Machine Learning Overview

Karan Nathwani

Outline

1. AI and the Machine Learning approach
2. ML problem types
    1. Based on data:
       Supervised, Unsupervised, Semi-supervised, Reinforcement
    2. Based on output:
       Regression, Classification
    3. Based on models:
       Generative, Discriminative
3. Disruption in Software Development
    – Software 1.0 and Software 2.0

History of AI

- Greek Mythology

Pygmalion of Cyprus

Sculpts marble Galatea that came to life (falls in love)
GBS' Pygmalion: Higgins teaches Eliza to speak Queen's English

Eliza is first AI program
(Weizenbaum, MIT)

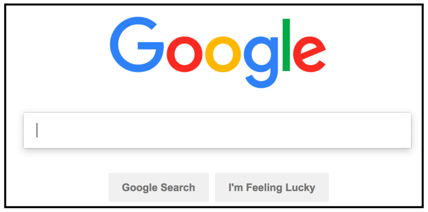
- Indian Mythology

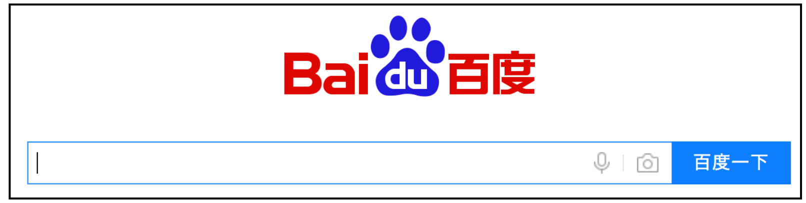
Ganesa (Buddhipriya)

Parvati forms sandalwood Ganesa, whose head is transplanted from an elephant

Today AI is ubiquitous

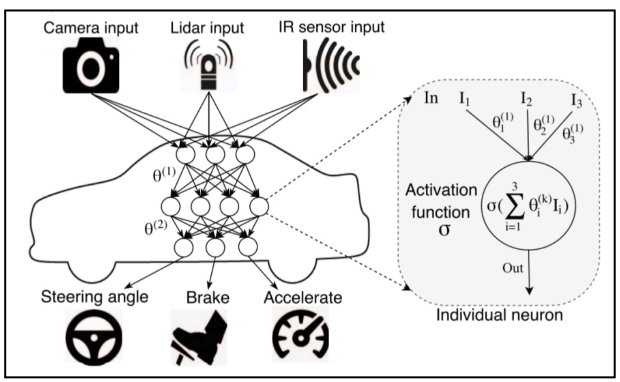
- Automate routine labor
  - Search



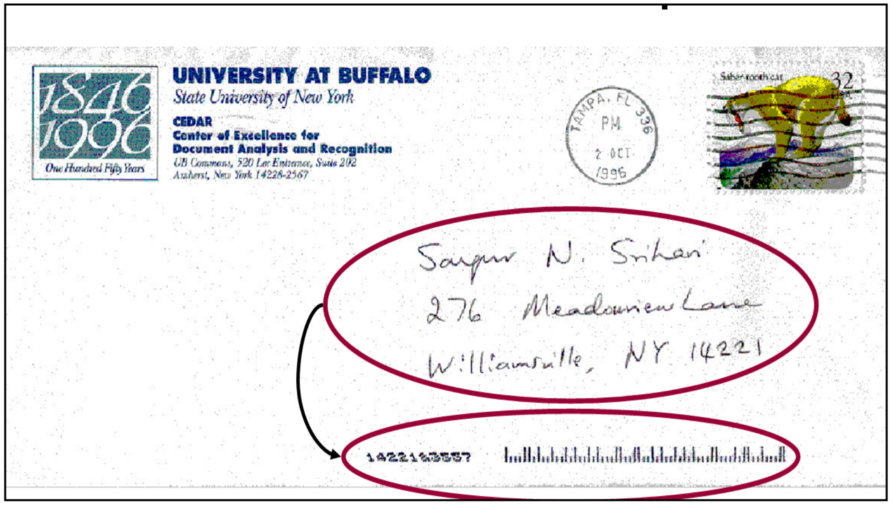


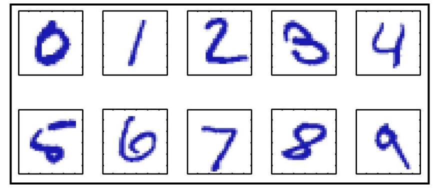
- Understand speech
  - SIRI, Alexa

- Autonomous Vehicles

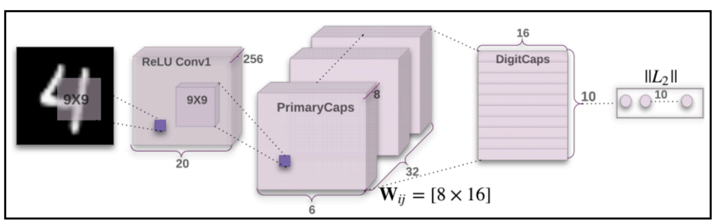


UB, AI and Fruit-fly





- Many handcrafted rules and exceptions
- Better learn from training set
- Handwriting rec cannot be done without ML!



