Informative projections

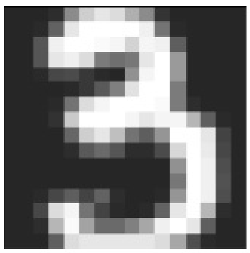
A: Linear projection

Compression via 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.

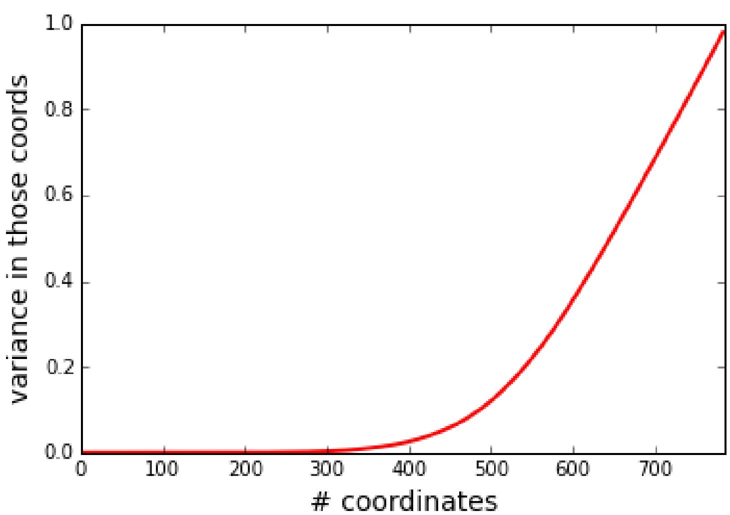
Example: are all the pixels in an image equally informative?



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

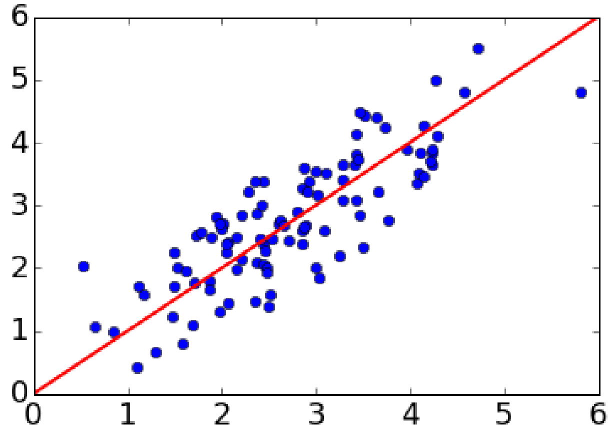
Eliminating low variance coordinates

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



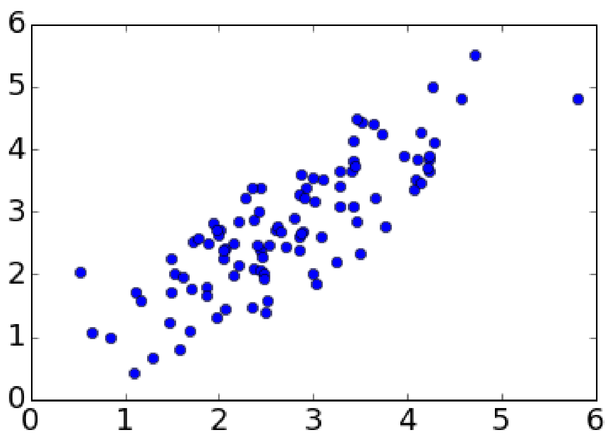
The effect of correlation

Suppose we wanted just one feature for the following data.



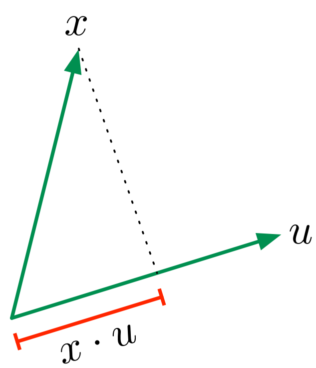
This is the **direction of maximum variance**.

Comparing projections



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

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

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 $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$?

B: The best direction

The best direction

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

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

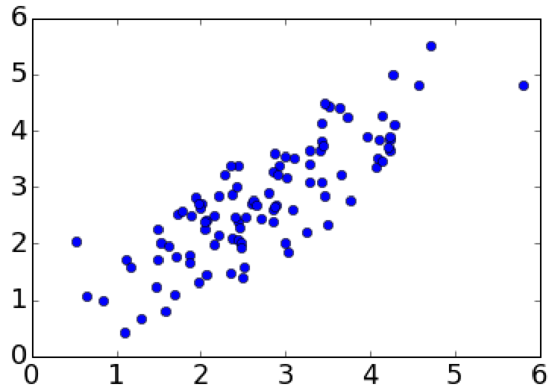
What is the direction u of maximum variance?

Useful fact 1:

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

$$u^T \Sigma u.$$

Best direction: example



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

The best direction

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

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

What is the direction u of maximum variance?

