Administration

Thank you for survey responses. Will email when digested.

Assignment 2 is out.

Remember tutorial sheets

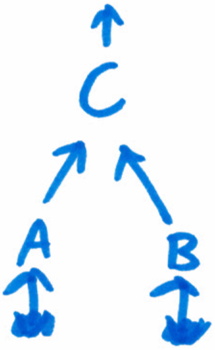
Ask — on the Hypothesis forum† — and you will receive*

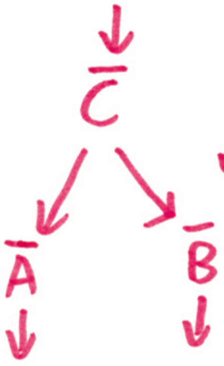
† Instructions on website

* help, and answers if you have put effort in.

Reverse mode differentiation

Piece of computation:





Backpropagate derivatives wrt final scalar output

For any Z,

$$\bar{Z}_{ij} = \frac{\partial \text{output}}{\partial Z_{ij}}$$

Use standard rules:

$$C = \cos A \Rightarrow \bar{A} = \bar{C} \odot \sin A$$

$$C = AB \Rightarrow \bar{A} = \bar{C}B^\top, \quad \bar{B} = A^\top \bar{C}$$

$$C = A + B \Rightarrow \bar{A} = \bar{C}, \quad \bar{B} = \bar{C}$$

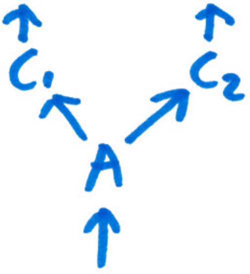
$$C = A^\top \Rightarrow \bar{A} = \bar{C}^\top$$

...

Stored in forward pass

Passed in by backprop

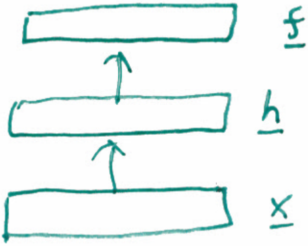
Multiple children



$$\begin{tikzcd}[column sep=small] & \bar{C}_1 \arrow[d] & & \bar{C}_2 \arrow[d] \\ \bar{A} = & \bar{A}_1 & + & \bar{A}_2 \end{tikzcd}$$

Apply rules separately for children and add

Autoencoder



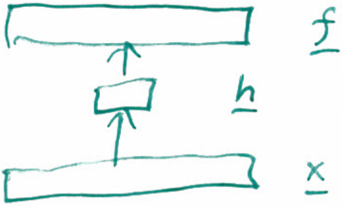
Learning task

$$\underline{f}(\underline{x}) \approx \underline{x}$$

not useful

```
def autoencode(x):
    ~~return x~~
    h = np.dot(I, x)
    return h
```

Dimensionality Reduction



K units $\quad K \ll D$

D inputs

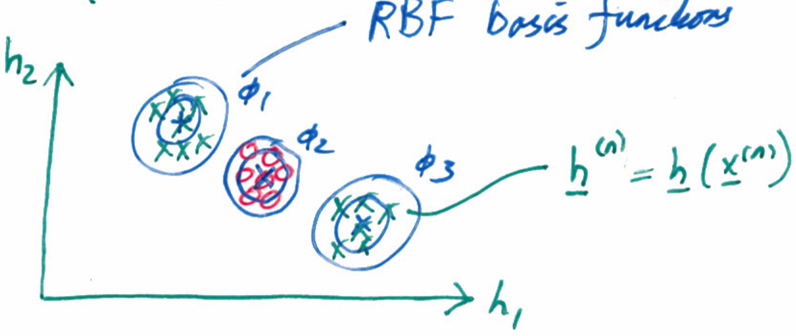
$$\underline{h} = g^{(1)}(W^{(1)}\underline{x} + \underline{b}^{(1)})$$

$$\underline{f} = g^{(2)}(W^{(2)}\underline{h} + \underline{b}^{(2)})$$

$\Rightarrow$ Use $\underline{h}$ as inputs to other ML method.

