Lecture 7

Machine Learning for Intelligent Systems

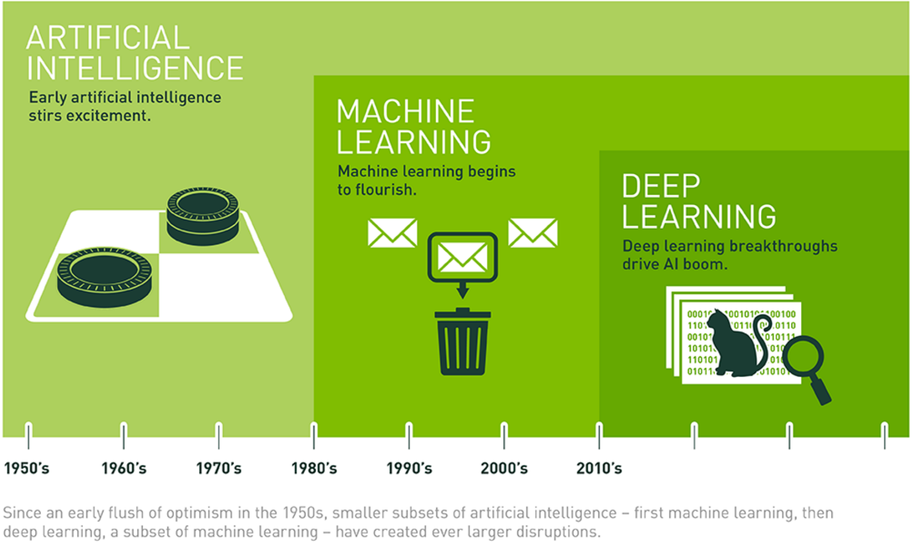
Introduction, Clustering, Classification, Regression, Evaluation

COMP 474/6741, Winter 2022

René Witte
Department of Computer Science
and Software Engineering
Concordia University

Outline

1. **Machine Learning Primer**
   - History
   - ML Types
   - Process
2. **Clustering Documents**
   - Motivation
   - k-Means Clustering
   - Application Example
3. **Classifications & Predictions**
   - Introduction
   - Classification with kNN
   - Regression with kNN
4. **Machine Learning Evaluation**
   - Evaluation Methodology
   - Evaluation Metrics
   - Error Analysis
   - Overfitting
   - Underfitting
   - Cross-Validation
5. **Notes and Further Reading**



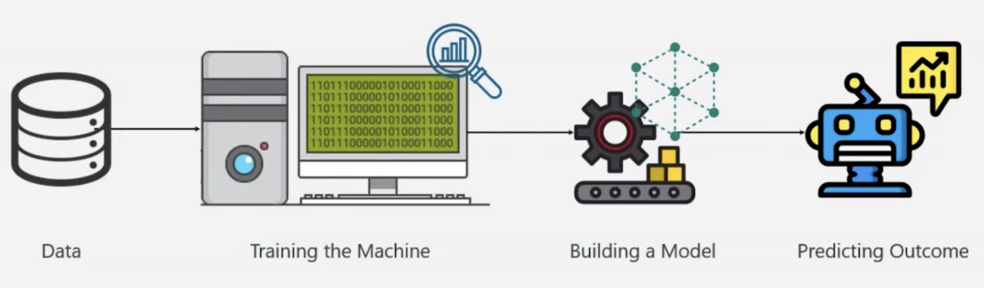
https://medium.com/machine-learning-for-humans/neural-networks-deep-learning-cdad8aeae49b

René Witte

Learn from experience

In 1959, Arthur Samuel first proposed the concept Machine Learning:

> *"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."*



René Witte

Inference

Process of deriving new facts from a set of premises

Types of logical inference

1. Deduction
2. Abduction
3. Induction

Deduction

aka Natural Deduction

- Conclusion follows necessary from the premises.
- From A $\Rightarrow$ B and A, we conclude that B
- We conclude from the general case to a specific example of the general case
- Example:
  1. All men are mortal.
  2. Socrates is a man.
  3. from 1 $\wedge$ 2 $\Rightarrow$ Socrates is mortal.
- Our subclass inference in RDFS also falls into this category.

René Witte

Concordia University

Abductive Reasoning

- Conclusion is one hypothetical (most probable) explanation for the premises
- From A $\Rightarrow$ B and B, we conclude A
- Example:
  1. Drunk people do not walk straight.
  2. John does not walk straight.
  3. from 1 $\wedge$ 2 $\Rightarrow$ John is drunk.
- Not sound... but may be most likely explanation for B
- Used in medicine...
  1. in reality: disease $\Rightarrow$ symptoms
  2. patient complains about some symptoms... doctor concludes a disease

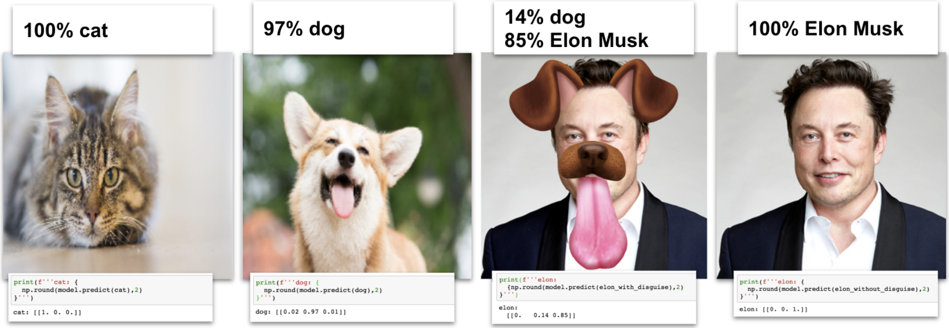
Inductive Reasoning

- Conclusion about all members of a class from the examination of only a few member of the class.
- From $A \wedge C \Rightarrow B$ and $A \wedge D \Rightarrow B$, we conclude $A \Rightarrow B$
- We construct a general explanation based on specific cases
- Example:
  1. All CS students in COMP 474 are smart.
  2. All CS students on vacation are smart.
  3. from 1 $\wedge$ 2 $\Rightarrow$ All CS students are smart.
- Not sound
- But, can be seen as hypothesis construction or generalisation

Inductive Learning

Learning from examples

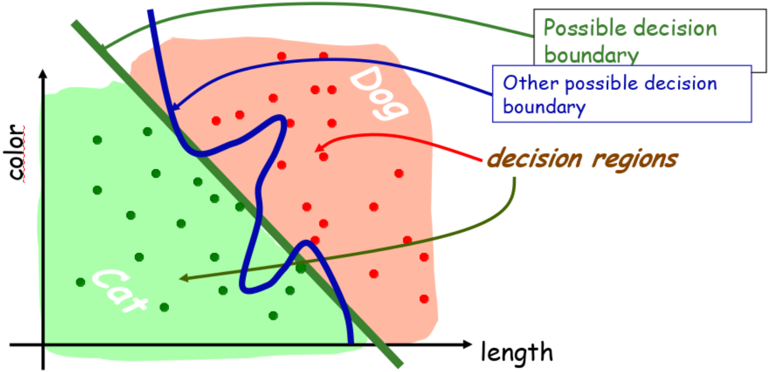
- Most work in ML
- Examples are given (positive and/or negative) to train a system in a classification (or regression) task
- Extrapolate from the training set to make accurate predictions about future examples
- Given a new instance X you have never seen, you must find an estimate of the function f(X) where f(X) is the desired output



From datascience.com, https://towardsdatascience.com/cat- dog- or- elon- musk- 145658489730

Example

- Given pairs $(X, f(X))$ (the training set – the data points)
- Find a function f that fits the training set well
- So that given a new X, you can predict its $f(X)$ value



Note: choosing one function over another beyond just looking at the training set is called inductive bias (eg. prefer "smoother" functions)

René Witte

Feature Vectors

- Input data are represented by a vector of features, X
- Each vector X is a list of (attribute, value) pairs.
- Ex: $X = $ `[nose:big, teeth:big, eyes:big, moustache:no]`
- The number of attributes is fixed (positive, finite)
- Each attribute has a fixed, finite number of possible values
- Each example can be interpreted as a point in a *n*-dimensional feature space, where *n* is the number of attributes (features)