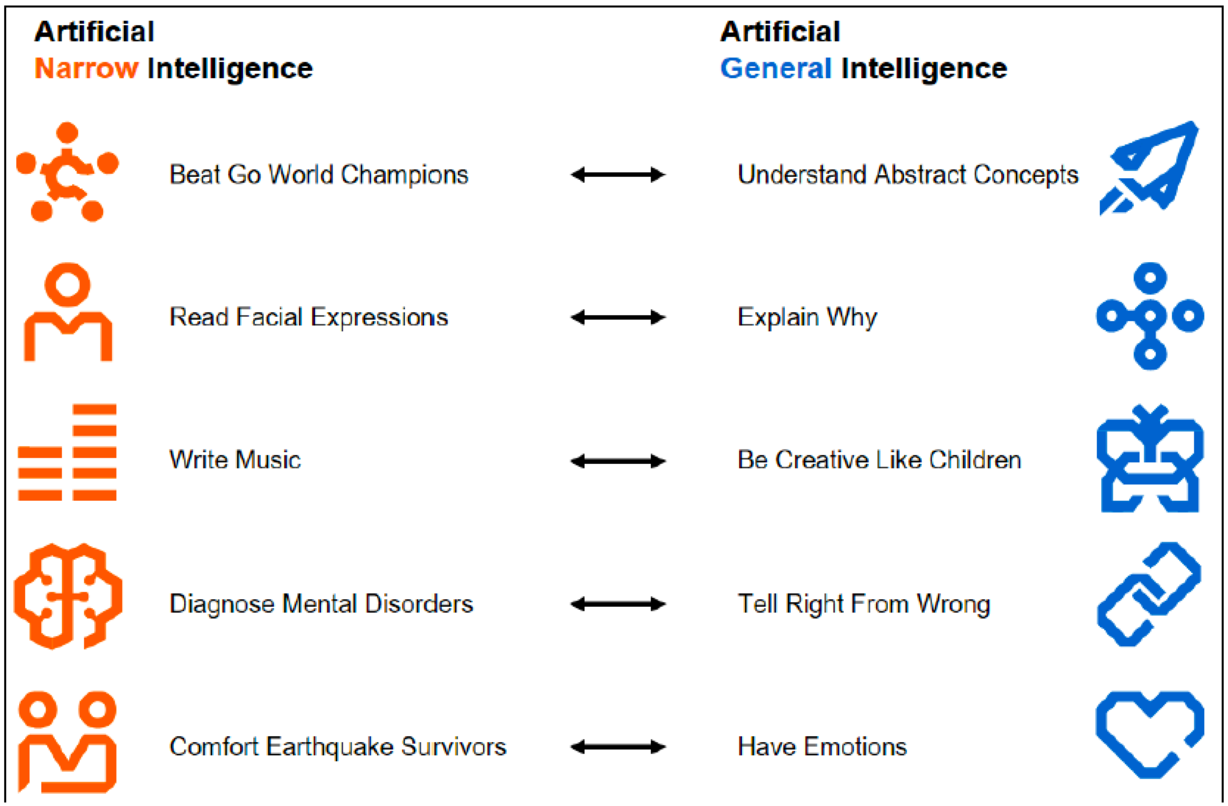
Handwriting is the fruit fly of AI

NYU, Toronto, Montreal, Baidu

AI Paradox

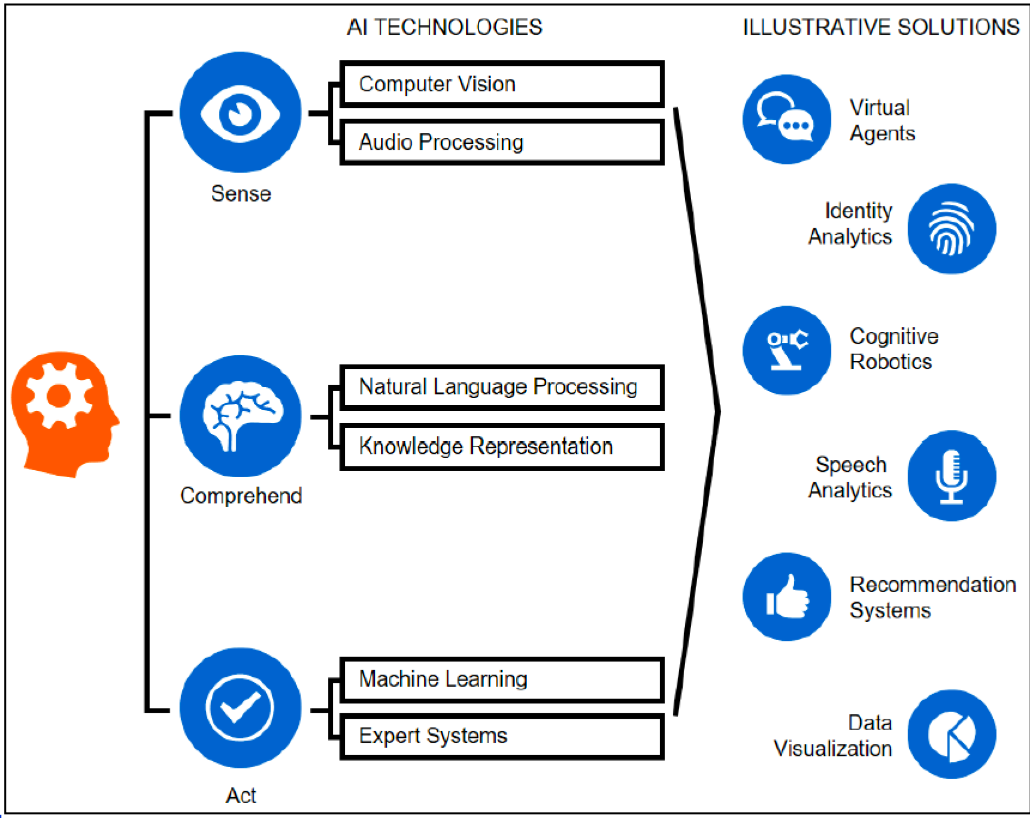
- Hard problems for people are easy for AI
- Easy problems are hard for AI
  - Narrow Intelligence General Intelligence

People easy tasks:



What tasks require intelligence?

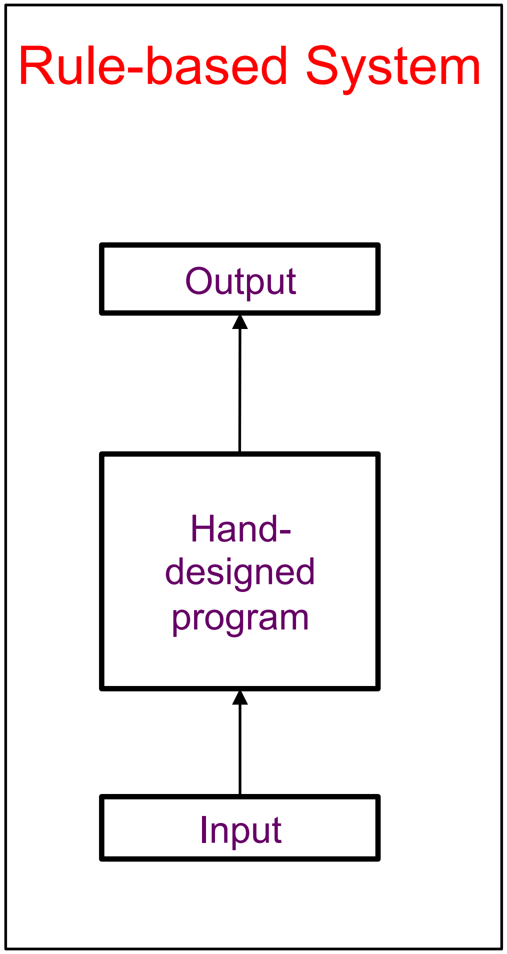
- Reasoning
  - Puzzles, Judgments
- Planning
  - Action sequences
- Learning
  - Improve with data
- Natural language
- Integrating skills
- Abilities to sense, act