Visualization

Set $K = 2$

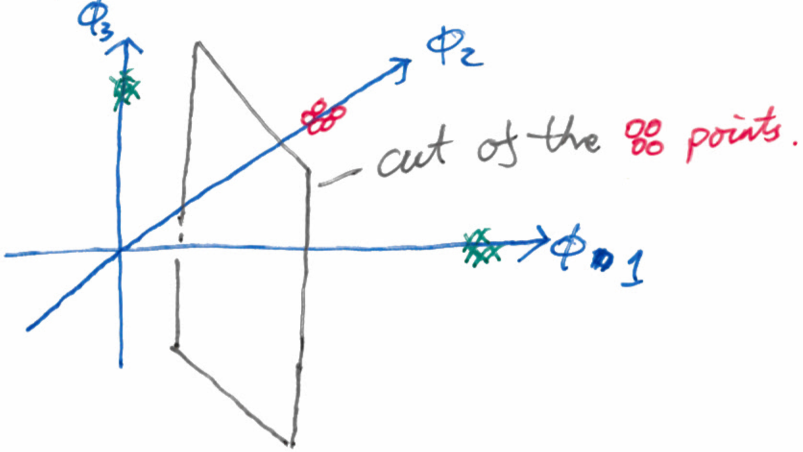


Use some labels to see if visualization makes sense..

○○ class 0

×× class 1

Or do classification in this space after learning mapping on unlabeled data.



Sparse Autoencoders

Force most elements of $\underline{h}$ to be zero.

Denoising Autoencoders

While training we mask out some of the inputs — set some x_d to zero.

$\underline{m}$ mask vector, of random 0's & 1's

Cost on one example $\quad \| \underline{f}(\underline{x}^{(n)} \odot \underline{m}) - \underline{x}^{(n)} \|^2$

Cost function $\displaystyle \sum_{\underline{m}} p(\underline{m}) \sum_{n=1}^{N} \| f(\underline{x}^{(n)} \odot \underline{m}) - \underline{x}^{(n)} \|^2$

Monte Carlo

$$\approx \| f(\underline{x}^{(n)} \odot \underline{m}) - \underline{x}^{(n)} \|^2$$

For random n and $\underline{m} \sim p(\underline{m})$

Principal Components Analysis (PCA)

It fits a linear autoencoder:

$$g^{(1)}(\underline{z}) = \underline{z}$$
$$g^{(2)}(\underline{z}) = \underline{z}$$

PCA advantages:

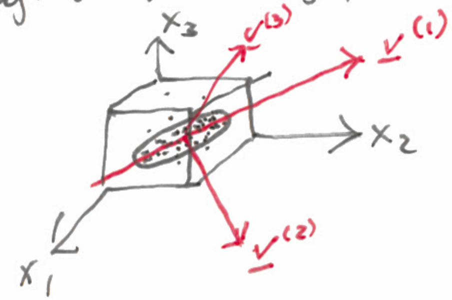
- Fit with linear algebra operations
- Same answer every time
- The solutions for different K are nested

  $h_1(\underline{x})$ is the same no matter K to be.

  $h_2(\underline{x})$ " " " for all $K \geqslant 2$

PCA

High-dim. ball of points — Here $D = 3$



$\underline{v}^{(k)}$ k^{th} eigenvector of $\mathrm{cov}[X]$

$$\underset{D \times K}{V} = \begin{bmatrix} | & | & & | \\ \underline{v}^{(1)} & \underline{v}^{(2)} & \dots & \underline{v}^{(K)} \\ | & | & & | \end{bmatrix}$$

Pick $K=2$ here

Reduce dim vector:

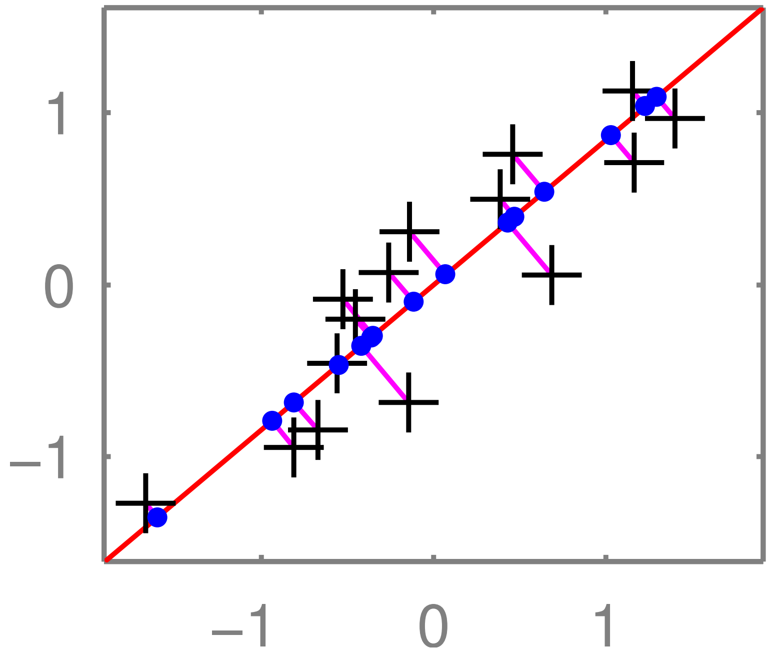
$$\underset{D \times 1}{\underline{x}} \rightarrow \underset{K \times 1}{V^T(\underline{x} - \underline{\mu})}$$

Reconstruct into $\overset{D}{3}$ dimensions:

$$\underset{D \times 1}{\hat{\underline{x}}} = \underset{D \times K}{V} \underbrace{V^T(\underline{x}}_{K \times 1} - \underline{\mu}) + \underline{\mu}$$

Should have mentioned data centering. Next time!

