Data Mining

Concepts and Techniques

Module 2

Data Preprocessing

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation

Why Data Preprocessing?

- Data in the real world is dirty
- incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., occupation=" "
- Inaccurate or noisy: containing errors, or values that deviate from the expected(outliers)
    - e.g., Salary="-10"
- inconsistent: containing discrepancies in department codes or names used to categorize the items
    - e.g., Age ="5 years", Birthday ="06/06/1990", Current Year ="2017"
    - e.g., Was rating "1,2,3", now rating "A, B, C"
    - e.g., discrepancy between duplicate records

Why Is Data Dirty?

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

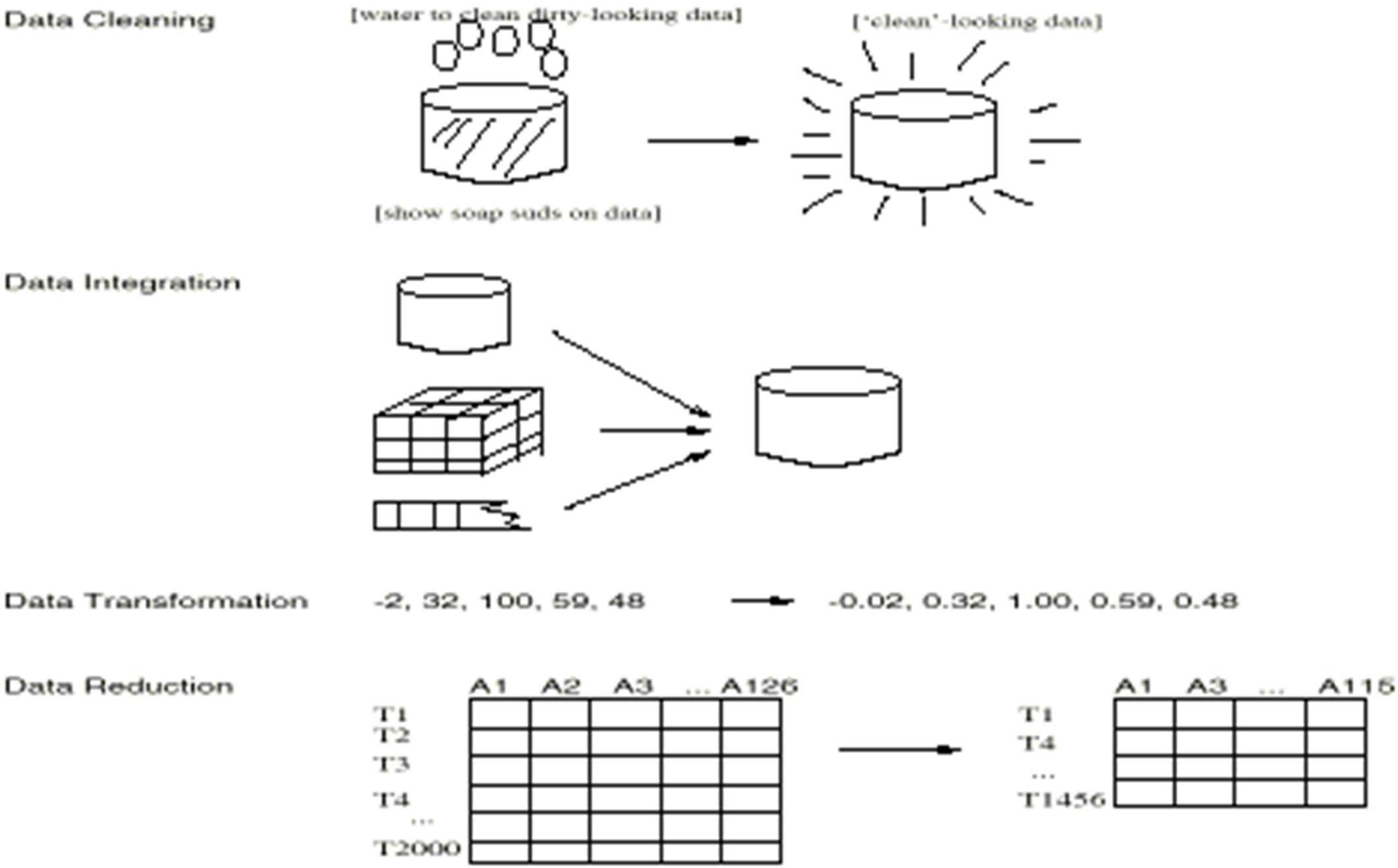
Why Is Data Preprocessing Important?

- 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

Major Tasks in Data Preprocessing

- Data cleaning
  - Fill in missing values, smooth out the noisy data, identify or remove outliers, and resolve inconsistencies
- Data integration
  - Integration of multiple databases, data cubes, or files
- Data transformation
  - Normalization and aggregation
- Data reduction
  - Obtains reduced representation in volume but produces the same or similar analytical results
- Data discretization
  - Part of data reduction but with particular importance, especially for numerical data

Forms of Data Preprocessing



Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Cleaning

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

Data Cleaning:Missing Data

- Data is not always available
  - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - 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
- Missing data may need to be inferred.

How to Handle Missing Data?

1. **Ignore the tuple:** usually done when class label is missing assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably.

1. **Fill in the missing value manually:** tedious + infeasible?

1. **Use a global constant to fill in the missing value**: Replace all missing attribute values by the same constant such as a label like *"Unknown"*

1. **Use a measure of central tendency for the attribute (e.g., the mean or median) to fill in the missing value**.

1. **Use the attribute mean or median for all samples belonging to the same class as the given tuple.**

1. **Use The most probable value: inference-based such as Bayesian formula or decision tree**

Data Cleaning:Noisy Data

- Noise: random error or variance in a measured variable
- Meaningless data that can not be interpreted by machines
- Noisy data (incorrect values) may come from
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission

How to Handle Noisy Data?

I. Binning

- Binning is a technique where we sort the data and then partition the data into equal frequency bins. Then you may either replace the noisy data with the bin mean bin median or the bin boundary.

- There are three methods for smoothing data in the bin.
  - **Smoothing by bin mean method:** In this method, the values in the bin are replaced by the mean value of the bin.
  - **Smoothing by bin median:** In this method, the values in the bin are replaced by the median value.
  - **Smoothing by bin boundary:** In this method, the minimum and maximum values of the bin values are taken, and the closest boundary value replaces the values.

How to Handle Noisy Data?

I. Binning Example

- 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

How to Handle Noisy Data?

II. Regression

- This is used to smooth the data and will help to handle data when unnecessary data is present.
- For the analysis, purpose regression helps to decide the variable which is suitable for our analysis.

