Hwk4

Dingxian Cao

# 5

n=100#sample size  
dat<- data.frame(x1=rnorm(n,3,2),x2=rexp(n,1),x3=rgamma(n,2,3))   
betaT<-c(beta0=2,beta1=1.2,beta2=-1.4,beta3=-0.5)  
muT<-as.vector(model.matrix(~x1+x2+x3,dat) %\*% betaT)#EY   
  
sigmasq<-runif(n,0.5,3)#variance  
sigma<-sqrt(sigmasq)  
yy<- muT+sigma\*rnorm(n)# ONE realization with different variance  
lm(yy~.,data=dat)

##   
## Call:  
## lm(formula = yy ~ ., data = dat)  
##   
## Coefficients:  
## (Intercept) x1 x2 x3   
## 2.3829488913872 1.1309187446949 -1.1914648156999 -1.0688531215963

lm(yy~.,data=dat,weights = 1/sigmasq)

##   
## Call:  
## lm(formula = yy ~ ., data = dat, weights = 1/sigmasq)  
##   
## Coefficients:  
## (Intercept) x1 x2   
## 2.23151090769105 1.13060983872591 -1.22384435687153   
## x3   
## -0.77027609908989

S<-10000 #simulation realization number  
YY<- muT + sigma \* matrix(rnorm(n \* S), nrow = n,ncol = S)  
  
beta\_diff<-function(y){  
 lmi<-lm(y~.,data = dat)  
 wlmi<-lm(y~.,data = dat,weights = 1/sigmasq)  
 c(lmi$coefficients-betaT,wlmi$coefficients-betaT)  
}  
beta\_dif<-apply(YY, 2,beta\_diff)  
  
beta\_dif\_lm<-beta\_dif[1:4,]  
beta\_dif\_wlm<-beta\_dif[5:8,]  
  
rowMeans(beta\_dif\_lm^2)#beta mse OF OLS

## (Intercept) x1 x2   
## 0.1170428204305474784 0.0051390544342502552 0.0146744368431458039   
## x3   
## 0.0852361571962295533

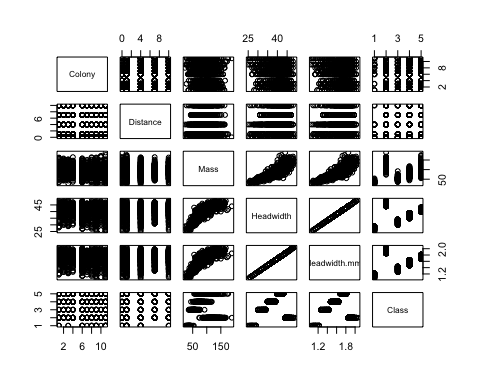
rowMeans(beta\_dif\_wlm^2)#beta mse OF WLS

## (Intercept) x1 x2   
## 0.0945511125447873746 0.0043695632031088725 0.0124599980288707633   
## x3   
## 0.0728182644475567076

# 6

### a

ants <- read.table("thatch\_ant\_c5del.txt", sep='', header=TRUE)  
pairs(ants)



There is no explicit relationship between Headwidth and colony or distance.

### b

ants2<-read.table("thatch\_ant\_c5del.txt", sep='', header=TRUE,  
 colClasses = c("factor","numeric","numeric","numeric","numeric"))  
#should take colony number as factor   
  
str(ants2)

## 'data.frame': 1104 obs. of 6 variables:  
## $ Colony : Factor w/ 10 levels "1","10","11",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Distance : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ Mass : num 109 120 94 61 72 134 94 113 111 106 ...  
## $ Headwidth : num 45 43 42 33 41 46 43 42 42 43 ...  
## $ Headwidth.mm.: num 1.9 1.81 1.77 1.39 1.73 ...  
## $ Class : Factor w/ 5 levels "<30",">43","30-34",..: 2 5 5 3 5 2 5 5 5 5 ...

result<-lm(Headwidth~0+Colony+Distance,data = ants2)  
summary(result)

