STA303/1002 - Methods of Data Analysis II

(Week 06 lecture note - Extra Topics)

Wei (Becky) Lin

Feb 7/9, 2017

Notes

- Assignment 2 is due 22:00, Sunday, Feb 26, 2017.
- Midterm is right after the reading week, Mar 2nd.
- Midterm will cover lecture note from week 2 to week 7.

Q & A: ANOVA and identifiability issues

- For multiple linear regression

$$Y = X\beta + \epsilon \rightarrow \hat{\beta} = (X'X)^{-1}X'Y$$

For same Y, different X leads to different $\hat{\beta}$

  - the LSE of β depends on the design matrix.
- The one-way ANOVA
  - Subcript i for treatment level indicator, assume i=1,2,3
  - Subscript j for replicates at each treatment level
  - Y_{ij} the j^{th} observation on level i treatment, ϵ_{ij} is the corresponding error.
  - Assumption: $\epsilon_{ij} \sim_{iid} N(0, \sigma^2)$
- The cell mean model form for one-way ANOVA $Y_{ij} = \mu_i + \epsilon_{ij}$
- Fixed effect model form for one-way ANOVA

$$Y_{ij} = (\mu + \alpha_i) + \epsilon_{ij} = \{(\mu - \delta_0) + (\alpha_i + \delta_0)\} + \epsilon_{ij}$$

  - Over parameterization: imposing constraint to estimate μ, α_i.

Q & A: ANOVA and identifiability issues

- Fixed effect model form for one-way ANOVA

$$Y_{ij} = (\mu + \alpha_i) + \epsilon_{ij}$$

  - Consider imposing $\alpha_1 = 0$, then this implies that

$$Y_{1j} = \mu + \epsilon_{ij} \rightarrow; \hat{\mu} = \bar{Y}_{1.}$$

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij} \rightarrow; \widehat{\mu + \alpha_i} = \bar{Y}_{i.} \rightarrow; \hat{\alpha}_i = \bar{Y}_{i.} - \bar{Y}_{1.}$$

Data

resp	trmt
Y_1	1
Y_2	1
Y_3	2
Y_4	2
Y_5	3
Y_6	3

  - The corresponding regression model under $\alpha_1 = 0$

$Y_{ij} = \beta_0 + \beta_1 I_{T=2} + \beta_2 I_{T=3} + \epsilon_{ij}$, where $I_{T=i} = 1$ if T=i, ow, 0

$$\begin{pmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \hat{Y}_3 \\ \hat{Y}_4 \\ \hat{Y}_5 \\ \hat{Y}_6 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}}_{= X_{6\times 3}} \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \end{pmatrix}, \quad \hat{\beta} = (X'X)^{-1}X'Y$$

I_2 I_3

$I_2 = \begin{cases} 1 & T=2 \\ 0 & ow \end{cases}$ $I_3 = \begin{cases} 1, & T=3 \\ 0, & ow \end{cases}$

$$\hat{\beta}_0 = \hat{\mu} = \bar{Y}_{1.}; \quad \hat{\beta}_i = \hat{\alpha}_i = \bar{Y}_{i.} - \bar{Y}_{1.}, i = 2, 3$$

Q & A: ANOVA and identifiability issues

- Fixed effect model form for one-way ANOVA

$$Y_{ij} = (\mu + \alpha_i) + \epsilon_{ij}$$

  - The corresponding regression model under $\sum_i^3 \alpha_i = 0$

$$\sum_i \sum_j Y_{ij} = n\mu + 0 + \sum_i \sum_j \epsilon_{ij} \rightarrow \hat{\mu} = \bar{Y}_{..}$$

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij} \rightarrow; \widehat{\mu + \alpha_i} = \bar{Y}_{i.} \rightarrow; \hat{\alpha}_i = \bar{Y}_{i.} - \bar{Y}_{..}$$

  - The corresponding regression model under $\sum_i^3 \alpha_i = 0$

$$Y_{ij} = \beta_0 + \beta_1 I_{T=1} + \beta_2 I_{T=2} + \epsilon_{ij}, \quad \text{where}$$

$$I_1 = I_{T=1} = \begin{cases} 1, T = 1 \\ -1, T = 3 \\ 0, ow. \end{cases} \quad , I_{T=2} = \begin{cases} 1, T = 2 \\ -1, T = 3 \\ 0, ow. \end{cases}$$

Q & A: ANOVA and identifiability issues

- Fixed effect model form for one-way ANOVA

$$Y_{ij} = (\mu + \alpha_i) + \epsilon_{ij}$$

  - The corresponding regression model under $\sum_i^3 \alpha_i = 0$

$$Y_{ij} = \beta_0 + \beta_1 I_{T=1} + \beta_2 I_{T=2} + \epsilon_{ij}, \quad \text{where}$$

$$I_{T=1} = \begin{cases} 1, & T = 1 \\ -1, & T = 3 \\ 0, & ow. \end{cases}, \quad I_{T=2} = \begin{cases} 1, & T = 2 \\ -1, & T = 3 \\ 0, & ow. \end{cases}$$

  - Matrix form

$$\begin{pmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \hat{Y}_3 \\ \hat{Y}_4 \\ \hat{Y}_5 \\ \hat{Y}_6 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \end{pmatrix}, \quad \hat{\beta} = (X'X)^{-1}X'Y$$

I_1 I_2 ⇒ new X; sum=0, sum=0

$$\hat{\beta}_0 = \hat{\mu} = \bar{Y}_{..}; \quad \hat{\beta}_i = \hat{\alpha}_i = \bar{Y}_{i.} - \bar{Y}_{..}, i = 2, 3$$

- Extend above idea to multiple-way ANOVA, then we obtain estimates of the main effect(s) $\alpha_i, \hat{\beta}_j$ and interactio effect $(\alpha\beta)_{ij}$

Topics learned before WLS (Weighted LS Method)

- Main objective: mean comparison or test equal means
  - one-sample t-test, paired-test: need to know assumption, test statistics, H_0, H_a.
  - two-sample t-test: equal/unequal variance, also need to know assumption, test statistics, H_0, H_a.
  - Generalize two-sample t-test to test equal means for more than two groups
    - One-way ANOVA
    - Two-way ANOVA (with/without interaction term)
    - ANCOVA: Regression+ANOVA, including confounding covariates (non-factor variables) in ANOVA analysis (Key assumption: homogeneous regression slopes, how to evaluate this assumption?)
- All above method assume normality, but due to CLT, all the testing methods proposed above are quite robust against non-normality. However, with non-normal and highly skewed distributions, it might be more appropriate to use nonparametric tests.
  - Reference: Khan, A., & Rayner, G. D. (2003). Robustness to non-normality of common tests for the many-sample location problem. Journal of Applied Mathematics and Decision Sciences, 7, 187-206

Extending Linear Regression to WLS

- For linear regression, we aim to minimize the following objective/loss function

$$Q_{ols} = \overbrace{\sum_{i}^{n}(y_i - \vec{x_i} \cdot \beta)^2}^{SSE} = (Y - X\beta)'(Y - X\beta)$$

$$\hat{\beta}_{ols} = (X'X)^{-1}X'Y$$

- To accomodate non-constant variance, Weighted Least Square (WLS) is proposed which minimizes Q_{wls} with weight matrix, W

$$Q_{wls} = \overbrace{\sum_{i}^{n} w_i(y_i - \vec{x_i} \cdot \beta)^2}^{\text{weighted SSE}} = (Y - X\beta)'W(Y - X\beta)$$

$$\hat{\beta}_{wls} = (X'WX)^{-1}X'WY$$

  - In reality, hard to find the weight matrix W
  - If we know all entries of var-covariance matrix, $Var(\epsilon) = \Sigma$, then $W = \Sigma^{-1}$

From WLS to LOESS and LOWESS

- LOESS: LOcally regrESSion
- LOWESS: LOcally Weighted regrESSion
- LOESS and LOWESS are two strongly related non-parametric regression methods that combine multiple regression models in a k-nearest-neighbor-based meta-model. "LOESS" is a later generalization of LOWESS. (from Wikipedia)
- Aim to minimize:

$$Q_{loc} = \sum_{i}^{n} w_i (y_i - \vec{x_i} \cdot \beta)^2$$

Same loss func. as WLS but $w_i \propto$ kernel func

where w_i is proportional to the kernels, $w_i(x) \propto K(x_i, x)$.

- For locally linear regression, a common choice of kernel is tri-cubic,

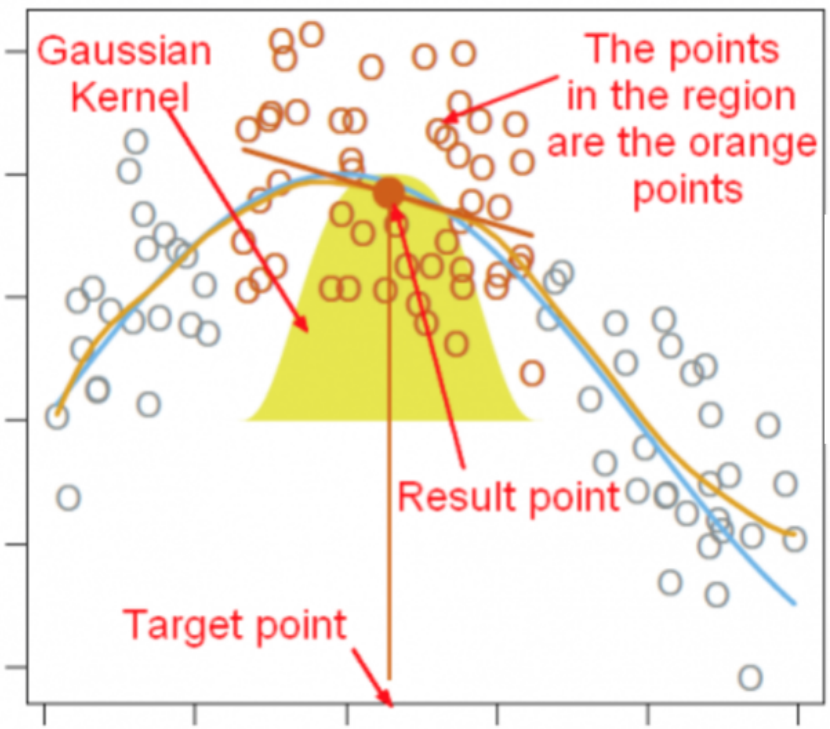
$$K(x_i, x) = \left(1 - \left(\frac{|x_i - x_0|}{h}\right)^3\right)^2 \text{ if } |x - x_i| < h, ow, 0$$

where h = width of the window.

