Homework 5

Advanced Methods for Data Analysis (36-402)

Due Friday February 22, 2019, at 3:00 PM

See the syllabus for general instructions regarding homework. Note that you should **always show all your work** and submit **exactly two files** (Rmd or R file as *code*, and knitted or scanned/merged pdf or html as *writeup*). Make sure that everything you submit is readable.

1. More on Optimism of the Training Error

Recall that in HW 1, we showed that

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

where the regression estimator $\hat{r}_n(\cdot)$ has been fit on an i.i.d training sample $(X_1, Y_1), \ldots, (X_n, Y_n)$ of size n from some distribution $F_{X,Y}$, and the m pairs $(X_1^{'}, Y_1^{'}), \ldots, (X_m^{'}, Y_m^{'})$ represent an i.i.d. test sample from the same distribution but with all observations independent from the training data.

We are now going to look closer at how large the so-called optimism in the training error is. For simplicity, consider fixed x_i's and let Y_i^* denote the value of a future observation of Y_i at covariate value x_i. Show that

$$\mathbb{E}\left[\left(Y_i - \hat{Y}_i\right)^2\right] = \mathbb{E}\left[\left(Y_i^* - \hat{Y}_i\right)^2\right] - 2\text{Cov}(\hat{Y}_i, Y_i),$$

where $\hat{Y}_i = \hat{r}_n(x_i)$. That is, the training error underestimates the prediction risk, and $2\text{Cov}(\hat{Y}_i, Y_i)$ measures the so-called optimism of the training error relative the (in-sample) prediction risk.

Hint: Follow what we discussed in lecture, and make sure you carefully explain all the steps taken in your calculations.

2. Housing Data (revisited)

Return to the housing data with which you worked in Homework 2. In this problem, we will take into account the locations of the census tracts using some combination of the two location variables `Latitude` and `Longitude`.

For parts (a) through (d) below, use *only* the training data set. Also, draw all scatter plots using the graphical parameter `,pch="."` in order to cut down on the overlap of points.

Part a

Draw the scatter plot of `Latitude` and `Longitude`. From the scatter plot, identify at least two places, locations in (`Latitude`, `Longitude`) space, where census tracts cluster.

Part b

Whether you did this or not in Homework 2, plot the residuals from Model 3 (the one with both `Mean_household_income` and `Median_household_income` as predictors) against each of the location variables. Are there patterns that suggest that location might be useful in predicting house values? Explain.

Part c

The most straightforward way to take location into account is to include the two location variables in a linear model. Define Model 4 to be the linear regression of `Median_house_value` on the four predictors consisting of the two in Model 3 and the two location variables.
Fit this model and plot its residuals against the four predictors and its fitted values. Between the residual plots and the summary of the regression, explain why it appears that one of Model 3 or Model 4 is better than the other.

Part d

It might be naive to assume that `Longitude` affects housing prices linearly over the long distance from California to Pennsylvania. For example, if house prices in Philadelphia (`Longitude` about -75 and `Latitude` about 40) are higher than house prices in Pittsburgh (`Longitude` about -80 and `Latitude` about 40.5), a linear contribution of `Longitude` that picks up the difference in price will be making a huge negative contribution to prices in California (`Longitude` about -120). But prices in California are also higher than in Pittsburgh.
To avoid such considerations, a nonparametric model in which the conditional mean of `Median_house_value` given the predictors is allowed to be a nonlinear function of the predictors. Fit two additional models:

Model 5: A kernel regression of `Median_house_value` on `Median_household_income` and `Mean_household_income`.

Model 6: A kernel regression of `Median_house_value` on `Median_household_income`, `Mean_household_income`, `Latitude`, and `Longitude`.

