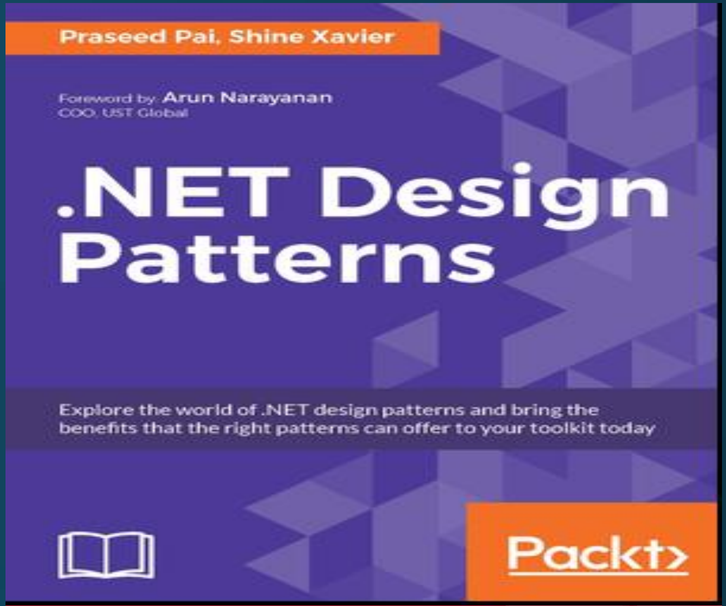
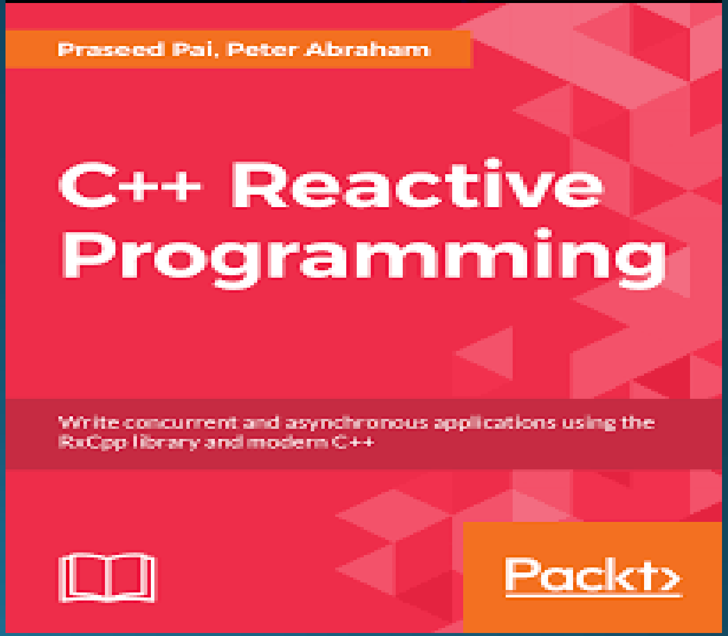
Machine Learning – An Overview

About the Trainer

- Co-Author of books titled,".NET Design Patterns" and "C++ Reactive Programming" – Packt Publishing
- The Primary Author of SLANGFORDOTNET Compiler Infrastructure System
- Published a University accredited paper on Ontology and Softwrae Engineering
- A specialist in "Cross Cultural encounters" in large software projects
- Has designed a Couse titled, "Philosophical Tools for Software Engineering" (Presented @ RubyConf India)
- Presented in more than 250 events in the past twenty five plus years





Who is an Architect?

Architect (n) – Any person who has “fooled” around in the Software Industry for a sizeable period of time (ever shrinking span) who is past his prime, as a Programmer Or Engineer, systematically moved up in the hierarchy to obey “Peter Principle”.

Who can seek Knowledge?

"**A science is any discipline** in which the **fool** of this generation can go beyond the point reached by the genius of the last generation"

- Max Gluckman , South African Antrhopologist

Agenda

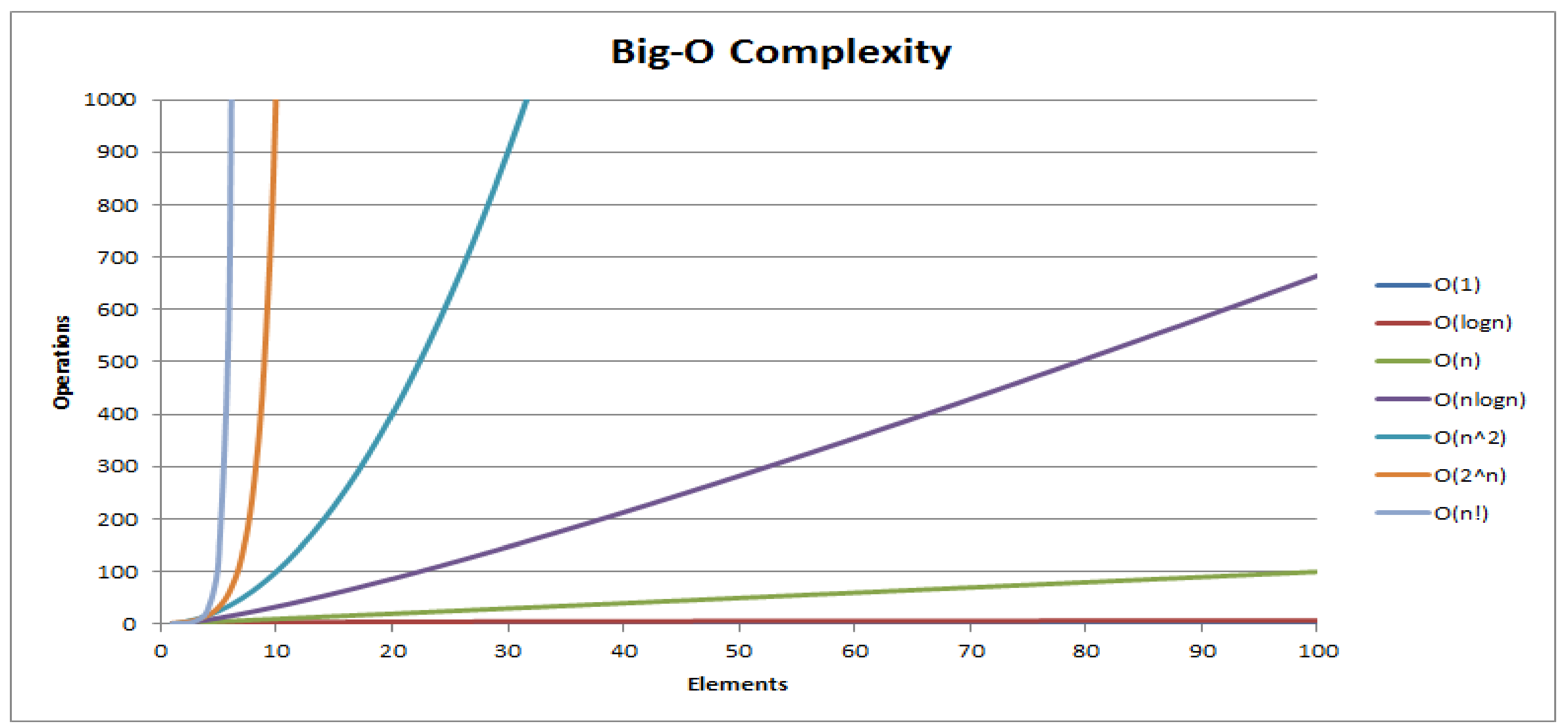
- Machine Learning – What/Why/How of the Discipline?
  - When should we use?
  - It's Lingo and Direction where it is heading
- Machine Learning – Technology/Algorithm and in Practice
  - Key models/methods & Algorithms which mostly works
  - Technology spectrum
  - It's Limits
- Machine Learning – Business/Practice Perspective
  - Operational Realm
  - Engagement Models , Delivery norms and Hedging against Risks
- Q&A

Machine Learning – What/Why/How of the Discipline (Part 1)

What is Machine Learning – Tom Mitchel's Definition

"A computer program is said to learn from experience E with respect to some class of tasks ***T*** and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."

