**Introduction to Machine Learning**

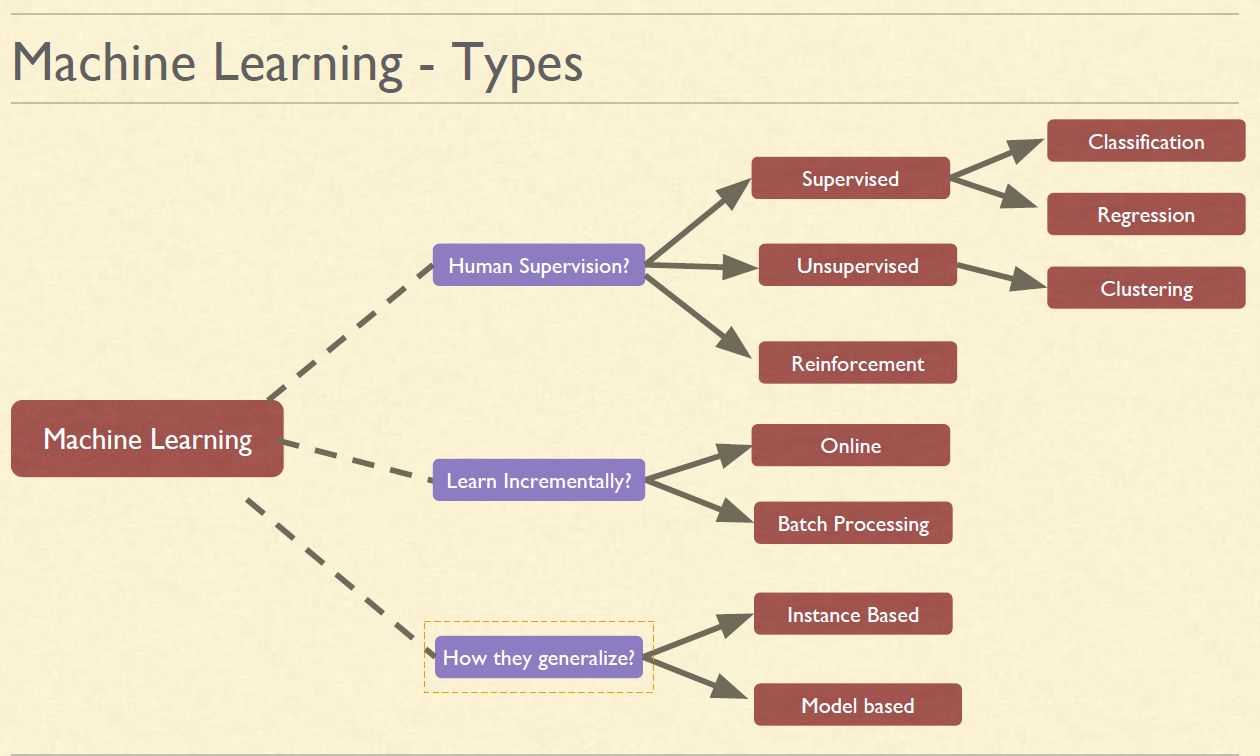
* ***Artificial Intelligence :*** Referring to computers behaving intelligently. Performs tasks as human Intelligence. AI to refer to a technical field which focused on programming computers to make decisions. **Sub Objectives :** Reasoning ,Navigate, NLP, Represent Knowledge, Perception
* ***Machine Learning :*** Referring to making machines (computers) learn certain patterns using Set of Algorithm and then to make predictions using those learnt patterns. ML focuses more on making predictions about the future. Data + Intelligence

***Example :*** Self Driving Cars, Recommendation System,

**Collect Data** (Use of Devices & Connectivity) 🡪 **Cleaning** 🡪 **Process** (Multi Core, GPU, Distributed)

If Data is Huge -> Use Distributed Processor

Math/Graphics -> GPUs



* **Supervised Classification :** Ex : Spam Filter Model
* **Supervised Regression :** Ex : Car prediction Model

***Gradient Descent :*** Instead of trying all lines, only try those lines which are useful. Uses Loss Function. Feature Scaling is required.

***Types :*** 1. **Batch GD** 2. **Mini Batch GD** 3. **Stochastic GD**

***Types of variable :***

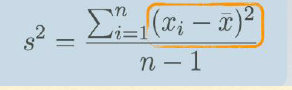
* Numerical : Continuous & Discrete
* Categorical : Regular & Ordinal(Review given by Customer)

***Box Plot :*** 25% down,25%Up,50% in box, line in middle is Median

***Measure of central Tendencies :***

* **Mean** : Average
* **Mode** : Most Frequent
* **Median** : Pick the middle number (if Odd), Sum the Middle two no & divide by 2 (If Even). **It is mostly preferred**.

***Variance :*** How well data is spread from center Point. Subtracts all no. from Median & Square it. Add all. divide by N-1. To remove the Negative, we are doing Square.



***Normal Distribution :***Curve plotted based on Mean(Miu) & Standard Deviation(Sigma) (Sqrt of Variance)

More Standard Deviation , Plot will spread Out

Area of Curve = 1

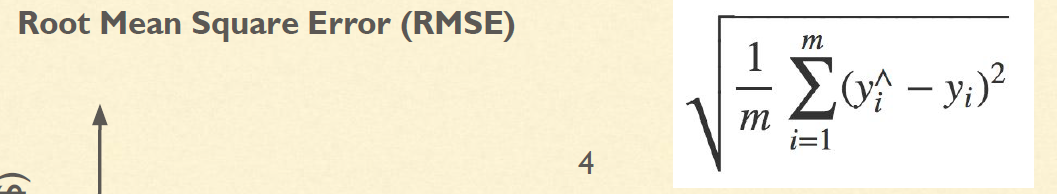
68% of Data 🡪 Between Mean-SD & Mean+SD

95% of Data 🡪 Between Mean-2SD & Mean+2SD

99.7% of Data 🡪 Between Mean-3SD & Mean+3SD







***Overfitting :*** The model performs well on the training data but does not generalize well on unknown data. Overfitting is often a result of an excessively complicated model.

***Regularization - Tackling overfitting :*** Regularization can be one way in which we can tackle the problem of overfitting. In overfitting we Put constrain the model to make it simpler. The amount of regularization to apply during learning can be controlled by a hyperparameter.

Regularization forced the model to have a smaller slope. Fits a bit less on the training data that the model was trained on but allows it to generalize better to new examples.

***Underfitting :*** It occurs when your model is too simple to learn the underlying structure of

the data. Underfitting is often a result of an excessively simple model. It happens when :

* Features do not provide enough information to make good predictions
* When the model or the algorithm does not fit the data well enough
* Model is too simple

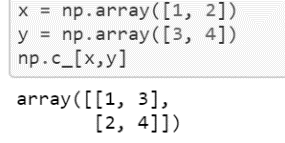
**Solution :**

* Select a more powerful model
* Feed better features to training algorithm
* Reduce the constraints on the model (e.g., reduce the regularization hyperparameter)

***Hyperparameters*** : Parameters that cannot be directly learned from the model training process. Ex : Number of leaves or depth of a tree, Number of hidden layers in a deep neural network etc.

***Skewed dataset* :** When some classes are much more frequent than others, then the dataset is said to be skewed. (Like in MNIST Data, when we convert it to Binary Dataset 5 or not 5 Problem).

***Np.c\_:***



**Standard Notations :**

* **m** is the no. of instances in Training Dataset.
* x(i) is a vector of all the feature values (excluding the label) of the ith instance in the dataset
* y(i) is its label (the desired output value for that instance)
* X is a matrix, contains all the feature values (excluding labels) of all instances in the dataset
  + There is one row per instance and the ith row is equal to the transpose of x(i) ,noted (x(i))T
* h is system’s prediction function, also called as hypothesis. When we train a model using instance feature vector x(i), We get predicted value ŷ(i) = h(x(i)) for that instance
* RMSE(X, h) is the cost function

**numpy.ravel() :** returns flattened one-dimensional Array.

**numpy.random.randint(low,high=None,size=None,dtype=’I’) :** random integers from the “Discrete Uniform Distributon”

**numpy.random.randn(d0,d1,…dn)**: returns d-shape of array of floats from “Standard Normal Distribution”. If nothing is passed, it will return one float.