HW7

Tianqi Wu

4/26/2020

```
library(faraway)
library(leaps)
library(splines)
attach(prostate)
```

Problem 1

1(a): Backward Elimination

If we set the p-value threshold to 15%, backward elimination selects the following predictors: lcavol, lweight, lbph, svi. The adjusted R-squared is 0.6208.

```
model.1a.1 = lm(lpsa~., data=prostate)
summary(model.1a.1)
```

```
##
## Call:
## lm(formula = lpsa ~ ., data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 **
## age         -0.019637   0.011173  -1.758  0.08229 .
## lbph         0.107054   0.058449   1.832  0.07040 .
## svi          0.766157   0.244309   3.136  0.00233 **
## lcp         -0.105474   0.091013  -1.159  0.24964
## gleason      0.045142   0.157465   0.287  0.77503
## pgg45        0.004525   0.004421   1.024  0.30886
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

```r
model.1a.2 = lm(lpsa~.-gleason, data=prostate)
summary(model.1a.2)
```

```
##
## Call:
## lm(formula = lpsa ~ . - gleason, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.73117 -0.38137 -0.01728  0.43364  1.63513
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.953926   0.829439   1.150  0.25319
## lcavol       0.591615   0.086001   6.879 8.07e-10 ***
## lweight      0.448292   0.167771   2.672  0.00897 **
## age         -0.019336   0.011066  -1.747  0.08402 .
## lbph         0.107671   0.058108   1.853  0.06720 .
## svi          0.757734   0.241282   3.140  0.00229 **
## lcp         -0.104482   0.090478  -1.155  0.25127
## pgg45        0.005318   0.003433   1.549  0.12488
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7048 on 89 degrees of freedom
## Multiple R-squared:  0.6544, Adjusted R-squared:  0.6273
## F-statistic: 24.08 on 7 and 89 DF,  p-value: < 2.2e-16
```

```r
model.1a.3 = lm(lpsa~.-gleason-lcp, data=prostate)
summary(model.1a.3)
```

```
##
## Call:
## lm(formula = lpsa ~ . - gleason - lcp, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.77711 -0.41708  0.00002  0.40676  1.59681
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.980085   0.830665   1.180  0.24116
## lcavol       0.545770   0.076431   7.141 2.31e-10 ***
## lweight      0.449450   0.168078   2.674  0.00890 **
## age         -0.017470   0.010967  -1.593  0.11469
## lbph         0.105755   0.058191   1.817  0.07249 .
## svi          0.641666   0.219757   2.920  0.00442 **
## pgg45        0.003528   0.003068   1.150  0.25331
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7061 on 90 degrees of freedom
## Multiple R-squared:  0.6493, Adjusted R-squared:  0.6259
## F-statistic: 27.77 on 6 and 90 DF,  p-value: < 2.2e-16
```

```
model.1a.4 = lm(lpsa~.-gleason-lcp-pgg45, data=prostate)
summary(model.1a.4)
```

```
##
## Call:
## lm(formula = lpsa ~ . - gleason - lcp - pgg45, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.83505 -0.39396  0.00414  0.46336  1.57888
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.95100    0.83175   1.143 0.255882
## lcavol       0.56561    0.07459   7.583 2.77e-11 ***
## lweight      0.42369    0.16687   2.539 0.012814 *
## age         -0.01489    0.01075  -1.385 0.169528
## lbph         0.11184    0.05805   1.927 0.057160 .
## svi          0.72095    0.20902   3.449 0.000854 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7073 on 91 degrees of freedom
## Multiple R-squared:  0.6441, Adjusted R-squared:  0.6245
## F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

```
model.1a.5 = lm(lpsa~.-gleason-lcp-pgg45-age, data=prostate)
summary(model.1a.5)
```

```
##
## Call:
## lm(formula = lpsa ~ . - gleason - lcp - pgg45 - age, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.82653 -0.42270  0.04362  0.47041  1.48530
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.14554    0.59747   0.244  0.80809
## lcavol       0.54960    0.07406   7.422 5.64e-11 ***
## lweight      0.39088    0.16600   2.355  0.02067 *
## lbph         0.09009    0.05617   1.604  0.11213
## svi          0.71174    0.20996   3.390  0.00103 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7108 on 92 degrees of freedom
## Multiple R-squared:  0.6366, Adjusted R-squared:  0.6208
## F-statistic: 40.29 on 4 and 92 DF,  p-value: < 2.2e-16
```

```
anova(model.1a.5, model.1a.1)
```

```
## Analysis of Variance Table
##
```

```
## Model 1: lpsa ~ (lcavol + lweight + age + lbph + svi + lcp + gleason + 
##     pgg45) - gleason - lcp - pgg45 - age
## Model 2: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason + 
##     pgg45
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     92 46.485                           
## 2     88 44.163  4    2.3218 1.1566 0.3355
```

