Homework 1

Due Thursday 1/27

Solutions for theoretical problems may be typed or handwritten and scanned. Solutions for applied solutions should include R code and output – try to use `R markdown`*. Homework must be submitted on Blackboard as one file.

Isabella De Leon, Armando Garcia, Jieyu Guo, and Zibo Hong will lead HW discussion.

1. (50 points) *Exploring Bias vs. Variance in Simulated Data – EoSL Figures 2.1-2.3.*

   Simulation is a numerical technique for assessing a method computationally. Simulations involve random sampling from probability distributions and statistical models so that the statistician knows and controls the truth. We will be using simulated data throughout this course and EoSL authors often use it for examples. In this problem you will use linear regression and k-nearest neighbor for predicting responses in simulated datasets generated as described in Scenario 1 (pg. 13 of EoSL).

   (a) Generate a training dataset of 100 observations for class $y =$ ORANGE with x_1 drawn from $\mathcal{N}(0,1)$ and x_2 drawn from $\mathcal{N}(1,1)$ using `rnorm` in R. Next generate a training dataset of 100 observations for class $y =$ BLUE with x_1 drawn from $\mathcal{N}(1,1)$ and x_2 drawn from $\mathcal{N}(0,1)$. Combine these two training datasets to form a dataset of 200 observations with variables $y \in (ORANGE, BLUE), x_1$, and x_2.

   Plot x_1 vs. x_2 and color the points according to their class y. Comment about the separation of classes in this data.

   (b) Fit a linear regression of y on x_1 and x_2 using `lm` in R. Note that you will need to convert character y to numeric: set $y = 1$ if ORANGE and $y = 0$ if BLUE. Calculate class predictions define by

   $$\hat{y}_i = \begin{cases} 1\,[ORANGE] \text{ if } \hat{\beta}_0 + x_{i1}\hat{\beta}_1 + x_{i2}\hat{\beta}_2 > 0.5 \\ 0\,[BLUE] \text{ if } \hat{\beta}_0 + x_{i1}\hat{\beta}_1 + x_{i2}\hat{\beta}_2 \leq 0.5 \end{cases}.$$

   Calculate and comment on the misclassification rate (i.e. discrepancy in predicted vs. observed class).

   (c) Add the decision boundary on your x_1 vs. x_2 scatterplot using `abline`. To calculate the intercept and slope of the decision boundary based on the classification cutoff of 0.5, use the fact that the decision boundary is defined by

   $$\hat{\beta}_0 + x_1\hat{\beta}_1 + x_2\hat{\beta}_2 = .5.$$

   Comment on the shape of the decision boundary and the misclassification observed visually on the plot.

   (d) Fit 15-NN with the `knn` function in package `class` using the simulated data as your training and test data. Obtain the predicted class from the `knn` object (assuming the 0.5 classification rule). Calculate and comment on the misclassification rate.

(e) Plot x_1 vs. x_2 and color the points according to their class y. Add the 15-NN decision boundary to this scatterplot. To add this boundary, fit 15-NN using the simulated data as your training data and a grid of points (covering the range of x_1 and x_2 from your scatterplot) as your test data. See the `expand.grid` function for generating the two-dimensional grid of points. Obtain the predicted probability of class ORANGE by extracting `attr(knn.object,"prob")` – note that this obtains the predicted probability for the predicted class, so to get the predicted probability of class ORANGE you need to take 1 minus the 15-NN probability for observations predicted in class BLUE.

Use `contour` to plot the 15-NN decision boundary using the grid of points as x and y and the probability of class ORANGE as z. Note that the `levels` option allows you to draw the line corresponding to 0.5. Comment on the shape of the decision boundary and the misclassification observed visually on the plot.

(f) Repeat (d) and (e) for 1-NN.

(g) Comment on differences and/or similarities in the decision boundaries for the three approaches. Would you use linear regression, 15-NN or 1-NN for prediction with this data? Explain your answer in the context of the bias-variance tradeoff.

(h) Repeat (a) - (g), but now generate x_1 and x_2 in the following way:

Generate a training dataset of 100 observations for class $y =$ ORANGE with x_1 drawn from $\mathcal{N}(0, .25^2)$ and x_2 drawn from $\mathcal{N}(1, .25^2)$. Next generate a training dataset of 100 observations for class $y =$ BLUE with x_1 drawn from $\mathcal{N}(1, .25^2)$ and x_2 drawn from $\mathcal{N}(0, .25^2)$ Note that `rnorm` takes the standard deviation as input, not the variance.

Make sure to comment as requested in (a) - (g).

(i) Explain why you came to similar or different conclusions about which method to use for the data simulated for (a)-(g) vs. the data simulated in (h). Make sure to explain in the context of the bias-variance tradeoff.

2. (15 points) *Deriving the Bias-Variance Decomposition for Expected Prediction Error with Squared Error Loss.*

Assuming squared error loss for penalizing errors in prediction, show that the expected prediction error can be written as

$$\mathrm{E}\left(Y - \hat{f}(X)\right)^2 \quad = \quad \mathrm{Var}\left(\hat{f}(X)\right) + \left[\mathrm{Bias}\left(\hat{f}(X)\right)\right]^2 + \sigma^2.$$

Assume that the form of the model is $Y = f(X) + \epsilon$, that $E(\epsilon) = 0$ and $Var(\epsilon) = \sigma^2$, and that the test data values X are fixed (i.e. not random).

Explain what the three components in the expected prediction error decomposition represent.

3. (15 points) *Unified Functional Form for Linear Regression and k-NN.* Do Exercise 2.7 part (a) in Chapter 2 of EoSL (`https://hastie.su.domains/ElemStatLearn/printings/ESLII_print12_toc.pdf`).

4. (20 points) *Squared Error Loss and Qualitative Output.* Suppose some $\hat{f}(x)$ assigns to each x the probability p_k that x is a member of the kth class. Define the K-dimensional vector

p of elements $(p_1, \ldots, p_K)$ where $\sum_{k=1}^{K} p_k = 1$. For each k with $1 \leq k \leq K$, let t_k be a K-dimensional target vector that has 1 in the kth position and 0 elsewhere. Show that choosing the class k for which p_k is largest is equivalent to choosing the class k that solves

$$\min_k ||t_k - \mathbf{p}||.$$

Also explain in words what this equivalence means.

* `R markdown`

There are a vast amount of tutorials on `R markdown` including:

- Introduction to R Markdown Video: `https://rmarkdown.rstudio.com/lesson-1.html`
- Introduction to R Markdown: `https://rmarkdown.rstudio.com/articles_intro.html`
- R Markdown Quick Tour: `https://rmarkdown.rstudio.com/authoring_quick_tour.html`

This cheat sheet has been particularly useful to me:

`https://rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf`