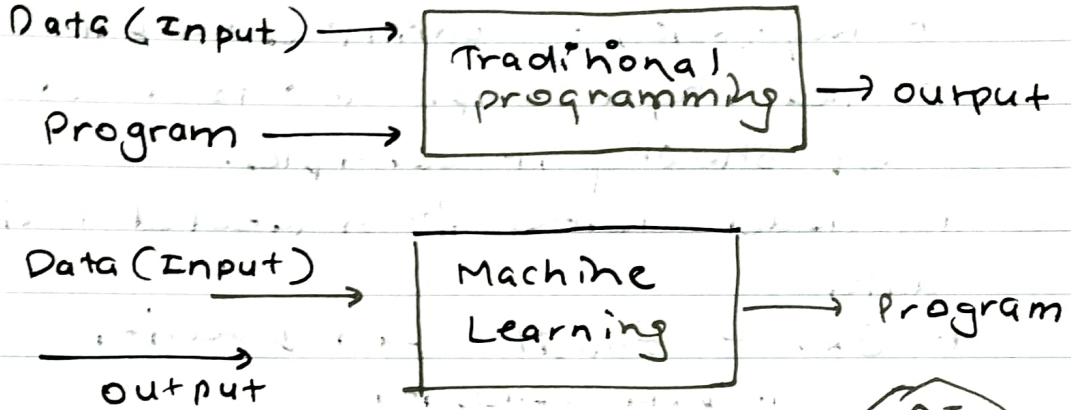
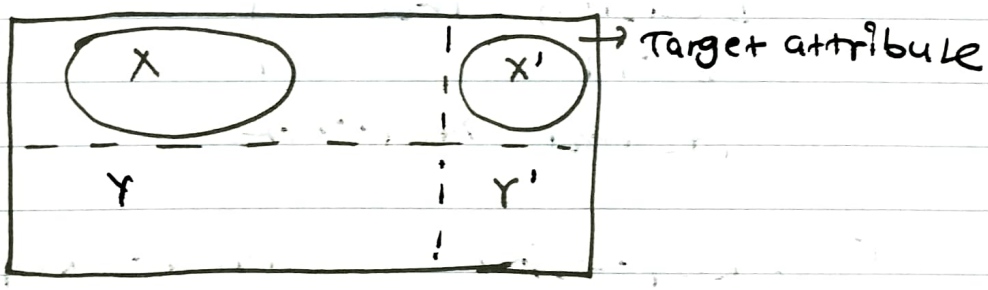
Machine Learning

It is the field of study that gives computers the capability to learn without being explicitly programmed.



Training & Testing



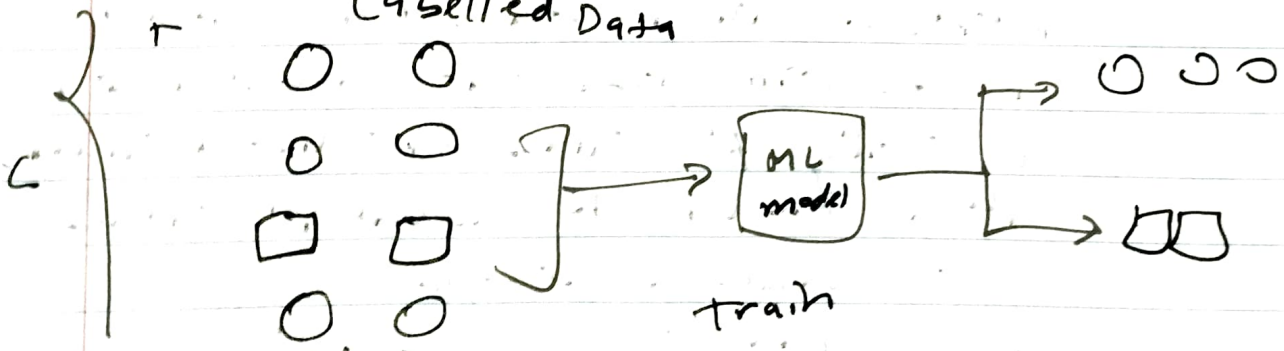
P

+ we take X & X' for training purpose. Then create one model.
+ We will test our model for Y
+ verify Y' values (predicted values) from model to actual Y' values to check accuracy.
+ Handwriting Recognition System

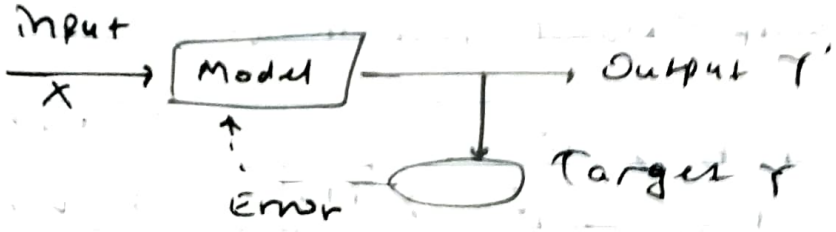
ex

ML approaches

① Supervised learning
+ classification - categorical output
+ Regression - continuous output.
+ teaching a comp. how to do something by showing it examples.
+ model learns from labeled training data
+ Ex. - Email Spam Detection
    1. Data Collection
    2. Training
    3. Testing
+ KNN
  Decision Tree
  SVM
  Neural Networks
  } classification
+ Regression
  } Linear Reg
  Polynomial Reg.
  Reg. Trees
+ Labelled Data



+

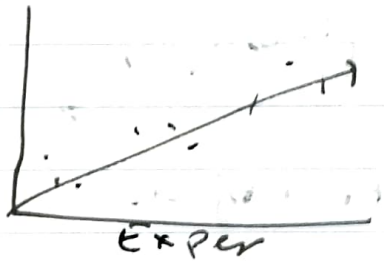


+ Ex.

1. Credit Card Fraud Det
2. Sentiment analysis
3. Chum Prediction
4. Medical Diagnosis
5. Spam mail Detection

} Class.

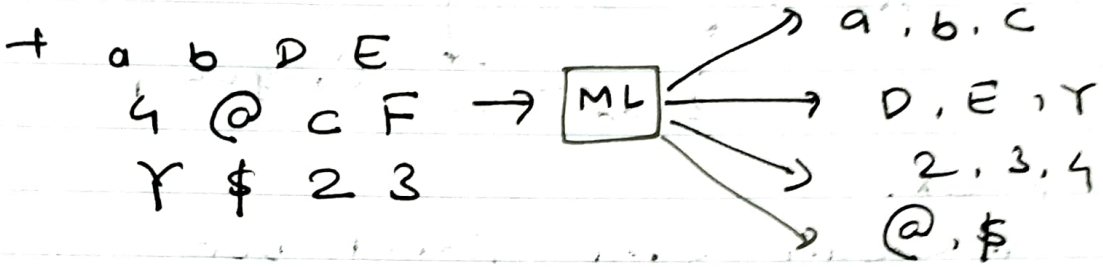
+ Regression



Ex.

