ch07output

```r
#### Example: Tool lifetime
tools <- read.table("http://statacumen.com/teach/ADA2/ADA2_notes_Ch07_tools.dat"
                  , header = TRUE)
str(tools)
```

```
## 'data.frame':    20 obs. of  3 variables:
##  $ lifetime: num  18.7 14.5 17.4 14.5 13.4 ...
##  $ rpm     : int  610 950 720 840 980 530 680 540 890 730 ...
##  $ type    : Factor w/ 2 levels "A","B": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
tools
```

```
##    lifetime  rpm type
## 1     18.73  610    A
## 2     14.52  950    A
## 3     17.43  720    A
## 4     14.54  840    A
## 5     13.44  980    A
## 6     24.39  530    A
## 7     13.34  680    A
## 8     22.71  540    A
## 9     12.68  890    A
## 10    19.32  730    A
## 11    30.16  670    B
## 12    27.09  770    B
## 13    25.40  880    B
## 14    26.05 1000    B
## 15    33.49  760    B
## 16    35.62  590    B
## 17    26.07  910    B
## 18    36.78  650    B
## 19    34.95  810    B
## 20    43.67  500    B
```

```r
library(xtable)
xtab.out <- xtable(tools[tools$type=="A",], digits=4)
print(xtab.out, floating=FALSE, math.style.negative=TRUE)
```

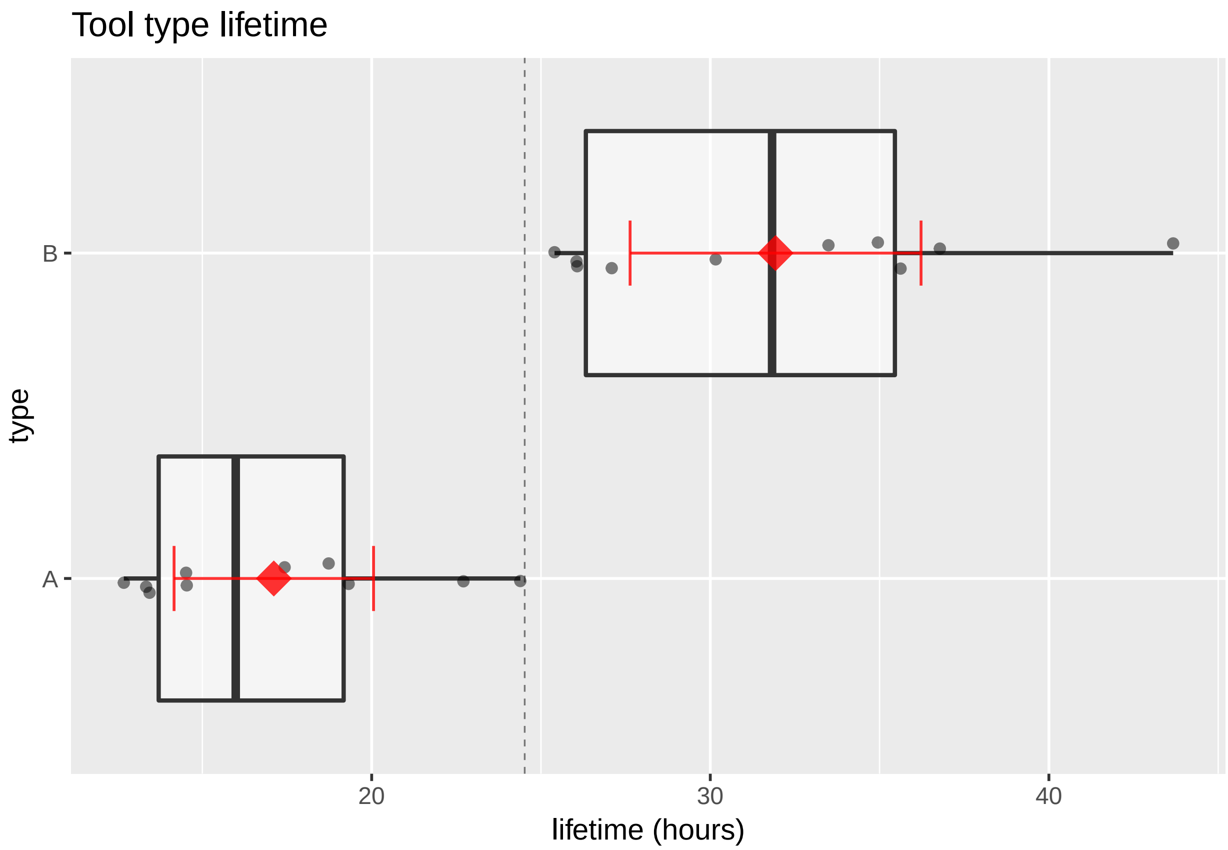
```
