InfoLab

IC612: Data Warehousing and Data Mining

Lecture 2: Data Preparation

Min-Soo Kim

DGIST 대구경북과학기술원
Daegu Gyeongbuk Institute of Science & Technology

Introduction

- **Data preparation: multistage process that comprises several individual steps**

1. **Feature extraction and portability**
   - it is desirable to derive meaningful features from the data
   - some algorithms may work only with a specific data type
     - but, data may contain heterogeneous types
     - data type portability becomes important
2. **Data cleaning**
   - some missing entries may also be estimated by a process known as *imputation*
3. **Data reduction, selection, and transformation**
   - algorithms are generally more efficient when the size of the data is reduced
   - quality of data mining is improved if irrelevant features or irrelevant records are removed

Feature extraction

- **Crucial, but very application-specific step**
- **Closely related to the concept of data type portability**
  - low-level features of one type may be transformed to higher-level features of another type
  - e.g., wavelet or Fourier transforms, deep neural networks

- **Sensor data**
  - often collected as large volumes of low-level signals (massive)
  - sometimes converted to higher-level features using wavelet or Fourier transforms
  - usually regarded as time series data (after some cleaning)
  - c.f., field of signal processing

Image data

- pixels: represent an image as a primitive form
- color histograms: represent the features in different segments of an image
- visual words: represent ‘iconic’ image patches or fragments
- c.f., convolution neural network
- one challenge : data is generally very high-dimensional

Image

Collection of visual words

Web logs

- typically text strings in a pre-specified format
- relatively easy to convert into a multidimensional representation of categorical and numeric attributes
  - because the fields in the logs are clearly specified and separated

Network traffic

- variety of features are extracted from network packets
  - e.g., number of bytes transferred, e.g., network protocol used

Document data

- raw and unstructured form
- rich linguistic relations between different entities
- remove stop-words; stem data; use a bag-of-words representation
- entity recognition: locate and classify atomic elements in text into
  - predefined expressions of names of persons, organizations, locations, actions, numeric quantities, and so on
  - e.g., Bill Clinton lives in Chappaqua

Data type portability

- **Data type portability: crucial in data mining process**
  - because the data is often heterogeneous, and may contain multiple types
  - e.g., a demographic data set may contain both numeric and mixed attributes
- **Mixing of data types restricts the ability of the analyst to use off-the-shelf tools for processing**
  - porting data types may lose representational accuracy and expressiveness
  - it is best to customize the algorithm to the particular combination of data types to optimize results
  - however, time consuming and sometimes impractical
- **Methods for converting between various data types**

Numeric to categorical data: discretization

- **Most commonly used conversion**
- **Dividing the ranges of numeric attribute into ϕ ranges**
  - attribute is assumed to contain ϕ different categorical labeled values
  - e.g., age: [0,10], [11, 20], [21, 30], …
  - variations within a range are not distinguishable after discretization

- **Equal-width ranges may not be helpful in discriminating**
  - Equi-depth ranges
    - ranges are selected so that each range has an equal number of records
    - sorting and selecting the division points on the sorted attribute value
  - Equi-log ranges
    - range [a, b] is chosen s.t. log(b) − log(a) has the same value
    - useful when the attribute has an exponential distribution across a range

Categorical to numeric data: binarization

- **Converting the categorical attributes to binary form**
  - because binary data is a special form of both numeric and categorical data
  - we can use numeric algorithms on the binarized data

- **If a categorical attribute has ϕ different values, then ϕ different binary attributes are created**
  - each binary attribute corresponds to one possible value of the categorical attribute
  - exactly one of the ϕ attributes takes on the value of 1
  - remaining attributes take on the value of 0

