Informative projections

DSE 220

Dimensionality reduction

Why reduce the number of features in a data set?

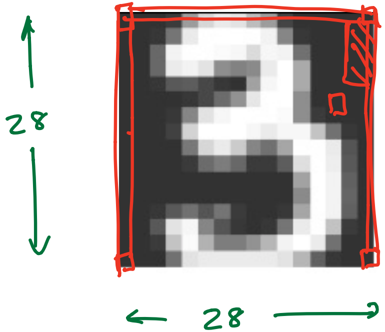
1. It reduces storage and computation time.
2. High-dimensional data often has a lot of redundancy.
3. Remove noisy or irrelevant features.

Dimensionality reduction

Why reduce the number of features in a data set?

1. It reduces storage and computation time.
2. High-dimensional data often has a lot of redundancy.
3. Remove noisy or irrelevant features.

Example: are all the pixels in an image equally informative?



If we were to choose a few pixels to discard, which would be the prime candidates?

① Pixels that always have the same value.

Next: ② Pixels with lowest variance

Eliminating low variance coordinates

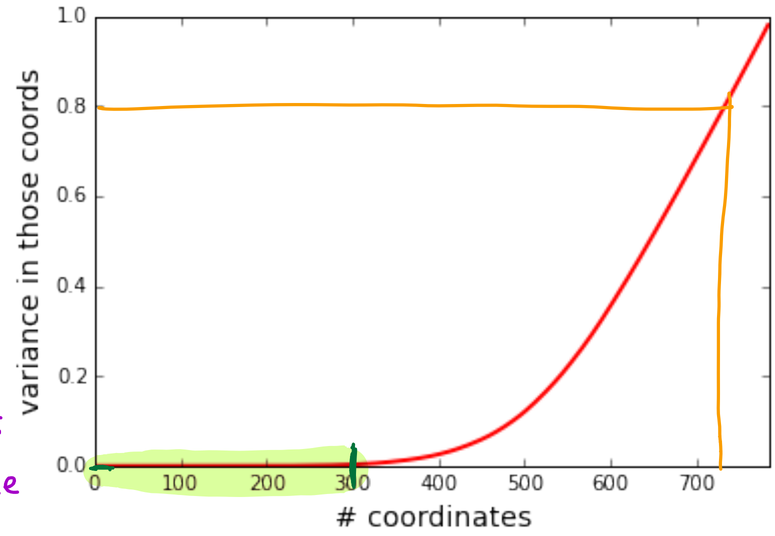
MNIST: what fraction of the total variance lies in the 100 (or 200, or 300) coordinates with lowest variance?

Pixels $X_1, X_2, \ldots, X_{784}$

$\mathrm{var}(X_i)$ = variance in i^{th} pixel

Total variance

$= \sum_{i=1}^{784} \mathrm{var}(X_i)$

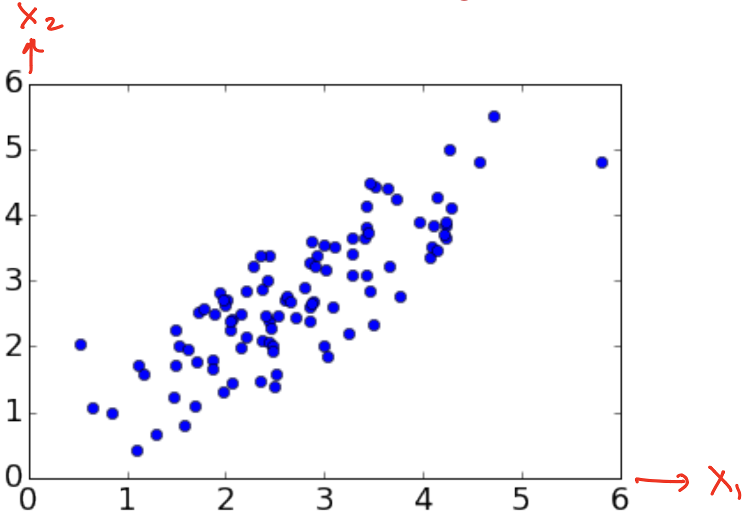


Can drop $\approx$ 300 pixels while losing very little information.

But this strategy is not suitable for reducing dimension to 50, or 100, or 200.

The effect of correlation

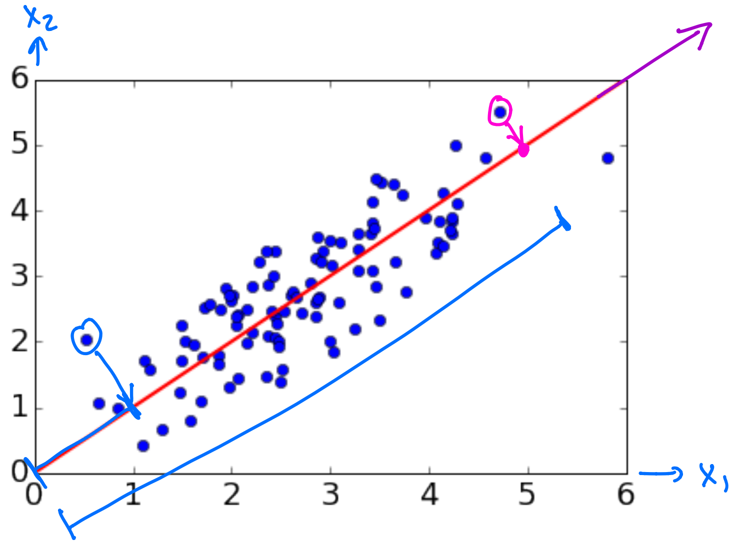
Suppose we wanted just one feature for the following data.



The effect of correlation

Suppose we wanted just one feature for the following data.

There is a lot of redundancy (correlation) between the two features. Projecting onto the shown direction retains most of the information in the data.

