An Introduction to Machine Learning in R
Basic Concepts

Maya Alsheh Ali
Ashley Thompson

maya.alsheh.ali@ki.se
ashley.thompson@ki.se

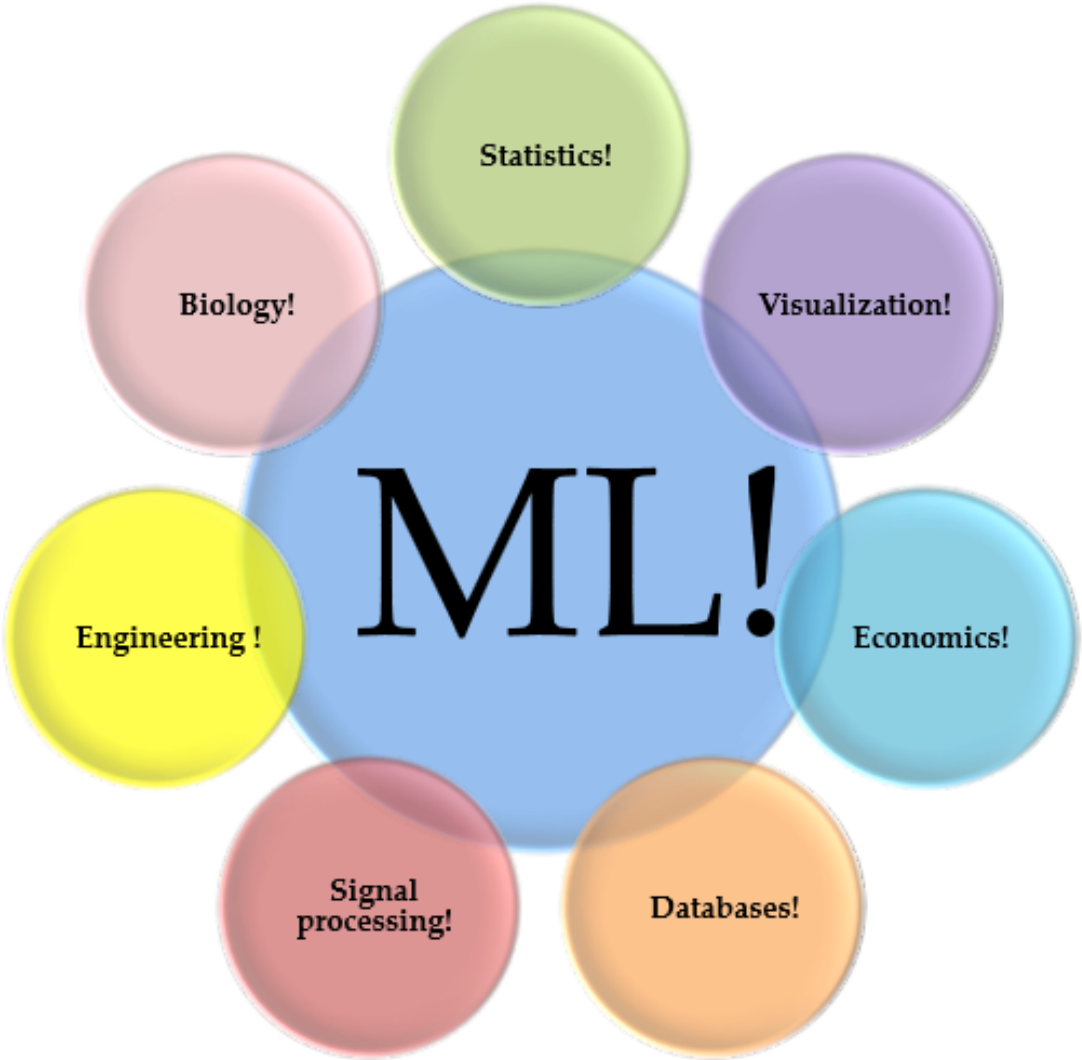
February 12, 2019


MACHINE LEARNING
MACHINE LEARNING EVERYWHERE!
memegenerator.net

Applications of ML

- Google search
- video recommendation
- image recognition
- fraud detection
- Medical diagnosis: diabetic retinopathy and skin cancer detection
- self-driving cars

ML:Interdisciplinary field



Overview

What is ML?

- At first ML may seem like magic, but once you dive in, you'll see that its a set of tools to derive meaning from data.
- ideas reach back to 1950s (Alan Turing)
- learning: automatic adaptation to data
- Using data to answer questions

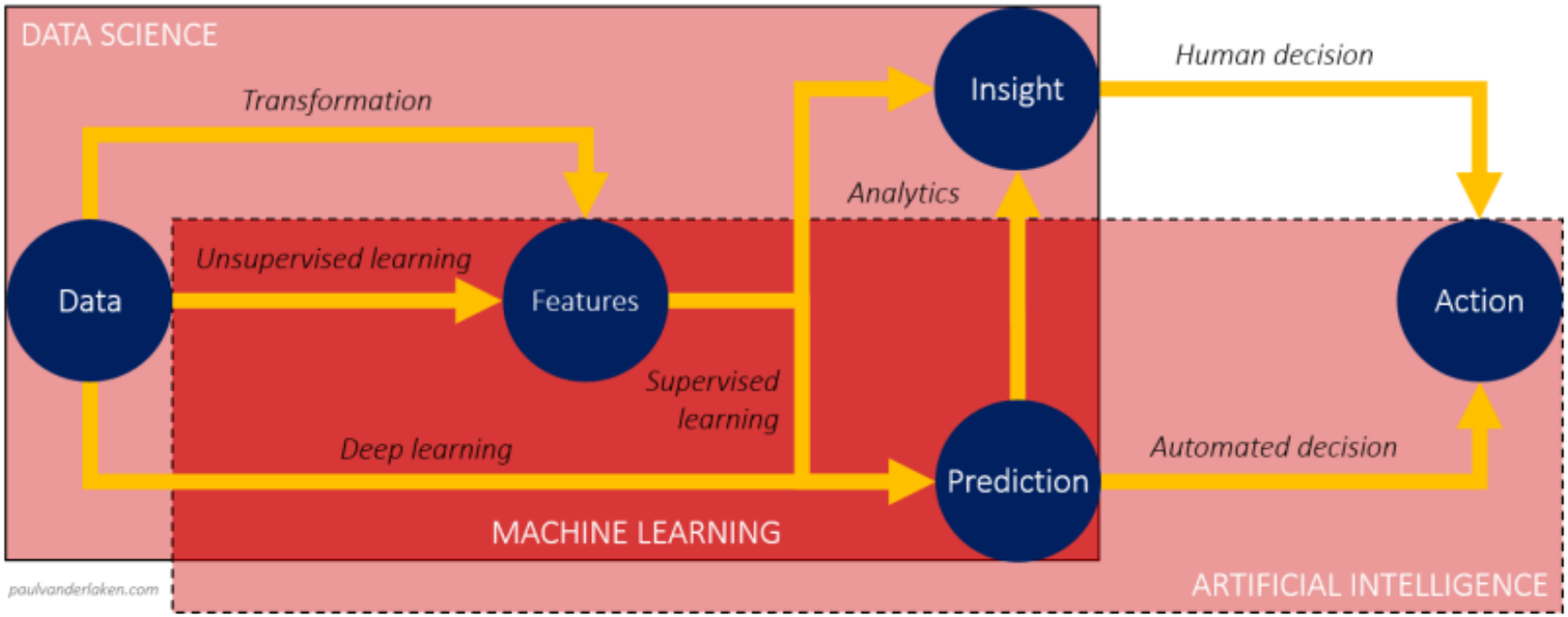
ML, Data science, AI

It is important to understand there are three major topics that are very different from one another:

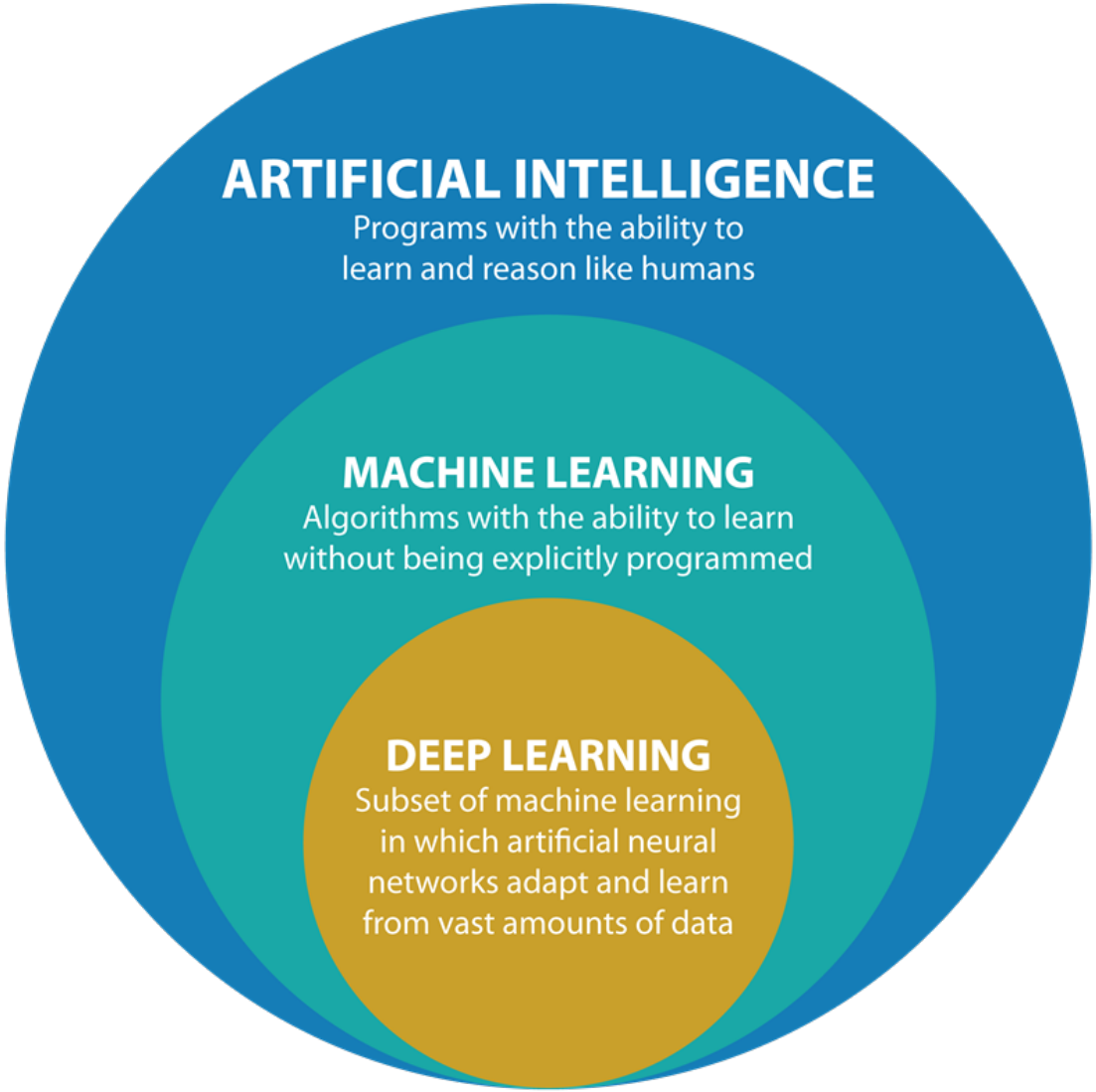
- Data Science: produces insights
- Machine Learning: produces predictions
- Artificial Intelligence: produces actions

There is a ton of overlap between these three fields and how humans learn.

ML, Data science, AI



ML, AI, Deep learning



Objectives of Machine Learning

Algorithms

- deal with large-scale problems
- make accurate predictions
- handle a variety of different learning problems

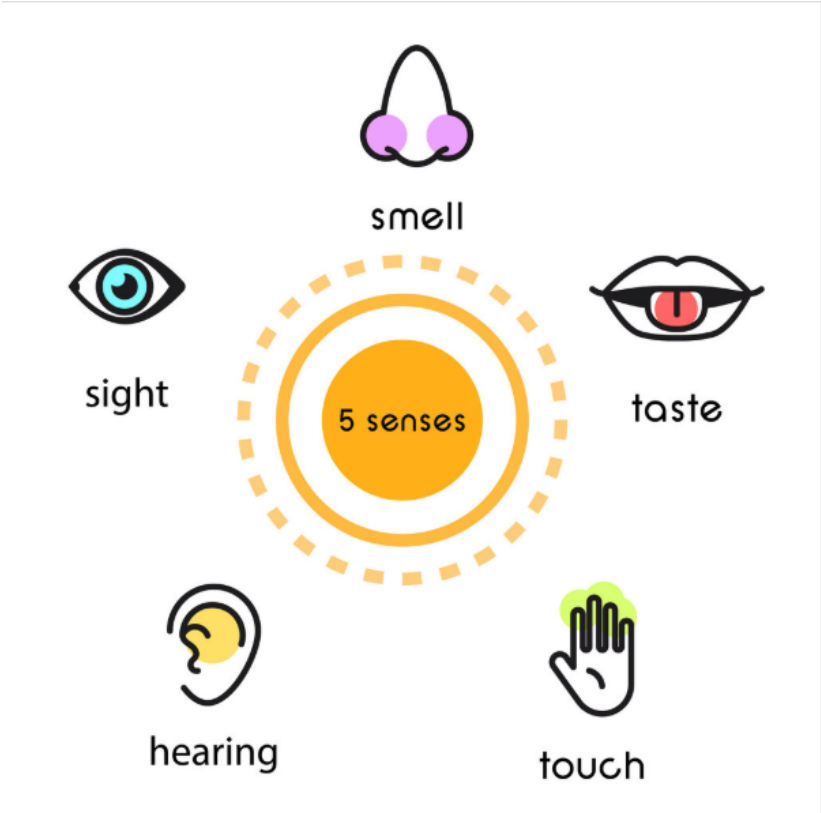
Theoretical questions

- what can be learned? Under what conditions?
- how well can it be learned computationally?

How do humans learn?

Humans have a superb natural ability to learn from experience.

- gather data through our sensors
- brain interprets this data
- makes decisions

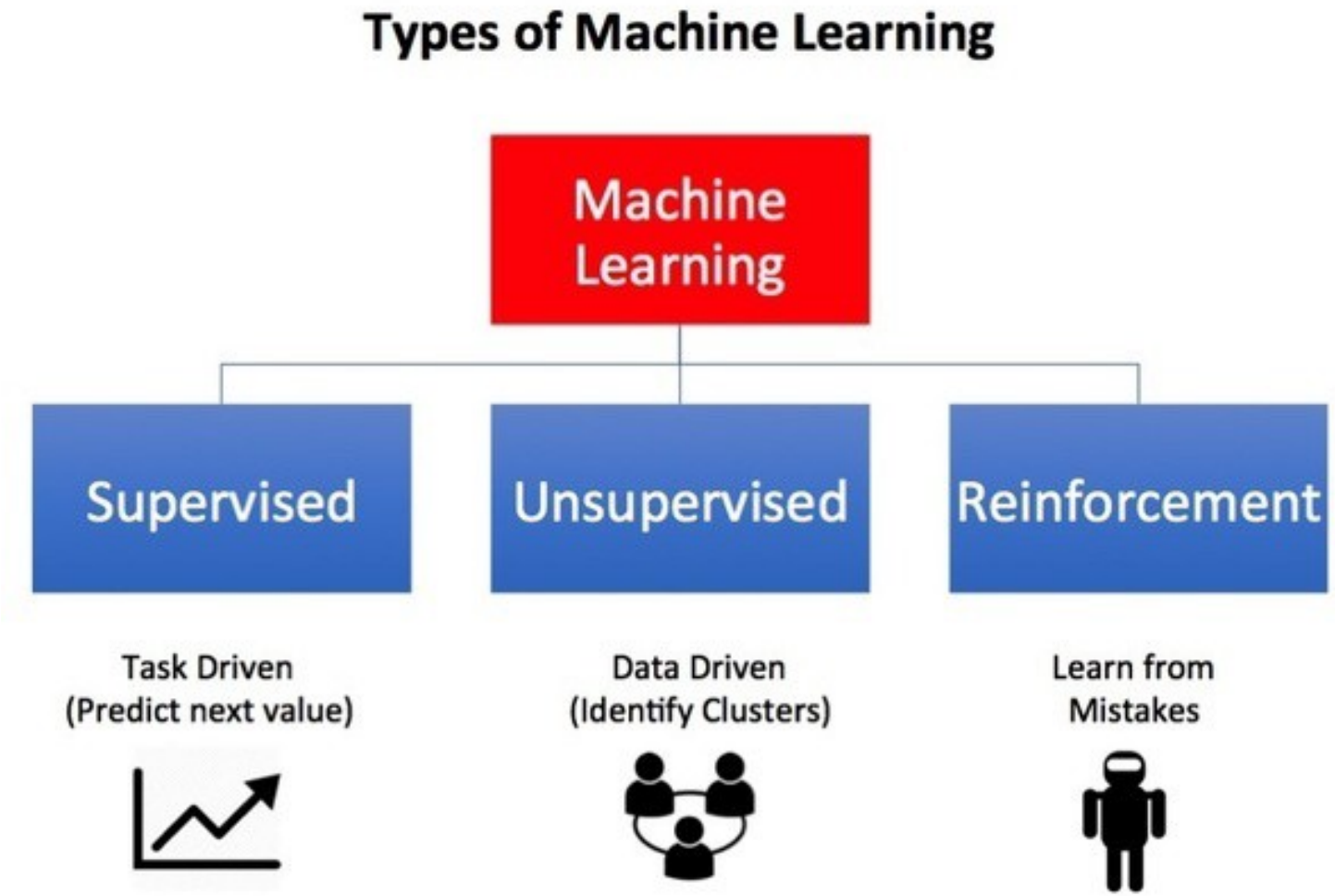


Basic Terminology

- Each separate row is a sample, example, observation or data point
- Each column is feature (or attribute) of that observation
- Usually there is one column (or feature), that we will call the target, label or response

