

Multivariate Data Analysis: Tasks for Week 1

Notes & Solutions

1) *Read the Study Guide for this course if you have not already done so*
Trust you have done this by now

2) *If A is any $p \times q$ matrix then $\text{var}(X'A) = A' \text{var}(X) A = A' S A$,*

This actually follows directly from the expression for $\text{var}(Y)$ putting $y_i = A'x_i$ etc and is essentially identical to the special case when $q=1$ and A is a vector.

3) *Access the Iris Dataset.*

i) *Find the 4-vector which is the mean of the four dimensions Sepal.l, Sepal.w, Petal.l, Petal.w and the 4x4 matrix which is their variance.*

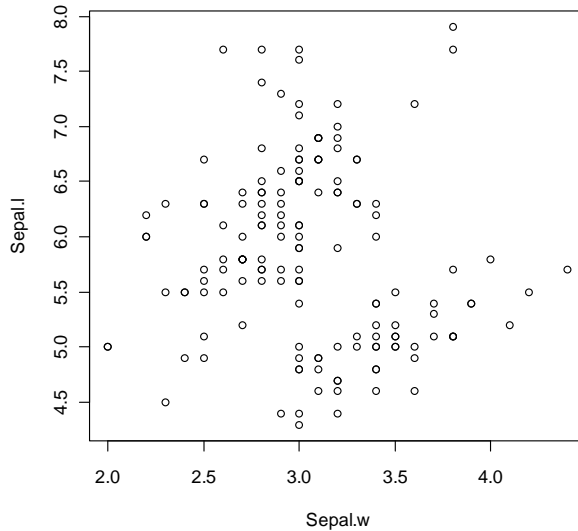
```
> attach(irisnf)
> apply(cbind(Sepal.l, Sepal.w, Petal.l, Petal.w), 2, mean)
Sepal.l Sepal.w Petal.l Petal.w
5.8433  3.0553  3.758  1.1993
> var(cbind(Sepal.l, Sepal.w, Petal.l, Petal.w))
      Sepal.l  Sepal.w  Petal.l  Petal.w
Sepal.l 0.685694 -0.040736  1.27432  0.51627
Sepal.w -0.040736  0.193629 -0.32873 -0.12124
Petal.l  1.274315 -0.328734  3.11628  1.29561
Petal.w  0.516271 -0.121238  1.29561  0.58101
>
```



ii) . Plot sepal length against sepal width using:

a) the default choices

```
> plot(Sepal.w, Sepal.l)
```

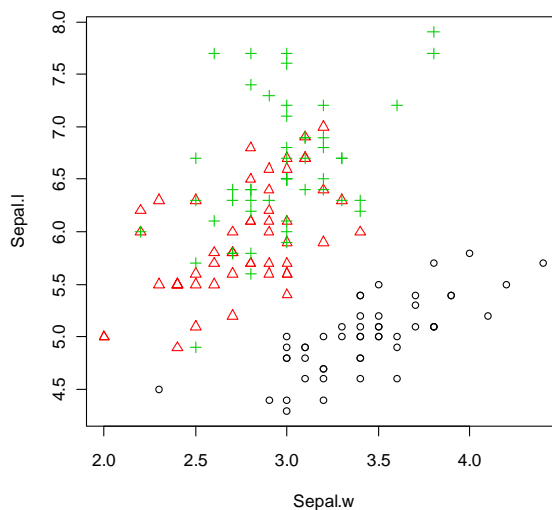


```
>
```

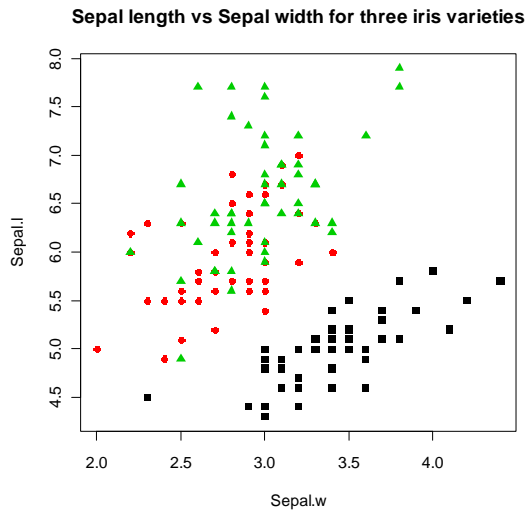
Note that to plot length **against** width you **must** have length on vertical and width on horizontal axis.

b) using different symbols for each variety (explore the menus and panels, and maybe the help system to find out how to do this). Also try adding titles etc.

```
> plot(Sepal.w, Sepal.l, pch=unclass(Variety),
+ col=unclass(Variety))
```

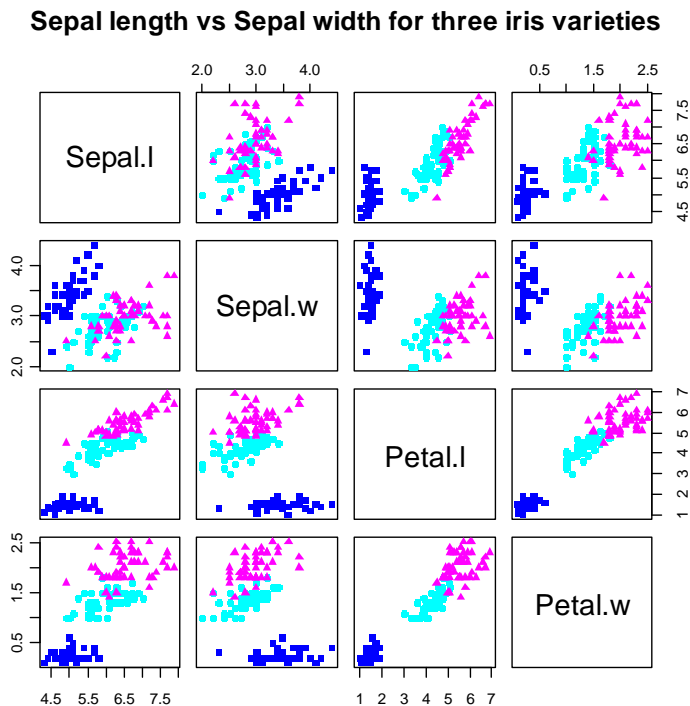


```
> plot(Sepal.w,Sepal.l, pch=unclass(Variety)+14,
+ col=unclass(Variety),
+ main="Sepal length vs Sepal width for three iris varieties")
```