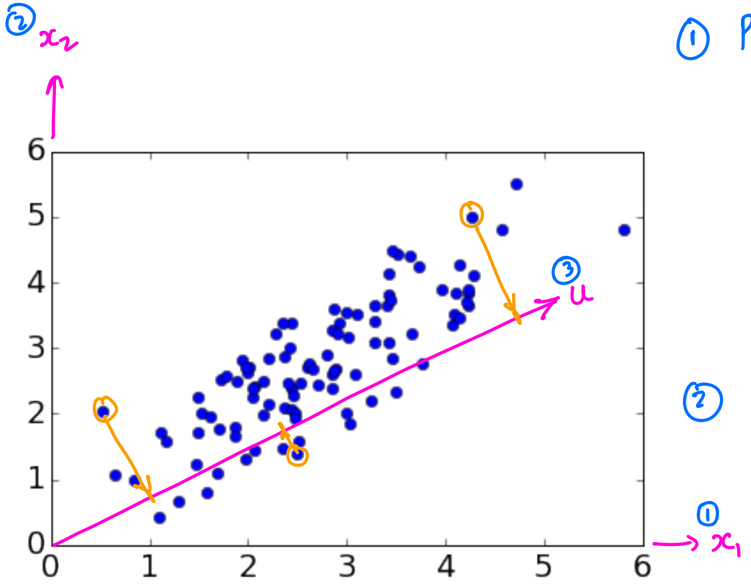


direction of maximum variance

This is the **direction of maximum variance**.

Comparing projections

How to evaluate a projection direction?



① Projection onto x_1 direction

$$u = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Maps $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \longmapsto x_1$

Look at the variance (X_1)

② Projection onto x_2 direction

$$u = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

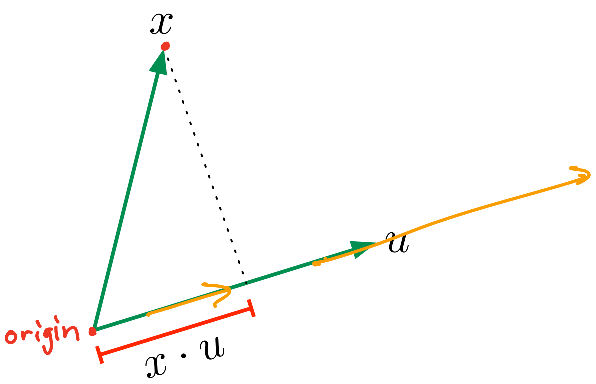
Look at variance of X_2

③ Projection onto arbitrary direction u.

- Project all n data points onto direction u. $\rightarrow$ we get n numbers.
- Look at the variance of those numbers.

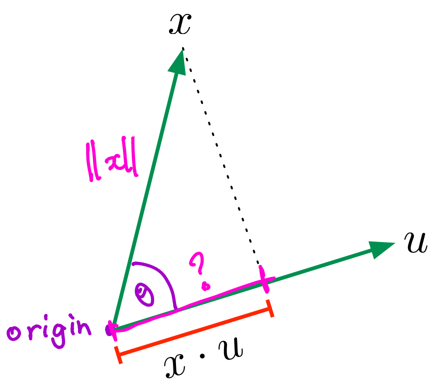
Projection: formally

What is the projection of $x \in \mathbb{R}^d$ in the **direction** $u \in \mathbb{R}^d$?
Assume u is a unit vector (i.e. $\|u\| = 1$).



Projection: formally

What is the projection of $x \in \mathbb{R}^d$ in the **direction** $u \in \mathbb{R}^d$?
Assume u is a unit vector (i.e. $\|u\| = 1$).



Projection is

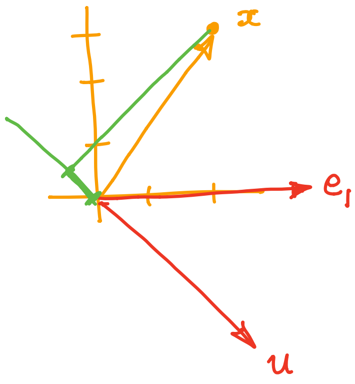
$$x \cdot u = u \cdot x = u^T x = \sum_{i=1}^{d} u_i x_i.$$

$$\cos\theta = \frac{x \cdot u}{\|x\| \, \|u\|} = \frac{\text{adjacent}}{\text{hypotenuse}} = \frac{?}{\|x\|} \Rightarrow ? = \frac{x \cdot u}{\|u\|} = x \cdot u$$

Examples

What is the projection of $x = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$ along the following directions?

1. The x_1-axis?
2. The direction of $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$?



① The projection direction is $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

$$x \cdot e_1 = \begin{pmatrix} 2 \\ 3 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 2$$

② The projection direction is $u = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

$$x \cdot u = \begin{pmatrix} 2 \\ 3 \end{pmatrix} \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = -\frac{1}{\sqrt{2}}$$

The best direction

Suppose we need to map our data $x \in \mathbb{R}^d$ into just **one** dimension:

$$\underbrace{x}_{\mathbb{R}^d} \mapsto \underbrace{u \cdot x}_{\text{number}} \quad \text{for some unit direction } u \in \mathbb{R}^d$$

What is the direction u of maximum variance?

The best direction

Suppose we need to map our data $x \in \mathbb{R}^d$ into just **one** dimension:

$$x \mapsto u \cdot x \qquad \text{for some unit direction } u \in \mathbb{R}^d$$

What is the direction u of maximum variance?