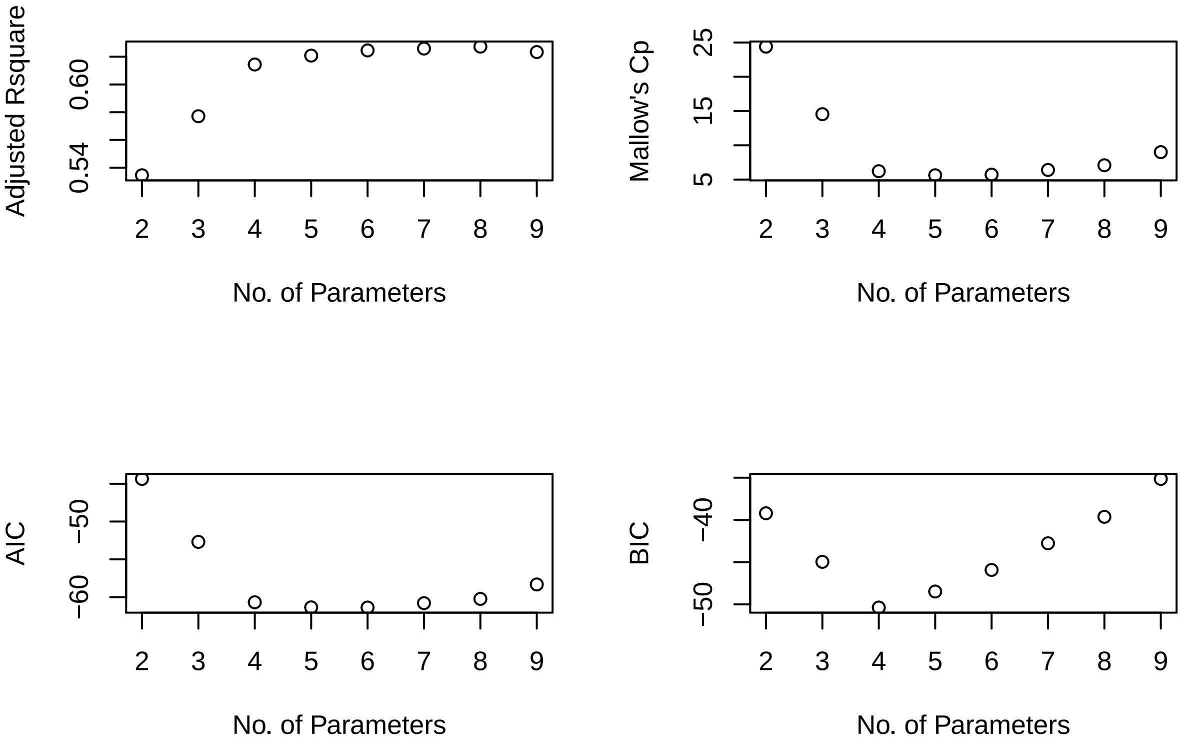
1(b): AIC

lcavol, lweight, age, lbph, svi

```r
b=regsubsets(lpsa~., data=prostate)
rs = summary(b)

n=dim(prostate)[1]; msize = 2:9;
par(mfrow=c(2,2))
plot(msize, rs$adjr2, xlab="No. of Parameters", ylab = "Adjusted Rsquare");
plot(msize, rs$cp, xlab="No. of Parameters", ylab = "Mallow's Cp");

Aic = n*log(rs$rss/n) + 2*msize;
Bic = n*log(rs$rss/n) + msize*log(n);
plot(msize, Aic, xlab="No. of Parameters", ylab = "AIC");
plot(msize, Bic, xlab="No. of Parameters", ylab = "BIC")
```



```r
## AIC
rs$which[which.min(Aic),]
```

```
## (Intercept)      lcavol     lweight         age        lbph         svi 
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE 
##         lcp     gleason       pgg45
##       FALSE       FALSE       FALSE
```

