Multi-dimensional Signal Analysis

Lecture 2E

Principal Component Analysis

Subspace representation

Note!

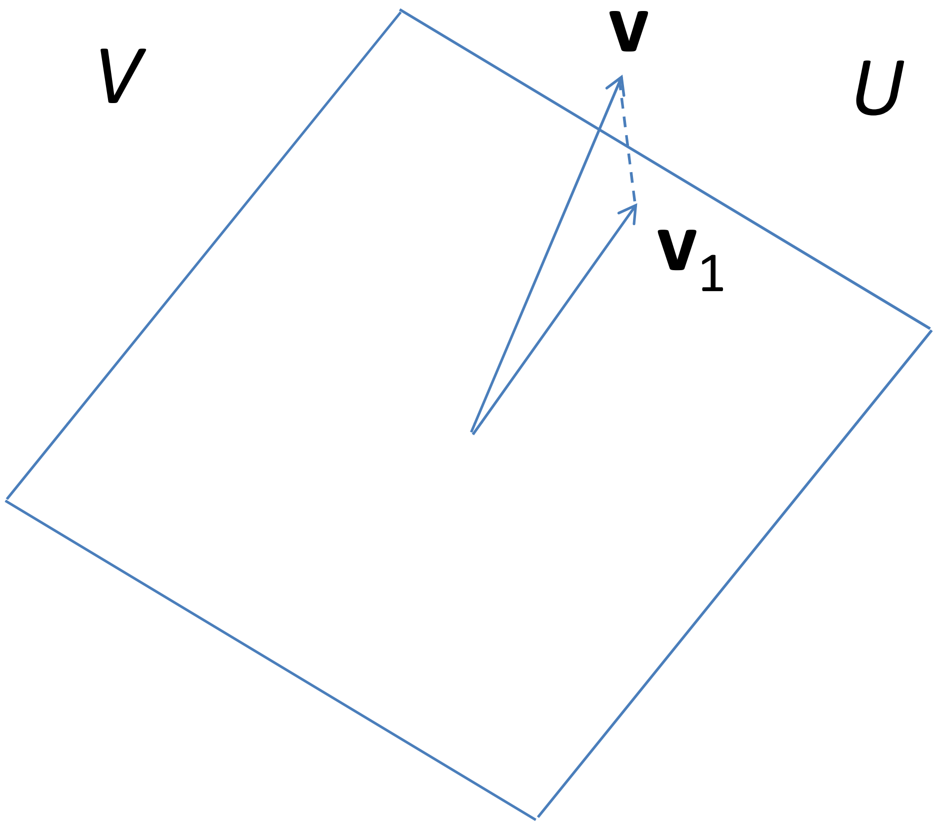
Given

- a vector space V of dimension N
- a scalar product defined by $\mathbf{G}_0$
- a subspace U of dimension $M < N$
- An $N \times M$ basis matrix $\mathbf{B}$ of the subspace U
- a vector $\mathbf{v} \in V$

we can determine $\mathbf{v}_1 \in U$ that is closest to $\mathbf{v}$

- $\mathbf{v}_1 \in U$ is a subspace representation of $\mathbf{v} \in V$
- $\mathbf{v}_1$ is independent of $\mathbf{B}$, it only depends on U

Subspace representation



Subspace representation

More precisely:

- the coordinates of $\mathbf{v}_1$ relative basis $\mathbf{B}$ is given by

$$\boxed{\mathbf{c} = \mathbf{G}^{-1}\,\tilde{\mathbf{c}}}$$

Where $\tilde{\mathbf{c}} = \mathbf{B}^T\mathbf{G}_0\mathbf{v}$ and $\mathbf{G} = \mathbf{B}^T\mathbf{G}_0\mathbf{B}$

Stochastic signals

- In the previous applications *normalised convolution* and *filter optimisation* the basis was fixed
  - This means that U is fixed
- An alternative approach is to allow the signal vector $\mathbf{v}$ to be a *stochastic variable* and to determine an M-dimensional subspace U such that $\mathbf{v}_1$ is as close as possible to $\mathbf{v}$ <u>in average</u>

