

UVA CS 4774: Machine Learning

S6: Lecture 27: Review

Dr. Yanjun Qi

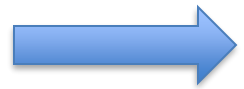
University of Virginia
Department of Computer Science

Disclaimer: it is quite hard to make important topics of machine learning fit on a one-semester schedule.

Disclaimer: it is quite hard to make important topics of deep learning fit on a one-session schedule.

We aim to make the content reasonably digestible in an introductory manner. We try to focus on a modularity view by introducing important variables to digest machine learning into chunks regarding data/representation / loss-functions / optimizations / model characteristics. That said, our goals here are to highlight the most foundational design choices in machine learning about algorithm designs, workflows, what to learn and how to learn it, and to expose the trade-offs in those choices. We think this teaching style provides students with context concerning those choices and helps them build a much deeper understanding.

Final Review

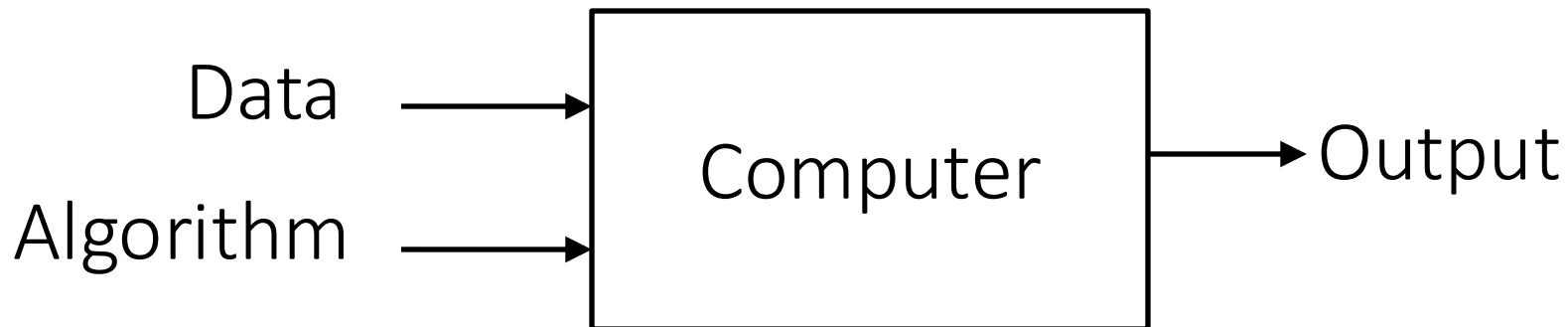


- ❑ Review of covered so far
- ❑ Five Tribes of Machine Learning

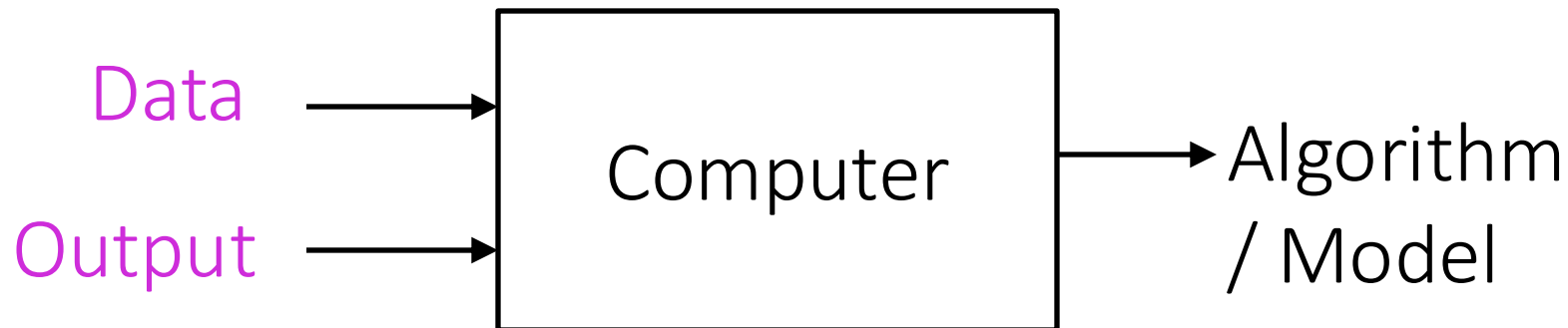
Objective

- To help students be able to build machine learning tools
 - (not just a tool user!!!)
- Key Results:
 - Able to build a few simple machine learning methods from scratch
 - Able to understand a few complex machine learning methods at the source code and equation level

Traditional Programming



Machine Learning



Digital and Physical: Lots of Data

Almost all aspects of
planet earth go digital NOW

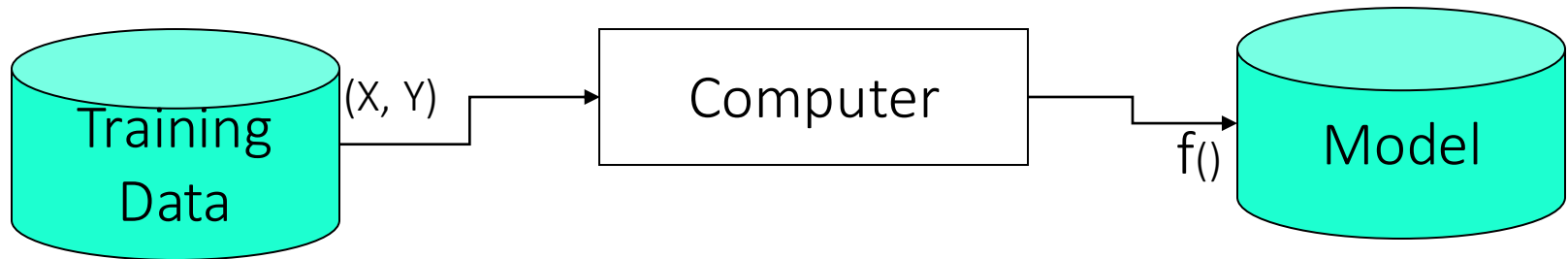
➔ Accessible and Large Amount of
Data Samples, Streams, ...



