Lecture 6/25

- HW3 demo
- next: $\boxed{\text{chapter 4}}$
  - boosting / ensemble
  - Multilabel data, in particular ECOC
  - Feature selection
  - Feature aggregation / PCA
  - Features from similarity / TSNE
  - HW4
- chapter 5: SVM + kernels
- chapter 6: advanced NN
  - CNN
  - RecNN vs Transformers

"random Forrests"

- Active learning.

$u_t(x)$ = score produced by t^{th} weak learner (not probab, not class prediction)

Boosting (weak learner) = additive ensemble

h_t

$$\boxed{F(x)} = \sum_{t=1}^{T} h_t(x)$$

final score

weak scores

$h(x)$

weak learner

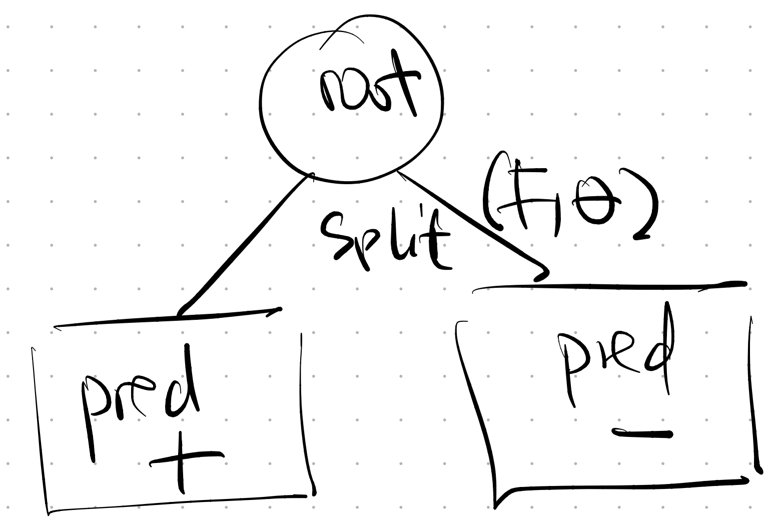
acc $\simeq$ 65%

better than random

but poor

default: shallow DecTree

dec stump: 1-split DT



datapoint

$X = (x^1 x^2 \dots x^D)$ $t = 1:T$

x_i

- Regression Mode label y = quantity

want $(F(x) - y)^2 \simeq 0$

all i $\sum_{i=1}^{N} (F(x_i) - y_i)^2 \simeq 0$

$y^* = \frac{1+y}{2}$

- Classification mode $y \in \{+1, -1\}$ $y^* \in \{0, 1\}$

$F(x)$ score $\xrightarrow{\text{map}}$ $\begin{cases} pr(y=1|x) \\ p(y=-1|x) = 1 - p(y=1|x) \end{cases}$

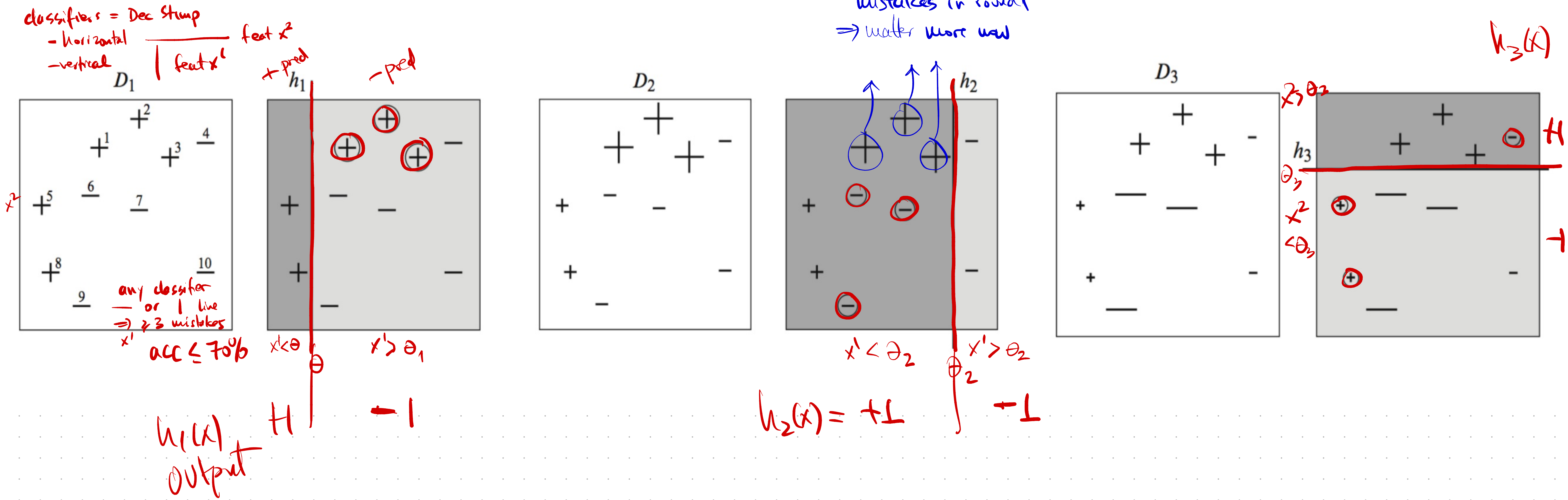
want max likelihood $\prod_{i=1}^{N} p(y^*|x)^{y^*} \cdot p(y^*=0|x)^{1-y^*}$

filter which probab

intuition cca 1992: F(x) cannot be much better than h(x) weak learners.
VERY WRONG

