

202A Fall 2017

Pro. Wu

Lecture 2

10/2/2017

Linear regression.

		Predictors (genes)		outcome
	1 2 ... j P			
1	$x_{11} x_{12} \dots x_{1P}$		y_1	
2	$x_{21} x_{22} \dots x_{2P}$		y_2	
:	$x_{ij} \dots x_{iP}$		y_i	
n			y_n	
(people)				

$$y_i = \sum_{j=1}^P x_{ij} \beta_j + \varepsilon_i$$

$$(\beta_0 + \sum_{j=1}^P x_{ij} \beta_j)$$

$$\downarrow \text{intersect} \quad \beta_i = \text{intersect}$$

$$R(\beta) = \sum_{i=1}^n (y_i - \sum_{j=1}^P x_{ij} \beta_j)^2$$

$n \gg P$ classic

$P \gg n$ modern

$$\begin{matrix} & \xrightarrow{P} \\ \xrightarrow{n} & X \quad Y \\ \xrightarrow{n \times P} & \xrightarrow{n \times 1} \end{matrix}$$

	1 2 ... j P	
1		
2		
i	x_i^T	
n		

$$x_i^T = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{iP} \end{pmatrix} \quad \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_P \end{pmatrix}$$

	1 2 ... j P	
1		
2		
i		
n		

$$Y = \sum_{j=1}^P x_{ij} \beta_j + \varepsilon \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

$$\begin{array}{c} Y \\ \nearrow x_1 \\ \nearrow \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \\ \searrow x_2 \end{array}$$

$$y_i = x_i^T \beta + \varepsilon_i$$

$$R(\beta) = \sum_{i=1}^n (y_i - x_i^T \beta_i)^2$$

$$R(\beta) = \| Y - \sum_{j=1}^P x_j \beta_j \|^2$$

least squares projection

$$\|Y - X\beta\|^2 + \lambda \|\beta\|$$

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

$$\|\beta\|_2^2 = \sum_{j=1}^p \beta_j^2$$

$n \times 1 - (n \times p) \times (p \times 1)$

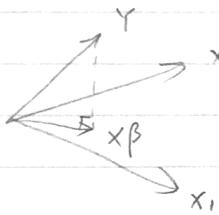
$$R(\beta) = \|Y - X\beta\|^2$$

$$R'(\beta) = -2X^T(Y - X\beta) = 0$$

$P \times 1$

$$X^T X \beta = X^T Y$$

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



$$x_j \perp Y - X\beta$$

$$X_j^T(Y - X\beta) = 0$$

$$X^T(Y - X\beta) = 0$$

$$E[(X^T X)^{-1} X^T \epsilon]$$

$$E[(X^T X)^{-1} X^T \epsilon] = E[X^T \epsilon]$$

$$\text{Var}(X) = E(X^2) - (E(X))^2$$

$$= E(X^2) - 0$$

$$(X^T X)^{-1} X^T \epsilon \sim N(0, \sigma^2 I_n)$$

✓

$$\text{True model } Y = X\beta_{\text{true}} + \epsilon$$

$$E[\epsilon] = 0 \quad \text{Var}(\epsilon) = \sigma^2 I_n$$

$$\epsilon \sim N(0, \sigma^2 I_n)$$

$$\begin{aligned} \hat{\beta} &= (X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T (X\beta_{\text{true}} + \epsilon) \\ &= \beta_{\text{true}} + (X^T X)^{-1} X^T \epsilon \end{aligned}$$

$$\begin{aligned} \hat{\beta} &= X(X^T X)^{-1} X^T Y \\ \hat{Y} &= \underline{H} Y \end{aligned}$$

$$|\epsilon|^2 = \sum_{i=1}^n (y_i - x_i^T \hat{\beta})^2$$

$$RSS = Y^T Y - \hat{Y}^T \hat{Y} = Y^T Y - Y^T H^T H Y = [Y^T Y - Y^T X (X^T X)^{-1} X^T Y]$$

$$\|Y - \hat{Y}\|^2 = (Y - \hat{Y})^T (Y - \hat{Y}) = Y^T Y - Y^T \hat{Y} - \hat{Y}^T Y + \hat{Y}^T \hat{Y}$$

Residual sum of square $(Y^T - \hat{Y}^T)(Y - \hat{Y})$

$$\begin{aligned} &= Y^T Y - 2Y^T \hat{Y} + \hat{Y}^T \hat{Y} \\ &= Y^T Y - 2Y^T (X\hat{\beta}) \neq (\hat{\beta}^T X^T)(X\hat{\beta}) \\ &= Y^T Y - 2Y^T X \hat{\beta} + \hat{\beta}^T X^T X \hat{\beta} \end{aligned}$$

Lecture 3

10/15/2017

Linear Regression

$$Y = X\beta + \varepsilon \quad V$$

$$E(\varepsilon) = 0 \quad V$$

$$\text{Var}(\varepsilon) = \sigma^2 I_n \quad V$$

$$R^2 = |Y - X\beta|^2$$

$$R'(\beta) = -2X^T(Y - X\beta) = 0$$

$$\hat{\beta}_{LS} = (X^T X)^{-1} X^T Y \quad \begin{matrix} \text{least square} \\ \text{square} \end{matrix} \quad E(\hat{\beta}) = \beta_0$$

$$\text{Var}(\hat{\beta}) = \sigma^2 (X^T X)^{-1}$$

$$e = Y - X\hat{\beta}$$

$$RSS = \|e\|^2 = \|Y^T Y - Y^T X(X^T X)^{-1} X^T Y\|$$

$$f^2 = \frac{RSS}{n} \quad \text{or} \quad \frac{RSS}{n-p} \quad (\text{over fitting})$$

Gauss-Jordan elimination

$$\begin{cases} x_1 + x_2 + x_3 = 5 \\ 2x_1 + 3x_2 + 5x_3 = 8 \\ 4x_1 + 5x_3 = 2 \end{cases} \quad A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 5 \\ 4 & 0 & 5 \end{bmatrix} \quad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad b = \begin{pmatrix} 5 \\ 8 \\ 2 \end{pmatrix}$$

$$Ax = b$$

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

$$\left[\begin{array}{ccc|c} 1 & 1 & 1 & 5 \\ 2 & 3 & 5 & 8 \\ 4 & 0 & 5 & 2 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 1 & 1 & 1 & 5 \\ 0 & 1 & 3 & -2 \\ 0 & -4 & 1 & -18 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 1 & 0 & -2 & 7 \\ 0 & 1 & 3 & -2 \\ 0 & 0 & 13 & -26 \end{array} \right]$$

$$\rightarrow \left[\begin{array}{ccc|c} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & -2 \end{array} \right]$$

$$[A | b] \xrightarrow{GJ[1..n]} [I | A^{-1}b]$$

$$GJ[1..n] = A^{-1}$$

$$A^{-1}[A|b] = [I|A^{-1}b]$$

$$[A|I] \xrightarrow{GJ[1..n]} A^{-1}[A|I] = [I|A^{-1}]$$

$$\begin{matrix} a_{j1} & a_{ij} \cdot a_{ij} \\ \uparrow & \frac{a_{ii}}{a_{ii}} \\ a_{11} & \frac{a_{ij}}{a_{ii}} \\ \uparrow & a_{ii} \quad a_{ij} \\ a_{ii} & a_{ij} \\ \uparrow & a_{ii} \\ a_{ij} & a_{ii} \end{matrix}$$

$$A = (a_{ij})_{n \times n}$$

$$B = \begin{matrix} n \\ \left[A | I \right] \end{matrix} \xrightarrow{GJ[\text{all}]} \begin{matrix} n \\ \left[\begin{matrix} I & \cdots & \frac{a_{ii}}{a_{ii}} \\ 0 & \cdots & 0 \\ 0 & a_{ij} - \frac{a_{ij}a_{ii}}{a_{ii}} & -\frac{a_{ii}}{a_{ii}} \end{matrix} \right] \end{matrix}$$

$$\begin{matrix} X^T X & X^T Y \\ \uparrow & \uparrow \\ \begin{matrix} A_{11} & A_{12} \\ m \times m & m \times m \end{matrix} & \left[\begin{matrix} I_1 & 0 \\ 0 & I_2 \end{matrix} \right] \end{matrix} \xrightarrow{GJ[1..m]} \begin{matrix} n \\ \left[\begin{matrix} I_1 & A_{11}^{-1} A_{12} & A_{11}^{-1} 0 \\ 0 & A_{22} - A_{21} A_{11}^{-1} A_{12} & -A_{21} A_{11}^{-1} I \end{matrix} \right] \end{matrix}$$

$$\begin{matrix} Y^T X & Y^T Y \\ \uparrow & \downarrow \\ \begin{matrix} A_{21} & A_{22} \\ n \times m & n \times m \end{matrix} & \left[\begin{matrix} 0 & I_2 \\ I_1 & 0 \end{matrix} \right] \end{matrix} \xrightarrow{GJ[1..m]} B$$

$$Z = (X \quad Y)$$

$n \times p \quad n \times l$

$$A_{11}^{-1}$$

$$B = A + I$$

$$n \times n$$

$$Z^T Z = \begin{pmatrix} X^T \\ Y^T \end{pmatrix} (X \quad Y) = \left[\begin{pmatrix} X^T X & X^T Y \\ Y^T X & Y^T Y \end{pmatrix} \mid I \right] \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$\xrightarrow{GJ[1..m]} = \left(\begin{matrix} I_1 & \hat{\beta} \\ 0 & RSS \end{matrix} \mid \begin{matrix} \frac{1}{\delta} & 0 \\ -\hat{\beta}^T & I_2 \end{matrix} \right) \begin{bmatrix} I & \frac{a_{12}}{a_{11}} \\ a_{21} & a_{22} \end{bmatrix}$$

$$n \left[\begin{matrix} A_{11} & A_{12} \\ (n-m) \times n & A_{22} \quad A_{21} \end{matrix} \right]$$

$$\begin{matrix} m & n \\ \underbrace{\hspace{1cm}}_{m \times m} & \underbrace{\hspace{1cm}}_{m \times (n-m)} \\ \hline & \end{matrix}$$

$$(n-m) \times m \quad (n-m) \times (n-m)$$

$$\begin{bmatrix} I & \frac{a_{12}}{a_{11}} \\ a_{21} - I \cdot a_{21} & a_{22} - \frac{a_{12}}{a_{11}} \cdot a_{21} \end{bmatrix}$$

my Gauss Jordan ← function (A, m)

n = dim(A) [1]

B ← cbind (A, diag (rep(1, n)))
In

for (k in 1:m)

{

a ← B [k, k]

for (j in 1:(n*2)) B[k, j] ← B[k, j]/a

for (i in 1:n)

if (i != k)

{

a ← B [i, k]

for (j in 1:(n*2))

B [i, j] ← B [i, j] - b * B [k, j]

}

}

return(B)

}

my Gauss Jordan ← funct(A, m)

n <

B <

for (k in 1:n)

{

B [k,] ← B [k,] / B [k, k]

for (i in 1:n)

if (i != k)

B [i,] ← B [i,] - B [k,] * B [i, k]

Python:

def myGaussJordan (A, m):

n = A.shape[0]

B = np.stack((A, np.identity(n)))

for k in range(m):

a = [B, k]

for j in range(n*2):

B[k, j] = B[k, j]/a

for i in range(n):

if i != k:

b = B[i, k],

B[i, :] = B[i, :]

- B[k, :] * B[i, k]

(for j in range(n*2):

B[i, j] = B[i, j] - B[k, j] * b;)

return B.

n = 100

p = 5

X = matrix(rnorm(n*p), n_row=n)

beta = matrix(1:p, n_row=p)

Y = X %*% beta + rnorm(n)

lm(Y ~ X)

???

Z = cbind(rep(1, n), X, Y)

A = t(Z) %*% Z

S = myGaussJordan (A + p*I)

beta =

RSS =

V =

Lecture 4
10/10/2017

Gauss-Jordan

$$\left[\begin{array}{c|c} A_{n \times p} & I \end{array} \right] \xrightarrow[A^{-1}]{GJ[1 \dots n]} \left[\begin{array}{c|c} I & A^{-1} \end{array} \right] \quad A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}_{n \times n}$$

$$\left(\begin{array}{cc|cc} A_{11} & A_{12} & I_1 & 0 \\ A_{21} & A_{22} & 0 & I_2 \end{array} \right) \longrightarrow \left(\begin{array}{c|c|c|c} I & A_{11}^{-1}A_{12} & A_{11}^{-1} & 0 \\ 0 & A_{22} - A_{21}A_{11}^{-1}A_{12} & -A_{21}A_{11}^{-1} & I_2 \end{array} \right)$$

$$\begin{pmatrix} A_{11}^{-1} & 0 \\ -A_{21}A_{11}^{-1} & I \end{pmatrix} A = \begin{pmatrix} I & A_{11}^{-1}A_{12} \\ 0 & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{pmatrix}$$

↓

GJ[A₁₁]

$$a_{21} - 1 \cdot a_{21}$$

$$a_{31} - 1 \cdot a_{31}$$

Sweep Operator

$$\left[\begin{array}{cccc} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{array} \right] \xrightarrow{\text{Swp}[1]} \left[\begin{array}{cccc} 1 & \frac{a_{12}}{a_{11}} & \dots & \frac{a_m}{a_{11}} \\ 0 & a_{22} - \frac{a_{11} - a_{12}}{a_{11}} & \dots & \frac{a_m - a_{11}}{a_{11}} \\ \vdots & & & \\ 0 & & & 1 \end{array} \right]$$

$$\tilde{a}_{kk} = -\frac{1}{a_{kk}}$$

$$\tilde{a}_{kj} = \frac{a_{kj}}{a_{kk}} \quad j \neq k$$

$$\tilde{a}_{ik} = \frac{a_{ik}}{a_{kk}} \quad i \neq k$$

$$\tilde{a}_{ij} = a_{ij} - \frac{a_{ik}a_{kj}}{a_{kk}} \quad i, j \neq k$$

Swp[K]

$$\left[\begin{array}{c} \tilde{a}_{ij} \end{array} \right]$$



$$\left(\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \end{array} \right) \xrightarrow{\text{Swp}[1 \dots m]} \left(\begin{array}{cc} -A_{11}^{-1} & A_{11}^{-1}A_{12} \\ A_{11}A_{11}^{-1} & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{array} \right)$$

