Lecture 14: PCR

Instead of regressing Y onto $X_{N \times p}$

we can regress Y onto $Z_{N \times q}$, $q \ll p$

Steps for PCR:

0. mean center X

$$X_c = \begin{bmatrix} | & | & \\ X_1 - \text{mean}(X_1) & X_2 - \text{mean}(X_2) & \cdots \\ | & | & \end{bmatrix}$$

1. do PCA: $X_c = UDV^T$

$$Z = X_c V_q$$

first q cols of V

2. regress Y onto Z

typically want to include intercept

so let

$$D = \begin{bmatrix} | & | \\ 1 & Z \\ | & | \end{bmatrix}$$

then $\hat{\beta}^{(PCR)} = (D^T D)^{-1} D^T Y \in \mathbb{R}^{q+1}$.

What about predicting on new data?

Let X^{test} is $M \times P$

for training $\hat{Y} = D\hat{\beta}^{(PCR)}$

Need to form D_{test} by applying same steps to X^{test}.

(0) center the data

means of training data

$$X_c^{test} = \begin{bmatrix} | & | & \\ X_1^{test} - \text{mean}(X_1) & X_2^{test} - \text{mean}(X_2) & \cdots \\ | & | & \end{bmatrix}$$

(1) apply PCA

from training

$$Z^{test} = X_c^{test} V_q$$

(2) $$D_{test} = \begin{bmatrix} | & | \\ 1 & Z^{test} \\ | & | \end{bmatrix}$$

then $$\hat{Y}^{test} = D_{test}\hat{\beta}^{(PCR)}$$

Compare w/ ridge regression

$$\hat{\beta}^{(ridge)} = (X^T X + \lambda I)^{-1} X^T Y$$

saw that

$$\hat{Y} = X\hat{\beta}^{ridge} = \cdots = \sum_{j=1}^{p} \left(\frac{\sigma_j^2}{\sigma_j^2 + \lambda}\right) u_j u_j^T Y$$

$\propto$ PCs

shrink contribution of small σ_i more than large σ_i

$$= \sum_{j=1}^{p} \Delta_j u_j u_j^T Y$$

$$\Delta_j = \frac{\sigma_j^2}{\sigma_j^2 + \lambda}$$

Same analysis for PCR:

$$\hat{Y} = Z\hat{\beta}^{(PCR)} = Z(Z^T Z)^{-1} Z^T Y$$

$Z = XV_q = U_q D_q$

$$= U_q D_q ((U_q D_q)^T U_q D_q)^{-1} (U_q D_q)^T Y$$

$$= U_q D_q (D_q \underbrace{U_q^T U_q}_{I} D_q)^{-1} D_q^T U_q^T Y$$

$$= U_q \underbrace{D_q (D_q^T D_q)^{-1} D_q}_{I} U_q^T Y$$

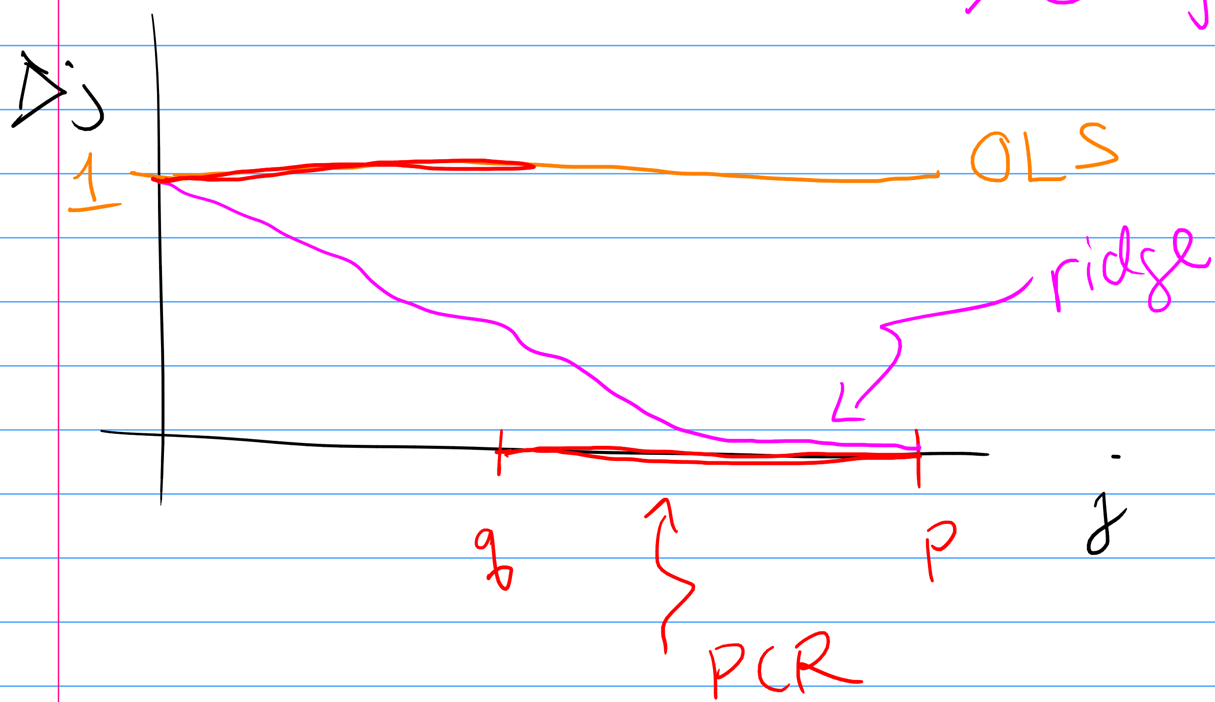
$$= U_q U_q^T Y$$

$$= \sum_{j=1}^{q} u_j u_j^T Y$$

j^{th} col of U

$$= \sum_{j=1}^{p} \Delta_j u_j u_j^T Y$$

$$\Delta_j = \begin{cases} 1 & j \leq q \\ 0 & j > q \end{cases}$$



Consider X to be full rank

then as $\lambda \to 0$ we have

$$\hat{\beta}^{ridge} \longrightarrow \hat{\beta}^{OLS}$$

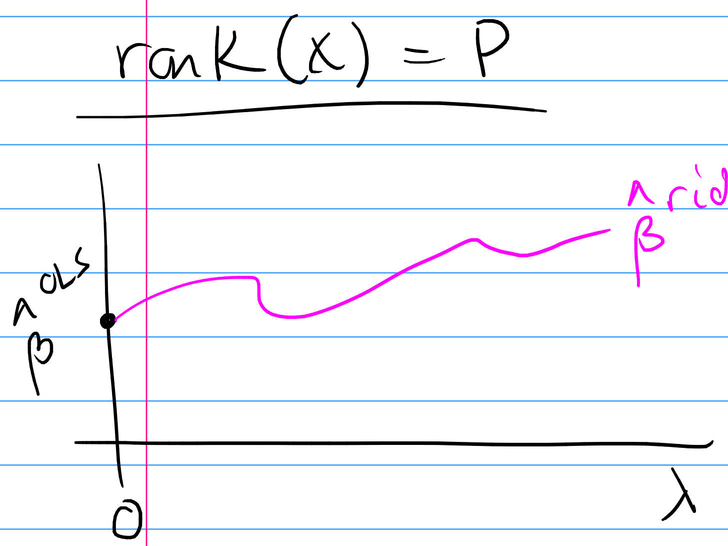
Similarly, as $q \to P$

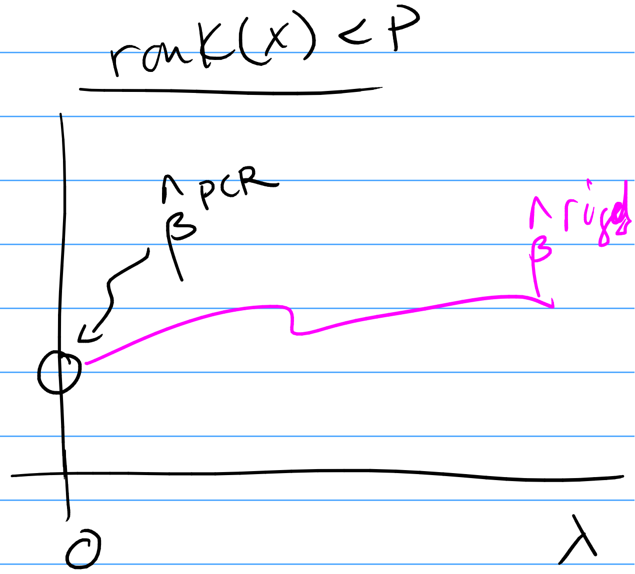
$$\hat{\beta}^{PCR} \longrightarrow \hat{\beta}^{OLS}$$

