

Unsupervised learning

Inteligencia Artificial en los Sistemas de Control Autónomo
Máster en Ciencia y Tecnología desde el Espacio

Departamento de Automática

Objectives

i. TODO

Bibliography

- TODO Bishop, Christopher M. *Pattern Recognition and Machine Learning*. 2nd edition. Springer-Verlag. 2011
- TODO Müller, Andreas C., Guido, Sarah. *Introduction to Machine Learning with Python*. O'Reilly. 2016

Table of Contents

- 1. Clustering
 - Applications
- 2. K-means
 - Overview
 - K-means algorithm
 - K-means limitations
 - Elbow's method
 - Application: Image segmentation
 - Application: for semi-supervised learning
 - K-means summary
- 3. Other clustering algorithms
 - GMM
- DBSCAN
- Summary
- Agglomerative clustering
- Agglomerative clustering: Summary
- 4. Anomaly detection
- 5. Dimensionality reduction
 - Main approaches for dimensionality reduction
 - PCA
 - Kernel PCA
 - Manifold learning
 - Locally Linear Embedding (LLE)
 - Other manifold techniques

Clustering

K-means, agglomerative clustering, DBSCAN and GMM

Clustering Applications

Set of unsupervised techniques that identify groups of data (named clusters)

- No universal definition of cluster: Centroid, medoid, dense regions, etc

Applications

- Customer segmentation
- Data analysis
- Dimensionality reduction
- Anomaly detection
- Semi-supervised learning
- Search engines
- Image segmentation

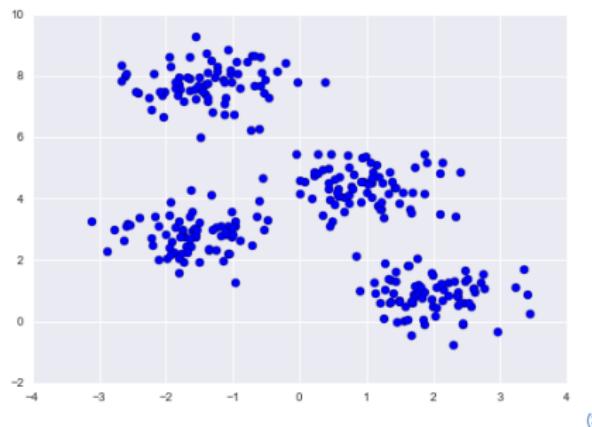
Main algorithms

- K-means, DBScan, GMM, hierarchical clustering, EM, ...

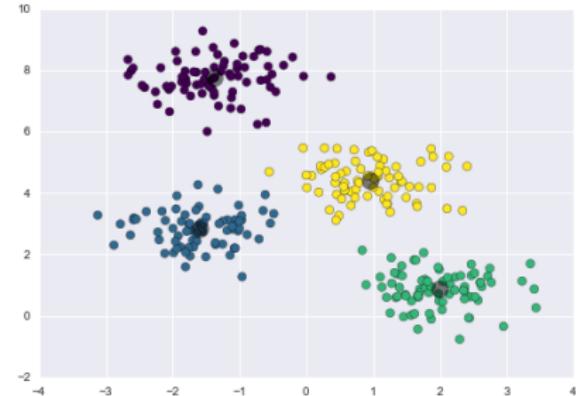
K-means

Overview

Original data



Clustered data



In k-means, clusters are identified by a **centroid**

K-means

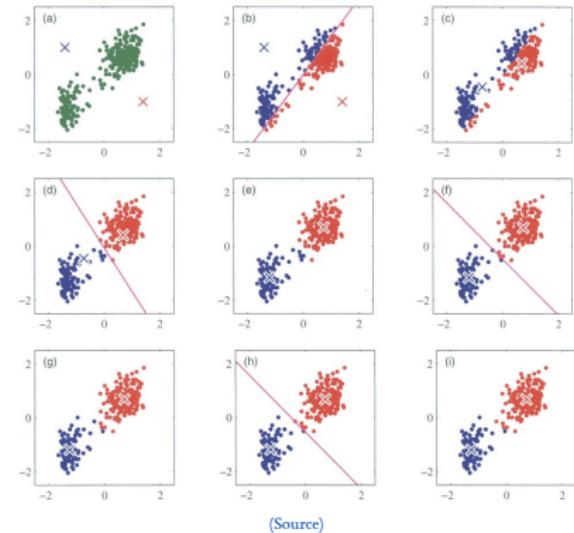
K-means algorithm (I)

