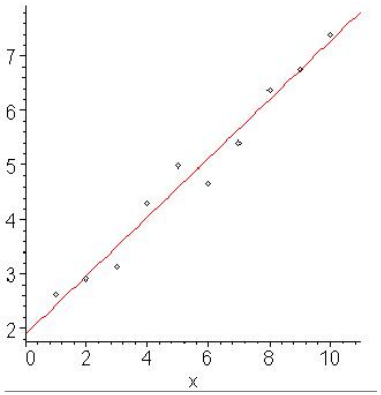
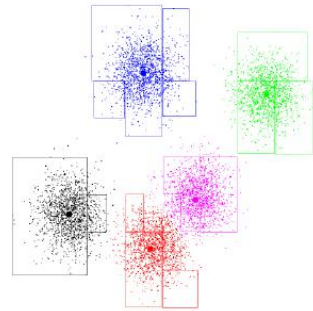
Machine Learning

Introduction



Marcello Restelli

March 3, 2020



Outline

Admin

- **Instructor**: Marcello Restelli – marcello.restelli@polimi.it
- **Teaching assistant**: Francesco Trovò – francesco1.trovo@polimi.it
- Class Website on **BeeP**: https://beep.metid.polimi.it/web/2016-17-machine-learning-marcello-restelli-/
- Assessment
  - **Written exam**
- Thesis (only a few)
  - Next week...

Relations to Other Courses

- A course of 5 credits is **not enough** to cover the main aspects of Machine Learning
- Fortunately, there are **other courses** that deal with some machine learning topics:
  - Data Mining and Text Mining
  - Artificial Neural Networks and Deep Learning
  - Soft Computing
  - Applied Statistics
  - Model Identification and Data Analysis

Practical classes

- For each topic there will be **practical classes**
- In these classes Francesco will present
  - Exercises similar to the ones you will find in the **exam**
  - Practical exercises using **Matlab**
- We suggest to bring **your laptop**

Schedule: first part

03-mar-2019	Introduction	Restelli	Bishop, Ch. 1, 2
05-mar-2019	Matlab	Trovò	
10-Mar-2019	Linear regression	Restelli	Bishop, Ch. 3.1, 3.2, 3.3 Bishop, Ch. 4.1.1, 4.1.2
12-Mar-2019	Linear regression	Restelli	4.1.3, 4.1.7, 4.3.1, 4.3.2
17-Mar-2019	Liner classification	Restelli	Bishop, Ch. 3.2, 1.3
19-Mar-2019	Ex. on linear regression	Trovò	
24-Mar-2019	Bias-Variance	Restelli	Bishop, Ch. 12.1, 14.2, 14.3
26-Mar-2019	Ex. on linear classification	Trovò	
31-Mar-2019	Model Selection	Restelli	Mitchell, Ch. 7.1, 7.2, 7.3
02-Apr-2019	PAC-Learning and VC dimension	Restelli	Mitchell, Ch. 7.4
16-Apr-2019	Ex. on learning theory	Trovò	
21-Apr-2019	Kernel Methods	Restelli	Bishop, Ch. 6.1, 6.2
23-Apr-2019	Gaussian Processes	Restelli	Bishop, Ch. 6.4
28-Apr-2019	Support Vector Machines	Restelli	Bishop, Ch. 7.1.1, 7.1.2
30-Apr-2019	Ex. on Gaussian Processes	Trovò	
05-May-2019	Ex. on Support Vector Machines	Trovò	

Schedule: second part

07-May-2019	Markov Decision Processes	Restelli	Sutton&Barto, Ch. 1, 2, 3
12-May-2019	Dynamic Programming	Restelli	Sutton&Barto, Ch. 4
14-May-2019	RL in finite MDPs	Restelli	Sutton&Barto, Ch. 5, 6
19-May-2019	Ex. on Dynamic Programming	Trovò	
21-May-2019	RL in finite MDPs	Restelli	Sutton&Barto, Ch. 7
26-May-2019	RL in finite MDPs	Restelli	Sutton&Barto, Ch. 8
28-May-2019	Multi-armed bandit	Trovò	
04-Jun-2019	Ex. on multi-armed bandit	Trovò	
??-Jun-2019	Ex. on RL in finte MDPs	Trovò	

Textbooks

- Supervised Learning
  - **Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.**
  - Mitchell, "Machine Learning", McGraw Hill, 1997.
  - Hastie, Tibshirani, Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer, 2009.
- Reinforcement Learning
  - **Sutton and Barto, "Reinforcement Learning: an Introduction", MIT Press, 1998.** New edition (2018) available at: http://www.incompleteideas.net/book/the-book-2nd.html
  - Buşoniu, Babuška, De Schutter and Ernst, "Reinforcement Learning and Dynamic Programming Using Function Approximators", CRC Press, 2010.
  - Szepesvari, "Algorithms for Reinforcement Learning", Morgan and Claypool, 2010.
  - Bertsekas and Tsitsiklis, "Neuro–Dynamic Programming", Athena Scientific, 1996.

Course Goals

- Learn to correctly **model** machine learning problems
- Learn the **principles** of ML and the main techniques
- Learn to **apply** ML to practical problems
- Learn **limitations** of ML techniques
- Provide the basic background to do **research** in this field
- My expectations
  - ask questions
  - interact
  - get involved

What is Machine Learning?

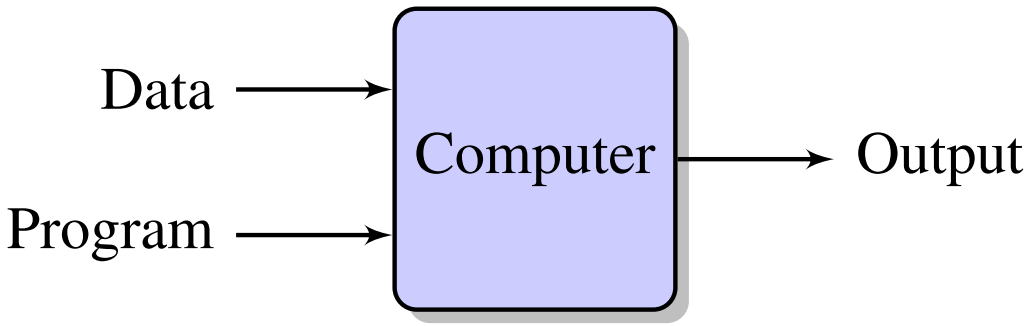
- The real question is: **what is learning?**
- Mitchell (1997):

  "A computer program is said to
  learn from ***experience E***
  with respect to some class of ***tasks T***
  and ***performance measure P,***
  improves with experience E"

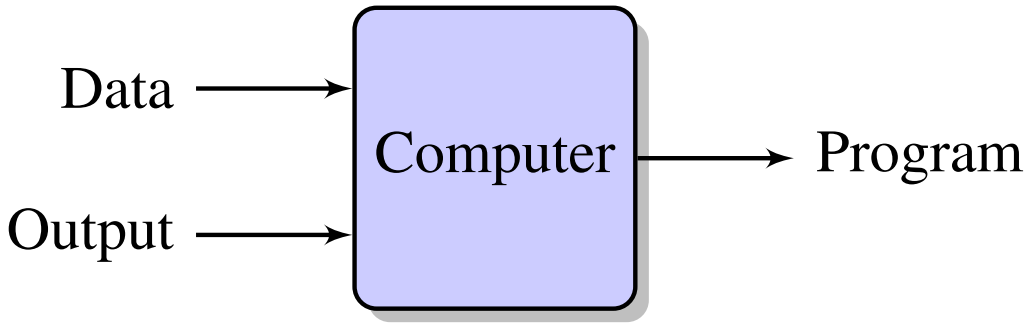
- ML is the sub-field of AI where the **knowledge** comes from:
  - **Experience**
  - **Induction**
- Machine learning **is not magic!**
  - You need to know **how it works**
  - You need to know **how to use it**
  - It can **extract** information from data, **not create** information

Traditional Programming vs ML

Tradition Programming



Machine Learning



Why Machine Learning?

