Lab Sheet 5 Solution

Jennifer Chan

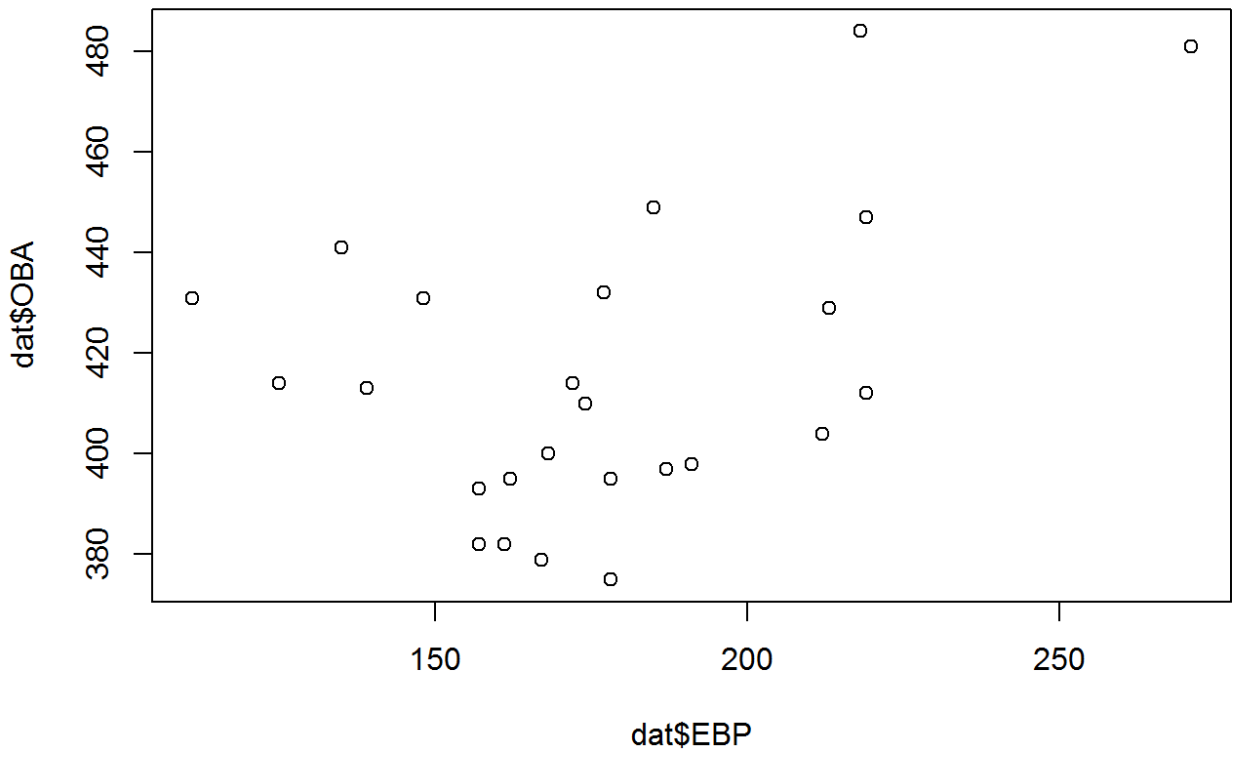
```
library(tidyverse)
knitr::opts_chunk$set(echo = TRUE, tidy = FALSE, fig.align = 'center', fig.pos = 'H')
```

Question 1

(a)

The question asks to start with a polynomial regression model of order up to 6. So perhaps a good starting point is to start with an orthogonal polynomial of order 6 to select the suitable order of the polynomial. In orthogonal polynomial, the different order terms are indpendent and so the p-value of each order term will not change much if one drops some higher order terms.

```
dat <- read.table(file="data/baseball.txt", header=TRUE)
plot(dat$EBP, dat$OBA)
```




```
poly.6 <- lm(OBA ~ poly(EBP, 6), data=dat)
summary(poly.6)
```

```
##
## Call:
## lm(formula = OBA ~ poly(EBP, 6), data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.428 -16.450  -1.945   9.169  44.511
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    415.520      4.757  87.346  < 2e-16 ***
## poly(EBP, 6)1   58.909     23.786   2.477  0.02342 *
## poly(EBP, 6)2   71.893     23.786   3.023  0.00732 **
## poly(EBP, 6)3  -27.946     23.786  -1.175  0.25534
## poly(EBP, 6)4  -22.840     23.786  -0.960  0.34966
## poly(EBP, 6)5   10.822     23.786   0.455  0.65456
## poly(EBP, 6)6   -8.831     23.786  -0.371  0.71478
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.79 on 18 degrees of freedom
## Multiple R-squared:  0.4988, Adjusted R-squared:  0.3318
## F-statistic: 2.986 on 6 and 18 DF,  p-value: 0.03328
```

```
poly.2 <- lm(OBA ~ poly(EBP, 2), data=dat)
summary(poly.2)
```

```
##
## Call:
## lm(formula = OBA ~ poly(EBP, 2), data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.139 -15.057  -4.834   8.839  56.401
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    415.520      4.609  90.162  < 2e-16 ***
## poly(EBP, 2)1   58.909     23.043   2.556  0.01799 *
## poly(EBP, 2)2   71.893     23.043   3.120  0.00499 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.04 on 22 degrees of freedom
## Multiple R-squared:  0.4251, Adjusted R-squared:  0.3729
## F-statistic: 8.135 on 2 and 22 DF,  p-value: 0.002266
```

