1 Sparsity and Model Selection

Tool: Subset selection, LASSO

The Analytics Edge: Modern day datasets often have many predictor variables that might help predict an output of interest. For example in cancer diagnosis, we need to identify which genes (among many possibilities) can predict cancer. Likewise in economics, different sets of variables have been proposed by researchers to predict the economic growth of countries. In these settings, it is important to find which variables are relevant. Techniques for subset selection play a major role here to identify sparse predictive models.

1.1 Overview

We will focus on some more recent ideas for regression and classification that have been particularly designed for problems with large datasets and many predictor variables. Our interest in these problems is in the predictive power of the models where the objective is to get good out-of-sample (test set) predictions when there are many possible predictor variables. While it is possible to get good in-sample predictions by getting more complicated models, the concern is that there is a chance of overfitting the data.

1. Simpler models often tend to work better for out-of-sample predictions and so we will penalize models for excessive model complexity.
2. With the increase in computational power, we can partition the data set into training, validation and test sets and conduct model assessment and selection. The training set is used to estimate the model parameters. The validation set is used to do model selection while the test set is the evaluation set on which we will simply evaluate or check how the model performs.

1.2 Bias-variance tradeoff

We will discuss this in the context of regression. Suppose we have a training set with observations $\{(x_i, y_i)\}$ for $i = 1, \ldots, n$ where $x_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$. Assume that the true model is given as:

$$Y = f(X) + \epsilon,$$

where ϵ is the noise term with mean 0 and variance σ^2.

Suppose we use a least squares method with regression to develop a model $\hat{f}(x)$ to approximate $f(x)$ where the predicted values from the model are $\hat{y} = \hat{f}(x)$. While we choose the fit of the model to minimize the sum of squared errors in the training set, we would also like it to do well for points outside the training set (generalizable to the test set).

Suppose (X_0, Y_0) is a random test observation that might not necessarily be observed in the training set. We would really like to have a model, that would give a small value for

$$\mathbb{E}\left[(Y_0 - \hat{f}(X_0))^2\right]$$

where the expectation is taken over all possible realizations from the true model (note aside: here y_0 and $\hat{f}$ bears the randomness, while x_0 is a fixed quantity). We can now check that:

$$\mathbb{E}\left[(Y_0 - \hat{f}(X_0))^2\right] = \text{Variance}[\hat{f}(X_0)] + \sigma^2 + \mathbb{E}\left[(f(X_0) - \hat{f}(X_0))^2\right].$$

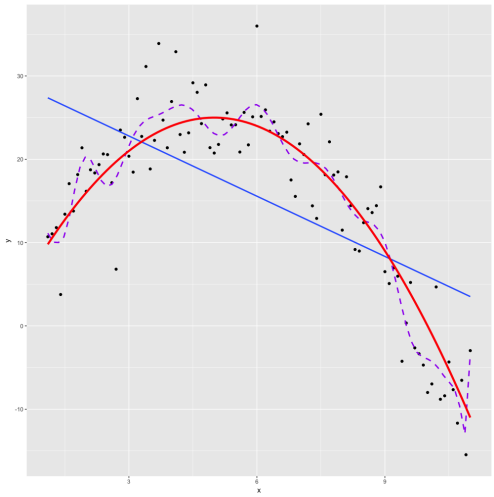

Figure 1: Blue line : linear regression (low variance and high bias); Purple curve: High degree polynomial regression (high variance and low bias); Red curve: the actual function f.

The test mean squared error is decomposed into three terms:

1. Variance of the estimator: Variance$[\hat{f}(X_0)]$;
2. Variance of the error term (irreducible error): σ^2;
3. Square of the bias of the estimator: $\mathbb{E}\left[(f(X_0) - \hat{f}(X_0))^2\right]$.

A complex model will typically have high variance of the estimator but low bias. On the other hand a simple model will have low variance of the estimator but a high bias. The key is to find the right balance between simplicity and complexity.

