Data Mining: for What? Clusters or Factors

by

Moody T. Chu

(Joint work with Bob Funderlic)

North Carolina State University

presented at
National Center of Theoretical Sciences
January 17, 2003

Outline

- Introduction
- Factor Analysis
  - ◇ Linear Model
  - ◇ Latent Semantic Indexing
- Cluster Analysis
  - ◇ Factors versus Clusters
- Centroid Method
  - ◇ Modified Centroid
  - ◇ Centroid Decomposition
  - ◇ Integer Programming on Hypercubes
- Truncated Data
  - ◇ Statistical Meaning of SVD
  - ◇ Truncation and Lower Rank Approximation
  - ◇ Centroid versus SVD

Introduction

- Data mining is about extracting interesting information from raw data.
- What constitutes “information”?
  - ◇ Patterns of appearance.
  - ◇ Association rules between sets of items.
  - ◇ Clustering of the data points.
  - ◇ Concepts or categories.
  - ◇ Principal components or factors.
  - ◇ . . .
- What should be counted as “interesting”?
  - ◇ Confidence and support.
  - ◇ Information content.
  - ◇ Unexpectedness.
  - ◇ Actionability — The ability to suggest concrete and profitable decision-making.
  - ◇ . . .
- For different information retrievals, different techniques should be used.
  - ◇ Factors — Rank reduction or lower dimension approximation.
  - ◇ Clusters — Centroids or *k*-means.
  - ◇ . . .

Factor Analysis

- Data analysis:
  - ◇ An indispensable task in almost every discipline of science.
  - ◇ Search for relationships between a set of externally caused and internal variables.
  - ◇ Especially important in this era of information and digital technologies when massive amounts of data are generated at almost all levels of applications.
- Data observed from complex phenomena:
  - ◇ Often represent the integrated result of several interrelated variables acting together.
  - ◇ These variables sometimes are less precisely defined.
  - ◇ What to distinguish which variable is related to which and how the variables are related.
- Factor analysis:
  - ◇ A class of procedures that can help identify and test what *constructs* might be used to explain the interrelationships among the variables.
  - ◇ Each construct itself is a complex image, idea, or theory formed from a number of simpler elements.

Basic Model

- Making observation, data gathering and processing:
  - ◇ Assume ℓ entities and n variable.
  - ◇ Record raw scores that entity j received from all variables.
  - ◇ Normalize raw scores to have mean 0 and standard deviation 1 (standardized score).
  - ◇ Let $Y = [y_{ij}] \in \mathbb{R}^{n\times\ell}$ denote the matrix of observed data.
    - ▷ y_{ij} = *standard score* of entity j on variable i.
  - ◇ Correlation matrix of all n variables:

$$R := \frac{1}{\ell} YY^T. \tag{1}$$

- Linear model:
  - ◇ Assume that y_{ij} is a linearly weighted score of entity j on several factors.

$$Y = AF. \tag{2}$$

  - ◇ $A = [a_{ik}] \in \mathbb{R}^{n\times m}$ is the factor loading matrix.
    - ▷ a_{ik} = the loading of variable i on factor k, or the influence of factor k on variable i.
  - ◇ $F = [f_{kj}] \in \mathbb{R}^{m\times\ell}$is the factor scoring matrix.
    - ▷ f_{kj} = the score of factor k on entity j, or the response of entity j to factor k.

An Example

- Each of the ℓ columns of the observed matrix Y represents the transcript of a college student (an entity) at his/her freshman year on n fixed subjects (the variables), e.g., Calculus, English, Chemistry, and so on.
- It is generally believed that a college freshman's academic performance depends on a number of factors including, for instance, family social status, finance, high school GPA, cultural background, and so on.
- Upon entering the college, each student could be asked to fill out a questionnaire inquiring these factors of his/her background. In turn, individual responses to those factors are translated into scores and placed in the corresponding column of the scoring matrix F.
- What is not clear to the educators/administrators is how to choose the factors to compose the questionnaire or how each of the chosen factors would be weighted (the loadings) to reflect the effect on each particular subject.
- In practice, we usually do not have a priori knowledge about the number and character of underlying factors in A. Sometimes we do not even know the factor scores in F.
- Only the data matrix Y is observable.
- Explaining the complex phenomena observed in Y with the help of a minimal number of factors extracted from the data matrix is the primary and most important goal of factor analysis.

Factor Analysis and Matrix Decompositin

- Two additional assumptions:
  - ◇ All sets of factors being considered are uncorrelated with each other.
  - ◇ Similar to Y, the scores in F for each factor are normalized.

$$\frac{1}{\ell} F F^T = I_m. \tag{3}$$

- The correlation matrix R can be expressed directly in terms of the loading matrix A, i.e.,

$$R = AA^T. \tag{4}$$

  - ◇ Factor extraction now becomes a problem of decomposing the correlation matrix R into the product AA^T.
  - ◇ Would like to use as few factors as possible.
- a_{i*} = how the data variable i is weighted across/influenced by the list of current factors.
  - ◇ $\|a_{i*}\|_2^2$ = the *communality* of variable i.
    - ▷ If $\|a_{i*}\|_2$ is small, variable i is of little consequence to the current factors.
- a_{*k} = correlations of the data variables with that particular kth factor.
  - ◇ $\|a_{*k}\|$ = the *significance* of factor k.
    - ▷ Variables with high factor loadings are more "like" the factor in some sense.
    - ▷ Variables with zero or near-zero loadings are treated as being unlike the factor.

