Linear Regression

Cp, AIC, BIC, and Adjusted *R*2

Prof. Asim Tewari
IIT Bombay

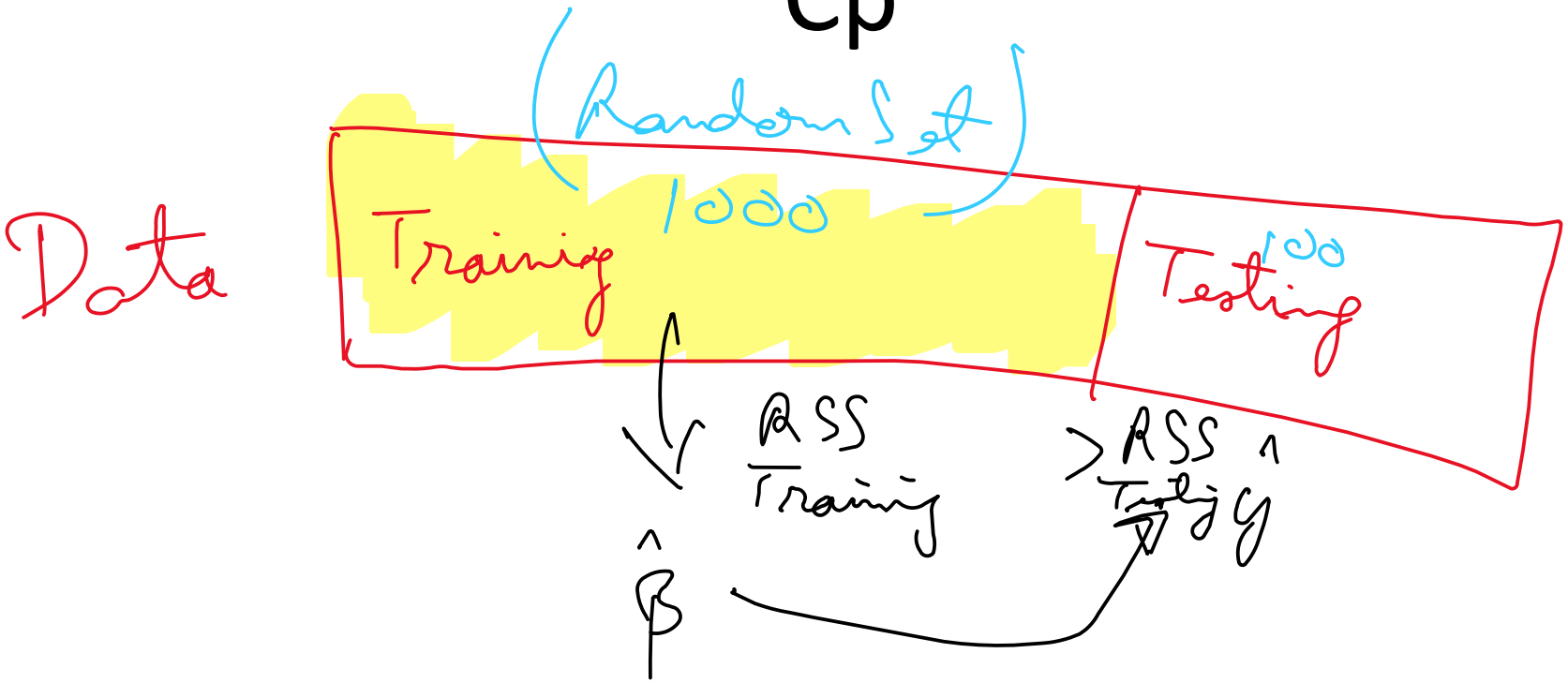
Cp

- For a fitted least squares model containing *d* predictors, the *Cp* estimate of test MSE is computed using the equation

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

- where ˆσ2 is an estimate of the variance of the error associated with each response measurement

Cp


Data
Random Set
1000
Training
100
Testing
RSS
Training
RSS
Testing
β̂

AIC

- The AIC criterion is defined for a large class of models fit by maximum likelihood. In the case of the model

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon$$

with Gaussian errors, maximum likelihood and least squares are the same thing.

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

BIC

- BIC is derived from a Bayesian point of view, but ends up looking similar to *Cp* (and AIC) as well. For the least squares model with *d* predictors, the BIC is, up to irrelevant constants, given by

$\log(n) > 2 \Rightarrow n > 7$

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

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

Adjusted R^2

$$R^2 = 1 - \frac{RSS}{TSS} \quad ; \quad TSS = \Sigma (y_i - \bar{y})^2$$

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

Choosing the Optimal Model

For a fitted least squares model containing d predictors

- *Cp*

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

- *Akaike information criterion* (AIC)

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

- *Bayesian information* (BIC)

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

- *Adjusted R2*

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

where ˆ σ 2 is an estimate of the variance of the error associated with each response measurement

Regression

$$y = f(x) + \epsilon$$

↑

Find 'f'

Data

$$D = \{(x_i, y_i)\}_{i=1}^{n}$$

$\hat{y} = h(x)$

↳ Parametric model/Function $h_\beta(x) = \hat{f}(x, \beta)$

E.g. $\hat{f}(x, \beta) = \beta_0 + \beta_1 x^{(1)} + \beta_2 x^{(2)}$

ML is to use data to find β.

Minimize Cost Function $L(\beta)$

$$\hat{\beta} = \arg\min_{\beta} L(\beta)$$

Null Hypothesis

⇒ $x_1, x_2 \ldots$ are true features or not?

Is β_i 0?

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon$$

Which of the X_i's are actually important?

Obj: Choose the most imp. 'd' variables from the given 'p' variables.

1.) Subset Selection:

2.) Shrinkage Methods

3.) Dimension Reduction

Subset Selection

$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$

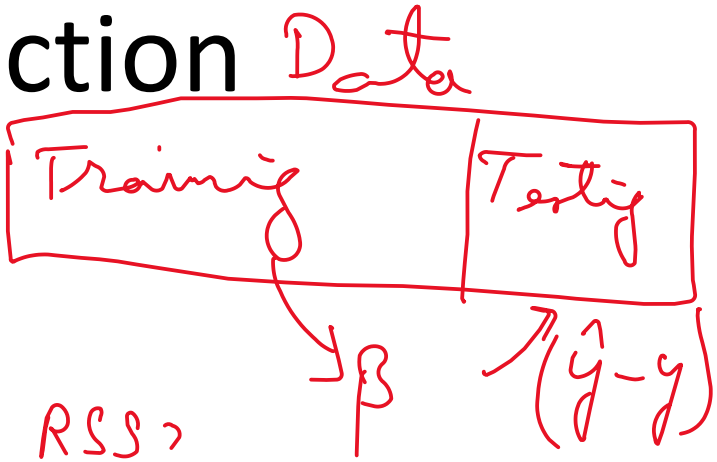
$h_\beta(x, \beta)$

$x_1, x_2 \ldots\ldots x_p$

- 0 variable: 1 way $(y = \beta_0)$
- 1 variable: p ways $(y = \beta_0 + \beta_1 x)$
- 2 Variables: $\frac{p(p-1)}{2} = {}^pC_2$
- d variables: ${}^pC_d = \frac{\lfloor p}{\lfloor p-d \; \lfloor d}$
- p variable: 1 way

$$1 + {}^pC_1 + {}^pC_2 + \cdots + {}^pC_d + \cdots + \underbrace{{}^pC_p}_{1} = (1+1)^p = 2^p$$

Subset Selection



Which one is better?

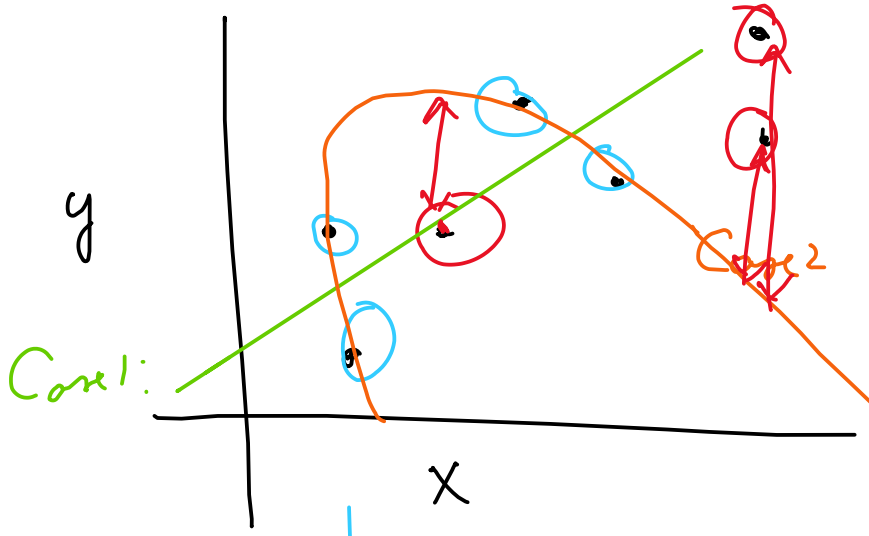
1. $\hat{y} = \beta_0 + \beta_1 X_1$
2. $\hat{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2$

RSS? (Testing? Training?)

RSS of 1. $\geq$ RSS of 2.

Better $\Rightarrow$ Lower MSE of Testing

Subset Selection



Train : 4

Test : 3

Case 1: $\hat{y} = \beta_0 + \beta_1 x$

Case 2: $\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$

$MSE = RSS/n$

RSS of case 1 $\gg$ RSS of case 2

MSE Training

MSE Testing

RSS of Training of Case 1 $<$ RSS of testing of Case 2

$(MSE)_{Test}$ Case 1 $<$ $(MSE)_{Test}$ Case 2

Subset Selection

- We cannot try all the 2^p models that contain subsets of p variables.
- Thus we have strategies for choosing the subset.
- Various statistics can be used to judge the quality of a model. These include
  - Mallow's C p , Akaike information criterion (AIC),
  - Bayesian information criterion (BIC), and
  - Adjusted R^2

Subset Selection

- We cannot try all the 2^p models that contain subsets of p variables.
- Thus we have strategies for choosing the subset.
- Various statistics can be used to judge the quality of a model. These include
  - Mallow's C p , Akaike information criterion (AIC),
  - Bayesian information criterion (BIC), and
  - Adjusted R^2

Strategies for choosing the subset

1. Best-Subset Selection:

   Best subset regression finds for each k $\epsilon\{0, 1, 2, \ldots, p\}$ the subset of size k that gives smallest MSE for testing

2. Forward- and Backward-Stepwise Selection:

   its selection starts with the intercept, and then sequentially adds into the model the predictor that most improves the fit.