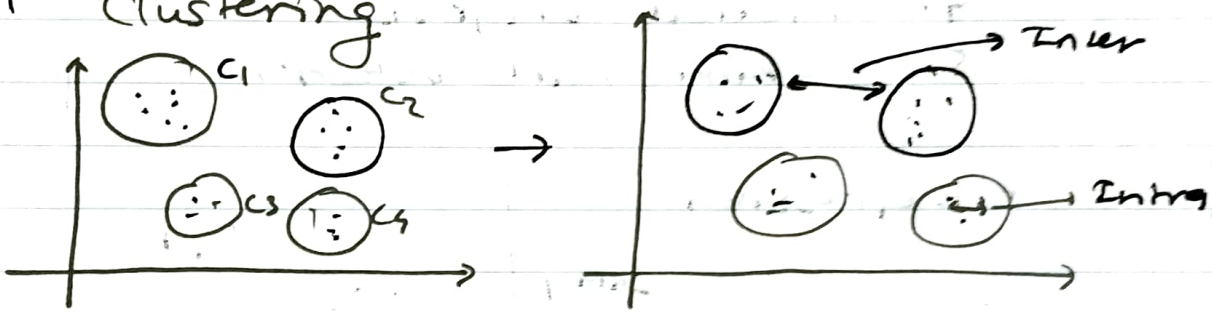
1. Time series prediction.
   a. Rainfall on certain region
   b. Spend on voice call
2. Price prediction
3. Trend analysis
   a. Linear or Exponential

② Unsupervised learning.



no labels

+ Clustering



data objects

Clustered data objects

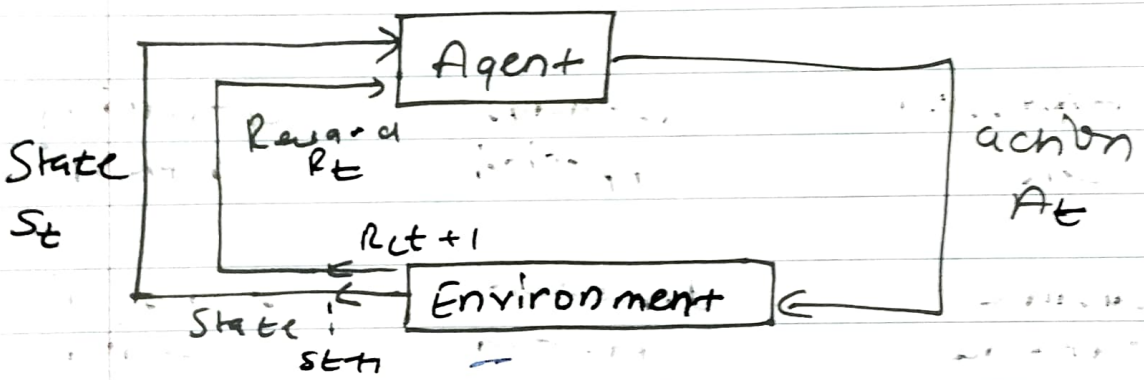
+ Clustering applications
  1. Customer data
     a. discover classes of cust
  2. Image pixels
     a. Discover regions
  3. Word synonym
  4. Document

+ Predicting co-occurrences
+ Market basket analysis
+ Time series analysis

+ K-means clustering

+ Association Rule mining
  a. mining frequent pattern
  b. associat^n rules
  c. condit^nal dependencies
  d. Bnd frequent pattern
  e. derive associat^n rule

(3) Reinforcement Learning

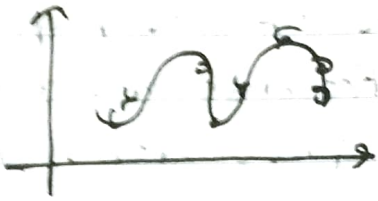


+ Applications
  a. Busineps - Strategizing
  b. Gaming - Playstation
  c. Recommendat^n System
  d. Intelligent tutorial system.

+ sup x , unsup x
+ react to envir. on their own
+ rapid grow , algo ↑

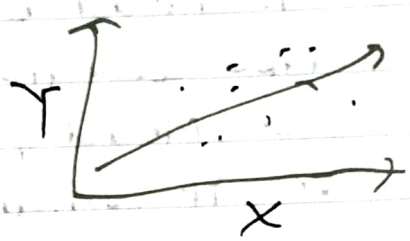
	Supervised	Unsupervised	Reinforcement
	labelled data	unlabelled data. no guidence	environment
Problems	Regression ~~Clust~~ Classif	clustering assocatn	Explorat n Exploitation
	extra supervision	no super	no super.
	KNN, SVM LR	ic-means Aprriori	q-learning SARSA
	calculate outcomes	discover Pattern	series of action
	risk evaluation forecast	Recommedatn anamily detec	selfdriving cars gaming healthcare

Overfitting



+ Extra features

Underfitting



+ less features

Underfitting –

+ ML algo. said to be underfitting when a model is too simple to capture data complexities

+ inability of model to learn the training data effectively

+ It result in poor performance both on ~~test~~ training & testing.

+ Inaccurate when applied to unseen ex.

Reas.

+ simplified assumptions

+ High Bias & low variance

+ size of training dataset is not enough

+ features are not scaled.

Tech

+ model complex

+ ↑ featu

+ remove noise

+ duratⁿ of training.

Overfitting

1. Model is said ... when it does not make accurate predictions of testing data.
2. Model gets trained from so much data so it contains (noise & inaccurate entries).
3. Non-parametric & non-linear
4. Linear Algor. (sol.) . D Tree

Reasons

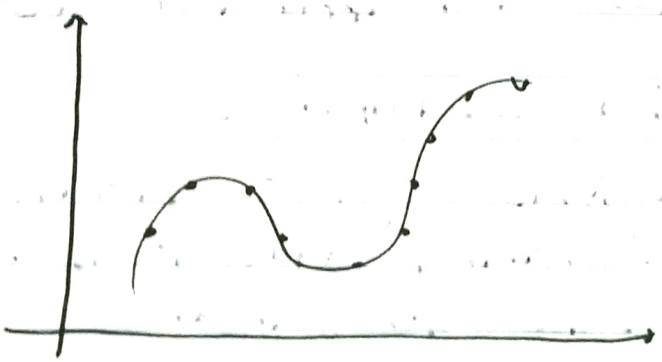
1. Low Bias & High Variance
2. model is too complex
3. size of training data.

Tech

1. Reduce model complexity
2. Ridge regularizatn & lasso regularizatn
3. Add accurate depth to the parameters

Ex. Ball

Diag.



Feature Selection Techniques

1) Filter methods
   + Information gain
   + chi-square test
   + correlation coeff.

