Homework 10

36-462/662, Data Mining, Spring 2020

Due at 10 pm on Thursday, 2 April 2020

AGENDA: Hammering home the importance of not evaluating predictive models on testing data.

1. *Online questions* (10) are online and due at the same time as the homework (but it will help if you do the reading first).

2. *Optimism and the "covariance penalty"* If we use data $(X_1, Y_1), \ldots (X_n, Y_n)$ to learn a predictive model $\hat{\mu}$, the "optimism" of our method is defined as how much worse that model would do on new data with the same values of X but *independent* Ys. That is, for each i, Y_i' has the same distribution as Y_i (conditional on X_i), but is independent of Y_i, and the optimism (for regression) is

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(Y_i' - \hat{\mu}(X_i))^2\right] - \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{\mu}(X_i))^2\right] \tag{1}$$

In this problem and the next, we'll see how to build a simple, unbiased estimator of the optimism.

(a) (5) Show that the optimism (as defined above) is equal to

$$\frac{1}{n}\sum_{i=1}^{n}\left(\mathbb{E}\left[(Y_i' - \hat{\mu}(X_i))^2\right] - \mathbb{E}\left[(Y_i - \hat{\mu}(X_i))^2\right]\right) \tag{2}$$

(b) (5) Show that $\mathbb{E}\left[Y_i' - \hat{\mu}(X_i)\right] = \mathbb{E}\left[Y_i - \hat{\mu}(X_i)\right]$.

(c) (5) Show that the optimism is equal to

$$\frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{Var}\left[Y_i' - \hat{\mu}(X_i)\right] - \mathrm{Var}\left[Y_i - \hat{\mu}(X_i)\right]\right) \tag{3}$$

(d) (5) Show that

$$\mathrm{Var}\left[Y_i - \hat{\mu}(X_i)\right] = \mathrm{Var}\left[Y_i\right] + \mathrm{Var}\left[\hat{\mu}(X_i)\right] - 2\mathrm{Cov}\left[Y_i, \hat{\mu}(X_i)\right] \tag{4}$$

(e) (5) Show that

$$\mathrm{Var}\left[Y_i' - \hat{\mu}(X_i)\right] = \mathrm{Var}\left[Y_i\right] + \mathrm{Var}\left[\hat{\mu}(X_i)\right] \tag{5}$$

(f) (5) Show that the optimism equals

$$\frac{2}{n}\sum_{i=1}^{n}\mathrm{Cov}\left[Y_i, \hat{\mu}(X_i)\right] \tag{6}$$

(g) (5) Explain why

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{\mu}(X_i))^2 + \frac{2}{n}\sum_{i=1}^{n}\mathrm{Cov}\left[Y_i, \hat{\mu}(X_i)\right] \tag{7}$$

is an unbiased estimate of

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(Y_i' - \hat{\mu}(X_i))^2\right]$$

3. *Optimism and degrees of freedom* The previous problem left us with an unbiased estimate of the risk on new data, but one which involved covariances that seem hard to estimate. In this problem, we add two assumptions:

- Our predictor is a linear smoother, so that $\hat{\mu}(X_0) = \sum_{j=1}^{n} w(X_0, X_j)Y_j$. In particular, $\hat{\mu}(X_i) = \sum_{j=1}^{n} w_{ij}Y_j$ for an $n \times n$ matrix $\mathbf{w}$.
- $Y_i = \mu(X_i)+\epsilon_i$, with $\mathbb{E}\left[\epsilon_i|X_i\right] = 0$, $\mathrm{Var}\left[\epsilon_i|X_i\right] = \sigma^2$, and $\mathrm{Cov}\left[\epsilon_i, \epsilon_j\right] = 0$ when $i \neq j$

(a) (5) Using these assumptions, show that $\mathrm{Cov}\left[Y_i, \hat{\mu}(X_i)\right] = \sigma^2 w_{ii}$.

(b) (5) Show that, under all these assumptions, the optimism is

$$\frac{2\sigma^2}{n}\,\mathrm{tr}\,\mathbf{w} \tag{8}$$

(c) (5) What is the optimism of a linear regression model with p coefficients? Answer in terms of σ^2, n and p (and numerical constants such as 2 or π). (*Hint:* Homework 6.) Does the optimism → 0 as $n \to \infty$ with p fixed?

(d) (5) What is the optimism of a k-nearest-neighbor regression? Answer in terms of σ^2, n and k (and numerical constants). Does the optimism → 0 as $n \to \infty$ with k fixed?

4. (a) (5) Explain what the following code does.

```
sim.poly <- function(n, degree) {
    x <- runif(n, min=-2, max=2)
    poly.x <- poly(x, degree=degree, raw=TRUE)
    alternating.signs <- rep(c(-1,1),length.out=degree)
    sum.poly <- poly.x %*% alternating.signs
    y <- sum.poly+rnorm(n,0,0.1)
    return(data.frame(x=x,y=y))
}
```

(You might need to look up the `poly()` function.)

(b) (2) Use the code from the previous part to generate 3 data frames, where Y is a cubic function of X plus noise. The data frames should contain 100, 1000 and 10000 rows. Check that they have the right dimensions, and that each one shows the expected cubic relationship between X and Y.

(c) (3) For each of the three data sets, do k-nearest-neighbor regression with k running from 1 to 90. Plot the *in-sample* error as a function of k for each n. (Ideally, this should be three curves, for the three sample sizes, in one plot.)

(d) (5) Use the formula for the optimism we derived in problem 2 to plot the estimated generalization error or risk as a function of k for each n. (Ideally, this should be three curves in one plot, added to the three curves from your previous plot.) Be sure to plot the risk and not the optimism itself. What k is best, according to this criterion, at each n? Why does it change with n?

(e) (5) Calculate the risk as estimated by leave-one-out cross-validation, and plot it as a function of k for each n. (Ideally, this is adding three more curves to your plot.) What k is best, according to this criterion, at each n? Why does it change with n?

Hint: The `knn.reg()` function from `FNN` is set up to do leave-one-out cross-validation for nearest-neighbor regression by default.

(f) (5) Plot the predictions you get from selected ks — are they getting visibly better as n grows?

Rubric (10): The text is laid out cleanly, with clear divisions between problems and sub-problems. The writing itself is well-organized, free of grammatical and other mechanical errors, and easy to follow. All plots and tables are generated using code embedded in the document and automatically re-calculated from the data. Plots are carefully labeled, with informative and legible titles, axis labels, and (if called for) sub-titles and legends; they are placed near the text of the corresponding problem. All quantitative and mathematical claims are supported by appropriate derivations, included in the text, or calculations in code. Numerical results are reported to appropriate precision. Code is properly integrated with a tool like R Markdown or knitr, and both the knitted file and the source file are submitted. The code is indented, commented, and uses meaningful names. All code is relevant; there are no dangling or useless commands. All parts of all problems are answered with actual coherent sentences, and raw computer code or output are only shown when explicitly asked for.