Algorithmic Complexity



Machine Learning – When Should we use?

After you have tried
If/else/while/select/update programming
Linear/Quadratic complexity algorithms
Patented algorithms
Heuristics based solutions
Approximate Solutions
Stochastic solutions

All of the above are control path programming. When these fail you can opt for Machine learning based solutions which will reason based on data.

ML – Key Ideas

- It is all about Learning from data
- Assumption is that, there is pattern in data (need not be true, always)
- Three Key Learning models are Supervised (mostly Classification), Unsupervised (Clustering Algorithms) and Association (Apriori and Correlation)
- Deep Learning – Neural network with multiple hidden layers (a practical definition)
- Probabilistic Graphical Models (Evidence Based AI)

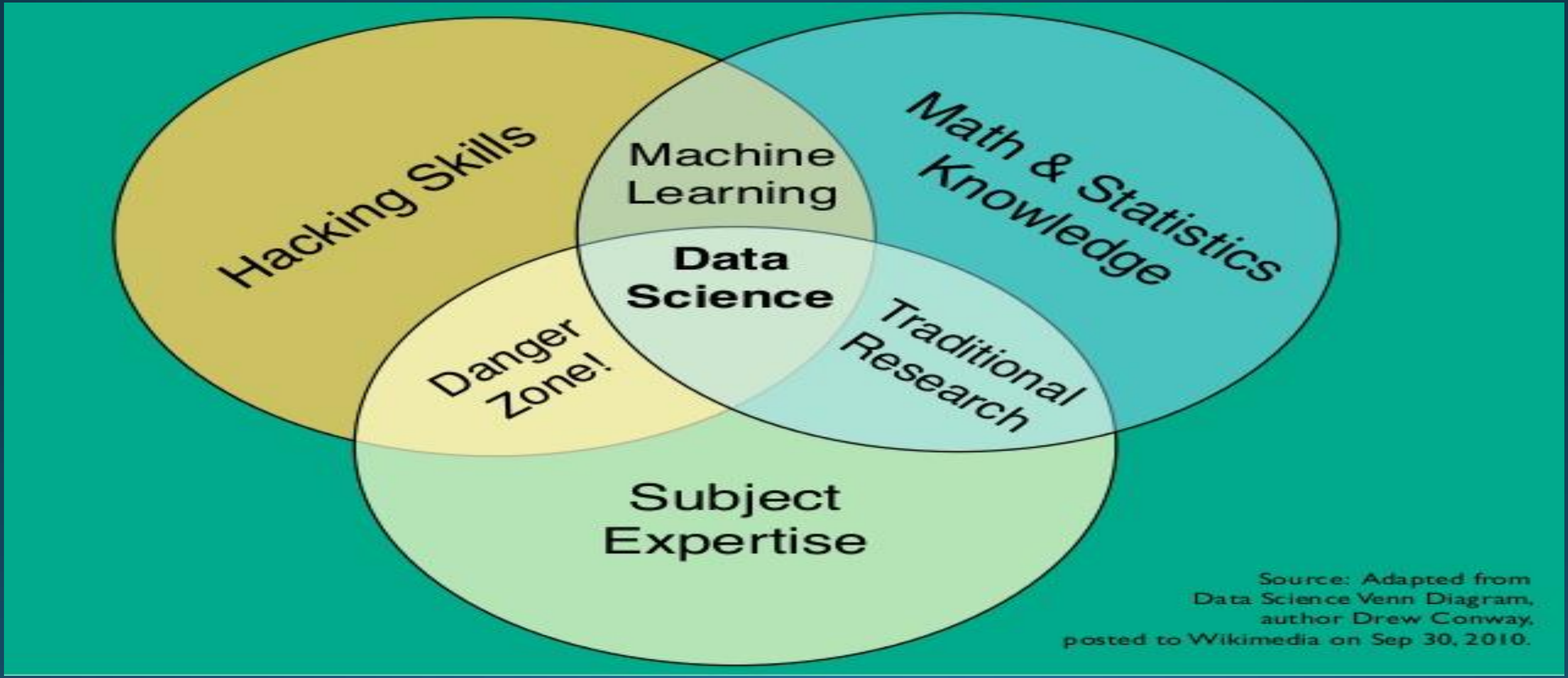
Dissecting Analytics

- Analysis/Synthesis Model of Problem Solving
- A Top Decomposition of the Problem into Parts to a granular level , until we have reached a state where we cannot decompose parts further or it has become fine-grained to be amenable for studying it.
- A Bottom up process of Synthesis
- In Western Philosophy and Science, Rene Descartes is regarded as the father of modern Analysis
- Reductionism vs Holism – Analytic Thinking vs System Thinking
- Assumption of Independence of Variables and Interdependence of Variables
- Additive factors (Linear) vs Non Linear Factors

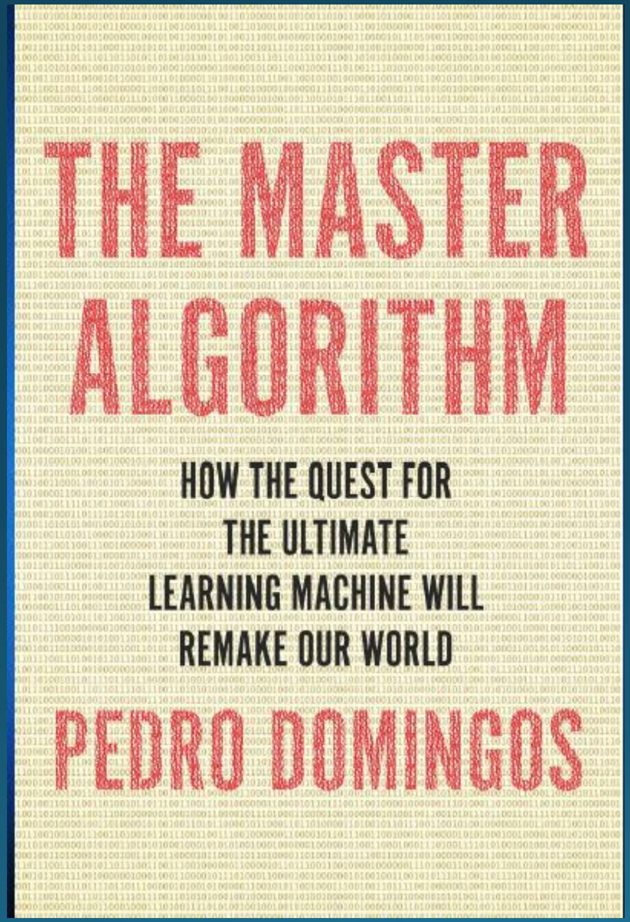
ML – How much math?