K-means algorithm

1. Set k random centroids
2. Assign each data point to its closest centroid
3. Recompute centroids
4. Go to 2 until no point reassignment

k is an hyperparameter

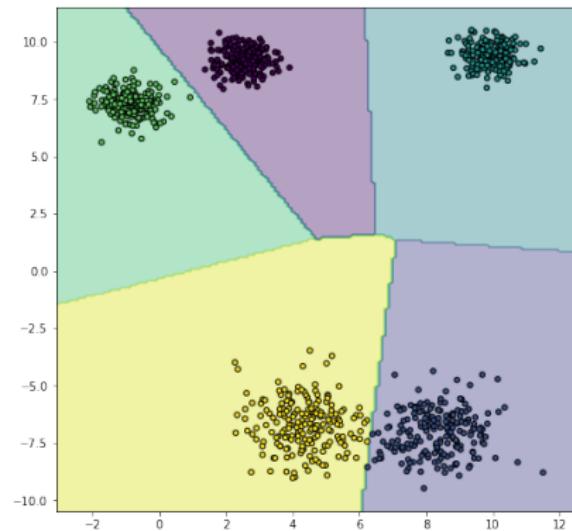
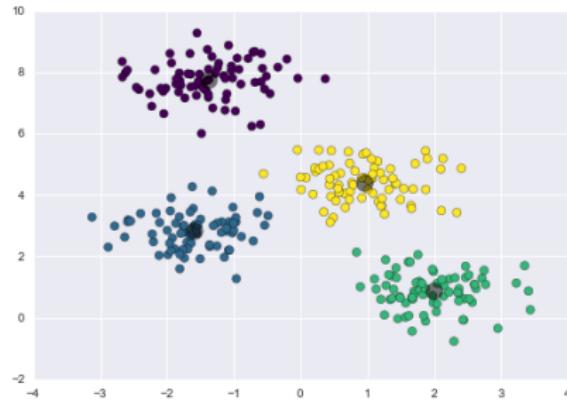
- Number of clusters



K-means

K-means algorithm (II)

New data points are assigned to its closest centroid

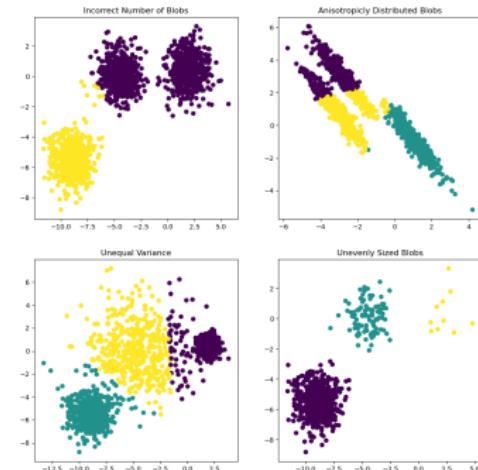


K-means

K-means limitations

K-means can fail in several conditions

- Incorrect number of clusters
- Different clusters variance
- Non-spheric clusters ⇒ normalization



(Source)

K-means

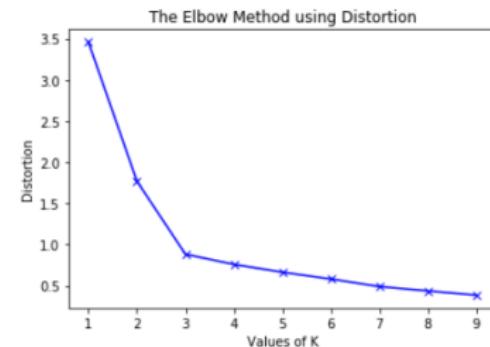
Elbow's method

Election of k

- Not a problem when domain information is available
- ... that is rarely the case

Elbow's method

1. Select $K = 1, \dots, n$
2. Visualize performance for each k
3. Choose K where metric stabilizes



Performance measures

- Inertia: mean squared error between each instance and its closest centroid
- Silhouette: $(b - a) / \max(a, b)$, where a mean intra-cluster distance, and b is the mean nearest-cluster distance

K-means

Application: Image segmentation



(Source)



(Source)

K-means

Application: Clustering for semi-supervised learning

Semi-supervised learning: Only a subset of the dataset is labeled

- Supervised and unsupervised learning
- Quite common in real-world applications (labels used to be expensive)

f_1	f_2	\dots	f_n	y
$a_{1,1}$	$a_{2,1}$	\dots	$a_{n,1}$	y_1
$a_{1,2}$	$a_{2,2}$	\dots	$a_{n,2}$	
$a_{1,3}$	$a_{2,3}$	\dots	$a_{n,3}$	
$a_{1,4}$	$a_{2,4}$	\dots	$a_{n,4}$	y_4
$a_{1,5}$	$a_{2,5}$	\dots	$a_{n,5}$	