iii) Construct a matrix plot of all four dimensions, using first the default choices and then enhancing the display as above.

```
> pairs(cbind(Sepal.l,Sepal.w,Petal.l,Petal.w),
+ pch=unclass(Variety)+14, col=unclass(Variety)+3,
+ main="Sepal length vs Sepal width for three iris varieties")
```



iv) Try the commands

```

var(irisnf)
diag(var(irisnf))

> options(digits=3)
> var(irisnf)
      Sepal.l Sepal.w Petal.l Petal.w Variety
Sepal.l 0.6857 -0.0407  1.274  0.516  0.531
Sepal.w -0.0407  0.1936 -0.329 -0.121 -0.152
Petal.l  1.2743 -0.3287  3.116  1.296  1.372
Petal.w  0.5163 -0.1212  1.296  0.581  0.597
Variety  0.5309 -0.1523  1.372  0.597  0.671
> diag(var(irisnf))
Sepal.l Sepal.w Petal.l Petal.w Variety
 0.686   0.194   3.116   0.581   0.671
>

```

4) Try these simple exercises both 'by hand' and R.

i) Let $a = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$, $A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$,

Find AB , $B'A'$, BA , $a'A$, $a'Aa$

```

> a<-matrix(c(1,2,3),3,1)
> A<-matrix(c(1,2,3,4,5,6),2,3,byrow=T)
> B<-matrix(c(1,2,3,4,5,6),3,2,byrow=T)
> A**B
      [,1] [,2]
[1,]   22  28
[2,]   49  64
> t(B)**t(A)
      [,1] [,2]
[1,]   22  49
[2,]   28  64
> B**A
      [,1] [,2] [,3]
[1,]    9  12  15
[2,]   19  26  33
[3,]   29  40  51
> t(a)**A
Error in t(a) ** A : non-conformable arguments
> t(a)**A**a
Error in t(a) ** A : non-conformable arguments
>

```

Note that $a'A$ and $a'Aa$ are not defined since the dimensions do not match.



- 5) *Read through the Sections on eigenvalues and eigenvectors, differentiation w.r.t. vectors and use of Lagrange Multipliers in the Background Results booklet. I trust that you have done this by now and would have contacted me if there were any problems.*
- 6) *Read the Study Guide for this course [again] if you have not already done so [or have done so only once]...and this also [again].*



Multivariate Data Analysis: Tasks for Week 2

Notes & Solutions

1)

i) Find the eigenvalues and normalized eigenvectors of the 2×2 matrix

$$\frac{1}{7} \begin{pmatrix} 208 & 144 \\ 144 & 292 \end{pmatrix}$$

Solving $\begin{vmatrix} \frac{208}{7} - \lambda & \frac{144}{7} \\ \frac{144}{7} & \frac{292}{7} - \lambda \end{vmatrix} = 0$ gives $\lambda^2 - 500/7\lambda + 40000/7^2 = 0$ so $\lambda_1 = 400/7$

and $\lambda_2 = 100/7$. Putting $(S - \lambda_1 I_2)a_1 = 0$ gives

$$-192a_{11} + 144a_{12} = 0 \text{ and } 144a_{11} - 108a_{12} = 0.$$

(Note that these two equations are essentially identical).

Using the normalizing constraint that $a_{11}^2 + a_{12}^2 = 1$ gives $a_{11} = 3/5 = 0.6$

and $a_{12} = 0.8$. Similarly $a_{21} = 0.8$ and $a_{22} = -0.6$

(note that $a_1 = (-0.6, -0.8)'$ and/or $a_2 = (-0.8, 0.6)'$ are equally acceptable solutions for a_1 and a_2 since the signs of eigenvectors are arbitrary).

```
> s<-matrix(c(208,144,144,292),nrow=2,ncol=2)/7
> eigen(s)
$values
[1] 57.14286 14.28571
```

```
$vectors
      [,1] [,2]
[1,]  0.6 -0.8
[2,]  0.8  0.6
```

```
>
```



ii) Find the eigenvalues and one possible set of normalized eigenvectors of

the 3×3 matrix $\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$

$$|S - \lambda I_3| = \lambda^3 - 6\lambda^2 + 9\lambda - 4 = (\lambda - 4)(\lambda - 1)^2, \text{ so } \lambda_1 = 4 \text{ and } \lambda_2 = \lambda_3 = 1.$$

