Boosting by Well-Designed Ensemble

Geometrical View of Ensemble Learning

Noboru Murata

June 21, 2023

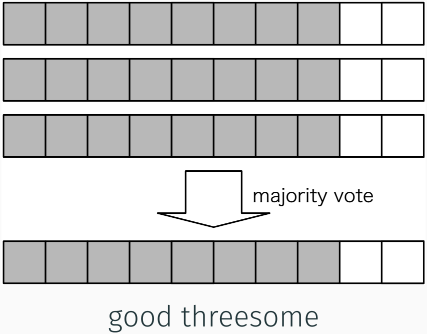
https://noboru-murata.github.io/

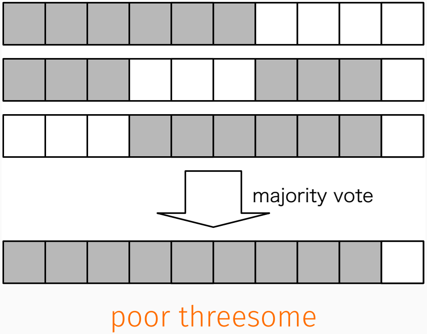
Introduction

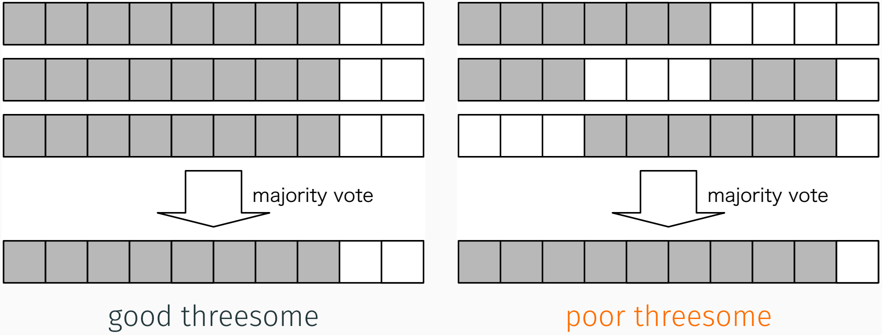
- consider participating a quiz show where threesome teams compete in answering various genre questions
  (10 genres such as history, politics, entertainment, sports)

- consider participating a quiz show where threesome teams compete in answering various genre questions
  (10 genres such as history, politics, entertainment, sports)
    - good threesome
- poor threesome

- consider participating a quiz show where threesome teams compete in answering various genre questions
  (10 genres such as history, politics, entertainment, sports)
    - good threesome
        - each member can answer 8 genres
        - all the members are weak in entertainment and sports
        - stereo-typed good members
- poor threesome
    - each member can answer 6 genres
    - all the member are weak in different genres
    - poor but varied members


majority vote
good threesome


majority vote
poor threesome



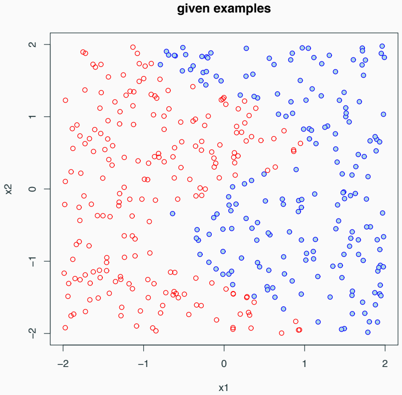
essence of ensemble learning

- collect as varied individuals as possible
- each individual does better than random guess

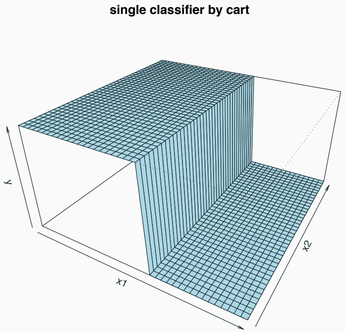
(Freund 1995; Freund and Schapire 1997)

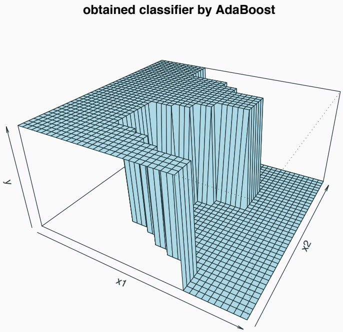
classification problem:

- predict label $y \in \mathcal{Y}$ from corresponding features $\boldsymbol{x} \in \mathcal{X}$
- construct a classifier $h(\boldsymbol{x}) = \hat{y}$ from finite samples



obtained classifier

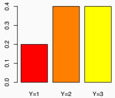




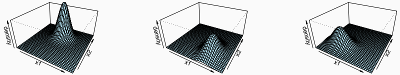
without boosting

with boosting

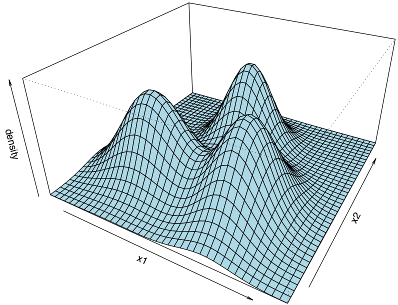
- select a Gaussian subject to categorical distribution

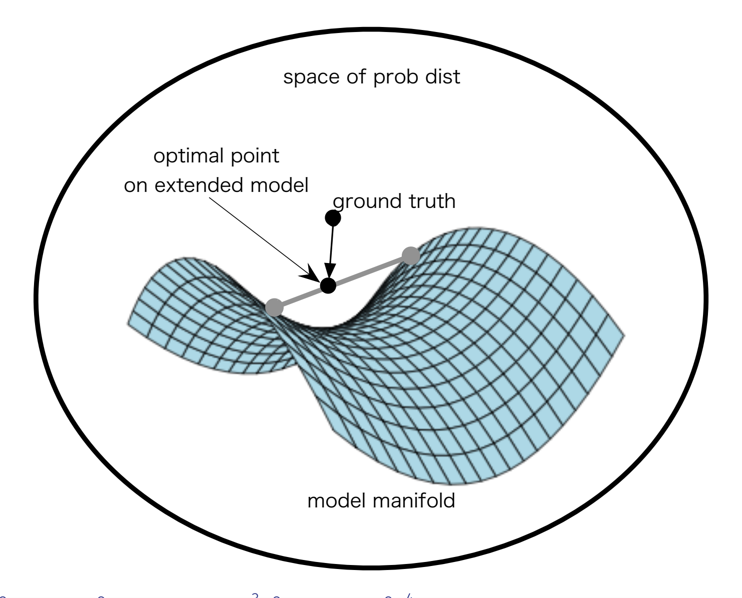


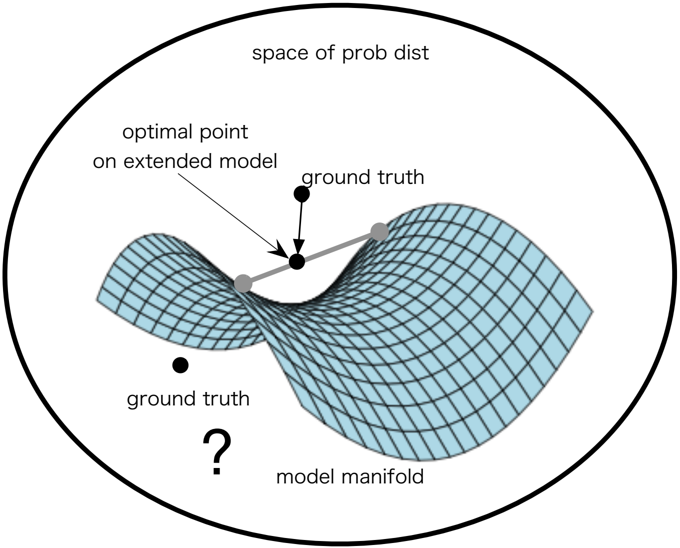
- generate a sample from a selected Gaussian