Label propagation

1. Obtain k clusters
2. Get a representative instance of each cluster (**medoid**) measuring the distance to the centroid
3. Label the members of each cluster with its medoid's label

K-means

K-means: Summary

Hyperparameters	Advantages	Disadvantages
k	Fast Few hyperparameters Scalable	Simple shapes Determine k Random initialization

Other clustering algorithms

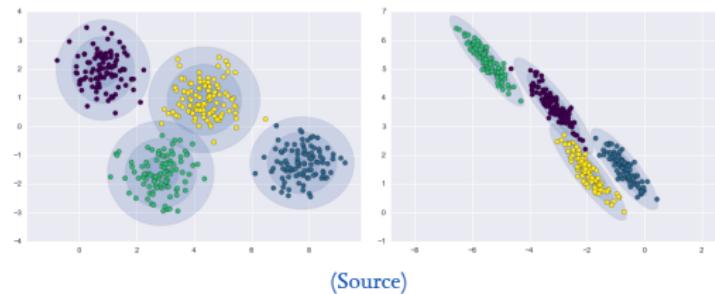
Gaussian Mixure Model (GMM) (I)

GMM is a generative clustering algorithm

- Assumes data coming from a set of multidimensional gaussian distributions

GMM fits a set $\{(\phi_i, \mu_i, \sigma_i)\}_{i=1,\dots,k}$

- ϕ is a weight
- μ is a multidimensional mean
- σ is a covariance matrix
- k is the number of clusters (hyperparameter)



Unsupervised learning

Other clustering algorithms

Gaussian Mixure Model (GMM) (II)

Gaussian parameters are fit with the Expectation-Maximization (E-M) algorithm

- E-M is a generalization of K-means

Expectation-Maximization algorithm

1. Init parameters randomly
2. Expectation step: Assign each instance to a cluster
 - Assignment is probabilistic
3. Maximization step: Update cluster parameters
 - Each cluster is updated using all the data
 - Instances contribution to a cluster parameters is weighted by the probability that it belongs to it
4. Go to 2

GMM can be seen as a fuzzy clustering algorithm

Other clustering algorithms

Gaussian Mixure Model (GMM) (III)

Gaussian parameters are fit with the Expectation-Maximization (E-M) algorithm

- E-M is a generalization of K-means

Expectation-Maximization algorithm

1. Init parameters randomly
2. Expectation step: Assign each instance to a cluster
 - Assignment is probabilistic
3. Maximization step: Update cluster parameters
 - Each cluster is updated using all the data
 - Instances contribution to a cluster parameters is weighted by the probability that it belongs to it
4. Go to 2

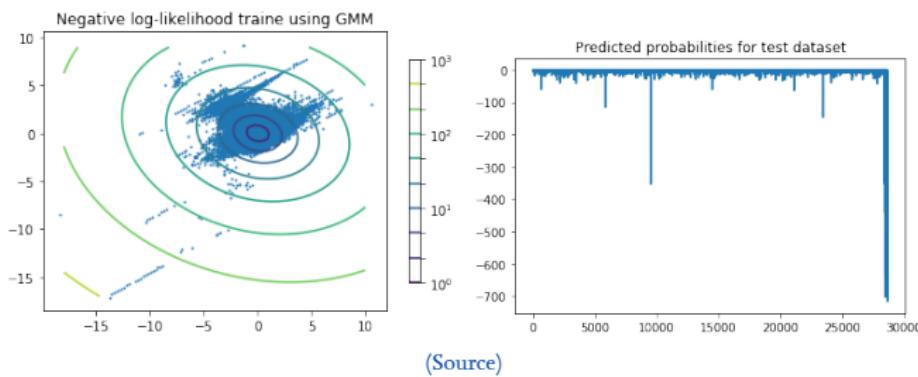
GMM can be seen as a fuzzy clustering algorithm

Other clustering algorithms

Gaussian Mixure Model (GMM) (IV)

GMM provides a probability of an instance to belong to a cluster

- This can be used to detect anomalies
- Just assign a probability threshold



Other clustering algorithms

GMM: Summary

Hyperparameters	Advantages	Disadvantages
Number of clusters	Probabilistic clustering	Number of clusters
Covariance matrix type	Generative model	Gaussian data
	Anomaly detection	Sensitive to outliers

