Statistical Machine Learning for Data Science
Assignment 2

Due Date: January 17, 2021 - By 11:59pm

All your answers must be written on a separate sheet, properly typeset and submitted in the form of a report, in pdf format. No MS Word report will be accepted. Make sure your report (in pdf format) is submitted by the deadline. No late report will accepted. Neatly handwritten reports that are clearly legible will also be accepted, but the parts involving computations must typeset properly. All graphs and plots must be properly captioned, labels and titled.

Exercise 1: (8 points each for a total of 56 points)

Consider a dataset $\mathcal{D} = \{(X_i, Y_i) \overset{iid}{\sim} p_{XY}(\mathbf{x}, \mathbf{y}), i = 1, \cdots, n\}$. Assume that $\mathcal{D}$ is randomly split into two sets namely a training set $\mathcal{D}_{\mathtt{tr}}$ and a test set $\mathcal{D}_{\mathtt{te}}$ such that $\mathcal{D} = \mathcal{D}_{\mathtt{tr}} \cup \mathcal{D}_{\mathtt{te}}$. Let

$$\widehat{\mathtt{R}}_{\mathtt{tr}}(\widehat{f}) = \frac{1}{|\mathcal{D}_{\mathtt{tr}}|} \sum_{i=1}^{n} \ell(Y_i, \widehat{f}(X_i)) \mathrm{S}_i^{\mathtt{tr}} = \frac{1}{|\mathcal{D}_{\mathtt{tr}}|} \sum_{i=1}^{n} \mathbb{1}(Y_i \neq \widehat{f}(X_i)) \mathrm{S}_i^{\mathtt{tr}}$$

where $\mathrm{S}_i^{\mathtt{tr}} = \mathbb{1}((X_i, Y_i) \in \mathcal{D}_{\mathtt{tr}})$. Now, consider the following predictions yielded by $\widehat{f}$ on a test set, along with the truth responses

$\mathbf{y}$	-1	1	1	-1	-1	1	-1	1	-1	-1	1	1	-1	-1	1	1
$\widehat{f}(\mathbf{x})$	-1	-1	1	1	-1	1	-1	1	-1	-1	1	1	-1	-1	1	1

Table 1: Predictions yielded by $\widehat{f}$ on a test set, along with the true responses

1. Compute $\widehat{\mathtt{PCC}}_{\mathtt{te}}(\widehat{f})$, the estimated probability of correct predictions yielded by $\widehat{f}$, and write down the theoretical expression of what is being estimated.
2. Deduce $\widehat{\mathtt{R}}_{\mathtt{te}}(\widehat{f})$, and write down the theoretical expression of what is being estimated.
3. Generate $\mathtt{M}_{\mathtt{te}}$, the confusing matrix of $\widehat{f}$.
4. Comment on the meaning of $\frac{\mathtt{trace}(\mathtt{M}_{\mathtt{te}})}{|\mathcal{D}_{\mathtt{te}}|}$
5. Compute $\widehat{\mathtt{TPR}}_{\mathtt{te}}(\widehat{f})$, then explain its meaning and write down its theoretical expression.
6. Compute $\widehat{\mathtt{FPR}}_{\mathtt{te}}(\widehat{f})$, then explain its meaning and write down its theoretical expression.
7. Compute the F-measure for this test set, and explain why one would want to compute such a thing.

Exercise 2 (20 points)

You are given a distance function $d(\cdot,\cdot)$ and four points $\{(\mathbf{x}_1, \mathrm{y}_1), (\mathbf{x}_2, \mathrm{y}_2), (\mathbf{x}_3, \mathrm{y}_3), (\mathbf{x}_{\texttt{new}}, \mathrm{y}_{\texttt{new}})\}$ with $\mathrm{y}_1 = 2, \mathrm{y}_2 = 1, \mathrm{y}_3 = 1$, and $d_1 = d(\mathbf{x}_{\texttt{new}}, \mathbf{x}_1) = 1$, $d_2 = d(\mathbf{x}_{\texttt{new}}, \mathbf{x}_2) = 2$ and $d_3 = d(\mathbf{x}_{\texttt{new}}, \mathbf{x}_3) = 5$. The goal of this exercise is to predict the label (class) $\mathrm{y}_{\texttt{new}}$ of $\mathbf{x}_{\texttt{new}}$ using a k-Nearest Neighbors classifier under various neighborhood sizes and weighting schemes. We'll consider 1-NN, 2-NN and 3-NN, and use two weighting schemes:

- (a) Uniform weighting: the weight of each member of the neighborhood is simply $1/k$.
- (b) Inverse distance weighting: the weight of each member of the neighborhood is

$$w_j = \frac{\frac{1}{d_j}}{\sum_{\ell=1}^{k} \frac{1}{d_\ell}}$$

1. Consider 1-NN, and determine $\widehat{\mathrm{y}_{\texttt{new}}} = \widehat{f}_{\texttt{kNN}}(\mathbf{x}_{\texttt{new}})$.
2. Consider 2-NN.
    1. Determine $\widehat{\mathrm{y}_{\texttt{new}}} = \widehat{f}_{\texttt{kNN}}(\mathbf{x}_{\texttt{new}})$ under the uniform weighting scheme. Comment on how to decide your answer.
    2. Determine $\widehat{\mathrm{y}_{\texttt{new}}} = \widehat{f}_{\texttt{kNN}}(\mathbf{x}_{\texttt{new}})$ under the inverse distance weighting scheme. You must show the details of your calculations.
3. Consider 3-NN
    1. Determine $\widehat{\mathrm{y}_{\texttt{new}}} = \widehat{f}_{\texttt{kNN}}(\mathbf{x}_{\texttt{new}})$ under the uniform weighting scheme. Comment on how to decide your answer.
    2. Consider the inverse distance weighting scheme.
        1. Compute the estimates of the probabilities $\Pr[Y_{\texttt{new}} = c | \mathbf{x}_{\texttt{new}}]$ for $c = \{1, 2\}$.
        2. Deduce $\widehat{\mathrm{y}_{\texttt{new}}} = \widehat{f}_{\texttt{kNN}}(\mathbf{x}_{\texttt{new}})$. You must show the details of your calculations and explain your answer.
4. Provide a useful comment on the two weighting schemes, indicating the one you would resort to and why?

Exercise 3: (4 points each for a total of 24 points)

This exercise features the analysis of the USPS digit recognition dataset using kNN with various neighborhood sizes.

```
library(mnist)
mnist <- download_mnist()
n     <- nrow(mnist)
p     <- ncol(mnist) - 1
pos   <- p + 1

mnist_train <- head(mnist, 60000)
mnist_test  <- tail(mnist, 10000)

xtrain <- mnist_train[,-pos]
ytrain <- mnist_train[, pos]
xtest  <- mnist_test[, -pos]
ytest  <- mnist_test[,  pos]

ninepics <- sample(sample(sample(n)))[1:9]
par(mfrow=c(3,3))
for(i in 1:9)
{
   show_digit(mnist, ninepics[i])
}
```

Consider classifying digit '1' against digit '7', with '1' representing positive and '7' representing negative. Store in memory your training set and your test set. Of course you must show the command that extracts only '1' and '7' from both the training and the test sets. The learning machines you will be using here at just `1NN`, `7NN`, and `9NN`.

1. Display both your training confusion matrix and your test confusion matrix
2. Display the comparative ROC curves of the three learning machines
3. Identify two false positives and two false negatives at the test phase, and in each case, plot the true image against its falsely predicted counterpart.
4. Comment on any pattern that might have emerged.
5. Perform principal component analysis on the data matrix and extract the first two components and plot them using the R Code provided
6. Compare the predictive performance of `7NN` on two PC scores to the one yielded by all the original variables. Provide a comprehensive comment.

Exercise (Bonus 1): 10 points

Given an iid sample $\{(X_i, Y_i),\ i = 1, \cdots, n\}$ for regression learning with $Y_i = f(X_i) + \varepsilon_i$, where $\varepsilon_i \overset{iid}{\sim} F_\varepsilon(\texttt{mean}(\varepsilon) = 0, \texttt{variance}(\varepsilon) = \sigma^2)$. Let $\widehat{f}$ be an estimator of f built using the random sample provided. Now let $\mathbf{x} \in \mathbb{X}$ be given. The pointwise bias-variance decomposition of the error at point $\mathbf{x}$ is given by

$$\mathbb{E}[(Y - \widehat{f}(\mathbf{x}))^2] = \texttt{variance}(\varepsilon) + \texttt{Bias}^2(\widehat{f}(\mathbf{x})) + \texttt{variance}(\widehat{f}(\mathbf{x})). \tag{1}$$

Develop and clearly write down the expression of the pointwise bias variance decomposition when $\widehat{f}$ is the kNearest Neighbors regression learner.

$$\begin{aligned} \widehat{f}_{\texttt{kNN}}(\mathbf{x}) &= \frac{1}{k}\sum_{i=1}^{n} Y_i\, \mathbb{1}(\mathbf{x}_i \in \mathcal{V}_k(\mathbf{x})) \\ &= \text{Average of the responses of the } k \text{ nearest neighbors} \end{aligned}$$

Exercise (Bonus): (20 points)

Consider the prostate cancer dataset containing the DNA MicroArray Gene Expression of both cancer and non cancer subjects.

```
prostate <- read.csv('prostate-cancer-1.csv')  # DNA MicroArray Gene Expression
```

You are supposed to provide a thorough comparison of four learning machines on this dataset, namely `1NN`, `3NN`, `5NN` and `7NN`, and your comparison will be solely based on the test error.

1. Using the whole data for training and the whole data for test, building the above four learning machines, then plot the comparative ROC curves on the same grid
2. Comment succinctly on what the ROC curves reveal for this data and argue in light of the theory whether or not that was to be expected.
3. Using `set.seed(19671210)` along with a 2/3 training 1/3 test basic stochastic holdout split of the data, compute $R = 100$ replications of the test error for all the above learning machines.
    - Plot the comparative boxplots (be sure to properly label the plots)
    - Comment on the distribution of the test error in light of (implicit) model complexity.