# Statistical Machine Learning
# Hilary Term 2018

**Pier Francesco Palamara**
Department of Statistics
University of Oxford

Slide credits and other course material can be found at:
http://www.stats.ox.ac.uk/~palamara/SML18.html

January 19, 2018

# Unsupervised Learning:
# Visualisation and Dimensionality Reduction

# Unsupervised Learning

Goals:

- Find the variables that summarise the data / capture relevant information.
- Discover informative ways to visualise the data.
- Discover the subgroups among the observations.

It is often much easier to obtain unlabeled data than labeled data!

# Exploratory Data Analysis

## Notation

- Data consists of $p$ variables (features/attributes/dimensions) on $n$ examples (items/observations).
- $\mathbf{X} = (x_{ij})$ is a $n \times p$-matrix with $x_{ij} :=$ the $j$-th variable for the $i$-th example

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{ip} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nj} & \dots & x_{np} \end{bmatrix}.$$

- Denote the $i$-th data item by $x_i \in \mathbb{R}^p$ (we will treat it as a column vector: it is the transpose of the $i$-th row of $\mathbf{X}$).
- Assume $x_1, \dots, x_n$ are **independently and identically distributed** samples of a **random vector** $X$ over $\mathbb{R}^p$. The $j$-th dimension of $X$ will be denoted $X^{(j)}$.

# Crabs Data ($n = 200$, $p = 5$)

Campbell (1974) studied rock crabs of the genus **leptograpsus**. One species, **L. variegatus**, had been split into two new species according to their colour: orange and blue. Preserved specimens lose their colour, so it was hoped that morphological differences would enable museum material to be classified.

Data are available on 50 specimens of each sex of each species. Each specimen has measurements on:

- the width of the frontal lobe `FL`,
- the rear width `RW`,
- the length along the carapace midline `CL`,
- the maximum width `CW` of the carapace, and
- the body depth `BD` in mm.

in addition to colour/species and sex (we will later view these as labels, but will ignore for now).

photo from: inaturalist.org

# Crabs Data

```
## load package MASS containing the data
library(MASS)

## extract variables we will look at
varnames<-c("FL","RW","CL","CW","BD")
Crabs <- crabs[,varnames]

## look at raw data
Crabs
```

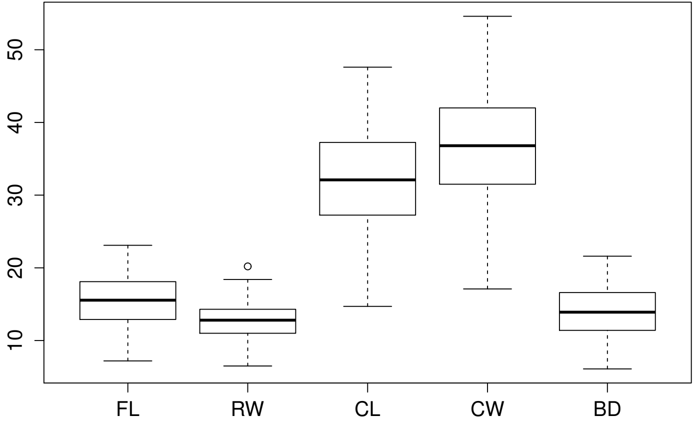
# Crabs Data

```
## look at raw data
Crabs
```

```
        FL   RW   CL   CW   BD
   1     8.1  6.7 16.1 19.0  7.0
   2     8.8  7.7 18.1 20.8  7.4
   3     9.2  7.8 19.0 22.4  7.7
   4     9.6  7.9 20.1 23.1  8.2
   5     9.8  8.0 20.3 23.0  8.2
   6    10.8  9.0 23.0 26.5  9.8
   7    11.1  9.9 23.8 27.1  9.8
   8    11.6  9.1 24.5 28.4 10.4
   9    11.8  9.6 24.2 27.8  9.7
   10   11.8 10.5 25.2 29.3 10.3
   11   12.2 10.8 27.3 31.6 10.9
   12   12.3 11.0 26.8 31.5 11.4
   13   12.6 10.0 27.7 31.7 11.4
   14   12.8 10.2 27.2 31.8 10.9
   15   12.8 10.9 27.4 31.5 11.0
   16   12.9 11.0 26.8 30.9 11.4
   17   13.1 10.6 28.2 32.3 11.0
   18   13.1 10.9 28.3 32.4 11.2
   19   13.3 11.1 27.8 32.3 11.3
   20   13.9 11.1 29.2 33.3 12.1
```

