CPSC 340: Machine Learning and Data Mining

More PCA

Fall 2016

Admin

- A2/Midterm:
  - Grades/solutions posted.
  - Midterms can be viewed during office hours.
- Assignment 4:
  - Due Monday.
- Extra office hours:
  - Thursdays from 4:30-5:30 in ICICS X836.

1. Decision trees
2. Naïve Bayes classification
3. Ordinary least squares regression
4. Logistic regression
5. Support vector machines
6. Ensemble methods
7. Clustering algorithms
8. Principal component analysis
9. Singular value decomposition
10. Independent component analysis

http://www.kdnuggets.com/2016/08/10-algorithms-machine-learning-engineers.html

Last Week: Principal Component Analysis (PCA)

- PCA is a linear model for unsupervised learning.
- Represents features as linear combination of latent factors:

$$x_{ij} = w_j^T z_i$$

$\hookrightarrow$ column 'j'

$$x_i = W^T z_i = z_{i1} \underbrace{w_1}_{\text{row 1}} + z_{i2} \underbrace{w_2}_{\text{row 2}} + \cdots + z_{ik} \underbrace{w_k}_{\text{row } k}$$

the means/factors

  - But we're learning the latent factors 'W' and latent features z_i.

- Can also be viewed as an approximate matrix factorization:

$$X \approx ZW$$

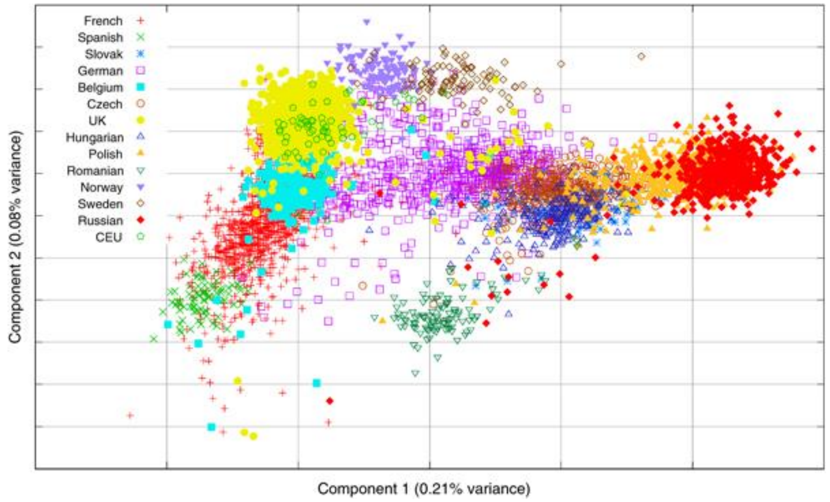
$$\underbrace{\begin{bmatrix} & & \\ & & \end{bmatrix}}_{n \times d} \approx \underbrace{\begin{bmatrix} \\ \\ \end{bmatrix}}_{n \times k} \underbrace{\begin{bmatrix} & k \times d & \end{bmatrix}}_{}$$

Last Week: Principal Component Analysis (PCA)

- PCA is a linear model for unsupervised learning.
- Represents features as linear combination of latent factors:

$$x_{ij} = w_j^T z_i \qquad\qquad x_i = W^T z_i$$

- Uses: dimensionality reduction, visualization, factor discovery.



Trait	Description
Openness	Being curious, original, intellectual, creative, and open to new ideas.
Conscientiousness	Being organized, systematic, punctual, achievement-oriented, and dependable.
Extraversion	Being outgoing, talkative, sociable, and enjoying social situations.
Agreeableness	Being affable, tolerant, sensitive, trusting, kind, and warm.
Neuroticism	Being anxious, irritable, temperamental, and moody.

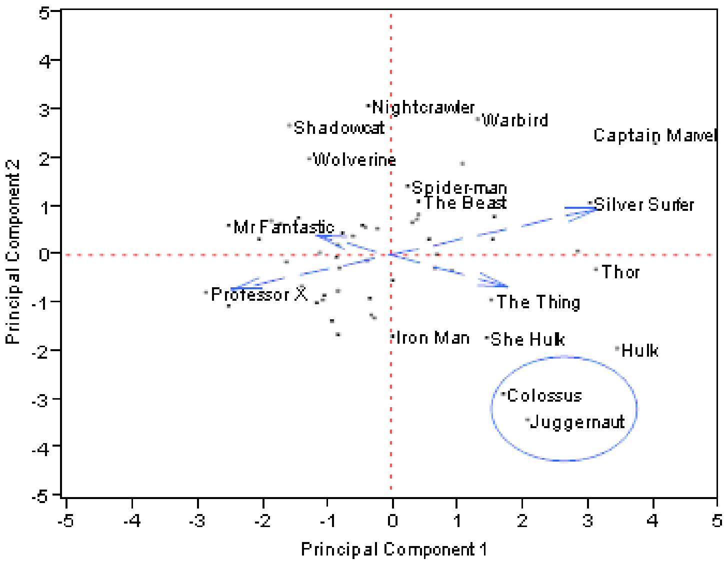
http://infoproc.blogspot.ca/2008/11/european-genetic-substructure.html
https://new.edu/resources/big-5-personality-traits

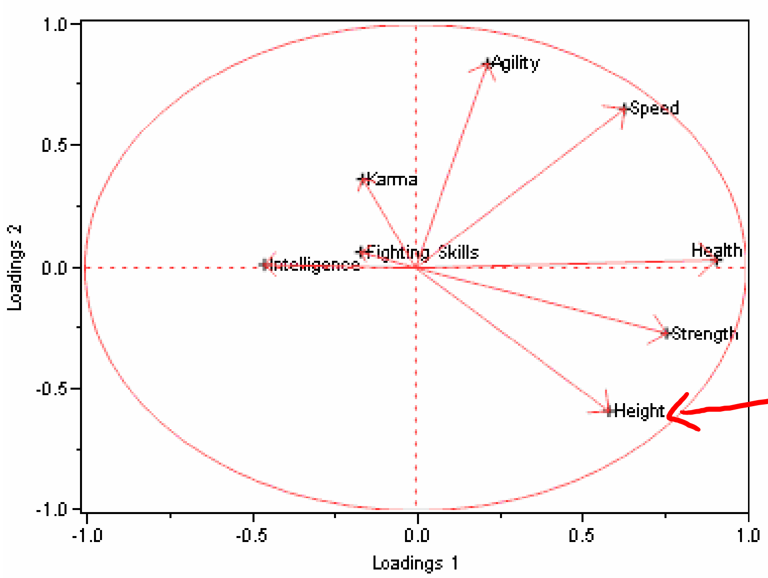
Last Week: Principal Component Analysis (PCA)

- PCA is a linear model for unsupervised learning.
- Represents features as linear combination of latent factors:

$$x_{ij} = w_j^T z_i \qquad x_i = W^T z_i$$

- Uses: dimensionality reduction, visualization, factor discovery.



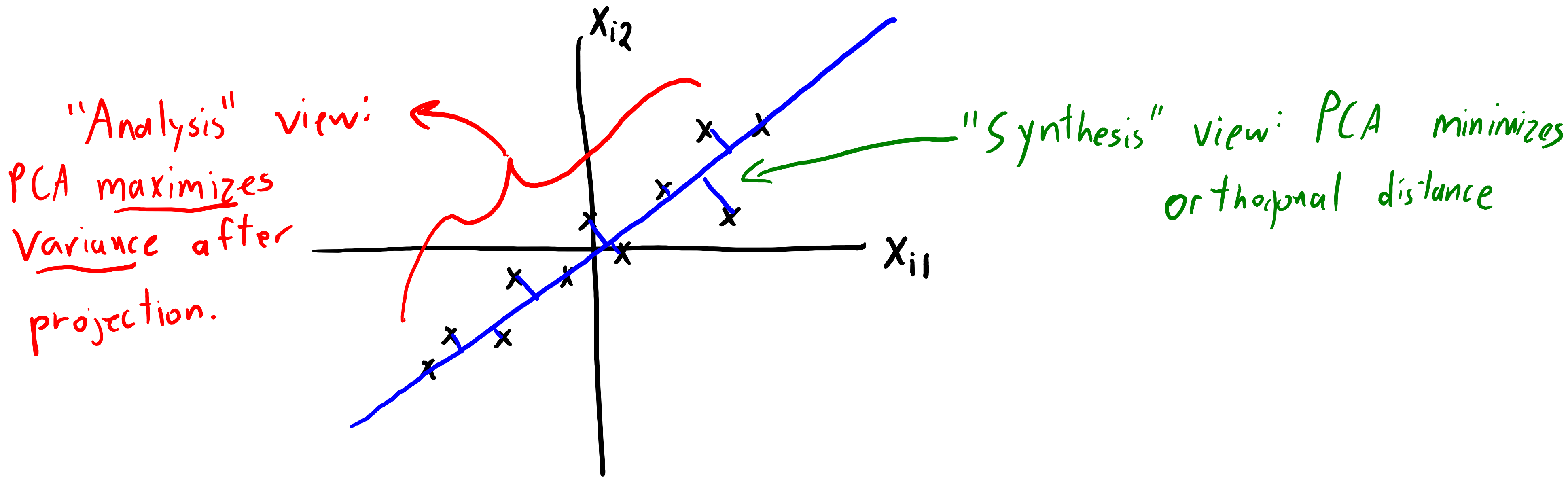


z_i value when $x_i = [1\ 0\ 0\ 0\ 0]$

http://www.prismtc.co.uk/superheroes-pca/

Maximizing Variance vs. Minimizing Error

- Our "synthesis" view that PCA minimizes approximation error.
  - Makes connection to k-means and our tricks for linear regression.



- Classic "analysis" view: PCA maximizes variance in z_i space.
  - You pick 'W' to explain as much variance in the data as possible.

Choosing Number of Latent Factors

- Common approach to choosing ‘k’:
  - Compute error with k=0: $\|X\|_F^2 = n * \text{var}(x_{ij})$ (remember that columns have mean of zero)
  - Compare to error with non-zero ‘k’:

$$\frac{\|ZW - X\|_F^2}{\|X\|_F^2}$$

  - Gives a number between 0 and 1, giving how much “variance remains”.
    - If you want to explain 90% of variance, choose smallest ‘k’ where ratio is < 0.10.