Some Machine Learning Techniques

Probabilistic Methods

- e.g., Naïve Bayes Classifier

Decision Trees

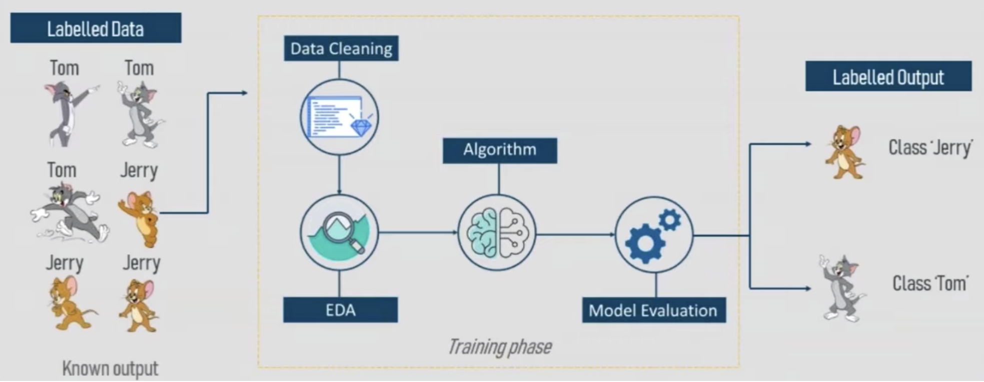
- Use only discriminating features as questions in a big *if-then-else* tree

Neural Networks

- Also called parallel distributed processing or connectionist systems
- Intelligence arise from having a large number of simple computational units

NB: Deep Learning $\approx$ Neural Networks “on steroids”

Supervised Learning

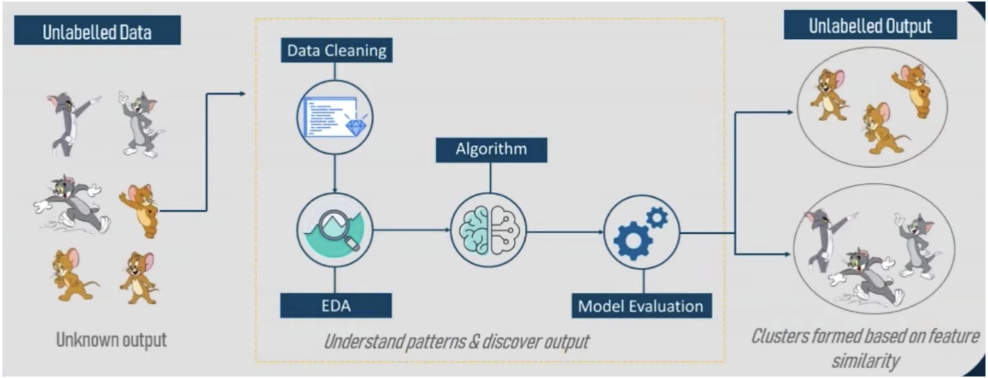


Labeled Data

In Supervised Learning, we train a system using data with known labels.

Unsupervised Learning

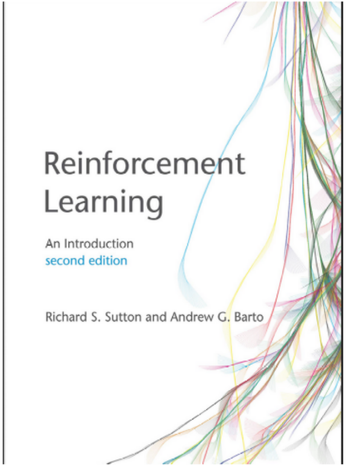
René Witte
Concordia University

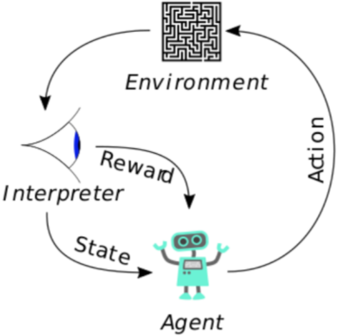


Unlabeled Data

In Unsupervised Learning, we have only unlabeled data and train a system without guidance from an expected output.

Reinforcement Learning





The typical RL scenario: an agent takes actions in an environment, which is interpreted into a reward and a representation of the state, which are fed back into the agent.

AI Learns to Park

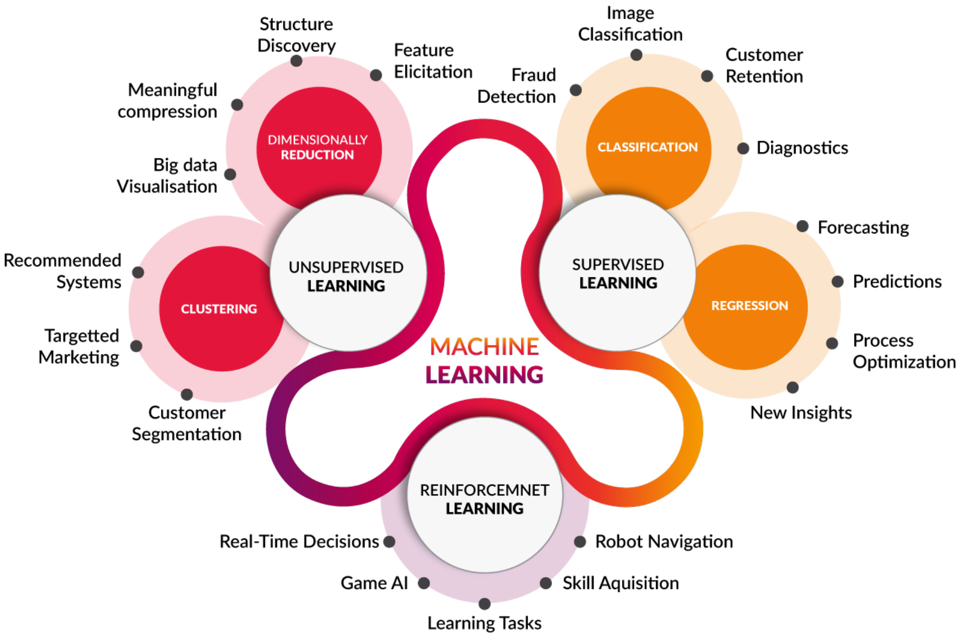
https://www.youtube.com/watch?v=VMp6pq6_QjI

Machine Learning Categories

René Witte

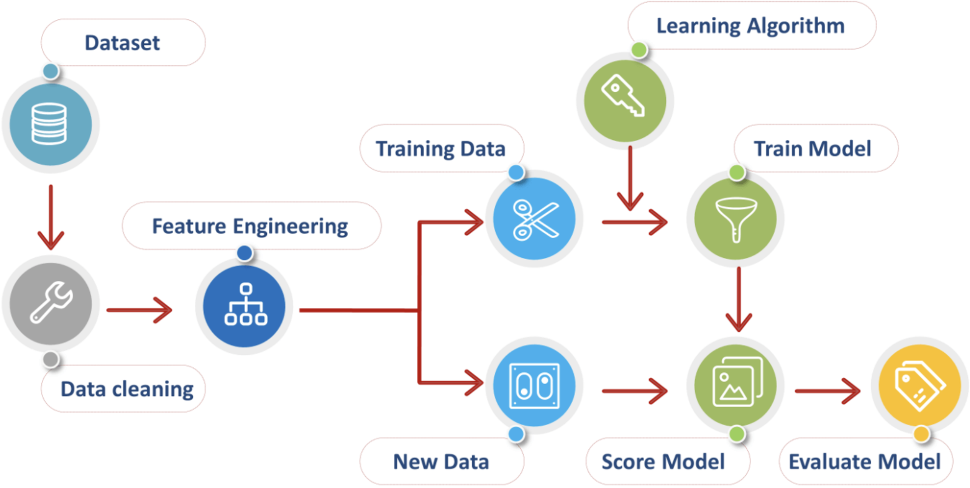
Concordia University

	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Definition	The machine learns by using labelled data	The machine is trained on unlabeled data without any guidance	An agent interacts with its environment by producing actions & discovers errors and rewards
Types of problems	Regression &Classification	Association & Clustering	Reward based
Type of data	Labelled data	Unlabelled data	No pre-defined data
Training	External supervision	No supervision	No supervision
Approach	Map labelled input to known output	Understand patterns and discover output	Follow trail and error method
Popular Algorithms	Linear Regression, Logistic Regression, KNN, etc	K-means, C-means, etc	Q-learning, etc



http://www.cognub.com/index.php/cognitive-platform/

General machine learning process



→ **Worksheet #6: Task 1**

René Witte

Outline

1. Machine Learning Primer
2. **Clustering Documents**
   - Motivation
   - k-Means Clustering
   - Application Example
3. Classifications & Predictions
4. Machine Learning Evaluation
5. Notes and Further Reading

