


Cardiovascular Diseases and Life Expectancy in Adults With Type 2 Diabetes: A Korean National Sample Cohort Study

Yu Mi Kang, Yun Kyung Cho, Seung Eun Lee, Joong-Yeol Park, Woo Je Lee, Ye-Jee Kim, Chang Hee Jung 

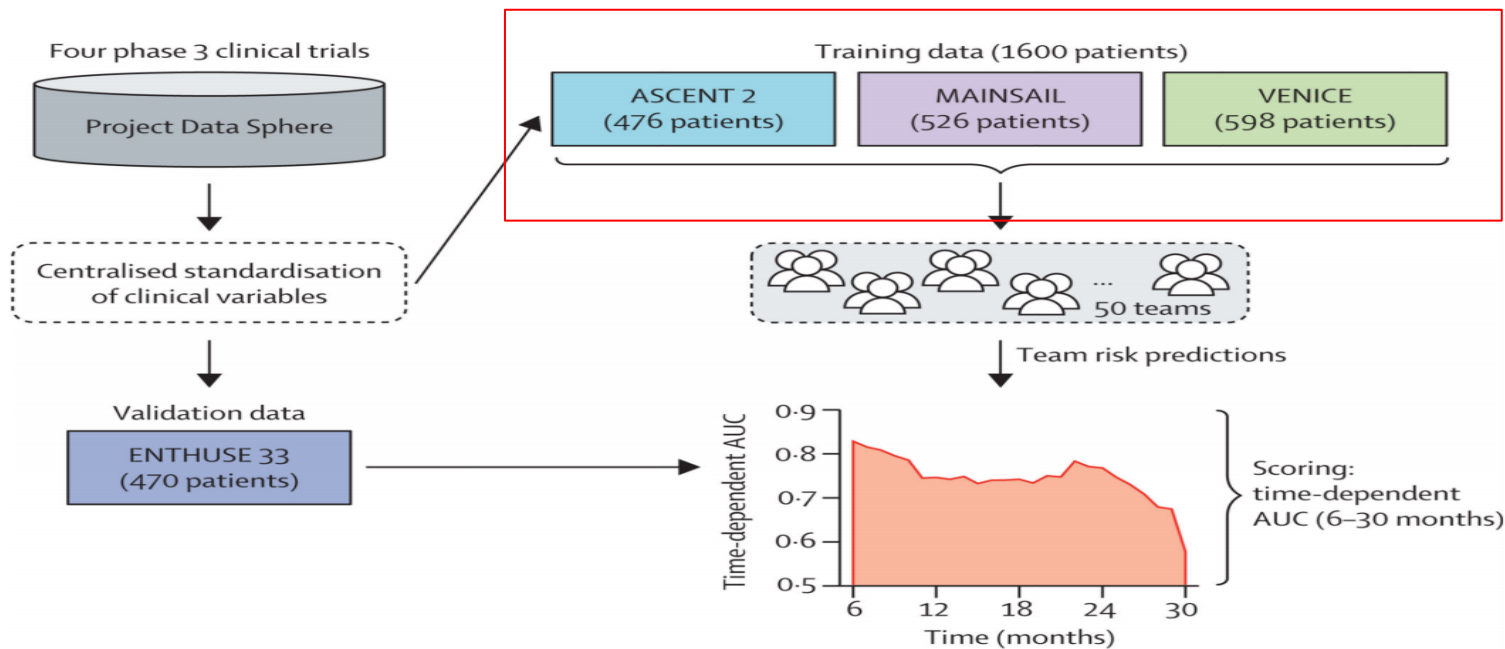
The Journal of Clinical Endocrinology & Metabolism, Volume 102, Issue 9, 1 September 2017, Pages 3443–3451, <https://doi.org/10.1210/jc.2017-00643>

Published: 30 June 2017 **Article history** ▼

전이성 거세 저항성 전립선암 예측

50개 팀이 참가

Data



ASCENT2, MAINSAIL, VENICE데이터가 훈련용

ENTHUSE 33은 validation을 위해 공개하지 않음.

Data

ASCENT2라는 임상 시험으로 부터 476 명.

MAINSAIL 임상 시험으로부터 526 명

VENICE라는 임상 시험으로부터 598명

ENTHUSE 33 trail docetaxel, placebo, 470 대상

인구학적 특징, 실험결과, Medical history, 병변 부위, previous treatment 등의

150개의 Clinical 변수

Summary

1. 성능의 척도

- IAUC (time dependent AUC)

