

Data Warehousing

Lecture 11 Data Reduction and Data Augmentation

CITS3401
CITS5504

Zeyi Wen

Computer Science and
Software Engineering

Faculty of Engineering,
Computing and
Mathematics

Acknowledgement: The lecture slides are based on online sources.

- **Data Reduction**
 - Data Reduction Overview
 - Attribute Reduction
 - Numerosity/Instance Reduction
- **Data Enhancement**
 - Data Augmentation
 - Oversampling

- **Data reduction:**
 - Obtain a reduced representation of the data set that is
 - much smaller in volume
 - but yet produces the same (or almost the same) analytical results.
- **Why data reduction?**
 - Increases storage capacity
 - Easy and efficient Mining, reduces time and memory requirement
 - Easy visualisation
 - Help to eliminate irrelevant /redundant features
 - Reduces noise

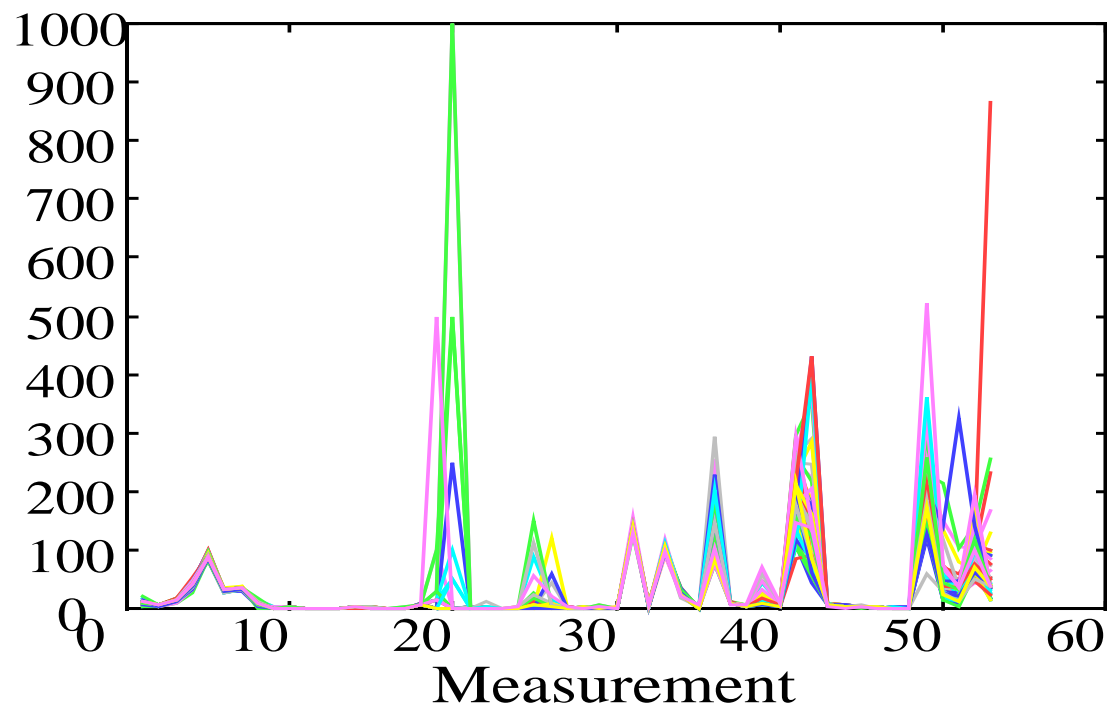
Example

- 53 Blood and urine measurements (wet chemistry) from 65 people (33 alcoholics, 32 non-alcoholics).
- Matrix Format (65 X 53): Difficult to see the correlation between the measurements.

		M1	M2	...	M52	M53	Alcoholics?
Instances	P1	-2.053920551	-1.50361144	...	1.02305704	-0.5052951	Yes
	P2	-0.004767651	0.04618693	...	-0.07452921	0.8229218	No
	P3	0.430102187	1.71553814	...	1.64038150	0.3130619	Yes
	P4	-0.817802417	1.56018735	...	-0.21835821	-0.6279286	No

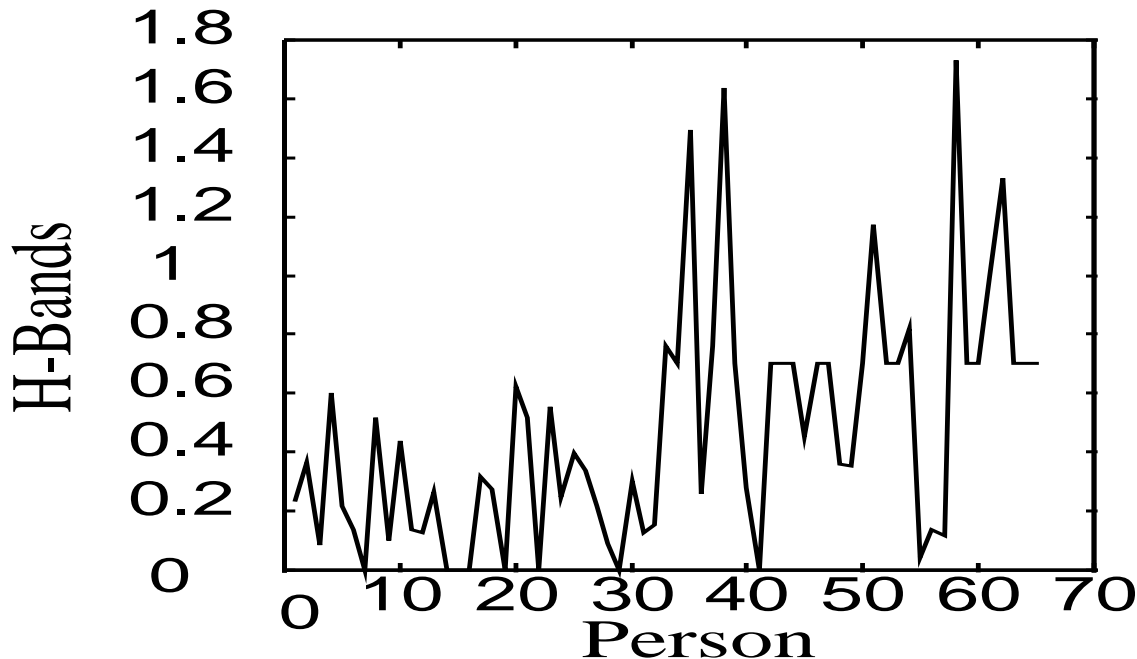
	P56	0.001701727	0.36459985	...	-1.59528279	2.5278118	No
		Features/attributes					

- Spectral format (65 lines, one for each person)



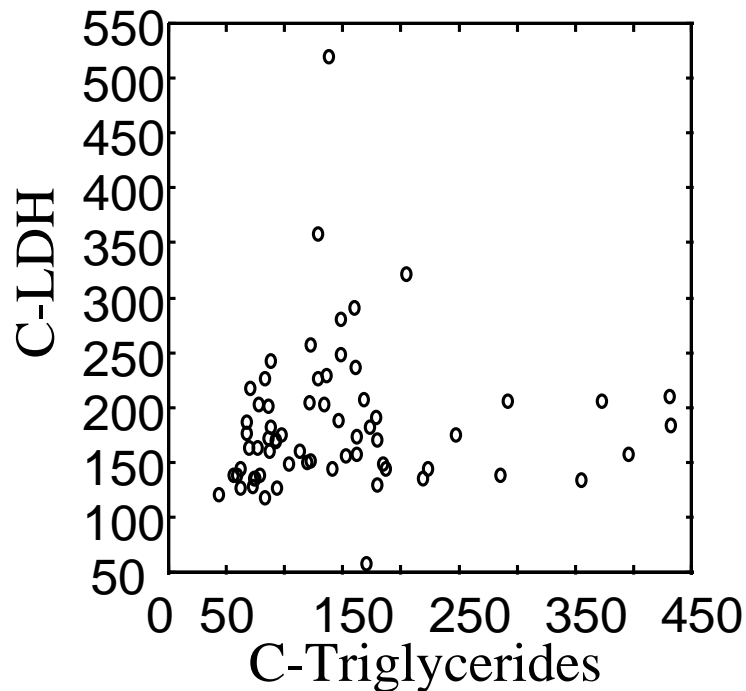
- Difficult to do cross patient analysis

- Spectral format (53 lines/figures, one for each measurement)

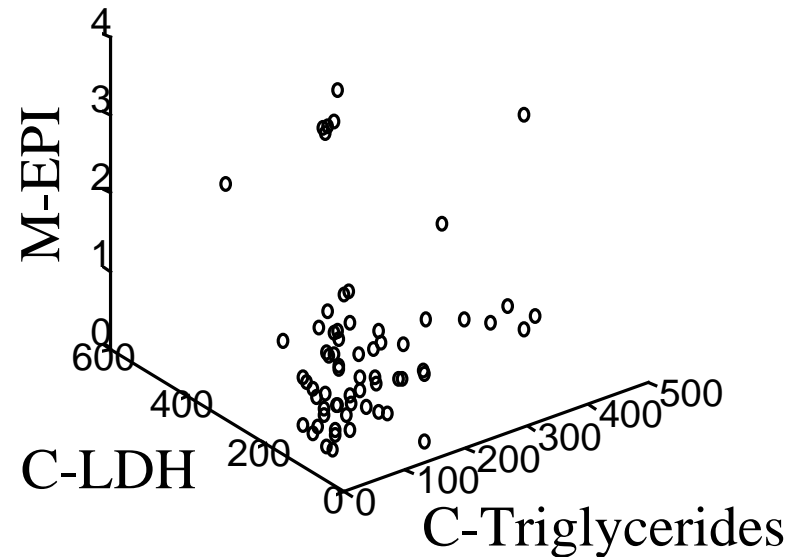


- Difficult to see the correlations between the measurements

Bi - variate



Tri - variate



- How can we visualise the other variables???
- ... difficult to see in 4 or higher dimensional spaces...

