Third Assignment
(Make function for Multiple Linear Regression Analysis MyReg( ))

Exploratory Data Analysis & Statistical Consulting

200903877 황 성 윤
Department of Statistics

Function MyReg()

주어진 자료에서 반응변수와 설명변수들에 관한 다중회귀분석(Multiple Linear Regression)을 실시하여 해당하는 통계량들을 계산하여 화면에 보여주고, 회귀분석 결과에 따른 산점도, 실제 반응변수의 값과 회귀모형에 의해 예측된 값을 비교하는 그림, 예측치와 표준화된 잔차에 대한 그래프, 그리고 표준화된 잔차의 절대값에 대하여 퍼진 정도를 알아보는 그래프를 그려서 보여준다.

Case 1

input

```
N <- 100
set.seed(1234)
tX1 <- rnorm(N,0,1)
tX2 <- rnorm(N,3,1)
tX3 <- rgamma(N,1,3)
tX4 <- sample(c(0,1),N,replace=T)
tX5 <- sample(1:3,N,replace=T,prob=c(1,2,3)/6)
tX6 <- rbinom(N,4,0.3)
tX <- cbind(rep(1,length(tX1)),tX1,tX2,tX3,tX4,tX5,tX6)
tbeta <- (0:6)/5
ty <- as.vector(tX%*%tbeta)+rnorm(N,0,2)
dat <- data.frame(y=ty, age=tX1, height=tX2, weight=tX3, smoking=tX4,
                  therapy=tX5, surgery=tX6)


MyReg <- function(data) {
  n <- nrow(data) ; p <- ncol(data)-1
  y <- as.matrix(data)[,1] ; X <- cbind(rep(1,n),as.matrix(data)[,-1])
  beta <- (solve(t(X)%*%X))%*%(t(X))%*%(y)
  pred <- X%*%beta
  residuals <- y-pred ; sig2 <- (t(residuals)%*%residuals)/(n-p-1)
  H <- (X)%*%(solve(t(X)%*%X))%*%(t(X))
  stdresid <- rep(0,n)
  for(i in 1:n) {
    stdresid[i] <- residuals[i]/sqrt(sig2*(1-H[i,i]))
  }
  SST <- sum((y-mean(y))^2) ; SSE <- sum((y-pred)^2) ; SSR <- SST - SSE
  MSR <- SSR/(p) ; MSE <- SSE/(n-p-1) ; F <- MSR/MSE
  P.value <- 1-pf(F,p,n-p-1) ; Rsq <- SSR/SST
  C <- solve(t(X)%*%X)
  std.error <- rep(0,p+1)
  for(i in 1:(p+1)) {
    std.error[i] <- sqrt(sig2*C[i,i])
  }
  t.value <- rep(0,p+1)
  for(i in 1:(p+1)) {
```

```
  t.value[i] <- beta[i]/std.error[i]
}
Pr.t <- rep(0,(p+1))
for(i in 1:(p+1)) {
  Pr.t[i] <- 2*(1-pt(abs(t.value[i]), n-p-1))
}
args <- list(Data=data, Beta=beta, Sigma2=sig2, Predict=pred,
             Residuals=residuals, Standard.Residuals=stdresid, SSR=SSR,
             SSE=SSE, F.Statistic=F, P.value=P.value)

aster <- function(p) {
  if (p >= 0.05 & p < 0.1) "."
  else if (p >= 0.01 & p < 0.05) "*"
  else if (p >= 0.001 & p < 0.1) "**"
  else if (p < 0.001) "***"
  else " "
}

cat("\n == ANALYSIS OF VARIANCE == \n\n",
    encodeString(c("Source", "    df", "    SS", "    MS", "    F", "    P-value"),
                 width=8, justify="right"), "\n",
    "-----------------------------------------------------------------", "\n",
    encodeString(c("Regression", p, round(SSR,2), round(MSR,2), round(F,2),
                   round(P.value,2)), width=8, justify="right"),
    aster(P.value), "\n", encodeString(c("       Error", n-p-1, round(SSE,2),
                                         round(MSE,2)), width=8, justify="right"),
    "\n", encodeString(c("       Total", n-1, round(SST,2)), width=8,
                       justify="right"),
    "\n",
    "-----------------------------------------------------------------", "\n",
    "Estimated error variance :", round(sig2,4), "\n","R-squares :",round(Rsq,4),
    "\n\n")

ind.name <- colnames(data)
ind.name[1] <- "(Intercept)"
test.mat <- cbind(ind.name, round(beta,4), round(std.error,4), round(t.value,4),
                  round(Pr.t,4))
cat("\n\n\n== PARAMETER ESTIMATES ==\n\n",
    encodeString(c(" ", "Estimate", "Std.Error", "t value", "Pr(>|t|)"), width=11,
                 justify="right"),
    "\n")
for(i in 1:nrow(test.mat)) {
  cat(encodeString(test.mat[i,], width=11, justify="right"), aster(Pr.t[i]),"\n")
}

cat("\n", "===Various Statistics in Multiple Regression===", "\n\n")
print(args)

outlier1 <- rep(0,n)
outlier2 <- rep(0,n)
std.sign <- rep(0,n)
```

```
for(i in 1:n) {
  outlier1[i] <- ifelse(abs(stdresid[i]) > 2, "out", "in")
  outlier2[i] <- ifelse(abs(stdresid[i]) > 2, 2, 1)
  std.sign[i] <- ifelse(stdresid[i] > 0, 24, 20)
}
abs.stdresid <- abs(stdresid)
ex1 <- as.data.frame(cbind(number=(1:n), stdresid=stdresid,
                           pred=pred, y=y,
                           outlier1=outlier1, outlier2=outlier2,
                           std.sign=std.sign, abs.stdresid=abs.stdresid))
ex1_1 <- as.matrix(ex1[,1:4,8])
ex1_1 <- as.numeric(ex1_1)
ex1_1 <- matrix(ex1_1,nrow=n,ncol=5)
colnames(ex1_1) <- c("number","stdresid","pred","y","abs.stdresid")
ex2 <- as.data.frame(ex1_1)
ex2 <- ex2[abs.stdresid > 2,]

pred1 <- as.vector(pred) ; abs.stdresid1 <- as.vector(abs.stdresid)
ord <- order(pred1,abs.stdresid1)
re1 <- as.data.frame(t(rbind(pred1,abs.stdresid1)[,ord]))

KDE <- function(x) {
  w <- rep(0,n)
  for (i in 1:n) {
    w[i] <- exp(-(pred[i]-x)^2)/sqrt(2*pi)
  }
  ans <- sum(w*abs(stdresid))/sum(w)
  return(ans)
}
d1 <- seq(min(re1$pred1),max(re1$pred1),0.01)
d2 <- rep(0,length(d1))
for (j in 1:length(d1)) {
  d2[j] <- KDE(d1[j])
}
d3 <- re1$pred1
d4 <- rep(0,length(d3))
for (j in 1:length(d3)) {
  d4[j] <- mean(re1$abs.stdresid1[j:(j+9)],na.rm=TRUE)
}

pairs(data, main="Scatterplot matrix from given data", cex=outlier2,
      pch=21, bg=c("yellow","red")[unclass(ex1$outlier1)])
dev.new()
plot(y~pred, main="Dependent variable VS Fitted values",
     xlab="Fitted response values",
     ylab="Response variables", cex=abs(stdresid)*7, pch=20,
     col=factor(outlier2, labels=c("black", "red")))
abline(0,1,col="blue",lwd=2)
for (i in 1:nrow(ex2)) {
  text(ex2$pred[i], ex2$y[i], ex2$number[i], col="light blue")
}
```

```
  dev.new()
  par(fig=c(0,0.7,0,1), new=FALSE)
  plot(stdresid~pred, main="Standard residuals VS predicted values",
       xlab="Fitted response values", ylab="Studentized residuals",
       cex=abs(stdresid)*3, pch=20,
       col=factor(outlier2, labels=c("black", "orange")))
       abline(h=c(-2,0,2),col="blue",lty=c(2,1,2))
  for (i in 1:nrow(ex2)) {
    text(ex2$pred[i], ex2$stdresid[i], ex2$number[i], col="yellow")
  }
  par(fig=c(0.7,1,0,1),new=TRUE)
  boxplot(stdresid,col="green")
  for (i in 1:6) {
    text(0.7,summary(stdresid)[i],round(summary(stdresid)[i],3),col="purple")
  }
  dev.new()
  par(fig=c(0,1,0,1),new=FALSE)
  plot(abs.stdresid~pred,
       main="Absolute Standard residuals VS predicted values",
       xlab="Fitted response values",
       ylab="Absolute studentized residuals", cex=1.5, pch=std.sign)
  legend("topright",c("positive","negative"),pch=c(24,20))
  lines(d1,d2,lty=3,lwd=2,col="red")
  lines(d3,d4,lty=2,lwd=2,col="blue")
  legend("topleft",c("KDE","MA10"),lty=c(3,2),lwd=2,col=c("red","blue"))
  dev.new()
  qqnorm(stdresid) ; qqline(stdresid)
  test.res <- shapiro.test(stdresid)
  print(test.res)

  return(args)
}

res1 <- MyReg(dat)

str(res1)

attributes(res1)
```

output

```
== ANALYSIS OF VARIANCE ==

   Source        df       SS        MS        F     P-value
 ------------------------------------------------------------------
 Regression        6     195.3     32.55     8.72        0 ***
      Error       93    347.11      3.73
      Total       99    542.41
 ------------------------------------------------------------------
 Estimated error variance : 3.7323
 R-squares : 0.3601
```