2. 가장 좋은 성능의 모델

- 양상블 Penalized cox regression

3. 해당 모델의 의의

- Immune biomarker 변수 + 신장기능 변수의 상호작용

- immune biomarker 변수 + 간장변수의 상호작용

Issue

1. IAUC(time dependent AUC)

2. Data curation 과정

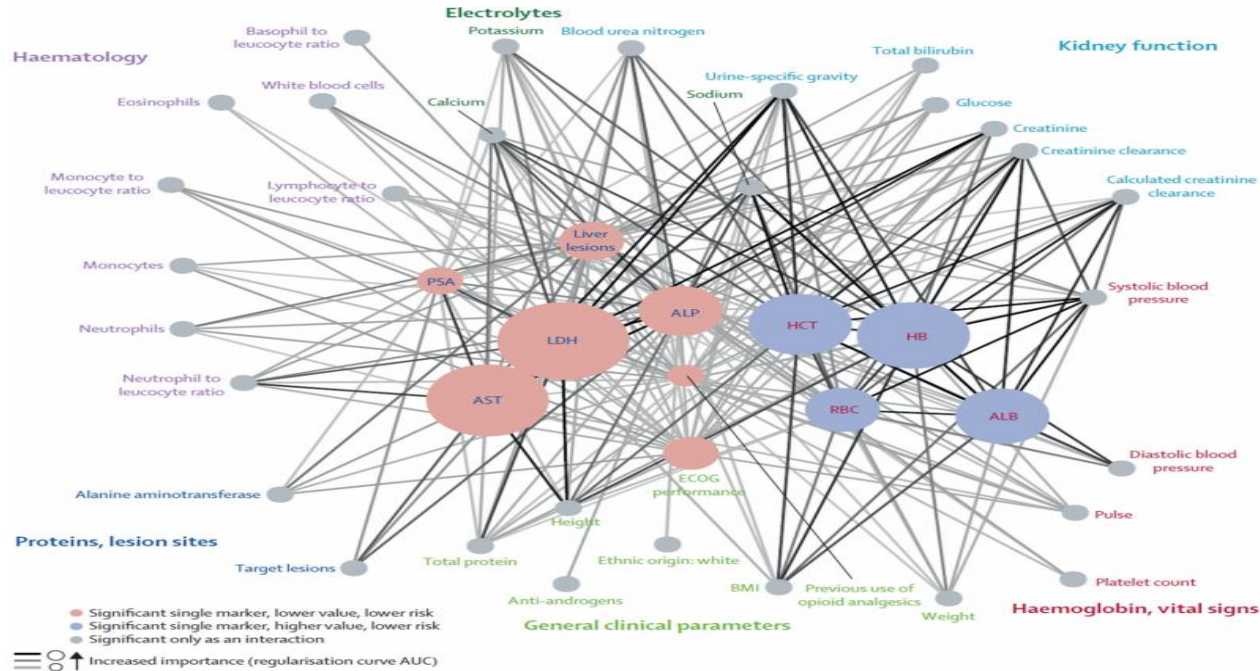
-PCA를 통한 데이터 셋의 유사도 검증

3. 변수간의 Network를 활용한 변수 선택 + 상호작용 변수 확인

4. Bootstrapping

5. Penalized regression

Issue



This importance was calculated as the area under the curve (AUC) of the penalised model predictors, as a function of penalisation parameter λ

Bootstrap

Estimating Regression Coefficients using Weighted Bootstrap with Probability

NORAZAN M. R.¹, HABSHAH MIDI² AND A. H. M. R. IMON³

¹Faculty of Computer and Mathematical Sciences,
University Technology MARA,
40450 Shah Alam, Selangor,
MALAYSIA

²Laboratory of Applied and Computational Statistics,
Institute for Mathematical Research,
University Putra Malaysia,
43400 Serdang, Selangor,
MALAYSIA

³Department of Mathematical Sciences,
Ball State University,
Muncie, IN 47306,
U.S.A.

Email: ¹norazan@tmsk.uitm.edu.my, ²habshahmidi@gmail.com, ³imon_ru@yahoo.com

Bootstrap

Original Data에서 복원 추출을 ->

예) 데이터 분포가 고르지 않는 경

-전이성 암 100명, 비전이성 암 1000명

예) Over-fitting 방지(Bagging)

-여러개의 모델을 만들어서 model ensemble(평균)

