Polynomial Regression:

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \ldots + \beta_{p-1} x_i^{p-1} + \varepsilon_i, \quad i = 1, 2, \ldots, n,$$

where ε_i's are independent Normal$(0, \sigma^2)$.

1. It is well known that the sales response to advertising usually follows a curve reflecting the diminishing returns to advertising expenditure. As a company increases its advertising expenditure, sales increase, but the rate of increase drops continually after a certain point. If we consider company sales profits as a function of advertising expenditure, we find that the response function can be very well approximated by a second-order (quadratic) model. For a particular company, the data on monthly sales y and monthly advertising expenditure x, both in hundred thousand dollars, are given in the table on the right.

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + e_i \qquad n = 21.$$

Sales, y	Advert, x
5.0	1.0
6.0	1.8
6.5	1.6
7.0	1.7
7.5	2.0
8.0	2.0
10.0	2.3
10.8	2.8
12.0	3.5
13.0	3.3
15.5	4.8
15.0	5.0
16.0	7.0
17.0	8.1
18.0	8.0
18.0	10.0
18.5	8.0
21.0	12.7
20.0	12.0
22.0	15.0
23.0	14.4

```
> plot(sales.dat$Advert,sales.dat$Sales)
```

```
> sales.dat
   Sales Advert
1    5.0    1.0
2    6.0    1.8
3    6.5    1.6
4    7.0    1.7
5    7.5    2.0
6    8.0    2.0
7   10.0    2.3
8   10.8    2.8
9   12.0    3.5
10  13.0    3.3
11  15.5    4.8
12  15.0    5.0
13  16.0    7.0
14  17.0    8.1
15  18.0    8.0
16  18.0   10.0
17  18.5    8.0
18  21.0   12.7
19  20.0   12.0
20  22.0   15.0
21  23.0   14.4

> sales.fit = lm(Sales ~ Advert + I(Advert^2), data = sales.dat)
> summary(sales.fit)

Call:
lm(formula = Sales ~ Advert + I(Advert^2), data = sales.dat)

Residuals:
    Min      1Q  Median      3Q     Max
-1.9175 -0.8333 -0.1948  0.9292  2.1385

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.51505    0.73847   4.760 0.000157 ***
Advert       2.51478    0.25796   9.749 1.32e-08 ***
I(Advert^2) -0.08745    0.01658  -5.275 5.14e-05 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 1.228 on 18 degrees of freedom
Multiple R-Squared: 0.9587,     Adjusted R-squared: 0.9541
F-statistic:   209 on 2 and 18 DF,  p-value: 3.486e-13
```

```
> X = cbind(rep(1,21),sales.dat$Advert,sales.dat$Advert^2)
> X
      [,1] [,2]   [,3]
 [1,]    1  1.0   1.00
 [2,]    1  1.8   3.24
 [3,]    1  1.6   2.56
 [4,]    1  1.7   2.89
 [5,]    1  2.0   4.00
 [6,]    1  2.0   4.00
 [7,]    1  2.3   5.29
 [8,]    1  2.8   7.84
 [9,]    1  3.5  12.25
[10,]    1  3.3  10.89
[11,]    1  4.8  23.04
[12,]    1  5.0  25.00
[13,]    1  7.0  49.00
[14,]    1  8.1  65.61
[15,]    1  8.0  64.00
[16,]    1 10.0 100.00
[17,]    1  8.0  64.00
[18,]    1 12.7 161.29
[19,]    1 12.0 144.00
[20,]    1 15.0 225.00
[21,]    1 14.4 207.36
```