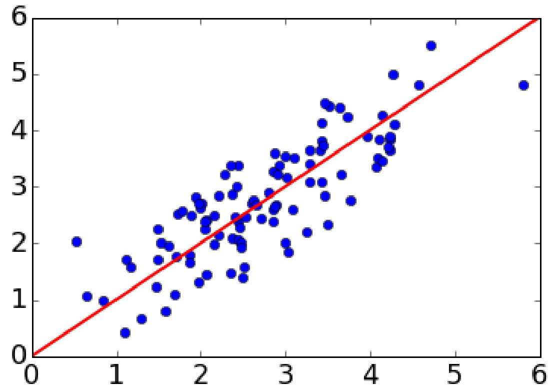
Useful fact 1:

- 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.

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

C: Principal component analysis

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 $i \neq j$

The projection is a k-dimensional vector:

$$(x \cdot u_1, x \cdot u_2, \ldots, x \cdot u_k) \;=\; \underbrace{\begin{pmatrix} \longleftarrow & u_1 & \longrightarrow \\ \longleftarrow & u_2 & \longrightarrow \\ & \vdots & \\ \longleftarrow & u_k & \longrightarrow \end{pmatrix}}_{\text{call this } U^T} \begin{pmatrix} \uparrow \\ x \\ \downarrow \end{pmatrix}$$

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

The best k-dimensional projection

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:

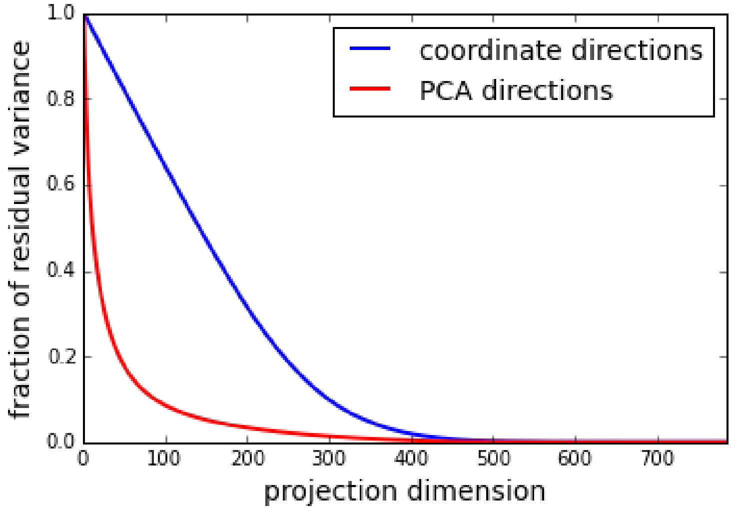
$$x \mapsto (u_1 \cdot x, u_2 \cdot x, \ldots, u_k \cdot x),$$

where u_i are the eigenvectors described above.

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

Example: MNIST

Contrast coordinate projections with PCA:



Applying PCA to MNIST: examples

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

$k = 200$

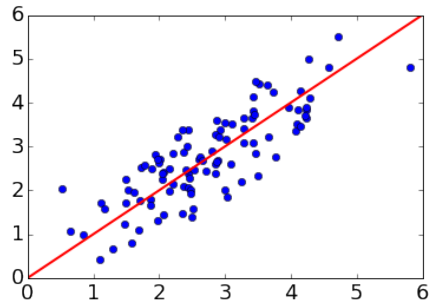
$k = 150$

$k = 100$

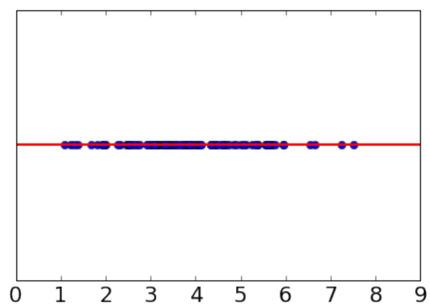
$k = 50$

How do we get these **reconstructions**?

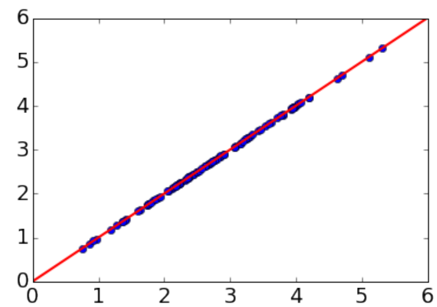
Reconstruction from a 1-d projection



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:

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

The reconstruction from this projection is:

$$(x \cdot u_1)u_1 + (x \cdot u_2)u_2 + \cdots + (x \cdot u_k)u_k = UU^T x.$$

MNIST: image reconstruction

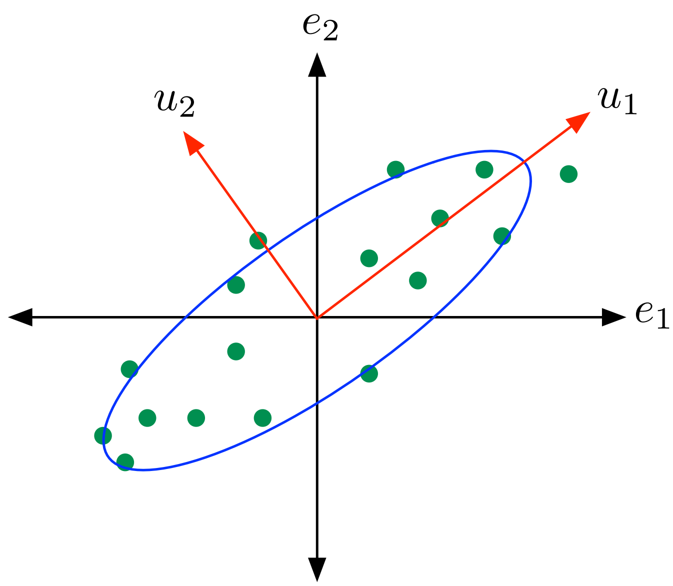
Reconstruct this original image x from its PCA projection to k dimensions.

$k = 200$ $k = 150$ $k = 100$ $k = 50$

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

D: Eigenvalues and eigenvectors

Linear algebra: review of eigendecomposition



Eigenvector and eigenvalue: definition

Let M be any $d \times d$ matrix.

- M defines a linear function, $x \mapsto Mx$. This maps $\mathbb{R}^d$ to $\mathbb{R}^d$.
- 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:

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

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

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

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

Diagonal matrices

What is the "natural coordinate system" for a diagonal matrix?

E: Spectral decomposition

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:

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

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

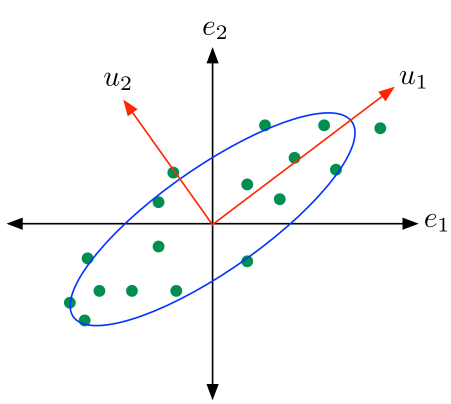
- U^T rewrites x in the $\{u_i\}$ coordinate system
- Λ is a simple coordinate scaling in that basis
- U sends the scaled vector back into the usual coordinate basis

Apply spectral decomposition to the matrix we saw earlier:

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

- 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 $\lambda_1 = -1, \ \lambda_2 = 3$.

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?

F: Case study

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

- Data collection: subjects whether these words describe them.

Personality assessment: the data

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

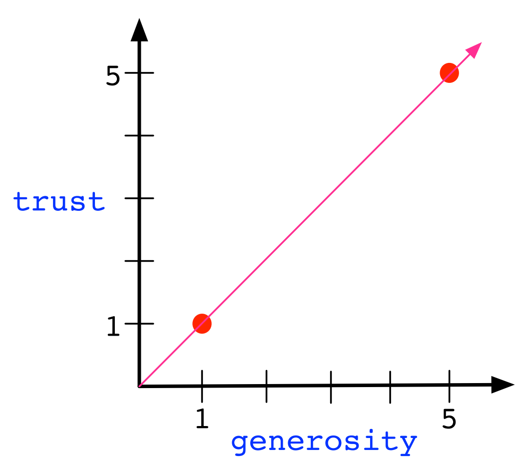
	shy	merry	tense	boastful	forgiving	quiet
Person 1	4	1	1	2	5	5
Person 2	1	4	4	5	2	1
Person 3	2	4	5	4	2	2
		⋮				

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.



A single PCA dimension would entirely account for both traits.

Personality assessment: the data

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

	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
		⋮				

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)

G: Optimality properties of PCA

Best approximating linear subspace

Given: n points $x_1, \ldots, x_n \in \mathbb{R}^d$ and $k < d$.

- Choose a k-dimensional linear subspace "close to the data".
- Approximate each x_i by its projection $\tilde{x}_i$ onto this subspace.

Goal: minimize the distortion

$$\sum_{i=1}^{n} \|x_i - \tilde{x}_i\|^2$$

Best approximating linear subspace: solution

Linear versus affine subspaces

Best approximating affine subspace

Pick any n points $x_1, \ldots, x_n \in \mathbb{R}^d$ and any $k < d$.

- Let μ be the empirical average of the $\{x_i\}$ and Σ the empirical covariance matrix.
- Let $u_1, \ldots, u_k$ be the top k eigenvectors of Σ. Make these the columns of a $d \times k$ matrix U.

Projection onto the best approximating affine subspace:

H: Random projection

Johnson-Lindenstrauss Lemma

Summary: **Any set of n points is approximately embeddable in $O(\log n)$ dimensions**.

- Pick any $0 < \epsilon \leq 1/2$ and set $k = (4/\epsilon^2) \log n$.
- Any n points in $\mathbb{R}^d$ can be embedded into $\mathbb{R}^k$, such that each of the interpoint (Euclidean) distances is distorted by at most a multiplicative factor of $1 \pm \epsilon$.
- Moreover, a projection into a random k-dimensional subspace will achieve this with probability close to 1.

How to project into a random subspace?