- We need computers to **make informed decisions** on new, unseen data
- Often it is too difficult to design a set of rules **"by hand"**
- Machine learning allows to **automatically extract relevant information** from data applying it to analyze new data
- **Automating automation**
- Getting computers to **program themselves**
  - Writing software is the **bottleneck**
  - let the **data** do the work instead!

What is ML useful for?

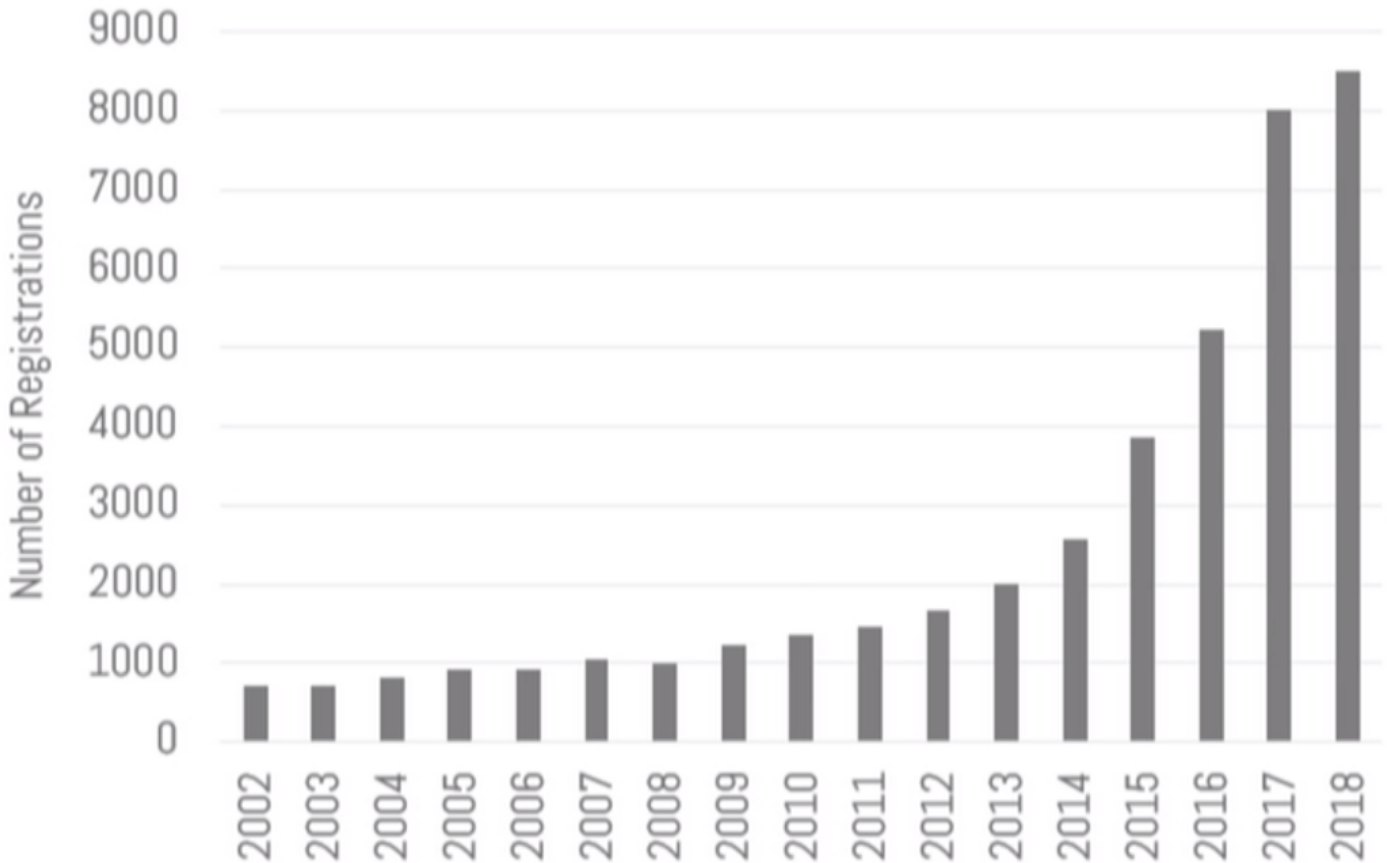
- These are **exciting times** for ML
- ML is becoming **widespread**
  - Computer vision and robotics
  - Speech recognition
  - Biology and medicine
  - Finance
  - Information retrieval, Web search, ...
  - Video gaming
  - Space exploration
  - Many application and many jobs...

A few quotes

- “A breakthrough in machine learning would be worth ten Microsofts” (Bill Gates, Chariman, Microsoft)
- “Machine learning is the next Internet” (Tony Tether, Director, DARPA)
- “Machine learning is the hot new things” (John Hennessy, President, Stanford)
- “Web rankings today are mostly a matter of machine learning” (Prabhakar Raghavan, Dir. Research Yahoo)
- “Machine learning is going to result in a real revolution” (Greg Papadopolus, Former CTO, Sun)
- “Machine learning is today’s discontinuity (Jerry Yang, Founder, Yahoo)
- ”Machine learning today is one of the hottest aspects of computer science“ (Steve Ballmer, CEO, Microsoft)

