) Fix a level $a \in (0, 1)$. Give a level a test for $H'_{0,i}$. *Note.* If the inequality in the null (2.7) throws you off—meaning this is a composite null—then feel free to solve this problem by making the null $H'_{0,i} : \mathbb{P}(R_{1jk} \leq R_{ijk}) = \frac{1}{2}$. *Hint.* You may use that if $B_{jk} \in \{0, 1\}$ are independent Bernoulli random variables, $j = 1, \ldots, p$, $k = 1, \ldots, n$, where for $\rho \in [0, 1]^p$ we define $\mathbb{P}_\rho(B_{jk} = 1) = \rho_j$, then for any threshold $t \in \mathbb{R}$,

$$\mathbb{P}_\rho\left(\sum_{j=1}^{p}\sum_{k=1}^{n} B_{jk} \geq t\right) \geq \mathbb{P}_{\rho'}\left(\sum_{j=1}^{p}\sum_{k=1}^{n} B_{jk} \geq t\right)$$

whenever $\rho_j \geq \rho'_j$ for each $j = 1, \ldots, p$. Define $B_{jk} = 1\{R_{1jk} \leq R_{ijk}\}$.

(h) The dataset `runtimes.csv` contains runtimes for several algorithms (named algorithms `A`, `B`, and so on) across multiple problems (labeled as integers) and data realizations. Note: these data are *runtimes* R, not transformed Y. Implement the tests you develop in parts (e) and (g) and apply them to the data. Report the p-values for each of the hypotheses $H_{0,i}$ and $H'_{0,i}$. In one or two sentences, describe whether you believe algorithm 1 to be the best algorithm.

As an aside to this question, the problem of comparing algorithmic performance in optimization, statistics, and other areas is quite interesting and important. One well-regarded technique in this area are *performance profiles*; see, e.g., the paper [4].

3 Nonparametric problems

Question 3.1 (Effective degrees of freedom)**:** In class, we said that an estimator $\widehat{f}$ making predictions $\widehat{Y}_i = \widehat{f}(x_i)$ from a linear model $Y = X\beta + \varepsilon$, $\varepsilon \sim (0, \sigma^2 I)$, had *effective degrees of freedom*

$$\mathsf{dof}(\widehat{f}) := \frac{1}{\sigma^2} \sum_{i=1}^n \mathrm{Cov}(\widehat{Y}_i, Y_i) = \mathbb{E}[(Y - \mathbb{E}[Y])^T \widehat{Y}].$$

Treating X as a fixed design, compute the degrees of freedom for the following estimators:

(a) The principal components regression estimator using r components. Recall that this estimator is as follows (here, we shall assume X is standardized so that $X^T\mathbf{1} = \mathbf{0}$ and $n^{-1}\sum_{i=1}^n X_{ij}^2 = 1$ for each $j \in \{1, \ldots, d\}$): we compute the principal components of variation in X, project the rows of X onto these, and perform ordinary least squares on the resulting low rank matrix. If $X = U\Sigma V^T$ is the SVD of X, then if $U_r = [u_1 \ \cdots \ u_r]$, $\Sigma_r = \mathrm{diag}(s_1, \ldots, s_r)$, $V_r = [v_1 \ \cdots \ v_r] \in \mathbb{R}^{d\times r}$, then this is equivalent to setting

$$\widehat{\beta}_{\mathrm{pcr}(r)} = \mathop{\mathrm{argmin}}_b \left\{ \left\| Y - XV_rV_r^Tb \right\|_2^2 \right\},$$

or

$$\widehat{\gamma}_r = \mathop{\mathrm{argmin}}_{\gamma\in\mathbb{R}^r} \left\{ \left\| Y - U_r\Sigma_r\gamma \right\|_2^2 \right\}, \quad \widehat{\beta}_{\mathrm{pcr}(r)} = V_r\widehat{\gamma}_r.$$

(b) The ridge regression estimator, where

$$\widehat{\beta}_\lambda = \mathop{\mathrm{argmin}}_b \left\{ \|Xb - Y\|_2^2 + \lambda \|b\|_2^2 \right\}.$$

(In all cases, $\widehat{Y} = X\widehat{\beta}$ for each estimator.)

Question 3.2 (Kernel machines)**:** *Kernel regression* is essentially a generalization of locally weighted linear regression and allows us to apply regression ideas to any space; we will consider trying to predict responses $y \in \mathbb{R}$ given $x \in \mathcal{X}$, where $\mathcal{X}$ is some space. One is given a *kernel function* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, which is a symmetric function with the property that the *Gram matrix* defined by

$$G_{ij} = k(x_i, x_j), \quad G \in \mathbb{R}^{n\times n}$$

is always positive semidefinite for any collection $\{x_1, \ldots, x_n\}$ of vectors. Some typical choices of kernels include

i. the *radial basis function* or *Gaussian* kernel, which for a bandwidth $\tau > 0$ is

$$k(x, z) = \exp\left(-\frac{\|x - z\|_2^2}{2\tau^2} \right),$$

ii. *polynomial kernels*, which for a degree d are given for vectors $x, z \in \mathbb{R}^p$ by

$$k(x, z) = (1 + x^Tz)^d,$$

iii. *the min-kernel* for data $x \in [0, 1]$, given by

$$k(x, z) = \min\{x, z\}.$$

One typically thinks of k as measuring something like similarity between its inputs: large values of k correspond to “close” vectors (though, as evidenced by the min kernel, this need not be precisely the case).

Consider the case of approximating $y = f(x) + \varepsilon$, $\varepsilon \sim (0, \sigma^2)$. Kernel regression approximates f by

$$h(x) = \sum_{i=1}^{n} k(x, x_i)\alpha_i. \tag{3.1}$$

We define the norm of such a function h by

$$\|h\|^2 := \alpha^T G \alpha$$

when G is the Gram matrix.[2] *Kernel ridge regression* then chooses $\widehat{f}$ by minimizing

$$\sum_{i=1}^{n} (y_i - h(x_i))^2 + \lambda \|h\|^2 \tag{3.2}$$

over h of the form (3.1), so $\widehat{f}(x) = \sum_{i=1}^{n} k(x, x_i)\widehat{\alpha}_i$ for some estimated $\widehat{\alpha} \in \mathbb{R}^n$.

(a) Explain in words why kernel regression makes sense, and why the resulting function $\widehat{f}$ is smooth as a function of its input x whenever the kernel function k is.

(b) Give an explicit formula for the coefficients $\widehat{\alpha}$ minimizing the squared error (3.2). Is your solution unique even when G is low rank?