- Do we need a 53 dimension space to view data?
- What if there are strong correlation between features?
- How to find the 'best' low dimension space that conveys maximum useful information?

- **Dimensionality Reduction:**
 - Wavelet Transforms, Principal Component Analysis (PCA)
 - Mapping or projecting to a lower dimension feature space.
 - Attribute Subset Selection
- **Numerosity reduction:**
 - Parametric methods
 - Nonparametric methods
- **Data Compression**
 - Lossless vs. lossy compression

Data Reduction Strategies: Dimensionality Reduction

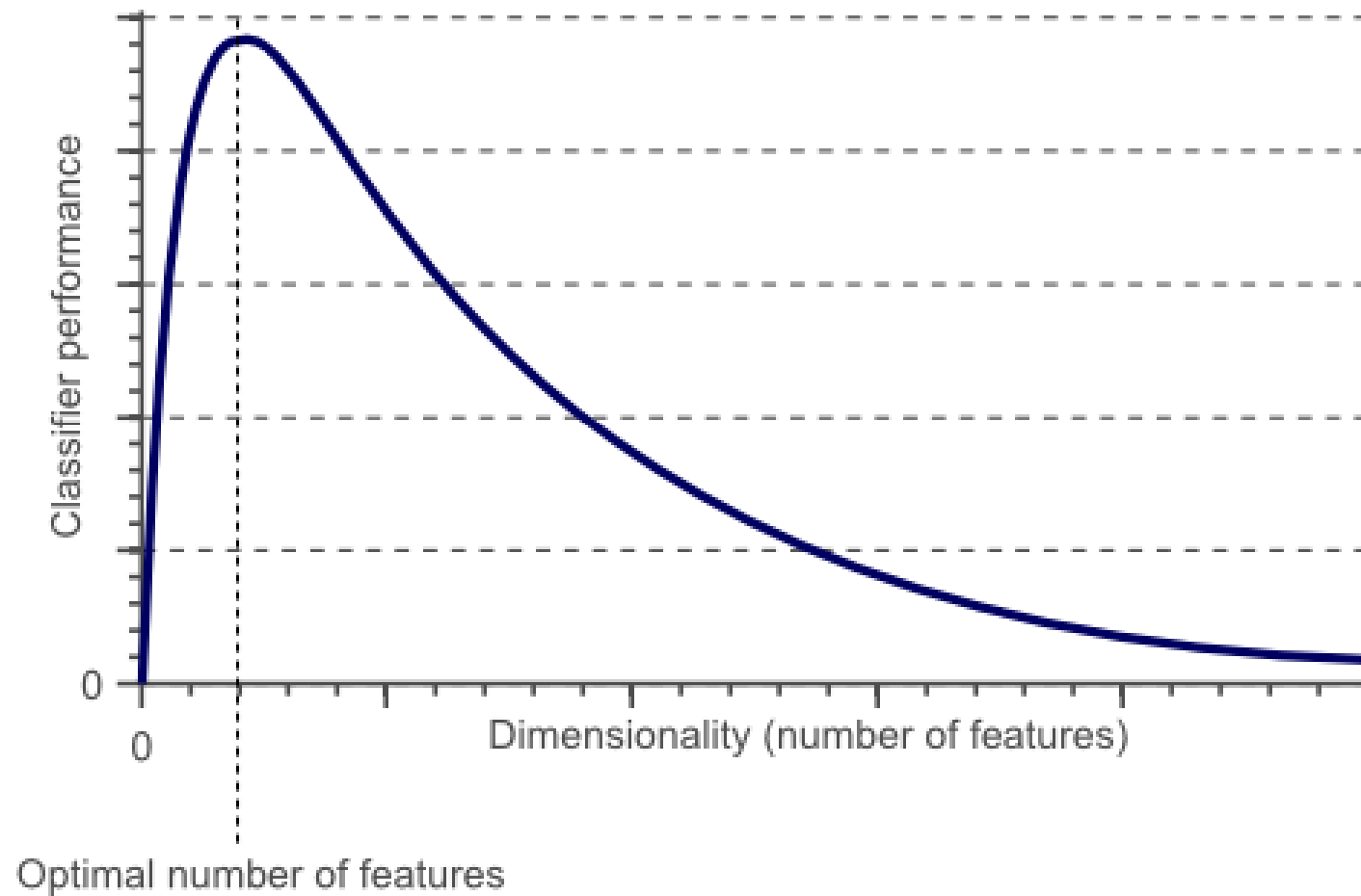
- **Dimensionality Reduction:**
 - Wavelet Transforms, Principal Component Analysis (PCA)
 - Mapping or projecting to a lower dimension feature space
 - Attribute Subset Selection
 - Selecting **only significant attributes**.
 - Optimal approach:
 - n-attributes have 2^n subsets in the simplest binary case, exhaustive search is **prohibitively expensive** for large 'n'
 - Greedy approach:
 - Irrelevant, weakly relevant or redundant attributes are deleted or removed

- **Numerosity Reduction:**
 - Parametric methods:
 - a **model is used to estimate the data**, so that typically only the data parameters need to be stored, instead of the actual data (outliers may also be stored).
 - regression models with Neural Networks are examples.
 - Nonparametric methods:
 - storing reduced representations of the data
 - histograms, **clustering**, **sampling**, and **data cube aggregation**
- **Data Compression**
 - Lossless vs. lossy compression

- **Data Reduction**
 - Data Reduction Overview
 - **Attribute Reduction**
 - Numerosity/Instance Reduction
- **Data Enhancement**
 - Data Augmentation
 - Oversampling

- When dimensionality increases, data becomes increasingly sparse
- The possible combinations of subspaces will grow exponentially.
 - Even in the simplest case of 'd' binary variables, the number of possible combinations is 2^d , exponential in the dimensionality.
- Density and distance between instances become less significant, which is critical for clustering and outlier analysis.
- Query accuracy and efficiency degrade rapidly as the dimension increases.

Performance vs. Dimensionality



- **Data Reduction**

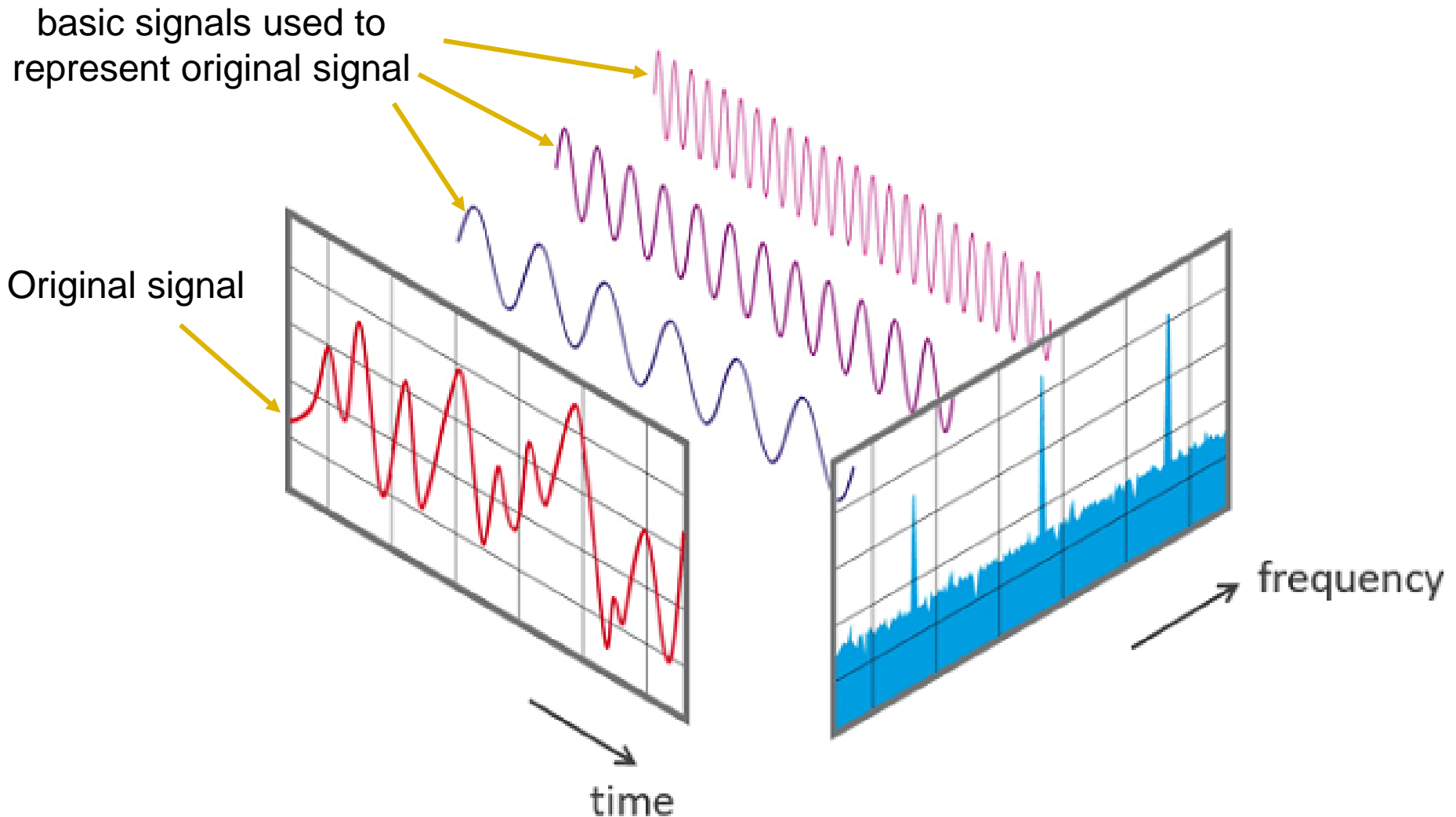
- Data Reduction Overview
- Attribute Reduction
 - **Discrete Wavelet Transformation (DWT)**
 - Principle Component Analysis (PCA)
 - Attribute Subset Selection
- Numerosity/Instance Reduction

