$$ZZ^T = V \operatorname{diag}\left(\frac{\sigma_i^2}{\sigma_i^2+\lambda I}\right) \underbrace{V^T V}_{I} \operatorname{diag}\left(\frac{\sigma_i^2}{\sigma_i^2+\lambda I}\right) V^T$$

$$\boxed{ZZ^T = V \operatorname{diag}\left(\frac{\sigma_i^2}{(\sigma_i^2+\lambda I)^2}\right) V^T}$$

* Ensemble Methods: Bagging & Boosting
  - Bagging: Parallel
  - Boosting: Sequential

* → Bagging: - Randomforest
  - Use multiple classifier

* Bootstrapping: Sampling technique where sample are derived from whole population using replacement procedure.

* Aggregation: in bagging is done to incorporate all possible outcomes of prediction & randomize the outcomes.

- Reduce the variance.

* → Boosting: Combine weak learning to form strong learner.
  * AdaBoost, Gradient boost, (XGBoost)

- Misclassified points are more important.

AdaBoost given Weighted Mode $(M_1, M_2 \dots M_n)$

$\hookrightarrow \sum_{i=1}^{n} w_i M_i$

→ use decision Stumps

* XGBoost
  - Use with AI learning.

Class imbalance

- Means # no of Points in some class is higher than other classes
  e.g. credit card fraud detection.
- bias towards majority class
- Acc. may not good performance measure for classifier for imbalance data

Soln
- Random Sampling
  Adv: simple to implement
  - Efficient
  dis: throws away lot of data
  - not represent original data

- Random oversampling:
  - dis: overfitting minority class.
  Adv: - no loss data
  - Empirically seem perform better than understanding

- SMOTE: Synthetic Minority Over sampling Technique.
  - Avoid overfitting due to exact replicas of minority class.
  - Subset of minority class is taken.

dis - No good for data is in higher dimension.
- may increase noise

Adv: Reduce a chance of overfitting.

- Other technique: Penalization based model.
  - Class weight based model.
  - Key to solve diff. evaluation measure
  - Ensemble methods.

AdaBoost :-

Setting :- data : $\{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$

$y_i \in \{-1, +1\}$ → binary class problem

weight → $w_i = \frac{1}{n} \quad \forall i$

⊙ learner which can work with weighted data points to give

$h(x) : \{-1, +1\}$

→ predicted o/p

Adaboost :- (1) initialize $\{w_i\}$ as $w_i^{(1)} = \frac{1}{n}$

for $i = 1 \dots n$

(2) for $t = 1 \dots T$ → this how many learner we are actually apply.

for

(a) fit $h_t(x)$ to training data by minimizing

$$J_t = \sum_{i=1}^{n} w_i^{(t)} I(h_t(x) \neq y_i) \quad \text{— (A)}$$

loss of the misclassify points.

0/1 loss

I is indicator which give o/p 0/1 for class labels.

(b) find

$$\epsilon_t = \frac{\sum_{i=1}^{n} w_i^{(t)} I(h_t(x) \neq y_i)}{\sum_{i=1}^{n} w_i^{(t)}}$$

fraction of miss classify points

set $\alpha_t = \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$ — (B)

(c) Update weight

$$w_i^{(t+1)} = w_i^{(t)} \exp\left(\alpha_t I(h_t(x_i) \neq y_i)\right) \quad \text{— (C)}$$

this means if correctly classify points then $I(h_t(x_i) \neq y_i)$ is zero then we did not change the weight. when it is incorrectly classify then weight will increase.

3). Final Model

$$H_T(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

→ why does it work

- Boosting as sequential min^n of exponential error fun^n.

$$E = \sum_{i=1}^{n} \exp(-y_i f_t(x_i))$$

where $f_t(x) = \frac{1}{2} \sum_{i=1}^{t=t} \alpha_t h_t(x)$

Goal: Minimize E w.r.t both α_t and $h_t(x)$ (parameters)

Assume $h_1(x) \dots h_{t-1}(x)$ & $\alpha_1 \dots \alpha_{t-1}$ all fixed minimize only w.r.t. $\alpha_t, h_t(x)$ $\left\{ \because f_t(x) = f_{t-1}(x) + f_t(x) \right\}$

$$\therefore E = \sum_{i=1}^{n} \exp\left\{ \underbrace{-y_i f_{t-1}(x_i)}_{\text{Constants}} - \frac{1}{2} y_i \alpha_t h_t(x) \right\}$$

$(e^{-a-b} = e^{-a} \cdot e^{-b})$ ← this Const.

$$= \sum_{i=1}^{n} w_i^{(t)} \exp\left(-\frac{1}{2} y_i \alpha_t h_t(x_i)\right)$$

$\left(\because w_i^{(t)} = \exp(-y_i f_{t-1}(x_i)) \right)$

- ~~Consider~~ Constants can be consider Constants.

