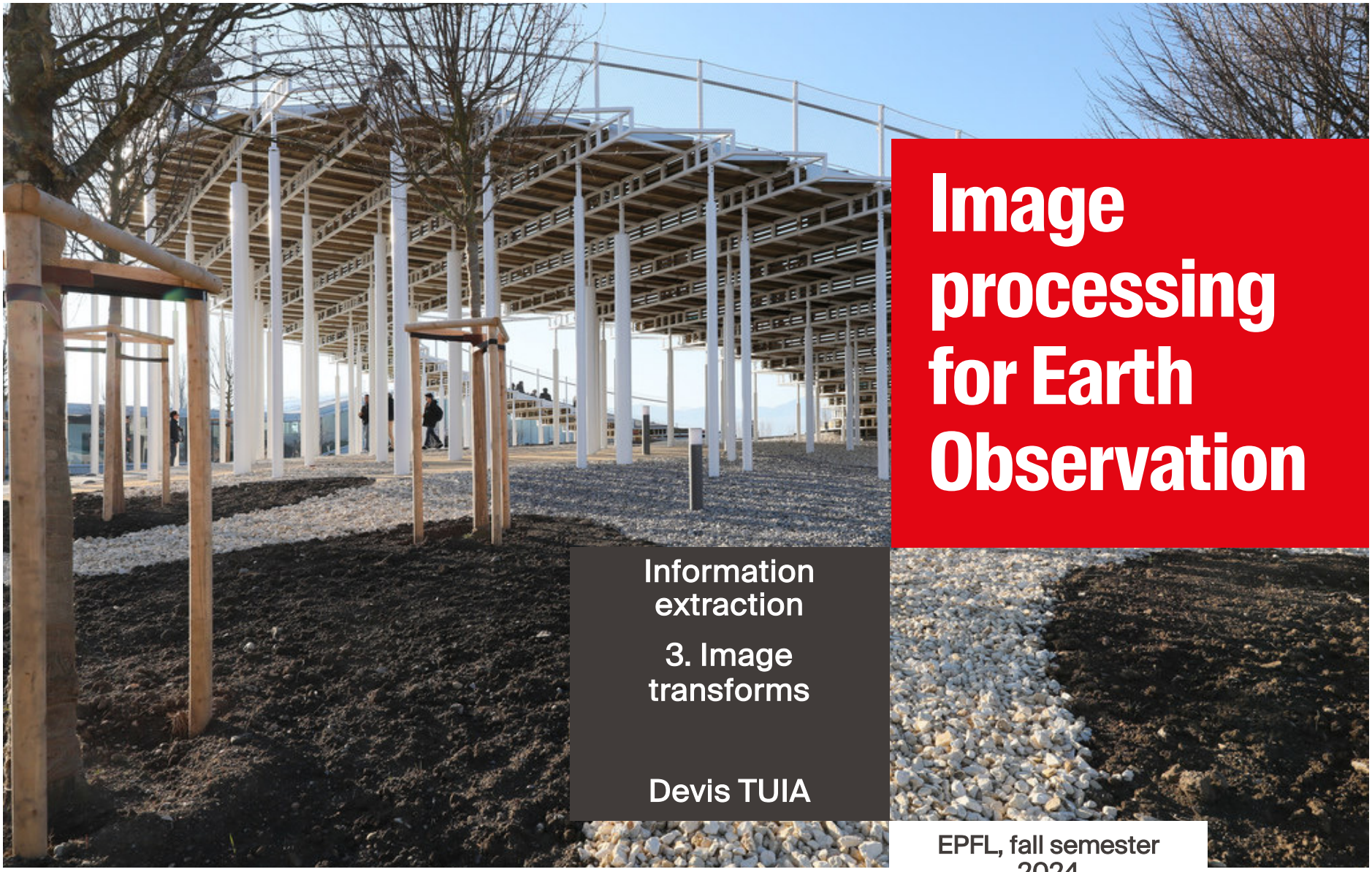


Image processing for Earth Observation

Information extraction

3. Image transforms

Devis TUIA

École polytechnique fédérale de Lausanne

EPFL, fall semester 2024

In the next two weeks

- We will study a number of information extraction techniques
  - spectral indices: enhance spectral relations between the bands of a pixel
  - spatial indices: extract information about spatial relationships
- We will also discuss how to deal with the increase in number of variables and see some data reduction techniques

Where we left last time

$$\mathbf{x}_i$$

- is a feature descriptors vector containing information about spatial context of a pixel within a patch / region / object
- is used to *train a classifier, learn* input-to-label information (next class)
- it can be in terms of colors or of typical visual patterns (the visual words in bow)

$$\mathbf{x}_i = [\mathbf{x}_i^{\text{av}} \quad \mathbf{x}_i^{\text{std}} \quad \mathbf{x}_i^{\text{entr}} \quad \mathbf{x}_i^{\text{hist}} \quad \mathbf{x}_i^{\text{bow}} \quad \dots]$$

How big is that?

$$\mathbf{x}_i$$

- is a high-dimensional feature vector

$$\mathbf{x}_i = [\mathbf{x}_i^{\text{av}} \quad \mathbf{x}_i^{\text{std}} \quad \mathbf{x}_i^{\text{entr}} \quad \mathbf{x}_i^{\text{hist}} \quad \mathbf{x}_i^{\text{bow}} \quad \ldots]$$

- its dimensionality can be from ~10-20 to 100-200000!!

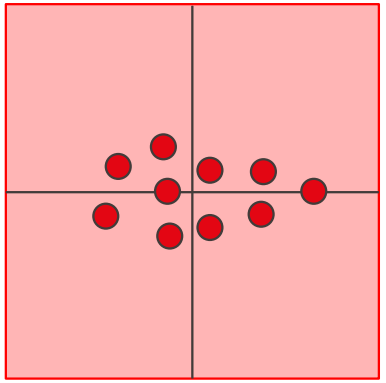
Is that a problem?