## % latex table generated in R 3.4.4 by xtable 1.8-2 package
## % Wed Mar  6 10:12:07 2019
## \begin{tabular}{rrrl}
##   \hline
##  & lifetime & rpm & type \\
##   \hline
## 1 & 18.7300 &   610 & A \\
##   2 & 14.5200 &   950 & A \\
##   3 & 17.4300 &   720 & A \\
##   4 & 14.5400 &   840 & A \\
##   5 & 13.4400 &   980 & A \\
##   6 & 24.3900 &   530 & A \\
##   7 & 13.3400 &   680 & A \\
##   8 & 22.7100 &   540 & A \\
```

```
##   9 & 12.6800 &   890 & A \\
##   10 & 19.3200 &   730 & A \\
##    \hline
## \end{tabular}
```

```
xtab.out <- xtable(tools[tools$type=="B",], digits=4)
print(xtab.out, floating=FALSE, math.style.negative=TRUE)
```

```
## % latex table generated in R 3.4.4 by xtable 1.8-2 package
## % Wed Mar  6 10:12:07 2019
## \begin{tabular}{rrrl}
##   \hline
##  & lifetime & rpm & type \\
##   \hline
## 11 & 30.1600 &   670 & B \\
##   12 & 27.0900 &   770 & B \\
##   13 & 25.4000 &   880 & B \\
##   14 & 26.0500 &  1000 & B \\
##   15 & 33.4900 &   760 & B \\
##   16 & 35.6200 &   590 & B \\
##   17 & 26.0700 &   910 & B \\
##   18 & 36.7800 &   650 & B \\
##   19 & 34.9500 &   810 & B \\
##   20 & 43.6700 &   500 & B \\
##    \hline
## \end{tabular}
```

```
library(ggplot2)
p <- ggplot(tools, aes(x = type, y = lifetime))
# plot a reference line for the global mean (assuming no groups)
p <- p + geom_hline(aes(yintercept = mean(lifetime)),
                    colour = "black", linetype = "dashed", size = 0.3, alpha = 0.5)