Math with Words

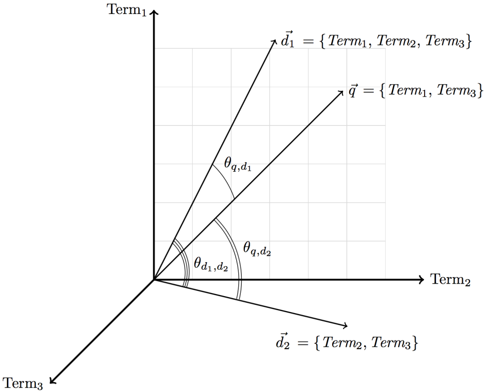
Vector Space Model

- A mathematical model to portray an n-dimensional space
- Entities are described by vectors with n coordinates in a real space $\mathbb{R}^n$
- Given two vectors, we can compute a similarity coefficient between them
- Cosine of the angle between two vectors reflects their degree of similarity

$$\mathrm{tf} = 1 + \log(\mathrm{tf}_{t,d}) \tag{1}$$

$$\mathrm{idf} = \log \frac{N}{\mathrm{df}_t} \tag{2}$$

$$cos(\vec{q}\,, \vec{d}\,) = \frac{\sum_{i=1}^{|v|} q_i \cdot d_i}{\sqrt{\sum_{i=1}^{|v|} {q_i}^2} \cdot \sqrt{\sum_{i=1}^{|v|} {d_i}^2}} \tag{3}$$



Motivation

Intelligent Systems for Investigative Journalism

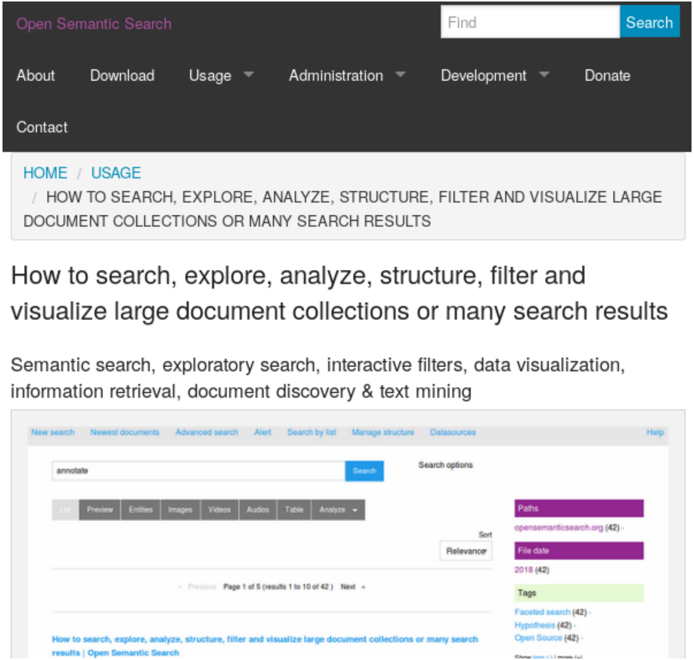
Organize large, unstructured document collections:

- Enron email dataset – ca. 500,000 emails from management
- Wikileaks – often releases millions of documents
  - Guantanamo Bay Files, TPP Agreements, CIA Documents, German BND-NSA Inquiry, . . .
- Facebook internal documents leaks (Cambridge Analytica scandal, 7000 documents)
- Luanda Leaks (715,000 emails, charts, contracts, audits, etc.)
- Paradise Papers (13.4 million confidential papers regarding offshore investments)

Canada Revenue Agency launches 100 audits after Paradise Papers leak

By **Alex Boutilier** Ottawa Bureau
Tues., Jan. 29, 2019 | 2 min. read

https://www.thestar.com/news/paradise-papers/2019/01/29/canada-revenue-agency-launches-100-audits-after-paradise-papers-leak.html

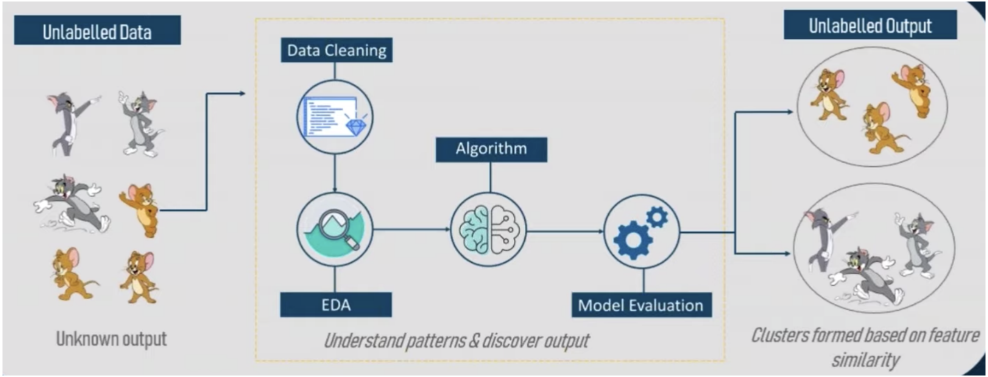


https://opensemanticsearch.org

René Witte

Unsupervised Learning

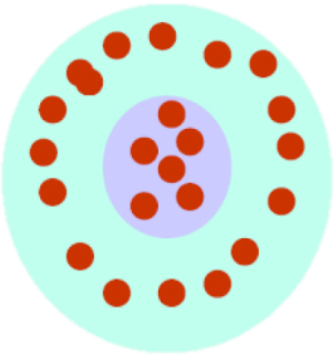
- Remember, we do not “classify” documents (like in “spam vs. ham”)
- Rather, we group similar documents together
- Often used as a first exploratory step in data analysis
  - Data points (here: documents) in individual clusters can be further analyzed, possibly with different methods

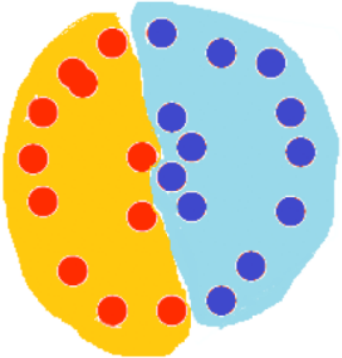


What are Clusters?

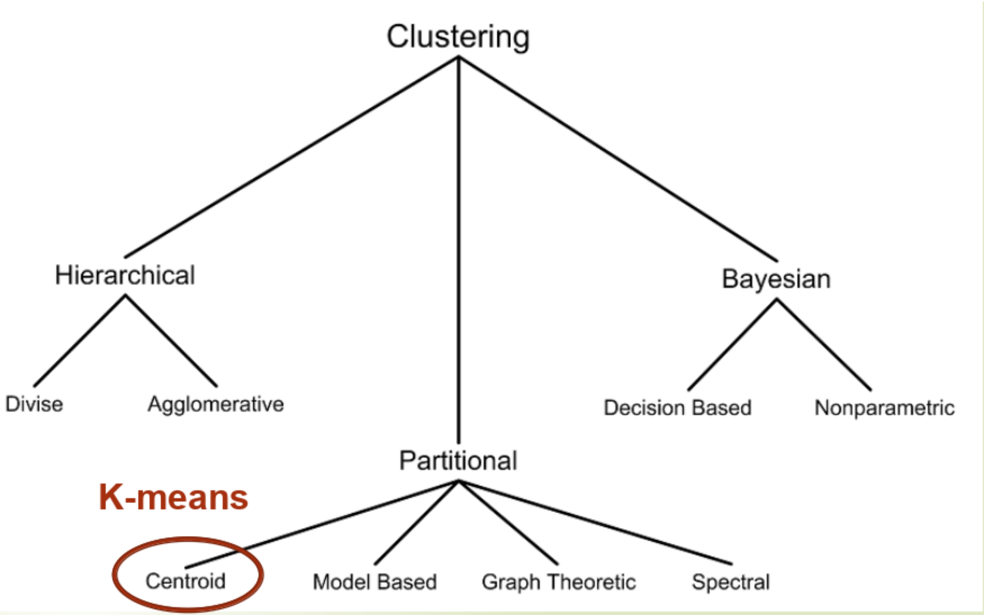
Clustering

- The organization of unlabeled data into similarity groups, called clusters
- A cluster is a collection of data items which are "similar" between them, and "dissimilar" to data items in other clusters.
- Generally, there is no right or wrong answer to what the clusters in a dataset are.





Clustering Techniques



Partition-based Clustering

K-means (MacQueen, 1967) is a partitional clustering algorithm:

- Given m vectors in an n-dimensional space, $\vec{x}_1, \ldots, \vec{x}_m \in \mathbb{R}^n$
- User defines k, the number of clusters

Algorithm

1. Pick k points from the dataset (usually at random).
   These points represent our initial group centroïds.
2. Assign each data point $\vec{x}_i$ to the nearest centroïd.
3. When all data points have been assigned, recalculate the positions of the k centroïds as the average of the cluster.
4. Repeat Steps 2 and 3 until none of the data instances change group (or changes stay below a given convergence limit Δ).

