Data Mining

Concepts and Techniques

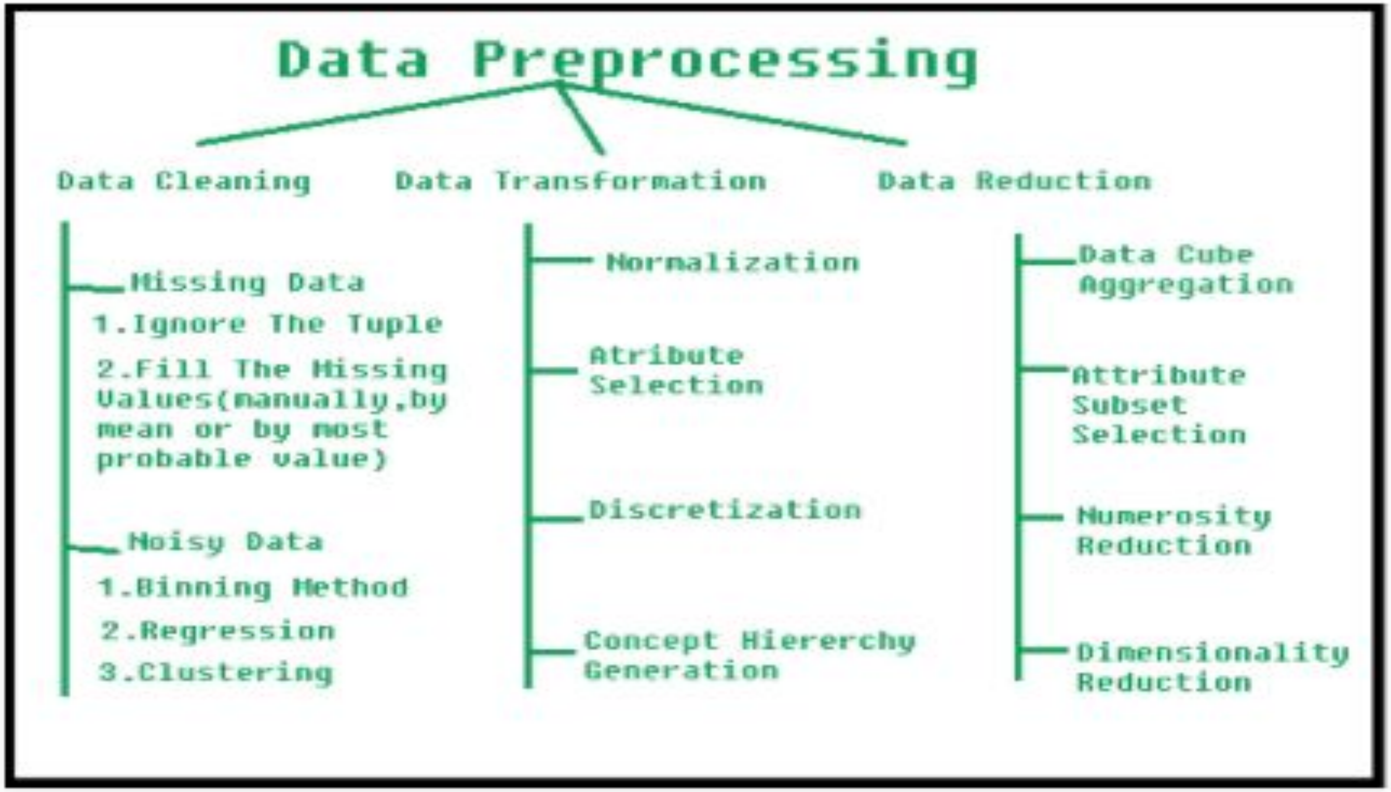
Data Preprocessing

Data Preprocessing

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

- **Preprocessing in Data Mining:**

  Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.

- 

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

Why Is Data Dirty?

- 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
- 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

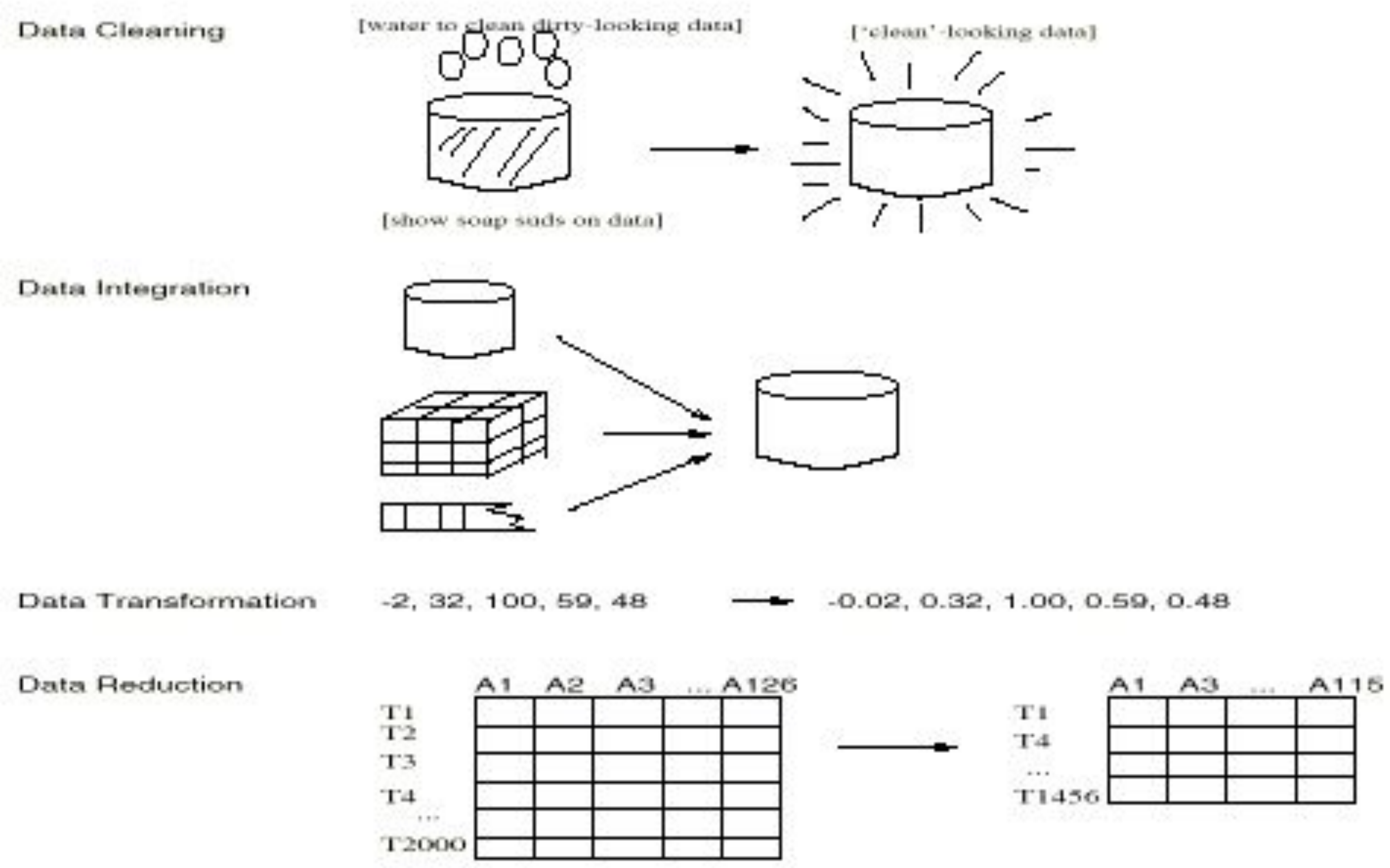
Multi-Dimensional Measure of Data Quality

- A well-accepted multidimensional view:
  - Accuracy
  - Completeness
  - Consistency
  - Timeliness
  - Believability
  - Value added
  - Interpretability
  - Accessibility
- Broad categories for data quality:
  - Intrinsic, contextual, representational, and accessibility

Major Tasks in Data Preprocessing

- Data cleaning
  - Fill in missing values, smooth 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 Cleaning

[water to clean dirty-looking data]

[show soap suds on data]

['clean'-looking data]

Data Integration

Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction

A1 A2 A3 ... A126

T1
T2
T3
T4
...
T2000

A1 A3 ... A115

T1
T4
...
T1456

Data Preprocessing

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

Data Cleaning

- Data cleaning is the process of filling missing values,smoothing noisy data,remove outliers and resolve inconsistencies.
- 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 tasks
  - Fill in missing values
  - Identify outliers and smooth out noisy data
  - Correct inconsistent data
  - Resolve redundancy caused by data integration

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
  - 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 (assuming the tasks in classification—not effective when the percentage 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

- **Handling missing data:**
- Ignore the tuple
- Fill the missing value manually
- Use Global constant to fill in the missing value.
- Use mean or median (measure of Central Tendency) to fill in the missing value.
- Use the attribute mean or median for all samples belonging to the same class as the given tuple.
- Use the most probable value to fill in the missing value.

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may due to
  - faulty data collection instruments
  - data entry problems
  - data transmission problems
  - technology limitation
  - inconsistency in naming convention
- Other data problems which requires data cleaning
  - duplicate records
  - incomplete data
  - inconsistent data

- • Noisy data is meaningless data.
- • It includes any data that cannot be understood and interpreted correctly by machines, such as unstructured text.
- • Noisy data unnecessarily increases the amount of storage space required and can also adversely affect the results of any data mining analysis.
- • Noisy data can be caused by faulty data collection instruments, human or computer errors occurring at data entry, data transmission errors, limited buffer size for coordinating synchronized data transfer, inconsistencies in naming conventions or data codes used and inconsistent formats for input fields(eg:date).

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)

- Binning method smooths a sorted data value by consulting its neighborhood that is value around.
- The values are distributed into number of buckets or bins.
- Data is sorted and partitioned into equal frequency size of bin.
- Smoothing by bin mean:Each value in the Bin is replaced by mean of bin.
- Smoothing by bin median:Each value in the Bin is replaced by median of bin.
- Smoothing by bin boundry:Each value in the Bin is replaced by minimum or maximum value

Simple Discretization Methods: 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
- 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

- Noisy data can be handled by following the given procedures:
- **Binning:**
- Binning methods smooth a sorted data value by consulting the values around it.
- The sorted values are distributed into a number of “buckets,” or bins.
- Because binning methods consult the values around it, they perform local smoothing.
- Similarly, smoothing by bin median scan be employed, in which each bin value is replaced by the bin median.
- In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries.
- Each bin value is then replaced by the closest boundary value.

- In general, the larger the width, the greater the effect of the smoothing.
- Alternatively, bins may be equal-width, where the interval range of values in each bin is constant.
- Binning is also used as a discretization technique.

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

- **Binning Methods for Data Smoothing**
- The binning method can be used for smoothing the data.
- Mostly data is full of noise. Data smoothing is a data pre-processing technique using a different kind of algorithm to remove the noise from the data set.
- This allows important patterns to stand out.
- Unsorted data for price in dollars
- Before sorting: 8 16, 9, 15, 21, 21, 24, 30, 26, 27, 30, 34
- First of all, sort the data
- After Sorting: 8, 9, 15, 16, 21, 21, 24, 26, 27, 30, 30, 34

- Smoothing the data by equal frequency bins
- Bin 1: 8, 9, 15, 16
- Bin 2: 21, 21, 24, 26,
- Bin 3: 27, 30, 30, 34
- Smoothing by bin means
- For Bin 1:
- (8+ 9 + 15 +16 / 4) = 12
- (4 indicating the total values like 8, 9 , 15, 16)
- Bin 1 = 12, 12, 12, 12
- For Bin 2:
- (21 + 21 + 24 + 26 / 4) = 23
- Bin 2 = 23, 23, 23, 23
- For Bin 3:
- (27 + 30 + 30 + 34 / 4) = 30
- Bin 3 = 30, 30, 30, 30

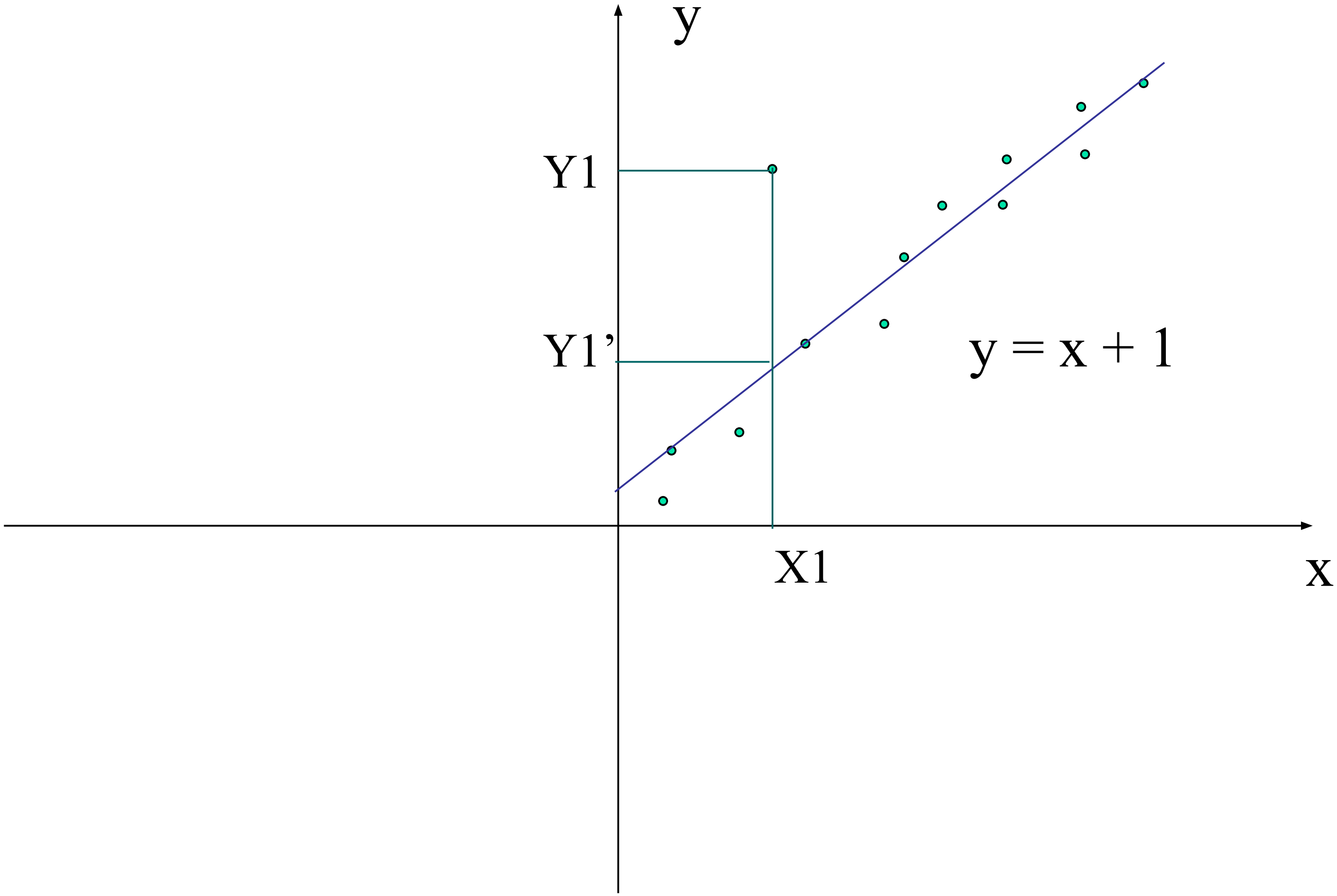
- Smoothing by bin boundaries
- **Bin 1:** 8, 8, 8, 15
- **Bin 2:** 21, 21, 25, 25
- **Bin 3:** 26, 26, 26, 34

- How to smooth data by bin boundaries?
- You need to pick the minimum and maximum value. Put the minimum on the left side and maximum on the right side.
- Now, what will happen to the middle values?
- Middle values in bin boundaries move to its closest neighbor value with less distance.