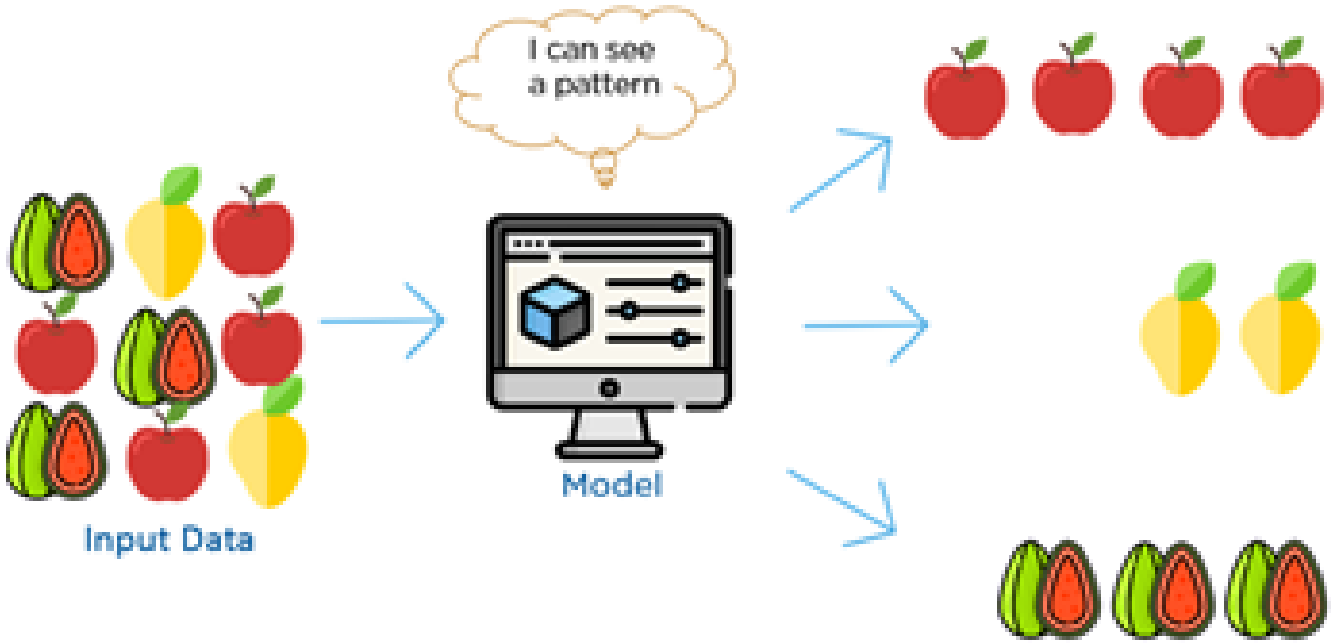
← Features →					Label
Position	**Experience**	**Skill**	**Country**	**City**	**Salary ($)**
Developer	0	1	USA	New York	103100
Developer	1	1	USA	New York	104900
Developer	2	1	USA	New York	106800
Developer	3	1	USA	New York	108700
Developer	4	1	USA	New York	110400
Developer	5	1	USA	New York	112300
Developer	6	1	USA	New York	114200
Developer	7	1	USA	New York	116100
Developer	8	1	USA	New York	117800
Developer	9	1	USA	New York	119700
Developer	10	1	USA	New York	121600

How do machines learn?

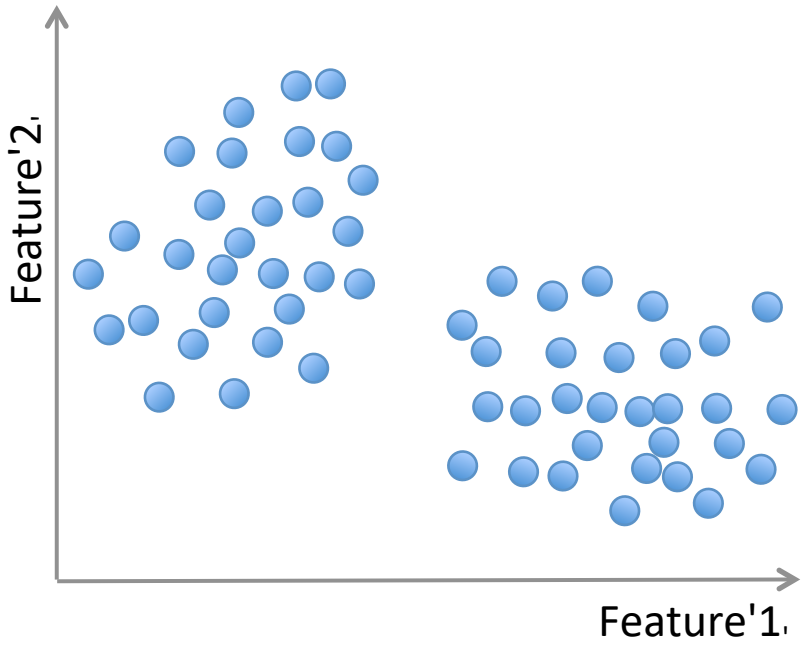


Unsupervised Learning

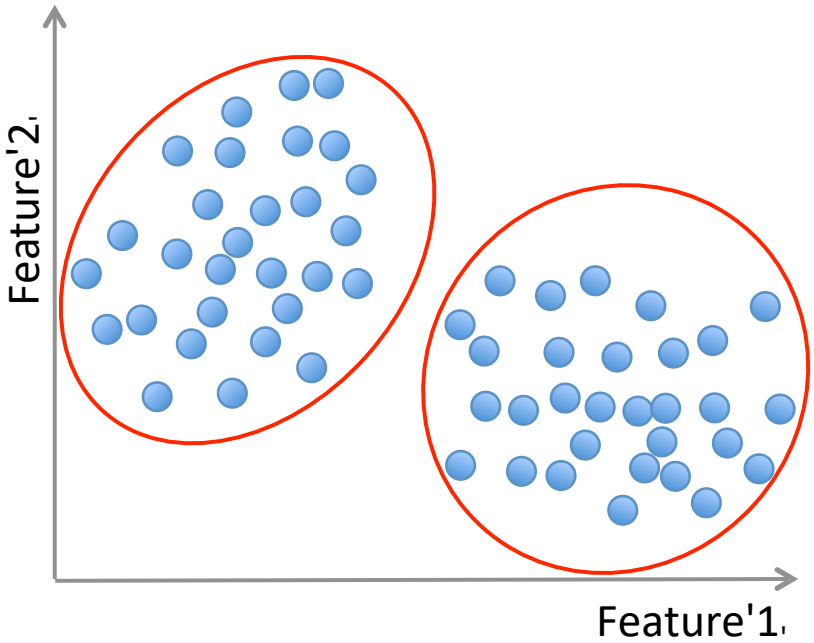
Discover patterns in data. It deals with unlabeled data of unknown structure and the goal is to explore the structure of the data to extract meaningful information, without the reference of a known outcome variable. (Clustering)



Unsupervised Learning: Clustering



Unsupervised Learning: Clustering



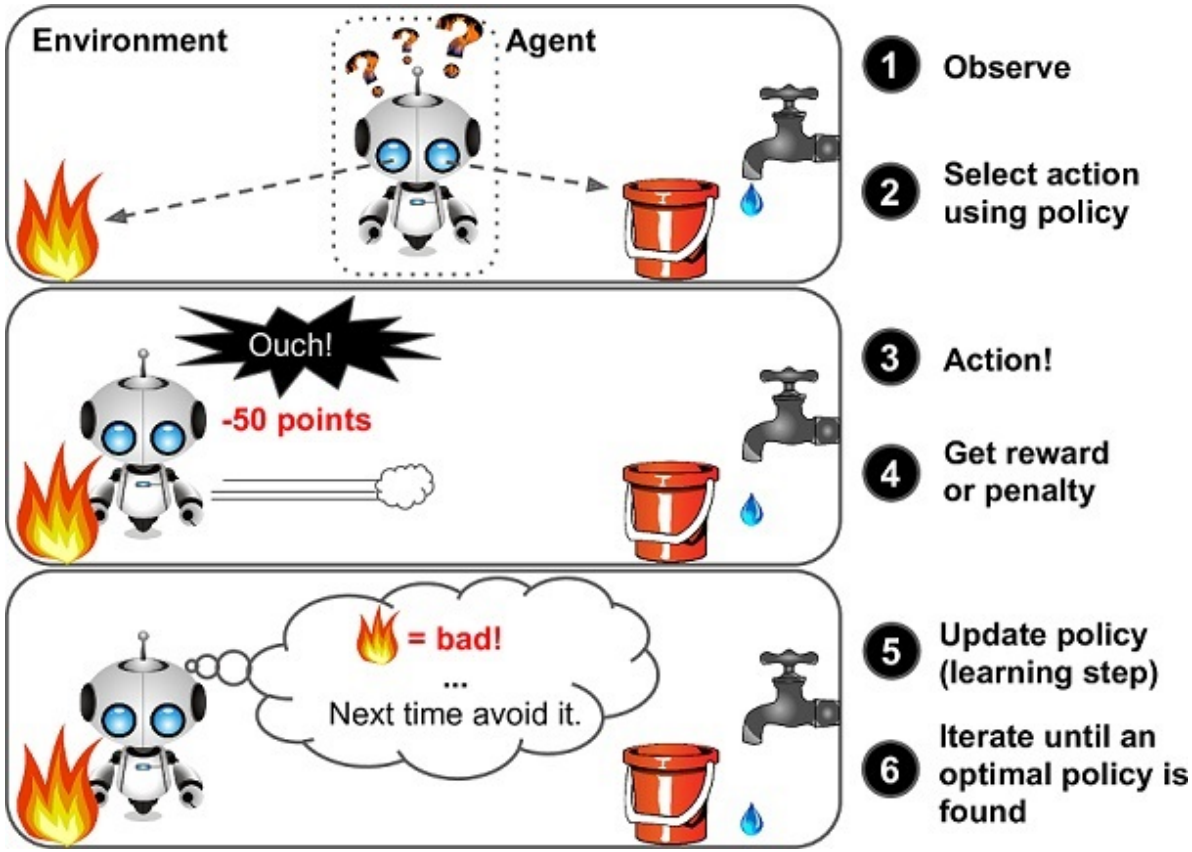
Methods: K-means, Gaussian mixtures, hierarchical clustering, spectral clustering, etc.