Once again, you will need to load the `np` package with `library(np)`. You need a different bandwidth for each predictor. For bandwidths, use the sample standard deviations of each predictor divided by $n^{1/5}$, where n is the number of data points used to fit the model. [1]

For example, in Model 6, the predictors in the order stated are variables 5, 6, 2, and 3 in the supplied data files. So the bandwidths can be supplied as the argument

```
traindat <- read.csv("housetrain.csv", header=TRUE)
n <- nrow(traindat)
bws <- apply(traindat[, c(5,6,2,3)], 2, sd)/n^(0.2)
```

to the `npreg` command, where n is the size of the training data, (5303 in this case but *different* in part (e).) Plot residuals against the fitted values and predictors, and comment on any patterns you see. The residuals will be in the fitted object in an item called `resid` if you add the argument `residuals=T` when you call `npreg`. The fitted values will be in an item called `mean` if you *don't* specify any `newdata`.

Part e

Now we will perform 5-fold cross-validation to choose between Models 3–6 as predictors. For each fold of cross-validation, create the test set by randomly selecting $n_k = 2121$ for each $k = 1, \ldots, 5$. The test folds and

[1] The choices of bandwidths in these exercises was made based on some theory that will appear in a later lecture. The specific choices are not "optimal," but they are popular rule-of-thumb choices.

corresponding training portions can be created in a manner similar to Demo 3:

```
dat <- read.csv("housing.csv", header=TRUE)
nfold <- 5
samp <- sample(rep(1:nfold, ceiling(nrow(dat)/nfold))[1:nrow(dat)])

for(k in 1:nfold){
 testd <- dat[which(samp==k), ]
 traind <- dat[-which(samp==k), ]
 ## Do stuff ##
}
```

The `npreg` function will compute predictions while fitting the model if you run it with a command like

```
library(np)
model5 <- npreg(Median_house_value ~ Median_household_income +
                Mean_household_income, data=traind, newdata=testd,
                bws=apply(traind[, c(5,6)], 2, sd)/nrow(traind)^(0.2))
```

This time n is the size of the `traind` data set (8484), which is different from the n in part (d). The predictions will then be in `model5$mean`. For each fold k and each model $m = 3, 4, 5, 6$, report the resulting values of the average squared prediction error within fold k for Model m. You can write a loop to do all the calculations and store the results in a 5×4 matrix.

Part f

Compute the average of the five cross-validation error values for each model along with the corresponding standard error as defined in lecture. Comment on which model or models are clearly better than the others. How much better is the best model than the second best compared to the standard errors? Comment on what this says about how good the models are compared to each other.

3. Practicing Inference

You can find the data set on abalones on the HW 5 page. The data set contains nine variables measured on 4173 abalones. Here is a description of the nine variables:

Name	Data Type	Units	Description
Sex	nominal	M, F, I (infant)	
Length	continuous	mm	
Diameter	continuous	mm	
Height	continuous	mm	
Whole.weight	continuous	grams	Entire abalone
Shucked.weight	continuous	grams	Weight of meat
Viscera.weight	continuous	grams	gut weight (after bleeding)
Shell.weight	continuous	grams	after being dried
Rings	integer		+1.5 gives the age in years

A fisherperson who wishes to sell the abalone for food is interested in the edible part. Prior to cutting one open for cooking, one can predict the weight of the edible part (`Shucked.weight`) through a regression model. A natural predictor that would be available, if a scale were available, is `Whole.weight`. Assume that the only predictors available, when the prediction of `Shucked.weight` is needed, are `Diameter`, `Length`, and `Height`. Here is one hint: Objects of uniform density have weights proportional to their volume. What functions of the predictors might be good predictors of the volume of an abalone?

Part a

Do exploratory analysis of the response, `Shucked.weight`, and the three predictors: `Diameter`, `Length`, and `Height`. Summarize what you see from the exploratory analysis.

Part b

We will explore linear and kernel regression models for predicting the response from some or all of the predictors. Whenever you need a bandwidth for a variable in a kernel regression, use the sample standard deviation of the variable divided by $n^{1/5}$, where n is the size of the sample. This applies to the full data or to a cross-validation calculation.