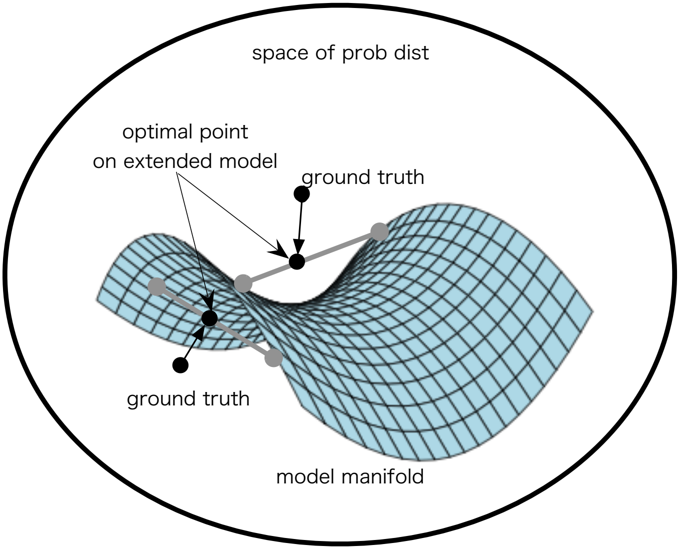


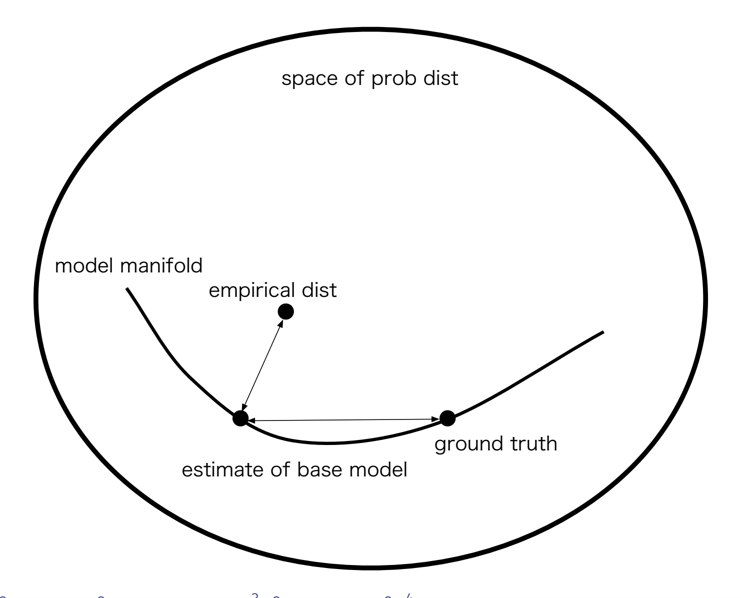
- total distribution is not a Gaussian

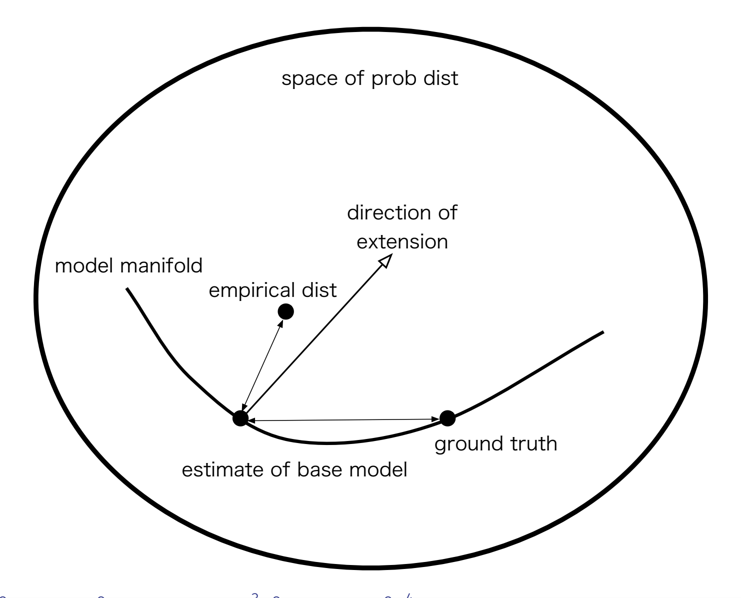


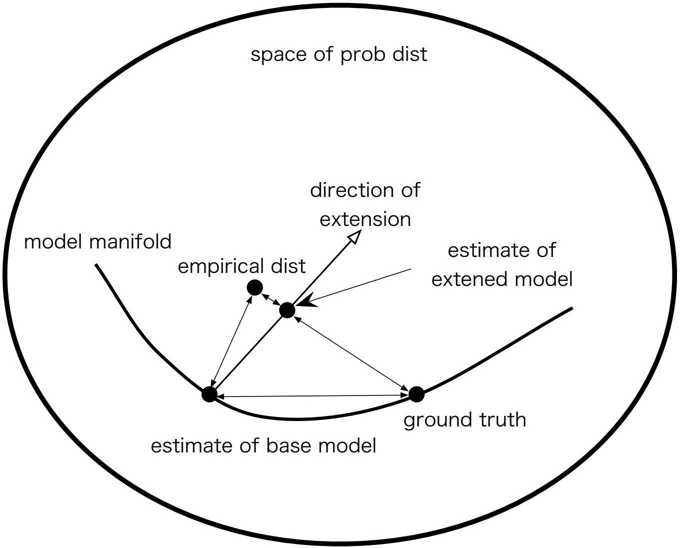

space of prob dist
optimal point
on extended model
ground truth
model manifold


space of prob dist
optimal point
on extended model
ground truth
ground truth
?
model manifold


space of prob dist
optimal point
on extended model
ground truth
ground truth
model manifold


space of prob dist
model manifold
empirical dist
ground truth
estimate of base model


space of prob dist
direction of extension
model manifold
empirical dist
ground truth
estimate of base model


space of prob dist
direction of extension
model manifold
empirical dist
estimate of extened model
ground truth
estimate of base model

