Displays for Statistics 5401

Lecture 32

November 21, 2005

Christopher Bingham, Instructor
612-625-1024

Class Web Page

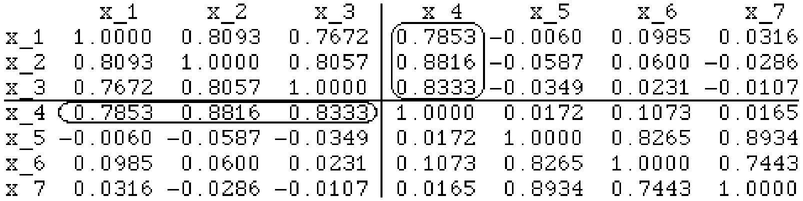
`http://www.stat.umn.edu/~kb/classes/5401`

Example with artificial data

```
Cmd> scores <- read("","scores") # read from fake_scores.txt
scores       100     7 format labels
) Artificial data representing scores of 100 college students on
) 7 standardized tests.
) Col. 1: x1 logical word problems                  Var 1 in set 1
) Col. 2: x2 pattern recognition                    Var 2 in set 1
) Col. 3: x3 graph interpretation skills            Var 3 in set 1
) Col. 4: x4 high school algebra                    Var 1 in set 2
) Col. 5: x5 role playing aptitude                  Var 2 in set 2
) Col. 6: x6 decision making under stress           Var 3 in set 2
) Col. 7: x7 ability to find disguised objects in pictures V4 ,2
Read from file "TP1:Stat5401:Data:fake_scores.txt"
```

Natural groups of variables would be x_1-x_4 and x_5-x_7. However, I will put x_1-x_3 in $\mathbf{X}^{(1)}$ and x_4-x_7 in $\mathbf{X}^{(2)}$.

```
Cmd> r <- cor(scores); print(format:"7.4f",r) # Corr matrix
r:
```



```
         x_1      x_2      x_3      x_4      x_5      x_6      x_7
x_1   1.0000   0.8093   0.7672   0.7853  -0.0060   0.0985   0.0316
x_2   0.8093   1.0000   0.8057   0.8816  -0.0587   0.0600  -0.0286
x_3   0.7672   0.8057   1.0000   0.8333  -0.0349   0.0231  -0.0107
x_4   0.7853   0.8816   0.8333   1.0000   0.0172   0.1073   0.0165
x_5  -0.0060  -0.0587  -0.0349   0.0172   1.0000   0.8265   0.8934
x_6   0.0985   0.0600   0.0231   0.1073   0.8265   1.0000   0.7443
x 7   0.0316  -0.0286  -0.0107   0.0165   0.8934   0.7443   1.0000
```

```
Cmd> p <- 3; q <- 4 # vars 1-3 in one set, 4-7 in the other

Cmd> N <- nrows(scores); fe <- N - 1

Cmd> J1 <- run(p)  # selector variable for variables 1-3

Cmd> J2 <- run(p+1,p+q) # selector for variables 4-7
```

I use `J1` and `J2` as "selector variables" (subscripts) to select info related to $\mathbf{X}^{(1)}$ = `scores[,J1]` or $\mathbf{X}^{(2)}$ = `scores[,J2]`.

Bonferronized tests of correlation coefficients

Transform correlations r_{ij} to t-statistics as

$$t_{ij} = \frac{(\sqrt{\{f_e - 1\}})r_{ij}}{\sqrt{\{1 - r_{ij}^2\}}}$$

```
Cmd> tstats <- sqrt(fe-1)*r[J1,J2]/sqrt(1 - r[J1,J2]^2)

Cmd> tstats
             x_4         x_5          x_6         x_7
x_1       12.555   -0.059804      0.98003     0.31315
x_2       18.492    -0.58258      0.59499    -0.28332
x_3       14.923    -0.34521      0.22844     -0.1062

Cmd> p*q # There are 12 correlations
(1)          12

Cmd> tcrit <- invstu(.025/(p*q),fe-1,upper:T)#Bonferronize

Cmd> tcrit # t-critical value Bonferronized by p*q = 12
(1)       2.9341   Bonferroni critical value, alpha=.05

Cmd> p*q*twotailt(tstats,fe-1) # Bonf. Pvalues
(1,1)  4.8283e-21       11.429       3.9538        9.058
(2,1)  1.2039e-32       6.7381       6.6386       9.3304
(3,1)    6.82e-26       8.7681       9.8373       10.988
```

There is strong evidence $\rho_{12} \neq \mathbf{0}$.