Σ : $d \times d$ matrix

$\Sigma_{ij} = cov(X_i, X_j)$

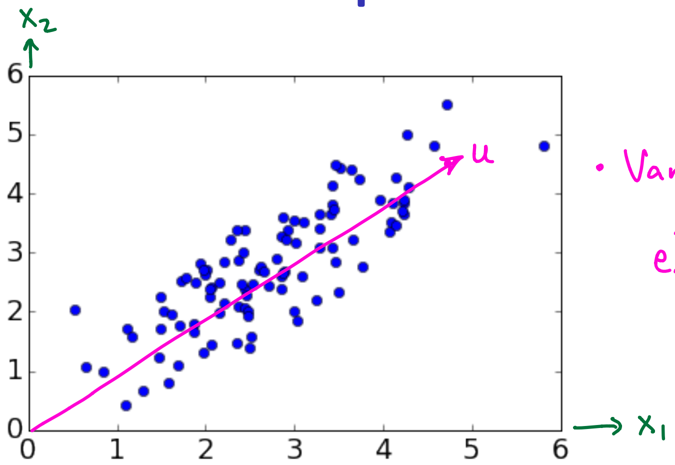
$= \mathbb{E}[X_i X_j] - \mathbb{E}[X_i]\mathbb{E}[X_j]$

Useful fact 1:

- Let Σ be the $d \times d$ covariance matrix of X.
- The variance of X in direction u (the variance of $X \cdot u$) is:

$$u^T \Sigma u.$$

Best direction: example



$$\Sigma = \begin{array}{ccc} & \text{var}(x_1) & \text{cov}(x_1,x_2) \\ & \begin{pmatrix} 1 & 0.85 \\ 0.85 & 1 \end{pmatrix} & \\ & & \text{var}(x_2) \end{array}$$

Here covariance matrix $\Sigma = \begin{pmatrix} 1 & 0.85 \\ 0.85 & 1 \end{pmatrix}$

But is there some other direction with even higher variance?

- Variance in direction $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

$$e_1^T \Sigma e_1 = (1 \quad 0) \begin{pmatrix} 1 & 0.85 \\ 0.85 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1$$

- Variance in direction $e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

$$e_2^T \Sigma e_2 = (0 \quad 1) \begin{pmatrix} 1 & 0.85 \\ 0.85 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 1$$

- Variance in direction $u = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

$$u^T \Sigma u = \frac{1}{2} \cdot (1 \quad 1) \begin{pmatrix} 1 & 0.85 \\ 0.85 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$= 1.85$$

Much higher than in directions e_1, e_2!

The best direction

Suppose we need to map our data $x \in \mathbb{R}^d$ into just **one** dimension:

$$x \mapsto u \cdot x \qquad \text{for some unit direction } u \in \mathbb{R}^d$$

What is the direction u of maximum variance?

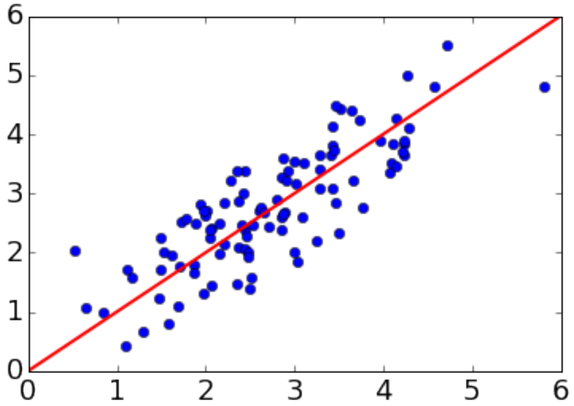
Useful fact 1:

- Let Σ be the $d \times d$ covariance matrix of X.
- The variance of X in direction u is given by $u^T \Sigma u$.

Useful fact 2:

- $u^T \Sigma u$ is maximized by setting u to the first **eigenvector** of Σ.
- The maximum value is the corresponding **eigenvalue**.

Best direction: example



Direction: **first eigenvector** of the 2×2 covariance matrix of the data.

The "first eigenvector" of matrix

$$\Sigma = \begin{bmatrix} 1 & 0.85 \\ 0.85 & 1 \end{bmatrix}$$

is in fact $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

Projection onto this direction: the top **principal component** of the data

And the actual variance in this direction is $u^T \Sigma u$.

Projection onto multiple directions

Projecting $x \in \mathbb{R}^d$ into the k-dimensional subspace defined by vectors $u_1, \ldots, u_k \in \mathbb{R}^d$.

Projection onto multiple directions

Projecting $x \in \mathbb{R}^d$ into the k-dimensional subspace defined by vectors $u_1, \ldots, u_k \in \mathbb{R}^d$.

This is easiest when the u_i's are **orthonormal**:

- They have length one.
- They are at right angles to each other: $u_i \cdot u_j = 0$ when $i \neq j$

$$\underset{d \times k}{U} = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_k \\ | & | & & | \end{bmatrix} \quad \updownarrow d, \quad \overset{k}{\longleftrightarrow}$$

$$U^T U = I_k$$

The projection is a k-dimensional vector:

$$\underbrace{(x \cdot u_1, x \cdot u_2, \ldots, x \cdot u_k)}_{\underbrace{U^T}_{k \times d} \underbrace{x}_{d \times 1} \in \mathbb{R}^k} = \underbrace{\begin{pmatrix} \longleftarrow & u_1 & \longrightarrow \\ \longleftarrow & u_2 & \longrightarrow \\ & \vdots & \\ \longleftarrow & u_k & \longrightarrow \end{pmatrix}}_{\text{call this } U^T} \begin{pmatrix} \uparrow \\ x \\ \downarrow \end{pmatrix}$$

U is the $d \times k$ matrix with columns $u_1, \ldots, u_k$.

