553.740 Project 1: Kernel Prediction and Cross-Validation.

Due on Monday February 25.

- **Please type your answers (using, e.g., LateX or MS Word.)**
- *The solution must be written with sufficient explanations and your results must be commented. Returning a series of numerical results and figures is not enough. A solution in the form of a program output is not acceptable either.*
- *Please return your program sources. They will not graded (so no direct credit for them), but they 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, Matlab or R are recommended.*
- *The ".csv" files associated with this project are available on Blackboard. In addition to the index column, they contain three columns, labeled X1, X2 (for a 2D variable X) and Y, and each row corresponds to a sample.*

Preamble. The Bayes estimator applied to probability density functions computed via kernel density estimation provides predictors called "kernel regression" or "kernel classification." This homework proposes to run some experiments with them. More details on the derivation of the predictors can be found in the lecture notes.

Assuming data in d dimensions, we will use

$$K(x) = \exp(-|x|^2/2)/(2\pi)^{d/2}$$

and $K_h(x) = K(x/h)/h^d$.

If X and Y are random variables taking values in $\mathbb{R}^d$ and $\mathbb{R}$ respectively, the kernel regression estimator, based on a training set $(x_1, y_1), \ldots, (x_N, y_N)$ is defined by

$$\hat{f}_h(x) = \frac{\sum_{k=1}^N y_k K_h(x - x_k)}{\sum_{k=1}^N K_h(x - x_k)}. \tag{1}$$

If Y takes value in a finite set $\mathcal{G}$ instead, one estimates the conditional density of X given $Y = g$ by

$$\hat{\psi}_h(x|g) = \frac{1}{N_g} \sum_{k=1}^{N} K_h(x - x_k) \chi_{y_k = g}.$$

Assuming that the prior distribution of all classes are equal, the posterior distribution of $Y = g$ given x is then

$$\pi_h(g|x) = \frac{\psi_h(x|g)}{\sum_{g' \in \mathcal{G}} \psi_h(x|g')} \tag{2}$$

The MAP estimator then is

$$\hat{f}_h(x) = \operatorname{argmax}\{\pi_h(g|x) : g \in \mathcal{G}\}. \tag{3}$$

Question 1.

Write a program which, given an (M, d) matrix $\mathcal{U}$, an (N, d) matrix $\mathcal{X}$, an N by 1 vector $\mathcal{Y}$ and a scalar parameter h, computes the kernel regression estimator learned from $\mathcal{X}$ and $\mathcal{Y}$ evaluated at $\mathcal{U}$, i.e., $\hat{f}_h(u_i)$, $i = 1, \ldots, M$ where the u_i are the rows of $\mathcal{U}$ and $\hat{f}_h$ is given by (1). The program should provide the output in the form of an M-dimensional vector.

Write also a program that, by calling the previous one with proper subsets of the training set, estimates the error using k-fold cross-validation. The program should take as input X, Y, h and k.

(1) You will run the following 1D experiments to illustrate these programs. Here X and Y are random variables with the following joint distribution: (i) X follows a uniform distribution over $[-1, 1]$ and (ii) the conditional distribution of Y given $X = x$ is Gaussian with mean $f(x)$ and standard deviation $\sigma^2 = 1$, taking $f(x) = 2x^3 - 0.5x + 1$.

1. Generate an M-sample $((x'_1, y'_1), \ldots, (x'_M, y'_M))$ of (X, Y), with $M = 1000$. This will be the testing data.
2. Generate a (deterministic) vector $\mathcal{U} = (u_1, \ldots, u_M)$ providing an evenly spaced discretization of the interval $[-1.1]$ in $M = 1000$ points.
3. For $N = 100$ and $N = 250$
   - Generate an N-sample $((x_1, y_1), \ldots, (x_N, y_N))$ of (X, Y). This data determines the kernel regression estimator $x \mapsto \hat{f}_h(x)$ in (1).
   - Then, for $h = 0.05, 0.10.0.25$ and 0.50:
     - Plot on a single frame the functions $(U, f(U))$, $(U, \hat{f}_h(U))$ and the sample points $(x_k, y_k), k = 1, \ldots, N$. (Using one frame for each pair (N, h).)
     - Evaluate and provide the values of

$$e_1 = \frac{1}{N} \sum_{k=1}^{N} (y_k - \hat{f}_h(x_k))^2,$$

$$e_2 = \frac{1}{M}\sum_{j=1}^{M}(f(u_j) - \hat{f}_h(u_j))^2$$

$$\text{and } e_3 = \frac{1}{M}\sum_{j=1}^{M}(y'_j - \hat{f}_h(x'_j))^2.$$

- Return the estimate of the error returned by 10-fold cross-validation.

(2) Explain why $E((Y - f_h(X))^2) \geq 1$. Which of e_1, e_2, e_3 estimates this quantity? (Justify your answer.)

Question 2.

In this question, we propose to *estimate* the kernel bandwidth, h, using cross-validation. You will use, for this, the datasets stored in the files "project1_S19_1_train.csv" (training set) and "project1_S19_1_test.csv" (test set) and perform the following operations.

1. Compute, for values of h uniformly spaced over the interval $[0.01, 1]$, the 10-fold cross-validation error $\mathcal{E}_{cv}(h)$ and plot $\mathcal{E}_{cv}(h)$ as a function of h.
2. Report the value of h that minimizes this error. Call it h_0.
3. Retrain $\hat{f}_{h_0}$ on the whole training set and provide its error evaluated on the test set.

Question 3.

We here make a similar analysis for a binary classification problem. Write a program which, given an (M, d) matrix $\mathcal{U}$, an (N, d) matrix $\mathcal{X}$, an N by 1 vector $\mathcal{Y}$ (consisting of zeros and ones) and a scalar parameter h, computes the kernel classification estimator learned from $\mathcal{X}$ and $\mathcal{Y}$ evaluated at U, i.e., $\hat{f}_h(u_i)$, $i = 1, \ldots, M$ in (3), where the u_i are the rows of U. The program should provide the output in the form of an M-dimensional binary vector and should also return the posterior probabilities in (2) (as an M by 2 matrix).

Write also a program that, by calling the previous one with proper subsets of the training set, estimates the error using k-fold cross-validation. The program should take as input $\mathcal{X}, \mathcal{Y}, h$ and the number of folds.

(1) Use the dataset "project1_S19_2_train.csv" as training data to estimate the bandwidth h using 10-fold cross-validation, where the error is the number of misclassifications divided by the sample size. Provide in your answer a plot of the cross-validation error evaluated as a function of h, the optimal value of

h, the misclassification error of the classifier evaluated on the whole training set, and the error of the same classifier evaluated on the test set provided in "project1_S19_2_test.csv." (This is similar to the previous question.)

(2) Slightly modify the cross-validation part of the previous experiment by replacing the error term used to evaluate, on a training subset T_2, the performance of a classifier trained on a training subset T_1 by the negative log-likelihood:

$$\mathcal{L}_{T_1,T_2} = -\frac{1}{|T_2|}\sum_{j\in T_2}\log \pi_{h,T_1}(y_j|x_j)$$

where π_{h,T_1} is given by (2) and trained on T_1.

Provide in your answer a plot of the cross-validation negative likelihood evaluated as a function of h, the optimal value of h, the misclassification error of the classifier evaluated on the whole training set, and the error of the same classifier evaluated on the test set.