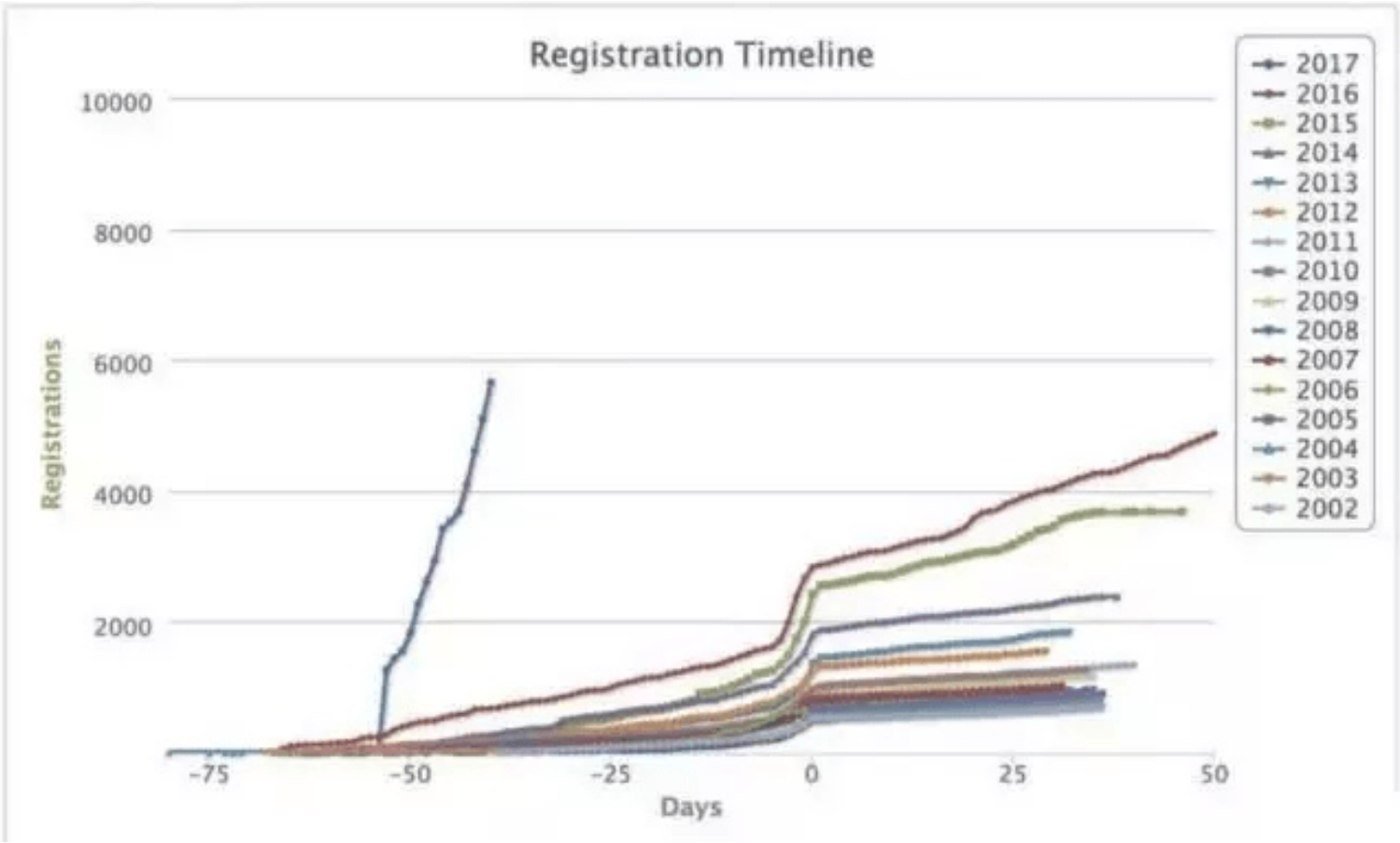
The Machine Learning Growth

NeurIPS attendence



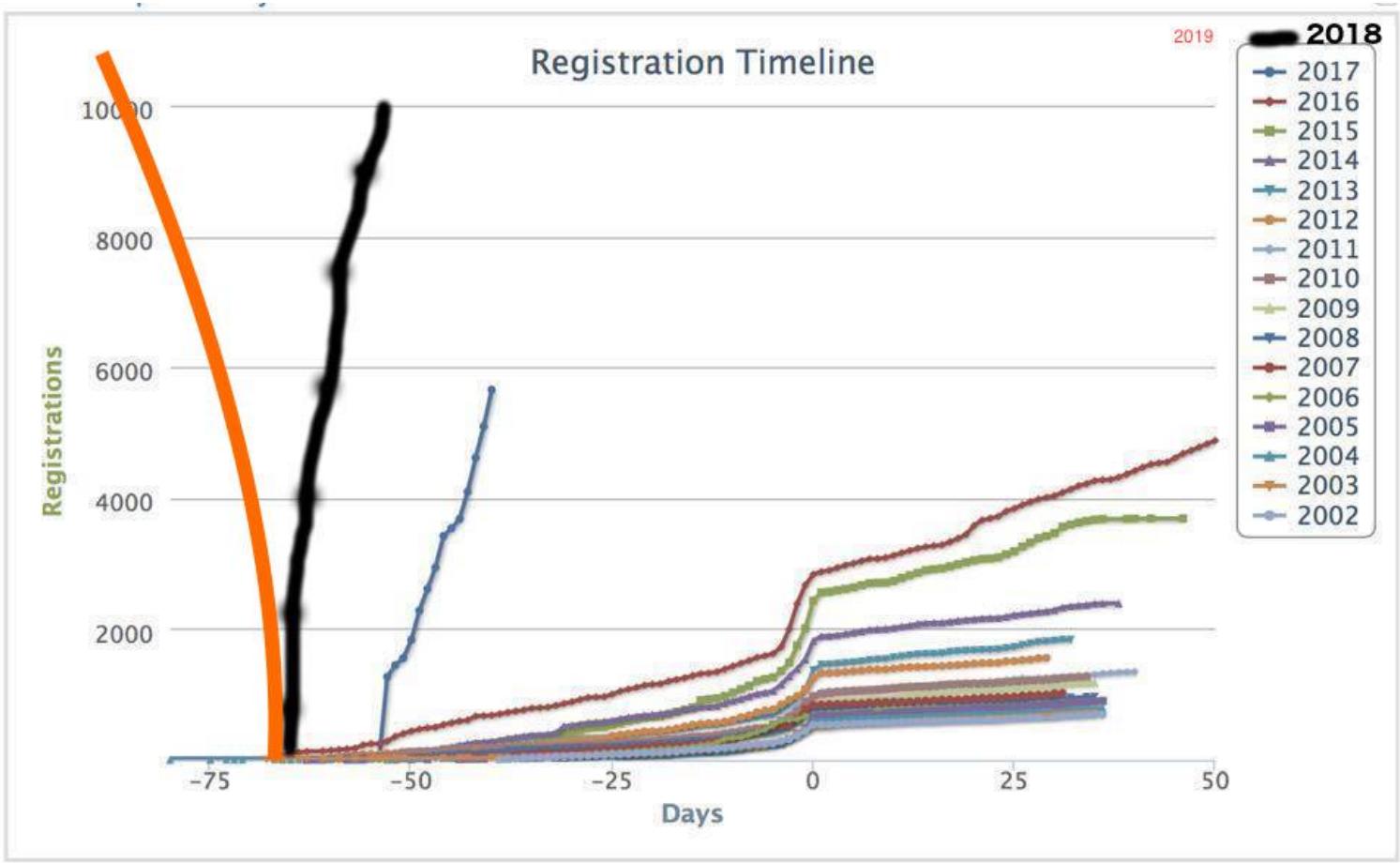
The Machine Learning Growth

NeurIPS attendence



The Machine Learning Growth

NeurIPS attendence



The Machine Learning Growth

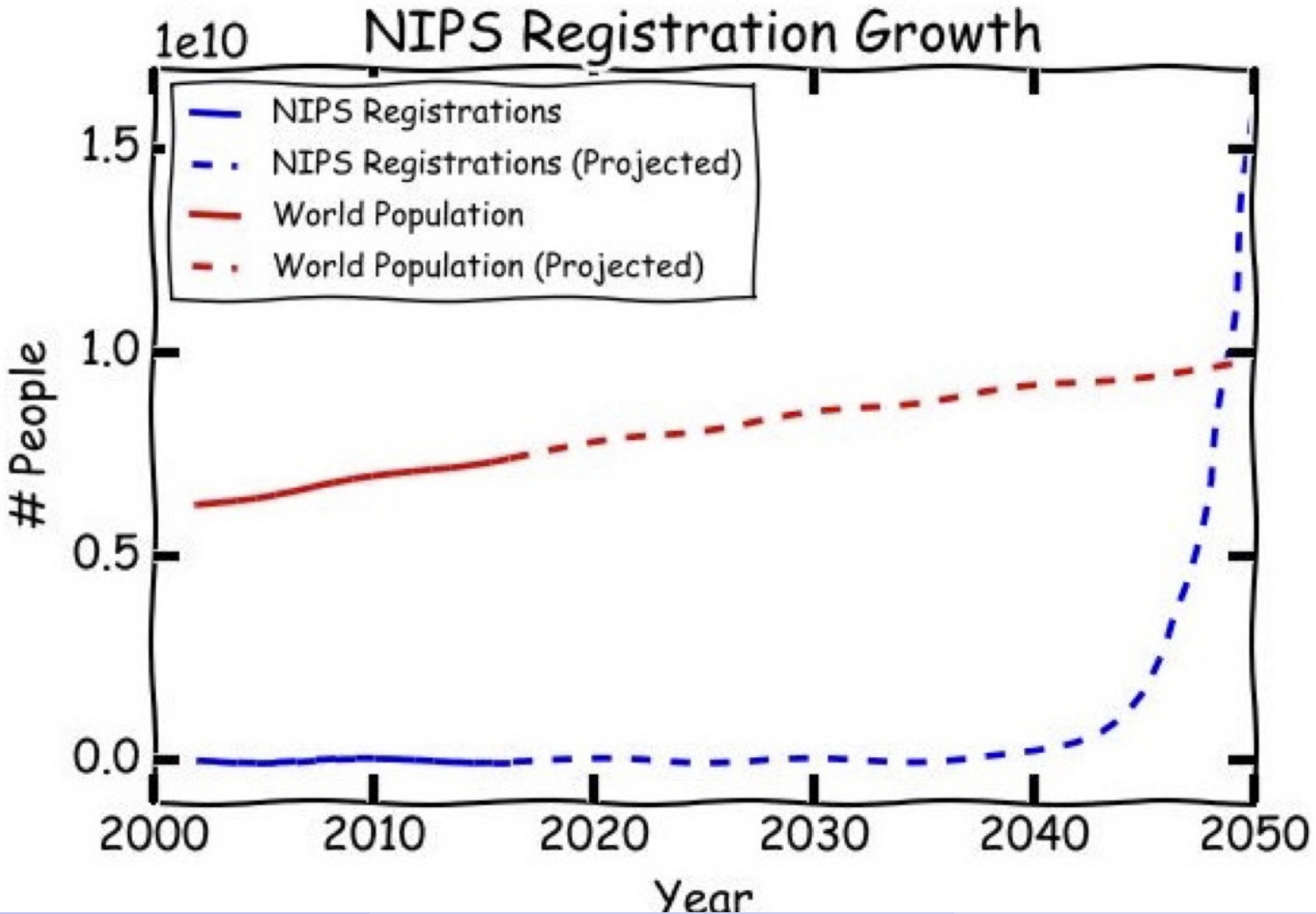
NeurIPS attendence

NeurIPS 2018 soldout time

11’38’’

The Machine Learning Growth

NeurIPS attendence



The Machine Learning Growth

NeurIPS attendence

What about 2019?