Tasks to Do

- Want to rewrite the loadings of variables over some *newly selected* factors.
    - ◇ Fewer factors.
    - ◇ Manifest more clearly the correlation between variables and factors.
- Suppose the newly selected factors are expressed in terms of columns of the orthogonal matrix

$$V := [\mathbf{v}_1, \ldots, \mathbf{v}_m] \in \mathbb{R}^{m \times m}. \tag{5}$$

    - ◇ The rewriting of factor loadings with respect to V is mathematically equivalent to a change of basis, i.e., A is now written as $B := AV$.
    - ◇ Determine some appropriate new basis for V.
    - ◇ Because $Y = AF = (AV)(V^T F) = BG$,
        - ▷ $B = AV$ denotes new factor loadings.
        - ▷ $G = V^T F$ denotes new factor scores.
    - ◇ The correlation matrix $R = AA^T = BB^T \in \mathbb{R}^{n \times n}$ is independent of factors selected.
        - ▷ Would like that the significance of factors concentrates on “fewer” columns of B.
        - ▷ Lower rank approximation of A.
- In the process of defining new factors it is often desirable to retrieve information ...
    - ◇ Directly from the correlation matrix R rather than from any particular loading matrix A, if A is not readily available; or
    - ◇ Approximate A, if A is too large or too expensive.

Latent Semantic Indexing

- Indexing matrix $H = [h_{ik}] \in \mathbb{R}^{n \times m}$:
  - ◇ Each document is represented by one row in H.
  - ◇ h_{ik} = the *weight* of one particular *term* k in document i.
    - ▷ Each term could be just one single word or a string of phrases.
    - ▷ The weight h_{ik} could simply be the number of occurrence of term k in document i.
    - ▷ More elaborate weighting schemes are available and yield better performance.
- Queries $\mathbf{q}_j = [q_{1j}, \ldots, q_{mj}]^T \in \mathbb{R}^m$:
  - ◇ q_{kj} = the weight of term k in the query j.
- Would like to find documents relevant to given queries.
  - ◇ To measure how the query $\mathbf{q}_j$ matches the documents,
    - ▷ Calculate the dots product

$$\mathbf{s}_j = H\mathbf{q}_j. \tag{6}$$

    - ▷ Rank the relevance of documents to $\mathbf{q}_j$ according to the *scores* in $\mathbf{s}_j$.

Comparison of LSI with Linear Model

- Analogies:

indexing matrix H	$\longleftrightarrow$	loading matrix A
document i	$\longleftrightarrow$	variable i
term k	$\longleftrightarrow$	factor k
weight h_{ik}	$\longleftrightarrow$	loading of factor k on variable i
one query $\mathbf{q}_j$	$\longleftrightarrow$	one column in scoring matrix F
weights q_{kj}	$\longleftrightarrow$	scores of entity j on factor k
scores in $\mathbf{s}_j$	$\longleftrightarrow$	scores in j column of data matrix Y

- Differences:
  - In LSI, terms/factors are predetermined.
    - How are the terms/factors predetermined?
    - What is the notion of “orthogonal words”?
    - What is the notion of “term/factor reduction”?
  - LSI is not trying to compute factors based on the scores in $\mathbf{s}_j$, $j = 1, \ldots, \ell$.
    - Though, this information may be used as a learning process for selecting terms/factors.
  - LSI emphasizes effective vector-matrix multiplication (6).
    - Want to represent the indexing matrix and the queries in a more *compact form* so as to facilitate the computation of the scores.

Cluster Analysis

- A procedure used to organize information about cases so that relatively homogenous groups, or ”clusters,” can be formed.
    - ◇ Group members should be highly internally homogenous (members are “similar” to one another in their characteristics) and highly externally heterogenous (members are not “like” members of other clusters).
        - ▷ Need a measurement of similarity or dissimilarity.
        - ▷ Need a decision on how many clusters to keep.
    - ◇ The classification has the effect of reducing the number of *rows* in the data table.
- The classification produced is very dependent upon the particular method used.
    - ◇ hierarchical — the resultant classification has an increasing number of nested classes.
    - ◇ k-means — partition the data between k clusters.

Examples of Cluster Analysis

- Split a shuffled deck of cards into two parts ($k = 2$).
- Classify living organisms into groups based on the shared possession of characteristics (taxonomy and dendrogram).
- AMS 1991 Mathematics Subject Classification
- Switch manufacturing in the telecommunication industry.
  - ◇ A cabinet consists of m slots.
  - ◇ Each slot may be filled with a selection from r types of boards.
  - ◇ History of past n customer orders have been recorded into a matrix $A \in \mathbb{R}^{n\times m}$.
  - ◇ Would like to preassemble q semi-finished cabinet models.
  - ◇ Determine the model configurations and the corresponding customer-to-model assignment of semi-finished cabinets based on A so as to minimize the total number of insertions required to manufacture the entire order.

Comparison of Cluster Analysis with Factor Analysis

- Consider a document-term matrix $A \in \mathbb{R}^{n \times m}$ with n documents and m terms.
- Analogies:

$$\begin{array}{lll} \text{(variable)} & \text{document } i & \longleftrightarrow \quad \text{case } i \\ \text{(factor)} & \text{term } k & \longleftrightarrow \quad \text{characteristic } k \end{array}$$

$$\begin{array}{rcl} (\text{row } h_{i*}) & \longleftrightarrow & \text{samples of document/case } i \text{ by terms/characteristics} \\ (\text{column } h_{*k}) & \longleftrightarrow & \text{samples of term/characteristic } k \text{ by document/cases} \\ & & \text{(These samples are not centered nor normalized.)} \end{array}$$

- Differences:
  - ◇ Factor analysis seeks to uncover the underlying structure of the set of factors/terms/characteristic.
    - ▷ Which original factors are highly correlated to the principal component factors?
    - ▷ Seek to reproduce the intercorrelation among the factors.
    - ▷ Reduce columns/terms/characteristics.
    - ▷ Generally known as the *R-mode factor analysis* or simply *factor analysis*.
  - ◇ Cluster analysis seeks to uncover the underlying structure of the set of variables/documents/cases.
    - ▷ Seek the intercorrelation among the variables.
    - ▷ Reduce rows/documents/cases.
    - ▷ Cluster analysis is also known as the *Q-mode factor analysis* or the *inverse factor analysis*.
- Same factor retrieval techniques can be applied to the transpose of the data table to retrieve clusters.