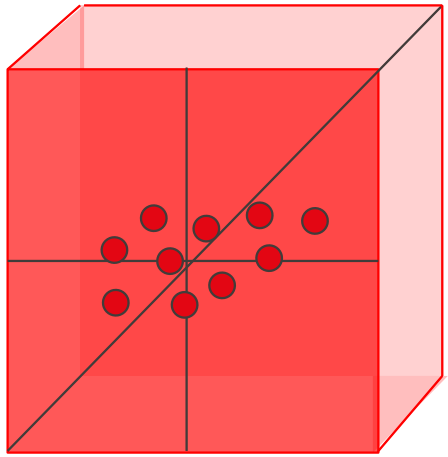
The problem of data dimensionality

- Yes.
- Problem 1: with so many dimensions, the informative ones remain hidden
- We call that the CURSE OF DIMENSIONALITY

The problem of data dimensionality

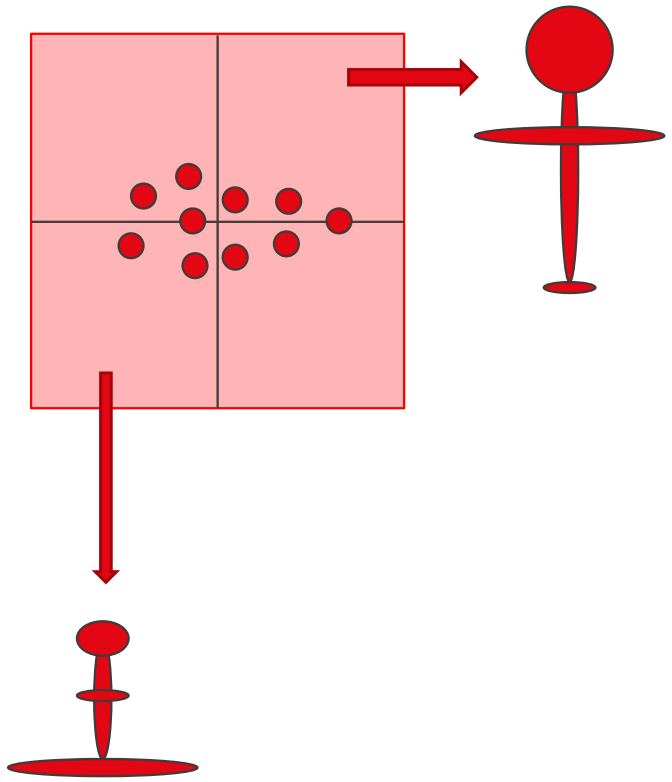
- Yes.
- Problem 1: with so many dimensions, the informative ones remain hidden
- Fact 1: the higher the dimension, the bigger and emptier the space.
  - 1 D is a line
  - 2D is a square
  - 3D is a cube
  - ...





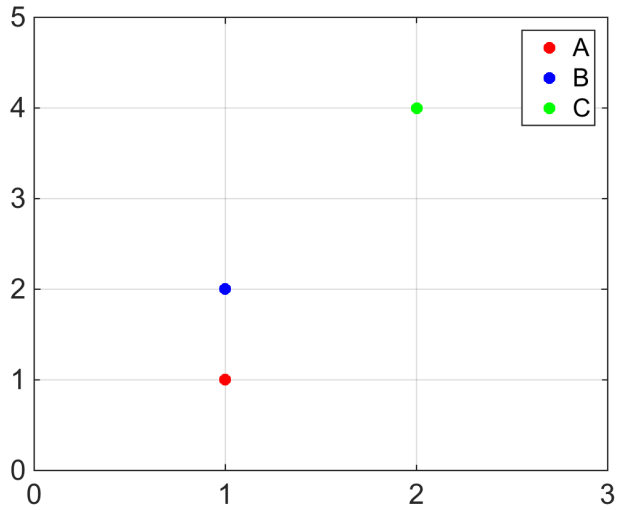
The problem of data dimensionality

- Yes.
- Problem 1: with so many dimensions, the informative ones remain hidden
- Fact 2: The emptier the space, the most crowded some areas become.
  - (not all possible combinations happen)
  - ex: people 2m high with 25 shoe size are unlikely.



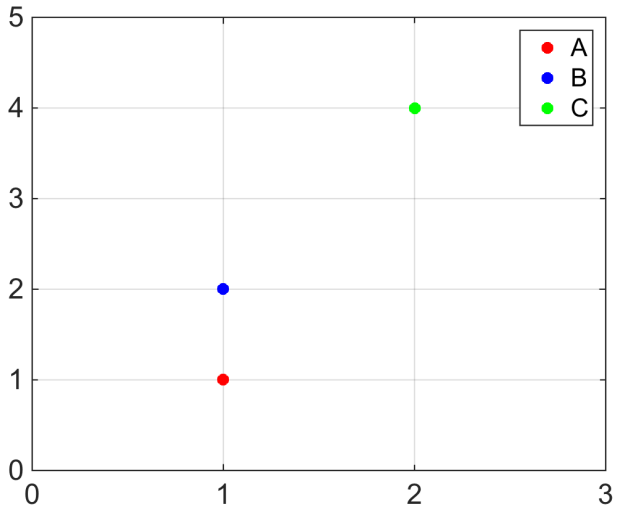
The problem of data dimensionality

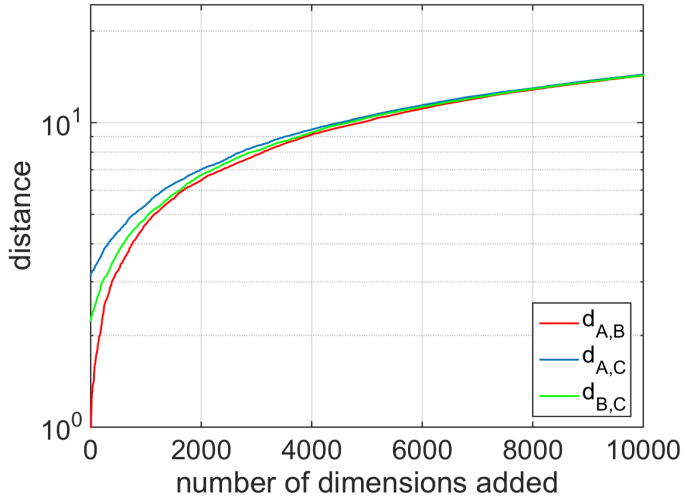
- Yes.
- Problem 1: with so many dimensions, the informative ones remain hidden
- Consequence: since most dimensions are uninformative, noise adds up and becomes stronger than the discriminative signal.



The problem of data dimensionality

- Yes.
- Problem 1: with so many dimensions, the informative ones remain hidden
- Consequence: **in high dimensions, standard distances measures lose their meaning**
- Many methods of ML are based on distances.





