Dimensionality Reduction by Random Mapping: Fast Similarity Computation for Clustering

Samuel Kaski
Helsinki University of Technology

1998

© Miki Rubinstein

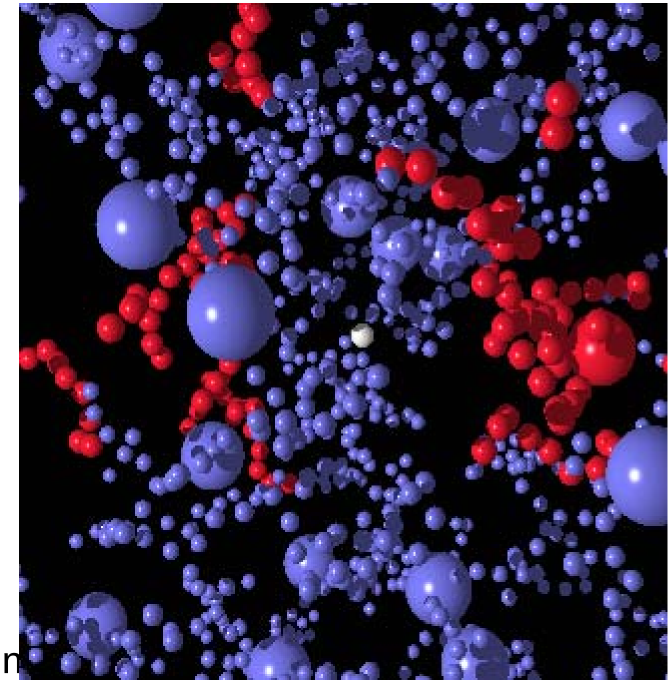
Outline

- Motivation
- Standard approaches
- Random mapping
- Results (Kaski)
- Heuristics
- *Very* general overview of related work
- Conclusion

© Miki Rubinstein

Motivation

- Feature vectors
  - Pattern recognition
  - Clustering
  - Metrics (distances), similarities
- High dimensionality
  - Images – large windows
  - Text – large vocabulary
  - ...
- Drawbacks
  - Computation
  - Noise
  - Sparse data



© Miki Rubinstein

Dimensionality reduction methods

- Feature selection
  - Adapted to nature of data. E.g. text:
    - Stemming (going → go, Tom's → Tom)
    - Remove low frequencies
  - Not generally applicable
- Feature transformation / Multidimensional scaling
  - PCA
  - SVD
  - ...
  - Computationally costly

→ Need for faster, generally applicable method

© Miki Rubinstein

Random mapping

- Almost as good: Natural similarities / distances between data vectors are approx. preserved
- Reasoning
  - Analytical
  - Empirical

© Miki Rubinstein

Related work

- Bingham, Mannila, ’01: results of applying RP on image and text data
- Indyk, Motwani ’99: use of RP for approximated NNS, a.k.a Locality-Sensitive Hashing
- Fern, Brodley ’03: RP for high dimensional data clustering
- Papadimitriou ’98: LSI by random projection
- Dasgupta ’00: RP for learning high dimensional Gaussian mixture models
- Goel, Bebis, Nefian ’05: Face recognition experiments with random projection
  - Thanks to Tal Hassner

© Miki Rubinstein

Related work

- *Johnson-Lindenstrauss lemma (1984):*

  for any $0 < \varepsilon < 1$ and any integer n, let k be a positive integer such that

  $$k \geq \frac{4 \ln n}{\varepsilon^2 / 2 - \varepsilon^3 / 3} = O(\varepsilon^{-2} \ln n)$$

  Then for any set P of n points in $\mathbb{R}^d$, there is a map $f : \mathbb{R}^d \rightarrow \mathbb{R}^k$ such that for all $p, q \in P$

  $$(1-\varepsilon) \| p - q \|^2 \leq \| f(p) - f(q) \|^2 \leq (1+\varepsilon) \| p - q \|^2$$

- Dasgupta [3]

© Miki Rubinstein

Johnson-Lindenstrauss Lemma

- Any n point set in Euclidian space can be embedded in suitably high (logarithmic in n, *independent of d*) dimension without distorting the pairwise distances by more that a factor of $(1 \pm \varepsilon)$

© Miki Rubinstein

Random mapping method

- Let $x \in \mathbb{R}^n$
- Let R be dxn matrix of random values where $||r_i||=1$ and each $r_{ij} \in R$ is normally i.i.d with mean 0

$$y_{[dx1]} = R_{[dxn]} x_{[nx1]} \qquad d << n$$

$$= \sum_{i=1}^{n} r_i x_i$$

© Miki Rubinstein

Random mapping method

$$\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} x_1 + \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} x_2 + \cdots + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} x_n = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = x$$

$$\begin{pmatrix} r_{11} \\ r_{21} \\ \vdots \\ r_{d1} \end{pmatrix} x_1 + \begin{pmatrix} r_{12} \\ r_{22} \\ \vdots \\ r_{d2} \end{pmatrix} x_2 + \cdots + \begin{pmatrix} r_{1n} \\ r_{2n} \\ \vdots \\ r_{dn} \end{pmatrix} x_n = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_d \end{pmatrix} = y$$

© Miki Rubinstein

Similarity

$$sim(u,v) = \cos\theta = \frac{u \cdot v}{\| u \| \ \| v \|}$$

$$\text{if } \| u \| = 1, \| v \| = 1$$

$$\text{then } \cos\theta = u \cdot v$$

© Miki Rubinstein

Random mapping method

- How will it affect the *mutual similarities* between the data vectors?
- As *R* is more orthonormal → the better. however
- R is generally not orthogonal
- Hecht-Nielsen [4]: in a high dimensional space, there exists a much larger number of almost orthogonal than orthogonal directions
- So, *R* might be *sufficiently good approximation* for a basis

© Miki Rubinstein

Transformation of similarities

- Similarity measure:

$$sim(u,v) = \cos\theta = \frac{u \cdot v}{\|u\| \ \|v\|} = u \cdot v \ \text{ for unit vectors}$$

$$x^T y = n^T R^T R m \qquad \text{where } n, m \in \mathbb{R}^n$$

$$R^T R = I + \varepsilon \qquad \text{where } \varepsilon_{ij} = r_i^T r_j \text{ for } i \neq j \text{ and } \varepsilon_{ii} = 0$$

- Properties of ε:
  - $E(\varepsilon_{ij}) = E(r_i^T r_j) = E\left(\sum_{k=1}^{d} r_{ik} r_{jk}\right) = \sum_{k=1}^{d} \left[E(r_{ik}) E(r_{jk})\right] = 0$

