R Notebook

load rdm

rdm\_ind = function(g1, g2, nreps){  
   
 # g1: data before treatment  
 # g2: data after treatment  
   
 n1 = length(g1);  
 n2 = length(g2);  
 n = n1+n2; # number of observations  
   
 diffObt = mean(g2)- mean(g1);  
   
 aux = c(g1, g2); # all observations  
   
 out = numeric(nreps); # to store differences  
   
 for(i in 1:nreps){  
   
 d = sample(aux, size = n, replace = FALSE);  
   
 out[i] = mean(d[(n1+1):n]) - mean(d[1:n1]);  
   
 }  
   
 highprob <- length(out[out >= abs(diffObt)])/nreps;  
 lowprob <- length(out[out <= -abs(diffObt)])/nreps;  
 prob2tailed <- lowprob + highprob;  
   
 return(list(pvalue = prob2tailed, resampMeanDiff = out,  
 diffObt = diffObt, nreps = nreps));  
   
}

set up the data

drug <- read.table("~/Downloads/abDrugEffect.dat",header = TRUE, check.names = TRUE)  
x = cbind(drug[drug$treatment == "1", c("treatment","score1")],   
 drug[drug$treatment == "1",c("score2")])  
colnames(x) = c("treatment","score1", "score2")  
g1=drug[,4]; # Drug A score  
g2=drug[,5]; # Drug B score  
x=cbind(g1,g2)

print x

x

## g1 g2  
## [1,] 6.8496 1.3007  
## [2,] 14.7009 14.4018  
## [3,] 8.9982 2.9965  
## [4,] 7.5940 0.1880  
## [5,] 14.2160 13.4321  
## [6,] 14.6928 14.3855  
## [7,] 10.4298 5.8596  
## [8,] 10.3169 5.6338  
## [9,] 5.4979 4.0041  
## [10,] 5.6657 3.6687  
## [11,] 13.1932 11.3864  
## [12,] 10.2387 5.4774  
## [13,] 13.5339 12.0679  
## [14,] 5.6718 3.6563  
## [15,] 14.5702 14.1405  
## [16,] 7.9719 0.9439  
## [17,] 7.7261 0.4522  
## [18,] 11.8993 8.7986  
## [19,] 14.7676 14.5353  
## [20,] 7.2651 0.4698  
## [21,] 11.8824 8.7647  
## [22,] 9.1276 3.2553  
## [23,] 10.5855 6.1711  
## [24,] 7.8723 0.7445

load randomization for independent variables

set.seed(1)  
R = rdm\_ind(g1, g2, nreps = 10000)

view r

View(R)

hist

pdf("drug.pdf")  
dev.new()  
hist(R$resampMeanDiff, xlab = "Mean difference", freq = F,  
 main = "Distribution under Null hypothesis")  
mtext("drugB - drugA", 3)  
text(R$diffObt, 0.3, expression(paste("obs diff",sep="")))  
arrows(R$diffObt, 75, R$diffObt, 0, length = .125);  
text(4, 0.2, paste("p-value= ", R$pvalue ,sep=""))

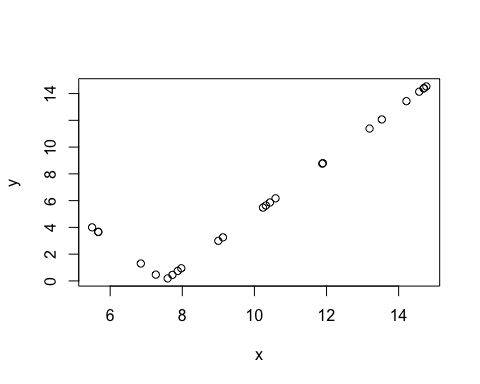
95% CI

n\_psc = length(g1); # number of observations in drugA group  
n\_npsc = length(g2); # number of observations in drugB group  
  
N = 1000; # number ofbootstrap samples  
out = NULL; # number of bootstrap samples  
  
for(i in 1:N){  
   
 b\_psc = sample(1:n\_psc, n\_psc, replace = TRUE);  
 b\_npsc = sample(1:n\_npsc, n\_npsc, replace = TRUE);  
   
 out = c(out, mean(g2[b\_npsc]) - mean(g2[b\_psc]));  
   