The problem of data dimensionality

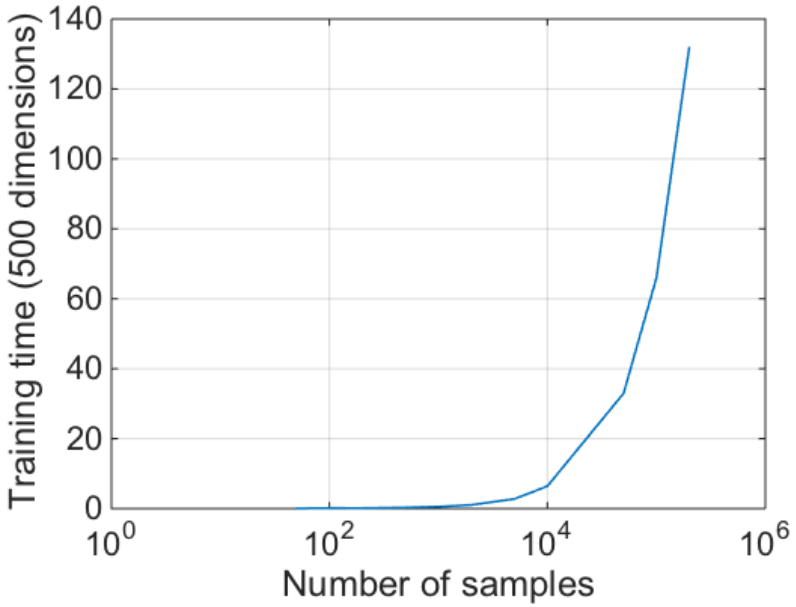
- Yes.
- Problem 2: with so many dimensions, methods are SLOW
- Ex: a support vector machine (SVM) is $O(n^2d)$ complex.

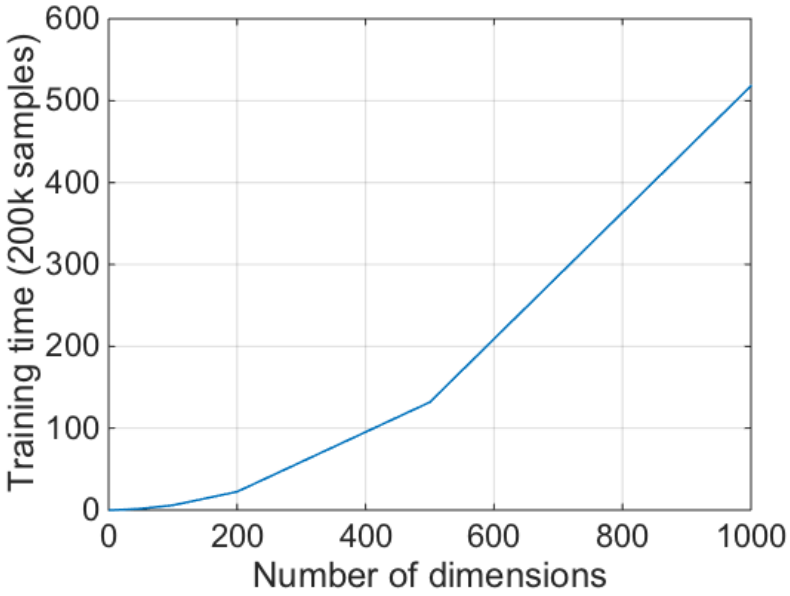
  It means:
  - for every additional example, it must compute n^2 operations
  - for every additional dimension, it must compute an additional operation.

Example: classifying an hyperspectral image

ROSIS sensor
102 bands + 1000 noise bands

Model training times (in seconds)





Revisiting the "How big is that?" slide

$$\mathbf{x}_i$$

- is a high-dimensional feature vector

$$\mathbf{x}_i = [\mathbf{x}_i^{\text{av}} \quad \mathbf{x}_i^{\text{std}} \quad \mathbf{x}_i^{\text{entr}} \quad \mathbf{x}_i^{\text{hist}} \quad \mathbf{x}_i^{\text{bow}} \quad \dots]$$

- its dimensionality can be from ~10-20 to 100-200000!!
- The amount of descriptors must be in accordance to the number of labeled samples and type of classifier
- The more diverse and abundant the information is, the better the classifier can perform!

This is where data transform come into play

So we need a way to fight the curse (of dimensionality)!

- Reduce the amount of data to be processed
- Keep only what is important (or informative) according to some criterion
- Throw away the garbage (noisy features, unrelated ones, …)

Orthogonal transforms

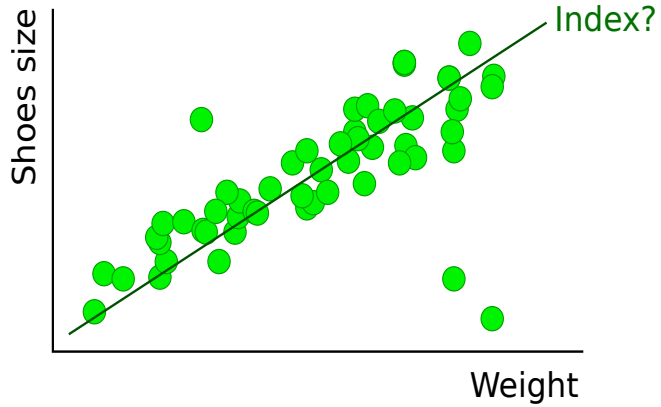
- An approach is to transform (globally) the spectral space to obtain an orthogonal variables system
- The orthogonal system is obtained by a linear transformation of the spectral space
- Since it is orthogonal, the obtained variables (components) are decorrelated
- BUT: in the orthogonal system, the bands lose their physical meaning
  (they become linear combination of the original features)

And one to rule them all: PCA

- Principal Components Analysis
- Good old statistical method from the ‘60s
- Finds the projection that maximizes information (described in terms of variance of the original data explained) contained in the data
- Again: new variables are decorrelated, but meaningless

Why decorrelated

- Bands are correlated to each other
- Maybe a linear combination can be more effective (1 index instead of 2)



- In a nutshell:
  - extracting new descriptors as combinations of the existing ones
  - if one descriptor summarizes all redundant information of the original data, we can use those subset instead

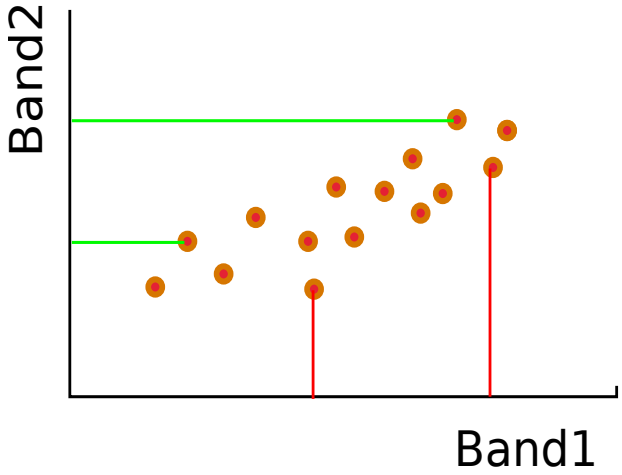
Projection (idea)