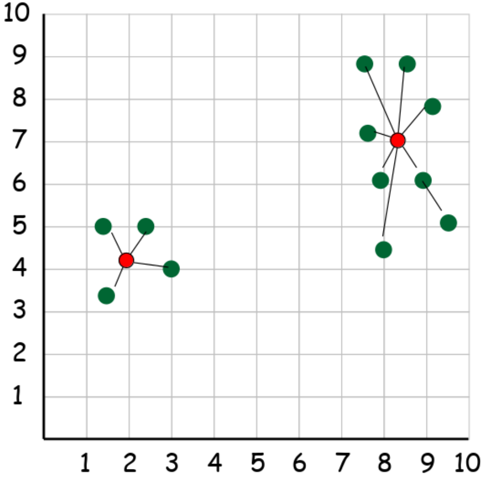
Euclidian Distance

To find the nearest centroïd:

- a possible metric is the Euclidean distance
- distance d between 2 points p, q

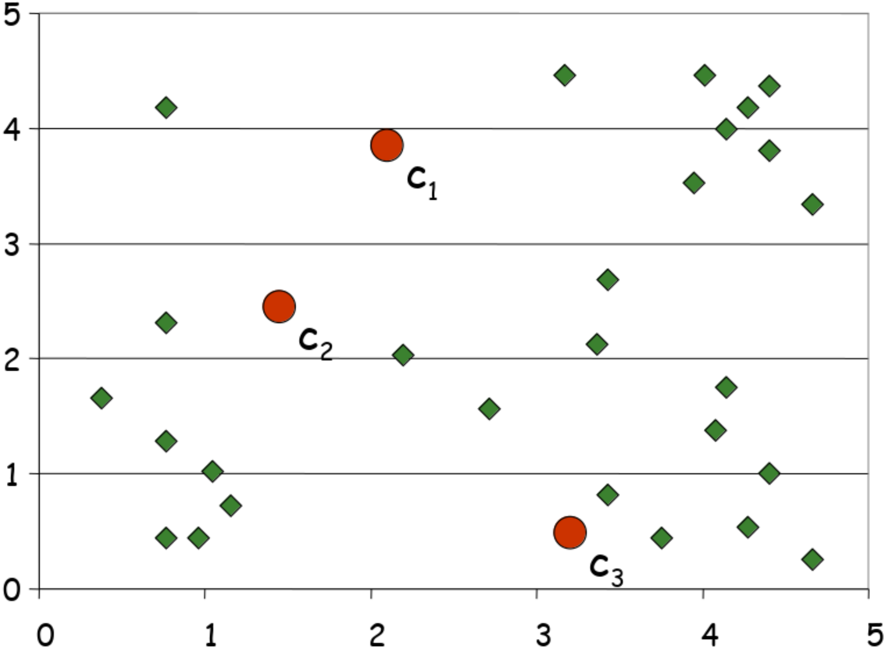
$$p = (p_1, p_2, \dots, p_n)$$
$$q = (q_1, q_2, \dots, q_n)$$
$$d = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

- where to assign a data point $\vec{x}$?
- $\rightarrow$ for all k clusters, choose the one where $\vec{x}$ has the smallest distance



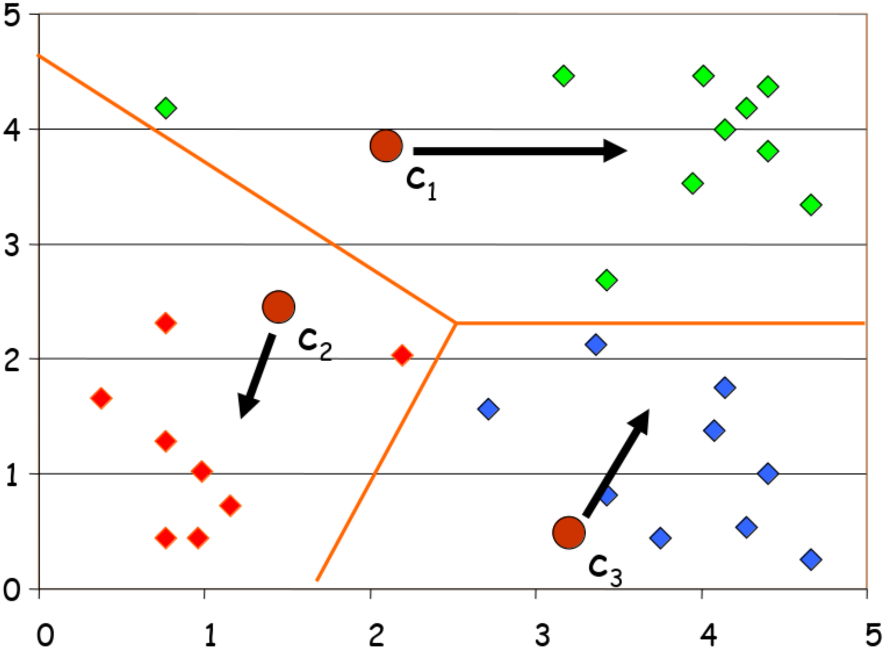
Example (1/5)

2D-vectors, k=3: Initialize random centroïds



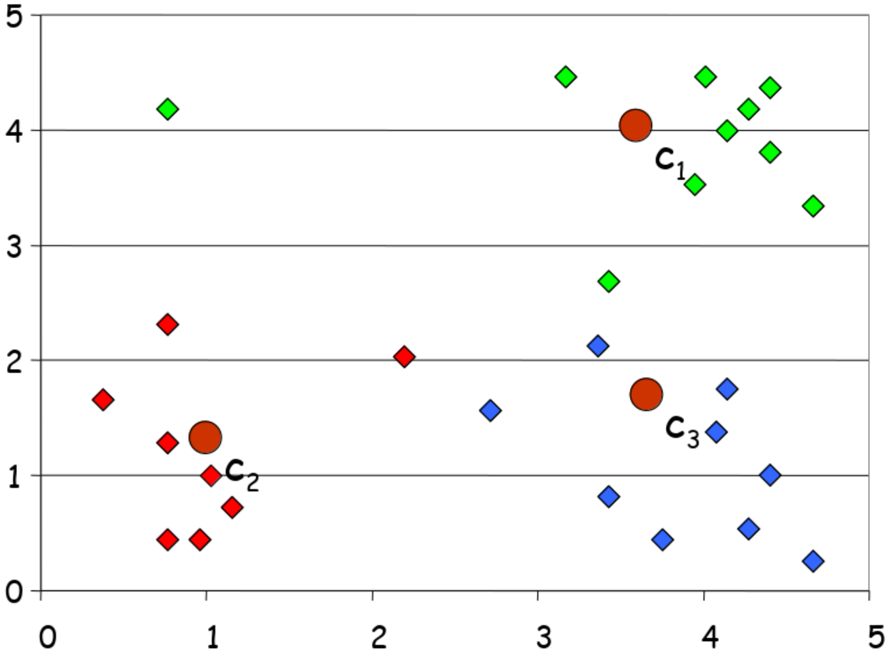
Example (2/5)

Partition data points to closest centroïds

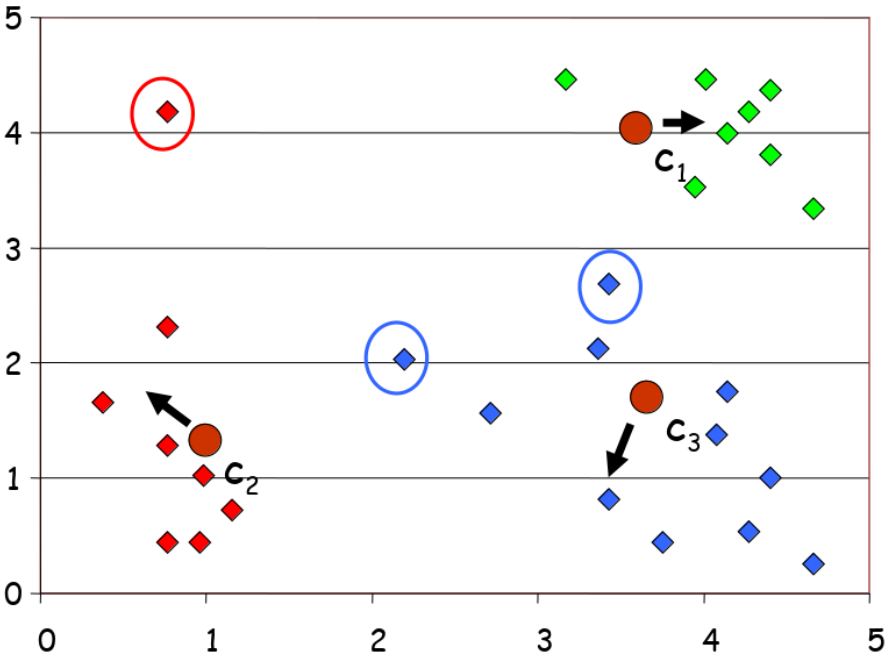


Example (3/5)