Initial problem formulation

- We want to minimise

$$\boxed{\epsilon = E \, \| \mathbf{v} - \mathbf{v}_1 \|^2}$$

E means here to take the expectation value or mean

where the expectation value is taken over all observations of the signal **v**

- ϵ is minimised over all *M*-dimensional subspaces *U*

Problem formulation

To simplify things, we assume that

- V is real = $\mathbb{R}^N$
- $\mathbf{G}_0 = \mathbf{I}$
- $\mathbf{B}$ is an ON-basis of subspace U:

Leads to:

$$\boxed{\langle \mathbf{u} \mid \mathbf{v} \rangle = \mathbf{u}^T\mathbf{v}} \qquad \boxed{\tilde{\mathrm{B}} = \mathrm{B}} \qquad \boxed{\mathbf{B}^T\mathbf{B} = \mathbf{I}}$$

Problem formulation

- This simplification also means that:

$$\mathbf{v}_1 = \mathbf{B}\,\mathbf{c} = \mathbf{B}\,\mathbf{B}^T\mathbf{v}$$

- We want to determine an *M*-dim subspace *U* (represented by ON-basis **B**) such that we minimise

$$\epsilon = E\,\|\,\mathbf{v} - \mathbf{B}\,\mathbf{B}^T\mathbf{v}\,\|^2$$

Solving the problem

- We assume that **v** has known statistical properties
- We want to determine **B** that minimises ϵ
- This is the same as maximising

$$\boxed{\epsilon_1 = E\,[\mathbf{v}^T\mathbf{B}\,\mathbf{B}^T\mathbf{v}]}$$

(why?)

for ON-basis **B**

- This is an optimisation problem with a set of constraints ($\mathbf{B}^T\mathbf{B} = \mathbf{I}$)

Practical solution

- As soon as M,
  = the dimension of U,
  = the number of columns in $\mathbf{B}$,
  is determined, we optimise ϵ_1 over the $N \times M$ elements in $\mathbf{B}$ with the constraints $\mathbf{B}^T\mathbf{B} = \mathbf{I}$
- Can be solved by means of standard techniques for constrained optimisation
  – Lagrange's method

A simple example

- Simple example: $M = 1$
  - $\mathbf{B}$ has only one single column $\mathbf{b}_1$
- We want to maximise

$$\epsilon_1 = E\,[\mathbf{v}^T\mathbf{b}_1\mathbf{b}_1{}^T\mathbf{v}] = E\,[\mathbf{b}_1{}^T\mathbf{v}\,\mathbf{v}^T\mathbf{b}_1] = \mathbf{b}_1{}^T\,E\,[\mathbf{v}\,\mathbf{v}^T]\,\mathbf{b}_1$$

with constraint $c = \mathbf{b}_1{}^T\,\mathbf{b}_1 = 1$

A simple example

- Use Lagrange's method:

$$\nabla \epsilon_1 = \lambda \, \nabla c$$

where the gradients are with respect to the elements in $\mathbf{b}_1$

- Leads to

$$E\,[\,\mathbf{v}\,\mathbf{v}^T\,]\,\mathbf{b}_1 = \lambda\,\mathbf{b}_1$$

A simple example

Consequently:

$\mathbf{b}_1$ is an eigenvector corresponding to eigenvalue λ of E [$\mathbf{v}\,\mathbf{v}^T$]

- Remember: $\|\mathbf{b}_1\| = 1$, since **B** is ON

The correlation matrix

- It is clear that the matrix

$$\mathbf{C} = E[\,\mathbf{v}\,\mathbf{v}^T\,]$$

Approximation of **C** from *P* samples:

$$\mathbf{C} \approx \frac{1}{P}\sum_{k=1}^{P} \mathbf{v}_k \mathbf{v}_k^T$$

  is an important thing to know in order to solve the problem
- **C** is called the *correlation matrix* of the signal
- **C** is symmetric and $N \times N$
- **C** is positive definite (why?)

A simple example

- With this choice of $\mathbf{b}_1$, ϵ_1 becomes

$$\epsilon_1 = \mathbf{b}_1^T \mathbf{C}\, \mathbf{b}_1 = \lambda\, \mathbf{b}_1^T \mathbf{b}_1 = \lambda$$

- Since we want to <u>maximise</u> ϵ_1 we should choose λ as large as possible

$\Rightarrow$ $\mathbf{b}_1$ is a normalised eigenvector corresponding to the largest eigenvalue of **C**

A simple example

- From

$$\epsilon_1 = \lambda_1 = \text{largest eigenvalue of } \mathbf{C}$$

follows that

$$\epsilon = \text{sum of all eigenvalues of } \mathbf{C} \text{ except } \lambda_1$$

(why?)