- **Data Enhancement**

- Data Augmentation
- Oversampling

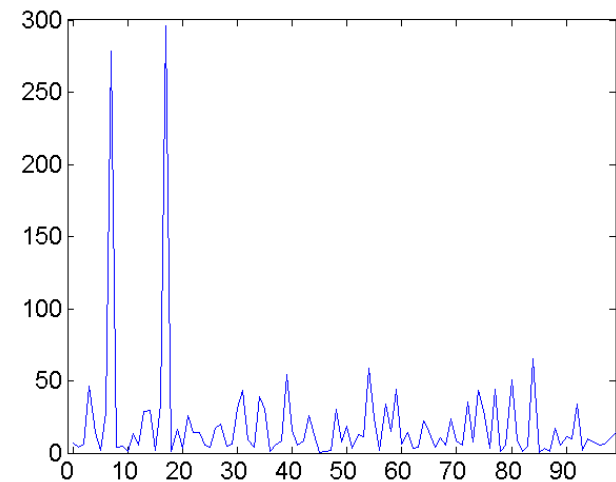
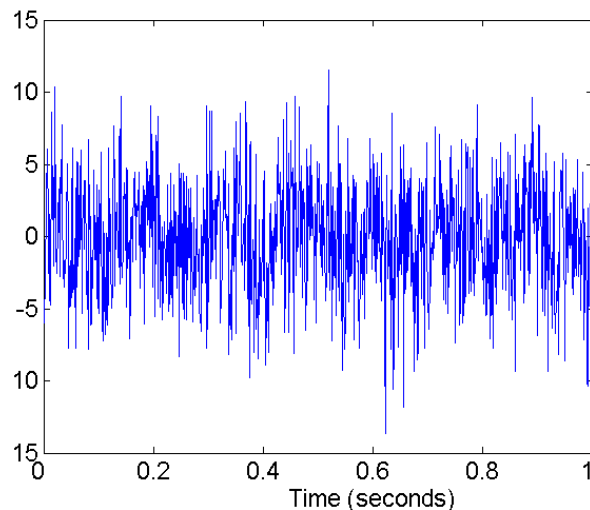
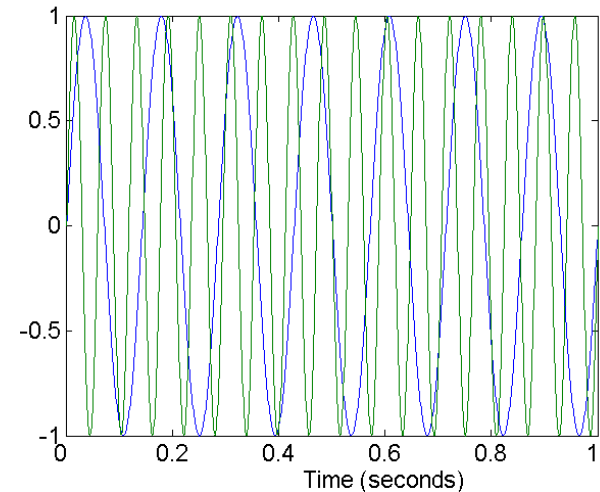
- Linear signal processing technique that transforms a data vector to a numerically different vector of wavelet coefficients.
- Properties
 - The original and resulting vectors are of the same size
 - But can be truncated if the coefficients is smaller than user specified value.
 - Closely related to Discrete Fourier Transform (DFT), but achieve better lossy compression, i.e. retain more accurate approximation of the original data, in general.
 - Better than JPEG for lossy image compression.

Discrete Fourier Transform (DFT)

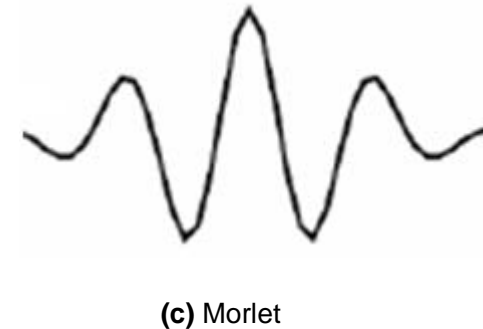
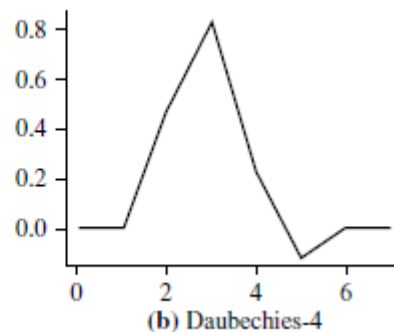
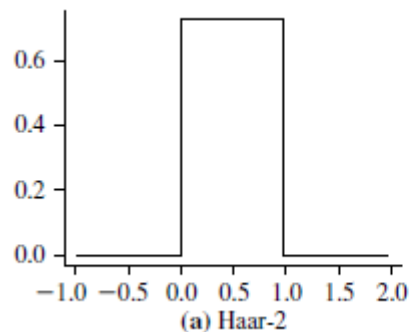


Mapping Data to a New Space

- **Discrete Fourier transform:**
 - Frequency Information 😊
 - Temporal Information ☹️



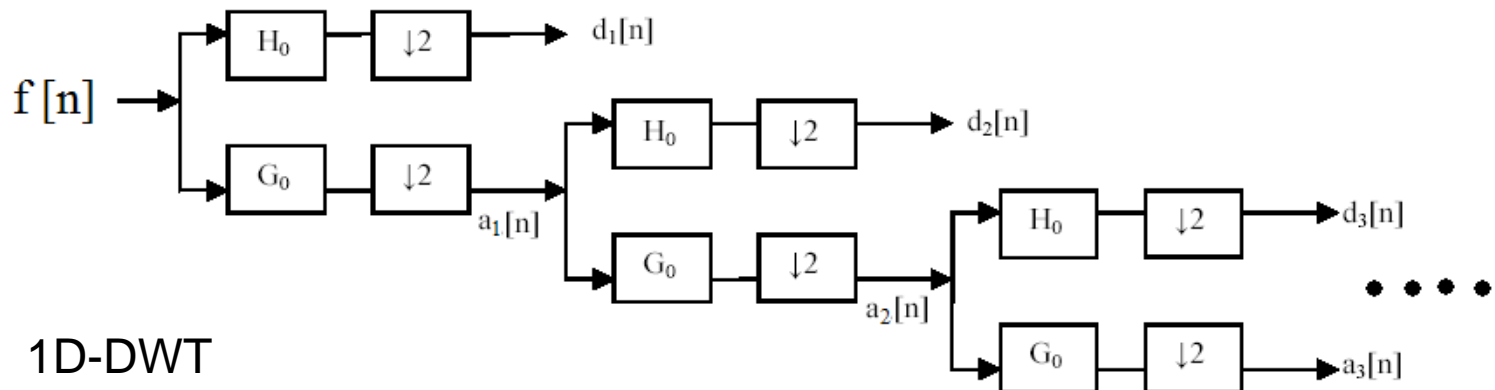
- There is only one DFT, but there are several families of DWTs:



- Wavelets are functions defined over a finite interval and having an average value of zero.
- Wavelet transform
 - Time–frequency resolution
 - DWT is successive low-pass (LP) and high-pass (HP) filtering.

A Hierarchical Pyramid Algorithm (aside)