Centroid Method

- Temporarily assuming that a loading matrix $A_1 \in \mathbb{R}^{n\times m}$ is given.
  - ◇ Coordinate axes in $\mathbb{R}^m$ represent a set of m abstractly defined factors.
  - ◇ Define a new coordinate system representing the *centroid factors*.
  - ◇ Loadings with respect to the centroid factors can be calculated without the knowledge of A_1 or even the centroid factors. No need to know A_1 a priori.
- Denote each row of A_1 as a point in the factor space $\mathbb{R}^m$.
  - ◇ The arithmetic mean of these points, the *centroid*, is used to indicate a collective trend of the variables.
  - ◇ Generally variables that tend to vary together form *clusters*.
    - ▷ Truly uncorrelated variables form no clusters at all.
    - ▷ If all variables depend on the same factor, then a single cluster should be formed.
    - ▷ Do not know a priori how many clusters are to be expected.

Compute the Centroid

- The centroid of these n variables:

$$\mathbf{c}_1 := \frac{A_1^T \mathbf{1}_n}{n} = \left[\frac{\sum_{i=1}^n a_{i1}}{n}, \ldots, \frac{\sum_{i=1}^n a_{im}}{n} \right]^T. \tag{7}$$

  - ◇ First *centroid factor*:

$$\mathbf{v}_1 := \frac{\mathbf{c}_1}{\|\mathbf{c}_1\|}. \tag{8}$$

  - ◇ New loadings of variables with respect to $\mathbf{v}_1$:

$$\mathbf{b}_1 = A_1 \mathbf{v}_1 = A_1 \frac{A_1^T \mathbf{1}_n}{\|A_1^T \mathbf{1}_n\|} = \frac{R_1 \mathbf{1}_n}{\sqrt{\mathbf{1}_n^T R_1 \mathbf{1}_n}}. \tag{9}$$

    - ▷ Loading vector $\mathbf{b}_1$ is extracted directly from R_1. No reference to A_1 or $\mathbf{v}_1$ is needed.

- Remove the influence $\mathbf{v}_1$ from the the loading matrix:
  - ◇ Orthogonally reduced loading matrix:

$$A_2 := A_1 - A_1 \mathbf{v}_1 \mathbf{v}_1^T. \tag{10}$$

  - ◇ A_2 inherits most of the loading information of the original A_1 except for the loadings along the direction $\mathbf{v}_1$.

Update the Loading Matrix

- The product moment of A_2 can be computed via

$$\begin{aligned} R_2 = A_2A_2^T &= \left(A_1 - A_1\mathbf{v}_1\mathbf{v}_1^T\right)\left(A_1^T - \mathbf{v}_1\mathbf{v}_1^TA_1^T\right) \\ &= A_1A_1^T - A_1\mathbf{v}_1\mathbf{v}_1^TA_1^T \\ &= R_1 - \frac{R_1\mathbf{1}_n\mathbf{1}_n^TR_1}{\mathbf{1}_n^TR_1\mathbf{1}_n}, \end{aligned}$$

  - ◇ No explicit reference to A_2 is needed.
  - ◇ R_2 is exactly one rank less than R_1 (Wedderburn formula).

- Repeat the procedure to extract the next centroid factor for A_2, to introduce the next reduced loading matrix, and so on.
  - ◇ No! It cannot be done.
  - ◇ The procedure cannot be repeated because $A_2^T\mathbf{1}_n = \mathbf{0}_m$.
  - ◇ The centroid of A_2 is residing squarely at the origin of $\mathbb{R}^m$. No factor is retrieved.
  - ◇ Really bad?

Modified Centroid Factor

- Heuristic reasons for modification:
  - ⋄ Signs of loadings indicates positive or negative linear correlation between the variable and the factor $\mathbf{v}_1$. Either sign is fine.
  - ⋄ Asymmetrically distributed points in $\mathbb{R}^m$ identifies the centroid factor more easily.
    - ▷ $\|\mathbf{c}_1\|$ measures the *eccentricity* of the system of variables with respect to the origin.
    - ▷ The farther $\mathbf{c}_1$ is away from the origin, the more variables are qualitatively scattered in a general area surrounding $\mathbf{c}_1$.
    - ▷ The larger $\|\mathbf{c}_1\|$ is, the better an essential factor $\mathbf{v}_1$ represents.
- Replacing one particular variable by its negative does not cause trouble in the identification of an essential factor.
  - ⋄ Would change the sign of certain rows, if that helps to bring out the eccentricity.
- Observe the relationship:

$$\mathbf{1}_n^T R \mathbf{1}_n = \|A_1^T \mathbf{1}_n\|^2 = n^2 \|\mathbf{c}_1\|^2. \tag{11}$$

  - ⋄ The problem is now changed to solving the integer programming problem

$$\max_{|\mathbf{z}|=1} \mathbf{z}^T R_1 \mathbf{z}, \tag{12}$$

    - ▷ $|\mathbf{z}| = 1$ means components of the column vector $\mathbf{z}$ are either 1 or -1.
    - ▷ There are only 2^n many sign vectors for a fixed n.

Ready to Go!

- Modified centroid and factor:

$$\mathbf{c}_1 := \frac{A_1^T \mathbf{z}_1}{n}, \tag{13}$$

$$\mathbf{v}_1 := \frac{A_1^T \mathbf{z}_1}{\|A_1^T \mathbf{z}_1\|}, \tag{14}$$

  - ◇ $\mathbf{z}_1$ is the optimizer of (12).

- Centroid value of A_1:

$$\mu_1 := \frac{1}{n} \max_{|\mathbf{z}|=1} \mathbf{z}^T R_1 \mathbf{z}. \tag{15}$$

- Updating information:
  - ◇ New loading: $\mathbf{b}_1 = A_1 \mathbf{v}_1$ and can be computed by

$$\mathbf{b}_1 = A_1 \mathbf{v}_1 = \frac{R_1 \mathbf{z}_1}{\sqrt{\mathbf{z}_1^T R_1 \mathbf{z}_1}}. \tag{16}$$

  - ◇ New product moment R_2:

$$R_2 = R_1 - \frac{R_1 \mathbf{z}_1 \mathbf{z}_1^T R_1}{\mathbf{z}_1^T R_1 \mathbf{z}_1}. \tag{17}$$

  - ◇ Repeat the procedure.

