Fast Distance Metrics in Low-dimensional Space for Neighbor Search Problems

Guihong Wan, Crystal Maung, Chenxu Zhang, Haim Schweitzer

Department of Computer Science

The University of Texas at Dallas

Contributions

- New formulas for improving the approximations of Euclidean distance and Mahalanobis distance in low-dimensional space

- The technique in which the new formulas are derived

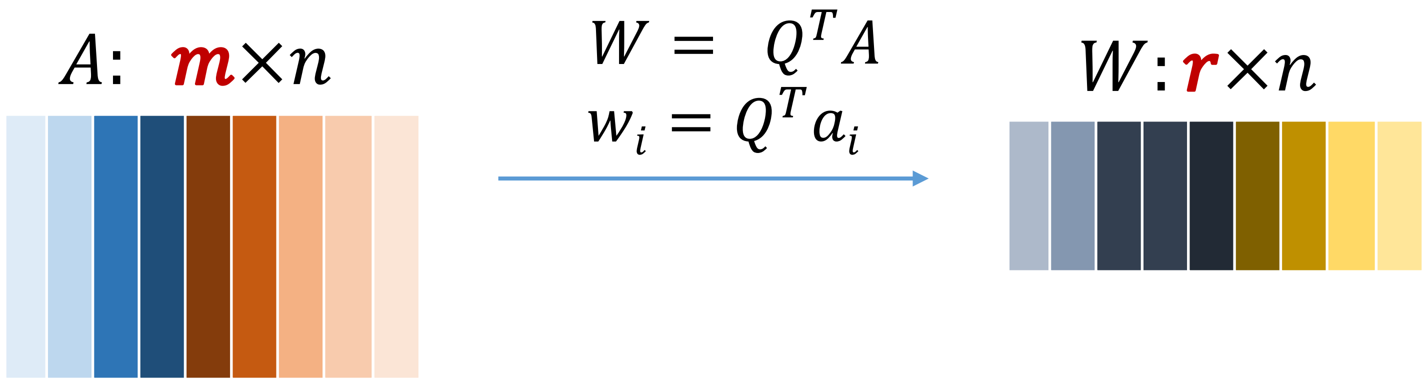
Outline

- Introduction
  - Dimension Reduction Techniques
  - Euclidean Distance
- Our Approach
  - Modeling the uncertainty
  - The maximum entropy method
  - Derivation of new formulas
- Experimental Results
  - k Nearest Neighbors
  - k Furthest Neighbors

Introduction

- Dimension Reduction Techniques
- Euclidean Distance

Dimension Reduction Techniques



- Linear dimension reduction techniques: from m to r
- $A \approx QW, \; a_i \approx Qw_i$
- $Q: m \times r$ is a matrix with orthogonal columns.

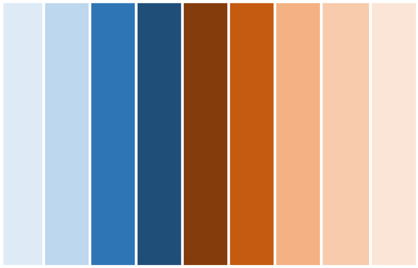
Dimension Reduction Techniques (cont.)

Three common choices for $Q_{m \times r}$:

1. r dominant left eigenvectors of A ← Principal Component Analysis (PCA)
2. r selected columns of A ← Column Subset Selection (CSS)
3. r vectors drawn from a Gaussian distribution with orthogonalization ← Johnson-Lindenstrauss (JL) random projections

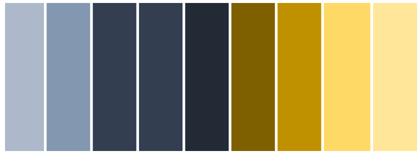
Dimension Reduction Techniques (cont.)

