DA410_Assignment6_MattGraham

Finding canonical correlationss

```
library(nnspat)  # used for dist2full()
library("dplyr")  # used to select numeric datatypes
library("ggplot2")
library(reshape)  # used for melting matricies
library(klaR)
library(ggvis)
library(class)
library(gmodels)
library(MASS)
library(readxl)
library(psych)
```

Determines set of canonical varirates that are orthogonal linear combinations that best explain variabilities within and between sets. It is common to have multiple canonical variables.

11.9

a

Get data

```
sons <- read.table("C:/mattgraham93.github.io/school/22_3_DA410/data/T3_8_SONS.DAT", header=F
ALSE)
colnames(sons) <- c('fs_hl', 'fs_hb', 'ss_hl', 'ss_hb')  ## hl = head length, hb = head bread
th, fs = first son, ss = second son
sons
```

fs_hl <int>	fs_hb <int>	ss_hl <int>	ss_hb <int>
191	155	179	145
195	149	201	152
181	148	185	149
183	153	188	149
176	144	171	142
208	157	192	152
189	150	190	149
197	159	189	152

fs_hl <int>	fs_hb <int>	ss_hl <int>	ss_hb <int>
188	152	197	159
192	150	187	151

1-10 of 25 rows Previous **1** 2 3 Next

Standardize variables and assign measure groups

```
sons.std <- sweep(sons, 2, sqrt(apply(sons, 2, var)), FUN="/")

y.meas <- sons.std[, 1:2] ## first son
x.meas <- sons.std[, 3:4] ## second son

sons.std
```

fs_hl <dbl>	fs_hb <dbl>	ss_hl <dbl>	ss_hb <dbl>
19.56600	21.02287	17.82824	21.60972
19.97576	20.20908	20.01942	22.65295
18.54160	20.07345	18.42583	22.20585
18.74648	20.75161	18.72463	22.20585
18.02941	19.53092	17.03144	21.16262
21.30748	21.29413	19.12303	22.65295
19.36112	20.34471	18.92383	22.20585
20.18064	21.56539	18.82423	22.65295
19.25868	20.61597	19.62102	23.69618
19.66844	20.34471	18.62503	22.50392

1-10 of 25 rows Previous **1** 2 3 Next

Find blocks of correlation matrix

```
R11 <- cor(x.meas)
R22 <- cor(y.meas)
R12 <- c(cor(y.meas[, 1], x.meas[, 1]),
         cor(y.meas[, 1], x.meas[, 2]),
         cor(y.meas[, 2], x.meas[, 1]),
         cor(y.meas[, 2], x.meas[, 2])
         )
R21 <- matrix(R12, ncol=ncol(R22), byrow=T)
```

Display matricies

```
R11
```

```
##           ss_hl     ss_hb
## ss_hl 1.0000000 0.8392519
## ss_hb 0.8392519 1.0000000
```

```
R22
```

```
##           fs_hl     fs_hb
## fs_hl 1.0000000 0.7345555
## fs_hb 0.7345555 1.0000000
```

```
### splitting the list to make a matrix for multiplication
r12.fs <- R12[1:2]
r12.ss <- R12[3:4]

R12 <- rbind(r12.fs, r12.ss)
R12
```

```
##             [,1]      [,2]
## r12.fs 0.7107518 0.7039807
## r12.ss 0.6931573 0.7085504
```

```
R21
```

```
##           [,1]      [,2]
## [1,] 0.7107518 0.7039807
## [2,] 0.6931573 0.7085504
```

Finding E1 and E2 matricies

```
E1 <- solve(R11) %*% R12 %*% solve(R22) %*% R21
E1
```

```
##            [,1]      [,2]
## ss_hl 0.3284513 0.3276745
## ss_hb 0.2910913 0.2958697
```

E2

```
E2 <- solve(R22) %*% R21 %*% solve(R11) %*% R12
E2
```

```
##             [,1]      [,2]
## fs_hl 0.3213612 0.3206147
## fs_hb 0.2980647 0.3029597
```