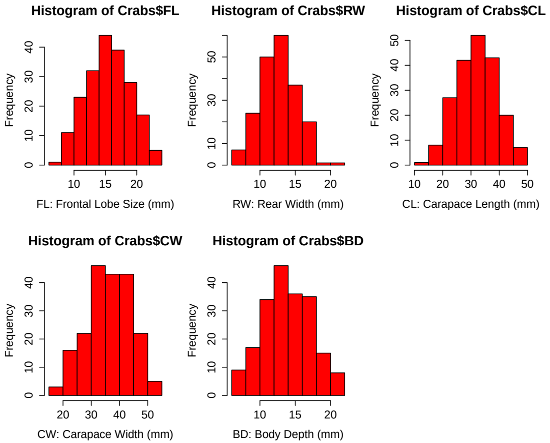
# Univariate Boxplots

```
boxplot(Crabs)
```



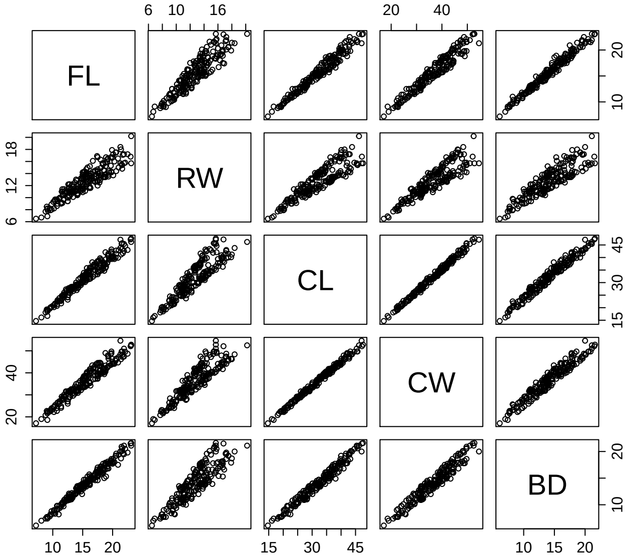
# Univariate Histograms

```
par(mfrow=c(2,3))
hist(Crabs$FL,col="red",xlab="FL: Frontal Lobe Size (mm)")
hist(Crabs$RW,col="red",xlab="RW: Rear Width (mm)")
hist(Crabs$CL,col="red",xlab="CL: Carapace Length (mm)")
hist(Crabs$CW,col="red",xlab="CW: Carapace Width (mm)")
hist(Crabs$BD,col="red",xlab="BD: Body Depth (mm)")
```



# Simple Pairwise Scatterplots

```
pairs(Crabs)
```



# Visualisation and Dimensionality Reduction

The summary plots are useful, but limited use if the dimensionality $p$ is high (a few dozens or even thousands).

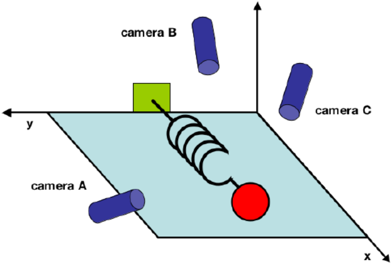
- Constrained to view data in 2 or 3 dimensions
- Approach: look for ‘interesting’ projections of $\mathbf{X}$ into lower dimensions
- Hope that even though $p$ is large, considering only carefully selected $k \ll p$ dimensions is just as informative.

## Dimensionality reduction

- For each data item $x_i \in \mathbb{R}^p$, find its lower dimensional representation $z_i \in \mathbb{R}^k$ with $k \ll p$.
- Map $x \mapsto z$ should preserve the **interesting statistical properties** in data.

# Dimensionality reduction

- deceptively many variables to measure, many of them redundant / correlated to each other (large $p$)
- often, there is a simple but unknown underlying relationship hiding
- example: ball on a frictionless spring recorded by three different cameras
  - our imperfect measurements obfuscate the true underlying dynamics
  - are our coordinates meaningful or do they simply reflect the method of data gathering?