Other clustering algorithms

DBSCAN (I)

DBSCAN: Density-Based Spatial Clustering of Applications with Noise

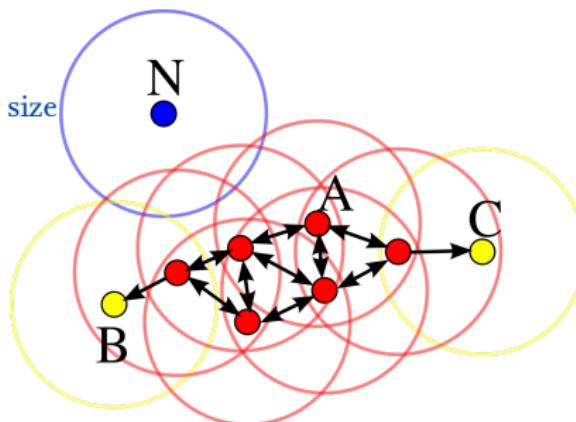
- Identifies high density regions (dense regions) in feature space
- Asumtion: Clusters form dense regions separated by empty areas

Hyperparameters

- ϵ : Radius of a neighborhood
- min_samples: Minumun cluster size

Type of points

- Core instance
- Outliers



(Source)

Other clustering algorithms

DBSCAN (II)

$\epsilon=0.05, \text{min_samples} = 5$



$\epsilon=0.2, \text{min_samples} = 5$



(Source)

Other clustering algorithms

DBSCAN: Summary

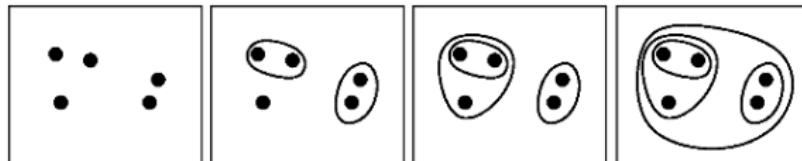
Hyperparameters	Advantages	Disadvantages
ϵ	No explicit number of clusters	Slower than K-means
min_samples	Scales relatively well Almost deterministic Robust to outliers Anomaly detection	Clusters with different densities

Other clustering algorithms

Agglomerative clustering (I)

Agglomerative clustering

1. Initially, each instance forms a cluster
2. Merge the two most similar clusters according to a metric
3. Repeat 2 until a stop criterion is satisfied



We need a similarity measure between two clusters

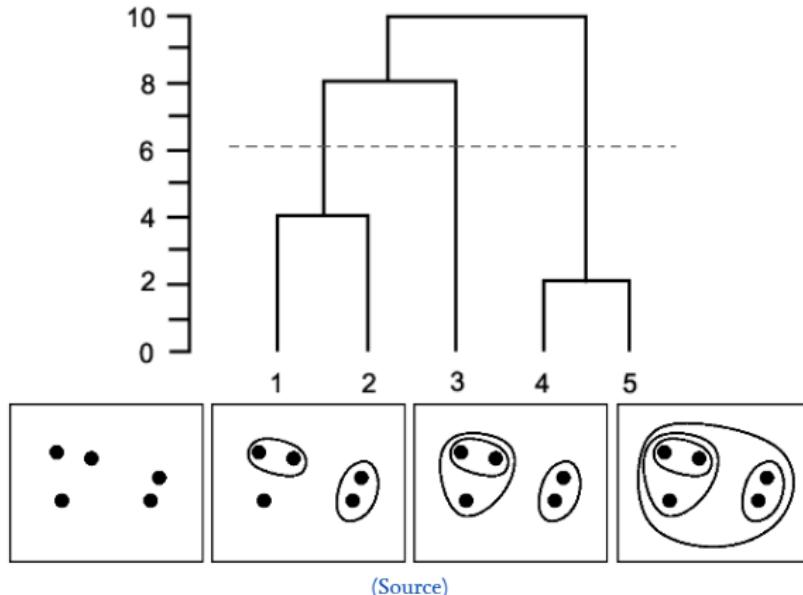
- **Ward:** Minimizes variance within merged clusters. Leads to equally sized clusters
- **Average:** Minimizes average distances between their points
- **Complete:** Minimizes maximum distance between their points

Other clustering algorithms

Agglomerative clustering (II)