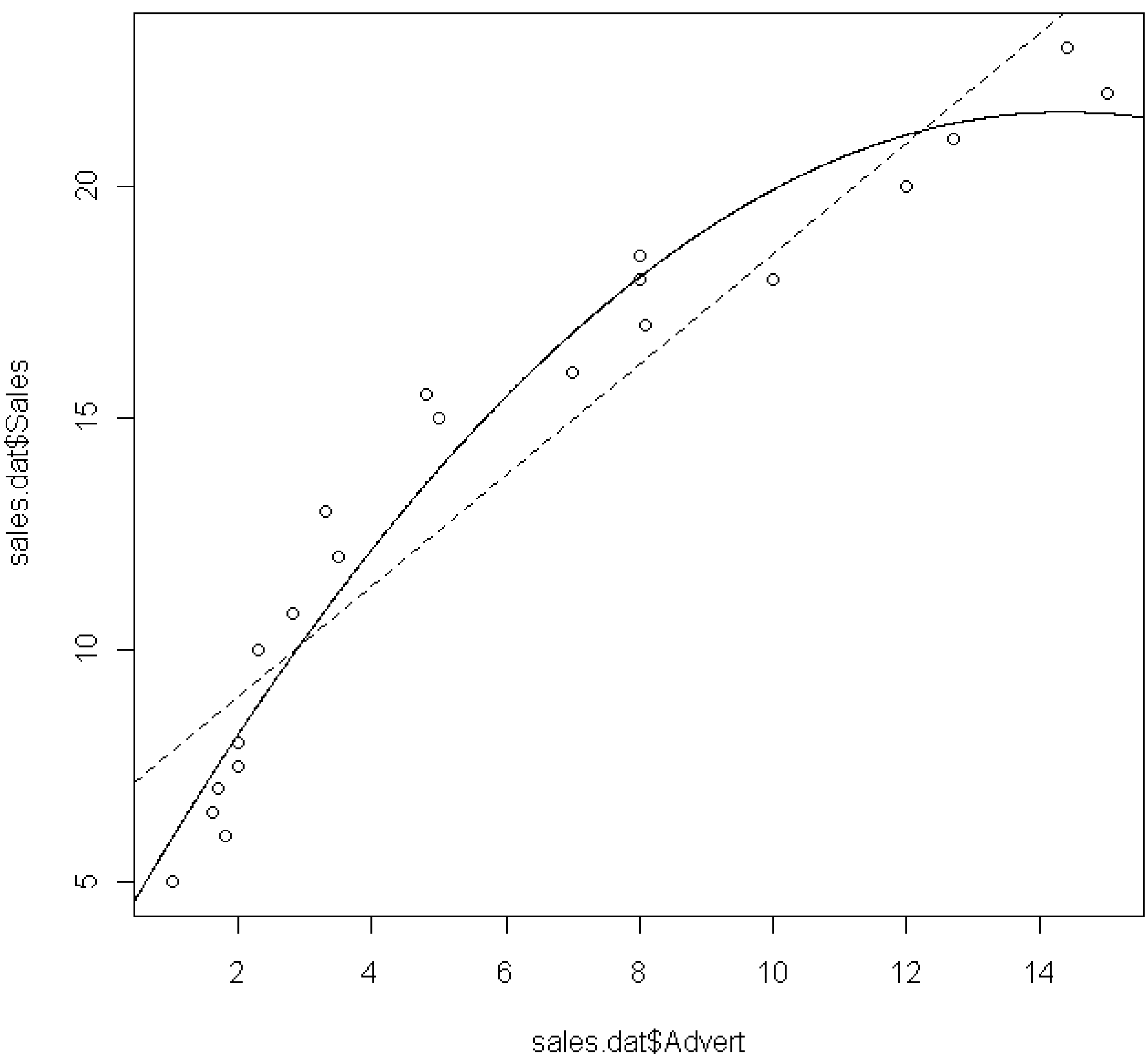
$$\mathbf{X}^{\mathrm{T}}\mathbf{X} = \begin{bmatrix} n & \sum x & \sum x^2 \\ \sum x & \sum x^2 & \sum x^3 \\ \sum x^2 & \sum x^3 & \sum x^4 \end{bmatrix},$$

```
> t(X) %*% X
        [,1]     [,2]      [,3]
[1,]   21.00   127.00   1182.26
[2,]  127.00  1182.26  13416.17
[3,] 1182.26 13416.17 166843.65
```

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{Y}$$

```
> solve(t(X) %*% X) %*% t(X)%*%sales.dat$Sales
            [,1]
[1,]  3.51504670
[2,]  2.51478201
[3,] -0.08745394
```

```
> plot(sales.dat$Advert,sales.dat$Sales)
> x = seq(0,16,by=0.01)
> y = sales.fit$coeff[1]+sales.fit$coeff[2]*x+sales.fit$coeff[3]*x^2
> lines(x,y)
> abline(lm(Sales~Advert,data=sales.dat),lty=2)
```



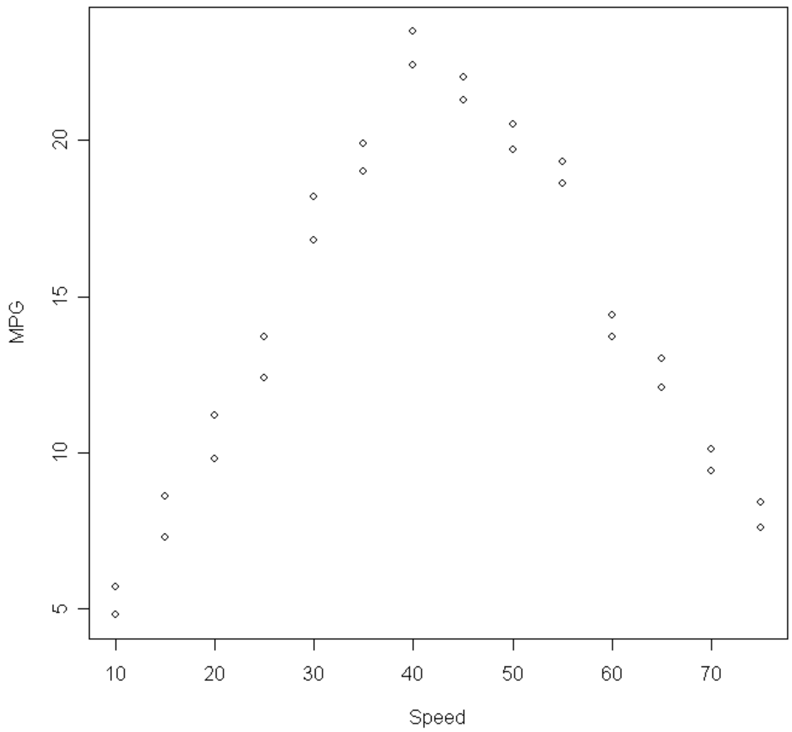
2. We wish to develop a model to predict miles per gallon based on highway speed for a particular brand of SUV. An experiment is designed in which a test car is driven at speeds ranging from 10 miles per hour to 75 miles per hour. The results are in the data file:

https://netfiles.uiuc.edu/stepanov/www/speed.csv

Fit a polynomial model and use it to predict the average mileage obtained when the car is driven at 55 miles per hour.

```
> speed
    MPG Speed
1   4.8    10
2   5.7    10
3   8.6    15
4   7.3    15
5   9.8    20
6  11.2    20
7  13.7    25
8  12.4    25
9  18.2    30
10 16.8    30
11 19.9    35
12 19.0    35
13 22.4    40
14 23.5    40
15 21.3    45
16 22.0    45
17 20.5    50
18 19.7    50
19 18.6    55
20 19.3    55
21 14.4    60
22 13.7    60
23 12.1    65
24 13.0    65
25 10.1    70
26  9.4    70
27  8.4    75
28  7.6    75
```

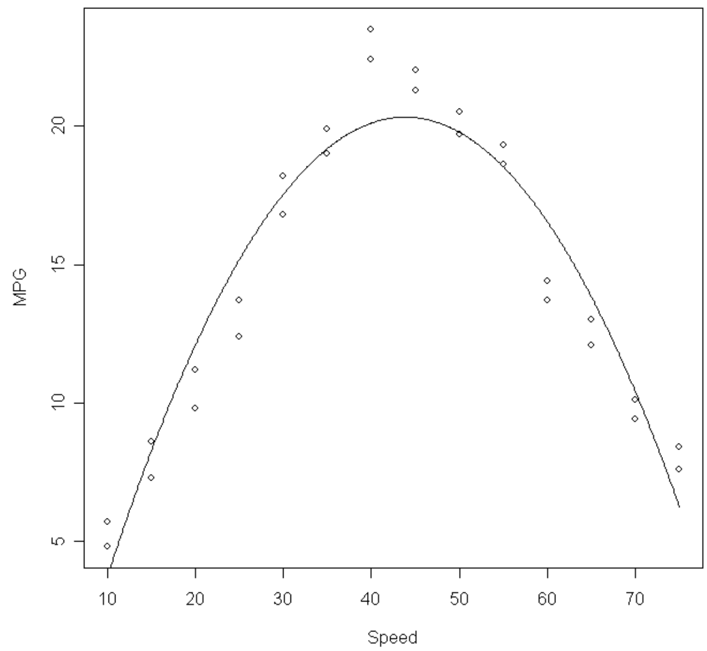
```
> attach(speed)
> plot(Speed, MPG)
```