$(S - \lambda I_3)a_1 = 0 \Rightarrow a_{12} + a_{13} = 2a_{11}, a_{11} + a_{13} = 2a_{12}$ and $a_{11} + a_{12} = 2a_{13}$ so $a_{11} = a_{12} = a_{13}$ and since $a_1' a_1 = 1$ we have $a_1 = 3^{-1/2}(1, 1, 1)'$. For a_2 and a_3 we need **any** two normalized orthogonal vectors which are also orthogonal to the unit vector:

e.g. $6^{-1/2}(1, 1, -2)'$ and $2^{-1/2}(1, -1, 0)'$ or equally well $38^{-1/2}(2, 3, -5)'$ and $114^{-1/2}(-8, 7, 1)$ or infinitely many other possibilities.

```
> options(digits=3)
> t<-matrix(c(2,1,1,1,2,1,1,1,2),nrow=3,ncol=3)
> eigen(t)
$values
[1] 4 1 1

$vectors
      [,1] [,2] [,3]
[1,] -0.577 0.816 0.000
[2,] -0.577 -0.408 -0.707
[3,] -0.577 -0.408 0.707
```

iii) Find the inverse of $\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix}$. $|S|=6; S^{-1} = \frac{1}{6} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 4 & -2 \\ 0 & -2 & 4 \end{pmatrix}$

```
> solve(matrix(c(2,0,0,0,2,1,0,1,2),nrow=3,ncol=3))
      [,1] [,2] [,3]
[1,] 0.5 0.000 0.000
[2,] 0.0 0.667 -0.333
[3,] 0.0 -0.333 0.667
```



2) (optional — but at least note the results, these are counterexamples to false assumptions that are all too easy to make since they contradict ‘ordinary’ algebra).

$$\text{Let } A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, C = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, D = \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix},$$

$$E = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \text{ and } F = \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} \text{ then show:—}$$

```
> A<-matrix(c(0,1,-1,0),2,2,byrow=T)
> B<-matrix(c(0,1,0,0),2,2,byrow=T)
> C<-matrix(c(1,1,1,-1),2,2,byrow=T)
> D<-matrix(c(1,-1,-1,-1),2,2,byrow=T)
> E<-matrix(c(1,1,1,1),2,2)
> F<-matrix(c(1,1,-1,-1),2,2,byrow=T)
```

<p>i) $A^2 = -I_2$ (so A is ‘like’ the square root of -1)</p> <pre>> A%%A [,1] [,2] [1,] -1 0 [2,] 0 -1</pre> <p>ii) $B^2 = 0$ (but $B \neq 0$)</p> <pre>> B%%B [,1] [,2] [1,] 0 0 [2,] 0 0</pre>	<p>iii) $CD = -DC$ (but $CD \neq 0$)</p> <pre>> C%%D [,1] [,2] [1,] 0 -2 [2,] 2 0</pre> <pre>> D%%C [,1] [,2] [1,] 0 2 [2,] -2 0</pre> <p>iv) $EF = 0$ (but $E \neq 0$ and $F \neq 0$)</p> <pre>> E%%F [,1] [,2] [1,] 0 0 [2,] 0 0</pre>
---	--



3) (see 0.10.1) The data file *openclosed.Rdata** consists of examination marks in five subjects labelled *mec*, *vec*, *alg*, *ana* and *sta*. Download the datafile to your own hard disk. Using Windows Explorer double click on the file. This will open **R**, change the working directory to that where you have stored the data and read in the data to dataframe *scor*. *Mardia, Kent & Bibby (1981).

i) Then issue the following commands and read the results

```
> ls() # see what objects are in the works space;
[1] "scor"
> # there should be only the dataframe scor
>
> X<-as.matrix(t(scor)) # define X to be the matrix
> # of the transpose of scor
>
> S<-var(t(X)) # calculate the variance matrix of X'=scor
>
> A<-eigen(S)$vectors # Calculate the eigenvectors of S
> # & store them in A
> V<-eigen(S)$values # and eigenvalues in V
> A # look at A
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] -0.5054457  0.74874751 -0.2997888  0.296184264 -0.07939388
[2,] -0.3683486  0.20740314  0.4155900 -0.782888173 -0.18887639
[3,] -0.3456612 -0.07590813  0.1453182 -0.003236339  0.92392015
[4,] -0.4511226 -0.30088849  0.5966265  0.518139724 -0.28552169
[5,] -0.5346501 -0.54778205 -0.6002758 -0.175732020 -0.15123239
> V # look at V
[1] 686.98981 202.11107 103.74731 84.63044 32.15329
> sum(diag(S))# look at trace(S)
[1] 1109.632
> sum(V) # look at sum of eigenvalues in V (they should
be the same)
[1] 1109.632
>
> options(digits=4) # only print four decimal places
>
> A%*%t(A) # check that A is an orthogonal matrix
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 1.000e+00  1.476e-16 -2.964e-17  4.014e-17 -1.586e-17
[2,] 1.476e-16  1.000e+00 -1.441e-16 -2.639e-16  3.010e-16
[3,] -2.964e-17 -1.441e-16  1.000e+00 -1.121e-16 -3.787e-16
[4,] 4.014e-17 -2.639e-16 -1.121e-16  1.000e+00 -3.263e-16
[5,] -1.586e-17  3.010e-16 -3.787e-16 -3.263e-16  1.000e+00
```



```

> t(A)%*%A # (as it should be, property of eigenvectors)
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 1.000e+00 -6.101e-17 1.099e-16 -2.397e-16 1.118e-16
[2,] -6.101e-17 1.000e+00 -1.115e-16 1.241e-16 1.837e-16
[3,] 1.099e-16 -1.115e-16 1.000e+00 8.888e-16 1.701e-16
[4,] -2.397e-16 1.241e-16 8.888e-16 1.000e+00 -1.225e-16
[5,] 1.118e-16 1.837e-16 1.701e-16 -1.225e-16 1.000e+00
>
> round(A%*%t(A)) # easier to see if round to whole numbers
      [,1] [,2] [,3] [,4] [,5]
[1,] 1 0 0 0 0
[2,] 0 1 0 0 0
[3,] 0 0 1 0 0
[4,] 0 0 0 1 0
[5,] 0 0 0 0 1
> round(t(A)%*%A)
      [,1] [,2] [,3] [,4] [,5]
[1,] 1 0 0 0 0
[2,] 0 1 0 0 0
[3,] 0 0 1 0 0
[4,] 0 0 0 1 0
[5,] 0 0 0 0 1
>
> t(A)%*%S%*%A # calculate A'SA
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 6.870e+02 2.381e-13 -1.029e-13 6.612e-14 4.718e-14
[2,] 2.595e-13 2.021e+02 3.081e-15 -3.109e-15 -2.730e-15
[3,] -1.219e-13 -1.259e-14 1.037e+02 5.388e-14 -8.734e-15
[4,] 7.972e-14 1.552e-14 4.434e-14 8.463e+01 3.257e-14
[5,] 3.606e-14 5.202e-15 -3.147e-15 3.728e-14 3.215e+01
>
> Y<-t(A)%*%X # let Y=A'X so that Y'=X'A, the data rotated
> # onto the principal components.
> var(t(Y)) # the variance of the data on the
principal components
      [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 6.870e+02 2.678e-13 -1.291e-13 9.386e-14 2.932e-14
[2,] 2.678e-13 2.021e+02 4.731e-15 1.460e-14 1.758e-15
[3,] -1.291e-13 4.731e-15 1.037e+02 3.553e-14 5.889e-15
[4,] 9.386e-14 1.460e-14 3.553e-14 8.463e+01 3.651e-14
[5,] 2.932e-14 1.758e-15 5.889e-15 3.651e-14 3.215e+01
> # note these are the same up to rounding errors
> round(t(A)%*%S%*%A) # easier to see if round to whole
numbers
      [,1] [,2] [,3] [,4] [,5]
[1,] 687 0 0 0 0
[2,] 0 202 0 0 0
[3,] 0 0 104 0 0
[4,] 0 0 0 85 0
[5,] 0 0 0 0 32

