A Strict Total Order and Derived Distance Function for Multivariate Finite Data Based on Data-derived Analytical Meshes

Ray-Ming Chen
School of Mathematics and Statistics, Baise University, 21, Zhongshan No.2 Road, Guangxi Province, China
E-mail: baotaoxi@163.com

Abstract. In this article, we show a way to linearize a set of finite multivariate data set. Such linearization utilizes analytical meshes, which are defined based on a given data, and associated forward walking paths related to the defined meshes. It further yields a strict linear ordering and a simple distance function via such walking paths and lengths for the given data set. This strict linear ordering and distance function could be applied in grouping the data set with respect to some optimal criteria.

1. Introduction
Data analysis exploits factor analysis, clustering analysis, attributes analysis, etc [1]. For factor analysis, one could apply data screening and assessment of some key assumptions for the real data analysis [2]. As for clustering, there are many techniques and indexes for data analysis [3, 4, 5]. A much more mentioned approach for multivariate data analysis is Principal Component Analysis [6, 7]. A survey for all sorts of data analysis for data structures could consult Dempster’s article [8]. Unlike the above-mentioned articles, we contrive a new method for multivariate data analysis. We call each finite real multivariate set $D$ becomes a vital problem in data analysis. Firstly, we define an analytical mesh based on the given $D$. Such mesh would indicate the relative positions of all the elements in $D$. Secondly, based on this defined mesh, we locate the elements and calculate their individual forward walking paths and lengths. $|S|$ and $P(S)$ denote the size and the power set of set $S$; $i(p)$ and $|\vec{v}|$ denote the $p$-th element and the size of vector $\vec{v}$, respectively. Let $\sigma_A : \{1, 2, ..., |A|\} \rightarrow A$ be an ascending sort function of a set $A$ and $\sigma_A^{-1}$ be its inverse. Let $\chi : \mathbb{N} \rightarrow \{0, 1\}$ by $\chi(k) = 1$, if $k$ is odd, and $\chi(k) = 0$, if $k$ is even. Let $\mathbb{D} = \bigcup_{j \in \mathbb{N}} \mathbb{D}_j$ and $\mathbb{R} = \bigcup_{j \in \mathbb{R}_j} \mathbb{R}_j$. Let $D \in \mathbb{D}$ be arbitrary. Let $\vec{v} \in \vec{n}$ be arbitrary. Let $\bar{\mathbb{D}}(a, b)$ denote the quotient of $a$ divided by $b$ and $\bar{R}(a, b)$ denote the remainder of $a$ divided by $b$. Let $\bar{n} = \{1, 2, ..., n\}$, $\bar{n} = \{1, 2, ..., n\}$, $\bar{n} - i = \{1, 2, ..., i - 1, i + 1, ..., n\}$. Let $\bar{n}^2 = \{(k_1, k_2) : k_1 \neq k_2\}$, $\bar{n}^2 = \{(k_1, k_2, ..., k_k) : \bar{n} : \forall i, j \in \bar{n}, k_i \neq k_j\}$. Let $[n^n] = \bar{n}^1 \cup \bar{n}^2 \cup ... \bar{n}^n$.  

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Published under licence by IOP Publishing Ltd
2. Definitions
Let \( D_2 \in \mathbb{D}_2, D_3 \in \mathbb{D}_3 \) and \( D_n \in \mathbb{D}_n \) be arbitrary. Let \( \vec{v} \in \vec{n}^n \) be arbitrary.

- (Projections) Define \( Proj^j : \mathbb{D}^\infty \to \mathcal{P}(\mathbb{R}^\infty) \) by \( Proj^j(D) := \{ c_i \in \mathbb{R} : \exists c_1, c_2, ..., c_{i-1}, c_i, c_{i+1}, ..., c_D \in \mathbb{R} \text{ such that } (c_1, c_2, ..., c_{i-1}, c_i, c_{i+1}, ..., c_D) \in D \} \).