##   
## Call:  
## lm(formula = Headwidth ~ 0 + Colony + Distance, data = ants2)  
##   
## Residuals:  
## Min 1Q Median   
## -14.05490008465392 -2.31778885140840 0.69894944968136   
## 3Q Max   
## 3.30228564245086 9.74771428434407   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## Colony1 40.627370229016130 0.468323702325208 86.75062 < 2.22e-16 \*\*\*  
## Colony10 40.054900084653909 0.477132748207408 83.94917 < 2.22e-16 \*\*\*  
## Colony11 38.146212626496670 0.465316330690553 81.97910 < 2.22e-16 \*\*\*  
## Colony2 40.524917683525672 0.493002283728282 82.20026 < 2.22e-16 \*\*\*  
## Colony3 39.084917683525688 0.493002283728282 79.27938 < 2.22e-16 \*\*\*  
## Colony4 40.364502744764003 0.460772597466875 87.60179 < 2.22e-16 \*\*\*  
## Colony6 38.899467304286460 0.434539585688822 89.51881 < 2.22e-16 \*\*\*  
## Colony7 39.514585328490497 0.482656717552078 81.86892 < 2.22e-16 \*\*\*  
## Colony8 39.788203674690223 0.502685948263432 79.15122 < 2.22e-16 \*\*\*  
## Colony9 38.530753141789440 0.461801670431354 83.43572 < 2.22e-16 \*\*\*  
## Distance 0.167368032130250 0.038568330948893 4.33952 1.5602e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1  
##   
## Residual standard error: 4.5582011795398003 on 1093 degrees of freedom  
## Multiple R-squared: 0.98752096240458, Adjusted R-squared: 0.98739537282219   
## F-statistic: 7863.0802303093 on 11 and 1093 DF, p-value: < 2.22044604925e-16

design<-model.matrix(~Colony+Distance-1,data = ants2)  
dim(design)

## [1] 1104 11

design[design[,11]==4  
 ,colnames(design)=="Colony4"]

## 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 122   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 141 142 143 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 225 226 227 228 229 230 231 306 307 308 309 310 311 312 313 314 315 316   
## 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1   
## 317 318 319 320 321 322 323 324 325 326 327 328 329 330 415 416 417 418   
## 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0   
## 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 437 438 439 440 441 442 443 444 445 446 447 448 449 522 523 524 525 526   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 545 546 547 548 609 610 611 612 613 614 615 616 617 618 619 620 621 622   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 623 624 625 626 627 628 629 701 702 703 704 705 706 707 708 709 710 711   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 712 713 714 715 716 717 718 719 720 721 798 799 800 801 802 803 804 805   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 806 807 808 809 810 811 812 813 814 815 816 817 818 900 901 902 903 904   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 923 924   
## 0 0

the coefficients of the each colony means the average headwidth in that colony at distance 0. And the coefficient of distance means that the average headwidth will grow 0.167 per meter.

m1<-lm(Headwidth~Colony+Distance-1,data = ants2)  
m2<-lm(Headwidth~1,data = ants2)  
anova(m1,m2)

## Analysis of Variance Table  
##   
## Model 1: Headwidth ~ Colony + Distance - 1  
## Model 2: Headwidth ~ 1  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 1093 22709.4774065221   
## 2 1103 23847.3333333333 -10 -1137.85592681124 5.47646 5.6827e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

H0: beta1.=beta2=0  
test statistic= . it's distribution is F(11,1093), the result is to reject the Null hypothesis which means headwidth is related to colony and distance.

m1<-lm(Headwidth~Colony+Distance-1,data = ants2)  
m2<-lm(Headwidth~Colony-1,data = ants2)  
anova(m1,m2)

## Analysis of Variance Table  
##   
## Model 1: Headwidth ~ Colony + Distance - 1  
## Model 2: Headwidth ~ Colony - 1  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 1093 22709.4774065221   
## 2 1094 23100.7417824617 -1 -391.264375939616 18.83143 1.5602e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

H0: beta2=0  
test statistic= . it's distribution is F(1,1093), the result is to reject the Null hypothesis which means headwidth is related to distance.

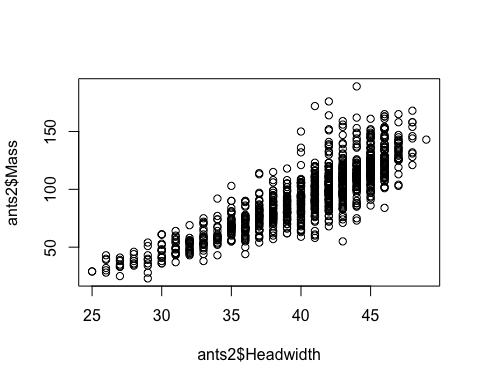
### c

library(lme4)  
library(car)  
m1<-lm(Headwidth~Colony+Distance-1,data = ants2)  
m1$coefficients  
rhs <- rep(0,9)  
hm<-rbind(c(0,1,0,0,0,0,0,0,-1,0,0)  
 ,c(1,-2,0,0,0,0,0,0,0,0,0)  
 ,c(0,-2,1,0,0,0,0,0,0,0,0)  
 ,c(0,-2,0,1,0,0,0,0,0,0,0)  
 ,c(0,-2,0,0,1,0,0,0,0,0,0)  
 ,c(0,-2,0,0,0,1,0,0,0,0,0)  
 ,c(0,-2,0,0,0,0,1,0,0,0,0)  
 ,c(0,-2,0,0,0,0,0,1,0,0,0)  
 ,c(0,-2,0,0,0,0,0,0,0,1,0))  
linearHypothesis(m1,hm,rhs)

we should reject null hypothesis.

