CS 6820 – Machine Learning

Lecture 1

Instructor: Eric S. Gayles, PhD.

Jan 3, 2018

Today's Goal

1. Introduction
2. Goals for the class
3. Sampling of ideas covered in the class
4. Specific topics discussed today
   - Introductory concepts
   - What is Machine Learning
   - Today's key technical enablers
   - Target Applications

Course Description

Advanced topics in Artificial Intelligence, including induction, decision trees, ensemble learning; current-best-hypothesis search, knowledge representation, explanation-based learning, relevance information, inductive logic programming; Bayesian networks, instance-based learning; neural networks and genetic algorithms; reinforcement learning, and adaptive dynamic programming. Prerequisites: CS 4810 or CS 6810 .

What Will We Cover?

- Where ML fits into the broader context of AI
- Technical enablers for ML
- Statistical modeling and optimization
- Supervised and Unsupervised Learning
- Classification – definition and approaches
- Effective tools for ML
- Solving real problems with “hands on” models

Topics Will Include

- Supervised VS Unsupervised learning
- Regression VS Classification
- Comparing ML solutions to other optimization and classification approaches
- Regression techniques
- Binary VS Multivariate Classification
- Clustering
- Support Vector Machines (SVM)
- Neural Networks
- Deep Learning

Course Description

Lecture 1	Introduction, What is Machine Learning, Technical Enablers, Target Applications
Lecture 2	Linear Regression, Linear Classifiers, Logistic Regression, Multiclass Logistic Regression
Lecture 3	Decision Trees, Induction, Knowledge Representation, Machine Learning Tools
Lecture 4	Gradient Descent, Bias, Loss Functions, Perceptrons, Inductive Logic Programming
Lecture 5	KNN Classification, Multi-class Classification
Lecture 6	Bayesian Learning and Inference, Probabilistic Classification
	Mid-term
Lecture 7	Genetic Algorithms, Reinforcement Learning, and Adaptive Dynamic Programming
Lecture 8	Neural networks, Back propagation, Deep Learning
Lecture 9	Kernel Methods, SVMs, Overfitting, Underfitting
Lecture 10	Unsupervised Learning, Clustering, PCA
Lecture 11	Markov decision processes, Gaussian Processes
	Final

What is Machine Learning ?

"**Machine learning** is a field of computer science that gives computers the ability to learn without being explicitly programmed." - Arthur Samuel (1959)

What is Machine Learning ?

"A computer program is said to learn from experience (E) with some class of tasks (T) and a performance measure (P) if its performance at tasks in T as measured by P improves with E" - Tom Mitchell (1998)

Information about the Instructor

- Instructor:
  - Eric S. Gayles, PhD.
- Education:
  - Ph.D. Computer Science and Engineering - Pennsylvania State University
- Industrial Experience:
  - Google, Intel, Teradata, Institute for Defense Analyses, Startups & VC
- Email: eric.gayles@csueastbay.edu
- Office: TBD
  - Office Hours: After class on Mondays, 8:00-9:00pm

Source Material

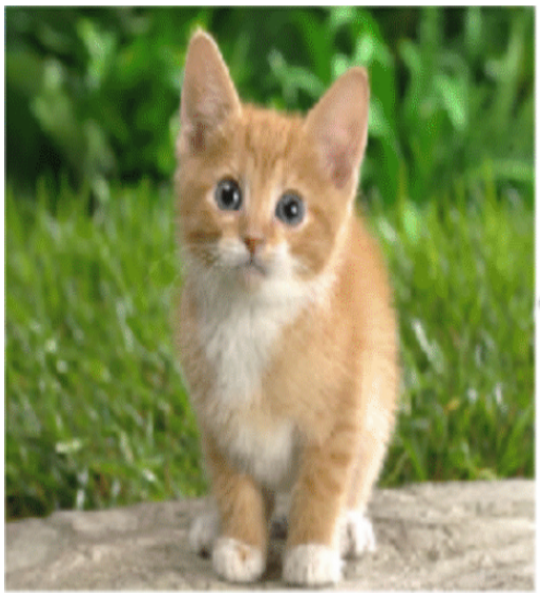
- Required:
  - E. Alpaydin - Introduction to Machine Learning, Third Edition
  - H. Daumé - A Course in Machine Learning (online)

- Good optional references:
  - C. Bishop, Pattern Recognition and Machine Learning
  - K. Murphy, Machine Learning: a Probabilistic Perspective
  - P. Klein - Coding the Matrix: Linear Algebra through Applications to Computer Science

Logistics

- Grading
  - 20% - Mid Term
  - 30% - Final
  - 5% - Project 1
  - 10% - Project 2
  - 10% - Project 3
  - 25% - Project 4 - Group

What am I ?



Source

Real World Example

- The 2010 Deepwater Horizon oil spill became the largest marine oil spill in human history
- An estimated four million barrels per day flowed freely into the gulf waters
- Straining the marine ecosystem and threatening the shoreline from Texas to Florida
- A sequence of failures involving multiple companies and work teams
- Repeated failures to follow safety procedures

- **Term Duration Question: Could Machine Learning have prevented this disaster?**

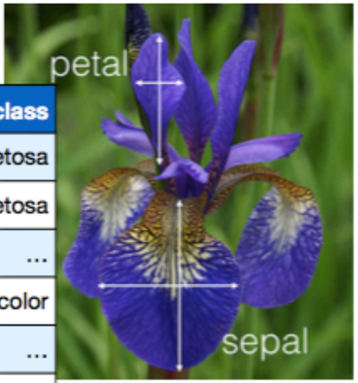
Some Intro Terminology

- Features - Distinct traits that can be used to describe each item within a set of samples in a quantitative manner.
- Feature vector - An n-dimensional vector of numerical quantized values that represent an object (sample).
- Feature extraction - Preparation of a feature vector that transforms the data from a high-dimensional space to a space of fewer dimensions.
- Training/Evolution set - Set of data input into our model to discover potentially predictive relationships.
- Training - Process of evaluation and optimizing the weights of a network.

Some Intro Terminology

IRIS

https://archive.ics.uci.edu/ml/datasets/Iris



Instances (samples, observations)

	sepal_length	sepal_width	petal_length	petal_width	class
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
...	...	...	...	...	...
50	6.4	3.2	4.5	1.5	veriscolor
...	...	...	...	...	...
150	5.9	3.0	5.1	1.8	virginica

Features (attributes, dimensions)

Classes (targets)

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1D} \\ x_{21} & x_{22} & \cdots & x_{2D} \\ x_{31} & x_{32} & \cdots & x_{3D} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{ND} \end{bmatrix}$$

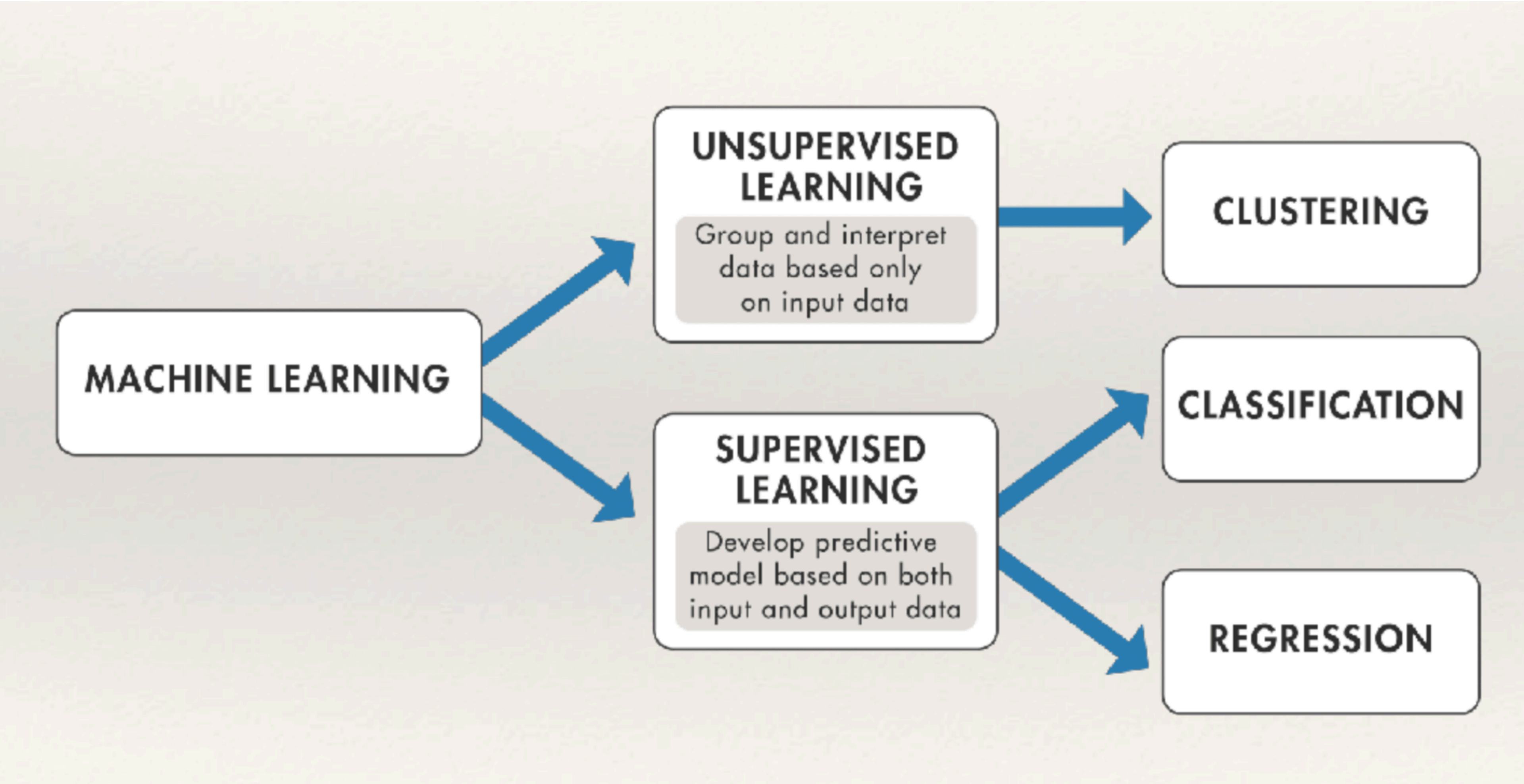
$$\mathbf{y} = [y_1, y_2, y_3, \cdots y_N]$$

