Today: · Ridge & Lasso
· Inference after model selection

(No class next week). Last time:

- Ridge $\hat{\beta}^{ridge} = (X^TX + \lambda I)^{-1} X^T Y$
  - assumes no intercepts, so center variables beforehand (as in script)
- Ridge and lasso are not scale-invariant, ie. in $\hat{Y} = X\hat{\beta}$, changing/scaling X also changes $\hat{\beta}$; $\hat{Y}$ is not changed.
  - ↪ so, standardize beforehand.
- Note: R package glmnet scales it by default (automatically)

MODEL SELECTION. Assume all variables are centered, so we don't need/use the intercept

($Y = X^T\beta + \varepsilon$, $\beta \in \mathbb{R}^p$).

- Best subset for model selection of size $\leq s$ ($s \leq p$)

(*) $\hat{\beta}_s^{subset} = \underset{\beta: \|\beta\|_0 \leq s}{\operatorname{argmin}} RSS(\beta)$, $\|\beta\|_0 = \sum_{j=1}^{p} 1_{\{\beta_j \neq 0\}}$

  - Equivalently, there is λ s.t. $\hat{\beta}_s^{subset} = \hat{\beta}_\lambda^{subset} = \underset{\beta}{\operatorname{argmin}} RSS(\beta) + \lambda \|\beta\|_0$ (Lagrange multiplier)
    akin to penalty term for AIC, BIC
  - While the function is convex, the constraint $(\beta: \|\beta\|_0 \leq s$ is non-convex! ⟹ computation is infeasible.
  - Best subset looks like ridge and lasso, but with a different norm.

- Similarly, we can reformulate ridge and lasso as constraint optimization problems:

(*) $\hat{\beta}_s^{ridge} = \underset{\beta: \|\beta\|_2^2 \leq s}{\operatorname{argmin}} RSS(\beta) \iff \hat{\beta}_\lambda^{ridge} = \underset{\beta}{\operatorname{argmin}} RSS(\beta) + \lambda \|\beta\|_2^2$

(*) $\hat{\beta}_s^{lasso} = \underset{\beta: \|\beta\|_1 \leq s}{\operatorname{argmin}} RSS(\beta) \iff \hat{\beta}_\lambda^{lasso} = \underset{\beta}{\operatorname{argmin}} RSS(\beta) + \lambda \|\beta\|_1$

$\|\beta\|_1 = \sum_{j=1}^{p} |\beta_j|$

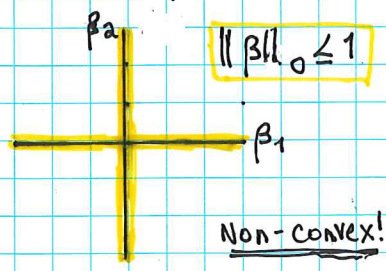
- We can also take the l_q-norm:

(*) $\|\beta\|_q^q = \sum_{j=1}^{p} \beta_j^q$. These are norms if $q \geq 1$.

- Lasso is in a sense the best convex relaxation of the all-subset problem.

 Computational Statistics Lecture Thu 3 May 2018

- All-subset problem:

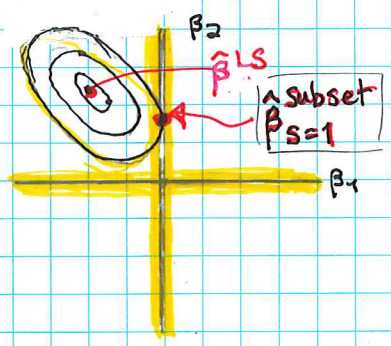


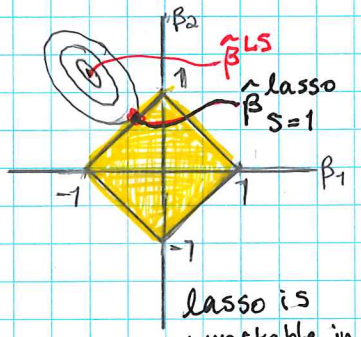
Non-convex!

- large $s \rightarrow$ small λ. (includes LS solution, so no penalty)
- for lasso, the constraint value can be some value other than 1.

