Chapter 7 Principal component analysis (PCA)

Large datasets with many variables have become across different disciplines. To effectively interpret these datasets, it’s important to use methods that can significantly reduce their dimensionality while retaining the majority of the information. Principal component analysis (PCA) stands out as one of the most methods for the purpose. PCA operates on a straightforward principle: reduce the dataset's dimensionality while maintaining maximal 'variability’ as much as possible.

The primary objective of PCA is to produce an optimal data summary using a much smaller number of Principal components (PCs). The first principal component represents the direction of maximum variability (covariance) within the data. The second principal component is the subsequent orthogonal (uncorrelated) direction of greatest variability. Therefore, the process involves eliminating all variability along the first component and subsequently identifying the next direction of greatest variability. This iterative procedure continues until all relevant principal components have been identified.

The math formula for PCA:

$$\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix}$$

$$\begin{aligned} Y_1 &= \phi_{11}X_1 + \phi_{12}X_2 + \cdots + \phi_{1p}X_p \\ Y_2 &= \phi_{21}X_1 + \phi_{22}X_2 + \cdots + \phi_{2p}X_p \\ &\vdots \\ Y_p &= \phi_{p1}X_1 + \phi_{p2}X_2 + \cdots + \phi_{pp}X_p \end{aligned}$$

$$\text{var}(Y_i) = \sum_{k=1}^{p}\sum_{l=1}^{p} \phi_{ik}\phi_{il}\sigma_{kl} = \phi_i'\Sigma\phi_i$$

PCs are linear combinations of the original variables

$$\begin{gathered} \mathbf{z_1} = v_{11}\mathbf{x_1} + \cdots + v_{p1}\mathbf{x_p} \\ \mathbf{z_2} = v_{12}\mathbf{x_1} + \cdots + v_{p2}\mathbf{x_p} \\ \vdots \\ \mathbf{z_k} = v_{1k}\mathbf{x_1} + \cdots + v_{pk}\mathbf{x_p} \end{gathered}$$

In matrix form:

$$\mathbf{Z} = \mathbf{XV}$$

PC1 The first principal component is the linear combination of x-variables that has maximum variance (among all linear combinations). The first pricinpal component is the corresponding eigen vector of the largest eigenvalue for the covariance matrix. It accounts for as much variation in the data as possible if only one component is used.

$$Z_1 = \phi_{11} X_1 + \phi_{21} X_2 + \cdots + \phi_{p1} X_p$$

$$\max_{\phi 11, \phi_{21}, \ldots, \phi_{p1}} \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2$$

$$\text{s. t.} \sum_{j=1}^{p} \phi_{j1}^2 = 1$$

Applications of PCA: Image Compression: it can be used to compress images by reducing their dimensionality while preserving most of the essential information. This is particularly useful in fields like image processing and computer vision.

Bioinformatics: In genomics and proteomics, where datasets often have a high dimensionality (e.g., gene expression data), it can help in identifying patterns and reducing noise, aiding in tasks like clustering, classification, and visualization of biological data.

