Lambton College

School of Computer Studies

CBD-3335 Data Mining and Analysis

Who am I?

❑Mohammad Saiful Islam

- PhD (ongoing) at Ryerson (ML & Post-quantum Cryptography)
- MSc in Computer Science at Ryerson (Time series analysis of Cloud components)
- 15 years of experience in Information Technology and Telecommunication Industries (Ericsson & NSN)

❑ Research Interests:

- Research on the applicability of Machine Learning and Deep Learning in the domain of Cryptography and Quantum Computing.
- Explore Recurrent Neural Network approaches to address ‘Time Series Anomaly Detection’ in Cloud Platform.

❑ Research Engagements

- Working as an IBM Centre for Advanced Studies (CAS) researcher from January 2019

How to get in touch?

Send email to Mohammad.Islam@cestarcollege.com

- Expect response within 24 hours

Please

- Include [**CBD-3335**] into the subject line
- Send email from your college account (college policy)

Who are you?

Required Materials

- Data Mining: The Textbook, 1st Edition
  Author: Aggarwal
  ISBN: 978-3319141411
  Publisher: Springer
  Published: April 2015

- R and Data Mining: Examples and Case Studies, 1st Edition
  Author: Zhao
  ISBN: 978-0123969637
  Publisher: Academic Press
  Published: December 2012

Learning Outcomes

- Data sources, data interpretation and methods of relating data to observations.
- Association pattern mining utilizing a variety of algorithms.
- Analyze clusters and outliers using a variety of methods.
- Perform mining for data streams and text data using algorithms.
- Apply data mining techniques for time series, spatial data and discrete sequences.

Review Policies

- Late Assignments/exams

Evaluation

- Evaluation methods

Evaluation Method	Weight
2 Tests - Each test 20%	40%
2 Assignments - Each 15%	30%
6 In class Assignments – Each 5%	30%
Total	**100%**

- Marking scheme

Mark(%)	Grade	Mark(%)	Grade
94-100	**A+**	67-69	C+
87-93	A	63-66	C
80-86	A-	60-62	C-
77-79	B+	50-59	D
73-76	B	0-49	F
70-72	B-		

Any questions so far?

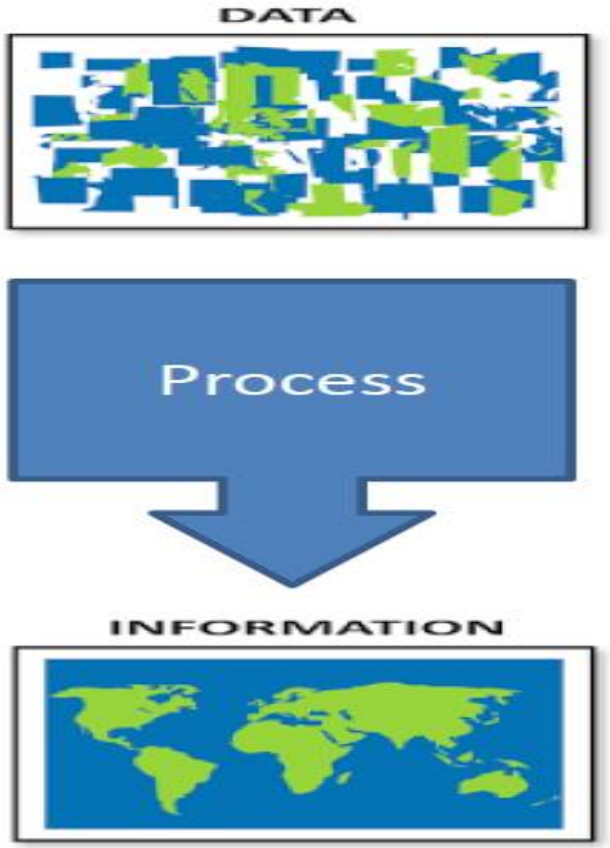
Any comments?

What is Data Mining?

Generally, data mining (sometimes called data or knowledge discovery) is the **process** of analyzing **data** from different perspectives and summarizing it into useful **information** - information that can be used to increase revenue, cuts costs, or both.




What is Data Mining?



Which one is a data mining task?

- Look up phone number in phone directory 
- Group together similar documents returned by search engine according to their context 
- Query a Web search engine for information about “Amazon” 

What is Data?

- 100117
  - First day of our course (10/01/17)
  - Average salary of data scientist (100,117)
  - Zip code of a neighborhood in SF

There is story behind every data. Story could be structure, semantic, relations, and so on.

Information from Data

- Summarizing the data
- Averaging the data
- Selecting part of the data
- Graphing the data
- Adding context
- Adding value
- Relationship between data

Data vs. Information

	Data	Information
Meaning	Data is **raw**, **unorganized facts** that need to be processed.	When data is **processed**, **organized**, **structured** or presented in a given context is called information.
Example	Each student's test score is one piece of data.	The average score of a class or of the entire school is information.

Quiz

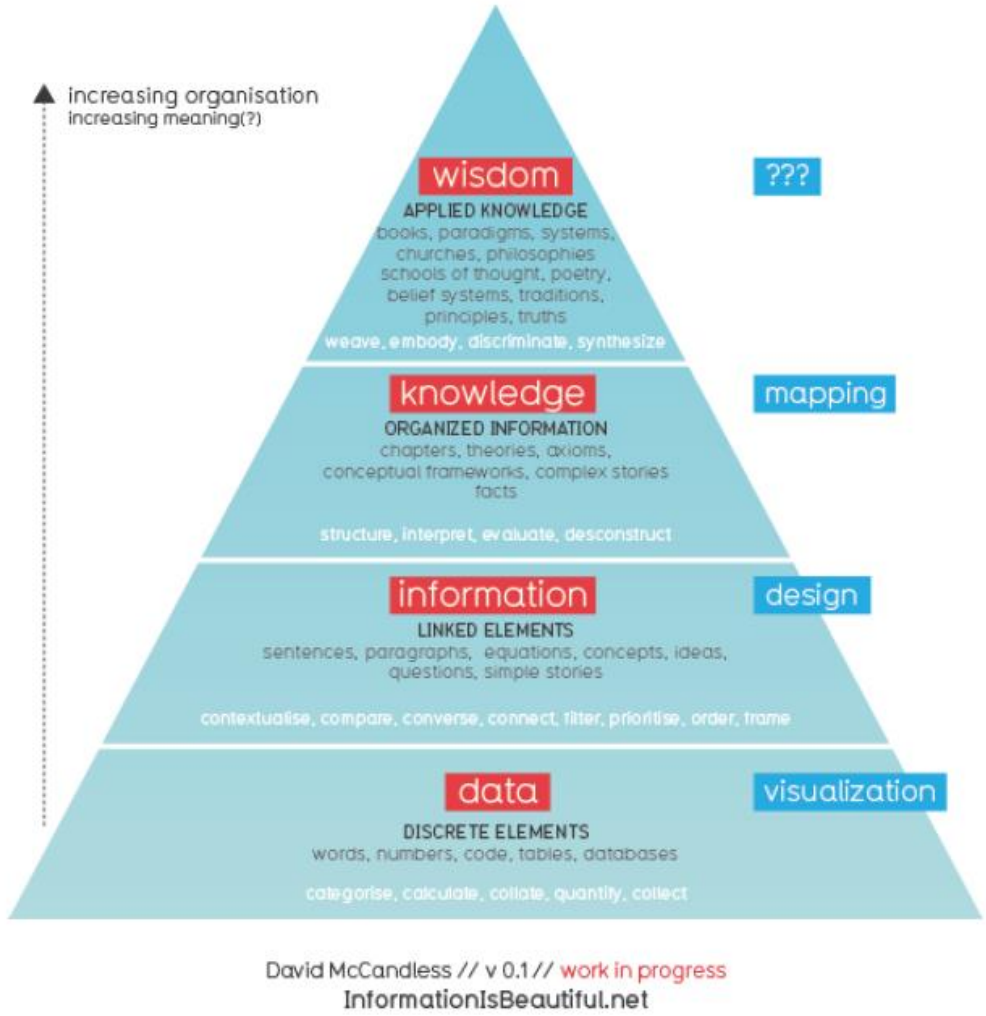
Which of the following is Information?

- Winning time for a race
- Transcriptionist accuracy
- Allergies
- Date of birth

Knowledge from Information

- How is the information tied to outcomes?
- Are there any patterns in the information?
- What information is relevant to the problem?
- How does this information affect the system?
- What is the best way to use the information?
- How can we add more value to the information?

Data, Information, Knowledge & Wisdom



Data Mining Definition

- Data mining is the process of discovering interesting patterns (or knowledge) from large amount of data.
- The data sources can include
  - Databases and data warehouses
  - Web data, social and traditional media
  - Demographic, financial, political data, purchases at department/grocery stores, Bank/Credit Card transactions
  - Health records, gene expression data, scientific and environmental data
  - Or any data that are streamed into system dynamically.