# boxplot, size=.75 to stand out behind CI
p <- p + geom_boxplot(size = 0.75, alpha = 0.5)
# points for observed data
p <- p + geom_point(position = position_jitter(w = 0.05, h = 0), alpha = 0.5)
# diamond at mean for each group
p <- p + stat_summary(fun.y = mean, geom = "point", shape = 18, size = 6,
                      colour="red", alpha = 0.8)
# confidence limits based on normal distribution
p <- p + stat_summary(fun.data = "mean_cl_normal", geom = "errorbar",
                      width = .2, colour="red", alpha = 0.8)
p <- p + labs(title = "Tool type lifetime") + ylab("lifetime (hours)")
p <- p + coord_flip()
print(p)
```




```
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot1.jpg")
```

```
## jpeg
##    3
```

```
dev.off()
```

```
## pdf
##   2
```

```
t.summary <- t.test(lifetime ~ type, data = tools)
t.summary
```

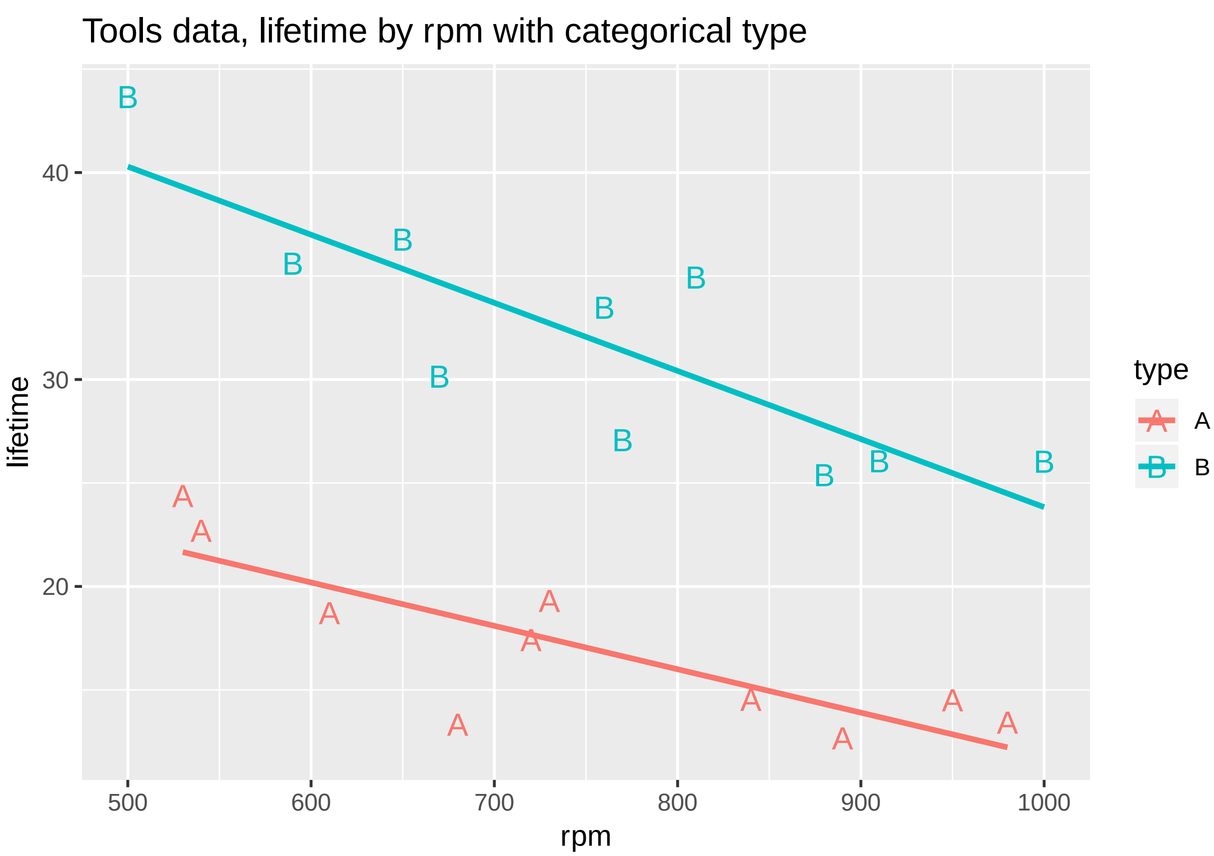
```
##
##  Welch Two Sample t-test
##
## data:  lifetime by type
## t = -6.435, df = 15.93, p-value = 8.422e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -19.70128  -9.93472
## sample estimates:
## mean in group A mean in group B
##          17.110          31.928
```

```
#### Example: Tool lifetime
library(ggplot2)
p <- ggplot(tools, aes(x = rpm, y = lifetime, colour = type, shape = type))
p <- p + geom_point(size=4)
```

```r
library(R.oo) # for ascii code lookup
```

