MATH 191 Topics in Data Science: Algorithms and Math. Foundations - Fall 2015

Lecture 25: November 30, 2015

Lecturer: Mihai Cucuringu *Scribe: Lawrence Ouyang*

1 Clustering (continued)

1.1 k-Means and k-Means++

Previously, we described the problem of partitioning a graph, with the minimum number of edges between partitions. The k-means algorithm is a solution to this problem, in which we group the nodes into k clusters, with the nodes in those clusters being closer to each other.

Although this solution is an effective one, the k-means algorithm selects the k clusters randomly, which can lead to poor partitions. We can somewhat resolve this by making an improvement to our k-means algorithm by randomly selecting only one center, and then choosing new centers determined by their weighted distance. The algorithm below describes this process:

Algorithm 1 k-Means++

Require: A set of n data points $N \subset \mathbb{R}^d$, the number of clusters k

1: Randomly select an initial center c_1 from N

2: **repeat** for $i \in 1, 2, \ldots, k-1, k$

Select the next center $c_i = x \in N$ with the probability

$$P(x) = \frac{D(x)^2}{\sum_{x' \in N} D(x')^2} \tag{1}$$

Where x' is the closest center that has already been chosen and $D(x')$ is the distance to that center.

3: Continue with the standard k-means algorithm

Despite the improvement in center selection, we still have the issue of randomization. k-means++ can still select a center that is close to another cluster, which would result in a poor partition.

1.2 Spectral Clustering

Another popular cluster method is spectral clustering, where data points are clustered based on similar attributes but not necessarily within a compact boundary, as opposed to k-means where each cluster is centered. Because of this, spectral clustering is more powerful and versatile which has made it a very widely used technique in a variety of fields involving data analysis.

Spectral clustering reduces our graph into matrix form with the graph Laplacian. We then use the Laplacian matrix's eigenvectors to determine the cluster the data points belong to. The algorithm is: Spectral clustering preserves both the connectivity and spacial closeness of the nodes,

Algorithm 2 Spectral Clustering

Require: A set of n data points $N \subset \mathbb{R}^d$, the number of clusters k

1: Find a similarity matrix A.
2: Construct a normalized Laplacian matrix defined as

$$L = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \tag{2}$$

3: Find the top k eigenvectors of L and congregate them as the column values of a matrix V.
4: Normalize each row to unit length, and cluster them using k-means
5: Assign points in V to the a cluster on the resulting ith result.

which allows a much more informative analysis of the data.

1.3 Signed Graphs

Our clustering techniques fall short with signed graphs, that is, edges with positive or negative weights. How do we cluster these graphs so that information can be usefully obtained?

A way of doing this would be to cluster our data points by those connected by positive edges, and separate out clusters by negative edges. We can use a balanced normalized cut

$$\min_{x_1,\ldots,x_k} \in I(\sum_{c=1}^{k} \frac{x_c^T (D^+ - A) x_c}{x_c^T x_c}) \tag{3}$$

to minimize the number of edges between clusters.

2 Shrinkage Methods

We end the course by going full circle back to the first few lessons of this course. The purpose of data analysis tools is to obtain readable and useful information. The interpretability of most data models will tend to have a large number of variables, which may or may not be related to the response.

A shrinking technique fits a model with all predictors, but constrains the coefficients such that they reduce towards zero, resulting in lower variance. The two popular methods for this are ridge regression and LASSO regression.