LOESS and LOWESS

Procedure

- A linear function is fitted only on a local set of point delimited by a region. The polynomial is fitted using weighted least squares. The weights are given by the heights of the kernel (the weighting function) giving:
  - more weight to points near the target point (x) whose response is being estimated
  - and less weight to points further away

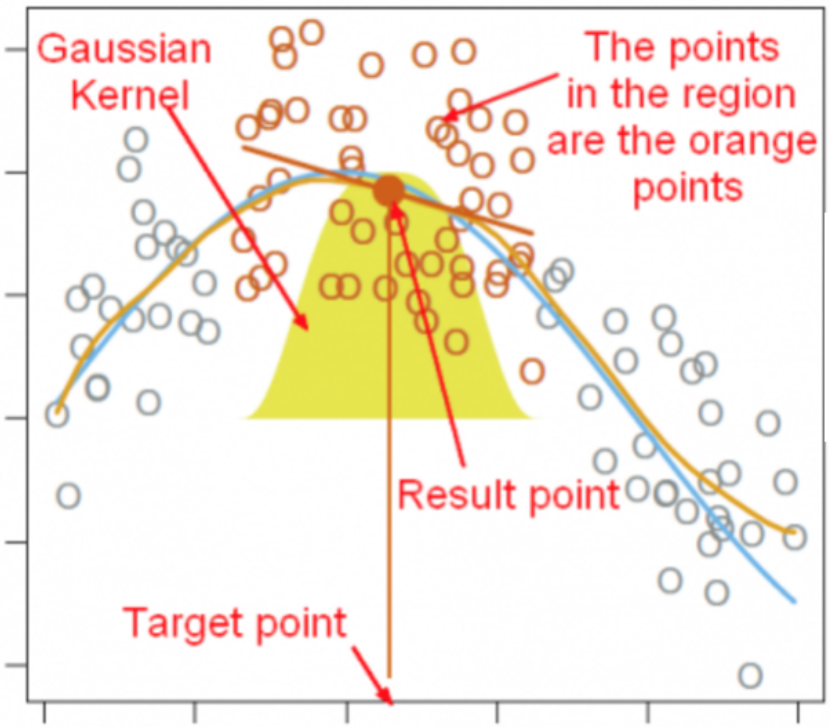


where:

- The orange curve is the fitted function.
- The points in the region are the orange one.
- The kernel is a Gaussian distribution.

- referece: `http://gerardnico.com/wiki/data_mining/local_regression`

LOESS and LOWESS



where:

- The orange curve is the fitted function.
- The points in the region are the orange one.
- The kernel is a Gaussian distribution.

- We obtain then a fitted polynomial model but retains only the point of the model at the target point (x). The target point then moves away on the x axis and the procedure repeats and that traces out the orange curve.
- Loess explained in a GIF:* ← watch it !!
  `http://simplystatistics.org/2014/02/13/loess-explained-in-a-gif/`

Ridge Regression

Bias Variance trade off: EPE decomposition

- $Y = f(x) + \epsilon$, $\hat{Y} = \hat{f}(x)$,prediction error $= Y - \hat{f}(x)$. We want to $(Y - \hat{f}(x))^2$ to be minimal. , $Var(\varepsilon) = \sigma^2$
- Decomposing of Expected Prediction Error, EPE$=$E$\left((Y - \hat{f})^2\right)$

$$EPE = E\left((Y - \hat{f})^2\right) = bias(\hat{f})^2 + var(\hat{f}) + \sigma^2$$

- As model complexity increases, bias decreases while variance increases. How to achieve a balance?

Proof: $EPE = E(Y^2 - 2Y\hat{f} + \hat{f}^2)$

$= E(Y^2) - 2E((f+\varepsilon)\hat{f}) + E(\hat{f}^2)$

$= Var(Y) + (EY)^2 - 2fE(\hat{f}) - 2E(\varepsilon\hat{f}) + Var(\hat{f}) + (E\hat{f})^2$ ⟩ $\varepsilon \perp \hat{f}$, $E(\varepsilon) = 0$

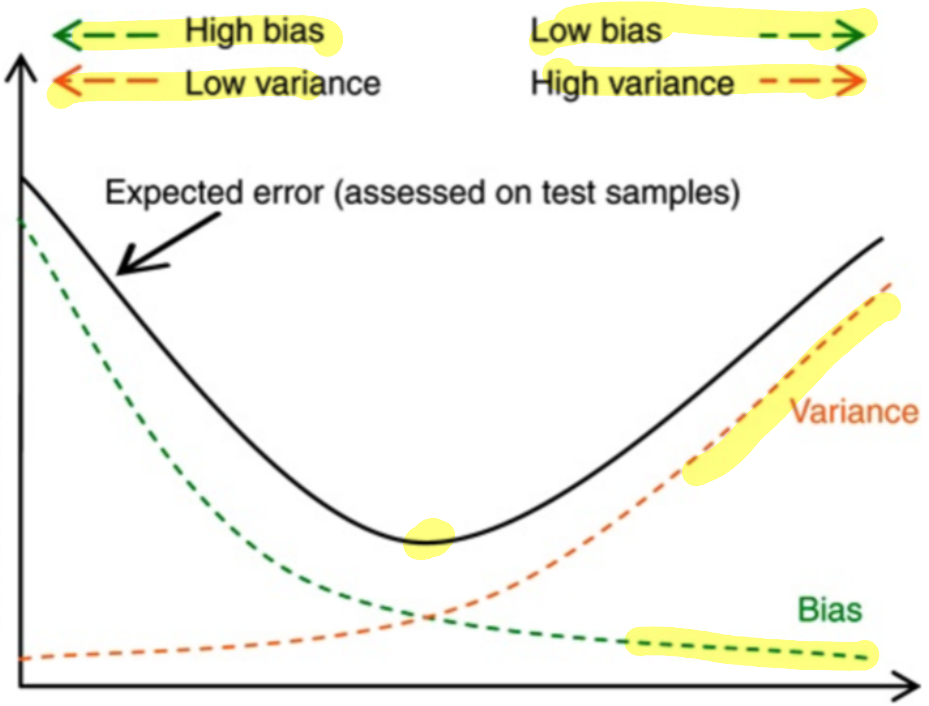
$= \sigma^2 + f^2 - 2fE(\hat{f}) + 2\underbrace{E(\varepsilon)E(\hat{f})}_{=0} + Var(\hat{f}) + (E\hat{f})^2$

$= \sigma^2 + Var(\hat{f}) + (E\hat{f})^2 - 2fE(\hat{f}) + f^2$

$= \sigma^2 + Var(\hat{f}) + \underbrace{(E\hat{f} - f)^2}_{bias(\hat{f})}$

Q.E.D.

Bias Variance trade off: MSE decomposition



want to have smaller MSE: bias-var trade-off

- The mean square error (MSE) of an estimator can be decomposed as follows

$$MSE(\hat{\theta}) = E\left((\theta - \hat{\theta})^2\right) = bias(\hat{\theta})^2 + Var(\hat{\theta})$$

proof: $MSE(\hat{\theta}) = E(\theta^2 - 2\theta\hat{\theta} + \hat{\theta}^2)$

$= \theta^2 - 2\theta E(\hat{\theta}) + E(\hat{\theta}^2) = \theta^2 - 2\theta E(\hat{\theta}) + V(\hat{\theta}) + (E\hat{\theta})^2$

$= (E(\hat{\theta}) - \theta)^2 + V(\hat{\theta})$

Q.E.D.

Model Shrinkage Methods

- Bias-variance trade-off

$$EPE = \sigma^2 + (\text{model bias})^2 + \text{model variance}$$

$$MSE(\hat{\theta}) = Bias(\hat{\theta})^2 + Var(\hat{\theta})^2$$

- Mallows C_p Statistics

From model i with p of X_s

$$C_p = SSE + 2p\hat{\sigma}, p = \text{number of X}$$

$\hat{\sigma}$ → estimated from full model ($Y \sim$ all X_s)

  - The second term puts "penalty" for model size.

- Today: Penalities based on $\hat{\beta}$.
  - Ridge Regression → L_1 penalty
  - LASSO → L_2 penalty

Motivation of Ridge Regression (or Shrinkage regression)

- Motivation 1: multicollineary exists
  - the matrix $X'X$ is not invertible, $\hat{\beta} = (X'X)^{-1}X'Y$ becomes a problem.
- Motivation 2: too many predictors, e.g. dim(X)=p, dim(Y)=n, p is a larger number
  - $p < n$ but large. With a large number of predictors, it can be helpful to identify a smaller subset of important variables. Linear regression doesn't do this
  - linear regression is not defined when $p > n$.