```
## Loading required package: R.methodsS3
## R.methodsS3 v1.7.1 (2016-02-15) successfully loaded. See ?R.methodsS3 for help.
## R.oo v1.22.0 (2018-04-21) successfully loaded. See ?R.oo for help.
##
## Attaching package: 'R.oo'
## The following objects are masked from 'package:methods':
##
##     getClasses, getMethods
## The following objects are masked from 'package:base':
##
##     attach, detach, gc, load, save
```

```r
p <- p + scale_shape_manual(values=charToInt(sort(unique(tools$type))))
p <- p + geom_smooth(method = lm, se = FALSE)
p <- p + labs(title="Tools data, lifetime by rpm with categorical type")
print(p)
```




```r
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot2.jpg")
```

```
## jpeg
##    3
```

```r
dev.off()
```

```
## pdf
##   2
```

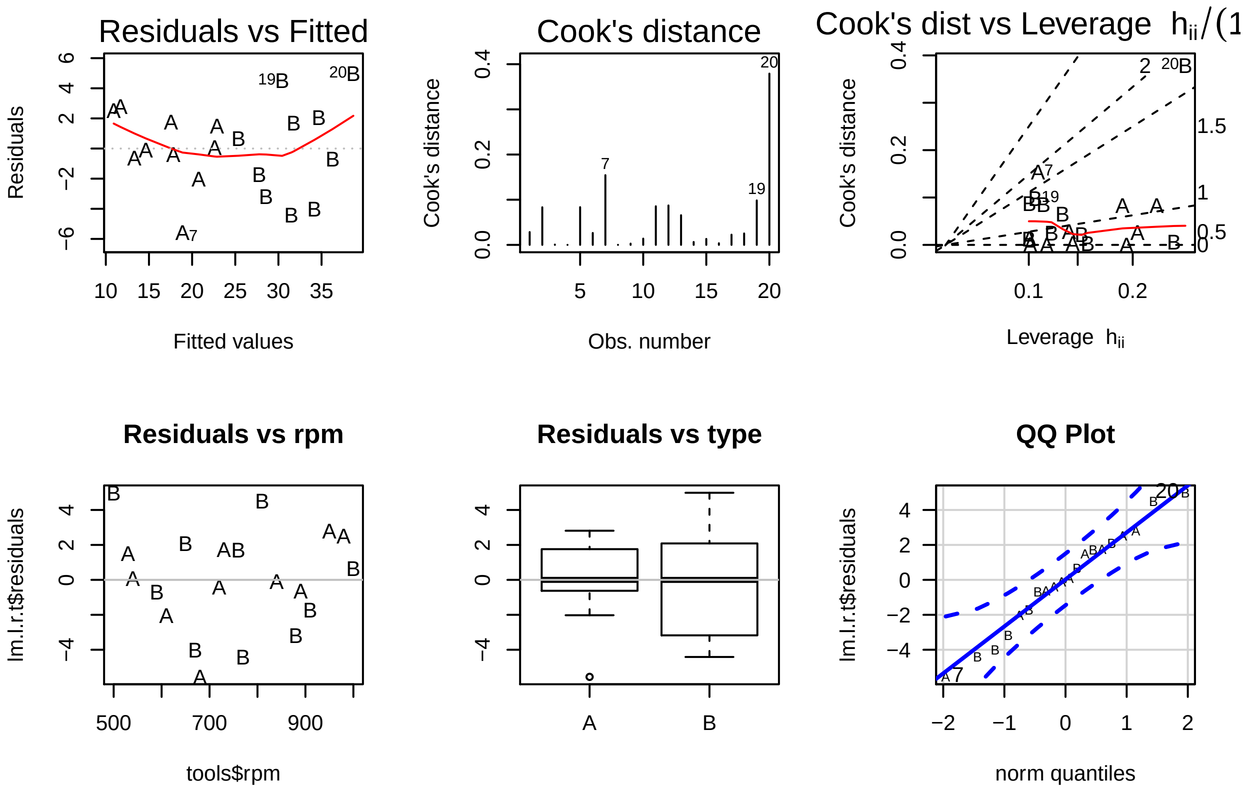
```r
lm.l.r.t <- lm(lifetime ~ rpm + type, data = tools)
#library(car)
#Anova(aov(lm.l.r.t), type=3)
summary(lm.l.r.t)
```

```
##
## Call:
## lm(formula = lifetime ~ rpm + type, data = tools)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5527 -1.7868 -0.0016  1.8395  4.9838
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.98560    3.51038  10.536 7.16e-09 ***
## rpm         -0.02661    0.00452  -5.887 1.79e-05 ***
## typeB       15.00425    1.35967  11.035 3.59e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.039 on 17 degrees of freedom
## Multiple R-squared:  0.9003, Adjusted R-squared:  0.8886
## F-statistic: 76.75 on 2 and 17 DF,  p-value: 3.086e-09
```

```r
# plot diagnostics
par(mfrow=c(2,3))
plot(lm.l.r.t, which = c(1,4,6), pch=as.character(tools$type))

plot(tools$rpm, lm.l.r.t$residuals, main="Residuals vs rpm", pch=as.character(tools$type))
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(tools$type, lm.l.r.t$residuals, main="Residuals vs type")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
```

```
## Loading required package: carData
```

```r
qqPlot(lm.l.r.t$residuals, las = 1, main="QQ Plot", pch=as.character(tools$type))
```




```
## [1]  7 20
```

```r
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot3.jpg")
```

```
## jpeg
##    3
```

```r
dev.off()
```

```
## pdf
##   2
```

```r
## residuals vs order of data
#plot(lm.l.r.t$residuals, main="Residuals vs Order of data")
#  # horizontal line at zero
#  abline(h = 0, col = "gray75")




#### Example: Twins
twins <- read.table("http://statacumen.com/teach/ADA2/ADA2_notes_Ch07_twins.dat"
                  , header = TRUE)
# set "L" as baseline level
twins$status <- relevel(twins$status, "L")
str(twins)
```