```



```

> round(var(t(Y)))
      [,1] [,2] [,3] [,4] [,5]
[1,] 687   0   0   0   0
[2,]  0  202   0   0   0
[3,]  0   0  104   0   0
[4,]  0   0   0   85   0
[5,]  0   0   0   0   32
> V          # eigenvalues of S, also same.
[1] 686.99 202.11 103.75  84.63  32.15
> sum(diag(S)) # find trace(S)
[1] 1110
> sum(V)       # same as above
[1] 1110
>

```

4) The data file *bodysize.Rdata** consists of measurements of the circumferences (in centimetres) of neck, chest, abdomen, hip, thigh, knee, ankle, biceps, forearm and wrist of 252 men. Download the datafile to your own hard disk. Using Windows Explorer double click on the file. This will open **R**, change the working directory to that where you have stored the data and read in the data to dataframe *bodysize*. Next, download the function *screepplot()* contained in scriptfile *scree.R* to the same directory on your hard disk. Using the menu in **R** open the script file *scree.R* (top left icon in the menu bar), highlight all the lines in the function and click the middle icon to run the selected lines. This will load the function into your current **R** session. *source: *Journal of Statistics Education Data Archive*

i) Then issue the following commands and read the results

```

bodysize[1:5,]          # gives first few lines of the data file
diag(var(bodysize))    # gives variances of variables
sqrt(diag(var(bodysize))) # gives standard deviations
# note standard deviations vary by a factor of > 10
# so perform PCA with correlation matrix
body.pc<-princomp(bodysize,cor=T)
body.pc
summary(body.pc)
body.pc$loadings
screepplot(bodysize,T)
print(body.pc$loadings, cutoff=0.01)

> # function to draw screeplots of cumulative
> # eigenvalues in principal component analysis
>
> screepplot<-function(mydata,cor=F,maxcomp=10) {

```



```

+ my.pc<-princomp(mydata, cor=cor)
+ k<-min(dim(mydata),maxcomp)
+ x<-c(0:k)
+ y<-my.pc$sdev[1:k]*my.pc$sdev[1:k]
+ y<-c(0,y)
+ z<-100*cumsum(y)/sum(my.pc$sdev*my.pc$sdev)
+
+ plot(x,z,type="l",xlab="number of dimensions",
+ cex.main=1.5, lwd=3, col="red",
+ ylim=c(0,100),
+ ylab="cumulative percentage of total variance",
+ main="Scree plot of variances",
+ xaxt="n", yaxt="n")
+
+ axis(1,at=x,lwd=2)
+ axis(2,at=c(0,20,40,60,80,100),lwd=2)
+ abline(a=100,b=0,lwd=2,lty="dashed",col="orange")
+ text(x,z,labels=x,cex=0.8,adj=c(1.2,-.1),col="blue")
+ }
>
> bodysize[1:5,] # gives first few lines of the data file
  neck chest abdomen  hip thigh knee ankle biceps forearm wrist
1 36.2  93.1    85.2 94.5  59.0 37.3  21.9  32.0  27.4  17.1
2 38.5  93.6    83.0 98.7  58.7 37.3  23.4  30.5  28.9  18.2
3 34.0  95.8    87.9 99.2  59.6 38.9  24.0  28.8  25.2  16.6
4 37.4 101.8    86.4 101.2 60.1 37.3  22.8  32.4  29.4  18.2
5 34.4  97.3   100.0 101.9 63.2 42.2  24.0  32.2  27.7  17.7
> diag(var(bodysize)) # gives variances of variables
  neck  chest abdomen  hip  thigh  knee  ankle  biceps forearm
wrist
 5.909  71.073 116.275 51.324 27.562  5.817  2.873  9.128  4.083
0.872
> sqrt(diag(var(bodysize))) # gives standard deviations
  neck  chest abdomen  hip  thigh  knee  ankle  biceps forearm
wrist
 2.431  8.430 10.783  7.164  5.250  2.412  1.695  3.021  2.021
0.934
> # note standard deviations vary by a factor of > 10
> # so perform PCA with correlation matrix
> body.pc<-princomp(bodysize,cor=T)
> body.pc
Call:
princomp(x = bodysize, cor = T)

Standard deviations:
  Comp.1  Comp.2  Comp.3  Comp.4  Comp.5  Comp.6  Comp.7  Comp.8  Comp.9
Comp.10
  2.650  0.853  0.819  0.701  0.547  0.528  0.452  0.405  0.278
0.253

 10 variables and 252 observations.
> summary(body.pc)
Importance of components:
              Comp.1  Comp.2  Comp.3  Comp.4  Comp.5  Comp.6  Comp.7  Comp.8
Standard deviation  2.650 0.8530 0.8191 0.7011 0.5471 0.5283 0.4520 0.4054
Proportion of Variance 0.702 0.0728 0.0671 0.0492 0.0299 0.0279 0.0204 0.0164
Cumulative Proportion 0.702 0.7749 0.8420 0.8912 0.9211 0.9490 0.9694 0.9859
              Comp.9  Comp.10
Standard deviation 0.27827 0.2530
Proportion of Variance 0.00774 0.0064
Cumulative Proportion 0.99360 1.0000