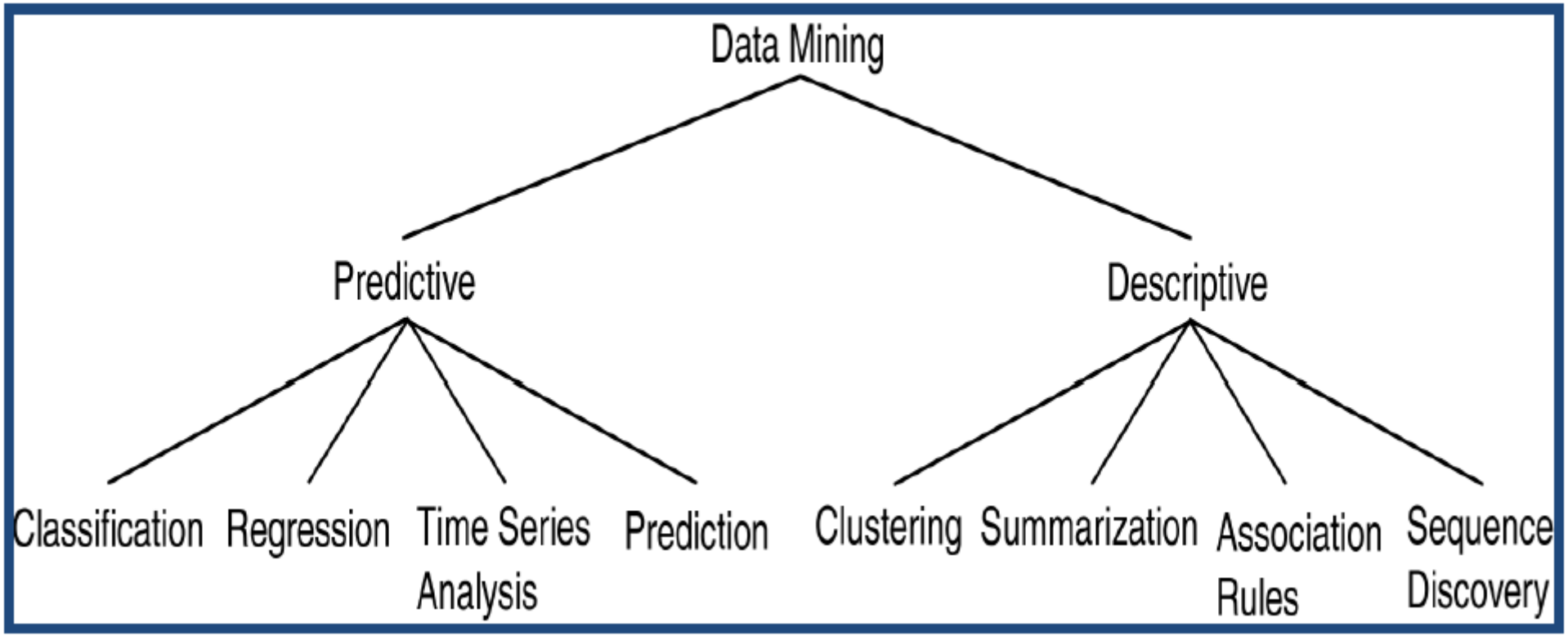
Streaming data mining

- Extracting knowledge structures from continuous, rapid data.
- A data stream is an ordered sequence of instances that in many applications of data stream mining can be read only once or a small number of times using limited computing and storage capabilities.

Data Mining Tasks

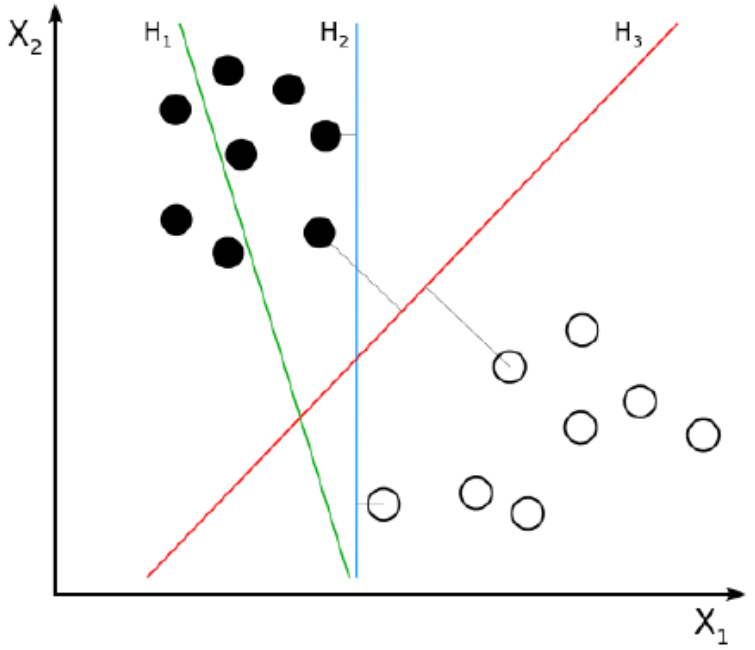
- Data Mining tasks can be classified into two categories:
  - **Descriptive**: Characterize general properties of data in the database
    - Examples: Association rules, frequent patterns, clustering
  - **Predictive**: perform inference on data to make predictions
    - Classification, regression

Data Mining Tasks



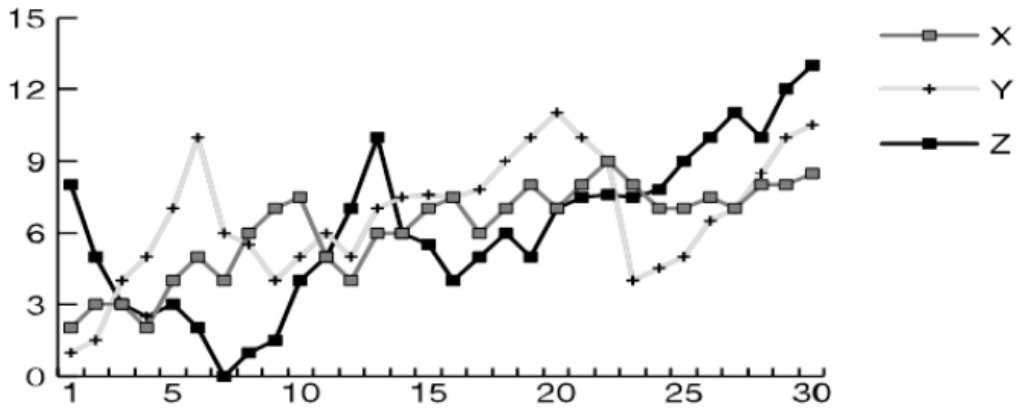
Classification

- Maps data into predefined groups or classes
  - (Semi)supervised learning
  - Pattern recognition
  - Prediction

