Lecture 9

principle components analysis

Multivariate statistics: PCA and linear discriminant analysis

Multivariate analysis

- Multivariate statistics are techniques applied to datasets with multiple measurements

- Some techniques reveal structure underlying *variables*
  - *Principal component analysis* reduces dimensionality or redundancy of variables
  - *Exploratory factor analysis* investigates existence of 'latent' or underlying factors

Multi dimensions reduced to one

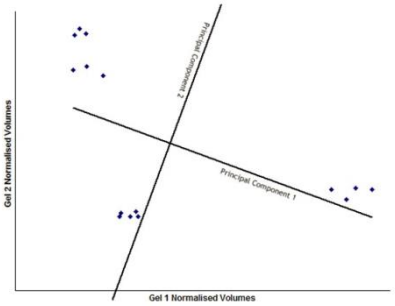
Multi reduced to two

- Others reveal structure underlying *cases*
  - *Cluster analysis* groups cases through measures of similarity
  - *Discriminant analysis* identifies variables that delimit groups

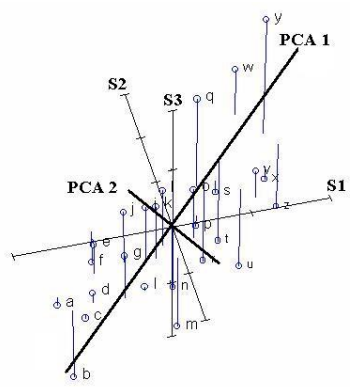
Clustered Into groups

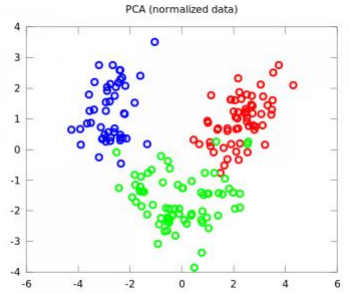
Principal component analysis (PCA)

- PCA checks whether information in a set of variables can be expressed through a reduced set of new variables or *principal components*



- PCs also allow *a posteriori* grouping of cases based on how they vary across the new components



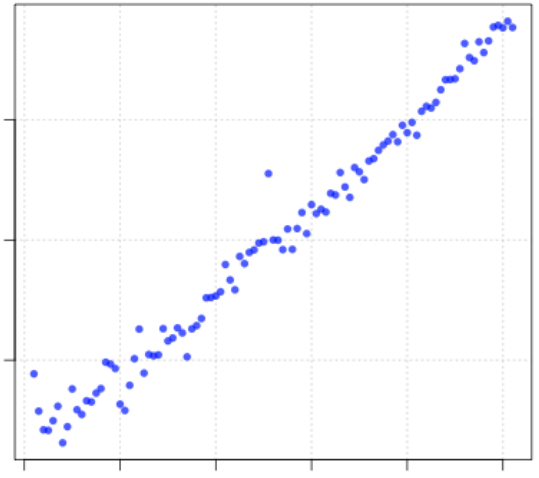


PCA: intuitively

- You have two measurements from 100 trousers
  - **leg length *x*, ankle width *y***;
  - Plotting them shows a straight line

- So how to describe the differences across trousers (or to describe each one)?
  - We can use the two measurements
  - or we define the diagonal line as a *new* axis

New axis: New principle components, indicating original elements x and y (or as many as possible, technically)



- The new axis is equivalent to a 'principal component'
  - new axis or variable is 'trousers size', (i.e. size 32 or size 10), although size was not an original measurement
  - new size axis reduced dimensionality (you don't need to ask for trousers with leg length *x* and ankle width *y*; you ask for size 10)

- ps. there is a second axis but it explains little variation
  - (possibly random error in measurements)

PCA: intuitively

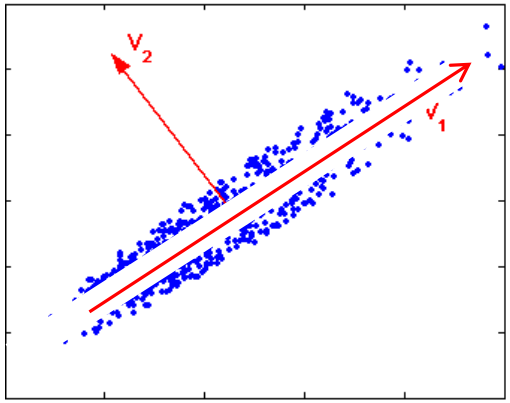
- Now assume that in addition to different ‘sizes’ your sample has two types of trousers: bootleg and skinny

