553.740 Project 3; Penalized Regression.
Fall 2024
Due on Friday, November 1st. 0

Please read carefully the directions below.

- **You must type your answers (using, e.g., LateX, Pages or MS Word) and return them in a pdf file.** *A five-point penalty out of a maximum of 100 will be applied otherwise. The .pdf file must be complete and contain the answers to the questions below (including figures).*
- *A solution in the form of a program output, or of a Jupyter notebook output, is not acceptable.*
- **You must provide your program sources.** *They must be added as a separate* **.zip file**. *They will not be graded (so no direct credit for them), but not providing them will result in a zero score in questions that use them.*

  The program sources will be useful in order to understand why results are not correct (and decide whether partial credit can be given) and to ensure that your work is original.

  You may use any programming language, although Python is strongly recommended. A Jupyter notebook is also acceptable as a program source.
- **Late return policy.** *Papers received after November 1st and before or on November 8th will have 5 points subtracted from the grade. No paper will be accepted after that extra week.*
- *This homework must be solved individually, possibly with the help of the teaching assistants or the instructor. Collaboration between students is not allowed. Neither is allowed the use of AI tools such as ChatGPT.*
- *The data used in this project is stored in csv format and is readable using, for example, the python pandas package.*

Question 1.

We consider a multivariate regression model with input $X : \Omega \to \mathbb{R}^d$, output $Y : \Omega \to \mathbb{R}^q$, $\beta_0 \in \mathbb{R}^q$ and b a $d \times q$ matrix, with predictor $f : \mathbb{R}^d \to \mathbb{R}^q$ given by:

$$f(x) = a_0 + b^T x.$$

This model will be trained using a variant of ridge regression, minimizing

$$F(a_0, b) = \sum_{k=1}^{N} |y_k - a_0 - b^T x_k|^2 + \lambda \text{trace}(\beta D \beta^T),$$

where D is a positive semi-definite $q \times q$ symmetric matrix and $\beta = \begin{pmatrix} a_0^T \\ b \end{pmatrix}$.

(1.1) Prove that the optimal solution $\hat{\beta}$ must satisfy the *Sylvester equation*

$$\mathcal{X}^T \mathcal{X} \beta + \lambda \beta D = \mathcal{X}^T \mathcal{Y}$$

with

$$\mathcal{Y} = \begin{pmatrix} y_1^T \\ \vdots \\ y_N^T \end{pmatrix}, \quad \mathcal{X} = \begin{pmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_N^T \end{pmatrix}$$

(1.2) Write a program that computes the solution of the multivariate problem in question (1.1), taking, as input, the array **Y**, containing each y_k as row vectors, the array **X**, containing each x_k as row vectors, the parameter λ and the penalty matrix D. The program will returning the optimal a_0 and b. (Note: you can use the *solve_sylvester* function in the scipy linear algebra package to solve the Sylvester equation.)

Test this program with the dataset provided in "project3_F2024_1.csv," for which $d = 3$ and $q = 200$, with $\lambda = 10$, $D = \mathrm{Id}$. Plot the values of $a_0(j)$, $b(1,j)$, $b(2,j)$ and $b(3,j)$ as functions of j (on the same chart).

(1.3) Provide the same illustration using the same dataset but with $\lambda = 1000$, and D a tridiagonal matrix with -1 above and below the diagonal, and 2 on the diagonal, except $D(1,1) = D(q,q) = 1$.

Question 2.

Write a program that computes the kernel version of ridge regression in two cases

1. Gaussian kernel $K(x,y) = \exp(-|x-y|^2/2\sigma^2)$.
2. Polynomial kernel of order h and width σ: $K(x,y) = \sum_{k=1}^{h} (x^T y)^k / \sigma^{2k-2}$.

The function should take as input an (N,d) array, **X**, an $(N,1)$ array **Y**, the choice of kernel and its parameter(s) and the penalty coefficient λ of kernel ridge regression.

The files "project3_F2024_2_Train.csv" and "project3_F2024_2_Test.csv" are training and test data for this question. They contain samples of a 10-dimensional variable **X** and a one-dimensional **Y**, with respectively $N = 250$ samples (training) and $M = 1000$ samples (test).

(2.1) Use your program with a Gaussian kernel, taking $\sigma = 2.5$ and $\lambda = 0.001, 0.002, \ldots, 0.1$. Use training data to estimate the parameters and, for each value of λ, compute the residual sum of squares on the test set. Provide in your answer:

(i) A plot of the RSS as a function of λ.

(ii) The minimum value of the RSS.

(iii) The value of λ that minimizes the RSS.

(2.2) Provide the same answers for polynomial kernels with $h = 1, 2, 3, 5, 8$ and $\sigma = 1.5$. Use $\lambda = 1, 2, \ldots, 100$ in this question.

Question 3.

(3.1) Fixing $\epsilon > 0$, let

$$V_\epsilon(t) = \begin{cases} \dfrac{t^2}{2\epsilon} \text{ if } |t| \le \epsilon \\ |t| - \dfrac{\epsilon}{2} \text{ if } |t| \ge \epsilon. \end{cases}$$

(a) Taking $\epsilon = 0.1$, plot $V_\epsilon(t)$ as a function of t for $t \in [-1, 1]$

(b) Check that V_ϵ is C^1 and compute its derivative with respect to t.