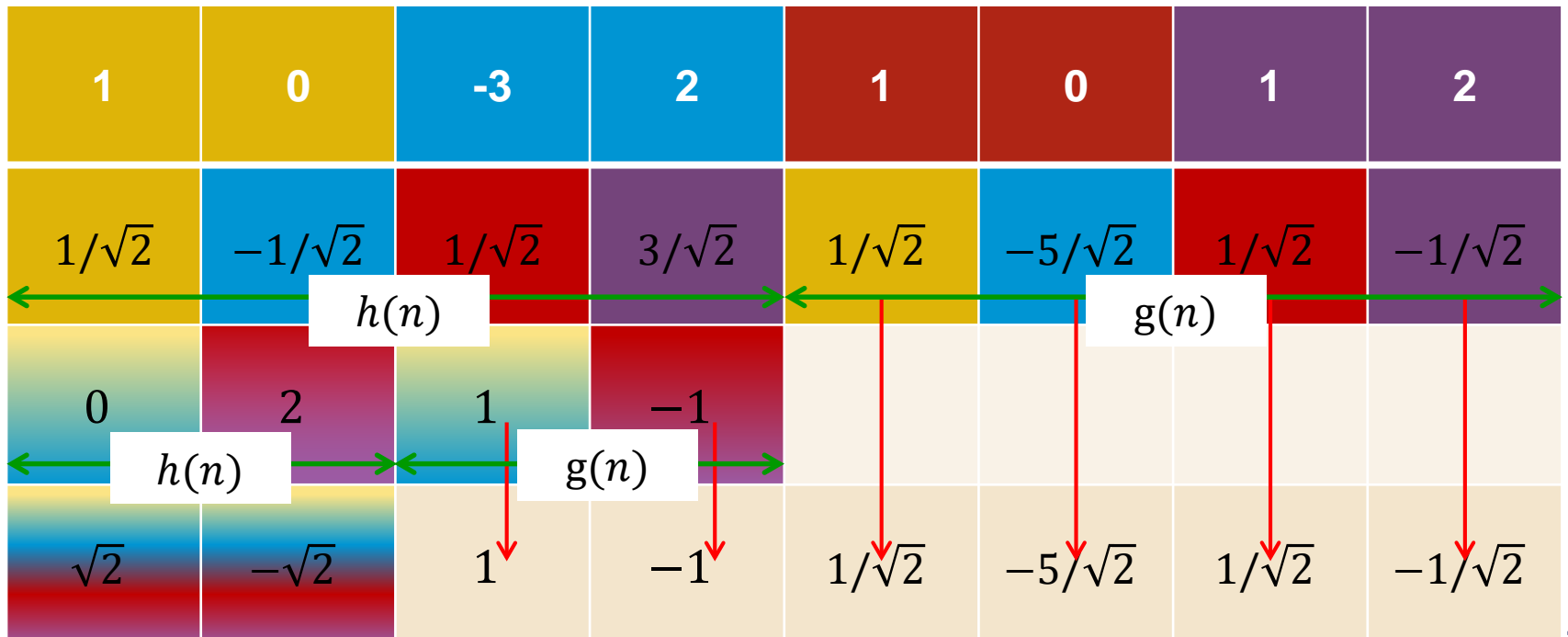
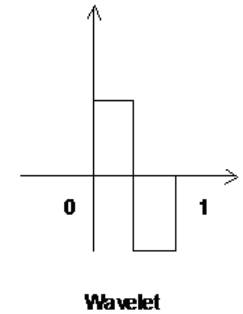
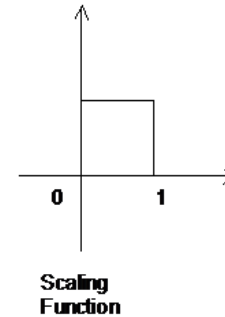
1. The length, L , of the input data vector must be an integer power of 2. This condition can be met by padding the data vector with zeros as necessary ($L \geq n$).
2. Each transform involves applying two functions. The first applies some data smoothing, such as a sum or weighted average. The second performs a weighted difference, which acts to bring out the detailed features of the data.
3. The two functions are applied to pairs of data points in X , that is, to all pairs of measurements (x_{2i}, x_{2i+1}) . This results in two data sets of length $L/2$. In general, these represent a smoothed or low-frequency version of the input data and the high-frequency content of it, respectively.
4. The two functions are recursively applied to the data sets obtained in the previous loop, until the resulting data sets obtained are of length 2.
5. Selected values from the data sets obtained in the previous iterations are designated the wavelet coefficients of the transformed data.



Example: Haar Wavelet Transform

$$h(n) = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

$$g(n) = \left[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} \right]$$



Reconstruction

$\sqrt{2}$	$-\sqrt{2}$	1	-1	$1/\sqrt{2}$	$-5/\sqrt{2}$	$1/\sqrt{2}$	$-1/\sqrt{2}$
------------	-------------	---	----	--------------	---------------	--------------	---------------

$$\mathbf{c}^{(0)} \begin{bmatrix} \sqrt{2} \end{bmatrix} \xrightarrow{\mathcal{H}^*} \begin{bmatrix} 1 & 1 \end{bmatrix}$$

$$\mathbf{d}^{(0)} \begin{bmatrix} -\sqrt{2} \end{bmatrix} \xrightarrow{\mathcal{G}^*} \begin{bmatrix} -1 & 1 \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 2 \end{bmatrix}$$

$$\mathbf{c}^{(1)} \begin{bmatrix} 0 & 2 \end{bmatrix} \xrightarrow{\mathcal{H}^*} \begin{bmatrix} 0 & 0 & \frac{2}{\sqrt{2}} & \frac{2}{\sqrt{2}} \end{bmatrix}$$

$$\mathbf{d}^{(1)} \begin{bmatrix} 1 & -1 \end{bmatrix} \xrightarrow{\mathcal{G}^*} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{3}{\sqrt{2}} \end{bmatrix}$$

$$\mathbf{c}^{(2)} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{3}{\sqrt{2}} \end{bmatrix} \xrightarrow{\mathcal{H}^*} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{3}{2} & \frac{3}{2} \end{bmatrix}$$

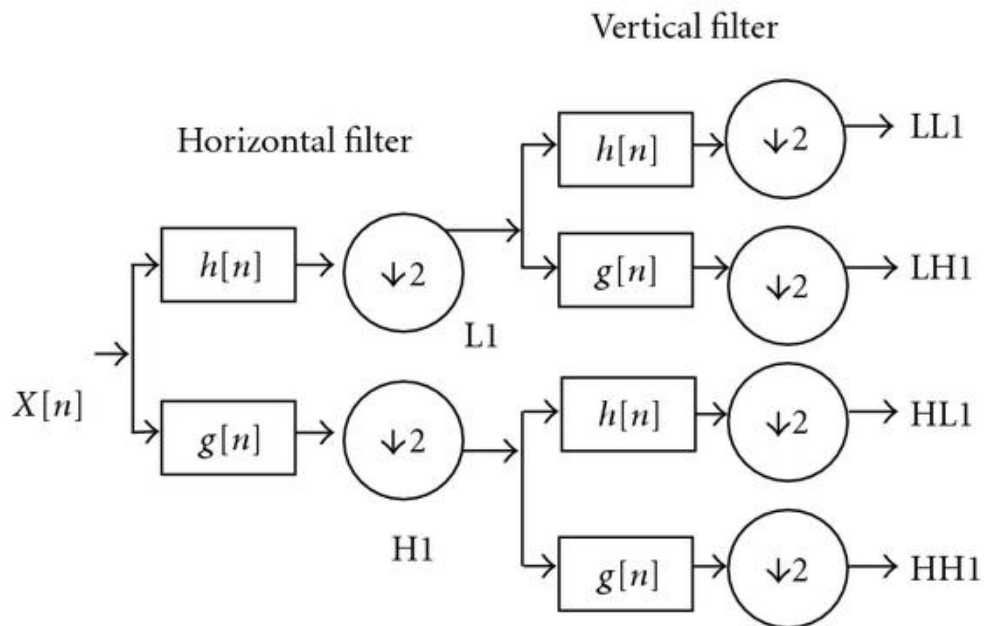
$$\mathbf{d}^{(2)} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{5}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \xrightarrow{\mathcal{G}^*} \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & -\frac{5}{2} & \frac{5}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$+ \begin{bmatrix} 1 & 0 & -3 & 2 & 1 & 0 & 1 & 2 \end{bmatrix}$$

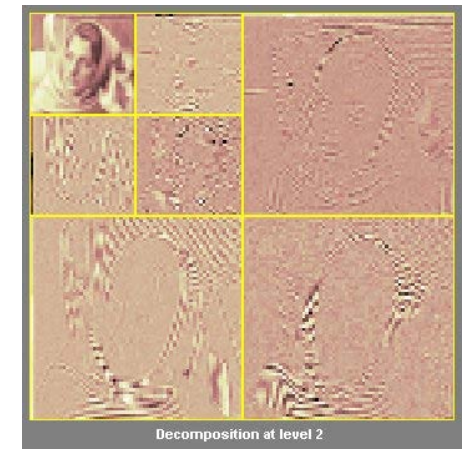
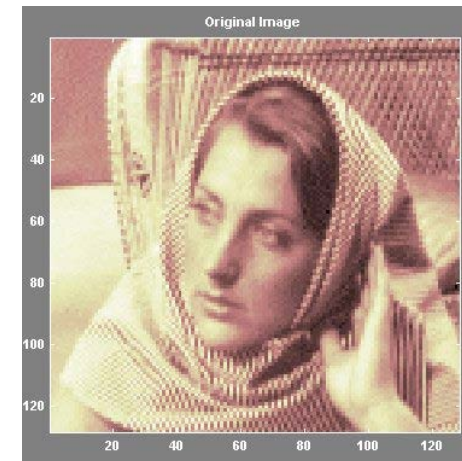
$$H^*(n) = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]$$

$$G^*(n) = \left[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} \right]$$

DWT for Image Processing



2D-DWT



- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients.
- Supports truncation.
- Effective removal of outliers:
 - High frequency subbands consist of the details.
 - High frequency subbands might be omitted without substantially affecting the main features of the data set.
 - Additionally, these small details are often those associated with noise; therefore, by **setting these coefficients to zero**, we can essentially remove the noise.
- Method:
 - Length, L , must be an integer power of 2 (padding with 0s)
- Difficult to apply on high dimensional data

- By factoring the matrix used into a product of a few sparse matrices, the resulting “fast DWT” algorithm has a complexity of $O(n)$ for an input vector of length n .
- Wavelet transforms can be applied to **multidimensional data** such as a data cube.
 - This is done by first applying the transform to the first dimension, then to the second, and so on.
 - The computational complexity involved is linear with respect to the number of cells in the cube.
- Wavelet transforms give good results on sparse or skewed data and on data with ordered attributes.
- Many real-world applications, including
 - the compression of fingerprint images, computer vision, analysis of time-series data, and data cleaning.

