July 7, 2014

Machine Learning Strategies for Big Data Utilization:

Assembling via Statistical Soft Label

Yasuo Matsuyama

Department of Computer Science and Engineering
Waseda University
Tokyo 169-8555, Japan

List of contents

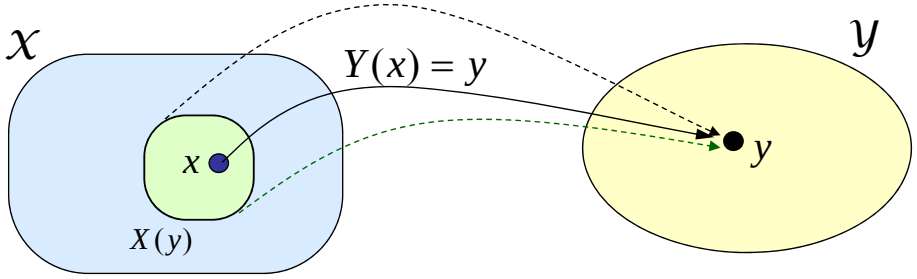
Big data concepts

- True big data has an inaccessible size ⇒ No way to manage.
- 3V for big data: Data's {Volume, Velocity, Variety} need to be manageable (D. Laney, META Group, 2001).
- 4V: Veracity (by Villanova University).
- 5V: Value (by HRBOSS blog).

- So many requirements for human power alone.
- Machine learning strategies: Novel methods inspired by unprecedented problems.
- The fusion with crowd sourcing will also be mentioned.

Concept 1: Incomplete data

Perfectly measured data are incomplete yet!



Complete data ⇔ Unobservable Observable ⇔ Incomplete data

Complete data pdf for modeling: $f(x \mid \theta)$

Incomplete data pdf: $g(y \mid \theta) = \int_{X(y)} f(x \mid \theta) dx$

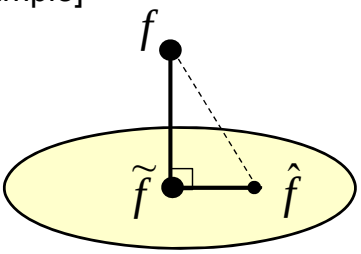
Separable complete data: $X = \{Y, Z\}$ (random variables)

Concept 2: Good models

(1) Essentially, all models are wrong, but some are useful.
(G. E. P. Box; In G. E. P. Box and N. R. Draper, Empirical model building and response surfaces, p. 424, John Wiley & Sons, 1987).

(2) Good models mismatch well, and are therefore dependable.
(The presenter of this talk, Y. Matsuyama).

[Example]



space of m-th order AR processes

$$d_{IS}(f,\hat{f}) = d_{IS}(f,\tilde{f}) + d_{IS}(\tilde{f},\hat{f})$$

modeling | VQ or vector k-means

R. M. Gray, A. Buzo, A. H. Gray, Jr., and Y. Matsuyama, Distortion measures for speech processing, IEEE Trans. vol. ASSP-28, pp. 367-376, 1980.

Case 1: Missing is a weight vector (least squares)

Don't underrate simple classic methods. ∵ Data size is big.

[Example : E. Hoerl & R. W. Kennard, Technometrics, vol.12, pp 69 - 82, 1970.]

$\mathbf{x} \cdots$ position vector

$\{\mathbf{x}_i\}_{i=1}^N \cdots$ observed data

linear model $\cdots \quad y = f(\mathbf{x};\mathbf{w}) = \mathbf{w}^{\mathrm{T}}\mathbf{x}$

RBF kernel model $\cdots y = f(\mathbf{x};\mathbf{w}) = \sum_{i=1}^N w_i \exp\left(-\gamma^2 \| \mathbf{x} - \mathbf{x}_i \|^2\right)$

Least squares learning :

$$\hat{w} = \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^N \left(f(\mathbf{x}_i;\mathbf{w}) - y_i \right)^2 + \lambda \| \mathbf{w} \|^2 \right\}$$

$$= (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

Here,

$$\mathbf{X} = \begin{cases} [\mathbf{x}_1 \cdots \mathbf{x}_N] \cdots \text{for the linear model (matrix)} \\ x_{ij} = \exp\left(-\gamma^2 \| \mathbf{x}_i - \mathbf{x}_j \|^2\right) \cdots \text{for the RBF kernel model} \end{cases}$$

(But, still sensitive to outliers ⇒ many improved methods to the rubustness.)

Case 2: Missing is a boundary set (supervised)

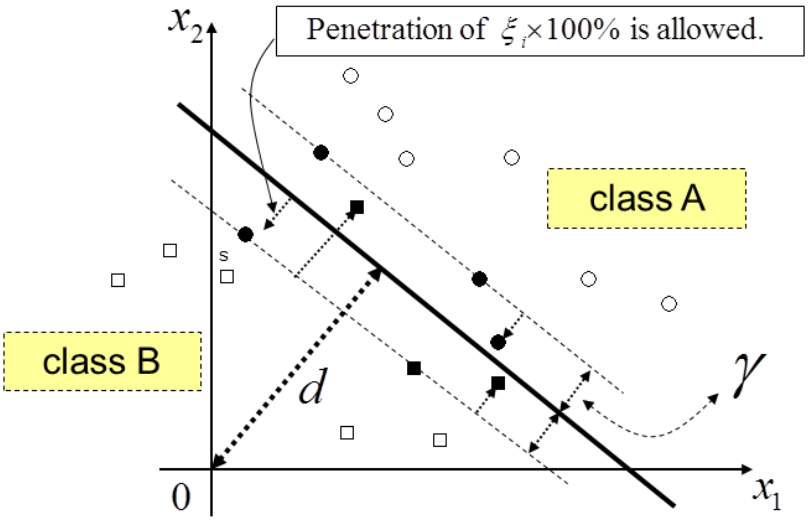
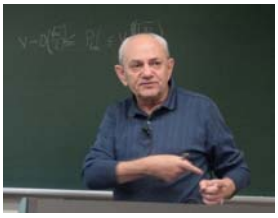


