```
> #2.
> Kidney <- read.csv("Kidney_Function_Data.csv", header = F)
> colnames(Kidney) <- c("Y", "X1", "X2", "X3")
> #2.(a)
> x1 <- Kidney$X1 - mean(Kidney$X1)
> x2 <- Kidney$X2 - mean(Kidney$X2)
> x3 <- Kidney$X3 - mean(Kidney$X3)
> Kidney.lm <- lm(Y~x1*x2*x3, data = Kidney)
> Kidney.lm

Call:
lm(formula = Y ~ x1 * x2 * x3, data = Kidney)

Coefficients:
(Intercept)              x1              x2
    82.66306       -45.10057       -0.74429
         x3           x1:x2           x1:x3
     0.95737         0.74158         0.36246
       x2:x3        x1:x2:x3
    -0.01617        -0.03316

> #Yi = 82.663.6 -45.10057Xi1 -
0.74429Xi2+0.95737Xi3+30.74158Xi1Xi2+0.36246Xi1Xi3-0.01617Xi2Xi3-
0.03316Xi1Xi2Xi3 +residul_i
>
> Kidney.lm2 <- lm(Y ~    x1*x2*x3 + poly(x1, 2) + poly(x2, 2) + poly(x3, 2), data =
Kidney)
> Kidney.lm2

Call:
lm(formula = Y ~ x1 * x2 * x3 + poly(x1, 2) + poly(x2, 2) + poly(x3,
    2), data = Kidney)

Coefficients:
 (Intercept)              x1              x2
    84.09757       -45.00064       -0.72517
          x3   poly(x1, 2)1   poly(x1, 2)2
     1.01668              NA        11.72818
```

```
poly(x2, 2)1    poly(x2, 2)2    poly(x3, 2)1
                NA           14.24919                    NA
poly(x3, 2)2                 x1:x2              x1:x3
     -17.89652               0.31876            0.44109
           x2:x3           x1:x2:x3
       -0.02136            -0.05720

>
> library("ALSM")
> library("leaps")
> BestSub(Kidney[, 2:4], Kidney[, 1], method = 'aic', num = 3)
   p 1 2 3           SSEp               r2         r2.adj
1 2 1 0 0 11068.478 0.6429007 0.63138133
1 2 0 1 0 17169.856 0.4460535 0.42818429
1 2 0 0 1 27286.682 0.1196571 0.09125894
2 3 1 1 0    7666.102 0.7526706 0.73618198
2 3 1 0 1    8713.158 0.7188897 0.70014903
2 3 0 1 1 12393.018 0.6001674 0.57351187
3 4 1 1 1    4499.974 0.8548186 0.83979979
               Cp         AICp         SBCp        PRESSp
1    42.33061 195.9065 198.8995 12402.603
1    81.65083 210.3953 213.3883 19676.109
1 146.84854 225.6823 228.6753 30198.308
2    22.40406 185.7858 190.2754    9348.907
2    29.15179 190.0107 194.5002 10326.342
2    52.86659 201.6366 206.1261 14864.376
3     4.00000 170.2055 176.1916    5963.853
> #subset            AIC
> #x1,x2,x3          170.2055
> #x1,x2              185.7858
> #x1,x3              190.0107
>
> #2.(b)
> #Not so much at all
>
> #3.
> PA <- read.csv("Performance_Ability_Data.csv", header = F)
> colnames(PA) <- c("Y", "X")
```

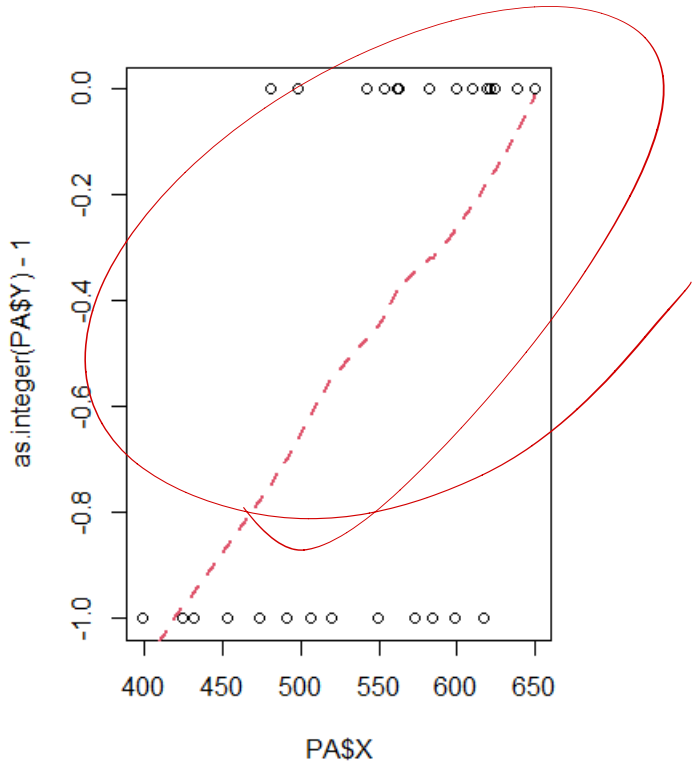
```
> PA
    Y   X
1   0 474
2   0 432
3   0 453
4   1 481
5   1 619
6   0 584
7   0 399
8   1 582
9   1 638
10 1 624
11 1 542
12 1 650
13 1 553
14 0 425
15 1 563
16 0 549
17 1 498
18 0 520
19 1 610
20 0 598
21 0 491
22 0 617
23 1 621
24 0 573
25 1 562
26 0 506
27 1 600
> #3.(a)
> PA.glm <- glm(Y ~ X,
+                         data = PA, family = "binomial")
> PA.glm

Call:  glm(formula = Y ~ X, family = "binomial", data = PA)

Coefficients:
(Intercept)                  X
```

```
    -10.30893          0.01892

Degrees of Freedom: 26 Total (i.e. Null);    25 Residual
Null Deviance:        37.39
Residual Deviance: 29.24       AIC: 33.24
> #odds = exp(-10.30893+0.01892X)
> #3.(b)
> plot(PA$X, as.integer(PA$Y)-1)
> lines(lowess(PA$X, as.integer(PA$Y)-1),
+             col = 2, lwd = 2, lty = 2)
>
> a <- min(PA$X)
> b <- max(PA$X)
> new.X <- data.frame(X = seq(a, b, len = 50))
> predicted.Y <- predict(PA.glm, new.X, type="response")
> new.X <- c(new.X)$X
> lines(new.X, predicted.Y, lwd = 2, col = "black")
```




```
> #2.
