STAT 6910-003 – SLDM II – Homework #5

Due: 5:00 PM 11/30/18

1. **Variance of correlated random variables (5 pts).** Suppose we have random variables $X_1, \ldots, X_B$ that are identically distributed but not independent. Assume that for each i $\mathbb{V}[X_i] = \sigma^2$ where $\mathbb{V}[X_i]$ denotes the variance of X_i. Furthermore, assume that each pair of random variables is positively correlated with correlation coefficient ρ. I.e., $corr(X_i, X_j) = \rho > 0$ for each $i \neq j$.

   (a) (3 pts) Prove that the variance of the sample mean is $\mathbb{V}\left[\frac{1}{B}\sum_{i=1}^{B} X_i\right] = \rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$. This provides motivation for selecting random features in the random forest algorithm.
   Hint: You may use the facts that $\mathbb{V}\left[\frac{1}{B}\sum_{i=1}^{B} X_i\right] = \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B} Cov(X_i, X_j)$ and $Cov(X_i, X_i) = \mathbb{V}[X_i]$.

   (b) (2 pts) Explain why it would not make sense (i.e. it isn't possible) for ρ to be negative when $B \geq 3$.

2. **Support Vector Regression (25 pts).** Support vector regression (SVR) is a method for regression analogous to the support vector classifier. Let $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$, $i = 1, \ldots, n$ be training data for a regression problem. In the case of linear regression, SVR solves

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

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

   (a) (5 pts) Show that for an appropriate choice of λ, SVR solves

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

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

   (b) (13 pts) The optimization problem is convex with affine contraints and therefore strong duality holds. Use the KKT conditions to derive the dual optimization problem in a manner analogous to the support vector classifier (SVC). As in the SVC, you should eliminate the dual variables corresponding to the constraints $\xi_i^+ \geq 0$, $\xi_i^- \geq 0$.

   (c) (4 pts) Explain how to kernelize SVR. Be sure to explain how to recover $\mathbf{w}^*$ and b^*.

   (d) (3 pts) Argue that the final predictor will only depend on a subset of training examples (i.e. support vectors) and characterize those training examples.

3. **Bagging and Logistic Regression (25 pts).** Download the `mnist_49_3000.mat` file from Canvas. This is the same handwritten digit dataset that was used in Homework 3. You will apply bagging to the logistic regression classifier on this dataset (you may use existing packages for logistic regression) and compare to other classifiers. Train using the first 2000 samples and test your classifier on the last 1000 samples.

(a) (20 pts) Report the test error for 1) Random forests (you may use existing software; report what package and any parameters you use), 2) SVM without bagging (you may use existing software; report what package and any parameters you use including those selected with cross-validation), 3) logistic regression without bagging, 4) bagging+logistic regression with 50 bootstrapped samples, and 5) bagging+logistic regression with 100 bootstrapped samples. Which classifier performs the best? Does bagging seem to help the logistic regression classifier in this case? Make sure to keep your trained models for part (b).

(b) (5 pts) Download the `mnist_49_1000_shifted.mat` file from Canvas. This is the same test data as before except each of the images have been shifted. Apply each of the six trained classifiers from part (a) to this new, shifted test data and report the test error. Which classifier performs the best? Does bagging seem to help the logistic regression classifier in this case? If you knew your test data were likely to be shifted but your training data wasn't, what training strategy or strategies could you employ to obtain better test results?
FYI, if the distribution of your test data differs from the distribution of your training data, then this problem is called *transfer learning* or *covariate shift*.

4. **Outlier Detection with Kernel Density Estimation (35 pts).** Download the `anomaly.mat` file from Canvas. This file has training data sampled from a univariate density f stored in the variable `X` and two test points `xtest1` and `xtest2`. The goal of this problem is to calculate a KDE of the density f using the training data and use the KDE to determine whether the two test points are outliers/anomalies.

(a) (5 pts) Using least squares leave-one-out cross-validation with a Gaussian kernel and the training data, select a value for the bandwidth parameter. Report the bandwidth parameter.

(b) (7 pts) Using the training data, estimate the density f at points uniformly spaced between -2 and 4 using a KDE with a Gaussian kernel and the bandwidth parameter selected in part (a). Include a plot of the KDE. Choose enough points between -2 and 4 so that the plot looks smooth. Based on the KDE, what do you think the true density f looks like?
Do not use packages that automatically compute the KDE but use the equations from class instead. You may use packages that calculate distances efficiently. If in doubt about a specific package, you can ask on Piazza.

(c) (4 pts) There are multiple approaches for using the KDE in anomaly detection. Here is one approach. Let N be the number of points in the training data. Let x_j, $j = 1, \ldots, N$ be the training data. Let x be a point that you wish to test. Define an outlier score for x as

$$OutlierScore_1(x) = \frac{\hat{f}_h(x)}{\frac{1}{n}\sum_{j=1}^{N} \hat{f}_h^{(-j)}(x_j)}.$$

See lecture slides for a definition of $\hat{f}_h^{(-j)}(x_j)$. Provide a conceptual interpretation of this score. I.e., would the score be low or high if x is an outlier? What if x is not an outlier? Are there any potential weaknesses of this score?

(d) (5 pts) Calculate the $OutlierScore_1$ for `xtest1` and `xtest2` using the same h you selected in part (a). Based on the calculated scores, do you think either of these points are outliers?

(e) (4 pts) $OutlierScore_1$ has some weaknesses. Let's define another score based on k-nearest neighbors. Let $\rho_k(x)$ be the distance of x to its kth nearest neighbor in the training data. Let $\mathcal{N}(x)$ be the set of k nearest neighbors of x in the training data. So $|\mathcal{N}(x)| = k$. Define another outlier score for x as

$$OutlierScore_2(x) = \frac{\rho_k(x)}{\frac{1}{k}\sum_{x_i \in \mathcal{N}(x)} \rho_k(x_i)},$$

where $\rho_k(x_i)$ is the distance of x_i to its kth nearest neighbor in the training data (not including x_i). Provde a conceptual interpretation of this score. I.e., would the score be low or high if x is an outlier? What if x is not an outlier? How is this score related to k-nn density estimation? What potential advantages would this score have over the one defined in part (c)?

(f) (5 pts) Calculate $OutlierScore_2$ for `xtest1` and `xtest2` using $k = 100$, 150, and 200. Based on the calculated scores, do you think either of these points are outliers?

(g) (5 pts) Include your code.

5. How long did this assignment take you? (5 pts)

6. Type up homework solutions (5 pts)