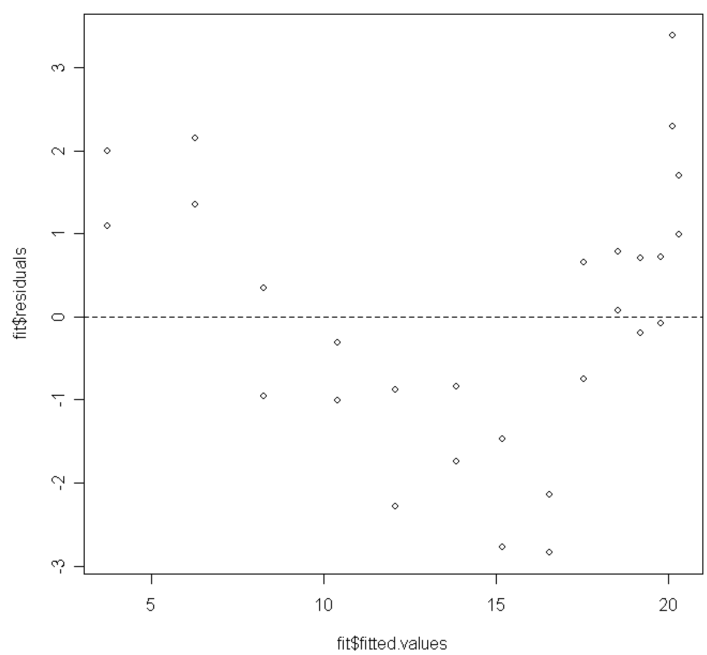

```
> fit = lm(MPG ~ Speed + I(Speed^2))
> summary(fit)

Call:
lm(formula = MPG ~ Speed + I(Speed^2))

Residuals:
      Min        1Q    Median        3Q       Max
-2.841126 -0.969354  0.001676  1.018149  3.390000

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.5555495  1.4241091  -5.305 1.69e-05 ***
Speed        1.2716937  0.0757321  16.792 3.99e-15 ***
I(Speed^2)  -0.0145014  0.0008719 -16.633 4.97e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.663 on 25 degrees of freedom
Multiple R-Squared: 0.9188,     Adjusted R-squared: 0.9123
F-statistic: 141.5 on 2 and 25 DF,  p-value: 2.338e-14
```

```
> X = model.matrix(fit)
> X
   (Intercept) Speed I(Speed^2)
1            1    10        100
2            1    10        100
3            1    15        225
4            1    15        225
5            1    20        400
6            1    20        400
7            1    25        625
8            1    25        625
9            1    30        900
10           1    30        900
11           1    35       1225
12           1    35       1225
13           1    40       1600
14           1    40       1600
15           1    45       2025
16           1    45       2025
17           1    50       2500
18           1    50       2500
19           1    55       3025
20           1    55       3025
21           1    60       3600
22           1    60       3600
23           1    65       4225
24           1    65       4225
25           1    70       4900
26           1    70       4900
27           1    75       5625
28           1    75       5625
attr(,"assign")
[1] 0 1 2


> t(X) %*% X
            (Intercept)   Speed I(Speed^2)
(Intercept)          28    1190      61950
Speed              1190   61950    3599750
I(Speed^2)        61950 3599750  222888750

> t(X) %*% MPG
                [,1]
(Intercept)    403.4
Speed        17589.0
I(Speed^2)  877520.0

> solve(t(X)%*%X) %*% t(X)%*%MPG
                   [,1]
(Intercept) -7.55554945
Speed        1.27169368
I(Speed^2)  -0.01450137
```

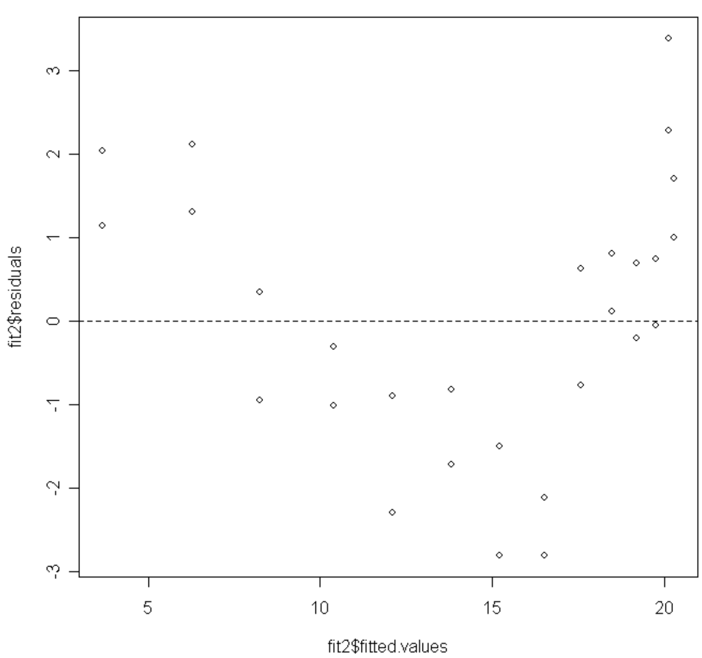
```
> x = seq(10,75,by=0.1)
> y = fit$coefficients[1] + fit$coefficients[2]*x + fit$coefficients[3]*x^2
> lines(x,y)
```