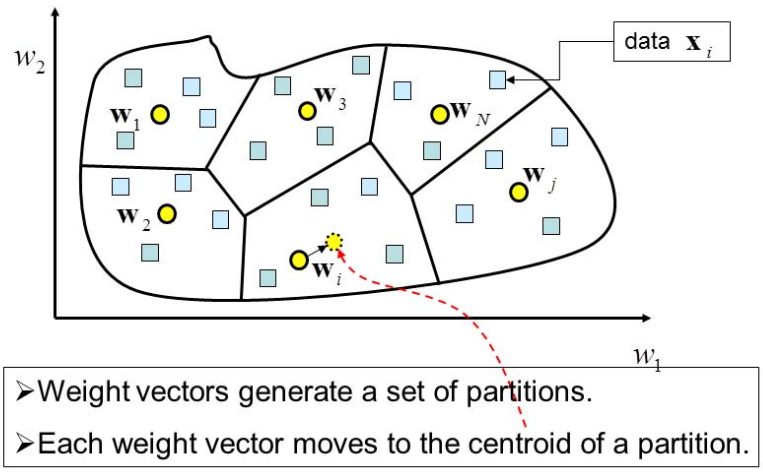
Fig. Support vector machine with a soft margin.

Fig. An SVM example using LIBSVM of C-C. Chang and C-J. Lin of NTU.



Prof. V. Vapnik on 2013-11-26 in Tokyo.
Photograph allowed by himself.
No usage other than academic one is allowed (© Y. Matsuyama).

Case 3-1: Missing is a partition set (unsupervised)



- Weight vectors generate a set of partitions.
- Each weight vector moves to the centroid of a partition.

- Vector k-means (vector quantization) is much alive for structurization of big data.
- J. Linde, A. Buzo, R.M. Gray, IEEE Trans. COM, vol. COM-28, pp.84-95, 1980 (batch mode).
- Y. Matsuyama, R. M. Gray, IEEE Trans. vol. IIT-27, pp.31-40, 1981 (tree + batch VQ).
- Y. Matsuyama, R. M. Gray, IEEE Trans. vol. COM-30, pp. 711-720, 1982 (tree + inverse filter VQ).
- T. Kohonen, Leaning vector quantization, In The Handbook of Brain Theory and Neural Networks., pp. 537–540. MIT Press, Cambridge, MA, 1995 (successive mode, or stochastic gradient descent).

Case 3-2: Missing is a boundary set (unsupervised)

- Vector quantization was unified as a competitive learning algorithm for composite cost functions.
  (Y. Matsuyama, Harmonic competition: A self-organizing multiple criteria optimization, IEEE Trans. NN, vol. 7, pp. 652-668, 1996.)

$$D = \sum_{n=0}^{N-1} D_n$$

$$D_n = \sum_{m=0}^{M-1} \left(f_n + \sum_{k=0}^{K-1} \lambda_{nk} g_{nk} \right) \left(\Pi_{l=0}^{L-1} h_{nl} \right) Q(\mathbf{x}_n, \mathbf{w}_m)$$

$f(\mathbf{x}_n, \mathbf{w}_m) \cdots$ distance between $\mathbf{x}_n$ and $\mathbf{w}_m$

$g_{nk}\left(\{\mathbf{x}_i\}_{i=0}^{N-1}, \{\mathbf{w}_j\}_{j=0}^{M-1} \right) \cdots$ constraint

$h_{nl}\left(\{\mathbf{x}_i\}_{i=0}^{N-1}, \{\mathbf{w}_j\}_{j=0}^{M-1} \right) \cdots$ penalty for label conflict

$Q(\mathbf{x}_n, \mathbf{w}_m) \cdots 1$ if $\mathbf{w}_m$ is the winner for the input $\mathbf{x}_n$; otherwise 0.

Case 3-3: Missing is an exemplar set (unsupervised)

- Affinity propagation (similar to VQ, but an unspecified number of exemplars is found).
  (B. J. Frey and D. Dueck, Clustering by passing messages between data points, Science, vol. 315, no. 5814, pp. 972-976, 2007.

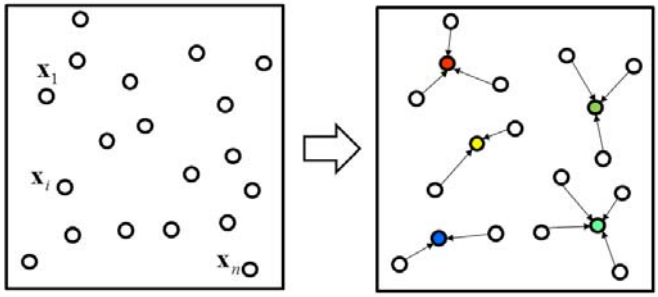


Fig.. Exemplars found by affinity propagation.

- Similarity matrix ⇨ exemplars and clusters associated with them.
- Convergence is not yet proved rigorously (but, often effective).
- For time series, we need its sophistication.

Case 4-1: Missing information is more general (unsupervised/supervised)

- EM algorithm (expectation-maximization algorithm):

This is regarded as a champion algorithm which estimates missing information.

(A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from Incomplete data via the EM Algorithm, J. Royal Statistical Society. Series B, vol. 39, pp. 1-38, 1977).

Complete data pdf: $f(\mathbf{x}\,|\,\boldsymbol{\theta}) = f(\mathbf{y},\mathbf{z}\,|\,\boldsymbol{\theta}) \Rightarrow f(\mathbf{x}\,|\,\boldsymbol{\theta})$

Incomplete data pdf: $g(\mathbf{y}\,|\,\boldsymbol{\theta}) = \int_{\mathcal{X}(\mathbf{y})} f(\mathbf{x}\,|\,\boldsymbol{\theta})\,d\mathbf{x}$

$$\log\frac{g(\mathbf{y}\,|\,\boldsymbol{\theta})}{g(\mathbf{y}\,|\,\boldsymbol{\alpha})} = \mathbf{E}_X\left[\log\frac{f(\mathbf{x}\,|\,\boldsymbol{\theta})}{f(\mathbf{x}\,|\,\boldsymbol{\alpha})}\,\middle|\,\mathbf{y},\boldsymbol{\alpha}\right] + D\left(f(\mathbf{x}\,|\,\mathbf{y},\boldsymbol{\alpha})\,\|\,f(\mathbf{x}\,|\,\mathbf{y},\boldsymbol{\theta})\right)$$

(1) We want this term to be nonnegative.

