K-Means clustering :

> It is a popular unsupervised machine learning Algorithm.

> It is used for partitioning a dataset into a pre-defined number of clusters.

> The goal is the group similar data points together & discover the patterns of the data.

Why. k-means :

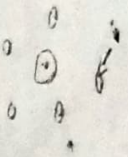
> The k-means clustering Technique is used to minimize the distance of the points in a cluster with their centroid.

> It is a distance-based Algorithm, where we calculate the distance to assign a point to a cluster.

Advantages:

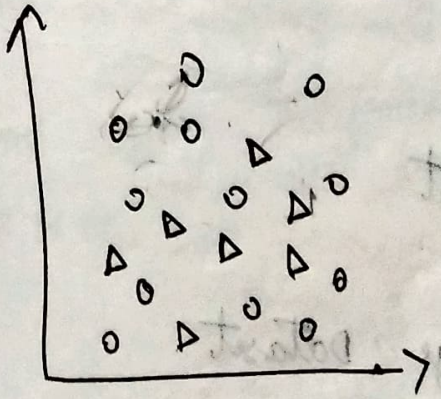
> simple & easy to implement
> Scales well to large dataset
> Generally efficient in practice

Dis advantages:



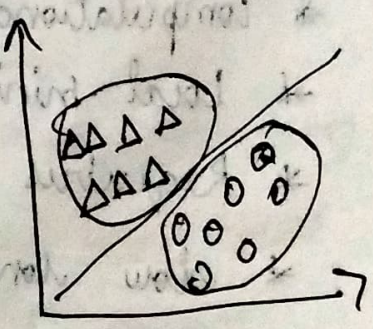
> ~~Not~~ outliers can affect the results
> It struggles with clusters of different shapes.
> predefined no of clusters.

Diagram



BEFORE
K- means

centroid → Mean Value



AFTER.

K - Means clustering:

Q) Cluster the following eight points (with (x, y) represents locations) into three clusters: $\{A_1(2,10), A_2(2,5), A_3(8,4), A_4(5,8), A_5(7,5), A_6(6,4), A_7(1,2), A_8(4,9)\}$.

$x_1 = 2, y_2 = 10$; $x_2 = 2, y_2 = 5$; $x_3 = 8, y_3 = 4$; $x_4 = 5, y_4 = 8$; $x_5 = 7, y_5 = 5$; $x_6 = 6, y_6 = 4$; $x_7 = 1, y_7 = 2$; $x_8 = 4, y_8 = 9$.

soln:

$k = 3$.

step 1:

Random selection of centroid

3 centroid $\rightarrow$ A1 (2, 10)

A4 (5, 8)

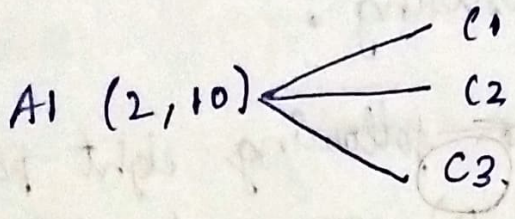
A7 (1, 2).

step 2: compute distance of Every points from centroid.

$c_1 \rightarrow (2, 10)$

$c_2 \rightarrow (5, 8)$

$c_3 \rightarrow (1, 2)$



Distance Formula :

$$d = |x_2 - x_1| + |y_2 - y_1|$$

1. $A_1(2,10)$ $C_1(2,10)$ $\Rightarrow d = |2-2| + |10-10|$
$= 0$

$A_1(2,10)$ $C_2(5,8)$ $\Rightarrow d = |2-5| + |10-8| = 5$

$A_1(2,10)$ $C_3(1,2)$ $\Rightarrow d = |2-1| + |10-2|$
$= 9.$

$\therefore$ **$C_1 = 0\ (2,10)$.**

2. $A_2(2,5)$ $C_1(2,10)$ $\Rightarrow d = 5$

$A_2(2,5)$ $C_2(5,8)$ $\Rightarrow d = 6$

$A_2(2,5)$ $C_3(1,2)$ $\Rightarrow d = 4$

$\therefore$ **$C_3 = 4\ (1,2)$.**

3. $A_3(8,4)\ C_1(2,10) \Rightarrow d = 12$

$A_3(8,4)\ C_2(5,8) \Rightarrow d = 17$

$A_3(8,4)\ C_3(1,2) \Rightarrow d = 9$

$\therefore C_2 = 7(5,8).$

4. $A_4(5,8)\ C_1(2,10) = 5$

$A_4(5,8)\ C_2(5,8) = 0$

$A_4(5,8)\ C_3(1,2) = 10$

$\therefore C_2 = 0(5,8).$

5. $A_5(7,5)\ C_1(2,10) = d = 10$

$A_5(7,5)\ C_2(5,8) = d = 5$

$A_5(7,5)\ C_3(1,2) = d = 9$

$\therefore C_2 = 5(5,8).$

6. $A_6(6,4)\ C_1(2,10) = 10$

$A_6(6,4)\ C_2(5,8) = 5$

$A_6(6,4)\ C_3(1,2) = 7$

$\therefore C_2 = 5(5,8).$

7. $A_7(1,2)$ $c_1(2,10) = 9$

$A_7(1,2)$ $c_2(5,8) = 10$

$A_7(1,2)$ $c_3(1,2) = 0$

$\therefore c_3 = 0(1,2)$

8. $A_8(4,9)$ $c_1(2,10) = 3$

$A_8(4,9)$ $c_2(5,8) = 2$

$A_8(4,9)$ $c_3(1,2) = 10$.

$\therefore c_2 = 2(5,8)$

update centroid :

$c_1 \rightarrow A_1(2,10)$.

$c_2 \rightarrow A_3(8,4)$

$A_4(5,8)$

$A_5(7,5)$

$A_6(6,4)$

$A_8(4,9)$.

$\frac{30}{5} \quad \frac{30}{5}$

$c_2 \rightarrow (6,6)$.

$C_3 \rightarrow A_2(2, 5)$

$A_1(1, 2)$.

$\frac{3}{1}, \frac{7}{2}$

$C_3^{new} = 1.5, 3.5$

step 3 :

For every cluster check it

$C^{new} = = C^{old}$ (Finish it).

else :

Again calculate distance (repeat it).

6) Cluster the following data set using the k-means algorithm with an initial values values of objects 2 and 5 with the coordinate values (4,6) and (12,4) as initial seeds.

Objects	x-coordinate	y-coordinate
1	2	4
2	4	6
3	6	8
4	10	4
5	12	4

soln:

1. Initial Data:

Object 1 : (2,4)
2 : (4,6)
3 : (6,8)
4 : (10,4)
5 : (12,4).

2. Initial Seeds :

(4,6)	(Object 2)
(12,4)	(" 5)

step 1 : Assign objects to the nearest seed.

$$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$$

Distances to seed 1 (4,6).

1. object 1 (2,4) :

$$\sqrt{(4-2)^2+(6-4)^2} = \sqrt{4+4} = \sqrt{8}$$
$$\approx 2.83$$

2. object 2 (4,6) :

$$\sqrt{(4-4)^2+(6-6)^2} = 0$$

3. object 3 (6,8) :

$$\sqrt{(6-4)^2+(8-6)^2} = \sqrt{4+4} = \sqrt{8}$$
$$\approx 2.83$$

4. object 4 (10,4) :

$$\sqrt{(10-4)^2+(4-6)^2} = \sqrt{36+4} = \sqrt{40}$$
$$\approx 6.32$$

5. object 5 (12,4):

$$\sqrt{(12-4)^2+(4-6)^2} = \sqrt{64+4}$$

$$= \sqrt{68} \approx 8.25.$$

Distances to seed 2(12,4):

1. object 1 (2,4):

$$\sqrt{(12-2)^2+(4-4)^2} = \sqrt{100+0} = \sqrt{100}$$

$$= 10.$$

2. object 2 (4,6):

$$\sqrt{(12-4)^2+(6-4)^2} = \sqrt{64+4} = \sqrt{68}.$$

$$\approx 8.25$$

3. object 3 (6,8):

$$\sqrt{(12-6)^2+(8-4)^2} = \sqrt{36+16} = \sqrt{52}$$

$$\approx 7.21$$

4. object 4 (10,4):

$$\sqrt{(12-10)^2+(4-4)^2} = \sqrt{4+0} = \sqrt{4}$$

$$= 2.$$

5. object 5 (12,4):

$$\sqrt{(12-12)^2+(4-4)^2} = 0$$

Step 2 : Calculate the new centroids

cluster 1 (objects 1, 2, 3) :

Mean x - coordinate : $\frac{2+4+6}{3}$

$= \frac{12}{3}$

$= 4.$

Mean y - coordinate : $\frac{4+6+8}{3}$

$= \frac{18}{3}$

$= 6$

New centroid : (4, 6).

cluster 2 (objects 4, 5) :

Mean x - coordinate : $\frac{10+12}{2} = \frac{22}{2} = 11$

" y- " : $\frac{4+4}{2} = \frac{8}{2} = 4$

New centroid : (11, 4)

step 3: Reassign objects to the nearest centroid.

(4,6):

1. object 1 (2,4):

$$\sqrt{(4-2)^2 + (6-4)^2} = \sqrt{8} \simeq 2.83$$

2. object 2 (4,6): 0

3. " 3 (6,8) $= \sqrt{8} \approx 2.83$

4. " 4 (10,4) $= \sqrt{40} \approx 6.32$

5. " 5 (12,4) $= \sqrt{68} \simeq 8.25$

(11,4):

1. object 1 (2,4):

$$\sqrt{(11-2)^2 + (4-4)^2} = 9.$$

2. object 2 (4,6) $\approx$ 7.28.

3. " 3 (6,8) $\approx$ 6.40

4. " 4 (10,4) = 1

5. " 5 (12,4) = 1

cluster 1 (4.6):

object 1, 2, 3.

cluster 2, (11, 4):

object 4, 5.

KNN :

> The K-Nearest Algorithm is a unsupervised machine learning method for classification and regression problems. Mostly in classification.

> It is also an unsupervised learning Algorithm (or) Technique.

> It assumes the similarity between the new case / data & available cases and put the new case into the category that is most similar to the available categories.

> It stores all the available data & classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K-NN Algorithm.

> It is also called a lazy learner Algorithm, because it does not learn from the training set, it performs an action on the dataset (at the time of classification).

Eg:

KNN classifier.

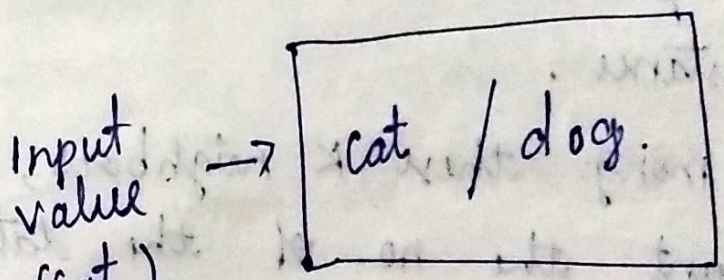
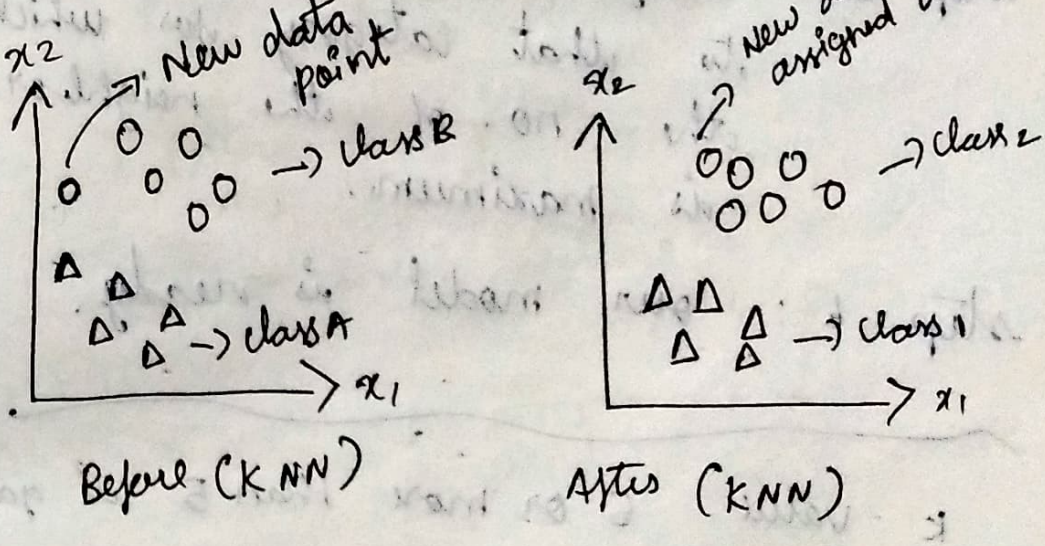


Diagram:



Working:

step 1: select the number k of the neighbors.

step 2: Calculate the Euclidean distance of k number of neighbors.

step 3: Take the k nearest neighbors as per the calculated Euclidean distance.

step 4: Among these k neighbors, count the no. of the data points in each category.

step 5: Assign the new data points to that category for which the no. of the neighbor is maximum.

step 6: our model is ready.

---

k-value 5 or more than 5 is good.

$k < 5$ can be noisy and lead to outliers in the model.

Advantages:

> It is simple to implement
> It is robust to the noisy training data
> It can be more effective if the training data is large.

Disadvantages:

> The computation cost is high.
> prone to overfitting.

Applications:

> Data preprocessing
> Pattern Recognition
> Recommender systems.

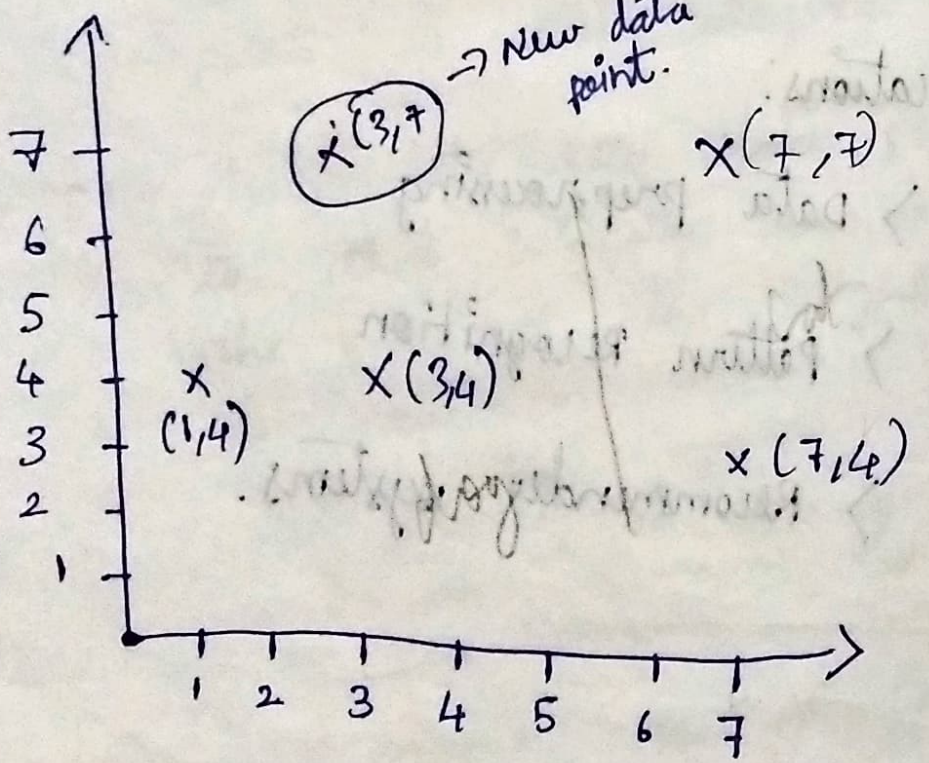
Q

x	y	class
7	7	Bad
7	4	Bad
3	4	Good
1	4	Good

soln:

K = 3,

New Data point (3, 7).



Distance Formula:

$$|x_2 - x_1| + |y_2 - y_1|.$$

(7,7) (3,7) $\Rightarrow |3-7| + |7-7| = 4$ (Bad)

(7,4) (3,7) $\Rightarrow |3-7| + |7-4| = 7$ (Bad)

(3,4) (3,7) $\Rightarrow |3-3| + |7-4| = 3$, (Good)

(1,4) (3,4) $\Rightarrow |3-1| + |7-4| = 5$ (Good).

G B G. K = 3

3, 4, 5, 7

2 Good so, (3,7) is Good.

- Full code $L = 2^{(K-1)} - 1$

$$W = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & +1 & +1 & +1 & +1 \\ -1 & +1 & +1 & -1 & -1 & +1 & +1 \\ +1 & -1 & +1 & -1 & +1 & -1 & +1 \end{bmatrix}$$

- With reasonable L, find W such that the Hamming distance between rows and between columns are maximized.
- Voting scheme are

$$y_i = \sum_{j=1}^{L} W_{ij} d_j$$

and then we choose the class with the highest Y_i.

- One problem with ECOC is that because the code matrix W is set a priori, there is no guarantee that the subtasks as defined by the columns of W will be simple.

3.2 Ensemble Learning

- The idea of ensemble learning is to employ multiple learners and combine their predictions. If we have a committee of M models with uncorrelated errors, simply by averaging them the average error of a model can be reduced by a factor of M.
- Unfortunately, the key assumption that the errors due to the individual models are uncorrelated is unrealistic; in practice, the errors are typically highly correlated, so the reduction in overall error is generally small.
- Ensemble modeling is the process of running two or more related but different analytical models and then synthesizing the results into a single score or spread in order to improve the accuracy of predictive analytics and data mining applications.
- Ensembles of classifiers is a set of classifiers whose individual decisions combined in some way to classify new examples.
- Ensemble methods combine several decision trees classifiers to produce better predictive performance than a single decision tree classifier. The main principle behind the ensemble model is that a group of weak learners come together to form a strong learner, thus increasing the accuracy of the model.
- Why do ensemble methods work ?
- Based on one of two basic observations :
  1. Variance reduction : If the training sets are completely independent, it will always helps to average an ensemble because this will reduce variance without affecting bias (e.g., bagging) and reduce sensitivity to individual data points.

2. Bias reduction : For simple models, average of models has much greater capacity than single model Averaging models can reduce bias substantially by increasing capacity and control variance by Citting one component at a time.

3.2.1 Bagging

- Bagging is also called Bootstrap aggregating. Bagging and boosting are meta - algorithms that pool decisions from multiple classifiers. It creates ensembles by repeatedly randomly resampling the training data.
- Bagging was the first effective method of ensemble learning and is one of the simplest methods of arching. The meta - algorithm, which is a special case of the model averaging, was originally designed for classification and is usually applied to decision tree models, but it can be used with any type of model for classification or regression.
- Ensemble classifiers such as bagging, boosting and model averaging are known to have improved accuracy and robustness over a single model. Although unsupervised models, such as clustering, do not directly generate label prediction for each individual, they provide useful constraints for the joint prediction of a set of related objects.
- For given a training set of size n, create m samples of size n by drawing n examples from the original data, with replacement. Each *bootstrap sample* will on average contain 63.2 % of the unique training examples, the rest are replicates. It combines the m resulting models using simple majority vote.
- In particular, on each round, the base learner is trained on what is often called a "**bootstrap replicate**" of the original training set. Suppose the training set consists of n examples. Then a bootstrap replicate is a new training set that also consists of n examples, and which is formed by repeatedly selecting uniformly at random and with replacement n examples from the original training set. This means that the same example may appear multiple times in the bootstrap replicate, or it may appear not at all.
- It also decreases error by decreasing the variance in the results due to *unstable learners*, algorithms (like decision trees) whose output can change dramatically when the training data is slightly changed.
- **Pseudocode :**
  1. Given training data $(x_1, y_1), \ldots, (x_m, y_m)$
  2. For $t = 1, \ldots, T$:
     a. Form bootstrap replicate dataset S_t by selecting m random examples from the training set with replacement.

b. Let h_t be the result of training base learning algorithm on S_t.

3. Output combined classifier :
$H(x) = \text{majority}(h_1(x), \ldots, h_T(x))$

Bagging Steps :

1. Suppose there are N observations and M features in training data set. A sample from training data set is taken randomly with replacement.
2. A subset of M features is selected randomly and whichever feature gives the best split is used to split the node iteratively.
3. The tree is grown to the largest.
4. Above steps are repeated n times and prediction is given based on the aggregation of predictions from n number of trees.

Advantages of Bagging :

1. Reduces over - fitting of the model.
2. Handles higher dimensionality data very well.
3. Maintains accuracy for missing data.

Disadvantages of Bagging :

1. Since final prediction is based on the mean predictions from subset trees, it won't give precise values for the classification and regression model.

3.2.2 Boosting

- Boosting is a very different method to generate multiple predictions (function estimates) and combine them linearly. Boosting refers to a general and provably effective method of producing a very accurate classifier by combining rough and moderately inaccurate rules of thumb.
- Originally developed by computational learning theorists to guarantee performance improvements on fitting training data for a ***weak learner*** that only needs to generate a hypothesis with a training accuracy greater than 0.5. Final result is the weighted sum of the results of weak classifiers.
- A learner is weak if it produces a classifier that is only slightly better than random guessing, while a learner is said to be strong if it produces a classifier that achieves a low error with high confidence for a given concept.
- Revised to be a practical algorithm, AdaBoost, for building ensembles that empirically improves generalization performance. Examples are given weights. At

each iteration, a new hypothesis is learned and the examples are reweighted to focus the system on examples that the most recently learned classifier got wrong.

- Boosting is a bias reduction technique. It typically improves the performance of a single tree model. A reason for this is that we often cannot construct trees which are sufficiently large due to thinning out of observations in the terminal nodes.
- Boosting is then a device to come up with a more complex solution by taking linear combination of trees. In presence of high - dimensional predictors, boosting is also very useful as a regularization technique for additive or interaction modeling.
- To begin, we define an algorithm for finding the rules of thumb, which we call a weak learner. The boosting algorithm repeatedly calls this weak learner, each time feeding it a different distribution over the training data. Each call generates a weak classifier and we must combine all of these into a single classifier that, hopefully, is much more accurate than any one of the rules.
- Train a set of weak hypotheses : $h_1, ..., h_T$. The combined hypothesis H is a weighted majority vote of the T weak hypotheses. During the training, focus on the examples that are misclassified.

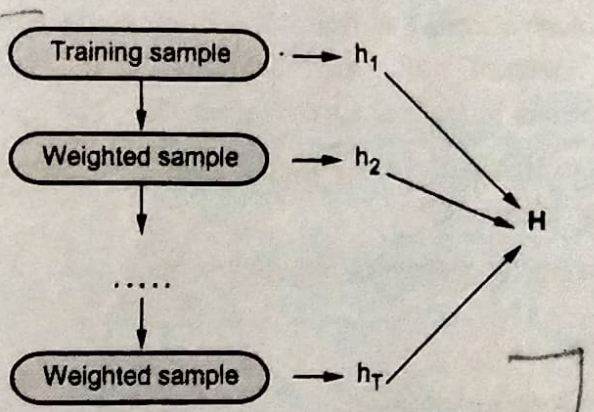


Fig. 3.2.1

AdaBoost :

- AdaBoost, short for "Adaptive Boosting", is a machine learning meta - algorithm formulated by Yoav Freund and Robert Schapire who won the prestigious "Gödel Prize" in 2003 for their work. It can be used in conjunction with many other types of learning algorithms to improve their performance.
- It can be used to learn weak classifiers and final classification based on weighted vote of weak classifiers.
- It is linear classifier with all its desirable properties. It has good generalization properties.

- To use the weak learner to form a highly accurate prediction rule by calling the weak learner repeatedly on different distributions over the training examples.
- Initially, all weights are set equally, but each round the weights of incorrectly classified examples are increased so that those observations that the previously classifier poorly predicts receive greater weight on the next iteration.
- **Advantages of AdaBoost :**
  1. Very simple to implement
  2. Fairly good generalization
  3. The prior error need not be known ahead of time.

Disadvantages of AdaBoost :

1. Suboptimal solution
2. Can over fit in presence of noise.

Boosting Steps :

1. Draw a random subset of training samples d1 without replacement from the training set D to train a weak learner C1
2. Draw second random training subset d2 without replacement from the training set and add 50 percent of the samples that were previously falsely classified/misclassified to train a weak learner C2
3. Find the training samples d3 in the training set D on which C1 and C2 disagree to train a third weak learner C3
4. Combine all the weak learners via majority voting.

Advantages of Boosting :

1. Supports different loss function.
2. Works well with interactions.

Disadvantages of Boosting :

1. Prone to over-fitting.
2. Requires careful tuning of different hyper - parameters.

3.2.3 Stacking

- Stacking, sometimes called stacked generalization, is an ensemble machine learning method that combines multiple heterogeneous base or component models via a meta-model.

- The base model is trained on the complete training data, and then the meta-model is trained on the predictions of the base models. The advantage of stacking is the ability to explore the solution space with different models in the same problem.
- The stacking based model can be visualized in levels and has at least two levels of the models. The first level typically trains the two or more base learners(can be heterogeneous) and the second level might be a single meta learner that utilizes the base models predictions as input and gives the final result as output. A stacked model can have more than two such levels but increasing the levels doesn't always guarantee better performance.
- In the classification tasks, often logistic regression is used as a meta learner, while linear regression is more suitable as a meta learner for regression-based tasks.
- **Stacking** is concerned with combining multiple classifiers generated by different learning algorithms $L_1,\ldots,L_N$ on a single dataset S, which is composed by a feature vector $S_i = (x_i, t_i)$
- The stacking process can be broken into two phases :
  1. Generate a set of base - level classifiers $C_1,\ldots,C_N$ where $C_i = L_i(S)$
  2. Train a meta - level classifier to combine the outputs of the base - level classifiers.
- Fig. 3.2.2 shows stacking frame.

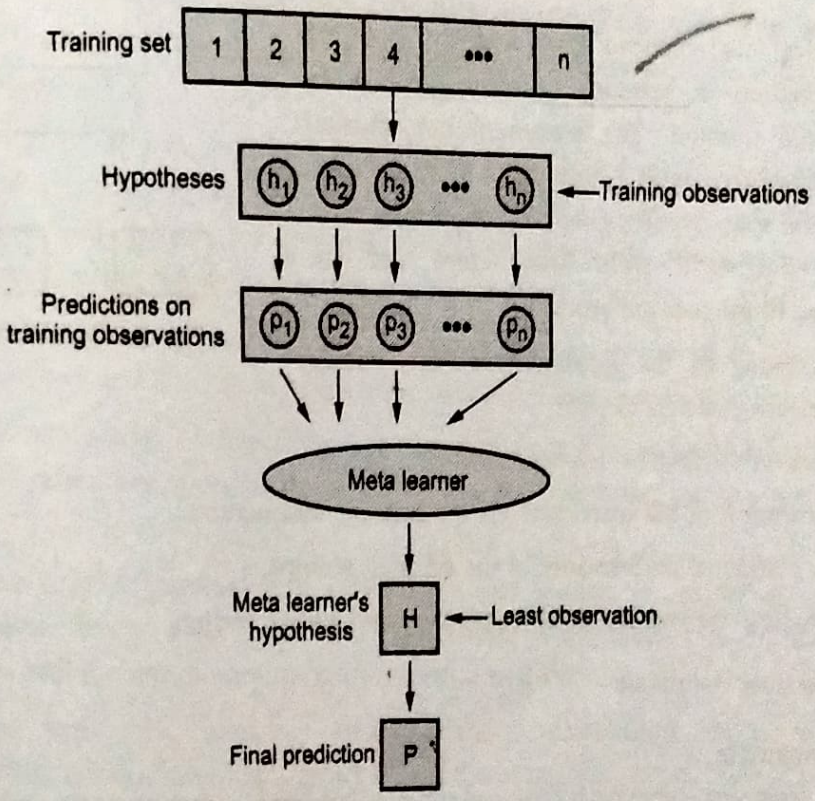


Fig. 3.2.2 Stacking frame