== PARAMETER ESTIMATES ==

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	-0.7866	1.182	-0.6655	0.5074	
age	0.0198	0.1951	0.1012	0.9196	
height	0.5986	0.1953	3.0653	0.0028	**
weight	-0.0803	0.6073	-0.1322	0.8951	
smoking	0.2927	0.4105	0.713	0.4776	
therapy	1.2801	0.2794	4.5809	0	***
surgery	1.0944	0.2018	5.4219	0	***

===Various Statistics in Multiple Regression===

$Data

	y	age	height	weight	smoking	therapy	surgery
1	7.6185092	-1.207065749	3.4145235	0.300656147	1	2	3
2	11.1395846	0.277429242	2.5252815	0.371353870	0	3	2
3	5.7838752	1.084441177	3.0659935	0.213260667	0	2	1
4	6.2143115	-2.345697703	2.4975222	0.372750095	1	2	1
5	7.7605163	0.429124689	2.1740014	0.248226012	0	2	1
6	7.7740276	0.506055892	3.1669893	0.394101281	0	3	1
7	2.0073677	-0.574739960	2.1037354	0.883828859	0	2	1
8	6.8965796	-0.546631856	3.1681854	0.531965694	1	2	1
9	7.7153296	-0.564451999	3.3549683	0.174454546	0	3	3
10	8.7708591	-0.890037829	2.9478949	0.007488234	1	1	2
11	5.1149679	-0.477192700	2.8040654	0.626116131	0	3	1
12	1.3942546	-0.998386445	2.3509302	1.179623070	1	1	1
13	8.0460598	-0.776253895	1.8902328	0.102581133	1	2	3
14	6.5964155	0.064458817	3.8492742	0.224677931	0	3	1
15	7.1039366	0.959494059	3.0223625	0.563909112	0	3	2
16	1.4979346	-0.110285494	3.8311406	0.167726840	0	2	0
17	3.2428745	-0.511009506	1.7557121	0.201034863	0	3	1
18	6.2127608	-0.911195417	3.1690264	0.254401211	1	3	1
19	8.1941169	-0.837171680	3.6731663	0.693440641	0	2	2
20	6.0718540	2.415835178	2.9737236	0.213521547	1	3	1
21	3.4667472	0.134088220	2.8086078	0.098975684	1	2	1
22	0.4867931	-0.490685897	2.2180934	0.176831684	1	3	1
23	4.7385541	-0.440547872	5.0581620	0.024831943	1	1	0
24	4.5970284	0.459589441	3.7505015	0.250382061	0	2	0
25	7.8639016	-0.693720247	4.8242083	0.757634302	0	3	0
26	6.9645135	-1.448204910	3.0800596	0.147808868	0	3	0
27	5.0848131	0.574755721	2.3685907	0.180896373	0	3	1
28	8.7539894	-1.023655723	1.4867119	0.278347274	0	2	3
29	8.3501663	-0.015138300	2.3639002	0.008344596	1	3	3
30	5.3399869	-0.935948601	3.2263015	0.315793550	0	3	2
31	5.1378487	1.102297546	4.0136903	0.013843684	1	3	1
32	5.0961565	-0.475593079	3.2527501	0.640095477	0	3	1

33	5.0096718	-0.709440038	1.8280517	0.275633695	1	3	0
34	4.4533125	-0.501258061	3.6687143	0.234801962	0	3	0
35	4.9573960	-1.629093469	1.3498991	0.235832515	0	3	1
36	5.3893573	-1.167619262	2.6341478	0.285561219	1	2	1
37	7.7728607	-2.180039649	2.6838817	0.510195492	0	3	2
38	3.9803126	-1.340993192	1.0517540	0.290288645	0	3	1
39	7.9956412	-0.294293859	3.9200575	0.359636780	0	3	0
40	5.5369194	-0.465897540	2.3771284	0.006432923	1	3	0
41	3.2315194	1.449496265	2.6659634	0.770510833	1	3	0
42	6.7268347	-1.068642724	4.3951479	0.226339440	0	2	3
43	3.5572367	-0.855364634	3.6366744	0.403122869	0	1	0
44	3.9056203	-0.280623002	2.8915683	1.045380347	1	1	1
45	5.7657251	-0.994340076	3.5137628	0.115163135	1	3	0
46	6.8175870	-0.968514318	3.3992718	1.304649095	1	2	1
47	6.4152961	-1.107318193	4.6628564	0.274436277	0	2	0
48	4.8616932	-1.251985886	3.2758934	0.175208053	1	3	1
49	5.4079627	-0.523828119	3.5062726	0.239968258	1	2	1
50	4.1239824	-0.496849957	3.3475520	0.032860857	1	2	1
51	5.3127850	-1.806031257	2.6227624	0.465253488	0	3	3
52	1.2245240	-0.582075925	3.0976195	0.999719975	0	2	1
53	6.4538941	-1.108889624	4.6387446	0.048526795	0	2	2
54	7.2596023	-1.014962009	2.1244075	0.016168668	0	3	2
55	7.5936876	-0.162309524	3.1217600	0.165215369	1	3	1
56	1.5583003	0.563055819	4.3621307	0.468037451	0	1	1
57	3.0788377	1.647817473	2.7653789	1.170797202	0	2	1
58	0.4581519	-0.773353424	1.9466172	0.527473505	1	1	0
59	3.3099758	1.605909629	2.1302164	0.523417265	0	2	1
60	5.2972146	-1.157808548	2.6098730	0.240719133	1	2	1
61	7.4377150	0.656588464	2.1526499	0.093950833	1	2	2
62	5.4576686	2.548991071	2.7393606	0.136985760	1	1	3
63	-1.9376925	-0.034760390	2.5855803	0.939535162	0	1	1
64	9.2465962	-0.669633580	2.8169492	0.243493845	1	3	2
65	7.3245442	-0.007604756	3.4070561	0.554455368	0	2	2
66	3.3409581	1.777084448	3.6246331	0.229048919	1	2	0
67	5.8706458	-1.138607737	4.6782057	0.290679589	1	3	2
68	6.0322781	1.367827179	2.9313063	0.310287531	1	3	1
69	4.7213034	1.329564791	2.6791601	1.945855520	0	2	0
70	7.5783331	0.336472797	4.4710057	0.127057828	1	1	2
71	7.6815165	0.006892838	4.7043294	0.030359993	0	3	0
72	4.4577125	-0.455468738	3.0432440	0.048792384	1	3	1
73	7.7402535	-0.366523933	2.6673427	1.275292553	0	3	2
74	9.7061666	0.648286568	1.1777646	0.395085718	1	3	2
75	3.7923051	2.070270861	4.4112624	0.694050412	0	3	0
76	2.3613461	-0.153398412	2.1624176	0.287642700	1	1	0
77	3.1786706	-1.390700947	1.8762372	0.082086647	1	3	2
78	5.1042255	-0.723581777	6.0437659	0.760985052	1	1	0
79	5.2019065	0.258261762	3.2350213	0.231048529	1	3	1
80	2.6156518	-0.317059115	2.9667414	0.237070641	0	2	1
81	5.2341128	-0.177789958	0.2677805	0.204189201	1	1	4
82	6.4998278	-0.169994077	2.9002094	0.474560230	1	1	2
83	-0.9153396	-1.372301886	3.9760317	0.489410169	1	1	0

```
84     5.9297626  -0.173787170  3.4138689  0.083323763        0        3        2
85     2.6109900   0.850232257  3.9123222  0.172416743        1        2        0
86     6.4501599   0.697608712  4.9837322  0.068972247        1        3        2
87     8.4683504   0.549997351  4.1691085  0.569063747        0        3        2
88     4.8777452  -0.402731975  2.4912630  0.430463151        0        3        0
89     7.8100242  -0.191593770  3.7041802  0.301186011        1        2        3
90     5.2398306  -1.194527880  2.8015837  0.138446892        0        2        1
91     5.2926870  -0.053158819  2.4619292  0.537928137        1        2        2
92     2.6914147   0.255196001  0.1442413  0.048790497        0        3        3
93     7.0006510   1.705964007  2.2103531  0.777457371        0        3        0
94     5.5954422   1.001513252  3.4878146  0.102332856        0        2        2
95     7.2981776  -0.495583443  5.1680325  0.767098073        0        2        3
96     4.8584337   0.355550297  3.5006946  0.120016478        1        1        1
97     7.0215120  -1.134608044  3.6202102  0.780808501        1        3        0
98     8.6139680   0.878203627  2.0340968  0.287524310        0        3        3
99     7.4413886   0.972916753  3.1626547  0.545354279        1        3        0
100    3.9999188   2.121117105  0.9217625  0.191202086        0        3        0

$Beta
               [,1]
        -0.78660499
age      0.01975009
height   0.59862082
weight  -0.08026628
smoking  0.29271521
therapy  1.28005300
surgery  1.09439426

$Sigma2
         [,1]
[1,] 3.732336

$Predict
           [,1]
  [1,] 7.345432
  [2,] 6.729701
  [3,] 4.707563
  [4,] 4.579432
  [5,] 4.157849
  [6,] 6.022136
  [7,] 4.044942
  [8,] 5.003657
  [9,] 8.319940
 [10,] 4.721444
 [11,] 5.766840
 [12,] 3.173471
 [13,] 6.457367
 [14,] 6.435443
 [15,] 7.025279
 [16,] 4.051261
 [17,] 5.172725
```