- (Generalized Projections) \( Proj^{ij}(D) := \{ \vec{w} \in \mathbb{R}^{[n]} \mid \exists Y \in \mathcal{D} \text{ such that } \vec{w} = (Y(\vec{v}(1)), Y(\vec{v}(2)), ..., Y(\vec{v}(n))) \} \) for all \( \vec{v} \in [n]^n \), in particular \( Proj^{ij}(D) := \{ c_{\vec{v}(i)} \in \mathbb{R} : \exists c_{\vec{v}(1)}, c_{\vec{v}(2)}, ..., c_{\vec{v}(i-1)}, c_{\vec{v}(i+1)}, ..., c_{\vec{v}(n)} \in \mathbb{R} \text{ such that } (c_{\vec{v}(1)}, c_{\vec{v}(2)}, ..., c_{\vec{v}(i-1)}, c_{\vec{v}(i)}, c_{\vec{v}(i+1)}, ..., c_{\vec{v}(n)}) \in D \} \).

- (Maximum, Minimum) Let \( \max Proj^j(D) \) and \( \min Proj^j(D) \) denote the maximum and minimum of \( Proj^j(D) \), respectively.

We use \( \vec{v}_k(D) \) or simply \( \vec{v}_k \) (when \( D \) is understood from the context) to denote the set \( Proj^{ij}(D) \). Let \(|\vec{v}_k|||\vec{v}_k|||\) and \( |\vec{v}_k| \) denote the cardinality \( |Proj^{ij}(D)| \), the value of the length \( max Proj^{ij}(D) \), and the real-valued interval \( [\min Proj^{ij}(D), \max Proj^{ij}(D)] \subseteq \mathbb{R} \), respectively.

For all \( a \in Proj^j(D) \), define \( a^- := \{ c \in Proj^j(D) : c < a \} \) and \( a^+ := \{ c \in Proj^j(D) : c > a \} \).

### 2.1. Data-derived Analytical Meshes

**Definition 2.1.** Define \( Mesh : \mathbb{D}^\infty \to \mathcal{P}(\mathbb{R}^\infty) \) by

- \( Mesh(D_2) := \{(x_1, x_2) \in \mathbb{R}^2 : \min Proj^j(D_2) \leq x_1 \leq \max Proj^j(D_2), x_2 \in Proj^2(D_2)\} \) \( \cup \{ (x_1, x_2) \in \mathbb{R}^2 : \min Proj^2(D_2) \leq x_2 \leq \max Proj^2(D_2), x_1 \in Proj^1(D_2) \} \).

- \( Mesh(D_3) := \{(x_1, x_2, x_3) \in \mathbb{R}^3 : \min Proj^j(D_3) \leq x_1 \leq \max Proj^j(D_3), (x_2, x_3) \in Proj^{2,3}(D_3) \} \cup \{ (x_1, x_2, x_3) \in \mathbb{R}^3 : \min Proj^{2,3}(D_3) \leq x_2 \leq \max Proj^{2,3}(D_3), (x_1, x_3) \in Proj^{1,3}(D_3) \} \cup \{ (x_1, x_2, x_3) \in \mathbb{R}^3 : \min Proj^{1,2,3}(D_3) \leq x_3 \leq \max Proj^{1,2,3}(D_3), (x_1, x_2) \in Proj^{1,2}(D_3) \} \).

- (Generalized Meshes) \( Mesh(D_n) = \bigcup_{i=1}^{n} \{(x_1, x_2, ..., x_n) \in \mathbb{R}^n : \min Proj^j(D_n) \leq x_i \leq \max Proj^j(D_n), (x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n) \in Proj^{\vec{v}_i}(D_n) \} \).

Based on this \( D_n \)-derived analytical mesh, we could then trace each element in \( D_n \) via forward walking starting from the point \( (\min Proj^{(1)}(D_n), \min Proj^{(2)}(D_n), ..., \min Proj^{(n)}(D_n)) \) to the point \( (\max Proj^{(1)}(D_n), \max Proj^{(2)}(D_n), ..., \max Proj^{(n)}(D_n)) \) along the direction \( \vec{v} \). The role of \( \vec{v} \) is to pin down the axes for the trace of elements.

