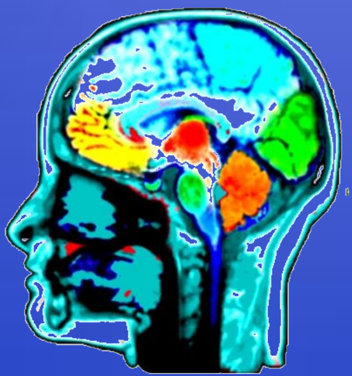
Introduction To Data Mining

Isfahan University of Technology (IUT)

Farvardin 1401



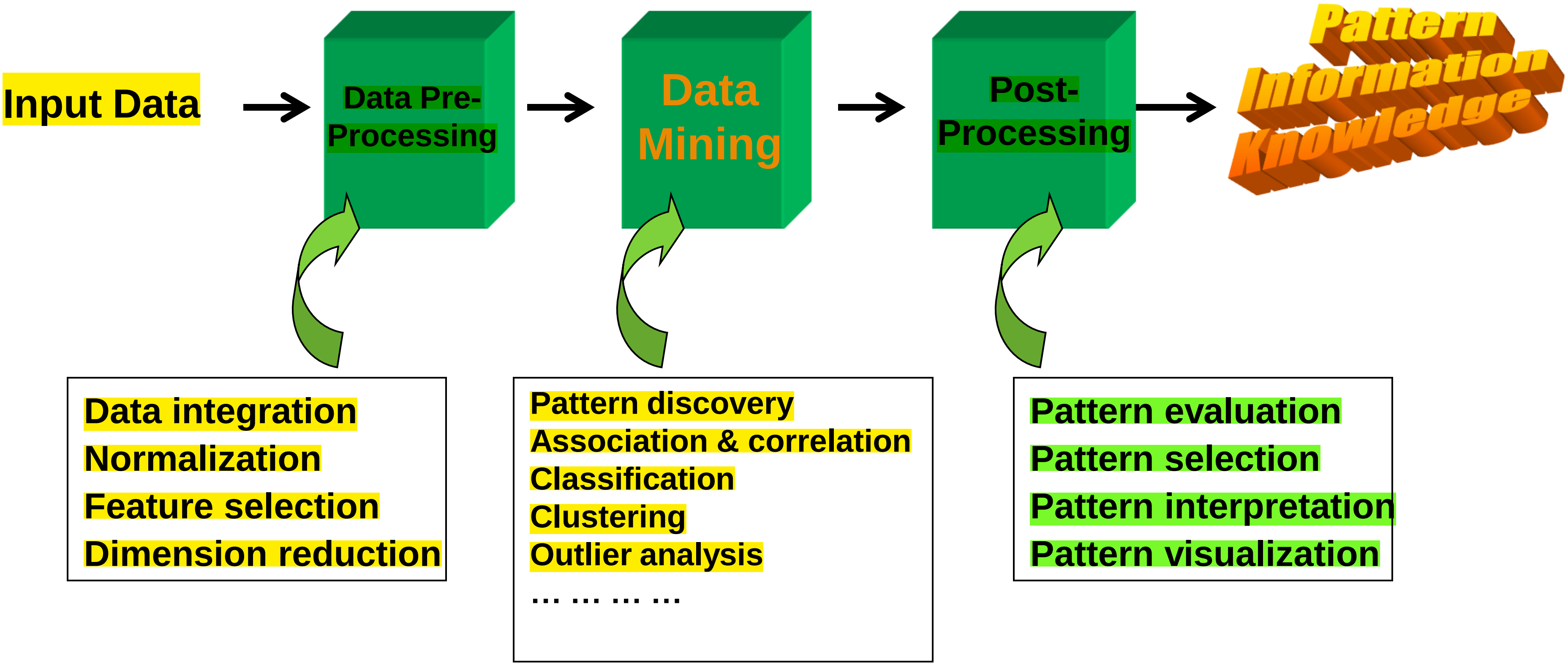
Preprocessing

Dr. Hamidreza Hakim
hamid.hakim.u@gmail.com

بسم الله الرحمن الرحیم

KDD Process: A Typical View from ML and Statistics

- This is a view from typical machine learning and statistics communities



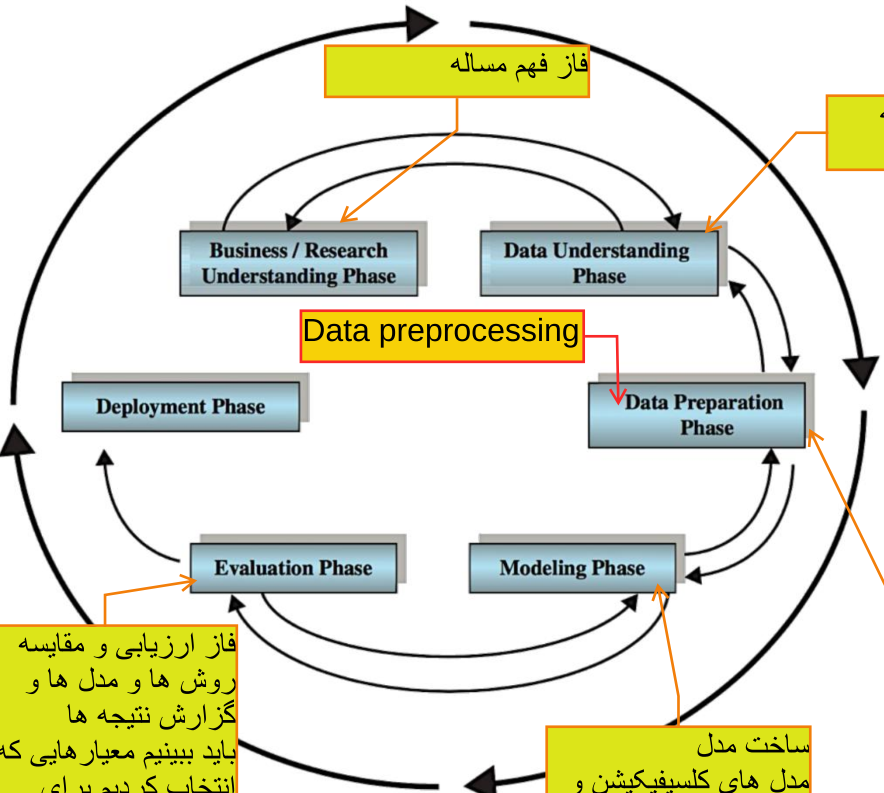
Standard process for data mining

- A cross-industry standard is clearly required, that is industry-neutral, toolneutral, and application-neutral.
- Wikipedia: Polls conducted at one and the same website (KDNuggets) in 2002, 2004, 2007 and 2014 show that CRISP-DM was the leading methodology used by industry data miners who decided to respond to the survey.
- CRISP-DM: Cross-Industry Standard Process for Data Mining.