2 Subset selection

In classical statistical prediction, we have:

$$\text{Number of observations } n \gg \text{ Number of predictors } p$$

However, there are many problems in the era of the big data age where:

$$\text{Number of observations } n < \text{ or } \approx \text{ Number of predictors } p$$

For example in cancer diagnosis in terms of genes, the number of genes is very large and one needs to check among then to identify the genes to predict the chances of getting cancer. Note that when $n < p$ or $n \approx p$, there is a significant risk of over-fitting and there might be no longer an unique fit too. In such cases, one needs to decide which variables are important in making predictions and drop those variables that are less useful. We are also interested in identifying from a large set of variables, a much smaller subset that has the strongest effects on the response (this is fundamentally a combinatorial problem).

Figure 2: Increasing model complexity leads to lower training set error, but test set error may increae.

Question: Given a set of observations $\{(x_i, y_i)\}$ for $i = 1 \ldots, n$ with $x_i \in \mathbb{R}^p$, what is the best subset of predictors for the output?

A possible algorithm is provided below:

Algorithm:

1. Let M_0 denote a null model with no predictors (only the intercept). This model would simply predict the mean of the training set for each observation for a linear regression model (or the fraction of ones in logistic regression).
2. For $k = 1, \ldots, p$, do:
   (a) Fit all $\binom{p}{k}$ models that contains exactly k predictors and the intercept.
   (b) Pick the best among these models by choosing the model with the minimum sum of squared errors for linear regression or maximum log-likelihood for logistic regression. Call this model M_k
3. Choose the best model among M_0, M_1, ..., M_p using cross-validation errors or adjusted R-squared (linear regression) or AIC (logistic regression).

The complexity of this algorithm is exponential since we are solving for $O(2^p)$ linear or logistic regressions.

2.1 Forward stepwise selection

We discuss a computationally efficient algorithm to solve this problem in comparison to the best subset selection but this method is not guaranteed to solve the problem to optimality. The methods is based on a greedy algorithm.

1. Let M_0 denote a null model with no predictors (only the intercept). This model would simply predict the mean of the training set for each observation (or the fraction of ones in logistic regression).

2. For $k = 1, \ldots, p$, do:
   (a) Fit all models that augment the predictors in model M_{k-1} with exactly one more predictor (a total of $p - k + 1$ models are fit in step k).
   (b) Pick the best among these models by choosing the model with the minimum sum of squared errors for linear regression or maximum log-likelihood for logistic regression. Call this model M_k
3. Choose the best model among M_0, M_1, ..., M_p using cross-validation errors or adjusted R-squared (linear regression) or AIC (logistic regression).

The complexity of this algorithm is much lower since we are solving for $O(p^2)$ linear or logistic regressions. On the other hand there is no guarantee that the greedy algorithm will always find the optimal solution for any fixed k.

Another similar algorithm often uses is *backward stepwise selection* where one starts with all the predictors and then reduce one by one the least useful ones. The complexity of this algorithm is also $O(p^2)$, and does not guarantee an optimal solution.

3 Cross Validation

Cross validation is a technique often used to assess the quality of the model. A model is considered good if it has a low *test set error*. Unfortunately, often we do not have the liberty of having a large test set to validate our model. One method of model assessment here is *Cross Validation*.

3.1 Validation Set Approach

1. Divide the data randomly into 2 subsets (often roughly of equal size): the *training set* and the *validation set* or *hold-out set*.
2. Use the training set to fit the model, and the validation set to predict and then estimate Mean squared error (MSE).

Potential drawbacks:

1. The method depends on the points chosen, hence different choices may lead to starkly different estimated MSEs.
2. Since we are only using a subset of the available data set, the performance of the model is worse than it would be on a larger data set. And the error estimates tend to be larger.

3.2 Leave one out Cross Validation

This method compensates for the drawbacks of the *Validation set approach* yet keeping the same spirit.

1. For every $i \in I = \{1, \ldots, n\}$, train the model on the set $I \setminus \{i\}$.
2. Use this model to predict the ith response, say it is $\hat{y}_i$. and compute $\text{MSE}_i = (y_i - \hat{y}_i)^2$.

3. Compute cross validation error

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} \text{MSE}_i.$$

Advantages:

1. This method has far less bias, since we are fitting the model to $n-1$ of the points.
2. Does not change depending on the random sample like the validation set method.