let $C_t =$

* Consider a error

$$E = \sum_{i=1}^{n} \exp\left(-y_i f_t(x_i)\right)$$

, where $f_t(x) = \frac{1}{2} \sum_{l=1}^{t} \alpha_l h_l(x)$

: this is.

$$E = \sum_{i=1}^{n} w_i^{(t)} \exp\left(-\frac{1}{2} y_i \alpha_t h_t(x_i)\right)$$

$$\left(\because w_i^{(t)} = \exp\left[y_i f_{t-1}(x_i)\right] \right)$$

C_t = data point correctly classified points

M_t = " " incorrectly " ".

$$E = e^{-\alpha_t/2} \sum_{i \in C_t} w_i^{(t)} + e^{\alpha_{t/2}} \sum_{i \in M_t} w_i^{(t)}$$

$$E = \left(e^{\alpha_t/2} - e^{-\alpha_t/2}\right) \sum_{i=1}^{n} w_i^{(t)} I\left(h_t(x_i) \neq y_i\right) + e^{-\alpha_t/2} \sum_{i=1}^{n} w_i^{(t)}$$

(A)

Which I for misclassified points & 0 = for correctly class. points

⇒ Minimize w.r.t. $h_t(x)$ Second term is Constant Equivalent to minimizing (A).

~~⇒ Similarly Mi~~

⇒ Similarly Minimize w.r.t. α_t we get (B)

$$\frac{\partial E}{\partial \alpha_t} = e^{\alpha_t/2}\left(\frac{1}{2}\right) - e^{-\alpha_t/2} \cdot \left(\frac{1}{2}\right) + e^{-\alpha_t/2} \cdot \left(\frac{1}{2}\right)$$

$$= e^{\alpha_t/2}\left(\frac{1}{2}\right)$$

$$0 = \sum_{i:1}^{n} w_i^{(t)} I(h_t(x_i) \neq y_i) e^{\alpha_t/2} \cdot \left(\frac{1}{2}\right)$$

$$- \sum_{i:1}^{n} w_i^{(t)} I(h_t(x_i) \neq y) \cdot \left(\frac{1}{2}\right) \cdot e^{-\alpha_t/2}$$

$$+ \sum_{i:1}^{n} w_i^{(t)} e^{-\alpha_t/2} \cdot \left(\frac{1}{2}\right)$$

$$0 = \sum_{i:1}^{n} w_i^{(t)} I(h_t(x_i) \neq y) \cdot e^{\alpha_t/2} + \sum_{i:1}^{n} w_i^{(t)} \cdot$$

$$+ \left(\sum_{i:1}^{n} w_i^{t} I(h_t(x_i) \neq y) \; e^{-\alpha_t/2} + \sum_{i:1}^{n} w_i^{(t)} \right)$$

$$e^{+\alpha_t/2}$$

$$0 = \sum_{i:1}^{n} w_i^{(t)} I(h_t(x_i) \neq y) \left(e^{\alpha_t/2}\right)^2 +$$

$$+ \sum_{i:1}^{n} w_i^{t} I(h_t(x_i) \neq y) + \sum_{i:1}^{n} w_i^{(t)}$$

$$\left(e^{\alpha_t/2}\right)^2 = -\left(\sum_{i:1}^{n} w_i^{(t)} I(h_t(x_i) \neq y) + \sum_{i:1}^{n} w_i^{(t)} \right)$$

Now Khushi
↓ derivative hu

$$e^{\alpha_t + 1} = \frac{1}{\epsilon_t}$$

$$e^{\alpha_t} = \frac{1 - \epsilon_t}{\epsilon_t}$$

$$\boxed{\alpha_t = \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)}$$

Now $w_i^{(t+1)} = w_i^{(t)} \exp\left\{-\frac{1}{2} y_i \alpha_t h_t(x_i)\right\}$ put this it here

Also $y_i h_t(x_i) = 1 - 2 I(h_t(x_i) \neq y_i)$

$$= w_i^{(t)} \exp\left(-\frac{1}{2}\alpha_t (1 - 2 I(h_t(x_i) \neq y_i))\right)$$

$\Rightarrow$ in next iteration this term is independent from the data point then ignore it

$$w_i^{(t+1)} = \alpha w_i^{(t)} \exp(-\alpha_t/2) \Rightarrow w_i^{(t)} \exp(\alpha_t)$$

$$+ w_i^{(t)} \exp(\alpha_t I(h_t(x_i) \neq y_i))$$

$$\boxed{w_i^{(t+1)} = w_i^{(t)} \exp(\alpha_t I(h_t(x_i) \neq y_i))}$$

this is (C)

GMM :- Gaussian dist^n by itself may not enough in modeling real datasets.

→ sometime it is as superposition of two or more Gaussians may compute data better.

- Such superposition can be formulated as mixture dist^n (this is why it is in unsupervised learning.

Mean combine two dist^n