CRISP-DM

چرخه ی حل مساله دیتا ماینینگ

فلش های برگشت برای وقتی است که ما یک کاری را میکنیم و میفهمیم که اشتباه است وباید برگردیم اصلاحش کنیم
مثلا میفهمیم دیتاهامون کافی نیستند



فاز فهم مساله

فاز فهم داده هایی که جمع اوری شده

فاز اماده سازی داده ها
شاید داده ها اماده نباشند و ما باید روشون کار کنیم
بحث های data preprocessing, data quality اینجا مطرح میشه
یتا رو به سمتی ببریم که باش بتونیم یه مدلی را استخراج کنیم

ساخت مدل
مدل های کلسیفیکیشن و پردیکشن و ...

فاز ارزیابی و مقایسه روش ها و مدل ها و گزارش نتیجه ها
باید ببینیم معیارهایی که انتخاب کردیم برای ارزیابی خوب هستند یا نه؟

CRISP-DM

1. Business/Research Understanding Phase
   – Clearly enunciate the project objectives and requirements.
   – Translate these goals into the formulation of a data mining problem.
   – Prepare a preliminary strategy for achieving these objectives.
2. Data Understanding Phase
   – Collect the data.
   – Use exploratory data analysis to familiarize yourself with the data, and discover initial insights.
   – Evaluate the quality of the data.
   – Select interesting subsets that may contain actionable patterns.
3. Data Preparation Phase
   – This labor-intensive phase covers all aspects of preparing the final data set, from the initial, raw, dirty data.
   – Select the cases and variables appropriate for your analysis.
   – Perform transformations on certain variables, if needed.
   – Clean the raw data so that it is ready for the modeling tools.
4. Modeling Phase
   – Select and apply appropriate modeling techniques.
   – Calibrate model settings to optimize results.
   – May require looping back to data preparation phase, in order to bring the form of the data into line with data mining technique.
5. Evaluation Phase
   – These models must be evaluated for quality and effectiveness.
   – Determine whether the model in fact achieves the objectives set for it in Phase 1.
   – Finally, come to a decision regarding the use of the data mining results.
6. Deployment Phase
   – Example of a simple deployment: Generate a report.
   – More complex: Implement a parallel data mining process in another department.
   – For businesses, the customer often carries out the deployment based on your model.

Outline

- Introduction
- Data Discretization

گسسته سازی داده ها

- Data Cleaning
- Data Integration

تجمیع داده ها

- Data Transformation

تبدیل داده ها

- Data Reduction

کاهش داده ها
چرا باید داده هایی که جمع کردیم را حذف کنیم؟ چقدرش رو باید حذف کنیم؟

- Summary

Why Preprocess the Data?

Data in the real world is dirty

- incomplete: lacking attribute values, lacking certain
  - attributes of interest, or containing only aggregate data
  - e.g., occupation=“ ”

یه جاهایی طرف
اطلاعات را پرنکرده

- noisy: containing errors or outliers
  - e.g., Salary=“-10”

اختلاف و تفاوت

اطلاعات غلط پرشده چون براش مهم نبوده

- inconsistent: containing discrepancies in codes or names
  - e.g., Age=“42” Birthday=“03/07/1997”
  - e.g., Was rating “1,2,3”, now rating “A, B, C”
  - e.g., discrepancy between duplicate records

داده ها با
رکوردهای
قبلی نمیخوانند
فرمت تاریخ
تولدش با سن
ش نمیخوانه

چقدر داده ها را میشناسی؟ چقدر
روی دیتا کلینینگ کار کردی؟

Why is data dirty?

چه اتفاقاتی روی داده ها افتاده؟ چطوری مدیریتشون کنیم؟

- Incomplete data may come from
  - “Not applicable” data value when collected
  - Different considerations between the time when the data was collected and when it is analyzed.
  - Human/hardware/software problems

زمان هایی که داده هامون از چندتا منبع داره میاد، مثلا از دوربین های مختلف داره میاد که هردوربین از یک سورس داره اطلاعات جمع میکنه

- Noisy data (incorrect values) may come from
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission
- Inconsistent data may come from
  - Different data sources
  - Functional dependency violation (e.g., modify some linked data)
  - Duplicate records also need data cleaning

Why is preprocessing important?

کیفیت داده یک بحث چندبعدی است.
وقتی میگیم داده مون باکیفیت است از مبناها و دیدگاه های مختلفی میتوانیم درباره ی کیفیتش صحبت کنیم
مثلا اینکه اطلاعاتی که اومده سمت ما درسته یا غلطه؟آیا اطلاعات بروز است؟
سازگاری رکوردها باهمدیگر
مثلا ایا همه رکوردها برای دانشجویان است یا وسطش یکی اومده رکوردهای کارمندان رو هم اضافه کرده

- No quality data, no quality mining results!
  - Quality decisions must be based on quality data
    - e.g., duplicate or missing data may cause incorrect or even misleading statistics.
  - Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse

Data Analytics

A multidimensional measure

- Measures for data quality: A multidimensional view
  - Accuracy: correct or wrong, accurate or not
  - Completeness: not recorded, unavailable, …
  - Consistency: some modified but some not, dangling, …
  - Timeliness: timely update?
  - Believability: how trustable the data are correct?
  - Interpretability: how easily the data can be understood?