*Raschka

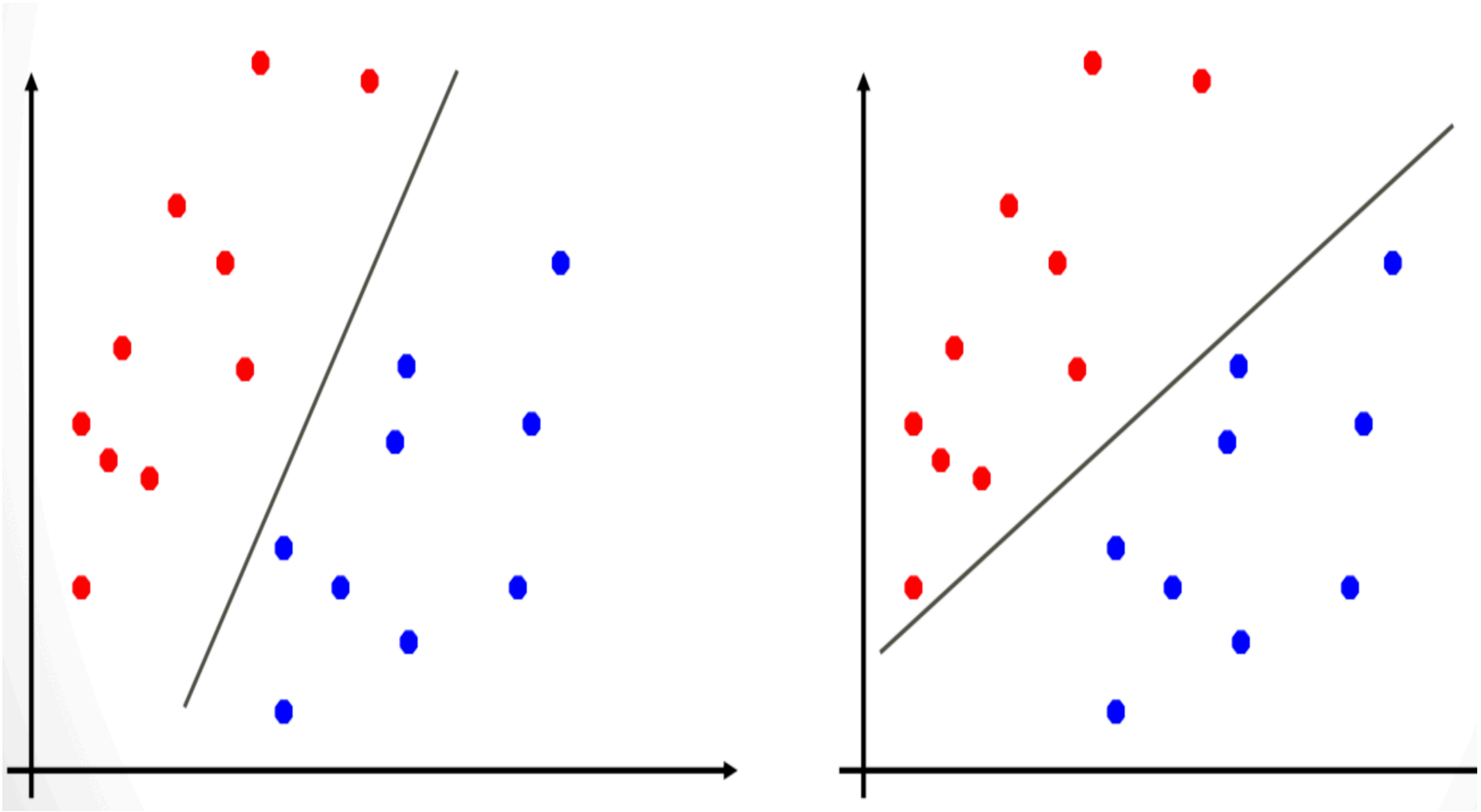
Some Intro Terminology

- Classification – prediction of a sample's class based on a priori observations (training data)
- Clustering - the assignment of a set of observations (samples) into subsets such that observations in the same cluster have meaningfully similar features that are statistically distinguishably from members of other subsets.
- Regression analysis – the statistical process to estimate relationships among variables.
- Regression - predicting the output value using training data.