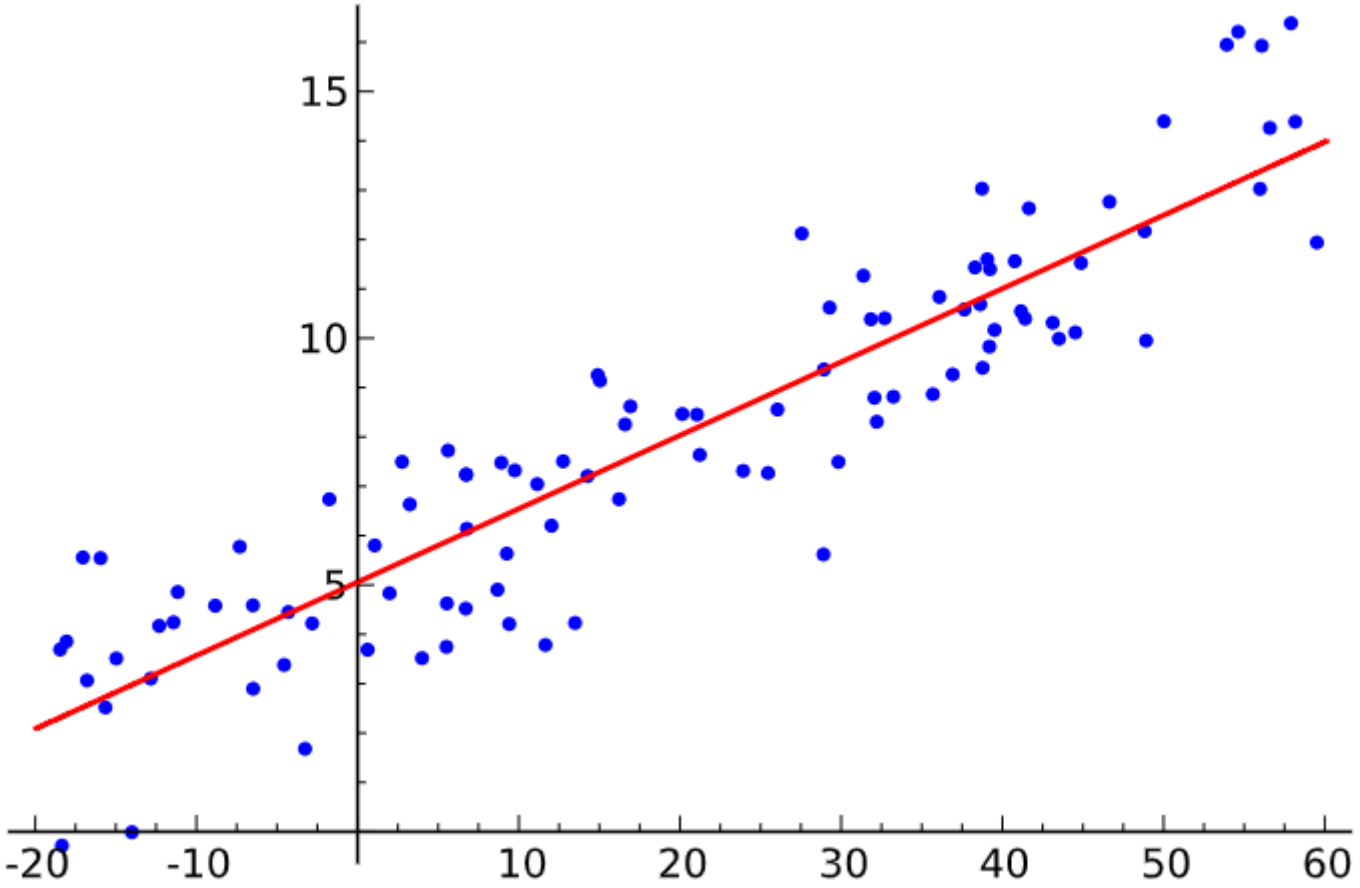


Regression

- Maps a data item to a real valued prediction variable.
- Example: Time series analysis (Stock Market)
  - Predict future values
  - Determine similar patterns over time
  - Classify behavior

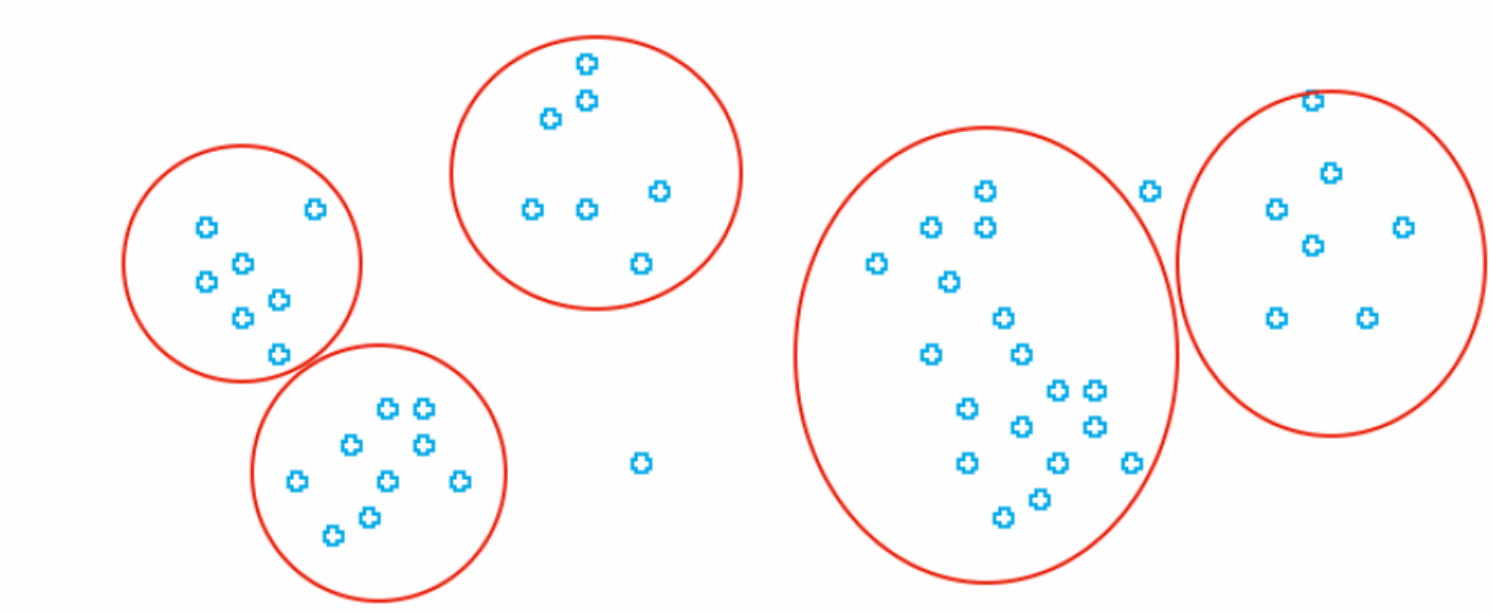


Regression



Clustering

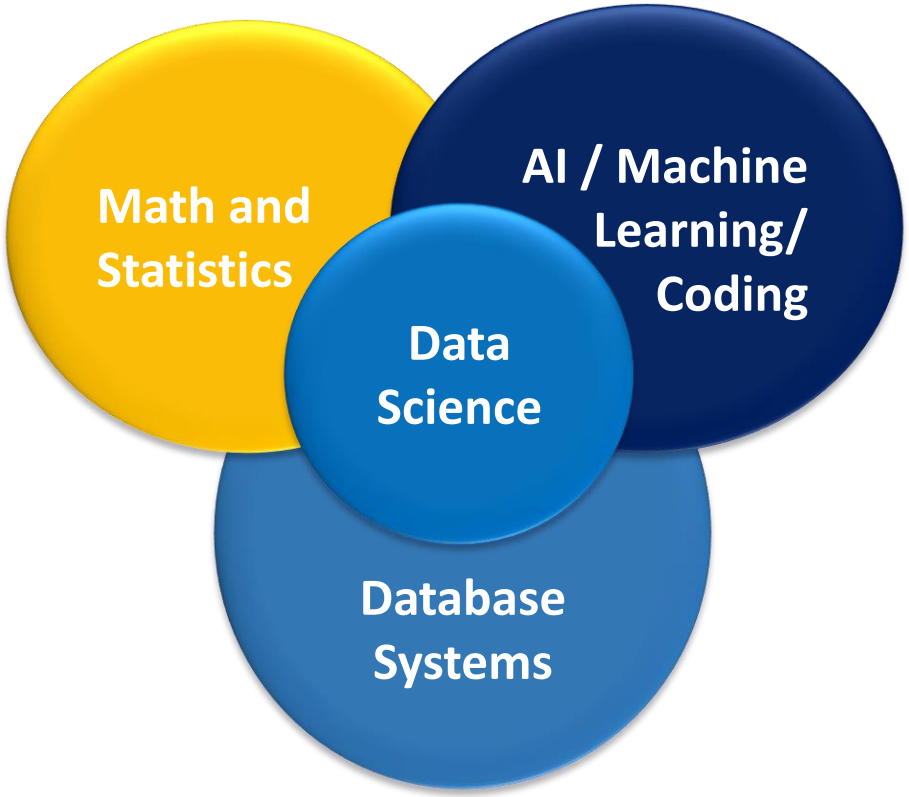
- Groups similar data together into clusters.
  - Unsupervised learning
  - Segmentation
  - Partitioning



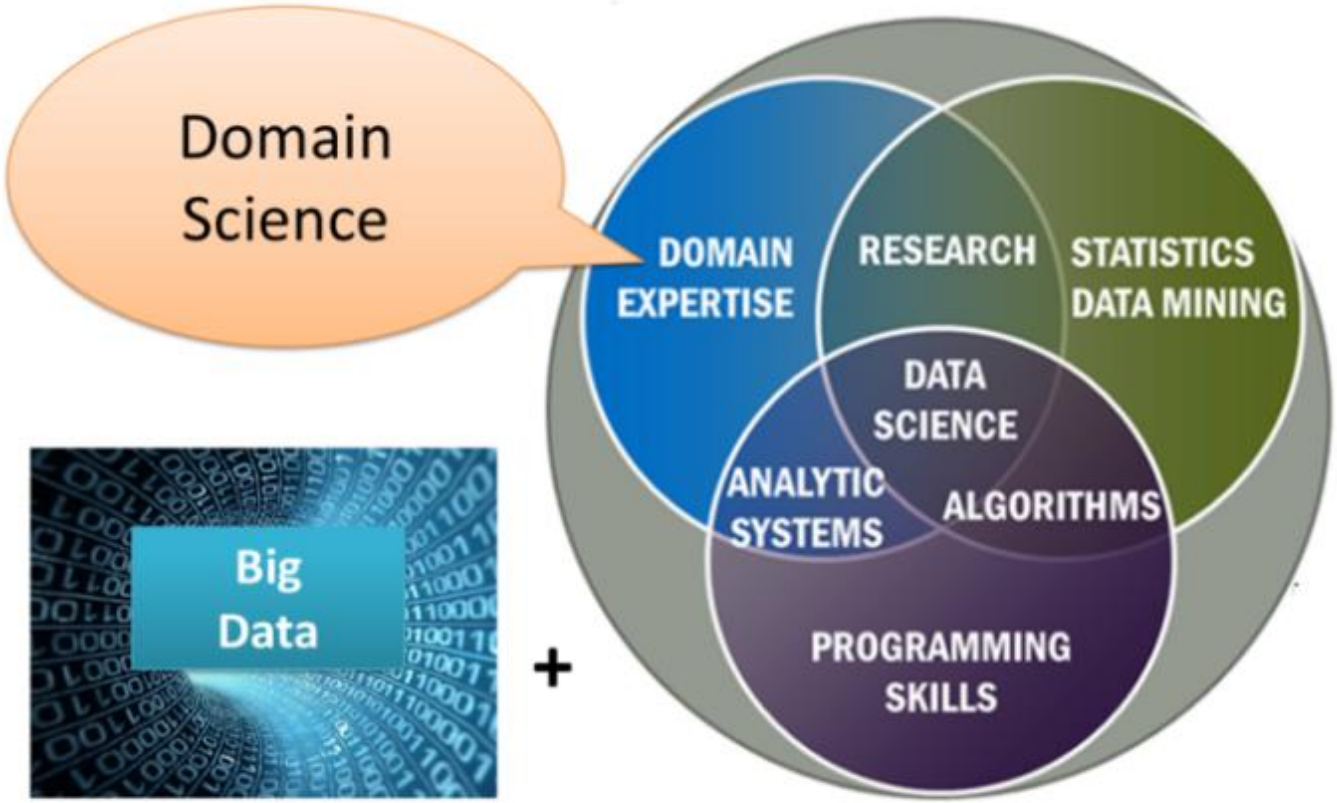
Are all patterns interesting?