1(b): BIC

lcavol, lweight, svi

```r
rs$which[which.min(Bic),]
```

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##        TRUE        TRUE        TRUE       FALSE       FALSE        TRUE
##         lcp     gleason       pgg45
##       FALSE       FALSE       FALSE
```

1(c): Adjusted R^2

lcavol, lweight, age, lbph, svi, lcp, pgg45

```r
rs$which[which.max(rs$adjr2),]
```

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##        TRUE        TRUE        TRUE        TRUE        TRUE        TRUE
##         lcp     gleason       pgg45
##        TRUE       FALSE        TRUE
```

1(d): Mallows Cp

lcavol, lweight, lbph, svi

```r
rs$which[which.min(rs$cp),]
```

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##        TRUE        TRUE        TRUE       FALSE        TRUE        TRUE
##         lcp     gleason       pgg45
##       FALSE       FALSE       FALSE
```

Problem 2

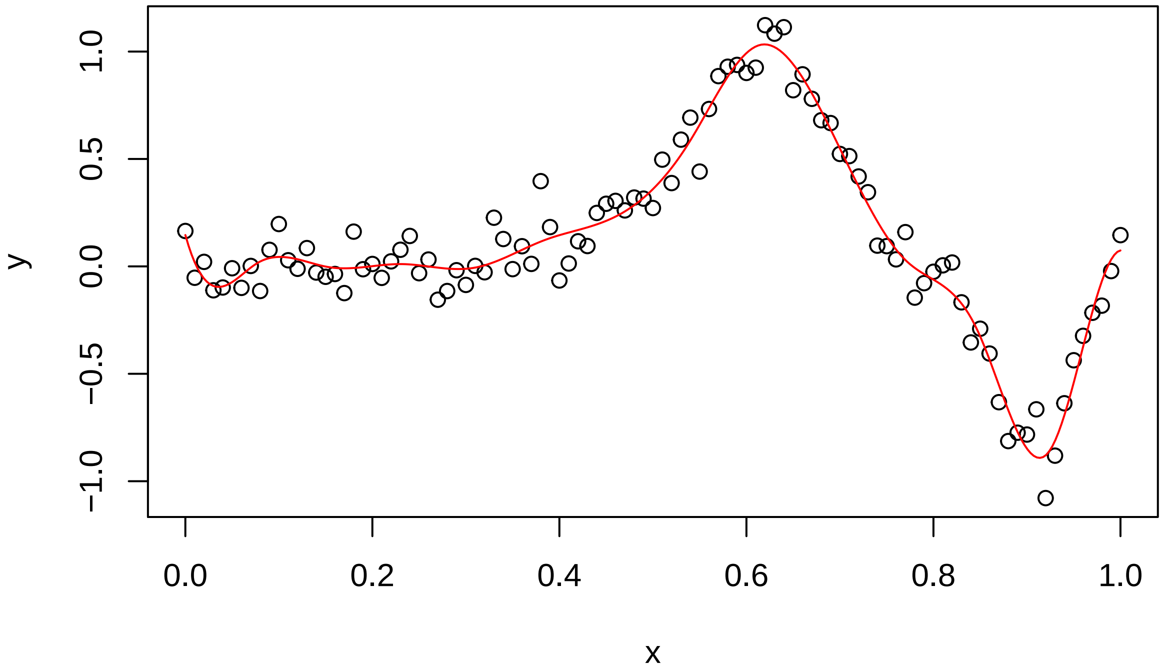
2(a)

```r
## prepare data
set.seed(130)
fun<-function(x) sin(2*pi*x^3)^3
x<-seq(0,1,by=0.01)
y<-fun(x) + 0.1*rnorm(101)