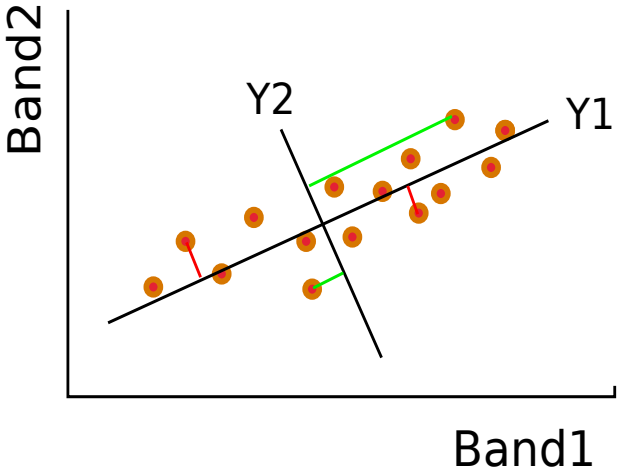
- Projecting a dataset means finding a linear combination of the original variables

- Ex:
  - given a dataset X (shoes size and weight, or … bands in an image)
  - we want to find a projection matrix W
  - projecting in an orthogonal space Y

$$Y = XW$$

Projection (idea)





Values on band 1: $x_i^{(b1)}$

Values on band 2: $x_i^{(b2)}$

Values on projected component Y1:
$y_1^{(1)} = x_1^{(b1)} w(1,1) + x_1^{(b2)} w(2,1)$

Values on projected component Y2:
$y_1^{(2)} = x_1^{(b1)} w(1,2) + x_1^{(b2)} w(2,2)$

The projection matrix

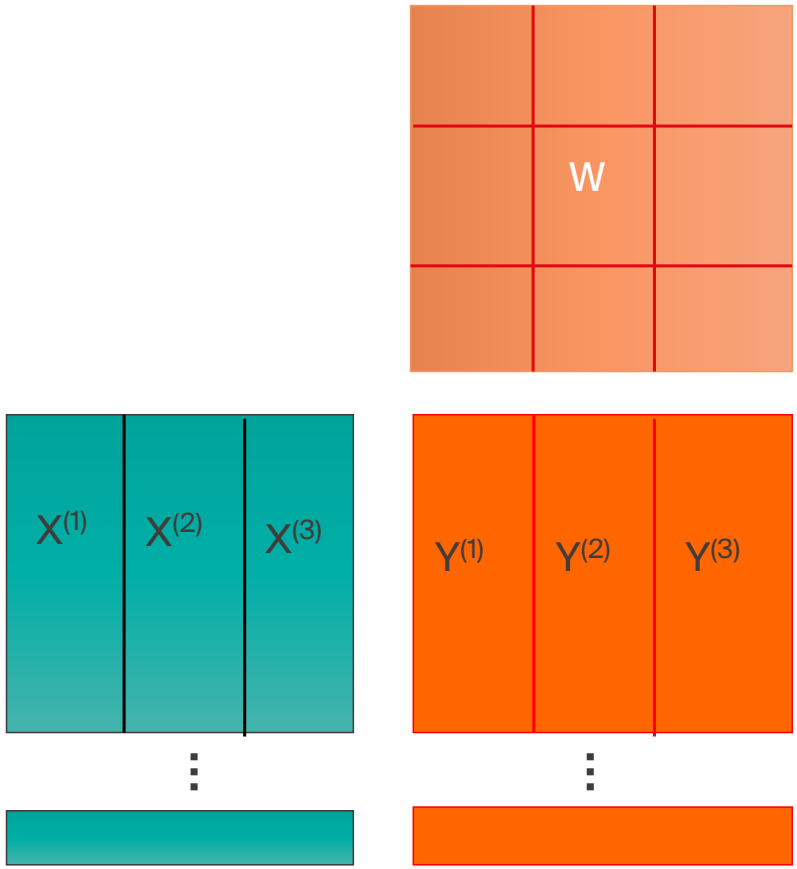
$$Y = XW$$

- W is a square matrix
- As many lines and columns as bands
- Ex: 3 bands

$$W = \begin{bmatrix} W(1,1) & W(1,2) & W(1,3) \\ W(2,1) & W(2,2) & W(2,3) \\ W(3,1) & W(3,2) & W(3,3) \end{bmatrix}$$

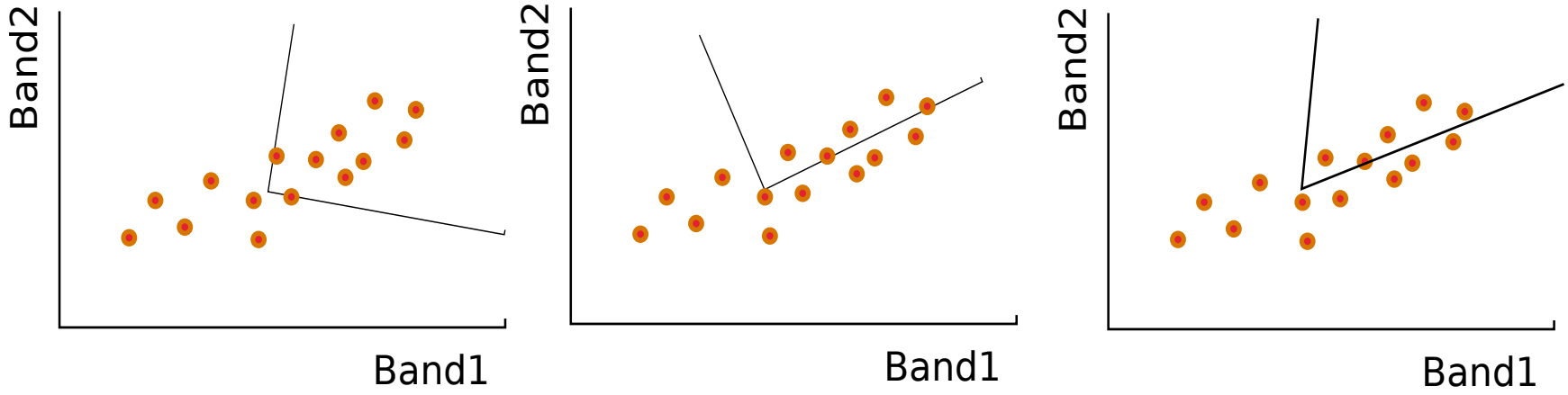
Multiplies band 2

Projects into component 1



D. Tuia. ECEO

Projection (idea)