Problem Formulation

- problem
  - predict labels $y \in \mathcal{Y}$ from given features $\mathbf{x} \in \mathcal{X}$
- notation
  - classifier: set-valued function h

$$h : \mathbf{x} \in \mathcal{X} \mapsto \mathcal{C} \subset \mathcal{Y}$$

  - decision function: another representation of classifier

$$f(\mathbf{x}, y) = \begin{cases} 1, & \text{if } y \in h(\mathbf{x}), \\ 0, & \text{otherwise}, \end{cases}$$

  - majority vote: linear combination of multiple classifiers

$$H(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t f_t(\mathbf{x}, y)$$

(start)

- input:
  n samples\; $\{(\boldsymbol{x}_i, y_i); \boldsymbol{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1, \ldots, n\}$,
  increasing convex function U.
- initialize:
  distribution $D_1(i, y) = 1/n(|\mathcal{Y}| - 1)$ $(i = 1, \ldots, n)$,
  combined decision function $F_0(\boldsymbol{x}, y) = 0$.
- repeat: repeat following steps $(t = 1, \ldots, T)$.

(iteration)

- step 1: select a decision function f (classifier h) which (approximately) minimizes with a distribution D_t:

$$\epsilon_t(f) = \sum_{i=1}^{n} \sum_{y \neq y_i} \frac{f(\mathbf{x}_i, y) - f(\mathbf{x}_i, y_i) + 1}{2} D_t(i, y)$$

$$f_t(\mathbf{x}, y) = \arg\min_{f \in \mathcal{F}} \epsilon_t(f).$$

(iteration)

- step 2: calculate reliability α_t:

$$\alpha_t = \arg\min_{\alpha} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} U\Big(F_{t-1}(\mathbf{x}_i, y) + \alpha f_t(\mathbf{x}_i, y) - F_{t-1}(\mathbf{x}_i, y_i) - \alpha f_t(\mathbf{x}_i, y_i)\Big).$$

(iteration)

- step 3: update the combined decision function F_t and the distribution D_t:

$$F_t(\mathbf{x}, y) = F_{t-1}(\mathbf{x}, y) + \alpha_t f_t(\mathbf{x}, y),$$

$$D_{t+1}(i, y) \propto U'\left(F_t(\mathbf{x}_i, y) - F_t(\mathbf{x}_i, y_i)\right),$$

$$\text{where } \sum_{i=1}^{n} \sum_{y \neq y_i} D_{t+1}(i, y) = 1.$$

(end)

- output:

  construct a majority vote classifier:

$$H(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} F_T(\mathbf{x}, y)$$
$$= \arg\max_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t f_t(\mathbf{x}, y).$$

special case of boosting algorithm:

- $U(z) = \exp(z)$ (following steps are simplified)
  - step 2:

$$\alpha_t = \frac{1}{2}\log\frac{1-\epsilon_t(f_t)}{\epsilon_t(f_t)},$$

  - step 3:

$$D_{t+1}(i,y) \propto \exp\{F_t(\mathbf{x}_i,y) - F_t(\mathbf{x}_i,y_i)\}$$

(Freund and Schapire 1997)

(start)

- input:
  n samples\; $\{(\boldsymbol{x}_i, y_i); \boldsymbol{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1, \dots, n\}$,
  increasing convex function U.
- initialize:
  $q_0(y|\boldsymbol{x})$ (set $\xi(q_0) = 0$ for simplicity, where $\xi = (U')^{-1}$)
- repeat: repeat following steps ($t = 1, \dots, T$).