As the highest order which is significant is 2 for the orthogonal polynomial, so we can conclude that order 2 is suitable for the polynomial regression model. A formal comparison of the polynomial model of order 2 vs order 6 using a F-test yields a p-value of 0.6265 indicating a simplier model should not be rejected.

```
RSS_H1 <- deviance(poly.6)
RSS_H0 <- deviance(poly.2)
p <- length(coef(poly.6))
q <- length(coef(poly.2))
n <- nrow(dat)
fobs <- (RSS_H0 - RSS_H1) / (p - q) / (RSS_H1 / (n - p))
1 - pf(fobs, p - q, n - p)
```

```
## [1] 0.6264954
```

Thus, the second-order polynomial model

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \epsilon_i$$

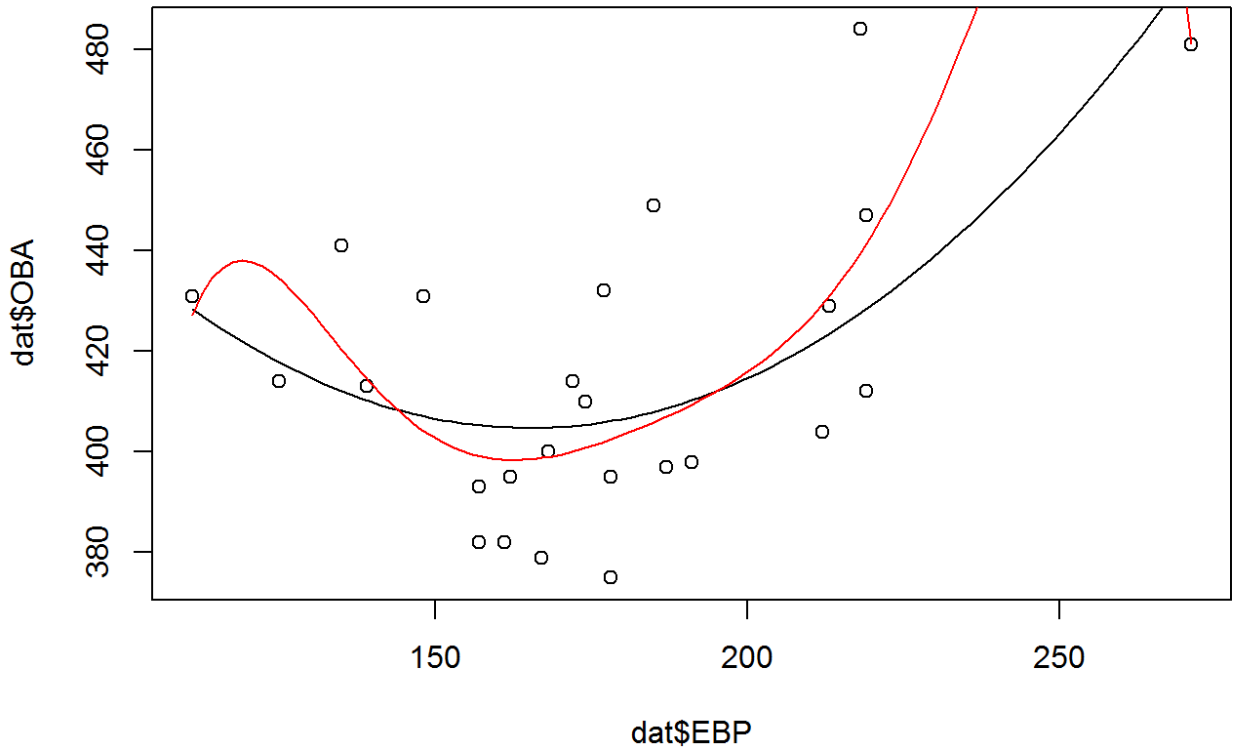
where x_i and Y_i is the EBP and OBP for the i-th observation may be a good fit to the data. The t-test for $H_0 : \beta_2$ yields a p-value of 0.00499 suggesting that we should not simplify the model further (using `base` plot in R).

```
poly.2 <- lm(OBA ~ EBP + I(EBP^2), data=dat)
summary(poly.2)
```

```
##
## Call:
## lm(formula = OBA ~ EBP + I(EBP^2), data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.139 -15.057  -4.834   8.839  56.401
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 625.729571  90.120360   6.943  5.7e-07 ***
## EBP          -2.679007   0.977776  -2.740  0.01196 *
## I(EBP^2)      0.008120   0.002603   3.120  0.00499 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.04 on 22 degrees of freedom
## Multiple R-squared:  0.4251, Adjusted R-squared:  0.3729
## F-statistic: 8.135 on 2 and 22 DF,  p-value: 0.002266
```