- What makes a pattern interesting?
  - Valid — valid: hold on new data with some certainty
  - novel — novel: non-obvious to the system
  - useful — useful: should be possible to act on the item
  - understandable — understandable: humans should be able to interpret the pattern

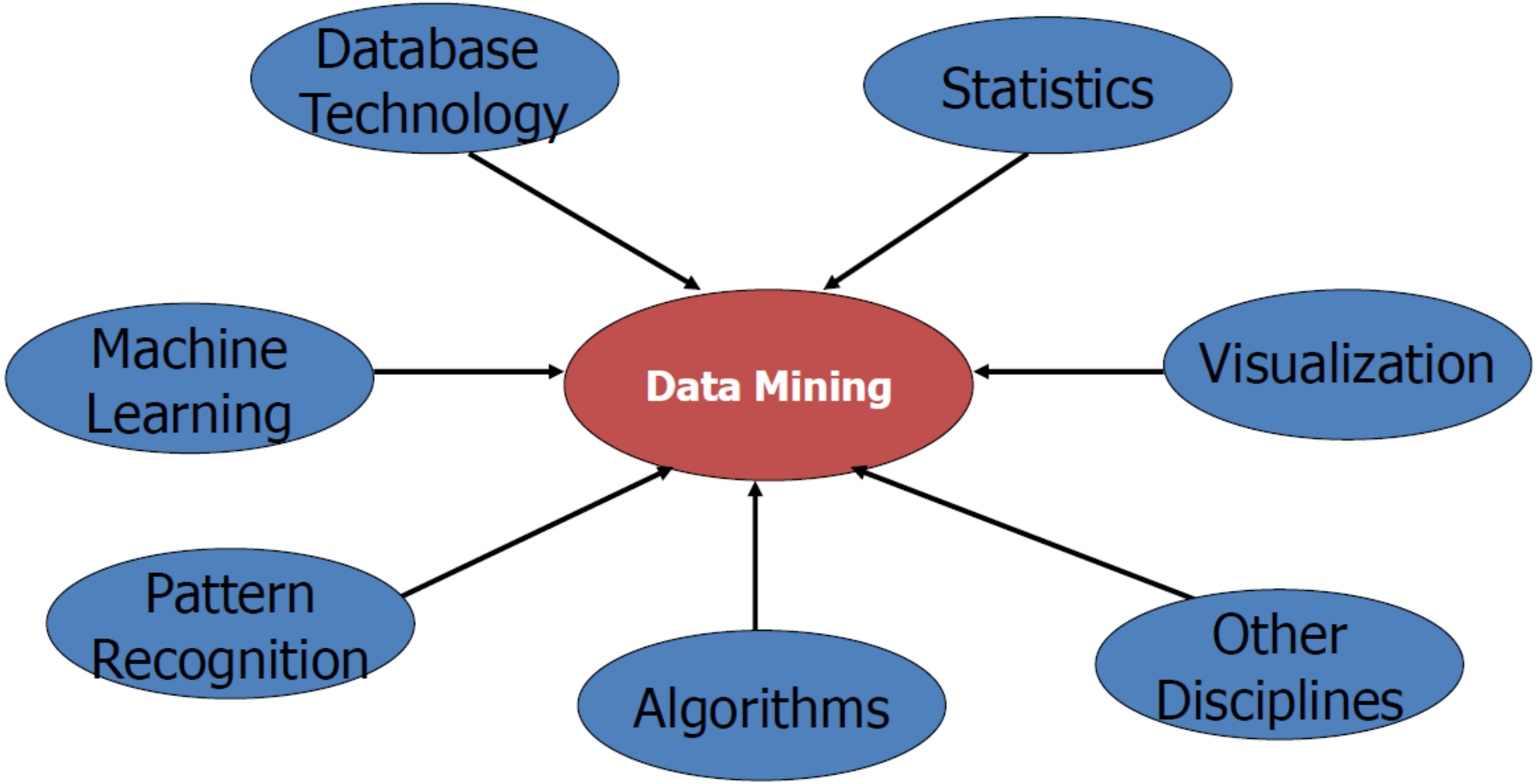
Data Science Concepts



Data Science Concepts



Related Disciplines

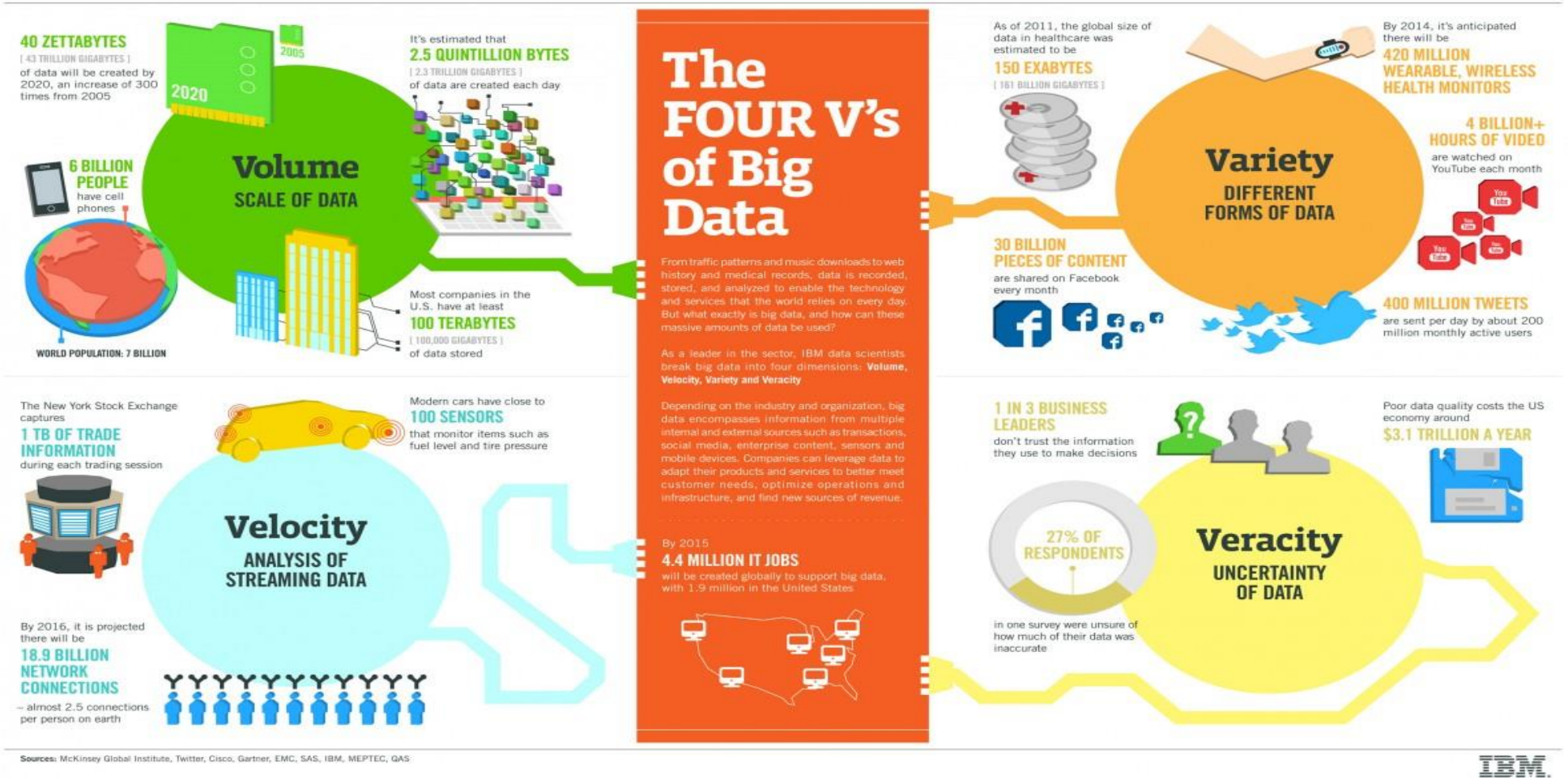


Other Related or Similar Fields

- Data science
- Data analytics
- Artificial intelligence
- Information extraction
- Natural language processing
- Computational linguistics
- Text and web mining
- Search and information retrieval