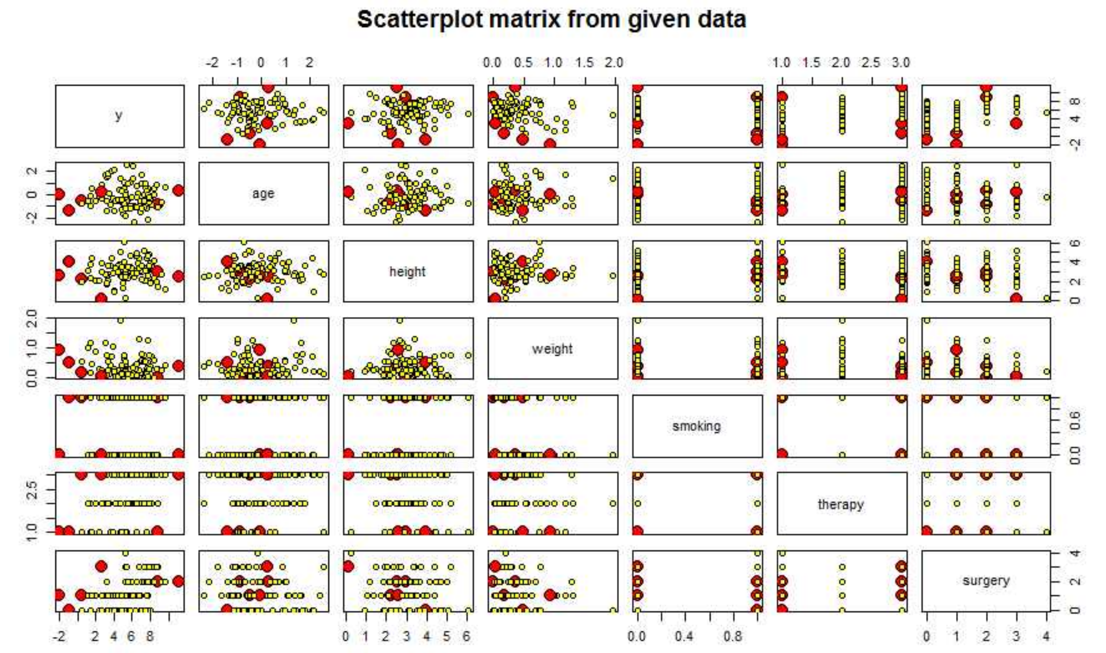
```
[18,] 6.299293
[19,] 6.088929
[20,] 6.251371
[21,] 4.836605
[22,] 5.744576
[23,] 3.803390
[24,] 4.007609
[25,] 5.866912
[26,] 4.856876
[27,] 5.562668
[28,] 5.904101
[29,] 8.043563
[30,] 7.129841
[31,] 6.864001
[32,] 6.034341
[33,] 4.404443
[34,] 5.220976
[35,] 4.904922
[36,] 4.691485
[37,] 6.764963
[38,] 4.727765
[39,] 5.365503
[40,] 4.759550
[41,] 4.908952
[42,] 7.648438
[43,] 2.621186
[44,] 3.522059
[45,] 5.420799
[46,] 5.071638
[47,] 4.520886
[48,] 6.362891
[49,] 5.229931
[50,] 5.152074
[51,] 7.833764
[52,] 4.630455
[53,] 6.713343
[54,] 6.492714
[55,] 6.292947
[56,] 4.172657
[57,] 4.461878
[58,] 1.893837
[59,] 4.132791
[60,] 4.680746
[61,] 5.549052
[62,] 5.748532
[63,] 3.059525
[64,] 7.188573
[65,] 5.957170
[66,] 4.252710
[67,] 8.289710
[68,] 6.197514
```

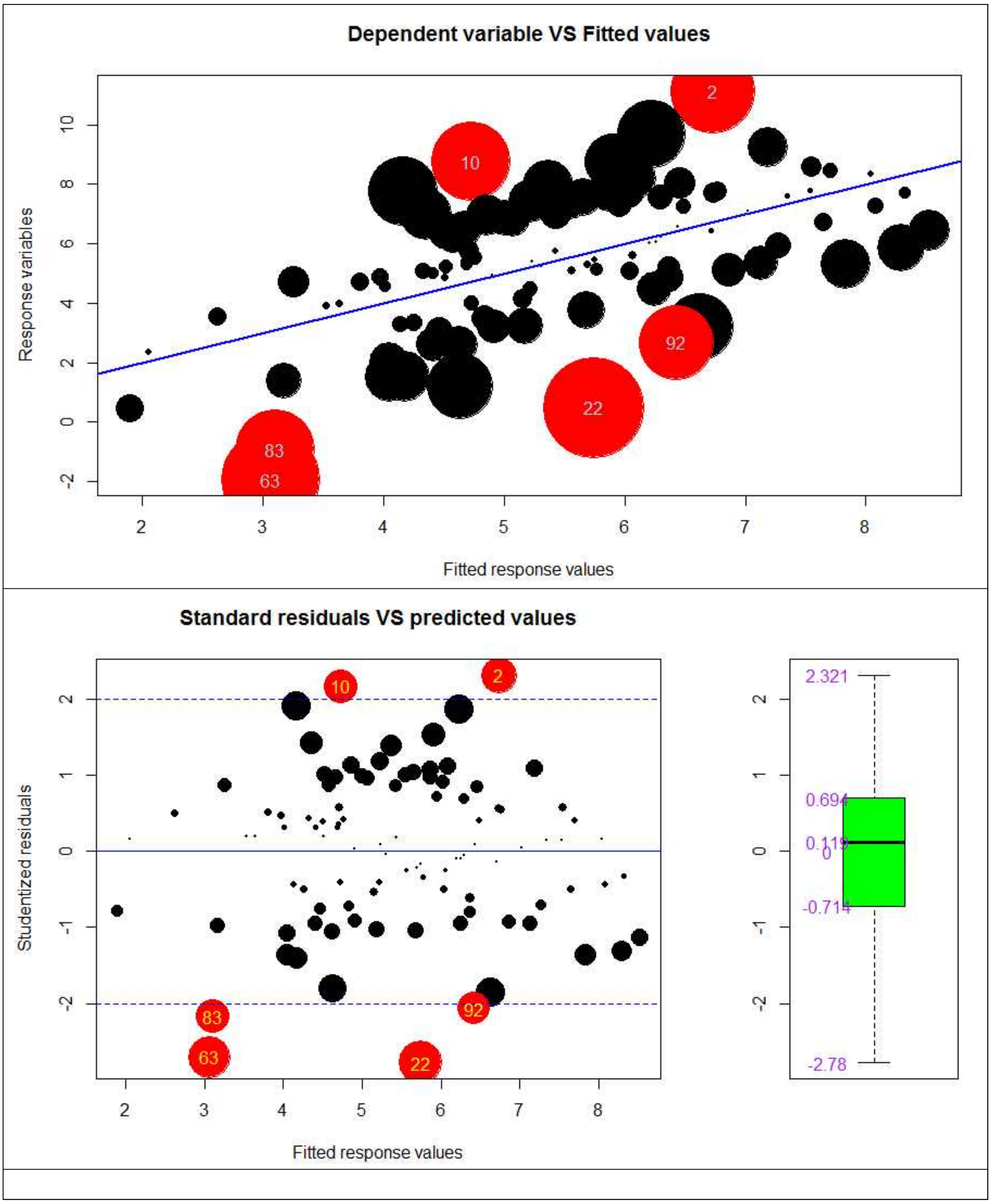
```
 [69,] 3.247374
 [70,] 5.647836
 [71,] 5.867363
 [72,] 6.249501
 [73,] 6.729468
 [74,] 6.221184
 [75,] 5.679407
 [76,] 2.054514
 [77,] 6.624157
 [78,] 4.328715
 [79,] 6.363770
 [80,] 4.618558
 [81,] 5.304138
 [82,] 4.669629
 [83,] 3.099912
 [84,] 7.275835
 [85,] 4.411167
 [86,] 8.526665
 [87,] 7.703244
 [88,] 4.502370
 [89,] 7.538839
 [90,] 4.510277
 [91,] 5.684539
 [92,] 6.424207
 [93,] 4.348007
 [94,] 6.061734
 [95,] 8.079016
 [96,] 3.973535
 [97,] 5.428321
 [98,] 7.548656
 [99,] 5.214942
[100,] 3.631885

$Residuals
              [,1]
  [1,]  0.27307748
  [2,]  4.40988390
  [3,]  1.07631221
  [4,]  1.63487922
  [5,]  3.60266744
  [6,]  1.75189198
  [7,] -2.03757453
  [8,]  1.89292234
  [9,] -0.60461022
 [10,]  4.04941549
 [11,] -0.65187166
 [12,] -1.77921650
 [13,]  1.58869294
 [14,]  0.16097255
 [15,]  0.07865770
 [16,] -2.55332600
```

```
[17,] -1.92985078
[18,] -0.08653188
[19,]  2.10518761
[20,] -0.17951677
[21,] -1.36985822
[22,] -5.25778252
[23,]  0.93516377
[24,]  0.58941938
[25,]  1.99698954
[26,]  2.10763788
[27,] -0.47785447
[28,]  2.84988813
[29,]  0.30660316
[30,] -1.78985430
[31,] -1.72615274
[32,] -0.93818463
[33,]  0.60522834
[34,] -0.76766374
[35,]  0.05247420
[36,]  0.69787267
[37,]  1.00789823
[38,] -0.74745238
[39,]  2.63013818
[40,]  0.77736946
[41,] -1.67743265
[42,] -0.92160296
[43,]  0.93605032
[44,]  0.38356091
[45,]  0.34492631
[46,]  1.74594924
[47,]  1.89440971
[48,] -1.50119816
[49,]  0.17803138
[50,] -1.02809195
[51,] -2.52097847
[52,] -3.40593097
[53,] -0.25944882
[54,]  0.76688853
[55,]  1.30074046
[56,] -2.61435704
[57,] -1.38304001
[58,] -1.43568476
[59,] -0.82281543
[60,]  0.61646826
[61,]  1.88866260
[62,] -0.29086317
[63,] -4.99721750
[64,]  2.05802371
[65,]  1.36737425
[66,] -0.91175167
[67,] -2.41906393
```