Text to numeric data

- **Vector-space representation**
  - a sparse numeric data set with very high dimensionality
  - not very amenable to conventional data mining algorithms
- **Latent Semantic Analysis (LSA)**
  - transform the text collection to a non-sparse representation with lower dimensionality
- **LSA+Scaling**
  - each document $\overline{X} = (x_1 \ldots x_d)$ needs to be scaled to $\frac{1}{\sqrt{\sum_{i=1}^{d} x_i^2}}(x_1 \ldots x_d)$
  - documents of varying length are treated in a uniform way
  - traditional numeric measures (e.g., Euclidean) work more effectively
  - traditional text-mining algorithms are directly applied to the reduced representation obtained from LSA

Time series to discrete sequence data

- **Symbolic Aggregate Approximation (SAX)**

- **Step1: window-based averaging**
  - times series is divided into windows of length w
  - the average time-series value over each window is computed

- **Step2: value-based discretization**
  - averaged time-series values are discretized into a smaller number of approximately equi-depth intervals
  - each equi-depth interval is mapped to a symbolic value
  - each symbol has an approximately equal frequency in time series
  - interval boundaries are constructed by assuming that the time-series values are distributed with a Gaussian assumption

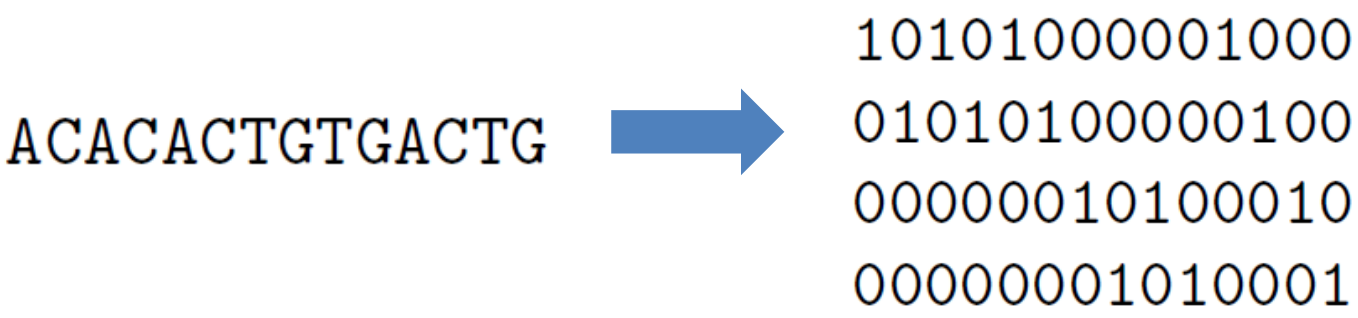
Time series to numeric data

- **Discrete wavelet transform (DWT)**
  - very useful because it enables the use of multidimensional algorithms for time-series data
  - convert time series to a set of coefficients that represent averaged differences between different portions of time series
  - a subset of the largest coefficients may be used to reduce the data size

- **c.f., discrete Fourier transform (DFT)**

Discrete sequence to numeric data

- **Step1: convert the discrete sequence to a set of (binary) time series**



- **Step2: map each of these time series into a multi-dimensional vector using the wavelet transform**

Any type to graphs for similarity-based applications

- **Neighborhood graph**
  - based on the notion of pairwise similarity
  - a given set of objects $O = \{O_1, \ldots, O_n\}$ become a set of nodes
  - an edge exists between O_i and O_j, if the distance $d(O_i, O_j)$ is less than a particular threshold
  - alternatively, the k-nearest neighbors of each node may be used
    - because the k-nearest neighbor relationship is not symmetric, this results in a directed graph
    - directions on the edges are ignored; the parallel edges are removed
  - larger edge weights indicate greater similarity: $w_{ij} = e^{-d(O_i,O_j)^2/t^2}$

Portability of different data types

Source Data Type	Destination Data Type	Methods
Numeric	Categorical	Discretization
Categorical	Numeric	Binarization
Text	Numeric	Latent Semantic Analysis (LSA)
Time Series	Discrete Sequence	SAX
Time Series	Numeric Multidimensional	DWT, DFT
Discrete Sequence	Numeric Multidimensional	DWT, DFT
Spatial	Numeric Multidimensional	2-d DWT
Graphs	Numeric Multidimensional	MDS, Spectral
Any Type	Graphs	Similarity Graph (Restricted Applicability)