- **Data Reduction**

- Data Reduction Overview
- Attribute Reduction
 - Discrete Wavelet Transformation (DWT)
 - **Principle Component Analysis (PCA)**
 - Attribute Subset Selection
- Numerosity/Instance Reduction

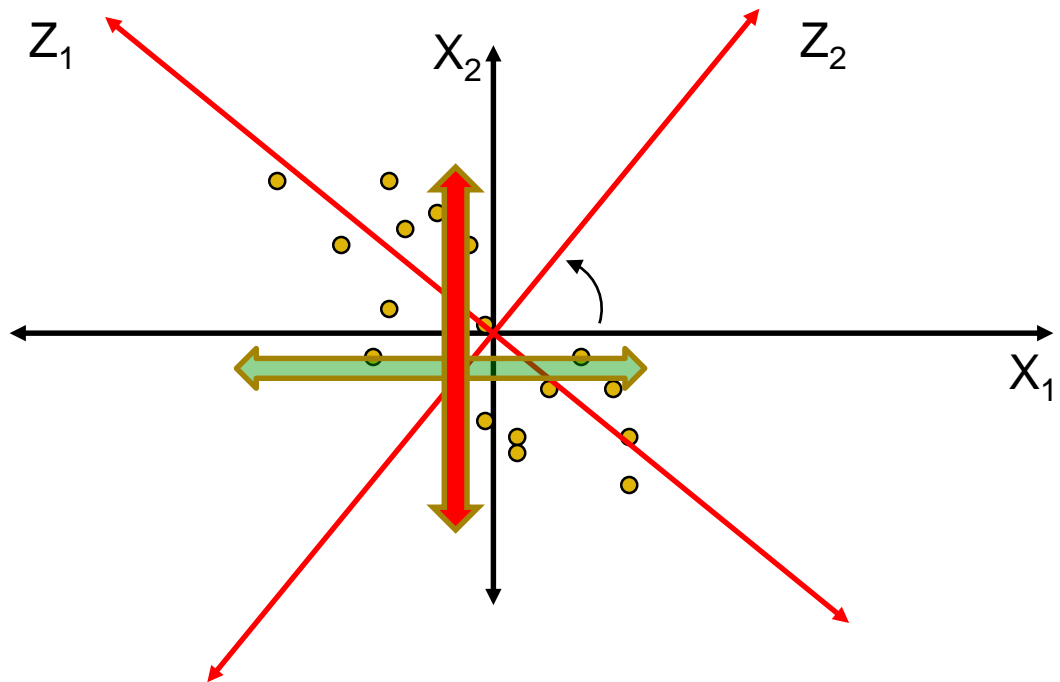
- **Data Enhancement**

- Data Augmentation
- Oversampling

Principal Component Analysis (PCA)

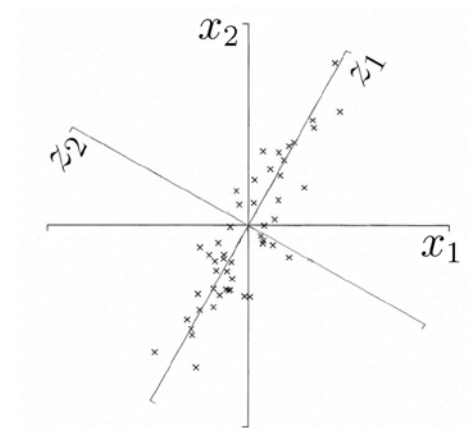
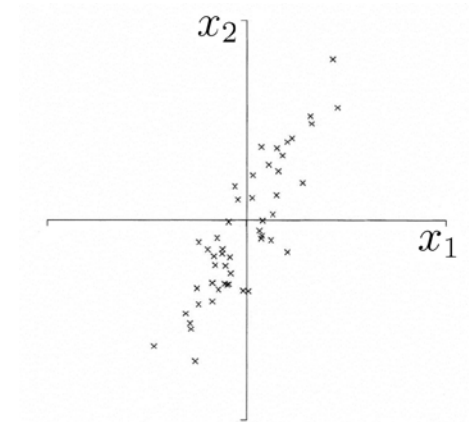
- Suppose we have attributes measured as p random variables (i.e. attributes) X_1, \dots, X_p .

- These random variables represent the p -axes of the Cartesian coordinate system in which the population (data points, instances) resides.
- Our goal is to develop a new set of p axes (linear combinations of the original p axes) in the directions of greatest variability.



Geometric Picture of Principal Components (PCs)

- The 1st PC Z_1 is a **minimum distance** fit to a line in the \mathbf{X} space
- the 2nd PC Z_2 is a **minimum distance** fit to a line in the plane perpendicular to the 1st PC
- PCs are a series of linear least squares fits to a sample, each **orthogonal** to all the previous.



Principal Component Analysis (Steps)

- Given N data vectors from n -dimensions, find $k \leq n$ orthogonal vectors (*principal components*) that can be best used to represent data
 - **Normalise** input data: Each attribute falls within the same range
 - Compute k orthonormal (unit) vectors, i.e., *principal components*
 - Each input data (vector) is a linear combination of the k principal component vectors
 - The principal components are **sorted in order of decreasing “significance” or strength**
 - Since the components are sorted, the size of the data can be reduced by **eliminating the weak components**, i.e., those with low variance (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data)
- Works for numeric data only. Complexity increases with size.

- **Data Reduction**

- Data Reduction Overview
- Attribute Reduction
 - Discrete Wavelet Transformation (DWT)
 - Principle Component Analysis (PCA)
 - **Attribute Subset Selection**
- Numerosity/Instance Reduction

- **Data Enhancement**

- Data Augmentation
- Oversampling

- Another way to reduce dimensionality of data
- **Redundant attributes**
 - Duplicate much or all of the information contained in one or more other attributes.
 - E.g. purchase price of a product and the amount of sales tax paid
- **Irrelevant attributes**
 - Contain no information that is useful for the data mining task at hand.
 - E.g. student ID is often **irrelevant** to the task of predicting students' GPA. Relevant attributes are 'credit hours', 'subject wise grade', ...

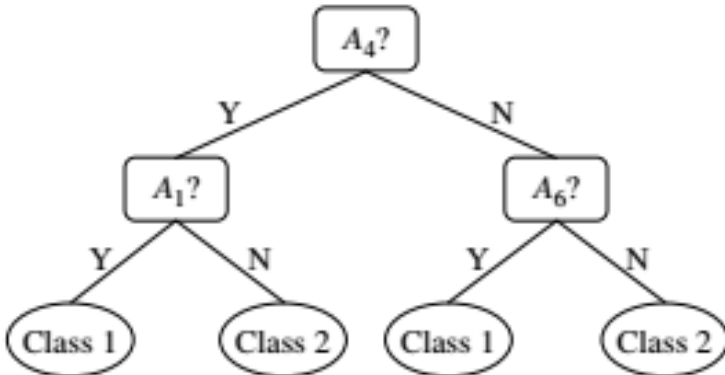
In dimensionality reduction, what is the difference between attribute selection and PCA?

- There are 2^d possible attribute combinations of d attributes
- Typical heuristic attribute selection methods:
 - **Best single attribute** under the attribute independence assumption: choose by significance tests (*information gain, decision tree, SVM, ..*)
 - Best step-wise **attribute selection**:
 - The **best** single-attribute is picked first
 - Then **next best** attribute condition to the first, ...
 - Step-wise **attribute elimination**:
 - Repeatedly eliminate the **worst** attribute
 - Best **combined attribute selection and elimination**
 - Optimal branch and bound:
 - Use attribute elimination and backtracking

Information Gain, Gini Index, etc. can be used for attribute selection.

Heuristic (Greedy) Search in Attribute Selection

- Best step-wise **attribute selection**:
 - The best single-attribute is picked first
 - Then next best attribute condition to the first, ...
- Step-wise **attribute elimination**:
 - Repeatedly eliminate the worst attribute
- Best **combined attribute selection and elimination**

Forward selection	Backward elimination	Decision tree induction
<p>Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$</p> <p>Initial reduced set: $\{\}$ $\Rightarrow \{A_1\}$ $\Rightarrow \{A_1, A_4\}$ \Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$</p>	<p>Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$</p> <p>$\Rightarrow \{A_1, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_4, A_5, A_6\}$ \Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$</p>	<p>Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$</p>  <pre> graph TD A4["A4?"] -- Y --> A1["A1?"] A4 -- N --> A6["A6?"] A1 -- Y --> C1_1((Class 1)) A1 -- N --> C2_1((Class 2)) A6 -- Y --> C1_2((Class 1)) A6 -- N --> C2_2((Class 2)) </pre> <p>\Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$</p>

- **Data Reduction**

- Data Reduction Overview
- Attribute Reduction
- **Numerosity/Instance Reduction**
 - Parametric methods:
 - a model is used to estimate the data.
 - Nonparametric methods:
 - storing reduced representations of the data
 - histograms, **clustering**, **sampling**, and **data cube aggregation**

- **Data Enhancement**

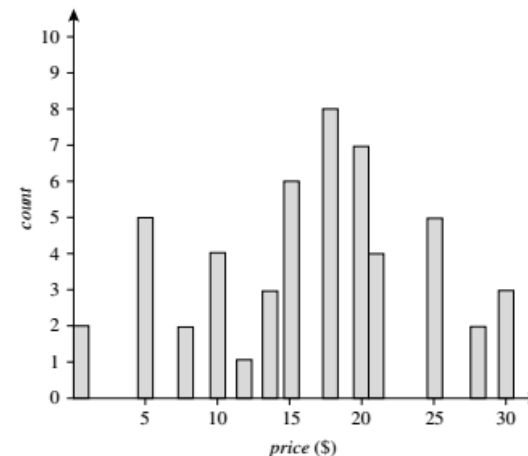
- Data Augmentation
- Oversampling

- Reduce data volume by choosing alternative, *smaller forms* of data representation
- Parametric methods (e.g. regression)
 - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
- Non-parametric methods
 - Major families: histograms, clustering, sampling, ...