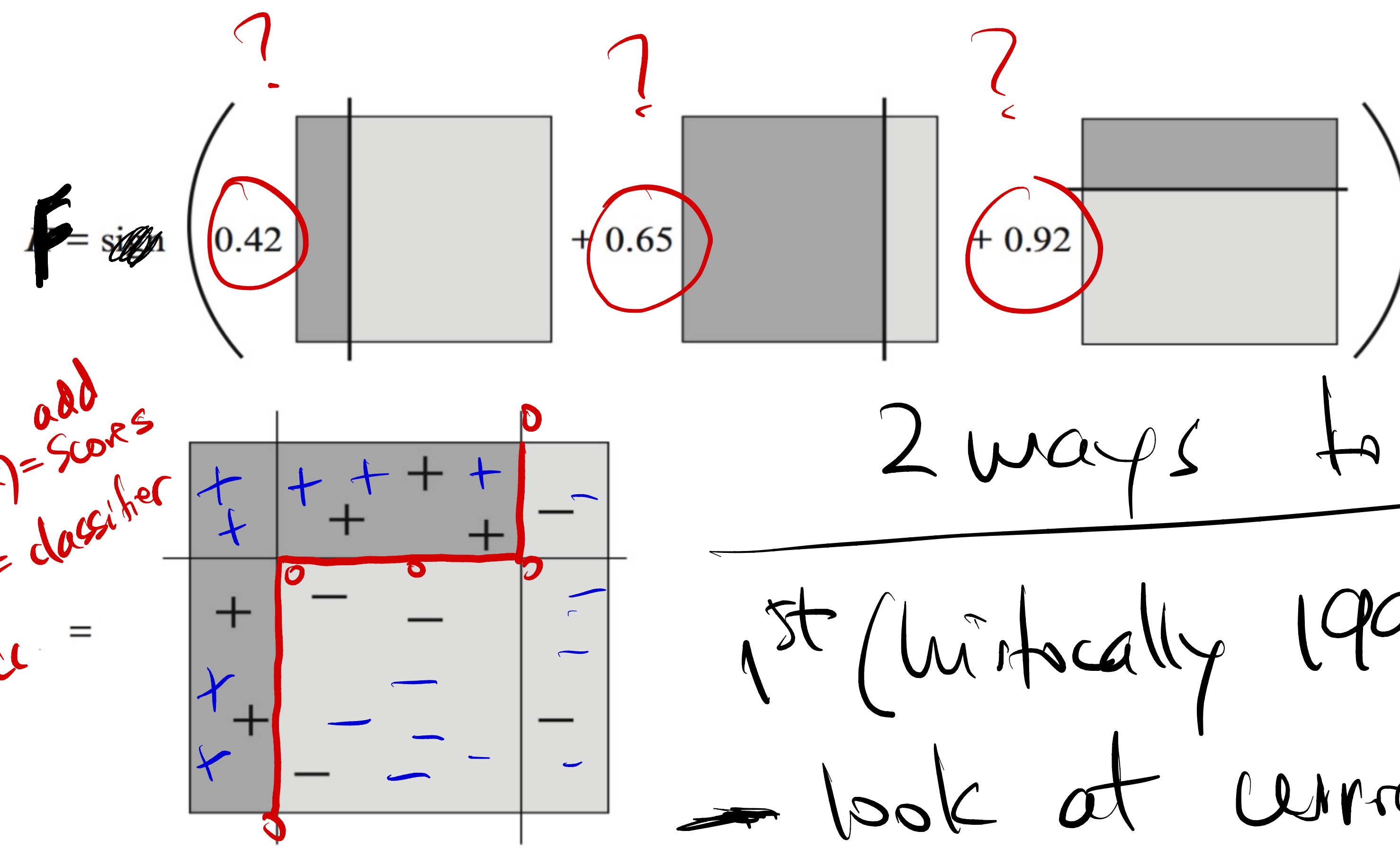
Gradient Boosting = Gradient Descent + Boosting

Adaboost



$$F(x) = \sum_t \rho_t h_t(x)$$

fixed coefficients ρ_t for $h_t()$



add scores $F(x) =$

$\text{sgn}(F(x)) =$ classifier

100% acc

2 ways to implement boosting

1st (historically 1996): ADABOOST

- look at current w.learner $h_t(x)$
- reweight data $D_{t+1}(x_i) =$

  = importance of datapoint x_i next round
- train $h_{t+1}(\,) =$ weak learner

with error weighted by $D_{t+1}()$ distribution

$$e_{t+1} = \sum_{i=1}^{N} 1_{[h_{t+1}(x_i) \neq y_i]} \cdot D_{t+1}(x_i)$$

$\overbrace{acc \begin{cases} 1 & \text{mistake} \\ 0 & \text{correct} \end{cases}}$ weight for round t+1