A: $\mathbf{m} \times n$



$$W = Q^T A$$
$$w_i = Q^T a_i$$

$W: \mathbf{r} \times n$



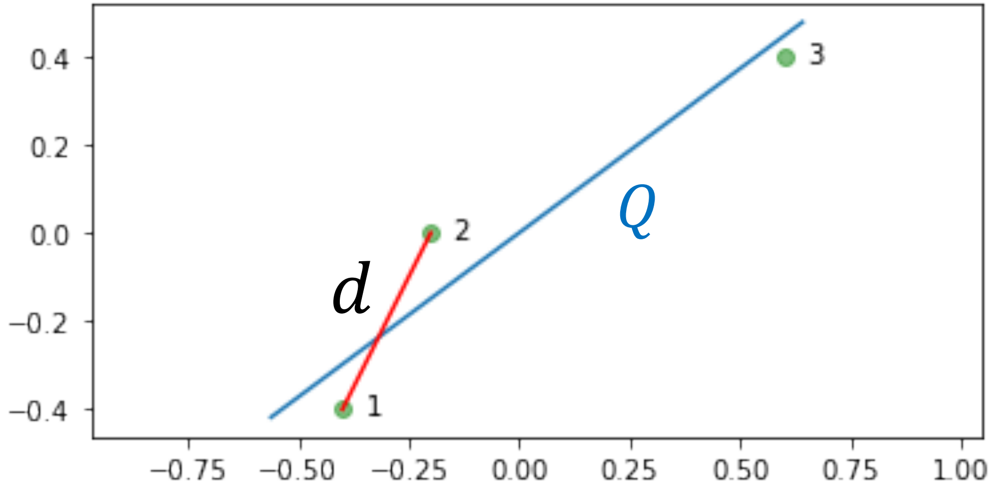
- W represents A in the low dimensional space
- W can be used for various approximations
- E.g. : Euclidean distance:

$$d^2(a_i, a_j) \approx d^2(w_i, w_j)$$

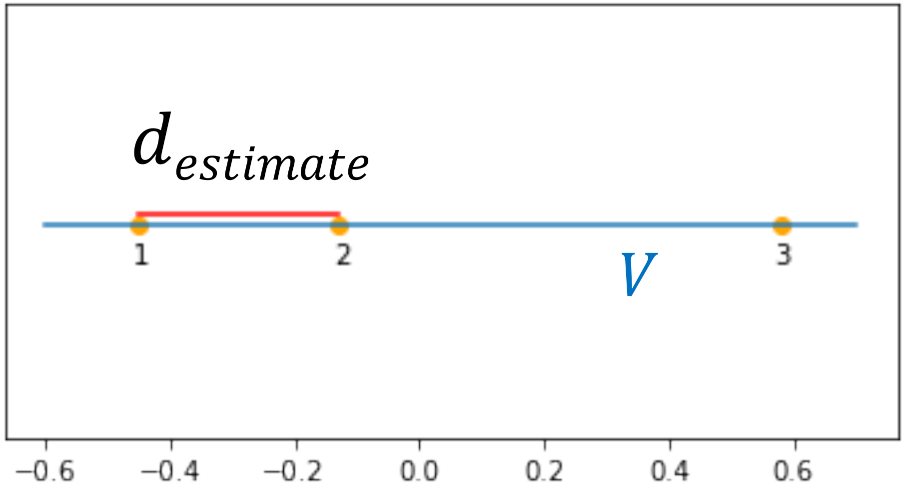
O(m) O(r)

Euclidean Distance

$$d^2(a_i, a_j) \approx d^2(w_i, w_j)$$



$$W = Q^T A$$



$$A = \begin{pmatrix} \boxed{\begin{matrix} -0.4 \\ -0.4 \end{matrix}} & \boxed{\begin{matrix} -0.2 \\ 0 \end{matrix}} & \boxed{\begin{matrix} 0.6 \\ 0.4 \end{matrix}} \end{pmatrix}$$

$$W = \begin{pmatrix} \boxed{0.56} & \boxed{0.16} & \boxed{-0.72} \end{pmatrix}$$

$$d = 0.45$$

$$d_{estimate} = 0.40$$

We show how to improve it.

Can we improve $d_{estimate}$?

