ECON 7310 Elements of Econometrics
Week 12: Prediction with Many Regressors and Big Data

David Du[1]

[1] University of Queensland

Draft

Outline

- OLS with many regressors
- Ridge regression and LASSO
- Principal components analysis
- Application to prediction of school test scores

What is “Big Data”?

“Big Data” means many things:

- Data sets with many observations (millions).
- Data sets with many variables (thousands, or more).
- Data sets with nonstandard data types, like text, voice, or images.

“Big Data” also has many different applications:

- Prediction using many predictors.
  - Given your browsing history, what products are you going to buy?
  - Given your loan application profile, how likely are you to repay?
- Prediction using highly nonlinear models (need many observations).
- Recognition problems, like facial and voice recognition.

What is “Big Data”?

“Big Data” has different jargon, which makes it seem very different than statistics and econometrics

- Machine learning: when a computer (machine) uses a large data set to learn (e.g., about your online shopping preferences)

But at its core, machine learning builds on familiar tools of prediction. This chapter focuses on one of the major big data applications, prediction with many predictors. We treat this as a regression problem, but with many predictors we need new methods that go beyond OLS.

For prediction, we do not need – and typically will not have – causal effects (coefficients) to estimate.

The Many-Predictor Problem

The many-predictor problem:

- The goal is to provide a good prediction of some outcome variable Y given a large number of X's, when the number of X's (k) is large relative to the number of observations (n) – in fact, maybe $k > n$!
- The goal is good out-of-sample prediction.
  - The *estimation sample* is the n observations used to estimate the prediction model.
  - The prediction is made using the estimated model, for an *out-of-sample* (OOS) observation – an observation not in the estimation sample.

The Predictive Regression Model

The *standardized predictive regression model* is the linear model, with the exception that all the X's are normalized (standardized) to have a mean of zero and a standard deviation of one, and Y is deviated from its mean:

$$Y = \beta_1 X_1 + \cdots + \beta_k X_k + u \tag{1}$$

- Assume $E[Y|X_1, ..., X_k] = \beta_1 X_1 + \cdots + \beta_k X_k$, so $E[u|X_1, ..., X_k] = 0$.
- Because all the variables, including Y, are deviated from their means, the intercept is zero – so is omitted from (1).
- (1) allows for the X's being squares, cubes, logs, interactions, etc.
- Throughout this lecture, we use standardized X's, demeaned Y's, and the standardized predictive regression model (1).

The Mean Squared Prediction Error

The *Mean Squared Prediction Error* (MSPE) is the expected value of the squared error made by predicting Y for an observation not in the estimation data set:

$$\text{MSPE} = E[Y^{OOS} - \hat{Y}(X^{OOS})]^2$$

- Y is the variable to be predicted.
- X denotes the k variables used to make the prediction, (X^{OOS}, Y^{OOS}) are the values of X and Y in the out-of-sample data set.
- The prediction $\hat{Y}(X^{OOS})$ uses a model estimated using the estimation data set, evaluated at X^{OOS}.

The MSPE measures the expected quality of the prediction made for an OOS observation. We use MSPE as a measure of predictive accuracy.

The First Least Squares Assumption for Prediction

For prediction, it does not matter whether model coefficients have a causal interpretation – so we do not need the first least squares assumption for causal inference, which is defined in terms of a causal effect.

- In the predictive model (1), β is defined to be the coefficient in the (linear) conditional expectation, $E[Y|X]$.

But it does matter that data for which we will be making the prediction is similar to the data used to estimate the model:

The First Least Squares Assumption for Prediction

(X^{OOS}, Y^{OOS}) are drawn from the same distribution as the estimation sample, (X_i, Y_i), $i = 1, ..., n$.

The Oracle Prediction

The *oracle prediction* is the best-possible prediction – the prediction that minimizes the MSPE – if you knew the joint distribution of Y and X.

It is easy to show that the oracle prediction is the conditional expectation, $E(Y^{OOS}|X = X^{OOS})$.

- $E(Y^{OOS}|X = X^{OOS})$ is not a feasible prediction as it is unknown (true population regression parameter).
- It is the benchmark against which to judge all feasible predictions.

In this lecture we focus on prediction based on linear model. We adopt the usual linear regression assumption:

$$E[Y|X] = \sum_{j=1}^{k} \beta_j X_j \text{ and } E[u|X] = 0$$

The MSPE for the Predictive Regression Model

OOS value of Y: $Y^{OOS} = \beta_1 X_1^{OOS} + \cdots + \beta_k X_k^{OOS} + u^{OOS}$.

Prediction: $\hat{Y}^{OOS} = \hat{\beta}_1 X_1^{OOS} + \cdots + \hat{\beta}_k X_k^{OOS}$.

Prediction error:

$$Y^{OOS} - \hat{Y}^{OOS} = (\beta_1 - \hat{\beta}_1) X_1^{OOS} + \cdots + (\beta_k - \hat{\beta}_k) X_k^{OOS} + u^{OOS}$$

Let $E(u^{OOS})^2 = \sigma_u^2$. The MSPE for the predictive regression model is

$$\mathrm{MSPE} = \sigma_u^2 + E[(\hat{\beta}_1 - \beta_1) X_1^{OOS} + \cdots + (\hat{\beta}_k - \beta_k) X_k^{OOS}]^2$$

The term σ_u^2 is the MSPE of the oracle forecast – cannot be beat!

The second term is the squared bias of the prediction error, which arises because the β's are unknown, so must be estimated using the estimation sample.

The MSPE of OLS

Suppose the β's are estimated by OLS. In the standardized predictive regression model, the MSPE of OLS is approximately

$$\mathrm{MSPE}_{OLS} \approx (1 + k/n)\, \sigma_u^2$$

This approximation holds if u is homoskedastic and k/n is small.

- For a given n, the MSPE of OLS increases linearly with the number of predictors. A big problem with many predictors!
- As OLS is unbiased, $\mathrm{MSPE}_{OLS} = V(\hat{\beta}^{OLS})$.
- Is there an estimator for which the MSPE increases more slowly than OLS, as more predictors are added?
  - We need such an estimator for hundreds or thousands of predictors.

The Principle of Shrinkage

James and Stein (1961) developed the first estimator that achieved this goal, which could reduce the MSPE, relative to OLS, by allowing the estimator to be biased in the right way.

- When the *X*'s are uncorrelated, these estimators are biased towards zero – or "shrunk" towards zero – and have the form,

  $$\hat{\beta}^{JS} = c\hat{\beta}^{OLS}$$

  where $0 < c < 1$ and "JS" stands for James-Stein.
- But how could introducing bias possibly help?

The Principle of Shrinkage

The James-Stein shrinkage estimator:

$$\hat{\beta}^{JS} = c\hat{\beta}^{OLS}$$

where $0 < c < 1$.

As c gets smaller:

- The squared bias of the estimator increases.
- But the variance decreases.
- This produces a bias-variance trade-off. If k is large, the benefit of smaller variance can beat out the cost of larger bias, for the right choice of c – thus reducing the MSPE.

The estimators introduced below all have a shrinkage interpretation.

Estimating the MSPE

The MSPE is a bit tricky to estimate – it is not just the regression SER, it is for an out-of-sample, not in-sample, observation.

Split-Sample Estimation of the MSPE

This method simulates the out-of-sample prediction exercise – but using only the estimation sample (which is all you have!):

1. Estimate the model using half the estimation sample.
2. Use the estimated model to predict Y for the other half of the data – called the “reserve” or “test” sample – and calculate the prediction error.
3. Estimate the MSPE using the prediction errors for the test sample:

$$\widehat{\text{MSPE}}_{\text{split-sample}} = \frac{1}{n_{\text{test}}} \sum_{\text{obs in test subsample}} (Y_i - \hat{Y}_i)^2$$

Estimating the MSPE by *m*-Fold Cross Validation

The split-sample estimate typically overstates the MSPE because the model is estimated on only 50% of the data.

This problem is reduced by using *m-fold cross validation*.

m-Fold Cross Validation ($m = 10$)

1. Estimate the model on 90% of the data and use it to predict the remaining 10%.
2. Repeat this on the remaining 9 possible subsamples (so there is no overlap on the test samples).
3. Estimate the MSPE using the full set of out-of-sample predictions.

Details on this algorithm in Key Concept 14.1.

Ridge Regression

The *ridge regression* estimator minimizes the penalized sum of squares

$$S^{Ridge}(b; \lambda_{Ridge}) = \underbrace{\sum_{i=1}^{n}(Y_i - b_1 X_{1i} - \cdots - b_k X_{ki})^2}_{(1)} + \underbrace{\lambda_{Ridge} \sum_{j=1}^{k} b_j^2}_{(2)}$$

where $\lambda_{Ridge} \geq 0$ is called the *shrinkage parameter*.

- (1) is the usual sum of squared residuals.
- (2) is called a *penalty term* as it penalizes the estimator for choosing a large estimate of β.
- (1) + (2) is called the *penalized sum of squared residuals*.

Ridge Regression

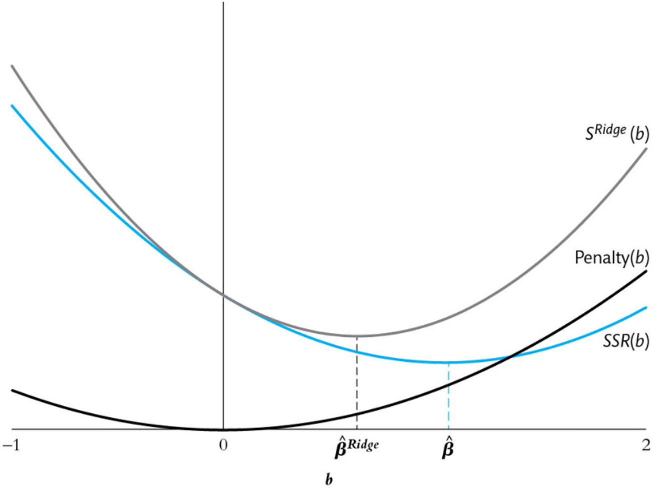
The penalty term shrinks the ridge estimator toward 0, and the magnitude of shrinkage depends on the value of λ_{Ridge}.

- $\lambda_{Ridge} = 0$, ridge estimator = OLS estimator
- Larger $\lambda_{Ridge} \Longrightarrow$ greater penalty $\Longrightarrow$ greater shrinkage toward 0.

Ridge estimator vs. OLS estimator

- Both ridge and OLS estimator have closed-form expressions.
- The presence of the penalty term introduces bias into the ridge estimator, but OLS is unbiased.
- Ridge regression can result in large improvements in MSPE compared to OLS by reducing the variance of the estimator.
- When $k > n$, ridge estimator can be computed, but OLS is infeasible (perfect multicollinearity).

Ridge Regression in a Picture



Choosing the Ridge Regression Penalty Factor

It would seem natural to choose λ_{Ridge} by minimizing (b, λ_{Ridge}) – but doing so would choose $\lambda_{Ridge} = 0$, which reduces to OLS! Instead, we...

Choose λ_{Ridge} using *m*-Fold CV

1. Choose a candidate value of λ_{Ridge}, say $\lambda^{(1)}$.
2. Apply the procedure described in Key Concept 14.1 to ridge regression with $\lambda_{Ridge} = \lambda^{(1)}$, and obtain the *m*-fold cross validation MPSE, $\widehat{\text{MPSE}}^{(1)}$.
3. Repeat steps 1-2 for candidate values of λ_{Ridge}, $\lambda^{(2)}, ..., \lambda^{(L)}$, and obtain $\widehat{\text{MPSE}}^{(2)}, ..., \widehat{\text{MPSE}}^{(L)}$.
4. An estimator of λ_{Ridge} is the one in $(\lambda^{(1)}, ..., \lambda^{(L)})$ that gives the smallest *m*-fold CV MPSE among $(\widehat{\text{MPSE}}^{(1)}, ..., \widehat{\text{MPSE}}^{(L)})$.

Example: Predicting School-level Test Scores

Data set: a school-level version of the California elementary district data set, augmented with additional variables describing school, student, and district characteristics.

The full data set has 3932 observations. Half of those (1966) are used now – the remaining 1966 are reserved for an out-of-sample comparison of the ridge and other prediction methods, done later.

The data set has 817 predictors...

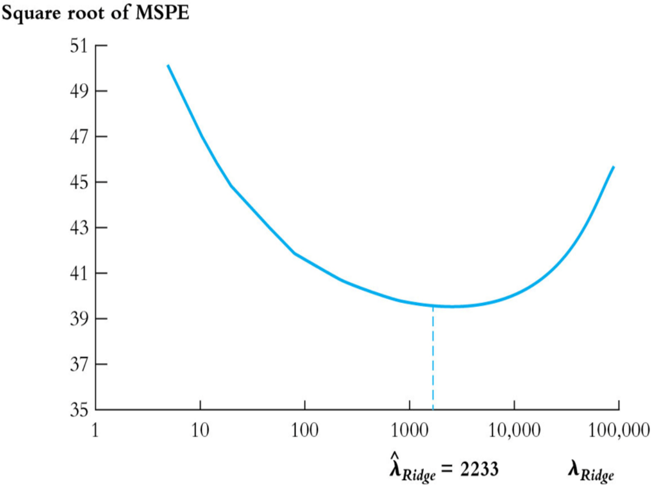
λ_{Ridge} is estimated by minimizing the 10-fold cross-validated MSPE.

The resulting estimate of the shrinkage parameter is $\hat{\lambda}_{Ridge} = 2233$.

Root MSPE's: OLS $=$ 78.2, Ridge $=$ 39.5.

Ridge results cuts the square root of the MSPE in half, compared to OLS!

Predicting School-level Test Scores: Ridge Regression



The Lasso

The Lasso estimator shrinks the estimate towards zero by penalizing large absolute values of the coefficients.

The Lasso estimator minimizes a penalized sum of squares, where the penalty term is the sum of the absolute values of the coefficients:

$$S^{Lasso}(b; \lambda_{Lasso}) = \sum_{i=1}^{n} (Y_i - b_1 X_{1i} - \cdots - b_k X_{ki})^2 + \lambda_{Lasso} \sum_{j=1}^{k} |b_j|$$

where $\lambda_{Lasso} \geq 0$ is called the *Lasso shrinkage parameter*.

This looks a lot like ridge estimation – but it turns out to have very different properties...

Shrinkage Using the Lasso

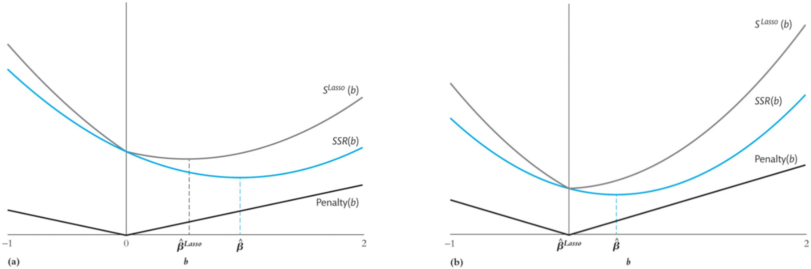
Both ridge and Lasso shrink estimated coefficients to 0. However,

- When $|b_j| > 1$, $|b_j| < b_j^2 \Rightarrow$ ridge penalty $>$ Lasso penalty, and so ridge shrinks (toward 0) more than Lasso.
- When $|b_j| < 1$, $|b_j| > b_j^2 \Rightarrow$ ridge penalty $<$ Lasso penalty, and so Lasso shrinks more than ridge, and in many cases, all the way to 0.

Unlike ridge, Lasso provides a way to select predictors and then estimate their coefficients with a modest amount of shrinkage.

- Lasso sets many of the estimated coefficients exactly 0, thereby dropping corresponding predictors from the model. This property gives the Lasso its name: the *Least Absolute Selection and Shrinkage Operator* (LASSO).
- The predictors selected by Lasso are subject to less shrinkage than with ridge.

Lasso in Pictures



(a) Lasso shrinks large β less than ridge

(b) Lasso shrinks small β all the way to 0

Thus, the Lasso estimator sets some – many – of the β's exactly to 0.

More on Lasso

In some applications, only a few predictors might be useful, with the rest irrelevant.

In a regression model, irrelevant predictors have zero coefficients. A regression model in which the coefficients are nonzero for only a small fraction of the predictors is called *sparse model*.

Lasso can work especially well when in reality many of the predictors are irrelevant.

Lasso produces sparse models, and works well when the population model is in fact sparse.

More on Lasso

Lasso can be used to models with $k > n$ (like ridge).

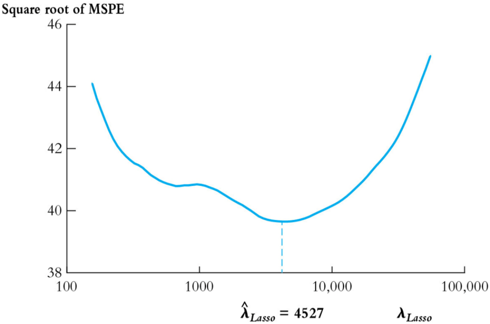
Unlike ridge or OLS, Lasso estimator has no closed-form expression. The Lasso minimization must be done numerically.

λ_{Lasso} can be selected using the same *m*-fold CV algorithm.

Technically, Lasso predictions are not invariant to linear transformations of the regressors, i.e., the estimated model, and selected variables, depends on how the variables are specified.

For example, if model A uses the intercept and a dummy variable for female; and model B uses both female dummy and male dummy but no intercept, then Lasso (and ridge) will in general give different predictions for models A and B, although OLS will give the same predictions.

Predicting School-level Test Scores: Lasso



$\hat{\lambda}_{Lasso} = 4527$ is estimated by minimizing the 10-fold CV MSPE.

Root MSPE's: OLS $=$ 78.2, Ridge $=$ 39.7.

The Lasso estimator retains only 56 of the 817 predictors.

Principal Components

Ridge and Lasso reduce the MSPE by shrinking (biasing) the estimated coefficients to zero – and in the case of Lasso, by eliminating many of the regressors entirely.

Instead, Principal components regression collapses the very many predictors into a much smaller number ($p \ll k$) of linear combinations of the predictors.

These the linear combinations – called the *principal components* (PC) of X – are computed so that they capture as much of the variation in the original X's as possible.

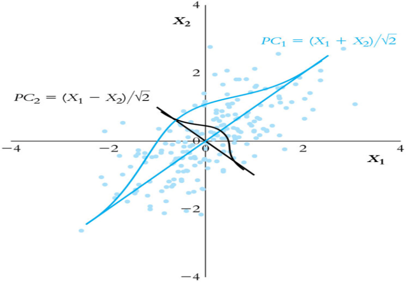
Because the number p of principal components is small, OLS can be used, with the principal components as (new) regressors.

Principal Components in Pictures, $k = 2$

Suppose you have 2 X's, and you want to choose a linear combination of those X's (say, $aX_1 + bX_2$) that captures as much of the variation of the X's as possible. What values of a and b would you use?

The Principal Components solution is to choose a and b to solve, $\max V(aX_1 + bX_2)$ subject to $a^2 + b^2 = 1$.

For 2 X's positively correlated, the choices of a and b are $a = b = \sqrt{2}/2$.



Principal Components, $k > 2$

When $k > 2$, the principal components are the linear combinations of the X's that have the greatest variance and that are uncorrelated with the previous principal components.

So, the jth principal component PC_j, solves,

$$\max V(\sum_{i=1}^{k} a_{ji} X_i) \text{ subject to } \sum_{i=1}^{k} a_{ji}^2 = 1$$

and subject PC_j to being uncorrelated with $PC_1, ..., PC_{j-1}$.

The first p principal components are the linear combinations of X that capture as much of the variation in X as possible.

Principal Components as Data Compression

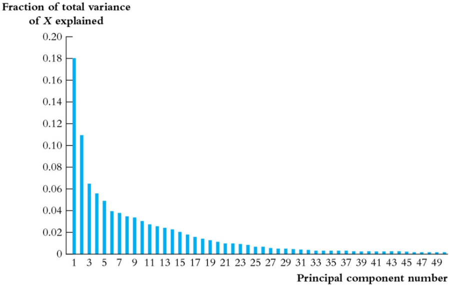
Principal components can be thought of as a data compression tool, so that the compressed data have fewer regressors with as little information loss as possible.

Data compression is used all the time to reduce very large data sets to smaller ones. A familiar example is image compression, where the goal is to retain as many of the features of the image (photograph) as possible, while reducing the file size.

In fact, many data compression algorithms build on or are cousins of principal components analysis.

How Many Principal Components?

One way to choose p is to plot the increase in the average R^2 resulting from adding the pth PC to a regression of X on $PC_1, \ldots, PC_{p-1}$. This plot is known as a *scree plot*.



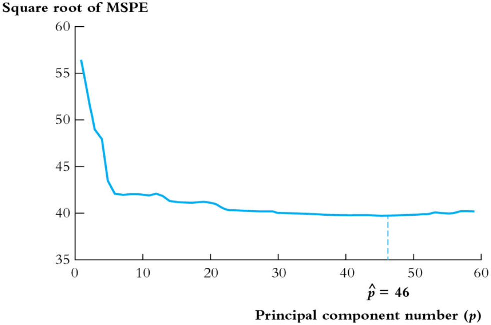
The 1st PC explains 18% of the variation in the 817 X's! The first 10 PC's explain 63% of the variation in the 817 X's! Still, it is rather hard to know where to draw the line...

How Many Principal Components?

The scree plot is informative (you should look at it) but does not provide a simple rule for choosing *p*.

- The number of principal components *p* is like the ridge and Lasso penalty factors λ_{Ridge} and λ_{Lasso} - all are additional parameters needed to implement the procedure.
- Like λ_{Ridge} and λ_{Lasso}, *p* can be estimated by minimizing the *m*-fold CV estimate of the MSPE.
    - For a given value of *p*, the principal components forecast is obtained by regressing *Y* on $PC_1, ..., PC_{p-1}$ using the estimation sample, then using that model to predict in the test sample.

Predicting School-level Test Scores: Principal Components



$\hat{p} = 46$ is estimated by minimizing the 10-fold CV MSPE.

Root MSPE's: OLS $=$ 78.2, PC $=$ 39.7.

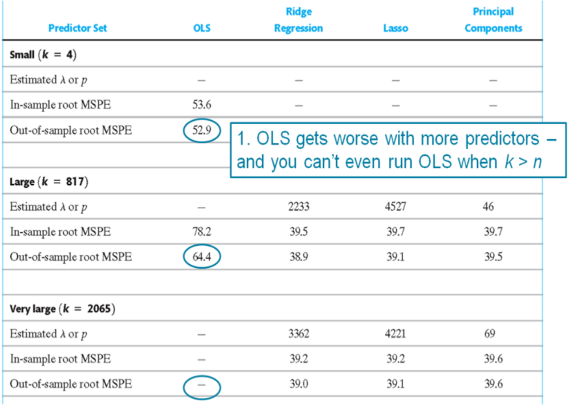
PC collapses the 817 predictors to 46.

Application to School Test Scores

Data set

- Half the observations (1966) used for model estimation including estimation of λ_{Ridge}, λ_{Lasso}, and p.
- The other half is reserved for an OOS test, comparing the various forecasts.
- Three sets of predictors are used:
  - Small ($k = 4$): Student-teacher ratio, median local income, teacher's average years of experience, instructional expenditures per student.
  - Large ($k = 817$): The regressors used up to now.
  - Very large ($k = 2065$): Additional school and demographic variables, squares and cubes, and interactions. For the large data set, $k > n$!

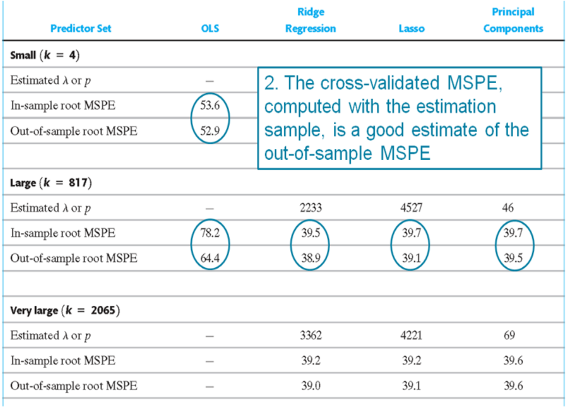
Application to School Test Scores



Predictor Set	OLS	Ridge Regression	Lasso	Principal Components
Small (***k*** **= 4**)				
Estimated λ or p	–	–	–	–
In-sample root MSPE	53.6	–	–	–
Out-of-sample root MSPE	52.9			
Large (***k*** **= 817**)				
Estimated λ or p	–	2233	4527	46
In-sample root MSPE	78.2	39.5	39.7	39.7
Out-of-sample root MSPE	64.4	38.9	39.1	39.5
Very large (***k*** **= 2065**)				
Estimated λ or p	–	3362	4221	69
In-sample root MSPE	–	39.2	39.2	39.6
Out-of-sample root MSPE	–	39.0	39.1	39.6

1. OLS gets worse with more predictors – and you can't even run OLS when $k > n$

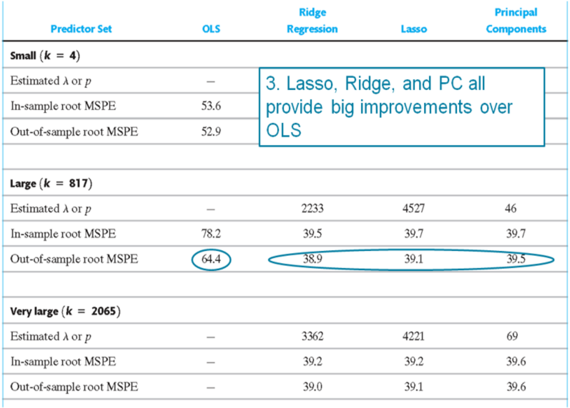
Application to School Test Scores



Predictor Set	OLS	Ridge Regression	Lasso	Principal Components
Small (***k*** = **4**)				
Estimated λ or p	–			
In-sample root MSPE	53.6			
Out-of-sample root MSPE	52.9			
Large (***k*** = **817**)				
Estimated λ or p	–	2233	4527	46
In-sample root MSPE	78.2	39.5	39.7	39.7
Out-of-sample root MSPE	64.4	38.9	39.1	39.5
Very large (***k*** = **2065**)				
Estimated λ or p	–	3362	4221	69
In-sample root MSPE	–	39.2	39.2	39.6
Out-of-sample root MSPE	–	39.0	39.1	39.6

2. The cross-validated MSPE, computed with the estimation sample, is a good estimate of the out-of-sample MSPE

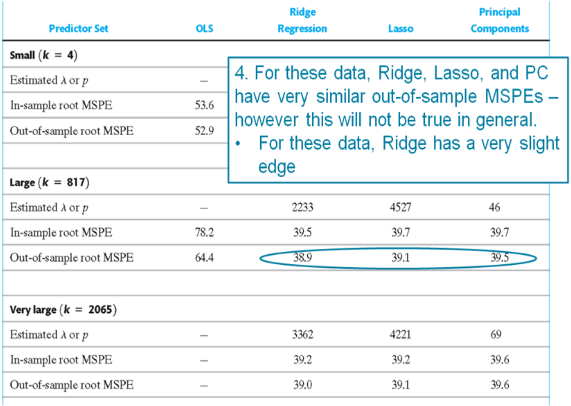
Application to School Test Scores



Predictor Set	OLS	Ridge Regression	Lasso	Principal Components
Small ($k = 4$)				
Estimated λ or p	–			
In-sample root MSPE	53.6			
Out-of-sample root MSPE	52.9			
Large ($k = 817$)				
Estimated λ or p	–	2233	4527	46
In-sample root MSPE	78.2	39.5	39.7	39.7
Out-of-sample root MSPE	64.4	38.9	39.1	39.5
Very large ($k = 2065$)				
Estimated λ or p	–	3362	4221	69
In-sample root MSPE	–	39.2	39.2	39.6
Out-of-sample root MSPE	–	39.0	39.1	39.6

3. Lasso, Ridge, and PC all provide big improvements over OLS

Application to School Test Scores

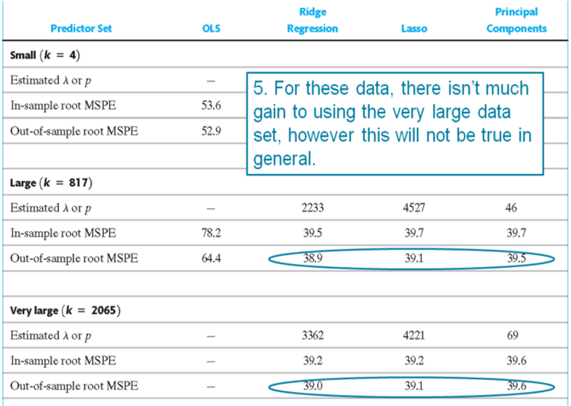


Predictor Set	OLS	Ridge Regression	Lasso	Principal Components
Small (***k*** **= 4**)				
Estimated λ or p	–			
In-sample root MSPE	53.6			
Out-of-sample root MSPE	52.9			
Large (***k*** **= 817**)				
Estimated λ or p	–	2233	4527	46
In-sample root MSPE	78.2	39.5	39.7	39.7
Out-of-sample root MSPE	64.4	38.9	39.1	39.5
Very large (***k*** **= 2065**)				
Estimated λ or p	–	3362	4221	69
In-sample root MSPE	–	39.2	39.2	39.6
Out-of-sample root MSPE	–	39.0	39.1	39.6

4. For these data, Ridge, Lasso, and PC have very similar out-of-sample MSPEs – however this will not be true in general.

- For these data, Ridge has a very slight edge

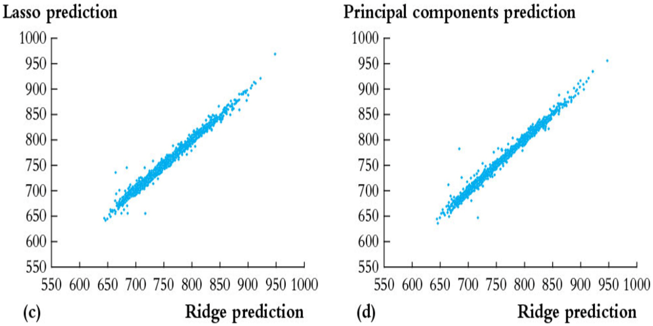
Application to School Test Scores



Predictor Set	OLS	Ridge Regression	Lasso	Principal Components
Small (*k* = 4)				
Estimated λ or p	–			
In-sample root MSPE	53.6			
Out-of-sample root MSPE	52.9			
Large (*k* = 817)				
Estimated λ or p	–	2233	4527	46
In-sample root MSPE	78.2	39.5	39.7	39.7
Out-of-sample root MSPE	64.4	38.9	39.1	39.5
Very large (*k* = 2065)				
Estimated λ or p	–	3362	4221	69
In-sample root MSPE	–	39.2	39.2	39.6
Out-of-sample root MSPE	–	39.0	39.1	39.6

5. For these data, there isn't much gain to using the very large data set, however this will not be true in general.

Application to School Test Scores



For these data, the predictions made by Ridge, Lasso, and principal components are similar to each other (as shown in these scatterplots), however they are quite different from the (worse) OLS predictions.

Summary

- With many predictors, OLS will produce poor OOS predictions.
- By introducing the right type of bias – shrinkage towards 0 – the variance of the prediction can be reduced by enough to offset the bias and result in smaller MSPE.
- Ridge and Lasso reduce the MSPE by shrinking the estimated coefficients to 0 – and in the case of Lasso, by eliminating many of the regressors entirely.
- Principal components collapses X into fewer uncorrelated linear combinations that capture as much of the variation of the X's as possible. Predictions are then made using the OLS regression of Y on the principal components.

Summary

- All three methods require an additional parameter: λ_{Ridge} for Ridge, λ_{Lasso} for Lasso, and p for principal components. This parameter can be estimated by minimizing the m-fold CV estimate of the MSPE.
- The different methods have strengths in different situations, and which works best in a given application is an empirical question.