© Miki Rubinstein

Recall

- Pearson correlation coefficient

$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$$

- Sample correlation

$$\hat{\rho}_{x,y} = r_{x,y} = \frac{\sum[(x_i - \overline{x})(y_i - \overline{y})]}{\sqrt{\sum(x_i - \overline{x})^2}\sqrt{\sum(y_i - \overline{y})^2}}$$

- Geometric interpretation

$$\cos\theta = \frac{x \cdot y}{\sqrt{x \cdot x}\sqrt{y \cdot y}}$$

© Miki Rubinstein

Recall

- Fisher (r2z) Transformation

  Let X, Y normally distributed
  and let r be correlation of sample of size N from X,Y

  $$z = \frac{1}{2}\log_e \frac{1+r}{1-r}$$

  then z is approximately normally distributed with standard deviation $\frac{1}{\sqrt{\text{N-3}}}$

- Variance of z is estimate of the variance of the population correlation

© Miki Rubinstein

Transformation of similarities

- Properties of ε:
  - ε_{ij} is an estimate of the correlation coefficient between two normally i.i.d random variables r_i and r_j

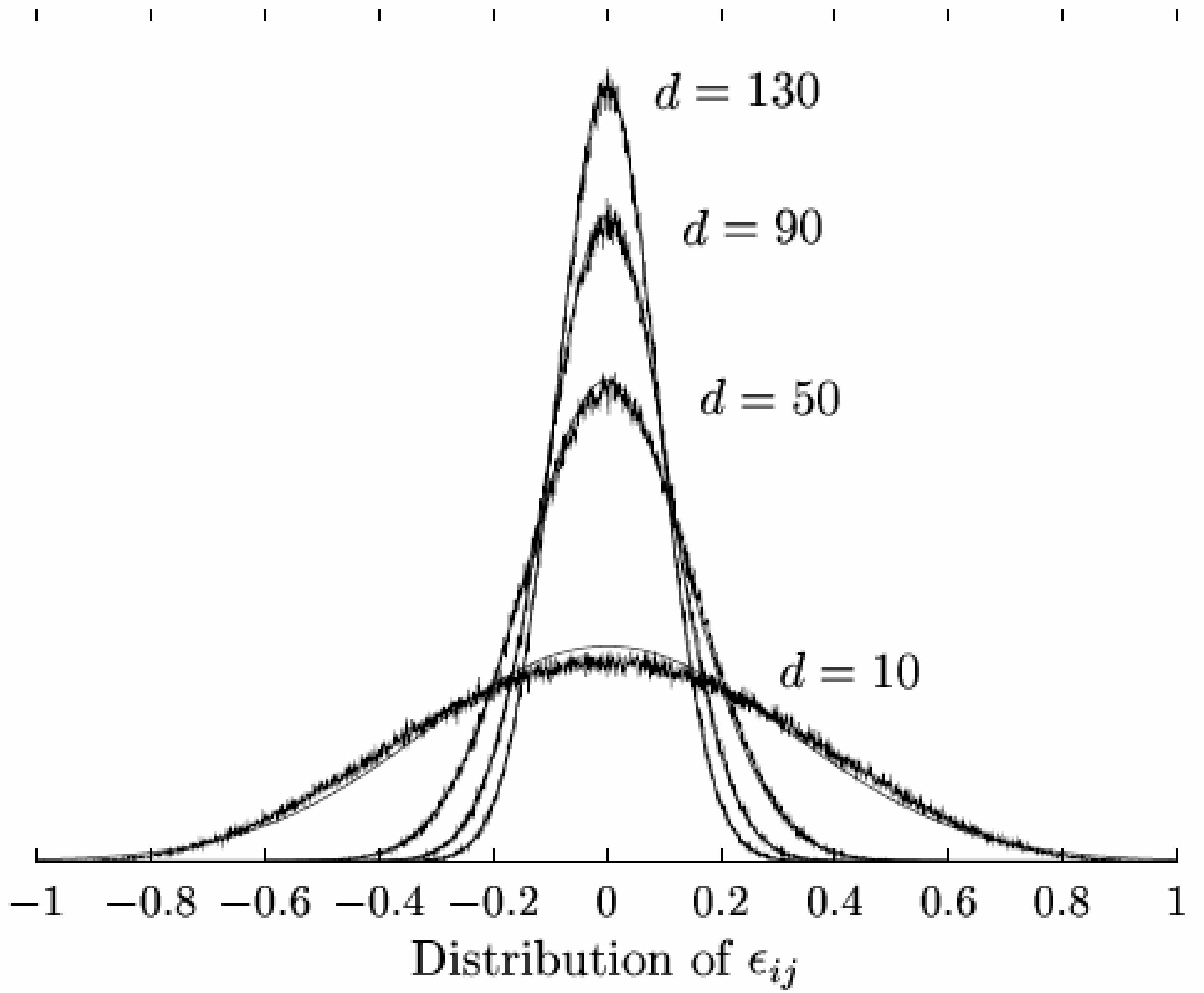
$$\frac{1}{2}\ln\frac{1+\varepsilon_{ij}}{1-\varepsilon_{ij}} \text{ is approximately normally distributed}$$

$$\text{with variance } \sigma_{\varepsilon}^{2} = \frac{1}{d\text{-}3} \approx 1/d \text{ for large d}$$

$$\Rightarrow \text{ as } d \to \infty,\ \ R^{T}R \to I$$

© Miki Rubinstein

Transformation of similarities



Transformation of similarities

- Statistical properties

Let $n, m \in \mathbb{R}^n$, and assume n, m are normalized

$$x^T y = n^T R^T R m = n^T (I + \varepsilon) m = n^T m + n^T \varepsilon m$$
$$= n^T m + \sum_{k \neq l} \varepsilon_{kl} n_k m_l \qquad (\text{recall } e_{kk} = 0)$$

let $\delta = \sum_{k \neq l} \varepsilon_{kl} n_k m_l$

- $$E(\delta) = E\left(\sum_{k \neq l} \varepsilon_{kl} n_k m_l \right) = \sum_{k \neq l} n_k m_l E(\varepsilon_{kl}) = 0$$