  Unsorted data for price in dollars:
- **Before sorting:** 8 16, 9, 15, 21, 21, 24, 30, 26, 27, 30, 34
- First of all, sort the data
- **After sorting:** 8, 9, 15, 16, 21, 21, 24, 26, 27, 30, 30, 34

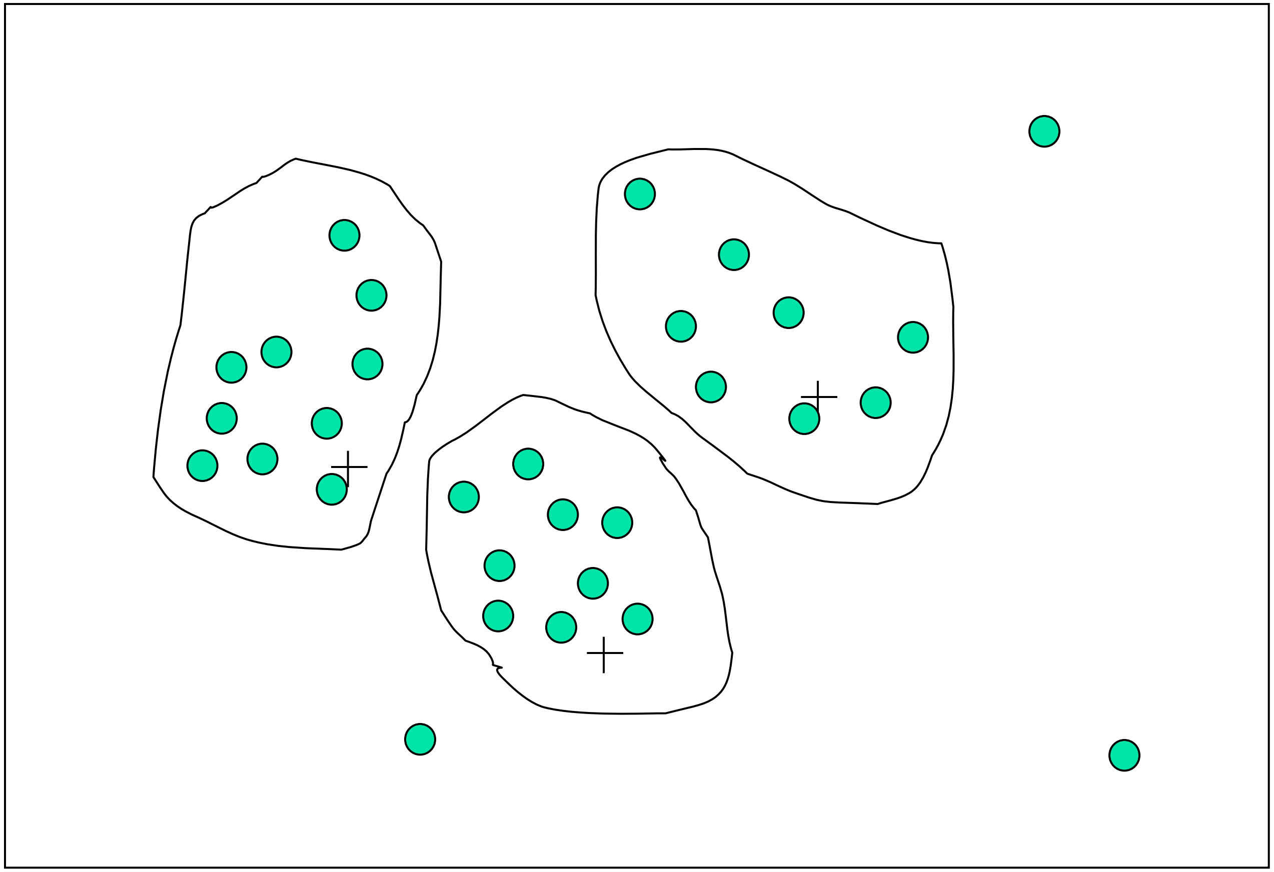
- Smoothing the data by equal frequency bins
- **Bin 1:** 8, 9, 15, 16
- **Bin 2:** 21, 21, 24, 26,
- **Bin 3:** 27, 30, 30, 34
- Smooth data after bin Boundary
- Before bin Boundary: Bin 1: 8, 9, 15, 16
- Here, 1 is the minimum value and 16 is the maximum value.
- 9 is near to 8, so 9 will be treated as 8.
- 15 is more near to 16 and farther away from 8. So, 15 will be treated as 16.
- After bin Boundary: Bin 1: 8, 8, 16, 16
- Before bin Boundary: Bin 2: 21, 21, 24, 26,
- After bin Boundary: Bin 2: 21, 21, 26, 26,
- Before bin Boundary: Bin 3: 27, 30, 30, 34
- **After bin Boundary:** Bin 3: 27, 27, 27, 34

Regression



- Here data can be smoothed by fitting the data to a function.
- Linear regression involves finding the “best” line to fit two attributes, so that one attribute can be used to predict the other.
- Multiple linear regression is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

Cluster Analysis



- Outliers may be detected by clustering, where similar values are organized into groups, or “clusters.”
- Similarly, values that fall outside of the set of clusters may also be considered outliers.

Data Cleaning as a Process

- Data discrepancy detection
  - Use metadata (e.g., domain, range, dependency, distribution)
  - Check field overloading
  - Check uniqueness rule, consecutive rule and null rule
  - Use commercial tools
    - Data scrubbing: use simple domain knowledge (e.g., postal code, spell-check) to detect errors and make corrections
    - Data auditing: by analyzing data to discover rules and relationship to detect violators (e.g., correlation and clustering to find outliers)
- Data migration and integration
  - Data migration tools: allow transformations to be specified
  - ETL (Extraction/Transformation/Loading) tools: allow users to specify transformations through a graphical user interface
- Integration of the two processes
  - Iterative and interactive (e.g., Potter's Wheels)

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
- 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

Handling Redundancy in Data Integration

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

Correlation Analysis (Numerical Data)

- **Correlation** is a bivariate analysis that measures the strength of association between two variables and the direction of the relationship.
- Specifically, in terms of the strength of relationship, the value of the correlation coefficient varies between +1 and -1.
- For instance, a value of ± 1 indicates a perfect degree of association between the two variables.
- On the other hand, as the correlation coefficient value goes towards 0, the relationship between the two variables will be weaker.
- Additionally, the sign of the coefficient indicates direction of the relationship a + sign indicates a positive relationship and a – sign indicates a negative relationship.

Correlation Coefficient Formula:

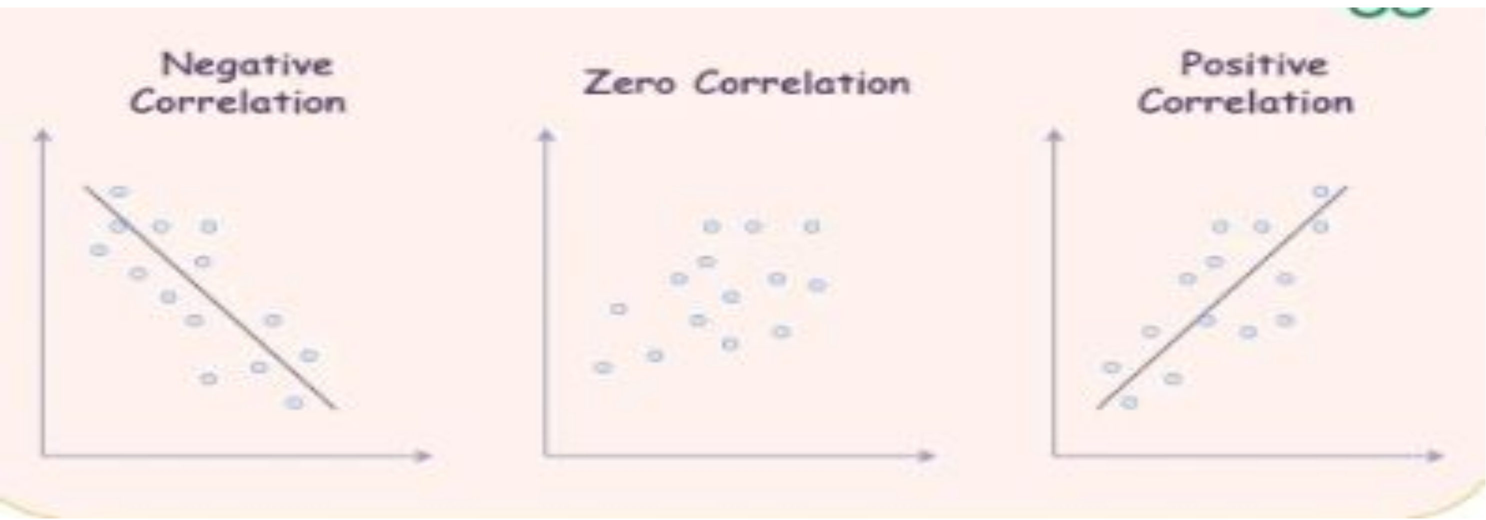
- The correlation coefficient is a statistical measure used to quantify the relationship between predicted and observed values in a statistical analysis.
- It provides insight into the degree of precision between these predicted and actual values.

What is Correlation?

Correlation is a statistical measure that describes the extent to which two variables are related to each other.

It quantifies the direction and strength of the linear relationship between variables. Generally, a correlation between any two variables is of three types that include:

- Positive Correlation
- Zero Correlation
- Negative Correlation



Correlation Coefficient Definition

A statistical measure that quantifies the strength and direction of the linear relationship between two variables is called the Correlation coefficient. Generally, it is denoted by the symbol ‘r’ and ranges from -1 to 1.

What is Correlation Coefficient Formula?

Correlation coefficient procedure is used to determine how strong a relationship is between the data. The correlation coefficient procedure yields a value between 1 and -1. In which,

- -1 indicates a strong negative relationship
- 1 indicates strong positive relationships
- Zero implies no connection at all

Types of Correlation:

Pearson correlation:

- Best used when data is normally distributed and a linear relationship is expected between variables.
- Example: Examining the correlation between study hours and exam scores.

Spearman correlation:

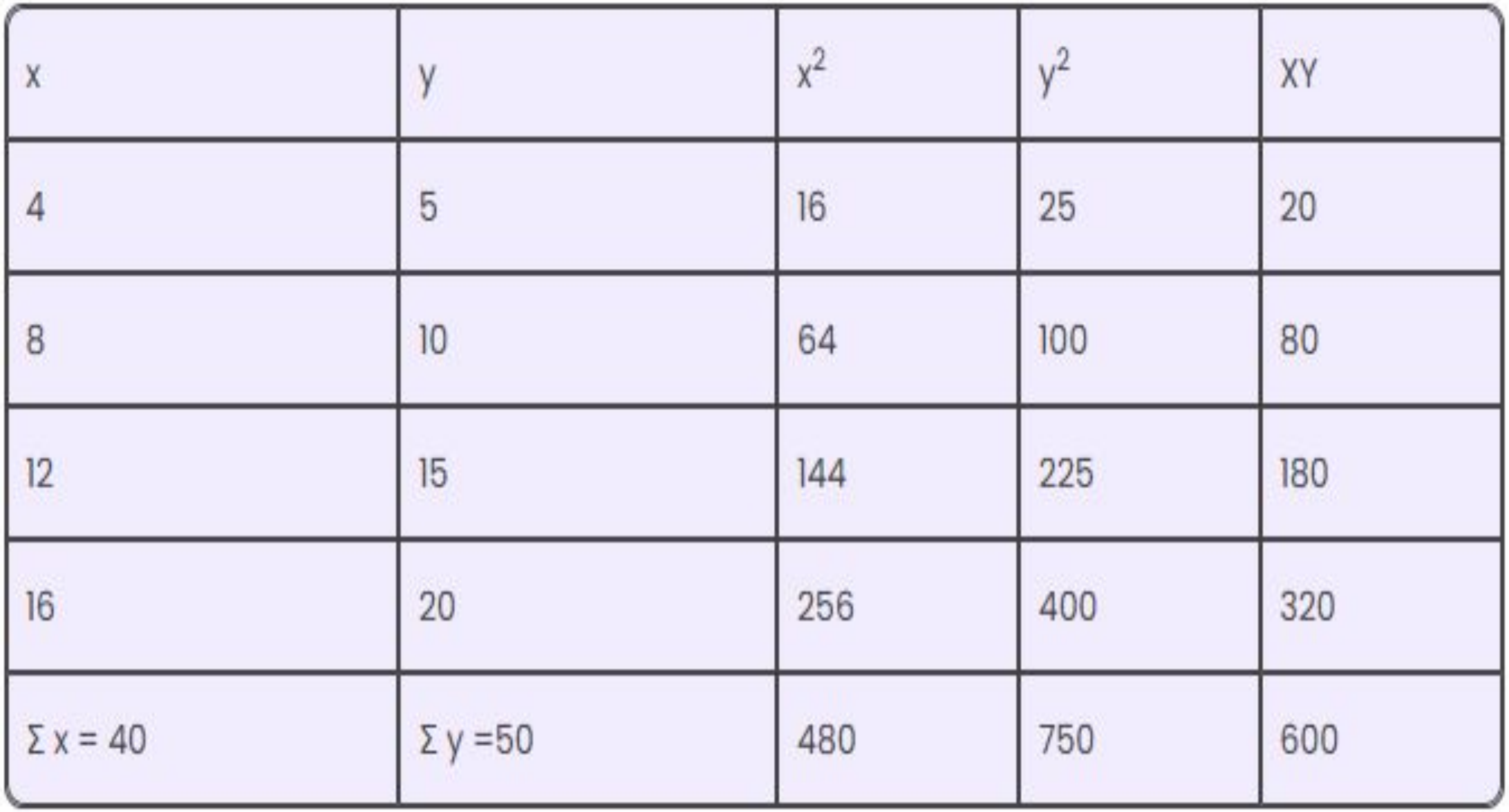
- More appropriate for data with outliers or when the relationship is not strictly linear but still shows a consistent trend (monotonic).
- Example: Analyzing the correlation between customer satisfaction ranking and their product review scores.
- Kindall **correlation:**Kendall rank correlation is a non-parametric test that measures the strength of dependence between two variables.

https://www.scribbr.com/statistics/pearson-correlation-coefficient/

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[\, n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}}$$

$$r_{xy} = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sqrt{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}\sqrt{n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2}}$$

Calculate the linear correlation coefficient for the following data. X = 4, 8 ,12, 16 and Y = 5, 10, 15, 20.



x	y	x^2	y^2	XY
4	5	16	25	20
8	10	64	100	80
12	15	144	225	180
16	20	256	400	320
Σ x = 40	Σ y =50	480	750	600

According to the formula of linear correlation we have,

$$r(xy) = \frac{(4\times600)-(40\times50)}{\sqrt{4(480)-40^2}\sqrt{4(750)-50^2}}$$

$$r(xy) = \frac{2400-2000}{\sqrt{1920-1600}\sqrt{3000-2500}}$$

$$r(xy) = \frac{400}{\sqrt{320}\sqrt{500}}$$

$$r(xy) = \frac{400}{17.89\times22.36}$$

$$r(xy) = \frac{400}{400} = 1$$

Therefore, r(xy) = 1

Correlation Analysis (Categorical Data)

- X^2 (chi-square) test

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

- Assumptions:
  - Random samples of categorical variables
  - Each individual appears in only one cell
- The larger the X^2 value, the more likely the variables are related
- 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

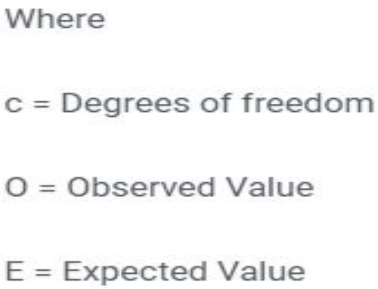
The Chi-Square test is a statistical procedure for determining the difference between observed and expected data.

This test can also be used to decide whether it correlates to our data's categorical variables.

It helps to determine whether a difference between two categorical variables is due to chance or a relationship between them.

Chi-Square Test Formula:

$$x_c^2 = \frac{\Sigma \ (O_i \ - E_i)^2}{E_i}$$



Where

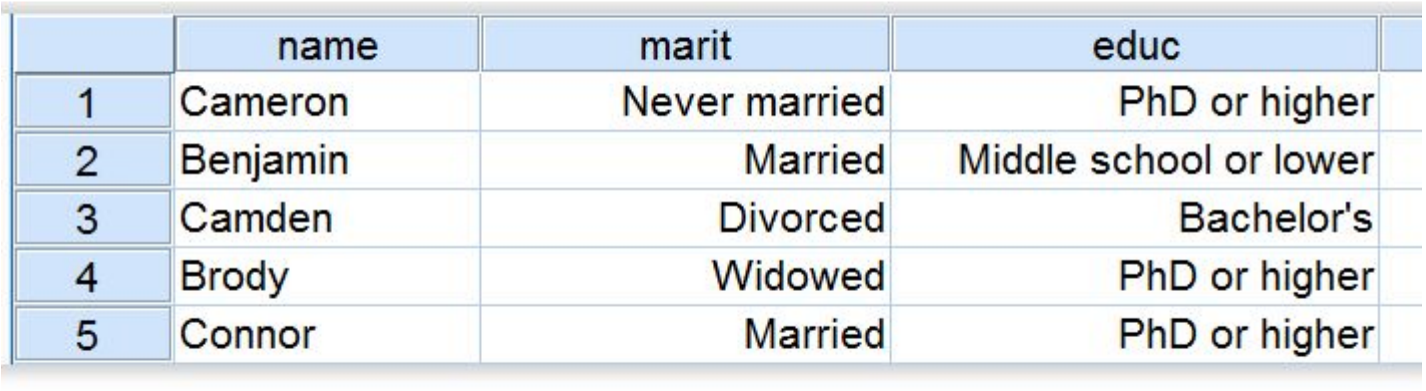
c = Degrees of freedom

O = Observed Value

E = Expected Value

- The degrees of freedom in a statistical calculation represent the number of variables that can vary.
- The degrees of freedom can be calculated to ensure that chi-square tests are statistically valid.
- These tests are frequently used to compare observed data with data expected to be obtained if a particular hypothesis were true.
- The Observed values are those you gather yourselves.
- The expected values are the anticipated frequencies, based on the null hypothesis.