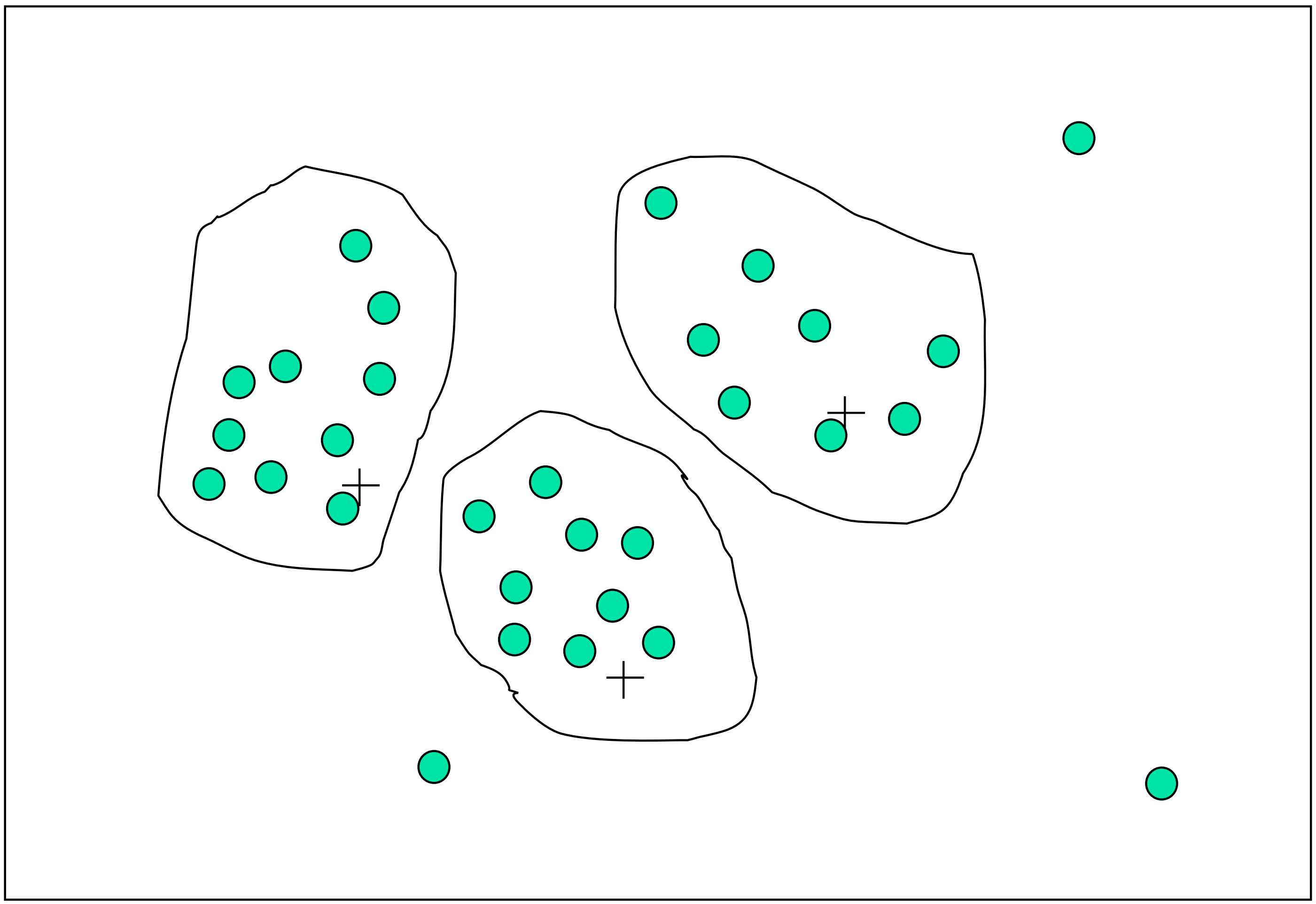
  - **Linear regression** refers to finding the best line to fit between two variables so that one can be used to predict the other.
  - **Multiple linear regression** involves more than two variables.

  - Using regression to find a mathematical equation to fit into the data helps to smooth out the noise.

How to Handle Noisy Data?

III. Outlier Analysis

- detect and remove outliers



III. Outlier Analysis

Outliers may be detected by clustering, where similar or close values are organized into the same groups or clusters. Thus, values that fall far apart from the cluster may be considered noise or outliers. Outliers are extreme values that deviate from other observations on data. They may indicate variability in measurement, experimental errors, or novelty. In other words, an outlier is an observation that diverges from an overall pattern on a sample. Outliers can be the following kinds, such as:

- ***Univariate outliers*** can be found when looking at a distribution of values in a single feature space.
- ***Multivariate outliers*** can be found in an n-dimensional space (of n-features). Looking at distributions in n-dimensional spaces can be very difficult for the human brain. That is why we need to train a model to do it for us.
- ***Point outliers*** are single data points that lay far from the rest of the distribution.
- ***Contextual outliers*** can be noise in data, such as punctuation symbols when realizing text analysis or background noise signal when doing speech recognition.
- ***Collective outliers*** can be subsets of novelties in data, such as a signal that may indicate the discovery of new phenomena.

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Integration

- Data integration:
  - Combines data from multiple sources into a coherent store to retain and provide a unified perspective of the data.
- Data Integration Challenges/Issues:**The semantic heterogeneity and structure of data.**

1.Schema integration:

Integrate metadata from different sources.

e.g., A.cust-id ≡ B.cust-#

Analyzing metadata statistics will prevent you from making errors during schema integration.

1. **Entity identification problem**:

   The problem of identifying object instances from different databases that correspond to the same real-world entity .

   e.g., Bill Clinton = William Clinton.

1. **Structural Integration:**. Ensure that any attribute functional dependencies and referential constraints in the source system match those in the target system.
2. **Redundancy and Correlation Analysis**
3. **Tuple Duplication**

Data Integration

- Data Integration Challenges/Issues:
  - **The semantic heterogeneity and structure of data** pose great challenges in data integration.

1.Schema integration:

Integrate metadata from different sources.

1. **Structural Integration: When matching attributes from one database to another during integration, special attention must be paid to the *structure* of the data i.e.** ensure that any attribute functional dependencies and referential constraints in the source system match those in the target system.

   For example, you were given client data from specialized statistics sites. Customer identity is assigned to an entity from one statistics supply, while a customer range is assigned to an entity from another statistics supply. Analyzing such metadata statistics will prevent you from making errors during schema integration.

1. **Redundancy and Correlation Analysis**
2. **Tuple Duplication**

Data Integration

- When matching attributes from one database to another during integration, special attention must be paid to the *structure* of the data i.e. ensure that any attribute functional dependencies and referential constraints in the source system match those in the target system.

  For example, assume that the discount is applied to the entire order in one machine, but in every other machine, the discount is applied to each item in the order. This distinction should be noted before the information from those assets is included in the goal system.

Handling Redundancy in Data Integration

- Redundant data occur often while integrating multiple databases.

- Unimportant data that are no longer required are referred to as redundant data.
  - *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,age
  - Inconsistencies in attribute or dimension naming can also cause redundancies in the resulting data set.

- Some redundancies can be detected by **correlation analysis**.

Suppose we have a data set that has three attributes - pizza_name, is_veg, is_nonveg

1. Is_veg is 1; if the selecting pizza is veg else, it is 0.
2. Is_nonveg is 1; if the selecting pizza is nonveg else, it is 0.

   On analyzing the above table, we have found that if a pizza is not veg (i.e., is_veg is 0 selecting the pizza_name), the pizza is surely non-veg (Since there are only two values in the pizza_name output class- Veg and Nonveg). Hence, one of these attributes became redundant. It means that the two attributes are very much related to each other, and one attribute can find the other. So, you can drop either the first or second attribute without any loss of information.

   Age and DoB attribute.

   Age can be derived from DoB.

Correlation Analysis

- Given two attributes , correlation analysis can measure how strongly one attribute implies the other, based on the available data.

- For nominal data, we use the x^2 (*chi-square*) test.

- For numeric attributes, we can use the *correlation coefficient* and *covariance*, both of which assess how one attribute values vary from those of another.

x^2 Correlation Test for Nominal Data

- Suppose attribute *A* has *c* distinct values, namely $a_1, a_2, \ldots . a_c$.
- *Attribute B* has *r* distinct values, namely $b_1, b_2, \ldots . b_r$.
- Let (A_i, Bj_j) denote the joint event that attribute *A* takes on value a_i and attribute *B* takes on value b_j
- The x^2 value (also known as the *Pearson* x^2 *statistic*) is computed as:

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}},$$

where o_{ij} is the *observed frequency* (i.e., actual count) of the joint event (A_i, B_j) and e_{ij} is the *expected frequency* of (A_i, B_j), which can be computed as

$$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n},$$

where n is the number of data tuples, $count(A = a_i)$ is the number of tuples having value a_i for A, and $count(B = b_j)$ is the number of tuples having value b_j for B.

x^2 Correlation Test for Nominal Data

$$\chi^2 = \sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(o_{ij}-e_{ij})^2}{e_{ij}}, \qquad e_{ij} = \frac{count(A=a_i)\times count(B=b_j)}{n},$$

- The x^2 statistic tests the hypothesis that *A* and *B* are *independent*, that is, there is no correlation between them.

- The test is based on a significance level, with (*r*-1)x(*c-1)* degrees of freedom.

- If the sample statistic x^2 > tabulated statistics $x^2_{\alpha,dof}$, the null hypothesis that A and B are independent is rejected, So, we say that *A* and *B* are statistically correlated.

- **DOF: No of values in the final calculation of the statistic that are free to vary.**
- **Level of significance: Prob of rejecting the null hypothesis when it is true.**

Example1: Correlation Analysis of Nominal attributes using x^2

2 x 2 Contingency Table Data

	male	female	Total
fiction	250	200	450
non_fiction	50	1000	1050
Total	300	1200	1500

Note: Are *gender* and *preferred_reading* correlated?

$$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n},$$

$$e_{11} = \frac{count(male) \times count(fiction)}{n} = \frac{300 \times 450}{1500} = 90,$$

2 x 2 Contingency Table Data

	male	female	Total
fiction	250 (90)	200 (360)	450
non_fiction	50 (210)	1000 (840)	1050
Total	300	1200	1500

Note: Are *gender* and *preferred_reading* correlated?