Projection onto multiple directions: example

$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \in \mathbb{R}^4$

E.g. project data in $\mathbb{R}^4$ onto the first two coordinates.

projection directions

Take vectors $u_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$, $u_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$ (notice: orthonormal) $\Bigg\}$ $U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$

$$\text{Write} \quad U^T = \underbrace{\begin{pmatrix} \longleftarrow & u_1 & \longrightarrow \\ \longleftarrow & u_2 & \longrightarrow \end{pmatrix}} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}$$

$$U^T x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The projection of $x \in \mathbb{R}^4$ is $U^T x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$

The best k-dimensional projection

$\lambda_1 = u_1^T \Sigma u_1$

$=$ variance in direction u_1

Let Σ be the $d \times d$ covariance matrix of X.
In $O(d^3)$ time, we can compute its **eigendecomposition**, consisting of

- real **eigenvalues** $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$
- corresponding **eigenvectors** $u_1, \ldots, u_d \in \mathbb{R}^d$ that are orthonormal (unit length and at right angles to each other)

Fact: Suppose we want to map data $X \in \mathbb{R}^d$ to just k dimensions, while capturing as much of the variance of X as possible. The best choice of projection is:

$$x \mapsto (u_1 \cdot x, u_2 \cdot x, \ldots, u_k \cdot x),$$

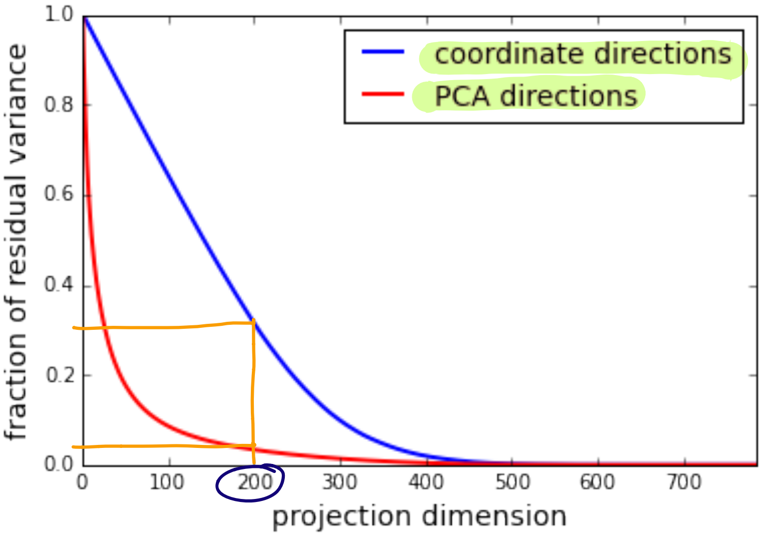
where u_i are the eigenvectors described above.

Top k directions $\equiv$ top k eigenvectors of the covariance matrix

This projection is called **principal component analysis** (PCA).

Example: MNIST

Contrast coordinate projections with PCA:



E.g. reduce dimension from 784 to 200.

Coordinate directions: lose 35% of overall variance

PCA: lose < 5% of overall variance

Applying PCA to MNIST: examples

Σ = covariance matrix of 60,000 pts

784×784

Reconstruct this original image from its PCA projection to k dimensions.

Top k eigenvectors of Σ: $u_1, u_2, \ldots, u_k \in \mathbb{R}^{784}$

$$U = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_k \\ | & | & & | \end{bmatrix}$$

x 784

$\Big\downarrow$ project

$U^T x$ 50

$\Big\downarrow$ reconstruct

$UU^T x$ 784

$k = 200$

$k = 150$

$k = 100$

$k = 50$

Projection:

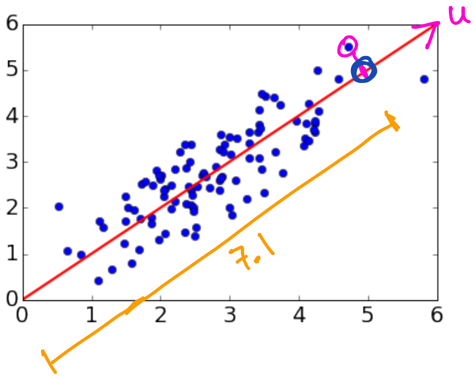
$$\underbrace{x}_{784} \longmapsto \underbrace{U^T x}_{50}$$

e.g. $k=50$:

$$\underset{784 \times 50}{U}: \begin{bmatrix} | & & | \\ u_1 & \cdots & u_{50} \\ | & & | \end{bmatrix}$$

How do we get these **reconstructions**?

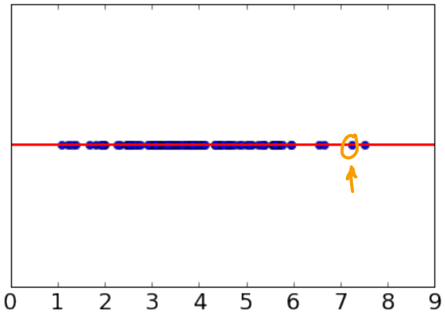
Reconstruction from a 1-d projection



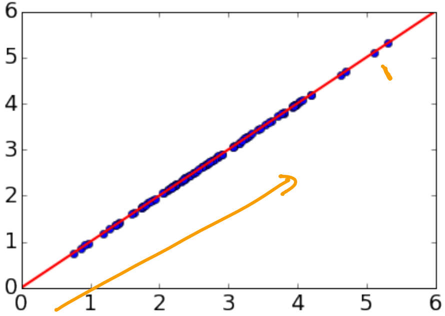
$\underbrace{x}_{\mathbb{R}^2} \longmapsto \underbrace{x \cdot u}_{\text{single number}}$

reconstruction is 7.1 u

Projection onto $\mathbb{R}$:



Reconstruction in $\mathbb{R}^2$:



Reconstruction from multiple projections

Projecting into the k-dimensional subspace defined by **orthonormal** $u_1, \ldots, u_k \in \mathbb{R}^d$.