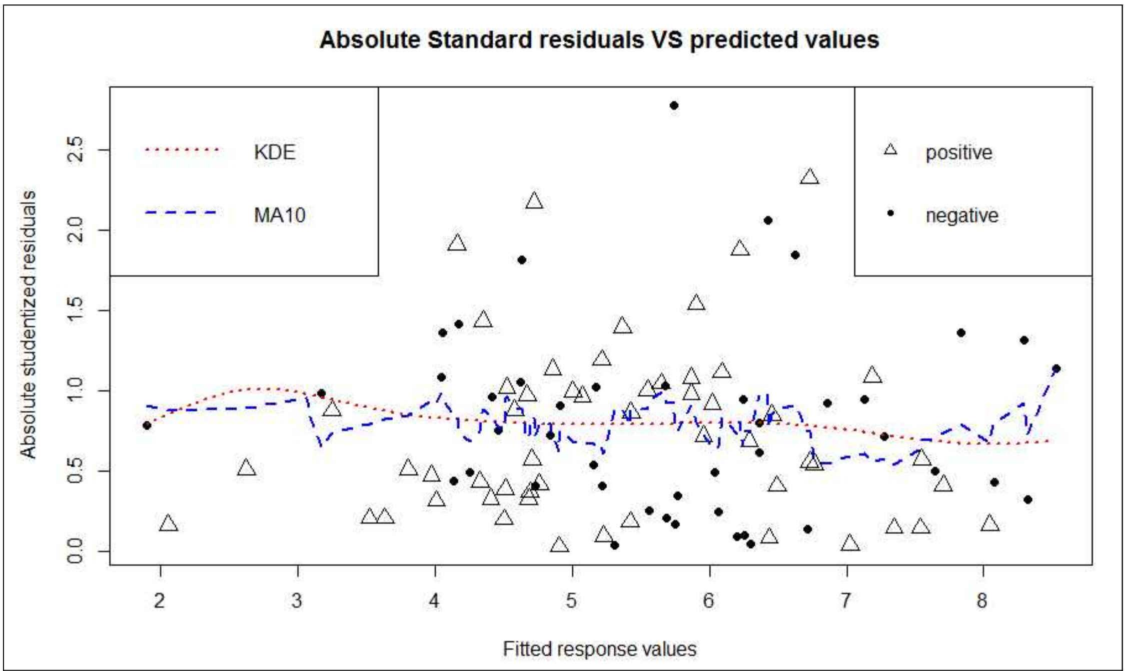
```
 [68,] -0.16523549
 [69,]  1.47392896
 [70,]  1.93049729
 [71,]  1.81415371
 [72,] -1.79178830
 [73,]  1.01078597
 [74,]  3.48498284
 [75,] -1.88710167
 [76,]  0.30683238
 [77,] -3.44548652
 [78,]  0.77551044
 [79,] -1.16186342
 [80,] -2.00290589
 [81,] -0.07002558
 [82,]  1.83019893
 [83,] -4.01525198
 [84,] -1.34607257
 [85,] -1.80017661
 [86,] -2.07650540
 [87,]  0.76510680
 [88,]  0.37537499
 [89,]  0.27118489
 [90,]  0.72955358
 [91,] -0.39185245
 [92,] -3.73279184
 [93,]  2.65264428
 [94,] -0.46629188
 [95,] -0.78083813
 [96,]  0.88489860
 [97,]  1.59319079
 [98,]  1.06531237
 [99,]  2.22644678
[100,]  0.36803345

$Standard.Residuals
  [1]  0.14621932  2.32114970  0.57153929  0.87974071  1.90940601  0.92011018 -1.08413684
  [8]  0.99306711 -0.32351339  2.17266615 -0.34288737 -0.98076112  0.84919099  0.08485406
 [15]  0.04173664 -1.36122109 -1.02336295 -0.04573283  1.11549389 -0.09804247 -0.71981894
 [22] -2.77953994  0.51186661  0.31366989  1.07755297  1.13136116 -0.25226494  1.53626742
 [29]  0.16453837 -0.94461232 -0.92611157 -0.49376898  0.32502459 -0.40735099  0.02834328
 [36]  0.36761431  0.54242613 -0.40399631  1.39370414  0.41608024 -0.91057152 -0.49983195
 [43]  0.50878722  0.20754560  0.18421732  0.96496966  1.01878534 -0.79633830  0.09331282
 [50] -0.54082447 -1.36291393 -1.81089345 -0.13990659  0.40781113  0.68549359 -1.41232270
 [57] -0.75325283 -0.78080575 -0.44049989  0.32484902  1.00074739 -0.16549547 -2.69924865
 [64]  1.08900085  0.72007226 -0.49087686 -1.31213965 -0.08811307  0.87626859  1.04331695
 [71]  0.97856093 -0.94605065  0.55642828  1.87485168 -1.03107986  0.16624905 -1.84199828
 [78]  0.43186179 -0.61288542 -1.05455478 -0.04005917  0.97389414 -2.16611993 -0.71160574
 [85] -0.95776832 -1.13700689  0.40890568  0.19899248  0.14514993  0.38777677 -0.20652954
 [92] -2.06344447  1.43414533 -0.24914333 -0.43376220  0.47286605  0.86279698  0.57223897
 [99]  1.19066657  0.20592839
```

```
$SSR
[1] 195.2996

$SSE
[1] 347.1072

$F.Statistic
[1] 8.721064

$P.value
[1] 1.556127e-07



        Shapiro-Wilk normality test

data:  stdresid
W = 0.9875, p-value = 0.4694
```

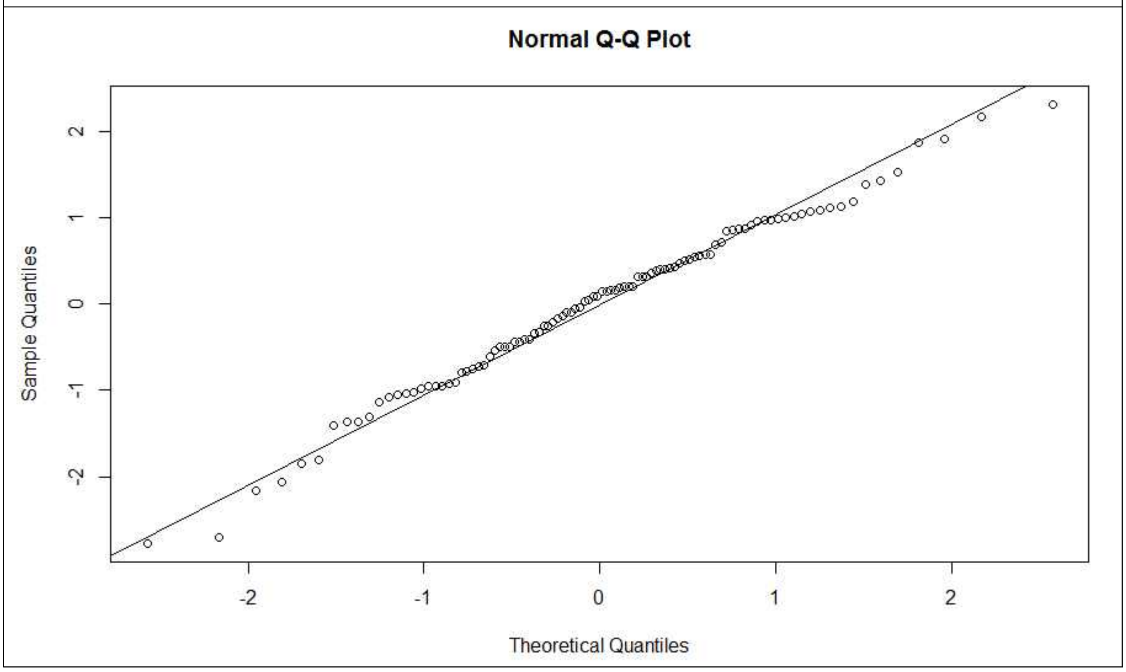




Dependent variable VS Fitted values
Response variables
Fitted response values
2
10
92
22
83
63
Standard residuals VS predicted values
Studentized residuals
Fitted response values
2.321
0.694
0.119
0
-0.714
-2.78



Absolute Standard residuals VS predicted values
KDE
MA10
positive
negative
Absolute studentized residuals
Fitted response values



Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles

Case 2

input

```
N <- 100
set.seed(1234)
tX1 <- rnorm(N,0,1)
tX2 <- rnorm(N,3,1)
tX3 <- rgamma(N,1,3)
tX4 <- sample(c(0,1),N,replace=T)
tX5 <- sample(1:3,N,replace=T,prob=c(1,2,3)/6)
tX6 <- rbinom(N,4,0.3)
tX <- cbind(rep(1,length(tX1)),tX1,tX2,tX3,tX4,tX5,tX6)
tbeta <- (0:6)/5
ty <- as.vector(tX%*%tbeta)+rnorm(N,0,2)
dat <- data.frame(y=ty, age=tX1, height=tX2, weight=tX3, smoking=tX4,
                  therapy=tX5, surgery=tX6)


MyReg <- function(data) {
  n <- nrow(data) ; p <- ncol(data)-1
  y <- as.matrix(data)[,1] ; X <- cbind(rep(1,n),as.matrix(data)[,-1])
  beta <- (solve(t(X)%*%X))%*%(t(X))%*%(y)
  pred <- X%*%beta
  residuals <- y-pred ; sig2 <- (t(residuals)%*%residuals)/(n-p-1)
  H <- (X)%*%(solve(t(X)%*%X))%*%(t(X))
  stdresid <- rep(0,n)
  for(i in 1:n) {
    stdresid[i] <- residuals[i]/sqrt(sig2*(1-H[i,i]))
  }
  SST <- sum((y-mean(y))^2) ; SSE <- sum((y-pred)^2) ; SSR <- SST - SSE
  MSR <- SSR/(p) ; MSE <- SSE/(n-p-1) ; F <- MSR/MSE
  P.value <- 1-pf(F,p,n-p-1) ; Rsq <- SSR/SST
  C <- solve(t(X)%*%X)
  std.error <- rep(0,p+1)
  for(i in 1:(p+1)) {
    std.error[i] <- sqrt(sig2*C[i,i])
  }
  t.value <- rep(0,p+1)
  for(i in 1:(p+1)) {
    t.value[i] <- beta[i]/std.error[i]
  }
  Pr.t <- rep(0,(p+1))
  for(i in 1:(p+1)) {
    Pr.t[i] <- 2*(1-pt(abs(t.value[i]), n-p-1))
  }
  args <- list(Data=data, Beta=beta, Sigma2=sig2, Predict=pred,
               Residuals=residuals, Standard.Residuals=stdresid, SSR=SSR,
               SSE=SSE, F.Statistic=F, P.value=P.value)

  aster <- function(p) {
```

```
  if (p >= 0.05 & p < 0.1) "."
  else if (p >= 0.01 & p < 0.05) "*"
  else if (p >= 0.001 & p < 0.1) "**"
  else if (p < 0.001) "***"
  else " "
}

cat("\n == ANALYSIS OF VARIANCE == \n\n",
    encodeString(c("Source", "    df", "    SS", "    MS", "    F", "    P-value"),
                 width=8, justify="right"), "\n",
    "-----------------------------------------------------------------", "\n",
    encodeString(c("Regression", p, round(SSR,2), round(MSR,2), round(F,2),
                   round(P.value,2)), width=8, justify="right"),
    aster(P.value), "\n", encodeString(c("      Error", n-p-1, round(SSE,2),
                                        round(MSE,2)), width=8, justify="right"),
    "\n", encodeString(c("      Total", n-1, round(SST,2)), width=8,
                       justify="right"),
    "\n",
    "-----------------------------------------------------------------", "\n",
    "Estimated error variance :", round(sig2,4), "\n","R-squares :",round(Rsq,4),
    "\n\n")

ind.name <- colnames(data)
ind.name[1] <- "(Intercept)"
test.mat <- cbind(ind.name, round(beta,4), round(std.error,4), round(t.value,4),
                  round(Pr.t,4))
cat("\n\n\n== PARAMETER ESTIMATES ==\n\n",
    encodeString(c(" ", "Estimate", "Std.Error", "t value", "Pr(>|t|)"), width=11,
                 justify="right"),
    "\n")
for(i in 1:nrow(test.mat)) {
  cat(encodeString(test.mat[i,], width=11, justify="right"), aster(Pr.t[i]),"\n")
}

cat("\n", "===Various Statistics in Multiple Regression===", "\n\n")
print(args)

outlier1 <- rep(0,n)
outlier2 <- rep(0,n)
std.sign <- rep(0,n)
for(i in 1:n) {
  outlier1[i] <- ifelse(abs(stdresid[i]) > 2, "out", "in")
  outlier2[i] <- ifelse(abs(stdresid[i]) > 2, 2, 1)
  std.sign[i] <- ifelse(stdresid[i] > 0, 24, 20)
}
abs.stdresid <- abs(stdresid)
ex1 <- as.data.frame(cbind(number=(1:n), stdresid=stdresid,
                           pred=pred, y=y,
                           outlier1=outlier1, outlier2=outlier2,
                           std.sign=std.sign, abs.stdresid=abs.stdresid))
ex1_1 <- as.matrix(ex1[,1:4,8])
```

```
ex1_1 <- as.numeric(ex1_1)
ex1_1 <- matrix(ex1_1,nrow=n,ncol=5)
colnames(ex1_1) <- c("number","stdresid","pred","y","abs.stdresid")
ex2 <- as.data.frame(ex1_1)
ex2 <- ex2[abs.stdresid > 2,]

pred1 <- as.vector(pred) ; abs.stdresid1 <- as.vector(abs.stdresid)
ord <- order(pred1,abs.stdresid1)
re1 <- as.data.frame(t(rbind(pred1,abs.stdresid1)[,ord]))

KDE <- function(x) {
  w <- rep(0,n)
  for (i in 1:n) {
    w[i] <- exp(-(pred[i]-x)^2)/sqrt(2*pi)
  }
  ans <- sum(w*abs(stdresid))/sum(w)
  return(ans)
}
d1 <- seq(min(re1$pred1),max(re1$pred1),0.01)
d2 <- rep(0,length(d1))
for (j in 1:length(d1)) {
  d2[j] <- KDE(d1[j])
}
d3 <- re1$pred1
d4 <- rep(0,length(d3))
for (j in 1:length(d3)) {
  d4[j] <- mean(re1$abs.stdresid1[j:(j+9)],na.rm=TRUE)
}

pairs(data, main="Scatterplot matrix from given data", cex=outlier2,
      pch=21, bg=c("yellow","red")[unclass(ex1$outlier1)])
dev.new()
plot(y~pred, main="Dependent variable VS Fitted values",
     xlab="Fitted response values",
     ylab="Response variables", cex=abs(stdresid)*7, pch=20,
     col=factor(outlier2, labels=c("black", "red")))
abline(0,1,col="blue",lwd=2)
for (i in 1:nrow(ex2)) {
  text(ex2$pred[i], ex2$y[i], ex2$number[i], col="light blue")
}
dev.new()
par(fig=c(0,0.7,0,1), new=FALSE)
plot(pred~stdresid, main="Standard residuals VS predicted values",
     xlab="Studentized residuals", ylab="Fitted response values",
     pch=21, cex=1, col="black")
abline(v=c(-2,0,2),col="saddlebrown",lty=c(2,1,2))
rug(stdresid,side=1,col="lightblue4")
for (i in 1:nrow(ex2)) {
  text(ex2$stdresid[i], ex2$pred[i], ex2$number[i], col="darkgreen", pos=2)
  points(ex2$stdresid[i], ex2$pred[i], pch=19, cex=2, col="tomato")
}
```

```
  par(fig=c(0.7,1,0,1),new=TRUE)
  boxplot(stdresid,col="green",main="Boxplot of Standard residuals")
  for (i in 1:6) {
    text(0.7,summary(stdresid)[i],round(summary(stdresid)[i],3),col="hotpink4")
  }
  dev.new()
  par(fig=c(0,1,0,1),new=FALSE)
  plot(abs.stdresid~pred,
       main="Absolute Standard residuals VS predicted values",
       xlab="Fitted response values",
       ylab="Absolute studentized residuals", cex=1.5, pch=std.sign)
  legend("topright",c("positive","negative"),pch=c(24,20))
  lines(d1,d2,lty=3,lwd=2,col="red")
  lines(d3,d4,lty=2,lwd=2,col="blue")
  legend("topleft",c("KDE","MA10"),lty=c(3,2),lwd=2,col=c("red","blue"))
  dev.new()
  qqnorm(stdresid) ; qqline(stdresid)
  dev.new()
  u <- seq(min(stdresid),max(stdresid),0.001)
  hist(stdresid,probability=T,col="bisque",main="Normality of standard residuals")
  lines(u,dnorm(u),lwd=2,lty=1,col="darkkhaki")
  test.res <- shapiro.test(stdresid)
  print(test.res)

  return(args)
}

res1 <- MyReg(dat)

str(res1)

attributes(res1)
```

output

```
== ANALYSIS OF VARIANCE ==

   Source        df        SS        MS         F     P-value
 -----------------------------------------------------------------
 Regression       6     195.3     32.55      8.72         0 ***
      Error      93    347.11      3.73
      Total      99    542.41
 -----------------------------------------------------------------
 Estimated error variance : 3.7323
 R-squares : 0.3601




== PARAMETER ESTIMATES ==
```

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	-0.7866	1.182	-0.6655	0.5074	
age	0.0198	0.1951	0.1012	0.9196	
height	0.5986	0.1953	3.0653	0.0028	**
weight	-0.0803	0.6073	-0.1322	0.8951	
smoking	0.2927	0.4105	0.713	0.4776	
therapy	1.2801	0.2794	4.5809	0	***
surgery	1.0944	0.2018	5.4219	0	***