-Voting

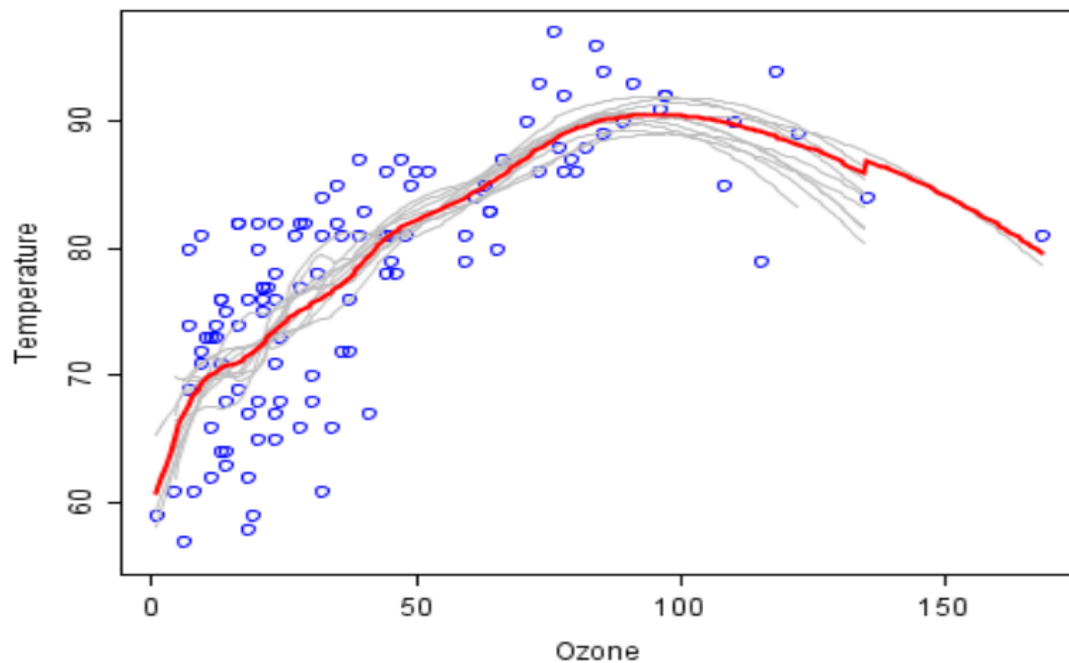
예) 회귀 계수, 분산, 신뢰구간 추정

같은 분산 σ^2 를 가지고 있는 서로 독립인 통계량 Z_1, \dots, Z_n 들의 평균, 즉 \bar{Z} 의 분산은 $\frac{\sigma^2}{n}$ 이란 것을 알고있다. 물론 독립에 등분산은 현실에선 많지 않지만, 우선 간단화 해보자면, **평균을 취하는 것은 분산을 감소시킨다.** 따라서 분산을 감소시키는, 그럼으로써 궁극적으로는 예측력을 높이는 자연스런 결론은 **모집단의 많은 데이터 셋에 적합**하여 여러개의 예측 모델을 만들고, 그 모델의 **예측결과를 평균내는것**일 것이다.

그러나 당연히 현실에서는 모집단에서 무수히 많은 데이터셋을 추출하는 것이 불가능하다. 대신, 5장에서 다루었던, 주어진 training set에서 무수히 많은 B번의 반복추출로 무수히 많은 data set을 만들어내는, Bootstrap을 진행하여 B개의 여러 모델을 만들수 있다. 이들을 평균냄으로써, 위의 논의를 따라가는 최종 결과물을 얻을 수 있을 것이다. 이것이 **Bootstrap**하여 합친다(**aggregating**한다), 즉 Bagging이다.

Bootstrap

예) Over-fitting 방지(Bagging)



Bootstrap

단점

Outlier에 많은 영향을 받음.

Resampling을 하다보면, Outlier가 많이 뽑힐 수 있음

->

Bootstrap을 통해 계산된 추정치들(분산, 신뢰구간)이 안좋아질 수 있음.

이를 방지하기 위함

-Diagnostic-Before Bootstrap, Weighted Bootstrap with Probability(WBP)

Bootstrap

Resampling residuals :

Model의 shape는 어느 정도 신뢰하나, 잔차의 분포에 대해서는 어떤 가정도 하지 않는다.

Resampling cases :

Model이 shape, 그리고 잔차의 분포에 대해서도 어떤 가정 그리고 신뢰도 하지 않는다. 각각의 관측값을 독립적으로 볼 뿐이다.