> Kidney <- read.csv("Kidney_Function_Data.csv", header = F)
> colnames(Kidney) <- c("Y", "X1", "X2", "X3")
```

```
> #2.(a)
> x1 <- Kidney$X1 - mean(Kidney$X1)
> x2 <- Kidney$X2 - mean(Kidney$X2)
> x3 <- Kidney$X3 - mean(Kidney$X3)
> Kidney.lm <- lm(Y~x1*x2*x3, data = Kidney)
> Kidney.lm

Call:
lm(formula = Y ~ x1 * x2 * x3, data = Kidney)

Coefficients:
(Intercept)                  x1                  x2
     82.66306        -45.10057         -0.74429
             x3               x1:x2               x1:x3
      0.95737           0.74158          0.36246
         x2:x3         x1:x2:x3
     -0.01617         -0.03316

> #Yi = 82.663.6 -45.10057Xi1 -
0.74429Xi2+0.95737Xi3+30.74158Xi1Xi2+0.36246Xi1Xi3-0.01617Xi2Xi3-
0.03316Xi1Xi2Xi3 +residul_i
>
> Kidney.lm2 <- lm(Y ~    x1*x2*x3 + poly(x1, 2) + poly(x2, 2) + poly(x3, 2), data =
Kidney)
> Kidney.lm2

Call:
lm(formula = Y ~ x1 * x2 * x3 + poly(x1, 2) + poly(x2, 2) + poly(x3,
       2), data = Kidney)

Coefficients:
  (Intercept)                   x1                   x2
      84.09757          -45.00064           -0.72517
              x3    poly(x1, 2)1    poly(x1, 2)2
        1.01668                    NA           11.72818
poly(x2, 2)1    poly(x2, 2)2    poly(x3, 2)1
                NA           14.24919                    NA
poly(x3, 2)2                x1:x2                x1:x3
```

```
    -17.89652        0.31876        0.44109
        x2:x3       x1:x2:x3
      -0.02136       -0.05720

>
> library("ALSM")
> library("leaps")
> BestSub(Kidney[, 2:4], Kidney[, 1], method = 'aic', num = 3)
  p 1 2 3       SSEp         r2       r2.adj
1 2 1 0 0 11068.478 0.6429007 0.63138133
1 2 0 1 0 17169.856 0.4460535 0.42818429
1 2 0 0 1 27286.682 0.1196571 0.09125894
2 3 1 1 0  7666.102 0.7526706 0.73618198
2 3 1 0 1  8713.158 0.7188897 0.70014903
2 3 0 1 1 12393.018 0.6001674 0.57351187
3 4 1 1 1  4499.974 0.8548186 0.83979979
              Cp       AICp       SBCp      PRESSp
1   42.33061 195.9065 198.8995 12402.603
1   81.65083 210.3953 213.3883 19676.109
1 146.84854 225.6823 228.6753 30198.308
2   22.40406 185.7858 190.2754  9348.907
2   29.15179 190.0107 194.5002 10326.342
2   52.86659 201.6366 206.1261 14864.376
3    4.00000 170.2055 176.1916  5963.853
> #subset          AIC
> #x1,x2,x3        170.2055
> #x1,x2           185.7858
> #x1,x3           190.0107
>
> #2.(b)
> #Not so much at all
>
> #3.