PCA Computation

- The PCA objective with general 'd' and 'k':

$$f(W,Z) = \sum_{i=1}^{n}\sum_{j=1}^{d}(w_j^T z_i - x_{ij})^2 = \|ZW - X\|_F^2$$

- 3 common ways to solve this problem:
  - Singular value decomposition (SVD) classic non-iterative approach.
  - Alternating between updating 'W' and updating 'Z'.

$\nabla_W f(W,Z) = Z^TZW - Z^TX$ so $W = (Z^TZ)^{-1}(Z^TX)$ and similarly $\nabla_Z f(W,Z) = ZWW^T - XW^T$ so $Z = XW^T(WW^T)^{-1}$

(writing gradient as a matrix)

  - Stochastic gradient: gradient descent based on random 'i' and 'j'.
    - (Or just plain gradient descent).
- Not convex, all these methods work with random initialization.

PCA Computation

- The PCA objective with general 'd' and 'k':

$$f(W,Z) = \sum_{i=1}^{n}\sum_{j=1}^{d}(w_j^T z_i - x_{ij})^2 = \|ZW - X\|_F^2$$

  – Where we've subtracted mean μ_j from each feature.

- At test time, to find optimal 'Z' given 'W' for new data:

Given factors 'W' and test data $\hat{X}$:

Subtract training mean μ_j for each feature 'j': $\hat{x}_i \leftarrow \hat{x}_i - \mu$

Solve for 'Z' given 'W': $Z = \hat{X}W^T(WW^T)^{-1}$

(If $k=1$ then $z_i = \frac{w_c^T x_i}{w_c^T w_c}$)

PCA Non-Uniqueness

- We have the scaling problem:

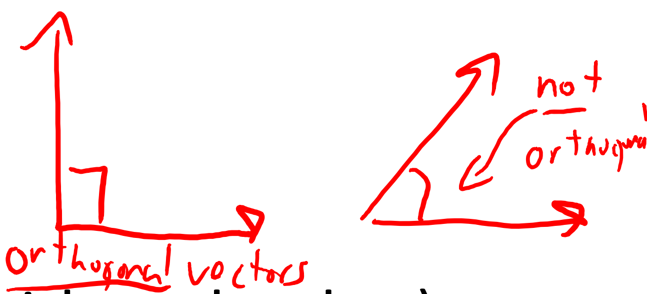
We get same $f(W,Z)$ if you replace 'W' by αW
and 'Z' by $(\frac{1}{\alpha})Z$ for any $\alpha \neq 0$

$$\left(\frac{1}{\alpha} Z\right)(\alpha W) = ZW$$

A standard fix: require that $\|w_c\| = 1$ for all factors 'c'.
↳ row 'c' of 'W'

PCA Non-Uniqueness

Orthogonality when $d=2$



- But with multiple PCs, we have new problems:
  - Factors could be non-orthogonal (components interfere with each other):

For $d=2$ and $k=2$
an optimal solution is $W=\begin{bmatrix} 1 & 0.99 \\ 0.99 & 1 \end{bmatrix}$ factors are almost identical

But with $d=2$ and $k=2$ we could equivalently take $W=\underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\text{Very interpretable}}$

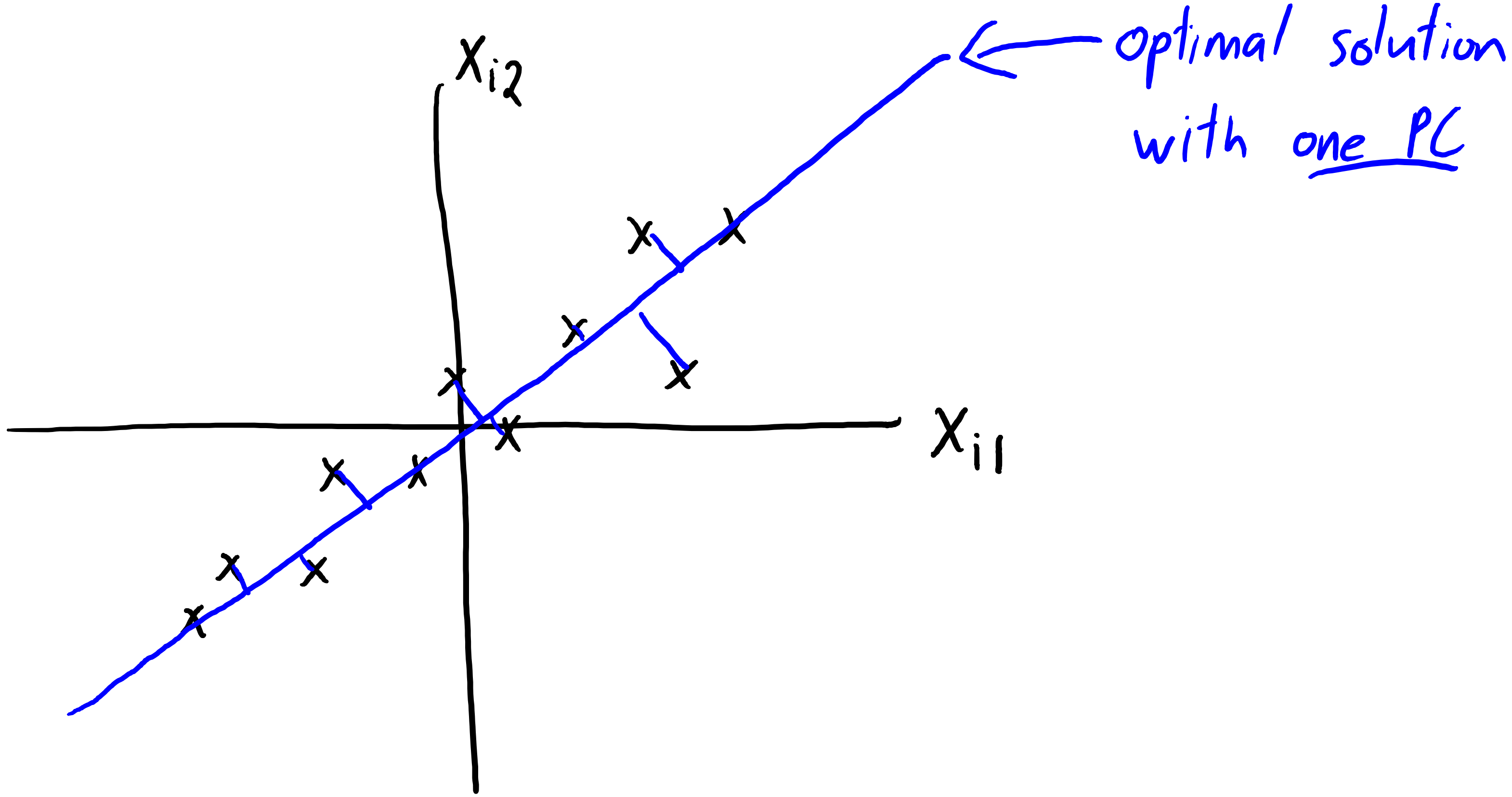
The standard fix is requiring orthogonal factors: $w_c^T w_{c'} = 0$ when $c \neq c'$

  - You can still "rotate" the factors and also have label switching.
    - A fix is to fit the PCs sequentially (can be done with SVD approach):

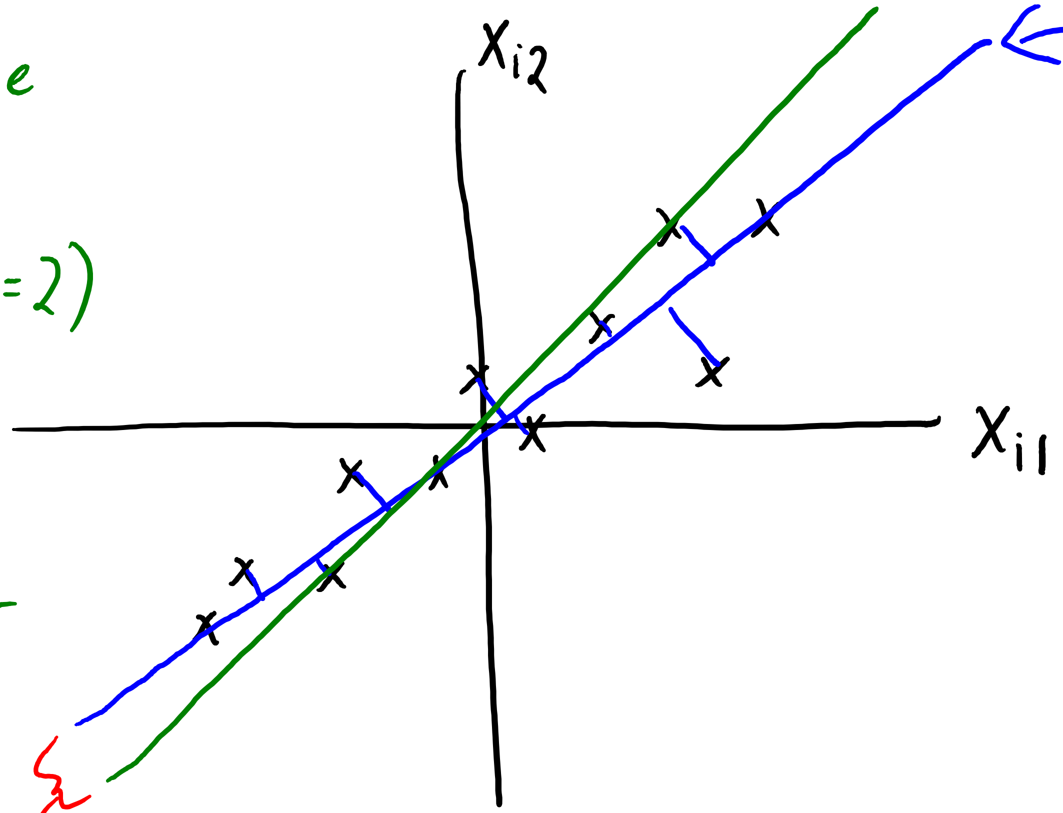
1. Find "first" PC w_c that minimizes $\|zw_c^T - X\|_F^2$ (PCA with $k=1$)
2. Fix "first" PC w_1 and find w_c minimizing $\|ZW-X\|_F^2$ where $w_1^T w_c = 0$
3. Fix "first" and "second" PC and find w_c with $w_1^T w_c = 0$ and $w_2^T w_c = 0$

⋮

Basis, Orthogonality, Sequential Fitting



Basis, Orthogonality, Sequential Fitting



optimal solution with one PC

Any non-parrallel line gives optimal solution to second PC (when d=2)

I can get 0 error on every data point.