2) Wrapper methods

① Variance threshold
+ simple appr.
+ removes features whose variance doesn't meet some threshold.
+ remove all 0 variance features
+ features that have same value in all samples.

```
from sklearn.feature_selection import VarianceT...

x = [ [0, a.b, 3] , [0...., 3.] ]

selector = VarianceThreshold()
selector.fit_transform(x)
    array ( [ [   ], [ ] ])
```

② K-best

+ uses statistical tests like chi-square, ANOVA F-test to rank features based relationship output.
+ select k features (highest score)
+ Parameters - score fn & K
+ Score fn evaluate → feature imp.
+ f-regression - LRP, F-value blw T & Fea
+ mutual_info_regression - 2 random variables
+ f_classif - classificatn, ANOVA fvalue F & T
+ mutual_info_classif - 2 random
+ chi2 - class, F & T chi² sta.
+ SelectPercentile - high %, score based.

```
f sklearn import load_digits
f sklearn.feature_selection import SelectKBest, chi2
X, y = load_digits(return_X_y=True)
```

```
X.shape
(1532, 64)
```

```
X_new = SelectKBest(chi2, k=20).
        fit_transform(X, y)
```

③ Select Percentile

+ select feature according to percentile of US.

Principal Component analysis

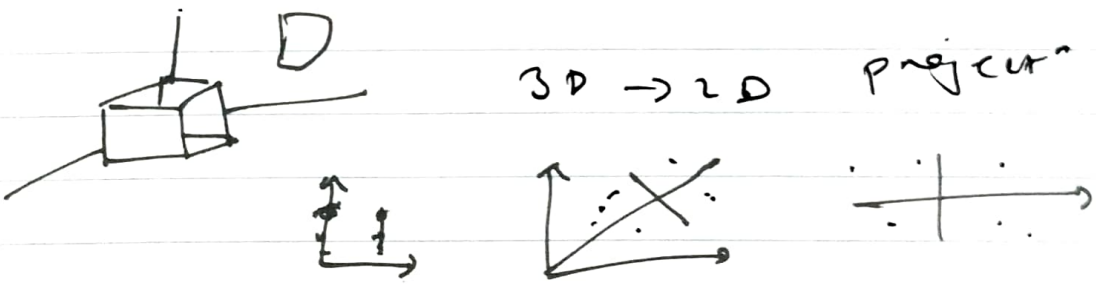
1. Dimensionality reduct^n tech
2. Higher Dim - Overfitting - not generalize
3. to project High D → Low Dim & also retain relat^n, info, pattern
4. set of features → principal componen

Advantages -

+ remove correlated features
+ improve mLalgo performance
+ reduce overfitting

Disadvantages -

+ info loss
+ Difficult to select features ovn PCA



Steps

1] Standardizatn
+ ~~smooth~~ Similar range - all variables equal contribut^n

2] Covariance matrix
+ understand variance
+ Find relat b/w Feats

3] Eigen value & eigen vectors
+ CM se compute → Principal components

Properties —

+ linear combinat^n of variables of original ds.
+ PCs are ⊥
+ 0 correlat^n blw pair of features
+ PC1 , ... PCN , Imp ↓

ML unit 2.

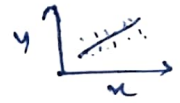
+ Linear Regression
  + statistical method used to used to model relationship between dependent variable y and one or more independent variable x.
  + It seeks to find best fitting straight line
  + $y = B_0 + B_1 x + \varepsilon$

    B_0 - intercept B_1 = slope.

    ε = error term - difference between error term and predicted values.

+ Assumptions of linear regression
  + linearity
  + Independance - observations are independant of each other
  + Homo scedasticity - variance of residuals - constant difference between observed and predicted value - across all independent variable.
  + Normality - residuals are normally distributed
  + No multicollinearity - Independent variables are not highly correlated with each other.

+ Bidimensional example.
  + Two variables involved (dependent and independent)
  + e.g. study hours and marks gained.



+. linear regression

$y = \alpha + \beta x.$

$y_i = B_0 + B_1 x_i + \varepsilon_i \quad i = 1, 2, \ldots n.$

+ Assumptions.
  + y_i depends only on x_i and variation is y_i is random.
  + variance of ε or y doesnot depend on x_i.
  + $E(\varepsilon_i) = 0$ for all i

    $var(\varepsilon_i) = \sigma^2$ for all i

    ~~other~~ $Var(y) = \sigma^2$
  + $Cov(\varepsilon_i, \varepsilon_j) = 0$, ~~other~~ $cov(y_i, y_j) = 0$.
  + ε variables or y variables are unrelated to each other.

+ $B_1 = \dfrac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} = \dfrac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$

$B_0 = \bar{y} - B_1\bar{x}$

Hours studied (x)	Test scores (y)	$x_i - \bar{x}$	$y - \bar{y}$	$(x_i - \bar{x})^2$
2	65	-2	-10	4
3	70	-1	-5	1
4	75	0	0	0
5	80	+1	5	1
6	85	+2	10	4

$\bar{x} = \dfrac{2+3+4+5+6}{5} = 4$

$\bar{y} = \dfrac{375}{5} = 75$

$B_1 = \dfrac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} = \dfrac{-2\times-10 + -1\times-5 + 5 + 20}{4+1+1+4.}$

$= \dfrac{20+5+5+20}{10}$

$= \dfrac{50}{10} = 5.$

$B_0 = \bar{y} - B_1\bar{x}$

$B_0 = 75 - 5\times 4 = 55$

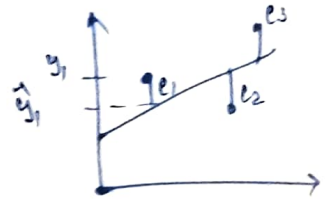
$\therefore \quad y = 55 + 5x$

+ Predicted and residuals.

predicted – find using linear Regression

residual – deviation of observed from predicted value.

$e_1 = y_1 - \hat{y}_1$



+ sum of square error (SSE)

$$SSE = \sum (e_i)^2 = \sum (y_i - \hat{y}_i)^2$$

(y_i: observed, $\hat{y}_i$: predicted)

low SSE → better fit model.

explains unexplained variation in y.

+ SST (Total sum of squares)

$$SST = \sum (y_i - \bar{y})^2$$

total variability in y.

+ R^2 = Coefficient of determination.

$$R^2 = 1 - \frac{SSE}{SST}$$

+ $(y_i - y_{mean})^2 = (y_i - y_{predi})^2 + (y_{predi} - y_{mean})^2$

$TSS = RSS + ESS$.

but $R^2 = 1 - \frac{RSS}{TSS}$

$$R^2 = \frac{ESS}{TSS}$$

+ R^2 → determines proportion of variance in dependent variable that can be explained by independent variable.

$$R^2 = \frac{\text{Variance explained by modal}}{\text{Total variance}} = \frac{SSR}{SST}$$

R^2 → coefficient of determination.

+ Adjusted R^2 → modified R^2 → adjusts to number of independent variable.

+ only those independent variable which helps in explaining proportion of variation in dependent variable

$$R^2 \text{ adjusted} = 1 - \frac{(1-R^2)(N-1)}{N-p-1}$$

R^2 = sample R square.

P = number of predictors. (no of independent variables).

N = Total sample size.

can be negative.

-ve → worse model.

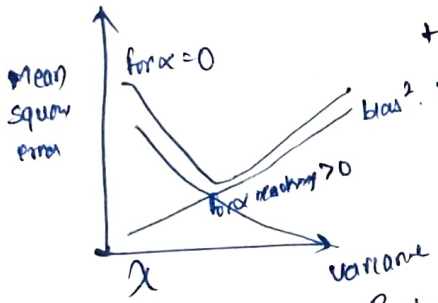
+ Regularization.