}  
  
dev.new()  
hist(out,main = "Bootstrap sample",xlab ="p\_hat");  
  
round(quantile(out, probs = c(0.025,0.975)), 4);#

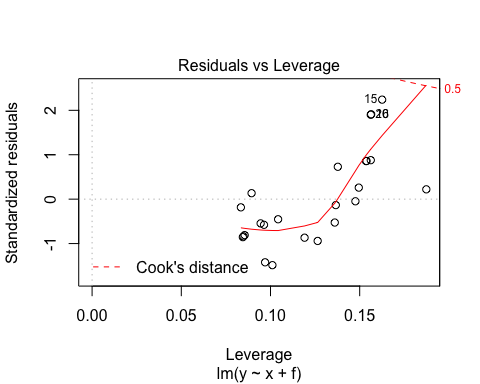
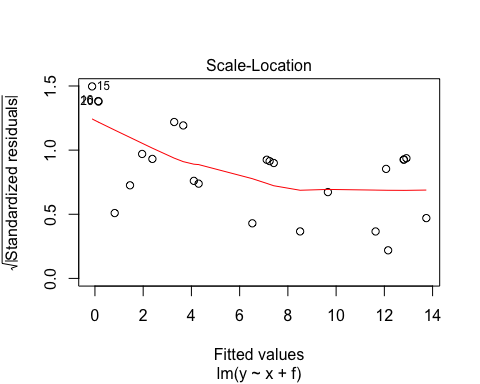
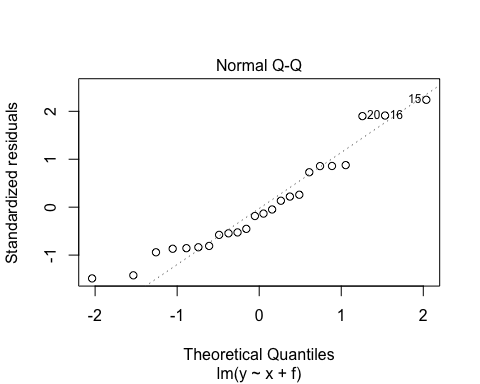
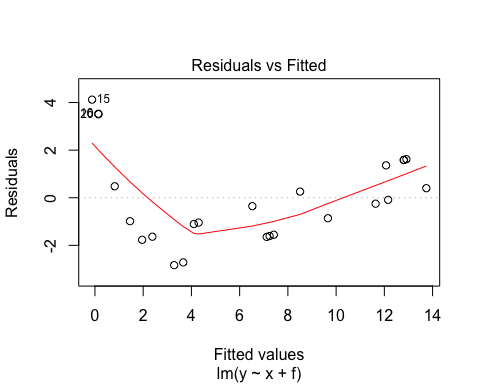
## 2.5% 97.5%   
## -2.7848 2.7343

covariate

Ax=drug[drug$treatment == "1", "score1"];  
Ay=drug[drug$treatment == "1", "score2"];  
Bx=drug[drug$treatment == "2", "score1"];  
By=drug[drug$treatment == "2", "score2"];  
  
y = c(Ay,By)   
x = c(Ax,Bx)  
  
n = length(y);  
  
f = c(rep("A", length(Ax)), rep("B", length(Bx))); # factor  
f = as.factor(f);  
  
fit = lm(y ~ x + f); # linear model  
  
plot(x,y);



plot(fit);



summary(fit);

##   
## Call:  
## lm(formula = y ~ x + f)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.8333 -1.5691 -0.3026 1.4166 4.1222   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -9.6432 1.5471 -6.233 3.50e-06 \*\*\*  
## x 1.5275 0.1339 11.410 1.83e-10 \*\*\*  
## fB 1.1270 0.8314 1.356 0.19   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.011 on 21 degrees of freedom  
## Multiple R-squared: 0.8613, Adjusted R-squared: 0.8481   
## F-statistic: 65.21 on 2 and 21 DF, p-value: 9.809e-10

anova(fit);

## Analysis of Variance Table  
##   
## Response: y  
## Df Sum Sq Mean Sq F value Pr(>F)   
## x 1 519.81 519.81 128.5746 2.055e-10 \*\*\*  
## f 1 7.43 7.43 1.8376 0.1896   
## Residuals 21 84.90 4.04   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Randomization test ####  
  
B = 10000;  
out = NULL;  
  
set.seed(1)  
for(i in 1:B){  
   
 b = sample(1:n, n, replace = FALSE);  
   
 bfit = lm(y[b] ~ x[b] + f);  
   
 out = c(out, summary(bfit)$coefficients[3,1]);  
   
}  
  
obs = summary(fit)$coefficients[3,1];  
pvalue = (sum(abs(obs) <= out) + sum(-abs(obs)>=out)) / B;  
pvalue

## [1] 0.1928

pdf("food\_AB.pdf")  
par(mar=c(4,4,3,1));  
dev.new()  
hist(out, xlab = expression(beta[compB]),  
 main = "Distribution under Null hypothesis", freq = F);  
obs = summary(fit)$coefficients[3,1];  
text(obs, 0.39, expression(paste("obs ", beta[compB]),sep=""));  
arrows(obs, 0.38, obs, 0, length = .125);  
text(2.5, .3, paste("p-value= ", pvalue ,sep=""));  
dev.off()