**Definition 2.2.** (Analytical-mesh Coordinates) Given \( D_n \in \mathbb{D}_n \), and \( \vec{v} \in \vec{n}^n \). For any \( \vec{A} \in D_n \), define the analytical-mesh coordinates of \( \vec{A} \) by \( \vec{A}_{\vec{v}} := \vec{A} \vec{v} := (\sigma_{\vec{v}_1}^{-1}(\vec{A}(\vec{v}(1))), \sigma_{\vec{v}_2}^{-1}(\vec{A}(\vec{v}(2))), ..., \sigma_{\vec{v}_n}^{-1}(\vec{A}(\vec{v}(n))) \).

**Claim 1.** For all \( \vec{A}, \vec{B} \in D_n(\vec{A}_{\vec{D},\vec{v}}) \neq (\vec{B}_{\vec{D},\vec{v}}) \) if and only if \( \vec{A} \neq \vec{B} \) for all \( \vec{v} \in \vec{n}^n \).

Define \( X_{\mathbb{R},\emptyset} : \mathbb{N} \to \{\mathbb{R}, \emptyset\} \) by \( X_{\mathbb{R},\emptyset}(k) := \mathbb{R} \), if \( k \) is odd; \( \emptyset \), if \( k \) is even.

**Definition 2.3.** (Cubic Walking Path) Define a complete forwarding walking path (or cubic walking path, CWP) along the direction \( \vec{v} \) inductively as follows:

- \( CWP^1_1 = [\vec{v}_1] \);

- \( CWP^2_1 = [CWP^1_1 \times \vec{v}_2] \cup \bigcup_{i=1}^{Q(|\vec{v}_2|+1.2)} \{\max Proj^{(1)}(D_n)\times[\sigma_{\vec{v}_2}(2i-1), \sigma_{\vec{v}_2}(2i)] \}

- \( CWP^1_2 = [CWP^1_1 \times \vec{v}_2] \cup \bigcup_{i=1}^{Q(|\vec{v}_2|-1.2)} \{\min Proj^{(1)}(D_n)\times[\sigma_{\vec{v}_2}(2i), \sigma_{\vec{v}_2}(2i+1)] \}; \)
Definition 2.4. (Cubic Walking Length)

\[
\text{Definition 2.4. (Cubic Walking Length)}
\]

\[
\text{Lemma 2.1.}
\]

\[
\text{Proof.}
\]

\[
\text{Example 1.}
\]

Each CWP records the trace of a complete forward walking along the defined analytical meshes under all sorts of dimensions. The lengths of the above cubic walking paths could be defined correspondingly as follows:

**Definition 2.4.** (Cubic Walking Length)

- \(||CW_{\bar{v}}P^2_1|| = ||\bar{v}_1||
- \(||CW_{\bar{v}}P^2_2|| = ||CW_{\bar{v}}P^2_1|| \cdot ||\bar{v}_2|| + ||\bar{v}_3||
- \ldots
- \(||CW_{\bar{v}}P^2_{k+1}|| = ||CW_{\bar{v}}P^2_k|| \cdot ||\bar{v}_{k+1}|| + ||\bar{v}_{k+1}|| \text{ for all } k + 1 \leq ||\bar{v}||

**Lemma 2.1.** \(||CW_{\bar{v}}P^2_k|| = \sum_{h=1}^{k} ||\bar{v}_h|| \cdot ||\bar{v}_{h+1}|| \cdot \ldots \cdot ||\bar{v}_k||

**Proof.** We show this by mathematical induction. For \(k = 2\), the result follows immediately from Definition 2.4. Moreover, by assumption, \(||CW_{\bar{v}}P^2_{k+1}|| = ||CW_{\bar{v}}P^2_k|| \cdot ||\bar{v}_{k+1}|| + ||\bar{v}_{k+1}|| = \sum_{h=1}^{k} ||\bar{v}_h|| \cdot ||\bar{v}_{h+1}|| \cdot \ldots \cdot ||\bar{v}_k|| \cdot ||\bar{v}_{k+1}|| + ||\bar{v}_{k+1}||\)

**Example 1.** Suppose \(\bar{v} = (1, 2, 3, 4)\). Let \(D_4 \in \mathbb{D}_4\) be arbitrary. If \(||\bar{v}_4||\) is odd, then the (continuous) forward walking path is shown as the left figure and if \(||\bar{v}_4||\) is even, then the forward walking path is shown as the right figure in Table 1.

