

Introduction to Machine Learning

Sourav Nandi

[ABOUT.ME/SOURAV.NANDI](https://about.me/sourav.nandi)

Symphony AI, IIT Kanpur

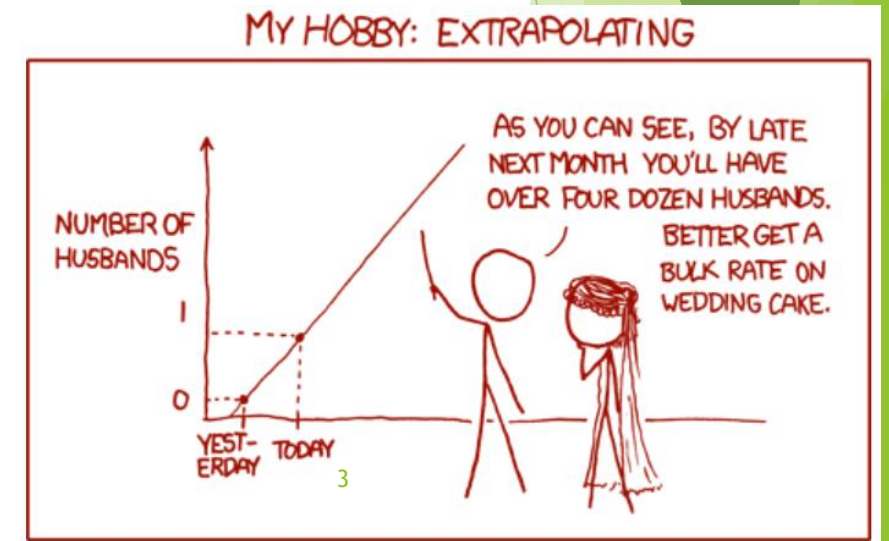
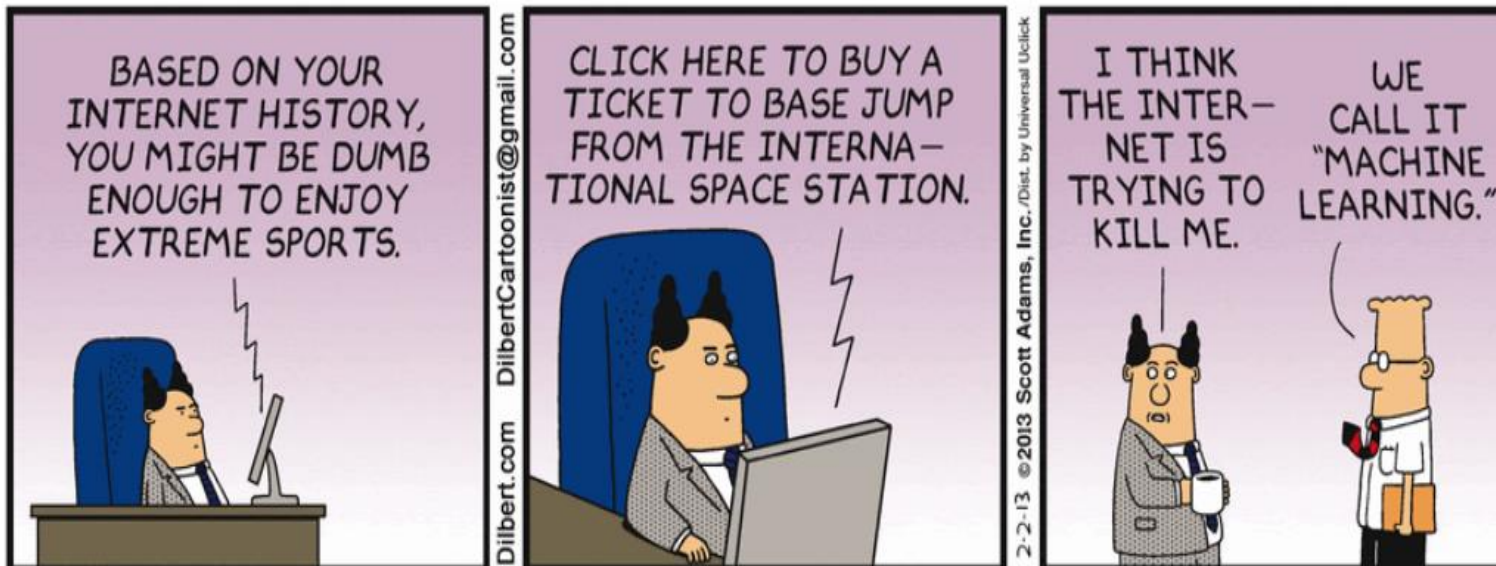
Social Handles- @souravstat

How much Data is Created Every Day?

- ▶ ‘Google’ has become a Verb! (**3.5 billion** search queries every day)
- ▶ **2.5 quintillion bytes** of data are produced by us every day. (18 Zeroes!)
- ▶ ~90% of the World’s Data created in last 2 years – Accelerating Pace
- ▶ **Every Day-**
 - ~250 billion emails are sent (**45% are Spam-** Hit the Unsubscribe button!)
 - 100+ million photos and videos are shared on Instagram
 - ~500 million Tweets are made (~45% of Covid-19 tweets estd to be **Bot-Generated!**)
- ▶ By the end of 2020, ~**31 Billion IoT devices**. The estimated size of the entire digital universe will be a whopping **44 zettabytes** (21 Zeroes in a ZB!)
- ▶ **What to do with all these Data?**