(c) Show that $V_\epsilon : \mathbb{R} \to \mathbb{R}$ is a convex function.

(d) Let q be a positive integer. Prove that $g_\epsilon : \mathbb{R}^q \to \mathbb{R}$ defined by $g_\epsilon(z) = V_\epsilon(|z|)$ is also convex.

(3.2) Fixing $w \in \mathbb{R}^q$ and $\gamma > 0$, let

$$f(z) = V_\epsilon(|z|) + \frac{1}{2\gamma}|z - w|^2, z \in \mathbb{R}^q.$$

Compute $\arg\min\{f(z) : z \in \mathbb{R}^q\}$ as a function of w and γ.

(3.3) Assume that a training set $(x_1, y_1, \ldots, x_N, y_N)$ is observed with $y_k \in \mathbb{R}^q$ and $x_k \in \mathbb{R}^d$, and fix $\rho, \epsilon > 0$. Define the function

$$F(a_0, b) = |b|^2 + \rho \sum_{k=1}^{N} g_\epsilon(y_k - a_0 - b^T x_k)$$

defined on $\mathbb{R}^q \times \mathcal{M}_{d,q}(\mathbb{R})$.

Introducing auxiliary variables $(z_1, \ldots, z_N)$, we reformulate the minimization of F as that of

$$G(\mathcal{Z}, a_0, b) = |b|^2 + \rho \sum_{k=1}^{N} g_\epsilon(z_k)$$

subject to $z_k = y_k - a_0 - x_k^T b$.

(a) Introduce the data matrices

$$\mathcal{Y} = \begin{pmatrix} y_1^T \\ \vdots \\ y_N^T \end{pmatrix} \in \mathcal{M}_{N,q}(\mathbb{R}), \quad \mathcal{X} = \begin{pmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_N^T \end{pmatrix} \in \mathcal{M}_{N,d}(\mathbb{R}),$$

and the variables

$$\mathcal{Z} = \begin{pmatrix} z_1^T \\ \vdots \\ z_N^T \end{pmatrix} \in \mathcal{M}_{N,q}(\mathbb{R}), \quad \beta = \begin{pmatrix} a_0^T \\ v \end{pmatrix} \in \mathcal{M}_{d+1,q}(\mathbb{R}).$$

Prove that one can use the ADMM algorithm to solve this problem using the following iterations with $\alpha > 0$, in which $\mathcal{U} \in \mathcal{M}_{N,q}(\mathbb{R})$ is an auxiliary variable.

(A) Initialize the algorithm with $\mathcal{U}^{(0)} = \mathcal{Z}^{(0)} = \beta^{(0)} = 0$.

(B) At step n, given $\mathcal{U}^{(n)}, \mathcal{Z}^{(n)}$, let

(i) Let $\beta^{(n+1)} = (\mathcal{X}^T\mathcal{X} + \frac{2\Delta}{\alpha})^{-1}\mathcal{X}^T(\mathcal{Y} - \mathcal{Z}^{(n)} - \mathcal{U}^{(n)})$

(ii) For $k = 1, \ldots, N$, $z_k^{(n+1)} = \text{prox}_{\rho g_\epsilon/\alpha}(w_k - u_k^{(n)})$ where $\mathcal{W} = \begin{pmatrix} w_1^T \\ \vdots \\ w_N^T \end{pmatrix} = \mathcal{Y} - \mathcal{X}\beta^{(n+1)}$.

(iii) $\mathcal{U}^{(n+1)} = \mathcal{U}^{(n)} - \mathcal{W} + \mathcal{Z}^{(n+1)}$

(C) Stop when the difference between all coordinates of the updated and previous variables is less, in absolute value, than a tolerance number τ.

In this algorithm, $\Delta = \begin{pmatrix} 0 & 0 \\ 0 & \text{Id}_{\mathbb{R}^d} \end{pmatrix}$ and $\mathcal{X} = \begin{pmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_N^T \end{pmatrix}$.

(b) Provide a closed form expression for the proximal operator in step (B.ii).

(3.4) Program the previous algorithm, taking as input the matrices $\mathbf{X}$ and $\mathbf{Y}$, the values of ϵ and ρ, and returning the optimal a_0 and b. You will use a small tolerance constant $\tau = 10^{-10}$ and you can take $\alpha = 1$..

You will test your algorithm on the data provided in the file "project3_F2024_3.csv", for all pairs ϵ, ρ with $\epsilon \in \{0.1, 0.5, 1.0, 2.0\}$ and $\rho \in \{10, 5, 1, 0.1\}$. For each pair (ϵ, ρ), you will provide in your answer the values of:

(i) The number of iterations needed by the algorithm of question (3.3.a).

(ii) The residual sum of squares: $\sum_{k=1}^{N}(y_k - a_0 - b^T x_k)^2$.

(iii) The estimated parameters $a_0, b_{2,1}, b_{10,2}$ and $b_{22,3}$.