Now we could trace any given point in \(D_n\) along the cubic walking path which is determined by the direction \(\bar{v}\). Let \(\bar{v}_n\) denote \(min_{Prj_{\bar{v}}^{(i)}(D_n)}\) and \(\bar{v}_n^*\) denote \(max_{Prj_{\bar{v}}^{(i)}(D_n)}\). Let \(D_3^{(1,2)}\) denote \(Prj_{\bar{v}}^{(j(1),j(2))}(D_3)\) and \(\bar{A}_{\bar{v}}^{(1,2)}\) denote \(Prj_{\bar{v}}^{(i(1),i(2))}(\bar{A})\).

**Definition 2.5.** (Point Walking Path)

- Suppose \((\bar{A})_{D_2,\bar{v}} = (a_1, a_2)\), where \(\bar{v} \in \mathbb{Z}^2\).
Definition 2.6. (Point Walking Length)

- Suppose \((\vec{A}_{(D_3, \vec{v})}) = (a_1, a_2, a_3)\), where \(\vec{v} \in \mathbb{R}^3\).
  
  \[ PW_{D_3}^{\vec{v}} (\vec{A}) = [CW_{D_3}^{\vec{v}} \times a_3] \]
  
  \[
  Q_{t(a_3+1,2)} \]
  
  \[
  \bigcup_{i=1}^{1} \{ (\vec{v}_1^*, \vec{v}_2^*) \} \times [\sigma_{\bar{v}_1}(2i-1), \sigma_{\bar{v}_1}(2i)] \]
  
  \[
  Q_{t(a_3-1,2)} \]
  
  \[
  \bigcup_{i=1}^{1} \{ (\vec{v}_1^*, \vec{v}_2^*) \} \times [\sigma_{\bar{v}_1}(2i), \sigma_{\bar{v}_1}(2i + 1)] \]
  
  \[
  \cup [(x_{R, \emptyset}(a_3) \cap PW_{D_{3}'}^{\vec{v}}(\vec{v})_1) \cup (x_{R, \emptyset}(a_3 + 1) \cap (CW_{D_3}^{\vec{v}} - PW_{D_{3}'}^{\vec{v}}(\vec{v})_1))] \]

- Suppose \((\vec{A}_{(D_{k+1}, \vec{v})}) = (a_1, a_2, ..., a_{k+1})\), where \(\vec{v} \in \mathbb{R}^{k+1}\).
  
  \[ PW_{D_{k+1}}^{\vec{v}} (\vec{A}) = [CW_{k}^{\vec{v}} \times a_{k+1}] \]
  
  \[
  Q_{t(a_{k+1}+1,2)} \]
  
  \[
  \bigcup_{i=1}^{1} \{ (\vec{v}_1^*, \vec{v}_2^*, ..., \vec{v}_k^*) \} \times [\sigma_{\bar{v}_k+1}(2i-1), \sigma_{\bar{v}_k+1}(2i)] \]
  
  \[
  Q_{t(a_{k+1}-1,2)} \]
  
  \[
  \bigcup_{i=1}^{1} \{ (\vec{v}_1^*, \vec{v}_2^*, ..., \vec{v}_k^*) \} \times [\sigma_{\bar{v}_k+1}(2i), \sigma_{\bar{v}_k+1}(2i + 1)] \]
  
  \[
  \cup [(x_{R, \emptyset}(a_{k+1}) \cap PW_{D_{k+1}}^{\vec{v}}(\vec{A}_1') \cup (x_{R, \emptyset}(a_{k+1} + 1) \cap (CW_{D_3}^{\vec{v}} - PW_{D_{k+1}}^{\vec{v}}(\vec{A}_2')))] \]
  
  where \(\vec{A}_1' = Prj_{\vec{v}}^{\vec{v}-(k+1)}(\vec{A})\) and \(D_{k+1}' = Prj_{\vec{v}}^{\vec{v}-(k+1)}(D_{k+1})\).

Now we could compute the corresponding lengths via following definitions.