- You expect two parallel lines (‘bootleg’ as the higher line)

- You can still define two new axes
  - the first is still ‘trousers size’
  - the second axis reflects ‘trousers shape’ (ankle width for a given leg length), separating bootleg from skinny trousers

- In this case, there is no reduction of dimensionality, but the two new axes ‘size’ (‘PC1’) and ‘shape’ (‘PC2’) provide a better description of variation in trousers
  - i.e. you ask for ‘size 10, skinny’ instead of ‘leg length *x*, ankle width *y*’)

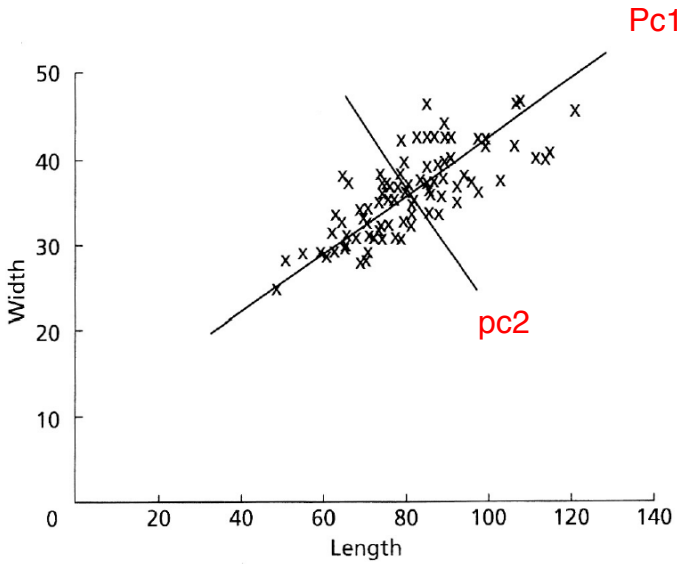
both two lines are informative, first shows basic info (a, b) and second additional info



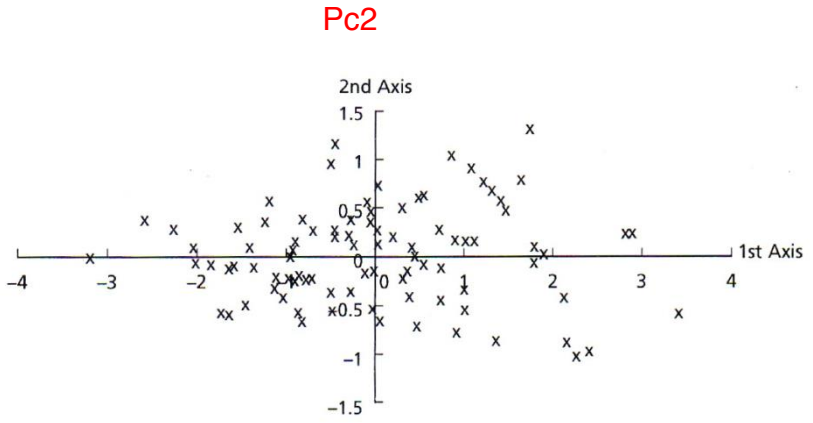
PCA: intuitively

- Each PC will account for a fraction of original variation (variance) in the sample; therefore, they should be
  - mutually independent (orthogonal)
  - ordered by importance (i.e. fraction of variance explained)

original graph

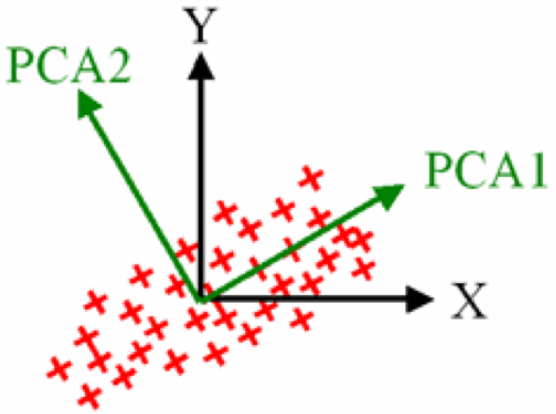


New:



PCA: geometry

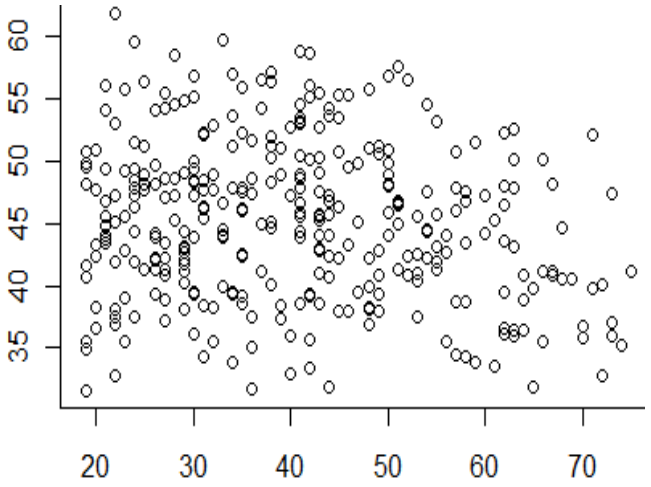
- PCs are obtained through rotation of original axes
  - *x* and *y* are orthogonal (perpendicular), therefore PCs remain orthogonal



- Axes rotation is done so that first PC accounts for most variation and so on
  - i.e. PC1, PC2, PC3 are axes of decreasing importance

- Procedure only works if original variables are correlated
  - but PCs themselves are uncorrelated

must be independent



PCA: mathematics

The number of pcs are the number of variables *2

1pc = 2 variables

- A principal component is a linear combination of existing variables

- PC1 will be:

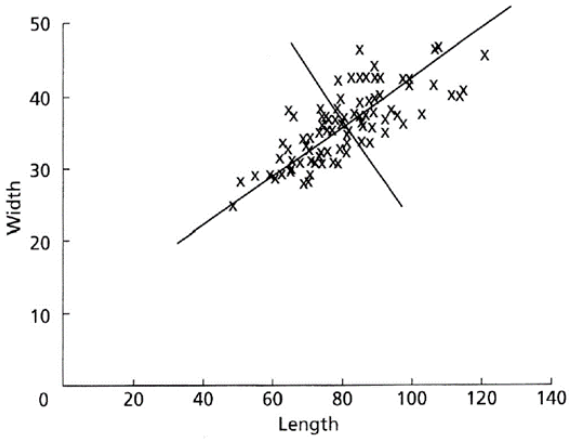
$$Y_{1,1} = b_{1,1}X_1 + b_{2,1}X_2 \ldots + b_{n,1}X_n$$

where

- Y_1 = *score* of case 1 on PC1, i.e. its 'measurement' on the new axis
- $b_{i,1}$ = *loading* of variable X_i on PC1 (and correlation between PC1 score and X_1 if values are standardised as *z*-scores)
- X_i=original measurements of case 1

And PC2 will be

$$Y_{1,2} = b_{1,2}X_1 + b_{2,2}X_2 \ldots + b_{n,2}X_n$$



Calculation of PCs

- file *lifehistory*
  - primate data (log values, base 10) on 4 variables: *weight*, *brain size*, *lifespan* and *age at first reproduction*

- PCA is useful only if variables show underlying correlations
  - analysis starts with a *variance-covariance matrix*, or calculation of covariances between all variables
  - note: to calculate covariance (and correlation) matrix and run PCA, data file must have neither NAs or non-numerical variables (group, species)

- To create a file including the four selected variables:

  > lifehistory2 <- subset(lifehistory, select= c(lifespan, weight, brain, firstrep))
- or just exclude *species* and *group* columns (notice "-" before c)

  > lifehistory2 <- subset(lifehistory, select= -c(group, species))

- To eliminate NAs

  > lifehistory <- lifehistory[complete.cases(lifehistory),]
- (note: to select complete cases based on one variable only:

  > newfile <- oldfile[complete.cases(oldfile$variable)]

Variance-covariance matrix

- Now we calculate all covariances between *lifehistory2* variables

```
> cov(lifehistory2)
```

	lifespan	weight	brain	firstrep
lifespan	0.03727196	0.09491868	0.08595568	0.03403697
weight	0.09491868	0.42676653	0.34460016	0.13945175
brain	0.08595568	0.34460016	0.29760857	0.12096556
firstrep	0.03403697	0.13945175	0.12096556	0.06359508

- Covariance of *x* and *x* is the variance of *x*; thus, diagonal of our square 4 x 4 matrix shows variances of *weight*, *brain*, *lifespan* and *first reproduction*
  - hence 'variance-covariance matrix'