## quartz\_off\_screen   
## 2

CI 95

B = 10000;  
out = NULL;  
  
for(i in 1:B){  
   
 b = sample(1:n, n, replace = TRUE);  
   
 f1 = relevel(f[b], "A"); # A as baseline  
 bfit = lm(y[b] ~ x[b] + f1);  
   
 out = c(out, summary(bfit)$coefficients[3,1]);  
   
}  
  
round(quantile(out, prob = c(0.025, 0.975)), 3); # bootstrap CI

## 2.5% 97.5%   
## -0.304 2.635

dev.new()  
hist(out, xlab = "Difference in average");  
text(4, 1500, paste("p-value= ", pvalue ,sep=""));  
obs = summary(fit)$coefficients[3,1];  
abline(v = obs, lty = 2);

95ci

y <- read.csv("~/Downloads/cavendish.txt", sep="")  
x <- y$Density;  
n = length(x);  
xbar = mean(x);  
  
pdf("cave.pdf")  
par(mar=c(4,4,3,1))  
dev.new()  
hist(x, freq=F, main = "The density of the earth",  
 xlab = "X");  
  
mu\_prior = 0;  
sigma\_prior = 1;  
(mu\_prior/sigma\_prior)^(-1)

## [1] Inf

# Posterior parameters  
mu\_post = (n\*xbar)/((n/xbar^2)+(1/sigma\_prior^2));  
sigma\_post = ((n/xbar^2)+(1/sigma\_prior^2))^-1/2;  
(mu\_post/sigma\_post)^(-1)

## [1] 0.003164757

# credible interval  
round(qnorm(c(0.025, 0.975), mean=mu\_post, sd=sigma\_post),4);

## [1] 79.4147 80.4060

Gibbs

gibbs = function(n, mu1, mu2, s1, s2, rho, x0){  
   
 x1\_0 = x0[1];  
 x2\_0 = x0[2];  
  
 out = NULL;  
   
 for(i in 1:n){  
   
 x1\_0 = rnorm(1, mean = mu1 + rho \* s1 \* (x2\_0 - mu2) / s2,   
 sd = sqrt((1-rho^2) \* s1^2));  
   
 x2\_0 = rnorm(1, mean = mu2 + rho \* s2 \* (x1\_0 - mu1) / s1,   
 sd = sqrt((1-rho^2) \* s2^2));  
   
 out = rbind(out, c(x1\_0, x2\_0));  
   
 }  
   
 return(out);  
   
}   
# Specifications  
n = 29; # sample size  
mu = c(5.4,5);  
rho = 0.0;  
s1 = 0.22;  
s2 = 0.12;  
  
set.seed(1)  
R1 = gibbs(n = 29, mu1 = mu[1], mu2 = mu[2], s1 = s1, s2 = s2,   
 rho = rho, x0 = c(1,1));  
  
pdf("lab2\_case1\_gibbs.pdf", width = 14, height = 3.5)  
layout(matrix(c(1,3,4,5,2,3,4,5), 2, 4, byrow = TRUE))  
par(mar= c(4,4,2,1))  
dev.new()  
plot(R1[,1], type = "l", xlab = "", ylab = "");  
mtext("Iteration", side = 1, line = 2.5);  
mtext(expression(X[1]), side = 2, line = 2.5);  
  
dev.new()  
plot(R1[,2], type = "l", xlab = "", ylab = "");  
mtext("Iteration", side = 1, line = 2.5);  
mtext(expression(X[1]), side = 2, line = 2.5);  
  
dev.new()  
plot(R1[,1], R1[,2], main = "", xlab = "", ylab = "");  
mtext(expression(X[1]), side = 1, line = 2.5);  
mtext(expression(X[2]), side = 2, line = 2.5);  
  
dev.new()  
hist(R1[,1], freq = FALSE, xlab ="", main = ""); curve(dnorm(x, mean=mu[1], sd =s1), add = TRUE);  
mtext(expression(X[1]), side = 1, line = 2.5);  
  
dev.new()  
hist(R1[,2], freq = FALSE, xlab = "", main = ""); curve(dnorm(x, mean=mu[2], sd =s2), add = TRUE)  
mtext(expression(X[2]), side = 1, line = 2.5);  
dev.off()

## quartz\_off\_screen   
## 2

coda::effectiveSize(R1[,1])

## var1   
## 29

coda::effectiveSize(R1[,2])

## var1   
## 54.01054

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.