Moreover, there is strong evidence that $\rho_{11}^{12} \neq 0$, $\rho_{21}^{12} \neq 0$ and $\rho_{31}^{12} \neq 0$.

Regression-based tests

You can test either

$$H_0: \boldsymbol{\beta}_{2.1} = \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1} = \mathbf{0} \qquad q \text{ by } p$$

or

$$H_0: \boldsymbol{\beta}_{1.2} = \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1} = \mathbf{0} \qquad p \text{ by } q$$

This is the standard *multivariate* linear model situation.

If you use, say, **manova()**, to compute the multivariate regression of $\mathbf{x}^{(2)}$ on $\mathbf{x}^{(1)}$, then

- Response dimension = q
- $\mathbf{H} = f_e\mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$, $q \times q$, $f_e = n - 1$
- $\mathbf{E} = f_e\mathbf{S}_{22.1}$, $q \times q$, $f_e = n - 1$
  $\mathbf{S}_{22.1} \equiv \mathbf{S}_{22} - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$
- Hypothesis DF = $\tilde{f}_h = p$
- Error DF = $\tilde{f}_e = f_e - p = n - 1 - p$

```
Cmd> x1 <- scores[,J1]; x2 <- scores[,J2]

Cmd> makecols(x1,x1_1,x1_2,x1_3)#split up x1 into columns
Column 1 saved as vector x1_1
Column 2 saved as vector x1_2
Column 3 saved as vector x1_3

Cmd> manova("x2 = x1_1 + x1_2 + x1_3",silent:T)
```

```
Cmd> list(SS)
SS              REAL    5     4     4     (labels)   5 by q by q
```

```
Cmd> TERMNAMES # names associated with each SS[j,,]
(1) "CONSTANT"
(2) "x1_1"                   5 terms counting CONSTANT and
(3) "x1_2"                   ERROR1;
(4) "x1_3"                   regression terms are 2, 3, 4
(5) "ERROR1"
```

```
Cmd> h2 <- matrix(sum(SS[run(2,4),,])); h2
            x_4         x_5         x_6         x_7
x_4      3205.9     -169.08      236.97     -66.032
x_5     -169.08      32.657      12.605      34.727
x_6      236.97      12.605      95.049      35.885
x_7     -66.032      34.727      35.885      43.652
```

`h2` is regression **H** from `manova()`.

MacAnova Note:
`sum()`sums over first dimension so `sum(SS[run(2,4),,])` is `SS[2,,]+SS[3,,]+ SS[4,,]` = regression SSCP matrix.

```
Cmd> e2 <- matrix(SS[5,,]); e2  # E from manova()
            x_4         x_5         x_6         x_7
x_4      686.85      235.82      271.73      135.25
x_5      235.82      3831.9      3890.9      3705.3
x_6      271.73      3890.9      5677.7      3772.3
x_7      135.25      3705.3      3772.3      4490.9
```

```
Cmd> releigenvals(h2,e2)
(1)       4.9149    0.030929   0.0090215  1.2698e-15
```

These are relative eigenvalues in regression of $\mathbf{X}^{(2)}$ on $\mathbf{X}^{(1)}$.

Find H and E from variance matrix S

```
Cmd> s <- tabs(scores,covar:T) # variance matrix

Cmd> setlabels(s,structure(getlabels(scores,2),\
        getlabels(scores,2))) # make things pretty

Cmd> s11 <- s[J1,J1]

Cmd> s12 <- s[J1,J2]

Cmd> s22 <- s[J2,J2]

Cmd> h2a <- fe*s12' %*% (s11 %\% s12) # H from manova()

Cmd> s22dot1 <- s22 - s12' %*% (s11 %\% s12)

Cmd> e2a <- fe* s22dot1 # = E from manova()

Cmd> h2a # Same as H from manova()
            x_4         x_5         x_6         x_7
x_4      3205.9     -169.08      236.97     -66.032
x_5     -169.08      32.657      12.605      34.727
x_6      236.97      12.605      95.049      35.885
x_7     -66.032      34.727      35.885      43.652

Cmd> e2a # Same as E from manova()
            x_4         x_5         x_6         x_7
x_4      686.85      235.82      271.73      135.25
x_5      235.82      3831.9      3890.9      3705.3
x_6      271.73      3890.9      5677.7      3772.3
x_7      135.25      3705.3      3772.3      4490.9
```

You can get relative eigenvalues directly from the pieces of **S**

```
Cmd> releigenvals(s12' %*% (s11 %\% s12), s22dot1)
(1)       4.9149    0.030929   0.0090215 -2.9477e-17
```

These are the eigenvalues of $\mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$ relative to $\mathbf{S}_{22.1}$ and are the same as eigenvalues of **H** relative to **E**

All the multivariate linear model tests are available. These include:

- Bonferronized (by q) F-statistics
- Bonferronized (by p×q) t-statistics, one for each regression coefficient.
- Roy's maximum root test
- Hotelling's trace test (T_0^2)
- Wilks' test (likelihood ratio)
- Pillai's trace test.