Correlation matrix

- However, covariances reflect magnitude of variation; it is thus preferable to use a ***correlation matrix***
  - **remember**: correlation is covariance after variables are standardised (to mean=0 and sd=1)

> cor(lifehistory2)

	lifespan	weight	brain	firstrep
lifespan	1.0000000	0.7526021	0.8161327	0.6991141
weight	0.7526021	1.0000000	0.9669357	0.8464807
brain	0.8161327	0.9669357	1.0000000	0.8792802
firstrep	0.6991141	0.8464807	0.8792802	1.0000000

- **Important**: in the correlation matrix, standard deviation (and thus variance) of all variables is 1, and total variance will be *n,* i.e. variable number

Each components will contribute to one

  - In the example, total variance=4

Eigenvectors and eigenvalues

- Now we need to understand a property of square matrices (such as our correlation matrix):

- If A is a square matrix (n rows by n columns), then there are n pairs of numbers λ and vectors v such that:

$$Av=\lambda v$$

  - vector v (n rows and a single column) is called an *eigenvector*
  - number λ is called an *eigenvalue*

- What does it mean?
  - if you multiply square matrix A by v, you get the same vector v times λ ('eigen' is German for 'same')
  - multiplying or dividing an eigenvector by another number does not change its corresponding eigenvalue

Example

- Square matrix *A* has 2 eigenvectors; *v* is one of them

A × v = λ × v

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

- If I multiply *v* by 2,
  - resulting vector is still an eigenvector (with a *v* basis; it does not count as a new one)
  - its eigenvalue is still λ=4

$$2 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 6 \\ 4 \end{pmatrix} = \begin{pmatrix} 24 \\ 16 \end{pmatrix} = 4 \times \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

- To find the two *v*, λ pairs solve equation
  - $Av=\lambda v$
  - $Av - \lambda Iv = 0$
  - $(A - \lambda I)v = 0$

(A = n x n matrix, I = identity matrix)

- determinant of $(A - \lambda I)$ must be 0
- i.e. you get a quadratic equation and two λs
- for each λ, a *v* can be calculated

PCA

- So what is PCA? It is an analysis of the eigenvectors and eigenvalues of the correlation (or variance-covariance) matrix *n* x *n*, where *n* is the number of variables in the dataset:

- If each PC is defined as $Y_1 = b_1X_1 + b_2X_2 \ldots + b_nX_n$

then

  - **eigenvector 1 contains the variable loadings b_i on PC 1**
  - **eigenvalue 1 is variance explained by PC 1**

…and so on until the last (nth) PC

- For *n* variables, we obtain *n* PCs (=eigenvectors)
  - PCA is about rotating and redefining axes, you start and end with the same number of axes or variables

PCA in R

- To run PCA, functions *prcomp* or *princomp* can be used
- We will be using *prcomp*

Calculation based on correlation matrix

```
> pca1 <- prcomp(lifehistory2, scale.=T, retx=T)
> pca1
Standard deviations:
[1] 1.8673661 0.5681131 0.4068723 0.1569917

Rotation:
             PC1        PC2        PC3        PC4
lifespan 0.4663392  0.8530973  0.1961560 -0.1275759
weight   0.5132214 -0.2365816 -0.5756064 -0.5910248
brain    0.5264700 -0.1008899 -0.2995261  0.7892621
firstrep 0.4918952 -0.4539553  0.7351764 -0.1071420
```

- We want PCA to be based on a correlation matrix (rather than covariance); for that, we need the arguments:

- scale.=T
  - scales variance to 1
- retx=T
  - sets means to 0

PC1 = 0.47*lifespan + 0.51*weight + 0.52*brain + 0.49*firstrep >>>> the value is located in new axis PC1

Output

```
> pca1 <- prcomp(lifehistory2, scale.=T, retx=T)
> pca1-
Standard deviations:
[1] 1.8673661 0.5681131 0.4068723 0.1569917

Rotation:
             PC1        PC2        PC3        PC4
lifespan 0.4663392  0.8530973  0.1961560 -0.1275759
weight   0.5132214 -0.2365816 -0.5756064 -0.5910248
brain    0.5264700 -0.1008899 -0.2995261  0.7892621
firstrep 0.4918952 -0.4539553  0.7351764 -0.1071420
```

