Statistical Learning, Tutorato #5

Veronica Vinciotti, Marco Chierici

April 21, 2023

Exercise 1

The `College` dataset (contained in the `ISLR2` library) collects statistics measured on 18 variables for 777 US colleges. Some of the variables include whether the college is a private or public institution, the number of application received, the number of applications accepted, etc. (for full details: `?College`)

Here, we want to predict the number of applications received using the other variables.

a. Split the data into a training/test set.
b. Fit a least squares linear model on the training set set and evaluate the error on the test set.
c. Fit a ridge regression model on the training set, choosing λ by cross-validation. Report the test error.
d. Fit a lasso model on the training set, choosing λ by cross-validation. Report the test error and the number of non-zero coefficient estimates.
e. Comment on the results obtained. How accurately can we predict the number of college applications received? Is there much difference among the test errors resulting from the different approaches? Draw a plot of the predictions vs the true response values on the test data for all of the models.

Hints

- To perform ridge regression and the lasso, use the `glmnet()` function (`glmnet` package);
- The syntax is `ridge_mod <- glmnet(x, y, alhpha, lambda)`:
  - note that **no formula notation** is supported;
  - `alpha=0` performs ridge regression;
  - `alpha=1` performs the lasso;
  - the optional parameter `lambda` is used to pass a grid of possible values of λ; if this parameter is omitted, the `glmnet()` function performs ridge regression for an automatically selected range of λ values.
- It may be convenient to use `model.matrix()` to prepare the `x` data for `glmnet()`;
- In order to get the predictions on new data, given a value L of λ: `predict(ridge_mod, s=L, newx=x_test)`
- In order to obtain the coefficients given a value L of λ: `predict(ridge_mod, s=L, type="coefficients")`
- Instead of arbitrarily choosing λ, it would be better to use cross-validation to choose the optimal λ. Use the built-in cross-validation function `cv_mod <- cv.glmnet(x, y, alpha, lambda)`, with the same syntax as `glmnet()`. The optimal lambda can be found in `cv_mod$lambda.min`.

Exercise 2

The `Boston` data (`MASS` library) contains housing values in the suburbs of Boston for a sample of 506 observations. The aim is to predict per capita crime rate (`crim`) from the other variables.

a. Try out some of the regression methods you learned so far, such as subset selection (forward and backward), lasso, ridge regression. Present and discuss results for the approaches that you consider.

b. Propose a model (or set of models) that seem to perform well on this data set, and justify your answer. Make sure that you are evaluating model performance using validation set error, cross-validation, or some other reasonable alternative, as opposed to using training error.
c. Inspect your selected model. Does it involve all of the features in the data set?

Exercise 3

For this exercise, we explore how lasso performs on simulated data.

a. For this first exercise, we will use again the simulated data of Exercise 3, Tutorato #4, ie:

- Generate a predictor x of length $n = 100$ and a noise vector ϵ of the same size, using the `rnorm()` function.
- Generate a response vector y of length $n = 100$ from the model

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \epsilon$$

where you can freely choose the values for the constants β_i.
We now pretend that we do not know the true model and fit a polynomial model with degree 10 on the simulated data. Fit a lasso model using the 10 predictors. Select the optimal λ by cross-validation (CV), specifying a grid of possible λ values (see the beginning of 6.6.1); plot the CV error as a function of λ. Report and discuss the coefficient estimates. Is the selected model close to the true model, i.e. the model that you used for simulating the data?

b. As a second simulation, generate a new response vector y according to the model

$$y = \beta_0 + \beta_7 x^7 + \epsilon$$

where you choose a value for β_7 (and reuse the previous value for β_0). Perform best subset selection and the lasso, again with a polynomial model of degree 10. For lasso, select the optimal λ by cross-validation, letting the function choose its own grid of values. Discuss the results obtained.