Compute new centroïds

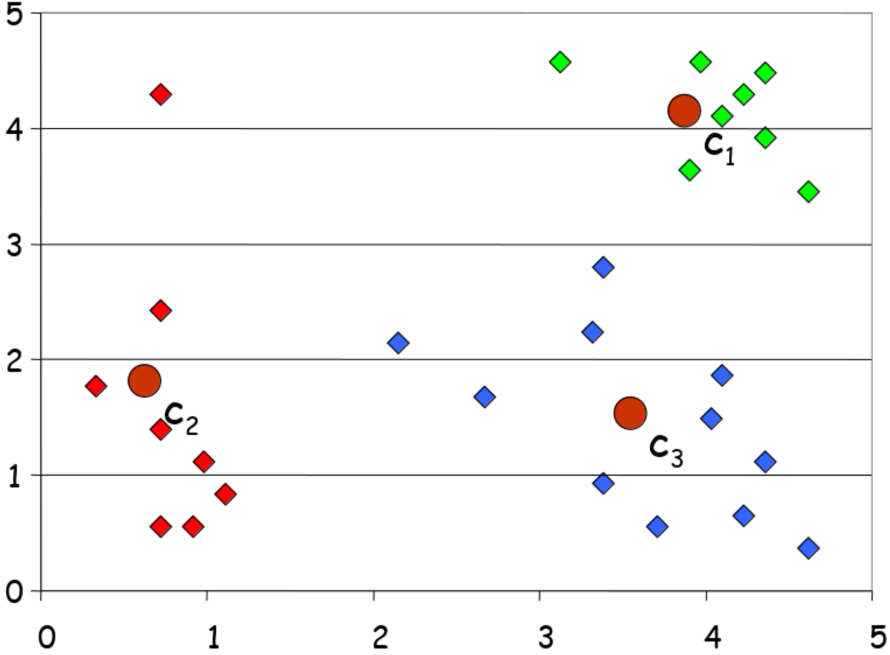


Example (4/5)
Re-assign data points to closest new centroïds



Example (5/5)

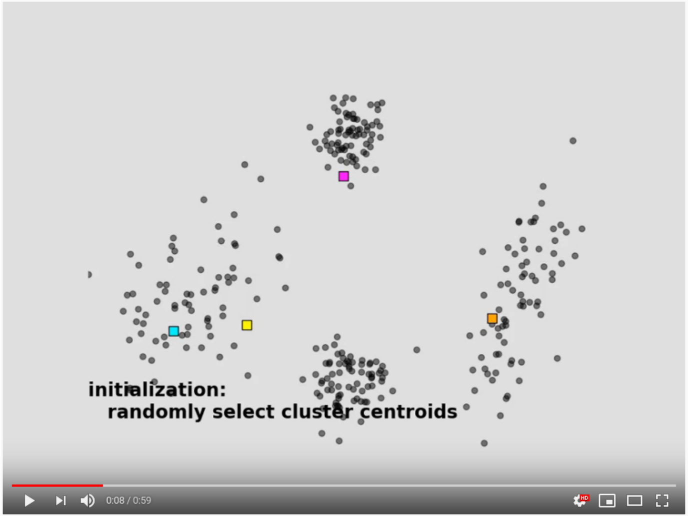
Repeat until clusters stabilize



René Witte

Concordia University

k-Means Clustering Illustrated



k-means clustering

https://www.youtube.com/watch?v=5I3Ei69I40s

→ Worksheet #6: Task 2

René Witte

Concordia University

k-Means: Pros & Cons

René Witte

Concordia University

Pros

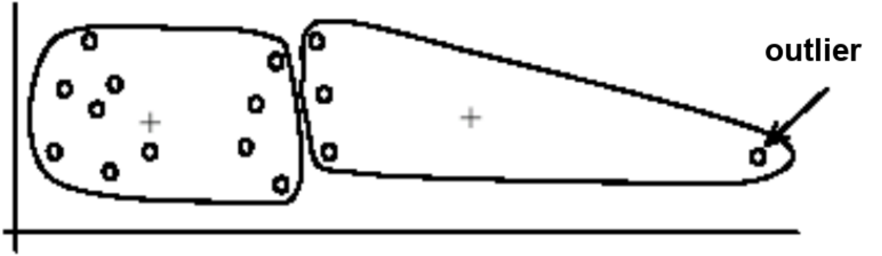
- Simple, easy to understand and implement
- Converges very fast
- Efficient: Time complexity $O(t \cdot k \cdot n)$, with
  - n number of data points
  - k number of clusters
  - t number of iterations

  $\rightarrow$ considered linear for practical purposes

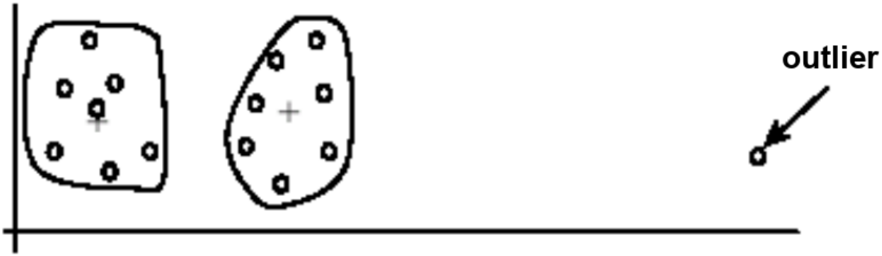
Cons

- User needs to choose k (usually not known)
- Sensitive to outliers
- Different results on same dataset, based on initial (random) centroïds