- Rotation: the 4 eigenvectors or PCs
- Column PC1: the *b* loadings of PC1
- PC1 is then:
  - Y_1 = 0.47(lifespan) + 0.51(weight)+ 0.529(brain)+ 0.49(firstrep)
  - all *b* loadings are positive
  - So PC1 seems to be a 'size' axis
- But on PC2, *weight*, *brain* and *lifespan* have negative loadings
  - (we'll interpret PCs later)
- If we are going to interpret PC1 as 'size', it should strongly correlate with variable *weight*

scale variable and scale effect

```
>cor(pcscores$PC1, lifehistory$weight)
[1] 0.9583722
```

Exercise:

PC1 = -0.63*lifespan - 0.59*schoolingyears - 0.54*GNI

- Run a PCA using the hdr dataset, and then write down the equation describing PC1 (i.e. Y1 = b1X1 + b2X2+…etc)
- What is the score of case 1 (first row) on PC1?

-25739.31

PC retention criteria

- Since PCs decrease in order of importance, do we need all of them?

Not necessarily;

- if you start with 20 variables you end up with 20 PCs, but we hope that a few of them (the most important ones) may be enough to explain most variation in sample
  - this is the point of reducing dimensionality!

- There are various criteria for retention of PCs based on their eigenvalues or proportion of variance explained:
  - $\lambda > 1$
  - scree plots
  - individual/cumulative variance thresholds
  - 'interpretability'

Rule 1: $\lambda > 1$

- A first criterion is only to keep PCs with $\lambda > 1$
  - rationale: variances of each original variable are scaled to one; it makes no sense to use a PC explaining less variance than an original variable

- To show eigenvalues (=variance, or sd^2):

```
> pca1$sdev^2
[1] 3.48705601 0.32275255 0.16554505 0.02464639
```

- Confirming that total variance is n=4:

```
> sum(pca1$sdev^2)
[1] 4
```

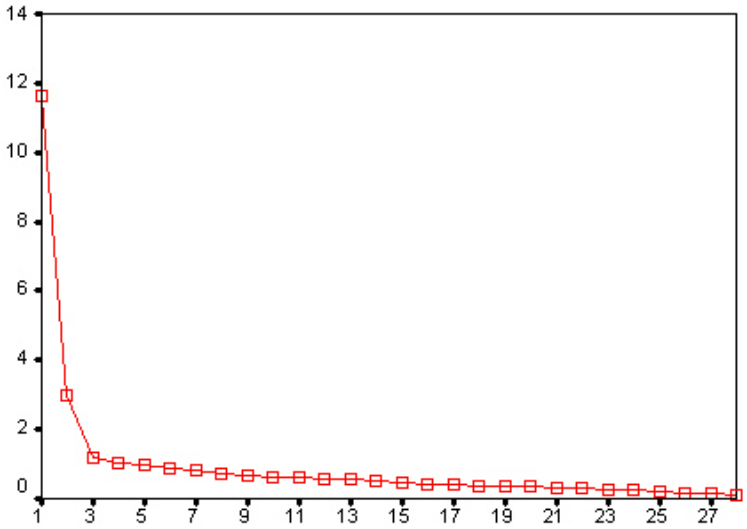
- Based on this criterion, we should retain only PC1 ($\lambda = 3.487$ = variance explained by PC1)

Rule 2: Scree plots

- We can plot eigenvalues λ from each PC and look at the pattern

- (ps. ‘scree’ is the debris accumulated at the base of a cliff)

- A dip in plot suggests cut-off point:
  - PCs before dip should be kept
  - the others (‘scree’), discarded

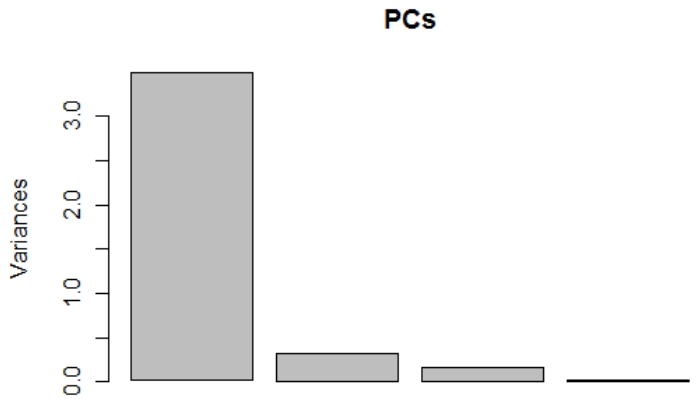


Scree plots

- Scree plot as bars:

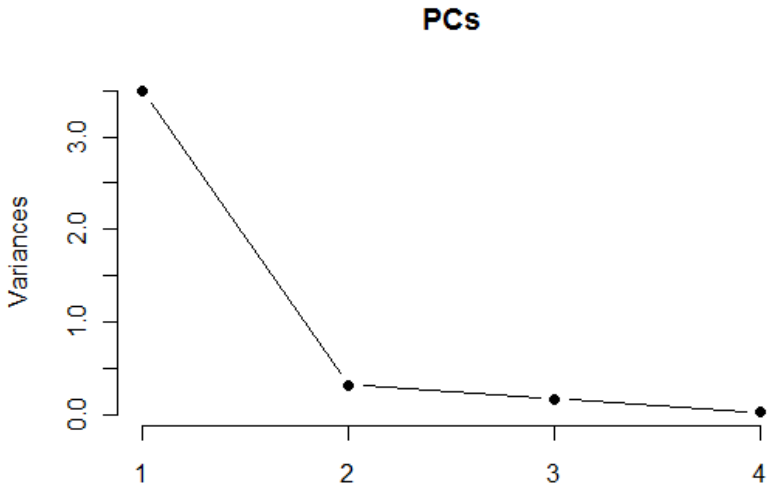
```
>screeplot(pca1, main="PCs", pch=16)
```

pch determines the type pf symbols, run ?pch



- Scree plot as line:
  - add *type="line*

```
>screeplot(pca1, main="PCs", type="line", pch=16)
```



- ps. all other graphical parameters apply (colour, x and y labels etc.)

PC1 above the rest

- Conclusion: only PC1 should be retained (PCs 2, 3 and 4 are 'scree')

Rule 3: fraction of variance explained

What kind of PCs could be kept?

Bottom line, requirements differ

- Another rule is to retain PCs only if they explain a given percentage of variance
  - *individually*: keep PCs that explain >10% or 15% of total variance
    - but this idea somehow clashes with the first rule ($\lambda>1$)
  - *as a set*: keep PCs that *cumulatively* explain >80% or 90%
    - this is more useful as a supplementary rule

- To check for individual and cumulative percentages of variance:

```
> summary(pca1)
Importance of components:
                          PC1     PC2     PC3     PC4
Standard deviation     1.8674 0.56811 0.40687 0.15699
Proportion of Variance 0.8718 0.08069 0.04139 0.00616
Cumulative Proportion  0.8718 0.95245 0.99384 1.00000
```

- PC1 explains 87% of all variance and should be kept
- Or if you want to be very safe, keep PC1 and PC2 (that cumulatively explain 95% of variance)
  - but PC2 explains less than 10% individually (and is in the scree); rules must be used together

Exercise:

Manually calculate the proportion of variance explained by each PC (i.e. why does PC1 explain 87.2% of variance?

Rule 4: ‘Interpretability’

- As we’ve seen, there is no test for identifying which PCs should be accepted

- We should use the rules together; they often imply different choices
  - e.g a PC may explain <10% of total variance (‘don’t keep it’), but may be required to explain >80% of total variance cumulatively (‘keep it’)

- This means that a reason for keeping a PC is its potential ‘meaning’
  - i.e. if I think that variation in sample is basically size and dimorphism, I’m happy to keep PC1 as ‘size’ (if all variables have positive loading on it), and PC2 as ‘sex’ (if variables loading on it are known/expected to vary between sexes)

Interpretability

```
> pca1 <- prcomp(lifehistory2, scale.=T, retx=T)
> pca1
Standard deviations:
[1] 1.8673661 0.5681131 0.4068723 0.1569917

Rotation:
              PC1        PC2        PC3        PC4
lifespan 0.4663392  0.8530973  0.1961560 -0.1275759
weight   0.5132214 -0.2365816 -0.5756064 -0.5910248
brain    0.5264700 -0.1008899 -0.2995261  0.7892621
firstrep 0.4918952 -0.4539553  0.7351764 -0.1071420
```

- Back to our PCs and loadings:
- PC1:
  - all variables have positive loadings
  - i.e. PC1 represents 'something' that is large when everything else is
  - this is a 'size' or 'scale' component
- PC2:
  - strong positive loading on lifespan, strong negative loading on firstrep
  - weak loadings (near 0) from weight and brain
    - i.e. a PC based on timing variables, not size variables
  - PC2 score is higher for species where lifespan is long but first reproduction is early
- In doubt, perhaps wiser not to keep PCs with no straightforward interpretation