Everyday life needs knowledge

- Knowledge is intuitive and subjective
  - Key challenge of AI is how to get this informal knowledge into a computer
- Knowledge-based Approach
  - Hard-code knowledge in a formal language
  - Computer can reason about statements in these languages using inference rules
  - Examples:
    - Expert systems for diagnosis (MYCIN, CADUCEUS)
    - Design (VAX)
-

Knowledge-Based AI



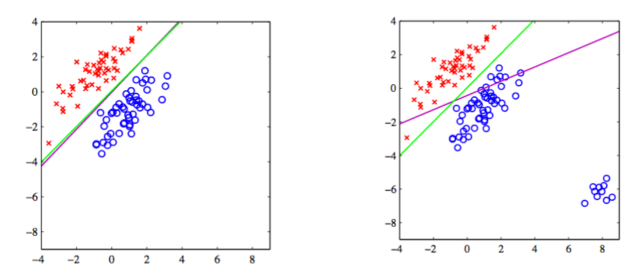
Disadvantage: Unwieldy process

- Time of human experts
- People struggle to formalize rules with enough complexity to describe the world

The Machine Learning approach

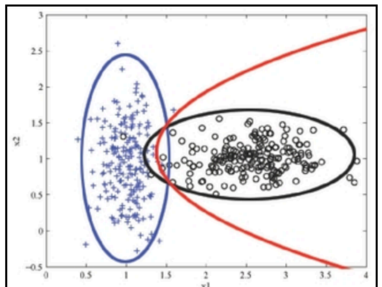
- Difficulties of hard-coded approach suggests:
  - Allow computers to learn from experience
- First determine what features to use
- Learn to map the features to outputs

Linear classifier



Magenta: Least Squares
Green: Logistic Regression (more robust)

Quadratic classifier



2D binary classification with Naive Bayes. A density contour is drawn for the Gaussian model of each class and the decision boundary is shown in red

The ML Approach

Data Collection
    Samples
Model Selection
    Probability distribution to model process
Parameter Estimation
    Values/distributions

Inference
    Find responses to queries

Generalization
(Training)

Decision
(Inference
OR
Testing)

Learning Problem Definition

- Improving some measure of performance P when executing some task T through some type of training experience E
- Example: Learning to detect credit card fraud

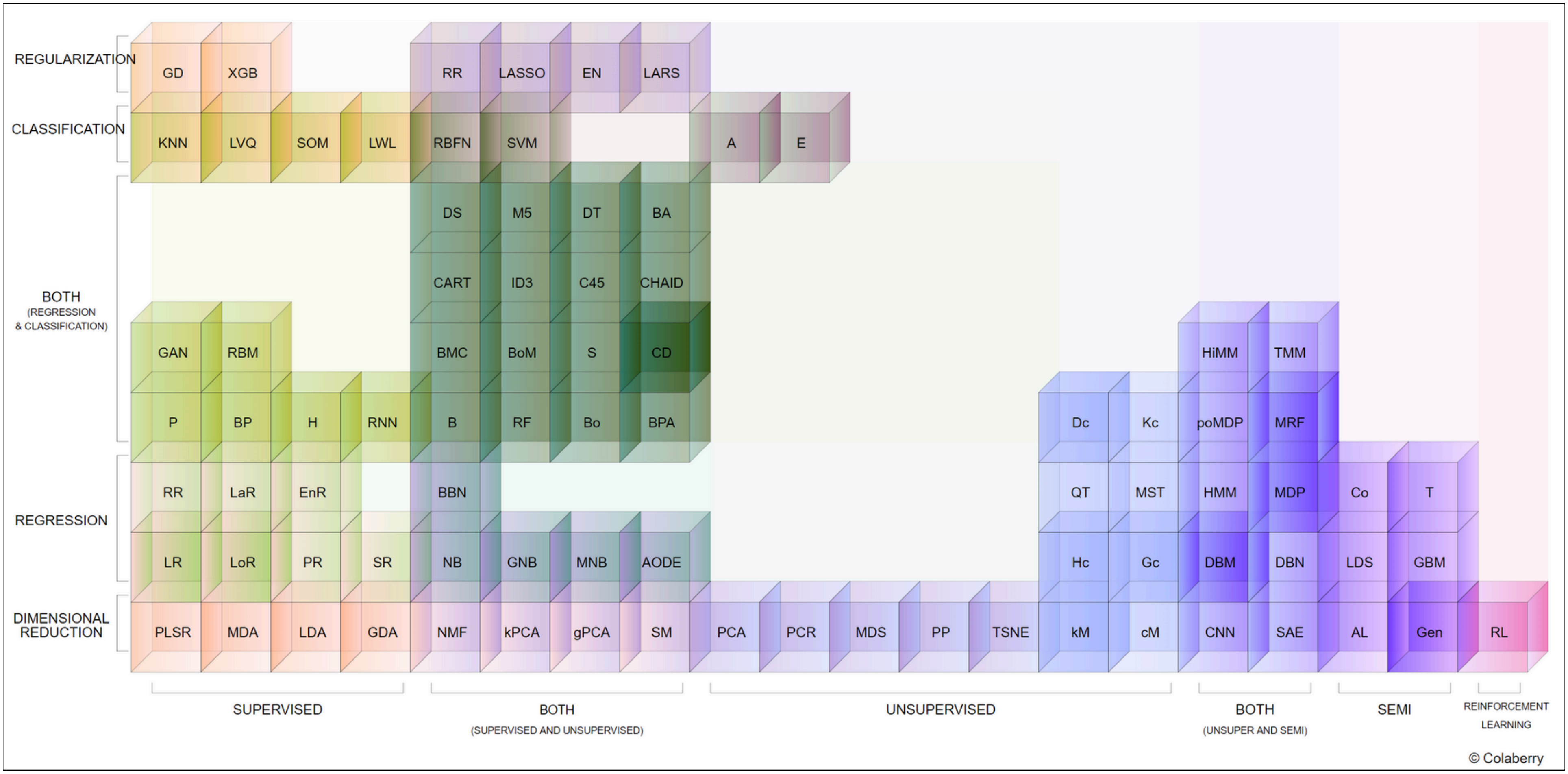
- **Task** T
  - Assign label of fraud or not fraud to credit card transaction
- **Performance measure** P
  - Accuracy of fraud classifier