- add $h_{t+1}()$ to $F(x)$

$$F(x) = F(x) + h_{t+1}(x)$$

$$F(x) = h_1(x) + h_2(x) \cdots + h_{t+1}(x)$$

Key: After $t = 1:T$ rounds of AdaBoost

weight on X $\quad D_{T+1}(x) \simeq$ # rounds t where X mistake $h_t(x) \neq y$

$$\simeq \exp\left(\cdot \sum_{t=1}^{T} \alpha_t \left[\boxed{h_t(x) \cdot -y} \right] \right)$$

correct $-1: h_t(x) = y$

$+1: h_t(x) \neq y$ incorrect

- $D_{T+1}(x)$ small $\Rightarrow$ most classif $h_t(x) = y$

$D_{T+1}(x) = \text{large} \Rightarrow$ next classifier $h_{t+1}(x)$ focus on datapoint x

- $D_{T+1}(x) = \text{large} \Rightarrow$ most classif $h_t(x) \neq y$

(Th) proper weights, h() training etc (see Adaboost details)
training can finish in one of those 2 ways.

- either train error $\simeq 0$ $\quad \sum_{i=1}^{N} 1_{[F(x_i) \neq y_i]} \longrightarrow 0$
  F(x) ensemble

- or stuck : $D_{T+1}()$ dist of point importance for next round is s.t. the best $h_{T+1}(x)$ classifier is random

$$e_{T+1} = \sum_{i=1}^{N} 1_{[h_{T+1}(x) \neq y_i]} \cdot D_{T+1}(x_i) \simeq 0.5$$

as long as $h_{T+1}(x)$ makes some progress (better than 0.5)
training error (F(x)) has to go to zero.

2nd way

much better

$F(x) = h_1(x) + h_2(x) + \cdots + h_T(x)$

Gradient Boosting.

want $F(x) \simeq y$ (regression mode)

"house prices"

next weak learner

$h_{T+1}(x) = F_{new}(x) - F_{current}(x)$

error $\frac{1}{2}(F(x) - y)^2 = J(x)$

$\boxed{\text{diff w.r.t } F(x)} = F(x) - y = \frac{\partial J}{\partial F()}$

?

Grad descent mode for $F(x)$ as variable ?? simple $\boxed{\lambda = 1}$

$F_{new}(x) = F(x) - \lambda \cdot \frac{\partial J}{\partial F()} = F(x) + y - F(x)$

trivial? want $F_{new}(x) = y$ (wrong thinking)

GD update : $F_{new}(x) = F(x) + \underbrace{h_{T+1}(x)}_{\simeq\, y - F(x)}$

want the next weak learner

$$h_{T+1}(x) \simeq y - F(x)$$

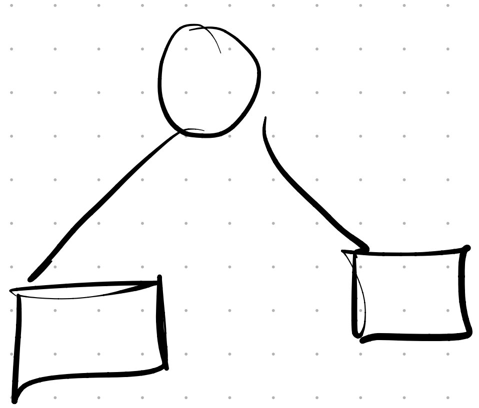
- if $\boxed{\underbrace{h_{T+1}(x)}_{\text{next w. learner}} \simeq \underbrace{y - F(x)}_{\text{residual label}}}$ well $\Rightarrow$

  $F_{new}(x) = F(x) + h_{T+1}(x) \underset{\text{well}}{\simeq} y$ (good)

- practice: train $h_{T+1}()$ to match $h_{T+1}(x) \simeq \boxed{y - F(x)}$ new label

  weak learner

  "dec stump"



- req updating label after every round $y_{new} = y_{orig} - F(x)$

- does not require weights like AdaBoost, modify train procedure

Train $h_{T+1}(x)$: DecTreeAlg$(X, y - F(x))$

Classification $y \in \{+1, -1\}$ $y^* = \frac{1+y}{2} \in \{1, 0\}$ categories

$$F(x) = \sum_{t=1}^{T} h_t(x)$$

Score

$$p(x) = p(y^* = 1 \mid x) = \frac{e^{F(x)}}{e^{F(x)} + e^{-F(x)}}$$

$$p(\bar{x}) = p(y^* = 0 \mid x) = \frac{e^{-F(x)}}{e^{F(x)} + e^{-F(x)}} = 1 - p(x)$$

filter $Y^* \begin{smallmatrix} \nearrow 0 \\ \searrow 1 \end{smallmatrix}$ which probas.

$$\text{log likelihood} = \log\left(\prod_{i=1}^{N} p(x)^{y^*} (1 - p(x))^{1-y^*} \right)$$

want MAX