Swp[A₁]

$$\begin{pmatrix} X^T X & X^T Y \\ Y^T X & Y^T Y \end{pmatrix} \xrightarrow{\text{sweep}[X^T X]} \begin{pmatrix} -(X^T X)^{-1} & (X^T X)^{-1} X^T Y \\ Y^T X (X^T X)^{-1} & Y^T Y - Y^T X (X^T X)^{-1} X^T Y \end{pmatrix}$$

$$= \begin{pmatrix} -\frac{\text{var}(\hat{\beta})}{\sigma} & \hat{\beta} \\ \hat{\beta}^T & RSS \end{pmatrix}$$

mysweep \leftarrow function(A, m)

{

n \leftarrow dim(A)[1]

for (k in 1:m)

{

for (i in 1:n)

for (j in 1:m) → (indent)

if (i != k, & j != k)

$A[i,j] \leftarrow A[i,j] - A[i,k] * A[k,j] / A[k,k]$

for (i in 1:n)

if (i != k)

$A[i,k] \leftarrow A[i,k] / A[k,k]$

for (j in 1:n)

if (j != k)

$A[k,j] \leftarrow A[k,j] / A[k,k]$

$A[k,k] \leftarrow -1 / A[k,k]$

}

return (A)

Gaussian density:

$$f(x) = \frac{1}{(2\pi)^{\frac{p}{2}} \det(\Sigma)^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (x-\mu)^T \Sigma^{-1} (x-\mu) \right]$$

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \xrightarrow{\text{Swp}} [x_2 | x_1] \sim N(\beta x, \tilde{\Sigma}) \sim N(\Sigma_{22} \Sigma_{11}^{-1} x, \Sigma_{22} - \Sigma_{21} \Sigma_{11}^{-1} \Sigma_{12})$$

10/12/2017

Lecture 5

```
#include <Rcpp.h>
using namespace Rcpp;
```

```
// [[ Rcpp::export]] ... have to include before every functions.
double linearFunction1(double, x) {
```

```
// set aside memory for ourout y
double y = 0.0;
```

```
// compute output
y = x + 3;
```

```
// Return output
```

```
return y
```

```
}
```

```
// [[ Rcpp::export]]
int linearFunction2(int, x) {
```

```
// print statement in Rcpp
Rcout << "x is equal to" << x <<
```

Looping in

```
Rcpp // [R ... ]
int loopexample(int n) {
    int x = 0
    for
```

```
// [[Rcpp:: ]]  
double rbetaACCp(Function f, double x1, double x2) {  
    double output = 0.0;  
  
    // Create a list to store R function output.  
    List temporary;  
  
    // Call R function and store results into our list  
    temporary = f(1, x1, x2);  
  
    // Access the 0th element of that list, aka our  
    output = as<double>(temporary[0]);  
  
    return output;  
}
```

Vector // [[...]]
and Numericvector vectorManipulation1(Numericvector x,
Matrix

double sumy = sum(y);

C++ python
A=clone(B) ~ ACOPY(B)

$$\begin{array}{c|cc|c} & P \\ \hline n & X & Y \\ & x_1 \dots x_p \end{array}$$

$$Y = X\beta + \varepsilon$$

$$\hat{\beta} = \arg \min \|Y - X\beta\|^2$$

QR Decomposition

$$(X \quad Y)_{n \times p \quad n \times 1} \xrightarrow{Q} (R, Y^*)_{n \times p \quad n \times 1} = \left(\begin{pmatrix} R^* \\ 0 \end{pmatrix}_{n \times p}, \begin{pmatrix} Y_1^* \\ Y_2^* \end{pmatrix}_{n \times 1} \right)$$

$$\hat{\beta} = \arg \min \|Y - X\beta\|^2$$

$$= \|Y^* - R\beta\|^2$$

$$= \|Y_1^* - R_1^*\beta\|^2 + \|Y_2^*\|^2$$

$$\downarrow = 0$$

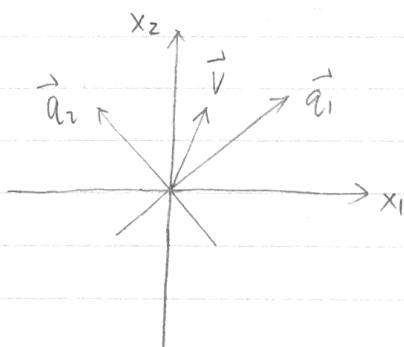
$$\hat{\beta} = R_1^{-1} Y_1^*, \quad RSS = \|Y_2^*\|^2$$



$$\begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$$

$$R_1^* \quad \beta = Y_1^*$$

$$Q_{n \times m} = (q_1 \ q_2 \ \dots \ q_n)$$



$$\vec{v} = u_1 \cdot \vec{q}_1 + u_2 \cdot \vec{q}_2 + \dots + u_n \cdot \vec{q}_n$$

$$= (\vec{q}_1 \dots \vec{q}_n) \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix} = Q^T u$$

$$u_1 = \langle \vec{v}, \vec{q}_1 \rangle = \vec{q}_1^T \vec{v}$$

$$u_2 = \langle \vec{v}, \vec{q}_2 \rangle = \vec{q}_2^T \vec{v} = \begin{pmatrix} \vec{q}_1^T \\ \vdots \\ \vec{q}_n^T \end{pmatrix} \vec{v} = Q^T \vec{v}$$

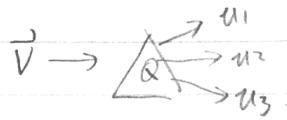
:

$$u_n = \langle \vec{v}, \vec{q}_n \rangle = \vec{q}_n^T \vec{v}$$

$$\vec{u} = Q^T \vec{v} \quad \text{analysis}$$

$$\vec{v} = Q \vec{u} \quad \text{synthesis}$$

$$QQ^T = Q^T Q = I$$



$$x \rightarrow x^*$$

$$(x_1, \dots, x_p) \rightarrow (x_1^*, \dots, x_p^*)$$

$$= \begin{pmatrix} x_1^* = \pm |x_1| \\ 0 & x_2^* \\ \vdots & \ddots & x_p^* \\ 0 & & & \end{pmatrix}$$

✓ OK

Householder reflection

$$\vec{u} = \frac{(x_1 - x_1^*)}{\|x_1 - x_1^*\|}$$

$$x_1^* = x_1 - 2 \langle \vec{x}_1, \vec{u} \rangle \cdot \vec{u}$$

$$x_j^* = x_j - 2 \langle \vec{x}_j, \vec{u} \rangle \cdot \vec{u} \quad j = 1, \dots, p$$

$$= (x_j - 2 \cdot \vec{u}, \vec{u}^T x_j)$$

$$H_1 = I - 2 \vec{u} \vec{u}^T$$

$$e_1 = \frac{x_1 - x_1^*}{\|x_1 - x_1^*\|}$$

Q1

$$\vec{x}_j \cdot \vec{u}$$

$$\langle \vec{x}_j, \vec{u} \rangle$$

// [Rcpp::depends(RcppArmadillo)]

include <RcppArmadillo.h>

using namespace aram;

using namespace Rcpp;

10/17/2017

Lecture 6.

Review:

QR decomposition: $[X | Y] \xrightarrow{Q} [R | Y]$

$$\begin{array}{|c|c|} \hline R & Y_1^* \\ \hline 0 & Y_2^* \\ \hline \end{array}$$

$$\hat{\beta} = R^{-1} Y_1^*$$

$$RSS = \|Y^*\|^2$$



$$\begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \rightarrow \begin{pmatrix} x_1^* & & & \\ 0 & & & \\ \vdots & & & \\ 0 & & & \end{pmatrix}$$

\vec{x}_1^* Householder reflection $\vec{y}_1 = \frac{\vec{x}_1 - \vec{x}_1^*}{\|\vec{x}_1 - \vec{x}_1^*\|} \rightarrow H_1$ $x_{11}^* = -\text{sign}(x_{11}) |x_{11}|$

$$\begin{aligned} \vec{x}_1^* &= \vec{x}_1 - 2 \langle \vec{x}_1, \vec{u} \rangle \vec{u} & \vec{x}_j^* &= \vec{x}_j - 2 \langle \vec{x}_j, \vec{u} \rangle \vec{u} \\ &= \vec{x}_1 - 2 \cdot \vec{u} \cdot \vec{u}^T \cdot \vec{x}_1 & &= H_1 \vec{x}_j \\ &= \underbrace{(I - 2 \vec{u} \cdot \vec{u}^T)}_H \vec{x}_1 \end{aligned}$$

$$X \xrightarrow[H_{n-1} \dots H_2 H_1]{Q^T} R \quad (X, I) \rightarrow (R, Q^T)$$

$$X = Q \cdot R$$

$$R = Q^T X$$

$$\begin{aligned} \vec{v} &= Q \vec{u} = (q_1, q_2, \dots, q_n) \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \\ &= u_1 \vec{q}_1 + \dots + u_n \vec{q}_n \end{aligned}$$

Eigen-decomposition:

$$\Sigma = Q \Lambda Q^T$$

orthogonal $\leftarrow \downarrow$
matrix diagonal matrix

$$\vec{u} = Q^T \vec{v} \quad u_i = \langle \vec{v}_i, \vec{q}_i \rangle$$

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

ordering $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$

$$\vec{V} = Q \vec{u}$$

$$\Sigma = Q \Lambda Q^T$$

$$\vec{u} = Q^T \vec{v}$$

$$\vec{w} = \Sigma \vec{v} = Q \Lambda Q^T \alpha \vec{u} = Q \begin{pmatrix} \vec{u} \\ \vec{z} \end{pmatrix}$$

$$\begin{matrix} \vec{w} = \Sigma \vec{v} \\ \downarrow \quad \downarrow \\ Q \vec{z} \quad Q \vec{u} \\ \vec{z} = \Lambda \vec{u} \end{matrix}$$

$$= \begin{pmatrix} \lambda_1 u_1 \\ \lambda_2 u_2 \\ \vdots \\ \lambda_n u_n \end{pmatrix}$$

$$\begin{matrix} \vec{w} = \Sigma^t \vec{v} \\ \downarrow \quad \downarrow \\ \vec{z} = \Lambda^t \vec{u} \\ = \begin{pmatrix} \lambda_1^t u_1 \\ \lambda_2^t u_2 \\ \vdots \\ \lambda_n^t u_n \end{pmatrix} \end{matrix}$$

dominant (λ_1^t)

Power method

$$\text{Start from } v \quad \tilde{v} = \frac{\vec{v}}{\|\vec{v}\|} = v = \sum \tilde{v}$$

$$v = \alpha \cdot R$$

Start from $v - n \times n$

$$\tilde{v} = \text{normalize}(v)$$

$$v = \sum \tilde{v} \quad \begin{pmatrix} 1 & 1 & \dots & 1 \\ v_1 & v_2 & \dots & v_n \end{pmatrix} = (q_1 \dots q_n)$$

$$\begin{pmatrix} R \\ \vdots \\ R \end{pmatrix}$$

10/19/2017

Lecture 7.

QR Decomposition

$$[X \ Y] \rightarrow [R \ Y^*] \rightarrow$$

```
Import numpy as np  
from scipy import linalg
```

```
def qr(A)
```

```
n, m = A.shape
```

```
R = A.copy()
```

```
Q = np.eye(n)
```

```
for k in range(m-1):
```

```
    X = np.zeros((n, 1))
```

```
    X[k:, 0] = R[k:, k]
```

```
V = X
```

```
V[k] = X[k] + np.sign(X[k, 0]) * np.linalg.norm(X)
```

```
S = np.linalg.norm(V)
```

```
U = V/S
```

```
R -= -2 * np.dot(U, np.dot(U.T, R))
```

```
Q -= 2 * np.dot(U, np.dot(U.T, Q))
```

```
Q = Q.T
```

```
return Q, R.
```

$$n = 100$$

$$p = 5$$

$X = np.random.random_sample((n, p))$

$\beta = np.array(range(1, p+1))$

$Y = np.dot(X, \beta) + np.random.standard_normal(n)$

$Z = np.hstack((np.ones(n).reshape(n, 1), X, Y.reshape((n, 1))))$

trace on R.

$$-, R = qr(Z)$$

$$R1 = R[:, :p+1]$$

$$Y1 = R[:, :p+1]$$

$$\beta = np.linalg.solve(R1, Y1)$$

print beta

$$X_{n \times n} = QR$$

$$\begin{pmatrix} | & | & | & \dots & | \\ x_1 & x_2 & \dots & x_n \end{pmatrix} = (q_1 \ q_2 \ \dots \ q_n) \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ 0 & r_{22} & & \\ 0 & 0 & \ddots & \\ & & & r_{nn} \end{pmatrix}$$

Gram-Schmidt



$$x_1 = r_{11}q_1$$

$$x_2 = r_{12}q_1 + r_{22}q_2$$

$$q_1 = \frac{x_1}{\|x_1\|} \quad r_{11} = \|x_1\|$$

$$r_{12} = x_2 - \langle x_2, q_1 \rangle q_1$$

$$q_2 = \frac{e}{\|e\|}$$

$$r_{22} = \|e\|$$

Powell Method

$$\Sigma = Q \Lambda Q^T$$

Power method Python code:

```
def eigen_qr(A)
```

T = 1000

A_copy = A.copy()

r, c = A_copy.shape

v = np.random.random_sample((r, 1))

for i in range(T):