```
## 'data.frame':    27 obs. of  3 variables:
##  $ IQF   : int  82 80 88 108 116 117 132 71 75 93 ...
##  $ IQN   : int  82 90 91 115 115 129 131 78 79 82 ...
##  $ status: Factor w/ 3 levels "L","H","M": 2 2 2 2 2 2 2 3 3 3 ...
```

```r
library(ggplot2)
p <- ggplot(twins, aes(x = IQN, y = IQF, colour = status, shape = status))
p <- p + geom_point(size=4)
  library(R.oo) # for ascii code lookup
  p <- p + scale_shape_manual(values=charToInt(sort(unique(twins$status))))
p <- p + geom_smooth(method = lm, se = FALSE)
p <- p + labs(title="Twins data, IQF by IQN with categorical status")
  # equal axes since x- and y-variables are same quantity
  dat.range <- range(twins[,c("IQF","IQN")])
  p <- p + xlim(dat.range) + ylim(dat.range) + coord_equal(ratio=1)
print(p)
```


```r
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot4.jpg")
```

```
## jpeg
##    3
```

```r
dev.off()
```

```
## pdf
##   2
```

```r
lm.f.n.s.ns <- lm(IQF ~ IQN*status, data = twins)
library(car)
Anova(aov(lm.f.n.s.ns), type=3)
```

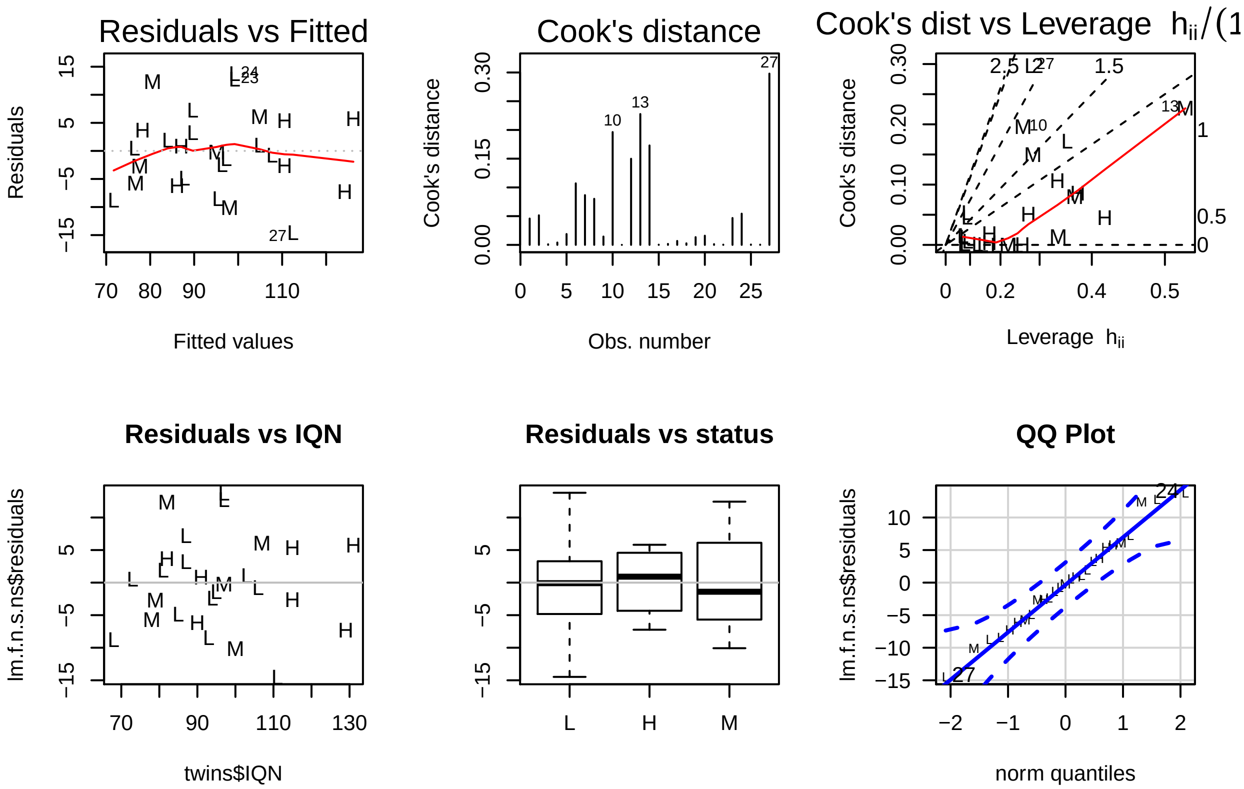
```
## Anova Table (Type III tests)
##
## Response: IQF
```

```
##               Sum Sq Df F value   Pr(>F)
## (Intercept)   11.61  1  0.1850   0.6715
## IQN         1700.39  1 27.1035 3.69e-05 ***
## status         8.99  2  0.0716   0.9311
## IQN:status     0.93  2  0.0074   0.9926
## Residuals   1317.47 21
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(lm.f.n.s.ns)
```

```
##
## Call:
## lm(formula = IQF ~ IQN * status, data = twins)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.479  -5.248  -0.155   4.582  13.798
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.20461   16.75126   0.430    0.672
## IQN          0.94842    0.18218   5.206 3.69e-05 ***
## statusH     -9.07665   24.44870  -0.371    0.714
## statusM     -6.38859   31.02087  -0.206    0.839
## IQN:statusH  0.02914    0.24458   0.119    0.906
## IQN:statusM  0.02414    0.33933   0.071    0.944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.921 on 21 degrees of freedom
## Multiple R-squared:  0.8041, Adjusted R-squared:  0.7574
## F-statistic: 17.24 on 5 and 21 DF,  p-value: 8.31e-07
```

```r
# plot diagnostics
par(mfrow=c(2,3))
plot(lm.f.n.s.ns, which = c(1,4,6), pch=as.character(twins$status))

plot(twins$IQN, lm.f.n.s.ns$residuals, main="Residuals vs IQN", pch=as.character(twins$status))
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(twins$status, lm.f.n.s.ns$residuals, main="Residuals vs status")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
qqPlot(lm.f.n.s.ns$residuals, las = 1, main="QQ Plot", pch=as.character(twins$status))
```




```
## [1] 27 24
```

```
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot5.jpg")
```

```
## jpeg
##    3
```

```
dev.off()
```

```