Subset Selection

- *Best Subset Selection*

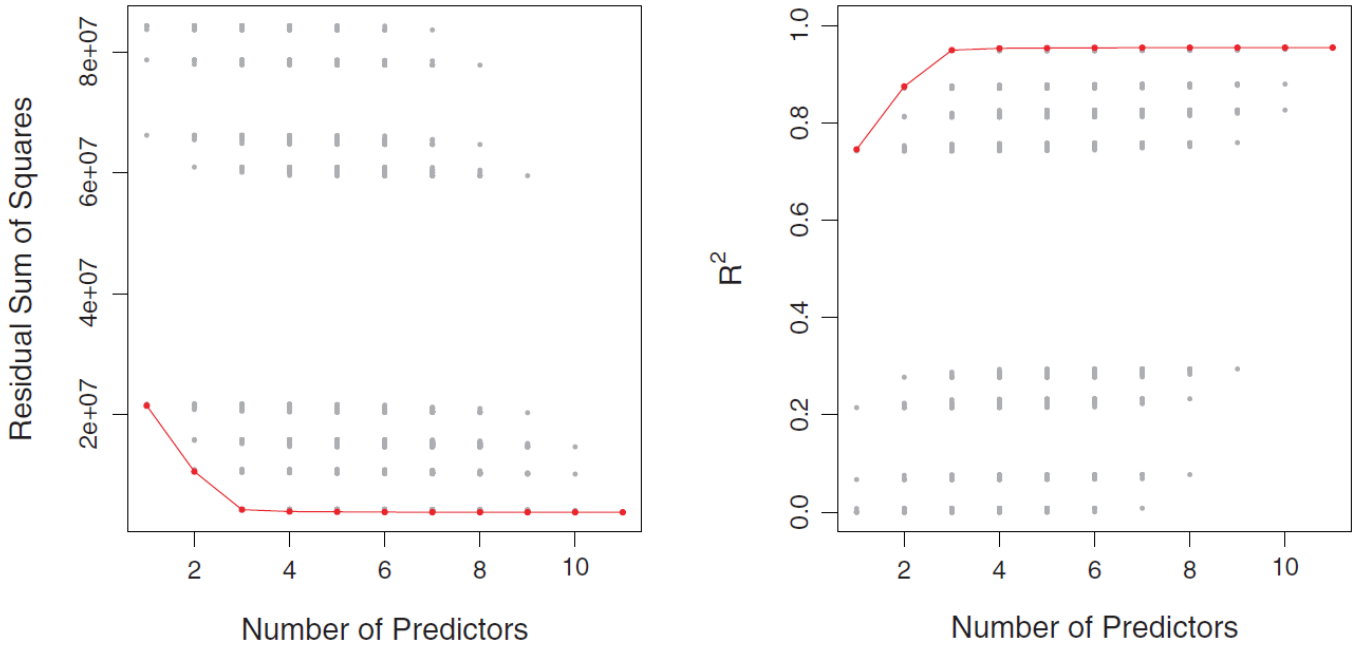
Algorithm

---

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$:
   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Subset Selection

- *Best Subset Selection*



For each possible model containing a subset of the ten predictors in the Credit *data set, the RSS and R2 are displayed. The red frontier tracks the* best *model for a given number of predictors, according to RSS and R2. Though the data set contains only ten predictors, the x-axis ranges from* 1 *to* 11*, since one of the variables is categorical and takes on three values, leading to the creation of two dummy variables.*

Strategies for choosing the subset

1. Forward selection

- We begin with the null model—a model that contains an intercept but no predictors.
- We then fit p simple linear regressions and add to the null model the variable that results in the lowest RSS.
- We then add to that model the variable that results in the lowest RSS for the new two-variable model.
- This approach is continued until some stopping rule is satisfied.

Subset Selection

- *Stepwise Selection*
  - Forward Stepwise Selection

---

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   (a) Consider all $p-k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   (b) Choose the *best* among these $p-k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Strategies for choosing the subset

2. Backward selection

- We start with all variables in the model, and remove the variable with the largest p-value—that is, the variable
- that is the least statistically significant.
- The new (p − 1)-variable model is fit, and the variable with the largest p-value is removed.
- This procedure continues until a stopping rule is reached.
- For instance, we may stop when all remaining variables have a p-value below some threshold.

Subset Selection

- *Stepwise Selection*
  - Backward Stepwise Selection

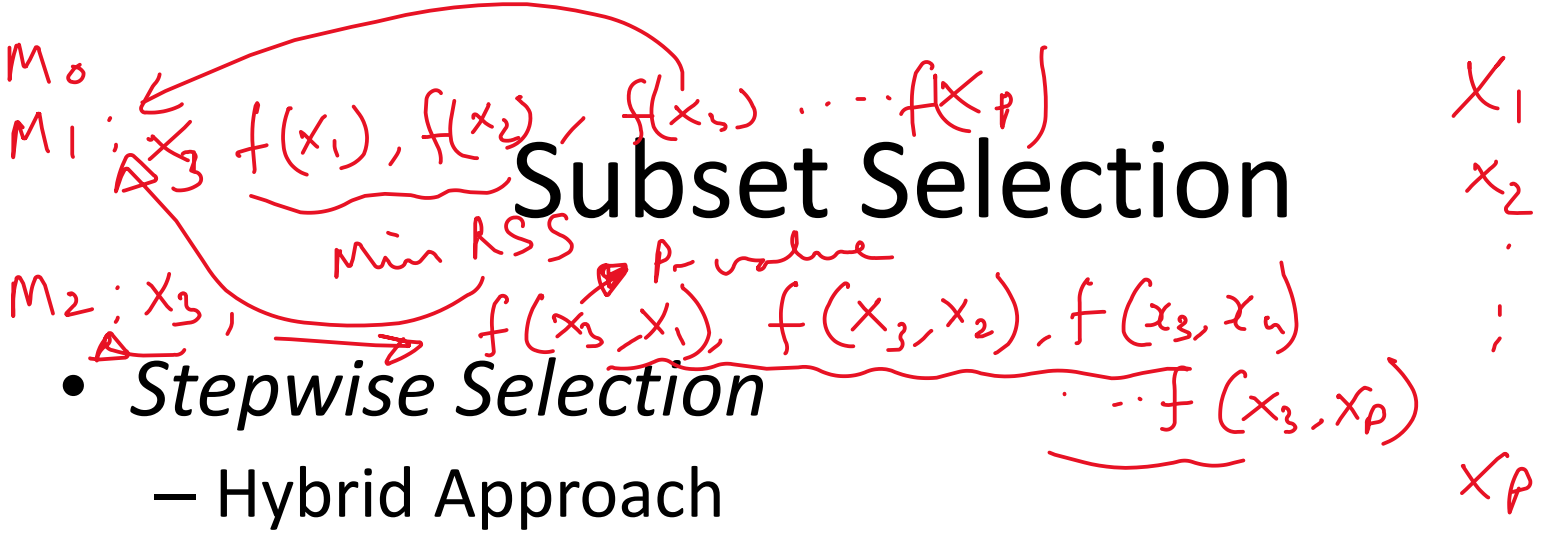
---

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   (b) Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Strategies for choosing the subset

3. Mixed selection

- This is a combination of forward and backward selection.
- We start with no variables in the model, and as with forward selection, we add the variable that provides the best fit.
- We continue to add variables one-by-one.
- It is possible sometimes that the p-values for variables can become larger as new predictors are added to the model.
- Hence, if at any point the p-value for one of the variables in the model rises above a certain threshold, then we remove that variable from the model.
- We continue to perform these forward and backward steps until all variables in the model have a sufficiently low p-value, and all variables outside the model would have a large p-value if added to the model.



Subset Selection

- *Stepwise Selection*
  - Hybrid Approach
    - Variables are added to the model sequentially, in analogy to forward selection.
    - However, after adding each new variable, the method may also remove any variables that no longer provide an improvement in the model fit.
    - Such an approach attempts to more closely mimic best subset selection while retaining the computational advantages of forward and backward stepwise selection.

Linear Model Selection and Regularization

- ***Shrinkage***. This approach involves fitting a model involving all *p* predictors. However, the estimated coefficients are shrunken towards zero relative to the least squares estimates. This shrinkage (also known as *regularization*) has the effect of reducing variance. Depending on what type of shrinkage is performed, some of the coefficients may be estimated to be exactly zero. Hence, shrinkage methods can also perform variable selection.

Find βs by Min RSS

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

Shrinkage

- By retaining a subset of the predictors and discarding the rest, subset selection produces a model that is interpretable and has possibly lower prediction error than the full model.
- However, because it is a discrete process variables are either retained or discarded—it often exhibits high variance, and so doesn't reduce the prediction error of the full model.
- Shrinkage methods are more continuous, and don't suffer as much from high variability.

Shrinkage Methods

1. Ridge Regression

- Ridge regression shrinks the regression coefficients by imposing a penalty on their size. The ridge coefficients minimize a penalized residual sum of squares,

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}.$$

- Here $\boldsymbol{\lambda} \geq 0$ is a complexity parameter that controls the amount of shrinkage. It's directly proportional to shrinkage .