(c) Let $\widehat{y}_i = \widehat{f}(x_i)$ for your estimator above. If $y = f(x) + \varepsilon$ for $\varepsilon \sim (0, \sigma^2)$ independent and mean zero, compute the effective degrees of freedom

$$\mathsf{dof}(\widehat{f}) = \frac{1}{\sigma^2} \sum_{i=1}^{n} \mathrm{Cov}(\widehat{y}_i, y_i).$$

(d) For the Gaussian kernel in item i, give the limits of the effective degrees of freedom $\mathsf{dof}(\widehat{f})$ as (i) $\tau^2 \uparrow \infty$ and (ii) $\tau^2 \downarrow 0$, assuming the x_i are all distinct. Interpret your results: if $\lambda \approx 0$ and $\tau \approx 0$, what are your predictions $\widehat{y}_i$?

(e) Let $\boldsymbol{f} = [f(x_1) \ \cdots \ f(x_n)]^T$ be the vector with entries $f(x_i)$, and let $\widehat{y}$ be the vector of predicted values. Show that for small regularization $\lambda > 0$, the (in-sample) bias satisfies

$$\boldsymbol{f} - \mathbb{E}[\widehat{y}] = \lambda G^{-1} \boldsymbol{f} + O(\lambda^2)$$

assuming that G is invertible, where $O(\lambda^2)$ indicates an error term that tends to zero as fast as λ^2. You may use that if A is a positive definite matrix, then $(A + \Delta)^{-1} = A^{-1} + \sum_{k=1}^{\infty} (-1)^k (A^{-1}\Delta)^k A^{-1}$ whenever $\|\Delta\|_{\mathrm{op}} < \lambda_{\min}(A)$.

(f) By question (e), for small $\lambda > 0$ the ridge regression estimator is (nearly) unbiased. Use this to argue that the expected residual sum of squares can give a natural variance estimate as follows. Let $H_\lambda = G(G + \lambda I)^{-1}$. Show that

$$\mathbb{E}[\mathrm{RSS}] = \mathbb{E}[\|\widehat{y} - y\|_2^2] = \sigma^2 \left(n - 2 \cdot \mathsf{dof}(\widehat{f}) + \mathrm{tr}(H_\lambda^2)\right) + O(\lambda^2)$$

[2] We attach a small appendix to this exercise set that is *completely* optional but may be of some interest to those of you who are analytically inclined; see Appendix A.

for small λ. Use this and part (c) to argue that

$$\widehat{\sigma}^2 := \frac{1}{n - 2\mathsf{dof}(\widehat{f}) + \mathrm{tr}(H_\lambda^2)} \left\|\widehat{y} - y\right\|_2^2$$

is a "reasonable" estimate of σ^2 when λ is small.

As a small aside, a careful calculation shows that $n - 2\mathsf{dof}(\widehat{f}) + \mathrm{tr}(H_\lambda^2) = O(\lambda^2)$ when λ is small, so sometimes a more involved approach is warranted. In this case, if we assume that G is full rank—which will be the case, for example, when one uses the Gaussian kernel $k(x, z) = \exp(-\frac{1}{2\tau^2}\left\|x - z\right\|_2^2)$—we can write $\boldsymbol{f} = G\alpha^\star$ for some $\alpha^\star \in \mathbb{R}^n$. Then for small λ we have

$$\mathbb{E}[\mathrm{RSS}] = \mathbb{E}[\|\widehat{y} - y\|_2^2] = \sigma^2 \left(n - 2 \cdot \mathsf{dof}(\widehat{f}) + \mathrm{tr}(H_\lambda^2)\right) + \lambda^2 \left\|\alpha^\star\right\|_2^2 + O(\lambda^3).$$

Thus, if we can estimate $\left\|\alpha^\star\right\|_2^2$, we can get some further corrections; a reasonable heuristic here is to substitute the $\left\|\cdot\right\|_2$-norm of $G^{-1}\widehat{y}$ for $\alpha^\star$, then use the estimate

$$\widehat{\sigma}^2 := \frac{1}{n - 2\mathsf{dof}(\widehat{f}) + \mathrm{tr}(H_\lambda^2)} \left[\left\|\widehat{y} - y\right\|_2^2 - \lambda^2 \left\|G^{-1}\widehat{y}\right\|_2^2\right].$$

(Sometimes this may be non-positive, and so further corrections may be necessary.)

Question 3.3 (Implementing and experimenting with kernel ridge regression)**:** In this question, you will use the data available in `lprostate.dat` from the paper [6] to investigate and implement various versions of kernel ridge regression. In this dataset, the goal is to predict the log prostate specific antigen `lpsa` from other prostate measurements.

(a) Implement a function `predictKRR` that takes as input five quantities:

- `X`, a $d \times n$ data matrix representing the training data with rows x_i^T
- `Z`, a $d \times m$ matrix of m data points $z_1, \ldots, z_m$ on which to make new predictions
- `alpha`, an n vector representing the vector $\widehat{\alpha}$ fit in Part (b) of Question 3.2
- `tau`, the bandwidth parameter τ for the Gaussian (RBF) kernel $k(x, z) = \exp(-\frac{\|x-z\|_2^2}{2\tau^2})$
- `offset`, an offset parameter $b \in \mathbb{R}$

The function should then output a vector of predictions $\widehat{y}(z_j) = b + \sum_{i=1}^n k(x_i, z_j)\alpha_i$, where k is the Gaussian kernel with the given bandwidth, $k(x, z) = \exp(-\frac{\|x-z\|_2^2}{2\tau^2})$

(b) Implement a method `fitKRR` that takes as input

- `X`, a $d \times n$ data matrix representing the training data with rows x_i^T
- `y`, an n vector of responses
- `lambda`, a positive scalar representing regularizatin
- `tau`, the bandwidth parameter τ

then returns the pair (`alpha`, `yMean`), where `alpha` is the solution $\widehat{\alpha}$ to the kernel ridge regression problem (3.2) with the Gaussian kernel where we replace y with $y - \mathbf{1}\overline{y}$, the *centered* y (here $\overline{y} = \frac{1}{n}\sum_{i=1}^n y_i$), and `yMean` is the mean response.

(c) Perform kernel ridge regression on the data in `lprostate.dat`, using `lpsa` as the response and use `lcavol` as the *only* covariate x. You should standardize the data in x first so that the (sample) variance of x is 1. Fix $\tau = .1$ and vary $\lambda \in \{.01, .5, 5\}$. Plot the resulting fitted predictions $\widehat{y}$ superimposed on a scatter plot of `lcavol` against `lpsa`. Describe what you see in words.

(d) Repeat an identical experiment to that above, but use $\tau = .5$.

(e) Now we use *all* the covariates to predict `lpsa`. Perform kernel ridge regression using `lprostate.dat`. Standardize the data (excepting `lpsa`), and for $\tau = .5$, fit a kernel ridge regression for $\lambda \in \{.01, .5, 5\}$. Plot the resulting fitted predictions $\widehat{y}$ superimposed on a scatter plot of `lcavol` against `lpsa`. Describe what you see in words.