Major Tasks in Data Preprocessing

- **Data cleaning**
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies

ثلا یه جاهایی سن دانشجویان با عدد گفته شده یه
جاهایی با تاریخ تولد گفته شده که بعد عدد سن رو
میشه از روش بدست اورد

- **Data integration**
  - Integration of multiple databases, data cubes, or files

معیار های اندازه گیری توی دیتابیس
ها متفاوت باشه مثلا یه جاهایی قد را
با متر گفتن یه جاهایی با سانتی متر

- **Data reduction**
  - Dimensionality reduction/feature reduction

چه زمان هایی باید دیتا را حذف کنیم؟
مثلا تعداد اتریبیوت ها خیلی زیاد باشه یا
فیچرهای بی ارزش داشته باشه
مثلا یه مساله ۲۰۰ تا ستون داره با ۱۰۰۰
تارکورد پس باید یه سری ویژگی ها را کنار
بذاریم

کاهش تعداد

  - Numerosity reduction
  - Data compression

مثلا حجم زیادی از
تصاویر را داریم که باید
فشرده کنیم و بعد ذخیره
کنیم

برای فشرده سازی میتونیم از تکنیک های
اشین لرنینگ هم استفاده کنیم و هوشمندانه عمل
کنیم مثلا یک تایم سری داریم که دنباله ای از
یمت کالاهاست یه راه اینه که دنباله قیمت کالاها
و روزها را نگه داریم، یه راه اینه که یک مدل
تایم سری پردیکشن روی این داده ها بزنیم وبه
جای ذخیره کردن خود داده ها این مدل را ذخیره
کنیم و ازش استفاده کنیم مثلا از یک شبکه
عصبی استفاده کنیم این مدله یک نماینده از داده
های ماست در جاهایی که حجم داده ها خیلی
زیاد است این کار به درد میخوره

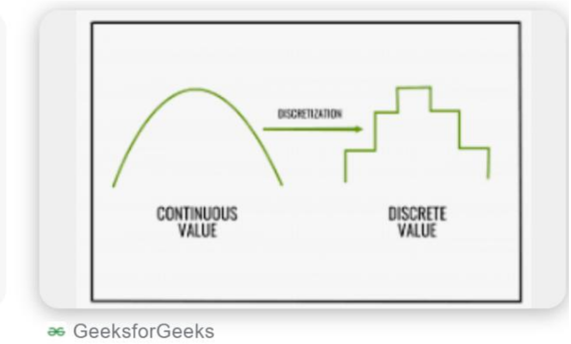
- **Data transformation and discretization**
  - Normalization

تبدیل داده ها :
اتربیوت های قد و وزن داریم که مقیاس اینها به هم نمیخوره یکی
کیلوگرمه یکی سانتی متره
الگوریتم فقط صفر و یک میشناسه پس باید اینها را نرمال کنیم که
مقادیر بزرگ قد تاثیر بیشتری نگیرن روی مدل ساختن

DATA DISCRETIZATION

Discretization

- Discretization: Divide the range of a continuous attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Reduce data size by discretization
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
  - Prepare for further analysis, e.g., classification



Simple Discretization: Binning

- Equal-width (distance) partitioning
  - Divides the range into *N* intervals of equal size: uniform grid
  - if *A* and *B* are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$.
  - The most straightforward, but outliers may dominate presentation
  - Skewed data is not handled well

Simple Discretization: Binning

- Equal-depth (frequency) partitioning
  - Divides the range into *N* intervals, each containing approximately same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky

Binning Methods for Data Smoothing

- Sorted data for price (in dollars):

  4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

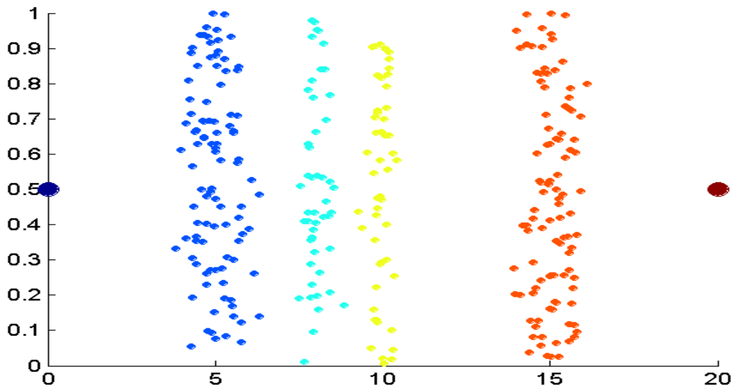
* Partition into equal-frequency (**equi-depth**) bins:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34

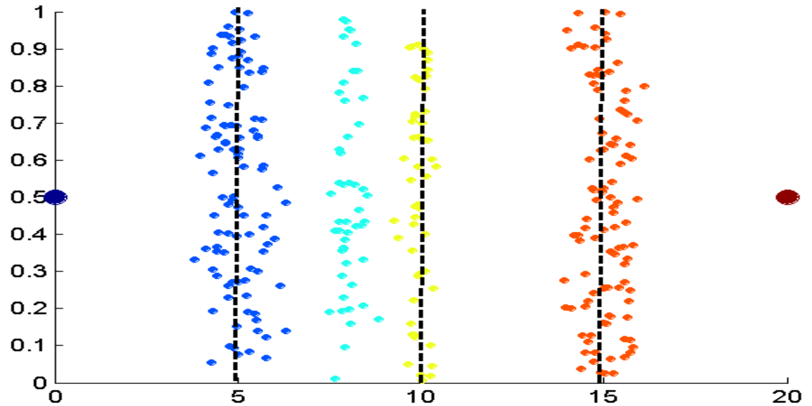
Data Discretization Methods

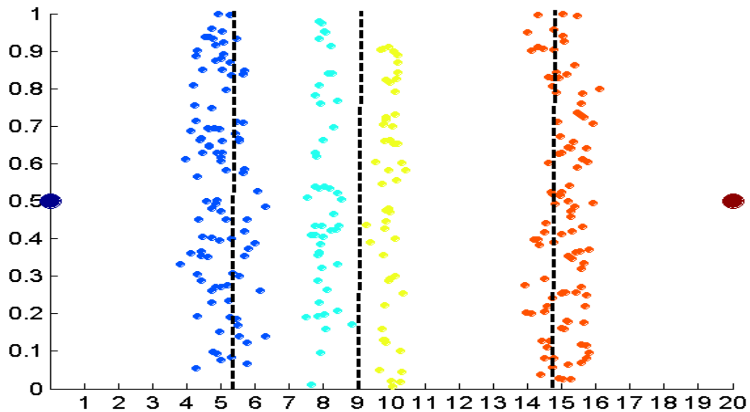
- Typical methods: All the methods can be applied recursively
  - Binning
    - Top-down split, unsupervised
  - Histogram analysis
    - Top-down split, unsupervised
  - Clustering analysis (unsupervised, top-down split or bottom-up merge)
  - Decision-tree analysis (supervised, top-down split)
  - …