+ prevent overfitting
+ penalty term
+ improve generalization performance of model.

+ Ridge regression

+ adds penalty to squared magnitude of coefficient.

+ large coefficient → heavy penalty.



+ shrinking the coefficient.

$y = 0.9 + 1.5x \rightarrow y = 0.9 + 2.0x$.

+ reduces variance of predictions.

Ridge R = Loss + $\alpha ||w||^2$ = RSS + $\alpha ||w||^2$. → $\Sigma (y_i - \hat{y})^2$

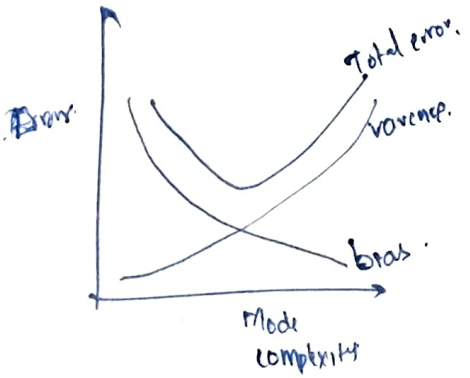
$||w||^2 = w_1^2 + w_2^2 \ldots w_n^2$.

$\alpha \uparrow \rightarrow$ more shrinking.

+ predictor variables → highly correlated → multicollinearity

↓

takes care of that predictor ← high variance.



variance → change in R if we train on different set.

bias → error introduced

Irreducible error → error that cannot be reduced.

+ Lasso Regression.
  + adds penalty to absolute value.
  + shrinks coefficient to 0
  + feature selection and removing irrelevant features

Lasso R = Loss + $\alpha \|w\|$
penalty

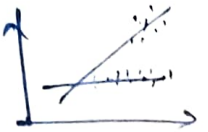
$$\|w\| = |w_1| + |w_2| \dots |w_n|$$

+ Elastic Net Regression.
  + Elastic Net Regression = loss + $\alpha_1 \|w\|^2 + \alpha_2 \|w\|$
  + for feature selection
  + for multicollinearity.
  + emerged because → lasso is too dependent on data.

$$\text{Inet}(\beta) \quad \frac{\sum_{i=1}^{n} (y_i - x_i \vec{\beta})^2}{2n} + \lambda \left(\frac{1-\alpha}{2} \sum_{j=1}^{n} \hat{\beta}_j^2 + \alpha \sum_{j=1}^{n} |\hat{\beta}_j| \right)$$

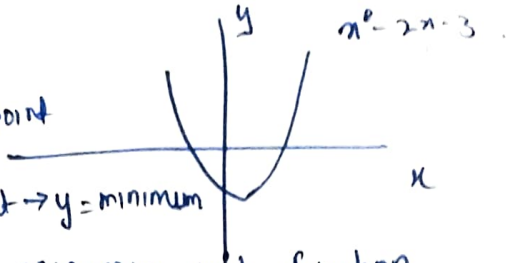
α = mixing parameter ridge ($\alpha = 0$)
lasso ($\alpha = 1$)

+ Robust Regression with Random Sample Consensus (RANSAC)
  + Outlier containing dataset.
  + outliers → biased result.
  + RANSAC → identifies outliers → ignores them
  + RANSAC → perform iterations → until inliers remain
  + noise ignored



  + threshold → defined → for outlier identification.

+. Gradient descent.



+ descending slope to reach lowest point
+ objective → find x such that → y = minimum
+ Iterative update parameters → minimize cost function.
+ minimize sum of squared residuals → errors.
+ Starts with random point → descend the slop → reach required position.
+ Algorithm
  + find slope → with respect to each function
  + pick a random initial value.
  + differentiate y with respect to x.
  + update the gradient function.
  + calculate step size for each feature
    step size = gradient * learning rate.
  + New parameter = old params − step size.
  + repeat it until gradient is almost 0.
+ stochastic gradient descent
  + for random small part of observation rather than all values.
+ convergence → ~~reched~~ reaching a threshold of iterations
  → thus reaching threshold of finding magnitude of gradient.

+ Hyper parameters
  + cannot be determined from data
  + not directly learned from training process
  + set prior to training
  + influences → complexity, optimization algorithm used etc.
  + eg. learning rate in gradient descent, number of hidden layers
    Regularization parameter.

+ Grid Search.
  + specify grid of hyperparameters
  + evaluate each
  + finds best performance.

+ Repeated k-fold → + Training data into k folds.
  + for each hyper paramet. train on k-1 fold.
  + select best paramet.
  + k models fit on $\frac{k-1}{k}$ data (training split).
  + evaluated on $\frac{1}{k}$ of data (called test split).

Iteration 1 | Iteration 2 | Iteration 3 | Iteration 4
fold1 testdata | fold1 | fold1 | fold1
fold1 | fold1 | fold1 | fold1
fold3 training data. | fold3 | fold3 | fold3
k | k | k | k

cross validation

+ All data training da

+ Polynomial regression.
  + linear model in non linearity dataset.

+ $y = \alpha_0 \sum \alpha_i x_i$ + fp
  ↳ polynomial functions
  x_1, x_1, x_1^2, x_1^3

+ covariance matrix.

relationship between variable and their variability.

$$\begin{bmatrix} \text{var}(x_1) & \text{cov}(x_1, x_2) \\ \text{cov}(x_2, x_1) & \text{var}(x_2) \end{bmatrix}$$

$\text{var}(x) = \frac{\sum (x_i - \bar{x})^2}{n-1}$

$\text{cov}(x, y) = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{n-1}$

+ symmetric
+ square.

+ Examples

Ridge → House price — square footage, bedroom, location.

Lasso → Car — Boat price, Bus price

Ridge → Exam marks

Gradient descent → rain prediction

~~Crude earth~~ Abnormal → crop yield.

Q.

Height x	Weight y	xy	x^2
50	15	750	2500
20	10	200	400
40	15	600	1600
30	12	360	900
50	15	750	2500
190	67	2660	7900

$y = bx + a$

$$b = \frac{n(\Sigma xy) - \Sigma x \Sigma y}{n \Sigma x^2 - (\Sigma x)^2}$$

$$= \frac{5(2660) - 190 \times 67}{5 \times 7900 - (190)^2}$$

$$= \frac{570}{3400} = 0.167$$

$a = \bar{y} - b\bar{x}$

$a = 13.4 - 0.167 \times 38$

$a = 7.054$

$y = ~~0~~\ 7.054 + 0.167x$

y	y_hat	$(yhat - y)^2$	$(y - \bar{y})^2$
5	6	+1	1.44
2	1	+1	3.24
4	5	1	0.04
3	4	1	0.64
5	7.	4	1.44.
19.		8.	6.8

$\bar{y} = \frac{19}{5} = 3.8$

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\Sigma(y_i - \hat{y})^2}{\Sigma(y_i - \bar{y})^2}$$

$$R^2 = 1 - \frac{8}{6.8}$$

$$= -0.176$$

Date
Page

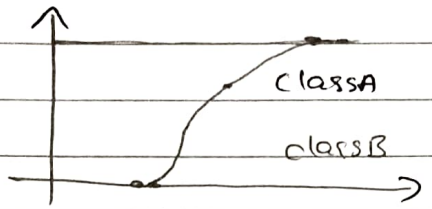
Linear Regression

1. A linear approach that models the relationship blw a DV & one or more IVs.
2. solved regression problems
3. estimates the DV when there is a change in the IV.
4. Output (DV) continuous in nature
5. Linear relationship
6. Uses a straight line
7. 1-D & 1-I → Simple Linear Reg.
   1-D & >1-I → Multiple Linear Reg
8. $y = \beta_1 + \beta_2 X + \varepsilon$ → error term
   ↓ intercept ↓ slope
9. 