The projection of x is a k-dimensional vector:

$$\underset{\mathbb{R}^d}{x} \xrightarrow[\text{project}]{} \underbrace{(\overset{2.3}{x \cdot u_1}, \overset{-0.5}{x \cdot u_2}, \ldots, \overset{8.6}{x \cdot u_k})}_{\mathbb{R}^k} = \underbrace{\begin{pmatrix} \longleftarrow & u_1 & \longrightarrow \\ \longleftarrow & u_2 & \longrightarrow \\ & \vdots & \\ \longleftarrow & u_k & \longrightarrow \end{pmatrix}}_{\text{call this } U^T} \begin{pmatrix} \uparrow \\ x \\ \downarrow \end{pmatrix}$$

The reconstruction from this projection is:

$$\underbrace{(x \cdot u_1)u_1}_{\substack{2.3 \\ \text{units in} \\ \text{direction } u_1}} + \underbrace{(x \cdot u_2)u_2}_{\substack{-0.5 \\ \text{units in} \\ \text{direction } u_2}} + \cdots + \underbrace{(x \cdot u_k)u_k}_{\substack{8.6 \\ \text{units in} \\ \text{direction } u_k}} = UU^T x.$$

MNIST: image reconstruction

$$\underset{\mathbb{R}^{784}}{x} \xmapsto{\text{project}} \underset{\mathbb{R}^{k}}{U^T x} \xmapsto{\text{reconstruct}} \underset{\mathbb{R}^{784}}{UU^T x}$$

Reconstruct this original image x from its PCA projection to k dimensions.

Projection:

$x \mapsto U^T x$

Reconstruction

$z \mapsto Uz$

$k = 200$

$k = 150$

$k = 100$

$k = 50$

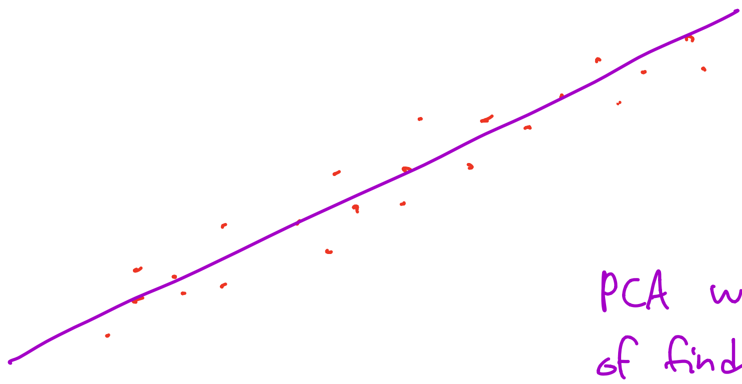
Worksheet 11 #1,3,4

Lab 3 # 1,2,3

Reconstruction $UU^T x$, where U's columns are top k eigenvectors of Σ.

PCA for nonlinear relationships

① Regular PCA is really good at discovering <u>linear</u> subspaces in which the data happen to lie.



e.g. In this case, 2-d data lies close to a 1-d subspace.

PCA will do a good job of finding this subspace.

② But what if the data looks like this:

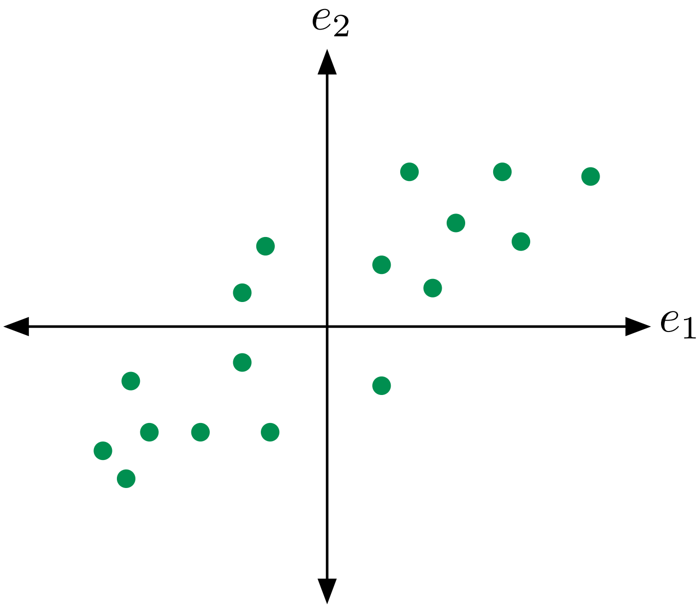
This is also, arguably, 1-dimensional, but is not a <u>linear</u> subspace.

PCA will fail.

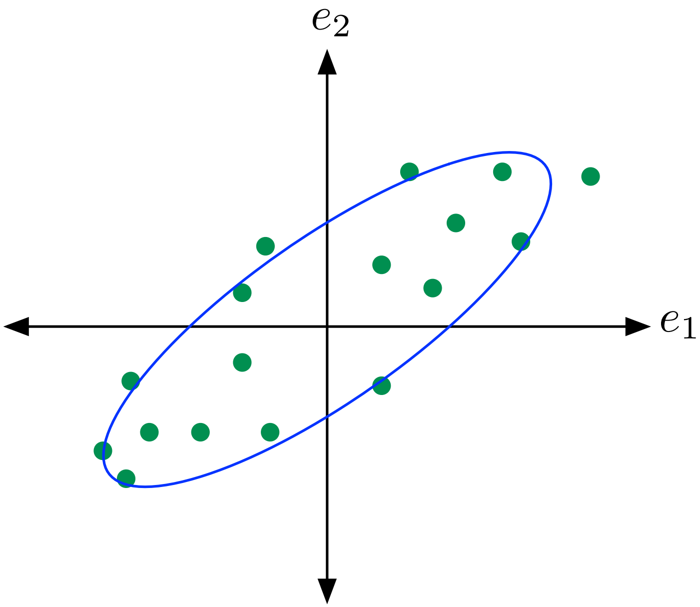
"Manifold"

→ There are separate methods for such data "manifold learning"