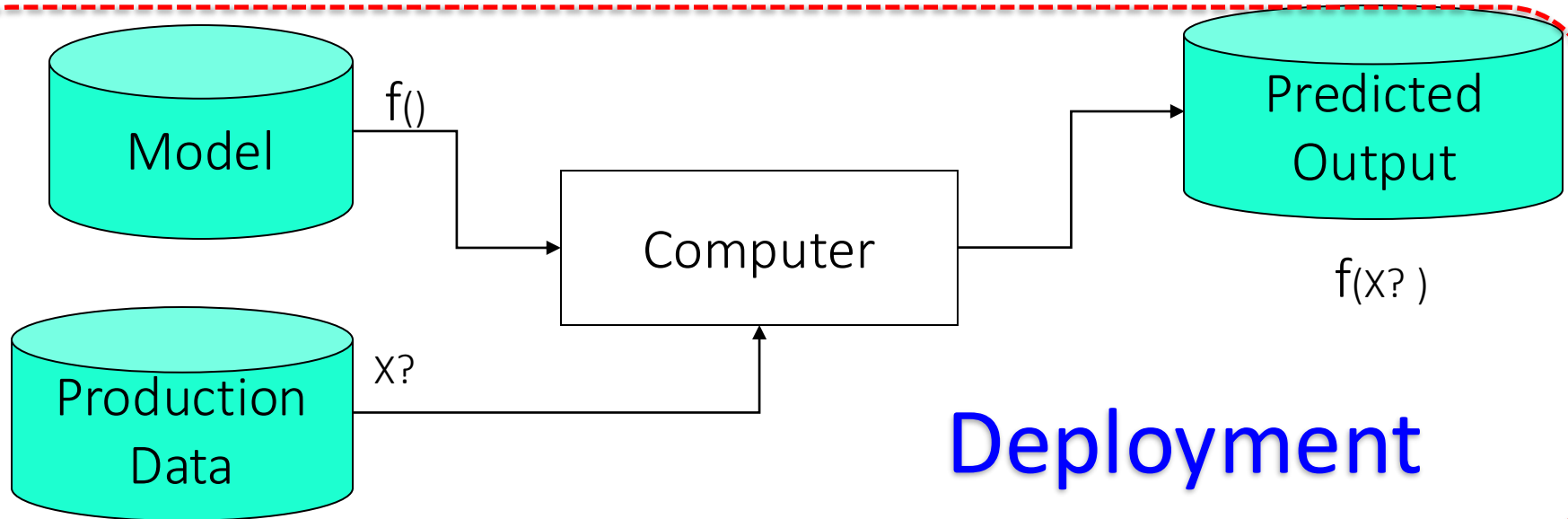
Two Modes of Machine Learning

Consists of **input-output** pairs

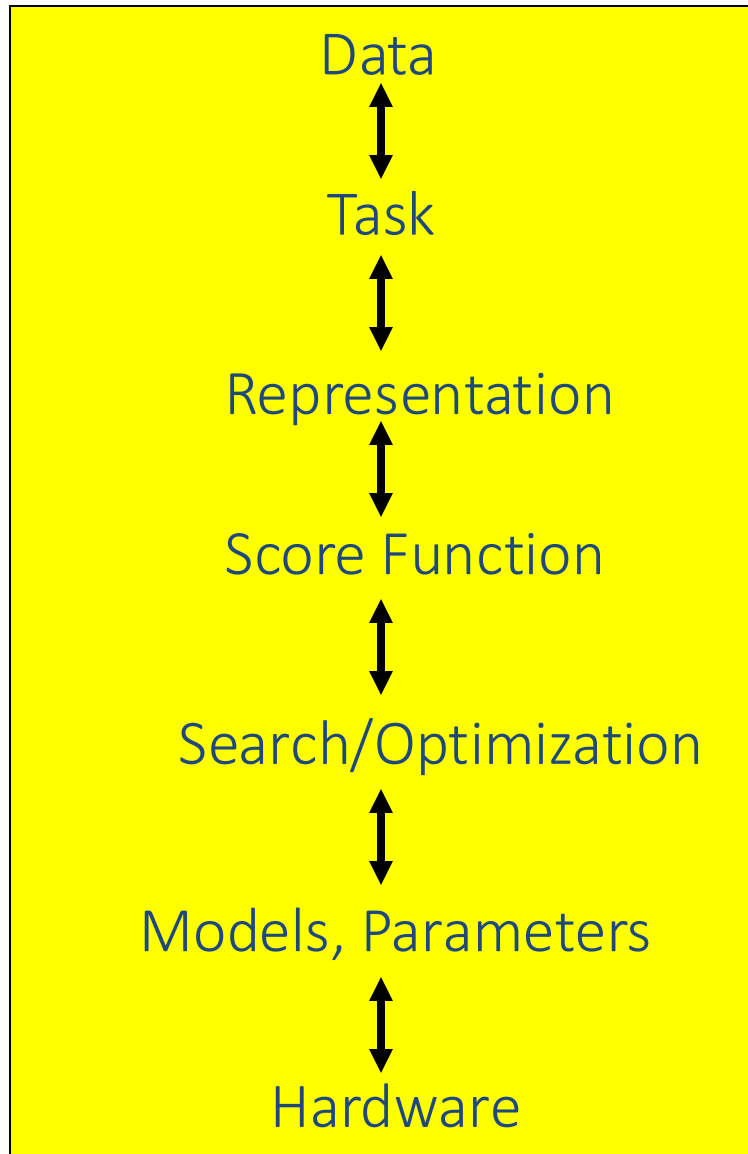
Training



Deployment



Machine Learning in a Nutshell



ML grew out of
work in AI

Optimize a
performance criterion
using example data or
past experience,

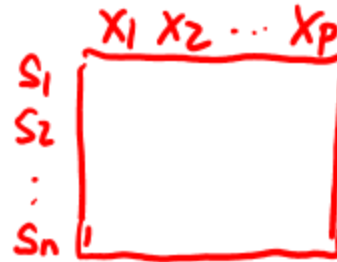
Aiming to generalize to
unseen data

Rough Sectioning of this Course

- S1. Basic Supervised Regression + Tabular Data
- S2. Basic Deep Learning + 2D Imaging Data
- S3. Generative and Deep + 1D Sequence Text Data
- S4. Advanced Supervised learning + Tabular Data
- S5. Not Supervised
- S6: Wrap Up + (a few invited tasks, e.g. tools)

Course Content Plan → Regarding Data

❑ Tabular / Matrix



❑ 2D Grid Structured: Imaging



❑ 1D Sequential Structured: Text

❑ Graph Structured (Relational)

❑ Set Structured / 3D /

Course Content Plan → Regarding Tasks

❑ Regression (supervised)

Y is a continuous

❑ Learning theory

About $f()$

❑ Classification (supervised)