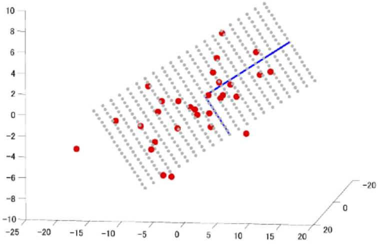
J. Shlens, A Tutorial on Principal Component Analysis, 2005

# Principal Components Analysis (PCA)

- PCA considers interesting directions to be those with greatest **variance**.
- A **linear** dimensionality reduction technique: looks for a **new basis** to represent a noisy dataset.
- Workhorse for many different types of data analysis (often used for data preprocessing before supervised techniques are applied).
- Often the first thing to run on high-dimensional data.

# Principal Components Analysis (PCA)

- For simplicity, we will assume from now on that our dataset is centred, i.e., we subtract the average $\bar{x}$ from each $x_i$.



## PCA

Find an orthogonal basis $v_1, v_2, \dots, v_p$ for the data space such that:

- The first principal component (PC) $v_1$ is the **direction of greatest variance** of data.
- The $j$-th PC $v_j$ (also called **loading vector**) is the **direction orthogonal to** $v_1, v_2, \dots, v_{j-1}$ **of greatest variance**, for $j = 2, \dots, p$.

# Principal Components Analysis (PCA)

- The $k$-dimensional representation of data item $x_i$ is the vector of projections of $x_i$ onto first $k$ PCs:

$$z_i = V_{1:k}^\top x_i = \left[v_1^\top x_i, \ldots, v_k^\top x_i\right]^\top \in \mathbb{R}^k,$$

  where $V_{1:k} = [v_1, \ldots, v_k]$
- Reconstruction of $x_i$:

$$\hat{x}_i = V_{1:k} V_{1:k}^\top x_i.$$

- PCA gives the **optimal linear reconstruction** of the original data based on a $k$-dimensional compression (problem sheets).

# Principal Components Analysis (PCA)

- Our data set is an i.i.d. sample $\{x_i\}_{i=1}^n$ of a random vector $X = \left[X^{(1)} \dots X^{(p)}\right]^\top$.
- For the $1^{st}$ PC, we seek a derived scalar variable of the form

$$Z^{(1)} = v_1^\top X = v_{11}X^{(1)} + v_{12}X^{(2)} + \cdots + v_{1p}X^{(p)}$$

  where $v_1 = [v_{11}, \dots, v_{1p}]^\top \in \mathbb{R}^p$ are chosen to maximise

$$\mathrm{Var}(Z^{(1)}).$$

- The $2^{nd}$ PC is chosen to be orthogonal with the $1^{st}$ and is computed in a similar way. It will have the largest variance in the remaining $p-1$ dimensions, etc.

# Deriving the First Principal Component

- for any fixed $v_1$,

$$\mathrm{Var}(Z^{(1)}) = \mathrm{Var}(v_1^\top X) = v_1^\top \operatorname{Cov}(X) v_1.$$

- we do not know the **true** covariance matrix $\operatorname{Cov}(X)$, so need to replace with the sample covariance matrix, i.e.

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^\top = \frac{1}{n-1}\sum_{i=1}^{n} x_i x_i^\top = \frac{1}{n-1}\mathbf{X}^\top\mathbf{X}.$$

- with no restriction on the norm of $v_1$, $\mathrm{Var}(Z^{(1)})$ grows without a bound: need constraint $v_1^\top v_1 = 1$, giving

$$\max_{v_1} \; v_1^\top S v_1$$

subject to: $v_1^\top v_1 = 1.$

# Deriving the First Principal Component

- Lagrangian of the problem is given by:

$$\mathcal{L}\left(v_1, \lambda_1\right) = v_1^\top S v_1 - \lambda_1 \left(v_1^\top v_1 - 1\right).$$

- The corresponding vector of partial derivatives is

$$\frac{\partial \mathcal{L}(v_1, \lambda_1)}{\partial v_1} = 2Sv_1 - 2\lambda_1 v_1.$$

- Setting this to zero gives $Sv_1 = \lambda_1 v_1$. Recognize the eigenvector equation, i.e. $v_1$ must be an eigenvector of $S$ and the dual variable $\lambda_1$ is the corresponding eigenvalue.
- Since $v_1^\top S v_1 = \lambda_1 v_1^\top v_1 = \lambda_1$, the first PC must be the eigenvector associated with the largest eigenvalue of $S$.

# PCA as eigendecomposition of the covariance matrix

- The eigenvalue decomposition of $S$ is given by

$$S = V \Lambda V^\top$$

where $\Lambda$ is a diagonal matrix with eigenvalues

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$$

and $V$ is a $p \times p$ orthogonal matrix whose columns are the $p$ eigenvectors of $S$, i.e. the principal components $v_1, \ldots, v_p$.