Discretization Without Using Class Labels (Binning vs. Clustering)

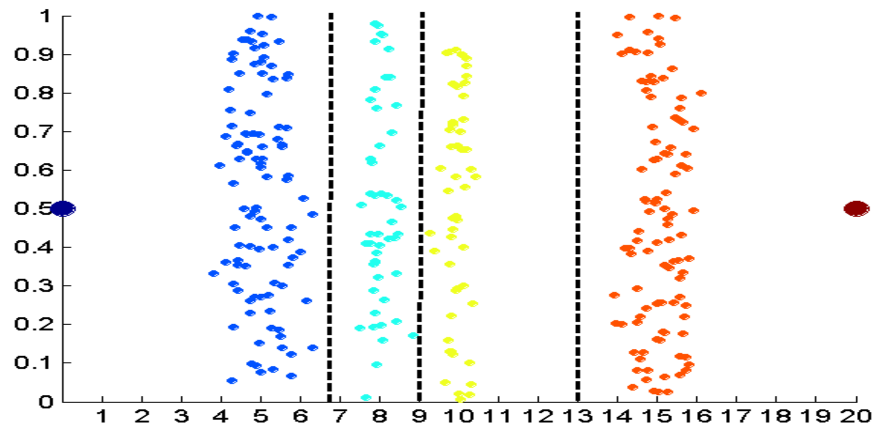
Data







Equal frequency (binning)



K-means clustering leads to better results

Discretization by Classification & Correlation Analysis

- Classification (e.g., decision tree analysis)
  - Supervised: Given class labels, e.g., cancerous vs. benign
  - Using *entropy* to determine split point (discretization point)
  - Top-down, recursive split
  - Details to be covered in Chapter 7

DATA CLEANING

Why data cleaning?

Importance

"Data cleaning is
one of the three biggest problems in data warehousing"

Ralph Kimball

"Data cleaning is
the number one problem in data warehousing"

DCI survey

Data Cleaning

- Data in the Real World Is Dirty: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, transmission error
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., *Occupation*=“ ” (missing data)
  - noisy: containing noise, errors, or outliers
    - e.g., *Salary*=“−10” (an error)
  - inconsistent: containing discrepancies in codes or names, e.g.,
    - *Age*=“42”, *Birthday*=“03/07/2010”
    - Was rating “1, 2, 3”, now rating “A, B, C”
    - discrepancy between duplicate records
  - Intentional (e.g., *disguised missing* data)
    - Jan. 1 as everyone’s birthday?

Data cleaning tasks

- Fill in missing values
- Identify outliers and smooth out noisy data
- Correct inconsistent data
- Resolve redundancy caused by data integration

Incomplete (Missing) Data may be due

- equipment malfunction
- inconsistent with other recorded data and thus deleted
- data not entered due to misunderstanding
- certain data may not be considered important at the time of entry
- not register history or changes of the data

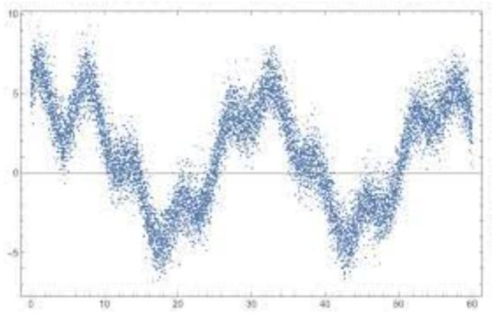
Missing data may need to be inferred!!

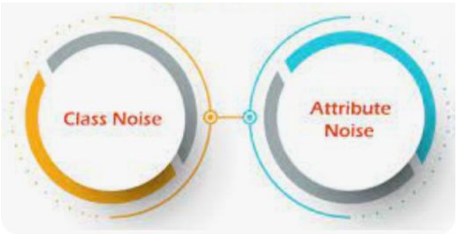
How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (when doing classification)—not effective when the % of missing values per attribute varies considerably
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
  - a global constant : e.g., “unknown”, a new class?!
  - the attribute mean
  - the attribute mean for all samples belonging to the same class: smarter
  - the most probable value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to
  - faulty data collection instruments
  - data entry problems
  - data transmission problems
  - technology limitation
  - inconsistency in naming convention





https://www.javatpoint.com/what-is-noise-in-data-mining

How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.

Binning Methods for Data Smoothing

- Sorted data for price (in dollars):

  4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* Partition into equal-frequency (**equi-depth**) bins:

- Bin 1: 4, 8, 9, 15
- Bin 2: 21, 21, 24, 25
- Bin 3: 26, 28, 29, 34

* Smoothing by **bin means**:

- Bin 1: 9, 9, 9, 9
- Bin 2: 23, 23, 23, 23
- Bin 3: 29, 29, 29, 29

* Smoothing by **bin boundaries**:

- Bin 1: 4, 4, 4, 15
- Bin 2: 21, 21, 25, 25
- Bin 3: 26, 26, 26, 34

Smoothing by bin boundaries is a technique used in histogram construction to reduce the noise caused by small fluctuations in data. In this technique, the values of neighboring bins are combined into a single bin to create smoother histograms.

Here's an example to illustrate the concept:

Suppose we have the following data set containing 10 values:
{2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

We want to construct a histogram with 4 bins of equal width. The bin width would be (11-2)/4=1.75.