Bootstrap

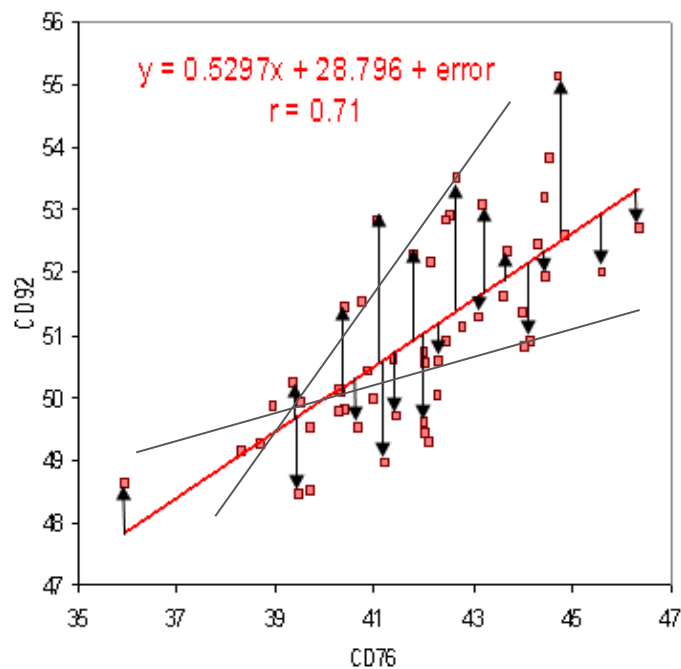
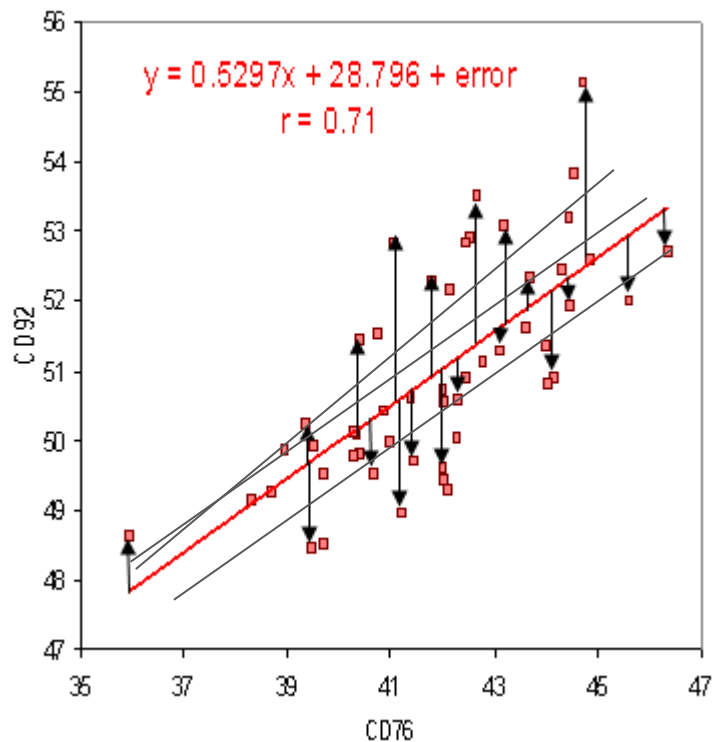
Resampling residuals :

Model의 shape는 어느 정도 신뢰하나, 잔차의 분포에 대해서는 어떤 가정도 하지 않는다.

Resampling cases :

Model의 shape, 그리고 잔차의 분포에 대해서도 어떤 가정 그리고 신뢰도 하지 않는다. 각각의 관측값을 독립적으로 볼 뿐이다. 즉 가장 안전한 방법이다.

Bootstrap



Bootstrap

Which Bootstrap??

Resampling cases가 가장 안전한 방법이나, 항상 쓰는 것은 아니다.

-> Bias-Variance trade off의 문제

1) Resampling cases -> 큰 분산, 작은 편차

2) Resampling residuals -> 적은 분산, 큰 편차

Bootstrap

Which Bootstrap??

Original data set에서 만든 모델이 정확하다면(알 수 없음)

-> Resampling residual이 Good

Original data set에서 만든 모델이 부정확하다면(알 수 없음)

-> Resampling cases가 Good

Bootstrap_종류

1. Residual resampling(fixed-x resampling)

-전체 데이터 셋을 이용하여 회귀 계수와 잔차를 구한다.

-잔차를 복원 Random sampling 한다.

```
y1 <- b[1] + b[2]*x + r1
```

```
m1 <- lm(y1~x)
```

```
b1[i,] <- m1$coeff
```

-이를 반복하여 회귀 계수의 집합을 구한다.

Bootstrap_종류

1. Residual resampling(fixed-x resampling)

```
m <- lm(y~x)
b <- m$coeff
r <- m$res
B <- 500
b1 <- matrix(NA, ncol=2, nrow=B)

for(i in 1:B){
  r1 <- sample(r, length(r), rep=T)
  y1 <- b[1] + b[2]*x + r1
  m1 <- lm(y1~x)
  b1[i,] <- m1$coeff
}
```

Bootstrap_종류

2.Case-resampling

- 우리가 흔히 하는 Bootstrapping
- 데이터를 복원 Random sampling한다.
- 회귀 계수를 추정한다
- 이를 반복하여 회귀 계수의 집합을 구한다.

Bootstrap_Diagnostic-Before Bootstrap

1) Bootstrap을 하기 이전 Outlier제거

2) Outlier는 Robust reweighted least squares(RLS) residuals를 통해 추출

$$w_i = \begin{cases} 0, & \text{if } \text{abs}(r_i) > 2.5 s \\ 1, & \text{otherwise} \end{cases}$$

$$s = 1.48268[1 + \{5 / (n - p)\}] \sqrt{\text{median}(r_i^2)}$$

(n=sample size, p=number of regression coefficients)

3) Residual resampling을 통해 회귀 계수의 집합을 구한다.