Find eigenvalues

```
eigen(E1)
```

```
## eigen() decomposition
## $values
## [1] 0.621431559 0.002889411
## 
## $vectors
##            [,1]       [,2]
## [1,] 0.7454699 -0.7093899
## [2,] 0.6665393  0.7048162
```

```
eigen(E2)
```

```
## eigen() decomposition
## $values
## [1] 0.621431559 0.002889411
## 
## $vectors
##            [,1]       [,2]
## [1,] 0.7301119 -0.7094738
## [2,] 0.6833276  0.7047318
```

Finding canonical correlation

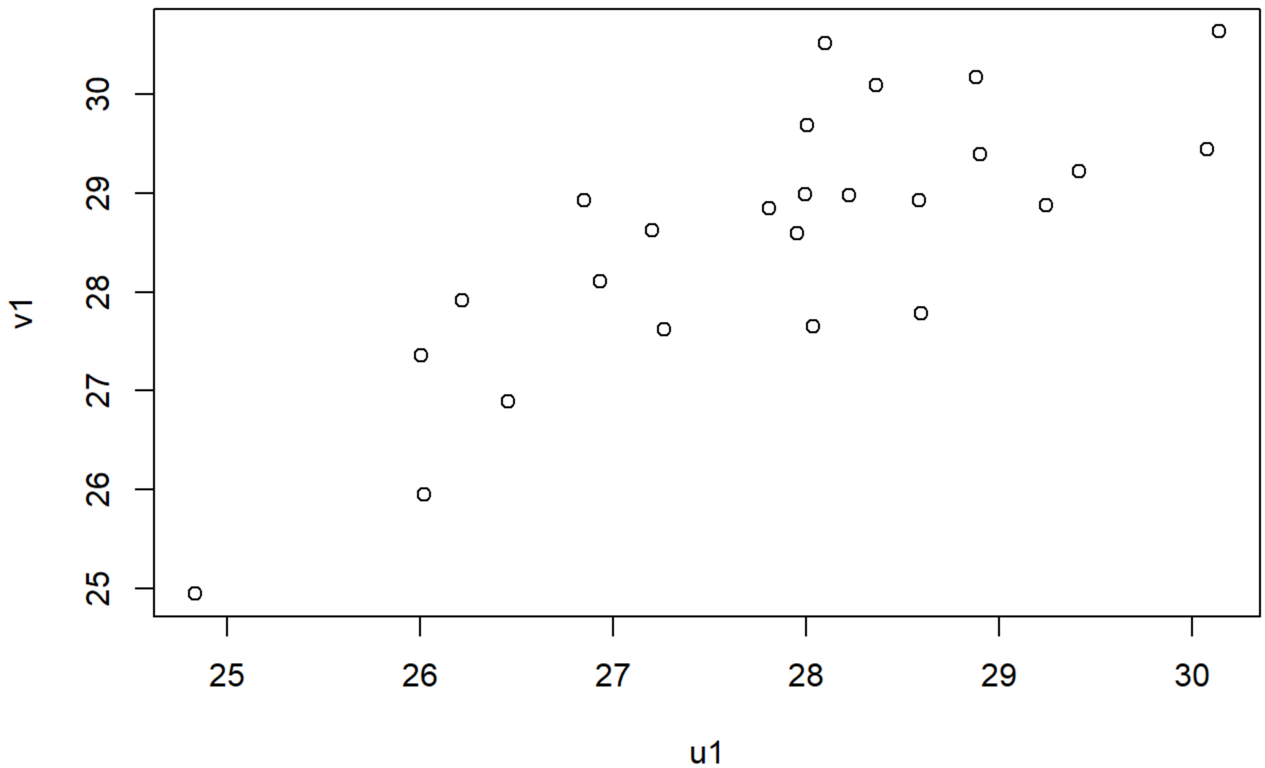
```
can.cor <- sqrt(eigen(E1)$values)
can.cor
```

```
## [1] 0.78830930 0.05375324
```