10. Ex. a. predicting the GDP of a country.
    b. predicting product prize
    c. score prediction.

Logistic Regression

1. A statistical model that predicts the probability of an outcome that can only have two values
2. solve classification problems (binary class.)
3. calculates the possibility of an event occuring.
4. Output (DV) is Discrete
5. Goal is to find best fitting model for IV & DV relationship.

6. uses S curve or sigmoid function
7. also called Logit Regression
8. IV can be continuous or binary → (DV)
9. Output ⇒ 1 (true, success)
10. 

$y = \frac{1}{1+e^{-x}}$
11. Ex. a. predicting whether an email is spam or not
b. predicting whether customer will take loan or not
c. credit card transaction is fraud or not.

12. $\log\left(\frac{y}{1-y}\right) = b_0 + b_1x_1 + \cdots + b_nx_n$

Confusion Matrix

actual	165	NO	YES	
		50	10	60
	NO	[TN]	[FP]	
		5	100	105
	YES	[FN]	[TP]	
		65	110	

Predicted

$$\text{Accuracy} = \frac{TP + TN}{\text{Total}} = \frac{100+50}{165} = 0.91$$

$$\text{Error rate} = 1 - \text{accuracy} = 1 - 0.91 = 0.09$$

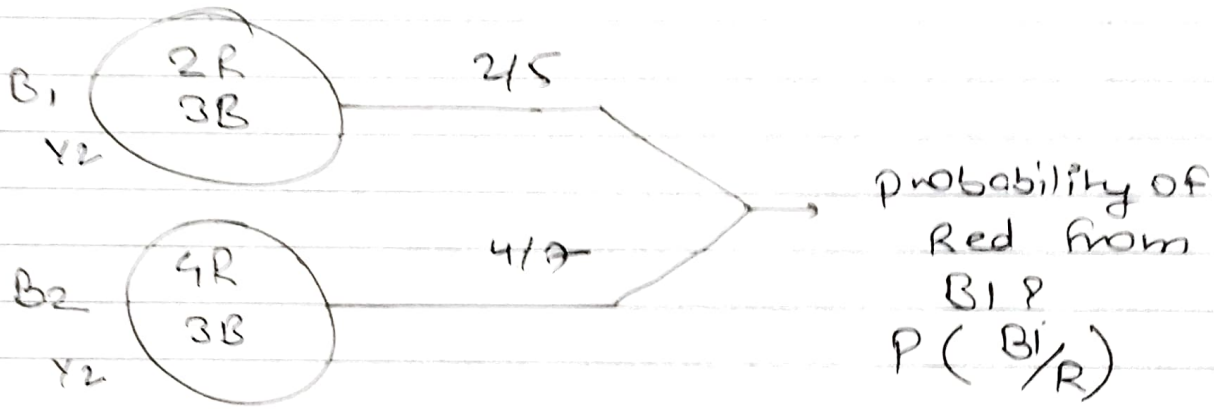
$$\text{Precision} = \frac{TP}{\text{Predicted Yes}} = \frac{100}{110} = 0.64$$

$$\text{Recall} = \frac{TP}{\text{actual yes}} = \frac{100}{105} = 0.95$$

Naive Bayes Classifier (Independent)

1. Supervised learning
2. classification
3. Based on Bayes Theorem

Q.



$$P(Y/X) = \frac{P(X/Y) * P(Y)}{P(X)}$$

$Y = B1$

$X = R$

$$P(R/B1) = 2/5$$

$$P(B1) = 1/2$$

$$P(R) = 1/2 \times 2/5 + 1/2 \times 4/7$$

$$P(B1/R) = \frac{2/5 \times 1/2}{1/2 \times 2/5 + 1/2 \times 4/7}$$

$$P(Y/X) = \frac{P(X/Y) * P(Y)}{P(X)}$$

$$P(Y/X_1, X_2, X_3 \cdots X_n) = \frac{P(X_1/Y) * P(X_2/Y) * \cdots P(X_n/Y) * P(Y)}{P(X_1) * P(X_2) \cdots P(X_n)}$$

Q. 1] Prior Probability

$P(\text{fever} = \text{Yes}) = 7/10$

$P(\text{fever} = \text{No}) = 3/10$

2] Conditional Probability

	fever	
	Yes	No
Covid	4/7	2/3
Flu	3/7	2/3

$P(\text{Yes} / \text{flu}, \text{covid}) = \frac{P(\text{flu}/\text{yes}) \times P(\text{covid}/\text{yes}) \times P(\text{yes})}{P(\text{covid}) \times P(\text{flu})}$

$P(\text{No} / \text{flu}, \text{covid}) = P(\text{flu}/\text{No}) * P(\text{covid}/\text{No}) \times P(\text{No})$

$P(\text{Yes}/\text{Covid}, \text{Flu}) = 3/7 \times 4/7 \times 7/10 = 12/70$

$P(\text{No}/\text{Covid}, \text{Flu}) = 2/3 \times 2/3 \times 3/10 = 4/30$

Q. X = { Chills = Y, Runny Nose = N, Headache = Mild, Fever = Y, Flu = ? }

$P(Y/test) = P(test/Y) \times P(Y)$

1) Prior Probability (flu)

P(Flu = Yes) = 5/8

P(Flu = No) = 3/8

2) Conditional Probability

Chills	Y	N
Flu (Y)	3/5	1/3
Flu (N)	1/5	1/3

	Flu Y	Flu N
Chills (Y)	3/5	1/3
Chills (N)	2/5	2/3

	Y	N
RN (Y)	4/5	1/3
RN (N)	1/5	2/3

		Y	N
H	(S)	2/5	1/3
	(M)	2/5	1/3
	(N)	1/5	1/3

		Y	N
Fever	(Y)	3/5	0/3
	(N)	0/5	3/3

Yes $\Rightarrow$ $(3/5 \times 1/5 \times 2/5 \times 1) \times 5/8 = 6/200 = 3/100 = 0.03$

No $\Rightarrow$ $1/3 \times 0 \Rightarrow 0$

Normalized $\Rightarrow$ $\frac{0.03}{0.03+0} = 1$

Q. $P(H=Yes) = 3/8$
$P(H=No) = 5/8$

		Happy Y	Happy N
①	Weather		
	Good	1/3	3/5
	Bad	2/3	2/5
②	Study		
	Pass	3/3	1/5
	Fail	0/3	4/5
③	Neighbor		
	Home	2/3	2/5
	Out	1/3	3/5

(a) H = Good, S = PASS, N = Out

Yes $\Rightarrow$ $1/3 \times 3/3 \times 1/3 \times 3/8 = 1/24 = 0.0416$

No $\Rightarrow$ $3/5 \times 1/5 \times 3/5 \times 5/8 = 9/200 = 0.045$

Yes $\Rightarrow$ $\frac{0.0416}{0.0416 + 0.045} = 0.48$

No $\Rightarrow$ $\frac{0.045}{0.045 + 0.0416} = 0.52$

(b) $1/3 \times 3/3 \times 2/8 \times 3/8 \Rightarrow 1/6 = 0.16$

$2/5 \times 1/5 \times 2/5 \times 5/8 \Rightarrow 4/200 \Rightarrow 0.02$

Yes $\Rightarrow$ $\frac{0.16}{0.16 + 0.02} = 0.89$

No $\Rightarrow$ 0.11

KNN (K-Nearest Neighbor)

NN classifiers are defined by their characteristic of classifying unlabled ex. by assigning them the class of most similar labeled exs.