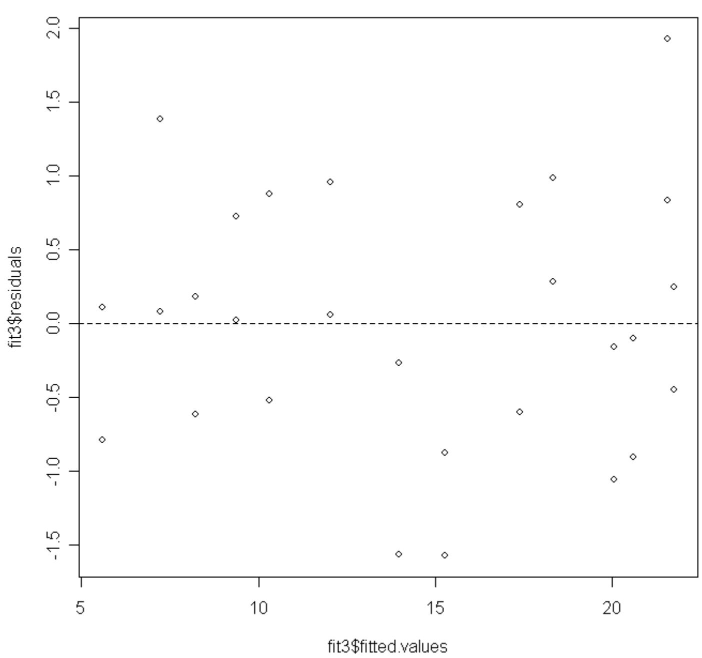

```
> plot(fit$fitted.values,fit$residuals)
> abline(h=0,lty=2)
```



```
> fit2 = lm(MPG ~ Speed + I(Speed^2) + I(Speed^3))
> summary(fit2)

Call:
lm(formula = MPG ~ Speed + I(Speed^2) + I(Speed^3))

Residuals:
     Min       1Q   Median       3Q      Max
-2.81124 -0.96768  0.02637  1.03454  3.38268

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.742e+00  2.768e+00  -2.797   0.0100 **
Speed        1.291e+00  2.529e-01   5.103  3.2e-05 ***
I(Speed^2)  -1.502e-02  6.604e-03  -2.274   0.0322 *
I(Speed^3)   4.066e-06  5.132e-05   0.079   0.9375
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.697 on 24 degrees of freedom
Multiple R-Squared: 0.9188,     Adjusted R-squared: 0.9087
F-statistic: 90.56 on 3 and 24 DF,  p-value: 3.170e-13



> plot(fit2$fitted.values,fit2$residuals)
> abline(h=0,lty=2)
```

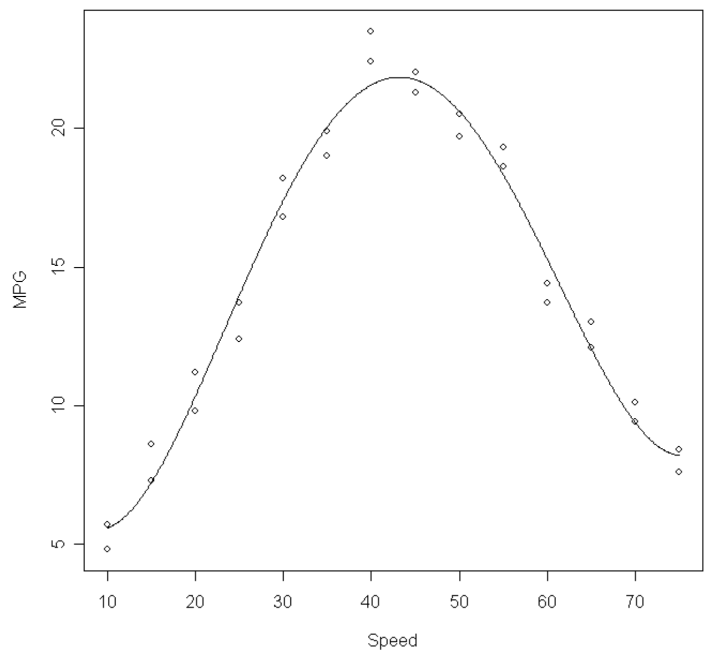


```
> fit3 = lm(MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4))
> summary(fit3)

Call:
lm(formula = MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4))

Residuals:
     Min       1Q   Median       3Q      Max 
-1.57410 -0.60308  0.04236  0.74481  1.93038 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.146e+01  2.965e+00   3.866 0.000785 ***
Speed       -1.468e+00  3.913e-01  -3.751 0.001042 ** 
I(Speed^2)   1.081e-01  1.673e-02   6.463 1.35e-06 ***
I(Speed^3)  -2.130e-03  2.844e-04  -7.488 1.31e-07 ***
I(Speed^4)   1.255e-05  1.665e-06   7.539 1.17e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 0.9307 on 23 degrees of freedom
Multiple R-Squared: 0.9766,     Adjusted R-squared: 0.9726 
F-statistic: 240.2 on 4 and 23 DF,  p-value: < 2.2e-16 

> plot(fit3$fitted.values,fit3$residuals)
> abline(h=0,lty=2)
```