Agglomerative clustering is a special case of hierarchical clustering

Dendrogram



Other clustering algorithms

Agglomerative clustering: Summary

Hyperparameters	Advantages	Disadvantages
	Complex shapes Hierarchical clustering	

Anomaly detection

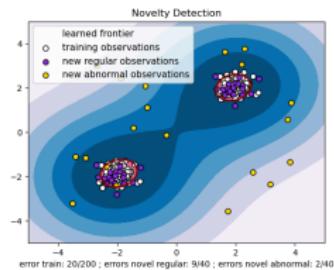
Two related concepts

- Outlayer detection and novelty detection

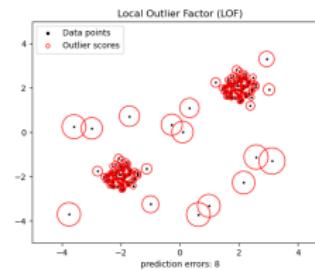
Adaptation of clustering and classification algorithms

- PCA, GMM, autoencoders, etc

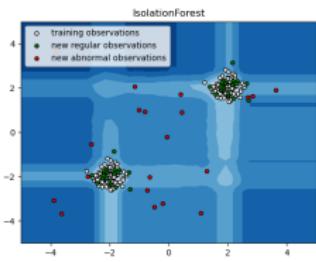
One-Class SVM



LOF



Isolation Forest



(Source)

Dimensionality reduction

PCA and manifold learning

Dimensionality reduction

Main approaches for dimensionality reduction (I)

Two main approaches to dimensionality reduction: Projection and manifold learning

Projection

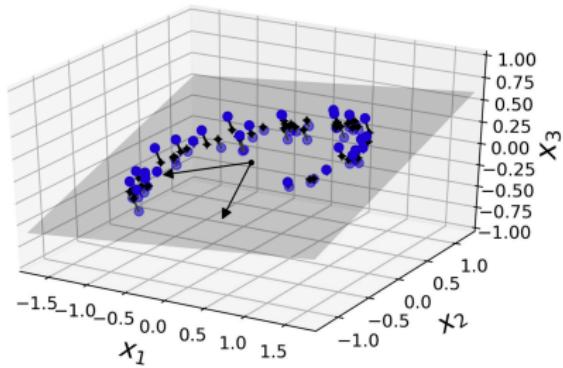


Figure 8-2. A 3D dataset lying close to a 2D subspace

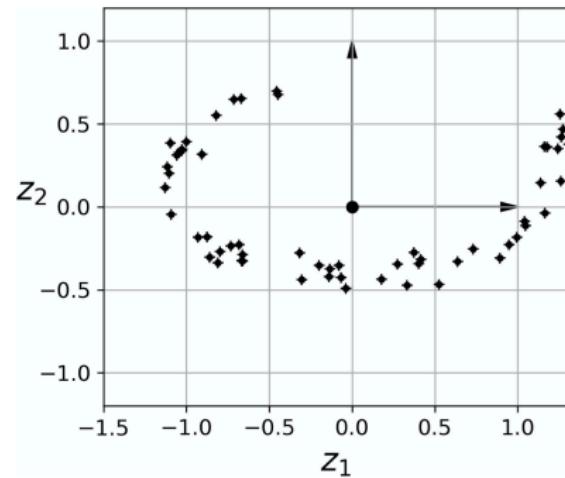


Figure 8-3. The new 2D dataset after projection

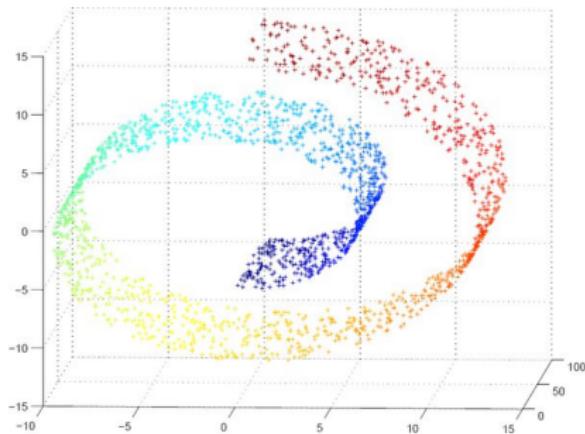
(Source)

Algorithms

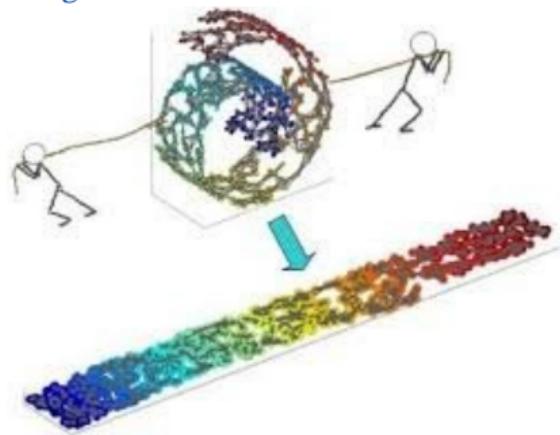
Dimensionality reduction

Main approaches for dimensionality reduction (II)

Manifold learning



(Source)



Manifold learning algorithms

- Isomap, T-distributed Stochastic Neighbor Embedding (t-SNE), Multi-dimensional Scaling (MDS), Locally Linear Embedding (LLE), ...

