MIS 381N

Stochastic Control and Optimization: Project 3

Introduction

Feature selection is very useful in data mining because the data contains many redundant or irrelevant features. The direct selection has been dismissed for a long time because of computational infeasibility. Lasso regression is one very popular indirect selection method because of its computational feasibility and good predictive performance. By introducing a penalty term, the coefficients shrink to zero. One essential weakness of Lasso is the downwards bias of estimates. On the other hand, the computational power of solving the direct problem has increased at an astonishing rate. On the same computer, the speedup factor between 1991 and 2013 is about 580,000. If we also consider the dramatic improvements in hardware, the total speedup is approximately 200 billion. In this project, we will learn and apply the direct selection method.

Lasso Regression

We consider the linear regression model with dependent variable y, independent variables X, regression coefficients β and errors ϵ:

$$y = X\beta + \epsilon.$$

Feature selection here is to choose at most k variables out of p variables (X has p variables).

Lasso selects the variables via solving the convex optimization problem

$$\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2 + \lambda \, \|\beta\|_1,$$

where $\|\beta\|_1 = \sum_i |\beta_i|$ and $\|\beta\|_2^2 = \sum_{i=1} \beta_i^2$. The penalty term $\lambda \, \|\beta\|_1$ shrinks the coefficients towards zero and naturally produces a sparse solution by setting many coefficients to be exactly zero

Direct Selection – MIQP Problem

If we use the sum of squared errors as a criterion, then variable selection problem is equivalent to

$$\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2$$

$$subject\ to, at\ most\ k\ \beta s\ are\ nonzero.$$

We can introduce a binary variable z_i to indicate whether β_i is zero. In other words, $z_i = 0$ is equivalent to $\beta_i = 0$. By using binary vector $z \in R^{p \times 1}$, the model can be reformed as

$$\min_{\beta} \frac{1}{2} \|y - X\beta\|_2^2$$

$$subject\ to: \sum_i z_i \leq k$$

$$-M\ z_i \leq \beta_i \leq M\ z_i, i = 1, \dots, p$$

$$z_i\ is\ binary, i = 1, \dots, p,$$

We guarantee the equivalence between $z_i = 0$ and $\beta_i = 0$ by the second constraint $-M\ z_i \leq \beta_i \leq M\ z_i$. The positive constant M will be discussed in detail later. Because vector β is continuous, vector z is binary, and the objective function is a quadratic function of β and z, this problem is called mixed integer quadratic programming problem (MIQP).

Choice of M

The constant M should satisfy that M is larger than the largest absolute value of the optimal solution. On the other hand, we need M to get the optimal solution. Thus, in theory, it is extremely hard to choose M before solving the problem. In this project, we will use the following practical method. Start with a M, solve the MIQP. If the largest absolute value of this solution is equal to M, then double M and solve the MIQP again. Keep doing this until the largest absolute value of this solution is strictly small than M. Because we always want to have a small number of feasible solutions, you may start with a small M.

Provided Files

data.Rdata contains independent variables X, dependent variable y, and the real coefficients beta_real.

Specifics

1. MIQP Solver – Gurobi

    a. Register an account for Gurobi. Please choose academic as the account type.

    b. Download the newest Gurobi Optimizer

    c. Get a free academic license

    d. Because Gurobi requires an academic domain, please install Gurobi on your computer and activate it using your license when you are on campus.

    e. In Rstudio,

    install.pacakges('slam')   (Because Gurobi depends on package 'slam')
    install.packages(directory, repo = NULL, method = 'source')
    (On Mac the default directory should be /Library/gurobi602/mac64/R/gurobi_6.0-2.tgz)
    (On Windows, it should end with /gurobi602/win64/R/gurobi_6.0-2.zip)
    library(gurobi)   (If no error occurs, then you have successfully installed Gurobi)

    f. Read the overview of R API

2. Apply both MIQP and Lasso to the given data. For MIQP, please choose the number of selected variables, k, as 8. For Lasso, you may use different λs. Please compare the number of non-zero coefficients and for each regression.

    Hint: For Lasso, you may use functions from glmnet instead of using gurobi.

3. Please compare the prediction error for each regression. Prediction error is defined as $\left\|X\tilde{\beta} - X\beta^0\right\|_2^2 / \; \|X\beta^0\|_2^2$,
    where $\tilde{\beta}$ is the estimation of the coefficients and β^0 is the real value of the coefficients.

Submission Instructions:

Name your report as **project3_gZ.pdf** (Z is your group number). Upload your report and your code as two separate files.

(We want to use speedgrader that requires separate files.)

Side Note:

MIQP is NP-hard. It is usually computational expensive. However, in practice, finding the optimal solution is not that time-consuming. It is proving the optimality that takes much time. If we care about performance more than optimality, we can stop the solver at a certain time limit and take the final solution.