Data cleaning

- **Several sources of missing entries and errors:**
  - Some data collection technologies, such as sensors, are inherently inaccurate because of the hardware limitations
  - Data collected using scanning technologies may have errors associated with it
    - e.g., optical character recognition techniques are far from perfect
  - Users may not want to specify their information for privacy reasons, or they may specify incorrect values intentionally
    - e.g., users specify their birthday incorrectly at registration sites
  - A significant amount of data is created manually
    - manual errors are common during data entry

Several important aspects of data cleaning:

- **Handling missing entries**
  - many entries in the data may remain unspecified
  - such missing entries may need to be estimated (called imputation)
- **Handling incorrect entries**
  - when the same information is available from multiple sources, inconsistencies may be detected
    - such inconsistencies can be removed as a part of analytical process
  - data points that are inconsistent with the remaining data distribution are often noisy (called outliers)
    - sometimes, such assumption is dangerous (e.g., credit-card fraud)
- **Scaling and normalization**
  - data may often be expressed in very different scales (e.g., age and salary)
  - important to normalize the different features

Data reduction and transformation

- **Goal of data reduction: represent it more compactly**
  - data size is smaller, it is much easier to apply sophisticated and computationally expensive algorithms
  - reduction of the data may be in terms of the number of rows (records) or in terms of the number of columns (dimensions)
  - data reduction does result in some loss of information

- **Different types of data reduction**
  - Data sampling
  - Feature selection
  - Data reduction with axis rotation
  - Data reduction with type transformation

Sampling

- **Simple, intuitive, and relatively easy to implement**
- **Sampling for static data**
  - number of base data points is known in advance
  - sampling without replacement: from a data set D with n records, a total of $\lceil n \cdot f \rceil$ records are randomly picked (no duplicates)
  - sampling with replacement: records are sampled sequentially and independently for a total of $\lceil n \cdot f \rceil$ times (duplicates are possible)
  - biased sampling: some parts of the data are intentionally emphasized due to their greater importance
    - e.g., temporal-decay bias: more recent records have a larger chance of being included in the sample

Reservoir sampling for data streams

- a sample of k points is dynamically maintained from a data stream

- for each incoming data point, we need to dynamically make two simple admission control decisions
  1. What sampling rule should be used to decide whether to include the newly incoming data point in the sample?
  2. What rule should be used to decide how to eject a data point from the sample to “make room” for the newly inserted data point?

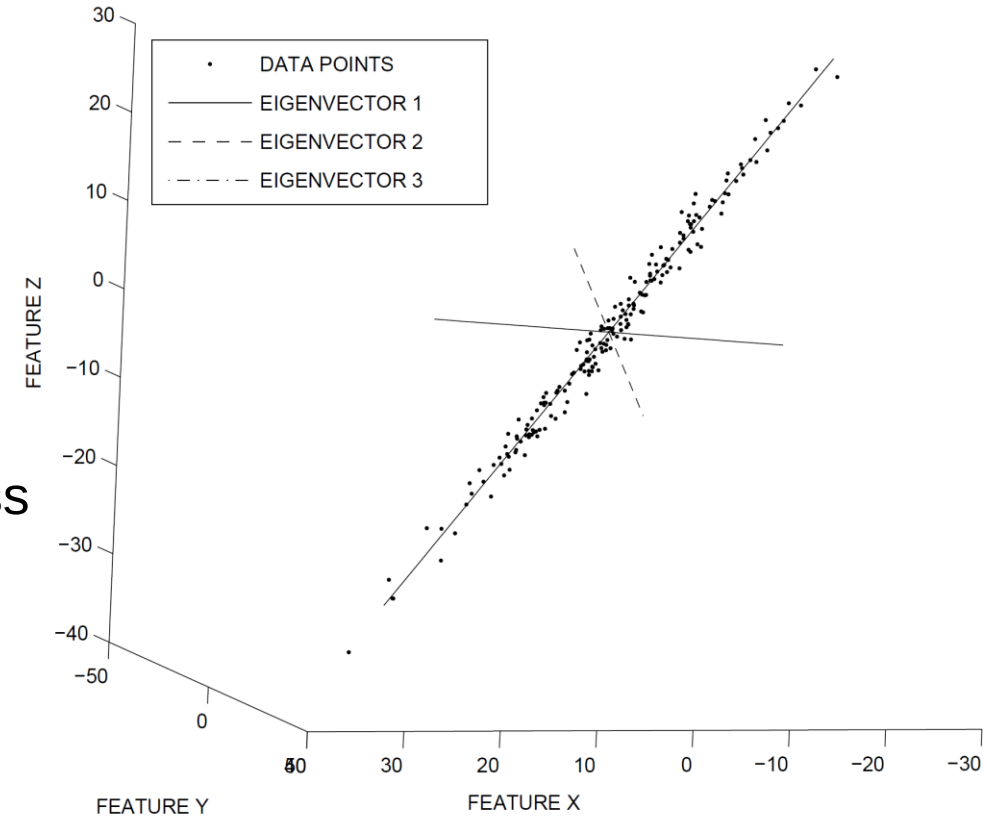
- unbiased reservoir sampling
  1. Insert the nth incoming stream data point into the reservoir with probability k/n
  2. If the newly incoming data point was inserted, then eject one of the old k data points at random to make room for the newly arriving point