PCA: Principal Component Analysis



`K` = `1`

+ = `X`

• = `Xproj`

— = `V(:,1)`

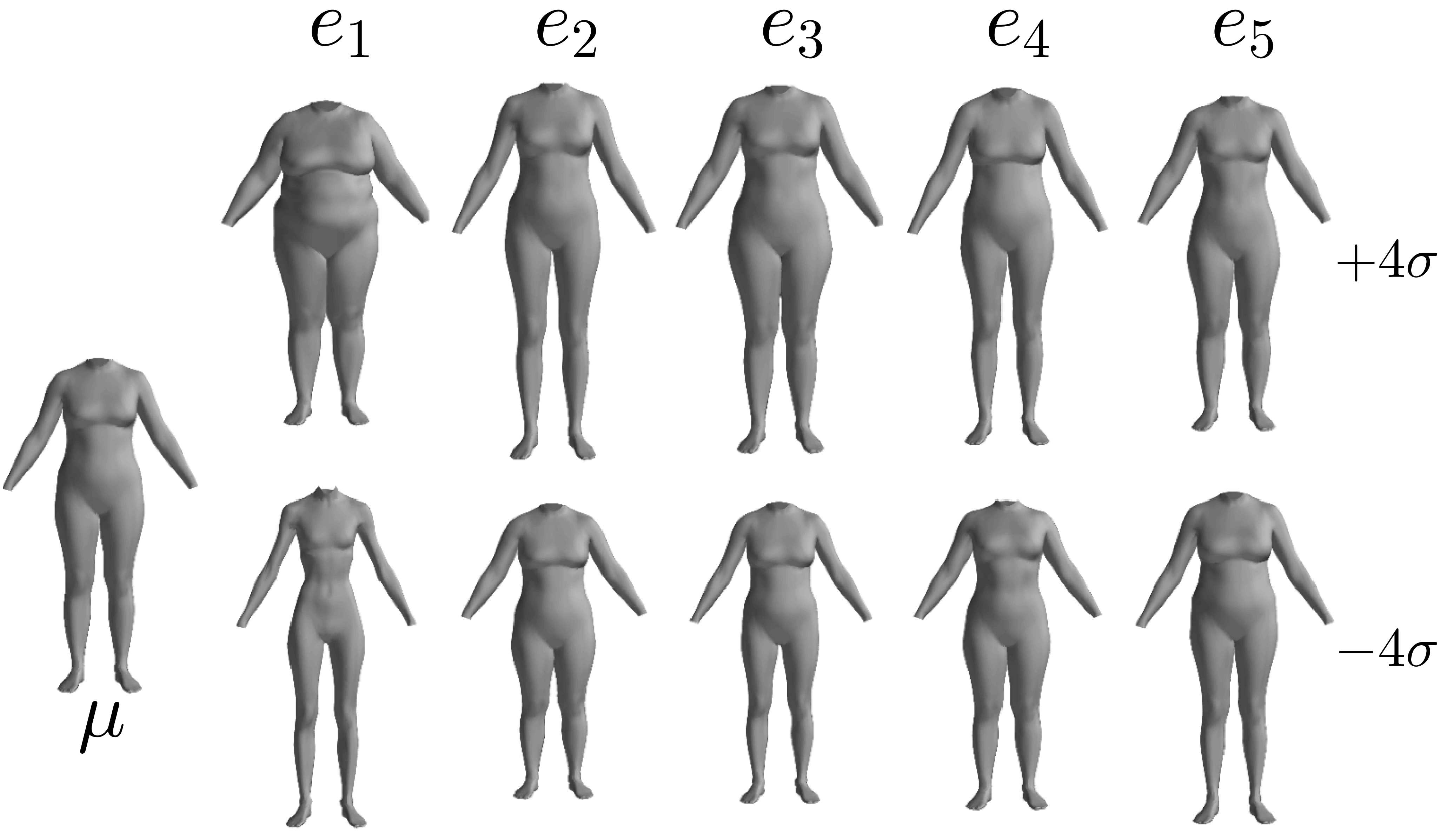
Code assuming `X` is zero-mean

```
% Find top K principal directions:
[V, E] = eig(X'*X);
[E,id] = sort(diag(E),1,'descend');
V = V(:, id(1:K));   % DxK

% Project to K-dims:
X_kdim = X*V;   % NxK

% Project back:
X_proj = X_kdim * V';   % NxD
```