Y is a discrete

❑ Unsupervised models

NO Y

❑ Graphical models

About interactions among $Y, X_1, \dots, X_p$

❑ Reinforcement Learning

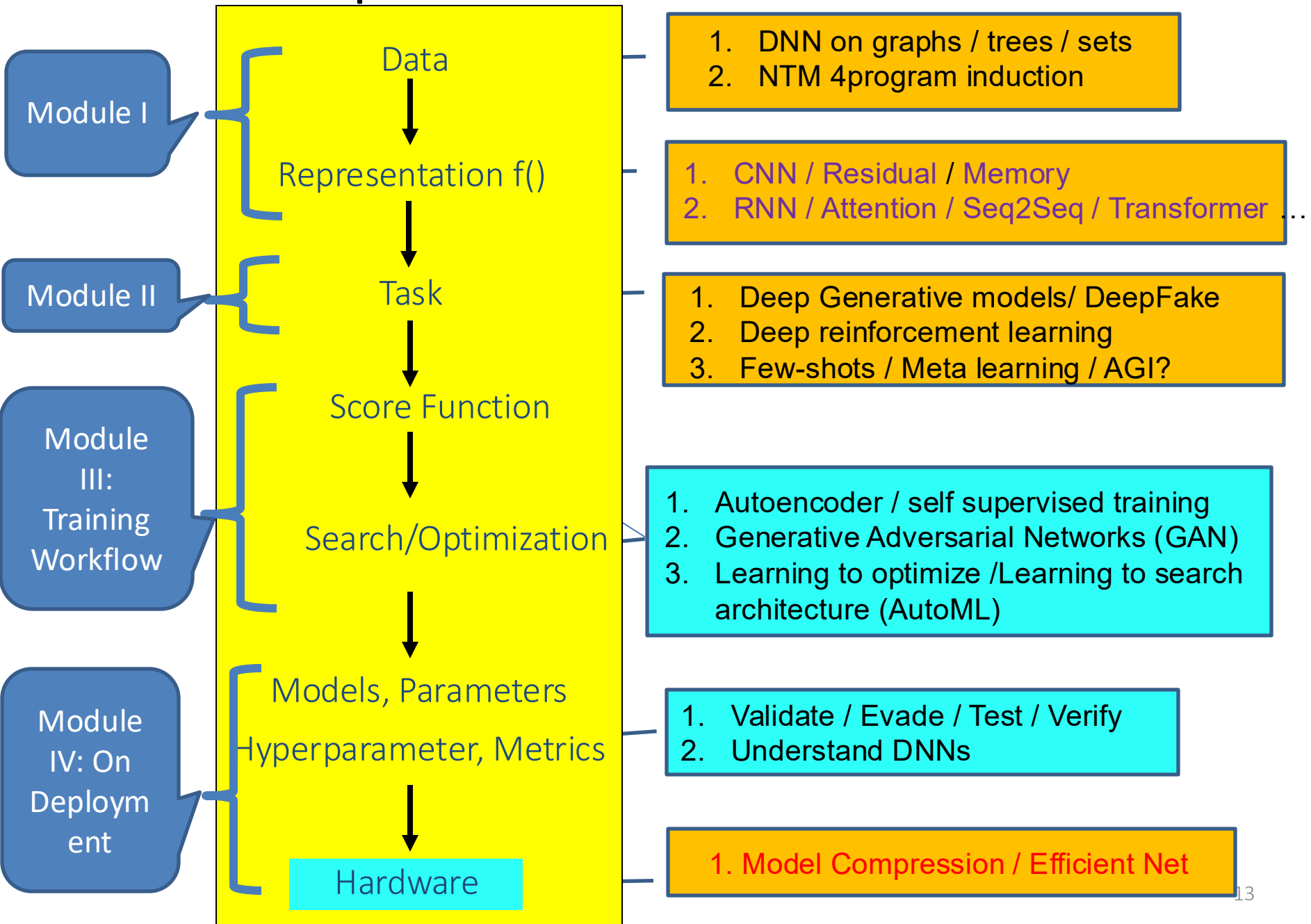
Learn to Interact with environment

Three major sections for classification

- We can divide the large variety of classification approaches into **roughly three major types**
 1. Discriminative classifier
 - directly estimate a decision rule/boundary
 - e.g., logistic regression, neural networks
 - e.g., support vector machine, decisionTrees
 2. Generative classifier :
 - build a generative statistical model
 - e.g., naïve bayes classifier, Bayesian networks
 3. Instance based classifiers:
 - Use observation directly (no models)
 - e.g. K nearest neighbors

Selected Deep Trends

<https://qdata.github.io/deep2Read/>



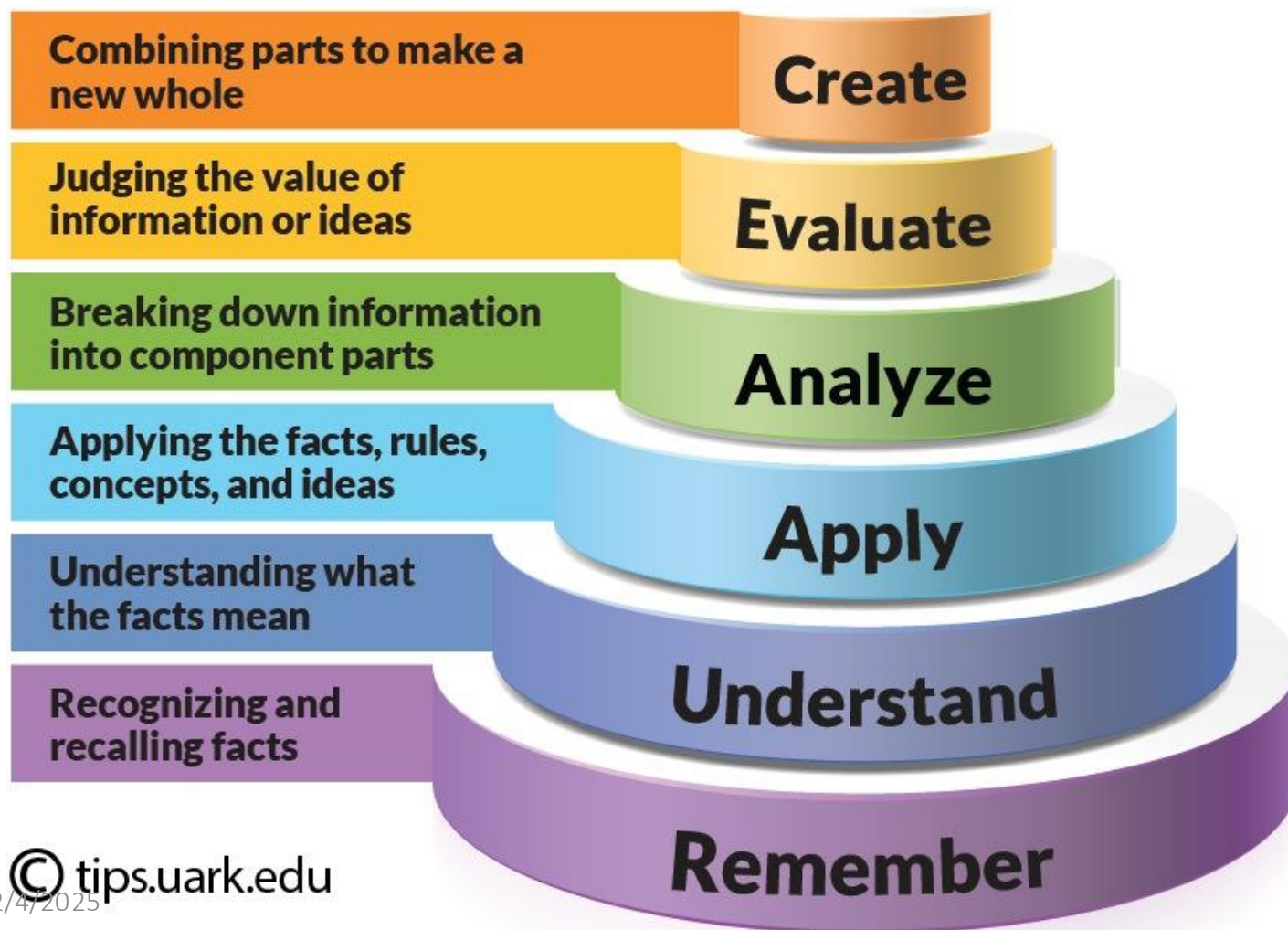
What we have covered (more)

- Learning theory / Model selection
 - K-folds cross validation / Model Selection
 - Expected prediction error
 - Bias and variance tradeoff (overfit / underfit)
 - Generative vs. Discriminative Classifiers
 - Remedy when Overfit / Underfit
 - Control / adjust model complexity, capacity
 - Control / adjust training size
 - Three plots:
 - Train / Vali Loss vs. Epochs
 - Train / Vali Loss vs. hyperparameter Values
 - Train / Vali Loss vs. Varying Size of Trainin

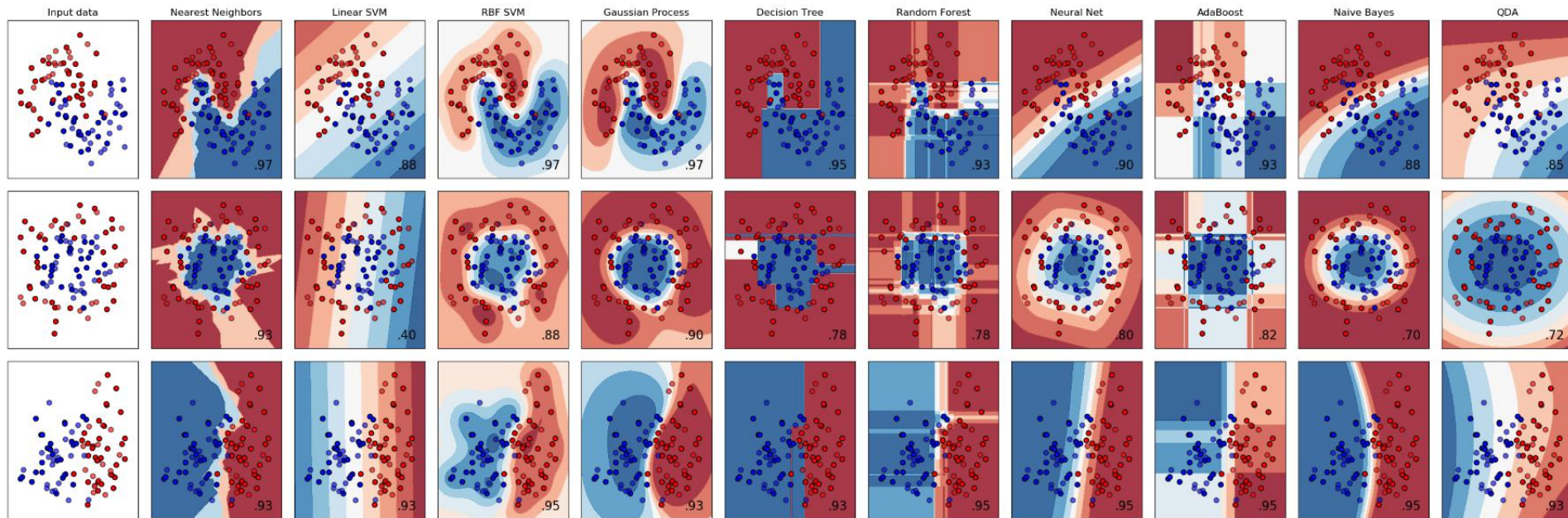
What we have covered for each component

Data	Tabular, 1-D sequential, 2-D Grid like Imaging, 3-D VR, Graph, Set
Task	Regression, classification, clustering, dimen-reduction
Representation	Linear func, nonlinear function (e.g. polynomial expansion), local linear, logistic function (e.g. $p(c x)$), tree, multi-layer, prob-density family (e.g. Bernoulli, multinomial, Gaussian, mixture of Gaussians), local func smoothness, kernel matrix, local smoothness, partition of feature space,
Score Function	MSE, Margin, log-likelihood, EPE (e.g. L2 loss for KNN, 0-1 loss for Bayes classifier), cross-entropy, cluster points distance to centers, variance, conditional log-likelihood, complete data-likelihood, regularized loss func (e.g. L1, L2) , goodness of inter-cluster similar
Search/ Optimization	Normal equation, gradient descent, stochastic GD, Newton, Linear programming, Quadratic programming (quadratic objective with linear constraints), greedy, EM, asyn-SGD, eigenDecomp, backprop
Models, Parameters	Linear weight vector, basis weight vector, local weight vector, dual weights, training samples, tree-dendrogram, multi-layer weights, principle components, member (soft/hard) assignment, cluster centroid, cluster covariance (shape), ...

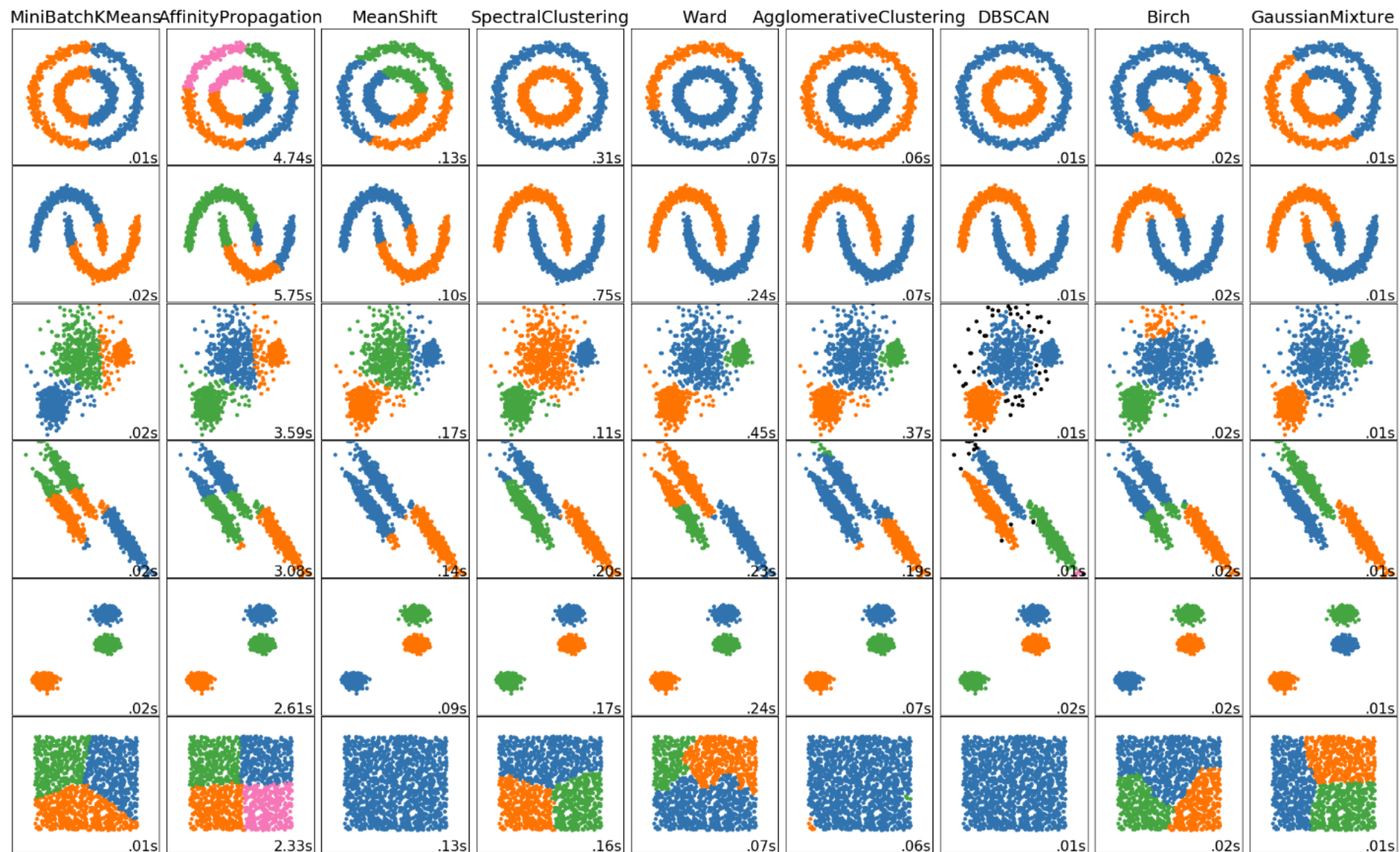
My Teaching Guide: Bloom's Taxonomy on Cognitive Learning



https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html



- ✓ different assumptions on data
- ✓ different scalability profiles at **training** time
- ✓ different latencies at prediction (**test**) time
- ✓ different model **sizes** (embedability in mobile devices)
- ✓ different level of model **interpretability / robustness**



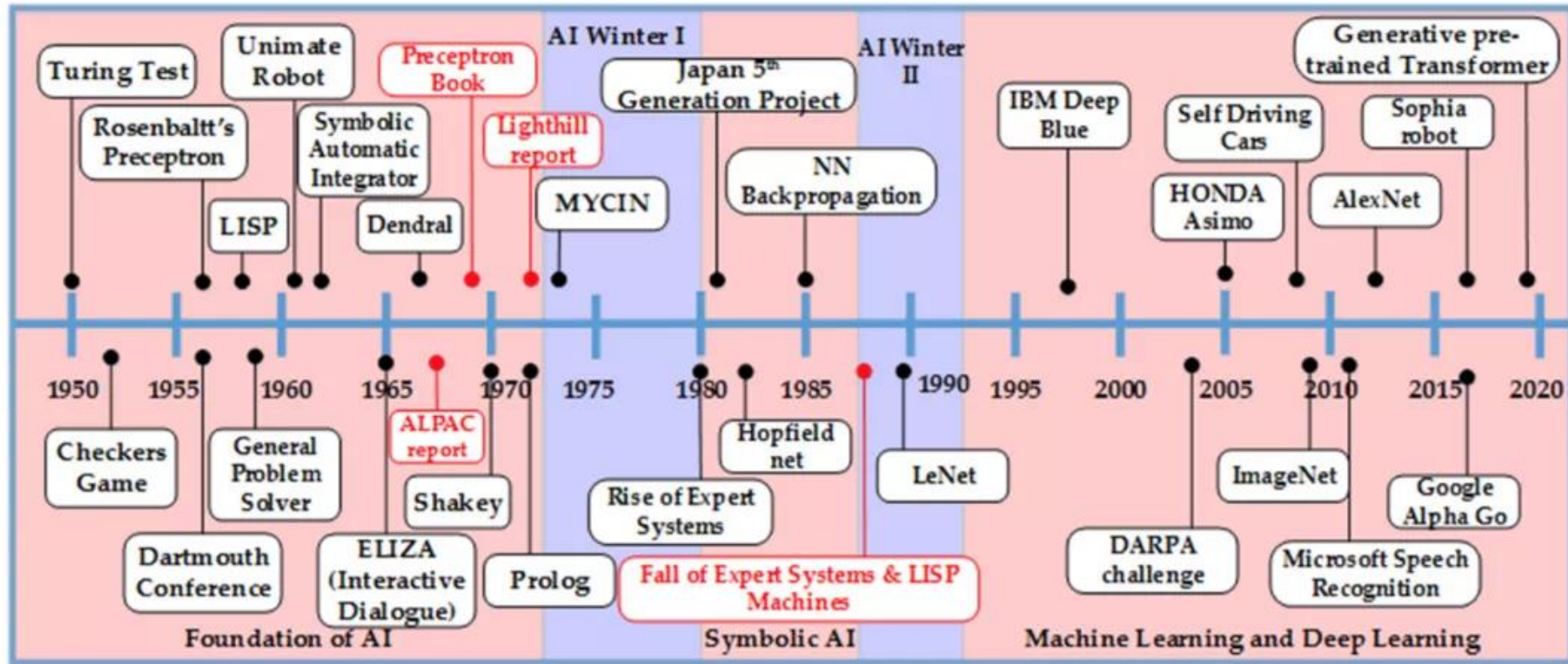
- ✓ different assumptions on data
- ✓ different scalability profiles
- ✓ different model **sizes** (embedability in mobile devices)

Final Review

- ❑ Review of covered so far



- ❑ Five Tribes of Machine Learning



Highly Recommend One Book: By Dr. Domingos: Master Algorithm

So How Do Computers Discover New Knowledge?

1. **Symbolists**--Fill in gaps in existing knowledge
2. **Connectionists**--Emulate the brain
3. **Evolutionists**--Simulate evolution
4. **Bayesians**--Systematically reduce uncertainty
5. **Analogizers**--Notice similarities between old and new

The Five Tribes of Machine Learning:

Tribe	Origins	Key Algorithm
Symbolists	Logic, philosophy	Inverse deduction
Connectionists	Neuroscience	Backpropagation
Evolutionists	Evolutionary biology	Genetic programming
Bayesians	Statistics	Probabilistic inference
Analogizers	Psychology	Kernel machines

Symbolists



Tom Mitchell



Steve Muggleton



Ross Quinlan

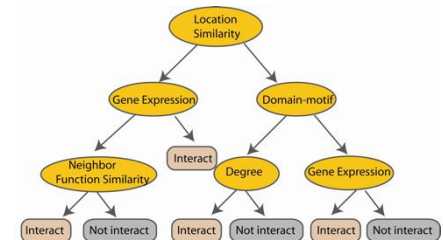
Tribe	Origins	Key Algorithm
Symbolists	Logic, philosophy	Inverse deduction

e.g., Decision Tree-building algorithms (1990s)

ID3: Iterative Dichotomiser 3. Developed in the 80s by Ross Quinlan.

C4.5: Successor of ID3, also developed by Quinlan ('93). Main improvements over ID3:

Adaboost: by Robert Schapire (1999)



Connectionists



Yann LeCun



Geoff Hinton

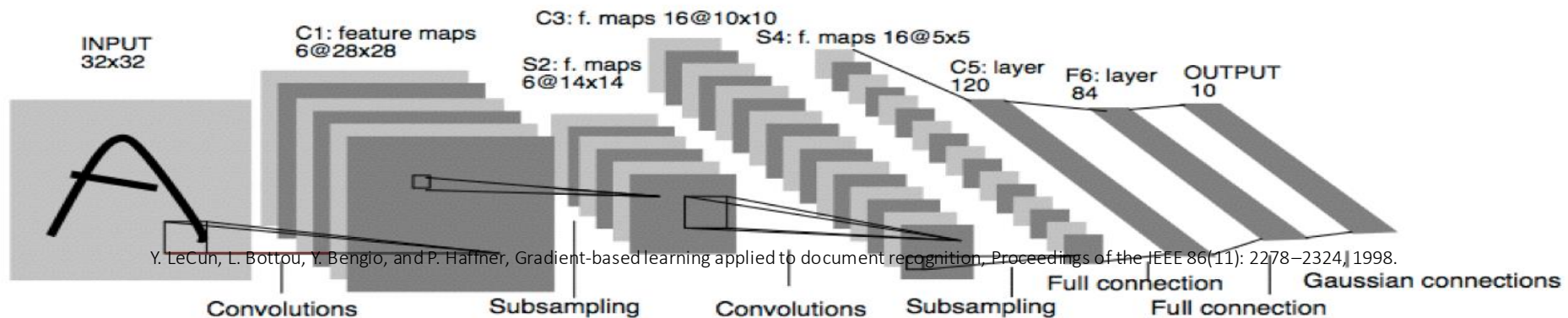


Yoshua Bengio

Tribe	Origins	Key Algorithm
Connectionists	Neuroscience	Backpropagation

Deep Learning (CNN) in the 90's

- Prof. Yann LeCun invented **Convolutional Neural Networks (CNN)** in 1998
- First NN successfully trained with many layers



Evolutionaries



John Koza



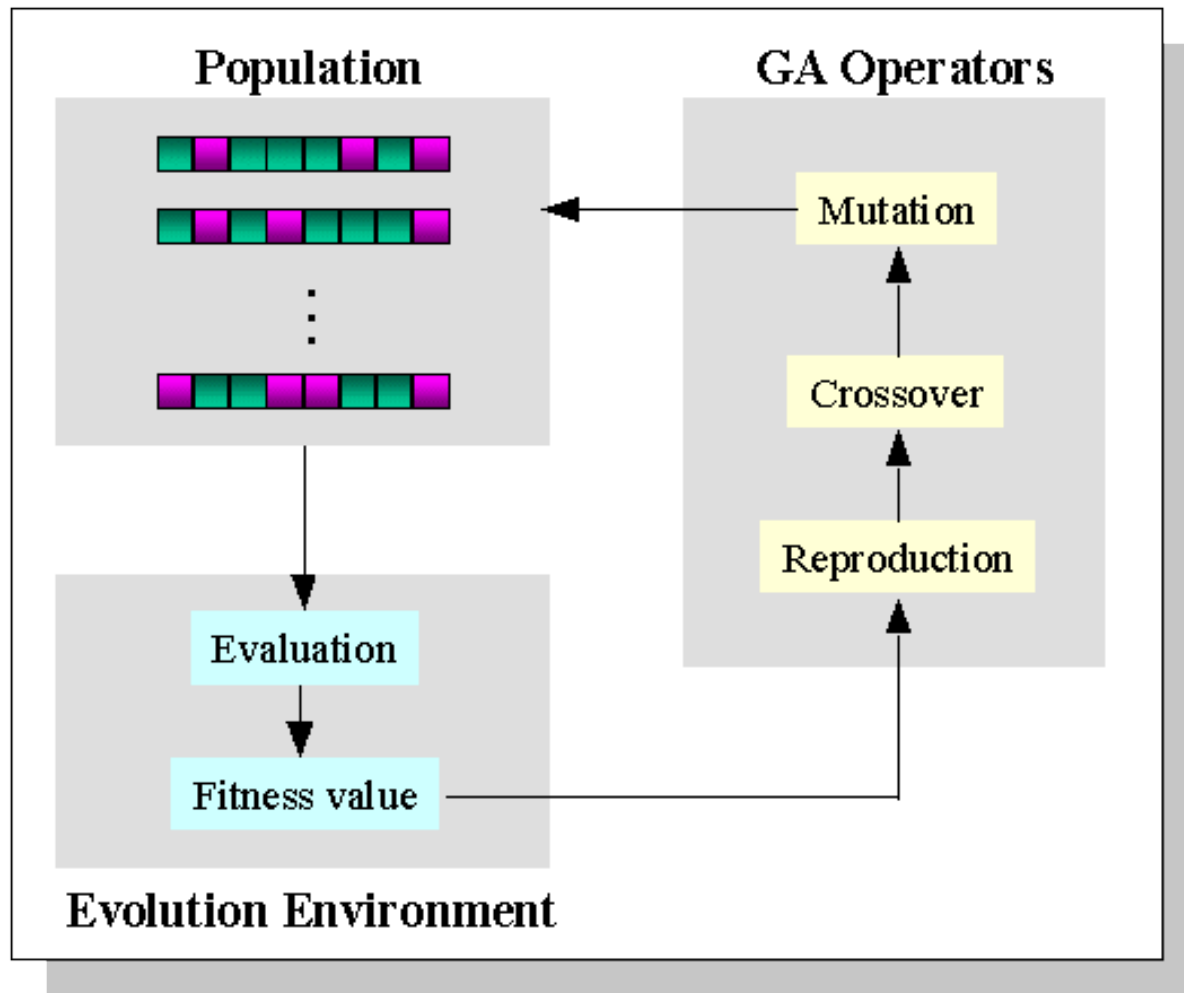
John Holland



Hod Lipson

Tribe	Origins	Key Algorithm
Evolutionists	Evolutionary biology	Genetic programming

Genetic Algorithms



Bayesians



David Heckerman



Judea Pearl

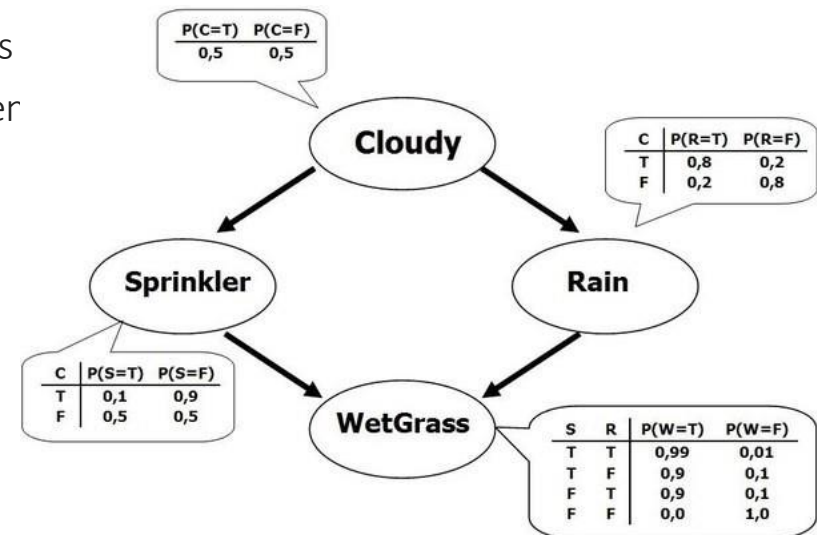


Michael Jordan

Tribe	Origins	Key Algorithm
Bayesians	Statistics	Probabilistic inference

Reasoning with uncertainty (Probabilistic Inference)

- “Bayesian network” was termed by [Judea Pearl](#) in 1985
- Bayes' conditioning is the basis for updating information in the graph
- The distinction between causal and evidential modes of reasoning
- In the late 1980s, established as a field of study.
 - Pearl's Probabilistic Reasoning in Intelligent Systems
 - [Neapolitan](#)'s Probabilistic Reasoning in Expert System



Likelihood

How probable is the evidence
given that our hypothesis is true?

$$P(H | e) = \frac{P(e | H) P(H)}{P(e)}$$

Prior

How probable was our hypothesis
before observing the evidence?

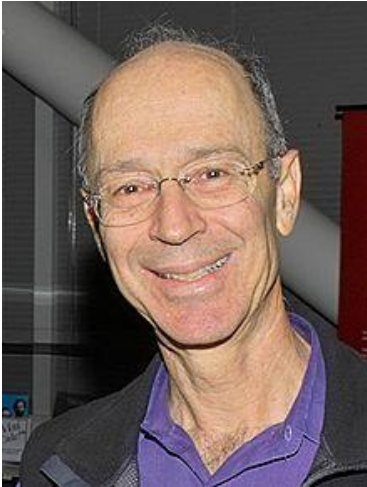
Posterior

How probable is our hypothesis
given the observed evidence?
(Not directly computable)

Marginal

How probable is the new evidence
under all possible hypotheses?
 $P(e) = \sum P(e | H_i) P(H_i)$

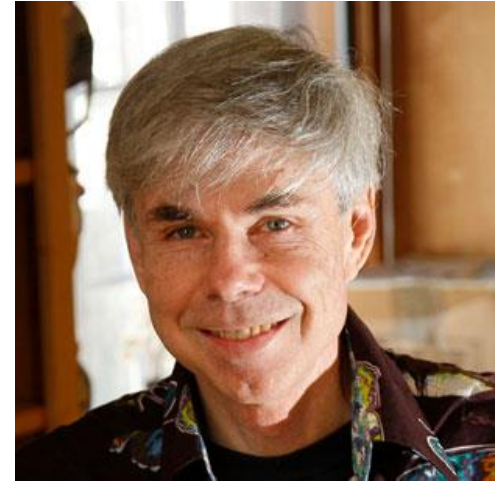
Analogizers



Peter Hart



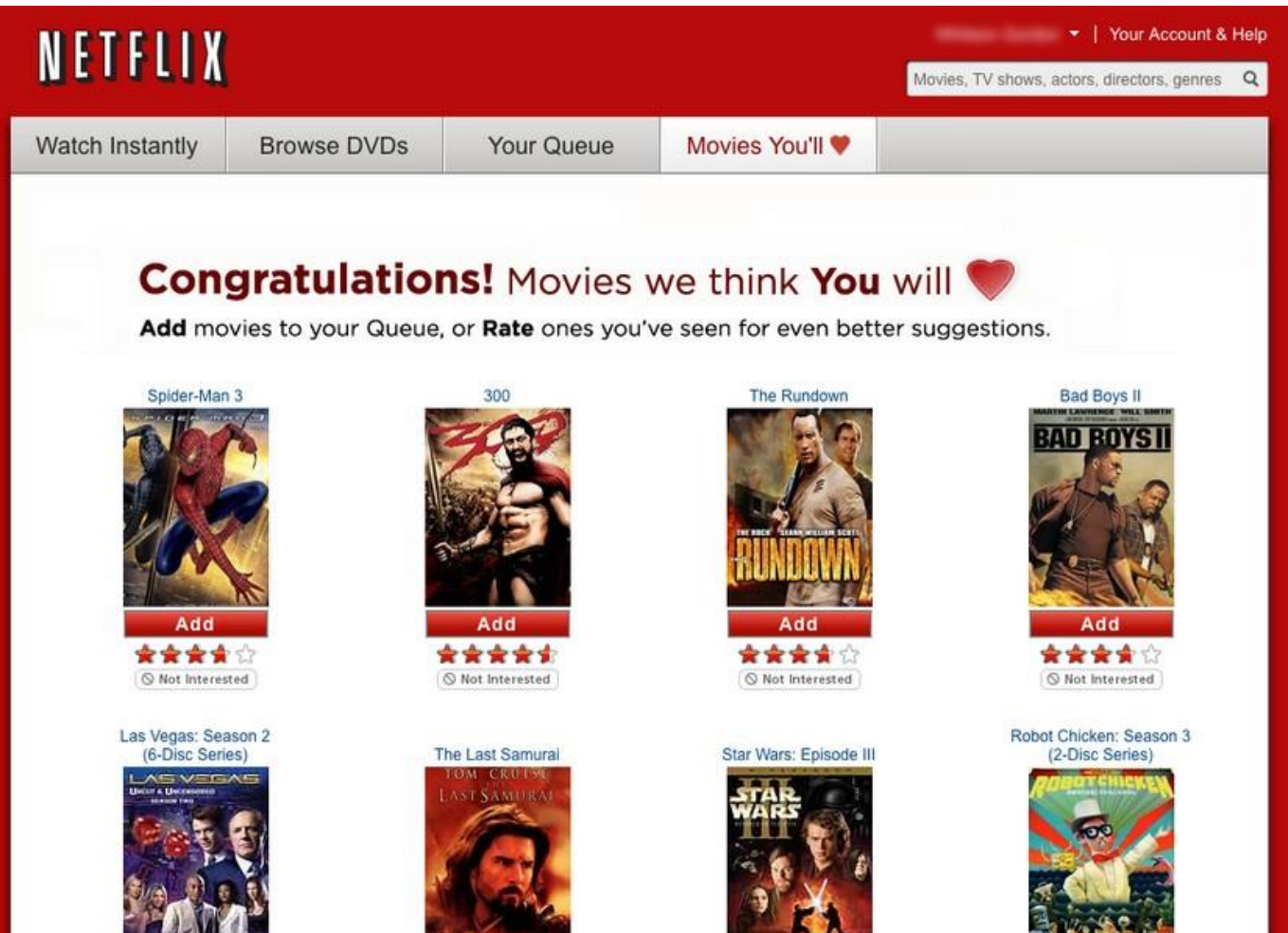
Vladimir Vapnik



Douglas Hofstadter

Tribe	Origins	Key Algorithm
Analogizers	Psychology	Kernel machines

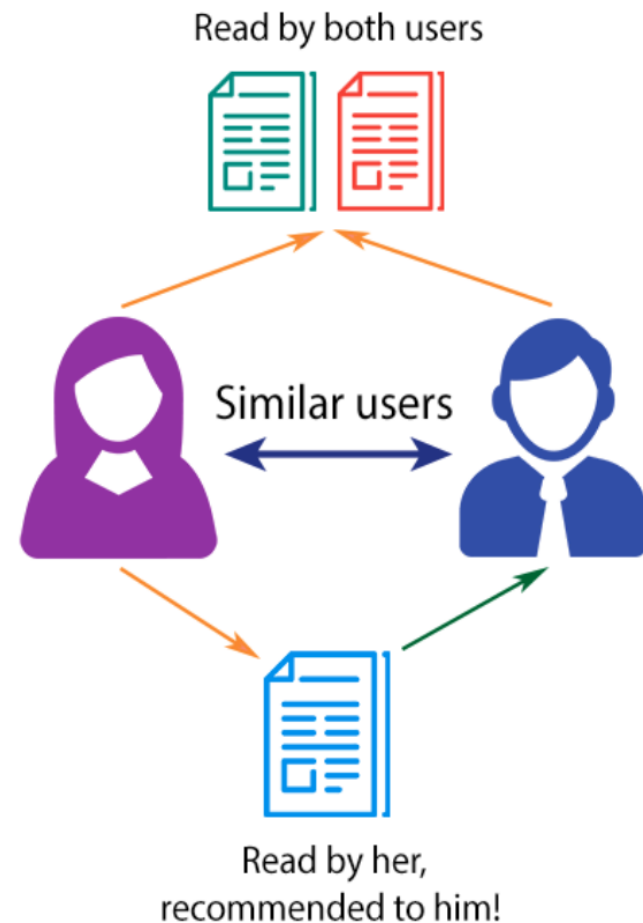
Recommender Systems



A little bit History

- **SVM** : first introduced in 1992, popular because of its success in handwritten digit recognition (1994); Regarded as an important example of “kernel methods”
- **Recommender Systems:**
 - E.g., Matrix Factorization

COLLABORATIVE FILTERING



The Big Picture

Tribe	Focus:	Origins	Solution	wrt our module in Nutshell
Symbolists	Knowledge composition	Logic, philosophy	Inverse deduction	Representations;
Connectionists	Credit assignment	Neuroscience	Backpropagation	Representations; Numerical Optimization
Evolutionaries	Search Structure discovery	Evolutionary biology	Genetic programming	Discrete Optimization;
Bayesians	Uncertainty	Statistics	Probabilistic inference	Likelihood type Score function;
Analogizers	Similarity	Psychology	Kernel machines	Representations; Reconstruction loss

Final Review

- ❑ Review of covered so far
- ❑ Five Tribes of Machine Learning

References

- ❑ Hastie, Trevor, et al. The elements of statistical learning. Vol. 2. No. 1. New York: Springer, 2009.
- ❑ Prof. Domingos' slides
- ❑ Prof. Andrew Ng's slides