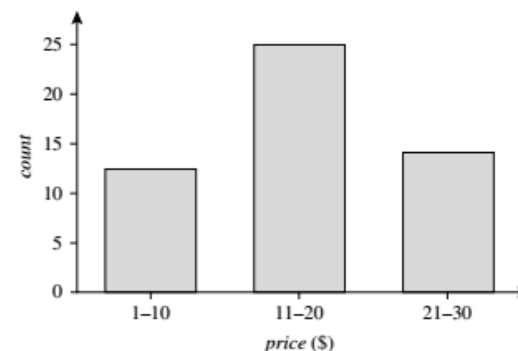
- **Linear regression: $Y = wX + b$**
 - Two regression coefficients, w and b , specify the line and are to be estimated by using the data at hand
 - Using the least squares criterion to the known values of $Y_1, Y_2, \dots, X_1, X_2, \dots$
- **Multiple regression (two or more predictors):**
 - $Y = b_0 + b_1 X_1 + b_2 X_2$
 - Many nonlinear functions can be transformed into the above
- **Log-linear models (log response): $\log Y_i = \alpha + \beta X_i + \epsilon_i$**
 - 1-unit increase in X multiplies the expected value of Y by e^β . Useful for skewed variables.
 - Useful for dimensionality reduction and data smoothing

Histogram Analysis

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
 - Equal-width: equal bucket range
 - Equal-frequency (or equal-depth)



A histogram for *price* using singleton buckets—each bucket represents one price-value/frequency pair.



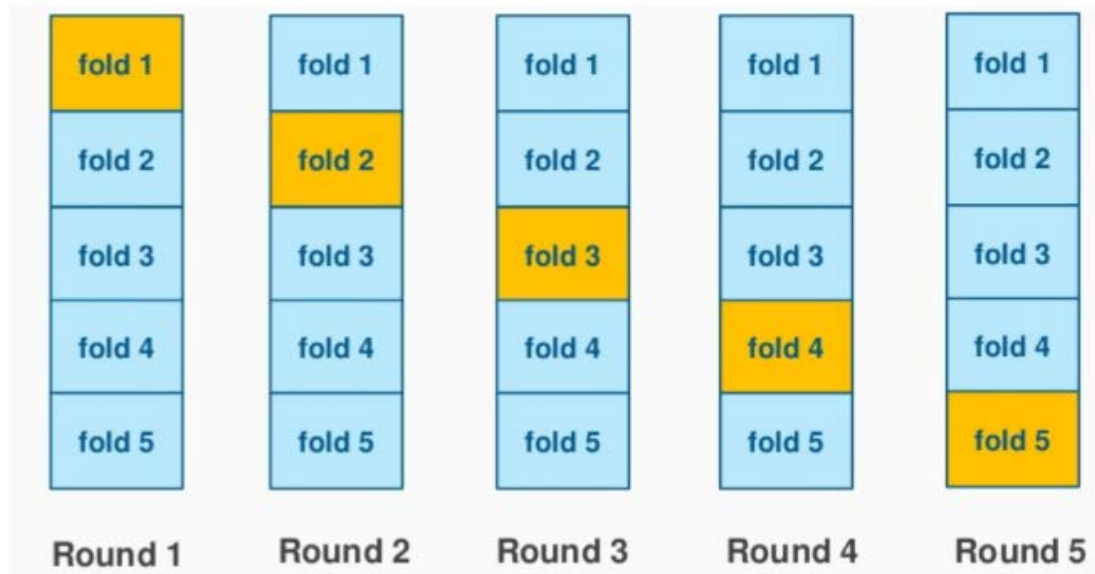
An equal-width histogram for *price*, where values are aggregated so that each bucket has a uniform width of \$10.

- Partition data set into clusters based on similarity, and store cluster representation (e.g. centroid and diameter) only.
- Can be very effective if data is clustered but not if data is “smeared”.
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures.
- There are many choices of clustering definitions and clustering algorithms.

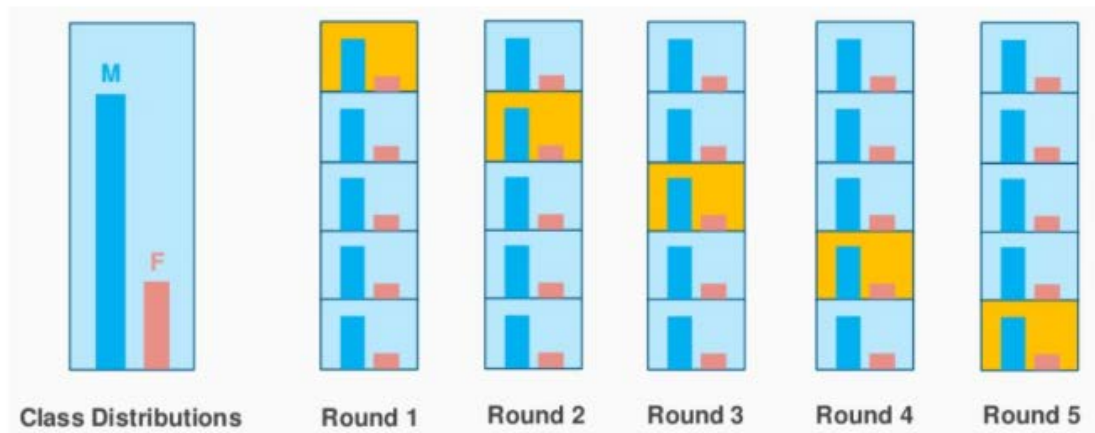
- Sampling: obtaining **a small sample s** to represent the whole data set N
- Cost of obtaining sample is proportional to s not N .
- Key principle: Choose a **representative** subset of the data
 - Simple random sampling may have very poor performance in the presence of skew
 - Develop adaptive sampling methods,
 - e.g. **stratified** sampling: data set is divided into mutually disjoint strata.
 - Ex: while drawing sample of customer: data can be divided according to age. This confirms representation from age group having smallest number of customers.

Stratified Cross-Validation (Lecture 9)

Example of 5 fold **Cross Validation**:



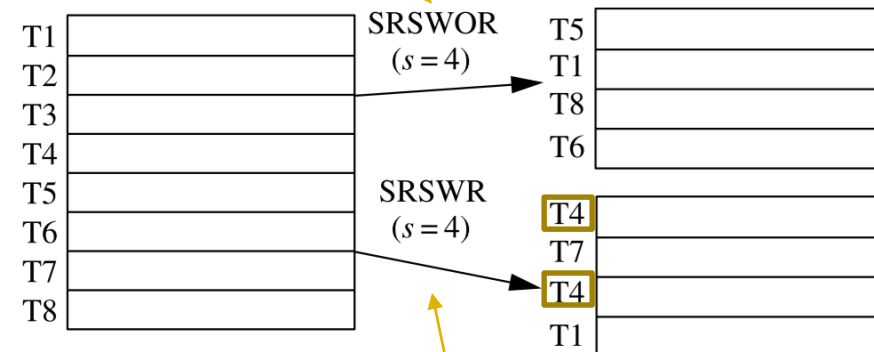
Example of 5 folds **Stratified Cross Validation**:



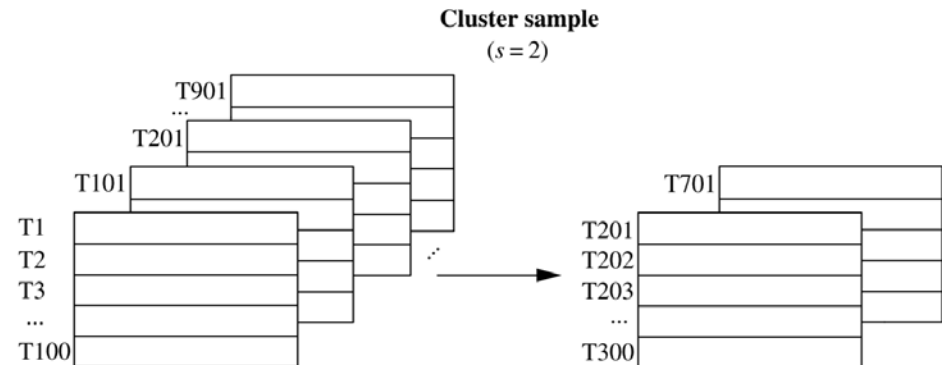
- **Simple random sampling (SRS)**
 - There is an equal probability of selecting any item
 - **Sampling without replacement (SRSWOR)**
 - Once an object is selected, it is **removed** from the population
 - **Sampling with replacement (SRSWR)**
 - A selected object is **not removed** from the population
- **Stratified sampling:**
 - Partition the data set into strata, and draw samples from each partition (proportionally, i.e. approximately the same percentage of the data)
 - Used in conjunction with skewed data