Feature subset selection

- **Some features can be discarded when they are irrelevant**
  - Which features are relevant?
- **Unsupervised feature selection**
  - remove noisy and redundant attributes from the data
  - best defined in terms of its impact on clustering applications
- **Supervised feature selection**
  - keep only the features that can predict the class attribute effectively (i.e., most relevant)
  - often closely integrated with analytical methods for classification

Dimensionality reduction with axis rotation

- **Significant number of correlations exist among different attributes**
  - e.g., date of birth attribute is perfectly correlated with age attribute
- **Axis rotation**
  - intrinsic dimension
  - low-variance dimensions
  - correlations and redundancies are removed without much information loss



Principal component analysis (PCA)

- **PCA**
  - goal: rotate the data into an axis-system where the greatest amount of variance is captured in a small number of dimensions
  - mean-centering: subtract the mean of the data set from each data point
  - variance of a data set along a particular axis can be expressed directly in terms of its covariance matrix

C be the $d \times d$ symmetric covariance matrix of the $n \times d$ data matrix D

x_k^m be the mth dimension of the kth record

μ_i represent the mean along the ith dimension

$$c_{ij} = \frac{\sum_{k=1}^{n} x_k^i x_k^j}{n} - \mu_i \mu_j \quad \forall i, j \in \{1 \dots d\}$$

$\overline{\mu} = (\mu_1 \ldots \mu_d)$ is the d-dimensional row vector representing the means

$$C = \frac{D^T D}{n} - \overline{\mu}^T \overline{\mu}$$

$\overline{\mathrm{v}}$: d-dimensional column vector

$$\overline{v}^T C \overline{v} = \frac{(D\overline{v})^T D\overline{v}}{n} - (\overline{\mu}\,\overline{v})^2 = \text{Variance of 1-dimensional points in } D\overline{v} \geq 0$$

- **Goal of PCA: successively determine orthonormal vectors v maximizing $\overline{v}^T C \overline{v}$**
  - covariance matrix can be diagonalized as follows

$$C = P \Lambda P^T$$

  - columns of P: orthonormal eigenvectors of C
  - Λ ; a diagonal matrix containing the nonnegative eigenvalues

  - entry Λ_{ii} : the eigenvalue corresponding to the i-th eigenvector of P
  - eigenvectors in P maximize the variance $\overline{v}^T C \overline{v}$ along the direction $\overline{v}$

