### Section 3.4 and 3.5 examples

###

#############################################

#############################################

## Section 3.4 examples:

# Sign test in R:

# Example 1 (with raw paired data)

scores.with.food <- c(74,71,82,77,72,81)

scores.without.food <- c(68,71,86,70,67,80)

# We can perform the sign test with the sign.test function:

sign.test<-function(x=0,y=NULL,alternative="two.sided"){

n<-sum((x-y)!=0)

T<-sum(x<y)

if (alternative=="less") {

p.value<-pbinom(T,n,0.5)}

if (alternative=="greater"){

p.value<- 1-pbinom(T-1,n,0.5)}

if (alternative=="two.sided"){

p.value<-2\*min(1-pbinom(T-1,n,0.5),pbinom(T,n,0.5))}

list(n=n,alternative=alternative,T=T,p.value=p.value)}

# Copy and paste this function into R, and then it can be implemented

# as in the following examples:

# When calculating the differences, be careful which variable is labeled "y" and which variable labeled "x":

sign.test(x=scores.with.food, y=scores.without.food, alternative="less")

# To test whether the population median of Y equals some constant.

# just put that constant in for x.

############################################################

# Example 2 (with counts given)

# We can perform the sign test with the sign.test.counts function:

sign.test.counts<-function(plus.count,minus.count,zero.count=0,alternative="two.sided"){

n<-plus.count+minus.count

T<-plus.count

if (alternative=="less") {

p.value<-pbinom(T,n,0.5)}

if (alternative=="greater"){

p.value<- 1-pbinom(T-1,n,0.5)}

if (alternative=="two.sided"){

p.value<-2\*min(1-pbinom(T-1,n,0.5),pbinom(T,n,0.5))}

list(n=n,alternative=alternative,T=T,p.value=p.value)}

# Using this function on the twin data set:

sign.test.counts(plus.count=10,minus.count=7,zero.count=1,alternative="two.sided")

############################################################

## Section 3.5 examples:

# The built-in function 'mcnemar.test' in R does the large-sample two-sided version of the test:

datamatrix<-matrix(c(90,12,8,90),ncol=2)

mcnemar.test(datamatrix, correct=FALSE)

# This gives T1 and a chi-squared based p-value.

# Note that R uses the large sample approximation to get the p-value.

# For the exact test, you could also use the binomial test of 0.5 for this example,

# where the number of (0,1) is 8 and number of (1,0) is 12.

binom.test(8,8+12,0.5)

############################################################

## Cox-Stuart test for trend:

global.temps <- c(-0.493, -0.457, -0.466, -0.497, -0.315, -0.077, 0.063, -0.036, -0.025, -0.002, 0.317, 0.563, 0.923)

n.prime <- length(global.temps)

c<-ifelse(n.prime/2 == round(n.prime/2), n.prime/2, (n.prime+1)/2)

early.values <- global.temps[1:(n.prime-c)]

later.values <- global.temps[(1+c):n.prime]

# Alternative is that y tends to be greater than x here:

sign.test(x=early.values, y=later.values, alternative="greater")

# Conclusion?