Centroid Decomposition

- Each application of this centroid factor retrieval reduces the rank of the loading matrix by one.
  - ◇ The procedure therefore has to come to a stop in finitely many steps.
  - ◇ With the recurrence
    $$A_i = A_{i-1} - A_{i-1}\mathbf{v}_{i-1}\mathbf{v}_{i-1}^T, \quad i = 2, \dots, \gamma, \tag{18}$$
    we may write
    $$A = A_1 = \mathbf{b}_\gamma \mathbf{v}_\gamma^T + \dots + \mathbf{b}_1 \mathbf{v}_1^T, \tag{19}$$
    - ▷ $\mathbf{v}_i$ = the modified centroid factor of A_i.
    - ▷ γ = rank of A_1.
    - ▷ $\mathbf{b}_i = A_i \mathbf{v}_i$.
  - ◇ This is called a *centroid decomposition* of A.
    - ▷ Closely related to the singular value decomposition (SVD).
- The modified centroid vectors (and factors) are mutually orthogonal, even though they are not explicitly calculated.

$$\mathbf{c}_1^T \mathbf{c}_2 = \frac{1}{n^2}\left(\mathbf{z}_1^T A_1\right)\left(A_2^T \mathbf{z}_2\right) = \frac{1}{n^2}\left(\mathbf{z}_1^T A_1\right)\left[A_1^T\left(\mathbf{z}_2 - \frac{\mathbf{z}_1^T R_1 \mathbf{z}_2}{\mathbf{z}_1^T R_1 \mathbf{z}_1}\mathbf{z}_1\right)\right] = 0.$$

- When $\|\mathbf{b}_i\|$ is small, the factor $\mathbf{v}_i$ is less significant.
  - ◇ Less significant factors can be discarded.
  - ◇ Closely related to the truncated SVD (TSVD).

Topology of n-dimensional Hypercubes

- To perform the centroid decomposition, a sequence of integer programming problems must be solved.
    - ◇ The feasible set consists of 2^n sign vectors.
    - ◇ An exhaustive search would be expensive.
- Representing hypercubes:
    - ◇ Identifying -1 as 0 and keeping 1 as 1, a unique binary tag can be assigned to each sign vector.
    - ◇ Each binary tag translated into a unique integer between 0 and $2^n - 1$ provides a natural ordering of the sign vectors.
    - ◇ Each sign vector as one node connected only to those sign vectors whose binary tags differ from its own by exactly one bit (Hamming metric 1).
    - ◇ The set of 2^n sign vectors can be identified as an n-dimensional *hypercube*.

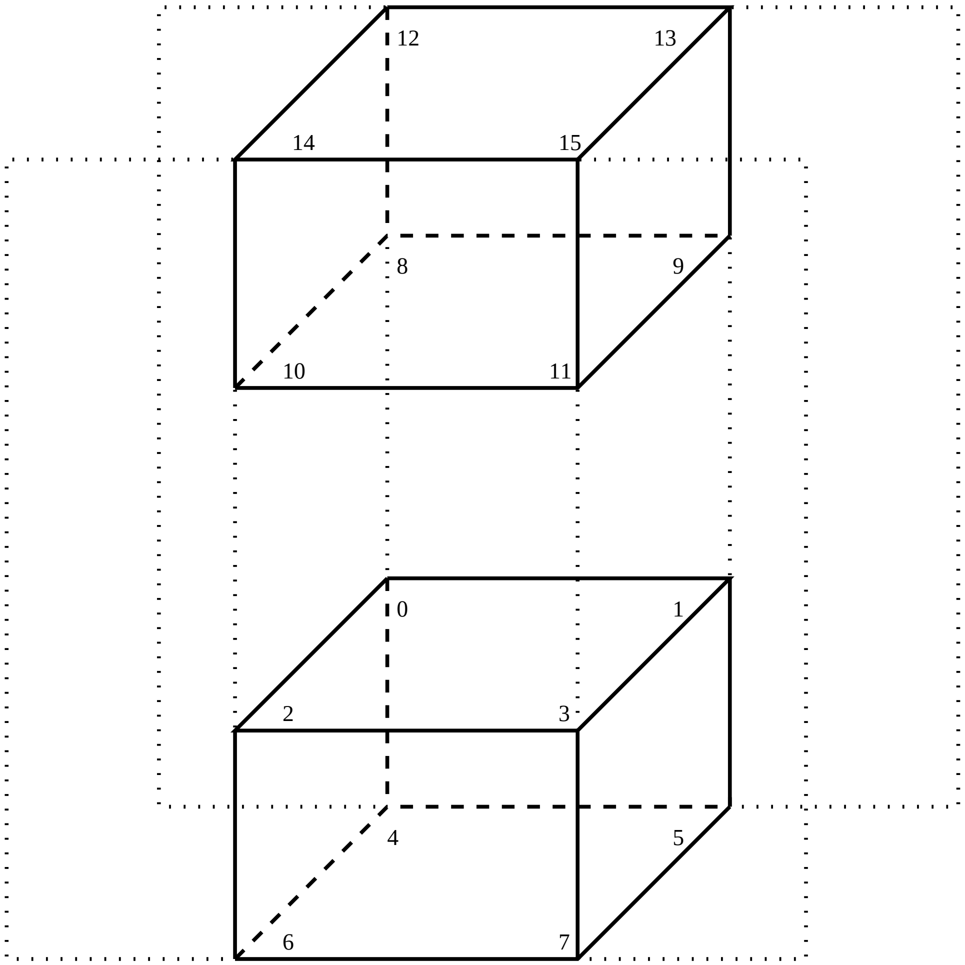

Figure 1: Topology of a 4-dimensional hypercube

- Each n-dimensional hypercube consists of two $(n-1)$-dimensional sub-hypercubes.
  - ◇ One sub-hypercube is simply a *bit reversal* of the other.
  - ◇ The objective values $z^T R z$ therefore always appear in pairs.

