Fourth essay Raul Adell

This essay may contain ideas given in class complemented by the book given on the references of the course as well as content from Stanford's online ITSL course given by R. Tibshirani and T. Hastie. Book section: 6. Linear Regression Selection and Regularization, continuation of the last essay. Along with reading the book chapter, and contrasting with online videos and class notes, this is what I considered most important.

We can really improve our linear model and overcome many problems like noise, collinearity, and high dimensionality with some alternative fitting procedures. In this essay, we are going to analyze how. There are three main strategies.

- Feature subset selection: identify a subset of the p predictors related to response.
- Shrinkage: use all p predictors but estimated coefficients are shrunk towards 0. This is known as regularization and reduced variance.
- Dimensionality reduction: Use M<p of different linear combinations/projections of the p predictors.

Feature subset selection

We already tackled subset selection in the last essay. However, let's make a quick recap. Selecting the best model up to the 2^p models that can be built with p predictors is not trivial since that number grows very fast. Thus, we may take a stepwise approach. This can be forward, by starting from 0 predictors and adding the one that yields the best result in terms of RSS and keeps iterating; backward, more or less in the reverse direction or done hybridly.

However, choosing the optimal model is not an easy task. RSS and R^2 are measures that tell us how a model is doing in the training data. What we really are interested in is how the model does in reality, that is, in the test error. If our model does very well on test data it could mean that is overfitting and thus losing its ability to generalize well in test data. The opposite can happen, our model could be undertrained so it can't perform well on new data. It is important to state that we cannot know test error, only estimate it.

We can either indirectly estimate test error by making an adjustment to training error to account for bias due to overfitting or rather directly estimate the test error using a validation set approach or a cross-validation approach.
This one last direct technique is very important, so an in-depth
Validation and cross-validation are techniques used to evaluate the performance of machine learning models. Validation involves splitting the available data into two subsets: a training set used to train the model, and a validation set used to evaluate its performance. The model is trained on the training set, and its performance is evaluated on the validation set. This process is repeated multiple times, with different subsets of the data used for training and validation, in order to get a more accurate estimate of the model's performance.
Cross-validation is a variation of validation in which the available data is split into multiple subsets, or "folds," with each fold used for both training and validation in turn.

In previous essays, we discussed how the R^2 could never decrease when a feature is added to the model. Now we'll see some measures of training error, which can increase when a feature is added to the model.

The following measures indirectly estimate test error.

$$C_p = \frac{1}{n}\left(\text{RSS} + 2d\hat{\sigma}^2\right) \qquad \text{BIC} = \frac{1}{n}\left(\text{RSS} + \log(n)d\hat{\sigma}^2\right)$$

$$AIC = -2ln(L) - 2d = \frac{1}{n}(RSS + 2d\hat{\sigma}^2) \qquad \text{BIC} = \frac{1}{n}\left(\text{RSS} + \log(n)d\hat{\sigma}^2\right)$$

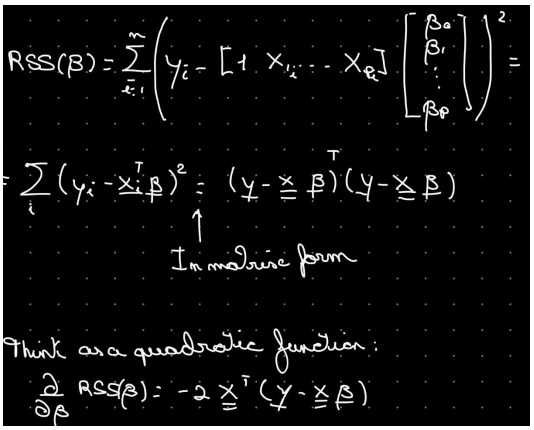
$$\text{Adjusted } R^2 = 1 - \frac{\text{RSS}/(n-d-1)}{\text{TSS}/(n-1)}$$

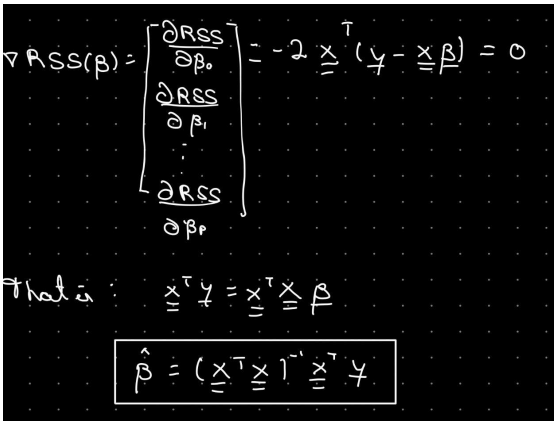
Where d is the degrees of freedom of our model, n is the total number of observations, the estimation of the variance is obtained using the full model containing all predictors, L is the maximum value of the likelihood function of the model. These all are indirect estimations of test error. Except for the adjusted R^2, the lower the value, the better. The intuition behind the adjusted R^2 is that adding additional noise variables to the model would lead to nearly no increase in R^2, the d term will penalise that. Thus, a higher adjusted R^2 means a better model.