## pdf
##   2
```

```
## residuals vs order of data
#plot(lm.f.n.s.ns$residuals, main="Residuals vs Order of data")
#  # horizontal line at zero
#  abline(h = 0, col = "gray75")



lm.f.n.s <- lm(IQF ~ IQN + status, data = twins)
library(car)
Anova(aov(lm.f.n.s), type=3)
```

```
## Anova Table (Type III tests)
##
## Response: IQF
##             Sum Sq Df F value    Pr(>F)
## (Intercept)   18.2  1  0.3181    0.5782
## IQN         4674.7  1 81.5521 5.047e-09 ***
## status       175.1  2  1.5276    0.2383
## Residuals   1318.4 23
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(lm.f.n.s)
```

```
##
## Call:
## lm(formula = IQF ~ IQN + status, data = twins)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.8235  -5.2366  -0.1111   4.4755  13.6978
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.6188     9.9628   0.564    0.578
## IQN           0.9658     0.1069   9.031 5.05e-09 ***
## statusH      -6.2264     3.9171  -1.590    0.126
## statusM      -4.1911     3.6951  -1.134    0.268
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.571 on 23 degrees of freedom
## Multiple R-squared:  0.8039, Adjusted R-squared:  0.7784
## F-statistic: 31.44 on 3 and 23 DF,  p-value: 2.604e-08
```

```r
lm.f.n <- lm(IQF ~ IQN, data = twins)
library(car)
Anova(aov(lm.f.n), type=3)
```

```
## Anova Table (Type III tests)
##
## Response: IQF
##             Sum Sq Df F value    Pr(>F)
## (Intercept)   58.6  1  0.9802    0.3316
## IQN         5231.1  1 87.5630 1.204e-09 ***
## Residuals   1493.5 25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(lm.f.n)
```

```
##
## Call:
## lm(formula = IQF ~ IQN, data = twins)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3512  -5.7311   0.0574   4.3244  16.3531
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.20760    9.29990   0.990    0.332
## IQN          0.90144    0.09633   9.358  1.2e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 7.729 on 25 degrees of freedom
## Multiple R-squared:  0.7779, Adjusted R-squared:  0.769
## F-statistic: 87.56 on 1 and 25 DF,  p-value: 1.204e-09
```

```r
##compare models
anova(lm.f.n.s.ns,lm.f.n.s)
```

```
## Analysis of Variance Table
##
## Model 1: IQF ~ IQN * status
## Model 2: IQF ~ IQN + status
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     21 1317.5
## 2     23 1318.4 -2  -0.93181 0.0074 0.9926
```

```r
anova(lm.f.n.s,lm.f.n)
```

```
## Analysis of Variance Table
##
## Model 1: IQF ~ IQN + status
## Model 2: IQF ~ IQN
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     23 1318.4
## 2     25 1493.5 -2   -175.13 1.5276 0.2383
```

```r
#formal tests for normality and constant variance assumptions
shapiro.test(lm.f.n$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lm.f.n$residuals
## W = 0.96329, p-value = 0.4377
```

```r