Dimensionality reduction

Principal Components Analysis (I)

Dimensionality reduction transforms data into more convenient representations

- Reduce data dimensionality
- Visualize multidimensional data

Main algorithms

Dimensionality reduction

Principal Components Analysis (I)

PCA create a new coordinate system

- New axes capture maximum variance and are orthogonal
 - They are named **principal components**
- The amount of variance captured by each principal component is captured
- PCA does not change the original dimensionality

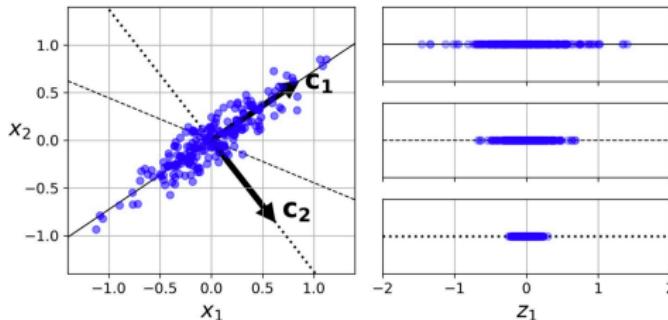
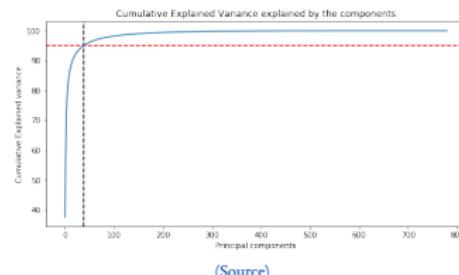


Figure 8-7. Selecting the subspace to project on

(Source)



(Source)

Dimensionality reduction

Principal Components Analysis (II)

PCA application: Image compression

Original image



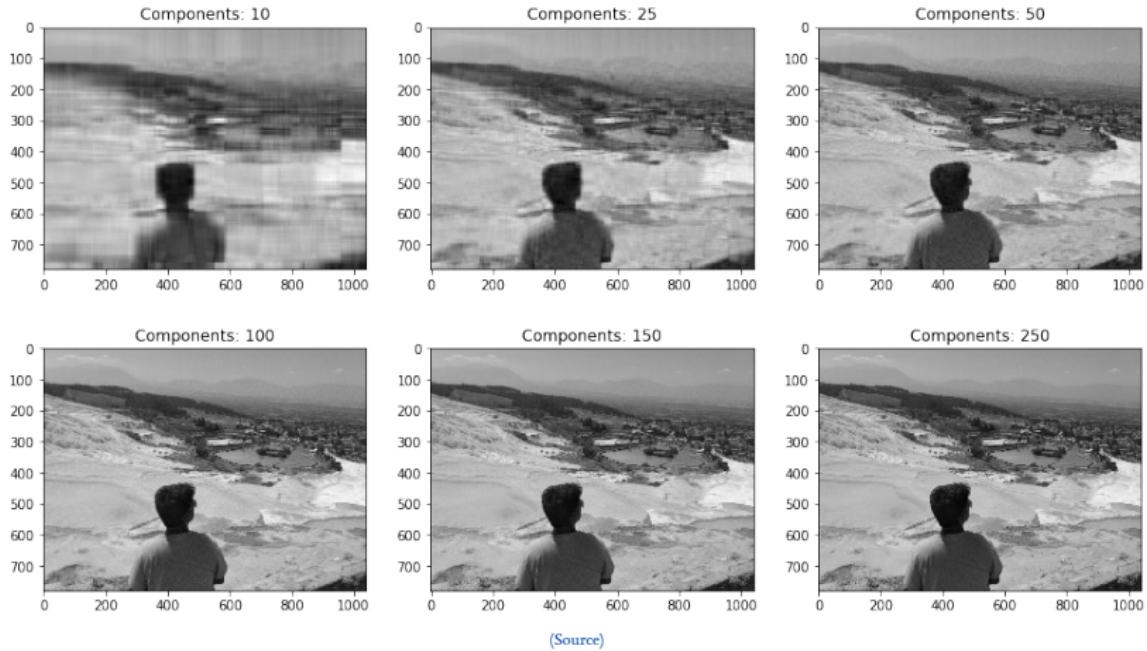
(Source)