===Various Statistics in Multiple Regression===

$Data

	y	age	height	weight	smoking	therapy	surgery
1	7.6185092	-1.207065749	3.4145235	0.300656147	1	2	3
2	11.1395846	0.277429242	2.5252815	0.371353870	0	3	2
3	5.7838752	1.084441177	3.0659935	0.213260667	0	2	1
4	6.2143115	-2.345697703	2.4975222	0.372750095	1	2	1
5	7.7605163	0.429124689	2.1740014	0.248226012	0	2	1
6	7.7740276	0.506055892	3.1669893	0.394101281	0	3	1
7	2.0073677	-0.574739960	2.1037354	0.883828859	0	2	1
8	6.8965796	-0.546631856	3.1681854	0.531965694	1	2	1
9	7.7153296	-0.564451999	3.3549683	0.174454546	0	3	3
10	8.7708591	-0.890037829	2.9478949	0.007488234	1	1	2
11	5.1149679	-0.477192700	2.8040654	0.626116131	0	3	1
12	1.3942546	-0.998386445	2.3509302	1.179623070	1	1	1
13	8.0460598	-0.776253895	1.8902328	0.102581133	1	2	3
14	6.5964155	0.064458817	3.8492742	0.224677931	0	3	1
15	7.1039366	0.959494059	3.0223625	0.563909112	0	3	2
16	1.4979346	-0.110285494	3.8311406	0.167726840	0	2	0
17	3.2428745	-0.511009506	1.7557121	0.201034863	0	3	1
18	6.2127608	-0.911195417	3.1690264	0.254401211	1	3	1
19	8.1941169	-0.837171680	3.6731663	0.693440641	0	2	2
20	6.0718540	2.415835178	2.9737236	0.213521547	1	3	1
21	3.4667472	0.134088220	2.8086078	0.098975684	1	2	1
22	0.4867931	-0.490685897	2.2180934	0.176831684	1	3	1
23	4.7385541	-0.440547872	5.0581620	0.024831943	1	1	0
24	4.5970284	0.459589441	3.7505015	0.250382061	0	2	0
25	7.8639016	-0.693720247	4.8242083	0.757634302	0	3	0
26	6.9645135	-1.448204910	3.0800596	0.147808868	0	3	0
27	5.0848131	0.574755721	2.3685907	0.180896373	0	3	1
28	8.7539894	-1.023655723	1.4867119	0.278347274	0	2	3
29	8.3501663	-0.015138300	2.3639002	0.008344596	1	3	3
30	5.3399869	-0.935948601	3.2263015	0.315793550	0	3	2
31	5.1378487	1.102297546	4.0136903	0.013843684	1	3	1
32	5.0961565	-0.475593079	3.2527501	0.640095477	0	3	1
33	5.0096718	-0.709440038	1.8280517	0.275633695	1	3	0
34	4.4533125	-0.501258061	3.6687143	0.234801962	0	3	0
35	4.9573960	-1.629093469	1.3498991	0.235832515	0	3	1
36	5.3893573	-1.167619262	2.6341478	0.285561219	1	2	1
37	7.7728607	-2.180039649	2.6838817	0.510195492	0	3	2
38	3.9803126	-1.340993192	1.0517540	0.290288645	0	3	1

39	7.9956412	-0.294293859	3.9200575	0.359636780	0	3	0
40	5.5369194	-0.465897540	2.3771284	0.006432923	1	3	0
41	3.2315194	1.449496265	2.6659634	0.770510833	1	3	0
42	6.7268347	-1.068642724	4.3951479	0.226339440	0	2	3
43	3.5572367	-0.855364634	3.6366744	0.403122869	0	1	0
44	3.9056203	-0.280623002	2.8915683	1.045380347	1	1	1
45	5.7657251	-0.994340076	3.5137628	0.115163135	1	3	0
46	6.8175870	-0.968514318	3.3992718	1.304649095	1	2	1
47	6.4152961	-1.107318193	4.6628564	0.274436277	0	2	0
48	4.8616932	-1.251985886	3.2758934	0.175208053	1	3	1
49	5.4079627	-0.523828119	3.5062726	0.239968258	1	2	1
50	4.1239824	-0.496849957	3.3475520	0.032860857	1	2	1
51	5.3127850	-1.806031257	2.6227624	0.465253488	0	3	3
52	1.2245240	-0.582075925	3.0976195	0.999719975	0	2	1
53	6.4538941	-1.108889624	4.6387446	0.048526795	0	2	2
54	7.2596023	-1.014962009	2.1244075	0.016168668	0	3	2
55	7.5936876	-0.162309524	3.1217600	0.165215369	1	3	1
56	1.5583003	0.563055819	4.3621307	0.468037451	0	1	1
57	3.0788377	1.647817473	2.7653789	1.170797202	0	2	1
58	0.4581519	-0.773353424	1.9466172	0.527473505	1	1	0
59	3.3099758	1.605909629	2.1302164	0.523417265	0	2	1
60	5.2972146	-1.157808548	2.6098730	0.240719133	1	2	1
61	7.4377150	0.656588464	2.1526499	0.093950833	1	2	2
62	5.4576686	2.548991071	2.7393606	0.136985760	1	1	3
63	-1.9376925	-0.034760390	2.5855803	0.939535162	0	1	1
64	9.2465962	-0.669633580	2.8169492	0.243493845	1	3	2
65	7.3245442	-0.007604756	3.4070561	0.554455368	0	2	2
66	3.3409581	1.777084448	3.6246331	0.229048919	1	2	0
67	5.8706458	-1.138607737	4.6782057	0.290679589	1	3	2
68	6.0322781	1.367827179	2.9313063	0.310287531	1	3	1
69	4.7213034	1.329564791	2.6791601	1.945855520	0	2	0
70	7.5783331	0.336472797	4.4710057	0.127057828	1	1	2
71	7.6815165	0.006892838	4.7043294	0.030359993	0	3	0
72	4.4577125	-0.455468738	3.0432440	0.048792384	1	3	1
73	7.7402535	-0.366523933	2.6673427	1.275292553	0	3	2
74	9.7061666	0.648286568	1.1777646	0.395085718	1	3	2
75	3.7923051	2.070270861	4.4112624	0.694050412	0	3	0
76	2.3613461	-0.153398412	2.1624176	0.287642700	1	1	0
77	3.1786706	-1.390700947	1.8762372	0.082086647	1	3	2
78	5.1042255	-0.723581777	6.0437659	0.760985052	1	1	0
79	5.2019065	0.258261762	3.2350213	0.231048529	1	3	1
80	2.6156518	-0.317059115	2.9667414	0.237070641	0	2	1
81	5.2341128	-0.177789958	0.2677805	0.204189201	1	1	4
82	6.4998278	-0.169994077	2.9002094	0.474560230	1	1	2
83	-0.9153396	-1.372301886	3.9760317	0.489410169	1	1	0
84	5.9297626	-0.173787170	3.4138689	0.083323763	0	3	2
85	2.6109900	0.850232257	3.9123222	0.172416743	1	2	0
86	6.4501599	0.697608712	4.9837322	0.068972247	1	3	2
87	8.4683504	0.549997351	4.1691085	0.569063747	0	3	2
88	4.8777452	-0.402731975	2.4912630	0.430463151	0	3	0
89	7.8100242	-0.191593770	3.7041802	0.301186011	1	2	3

```
90    5.2398306 -1.194527880 2.8015837 0.138446892        0        2        1
91    5.2926870 -0.053158819 2.4619292 0.537928137        1        2        2
92    2.6914147  0.255196001 0.1442413 0.048790497        0        3        3
93    7.0006510  1.705964007 2.2103531 0.777457371        0        3        0
94    5.5954422  1.001513252 3.4878146 0.102332856        0        2        2
95    7.2981776 -0.495583443 5.1680325 0.767098073        0        2        3
96    4.8584337  0.355550297 3.5006946 0.120016478        1        1        1
97    7.0215120 -1.134608044 3.6202102 0.780808501        1        3        0
98    8.6139680  0.878203627 2.0340968 0.287524310        0        3        3
99    7.4413886  0.972916753 3.1626547 0.545354279        1        3        0
100   3.9999188  2.121117105 0.9217625 0.191202086        0        3        0

$Beta
                [,1]
         -0.78660499
age       0.01975009
height    0.59862082
weight   -0.08026628
smoking   0.29271521
therapy   1.28005300
surgery   1.09439426

$Sigma2
         [,1]
[1,] 3.732336

$Predict
           [,1]
  [1,] 7.345432
  [2,] 6.729701
  [3,] 4.707563
  [4,] 4.579432
  [5,] 4.157849
  [6,] 6.022136
  [7,] 4.044942
  [8,] 5.003657
  [9,] 8.319940
 [10,] 4.721444
 [11,] 5.766840
 [12,] 3.173471
 [13,] 6.457367
 [14,] 6.435443
 [15,] 7.025279
 [16,] 4.051261
 [17,] 5.172725
 [18,] 6.299293
 [19,] 6.088929
 [20,] 6.251371
 [21,] 4.836605
 [22,] 5.744576
 [23,] 3.803390
```

```
[24,] 4.007609
[25,] 5.866912
[26,] 4.856876
[27,] 5.562668
[28,] 5.904101
[29,] 8.043563
[30,] 7.129841
[31,] 6.864001
[32,] 6.034341
[33,] 4.404443
[34,] 5.220976
[35,] 4.904922
[36,] 4.691485
[37,] 6.764963
[38,] 4.727765
[39,] 5.365503
[40,] 4.759550
[41,] 4.908952
[42,] 7.648438
[43,] 2.621186
[44,] 3.522059
[45,] 5.420799
[46,] 5.071638
[47,] 4.520886
[48,] 6.362891
[49,] 5.229931
[50,] 5.152074
[51,] 7.833764
[52,] 4.630455
[53,] 6.713343
[54,] 6.492714
[55,] 6.292947
[56,] 4.172657
[57,] 4.461878
[58,] 1.893837
[59,] 4.132791
[60,] 4.680746
[61,] 5.549052
[62,] 5.748532
[63,] 3.059525
[64,] 7.188573
[65,] 5.957170
[66,] 4.252710
[67,] 8.289710
[68,] 6.197514
[69,] 3.247374
[70,] 5.647836
[71,] 5.867363
[72,] 6.249501
[73,] 6.729468
[74,] 6.221184
```

```
 [75,] 5.679407
 [76,] 2.054514
 [77,] 6.624157
 [78,] 4.328715
 [79,] 6.363770
 [80,] 4.618558
 [81,] 5.304138
 [82,] 4.669629
 [83,] 3.099912
 [84,] 7.275835
 [85,] 4.411167
 [86,] 8.526665
 [87,] 7.703244
 [88,] 4.502370
 [89,] 7.538839
 [90,] 4.510277
 [91,] 5.684539
 [92,] 6.424207
 [93,] 4.348007
 [94,] 6.061734
 [95,] 8.079016
 [96,] 3.973535
 [97,] 5.428321
 [98,] 7.548656
 [99,] 5.214942
[100,] 3.631885

$Residuals
              [,1]
  [1,]  0.27307748
  [2,]  4.40988390
  [3,]  1.07631221
  [4,]  1.63487922
  [5,]  3.60266744
  [6,]  1.75189198
  [7,] -2.03757453
  [8,]  1.89292234
  [9,] -0.60461022
 [10,]  4.04941549
 [11,] -0.65187166
 [12,] -1.77921650
 [13,]  1.58869294
 [14,]  0.16097255
 [15,]  0.07865770
 [16,] -2.55332600
 [17,] -1.92985078
 [18,] -0.08653188
 [19,]  2.10518761
 [20,] -0.17951677
 [21,] -1.36985822
 [22,] -5.25778252
```

```
[23,]  0.93516377
[24,]  0.58941938
[25,]  1.99698954
[26,]  2.10763788
[27,] -0.47785447
[28,]  2.84988813
[29,]  0.30660316
[30,] -1.78985430
[31,] -1.72615274
[32,] -0.93818463
[33,]  0.60522834
[34,] -0.76766374
[35,]  0.05247420
[36,]  0.69787267
[37,]  1.00789823
[38,] -0.74745238
[39,]  2.63013818
[40,]  0.77736946
[41,] -1.67743265
[42,] -0.92160296
[43,]  0.93605032
[44,]  0.38356091
[45,]  0.34492631
[46,]  1.74594924
[47,]  1.89440971
[48,] -1.50119816
[49,]  0.17803138
[50,] -1.02809195
[51,] -2.52097847
[52,] -3.40593097
[53,] -0.25944882
[54,]  0.76688853
[55,]  1.30074046
[56,] -2.61435704
[57,] -1.38304001
[58,] -1.43568476
[59,] -0.82281543
[60,]  0.61646826
[61,]  1.88866260
[62,] -0.29086317
[63,] -4.99721750
[64,]  2.05802371
[65,]  1.36737425
[66,] -0.91175167
[67,] -2.41906393
[68,] -0.16523549
[69,]  1.47392896
[70,]  1.93049729
[71,]  1.81415371
[72,] -1.79178830
[73,]  1.01078597
```