The following is just a sanity check.

```
plot(dat$EBP, dat$OBA)
beta2 <- coef(poly.2)
poly.6 <- lm(OBA ~ poly(EBP, 6, raw=TRUE), data=dat)
beta6 <- coef(poly.6)
poly2 <- function(x) beta2[1] + beta2[2]*x + beta2[3]*x^2
poly6 <- function(x) beta6[1] + beta6[2]*x + beta6[3]*x^2 + beta6[4]*x^3 + beta6[5]*x^4 + beta6[6]*
x^5 + beta6[7]*x^6
curve(poly2, from=min(dat$EBP), to=max(dat$EBP), add=T)
curve(poly6, from=min(dat$EBP), to=max(dat$EBP), add=T, col="red")
```



In the above plot, the red and black lines correspond to the fitted lines for `poly.6` and `poly.2`, respectively. It's not clear whether `poly.2` is the best model however `poly.6` is clearly an overfitted model.

(b)

The fitted second order polynomial model is

$$\hat{Y}_i = 625.729571 - 2.679007x_i + 0.008120x_i^2.$$

(c)

```
predict(poly.2, newdata=data.frame(EBP=250, EBP2=62500), interval="prediction", level=0.9)
```

```
##        fit      lwr      upr
## 1 463.4753 417.9762 508.9744
```

The predicted OBA at EBP of 250 is 463.4753. The associated 90% prediction interval is (417.9762, 508.9744).

Extra for orthogonal polynomial

Question 2

(a)

```
dat <- read.csv("data/diabetes.csv")
skimr::skim(dat)
```

Data summary

Name	dat
Number of rows	442
Number of columns	11

Column type frequency:

numeric	11

Group variables

Variable type: numeric

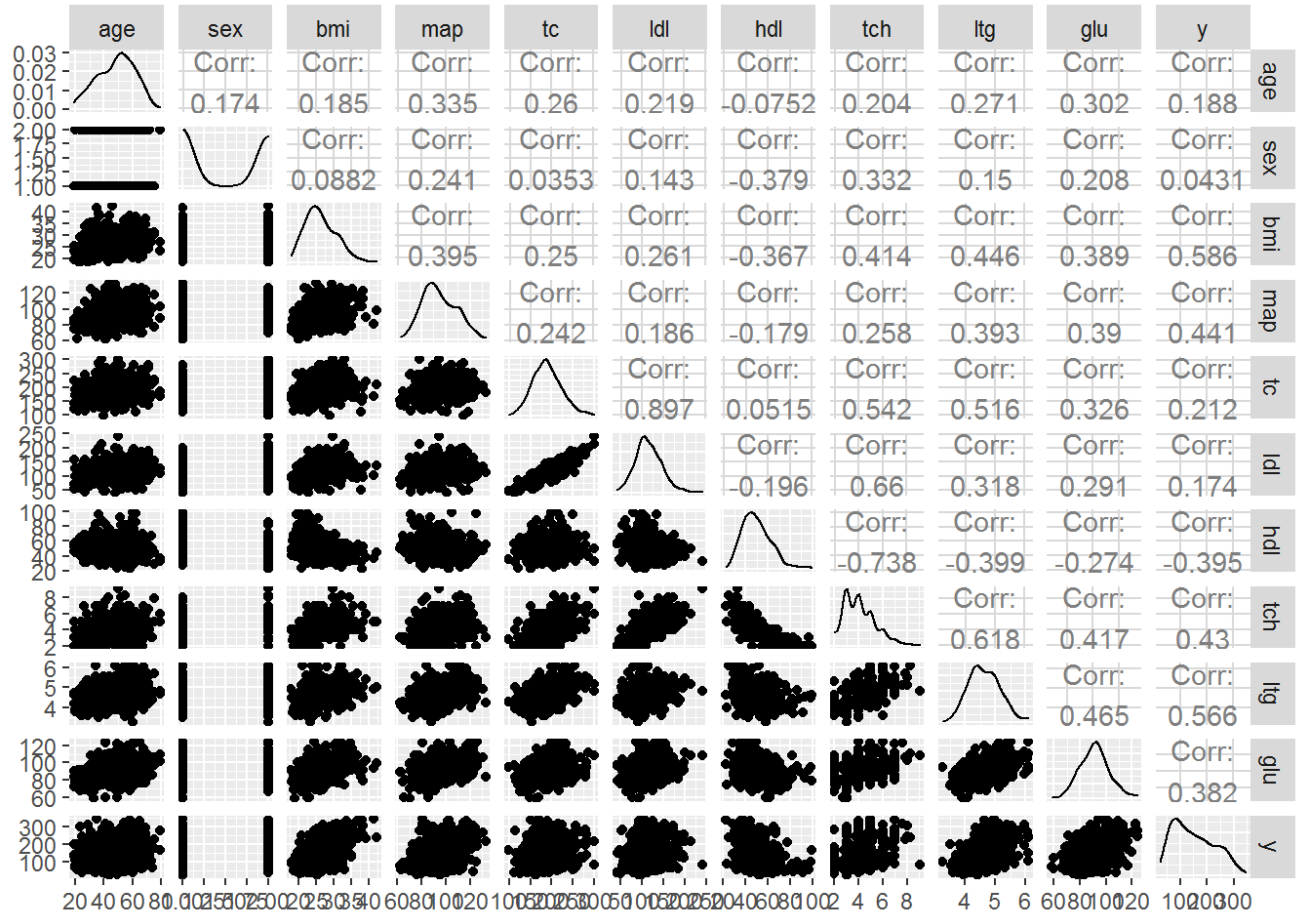
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	48.52	13.11	19.00	38.25	50.00	59.00	79.00	
sex	0	1	1.47	0.50	1.00	1.00	1.00	2.00	2.00	
bmi	0	1	26.38	4.42	18.00	23.20	25.70	29.28	42.20	
map	0	1	94.65	13.83	62.00	84.00	93.00	105.00	133.00	
tc	0	1	189.14	34.61	97.00	164.25	186.00	209.75	301.00	
ldl	0	1	115.44	30.41	41.60	96.05	113.00	134.50	242.40	
hdl	0	1	49.79	12.93	22.00	40.25	48.00	57.75	99.00	
tch	0	1	4.07	1.29	2.00	3.00	4.00	5.00	9.09	
ltg	0	1	4.64	0.52	3.26	4.28	4.62	5.00	6.11	
glu	0	1	91.26	11.50	58.00	83.25	91.00	98.00	124.00	
y	0	1	152.13	77.09	25.00	87.00	140.50	211.50	346.00	

There are 11 variables in all (10 numerical variables (integers) and 1 categorical variable) including the response variable 5 specific 5 are to be considered as data contains 442 observations.

(b)

```
GGally::ggpairs(dat, progress=F)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



The explanatory variable that is most highly correlated with the response is the body mass index (correlation coefficient of 0.586).

(c)

```
M0 <- lm(y ~ 1, data=dat)
Mf <- lm(y ~ ., data=dat)
```

(d)

The first model can be found by using:

```
MforwardAIC <- step(M0, direction="forward", trace=0, k=2, scope= list(lower=M0, upper=Mf))
summary(MforwardAIC)
```