Without smoothing by bin boundaries, the histogram would look like this:

Bin	Frequency
[2, 3.75)	1
[3.75, 5.5)	2
[5.5, 7.25)	2
[7.25, 9)	3
[9, 11]	2

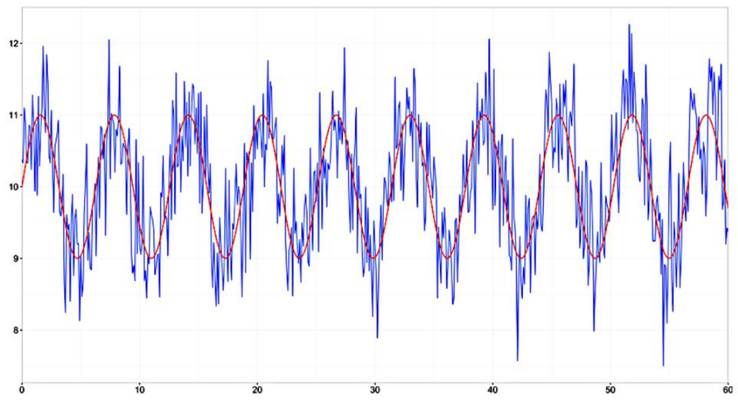
However, if we apply smoothing by bin boundaries, we can combine the first two bins and last two bins to create a smoother histogram. We take the lower boundary of the first bin and the upper boundary of the second bin as the new boundaries for the first bin, and the lower boundary of the fourth bin and the upper boundary of the fifth bin as the new boundaries for the last bin. The resulting histogram would look like this:

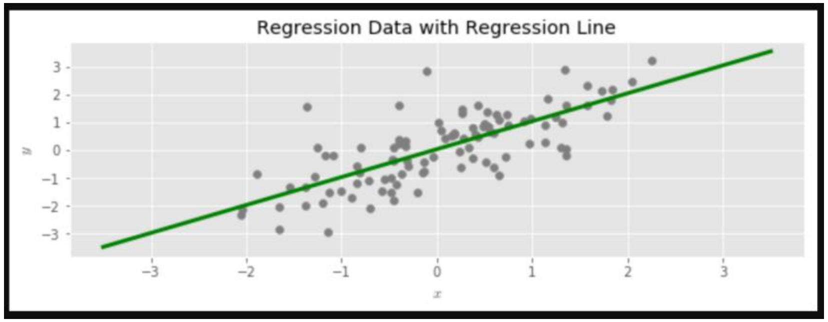
Bin	Frequency
[2, 5.5)	3
[5.5, 7.25)	2
(7.25, 9.5]	5

As you can see, the smoothed histogram has fewer bins and is less noisy than the unsmoothed histogram.

How to Handle Noisy Data?

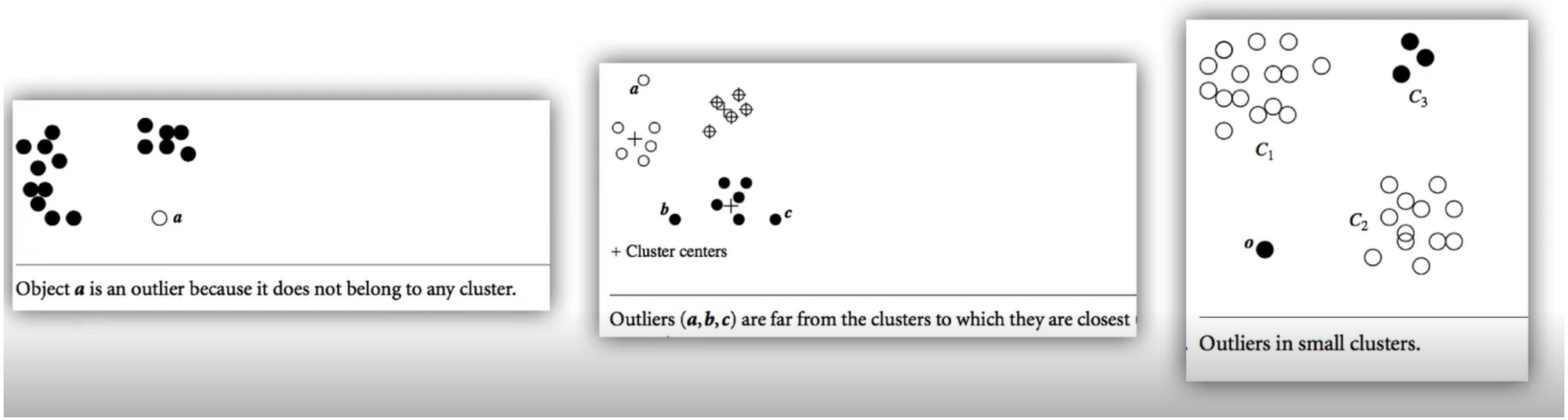
- Regression
  - smooth by fitting the data into regression functions





How to Handle Noisy Data?

- Clustering
  - detect and remove outliers



How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by human (e.g., deal with possible outliers)

DATA INTEGRATION

Data Integration

- **Data integration**:
  - Combines data from multiple sources into a coherent store
- Schema integration: e.g., A.cust-id ≡ B.cust-#
  - Integrate metadata from different sources
- Entity identification problem:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
- Detecting and resolving data value conflicts
  - For the same real world entity, attribute values from different sources are different
  - Possible reasons: different representations, different scales, e.g., metric vs. British units