If $\text{rank}(X) < P$ then $\hat{\beta}^{OLS}$ doesn't exist

equivalently $\hat{\beta}^{ridge}$ w/ $\lambda = 0$ doesn't exist.

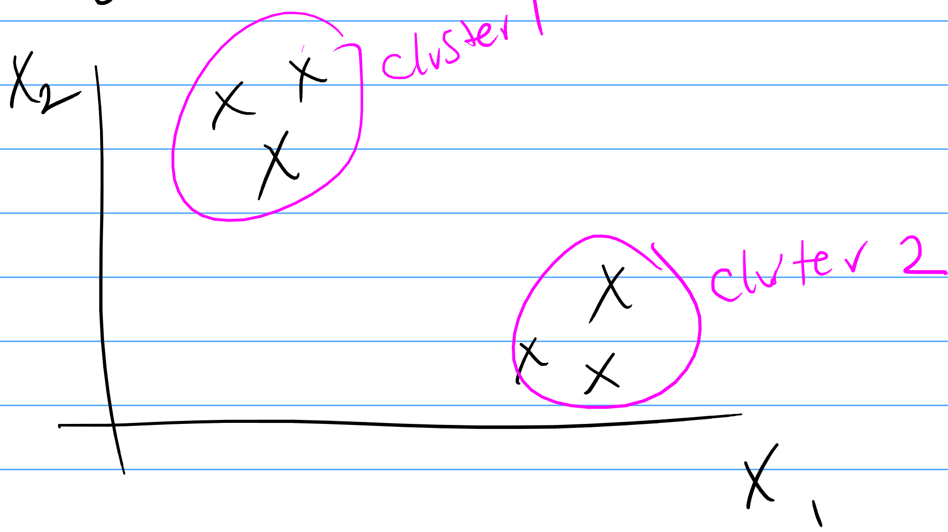
However, $\lim_{\lambda \to 0} \hat{\beta}^{ridge} = \hat{\beta}^{PCR}$ w/ $q = \text{rank}(X)$





Back to unsupervised methods

Another class of unsupervised methods is <u>clustering methods</u>



To find clusters I need some way of defining how pts are similar/dissimilar

<u>Need</u>: Some dissimilarity measure

If I have N observations then I need to define some matrix

$$D \quad (N \times N)$$

where $D_{ii'}$ = dissimilarity measure between obs. i and i'.

For many methods don't explicitly need X only need D

Need some properties to be true of D!

(1) $D_{ii} = 0$

(2) $D_{ii'} \geq 0$

(3) $D = D^T$ (symmetric)

K-means clustering

Assume each data point belongs to one of K clusters (or groups)

$$G_1, G_2, \ldots, G_K$$

want! assign each point i to some cluster G_k

how! want to make assignments so that I minimize some measure of not being well clustered ("loss")

classic loss for clustering is

$$W = \text{total w/in cluster dissimilarity} = \sum_{k=1}^{K} \sum_{i,i' \in G_k} D_{ii'}$$

- W should be large if clustering is bad
- W is small if clustering is good

$$T = \text{total dissim} = \sum_{i,i'} D_{ii'}$$

$$B = \text{total between cluster dissim} = \sum_{k,k'} \sum_{i \in G_k} \sum_{i' \in G_{k'}} D_{ii'}$$

One can show that

$$T = W + B$$

So to find $G_1, \ldots, G_K$ we should

either (1) minimize W

or (2) maximize B

Ideally: try all possible cluster assignments

practically: not comp. tractible for reasonably large N or K