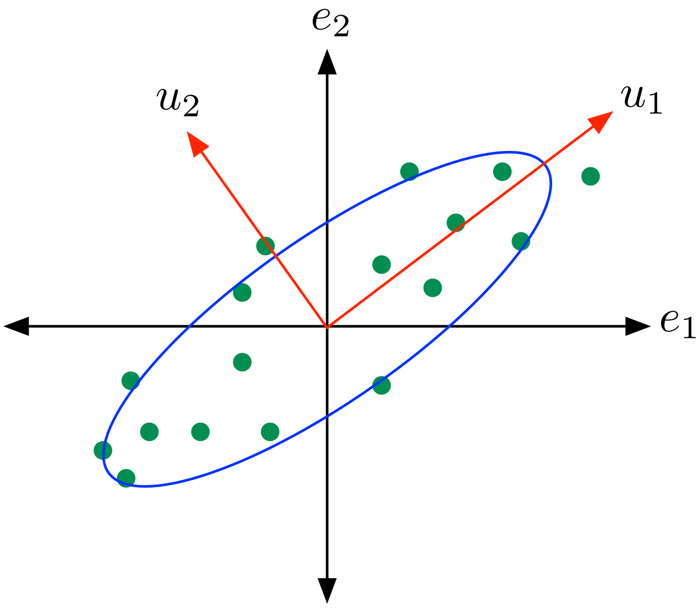
Linear algebra: eigenvalues and eigenvectors



Linear algebra: eigenvalues and eigenvectors



Linear algebra: eigenvalues and eigenvectors



Σ = cov. matrix of data

$\Sigma e_1 \neq \lambda e_1$

$\Sigma e_2 \neq \lambda e_2$

$$\begin{cases} \Sigma u_1 = \lambda_1 u_1 \\ \Sigma u_2 = \lambda_2 u_2 \end{cases}$$

The linear function defined by a matrix

- Any matrix M defines a linear function, $x \mapsto Mx$.
  If M is a $d \times d$ matrix, this maps $\mathbb{R}^d$ to $\mathbb{R}^d$.

- This function is easy to understand when M is **diagonal**:

$$\underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 10 \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}}_{x} = \underbrace{\begin{pmatrix} 2x_1 \\ -x_2 \\ 10x_3 \end{pmatrix}}_{Mx}$$

  In this case, M simply scales each coordinate separately.

- General symmetric matrices also just scale coordinates separately... but in a **different coordinate system!**

Eigenvector and eigenvalue: definition

Let M be a $d \times d$ matrix. We say $u \in \mathbb{R}^d$ is an **eigenvector** of M if

$$Mu = \lambda u$$

for some scaling constant λ. This λ is the **eigenvalue** associated with u.

Key point: M **maps eigenvector** u **onto the same direction**.

Question: What are the eigenvectors and eigenvalues of:

$$M\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2x_1 \\ -x_2 \\ 10x_3 \end{pmatrix}$$

$$M = \begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 10 \end{pmatrix} ?$$

$$Me_1 = \begin{pmatrix} 2 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 10 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix} = 2e_1 \quad ; \quad \text{so} \quad \lambda_1 = 2$$

$$Me_2 = -e_2 \quad ; \quad \text{so} \quad \lambda_2 = -1$$

$$Me_3 = 10\, e_3 \quad ; \quad \text{so} \quad \lambda_3 = 10$$

The eigenvectors of this matrix M are e_1, e_2, e_3, with eigenvalues $\lambda_1 = 2$, $\lambda_2 = -1$, $\lambda_3 = 10$.

Eigenvectors of a real symmetric matrix

Fact: Let M be any real symmetric $d \times d$ matrix. Then M has

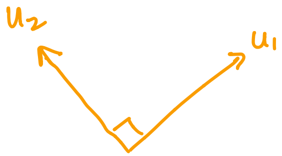
- d eigenvalues $\lambda_1, \ldots, \lambda_d$
- corresponding eigenvectors $u_1, \ldots, u_d \in \mathbb{R}^d$ that are orthonormal

Can think of $u_1, \ldots, u_d$ as the axes of the natural coordinate system for M.

Example

$$M = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} \text{ has eigenvectors } \underline{u_1} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \ \underline{u_2} = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

1. Are these orthonormal?
2. What are the corresponding eigenvalues?



① Unit length: $\|u_1\| = \|u_2\| = 1$

Dot product $u_1 \cdot u_2 = 0$

② $Mu_1 = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} \cdot \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}} = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ -1 \end{pmatrix} \Rightarrow Mu_1 = -u_1 \Rightarrow \lambda_1 = -1$

$Mu_2 = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} -3 \\ 3 \end{pmatrix} \Rightarrow Mu_2 = 3u_2 \Rightarrow \lambda_2 = 3$

M has eigenvectors u_1 and u_2 with eigenvalues λ_1 and λ_2.

Spectral decomposition

Fact: Let M be any real symmetric $d \times d$ matrix. Then M has orthonormal eigenvectors $u_1, \ldots, u_d \in \mathbb{R}^d$ and corresponding eigenvalues $\lambda_1, \ldots, \lambda_d$.

Spectral decomposition: Another way to write M:

We can write M entirely in terms of its eigenvectors and eigenvalues!

$$M = \underbrace{\begin{pmatrix} \uparrow & \uparrow & & \uparrow \\ u_1 & u_2 & \cdots & u_d \\ \downarrow & \downarrow & & \downarrow \end{pmatrix}}_{U\text{: columns are eigenvectors}} \underbrace{\begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_d \end{pmatrix}}_{\Lambda\text{: eigenvalues on diagonal}} \underbrace{\begin{pmatrix} \longleftarrow & u_1 & \longrightarrow \\ \longleftarrow & u_2 & \longrightarrow \\ & \vdots & \\ \longleftarrow & u_d & \longrightarrow \end{pmatrix}}_{U^T} \begin{bmatrix} | \\ x \\ | \end{bmatrix}$$

Thus $Mx = U \Lambda U^T x$:

- U^T rewrites x in the $\{u_i\}$ coordinate system
- Λ is a simple coordinate scaling in that basis
- U sends the scaled vector back into the usual coordinate basis

Apply spectral decomposition to the matrix we saw earlier:

Claim:

$$M = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} \qquad M = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} — u_1 — \\ — u_2 — \end{bmatrix}$$

- Eigenvectors $u_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $u_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 1 \end{pmatrix}$
- Eigenvalues $\lambda_1 = -1,\ \lambda_2 = 3$.