- A chi-squared test is a statistical hypothesis test that is valid to perform when the test statistic is chi-squared distributed under the null hypothesis, specifically Pearson's chi-squared test and variants thereof.
-

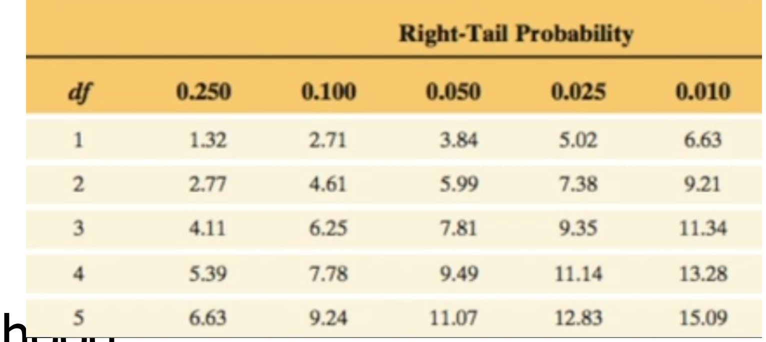


	name	marit	educ
1	Cameron	Never married	PhD or higher
2	Benjamin	Married	Middle school or lower
3	Camden	Divorced	Bachelor's
4	Brody	Widowed	PhD or higher
5	Connor	Married	PhD or higher

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 2

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:

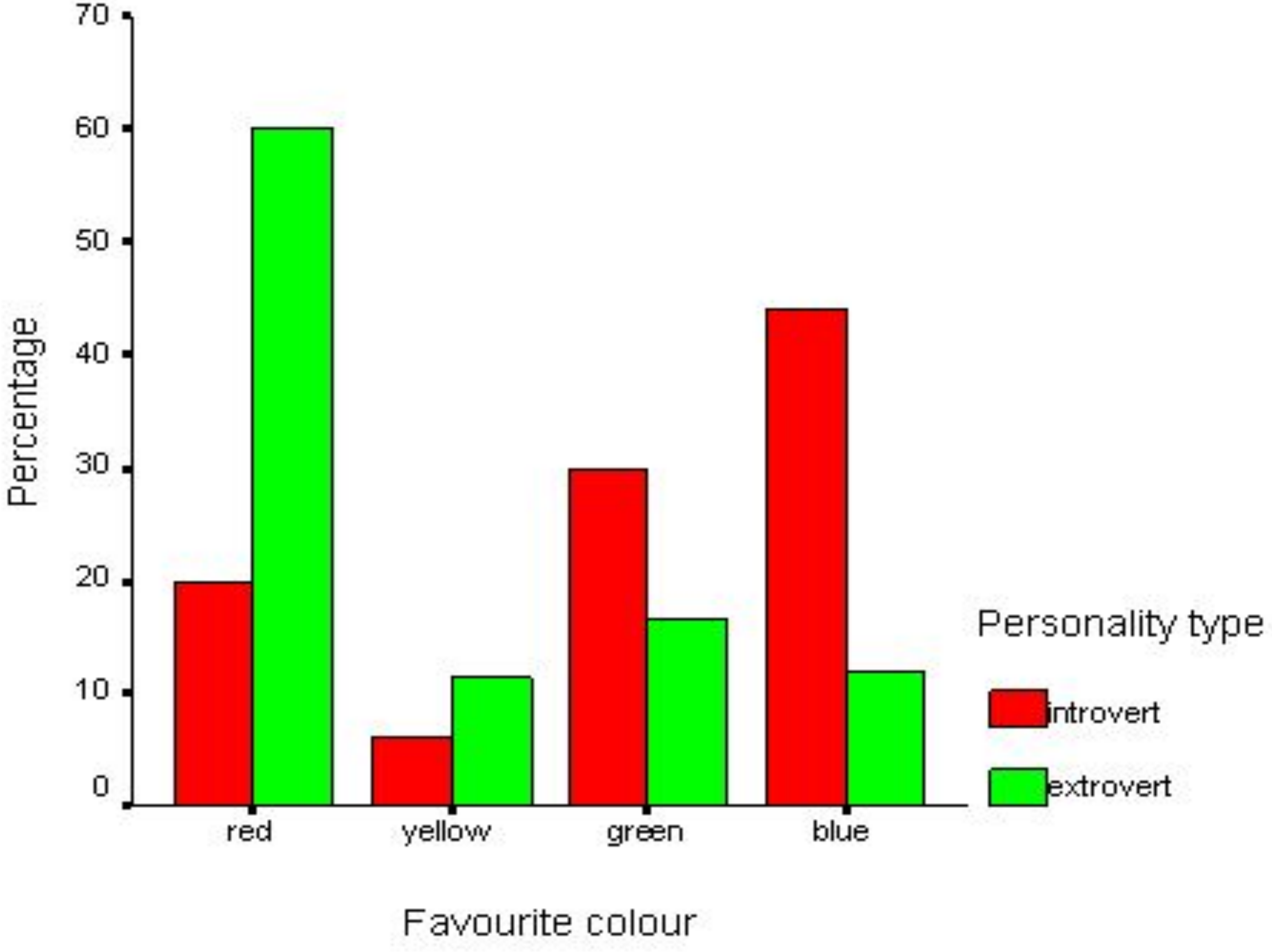
(Observed counts)	Colors				
	Red	**Yellow**	**Green**	**Blue**	**Totals**
Introvert personality	20	6	30	44	100
Extrovert personality	180	34	50	36	300
Totals	200	40	80	80	400

Suitable null and alternative hypotheses might be:

- H_0: Color preference is not associated with personality, and
- 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: Example 2

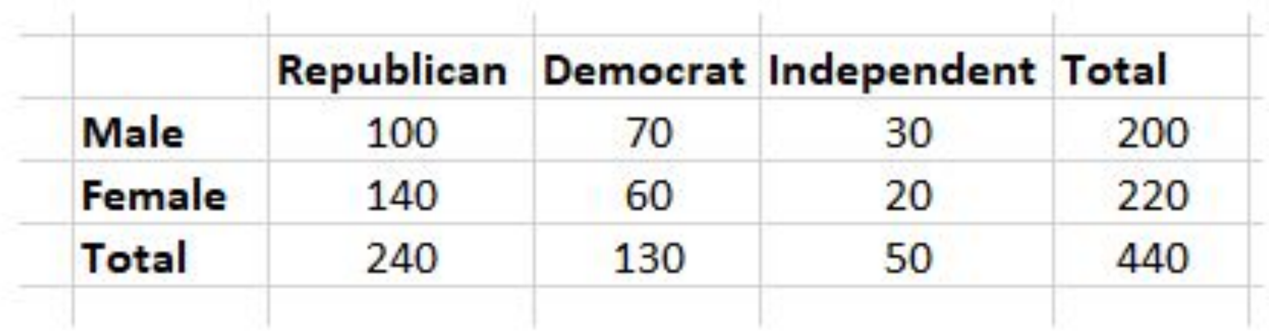


Chi-Square Calculation: Example 3

- 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

Let's say you want to know if gender has anything to do with political party preference. You poll 440 voters in a simple random sample to find out which political party they prefer.



	Republican	Democrat	Independent	Total
Male	100	70	30	200
Female	140	60	20	220
Total	240	130	50	440

To see if gender is linked to political party preference, perform a Chi-Square test of independence using the steps below:

Step 1: Define the Hypothesis

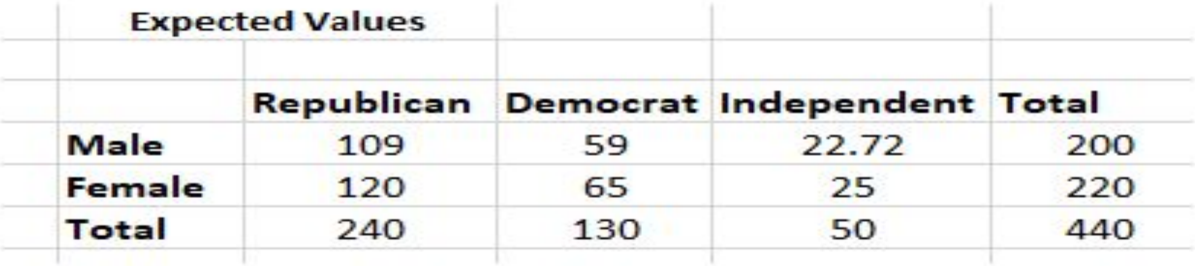
H0: There is no link between gender and political party preference.

H1: There is a link between gender and political party preference.

Step 2: Calculate the Expected Values

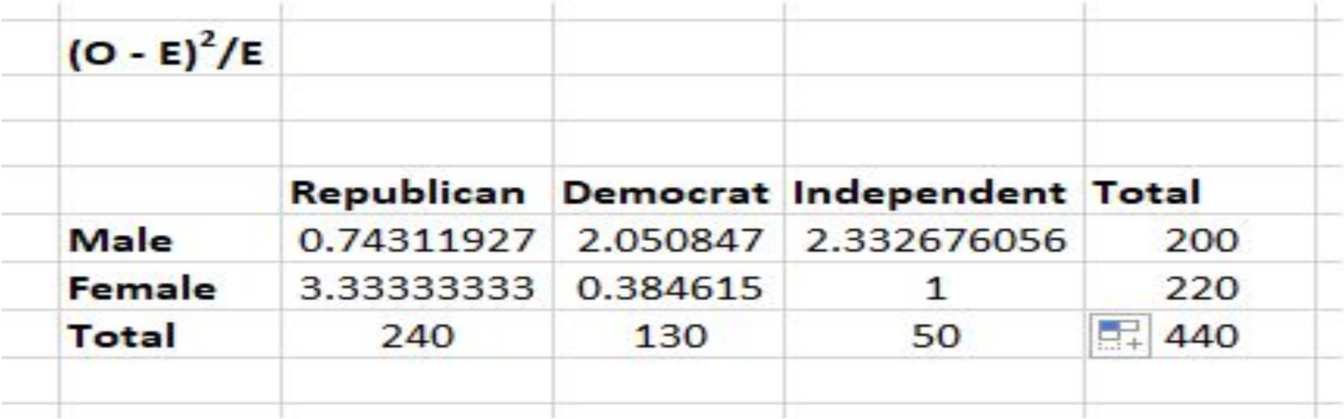
Now you will calculate the expected frequency.

$$\text{Expected Value} = \frac{(Row\ Total) * (Column\ Total)}{Total\ Number\ Of\ Observations}$$



Expected Values				
	Republican	Democrat	Independent	Total
Male	109	59	22.72	200
Female	120	65	25	220
Total	240	130	50	440

Step 3: Calculate (O-E)2 / E for Each Cell in the Table



$(O - E)^2/E$				
	Republican	Democrat	Independent	Total
Male	0.74311927	2.050847	2.332676056	200
Female	3.33333333	0.384615	1	220
Total	240	130	50	440

Step 4: Calculate the Test Statistic X2

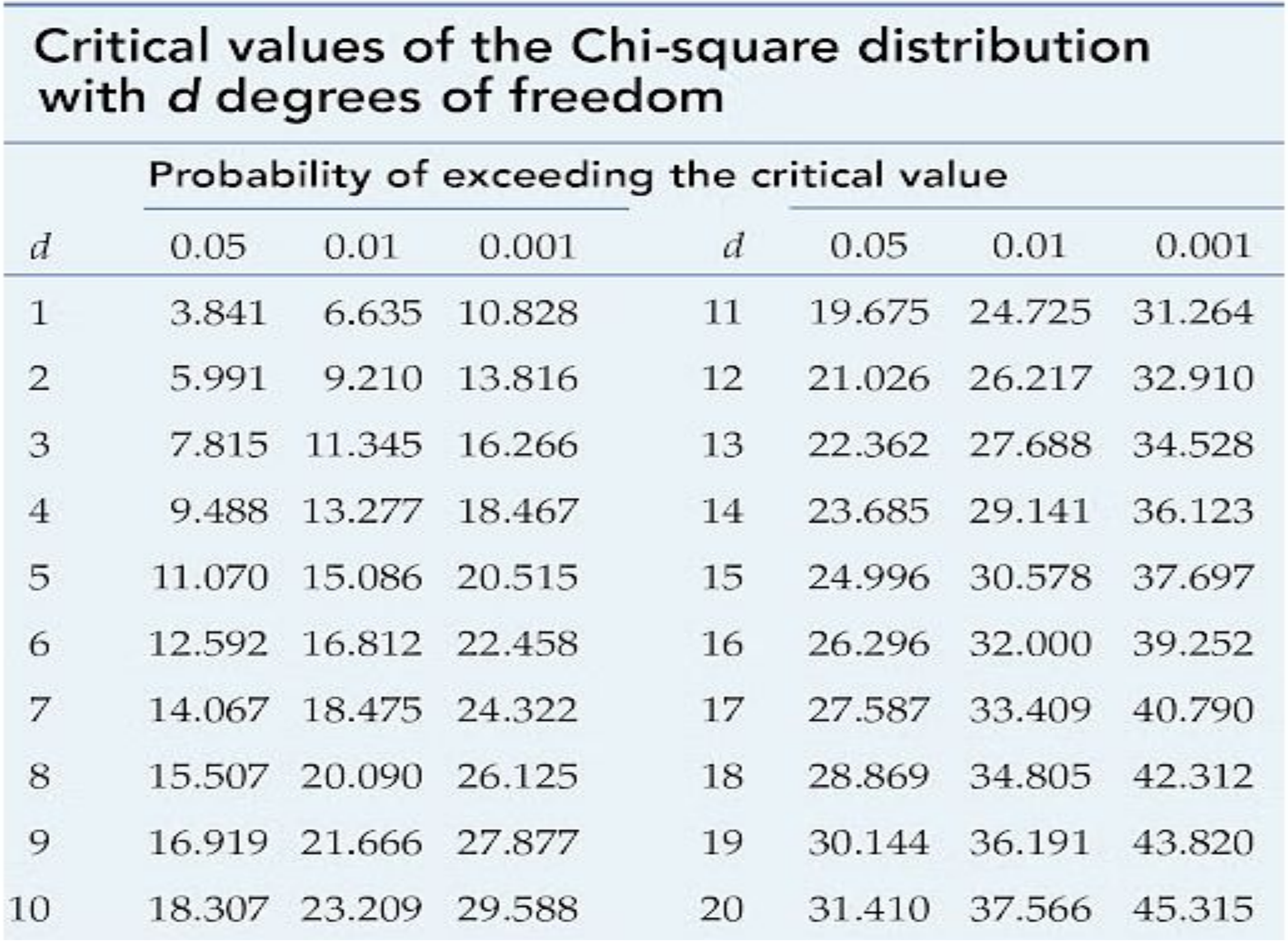
X2 is the sum of all the values in the last table

= 0.743 + 2.05 + 2.33 + 3.33 + 0.384 + 1

= 9.837

Determine the critical statistic, which requires determining our degrees of freedom. The degrees of freedom in this case are equal to the table's number of columns minus one multiplied by the table's number of rows minus one, or (r-1) (c-1). We have (3-1)(2-1) = 2.

- Finally, you compare our obtained statistic to the critical statistic found in the chi-square table. As you can see, for an alpha level of 0.05 and two degrees of freedom, the critical statistic is 5.991, which is less than our obtained statistic of 9.83. You can reject our null hypothesis because the critical statistic is higher than your obtained statistic.
- This means you have sufficient evidence to say that there is an association between gender and political party preference.

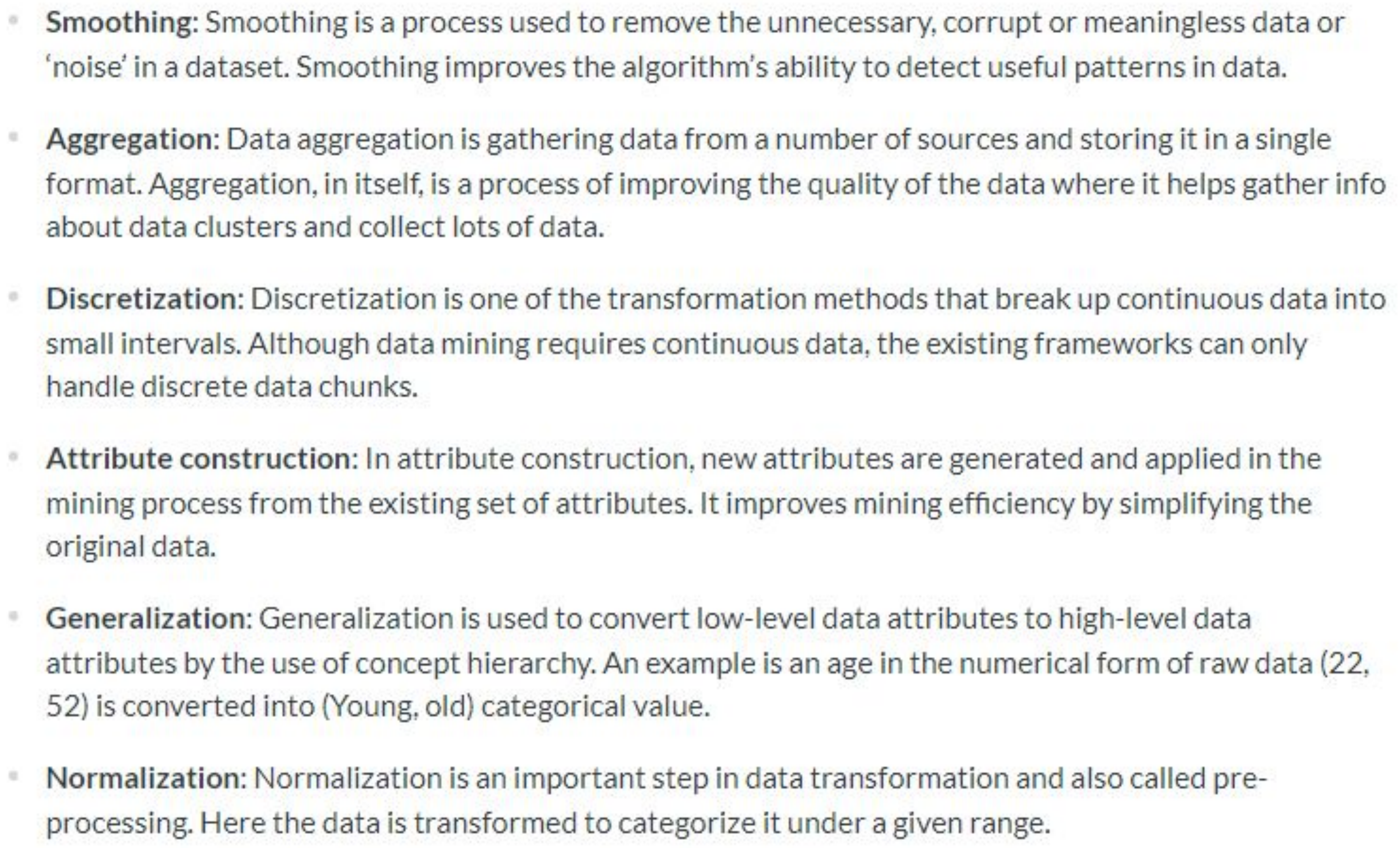


Critical values of the Chi-square distribution with *d* degrees of freedom

	Probability of exceeding the critical value						
d	0.05	0.01	0.001	*d*	0.05	0.01	0.001
1	3.841	6.635	10.828	11	19.675	24.725	31.264
2	5.991	9.210	13.816	12	21.026	26.217	32.910
3	7.815	11.345	16.266	13	22.362	27.688	34.528
4	9.488	13.277	18.467	14	23.685	29.141	36.123
5	11.070	15.086	20.515	15	24.996	30.578	37.697
6	12.592	16.812	22.458	16	26.296	32.000	39.252
7	14.067	18.475	24.322	17	27.587	33.409	40.790
8	15.507	20.090	26.125	18	28.869	34.805	42.312
9	16.919	21.666	27.877	19	30.144	36.191	43.820
10	18.307	23.209	29.588	20	31.410	37.566	45.315