© Miki Rubinstein

Transformation of similarities

- Variance of δ:

$$\sigma_\delta^2 = E(\delta^2) - (E(\delta))^2 = E[(\sum_{k\neq l} \varepsilon_{kl} n_k m_l)(\sum_{p\neq q} \varepsilon_{pq} n_p m_q)] - 0 =$$

$$= \sum_{k\neq l} \sum_{p\neq q} n_k m_l n_p m_q E[\varepsilon_{kl} \varepsilon_{pq}]$$

$$E[\varepsilon_{kl} \varepsilon_{pq}] = E\left[\sum_i r_{ki} r_{li} \sum_j r_{pj} r_{qj}\right] = E\left[\sum_i \sum_j r_{ki} r_{li} r_{pj} r_{qj}\right]$$

$$E[\varepsilon_{kl} \varepsilon_{pq}] \neq 0 \text{ only for } (k = p \text{ and } l = q) \text{ or } (k = q \text{ and } l = p)$$

$$\text{denote } c_1 = (k = p, l = q), c_2 = (k = q, l = p)$$

© Miki Rubinstein

Transformation of similarities

- Variance of δ:

$$\sigma_\delta^2 = \sum_{k \neq l} n_k^2 m_l^2 \sigma_\varepsilon^2 + \sum_{k \neq l} n_k m_l n_l m_k \sigma_\varepsilon^2 \qquad \text{(corresponds to } c_1,\ c_2 \text{ respectively)}$$

$$= \left[\sum_k n_k^2 \sum_{l \neq k} m_l^2 + \sum_k n_k m_k \sum_{l \neq k} n_l m_l \right] \sigma_\varepsilon^2$$

$$= \left[\sum_k n_k^2 (1 - m_k^2) + \sum_k n_k m_k \left(\sum_l n_l m_l - n_k m_k \right) \right] \sigma_\varepsilon^2 \quad (\|n\| = 1, \|m\| = 1)$$

$$= \left[1 - \sum_k n_k^2 m_k^2 + (\sum_k n_k m_k)^2 - \sum_k n_k^2 m_k^2 \right] \sigma_\varepsilon^2$$

$$= \left[1 + (\sum_k n_k m_k)^2 - 2 \sum_k n_k^2 m_k^2 \right] \sigma_\varepsilon^2$$

© Miki Rubinstein

Transformation of similarities

- Variance of δ:

$$(\sum_k n_k m_k)^2 \leq 1 \text{ by Cauchy-Schwartz } (n, m \text{ normalized})$$

$$\Rightarrow \quad \sigma_\delta^2 \leq 2\sigma_\varepsilon^2 \approx 2/d$$

That is, ***the distortion of the inner products as a result of applying random mapping is 0 on average and its variance is proportional to the inverse of the dimensionality of the reduced space*** (x 2)

© Miki Rubinstein

Sparsity of the data

- Say we constrain the input vectors to have L 1's, and say K of those occur in same position in both vectors

$$\Rightarrow n^T m = \frac{K}{\sqrt{L}\sqrt{L}} = \frac{K}{L}$$

Now, let's normalize n, m and we get K corresponding dimenstions, each with value $\left(\sqrt{L}\right)^{-1}$

$$\Rightarrow \sigma_\delta^2 = [1 + (\sum_k n_k m_k)^2 - 2\sum_k n_k^2 m_k^2]\sigma_\varepsilon^2 = [1 + (\frac{K}{L})^2 - 2(\frac{K}{L^2})]\sigma_\varepsilon^2$$

$$= [1 + (\frac{K}{L})^2 - 2(\frac{K}{L})\frac{1}{L}]\sigma_\varepsilon^2$$

→ Sparser data ➔ smaller variance of error!