Bootstrap_Weighted Bootstrap with Probability

1) Original Data set에서 LMS(least median squares)를 통해 잔차를 구한다

$$2) \quad u_i = \frac{r_i}{\text{MAD}(r_i)} \quad \Rightarrow \quad w_i(u_i) = \frac{\psi(u_i)}{u_i} \quad \Rightarrow \quad p_i = \frac{w_i}{\sum_{j=1}^n w_j}$$

$$\psi_{\text{Hampel}}(u) = \begin{cases} u & , 0 \leq \text{abs}(u) \leq a \\ a \text{ sign}(u), & , a \leq \text{abs}(u) \leq b \\ a(c - \text{abs}(u)) / (c - b) \text{ sign}(u) & , b \leq \text{abs}(u) \leq c \\ 0 & , c \leq \text{abs}(u) \end{cases}$$

$$\text{MAD} = \text{median}(|X_i - \tilde{X}|)$$

Bootstrap_Weighted Bootstrap with Probability

Least median squares method

1) 일반적인 회귀 모형과 달리 이상치나, 회귀 분석의 일반적인 가정들의 위반에 많은 영향을 받지 않는 방법

2) 이상치를 찾을 때 종종 이용 된다.

Bootstrap_Weighted Bootstrap with Probability

Least median squares method

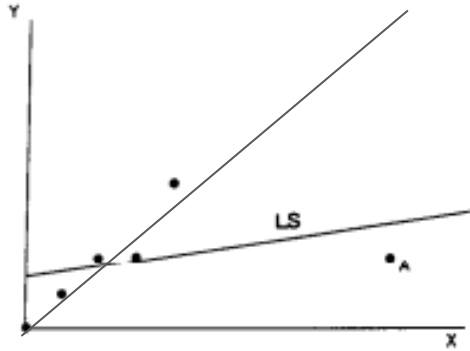


Fig. 1. Effect of outlier on a least-squares line.

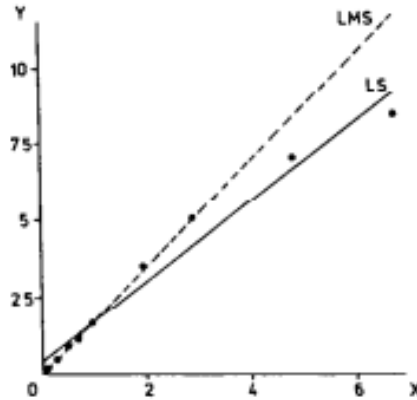


Fig. 2. Detection of deviation from the linear model on a calibration line (data from [10]).

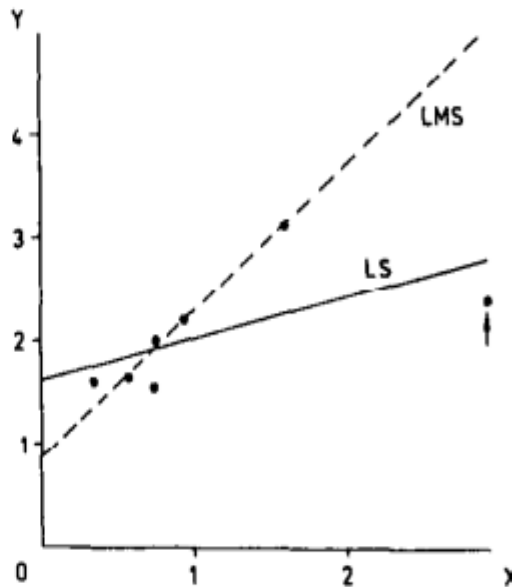
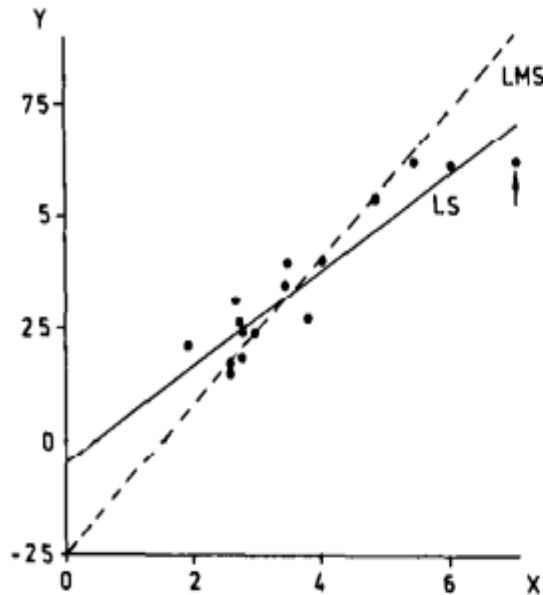
minimize $\text{med}_i(y_i - ax_i - b)^2$

instead of

minimize $\text{sum}_i(y_i - ax_i - b)^2$.

Bootstrap_Weighted Bootstrap with Probability

Least median squares method



minimize $\text{med}_i(y_i - ax_i - b)^2$

instead of

minimize $\text{sum}_i(y_i - ax_i - b)^2$.

Bootstrap_Weighted Bootstrap with Probability

Least median squares method

->일반적인 회귀선(LS)과, LMS의 회귀선이 많이 다르다면,

극단적인 이상치가 존재

->소수의 actual outliers를 찾는 것을 추천.