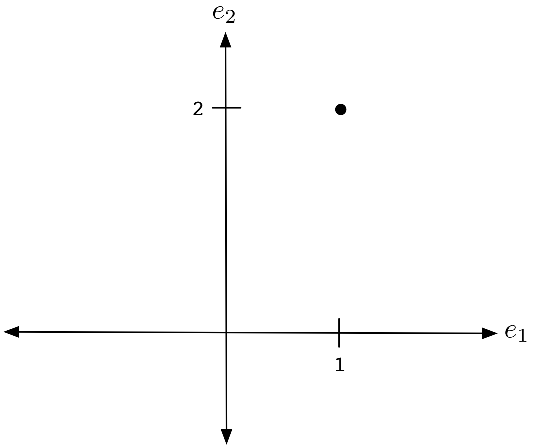
$$\begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} — u_1 — \\ — u_2 — \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} -1 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}}$$

$$= \frac{1}{2} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & -1 \\ -3 & 3 \end{bmatrix}$$

$$= \frac{1}{2} \begin{bmatrix} 2 & -4 \\ -4 & 2 \end{bmatrix} = \begin{bmatrix} 1 & -2 \\ -2 & 1 \end{bmatrix} = M$$

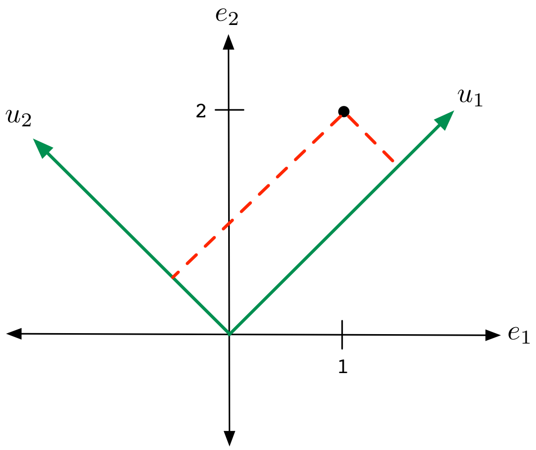
$$M = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} = \underbrace{\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}}_{U} \underbrace{\begin{pmatrix} -1 & 0 \\ 0 & 3 \end{pmatrix}}_{\Lambda} \underbrace{\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}}_{U^T}$$

$$M \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

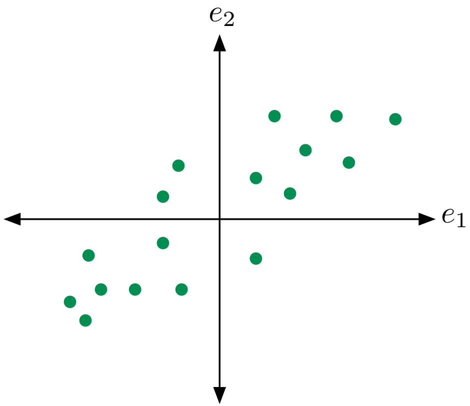


$$M = \begin{pmatrix} 1 & -2 \\ -2 & 1 \end{pmatrix} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}}_{U} \underbrace{\begin{pmatrix} -1 & 0 \\ 0 & 3 \end{pmatrix}}_{\Lambda} \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}}_{U^T}$$

$$M \begin{pmatrix} 1 \\ 2 \end{pmatrix} = U \Lambda U^T \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

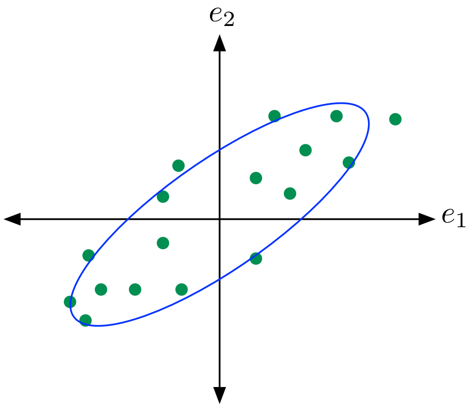


Principal component analysis revisited



Data vectors $X \in \mathbb{R}^d$

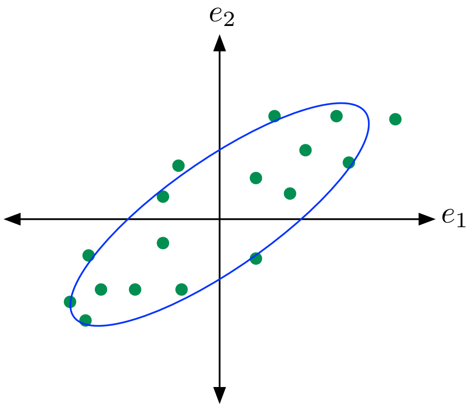
Principal component analysis revisited



Data vectors $X \in \mathbb{R}^d$

- $d \times d$ covariance matrix Σ is symmetric.

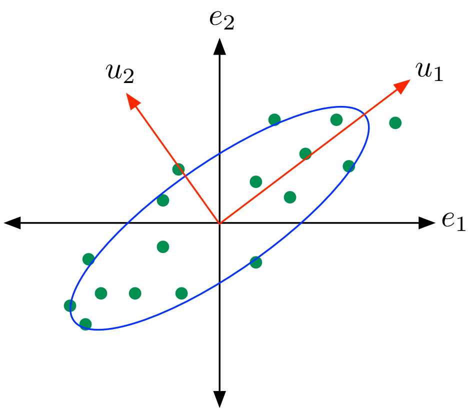
Principal component analysis revisited



Data vectors $X \in \mathbb{R}^d$

- $d \times d$ covariance matrix Σ is symmetric.
- Eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$
  Eigenvectors $u_1, \ldots, u_d$.