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}},$$

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840}$$
$$= 284.44 + 121.90 + 71.11 + 30.48 = 507.93.$$

- For this 2x2 table, the degrees of freedom are (2-1)(2-1)= 1.
- For 1 degree of freedom , the x^2 value needed to reject the hypothesis at the 0.001 significance level is **10.828** (From Chi square distribution table)
- Since our computed value is greater than tabulated value, we can reject the hypothesis that *gender* and *preferred reading* are independent and conclude that the two attributes are (strongly) correlated for the given group of people.

Chi-Square distribution table

	P (<)										
DF	0.995	0.975	0.2	0.1	0.05	0.025	0.02	0.01	0.005	0.002	0.001
1	.0004	.00016	1.642	2.706	3.841	5.024	5.412	6.635	7.879	9.55	10.828
2	0.01	0.0506	3.219	4.605	5.991	7.378	7.824	9.21	10.597	12.429	13.816
3	0.0717	0.216	4.642	6.251	7.815	9.348	9.837	11.345	12.838	14.796	16.266
4	0.207	0.484	5.989	7.779	9.488	11.143	11.668	13.277	14.86	16.924	18.467
5	0.412	0.831	7.289	9.236	11.07	12.833	13.388	15.086	16.75	18.907	20.515
6	0.676	1.237	8.558	10.645	12.592	14.449	15.033	16.812	18.548	20.791	22.458
7	0.989	1.69	9.803	12.017	14.067	16.013	16.622	18.475	20.278	22.601	24.322
8	1.344	2.18	11.03	13.362	15.507	17.535	18.168	20.09	21.955	24.352	26.124
9	1.735	2.7	12.242	14.684	16.919	19.023	19.679	21.666	23.589	26.056	27.877
10	2.156	3.247	13.442	15.987	18.307	20.483	21.161	23.209	25.188	27.722	29.588
11	2.603	3.816	14.631	17.275	19.675	21.92	22.618	24.725	26.757	29.354	31.264
12	3.074	4.404	15.812	18.549	21.026	23.337	24.054	26.217	28.3	30.957	32.909
13	3.565	5.009	16.985	19.812	22.362	24.736	25.472	27.688	29.819	32.535	34.528
14	4.075	5.629	18.151	21.064	23.685	26.119	26.873	29.141	31.319	34.091	36.123
15	4.601	6.262	19.311	22.307	24.996	27.488	28.259	30.578	32.801	35.628	37.697
16	5.142	6.908	20.465	23.542	26.296	28.845	29.633	32	34.267	37.146	39.252
17	5.697	7.564	21.615	24.769	27.587	30.191	30.995	33.409	35.718	38.648	40.79
18	6.265	8.231	22.76	25.989	28.869	31.526	32.346	34.805	37.156	40.136	42.312
19	6.844	8.907	23.9	27.204	30.144	32.852	33.687	36.191	38.582	41.61	43.82
20	7.434	9.591	25.038	28.412	31.41	34.17	35.02	37.566	39.997	43.072	45.315

Example 2

A food services manager for a baseball park wants to know if there is a relationship between gender (male or female) and the preferred condiment on a hot dog. The following table summarizes the results. Test the hypothesis with a significance level of 10%.

		Condiment			
		Ketchup	Mustard	Relish	Total
Gender	Male	15	23	10	48
	Female	25	19	8	52
	Total	40	42	18	100

Step 1: The hypotheses are:

H0: Gender and condiments are independent

- H1 : Gender and condiments are not independent

Example 2

A food services manager for a baseball park wants to know if there is a relationship between gender (male or female) and the preferred condiment on a hot dog. The following table summarizes the results. Test the hypothesis with a significance level of 10%.

		Condiment			
		Ketchup	Mustard	Relish	Total
Gender	Male	15	23	10	48
	Female	25	19	8	52
	Total	40	42	18	100

Step 2: Find expected frequencies table

$$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n},$$

		Condiment			
		Ketchup	Mustard	Relish	Total
Gender	Male	15 (19.2)	23 (20.16)	10 (8.64)	48
	Female	25 (20.8)	19 (21.84)	8 (9.36)	52
	Total	40	42	18	100

Example 2

		Condiment			
		Ketchup	Mustard	Relish	Total
Gender	Male	15 (19.2)	23 (20.16)	10 (8.64)	48
	Female	25 (20.8)	19 (21.84)	8 (9.36)	52
	Total	40	42	18	100

Step 3: Calculate Chi square statistic

$$\chi^2 = \sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(o_{ij}-e_{ij})^2}{e_{ij}},$$

$$\boxed{\chi^{2*} = \frac{(15-19.2)^2}{19.2} + \frac{(23-20.16)^2}{20.16} + \ldots + \frac{(8-9.36)^2}{9.36} = 2.95}$$

Step 4: DoF: (c-1)(r-1) =3 x 1 =3

Example 2

Step 3: Calculate Chi square statistic

$$\chi^2 = \sum_{i=1}^{c}\sum_{j=1}^{r}\frac{(o_{ij}-e_{ij})^2}{e_{ij}},$$

$$\chi^{2*} = \frac{(15-19.2)^2}{19.2} + \frac{(23-20.16)^2}{20.16} + \ldots + \frac{(8-9.36)^2}{9.36} = 2.95$$

Step 4: DoF: (c-1)(r-1) =3 x 1 =3

Step 5: Compare calculated test statistic with tabulated one.
Her Tabulated statistic(with *significance* level =0.01) = 11.345 > 2.95 .
Hence H0 is accepted. i.e attributes are independent.

Correlation Analysis - Hypotheses

- H_0 : Null Hypothesis - attributes are not related (ie Independent)
- H_1 : Alternate Hypothesis - attributes are related (ie dependent)

- Degree of freedom = (#rows-1 *#columns -1)
- Given df and chi :
  - Find the p-value in Chi squared distribution table depending on the prob and chi value
  - The p-value *100 gives the % of likelihood
- Given the df and p-value
  - Get the chi value from the table
  - Reject the null hypothesis if the value of chi is higher than the chi value obtained form tale
  - Else accept the null hypothesis if the obtained value of chi square is less than the tabular value.

	Right-Tail Probability				
df	0.250	0.100	0.050	0.025	0.010
1	1.32	2.71	3.84	5.02	6.63
2	2.77	4.61	5.99	7.38	9.21
3	4.11	6.25	7.81	9.35	11.34
4	5.39	7.78	9.49	11.14	13.28
5	6.63	9.24	11.07	12.83	15.09

Chi-Square Calculation: Example 1

	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

- H_0: Science fiction is not associated with playing chess, and
- H_1: Science fiction is associated with playing chess

- df = (2-1)*(2-1) = 1
- X^2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

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

- P-value comes out to be 0 i.e there is 0 % likelihood of null hypothesis to be true.
- It shows that like_science_fiction and play_chess are correlated in the group

Chi-Square Calculation: Example 3

A group of students were classified in terms of personality (introvert or extrovert) and in terms of color preference (red, yellow, green or blue) with the purpose of seeing whether there is an association (relationship) between personality and color preference. Data was collected from 400 students and presented in the 2 (rows) x 4 (cols) contingency table below:

(Observed counts)	Colors				
	Red	**Yellow**	**Green**	**Blue**	**Totals**
Introvert personality	20	6	30	44	100
Extrovert personality	180	34	50	36	300
Totals	200	40	80	80	400

Suitable null and alternative hypotheses might be:

•H_0: Color preference is not associated with personality, and X2 = 71.20

•H_1: Color preference is associated with personality

To perform a chi-squared test, the number of students expected in each cell of the table if the null hypothesis is true, is calculated.