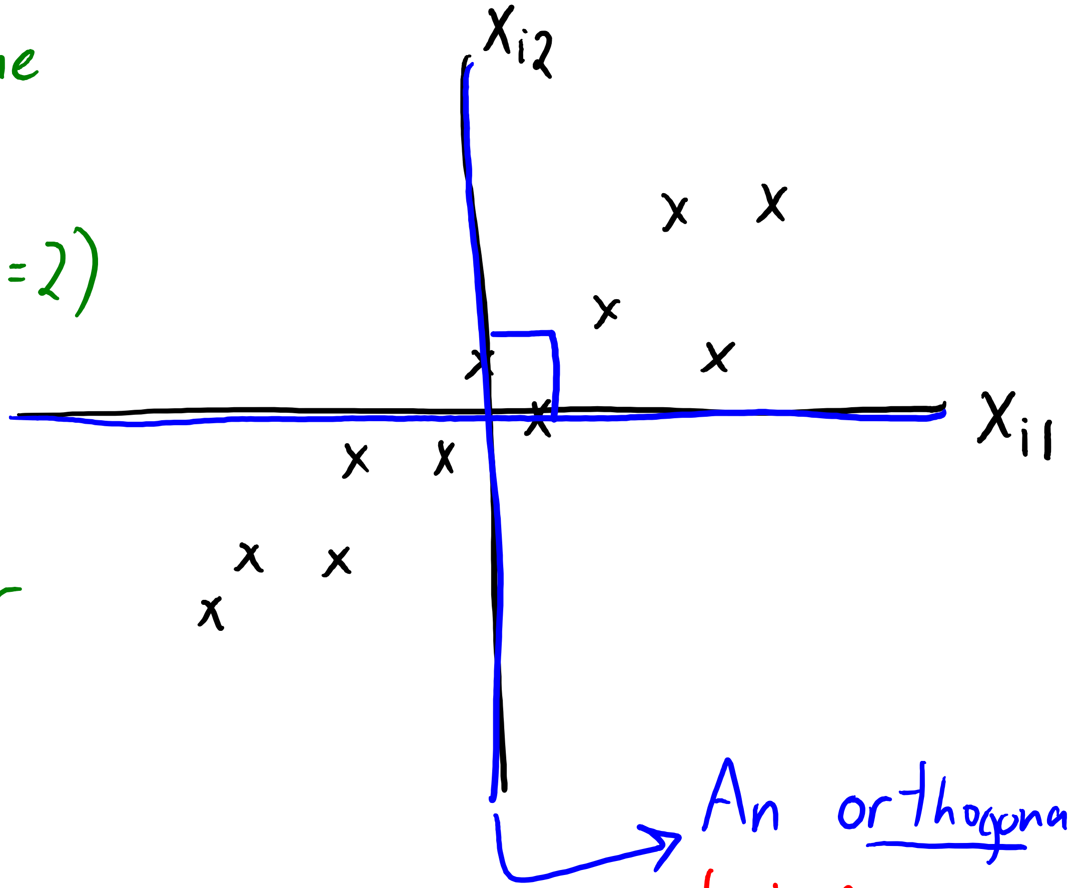
An optimal solution but not orthogonal.

(both PCs give similar information)

Basis, Orthogonality, Sequential Fitting

Any non-parrallel line gives optimal solution to second PC (when $d=2$)

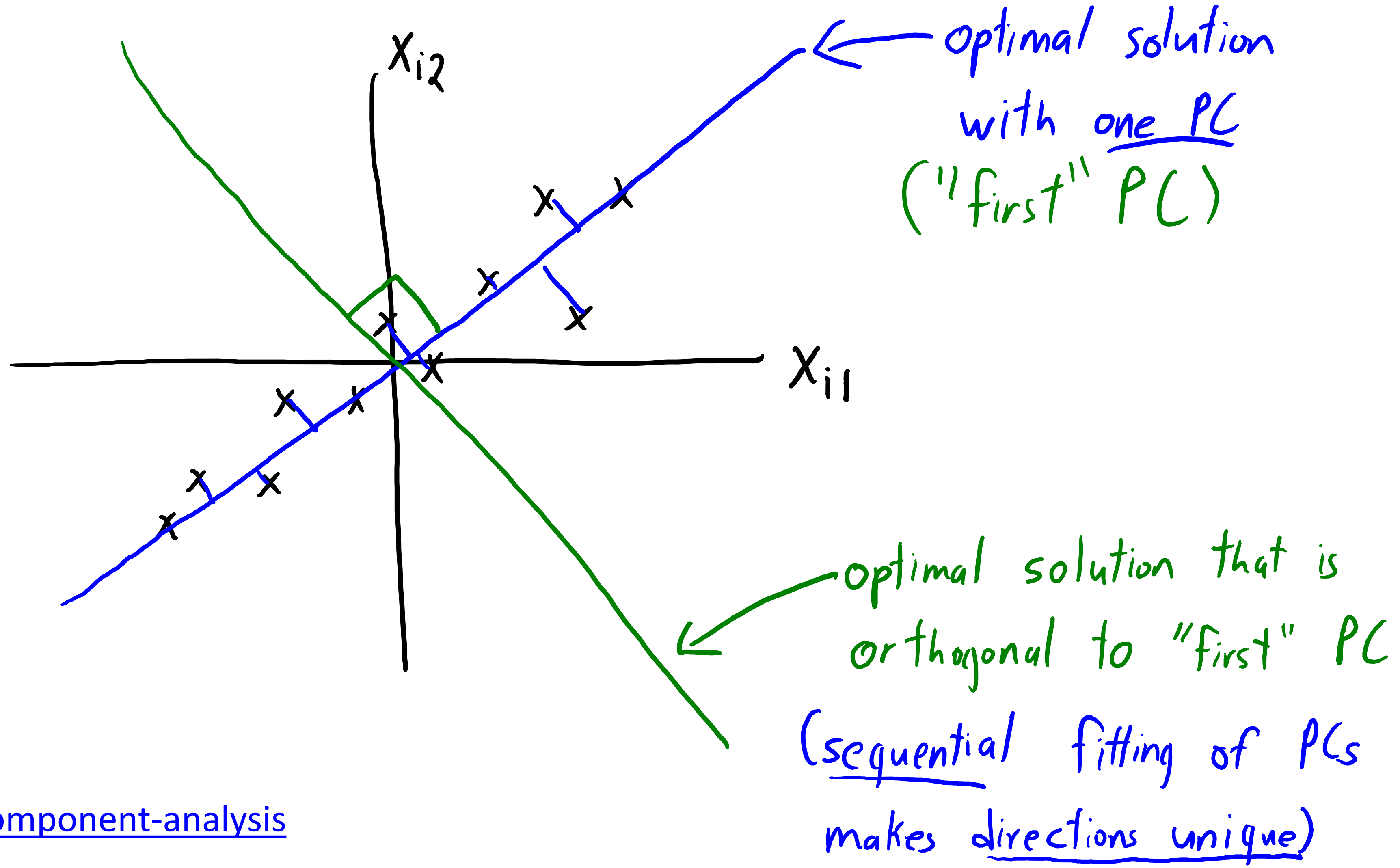
I can get 0 error on every data point.



An orthogonal solution (PCs are not redundant)

but PCs have nothing to do with data

Basis, Orthogonality, Sequential Fitting



http://setosa.io/ev/principal-component-analysis

Notice that if we require $\|w_c\| = 1$ then $w_c^T w_c = 1$

and if we also require orthogonality $w_c^T w_{c'} = 0$ for $c \neq c'$ then:

$$WW^T = \begin{bmatrix} — & w_1^T & — \\ — & w_2^T & — \\ & \vdots & \\ — & w_k^T & — \end{bmatrix} \begin{bmatrix} | & | & & | \\ w_1 & w_2 & \cdots & w_k \\ | & | & & | \end{bmatrix} = \begin{bmatrix} w_1^T w_1 & w_1^T w_2 & \cdots & w_1^T w_k \\ w_2^T w_1 & w_2^T w_2 & \cdots & w_2^T w_k \\ \vdots & & \ddots & \vdots \\ w_k^T w_1 & w_k^T w_2 & \cdots & w_k^T w_k \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & & \vdots \\ 0 & 0 & 1 & \ddots & \\ \vdots & & & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} = I$$

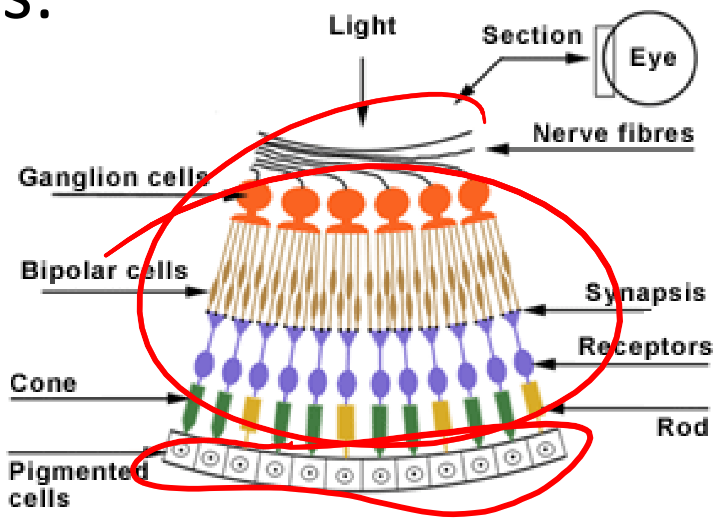
So finding Z simplifies: $Z = XW^T(WW^T)^{-1} = XW^T$

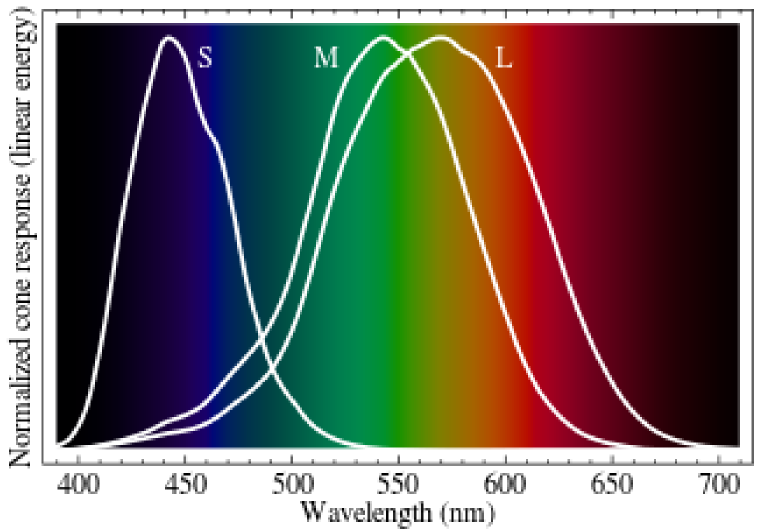
(If $k=1$ then $z_i = \underbrace{w_c^T x_i}_{w_c^T w_c} = w_c^T x_i$)

Do need all this math?