Orthogonal

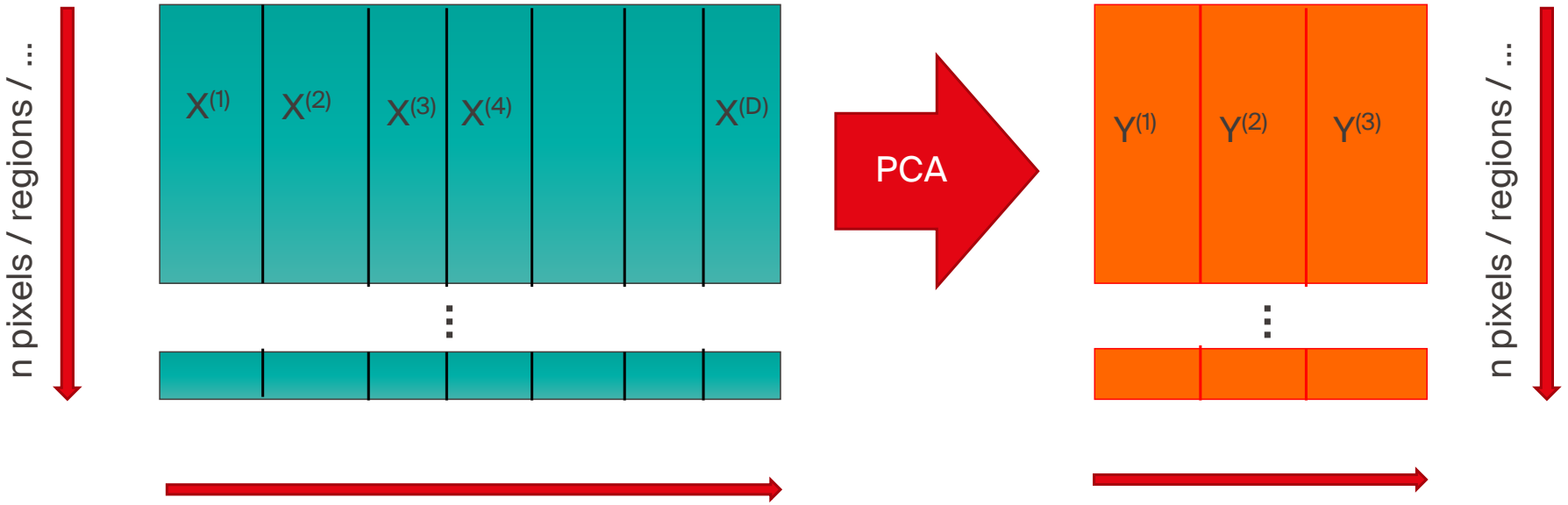
Orthogonal

Not orthogonal!

Variance explained as a way to select "how many components to keep"

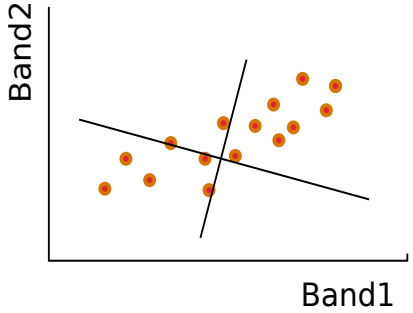
- To choose among projections, we apply a variance maximization criterion
- The bands are correlated between each other
- Projecting must be a way of compacting information (in terms of variance)
- In a nutshell: we group relevant information in the first component, and then leave the less informative for the last components

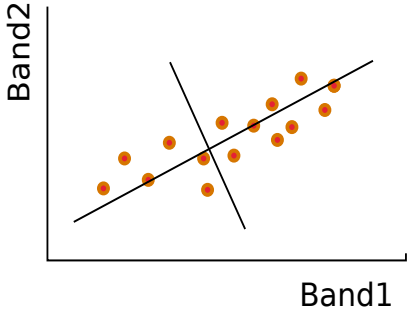
At the end, it helps reducing dimensionality

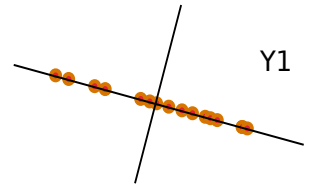


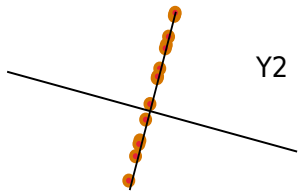
Projecting onto principal components

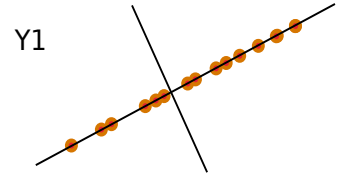
- Each projection corresponds to a different view on data
- left: not interesting, does not compact variance
- right, the first component maximizes variance, the second has little fluctuations

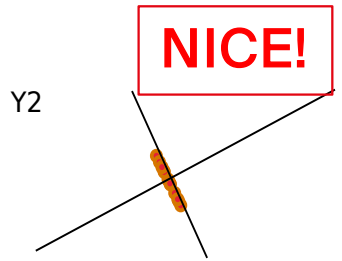






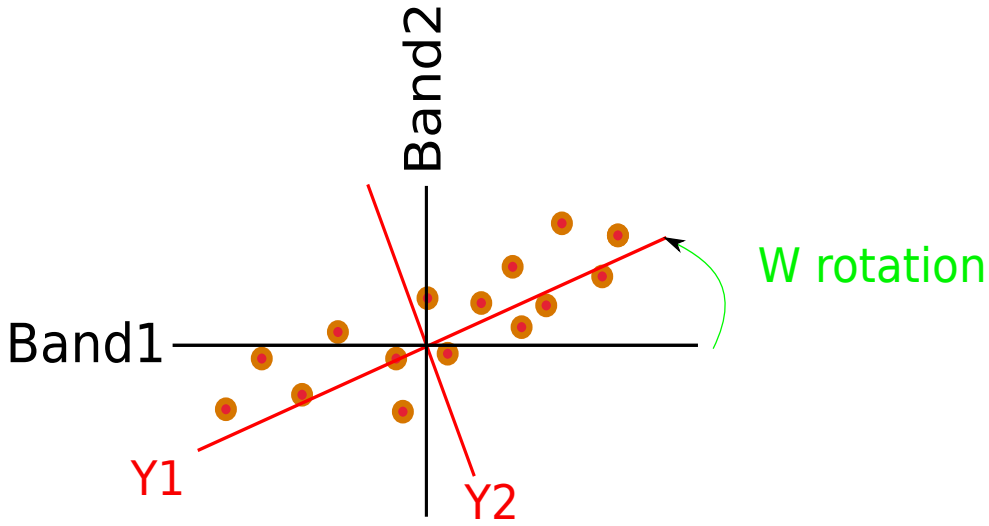






Projecting is basically...