The only potential drawback is that it may be computationally intensive: we need to fit n models.

3.3 k-Fold Cross Validation

This method tries to balance the drawbacks of the previous two methods.

1. Divide the data randomly into k subsets (folds) of (roughly) equal size.
2. Start with the first fold as a validation set and use the remaining $k-1$ folds to fit the model.
3. Compute the error of the fitted model in the held-out fold.
4. Repeat steps 2 and 3 by using the second, third and so on folds as the hold-out fold with the remaining $k-1$ folds to fit the model.
5. Average the error across all the k fitted models to estimate the cross-validation error.

Some of the commonly used choices for k are 5 or 10.

When we set $k = n$, this reduces to the leave out one cross validation scheme.

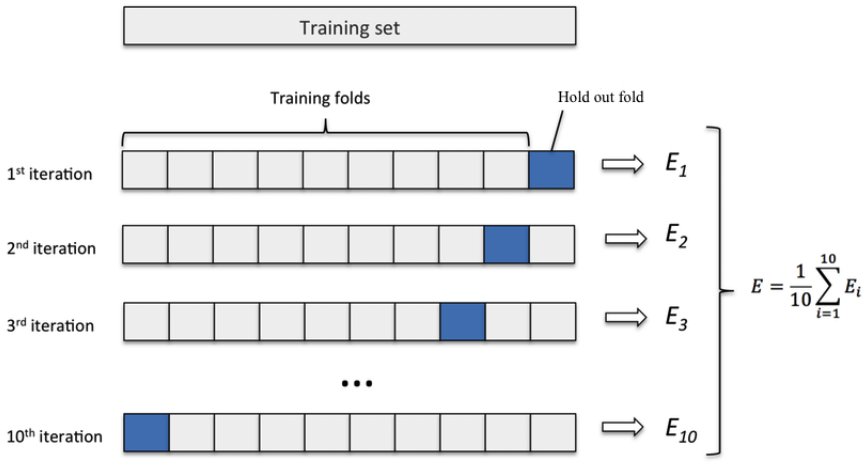

Figure 3: k-fold cross validation.

4 Least absolute shrinkage and selection operator (LASSO)

In standard linear regression, the problem we solve is:

$$\min_{\beta_0,\beta_1,\ldots,\beta_p} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2 = \min_{\beta_0,\beta_1,\ldots,\beta_p} \mathsf{RSS}(\boldsymbol{\beta}).$$

LASSO modifies the linear regression model by accounting for model complexity as follows where $\lambda \geq 0$ is a parameter:

$$\min_{\beta_0,\beta_1,\ldots,\beta_p} \mathsf{RSS}(\boldsymbol{\beta}) + \lambda \sum_{\mathrm{j}=1}^{\mathrm{p}} |\beta_{\mathrm{j}}|.$$

Here $\lambda \geq 0$ is a tuning parameter that provides a tradeoff between fitting the data (first term) with model complexity (second term in terms of non-zero values of the beta coefficients).

1. When $\lambda = 0$, LASSO reduces to standard linear regression.
2. When $\lambda \uparrow \infty$, the second term dominates and LASSO will make all the beta coefficients for the predictor variables go to zero.

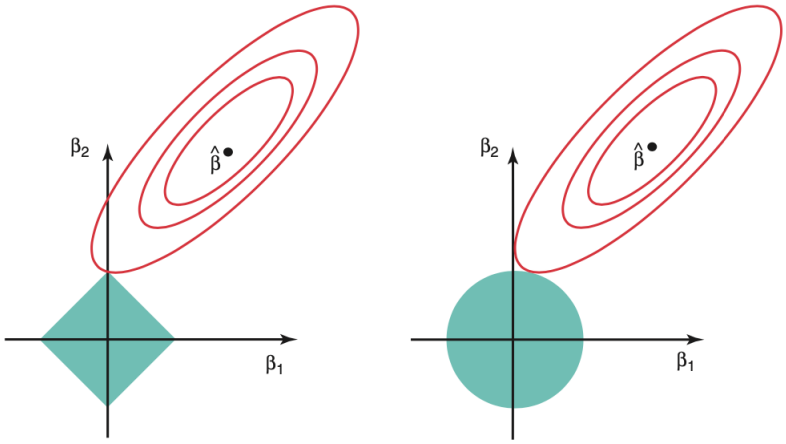

Figure 4: LASSO vs. Ridge regression.