Ridge Regression

$$RSS(\beta)_{Ridge} = \|X\beta - Y\|_2^2 + \lambda \|\beta\|_2^2$$

$$RSS(\beta) = \|X\beta - Y\|_2^2$$

↑ min

constrain $\|\beta\|_2^2 < t$, t ↑ +ve

Ridge Regression

$$RSS(\beta)_{Ridge} = \|X\beta - Y\|_2^2 + \lambda \|\beta\|_2^2$$

$$RSS = \|X\beta - Y\|_2^2$$

↑ min

1000 kg

$10^3 \times 10^3$ g

Constain

β_0

$$\sum \beta_i^2 < t$$

$\sum \beta_i^2 =$

↑ +ve user defined

Ridge Regression

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \cdots + \hat{\beta}_p x_p$$

$$\sum \beta_i^2 < t$$

$$RSS = \sum_{i=1}^{n} (\hat{y} - y)^2 = \sum \left(y - \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots \right)^2$$

Ridge Regression

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \cdots + \hat{\beta}_p x_p$$

$$\sum \beta_i^2 < t$$

$$RSS = \sum_{i=1}^{n} (\hat{y} - y)^2 = \sum \left(y - \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots \right)^2$$

Ridge Regression

$$RSS(\beta)_{Ridge} = \|X\beta - Y\|_2^2 + \lambda \|\beta\|_2^2$$

min

Tends to 0 as $\lambda \uparrow$

$$\equiv \min RSS(\beta) = \|X\beta - Y\|_2^2$$

Constraint $\|\beta\|_2^2 < t$

- Has no β_0
- X are normalized

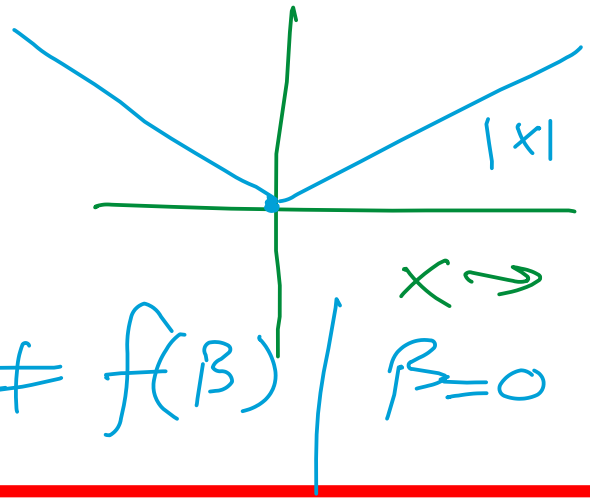
Lasso Regression

$$RSS(\beta)_{Ridge} = \|X\beta - Y\|_2^2 + \lambda \|\beta\|$$

$$\min RSS(\beta) = \|X\beta - Y\|_2^2$$

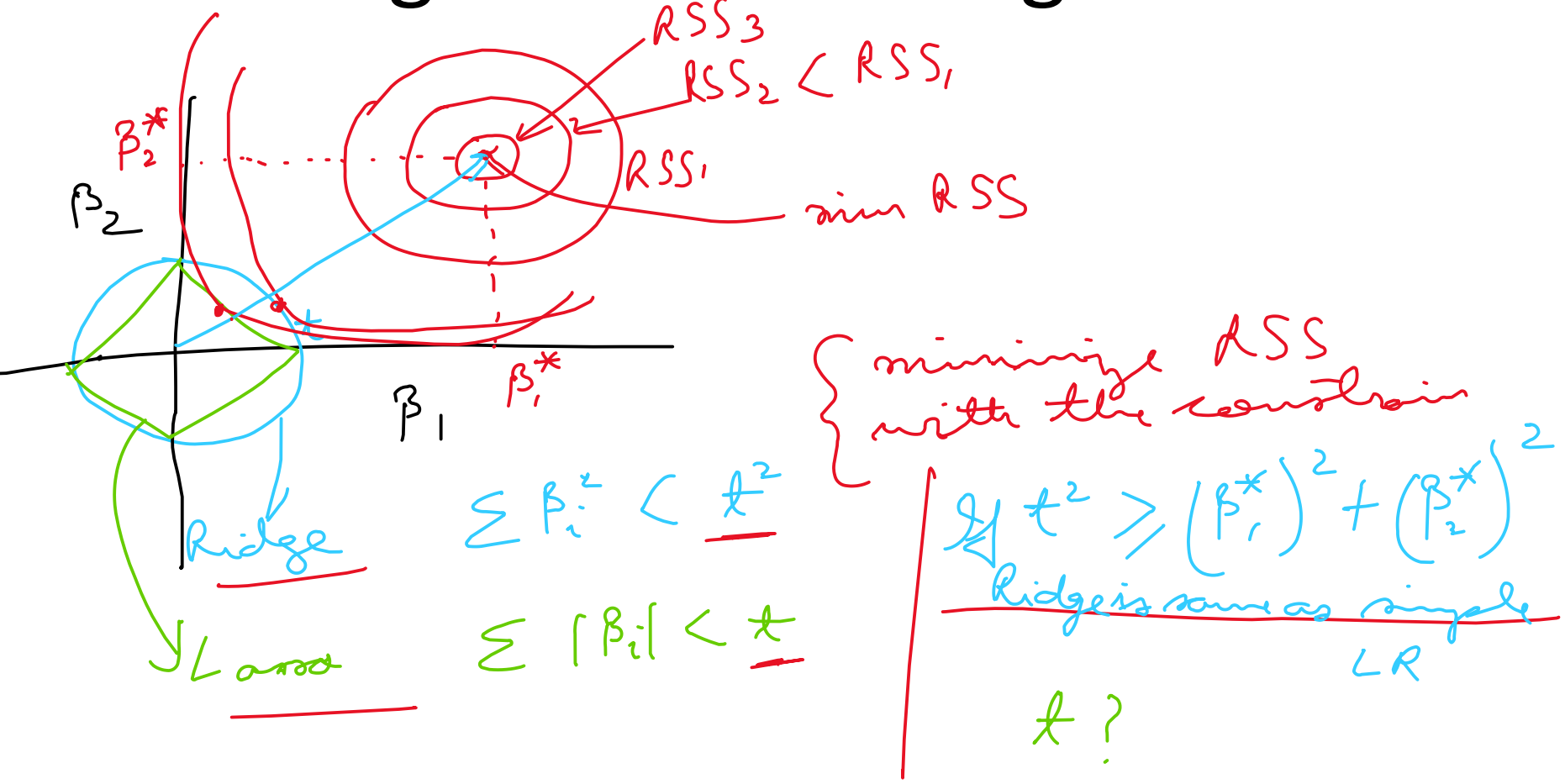
construct $\sum |\beta| < t$

$$\lambda \frac{\partial \beta^2}{\partial \beta} = 2\lambda\beta$$

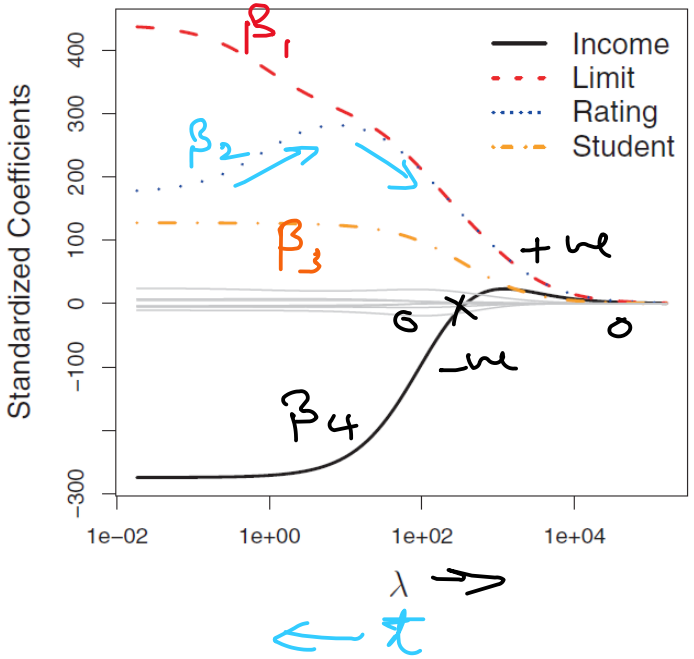


$$\lambda \frac{\partial |\beta|}{\partial \beta} \neq f(\beta) \Big|_{\beta=0}$$

