Multipopulation Statistical Test_anova with blocks

Weihao Lu

2018/4/7

Reference: Wang. J. L., Non-parametric statistical analysis,2006.4

Friedman test

page 110

```
square <- matrix(
    c(73,83,73,58,77,
      75,81,60,64,75,
      67,99,73,64,73,
      61,82,77,71,59,
      69,85,68,77,85,
      79,87,74,74,82),
    byrow=T,ncol = 5,
    dimnames = list(1:6,c("A","B","C","D","E"))
)
friedman.test(square)
```

```
##
##  Friedman rank sum test
##
## data:  square
## Friedman chi-squared = 14.609, df = 4, p-value = 0.005586
```

Hodges-Lehmann Test

```
hodges.lehmann.test=function (y, fixed.knot=FALSE)
{
    DNAME <- deparse(substitute(y))
    blocks <- factor(c(row(y)))

    k <- ncol(y)
    y <- matrix(unlist(split(c(y), blocks)), ncol = k, byrow = TRUE)
    y <- y[complete.cases(y), ]
    n <- nrow(y)
    y <- t(apply(y,1,function(x){x-mean(x)}))
    r <- matrix(rank(y),n,k)
    meanri=apply(r,2,mean)
    SSB=n*sum((meanri-(n*k+1)/2)^2)
    if(fixed.knot){
        knot=as.numeric(table(rank(y)[duplicated(rank(y))])+1)
        E.SSB=sum((t(apply(r,1,function(x){x-mean(x)})))^2)-sum(knot^3-knot)/12
    }else{
        E.SSB=sum((t(apply(r,1,function(x){x-mean(x)})))^2)
    }
    STATISTIC <- (k-1)*n*SSB/E.SSB
```

```
    PARAMETER <- k - 1
    PVAL <- pchisq(STATISTIC, PARAMETER, lower.tail = FALSE)
    names(STATISTIC) <- "Hodges-Lehmann Test statistic"
    names(PARAMETER) <- "df"
    structure(list(statistic = STATISTIC, parameter = PARAMETER,
                   p.value = PVAL, method = "Hodges-Lehmann Test", data.name = DNAME),
              class = "htest")

}
```

```
square <- matrix(
    c(73,83,73,58,77,
      75,81,60,64,75,
      67,99,73,64,73,
      61,82,77,71,59,
      69,85,68,77,85,
      79,87,74,74,82),
    byrow=T,ncol = 5,
    dimnames = list(1:6,c("A","B","C","D","E"))
)
hodges.lehmann.test(square)
```

```
##
##  Hodges-Lehmann Test
##
## data:  square
## Hodges-Lehmann Test statistic = 13.426, df = 4, p-value = 0.00937
```

```
hodges.lehmann.test(square, fixed.knot = TRUE)
```

```
##
##  Hodges-Lehmann Test
##
## data:  square
## Hodges-Lehmann Test statistic = 13.454, df = 4, p-value = 0.009259
```

Durbin Test

BIB design

```
durbin.test=function(y){
    DNAME <- deparse(substitute(y))
    blocks <- factor(c(row(y)))

    k=ncol(y)
    r=nrow(y)*mean(!is.na(y))
    t=k*mean(!is.na(y))

    if(!identical(as.numeric(apply(!is.na(y),2,sum)),rep(r,k))){
        stop("The number of times each process occurs is not the same.")
    }
    if(!identical(as.numeric(apply(!is.na(y),1,sum)),rep(t,k))){
        stop("Inconsistent number of processes appearing in each block.")
    }
```

```
    y <- matrix(unlist(split(c(y), blocks)), ncol = k, byrow = TRUE)
    s=apply(apply(y,1,rank,na.last="keep"),1,sum,na.rm=T)

    STATISTIC <- ((12*(k-1))/(r*k*(t^2-1)))*sum(s^2)-(((3*r*(k-1)*(t+1)))/(t-1))

    PARAMETER <- k - 1
    PVAL <- pchisq(STATISTIC, PARAMETER, lower.tail = FALSE)
    names(STATISTIC) <- "Durbin test statistic"
    names(PARAMETER) <- "df"
    structure(list(statistic = STATISTIC, parameter = PARAMETER,
                   p.value = PVAL, method = "Durbin Test", data.name = DNAME),
              class = "htest")
}
```

```
weight=matrix(
    c(73,NA,74,75,
      74,75,75,NA,
      NA,67,68,72,
      71,72,NA,75),
    byrow=T,ncol = 4,
    dimnames = list(1:4,c("A","B","C","D"))
)
durbin.test(weight)
```

```
##
##  Durbin Test
##
## data:  weight
## Durbin test statistic = 6.9375, df = 3, p-value = 0.07392
```

Page's Trend Test

When number of groups k=2, equivalent to Wilcoxon Signed Rank Test

```
page.trend.test=function (y){
    DNAME <- deparse(substitute(y))
    blocks <- factor(c(row(y)))

    k <- ncol(y)
    y <- matrix(unlist(split(c(y), blocks)), ncol = k, byrow = TRUE)
    y <- y[complete.cases(y), ]
    b <- nrow(y)
    r <- matrix(t(apply(y,1,rank)),b,k)
    P.stat=sum(1:k*apply(r,2,sum))
    EP.H0=b*k*(k+1)^2/4
    DP.H0=k^2*(k+1)^2*b*(k-1)/144

    PVAL <- pnorm((P.stat-EP.H0)/sqrt(DP.H0), lower.tail = F)
    names(P.stat) <- "Page's trend test statistic"
    structure(list(statistic = P.stat, p.value = PVAL,
                   method = "Page's Trend Test", data.name = DNAME),
              class = "htest")
}
```

```
dose=matrix(
    c(36,51,71,63,82,128,
      62,91,40,51,33,81,
      53,81,67,75,116,38,
      105,63,49,65,107,33,
      36,46,62,63,42,104,
      118,65,126,96,122,112,
      42,108,123,32,69,102,
      51,63,55,86,41,121,
      114,51,30,109,97,86),
    byrow=T,ncol = 6,
    dimnames = list(1:9, paste(1:6,"mg",sep=""))
)
page.trend.test(dose)
```

```
##
##  Page's Trend Test
##
## data:  dose
## Page's trend test statistic = 685, p-value = 0.1584
```

Cochran's Q Test

```
cochranQ.test=function (y){
    DNAME <- deparse(substitute(y))

    b=nrow(y)
    k=ncol(y)
    n.j=apply(y,2,sum)
    ni.=apply(y,1,sum)
    Q=k*(k-1)*sum((n.j-mean(n.j))^2)/(k*sum(n.j)-sum(ni.^2))
    p.value=pchisq(Q,k-1,low=F)
    names(Q) <- "Cochran's Q test statistic"
    structure(list(statistic = Q, p.value = p.value,
                   method = "Cochran's Q Test", data.name = DNAME),
              class = "htest")
}
```

```
brand=matrix(
    c(0,0,0,1,0,0,0,0,0,1,
      1,1,0,1,0,1,0,0,1,1,
      1,1,1,1,1,1,1,1,1,0),
    ncol = 3,
    dimnames = list(1:10, c("A","B","C"))
)
cochranQ.test(brand)
```

```
##
##  Cochran's Q Test
##
## data:  brand
## Cochran's Q test statistic = 8.2222, p-value = 0.01639
```