- Linear Algebra
  - Matrix Inversion, Eigen Values , SVD (emphasis on reading notation)
  - Quadratic Programming (Notion should be familiar … "Trust the Libraries")
- Calculus of one variable and elementary partial derivatives
  - Computation of the Gradient Descent
  - Derivation of certain results
  - Numerical Computation
- Probability and Statistics
- More advanced mathematics is mostly used for "Distinguishing between Cats and Dogs", tasks which are trivial for Human beings, which are difficult for computing machines (Moravec Paradox)

Drew Conway's Venn Diagram



ML – Where it is heading? (a potential path)



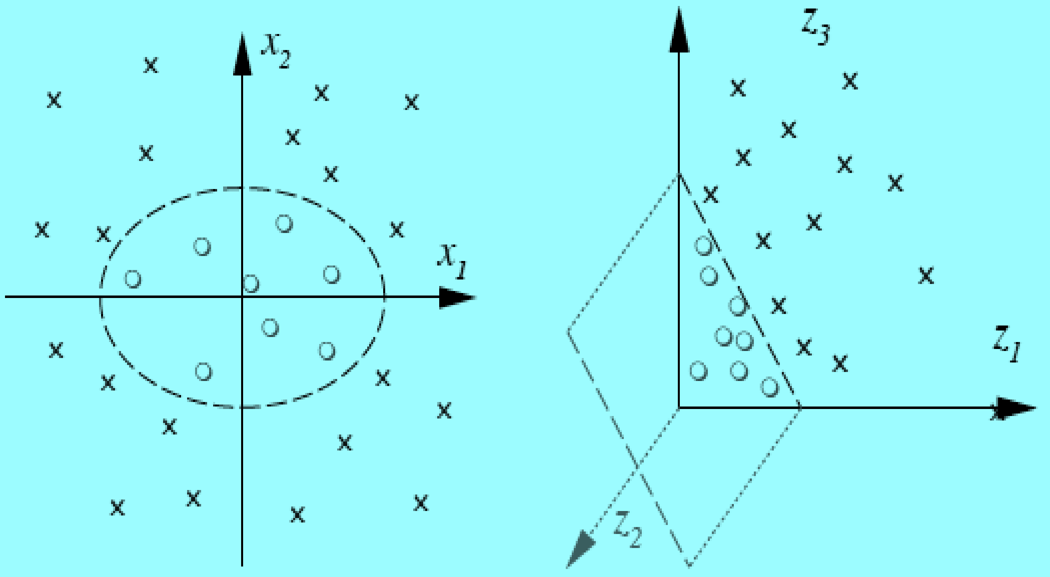
Machine Learning – Technology/Algorithm and in Practice (Part 2)

Algorithmic Techniques

- Hilbert Space Methods
- Statistical Learning
- Deep Learning

Hilbert Space Methods

$$\Phi : R^2 \to R^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt(2)x_1x_2, x_2^2)$$



Hilbert Spaces

A real Hilbert Space X is endowed with the following operations:

1. Vector addition: $\mathrm{x} + \mathrm{y}$
2. Scalar multiplication: $a\mathrm{x}, \quad a \in \Re, \mathrm{x} \in X$
3. Inner product $\langle \mathrm{x}, \mathrm{y} \rangle \in \Re$, with properties:

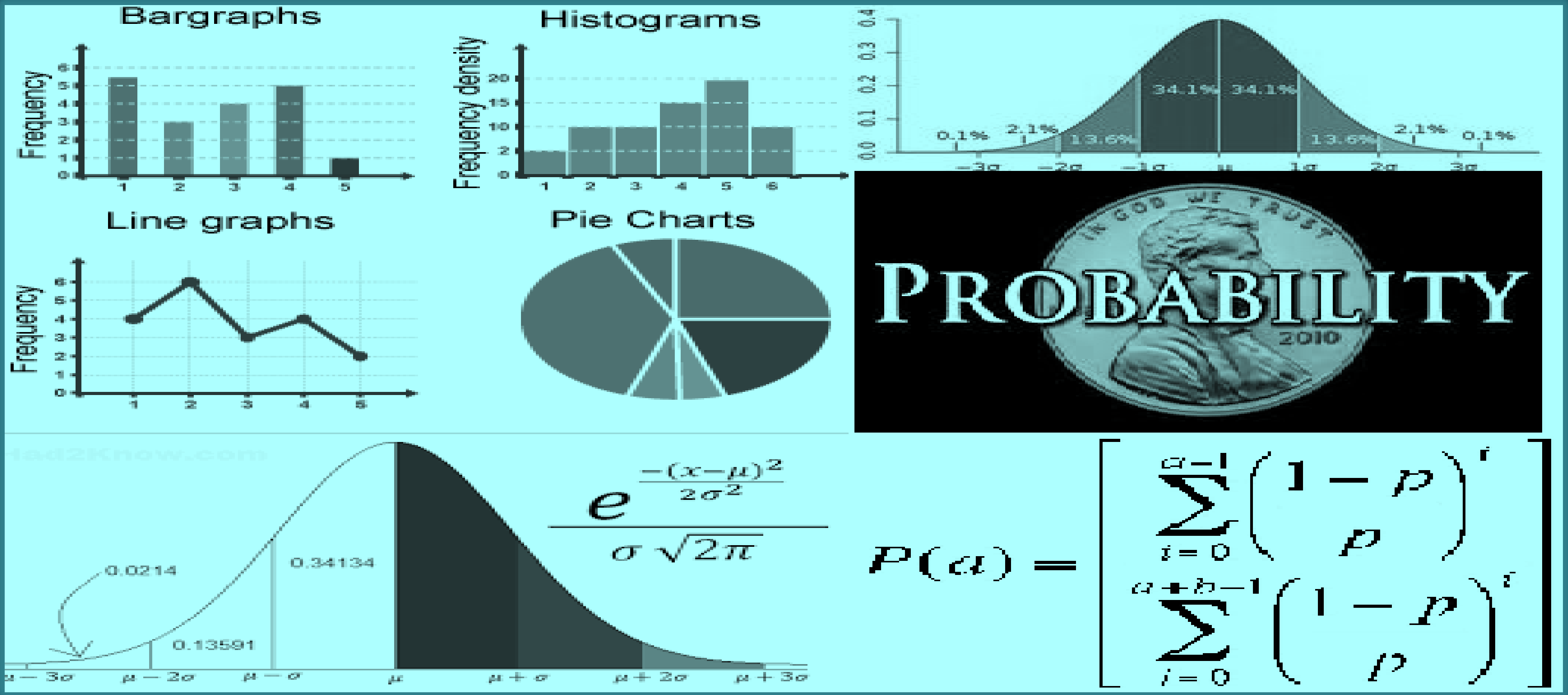
$$\langle \mathrm{x}, \mathrm{y} \rangle = \langle \mathrm{y}, \mathrm{x} \rangle \qquad \langle a\mathrm{x} + b\mathrm{y}, \mathrm{z} \rangle = a\langle \mathrm{x}, \mathrm{z} \rangle + b\langle \mathrm{y}, \mathrm{z} \rangle \qquad \langle \mathrm{x}, \mathrm{x} \rangle \geq 0$$

4. Norm $||\mathrm{x}|| = \langle \mathrm{x}, \mathrm{x} \rangle^{1/2} \qquad ||\mathrm{x}|| = 0 \Leftrightarrow \mathrm{x} = 0$

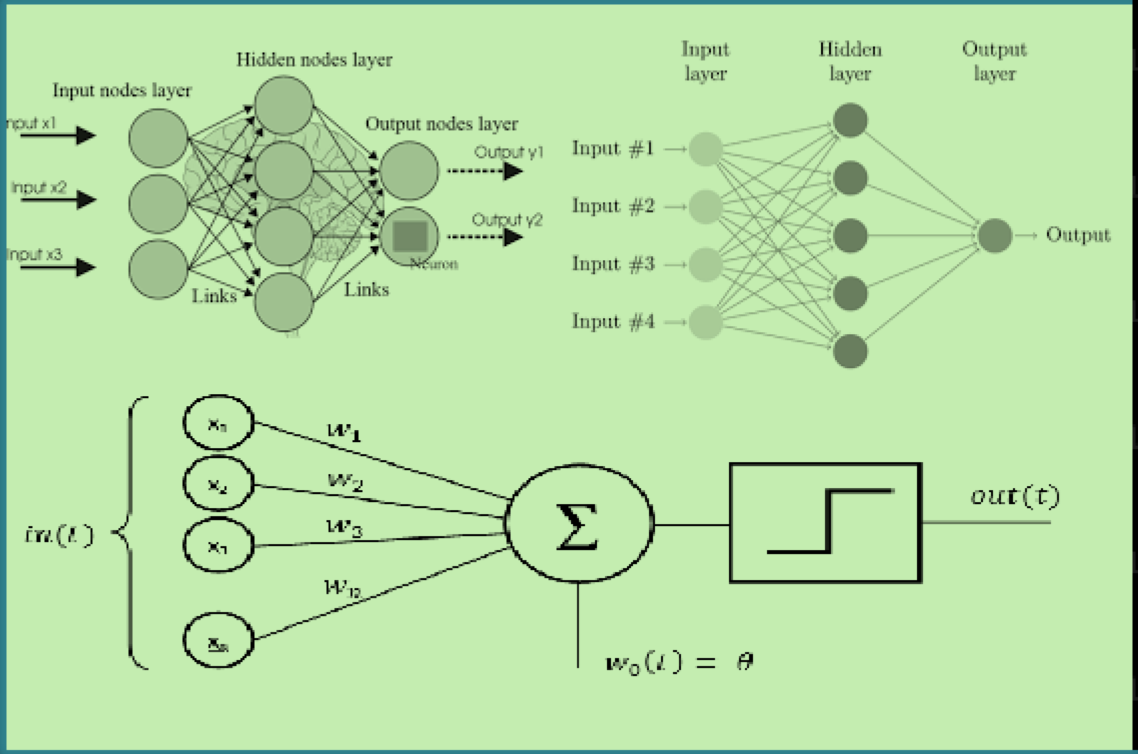
Basic facts of a Hilbert Space X

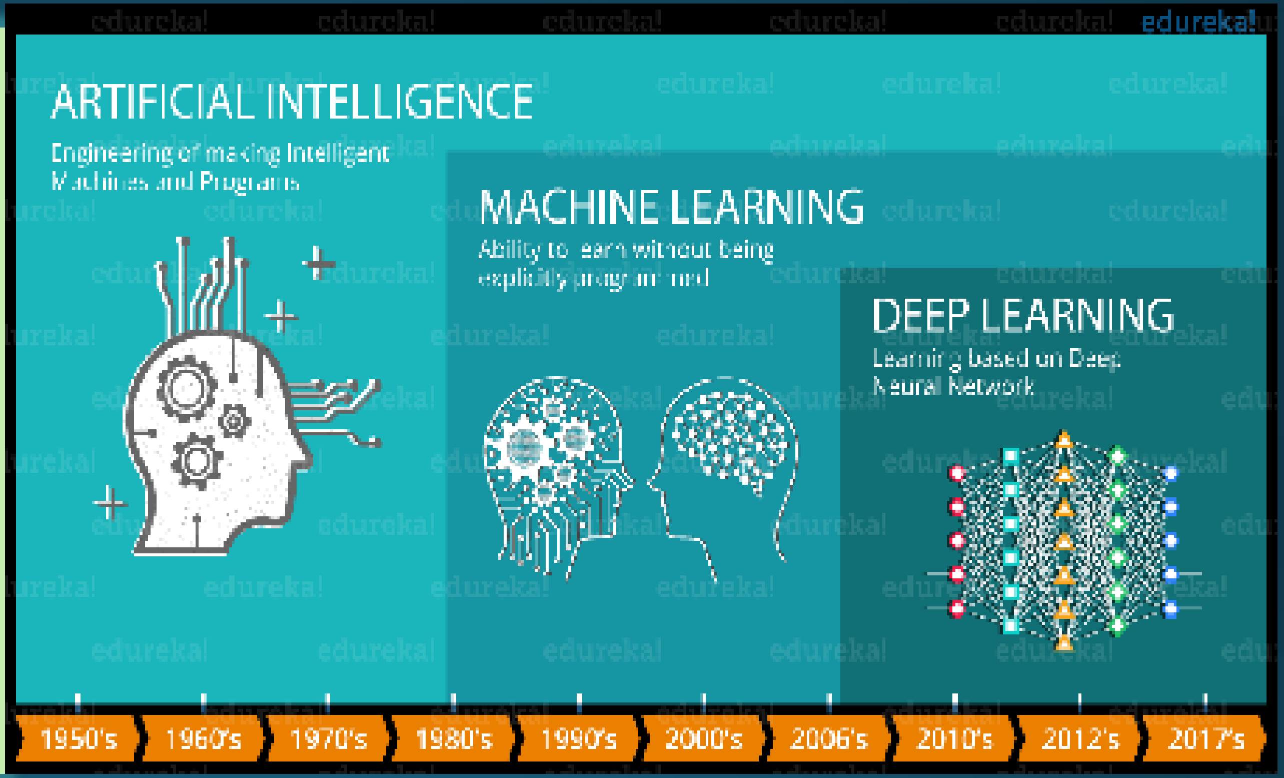
1. X is complete
2. Cauchy-Schwarz inequality $|\langle \mathrm{x}, \mathrm{y} \rangle| \leq ||\mathrm{x}|| \, ||\mathrm{y}||$ where the equality holds if and only if $\mathrm{x} = \lambda \mathrm{y}$

Statistical Methods



Deep Learning Methods





Apriori Algorithm

Transaction ID	Items Bought
1	Shoes, Shirt, Jacket
2	Shoes,Jacket
3	Shoes, Jeans
4	Shirt, Sweatshirt

If the *minimum support* is 50%, then {Shoes, Jacket} is the only 2- itemset that satisfies the minimum support.

Frequent Itemset	Support
{Shoes}	75%
{Shirt}	50%
{Jacket}	50%
{Shoes, Jacket}	50%

$$confidence(A \Rightarrow B) = \frac{\#\_tuples\_containing\_both\_A\_and\_B}{\#\_tuples\_containing\_A}$$

If the *minimum confidence* is 50%, then the only two rules generated from this 2-itemset, that have confidence greater than 50%, are:

$$support(A \Rightarrow B) = \frac{\#\_tuples\_containing\_both\_A\_and\_B}{total\_\#\_of\_tuples}$$

Shoes $\Rightarrow$ Jacket Support=50%, Confidence=66%
Jacket $\Rightarrow$ Shoes Support=50%, Confidence=100%

The Apriori Algorithm — Example

Min support =50%

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan D

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

Scan D

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_3

itemset
{2 3 5}

Scan D

L_3

itemset	sup
{2 3 5}	2

$X = (x1,x2,x3 .....xn)$

$W = (w1,w2,w3......wn)$

$Sum(wi,xi) = w1*x1 + w2*x2 + w3*x3.......wn*xn;$

In Statistical Methods

(w1...wn) might be the probability of each coefficient

In Hilbert Space Methods

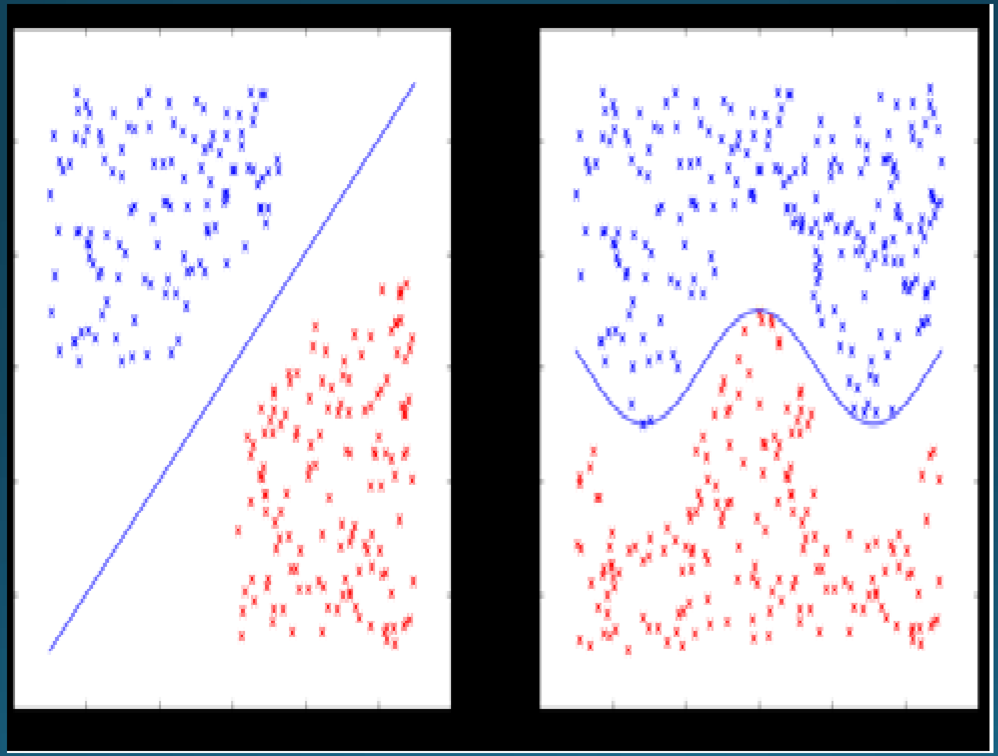
(w1....wn) defines a hyperplane which partitions data

In Deep Learning

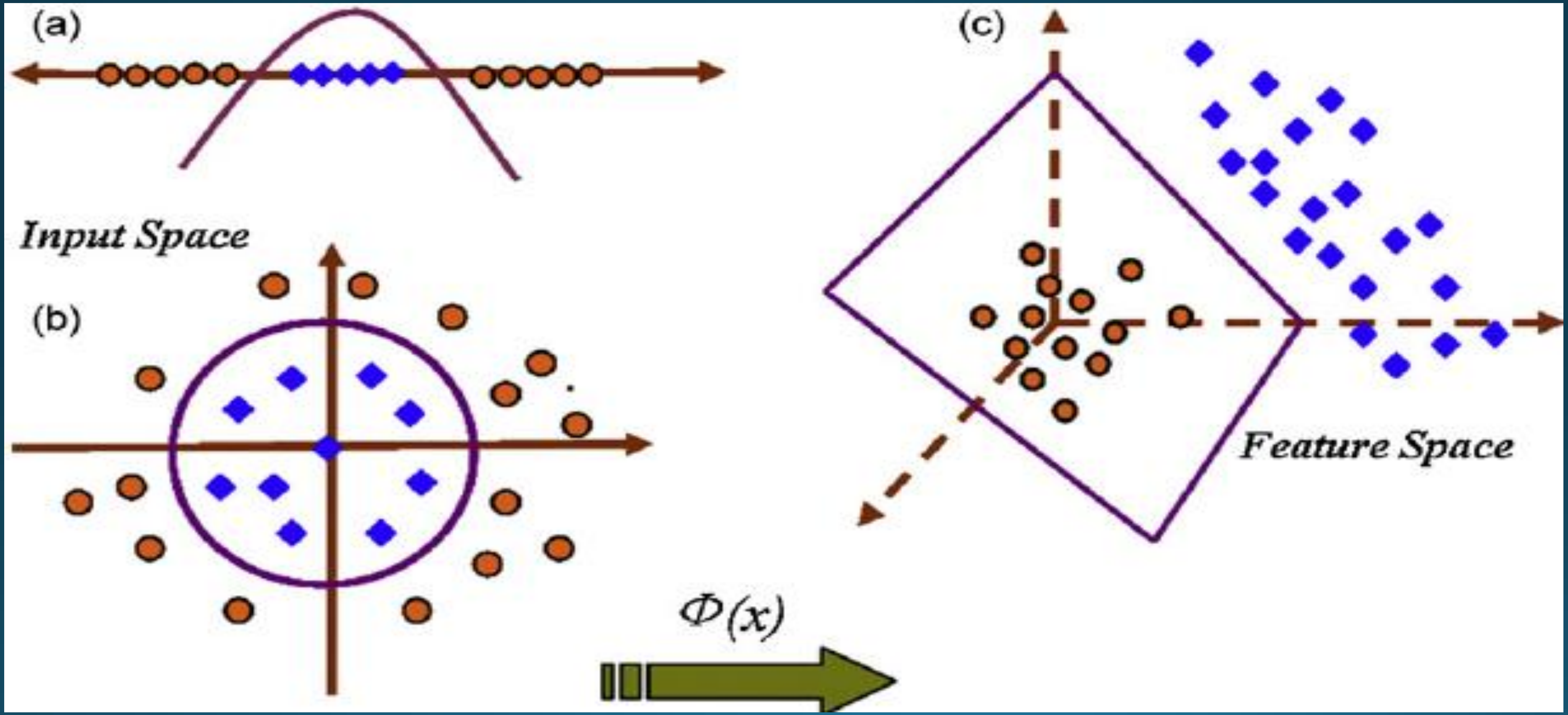
(w1....wn) defines the Weights of Input neuron

In a way, the weight is considered as the degree of influence of each variable on the target. Learning is about finding the weights of each co-efficients of equation

It is all about Linear Seperability



Linear Seperability @ Higher dimensions



Linear Algebra for “Rookies”

Learn Matrix Algebra through Python

```
import numpy as np
# Defining the matrices
A = np.matrix([[3, 6, -5],[1, -3, 2],[5, -1, 4]])
B = np.matrix([[12],[-2], [10]])
# Solving for the variables, where we invert A
X = A ** (-1) * B
print X
x = np.linalg.inv(A).dot(B)
print x
```

Eigen Value - Intuition

Is $\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$ an eigenvector of $A = \begin{bmatrix} 3 & 6 & 7 \\ 3 & 3 & 7 \\ 5 & 6 & 5 \end{bmatrix}$. If yes, find the corresponding eigenvalue.

Eigenvector of Matrix **A**

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$$

Eigenvalue of Matrix **A**

$$\begin{bmatrix} 3 & 6 & 7 \\ 3 & 3 & 7 \\ 5 & 6 & 5 \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 3-12+7 \\ 3-6+7 \\ 5-12+5 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix}$$

$$\mathbf{A} \cdot \mathbf{v}_1 = \lambda_1 \cdot \mathbf{v}_1$$

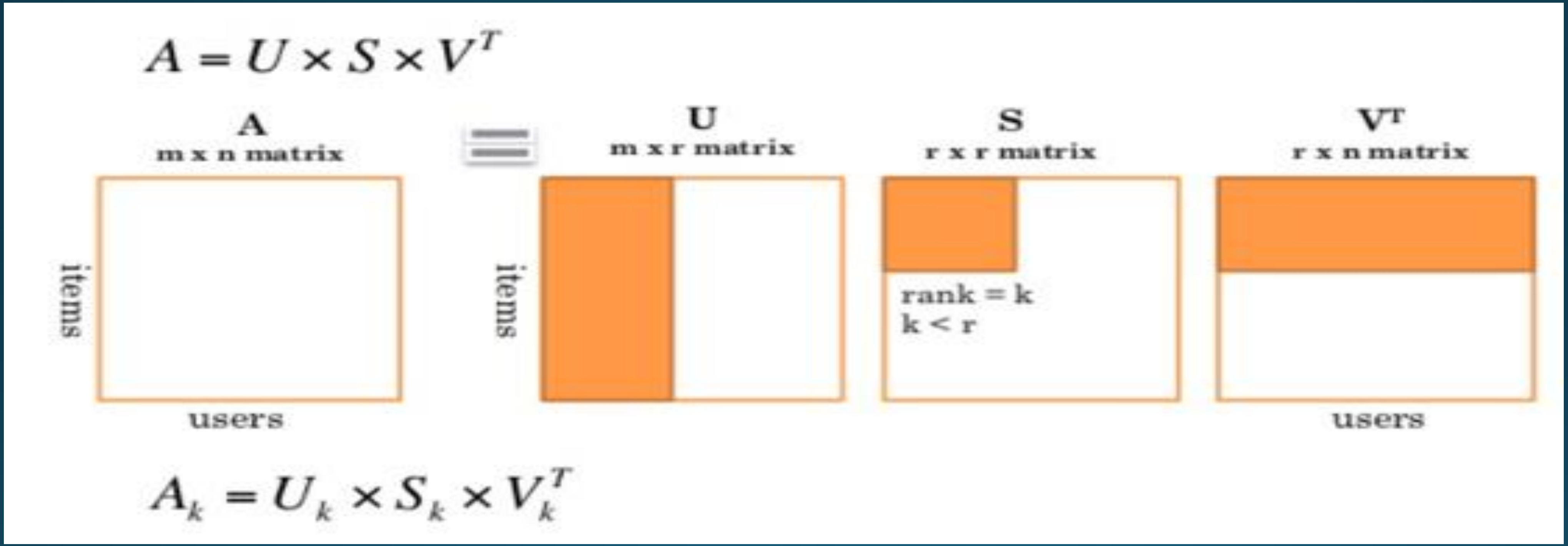
$$(\mathbf{A} - \lambda_1) \cdot \mathbf{v}_1 = 0$$

The new vector is -2 times the old vector. So $\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$ is an eigenvector of A with eigenvalue $\lambda = -2$.

Eigen Value – Python Code

```
import numpy as np
A = np.mat("3 -2;1 0")
print "A\n", A
print "Eigenvalues", np.linalg.eigvals(A)
eigenvalues, eigenvectors = np.linalg.eig(A)
print "First tuple of eig", eigenvalues
print "Second tuple of eig\n", eigenvectors
for i in range(len(eigenvalues)):
    print "Left", np.dot(A, eigenvectors[:,i])
    print "Right", eigenvalues[i] * eigenvectors[:,i]
    print
```

SVD for Dimensionality Reduction



SVD and the Recommender System

SVD/MF

$$\boxed{X[n \times m] = U[n \times r]\ S\,[r \times r]\ (V[m \times r])^{T}}$$

$$\underset{m \times n}{\overset{X}{\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & \\ \vdots & \vdots & \ddots & \\ x_{m1} & & & x_{mn} \end{pmatrix}}} = \underset{m \times r}{\overset{U}{\begin{pmatrix} u_{11} & \cdots & u_{1r} \\ \vdots & \ddots & \\ u_{m1} & & u_{mr} \end{pmatrix}}} \underset{r \times r}{\overset{S}{\begin{pmatrix} s_{11} & 0 & \cdots \\ 0 & \ddots & \\ \vdots & & s_{rr} \end{pmatrix}}} \underset{r \times n}{\overset{V^{T}}{\begin{pmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \\ v_{r1} & & v_{rn} \end{pmatrix}}}$$

- **X**: *m x n* matrix (e.g., m users, n videos)
- **U**: *m x r* matrix (m users, r factors)
- **S**: *r x r* diagonal matrix (strength of each 'factor') (r: rank of the matrix)
- **V**: r x n matrix (n videos, r factor)

SVD

```
import numpy as np
A = np.mat("4 11 14;8 7 -2")
print "A\n", A
U, Sigma, V = np.linalg.svd(A, full_matrices=False)
print "U"
print U
print "Sigma"
print Sigma
print "V"
print V
print "Product\n", U * np.diag(Sigma) * V
```

Linear and Non Linear Regression

slNo	Diameter (inches)	Number of toppings	Price ($)
1	6	2	7
2	8	1	9
3	10	0	13
4	14	2	17.5
5	18	0	18

slNo	Diameter (inches)	Number of toppings	Price ($)
1	8	2	11
2	2 9	0	8.5
3	11	2	2 15
4	16	2	18
5	12	0	11

slNo	Diameter (inches)	Number of toppings	Price ($)
1	6	2	7
2	8	1	9
3	10	0	13
4	14	2	17.5
5	18	0	18

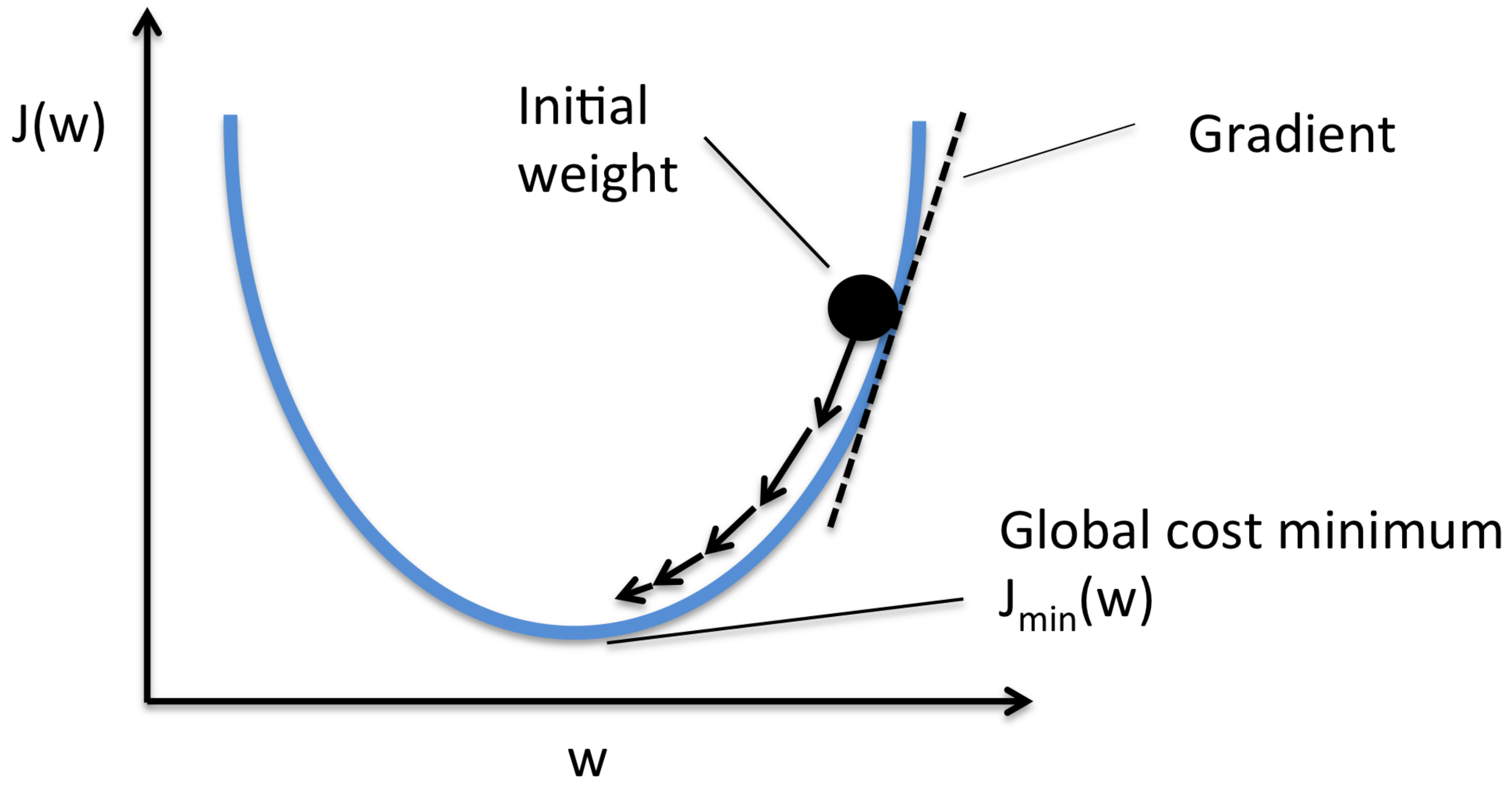
$$\begin{bmatrix} 9 & 0 & 1 \\ 0 & 11 & 1 \\ 1 & 1 & 4 \\ 1 & 0 & 1 \end{bmatrix} \xRightarrow{\text{Transpose}} \begin{bmatrix} 9 & 0 & 1 & 1 \\ 0 & 11 & 1 & 0 \\ 1 & 1 & 4 & 1 \end{bmatrix}$$

$$Y = X\beta$$

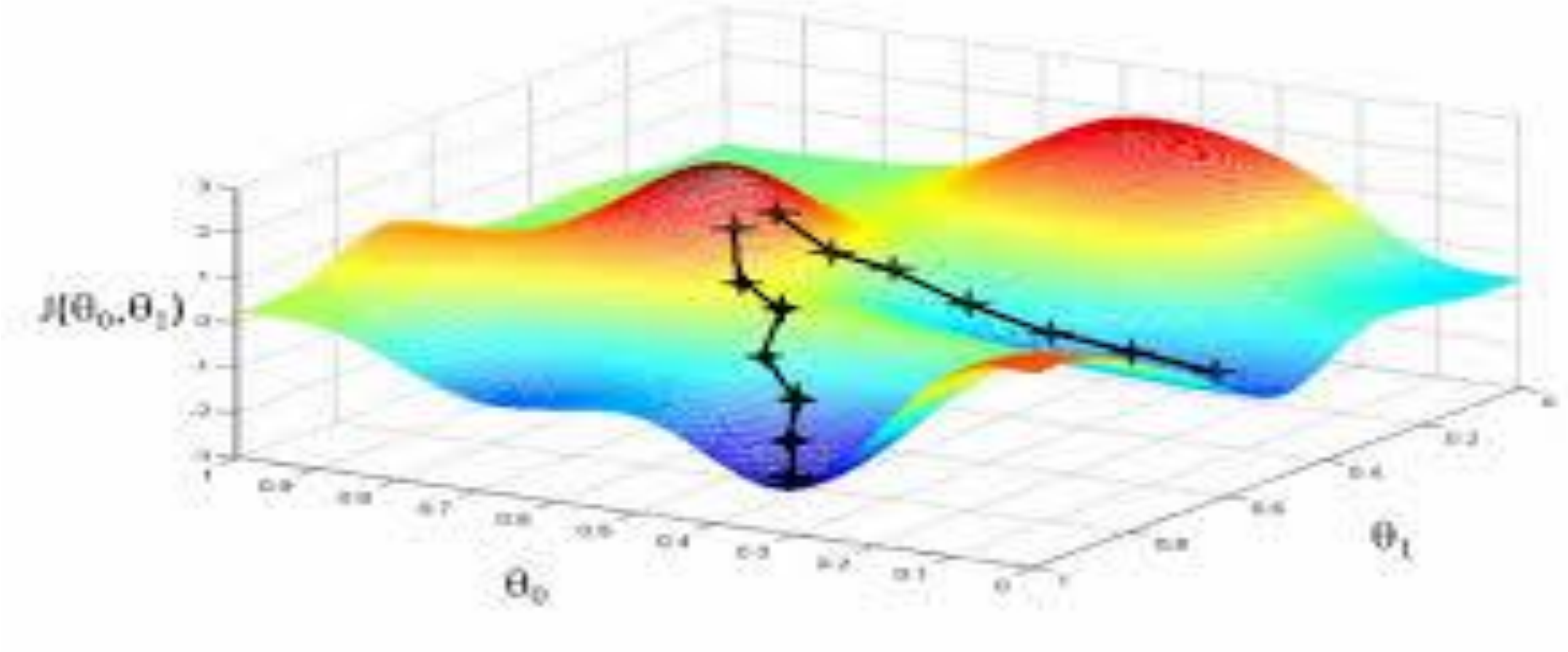
$$\beta = \left(X^T X\right)^{-1} X^T Y$$

```
from numpy.linalg import inv
from numpy import dot, transpose
X = [[1, 6, 2], [1, 8, 1], [1, 10, 0], [1, 14, 2], [1, 18, 0]]
y = [[7], [9], [13], [17.5], [18]]
print dot(inv(dot(transpose(X), X)), dot(transpose(X), y))
```

Gradient Descent – One Variable



Gradient Descent – Two Variable



STATISTICS – a Bird's Eye View

Data Types

- Nominal or Categorical
- Boolean/Binary (Y/N,M/F,B/M)
- Ordinal data
- Interval data
- Ratio data

The Subject Matter of Statistics

- Statistics
  - Parametric Statistics
    - Descriptive Statistics
      - Measure of Central Tendency
      - Measure of Dispersion
      - Measure of Association
    - Inferential Statistics
      - Perform Descriptive Statistics on Sample and extrapolate into Population
  - Non Parametric Statistics
    - Does not assume any distribution

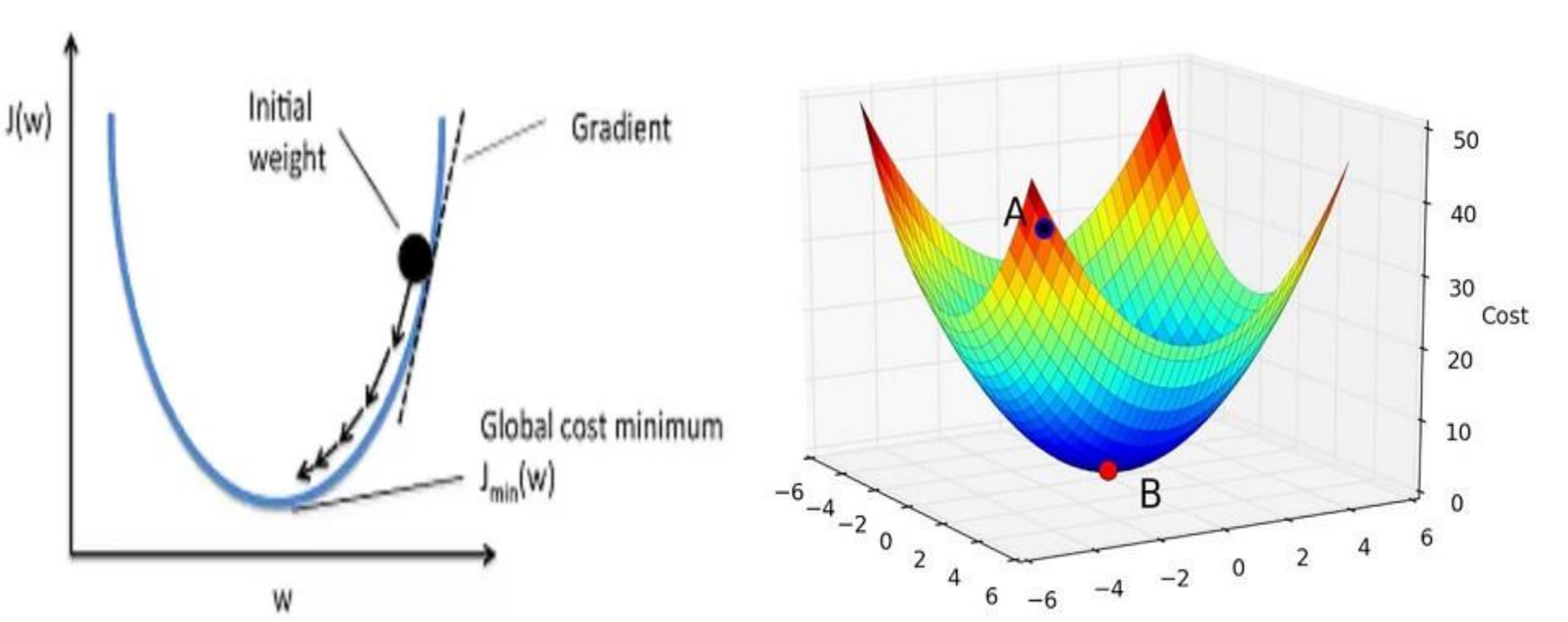
Pragmatics

- Anecdotal vs Statistical reasoning
- Exactitude and Small data syndrome
- Correlation vs Causation
- Randomized Algorithms
- Drug Testing
- Sally Clark Case and Misuse of Statistics
- Oil Spill impact
- Induction vs Deduction

Ohh…….Calculus!

- How Much Calculus One should Know?

Machine Learning as Optimization



What Calculus You Should Know?!

$$
\begin{aligned}
&\frac{\partial}{\partial\hat{\alpha}}\left(\mathrm{SSE}\left(\hat{\alpha},\hat{\beta}\right)\right) = -2\sum_{i=1}^{n}\left(y_i - \hat{\alpha} - \hat{\beta}x_i\right) = 0 \\
\Rightarrow\ &\sum_{i=1}^{n}\left(y_i - \hat{\alpha} - \hat{\beta}x_i\right) = 0 \\
\Rightarrow\ &\sum_{i=1}^{n} y_i = \sum_{i=1}^{n}\hat{\alpha} + \hat{\beta}\sum_{i=1}^{n} x_i \\
\Rightarrow\ &\sum_{i=1}^{n} y_i = n\hat{\alpha} + \hat{\beta}\sum_{i=1}^{n} x_i \\
\Rightarrow\ &\frac{1}{n}\sum_{i=1}^{n} y_i = \hat{\alpha} + \frac{1}{n}\hat{\beta}\sum_{i=1}^{n} x_i \\
\Rightarrow\ &\bar{y} = \hat{\alpha} + \hat{\beta}\bar{x}
\end{aligned}
$$

Tools and Technologies (Open Source)

- Weka WorkBench and Weka Java Library
- SciKit Learn/ TensorFlow
- NLTK/OpenNLP
- NumPy/SciPy
- GNU R and R Studio
- OpenCV
- Apache Mahout/Mlib

Tools and Technologies (Proprietory)

- Offerings from Cloud Vendors (Google,AWS,Microsoft)
- IBM Watson Studio and SPSS
- SAS Miner
- Tableau

Limits! – Everything which has got Scope has its Limits

Machine Learning “anomalies”

- Algorithmic Intractability with Turing Machine/Lambda/Predicate Logic
- Linearity assumption with Hilbert space method
- Inductive errors inherent in statistical methods

Algorithmic Intractibility – A simple example

- The notion of NP-Hard and NP-Complete
- (26 factorial divided by 1 million)/(3600*24*365)/10000000

Problems with Hilbert space method

- It assumes that Variables independently act on the output
- The above assumption does not hold in most real life situations
- The variables are inter-dependent
- Some Anecdotes explaining this
    - Combination of people give different results

Statistical methods - anomalies

- The problem of induction
- Changes in the environment
- Effect of Exogeneous data
- Lack of pattern in data
    - Meta Trader Example

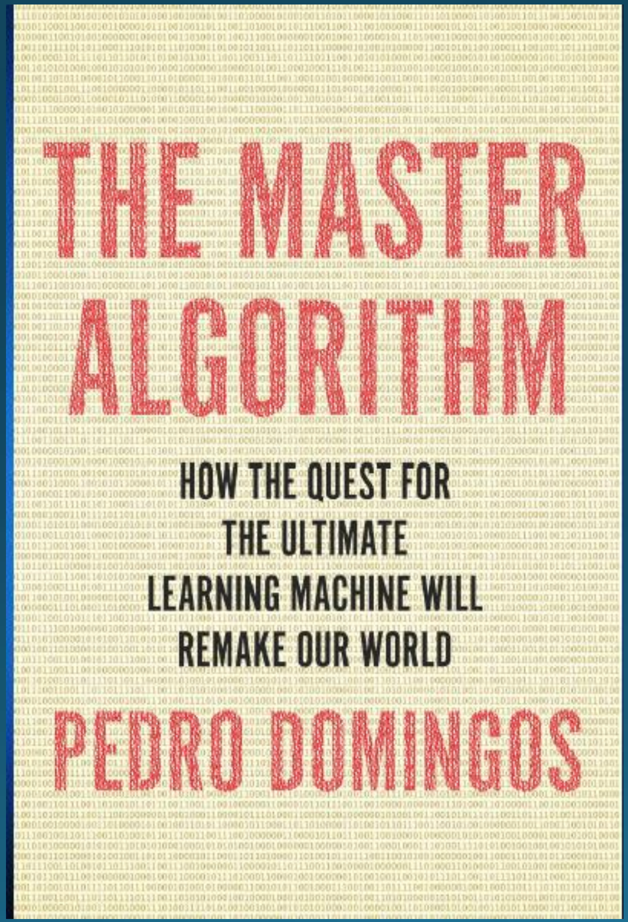
Who does ML in a viable manner?

- The Biggies , their ridiculous budget and the resources
- Why they can afford it?
- “We all can drive a rolls Royce, but Ambani’s son can own and operate it”

Will more data helps us make NP problems tractable?

- No...."I have discovered a wonderful proof, where this session is too short for the explanation"
- More data won't produce better result (Central Limit theorem)

ML – A Remarkable Book!



Five Tribes of Machine Learning and their concerns

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

Tribe	Problem	Solution
Symbolists	Knowledge composition	Inverse deduction
Connectionists	Credit assignment	Backpropagation
Evolutionaries	Structure discovery	Genetic programming
Bayesians	Uncertainty	Probabilistic inference
Analogizers	Similarity	Kernel machines

Topology of the Master Algorithm

- Representation
  - Probabilistic Logic (Eg:- Markov Logic Networks)
  - Weighted Formulas (Eg:- Distribution over States)
- Evaluation
  - Posterior Probability
  - User Defined Objective Function
- Optimization
  - Formula Discovery (Genetic Algorithm)
  - Weight Learning (BackPropogation Algorithm)

Operational Realm

- Machine Learning is data intensive (More Data)
- Machine Learning Projects are people Intensive
- Machine Learning Projects are hardware resource intensive
- There are compliance and governance issues around data

Engagement Models

- Technology Partner Approach (in Business driven Engagements)
- Upselling though the Accounts
- Cross selling works in certain accounts
- Exploring “Green Field Projects” do pay off
- Technology driven Engagements (Pressure from the Top)

Q&A (Part 4)