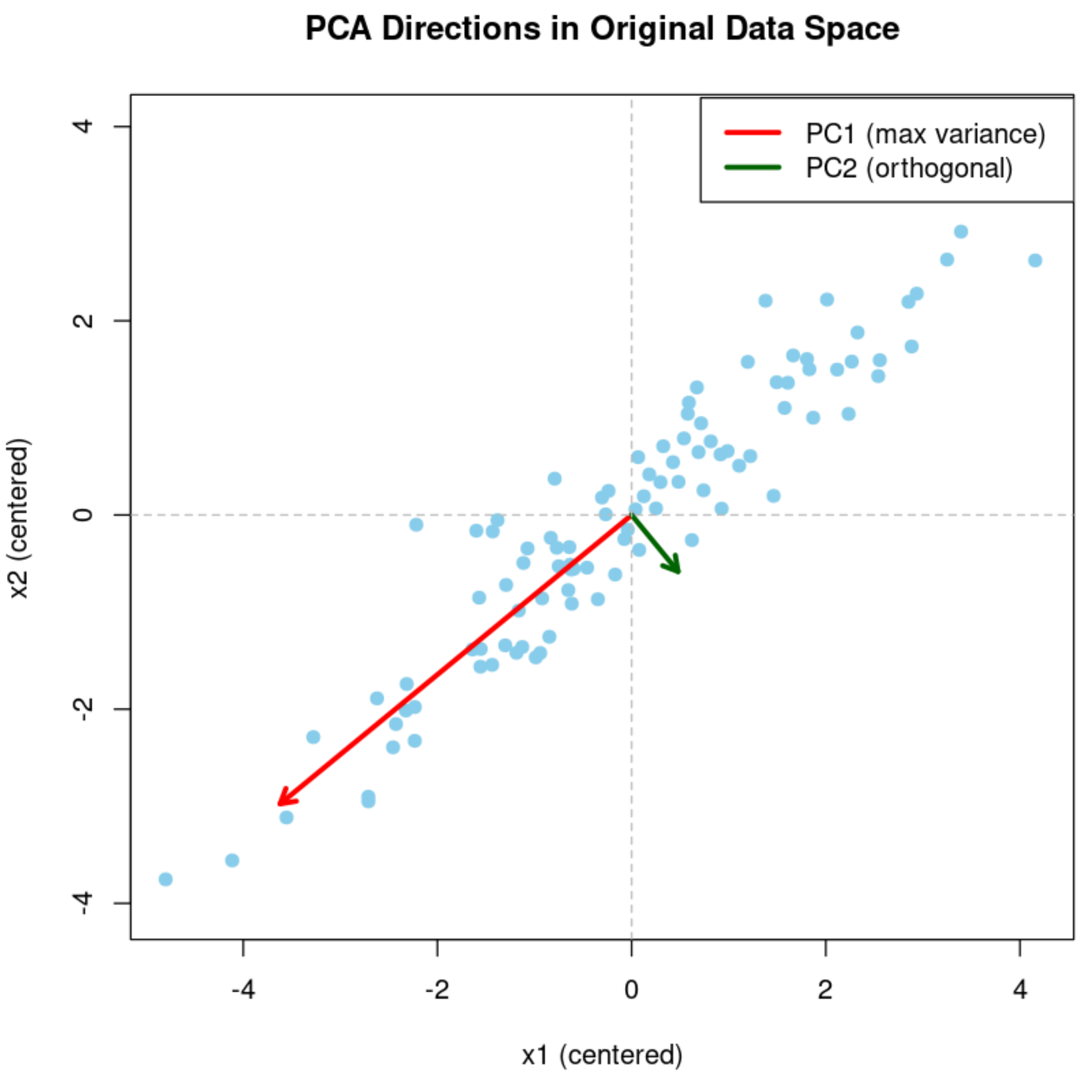
Finance: it is widely used in financial modeling and portfolio management. It helps in analyzing the correlations between different financial assets and in constructing diversified portfolios that maximize returns while minimizing risk.

Signal Processing: It can be applied to analyze signals and extract relevant features, making it useful in fields such as telecommunications, speech recognition, and seismic analysis.

Chemometrics: In chemistry, PCA is used for analyzing spectroscopic data and identifying patterns in chemical compounds. It helps in reducing the complexity of data while retaining the most relevant information for further analysis.

Neuroscience: It is utilized in analyzing neuroimaging data (e.g., fMRI, EEG) to identify patterns of brain activity and to reduce the dimensionality of data for visualization and interpretation.

We would like to use a few Pcs to capture most of the variation of all variables. Totally , if there are p variable, we can have p principal components . We would like not to keep all p Pcs.