PCA applied to bodies



Freifeld and Black, ECCV 2012

PCA applied to DNA

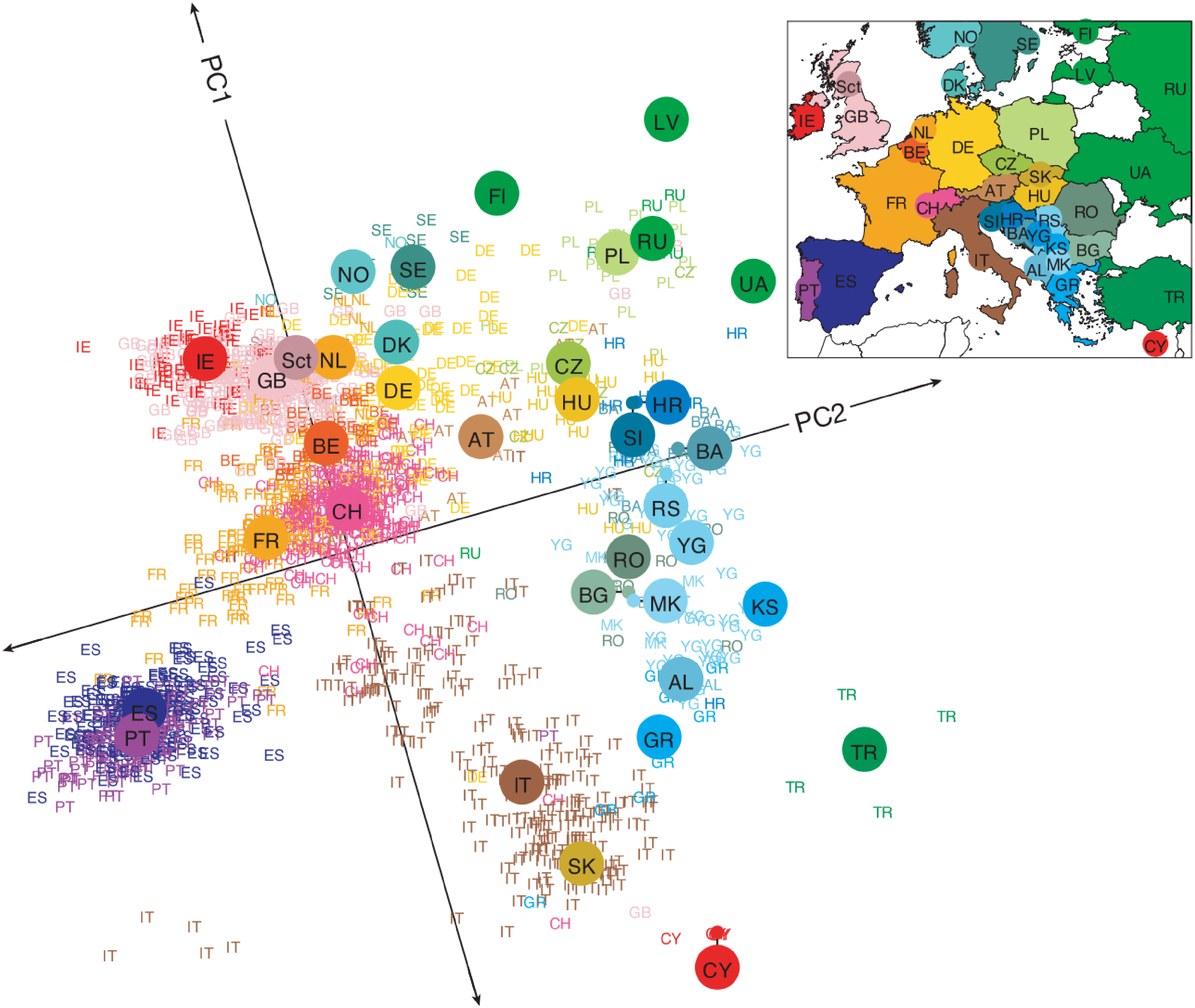
Novembre et al. (2008) — doi:10.1038/nature07331

