# Homework 3

Zhitian Liu

2021/2/13

## 5.8

We will now perform cross-validation on a simulated data set.

### (a)

Generate a simulated data set as follows:

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

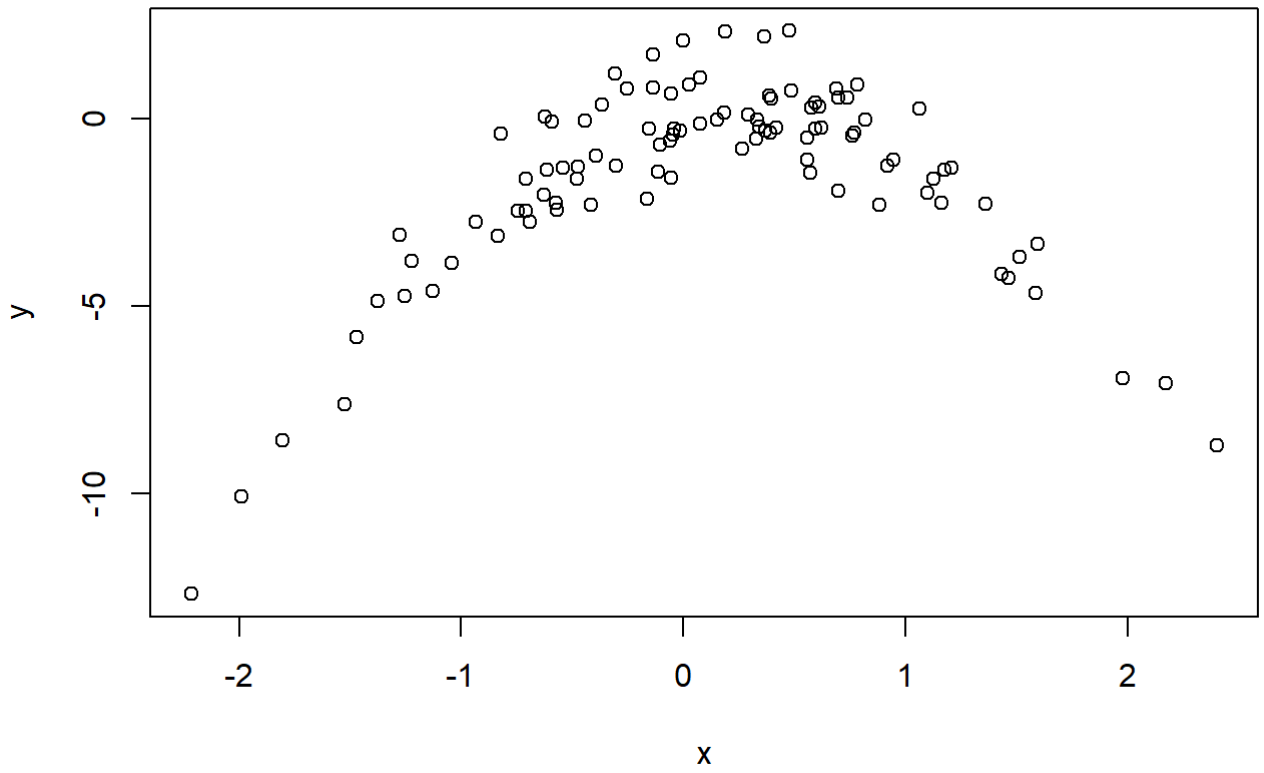
In this data set, what is n and what is p? Write out the model used to generate the data in equation form.

n=100, p=2 the model is $Y = X - 2X^2 +$

### (b)

Create a scatterplot of X against Y . Comment on what you find.

```
plot(x,y)
```




We found that y and x has a quadratic relationship.

# (c)

Set a random seed, and then compute the LOOCV errors that result from fitting the following four models using least squares:

i. $Y = \beta_0 + \beta_1 X + \epsilon$
ii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$
iii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon$
iv. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4 + \epsilon.$

Note you may find it helpful to use the data.frame() function to create a single data set containing both X and Y.

```
data1=data.frame(x,y)
set.seed(2)
cv.error=rep(0,4)
for (i in 1:4){
glm.fit=glm(y~poly(x ,i),data=data1)
cv.error[i]=cv.glm(data1 ,glm.fit)$delta [1]
}
cv.error
```

```
## [1] 7.2881616 0.9374236 0.9566218 0.9539049
```

# (d)

Repeat (c) using another random seed, and report your results. Are your results the same as what you got in (c)? Why?

```
set.seed(1)
cv.error1=rep(0,4)
for (i in 1:4){
glm.fit=glm(y~poly(x ,i),data=data1)
cv.error1[i]=cv.glm(data1 ,glm.fit)$delta [1]
}
cv.error1
```

```
## [1] 7.2881616 0.9374236 0.9566218 0.9539049
```

# (e)

Which of the models in (c) had the smallest LOOCV error? Is this what you expected? Explain your answer.

we can see a sharp drop in the estimated test MSE between the linear and quadratic fits, but then no clear improvement from using higher-order polynomials. So the second model has the smallest LOOCV error, Which is perfectly proved my expectation. Because from the scatterplot we can see there is a quadratic relationship between x and y.

# (f)

Comment on the statistical significance of the coefficient estimates that results from fitting each of the models in (c) using least squares. Do these results agree with the conclusions drawn based on the cross-validation results?

```
fit1=glm(y~x,data=data1)
fit2=glm(y~x+I(x^2),data=data1)
fit3=glm(y~x+I(x^2)+I(x^3),data=data1)
fit4=glm(y~x+I(x^2)+I(x^3)+I(x^4),data=data1)
summary(fit1)$coef
```

```
##             Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -1.625427  0.2619366 -6.205420 1.309300e-08
## x            0.692497  0.2909418  2.380191 1.923846e-02
```

```
summary(fit2)$coef
```

```
##               Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)  0.05671501  0.1176555   0.482043 6.308613e-01
## x            1.01716087  0.1079827   9.419666 2.403287e-15
## I(x^2)      -2.11892120  0.0847657 -24.997388 4.584330e-44
```

```
summary(fit3)$coef
```

```
##               Estimate Std. Error     t value     Pr(>|t|)
## (Intercept)  0.06150718 0.11950374   0.5146883 6.079538e-01
## x            0.97528027 0.18728149   5.2075636 1.089350e-06
## I(x^2)      -2.12379099 0.08700251 -24.4106856 5.873444e-43
## I(x^3)       0.01763858 0.06429037   0.2743580 7.843990e-01
```

```
summary(fit4)$coef
```

```
##                Estimate Std. Error     t value     Pr(>|t|)
## (Intercept)  0.156702953 0.13946192   1.1236253 2.640034e-01
## x            1.030825643 0.19133655   5.3874999 5.174326e-07
## I(x^2)      -2.409898183 0.23485506 -10.2612148 4.575229e-17
## I(x^3)      -0.009132904 0.06722881  -0.1358481 8.922288e-01
## I(x^4)       0.069785421 0.05324006   1.3107691 1.930956e-01
```

From th summary of 4 models fit by glm, we can see from the coefficient p-value the X^3 and X^4 term is not statistical significance, we can get the same conclusion.

# 6.2

For parts (a) through (c), indicate which of i. through iv. is correct. Justify your answer.

## (a)

The lasso, relative to least squares, is: i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance. ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias. iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance. iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

## (b)

Repeat (a) for ridge regression relative to least squares.

## (c)

Repeat (a) for non-linear methods relative to least squares.

# 6.10

We have seen that as the number of features used in a model increases, the training error will necessarily decrease, but the test error may not. We will now explore this in a simulated data set.

## (a)

Generate a data set with p = 20 features, n = 1,000 observations, and an associated quantitative response vector generated according to the model

$$Y = X\beta + \epsilon$$

, where $\beta$ has some elements that are exactly equal to zero.

```
set.seed(1)
p <- 20
n <- 1000
data = rnorm(n * p, mean = 0, sd = 1)
X <- matrix(data, nrow = n, ncol = p)

B <- rnorm(p, mean = 1, sd = 2)
B[7]=0
B[14]=0

eps=rnorm(p, mean = 0, sd = 3)

Y = X %*% B+eps
df <- cbind(Y, as.data.frame(X))
names(df) <- c("Y", paste0("X", 1:20))
```

# (b)

Split your data set into a training set containing 100 observations and a test set containing 900 observations.

```
train1=sample(1:1000, 100)

train=df[train1, ]
test=df[-train1, ]
```

# (c)

Perform best subset selection on the training set, and plot the training set MSE associated with the best model of each size.

# (d)

Plot the test set MSE associated with the best model of each size.

# (e)

For which model size does the test set MSE take on its minimum value? Comment on your results. If it takes on its minimum value for a model containing only an intercept or a model containing all of the features, then play around with the way that you are generating the data in (a) until you come up with a scenario in which the test set MSE is minimized for an intermediate model size.

# (f)

How does the model at which the test set MSE is minimized compare to the true model used to generate the data? Comment on the coefficient values.

# (g)

Create a plot displaying $\sqrt{\sum_{j=1}^{p}(\beta_j - \widehat{\beta}_j^r)^2}$ for a range of values of r, where $\widehat{\beta}_j^r$ is the jth coefficient estimate for the best model containing r coefficients. Comment on what you observe. How does this compare to the test MSE plot from (d)?