- Social network analysis
- Graph theory and network science
- Recommender systems
- Link mining

Big Data


40 ZETTABYTES
[43 TRILLION GIGABYTES]
of data will be created by 2020, an increase of 300 times from 2005
2005
2020
6 BILLION PEOPLE
have cell phones
WORLD POPULATION: 7 BILLION
Volume
SCALE OF DATA
It's estimated that
2.5 QUINTILLION BYTES
[2.3 TRILLION GIGABYTES]
of data are created each day
Most companies in the U.S. have at least
100 TERABYTES
[100,000 GIGABYTES]
of data stored
The New York Stock Exchange captures
1 TB OF TRADE INFORMATION
during each trading session
Velocity
ANALYSIS OF STREAMING DATA
Modern cars have close to
100 SENSORS
that monitor items such as fuel level and tire pressure
By 2016, it is projected there will be
18.9 BILLION NETWORK CONNECTIONS
– almost 2.5 connections per person on earth
The FOUR V's of Big Data
From traffic patterns and music downloads to web history and medical records, data is recorded, stored, and analyzed to enable the technology and services that the world relies on every day. But what exactly is big data, and how can these massive amounts of data be used?
As a leader in the sector, IBM data scientists break big data into four dimensions: Volume, Velocity, Variety and Veracity
Depending on the industry and organization, big data encompasses information from multiple internal and external sources such as transactions, social media, enterprise content, sensors and mobile devices. Companies can leverage data to adapt their products and services to better meet customer needs, optimize operations and infrastructure, and find new sources of revenue.
By 2015
4.4 MILLION IT JOBS
will be created globally to support big data, with 1.9 million in the United States
As of 2011, the global size of data in healthcare was estimated to be
150 EXABYTES
[161 BILLION GIGABYTES]
30 BILLION PIECES OF CONTENT
are shared on Facebook every month
Variety
DIFFERENT FORMS OF DATA
By 2014, it's anticipated there will be
420 MILLION WEARABLE, WIRELESS HEALTH MONITORS
4 BILLION+ HOURS OF VIDEO
are watched on YouTube each month
400 MILLION TWEETS
are sent per day by about 200 million monthly active users
1 IN 3 BUSINESS LEADERS
don't trust the information they use to make decisions
27% OF RESPONDENTS
in one survey were unsure of how much of their data was inaccurate
Veracity
UNCERTAINTY OF DATA
Poor data quality costs the US economy around
$3.1 TRILLION A YEAR
Sources: McKinsey Global Institute, Twitter, Cisco, Gartner, EMC, SAS, IBM, MEPTEC, QAS
IBM

Big Data

- A collection of very large and complex data sets very difficult to be handled by conventional database management systems and classical data processing and mining techniques.
- Four characteristics of big data
  - Volume: Constantly increasing data volume
  - Velocity: High speed data in and out (sensors)
  - Variety: Very diverse range of data types and resources
  - Veracity: Big data as honest data
- Every day more than 2.5 quintillion (2.5×10^{18}) bytes of data are created (in 2013)

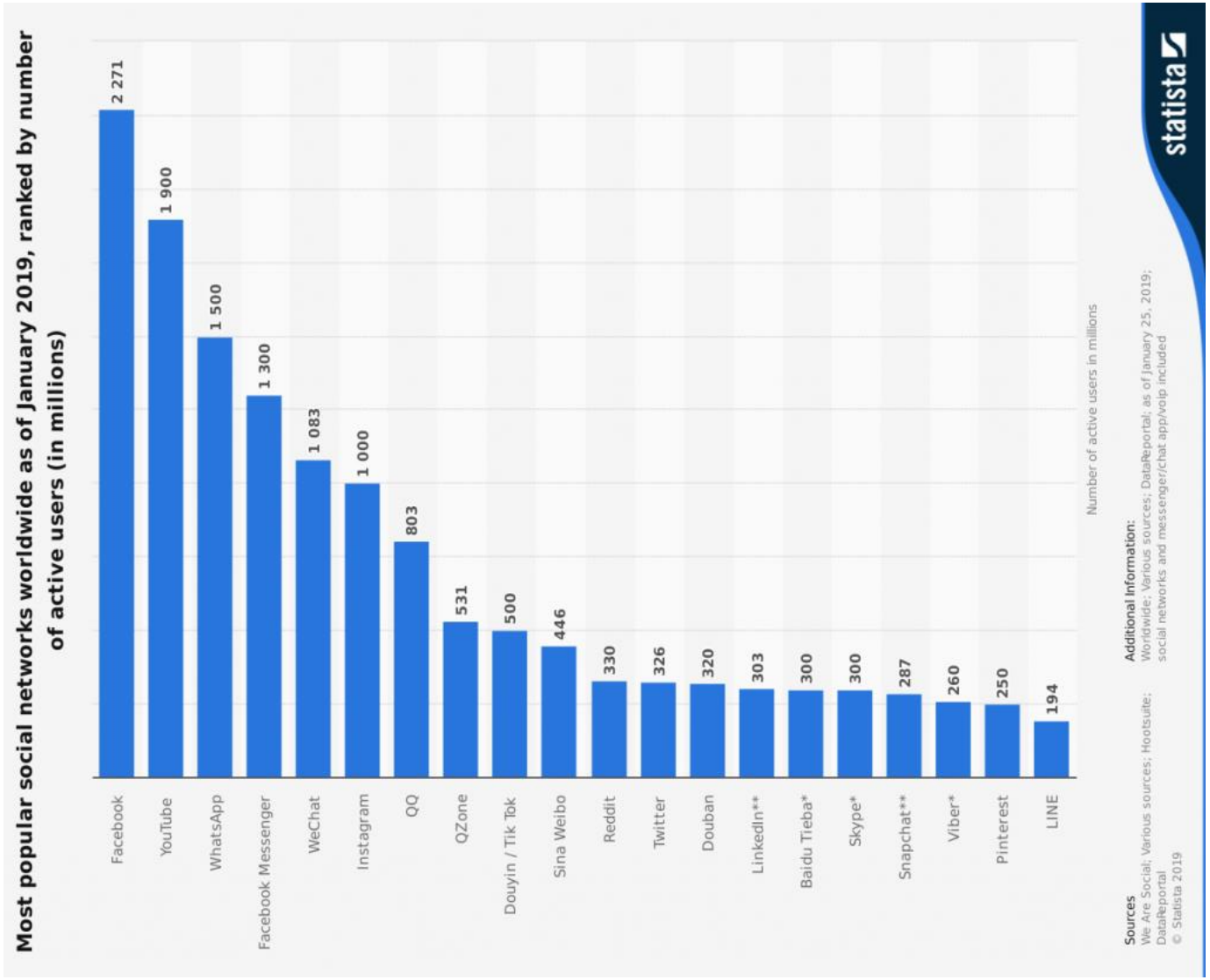
Big Data

- Facebook
  - Daily active users: 968 million
  - Monthly active users: 1.393 billion
  - 751 million mobile users access Facebook every month
- Twitter
  - 44% growth from June 2012 to March 2013
  - 700 million monthly active users
  - 21% of the world's internet population are using Twitter every month (in 2013)
  - Over 500 million registered accounts

Big Data

- YouTube
  - 1 billion unique monthly visitors
  - 6 billion hours of videos are watched every month
- Google+
  - 359 million monthly active users
- LinkedIn
  - Over 200 million users
  - 2 new users join it every second
  - 64% of users are outside the USA

Big Data


Most popular social networks worldwide as of January 2019, ranked by number of active users (in millions)
Facebook 2 271
YouTube 1 900
WhatsApp 1 500
Facebook Messenger 1 300
WeChat 1 083
Instagram 1 000
QQ 803
QZone 531
Douyin / Tik Tok 500
Sina Weibo 446
Reddit 330
Twitter 326
Douban 320
LinkedIn** 303
Baidu Tieba* 300
Skype* 300
Snapchat** 287
Viber* 260
Pinterest 250
LINE 194
Number of active users in millions
Sources
We Are Social; Various sources; Hootsuite; DataReportal
© Statista 2019
Additional Information:
Worldwide; Various sources; DataReportal; as of January 25, 2019; social networks and messenger/chat app/voip included
statista

Big Data

- The Large Hadron Collider (LHC)
  - 150 million sensors delivering data 40 million times per second.
  - There are nearly 600 million collisions per second. After filtering and refraining from recording more than 99.999% of these streams, there are 100 collisions of interest per second (less than 0.001%).
  - The data flow would exceed 150 million petabytes annual rate.

Big Data

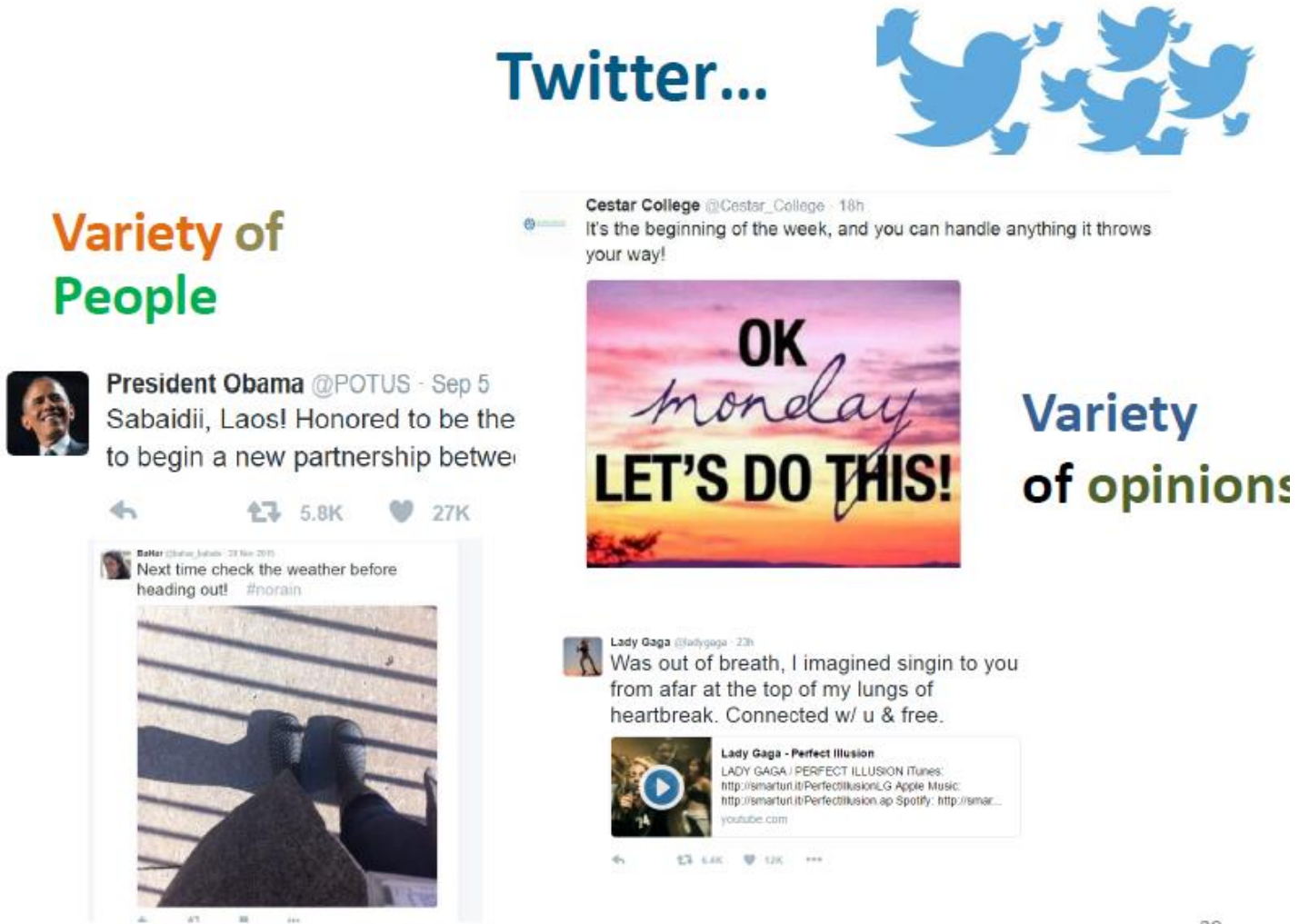
- As a result, only working with less than 0.001% of the sensor stream data, the data flow from all four LHC experiments represents 25 petabytes annual rate before replication (as of 2012). This becomes nearly 200 petabytes after replication.

LHC: https://www.youtube.com/watch?v=bTHzB4h0po4

Big Data

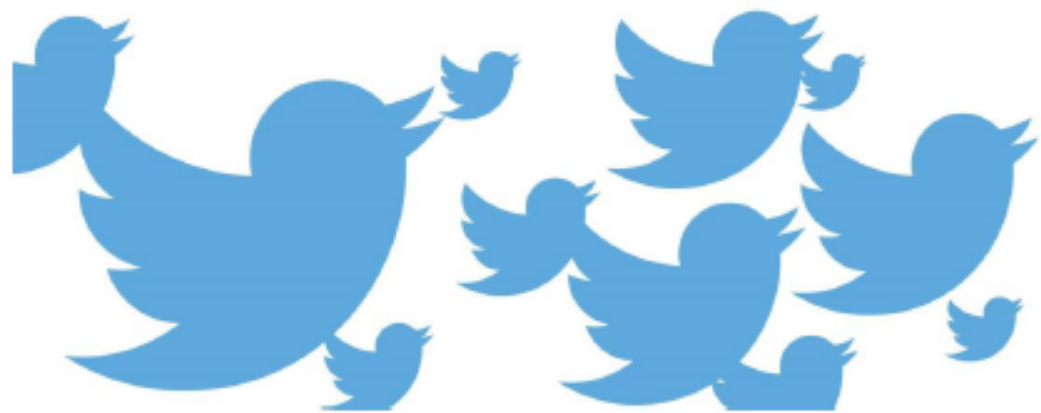
- If all sensor data were to be recorded in LHC, the data flow would be extremely hard to work with. The data flow would exceed 150 million petabytes annual rate, or nearly 500 exabytes (500×10^{18}) per day, before replication. To put the number in perspective, this is equivalent to 500 quanitilion (5×10^{20}) bytes per day, almost 200 times higher than all the other sources combined in the world.

Big Data


Twitter…
Variety of People
President Obama @POTUS · Sep 5
Sabaidii, Laos! Honored to be the
to begin a new partnership betwe
5.8K
27K
Next time check the weather before heading out! #norain
Cestar College @Cestar_College · 18h
It's the beginning of the week, and you can handle anything it throws your way!
OK monday
LET'S DO THIS!
Variety of opinions
Lady Gaga @ladygaga · 23h
Was out of breath, I imagined singin to you from afar at the top of my lungs of heartbreak. Connected w/ u & free.
Lady Gaga - Perfect Illusion
LADY GAGA / PERFECT ILLUSION iTunes:
youtube.com

Twitter is useful…

Stock market
Customer satisfaction
Outbreak of diseases
Crime prediction
Politics
Social science
Legal documents
…



Data Mining Challenges

Scalability
Dimensionality
Complex and Heterogeneous data
Data quality (noise, outliers, missing, gold)
Data ownership
privacy preservation
Streaming data in real-time
Ethical issues

- Scalability
- Dimensionality (curse of dimensionality)
- Complex and Heterogeneous Data
- Data Quality (noise, outliers, missing data, lack of gold or training data)
- Data Ownership and Distribution
- Privacy Preservation
- Streaming Data and real-time data
- Ethical issues

Data mining project approach

Don't be afraid to explore Web. It has all you need to learn

Web-based documentations, Github, Stackoverflow,…

They are your friends and enemies, **You choose**.

Noise vs. Outliers

- Noise is anything that is not the "true" signal.
  - It may have values close to the true signal.

Noise vs. Outliers

- An outlier is something that is much different than the other values.
  - Extreme feature values in one or more dimensions
  - Examples with the same feature values but different labels

- The vast majority of the time outliers are noise but sometimes a data point that is true signal can be an outlier.

Noise vs. Outliers

Example: IQ of our class plus Stephen Hawking.

- Hawking would be an outlier even though we accurately measured his IQ.

- If we measured Stephen's IQ as 90, then that would be noise since his real IQ is much higher than that.

Python for Data Mining

Installing Python and Pycharm

- https://www.python.org/downloads/
- https://www.jetbrains.com/pycharm/download/#section=windows (download the free version)

Installing Python and Pycharm

- Install with pip https://packaging.python.org/installing/
  - –python -m pip install -U pip setuptools(installing pip)
  - –pip install -U pip (updating pip)
  - –Pip install “library_name”

- Install with easy_install
  - –easy_install“library_name”

- Install with wheel (.whlfiles)
  - Go to binaries for python packages http://www.lfd.uci.edu/~gohlke/pythonlibs/
  - Pip install wheel
  - pip install PATH+“library_name”.whl

