Statistical Learning and Data Analysis 2021
Lab 3 - Regression

Yuval Benjamini

Due June 15 before 4:30pm

Hand In Procedure: Labs can be handed in alone or in pairs (no more than 2 per lab!). Please prepare a file with a writeup and code (the writeup can be in Hebrew or English). Please make sure the reports to Sections 2, 3 are less than 6 pages long together (not including code).

1 Simulation

1. Write a function that can sample iid data-points (X, Y) from the following model:

$$Y|X \sim f(X) + \epsilon \quad f(x) = sin(\lambda x) + 0.25x^2 + ((x - 0.4)/3)^3, \tag{1}$$

   for $X \sim \text{Uniform}(-2, 2), \epsilon \sim \text{Normal}(0, 0.3)$ and ϵ independent of X.
   The function should input either a vector of x values or the number of samples n, as well get as the parameter λ. It should return a vector of predictors (x's) and a vector of responses (y's).

2. Implement a Kernel Regression function that takes a training set, a bandwidth parameter h, and an input x, and returns a prediction for y. Use a Gaussian kernel. You can assume x is 1-dim.You can compare your code to the `ksmooth` R function [1].

3. For each combination of $\lambda = 1.5, 5$ and $n = 50, 200$ data points, sample a training set of size n from the function in (1). Now, choose multiple values of h. [2] For each h, use the Kernel Regression function to compute or estimate the following quantities:

   (a) Compute the expected optimism [*Eop*] of regression function based on the x's in the training set. (What is $\mathbf{w}$? What is σ^2?)

   (b) Estimate the accuracy of the regression using 5-fold cross-validation error. Write your own code.

   (c) Compute the in-sample expected error (EPE_{IN}) of your regression for multiple values of h.
   Hint: use the true regression function in Equation 1.

   (d) Estimate the out-of-sample expected prediction error (EPE) of your regression function. (Do this by sampling new data-points from the model).

4. Repeat these steps for a quadratic regression prediction model:

$$\hat{y} = b_1 + b_2 x + b_3 x^2,$$

   meaning that the predictor vector for x should be $z = (1, x, x^2)'$ and (b_1, b_2, b_3) fit using ordinary least squares (OLS).

[1] The bandwidth of the kernel (h) might be defined differently in `ksmooth`

[2] Idea is to see how this value changes with h. Start with a value of h that is too small, and use multiple values including a value that is too large. Use the same h's for each λ, n combination.

Your results should be presented in one or multiple graphs, with h in the x axis and error in the y axis. Discuss your results.

2 Covid-19 Mortality Data

Your goal is to extract estimate the rate of change in Covid-19 (Coronavirus) case data in Israel. The data can be found in *https://github.com/idandrd/israel-covid19-data/blob/master/IsraelCOVID19.csv*.

Please prepare the following two plots:

1. A figure showing the number of new detected Covid-19 cases per day, with both the observed values and a regression curve. The regression should be estimated outside the figure (do not use *ggsmooth*). Make sure the regression line is easy to see (you can make the true points small or in light color).
2. A figure showing the daily change in rate of new detections per day. To estimate these use the first-derivative of the regression curve from Part 1.

Choose the tuneable parameters deliberately so the curve would fit well, and so that the rate of new detections per day is smooth enough.

3 fMRI Data

We will fit prediction models for the response voxels (Y) in V1 in response to natural images (X). The data consists of 3 measured responses to each image (at 3 locations, *voxels*, in the brain). Your main goal is to predict the response of each voxel to new images. The secondary goal is to interpret the prediction models in relation to the scientific problem.

We will fit models using the ($n = 1500$) training examples, and estimate the accuracy of prediction on $n^* = 250$ validation examples. The training data consists of the BOLD response summary of voxel to 1500 images. I will provide 3 different responses, meaning that you will fit 3 separate regression models.

The prediction models should be able to predict a response for out-of-sample images. We will validate thier accuracy based on how well they predict the validation data:

$$MSE(\hat{f}) = \frac{1}{120}\sum(\hat{f}(I_j) - Y_j)^2),$$

and the square-root of the mse:

$$MSE(\hat{f}) = \sqrt{MSE(\hat{f})}.$$

The basic data is as follows:

- `train_resp.csv` 1500×3 responses of training data.
- `feature_train.csv` 1500×2729 features for each train image.
- `feature_valid.csv` 250×2729 features for each validation image.

The idea is to fit a regression model for each column of `train_resp.csv` using the matrix of features in `feature_train.csv`. Then, you will report the predictions for the feature vectors in `feature_test.csv`.

In order to understand the data, we also provide to original images and the transformation matrix for the features.

- `train_stim.csv` 1500×16384 images of training data

- `valid_stim.csv` 250×16384 images of validation data
- `feature_pyramid.csv` 2729×16384complex-number description of filters

In particular, if `feature_pyramid` is the Gabor pyramid and `fit_stim` then the transformation that was used to get from the image vectors to the feature vectors is

$$features = log(abs(fit\_stim\% * \%feature\_pyramid) + 1))$$

3.1 Prediction model

For each voxel, fit a linear model of the features. Because there are more features than responses, some method for high-dimensional regression will be needed. Consider using a data-splitting technique to estimate the success of your model, which will be faster than using cross validation.

3.2 Analysis of results

Choose the voxel (1 response of the 3) for which you model works best. You goal is to get insight about how your model works, and try to use what you learned to further improve the model.

Here are several ideas you can try. Try 2 of the following:

- Use a metric to identify several important features in your model. Explain your choice of metric. What do those features have in common (location in the image, orientation, level of the pyramid)?
- Take the single most important feature (using metric as before), and plot its relation with the response. Does the plot suggest a linear relation?
- Look at images set aside for the test set. Which images get the highest predictions? Which images get the lowest predictions? Can you learn anything by comparing the two sets? Can you find a nice way to visualize what you found?
- If you gained insight, you can try to use it to improve the current prediction model.

3.3 Submit predictions for the validation data on the 3 voxels

Save the following two objects into an Rda file and upload into moodle:

- A 250×3 R matrix called `preds` with predictions of the mean response for the 250 validation images. That means that column 1 row 2 should contain the prediction of the model corresponding to voxel 1 to image 2.
- A 1×3 vector called `mses` with *your* estimate of the MSE (not root-mse) for each prediction model.

We will provide an example submission for you to follow.