Set up:

- From MSE decomposition, we see that with unbiased estimators, minimizing MSE is the same as minimizing the variance.
- Linear regression has low bias (zero bias) but suffers from high variance. So it may be worth sacrificing some bias to achieve a lower MSE.
- So we should consider some biased estimators if they have significantly lower variance

Ridge regression: fix singularity problem

Problem

In case of singular, $X'X$, its inverse is not defined. Consequently, the OLS estimator

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

does not exist. This happens in high-dimensional data.

Solution

- Hoerl and Kennard (1970) proposed that potential instability in the LS estimator could be improved by adding a small constant value λ to the diagonal entries of the matrix $X'X$ before taking its inverse.
- The ad-hoc solution adds λI to $X'X$

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

This is called the ridge estimator.

Ridge regression: fix singularity problem

Example:

$$Y = \begin{pmatrix} 1.3 \\ -0.5 \\ 2.6 \\ 0.9 \end{pmatrix}, \; X = \begin{pmatrix} 1 & -1 & 2 \\ 1 & 0 & 1 \\ 1 & 2 & -1 \\ 1 & 1 & 0 \end{pmatrix}, \; X'X = \begin{pmatrix} 4 & 2 & 2 \\ 2 & 6 & -4 \\ 2 & -4 & 6 \end{pmatrix}$$

$det(X'X) = 0$ since the eigenvalues of $X'X$ has 10, 6 and 0. With the "ridge-fix", for $\lambda = 1$ we get

$X'X$ is not invertible.

$$X'X + \lambda I = \begin{pmatrix} 5 & 2 & 2 \\ 2 & 7 & -4 \\ 2 & -4 & 7 \end{pmatrix}$$

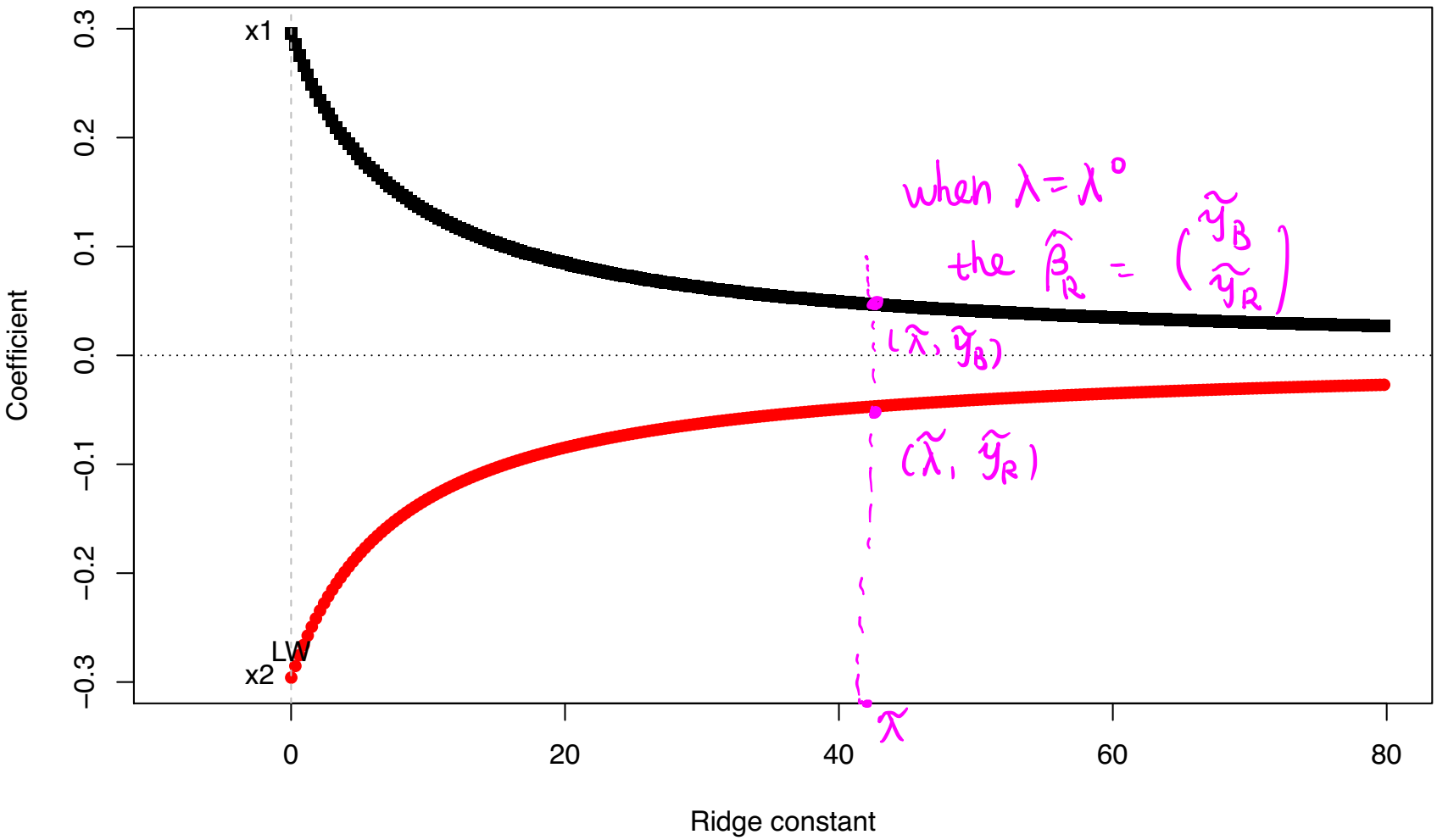
fix it

From given data, we have

$$\lambda = 1, \hat{\beta}_\lambda = \begin{pmatrix} 0.614 \\ .548 \\ 0.066 \end{pmatrix}, \quad \lambda = 10, \hat{\beta}_\lambda = \begin{pmatrix} 0.269 \\ 0.267 \\ 0.002 \end{pmatrix},$$

Ridge Regression: trace-plot

```
Y = c(1.3, -0.5, 2.6, 0.9)
x1 = c(-1,0,2,1); x2 = c(2,1,-1,0)
library(genridge)
fit = ridge(Y~x1+x2,lambda = seq(0.01, 80, 0.3))
traceplot(fit)
```

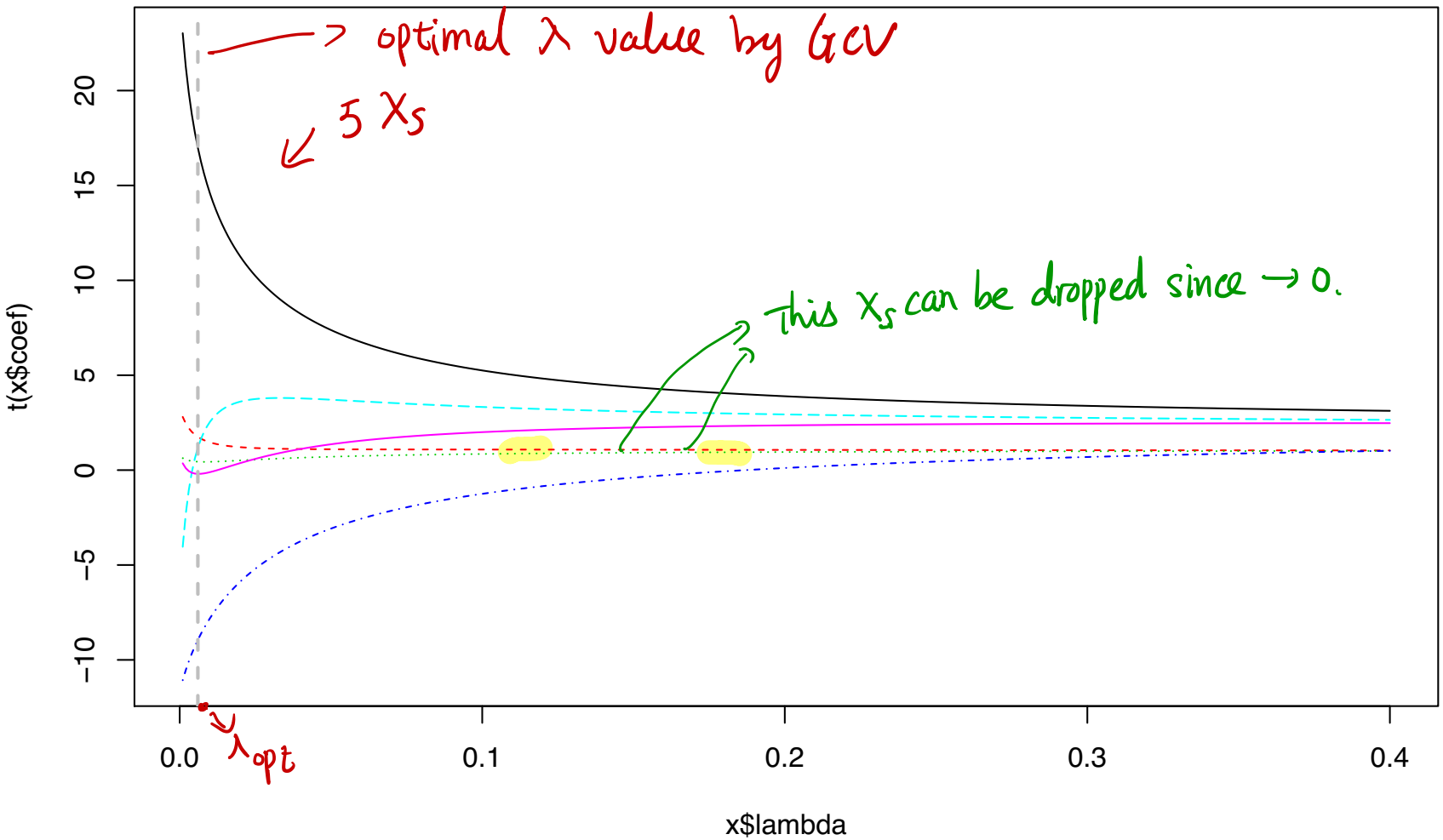


Ridge regression: learned from example

- Predictions tend to be more precise, and estimates are more stable
- Can use the tranceplot to reduce number of predictor variables
  - drop ones with unstable trace, or a trace that tends to zero
- Exact distributional properties of estimators and fitted values are not known.
  - use bootstrapping to get SEs.

data example

```
library(MASS)
data(longley); names(longley)[1] <- "y"
mod = lm.ridge(y ~ ., longley,lambda = seq(0.001,0.4,0.001))
plot(mod); abline(v=mod$lambda[mod$GCV==min(mod$GCV)],lty=2,col="gray",lwd=2)
```



Ridge Regression

We assume only that X's and Y have been centered, so that we have no need for a constant term in the regression β_0:

- X is a n by p matrix with centered column
- Y is a centered vector of size n.

