October 6, 2017 Lecture 7, Sep. 27, 2017

Validation and Decision Tree

Lecturer: Justin Domke *Scribe: Boya Ren*

1 Summary

Last time we talked about different regularization methods and today we will continue to discuss validation methods and proceed to decision tree models.

Reading “The Elements of Statistical Learning” Sec 9.2.1, 9.2.2

2 Validation

We know the learning targets for ridge and lasso regression are as below.

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij}) + \lambda \|\beta\|_2^2$$

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij}) + \lambda \|\beta\|_1$$

Because the objective function of Lasso regression has hinge points, it can return sparse solutions in some cases. However, Ridge regression only has dense results. Then the question arises, how to choose capacity, i.e. choose λ?

2.1 Train - Validation

Recipe

1. Given data, randomly partition into training set (Tr) and validation set (V). Example would be 70% training and 30% validation.
2. Fit each model to Tr, and then evaluate on V.
3. Pick model with the best score on validation data.
4. Retrain that model on all data and return it.

Now we will analyze bias and variance with Train - Validation approach. It is the noted that here bias and variance are statistical terms. Suppose we take N data, train the model on $N/2$ data, and then evaluate on the other $N/2$ data. Compare the following

- Validation-error (what you compute)
- Generalization-error (average error if you train on all N data and evaluate with new data)

We want $validation - error \approx generalization - error$, but

- There is bias, since we trained on $N/2$ data not N
- There is variance, since the $N/2$ training samples are chosen randomly

3 K-fold Cross - Validation

Recipe

1. Randomly partition data into K blocks $B_1, B_2, \cdots, B_K$.
2. For each “fold”, $k = 1, 2, \cdots, K$.
   - Let $V = B_k$ and $Tr = Data \backslash B_k$.
   - Train each model on Tr and evaluate on V.
3. Average validation errors over folds.
4. Pick the best model, and retrain on all data, return.

Compare to train - validation, there are two points to be discussed below.

- K - fold cross - validation usually requires fitting K times for each model, so it is factor K slower. A notable exception is nearest neighbors, since it does not require training as long as data is loaded
- If $K = N$, it is called “leave one out” cross - validation

Now we will discuss bias and variance with K - fold cross - validation, using the following scenarios.
Q) Bias of K - fold v.s. bias of a single validation on set of size N/K?
A) Same
Q) Variance of K - fold v.s. bias of a single validation on set of size N/K?
A) Variance of K - fold is lower, since it averages K validation errors
Q) Bias of K - fold v.s. $K + 1$ - fold?
A) Bias of $K + 1$ - fold is lower, since it uses more training data each time
Q) So does it mean “leave one out” is the best?
A) No. Because lower K means lower variance. When K is high, the error estimates from all folds will be highly correlated, which means that the variance is higher

4 Regression Trees

The idea of decision trees for regression is to set piecewise constants on regions. Specifically it can be defined as below.

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m)$$

The equation is saying, regression tree will divide the input data into M regions $R_1, R_2, \cdots, R_M$ and for an input data, it outputs c_m is input falls in region R_m. We will illustrate it through the following example. Suppose we have a decision tree on the left, then it divides the input as on the right.

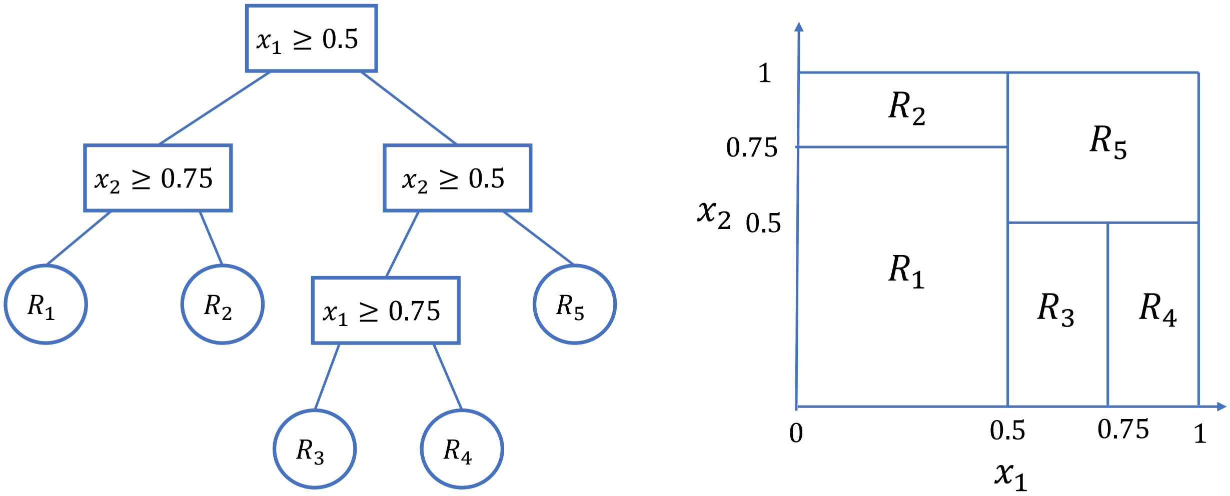

Figure 4.1: An Example of Regression Tree

Given test samples, we can easily match them to the five regions, as shown below.

$x = (0.3, 0.3) \to R_1$
$x = (0.9, 0.9) \to R_5$
$x = (0.6, 0.3) \to R_3$

The above shows how to evaluate a decision tree which is fairly simple and straightforward. Then we will talk about how to learn a decision tree.

First we try to answer this question. Given $(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$, if $R_1, R_2, \cdots, R_M$ are fixed, how to set $c_1, c_2, \cdots, c_M$? Our objective is to minimize the following

$$\begin{aligned} &\min_{c, c_2, \cdots, c_M} \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} (y_i - c_{m_i}),\ \ x_i \in R_{m_i} \end{aligned}$$

As usual, we compute the derivative of $c_m, m = 1, 2, \cdots, M$ and derive its solution.

$$\begin{aligned} \frac{d}{dc_m}\sum_{i=1}^{N}(y_i - c_{m_i})^2 &= \sum_{i=1}^{N} 2(y_i - c_{m_i})\frac{d}{dc_m}(y_i - c_{m_i}) \\ &= -2\sum_{i=1}^{N}(y_i - c_m)I[m = m_i] \end{aligned}$$

$$\hat{c}_m = \frac{\sum_{i=1}^{N} y_i I[m = m_i]}{\sum_{i=1}^{N} I[m = m_i]}$$