  - diagonal matrix Λ is the new covariance matrix after axis-rotation

- **Principal components**
  - eigenvectors with large eigenvalues preserve greater variance
  - it generally suffices to retain only a small number of eigenvectors with large eigenvalues
  - columns of P: arranged from left to right s.t. decreasing eigenvalues

- **Transformed data matrix D' in new coordinate system**

$$D' = DP$$

  - its first (leftmost) k << d columns show significant variation in values

Singular value decomposition

- **SVD is more general than PCA**
  - SVD provides basis vectors of both the rows and columns of the data matrix
  - PCA only provides basis vectors of the rows of the data matrix

- **Difference in terms of mean translation**
  - basis vectors of PCA are invariant to mean-translation
  - basis vectors of SVD are not
    - when the data is not mean-centered, the basis vectors of SVD and PCA will not be the same
    - SVD is often applied without mean-centering to sparse nonnegative data (e.g., document-term matrices)

SVD: decomposable product of three matrices

$$D = Q\Sigma P^T$$

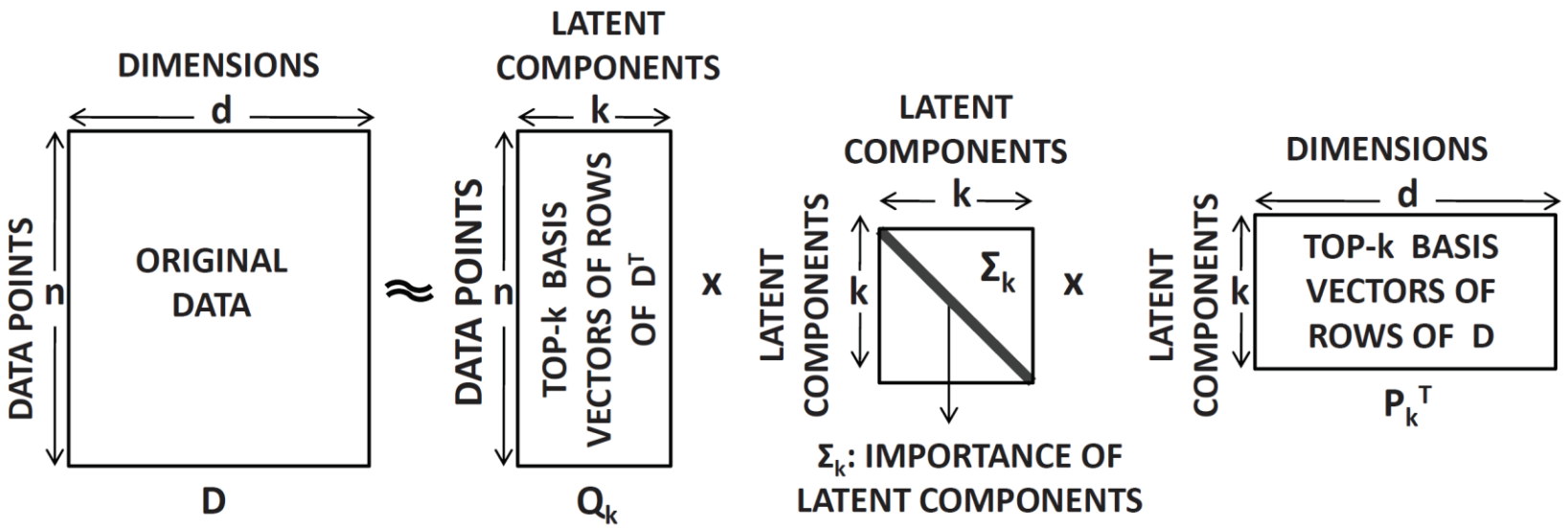
- Q : n × n matrix with orthonormal columns (left singular vectors)
- Σ : n×d diagonal matrix containing the singular values
  - always nonnegative
  - by convention, arranged in non-increasing order
- P : d×d matrix with orthonormal columns (right singular vectors)