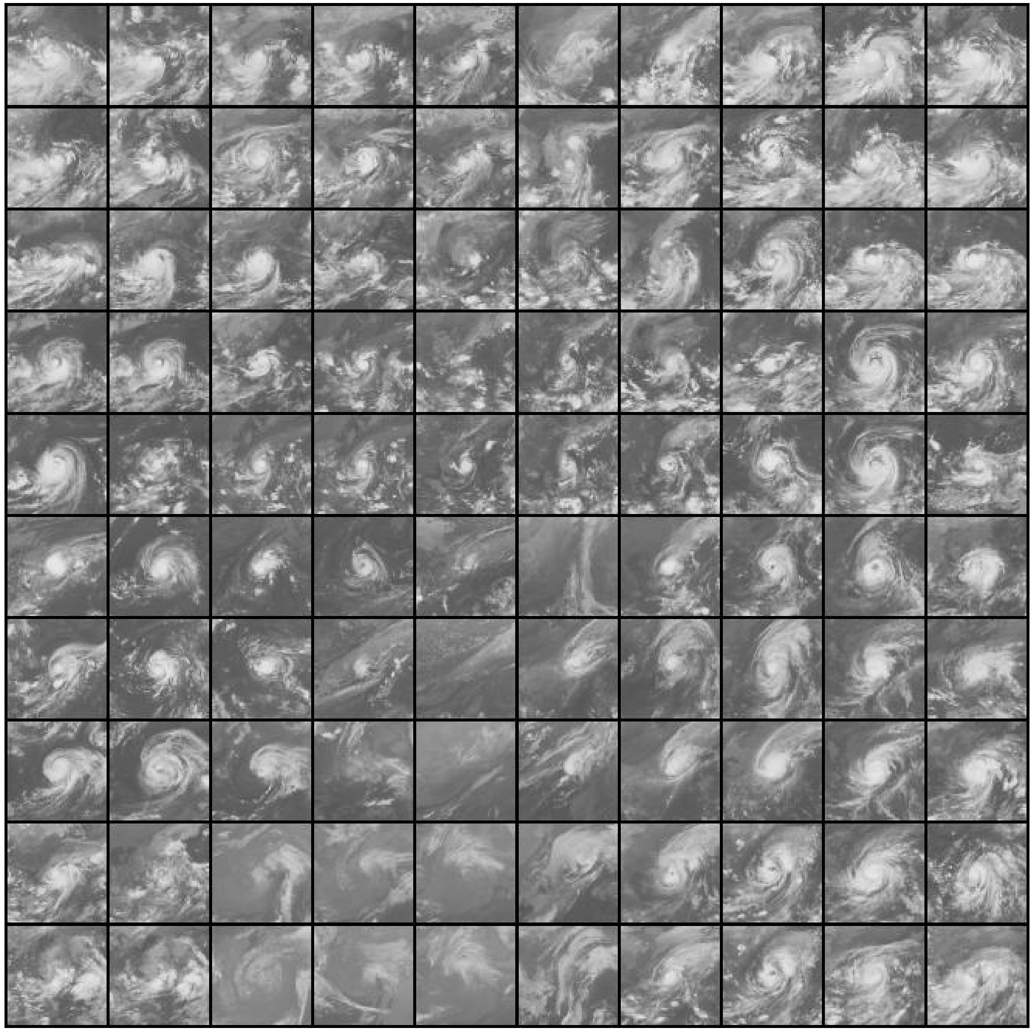
© Miki Rubinstein

Till now

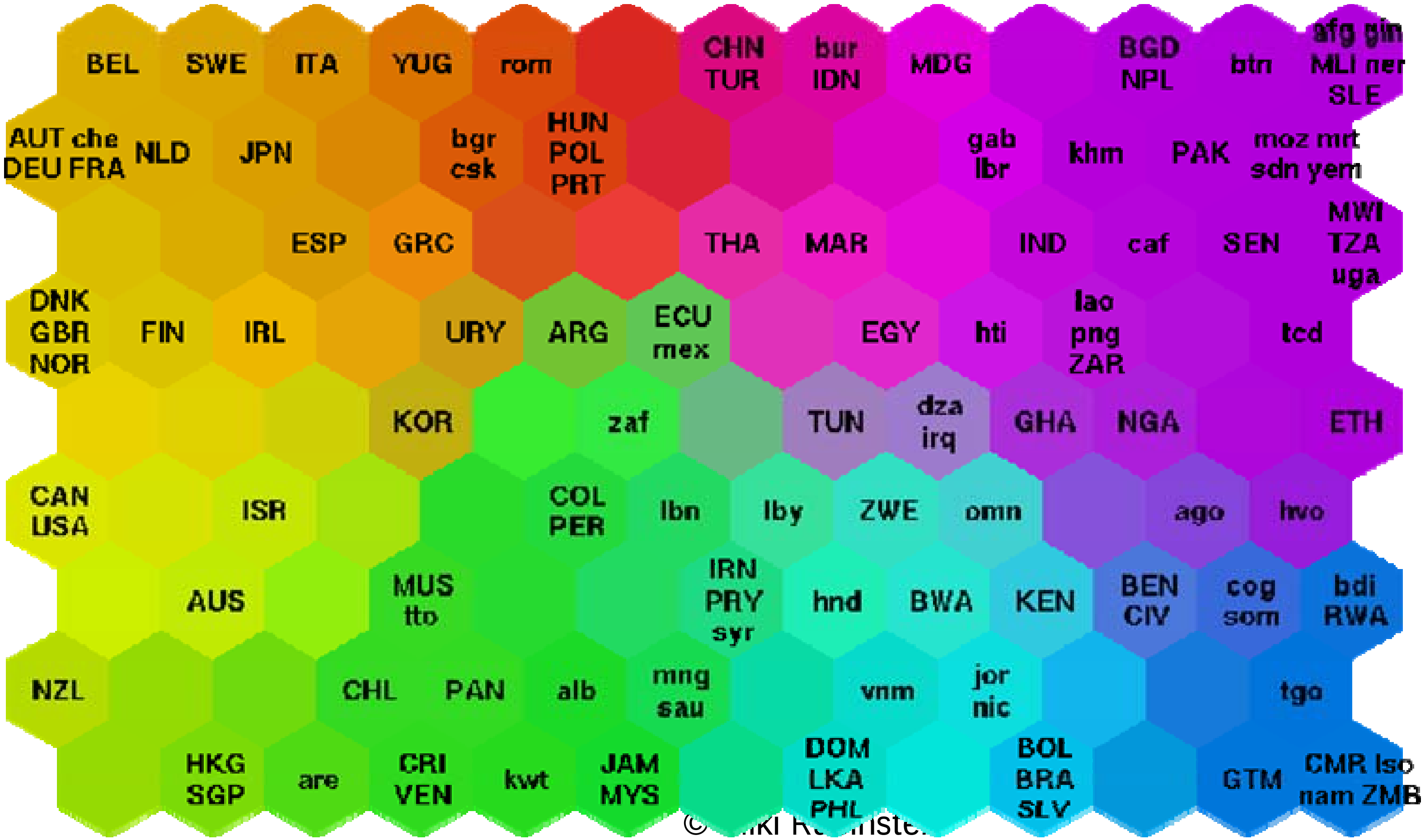
- $x^T y = n^T R^T Rm$
- Error matrix
  - Expected = 0
  - Variance proportional to 1/d
- Added distortion
  - Expected = 0
  - Variance is O(2/d)
- Behaves better on sparse data

© Miki Rubinstein

Self Organizing Maps



Self Organizing Maps



Self Organizing Maps

- Kohonen Feature Maps
- Usually 1D or 2D
- Each map unit associated with an R^n vector
- Unsupervised, Single layer, Feed-Forward network

© Miki Rubinstein

SOM algorithm

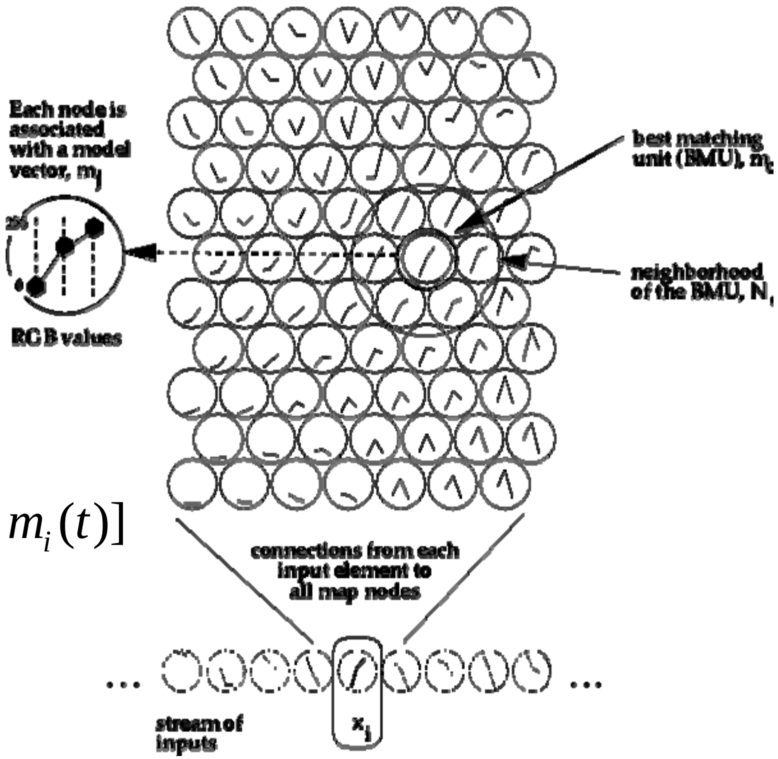
- Initialization
  - Random
  - Pattern
- For each sample vector n
  - Find winner, or BMU

$$c(n) = \arg\min_i \{\| n - m_i \|\}$$

  - Update rule:

$$m_i(t+1) = m_i(t) + h_{c(n),i}(t)\alpha(t)[n - m_i(t)]$$

Where $h_{c(n),i}$ is the neighborhood kernel and $\alpha(t)$ is the learning rate factor