Compressed image (38 dimensions)



Dimensionality reduction

Principal Components Analysis (III)



Dimensionality reduction

Principal Components Analysis (III)

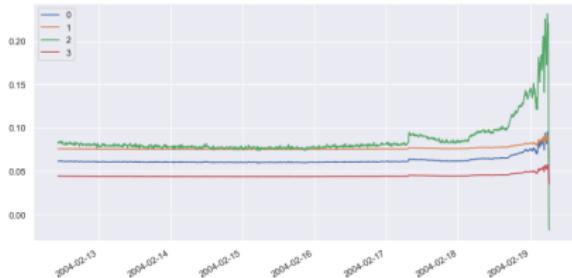
PCA application: Anomaly detection to predict bearing failure

- Vibrations of four bearings

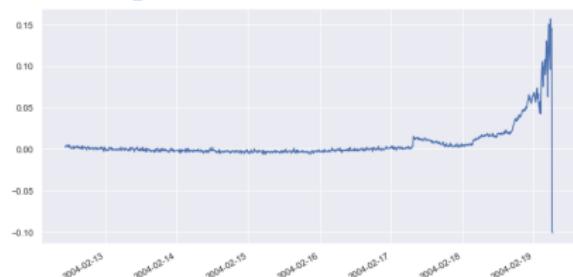
Vibration time series



Reconstructed time series



First component



Reconstruction error



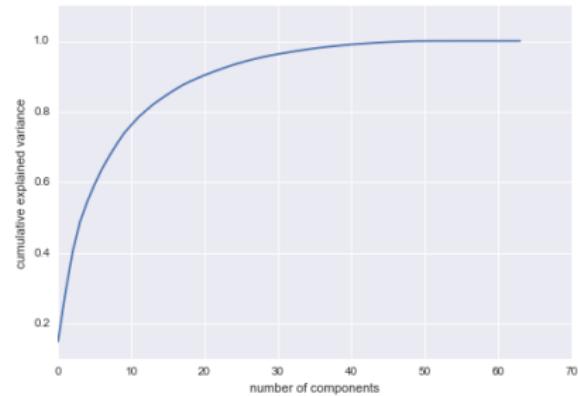
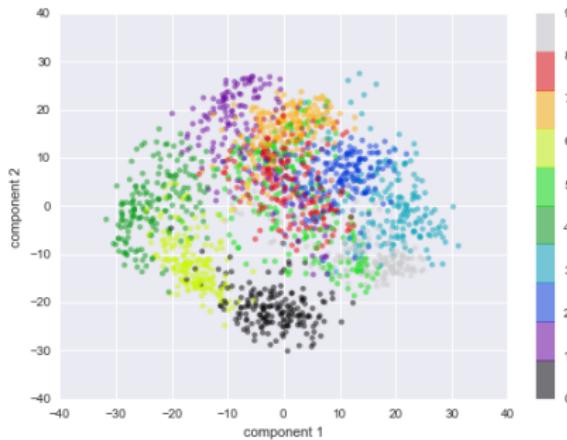
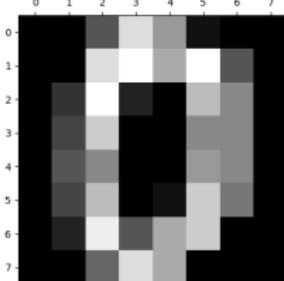
(Source)

Dimensionality reduction

Principal Components Analysis (IV)

Example: Hand-written digits recognition

- Images of hand-written digits
- 8x8 images (64 dimensions)
- 10 digits
- Classification problem