> PA <- read.csv("Performance_Ability_Data.csv", header = F)
> colnames(PA) <- c("Y", "X")
> PA
   Y   X
1  0 474
```

```
2    0 432
3    0 453
4    1 481
5    1 619
6    0 584
7    0 399
8    1 582
9    1 638
10 1 624
11 1 542
12 1 650
13 1 553
14 0 425
15 1 563
16 0 549
17 1 498
18 0 520
19 1 610
20 0 598
21 0 491
22 0 617
23 1 621
24 0 573
25 1 562
26 0 506
27 1 600
> #3.(a)
> PA.glm <- glm(Y ~ X,
+                          data = PA, family = "binomial")
> PA.glm

Call:    glm(formula = Y ~ X, family = "binomial", data = PA)

Coefficients:
(Intercept)                    X
   -10.30893           0.01892

Degrees of Freedom: 26 Total (i.e. Null);    25 Residual
```

```
Null Deviance:        37.39
Residual Deviance: 29.24       AIC: 33.24
> #odds = exp(-10.30893+0.01892X)
> #3.(b)
> plot(PA$X, as.integer(PA$Y)-1)
> lines(lowess(PA$X, as.integer(PA$Y)-1),
+             col = 2, lwd = 2, lty = 2)
>
> a <- min(PA$X)
> b <- max(PA$X)
> new.X <- data.frame(X = seq(a, b, len = 50))
> predicted.Y <- predict(PA.glm, new.X, type="response")
> new.X <- c(new.X)$X
> lines(new.X, predicted.Y, lwd = 2, col = "black")
> #3.(c)
> summary(PA.glm)

Call:
glm(formula = Y ~ X, family = "binomial", data = PA)

Deviance Residuals:
      Min            1Q      Median            3Q           Max
-1.7845    -0.8350      0.5065      0.8371      1.7145

Coefficients:
                        Estimate Std. Error z value
(Intercept) -10.308925      4.376997    -2.355
X                       0.018920      0.007877      2.402
                    Pr(>|z|)
(Intercept)      0.0185 *
X                       0.0163 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

      Null deviance: 37.393    on 26    degrees of freedom
```

```
Residual deviance: 29.242  on 25  degrees of freedom
AIC: 33.242

Number of Fisher Scoring iterations: 4

> #3.(c)
> summary(PA.glm)

Call:
glm(formula = Y ~ X, family = "binomial", data = PA)

Deviance Residuals:
      Min         1Q     Median         3Q        Max
-1.7845  -0.8350   0.5065   0.8371   1.7145

Coefficients:
                    Estimate Std. Error z value
(Intercept) -10.308925   4.376997  -2.355
X                   0.018920   0.007877   2.402
                  Pr(>|z|)
(Intercept)   0.0185 *
X                   0.0163 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

      Null deviance: 37.393  on 26  degrees of freedom
Residual deviance: 29.242  on 25  degrees of freedom
AIC: 33.242

Number of Fisher Scoring iterations: 4

> #b1 = 0.018920
> exp(b1)
Error: object 'b1' not found
> b1 = 0.018920
```

```
> exp(b1)
[1] 1.0191
> #3.(d)
> predicted.Y <- predict(PA.glm, 550, type="response")
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev =
object$xlevels) :
   'data' must be a data.frame, environment, or list
> #3.(d)
> data.frame(550)
   X550
1   550
> predicted.Y <- predict(PA.glm,new.X , type="response")
Error in model.frame.default(Terms, newdata, na.action = na.action, xlev =
object$xlevels) :
   'data' must be a data.frame, environment, or list
> new.X <- data.frame(MonthExperience = seq(a, b, len = 50))
> new.X
    MonthExperience
1          399.0000
2          404.1224
3          409.2449
4          414.3673
5          419.4898
6          424.6122
7          429.7347
8          434.8571
9          439.9796
10         445.1020
11         450.2245
12         455.3469
13         460.4694
14         465.5918
15         470.7143
16         475.8367
17         480.9592
18         486.0816
19         491.2041
20         496.3265
```

```
21          501.4490
22          506.5714
23          511.6939
24          516.8163
25          521.9388
26          527.0612
27          532.1837
28          537.3061
29          542.4286
30          547.5510
31          552.6735
32          557.7959
33          562.9184
34          568.0408
35          573.1633
36          578.2857
37          583.4082
38          588.5306
39          593.6531
40          598.7755
41          603.8980
42          609.0204
43          614.1429
44          619.2653
45          624.3878
46          629.5102
47          634.6327
48          639.7551
49          644.8776
50          650.0000
> #3.(c)
> summary(PA.glm)

Call:
glm(formula = Y ~ X, family = "binomial", data = PA)

Deviance Residuals:
      Min         1Q     Median         3Q        Max
```

```
-1.7845    -0.8350      0.5065      0.8371      1.7145

Coefficients:
                        Estimate Std. Error z value
(Intercept) -10.308925      4.376997    -2.355
X                       0.018920      0.007877      2.402
                    Pr(>|z|)
(Intercept)      0.0185 *
X                       0.0163 *
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

      Null deviance: 37.393     on 26     degrees of freedom
Residual deviance: 29.242     on 25     degrees of freedom
AIC: 33.242

Number of Fisher Scoring iterations: 4

>
> b1 = 0.018920
> exp(b1)
[1] 1.0191
>
> #3.(d)
> new.X <- data.frame(550)
>predicted.Y <- predict(PA.glm,new.X , type="response")
```