Data Transformation

- Smoothing: remove noise from data
- Aggregation: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling
- Attribute/feature construction
  - New attributes constructed from the given ones



- **Smoothing:** Smoothing is a process used to remove the unnecessary, corrupt or meaningless data or 'noise' in a dataset. Smoothing improves the algorithm's ability to detect useful patterns in data.
- **Aggregation:** Data aggregation is gathering data from a number of sources and storing it in a single format. Aggregation, in itself, is a process of improving the quality of the data where it helps gather info about data clusters and collect lots of data.
- **Discretization:** Discretization is one of the transformation methods that break up continuous data into small intervals. Although data mining requires continuous data, the existing frameworks can only handle discrete data chunks.
- **Attribute construction**: In attribute construction, new attributes are generated and applied in the mining process from the existing set of attributes. It improves mining efficiency by simplifying the original data.
- **Generalization**: Generalization is used to convert low-level data attributes to high-level data attributes by the use of concept hierarchy. An example is an age in the numerical form of raw data (22, 52) is converted into (Young, old) categorical value.
- **Normalization**: Normalization is an important step in data transformation and also called pre-processing. Here the data is transformed to categorize it under a given range.

Data Transformation: Normalization

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

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

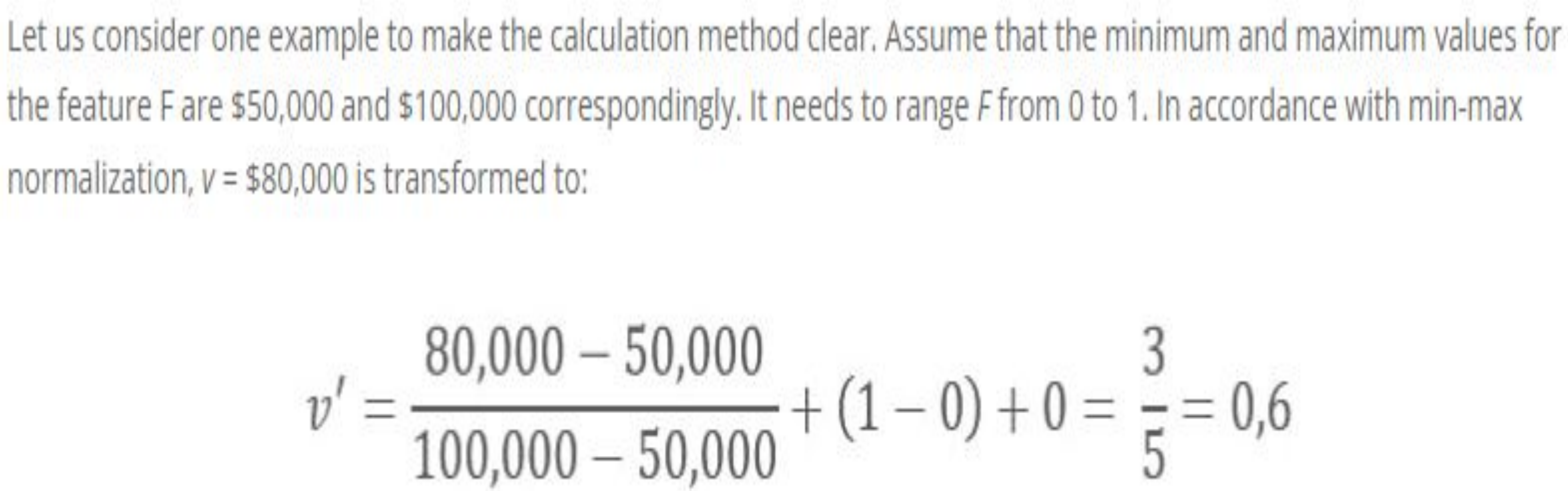
  - Ex. Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,600 is mapped to $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$
- Z-score normalization (μ: mean, σ: standard deviation):

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

  - Ex. Let μ = 54,000, σ = 16,000. Then $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$
- Normalization by decimal scaling

$$v' = \frac{v}{10^j}$$ Where j is the smallest integer such that Max(|v'|) < 1

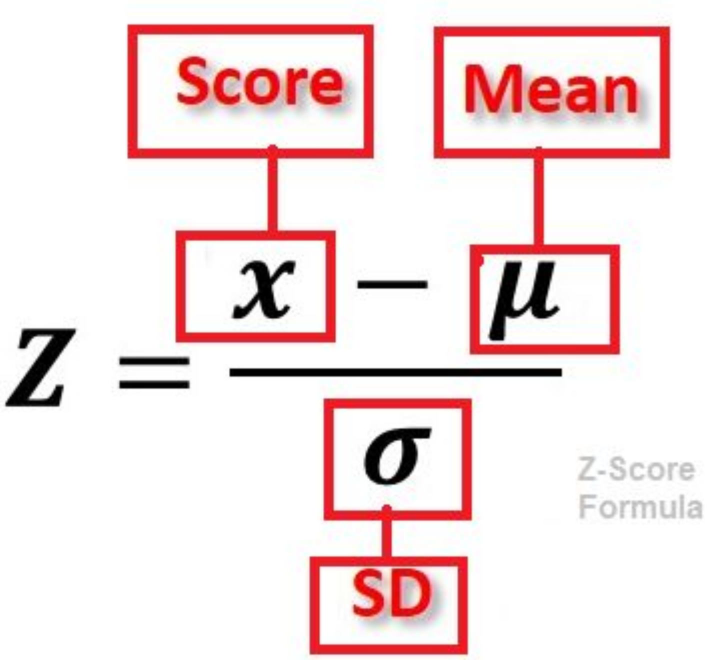
- Min-Max normalization
- The first technique we will cover is min-max normalization. It is the linear transformation of the original unstructured data. It scales the data from 0 to 1. It is calculated by the following formula:
- $$v' = \frac{v - \mathrm{minF}}{\mathrm{maxF} - \mathrm{minF}}(new\_max_F - new\_min_F) + new\_min_F \ ,$$
- where is the current value of feature F.



Let us consider one example to make the calculation method clear. Assume that the minimum and maximum values for the feature F are $50,000 and $100,000 correspondingly. It needs to range *F* from 0 to 1. In accordance with min-max normalization, *v* = $80,000 is transformed to:

$$v' = \frac{80{,}000 - 50{,}000}{100{,}000 - 50{,}000} + (1 - 0) + 0 = \frac{3}{5} = 0{,}6$$

- Z-Score Normalization – (Data Mining)
- Z-Score helps in the normalization of data.
- If we normalize the data into a simpler form with the help of z score normalization, then it’s very easy to understand by our brains.
- Z- Score Formula

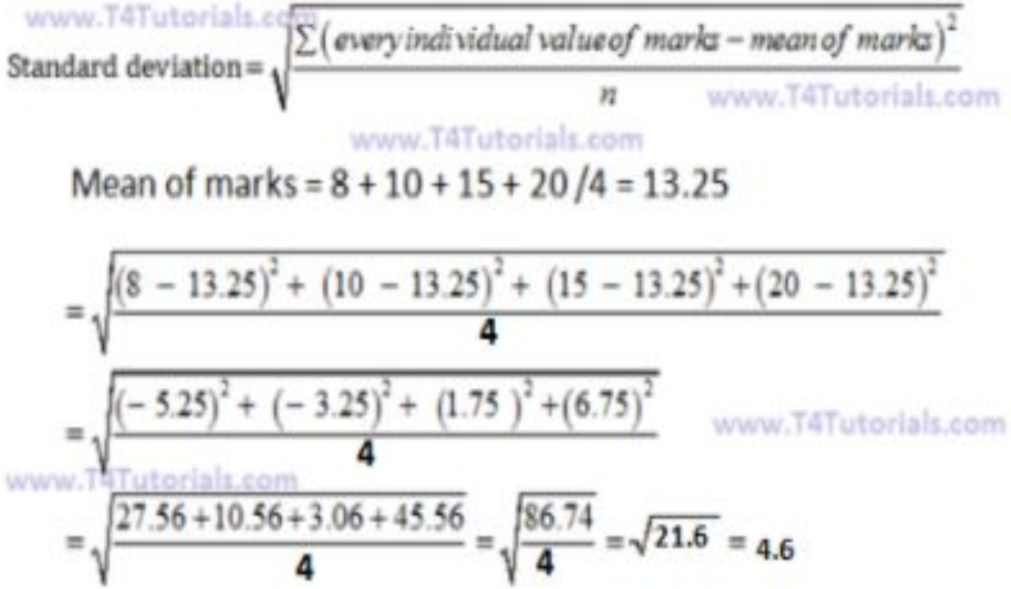


$$Z = \frac{x - \mu}{\sigma}$$

How to calculate Z-Score of the following data?

marks
8
10
15
20

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n}}$$



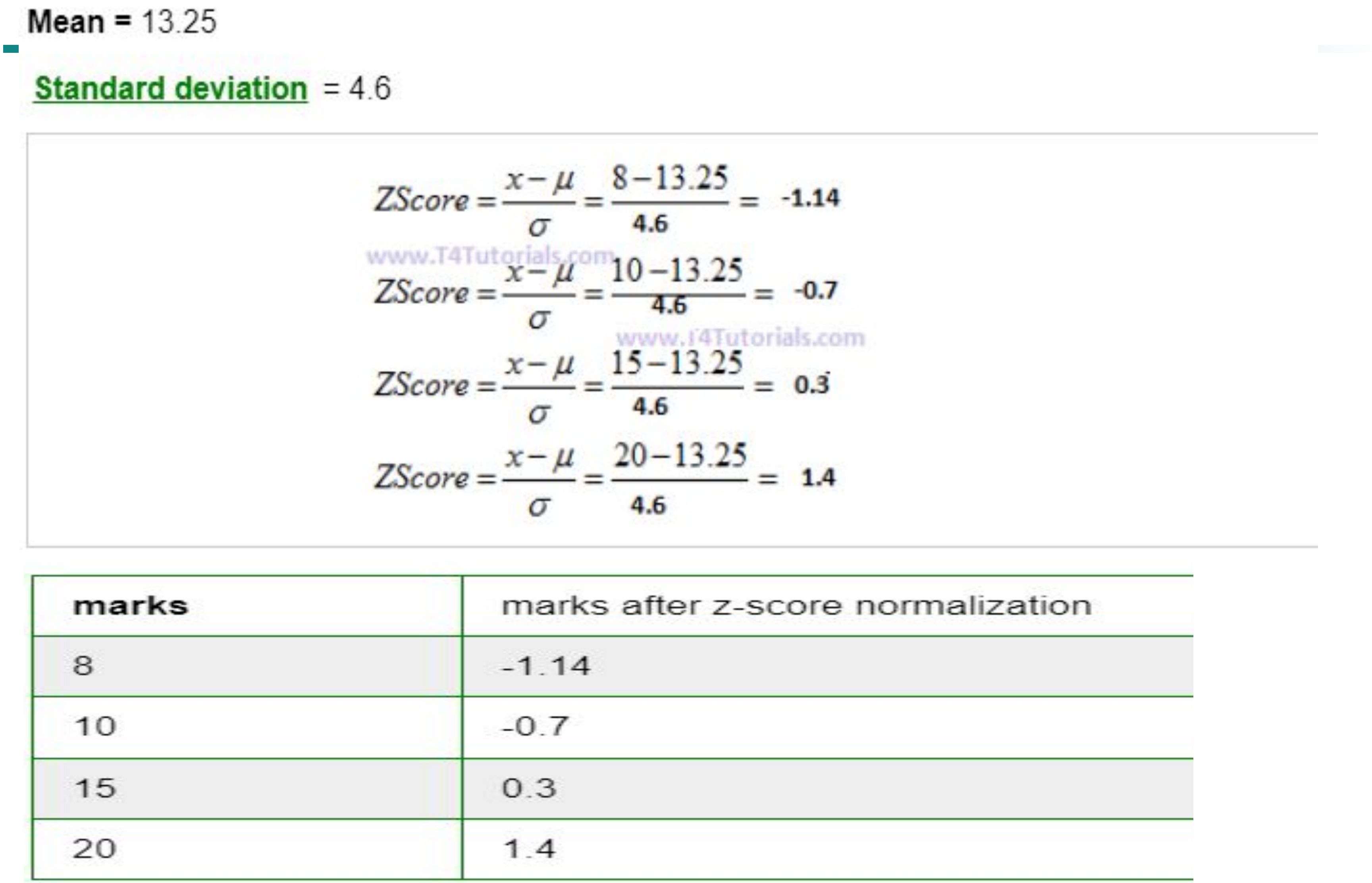
$$\text{Standard deviation} = \sqrt{\frac{\sum(\text{every individual value of marks} - \text{mean of marks})^2}{n}}$$

Mean of marks = 8 + 10 + 15 + 20 /4 = 13.25

$$= \sqrt{\frac{(8 - 13.25)^2 + (10 - 13.25)^2 + (15 - 13.25)^2 + (20 - 13.25)^2}{4}}$$

$$= \sqrt{\frac{(-5.25)^2 + (-3.25)^2 + (1.75)^2 + (6.75)^2}{4}}$$

$$= \sqrt{\frac{27.56 + 10.56 + 3.06 + 45.56}{4}} = \sqrt{\frac{86.74}{4}} = \sqrt{21.6} = 4.6$$



Mean = 13.25

Standard deviation = 4.6

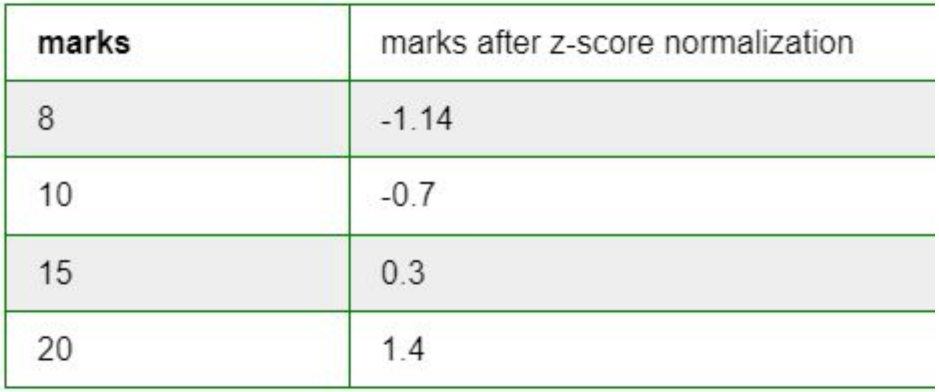
$$ZScore = \frac{x-\mu}{\sigma} = \frac{8-13.25}{4.6} = -1.14$$