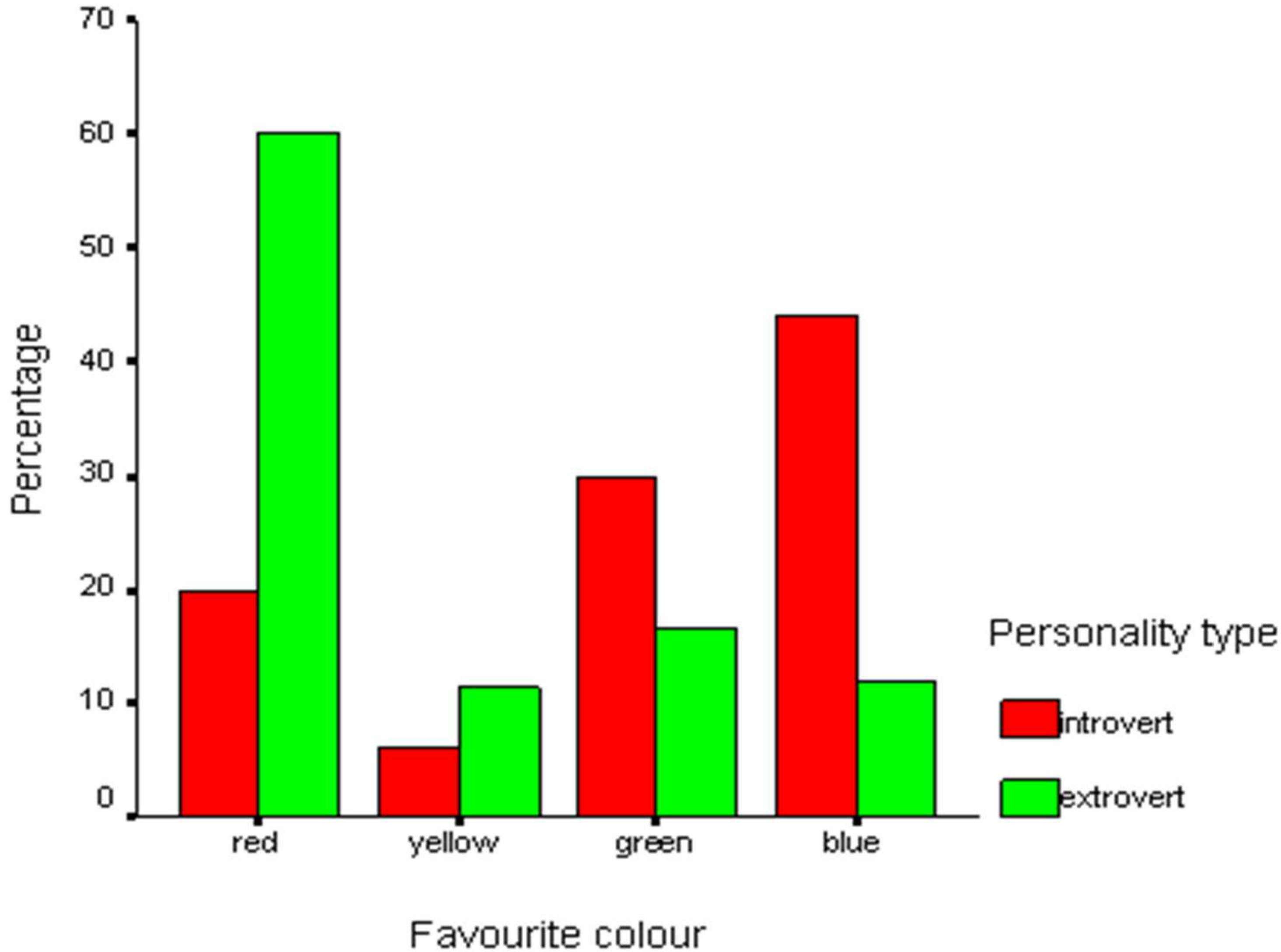
Chi-Square Calculation: Example2

A group of students were classified in terms of personality (introvert or extrovert) and in terms of color preference (red, yellow, green or blue) with the purpose of seeing whether there is an association (relationship) between personality and color preference. Data was collected from 400 students and presented in the 2 (rows) x 4 (cols) contingency table below. Check whether color and personality attributes are correlated.

(Observed counts)	**Colors**				
	Red	**Yellow**	**Green**	**Blue**	**Totals**
Introvert personality	20	6	30	44	100
Extrovert personality	180	34	50	36	300
Totals	200	40	80	80	400

X2 = 71.20

Chi-Square Calculation: Example 2



Chi-Square Calculation: Example 3

- Children of three ages are asked to indicate their preference for three photographs of adults. Do the data suggest that there is a significant relationship between age and photograph preference?

Age of child		**photograph**		
		A	B	C
	5-6 years	18	22	20
	7-8 years	2	28	40
	9-10 years	20	10	40

Correlation Analysis (Numerical Data)

- For numeric attributes, we can evaluate the correlation between two attributes, *A* and *B*,by computing the **correlation coefficient** (also known as **Pearson's product moment coefficient**

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

where n is the number of tuples, a_i and b_i are the respective values of A and B in tuple i, $\bar{A}$ and $\bar{B}$ are the respective mean values of A and B, σ_A and σ_B are the respective standard deviations of A and B and $\Sigma(a_i b_i)$ is the sum of the AB

Note that $-1 \leq r_{A,B} \leq +1$.

- If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher the value, the stronger correlation. Hence, a higher value may indicate that *A* (or *B*) may be removed as a redundancy.
- $r_{A,B} = 0$: independent(no correlation);
- $r_{A,B} < 0$: negatively correlated (A's values increase as B's decrease). This means that each attribute discourages the other

Covariance for Numerical Data

- In probability theory and statistics, correlation and covariance are two similar measures for assessing how much two attributes change together.
- Consider two numeric attributes *A* and *B*, and a set of *n* observations {(*a*1,*b*1),(a2,b2)…, (*an*,*bn)}*
- The mean values of *A* and *B*, respectively, are also known as the **expected values** on *A* and *B*, that is, $E(A) = \bar{A} = \frac{\sum_{i=1}^{n} a_i}{n}$ and $E(B) = \bar{B} = \frac{\sum_{i=1}^{n} b_i}{n}.$

The **covariance** between A and B is defined as

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

- Also it can be shown that

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

Covariance for Numerical Data

- For two attributes *A* and *B* that tend to change together, if *A* is larger than $\bar{A}$ (the expected value of *A*), then *B* is likely to be larger than $\bar{B}$ (the expected value of *B*). Therefore, the covariance between *A* and *B* is *positive*.

- On the other hand, if one of the attributes tends to be above its expected value when the other attribute is below its expected value, then the covariance of *A* and *B* is *negative*.

- If *A* and *B* are *independent* (i.e., they do not have correlation), then *E*(*A.B)*= *E*(*A*).*E(B)* Therefore, the covariance is $Cov(A,B) = E(A \cdot B) - \bar{A}\bar{B} = E(A) \cdot E(B) - \bar{A}\bar{B} = 0.$

- However, the converse is not true. Some pairs of random variables (attributes) may have a covariance of 0 but are not independent

Example: Covariance for Numerical Data

- Stock Prices for *AllElectronics* and *HighTech*

Time point	*AllElectronics*	*HighTech*
t1	6	20
t2	5	10
t3	4	14
t4	3	5
t5	2	5

if the stocks are affected by the same industry trends will their prices rise or fall together?

$$E(AllElectronics) = \frac{6+5+4+3+2}{5} = \frac{20}{5} = \$4$$

and

$$E(HighTech) = \frac{20+10+14+5+5}{5} = \frac{54}{5} = \$10.80.$$

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

$$Cov(AllElectroncis, HighTech) = \frac{6 \times 20 + 5 \times 10 + 4 \times 14 + 3 \times 5 + 2 \times 5}{5} - 4 \times 10.80$$

$$= 50.2 - 43.2 = 7.$$

- Therefore, given the positive covariance we can say that stock prices for both companies rise together.

Tuple Duplication

- In addition to detecting redundancies between attributes, duplication should also be detected at the tuple level (e.g., where there are two or more identical tuples for a given unique data entry case).

Data Value conflict detection and Resolution

- For the same real-world entity, attribute values from different sources may differ.
- This may be due to differences in representation, scaling, or encoding.

- Attributes may also differ on the abstraction level, where an attribute in one system is recorded at, say, a lower abstraction level than the “same” attribute in another.

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Reduction Strategies

https://www.javatpoint.com/data-reduction-in-data-mining

Data Reduction Strategies

- Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results
- Sometimes, it is also performed to find the most suitable subset of attributes from a large number of attributes. This is known as dimensionality reduction.
- Data reduction also involves reducing the number of attribute values and/or the number of tuples.

Data Reduction Techniques

1. **Data cube aggregation:** In this technique the data is reduced by applying OLAP operations like slice, dice or rollup. It uses the smallest level necessary to solve the problem.
2. **Dimensionality reduction:** The data attributes or dimensions are reduced. Not all attributes are required for data mining. The most suitable subset of attributes are selected by using techniques like forward selection, backward elimination, decision tree induction or a combination of forward selection and backward elimination.
3. **Data compression:** In this technique. large volumes of data is compressed i.e. the number of bits used to store data is reduced. This can be done by using lossy or lossless compression. In *loss compression,* the quality of data is compromised for more compression. In *lossless compression,* the quality of data is not compromised for higher compression level.
4. **Numerosity reduction :** This technique reduces the volume of data by choosing smaller forms for data representation. Numerosity reduction can be done using histograms, clustering or sampling of data. Numerosity reduction is necessary as processing the entire data set is expensive and time consuming.