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
bptest(IQF ~ IQN, data = twins,studentize=FALSE)
```

```
##
##  Breusch-Pagan test
##
## data:  IQF ~ IQN
## BP = 0.038304, df = 1, p-value = 0.8448
```

```r
#cooks distance
cooks.distance(lm.f.n)
```

```
##            1            2            3            4            5
## 7.811552e-04 4.028391e-02 3.799380e-03 2.374148e-02 9.778893e-03
##            6            7            8            9           10
```

```
## 2.083391e-01 7.440222e-02 6.039542e-02 2.266608e-02 6.014572e-02
##           11           12           13           14           15
## 9.185519e-04 4.740665e-02 1.568698e-02 1.003427e-01 4.815230e-03
##           16           17           18           19           20
## 9.466190e-03 4.002393e-03 1.269088e-02 3.865424e-02 1.249079e-02
##           21           22           23           24           25
## 1.110399e-06 4.468747e-04 7.983010e-02 9.056798e-02 6.621859e-03
##           26           27
## 2.578073e-03 9.362673e-02
```

```
highcook <- which((cooks.distance(lm.f.n)) > qf(0.5,2,25))
cooks.distance(lm.f.n)[highcook]
```

```
## named numeric(0)
```

```
# plot diagnostics
par(mfrow=c(2,3))
plot(lm.f.n, which = c(1,4,6), pch=as.character(twins$status))

plot(twins$IQN, lm.f.n.s.ns$residuals, main="Residuals vs IQN", pch=as.character(twins$status))
# horizontal line at zero
abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
qqPlot(lm.f.n$residuals, las = 1, main="QQ Plot", pch=as.character(twins$status))
```

```
## [1] 24 23
```

```
dev.copy(jpeg,filename="~/Desktop/jenn/teaching/ADA2/lecture notes/plots/ch07plot6.jpg")
```

```
## jpeg
##    3
```

```
dev.off()
```

```
## pdf
##   2
```

```
##recall general model
lm.f.n.s.ns <- lm(IQF ~ IQN*status, data = twins)
coef(lm.f.n.s.ns)
```

```
## (Intercept)         IQN     statusH     statusM IQN:statusH IQN:statusM
##  7.20460986  0.94842244 -9.07665352 -6.38858548  0.02913971  0.02414450
```

```
library(aod) # for wald.test()
# Typically, we are interested in testing whether individual parameters or
#   set of parameters are all simultaneously equal to 0s
# However, any null hypothesis values can be included in the vector coef.test.values.
coef.test.values <- rep(0, length(coef(lm.f.n.s.ns)))
wald.test(b = coef(lm.f.n.s.ns) - coef.test.values
        , Sigma = vcov(lm.f.n.s.ns)
        , Terms = c(5,6))
```

```
## Wald test:
## ----------
##
## Chi-squared test:
```

```
## X2 = 0.015, df = 2, P(> X2) = 0.99
```

```r
library(aod) # for wald.test()
coef.test.values <- rep(0, length(coef(lm.f.n.s.ns)))
wald.test(b = coef(lm.f.n.s.ns) - coef.test.values
        , Sigma = vcov(lm.f.n.s.ns)
        , Terms = c(4,6))
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 1.2, df = 2, P(> X2) = 0.55
```

