Model Selection

Model selection is actually variable selection. A common sense we got from linear regression is that adding predictors to our model would always decreases the training error or RSS. However, this is not the case for test error which we should really care about. Therefore, we should find a way to choose significant predictors which could take a balance between training error and test error.

Best Subset Method

The idea of best subset method is to compare all models with k predictors (possible models are C_p^k) and then choose the model with the smallest RSS or the largest R^2. We'll do this for every possible k. Naturally, the RSS would decrease and R^2 would increase as we increase k. As we stated before, the thing does not go like this for test error. Therefore, we should take test error into consideration to decide best k.

How to choose k?

The best k should minimize the test error, not the training error. There are several ways to estimate the test error.

- Akaike Information Criterion (AIC)

  $$2k - 2\ln(L)$$

  where L is the value of the likehood, k is the number of predictors. AIC rewards models that achieve a high goodness-of-fit score (maximize likelihood) and penalizes them if they become overly complex (minimize the number of predictors). The smaller AIC, the better model. It is similar to Occam's Razor
- Bayesian Information Criterion (BIC)

  $$2\ln(N)k - 2\ln(L)$$

  where L is the value of the likehood, k is the number of predictors and N is the number of samples. BIC penalizes the complexity of model more than AIC. When N is larger, it could control over-fitting better.
- Adjusted R^2:

  $$R^2_{adj} = 1 - \frac{\frac{RSS}{n-k-1}}{\frac{TSS}{n-1}}$$
- Cross-validation estimator
  - How to perform cross validation on model selection?
    - Randomly split data into three sets: training, validation and test.
    - Train different models on the training set.
    - Evaluate each trained model on the validation set (i.e. compute prediction error).
    - Select the model with lowest prediction error.
    - Estimate the prediction error of the selected model on the test set.
  - Two ways of cross validation

    - K-fold cross validation

      K-fold CV splits data into K equally sized blocks and then train a model using all blocks except block k. When an iteration is completed (K times), we would take an average as the estimate of prediction error given a specific parameter.

    - Leave one out cross validation

      LOOCV train different models on evert point except i and then compute the test error on the held out point.

    - K-fold CV v.s. LOOCV

      a. K-fold CV depends on the chosen split

      b. In K-fold CV, we train the model on less data than what is available. This introduces **bias** into the estimates of test error.

      c. In LOOCV, the training samples highly resemble each other. This introduces the **variance** of the test error estimate.

    - Bootstrap v.s. CV

      CV helps us estimate the test error while boostrap (sample data with replacement) could simulate the distribution of data.

What's the problems?

- It is often very expensive computationally. We have to fit 2^p models.
- If for a fixed k, there are too many possibilities, we increase our chances of overfitting. The model selected has high variance.

In order to avoid these problems, we can restrict our search space for the best model. This reduces the variance of the selected model at the expense of an increase in bias.

Stepwise Selection

Forward Selection

- Start from a null model.
- In each iteration, we would choose 1 variable which minimizes RSS or maxmizes R^2 the most.
- Stop iteration until the change of RSS or R^2 is less than the threshold.

Backward Selection

- Start from a full model.
- In each iteration, we would delete 1 variable from the model which could minimizes RSS or maxmizes R^2 the most after deletion.
- Stop iteration until the change of RSS or R^2 is less than the threshold.

Mixed Stepwise Selection

This method looks like a combination of forward selection and backward selection. Its idea is that do forward selection, but at every step, remove any variables that are no longer "necessary".

References

1. The Akaike Information Criterion