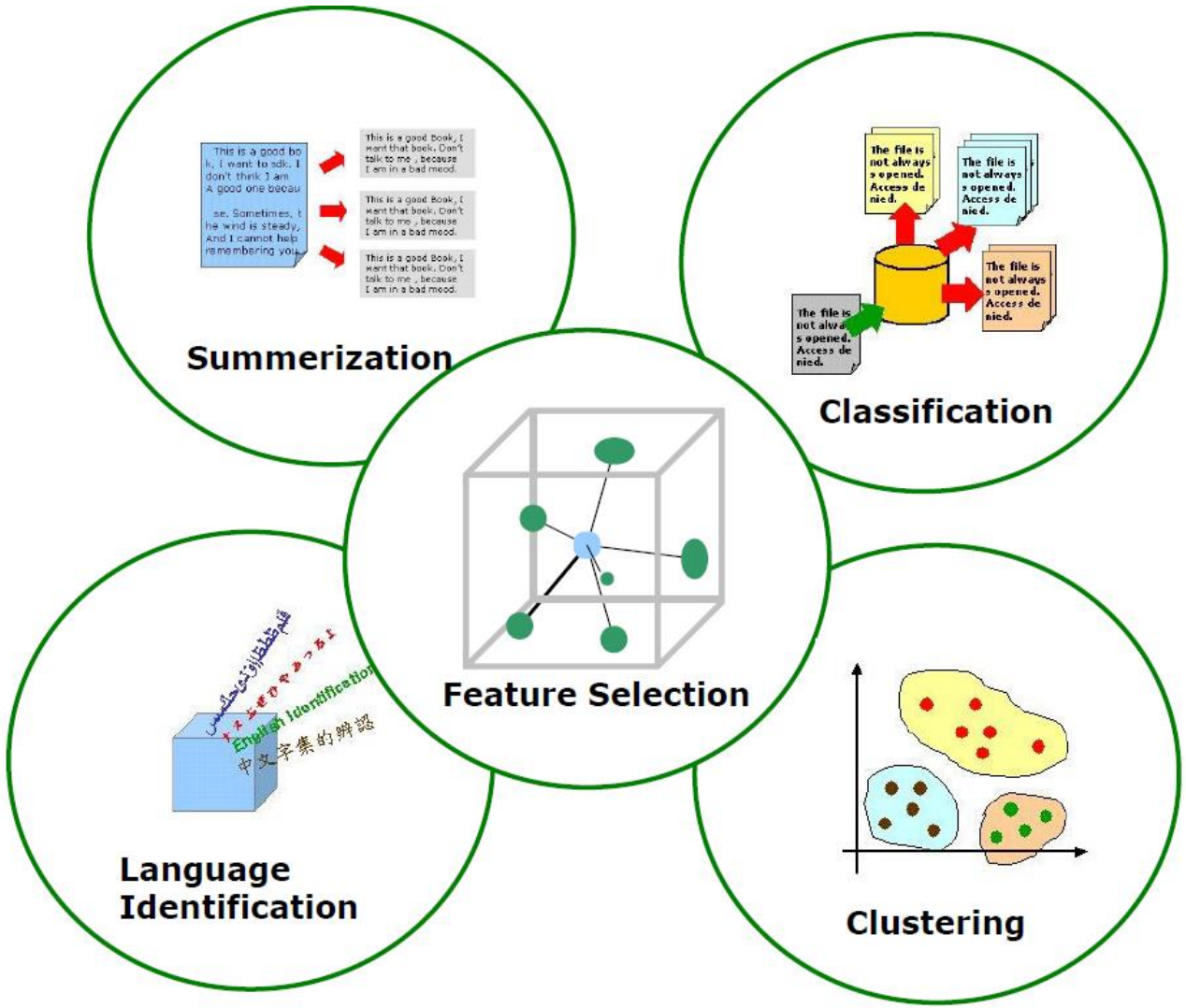
Twitter data collection

- Create your Twitter account
- Got to https://apps.twitter.com/app/new
- Create new app
- Save your credentials

Twitter data collection

- Install tweepy (follow pervious slide)

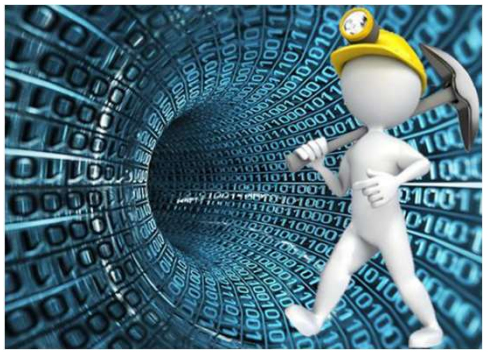
Text Analysis



Further Readings

- Data-Mining Boosts Netflix's Subscriber Base
- The Secret Sauce Behind Netflix's Hit, "House Of Cards": Big Data
- Everything You Wanted to Know About Data Mining but Were Afraid to Ask

- Presentation reference: Cestar College course presentation by Somayyeh (Bahar) Aghababaei



Lambton College

School of Computer Studies

Lecture 2

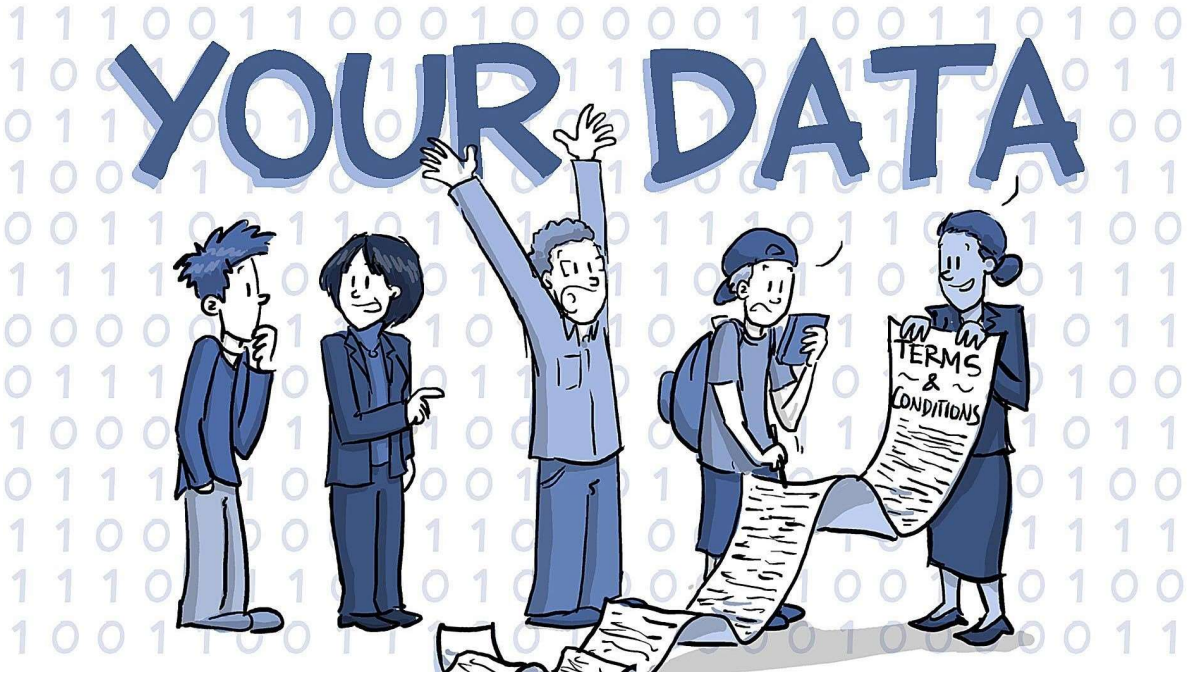
CBD-3335 Data Mining and Analysis

Learning Outcomes

- Data sources, data interpretation and methods of relating data to observations.
- Association pattern mining utilizing a variety of algorithms.
- Analyze clusters and outliers using a variety of methods.
- Perform mining for data streams and text data using algorithms.
- Apply data mining techniques for time series, spatial data and discrete sequences.

Data Taxonomies

- Categorizing data from different aspects
  - Data source
  - Data type
  - Structure
  - Time
  - Dimensionality
  - Quality



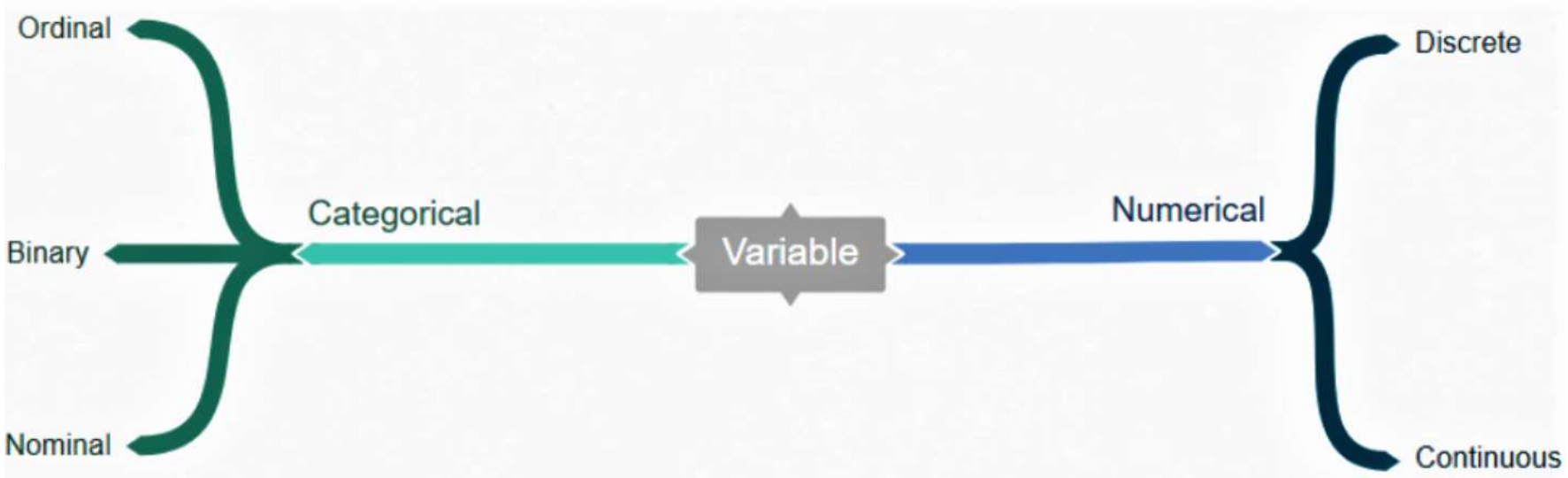
Data Sources
Where data comes from?