```



> **body.pc\$loadings**

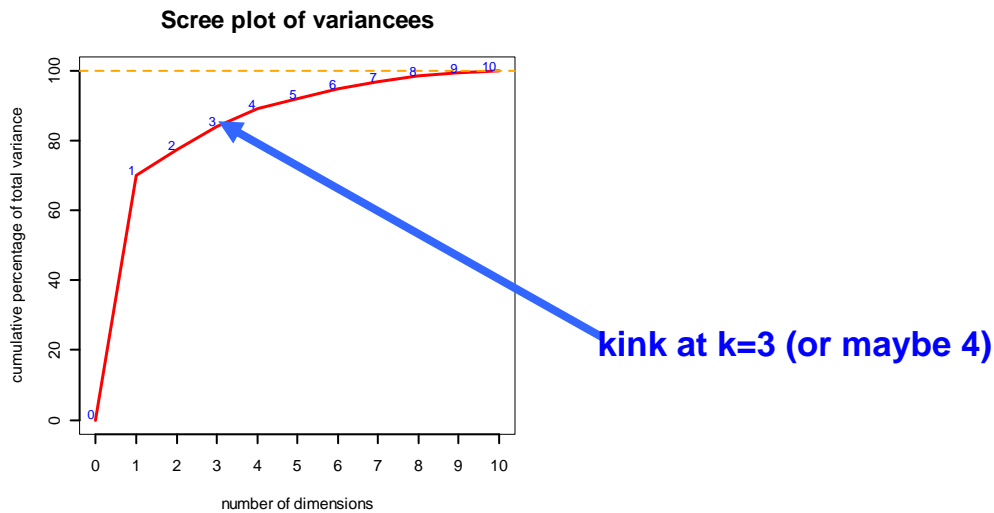
Loadings:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9	Comp.10
neck	-0.327		-0.259	0.339		0.288	0.719	0.318		
chest	-0.339	0.273		0.243	-0.447		-0.235	0.127	-0.543	-0.419
abdomen	-0.334	0.398		0.216	-0.310	-0.147	-0.134		0.303	0.669
hip	-0.348	0.255	0.210	-0.119				-0.349	0.551	-0.563
thigh	-0.333	0.191	0.180	-0.411	0.255	0.105	0.289	-0.404	-0.524	0.234
knee	-0.329		0.273	-0.135	0.446	-0.442	-0.118	0.624		
ankle	-0.247	-0.625	0.583		-0.416	0.168				
biceps	-0.322		-0.256	-0.304		0.671	-0.471	0.197	0.130	
forearm	-0.270	-0.363	-0.590	-0.404	-0.262	-0.440				
wrist	-0.299	-0.377	-0.141	0.568	0.429		-0.271	-0.396		

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9
SS loadings	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Proportion Var	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Cumulative Var	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9

	Comp.10
SS loadings	1.0
Proportion Var	0.1
Cumulative Var	1.0

> **screeplot(bodysize,T)**



> **print(body.pc\$loadings, cutoff=0.01)**

Loadings:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9	Comp.10
neck	-0.327		-0.259	0.339	0.054	0.288	0.719	0.318	0.079	-0.023
chest	-0.339	0.273	-0.059	0.243	-0.447	-0.081	-0.235	0.127	-0.543	-0.419
abdomen	-0.334	0.398	0.066	0.216	-0.310	-0.147	-0.134	-0.061	0.303	0.669
hip	-0.348	0.255	0.210	-0.119	0.059	-0.070	0.071	-0.349	0.551	-0.563
thigh	-0.333	0.191	0.180	-0.411	0.255	0.105	0.289	-0.404	-0.524	0.234
knee	-0.329	-0.022	0.273	-0.135	0.446	-0.442	-0.118	0.624	-0.011	0.013
ankle	-0.247	-0.625	0.583	-0.022	-0.416	0.168	0.066	0.016	0.022	0.047
biceps	-0.322	-0.022	-0.256	-0.304	0.094	0.671	-0.471	0.197	0.130	0.031
forearm	-0.270	-0.363	-0.590	-0.404	-0.262	-0.440	0.087	-0.092	0.068	0.029
wrist	-0.299	-0.377	-0.141	0.568	0.429	-0.073	-0.271	-0.396	-0.076	0.033



```

                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
SS loadings    1.0    1.0    1.0    1.0    1.0    1.0    1.0    1.0    1.0
Proportion Var 0.1    0.1    0.1    0.1    0.1    0.1    0.1    0.1    0.1
Cumulative Var 0.1    0.2    0.3    0.4    0.5    0.6    0.7    0.8    0.9
                Comp.10
SS loadings    1.0
Proportion Var 0.1
Cumulative Var 1.0
>

```

ii) *How many principal components would you suggest adequately contain the main sources of variation within the data.*

Looking at the scree plot, 3 or maybe 4 components (accounting for 84% or 89% of total variation). Ignore obvious kink at k=1

iii) *What features of the body sizes do the first three [four?] components reflect?*

It is maybe clearer to see what is going on if we suppress as many decimal places as possible and use a fairly high cutoff value (it isn't possible to round to zero digits so try with just one):

```
> print(body.pc$loadings, cutoff=0.1,digits=1)
```

Loadings:

```

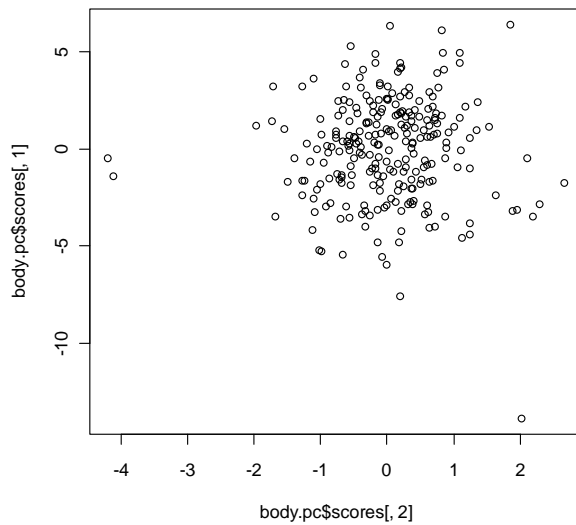
                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
neck    -0.3          -0.3    0.3          0.3    0.7    0.3
chest   -0.3    0.3          0.2   -0.4          -0.2    0.1   -0.5   -0.4
abdomen -0.3    0.4          0.2   -0.3   -0.1   -0.1          0.3    0.7
hip     -0.3    0.3    0.2   -0.1          -0.3    0.6   -0.6
thigh   -0.3    0.2    0.2   -0.4    0.3    0.1    0.3   -0.4   -0.5    0.2
knee    -0.3          0.3   -0.1    0.4   -0.4   -0.1    0.6
ankle   -0.2   -0.6    0.6          -0.4    0.2
biceps  -0.3          -0.3   -0.3          0.7   -0.5    0.2    0.1
forearm -0.3   -0.4   -0.6   -0.4   -0.3   -0.4
wrist   -0.3   -0.4   -0.1    0.6    0.4          -0.3   -0.4

```

Now it is easy to see that the first PC reflects variations in overall size of body, the second contrasts arm size (mostly) with body and leg size, the third contrasts leg with rest of the body and the fourth is body versus limbs. If we plot PC1 against PC2 (i.e. the scores of on first principal component against those on the second) with

```
> plot(body.pc$scores[,2],body.pc$scores[,1])
```



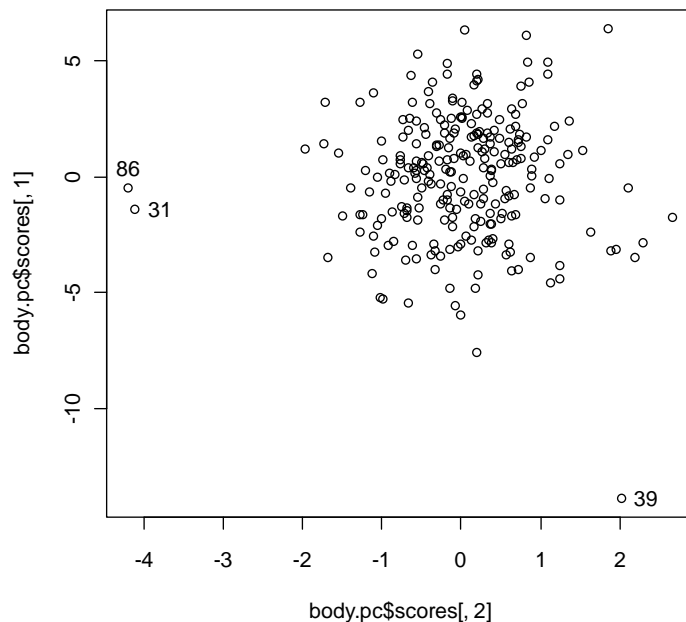


we can see one outlier at the bottom of the plot and two to the left. Noting the signs of the loadings on the first PC (vertical axis) we can see that the outlier at the bottom arises from a subject with large measurements (i.e. a large person). The two outliers to the left are from people of average size but with

proportionately well-developed arms by comparison with their legs.

Using `identify()` gives

```
> identify(body.pcscores[, 2], body.pcscores[, 1])
[1] 31 39 86
```



which reveals that these outliers are observations 39 (lower), 31 & 86 (rightmost).

If it is preferred to plot with the large people at the top of the plot then do

```
> plot(body.pcscores[, 2], -body.pcscores[, 1])
```



5) Calculate the principal components of the four measurements on Irises:

- i) using the 'ready made' facility for principal component analysis
- ii) by first calculating the covariance matrix and then looking at the eigenanalysis of the matrix.

```
> attach(irisnf)
> options(digits=2)
> iris.pc<-princomp(cbind(Sepal.l,Sepal.w,Petal.l,Petal.w))
> iris.pc
Call:
princomp(x = cbind(Sepal.l, Sepal.w, Petal.l, Petal.w))
```

```
Standard deviations:
Comp.1 Comp.2 Comp.3 Comp.4
  2.05   0.49   0.28   0.15
```

```
4 variables and 150 observations.
> summary(iris.pc)
Importance of components:
                Comp.1 Comp.2 Comp.3 Comp.4
Standard deviation  2.05  0.494  0.280  0.1542
Proportion of Variance  0.92  0.054  0.017  0.0052
Cumulative Proportion  0.92  0.978  0.995  1.0000
> iris.pc$loadings
```

```
Loadings:
      Comp.1 Comp.2 Comp.3 Comp.4
Sepal.l  0.361 -0.650  0.590  0.316
Sepal.w   -0.737 -0.592 -0.314
Petal.l  0.857  0.171   -0.480
Petal.w  0.358   -0.543  0.756
```

```
                Comp.1 Comp.2 Comp.3 Comp.4
SS loadings      1.00   1.00   1.00   1.00
Proportion Var   0.25   0.25   0.25   0.25
Cumulative Var   0.25   0.50   0.75   1.00
> screeplot(cbind(Sepal.l,Sepal.w,Petal.l,Petal.w))
> iris.cov<-var(cbind(Sepal.l,Sepal.w,Petal.l,Petal.w))
> iris.cov
```

```
      Sepal.l Sepal.w Petal.l Petal.w
Sepal.l  0.686 -0.041  1.27   0.52
Sepal.w -0.041  0.194 -0.33  -0.12
Petal.l  1.274 -0.329  3.12   1.30
Petal.w  0.516 -0.121  1.30   0.58
> eigen(iris.cov)
$values
[1] 4.228 0.246 0.079 0.024
```

```
$vectors
      [,1] [,2] [,3] [,4]
[1,] 0.361 -0.650 -0.59  0.32
[2,] -0.084 -0.737  0.59 -0.31
[3,] 0.857  0.171  0.08 -0.48
[4,] 0.358  0.073  0.54  0.76
```



Note that instead of `cbind(Sepal.l, Sepal.w, Petal.l, Petal.w)` we could use `irisnf[, -5]` which is the data set without column 5 which contains variety.

```
> cov(irisnf[, -5])
      Sepal.l Sepal.w Petal.l Petal.w
Sepal.l  0.686 -0.041  1.27  0.52
Sepal.w -0.041  0.194 -0.33 -0.12
Petal.l  1.274 -0.329  3.12  1.30
Petal.w  0.516 -0.121  1.30  0.58
```

Note that one of the covariances is negative and thus the first PC does not have loadings all of the same sign, though the negative covariance is very small by comparison with the others and so the corresponding coefficient is negligible and thus we can regard the first PC as reflecting variations in overall size.



Multivariate Data Analysis: Tasks for Week 3

Notes & Solutions

Note and MEMORIZE the interesting identity

$$|I_p + AB| = |I_n + BA| \text{ where } A \text{ is } p \times n \text{ and } B \text{ is } n \times p.$$

A key application of this result, which is used extensively later in this course, is when $n=1$. To evaluate $|I_p + xx'|$ where x is $p \times 1$ we have that this = $|I_1 + x'x|$ which is the determinant of a 1×1 matrix (i.e. a scalar) and so $= 1 + x'x = 1 + \sum x_i^2$.

A variant on the result is the following (where c and d are scalars):

$$|cI_p + dAB| = c^p |I_p + dAB/c| = c^p |I_n + dBA/c| = c^{p-n} |cI_n + dBA|$$

(noting that if Z is a $p \times p$ matrix and c a scalar then $|cZ| = c^p |Z|$)

In particular, $|cI_p + dxx'| = c^{(p-1)}(c + d\sum x_i^2)$ and **especially**, if $x = 1_p$ then $|cI_p + d1_p 1_p'| = c^{(p-1)}(c + pd)$ since $1_p' 1_p = p$.



1) Suppose the variance matrix takes the equicorrelation form

$$\mathbf{S}_{p \times p} = \sigma^2 \begin{pmatrix} 1 & \rho & \rho & \cdots & \rho \\ \rho & 1 & \rho & \cdots & \rho \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ \rho & & & \ddots & \rho \\ \rho & \rho & \cdots & \rho & 1 \end{pmatrix}. \text{ By writing } S \text{ in the form } S = aI_p + b1_p1_p', \text{ for}$$

appropriate a and b and using result above, show that if $\rho > 0$ then the first principal component accounts for a proportion $(1 + \rho(p-1))/p$ of the total variation in the data. What can be said about the other $p-1$ components? What can be said if $\rho < 0$? (but note that necessarily $\rho >$ some constant bigger than -1 which you should determine, noting that S is a correlation matrix)

We can see that $S = \sigma^2[(1-\rho)I_p + \rho J_p]$ where J_p is the $p \times p$ matrix with all entries = 1 and easy to see that $J_p = 1_p1_p'$ where 1_p is the unit p -vector with all entries = 1. Then, to obtain the eigenvalues we need $|S - \lambda I_p|$ and obtain the roots of this p -degree polynomial in λ . We could use row and column manipulation of the determinant but using the result above we have

$$\begin{aligned} |S - \lambda I_p| &= |\sigma^2[(1-\rho)I_p + \rho 1_p1_p'] - \lambda I_p| \\ &= |[(1-\rho)\sigma^2 - \lambda]I_p + \rho\sigma^2 1_p1_p'| = [(1-\rho)\sigma^2 - \lambda]^{(p-1)} [(1-\rho)\sigma^2 - \lambda + \rho\sigma^2 1_p'1_p] \\ &= [(1-\rho)\sigma^2 - \lambda]^{(p-1)} [(1-\rho)\sigma^2 - \lambda + p\rho\sigma^2] \text{ (noting that } 1_p'1_p = p) \end{aligned}$$

Thus the eigenvalues of S are $(1 + (p-1)\rho)\sigma^2$ and $(1-\rho)\sigma^2$ (the latter with multiplicity $(p-1)$). If $\rho > 0$ then the first of these is the largest (i.e. $\lambda_1 = (1 + (p-1)\rho)\sigma^2$).



If $\rho < 0$ then we must have $\rho > -(p-1)^{-1}$ since we must have $|S| > 0$:

if $\rho < -(p-1)^{-1}$ then one and only one eigenvalue is negative and since $|S| = \prod \lambda_i$ this would give $|S| < 0$.

When $\rho > 0$, the first principal component is a_1 where $Sa_1 - \lambda_1 a_1 = 0$, i.e. where $Sa_1 = (1 + (p-1)\rho)\sigma^2 a_1$ and $a_1' a_1 = 1$.

Easily seen that $a_1 = p^{-1/2} \mathbf{1}_p$ (i.e. proportional to the unit vector).

The other $(p-1)$ p.c.s are solutions of $Sa - (1-\rho)a = 0$ with $a'a = 1$ (normalizing constraint) and $a' \mathbf{1}_p = 0$ (orthogonality with a_1) (i.e. $\sum a_j^2 = 1$ and $\sum a_j = 0$) and there are **infinitely many possibilities**. One possible set is proportional to $(1, -1, 0, 0, \dots, 0)'$; $(1, 1, -2, 0, 0, \dots, 0)'$; $(1, 1, \dots, 1, -(p-1))'$.

Note: This example explains intuitively why the first principal component of a data set consisting of dimensional measurements on physical objects is often a measure of overall size: generally, if one of the objects is big then **all** of its dimensions will be big (presuming that the objects are more or less the same shape). This means that, generally, the measurements of all the dimensions will be *positively correlated* with each other. Consequently, the correlation (or covariance) matrix will be approximately like the equicorrelation matrix and so the first p.c. will be approximately proportional to the unit p -vector and so the score of any datum on the first p.c. will be proportional (approx) to the sum of its individual components). In fact the Perron-Frobenius theorem states that if all the elements of a (not necessarily symmetric) matrix are strictly positive then there is a unique positive eigenvalue corresponding to an eigenvector which can be chosen to have all positive elements and so could be interpreted as a weighted average of all measurements. The closer the correlations are in value the closer the coefficients of the first eigenvector are to a common multiple of the unit p -vector. If a small number of correlations are negative then it is often the case that the 2nd or 3rd (or.....) PC is size measure. Note also that there can be only one PC at most which is a weighted average of all variables since PCs are necessarily orthogonal.



- 2) If the variance matrix takes the form $S = \alpha I_p + \beta z z'$ where z is a p -vector, show that z is an eigenvector of S . Under what circumstances is Z proportional to the first principal component of the data?

$$S = \alpha I_p + \beta z z' \quad \text{so} \quad S z = (\alpha I_p + \beta z z') z = \alpha z + \beta z z' z = \alpha z + \beta z (z' z)$$

then, noting $z' z$ is a scalar and so commutes with z ,

$= (\alpha + \beta z' z) z = \lambda z$ where $\lambda = \alpha + \beta z' z$. So z is an eigenvector of S with eigenvalue $\alpha + \beta z' z$. Thus $z / (z' z)^{1/2}$ is the first p.c. if $\alpha + \beta z' z$ is the largest eigenvalue. The other $p-1$ eigenvalues are easily seen to be α and so for z to be the first we need $\beta > 0$ (since $z' z = \sum z_i^2 > 0$).

- 3) If the variance matrix takes the form (with $\alpha > 0$)

$$S = \begin{pmatrix} 1 + \alpha & 1 & \beta \\ 1 & 1 + \alpha & \beta \\ \beta & \beta & \alpha + \beta^2 \end{pmatrix} \quad \text{find the first principal component and show}$$

that it accounts for a proportion $(\beta^2 + \alpha + 2) / (\beta^2 + 3\alpha + 2)$ of the total variation.

$S = \alpha I_3 + \gamma \gamma'$ where $\gamma = (1, 1, \beta)'$ (notice that the diagonal of S contains $+\alpha$ in each entry so subtracting αI_3 leaves a matrix which is easier to make an intelligent guess at factorizing).

$|S - \lambda I_3| = |(\alpha - \lambda) I_3 + \gamma \gamma'| = (\alpha - \lambda)^2 (\alpha - \lambda + \gamma' \gamma) = (\alpha - \lambda)^2 (\alpha - \lambda + 2 + \beta^2)$ and so the eigenvalues of S are $\beta^2 + \alpha + 2$ and α (twice). The largest must be the first (since $\beta^2 + 2 > 0$) and so accounts for a proportion $(\beta^2 + \alpha + 2) / (\beta^2 + 3\alpha + 2)$ of the total variation.



- 4) Referring to Q3 on Task Sheet 2, examination results in five mathematical papers, some of which were 'open-book' and others 'closed-book', what interpretations can you give to the principal components? .

The principal components can be read from the eigenvectors calculated in Q3 or easily from

```
> options(digits=1)
> prcomp(scor)$rotation
      PC1   PC2   PC3   PC4   PC5
mec -0.5 -0.75  0.3 -0.296 -0.08
vec -0.4 -0.21 -0.4  0.783 -0.19
alg -0.3  0.08 -0.1  0.003  0.92
ana -0.5  0.30 -0.6 -0.518 -0.29
sta -0.5  0.55  0.6  0.176 -0.15
>
```

PC1 is a measure of overall ability across the five mathematical subjects, with low scores indicating high marks (not signs of PCs are arbitrary) . PC2 is a contrast between Pure&Statistics versus Applied Mathematics, with high scores indicating higher marks in Pure and Statistics than in Applied. PC3 is a contrast of the more applied subjects of Statistics and Mechanics versus the more theoretical Pure and Vectors, with high scores indicating preference for the applied. PC4 is primarily vectors versus analysis and PC5 is primarily ability at Algebra.