Q, R = qr(v)

v = np.dot(A_copy, Q)

return R.diagonal(), Q

matrix calculation

packaging $Q = (q_1, q_2, \dots, q_n)$

$$Q \cdot u = (q_1, q_2, \dots, q_n) \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \sum q_i u_i$$

representation / view point

$$v = Q \cdot u$$

$$\Sigma = Q \Lambda Q^T$$

$$w = \Sigma v$$

$$u = Q^T v$$

$$\Sigma Q = Q \Lambda$$

$$= \sum Q \cdot u$$

$$\Sigma \quad v \xrightarrow{\Sigma} w$$

$$= Q \Lambda Q^T Q \cdot u$$

$$\begin{array}{ccc} \downarrow Q & & \downarrow Q \\ u & \xrightarrow{\Lambda} & z \end{array}$$

$$= Q \Lambda u$$

$$v \xrightarrow{\Sigma^T} w$$

$$= Q z$$

$$\begin{array}{ccc} \downarrow Q & & \downarrow Q \\ w & \xrightarrow{\Lambda t} & z \end{array} = \begin{pmatrix} \lambda_1^t u_1 \\ \vdots \\ \lambda_n^t u_n \end{pmatrix} \rightarrow$$

$$V \xrightarrow{E^t} W$$

get q_1

$$u \xrightarrow{Q} z = \begin{pmatrix} \lambda_1 u_1 \\ \vdots \\ \lambda_n u_n \end{pmatrix} \xrightarrow{\text{normalize}} \begin{pmatrix} 1 \\ \vdots \\ 0 \end{pmatrix}$$

could randomly choose a vector v which is perpendicular to q_1 .

$$V \perp q_1 \xrightarrow{E^t} W$$

get q_2

$$u = \begin{pmatrix} 0 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \xrightarrow{Q} z = \begin{pmatrix} 0 \\ \lambda_2 u_2 \\ \vdots \\ \lambda_n u_n \end{pmatrix} \xrightarrow{\text{normalize}} \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = q_2$$

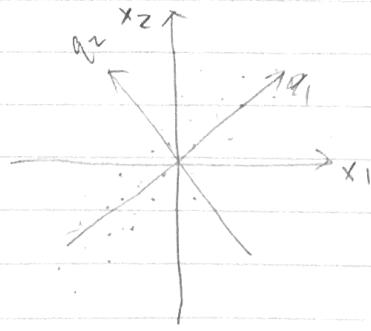
$$V \perp q_1 + q_2 \xrightarrow{E^t} q_3$$

get q_3

$$u = \begin{pmatrix} 0 \\ 0 \\ u_3 \\ \vdots \\ 0 \end{pmatrix} \xrightarrow{Q} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\sum Q = Q \Lambda Q^T Q = \underline{Q \Lambda} \\ QR$$

ex:	1 2	Y	$Z_1 Z_2$	row presentation
1	$X_{11} X_{12}$		$Z_{11} Z_{12}$	
2	$Y_{21} Y_{22}$		$Z_{21} Z_{22}$	
:	:			
n	$X_{n1} X_{n2}$			



column representation:



10/24/2017

Lecture 8

Principle Component analysis

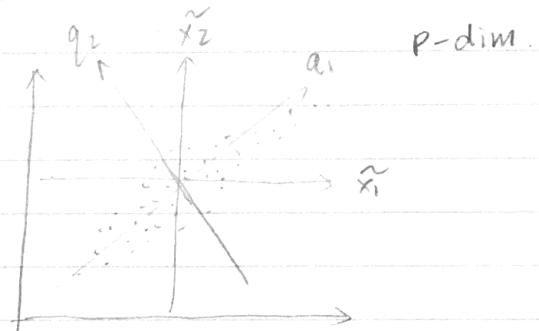
	1	2	\bar{x}	\dots	P	Y
1						
2						
:						
i						
:						
n						

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ip} \end{pmatrix} \sim f(x)$$

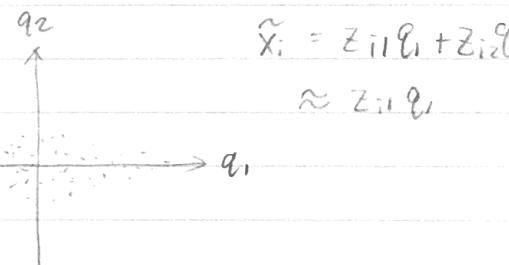
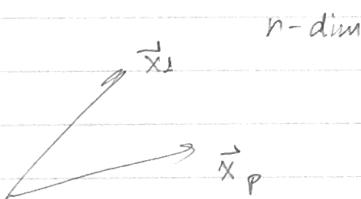
$p \times 1$

centerize

1	2	
$x_{11} - \bar{x}_1$	$x_{12} - \bar{x}_2$	
:	:	
$x_{n1} - \bar{x}_1$	$x_{n2} - \bar{x}_2$	



Column picture



$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad A \bar{x} = \frac{1}{n} \sum_{i=1}^n Ax_i$$

↓ ↓

$$E[x] \quad AE[x] = E[AX]$$

$$S = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T$$

$p \times 1 \quad 1 \times p$
 $\underbrace{\qquad\qquad}_{p \times p}$

$$\frac{1}{n} \sum_{i=1}^n (Ax_i - A\bar{x})(Ax_i - A\bar{x})^T = \frac{1}{n} ASAT$$

↓

$$\Sigma = E[(x-\mu)(x-\mu)^T]$$

$$\text{Var}(Ax) = A \text{Var}(x) A^T$$

$$\Sigma = \text{Var}(x) = Q \Lambda Q^T$$

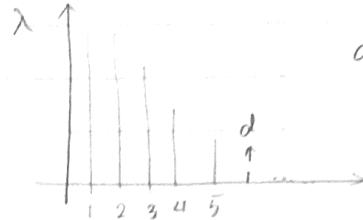
$$\text{Var}(Q^T x) = Q^T \Sigma Q = \Lambda$$

$$\begin{cases} Z = Q^T X \\ X \sim (\mu, \Sigma) \\ Y = QZ \end{cases} \quad Z \sim (\Lambda)$$

principle component.

$$X = b_1 q_1 + b_2 q_2 + \dots + b_d q_d \approx b_1 q_1 + \dots + b_d q_d$$

$$E[X] = E[Z] = 0$$



dimension reduction

X : = $Z_{11} \begin{matrix} \text{---} \\ q_1 \end{matrix} + Z_{12} \begin{matrix} \text{---} \\ q_2 \end{matrix} + \dots + Z_{1d} \begin{matrix} \text{---} \\ q_d \end{matrix}$

face image

$$\text{PCA: (1) } \Sigma \text{ for } S = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T$$

$$(2) \Sigma = Q \Lambda Q^T$$

(3) Cut off

	1 2	1 2
1	\tilde{x}_{11} \tilde{x}_{12}	z_{11} z_{12}
2	\tilde{x}_{21} \tilde{x}_{22}	z_{21} z_{22}
\vdots		
n	\tilde{x}_1 \tilde{x}_2	z_1 z_2

$\tilde{x}_i = \begin{pmatrix} x_i \\ \vdots \\ x_i \end{pmatrix}$ n-dim

$\| \tilde{x}_i \| = \sqrt{\sum_{j=1}^n x_{ij}^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T} = \sqrt{\text{Var}(x_i)}$

$\| z_i \| = \sqrt{\sum_{j=1}^d z_{ij}^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n z_i^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2} = \sqrt{\text{Var}(z_i)}$

$$\tilde{X} Q = Z$$

$$\| z_i \| = \sqrt{\text{Var}(z_i)}$$

$$\begin{aligned}
 \mathbf{x} &= \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \text{Var}(\mathbf{x}) = E[(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T] \\
 &= E\left[\left[\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}\right]\left[\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}\right]^T\right] \\
 &= E\left(\begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} (x_1 - \mu_1 \ x_2 - \mu_2)\right) = E\left[\begin{array}{cc} (x_1 - \mu_1)^2 & (x_1 - \mu_1)(x_2 - \mu_2) \\ (x_2 - \mu_2)(x_1 - \mu_1) & (x_2 - \mu_2)^2 \end{array}\right] \\
 &= \begin{pmatrix} \text{Var}(x_1) & \text{Cov}(x_1 x_2) \\ \text{Cov}(x_1 x_2) & \text{Var}(x_2) \end{pmatrix}
 \end{aligned}$$

How to get $\bar{\mathbf{x}}$? (vector)

$$\frac{1}{n} \begin{pmatrix} 1 & \dots & 1 \end{pmatrix}_{n \times 1} \begin{pmatrix} x \end{pmatrix}_p^T = \bar{x} = \begin{pmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \vdots \\ \bar{x}_p \end{pmatrix}$$

Multivariate Calculus.

X	Y

$$\begin{aligned}
 R(\beta) &= \|Y - X\beta\|^2 \\
 \dot{R}(\beta) &= -2X^T(Y - X\beta)
 \end{aligned}$$

$$R'(\beta) = \sum_{i=1}^n (y_i - x_i \beta)^2 \quad \textcircled{3}$$

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

$$R'(\beta) = -2 \sum_{i=1}^n (y_i - x_i^T \beta) x_i \quad \textcircled{2}$$

$$\begin{pmatrix} R'(\beta) \end{pmatrix} = -2 \sum_{i=1}^n \left(y_i - \sum_{j=1}^p x_{ij} \beta_j \right) \begin{pmatrix} x_{ij} \\ \vdots \\ x_{ip} \end{pmatrix} \quad \textcircled{1}$$

2nd derivative

$$R(\beta) = \frac{1}{2} \|Y - X\beta\|^2$$

$$y_{i|x_i} = f(x_{n|x_i})$$

$$\frac{\partial R}{\partial \beta^T} = \frac{\partial R}{\partial e^T} \cdot \frac{\partial e}{\partial \beta^T} = -2e^T X$$

$$\frac{\partial^2 y}{\partial x \partial x^T} = \left(\frac{\partial^2 y}{\partial x_i \partial x_j} \right)_{n \times n} = \frac{\partial z}{\partial x^T} \quad n \times n$$

$$R'(\beta) = \frac{\partial R}{\partial \beta} = -2X^T(Y - X\beta)$$

$$R''(\beta) = \frac{\partial^2 z}{\partial \beta^T} = 2X^T X \geq 0$$

$A \geq 0$ positive definite

$$a^T A a \geq 0$$

$$\Sigma = \text{Var}(x) \geq 0$$

$$a^T \Sigma a = \text{Var}(a^T x) \geq 0 \quad j \geq 0.$$

10/26/2017
Lecture 9.

Orthogonal matrix Q new viewpoint: (q_1, q_2, \dots, q_n)

$$\begin{array}{ccc} u & \xrightarrow{\Sigma} & \Sigma u \\ \downarrow Q & & \downarrow Q \\ v & \xrightarrow{\Lambda} & \Lambda v \end{array}$$

$x_{px1} \sim N(0, \Sigma)$
 $\sim N(0, \Lambda)$

$$\begin{cases} u = Qv \\ v = Q^T u \end{cases} \quad \begin{cases} \Sigma = Q\Lambda Q^T \\ \Lambda = Q^T \Sigma \end{cases} \quad \begin{cases} x = Qz \\ z = Q^T x \end{cases} \quad \Sigma = E[(x-\mu)(x-\mu)^T] = Q\Lambda Q^T$$

$$|u|=1$$

$$\max x_i \quad u^* = q_i$$

$$\text{Var}_{px1}(u^T x) = u^T \Sigma u$$

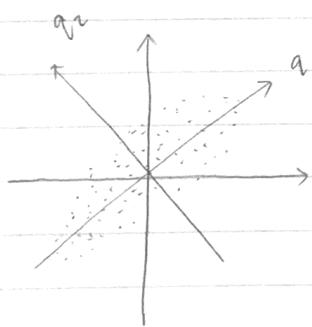
↑
positive-definite.

$$\text{Var}_{px1}(v^T z) = v^T \Lambda v$$

$$= \sum_{j=1}^p \lambda_j \cdot v_j^2$$

↓ max over v

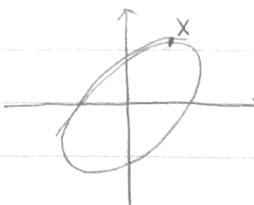
$$x_i, v^* = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$



$$x^T \Sigma^{-1} x = \text{const}$$

$$u \downarrow$$

$$z^T \Lambda^{-1} z = \sum_{j=1}^p \frac{z_j^2}{\lambda_j} = \text{const.}$$



Hessian matrix

	1	2	\dots	P	
1	x_{11}	x_{12}	\dots	x_{1P}	y_1
2					
\vdots					
i					
\vdots					
n					

$y_i \in \{0, 1\}$

understanding
classification

Linear Regression

$$\begin{cases} y_i \sim N(\mu_i, \sigma^2) \\ \mu_i = x_i^\top \beta \end{cases}$$

Ligistic regression

$$y_i \sim \text{Bernoulli}(p_i)$$

$$p_i = \frac{e^{x_i^\top \beta}}{1 + e^{x_i^\top \beta}}$$

$$P(y_i = 1 | x_i) = p_i$$

$$(\beta^{(0)} = \beta, \beta^{(1)} = 0)$$

score $x_i^\top \beta \begin{cases} \rightarrow \infty & p_i = 1 \\ \rightarrow 0 & p_i = \frac{1}{2} \\ \rightarrow -\infty & p_i = 0 \end{cases}$

$$y_i \in \{1, 2, \dots, K\}$$

$$P(y_i = k) = p_{ik} \quad \sum_{k=1}^K p_{ik} = 1. \quad p_{ik} = \frac{e^{x_i^\top \beta^{(k)}}}{\sum_{k=1}^K e^{x_i^\top \beta^{(k)}}}$$

max likelihood — most plausible expectation.

$$L(\beta) = \prod_{i=1}^n P(y_i | x_i, \beta) = \prod_{i=1}^n \frac{e^{y_i x_i^\top \beta}}{1 + e^{x_i^\top \beta}}$$

$$P(y_i | x_i, \beta) = \begin{cases} p_i = \frac{e^{x_i^\top \beta}}{1 + e^{x_i^\top \beta}} & y_i = 1 \\ 1 - p_i = \frac{1}{1 + e^{x_i^\top \beta}} & y_i = 0 \end{cases} = \frac{e^{y_i x_i^\top \beta}}{1 + e^{x_i^\top \beta}}$$

$$\ell(\beta) = \log L(\beta) = \sum_{i=1}^n \left[y_i x_i^\top \beta - \log(1 + e^{x_i^\top \beta}) \right] \quad \left(\begin{array}{l} \text{Recall } Y = AX \\ \frac{\partial Y}{\partial X} = A \end{array} \right)$$

$$\ell'(\beta) = \sum_{i=1}^n \left[x_i \cdot y_i - \frac{e^{x_i^\top \beta}}{1 + e^{x_i^\top \beta}} \cdot x_i \right]$$

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^n x_i (y_i - p_i)$$

$$\frac{\partial \ell}{\partial \beta \cdot \partial \beta^T} = - \sum_{i=1}^n x_i x_i^T$$

$\frac{e^{x_i^T \beta}}{(1+e^{x_i^T \beta})^2}$

$$= p_i(1-p_i) \leq 0.$$

$$* \frac{\partial}{\partial \beta^T} = (1 - \frac{1}{1+e^{x_i^T \beta}})$$

$$= \frac{e^{x_i^T \beta}}{(1+e^{x_i^T \beta})^2} x_i^T$$

\rightarrow concave

obtain one max point.