A slightly more complicated example

- Let U be 2-dimensional:
  $\mathbf{B}$ has two columns $\mathbf{b}_1$ and $\mathbf{b}_2$

- We want to maximise
  $\epsilon_1 = E\,[\mathbf{v}^T\mathbf{B}\,\mathbf{B}^T\mathbf{v}]$
  with the constraints
  $c_1 = \mathbf{b}_1^T\mathbf{b}_1 = 1, \quad c_2 = \mathbf{b}_1^T\mathbf{b}_2 = 0, \qquad c_3 = \mathbf{b}_2^T\mathbf{b}_2 = 1$

A slightly more complicated example

- In the general case, when $M > 1$, the subspace ON-basis matrix **B** is not unique
- **B**' = **B Q** with **Q** $\in O(M)$ is also a solution (why?)
- We need additional constraints on **B** in order to make **B** unique
- We choose: $\mathbf{B}^T\mathbf{C}\,\mathbf{B}$ is diagonal
  - Subspace basis vectors are *uncorrelated*

A slightly more complicated example

This additional constraint leads to

$$\boxed{\begin{array}{c} \mathbf{C}\,\mathbf{b}_1 = \lambda_1\,\mathbf{b}_1 \\ \mathbf{C}\,\mathbf{b}_2 = \lambda_2\,\mathbf{b}_2 \end{array}}$$

$\Rightarrow$ Both $\mathbf{b}_1$ and $\mathbf{b}_2$ must the normalised and mutually orthogonal eigenvectors of **C**, with eigenvalues λ_1 and λ_2

A slightly more complicated example

- We want to maximise (why?)

$$\epsilon_1 = E\,[\mathbf{v}^T\mathbf{B}\,\mathbf{B}^T\mathbf{v}] = \mathbf{b}_1^T\,\mathbf{C}\,\mathbf{b}_1 + \mathbf{b}_2^T\,\mathbf{C}\,\mathbf{b}_2 = \lambda_1 + \lambda_2$$

  and therefore we should choose $\mathbf{b}_1$ and $\mathbf{b}_2$ as normalised eigenvectors corresponding to the two largest eigenvalues of $\mathbf{C}$
  - Remember multiplicity of eigenvalues
- Since $\mathbf{C}$ is symmetric, $\mathbf{b}_1$ and $\mathbf{b}_2$ can always be chosen as orthogonal!

Generalisation

- Based on these examples we present a general result:

  We want to determine an M-dimensional subspace U, described by a ON-basis matrix $\mathbf{B}$.

1. Form the correlation matrix $\mathbf{C}$
2. Compute eigenvalues and eigenvectors of $\mathbf{C}$
3. The basis $\mathbf{B}$ consists of M eigenvectors corresponding to the M largest eigenvalues of $\mathbf{C}$

Generalisation

- This gives

$$\boxed{\epsilon_1 = [\mathbf{v}^T\mathbf{B}\ \mathbf{B}^T\mathbf{v}] = \lambda_1 + \ldots + \lambda_M}$$

and

$$\boxed{\epsilon = \lambda_{M+1} + \ldots + \lambda_N}$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$ are the eigenvalues of **C**

Principal components

- The eigenvectors of **C** are in this context referred to as *principal components*
- The *magnitude* of a principal component is given by the corresponding eigenvalue
- The *M*-dimensional subspace *U* is spanned by the *M* largest principal components of **C**
- To determine the basis **B** in this way is called *principal component analysis* or PCA