Ridge was developed before LASSO, it is based on the idea of constrained minimization:

$$\min_{\beta} \sum_{i}^{n} (Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2, \quad \text{Subject to} \sum_{j=1}^{p} \beta_j^2 < C$$

By the Lagrange multiplier method, this is equivalent to:

$$\min_{\beta} \sum_{i}^{n} (Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda_C \sum_{j=1}^{p} \beta_j^2$$

The second term is a penalty that depends $||\beta||_2^2$, L_2 norm penalty.

Ridge Regression

Ridge regression minimizes the following loss/objective function Q_R

$$Q_R = \sum_{i}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

SSE

which can be written in matrix form

$$Q_R = (Y - X\beta)'(Y - X\beta) + \lambda\beta'\beta$$

SSE + $\lambda\beta'\beta$

- In statistics this is also called "shrinkage": you are shrinking $||\beta||^2$ towards 0.
- λ is a shrinkage or tuning parameter that you have to choose.
- The Ridge solution $\hat{\beta}_\lambda$ is easy to solve, because the above is still a quadratic function in β.

$\hat{\beta}_\lambda$: has an analytical form

Show the Ridge solution, $\hat{\beta}_\lambda$

$$Q_R = (Y - X\beta)'(Y - X\beta) + \lambda\beta'\beta \rightarrow \hat{\beta}_\lambda = (X'X + \lambda I)^{-1}X'Y$$

$$= (Y' - \beta'X')(Y - X\beta) + \lambda\beta'\beta$$

$$= Y'Y - Y'X\beta - \beta'X'Y + \beta'X'X\beta + \lambda\beta'\beta$$

$$= Y'Y - Y'X\beta - \beta'X'Y + \beta'(X'X + \lambda I)\beta$$

$$(X'X + \lambda I)' = X'X + \lambda I$$

$$\frac{\partial Q_R}{\partial \beta} = -Y'X - (X'Y)' + 2\beta'(X'X + \lambda I)$$

$$= -2Y'X + 2\beta'(X'X + \lambda I)$$

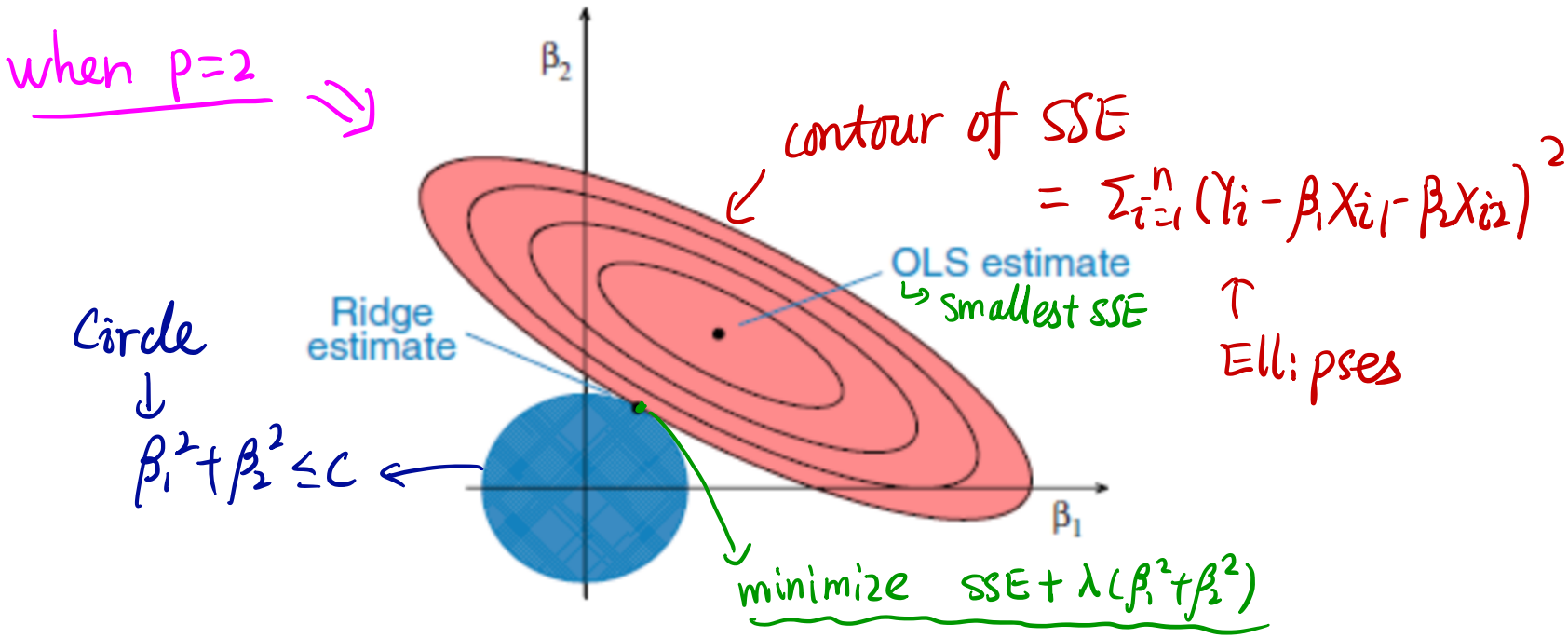
$$= 0$$

$$\Rightarrow \quad X'Y = (X'X + \lambda I)\beta$$

$$\Rightarrow \quad \hat{\beta}_{Ridge} = (X'X + \lambda I)X'Y$$

QED.

Geometric Interpretation of Ridge Regression



- The ellipses correspond to the contours of residual sum of squares (SSE): the inner ellipse has smaller SSE, and SSE is minimized at ordinary least square (OLS) estimates.
- For p=2, the constraint in ridge regression corresponds to a circle, $\beta_1^2 + \beta_2^2 = c$
- The larger the λ is, the more you prefer the circle is close to zero, that is the same to prefer the $\beta_j's$ close to zero.

credit to this slide: `https://onlinecourses.science.psu.edu/stat857/node/155`

Ridge Solutions

- $\hat{\beta}_\lambda = (X'X + \lambda I)^{-1}X'Y$, the solution is indexed by the parameter λ
  - For each λ, we have a solution
  - The λ's trance out a path of solution.
  - λ is the shrinkage parameter: it controls the size of the coefficients and the amount of regularization. As $\lambda \to 0$, we have OLS, as $\lambda \to \infty$, we have $\hat{\beta}_\lambda = 0$(intercept model only).
- Whereas the LS solutions $\hat{\beta} = (X'X)^{-1}X'Y$ are unbiased if model is correctly specified, ridge solutions are biased

$$E(\hat{\beta}_\lambda) \neq \beta$$

- However, at the cost of bias, Ridge reduces the variance, and thus might reduce MSE

bias-var trade-off.

$$MSE = bias^2 + variance$$

- Ridge solutions are hard to interpret, because it is not sparse. (Sparse means that some β_j are set exactly to 0)

How do we choose λ in Ridge regression

- We want to choose λ that minimize the MSE
- How: Cross Validation, LOOCV, or GCV.
  - A convenient approximation to LOOCV (leave one out) is called the generalized cross validation (GCV)
- K-fold cross validation
  - 1. Split the training data T into K=5 or 10 subset with equal sample size
  - 2. For i=1,. . . , K, find the $\hat{\beta}_\lambda$ to the training data excluding k-th fold data
  - 3. Find the fitted values for the k-th fold data
  - 4. compute the cross-validation (CV) error $CVE_k^\lambda = \frac{1}{n_k}\sum_i (Y_i - \hat{Y}_i)^2$
  - the model then has overall cross-validation error $CVE^\lambda = \frac{1}{K}\sum_k CVE_k^\lambda$
  - select λ^* as the one with minimum CVE^λ
- Apply $\hat{f}(x, \lambda^*)$ to the test set to assess test error.

LASSO: L_1 penalties

What if we constrain the L1 norm instead of the Euclidean norm?

$$\min_{\beta} \sum_{i}^{n} (Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2, \quad \text{Subject to } \sum_{j=1}^{p} |\beta_j| < C$$

This is a subtle, but important change.