© Miki Rubinstein

SOM visualization

- wsom

© Miki Rubinstein

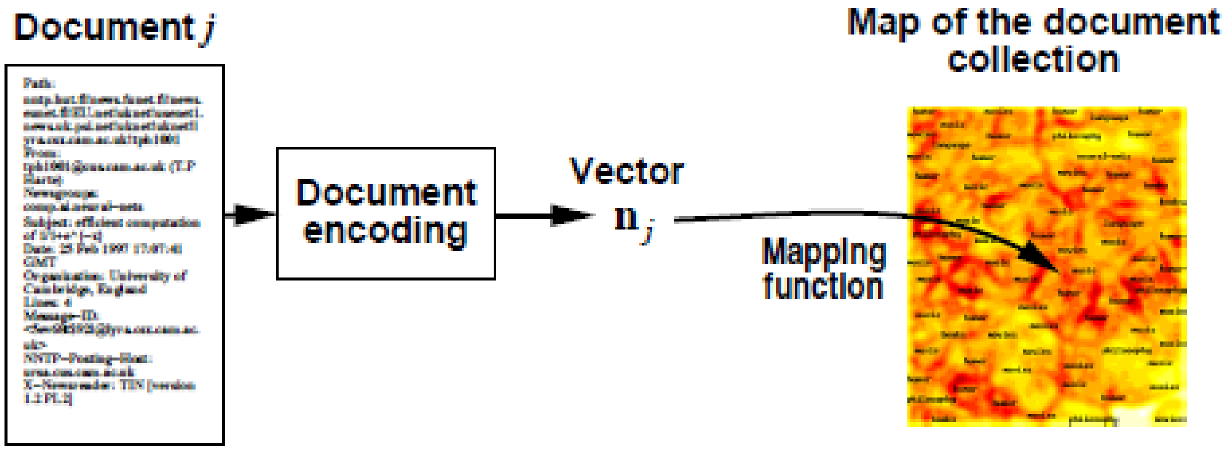
Back to Random Mapping

- SOM should not be too sensitive to distortions by random mapping
  - Small neighborhoods in R^n will be mapped to small neighborhoods in R^d ➔ will probably be mapped to single MU or a set of close-by MUs

© Miki Rubinstein

WEBSOM document organizing system

- Vector space model (Salton 1975)
  - Vectors are histograms of words
    - i'th element indicates (function of) frequency of the i'th vocabulary term in the document
  - Direction of vector reflects doc context



WEBSOM – example?

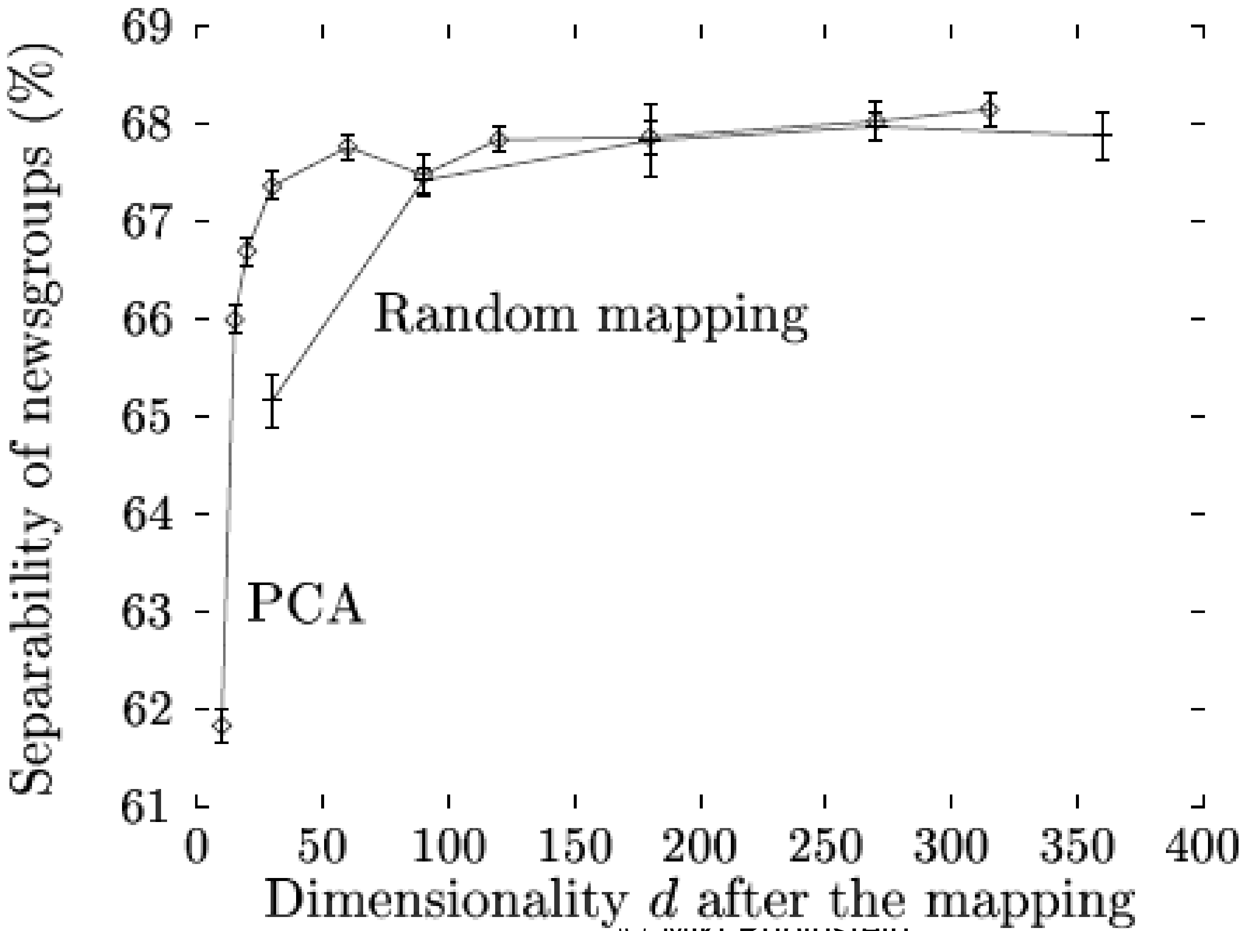
http://websom.hut.fi/websom/comp.ai.neural-nets-new/html/root.html