Integer Programming on Hypercubes

- Write $R = [r_{ij}] = P + diag(\mathrm{diag}(R))$.
  - ◇ $\mathbf{z}^T R\mathbf{z} = \mathbf{z}^T P\mathbf{z} + \sum_{i=1}^{n} r_{ii}$.
  - ◇ Suffice to maximize
    $$f(\mathbf{z}) := \mathbf{z}^T P\mathbf{z}$$
    with $|\mathbf{z}| = 1$.
- Classical centroid method:
  - ◇ Given any sign vector $\mathbf{z}^{(0)}$ and machine zero threshold ϵ.
  - ◇ Define $\mathbf{w}^{(0)} := P\mathbf{z}^{(0)}$.
  - ◇ Repeat the following steps for $i = 0, 1, \ldots$:
    1. If $\mathrm{sgn}(\mathbf{w}_k^{(i)}) = \mathrm{sgn}(\mathbf{z}_k^{(i)})$ for all $k = 1, \ldots, n$, then stop; otherwise, choose k so that $|\mathbf{w}_k^{(i)}| > \epsilon$ and is the largest among all $|\mathbf{w}_j^{(i)}|$'s where $\mathrm{sgn}(\mathbf{w}_j^{(i)}) \neq \mathrm{sgn}(\mathbf{z}_j^{(i)})$.
    2. Define $\mathbf{z}^{(i+1)}$ by simply changing the sign of $\mathbf{z}_k^{(i)}$.
    3. Define $\mathbf{w}^{(i+1)} := \mathbf{w}^{(i)} + 2\mathrm{sgn}(\mathbf{z}_k^{(i+1)})P(:, k)$.

- Main ideas:
  - ◇ At each given node on the hypercube, check through its neighboring nodes and move to the node with highest bit.
  - ◇ At most one bit is changed in each cycle.
$$\mathbf{z}^{(i+1)} := \mathbf{z}^{(i)} - 2\mathrm{sgn}(\mathbf{z}_k^{(i)})\mathbf{e}_k$$
  - ◇ The sequence $\{f(\mathbf{z}^{(i)})\}$ generated by the centroid method from any starting value $\mathbf{z}^{(0)}$ is finite and increasing.
$$\begin{aligned} f(\mathbf{z}^{(i+1)}) &= \left(\mathbf{z}^{(i)} - 2\mathrm{sgn}(\mathbf{z}_k^{(i)})\mathbf{e}_k\right)^T P \left(\mathbf{z}^{(i)} - 2\mathrm{sgn}(\mathbf{z}_k^{(i)})\mathbf{e}_k\right) \\ &= f(\mathbf{z}^{(i)}) - 4\mathrm{sgn}(\mathbf{z}_k^{(i)})(\mathbf{e}_k^T P\mathbf{z}^{(i)}) \\ &= f(\mathbf{z}^{(i)}) - 4\mathrm{sgn}(\mathbf{z}_k^{(i)})\mathbf{w}_k^{(i)}. \end{aligned}$$
  - ◇ The centroid method is a steepest ascent method along the nodes of the hypercube.
- Cost:
  - ◇ It takes at most n iterations to locate a maximum.
  - ◇ Tje expected number of iterations required for convergence is $n/2$.

Statistical Meaning of TSVD

- Let $\mathcal{X} \in \mathbb{R}^n$ denote a random column vector.
  - ⋄ $C := \mathcal{E}[(\mathcal{X} - \mathcal{E}[\mathcal{X}])(\mathcal{X} - \mathcal{E}[\mathcal{X}])^T] \in \mathbb{R}^{n\times n}$ is defined as the *covariance matrix* of $\mathcal{X}$.
  - ⋄ In practice, often only random samples are available. A data matrix $X = [x_{ij}] \in R^{n\times m}$ is collected where each column represents one sample of $\mathcal{X}$.
  - ⋄ Sample mean and sample covariance matrix approximate the true mean and the true covariance matrix of the random variable $\mathcal{X}$, if m is large enough.
- Assume that C has a spectral decomposition

$$cov(\mathcal{X}) = \sum_{j=1}^{n} \lambda_j \mathbf{u}_j \mathbf{u}_j^T.$$

  - ⋄ $\mathbf{u}_1, \ldots, \mathbf{u}_p$ are deterministic and form an orthonormal basis for R^n.
  - ⋄ The random column vector $\mathcal{X}$ can be expressed as

$$\mathcal{X} = \sum_{j=1}^{n} (\mathbf{u}_j^T \mathcal{X}) \mathbf{u}_j.$$

  - ⋄ Each coefficient $\alpha_j := \mathcal{X}^T u_j$ itself is a random variable.
- Properties of α:

$$\begin{array}{rcl} \mathcal{E}[\alpha] & = & U^T \mathcal{E}[\mathcal{X}], \\ cov(\alpha) & = & \mathrm{diag}\{\lambda_1, \ldots, \lambda_n\}. \end{array}$$

Meaning of Truncation

- Since $\mathbf{u}_j$, $j = 1, \ldots n$, are deterministic, stochastic properties of $\mathcal{X}$ are caused only by the stochastic properties of coefficients $\alpha_j, j = 1, \ldots, n$.
    - ◇ The randomness of $\mathcal{X}$ is due to the randomness of α.
    - ◇ Variance measures the unpredictability of a random variable.
    - ◇ Random variables $\alpha_j, j = 1, \ldots, n$, are mutually stochastically independent.
- The larger the eigenvalue λ_j is, the larger the variance of α_j is and, hence, the more randomness it contributes.
    - ◇ Those coefficients with larger variances and the corresponding directions are the more important components in representing the stochastic nature of $\mathcal{X}$.
    - ◇ Rank the importance of corresponding eigenvectors $\mathbf{u}_j$ as *essential* components for the variable $\mathcal{X}$ according to the magnitude of λ_j.
    - ◇ If truncation is necessary, those eigenvectors corresponding to smaller variances should be thrown away first.