34

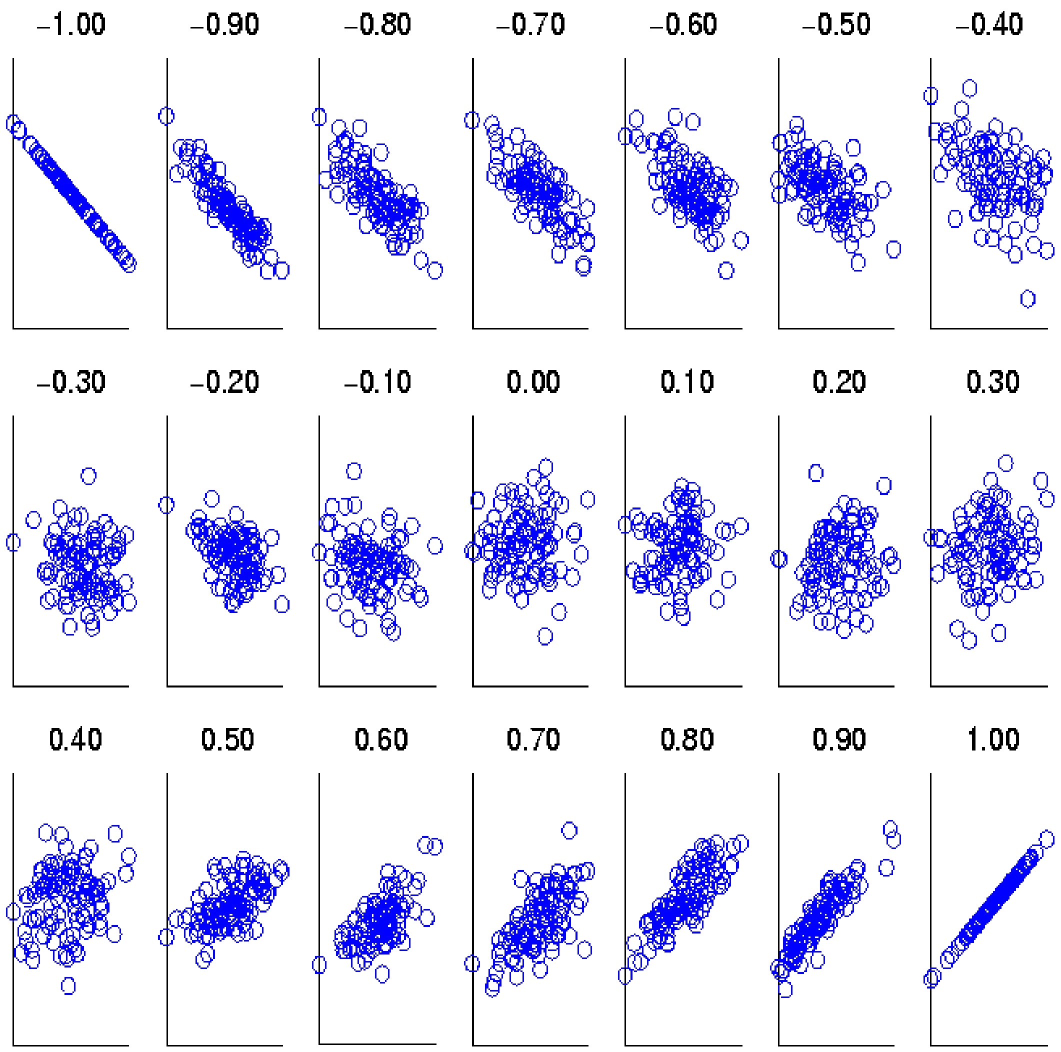
Handling Redundancy

- Redundant data occur often when integration of multiple databases
  - *Object identification*: The same attribute or object may have different names in different databases
  - *Derivable data:* One attribute may be a "derived" attribute in another table, e.g., annual revenue

Redundant attributes may be able to be detected by *correlation analysis* and *covariance analysis*

Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.

Correlation Analysis (Nominal Data)

- **X^2 (chi-square) test**

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- The larger the X^2 value, the more likely the variables are related
- The cells that contribute the most to the X^2 value are those whose actual count is very different from the expected count
- Correlation does not imply causality
  - # of hospitals and # of car-theft in a city are correlated
  - Both are causally linked to the third variable: population

Chi-Square Calculation: An Example

	Play chess	Not play chess	Sum (row)
Like science fiction	250(90)	200(360)	450
Not like science fiction	50(210)	1000(840)	1050
Sum(col.)	300	1200	1500

- X^2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

- It shows that like_science_fiction and play_chess are correlated in the group

Correlation Analysis (Numeric Data)

- Correlation coefficient (also called Pearson's product moment coefficient)

$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum_{i=1}^{n}(a_i b_i) - n\bar{A}\bar{B}}{(n-1)\sigma_A \sigma_B}$$

where n is the number of tuples, $\bar{A}$ and $\bar{B}$ are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(a_i b_i)$ is the sum of the AB cross-product.

- If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.
- $r_{A,B} = 0$: independent; $r_{AB} < 0$: negatively correlated

Correlation (viewed as linear relationship)

- Correlation measures the linear relationship between objects
- To compute correlation, we standardize data objects, A and B, and then take their dot product

$$a'_k = (a_k - mean(A)) / std(A)$$

$$b'_k = (b_k - mean(B)) / std(B)$$

$$correlation(A, B) = A' \bullet B'$$

Covariance (Numeric Data)

- Covariance is similar to correlation

$$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n} \qquad r_{A,B} = \frac{Cov(A,B)}{\sigma_A \sigma_B}$$

  where n is the number of tuples, and are the respective mean or **expected values** of A and B, σ_A and σ_B are the respective standard deviation of A and B.

- **Positive covariance**: If $Cov_{A,B} > 0$, then A and B both tend to be larger than their expected values.
- **Negative covariance**: If $Cov_{A,B} < 0$ then if A is larger than its expected value, B is likely to be smaller than its expected value.
- **Independence**: $Cov_{A,B} = 0$ but the converse is not true:
  - Some pairs of random variables may have a covariance of 0 but are not independent. Only under some additional assumptions (e.g., the data follow multivariate normal distributions) does a covariance of 0 imply independence

