Machine Learning with Graphs: Introduction

Arlei Silva

Spring 2022

What is machine learning? How did it evolve? What are typical applications, tasks, and components of a machine learning problem and its solution? We will briefly answer some of these questions here. We will also motivate the use of graphs in machine learning using non-linear dimensionality reduction.

Machine learning

This is a brief overview of machine learning (ML) in a broad sense.

What is machine learning?

There are several good definitions for machine learning in the literature. Here are some representative examples:

"The field of pattern recognition is concerned with the automatic discovery of regularities in data through the use of computer algorithms and with the use of these regularities to take actions such as classifying the data into different categories" [1].

*"Vast amounts of data are being generated in many fields and the statistician's job is to make sense of it all: to extract important patterns and trends, and understand what the data says, we call this **learning from data**."* [3].

"An agent is learning if it improves its performance after making observations about the world... When the agent is a computer, we call it machine learning: a computer observes some data, builds a model based on the data, and uses the model as both a hypothesis about the world and a piece of software that can solve problems." [6].

A little bit of history

This is a timeline of some of the major events in machine learning and related fields, it helps us to gain some perspective:

- 1670 (?): Calculus

- 1805: Least squares
- 1812: Bayes theorem
- 1812: Computer
- 1847: Gradient descent
- 1950: Turing's learning machine
- 1951: First neural network
- 1952: Perceptron
- 1955: K-means
- 1959: Term "machine learning"
- 1963: Decision trees
- 1967: Nearest neighbor and limitations of neural nets (AI Winter)
- 1969: Marvin Minsky receives the Turing Award
- 1970: Backpropagation and graphical models
- 1971: John McCarthy receives the Turing Award
- 1975: Allen Newell and Herbert A. Simon receive the Turing Award
- 1980: First ICML
- 1986: First Neurips
- 1994: Edward Feigenbaum and Raj Reddy receive the Turing Award
- 1995: Random forest and SVMs
- 1995: Deep blue vs Kasparov
- 1999: First GPU
- 2009: Imagenet
- 2010: Leslie Valiant receives the Turing Award
- 2011: Judea Pearl receives the Turing Award
- 2013: Watson wins Jeopardy
- 2016: AlphaGo
- 2018: Hinton, Bengio, and LeCun receive the Turing Award
- 2020: AlphaFold

Applications

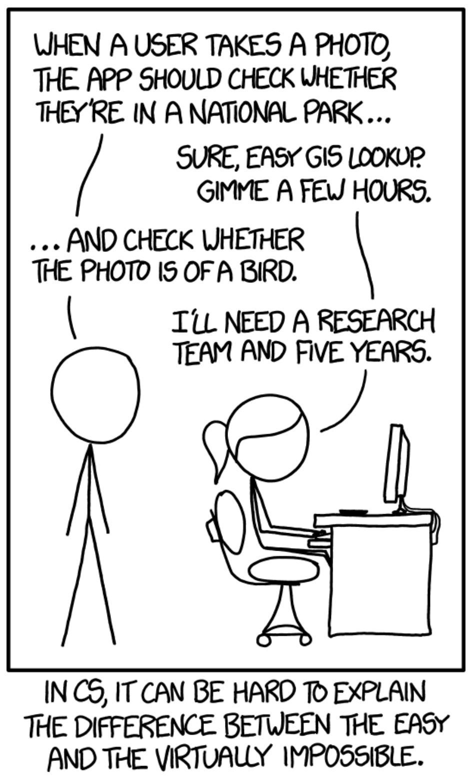
There are likely very few problems where machine learning has not been applied yet. Recently, a fully-connected feedforward network has "enabled" contributions in Topology and Representation Theory [2]. Here is a (non-exhaustive) list of a bit more mature ML applications:

- **Text:** Topic modeling, document classification, spam detection
- **Finance:** Fraud detection, algorithmic trading, credit scoring
- **Natural language processing:** Part-of speech tagging, named-entity recognition, context-free parsing
- **Speech processing:** speech recognition, speech synthesis, speaker identification
- **Computer vision:** object detection, pose estimation, image summarization
- **Computational biology:** protein function prediction, protein interaction prediction
- **Healthcare:** diagnosis, drug discovery, epidemiology
- **Recommender systems:** collaborative filtering, content-based filtering, cold-start
- **Autonomous control:** self-driving cars, bots, robots

Tasks

Most applications of ML fall into one of these tasks:

- **Classification:** Assigning a category with each item in a database
- **Regression:** Predicting a real value associated with each item in a database
- **Clustering:** Partitioning a database into sets of homogenous items
- **Dimensionality reduction:** Transforming an initial high-dimensional representation of a dataset into a low-dimensional one
- **Ranking:** Sorting a set of items according to some criteria



In the 60s, Marvin Minsky assigned a couple of undergrads to spend the summer programming a computer to use a camera to identify objects in a scene. He figured they'd have the problem solved by the end of the summer. Half a century later, we're still working on it. Source: `https://xkcd.com/1425/`

Key components

These are the fundamental components of most ML problems and their solutions:

- **Examples (or data):** Instances used for training or evaluation
- **Training sample:** Example used to train the model
- **Validation sample:** Example used to tune the parameters of the model
- **Test sample:** Example used to evaluate the model
- **Labels:** Values or categories of the examples
- **Hyperparameters:** Parameters that are not determined by the learning algorithm
- **Loss (or objective) function:** Measure of discrepancy between predicted and true labels (or some other measure of fitness)
- **Hypothesis:** Function that maps inputs (examples) to labels/predictions, part of a *hypothesis set*
- **Algorithm:** Applies hypothesis set and examples to minimize (optimize) loss function (objective)

Settings

Typical ML tasks assume one of the following settings:

- **Supervised learning:** Labeled examples (e.g. image classification on ImageNet)
- **Unsupervised learning:** Only unlabeled examples (e.g. clustering)
- **Semi-supervised learning:** Very few labeled examples (e.g. classification in most applications)
- **Transductive learning:** Labeled examples with unlabeled test instances known a priori (e.g. link prediction)
- **Online learning:** Training and testing phases are intermixed (e.g. stock market prediction)
- **Reinforcement learning:** Similar to online learning, but learning agent interacts with the environment via actions (e.g. game playing)
- **Active learning:** Learner selects training examples (e.g. human-in-the-loop settings)

Machine learning with graphs

How are graphs and machine learning related? And why there should be a course on this subject?

1. Graph data (e.g. social/information/biological networks);
2. Encoding structure in the data.

Non-linear dimensionality reduction

Consider the problem of *dimensionality reduction*, where the goal is to represent a dataset $X = [\mathbf{x}_1; \dots \mathbf{x}_n]$, where $\mathbf{x}_i \in \mathbb{R}^D$, into a lower-dimensional space $\mathbb{R}^{D'}$ ($D' << D$). The classical approach to solve this problem is *Principal Component Analysis (PCA)*. However, PCA is linear:

$$\mathbf{x}'_i = U^T(\mathbf{x}_i - \mu)$$

where μ is the mean value of the dataset and $U \in \mathbb{R}^{D \times D'}$ is the matrix with top D' eigenvectors of the covariance matrix $(X - \mu)(X - \mu)^T$.

Isomap is a non-linear graph-based alternative to PCA [4]. The algorithm works as follows:

1. Compute the pairwise distance $d_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||_2$;
2. Build a graph G connecting each point i to its k nearest neighbors with weights d_{ij};
3. Compute shortest-path distances d'_{ij} between all pairs of vertices in G;
4. *Embed* shortest-path matrix into X'

The embedding maps a shortest-path matrix $M \in \mathbb{R}^{n \times n}$ to new data representations X' and is performed using *multidimensional scaling*. A more formal introduction to Isomap will be given later in the course.

References

[1] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[2] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, et al. Advancing mathematics by guiding human intuition with ai. *Nature*, 600(7887):70–74, 2021.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning*. Springer, 2009.

[4] John A Lee and Michel Verleysen. *Nonlinear dimensionality reduction.* Springer Science & Business Media, 2007.

[5] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning.* MIT press, 2018.

[6] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.* Pearson, 2021.