Resampling and regularization Homework

Jiachen Feng

2021/2/12

5.8

(a)

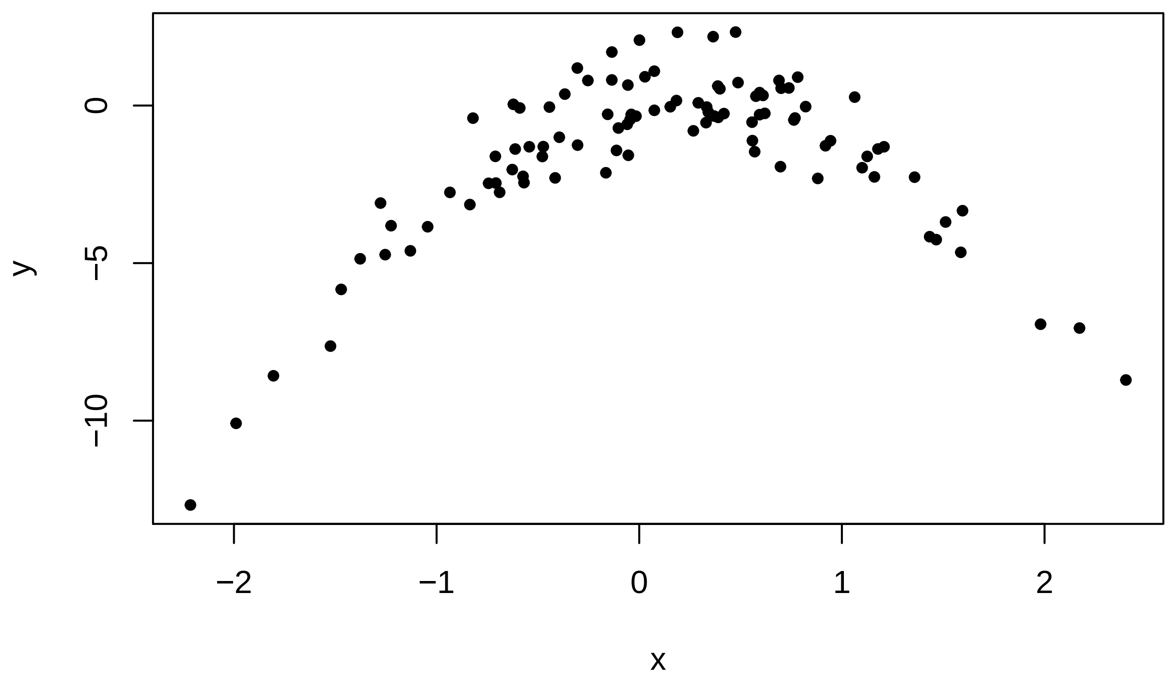
```r
set.seed(1)
x <- rnorm(100)
y <- x-2*x^2+rnorm(100)
```

n is 100, p is 2.

$$y = x - x^2 + \epsilon$$

(b)

```r
plot(x,y,pch=20)
```



The scatterplot is like a quadratic function.

(c)

```r
library(boot)
set.seed(100)
data <- data.frame(x,y)
mod1 <- glm(y~x)
cv.glm(data,mod1)$delta[1]
```

```
## [1] 7.288162
```

```r
mod2 <- glm(y~poly(x,2))
cv.glm(data,mod2)$delta[1]
```

```
## [1] 0.9374236
```

```r
mod3 <- glm(y~poly(x,3))
cv.glm(data,mod3)$delta[1]
```

```
## [1] 0.9566218
```

```r
mod4 <- glm(y~poly(x,4))
cv.glm(data,mod4)$delta[1]
```

```
## [1] 0.9539049
```

(d)

```r
set.seed(200)
mod1 <- glm(y~x)
cv.glm(data,mod1)$delta[1]
```

```
## [1] 7.288162
```

```r
mod2 <- glm(y~poly(x,2))
cv.glm(data,mod2)$delta[1]
```

```
## [1] 0.9374236
```

```r
mod3 <- glm(y~poly(x,3))
cv.glm(data,mod3)$delta[1]
```

```
## [1] 0.9566218
```

```r
mod4 <- glm(y~poly(x,4))
cv.glm(data,mod4)$delta[1]
```

```
## [1] 0.9539049
```

Same.

(e) model 2. The function is quadratic, so that this model fits best.

(f)

```r
summary(mod1)
```

```
##
## Call:
## glm(formula = y ~ x)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -9.5161  -0.6800   0.6812   1.5491   3.8183
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.6254     0.2619  -6.205 1.31e-08 ***
## x             0.6925     0.2909   2.380   0.0192 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6.760719)
##
##     Null deviance: 700.85  on 99  degrees of freedom
## Residual deviance: 662.55  on 98  degrees of freedom
## AIC: 478.88
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(mod2)
```

```
##
## Call:
## glm(formula = y ~ poly(x, 2))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9650  -0.6254  -0.1288   0.5803   2.2700
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.5500     0.0958  -16.18  < 2e-16 ***
## poly(x, 2)1   6.1888     0.9580    6.46 4.18e-09 ***
## poly(x, 2)2 -23.9483     0.9580  -25.00  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.9178258)
##
##     Null deviance: 700.852  on 99  degrees of freedom
## Residual deviance:  89.029  on 97  degrees of freedom
## AIC: 280.17
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(mod3)
```

```
##
## Call:
## glm(formula = y ~ poly(x, 3))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9765  -0.6302  -0.1227   0.5545   2.2843
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.55002    0.09626 -16.102  < 2e-16 ***
## poly(x, 3)1   6.18883    0.96263   6.429 4.97e-09 ***
```

```
## poly(x, 3)2 -23.94830    0.96263 -24.878  < 2e-16 ***
## poly(x, 3)3   0.26411    0.96263   0.274    0.784
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.9266599)
##
##     Null deviance: 700.852  on 99  degrees of freedom
## Residual deviance:  88.959  on 96  degrees of freedom
## AIC: 282.09
##
## Number of Fisher Scoring iterations: 2
```

```r
summary(mod4)
```

```
##
## Call:
## glm(formula = y ~ poly(x, 4))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0550  -0.6212  -0.1567   0.5952   2.2267
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.55002    0.09591 -16.162  < 2e-16 ***
## poly(x, 4)1   6.18883    0.95905   6.453 4.59e-09 ***
## poly(x, 4)2 -23.94830    0.95905 -24.971  < 2e-16 ***
## poly(x, 4)3   0.26411    0.95905   0.275    0.784
## poly(x, 4)4   1.25710    0.95905   1.311    0.193
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.9197797)
##
##     Null deviance: 700.852  on 99  degrees of freedom
## Residual deviance:  87.379  on 95  degrees of freedom
## AIC: 282.3
##
## Number of Fisher Scoring iterations: 2
```

All of the four models are statistically significant.

6.2

(a)

i. Wrong.
ii. Wrong.
iii. Right.
iv. Wrong. It is less flexible because the complexity of lasso is related to λ.

(b) Ridge regression is less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

(c) Non-linear methods are more flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

6.10

(a)

```r
set.seed(1000)
mat <- matrix(rnorm(1000 * 20), 1000, 20)
p <- rnorm(20)
p[1] <- 0
p[2] <- 0
p[3] <- 0
epsilon <- rnorm(1000)
y <- mat %*% p + epsilon
```

(b)

```r
set.seed(1000)
mat1 <- matrix(rnorm(100 * 20), 100, 20)
p1 <- rnorm(20)
p1[1] <- 0
p1[2] <- 0
p1[3] <- 0
epsilon1 <- rnorm(100)
train <- mat1 %*% p1 + epsilon1

mat2 <- matrix(rnorm(900 * 20), 900, 20)
p2 <- rnorm(20)
p2[1] <- 0
p2[2] <- 0
p2[3] <- 0
epsilon2 <- rnorm(900)
test <- mat2 %*% p2 + epsilon2
```

(c)