Bootstrap_Weighted Bootstrap with Probability

3)위에서 구한 잔차와, 각 잔차에 대한 확률값을 이용하여

fixed-x resampling을 한다.

4)회귀 계수의 집합을 구한다.

5)회귀 계수를 구한다.

$$\hat{\beta}_j^* = \frac{1}{5000} \sum_{B=1}^{5000} \hat{\beta}_j^{*B} = \frac{1}{5000} \sum_{B=1}^{5000} \hat{\beta}_j^{*B(-D)}$$

Bootstrap_Weighted Bootstrap with Probability

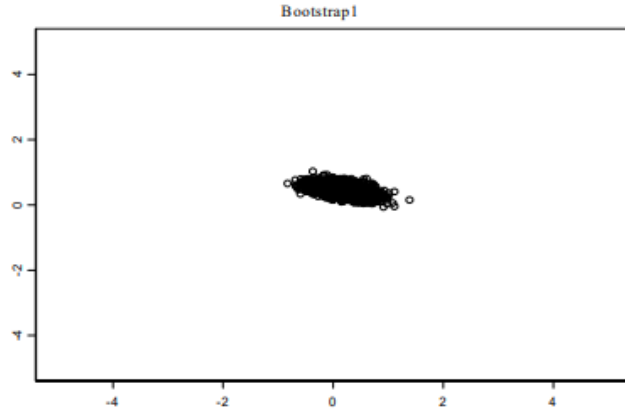


Figure 1: Plots of $(\hat{\beta}_0^{*B} - \hat{\beta}_0)$ versus $(\hat{\beta}_1^{*B} - \hat{\beta}_1)$ for Hawkins-Bradud-Kass data using Bootstrap1 method.

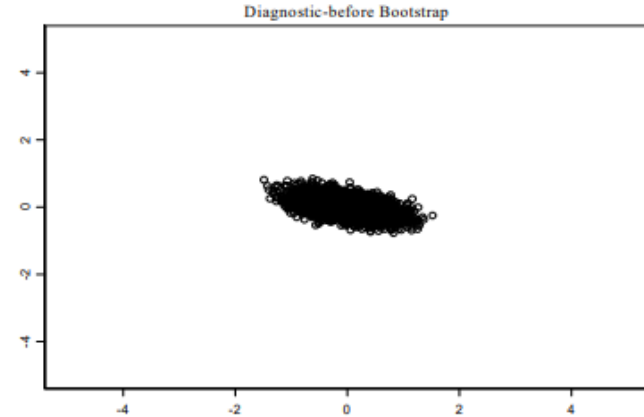


Figure 2: Plots of $(\hat{\beta}_0^{*B} - \hat{\beta}_0)$ versus $(\hat{\beta}_1^{*B} - \hat{\beta}_1)$ for Hawkins-Bradud-Kass using Diagnostic-before Bootstrap method.

Bootstrap_Weighted Bootstrap with Probability

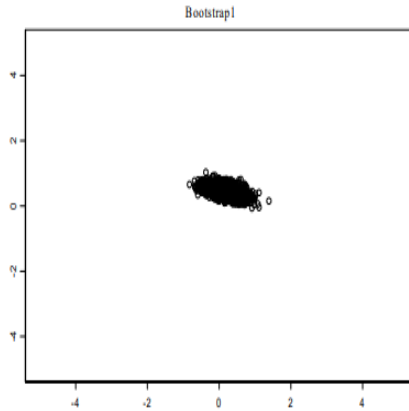


Figure 1: Plots of $(\hat{\beta}_0^{*B} - \hat{\beta}_0)$ versus $(\hat{\beta}_1^{*B} - \hat{\beta}_1)$ for Hawkins-Bradú-Kass data using Bootstrap1 method.

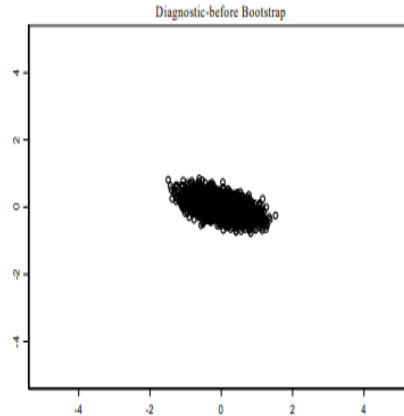


Figure 2: Plots of $(\hat{\beta}_0^{*B} - \hat{\beta}_0)$ versus $(\hat{\beta}_1^{*B} - \hat{\beta}_1)$ for Hawkins-Bradú-Kass using Diagnostic-before Bootstrap method.