Unsupervised Learning: example

Recommender Systems (Netflix): We know things about videos, maybe their length, their genre, etc. We also know the watch history of many users. Taking into account users that have watched similar videos as you and then enjoyed other videos that you have yet to see, a recommender system can see this relationship in the data and prompt you with such a suggestion.

Reinforcement Learning

It is the science of making optimal decisions (Set the rules, let the algorithm learn by itself). It helps us formulate reward-motivated behaviour exhibited by living species.



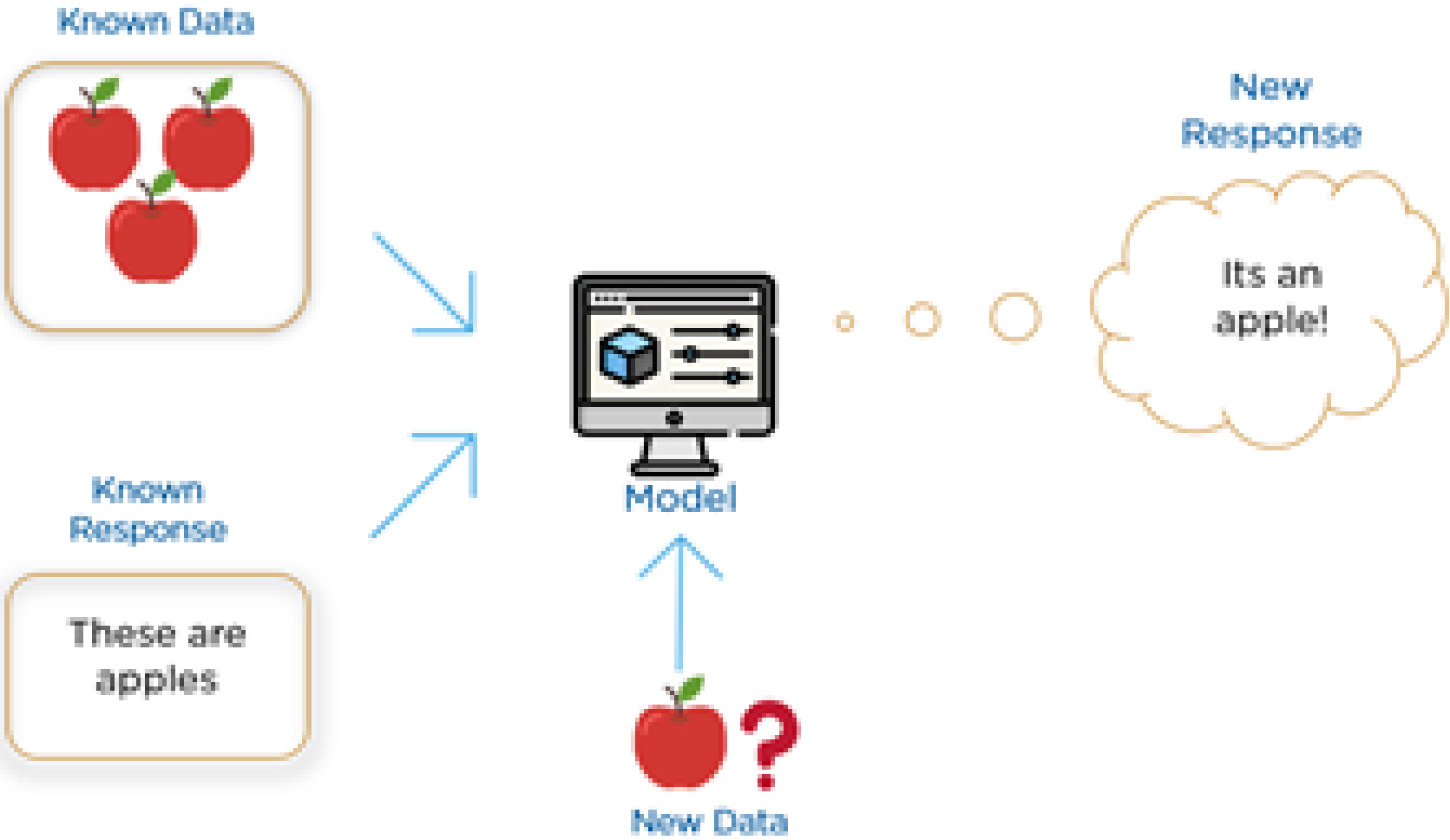
Supervised Learning

It relies on a supervisor (the human) and it learns how to combine input to produce useful predictions on never-before-seen data. It is used to solve two types of problems:

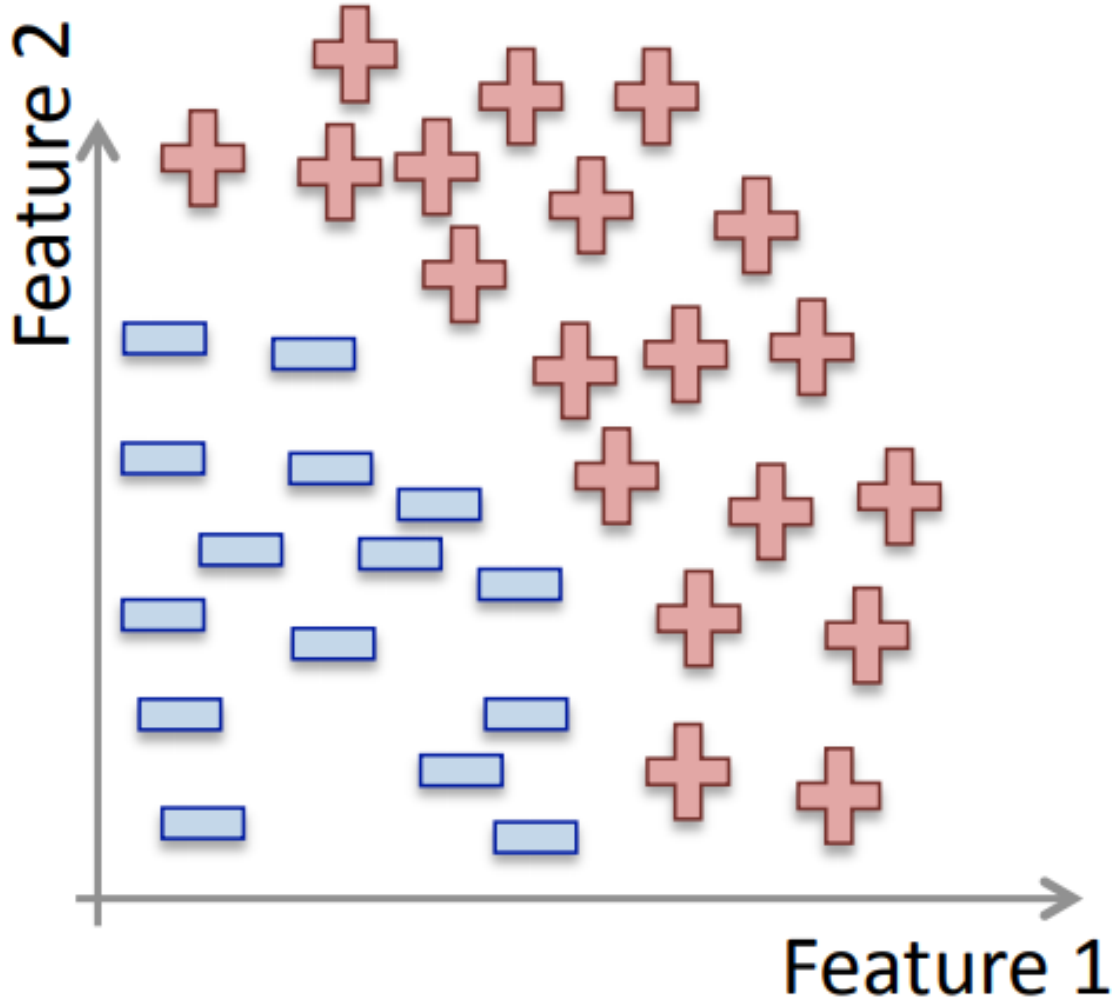
- Classification: grouping of like objects based on a defined characteristic (model predicts discrete values)
- Regression: dependent variable(s) associated with independent variable(s).(model predicts continuous values)

models are trained with a set of samples where the desired output signals (or labels) are already known.