(f) Find the estimate $\widehat{\sigma}^2$ defined in part (f) of question 3.2 using $\tau = .5$ and $\lambda = .1$, using all covariates (standardized) as in the preceding part. (Recall as in class that estimating $\widehat{\sigma}$ with a "more powerful" model is often a good way to estimate the smallest irreducible error.) Now, perform kernel ridge regression as in part (d) using only `lcavol` as the covariate with $\tau = .5$ and $\lambda = .5$. Estimate the standard error of the fit for each predicted value $\widehat{y}_i$, and superimpose your predictions $\widehat{y}$ with 2× standard error bands on a scatter of `lcavol` and `lpsa`.

Question 3.4: Consider the observation model

$$y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \overset{\text{iid}}{\sim} (0, \sigma^2), \tag{3.3}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is $f(x_i) = \mathbb{E}[y_i \mid x_i]$. Let $X \in \mathbb{R}^{n \times d}$, $n \geq d$, be a full-rank data matrix with singular value decomposition $X = U\Gamma V^T$, so $U \in \mathbb{R}^{n \times d}$, $V \in \mathbb{R}^{d \times d}$, and $U^T U = I_d = V^T V = V V^T$. We use the notation $U = [u_1 \ \cdots \ u_d]$ and $V = [v_1 \ \cdots \ v_d]$. We consider a principal-components ridge regression estimator

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

where $\Lambda = \operatorname{diag}(\lambda) = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$ is a positive semidefinite matrix (so $\lambda_j \geq 0$ for all j, i.e. the vector $\lambda \in \mathbb{R}^d_+$). Here the idea is that we can penalize the scale of β in the directions of the various principal components in a more intelligent way than naive ridge regression, perhaps shrinking components where we have less "information" more aggressively than others.

Define

$$\widehat{y}_\lambda = X\widehat{\beta}_\lambda = H_\lambda y, \quad H_\lambda = X(X^T X + V\Lambda V^T)^{-1} X^T.$$

(a) Show that $\widehat{\beta}_\lambda = V\Gamma(\Gamma^2 + \Lambda)^{-1} U^T y$ and $H_\lambda = U\Gamma^2(\Gamma^2 + \Lambda)^{-1} U^T$.

Recall the in-sample risk

$$R_{\text{in}}(\widehat{\beta}_\lambda) := \frac{1}{n} \sum_{i=1}^n \mathbb{E}[(\widehat{y}_i - f(x_i))^2] = \frac{1}{n} \mathbb{E}\left[\|\widehat{y}_\lambda - \mathbb{E}[y]\|_2^2 \right].$$

(b) Show that the residual sum of squares satisfies

$$\frac{1}{n} \mathbb{E}\left[\|\widehat{y}_\lambda - y\|_2^2 \right] = R_{\text{in}}(\widehat{\beta}_\lambda) + \sigma^2 - \frac{2}{n} \sum_{i=1}^n \operatorname{Cov}(\widehat{y}_i, y_i).$$

Conclude that

$$\frac{1}{n}\mathbb{E}\left[\|\widehat{y}_\lambda - y\|_2^2\right] + \frac{2\sigma^2}{n}\operatorname{tr}(H_\lambda) = R_{\text{in}}(\widehat{\beta}_\lambda) + \sigma^2.$$

(*Note:* we have basically done this in class. It is not a trick question.)

By part (b), you have shown the fact (as proved in class) that residual sum of squares plus a trace is unbiased for in-sample risk, and so in particular if $\widehat{\sigma}^2$ is an unbiased estimate for σ^2 in the model (3.3), then

$$r(\lambda) := \frac{1}{n}\|\widehat{y}_\lambda - y\|_2^2 + \frac{2\widehat{\sigma}^2}{n}\operatorname{tr}(H_\lambda)$$

is unbiased for $R_{\text{in}}(\widehat{\beta}_\lambda) + \sigma^2$ (note that $\lambda \in \mathbb{R}_+^d$ is a vector). We consider using this unbiased estimate to choose a $\lambda \in \mathbb{R}^d$ to obtain—we hope—better predictions than a naive ridge regression estimate would provide. We shall assume that we have a reasonable estimate $\widehat{\sigma}^2$.

(c) Let $U_\perp \in \mathbb{R}^{n\times(n-d)}$ be an orthogonal basis for $\operatorname{span}(U)^\perp = \{v \in \mathbb{R}^n \mid U^T v = \mathbf{0}\}$, i.e., any matrix so that $I_n = UU^T + U_\perp U_\perp^T$ and $U_\perp^T U = \mathbf{0}$. Show that

$$r(\lambda) = \frac{1}{n}\sum_{j=1}^d \left[\frac{\lambda_j^2}{(\gamma_j^2+\lambda_j)^2}\left(u_j^T y\right)^2 + 2\widehat{\sigma}^2\frac{\gamma_j^2}{\gamma_j^2+\lambda_j}\right] + \frac{1}{n}\left\|U_\perp^T y\right\|_2^2.$$

Conclude that

$$\begin{aligned}\frac{\partial}{\partial\lambda_j} r(\lambda)\cdot\frac{n}{2} &= \frac{\lambda_j}{\gamma_j^2+\lambda_j}\left[\frac{\gamma_j^2}{(\gamma_j^2+\lambda_j)^2}\right](u_j^T y)^2 - \widehat{\sigma}^2\frac{\gamma_j^2}{(\gamma_j^2+\lambda_j)^2} \\ &= \frac{\gamma_j^2}{(\gamma_j^2+\lambda_j)^2}\left[\frac{\lambda_j}{\gamma_j^2+\lambda_j}(u_j^T y)^2 - \widehat{\sigma}^2\right].\end{aligned}$$

(d) We now consider minimizing $r(\lambda)$ over $\lambda \ge 0$, i.e., elementwise $\lambda_j \ge 0$. Using the result of part (c), argue that $\frac{\partial}{\partial\lambda_j}r(\lambda) < 0$ for all $\lambda \ge 0$ whenever $\widehat{\sigma}^2 \ge (u_j^T y)^2$. Then conclude that the $\lambda^\star$ minimizing $r(\lambda)$ satisfies

$$\lambda_j^\star = \begin{cases} +\infty & \text{if } \widehat{\sigma}^2 \ge (u_j^T y)^2 \\ \dfrac{\widehat{\sigma}^2\gamma_j^2}{(u_j^T y)^2 - \widehat{\sigma}^2} & \text{otherwise.}\end{cases} \tag{3.4}$$

Interpret this in one or two sentences.

