Machine Learning

Introduction

Gabriel Valverde Castilla

gvalverd@ucm.es

Universidad Complutense *de*

Madrid

Table of contents

Table of contents

What is Machine Learning?

What is Machine Learning?

What is Machine Learning?

What is Machine Learning?

Machine Learning is any process by which a system improves performance in a task from experience

(Herbert Simon)

A well defined learning task is given by $< T, E, P >$:

- A task T
- A set of data examples (experience) E
- A function to measure improvement P

Traditional programming vs Machine Learning

Traditional programming

Traditional programming vs Machine Learning

Traditional programming

Machine Learning

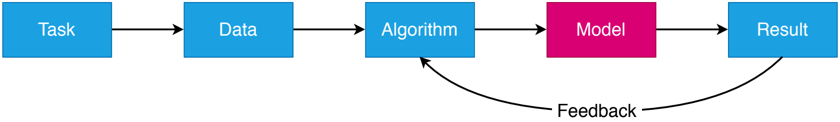
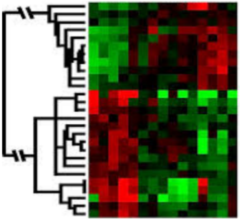


Table of contents

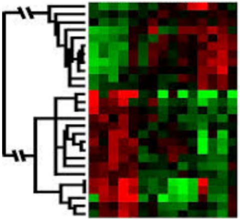
When do we use Machine Learning?

Tasks involving Big Data

- Genomics
- Internet search
- Anomaly detection



When do we use Machine Learning?



Tasks involving Big Data

- Genomics
- Internet search
- Anomaly detection

Tasks requiring customization

- Email filters
- Personalized medicine
- Movie recommendation

When do we use Machine Learning?

Tasks for which it is challenging to specify our knowledge

- Facial recognition
- Speech recognition
- Medical diagnosis

When do we use Machine Learning?

Tasks for which it is challenging to specify our knowledge

- Facial recognition
- Speech recognition
- Medical diagnosis

Tasks which we don't have human expertise

- Space exploration
- Undersea manipulation
- House price estimation

When we don't use it

Tasks with lack of data / good data

- Lack of mammography data from black women

Tasks for which a simpler solution may suffice

- Estimate taxes

Tasks with ethical considerations

- Uyghur facial recognition for persecution

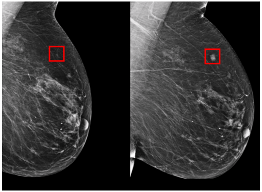


Table of contents

Types of learning

Supervised learning

- Training data + desired outputs (labels)

Unsupervised learning

- Training data (without labels)

Semi-supervised learning

- Training data + some labels

Reinforcement learning

- Observations and periodic rewards as the agent takes sequential action in an environment

Supervised learning

	X_1	$\dots$	X_n	Y
$(\mathbf{x}^{(1)}, y^{(1)})$	$x_1^{(1)}$	$\dots$	$x_n^{(1)}$	$y^{(1)}$
$(\mathbf{x}^{(2)}, y^{(2)})$	$x_1^{(2)}$	$\dots$	$x_n^{(2)}$	$y^{(2)}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$(\mathbf{x}^{(m)}, y^{(m)})$	$x_1^{(m)}$	$\dots$	$x_n^{(m)}$	$y^{(m)}$

Regression

- X_i is continuous
- Y is continuous

Classification

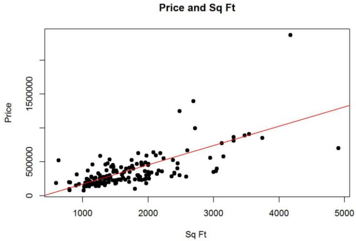
- X_i is discrete/continuous
- Y is discrete

Supervised learning: Regression

How much money should we ask for it?

Supervised learning: Regression

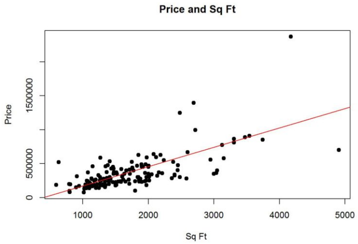
- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is continuous



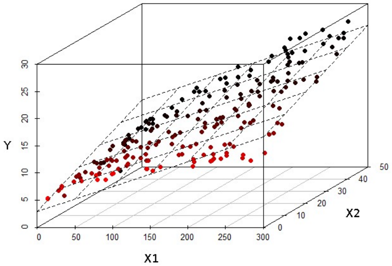
One-dimensional

Supervised learning: Regression

- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is continuous

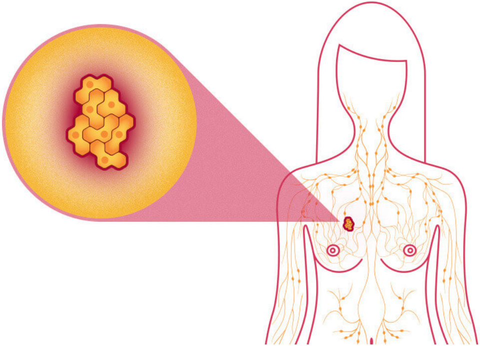


One-dimensional



Multi-dimensional

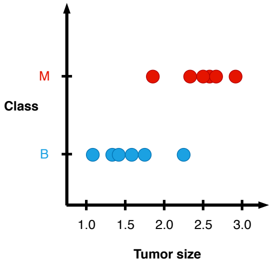
Supervised learning: Classification



Predict if a tumor is malign

Supervised learning: Classification

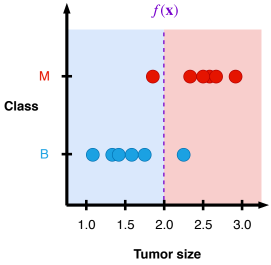
- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is discrete



One-dimensional

Supervised learning: Classification

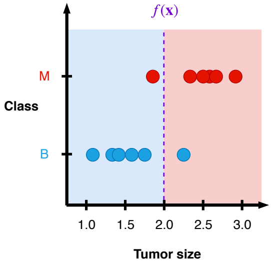
- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is discrete



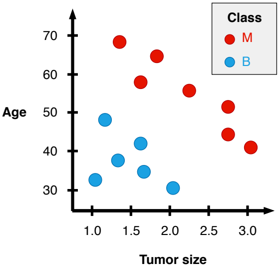
One-dimensional

Supervised learning: Classification

- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is discrete



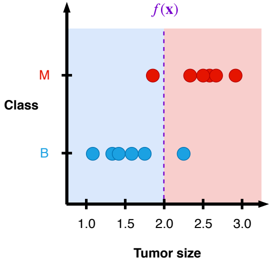
One-dimensional



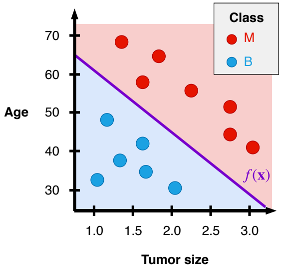
Multi-dimensional

Supervised learning: Classification

- Given $(\mathbf{x}^{(1)}, y^{(1)})$ learn a function $f(\mathbf{x})$ to predict y given $\mathbf{x}$
- y is discrete



One-dimensional



Multi-dimensional

Unsupervised learning

	X_1	$\dots$	X_m	Y
$(\mathbf{x}^{(1)}, ?)$	$x_1^{(1)}$	$\dots$	$x_n^{(1)}$	$?$
$(\mathbf{x}^{(2)}, ?)$	$x_1^{(2)}$	$\dots$	$x_n^{(2)}$	$?$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$(\mathbf{x}^{(m)}, ?)$	$x_1^{(m)}$	$\dots$	$x_n^{(m)}$	$?$

Clustering

- X_i is discrete/continuous
- Y is discrete

Dimensionality reduction

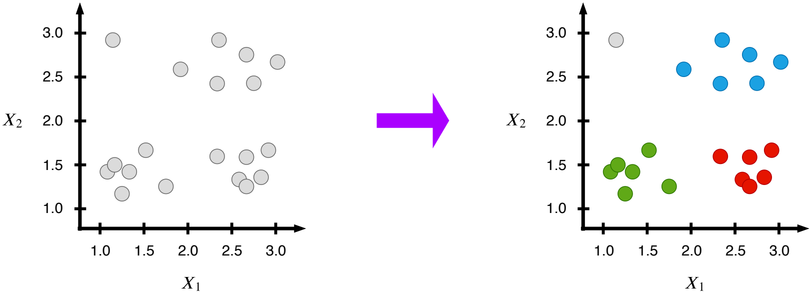
- X_i is continuous
- $Y_1 \dots Y_K$ is continuous

Unsupervised learning: Clustering

How many groups of customers?

Unsupervised learning: Clustering

- Given a set of instances $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(m)}$ (no labels)
- Output the **hidden** structure behind data
- Allows statistical inference



Reinforcement learning

How can we teach a machine to play Go?

Reinforcement learning

Why not simply use supervised learning?

We need a dataset with millions of games where

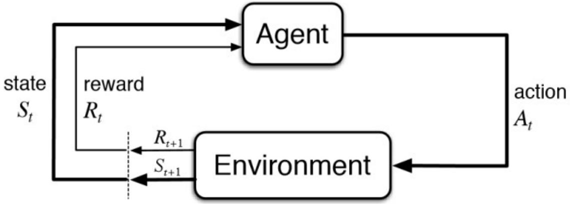
- Each possible game state is a row
- The class we want to predict is a sequence of optimal moves

Problems

- Datasets like this don't exist for all domains we care about
- Creating such dataset could be expensive and infeasible
- Difficult to learn such a pattern (how many class labels?)
- It would only **imitate** instead of actually learn **the best strategy**

Reinforcement learning

- A policy is produced from a sequence of states $\mathbf{S}$ and actions $\mathbf{A}$, where each step $\{S_t, A_t\}$has an associated reward R_t
- Discrete time with Markov property



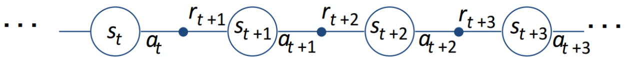


Table of contents

Components of every ML application

*Learning is equivalent to searching (**optimization**) through the space of potential models (**representation**) to find the one that best performs (**evaluation**) with the training data*

Representation

- How is data specified?
- What is the form of the model?

Optimization

- How is the model trained on the data?

Evaluation

- How are we assessing the model's performance?

Representation

Numerical functions	Symbolic functions	Instance-based functions	Probabilistic Graphical Models
Linear regression Linear discriminant analysis Neural network	Decision tree Propositional logic rules First-order logic rules	Nearest-neighbor Case-based	Naïve Bayes Bayesian network HMM Markov network

Optimization and search techniques

- Gradient descent
- Expectation-Maximization
- Maximum likelihood estimation
- Dynamic programming
- Divide and Conquer
- Evolutionary Computation
- etc.

Evaluation

- Squared error
- Likelihood
- Posterior probability
- Information gain
- Accuracy
- Precision/recall
- etc.

Table of contents

Machine Learning Process

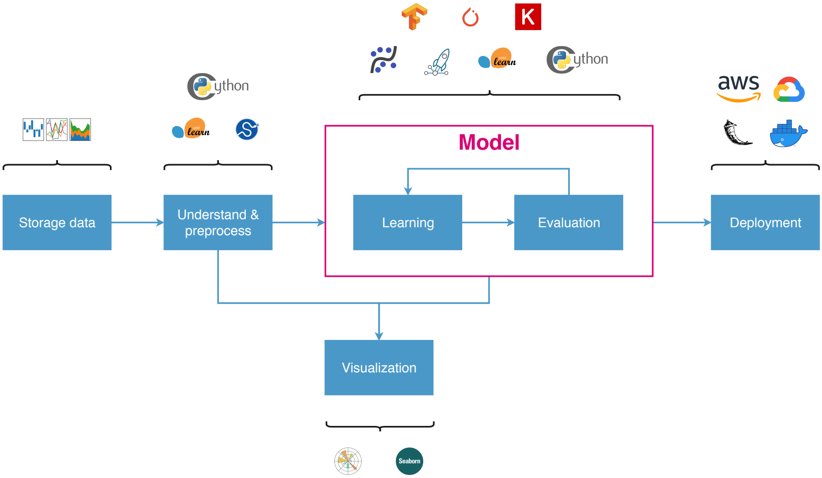


Table of contents

Some interesting links

Courses:

- An introduction to statistical learning
- An introduction to statistical learning (Python assignments)
- Coursera: Machine Learning
- Coursera: Machine Learning (Python assignments)
- Coursera: Deep Learning
- Dive into Deep Learning
- Bayesian methods for hackers

Webs:

- Kaggle
-

Machine Learning

Introduction