Data Reduction Techniques

1.Dimensionality reduction:

- It eliminates outdated or redundant features.
- 3 methods are :
- **Wavelet Transform:**The **discrete wavelet transform (DWT)** is a linear signal processing technique that, when applied to a data vector ***X***, transforms it to a numerically different vector, ***X'*** of **wavelet coefficients**.
  The compressed data is obtained by retaining the smallest fragment of the strongest wavelet coefficients. Wavelet transform can be applied to data cubes, sparse data, or skewed data.

- **Principal Component Analysis:** Suppose we have a data set to be analyzed that has tuples with n attributes. The principal component analysis identifies k independent tuples with n attributes that can represent the data set.
  In this way, the original data can be cast on a much smaller space, and dimensionality reduction can be achieved. Principal component analysis can be applied to sparse and skewed data.

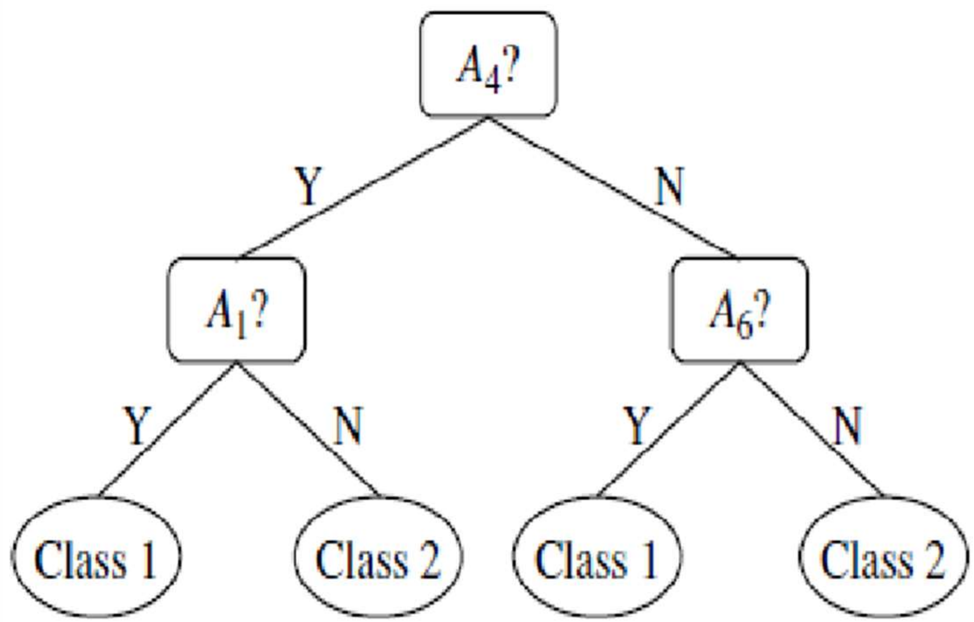
- **Attribute Subset Selection:**

Data Reduction Techniques

Dimensionality reduction:

- **Attribute Subset Selection:** The attribute subset selection reduces the volume of data by eliminating redundant and irrelevant attributes.
  The most suitable subset of attributes are selected by using techniques like forward selection, backward elimination, decision tree induction or a combination of forward selection and backward elimination.
  The attribute subset selection ensures that we get a good subset of original attributes even after eliminating the unwanted attributes. The resulting probability of data distribution is as close as possible to the original data distribution using all the attributes.

Greedy(heuristic) methods for attribute subset selection

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ => $\{A_1\}$ => $\{A_1, A_4\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_4, A_5, A_6\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  => Reduced attribute set: $\{A_1, A_4, A_6\}$

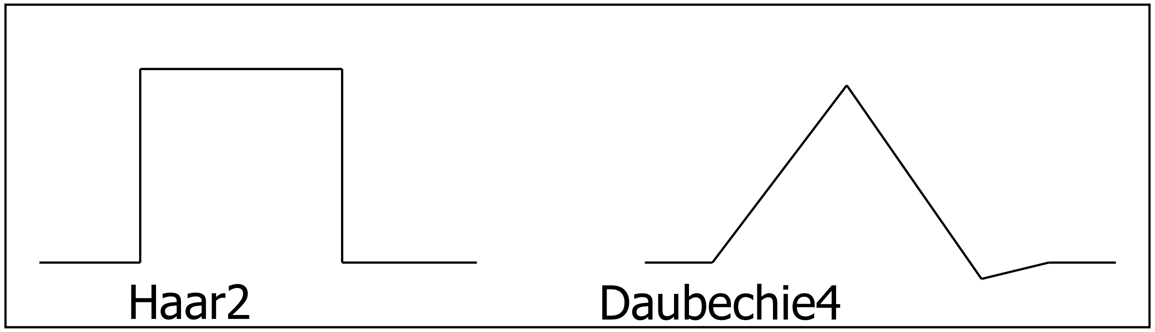
1. Stepwise forward selection: The procedure starts with an empty set of attributes as the reduced set. The best of the original attributes is determined and added to the reduced set. At each subsequent iteration or step, the best of the remaining original attributes is added to the set.
2. Stepwise backward elimination: The procedure starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.
3. Combination of forward selection and backward elimination: The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.
4. Decision tree induction: Decision tree algorithms (e.g., ID3, C4.5, and CART) were originally intended for classification. Decision tree induction constructs a flowchart like structure where each internal (nonleaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the "best" attribute to partition the data into individual classes.
When decision tree induction is used for attribute subset selection, a tree is constructed from the given data. All attributes that do not appear in the tree are assumed to be irrelevant. The set of attributes appearing in the tree form the reduced subset of attributes.
The stopping criteria for the methods may vary. The procedure may employ a threshold on the measure used to determine when to stop the attribute selection process.

Data Reduction Strategies

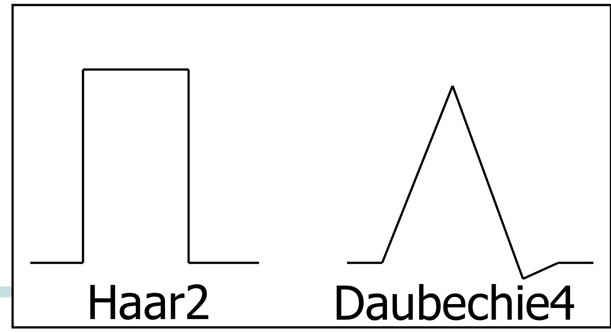
- Why data reduction?
  - A database/data warehouse may store terabytes of data
  - Complex data analysis/mining may take a very long time to run on the complete data set
- Data reduction
  - Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results
- Data reduction strategies
  - Dimensionality reduction — e.g., remove unimportant attributes
    - Wavelet Transform, PCA, Attribute subset selection
  - Numerosity reduction — e.g., fit data into models
    - Parametric methods: Regression, log-linear models
    - Non parametric methods: histograms, clustering, sampling and data cube aggregation
  - Data Compression

Dimensionality Reduction: Wavelet Transformation

- The **discrete wavelet transform (DWT)** is a linear signal processing technique that, when applied to a data vector ***X***, transforms it to a numerically different vector, ***X'*** of **wavelet coefficients**.
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients.
- Similar to discrete Fourier transform (DFT), but better lossy compression.
- Popular wavelet transforms include the Haar-2, Daubechies-4,and Daubechies-6.



Dimensionality Reduction: Wavelet Transformation



- Method:

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 ($L >= n$).

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.