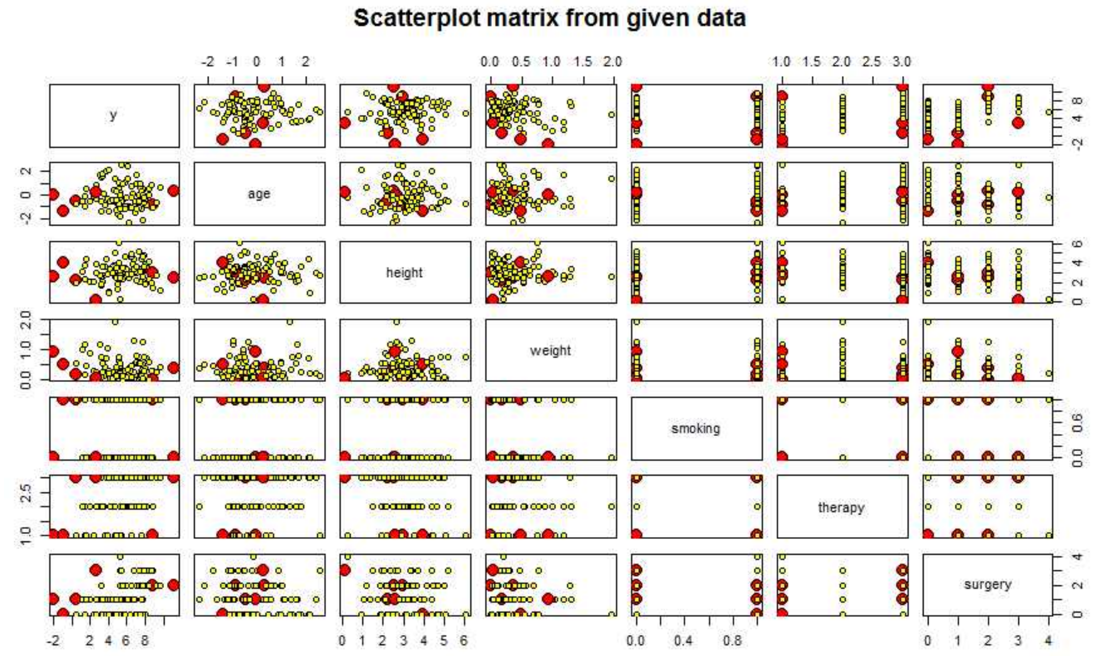
```
 [74,]   3.48498284
 [75,]  -1.88710167
 [76,]   0.30683238
 [77,]  -3.44548652
 [78,]   0.77551044
 [79,]  -1.16186342
 [80,]  -2.00290589
 [81,]  -0.07002558
 [82,]   1.83019893
 [83,]  -4.01525198
 [84,]  -1.34607257
 [85,]  -1.80017661
 [86,]  -2.07650540
 [87,]   0.76510680
 [88,]   0.37537499
 [89,]   0.27118489
 [90,]   0.72955358
 [91,]  -0.39185245
 [92,]  -3.73279184
 [93,]   2.65264428
 [94,]  -0.46629188
 [95,]  -0.78083813
 [96,]   0.88489860
 [97,]   1.59319079
 [98,]   1.06531237
 [99,]   2.22644678
[100,]   0.36803345

$Standard.Residuals
  [1]  0.14621932  2.32114970  0.57153929  0.87974071  1.90940601  0.92011018 -1.08413684
  [8]  0.99306711 -0.32351339  2.17266615 -0.34288737 -0.98076112  0.84919099  0.08485406
 [15]  0.04173664 -1.36122109 -1.02336295 -0.04573283  1.11549389 -0.09804247 -0.71981894
 [22] -2.77953994  0.51186661  0.31366989  1.07755297  1.13136116 -0.25226494  1.53626742
 [29]  0.16453837 -0.94461232 -0.92611157 -0.49376898  0.32502459 -0.40735099  0.02834328
 [36]  0.36761431  0.54242613 -0.40399631  1.39370414  0.41608024 -0.91057152 -0.49983195
 [43]  0.50878722  0.20754560  0.18421732  0.96496966  1.01878534 -0.79633830  0.09331282
 [50] -0.54082447 -1.36291393 -1.81089345 -0.13990659  0.40781113  0.68549359 -1.41232270
 [57] -0.75325283 -0.78080575 -0.44049989  0.32484902  1.00074739 -0.16549547 -2.69924865
 [64]  1.08900085  0.72007226 -0.49087686 -1.31213965 -0.08811307  0.87626859  1.04331695
 [71]  0.97856093 -0.94605065  0.55642828  1.87485168 -1.03107986  0.16624905 -1.84199828
 [78]  0.43186179 -0.61288542 -1.05455478 -0.04005917  0.97389414 -2.16611993 -0.71160574
 [85] -0.95776832 -1.13700689  0.40890568  0.19899248  0.14514993  0.38777677 -0.20652954
 [92] -2.06344447  1.43414533 -0.24914333 -0.43376220  0.47286605  0.86279698  0.57223897
 [99]  1.19066657  0.20592839

$SSR
[1] 195.2996

$SSE
[1] 347.1072
```