In [ ]:

```

```

In [27]:

```
#Example:   the sparrow bird data. Run a principle component analysis.
#https://search.r-project.org/CRAN/refmans/Sleuth2/html/ex2016.html

#Load the file in datasciences.tamucc.edu
loc1 ="~//distribute_files//data//Bumpus_sparrows.csv"
sparrow1.dat = read.csv( loc1,  header = 1 )
#head(sparrow1.dat)

#Data matrix
dat = sparrow1.dat[, -1]
head(dat)

#To run PCA analysis

## R function prcomp
#PCA on the raw data
#Data without any scaling
fit = prcomp(dat)
```

```
#Extract the first component
PC1 = fit$rotation[,1]
PC1

#How much information (percentage of total information is contained in the first pr

#Based on the correlation matrix,  data with scaling
```

A data.frame: 6 × 5

	Total_length	Alar_extent	L_beak_head	L_humerous	L_keel_sternum
	<int>	<int>	<dbl>	<dbl>	<dbl>
1	156	245	31.6	18.5	20.5
2	154	240	30.4	17.9	19.6
3	153	240	31.0	18.4	20.6
4	153	236	30.9	17.7	20.2
5	155	243	31.5	18.6	20.3
6	163	247	32.0	19.0	20.9

Total_length: 0.536500524536968 **Alar_extent:** 0.829015353815653 **L_beak_head:** 0.0964961476899139 **L_humerous:** 0.0743521915425749 **L_keel_sternum:** 0.100304413678004

The first Principal component: 0.536xTotal_length + 0.829xAlar_extent + 0.0964x L_beak_head + 0.07435xL_humerous + 0.100304xL_keel_sternum

In [26]:

```
# PCA on the data after scaling
library(tidyverse)

#Transformed data : all varialbes have variance of 1; the mean of 0.
s.dat = dat %>%  mutate_all(scale)

#To run PCA analysis on the scaled data

fit.s = prcomp(s.dat)

#Extract the first component
PC1 = fit.s$rotation[,1]
round(PC1,3)
#The first PC is
#PC1 = Total_lengthx 0.452 + Alar_extentx0.462 + L_beak_headx 0.451 + L_humerousx 0

PC2 = fit.s$rotation[,2]

#The second principal component
PC2

###Simple version, without data transformation,  PCA on the scaling data
```

```r
fit = prcomp(dat, center = TRUE, scale = TRUE)
round(fit$rotation[,1],3)
```

Total_length: 0.452 **Alar_extent:** 0.462 **L_beak_head:** 0.451 **L_humerous:** 0.471
L_keel_sternum: 0.398

Total_length: -0.0507213669225457 **Alar_extent:** 0.299563545556606 **L_beak_head:** 0.324572424710128 **L_humerous:** 0.184684031624153 **L_keel_sternum:** -0.8764893465935

Total_length: 0.452 **Alar_extent:** 0.462 **L_beak_head:** 0.451 **L_humerous:** 0.471
L_keel_sternum: 0.398

The optimal number of components

Determining the Number of Principal Components

Determining the **optimal number of principal components** is not a straightforward task. It often involves a degree of subjectivity and depends on the goals of the analysis, the structure of the data, and the desired balance between simplicity and accuracy. Although several quantitative tools can assist in this decision-making process, they may not always yield a definitive answer.

Commonly used approaches for selecting the number of components include:

1. **Cumulative variance criterion:**
   Retain enough components to collectively explain at least **80% (or more)** of the total variance in the data.

2. **Kaiser's criterion:**
   Keep components with **eigenvalues greater than 1**, as each such component explains more variance than a single standardized variable.

3. **Scree plot examination:**
   Identify the **point of inflection** (the "elbow") in the scree plot, where additional components contribute only marginally to the total variance explained.

The **eigenvalues** corresponding to each principal component provide direct insight into how much variance that component explains within the dataset.
For example, suppose the total variance equals 5.000. If the first principal component has an eigenvalue of 3.616, it accounts for:

$$\frac{3.616}{5.000} \times 100\% = 72.3\%$$

of the total variance. The next components might explain **10.6%**, **7.7%**, **6.0%**, and **3.3%**, respectively—indicating that the importance of each successive component diminishes rapidly after the first.