(3) Keep this term nonnegative!

(2) This term is always nonnegative.

Case 4-2: α-EM algorithm (EM algorithm is yet a special case)

α-EM algorithm's basic equation:

$$L_Y^{(\alpha)}(\psi\,|\,\varphi) = E_{p_{X|Y,\varphi}}[L_X^{(\alpha)}(\psi\,|\,\varphi)] + \frac{1-\alpha}{2}\left\{\frac{p_{Y|\psi}(y\,|\,\psi)}{p_{Y|\varphi}(y\,|\,\varphi)}\right\}^{\frac{1+\alpha}{2}} D_{X|Y}^{(\alpha)}(\varphi\,\|\,\psi)$$

(1) We want to keep this them to be nonnegative.

(3) Keep this term nonnegative

(2) This term is nonnegative for α<1.

The case of α = -1 is the traditional log-EM algorithm.

Y. Matsuyama,
The alpha-EM algorithm: Surrogate likelihood maximization using alpha-logarithmic information measures, IEEE Trans. On Information Theory, vol. 49, No. 3, pp. 692-706, March, 2003.

Case 4-3: Relationship between EM and vector k-means (VQ)

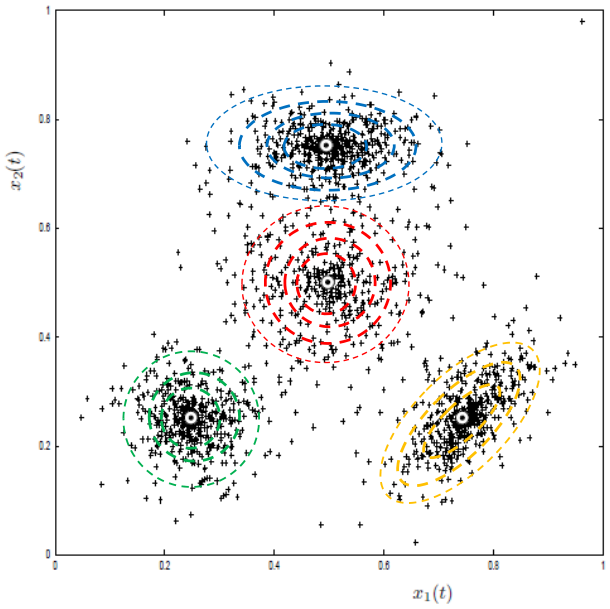


Fig. Gaussian mixture model.

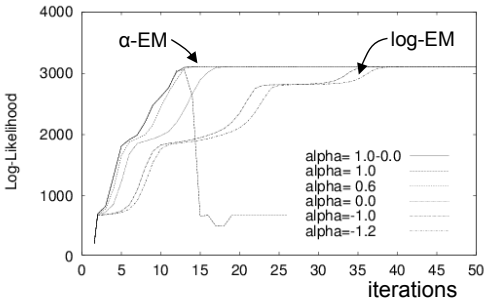


Fig. Convergence of α-EM and log-EM.

- α-EM outperforms log-EM on the convergence speed.
- EM-family ⊃ vector quantization

Case 4-4: α-HMM estimation

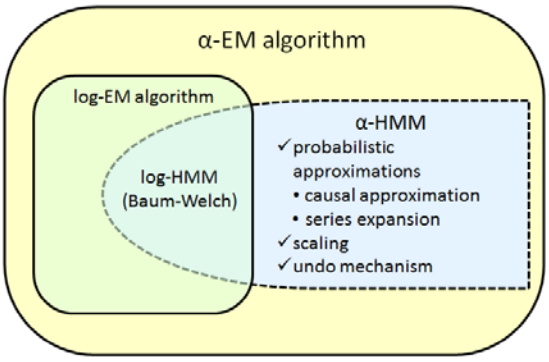


Fig. Relationship of alpha-HMM and log-HMM

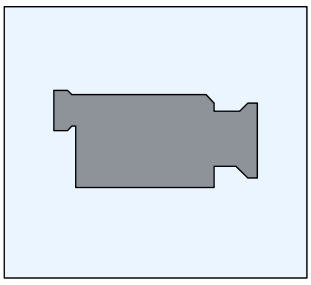


Fig. Alpha-HMM vs. log-HMM (Baum-Welch) for brain signals (NIRS signals).

- Y. Matsuyama, Hidden Markov model estimation based on the alpha-EM algorithm: Discrete and continuous alpha-HMMs, Proceedings of International Joint Conference on Neural Networks, pp.809-816, San Jose, CA, 2011.
- A. Rahimi's correction to L. Rabiner's flaws on scaling was generalized too.

Case 4-5: ICA (missing are hidden components).