### d

plot(ants2$Headwidth,ants2$Mass)



m1\_e<-lm(Headwidth~Colony+Distance+Mass-1,data = ants2)  
summary(m1\_e)

##   
## Call:  
## lm(formula = Headwidth ~ Colony + Distance + Mass - 1, data = ants2)  
##   
## Residuals:  
## Min 1Q Median 3Q   
## -9.4943227071750 -1.4630029672001 0.0833239427996 1.5352005467690   
## Max   
## 7.0764367285327   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## Colony1 26.9536307845344467 0.3469959861032736 77.67707 < 2.22e-16 \*\*\*  
## Colony10 26.2465755402734828 0.3519234070243870 74.58036 < 2.22e-16 \*\*\*  
## Colony11 25.4439095656268037 0.3334045755804524 76.31542 < 2.22e-16 \*\*\*  
## Colony2 26.9479144111577220 0.3547484018748394 75.96346 < 2.22e-16 \*\*\*  
## Colony3 25.9244311761678503 0.3495023446461054 74.17527 < 2.22e-16 \*\*\*  
## Colony4 27.0170571779681197 0.3400203754510464 79.45717 < 2.22e-16 \*\*\*  
## Colony6 26.5068203775475659 0.3181337603287494 83.31973 < 2.22e-16 \*\*\*  
## Colony7 26.7464013150806004 0.3407208655307710 78.49945 < 2.22e-16 \*\*\*  
## Colony8 26.8115216652756558 0.3508910857981444 76.40981 < 2.22e-16 \*\*\*  
## Colony9 25.3888833006427710 0.3377277167921452 75.17560 < 2.22e-16 \*\*\*  
## Distance 0.1262375434196212 0.0199122736649178 6.33969 3.3629e-10 \*\*\*  
## Mass 0.1402413350202434 0.0025543455516245 54.90304 < 2.22e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1  
##   
## Residual standard error: 2.3516675069870998 on 1092 degrees of freedom  
## Multiple R-squared: 0.99668144951987, Adjusted R-squared: 0.99664498193217   
## F-statistic: 27330.610894499 on 12 and 1092 DF, p-value: < 2.22044604925e-16

### e

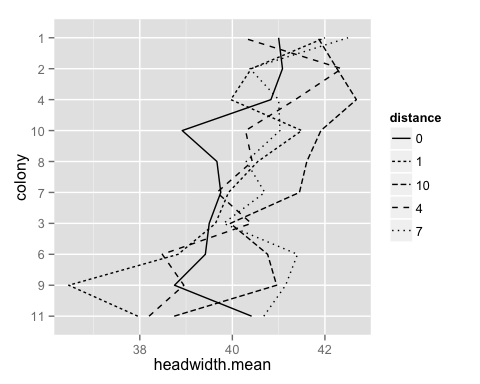
library(reshape2)  
library(dplyr)

##   
## Attaching package: 'dplyr'  
##   
## The following objects are masked from 'package:stats':  
##   
## filter, lag  
##   
## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)  
  
ants3<-read.table("thatch\_ant\_c5del.txt", sep='', header=TRUE,  
 colClasses=c("factor","factor","numeric","numeric","numeric"))  
  
av1<-aov(Headwidth~Colony\*Distance,data = ants3)#two factors  
  
result<-model.tables(av1,type = "means")  
interact<-result$tables$`Colony:Distance`  
interact<-interact[c(3,10,7,5,8,9,2,6,4,1),]   
  
inter<-matrix(as.vector(interact),nrow=10)  
colnames(inter)<-colnames(interact);rownames(inter)<-rownames(interact)  
inter<- as.data.frame(t(inter))  
  
inter$distance<-rownames(inter)  
inter<-melt(inter)

## Using distance as id variables

inter<-rename(inter,colony=variable,headwidth.mean=value)  
ggplot(inter,aes(x =headwidth.mean,y=colony,group=distance)) + geom\_path(aes(lty=distance))



reproduce the plot of the problem.

m1<-aov(Headwidth~Colony\*Distance,data = ants3)   
m2<-aov(Headwidth~1,data = ants3)  
anova(m1,m2)