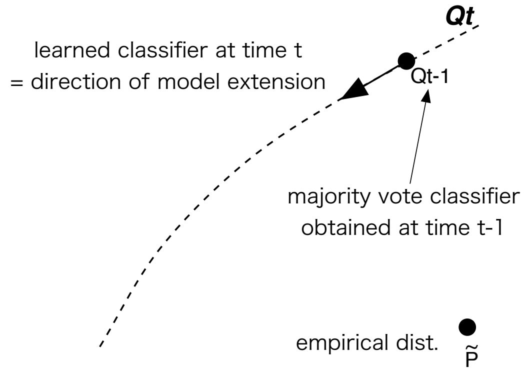
(iteration)

- step 1: select decision function f_t (classifier h_t) such that $f - b'$ and $q_{t-1} - \tilde{p}$ should direct as similar as possible:

$$f_t(\mathbf{x}, y) = \arg\max_{f \in \mathcal{F}} \langle q_{t-1} - \tilde{p}, f - b' \rangle_{\tilde{\mu}}$$

where

$$q = u\Big(\xi(q_{t-1}) + \alpha f - b(\alpha)\Big), \quad u = U'.$$


Qt
learned classifier at time t
= direction of model extension
Qt-1
majority vote classifier
obtained at time t-1
empirical dist.
P̃

(iteration)

- step 2: with one dimensional model

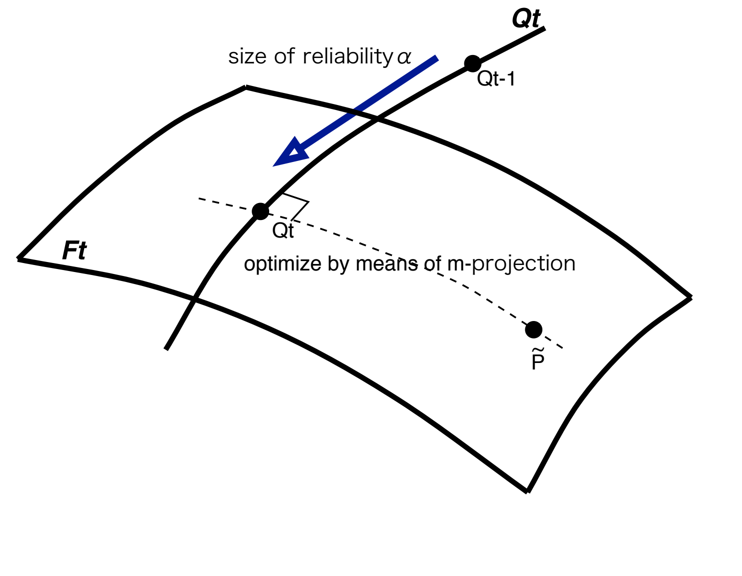
$$\mathcal{Q}_t = \left\{ q \,\middle|\, \xi(q) = \xi(q_{t-1}) + \alpha f_t - b_t(\alpha),\ \alpha \in R \right\}$$

construct orthogonal foliation $\{\mathcal{T}(q); q \in \mathcal{Q}_t\}$ as

$$\mathcal{T}(q) = \{p \in \mathcal{P} | \langle p - q, f_t - b' \rangle_{\tilde{\mu}} = 0\},$$

then find α_t with a leaf of the empirical distribution $\tilde{p}$ and model $\mathcal{Q}_t$:

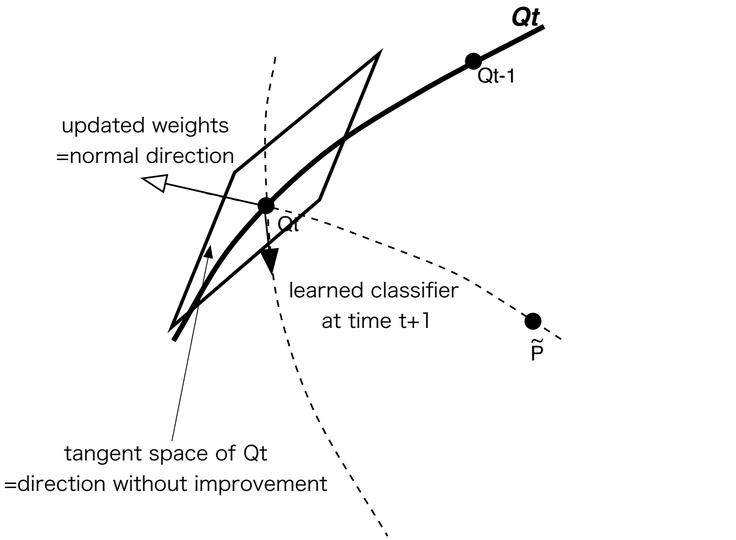
$$\alpha_t = \arg\min_{q \in \mathcal{Q}_t} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U\big(\xi(q(y|\mathbf{x}_i))\big) - \xi(q(y_i|\mathbf{x}_i)) \right].$$