Lower Dimensional Minimum-Variance Approximation

- Given a random vector $\mathcal{X} \in \mathbb{R}^n$ with mean zero, let its covariance matrix be spectrally decomposed as

$$cov(\mathcal{X}) = \sum_{j=1}^{n} \lambda_j \mathbf{u}_j \mathbf{u}_j^T.$$

Then among *all* unbiased variables restricted to *any* r-dimensional subspaces in R^n, the random variable

$$\hat{\mathcal{X}} := \sum_{j=1}^{r} (\mathbf{u}_j^T \mathcal{X}) \mathbf{u}_j \tag{20}$$

is the best linear minimum-variance estimate of $\mathcal{X}$ in the sense that $\mathcal{E}[\|\mathcal{X} - \hat{\mathcal{X}}\|^2]$ is minimized.

Truncation in Sample Space

- The distribution of a random variable is often simulated by a collection of ℓ random samples.
  - ⋄ Samples are recorded in a $n \times \ell$ matrix X.
  - ⋄ Each column of X represents one random sample of the underlying random (column vector) variable $\mathcal{X} \in \mathbb{R}^n$.
  - ⋄ When ℓ is large enough, many of the stochastic properties of $\mathcal{X}$ can be recouped from X.
- How to retrieve a sample data matrix from X to represent the minimum-variance approximation $\hat{\mathcal{X}}$ of $\mathcal{X}$?
  - ⋄ Sample covariance:

$$R = \frac{XX^T}{\ell}.$$

  - ⋄ Spectral decomposition of sample variance:

$$R = \sum_{i=1}^{n} \mu_i \mathbf{u}_i \mathbf{u}_i^T. \tag{21}$$

  - ⋄ Best low dimensional minimum-variance estimate $\hat{\mathcal{X}}$ to $\mathcal{X}$:

$$\hat{X} := \sum_{j=1}^{r} \mathbf{u}_j(\mathbf{u}_j^T X). \tag{22}$$

TSVD

- The *low dimension* estimate $\hat{\mathcal{X}}$ to the (continuous) random variable $\mathcal{X}$ is ntranslated into a *low rank* approximation $\hat{X}$ to the (discrete) random sample matrix X.
- The singular value decomposition of X:
$$X = U\Sigma V^T = \sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \tag{23}$$
  - ◇ Share the same eigenvectors of R as its left singular vectors, i.e., $U = [\mathbf{u}_1, \ldots, \mathbf{u}_n]$.
  - ◇ Singular values $\sigma_i = \sqrt{\ell \mu_i}$ are ranked in the same ordering as eigenvalues μ_i, $i = 1, \ldots n$.
  - ◇ The notion of the truncated singular value decomposition of X is simply the partial sum $\sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$.
- The TSVD of a give data matrix X representing random samples of an unknown random variable $\mathcal{X}$ has a statistical meaning.
  - ◇ The truncated rank-r SVD represents random samples of the best minimum-variance linear estimate $\hat{\mathcal{X}}$ to $\mathcal{X}$ among all possible r-dimensional subspaces.

Centroid Decomposition versus SVD

- Observe that that

$$\begin{aligned}\lambda_1(R_1) &= (\mathbf{u}_1(R_1))^T R_1 \mathbf{u}_1(R_1) = \max_{\|\mathbf{u}\|=1} \mathbf{u}^T R_1 \mathbf{u} = \max_{\|\mathbf{u}\|=1} \|A_1^T \mathbf{u}\|^2 \\ &\geq \mu_1 = \frac{1}{n}\mathbf{z}_1^T R_1 \mathbf{z}_1 = \frac{1}{n}\max_{|\mathbf{z}|=1} \mathbf{z}^T R_1 \mathbf{z} = \frac{1}{n}\max_{|\mathbf{z}|=1} \|A_1^T \mathbf{z}\|^2, \qquad (24)\end{aligned}$$

where $\mathbf{z}_1$ is the sign vector defining the first modified centroid.

- ◇ The sign vector $\mathbf{z}_1$ and the centroid value μ_1 is *mimicking* the left singular vector $\mathbf{u}_1$ and the square of the singular value λ_1 of A_1, respectively.
- ◇ Geometric meaning of the variational formulation.

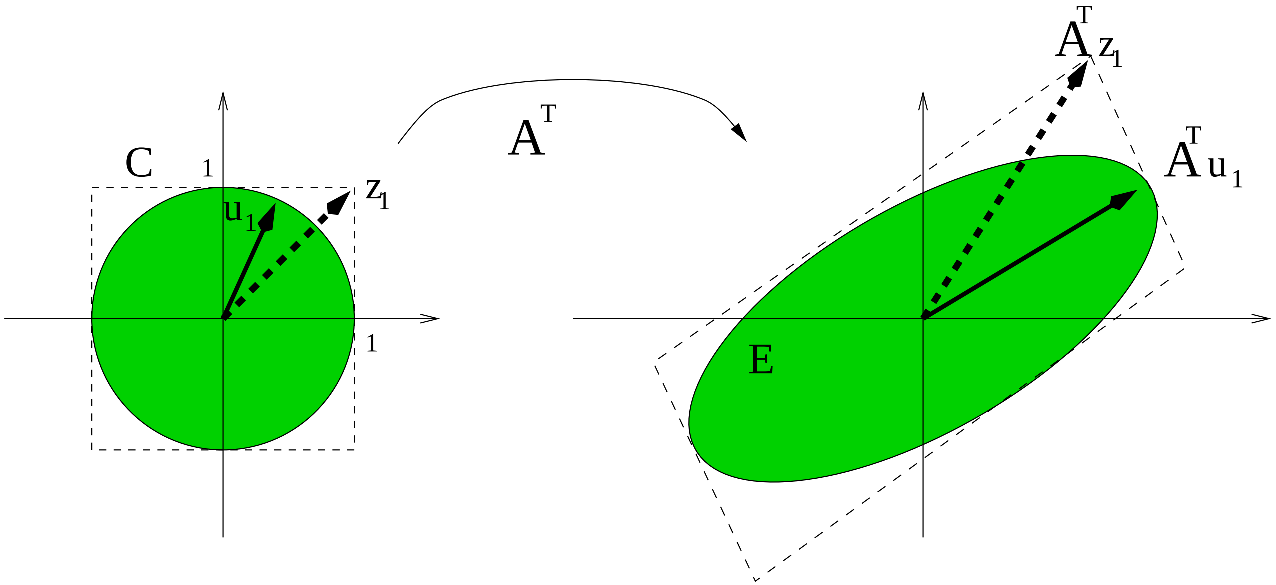

Figure 2: Comparison of geometric meanings of $\mathbf{z}_1$ and $\mathbf{u}_1(R_1)$ when $n = 2$.