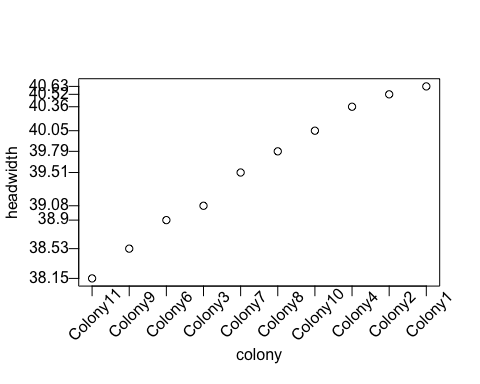
## Analysis of Variance Table  
##   
## Model 1: Headwidth ~ Colony \* Distance  
## Model 2: Headwidth ~ 1  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 1054 21921.3659154694   
## 2 1103 23847.3333333333 -49 -1925.96741786395 1.88984 0.00027006 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

H0:the effect of colony and distance is not significant.  
result: reject the nul hypothesis, so the data supports the general observation from this plot.

Another way to use variable distance: we can treat as not a factor but a normal variable into the model.

### f

m1<-lm(Headwidth~Colony+Distance-1,data = ants2)#one factors  
result<-summary(m1)  
interact<-result$coefficients[-11,1]  
interact<-interact[c(3,10,7,5,8,9,2,6,4,1)]   
inter<- as.data.frame((interact))  
inter$colony<-rownames(inter)  
colnames(inter)[1]<-"headwidth.mean"  
inter<-arrange(inter,headwidth.mean)  
  
plot(inter$headwidth.mean,axes = F,xlab="colony",ylab="headwidth")  
  
lablist.x<-inter$colony  
lablist.y<-inter$headwidth.mean  
axis(1, at=seq(1, 10, by=1), labels = FALSE)  
  
text(x = seq(1, 10, by=1), par("usr")[3] - 0.2, labels = lablist.x, srt = 45, pos = 1, xpd = TRUE)  
  
axis(2, at=inter$headwidth.mean, labels = FALSE)   
  
text(y=inter$headwidth.mean, par("usr")[1], labels = round(lablist.y,2), pos = 2, xpd = TRUE)   
box()



m1<-lm(Headwidth~Colony+Distance-1,data = ants2)  
m2<-lm(Headwidth~1,data = ants2)  
anova(m1,m2)

## Analysis of Variance Table  
##   
## Model 1: Headwidth ~ Colony + Distance - 1  
## Model 2: Headwidth ~ 1  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 1093 22709.4774065221   
## 2 1103 23847.3333333333 -10 -1137.85592681124 5.47646 5.6827e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

# 7

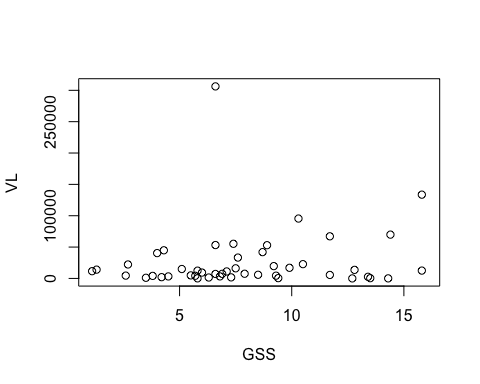
no data on the dropbox

### a&b

library(car)  
DataURL <- "http://www.stat.wisc.edu/~st849-1/data/"  
str(ql <- read.table(paste(DataURL, "hmw3q1\_data.txt", sep=""), header=TRUE))

## 'data.frame': 48 obs. of 2 variables:  
## $ GSS: num 4 13.4 7.4 2.7 13.5 10.3 11.7 4.5 14.3 5.8 ...  
## $ VL : int 40406 2603 55246 22257 400 95505 5537 3205 90 12394 ...

plot(ql)



outlierTest(lm(VL~GSS,data = ql))

## rstudent unadjusted p-value Bonferonni p  
## 28 10.584279549417912 1.1262e-13 5.2930999999999997e-12

outlierTest(lm(VL~GSS,data = ql[-28,]))

## rstudent unadjusted p-value Bonferonni p  
## 16 4.4996439861964017 5.1131000000000002e-05 0.0023519999999999999

outlierTest(lm(VL~GSS,data = ql[-c(16,28),]))

## rstudent unadjusted p-value Bonferonni p  
## 6 3.907709169616659 0.00033307999999999998 0.014989000000000001

outlierTest(lm(VL~GSS,data = ql[-c(6,16,28),]))