**Figure 1.** Continuous Forward Walking.
and a distance function $d_{\vec{A}}$ length of $(\vec{A})$ information for the data set $D$ dimension data and linearize any finite data set. Such linearization provides some insightful directions. Then given every walking direction $\vec{v}$ directions to connect the elements in $D$. We associate each data set with an analytical mesh. This mesh provides the possible directions to connect the elements in $D_n$. In this article, we consider only the forward walking directions. Then given every walking direction $\vec{v}$, we could then locate each element $\vec{A}$ in $D_n$ by $(\vec{A})_{D_n,\vec{v}}$. With such location, one could further indicate the trace of the walking path and its length of $\vec{A}$. Through the definition of the lengths, one could construct a strict total ordering.

**Example 2.** Let us continue Example ?? and suppose $\vec{v} = (1, 2)$. Then $|CPW_{\vec{v}}(\vec{A})| = w - a, \sigma_{v_2} - \sigma_{v_2}(1) = i - v$. Hence $|CPW_{\vec{v}}(\vec{A})| = (w - a) \cdot 3 + (i - v) + ((w - a) - (a - p)) = 4w - 3a + i - v - p$.

**Lemma 2.2.** $PW_{\vec{v}}(\vec{A}) \subseteq PW_{\vec{v}}(\vec{B})$ if and only if $|PW_{\vec{v}}(\vec{A})| \leq |PW_{\vec{v}}(\vec{B})|$.

**Corollary 1.** $PW_{\vec{v}}(\vec{A}) = PW_{\vec{v}}(\vec{B})$ if and only if $|PW_{\vec{v}}(\vec{A})| = |PW_{\vec{v}}(\vec{B})|$.

Define $ds : D_n \rightarrow \mathbb{R}^+$ by $ds(\vec{A}, \vec{B}) := |PW_{\vec{v}}(\vec{A})| - |PW_{\vec{v}}(\vec{B})|$.

**Theorem 2.3.** $(D_n, ds)$ is a metric space.

3. Conclusions
We associate each data set $D_n$ with an analytical mesh. This mesh provides the possible directions to connect the elements in $D_n$. In this article, we consider only the forward walking directions. Then given every walking direction $\vec{v}$, we could then locate each element $\vec{A}$ in $D_n$ by $(\vec{A})_{D_n,\vec{v}}$. With such location, one could further indicate the trace of the walking path and its length of $\vec{A}$. Through the definition of the lengths, one could construct a strict total ordering and a distance function $ds$ on $D_n$. This article has a strong analytical ability to handle high-dimensional data and linearize any finite data set. Such linearization provides some insightful information for the data set $D_n$ and is very efficient in grouping or categorizing the data set via some optimal criteria.

4. References
[1] Joseph F. Hair Jr., William C. Black, Barry J. Babin, and Rolph E. Anderson. 2014. Multivariate Data Analysis, Seventh Edition, Pearson Education Limited.
[2] James W. Frane, and Maryann Hill. 1976. “Factor analysis as a tool for data analysis”, Communications in Statistics - Theory and Methods.
[3] Dongkuan Xu, and Yingjie Tian. 2015. A Comprehensive Survey of Clustering Algorithms. 2(2):165-193.
[4] Asa Ben-Hur, David Horn, Hava T. Siegelmann, and Vladimir Vapnik. 2001. “Support Vector Clustering”, Journal of Machine Learning Research. 2: 125-137.
[5] Sebastien Bubeck, and Ulrike von Luxburg. 2009. “Nearest Neighbor Clustering: A Baseline Method for Consistent Clustering with Arbitrary Objective Functions”, Journal of Machine Learning Research. 10: 657-698
[6] Glassy J. 2013. “Multivariate data analysis for advancing the interpretation of bioprocess measurement and monitoring data”, Adv Biochem Eng Biotechnol. 132:167-91.
[7] Ray-Ming Chen. 2014. “PCA-based algorithmic approximation of crisp target sets.” UKCI 1-8.
[8] A.P.Dempster. 1971. “An overview of multivariate data analysis”, Journal of Multivariate Analysis 1(3): 316-346.

Acknowledgments
This work is supported by the Humanities and Social Science Research Planning Fund Project under the Ministry of Education of China (Grant No. 20XJAGAT001).