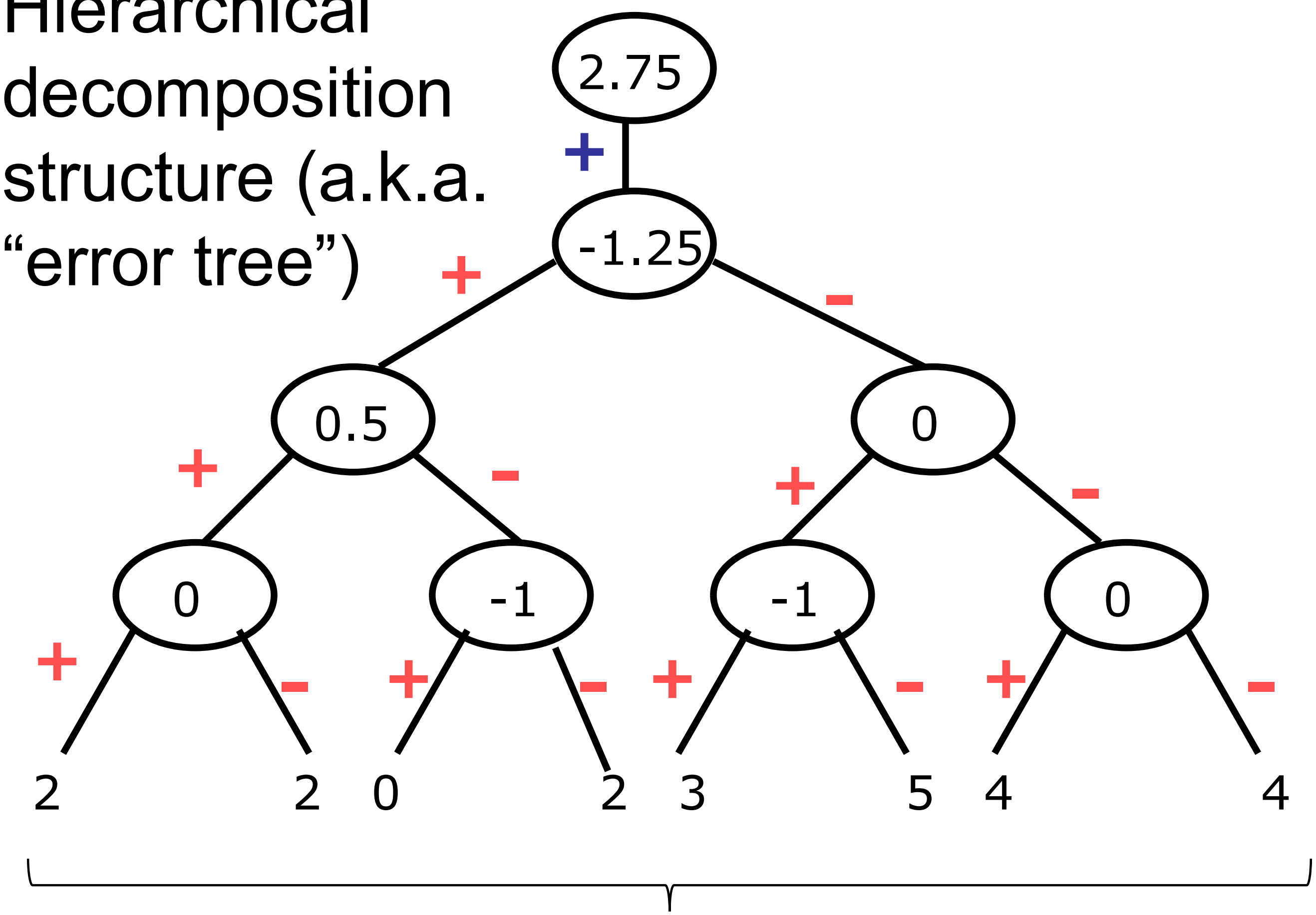
Wavelet Decomposition

- Wavelets: A math tool for space-efficient hierarchical decomposition of functions
- S = [2, 2, 0, 2, 3, 5, 4, 4] can be transformed to $S_\wedge$ = [$2^3/_4$, $-1^1/_4$, $^1/_2$, 0, 0, -1, -1, 0]
- Compression: many small detail coefficients can be replaced by 0's, and only the significant coefficients are retained

Resolution	Averages	Detail Coefficients
8	$[2, 2, 0, 2, 3, 5, 4, 4]$	
4	$[2, 1, 4, 4]$	$[0, -1, -1, 0]$
2	$[1\frac{1}{2}, 4]$	$[\frac{1}{2}, 0]$
1	$[2\frac{3}{4}]$	$[-1\frac{1}{4}]$

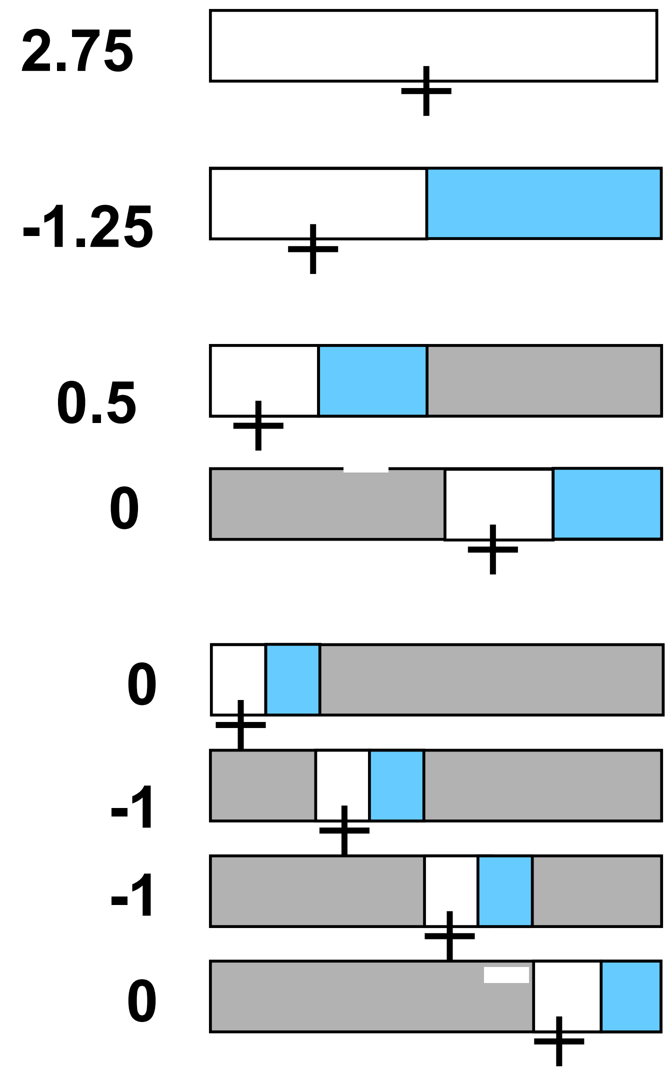
Haar Wavelet Coefficients

Hierarchical decomposition structure (a.k.a. “error tree”)



Original frequency distribution

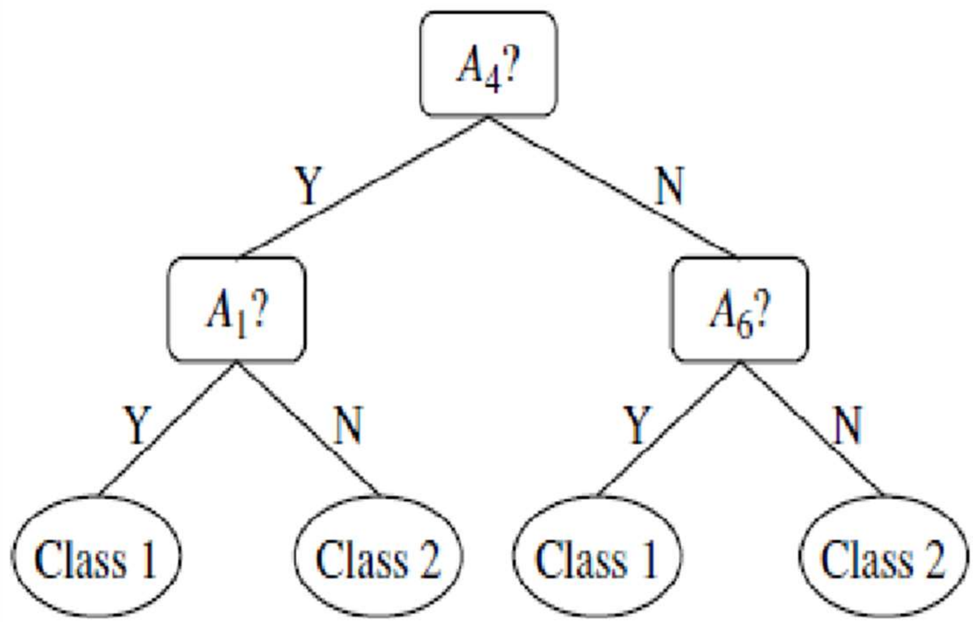
Coefficient “Supports”



Attribute Subset Selection

- **Attribute subset selection** reduces the data set size by removing irrelevant or redundant attributes (or dimensions).
- The goal of attribute subset selection is to find a minimum set of attributes such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all attributes.
- Mining on a reduced set of attributes has an additional benefit: It reduces the number of attributes appearing in the discovered patterns, helping to make the patterns easier to understand.
- Basic heuristic methods of attribute subset selection include
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

Greedy(heuristic) methods for attribute subset selection

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ => $\{A_1\}$ => $\{A_1, A_4\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_4, A_5, A_6\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  => Reduced attribute set: $\{A_1, A_4, A_6\}$

1. Stepwise forward selection: The procedure starts with an empty set of attributes as the reduced set. The best of the original attributes is determined and added to the reduced set. At each subsequent iteration or step, the best of the remaining original attributes is added to the set.
2. Stepwise backward elimination: The procedure starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.
3. Combination of forward selection and backward elimination: The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.
4. Decision tree induction: Decision tree algorithms (e.g., ID3, C4.5, and CART) were originally intended for classification. Decision tree induction constructs a flowchartlike structure where each internal (nonleaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the "best" attribute to partition the data into individual classes.
When decision tree induction is used for attribute subset selection, a tree is constructed fromthe given data. All attributes that do not appear in the tree are assumed to be irrelevant. The set of attributes appearing in the tree form the reduced subset of attributes.
The stopping criteria for the methods may vary. The procedure may employ a threshold on the measure used to determine when to stop the attribute selection process.