[RapidICA]

Step 1: $\mathbf{y} \leftarrow \mathbf{W}\mathbf{z}$

Step 2: $\mathbf{W}_{old} \leftarrow \mathbf{W}$

Step 3: $\mathbf{W} \leftarrow \mathbf{W} + \Delta\mathbf{W}_1$

where

$$\Delta\mathbf{W}_1 = -\alpha\,\mathrm{diag}(1/E[\partial^2 G(y_i)])\,E[\partial G(\mathbf{y})\mathbf{z}^T]$$

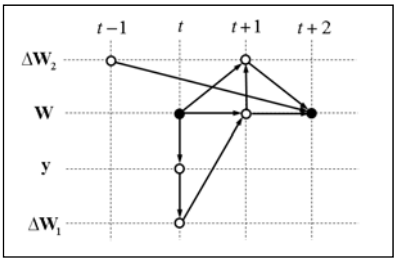
Step 4: $\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^T)^{-1/2}\mathbf{W}$

Step 5: $\Delta\mathbf{W}_{2,old} \leftarrow \Delta\mathbf{W}_2$

Step 6: $\Delta\mathbf{W}_2 \leftarrow \mathbf{W} - \mathbf{W}_{old}$

Step 7: $\mathbf{W} \leftarrow \mathbf{W} + \mathrm{diag}(\eta_i)\Delta\mathbf{W}_2$

Step 8: $\mathbf{W} \leftarrow (\mathbf{W}\mathbf{W}^T)^{-1/2}\mathbf{W}$



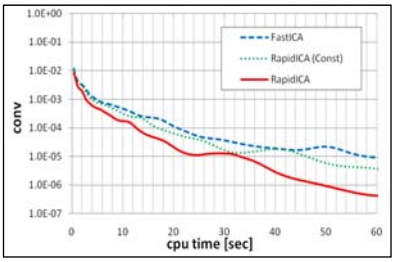


Fig. Rapid ICA

- R. Yokote, Y. Matsuyama, Rapid algorithm for independent component analysis, J. Signal and Info. Proc., vol. 3, pp. 275-285, 2013.
- Fast ICA was by A Hyvärinen; Fast and robust fixed-point algorithms for independent component analysis, IEEE Trans. NN, vol. 10, No. 3, 1999, pp. 626- 634., 1999.

Case studies: Unstructured data management

- Tremendous amount of data is accumulated continuously.
- Mostly, data are unstructured.
- Some are manually labeled, however, labels are usually personal.
- Structurization by machine learning strategies is effective and fashionable.
- New target problems ⇒ new methods
  ⇒ new theory

(1) Similar video retrieval: Numerical labelling

- Video retrieval: One of the most familiar target to be structured.
- The size of each element is already big.
- Lengths are different.

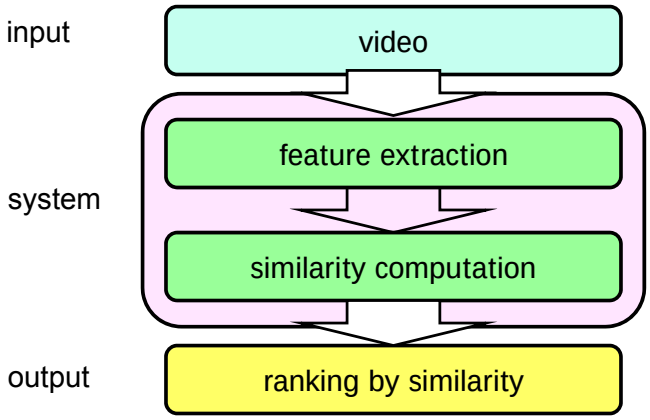


Fig. Similar video retrieval.

(1-2) Similar video retrieval: Similarity

- How do we define similarity measures?

i \ j		E	L	E	P	H	A	N	T
	0	← -6	← -12	← -18	← -24	← -30	← -36	← -42	← -48
P	↑ -6	↖ -1	← -7	↖← -13	↖ -11	← -17	← -23	← -29	← -35
A	↑ -12	↖ ↑ -7	↖ -2	↖← -8	↖← -14	↖ -13	↖ -13	← -19	← -25
N	↑ -18	↖ -12	↑ -8	↖ -2	← -8	↖ -13	↖ -15	↖ -7	← -13
D	↑ -24	↖ -16	↑ -14	↖ -6	↖ -3	↖ ← -9	↖ ← -15	↑ -13	↖ -8
A	↑ -30	↑ -22	↖ -17	↑ -12	↖ -7	↖ -5	↖ -5	← -11	↖ -13

Fig. Needleman-Wunsch algorithm in bioinformatics.

(1-3) Similar video retrieval: Similarity

- Too simple: L-distance (Levenshtein distance) from another big data area is not applicable.
  - ✓Only letter-wise.
  - ✓Not context-aware.

- Exemplars are expressed by floating-point numbers.
- A more general distance measure・・・M-distance*.
- Each video can be numerically labelled using positions and expressions of exemplars.

(*) Y. Matsuyama and M. Moriwaki

(1-4) Similar video retrieval: Similarity

- How do we express each as a vector.
- You can find many effective ways.
- We here show you an example by CSD.

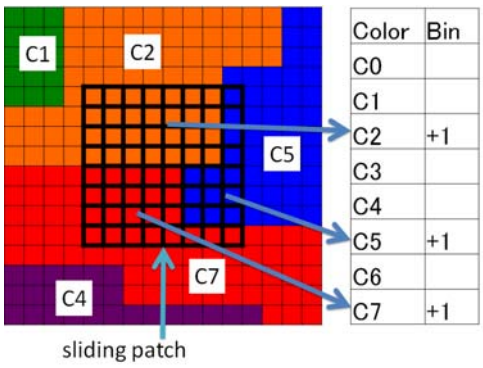


Fig. A conceptual CSD (color structure descriptor).

(1-5) Similar video retrieval: Exemplars

- Key frames ⇒ exemplars ⇒ their numbers are variable.
- Each frame is a feature-extracted vector such that

$$\mathbf{x}_i = [x_{i1}, \cdots, x_{id}]^T, \quad x_{ik} \geq 0, \quad \sum_{k=1}^{d} x_{ik} = 1.$$

- Then, $\mathbf{x}_i$ is in a simplex.
- Exemplars are in this space.

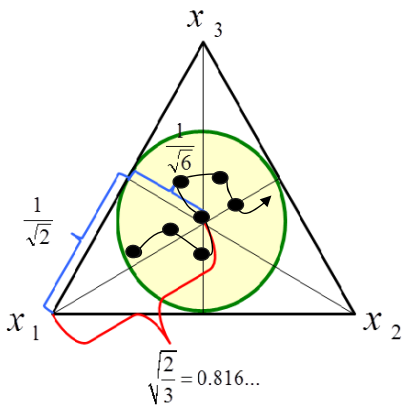


Fig. 2D simplex.