- Classical solution: increasing r
- **Our solution**:

Model the uncertainty with random variables → Use the Maximum Entropy Method to infer the distribution → Take expectations to obtain deterministic formulas

Modeling the Uncertainty

- $A \approx Q_1 W_1, \ a^i \approx Q_1 w_1^i$
- Q_1: m×r has orthogonal columns. Can be extended to an orthogonal basis of $\mathbb{R}^m$.

Q_2: $\mathrm{m}\times(m-r)$ is such an extension.

$$A = Q_1 W_1 + Q_2 W_2, \quad a_i = Q_1 w_1^i + Q_2 w_2^i \qquad \text{<1>}$$

$$Q_1^T Q_1 = I, \quad Q_2^T Q_2 = I, \quad Q_1^T Q_1 + Q_2^T Q_2 = I$$

Observe: W_2 is unknown.

Modeling the Uncertainty (cont.)

$$A = Q_1 W_1 + Q_2 W_2, \qquad a_i = Q_1 w_1^i + Q_2 w_2^i$$

$$A \approx Q_1 W_1, \qquad a^i \approx Q_1 w_1^i$$

We propose to view W_2 as a random matrix with entries that are random variables:

$$\hat{A} = Q_1 W_1 + Q_2 \widehat{W}_2, \qquad \hat{a}_i = Q_1 w_1^i + Q_2 \widehat{w}_2^i \quad \text{<2>}$$

Problem: how to infer the probability distribution.

Our solution: use the Maximum Entropy Method.

The Maximum Entropy Method (MEM)

- MEM: a well-known technique for inferring probability distributions.
- When given constraints that the probability distribution must satisfy, the MEM asserts that:

the "most likely distribution" is the distribution with the largest entropy that satisfies the constraints.

Example

Consider coin flipping. Let (p_1, p_2) be the probability distribution.

Constraint: $p_1 + p_2 = 1$

What is the most likely probability distribution ?

Maximize entropy: $-p_1 \log(p_1) - p_2 \log(p_2)$

Subject to: $p_1 + p_2 = 1$

Solution: $p_1 = p_2 = \frac{1}{2}$

The Maximum Entropy Method (cont.)

Theorem 1: Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ be a random vector, where x_i are n random variables.

Given the correlation matrix $R = \mathrm{E}\{\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}\}$ (R is known), then according to the MEM, the probability density $f(\boldsymbol{x})$ and the entropy $H(\boldsymbol{x})$ are :

$$f(x) = \frac{1}{\sqrt{(2\pi)^n \Delta}} e^{-\frac{1}{2} x^t R^{-1} x}$$

$$H(x) = \ln \sqrt{(2\pi e)^n \Delta}$$

where Δ is the determinant of R.

A. Papoulis, *Probability, random Variables, and Stochastic Processes*, 2nd ed.

The Maximum Entropy Method (cont.)

$R = \mathrm{E}\{\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}\}$ is partially known

Missing parts can be determined by maximizing Δ.

$$\boxed{H(x) = \ln\sqrt{(2\pi e)^n \Delta}}$$

Hadamard's inequality:

$$\boxed{\Delta \leq R_{11} \ldots R_{nn}}$$

$$\begin{bmatrix} R_{11} & 0 & \ldots & 0 \\ 0 & R_{22} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & R_{nn} \end{bmatrix}$$

with equality iff R is diagonal.

A. Papoulis, *Probability, random Variables, and Stochastic Processes*, 2nd ed.

The Maximum Entropy Method (cont.)

Theorem 2: Let $W = (w_1, w_2, \ldots, w_n)$ be a random matrix of dimensions $r \times n$.