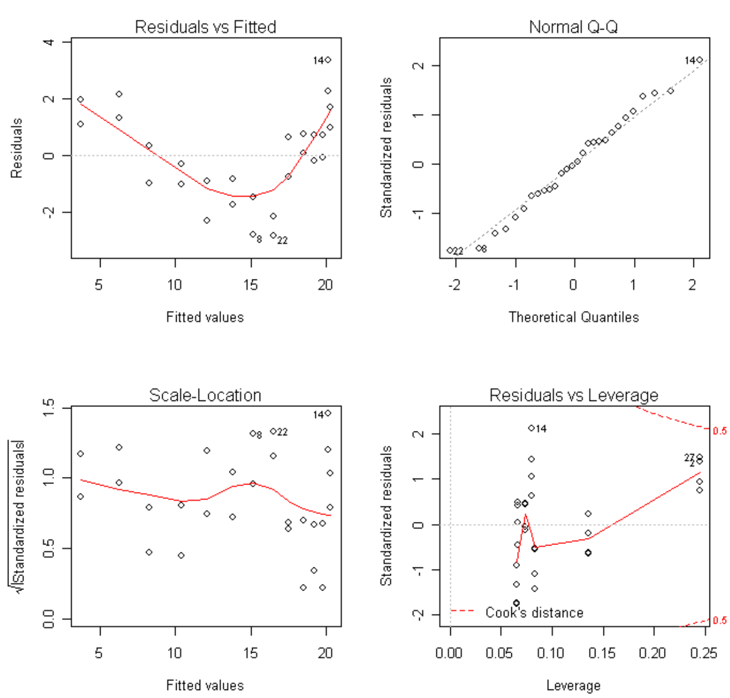



```
> predict.lm(fit3,data.frame(Speed=55),interval=c("prediction"),level=0.95)
          fit      lwr      upr
[1,] 18.31717 16.28547 20.34887
```

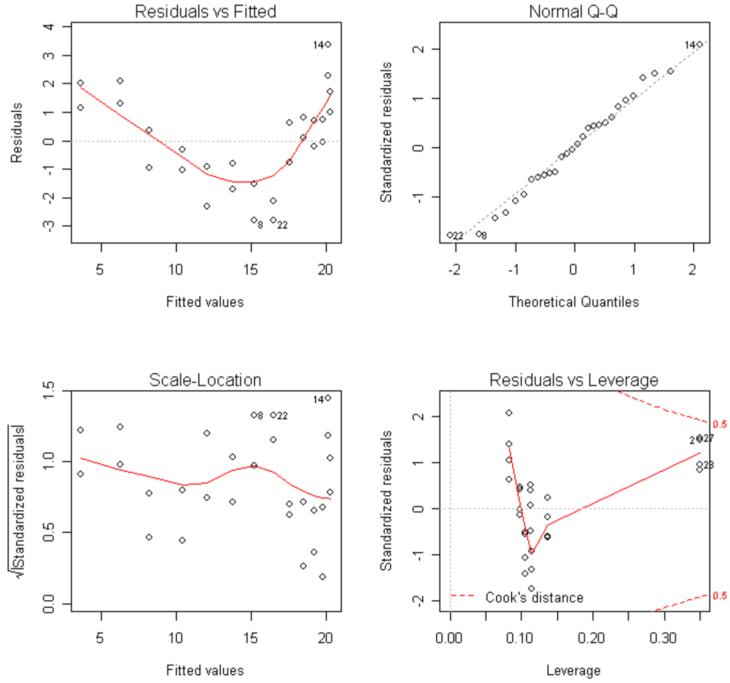
```
> plot(Speed, MPG)
> y3 = fit3$coeff[1] + fit3$coeff[2]*x + fit3$coeff[3]*x^2 +
fit3$coeff[4]*x^3 + fit3$coeff[5]*x^4
> lines(x,y3)
```



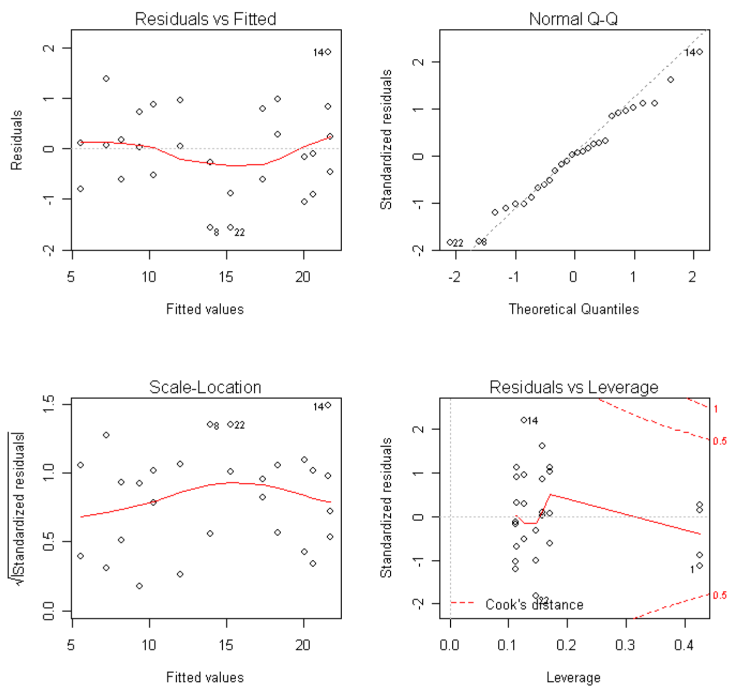

```
> par(mfrow=c(2,2))
> plot(fit)
```



```
> plot(fit2)
```