(1-6) Similar video retrieval: Exemplars (cnt)

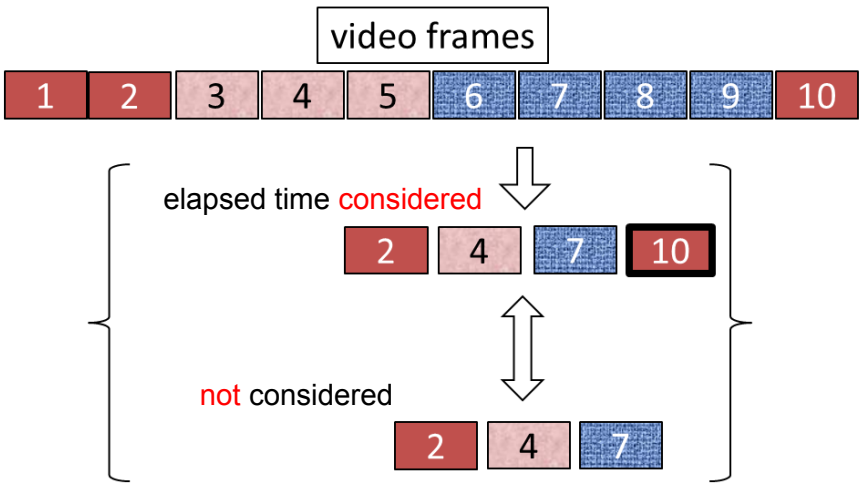


Fig. Exemplar frames reflecting time course.

(1-7) Similar video retrieval: How to find such exemplars?

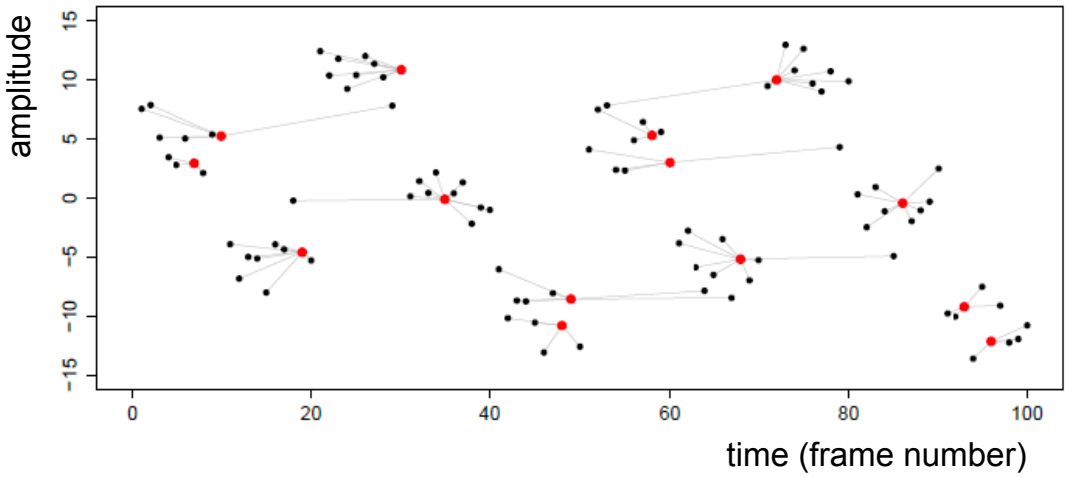


Fig. Time bound affinity propagation, w=20 (computed by Mr. M. Moriwaki).

- Time-bound affinity propagation (TBAP) ・・・ windowing.
- Exemplars per se are found.
- The number of exemplars is variable.

(1-7) Similar video retrieval: How to find such exemplars? (cnt)

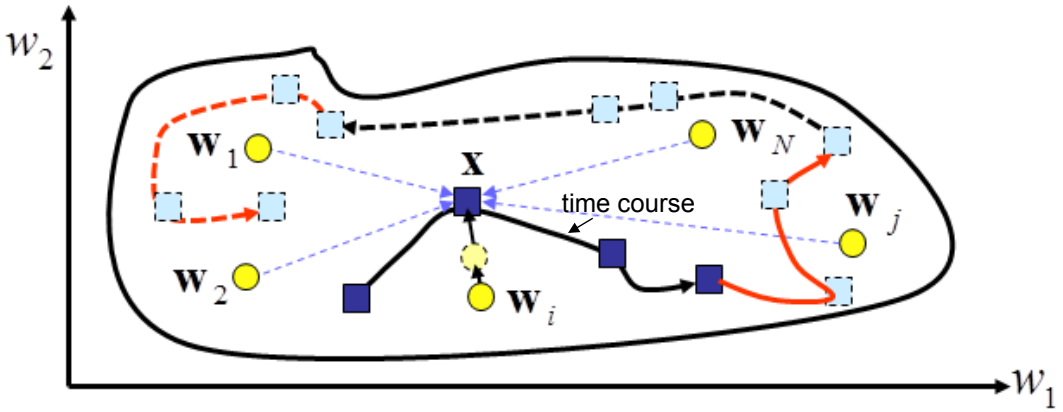


Fig. Harmonic competition type (generalized vector k-means or VQ).

- A class of the harmonic competitive learning.
- Generalized vector k-means are found.
- Number of competitive representatives is pre-specified.
- Each exemplar is found as the nearest frame to the representative.

(1-8) How can we compare exemplar sets?

$$\text{M-distance} \equiv d\left(\{\text{exemplar}_i, \text{context}_i\}_{i=1}^{N_A}, \{\text{exemplar}_j, \text{context}_j\}_{j=1}^{N_B} \right)$$

- M-distance 1 (context-aware global alignment distance)
- Naming is after the Levenshtein distance (edit distance).
- L-distance was a very special case:
  - ✓ Discrete alphabet
  - ✓ Identical or not identical･･･{0, 1}
  - ✓ Letter-to-letter (no ability reflect contexts)
  - ✓ Gap penalty is only letter-wise.
- M-distance 2 (context-aware local alignment distance) exists too.