- …a rotation! (W is a rotation matrix)
- We must then select the rotation matrix maximizing variance for the first component

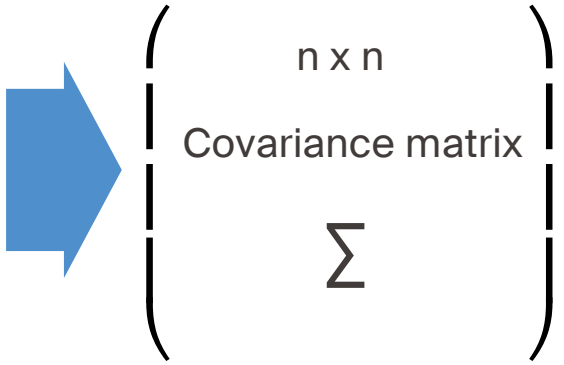


And now mathematically

- The base of PCA is the Covariance matrix S
- Covariance provides an insight on how similar two variables are

Bands

Bands	b_1	b_2	...	b_n
b_1	var(1)	cov(1,2)	...	cov(1,n)
b_2	cov(2,1)	var(2)	...	cov(2,n)
...	...	...	...	...
b_n	cov(n,1)	cov(n,2)	...	var(n)



$$\frac{1}{N+1} X^\top X$$

n bands → n lines, n columns

And now mathematically

- The off-diagonal terms, cov($band_1$,$band_2$), show how much variables "covariate"
- The diagonal terms, var($band_1$), shows the internal variance of each variable → its information, its energy
- We want to diagonalize S, to decorrelate the components (no covariance between components)
- We keep them orthogonal by adding a constraint

And now mathematically

- The final solution is the following

Rotation matrix

$$\Sigma = W \Lambda W^{\top}$$

Diagonal matrix with variance of each component

- Which can be obtained by the following linear equations system:

$$\Sigma w = \lambda w$$

Rotation vectors = eigenvectors

Variance of each component (entries of Λ) = eigenvalues

And now mathematically

The objective to be pursued is to have projected data of maximal variance

This problem can be solved by the diagonalization of the covariance matrix

(mathematical details on request, “Bie et al., Eigenproblems in Pattern Recognition”)

data are centered (zero mean)

$$\text{var}\ (\text{y}) = \sum_i (y_i - \bar{y})^2 = \sum_i (y_i - 0)^2$$

$$w = \arg \max_{||w||=1} var(Y)$$

$$\arg \max_{||w||=1} ||Xw||^2$$

$$\arg \max_{||w||=1} (Xw)^\top (Xw)$$

$$\arg \max_{||w||=1} w^\top X^\top X w$$

$$\arg \max_{||w||=1} w^\top \Sigma w$$

It's in most software suites (e.g. Matlab)

```
[eigenVect,proj,eigenVal] = pca(X);
```

In

X: your original data (samples x features)

Out

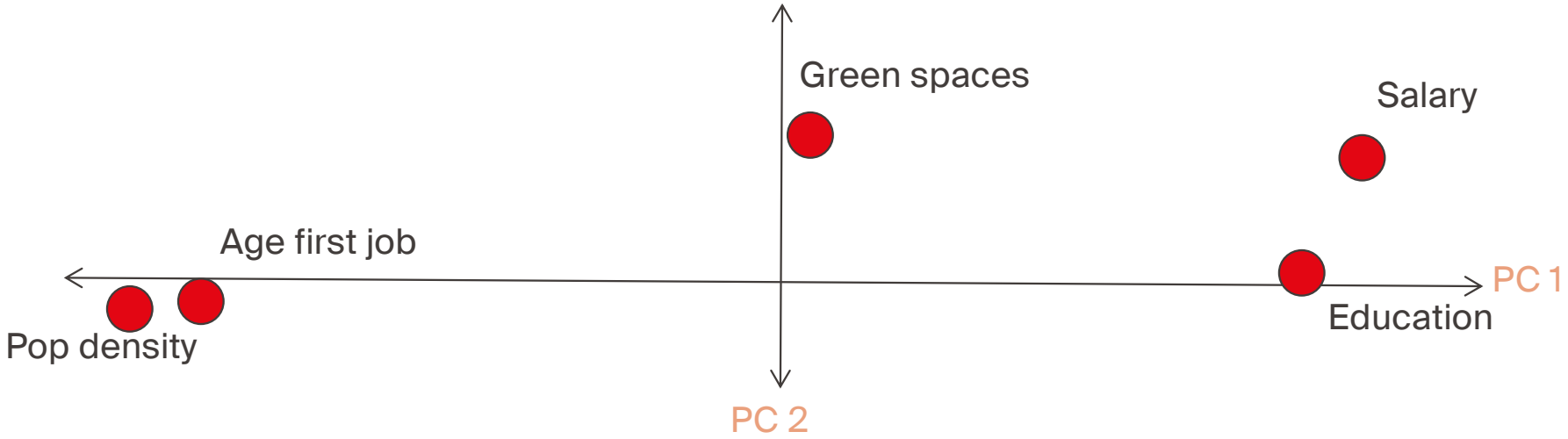
eigenVect: the projection vectors, or eigenvectors (W, of size (features x components))

proj: the projected data, you don't even have to compute that. (samples x components)

eigenVal: the variance of each component in decreasing order, ordered as the eigenvectors. (components x 1)

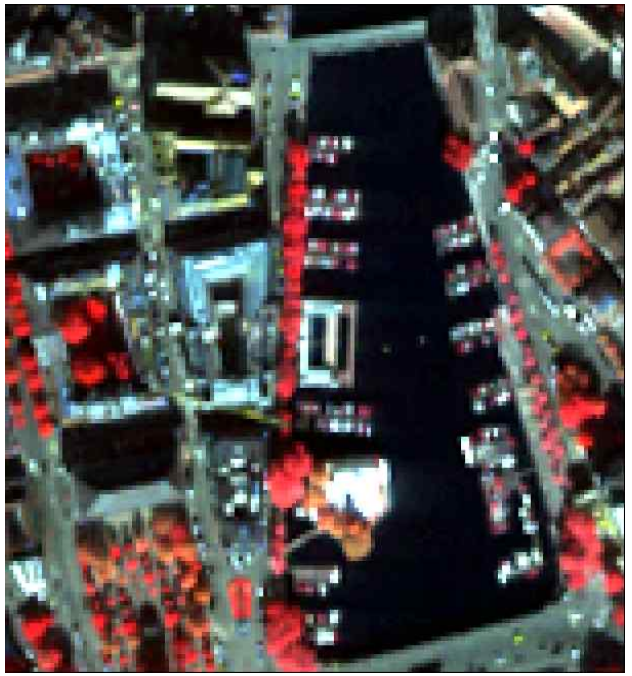
Is that REALLY meaningless?