# Properties of the Principal Components

- Derived scalar variable (projection to the $j$-th principal component) $Z^{(j)} = v_j^\top X$ has sample variance $\lambda_j$, for $j = 1, \ldots, p$
- $S$ is a real symmetric matrix, so eigenvectors (principal components) are orthogonal.
- Projections to principal components are **uncorrelated**: $\mathrm{Cov}(Z^{(i)}, Z^{(j)}) \approx v_i^\top S v_j = \lambda_j v_i^\top v_j = 0$, for $i \neq j$.
- The **total sample variance** is given by $\sum_{i=1}^{p} S_{ii} = \lambda_1 + \ldots + \lambda_p$, so the **proportion of total variance explained** by the $j^{th}$ PC is $\frac{\lambda_j}{\lambda_1 + \lambda_2 + \ldots + \lambda_p}$

# R code

This is what we have had before:

```
> library(MASS)
> varnames<-c('FL','RW','CL','CW','BD')
> Crabs <- crabs[,varnames]
```

Now perform PCA with function `princomp`.
(Alternatively, solve for the PCs yourself using `eigen` or `svd`)

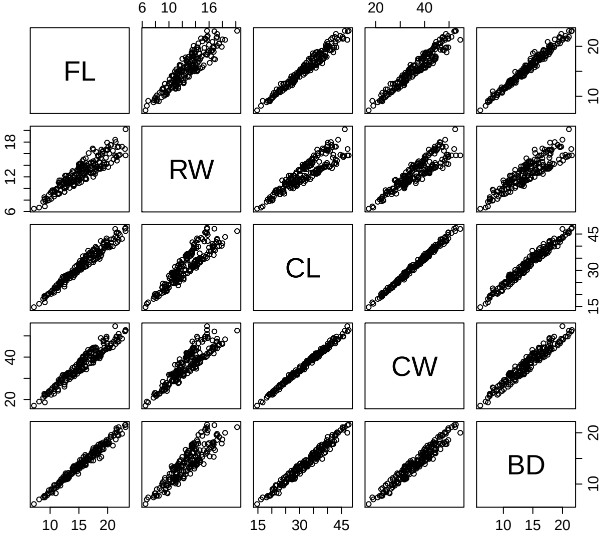
```
> Crabs.pca <- princomp(Crabs)
```

# Exploring PCA output

```
> Crabs.pca <- princomp(Crabs)
> summary(Crabs.pca)

Importance of components:
                          Comp.1      Comp.2      Comp.3       Comp.4       Comp.5
Standard deviation     11.8322521 1.135936870 0.997631086 0.3669098284 0.2784325016
Proportion of Variance  0.9824718 0.009055108 0.006984337 0.0009447218 0.0005440328
Cumulative Proportion   0.9824718 0.991526908 0.998511245 0.9994559672 1.0000000000

> loadings(Crabs.pca)

Loadings:
   Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
FL -0.289 -0.323  0.507  0.734  0.125
RW -0.197 -0.865 -0.414 -0.148 -0.141
CL -0.599  0.198  0.175 -0.144 -0.742
CW -0.662  0.288 -0.491  0.126  0.471
BD -0.284 -0.160  0.547 -0.634  0.439
```

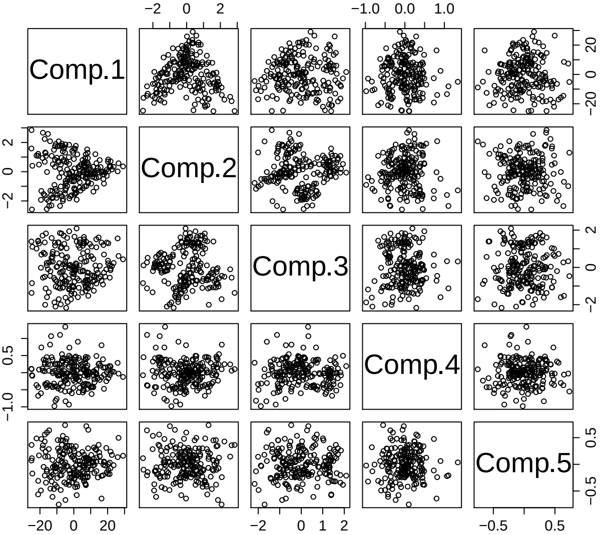
# Raw Crabs Data

```
> pairs(Crabs)
```