##   
## No Studentized residuals with Bonferonni p < 0.050000000000000003  
## Largest |rstudent|:  
## rstudent unadjusted p-value Bonferonni p  
## 44 2.8884453793703928 0.0061574000000000004 0.27091999999999999

outlierTest(lm(VL~GSS,data = ql[-c(6,16,28,44),]))

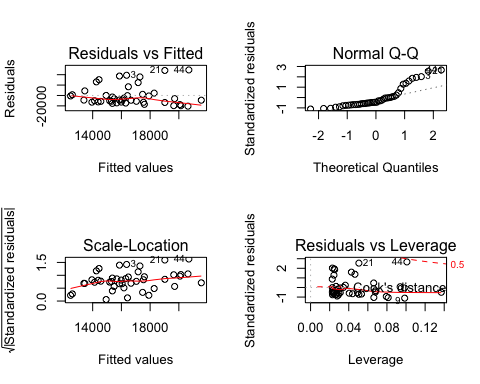
##   
## No Studentized residuals with Bonferonni p < 0.050000000000000003  
## Largest |rstudent|:  
## rstudent unadjusted p-value Bonferonni p  
## 21 3.3229078831048207 0.0019123 0.082226999999999995

model<-lm(VL~GSS,data = ql[-c(6,16,28),])  
summary(model)

##   
## Call:  
## lm(formula = VL ~ GSS, data = ql[-c(6, 16, 28), ])  
##   
## Residuals:  
## Min 1Q Median 3Q   
## -20544.4703669918 -12787.4039684058 -8729.0317712772 2756.1960958608   
## Max   
## 49173.6099165087   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 11779.95090755877 6899.95902292302 1.70725 0.095159 .  
## GSS 619.19716499531 811.54292396476 0.76299 0.449735   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1  
##   
## Residual standard error: 19460.438736979999 on 42 degrees of freedom  
## (1 observation deleted due to missingness)  
## Multiple R-squared: 0.013671222197528, Adjusted R-squared: -0.0098127963215788   
## F-statistic: 0.58215003477385 on 1 and 42 DF, p-value: 0.44973525666658

### c

par(mfrow=c(2,2))  
plot(model,ask = FALSE)

 >it seems that constant variance and normality assumption are both violated.

### d

b<-boxCox(VL~GSS,data=ql,lambda = seq(-2, 2, 1/10) )

## Warning in plot.window(...): "data" is not a graphical parameter

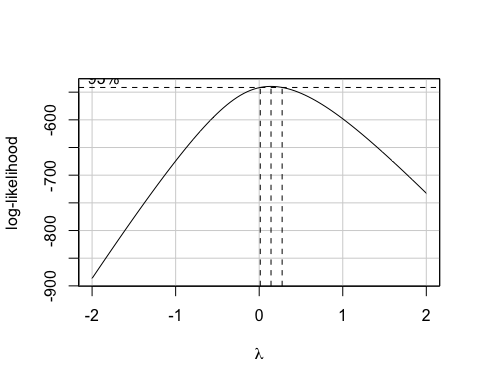
## Warning in plot.xy(xy, type, ...): "data" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "data" is not  
## a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "data" is not  
## a graphical parameter

## Warning in box(...): "data" is not a graphical parameter

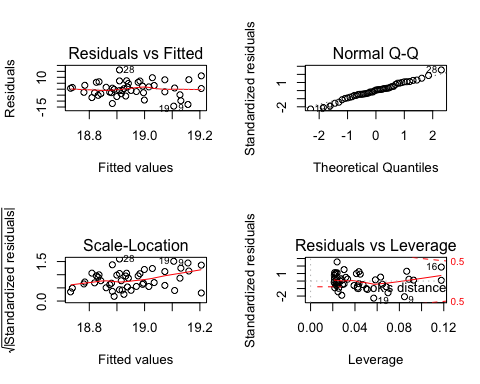
## Warning in title(...): "data" is not a graphical parameter



lambda<- b$x[which.max(b$y)]  
ql$y<-(ql$VL^lambda-1)/lambda  
model2<-lm(y~GSS,data = ql)  
summary(model2)

##   
## Call:  
## lm(formula = y ~ GSS, data = ql)  
##   
## Residuals:  
## Min 1Q Median   
## -14.26274654425964 -3.38560545171216 0.41149621782945   
## 3Q Max   
## 4.04458689602684 16.21973480377125   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 18.696884638374907 2.212235627207622 8.45158 7.6605e-11 \*\*\*  
## GSS 0.032167668241144 0.253477493745932 0.12691 0.89958   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1  
##   
## Residual standard error: 6.4461075959493996 on 45 degrees of freedom  
## (1 observation deleted due to missingness)  
## Multiple R-squared: 0.00035776054075995, Adjusted R-squared: -0.021856511447223   
## F-statistic: 0.01610498606272 on 1 and 45 DF, p-value: 0.89958062788812

par(mfrow=c(2,2))  
plot(model2,ask = FALSE)