WEBSOM – experiment setup

- Intput
  - 18000 articles
  - 20 Usenet newsgroups
  - Different topic areas
- Vectorizing
  - After removing outliers → n = 5781
  - Each word weighted entropy based
- SOM
  - 768 MUs
  - MUs labeled according to dominated group
- Separability measure
  - Percentage of articles falling into MU labeled with their own class as majority
- 7 experiments for each dimension

© Miki Rubinstein

WEBSOM - results



© Miki Rubinstein

heuristics

- Distance metric: $\| x_1 - x_2 \| \Rightarrow \sqrt{n/d}\, \| Rx_1 - Rx_2 \|$
  - $\sqrt{n/d}$ = expected norm of projection of unit vector to random subspace through the origin (JL scaling term)
  - Image data

- Constructing R:
  - Set each entry of the matrix to an i.i.d. $N(0,1)$ value
  - Orthogonalize the matrix using the Gram-Schmidt algorithm
  - Normalize the columns of the matrix to unit length

© Miki Rubinstein

heuristics

- Achlioptas [2]:
  - Simpler distributions that are JL compatible

$$r_{ij} = \begin{cases} +1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}$$

$$r_{ij} = \sqrt{3} \cdot \begin{cases} +1 & \text{with probability } 1/6 \\ 0 & \text{with probability } 2/3 \\ -1 & \text{with probability } 1/6 \end{cases}$$

  - Only 1/3 of the operations

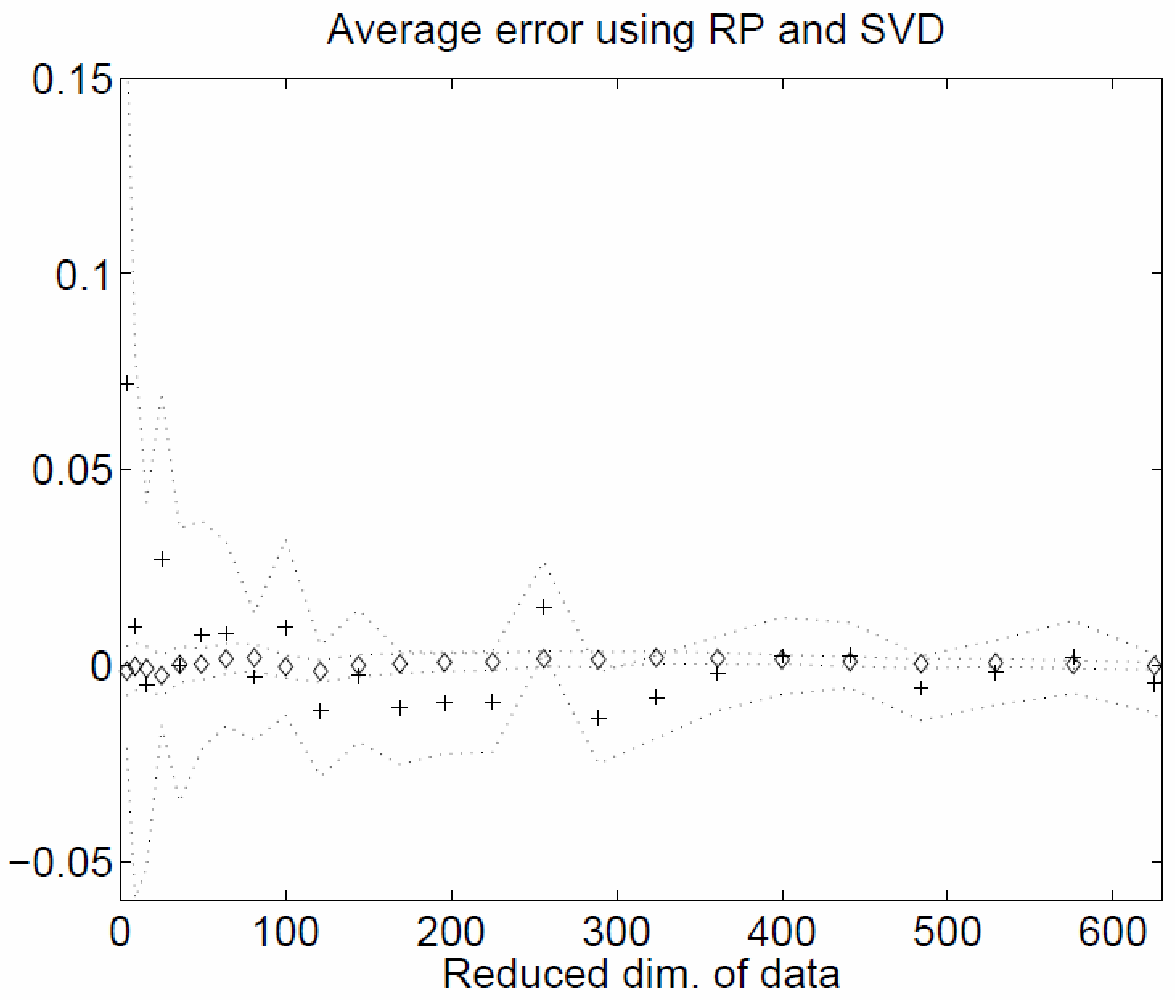
© Miki Rubinstein

RP vs. SVD - Bingham [10]

- n = 5000
- 2262 newsgroup documents
- Randomly chosen pairs of data vectors u, v
- Error = uv – (Ru)(Rv)
- 95% confidence intervals over 100 pairs of (u,v)

© Miki Rubinstein

RP vs. SVD - Bingham [10]