  With higher penalty when fraud is labeled as not fraud
- **Training experience** E
  - Historical credit card transactions labeled as fraud or not

ML Problem Types

1. Based on Type of Data
   1. Supervised, Unsupervised, Semi-supervised
   2. Reinforcement Learning
2. Based on Type of Output
   – Regression, Classification
3. Based on Type of Model
   – Generative, Discriminative

Periodic Table of ML algorithms

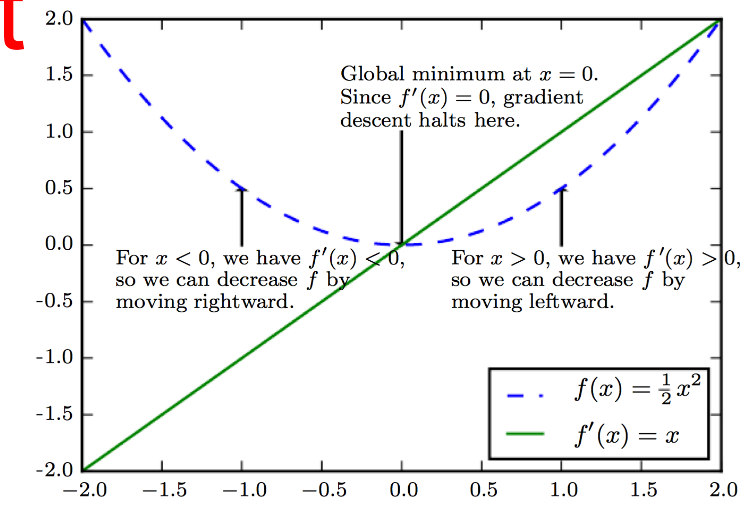


We will look at examples first proceeding along the horizontal axis and then along the vertical

Supervised Learning

- Most widely used methods of ML, e.g.,
  - Spam classification of email
  - Face recognizers over images
  - Medical diagnosis systems
- Inputs $\boldsymbol{x}$ are vectors or more complex objects
  - documents, DNA sequences or graphs
- Outputs are binary, multiclass(K),
  - Multi-label (more than one class), ranking,
  - Structured:
    - $\boldsymbol{y}$ is a graph satisfying constraints, e.g., POS tagging
  - Real-valued or mixture of discrete and real-valued

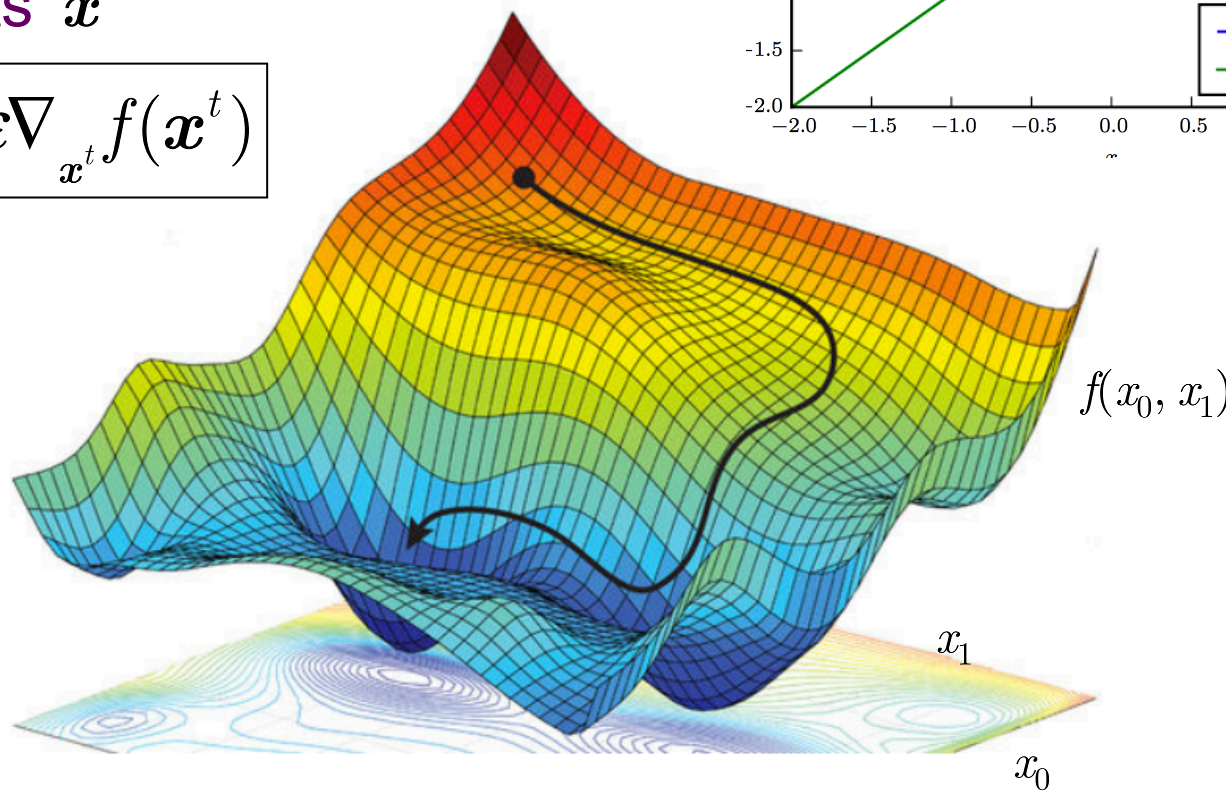
Supervised Learning by Gradient Descent



Classification Task:

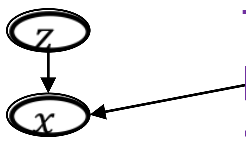
Loss function $f(\boldsymbol{x})$,
e.g., sum of squared errors,
given weights $\boldsymbol{x}$

$$\boxed{\boldsymbol{x}^{t+1} = \boldsymbol{x}^{t} - \varepsilon \nabla_{\boldsymbol{x}^t} f(\boldsymbol{x}^t)}$$

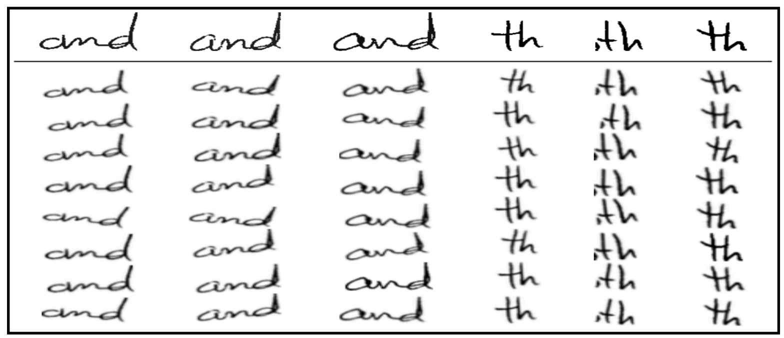


Ability to Generalize

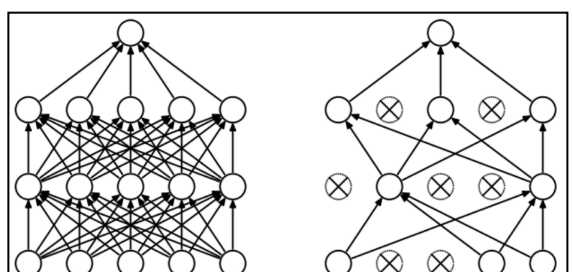
- ML algorithms need to perform well not just on training data but on new inputs as well
  1. Parameter Norm Penalties (L^2- and L^1- regularization)
  2. Data Set Augmentation



Translation
Rotation
Scaling



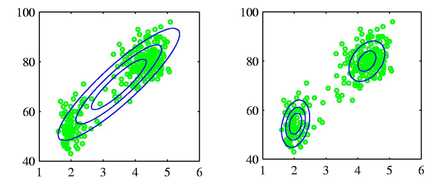
  3. Early Stopping
  4. Dropout



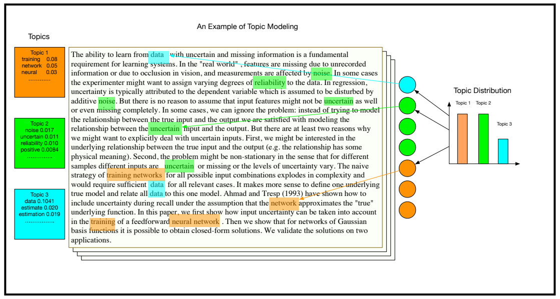
(a) Standard Neural Net (b) After applying dropout.

Unsupervised Learning

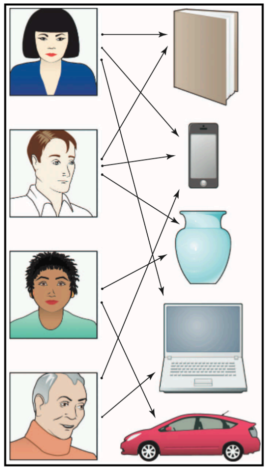
- Unlabeled data assuming underlying structure



1. Clustering to find partition of data
2. Identify a low-dimensional manifold
   - PCA, Autoencoder



3. Topic modeling
   - Topics are distributions over words
   - Document: a distribution across topics
     – Methods: SVD, Collaborative Filtering



4. Recommendation Systems

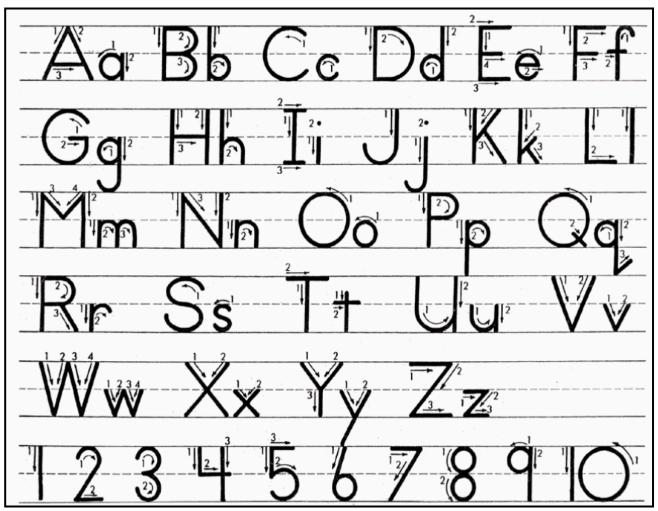
– Data links between users and items

– Suggest other items to user

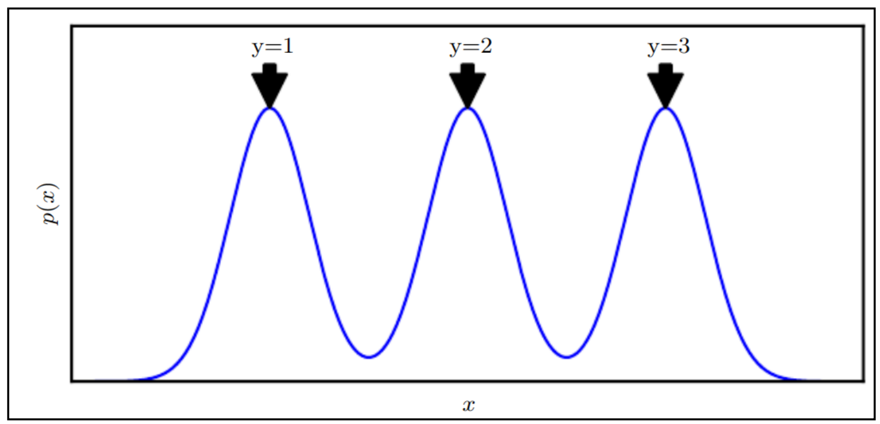
– Solution: SVD, Collaborative Filtering

How semi-supervised can succeed

- Ex: density over $\boldsymbol{x}$ is a mixture over three components, one per value of $\mathbf{y}=$ cap/small/digit
- If components well-separated:
  - modeling $p(\mathbf{x})$ reveals where each component is
    - A single labeled example per class enough to learn $p(\mathbf{y}|\mathbf{x})$



$\boldsymbol{x}$ = no. of black pixels



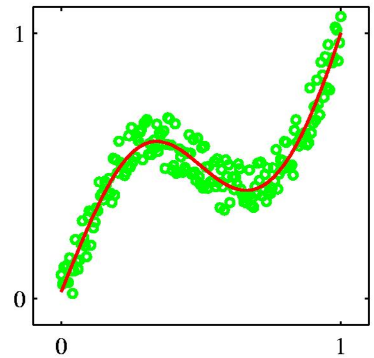
In this case $p(\mathbf{y}|\mathbf{x})$ is a univariate Gaussian for $\mathbf{y}$=1,2,3

Reinforcement Learning

- Training data inbetween supervised/unsupervised
  - Indication of whether action is correct or not
  - Reward signal may refer to entire input sequence
  - Dog is given a reward/punishment for an action
    - Policies: what actions to take in a particular situation
    - Utility estimation: how good is state (→used by policy)
- No supervised output but delayed reward
- Credit assignment
  - what was responsible for outcome
- Applications:
  - Game playing, Robot in a maze, Multiple agents, partial observability, …

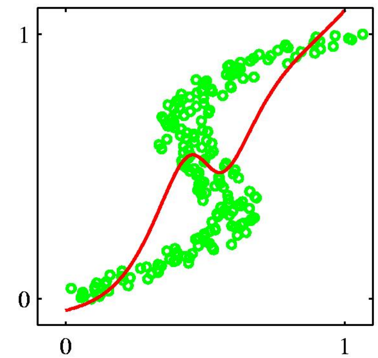
Regression

Problem data set



Red curve is result of fitting a two-layer neural network by minimizing squared error

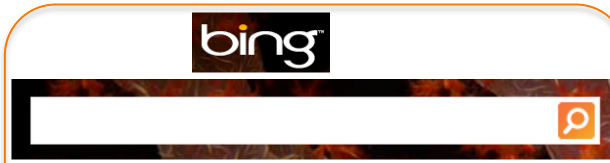
Corresponding inverse problem by reversing x and t



Very poor fit
to data:
GMMs used here

Regression: Learning to Rank

$(d > 200)$



In LETOR 4.0 dataset
46 query-document features
Maximum of 124 URLs/query

Input (x_i):
(d Features of Query-URL pair)

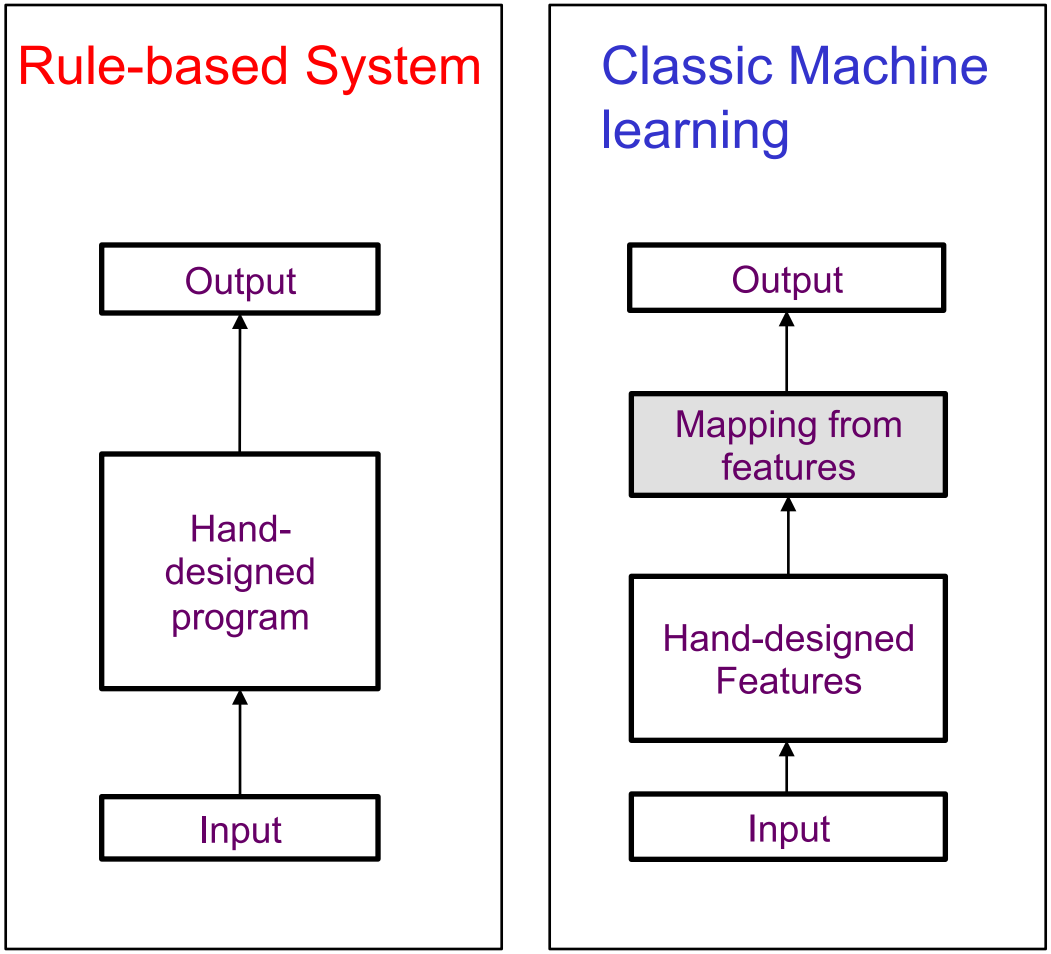
- Log frequency of query in anchor text
- Query word in color on page
- # of images on page
- # of (out) links on page
- PageRank of page
- URL length
- URL contains "~"
- Page length

Traditional IR uses TF/IDF

Output (y):
Relevance Value

Target Variable
- Point-wise (0,1,2,3)
- Regression returns continuous value
  - Allows fine-grained ranking of URLs

Two paradigms in AI

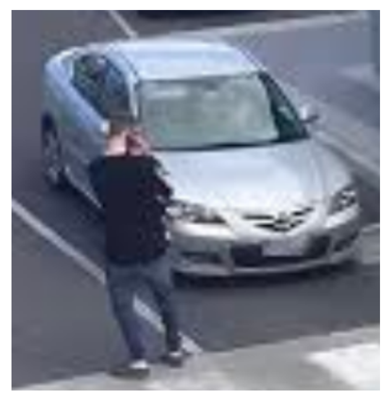


Shaded boxes indicate components that can learn from data

Designing right set of features

- Simple Machine Learning depends heavily on *representation* of given data
- For detecting a car in photographs
  - Tire shape difficult in terms of pixel values
    - Shadows, glare, occlusion



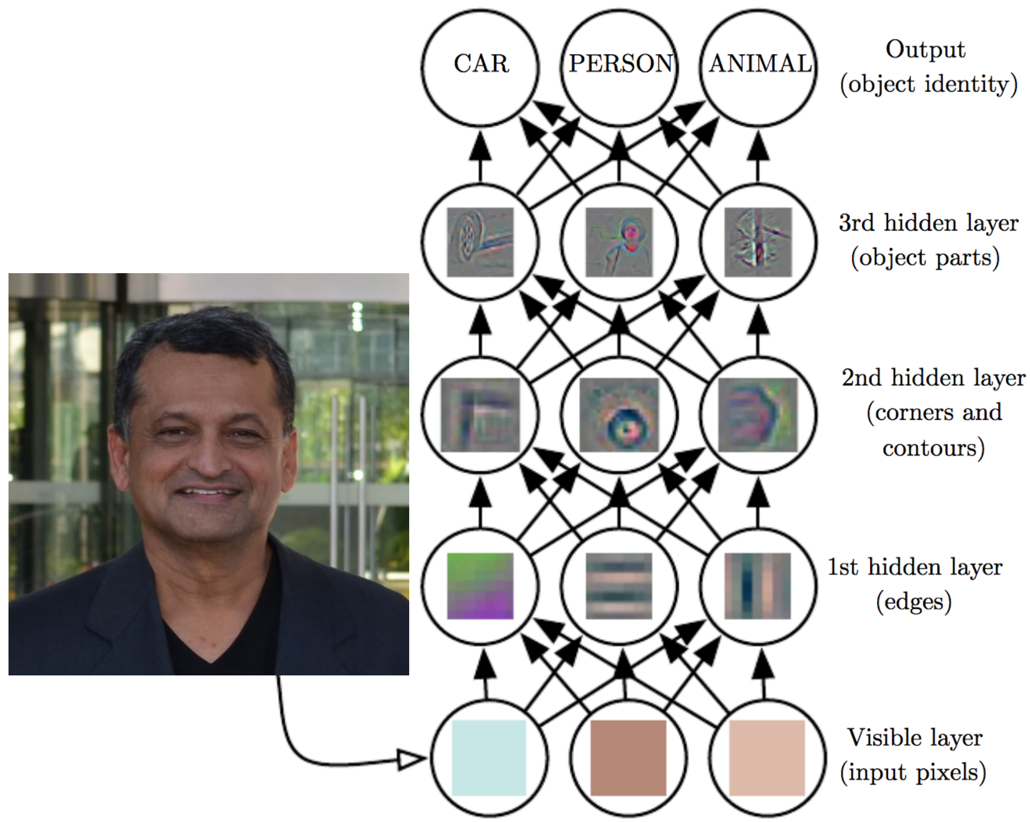


Representation Learning

- Solution: use ML to not only learn mapping from representation to output but representation itself
  - Better results than hand-coded representations
- Allows AI systems to rapidly adapt to new tasks
  - Designing features can take great human effort
    - Can take decades for a community of researchers
- Does not need programmer to have deep knowledge of the problem domain

Feature Learning for Classification

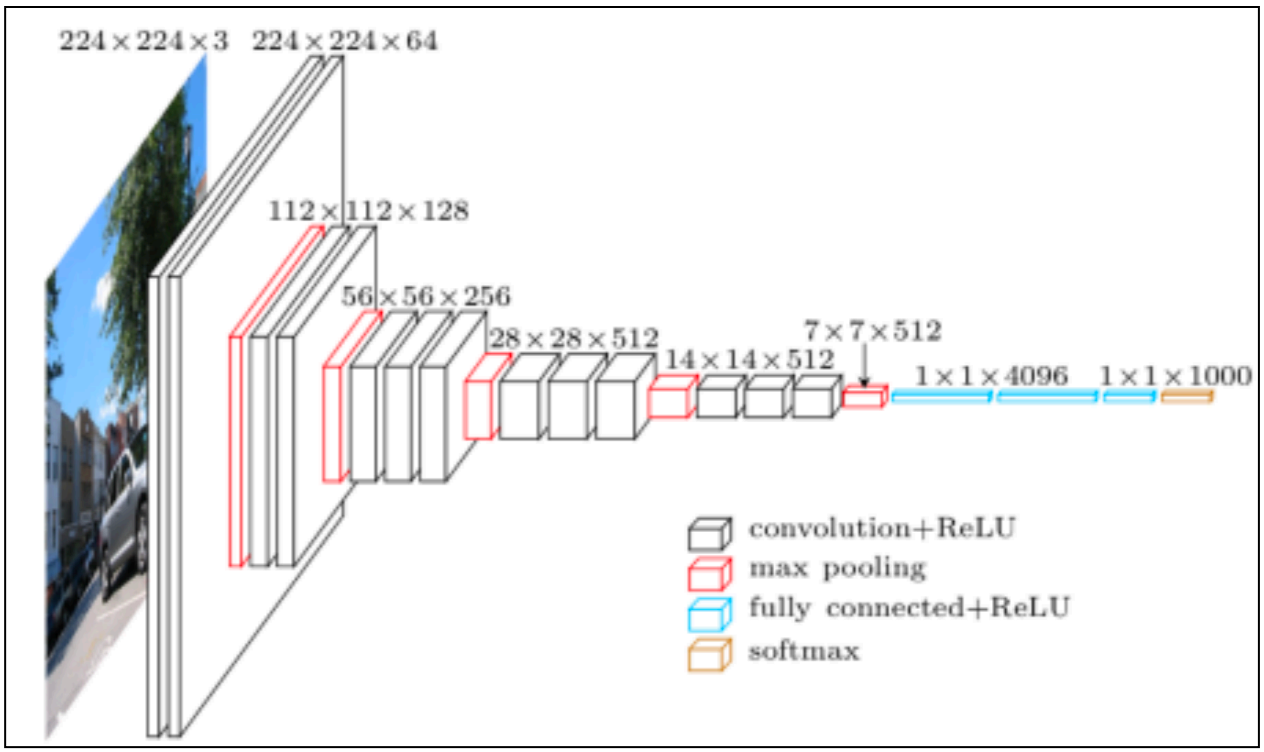
- Function to map pixels to object identity is complicated
- Series of hidden layers extract increasingly abstract features
- Final decision made by a simple classifier

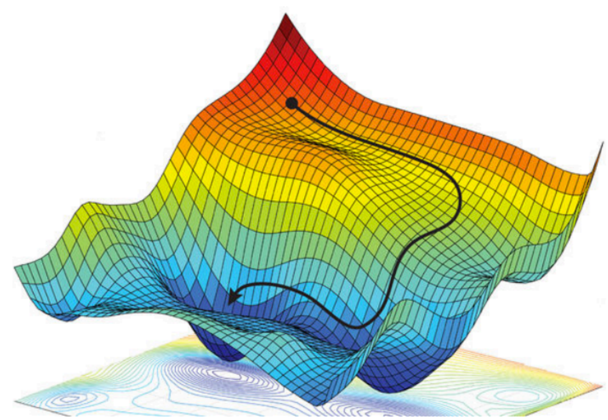


Deep Learning

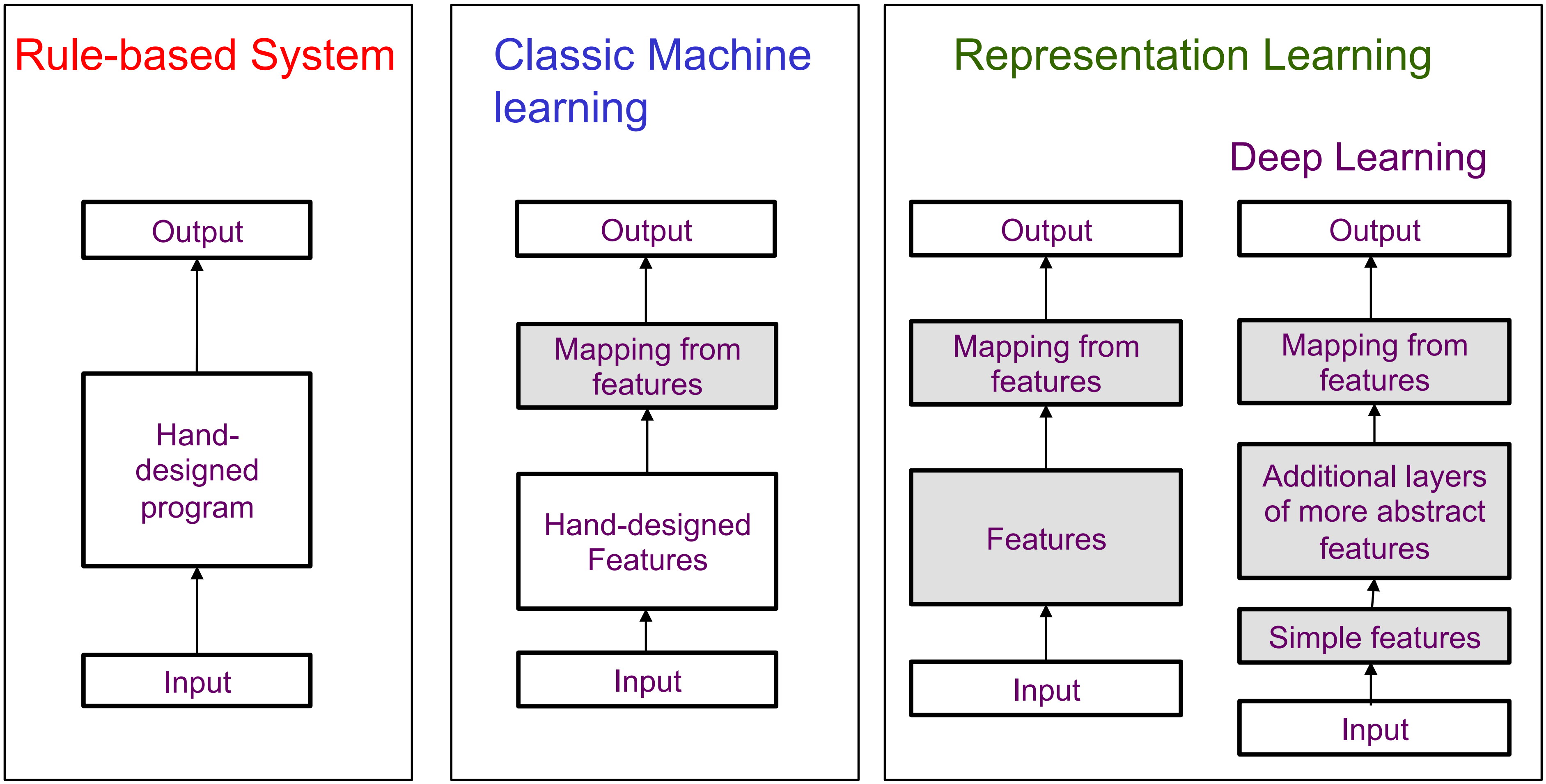
- Understand the world as hierarchy of concepts
  - How these concepts are built on top of each other is deep, with many layers
  - Weights learnt by gradient descent

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^{t} - \varepsilon \nabla_{\boldsymbol{x}^{t}} f(\boldsymbol{x}^{t})$$





Summary of AI Models



Shaded boxes indicate components that can learn from data

AI Paradigm Shift

→

- Physics paradigm shift
  - Newtonian Physics
    - Cannot explain black-body radiation
  - Quantum Mechanics
- AI paradigm shift
  - Knowledge-based systems
    - Cannot perform simple recognition tasks
  - Simple machine learning methods
    - Cannot perform complex recognition tasks
  - Deep Learning methods

ML Models rely on Probability Theory

- Sum Rule for Marginalization $$p(X = x_i) = \sum_{j=1}^{L} p(X = x_i, Y = y_j)$$
- Product Rule: for combining $$p(X,Y) = \frac{n_{ij}}{N} = p(Y \mid X)p(X)$$
- Bayes Rule $$p(Y \mid X) = \frac{p(X \mid Y)p(Y)}{p(X)}$$ where $$p(X) = \sum_{Y} p(X \mid Y)p(Y)$$

  Viewed as Posterior α likelihood x prior
- Fully Bayesian approach
  - Conjugate distributions
  - Feasible with increased computational power
  - Intractable posterior handled using either
    - Variational Bayes or
    - Stochastic sampling
    - e.g., Markov Chain Monte Carlo, Gibbs

Generative/Discriminative Models

- Generative
  - Naïve Bayes
  - Mixtures of multinomials
  - Mixtures of Gaussians
  - Hidden Markov Models (HMM)
  - Bayesian networks
  - Markov random fields

- Discriminative
  - Logistic regression
  - SVMs
  - Traditional neural networks
  - Nearest neighbor
  - Conditional Random Fields (CRF)

Summary

- Machine Learning as an AI approach
  - Overcomes limitations of knowledge-based systems
- Types of AI tasks
  - Data: Supervised, Unsupervised, Semi-supervised, Reinforcement
  - Output: Classification, Regression, Ranking
  - Model: Generative, Discriminative
- Disruption of computer science
  - Principled approach to develop IT systems
  - Drivers are mobile systems (big data), personalization