(e) Now you will perform some comparisons between your "tuned" principal components ridge regression with penalties $\lambda^\star \in \mathbb{R}_+^d$ as in (3.4) and a standard ridge estimator. Using the data in `lprostate.dat`, you will run ridge (or this optimized ridge) regression with `lpsa` as a response, performing the following experiment 25 times:

i. Load the data in `lprostate.dat`: split the data into a training set $(X_{\text{train}}, y_{\text{train}})$ containing a random .6 proportion of the data and test set $(X_{\text{test}}, y_{\text{test}})$ containing the remaining data.

ii. Standardize the data so that y_{train} is mean zero and so each column of data matrix X_{train} has mean zero and variance 1. Apply the same normalization to X_{test} and y_{test}. (Note that you should use the transformation you apply to the *training* data.)

iii. Fit the "optimal" ridge estimator $\widehat{\beta}_{\lambda^\star}$ using the $\lambda^\star$ in Eq. (3.4) and compute the held-out risk $\widehat{r}_\star = \frac{1}{n_{\text{test}}} \|X_{\text{test}} \widehat{\beta}_{\lambda^\star} - y_{\text{test}}\|_2^2$, where n_{test} is the sample size of the test data. You should use the usual estimate $\widehat{\sigma}^2 = \frac{1}{n_{\text{train}} - d} \min_b \|X_{\text{train}} b - y_{\text{train}}\|_2^2$

iv. For each $\tau = 10^{i/10}$, $i = -10, -9, \ldots, 20$, fit a standard ridge regression estimate

$$\widehat{\beta}_\tau = (X_{\text{train}}^T X_{\text{train}} + \tau I)^{-1} X_{\text{train}}^T y_{\text{train}}$$

and compute the held-out risk $\widehat{r}_\tau = \frac{1}{n_{\text{test}}} \|X_{\text{test}} \widehat{\beta}_\tau - y_{\text{test}}\|_2^2$

Then plot the average gap $\widehat{r}_\tau - \widehat{r}_\star$ over the 25 experiments as a function of τ, where your horizontal axis should correspond to τ on a logarithmic scale. We have written code in the file `ridge-prostate-dataprep.*` that will perform steps i–ii for you.

Give one potential explanation for what you see in a sentence or two.

4 Validation and validity

Question 4.1 (Leave-one-out solutions in M-estimation)**:** Consider the M-estimation (empirical risk minimization) problem:

$$\underset{b}{\text{minimize}} \; L_n(b) := \frac{1}{n}\sum_{i=1}^n \ell(y_i - x_i^T b)$$

where ℓ is twice-continuously differentiable, convex, and symmetric. In this problem, we will develop a method to efficiently *approximate* the minimizers of the leave-one-out ℓ_2 (ridge)-regularized objective in cross validation by solving a sequence of quadratic problems; even more, our method will involve only performing rank-one updates to a matrix inverse.[3] Let $\widehat{\beta} = \text{argmin}_b\{L_n(b) + \frac{\lambda}{2}\left\|b\right\|_2^2\}$ be the minimizer of the empirical risk (the M-estimator), and for $k = 1, 2, \ldots, n$, define the leave-one-out objective

$$L_{-k}(b) := \frac{1}{n}\sum_{i \neq k} \ell(y_i - x_i^T b).$$

Note that both L_n and L_{-k} are convex functions (bowl-shaped).

Consider finding minimizers of $L_{-k}(b)$, equivalently, finding the perturbation to $\widehat{\beta}$:

$$\Delta_k := \underset{\Delta}{\text{argmin}}\left\{L_{-k}(\widehat{\beta} + \Delta) + \frac{\lambda}{2}\|\widehat{\beta} + \Delta\|_2^2\right\}.$$

In this problem, you will show that this is *very* well-approximated by a simpler quadratic minimizer (at least under appropriate conditions). By Taylor's theorem, for $\Delta \in \mathbb{R}^d$ we have

$$L_{-k}(\widehat{\beta} + \Delta) = L_{-k}(\widehat{\beta}) + \nabla L_{-k}(\widehat{\beta})^T\Delta + \frac{1}{2}\Delta^T\nabla^2 L_{-k}(\beta_\Delta)\Delta$$

for some β_Δ between $\widehat{\beta}$ and $\widehat{\beta} + \Delta$. Define the gradients and Hessian matrices

$$g_k := \nabla L_{-k}(\widehat{\beta}) + \lambda\widehat{\beta}, \quad H = \nabla^2 L_n(\widehat{\beta}) + \lambda I, \quad H_k = \nabla^2 L_{-k}(\widehat{\beta}) + \lambda I,$$

so that if we define the empirical errors $\widehat{\varepsilon}_i = y_i - x_i^T\widehat{\beta}$, these have the explicit forms

$$g_k = -\frac{1}{n}\sum_{i \neq k}\ell'(\widehat{\varepsilon}_i)x_i + \lambda\widehat{\beta} = \frac{1}{n}\ell'(\widehat{\varepsilon}_k)x_k,$$

$$H = \frac{1}{n}\sum_{i=1}^n \ell''(\widehat{\varepsilon}_i)x_i x_i^T + \lambda I, \qquad H_k = H - \frac{1}{n}\ell''(\widehat{\varepsilon}_i)x_i x_i^T.$$

(To simplify g_k, we used that $\nabla L_n(\widehat{\beta}) + \lambda\widehat{\beta} = \mathbf{0}$.) Then

$$L_{-k}(\widehat{\beta} + \Delta) + \frac{\lambda}{2}\|\widehat{\beta} + \Delta\|_2^2 \approx L_{-k}(\widehat{\beta}) + g_k^T\Delta + \frac{1}{2}\Delta^T H_k\Delta.$$

We define $\widehat{\Delta}_k$ to be the minimizer of this quadratic approximation to $L_{-k} + \frac{\lambda}{2}\left\|\cdot\right\|_2^2$ around $\widehat{\beta}$,

$$\widehat{\Delta}_k = \underset{\Delta}{\text{argmin}}\left\{L_{-k}(\widehat{\beta}) + g_k^T\Delta + \frac{1}{2}\Delta H_k\Delta\right\} = -H_k^{-1}g_k.$$

[3] In Question 4.2 (which is extra credit), we work through an argument making approximations here rigorous and showing that, in fact, we lose very little by our computationally efficient approach.

(a) Assume you have computed H^{-1}. Show how to compute $H_k^{-1} g_k$ efficiently, that is, without recomputing the full inverse of H_k. (It is sufficient to simply give the formula.)

(b) Let $\widehat{y}_{-k} = x_k^T(\widehat{\beta} + \widehat{\Delta}_k)$ be the (approximate) leave-one-out prediction. Show that $\widehat{y}_{-k} = \widehat{y}_k - x_k^T H_k^{-1} g_k$ and that $\widehat{\varepsilon}_{-k} = y_k - \widehat{y}_{-k} = \widehat{\varepsilon}_k + x_k^T H_k^{-1} g_k$. Using these identities, show how to compute the vector $[\widehat{y}_{-k}]_{k=1}^n$ in time $O(nd^2)$, assuming that you have $\widehat{\beta}$ already.