## Regression Splines
model.2a = lm(y~bs(x, df=16, intercept=TRUE))
summary(model.2a)
```

```
## 
## Call:
## lm(formula = y ~ bs(x, df = 16, intercept = TRUE))
## 
## Residuals:
##       Min        1Q    Median        3Q       Max 
## -0.218389 -0.059194 -0.002688  0.062515  0.281473 
## 
## Coefficients: (1 not defined because of singularities)
##                                    Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                         0.07418    0.08644   0.858 0.393247    
## bs(x, df = 16, intercept = TRUE)1   0.07184    0.12225   0.588 0.558354    
## bs(x, df = 16, intercept = TRUE)2  -0.32857    0.14281  -2.301 0.023858 *  
## bs(x, df = 16, intercept = TRUE)3   0.07656    0.14145   0.541 0.589759    
## bs(x, df = 16, intercept = TRUE)4  -0.12880    0.12416  -1.037 0.302505    
## bs(x, df = 16, intercept = TRUE)5  -0.02931    0.11983  -0.245 0.807386    
## bs(x, df = 16, intercept = TRUE)6  -0.13453    0.11843  -1.136 0.259172    
## bs(x, df = 16, intercept = TRUE)7   0.07815    0.11837   0.660 0.510915    
## bs(x, df = 16, intercept = TRUE)8   0.11557    0.11772   0.982 0.329042    
## bs(x, df = 16, intercept = TRUE)9   0.42498    0.11876   3.578 0.000574 ***
## bs(x, df = 16, intercept = TRUE)10  1.18992    0.11691  10.178 2.25e-16 ***
## bs(x, df = 16, intercept = TRUE)11  0.56057    0.12087   4.638 1.26e-05 ***
## bs(x, df = 16, intercept = TRUE)12 -0.17489    0.11528  -1.517 0.132942    
## bs(x, df = 16, intercept = TRUE)13 -0.10874    0.13223  -0.822 0.413185    
## bs(x, df = 16, intercept = TRUE)14 -1.57134    0.12418 -12.654  < 2e-16 ***
## bs(x, df = 16, intercept = TRUE)15 -0.01907    0.17011  -0.112 0.910983    
## bs(x, df = 16, intercept = TRUE)16       NA         NA      NA       NA    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.1 on 85 degrees of freedom
## Multiple R-squared:  0.9561, Adjusted R-squared:  0.9484 
## F-statistic: 123.5 on 15 and 85 DF,  p-value: < 2.2e-16
```

```
plot(x, y)
lines(spline(x, predict(model.2a)), col="red", lty=1)
```



2(b)

AIC is -450.451 and BIC is -408.6091.

```
rs = anova(model.2a)$`Sum Sq`[2]

n=101; msize = 16;
Aic = n*log(rs/n) + 2*msize;
Bic = n*log(rs/n) + msize*log(n)
Aic
```