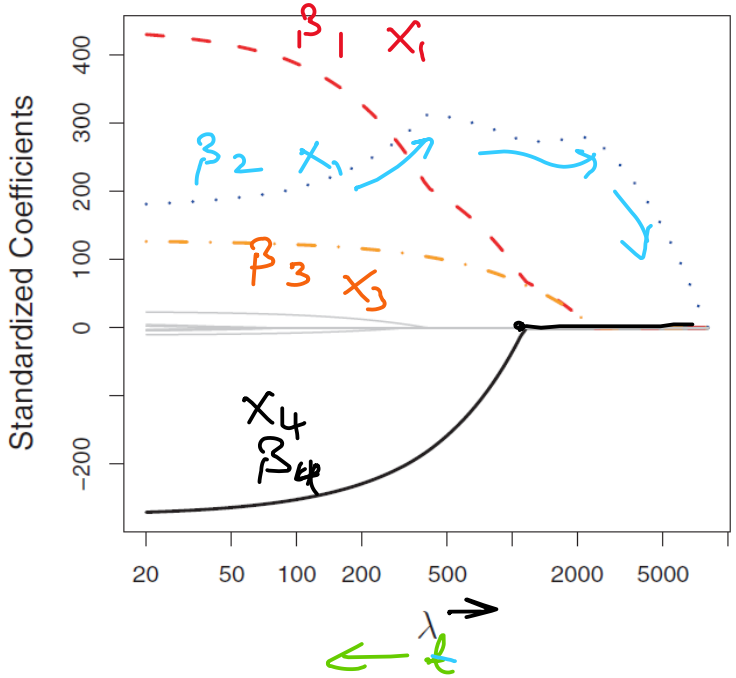
Ridge and Lasso Regression



Ridge and Lasso Regression



Ridge Regression



Lasso Regression

Ridge and Lasso Regression

$\|X\beta - Y\|_2^2$

$RSS_m < RSS_1 < RSS_2 < RSS_3$

Tangent Point

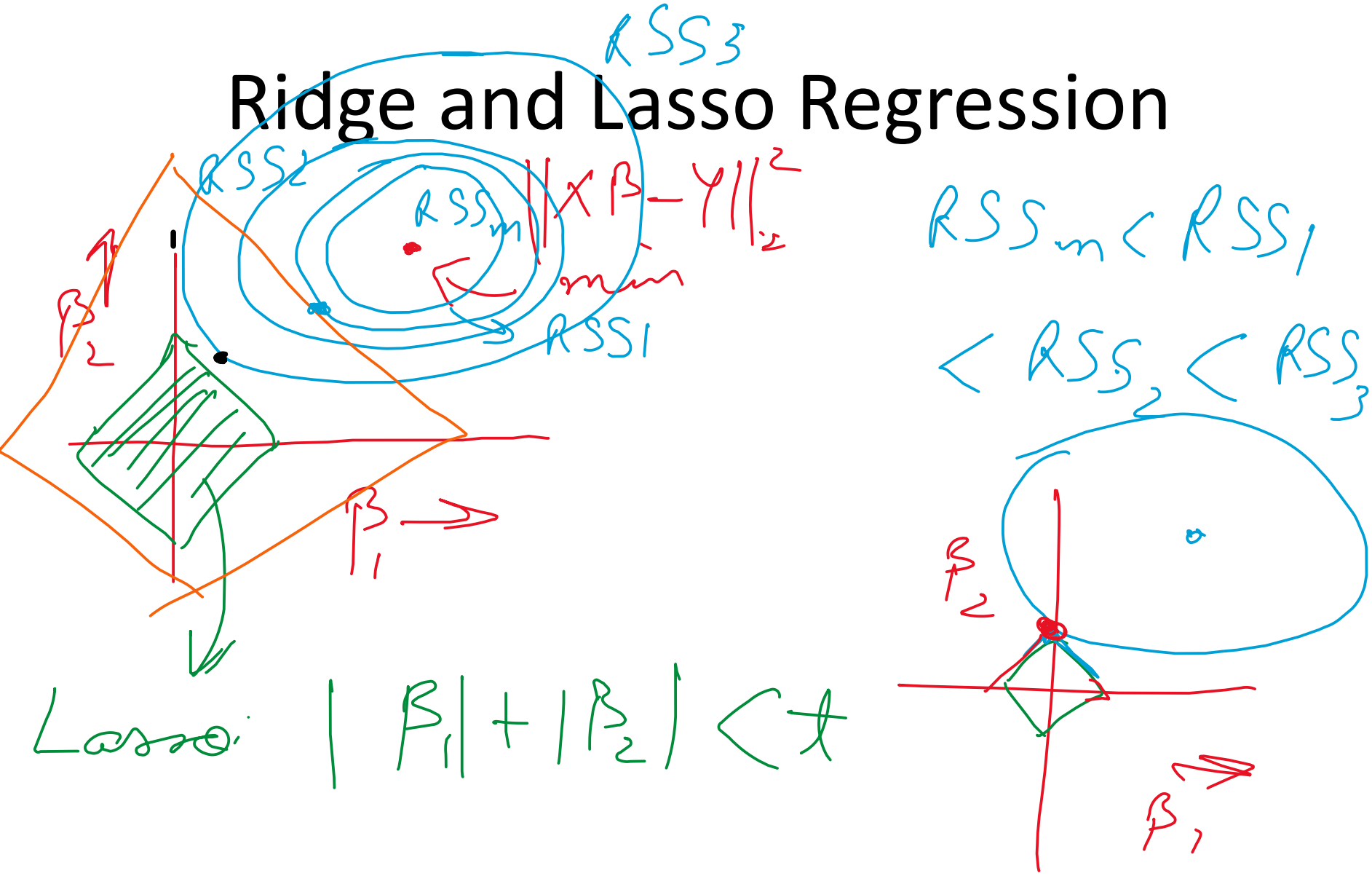
Ridge

$\sum \beta_i^2 < t$

$\beta_1^2 + \beta_2^2 < t$

Circle

Ridge and Lasso Regression



Ridge and Lasso Regression

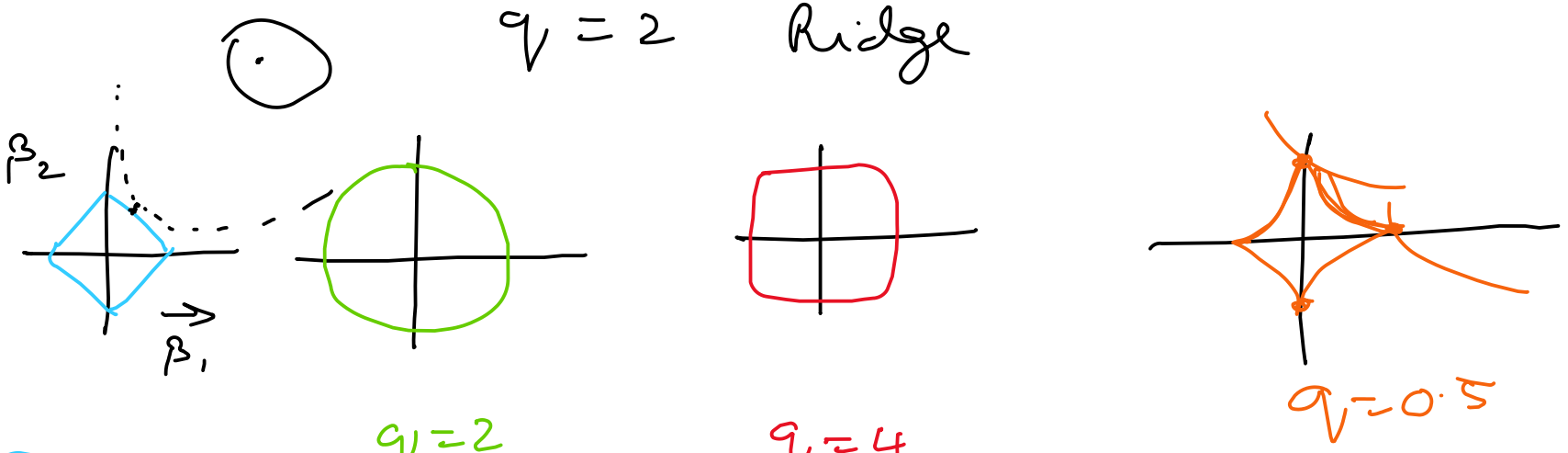
$$RSS(\beta) + \lambda \sum |\beta_i|^q$$

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

If $q = 0$ → OLR

$q = 1$ Lasso

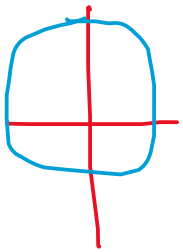
$q = 2$ Ridge

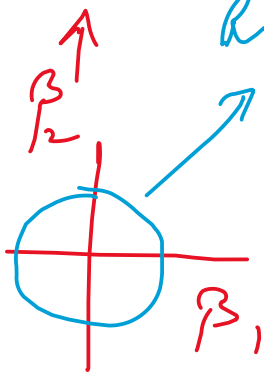


Ridge and Lasso Regression

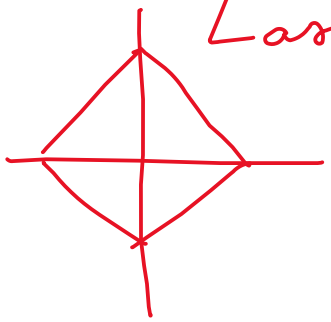
General Lasso $\lambda \sum |\beta_i|^q$

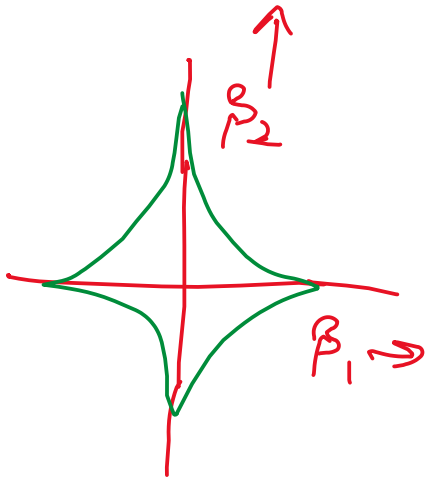
Ridge $\lambda \sum \beta_i^2$





Lasso $\lambda \sum |\beta_i|$





$q = 4$ $\quad$ $q = 2$ $\quad$ $q = 1$ $\quad$ $q = 0.5$

$\sum \beta_i^4 < t$

$|\beta_1|^{0.5} + |\beta_2|^{0.5} < t$

Shrinkage Methods

- An equivalent way to write the ridge problem is

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2,$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 \leq t,$$

Ridge Regression

- Ridge regression is very similar to least squares, except that the coefficients ridge regression are estimated by minimizing a slightly different quantity.

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

- where $\lambda \geq 0$ is a tuning parameter. The second term, is called a shrinkage penalty, it is small when $\beta_1,\ldots,\beta_p$ are close to zero

Shrinkage Methods

2.The lasso

Lasso is a shrinkage method like ridge, with subtle but important differences. The lasso estimate is defined by

$$\hat{\beta}^{\text{lasso}} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2$$

$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t.$$

Shrinkage Methods

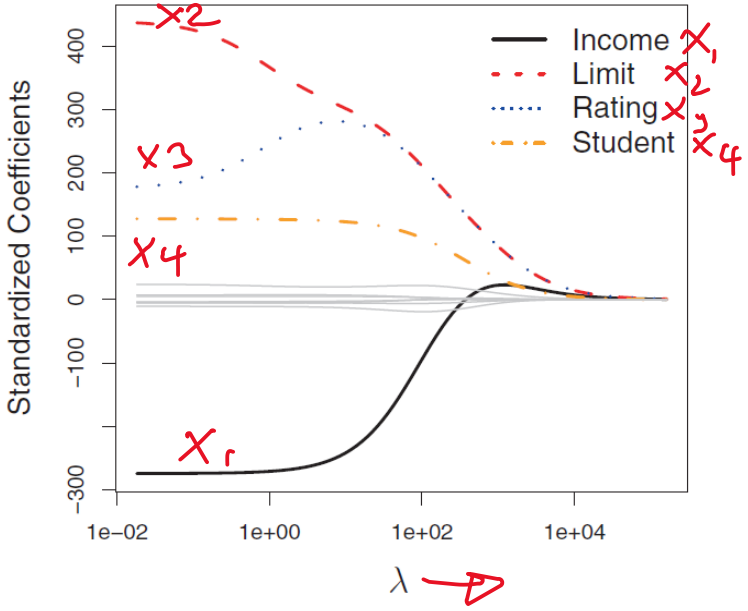
- We can also write the lasso problem in the equivalent Lagrangian form

$$\hat{\beta}^{\text{lasso}} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

$\frac{\partial (RSS)_{Lasso}}{\partial \beta_1} = \ldots\ldots + \lambda$ Lasso

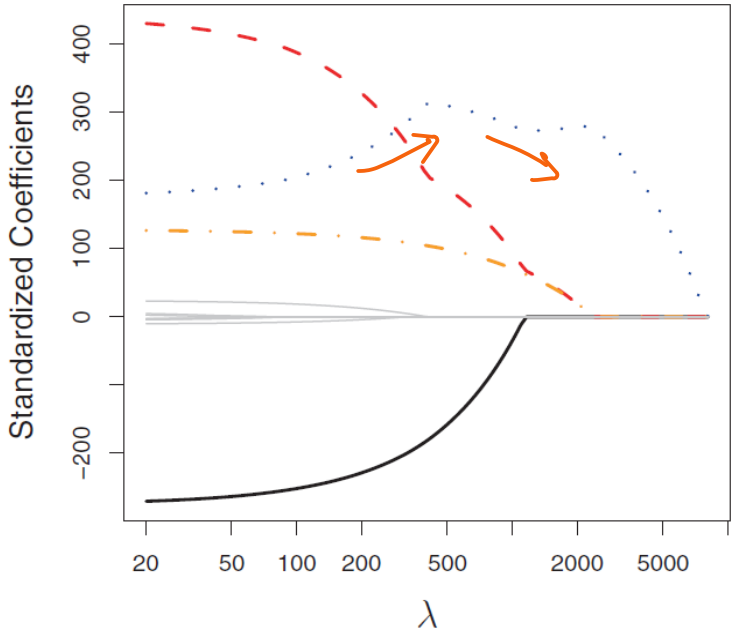
$\frac{\partial (RSS)_{Ridge}}{\partial \beta_1} = \ldots\ldots + 2\lambda\beta_1$

Ridge and Lasso Regression



Ridge Regression

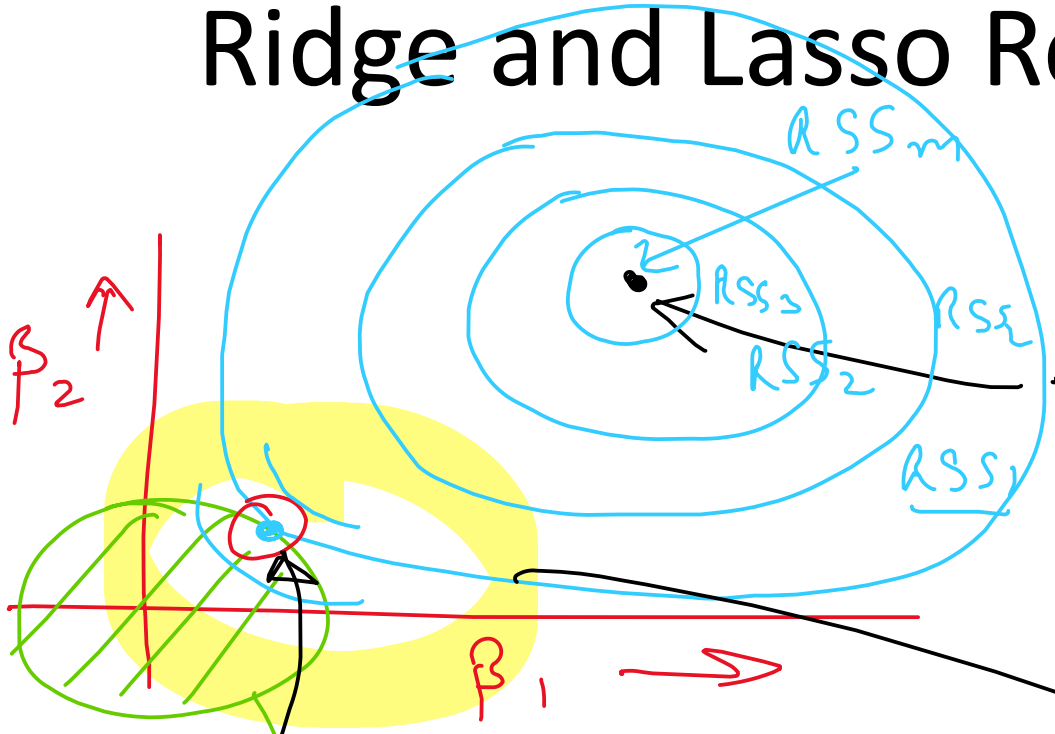
$\lambda \sum \beta_i^2$



Lasso Regression

$\lambda \sum |\beta_i|$

Ridge and Lasso Regression



$RSS = \|X\beta - y\|_2^2$

$\min$

$\min RSS$

$RSS_m < RSS_3 < RSS_2 < RSS_1$

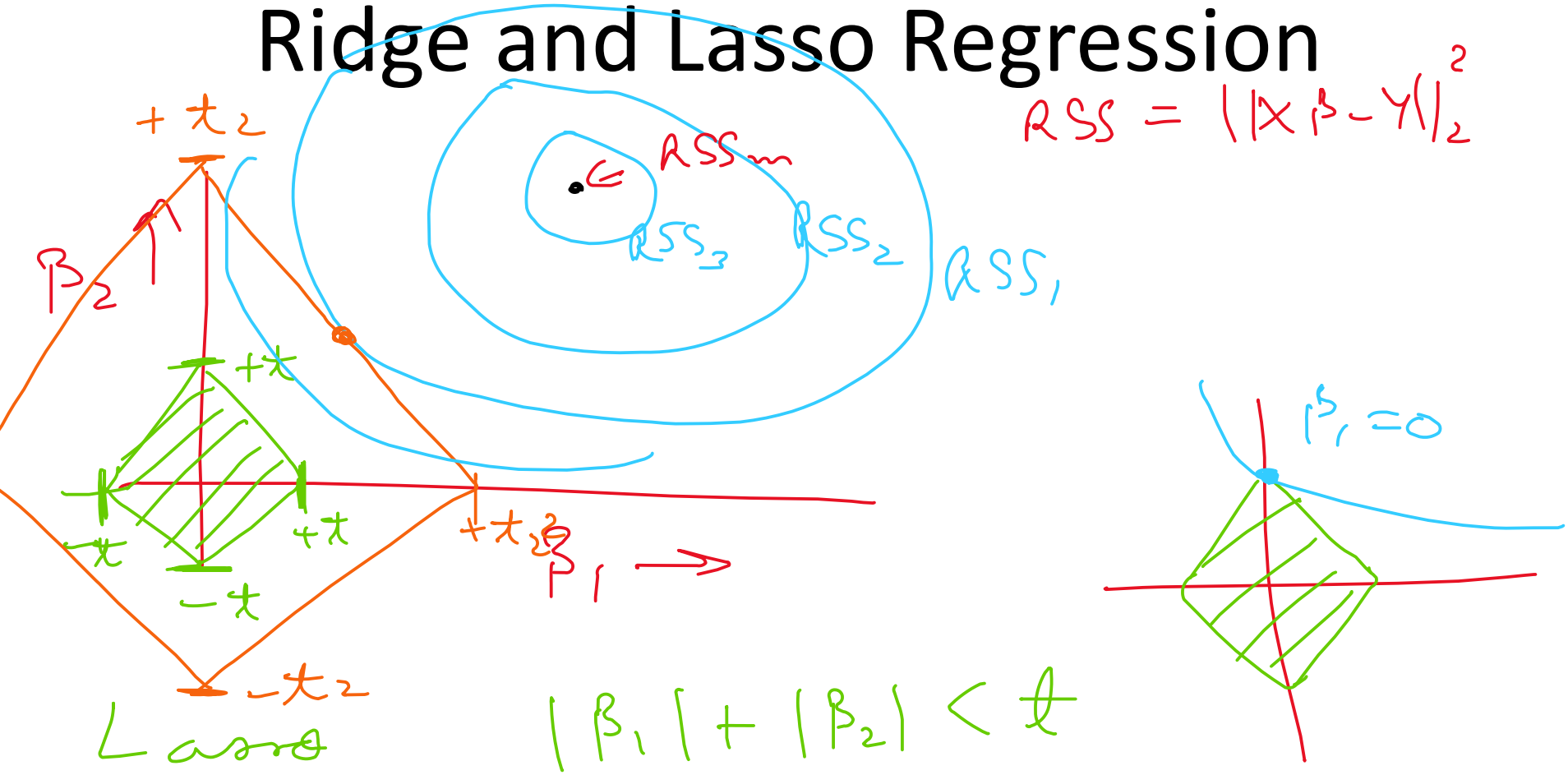
Ridge :

$\sum \beta_i^2 < t$

$\beta_1^2 + \beta_2^2 < t$

Tangent is the sol. for Ridge Regr...

Ridge and Lasso Regression



Summary

- Ridge regression does a proportional shrinkage.
- Lasso translates each coefficient by a constant factor , truncating at zero. This is called “soft thresholding,”.
- Best-subset selection drops all variables with coefficients smaller than the Mth largest; this is a form of “hard-thresholding.”

Dimensionality reduction

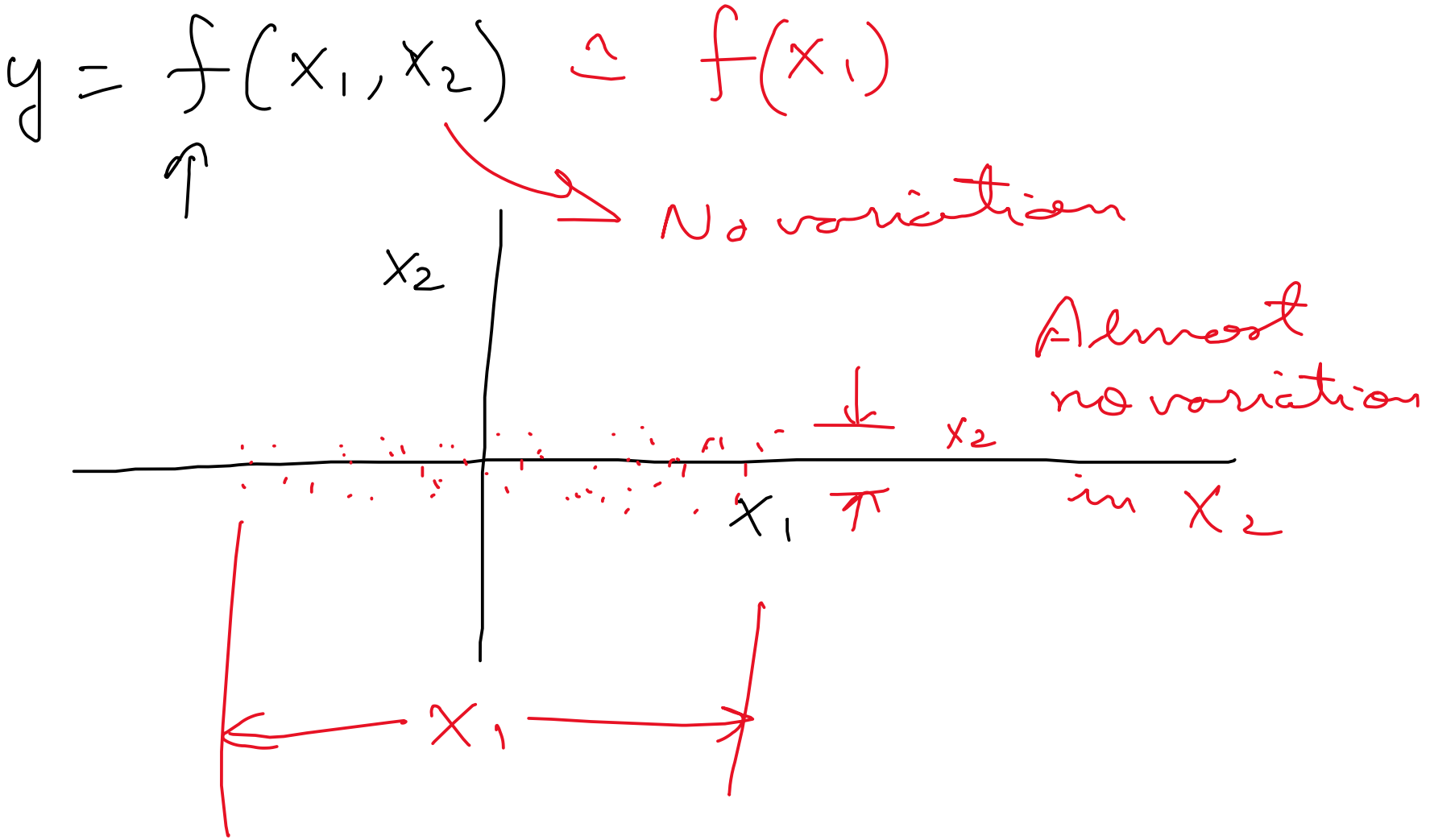
- Feature selection: Feature selection approaches try to find a subset of the original variables (also called features or attributes)
  - Filter strategy (e.g. information gain)
  - Wrapper strategy (e.g. search guided by accuracy)
  - Embedded strategy (features are selected to add or be removed while building the model based on the prediction errors)

- Feature projection: Feature projection transforms the data in the high-dimensional space to a space of fewer dimensions. The data transformation may be linear, as in principal component analysis (PCA), but many nonlinear dimensionality reduction techniques also exist.For multidimensional data, tensor representation can be used in dimensionality reduction through multilinear subspace learning.

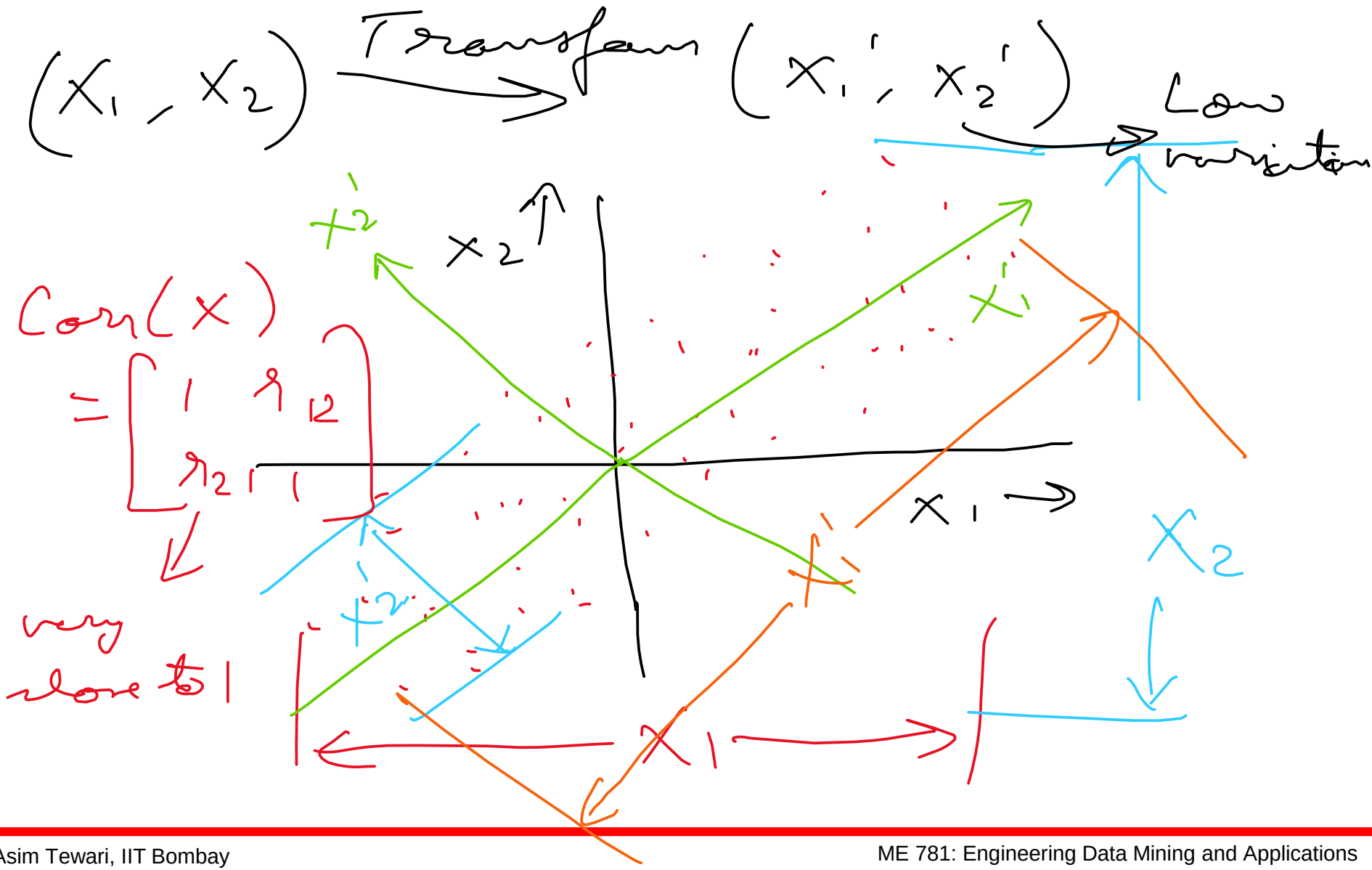
Feature projection

- Principal component analysis (PCA)
- Non-negative matrix factorization (NMF)
- Kernel PCA
- Graph-based kernel PCA
- Linear discriminant analysis (LDA)
- Generalized discriminant analysis (GDA)
- Autoencoder

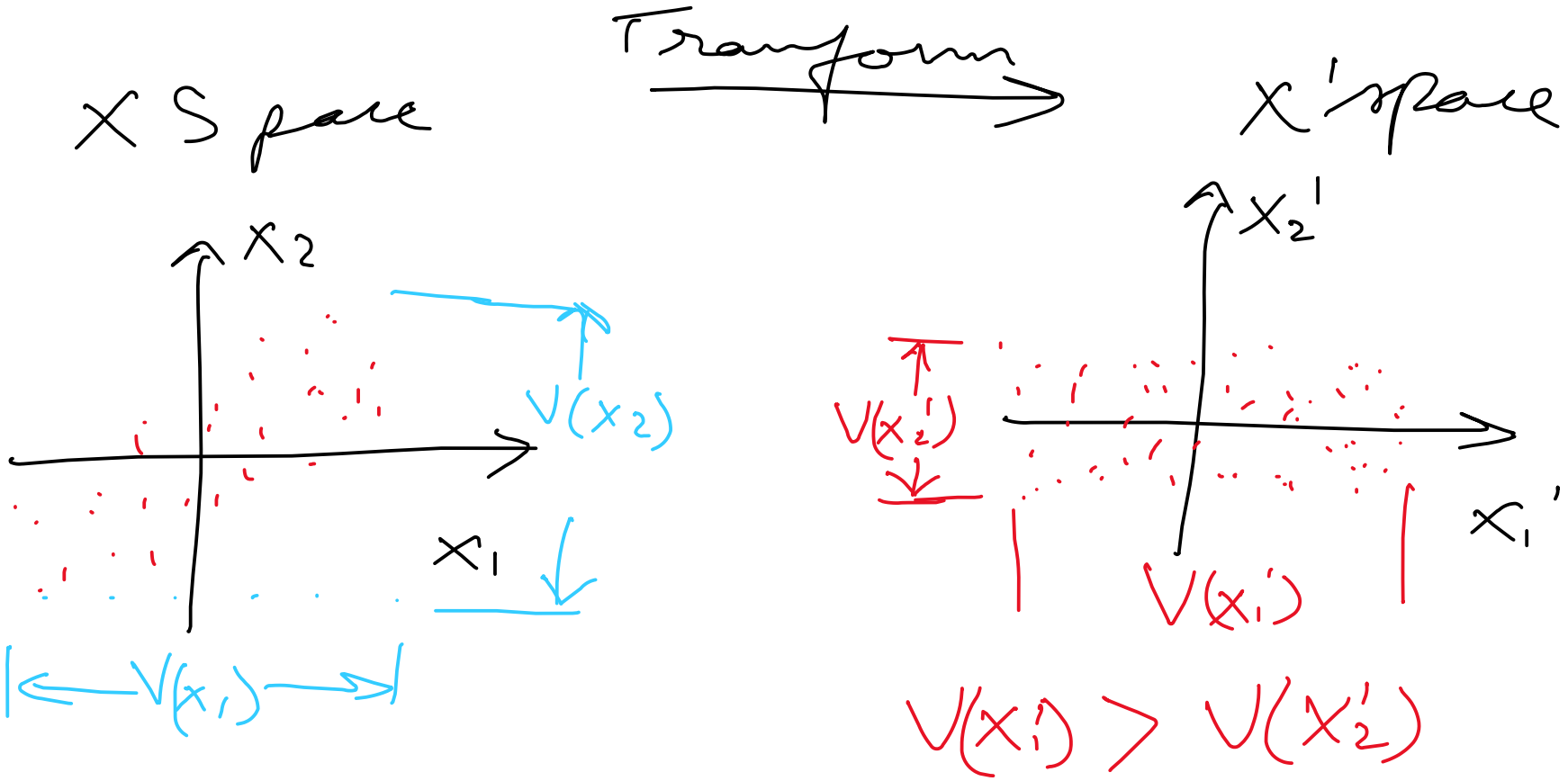
Principal Component Analysis



Principal Component Analysis



Principal Component Analysis



Principal Component Analysis

In 2D space with n data points

X space $\rightarrow$ X' space

$$X_{n\times 2} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \\ x_{n1} & x_{n2} \end{bmatrix} \Rightarrow X'_{n\times 2} = \begin{bmatrix} x'_{11} & x'_{12} \\ x'_{21} & x'_{22} \\ \vdots & \\ x'_{n1} & x'_{n2} \end{bmatrix}$$

variance

$V(x'_1) > V(x'_2)$

$\uparrow$

highest possible variance

Principal Component Analysis

In p-dimensional Space

PCA

x space $\rightarrow$ x' space

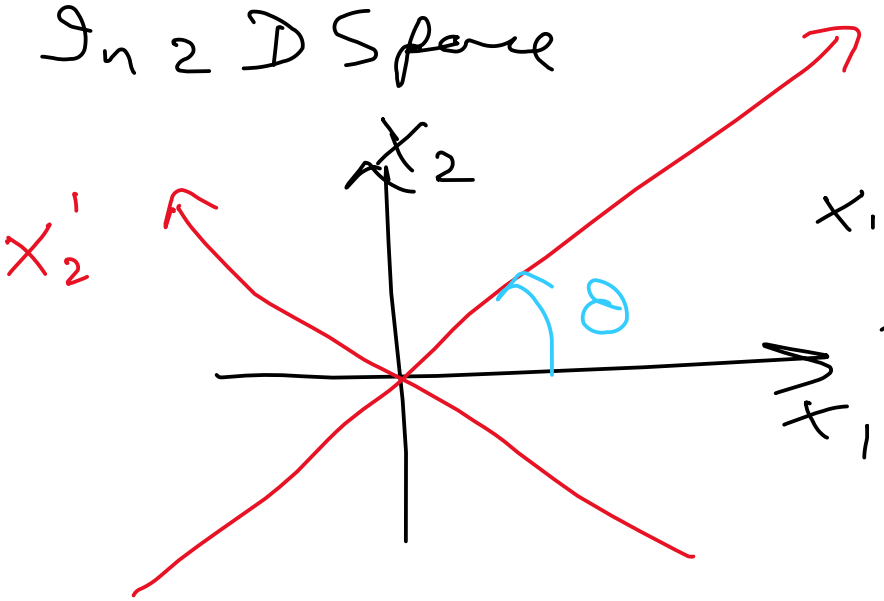
$$X_{n\times p} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ & & & \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

$$X'_{n\times p} = \begin{bmatrix} x'_{11} & x'_{12} & \cdots & x'_{1p} \\ x'_{21} & x'_{22} & \cdots & x'_{2p} \\ & & & \\ x'_{n1} & x'_{n2} & \cdots & x'_{np} \end{bmatrix}$$

$$V(x_1') \geq V(x_2') \geq \cdots V(x_p')$$

Principal Component Analysis

In 2D Space



$$x_1' = x_1 \cos\theta + x_2 \sin\theta$$

$$x_2' = -x_1 \sin\theta + x_2 \cos\theta$$

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$X' = A^T X$$

Principal Component Analysis

In 3D $\quad X' = A^T X$

$$\begin{bmatrix} x_1' \\ x_2' \\ x_3' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{R_3(\theta) = A^T} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

For rotation about x_3

Principal Component Analysis

$$R_2(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}$$

For a general rotation

$$A^T = R = R_1(\alpha)\, R_2(\beta)\, R_3(\gamma)$$

Principal Component Analysis

In general 3D rotation you can always find an axis that is not changed.

$$x' = Rx$$

x': new axis, x: old axis

Invariant axis

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = R \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \Rightarrow (R - I)x = 0$$

Invariant vector

Principal Component Analysis

$$x' = A^T x$$

$P \times 1 \quad P \times P \quad P \times 1$

In 2D

$$A^T = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

$$\Rightarrow A = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \end{bmatrix}$$

$$a_1 = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} ; \quad a_2 = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix}$$

Principal Component Analysis

$$A = \underset{2\times 2}{\begin{bmatrix} a_1 & a_2 \end{bmatrix}} \qquad \underset{2\times 1}{a_1} \qquad \underset{2\times 1}{a_2}$$

$$a_1 = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

$$a_2 = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix}$$

$$a_1^T a_1 = \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix} \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \cos^2\theta + \sin^2\theta = 1$$

$$a_2^T a_2 = 1$$

$$a_1^T a_2 = \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix} \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} = 0$$

Principal Component Analysis

$$A^T A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I = A A^T = A A^{-1}$$

$$= A^{-1} A$$

$$\Rightarrow A^T = A^{-1}$$

$\therefore$ A is an orthogonal Transformation

Principal Component Analysis

In P-dimension

$$A = [a_1 \; a_2 \; \ldots \; a_p]$$

$$a_i^T a_j = \delta_{ij} = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases}$$

In PCA we want this transformation to be such that variance of x' are as follows:

$$V(x_1') \geq V(x_2') \geq V(x_3') \ldots \geq V(x_p')$$

Principal Component Analysis

p-dimensional Space

$$X_{n \times p} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ \vdots & & & \\ x_{k1} & x_{k2} & \cdots & x_{kp} \\ \vdots & & & \\ x_{n1} & \cdots & \cdots & x_{np} \end{bmatrix}_{n \times p}$$

In set form $X = \{ x_1, x_2 \cdots x_n \}$

where $x_k = (x_{k1}\ x_{k2} \cdots x_{kp})$

Principal Component Analysis

∴ for the "k^{th}" Set

$$x_k = \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{kp} \end{bmatrix}^T \quad ; \quad \text{Similarly} \quad \begin{bmatrix} x'_{k1} \\ x'_{k2} \\ \vdots \\ x'_{kp} \end{bmatrix}^T = \begin{bmatrix} x'_{k1} & x'_{k2} & \cdots & x'_{kp} \end{bmatrix} = x'_k$$

$$\begin{bmatrix} x'_{k1} \\ x'_{k2} \\ \vdots \\ x'_{kp} \end{bmatrix} = A^T \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{kp} \end{bmatrix}$$

Principal Component Analysis

$$\left[\begin{bmatrix} x'_{k1} \\ x'_{k2} \\ \vdots \\ x'_{kp} \end{bmatrix}\right]^T = \left[A^T \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{kp} \end{bmatrix}\right]^T$$

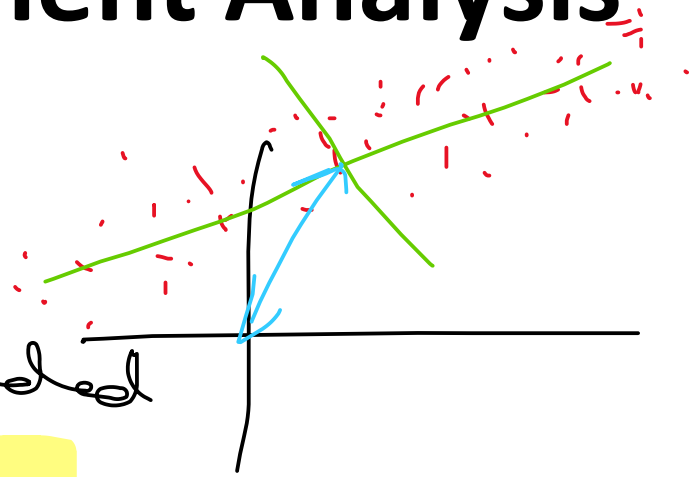
$$\begin{bmatrix} x'_{k1} \\ x'_{k2} \\ \vdots \\ x'_{kp} \end{bmatrix}^T = \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{kp} \end{bmatrix}^T A$$

$$x'_k = x_k A$$

q $\uparrow$

Principal Component Analysis

$$x'_k = x_k A$$



Translation is also needed

$$\underset{1\times p}{x'_k} = \left(\underset{1\times p}{x_k} - \bar{x}\right)\underset{p\times p}{A}$$

$$\bar{x} = \frac{1}{n}\sum_{k=1}^{n} x_k = \frac{1}{n}\left[\sum x_{k1} \quad \sum x_{k2} \cdots \quad \sum x_{kp}\right]$$

$$x_k = x'_k A^T + \bar{x}$$

Principal Component Analysis

$$x_k = x'_k A^T + \bar{x}$$

variance vector of $x' = \begin{bmatrix} var(x'_1) & var(x'_2) & \cdots & var(x'_p) \end{bmatrix}$

covariance of $x' = \Sigma_{x'} = \frac{1}{n-1} \sum_{k=1}^{n} (x'_k)^T (x'_k)$

$\Sigma_{x'}$: $p \times p$; $(x'_k)^T$: $p \times 1$; (x'_k): $1 \times p$

$$\Sigma_{x'} = \frac{1}{n-1} \sum_{k=1}^{n} (x'_k)^T (x'_k)$$

Principal Component Analysis

$$\Sigma_{X'} = \frac{1}{n-1} \sum_{k=1}^{n} (X'_k)^T (X'_k)$$

$$X'_k = (X_k - \bar{X}) A$$

$$\Sigma_{X'} = \frac{1}{n-1} \sum \left[(X_k - \bar{X}) A \right]^T \left[(X_k - \bar{X}) A \right]$$

$$= \frac{1}{n-1} \sum_{k=1}^{n} A^T (X_k - \bar{X})^T \cdot (X_k - \bar{X}) A$$

$$= A^T \left[\frac{1}{n-1} \sum_{k=1}^{n} (X_k - \bar{X})^T \cdot (X_k - \bar{X}) \right] A$$

$\Sigma_{X'} = A^T C A$ ← Covariance matrix of X

Principal Component Analysis

$$\Sigma_{x'} = A^T C A$$

C ← covariance matrix of X

$$c_{ij} = \frac{1}{n-1} \sum_{k=1}^{n} \left[x_{ki} - \frac{1}{n} \sum_{l=1}^{n} x_{li} \right] \left[x_{kj} - \frac{1}{n} \sum_{l=1}^{n} x_{lj} \right]$$

$$i, j \in \{1, 2, \ldots p\}$$

Choose A such that $\Sigma_{x'}$ is maximized.

However $A^T A = I$

Principal Component Analysis

Maximise $V_{x'}$ given $A^T A = I$

Constrained optimization.

Construct a Lagrange function

$$L = A^T C A - \lambda (A^T A - I)$$

Sol is λ which maximises L

$$\Rightarrow \frac{\partial L}{\partial A} = 0 \Rightarrow C A - \lambda A = 0$$

$$\Rightarrow (C - \lambda I) A = 0$$

Principal Component Analysis

$(C - \lambda I)A = 0$ Solve for A.

This is an Eigenvalue / Eigenvector Problem.