k-Means & Outliers

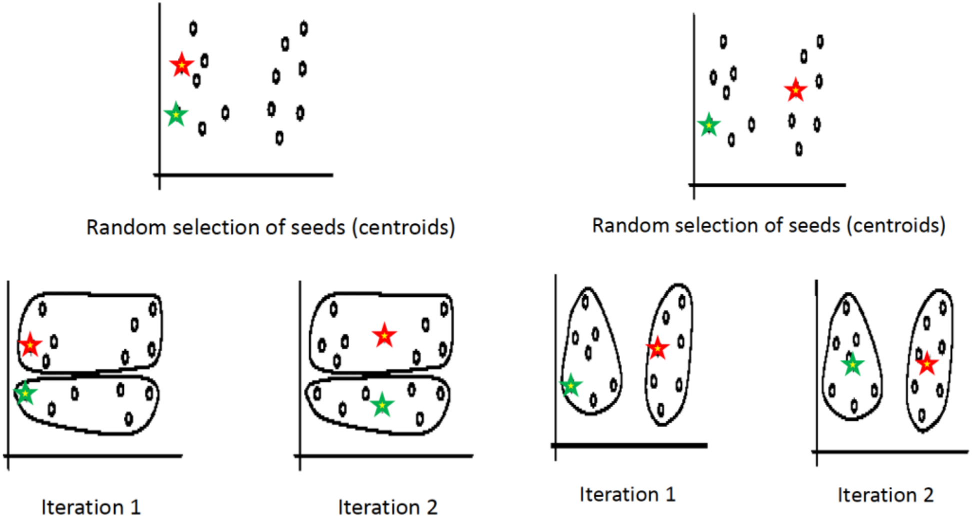


(A) Undesirable clusters



(B) Ideal clusters

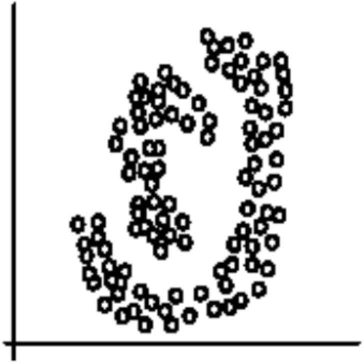
k-Means: Sensitivity to Initial Seeds

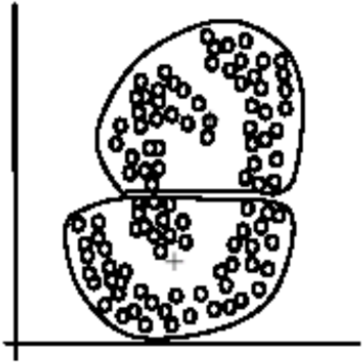


René Witte

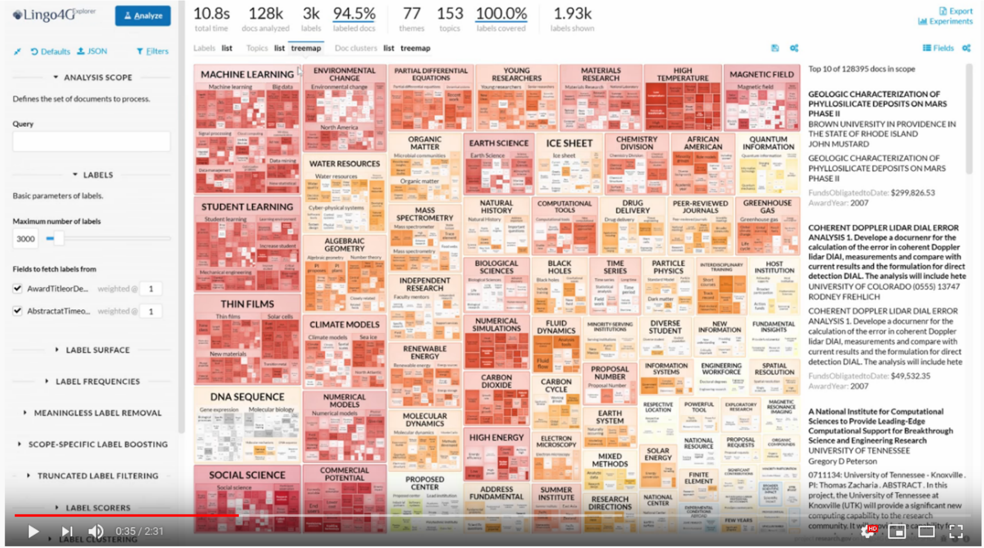
Summary

- Despite weaknesses, k-means is still one of the most popular algorithms, due to its simplicity and efficiency
- No clear evidence that any other clustering algorithm performs better in general
- Comparing different clustering algorithms is a difficult task: No one knows the correct clusters!





Document Clustering Example: Analyzing NSF Research Grants



Analyzing US National Science Foundation research proposals using Lingo4G text clustering engine

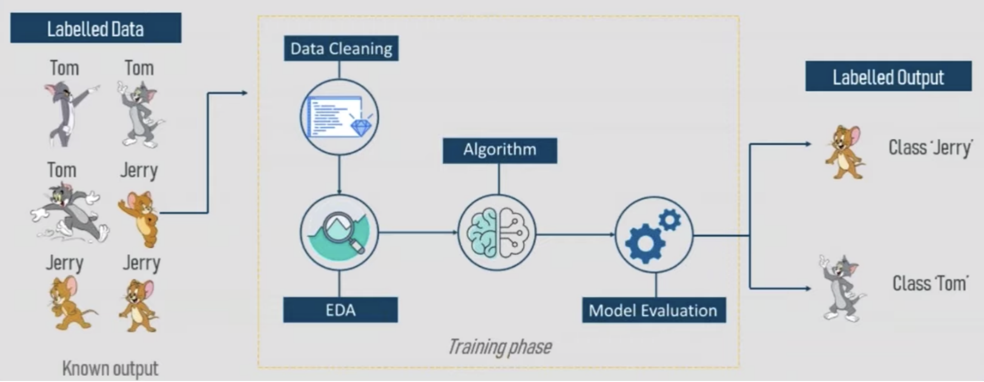
https://www.youtube.com/watch?v=85fZcK5EpnA

René Witte

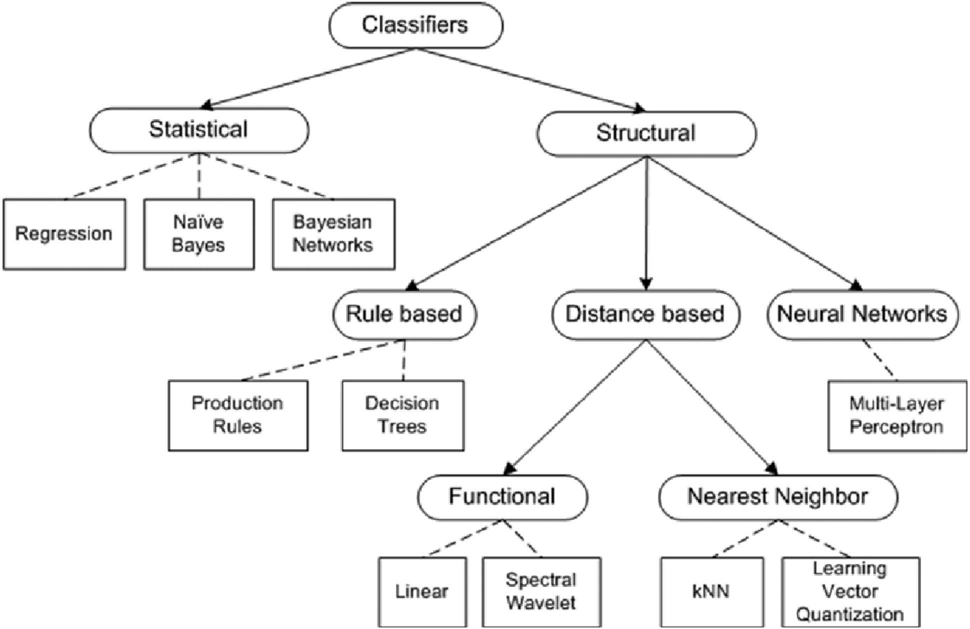
Outline

1. Machine Learning Primer
2. Clustering Documents
3. **Classifications & Predictions**
   - Introduction
   - Classification with kNN
   - Regression with kNN
4. Machine Learning Evaluation
5. Notes and Further Reading

Classification of Data



Classification Algorithms



René Witte

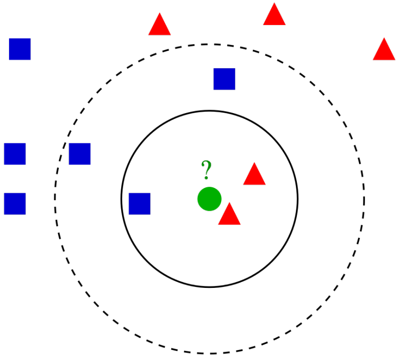
k-Nearest-Neighbor (kNN) Classification

kNN Algorithm

Training: only store feature vectors + class labels

Testing: Find the k data points nearest (e.g., Euclidian distance) to the new value. Resulting class is decided by majority vote.

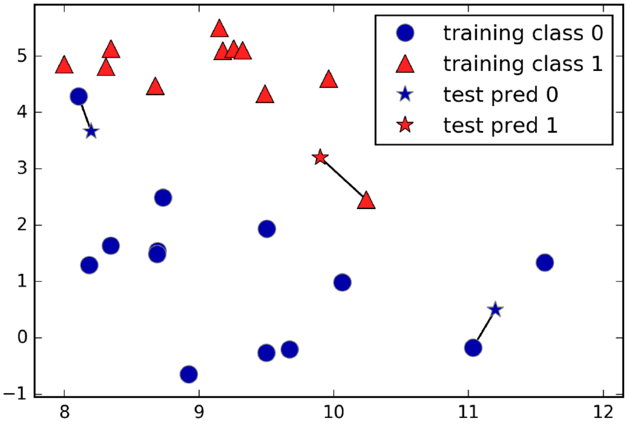
Note: in this simple form, kNN has no training effort, but large testing effort (so-called lazy learning)



kNN Classification

With $k = 1$

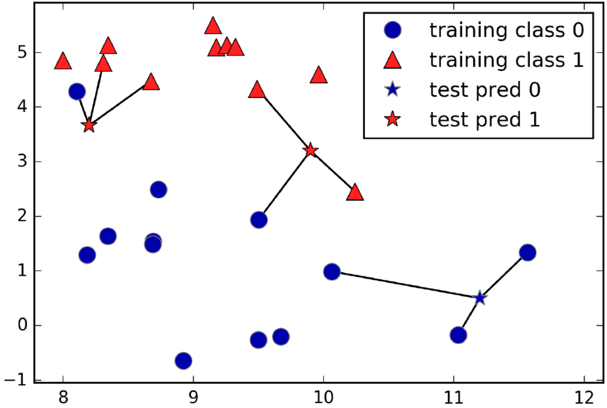
- Compute the distance of the unknown sample to all existing samples
- Assign the class of the *closest* neighbor to the new sample
  - Distance can be computed with different metrics, e.g., Euclidean distance or Manhattan distance



kNN Classification: General case

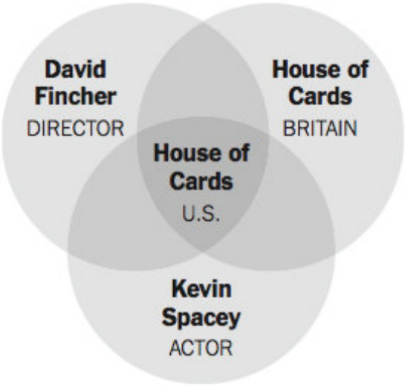
With arbitrary *k*

- kNN classification becomes a voting algorithm
- assign the same class as the majority of the *k* closest neighbors to the new sample
- Choice of *k* is dependent on data set