# PCA of Crabs Data

```
> Crabs.pca <- princomp(Crabs)
> pairs(predict(Crabs.pca))
```



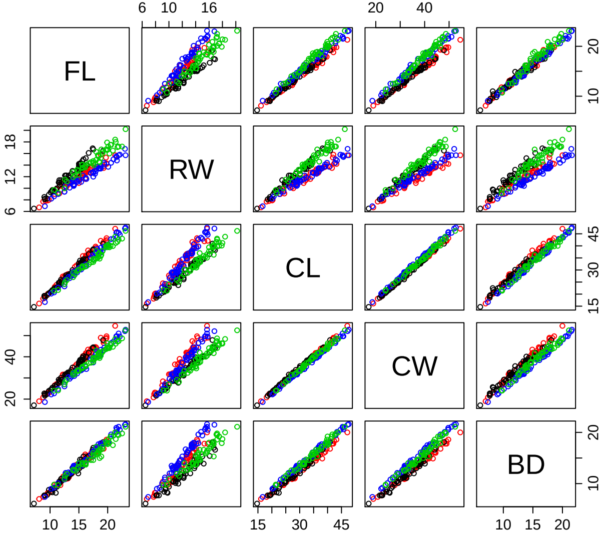
# What did we discover?

Now let us use our label information (species+sex).

```
> Crabs.class <- factor(paste(crabs$sp,crabs$sex,sep=""))
> Crabs.class
  [1] BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM
 [27] BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BM BF BF
 [53] BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF
 [79] BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF BF OM OM OM OM
[105] OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM
[131] OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OM OF OF OF OF OF OF
[157] OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF
[183] OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF OF
Levels: BF BM OF OM
```

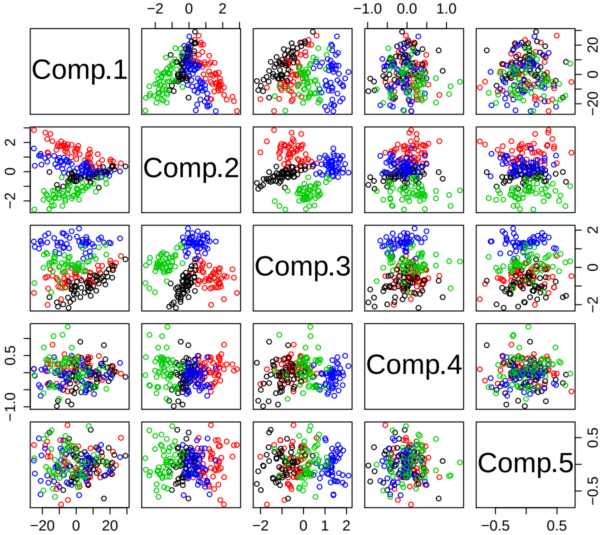
# Raw Crabs Data - with labels

```
> pairs(Crabs,col=unclass(Crabs.class))
```



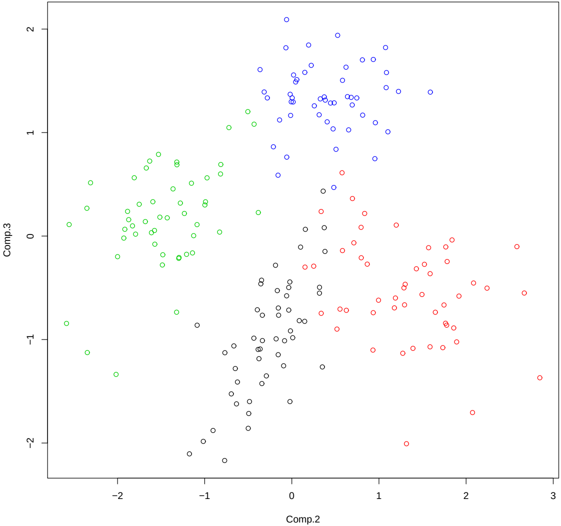
# PCA of Crabs Data - with labels

```
> Crabs.pca <- princomp(Crabs)
> pairs(predict(Crabs.pca),col=unclass(Crabs.class))
```