(c) Implement the (approximate) leave-one-out cross validation procedure above to choose the level λ of ridge regularization when solving the M-estimation problem

$$\underset{b}{\text{minimize}} \;\; L_n(b) + \frac{\lambda}{2} \|b\|_2^2 , \tag{4.1}$$

where $\ell(t) = \log(1 + e^t) + \log(1 + e^{-t})$. Use your procedure to evaluate the (approximate) leave-one-out (LOO) validation error for the randomly generated data in the file `generate-outlier-data.*`.[4] Note that the actual *error* you should be plotting is

$$\text{LOO} = \frac{1}{n} \sum_{i=1}^n \ell(\widehat{\varepsilon}_{-i}).$$

Plot your LOO error against λ for 25 logarithmically spaced values of λ from .01 to 10 (your horizontal axis should be on the log scale as well). Include your code in your solution.

(d) Choose the $\widehat{\lambda}$ minimizing the LOO error, and let $\widehat{\beta}$ be the minimizer of the ℓ_2-regularized objective (4.1) on the training data. Evaluate its median absolute prediction error on the held-out test set generated by `generate.data` (call this ERR_{test}). For 25 values of λ logarithmically spaced between 2 and 50, let

$$\widehat{\beta}_\lambda^{\text{ls}} = \underset{b}{\text{argmin}} \left\{ \frac{1}{2n} \|Xb - y\|_2^2 + \frac{\lambda}{2} \|b\|_2^2 \right\}$$

be the ℓ_2-regularized least-squares solution, and let $\text{ERR}_{\text{test}}^{\text{ls}(\lambda)}$ be its median absolute prediction error on the held-out test set. Plot $\text{ERR}_{\text{test}}^{\text{ls}(\lambda)} - \text{ERR}_{\text{test}}$ against λ. What do you observe?

(e) Repeat the same experiment as in part (d), except that you should vary the number of outliers (see the method `generate.data`) in $\{0, 5, 10, 15, 20, 25\}$. Plot the same results as above. What do you observe?

Question 4.2 (Mathematically challenging): We revisit the setting in Question 4.1, but in this variant, we shall develop a perturbation guarantee to show that our heuristic of using the second-order approximation is actually quite accurate. Roughly, you will show that if the Hessian $H = \nabla^2 L_n(\widehat{\beta})$ is positive definite (a relatively easy condition to satisfy), then

$$\max_{k \le n} \|\Delta_k - \widehat{\Delta}_k\| = O\left(\frac{1}{n^2}\right)$$

[4] This file provides a method for minimizing the robust loss with ridge regularization, which you should use to find your initial estimate of $\widehat{\beta}$ To run it, you will need to install CVXPY or CVXR, which solve convex optimization problems.

for each $k \in [n]$.[5] That is, the difference between the "true" leave-one-out minimizer and the approximation is of much smaller order than any statistical/sampling error.

Unfortunately, to do this fully rigorously requires nontrivial setup and a bit of analysis. We review the tools and assumptions here. For simplicity through this derivation, we assume the first and second derivatives satisfy the boundedness conditions

$$\sup_{t \in \mathbb{R}} \left|\ell'(t)\right| \le 1, \quad \sup_{t \in \mathbb{R}} \left|\ell''(t)\right| \le 1, \quad \left|\ell''(t) - \ell''(s)\right| \le |t - s|, \quad \text{all } t, s \in \mathbb{R}.$$

We also assume that the covariate vectors $x_i \in \mathbb{R}^d$ satisfy $\|x_i\|_2 \le \mathsf{D}_x$ for all i. Let $X = [x_1 \ \cdots \ x_n]^T \in \mathbb{R}^{n \times d}$ be the usual design matrix.

A few useful inequalities and notational simplifications follow. For symmetric matrices A, B, we say $A \succeq B$ if $A - B$ is positive semidefinite, that is, $\lambda_{\min}(A - B) \ge 0$. We also have Weyl's inequality, that is, $|\lambda_i(A + B) - \lambda_i(A)| \le |\!|\!| B |\!|\!|_{\text{op}}$ for any symmetric A, B, where $|\!|\!| B |\!|\!|_{\text{op}} = \sup_{\|v\|_2 = 1} \|Bv\|_2$ is the usual operator norm. A useful convexity inequality is the following: if f is a convex function, and for some $\lambda, c > 0$ its second derivative satisfies $\nabla^2 f(b) \succeq \lambda I$ for all $b \in \mathbb{R}^d$ satisfying $\|b - \beta\|_2 \le c$, then

$$f(b) \ge f(\beta) + \nabla f(\beta)^T (b - \beta) + \frac{\lambda}{2} \min\left\{c, \|b - \beta\|_2\right\} \|b - \beta\|_2 . \tag{4.2}$$

The rough proof outline is the following: first, we show that the Hessian $\nabla^2 L_{-k}$ is positive definite, so that L_{-k} has reasonable growth away from $\widehat{\beta}$. This implies that the minimizer Δ_k cannot be too large, as otherwise, $L_{-k}(\widehat{\beta} + \Delta) > L_{-k}(\widehat{\beta})$. Once we have this, then we can perform a more careful second-order Taylor approximation of $L_{-k}(\widehat{\beta} + \Delta)$, solving directly.

(a) Show that

$$\nabla^2 L_n(\widehat{\beta} + \Delta) \succeq \nabla^2 L_n(\widehat{\beta}) - \mathsf{D}_x \|\Delta\|_2 \frac{1}{n} X^T X.$$

Conclude that if that $H = \nabla^2 L_n(\widehat{\beta}) \succeq 2\lambda I$, then whenever $\|\Delta\|_2 \le \lambda / (2\mathsf{D}_x |\!|\!| n^{-1} X^T X |\!|\!|_{\text{op}})$,

$$\nabla^2 L_n(\widehat{\beta} + \Delta) \succeq \frac{3\lambda}{2} I.$$

(b) Show that if $H = \nabla^2 L_n(\widehat{\beta}) \succeq 2\lambda I$, then

$$\nabla^2 L_{-k}(\widehat{\beta} + \Delta) \succeq \lambda I$$

when $\|\Delta\|_2 \le \lambda / (2\mathsf{D}_x |\!|\!| n^{-1} X^T X |\!|\!|_{\text{op}})$ and n is large enough that $\frac{2\mathsf{D}_x^2}{n} \le \lambda$, i.e., $n \ge \frac{2\mathsf{D}_x^2}{\lambda}$.

(c) As before assume $H = \nabla^2 L_n(\widehat{\beta}) \succeq 2\lambda I$. Assume that n is large enough that

$$n \ge \frac{4\mathsf{D}_x^2 |\!|\!| n^{-1} X^T X |\!|\!|_{\text{op}}}{\lambda^2}.$$

Argue that the minimizer

$$\Delta_k := \operatorname*{argmin}_{\Delta} L_{-k}(\widehat{\beta} + \Delta)$$

satisfies $\|\Delta_k\|_2 \le \frac{2\mathsf{D}_x}{\lambda n}$.

[5] A similar argument is precisely what shows that robust M-estimators are indeed robust.