$$ZScore = \frac{x-\mu}{\sigma} = \frac{10-13.25}{4.6} = -0.7$$

$$ZScore = \frac{x-\mu}{\sigma} = \frac{15-13.25}{4.6} = 0.\dot{3}$$

$$ZScore = \frac{x-\mu}{\sigma} = \frac{20-13.25}{4.6} = 1.4$$

marks	marks after z-score normalization
8	-1.14
10	-0.7
15	0.3
20	1.4



marks	marks after z-score normalization
8	-1.14
10	-0.7
15	0.3
20	1.4

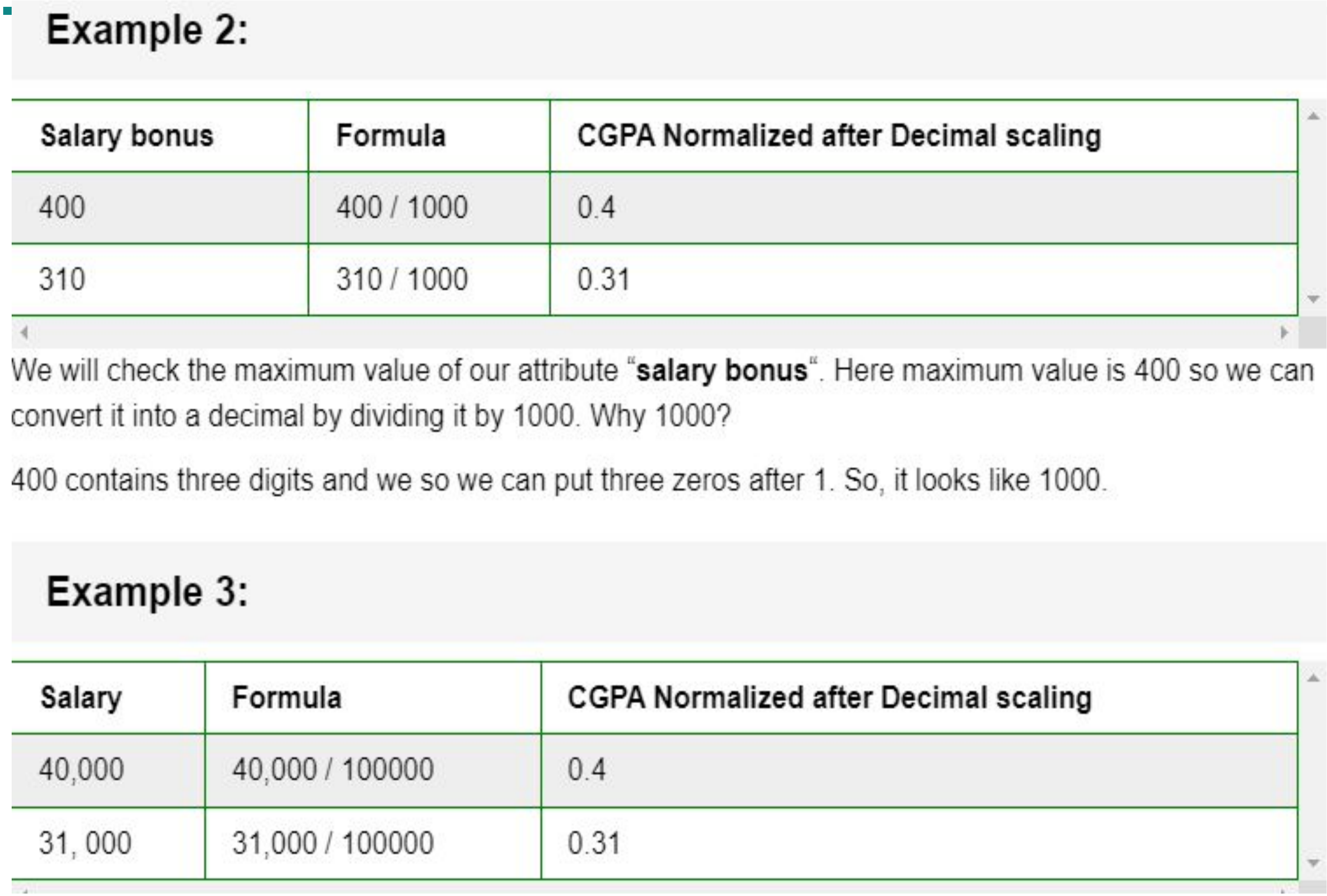
- Decimal scaling with Examples
- Decimal scaling is a data normalization technique like Z score, Min-Max, and normalization with standard deviation.
- Decimal scaling is a data normalization technique.
- In this technique, we move the decimal point of values of the attribute.
- This movement of decimal points totally depends on the maximum value among
- A value v of attribute A is can be normalized by the following formula

- Normalized value of attribute = (vi / 10j)

- We will check the maximum value among our attribute CGPA. Here maximum value is 3 so we can convert it to a decimal by dividing by 10. Why 10?

- we will count total numbers in our maximum value and then put 1 and after 1 we can put zeros equal to the length of the maximum value.

- Here 3 is the maximum value and the total numbers in this value are only 1. so we will put one zero after one.



Example 2:

Salary bonus	Formula	CGPA Normalized after Decimal scaling
400	400 / 1000	0.4
310	310 / 1000	0.31

We will check the maximum value of our attribute “**salary bonus**“. Here maximum value is 400 so we can convert it into a decimal by dividing it by 1000. Why 1000?

400 contains three digits and we so we can put three zeros after 1. So, it looks like 1000.

Example 3:

Salary	Formula	CGPA Normalized after Decimal scaling
40,000	40,000 / 100000	0.4
31, 000	31,000 / 100000	0.31

Data Preprocessing

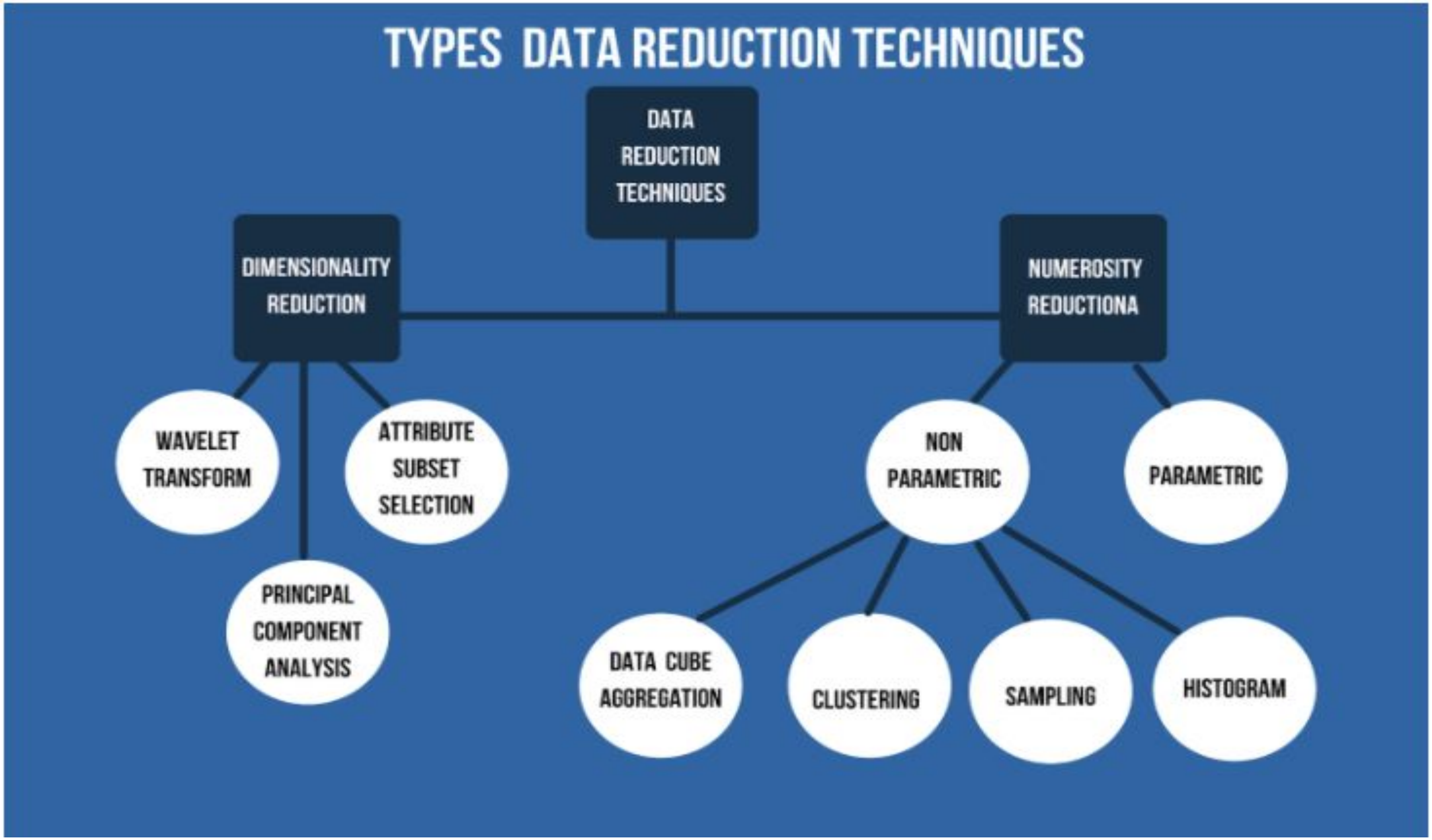
- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary
- https://t4tutorials.com/what-are-quartiles-in-data-mining/

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
  - Data cube aggregation:
  - Dimensionality reduction — e.g., remove unimportant attributes
  - Data Compression
  - Numerosity reduction — e.g., fit data into models
  - Discretization and concept hierarchy generation

- Data reduction is the transformation of numerical or alphabetical digital information derived empirically or experimentally into a corrected, ordered, and simplified form.”

DATA REDUCTION TECHNIQUES



- **A) Dimensionality Reduction**
- Dimensionality Reduction is the process of reducing the number of dimensions the data is spread across.
- It means, the attributes or features, that the data set carries as the number of dimensions increases the sparsity.
- This sparsity is critical to clustering, outlier analysis and other algorithms.
- With reduced dimensionality, it is easy to visualize and manipulate data. There are three types of Dimensionality reduction.
- **Wavelet Transform:**
- Wavelet Transform is a lossy method for dimensionality reduction, where a data vector X is transformed into another vector X', in such a way that both X and X' still represent the same length.
- The result of wavelet transform can be truncated, unlike its original, thus achieving dimensionality reduction.
- Wavelet transforms are well suited for data cube, sparse data or data which is highly skewed. Wavelet transform is often used in image compression.
- **Principal Component Analysis**
- This method involves the identification of a few independent tuples with 'n' attributes that can represent the entire data set. This method can be applied to skewed and sparse data.

- **Attribute Subset Selection**
- Here, attributes irrelevant to data mining or redundant ones are not included in a core attribute subset. The core attribute subset selection reduces the data volume and dimensionality.

- **B) Numerosity Reduction**
- This method uses alternate, small forms of data representation, thus reducing data volume. There are two types of Numerosity reduction, Parametric and Non-Parametric.

- **F) Data Cube Aggregation**
- Data Cube Aggregation is a multidimensional aggregation that uses aggregation at various levels of a data cube to represent the original data set, thus achieving data reduction. Data Cube Aggregation, where the data cube is a much more efficient way of storing data, thus achieving data reduction, besides faster aggregation operations.
- **G) Data Compression**
- It employs modification, encoding or converting the structure of data in a way that consumes less space. Data compression involves building a compact representation of information by removing redundancy and representing data in binary form. Data that can be restored successfully from its compressed form is called Lossless compression while the opposite where

Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
  - The aggregated data for an individual entity of interest
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

Class Characterization: An Example

Initial Relation

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
…	…	…	…	…	…	…	…
Removed	Retained	Sci,Eng, Bus	Country	Age range	City	Removed	Excl, VG,..

Prime Generalized Relation

Gender	Major	Birth_region	Age_range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very-good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
…	…	…	…	…	…	…

Birth_Region Gender	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

Basic Principles of Attribute-Oriented Induction

- Data focusing: task-relevant data, including dimensions, and the result is the *initial relation*
- Attribute-removal: remove attribute *A* if there is a large set of distinct values for *A* but (1) there is no generalization operator on *A*, or (2) *A*'s higher level concepts are expressed in terms of other attributes
- Attribute-generalization: If there is a large set of distinct values for *A*, and there exists a set of generalization operators on *A*, then select an operator and generalize *A*
- Attribute-threshold control: typical 2-8, specified/default
- Generalized relation threshold control: control the final relation/rule size

Attribute-Oriented Induction: Basic Algorithm

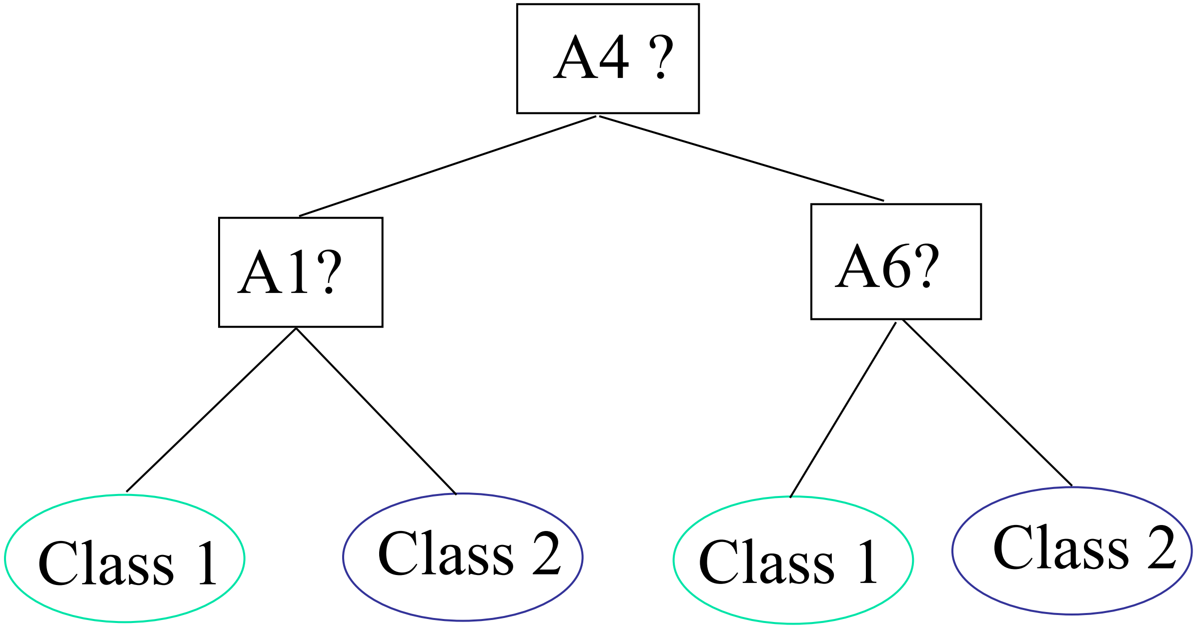
- InitialRel: Query processing of task-relevant data, deriving the *initial relation*.
- PreGen: Based on the analysis of the number of distinct values in each attribute, determine generalization plan for each attribute: removal? or how high to generalize?
- PrimeGen: Based on the PreGen plan, perform generalization to the right level to derive a “prime generalized relation”, accumulating the counts.
- Presentation: User interaction: (1) adjust levels by drilling, (2) pivoting, (3) mapping into rules, cross tabs, visualization presentations.

Attribute Subset Selection

- Feature selection (i.e., attribute subset selection):
  - Select a minimum set of features such that the probability distribution of different classes given the values for those features is as close as possible to the original distribution given the values of all features
  - reduce # of patterns in the patterns, easier to understand
- Heuristic methods (due to exponential # of choices):
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

Example of Decision Tree Induction

Initial attribute set:
{A1, A2, A3, A4, A5, A6}



⟶ Reduced attribute set: {A1, A4, A6}

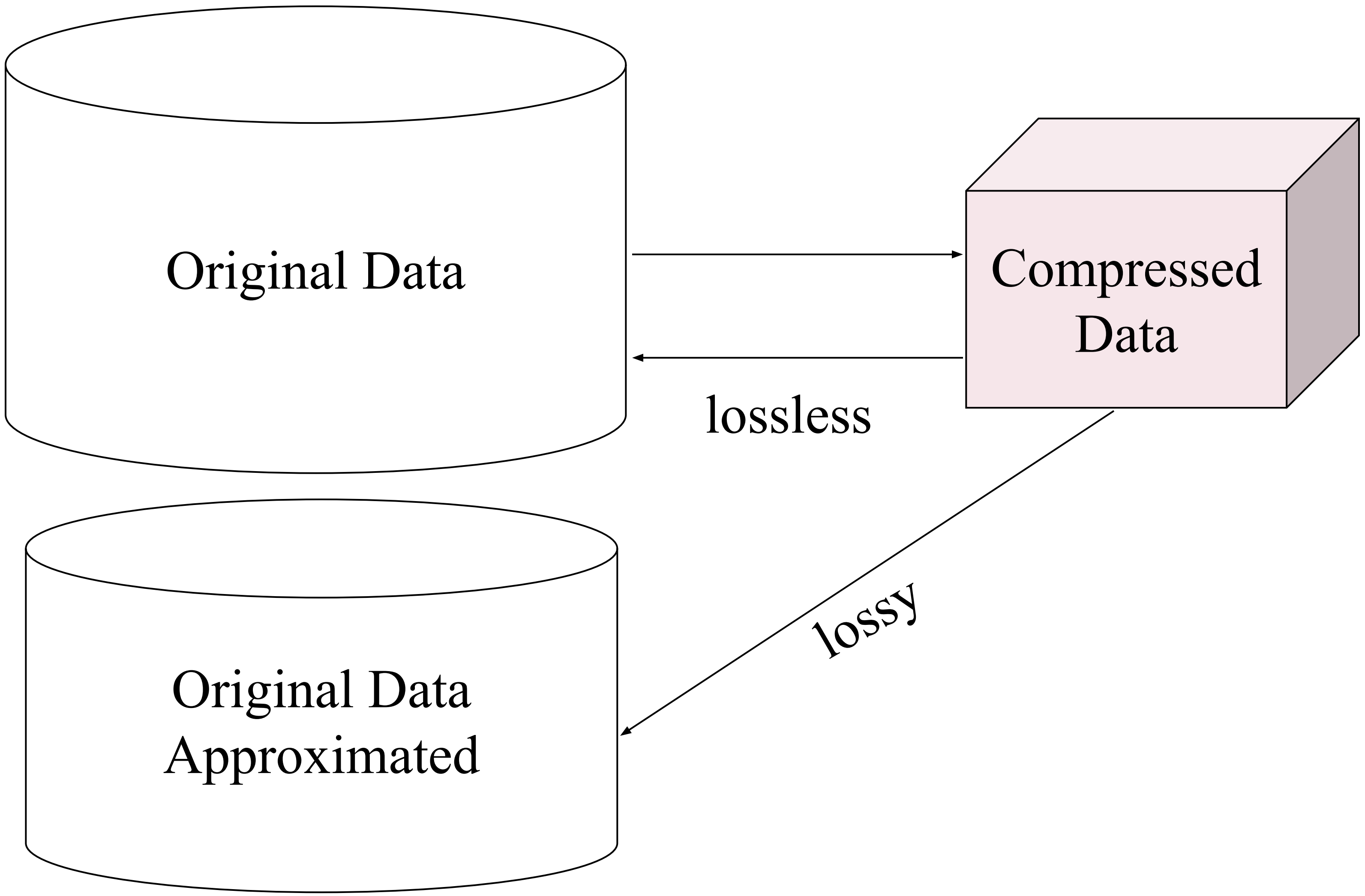
Heuristic Feature Selection Methods

- There are 2^d possible sub-features of *d* features
- Several heuristic feature selection methods:
  - Best single features under the feature independence assumption: choose by significance tests
  - Best step-wise feature selection:
    - The best single-feature is picked first
    - Then next best feature condition to the first, ...
  - Step-wise feature elimination:
    - Repeatedly eliminate the worst feature
  - Best combined feature selection and elimination
  - Optimal branch and bound:
    - Use feature elimination and backtracking