The LASSO method for linear regression was popularized in 1996 by the paper "Regression Shrinkage and Selection via the lasso" by Tibshirani. The paper has around 36000 citations as of October 2020. The objective coefficient in LASSO is convex and tries to roughly promote sparsity. One of the advantages of LASSO is that since it is convex, the local optimum is the global optimum and there are efficient ways to solve the problem to optimality. However this objective function is not differentiable unlike standard linear regression.

To choose the λ values, we use a grid of possible values and compute the cross-validation error for each value of λ. We can then choose the λ with the smallest cross-validation error. You can then refit the final model using all the observations for the selected value of λ.

4.1 Related formulation: Best subset selection problem

$$\begin{aligned} \min_{\beta_0,\beta_1,\ldots,\beta_p} \quad & \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip}\right)^2 \\ \text{s.t.} \quad & \sum_{j=1}^{p} \mathbb{I}(\beta_j \neq 0) \leq k \end{aligned}$$

where $\mathbb{I}(\beta_j \neq 0) = 1$ if $\beta_j \neq 0$ and 0 otherwise. This can be reformulated as a quadratic integer optimization problem as follows:

$$\begin{aligned} \min_{\beta_0,\beta_1,\ldots,\beta_p} \quad & \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip}\right)^2 \\ \text{s.t.} \quad & \sum_{j=1}^{p} z_j \leq k \\ & z_j \in \{0,1\}, \quad j = 1,\ldots,p \\ & -M z_j \leq \beta_j \leq M z_j, \quad j = 1,\ldots,p, \end{aligned}$$

where M is a sufficiently large positive value.

4.2 Ridge regression

Ridge regression is very similar to least squares, and uses an idea alike LASSO to regularize. Its usage pre-dates the use of LASSO. The ridge regression coefficients are found by solving:

$$\min_{\beta_0,\beta_1,\ldots,\beta_p} \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip}\right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = \min_{\beta_0,\beta_1,\ldots,\beta_p} \mathsf{RSS}(\boldsymbol{\beta}) + \lambda \sum_{\mathsf{j}=1}^{\mathsf{p}} \beta_{\mathsf{j}}^2$$

where $\lambda \geq 0$ is a tuning parameter.

As with least squares, ridge regression seeks coefficient estimates that fit the data well, by making the RSS small. How- ever, the second term, $\lambda \sum_{j=1}^{p} \beta_j^2$ called a shrinkage penalty, is small when are close to zero $\beta_0, \beta_1, \ldots, \beta_p$, and so it has the effect of shrinking the estimates of β_j towards zero. The tuning parameter serves to control the relative impact of these two terms on the regression coefficient estimates. When $\lambda = 0$, the penalty term has no effect, and ridge regression will produce the least squares estimates. However, as $\lambda \to \infty$ the impact of the shrinkage penalty grows, and the ridge regression coefficient estimates will approach zero. Unlike least squares, which generates only one set of coefficient estimates, ridge regression will produce a different set of coefficient estimates for each value of λ. Selecting a good value for λ is critical.

4.2.1 Properties of ridge regression

1. Ridge regressions advantage over least squares is rooted in the bias-variance trade-off. As λ increases, the flexibility of the ridge regression fit decreases, leading to decreased variance but increased bias.

2. Ridge regression will include all p predictors in the final model. The penalty $\lambda \sum_{j=1}^{p} \beta_j^2$ will shrink all of the coefficients toward zero, but it will not set any of them exactly to zero.

3. Ridge regression also has substantial computation advantage over best subset selection, which requires searching through 2^p models.