Analysis and reconstruction

- In PCA we *analyse* the signal $\mathbf{v}$ to get its coordinates relative to the subspace basis $\mathbf{B}$:

$$\mathbf{c} = \mathbf{B}^{T} \mathbf{v}$$

- If necessary, we can then *reconstruct* $\mathbf{v}_1$ as

$$\mathbf{v}_1 = \mathbf{B}\,\mathbf{c}$$

Analysis and reconstruction

Stochastic process

$\mathbf{v} \in \mathbb{R}^N$

PCA

$\mathbf{c} \in \mathbb{R}^M$

Reconstruction

$\mathbf{v}_1 \in \mathbb{R}^N$

$\mathbf{C} \in \mathbb{R}^{N \times N}$

In some application, for example, clustering, the reconstruction step is not relevant

Applications

In signal processing, PCA can be used for

- General data analysis
  - For an unknown data/signal determine if it can be reduced in dimension
- Data/signal compression
  - An N-dimensional signal can be represented with an M-dimensional basis ($M < N$)
  - Reduces the amount to data needed to store/transmit the signal
  - Data dependent compression
    - (+) Effective since the compression is data dependent
    - (-) Overhead since **B** must be stored/transmitted as well

Signal model

- PCA can typically be applied to signals with a statistical model (mean and **C** known)
- PCA can also be applied to a finite data set in the form of high-dimensional vectors
  - Dimensionality reduction
  - **C** is estimated from the finite data set

How large subspace?

- The idea behind PCA is to estimate **C** from a larges set of observation of the data/signal **v** and then choose the basis **B** as the *M* largest principal components
- How do we know a suitable value for *M*?
- No optimal strategy!
- Application dependent

How large subspace?

- In some applications *M* may be fixed, i.e., already given
- In other applications it makes sense to analyse the eigenvalues λ_1, ..., λ_N in order to choose a suitable *M*
  - For example: ϵ / ϵ_1 small (why?)
- Usually each dimension of *U* corresponds to a cost
  - Represents a coordinate of $\mathbf{v}_1$ that needs to be stored or transmitted
  - We want to keep *M* as small as possible
- Balance between cost and decrease in ϵ