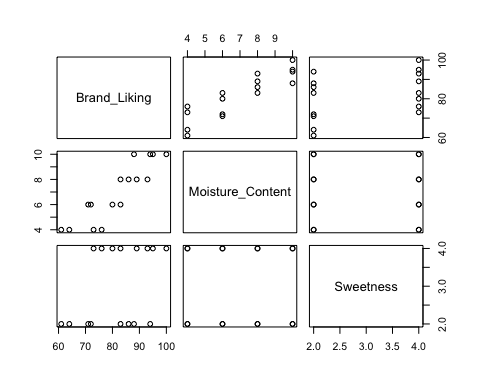
# 8

### a

str(br <- read.table("brand\_preference.txt", sep="", header=TRUE))

## 'data.frame': 16 obs. of 3 variables:  
## $ Brand\_Liking : num 64 73 61 76 72 80 71 83 83 89 ...  
## $ Moisture\_Content: num 4 4 4 4 6 6 6 6 8 8 ...  
## $ Sweetness : num 2 4 2 4 2 4 2 4 2 4 ...

pairs(br)



### b

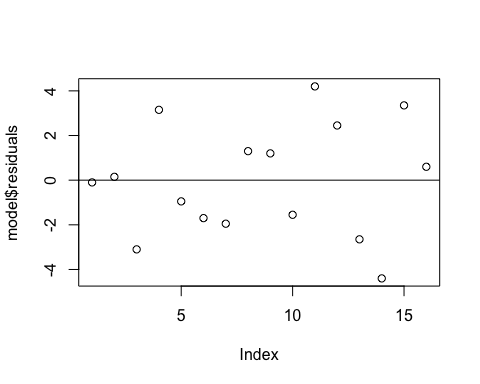
model<-lm(Brand\_Liking~Moisture\_Content+Sweetness,data = br)  
summary(model)

##   
## Call:  
## lm(formula = Brand\_Liking ~ Moisture\_Content + Sweetness, data = br)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.4000 -1.7625 0.0250 1.5875 4.2000   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 37.65000000000001 2.99610323843899 12.56632 1.1996e-08  
## Moisture\_Content 4.42500000000000 0.30111970530518 14.69515 1.7780e-09  
## Sweetness 4.37500000000000 0.67332413042708 6.49761 2.0110e-05  
##   
## (Intercept) \*\*\*  
## Moisture\_Content \*\*\*  
## Sweetness \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1  
##   
## Residual standard error: 2.6932965217082998 on 13 degrees of freedom  
## Multiple R-squared: 0.95205897305541, Adjusted R-squared: 0.94468343044855   
## F-statistic: 129.08324496288 on 2 and 13 DF, p-value: 2.658261267082e-09

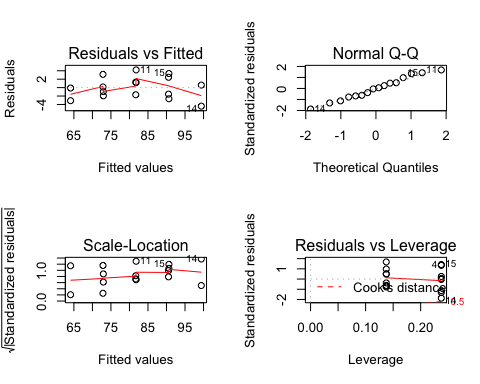
keep sweetness equal, the average increase of brand\_liking score is 4.43.

### c

par(mfrow=c(1,1))  
plot(model$residuals)  
abline(h=mean(model$residuals))#E(e)=0



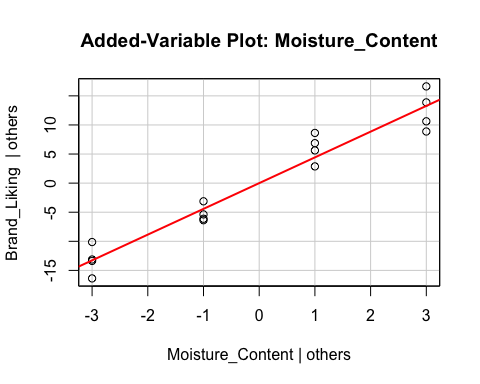
par(mfrow=c(2,2))  
plot(model,ask=FALSE)



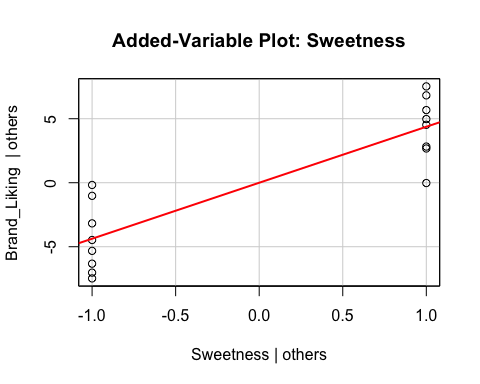
it seems the model satisfy the Gauss-Markov assumption.