- Tibshirani (Journal of the Royal Statistical Society 1996) introduced the LASSO: least absolute shrinkage and selection operator.
- LASSO coefficients are the solution to the L_1 optimization problem of the following loss function

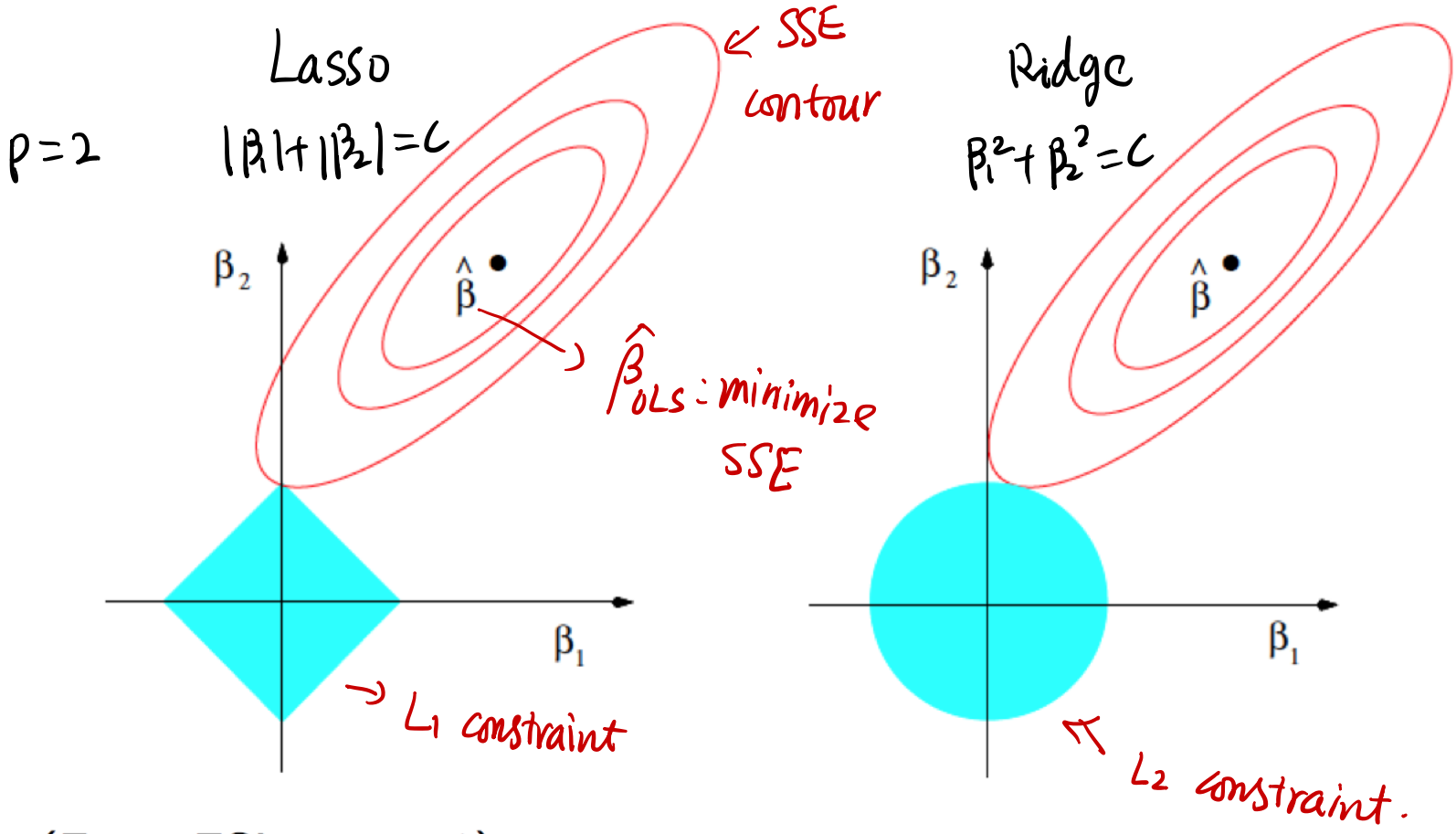
$$f(\beta) = (Y - X\beta)'(Y - X\beta) + \lambda||\beta||_1$$

- Unlike Ridge, there is no analytic solution for the LASSO
- Efron et al. (2002) gave an efficient algorithm *lars* (Least Angle Regression) to solve the Lasso.
- λ can be selected based on any of the model selection criterions.
- LASSO solutions are sparse: it drives some β_j to be zero.
- LASSO: estimation and variable selection are obtained simultaneously.

Cool ^^

Ridge vs LASSO

The lasso combines some of the shrinking advantages of ridge with variable selection



(From ESL page 71)

Take a break, and see you on Thursday

Definiton of Norms

- In mathematics, the L^p spaces are function spaces defined using a natural generalization of the p-norm for finite-dimensional vector spaces. They are sometimes called Lebesgue spaces.
  - Definition: for a real number $p \geq 1$, the p-norm or L^p-norm of x is defined by
$$||x||_p = (|x_1|^p + \ldots + |x_n|^p)^{1/p}$$

- The p-norm can be extended to vectors that have an infinite number of components, which yields sthe ℓ^p norm, some special ℓ^p cases
  - ℓ^1, the space of sequences whose series is absolutely convergent.
  - ℓ^2, the space of squared-summable sequences,which is a Hilbert space.
  - ℓ^∞, the space of bounded sequences.

Above from `https://en.wikipedia.org/wiki/Lp_space`

L1 and L2 penalty in penalized regression

$\sum_{j=1}^{p} |\beta_j| < c$ $\sum_{j=1}^{p} \beta_j^2 < c$

- In penalized regression, **L1 penalty** and **L2 penalty** refer to penalizing either the L1 norm of a solution's vector of parameter values (i.e. the sum of its absolute values), or its L2 norm (its Euclidean length).
  - Techniques which use an L1 penalty, like LASSO, encourage solutions where many parameters are zero.
  - Techniques which use an L2 penalty, like ridge regression, encourage solutions where most parameter values are small.
  - Elastic net regularization uses a penalty term that is a combination of the L1 norm and the L2 norm of the parameter vector.

Above from `https://en.wikipedia.org/wiki/Lp_space`

Ridge regression: simulation study

```
set.seed(123456789)
N = 30       # Sample size
x1 = runif(n=N)
x2 = runif(n=N)
x3 = runif(n=N)
x3c = 10*x1 + x3 # x3c is correlated with x1,x3
ep = rnorm(n=N)
y = x1 + x2 + ep
cor(cbind(x1,x2,x3,x3c))
```

- X_1, X_2, X_3 are indep.
- X_{3c} depends on X_1, X_3

```
##              x1          x2          x3        x3c
## x1   1.00000000  0.01764729 -0.05902616 0.99588304
## x2   0.01764729  1.00000000 -0.08378183 0.01006135
## x3  -0.05902616 -0.08378183  1.00000000 0.03170629
## x3c  0.99588304  0.01006135  0.03170629 1.00000000
```

Ridge regression: simulation study

```r
# OLS fit: model with independent covariates
ols <- lm(y~ x1 + x2 + x3)
summary(ols)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0313 -0.6480 -0.1313  0.7306  1.7187
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.12632    0.58723  -0.215   0.8314
## x1           1.42031    0.58447   2.430   0.0223 *
## x2           1.06167    0.60878   1.744   0.0930 .
## x3          -0.05271    0.64662  -0.082   0.9357
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9822 on 26 degrees of freedom
## Multiple R-squared:  0.2613, Adjusted R-squared:  0.176
## F-statistic: 3.065 on 3 and 26 DF,  p-value: 0.04564
```

Ridge regression: simulation study

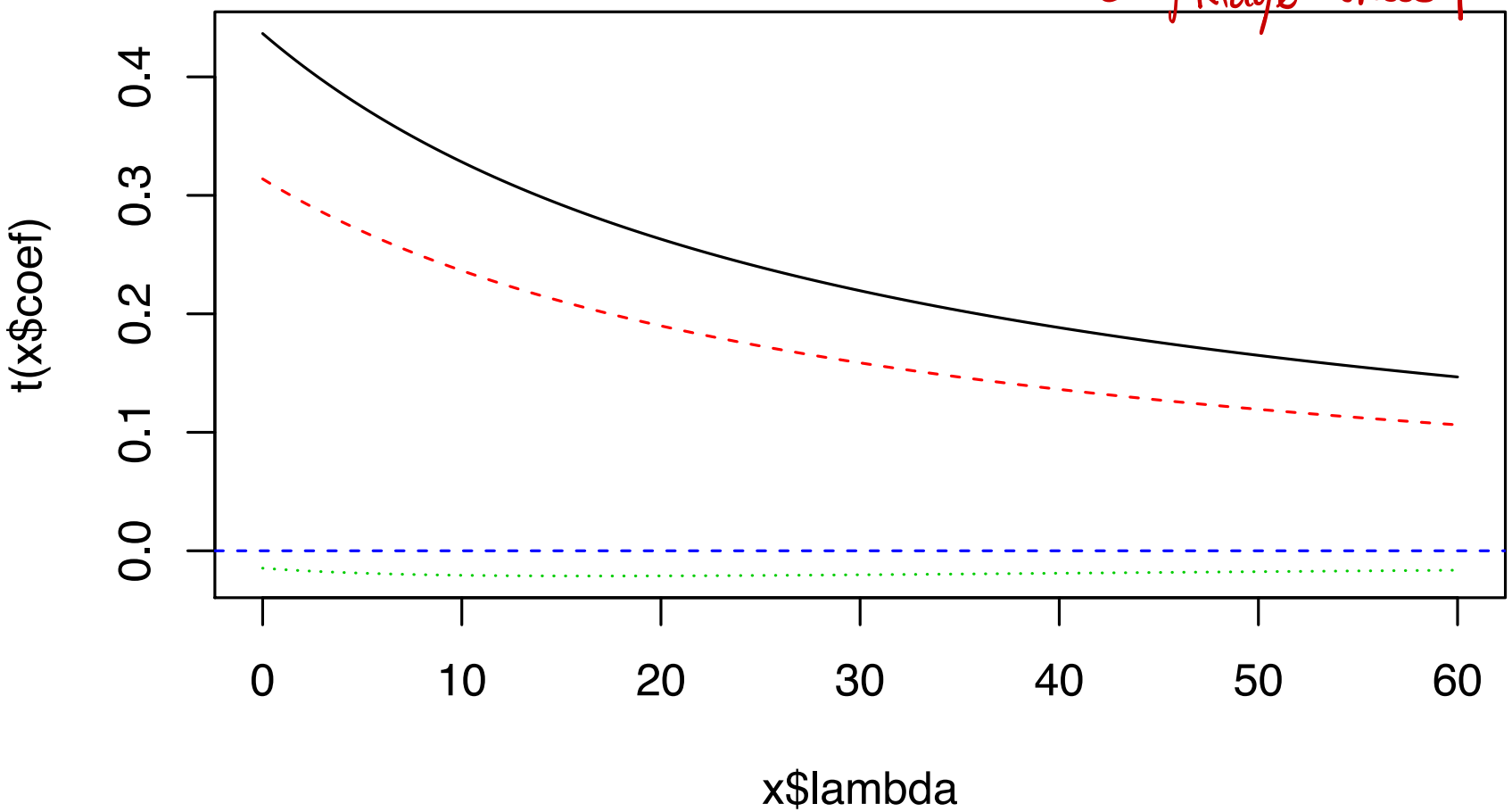
```r
# RLS fit: model with independent covariates
library(MASS)
rls <- lm.ridge(y~ x1 + x2 + x3, lambda = seq(0, 60, 0.01))
summary(rls)
```

```
##         Length Class  Mode
## coef    18003  -none- numeric
## scales      3  -none- numeric
## Inter       1  -none- numeric
## lambda   6001  -none- numeric
## ym          1  -none- numeric
## xm          3  -none- numeric
## GCV      6001  -none- numeric
## kHKB        1  -none- numeric
## kLW         1  -none- numeric
```