Numerosity Reduction

- Reduce data volume by choosing alternative, smaller forms of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Example: Log-linear models—obtain value at a point in m-D space as the product on appropriate marginal subspaces
- Non-parametric methods
  - Do not assume models
  - Major families: histograms, clustering, sampling

Regression Analysis and Log-Linear Models: Parametric Data Reduction

- Linear regression: Data are modeled to fit a straight line.
- $Y = w X + b$
  - 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 $Y1, Y2, \ldots, X1, X2, \ldots$.

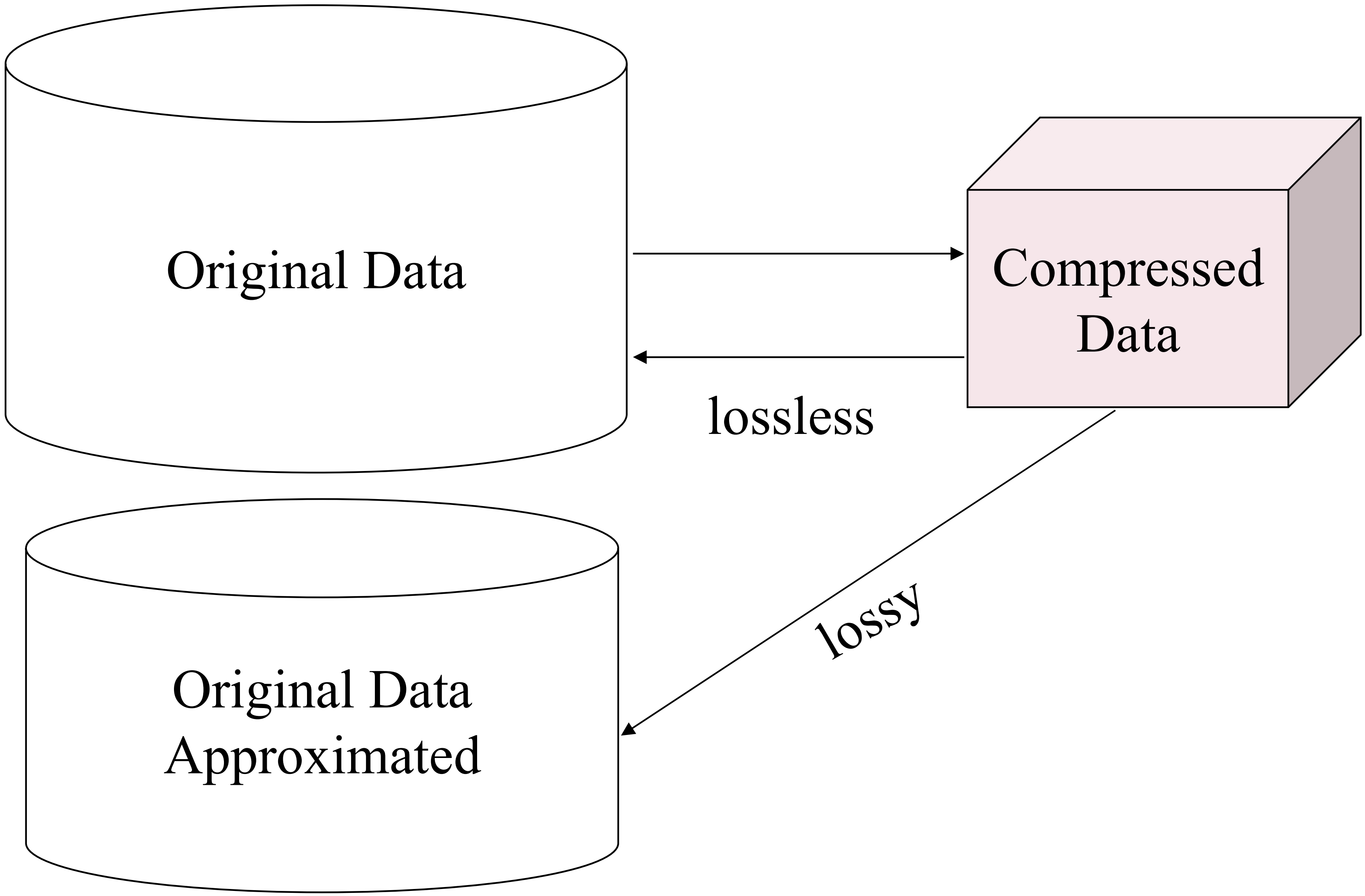
- Multiple regression: allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- $Y = b0 + b1\ X1 + b2\ X2.$
  - Many nonlinear functions can be transformed into the above

- Log-linear models: approximates discrete multidimensional probability distributions
  - The multi-way table of joint probabilities is approximated by a product of lower-order tables

Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless
  - But only limited manipulation is possible without expansion
- Audio/video compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence is not audio
  - Typically short and vary slowly with time

Data Compression



Data Transformation

- Data transformation is a technique used to **convert the raw data into a suitable format** that efficiently eases data mining and retrieves strategic information.

- Data transformation includes data cleaning techniques and a data reduction technique to convert the data into the appropriate form.

- Data transformation is an essential data preprocessing technique that must be performed on the data before data mining to provide patterns that are easier to understand.

- Data transformation changes the format, structure, or values of the data and converts them into clean, usable data.

Data Transformation Techniques

1. Data Smoothing
2. Data Aggregation
3. Data Generalization: concept hierarchy climbing
4. Data Normalization: scaled to fall within a small, specified range
   - min-max normalization
   - z-score normalization
   - normalization by decimal scaling
5. Attribute/feature construction
   - New attributes constructed from the given ones
6. Discretization

1. Data Smoothing

- Data smoothing is a process that is used to remove noise from the dataset using some algorithms.
- It allows for highlighting important features present in the dataset. It helps in predicting the patterns.
- When collecting data, it can be manipulated to eliminate or reduce any variance or any other noise form.
- The concept behind data smoothing is that it will be able to identify simple changes to help predict different trends and patterns. This serves as a help to analysts or traders who need to look at a lot of data which can often be difficult to digest for finding patterns that they wouldn't see otherwise.

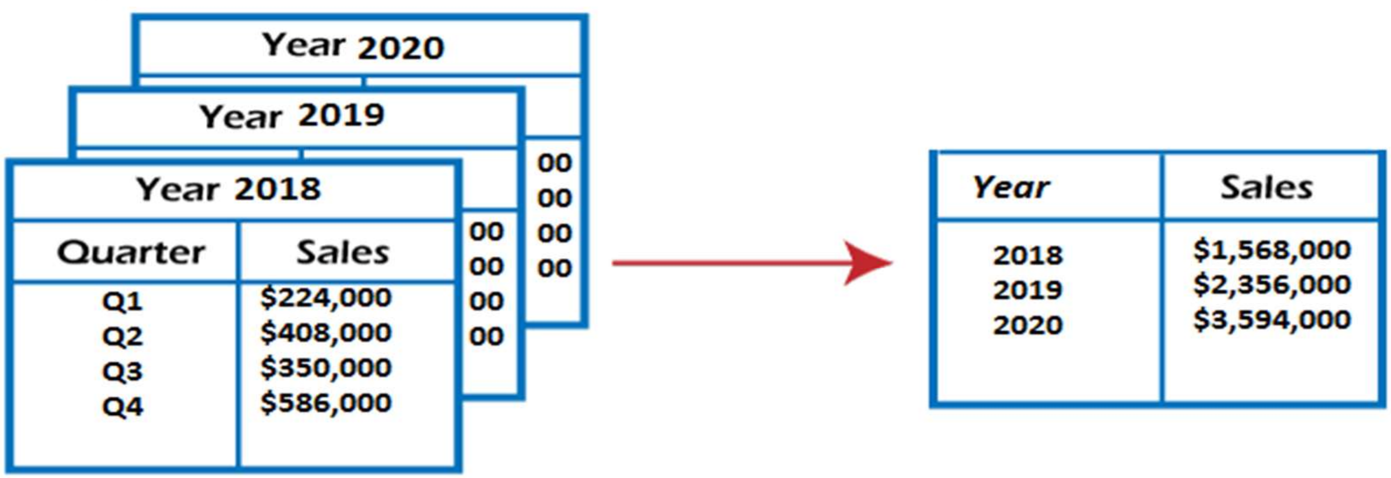
The noise is removed from the data using the techniques such as binning, regression, clustering.