The models that you fit should include the following, but feel free to fit other models if you can think of reasons to do so:

Model 1: A linear regression of log(`Shucked.weight`), on the logarithms of all three predictors.

Model 2: A kernel regression of `Shucked.weight`, on all three predictors: `Diameter`, `Length`, and `Height`.

Note: When making predictions with Model 1, the `predict` function will use the predictors correctly, but it will predict the logarithm of the response. You will need to take `exp` of the predictions before comparing the predictions to `Shucked.weight` in the calculation of cross-validation error.

Draw a scatter plot of the fitted values for both models against each other, and comment on how similar or different the predictions seem to be.

Examine residuals from each model and say what you learn about the possible distributions of the noise terms and how those distributions might be related to other variables.

Part c

Perform five-fold cross-validation and compute a rough estimate of its standard error. Based on this result and your residual analysis, which model appears to perform best?

Part d

Under the assumption that $\hat{\beta}$ from the linear model is multivariate normal, compute a 95% confidence interval for $\beta_{\text{log-Diameter}}$. Does it include 0? Confirm your calculations with the `confint` command. Based on the EDA and residual analysis, comment on how reliable you think this confidence interval is.

Hint: You can retrieve an estimate of the variance-covariance matrix using the `vcov` command in `R`. For the question, think about what assumptions went into the computation of this confidence interval, and whether it is reasonable to assume that $\hat{\beta}$ is multivariate normal.

Part f

Now let's get a confidence interval for $\mathbb{E}\left[\log(Y) \mid \text{log-Diameter} = 5.93, \text{log-Length} = 6.10, \text{log-Height} = 6.24\right]$. Using again the variance-covariance matrix for $\hat{\beta}$, provide a 95% confidence interval under the assumption that $\hat{\beta}$ is distributed as a multivariate normal. Confirm your calculations with the `predict` function and `interval = "confidence"` argument.

Part g (extra credit)

One issue with transforming the response in a regression is the following. We defined $r(x)$ to be $\mathbb{E}\{Y \mid X = x\}$, the conditional mean of Y given $X = x$. In Model 1, we model

$$\log(Y) = \tilde{r}(x) + \varepsilon, \tag{1}$$

where ε is independent of X and has a distribution centered at 0. However, if the mean of ε is 0, $\exp(\tilde{r}(x))$ is *not* the conditional mean of Y given $X = x$. We can still use $\exp\left(\widehat{\tilde{r}(x)}\right)$ as an estimator of $r(x)$, but it will be biased. However, let's ignore this issue for now (in practice though, this cannot be ignored. For instance, one could fit a glm instead). What is the variance of $\exp\left(\widehat{\tilde{r}(x_0)}\right)$, where $\hat{r}(x_0) = \hat{\mathbb{E}}[\log(Y) \mid \text{log-Diameter} = 5.93, \text{log-Length} = 6.10, \text{log-Height} = 6.24]$?

This is not very easy to compute, but we can be strategic. Soon we will introduce the bootstrap as a way of quantifying the uncertainty when it is hard to do that analytically. Here, we propose a simple trick. Do a first-order Taylor expansion of the function $f(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3) \equiv \exp\left(\widehat{\tilde{r}(x)}\right)$ around the true value of $\beta = (\beta_0, \beta_1, \beta_2, \beta_3)$. This should give you a linear expression in $\hat{\beta}$. The approximation will be good if the values of $\hat{\beta}$ and β are close. For the linear approximation, it is easy to compute a variance by using the usual estimate of the variance-covariance matrix of $\hat{\beta}$ provided by the `vcov` command.

Hint: Recall, that a first-order Taylor expansion of a function $f(\mathbf{x})$ around a point $\mathbf{a}$ is

$$f(\mathbf{x}) \approx f(\mathbf{a}) + Df(\mathbf{a})(\mathbf{x} - \mathbf{a})$$

where $Df(\mathbf{a})$ is the matrix of partial derivatives evaluated at $\mathbf{a}$ (in this case it is a 1×4 vector).