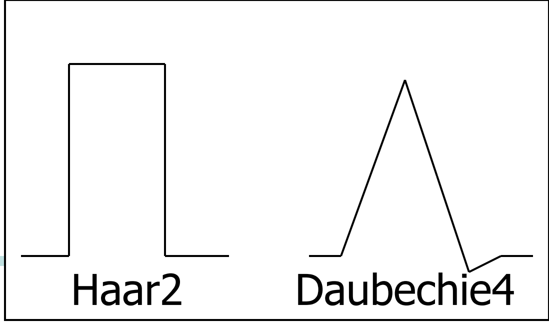
Data Compression

- 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



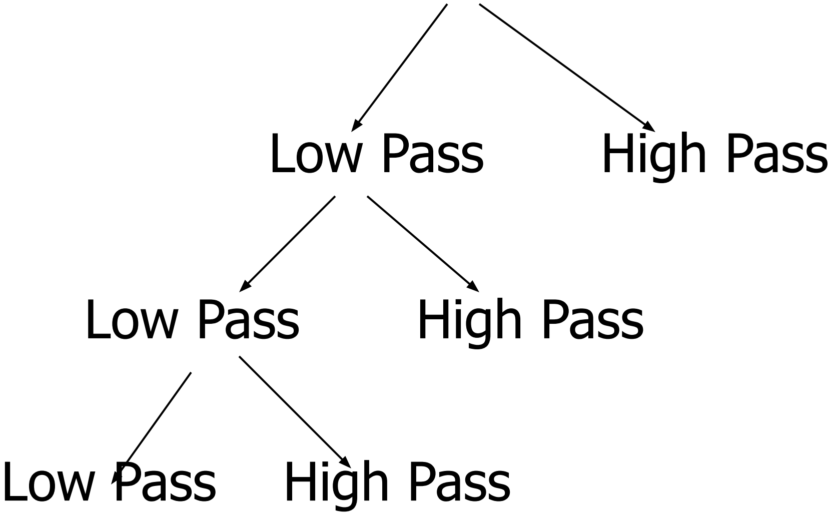
Dimensionality Reduction: Wavelet Transformation



- Discrete wavelet transform (DWT): linear signal processing, multi-resolutional analysis
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients
- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space
- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two set of data of length L/2
  - Applies two functions recursively, until reaches the desired length

DWT for Image Compression

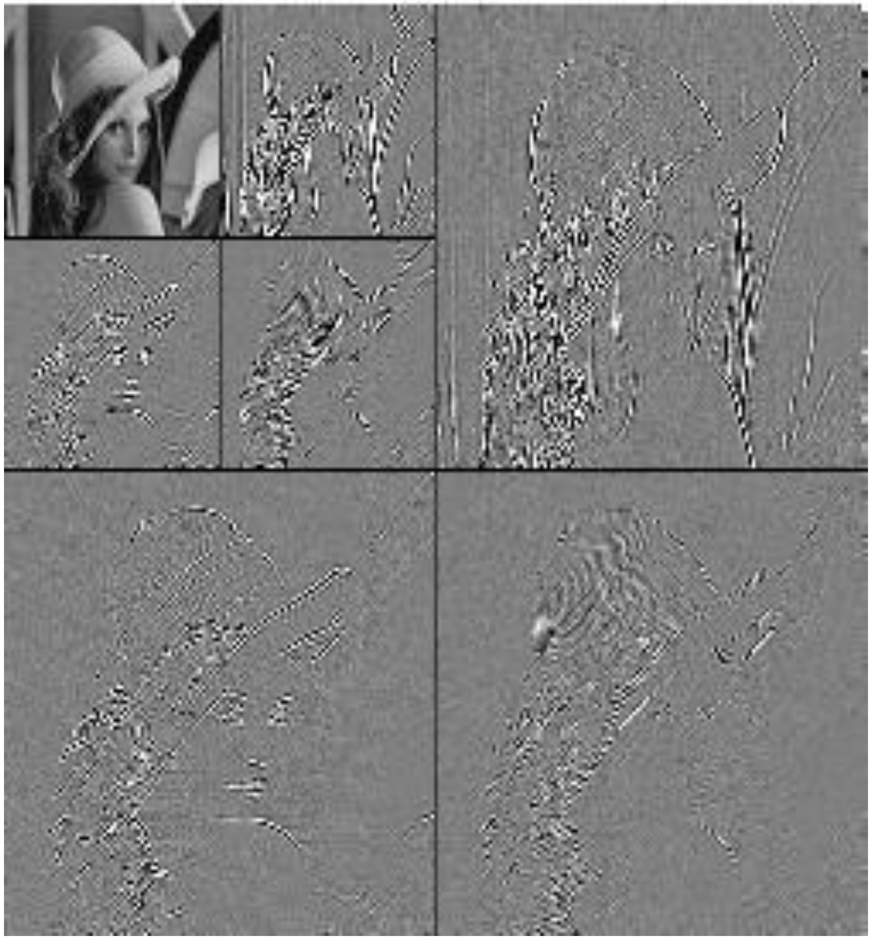
- Image



Low Pass High Pass

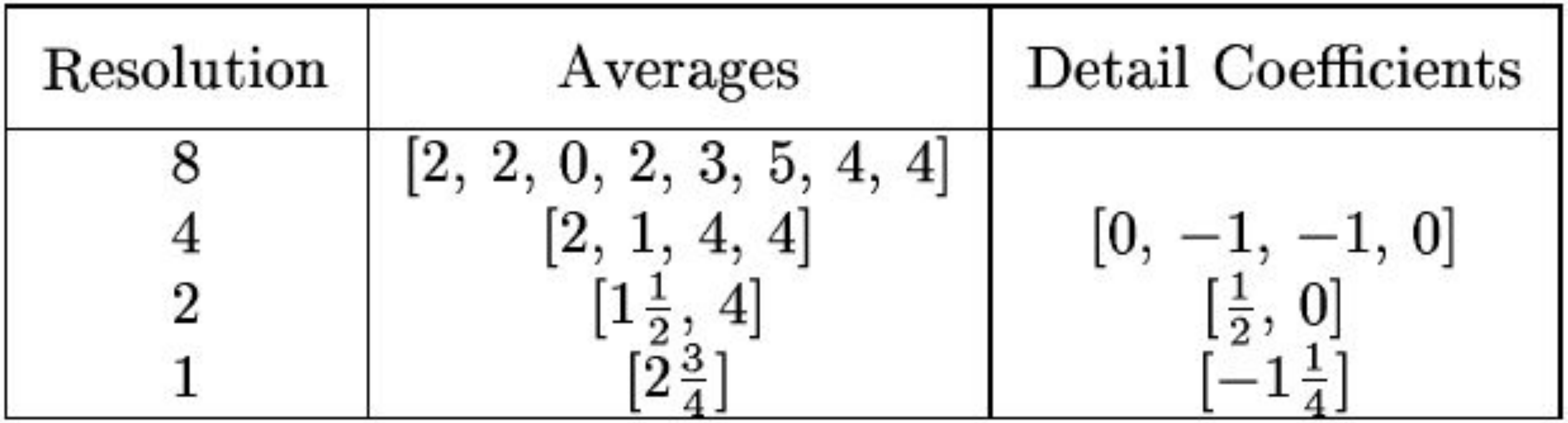
Low Pass High Pass

Low Pass High Pass



Wavelet Decomposition

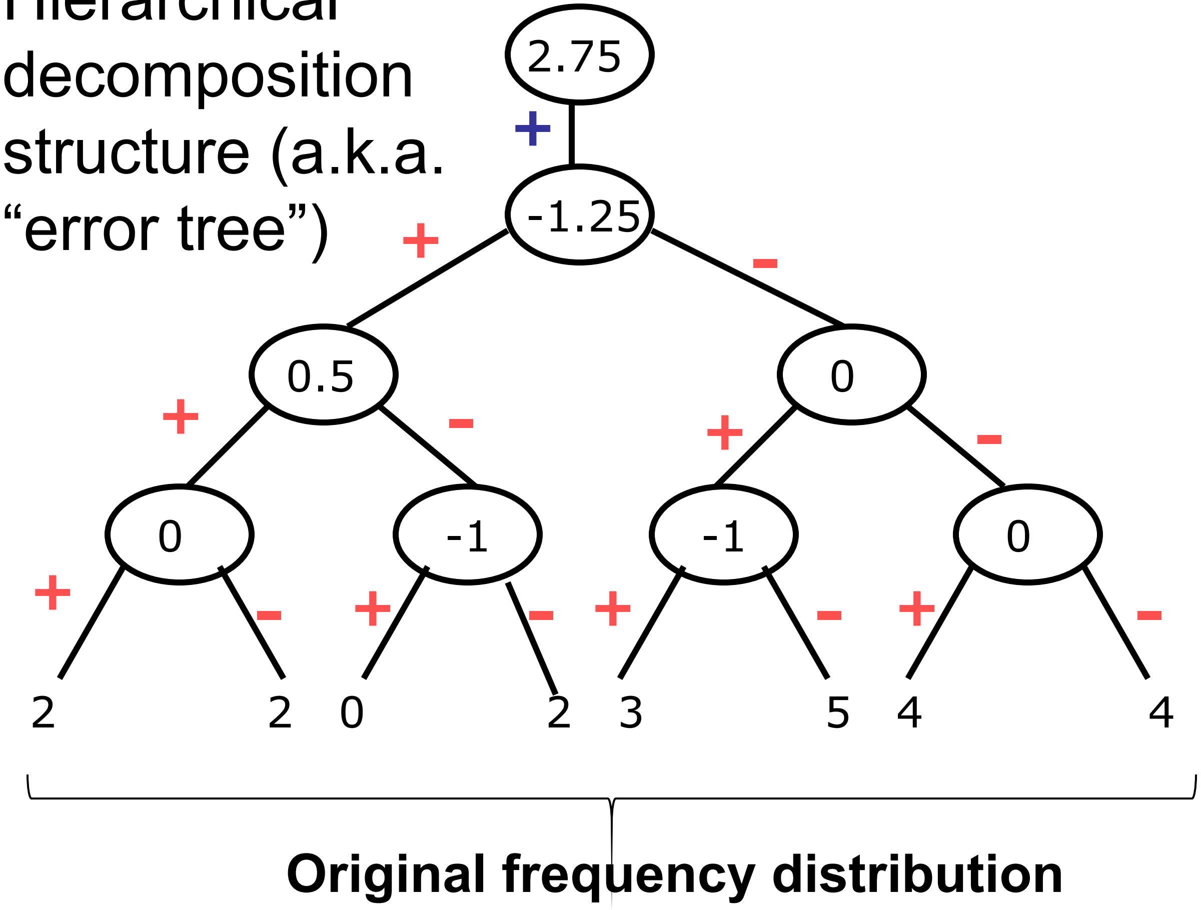
- Wavelets: A math tool for space-efficient hierarchical decomposition of functions
- S = [2, 2, 0, 2, 3, 5, 4, 4] can be transformed to $S_{\wedge}$ = $[2^{3}/_{4}, -1^{1}/_{4}, {}^{1}/_{2}, 0, 0, -1, -1, 0]$
- Compression: many small detail coefficients can be replaced by 0's, and only the significant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

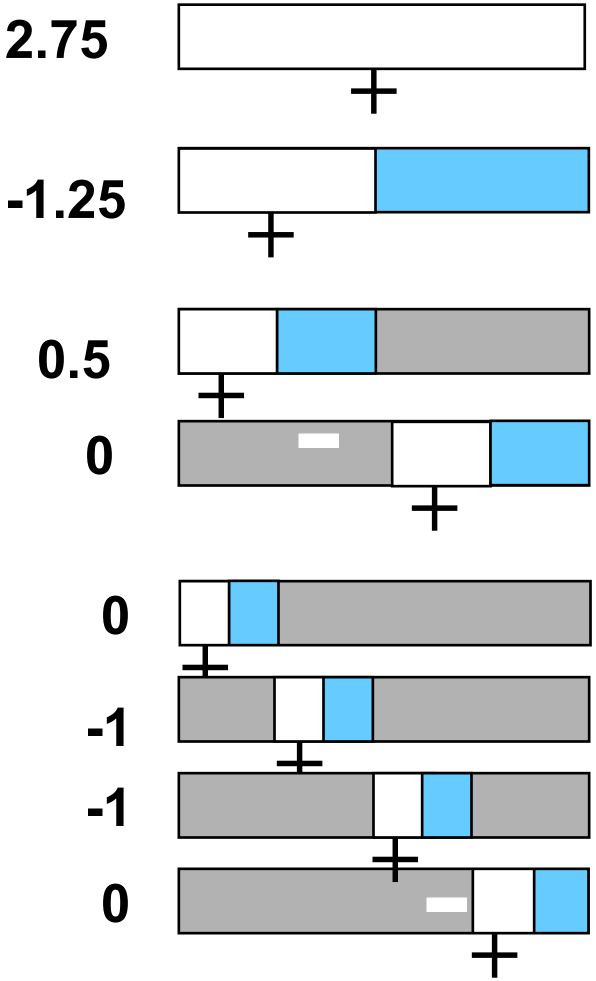
Haar Wavelet Coefficients

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



Original frequency distribution

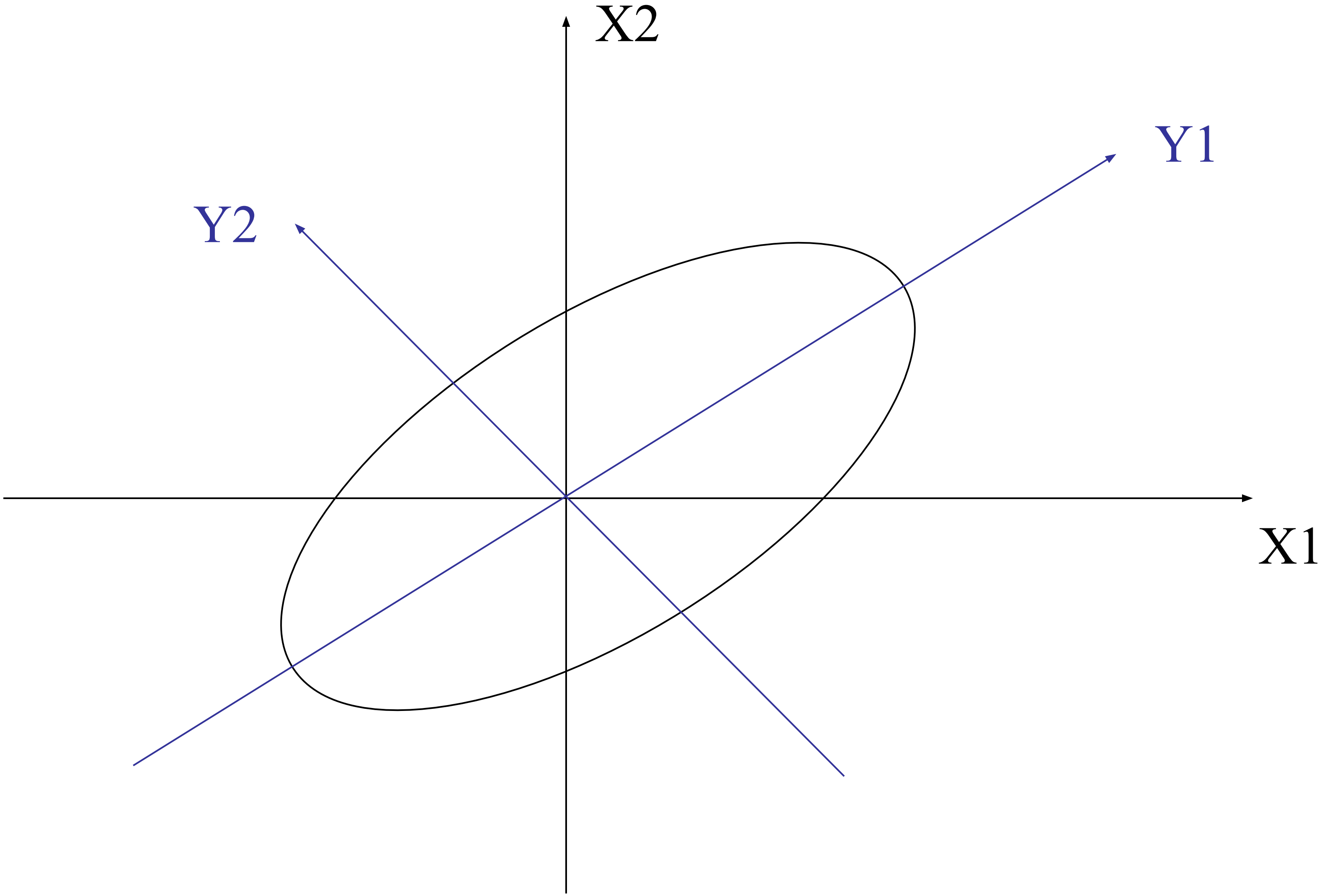
Coefficient “Supports”