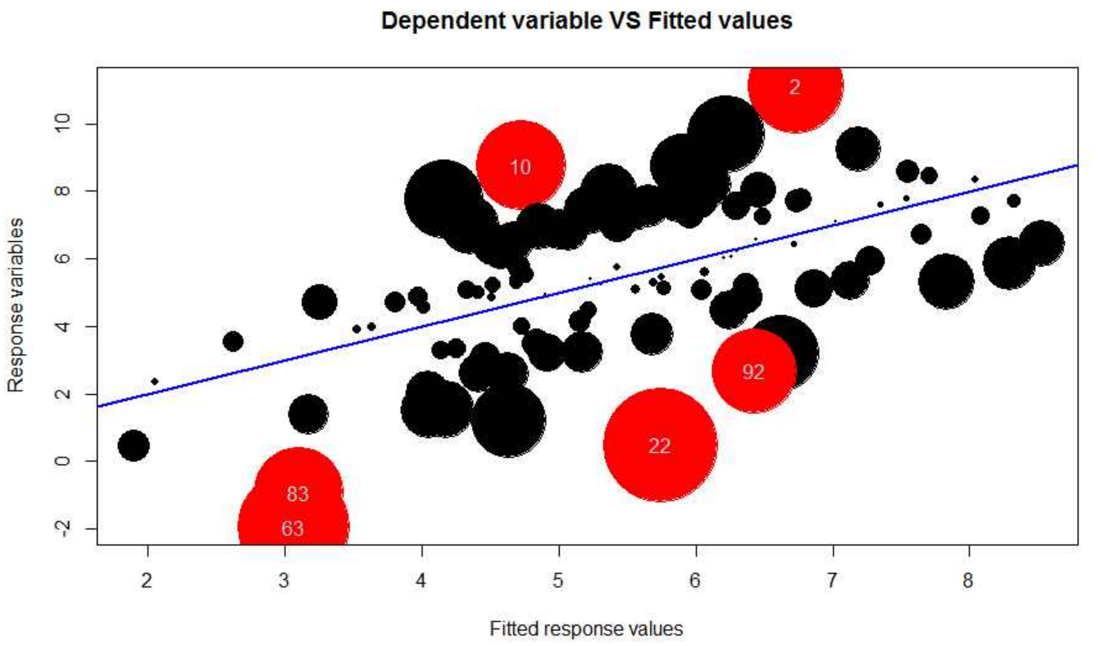
$F.Statistic
[1] 8.721064

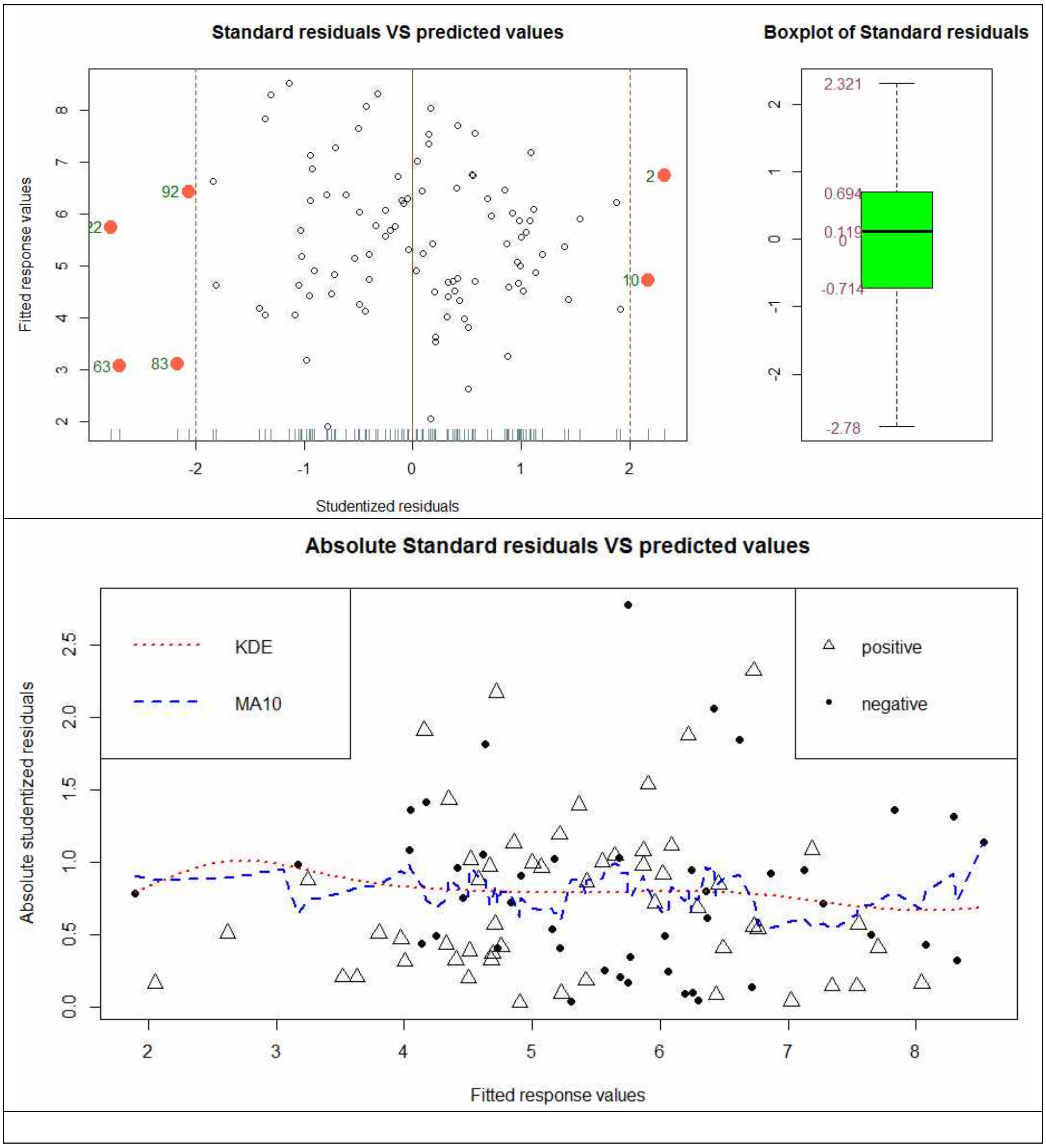
$P.value
[1] 1.556127e-07

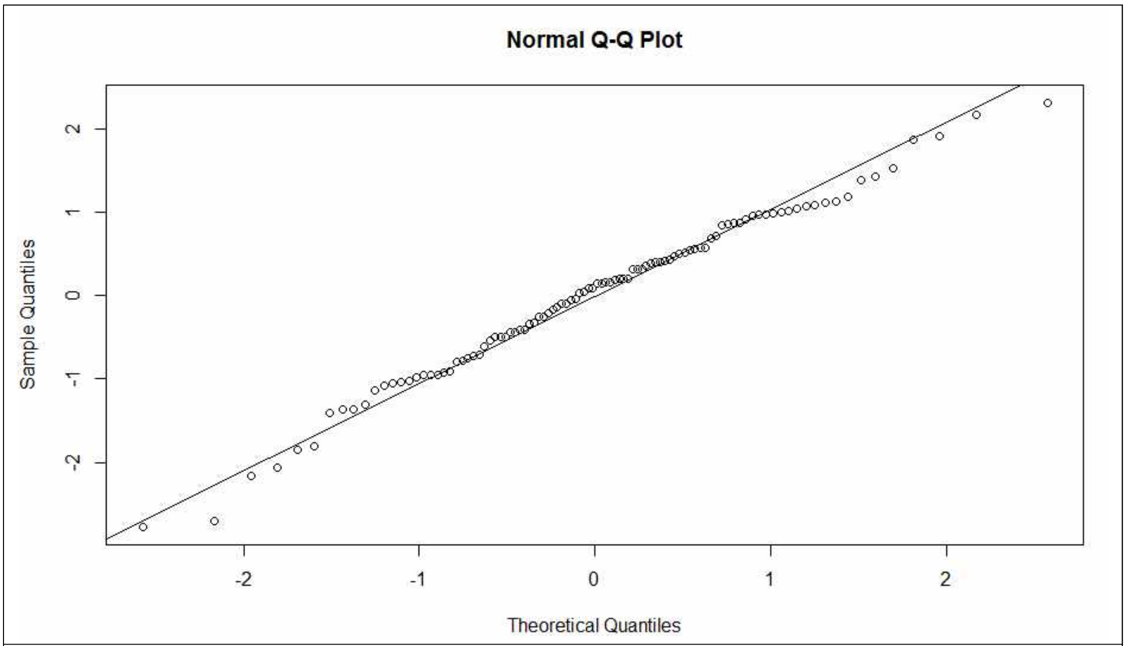
Shapiro-Wilk normality test

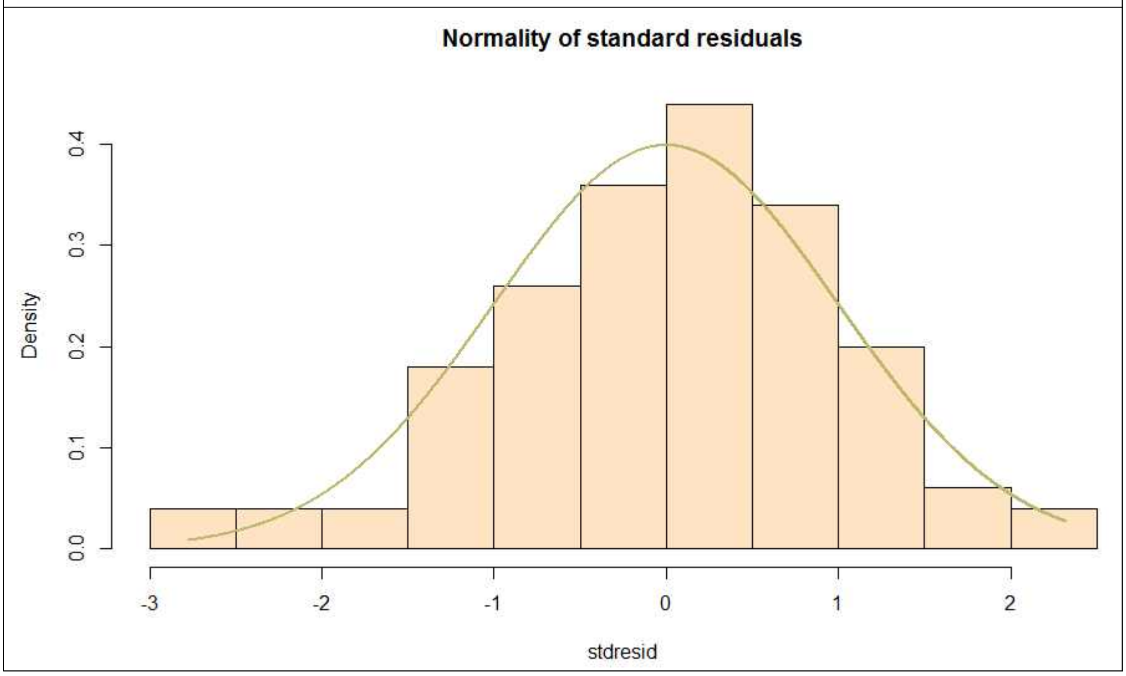
data: stdresid
W = 0.9875, p-value = 0.4694






Standard residuals VS predicted values
Fitted response values
Studentized residuals
92
22
2
10
63
83
Boxplot of Standard residuals
2.321
0.694
0.119
-0.714
-2.78
Absolute Standard residuals VS predicted values
KDE
MA10
positive
negative
Absolute studentized residuals
Fitted response values


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
-2
-1
0
1
2


Normality of standard residuals
Density
0.0
0.1
0.2
0.3
0.4
-3
-2
-1
0
1
2
stdresid