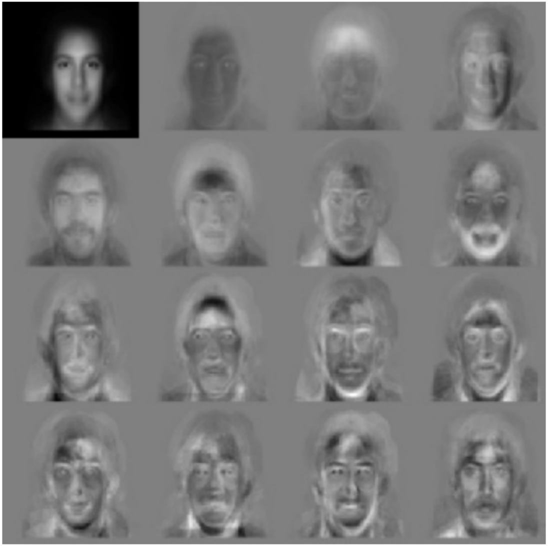


# PC 2 vs PC 3

```
> Z<-predict(Crabs.pca)
> plot(Comp.3~Comp.2,data=Z,col=unclass(Crabs.class))
```

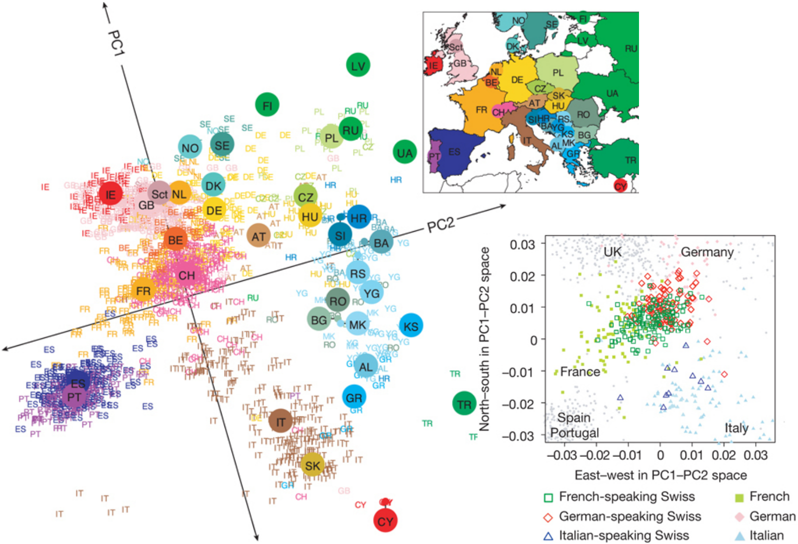


# PCA on Face Images: Eigenfaces



http://vismod.media.mit.edu/vismod/demos/facerec/basic.html

# PCA on European Genetic Variation



Genes mirror geography within Europe, Nature 2008

# Comments on the use of PCA

- PCA commonly used to project data $X$ onto the first $k$ PCs giving the $k$-dimensional view of the data that best preserves **the first two moments**.
- Although PCs are uncorrelated, scatterplots sometimes reveal structures in the data other than linear correlation.
- Emphasis on variance is where the weaknesses of PCA stem from:
  - Assuming large variances are meaningful (high signal-to-noise ratio)
  - The PCs depend heavily on the units measurement. Where the data matrix contains measurements of vastly differing orders of magnitude, the PC will be greatly biased in the direction of larger measurement. In these cases, it is recommended to calculate PCs from $\mathrm{Corr}(X)$ instead of $\mathrm{Cov}(X)$ (`cor=True` in the call of `princomp`).
  - Lack of robustness to outliers: variance is affected by outliers and so are PCs.
  - Sample size (e.g. the number of crabs collected for each sub-species) will have an effect on the PCs.

# Summary: PCA

## PCA

Find an orthogonal basis $\{v_1, v_2, \ldots, v_p\}$ for the data space such that:

- The first principal component (PC) $v_1$ is the **direction of greatest variance** of data.
- The $j$-th PC $v_j$ is the **direction orthogonal to** $v_1, v_2, \ldots, v_{j-1}$ **of greatest variance**, for $j = 2, \ldots, p$.