### d

library(car)  
avPlot(model,"Moisture\_Content")



avPlot(model,"Sweetness")

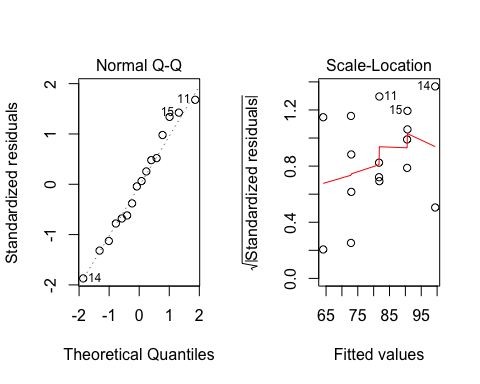


### e

According to the added-variable plot, the model in part b is appropriate that both variables are important.

### f

layout(matrix(1:2,ncol=2))  
plot(model,which=c(2,3))



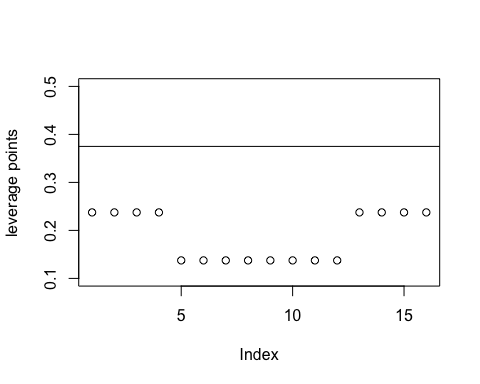
ext\_sr<- rstudent(model)#t(16-3-1)  
# which(abs(ext\_sr)>2)   
(1-pt(abs(ext\_sr),df = 16-3-1))<0.1/2#the true are the outliers

## 1 2 3 4 5 6 7 8 9 10 11 12   
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE   
## 13 14 15 16   
## FALSE TRUE FALSE FALSE

the desicion rule is that if the absolute value of external residual is larger than the critical value(but i use p value instead), we should take it as the outlier. The result shows that the outliers are the 11th and 14th points.

### g

par(mfrow=c(1,1))  
x<-model.matrix(model)  
lev<-hat(x)  
plot(lev,ylim = c(0.1,0.5),ylab = "leverage points")  
abline(h=(2\*3)/16)



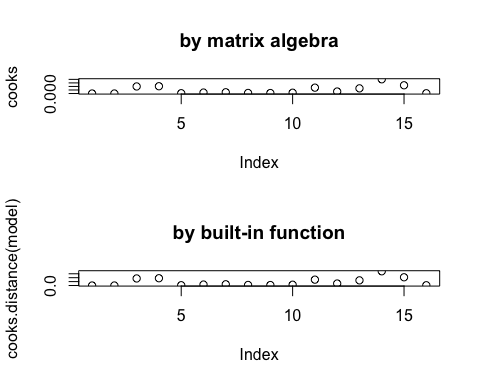
there is no leverage point.

### h

e<-diag(model$residuals)  
h<-diag(1/(1-hat(x)))  
H<-x%\*%solve(t(x)%\*%x)%\*%t(x)  
(cooks<-apply(((h%\*%e%\*%H)^2)/(3\*var(br$Brand\_Liking)),1,sum))

## [1] 1.0383638555377820e-05 2.3363186749601975e-05 9.9786766517192677e-03  
## [4] 1.0303165356574837e-02 4.2403074377726145e-04 1.3578380604058523e-03  
## [7] 1.7865672057762159e-03 7.9402986923387208e-04 6.7656982940637498e-04  
## [10] 1.1287909827422343e-03 8.2879804102281018e-03 2.8202155562581763e-03  
## [13] 7.2919101755149077e-03 2.0102724243213781e-02 1.1653038368774089e-02  
## [16] 3.7381098799364547e-04

par(mfrow=c(2,1))  
plot(cooks,main = "by matrix algebra")  
plot(cooks.distance(model),main = "by built-in function")



# 9

### a

str(hayfever <- within(read.table("hayfever.txt", sep="", header = TRUE),  
{  
 A <- factor(A)  
 B <- factor(B)  
 id <- factor(id)  
}))