Machine Learning- The Art and Science of Learning from Data

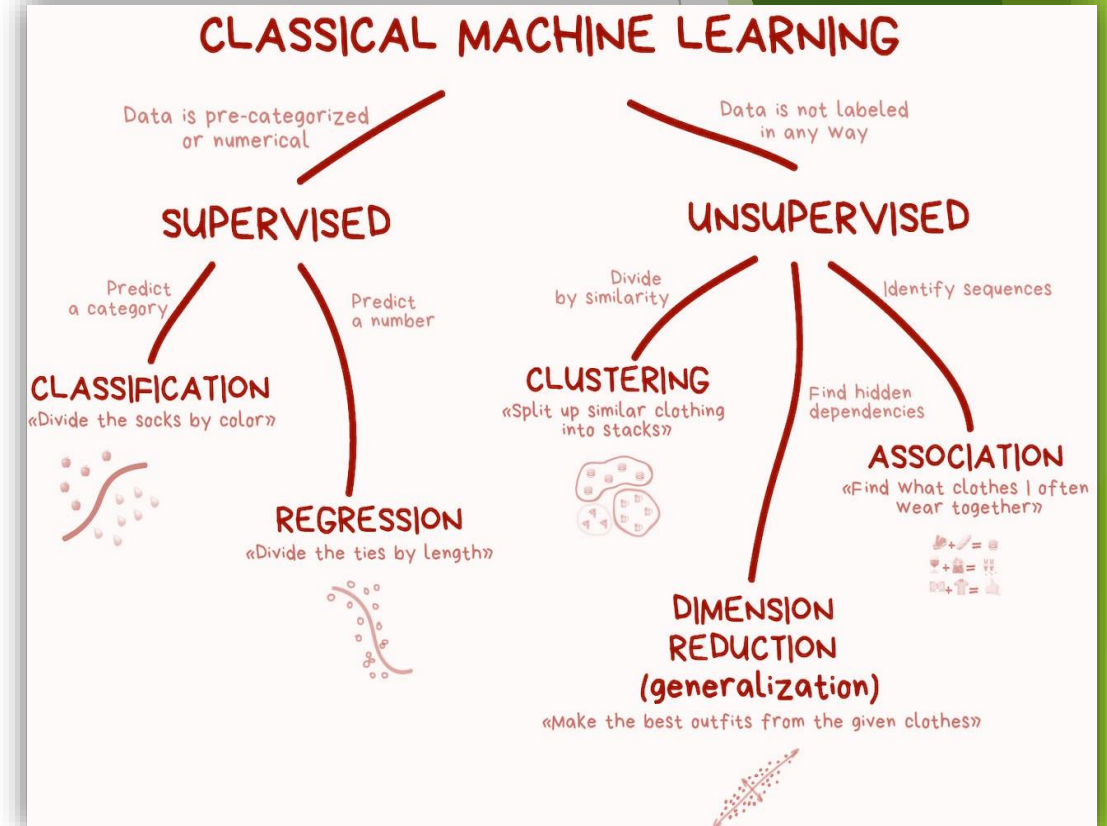
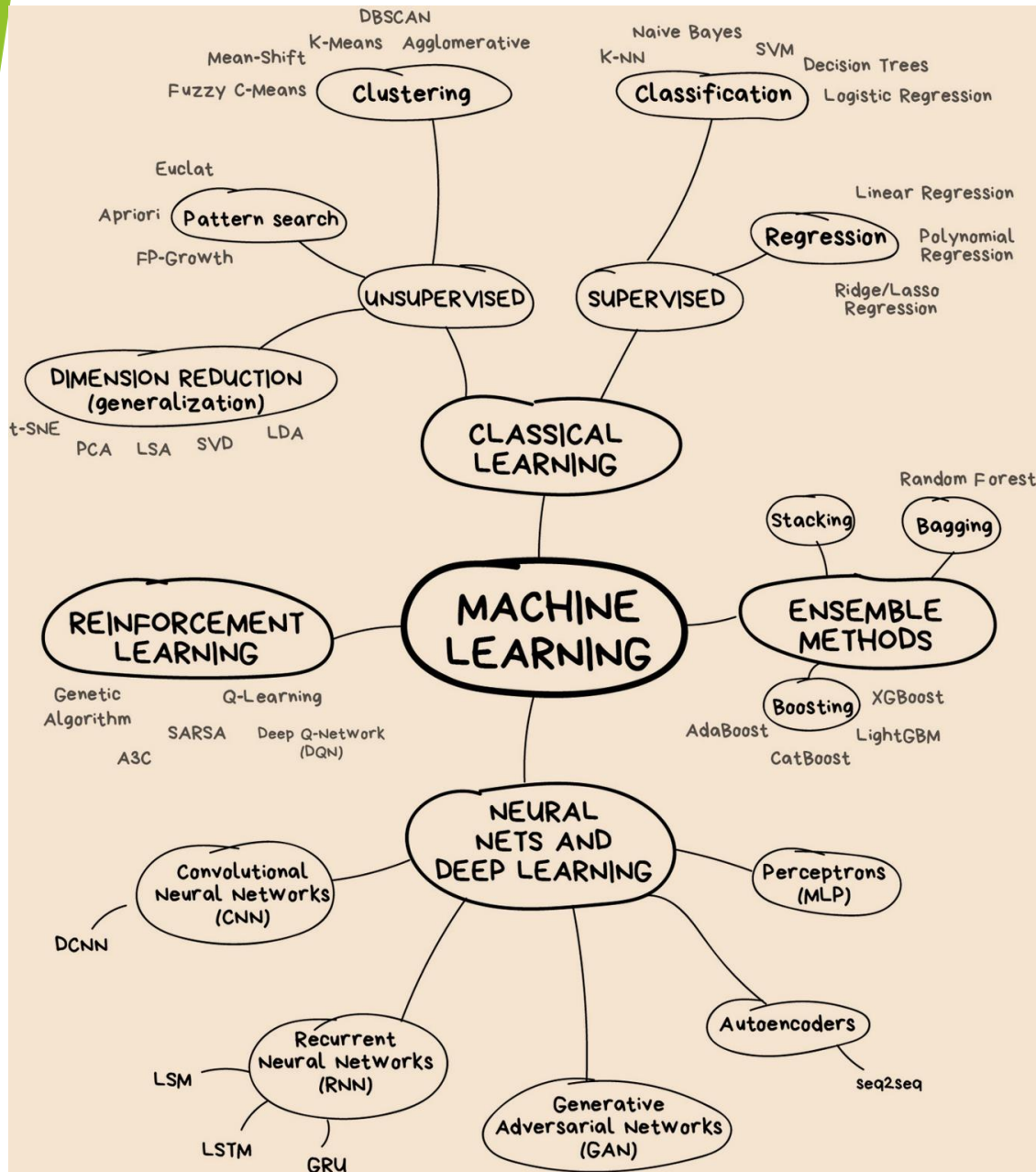
- ▶ *We are drowning in Information and starving for Knowledge* — **John Naisbitt** (Author of 'Megatrends')
- ▶ Is Learning Possible?
- ▶ **Generalization**/ Pattern Recognition (Easy) vs **Extrapolation**/ Finding Higher Dimensional Insights (Hard)



Timeline of Machine Learning (Wikipedia)

Decade ↕	Summary ↕
<1950s	Statistical methods are discovered and refined.
1950s	Pioneering machine learning research is conducted using simple algorithms.
1960s	Bayesian methods are introduced for probabilistic inference in machine learning. ^[1]
1970s	' AI Winter ' caused by pessimism about machine learning effectiveness.
1980s	Rediscovery of backpropagation causes a resurgence in machine learning research.
1990s	Work on Machine learning shifts from a knowledge-driven approach to a data-driven approach. Scientists begin creating programs for computers to analyze large amounts of data and draw conclusions – or "learn" – from the results. ^[2] Support vector machines (SVMs) and ^[3] recurrent neural networks (RNNs) become popular. The fields of ^[4] computational complexity via neural networks and super-Turing computation started.
2000s	Support Vector Clustering ^[5] and other Kernel methods ^[6] and unsupervised machine learning methods become widespread. ^[7]
2010s	Deep learning becomes feasible, which leads to machine learning becoming integral to many widely used software services and applications.

The ML Family Tree



Classification- Split into Categories

► **Usage:** Fraud Detection (Online Transaction), Spam Filtering (Email), Sentiment Analysis (+ve/-ve/neutral), Handwriting Recognition (MNIST) etc.

► **Overview of Popular Algorithms**

➤ **Logistic Regression** (GLM* with Logit link), Multinomial Logit

➤ **Decision Tree**, Random Forest, Bagging, Boosting

➤ **Naïve Bayes** (Conditional Independence b/w Features, Given a Category)

$$p(C_k | x_1, \dots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^n p(x_i | C_k)$$

where the evidence $Z = p(\mathbf{x}) = \sum_k p(C_k) p(\mathbf{x} | C_k)$

$$\mathbf{X}\beta = \ln\left(\frac{\mu}{1 - \mu}\right)$$



* Allows Response (Dependent) Variables to have error distribution models other than a normal distribution

Classification (Contd.)