Netflix: Predict Success of Original Content

In 2013, Netflix decided to commission two seasons of the U.S. remake of the British series *House of Cards* based on an analysis of its customers' data

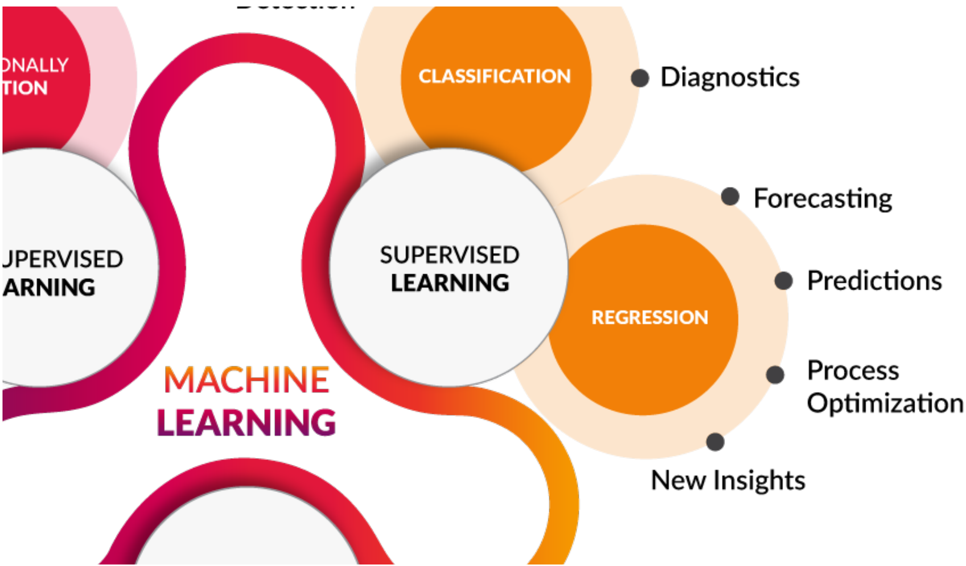


https://informationstrategyrsm.wordpress.com/2014/10/19/
big-data-analytics-house-of-cards-and-future-of-television-creation-consumption/

→ Worksheet #6: Task 3

Regression

Forecasting or predicting a value: e.g., house price, movie rating, temperature at noon, ...

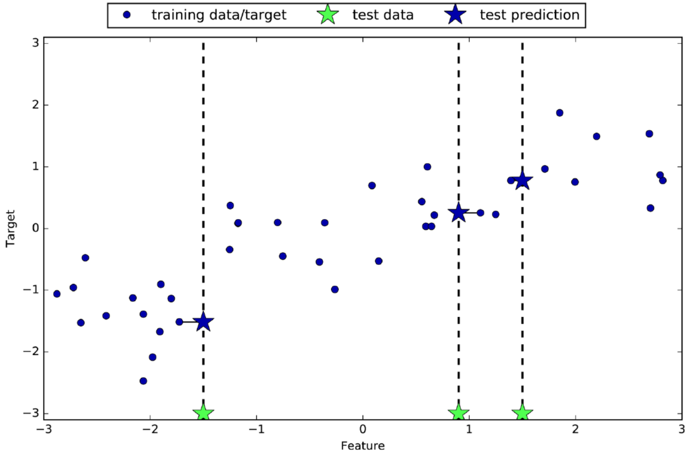


http://www.cognub.com/index.php/cognitive-platform/

kNN Regression

With $k = 1$

- Find the nearest existing data point to a new sample as before
- Assign the value of this point (e.g., *price, rating, ...*) to the new instance
  - Note: given n-dimensional vectors, we are using $n - 1$ dimensions for the similarity and the final for the predicted value

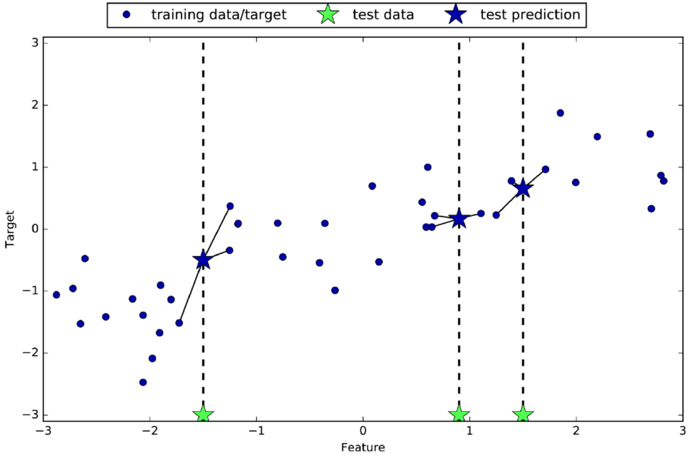


kNN Regression: General Case

Find the *k* nearest existing data points

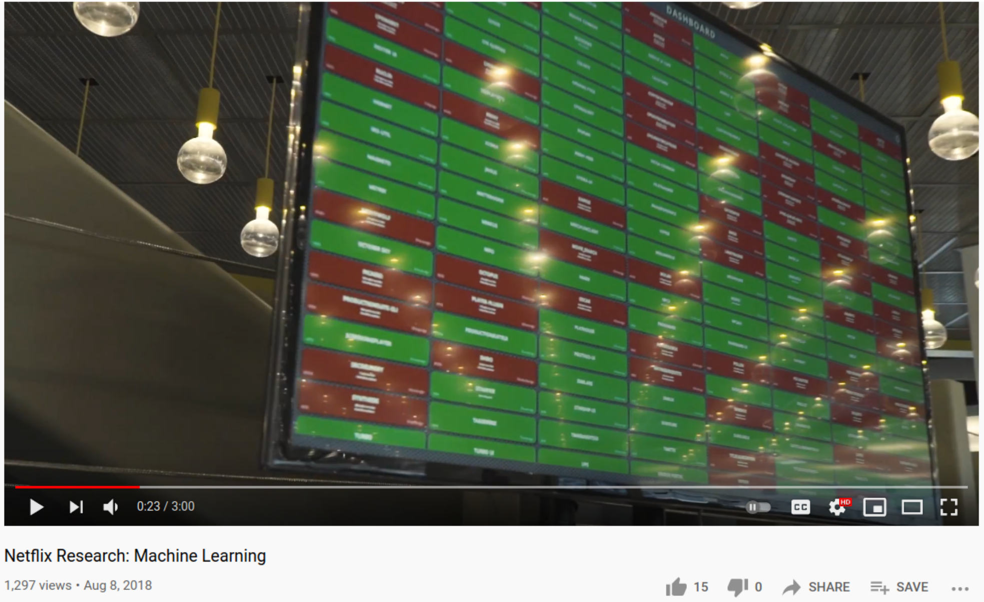
Assign the average of their values to the new point

- Note that this algorithm cannot extrapolate



→ Worksheet #6: Task 4

Machine Learning at Netflix



https://www.youtube.com/watch?v=X9ZES-fsxgU

René Witte

Concordia University

Outline

1. Machine Learning Primer
2. Clustering Documents
3. Classifications & Predictions
4. **Machine Learning Evaluation**
   - Evaluation Methodology
   - Evaluation Metrics
   - Error Analysis
   - Overfitting
   - Underfitting
   - Cross-Validation
5. Notes and Further Reading

Evaluation of a ML Model

Methodology

- How do you know if what you learned is correct?
- You run your classifier on a data set of unseen examples (that you did not use for training) for which you know the correct classification ("gold standard")

Training vs. testing data

- Split data into training (80%) and testing (20%) sets
- Depending on ML algorithm, the training set can be further split into:
  - Actual training set (80%)
  - Validation set (20%)

Standard Methodology

1. Collect a large set of examples (all with correct classifications)
2. Divide collection into training, validation and test set
3. Apply learning algorithm to training set to learn the parameters
4. Measure performance with the validation set, and adjust *hyper-parameters* to improve performance
5. Performance not good enough? $\Rightarrow$ 3
6. Measure performance with the test set

DO NOT LOOK AT THE TEST SET
until you arrived at Step 6.