Another way to interpret the results is by comparing the **variance of each principal component** to the variance of the original standardized variables (each having a variance of 1.0). In this context, the first principal component exhibits a variance equivalent to **3.616 original variables**, while the second accounts for only **0.532** of a single original variable, with subsequent components contributing even less. This comparison reinforces the **dominant role of the first principal component**, highlighting its ability to capture the majority of the underlying structure in the dataset, while later components primarily represent smaller, residual sources of variation.

In [12]:

```
#PCA is related to the eigen value decompsition
mat= cov(s.dat)
eigen(mat)
```

```
eigen() decomposition
$values
[1] 3.6159783 0.5315041 0.3864245 0.3015655 0.1645275

$vectors
          [,1]        [,2]       [,3]        [,4]       [,5]
[1,] 0.4517989  0.05072137  0.6904702  0.42041399 -0.3739091
[2,] 0.4616809 -0.29956355  0.3405484 -0.54786307  0.5300805
[3,] 0.4505416 -0.32457242 -0.4544927  0.60629605  0.3427923
[4,] 0.4707389 -0.18468403 -0.4109350 -0.38827811 -0.6516665
[5,] 0.3976754  0.87648935 -0.1784558 -0.06887199  0.1924341
```

In [21]:

```
#The proprtion of variance captured by the first PC
(fit.s$sdev[1])^2/sum((fit.s$sdev)^2)
#About 72.3% of variance of all variables can be captured by the first pricipal com


#The proprtion of variance captured by the first three PCs individually
 prop.var3 = (fit.s$sdev[1:3])^2/sum((fit.s$sdev)^2)
 prop.var3
#The total proportion variance captured by the first three PCs
 sum(prop.var3)
#About 90.67% of variance of all variables can be captured by the first three prici
```

```
0.723195668900853
0.906781394041676
```

Scree Plot

A **scree plot** is a visual diagnostic tool used to assess the effectiveness of a **Principal Component Analysis (PCA)** and to help determine how many principal components should be retained.

In a scree plot, the **eigenvalues** (or the proportion of variance explained) of each principal component are plotted against the **component number**. The components are ordered by the amount of variance they explain:

- **PC1** captures the **largest amount of variance** in the dataset.
- **PC2** captures the **second largest**, and so on.

Each successive component explains a smaller portion of the total variability, and together they provide a cumulative picture of how variance is distributed across components.

The total number of principal components equals the number of original variables (features) in the dataset. However, keeping all of them defeats the purpose of dimensionality reduction. The goal is to retain only those components that capture most of the information while discarding those that represent noise or minor variation.

A **scree plot** typically displays a curve that starts high and gradually levels off. The point where the curve begins to flatten (known as the **"elbow"**) indicates a natural cutoff for selecting the number of components. Components before the elbow account for the majority of meaningful variation, while those beyond it contribute little additional explanatory power.

Mathematically, if the eigenvalues are denoted as $\lambda_1, \lambda_2, \ldots, \lambda_p$ (with $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$), the proportion of variance explained by the k-th component is given by:

$$\mathrm{PVE}_k = \frac{\lambda_k}{\sum_{i=1}^{p} \lambda_i}$$

Plotting PVE_k or λ_k against k produces the scree plot.