```
##
## Call:
## lm(formula = y ~ bmi + ltg + map + tc + sex + ldl, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -158.275  -39.476   -2.065   37.219  148.690
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -313.7666    25.3848 -12.360  < 2e-16 ***
## bmi            5.7111     0.7073   8.075 6.69e-15 ***
## ltg           73.3065     7.3083  10.031  < 2e-16 ***
## map            1.1266     0.2158   5.219 2.79e-07 ***
## tc            -1.0429     0.2208  -4.724 3.12e-06 ***
## sex          -21.5910     5.7056  -3.784 0.000176 ***
## ldl            0.8433     0.2298   3.670 0.000272 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.06 on 435 degrees of freedom
## Multiple R-squared:  0.5149, Adjusted R-squared:  0.5082
## F-statistic: 76.95 on 6 and 435 DF,  p-value: < 2.2e-16
```

For the second model using F-test you must do it by “hand”:

```
drop1(Mf, test="F")
```

```
## Single term deletions
##
## Model:
## y ~ age + sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>              1263986 3539.6
## age     1        82 1264068 3537.7  0.0281 0.8670306
## sex     1     44999 1308984 3553.1 15.3439 0.0001042 ***
## bmi     1    179033 1443019 3596.2 61.0477 4.296e-14 ***
## map     1     72100 1336086 3562.2 24.5852 1.024e-06 ***
## tc      1     10600 1274586 3541.3  3.6144 0.0579476 .
## ldl     1      5799 1269785 3539.7  1.9774 0.1603902
## hdl     1       663 1264649 3537.9  0.2260 0.6347233
## tch     1      3526 1267512 3538.9  1.2024 0.2734587
## ltg     1     56016 1320001 3556.8 19.1005 1.556e-05 ***
## glu     1      3080 1267066 3538.7  1.0504 0.3059895
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MstepFtest <- update(Mf, . ~ . - age)
drop1(MstepFtest, test="F")
```

```
## Single term deletions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>              1264068 3537.7
## sex     1     45830 1309898 3551.4 15.6624 8.850e-05 ***
## bmi     1    179084 1443152 3594.2 61.2027 3.993e-14 ***
## map     1     73847 1337915 3560.8 25.2376 7.432e-07 ***
## tc      1     10569 1274637 3539.4  3.6120   0.05803 .
## ldl     1      5751 1269820 3537.7  1.9656   0.16163
## hdl     1       646 1264715 3535.9  0.2209   0.63856
## tch     1      3543 1267611 3536.9  1.2107   0.27180
## ltg     1     55964 1320032 3554.8 19.1258 1.535e-05 ***
## glu     1      3001 1267069 3536.7  1.0257   0.31173
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MstepFtest <- update(MstepFtest, . ~ . - hdl)
add1(MstepFtest, test="F", scope=Mf)
```

```
## Single term additions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + tch + ltg + glu
##        Df Sum of Sq     RSS    AIC F value Pr(>F)
## <none>              1264715 3535.9
## age     1     65.92 1264649 3537.9  0.0225 0.8808
## hdl     1    646.48 1264068 3537.7  0.2209 0.6386
```

```
drop1(MstepFtest, test="F")
```

```
## Single term deletions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + tch + ltg + glu
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>              1264715 3535.9
## sex     1     46381 1311096 3549.8 15.8794 7.920e-05 ***
## bmi     1    178542 1443256 3592.3 61.1273 4.111e-14 ***
## map     1     73533 1338248 3558.9 25.1756 7.655e-07 ***
## tc      1     26839 1291554 3543.2  9.1890  0.002581 **
## ldl     1      7505 1272219 3536.5  2.5694  0.109677
## tch     1      3247 1267961 3535.0  1.1116  0.292320
## ltg     1     97508 1362223 3566.7 33.3840 1.447e-08 ***
## glu     1      3093 1267808 3535.0  1.0590  0.304011
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MstepFtest <- update(MstepFtest, . ~ . - glu)
add1(MstepFtest, test="F", scope=Mf)
```

```
## Single term additions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + tch + ltg
##        Df Sum of Sq     RSS    AIC F value Pr(>F)
## <none>              1267808 3535.0
## age     1      0.44 1267807 3537.0  0.0001 0.9902
## hdl     1    738.36 1267069 3536.7  0.2523 0.6157
## glu     1   3093.23 1264715 3535.9  1.0590 0.3040
```

```
drop1(MstepFtest, test="F")
```

```
## Single term deletions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + tch + ltg
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>              1267808 3535.0
## sex     1     44684 1312492 3548.3 15.2965 0.0001066 ***
## bmi     1    189976 1457784 3594.7 65.0331 7.248e-15 ***
## map     1     82152 1349960 3560.7 28.1225 1.818e-07 ***
## tc      1     26378 1294186 3542.1  9.0298 0.0028101 **
## ldl     1      7472 1275280 3535.6  2.5577 0.1104828
## tch     1      3686 1271494 3534.3  1.2619 0.2619190
## ltg     1    102520 1370328 3567.3 35.0950 6.399e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MstepFtest <- update(MstepFtest, . ~ . - tch)
add1(MstepFtest, test="F", scope=Mf)
```