Dimensionality Reduction: Principal Component Analysis (PCA)

- Given *N* data vectors from *n*-dimensions, find $k \le n$ orthogonal vectors (*principal components*) that can be best used to represent data
- Steps
  - 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
- Used when the number of dimensions is large

Principal Component Analysis



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

Data Reduction Method (1): Regression and Log-Linear Models

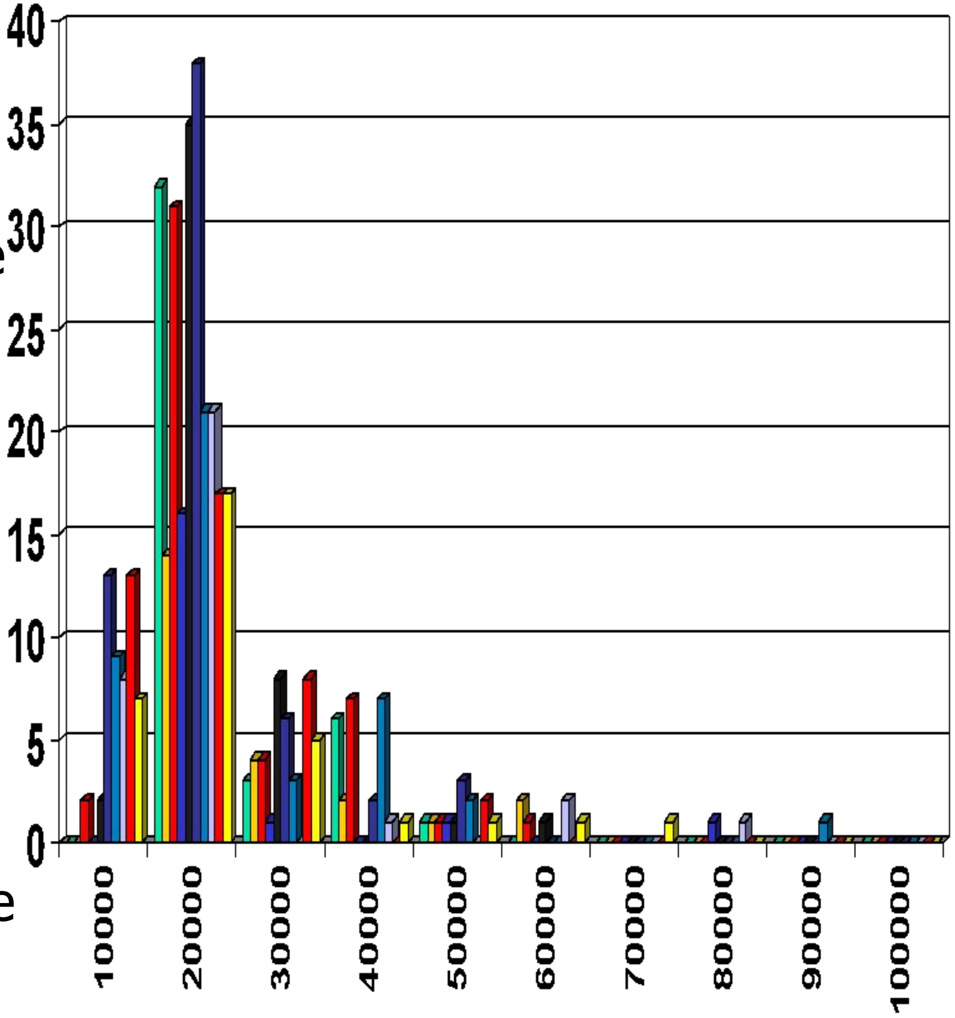
- Linear regression: Data are modeled to fit a straight line
  - Often uses the least-square method to fit the line
- Multiple regression: allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- Log-linear model: approximates discrete multidimensional probability distributions

Regress Analysis and Log-Linear Models

- Linear regression: $Y = w X + b$
  - 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 $Y_1, Y_2, \ldots, X_1, X_2, \ldots$.
- Multiple regression: $Y = b0 + b1\ X1 + b2\ X2.$
  - Many nonlinear functions can be transformed into the above
- Log-linear models:
  - The multi-way table of joint probabilities is approximated by a product of lower-order tables
  - Probability: $p(a, b, c, d) = \alpha_{ab}\ \beta_{ac} \chi_{ad}\ \delta_{bcd}$

Data Reduction Method (2): Histograms

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)
  - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)
  - MaxDiff: set bucket boundary between each pair for pairs have the β–1 largest differences



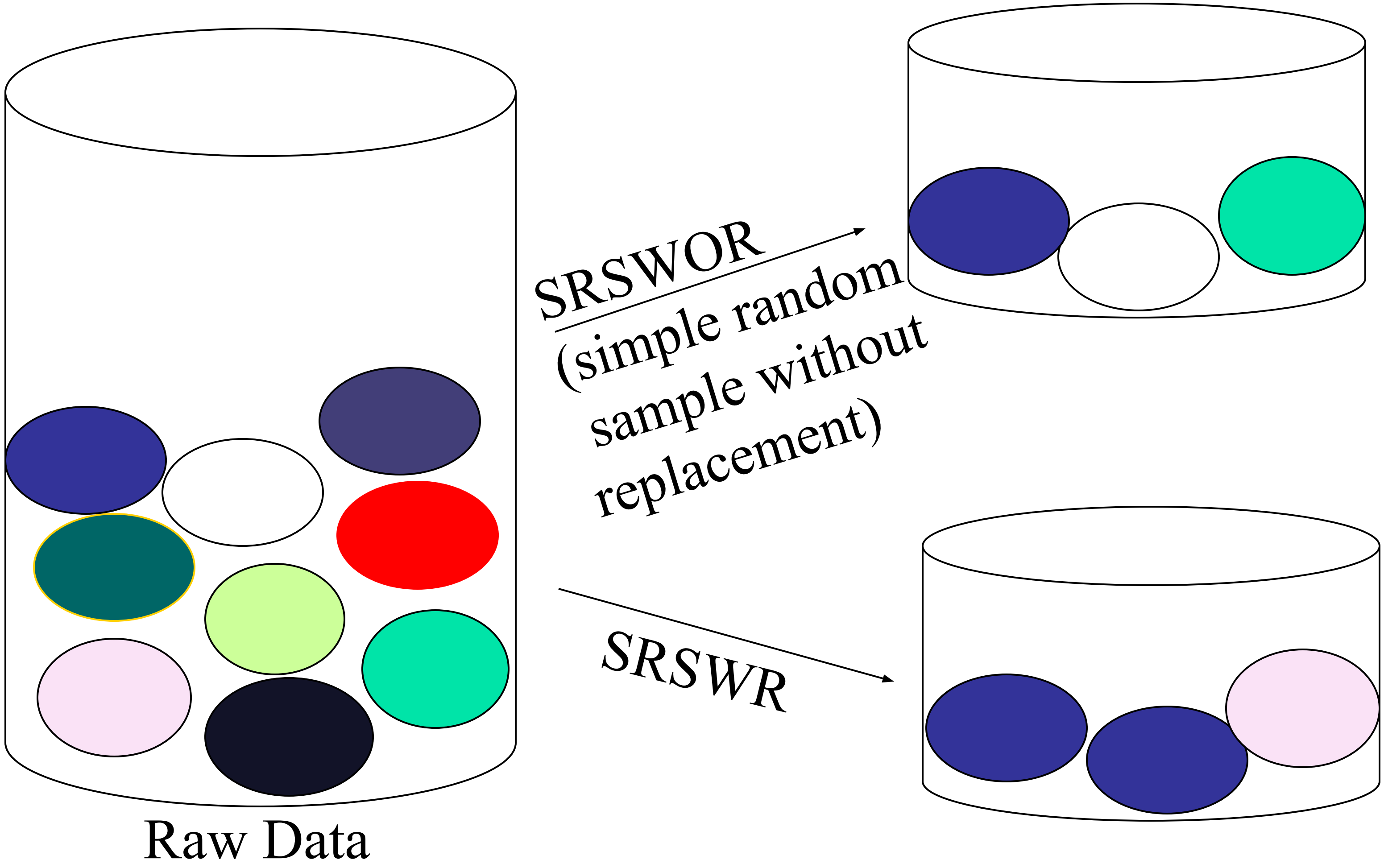
Data Reduction Method (3): Clustering

- 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
- Cluster analysis will be studied in depth in Chapter 7

Data Reduction Method (4): Sampling

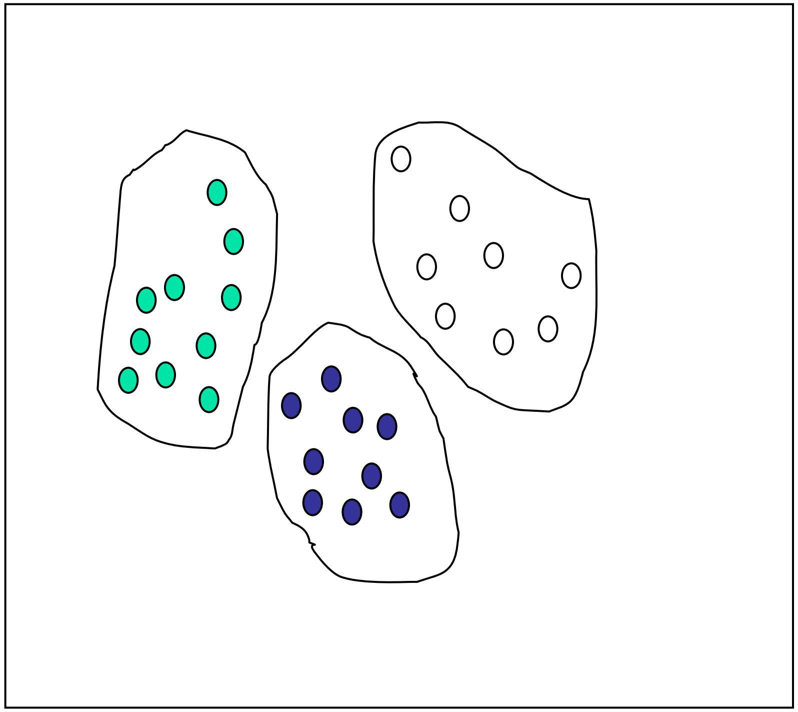
- Sampling: obtaining a small sample *s* to represent the whole data set *N*
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Choose a representative subset of the data
  - Simple random sampling may have very poor performance in the presence of skew
- Develop adaptive sampling methods
  - Stratified sampling:
    - Approximate the percentage of each class (or subpopulation of interest) in the overall database
    - Used in conjunction with skewed data
- Note: Sampling may not reduce database I/Os (page at a time)

Sampling: with or without Replacement

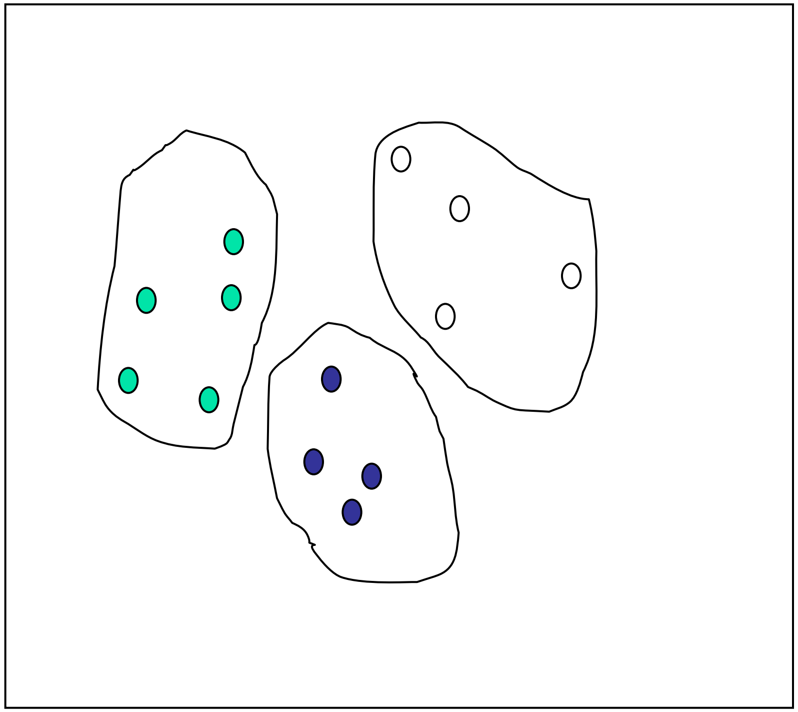


Sampling: Cluster or Stratified Sampling

Raw Data



Cluster/Stratified Sample



Data Preprocessing

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

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 and Concept Hierarchy

- Discretization
  - Reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
- Concept hierarchy formation
  - Recursively reduce the data by collecting and replacing low level concepts (such as numeric values for age) by higher level concepts (such as young, middle-aged, or senior)

Discretization and Concept Hierarchy Generation for Numeric Data

- Typical methods: All the methods can be applied recursively
  - Binning (covered above)
    - Top-down split, unsupervised,
  - Histogram analysis (covered above)
    - Top-down split, unsupervised
  - Clustering analysis (covered above)
    - Either top-down split or bottom-up merge, unsupervised
  - Entropy-based discretization: supervised, top-down split
  - Interval merging by χ^2 Analysis: unsupervised, bottom-up merge
  - Segmentation by natural partitioning: top-down split, unsupervised

Entropy-Based Discretization

- Given a set of samples S, if S is partitioned into two intervals S_1 and S_2 using boundary T, the information gain after partitioning is

$$I(S,T) = \frac{|S_1|}{|S|} Entropy(S_1) + \frac{|S_2|}{|S|} Entropy(S_2)$$

- Entropy is calculated based on class distribution of the samples in the set. Given *m* classes, the entropy of S_1 is

$$Entropy(S_1) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

  where p_i is the probability of class *i* in S_1

- The boundary that minimizes the entropy function over all possible boundaries is selected as a binary discretization
- The process is recursively applied to partitions obtained until some stopping criterion is met
- Such a boundary may reduce data size and improve classification accuracy

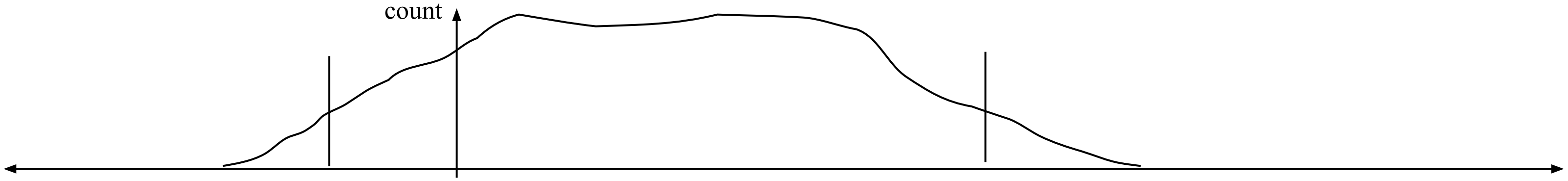
Interval Merge by χ^2 Analysis

- Merging-based (bottom-up) vs. splitting-based methods
- Merge: Find the best neighboring intervals and merge them to form larger intervals recursively
- ChiMerge [Kerber AAAI 1992, See also Liu et al. DMKD 2002]
  - Initially, each distinct value of a numerical attr. A is considered to be one interval
  - χ^2 tests are performed for every pair of adjacent intervals
  - Adjacent intervals with the least χ^2 values are merged together, since low χ^2 values for a pair indicate similar class distributions
  - This merge process proceeds recursively until a predefined stopping criterion is met (such as significance level, max-interval, max inconsistency, etc.)