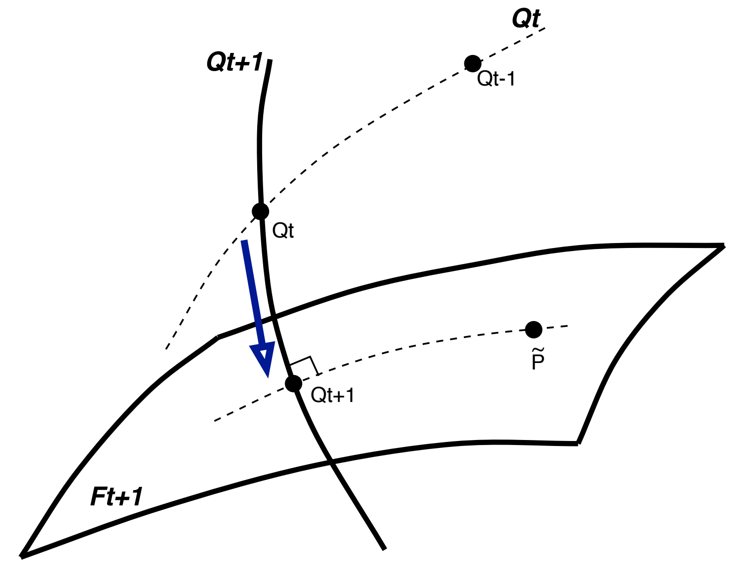

Qt
size of reliability α
Qt-1
Qt
Ft
optimize by means of m-projection
P̃

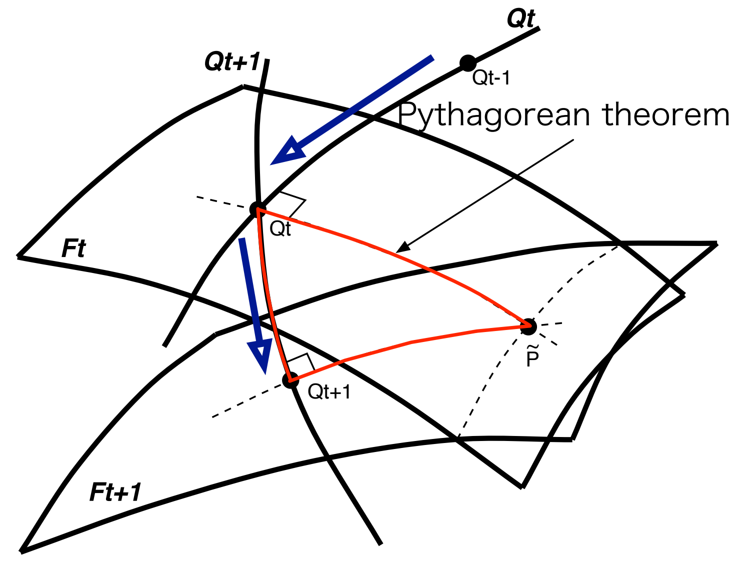
(iteration)

- step 3: update q_t:

$$q_t(y|\mathbf{x}) = u\Big(\xi(q_{t-1}(y|\mathbf{x})) + \alpha_t f_t(\mathbf{x}, y) - b_t(\mathbf{x}, \alpha_t)\Big).$$


Qt
Qt-1
updated weights
=normal direction
Qt
learned classifier
at time t+1
P̃
tangent space of Qt
=direction without improvement


Qt
Qt+1
Qt-1
Qt
Qt+1
P̃
Ft+1

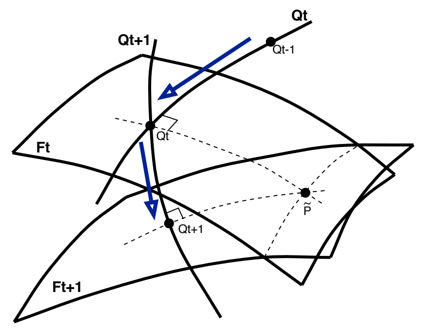

Qt
Qt+1
Qt-1
Pythagorean theorem
Qt
Ft
P̃
Qt+1
Ft+1

(end)

- output:
  construct a majority vote classifier:

$$H(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} F_T(\boldsymbol{x}, y) = \arg\max_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y).$$

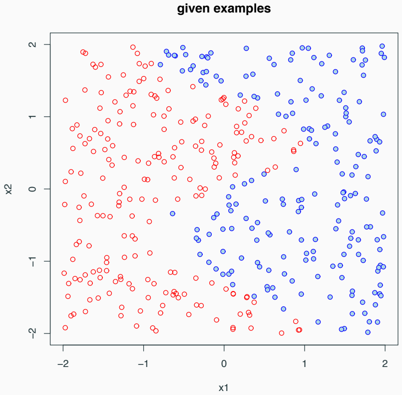
- global model extension:
  - by using appropriately weighted training data, the learning model is extended to the direction to which the total performance can be improved
  - by extending the search space to outside of probability distributions, an efficient algorithm (coordinate descent) is derived



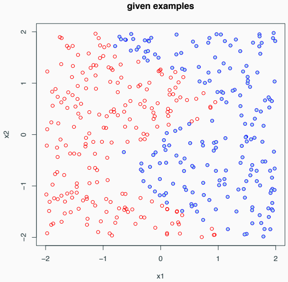
Illustrative Example

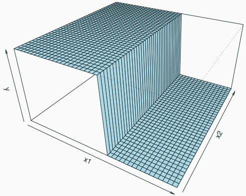
classification problem:

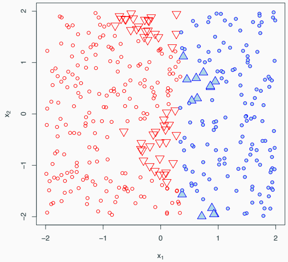
- predict label $y \in \mathcal{Y}$ from corresponding features $\boldsymbol{x} \in \mathcal{X}$
- construct a classifier $h(\boldsymbol{x}) = \hat{y}$ from finite samples



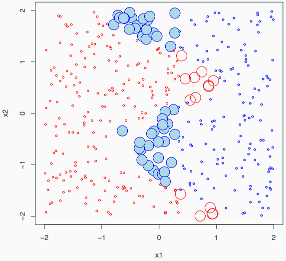
first round

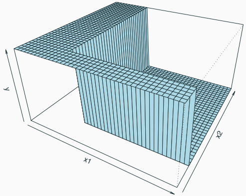


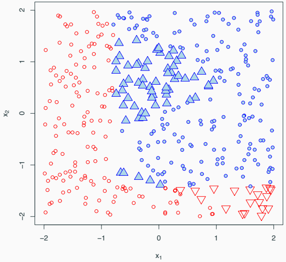




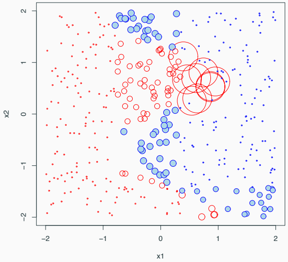
second round

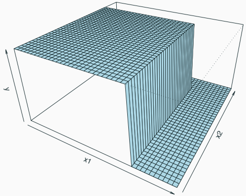


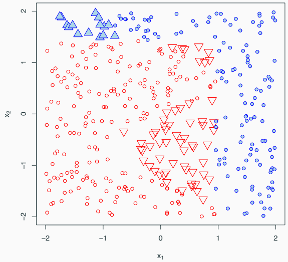




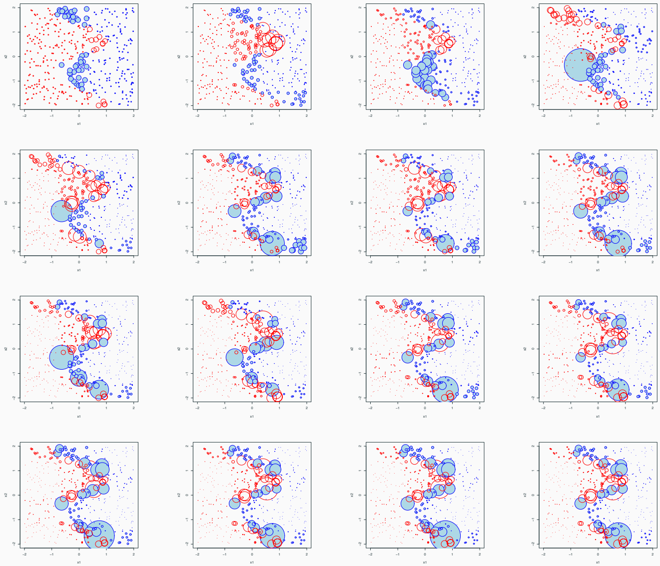
third round



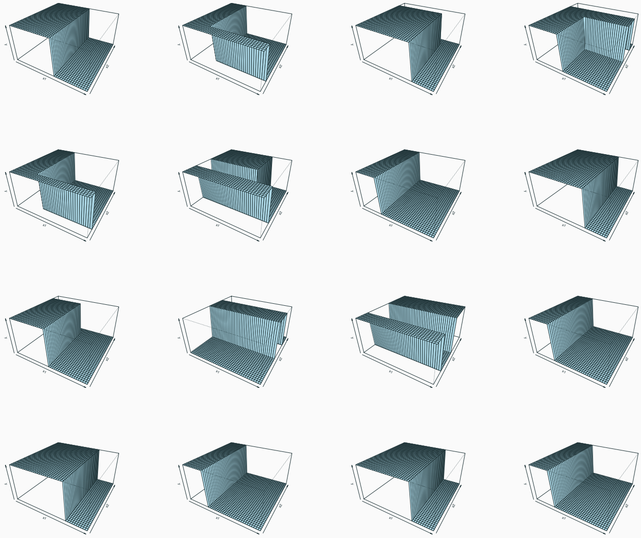




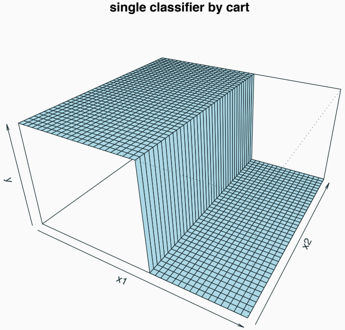
sample weights at each round

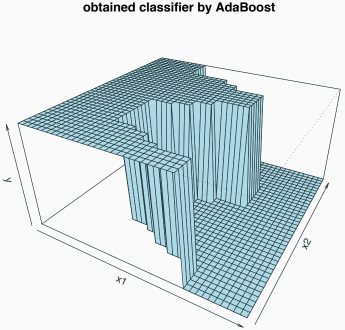


obtained classifier at each round



obtained classifier

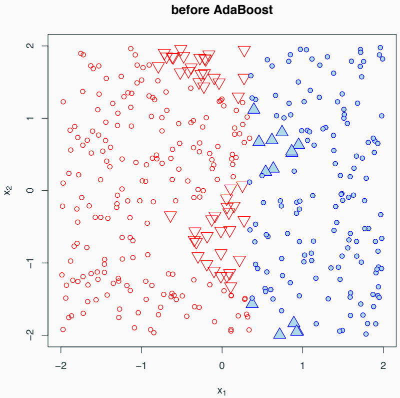




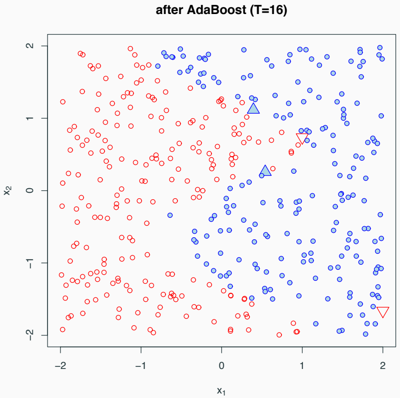
without boosting

with boosting

classification error



without boosting



with boosting

Face Detection

Paul Viola and Michael J. Jones (May 2004). "Robust Real-Time Face Detection." In: *International Journal of Computer Vision* 57 (2), pp. 137–154. DOI: 10.1023/B:VISI.0000013087.49260.fb

- famous boosting application to computer vision
- adopt simple rectangle detectors as weak learners
- construct an efficient classifier with AdaBoost

Conclusion

we presented the following

- some characterization of mixture models
- some geometrical properties of U functions
  - coordinate descent algorithm
  - Pythagorean relation

in addition, possible extensions would be

- characterization of U
- stopping rules for the number of boosting

Domingo, Carlos and Osamu Watanabe (2000). “MadaBoost: A Modification of AdaBoost.” In: *Proceedings of COLT 2000*. the Thirteenth Annual Conference on Computational Learning Theory (Palo Alto, CA, USA, June 28–July 1, 2000). Ed. by Nicolò Cesa-Bianchi and Sally A. Goldman. Morgan Kaufmann, pp. 180–189.

Freund, Yoav (Sept. 1995). “Boosting a Weak Learning Algorithm by Majority.” In: *Information and Computation* 121.2, pp. 256–285. DOI: 10.1006/inco.1995.1136.

Freund, Yoav and Robert E. Schapire (Aug. 1997). “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting.” In: *Journal of Computer and System Sciences* 55.1, pp. 119–139. DOI: 10.1006/jcss.1997.1504.

Murata, Noboru et al. (July 2004). “Information Geometry of U-Boost and Bregman Divergence.” In: *Neural Computation* 16.7, pp. 1437–1481. DOI: 10.1162/089976604323057452.

Viola, Paul and Michael J. Jones (May 2004). “Robust Real-Time Face Detection.” In: *International Journal of Computer Vision* 57 (2), pp. 137–154. DOI: 10.1023/B:VISI.0000013087.49260.fb.