+. CV applica (facial)
+ movie enjoys
+ Identify patterns in genetic data

Euclidean

$$dist(x,y) = \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$$

1) blalance blw overfitting & underfitting training data → bias – variance
2) K↑ → noisy data impact ↓ but small pattern i/o imp ✗ ignore

3) no. of records & difficulty
4) 3 to 10
5) √ of no. of train. ex.
6) 15 → 4

163, 63

Q.

162	61	√5	5
165	62	√5	5
167	63	4	5
170	66	√58 = 7.61	
165	63	√2	5
167	64	√17	
170	65	√53	
173	68	x	
165	63		
163	63	2.0	5

Metrics to evaluate the performance of the classification model.

1) Accuracy →
2) Precision
3) Recall
4) F1-score
5) AUC-ROC

Confusion matrix →

A confusion matrix is defined as the table that is often used to describe the performance of a classification model on a set of the test data for which the true values are known.

		AC. Values	
PV		P(1)	N(0)
	P(1)	TP	FP
	N(0)	FN	TN

TP → model predicted +ve & true
TN → ——— -ve & ———
Type1 FP → ——— +ve & False
Type2 FN → ——— -ve & False

		AV	
		P	N
PV	P	TP	FP
	N	FN	TN

① Accuracy

1] measures of how often classifier predicts correctly

2] ratio of correct P / Total P.

3] DS → Unbalanced ✗

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

② Precision

1) explained how many of the correctly predicted cases turn out to be positive

2] Case FP > FN

3] TP / TP + FP

$$P = \frac{TP}{TP + FP}$$

③ Recall (Sensitivity)

1] explains how many of the actual positive cases the model is able to predict correctly.

2] FN > FP

3] medical false ✗, True matter

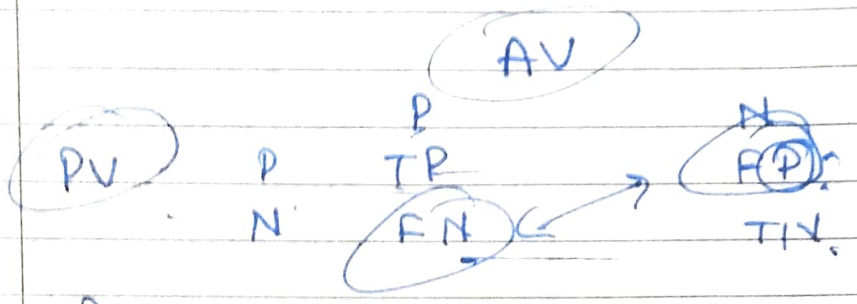
$$R = \frac{TP}{TP + FN}$$

④ F1 Score

combined idea, max P=R, HM of P & R

FP = FN, TN ?,

$$F1 = 2 \times \frac{P \times R}{P + R}$$



1] $A = \frac{TN + TP}{All}$

measures how model predict correctly.

2] $P = \frac{TP}{TP + FP}$

measures how many OF correctly predicted are positive. FP > FN

3] $R = \frac{TP}{TP + FN}$ FP < FN

measures how many of the actual true cases model predicts correctly

4] F1Score $2 \times \frac{P \times R}{P + R}$ FP = FN, TN > P

5] AOC-ROC curve (receiver operating characteristic) is

Showing the performance of a classi. model at all classi. threshold.

True +ve R => $\frac{TP}{TP + FN}$

FPR => $\frac{FP}{FP + TN}$

Q.

AV \ PV	P	N
P	TP = 3	FN = 8
N	FP = 2	TN = 87

TPR = TP

Actual Value

PV \	P	N
P	3	2
N	8	87

PV

AV \	P	N
P	3 = TP	8 = FN
N	2 = FP	87 = TN

$$TPR \Rightarrow \frac{TP}{TP+FN} = \frac{3}{3+8} = \frac{3}{11}$$

$$FPR \Rightarrow \frac{FP}{FP+TN} = \frac{2}{89}$$

Q.

AV

PV \	P	N
P	TP = 12	FP = 4
N	FN = 3	TN = 96

$$TPR = \frac{TP}{TP+FN} = \frac{12}{15} = \frac{4}{5}$$

$$FPR = \frac{FP}{FP+TN} = \frac{4}{20} = \frac{1}{5}$$

Cross Validation

1. used to asses how well our ML models perform on unseen data.
2. resampling tech. DS — 2 parts Test & Train
3. Train D → train model
   Test D → unseen test & Predict
4. TD → PV AT → ✓

Techniques

1. Hold out method
2. Leave one out cross-validation
3. K-fold cross validation
4. Stratified K-fold Cross validation.

① Hold Out

1. simplest to evaluate a classifier
2. DS
   Train Test
3. classifier performs function of assigning data items in a given collection to a target category
4. DS.