4.3 Comparing LASSO and ridge regression

1. The LASSO has a major advantage over ridge regression, in that it produces simpler and more interpretable models that involve only a subset of the predictors.
2. Neither ridge regression nor the LASSO will universally dominate the other.
3. In general, one might expect the LASSO to perform better in a setting where a relatively small number of predictors have substantial coefficients, and the remaining predictors have coeffiients that are very small or that equal zero.
4. Ridge regression will perform better when the response is a function of many predictors, all with coefficients of roughly equal size.
5. A technique such as cross-validation can be used in order to determine which approach is better on a particular data set.

5 Cross-country growth regression: application in econometrics

Economists are interested in understanding the factors such as economic policy, political and other factors that are linked to the rate of economic growth. Economists have studied this problem by identifying a few factors at a time. For example some of these factors include the initial GDP, degree of capitalism, population growth, equipment investment - each of which has been proposed to try and explain the rate of economic growth in countries. In an influential paper in 1991, **Economic growth in a cross-section of countries**, in *The Quarterly Journal of Economics*, Robert Barro used data from various countries over the period 1960 to 1985 to show that the growth rate is positively related to school enrolment rates and negatively related to the initial 1960 level of real per capita GDP. For example, this might be partly argued by the fact that poorer countries with low capital-to-labor ratios have higher growth rates. This paper had close to 18000 citations as of October 2019.

However there have been many such variables that have been proposed and hence it is often hard to assess which variables are really correlated with growth. While there is a proliferation of possible explanatory variables and little guidance from economic theory on how to choose among these variables, it is possible to use ideas on subset selection from linear regression to aid towards this. The goal of these methods is to help and identify variables that are most important using cross-country growth data. Economists have found that it is quite possible for one of the variables (say x_1) to be significant in a regression when x_2 and x_3 are included but it becomes insignificant when a new variable x_4 is included. In these cases, it is useful to obtain some guidance on which variables are really important and to validate if such a dependence on economic growth is really robust.

We use a dataset that was used in the paper:s **I just ran two million regressions** by Sala-I-Martin and **Model uncertainty in cross country growth regression** by Fernandez et. al. The dataset has 41 possible explanatory variables with 72 countries. The data description is provided next. Note that if you try all 2^{41} possible combinations, it leads to around 2 trillion possibilities.

1. Country: Country name in abbreviation
2. y numeric: Economic growth 1960-1992 as from the Penn World Tables Rev 6.0
3. Abslat numeric: Absolute latitude
4. Spanish numeric: Spanish colony dummy
5. French numeric: French colony dummy
6. Brit numeric: British colony dummy
7. WarDummy numeric: War dummy
8. LatAmerica numeric: Latin America dummy
9. SubSahara numeric; Sub-Sahara dummy
10. OutwarOr numeric: Outward Orientation
11. Area numeric: Area surface
12. PrScEnroll numeric: Primary school enrolment
13. LifeExp numeric: Life expectancy
14. GDP60 numeric: Initial GDP in 1960
15. Mining numeric: Fraction of GDP in mining
16. EcoOrg numeric: Degree of capitalism
17. YrsOpen numeric: Number of years having an open economy
18. Age numeric: Age
19. Buddha numeric: Fraction Buddhist
20. Catholic numeric: Fraction Catholic
21. Confucian numeric: Fraction Confucian
22. EthnoL numeric: Ethnolinguistic fractionalization
23. Hindu numeric: Fraction Hindu
24. Jewish numeric: Fraction Jewish
25. Muslim numeric: Fraction Muslim
26. PrExports numeric: Primary exports 1970
27. Protestants numeric: Fraction Protestants
28. RuleofLaw numeric: Rule of law
29. Popg numeric: Population growth
30. WorkPop numeric: workers per inhabitant
31. LabForce numeric: Size of labor force
32. HighEnroll numeric: Higher education enrolment
33. PublEdupct numeric: Public education share
34. RevnCoup numeric: Revolutions and coups
35. PolRights numeric: Political rights
36. CivlLib numeric: Civil liberties
37. English numeric: Fraction speaking English
38. Foreign numeric: Fraction speaking foreign language
39. RFEXDist numeric: Exchange rate distortions
40. EquipInv numeric: Equipment investment
41. NequipInv numeric: Non-equipment investment
42. stdBMP numeric: stand. dev. of black market premium
43. BlMktPm numeric: black market premium

Figure 5: Data description.