Colour Opponency in the Human Eye

- Classic model of the eye is with 4 photoreceptors:
  - Rods (more sensitive to brightness).
  - L-Cones (most sensitive to red).
  - M-Cones (most sensitive to green).
  - S-Cones (most sensitive to blue).
- Two problems with this system:
  - Correlation between receptors (not orthogonal).
    - Particularly between red/green.
  - We have 4 receptors for 3 colours.

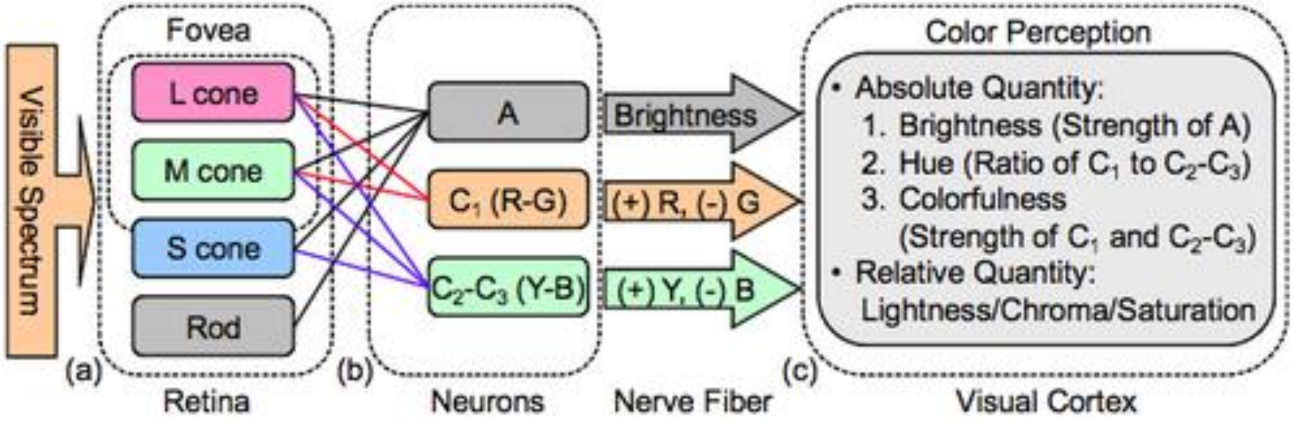


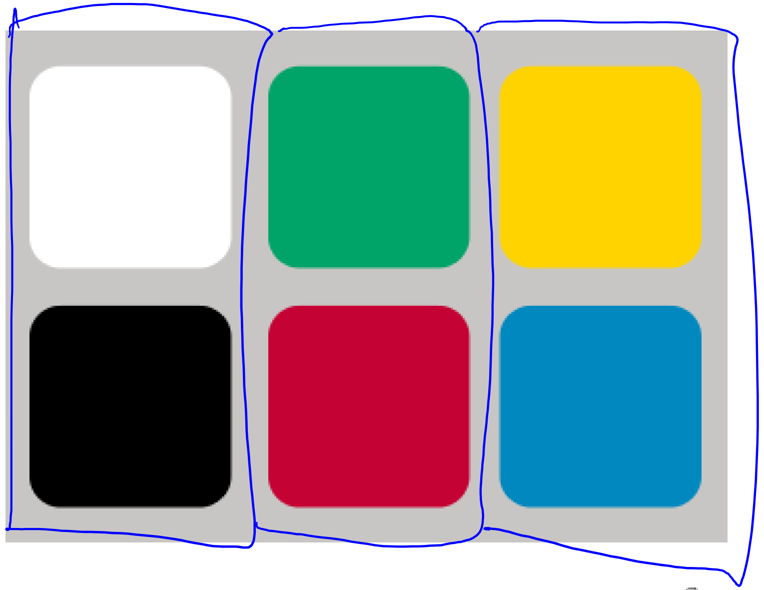


http://oneminuteastronomer.com/astro-course-day-5/
https://en.wikipedia.org/wiki/Color_visio

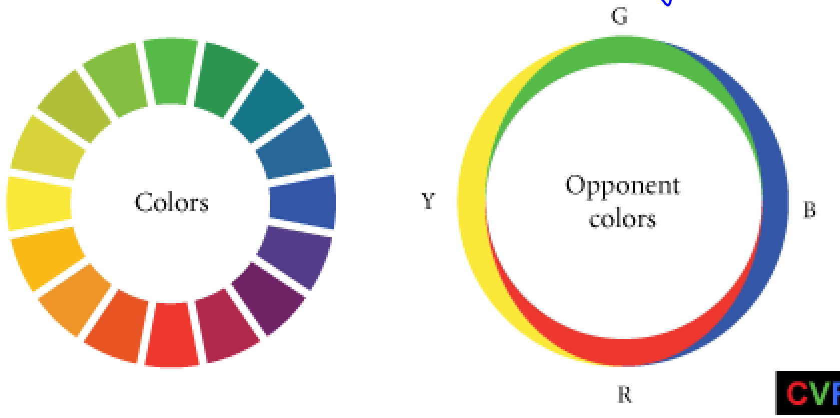
Colour Opponency in the Human Eye

- Bipolar and ganglion cells seem to code using "opponent colors":
  - 3-variable orthogonal basis:





- This is similar to PCA (d = 4, k = 3).



http://oneminuteastronomer.com/astro-course-day-5/
https://en.wikipedia.org/wiki/Color_visio
http://5sensesnews.blogspot.ca/

Colour Opponency Representation

For this pixel, eye gets 4 signals

Can represent 4 original values with these 3 z_i values and matrix 'W'

$\approx w_1$

First row of W (First PC)

Analogous to means in k-means.

brightness

$+ w_2$

Second row (4×1)

red/green

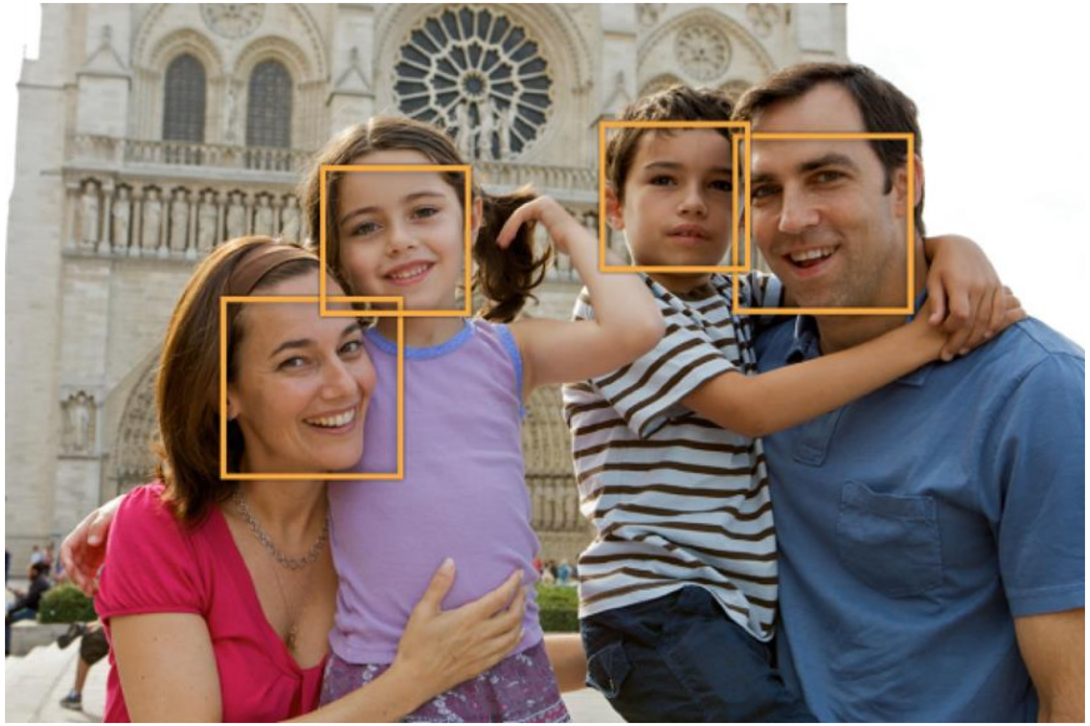
$+ w_3$

Third row (4×1)

blue/yellow

Application: Face Detection

- Consider problem of face detection:



- Classic methods use “eigenfaces” as basis:
  - PCA applied to images of faces.

https://developer.apple.com/library/content/documentation/GraphicsImaging/Conceptual/CoreImaging/ci_detect_faces/ci_detect_faces.html

Eigenfaces

- Collect a bunch of images of faces under different conditions:

Each row of X will be pixels in one image:

$X =$

If have 'n' images that are 'm' by 'm' then X is 'n' by m^2.

Eigenfaces

Compute <u>mean</u> μ_j of each column.