TS	rs

$$CV_{(k)} = \frac{1}{K} \sum_{i=1}^{K} MSE_i$$

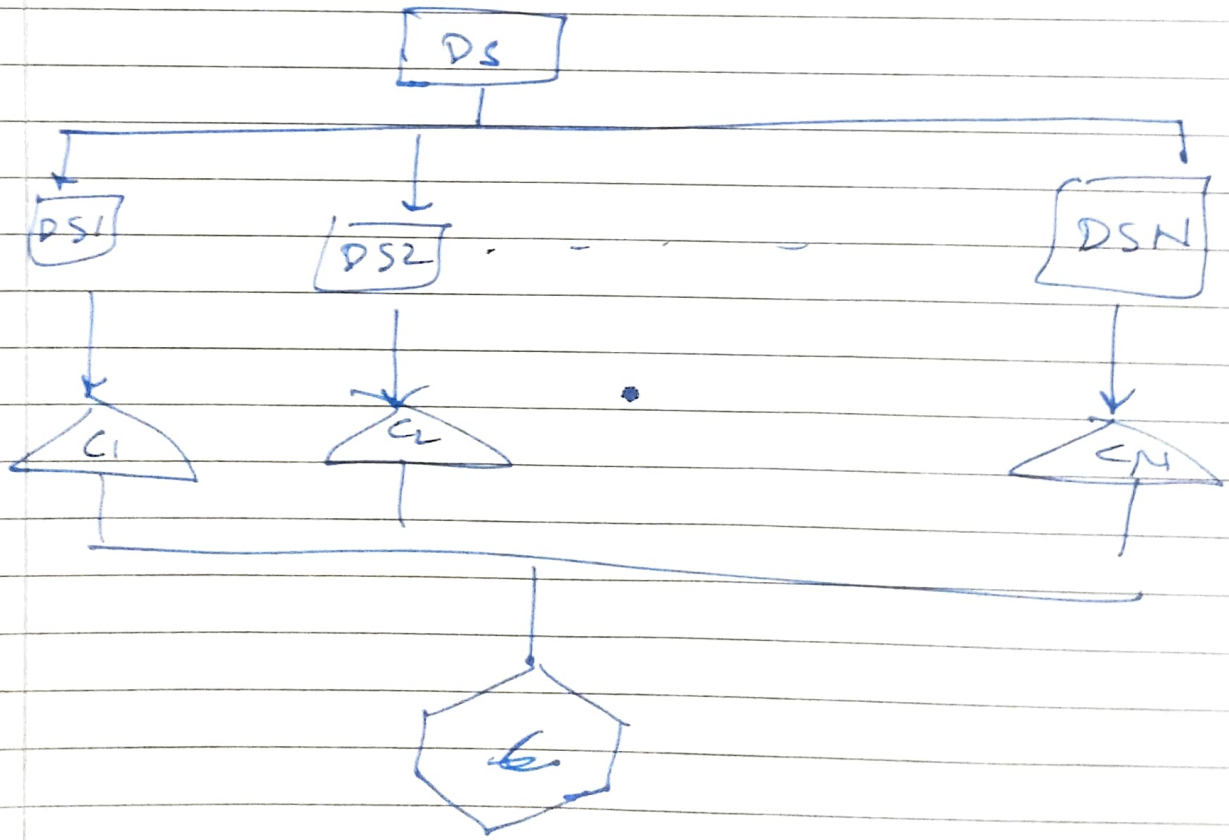
Weighted KNN

$$\hat{f}(x_q) \leftarrow \underset{v \in V}{\operatorname{argmax}} \sum_{i=1}^{k} \delta(v, f(x_i))$$

$$0.45\, d(M, M) + 0.15$$

Ensemble Learning (RF) (GB)

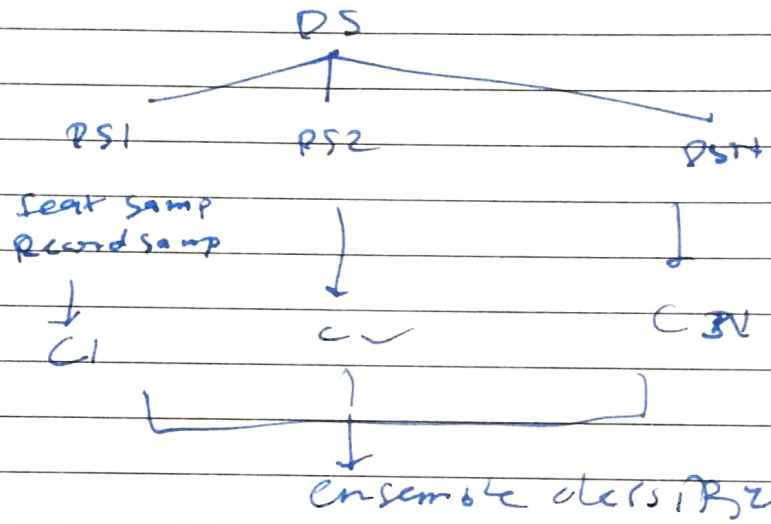
1. Improve ML results by combining several models.
2. better predictive performance compare to single model.
3. learn a set of classifiers (experts) and allow them to vote.



Types

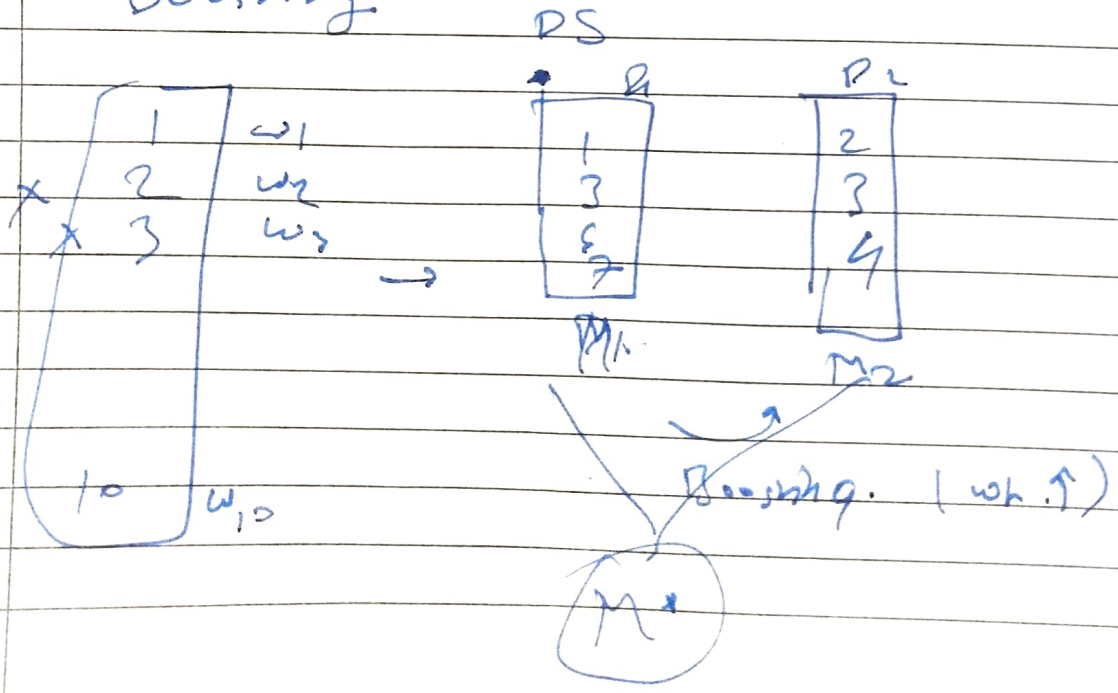
1. Bootstrap aggregating
2. Stacking
3. ~~Etc~~ Voting
4. Boosting