- Eigendecomposition of the sample covariance matrix $S = \frac{1}{n-1}\sum_{i=1}^{n} x_i x_i^\top$.

$$S = V \Lambda V^\top.$$

  - $\Lambda$ is a diagonal matrix with eigenvalues (variances along each principal component) $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$
  - $V$ is a $p \times p$ orthogonal matrix whose columns are the $p$ eigenvectors of $S$, i.e. the principal components $v_1, \ldots, v_p$
- Dimensionality reduction by projecting $x_i \in \mathbb{R}^p$ onto first $k$ principal components:

$$z_i = \left[v_1^\top x_i, \ldots, v_k^\top x_i\right]^\top \in \mathbb{R}^k.$$

# Summary: PCA

$$S = \frac{1}{n-1}\sum_{i=1}^{n} x_i x_i^\top = \frac{1}{n-1}\mathbf{X}^\top\mathbf{X}.$$

- $S$ is a **real and symmetric** matrix, so there exist $p$ eigenvectors $v_1, \dots, v_p$ that are pairwise orthogonal and $p$ associated eigenvalues $\lambda_1, \dots, \lambda_p$ which satisfy the eigenvalue equation $Sv_i = \lambda_i v_i$. In particular, $V$ is an orthogonal matrix:

$$VV^\top = V^\top V = I_p.$$

- $S$ is a **positive-semidefinite** matrix, so the eigenvalues are non-negative:

$$\lambda_i \geq 0, \ \forall i.$$

Why is $S$ symmetric? Why is $S$ positive-semidefinite?
Reminder: A symmetric $p \times p$ matrix $R$ is said to be positive-semidefinite if

$$\forall a \in \mathbb{R}^p, a^\top Ra \geq 0.$$

# Singular Value Decomposition (SVD)

### SVD

Any real-valued $n \times p$ matrix $\mathbf{X}$ can be written as $X = UDV^\top$ where

- $U$ is an $n \times n$ orthogonal matrix: $UU^\top = U^\top U = I_n$
- $D$ is a $n \times p$ matrix with decreasing **non-negative** elements on the diagonal (the singular values) and zero off-diagonal elements.
- $V$ is a $p \times p$ orthogonal matrix: $VV^\top = V^\top V = I_p$

- SVD **always** exists, even for non-square matrices.
- Fast and numerically stable algorithms for SVD are available in most packages.The relevant R command is `svd`.

# SVD and PCA

- Let $\mathbf{X} = UDV^\top$ be the SVD of the $n \times p$ data matrix $\mathbf{X}$.
- Note that

  $$(n-1)S = \mathbf{X}^\top\mathbf{X} = (UDV^\top)^\top(UDV^\top) = VD^\top U^\top UDV^\top = VD^\top DV^\top,$$

  using orthogonality ($U^\top U = I_n$) of $U$.
- The eigenvalues of $S$ are thus the diagonal entries of $\Lambda = \frac{1}{n-1}D^\top D$.
- We also have (using orthogonality $V^\top V = I_p$)

  $$\mathbf{X}\mathbf{X}^\top = (UDV^\top)(UDV^\top)^\top = UDV^\top VD^\top U^\top = UDD^\top U^\top,$$

### Gram matrix

$\mathbf{B} = \mathbf{X}\mathbf{X}^\top$, $\mathbf{B}_{ij} = x_i^\top x_j$ is called the Gram matrix of dataset $\mathbf{X}$.
$\mathbf{B}$ and $(n-1)S = \mathbf{X}^\top\mathbf{X}$ have the same nonzero eigenvalues, equal to the non-zero squared singular values of $\mathbf{X}$.

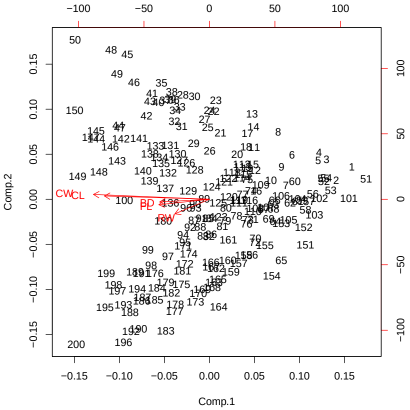
Projection:

$$Z = \mathbf{X}V = UDV^\top V = UD.$$

Can be obtain by eigendecomposition of $\mathbf{B}$, less computation if $p > n$.

# Biplots

```
> biplot(Crabs.pca,scale=1)
```



- PCA plots show the data items (rows of $\mathbf{X}$) in the space spanned by PCs.
- **Biplots** allow us to visualize the **original variables** $X^{(1)}, \ldots, X^{(p)}$ (corresponding to columns of $\mathbf{X}$) in the same plot.