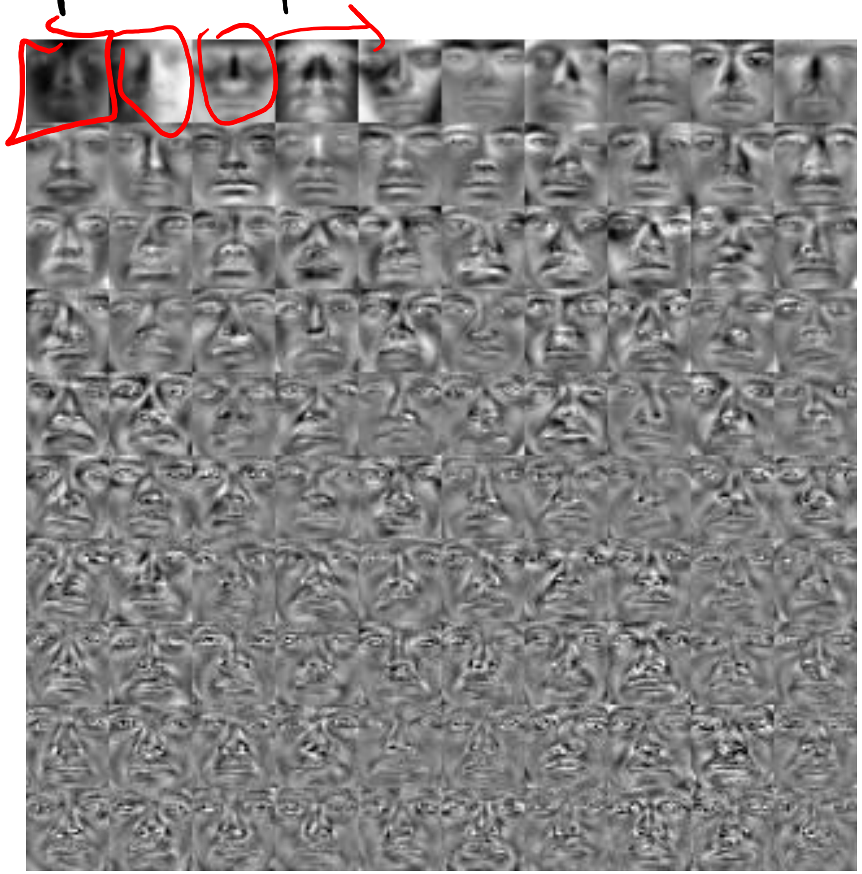
Replace each x_{ij} by $x_{ij} - \mu_j$

Each row of X will be pixels in <u>one</u> image:

$$X = \begin{bmatrix} \text{———} & x_1 - \mu & \text{———} \\ \text{———} & x_2 - \mu & \text{———} \\ & \vdots & \\ \text{———} & x_n - \mu & \text{———} \end{bmatrix}$$

Eigenfaces

Compute top 'k' PCs on centered data:

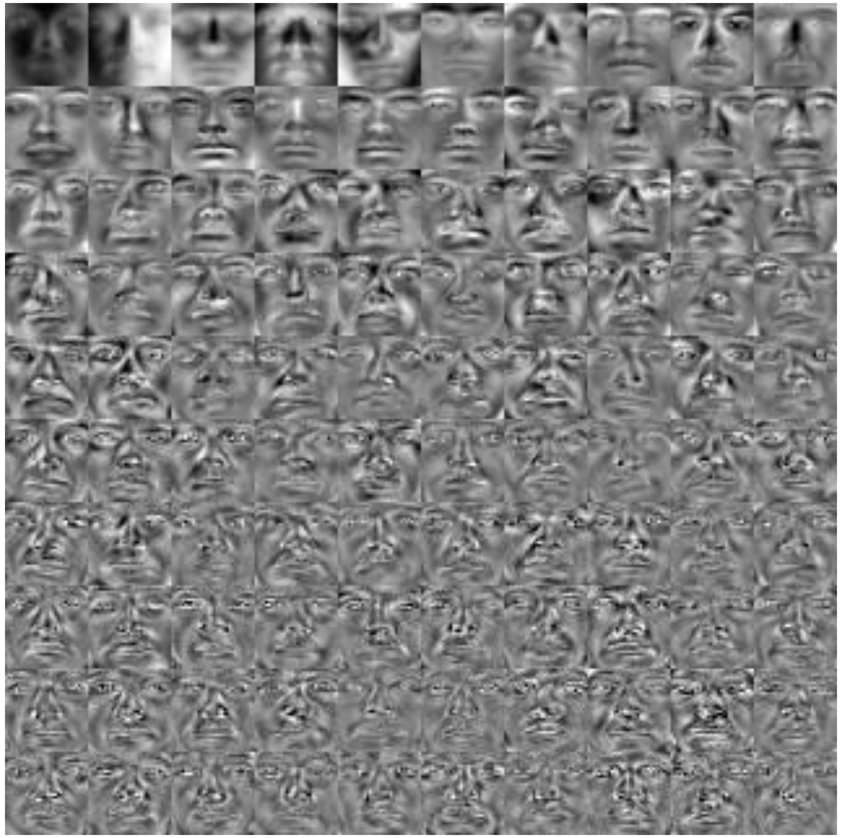


Each row of X will be pixels in <u>one</u> image:

$$X = \begin{bmatrix} — & x_1 - \mu & — \\ — & x_2 - \mu & — \\ & \vdots & \\ — & x_n - \mu & — \end{bmatrix}$$

Eigenfaces

Compute top 'k' PCs on centered data:



"Eigenface" representation:

 =  $+ z_{i1}$  $+ z_{i2}$  $+ z_{i3}$  $+ \dots$

x_i μ PC1 (first row of W)   PC2   PC3

Eigenfaces

100 of the original faces:

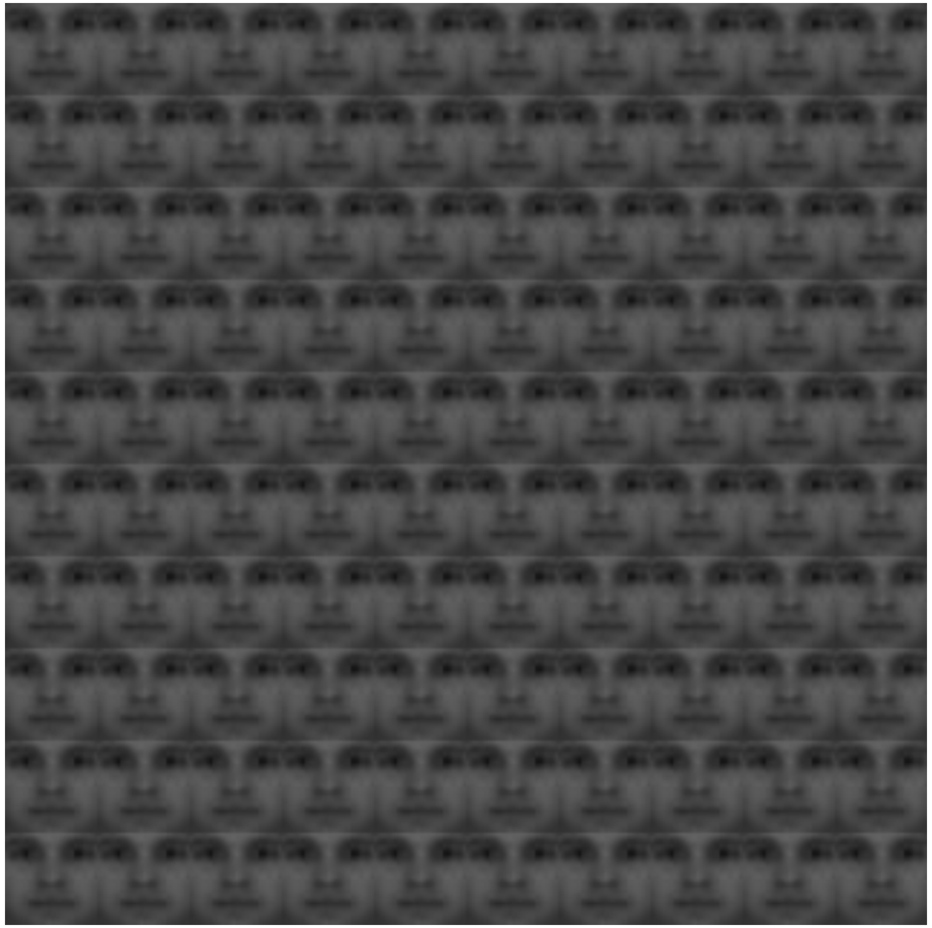
"Eigenface" representation:

 =  $+ z_{i1}$  $+ z_{i2}$  $+ z_{i3}$  $+ \dots$

x_i $\quad$ μ $\quad$ PC1 (first row of W) $\quad$ PC2 $\quad$ PC3

Eigenfaces

Reconstruction with $k = 0$



Variance explained: 0%

"Eigenface" representation:

x_i

=

μ

$+ z_{i1}$

PC1

(first row of W)

$+ z_{i2}$

PC2

$+ z_{i3}$

PC3

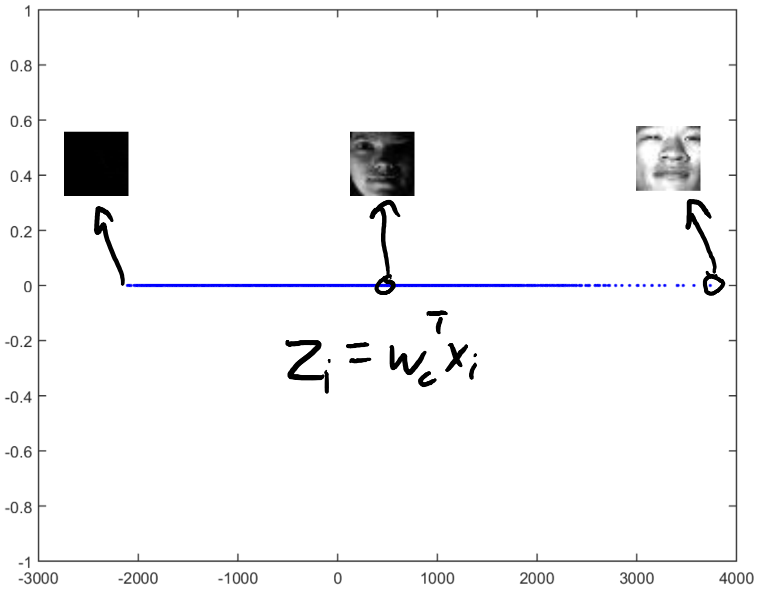
$+ \dots$

Eigenfaces

Reconstruction with $k=1$

Variance explained: 34%

PCA Visualization:



$z_i = w_c^T x_i$

"Eigenface" representation:

 $=$  $+ z_{i1}$  $+ z_{i2}$  $+ z_{i3}$  $+ \cdots$

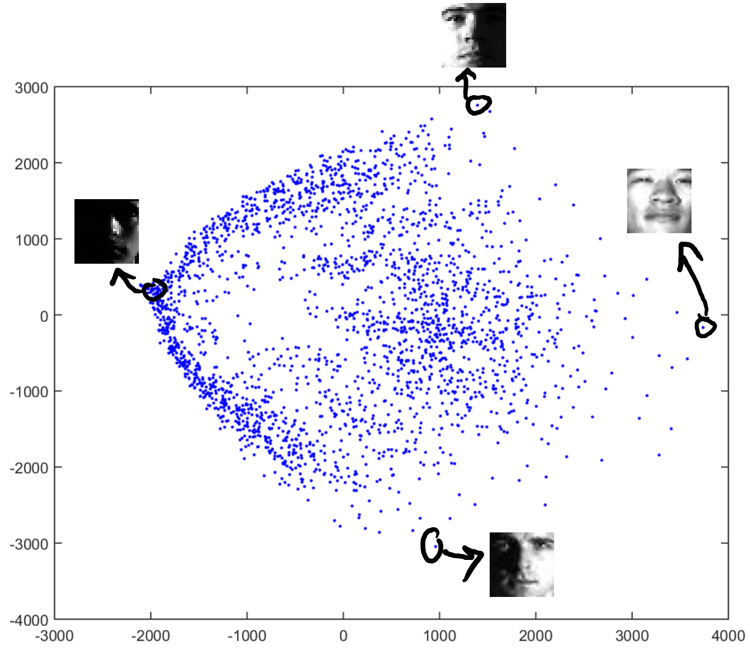
x_i $\quad$ μ $\quad$ PC1 (first row of W) $\quad$ PC2 $\quad$ PC3

Eigenfaces

Reconstruction with $k=2$

Variance explained: 71%

PCA Visualization:



"Eigenface" representation:

 $=$  $+ z_{i1}$  $+ z_{i2}$  $+ z_{i3}$  $+ \dots$

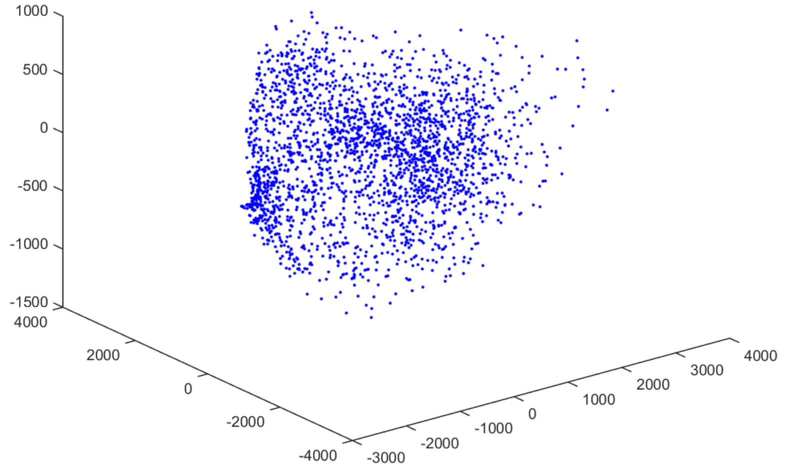
x_i $\quad \mu$ $\quad$ PC1 (first row of W) $\quad$ PC2 $\quad$ PC3

Eigenfaces

Reconstruction with $k = 3$

Variance explained: 76%

PCA Visualization:



"Eigenface" representation:

x_i

$=$

μ

$+ z_{i1}$

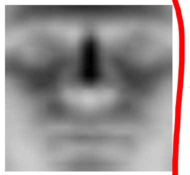
PC1

(first row of W)