- number of non-zero diagonal entries of Σ is equal to the rank of D

Approximate d-dimensional data representation of D

- Q_k : truncated n × k matrices obtained by selecting the first k columns of Q
- Σ_k : k × k square matrix containing the top k singular values
- P_k : truncated d × k matrices obtained by selecting the first k columns of P

$$D \approx Q_k \Sigma_k P_k^T$$



- **Truncated SVD expresses the data in terms of k dominant latent components**

$$D = \begin{pmatrix} 2 & 2 & 1 & 2 & 0 & 0 \\ 2 & 3 & 3 & 3 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 3 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 2 & 1 & 2 \end{pmatrix} \approx Q_2 \Sigma_2 P_2^T$$

$$\approx \begin{pmatrix} -0.41 & 0.17 \\ -0.65 & 0.31 \\ -0.23 & 0.13 \\ -0.56 & -0.20 \\ -0.10 & -0.46 \\ -0.19 & -0.78 \end{pmatrix} \begin{pmatrix} 8.4 & 0 \\ 0 & 3.3 \end{pmatrix} \begin{pmatrix} -0.41 & -0.49 & -0.44 & -0.61 & -0.10 & -0.12 \\ 0.21 & 0.31 & 0.26 & -0.37 & -0.44 & -0.68 \end{pmatrix}$$

$$= \begin{pmatrix} 1.55 & 1.87 & \underline{1.67} & 1.91 & 0.10 & 0.04 \\ 2.46 & 2.98 & 2.66 & 2.95 & 0.10 & -0.03 \\ 0.89 & 1.08 & 0.96 & 1.04 & 0.01 & -0.04 \\ 1.81 & 2.11 & 1.91 & 3.14 & 0.77 & 1.03 \\ 0.02 & -0.05 & -0.02 & 1.06 & 0.74 & 1.11 \\ 0.10 & -0.02 & 0.04 & 1.89 & 1.28 & 1.92 \end{pmatrix}$$

Latent semantic analysis (LSA)

- **Application of the SVD method to the text domain**
  - data matrix D : n × d document-term matrix containing normalized word frequencies in the n documents
  - d : size of the lexicon

- **Text domain suffers from two problems : synonymy and polysemy**
  - synonymy: two words may have the same meaning
  - polysemy: the same word may mean two different things (e.g., “jaguar”)

- **Truncated representation after LSA typically removes the noise effects of synonymy and polysemy**
  - singular vectors represent the directions of correlation in the data
  - appropriate context of the word is implicitly represented along these directions