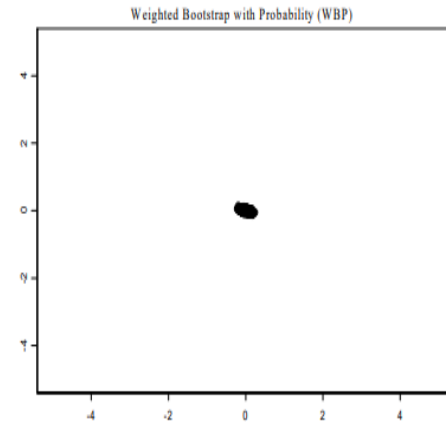


Figure 3: Plots of $(\hat{\beta}_0^{*B} - \hat{\beta}_0)$ versus $(\hat{\beta}_1^{*B} - \hat{\beta}_1)$ for Hawkins-Bradú-Kass using Weighted Bootstrap with Probability method.

Penalized regression

P개의 독립변수에 대한 회귀분석 $\rightarrow 2^P$ 개의 회귀 모형 적합

\rightarrow 너무 많은 시간 소요

\rightarrow **Shrinkage** 방법

Shrinkage란?

- 1) 계수 추정치를 제한
- 2) Training data의 변화에 따른 계수의 변화(편차) 감소 \rightarrow Overfitting 방지
- 3) 불필요한 계수는 0에 가까워짐 \rightarrow 다중공선성 방지

Penalized regression

1. Ridge regression

$$\text{cost} = \sum e_i^2 + \lambda \sum w_i^2 \quad \Rightarrow \quad \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^p \beta_j^2,$$

2. Lasso regression

$$\text{cost} = \sum e_i^2 + \lambda \sum |w_i|$$

3. Elastic regression

$$\text{cost} = \sum e_i^2 + \lambda_1 \sum |w_i| + \lambda_2 \sum w_i^2$$

Penalized regression

Ridge regression

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

2.Ridge의 계수 추정치는 람다이외도, 독립변수의 단위에 의존한다.

예) $Y = Bx$

$Y = 10,20 / x = 1,2 \rightarrow B = 10$

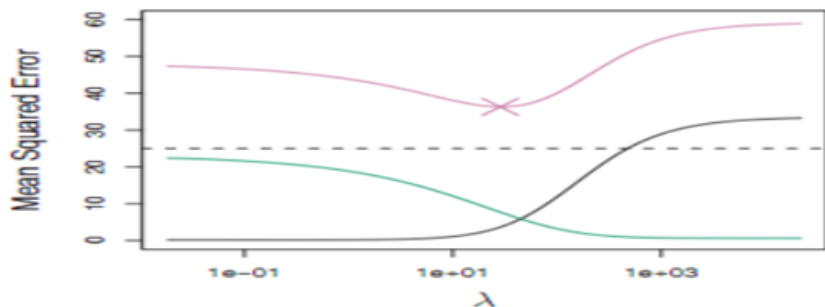
$Y = 10,20 / x = 10,20 \rightarrow B = 1..$ 따라서