1) Bagging / Bootstrap Aggregating



classification → voting

2] Boosting



Ensemble Learning

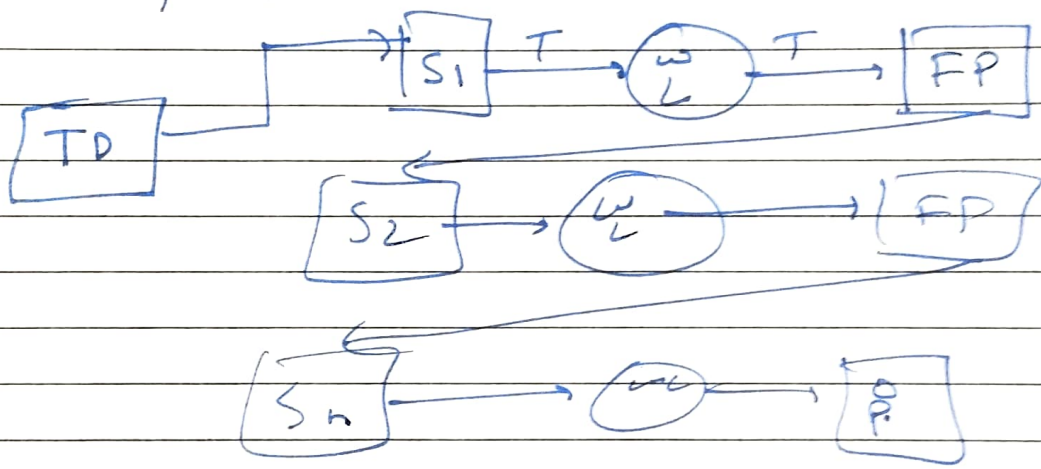
1. Type of ML uses multiple ML models to predict.
2. ex. RF, Gradient Boosting
3. Improve ML results by combining several models
4. Better result compare to m.
5. Individual model → weak learner coz ↑ variance or bias.
6. HB → not learning from data
7. HV → learning too well & varies for each data
8. Generalize ✗,
9. bias-variance trade off
   underfit → HBLV
   Overfit → LBHV
   ensemble balance karta hai BV to
10. final strong model

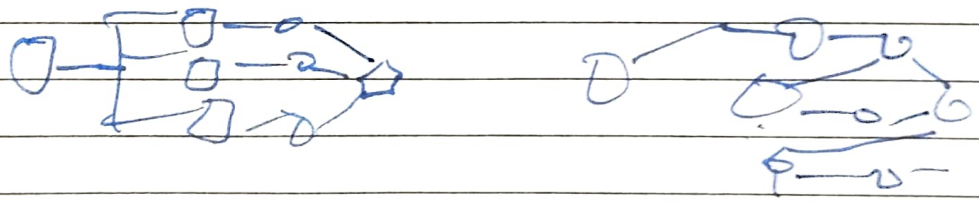
① Bagging

1. Reduce the variance from ~~pe~~ predictions by generating additional data for Training from DS
2. BA
3. Bootstrapping
   resampling subsets of data
4. Aggregating
5. Max voting

① Boosting

1. Sequentially training weak learners
2. Combining weak learners with HB.
3. Sample of data taken
4. Sample → 1 m → Pre
5. CP & IP
6. IP → used again 2M
7. at each step error improve
8. weighted averaging
9. M1 - w_1, M2 - w_2 by predictive power.





Decision Tree (Sup)

1. predict output of target variable
2. graphical presentation
3. all pssbl. solutions based on decisions on certain conditions.

Root Node
Decision Nodes
leaf Node / Terminal
Subtree
Pruning.

Q. A (outlook)
Values(outlook) = S, O, R

Entropy

1. Entropy is info theory metric that measures impurity or uncertainty in a grp. of observations.
2. DT to help to split data.
3. N class

$$E = -\sum_{i=1}^{N} p_i \log_2 p_i$$

4. p_i probability of randomly selecting an example in class i
5. categorical data.
6. E ↓ → ↑ homogeneity

$S = [9+, 5-]$

$$\text{Entropy}(S) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$$

$$= 0.94$$

$S_{\text{sunny}} \Rightarrow [2+, 3-] = 0.971$

$S_{\text{overcast}} \Rightarrow [4+, 0-] = 0$

$S_{\text{rain}} [3+, 2-] = 0.971$

Information Gain

1. measures how much info. a feature provided about a class
2. order of attribute in the nodes of DT
3. Gain = Eparent − Echildren

$$\text{Gain}(S, \text{outlook}) = \text{Entropy}(S) - \sum_{v \in (S,O,R)} \frac{|S_v|}{|S|} \text{Entropy}(S_v)$$

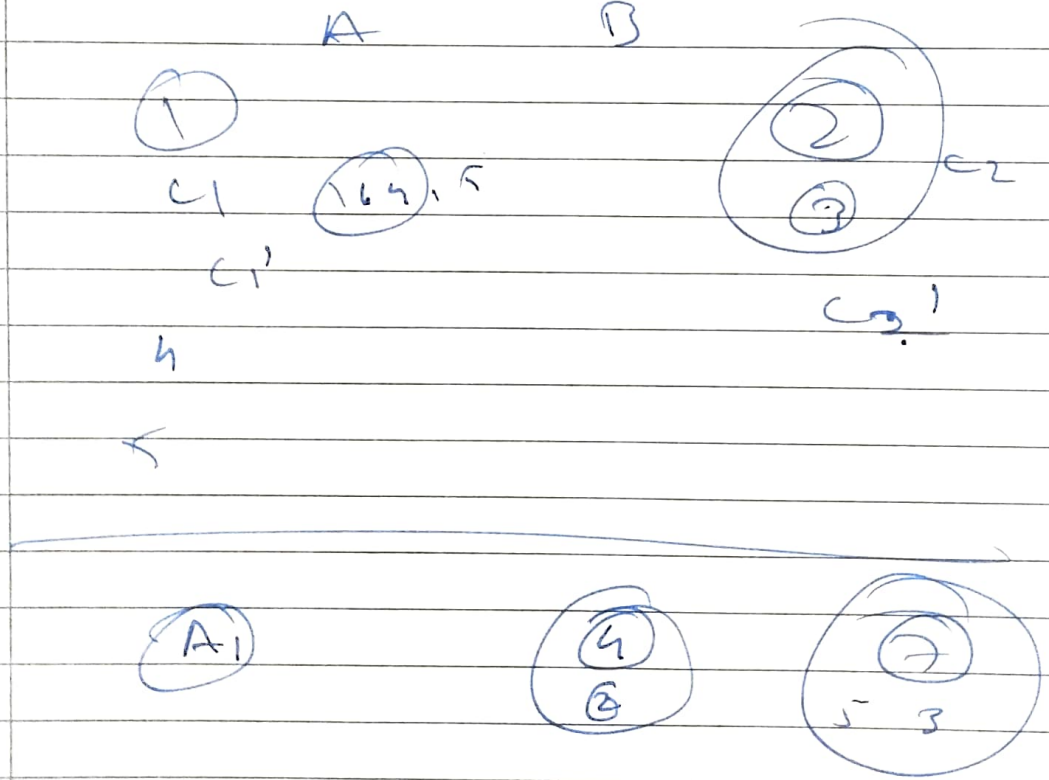
$$\text{Gain}(S, \text{outlook}) = E(S) - \frac{5}{14} \times 0.971 - \frac{4}{14} \times 0 - \frac{5}{14} \times 0.971$$

$$= 0.2464$$

RF

1. Class & Req
2. T ↓
3. A ↑
4. Mis ✓
5. H. Requ.
6. Size ↑
7. B. Box

K-Means (US)



from SKL. metrics import accuracy-score

y_true = [0, 1, 1]

y_pre = [0, 0, 1]

acc = acc-sco ( )