An alternative approach is to estimate test error directly, using the validation set and cross-validation methods. It is very practical and the preferred method since there is no need to estimate anything, which is beneficial since sometimes it is challenging to estimate the error variance.
If the test error is relatively flat it means that if we repeated a validation approach using different sets or folds our lowest estimated test error would change, thus leading to another 'best' model. In this setting, we can select a model using the one-standard-error rule. The rule involves selecting the simplest model whose performance is within one standard error of the performance of the best model. This is done to avoid overfitting, which can occur when a model is too complex and fits the training data too closely, resulting in poor performance on new data (generalization).

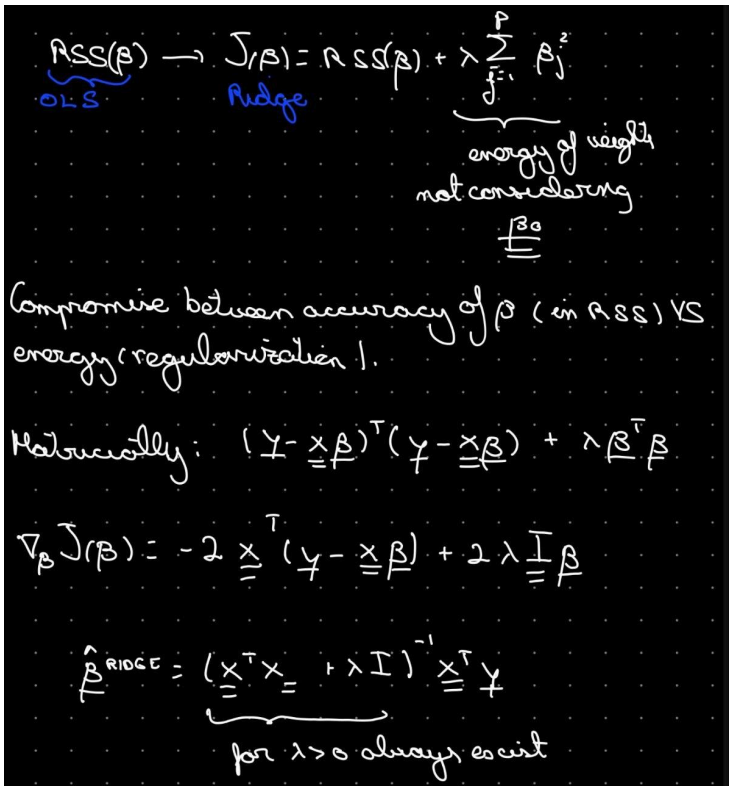
Shrinkage

The linear regression with OLS fitting method has the following mathematical expressions.





Obviously, if the matrix **x** is not full rank, problems arise when doing the inverse. This is the case of collinearity, explained in the last essay. As an alternative, we can fit a model containing all p predictors using a technique that constrains or regularizes the coefficient estimates, that is, shrinks them towards 0. This can significantly reduce variance, making the model rebut to collinearity and noisy data.

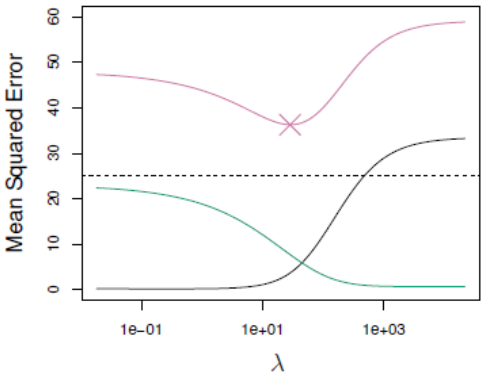


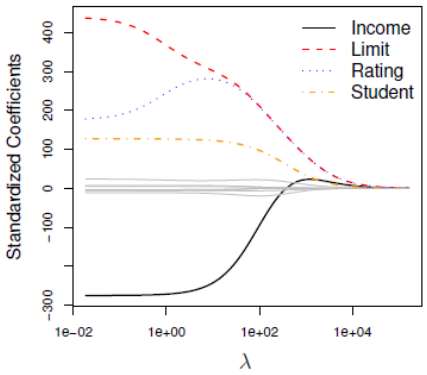
Ridge regression minimizes a similar cost function, but not exactly RSS. It adds a shrinking penalty with λ as a tuning parameter. When λ=0 Ridge regression is exactly OLS. As λ→∞, the impact of the shrinkage penalty grows. We can see this from the denominator of β^{RIDGE}. Also, the inverse of that matrix will always exists, solving collinearity problems.