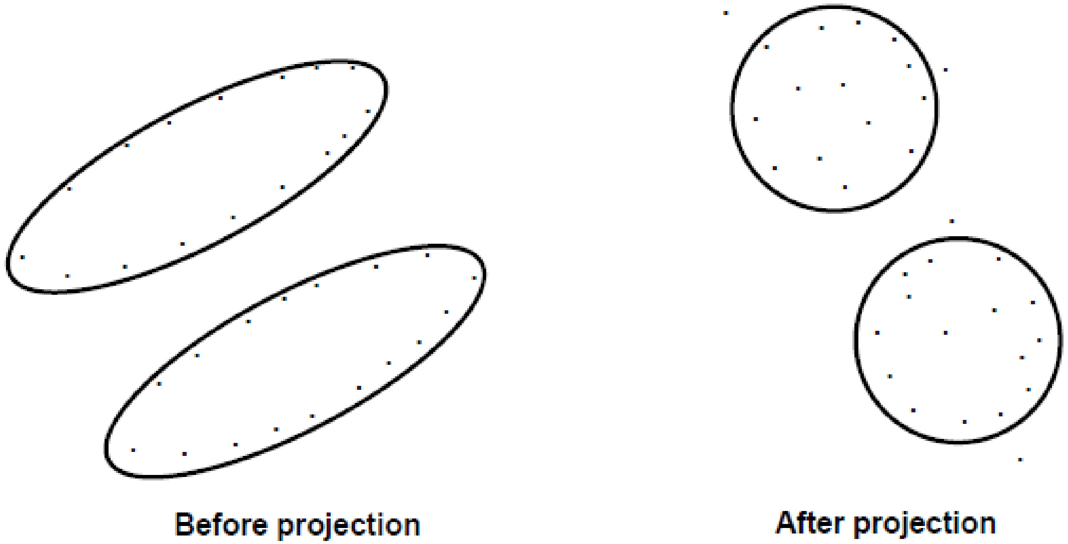
RP on Mixture of Gaussians

- data from a mixture of k Gaussians can be projected into O(logk) dimensions while still retaining the approximate level of separation between the clusters
  - Projected dimension independent of number of points and original dimension
  - Empirically shown for 10lnk
  - Decision of reduced dimension is highly studied
- Dasgupta [9] – for further details!

© Miki Rubinstein

RP on Mixture of Gaussians

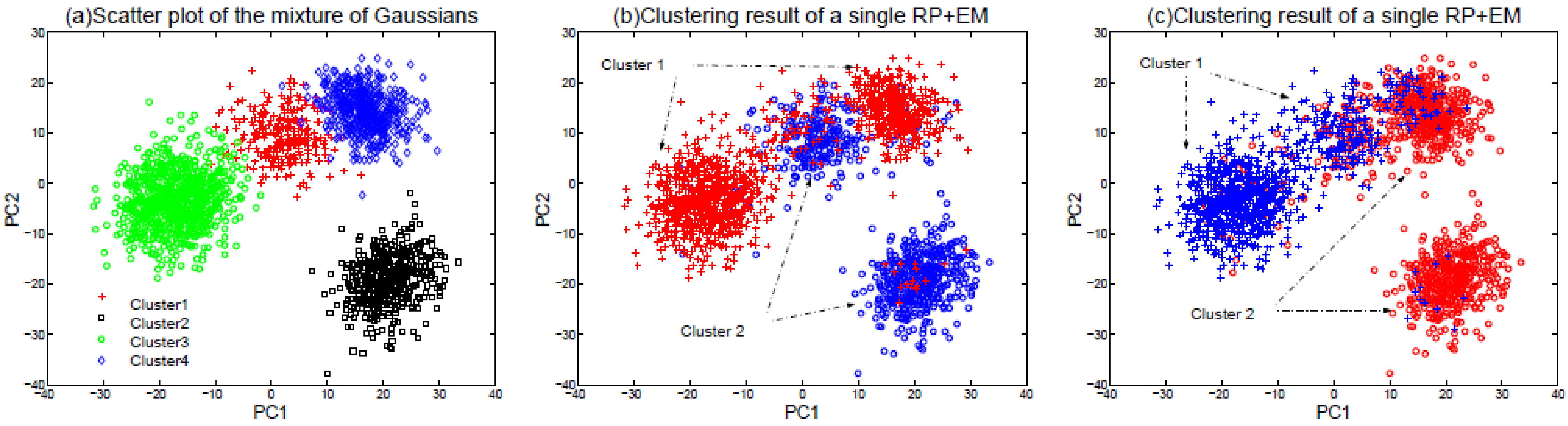
- The dimension is drastically reduced while eccentric clusters remain well separated and become more spherical



© Miki Rubinstein

RP ensembles – Fern [7]

- Experience of distorted, unstable clustering performance
- Different runs may uncover different parts of the structure in the data that complement one another



RP ensembles

- Multiple runs of RP + EM:
  - Project to lower subspace d
  - Use EM to generate a probabilistic model of a mixture of k Gaussians

$$P_{ij}^{\theta} = \sum_{l=1}^{k} P(l \mid i, \theta) P(l \mid j, \theta)$$

  - Average the P_{ij}s across n runs
  - Generate final clusters based on P

- Can iterate of different (reduced) subspaces
- Fern [7] - for more details!

© Miki Rubinstein

Face recognition with RP – Goal [11]

- Training set: M NxN vectors (each represents a face)

Algorithm:

1. compute average face:

$$\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i$$

2. Subtract mean face from each face:

$$\Phi_i = \Gamma_i - \Psi$$

3. Generate random operator R
4. Project normalized faces to random subspace:

$$w_i = R\Phi_i$$

© Miki Rubinstein

Face recognition with RP

Recognition:

1. Normalize
2. Project to same random space
3. Compare projection to database

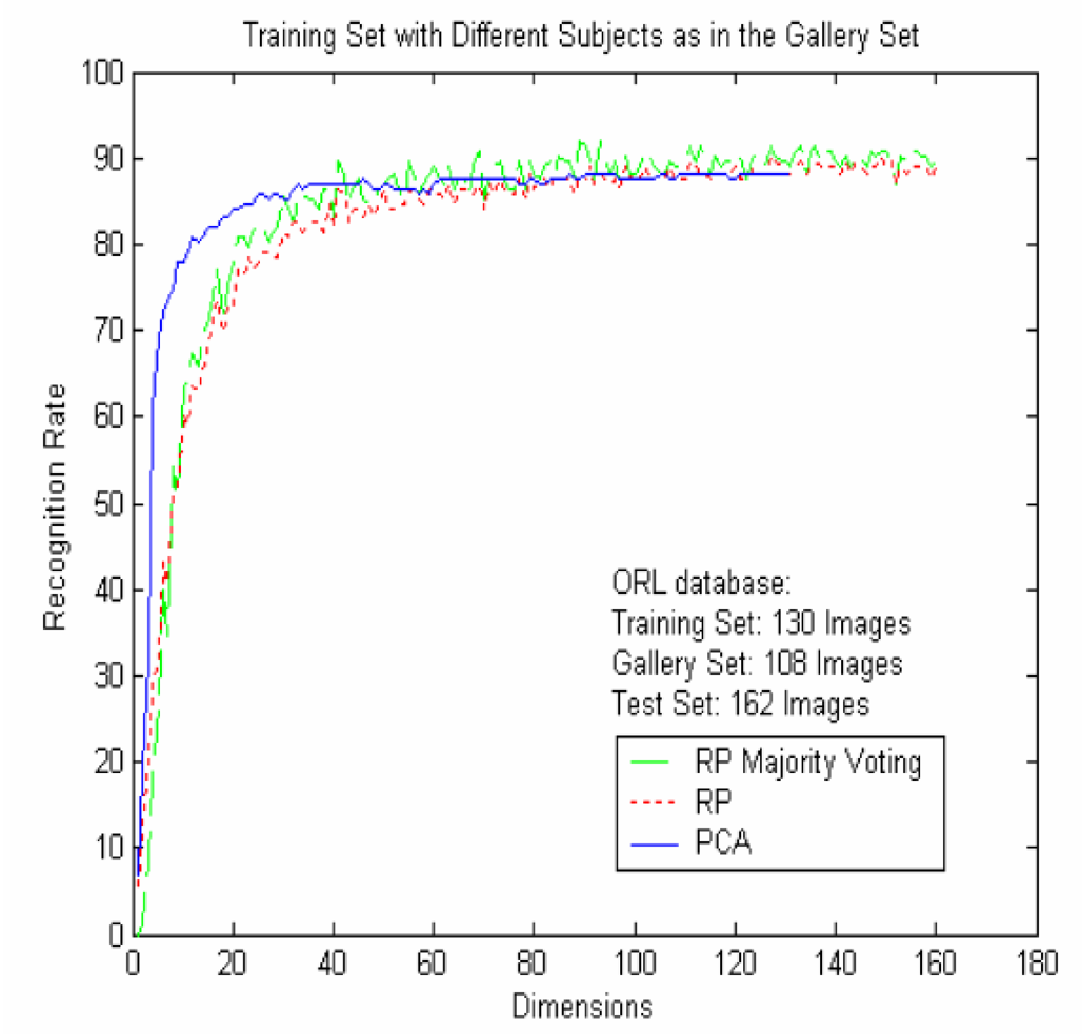
© Miki Rubinstein

Face recognition with RP

- Face representations need not be updated when face database changes
- Using ensembles of RPs seems promising
- Goel [11] – for more details!

© Miki Rubinstein

Face recognition with RP – example results



Conclusions

- Computationally much simpler
  - k data vectors, d << N
  - RP: O(dN) to build, O(dkN) to apply
    - If R has c nonzero entries: O(ckN) to apply
  - PCA: $O(kN^2)+O(N^3)$
- Independent of the data
- Has been applied on various problems and shown satisfactory results:
  - Information retrieval
  - Machine learning
  - Image/text analysis

© Miki Rubinstein

Conclusions

- Computation vs. Performance
- Bad results?
- Applying Johnson-Lindenstrauss on Kaski's setup yields k~2000 (?)

© Miki Rubinstein

?

Sanjoy Dasgupta

© Miki Rubinstein

References

- [1] S. Kaski. Dimensionality reduction by random mapping. In Proc. Int. Joint Conf. on Neural Networks, volume 1, pages 413–418, 1998.
- [2] D. Achlioptas. Database-friendly random projections. In Proc. ACM Symp. on the Principles of Database Systems, pages 274–281, 2001.
- [3] S. Dasgupta and A. Gupta. An elementary proof of the Johnson-Lindenstrauss lemma. Technical Report TR-99-006, International Computer Science Institute, Berkeley, California, USA, 1999.
- [4] R. Hecht-Nielsen. Context vectors: general purpose approximate meaning representations self-organized from raw data. In J.M. Zurada, R.J. Marks II, and C.J. Robinson, editors, Computational Intelligence: Imitating Life, pages 43–56. IEEE Press, 1994.
- [5] P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In Proc. of 30th STOC, 1998
- [6] D. Fradkin and D. Madigan. Experiments with random projections for machine learning. In KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 517--522, 2003

© Miki Rubinstein

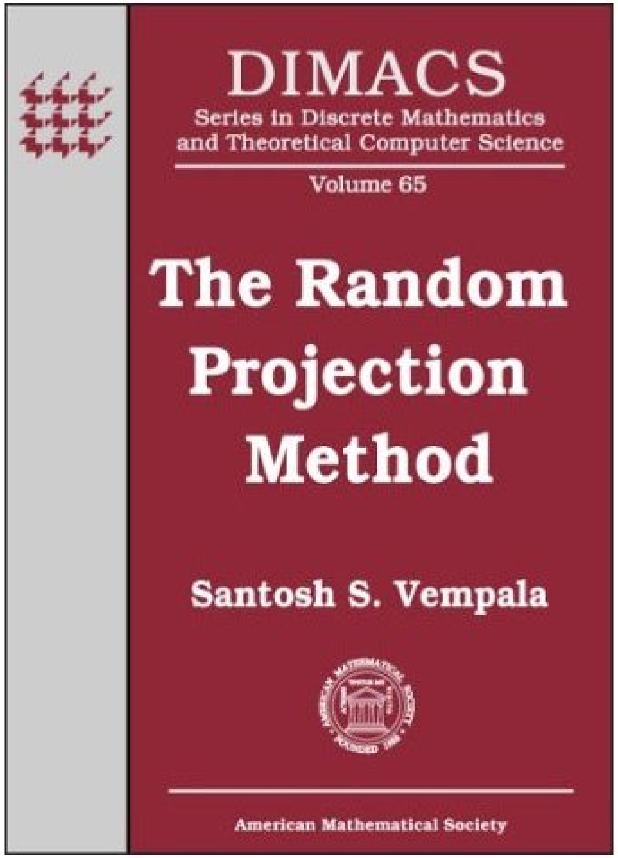
References

- [7] Xiaoli Z. Fern and Carla E. Brodley, "Cluster ensembles for high dimensional data clustering: An empirical study", Techenical report CS06-30-02
- [8] **C.H. Papadimitriou, P. Raghvan, H. Tamaki and S. Vempala, "Latent semantic analysis: A probabilistic analysis," in** ***Proceedings of 17th ACM Symp. On the principles of Database Systems*****, pp. 159–168, 1998.**
- [9] **Sanjoy Dasgupta, Experiments with Random Projection, Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, p.143-151, June 30-July 03, 2000**
- [10] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 245–250, 2001
- [11] N. Goel, G. Bebis, and A. Nefian. *Face recognition experiments with random projection*. In Proceedings SPIE Vol. 5779, pages 426--437, 2005

© Miki Rubinstein

References

Santosh S. Vempala (2004)
The Random Projection
Method



© Miki Rubinstein