Given $z_i = \mathrm{E}\{||w_i||^2\}$, then according to the MEM:

1. All entries of the matrix W have 0 mean: $\mathrm{E}\{\mathrm{w_{ij}}\} = 0$
2. $\mathrm{w}_{i_1,\mathrm{j}_1}$ and $\mathrm{w}_{i_2,\mathrm{j}_2}$ are independent
3. $\mathrm{E}\{w_{ij}^2\} = \frac{z_i}{r}$
4. $f(W) = \frac{1}{\sqrt{(2\pi)^{rn}\Delta}} e^{-s(W)}$ where: $\Delta = \frac{\prod_{i=1}^{n} z_i^r}{r^{rn}}$, $s(W) = \frac{r}{2} \sum_{i,j} \frac{w_{i,j}^2}{z_i}$.

Derivation of New Formulas

The distance (squared) between a_i and a_j:

Exact : $d^2(a_i, a_j) = \, ||w_1^i - w_1^j||^2 + z_i + z_j - 2\left(w_2^i\right)^T w_2^j$

MEM : $d^2(a_i, a_j) \approx \, ||w_1^i - w_1^j||^2 + z_i + z_j$

Classical: $d^2(a_i, a_j) \approx \, ||w_1^i - w_1^j||^2$

Recall: $z_i = ||a_i||^2 - \, ||w_1^i||^2$

The new formula is much more accurate than the classical formula when a_i, a_j are nearly orthogonal.

Derivation of New Formulas (cont.)

Derivation:

$$\hat{a}_i = Q_1\, w_1^i + Q_2\, \hat{w}_2^i, \quad \hat{a}_j = Q_1\, w_1^j + Q_2\, \hat{w}_2^j$$

$$||\, \hat{a}_i - \hat{a}_j||^2 = \; ||{w_1}^i - {w_1}^j||^2 + ||\hat{w}_2^i||^2 + ||\hat{w}_2^j||^2 - 2\left(\hat{w}_2^i\right)^T \hat{w}_2^j$$

Expectation:

$$E\{||\, \hat{a}_i - \hat{a}_j||^2\} = \; ||w_1^i - w_1^j||^2 + z_i + z_j$$

Experimental Results

- k Nearest Neighbors
- k Furthest Neighbors

k Nearest Neighbors (KNN)

The error: $(\sum_{i=1}^{k} d_i)/k$

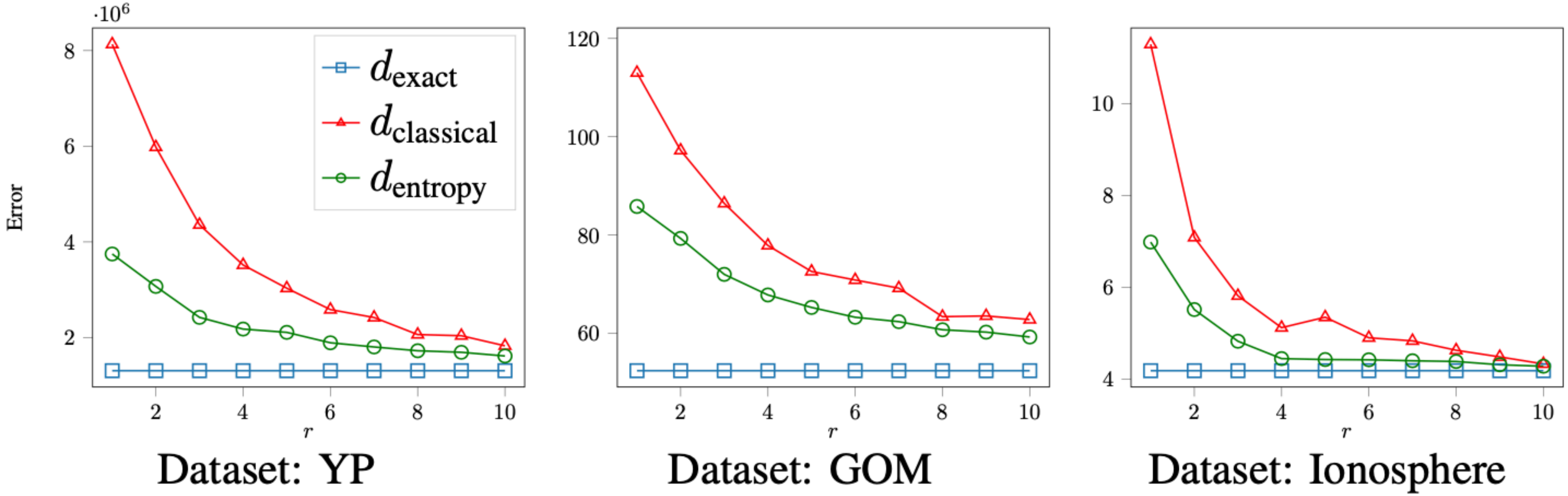


Figure 3: Error of KNN using PCA. k=10.

k Nearest Neighbors (cont.)

