#------------------------------------------------------------------

# 昭和大学　第4回実践臨床統計学セミナー　Rハンズオン

# 対応のあるデータと3群以上の比較

# 2021年9月13日

#------------------------------------------------------------------

# Rのプログラム作成には、高機能なテキストエディタを使用することをお勧めします。

# 初心者の方向けのソフトとしては、Notepad++をお勧めします（フリーソフトです）。以下のURLからダウンロードすることができます。

#　https://github.com/notepad-plus-plus/notepad-plus-plus/releases/download/v7.9.5/npp.7.9.5.Installer.exe

# 拡張子を「.r」として保存したプログラムのファイルを開くと、基本関数やコメント箇所などが色分けされて表示され、効率的なプログラミングを行うことができます。

# Macユーザーの方は、Rのパッケージのインストールができないことがあるようです。

# RStudioを使えば、特に問題なく、インストールができることがあるようですので、もし支障があるようでしたら、そちらをお試しください（以下のURLからインストールパッケージをDLできます）。

# https://download1.rstudio.org/desktop/macos/RStudio-1.4.1717.dmg

#------------------------------------------------------------------

# 1. データセットの読み込み

#------------------------------------------------------------------

setwd("C:\\R2021")

# setwdで、カレントのディレクトリを指定します。

# 他のソフトウェアと違う点は、フォルダのパスの指定において、「\」を二重にしなくてはいけないという点です。

# 通常は、「C:\」で済みますが、「C:\\」という風に「\」を「\\」と表記しなくてはいけません。

# メニューバーの中から、「ファイル」→「ディレクトリの変更」としても、同様の操作を行うことができます。

intake <- read.csv(file="intake.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

intake

# データセットの中身を表示します。

#------------------------------------------------------------------

# 2. 対応のあるt検定

#------------------------------------------------------------------

attach(intake)

# attach関数を使うことで、データセットの変数名から、直接、変数のデータにアクセスすることができるようになります。

t.test(pre, post, paired=TRUE)

# t.test関数で、"paired=TRUE" という引数を加えることで、対応のあるt検定を行うことができます。

#------------------------------------------------------------------

# 3. 対応のあるWilcoxon検定

#------------------------------------------------------------------

wilcox.test(pre, post, paired=TRUE)

# wilcox.test関数で、"paired=TRUE" という引数を加えることで、対応のあるWilcoxon検定を行うことができます。

#------------------------------------------------------------------

# 4. データセットの読み込み

#------------------------------------------------------------------

ashina <- read.csv(file="ashina.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

ashina

# データセットの中身を表示します。

#------------------------------------------------------------------

# 5. 対応のあるt検定

#------------------------------------------------------------------

attach(ashina)

# attach関数を使うことで、データセットの変数名から、直接、変数のデータにアクセスすることができるようになります。

t.test(vas.plac, vas.active, paired=TRUE)

# t.test関数で、"paired=TRUE" という引数を加えることで、対応のあるt検定を行うことができます。

#------------------------------------------------------------------

# 6. データセットの読み込み

#------------------------------------------------------------------

cereb <- read.csv(file="cereb.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

cereb

# データセットの中身を表示します。

#------------------------------------------------------------------

# 7. McNamer検定

#------------------------------------------------------------------

attach(cereb)

# attach関数を使うことで、データセットの変数名から、直接、変数のデータにアクセスすることができるようになります。

tab1 <- table(y1,y0)

tab1 # 分割表の集計結果

mcnemar.test(tab1)

# mcnemar.test関数で、McNamer検定を行うことができます。

# "correct=FALSE" という引数を加えることで、連続修正を解除することができます（デフォルトでは、連続修正がかかります）。

#------------------------------------------------------------------

# 8. データセットの読み込み

#------------------------------------------------------------------

gold <- read.csv(file="gold.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

gold

# データセットの中身を表示します。

#------------------------------------------------------------------

# 9. McNamer検定

#------------------------------------------------------------------

attach(gold)

# attach関数を使うことで、データセットの変数名から、直接、変数のデータにアクセスすることができるようになります。

tab2 <- table(case,control)

tab2 # 分割表の集計結果

mcnemar.test(tab2)

# mcnemar.test関数で、McNamer検定を行うことができます。

# "correct=FALSE" という引数を加えることで、連続修正を解除することができます（デフォルトでは、連続修正がかかります）。

OR <- (9/2) # オッズ比の推定値

V <- (1/9 + 1/2) # 対数オッズ比の分散の推定値

OR # オッズ比の推定値

exp(log(OR) - 1.96\*sqrt(V)) # オッズ比の95%信頼区間の下限

exp(log(OR) + 1.96\*sqrt(V)) # オッズ比の95%信頼区間の上限

#------------------------------------------------------------------

# 10. データセットの読み込み

#------------------------------------------------------------------

gold2 <- read.csv(file="gold2.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

gold2

# データセットの中身を表示します。

#------------------------------------------------------------------

# 11. 条件付きロジスティック回帰分析

#------------------------------------------------------------------

library(survival) # ライブラリの読み込み

gm1 <- clogit(y ~ x + strata(pairs), data=gold2)

summary(gm1)

# clogit関数を使うことで、条件付きロジスティック回帰分析を行うことができます。

# McNamer検定は、1:1マッチングのときにしか使えませんが、条件付きロジスティック回帰は、任意のマッチング比に対して使うことができます。

# また、マッチングをとっていない要因を説明変数に加えることで、多変量モデルによる交絡調整を行うことも可能です。

# マッチングの組（対応のあるデータの組）を表す変数を、strata() で指定する必要があります。

#------------------------------------------------------------------

# 12. データセットの読み込み

#------------------------------------------------------------------

rcf <- read.csv(file="red.cell.folate.csv")

# CSV形式のデータセットを読み込むためには、read.csvという関数を使います。

# fileという引数で、ファイル名を指定します。カレントのディレクトリにファイル名が一致するCSVファイルがあれば、読み込まれます。

rcf

# データセットの中身を表示します。

#------------------------------------------------------------------

# 13. 一元配置分散分析

#------------------------------------------------------------------

attach(rcf)

# attach関数を使うことで、データセットの変数名から、直接、変数のデータにアクセスすることができるようになります。

boxplot(folate ~ ventilation)

# 箱ひげ図の作図

anova(lm(folate ~ ventilation))

# 一元配置分散分析。lmという関数の中に、「アウトカム変数（左辺） ~ グループを表す変数（右辺）」という構文で、変数の指定を行います。

# これを、anova関数にかけることで、一元配置分散分析の結果を得ることができます。

#------------------------------------------------------------------

# 14. Kruskal-Wallis検定

#------------------------------------------------------------------

kruskal.test(folate ~ ventilation)

# Kruskal-Wallis検定は、kruskal.test関数で行うことができます。

#------------------------------------------------------------------

# 15. 3群以上のカテゴリカルデータの比較

#------------------------------------------------------------------

actg175 <- matrix(c(351,419,415,433,181,103,109,128), 4, 2)

chisq.test(actg175)

# 4群比較のカイ二乗検定。chisq.testで行うことができます。

fisher.test(actg175)

# 4群比較のFisherの正確検定。fisher.testで行うことができます。