- No.
- The components lose their physical meaning, but are still combinations of the original variables.
- Actually looking at the eigenvectors (eigenVect) allows you to see which variable correlates with which factor and to *a-posteriori* interpret them.



Back to remote sensing: an example

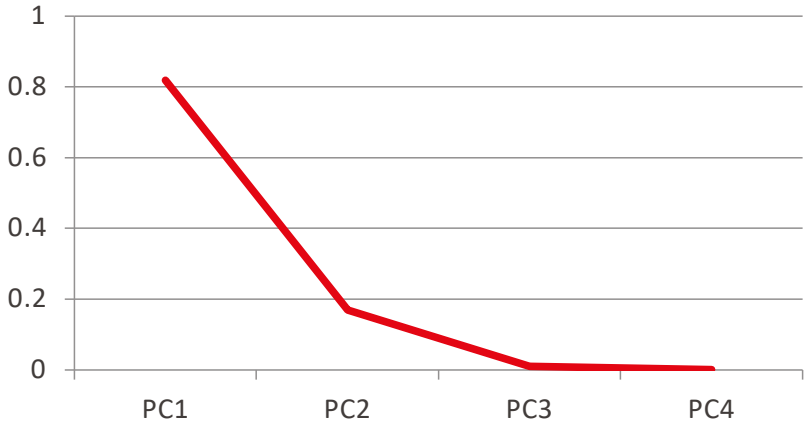
- QuickBird sensor
- 4 bands (B – G – R – NIR)
- 2.4m resolution
- Frauenbad, Zurich



Question: how many PCs will we be able to extract?

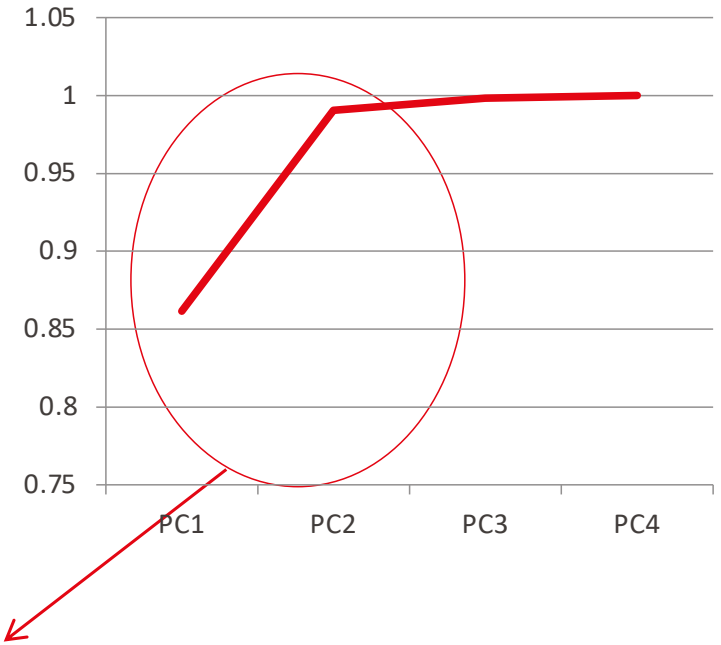
Back to remote sensing: an example

Variance explained



$$\frac{\lambda_b}{\sum_{b=1}^{B} \lambda_b}$$

Cumulative variance explained



99% of information in the two first components!

And here the four components!

PC1

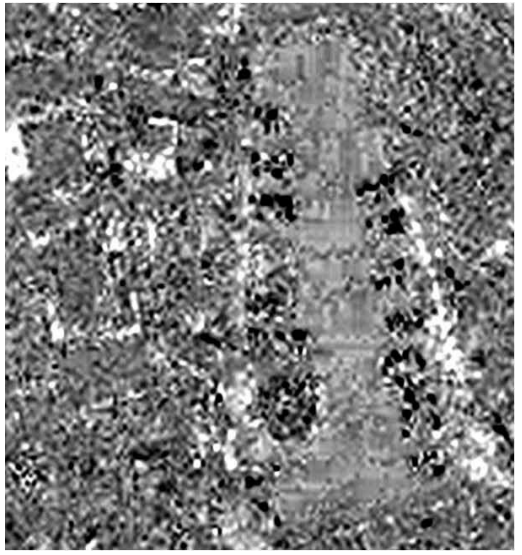
How many would you keep?

Ok, it is maybe not super impressive.. but we had only 4 descriptors to start with (4 bands)

What if we had 10’000?

PC3

PC2



PC4

Other possibilities

Feature selection instead of data transforms:
select the most informative features in the set of features you have

- By clustering:
  - Transpose your dataset X = X', so that samples are features and variables are datapoints;
  - Run a kmeans with k = "how many feature you want to keep";
  - Therefore k-means "groups" features instead of samples
  - For each cluster, keep the one closest to the cluster center.

- By recursive feature elimination (Guyon et al., 2002):
  - Run a model with all features
  - Run submodels with one feature removed each,
  - Remove feature with least impaxct on result
  - Repeat

Guyon, I., Weston, J., Barnhill, S. and Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. *Machine learning*, *46*, pp.389-422.