(d) Assume the conditions in parts (a)–(c). Show that under these,

$$\|\widehat{\Delta}_k - \Delta_k\|_2 = O\left(\frac{1}{n^2}\right).$$

In our answer, we obtain

$$\|\widehat{\Delta}_k - \Delta_k\|_2 \le \frac{2\mathsf{D}_x^5}{\lambda^2 - 2\mathsf{D}_x^4/n} \cdot \frac{1}{n^2}.$$

Hint: Perform a Taylor approximation, recognizing that $\mathbf{0} = \nabla L_{-k}(\widehat{\beta}+\Delta_k) \approx g_k + H_k\Delta_k$. The following bound on matrix inverses may be useful: if A is positive definite and the error matrix E satisfies $\|E\|_{\rm op} \le \lambda_{\min}(A)$, then $\left\|(A+E)^{-1} - A^{-1}\right\|_{\rm op} \le \frac{\|E\|_{\rm op}}{\lambda_{\min}(A) - \|E\|_{\rm op}}$.[6]

Question 4.3 (Making valid predictions)**:** In this question, you will develop a method that performs predictive inference—it guarantees (a type of) validity for confidence intervals for future responses Y based on observed data. The starting point is to revisit permutation-based p-values as in class; reviewing Section B below may be useful. The goal in this question is to develop a confidence set mapping

$$\widehat{C}_n : \mathcal{X} \to \mathcal{I},$$

where $\mathcal{I}$ denotes the collection of all intervals in $\mathbb{R}$,

$$\mathcal{I} = \{[a,b] \text{ s.t. } a < b \in \mathbb{R}\}.$$

For a given $\alpha > 0$, we would like to construct such a confidence interval based on observed data $(X_i, Y_i) \in \mathcal{X} \times \mathbb{R}$, $i = 1, \ldots, n$, which provides the coverage guarantee that

$$\mathbb{P}\left(Y_{n+1} \in \widehat{C}_n(X_{n+1})\right) \ge 1 - \alpha,$$

no matter the data generating distribution, as long as the data are i.i.d.

The starting point is an extension of Lemma B.1 below. To provide the extension we require a bit of notation. Let $Z_1, \ldots, Z_n, Z_{n+1} \in \mathbb{R}$ be exchangeable random variables. Define

$$F_n(t) := \frac{1}{n}\sum_{i=1}^n \mathbf{1}\{Z_i \le t\}$$

and the empirical quantile of the first n observations (not including $n+1$) by

$$\widehat{Q}_n(\alpha) = F_n^{-1}(\alpha) := \inf\{t \in \mathbb{R} \mid F_n(t) \ge \alpha\}.$$

Let $Z_{(1,n)} \le Z_{(2,n)} \le \cdots \le Z_{(n,n)}$ denote the order statistics of $Z_1, \ldots, Z_n$ and $Z_{(i,n+1)}$ the order statistics of $Z_1, \ldots, Z_{n+1}$, so that $\widehat{Q}_n(\alpha) = Z_{(\lceil n\alpha \rceil, n)}$.

(a) Show that for any $k \in \mathbb{N}$, we have

$$Z_{n+1} \le Z_{(k,n)} \text{ if and only if } Z_{n+1} \le Z_{(k,n+1)}.$$

[6] This identity follows from the equality $(A+E)^{-1} = A^{-1} + \sum_{i=1}^\infty (-1)^i (A^{-1}E)^i A^{-1}$.

(b) Conclude from Lemma B.1 and part (a) that a slightly inflated quantile provides the following guarantees:

$$\mathbb{P}\left[Z_{n+1} \le \widehat{Q}_n\left(\left(1+\frac{1}{n}\right)\alpha\right)\right] \ge \alpha.$$

Now consider the following conformalization approach, known as *split conformal inference.* We assume we have two samples,

$$(X_{\text{train}}, Y_{\text{train}}) \in \mathbb{R}^{n_{\text{train}} \times d} \times \mathbb{R}^{n_{\text{train}}} \quad \text{and} \quad (X, Y) \in \mathbb{R}^{n \times d} \times \mathbb{R}^n,$$

where $X_{\text{train}}, Y_{\text{train}}$ are the n_{train} *training data points* and (X, Y) are the n validation data points, both consisting of i.i.d. (x, y) pairs. We use $(X_{\text{train}}, Y_{\text{train}})$ to fit a prediction model, call it $\widehat{f}(x)$, and we will represent the confidence set as

$$\widehat{C}(x) := \left\{y \in \mathbb{R} \text{ s.t. } |y - \widehat{f}(x)| \le \tau\right\} = \left[\widehat{f}(x) - \tau, \widehat{f}(x) + \tau\right], \tag{4.3}$$

where τ is a threshold to be chosen based on the validation data.

(c) For each (x_i, y_i) pair in the validation data, define the score

$$Z_i := \left|y_i - \widehat{f}(x_i)\right|$$

and for an $\alpha \in (0,1)$, let $\widehat{\tau}_n = \widehat{Q}_n(\frac{n+1}{n}(1-\alpha))$ be the $\frac{n+1}{n}(1-\alpha)$ quantile of $\{Z_i\}_{i=1}^n$. Suppose $(X_{n+1}, Y_{n+1}) \in \mathbb{R}^d \times \mathbb{R}$ is drawn from the same distribution as the *validation* data (x_i, y_i). Show that for $Z_{n+1} = |Y_{n+1} - \widehat{f}(X_{n+1})|$,

$$\mathbb{P}(Z_{n+1} > \widehat{\tau}_n) \le \alpha.$$

Hint. This should be trivial from part (b).

(d) Using the preceding result in part (c), conclude that if we take $\tau = \widehat{\tau}_n$ in the confidence sets (4.3), then

$$\mathbb{P}\left(Y_{n+1} \in \widehat{C}(X_{n+1})\right) \ge 1 - \alpha.$$

As an alternative to the symmetric sets (4.3), we could instead use quantile regression to construct confidence sets. Suppose that we fit lower and upper prediction models via quantile regression, so that for the loss

$$\ell_\alpha(t) = \alpha \cdot (t)_+ + (1-\alpha) \cdot (-t)_+ = \alpha \cdot \max\{t, 0\} + (1-\alpha)\max\{-t, 0\}$$

and a collection of prediction functions $\mathcal{F}$, for $0 < \delta_{\text{low}} < \delta_{\text{high}} < 1$ we define the δ_{low}-lower and δ_{high}-upper quantile predictors

$$\begin{aligned}
\widehat{q}_{\delta_{\text{low}}} &= \operatorname*{argmin}_{f \in \mathcal{F}} \sum_{(x,y) \in (X_{\text{train}}, Y_{\text{train}})} \ell_{\delta_{\text{low}}}(y - f(x)) \\
\widehat{q}_{\delta_{\text{high}}} &= \operatorname*{argmin}_{f \in \mathcal{F}} \sum_{(x,y) \in (X_{\text{train}}, Y_{\text{train}})} \ell_{\delta_{\text{high}}}(y - f(x)).
\end{aligned}$$