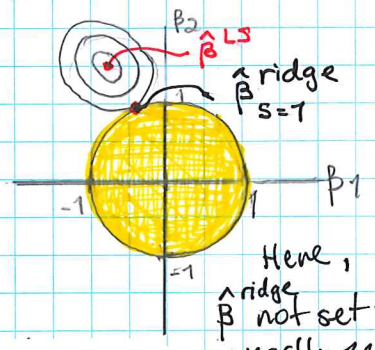
Given a constraint, how does RSS β behave? (contours)

- Subset $\|\beta\|_0 \leq 1$
- lasso $\|\beta\|_1 \leq 1$
- ridge $\|\beta\|_2^2 \leq 1$





lasso is v. unstable in variable selection



Here, $\hat{\beta}^{ridge}$ not set to exactly zero (but close to axis, ie $\beta_1 = 0$).

- Often, the solution is in a "corner" ⇒ variable selection.

- (Ref. slides) For ridge, as $\lambda \nearrow$, some coefficients can $\nearrow$ if there are interactions among/between variables (predictors)
  - instead of λ, we can also look at regression coefs over R^2 axis ($\|\hat{\beta}_\lambda^R\|_2 / \|\hat{\beta}\|_2$) since λ can have different values (lasso v. ridge)

- If we ...
  - are doing variable selection or think that the true is sparse → use lasso
  - are doing prediction or don't know if true is sparse → use ridge.

(- lasso and ridge both do shrinkage, but in different ways)

- (Ref R code 10) given 80 samples, 26 = 25 predictors (X) and outcome (y col)
- packages: step; leaps - default is backward regression.

(line 9) Output table

X	AIC
X_1	4
X_6	5
⋮	⋮

- (removing that X var → AIC score)
- Variable is significant if score is high w/out it.

(Be aware of hidden multiple testing! ↑ p val test is for one model test for a)

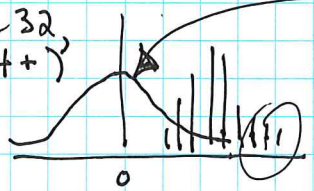
↳ can correct with ex. Bonferroni

→ exploring data, formulating hypotheses vs.
- confirming hypotheses, testing.
- non-random selection of data
- model selection

- Rcode10, cont'd

(line 9) prob is from standard Gaussian distribution

(line 32, 54+)



We should look @ the 95th < part of data not in the standard Gaussian.

INFERENCE AFTER MODEL SELECTION.

- Sample Splitting (starting @ line 62)

(line 78). Shows how model selection (of data), multiple testing, ... can erroneously yield small p values.

- is a solution to this - gives us "honest" p values.
  - issues - lose power
  - Don't trust p values from data that was non-randomly selected unless you also do the (line 78)

- idea: make a random split, learn one half, test on the other. fit model, test it and get p-values, and correct them (ex. by Bonferroni)

ex. p values

Split #	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
1	$3p_1^{(1)}$	1	1	$3p_4^{(1)}$	$3p_5^{(1)}$	1	1	1	1	1
2	$4p_1^{(2)}$	1	1		$4p_6^{(2)}$	$4p_6^{(2)}$	$4p_7^{(2)}$			
3	1									
4	1									
5	$2p_1^{(5)}$									
6	1									
7	1									

Can take the q^{th} quantile and divide by q to correct the p value for X_1's. (ex. if $q = 0.5$, then take the median and multiply by 2)

- where $p_j^{(i)}$ is the unadjusted p value corresponding to X_j in split i

- Sample split is more brute force, but works. ∃ other methods that can use all the data to fit the model.

//

COMPUTATIONAL STATISTICS Lecture (+Section)

Fri 4 May 2018

(Ref to ISLR Ch 7).

(No Lecture/section next week, ie resume Thu May 17).

Beyond linearity

- Ridge & lasso are modifications of linear regression. Idea:
  - reduce model complexity to decrease model variance at the cost of hopefully a small bias.
- Now, we consider the reverse. We increase model complexity to decrease model bias at the cost of hopefully a small increase in variance

- Setup $y_i = m(x_i) + \varepsilon_i$

Overview:
- polynomial regression
- step functions
- regression splines
- smoothing splines
- local regression

} univariate: 1 predictor

- generalized additive models (GAMs)
  - ↳ way to add dimensions without curse of dimensionality
  - ↳ GAMs can have non-linear predictor terms, but they act on it in a linear (additive) fashion so that we may avoid curse of dimensionality