Taylor expansion.

$$\ell(\beta) = \ell(\beta_t) + \langle \ell'(\beta), \beta - \beta_t \rangle + \frac{1}{2} (\beta - \beta_t)^T \frac{\partial^2 \ell}{\partial \beta \partial \beta^T} \Big|_{\beta_t} (\beta - \beta_t)$$

↓
max

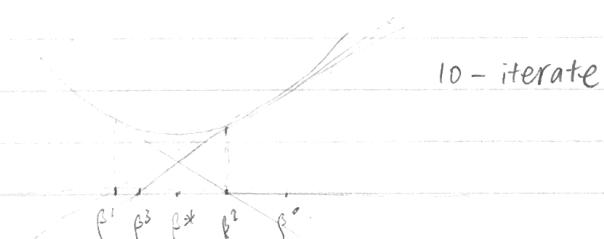
$$\frac{\partial \ell}{\partial \beta} = \ell'(\beta_t) + \ell''(\beta_t)(\beta - \beta_t) = 0 \Rightarrow \beta_{t+1} = \beta_t - \ell''(\beta_t)^{-1} \ell'(\beta_t)$$

Newton
- Raphson

$$\ell''(\beta_t)(\beta - \beta_t) = -\ell'(\beta_t)$$

Newton - Method

$$\beta - \beta_t = -[\ell''(\beta_t)]^{-1} \ell'(\beta_t)$$



10 - iterate.

more detailed:

$$\beta_{t+1} = \beta_t + \sum_{i=1}^n p_i(1-p_i)x_i x_i^T \sum_{i=1}^n x_i(y_i - p_i)$$

let $w_i = p_i(1-p_i)$

$$= \left(\sum_{i=1}^n w_i x_i x_i^T \right)^{-1} \cdot \sum_{i=1}^n \underbrace{(w_i x_i (x_i^T \beta_t + \frac{y_i - p_i}{w_i}))}_{z_i}$$

$$= \left(\sum_{i=1}^n \tilde{x}_i \tilde{x}_i^T \right)^{-1} \left(\sum_{i=1}^n \tilde{x}_i \tilde{y}_i \right)$$

$$\tilde{x}_i = \sqrt{w_i} x_i$$

$$\tilde{y}_i = \sqrt{w_i} z_i$$

Iterate reweighted least squares

$$\beta_t \rightarrow y = x^T \beta_t \rightarrow p_i = \frac{1}{1 + e^{g_i}} \rightarrow w_i = p_i(1 - p_i)$$
$$\beta_{t+1} \leftarrow \text{linear Regression} \leftarrow \begin{aligned} \tilde{x} &= \sqrt{n} x_i \\ \tilde{y} &= \sqrt{n} z_i \end{aligned} \quad \leftarrow z_i = y_i + \frac{y_i - p_i}{w_i}$$

(QR)

10/31/2017

Logistic Regression

Lecture 10

	P
n	x_i^T
	y_i

generalized linear model (GLM).

$$\begin{cases} y_i \sim P(y_i | \theta_i) \\ \theta_i = f(x_i^T \beta) \end{cases}$$

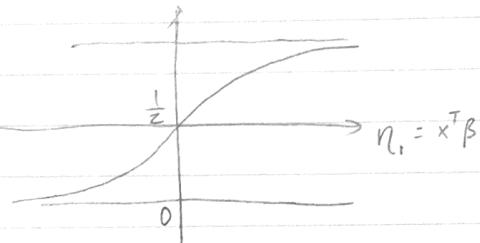
one dim η_i score.

$$y_i \sim \text{Bernoulli}(\beta_i)$$

$$\theta_i = F^{(x_i)}_{p \text{ dim}}$$

$$P_i = \frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} = \text{sigmoid}(x_i^T \beta)$$

$$= \frac{1}{1 + e^{x_i^T \beta}}$$

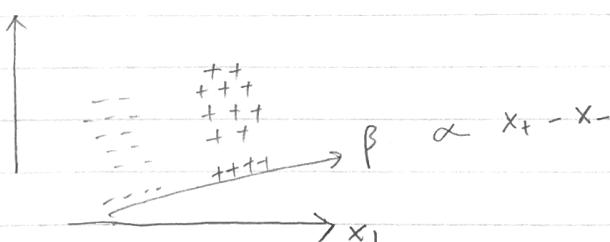
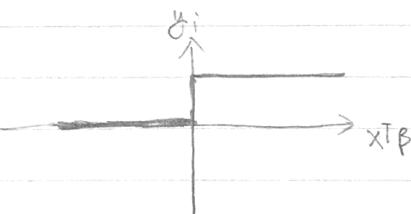


$$\text{logit}(P_i) = \log \frac{P_i}{1 - P_i} \Leftrightarrow y_i = x_i^T \beta$$

perceptron.

$$y_i = \text{sign}(x_i^T \beta) = \begin{cases} 1 & x_i^T \beta \geq 0 \\ 0 & \text{o.w.} \end{cases}$$

$$\text{or } y_i = \begin{cases} +1 \\ -1 \end{cases}$$



$$\text{what if } y = \begin{cases} +1 \\ -1 \end{cases} \quad P(y_i = 1 | x_i, \beta) = \frac{1}{1 + e^{-x_i^T \beta}} = \beta(y_i | x_i, \beta) = \frac{1}{1 + e^{-y_i x_i^T \beta}}$$

maximum Likelihood.

$$L(\beta) = \text{Prob}(Y|X, \beta)$$

$$= \prod_{i=1}^n P(Y_i|x_i, \beta)$$

$$= \prod_{i=1}^n P_i^{y_i} (1-P_i)^{1-y_i}$$

$y_i=1, P_i$

$y_i=0, 1-P_i$

$$= \prod_{i=1}^n \left(\frac{e^{x_i^\top \beta}}{1+e^{x_i^\top \beta}} \right) \left(\frac{1}{1+e^{x_i^\top \beta}} \right)^{1-y_i}$$

$$= \prod_{i=1}^n \frac{e^{x_i^\top \beta}}{1+e^{x_i^\top \beta}} \rightarrow P(Y_i|x_i, \beta)$$

what if $y = \begin{cases} +1 \\ -1 \end{cases}$ $P(Y_i=1|x_i, \beta) = \frac{1}{1+e^{-x_i^\top \beta}}$

$$P(Y_i=-1|x_i, \beta) = \frac{1}{1+e^{x_i^\top \beta}}$$

$$P(Y_i|x_i, \beta) = \frac{1}{1+e^{-x_i^\top \beta}}$$

$$\ell(\beta) = \text{Log } L(\beta) = \sum_{i=1}^n [y_i x_i^\top \beta - \log(1+e^{x_i^\top \beta})]$$

$$\ell'(\beta) = \sum_{i=1}^n y_i x_i - \underbrace{\left[\frac{e^{x_i^\top \beta}}{1+e^{x_i^\top \beta}} \right] \cdot x_i}_{\hat{P}_i} = \sum_{i=1}^n x_i(y_i - \hat{P}_i).$$

Learning rule: $\beta_{t+1} = \beta_t + \gamma_t \ell'(\beta_t)$ error

$$= \beta_t + \gamma_t \sum_{i=1}^n x_i(y_i - \hat{P}_i) \rightarrow \text{Version 1}$$

β_t

$$y_i = \begin{cases} +1 \\ -1 \end{cases} \quad \frac{\partial}{\partial \beta} \log(1+e^{-y_i x_i^\top \beta})$$

$$= -\frac{e^{-y_i x_i^\top \beta}}{1+e^{-y_i x_i^\top \beta}} y_i x_i = -\frac{1}{1+e^{y_i x_i^\top \beta}} y_i x_i \rightarrow \text{Version 2}$$

Stochastic algorithm:

randomly select i $\beta_{t+1} = \beta_t + r_i x_i (y_i - p_i)$

learn from error r_i decrease.

re alignment:

$$x_i y_i = t \quad p_i \approx 0$$

$$\beta_{t+1} = \beta_t + r_t \cdot x_i$$

$$y_i = -p_i \approx 1$$

$$\beta_{t+1} = \beta_t - r_t x_i$$

Perceptron algorithm:

$\beta_0 = 0$. randomly select i

$$\beta_{t+1} = \beta_t + \begin{cases} x_i & \text{if } y_i = 1 \quad x_i^\top \beta < 0 \\ -x_i & \quad \quad \quad = -1 \quad x_i^\top \beta > 0 \end{cases}$$

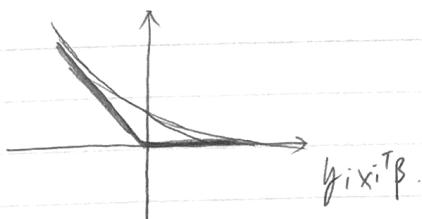
$$= \beta_0 + \mathbb{1}(y_i \neq \text{sign}(x_i^\top \beta)) y_i x_i \rightarrow \text{Version 3}$$

$$\text{Loss function } \ell(\beta) = \sum_{i=1}^n \max(0, -y_i x_i^\top \beta)$$

$$\frac{\partial}{\partial \beta} \log(1 + e^{-y_i x_i^\top \beta})$$

$$y_i x_i^\top \beta \rightarrow \infty$$

$$-y_i x_i^\top \beta$$



11/17/2017

Logistic regression.

Linear structure $x_i^\top \beta$

Lecture 12

non-linear link sigmoid (r): $\frac{1}{1+e^{-r}}$



linear mode!

$$R(\beta) = \|Y - X\beta\|^2 \Rightarrow \hat{\beta} = (X^\top X)^{-1} X^\top Y$$

generalized linear model

$\ell(\beta)$ = iterated reweighted LS.

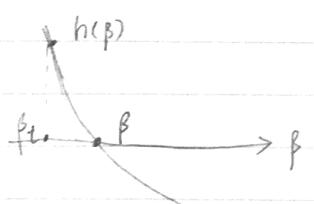
gradient ascent:

$$\beta_{t+1} = \beta_t + \gamma_t \ell'(\beta_t).$$

Newton Method

1-dim

$$h(\beta) = 0$$



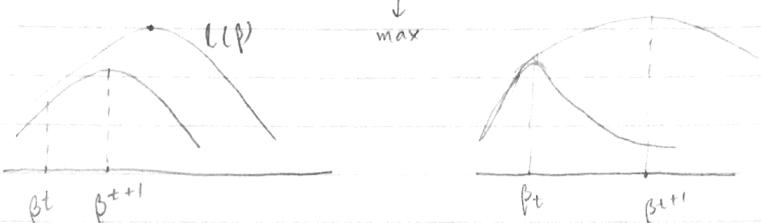
$$h(\beta) = h(\beta_t) + h'(\beta_t)(\beta - \beta_t)$$

$$\beta - \beta_t = - \frac{h(\beta_t)}{h'(\beta_t)}$$

$$\beta_{t+1} = \beta_t - \frac{h(\beta_t)}{h'(\beta_t)}$$

[2-dim]

$$l(\beta) = l(\beta_t) + l'(\beta_t)(\beta - \beta_t) + \frac{1}{2} l''(\beta_t)(\beta - \beta_t)^2$$



$$\beta_{t+1} = \beta_t - \frac{l'(\beta_t)}{l''(\beta_t)}$$

$$P\text{-dim: } \beta_{t+1} = \beta_t - [l''(\beta_t)]^{-1} \cdot l'(\beta_t)_{p \times 1}$$

$$l(\beta) = l(\beta_t) + \langle l'(\beta_t), \beta - \beta_t \rangle + \frac{1}{2} (\beta - \beta_t)^T l''(\beta_t) (\beta - \beta_t).$$

$$\beta = \beta_t + \alpha \cdot \vec{u}$$

$$l(\beta) = l(\underbrace{\beta_t + \alpha \cdot \vec{u}}_{\beta}). = f(\alpha)$$

$$\alpha = \frac{\beta - \beta_t}{\|\beta - \beta_t\|}$$

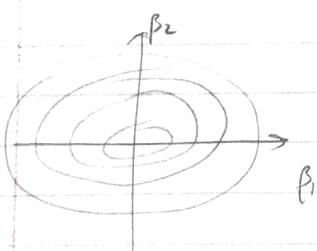
$$\begin{aligned} f(\alpha) &= f(0) + f'(0) \cdot \alpha + \frac{1}{2} f''(0) \cdot \alpha^2 \\ &= l(\beta_t) + \langle l'(\beta_t), \vec{u} \rangle \cdot \alpha + \vec{u}^T l''(\beta) \vec{u} \cdot \frac{1}{2} \alpha^2 \end{aligned}$$

$$\frac{df(\alpha)}{d\alpha} = \frac{\partial l}{\partial \alpha} = \frac{\partial l}{\partial \beta^T} \frac{\partial \beta}{\partial \alpha} = [l'(\beta)^T \cdot \vec{u}]$$

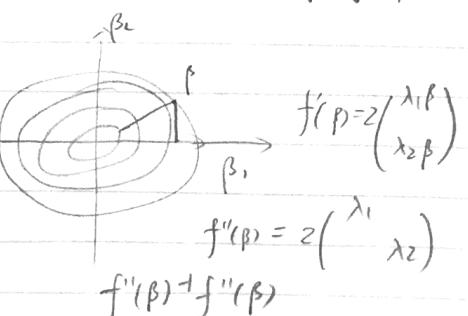
$$= \langle l'(\beta), \vec{u} \rangle.$$

$$l''(\beta) = Q \lambda Q^T$$

$$\text{Assume } l'' = \lambda, (\beta - \beta_t)^T l''(\beta_t) (\beta - \beta_t) = (\beta - \beta_t)^T \lambda (\beta - \beta_t) = \sum_j \lambda_j (\beta_j - \beta_{jt})^2$$



$$f(\beta) = \sum_{j=1}^p \lambda_j \beta_j^2 \quad (\lambda_1 < \lambda_2)$$



Regularize learning. avoid overfitting.