(e) For each (x_i, y_i) pair in the *validation* data, define the score

$$Z_i := \max\left\{\widehat{q}_{\delta_{\text{low}}}(x_i) - y_i, y_i - \widehat{q}_{\delta_{\text{high}}}(x_i)\right\},$$

and set $\widehat{\tau}_n = \widehat{Q}_n(\frac{n+1}{n}(1-\alpha))$ be the $\frac{n+1}{n}(1-\alpha)$ quantile of $\{Z_i\}_{i=1}^n$. Define the confidence set

$$\widehat{C}_\delta(x) := \left[\widehat{q}_{\delta_{\text{low}}}(x) - \widehat{\tau}_n, \widehat{q}_{\delta_{\text{high}}}(x) + \widehat{\tau}_n\right].$$

Show that if (X_{n+1}, Y_{n+1}) is drawn from the same distribution as the validation data, then

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

(f) In the final part of the question, you will perform two experiments comparing different kernel regression models (recall the previous études) for constructing confidence sets. You will use the data in `generate-heteroskedastic-sin.*` and methods in `kernel-quantile-fitting.*` to perform the following conformalization procedures. We use a Gaussian kernel with $k(x,z) = \exp(-\frac{1}{2\tau^2}\|x-z\|_2^2)$, recalling the Gram matrix of a sample $\{x_i\}_{i=1}^n$ with entries $G_{ij} = k(x_i, x_j)$, $G \in \mathbb{R}^{n\times n}$. We have three datasets: $(X_{\text{train}}, y_{\text{train}})$, $(X_{\text{valid}}, y_{\text{valid}})$, and $(X_{\text{test}}, y_{\text{test}})$, each i.i.d. Using the training data, you should fit three models using the Gram matrix G from the *training* data:

$$\begin{aligned}
\widehat{\beta}_{\text{low}} &= \operatorname*{argmin}_b \left\{\frac{1}{n}\sum_{i=1}^n \ell_{\delta_{\text{low}}}(y_i - G_i^T b) + \frac{\lambda}{2} b^T G b\right\} \\
\widehat{\beta}_{\text{high}} &= \operatorname*{argmin}_b \left\{\frac{1}{n}\sum_{i=1}^n \ell_{\delta_{\text{high}}}(y_i - G_i^T b) + \frac{\lambda}{2} b^T G b\right\} \\
\widehat{\beta}_{\text{ridge}} &= \operatorname*{argmin}_b \left\{\|Gb - y\|_2^2 + \lambda b^T G b\right\}
\end{aligned}$$

where $\delta_{\text{low}} = .1$, $\delta_{\text{high}} = .9$, $y = y_{\text{train}}$, $\tau = .1$, and $\lambda = .01$. The methods `fitQuantileKernel` and `fitKernelRidge` perform these tasks for you.

Given any of these three vectors $\beta \in \mathbb{R}^n$, define the function $f_\beta : \mathbb{R} \to \mathbb{R}$ by

$$f_\beta(x) := \sum_{i=1}^n k(x_i, x)\beta_i,$$

where the x_i belong to the training data. The method `predictKRR` provides an evaluation of this function on any given collection of new input points x.

- For the ridge regression estimate $\widehat{\beta}_{\text{ridge}}$, implement the procedure in parts (c) and (d) by computing the errors $|f_{\widehat{\beta}}(x) - y|$ for (x, y) in the validation data.
- For the low/high estimates $\widehat{\beta}_{\text{low}}$ and $\widehat{\beta}_{\text{high}}$, construct calibrated lower/upper predictions using the procedure in part (e) (note that $\widehat{q}_{\delta_{\text{low}}} = f_{\widehat{\beta}_{\text{low}}}$ and similarly for $\widehat{q}_{\delta_{\text{high}}}$).

For each of these, use $\alpha = .1$, and plot the resulting confidence bands on the values Y_i in the *test data*; the method `plotUpperAndLower` will be useful for this. Report the coverage rate (fraction of data Y_i in the test set satisfying $Y_i \in \widehat{C}(X_i)$).

A Reproducing kernel Hilbert spaces

A symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is *positive semidefinite* if for any $n \in \mathbb{N}$ and collection of points $x_1, \ldots, x_n$, the Gram matrix $G \in \mathbb{R}^{n \times n}$ with entries $G_{ij} = k(x_i, x_j)$ is positive semidefinite. Any such k gives rise to a Hilbert space $\mathcal{H}$ of functions from $\mathcal{X} \to \mathbb{R}$ for which k is the *reproducing kernel* (sometimes called the *representer of evaluation*), meaning that for any $h \in \mathcal{H}$, we have

$$\langle h, k(x, \cdot) \rangle = h(x),$$

as follows. For any finite n, m, and $x_i, z_i \in \mathcal{X}$, we define the inner product between $h_0(x) = \sum_{i=1}^n \alpha_i k(x_i, x)$ and $h_1(x) = \sum_{i=1}^m \beta_i k(z_i, x)$ by

$$\langle h_0, h_1 \rangle = \sum_{i=1}^n \sum_{j=1}^m \alpha_i \beta_j k(x_i, z_j).$$

It is evident that if $h(\cdot) = \sum_{i=1}^n \alpha_i k(x_i, \cdot)$, then $\langle h, k(x, \cdot) \rangle = h(x)$. By taking $\mathcal{H}$ to be the completion of this space with the given inner product, we then have a complete inner product space with norm $\|h\|^2 = \langle h, h \rangle$. If h has the form $h(x) = \sum_{i=1}^n k(x, x_i) \alpha_i$ and $G = [k(x_i, x_j)]_{i,j} \in \mathbb{R}^{n \times n}$ is the Gram matrix, then the norm evidently satisfies

$$\|h\|^2 = \langle h, h \rangle = \alpha^T G \alpha.$$

Note that our construction of $\mathcal{H}$ as the completion via the reproducing kernel k shows that for any such positive semidefinite function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, there exists a feature mapping $\phi : \mathcal{X} \to \mathcal{H}$ such that

$$k(x, z) = \langle \phi(x), \phi(z) \rangle, \tag{A.1}$$

so that k can represent a (potentially) infinite dimensional inner product. Additionally, for any feature mapping $\phi(x)$, the function *defined* by $k_\phi(x, z) = \langle \phi(x), \phi(z) \rangle$ evidently is positive semidefinite and gives rise to a Hilbert space in the same way as above.