Examples of Sampling

Sampling **without** replacement



Sampling **with** replacement



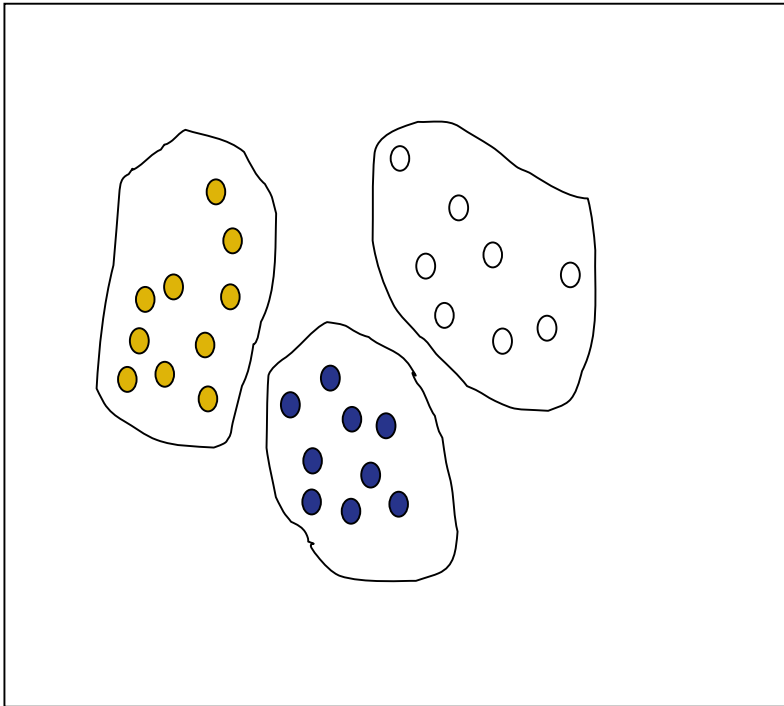
Stratified sample (according to age)

T38	youth
T256	youth
T307	youth
T391	youth
T96	middle_aged
T117	middle_aged
T138	middle_aged
T263	middle_aged
T290	middle_aged
T308	middle_aged
T326	middle_aged
T387	middle_aged
T69	senior
T284	senior

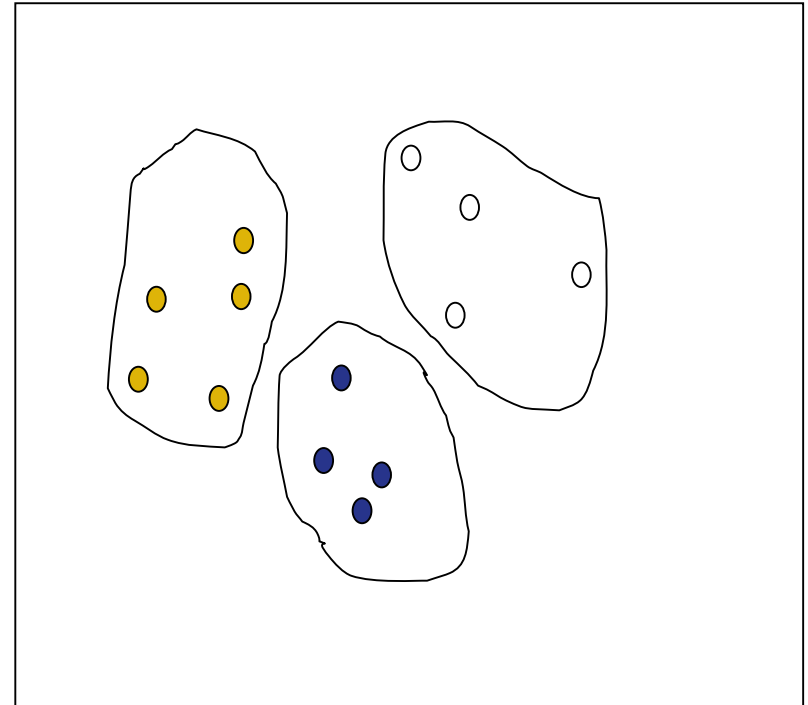
T38	youth
T391	youth
T117	middle_aged
T138	middle_aged
T290	middle_aged
T326	middle_aged
T69	senior

Clustering and Stratified Sampling

Raw Data



Cluster and Stratified Sample



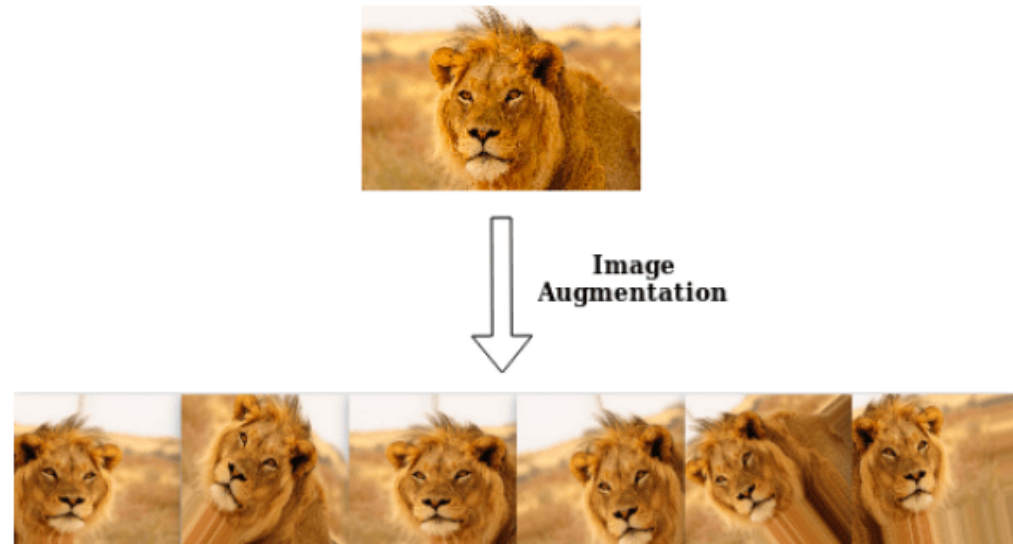
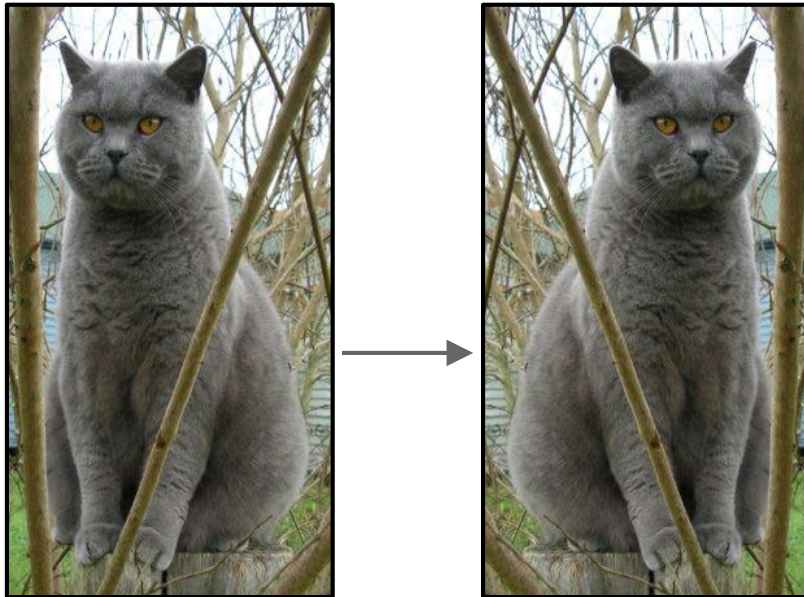
- The lowest level of a data cube is called *base cuboid*.
- High level of data cube is called *apex cuboid*.
 - The aggregated data for an **individual entity of interest**
 - E.g. sales data for *AllElectronics* company: branches can be grouped into regions, quarterly sales into yearly.
- Lowest level should be usable and contain all entity of interest.
- Reference appropriate levels
 - Use the smallest representation which is enough for a task
- Queries regarding aggregated information should be answered using data cube, when possible.

- **Data Reduction**
 - Data Reduction Overview
 - Attribute Reduction
 - Numerosity/Instance Reduction
- **Data Enhancement**
 - Data Augmentation
 - Oversampling

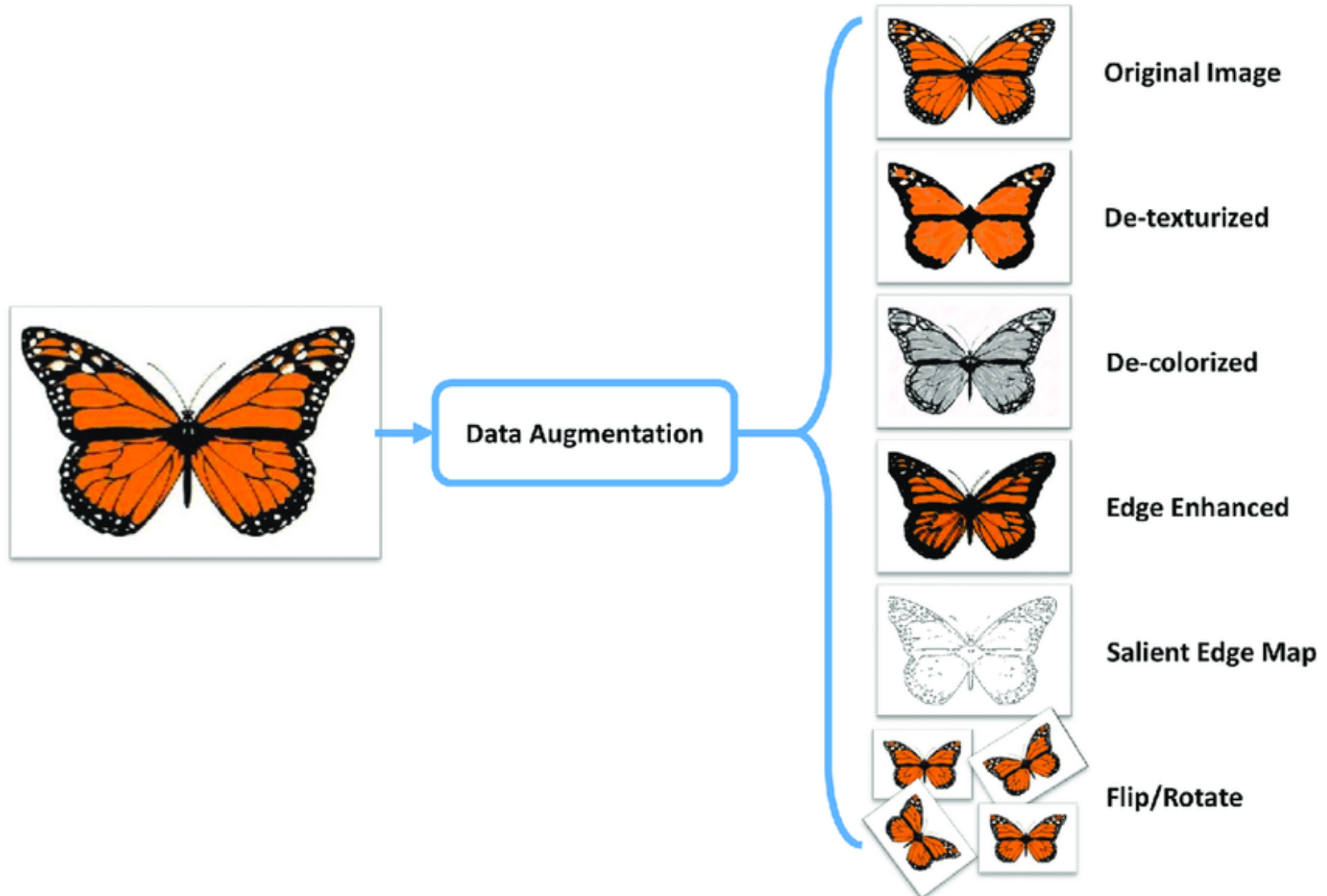
- Data augmentation in data analysis are techniques used to **increase** the amount of data by adding **slightly modified copies of already existing data** or newly created synthetic data from existing data.
- It is closely related to **oversampling** in data analysis.
 - using the same data more than once is oversampling.
 - Data augmentation involves creating new training data based on the existing data.