Co-Variance: An Example

$$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}$$

- It can be simplified in computation as

$$Cov(A,B) = E(A \cdot B) - \bar{A}\bar{B}$$

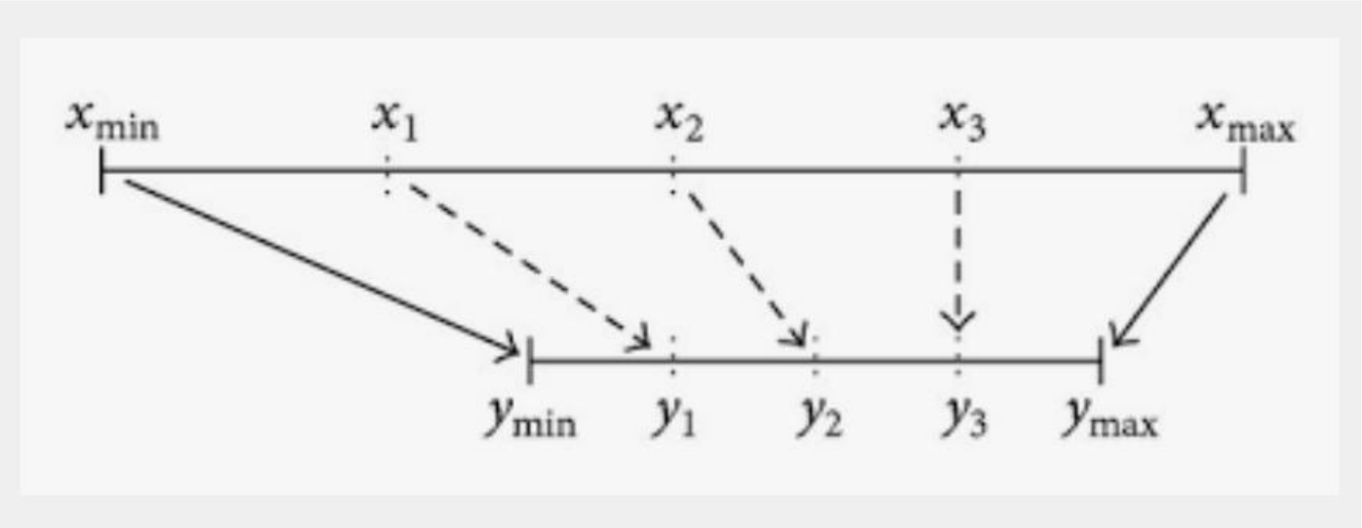
- Suppose two stocks A and B have the following values in one week: (2, 5), (3, 8), (5, 10), (4, 11), (6, 14).
- Question: If the stocks are affected by the same industry trends, will their prices rise or fall together?
  - E(A) = (2 + 3 + 5 + 4 + 6)/ 5 = 20/5 = 4
  - E(B) = (5 + 8 + 10 + 11 + 14) /5 = 48/5 = 9.6
  - Cov(A,B) = (2×5+3×8+5×10+4×11+6×14)/5 − 4 × 9.6 = 4
- Thus, A and B rise together since Cov(A, B) > 0.

DATA TRANSFORMATION

Data Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values s.t. each old value can be identified with one of the new values
- Methods
  - Smoothing: Remove noise from data
  - Normalization: Scaled to fall within a smaller, specified range
    - min-max normalization
    - z-score normalization
    - normalization by decimal scaling
  - Attribute/feature construction
    - New attributes constructed from the given ones
  - Aggregation: Summarization, data cube construction
  - Discretization: Concept hierarchy climbing

Normalization



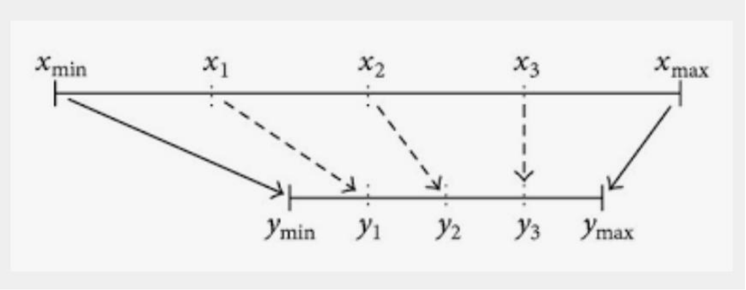
Normalization

- **Min-max normalization**: to [new_min$_A$, new_max$_A$]

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

  - Ex. Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to

$$\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$$



Normalization

- **Z-score normalization** (μ: mean, σ: standard deviation):
  - Ex. Let μ = 54,000, σ = 16,000. Then

$$v' = \frac{v - \mu_A}{\sigma_A}$$

$$\frac{73,600 - 54,000}{16,000} = 1.225$$

DATA REDUCTION

Data Reduction Strategies

- **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? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.

Data Reduction Strategies

- Dimensionality reduction, e.g., remove unimportant attributes
  - Wavelet transforms
  - Principal Components Analysis (PCA)
  - Feature subset selection, feature creation
- Numerosity reduction (some simply call it: Data Reduction)
  - Regression and Log-Linear Models
  - Histograms, clustering, sampling
  - Data cube aggregation
- Data compression

Data Reduction 1: Dimensionality Reduction

- **Curse of dimensionality**
  - When dimensionality increases, data becomes increasingly sparse
  - Density and distance between points, which is critical to clustering, outlier analysis, becomes less meaningful
  - The possible combinations of subspaces will grow exponentially

Data Reduction 1: Dimensionality Reduction

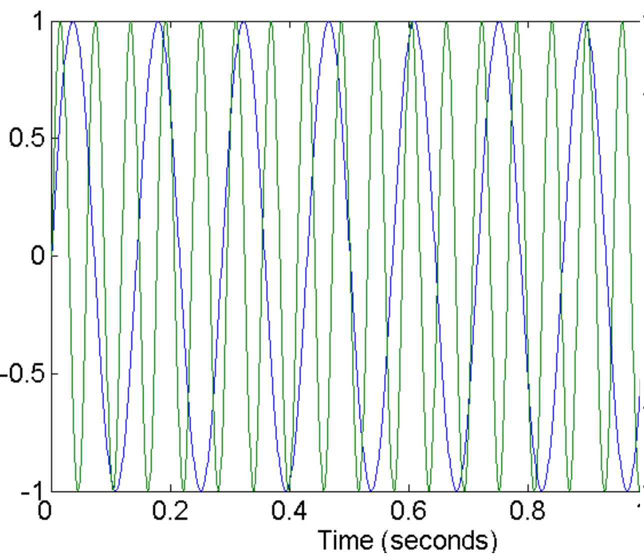
- **Dimensionality reduction**
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization

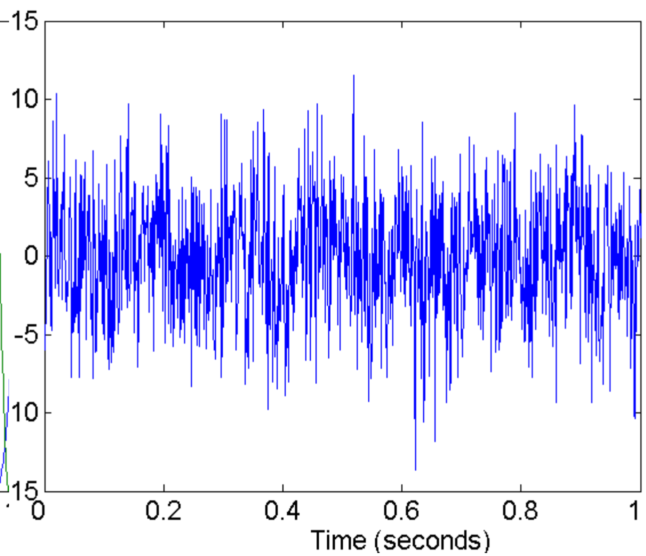
Data Reduction 1: Dimensionality Reduction

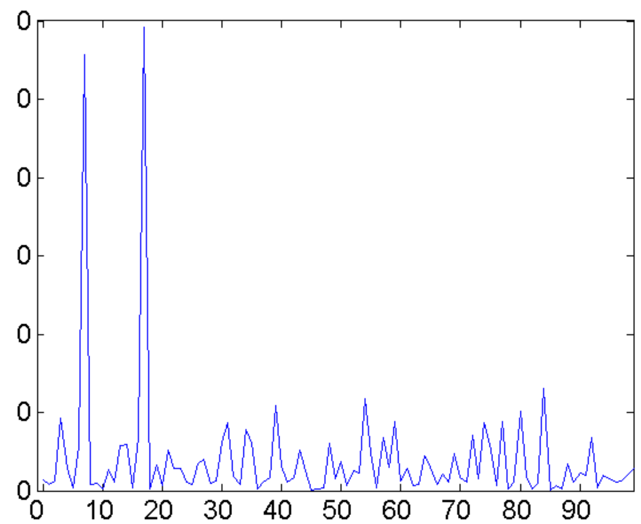
- **Dimensionality reduction techniques**
  - Wavelet transforms
  - Principal Component Analysis
  - Supervised and nonlinear techniques (e.g., feature selection)

Mapping Data to a New Space

- Fourier transform
- Wavelet transform







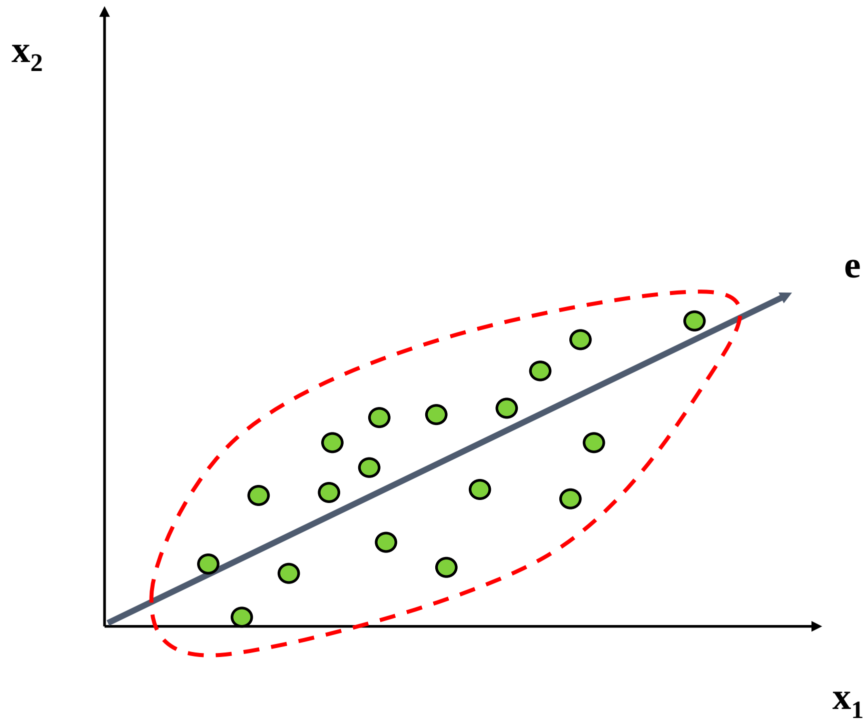
Two Sine Waves

Two Sine Waves + Noise

Frequency

Principal Component Analysis (PCA)

- Find a projection that captures the largest amount of variation in data
- The original data are projected onto a much smaller space, resulting in dimensionality reduction. We find the eigenvectors of the covariance matrix, and these eigenvectors define the new space



Principal Component Analysis (Steps)

- Given *N* data vectors from *n*-dimensions, find $k \le n$ orthogonal vectors (*principal components*) that can be best used to represent data
  - Normalize 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

Attribute Subset Selection

- 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., students' ID is often irrelevant to the task of predicting students' GPA

Heuristic Search in Attribute Selection

- 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
  - Best step-wise feature 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

Attribute Creation (Feature Generation)

- Create new attributes (features) that can capture the important information in a data set more effectively than the original ones
- Three general methodologies
  - Attribute extraction
    - Domain-specific
  - Mapping data to new space (see: data reduction)
    - E.g., Fourier transformation, wavelet transformation, manifold approaches (not covered)
  - Attribute construction
    - Combining features (see: discriminative frequent patterns in Chapter 7)
    - Data discretization

Summary

- **Data quality**: accuracy, completeness, consistency, timeliness, believability, interpretability
- **Data cleaning**: e.g. missing/noisy values, outliers
- **Data integration** from multiple sources:
  - Entity identification problem
  - Remove redundancies
  - Detect inconsistencies
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation