Chapter 6: Linear Model Selection & Regularization

response numeric
In the regression setting, the standard linear model is commonly used to describe the relationship between a response Y and a set of variables $X_1, \ldots, X_p$.

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \varepsilon$$

typically fit model using least squares.

↓

we will talk about other ways we could approach this fitting problem.

(later will go non-linear).

The linear model has distinct advantages in terms of inference and is often surprisingly competitive for prediction. How can it be improved?

replace least squares with alternative fitting procedures.

We can yield both better *prediction accuracy* and *model interpretability*:

- prediction accuracy: If true relationship is ≈ linear, least squares will have low bias.
  - If $n >> p$ ⇒ also low variance ⇒ perform well on test data!
  - But n not much larger than p ⇒ high variability ⇒ poor performance on test data.
  - If $n < p$ ⇒ no longer have unique solution ⇒ variance = ∞ ⇒ cannot use this at all!

  goal: reduce variance without adding too much bias.

- model interpretability: often many variables used in regression are not in fact associated w/ response.

  By removing them (setting $\hat{\beta}_i = 0$), we can obtain a more interpretable model.

  Note: Least squares will hardly ever result in $\hat{\beta}_i = 0$.

  goal: need variable selection.

Same ideas apply to logistic regression.

1 Subset Selection

We consider methods for selecting subsets of predictors.

1.1 Best Subset Selection

To perform *best subset selection*, we fit a separate least squares regression for each possible combination of the p predictors. $\binom{p}{k}$ models for each # of predictors in model (k).

Algorithm:

1. Let $\mathcal{M}_0$ denote null model — no predictors.
2. For $k = 1, \ldots, p$
   (a) Fit all $\binom{p}{k}$ models that contain k predictors.
   (b) Pick the best of those call it $\mathcal{M}_k$. "Best" defined by $\downarrow$RSS, $\uparrow R^2$
3. Select a single best model from $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$ using CV error, C_p, AIC/BIC, or adjusted R^2 (more later.)

We can't use R^2 for step 3. as $k\uparrow$ $R^2\uparrow$ always.

Why might we not want to do this procedure at all?

We can perform something similar with logistic regression. Fitting 2^p models! $p=10 \Rightarrow 1000$ models.

1.2 Stepwise Selection

For computational reasons, best subset selection cannot be performed for very large p. → impossible for $p \geq 40$

Best subset also suffers when p is large because w/ large search space we can find good models on training data that perform poorly on test data. high variability & overfitting of coeffs can occur.

Stepwise selection is a computationally efficient procedure that considers a much smaller subset of models.

Forward Stepwise Selection: start w/ no predictors and add predictors one at a time until all predictors are in the model. Choose the "best" from these.

1. Let $\mathcal{M}_0$ denote the null model — no predictors.
2. For $k = 0, \ldots, p-1$
   (a) consider all $p-k$ models that augment predictors in $\mathcal{M}_k$ w/ 1 additional predictor.
   (b). Choose the best among $p-k$ and call it $\mathcal{M}_{k+1}$ ($\uparrow R^2$ / $\downarrow$RSS)
3. Select a single best model from $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using CV error, C_p, AIC/BIC, or adjusted R^2.

Now we are fitting $1 + \sum_{k=0}^{p-1} (p-k) = 1 + \frac{p(p+1)}{2}$ models!

Backward Stepwise Selection: Begin w/ full model and take predictors away one at a time until we get to null model.

1. Let $\mathcal{M}_p$ denote the full model – all p predictors
2. $k = p, p-1, \ldots, 1$:
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$ ($k-1$ predictors).
   (b) Choose best among them and call it $\mathcal{M}_{k-1}$ ($\uparrow R^2$, $\downarrow RSS$).
3. Select single best model using CV error, etc.

* Neither forward nor backwards stepwise selection are guaranteed to find the best model containing a subset of the p predictors.

When $p > n$: forward selection can be used (but only up to $n-1$ predictors, not p).

1.3 Choosing the Optimal Model

Best subset, forward, backward select all need to pick "best" model – according to test error.

RSS & R^2 are proxies for training error ⇒ not good estimates of test error. – ① estimate this directly ② adjust training for model size.

② $C_p = \frac{1}{n}(RSS + 2d\hat{\sigma}^2)$

d: # predictors in subset model; $\hat{\sigma}^2$: estimate of variance of ε full model

add penalty to training error (RSS) to adjust for underestimation of test error

as $d\uparrow$, $C_p \uparrow$

② AIC & BIC* For maximum likelihood fits (! linear fits w/ least squares).

$$AIC = \frac{1}{n\hat{\sigma}^2}(RSS + 2d\hat{\sigma}^2)$$

proportional ⇒ same answer (as C_p)

$$BIC = \frac{1}{n\hat{\sigma}^2}(RSS + \log(n)\, d\, \hat{\sigma}^2)$$

[Choose model w/ lowest BIC]

Since for $n > 7$ ⇒ $\log(n) > 2$ ⇒ BIC is heavier penalty for adding variables ⇒ results in smaller models.

② Adjusted R^2 (least squares models)

$R^2 = 1 - \frac{RSS}{TSS}$ always ↑ as $d\uparrow$

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

[choose model w/ highest Adj R^2.

① Validation and Cross-Validation

– Directly estimate test error w/ validation or CV and choose model w/ lowest estimated error.

– Very general (can be used w/ any model) even when it's not clear how many "predictors" are in the model.

Now we have fast computers ⇒ CV is preferred.

2 Shrinkage Methods

The subset selection methods involve using least squares to fit a linear model that contains a subset of the predictors. As an alternative, we can fit a model with all p predictors using a technique that constrains (*regularizes*) the estimates.

↳ shrink towards zero

Shrinking the coefficient estimates can significantly reduce their variance!

Helps us avoid overfitting.

2.1 Ridge Regression

Recall that the least squares fitting procedure estimates $\beta_1, \ldots, \beta_p$ using values that minimize

"residual sum of squares"

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

Ridge Regression is similar to least squares, except that the coefficients are estimated by minimizing $\hat{\beta}^R$

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

note we do not penalize β_0 we want to penalize relationships not the intercept (mean response when $x_{i1} = \ldots = x_{ip} = 0$)

$\lambda \geq 0$ "how much to penalize magnitude of coefs" tuning parameter (determined separately from fitting procedure).

trades off 2 criteria: minimize RSS to fit data well

$\lambda \sum_{j=1}^{p} \beta_j^2$ shrinkage penalty small when β_j close to zero $\Rightarrow$ shrinks estimates towards zero.

The tuning parameter λ serves to control the impact on the regression parameters.

when $\lambda = 0$ penalty has no effect and ridge regression = least squares.

As $\lambda \to \infty$, impact of penalty grows $\hat{\beta}^R \to 0$

Ridge regression will produce a different set of coefficients for each penalty ($\hat{\beta}_\lambda^R$)

Selecting a good λ is critical! How to choose? CV.

The standard least squares coefficient estimates are scale invariant.

Multiply X_j by a constant c leads to a scaling of least squares coef by a factor of $1/c$.

$\Rightarrow$ regardless of how j^{th} predictor is scaled, $X_j \hat{\beta}_j$ will remain the same.

In contrast, the ridge regression coefficients $\hat{\beta}_\lambda^R$ can change substantially when multiplying a given predictor by a constant.

e.g. say we have an income variable in (1) dollars and (2) thousands of dollars.

(1) = 1000 × (2)

due to the sum of squared coef term, this change does not simply result in the coefficient estimate to change by a factor of 1000.

$\Rightarrow X_j \hat{\beta}_{j\lambda}^R$ depends not only on λ but also on the scaling of X_j

(may even depend on scaling of other predictors!).

best practice

Therefore, it is best to apply ridge regression *after standardizing the predictors* so that they are on the same scale:

i.e. standard deviation of one.

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}}$$

estimate of st. dev. of j^{th} predictor.

(1) standardize data

(1.5) tune model to choose λ (w/ CV)

(2) fit ridge regression w/ chosen λ

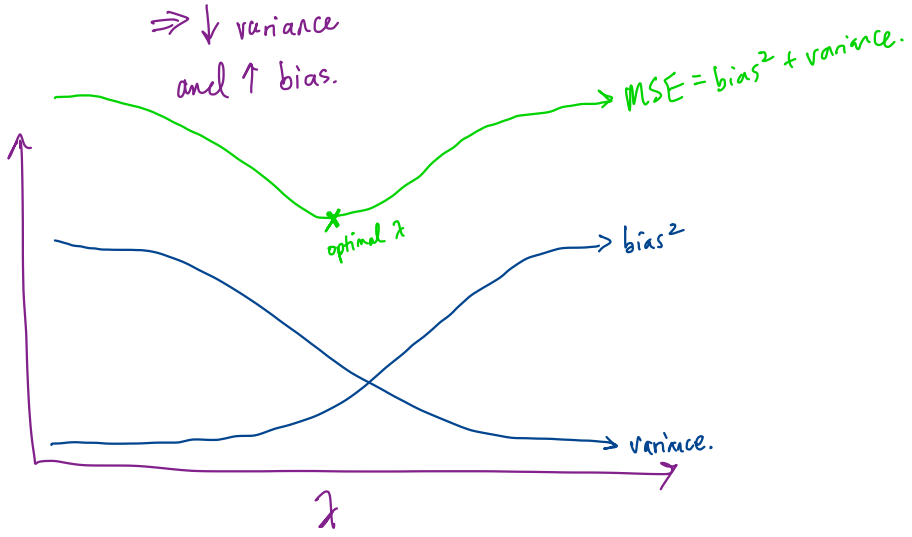
Why does ridge regression work?

Because of the bias-variance trade-off!

As $\lambda \uparrow$: the flexibility of the ridge regression fit $\downarrow$

$\Rightarrow \downarrow$ variance

and $\uparrow$ bias.



In situations where relationship between response and predictors $\approx$ linear.

least squares will have low bias in its estimates

When p almost as large as n $\rightarrow$ least squares has variability!

If $p > n$ least squares doesn't have a solution!

ridge regression can still perform well in these scenarios by trading off a small amount of bias for a decrease in variance.

$\Rightarrow$ ridge regression works best in high variance scenarios.

Also

Cost advantage of ridge regression over subset selection

b/c for a fixed λ, only fitting one model! (very fast model to fit).

Ridge regression improves predictive performance.

Does it also help us with interpretation? NO!

2.2 The Lasso

Ridge regression does have one obvious disadvantage.

Unlike best subset, forward and backward selection ridge regression will include all p variables in final model.

penalty $\lambda \sum \beta_j^2$ will shrink all $\beta_j \to 0$ but $\beta_j \neq 0$ (unless $\lambda = \infty$).

This may not be a problem for prediction accuracy, but it could be a challenge for model interpretation when p is very large.

We will always have all variables in model, whether there is a true relationship w/ Y or not!

"Least absolute shrinkage and selection operator"

The *lasso* is an alternative that overcomes this disadvantage. The lasso coefficients $\hat{\beta}_\lambda^L$ minimize

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p}|\beta_j| = RSS + \lambda \underbrace{\sum_{j=1}^{p}|\beta_j|}_{\text{"}\ell_1 \text{ penalty"}}$$

$= \|\beta_j\|_1$, ℓ_1 norm

$\sum_{j=1}^{p}\beta_j^2$ = "ℓ_2 norm"

As with ridge regression, the lasso shrinks the coefficient estimates towards zero.

ℓ_1 penalty also has the effect of forcing some coefficients to be exactly zero when λ is sufficiently large.

$\Rightarrow$ much like best subset selection, lasso performs variable selection!

As a result, lasso models are generally easier to interpret.

The lasso yields sparse models - models w/ only a subset of variables.

Again, selecting a good λ is critical! (using CV).

Why does the lasso result in estimates that are exactly equal to zero but ridge regression does not? One can show that the lasso and ridge regression coefficient estimates solve the following problems

$$\text{lasso}: \text{minimize} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right)^2 \right\} \text{ subject } \sum_{j=1}^{p} |\beta_j| \le s$$

$$\text{Ridge}: \text{minimize} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}\right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \beta_j^2 \le s$$

constrained optimization problems.

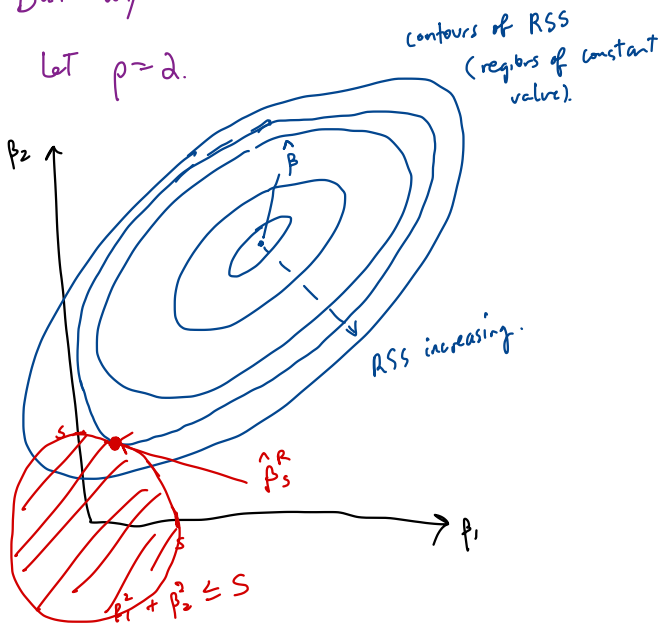
$\Longleftrightarrow$ equivalent to previous formulation

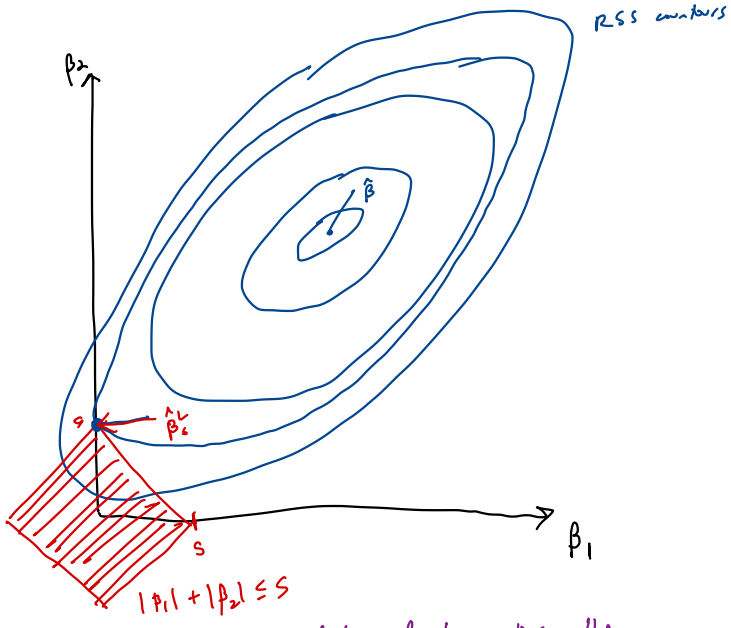
In other words, when we perform the lasso we are trying to find the set of coefficient estimates that lead to the smallest RSS, subject to the contraint that there is a budget s for how large $\sum_{j=1}^{p} |\beta_j|$ can be.

When s is very large, this is not much of a constraint => coef estimates can be large. (similar for ridge regression).

But why does the lasso result in coef estimates exactly equal to zero?

Let $p = 2$.





The solution to ridge regression is the point of intersection between the constrained circle and the contour ellipses.
lasso: constrained diamond

Since ridge has a circular constraint w/ no sharp points the intersection is generally not on the axis. lasso has corners on axes => ellipse often intersect at the axis => one or more of coefficient will equal zero.

If we believe there are predictors that do not have a relationship with Y (we just don't know which ones), lasso will perform better (bias + variance).

If not (everything is important), ridge regression will perform better.

2.3 Tuning

We still need a mechanism by which we can determine which of the models under consideration is "best".

For subset we used C_p, AIC/BIC, adjusted R^2, CV error.

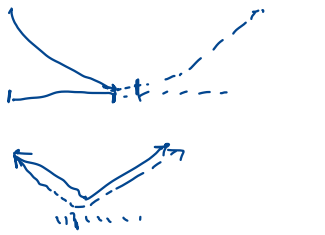
For both the lasso and ridge regression, we need to select λ (or the budget s). equivalently.

λ: penalization parameter

How?

0. Scale predictors to have st. dev. = 1.
1. choose a grid of λ values.
2. Compute CV error (LOOCV or K-fold.) for each λ.
3. Select λ for which CV error is smallest.
4. refit model using all available observations and selected λ.

if I haven't picked big enough grid, pick a bigger grid and start over.



*Note: still important to scale variables $x_1, \ldots, x_p$ for lasso to have st. dev = 1.