2.1 Ridge Regression

Ridge regression shrinks adds a constant λ and attempts to shrink all βs to zero. We first obtain the βs that minimize the residual sum of squares:

$$RSS = \sum_{i=1}^{n}(y_i\beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \tag{4}$$

Now we, once again, find the minimum of β with our λ value added:

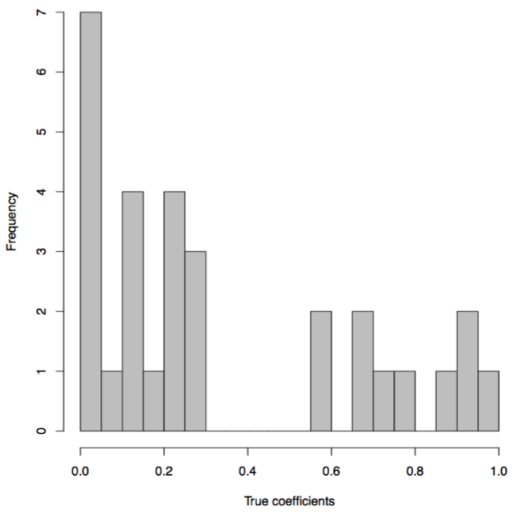
$$\begin{aligned}\hat{\beta}^{ridge} &= arg\min_{\beta\in\mathbb{R}^p} RSS + \lambda\sum_{j=1}^{p}\beta_j^2 \\ &= \sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \\ &= ||y - X\beta||_2^2 + \lambda||\beta||_2^2 \end{aligned} \tag{5}$$

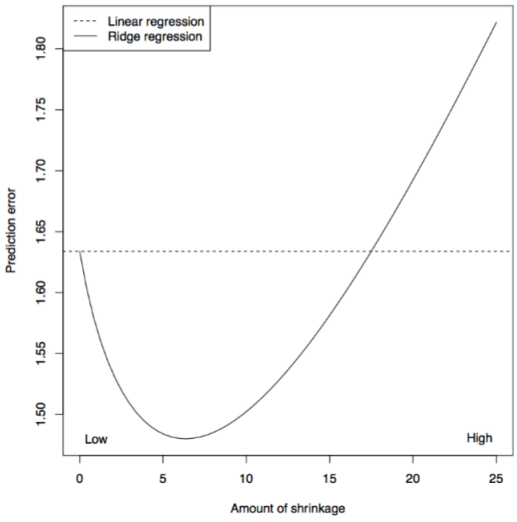
λ is a tuning parameter, that is, it is chosen so that the model either fits the data or shrinks the coefficients.

- $\lambda = 0$ results in no shrinkage
- $\lambda \to \infty$ results in $\hat{\beta}^{ridge}$ going to zero

Below is an example experiment using ridge regression with the following parameters:

- $n = 50$, $p = 30$, $\sigma^2 = 1$
- True linear model with 10 large coefficients and 20 small ones





It is quite clear that there is a proper amount of shrinkage(λ) such that the prediction error is at its minimum. The goal of ridge regression is to use that shrinkage for better prediction values. However, the shrinkage creates bias. As λ increases, variance decreases while bias increases.

2.2 LASSO

LASSO follows very similarly to ridge regression, however, LASSO actually shrinks coefficients to zero due to the L_1 norm of β.

$$\begin{aligned}\hat{\beta}^{LASSO} &= arg\min_{\beta\in\mathbb{R}^p} RSS + \lambda\sum_{j=1}^{p}|\beta_j| \\ &= \sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\sum_{j=1}^{p}|\beta_j| \\ &= ||y - X\beta||_2^2 + \lambda||\beta||_1 \end{aligned} \tag{6}$$

Similar to ridge regression, the following also apply to LASSO:

- $\lambda = 0$ results in no shrinkage
- $\lambda \to \infty$ results in $\hat{\beta}^{ridge}$ going to zero
- λ increases, variance decreases, bias increases

Since LASSO can shrink some coefficients to zero, it can be selective with the variables used in the model. As the shrinkage increases, more variables become zero, which result in the remaining variables shrinking even more. Both ridge and LASSO are effective shrinking methods that can be used for a wide variety of data. Due to the way they handle coefficient shrinking, ridge works better with more predictors with small true coefficients, while LASSO works better with less predictors and higher true coefficients as there will be less "overshrinkage" from zeroing out variables.