Ridge regression: simulation study

```
# RLS fit: model with independent covariates
plot(rls); abline(h=0,lty=2,col="blue")
```

↓ $\hat{\beta}_{\text{Ridge}}$ trace-plot.




```
select(rls)

## modified HKB estimator is 3.334439
## modified L-W estimator is 3.262647
## smallest value of GCV  at 14.26
```

Ridge regression: simulation study

```r
# OLS: model with variables that are correlated
olsc <- lm(y~ x1 + x2 + x3c)
summary(olsc)
```

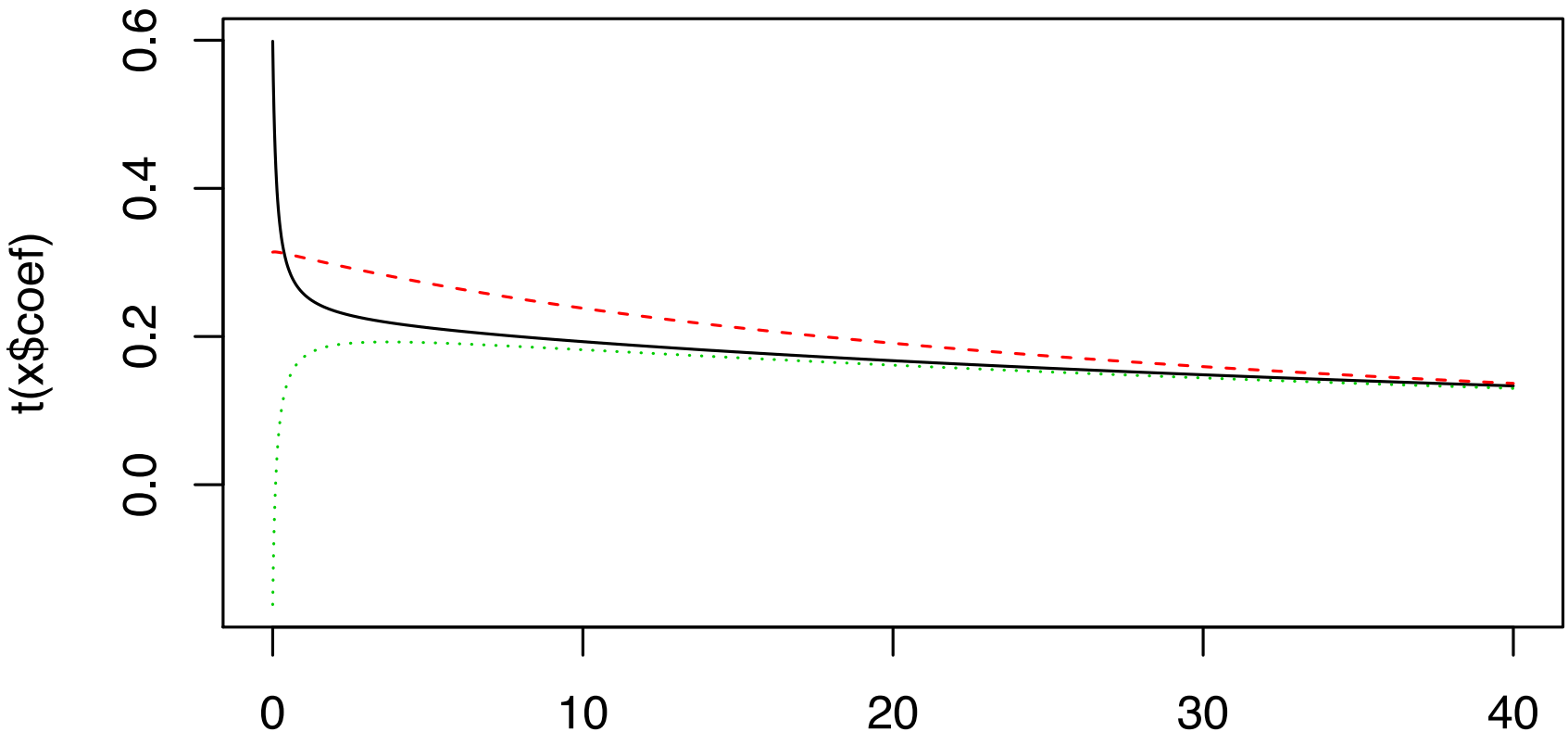
```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3c)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0313 -0.6480 -0.1313  0.7306  1.7187
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.12632    0.58723  -0.215    0.831
## x1           1.94742    6.45883   0.302    0.765
## x2           1.06167    0.60878   1.744    0.093 .
## x3c         -0.05271    0.64662  -0.082    0.936
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9822 on 26 degrees of freedom
## Multiple R-squared:  0.2613, Adjusted R-squared:  0.176
## F-statistic: 3.065 on 3 and 26 DF,  p-value: 0.04564
```

Ridge regression: simulation study

```
# OLS: model with variables that are correlated
rlsc <- lm.ridge(y~ x1 + x2 + x3c,lambda=seq(0,40,0.02))
select(rlsc)
```

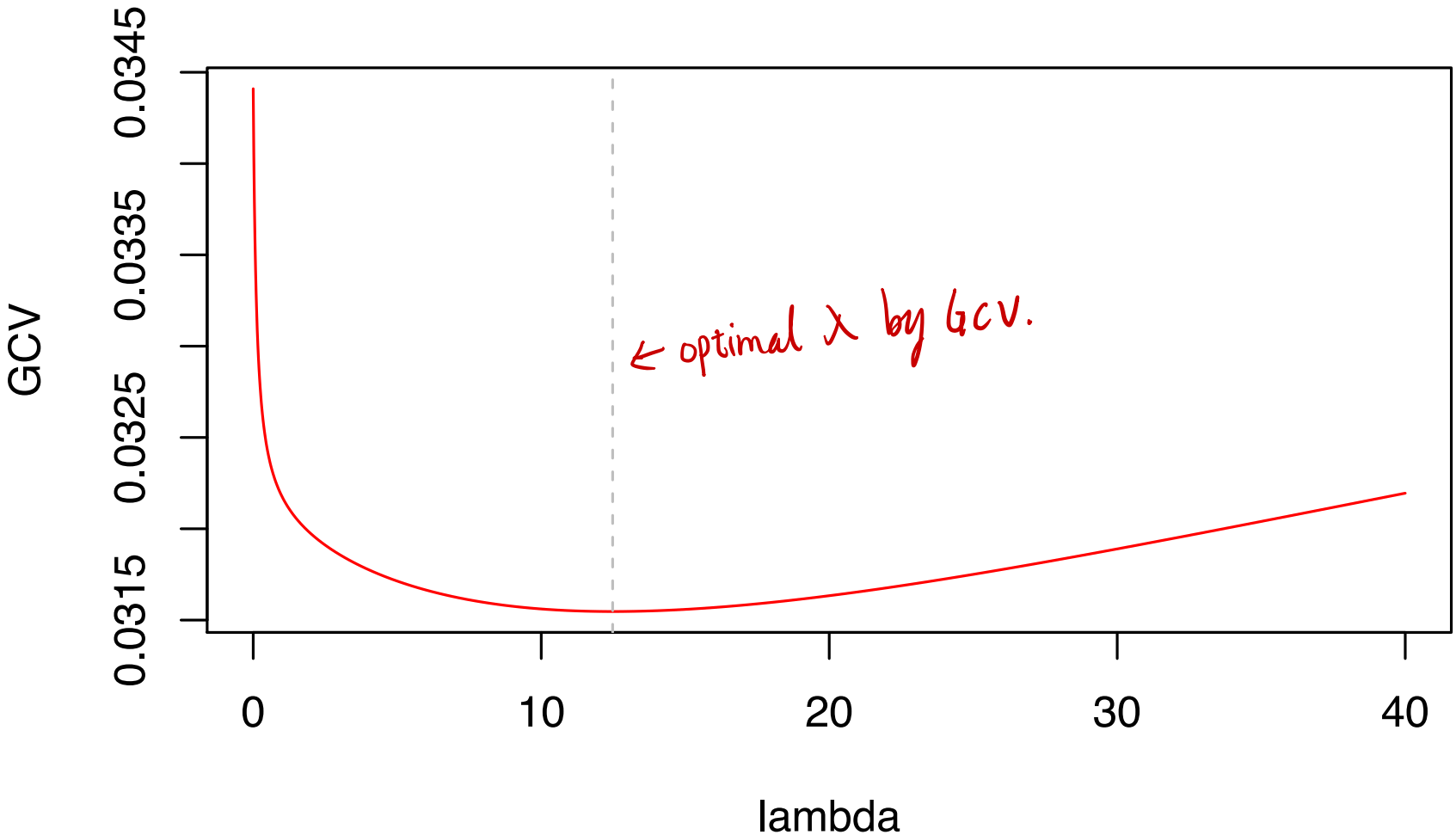
```
## modified HKB estimator is 1.9973
## modified L-W estimator is 3.262647
## smallest value of GCV  at 12.48
```

```
plot(rlsc)
```



Ridge regression: simulation study

```
# OLS: model with variables that are correlated
# find the optimal lambda value by GCV
with(rlsc, plot(lambda,GCV,type="l",col="red"))
abline(v=12.48,lty=2,col="gray")
```



Ridge regression: simulation study

```
# OLS: model with variables that are correlated
# final ridge model
rlsfin = lm.ridge(y~ x1 + x2 + x3c,lambda=12.48)
rlsfin
```

```
##                    x1          x2          x3c
## 0.07558473 0.60435833 0.75903709 0.05755742
```

$\hat{\beta}_{Ridge}$ under $\lambda = \hat{\lambda}_{optimal}$

```
summary(rlsfin)
```

```
##        Length Class  Mode
## coef   3      -none- numeric
## scales 3      -none- numeric
## Inter  1      -none- numeric
## lambda 1      -none- numeric
## ym     1      -none- numeric
## xm     3      -none- numeric
## GCV    1      -none- numeric
## kHKB   1      -none- numeric
## kLW    1      -none- numeric
```

Ridge regression: simulation study

```
# Compare MSPE
test = expand.grid(x1 = seq(.05,.95,.1), x2 = seq(.05,.95,.1),
                   x3=seq(.05,.95,.1))
mu = test$x1 + test$x2
test$x3c = 10*test$x1 + test$x3
pred.ols = predict(ols,newdata=test)   # y ~ x1 + x2+ x3
pred.olsc = predict(olsc,newdata=test) # y ~ x1 + x2 + x3c
crls=coef(rlsfin)
pred.ridge = crls[1]+crls[2]*test[,1]+crls[3]*test[,2]+crls[4]*test[,4]
MSPE.ols <- sum((pred.ols - mu)^2)/length(mu)
MSPE.olsc <- sum((pred.olsc - mu)^2)/length(mu)
MSPE.ridge <- sum((pred.ridge - mu)^2)/length(mu)
MSPE.ols
```

create test data

MSPE = mean square predictim Error.

```
## [1] 0.0229161
```

```
MSPE.olsc
```

```
## [1] 0.0229161
```

```
MSPE.ridge
```

```
## [1] 0.01318808
```

small MSPE

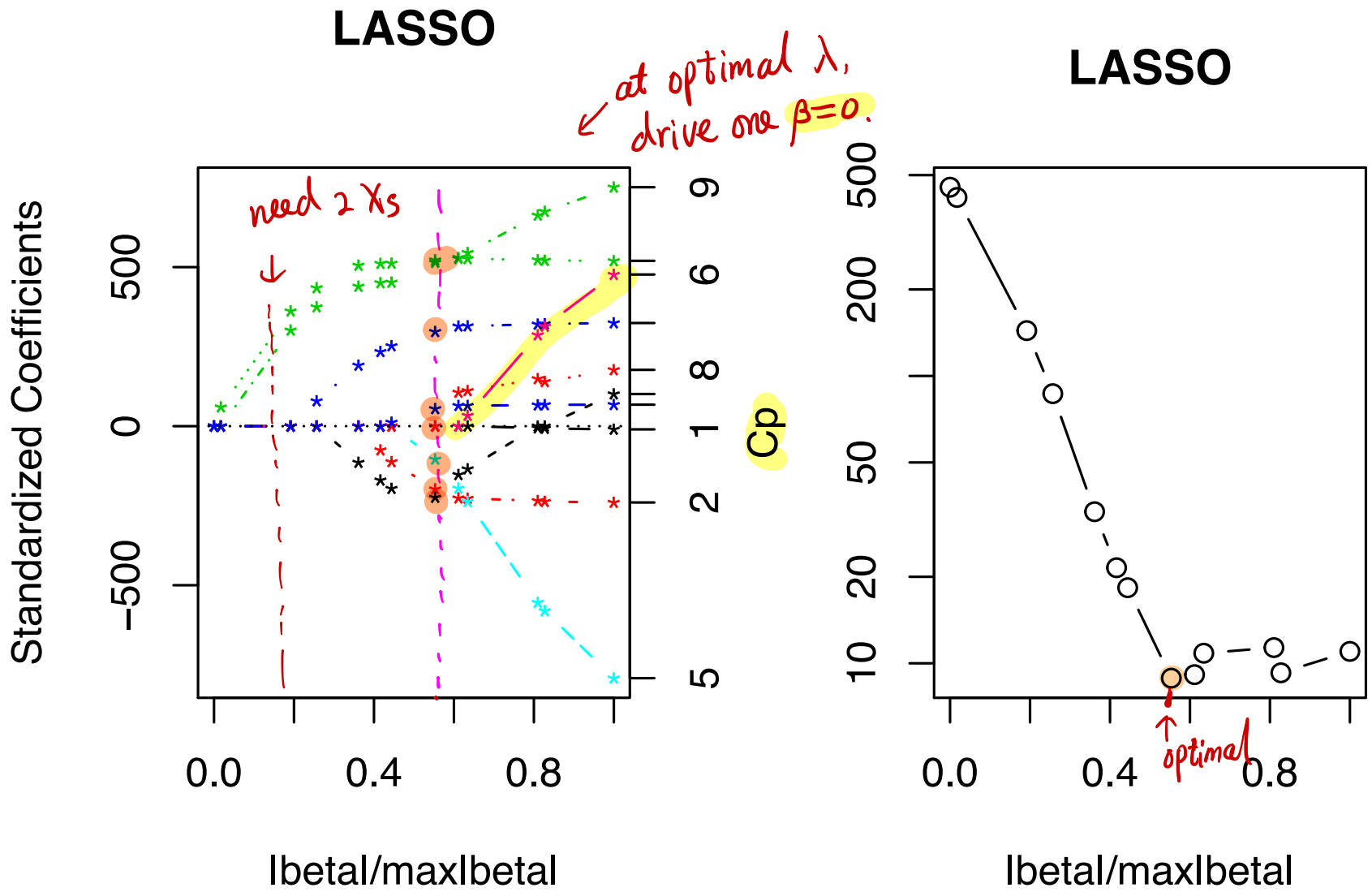
LASSO: data exmaple

```
library(lars)
data(diabetes, package = "lars")# data set from the "LARS paper"
m.lass <- with(diabetes, lars(x,y, trace = TRUE)) ## Lasso and show steps
```

```
## LASSO sequence
## Computing X'X .....
## LARS Step 1 :     Variable 3     added
## LARS Step 2 :     Variable 9     added
## LARS Step 3 :     Variable 4     added
## LARS Step 4 :     Variable 7     added
## LARS Step 5 :     Variable 2     added
## LARS Step 6 :     Variable 10    added
## LARS Step 7 :     Variable 5     added
## LARS Step 8 :     Variable 8     added
## LARS Step 9 :     Variable 6     added
## LARS Step 10 :    Variable 1     added
## Lasso Step 11 :   Variable 7     dropped
## LARS Step 12 :    Variable 7     added
## Computing residuals, RSS etc .....
```

LASSO: data exmaple

```
par(mfrow=c(1,2))
plot(m.lass, breaks=FALSE, mar4=2.2) ;
plot(m.lass, plottype = "Cp", log = "y")
```



Logistic Regression model
Reading: CH6 by J.Q. Fan

Define Odds and OR

Condition	disease	no disease
exposed	a	b
unexposed	c	d

- Odds. The odds of an event $\omega = \frac{p}{1-p}$
  - e.g. P(success)=p, P(failure)=1-p=q, Odds(success)=p/q, odds(failure)=q/p
  - $\omega \in (0, \infty)$
  - $p = 0.5 \leftrightarrow \omega = 1$
  - if the odds of success is ω, then the odds of failure is $1/\omega$
  - if the odds of success is ω, then the probability of success is

$$p = \frac{\omega}{1+\omega}$$

- Odds Ratio (OR) is ratio of two odds.
  - e.g. from above table, OR is the odds of disease among exposed individuals divided by the odds of disease among unexposed.

$$OR = \frac{p_1/(1-p_1)}{p_2/(1-p_2)} = \frac{P(D|E)/(1-P(D))}{P(D|UE)/(1-P(D|UE))} = \frac{a/b}{c/d} = \frac{ad}{bc}$$

Define RR

Condition	disease	no disease
exposed	a	b
unexposed	c	d

- Relative Risk (RR):=p_1/p_2, the probability of one event in one condition relative to the probablity of same event in another condition.
  - e.g. the relative risk of disease associated with condition exposure

$$RR = \frac{a/(a+b)}{c/(c+d)}$$

- For rare events, $OR = \frac{p_1/(1-p_1)}{p_2/(1-p_2)}$, when p_1, p_2 are both small, $OR \approx RR$

$1-p_1 \approx 1-p_2$

$\Rightarrow OR \approx p_1/p_2 = RR.$

Binary response

- Recall in linear regression with normal error model

$$Y_i = X_i\beta + \epsilon_i, E(Y_i|X_i) = \mu_i = X_i\beta$$

  - continous response.
  - errors assumed to be normal, so $Y_i \sim N(X_i\beta, \sigma^2)$
  - variance of errors don't dependent on mean.
- Binary outcome
  - Response (Y) is either 0 or 1
  - E(Y) is restricted in [0,1] interval
  - The idea of "normal error" does not apply here.
- Now consider binary depdent variable. Let Y_i be the response for the i-th observation where

$$Y_i = \begin{cases} 1, & \text{with prob}=\pi(x_i) \\ 0, & \text{with prob}=1-\pi(x_i) \end{cases}$$

then

$$Y_i \sim Bernoulli(\pi(x_i)) = Bin(1, \pi(x_i)), E(Y_i|x_i) = \pi(x_i)$$

where x_i is the level of the predictor of observation i.