- ▶ **KNN (K-Nearest Neighbors) Algorithm**
(Idea: Find Closest K Neighbors)
- ▶ Distance Measures: Euclidean distance, Mahalanobis distance, Manhattan distance, Cosine Distance etc
- ▶ **Support Vector Machine and Kernel Trick**

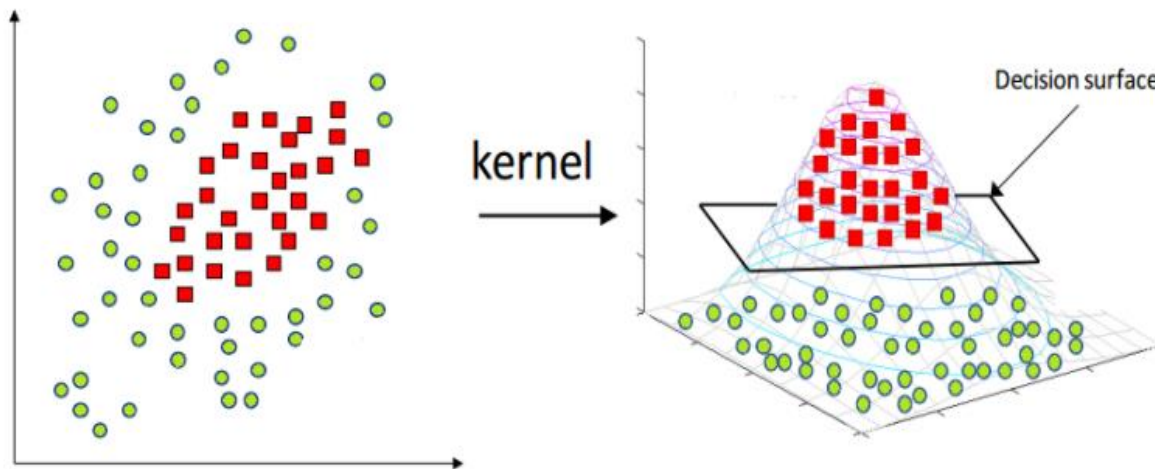
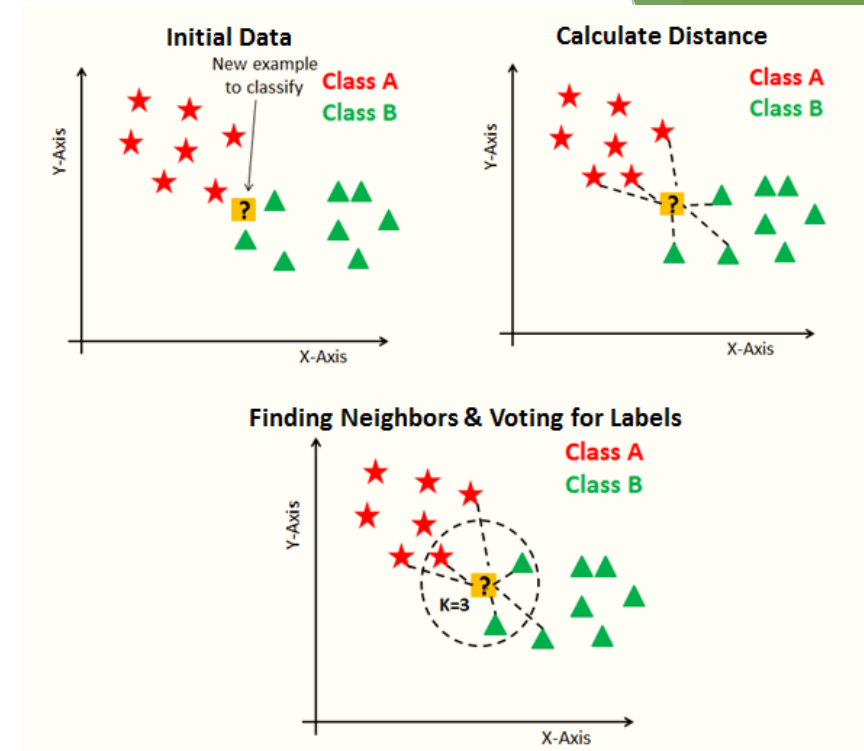
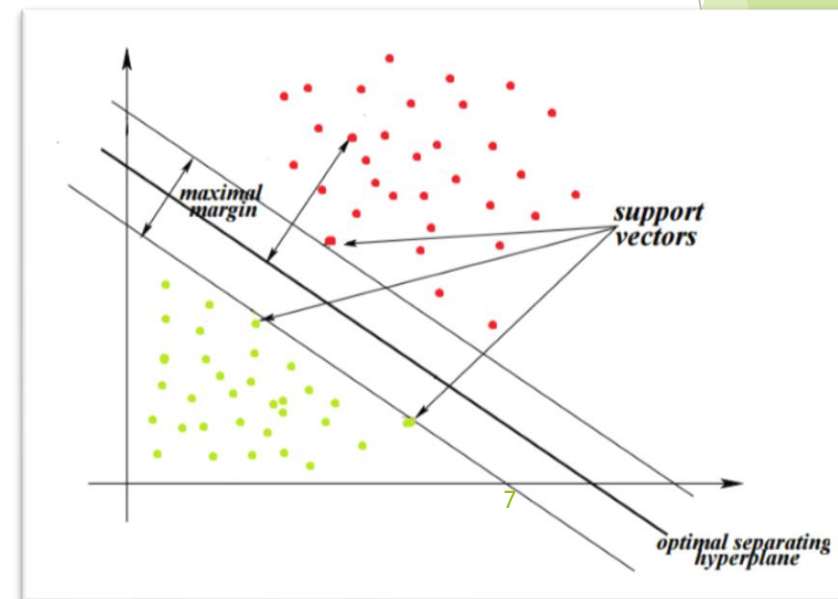


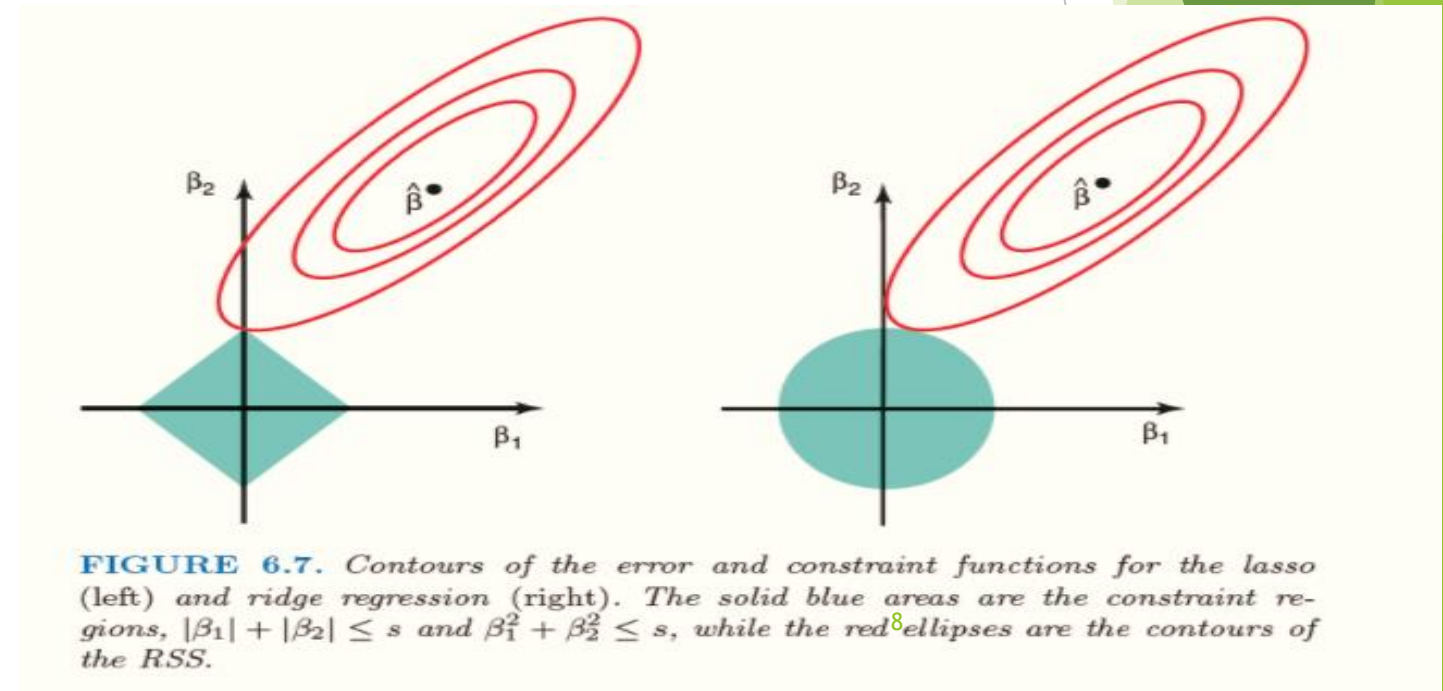
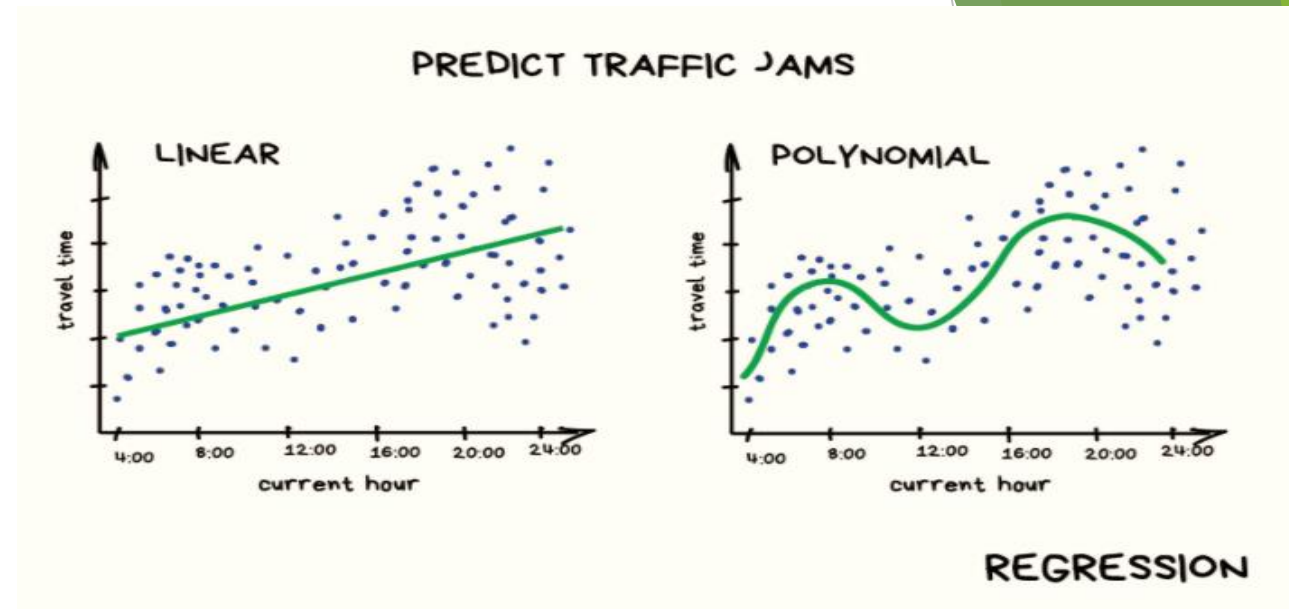
Image Credit: Towards data science



Regression

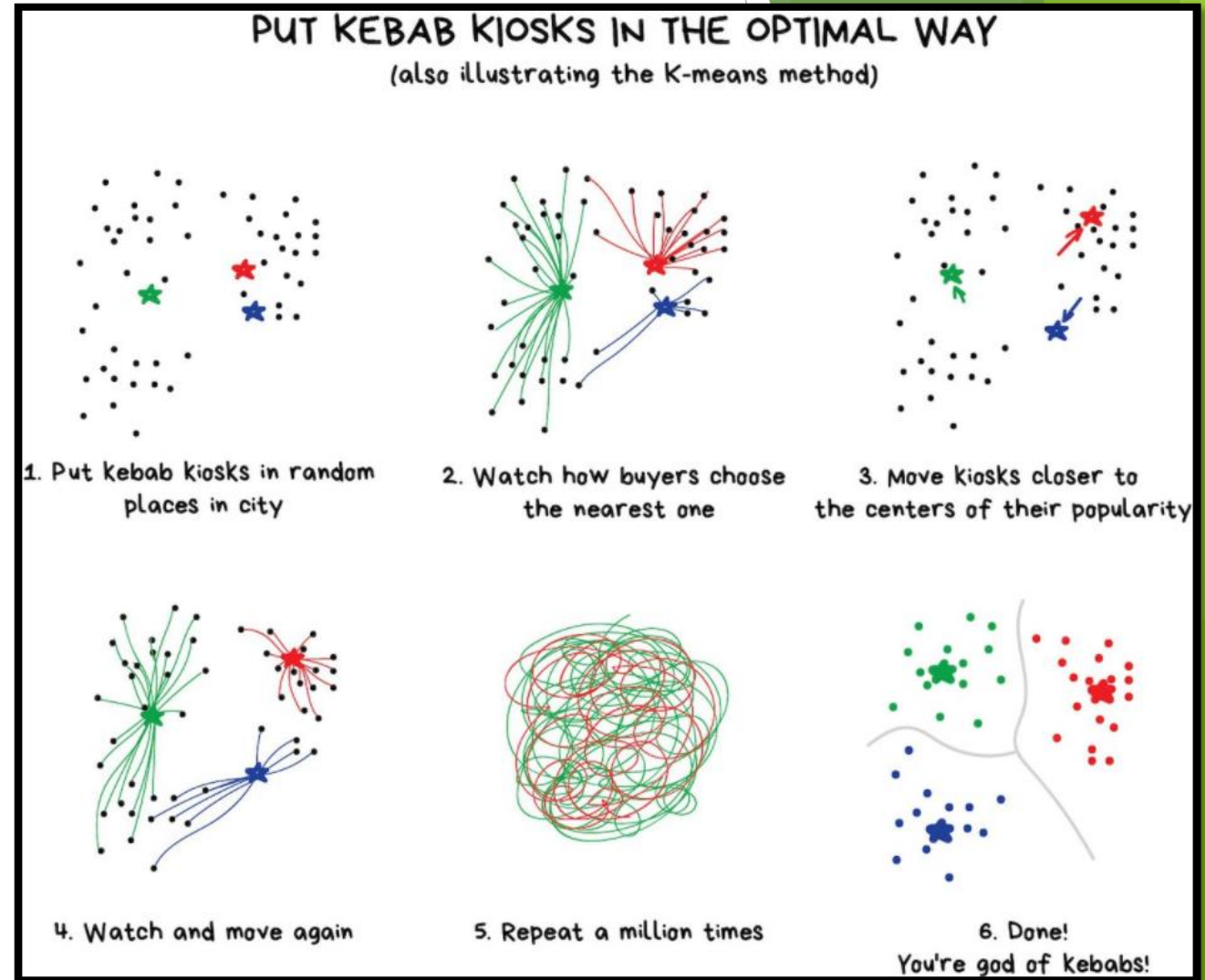
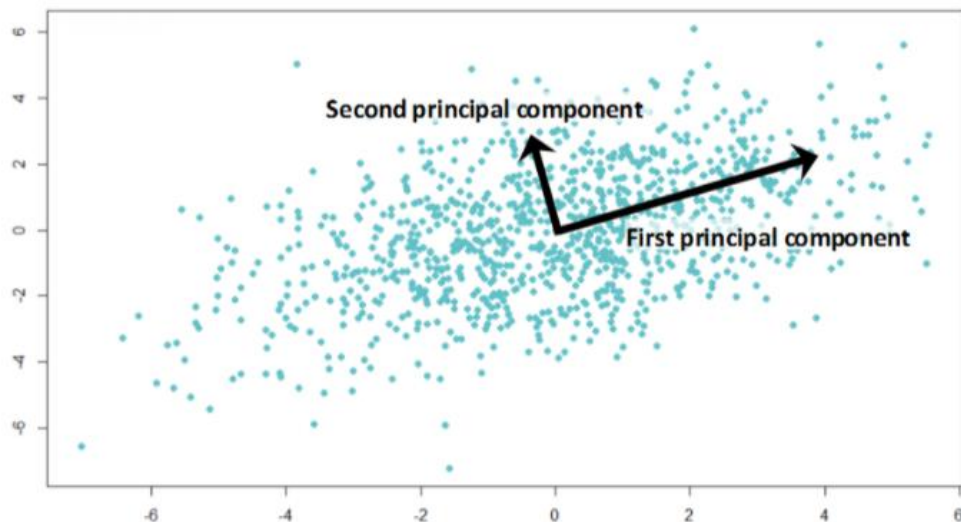
- ▶ Multiple **Linear** Regression
(Discussed in earlier classes in detail)
- ▶ **Ordinary Least Square** Method:
Computes the unique line (or hyperplane) that minimizes the sum of squared distances (usually vertical) between the true data and that line
- ▶ **Ridge Regression** and **LASSO**
(reducing model complexity to prevent overfitting, Variable Selection)

Image Credit: ISLR book (Ref 3)



Unsupervised Learning

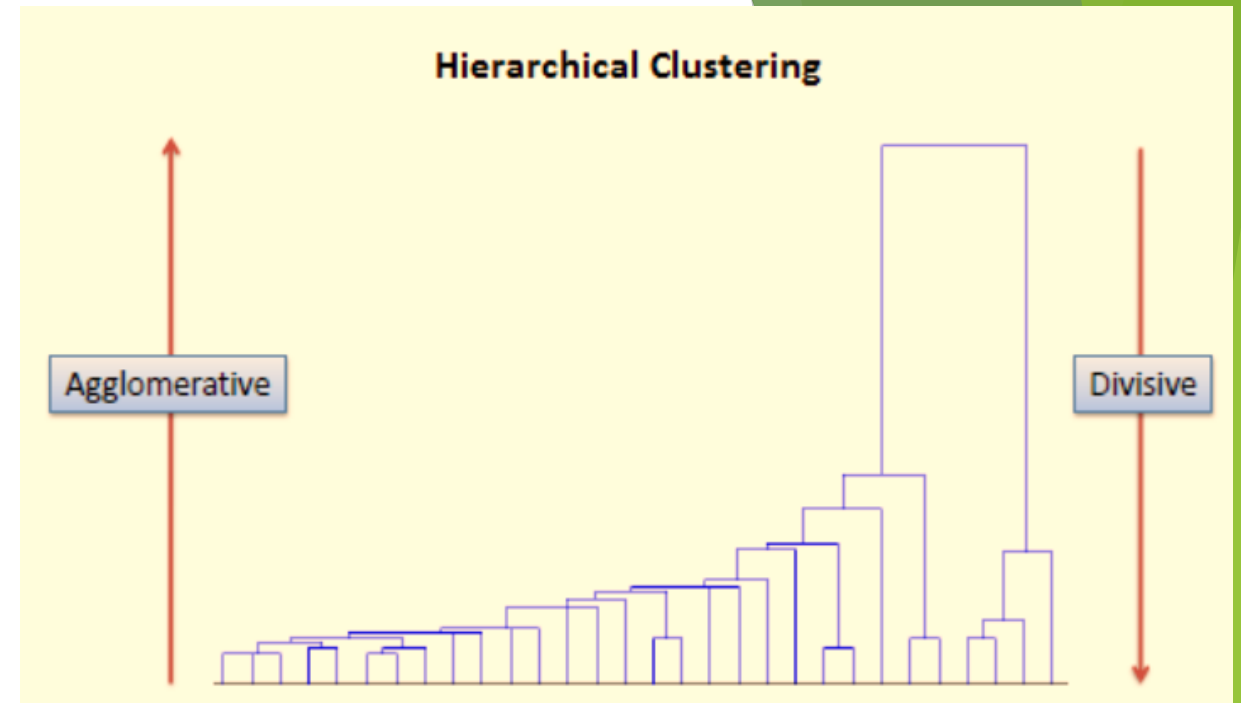
- ▶ Market Segmentation (**Clustering**), Anomaly Detection, Image Compression (**Dimensionality Reduction**) etc
- ▶ **K-means Clustering**
- ▶ **Principal Component Analysis** (Projection into Lower Dimensional Space, Summarizing Information)



Unsupervised Learning

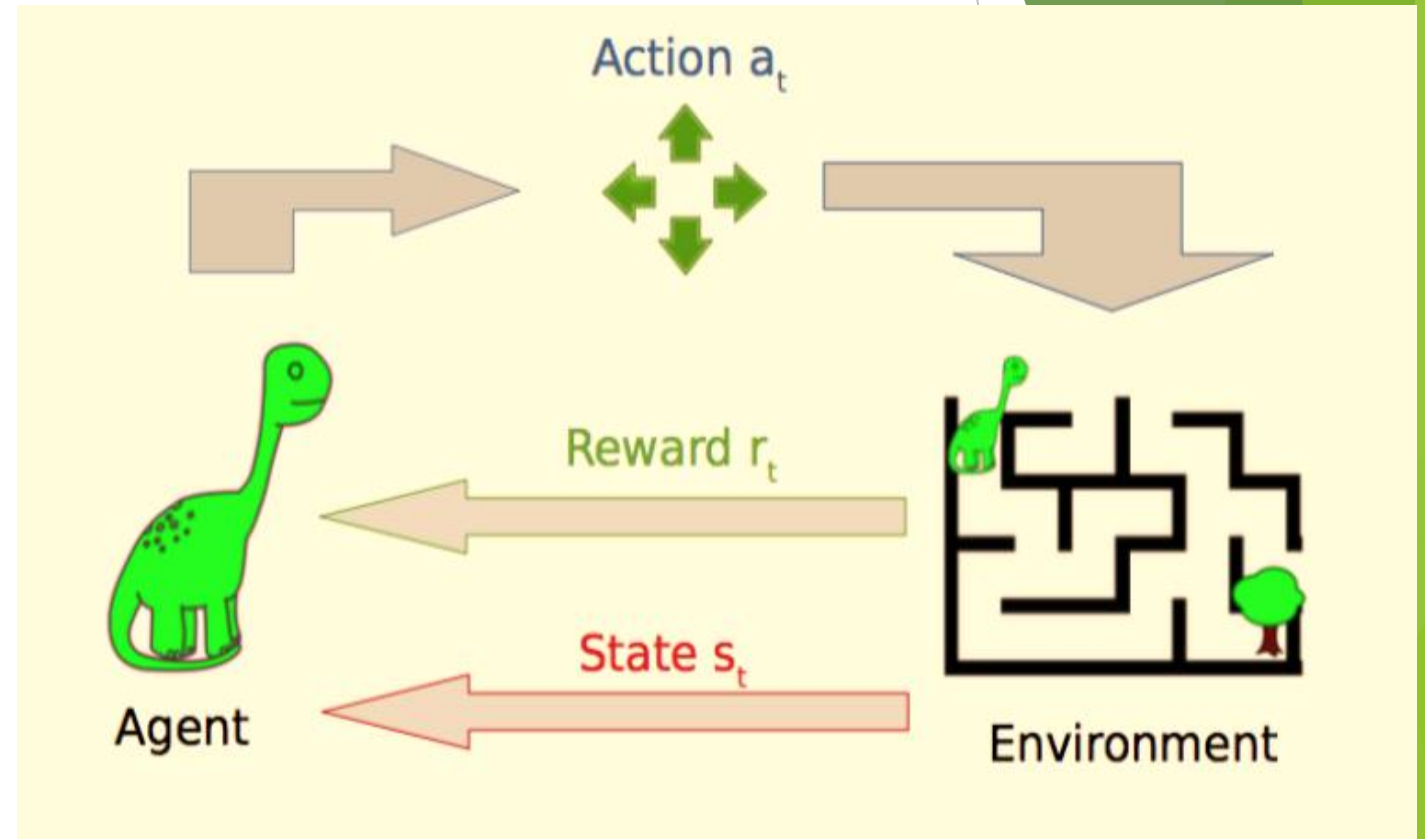
- ▶ **Hierarchical Clustering**
- ▶ Distance between Clusters
 - Single Linkage (Min)
 - Complete Linkage (Max)
 - Average Linkage (All Pairs)
 - Centroid Linkage
- ▶ **Association Rule Mining** (Looking for patterns, eg, analyzing Shopping behavior, Marketing Strategy)

Image Credit: saedsayad.com



Reinforcement Learning

- ▶ Model is trained by having an Agent interact with environment
- ▶ **Desired Action gets Rewarded**
- ▶ “Good Behaviors are Reinforced”
- ▶ One of **Three fundamental ML Paradigms** (along with Supervised learning and Unsupervised learning)
- ▶ When in an active Environment, like Video Games (Super Mario!), **Self Driving Car** etc
- ▶ Goal is to Minimize Error (maybe difficult to predict all possible moves)



Credit: TWIML Online

Diving Deeper into some Core Ideas

Correlation does not Imply Causation

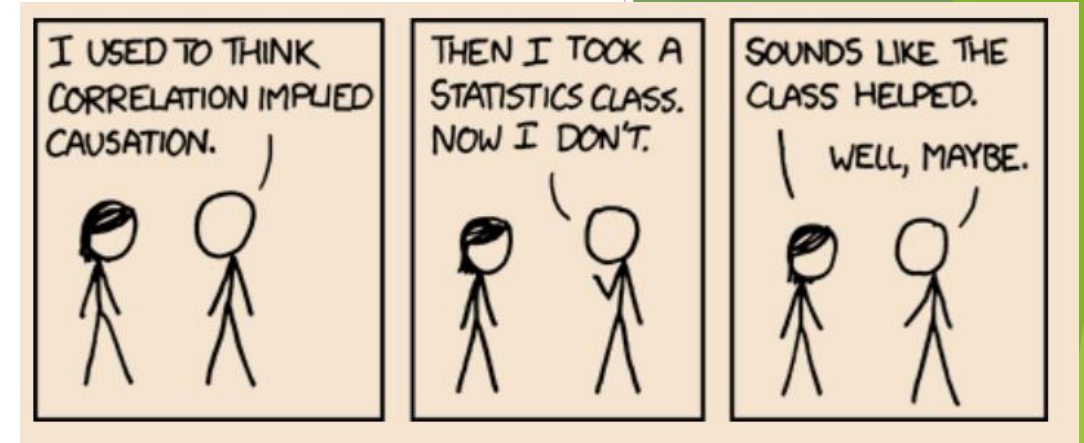
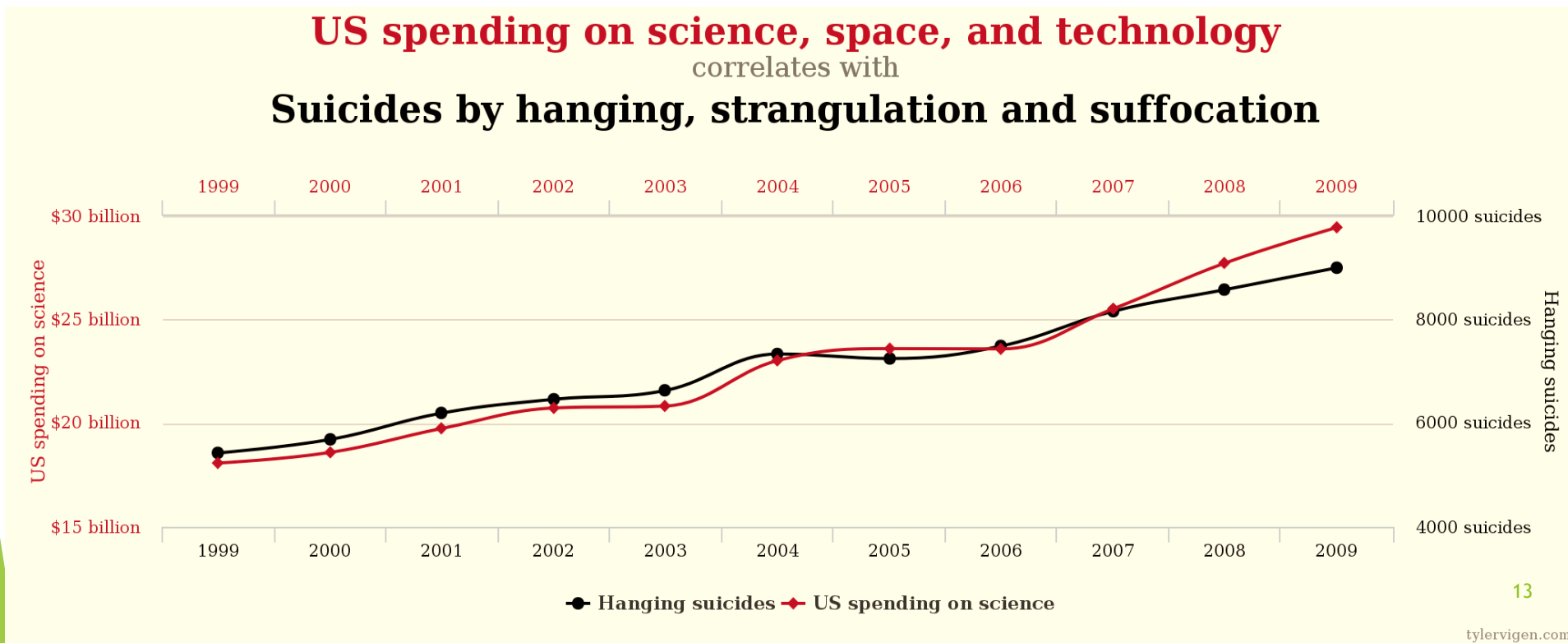


Image Credit: xkcd



Bias-Variance Tradeoff: Underfitting vs Overfitting

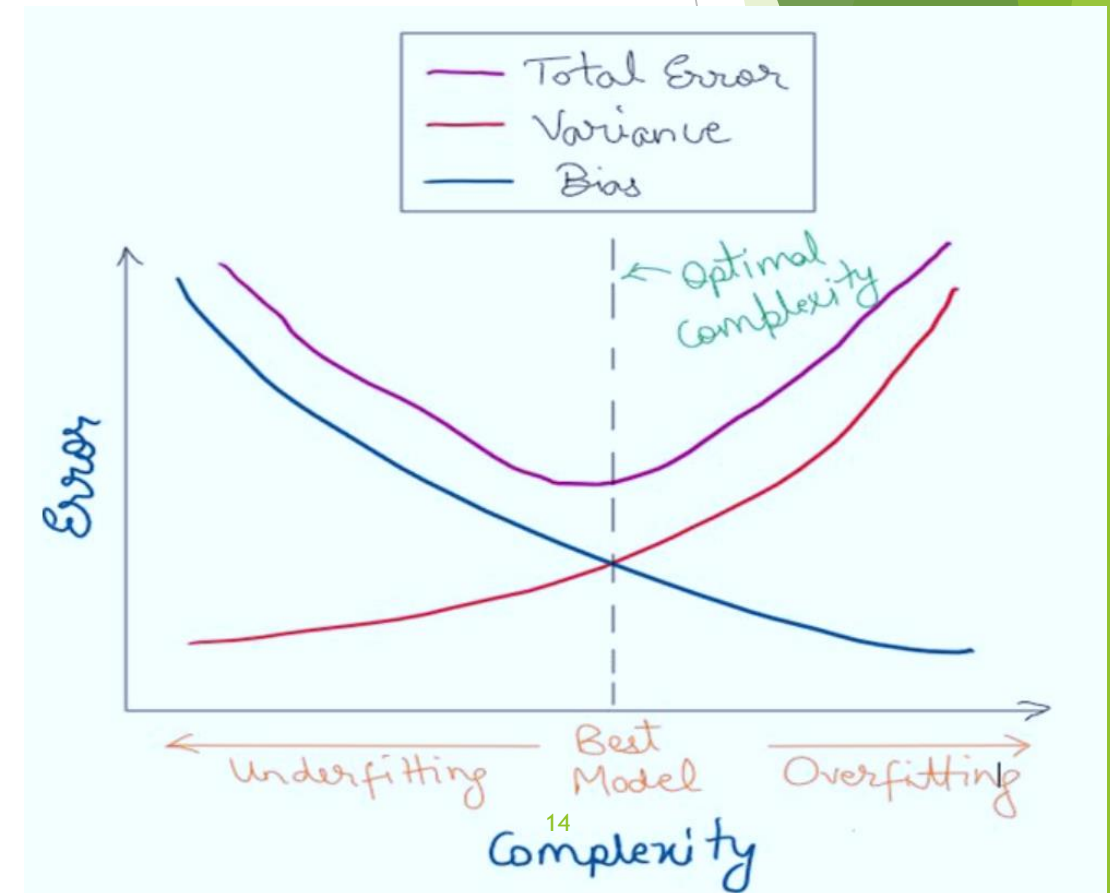
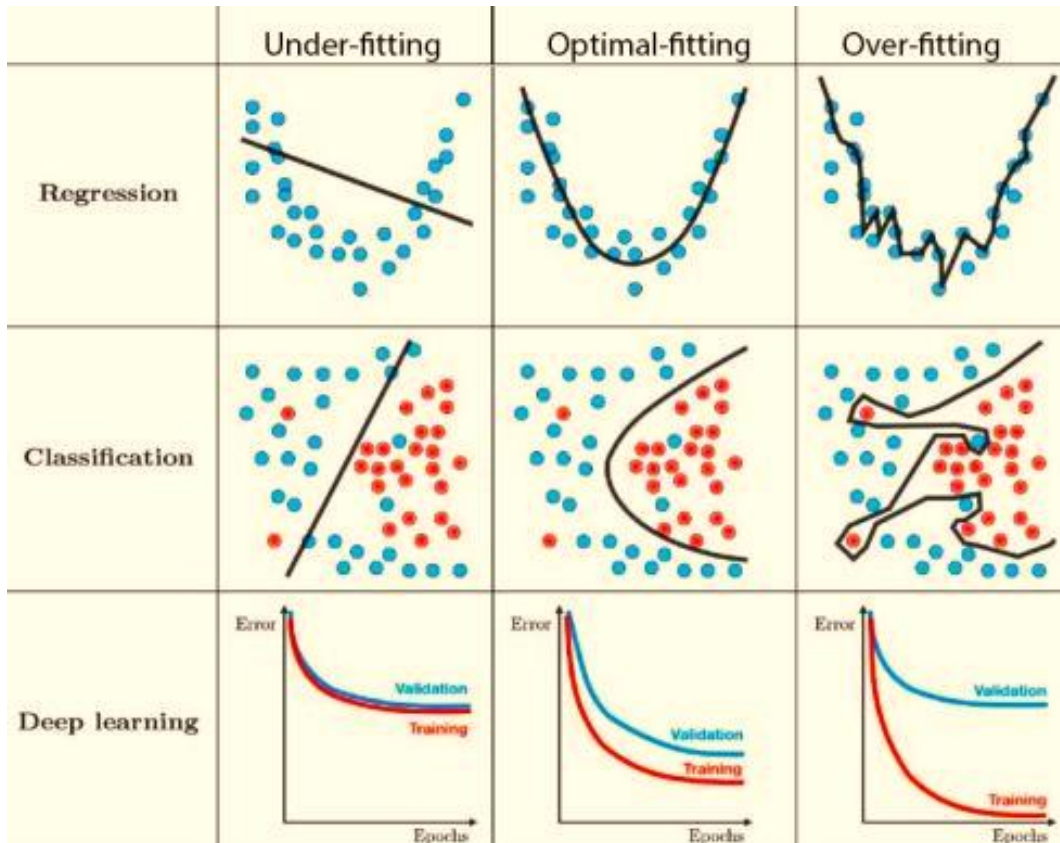
$$\mathbb{E}_D \left[(y - \hat{f}(x; D))^2 \right] = \left(\text{Bias}_D [\hat{f}(x; D)] \right)^2 + \text{Var}_D [\hat{f}(x; D)] + \sigma^2$$

where

$$\text{Bias}_D [\hat{f}(x; D)] = \mathbb{E}_D [\hat{f}(x; D)] - f(x)$$

and

$$\text{Var}_D [\hat{f}(x; D)] = \mathbb{E}_D [\hat{f}(x; D)^2] - \mathbb{E}_D [\hat{f}(x; D)]^2.$$



Curse of Dimensionality

- Problems in High Dimension due to Data Sparsity
- Adding each new dimension (ie, adding a feature) increases the data set requirement exponentially
- Separation of Wind Turbines- 2D vs 3D view

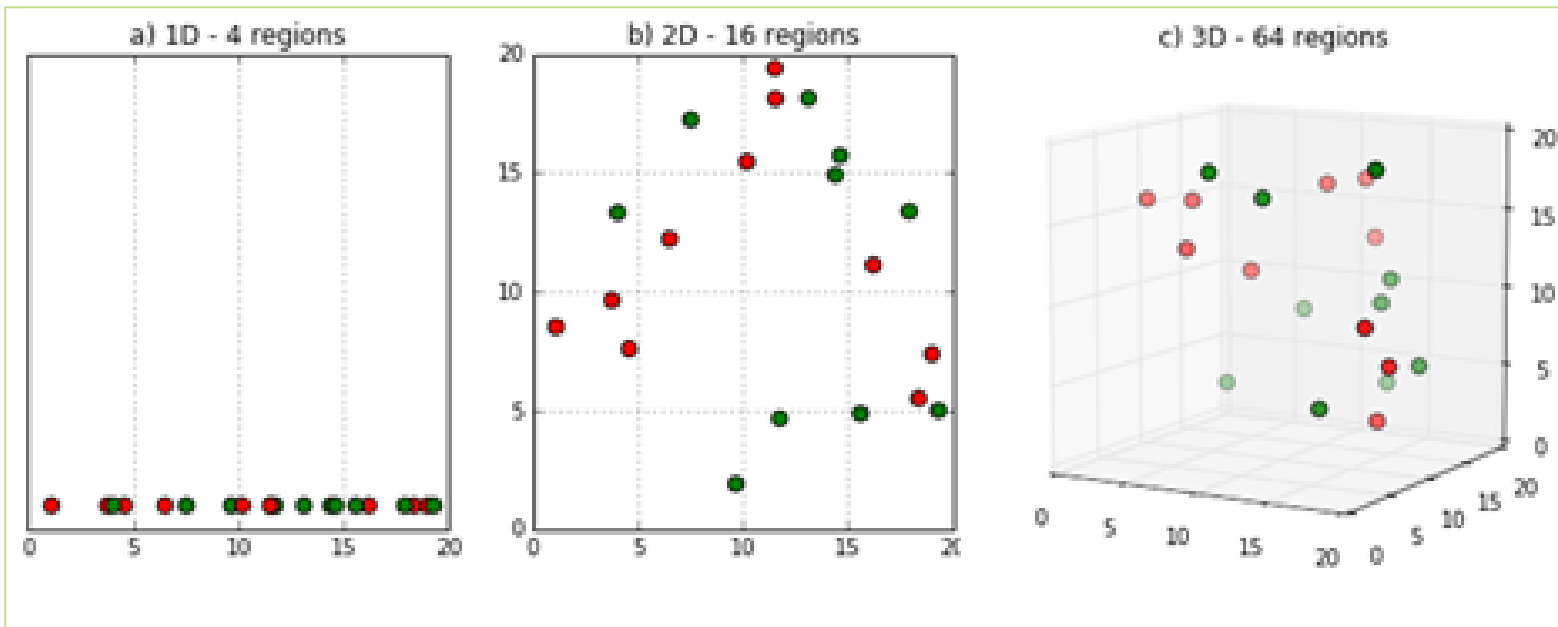





Image Credit: deepai.org

	TYPE	NAME	DESCRIPTION	ADVANTAGES	DISADVANTAGES
Linear		Linear regression	The “best fit” line through all data points. Predictions are numerical.	Easy to understand -- you clearly see what the biggest drivers of the model are.	<ul style="list-style-type: none"> X Sometimes too simple to capture complex relationships between variables. X Tendency for the model to “overfit”.
		Logistic regression	The adaptation of linear regression to problems of classification (e.g., yes/no questions, groups, etc.)	Also easy to understand.	<ul style="list-style-type: none"> X Sometimes too simple to capture complex relationships between variables. X Tendency for the model to “overfit”.
Tree-based		Decision tree	A graph that uses a branching method to match all possible outcomes of a decision.	Easy to understand and implement.	<ul style="list-style-type: none"> X Not often used on its own for prediction because it's also often too simple and not powerful enough for complex data.
		Random Forest	Takes the average of many decision trees, each of which is made with a sample of the data. Each tree is weaker than a full decision tree, but by combining them we get better overall performance.	A sort of “wisdom of the crowd”. Tends to result in very high quality models. Fast to train.	<ul style="list-style-type: none"> X Can be slow to output predictions relative to other algorithms. X Not easy to understand predictions.
		Gradient Boosting	Uses even weaker decision trees, that are increasingly focused on “hard” examples.	High-performing.	<ul style="list-style-type: none"> X A small change in the feature set or training set can create radical changes in the model. X Not easy to understand predictions.
Neural networks		Neural networks	Mimics the behavior of the brain. Neural networks are interconnected neurons that pass messages to each other. Deep learning uses several layers of neural networks put one after the other.	Can handle extremely complex tasks - no other algorithm comes close in image recognition.	<ul style="list-style-type: none"> X Very, very slow to train, because they have so many layers. Require a lot of power. X Almost impossible to understand predictions.

Comparison of some Popular Algorithms

Table Credit: dataiku.com

Things to keep in mind

► **Splitting the Dataset:**

- Training Set (Data Sample used to Fit the Model, to get the Parameters)
- Validation Set (Tuning Hyperparameters to choose final model)
- Test Set (To evaluate the final model, should not be used for training)

► **Other Common Pitfalls in ML (Violation of Assumptions):**

- Non-Linearity (Plotting Residuals against Fitted Values, Non-linear Transformation)
- High Leverage Points (Cook's Distance Plot)
- Correlation of Error Terms (eg, Time Series data)- Controlled Experiment
- Heteroscedasticity (Non-constant Variance of Error Term)
- Multicollinearity (Correlated Predictors, eg. Dummy Variable Trap)

That's All, Friends!

STAT&ML Lab is a non-profit organization to bring young minds into research projects on Statistics & Machine Learning.

We aim to provide training and research projects on Statistics, Data Science, and ML.

The primary goal of this lab is to promote research in Statistics in India and throughout the world



LEARN, EXPLORE AND
INNOVATE

Training Programs

RESEARCH PROJECTS
ON STAT, DS & ML

Research Internships

STAT&ML LAB

 **EXPLORE MORE AT** 
[HTTPS://WWW.CTANUJIT.ORG/STATML-LAB.HTML](https://www.ctanujit.org/statml-lab.html)

Thanks & References

- 1) Special Thanks to All the Participants, BKC College, WBSU and STAT & ML Lab:
<https://www.ctanujit.org/statml-lab.html>
- 2) Image Credit- Wikipedia, Reddit, SlideShare, me.me, Imgflip, xkcd,
- 3) <http://faculty.marshall.usc.edu/gareth-james/ISL/> (ISLR Book)
- 4) <https://developers.google.com/machine-learning/guides/good-data-analysis>
- 5) <https://hackernoon.com/choosing-the-right-machine-learning-algorithm-68126944ce1f>
- 6) https://en.wikipedia.org/wiki/Reinforcement_learning
- 7) Download Latest Version Of this PPT (& other Materials):
<https://github.com/souravstat/>
- 8) **Please feel free to reach out to me anytime for a discussion:**
<https://about.me/sourav.nandi> , souravsijna@gmail.com