Data Augmentation Examples

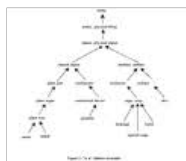
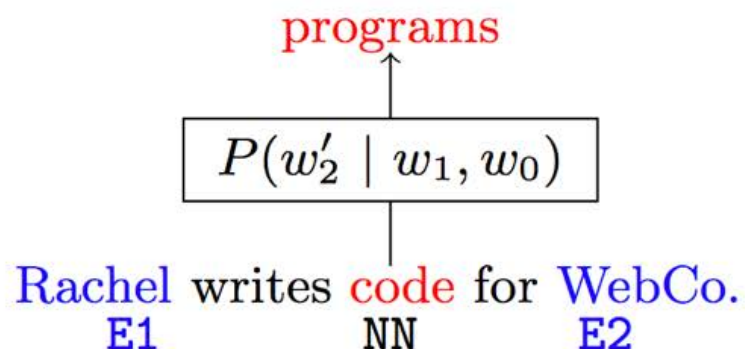
- Horizontal flips of an image; rotation, stretching,, ...



Data Augmentation Examples

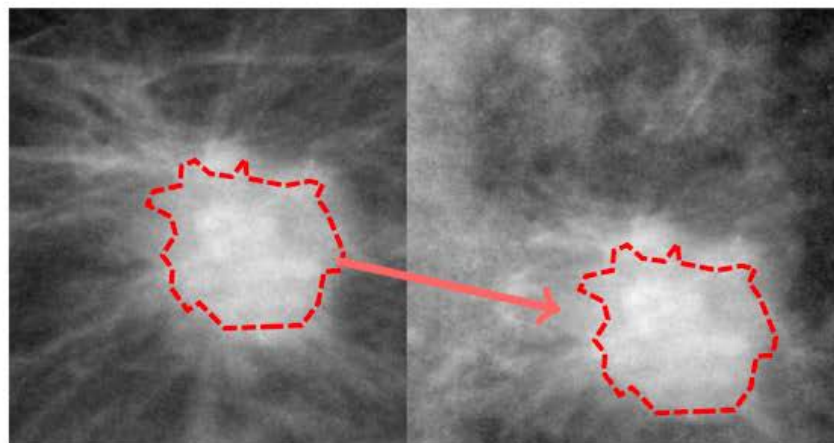


Text



- Synonymy
- Positional Swaps
- Etc...

Medical



Domain-specific transformations.

Ex:

1. *Segment tumor mass*
2. *Move*
3. *Resample background tissue*
4. *Blend*

- **Data Reduction**
 - Data Reduction Overview
 - Attribute Reduction
 - Numerosity/Instance Reduction
- **Data Enhancement**
 - Data Augmentation
 - **Oversampling**

Imbalanced Data

	Negative/healthy	Positive/cancerous
Number of cases	10,923	260
Category	Majority	Minority
Imbalanced accuracy	$\approx 100\%$	0-10 %

**Imbalance can be on the order of
100 : 1 up to 10,000 : 1!**

- **Expand the minority or shrink the majority**

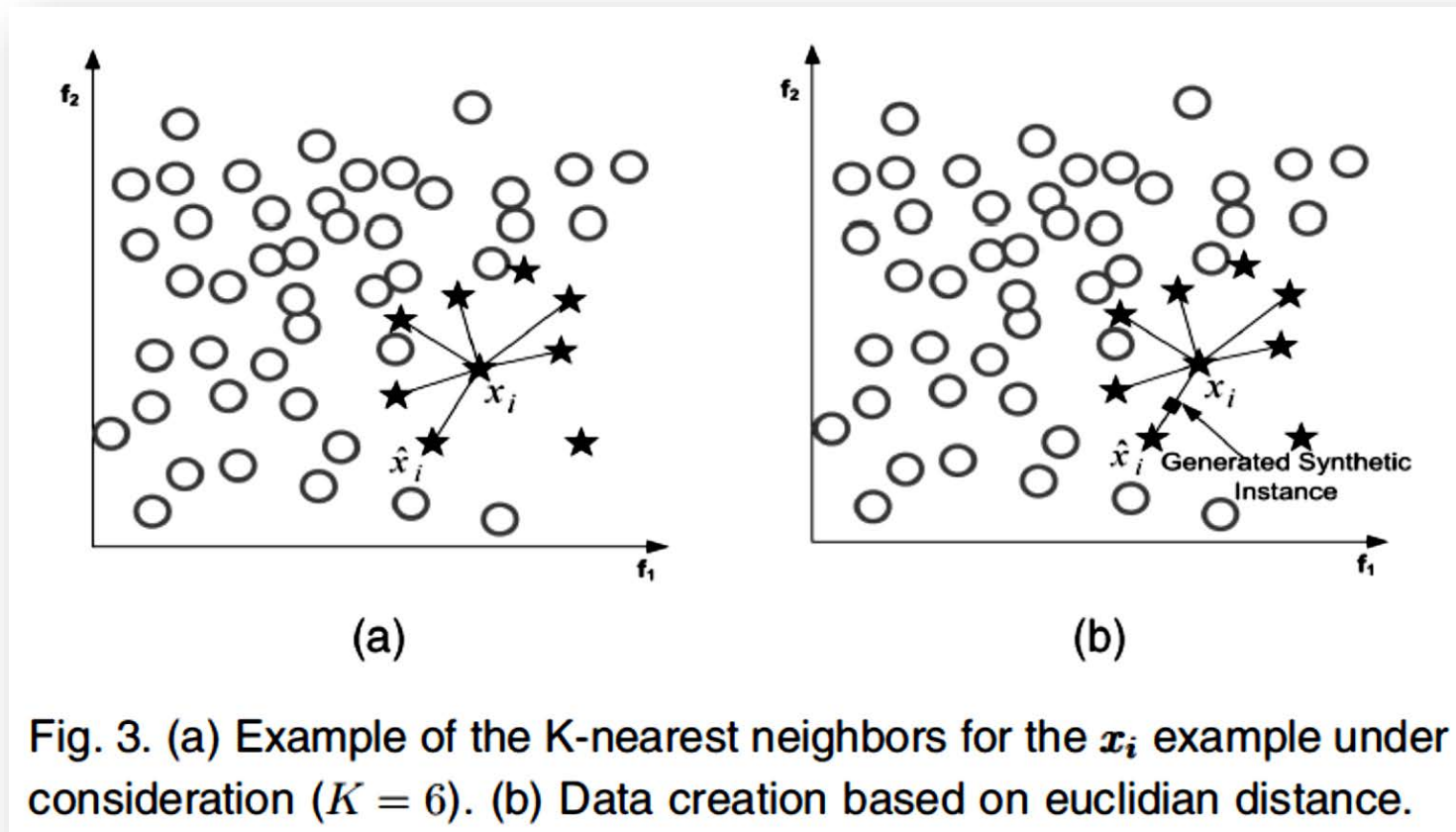
Random oversampling

- Expand the minority
- $|S'_{min}| \leftarrow |S_{min}| + |E|$
- $|S'| \leftarrow |S_{min}| + |S_{maj}| + |E|$
- Overfitting due to multiple “tied” instances

Random undersampling

- Shrink the majority
- $|S'_{maj}| \leftarrow |S_{maj}| - |E|$
- $|S'| \leftarrow |S_{min}| + |S_{maj}| - |E|$
- Loss of important concepts

- Synthetic minority oversampling technique (SMOTE)



- **Some slides are from**
 - http://www.cs.cmu.edu/afs/cs/academic/class/15385-s12/www/lec_slides/lec-18.ppt
- **Readings**
 - Chapter 3.4 of Han et al.'s book