```
## Single term additions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + ltg
##        Df Sum of Sq     RSS    AIC F value Pr(>F)
## <none>              1271494 3534.3
## age     1      10.9 1271483 3536.3  0.0037 0.9515
## hdl     1     394.8 1271099 3536.1  0.1348 0.7137
## tch     1    3686.2 1267808 3535.0  1.2619 0.2619
## glu     1    3532.6 1267961 3535.0  1.2091 0.2721
```

```
drop1(MstepFtest, test="F")
```

```
## Single term deletions
##
## Model:
## y ~ sex + bmi + map + tc + ldl + ltg
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>              1271494 3534.3
## sex     1     41856 1313350 3546.6  14.320 0.0001758 ***
## bmi     1    190592 1462086 3594.0  65.205 6.687e-15 ***
## map     1     79625 1351119 3559.1  27.241 2.787e-07 ***
## tc      1     65236 1336730 3554.4  22.318 3.123e-06 ***
## ldl     1     39377 1310871 3545.7  13.472 0.0002723 ***
## ltg     1    294092 1565586 3624.2 100.614 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(MstepFtest)
```

```
##
## Call:
## lm(formula = y ~ sex + bmi + map + tc + ldl + ltg, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -158.275  -39.476   -2.065   37.219  148.690
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -313.7666    25.3848 -12.360  < 2e-16 ***
## sex          -21.5910     5.7056  -3.784 0.000176 ***
## bmi            5.7111     0.7073   8.075 6.69e-15 ***
## map            1.1266     0.2158   5.219 2.79e-07 ***
## tc            -1.0429     0.2208  -4.724 3.12e-06 ***
## ldl            0.8433     0.2298   3.670 0.000272 ***
## ltg           73.3065     7.3083  10.031  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.06 on 435 degrees of freedom
## Multiple R-squared:  0.5149, Adjusted R-squared:  0.5082
## F-statistic: 76.95 on 6 and 435 DF,  p-value: < 2.2e-16
```

For the third model:

```
MforwardAIC <- step(Mf, direction="backward", trace=0, k=2, scope= list(lower=M0, upper=Mf))
summary(MforwardAIC)
```

```
##
## Call:
## lm(formula = y ~ sex + bmi + map + tc + ldl + ltg, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -158.275  -39.476   -2.065   37.219  148.690
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -313.7666    25.3848 -12.360  < 2e-16 ***
## sex          -21.5910     5.7056  -3.784 0.000176 ***
## bmi            5.7111     0.7073   8.075 6.69e-15 ***
## map            1.1266     0.2158   5.219 2.79e-07 ***
## tc            -1.0429     0.2208  -4.724 3.12e-06 ***
## ldl            0.8433     0.2298   3.670 0.000272 ***
## ltg           73.3065     7.3083  10.031  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.06 on 435 degrees of freedom
## Multiple R-squared:  0.5149, Adjusted R-squared:  0.5082
## F-statistic: 76.95 on 6 and 435 DF,  p-value: < 2.2e-16
```

And for the third model:

```
MstepBIC <- step(Mf, direction="both", trace=0, k=log(nrow(dat)), scope= list(lower=M0, upper=Mf))
summary(MstepBIC)
```

```
##
## Call:
## lm(formula = y ~ sex + bmi + map + tc + ldl + ltg, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -158.275  -39.476   -2.065   37.219  148.690
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -313.7666    25.3848 -12.360  < 2e-16 ***
## sex          -21.5910     5.7056  -3.784 0.000176 ***
## bmi            5.7111     0.7073   8.075 6.69e-15 ***
## map            1.1266     0.2158   5.219 2.79e-07 ***
## tc            -1.0429     0.2208  -4.724 3.12e-06 ***
## ldl            0.8433     0.2298   3.670 0.000272 ***
## ltg           73.3065     7.3083  10.031  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.06 on 435 degrees of freedom
## Multiple R-squared:  0.5149, Adjusted R-squared:  0.5082
## F-statistic: 76.95 on 6 and 435 DF,  p-value: < 2.2e-16
```

In this case all four models choose the same (remember this is not always the case). The chosen model includes `bmi` , `ltg` , `map` , `tc` , `sex` and `ldl` .