Centroid Decomposition	Singular value Decomposition
$\mu_1 = \frac{1}{n} \max_{\lvert\mathbf{z}\rvert=1} \mathbf{z}^T R_1 \mathbf{z}$ (centroid value)	$\lambda_1 = \max_{\lVert\mathbf{x}\rVert=1} \mathbf{x}^T R_1 \mathbf{x}$ (eigenvalue)
$\mathbf{z}_1 = \arg\max_{\lvert\mathbf{z}\rvert=1} \mathbf{z}^T R_1 \mathbf{z}$ (sign vector for modified centroid)	$\mathbf{u}_1 = \arg\max_{\lVert\mathbf{x}\rVert=1} \mathbf{x}^T R_1 \mathbf{x}$ (left singular vector)
easy to obtain $\mathbf{z}_1$ in O(n) steps (tranverse hypercube)	not easy to obtain $\mathbf{u}_1$ via iterations (nonlinear iteration)
$\mathbf{v}_1 = \frac{A_1^T \mathbf{z}_1}{\sqrt{n\mu_1}}$ (centroid factor)	$\hat{\mathbf{v}}_1 = \frac{A_1^T \mathbf{u}_1}{\sqrt{\lambda_1}}$ (right singular vector)
$\gamma_1 = \lVert A_1 \mathbf{v}_1 \rVert$ (significance)	$\sigma_1 = \sqrt{\lambda_1} = \lVert A_1 \hat{\mathbf{v}}_1 \rVert$ (largest singular value)
$b_1 = A_1 \mathbf{v}_1$ (loading vector)	$\sigma_1 \mathbf{u}_1 = A_1 \hat{\mathbf{v}}_1$ (internal relation)
$A_1 = \sum b_i \mathbf{v}_i^T$ (centroid decomposition)	$A_1 = \sum \sigma_i \mathbf{u}_i \hat{\mathbf{v}}_i^T$ (singular value decomposition)
$R = \sum b_i b_i^T = \sum \gamma_i^2 \frac{b_i}{\lVert b_i \rVert} \left(\frac{b_i}{\lVert b_i \rVert} \right)^T$ (factor decomposition)	$R = \sum \lambda_i \mathbf{u}_i \mathbf{u}_i^T = \sum \sigma_i^2 \mathbf{u}_i \mathbf{u}_i^T$ (spectral decomposition)
$R_2 = R_1 - \frac{R_1 \mathbf{z}_1 \mathbf{z}_1^T R_1}{\mathbf{z}_1^T R_1 \mathbf{z}_1} = R_1 - \gamma_1^2 \frac{b_1}{\lVert b_1 \rVert} \left(\frac{b_1}{\lVert b_1 \rVert} \right)^T$ (rank reduction)	$\overline{R}_2 = R_1 - \frac{R_1 \mathbf{u}_1 \mathbf{u}_1^T R_1}{\mathbf{u}_1^T R_1 \mathbf{u}_1} = R_1 - \lambda_1 \mathbf{u}_1 \mathbf{u}_1^T$ (rank reduction)

Table 1: Comparison of centroid decomposition and singular value decomposition.