```r
# Another way to do this is to define the matrix r and vector r, manually.
mR <- as.matrix(rbind(c(0, 0, 0, 1, 0, 0), c(0, 0, 0, 0, 0, 1)))
mR
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    0    0    1    0    0
## [2,]    0    0    0    0    0    1
```

```r
vR <- c(0, 0)
vR
```

```
## [1] 0 0
```

```r
wald.test(b = coef(lm.f.n.s.ns)
        , Sigma = vcov(lm.f.n.s.ns)
        , L = mR, H0 = vR)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 1.2, df = 2, P(> X2) = 0.55
```

```r
mR <- as.matrix(rbind(c(0, 0, 1, -1, 0, 0), c(0, 0, 0, 0, 1, -1)))
mR
```

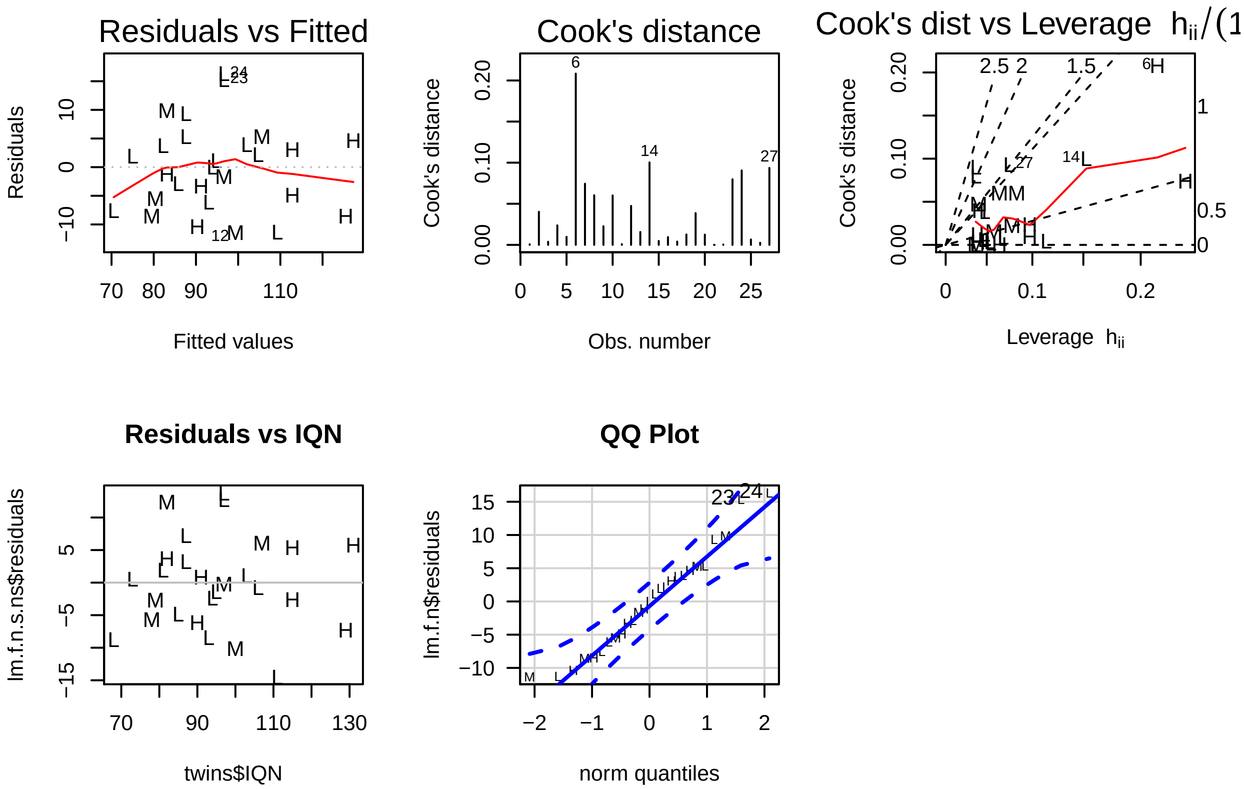
```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    0    1   -1    0    0
## [2,]    0    0    0    0    1   -1
```

```r
vR <- c(0, 0)
vR
```

```
## [1] 0 0
```

```r
wald.test(b = coef(lm.f.n.s.ns)
        , Sigma = vcov(lm.f.n.s.ns)
        , L = mR, H0 = vR)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 0.19, df = 2, P(> X2) = 0.91
```


Residuals vs Fitted
Residuals
Fitted values
Cook's distance
Cook's distance
Obs. number
Cook's dist vs Leverage
Cook's distance
Leverage h_{ii}
Residuals vs IQN
lm.f.n.s.ns$residuals
twins$IQN
QQ Plot
lm.f.n$residuals
norm quantiles