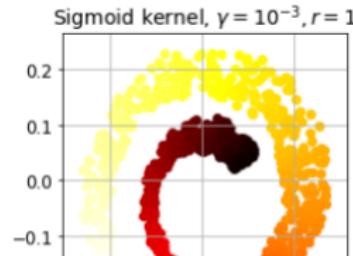
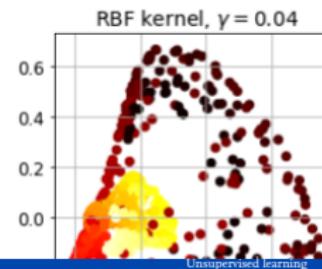
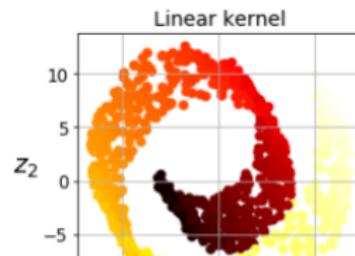
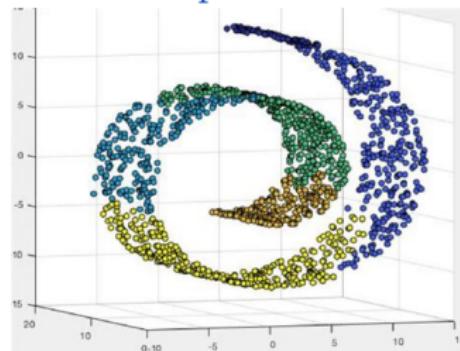
Unsupervised learning

Dimensionality reduction

Kernel PCA

The kernel trick applies to PCA

- kPCA captures non-linear structures



Dimensionality reduction

Manifold learning

TODO

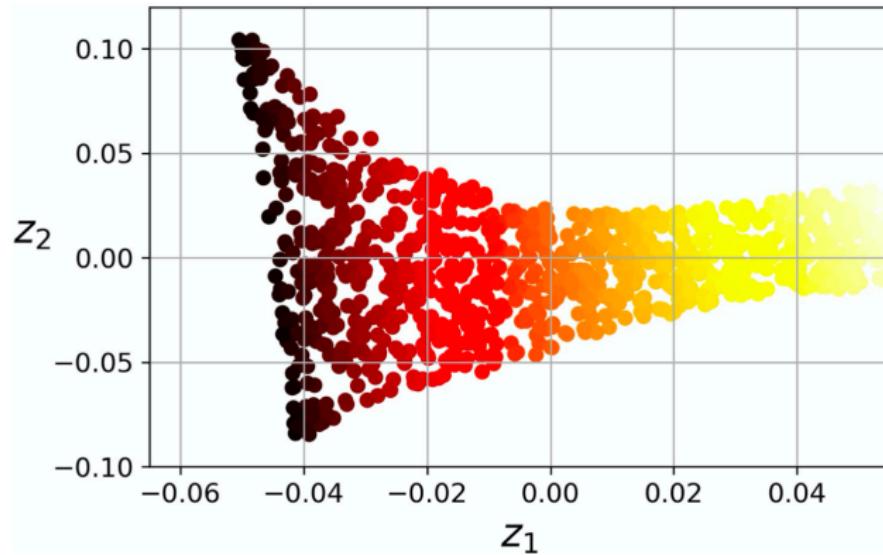


Figure 8-12. Unrolled Swiss roll using LLE

(Source)

Dimensionality reduction

Manifold learning

Measures how much each training instance linearly relates to its closest neighbors
preserves local relations

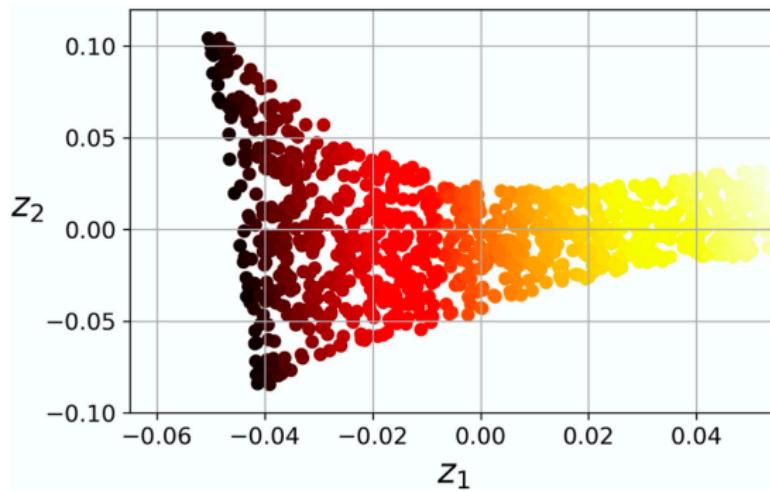


Figure 8-12. Unrolled Swiss roll using LLE

(Source)

Dimensionality reduction

Manifold learning

Multidimensional Scaling (MDS)

- Preserves distances

Isomap

- Preserves geodesic distance

t-Distributed Stochastic Neighbor Embedding (t-SNE)

- Preserves local distances and keep dissimilar instances apart

Manifold Learning with 1000 points, 10 neighbors