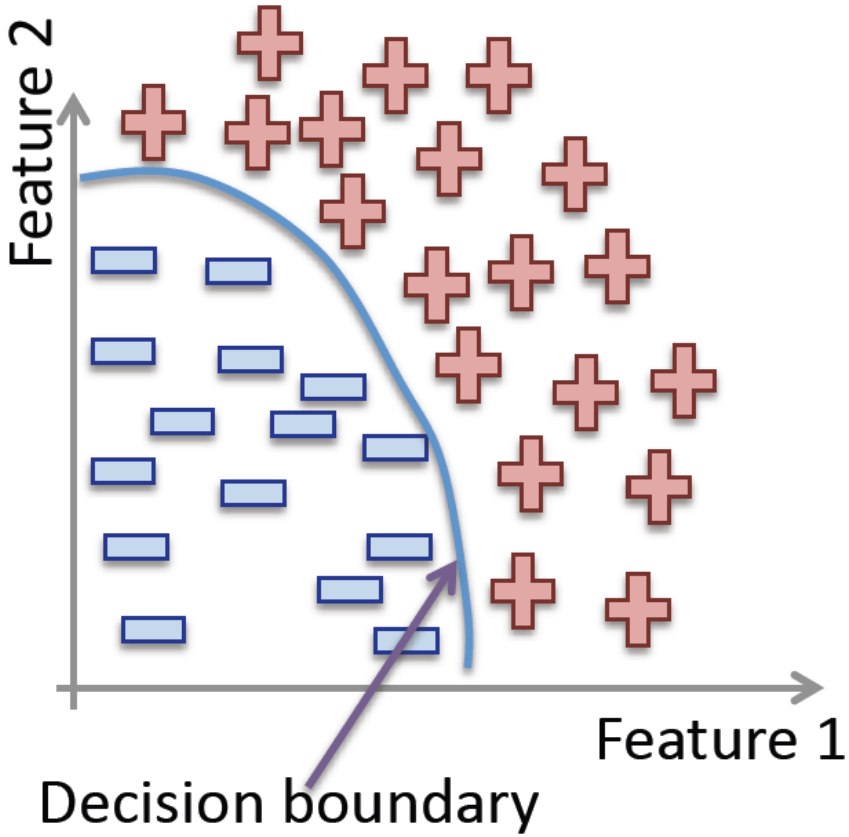
Supervised Learning: classification



Supervised Learning: classification

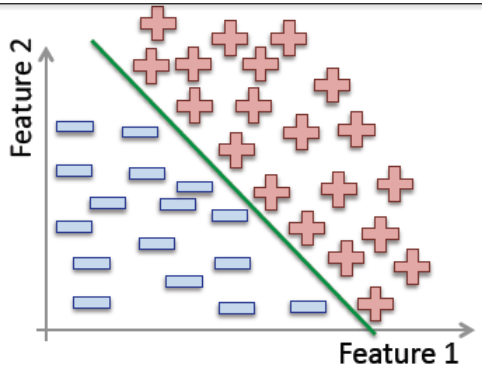


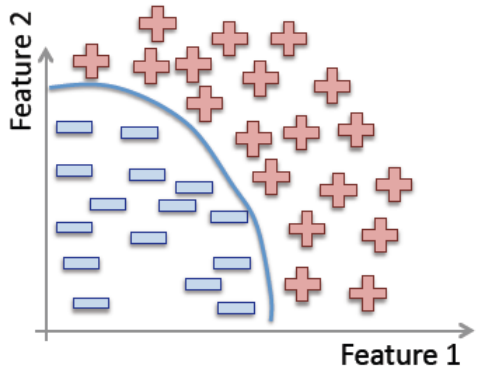
Supervised Learning: classification

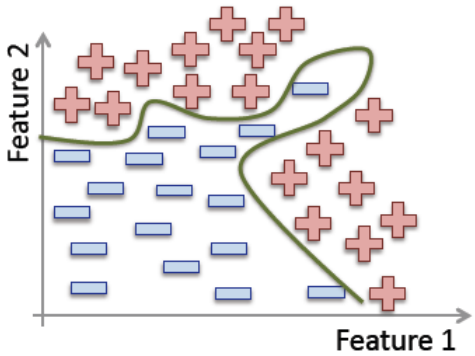


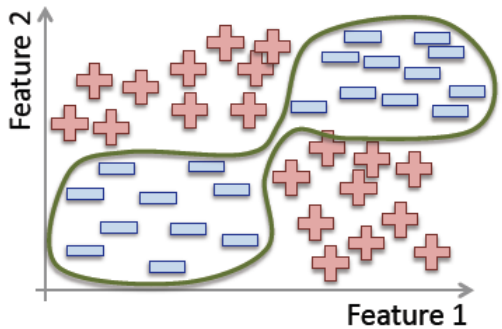
Methods: Support Vector Machines, neural networks, decision trees, K-nearest neighbors, naive Bayes, etc.

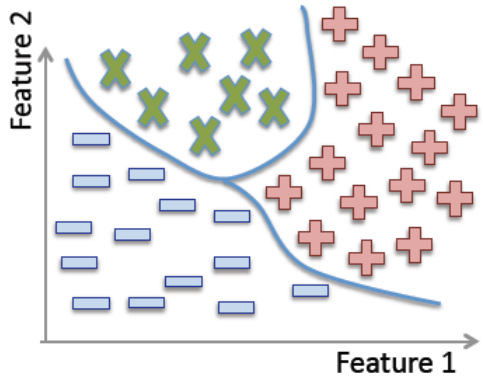
Supervised Learning: classification





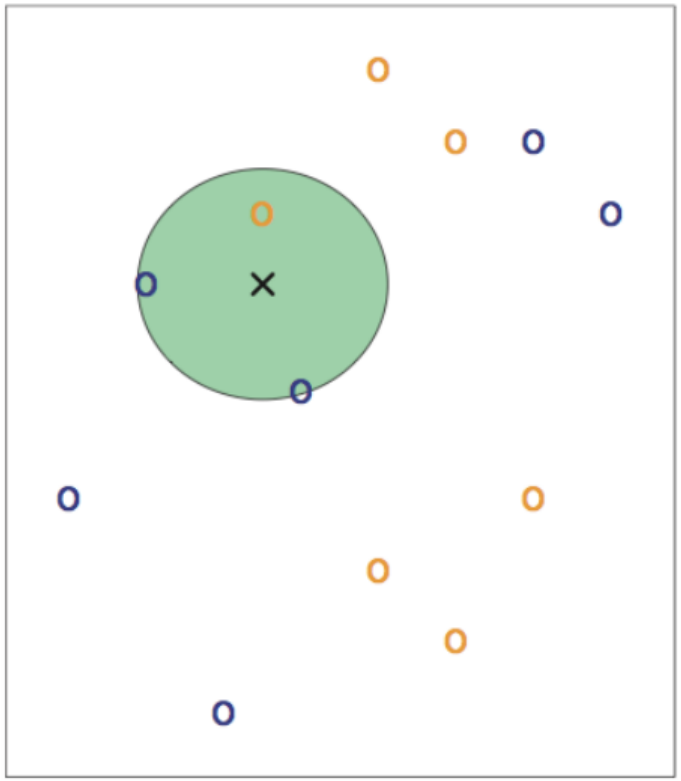






Classification: K-nearest neighbors

- Not every ML method builds a model!
- Main idea: Uses the similarity between examples.
- Assumption: Two similar examples should have same labels