Grouping cases

- We can also look at grouping patterns formed by PCs

- We can extract PC scores (Y_1, Y_2,…) for each case
  - remember: each case has *n* variables, *n* new PCs and *n* new measurements ('PC scores')

- Extracting PC scores into a matrix: command *$x*

  > matrixpc <- pca1$x

- We saved matrix with scores; now we convert matrix into data frame

  > pcscores <- data.frame(matrixpc)

- Now we add *pcscores* (with the new variables or PCs) to our original *lifehistory* dataset (which still has *group* and *species* columns) using the function data.frame, which creates new datasets

(ps. We are doing this to file *lifehistory*, not *lifehistory2*)

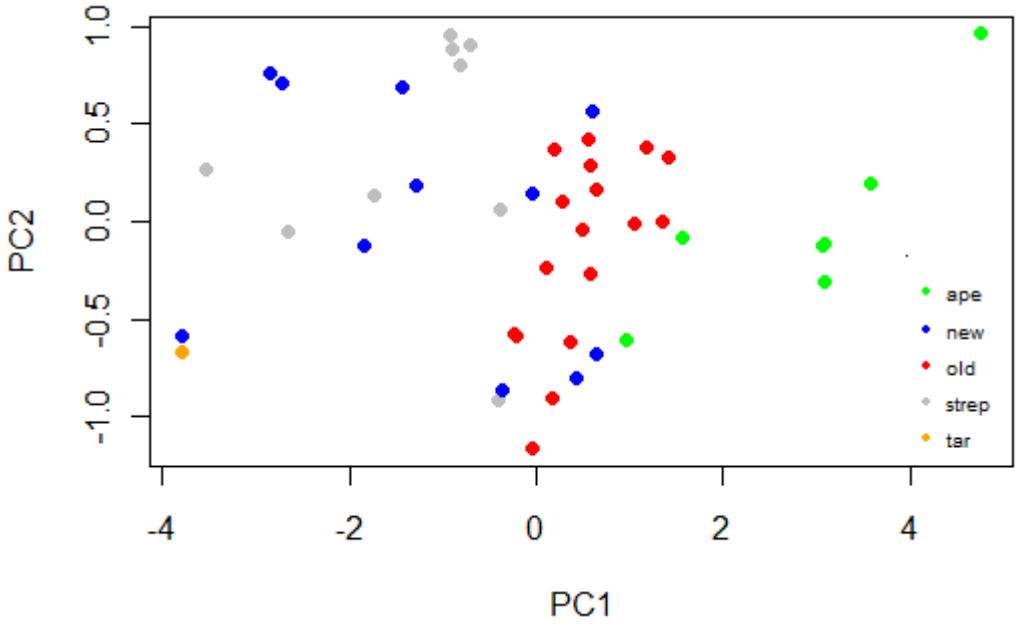
> lifehistorypc <- data.frame(lifehistory, pcscores)

Grouping cases

- Now we can plot PCs against each other and label groups

- PC1 x PC2 by each of 5 primate groups

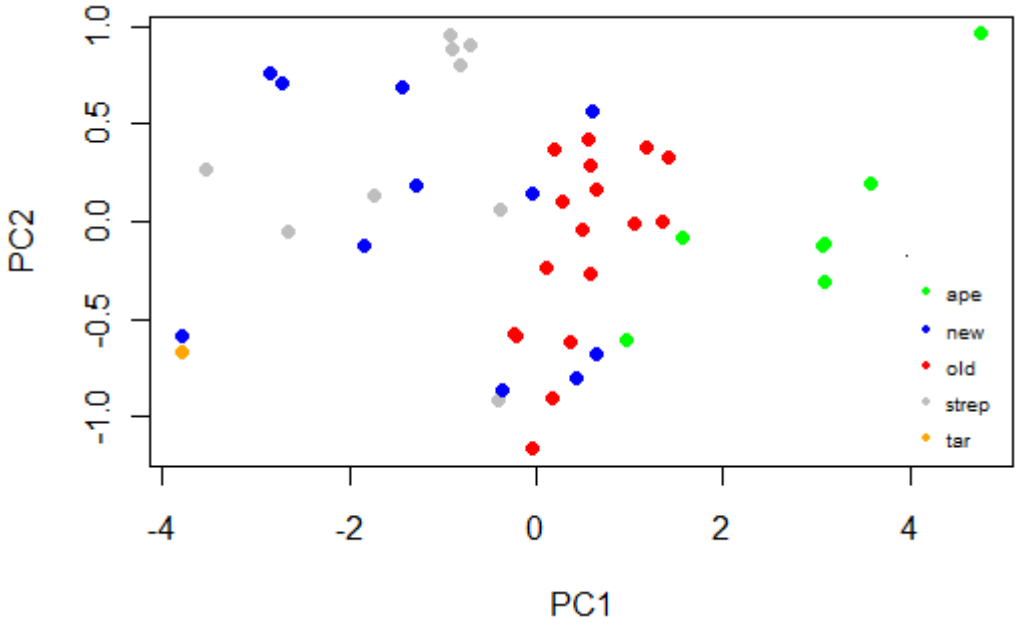
```
> plot(PC2~PC1, pch=16,
col=c("green", "blue", "red", "grey",
"orange")[group],
  data=lifehistorypc)
```