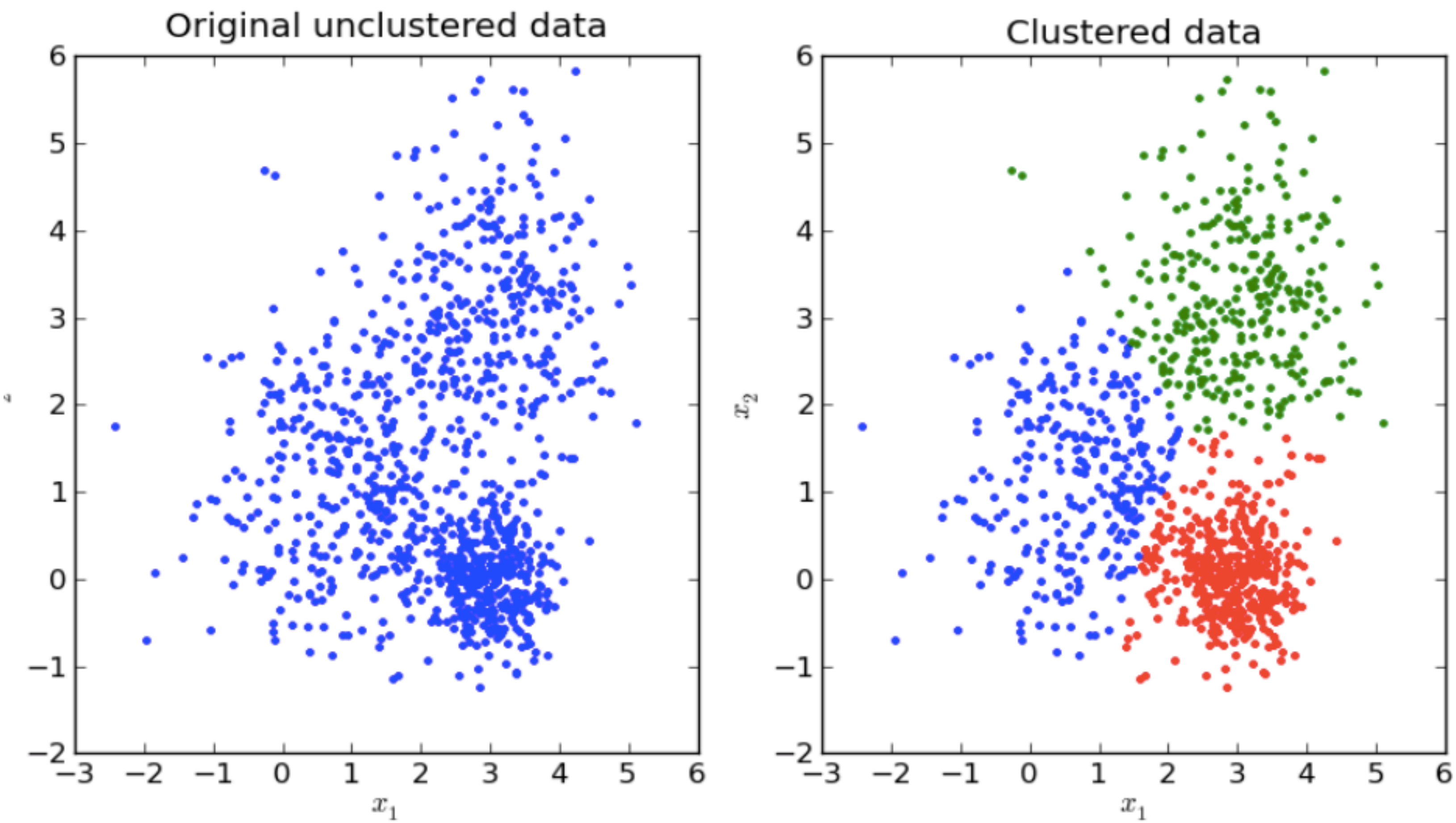
Some Intro Terminology



Linear Classifiers

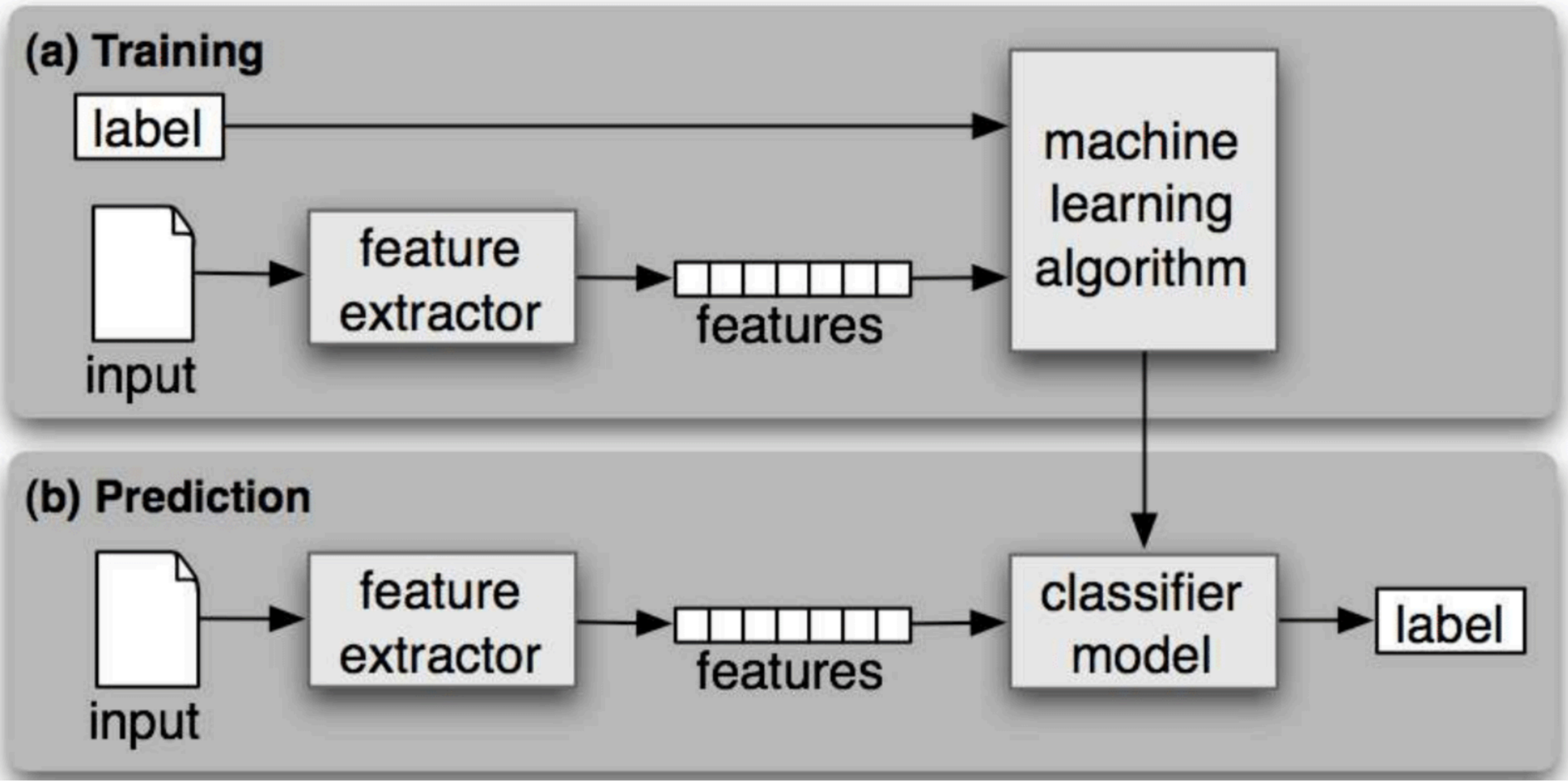


Clustering Example



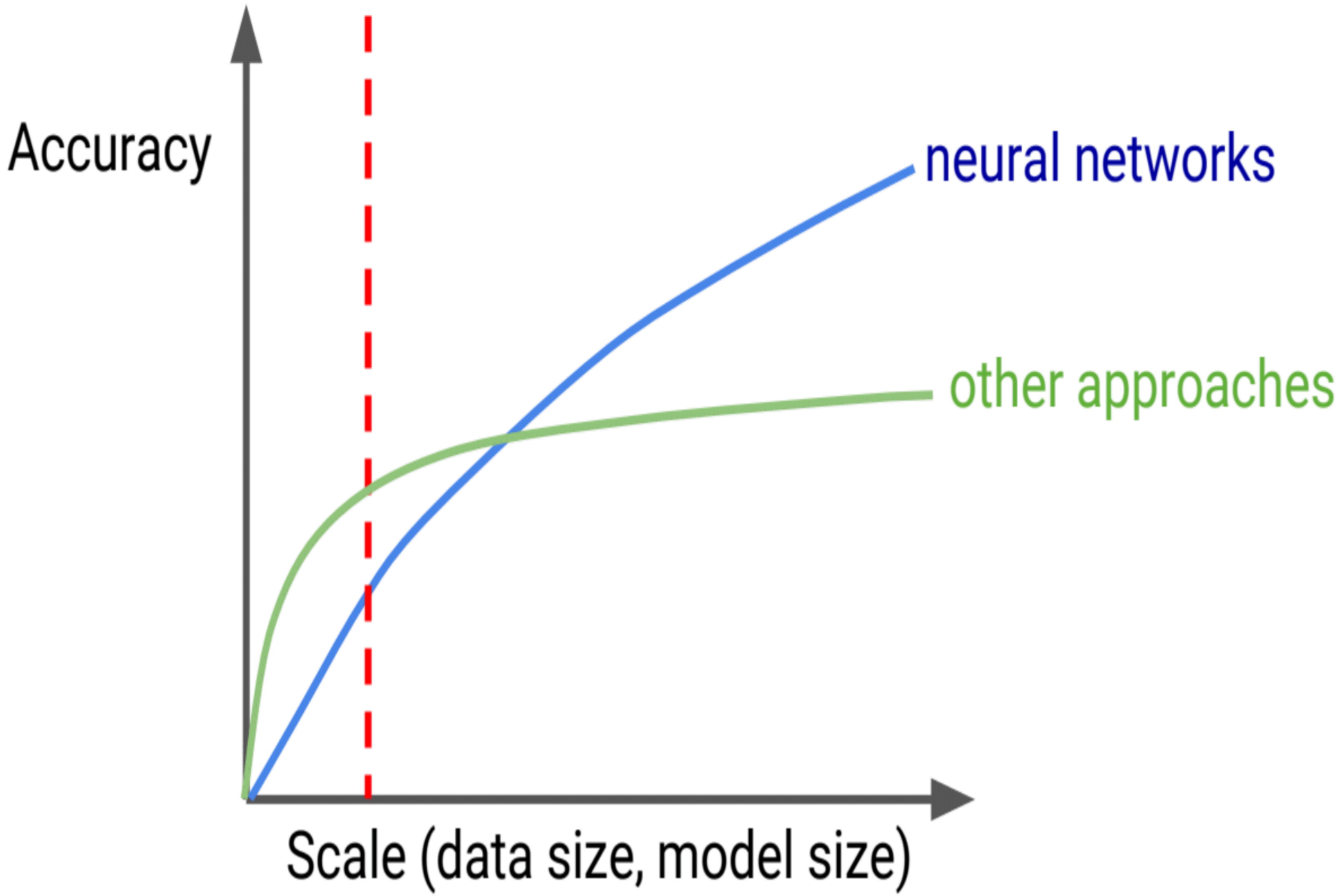
http://pypr.sourceforge.net/kmeans.html

The Process

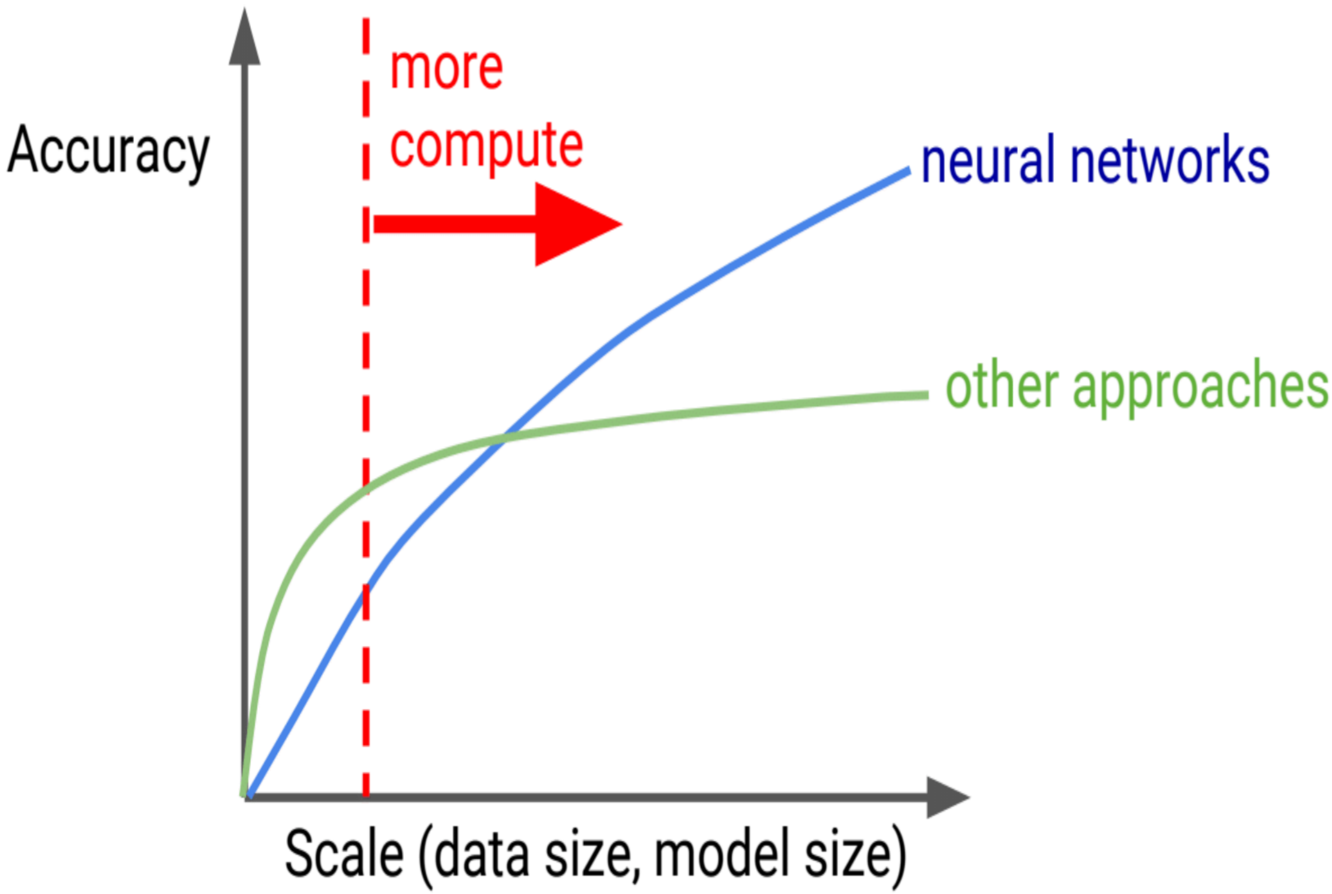


* Jain, Apache/Solr Data Analytics Group

1980s and 1990s

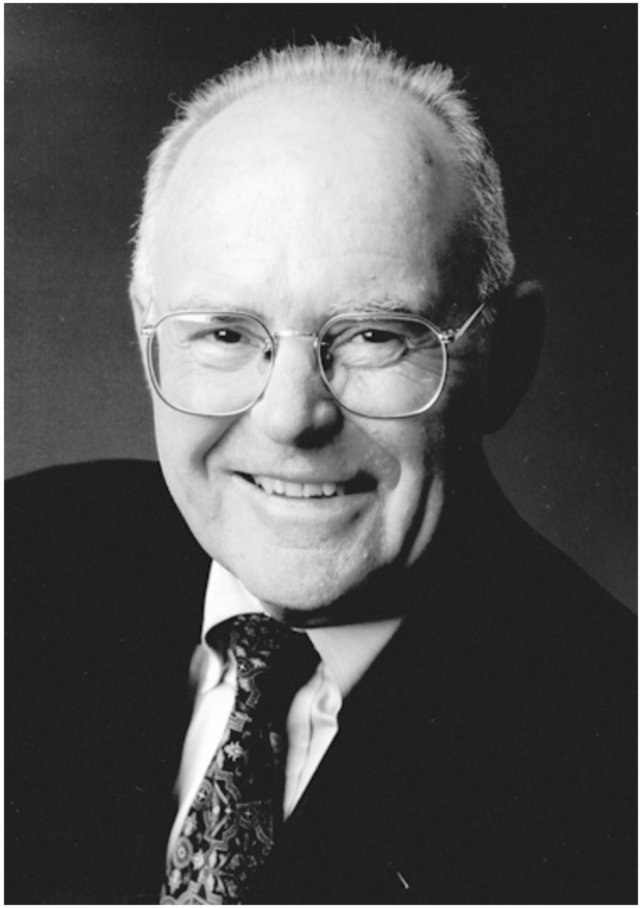


1980s and 1990s

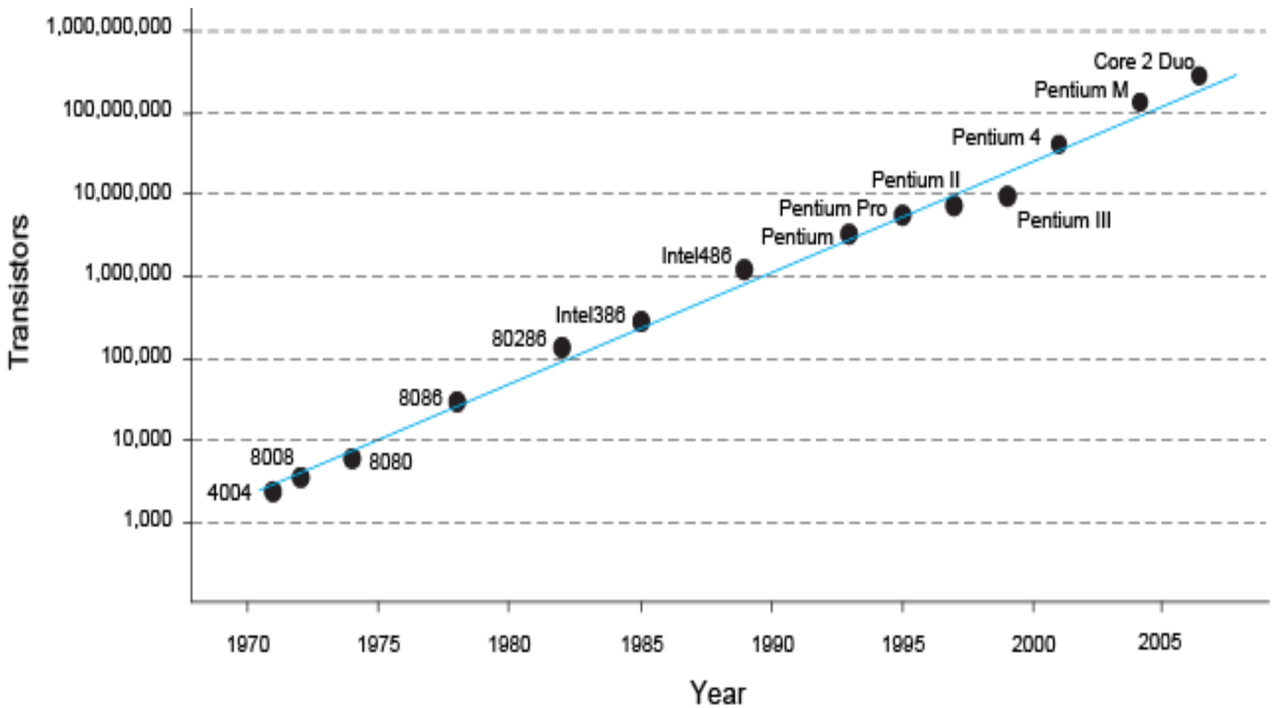


Gordon Moore, 1929 -

- Cofounded Intel in 1968 with Robert Noyce.
- **Moore's Law:** the number of transistors on a computer chip doubles every year (observed in 1965)
- Since 1975, transistor counts have doubled every two years.



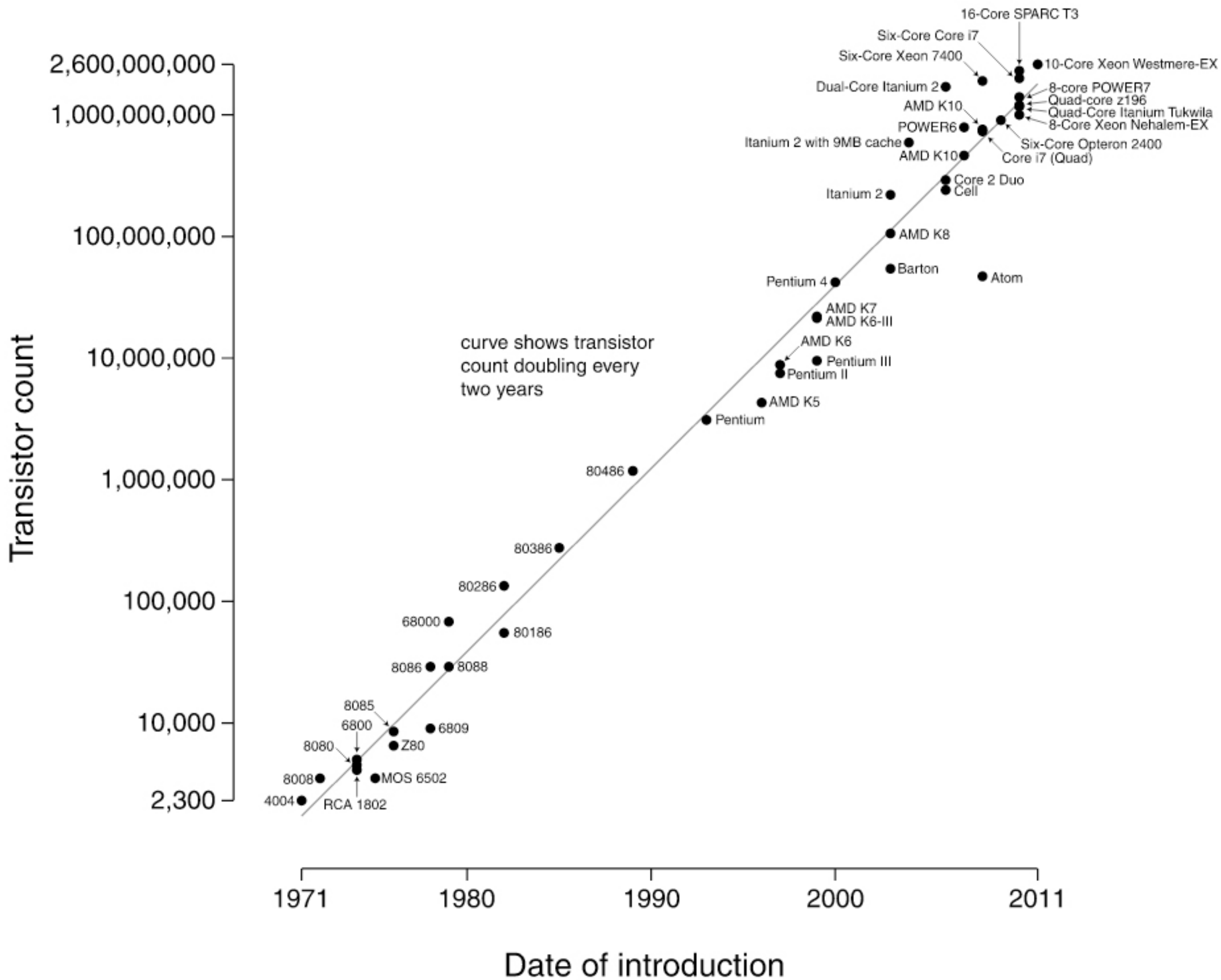
Moore's Law



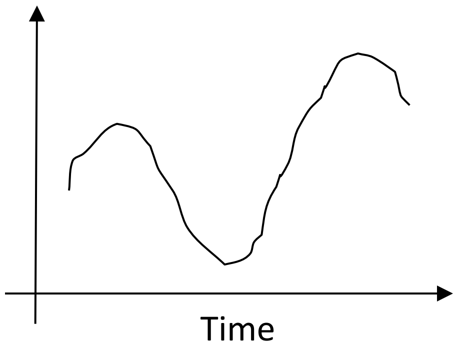
- *"If the automobile had followed the same development cycle as the computer, a Rolls-Royce would today cost $100, get one million miles to the gallon, and explode once a year . . ."*

  – Robert Cringley

Microprocessor Transistor Counts 1971-2011 & Moore's Law



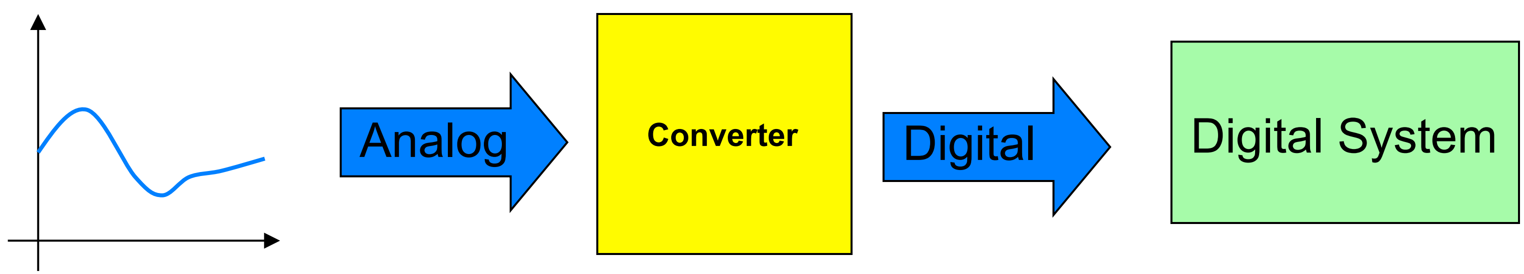
First - What Digital Logic is Not!

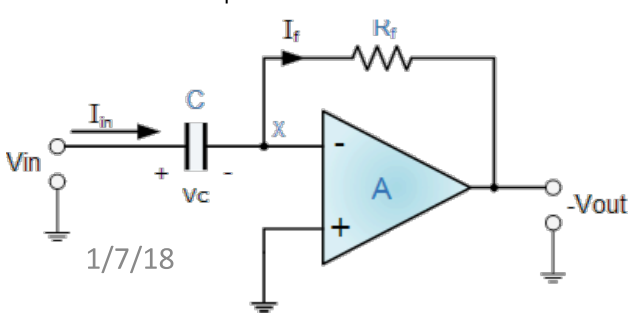


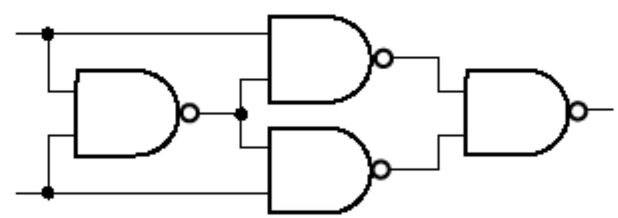
Real “life” signals are actually analog in nature

- Computing systems integrate with the real world by quantizing and approximating at the interfaces – converting ANALOG to DIGITAL [Meaning 0’s and 1’s]

- Digital solutions offer significant advantages in terms of performance, power, area, predictability and tools - enabling complex design solutions.

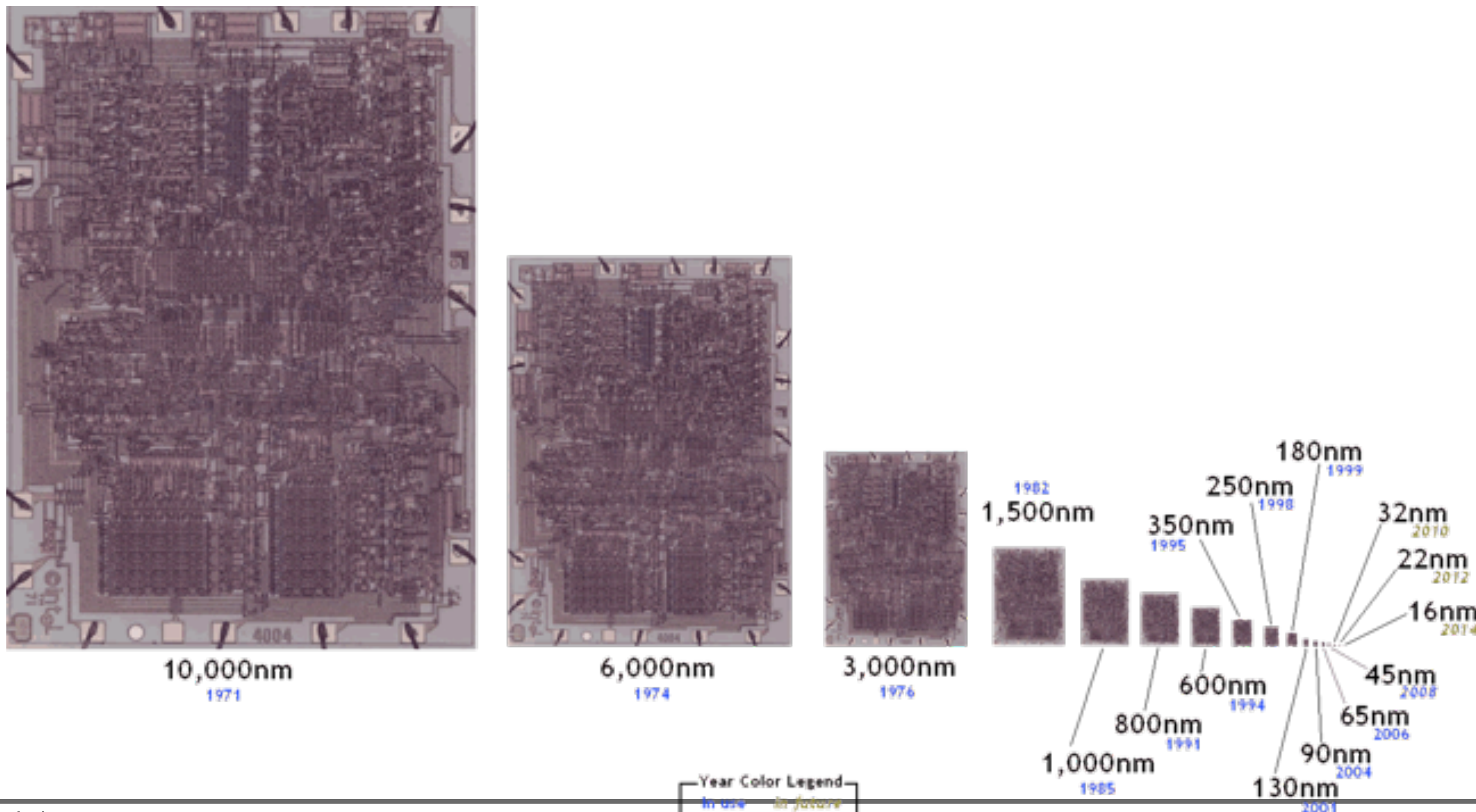






The Microprocessor

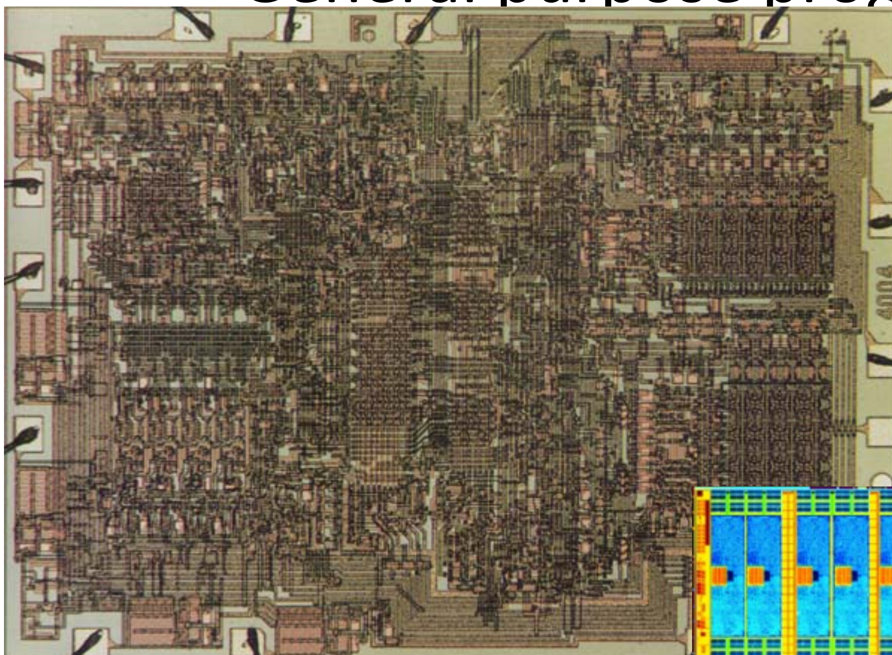
More compute power in a smaller area



Courtesy of Nabil Chouba

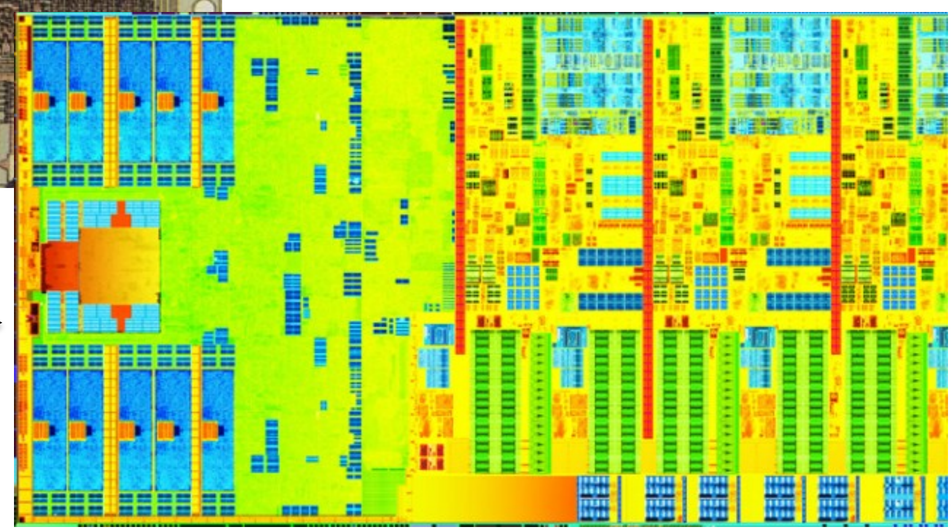
The Microprocessor

- 1971: Intel introduces the 4004
  - General purpose programmability



2,300 Transistors

Haswell – 2013

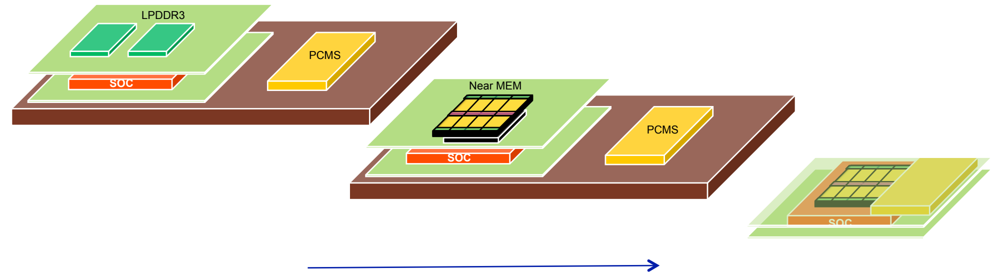


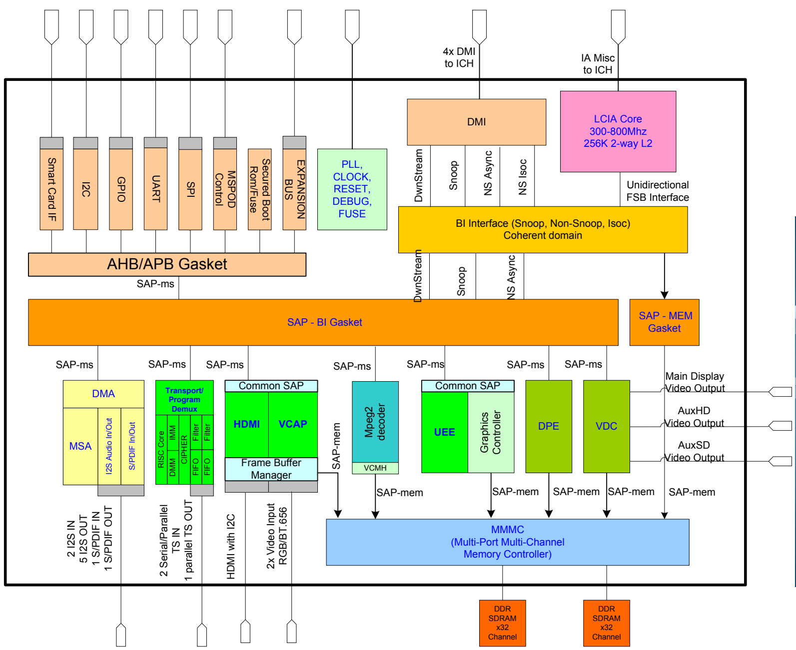
1.4 Billion Transistors

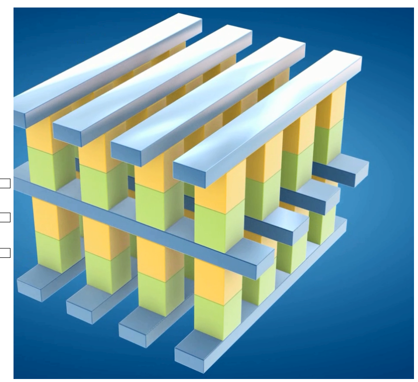
Die photos courtesy of Intel© Corporation

Increasing levels of Integration

- Today's IC incorporate IP from across the globe leveraging varying degrees of partnerships
- Density gains are exponentially driving this trend – both in number of IP and complexity of each IP
- New usages are emerging

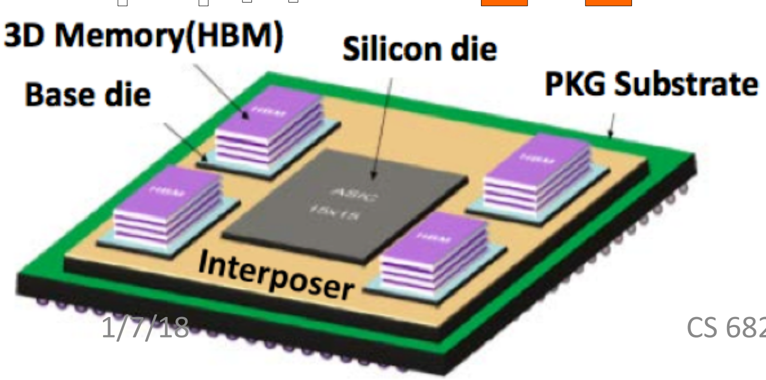




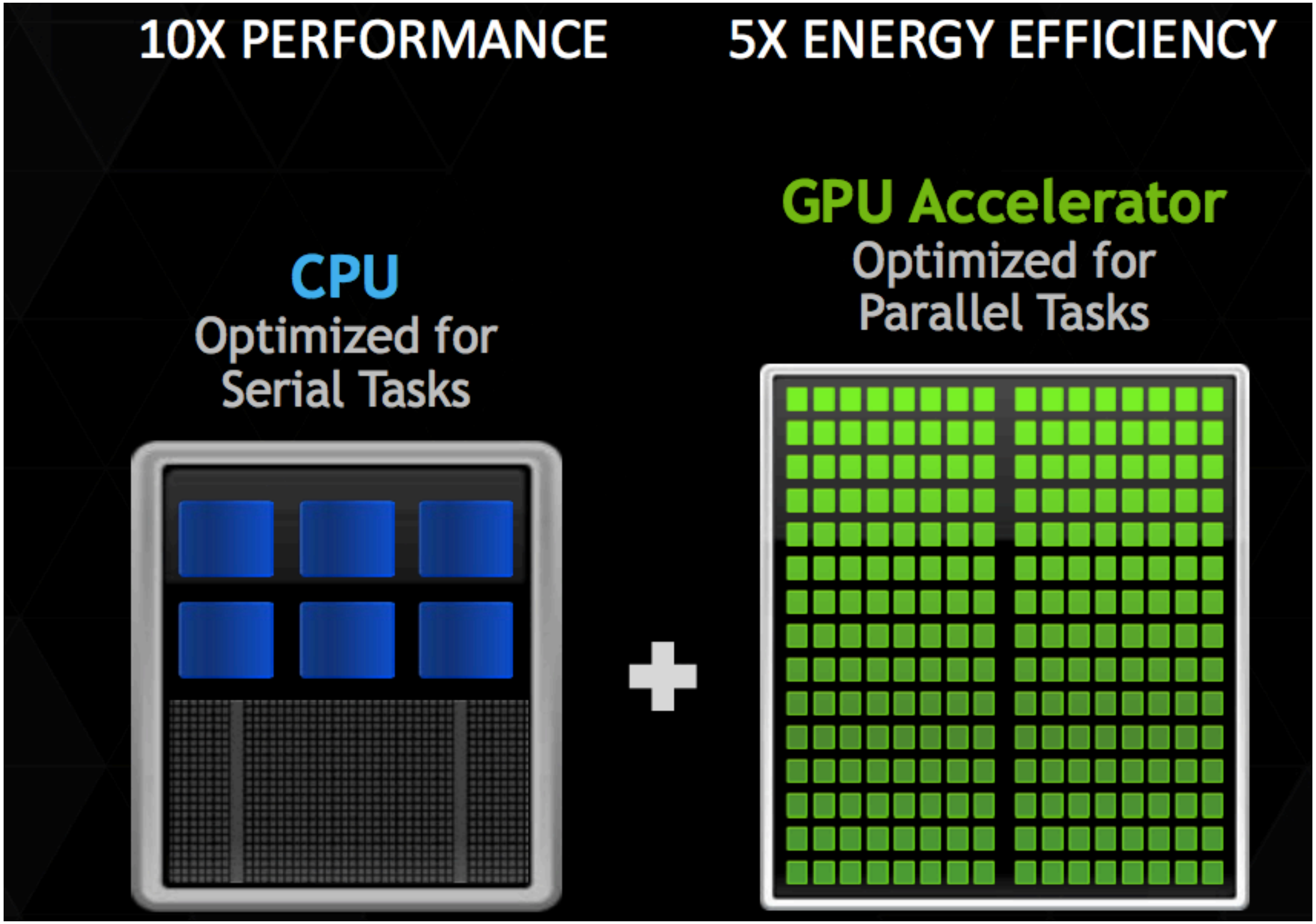


New Enabled Technologies

- MEMs
- Remote sensors
- "Smart Dust" & IoT
- Mobile applications reaching Terascale capabilities
- Increased dependency on automation
- Remote programmability of deeply embedded hardware components in the Data Center



GPUs and Fined Grain Architectures



*Nvidia

Machine Learning – Definition Through Examples

- How does our brain process the images?
- How do we grouping happen and "tag"?
- Human undergo a "Learning" process.
- After learning the brain kind off looks at new images and groups or "classifies".
- In Machine Learning we write programs for learning and then classification.

Applications of Machine Learning

- Banking
- Telecom
- Retail
  - Potential customers
  - Dissatisfied customers
  - Targeted advertising
- Financial Services
  - Good customers
  - Bad payers
  - Fraud risks
  - Reduced churn

Applications of Machine Learning

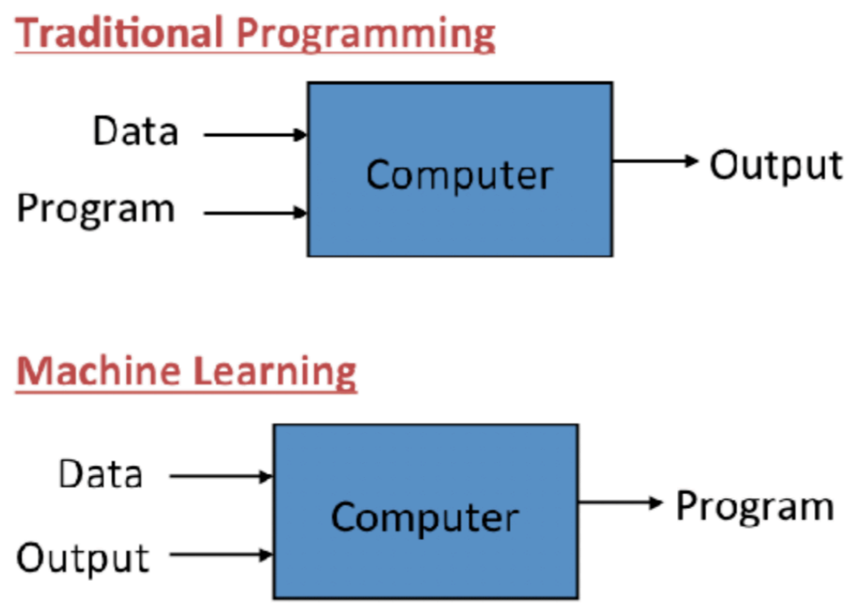
- Biomedical
  - Biometrics
  - Development of Medicines - Discovery
  - Screening
  - Diagnosis and prognosis

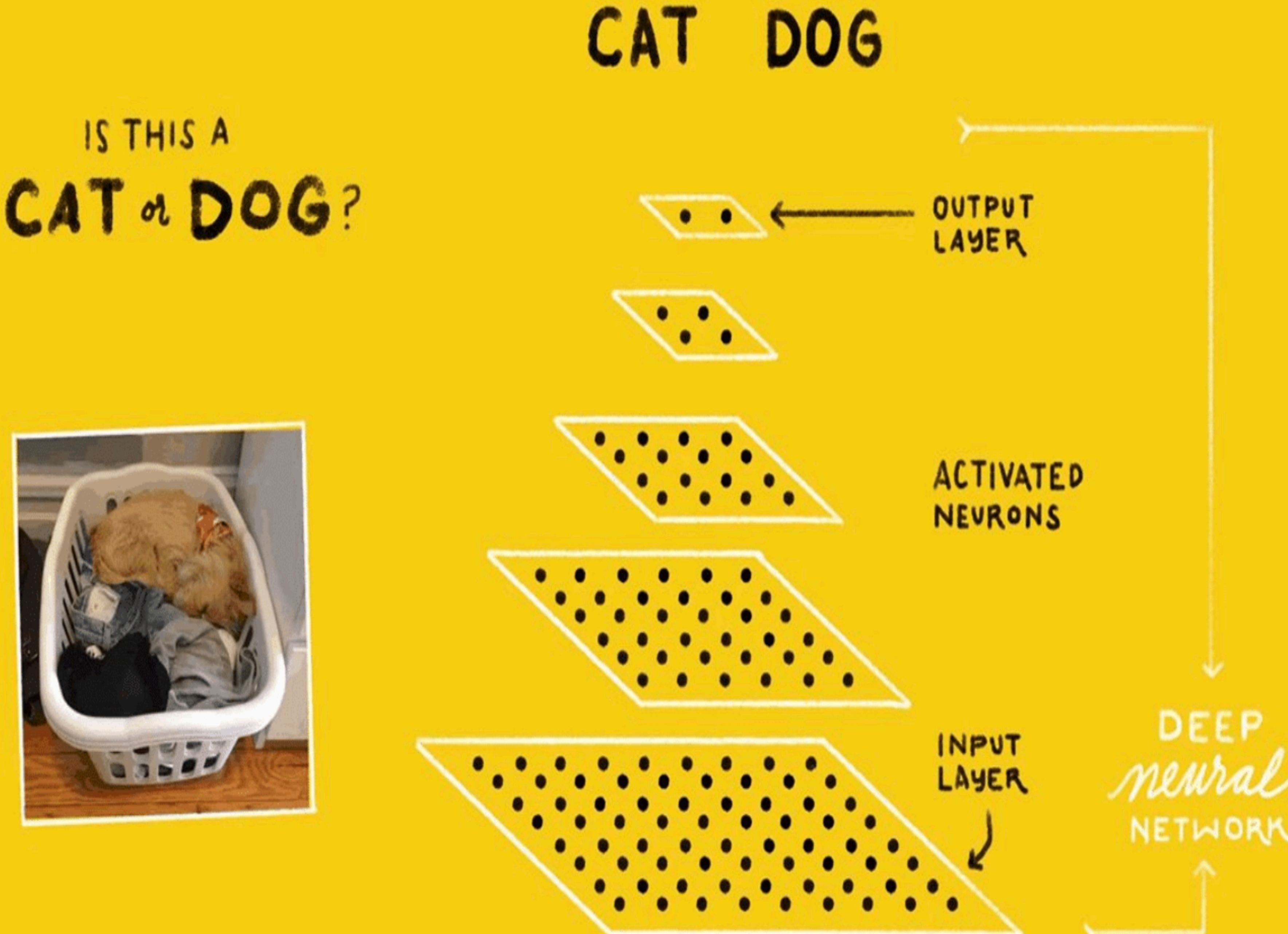
Applications of Machine Learning

- Computer / Internet / Data Warehouse
  - Troubleshooting
  - Failure detection and prediction
  - Handwriting and speech recognition
  - Hit ranking
  - Spam filtering

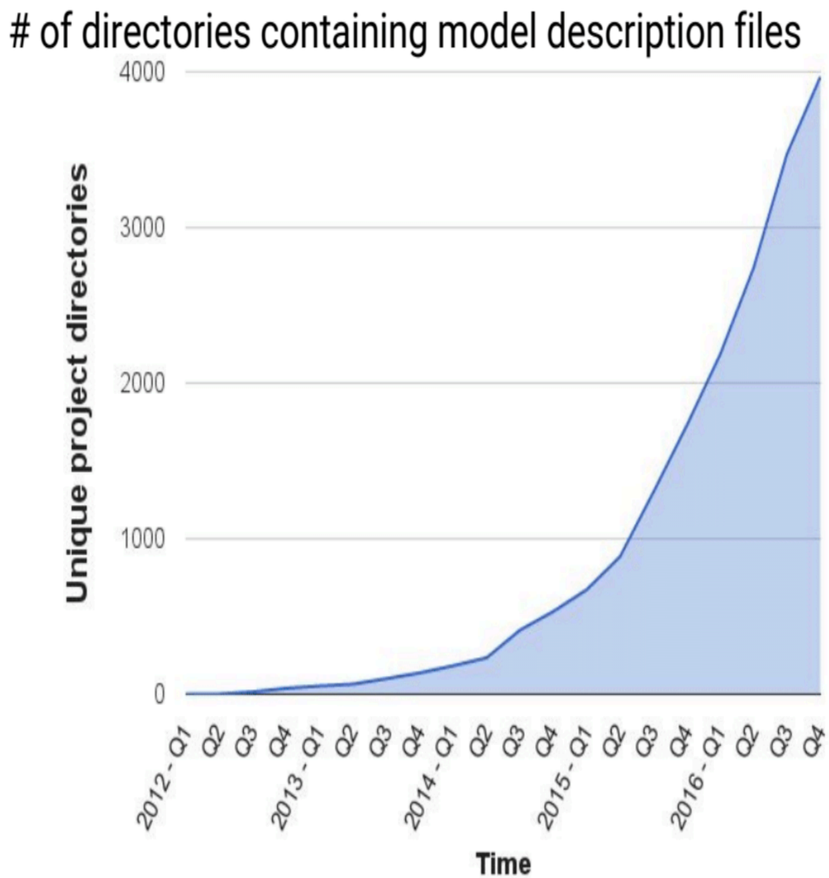
Anatomy of a Machine Learning Solution

- Any Machine Learning algorithm has three parts
  - The Output (prediction)
  - The Objective Function or Performance Matrix
  - The Input (samples)




CAT DOG
IS THIS A
CAT or DOG?
OUTPUT LAYER
ACTIVATED NEURONS
INPUT LAYER
DEEP
neural
NETWORK

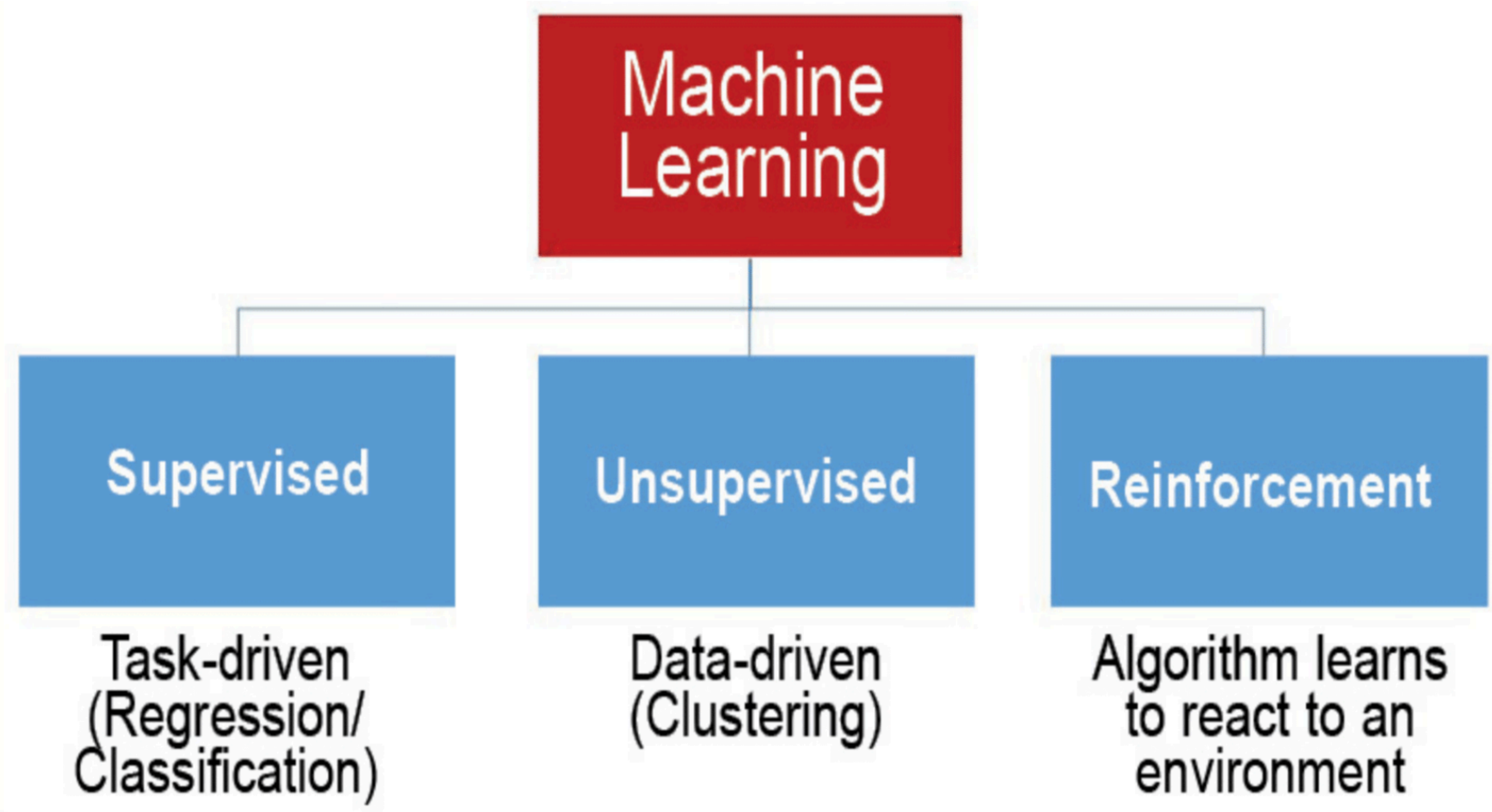
Interesting Fact



Across many products/areas:

Android
Apps
drug discovery
Gmail
Image understanding
Maps
Natural language understanding
Photos
Robotics research
Speech
Translation
YouTube
... many others ...

Categories of Machine Learning



Categories of Machine Learning

- Supervised Learning – Given examples of inputs and the desired categories, predict the outputs of future samples
  - Classification
  - Regression
  - Time series prediction

- Unsupervised Learning: Given only inputs (no categories), automatically discover representations, correlations, features, structure, etc.
  - Clustering
  - Outlier detection
  - Compression

- Reinforcement Learning
  - allows machines and software agents to automatically determine the ideal behavior within a specific context, in order to maximize its performance.

Supervised Learning

- Given: Training set $\{(x_i, y_i) \mid i = 1 \ldots N\}$
- Find: A good approximation to $f : X \rightarrow Y$

example										label
				train						
aardvark	→	1	1	18	4	22	1	18	11	−
cow	→	3	15	23						+
giraffe	→	7	9	18	1	6	6	5		−
termite	→	20	5	18	13	9	20	5		−
oyster	→	15	25	19	20	5	18			+
dove	→	4	15	22	5					−
spider	→	19	16	9	4	5	18			+
dog	→	4	15	7						+
elephant	→	5	12	5	16	8	1	14	20	+
				test						
rabbit	→	18	1	2	2	9	20			
frog	→	6	18	15	7					
kangaroo	→	11	1	14	7	1	18	15	15	

*Schapire

The Training Process