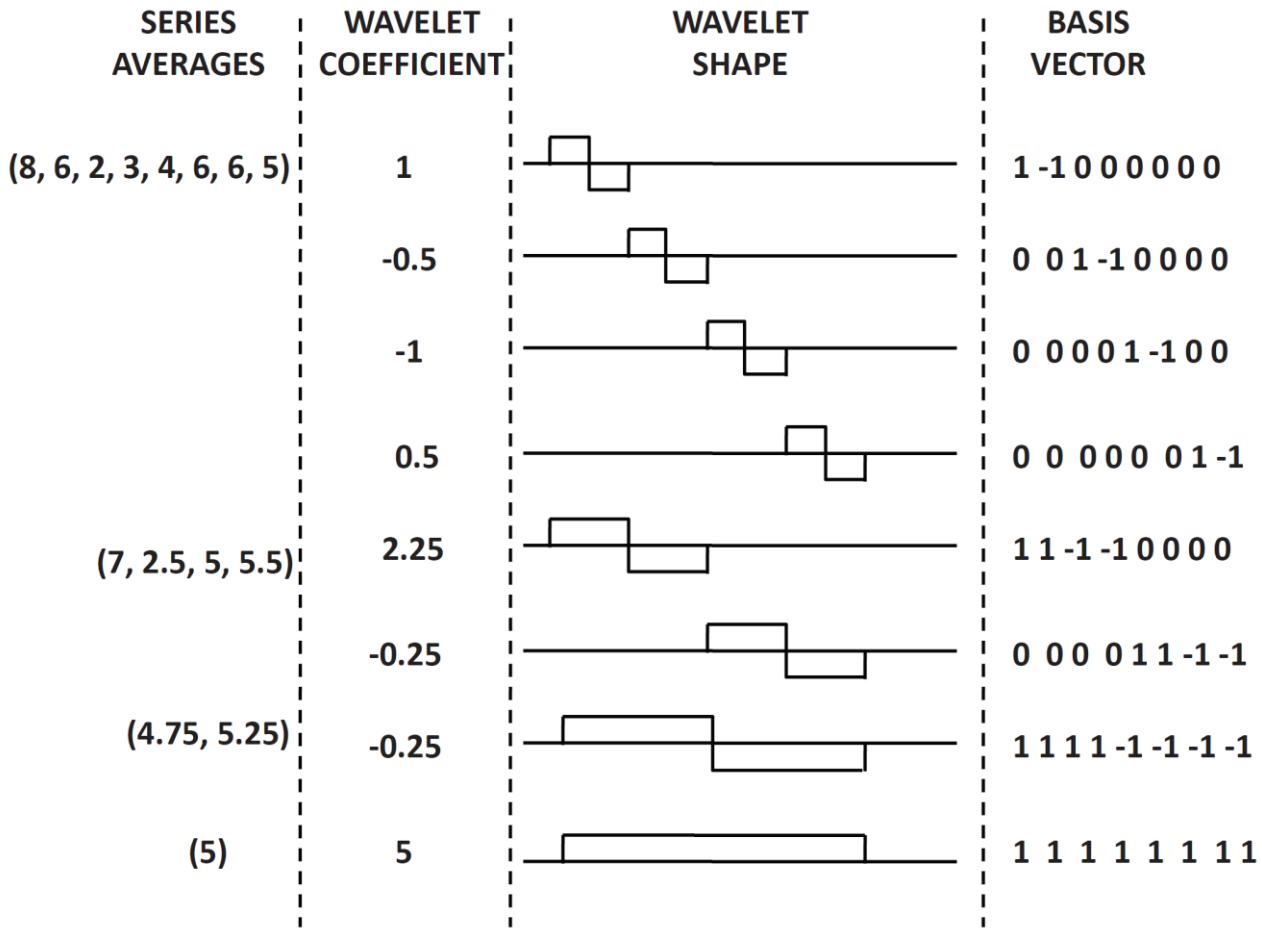
Applications of PCA and SVD

- **Primarily used for data reduction and compression**
- **Many other applications**
  - Noise reduction : improving the quality of data by removing the smaller eigenvectors/singular vectors
  - Data imputation : entire matrix can be approximately reconstructed as $Q_k \Sigma_k P_k^T$
  - Linear equations
  - Matrix inversion
  - Matrix algebra

Dimensionality reduction with type transformation

- **Haar wavelet: popular form of wavelet decomposition**
  - because of its intuitive nature and ease of implementation
- **e.g., sensor samples temperatures at the rate of 1 sample per second**
  - a sensor will collect 12×60×60 = 43, 200 readings for 12 hours
  - not scale well over many days and many sensors
  - question: how can we determine the key regions where “variations” occur, and store these variations instead of repeating values?
- **Wavelet technique**
  - create a decomposition of the time series into a set of coefficient-weighted wavelet basis vectors
  - higher-order coefficients represent the broad trends in the series

Granularity (Order k)	Averages (Φ values)	DWT Coefficients (ψ values)
$k = 4$	(8, 6, 2, 3, 4, 6, 6, 5)	-
$k = 3$	(7, 2.5, 5, 5.5)	(1, -0.5,-1, 0.5)
$k = 2$	(4.75, 5.25)	(2.25, -0.25)
$k = 1$	(5)	(-0.25)