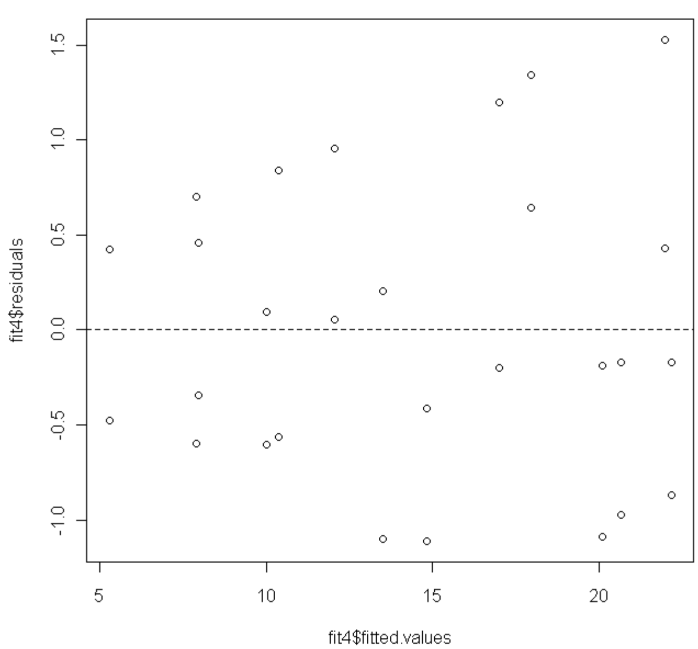



```
> plot(fit3)
```

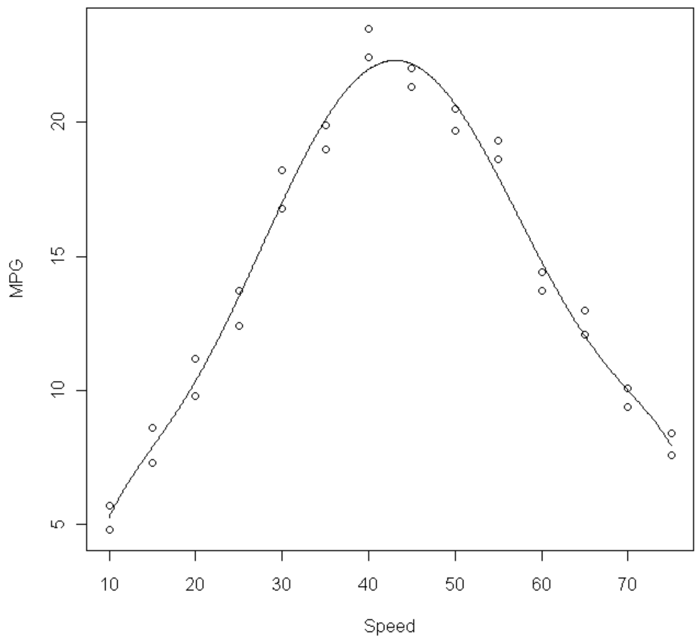



```
> shapiro.test(fit3$residuals)

        Shapiro-Wilk normality test

data:  fit3$residuals
W = 0.9822, p-value = 0.9
```

```
> fit4 = lm(MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4) + I(Speed^5)
+ I(Speed^6))
> summary(fit4)

Call:
lm(formula = MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4) +
    I(Speed^5) + I(Speed^6))

Residuals:
    Min      1Q  Median      3Q     Max
-1.1129 -0.5717 -0.1707  0.5025  1.5288

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.421e+01  1.204e+01  -1.180   0.2514
Speed        4.203e+00  2.553e+00   1.646   0.1146
I(Speed^2)  -3.521e-01  2.012e-01  -1.750   0.0947 .
I(Speed^3)   1.579e-02  7.691e-03   2.053   0.0527 .
I(Speed^4)  -3.473e-04  1.529e-04  -2.271   0.0338 *
I(Speed^5)   3.585e-06  1.518e-06   2.362   0.0279 *
I(Speed^6)  -1.402e-08  5.941e-09  -2.360   0.0280 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8657 on 21 degrees of freedom
Multiple R-squared: 0.9815,     Adjusted R-squared: 0.9762
F-statistic:   186 on 6 and 21 DF,  p-value: < 2.2e-16

> plot(fit4$fitted.values,fit4$residuals)
> abline(h=0,lty=2)
```



```
> plot(Speed, MPG)
> y4 = fit4$coeff[1] + fit4$coeff[2]*x + fit4$coeff[3]*x^2 +
fit4$coeff[4]*x^3 + fit4$coeff[5]*x^4 + fit4$coeff[6]*x^5 +
fit4$coeff[7]*x^6
> lines(x,y4)
```




```
> anova(fit3,fit4)
Analysis of Variance Table

Model 1: MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4)
Model 2: MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4) + I(Speed^5) +
    I(Speed^6)
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     23 19.9215
2     21 15.7387  2    4.1828 2.7905 0.0842 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> anova(fit,fit3)
Analysis of Variance Table

Model 1: MPG ~ Speed + I(Speed^2)
Model 2: MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1     25 69.174
2     23 19.922  2    49.252 28.432 6.066e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> fit5 = lm(MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4) + I(Speed^5)
+ I(Speed^6) + I(Speed^7) + I(Speed^8))
> summary(fit5)

Call:
lm(formula = MPG ~ Speed + I(Speed^2) + I(Speed^3) + I(Speed^4) +
    I(Speed^5) + I(Speed^6) + I(Speed^7) + I(Speed^8))

Residuals:
     Min       1Q   Median       3Q      Max
-1.21938 -0.50464 -0.09105  0.49029  1.45440

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.202e+01  7.045e+01  -0.313    0.758
Speed        6.021e+00  2.014e+01   0.299    0.768
I(Speed^2)  -5.037e-01  2.313e+00  -0.218    0.830
I(Speed^3)   2.121e-02  1.408e-01   0.151    0.882
I(Speed^4)  -4.008e-04  5.017e-03  -0.080    0.937
I(Speed^5)   1.789e-06  1.080e-04   0.017    0.987
I(Speed^6)   4.486e-08  1.381e-06   0.032    0.974
I(Speed^7)  -6.456e-10  9.649e-09  -0.067    0.947
I(Speed^8)   2.530e-12  2.835e-11   0.089    0.930

Residual standard error: 0.9034 on 19 degrees of freedom
Multiple R-squared: 0.9818,     Adjusted R-squared: 0.9741
F-statistic: 128.1 on 8 and 19 DF,  p-value: 7.074e-15
```