- Remember that *R* reads factor levels *alphabetically*: apes, New World monkeys, Old World monkeys, strepshirrhines, tarsier

Grouping cases

- PC1 increases from tarsier and strepsirrhines to New and Old World monkeys and finally apes
  - it makes sense to interpret it as ‘body size’
- PC2 does not show primate group patterning
  - this does not necessarily mean that interpreting PC2 as ‘adult lifespan’ or ‘reproductive span’ is wrong



Identifying cases

- We may want to look at specific cases on the plot

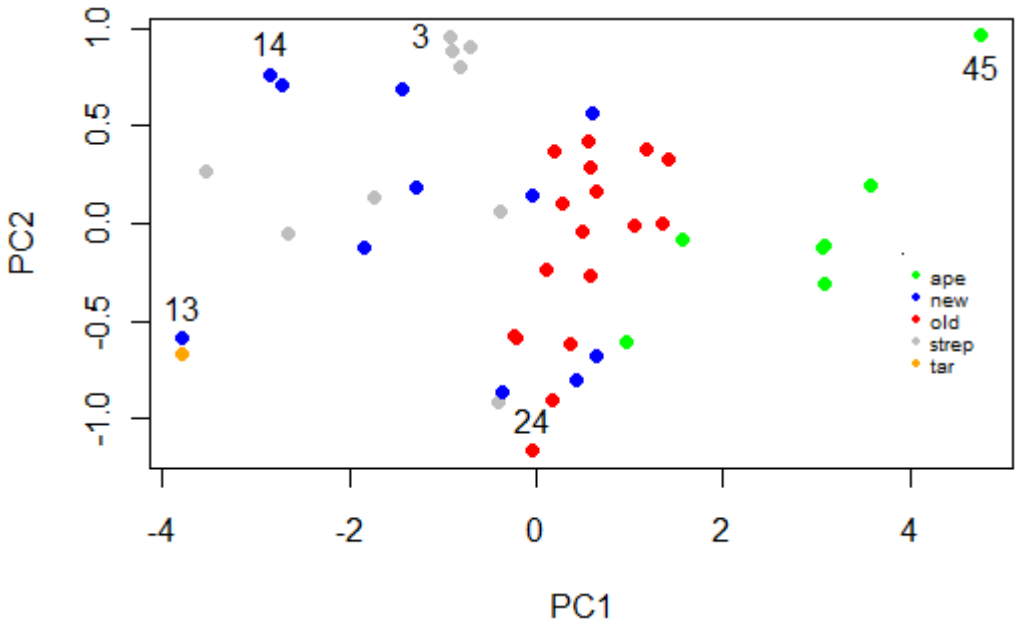
- Done with function *identify*
  - and right-clicking point of interest

> identify(PC2~PC1)

- Using *lifehistory* file, we can identify the species:
  - 3 V. variegata
  - 13 C pygmaea (pygmy marmoset)
  - 14 C. jacchus
  - 24 C. aethiops
  - 45 H. sapiens

  - (note: this plot is based on line number, not column *row.names*)

- Try plotting PC3~PC1, PC3~PC2



Notes

- It may be relevant to estimate correlations between PC scores and each variable; this may show whether variable has a ‘significant’ loading on a PC
  - some authors argue that PCs with less than 3 significant variable loadings should not be kept

- PCA is most useful when dataset includes many variables

- Remember to use a correlation matrix rather than a covariance matrix, unless all variables have the same scale and variance (which is very rare!)

Exercise:

- Based on your PCA using the hdr dataset, how many PCs should you keep?
- What would you call PC1?