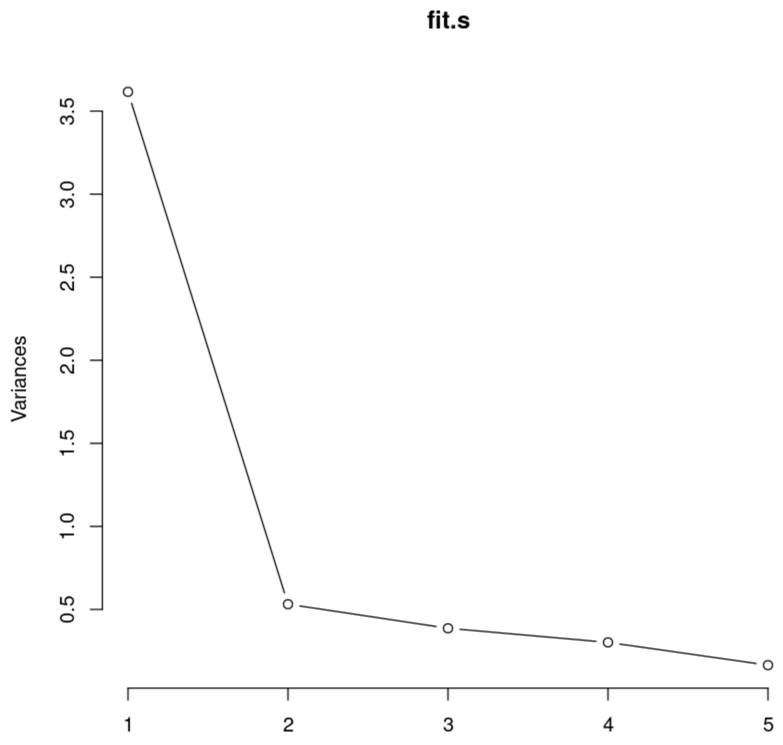
By visually analyzing this plot, one can strike a balance between **data representation** and **dimensional simplicity**, ensuring that most of the original variability is captured with a minimal number of components.

In [16]:

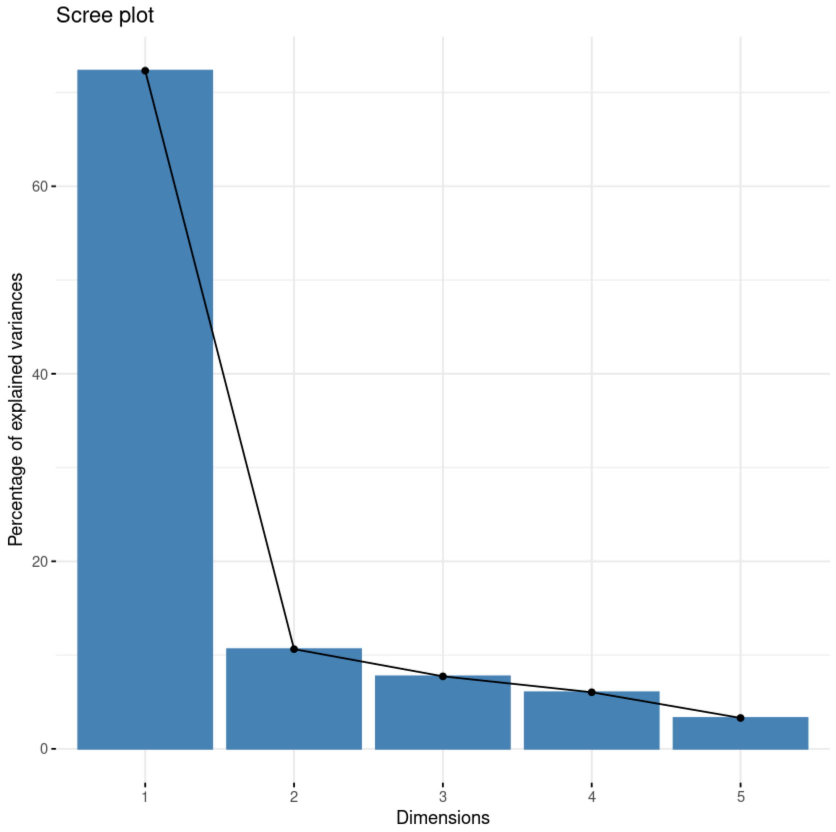
```
#Spree plot
## A scree plot serves as a diagnostic tool to assess the effectiveness of PCA on y
# Principal components are generated in order of the amount of variation:
#PC1 captures the highest variance, PC2 follows, and so forth.
#Each component contributes valuable information about the data, and in a PCA,
#the number of principal components matches the number of characteristics.
#Omitting any principal components results in loss of information.

plot.pca = plot(fit.s, type="l")

#The varaiance captured by PC2, PC3, are much smaller than that of PC1
```