- **Binning:** This method splits the sorted data into the number of bins and smoothens the data values in each bin considering the neighborhood values around it.
- **Regression:** This method identifies the relation among two dependent attributes so that if we have one attribute, it can be used to predict the other attribute.
- **Clustering:** This method groups similar data values and form a cluster. The values

2. Data Aggregation

- Data collection or aggregation is the method of storing and presenting data in a summary format.
- This is a crucial step since the accuracy of data analysis insights is highly dependent on the quantity and quality of the data used.
- Gathering accurate data of high quality and a large enough quantity is necessary to produce relevant results.

For example, we have a data set of sales reports of an enterprise that has quarterly sales of each year. We can aggregate the data to get the enterprise's annual sales report.



Aggregated Data

3. Data Generalization

- It converts low-level data attributes to high-level data attributes using concept hierarchy. This conversion from a lower level to a higher conceptual level is useful to get a clearer picture of the data. Data generalization can be divided into two approaches:
  - Data cube process (OLAP) approach.
  - Attribute-oriented induction (AOI) approach.

- For example, age data can be in the form of (20, 30) in a dataset. It is transformed into a higher conceptual level into a categorical value (young, old).

3. Data Generalization:concept hierarchy climbing

- It converts low-level data attributes to high-level data attributes using concept hierarchy.
- This conversion from a lower level to a higher conceptual level is useful to get a clearer picture of the data.

- Data generalization can be divided into two approaches:
  - Data cube process (OLAP) approach.
  - Attribute-oriented induction (AOI) approach.

For example, age data can be in the form of (20, 30) in a dataset. It is transformed into a higher conceptual level into a categorical value (young, old).

4. Normalization

(data scaled to fall within a small, specified range)

- Min-max normalization:

Min-max normalization performs a linear transformation on the original data. Suppose that min_A and max_A are the minimum and maximum values of an attribute, A. Min-max normalization maps a value, v_i, of A to v_i' in the range $[new\_min_A, new\_max_A]$ by computing

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

Min-max normalization preserves the relationships among the original data values. It will encounter an "out-of-bounds" error if a future input case for normalization falls outside of the original data range for A.

Example **Min-max normalization.** Suppose that the minimum and maximum values for the attribute *income* are \$12,000 and \$98,000, respectively. We would like to map *income* to the range [0.0, 1.0]. By min-max normalization, a value of \$73,600 for *income* is transformed to $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$.

Data Transformation by Normalization

- Z-score normalization

In **z-score normalization** (or *zero-mean normalization*), the values for an attribute, A, are normalized based on the mean (i.e., average) and standard deviation of A. A value, v_i, of A is normalized to v_i' by computing

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A},$$

where $\bar{A}$ and σ_A are the mean and standard deviation, respectively, of attribute A.

$\bar{A} = \frac{1}{n}(v_1 + v_2 + \cdots + v_n)$ and σ_A is computed as the square root of the variance of A

- This method of normalization is useful when the actual minimum and maximum of attribute *A* are unknown, or when there are outliers that dominate the min-max normalization.

Example **z-score normalization.** Suppose that the mean and standard deviation of the values for the attribute *income* are \$54,000 and \$16,000, respectively. With z-score normalization, a value of \$73,600 for *income* is transformed to $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$.

Data Transformation by Normalization

- Decimal scaling normalization:

Normalization by decimal scaling normalizes by moving the decimal point of values of attribute A. The number of decimal points moved depends on the maximum absolute value of A. A value, v_i, of A is normalized to v_i' by computing

$$v_i' = \frac{v_i}{10^j},$$

where j is the smallest integer such that $max(|v_i'|) < 1$.

Example **Decimal scaling.** Suppose that the recorded values of A range from -986 to 917. The maximum absolute value of A is 986. To normalize by decimal scaling, we therefore divide each value by 1000 (i.e., $j = 3$) so that -986 normalizes to -0.986 and 917 normalizes to 0.917.

5. Attribute Construction

- In the attribute construction method, the new attributes consult the existing attributes to construct a new data set that eases data mining.
- New attributes are created and applied to assist the mining process from the given attributes. This simplifies the original data and makes the mining more efficient.
- For example, suppose we have a data set referring to measurements of different plots, i.e., we may have the height and width of each plot. So here, we can construct a new attribute 'area' from attributes 'height' and 'weight'.
- Attribute construction also helps understand the relations among the attributes in a data set.

6. Data Discretization

- This is a process of converting continuous data into a set of data intervals. Continuous attribute values are substituted by small interval labels.
- This makes the data easier to study and analyze.
- If a data mining task handles a continuous attribute, then its discrete values can be replaced by constant quality attributes. This improves the efficiency of the task.
- This method is also called a data reduction mechanism as it transforms a large dataset into a set of categorical data.
- Discretization also uses decision tree-based algorithms to produce short, compact, and accurate results when using discrete values.

For example, the values for the age attribute can be replaced by the interval labels such as (0-10, 11-20...) or (kid, youth, adult, senior).

Discretization

- Three types of attributes:
  - Nominal — values from an unordered set, e.g., color, profession
  - Ordinal — values from an ordered set, e.g., military or academic rank
  - Continuous — real numbers, e.g., integer or real numbers
- Discretization:
  - Divide the range of a continuous attribute into intervals
  - Some classification algorithms only accept categorical attributes.
  - Reduce data size by discretization
  - Prepare for further analysis

Discretization by Binning

- Binning is a top-down splitting technique based on a specified number of bins.
- These methods are also used as discretization methods for data reduction and concept hierarchy generation.
- For example, attribute values can be discretized by applying equal-width or equal-frequency binning, and then replacing each bin value by the bin mean or median, as in *smoothing by bin means* or *smoothing by bin medians*, respectively. These techniques can be applied recursively to the resulting partitions to generate concept hierarchies.
- It is unsupervised discretization technique.

Discretization by Histogram Analysis

- Like binning, histogram analysis is an unsupervised discretization technique.
- Various partitioning rules(equal width/equal size) can be used to define histograms.
- The histogram analysis algorithm can be applied recursively to each partition in histogram in order to automatically generate a multilevel concept hierarchy, with the procedure terminating once a prespecified number of concept levels has been reached.
- A *minimum interval size* can also be used per level to control the recursive procedure. This specifies the minimum width of a partition, or the minimum number of values for each partition at each level.

Discretization by Clustering ,Decision Tree, Correlation Analysis

Discretization by Clustering

- Clustering, decision tree analysis, and correlation analysis can be used for data discretization.
- A clustering algorithm can be applied to discretize a numeric attribute, *A*, by partitioning the values of *A* into clusters or groups. Clustering takes the distribution of *A* into consideration, as well as the closeness of data points, and therefore is able to produce high-quality discretization results.
- Clustering can be used to generate a concept hierarchy for *A* by following either a top-down splitting strategy or a bottom-up merging strategy, where each cluster forms a node of the concept hierarchy.

Discretization by Decision Tree:

- Techniques to generate decision trees for classification can be applied to discretization.
- These techniques employ a top-down splitting approach.
- Decision tree approaches to discretization are supervised.

Discretization by Decision Tree, Correlation Analysis

Discretization by Correlation Analysis:

- Measures of correlation can be used for discretization.
- *ChiMerge* is a x^2-based discretization method which employs a bottom-up approach by finding the best neighboring intervals and then merging them to form larger intervals, recursively.
- Supervised method
- The basic notion is that for accurate discretization, the relative class frequencies should be fairly consistent within an interval. Therefore, if two adjacent intervals have a very similar distribution of classes, then the intervals can be merged. Otherwise, they should remain separate.