Note: For tests based on relative eigenvalues

$$s = \min(\tilde{f}_h, \text{dimension}) = \min(p,q)$$

$$m = (|\tilde{f}_h - \text{dimension}| - 1)/2 = (|p - q| - 1)/2$$

$$n = (\tilde{f}_e - \text{dimension} - 1)/2 = (f_e - p - q - 1)/2 = (N - p - q - 2)/2$$

These are *symmetric* in p and q; you can swap the values of p and q without changing s, m and n.

```
Cmd> fe2 <- fe - p; fh2 <- p

Cmd> vector(fh2,fe2)
(1)          3          96

Cmd> fstats <- diag(h2/fh2)/diag(e2/fe2); fstats
(1)     149.36    0.27272    0.53571    0.31104

Cmd> q*cumF(fstats,fh2,fe2,upper:T) # Bonferronized P values
(1)  1.969e-35     3.3797     2.6357     3.2694
```

This shows clearly that x_4 depends strongly on $\mathbf{x}^{(1)}$, but not x_5. x_6 or x_7.

```
Cmd> eigs2 <- releigen(h2,e2)

Cmd> eigs2$values # very dominant first dimension
(1)     4.9149    0.030929    0.0090215  1.2698e-15

Cmd> sd2 <- sqrt(diag(s22)) # residual standard deviations

Cmd> u2 <- eigs2$vectors[,-4]*sd2; u2/max(abs(u2))
             (1)        (2)        (3)
x_4            1   -0.10645  -0.037148
x_5     -0.21979         -1  -0.014591
x_6     0.036036    0.95776    0.28063
x_7      0.13381    0.27312         -1
```

$\hat{\lambda}_1$ dominates and the first canonical variable weights most heavily on x_4.

Roy's test

```
Cmd> s <- min(vector(q,fh2)) # or min(p,q)

Cmd> m <- (abs(p-q)-1)/2; n <- (N-p-q-2)/2

Cmd> vector(s,m,n) # use these to get critical value from chart
(1)          3          0       45.5

Cmd> thetamax <- eigs2$values[1]/(1+eigs2$values[1]);thetamax
(1)    0.83093    Maximum value of theta = 1/(1+lambda)

Cmd> # This exceeds 0.186, the 1% point from the s = 3 chart
```

Hotelling's trace test $\sum\hat{\lambda}_i$

```
Cmd> cumtrace(sum(eigs2$values),fh2,fe2,q,upper:T)
(1) 1.8554e-86    Extremely small Pvalue
```

Wilks (likelihood ratio) test $1/\prod(1+\hat{\lambda}_i)$

```
Cmd> cumwilks(1/prod(1+eigs2$values),fh2,fe2,q)
(1) 1.5447e-30    Extremely small Pvalue
```

Both P-values are extremely small, again leading to rejection of H_0.

F-statistics and relative eigenvalues are the same when computed from

- $\tilde{H} \equiv \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$
- $\tilde{E} = \mathbf{S}_{22.1} = \mathbf{S}_{22} - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$

```
Cmd> s22dot1 <- s22 - s12' %*% solve(s11) %*% s12

Cmd> releigenvals(s12' %*% solve(s11) %*% s12, s22dot1)
(1)     4.9149   0.030929  0.0090215 1.4957e-16

Cmd> thetas <- eigs2$values/(1 + eigs2$values); thetas
(1)    0.83093   0.030001  0.0089408 1.2698e-15

Cmd> releigenvals(s12' %*% solve(s11) %*% s12,s22)
(1)    0.83093   0.030001  0.0089408 6.5688e-18
```