- POLYNOMIAL REGRESSION

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_d x_i^d + \varepsilon_i$$

$$\equiv \beta_0 + \sum_{j=1}^{d} \beta_j x_i^j + \varepsilon_i \qquad \equiv \sum_{j=0}^{d} \beta_j x_i^j + \varepsilon_i$$

Pros ⊕ easy to fit via LS Cons ⊖ unstable at boundaries
⊕ inference goes through

(Ref R Code 11)

(line 26) "I(~)" is indicator? (of variable in arg)
(line ?) "coefs(summary(fit))"
need to construct columns of matrix s.t. they are orthogonal to each other
(line 35) "cbind(preds$fit + 2, ...)" ie for 2 standard deviations

- Recall the anova approach, which (can be) used for nested models (ie. model of degree 3 is a submodel of model of degree 4, which is...

(line 53, 54) anova approach: compare models of degree d and d+1 to
- see whether the (d+1)st variable contributes/has a significant effect
- How does p value change? Which has the smaller p val?

(line 59, 60) If we use (regular lm) for non-orthogonal matrices:
- pvalues are not fair, ie. p val for degree 1, $p_1 < p_2$ but $p_3 < p_1$. Which (degree 1 or 3) is better??
- Best to use ANOVA.

(POLYNOMIAL REGRESSION, cont'd)

ANOVA : $H_0 : y_i = \alpha + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i$

(ex. for degree 3) $H_a : y_i = \alpha + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \varepsilon_i$

LS output : 3rd p-value tests. → ie for degree k.

$H_0 : y_i = \alpha + \sum_{j=1}^{k-1} \beta_j x_i^j + \varepsilon_i$

$H_a : y_i = \alpha + \sum_{j=1}^{k} \beta_j x_i^j + \varepsilon_i$

$H_0 : y_i = \alpha + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \beta_4 b_4(x_i) + \beta_5 b_5(x_i) + \varepsilon_i$

$H_a : y_i = \underbrace{\alpha + \beta_1 b_1(x_i) + \beta_2 b_2(x_i)}_{1,2} + \beta_3 b_3(x_i) \underbrace{+ \beta_4 b_4(x_i) + \beta_5 b_5(x_i)}_{4,5} + \varepsilon_i$

These b_j's are orthogonal → ie for ℓ^{th} p val test,

$H_0 : y_i^{H_0} = \alpha + \sum_{j=1,\dots,k;\ j \neq \ell} \beta_j b_j(x_i) + \varepsilon_i$

$H_a : y_i^{H_a} = \alpha + \sum_{j=1,\dots,k} \beta_j b_j(x_i) + \varepsilon_i$

$\equiv y_i^{H_0} + \beta_\ell b_\ell(x_i)$

- **Step functions**

(- similar idea : (create new variables))

- Take cut points $c_1, \dots, c_k$ and create $k+1$ new variables s.t.

$C_0(x) = 1\{x \le c_1\}$
$C_1(x) = 1\{c_1 < x \le c_2\}$
$C_2(x) = 1\{c_2 < x \le c_3\}$
$\vdots$
$C_k(x) = 1\{c_k < x\}$.

more generally,

$C_0(x) = 1\{x \le c_1\}$
$C_j(x) = 1\{c_j < x < c_{j+1}\}$
$C_k(x) = 1\{c_k < x\}$
where $j = 1, \dots, k-1$

($C_0, C_1, \dots, C_k$ are dummy variables for the partition)

So, $y_i = \beta_0 + \sum_{j=1}^{k} \beta_j C_j(x_i) + \varepsilon_i$.

Note that C_0 is omitted because of (we have) the intercept β_0.

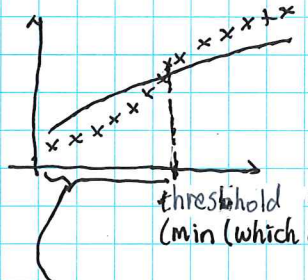
$\beta_0 = \mathbb{E}[Y \mid X \le c_1]$

$\beta_0 + \beta_j = \mathbb{E}[Y \mid c_j < X \le c_{j+1}]$

* β_j is the increase in the response for $X \in (c_j, c_{j+1}]$ compared to $x \le c_1$.

Section (Multiple testing, Model selection) - see posted notes

- Multiple testing (assume distributed N)
- Bonferroni correction is pretty pessimistic ⇝ lose power.
- Ref. FDR ctrl: $(p_i)_{i=1}^{V}$



ISLR: Ch6.1, Ch6.2 are v. helpful.

Script