PUBH 6886 Fall 2025 Homework 1

Due Date: Tuesday, September 16, 2025; Total Points: 100

Complete each problem below. Show `R` code and relevant output from `R`.

1. (50 pts.) The dataset bdiag.csv contains quantitative information from digitized images of a diagnostic test (fine needle aspirate (FNA) test on breast mass) for the diagnosis of breast cancer. The variables describe characteristics of the cell nuclei present in the image.

Variable Information:

- `id`
- `diagnosis` (`M` = malignant, `B` = benign)

and ten real-valued features are computed for each cell nucleus:

- `radius` (mean of distances from center to points on the perimeter)
- `texture` (standard deviation of gray-scale values)
- `perimeter`
- `area`
- `smoothness` (local variation in radius lengths)
- `compactness` (perimeter^2 / area - 1.0)
- `concavity` (severity of concave portions of the contour)
- `concave points` (number of concave portions of the contour)
- `symmetry`
- `fractal dimension` ("coastline approximation" - 1)

The mean, standard error (se), and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, the third column is `radius_mean`, the thirteenth column is `radius_se`, and the twenty third column is `radius_worst`.

Part I:

(a) (8 pts.) Conduct a PCA on the 30 features corresponding to mean, se, and worst for each measure. Do this for the original unscaled data. Suppose that we want to retain the smallest number of PCs that explain at least 75% of the variance in the data. How many PCs should be retained (provide justification)? Explain why the first PC accounts for so much of the variance in the data set. (Constructing a biplot of PC 1 vs. PC 2 might be useful but is not necessary.)

(b) (5 pts.) Conduct a PCA on the 30 features, but this time, scale them to each have a variance of 1. Suppose that we want to retain the smallest number of PCs that explain at least 75% of the variance in the data. How many PCs should be retained (provide justification)?

(c) (12 pts.) Using the first 3 PCs derived from the scaled data, is there one pair of these PCs that appears to be predictive of `diagnosis` (`M` = malignant, `B` = benign)? Make some scatterplots to justify your response. (You do not have to do any predictive modelling here, you are simply asked to select the pair and justify the selection based on graphical evidence.) Do your best to charcterize/interpret the two PCs that appear to predict `diagnosis`.

(d) (0 pts. but try it anyways) Provide the R code to compute the `x` object output by the `prcomp` function on the scaled data using (1) the scaled data object and (2) the `rotation` output by the `prcomp` function.

Part II:

(e) (14 pts.) Use K-means clustering with squared Euclidean distance to cluster the observations based on the 30 features corresponding to mean, se, and worst (scale them to each have variance 1). Consider $K = 2$ to 10 cluster solutions and select the best solution based on a plot of the total within cluster sum of squares vs. the number of clusters. Use 100 random starts and set the seed to 1 (`set.seed(1)`) prior to running the code for obtaining all 9 clusterings.

(f) (3 pts.) Does the 2-cluster solution from part (e) correspond to the `diagnosis` (`M` = malignant, `B` = benign)? Justify your response.

(g) (8 pts.) Use agglomerative clustering based on Euclidian distance with complete linkage to cluster the observations based on the 30 features corresponding to mean, se, and worst (scale them to each have variance 1). Does the 2-cluster solution correspond to the `diagnosis` (`M` = malignant, `B` = benign)? Justify your response.

2. (50 pts.) In this problem, you will simulate some data and fit a set of linear regression models to assess their predictive performance. You will also construct a set of KNN predictions and compare them with the linear model predictions.

(a) (2 pts.) Using the `rnorm()` function, create a vector, `x`, containing 1100 observations drawn from a N(0,1) distribution. This represents a feature, X. Set the seed to 1 (`set.seed(1)`) prior to creating `x`.

(b) (2 pts.) Using the `rnorm()` function, create a vector, `eps`, containing 1100 observations drawn from a N(0,0.25) distribution—a normal distribution with mean zero and variance 0.25. This represents the error term ε. Set the seed to 2 (`set.seed(2)`) prior to creating `eps`.

(c) (2 pts.) Using `x` and `eps`, generate a vector `y` according to the model: $Y = -1 + 0.5X + \varepsilon$.

(d) (3 pts.) Create a training data set containing the first 100 `x` and `y` observations and a test data set containing the remaining 1000 `x` and `y` observations.

(e) (2 pts.) Create a scatterplot displaying the relationship between `x` and `y` in the training data set. Comment on what you observe.

(f) (4 pts.) Fit a least squares linear model to predict `y` using `x`. How do the estimated intercept and slope compare to the true values?

(g) (5 pts.) Compute the training MSE (using the training set) and test MSE (using the test data set). Is it reasonable to expect the training MSE to be smaller than the test MSE? Justify your response.

(h) (3 pts.) Display the least squares line on the scatterplot obtained in (e). Draw the population regression line on the plot, in a different color. Use the `legend()` command to create an appropriate legend.

(i) (7 pts.) Now fit a polynomial regression model that predicts `y` using `x` and the square of `x` (use `I(x^2)` in your code). Compute the training MSE (using the training set) and test MSE (using the test data set). Is it reasonable to expect the test MSE for the linear model to be smaller than the test MSE for the polynomial model? Justify your response.

(j) (8 pts.) Using the training data, fit KNN regressions of `y` on `x` for K = 1, 3, 5, and 10. Based on the test MSE, which value of K is optimal?

(k) (12 pts.) Using the `rnorm()` function repeatedly, create vectors `N1, N2, N3, N4, N5, N6, N7, N8, N9` and `N10`, each containing 1100 observations independently drawn from a N(0,1) distribution. Set the seed to 3 (`set.seed(3)`) prior to running the code to create all 10 vectors. These will serve as noise variables that are independent of the response, Y (and therefore are not predictive of Y). Append the first 100 observations from each of these noise variables to your training set that already contains `x` and `y`. Also append the next 1000 observations from each of these noise variables to your test set that already contains `x` and `y`. Fit a linear regression model on the new training data set with `y` regressed on `x` and the 10 noise variables simultaneously. Compute the test MSE for the fitted model. Compare this test MSE to the test MSE from part (g). Then fit a KNN regression on the new training data set with `y` regressed on `x` and the 10 noise variables simultaneously using the optimal K from part (j). Compute the test MSE for the fitted model. Compare this test MSE to the test MSE from part (j).