Now do the regression of x1 on x2.

```
Cmd> makecols(x2,x2_1,x2_2,x2_3,x2_4)
Column 1 saved as vector x2_1
Column 2 saved as vector x2_2
Column 3 saved as vector x2_3
Column 4 saved as vector x2_4

Cmd> manova("x1=x2_1+x2_2+x2_3+x2_4",silent:T)

Cmd> h1 <- matrix(sum(SS[run(2,q+1),,]))

Cmd> e1 <- matrix(SS[6,,])

Cmd> fh1 <- q;fe1 <- N - 1 - fh1;vector(p, fh1,fe1)
(1)          3          4         95

Cmd> eigs1 <- releigen(h1,e1)

Cmd> eigs1$values # same non-zero eigenvalues as before
(1)     4.9149    0.030929   0.0090215
```

Fact: The non-zero relative eigenvalues from the $\mathbf{x}^{(1)}$ on $\mathbf{x}^{(2)}$ regression are the same as from the $\mathbf{x}^{(2)}$ on $\mathbf{x}^{(1)}$ regression.

```
Cmd> sd1 <- sqrt(diag(s11)) # standard deviations

Cmd> u1 <- eigs1$vectors * sd1; u1/max(abs(u1))
             (1)          (2)          (3)
x_1      0.18701     -0.57638      0.88276
x_2            1      -0.4284           -1
x_3      0.55634            1      0.22803
```

This provides very little information on the structure of the association.

```
Cmd> fstats1 <- diag(h1/fh1)/diag(e1/fe1); fstats1
(1)      39.801       87.995      55.671

Cmd> p*cumF(fstats1,fh1,fe1,upper:T) # Bonferronized P-values
(1) 4.5776e-19 1.3022e-30 1.2866e-23
```

The univariate F's all reject H_0.

Population Canonical Correlations

The goal of canonical correlation to understand the structure of correlations between p variables in $\mathbf{x}^{(1)}$ and q variables in $\mathbf{x}^{(2)}$.

There are least two approaches.

1. Find a low rank approximation to a matrix summarizing correlations between $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$.
2. Find *features* or *summaries* $g(\mathbf{x}^{(1)})$ and $h(\mathbf{x}^{(2)})$ of $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ such that $\rho^2[g(\mathbf{x}^{(1)}), h(\mathbf{x}^{(2)})]$ is as large as possible.

You know best how to work with **linear** features, that is, where $g(\mathbf{x}^{(1)}) = \mathbf{u}'\mathbf{x}^{(1)}$ and $h(\mathbf{x}^{(2)}) = \mathbf{v}'\mathbf{x}^{(2)}$ are *linear combinations*.

Traditional canonical correlation finds $\mathbf{u}$ and $\mathbf{v}$ such that $\rho^2[\mathbf{u}'\mathbf{x}^{(1)}, \mathbf{v}'\mathbf{x}^{(2)}]$ is as large as possible.

Finding an approximation a matrix summarizing correlations.

Recall we are trying to characterize the correlations between two sets of variables, $\mathbf{x}^{(1)}$ with p variables and $\mathbf{x}^{(2)}$ with q variables.

Correlations are easiest to understand with standarized data. This suggests trying to find a simpler matrix that approximates

$$\tilde{\boldsymbol{\rho}}_{12} = \text{corr}[\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)}],$$

where $\tilde{\mathbf{x}}^{(1)}$ and $\tilde{\mathbf{x}}^{(2)}$ are *multistandardized* versions of $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$, that is

- $E[\tilde{\mathbf{x}}^{(1)}] = \mathbf{0}_p$, $V[\tilde{\mathbf{x}}^{(1)}] = \mathbf{I}_p$
- $E[\tilde{\mathbf{x}}^{(2)}] = \mathbf{0}_q$, $V[\tilde{\mathbf{x}}^{(2)}] = \mathbf{I}_q$

The zero means are not important; the identity matrix variances is important.

Notation:

- $\mathbf{A}^T$ is another notation for $\mathbf{A}'$.

When $\mathbf{A}$ is a positive semi-definite symmetric matrix, you can find a "square root" $\mathbf{A}^{1/2}$ of $\mathbf{A}$ in many ways.

- $\mathbf{A}^{1/2}$ is any matrix such that
$$(\mathbf{A}^{1/2})^T\mathbf{A}^{1/2} = \mathbf{A}$$
- $\mathbf{A}^{T/2}$ is shorthand for $(\mathbf{A}^{1/2})^T$
$$\mathbf{A}^{T/2}\mathbf{A}^{1/2} = \mathbf{A}$$
- $\mathbf{A}^{-1/2} = (\mathbf{A}^{1/2})^{-1}$
- $\mathbf{A}^{-T/2} \equiv (\mathbf{A}^{T/2})^{-1} = (\mathbf{A}^{-1/2})^T \Rightarrow \mathbf{A}^{-1/2}\mathbf{A}^{-T/2} = \mathbf{A}^{-1}$

When $\mathbf{A}^{1/2}$ is upper triangular, $\mathbf{A}^{T/2}$ is lower triangular and $(\mathbf{A}^{1/2})^T\mathbf{A}^{1/2}$ is the Cholesky decomposition of $\mathbf{A}$ and you can compute $\mathbf{A}^{1/2}$ using MacAnova function `cholesky()`.

Recall the partitions of $\mathbf{\Sigma}$ and $\mathbf{x}$:

$$\mathbf{\Sigma} = \begin{array}{cc} \left[\begin{array}{cc} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{array}\right] & \begin{array}{c} p \\ q \end{array} \\ \begin{array}{cc} p & q \end{array} & \end{array}, \quad \mathbf{x} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{bmatrix}$$

Define

- $\tilde{\mathbf{x}}^{(1)} = \mathbf{\Sigma}_{11}^{-T/2}(\mathbf{x}^{(1)} - \boldsymbol{\mu}^{(1)})$
- $\tilde{\mathbf{x}}^{(2)} = \mathbf{\Sigma}_{22}^{-T/2}(\mathbf{x}^{(2)} - \boldsymbol{\mu}^{(2)})$

Then $V[\tilde{\mathbf{x}}^{(1)}] = \mathbf{I}_p$, $V[\tilde{\mathbf{x}}^{(2)}] = \mathbf{I}_q$ so both $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ are multistandardized.

Combine $\tilde{\mathbf{x}}^{(1)}$ and $\tilde{\mathbf{x}}^{(2)}$ in one vector with p+q elements:

- $\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{\mathbf{x}}^{(1)} \\ \tilde{\mathbf{x}}^{(2)} \end{bmatrix}$.

Then

$$V[\widetilde{\mathbf{x}}] = \widetilde{\mathbf{\Sigma}} = \begin{bmatrix} \mathbf{I}_p & \widetilde{\mathbf{\Sigma}}_{12} \\ & \\ \widetilde{\mathbf{\Sigma}}_{21} & \mathbf{I}_q \end{bmatrix}, \widetilde{\mathbf{\Sigma}}_{12} = \widetilde{\mathbf{\Sigma}}_{21}^T$$

where $\widetilde{\mathbf{\Sigma}}_{12} \equiv \mathrm{Cov}[\widetilde{\mathbf{x}}^{(1)}, \widetilde{\mathbf{x}}^{(2)}] = \mathbf{\Sigma}_{11}^{-T/2}\mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1/2}$.

Because $V[\widetilde{\mathbf{x}}^{(1)}] = \mathbf{I}_p$, and $V[\widetilde{\mathbf{x}}^{(2)}] = \mathbf{I}_q$,

$$\widetilde{\mathbf{\Sigma}}_{12} = \mathrm{Corr}[\widetilde{\mathbf{x}}^{(1)}, \widetilde{\mathbf{x}}^{(2)}].$$

$\widetilde{\mathbf{\Sigma}}_{12}$ is p by q and contains all the information on the linear association of $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$.

$$\mathrm{rank}(\widetilde{\mathbf{\Sigma}}_{12}) \leq \min(p,q)$$

Note that $\mathbf{\Sigma}_{12} = \mathbf{0} \Leftrightarrow \widetilde{\mathbf{\Sigma}}_{12} = \mathbf{0}$.

One way to try to describe the correlation between $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ is to use the SVD of $\widetilde{\mathbf{\Sigma}}_{12}$ to find a rank $m < \min(p,q)$ matrix $\widetilde{\mathbf{\Sigma}}_{12}^{(m)}$ which is as close as possible to $\widetilde{\mathbf{\Sigma}}_{12}$.

The **SVD** of $\widetilde{\mathbf{\Sigma}}_{12}$ is $\widetilde{\mathbf{\Sigma}}_{12} = \mathbf{LTR}^T$

- $\mathbf{L} = [\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_q]$, p by q. $\mathbf{L}^T\mathbf{L} = \mathbf{I}_p$
  $\boldsymbol{\ell}_j$ = (left singular vector of $\widetilde{\mathbf{\Sigma}}_{12}$)
  = (eigenvector of $\widetilde{\mathbf{\Sigma}}_{12}\widetilde{\mathbf{\Sigma}}_{12}^T$)
- $\mathbf{T} = \mathrm{diag}[\tau_1 \geq \tau_2 \geq \ldots \geq \tau_q]$,
  τ_j = singular value of $\widetilde{\mathbf{\Sigma}}_{12}$
  = square root of eigenvalue θ_j of $\widetilde{\mathbf{\Sigma}}_{12}\widetilde{\mathbf{\Sigma}}_{12}^T$ or of $\widetilde{\mathbf{\Sigma}}_{12}^T\widetilde{\mathbf{\Sigma}}_{12}$.
- $\mathbf{R} = [\mathbf{r}_1, \ldots, \mathbf{r}_q]$, q by q, $\mathbf{R}^T\mathbf{R} = \mathbf{I}_q$
  $\mathbf{r}_j$ = (right singular vector of $\widetilde{\mathbf{\Sigma}}_{12}$)
  = (eigenvector of $\widetilde{\mathbf{\Sigma}}_{12}^T\widetilde{\mathbf{\Sigma}}_{12}$)

Then the best rank m approximation is

$$\widetilde{\mathbf{\Sigma}}_{12}^{(m)} = \mathbf{L}^{(m)}\mathbf{T}^{(m)}\mathbf{R}^{(m)} = \sum_{1\leq i\leq m}\tau_i\boldsymbol{\ell}_i\mathbf{r}_i^T$$

- $\mathbf{L}^{(m)} = [\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_m]$, p by m
- $\mathbf{R}^{(m)} = [\mathbf{r}_1, \ldots, \mathbf{r}_m]$, q by m
- $\mathbf{T}^{(m)} = \mathrm{diag}[\tau_1, \ldots, \tau_m]$

The singular values $\tau_1 \geq \tau_2 \geq \dots$ of $\tilde{\boldsymbol{\Sigma}}_{12}$ are the **population canonical correlations**.

Their squares, $\theta_j = \tau_j^2$, are eigenvalues of both $\tilde{\boldsymbol{\Sigma}}_{12}\tilde{\boldsymbol{\Sigma}}_{12}^T$ and $\tilde{\boldsymbol{\Sigma}}_{12}^T\tilde{\boldsymbol{\Sigma}}_{12}$

Now $\tilde{\boldsymbol{\Sigma}}_{12} = \boldsymbol{\Sigma}_{11}^{-T/2}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1/2}$ implies that

- $\tilde{\boldsymbol{\Sigma}}_{12}\tilde{\boldsymbol{\Sigma}}_{12}^T = \boldsymbol{\Sigma}_{11}^{-T/2}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1/2}$
- $\tilde{\boldsymbol{\Sigma}}_{12}^T\tilde{\boldsymbol{\Sigma}}_{12} = \boldsymbol{\Sigma}_{22}^{-T/2}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1/2}$

Therefore $\theta_j = \tau_j^2$ is

- eigenvalue of $\boldsymbol{\Sigma}_{11}^{-T/2}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1/2}$
- eigenvalue of $\boldsymbol{\Sigma}_{22}^{-T/2}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1/2}$

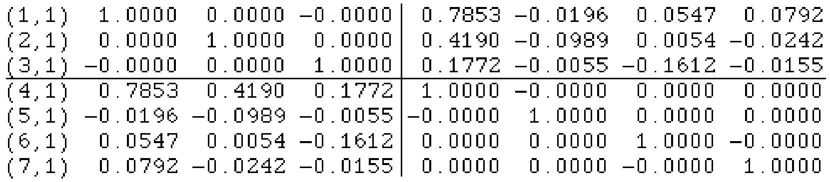
Moreover, θ_j is

- eigenvalue of $\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$ relative to $\boldsymbol{\Sigma}_{11}$
- the regular eigenvalue of $\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$
- eigenvalue of $\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}$ relative to $\boldsymbol{\Sigma}_{22}$
- the regular eigenvalue of $\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}$

The best rank 1 and 2 approximations to $\tilde{\boldsymbol{\Sigma}}_{12}$ are

$$\tilde{\boldsymbol{\Sigma}}_{12}^{(1)} \equiv \tau_1\boldsymbol{\ell}_1\mathbf{r}_1^T \text{ and } \tilde{\boldsymbol{\Sigma}}_{12}^{(1)} \equiv \tau_1\boldsymbol{\ell}_1\mathbf{r}_1^T + \tau_2\boldsymbol{\ell}_2\mathbf{r}_2^T$$

```
Cmd> x1tilde <- (x1 - sum(x1)/N) %*% solve(cholesky(s11))

Cmd> x2tilde <- (x2 - sum(x2)/N) %*% solve(cholesky(s22))

Cmd> xtilde <- hconcat(x1tilde, x2tilde)

Cmd> list(xtilde)
xtilde          REAL   100   7

Cmd> rtilde <- tabs(xtilde,covar:T)

Cmd> print(rtilde,format:"7.4f")
MATRIX:
```



```
(1,1)  1.0000  0.0000 -0.0000 | 0.7853 -0.0196  0.0547  0.0792
(2,1)  0.0000  1.0000  0.0000 | 0.4190 -0.0989  0.0054 -0.0242
(3,1) -0.0000  0.0000  1.0000 | 0.1772 -0.0055 -0.1612 -0.0155
(4,1)  0.7853  0.4190  0.1772 | 1.0000 -0.0000  0.0000  0.0000
(5,1) -0.0196 -0.0989 -0.0055 |-0.0000  1.0000  0.0000  0.0000
(6,1)  0.0547  0.0054 -0.1612 | 0.0000  0.0000  1.0000 -0.0000
(7,1)  0.0792 -0.0242 -0.0155 | 0.0000  0.0000 -0.0000  1.0000
```

```
Cmd> rtilde12 <- rtilde[J1,J2]; rtilde12
(1,1)     0.78526  -0.019556   0.054711   0.079215
(2,1)     0.41898   -0.09892  0.0054408  -0.024196
(3,1)     0.17715 -0.0054845   -0.16123  -0.015524
```

The upper right hand corner is $\tilde{\mathbf{R}}_{12}$, the matrix of correlations between the multistandardized vectors $\tilde{\mathbf{x}}^{(1)}$ and $\tilde{\mathbf{x}}^{(2)}$.

Let's see how well the rank 1 and rank 2 approximations to $\hat{\tilde{\boldsymbol{\rho}}}_{12} = \hat{\tilde{\boldsymbol{\Sigma}}}_{12}$ are

```
Cmd> svdall <- svd(rtilde12,all:T) # complete SVD

Cmd> J <- 1 # selector for rank 1 approximation

Cmd> leftvecs <- svdall$leftvectors[,J];

Cmd> rightvecs <- svdall$rightvectors[,J]

Cmd> svals <- dmat(svdall$values[J])

Cmd> leftvecs %*% svals %*% rightvecs'# rank 1 approximation
(1,1)     0.78549  -0.055253   0.016781   0.047095
(2,1)     0.42081  -0.029601  0.0089902   0.025231
(3,1)     0.17162  -0.012072  0.0036664    0.01029

Cmd> J <- run(2) # selector for rank 2 approximation

Cmd> leftvecs <- svdall$leftvectors[,J]

Cmd> rightvecs <- svdall$rightvectors[,J]

Cmd> svals <- dmat(svdall$values[J])

Cmd> leftvecs %*% svals %*% rightvecs'
(1,1)      0.78411   -0.052391    0.061766    0.057415
(2,1)      0.42137   -0.030758  -0.0092031    0.021057
(3,1)      0.17656   -0.022332    -0.15762   -0.026709
```

Here I used a useful MacAnova "trick". I set **J** to the indices of the singular values and vectors to be used so I could use the same MacAnova expressions for both rank 1 and rank 2 approximations.

Define the first pair of **correlation canonical variables** by

- $z_1^{(1)} = \boldsymbol{\ell}_1^T\tilde{\mathbf{x}}^{(1)} = \boldsymbol{\ell}_1^T\boldsymbol{\Sigma}_{11}^{-T/2}(\mathbf{x}^{(1)} - \boldsymbol{\mu}^{(1)})$
  $= (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_1)^T\mathbf{x}^{(1)} - \nu_1^{(1)}$,
  $\nu_1^{(1)} = (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_1)^T\boldsymbol{\mu}^{(1)}$
- $z_1^{(2)} = \mathbf{r}_1^T\tilde{\mathbf{x}}^{(2)} = \mathbf{r}_1^T\boldsymbol{\Sigma}_{22}^{-T/2}(\mathbf{x}^{(2)} - \boldsymbol{\mu}^{(2)})$
  $= (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_1)^T\mathbf{x}^{(2)} - \nu_1^{(2)}$,
  $\nu_1^{(2)} = (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_1)^T\boldsymbol{\mu}^{(2)}$

Then $\text{corr}[z_1^{(1)}, z_1^{(2)}] = \boldsymbol{\ell}_1^T\tilde{\boldsymbol{\Sigma}}_{12}\mathbf{r}_1 = \tau_1$.

More generally, you can define min(p,q) pairs of canonical variables $z_1^{(j)}$, $z_2^{(j)}$ by

- $z_j^{(1)} = \boldsymbol{\ell}_j^T\tilde{\mathbf{x}}^{(1)} = (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_j)^T\mathbf{x}^{(1)} - \nu_j^{(1)}$,
  $\nu_j^{(1)} = (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_j)^T\boldsymbol{\mu}^{(1)}$
- $z_j^{(2)} = \mathbf{r}_j^T\tilde{\mathbf{x}}^{(2)} = (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_j)^T\mathbf{x}^{(2)} - \nu_j^{(1)}$,
  $\nu_j^{(2)} = (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_j)^T\boldsymbol{\mu}^{(2)}$
- $\text{corr}[z_j^{(1)}, z_j^{(2)}] = \boldsymbol{\ell}_j^T\tilde{\boldsymbol{\Sigma}}_{12}\mathbf{r}_j = \tau_j$,
  $1 \le j \le \min(p,q)$

All other correlations among canonical variables are 0. Specifically,

$$\mathrm{cov}[z_j^{(1)}, z_k^{(1)}] = \boldsymbol{\ell}_j^T\boldsymbol{\ell}_k = 0,\ j \neq k$$

$$\mathrm{cov}[z_j^{(2)}, z_k^{(2)}] = \mathbf{r}_j^T\mathbf{r}_k = 0,\ j \neq k$$

$$\mathrm{cov}[z_j^{(1)}, z_k^{(2)}] = \boldsymbol{\ell}_j^T\widetilde{\boldsymbol{\Sigma}}_{12}\mathbf{r}_k = 0,\ j \neq k$$

In a sense, all the correlations between $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ are "squeezed" into τ_1, ..., $\tau_{\min(p,q)}$.

```
Cmd> z1 <- x1tilde %*% svdall$leftvectors[,-4]

Cmd> z2 <- x2tilde %*% svdall$rightvectors[,-4]

Cmd> list(z1,z2) # matrices of canonical variables
z1              REAL   100   3
z2              REAL   100   3

Cmd> print(cor(z1,z2),format:"8.5f")
MATRIX:
(1,1)  1.00000  0.00000  0.00000: 0.91156 -0.00000 -0.00000
(2,1)  0.00000  1.00000  0.00000: 0.00000  0.17321  0.00000
(3,1)  0.00000  0.00000  1.00000:-0.00000 -0.00000  0.09456
-------------------------------------------------------------
(4,1)  0.91156  0.00000 -0.00000: 1.00000  0.00000  0.00000
(5,1) -0.00000  0.17321 -0.00000: 0.00000  1.00000 -0.00000
(6,1) -0.00000  0.00000  0.09456: 0.00000 -0.00000  1.00000
```

What you really want are how to get canonical variables directly from $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$.

- $z_j^{(1)} = \boldsymbol{\ell}_j^T\widetilde{\mathbf{x}}^{(1)} = \boldsymbol{\ell}_j^T\boldsymbol{\Sigma}_{11}^{-T/2}(\mathbf{x}^{(1)} - \boldsymbol{\mu}^{(1)})$
  $= (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_j)^T(\mathbf{x}^{(1)} - \boldsymbol{\mu}^{(1)})$
  $= (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_1)^T\mathbf{x}^{(1)} - \nu_j^{(1)}$,
  $\nu_j^{(1)} = (\boldsymbol{\Sigma}_{11}^{-1/2}\boldsymbol{\ell}_j)^T\boldsymbol{\mu}^{(1)}$
- $z_j^{(2)} = \mathbf{r}_j^T\widetilde{\mathbf{x}}^{(2)} = \mathbf{r}_j^T\boldsymbol{\Sigma}_{22}^{-T/2}(\mathbf{x}^{(2)} - \boldsymbol{\mu}^{(2)})$
  $= (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_j)^T(\mathbf{x}^{(2)} - \boldsymbol{\mu}^{(2)})$
  $= (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_1)^T\mathbf{x}^{(2)} - \nu_j^{(2)}$,
  $\nu_j^{(2)} = (\boldsymbol{\Sigma}_{22}^{-1/2}\mathbf{r}_j)^T\boldsymbol{\mu}^{(2)}$