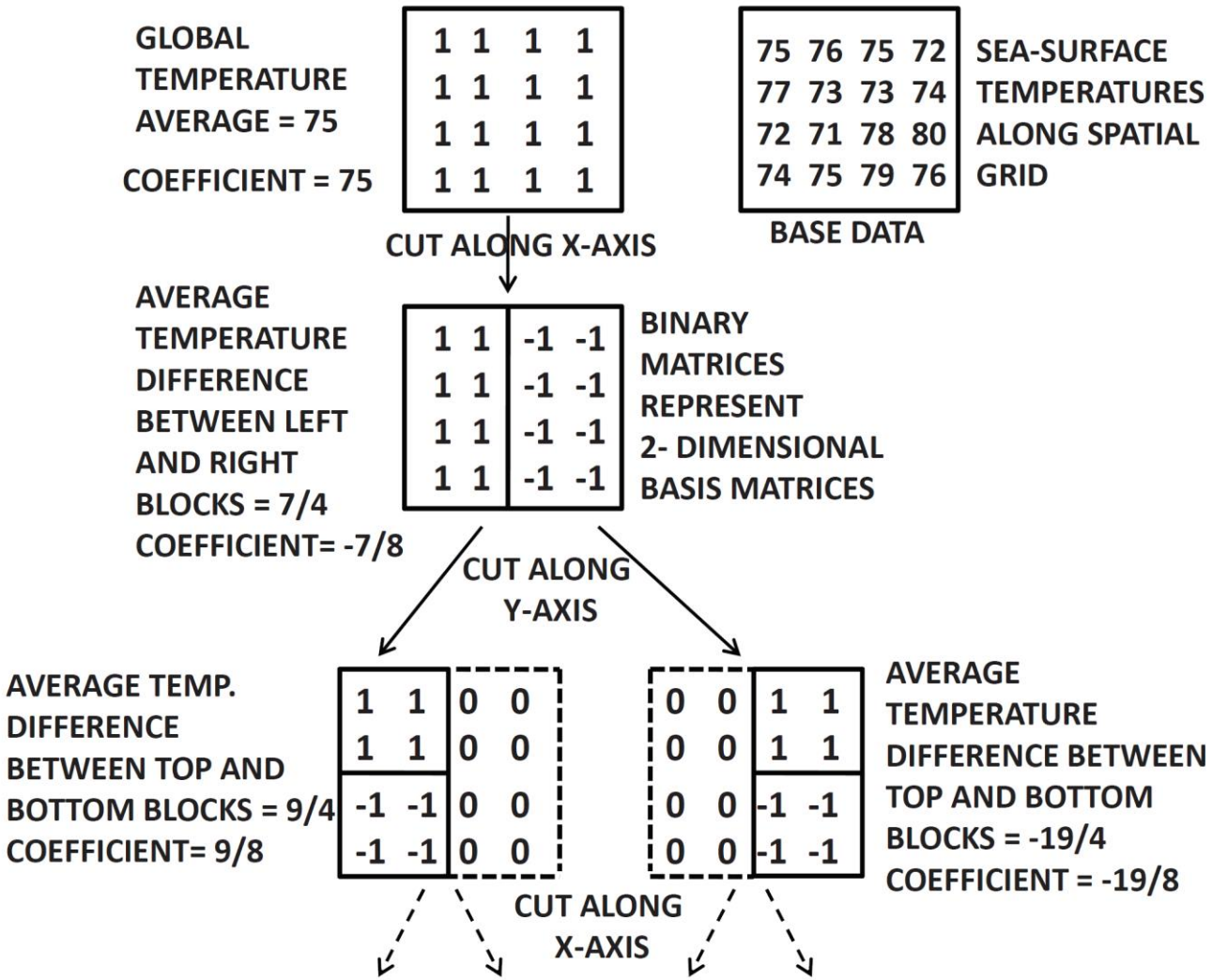


- **List of basis vectors**
  - dot product of any pair of basis vectors is 0
  - these series are orthogonal to one another

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

- **Wavelet decomposition: a natural method for dimensionality reduction (and data-type transformation)**
  - by retaining only a small number of coefficients

Wavelet decomposition with multiple contextual attributes



InfoLab

Thank you!

DGIST 대구경북과학기술원
Daegu Gyeongbuk Institute of Science & Technology