Parameters
Basic values learned by the ML model, e.g.:

- for NB: prior & conditional probabilities
- for DTs: features to split
- for ANNs: weights

Hyper-Parameters
Parameters used to set up the ML model, e.g.:

- for NB: value of delta for smoothing
- for DTs: pruning level
- for ANNs: # of hidden layers, # of nodes per layer. . .

Metrics

Accuracy

- % of instances of the test set the algorithm correctly classifies
- when all classes are equally important and represented

Recall & Precision

- when one class is more important than the others

F-Measure

- Combined Precision & Recall (harmonic mean)

René Witte

Concordia University

Evaluation of Classifiers

What kind of errors can we make?

		Reality says...	
		Positive	**Negative**
Model predicts...	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

This is a so-called (binary) confusion matrix

Error Types

- False positive classification: Type I error
  (*"convict the innocent!"*)
- False negative classification: Type II error
  (*"free the guilty!"*)

Important realization: not all errors are created equal!

Voltaire: *"It is better to risk saving a guilty man than to condemn an innocent one."*

Evaluation Metrics

Commonly used

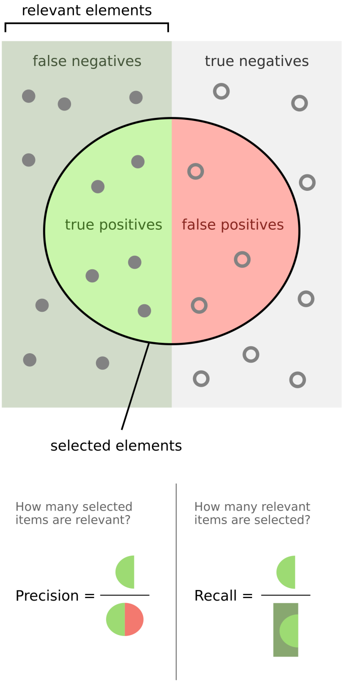
- Accuracy $= (TP + TN)/(TP + TN + FP + FN)$
- Recall $= TP/(TP + FN)$
- Precision $= TP/(TP + FP)$
- F_1-score $= \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$ (harmonic mean)

Mind the evaluation task

Precision, recall etc. are defined slightly differently for:

- Information retrieval tasks
- Classification tasks
- Ranked retrieval tasks
- Information extraction tasks

$\rightarrow$ Worksheet #6: Task 5

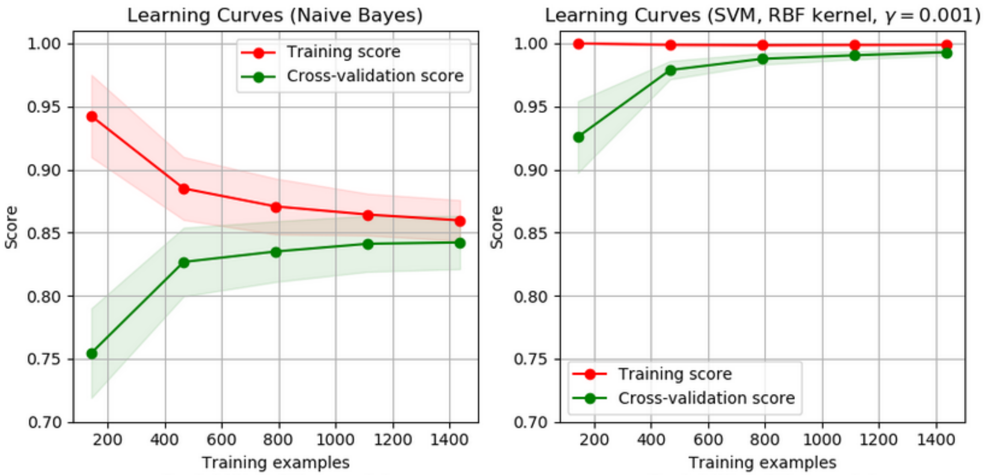


Confusion Matrix

- Where did the learner go wrong ?
- Use a confusion matrix (contingency table)

correct class (that should have been assigned)	classes assigned by the learner							
	C1	C2	C3	C4	C5	C6	...	Total
C1	94	3	0	0	3	0		100
C2	0	93	3	4	0	0		100
C3	0	1	94	2	1	2		100
C4	0	1	3	94	2	0		100
C5	0	0	3	2	92	3		100
C6	0	0	5	0	10	85		100
...								

Learning Curve



Plot evaluation metric vs. size of training set

- the more, the better
- but after a while, not much improvement. . .

Some Words on Training…

Watch out for:

- Noisy Data
- Overfitting/Underfitting

Noisy Data

René Witte

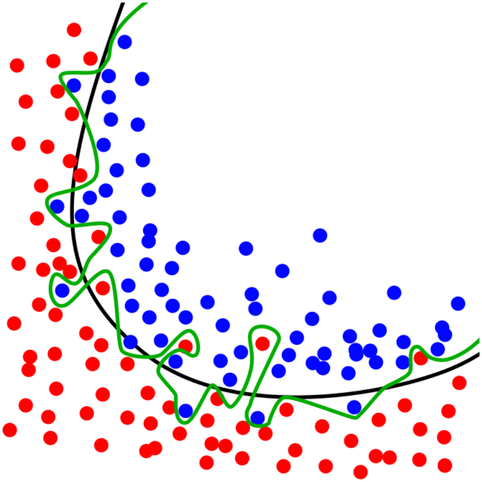
Common issues

- Two examples have the same feature-value pairs, but different outputs
- Some values of features are incorrect or missing (ex. errors in the data acquisition)
- Some relevant attributes are not taken into account in the data set

Size	Color	Shape	Output
Big	Red	Circle	+
Big	Red	Circle	-

Overfitting

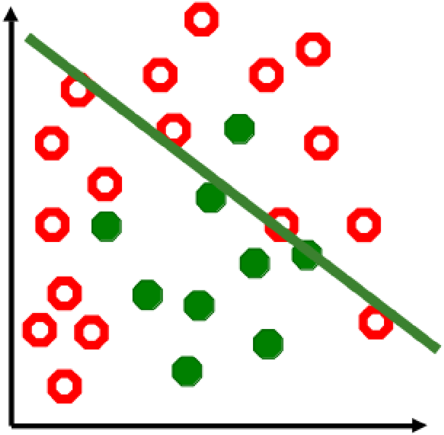
- If a large number of irrelevant features are there, we may find meaningless regularities in the data that are particular to the training data but irrelevant to the problem.
- Complicated boundaries overfit the data (a.k.a. *overtraining*)
- they are too tuned to the particular training data at hand
- They do not generalize well to the new data
- Extreme case: “rote learning”
  - Training error is low
  - Testing error is high



Underfitting

René Witte

- We can also underfit data, i.e. use too simple decision boundary
- Model is not expressive enough (not enough features)
- a.k.a. Undertraining
- There is no way to fit a linear decision boundary so that the training examples are well separated
  - Training error is high
  - Testing error is high



Example: Animal Classification

Features

What about cat vs. dog?

has-hair?	has-scales?	has-feathers?	flies?	lives in water?	lays eggs?	
1	0	0	0	0	0	Dog
1	0	0	0	0	0	Cat
1	0	0	1	0	0	Bat
1	0	0	0	1	0	Whale
0	0	1	1	0	1	Canary
0	0	1	1	0	1	Robin
0	0	1	1	0	1	Ostrich
0	1	0	0	0	1	Snake
0	1	0	0	0	1	Lizard
0	1	0	0	1	1	Alligator

[from: Alison Cawsey: The Essence of AI (1997)]

Cross-Validation

René Witte
Concordia University

Data Scarcity

- there is never enough training data
- so testing data is precious as well

k-fold Cross-Validation

'Re-use' different parts of the training data for testing. E.g., 10-fold cross-validation:

- split data into 10 equal parts
- train on 9 of these, test on the 10th
- repeat 10 times, resulting in 10 different performance results
- average these for overall performance

exp1:	train									test
exp2:	train								test	train
exp3:	train							test	train	
...	...									

Outline

1. Machine Learning Primer
2. Clustering Documents
3. Classifications & Predictions
4. Machine Learning Evaluation
5. **Notes and Further Reading**

Reading Material

Required

- [MG17, Chapters 2, 3, 5] (kNN, k-Means, Evaluation)

Supplemental

- [PS12, Chapter 7] (ML Training)
- [PS12, Chapter 8] (Testing and Evaluation)

References

[MG17] Andreas C Müller and Sarah Guido.
Introduction to Machine Learning with Python.
O'Reilly, 2017.
https://concordiauniversity.on.worldcat.org/oclc/960211579.

[PS12] James Pustejovsky and Amber Stubbs.
Natural Language Annotation for Machine Learning.
O'Reilly, 2012.
https://concordiauniversity.on.worldcat.org/oclc/801812987.