Classification: K-nearest neighbors

Pros

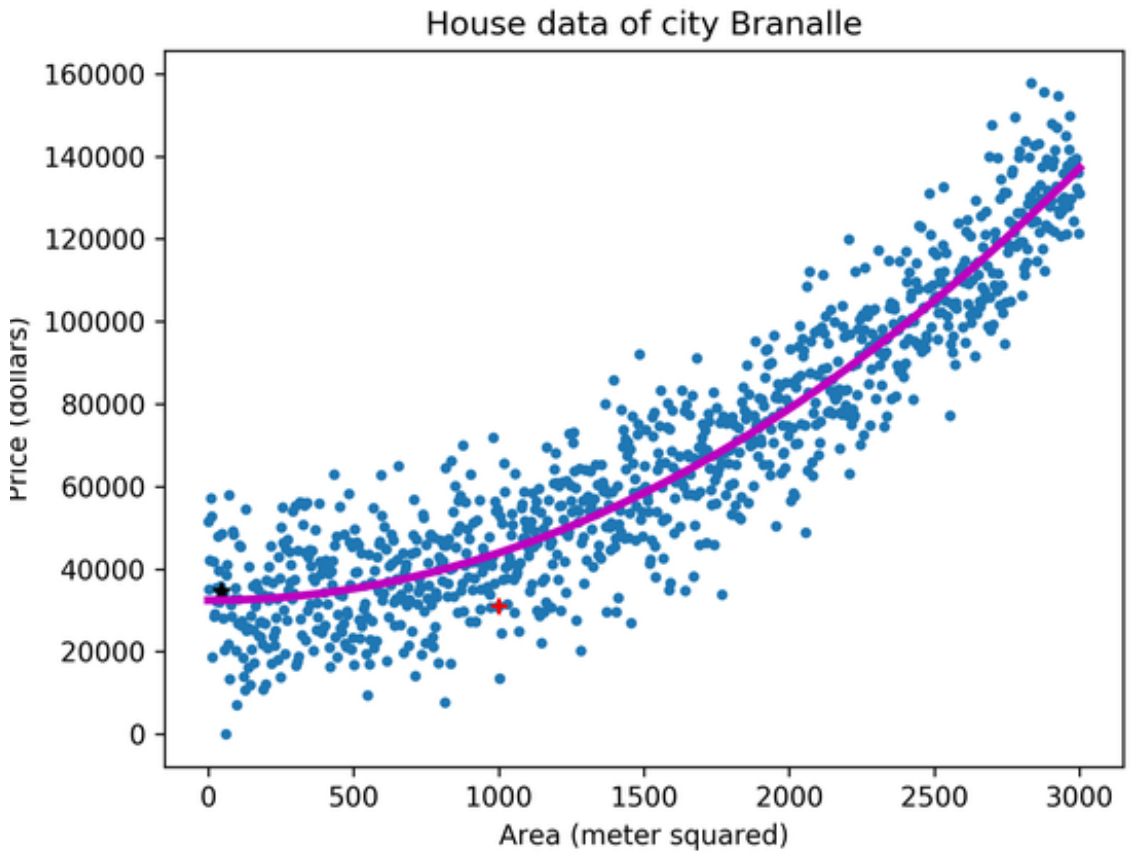
- Simple to implement.
- Works well in practice.
- Does not require to build a model, make assumptions, tune parameters
- Can be extended easily with news examples.

Cons

- Requires large space to store the entire training dataset.
- Slow!
- Suffers from the curse of dimensionality

Supervised Learning: regression

The goal of this regression is to be able to predict the price of a given house after knowing the area of a given house.



Machine Learning workflow

- Gathering data
- Data pre-processing
- Researching the model that will be best for the type of data
- Training and testing the model
- Evaluation
- Prediction

Gathering data

This step is very important because the quality and quantity of data that you gather will directly determine how good your predictive model can be. But most of the real-world data is messy:

- Missing data: not continuously created or due to technical issues in the application
- Noisy data: Also called outliners
- Inconsistent data: (mistakes with the name or values)

That's why we need data pre-processing

Data pre-processing

Also called feature engineering

- Data cleaning: remove/fill in missing values, smooth noisy data, identify or remove outliers, randomize the ordering, and resolve inconsistencies.
- Data integration: using multiple databases, data cubes, or files.
- Data transformation: normalization and aggregation.

Researching the model

- Although the general idea in supervised learning is the same, there is a large number of different models available, with different properties and performance.
- There is no generally best model - prediction performance depend on the the specific problem.

Training and testing the model

Training means creating or learning the model. That is, you show the model labeled examples and enable the model to gradually learn the relationships between features and label.

Testing (or inference) means applying the trained model to unseen examples. That is, you use the trained model to make useful predictions

Data partition

Divide your data set into two subsets:

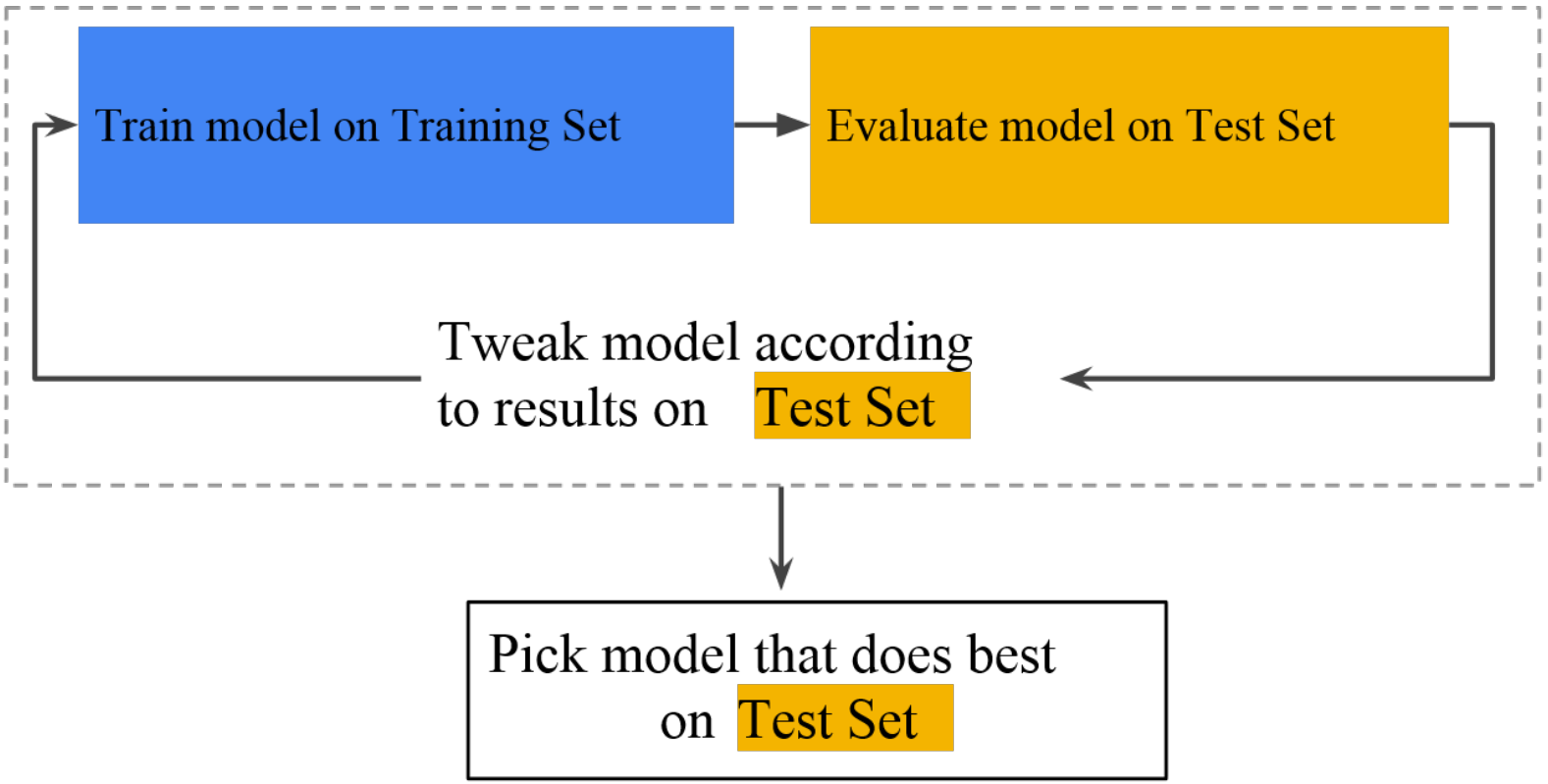
- training set: a subset to train a model.
- test set: a subset to test the trained model.

Make sure that your test set meets the following two conditions:

1. large enough to yield statistically meaningful results.
2. representative of the data set as a whole. In other words, don't pick a test set with different characteristics than the training set

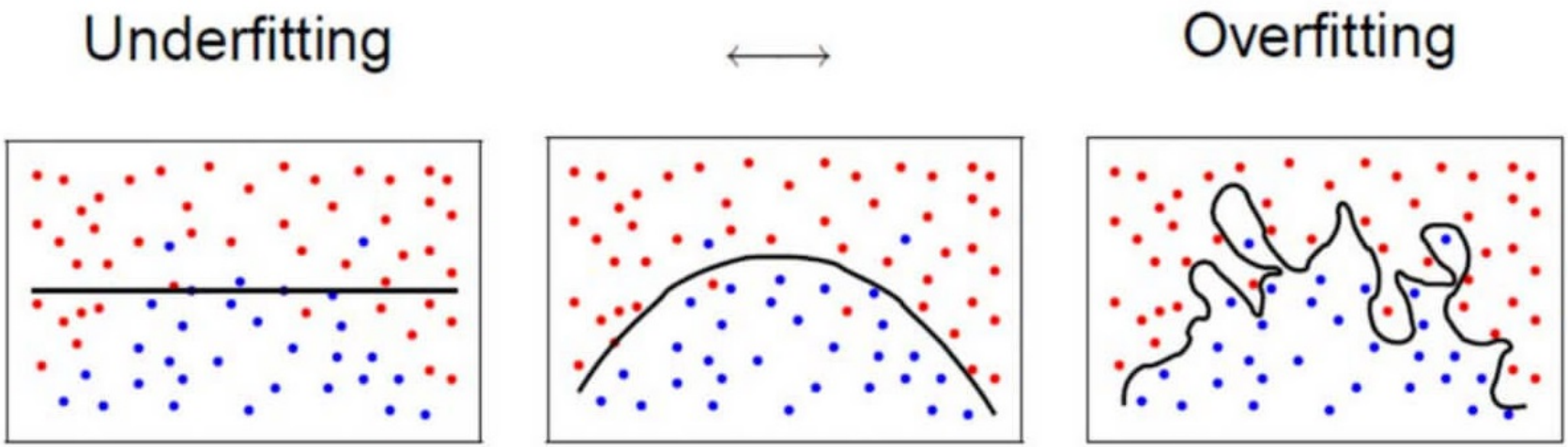
Data partition: A possible workflow?

Never train on test data: If you are seeing surprisingly good results on your evaluation metrics, it might be a sign that you are accidentally training on the test set.



Overfitting

It is defined by a model that fits very well to the training data, while it generalizes poorly to new data and consequently does not predict future observations well.



Remember! Generalization: not memorization.

Avoid overfitting

In general, use simple models!

- Reduce the number of features manually or do feature selection.
- Do a model selection.
- Use regularization (keep the features but reduce their importance by setting small parameter values).
- Do a cross-validation to estimate the test error

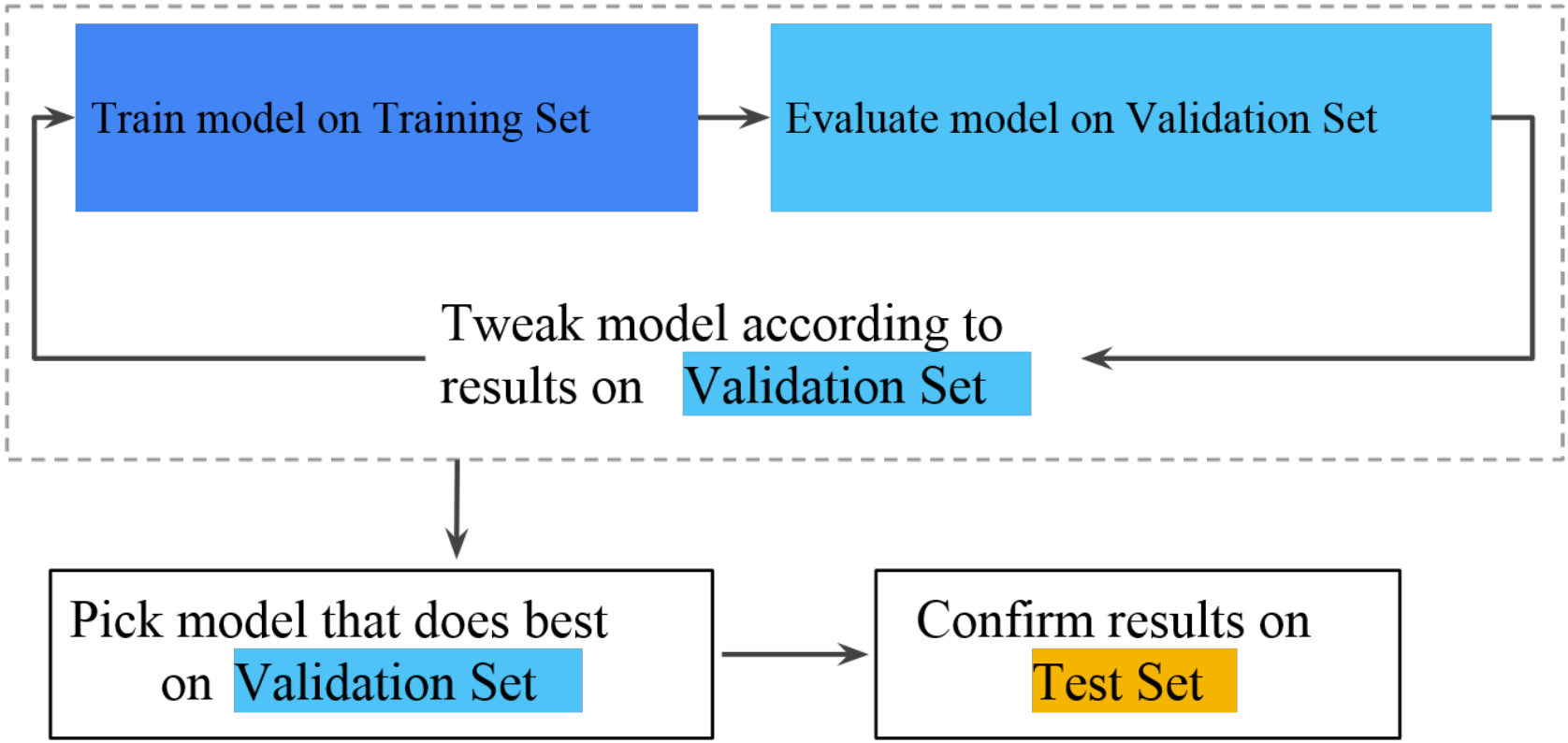
Another partition: Holdout validation

You can greatly reduce your chances of overfitting by partitioning the data set into the three subsets

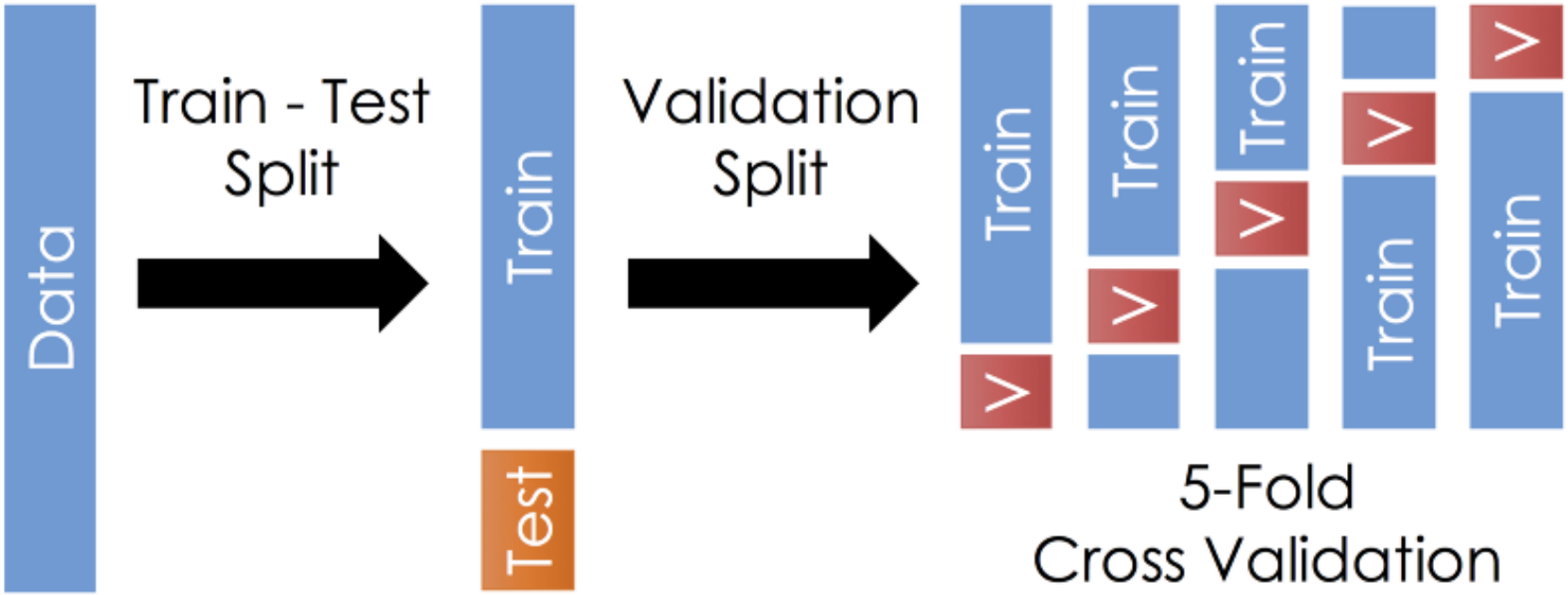
1. Pick the model that does best on the validation set
2. Double-check that model against the test set.

Another partition: A better workflow

This is a better workflow because it creates fewer exposures to the test set. But we use less observations to train the model.



Another partition: Cross Validation



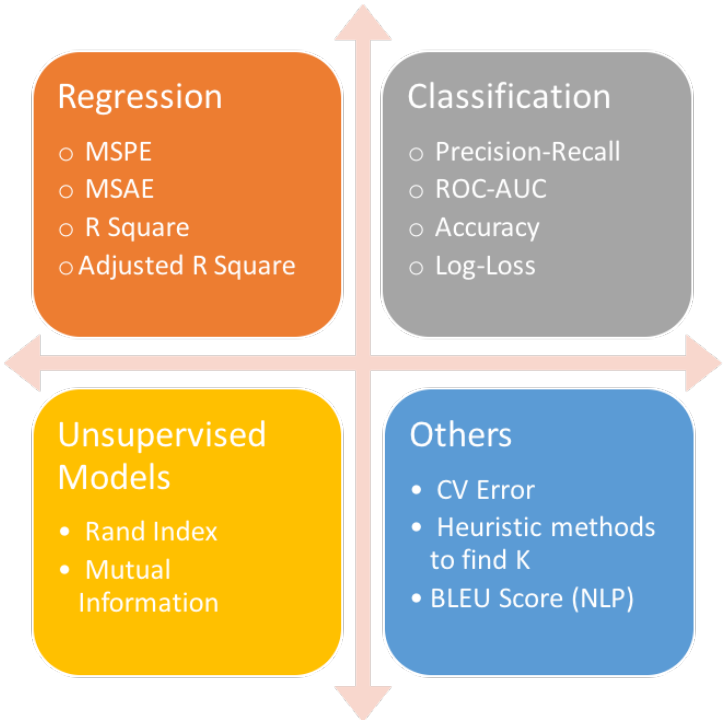
Another partition: Cross Validation

A few common methods used for cross validation:

- Leave out one cross validation (LOOCV)
- k-fold cross validation
- Stratified k-fold cross validation
- Adversarial validation
- Cross validation for time series
- Custom cross validation techniques

Metrics to assess models performance

Each machine learning model is trying to solve a problem with a different objective using a different dataset and hence, it is important to understand the context before choosing a metric.



Metrics for classification

- Confusion matrix
- Sensitivity and specificity
- Receiver Opertaing Characteristic (ROC)

Confusion matrix

- helps us to summarize the confusion created by the model which makes incorrect predictions.
- there are four important terms which define the confusion matrix
- important metrics are computed from a Confusion Matrix

		Actual Label	
		Positive	Negative
Predicted Label	Positive	**True Positive (TP)**	**False Positive (FP)**
	Negative	**False Negative (FN)**	**True Negative (TN)**

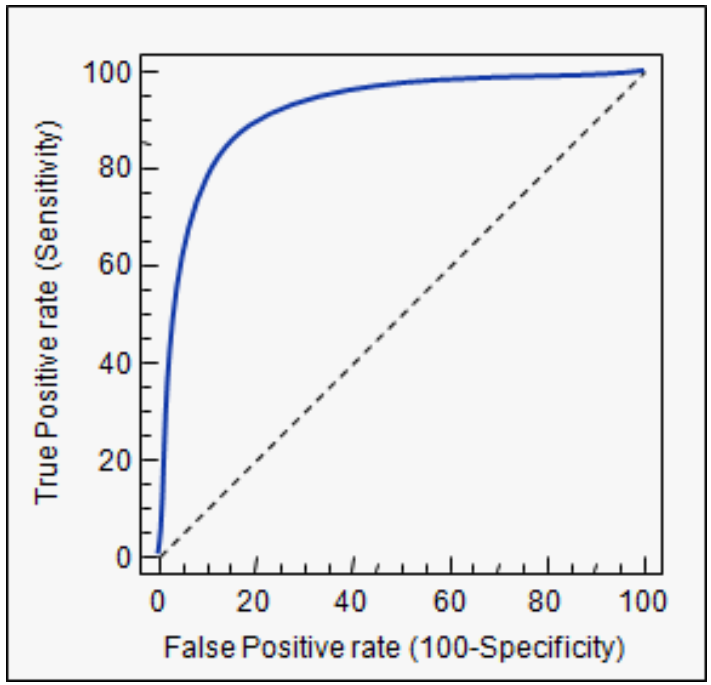
Confusion matrix

		Actual Label	
		Positive	Negative
Predicted Label	Positive	**True Positive (TP)**	**False Positive (FP)**
	Negative	**False Negative (FN)**	**True Negative (TN)**

Accuracy	(TP + TN) / (TP + TN + FP + FN)	The percentage of predictions that are correct
Precision	TP / (TP + FP)	The percentage of positive predictions that are correct
Sensitivity (Recall)	TP / (TP + FN)	The percentage of positive cases that were predicted as positive
Specificity	TN / (TN + FP)	The percentage of negative cases that were predicted as negative

AUC ROC Curve

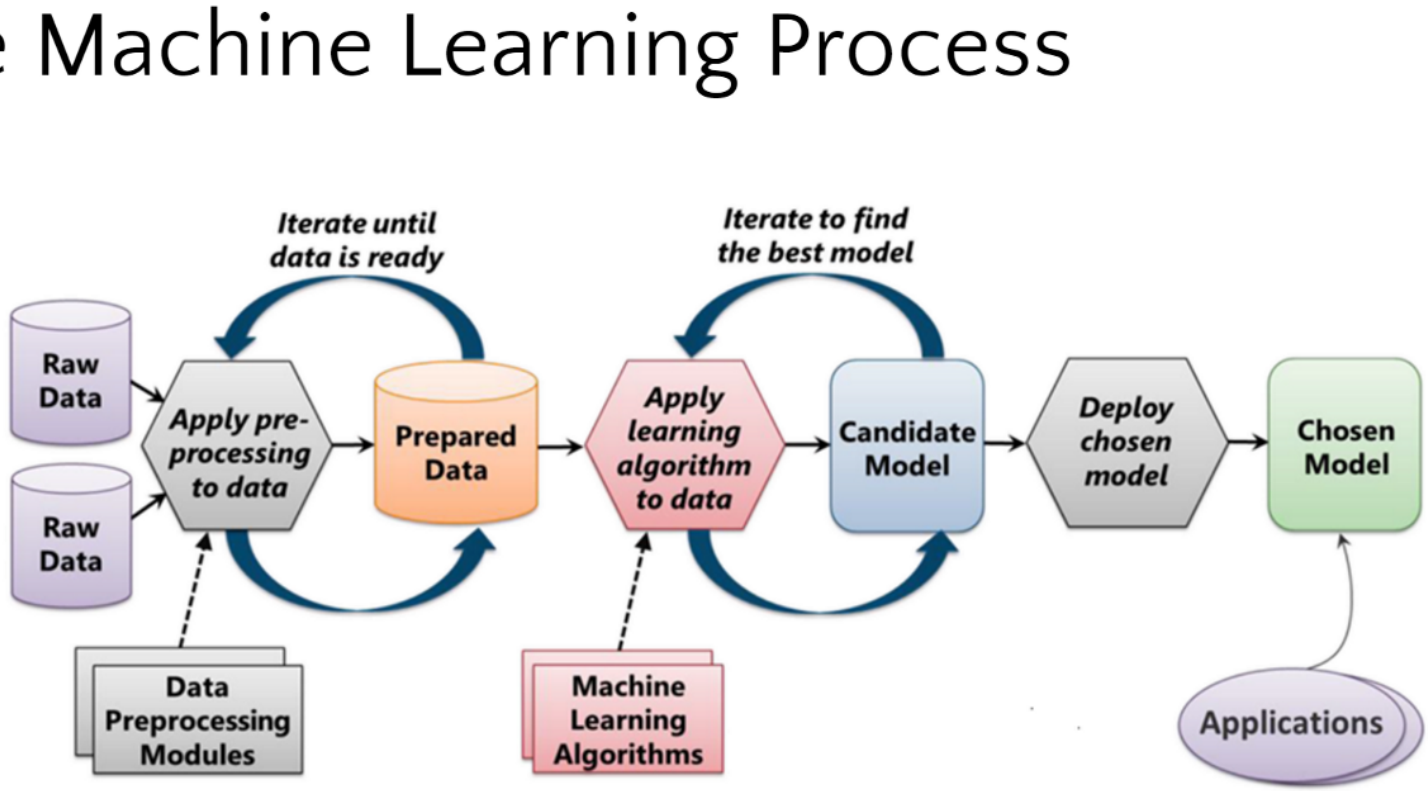
ROC (Receiver Operating Characteristic) Curve tells us about how good the model can distinguish between two things (e.g If a patient has a disease or no). AUC is an abbrevation for area under the curve.



Caution: Typically, no single correct evaluation metric

- evaluation metrics can introduce unfairness / bias especially when training sets are unbalanced (many more no than yes cases, prevalence/lack of input feature combinations)
- use great care when constructing training sets
- use multiple evaluation metrics perform

Summary



From "Introduction to Microsoft Azure" by David Chappell

Limitation of ML

- Requires Massive Stores of Training Data
- Labeling Training Data Is a Tedious Process
- Machines Cannot Explain Themselves
- There is Bias in the Data

Usefull links

- https://www.kaggle.com/camnugent/introduction-to-machine-learning-in-r-tutorial
- https://www.r-bloggers.com/in-depth-introduction-to-machine-learning-in-15-hours-of-expert-videos/

An Introduction to Machine Learning in R
Basic Concepts

Maya Alsheh Ali
Ashley Thompson

maya.alsheh.ali@ki.se
ashley.thompson@ki.se

February 12, 2019