The best value of λ can be obtained by cross-validation.

While standard least squares coefficient estimates are scale equivariant, ridge regression coefficient estimates can change when multiplying a given predictor by a constant. Thus, one should standardize the predictors before applying ridge regression if they are not on the same scale.

Its advantage over least squares fitting 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. Thus, a sweet spot in terms of minimum $MSE\ (red)\ =\ bias^2(black)\ +\ variance\ (green)$ can be found, much smaller than for OLS, which corresponds to λ=0.





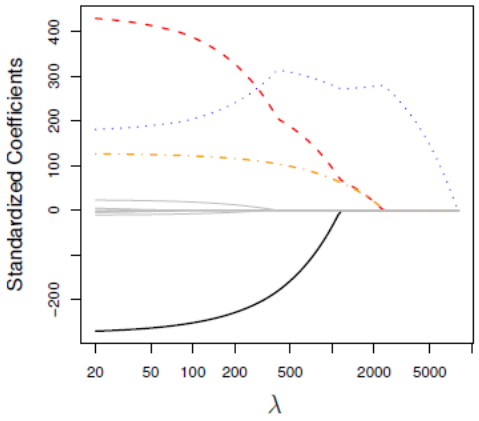
It is common to plot the magnitude of the coefficient as a function of log(λ). That plot its named the regularization path plot. Despite not having this exact plot we see how coefficients tend to 0 as λ→∞. However, for a finite value of λ, the coefficients that have shrunk the most will not be 0, but practically 0. This may be inconvenient if p is very large and a given subset of predictors are not giving much information.

The next shrinking regression deals with that by instead of adding a penalty based on the squares of the coefficients it adds one proportional to the absolute value of them. More precisely the LASSO (least absolute shrinkage and selection operator) regression has the following cost function.

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|\ =\mathrm{RSS}+\lambda\sum_{j=1}^{p}|\beta_j|$$

Mathematically, the only difference is the l_1 norm of the coefficients have been used instead of the l_2. This forces some coefficient estimates to shrink not towards but to exactly 0, thus performing variable subset selection when λ is sufficiently large. We say that the lasso yields sparse models, that is, models that involve only a subset of the variables. Thus, it produces simpler and more interpretable models than Ridge regression.

When λ = 0, then the lasso simply gives the least squares fit, and when λ becomes sufficiently large, the lasso gives the null model which all coefficient estimates equal zero. From a Bayesian perspective, this penalty term can be interpreted as a way of introducing prior knowledge about the magnitude of the coefficients into the regression model.

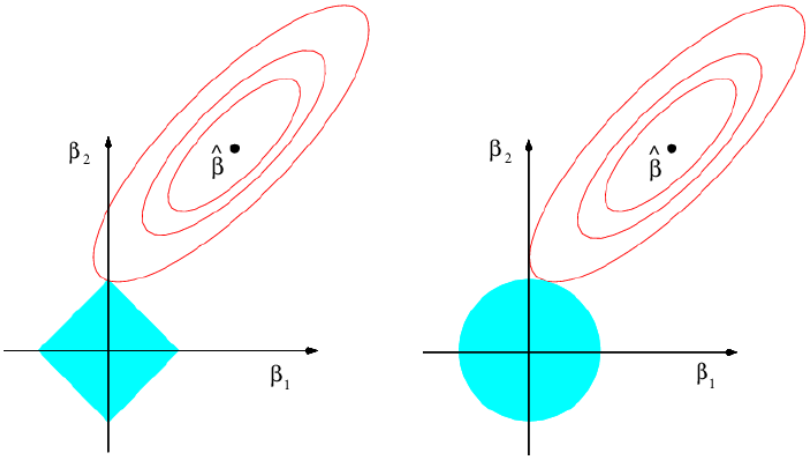


in

Despite λ having different meanings in LASSO and Ridge we could plot their coefficients together by using R^2 on training data as a common x value.

The formulation of the minimizing function, that is, RSS for OLS regression and its variations for Ridge and LASSO are built by Lagrange multipliers. That is, one may understand this as making minimum the RSS subject to a given restriction. The restriction is that the sum of all the squared coefficient estimates is less or equal to quantity s for Ridge or the sum of all the absolute value coefficient estimates is less or equal to quantity s.

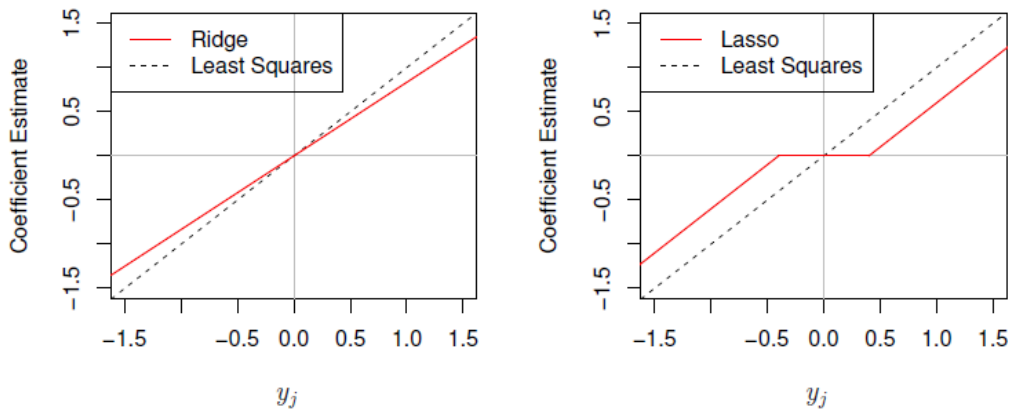
Imagine a p=2 scenario with the red lines being the contour of RSS, that is, the parameter space points where RSS has the same value. The constrained optimization problem is telling us to take the combination of parameters that intersects the contour of RSS with our restriction. The restriction is a diamond of height s (because of the absolute value) in LASSO and a circle of radius s (because of the square) in Ridge.



Notice that the corners of the diamond are parameter space points with some coordinates equal to 0. Hitting a corner is much easier on the diamond is much easier than hitting anywhere else. In the circle hitting one of those points is not generally easy. That's why LASSO shrinks coefficient estimates to 0 and why Ridge fails to do so.

Neither Ridge regression nor the LASSO will universally dominate the other. 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 coefficients that are very small or that equal zero. Ridge regression will perform better when the response is a function of many predictors, all with coefficients of roughly equal size. A technique such as cross-validation can be used in order to determine which approach is better on a particular data set.

Let's take a deeper look at what these shrinkage methods do with the coefficient estimates. A simple special case with n=p and X being the identity yields the following:



We can see that ridge regression and the lasso perform two very different types of shrinkage. In Ridge regression, each least squares coefficient estimate is shrunk by the same proportion. In contrast, the Lasso shrinks each least squares coefficient towards zero by a constant amount, λ/2; the least squares coefficients that are less than λ/2 in absolute value are shrunk entirely to zero. The type of shrinkage performed by the lasso in this simple setting is known as soft thresholding. The fact that some lasso coefficients are shrunk entirely to zero explains why the lasso performs feature selection.

Dimensionality reduction

The methods that we have discussed so far in this chapter have controlled variance in two different ways, either by using a subset of the original variables or by shrinking their coefficients toward zero. All of these methods are defined using the original predictors, X1,X2, . . . ,Xp. We now explore a class of approaches that transform the predictors and then fit a least squares model using the transformed variables.

Let M<p and Z_i be a linear combination of the original predictors.

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

We can fit these new M predictors using least squares, obtaining the following relationships:

$$\sum_{m=1}^{M} \theta_m z_{im} = \sum_{m=1}^{M} \theta_m \sum_{j=1}^{p} \phi_{jm} x_{ij} = \sum_{j=1}^{p} \sum_{m=1}^{M} \theta_m \phi_{jm} x_{ij} = \sum_{j=1}^{p} \beta_j x_{ij}$$

where

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}.$$

We will refer to these techniques as dimension reduction methods. It is important to state that these methods are not feature selection methods. Thus, dimensionality reduction can be thought as a special case of a linear regression model where coefficients β_j have a given expression. To approaches for this task are PCA and PLS.

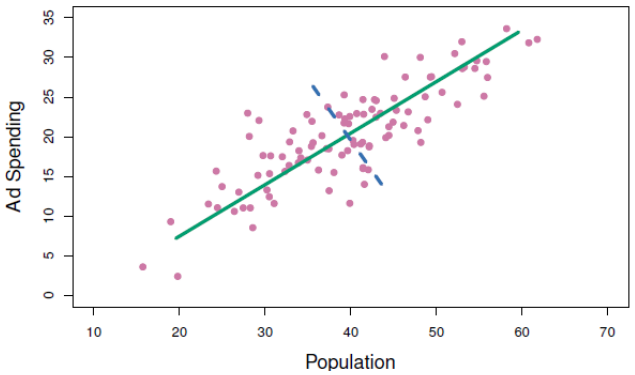
PCA (Principal Component Analysis) is a technique used in multivariate data analysis to transform a set of correlated variables into a smaller set of uncorrelated variables called principal components. The goal of PCA is to find a new set of variables that capture as much of the variation in the original data as possible while reducing its dimensionality. Maximum variation in data can be translated into maximum information using basic Information Theory concepts.

Mathematically, PCA involves calculating the eigenvectors and eigenvalues of the covariance matrix of the original data. The covariance matrix describes the pairwise relationships between the variables and provides information about the amount of variation in the data that can be explained by each variable.

To obtain the components, the covariance or correlation matrix is broken down into singular values. In this decomposition, the eigenvalues and the eigenvectors are determined. The principal components are linear combinations of the original variables, and there are as many as variables, with the coefficients of these combinations being the eigenvectors associated with the covariance or correlation matrix of the original variables. The ith principal component, PCi, is given by:

$PC_i = u_{i1}X_1 + \cdots + u_{ip}X_p$ where $u_i = (u_{i1}, u_{i2}, \ldots, u_{ip})$ is the eigenvector of the covariance or correlation matrix associated with the eigenvalue λ_i. The eigenvalues are ordered from largest to smallest and the principal component i has a variance equal to the eigenvalue λ_i. The first component is characterized by explaining the highest proportion of variance associated with the data, the projection of the data on it provides maximum variability.

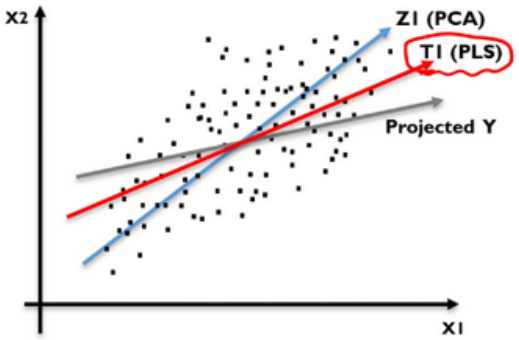
When using PCA it is recommended to standardize each predictor so all variables are on the same scale.



In this picture we can see in green the first PC. This is the direction among data varies the most, or equivalently, normal distance of points to the green line is minimum. Blue represents the second PC, completely perpendicular (uncorrelated) to first PC.

Extra information about correlation, variance, inertia and more here.

When applying PCA we are assuming the directions in which X1,X2, . . . ,Xp show the most variation are the directions that are associated with Y. This, in general, is not guaranteed. We say that the directions are chosen in an unsupervised way, since response Y is not used at all. PLA or partial least squares is a supervised alternative of PCA that attempts to find directions that help explain both the response and the predictors.



In practice, PLA does not perform better than PCA because features are already expected to be relevant in explaining the response variable.

High dimensionality considerations

In high-dimensionality, that is n<<p classical approaches such as least squares linear regression are not appropriate in this setting. The problem is simple: when p > n or p ≈ n, a simple least squares regression line is too flexible and hence overfits the data. In this scenario, regardless of whether or not there truly is a relationship between the features and the response, least squares will yield a set of coefficient estimates that result in a perfect fit to the data, such that the residuals are zero, even though the features are completely uncorrelated to the response. This results in overfitting and poor performance in terms of generalization into test data. Also, the multicollinearity issue is extreme in high dimensions. Alternative approaches such as the ones described in this essay are required to overcome what is known as the curse of dimensionality.

Forward stepwise selection, Ridge, LASSO and PCA are particularly useful for performing regression in the high-dimensional setting.
When it comes to shrinking methods, this statements have to be taken into acccount:

1. Regularization or shrinkage plays a key role in high-dimensional problems.
2. Appropriate tuning parameter selection is crucial for good predictive performance.
3. The test error tends to increase as the dimensionality of the problem (i.e. the number of features or predictors) increases, unless the additional features are truly associated with the response.