Find and plot characteristic equation

```
# slope1 <- 1
# slope2 <- 1
#
# u1 <- (slope1 * y.meas$fs_hl) + (slope2 * y.meas$fs_hb)

u1 <- as.matrix(sons.std[, 1:2]) %*% as.matrix(eigen(E1)$vectors[,1])
v1 <- as.matrix(sons.std[, 3:4]) %*% as.matrix(eigen(E2)$vectors[,1])
plot(u1, v1)
```



Correlation for u1, v1

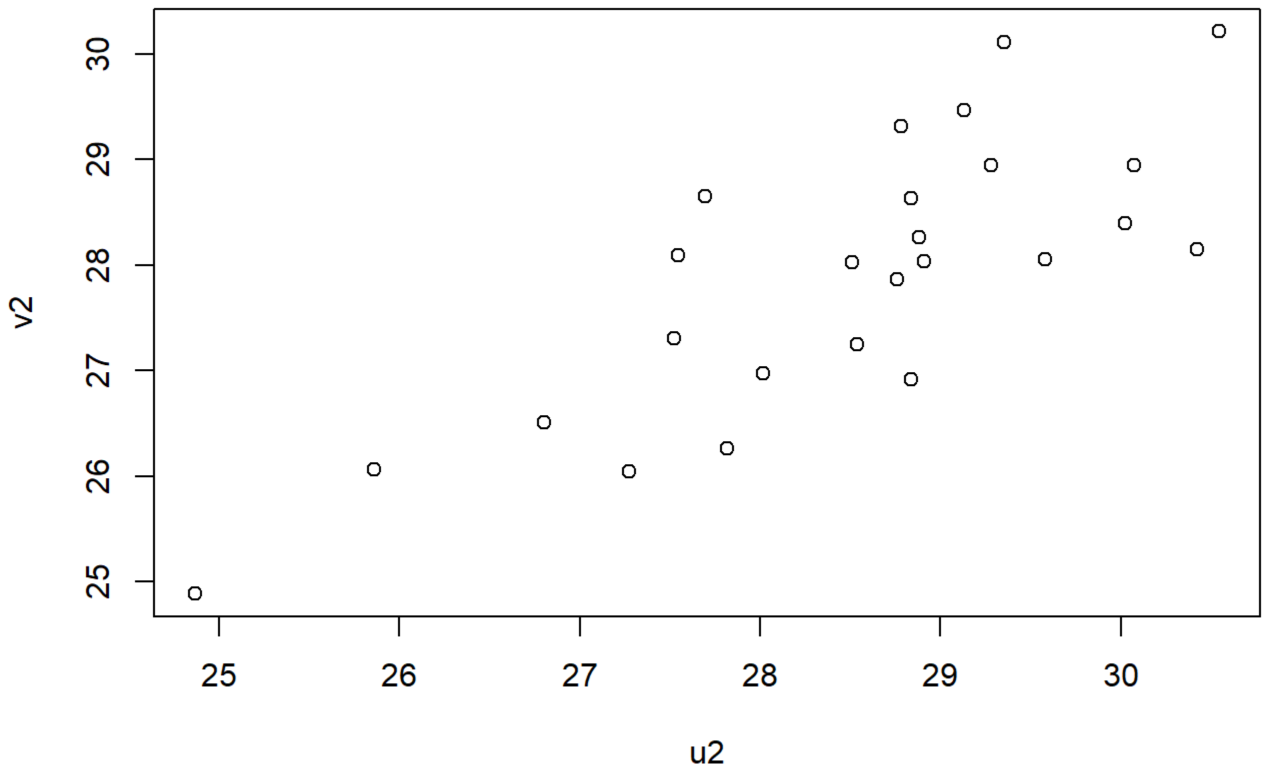
```
cor(u1, v1)
```

```
##           [,1]
## [1,] 0.7883259
```

Interpretation

For every shift in head length, head breadth decreasess when all other variables are held constant.

```
u2 <- as.matrix(sons.std[, 3:4]) %*% as.matrix(eigen(E1)$vectors[,1])
v2 <- as.matrix(sons.std[, 1:2]) %*% as.matrix(eigen(E2)$vectors[,1])
plot(u2, v2)
```



Correlation for u2, v2

```
cor(u2, v2)
```

```
##           [,1]
## [1,] 0.7882402
```

Interpretation:

For every shift in head breadth, head length decreases when all other variables are held constant.