In [23]:
```
#Additional spree plot to visulize the results of PCA
library(factoextra)
fviz_eig(fit.s)
```

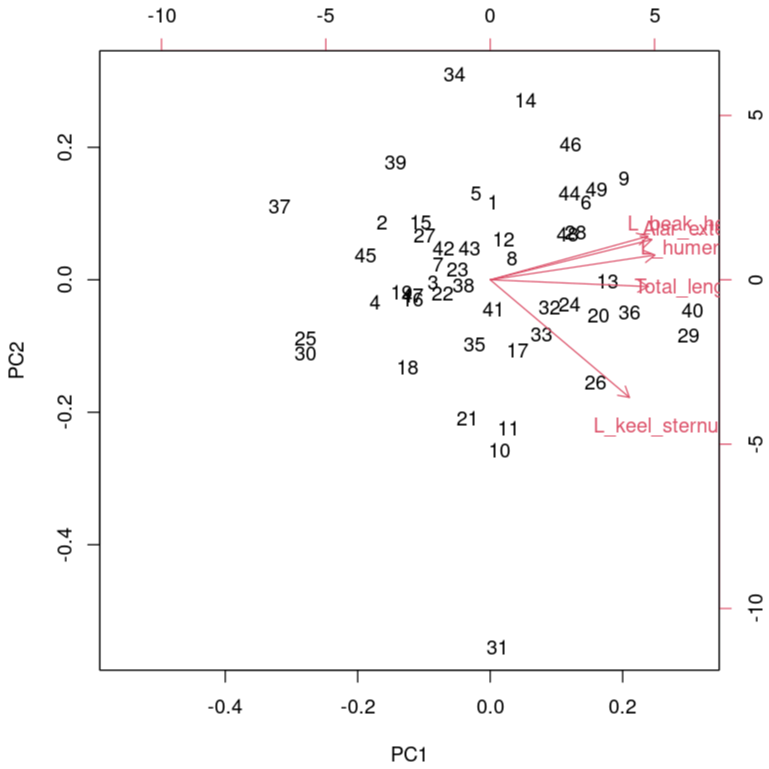


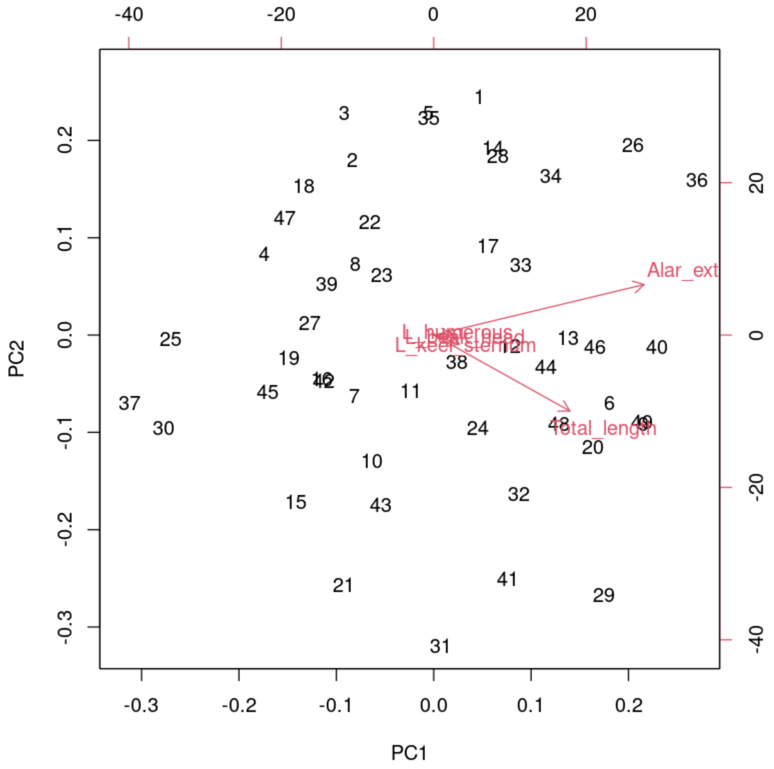
In [29]:

```
#biplot()

biplot(fit.s)

biplot(fit)
#Show the relationship between the variables in terms of PC1 and PC2.
```