An example

A 512 $\times$ 512 pixel image

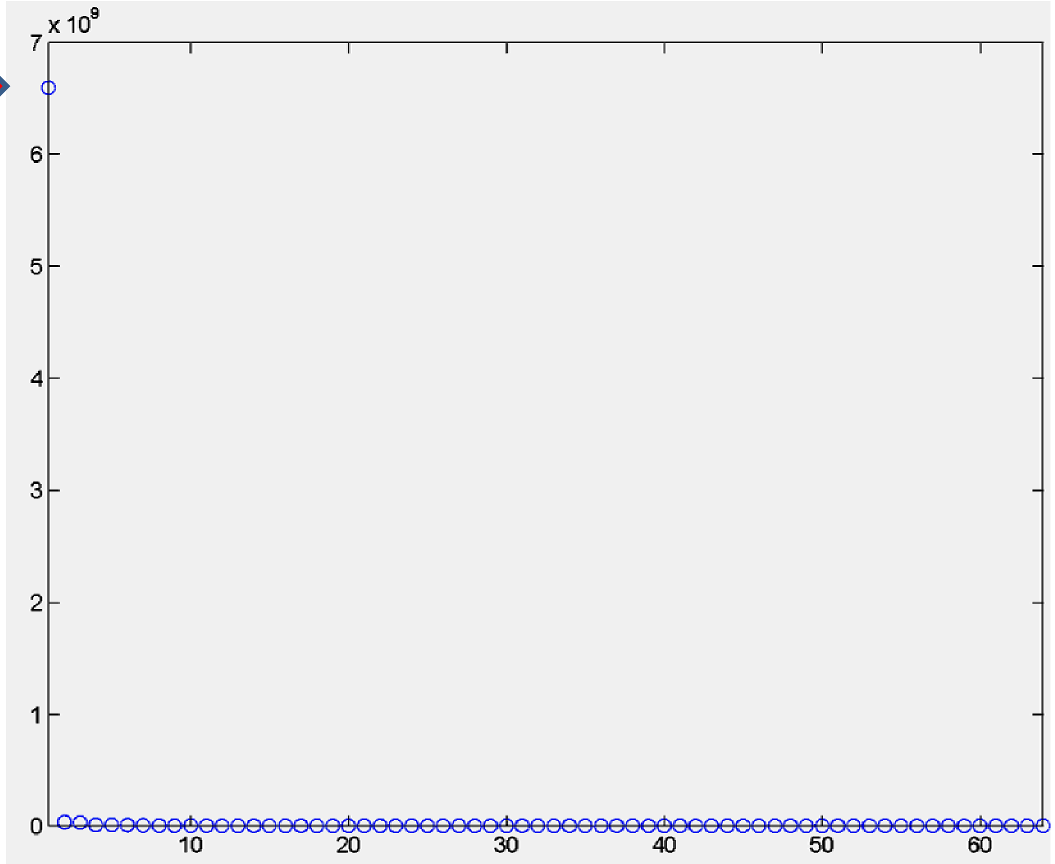
An example

- Let us divide this image into 8×8 pixels block
- This gives in total $64 \times 64 = 4096$ blocks
- The pixels of each block constitute our signal **v**, a vector in $\mathbb{R}^{64}$
  - (8×8 reshaped to a 64-dim column vector)
- From the 4096 observations of **v** we form the 64×64 correlation matrix **C**
- We also compute the eigenvalues and eigenvectors of **C**

The eigenvalues of **C**

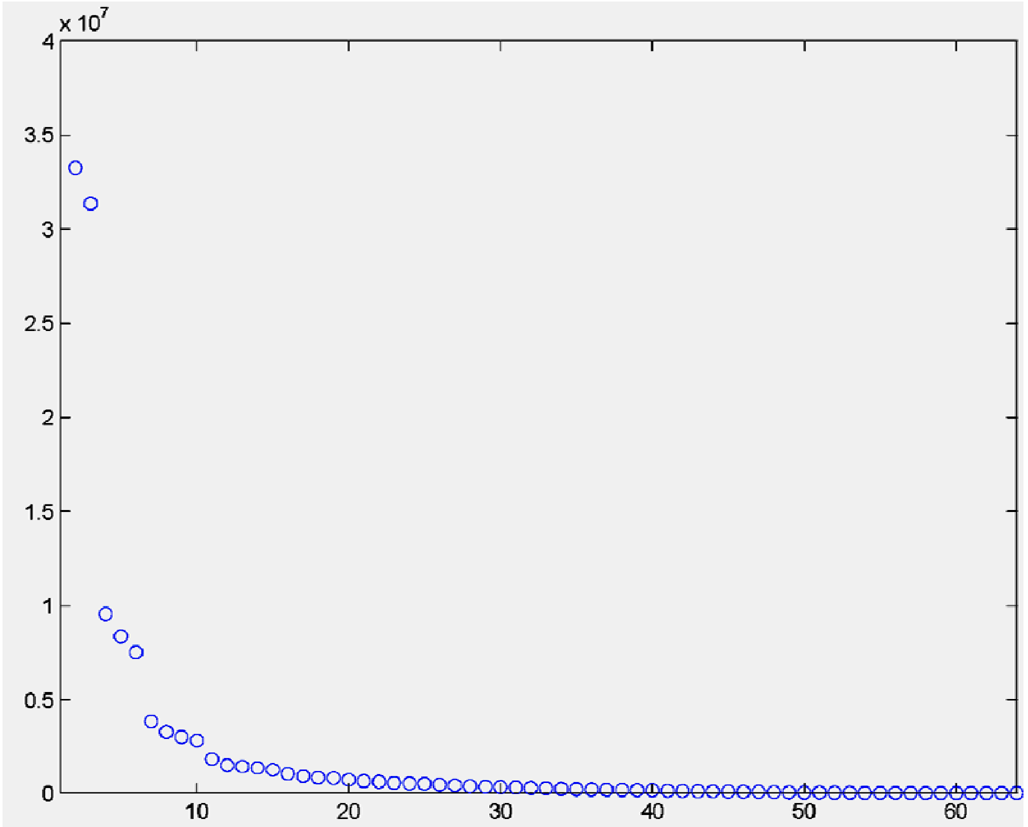
- Here are the 64 eigenvalues of **C**

One large eigenvalue, the rest are relatively small



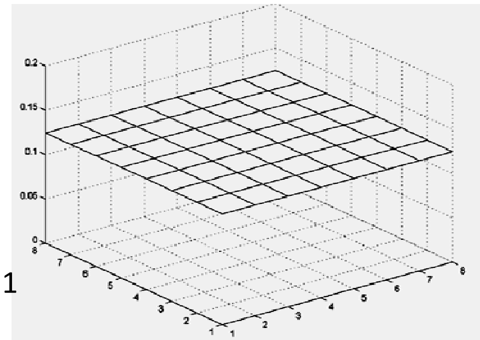
The eigenvalues of **C**

- If we remove the largest eigenvalue and plot the rest:

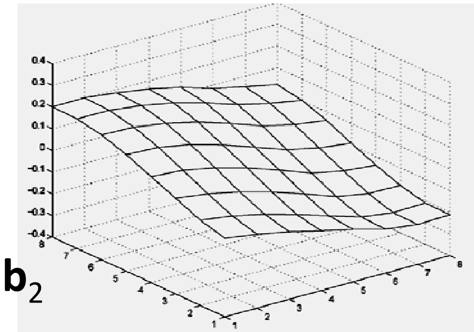


The eigenvectors of **C**

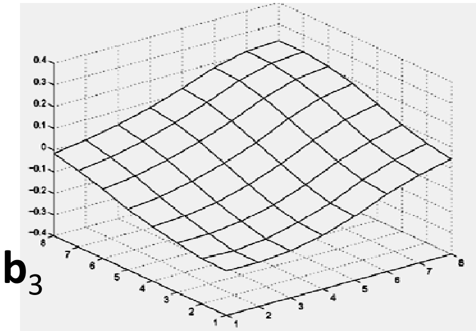
- Each eigenvector of **C** is a 64-dimensional vector that can be reshaped into an 8×8 block



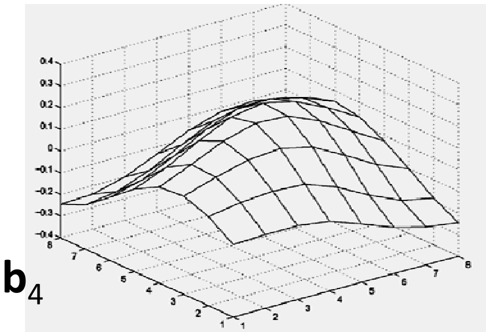
$\mathbf{b}_1$



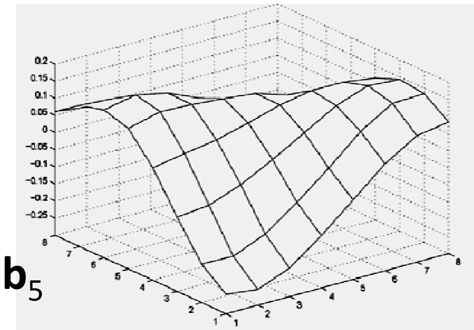
$\mathbf{b}_2$



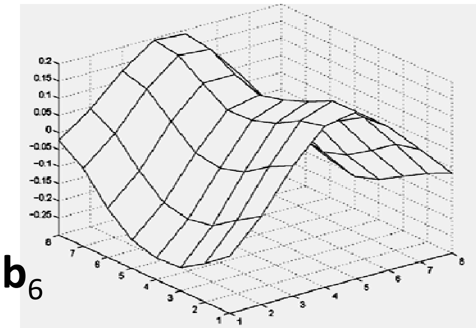
$\mathbf{b}_3$



$\mathbf{b}_4$



$\mathbf{b}_5$



$\mathbf{b}_6$

The eigenvectors of **C**

We notice that

- $\mathbf{b}_1$ is approximately flat or constant (DC)
- $\mathbf{b}_2$ and $\mathbf{b}_3$ are approximately shaped like a plane with a slope in two perpendicular directions. Linear
- $\mathbf{b}_4$, $\mathbf{b}_5$ and $\mathbf{b}_6$ are approximately shaped like quadratic surfaces

An example

- The distribution of the eigenvalues implies that already by going from 0 to 1 dimension of U the error ϵ is reduced quite a lot
- By adding a few more dimensions it should be possible to represent the signal quite well
- Let us try with some different values for M and look at the result

An example

- Each block, as a vector $\mathbf{v} \in \mathbb{R}^{64}$, is projected onto $\mathbf{v}_1 \in U$
- $\mathbf{v}_1$ can be represented with only M coordinates
- $\mathbf{v}_1$ is a reasonably good approximation of $\mathbf{v}$, at least the mean error should be low
  - How good is determined by the distribution of eigenvalues relative to M

$M = 1$

- Here **v** is projected onto the first principal component $\mathbf{b}_1$
- Since $\mathbf{b}_1$ is flat or constant, each block becomes flat or constant

$M = 2$

- Here **v** is projected onto the first two principal components $\mathbf{b}_1$ and $\mathbf{b}_2$
- Since $\mathbf{b}_2$ is a slope in one direction we can now represent that change in each block
  - But not a slope in the orthogonal direction!

