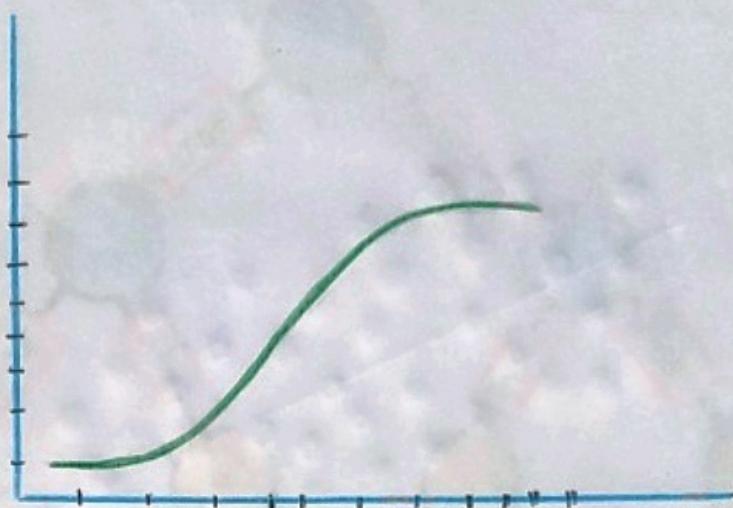


## Linear Regression

**MERITS :-** Simple to implement and efficient to train.  
- Overfitting can be reduced by regularization.  
- Performs well when the dataset is linearly separable.

**DEMERITS :-** Assumes that the data is independent which is rare in real life.  
- Prone to noise and overfitting.  
- Sensitive to outliers.



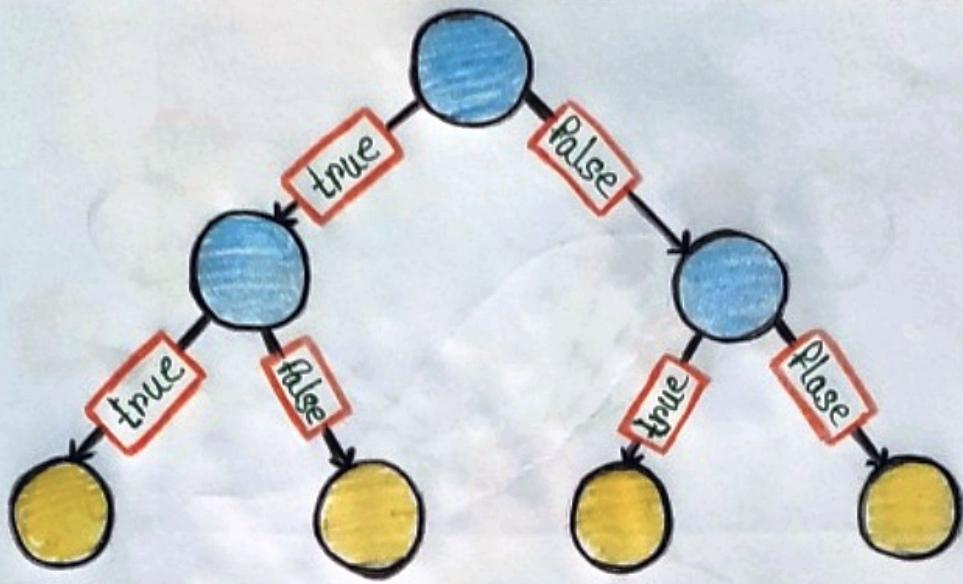
## Logistic Regression

**MERITS :-** Less prone to over-fitting but it can overfit in high dimensional datasets.

- Efficient when the dataset has features that are linearly separable.
- Easy to implement and efficient to train.

**DEMERITS :-** Should not be used when the number of observations are lesser than the number of features.

- Assumption of linearity which is ~~rare~~ in practise.
- Can only be used to predict discrete function.



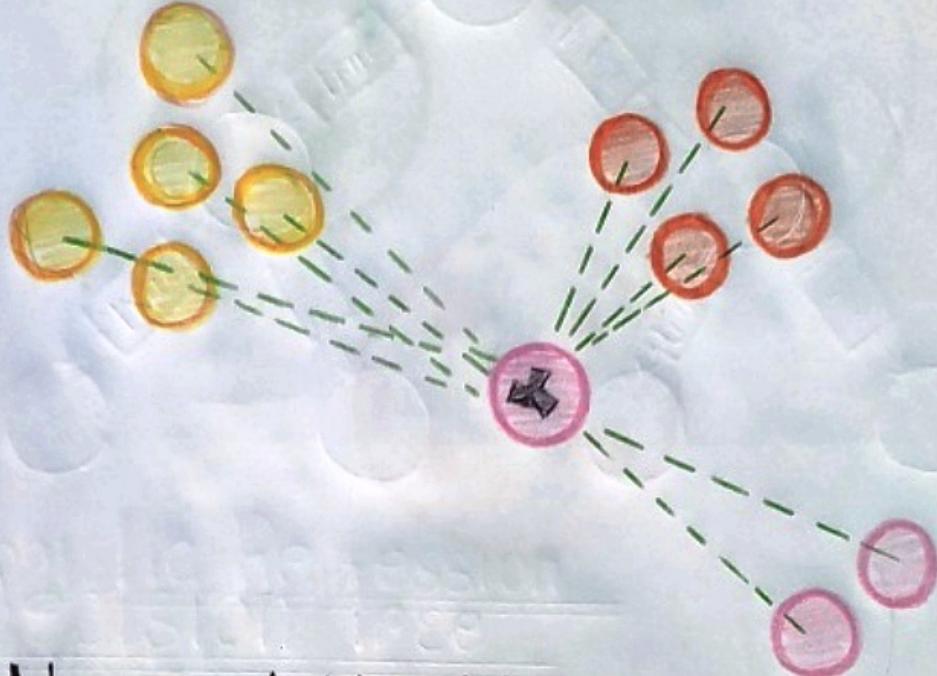
## Decision Tree

**MERITS :-** Can solve non-linear problems.

- Can work on high-dimensional data with excellent accuracy.
- Easy to visualize and explain.

**DEMERITS :-** Overfitting. Might be resolved by random forest.

- A small change in the data can lead to a large change in the structure of the optimal decision tree.
- Calculations can get very complex.



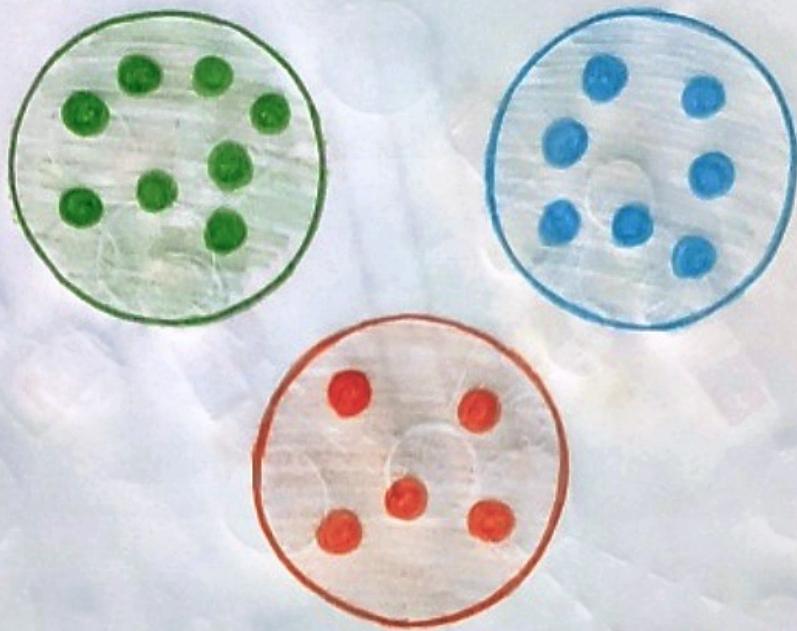
## K Nearest Neighbour

**MERITS :-** Can make predictions without training.

- Time complexity is  $O(n)$ .
- Can be used for both classification and regression.

**DEMERITS :-** Does not work well with large dataset.

- Sensitive to noisy data, missing values and outliers.
- Need feature scaling.
- Choose the correct K value.



## K Means

MERITS :- Simple to implement.

- Scales to large data sets.
- Guarantees convergence.
- Easily adapts to new examples.
- Generalizes to clusters of different shapes and sizes.

DEMERITE :- Sensitive to the outliers.

- Choosing the k values manually is tough.
- Dependent on initial values.
- Scalability decreases when dimension increases.



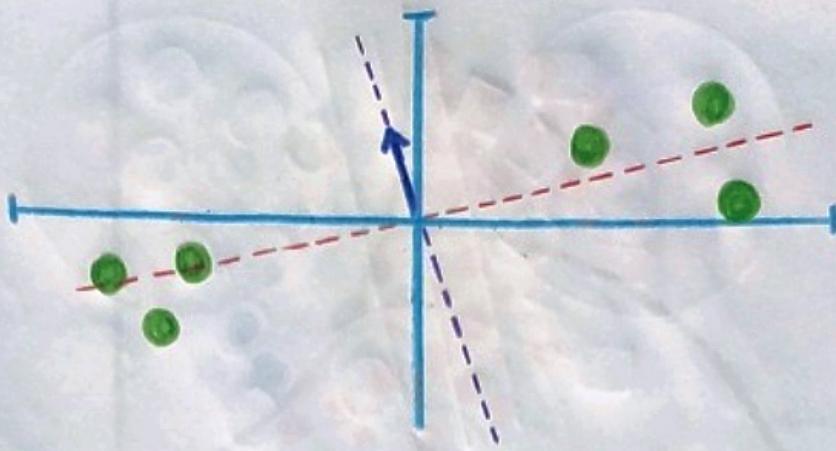
## Support Vector Machine

MERITS :- Good at high dimensional data.

- Can work on small dataset.
- Can solve non-linear problems.

DEMERITS :- Inefficient on large data.

- Requires picking the right kernel.



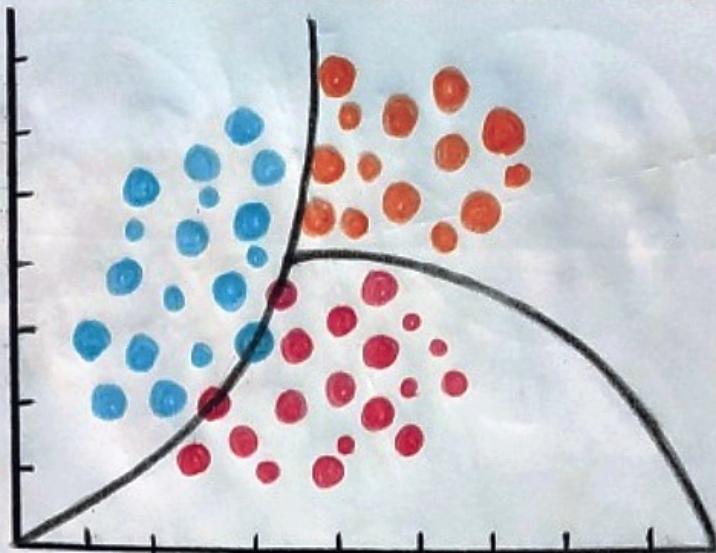
## Principal Component Analysis

**MERITS :-** Reduce correlated features.

- Improve performance.
- Reduce overfitting.

**DEMERITS :-** Principal components are less interpretable.

- Information loss.
- Must standardize data before implementing PCA.



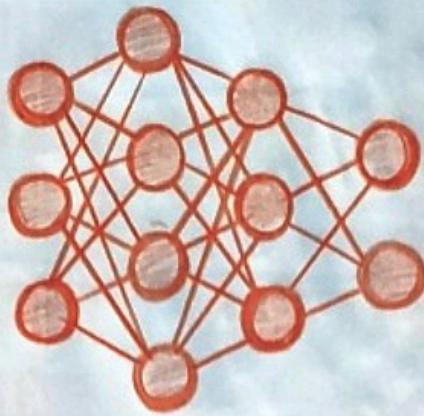
## Naive Bayes

**MERITS :-** Training period is less.

- Better suited for categorical inputs.
- Easy to implement.

**DEMERITS :-** Assumes that all features are independent which is rarely happening in real life.

- Zero Frequency.
- Estimations can be wrong in some cases.



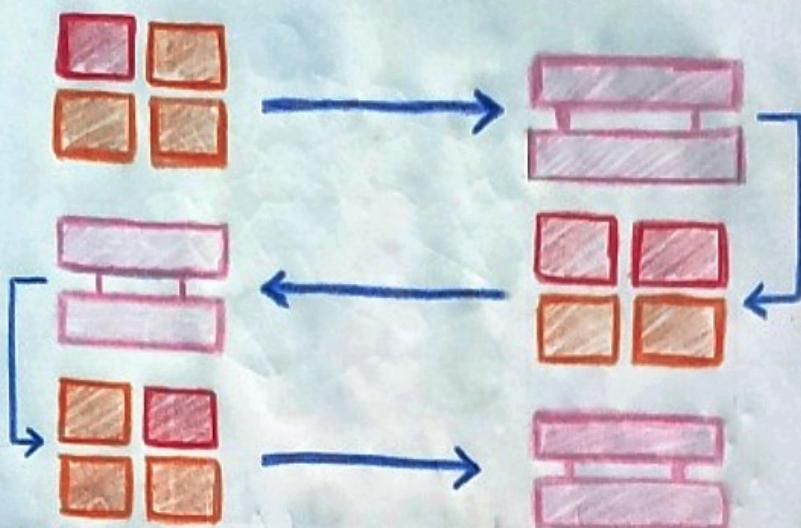
## Artificial Neural Network (ANN)

**MERITS :-** Have fault tolerance.

- Have the ability to learn and model non-linear and complex relationships.
- Can generalize on unseen data.

**DEMERITS :-** Long training time.

- Non-guaranteed convergence.
- Black box. Hard to explain solution.
- Hardware dependence.
- Requires user's ability to translate the problem.



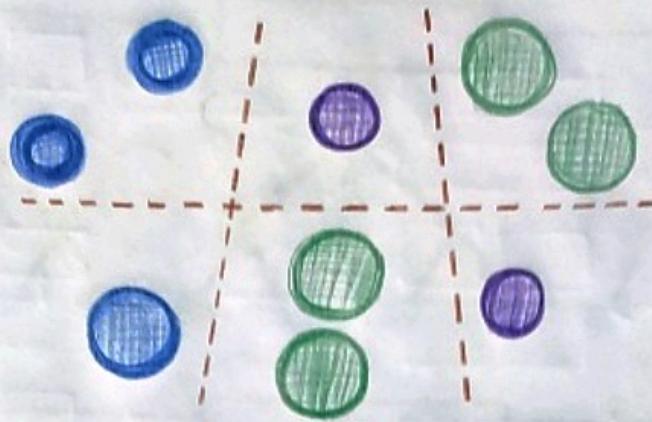
## Adaboost

MERITS :- Relatively robust to overfitting.

- High accuracy.
- Easy to understand and to visualize.

DEMERITS :- Sensitive to noise data.

- Affected by outliers.
- Not optimized for speed.



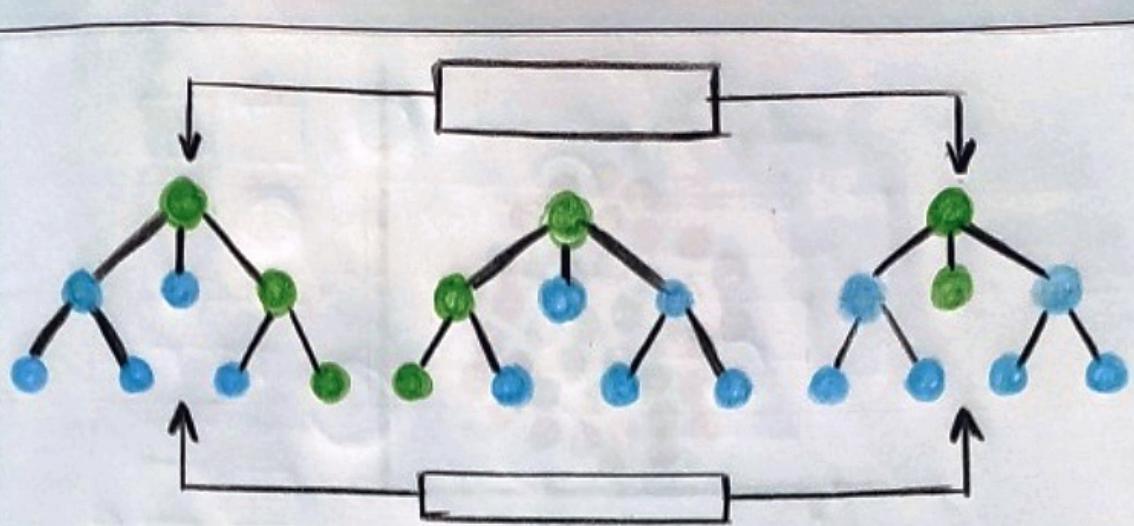
## Gradient Boosting

**MERITS :-** Generally more accurate compare to other modes.

- Train faster especially on larger datasets.
- Most of them provide support handling categorical features.
- Some of them handle missing values natively.

**DEMERITS :-** Can overemphasize outliers and cause overfitting.

- Requires many trees ( $> 1000$ ) which can be time and memory exhaustive.



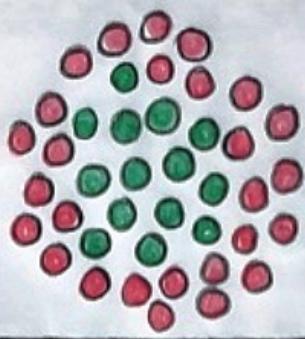
## Random Forest

**MERITS :-** Capable of performing both Classification and Regression tasks.

- Capable of handling large datasets with high dimensionality.
- Enhances the accuracy of the model and prevents the overfitting issue.

**DEMERITS :-** Requires much computational power for large datasets.

- Requires much time for training.
- Can't describe relationships within data.



## DBSCAN

**MERITS :-** Does not require a-priori specification of number of clusters.

- Able to handle outliers within the dataset.
- Able to find arbitrarily size and arbitrarily shaped clusters.

**DEMERITS :-** Struggles with clusters of similar density.

- Struggles with high dimensionality data.