Accept for estimate error or uncertainty.

Random
fixed

frequencies. $f \sim N(\beta_{true}, V)$

Bayesian $\beta_{true} \sim N(\beta, V)$

fixed random

\rightarrow prior $\beta_{true} \sim P(\beta_{true})$

linear regression.

$$f(\beta) = \|Y - X\beta\|^2 + \lambda \|\beta\|_2^2 \rightarrow \text{ridge regression shrinkage}$$

$$+ \lambda \|\beta\|_1 \rightarrow \text{Lasso Selection}$$

$$\text{LS Assume } X^T X = I \quad Y = X\beta_{true} + \varepsilon = \sum_{j=1}^p x_j \beta_{true,j} + \varepsilon.$$

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

$$\hat{\beta}_{LS,j} = \langle Y_j, X_j \rangle = \beta_{true,j} + \langle \varepsilon, x_j \rangle = \beta_{true,j} + \delta_j \quad j=1 \dots p.$$

$$\text{Training error: } \|Y - X\hat{\beta}_{LS}\|^2 = \left\| \sum_j x_j \beta_{true,j} + \varepsilon - \sum_j x_j \hat{\beta}_{LS,j} \right\|^2.$$

$$= \|\varepsilon - \sum_j x_j \delta_j\|^2$$

$$= \left\| \sum_{j=p+1}^n x_j \delta_j \right\|^2 = \sum_{j=p+1}^n \delta_j^2 \quad \varepsilon = x_1 \delta_1 + x_2 \delta_2 + \dots + x_p \delta_p + x_{p+1} \delta_{p+1} + \dots + x_n \delta_n.$$

$$E(\text{training error}) = (n-p)\delta^2$$

$$E(|\varepsilon|^2) = n\delta^2$$

we can do better than the true value.

$$\begin{aligned} \text{Testing error: } \|Y - X\hat{\beta}_{LS}\|^2 &= \|X\beta_{true} + \tilde{\varepsilon} - X\hat{\beta}_{LS}\|^2 = \|\tilde{\varepsilon} - \sum_{j=1}^p x_j \delta_j\|^2 \\ &= \|\tilde{\varepsilon}\|^2 + \left\| \sum_{j=1}^p x_j \delta_j \right\|^2 - 2 \langle \tilde{\varepsilon}, \sum_{j=1}^p x_j \delta_j \rangle \\ &= n\delta^2 + p\delta^2 - 0 = (n+p)\delta^2. \end{aligned}$$

That's why we need to regularization:

How?

(1) Hypothesis test. $H_0: \beta_{true, j} = 0$

$H_1: \beta_{true, j} \neq 0$

(2) more smooth b/w null and alternative.

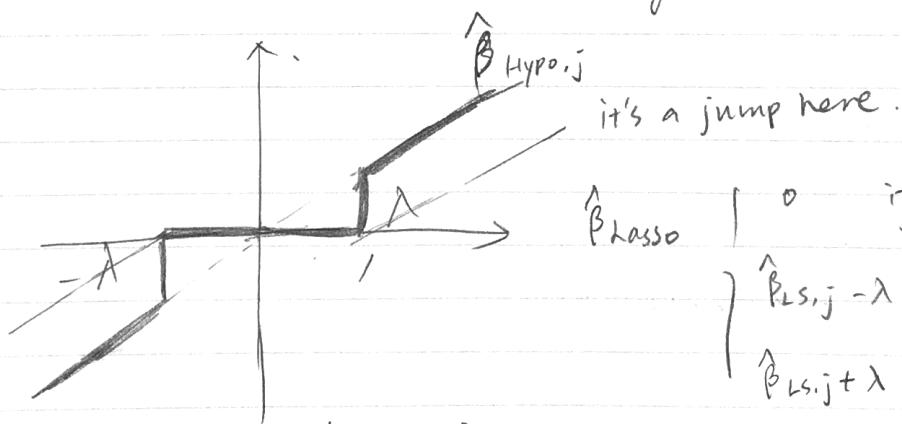
Fridge estimator is shrinkage.

$$\hat{\beta}_{shrinkage, j} = \frac{\hat{\beta}_{LS, j}}{1 + \lambda} \quad \rightarrow \lambda > 0.$$

$$|Y - X\beta|^2 + \lambda |\beta|^2 \rightarrow \ell_2, \text{ wedge regression.}$$

(3) Still from (1)

Lasso estimator is soft thresholding



$$\beta_{Lasso} = \begin{cases} 0 & \text{if } |\hat{\beta}_{LS, j}| \leq \lambda \\ \hat{\beta}_{LS, j} - \lambda & \hat{\beta}_{LS, j} > \lambda \\ \hat{\beta}_{LS, j} + \lambda & \hat{\beta}_{LS, j} < -\lambda. \end{cases}$$

$$\frac{1}{2} |Y - X\beta|^2 + \lambda |\beta|_{L1}$$

$$\hat{\beta}_{j, Lasso, \lambda} = \text{sign}(\hat{\beta}_j) \max(0, |\hat{\beta}_j| - \lambda).$$

(4) Bayesian prior.

$$\beta_{\text{true}, j} \sim N(0, \tau^2)$$

$$\beta_{\text{true}, j} \sim 99\% \delta_0 + 1\% N(0, r^2)$$

→ large

Spike/slab prior → Selection

Mallow CP

$$|Y - X\beta|^2 + 2\|\beta\|^2 = |Y - X\beta|^2 + 2\|\beta\|_0$$

↪ # of non-zero elements.

relax ↓

non-convex.

$$\|\beta\|_1, \|\beta\|_0$$

Ridge Regression.

$$\ell(\beta) = |Y - X\beta|^2 + \lambda \|\beta\|^2$$

how to minimize?

$$\begin{aligned}\frac{\partial \ell}{\partial \beta^T} &= \frac{\partial \ell}{\partial e^T} \cdot \frac{\partial e}{\partial \beta^T} + \lambda 2\beta^T \\ &= 2e^T(-X) + 2\lambda \beta^T\end{aligned}$$

$$e = Y - X\beta$$

$$\ell'(\beta) = -2X^T(Y - X\beta) + 2\lambda\beta = 0$$

$$-X^TY + X^TX\beta + \lambda\beta = 0$$

$$X^TX\beta + \lambda\beta = X^TY$$

$$(X^TX + \lambda I)\beta = X^TY$$

$$\hat{\beta}_{\text{ridge}} = (X^TX + \lambda I)^{-1}X^TY$$

Code part: myRidge ← function(x, Y, lambda).

(type on computer).

how to do in QR?

$$l(\beta) = \|Y - X\beta\|^2 + \alpha + \sqrt{\lambda} \|I \cdot \beta\|^2 = \left\| \begin{pmatrix} Y \\ 0 \end{pmatrix} - \begin{pmatrix} X \\ I \end{pmatrix} \beta \right\|^2$$

$$\begin{array}{|c|c|} \hline & X \\ \hline P & R \\ \hline P \times P & 0 \\ \hline I & 0 \\ \hline \end{array}$$

11/14/2017

Lecture 15.

Oversetting.

	P	n big
X		p also big, sometimes $p \gg n$,
n	Y	high dimension statistics.

Ridge:

$$|Y - X\beta|^2 + \lambda |\beta|_{l_2}^2$$

Spline:

$$y = \beta_0 + \sum_{j=1}^p \beta_j \max(0, x - k_j)$$

\rightarrow change of slope

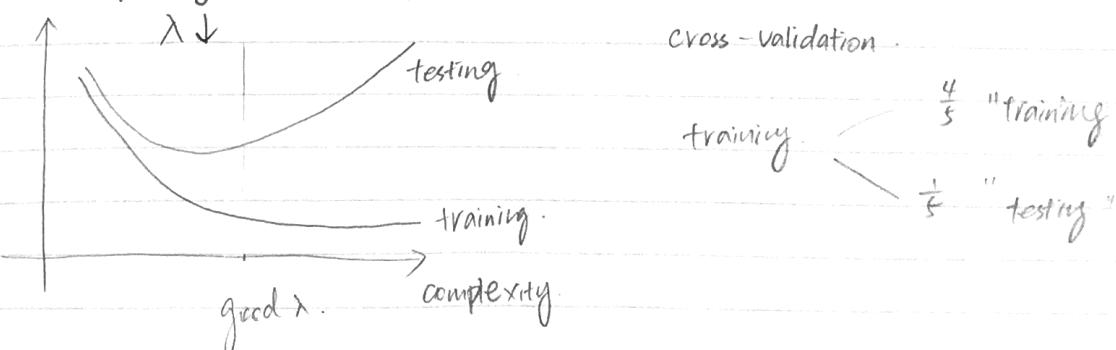
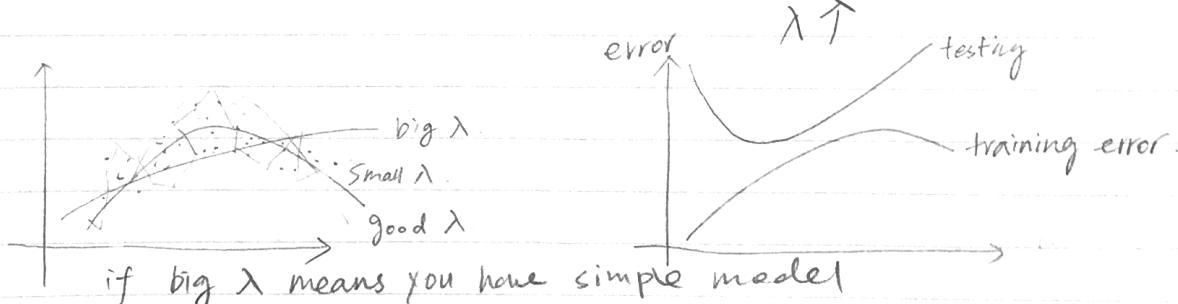
$$\lambda |\beta|_{l_2}^2 \rightarrow \text{Smooth.}$$

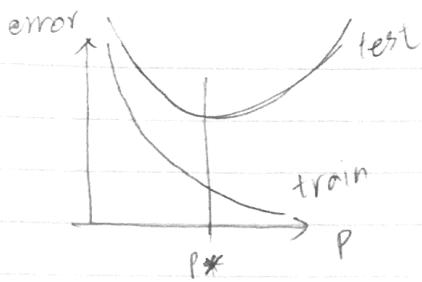
$$\hat{\beta} = (X^T X + \lambda I_p)^{-1} X^T Y \quad \text{special case: } X^T X = I_{p \times p}$$

$$\hat{\beta}_{j,\lambda} = \frac{\langle X_j, Y \rangle}{1+\lambda} = \frac{\hat{\beta}_{j,\text{es}}}{1+\lambda}$$

\downarrow
shrinkage estimator.

$$H_0: \beta_{j,\text{true}} = 0 \quad H_1: \beta_{j,\text{true}} \neq 0$$





we do not do least square but regularization.
 if you increase the complex of model
 model bias ↓
 overfitting ↓
 but $E(\hat{Y} - Y)^2 \uparrow$.

$$\text{Ridge: } \hat{\beta}_\lambda = (X^T X + \lambda I_p)^{-1} X^T Y$$

$$\text{training: } \|Y - X\hat{\beta}_\lambda\|^2 = \|f + \varepsilon - X\hat{\beta}_\lambda\|^2$$

$$\text{testing: } \|\hat{Y} - X\hat{\beta}_\lambda\|^2 = \|f + \hat{\varepsilon} - X\hat{\beta}_\lambda\|^2$$

$$\begin{aligned} E[\text{testing} - \text{training}] &= E(z<\varepsilon, X\hat{\beta}_\lambda> - z<\hat{\varepsilon}, X\hat{\beta}_\lambda>) \\ &= 2E<\varepsilon, X\hat{\beta}_\lambda> - 2<\hat{\varepsilon}, X\hat{\beta}_\lambda> \\ &= 2E<\varepsilon, X\hat{\beta}_\lambda> = 2\hat{\beta}_\lambda \delta^2 \quad \hat{\beta}_\lambda = \frac{E<\varepsilon, X\hat{\beta}_\lambda>}{\delta^2} \end{aligned}$$

$$\text{Recall LS. } E[\text{testing} - \text{training}] = 2p\delta^2$$

$$\text{Spline: 1000 knots. } p=1000 \quad \hat{\beta}_\lambda = 10.$$

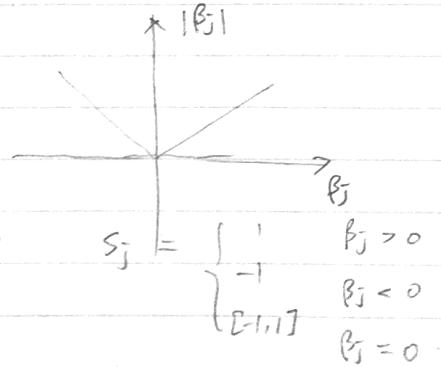
effective degree of freedom.