$M = 3$

- Here **v** is projected onto the first three principal components $\mathbf{b}_1 \ldots \mathbf{b}_3$
- Since $\mathbf{b}_3$ is a slope in the other direction we can now have slopes in any direction within each block

M=6

- Here **v** is projected onto the first six principal components $\mathbf{b}_1 \dots \mathbf{b}_6$
- With 3 more dimensions, each block can contain more details
- Here, data has been reduced by a factor $64/6 \approx 11$

General observation

A general observation:

- In areas of low spatial frequency the approximation is good
- In areas of high spatial frequency the approximation is worse

- The more details or higher spatial frequency, the more dimensions are needed for an accurate representation

Karhunen-Loève transformation

- Introducing the new basis **B** in this way and computing the coordinates of **v** relative to **B** is sometimes also referred to as *Karhunen-Loève transformation*

$\mathbf{B}^T\mathbf{v}$ gives the "transformed signal"
(= the coordinates of **v** relative to basis **B**)

$\mathbf{B}\,\mathbf{B}^T\mathbf{v}$ reconstructs the projected signal $\mathbf{v}_1$

Covariance and mean

- In some application PCA is used to make a statistical analysis of a signal $\mathbf{v}$
- It is then very common to describe $\mathbf{v}$ in terms of its mean $\mathbf{m}_\mathbf{v}$ and its **covariance matrix**

$$\mathbf{m}_\mathbf{v} = E\,[\,\mathbf{v}\,]$$

$$\mathrm{Cov}_\mathbf{v} = E\,[\,(\mathbf{v} - \mathbf{m}_\mathbf{v})\,(\mathbf{v} - \mathbf{m}_\mathbf{v})^T\,]$$

Covariance and mean

- The PCA is then based on the covariance matrix $\mathrm{Cov}_{\mathbf{v}}$ rather than the correlation $\mathbf{C}$
- This corresponds to translating the origin of the subspace U to the mean $\mathbf{m}_{\mathbf{v}}$ and do correlation based PCA there
- Both approaches are called PCA
  - Statistical data analysis typically use $\mathrm{Cov}_{\mathbf{v}}$
  - Data compression typically use $\mathbf{C}$

The data matrix

- In the case that we approximate the statistics of **v** from a finite number of *P* observations, these can be described by a data matrix **A** that holds all these **v** in its columns
- An estimate of $\mathbf{C} = E[\mathbf{v}\,\mathbf{v}^T]$ is then given by

$$\mathbf{C} = (1/P)\,\mathbf{A}\,\mathbf{A}^T$$

SVD vs EVD

- The principal components are the eigenvectors of $\mathbf{A}\,\mathbf{A}^T$
- These are also the left singular vectors of $\mathbf{A}$
- An alternative to computing the PCA, that in some cases may have better numerical properties:
  1. Form the data matrix $\mathbf{A}$
  2. Compute its SVD
  3. Form $\mathbf{B}$ from the left singular vector that corresponds to the M largest singular values

What you should know includes

- Formulation of the problem that PCA solves:
  - Find a subspace *U* that minimises $\epsilon = \mathrm{E} \, \| \mathbf{v} - \mathbf{B} \, \mathbf{B}^T \mathbf{v} \|^2$
  - This subspace is represented by an ON-basis **B**
- Solution: **B** consists of the *M* "largest" eigenvectors of $\mathbf{C} = \mathrm{E}[\, \mathbf{v} \, \mathbf{v}^T \,]$
  - Called *principal components*
  - Alternatively computed by means of SVD
  - ϵ = sum of residual eigenvalues of C
- Applications:
  - Dimensionality reduction
  - Signal compression