```
## [1] -450.451
```

```
Bic
```

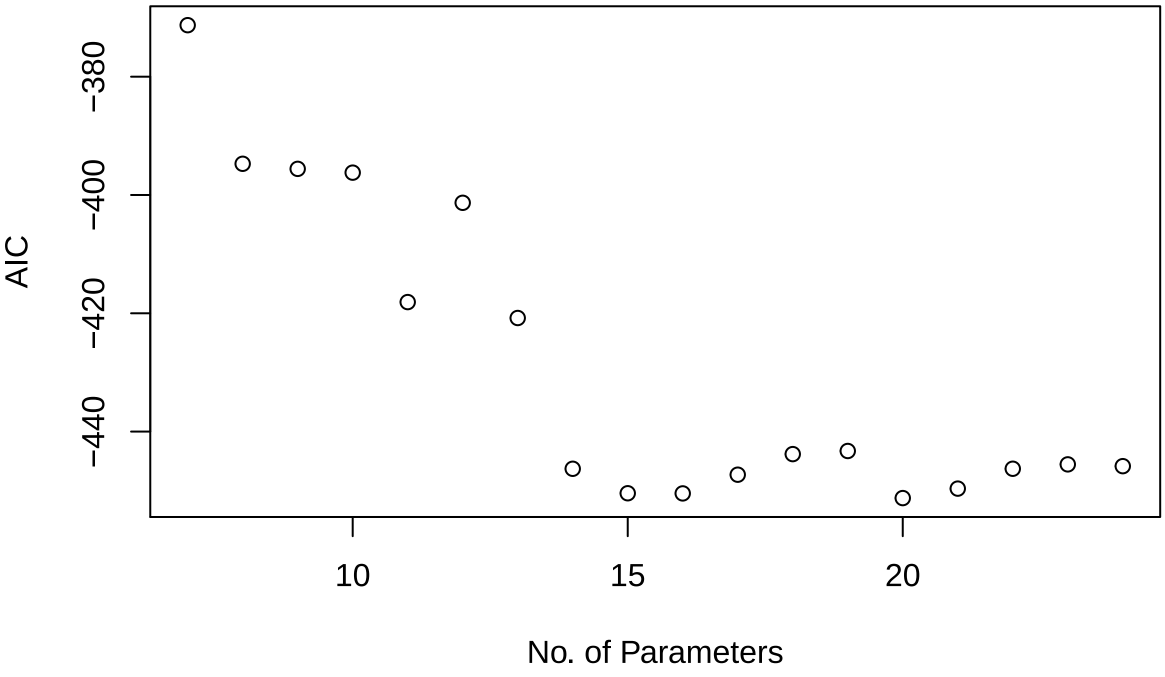
```
## [1] -408.6091
```

2(c)

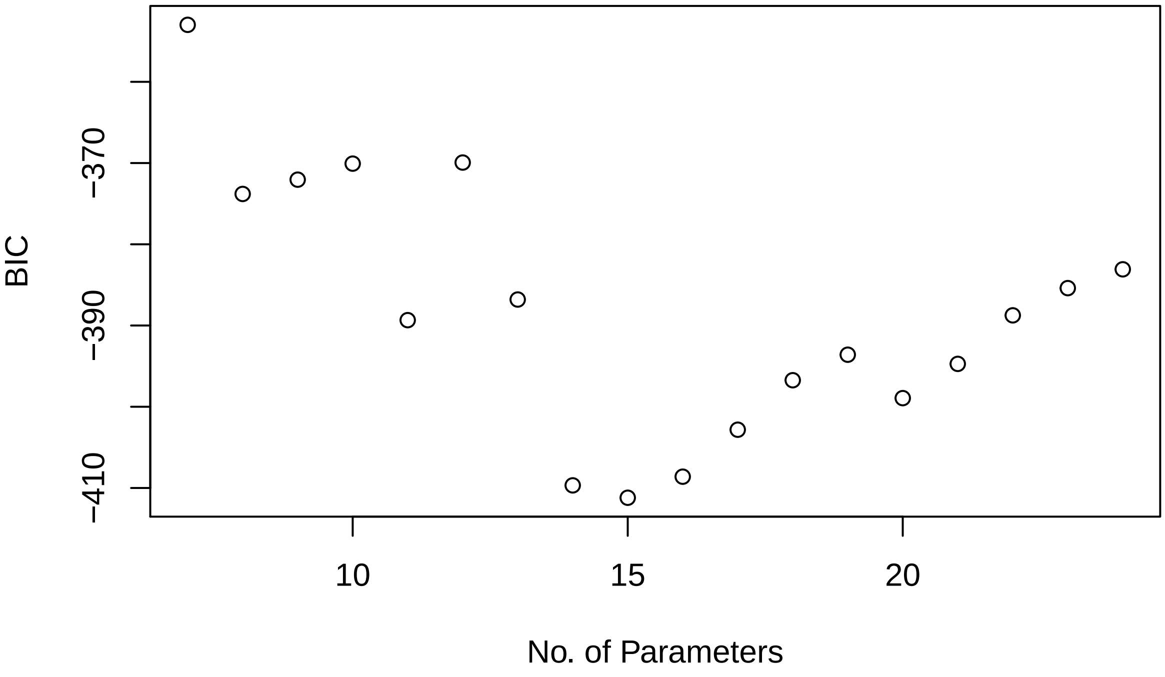
For Aic, the best model has 20 knots. For Bic, the best model has 15 knots.

```
n=101;
Aic = c(); Bic = c(); nparameter = c()
for (i in 3:20) {
  msize = i+4
  model.2c = lm(y~bs(x, df=msize, intercept = T))
  nparameter = append(nparameter, msize)
  rs = anova(model.2c)$`Sum Sq`[2]
  Aic = append(Aic, n*log(rs/n) + 2*msize)
  Bic = append(Bic, n*log(rs/n) + msize*log(n))
}

plot(nparameter, Aic, xlab="No. of Parameters", ylab = "AIC");
```



```
plot(nparameter, Bic, xlab="No. of Parameters", ylab = "BIC")
```



```
nparameter[which.min(Aic)]
```

```
## [1] 20
```

```
nparameter[which.min(Bic)]
```

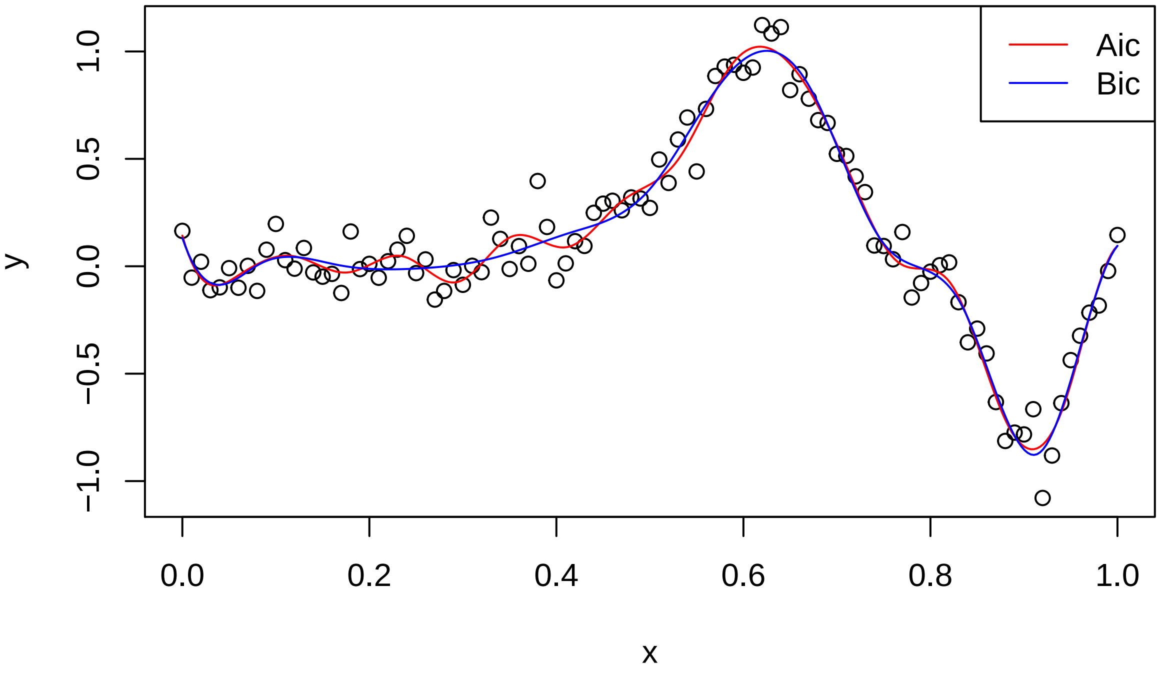
```
## [1] 15
```

2(d)

```
## Aic
model.aic = lm(y~bs(x, df=20, intercept = T))

## Bic
model.bic = lm(y~bs(x, df=15, intercept = T))

plot(x, y)
lines(spline(x, predict(model.aic)), col="red", lty=1)
lines(spline(x, predict(model.bic)), col="blue", lty=1)
legend("topright", lty=rep(1,1), col=c("red", "blue"), legend=c("Aic", "Bic"))
```


Aic
Bic
y
x