(1-9) How can we compare exemplar sets? (cnt)

				Video B			
	i ↓ j →			exemplar 1	exemplar 2	exemplar 3	exemplar 4
				$E^B_1 = 2$	$E^B_2 = 3$	$E^B_3 = 2$	$E^B_4 = 1$
			0	← -0.4	← -1	← -1.4	← -1.6
Video A	exemplar 1	$E^A_1 = 2$	↑ -0.4	↖ 1.054	↖ 0.743	← 0.343	← 0.143
	exemplar 2	$E^A_2 = 2$	↑ -0.8	↖ 0.834	↖ 2.597	← 2.197	← 1.997
	exemplar 3	$E^A_3 = 3$	↑ -1.4	↖ 0.468	↖ 2.176	↖ 3.715	← 3.515

Fig. Computation of M-distance for the global alignment.

(1-10) How do we compare exemplar sets in DB?

$$u(A,B) = h(f_{\text{last}}) / w\left(\sum_i E_i^A, \sum_j E_j^B\right)$$

$h(x) \cdots$ monotone increasing positive function

$w(a,b) \cdots$ average function between a and b

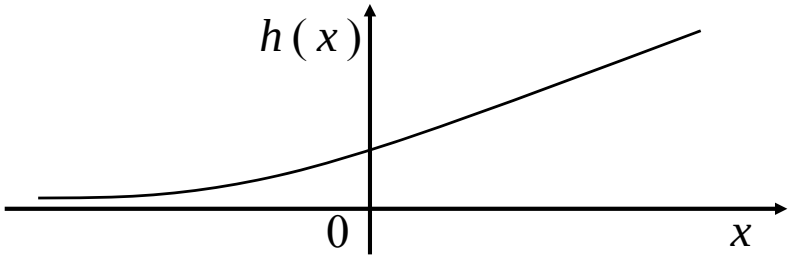


Fig. A monotone increasing positive function for videos in DB.

(1-11) How do we compare exemplar sets in DB? (cnt)

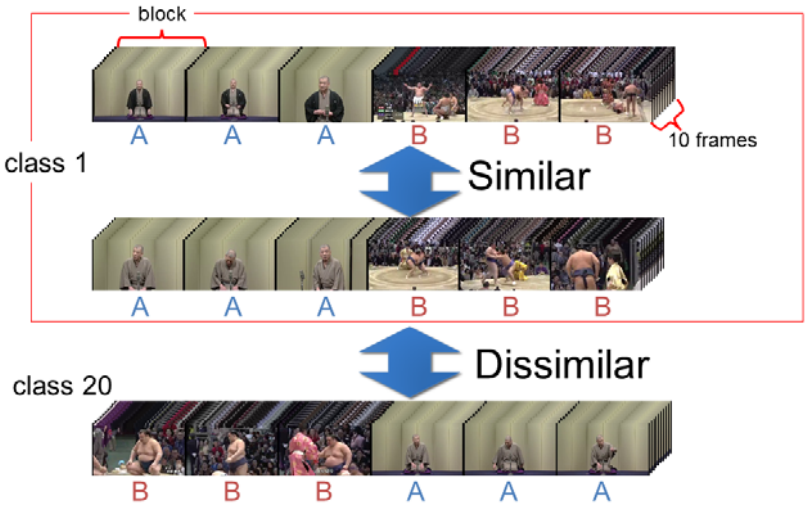


Fig. Exemplar set shuffling for similarity measurement:
Sumo wrestlers and comic story tellers from NHK archive.

(1-12) How do we compare exemplar sets in DB?

ROI (region of interest)

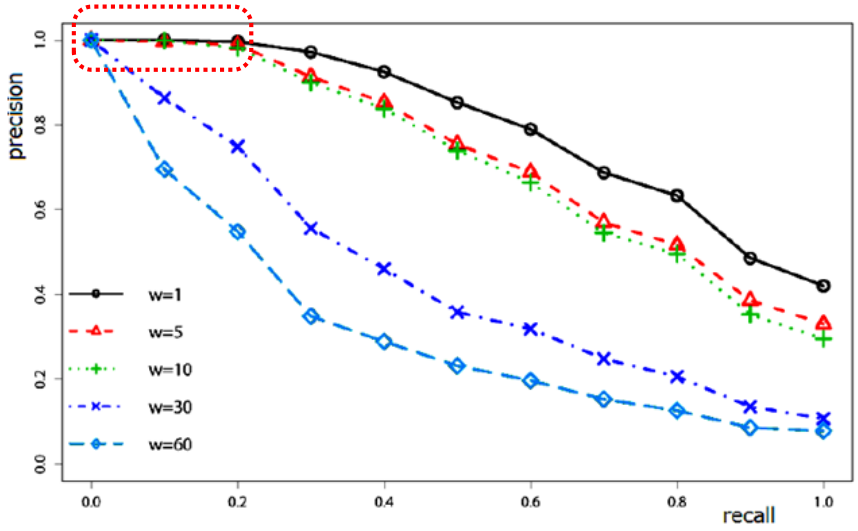


Fig. 11-point interpolated precision and recall.

(2-1) Numerical data and GUI

$$\text{Enquete data}: \mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_i, \cdots, \mathbf{x}_n]$$

$$i\text{-th person}: \quad \mathbf{x}_i = [x_{i1}, \cdots, x_{ij}, \cdots, x_{id}]^T$$

$$x_{ij} \in \{-1, 0, 1\}$$

$$\text{dissimilarity} \quad d(\mathbf{x}_i, \mathbf{x}_k) \equiv -\mathbf{x}_i^T \mathbf{x}_k$$

- Enquete has many queries.
- Target persons are tired of answering ⇒ many blanks.
- If we need to estimate blank items ⇒ EM family, NNMD.
- We put 0 to blank items.
- In this example, we consider a problem of GUI design.

(2-2) Numerical data and GUI (cnt)

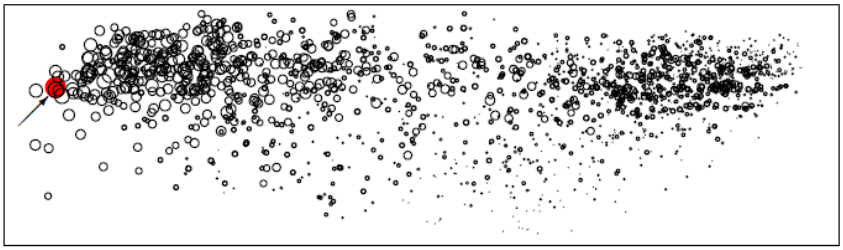


Fig. Job hunters' global map (from a real Web site).

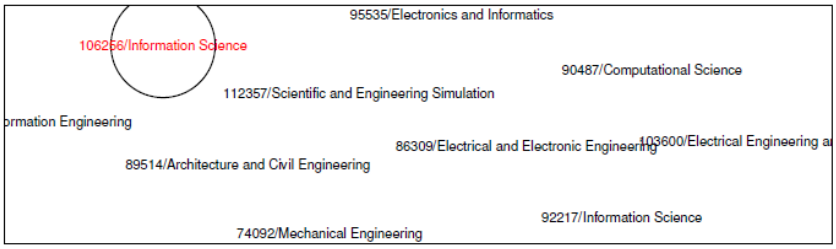


Fig. Enlarged local user map: Icon positions are more uniform.

(with Mr. H, Kamiya and Dr. R. Yokote)

(2-3) Numerical data and GUI (cnt)

Fig. Doubly spherical GUI for all-recording TV database (with Mr. M. Maejima).

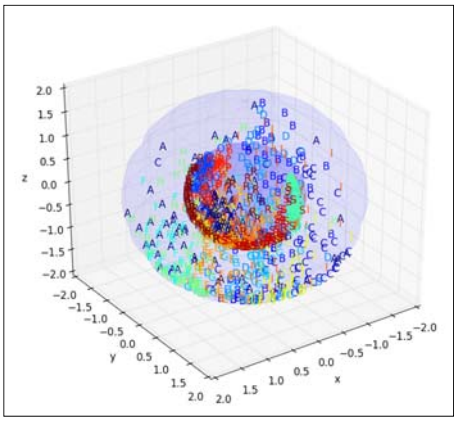


Fig. Doubly spherical GUI for NIRS brain signals with/without a task (with Mr. T. Horie).

(3-1) Security via brain signals

- Big data per se ⇒ GUI ⇒ security by authentication.
- Bypassing the password
  (DARPA, the N.Y. Times, Mar. 17, 2012).
- Brain signals (NIRS, near infrared spectroscopy).

Fig. NIRS measurement for user authentication.

(3-2) Security via brain signals (cnt)

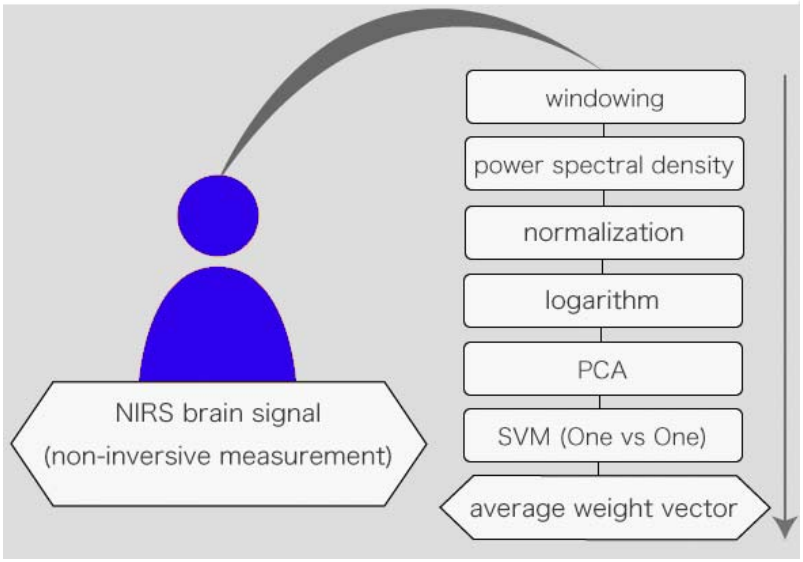


Fig. SVM used as a filter.

(3-3) Security via brain signals (cnt)

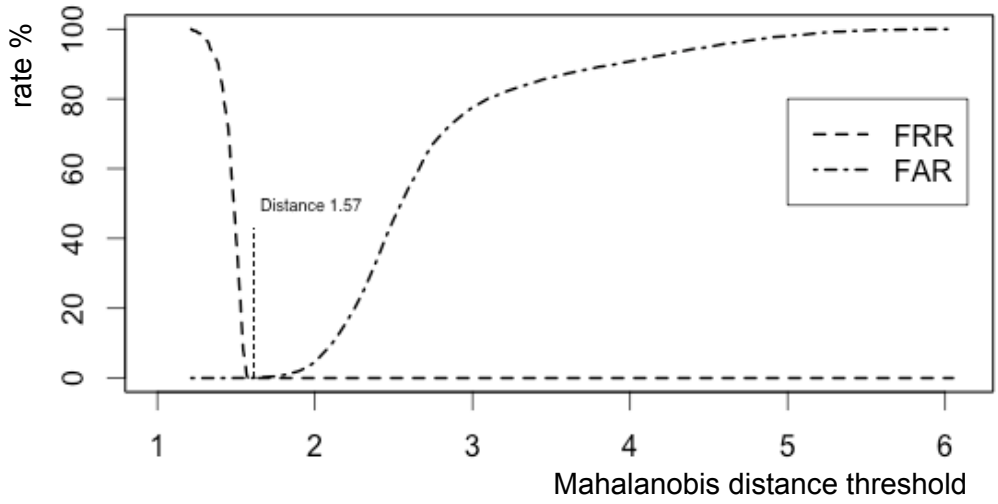


Fig. False acceptance rate and false rejection rate: Equal error rate = 99.97%.

(4-1) Committee decision with crowd sourcing

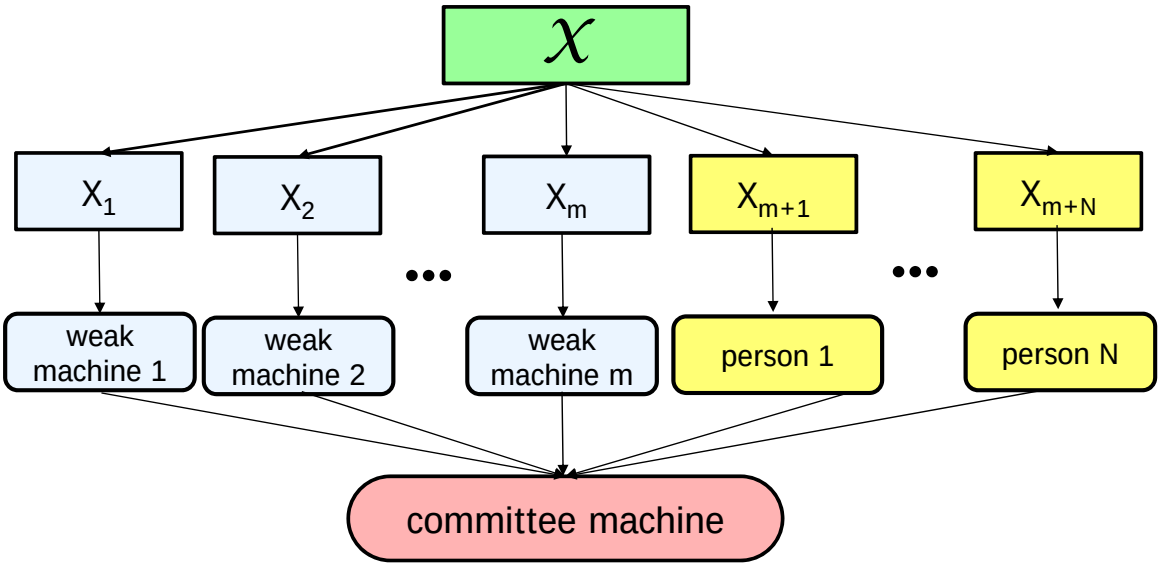


Fig. Bagging (bootstrap aggregation) including non-professional crowd.

(4-2) Committee decision with crowd sourcing

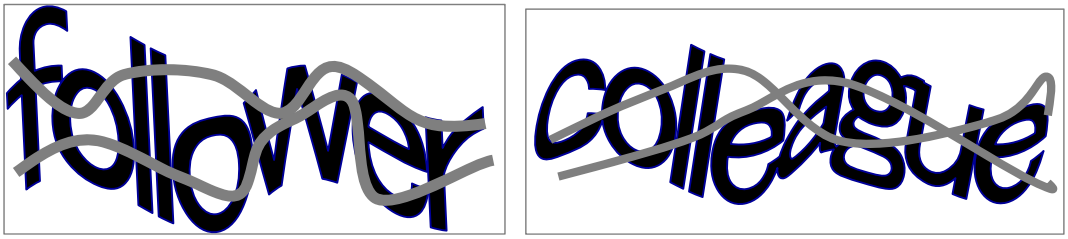
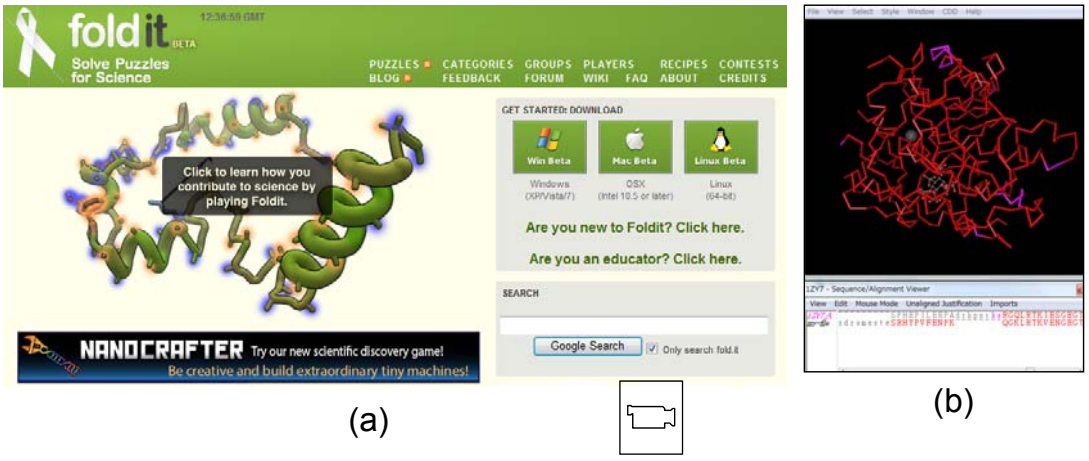


Fig. Identification of human and nonhuman; e. g., Google reCAPTCHA.

Just one of these two queries are used for the authentication.
The other is used for the improvement of machine's ability.

(4-3) Committee decision with crowd sourcing



(a) (b)

Fig. Crowd sourcing example in bioinformatics.
(a) Foldit by University of Washington.
(b) 3D view is this presenter's own by using an NCBI's tool.

Note: These illustrations should be used only for academic uses.

Concluding remarks

- Common methods exist for a variety of big data.
- Machine learning gives one of such foundations. Seeds for new methods exist.
- Compatibility with human sensibility needs to be enhanced more.
- Committee-based methods with crowd sourcing will be quite powerful.

Thank you!

謝々!