Link function

- For the binary response,

$$E(Y_i|x_i) = \mu(Y_i|x_i) = \pi(x_i)$$

  - one possible model to model the mean in this case would be

$$\pi(x_i) = \beta_0 + \beta_1 x_i, \; L.H.S \in [0,1], \; R.H.S \in R = (-\infty, \infty)$$

- To keep the linear type predictor, we need a link function $g(\cdot)$ to transform the mean to a linear function,

$$g(\pi) = g(\mu) = \beta_0 + \beta_1 x_i$$

  - This is equivalent to find a function g to transform the [0,1] to $(-\infty, \infty)$
    - Logit: $g(\pi) = \log \frac{\pi}{1-\pi} = logit(\pi) = \eta$
    - Probit: $g(\pi) = \Phi^{-1}(\pi)$
    - cloglog (complementary log-log link): $g(\pi) = \log(-\log(1-\pi))$

Logistic regression

- Instead of looking at things on the probability scale, let's look at things on the log odds (η) scale.

$$g(\pi) = \log \frac{\pi}{1-\pi} = \log \omega = \mathit{logit}(\pi) = \eta$$

ω = odds = $\frac{p}{1-p}$

η = log(odds) = $\log \frac{p}{1-p}$

- Transformating back gives

$$\frac{\pi}{1-\pi} = \omega = e^{\eta} = \exp\{\beta_0 + \beta_1 x\}$$

and

$$E(Y|x) = \pi = \frac{\omega}{1+\omega} = \text{Sigmoid}(\eta) = \frac{e^{\eta}}{1+e^{\eta}} = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}$$

- It is easy to extend this to multiple predictors, just set

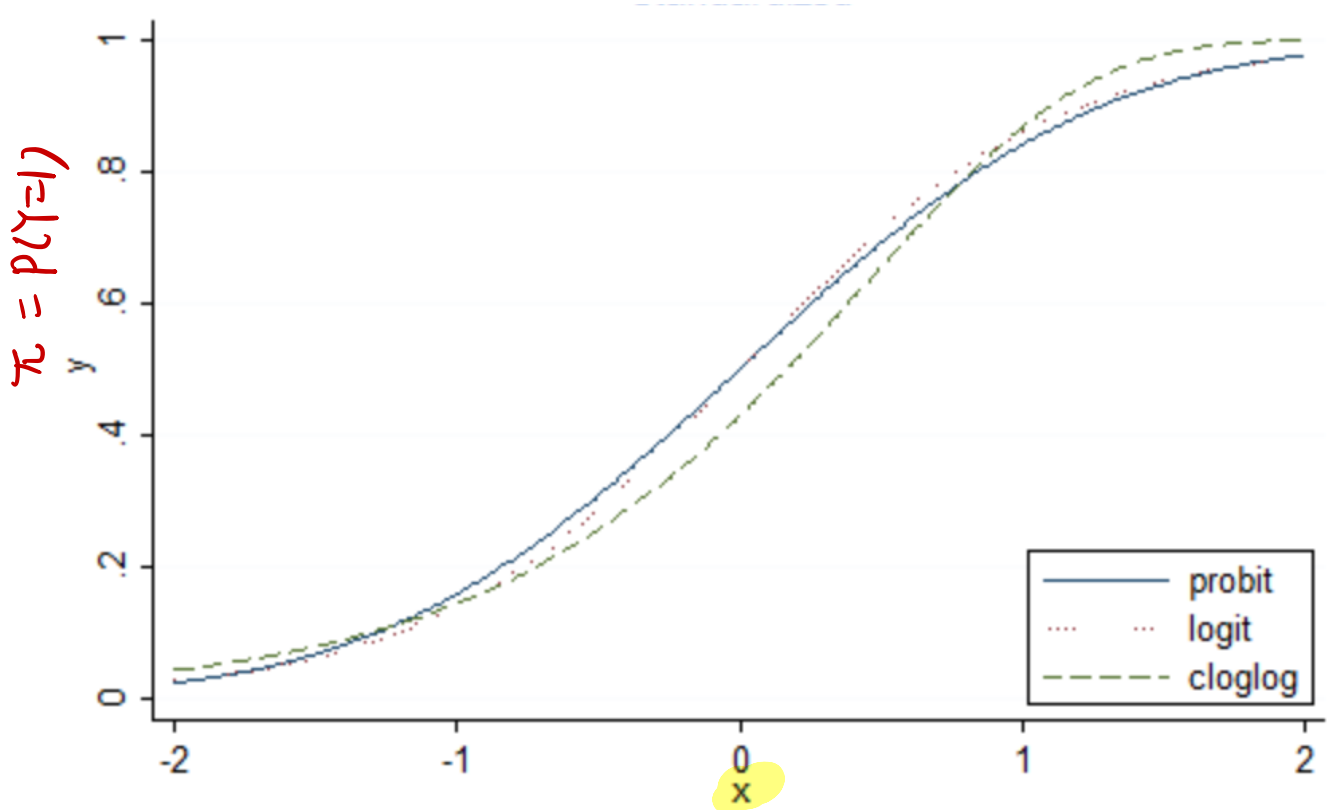
$$\mathit{logit}(\pi) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$$
$$= X\beta$$

giving

Sigmoid(t) = $\frac{e^t}{1+e^t}$

$$\omega = \frac{\pi}{1-\pi} = e^{\eta} = e^{X\beta} \rightarrow \quad \pi = \frac{\omega}{1-\omega} = \frac{e^{X\beta}}{1+e^{X\beta}}$$

Logistic regression



Effect of change X on log odds (η), odds (ω) and $\pi = \mu$

In single predictor case, consider what happens as x goes to $x + \delta_x$

$$\eta(x + \delta_x) = \beta_0 + \beta_1(x + \delta_x) = (\beta_0 + \beta_1 x) + \beta_1\delta_x = \eta(x) + \beta_1\delta_x$$

- So the **log odds** work the same way as linear regression. Changing x by one leads to a change in log odds of β_1

η = log odds
ω = odds

$$\omega(x + \delta_x) = e^{\beta_0+\beta_1(x+\delta_x)} = \omega(x)e^{\beta_1\delta_x}$$

- so the changing x has a multiplicative effect on the odds. Increasing x by 1 leads to multiplying the odds by e^{β_1}. Increasing x by another δ_x leads to another multiplication of $e^{\beta_1\delta_x}$
- Above discussion also imply the sign of β_1 indicate whether η and π increases ($\beta_1 > 0$) or decreases ($\beta_1 < 0$) as increases.
- For π there is not a nice relationship as $\pi(x)$ has an S shape.

$$\pi(x + \delta_x) = \frac{e^{\beta_0+\beta_1(x+\delta_x)}}{1 + e^{\beta_0+\beta_1 x}} = \pi(x) + ?$$

hard to find.

Effect of change one covaraite on log odds (η), odds (ω)

- In the case of multiple predictors, we need to be a bit more careful. If there are no interaction terms in the model,

$$\eta = \mathit{logit}(\pi) = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

  - the previous ideas go through if we fix all predictors but only allow one to vary.
  - The log odds satisfy

$$\eta(X_1 + \delta_x, X_2) = \beta_0 + \beta_1(X_1 + \delta_x) + \beta_2 X_2 = \eta(X_1, X_2) + \beta_1 \delta_x$$

  - also imply that the odds satisfy

$$\omega(X_1 + \delta_x, X_2) = \exp\{\beta_0 + \beta_1(X_1 + \delta_x) + \beta_2 X_2\} = \omega(X_1, X_2) e^{\beta_1 \delta_x}$$

Effect of change one covaraite on log odds (η), odds (ω)

- In the case of the interaction model

$$\eta = \mathit{logit}(\pi) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2$$

- The log odds satisfy

$$\begin{aligned}\eta(X_1 + \delta_x, X_2) &= \beta_0 + \beta_1(X_1 + \delta_x) + \beta_2 X_2 + \beta_{12}(X_1 + \delta_x)X_2 \\ &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \beta_1 \delta_x + \beta_{12}\delta_x X_2 \\ &= \eta(X_1, X_2) + \delta_x(\beta_1 + \beta_{12} X_2)\end{aligned}$$

  - also imply that the odds satisfy

$$\omega(X_1 + \delta_x, X_2) = \omega(X_1, X_2) e^{\delta_x(\beta_1 + \beta_{12} X_2)}$$

After Lecture This Week

Practice problems

- Review all the slides
- Try all the R example in slides.

Topics for next week:

- Logistic regression model (Reading: CH6 by J.Q. Fan)
  - Estimation of β in the model
  - Inference of β in the model
  - data example