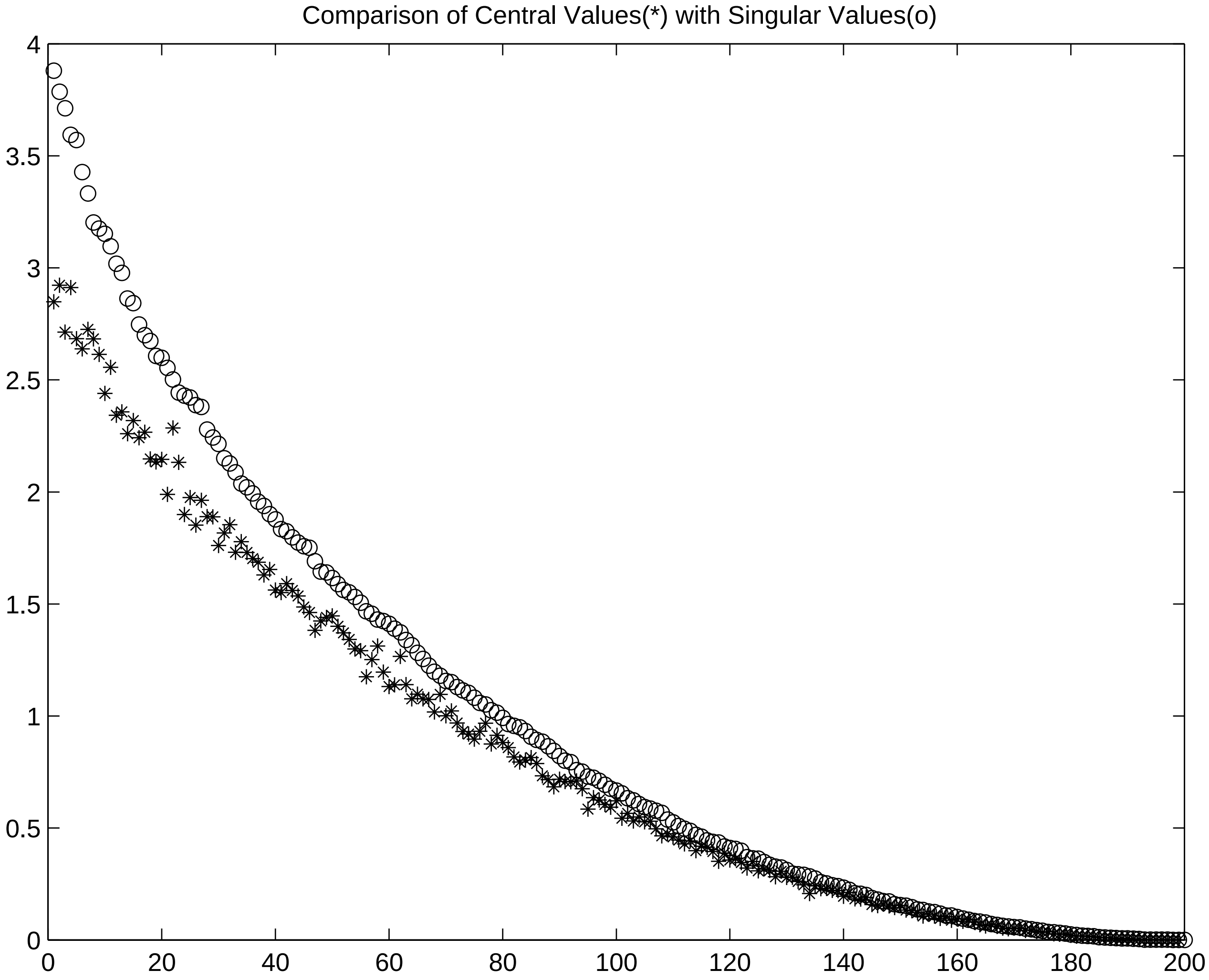

Figure 3: Comparison of centroid values and singular values for correlation matrix of $n = 200$.

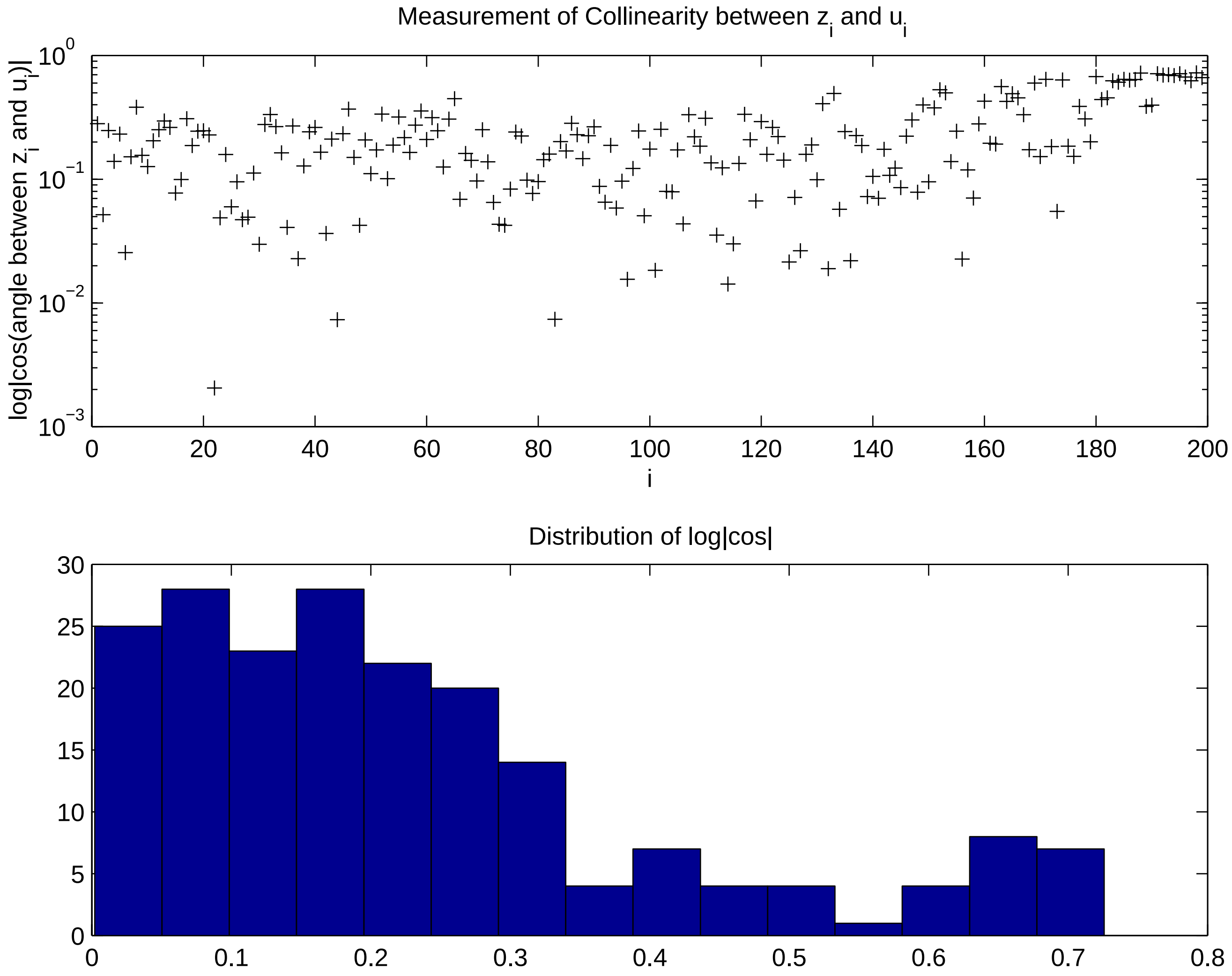

Figure 4: Degree of Alignment between $\mathbf{z}_i$ and u_i.

Conclusion

- We try to clarify the notions of objectives in data mining.
- We compare similarities and differences between factor analysis and cluster analysis.
- The centroid method is cast as an $\mathcal{O}(n)$-step optimization problem on a hypercube.
- Centroid decomposition is a cheaper simulator of the SVD.
- We offer the insight explaining why, how, and when a low rank approximation makes sensible approximation tot he original matrix.
- We show empirically that the centroid decomposition provides a measurement of second order statistical information of the original data.
- The information of significance of a loading vector provides a decision-making on when principal factors have been found.