3. cent.dat contains data on x = cure temperature (°F) and y = ultimate shear strength of rubber compound (psi). Read the data into a dataframe in R and fit a quadratic model.

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \varepsilon_i \qquad\qquad Y_i = \beta_0^* + \beta_1^* (x_i - \bar{x}) + \beta_2^* (x_i - \bar{x})^2 + \varepsilon_i$$

The quadratic parameters are identical $(\beta_2 = \beta_2^*)$. The estimated standard deviations indicate that β_0^* and β_1^* have been more accurately estimated than β_0 and β_1.

When the x_i's all lie far from 0, it is helpful to center the x values to gain the computational accuracy, not only in quadratic but also in higher-degree models.

```
> cent.dat
    x   y
1 280 770
2 284 800
3 292 840
4 295 810
5 298 735
6 304 640
7 308 590
8 315 560

> xbar = mean(cent.dat[,1])
> xbar
[1] 297

> cent.dat$xcent = cent.dat$x - xbar
> cent.dat
    x   y xcent
1 280 770   -17
2 284 800   -13
3 292 840    -5
4 295 810    -2
5 298 735     1
6 304 640     7
7 308 590    11
8 315 560    18

> cent.fit1 = lm(y ~ x + I(x^2), cent.dat)
> summary(cent.fit1)

Call:
lm(formula = y ~ x + I(x^2), data = cent.dat)
```

```
Residuals:
     1      2      3      4      5      6      7      8 
-24.09  -1.93  52.91  38.97 -14.25 -48.53 -45.33  42.24 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.503e+04  1.241e+04  -2.016   0.0998 .
x            1.812e+02  8.364e+01   2.167   0.0825 .
I(x^2)      -3.179e-01  1.407e-01  -2.259   0.0734 .
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1 

Residual standard error: 47.54 on 5 degrees of freedom
Multiple R-Squared: 0.8596,     Adjusted R-squared: 0.8035 
F-statistic: 15.31 on 2 and 5 DF,  p-value: 0.007383 

> cent.fit2 = lm(y ~ xcent + I(xcent^2), cent.dat)
> summary(cent.fit2)

Call:
lm(formula = y ~ xcent + I(xcent^2), data = cent.dat)

Residuals:
     1      2      3      4      5      6      7      8 
-24.09  -1.93  52.91  38.97 -14.25 -48.53 -45.33  42.24 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 757.1458    24.1013  31.415 6.14e-07 ***
xcent        -7.5775     1.5175  -4.993  0.00413 ** 
I(xcent^2)   -0.3179     0.1407  -2.259  0.07344 .  
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1 

Residual standard error: 47.54 on 5 degrees of freedom
Multiple R-Squared: 0.8596,     Adjusted R-squared: 0.8035 
F-statistic: 15.31 on 2 and 5 DF,  p-value: 0.007383 

> cent.dat$x2 = cent.dat$x^2
> cent.dat
    x   y xcent    x2
1 280 770   -17 78400
2 284 800   -13 80656
3 292 840    -5 85264
4 295 810    -2 87025
5 298 735     1 88804
6 304 640     7 92416
7 308 590    11 94864
8 315 560    18 99225

> cor(cent.dat$x,cent.dat$x2)
[1] 0.9998355
```