This started by solving DEs:

$$\frac{dy}{dt} = \underset{n \times n}{A}\, y \Big\} \; n \text{ linear DEs}$$

Sol is of the type $y(t) = e^{\lambda t} X$

Principal Component Analysis

$$y(t) = e^{\lambda t} x$$

$$\Rightarrow \lambda e^{\lambda t} x = A e^{\lambda t} x$$

$$\Rightarrow \lambda x = A x$$

$$(A - \lambda I) x = 0$$

$n \times n$ $\quad$ $n \times n$ $\quad$ $n \times 1$

$n \times 1$

$$\frac{dy}{dt} = A y \Big\} \text{ } n \text{ linear DEs}$$

$$(C - \lambda I) A = 0$$

$p \times p$ $\quad$ $p \times p$ $\quad$ $p \times p$

$p \times p$

$(A - \lambda I)X = 0$ $\}$ gives eigenvalues of λ and eigenvectors of X

$\uparrow$

$n \times 1$

Eg. $y_1' = 5y_1 + y_2$

$y_2' = 3y_1 + 3y_2$ $\Big\}$ $\dfrac{dy}{dt} = \begin{bmatrix} 5 & 1 \\ 3 & 3 \end{bmatrix} \begin{bmatrix} y \end{bmatrix}$

$\lambda_1 = 6 \; ; \; X_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \; ; \; \lambda_2 = 2 \; ; \; X_2 = \begin{bmatrix} 1 \\ -3 \end{bmatrix}$

$$y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = c_1 e^{\lambda_1 t} X_1 + c_2 e^{\lambda_2 t} X_2$$

$$y(t) = c_1 e^{6t} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + c_2 e^{2t} \begin{bmatrix} 1 \\ -3 \end{bmatrix}$$

c_1, c_2 $\leftarrow$ get from BCs

Case 1

If we know the eigenvector for $Ax = \lambda_1 x$, then we also know the eigenvector for $A^2 x = \lambda_2 x$

$$A^2 x = \lambda_2 x \Rightarrow A(Ax) = \lambda_2 x$$

$$\Rightarrow A(\lambda_1 x) = \lambda_2 x$$

$$\Rightarrow \lambda_1^2 x = \lambda_2 x$$

$$\Rightarrow \lambda_2 = \lambda_1^2$$

and they have the same eigenvector

Case 2

If $(A + cI)x = \lambda_3 x$

$\lambda_3 = (\lambda_1 + c)$

Similarly $A^n x = \lambda_n x$

then $\lambda_n = (\lambda_1)^n$

$(C - \lambda I)A = 0$ → This is a concatenation of p eigenvectors

A: $p \times p$

$(A - \lambda I)x = 0$

x: $n \times 1$

$A = [a_1\ a_2 \cdots a_p]$

A: $p \times p$

$$(C - \lambda I) A = 0$$

$$A = (a_1, a_2 \ldots a_p)$$

eigenvectors $\longleftrightarrow$ $(\lambda_1, \lambda_2 \ldots \lambda_p)$ eigenvalues.

$\Rightarrow$ Variance in X' corresponds to the eigenvalue of $\lambda_1 \; \lambda_2 \ldots \lambda_p$

$$CA = \lambda A \Rightarrow A^T C A = A^T \lambda A$$

$A^T A = I$

$$\lambda = A^T C A \equiv \Sigma_{X'}$$

$$\lambda_1 \geq \lambda_2 \geq \lambda_3 \ldots \geq \lambda_p$$

Principal Component Analysis

Thus, if we want to capture 95% variance, then we reduce the # of dimensions to q such that

$$\frac{\sum_{i=1}^{q} \lambda_i}{\sum_{i=1}^{p} \lambda_i} \geq \frac{95}{100}$$

Principal Component Analysis

Eg. $p = 2, \quad n = 4$

$$X = \{(1,1), (2,1), (2,2), (3,2)\}$$

$$\Rightarrow \bar{X} = \left(2, \frac{3}{2}\right); \quad C = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

The eigenvalues are

$$\lambda_1 = 0.8727, \quad a_1 = \begin{bmatrix} -0.85065 \\ -0.52571 \end{bmatrix}$$

$$\lambda_2 = 0.1273, \quad a_2 = \begin{bmatrix} 0.52571 \\ -0.85065 \end{bmatrix}$$

$$A = [a_1, a_2] = \begin{bmatrix} -0.85065 & 0.52571 \\ -0.52571 & -0.85065 \end{bmatrix}$$

Principal Component Analysis

$$x' = A^T(x - \bar{x}) \Rightarrow x_1' = \begin{bmatrix} -0.85065 & -0.52571 \\ 0.52571 & -0.85065 \end{bmatrix} \begin{bmatrix} 1 - 2 \\ 1 - 1.5 \end{bmatrix}$$

$$\Rightarrow \underline{x_1}' = \begin{bmatrix} 1.1135 \\ -0.10037 \end{bmatrix}$$

$$\underline{x_2}' = \begin{bmatrix} 0.2628 \\ 0.42533 \end{bmatrix}, \quad \underline{x_3}' = \begin{bmatrix} -0.2628 \\ -0.42533 \end{bmatrix}$$

$$\underline{x_4}' = \begin{bmatrix} -1.1135 \\ 0.10037 \end{bmatrix}$$

$X \xrightarrow[A^T]{\text{Transformed}} X'$

Principal Component Analysis

$X \xrightarrow{A^T} X' \quad ; \quad A^T = \begin{bmatrix} -0.85065 & -0.52571 \\ 0.52571 & -0.85065 \end{bmatrix}$

X

$\{(1,1), (2,1), (2,2), (3,2)\}$

$\text{mean } X = (2, 1.5)$

Covariance of X

$= \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} \end{bmatrix}$

X'

$\{(1.1135, -0.10039),$
$(0.26286, 0.42533),$
$(-0.26286, -0.42533)$
$(-1.1135, -0.10039)\}$

$\text{Mean } X' = (0, 0)$

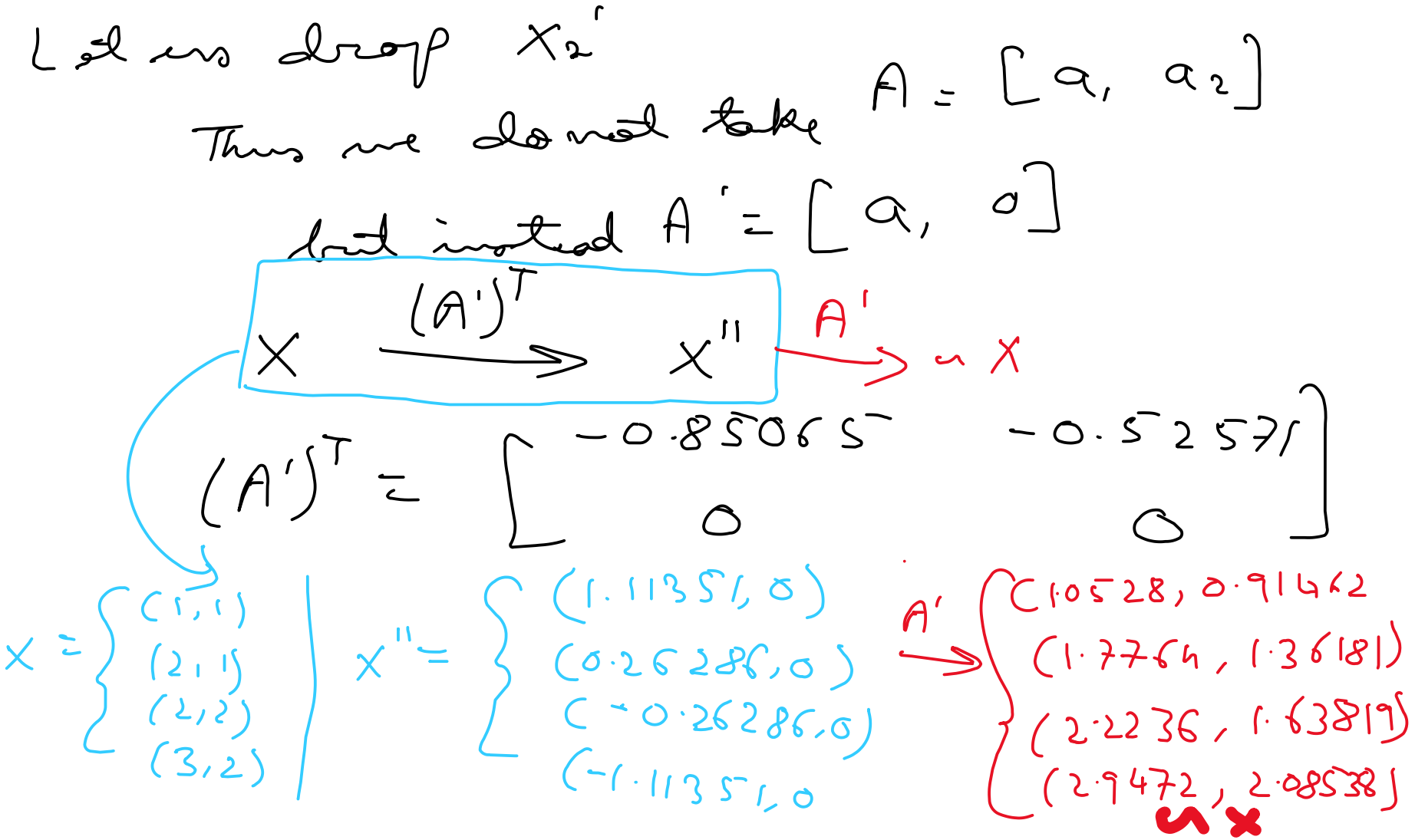
$\text{Cov } X' = \begin{bmatrix} 0.87266 & 0 \\ 0 & 0.12732 \end{bmatrix}$ (λ_1, λ_2)

Principal Component Analysis

$$Cov\,X' = \begin{bmatrix} \lambda_1 & 0 & \\ 0 & \lambda_2 & \\ & & \lambda_p \end{bmatrix}$$

Since X_1' containing 87% of variance, we can drop X_2'

Principal Component Analysis



Principal Component Analysis

$$X = \{(1,1),(2,1),(2,2),(3,2)\}$$

$$\bar{x} = \frac{1}{2} \cdot (4,3)$$

$$C = \frac{1}{3} \cdot \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$$

$$\lambda_1 = 0.8727$$

$$\lambda_2 = 0.1273$$

$$v_1 = \begin{pmatrix} -0.85065 \\ -0.52573 \end{pmatrix}$$

$$v_2 = \begin{pmatrix} 0.52573 \\ -0.85065 \end{pmatrix}$$

$$E = v_1 = \begin{pmatrix} -0.85065 \\ -0.52573 \end{pmatrix}$$

The projected data

$$Y = \{1.1135, 0.2629, -0.2629, -1.1135\}$$

Inverse PCA yields

$$X' = \{\ (1.0528, 0.91459),\ (1.7764, 1.3618),\ (2.2236, 1.6382),\ (2.9472, 2.0854)\ \} \neq X$$