$+ z_{i2}$

PC2

$+ z_{i3}$



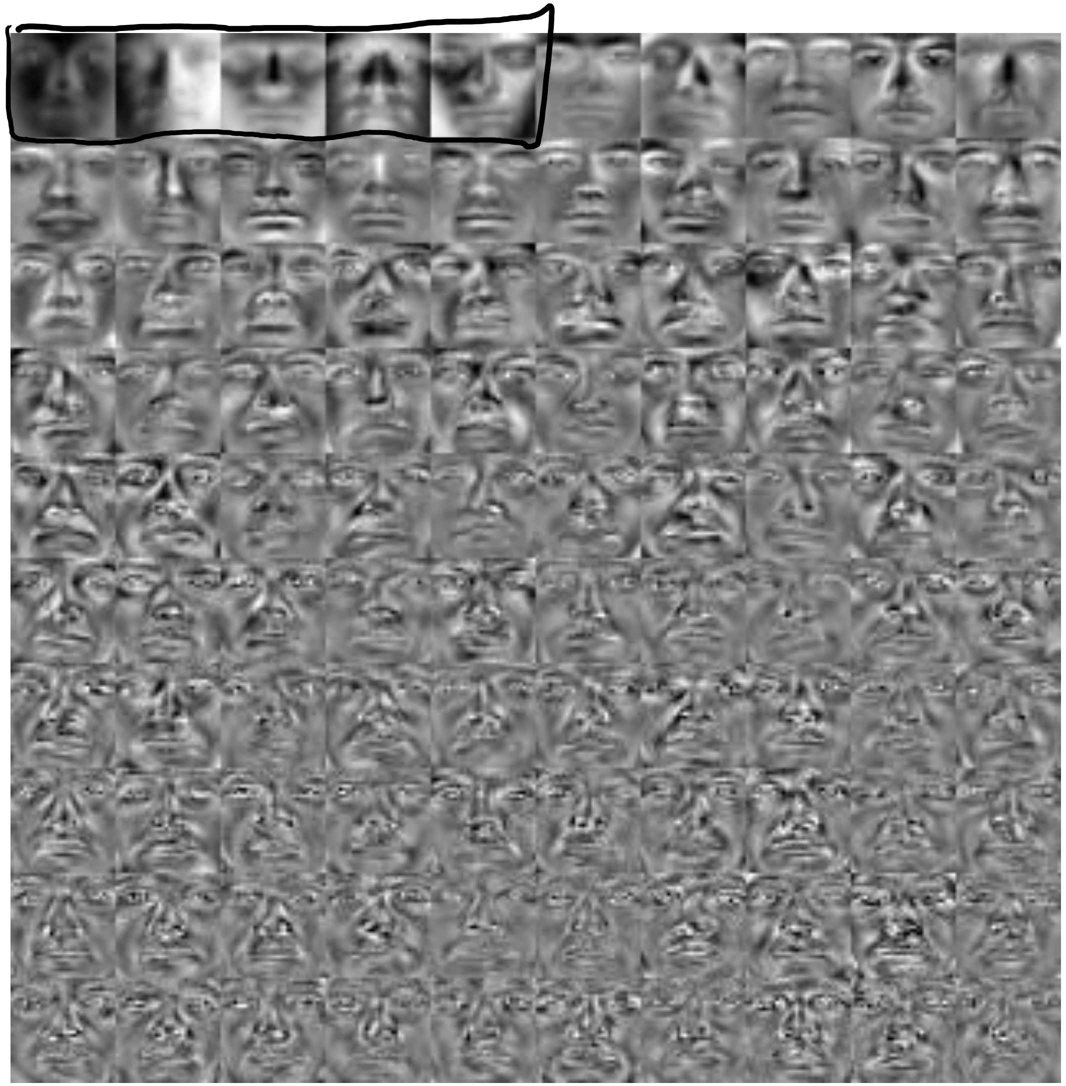
PC3

$+ \dots$

Eigenfaces

Reconstruction with $k=5$

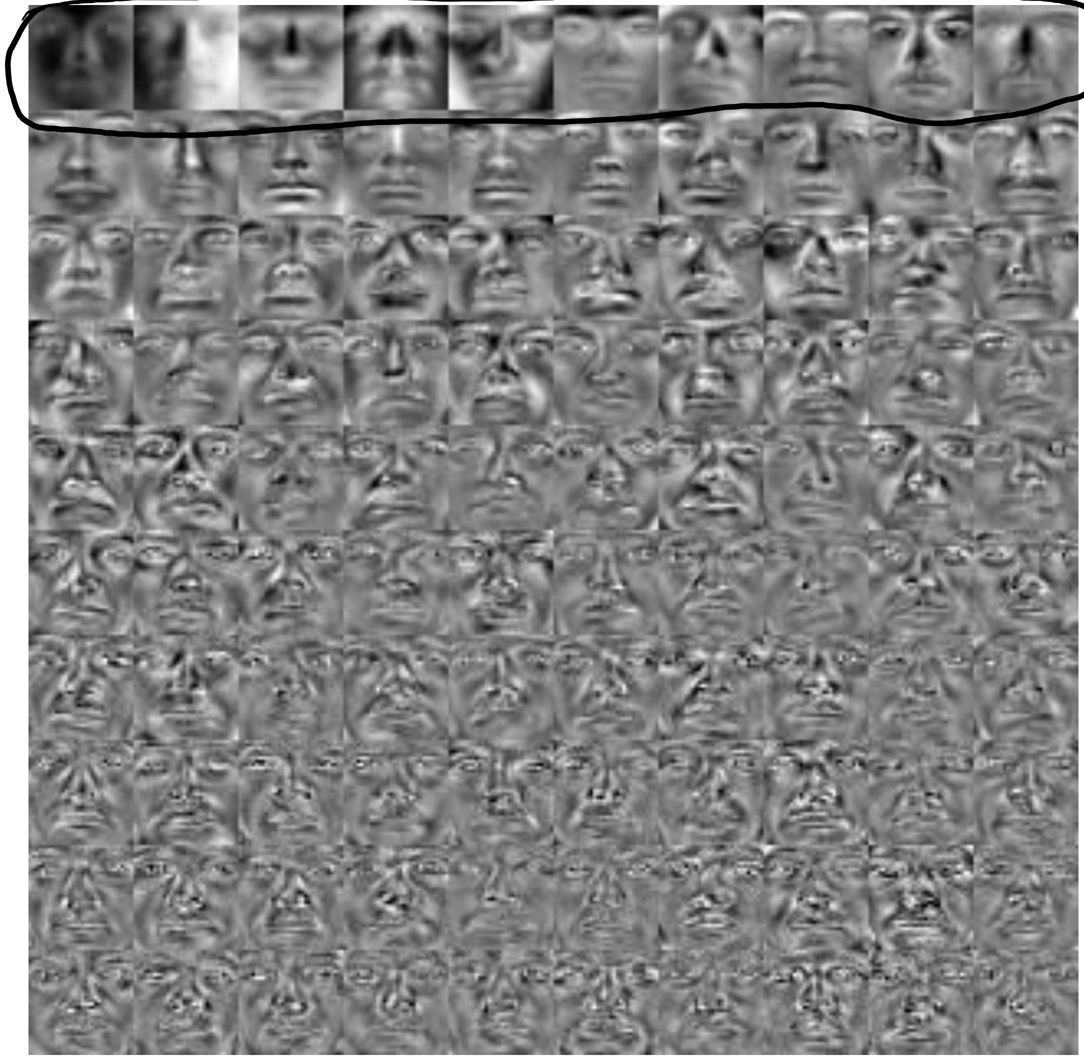
Variance explained: 80%



Eigenfaces

Reconstruction with k=10

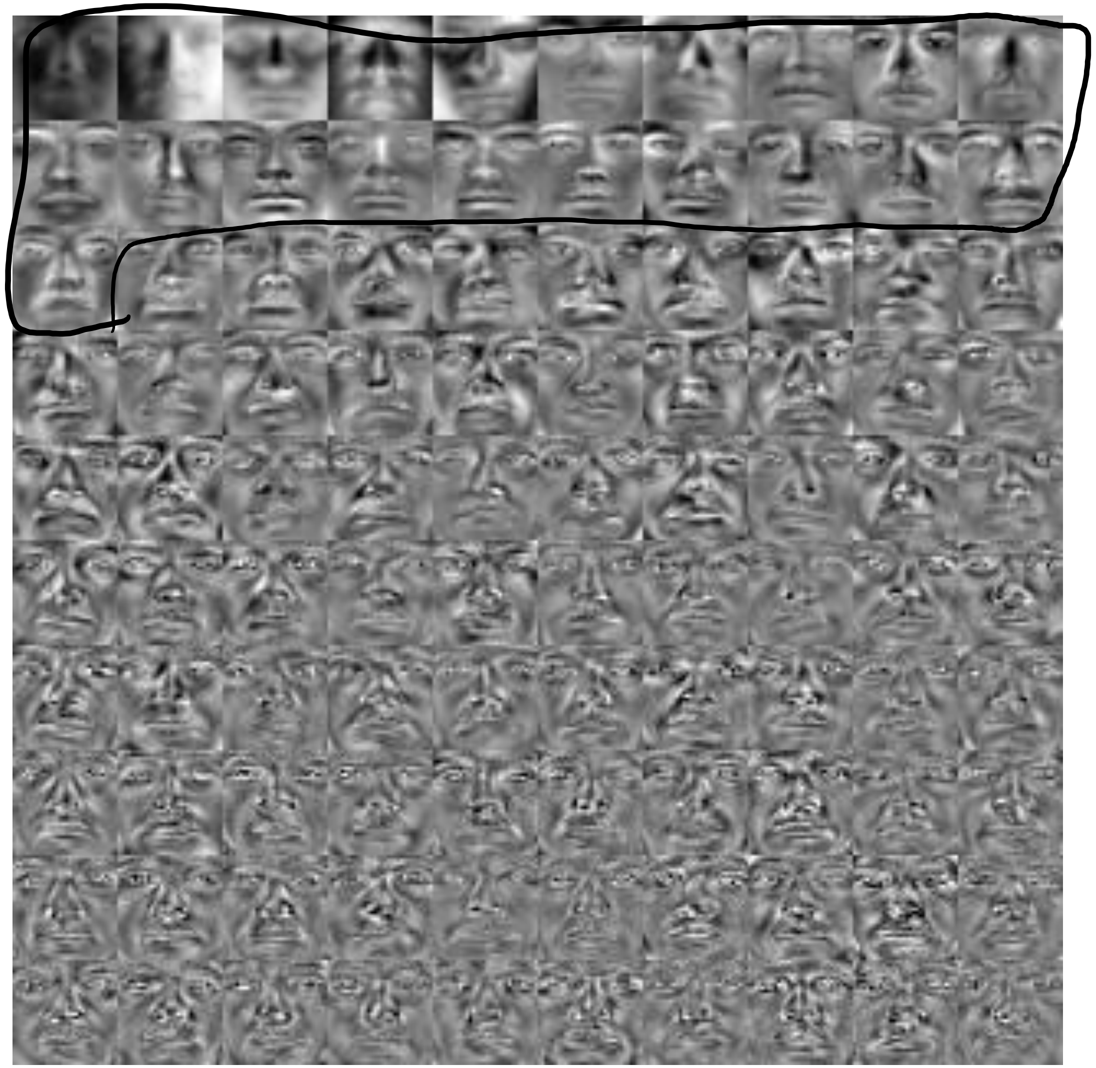
Variance explained: 85%



Eigenfaces

Reconstruction with $k=21$

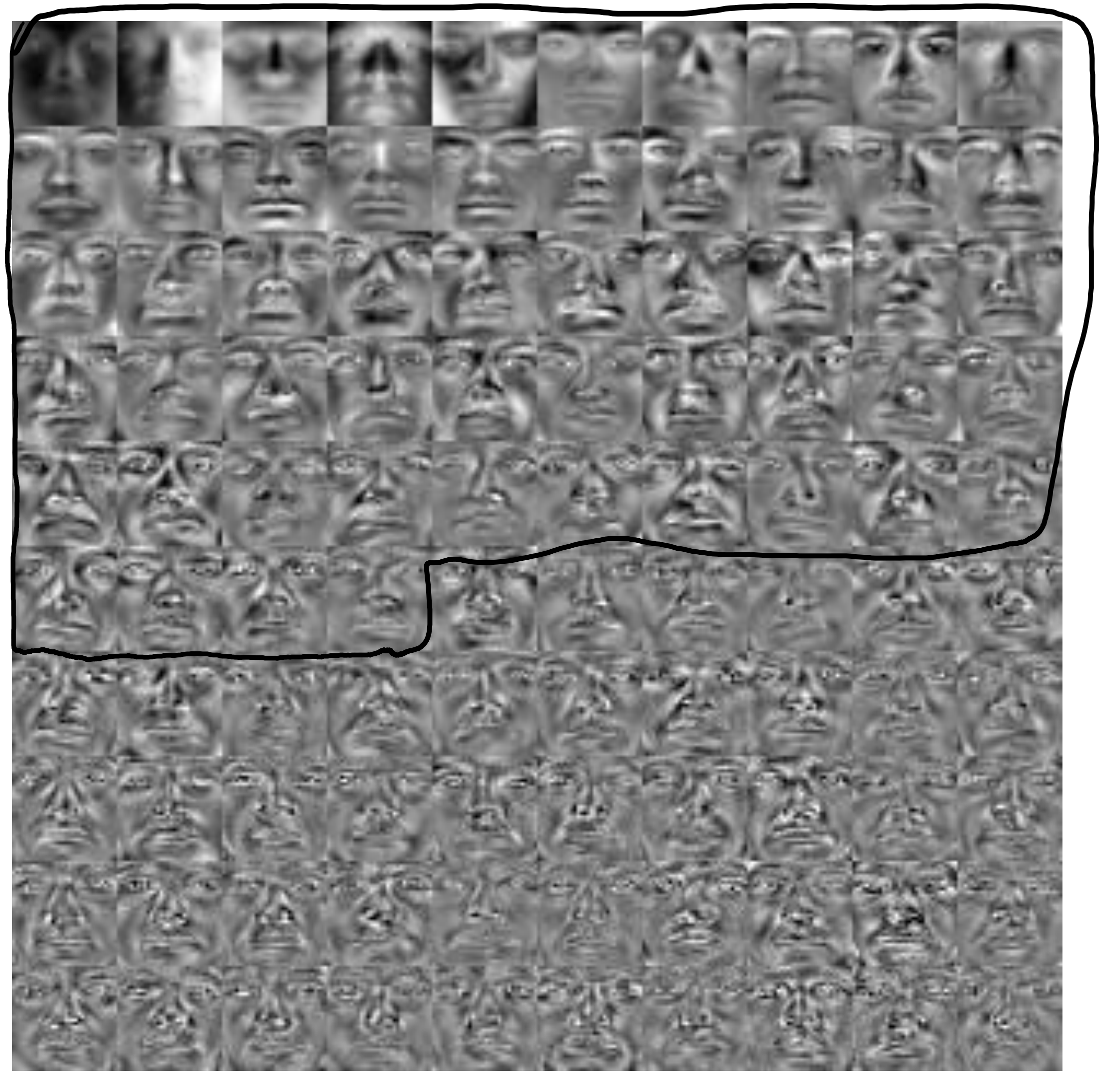
Variance explained: 90%



Eigenfaces

Reconstruction with k = 54

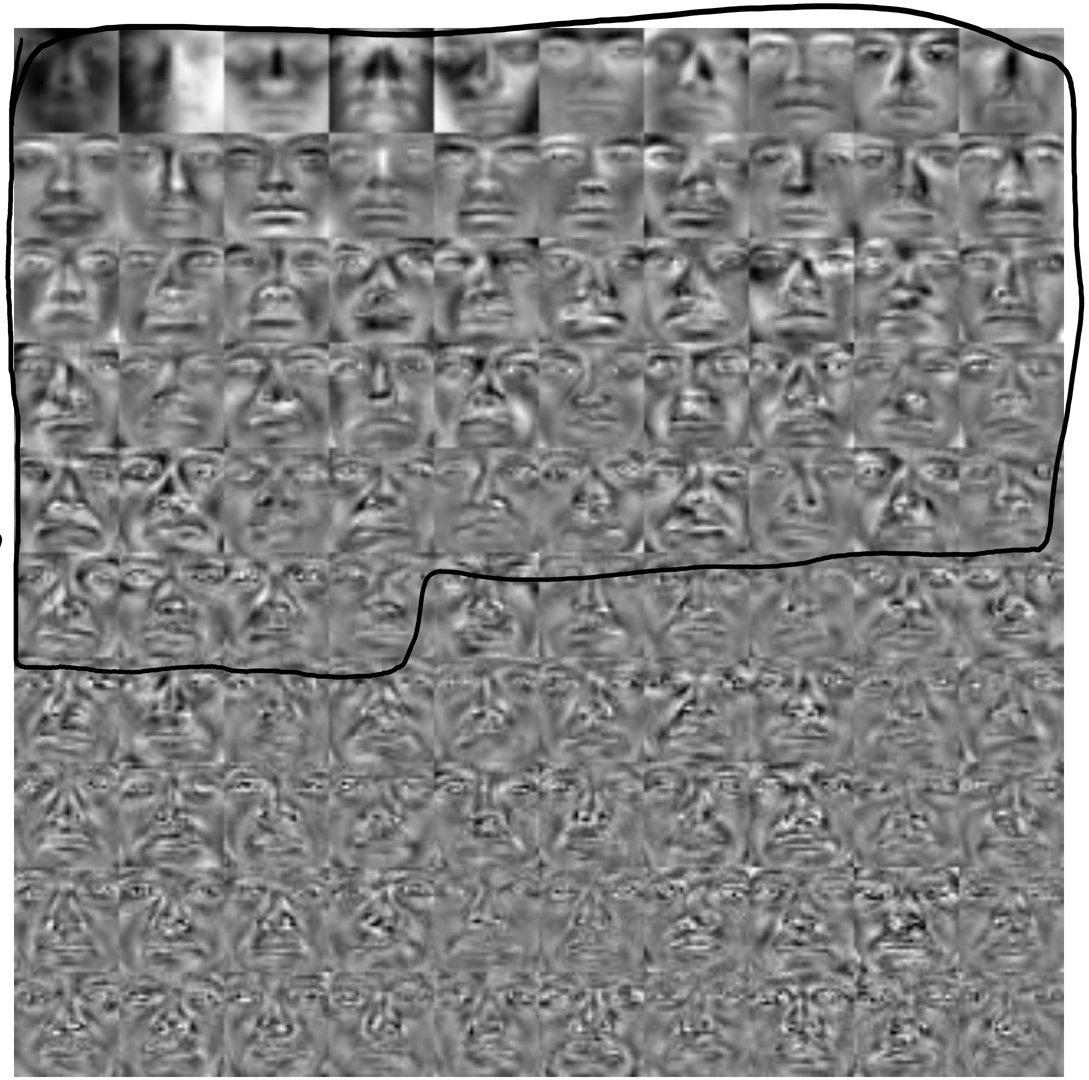
Variance explained: 95%



Eigenfaces

Original Images again:

Plus these "eigenfaces" and the mean.



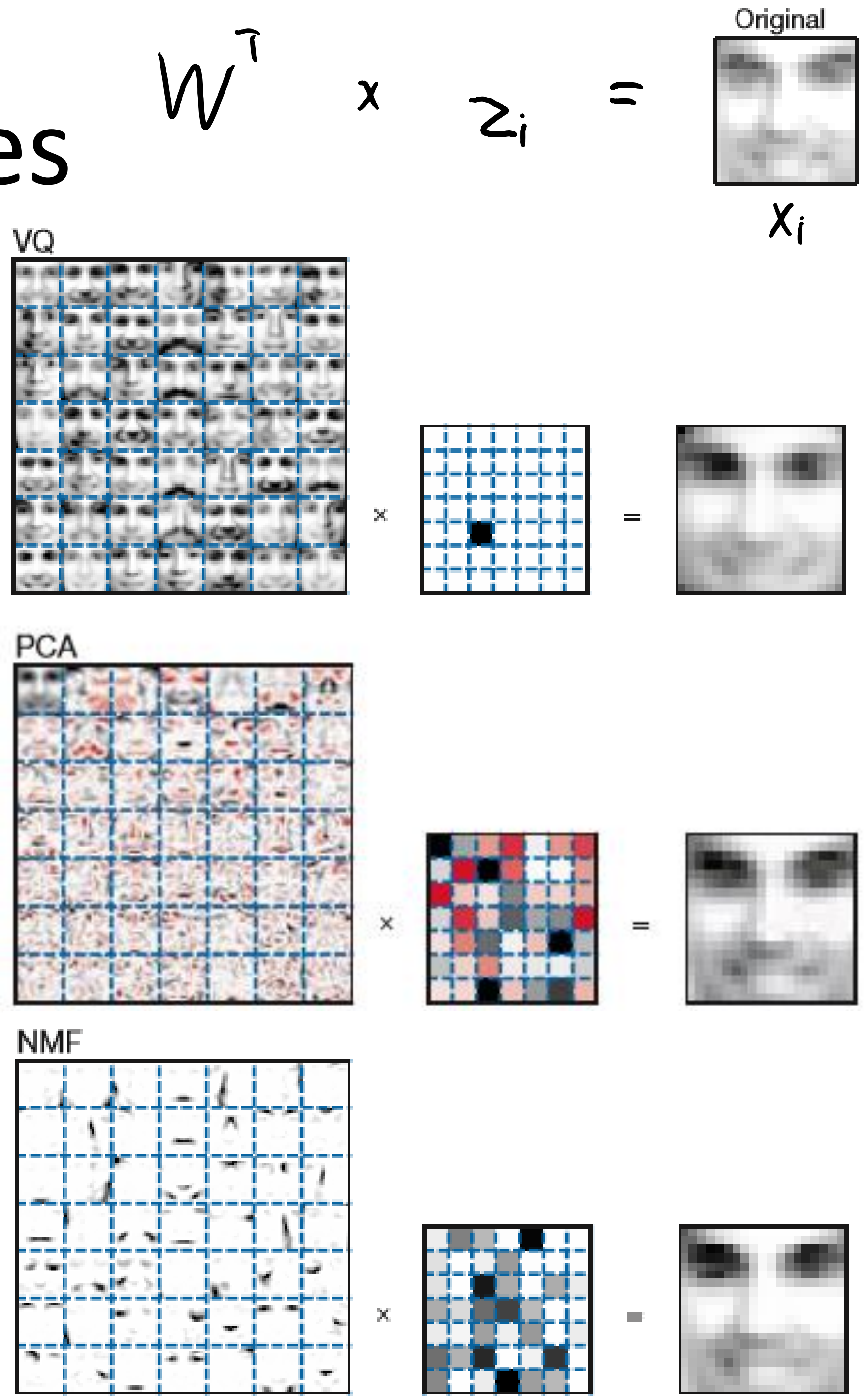
We can replace 1024 x_i values by 54 z_i values

Representing Faces

But how *should* we represent faces?

- K-means:
  - 'Grandmother cell': one neuron = one face.
  - Almost certainly not true: too few neurons.
- PCA:
  - "Distributed representation".
    - Coded by pattern of group of neurons.
    - Can represent more concepts.
  - But PCA uses positive/negative cancelling parts.
- Non-negative matrix factorization (NMF):
  - Latent-factor where W and Z are non-negative.
  - Example of "sparse coding":
    - Coded by small number of neurons in group.
  - NMF makes object out of of 'parts'.

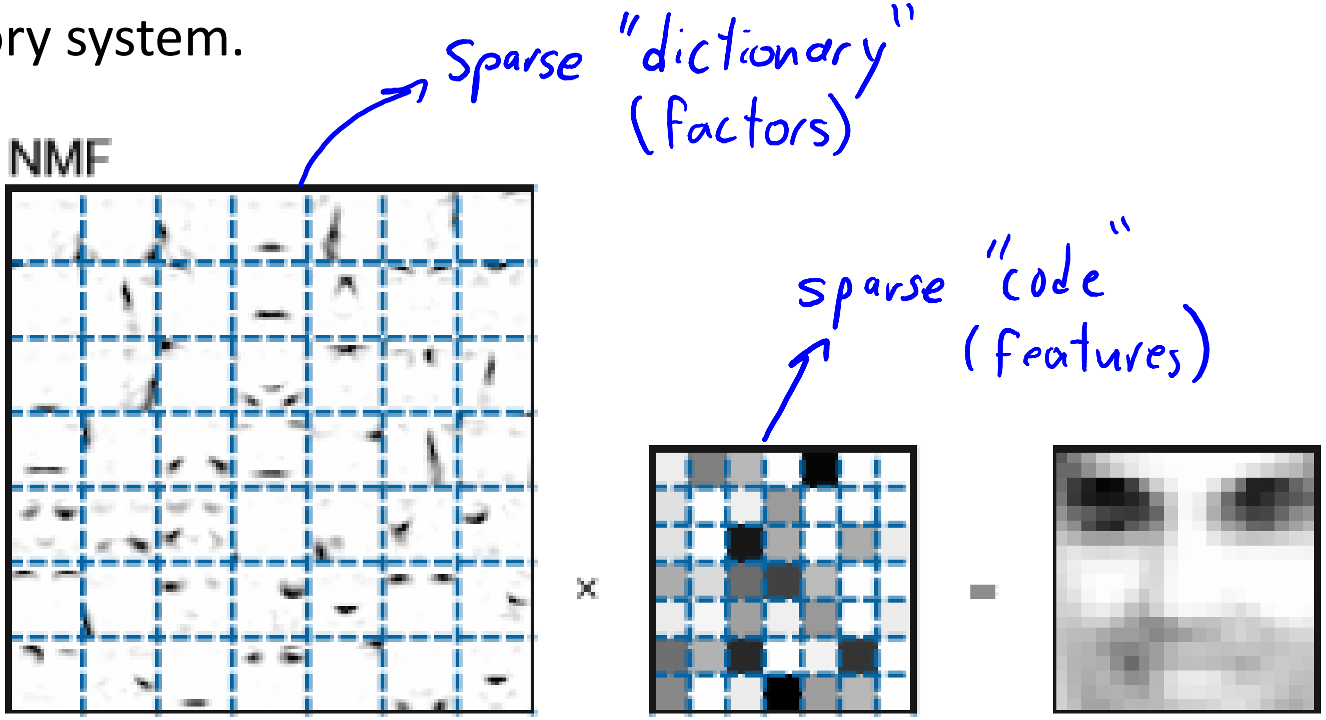
$W^T \times z_i = x_i$



http://www.columbia.edu/~jwp2128/Teaching/W4721/papers/nmf_nature.pdf

Representing Faces

- Why sparse coding?
  - ‘Parts’ are intuitive, and brains seem to use sparse representation.
  - Energy efficiency if using sparse code.
  - Increase number of concepts you can memorize?
    - Some evidence in fruit fly olfactory system.



http://www.columbia.edu/~jwp2128/Teaching/W4721/papers/nmf_nature.pdf

Summary

- Analysis view of PCA is that it maximizes variance.
  - We can choose ‘k’ to explain x% of the variance in the data.
- Orthogonal basis and sequential fitting of PCs:
  - Leads to non-redundant PCs with unique directions.
- Biological motivation for orthogonal and/or sparse latent factors.
- Non-negative matrix factorization leads to sparse LFM.

- Next time: modifying PCA so it splits faces into ‘eyes’, ‘mouths’, etc.