## 'data.frame': 36 obs. of 4 variables:  
## $ hours: num 2.4 2.7 2.3 2.5 4.6 4.2 4.9 4.7 4.8 4.5 ...  
## $ A : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...  
## $ B : Factor w/ 3 levels "1","2","3": 1 1 1 1 2 2 2 2 3 3 ...  
## $ id : Factor w/ 4 levels "1","2","3","4": 1 2 3 4 1 2 3 4 1 2 ...

model1<-aov(hours~A\*B,data = hayfever)  
(effect<-model.tables(model1))

## Tables of effects  
##   
## A   
## A  
## 1 2 3   
## -3.29999999999999982 0.65000000000000002 2.64999999999999991   
##   
## B   
## B  
## 1 2 3   
## -2.5499999999999998 0.7500000000000000 1.8000000000000000   
##   
## A:B   
## B  
## A 1 2 3   
## 1 1.141699999999999937 -0.033300000000000003 -1.108300000000000063  
## 2 0.166699999999999987 0.341700000000000004 -0.508299999999999974  
## 3 -1.308300000000000018 -0.308300000000000018 1.616700000000000026

(means<-model.tables(model1,type = "means"))

## Tables of means  
## Grand mean  
##   
## 7.1833333333333318   
##   
## A   
## A  
## 1 2 3   
## 3.8830000000000000 7.8330000000000002 9.8330000000000002   
##   
## B   
## B  
## 1 2 3   
## 4.6330000000000000 7.9329999999999998 8.9830000000000005   
##   
## A:B   
## B  
## A 1 2 3   
## 1 2.4750000000000001 4.5999999999999996 4.5750000000000002  
## 2 5.4500000000000002 8.9250000000000007 9.1250000000000000  
## 3 5.9749999999999996 10.2750000000000004 13.2500000000000000

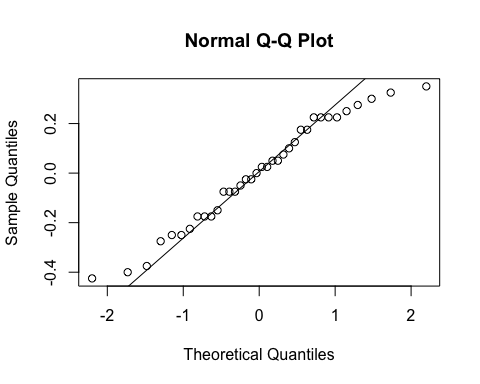
7.1833+0.65+1.8-0.5083

## [1] 9.125

since the model is from the table we can see that when factor A=2 and factor B=3, the estimated mean is about 9.125.

### b

qqnorm(model1$residuals)  
qqline(model1$residuals)



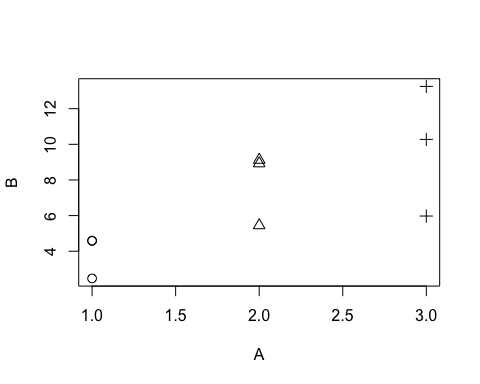
Apparently, it violate the nomality assumption.

### c

levels(hayfever$A)

## [1] "1" "2" "3"

x<-rep(1:3,3)  
y<-as.vector(means$tables$`A:B`)  
plot(x,y,pch=1:3,cex=1.2,xlab = "A",ylab = "B")



>apparently, there is interaction between A AND B.

### d&e

summary(model1)

## Df Sum Sq Mean Sq F value Pr(>F)   
## A 2 220.020 110.010000000000 1827.85846 < 2.22e-16 \*\*\*  
## B 2 123.660 61.830000000000 1027.32923 < 2.22e-16 \*\*\*  
## A:B 4 29.425 7.356250000000 122.22692 < 2.22e-16 \*\*\*  
## Residuals 27 1.625 0.060185185185   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

model2<-aov(hours~A+B,data = hayfever)  
anova(model1,model2)

## Analysis of Variance Table  
##   
## Model 1: hours ~ A \* B  
## Model 2: hours ~ A + B  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 27 1.625   
## 2 31 31.050 -4 -29.425 122.22692 < 2.22e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.050000000000000003 '.'  
## 0.10000000000000001 ' ' 1

they are significant at level of 0.05.