C

Ho: fs_ss_lth = O, independence between head length between first and second son *Ha:* fs_ss_lth != O, dependence between head length between first and second son

Calculating F stat

```
k = 2
p = 2
q = 2
n = 25

## formula on page 394
w = n - ( (1/2) * (p + q + 3) )
df1 = (p - k + 1)* (q - k + 1)
t.1 = ( ( (p - k + 1)**2 * (q - k + 1)**2) - 4) /
      ( ( (p - k + 1)**2 + (q - k + 1)**2) - 5)
df2 =(w *  t.1) - ( (1/2) * ( (p - k + 1) * (q - k + 1) + 1) )
F.1 = ( (1 - cor(u1, u2)**(1/t.1)) * df2) / ( (cor(u1, u2)**(1/t.1) ) * df1)
F.1
```

```
##          [,1]
## [1,] 5.508333
```

Find rejection region

```
qf(.05, df1, df2, lower.tail=FALSE)
```

```
## [1] 4.337667
```

Interpretation

Given our rejection region = 4.072654, and our calculated statistic = 5.508333, there is enough evidence to conclude that there is dependence between the head length between the first and second born sons!

Ho: fs_ss_bth = O, independence between head breadth between first and second son *Ha:* fs_ss_bth != O, dependence between head breadth between first and second son

Calculating F stat

```
## use same k, p, q, n, df1, and df2 as above
t.2 = ( ( (p - k + 1)**2 * (q - k + 1)**2) - 4) /
      ( ( (p - k + 1)**2 + (q - k + 1)**2) - 5)
df2 =(w *  t.2) - ( (1/2) * ( (p - k + 1) * (q - k + 1) + 1) )
F.2 = ( (1 - cor(v1, v2)**(1/t.2)) * df2) / ( (cor(v1, v2)**(1/t.2) ) * df1)
F.2
```

```
##          [,1]
## [1,] 5.503333
```

```
# (repeat for 1 and 2)
```

Interpretation

Given our rejection region = 4.072654, and our calculated statistic = 5.503333, there is enough evidence to conclude that there is dependence between the head breadth between the first and second born sons!

Discussion example

```r
library(CCA)
```

```
## Warning: package 'CCA' was built under R version 4.2.2
```

```
## Loading required package: fda
```

```
## Warning: package 'fda' was built under R version 4.2.2
```

```
## Loading required package: splines
```

```
## Loading required package: fds
```

```
## Warning: package 'fds' was built under R version 4.2.2
```

```
## Loading required package: rainbow
```

```
## Warning: package 'rainbow' was built under R version 4.2.2
```

```
## Loading required package: pcaPP
```

```
## Warning: package 'pcaPP' was built under R version 4.2.2
```

```
## Loading required package: RCurl
```

```
## Loading required package: deSolve
```

```
## Warning: package 'deSolve' was built under R version 4.2.2
```

```
##
## Attaching package: 'fda'
```

```
## The following object is masked from 'package:graphics':
##
##     matplot
```

```
## Loading required package: fields
```

```
## Warning: package 'fields' was built under R version 4.2.2
```

```
## Loading required package: spam
```

```
## Warning: package 'spam' was built under R version 4.2.2
```

```
## Spam version 2.9-1 (2022-08-07) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

```
## Loading required package: viridis
```

```
## Warning: package 'viridis' was built under R version 4.2.2
```

```
## Loading required package: viridisLite
```

```
##
## Try help(fields) to get started.
```

```
##
## Attaching package: 'fields'
```

```
## The following object is masked from 'package:psych':
##
##     describe
```