The Machine Learning Growth

NeurIPS attendence

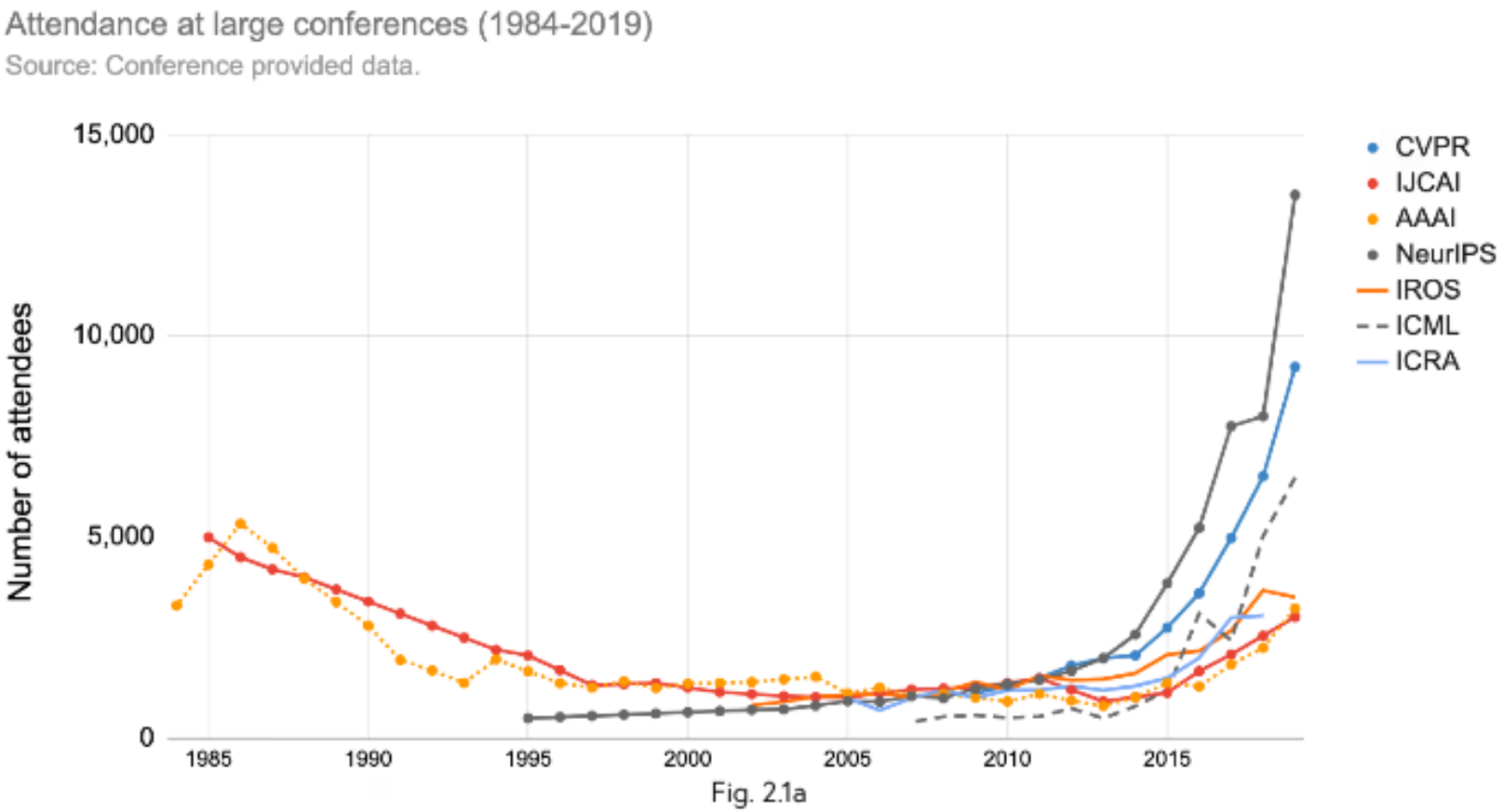
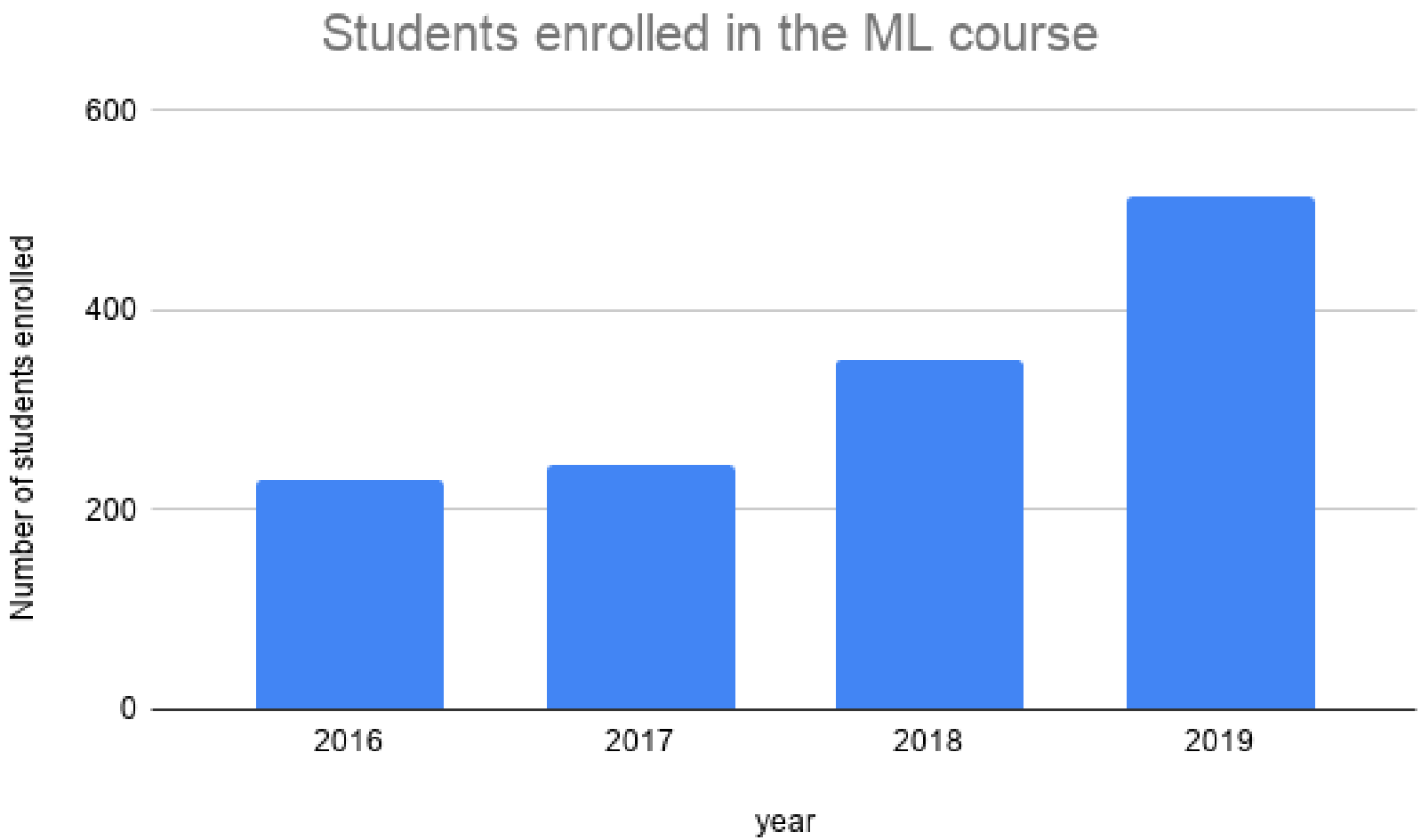


Fig. 2.1a

The Machine Learning Growth

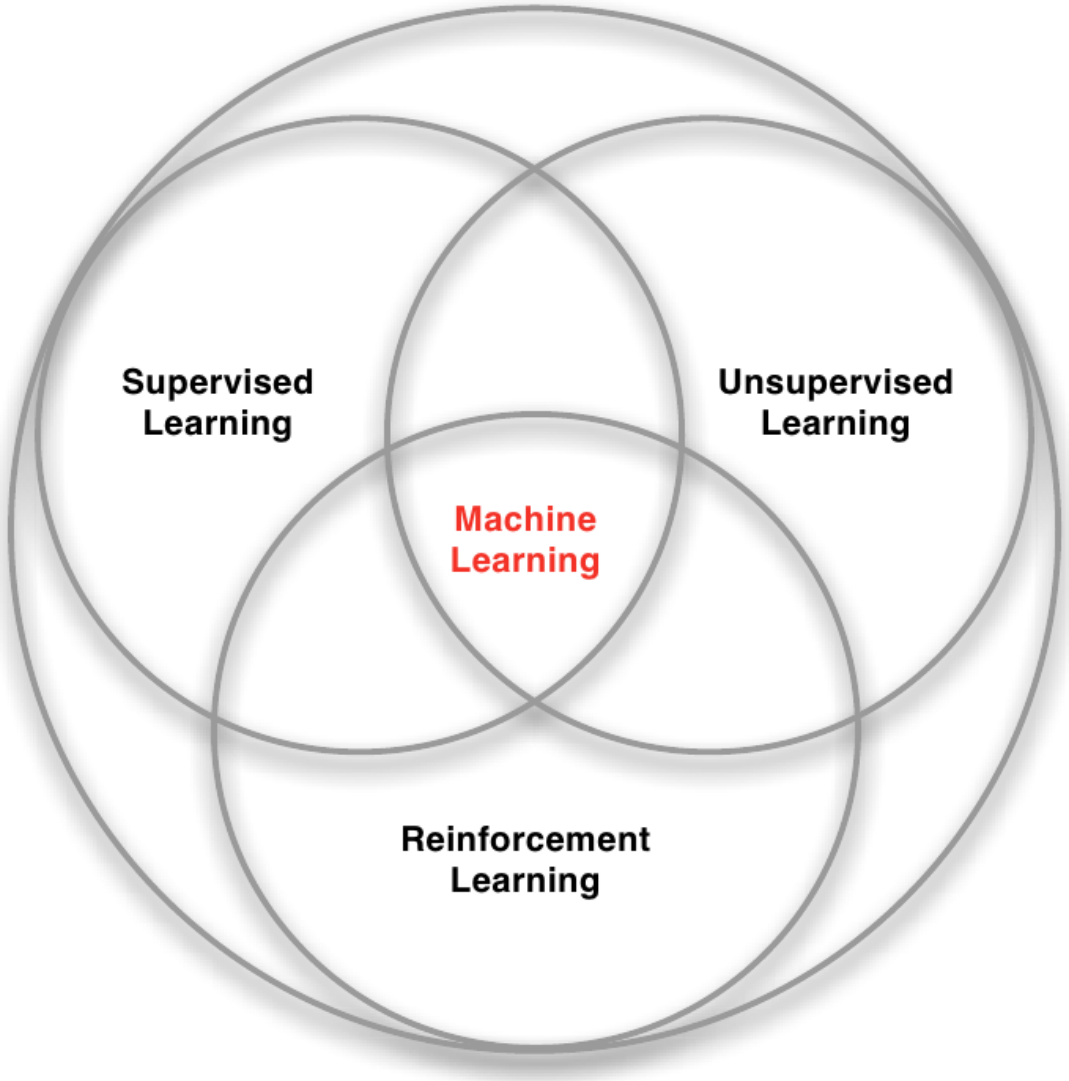
ML course attendance



ML Top Venues

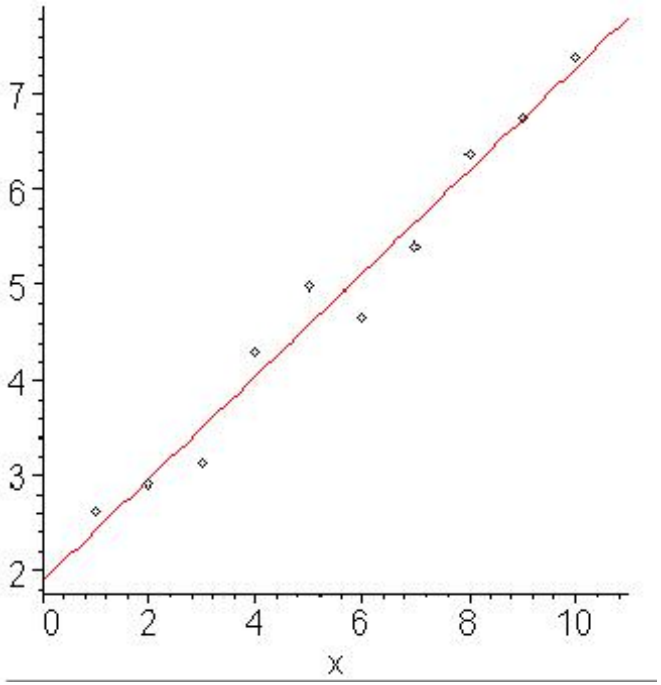
- Journals
  - Journal of Machine Learning Research (JMLR)
  - Machine Learning Journal (MLJ)
  - Journal of Artificial Intelligence Research (JAIR)
- Conferences
  - International Conference on Machine Learning (ICML)
  - Neural Information and Processing Systems (NIPS)
  - American Association on Artificial Intelligence (AAAI)
  - International Joint Conference on Artificial Intelligence (IJCAI)
  - International Conference on Learning Representations (ICLR)
  - Uncertainty in Artificial Intelligence (UAI)
  - Artificial Intelligence and Statistics (AI&Stats)
  - Conference on Learning Theory (CoLT)

Machine Learning



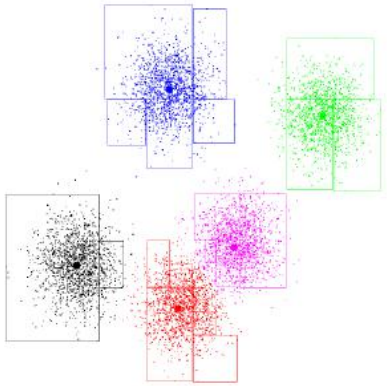
Machine Learning Models

- Supervised Learning
  - Learn the model
- Unsupervised Learning
  - Learn the representation
- Reinforcement Learning
  - Learn to control



Machine Learning Models

- Supervised Learning
  - Learn the model
- Unsupervised Learning
  - Learn the representation
- Reinforcement Learning
  - Learn to control



Machine Learning Models

- Supervised Learning
  - Learn the model
- Unsupervised Learning
  - Learn the representation
- Reinforcement Learning
  - Learn to control

Supervised Learning

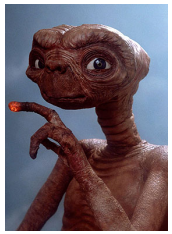
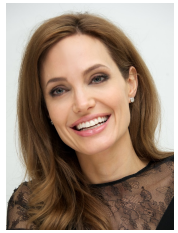
- Goal
  - Estimating the **unknown model** that maps **known inputs** to **known outputs**
  - Training set: $\mathcal{D} = \{\langle x, t \rangle\} \Rightarrow t = f(x)$
- Problems
  - Classification
  - Regression
  - Probability estimation
- Techniques
  - Artificial Neural Networks
  - Support Vector Machines
  - Decision trees
  - Etc.

Supervised Learning: Classification Example

Training

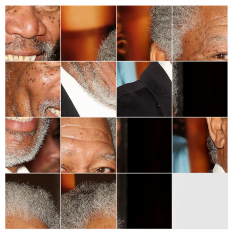
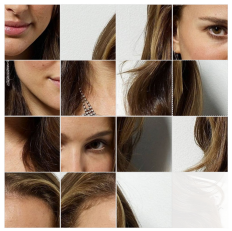
Input	Output
	0
	1
	1
	0

Testing

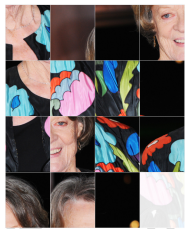
Input	Output
	0
	0
	1
	0

Supervised Learning: Regression Example

Training

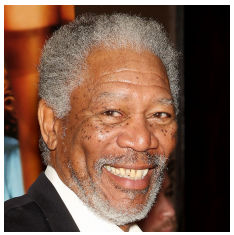
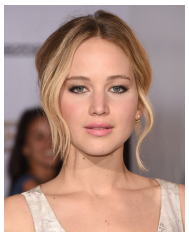
Input	Output
	78
	34
	25

Testing

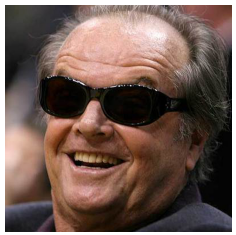
Input	Output
	78
	81
	41

Supervised Learning: Regression Example

Training

Input	Output
	78
	34
	25

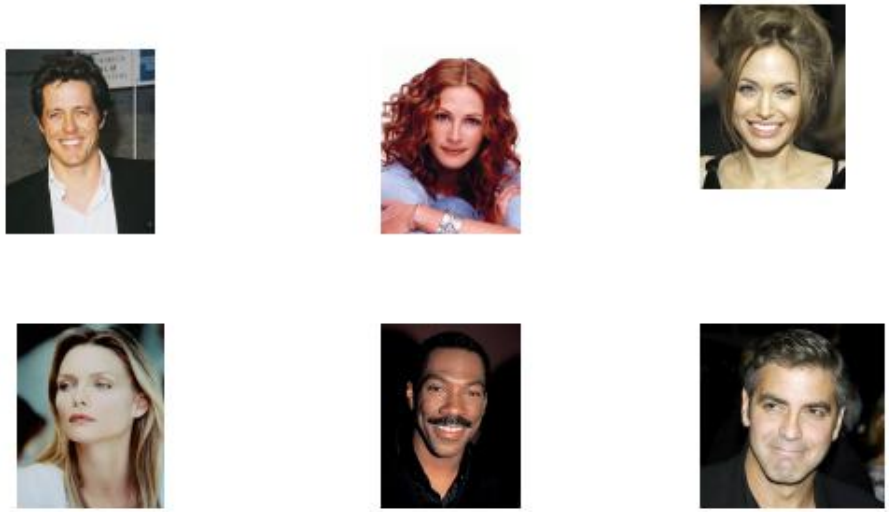
Testing

Input	Output
	78
	81
	41

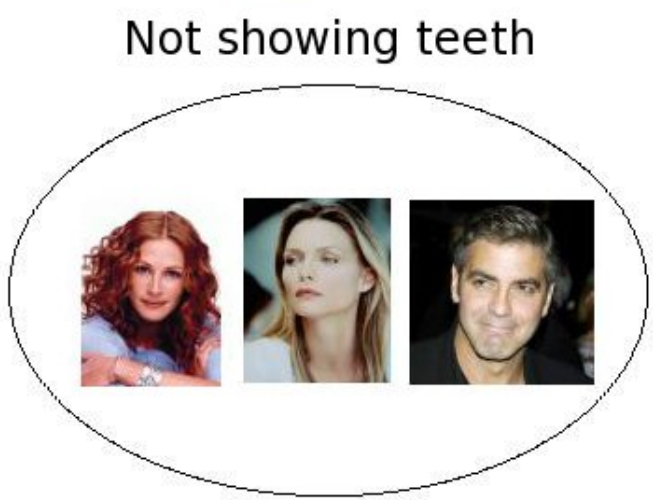
Unsupervised Learning

- Goal
  - Learning a more efficient representation of a set of unknown inputs
  - Training set: $\mathcal{D} = \{x\} \Rightarrow ? = f(x)$
- Problems
  - Compression
  - Clustering
- Techniques
  - K-means
  - Self-organizing maps
  - Principal Component Analysis
  - Etc.

Unsupervised Learning: Clustering Example

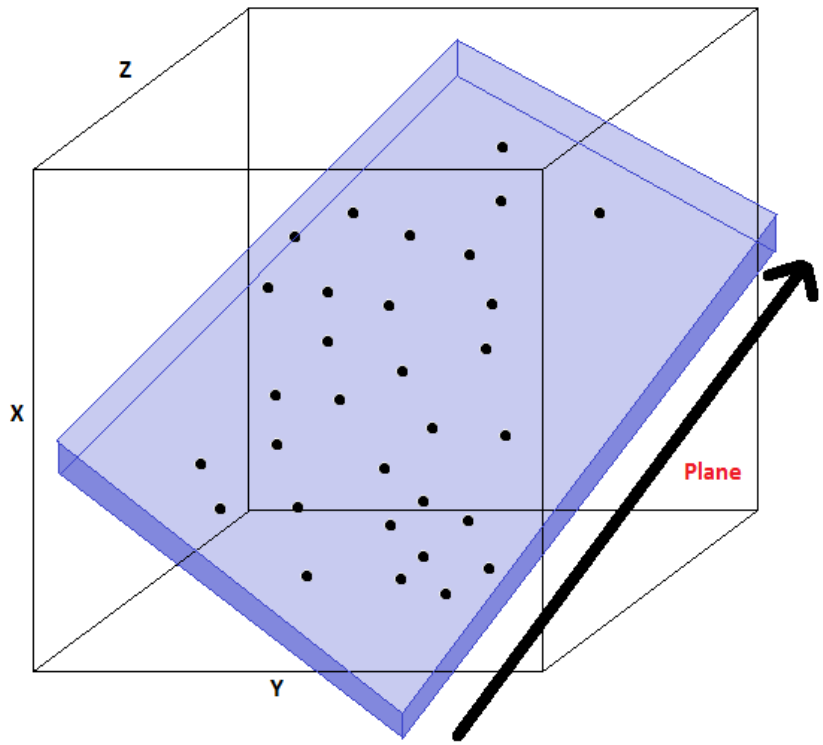


Unsupervised Learning: Clustering Example



Unsupervised Learning: Dimensionality Reduction Example

X	Y	W	Z
2	3	1	10
5	8	1	2
7	2	1	6
9	8	1	-2
8	1	1	6
4	10	1	1
8	8	1	-1
8	1	1	6
4	5	1	6
3	10	1	2



- W is useless
- Actually the points lie on a 2-dimensional plane

Reinforcement Learning

- Goal
  - Learning the **optimal policy**
  - Training set: $\mathcal{D} = \{\langle x, u, x', r\rangle\} \Rightarrow \pi^*(x) = arg\max_u\{Q^*(x, u)\}$, where $Q^*(x, u)$ must be estimated.
- Problems
  - Markov Decision Process (MDP)
  - Partially Observable MDP (POMDP)
  - Stochastic Games (SG)
- Techniques
  - Q-learning
  - SARSA
  - Fitted Q-iteration
  - Etc.

Reinforcement Learning: Example

But Who's Counting?

But Who's Counting?

- First game
  - Best possible value: 75421
  - Value following the optimal policy: 75142
- Second game
  - Best possible value: 76530
  - Value following the optimal policy: 75630

Supervised Learning

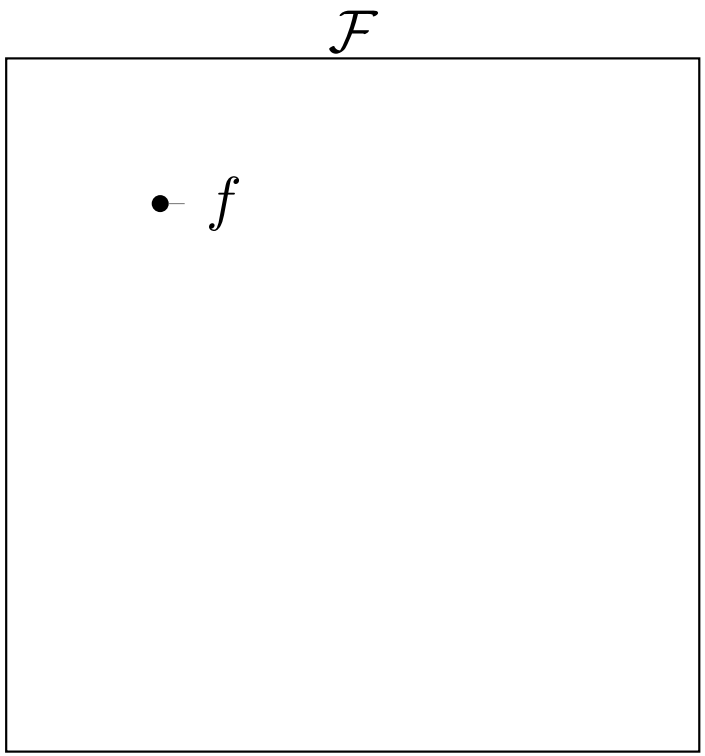
- Supervised (inductive) learning is the **largest**, **most mature**, **most widely used** sub-field of machine learning
- **Given**: training data set including **desired outputs**: $\mathcal{D} = \{\langle x, t \rangle\}$ from some **unknown** function f
- **Find**: A **good approximation** of f that **generalizes** well on **test** data
- **Input variables** x are also called **features**, **predictors**, **attributes**
- **Output variables** t are also called **targets**, **responses**, **labels**
  - If t is discrete: classification
  - if t is continuous: regression
  - if t is the probability of x: probability estimation

Appropriate applications

- There is **no human expert**
  - e.g., DNA analysis
- Humans can perform the task but **cannot explain how**
  - e.g., character recognition
- Desired function **changes frequently**
  - e.g., predicting stock prices based on recent trading data
- Each user needs a **customized** function f
  - e.g., email filtering

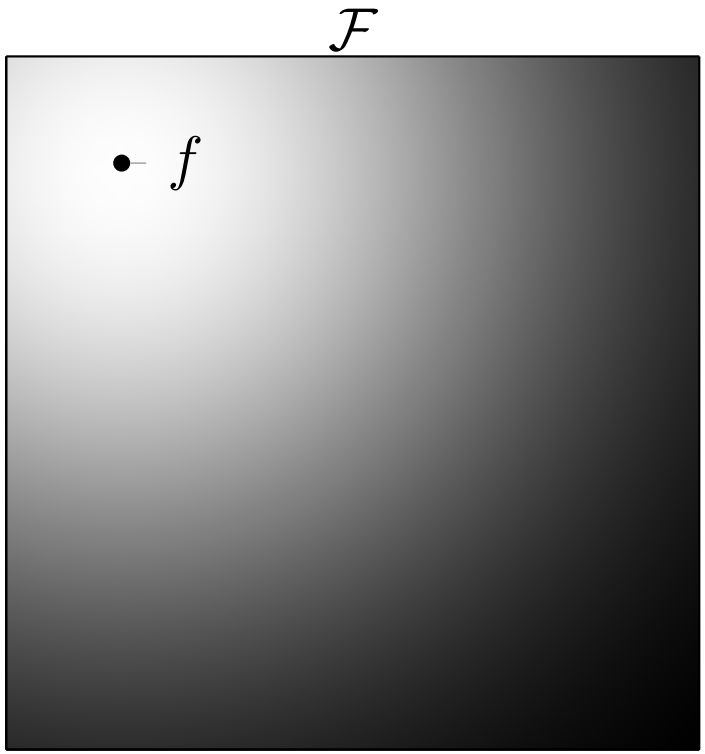
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



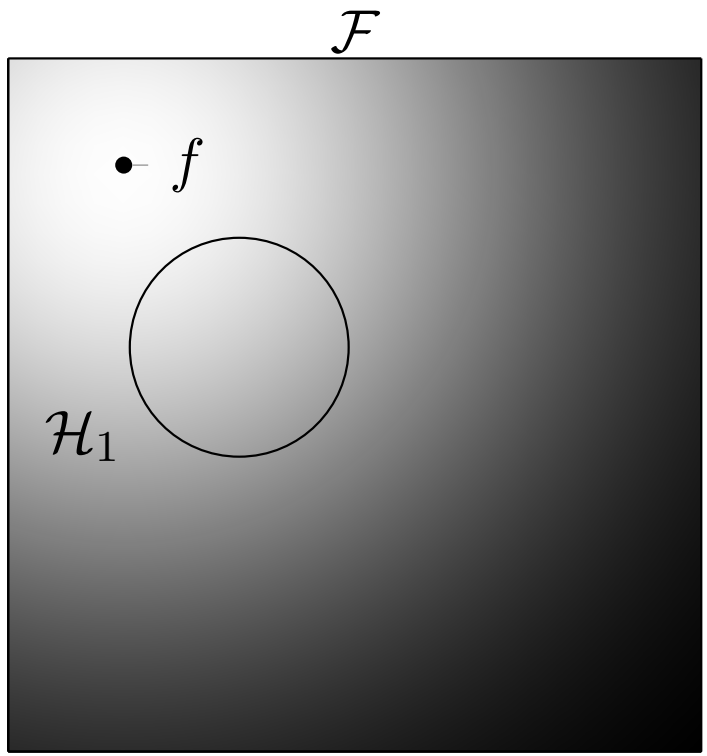
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



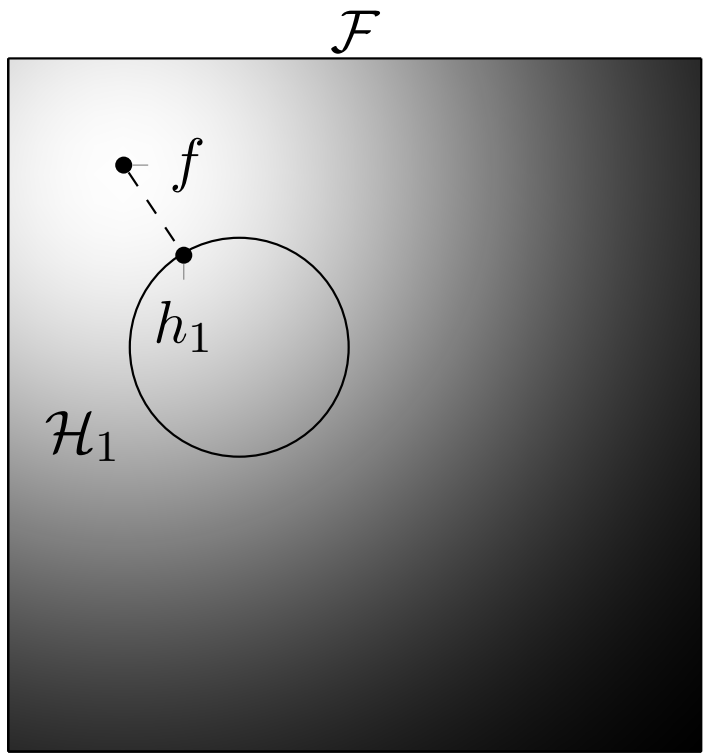
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



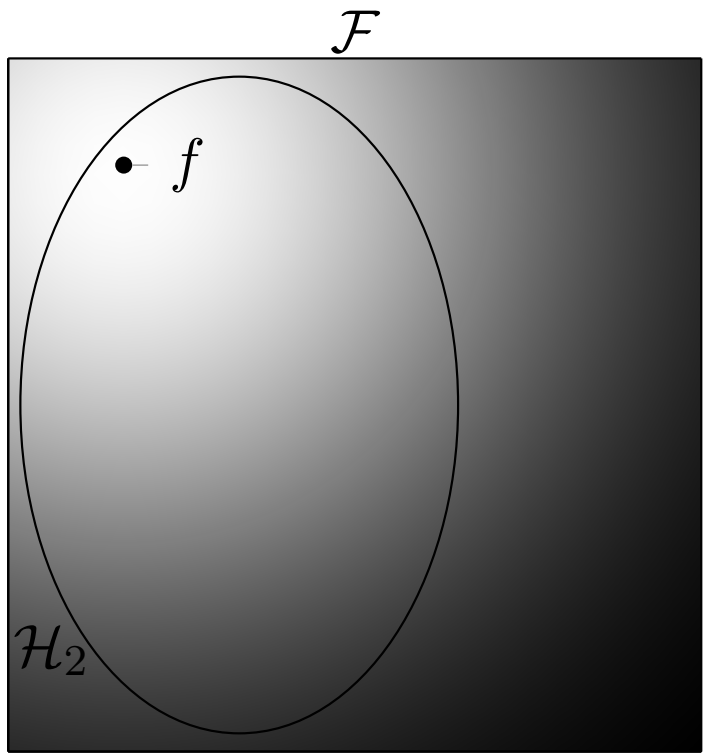
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



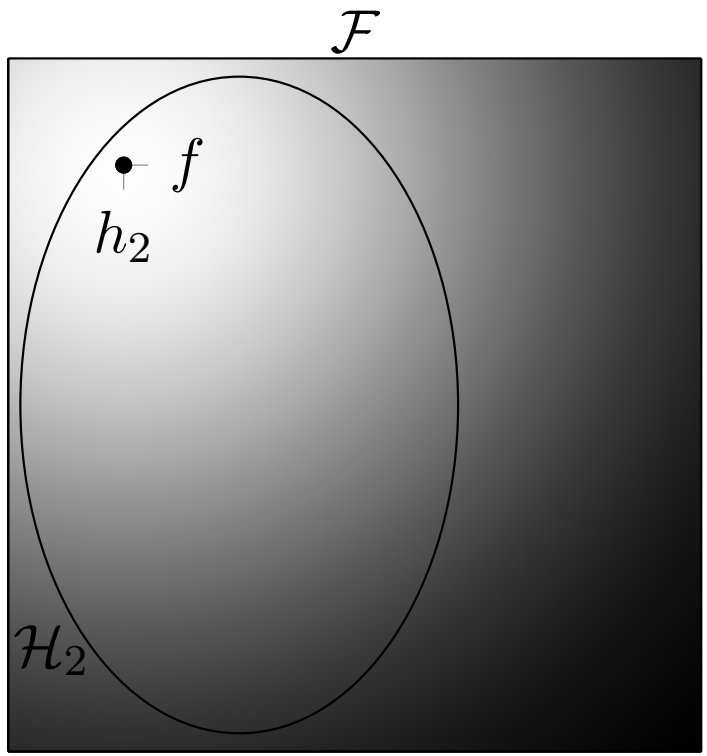
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



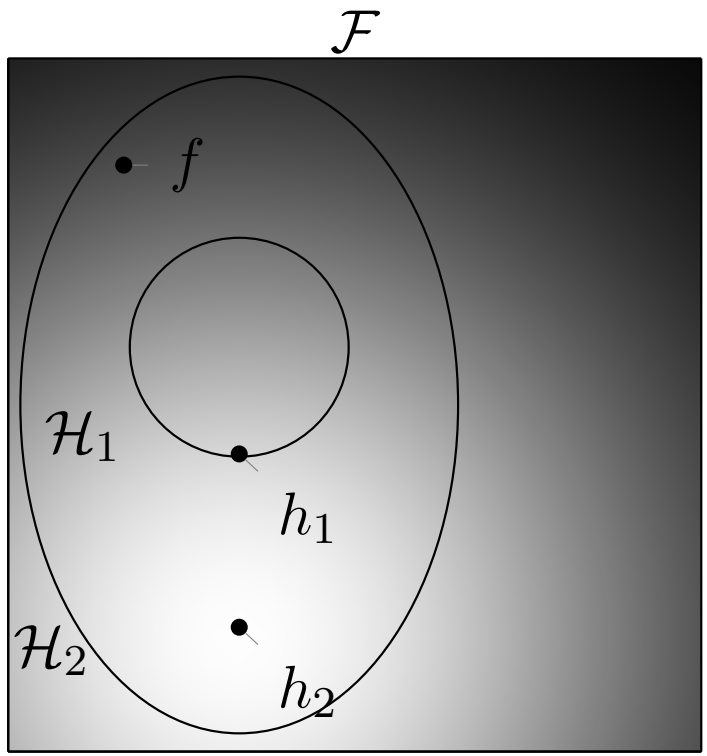
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



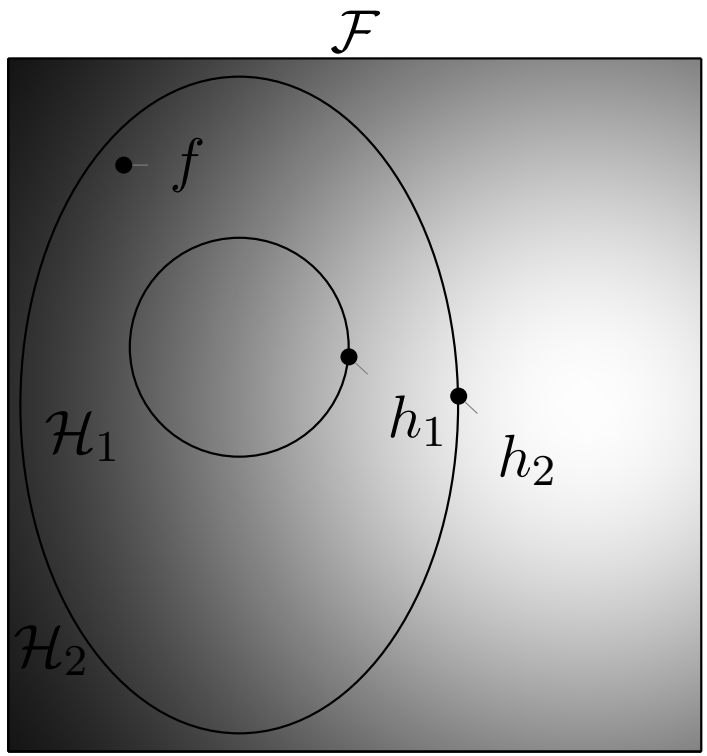
Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



Overview of Supervised Learning

- We want to **approximate** f given the data set $\mathcal{D}$
- The steps are
  1. Define a **loss function** L
  2. Choose some **hypothesis space** $\mathcal{H}$
  3. **Optimize** to find an approximate model h
- What happens if we **enlarge** the hypothesis space?
- We do not know f and we have only a **finite number of samples**



Key Elements of Supervised Learning

- **Ten of thousands** of machine learning algorithms
- Hundreds new **every year**
- Every machine learning algorithm has three Components;
  - **Representation**
  - **Evaluation**
  - **Optimization**

Representation

- Linear models
- Instance-based
- Decision trees
- Set of rules
- Graphical models
- Neural networks
- Gaussian Processes
- Support vector machines
- Model ensembles
- etc.

Representation

- **Linear models**
- **Instance-based**
- Decision trees
- Set of rules
- *Graphical models*
- *Neural networks*
- **Gaussian Processes**
- **Support vector machines**
- **Model ensembles**
- etc.

Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost/Utility
- Margin
- Entropy
- KL divergence
- Etc.

Optimization

- Combinatorial optimization
  - e.g.: Greedy search
- Convex optimization
  - e.g.: Gradient descent
- Constrained optimization
  - e.g.: Linear programming

Dichotomies in ML

- Parametric vs Nonparametric
  - Parametric: **fixed and finite** number of parameters
  - Nonparametric: the number of parameters depends on the **training set**
- Frequentist vs Bayesian
  - Frequentist: use probabilities to model the **sampling** process
  - Bayesian: use probability to **model uncertainty** about the estimate
- Generative vs Discriminative
  - Generative: Learns the **joint** probability distribution $p(x, t)$
  - Discriminative: Learns the **conditional** probability distribution $p(t|x)$
- Empirical Risk Minimization vs Structural Risk Minimization
  - Empirical Risk: Error over the **training set**
  - Structural Risk: Balance training error with **model complexity**