Example with RFE

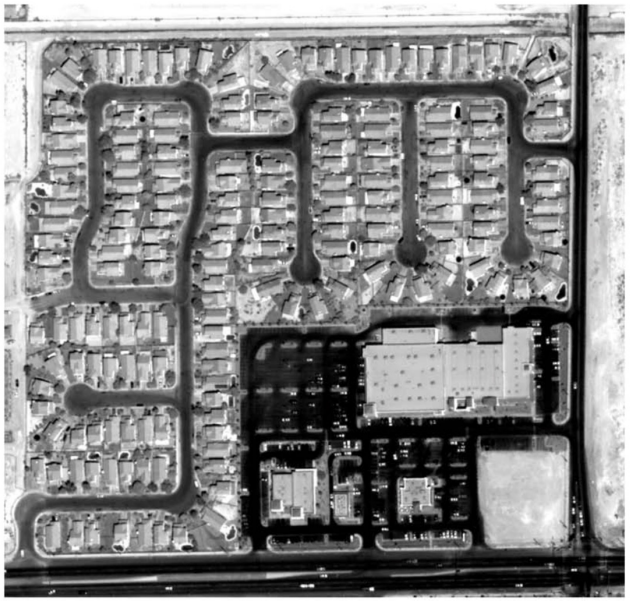


Image
1 band, panchromatic

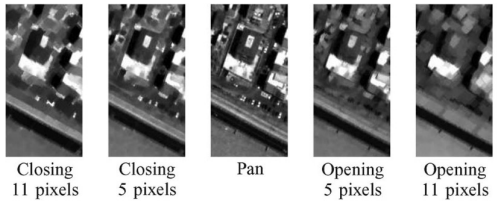


Fig. 2. Progressive opening and closing using a diamond-shaped SE. On the left end is the closing image produced using an 11-pixel SE, in the middle is the original image, and on the right end is the opening image produced using an 11-pixel SE.

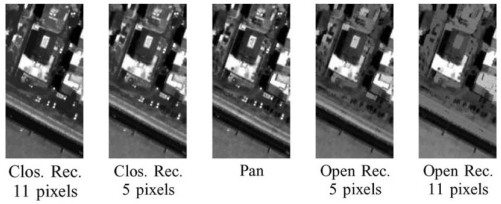


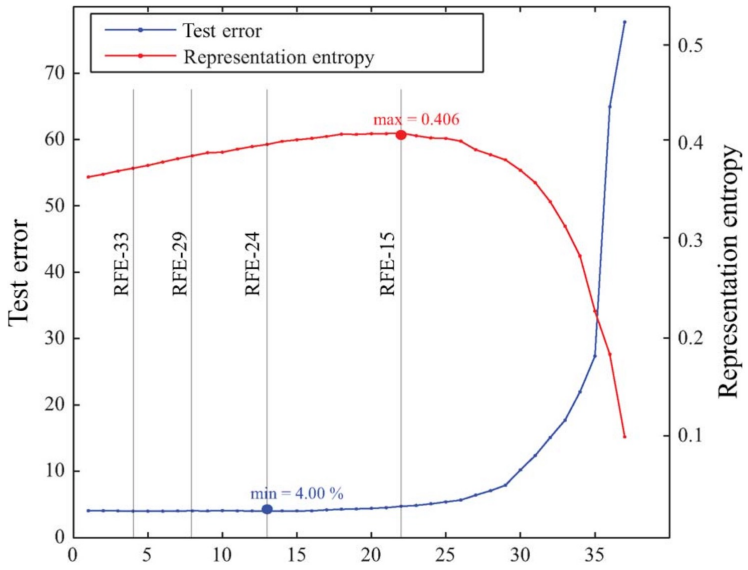
Fig. 3. Progressive opening and closing by reconstruction using a diamond-shaped SE. On the left end is the CR image produced using an 11-pixel SE, in the middle is the original image, and on the right end is the OR image produced using an 11-pixel SE.

54 features
Mathematical morphology
(texture features)

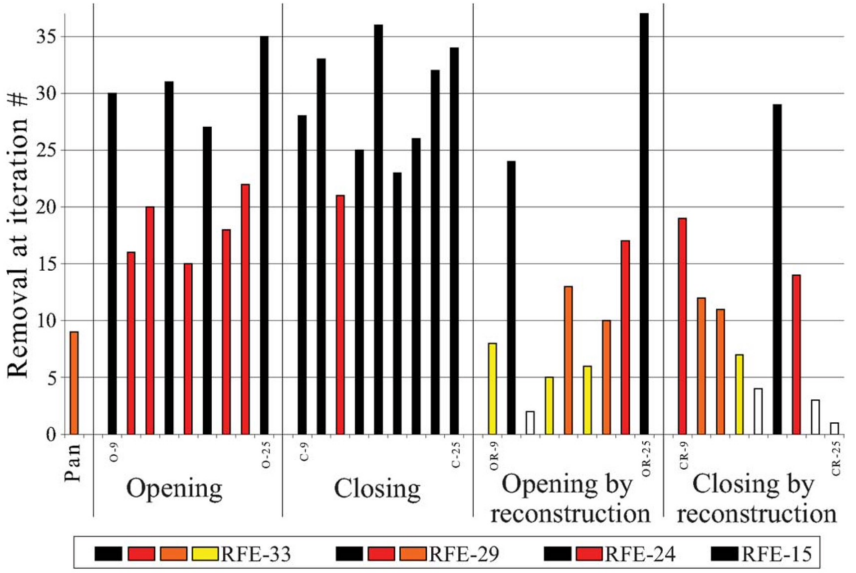
Result

D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery. Classification of very high spatial resolution imagery using mathematical morphology and support vector machines. *IEEE Trans. Geosci. Remote Sens.*, 47(11):3866–3879, 2009.

Example with RFE



Evolution of error (blue) when removing Features one by one

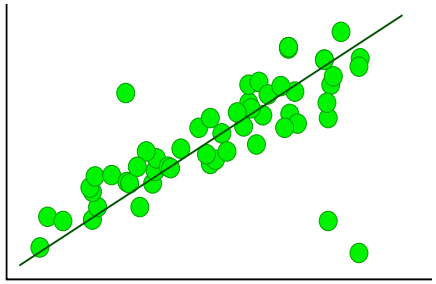


Interpretability: which features were retained when keeping 33, 29, 24 or 15.

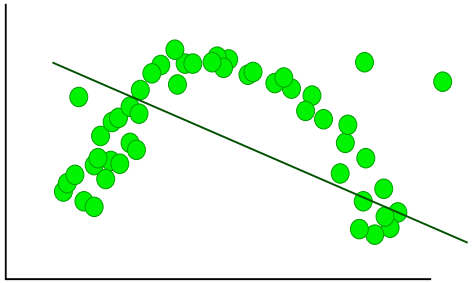
D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery. Classification of very high spatial resolution imagery using mathematical morphology and support vector machines. *IEEE Trans. Geosci. Remote Sens.*, 47(11):3866–3879, 2009.

In summary

- We have seen why high dimensionality can be a problem for machine learning methods
- We have seen one data reduction method, principal component analysis
- It is very much used to reduce the number of variables (descriptors), and in all areas of science
- But remember! Decorrelation is not independence!



Linear relation
PCA works well



Non linear relation
PCA does not work well

- We have also briefly discussed feature selection.