```
model <- cc(x.meas, y.meas)
model
```

```
## $cor
## [1] 0.7885079 0.0537397
##
## $names
## $names$Xnames
## [1] "ss_hl" "ss_hb"
##
## $names$Ynames
## [1] "fs_hl" "fs_hb"
##
## $names$ind.names
##  [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14" "15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25"
##
##
## $xcoef
##             [,1]      [,2]
## ss_hl -0.5044484 -1.768570
## ss_hb -0.5382877  1.758566
##
## $ycoef
##             [,1]      [,2]
## fs_hl -0.5521896 -1.366374
## fs_hb -0.5215372  1.378365
##
## $scores
## $scores$xscores
##               [,1]        [,2]
##  [1,]  0.583317147 -0.25867829
##  [2,] -1.083576808 -2.29934799
##  [3,] -0.039028038 -0.26723166
##  [4,] -0.189755833 -0.79567548
##  [5,]  1.225925130  0.36425453
##  [6,] -0.631393423 -0.71401655
##  [7,] -0.290241030 -1.14797136
##  [8,] -0.480665628 -0.18557273
##  [9,] -1.444163206  0.23982870
## [10,] -0.299958033 -0.09536041
## [11,] -0.009048238 -0.70546317
## [12,]  0.674085341  1.14622854
## [13,] -0.279695235  0.51901903
## [14,] -1.183233212  0.06795746
## [15,] -0.861514824  1.73922453
## [16,]  2.691019899 -1.01926884
## [17,] -0.660544431  2.44381628
## [18,]  0.644105541  1.58446004
## [19,]  0.352366953 -0.52503854
## [20,]  1.928492714  0.11072435
## [21,] -0.279695235  0.51901903
## [22,]  0.473114948  0.44163678
## [23,]  0.894489740 -0.25440160
```

```
## [24,] -1.514668603 -0.55069868
## [25,] -0.219735634 -0.35744398
##
## $scores$yscores
##              [,1]        [,2]
##  [1,] -0.57312842 -0.01368291
##  [2,] -0.37497221 -1.69526490
##  [3,]  0.48769136  0.07738080
##  [4,]  0.02087481  0.73218662
##  [5,]  1.05346966  0.02943785
##  [6,] -1.67622740 -2.01929228
##  [7,] -0.10631187 -0.66848874
##  [8,] -1.19547291 -0.10571105
##  [9,] -0.19121934 -0.15461844
## [10,] -0.27601046 -1.08840202
## [11,] -0.10654457  2.22681901
## [12,]  0.44529580 -0.38951099
## [13,]  0.74218106  1.43107765
## [14,] -0.79950955  0.87408659
## [15,] -0.12048251 -0.34156804
## [16,]  2.28398802  0.54041483
## [17,] -0.79939320 -0.57356728
## [18,] -0.14882378  0.31227335
## [19,]  0.69990185 -0.48346801
## [20,]  1.39298318 -0.57838947
## [21,] -0.55895778 -0.34060361
## [22,]  1.23733888  0.12243043
## [23,]  1.40715381 -0.90531017
## [24,] -1.76136757  1.38988577
## [25,] -1.08245687  1.62188500
##
## $scores$corr.X.xscores
##             [,1]       [,2]
## ss_hl -0.9562074 -0.2926900
## ss_hb -0.9616470  0.2742901
##
## $scores$corr.Y.xscores
##             [,1]        [,2]
## fs_hl -0.7374817 -0.01901786
## fs_hb -0.7310660  0.02013559
##
## $scores$corr.X.yscores
##             [,1]        [,2]
## ss_hl -0.7539771 -0.01572908
## ss_hb -0.7582663  0.01474027
##
## $scores$corr.Y.yscores
##             [,1]       [,2]
## fs_hl -0.9352877 -0.3538884
## fs_hb -0.9271512  0.3746875
```

```
plt.cc(model, var.label = TRUE)
```