$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}},$$

Penalized regression

3. 최적의 람다 (MSE를 최소화)
$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2.$$



Squared bias (black), variance (green),

작은 람다 → 적합의 유연함(Overfitting) → 높은 분산, 작은 편차
→ 학습자료의 작은 변화가 회귀 추정에 큰 변화

큰값 람다 → 적합의 경직(Underfitting) → 적은 분산, 높은 편차

MSE(분산 + 편차)를 최소화 해주는 람다를 계산

Penalized regression

3. 장점

독립변수가 p 개, 관측치의 개수 n 개

$[p > n] \rightarrow$ 최소제곱추정치는 큰 분산

하지만 Ridge의 Bias-Variance trade off로 인해 편차를 증가 \rightarrow 분산의 감소

\rightarrow 최소제곱 추정이 큰 분산을 가지는 경우 좋은 성능

Penalized regression

Lasso regression

$$\text{cost} = \sum e_i^2 + \lambda \sum |w_i|$$

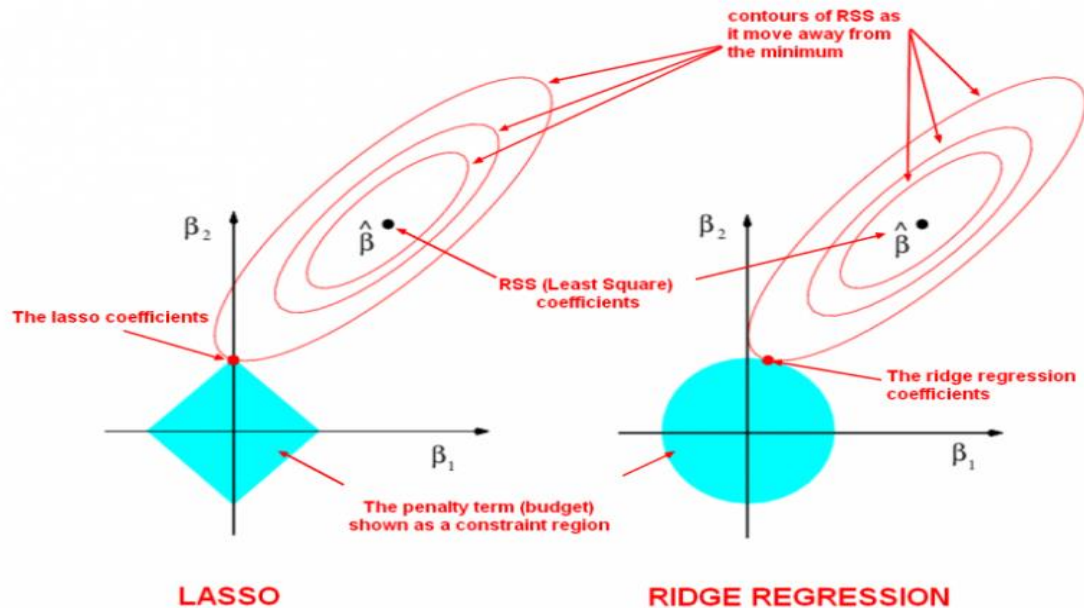
1. Ridge와 거의 유사 하지만 계수가 완전 0에 수렴

2. Ridge와 마찬가지로 Overfitting 방지

...도대체 어떤 차이가?.. Lasso 또한 계수들이 \sum 로 묶여있어 Scaling을 해준다

Penalized regression

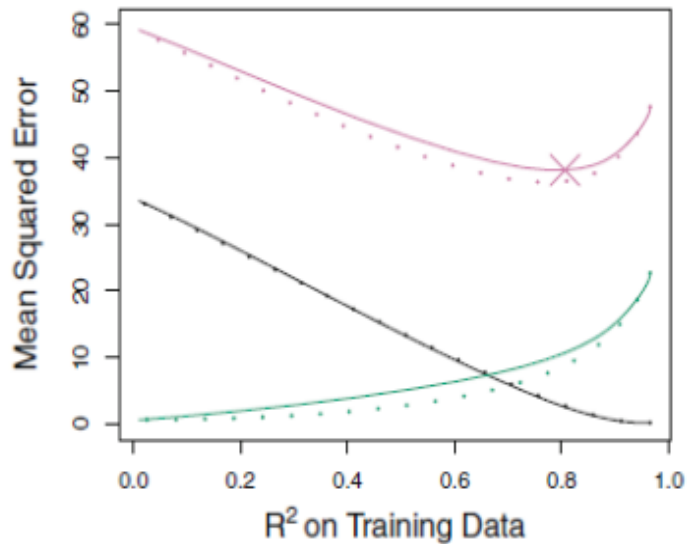
Ridge 모형은 가중치 계수를 한꺼번에 축소시키는데 반해 Lasso 모형은 일부 가중치 계수가 먼저 0으로 수렴하는 특성이 있다.



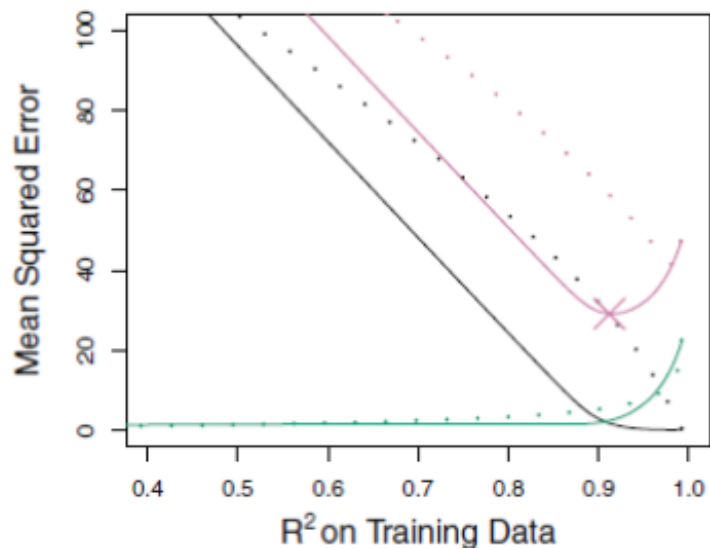
Lasso가 몇몇 계수들을 Absoulte Zero로 보내는 이유

Penalized regression

경우 1)



경우 2)



실선 - Lasso 점선 - Ridge

Penalized regression

Ridge and Lasso

https://www.researchgate.net/publication/321026412_Filter-Wrapper_Combination_and_Embedded_Feature_Selection_for_Gene_Expression_Data

Abstract

Biomedical and bioinformatics data sets are generally large in terms of their number of features - and include redundant and irrelevant features, which affect the effectiveness and efficiency of classification of these data sets. Several different features selection methods have been utilised in various fields, including bioinformatics, to reduce the number of features. This study utilised Filter-Wrapper combination and embedded (LASSO) feature selection methods on both high and low dimensional data sets before classification was performed. The results illustrate that the combination of filter and wrapper feature selection to create a hybrid form of feature selection provides better performance than using filter only. In addition, LASSO performed better on high dimensional data.