Lasso (Least absolute shrinkage selection operator)

$$R(\beta) = \frac{1}{2} \|Y - X\beta\|^2 + \lambda |\beta|_1$$

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

$$R'(\beta) = -X^T(Y - X\beta) + \lambda S = 0$$



minimize fix $\beta_j \neq 0$

$$\text{Solve } \beta_k, \quad R(\beta_k) = \frac{1}{2} \|Y - \sum_{j \neq k} x_j \beta_j - x_k \beta_k\|^2 + \lambda \sum_{j \neq k} |\beta_j| + \lambda |\beta_k|$$

R

$$R(\beta_k) = \frac{1}{2} \|R - X_k \beta_k\|^2 + \lambda |\beta_k| + \text{const.}$$

$$\begin{aligned} R'(\beta_k) &= -X_k^T R (R - X_k \beta_k) + \lambda S_k = 0 \\ &\equiv -\langle R, X_k \rangle + \|X_k\| \beta_k + \lambda S_k = 0 \end{aligned}$$

$$\|X_k\|^2 \cdot \beta_k = \langle R, X_k \rangle - \lambda S_k$$

$$\beta_k = \begin{cases} \frac{\langle R, X_k \rangle - \lambda}{\|X_k\|^2} & \text{if } \langle R, X_k \rangle > \lambda \\ 0 & \text{if } \langle R, X_k \rangle \in [-\lambda, \lambda] \\ \frac{\langle R, X_k \rangle + \lambda}{\|X_k\|^2} & \text{if } \langle R, X_k \rangle < -\lambda \end{cases}$$

$$\langle R, X_k \rangle \in [-\lambda, \lambda]$$

$$\hat{\beta}_k = \text{Sign}(\hat{\beta}_{R, LS}) \max(0, |\hat{\beta}_{R, LS}| - \frac{\lambda}{\|X_k\|^2})$$

= soft thresholding ($\hat{\beta}_{R, LS}$)

$$\hat{\beta}_{R, LS} = \frac{\langle R, X_k \rangle}{\|X_k\|^2}$$

$$\begin{cases} H_0 : \beta_{k, \text{true}} = 0 \\ H_1 : \beta_{k, \text{true}} \neq 0 \end{cases}$$

Coordinate descent:

$R(\beta)$ for ($k=1, \dots, p$) find β_k given $\beta_{j \neq k} = (\beta_j, j \neq k)$, minimize w.r.t. β_k

$$n = 50$$

$$p = 200$$

$$S = 10$$

$$\text{lambda_all} = (100:1)*10$$

$$L = \text{length}(\text{lambda_all})$$

$$X = \text{matrix}(n*p, nrow=n)$$

$$\beta_{\text{true}}[1:S] = 1:S$$

$$Y = X \% * \% \beta_{\text{true}} + rnorm(n)$$

from λ big to small

all 0 $\rightarrow \beta_k$ start to get non-zero value \rightarrow more value

$$Y = X \beta_{\text{true}} + \varepsilon.$$

$$\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

11/16/2017.

Lecture 16

Lasso:

$$[R, \beta] = \frac{1}{2} \|Y - X\beta\|^2 + \lambda \|\beta\|$$

$$R(\beta) = -X^T(Y - X\beta) + \lambda S(\beta) = 0$$

$$S = \begin{pmatrix} s_1 \\ \vdots \\ s_p \end{pmatrix} \quad S\beta = \begin{cases} 1 & \beta > 0 \\ -1 & \beta < 0 \\ (-1, 1) & \beta = 0 \end{cases}$$

Calculate

$$\beta(\beta_{1:t}, \beta_t, \beta_{p:t})$$

for ($t=1 \dots T$)

for ($K=1 \dots p$)

$$\min_k K(\beta_1, \dots, \beta_K)$$

Solution path:

λ : big $\beta = 0$.

gradually reduce λ .

$X_{n \times p} \quad Y_{n \times 1}$

length_all = (100 : 1) * 10.

L = Length (length_all).

R = Y

ss = rep(0, p).

for (j in 1:p) :

$$SS[j] = sum(X[, j]^2)$$

for (l in 1:L)

{ lambda = lambda_all[l].

for (t in 1:(T)) \rightarrow tiny different.

{ for (K in 1:p) :

{

$$\text{db} = sum[R * Y[, K]] / SS[K]$$

$$b = beta[K] + db$$

no intercept.

$$\rightarrow b = \text{Sign}[b] * \max[0, d[b] - \frac{\lambda}{\text{Jordon}} / \text{ss}[k]]$$

$$db = b - \beta_{\text{beta}}[k]$$

$$R = R - X[k, k] * db.$$

$$\beta_{\text{beta}}[R] = b.$$

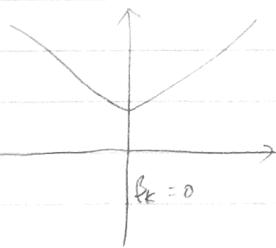
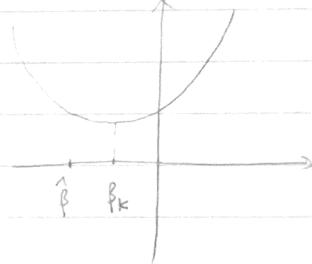
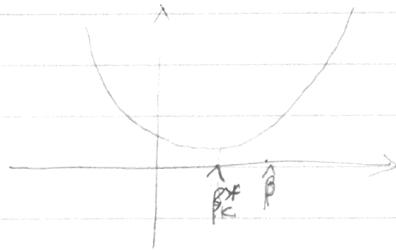
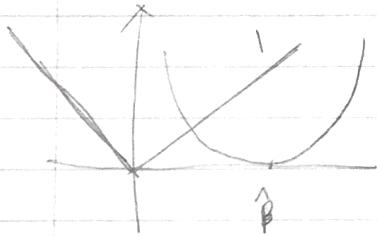
}

}

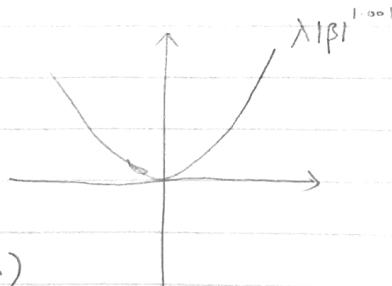
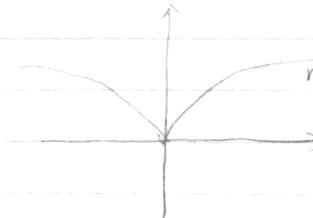
$$\beta_{\text{beta-all}}[t] = \beta_{\text{beta}}$$

}

$$\text{Solve } \frac{1}{2}(\beta - \hat{\beta})^2 + \lambda |\beta| = R(\beta).$$



big λ , slope $+\lambda \rightarrow -\lambda$.



$$\beta_t = \text{sign}(\hat{\beta}) \max(0, |\hat{\beta}| - \lambda)$$

$$\left\{ \begin{array}{l} H_0 : \beta = 0 \\ H_1 : \beta \neq 0 \end{array} \right.$$

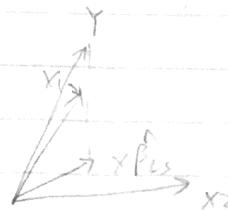
$$\frac{1}{2} |Y - X\beta|^2 + \lambda |\beta|_L$$

$$= \frac{1}{2} |Y - X\hat{\beta}_{LS}|^2 + \frac{1}{2} |\beta - \hat{\beta}|^2 + \lambda |\beta|_L,$$

$$X^T X = I_p$$

$$Y$$

$$\hat{\beta}_k = \text{sign}(\hat{\beta}_{LS,k}) \max(0, |\beta_{LS,k}| - \lambda)$$



$$\text{Lasso: } \hat{\beta}_{\text{Lasso}, k} = \text{soft-threshold}(\hat{\beta}_{\text{LS}, k}, \lambda)$$

$$\hat{\beta}_{\text{ridge}, k} = \text{shrinkage}(\hat{\beta}_{\text{LS}, k}, \lambda) = \frac{\hat{\beta}_{\text{LS}, k}}{1 + \lambda} \quad \text{when } x^T x = I$$

$E[\text{testing error}]$

$$= \|f - x\beta_{\text{best}}\|^2 + n\sigma^2 + E(\|\hat{\beta} - \beta_{\text{best}}\|^2)$$

model bias + noise
 \downarrow
 $E(\hat{\beta}) - \beta_{\text{best}}\|^2 + \text{Var}(\hat{\beta})$
 \downarrow
 estimation bias. variance

testing error:

$$\begin{aligned} \|f - x\hat{\beta}\|^2 &= \|f + \tilde{\epsilon} - x\hat{\beta}\|^2 \\ &= \|f - x\hat{\beta}\|^2 + \|\tilde{\epsilon}\|^2 \rightarrow n\sigma^2 \\ &= \|f - x\beta_{\text{best}}\|^2 + \|\hat{\beta} - \beta_{\text{best}}\|^2 \end{aligned}$$

sten estimator

$$\hat{\beta}_{\text{sten}} = \left(1 - \frac{(p-2)\delta^2}{\|\hat{\beta}_{\text{LS}}\|^2}\right) \hat{\beta}$$

$$= \hat{\beta}_{\text{ridge}, \lambda_{\text{sten}}}$$

$$\star E(\|\hat{\beta}_{\text{sten}} - \beta_{\text{best}}\|^2) \leq E(\|\hat{\beta}_{\text{LS}} - \beta_{\text{best}}\|^2)$$

Linear Regression

$$\begin{cases} \|Y - X\beta\|^2 + \lambda \|\beta\|_2^2 \rightarrow \text{Ridge} \rightarrow \text{Shrinkage: good for smooth} \\ \lambda \|\beta\|_1 \rightarrow \text{Lasso} \rightarrow \text{selection: good for sparsity} \end{cases}$$

Logistic Regression

$$\begin{cases} \text{Loss}(Y, X, \beta) + \lambda \|\beta\|_2^2 \rightarrow \text{hinge loss} \quad \sum_{i=1}^n \max(0, 1 - y_i x_i^T \beta) + \lambda \|\beta\|_2^2 \rightarrow \text{SVM} \\ \lambda \|\beta\|_1 \rightarrow \text{exponential loss} + \text{coordinate descent} \rightarrow \text{Adaboost.} \end{cases}$$

Neural network.

11/28/2017.

Lasso.

lecture 17

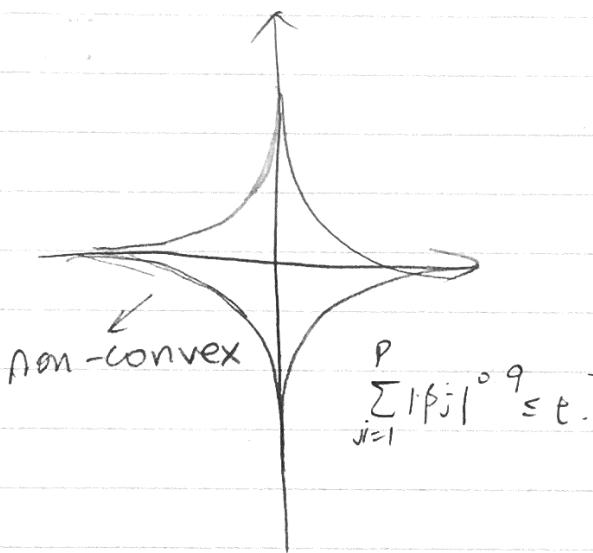
$$\frac{1}{2} \|Y - X\beta\|_F^2 + \lambda \|\beta\|_1$$

principle error: $\min \frac{1}{2} \|Y - X\beta\|_F^2$
 s. b. $\|\beta\|_1 \leq t$

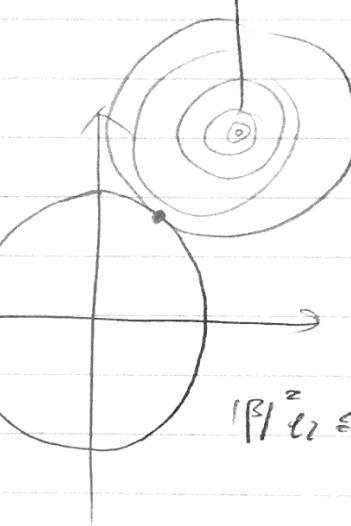
$$\frac{1}{2} \|Y - X\beta\|_F^2$$

$$\|\beta\|_1 \leq t$$

$$\|\beta\|_1 \leq t$$



$$\sum_{j=1}^p |\beta_j|^{\alpha_q} \leq t$$



$$\|\beta\|_2 \leq t$$

Forward regression. (matching pursuit).

Selection.

$$R(\beta) = \frac{1}{2} \|Y - X\beta\|_F^2$$

Gradient descent from $\beta=0$.

$$\frac{\partial R}{\partial \beta} = -X^T(Y - X\beta)$$

$$\frac{\partial R}{\partial \beta_j} = -\langle X_j, Y - X\beta \rangle$$

choose j over $\langle X_j, Y - X\beta \rangle$.

X_j Y

$$R = \frac{1}{2} \|Y - X\beta - x_j \delta \beta_j\|_F^2$$

$$\delta \beta_j = \frac{\langle Y - X\beta, x_j \rangle}{\|x_j\|^2}$$

Connection to Lasso.

$$\min \frac{1}{2} (\gamma - X\beta)^2$$

$$\Delta\beta_i = \varepsilon$$

$$|\beta|_{\ell_1} \leq t \rightarrow |\beta|_{\ell_1} \leq t + \Delta\beta_i$$

\downarrow
choose j

$\tilde{\ell}$ -boosting $\rightarrow \approx$ Lasso.

	j	p
i	$x_{ii} \quad x_{ij} \quad x_{ip}$	$y_{ie} \in \{+1, -1\}$
n		weak classifier.

$$y_i \approx \text{sign} \left\{ \sum_{j=1}^p x_{ij} \beta_j \right\}$$

committe.
 \downarrow
score.

$$\text{margin of } i = y_i \sum_{j=1}^p x_{ij} \beta_j$$

$$L(\beta) = \sum_{i=1}^n e^{-y_i \sum_{j=1}^p x_{ij} \beta_j}$$

$$\frac{\partial L}{\partial \beta_k} = \sum_{i=1}^n e^{-y_i \sum_{j=1}^p x_{ij} \beta_j} \frac{(-y_i x_{ik})}{w_i}$$

$$\textcircled{1} \quad \text{find } K \text{ to max } \sum_{i=1}^n w_i y_i x_{ik}$$

\downarrow
pay attention for examples. not classify well.

choose classifier K perform the best on (x_i, y_i, w_i) .

$$\begin{aligned} L(\Delta\beta_k) &= \sum_{i=1}^n e^{-y_i \left(\sum_{j=1}^{p-1} x_{ij} \beta_j + x_{ik} \Delta\beta_k \right)} \\ &= \sum_{i=1}^n w_i e^{-y_i x_{ik} \Delta\beta_k} \end{aligned}$$

\nearrow correct
 \searrow incorrect

$$= \sum_{i=1}^n w_i e^{-\Delta\beta_k} + \underbrace{\sum_{i=1}^n w_i e^{\Delta\beta_k}}_{\varepsilon}$$

$y_B = x_{ik}$

$$\sum_{i=1}^n w_i = 1$$

$$a+b \geq z \sqrt{ab} \quad \geq z \sqrt{\varepsilon(1-\varepsilon)}$$

$$(z \sqrt{ab})^2 \geq 0$$

$$(1-\varepsilon) e^{-\Delta\beta_k} = \varepsilon e^{\Delta\beta_k}$$

$$\Delta\beta_k = \frac{1}{2} \log \frac{1-\varepsilon}{\varepsilon}$$

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 | 144 | 145 | 146 | 147 | 148 | 149 | 150 | 151 | 152 | 153 | 154 | 155 | 156 | 157 | 158 | 159 | 160 | 161 | 162 | 163 | 164 | 165 | 166 | 167 | 168 | 169 | 170 | 171 | 172 | 173 | 174 | 175 | 176 | 177 | 178 | 179 | 180 | 181 | 182 | 183 | 184 | 185 | 186 | 187 | 188 | 189 | 190 | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 | 210 | 211 | 212 | 213 | 214 | 215 | 216 | 217 | 218 | 219 | 220 | 221 | 222 | 223 | 224 | 225 | 226 | 227 | 228 | 229 | 230 | 231 | 232 | 233 | 234 | 235 | 236 | 237 | 238 | 239 | 240 | 241 | 242 | 243 | 244 | 245 | 246 | 247 | 248 | 249 | 250 | 251 | 252 | 253 | 254 | 255 | 256 | 257 | 258 | 259 | 260 | 261 | 262 | 263 | 264 | 265 | 266 | 267 | 268 | 269 | 270 | 271 | 272 | 273 | 274 | 275 | 276 | 277 | 278 | 279 | 280 | 281 | 282 | 283 | 284 | 285 | 286 | 287 | 288 | 289 | 290 | 291 | 292 | 293 | 294 | 295 | 296 | 297 | 298 | 299 | 300 | 301 | 302 | 303 | 304 | 305 | 306 | 307 | 308 | 309 | 310 | 311 | 312 | 313 | 314 | 315 | 316 | 317 | 318 | 319 | 320 | 321 | 322 | 323 | 324 | 325 | 326 | 327 | 328 | 329 | 330 | 331 | 332 | 333 | 334 | 335 | 336 | 337 | 338 | 339 | 340 | 341 | 342 | 343 | 344 | 345 | 346 | 347 | 348 | 349 | 350 | 351 | 352 | 353 | 354 | 355 | 356 | 357 | 358 | 359 | 360 | 361 | 362 | 363 | 364 | 365 | 366 | 367 | 368 | 369 | 370 | 371 | 372 | 373 | 374 | 375 | 376 | 377 | 378 | 379 | 380 | 381 | 382 | 383 | 384 | 385 | 386 | 387 | 388 | 389 | 390 | 391 | 392 | 393 | 394 | 395 | 396 | 397 | 398 | 399 | 400 | 401 | 402 | 403 | 404 | 405 | 406 | 407 | 408 | 409 | 410 | 411 | 412 | 413 | 414 | 415 | 416 | 417 | 418 | 419 | 420 | 421 | 422 | 423 | 424 | 425 | 426 | 427 | 428 | 429 | 430 | 431 | 432 | 433 | 434 | 435 | 436 | 437 | 438 | 439 | 440 | 441 | 442 | 443 | 444 | 445 | 446 | 447 | 448 | 449 | 450 | 451 | 452 | 453 | 454 | 455 | 456 | 457 | 458 | 459 | 460 | 461 | 462 | 463 | 464 | 465 | 466 | 467 | 468 | 469 | 470 | 471 | 472 | 473 | 474 | 475 | 476 | 477 | 478 | 479 | 480 | 481 | 482 | 483 | 484 | 485 | 486 | 487 | 488 | 489 | 490 | 491 | 492 | 493 | 494 | 495 | 496 | 497 | 498 | 499 | 500 | 501 | 502 | 503 | 504 | 505 | 506 | 507 | 508 | 509 | 510 | 511 | 512 | 513 | 514 | 515 | 516 | 517 | 518 | 519 | 520 | 521 | 522 | 523 | 524 | 525 | 526 | 527 | 528 | 529 | 530 | 531 | 532 | 533 | 534 | 535 | 536 | 537 | 538 | 539 | 540 | 541 | 542 | 543 | 544 | 545 | 546 | 547 | 548 | 549 | 550 | 551 | 552 | 553 | 554 | 555 | 556 | 557 | 558 | 559 | 560 | 561 | 562 | 563 | 564 | 565 | 566 | 567 | 568 | 569 | 570 | 571 | 572 | 573 | 574 | 575 | 576 | 577 | 578 | 579 | 580 | 581 | 582 | 583 | 584 | 585 | 586 | 587 | 588 | 589 | 590 | 591 | 592 | 593 | 594 | 595 | 596 | 597 | 598 | 599 | 600 | 601 | 602 | 603 | 604 | 605 | 606 | 607 | 608 | 609 | 610 | 611 | 612 | 613 | 614 | 615 | 616 | 617 | 618 | 619 | 620 | 621 | 622 | 623 | 624 | 625 | 626 | 627 | 628 | 629 | 630 | 631 | 632 | 633 | 634 | 635 | 636 | 637 | 638 | 639 | 640 | 641 | 642 | 643 | 644 | 645 | 646 | 647 | 648 | 649 | 650 | 651 | 652 | 653 | 654 | 655 | 656 | 657 | 658 | 659 | 660 | 661 | 662 | 663 | 664 | 665 | 666 | 667 | 668 | 669 | 670 | 671 | 672 | 673 | 674 | 675 | 676 | 677 | 678 | 679 | 680 | 681 | 682 | 683 | 684 | 685 | 686 | 687 | 688 | 689 | 690 | 691 | 692 | 693 | 694 | 695 | 696 | 697 | 698 | 699 | 700 | 701 | 702 | 703 | 704 | 705 | 706 | 707 | 708 | 709 | 710 | 711 | 712 | 713 | 714 | 715 | 716 | 717 | 718 | 719 | 720 | 721 | 722 | 723 | 724 | 725 | 726 | 727 | 728 | 729 | 730 | 731 | 732 | 733 | 734 | 735 | 736 | 737 | 738 | 739 | 740 | 741 | 742 | 743 | 744 | 745 | 746 | 747 | 748 | 749 | 750 | 751 | 752 | 753 | 754 | 755 | 756 | 757 | 758 | 759 | 760 | 761 | 762 | 763 | 764 | 765 | 766 | 767 | 768 | 769 | 770 | 771 | 772 | 773 | 774 | 775 | 776 | 777 | 778 | 779 | 780 | 781 | 782 | 783 | 784 | 785 | 786 | 787 | 788 | 789 | 790 | 791 | 792 | 793 | 794 | 795 | 796 | 797 | 798 | 799 | 800 | 801 | 802 | 803 | 804 | 805 | 806 | 807 | 808 | 809 | 810 | 811 | 812 | 813 | 814 | 815 | 816 | 817 | 818 | 819 | 820 | 821 | 822 | 823 | 824 | 825 | 826 | 827 | 828 | 829 | 830 | 831 | 832 | 833 | 834 | 835 | 836 | 837 | 838 | 839 | 840 | 841 | 842 | 843 | 844 | 845 | 846 | 847 | 848 | 849 | 850 | 851 | 852 | 853 | 854 | 855 | 856 | 857 | 858 | 859 | 860 | 861 | 862 | 863 | 864 | 865 | 866 | 867 | 868 | 869 | 870 | 871 | 872 | 873 | 874 | 875 | 876 | 877 | 878 | 879 | 880 | 881 | 882 | 883 | 884 | 885 | 886 | 887 | 888 | 889 | 890 | 891 | 892 | 893 | 894 | 895 | 896 | 897 | 898 | 899 | 900 | 901 | 902 | 903 | 904 | 905 | 906 | 907 | 908 | 909 | 910 | 911 | 912 | 913 | 914 | 915 | 916 | 917 | 918 | 919 | 920 | 921 | 922 | 923 | 924 | 925 | 926 | 927 | 928 | 929 | 930 | 931 | 932 | 933 | 934 | 935 | 936 | 937 | 938 | 939 | 940 | 941 | 942 | 943 | 944 | 945 | 946 | 947 | 948 | 949 | 950 | 951 | 952 | 953 | 954 | 955 | 956 | 957 | 958 | 959 | 960 | 961 | 962 | 963 | 964 | 965 | 966 | 967 | 968 | 969 | 970 | 971 | 972 | 973 | 974 | 975 | 976 | 977 | 978 | 979 | 980 | 981 | 982 | 983 | 984 | 985 | 986 | 987 | 988 | 989 | 990 | 991 | 992 | 993 | 994 | 995 | 996 | 997 | 998 | 999 | 1000 | 1001 | 1002 | 1003 | 1004 | 1005 | 1006 | 1007 | 1008 | 1009 | 1010 | 1011 | 1012 | 1013 | 1014 | 1015 | 1016 | 1017 | 1018 | 1019 | 1020 | 1021 | 1022 | 1023 | 1024 | 1025 | 1026 | 1027 | 1028 | 1029 | 1030 | 1031 | 1032 | 1033 | 1034 | 1035 | 1036 | 1037 | 1038 | 1039 | 1040 | 1041 | 1042 | 1043 | 1044 | 1045 | 1046 | 1047 | 1048 | 1049 | 1050 | 1051 | 1052 | 1053 | 1054 | 1055 | 1056 | 1057 | 1058 | 1059 | 1060 | 1061 | 1062 | 1063 | 1064 | 1065 | 1066 | 1067 | 1068 | 1069 | 1070 | 1071 | 1072 | 1073 | 1074 | 1075 | 1076 | 1077 | 1078 | 1079 | 1080 | 1081 | 1082 | 1083 | 1084 | 1085 | 1086 | 1087 | 1088 | 1089 | 1090 | 1091 | 1092 | 1093 | 1094 | 1095 | 1096 | 1097 | 1098 | 1099 | 1100 | 1101 | 1102 | 1103 | 1104 | 1105 | 1106 | 1107 | 1108 | 1109 | 1110 | 1111 | 1112 | 1113 | 1114 | 1115 | 1116 | 1117 | 1118 | 1119 | 1120 | 1121 | 1122 | 1123 | 1124 | 1125 | 1126 | 1127 | 1128 | 1129 | 1130 | 1131 | 1132 | 1133 | 1134 | 1135 | 1136 | 1137 | 1138 | 1139 | 1140 | 1141 | 1142 | 1143 | 1144 | 1145 | 1146 | 1147 | 1148 | 1149 | 1150 | 1151 | 1152 | 1153 | 1154 | 1155 | 1156 | 1157 | 1158 | 1159 | 1160 | 1161 | 1162 | 1163 | 1164 | 1165 | 1166 | 1167 | 1168 | 1169 | 1170 | 1171 | 1172 | 1173 | 1174 | 1175 | 1176 | 1177 | 1178 | 1179 | 1180 | 1181 | 1182 | 1183 | 1184 | 1185 | 1186 | 1187 | 1188 | 1189 | 1190 | 1191 | 1192 | 1193 | 1194 | 1195 | 1196 | 1197 | 1198 | 1199 | 1200 | 1201 | 1202 | 1203 | 1204 | 1205 | 1206 | 1207 | 1208 | 1209 | 1210 | 1211 | 1212 | 1213 | 1214 | 1215 | 1216 | 1217 | 1218 | 1219 | 1220 | 1221 | 1222 | 1223 | 1224 | 1225 | 1226 | 1227 | 1228 | 1229 | 1230 | 1231 | 1232 | 1233 | 1234 | 1235 | 1236 | 1237 | 1238 | 1239 | 1240 | 1241 | 1242 | 1243 | 1244 | 1245 | 1246 | 1247 | 1248 | 1249 | 1250 | 1251 | 1252 | 1253 | 1254 | 1255 | 1256 | 1257 | 1258 | 1259 | 1260 | 1261 | 1262 | 1263 | 1264 | 1265 | 1266 | 1267 | 1268 | 1269 | 1270 | 1271 | 1272 | 1273 | 1274 | 1275 | 1276 | 1277 | 1278 | 1279 | 1280 | 1281 | 1282 | 1283 | 1284 | 1285 | 1286 | 1287 | 1288 | 1289 | 1290 | 1291 | 1292 | 1293 | 1294 | 1295 | 1296 | 1297 | 1298 | 1299 | 1300 | 1301 | 1302 | 1303 | 1304 | 1305 | 1306 | 1307 | 1308 | 1309 | 1310 | 1311 | 1312 | 1313 | 1314 | 1315 | 1316 | 1317 | 1318 | 1319 | 1320 | 1321 | 1322 | 1323 | 1324 | 1325 | 1326 | 1327 | 1328 | 1329 | 1330 | 1331 | 1332 | 1333 | 1334 | 1335 | 1336 | 1337 | 1338 | 1339 | 1340 | 1341 | 1342 | 1343 | 1344 | 1345 | 1346 | 1347 | 1348 | 1349 | 1350 | 1351 | 1352 | 1353 | 1354 | 1355 | 1356 | 1357 | 1358 | 1359 | 1360 | 1361 | 1362 | 1363 | 1364 | 1365 | 1366 | 1367 | 1368 | 1369 | 1370 | 1371 | 1372 | 1373 | 1374 | 1375 | 1376 | 1377 | 1378 | 1379 | 1380 | 1381 | 1382 | 1383 | 1384 | 1385 | 1386 | 1387 | 1388 | 1389 | 1390 | 1391 | 1392 | 1393 | 1394 | 1395 | 1396 | 1397 | 1398 | 1399 | 1400 | 1401 | 1402 | 1403 | 1404 | 1405 | 1406 | 1407 | 1408 | 1409 | 1410 |<th
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

11/30/2017.

Lecture 18.

A daboost.

$x_1 \dots x_j$	$y_i \in \{+1, -1\}$
$x_j \in \{+1, -1\}$	

$$y_i = \text{sign} \left(\sum_{k=1}^d \beta_k h_k(x_i) \right)$$

$$L(\beta) = \sum_{i=1}^n e^{-y_i \sum_{j=1}^p \beta_j x_{ij}}$$

exp loss

$$e^{-y_i x_i^\top \beta} = e^{-r}$$

restrict situation

x_i : raw observ.



$h_1, h_2, \dots, h_k, \dots$ hid weak classifiers

$$h_k(x_i) \in \{+1, -1\}$$

$$w_i = \frac{1}{n} \quad i=1 \dots n$$

Iterate: correct committee = $\sum_{\beta=1}^d \beta_k h_k(x_k)$ $\xrightarrow{\text{boost}}$ correct + $\beta_{\text{true}} h_{\text{true}}(x_i)$.

$$w_i = e^{-y_i \sum_{k=1}^d \beta_k h_k(x_i)}$$

$$w_i \leftarrow \frac{w_i}{\sum_{i=1}^n w_i}$$

$h_{\text{new}} \leftarrow \text{weak learn}(\{x_i, y_i, w_i\})$

$$\sum_{\text{mistakes of } h_{\text{new}}} = \sum_{i=1}^n w_i \quad y_i \neq h(x_i) \quad \beta_{\text{true}} = \frac{1}{2} \log \frac{1-\varepsilon}{\varepsilon}$$



Supported Vector Machine: hinge loss

$$x_{ij} : \text{continuous.} \quad L(\beta) = \sum_{i=1}^n \max(0, 1 - y_i x_i^\top \beta) + \frac{\lambda}{2} \|\beta\|_2^2$$

\hookrightarrow better beyond 1.
then you won't suffer the loss

$$L'(\beta) = -\sum_{i=1}^n \mathbb{1}(y_i x_i^\top \beta \leq 1) y_i x_i + \lambda \beta$$

↓
support vectors

my-SVM ← function (X-train, Y-train, X-test, Y-test, lambda = 0.1, num-iterations = 1000, learning-rate = .1)

for (it in 1: numIterations) {

 score ← X-train %*% beta

 delta ← score * Y-train < 1

 d_beta ← matrix(rep(1, n), nrow) %*% ((matrix(delta * Y, n, p) * X-train)) / n

 beta ← beta + learning-rate * t(d_beta)

 beta[2:p] ← beta[2:p] - lambda + beta[2:p]

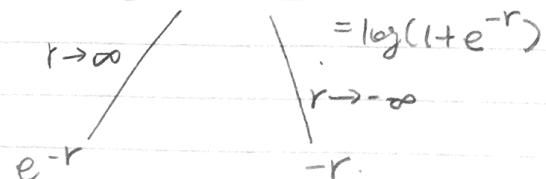
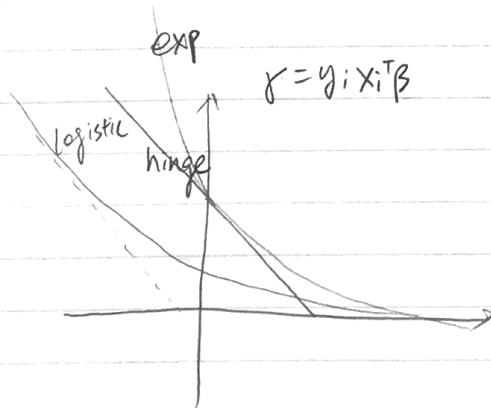
}

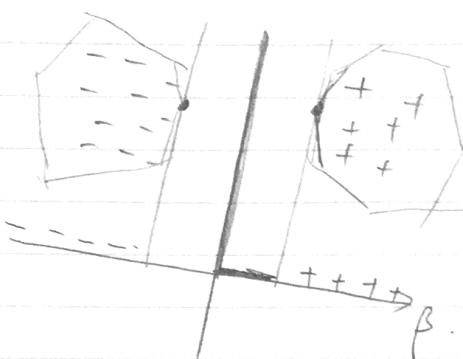
Logistic Regression.

$$P(y_i = +1) = \frac{e^{x_i^\top \beta}}{1 + e^{x_i^\top \beta}} = \text{sigmoid}(x_i^\top \beta) = \frac{1}{1 + e^{-x_i^\top \beta}} \quad \left. \Rightarrow P(y_i | x_i^\top \beta) \right. \\ = \frac{1}{1 + e^{-y_i + x_i^\top \beta}}$$

$$P(y_i = -1) = \frac{1}{1 + e^{x_i^\top \beta}}$$

$$-\log P(y_i | x_i^\top \beta) = \log(1 + e^{-y_i x_i^\top \beta}) \quad \text{logistic loss.}$$





$$y_i = 1, \quad x_i^T \beta \geq 1$$

$$y_i = -1, \quad x_i^T \beta \leq -1$$

$$\vec{u} = \frac{\beta}{\|\beta\|_2} \quad \text{marginal margin}$$

$$\langle x_i, \vec{u} \rangle = \langle x_i, \frac{\beta}{\|\beta\|_2} \rangle = \frac{x_i^T \beta}{\|\beta\|_2}$$

$$= \frac{1}{\|\beta\|_2}$$

$$\min \frac{1}{2} \|\beta\|_2^2$$

$$\text{subject to } y_i x_i^T \beta \geq 1 \quad i=1, \dots, n$$

line-separable slack-variable ξ_i

$$\min \frac{1}{2} \|\beta\|_2^2 + C \#(\xi_i > 0)$$

$$\text{s.t. } y_i x_i^T \beta \geq 1 - \xi_i \quad \rightarrow C \sum_{i=1}^n \xi_i$$

$$\xi_i = \max(0, 1 - y_i x_i^T \beta) \quad \xi_i \geq 0.$$

$$x_+ = \sum_{i \in +} c_i x_i$$

$$c_i \geq 0 \quad \sum_{i=1}^n c_i$$

$$x_- = \sum_{i \in -} c_i x_i$$

$$c_i \geq 0 \quad \sum_{i=1}^n c_i$$

Condition

$$\min \|x_+ - x_-\|^2 :$$

$$C = \{c_i\}$$

$$\|x_+ - x_-\|^2 = \left\| \sum_{i=1}^n c_i y_i x_i \right\|^2$$

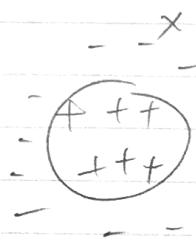
$$= \sum_{i,j} c_i c_j y_i y_j \langle x_i, x_j \rangle.$$

realistic situation.

x_i : raw observ.

↓

$$(h_1, \dots, h_r, \dots, h_d) = h(x)$$



$$c_i > 0 \rightarrow \text{s.v.'s } \langle h(x_i), h(x_j) \rangle$$

sequential marginal optimial (SMO)
each time change 2 ci.

$$\text{kernal trick. } = k \langle x_i, x_j \rangle$$

$$K(x_i, x_j) = e^{-\frac{(x_i - x_j)^2}{2\sigma^2}}$$

y_i

$$h_{ik} \quad h_{jk} \quad \dots = h(x)$$

$y_i \sim \text{logistic regression}$

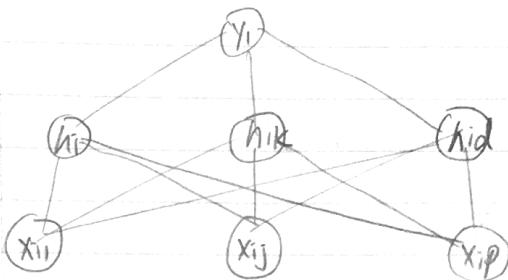
(x_{i1}, \dots, x_{id})

$h_{ik} \sim \text{logistic regression}$

$(x_{ii}, x_{iz}, \dots, x_{ip})$

12/5/2017

Lecture 19



AdaBoost: \$h_{ik}\$: weak classifier
hypothesis.

Selected sequentially.

$$\text{SVM: } h_i = \begin{pmatrix} h_i \\ \vdots \\ h_{ik} \\ \vdots \\ h \end{pmatrix} \quad h_i = h(x_i) \quad h_i: \text{implicit.} \\ d = \infty$$

$$y_i = \text{sign}(\langle h(x), \beta \rangle)$$

kernel trick: reproducing kernel

learning rate.

$$\langle h(x_i), h(x_j) \rangle = K(x_i, x_j)$$

$$\beta_{t+1} = \beta_t + \gamma_t \sum_{i=1}^n 1(\langle h(x_i), \beta \rangle < 1) \cdot y_i h(x_i)$$

$$\rightarrow y = \text{sign}(\langle h(x), \beta \rangle) = \text{sign}(\langle h(x_i), \sum_{i=1}^n \alpha_i h(x_i) \rangle) \\ = \text{sign}(\sum_{i=1}^n \alpha_i K(x, x_i))$$

Neural network:

y_i : logistic regression on \$(h_{i1}, \dots, h_{id})\$. cross entropy

$$P(y_i=1 | h_i) = \text{sigmoid}(h_i^T \beta = \sum_{k=1}^d \beta_k \cdot h_{ik})$$

$$h_{ik} = \text{sigmoid}(x_i^T \underbrace{\alpha_k}_{\gamma_{ik}} = \sum_{j=1}^p \alpha_{kj} \cdot x_{ij})$$

$$\text{maximum likelihood: } L(\alpha, \beta) = \prod_{i=1}^n p_i^{y_i} (1-p_i)^{1-y_i}$$

$$= \prod_{i=1}^n \left(\frac{e^{h_i^T \beta}}{1+e^{h_i^T \beta}} \right)^{y_i} \left(\frac{1}{1+e^{h_i^T \beta}} \right)^{1-y_i}$$

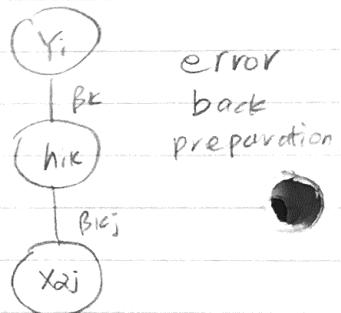
$$= \prod_{i=1}^n \frac{e^{y_i + h_i^\top \beta}}{1 + e^{y_i + h_i^\top \beta}}$$

$$\ell(\alpha, \beta) = \log L(\alpha, \beta) = \sum_{i=1}^n [y_i h_i^\top \beta - \log(1 + e^{h_i^\top \beta})]$$

$$\frac{\partial \ell}{\partial \beta_k} = \sum_{i=1}^n (y_i h_{ik} - \frac{e^{h_i^\top \beta}}{1 + e^{h_i^\top \beta}} \cdot h_{ik}) = \sum_{i=1}^n (y_i - p_i) \cdot h_{ik}$$

$$\frac{\partial \ell}{\partial \alpha_j} = \sum_{i=1}^n \frac{\partial \ell}{\partial h_{ik}} \cdot \frac{\partial h_{ik}}{\partial \alpha_j} = \sum_{i=1}^n (y_i \cdot \beta_k - p_i \cdot \beta_k) \cdot \frac{\text{d sigmoid}}{\text{d } y_{ik}} x_{ij}$$

$$= \sum_{i=1}^n \frac{(y_i - p_i) \beta_k \cdot h_{ik}(1 - h_{ik})}{\text{error}} x_{ij}$$



My_NN ← function (X_train, Y_train, X_test, Y_test, num_hidden=20,
num_iterations=1000, learning_rate = ...)
 { n, p

for (it in 1:num_iterations)

{

$$h \leftarrow 1 / (1 + \exp(-x_train \%*\% alpha))$$

hxp pxl

$$p \leftarrow 1 / (1 + \exp(-h \%*\% beta))$$

$$dbeta \leftarrow matrix(rep(1, n), nrow=1) \%*\% ((matrix(dbeta - pr, n, d) * h)) / n.$$

	p	d
X	x	h
Y		y

Rectified linear unit:

$$h_k = \max(0, x_i^\top \beta_k)$$

Sigmoid