A biplot is a graphical representation that simultaneously displays both objects (observations) and variables in the space defined by the principal components. Key aspects of interpreting a biplot include the angles between vectors, the lengths of vectors, and the relative positions of objects and variables.

1. Angles Between Variable Vectors

The angle between two variable vectors reflects the correlation between the corresponding variables:

90°: Indicates no correlation between the variables.

Less than 90°: Suggests a positive correlation. The smaller the angle, the stronger the positive correlation.

Greater than 90°: Indicates a negative correlation. The closer the angle is to 180°, the stronger the negative correlation.

2. Length of Variable Vectors

The length of a variable vector represents the variance of that variable:

Longer vectors: Indicate higher variance and stronger contribution to the principal components.

Shorter vectors: Represent lower variance and a smaller influence on the PCA space.

3. Position of Objects Relative to Variables

The position of an object (observation) relative to a variable vector provides information about its value:

Closer projection onto a variable vector: Reflects a higher value of that variable for the object.

Farther projection from a variable vector: Indicates a lower value of the variable for that object.

4. Distance Between Objects

Closer distance between two objects: Suggests that the objects have similar values across the variables represented in the plot.

Greater distance between objects: Indicates more dissimilarity between their variable profiles.

In [ ]:

In [ ]:

Source: https://cran.r-project.org/web/packages/Sleuth2/refman/Sl

Description Hermon Bumpus analysed various characteristics of some house sparrows that were found on the ground after a severe winter storm in 1898. Some of the sparrows survived and some perished. This data set contains the survival status, age, the length from tip of beak to tip of tail (in mm), the alar extent (length from tip to tip of the extended wings, in mm), the weight in grams, the length of the head in mm, the length of the humerus (arm bone, in inches), the length of the femur (thigh bones, in inches), the length of the tibio–tarsus (leg bone, in inches), the breadth of the skull in inches and the length of the sternum in inches.

Excerise: Run a principle component analysis.

In [3]:

```
#https://cran.r-project.org/web/packages/Sleuth2/refman/Sleuth2.html#ex2016
#library(Sleuth2)
#data(ex2016, package = "Sleuth2")
#dat = ex2016

url <- "https://cran.r-project.org/src/contrib/Sleuth2_2.0-7.tar.gz"
download.file(url, destfile="Sleuth2_2.0-7.tar.gz")
untar("Sleuth2_2.0-7.tar.gz", files = "Sleuth2/data/ex2016.rda", exdir="Sleuth2_unp
```

```r
load("Sleuth2_unpacked/Sleuth2/data/ex2016.rda")
head(ex2016)

dat = ex2016
```

A data.frame: 6 × 11

	Status	AG	TL	AE	WT	BH	HL	FL	TT	SK	KL
	<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	Survived	adult	154	241	24.5	31.2	0.687	0.668	1.022	0.587	0.830
2	Survived	adult	160	252	26.9	30.8	0.736	0.709	1.180	0.602	0.841
3	Survived	adult	155	243	26.9	30.6	0.733	0.704	1.151	0.602	0.846
4	Survived	adult	154	245	24.3	31.7	0.741	0.688	1.146	0.584	0.839
5	Survived	adult	156	247	24.1	31.5	0.715	0.706	1.129	0.575	0.821
6	Survived	adult	161	253	26.5	31.8	0.780	0.743	1.144	0.607	0.893