Principal component analysis revisited



Data vectors $X \in \mathbb{R}^d$

- $d \times d$ covariance matrix Σ is symmetric.
- Eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$
  Eigenvectors $u_1, \ldots, u_d$.
- $u_1, \ldots, u_d$: another basis for data.
- Variance of X in direction u_i is λ_i.
- Projection to k dimensions:
  $x \mapsto (x \cdot u_1, \ldots, x \cdot u_k)$.

What is the covariance of the projected data?

$$k \; \underbrace{\begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_k \end{bmatrix}}_{k}$$

uncorrelated

Case study: Quantifying personality

What are the dimensions along which personalities differ?

- *Lexical hypothesis*: most important personality characteristics have become encoded in natural language.
- Allport and Odbert (1936): identified 4500 words describing personality traits.
- Group these words into (approximate) synonyms, by manual clustering.
  E.g. Norman (1967):

Spirit	Jolly, merry, witty, lively, peppy
Talkativeness	Talkative, articulate, verbose, gossipy
Sociability	Companionable, social, outgoing
Spontaneity	Impulsive, carefree, playful, zany
Boisterousness	Mischievous, rowdy, loud, prankish
Adventure	Brave, venturous, fearless, reckless
Energy	Active, assertive, dominant, energetic
Conceit	Boastful, conceited, egotistical
Vanity	Affected, vain, chic, dapper, jaunty
Indiscretion	Nosey, snoopy, indiscreet, meddlesome
Sensuality	Sexy, passionate, sensual, flirtatious

500

- Data collection: subjects whether these words describe them.

Personality assessment: the data

Matrix of data (1 = strongly disagree, 5 = strongly agree)

500 words

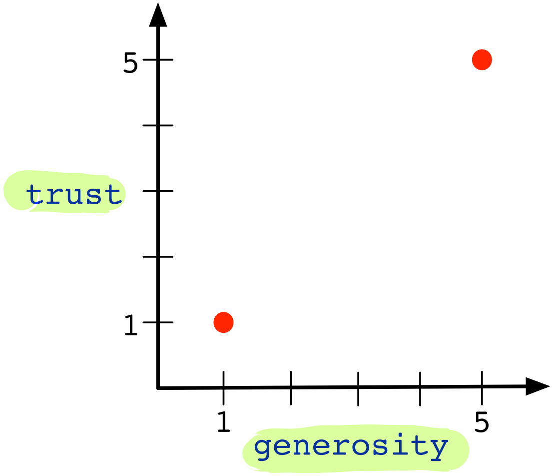
	shy	merry	tense	boastful	forgiving	quiet
Person 1	4	1	1	2	5	5
Person 2	1	4	4	5	2	1
Person 3	2	4	5	4	2	2
		⋮				

How to extract important directions?

- Treat each column as a data point, find tight clusters
- Treat each row as a data point, apply PCA
- Or factor analysis, independent component analysis, etc.

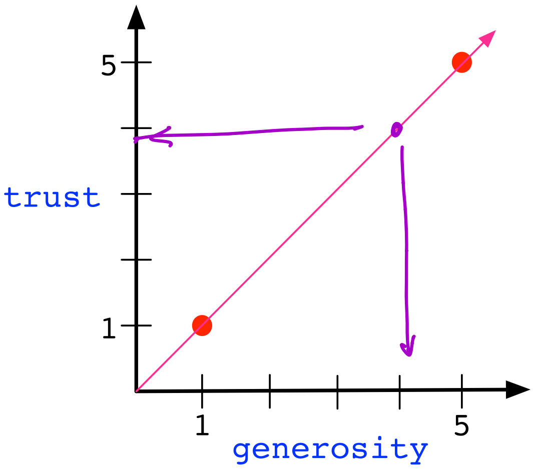
What would PCA accomplish?

E.g.: Suppose two traits (generosity, trust) are so highly correlated that each person either answers “1” to both or “5” to both.



What would PCA accomplish?

E.g.: Suppose two traits (generosity, trust) are so highly correlated that each person either answers “1” to both or “5” to both.



A single PCA dimension would entirely account for both traits.

Personality assessment: the data

Matrix of data (1 = strongly disagree, 5 = strongly agree)

$\longleftarrow$ 500 $\longrightarrow$

	shy	merry	tense	boastful	forgiving	quiet
Person 1	4	1	1	2	5	5
Person 2	1	4	4	5	2	1
Person 3	2	4	5	4	2	2
		⋮				

1st PCA direction u_1	2nd PCA direction u_2	5th PCA direction u_5
2.3	3.8	-0.1
-0.6	10.2	3.9
4.5	-0.3	8.2
F_1		

F_1: first "synthetic" personality trait

Methodology: apply PCA to the rows of this matrix.

The "Big Five" taxonomy

Extraversion

−: quiet (-.83), reserved (-.80), shy (-.75), silent (-.71)

+: talkative (.85), assertive (.83), active (.82), energetic (.82)

Agreeableness

−: fault-finding (-.52), cold (-.48), unfriendly (-.45), quarrelsome (-.45)

+: sympathetic (.87), kind (.85), appreciative (.85), affectionate (.84)

Conscientousness

−: careless (-.58), disorderly (-.53), frivolous (-.50), irresponsible (-.49)

+: organized (.80), thorough (.80), efficient (.78), responsible (.73)

Neuroticism

−: stable (-.39), calm (-.35), contented (-.21)

+: tense (.73), anxious (.72), nervous (.72), moody (.71)

Openness

−: commonplace (-.74), narrow (-.73), simple (-.67), shallow (-.55)

+: imaginative (.76), intelligent (.72), original (.73), insightful (.68)