Carefully selected both individuals and features

1,387 individuals

197,146 single nucleotide polymorphisms (SNPs)

Each person reduced to two(!) numbers with PCA



PC1
PC2
LV
FI
RU
PL
UA
NO
SE
DK
IE
Sct
GB
NL
DE
CZ
HU
HR
SI
BA
AT
BE
CH
FR
RS
YG
RO
BG
MK
KS
AL
GR
TR
ES
PT
IT
SK
CY

MSc course enrollment data

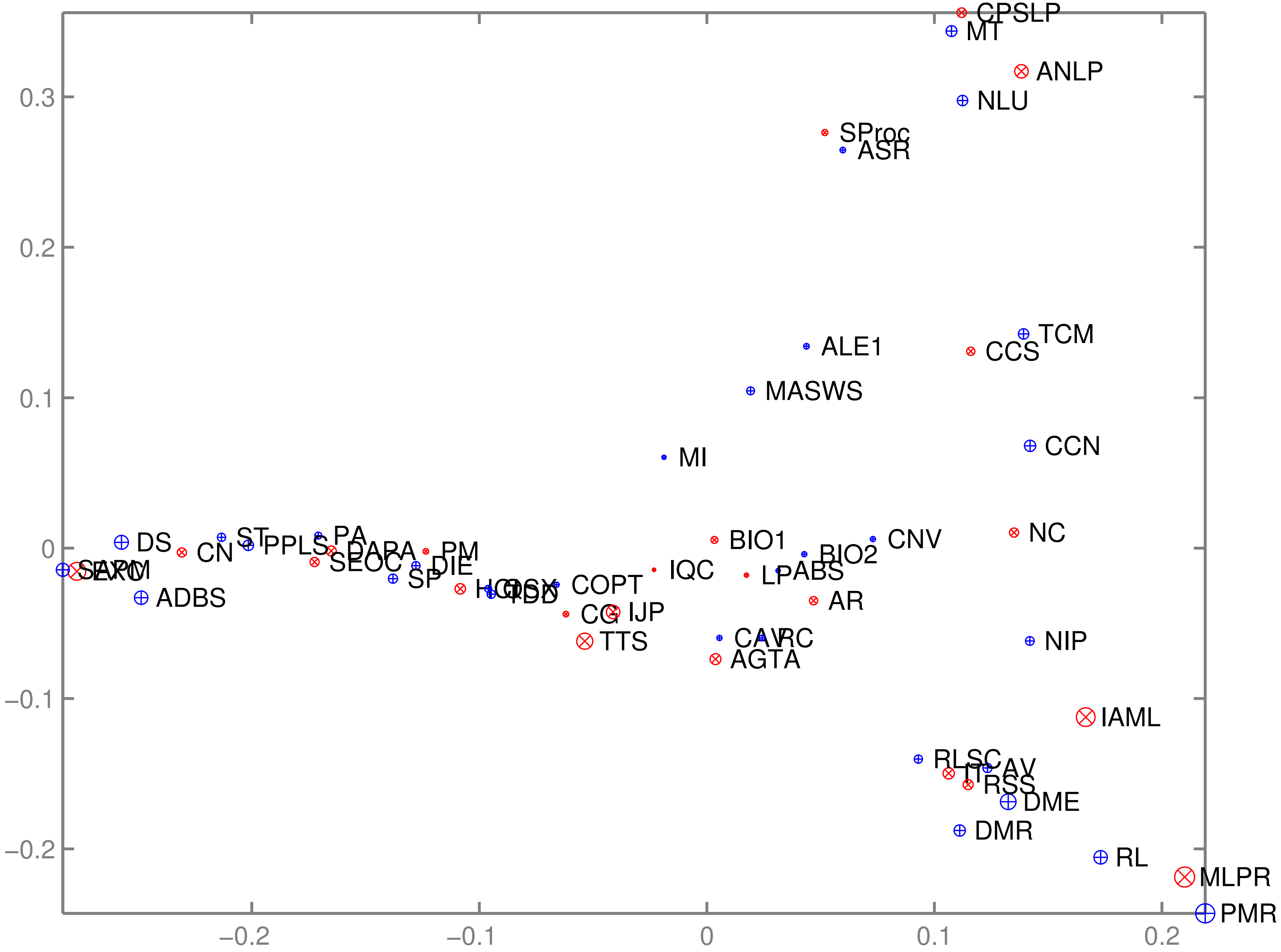
Binary $S \times C$ matrix M

$M_{sc} = 1,$ if student s taking course c

Each course is a length S vector

. . . OR each student is a length C vector

PCA applied to MSc courses



PCA applied to MSc students