The representer theorem

We can then frame the optimization problem (3.2) in a more general way: consider the problem to

$$\underset{h \in \mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^n (y_i - h(x_i))^2 + \lambda \|h\|^2, \tag{A.2}$$

where we just let $h \in \mathcal{H}$. Now, we can write any $h \in \mathcal{H}$ as $h(x) = \sum_{i=1}^n \alpha_i k(x_i, x) + g(x)$ where $g \in \mathcal{H}$, and $\langle g, k(x_i, \cdot) \rangle = 0$ for each i by orthogonalizing. In this case,

$$\|h\|^2 = \alpha^T G \alpha + \|g\|^2,$$

while $h(x_i) = \sum_{j=1}^n \alpha_i k(x_j, x_i) + \langle k(x_i, \cdot), g \rangle = \sum_{j=1}^n \alpha_i k(x_j, x_i)$. Thus including such a non-zero g can only increase $\|h\|^2$ without modifying $h(x_i)$, and the minimizer of problem (A.2) is necessarily of the form

$$h(x) = \sum_{i=1}^n \alpha_i k(x_i, x).$$

This result is called the *representer theorem.* There are whole books devoted to the topic; see, for example, [7, 3, 1]. A major theme in this line of work is to develop efficient and effective kernel functions k for problems of interest.

Minimizing squared error with a feature mapping

The representer theorem, coupled with the feature mapping (A.1), give an alternative interpretation of kernel ridge regression (A.2): we can instead consider the problem

$$\underset{\beta}{\text{minimize}} \sum_{i=1}^{n} (y_i - \langle \phi(x_i), \beta \rangle)^2 + \lambda \|\beta\|^2 \tag{A.3}$$

(where β is in the same space as $\phi(x_i)$). Then by the representer theorem, for any $\widehat{\beta}$ minimizing this objective there is some $\alpha \in \mathbb{R}^n$ such that

$$\langle \widehat{\beta}, \phi(x) \rangle = \sum_{i=1}^{n} \alpha_i k(x, x_i),$$

and so the problem (A.3) is equivalent to kernel ridge regression (A.2), and so also to problem (3.2). As a side benefit to this perspective, we see how to include an intercept term in problem (3.2) with kernel k: let ϕ be the feature mapping associated to the kernel, so $k(x, z) = \langle \phi(x), \phi(z) \rangle$. Then evidently, replacing ϕ with

$$\phi_{\text{int}}(x) = \begin{bmatrix} 1 \\ \phi(x) \end{bmatrix}$$

gives rise to the kernel function $k_{\text{int}}(x, z) = 1 + \langle \phi(x), \phi(z) \rangle = 1 + k(x, z)$. Then instead of solving problem (3.2), we solve

$$\underset{\alpha \in \mathbb{R}^n}{\text{minimize}} \left\| y - \alpha^T (\mathbf{1}\mathbf{1}^T + G)\alpha \right\|_2^2 + \lambda \alpha^T G \alpha,$$

where $G = [k(x_i, x_j)]_{i,j}$ is the usual Gram matrix for k.

B Quantiles and permutations

Here we review/restate some of the results on permutations we claimed in class. Real-valued random variables $Z_1, Z_2, \ldots, Z_n \in \mathbb{R}$ are *exchangeable* if the distribution of $(Z_1, \ldots, Z_n)$ is identical to the distribution of $(Z_{\pi(1)}, \ldots, Z_{\pi(n)})$ for all permutations $\pi : [n] \to [n]$. Recall also that we defined the empirical CDF of a sample $Z_1, \ldots, Z_n$ by

$$F_n(t) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{Z_i \le t\},$$

which is continuous from the right. The *inverse CDF* or *quantile function* is

$$F_n^{-1}(\alpha) := \inf \{t \in \mathbb{R} \mid F_n(t) \ge \alpha\}.$$

Note that if we sort the values in the sample $Z_1^n = (Z_1, \ldots, Z_n)$, i.e. into order statistics

$$Z_{(1)} \le Z_{(2)} \le \cdots \le Z_{(n)},$$

then

$$F_n^{-1}(\alpha) = Z_{(\lceil \alpha n \rceil)}.$$

For shorthand, we often write $\widehat{q}_n(\alpha) := F_n^{-1}(\alpha)$ for the *quantile function.*

We then have the result that forms the basis of all of our permutation tests (and similar).

Lemma B.1 (The p-value for an exchangeable sample)**.** *Assume that $Z_1^n = (Z_1, \ldots, Z_n)$ are exchangeable. Then*

$$\mathbb{P}(Z_n \le \widehat{q}_n(\alpha)) \ge \alpha.$$

If additionally the Z_i are distinct with probability 1, then

$$\mathbb{P}(Z_n \le \widehat{q}_n(\alpha)) \le \alpha + \frac{1}{n}.$$

An immediate consequence of Lemma B.1 is the equivalent statement that

$$\mathbb{P}\left(Z_n > \widehat{q}_n(1-\alpha)\right) \ge \alpha \tag{B.1a}$$

always, and whenever the Z_i are distinct with probability 1,

$$\mathbb{P}\left(Z_n > \widehat{q}_n(1-\alpha)\right) \le \alpha + \frac{1}{n}. \tag{B.1b}$$

Proof We note that

$$F_n(\widehat{q}_n(\alpha)) = F_n(F_n^{-1}(\alpha)) \ge \alpha$$

always, as F_n is continuous from the right. Then we note that $Z_n \le \widehat{q}_n(\alpha)$ by exchangeability, we have

$$\mathbb{P}(Z_n \le \widehat{q}_n(\alpha)) = \mathbb{P}(Z_i \le \widehat{q}_n(\alpha))$$

for each i, and so

$$\alpha \le \mathbb{E}\left[F_n(\widehat{q}_n(\alpha))\right] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\mathbf{1}\left\{Z_i \le \widehat{q}_n(\alpha)\right\}\right] = \mathbb{E}\left[\mathbf{1}\left\{Z_n \le \widehat{q}_n(\alpha)\right\}\right] = \mathbb{P}(Z_n \le \widehat{q}_n(\alpha)).$$

When the Z_i are distinct with probability 1, the "jumps" in F_n are at most $1/n$, that is, for any $t_0 \in \mathbb{R}$

$$0 \le F_n(t_0) - \lim_{t \uparrow t_0} F_n(t) \le \frac{1}{n}.$$

Thus (with probability 1) $F_n(\widehat{q}_n(\alpha)) \le \alpha + \frac{1}{n}$, and we still have $\mathbb{P}(Z_n \le \widehat{q}_n(\alpha)) = \mathbb{E}[F_n(\widehat{q}_n(\alpha))]$ as above. □

References

[1] A. Berlinet and C. Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer Academic Publishers, 2004.

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

[3] N. Cristianini and J. Shawe-Taylor. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[4] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213, 2002.

[5] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer, second edition, 2009.

[6] T. A. Stamey, J. N. Kabalin, J. E. Mcneal, I. M. Johnstone, F. Freiha, E. A. Redwine, and N. Yang. Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate: Ii. radical prostatectomy treated patients. *Journal of Urology*, 141(5): 1076–1083, 1989.

[7] G. Wahba. *Spline Models for Observational Data.* Society for Industrial and Applied Mathematics, Philadelphia, 1990.