Segmentation by Natural Partitioning

- A simply 3-4-5 rule can be used to segment numeric data into relatively uniform, “natural” intervals.
  - If an interval covers 3, 6, 7 or 9 distinct values at the most significant digit, partition the range into 3 equi-width intervals
  - If it covers 2, 4, or 8 distinct values at the most significant digit, partition the range into 4 intervals
  - If it covers 1, 5, or 10 distinct values at the most significant digit, partition the range into 5 intervals

Example of 3-4-5 Rule

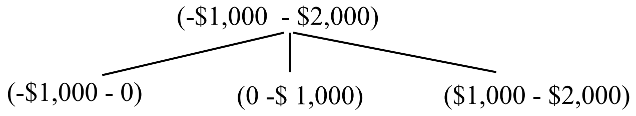


Step 1: -$351 -$159 profit $1,838 $4,700

Min Low (i.e, 5%-tile) High(i.e, 95%-0 tile) Max

Step 2: msd=1,000 Low=-$1,000 High=$2,000

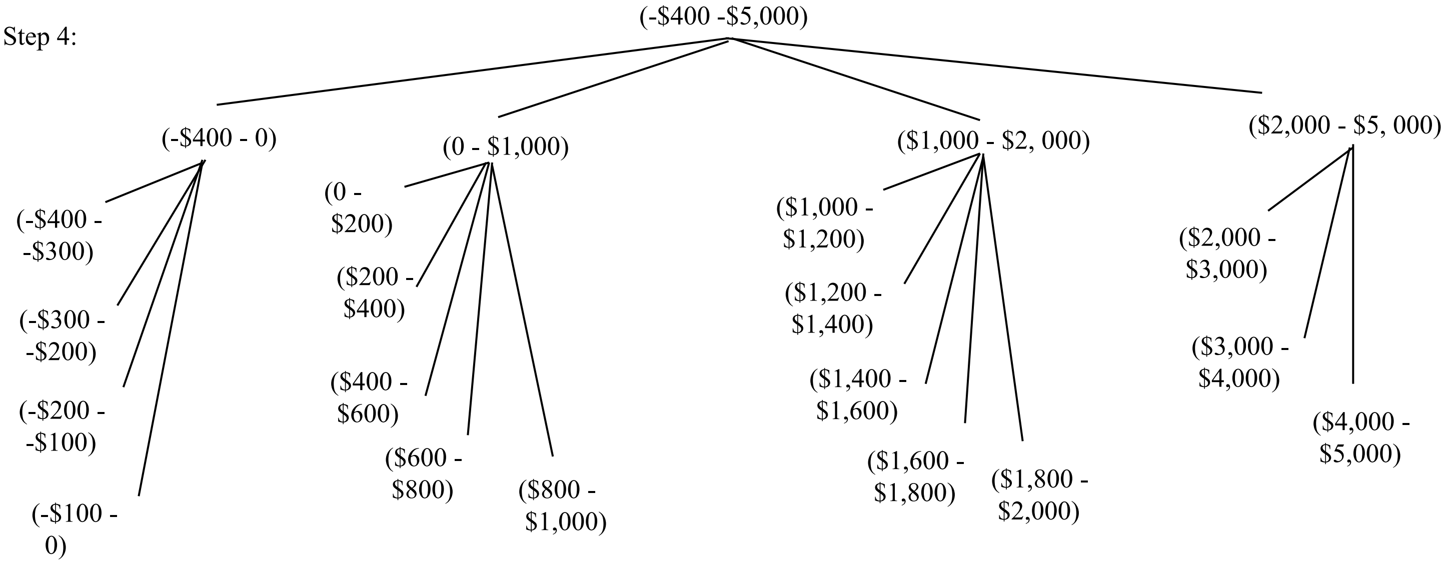
Step 3:



(-$1,000 - $2,000)

(-$1,000 - 0) (0 -$ 1,000) ($1,000 - $2,000)

Step 4:



(-$400 -$5,000)

(-$400 - 0)

(-$400 - -$300)

(-$300 - -$200)

(-$200 - -$100)

(-$100 - 0)

(0 - $1,000)

(0 - $200)

($200 - $400)

($400 - $600)

($600 - $800)

($800 - $1,000)

($1,000 - $2, 000)

($1,000 - $1,200)

($1,200 - $1,400)

($1,400 - $1,600)

($1,600 - $1,800)

($1,800 - $2,000)

($2,000 - $5, 000)

($2,000 - $3,000)

($3,000 - $4,000)

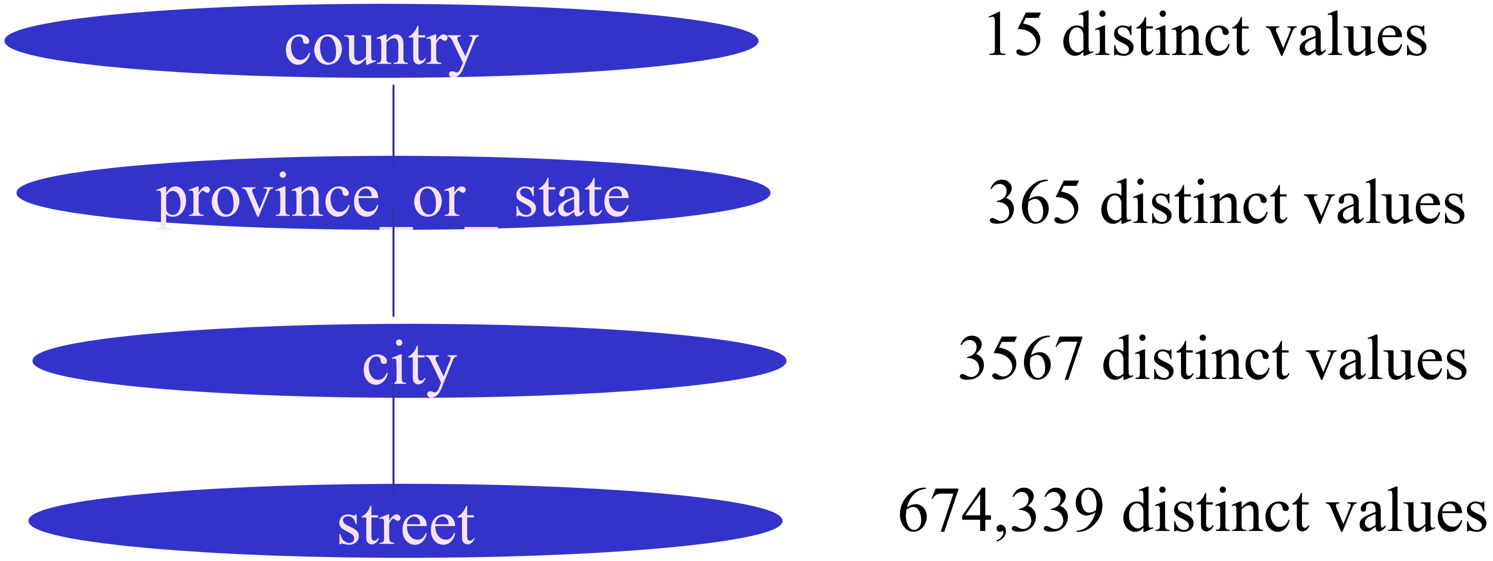
($4,000 - $5,000)

Concept Hierarchy Generation for Categorical Data

- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
  - street < city < state < country
- Specification of a hierarchy for a set of values by explicit data grouping
  - {Urbana, Champaign, Chicago} < Illinois
- Specification of only a partial set of attributes
  - E.g., only street < city, not others
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
  - E.g., for a set of attributes: {street, city, state, country}

Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
  - The attribute with the most distinct values is placed at the lowest level of the hierarchy
  - Exceptions, e.g., weekday, month, quarter, year



Data Preprocessing

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

Summary

- Data preparation or preprocessing is a big issue for both data warehousing and data mining
- Discriptive data summarization is need for quality data preprocessing
- Data preparation includes
  - Data cleaning and data integration
  - Data reduction and feature selection
  - Discretization
- A lot a methods have been developed but data preprocessing still an active area of research

References

- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. Communications of ACM, 42:73-78, 1999
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley & Sons, 2003
- T. Dasu, T. Johnson, S. Muthukrishnan, V. Shkapenyuk. Mining Database Structure; Or, How to Build a Data Quality Browser. SIGMOD'02.
- H.V. Jagadish et al., Special Issue on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering, 20(4), December 1997
- D. Pyle. Data Preparation for Data Mining. Morgan Kaufmann, 1999
- E. Rahm and H. H. Do. Data Cleaning: Problems and Current Approaches. *IEEE Bulletin of the Technical Committee on Data Engineering. Vol.23, No.4*
- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001
- T. Redman. Data Quality: Management and Technology. Bantam Books, 1992
- Y. Wand and R. Wang. Anchoring data quality dimensions ontological foundations. Communications of ACM, 39:86-95, 1996
- R. Wang, V. Storey, and C. Firth. A framework for analysis of data quality research. IEEE Trans. Knowledge and Data Engineering, 7:623-640, 1995

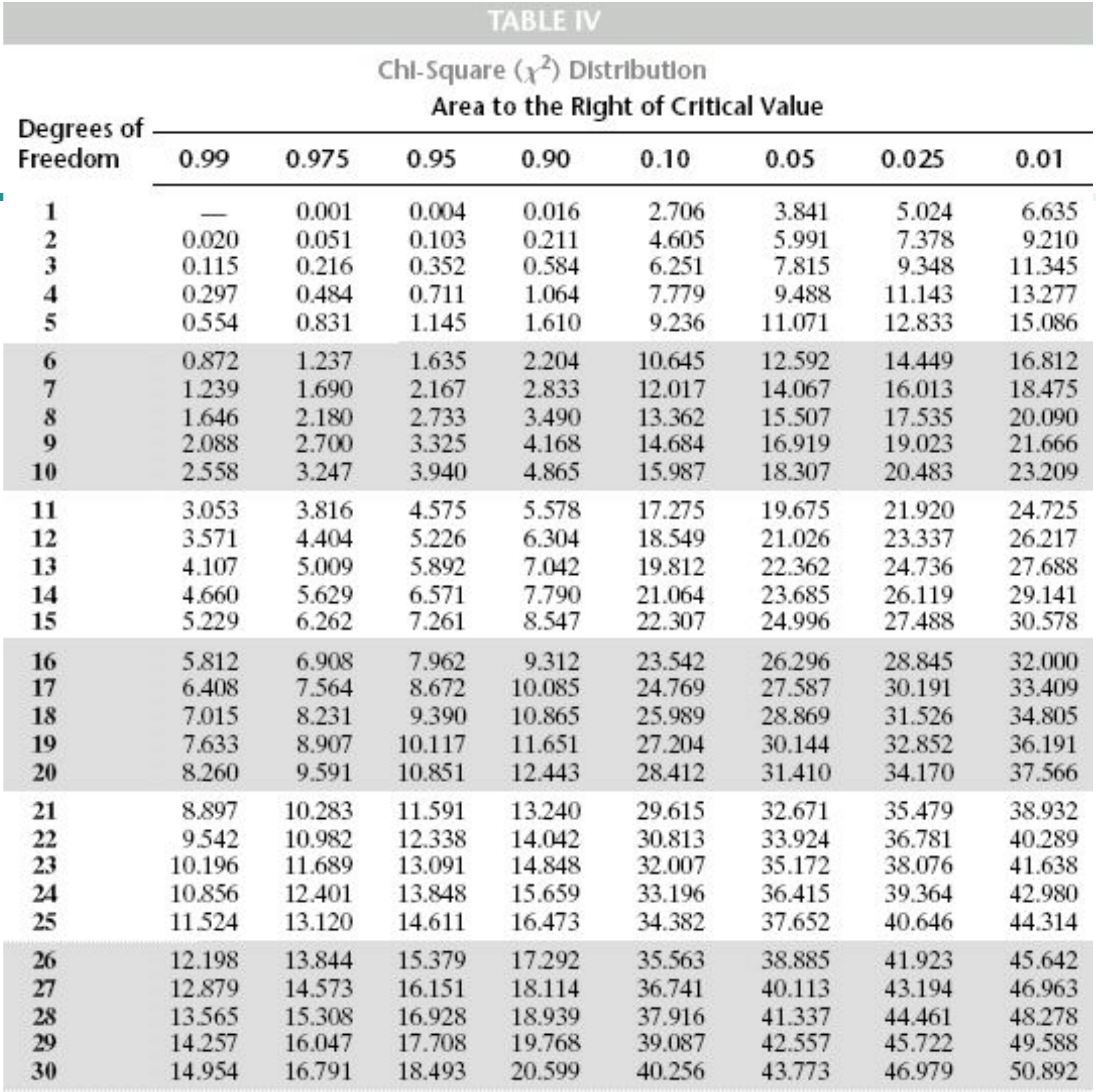


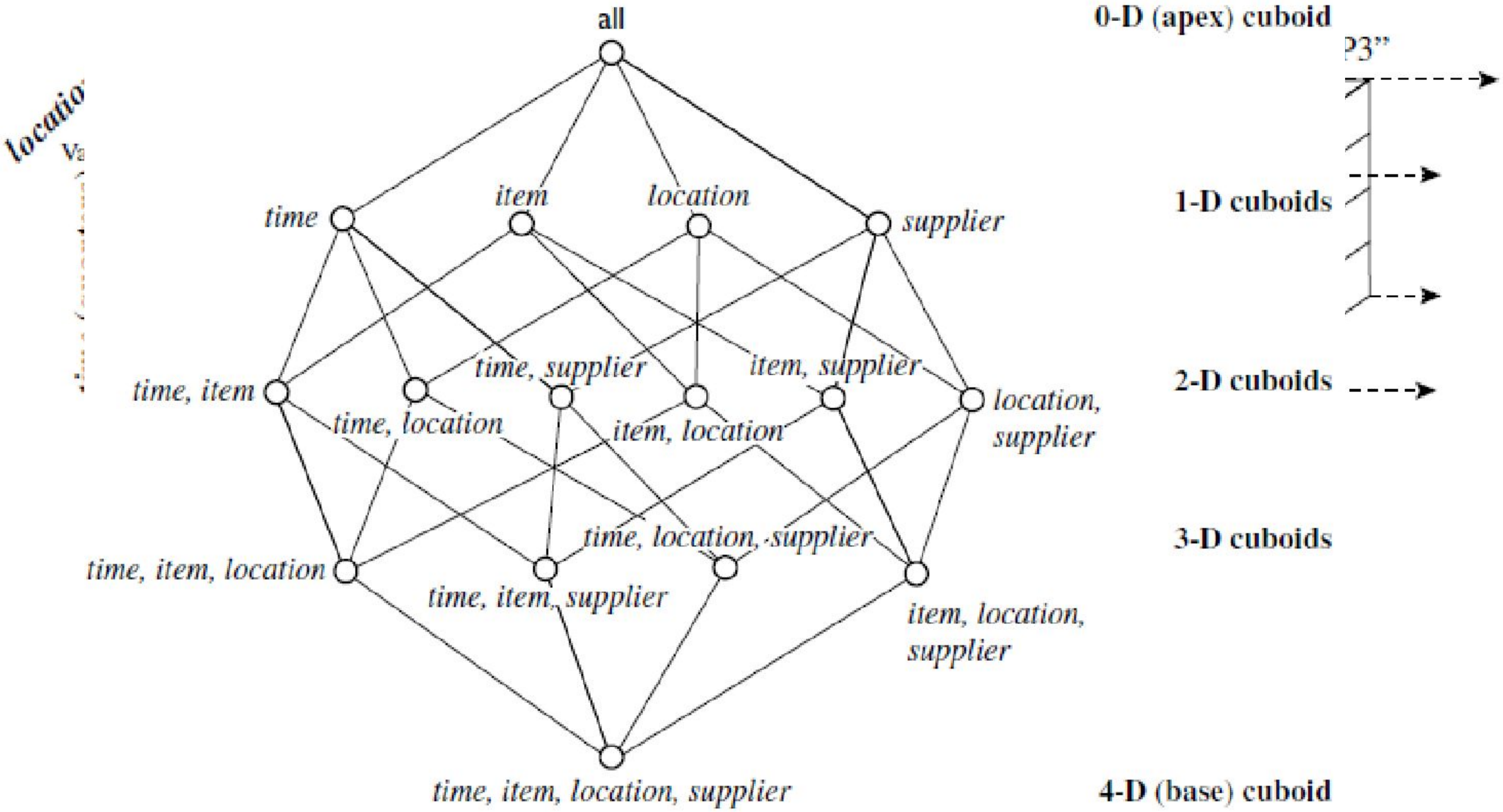
TABLE IV

Chi-Square (χ^2) Distribution

Degrees of Freedom	Area to the Right of Critical Value							
	0.99	0.975	0.95	0.90	0.10	0.05	0.025	0.01
1	—	0.001	0.004	0.016	2.706	3.841	5.024	6.635
2	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210
3	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345
4	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277
5	0.554	0.831	1.145	1.610	9.236	11.071	12.833	15.086
6	0.872	1.237	1.635	2.204	10.645	12.592	14.449	16.812
7	1.239	1.690	2.167	2.833	12.017	14.067	16.013	18.475
8	1.646	2.180	2.733	3.490	13.362	15.507	17.535	20.090
9	2.088	2.700	3.325	4.168	14.684	16.919	19.023	21.666
10	2.558	3.247	3.940	4.865	15.987	18.307	20.483	23.209
11	3.053	3.816	4.575	5.578	17.275	19.675	21.920	24.725
12	3.571	4.404	5.226	6.304	18.549	21.026	23.337	26.217
13	4.107	5.009	5.892	7.042	19.812	22.362	24.736	27.688
14	4.660	5.629	6.571	7.790	21.064	23.685	26.119	29.141
15	5.229	6.262	7.261	8.547	22.307	24.996	27.488	30.578
16	5.812	6.908	7.962	9.312	23.542	26.296	28.845	32.000
17	6.408	7.564	8.672	10.085	24.769	27.587	30.191	33.409
18	7.015	8.231	9.390	10.865	25.989	28.869	31.526	34.805
19	7.633	8.907	10.117	11.651	27.204	30.144	32.852	36.191
20	8.260	9.591	10.851	12.443	28.412	31.410	34.170	37.566
21	8.897	10.283	11.591	13.240	29.615	32.671	35.479	38.932
22	9.542	10.982	12.338	14.042	30.813	33.924	36.781	40.289
23	10.196	11.689	13.091	14.848	32.007	35.172	38.076	41.638
24	10.856	12.401	13.848	15.659	33.196	36.415	39.364	42.980
25	11.524	13.120	14.611	16.473	34.382	37.652	40.646	44.314
26	12.198	13.844	15.379	17.292	35.563	38.885	41.923	45.642
27	12.879	14.573	16.151	18.114	36.741	40.113	43.194	46.963
28	13.565	15.308	16.928	18.939	37.916	41.337	44.461	48.278
29	14.257	16.047	17.708	19.768	39.087	42.557	45.722	49.588
30	14.954	16.791	18.493	20.599	40.256	43.773	46.979	50.892

Back

4-D data representation
Lattice of cuboids



Back