The recall: $[\text{COUNT}_{i=1}^{k}(d_i \leq d^*_{max})]/k$

Table VI: Recall for KNN using PCA on various datasets

$r : k$	YP		GOM		Ionosphere	
	$d_{\text{classical}}$	d_{entropy}	$d_{\text{classical}}$	d_{entropy}	$d_{\text{classical}}$	d_{entropy}
1 : 10	0	**0.005**	0.040	**0.080**	0.215	**0.310**
5 : 10	0.035	**0.060**	0.365	**0.390**	**0.735**	**0.735**
10 : 10	0.220	**0.235**	0.490	**0.570**	0.860	**0.885**
20 : 10	0.545	**0.560**	**0.705**	0.700	0.910	**0.955**
1 : 20	0	**0.005**	0.040	**0.108**	0.323	**0.478**
5 : 20	0.043	**0.073**	0.393	**0.470**	0.740	**0.838**
10 : 20	0.243	**0.280**	0.570	**0.648**	0.863	**0.915**
20 : 20	0.573	**0.625**	0.723	**0.770**	0.925	**0.958**

k Furthest Neighbors (KFN)

The ratio: $\sum_{i=1}^{k} d_i^* / d_i$

The recall: $[\mathrm{COUNT}_{i=1}^{k}(d_i \geq d_{min}^*)]/k$

methods		PCA				QRP				JL			
		$d_{\text{classical}}$		d_{entropy}		$d_{\text{classical}}$		d_{entropy}		$d_{\text{classical}}$		d_{entropy}	
	$r:k$	ratio	recall	ratio	recall	ratio	recall	ratio	recall	ratio	recall	ratio	recall
Linear Scan	5 : 10	1.0189	76.8%	**1.0019**	**91.4%**	1.0481	62.6%	**1.0056**	**87.2%**	1.1540	39.4%	**1.0252**	**77.4%**
	25 : 10	1.0011	92.8%	**1.0003**	**96.4%**	1.0043	85.0%	**1.0002**	**96.6%**	1.0366	65.6%	**1.0096**	**83.6%**
QDAFN	5 : 10	1.0189	76.8%	**1.0020**	**91.2%**	1.0481	62.6%	**1.0043**	**87.2%**	1.1537	39.4%	**1.0367**	**73.6%**
	25 : 10	1.0011	92.8%	**1.0003**	**96.4%**	1.0043	85.0%	**1.0002**	**96.6%**	1.0366	65.6%	**1.0096**	**83.6%**
Drusilla	5 : 10	1.0185	77.0%	**1.0019**	**91.4%**	1.0481	62.6%	**1.0056**	**87.2%**	1.1494	39.8%	**1.0264**	**77.2%**
	25 : 10	1.0013	92.0%	**1.0006**	**95.4%**	1.0045	84.8%	**1.0006**	**95.6%**	1.0369	65.6%	**1.0161**	**79.8%**
RQALSH	5 : 10	1.0193	76.8%	**1.0064**	**87.2%**	1.0538	60.6%	**1.0302**	**68.8%**	1.1542	39.0%	**1.0590**	**62.6%**
	25 : 10	1.0036	90.0%	**1.0029**	**92.8%**	1.0088	82.4%	**1.0051**	**92.4%**	1.0572	57.6%	**1.0430**	**63.0%**

Table IV: Improvement for KFN on GOM dataset

Thank You!

Email:guihong.wan@utdallas.edu