- Database and data warehouses: Queries
- Sensory data (usually real-time): temperature data
- Data entry: questionaries' data, data surveys
- Online data: data from other computers
- Embedded data: data from computers inside other devices such as mobile data
- Web data: data collected from web resources
- User-generated data: Content generated by user

Data Types

Data

- Numerical
- Categorical



Data Types

Numerical data is information that is measurable and represented as numbers. It can be

a) Discreate, or

b) Continuous.

1. Discreate: Numerical data that have a logical end. Examples: Variables for days in the month, or number of bugs logged.

2. Continuous : Numerical numbers that don't have a logical end.

Examples: Variables that represent money or height.

Data Types

Categorical data is any data that isn't number; which can mean a string of text or date. It can be mainly

a) Ordinal, or

b) Nominal.

1. Ordinal: Categorical data that have a set order to them.
   Examples: Having a priority on a bug such as "Critical" or "Low" or the ranking of a race as "First" or "Third".

2. Nominal: represent values with no set order to them.

Examples: Variables such as "Country" or "Marital Status".

Data Types

Binary data a special type of categorical data type having only two values – yes or no.

- This can be represented in different ways such as “True” and “False” or 1 and 0.
- Often used to represent one of two conceptually opposed values, e.g: the outcome of an experiment ("success" or "failure")

- Binary data occurs in many different technical and scientific fields, where it can be called by different names:
  - "bit" (binary digit) in computer science,
  - "truth value" in mathematical logic and related domains,
  - "binary variable" in statistics.

Structured & unstructured data

Structured data:

- Data that can be stored in a tabular form
- Every instance has the same structure
- Can be easily stored, organized, searched, recorded and merged with other structured data.
- Suitable for integration into an analytics records.

Example: The demographic data for a population where each row in the table describe one person (attributes: name, age, date of birth, gender, address, education, employment status etc.)

Structured & unstructured data

Unstructured data:

- Structure of data might not necessarily the same in every instance
- Each instance might have its own internal structure
- More common data type in real world; email tweets, text, posts, image, music, video, input from sensors etc. can be some examples.
- Difficult to analyze due to variation in structure.

Example: Dataset of webpages; each website might have data of an unique type).

Semi-structured data

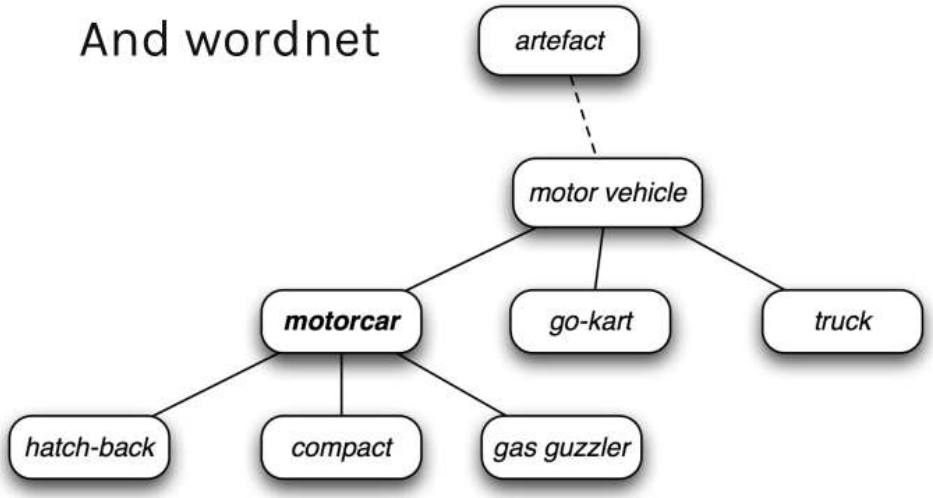
- Most XML data
- Werdnet:

  - WordNet is a lexical database of semantic relations between words

  - WordNet links words into semantic relations including synonyms, hyponyms, and meronyms.

python™
Natural Language Analyses with NLTK

And wordnet



Time

- Temporal data: financial data, twitter streaming data
  - Real-time data: time sensitive, if we miss reading any data, the consequence might be disastrous.
  - Non-time sensitive: We may miss some data without any dramatic consequences.
  - If data is numeric in type, data is a time series.
  - Static data: Fingerprint or biometric data

Dimension

- One dimensional:
  - body temperature, crime index, financial data
- Two-dimensional:
  - Image data (nxn matrix of pixels)
- N- dimensional:
  - Demographic data (age, height, weight, eye-color, race, DOB, POB, gender, occupation, ….)
- High dimensional:
  - Text data, Gene-expression data

Quality (1)

- Good quality data: Twitter of COVID-19 about recent Pandemic.
- Noise:
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission
- Outlier: IQ>160 when collecting IQ of high school students in public schools
- Inconsistent:(salary= **-5000$**)
  - containing discrepancies in codes or names

Quality (2)

- Twitter data example
- Incomplete: A broken tweet, (John, 21, male, ??, 160 lb, American, ??)
  - lacking attribute values, lacking certain attributes of interest, or containing only aggregated data
- Missing: Some missing tweets because of rate of sampling
- Duplicate: Many copies of a single tweet
- Irrelevant: Lady GAGA concert in Chicago

Quality (3)

- 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.

Textual Data Challenges

- Information is in **unstructured** textual format
- **Large** textual database
- **Very high** number of possible "**dimension**s" (but sparse):
  - all possible words and phrase types in the language!!
- **Complex** and subtle **relationships** between concepts in text
  - "AOL merges with Time-Warner" "Time-Warner is bought by AOL"

Textual Data Challenges

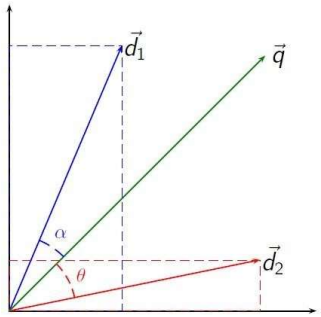
- **Word ambiguity** and context sensitivity
  - automobile = car = vehicle = Toyota
  - Apple (the company) or apple (the fruit)
- **Noisy data**: Spelling mistakes

Features (1)

- The piece of input data for which an output value is generated is formally called an instance.
  - Record, instance, object, observation, data
- The instance is formally described by a vector of features, which together constitute a description of all known characteristics of the instance.
  - Field, feature, variable, measurement
- The feature vectors can be seen as defining points in an appropriate multidimensional space.

Features (2)

- Vector methods can be correspondingly applied to them, such as computing the dot product or the angle between two vectors.
- Vector space model: Mostly in text data but can be used in other Pattern Recognition and data mining tasks.
  - Every data is a vector in a multi (high) dimensional space



Type of Features (1)

- Categorical aka nominal: consisting of one of a set of unordered items
  - Such as a gender of "male" or "female", or a blood type of "A", "B", "AB" or "O"
- Ordinal: consisting of one of a set of ordered items
  - Such as "large", "medium" or "small"
- Integer-valued
  - Such as a count of the number of occurrences of a particular word in an email

Type of Features (2)

- Real-valued
  - Such as a measurement of blood pressure
- Often, categorical and ordinal data are grouped together; likewise for integer-valued and real-valued data.
- Many algorithms work only with categorical data, such Naïve Bayes Classifier. How can we use numerical features? and require that real-valued or integer-valued data be discretized into groups (e.g., less than 5, between 5 and 10, or greater than 10).

Type of Features (2)

- Real-valued
  - Such as a measurement of blood pressure
- Often, categorical and ordinal data are grouped together; likewise for integer-valued and real-valued data.
- Many algorithms work only with categorical data, such **Naïve Bayes Classifier**. In these cases, real-valued or integer-valued data are discretized into groups (e.g., less than 5, between 5 and 10, or greater than 10).

Feature Selection (1)

- When do we employ feature selection?
  - For **very high dimensional data**, in which feature extraction might be expensive
  - **Features are not numeric**
  - **We are looking for meaningful features**

$$y_j = a_{j1}x_1 + a_{j2}x_2 + \square + a_{jm}x_m$$

Feature Selection(2)

- FeatureSelection
  - Searching the feature space for a subset of features maximizing an objective function (quality index)
  - Wrappers
  - Filters
    - Feature ranking
  - Embedded
  - Markov Blanket

Feature Selection (3)

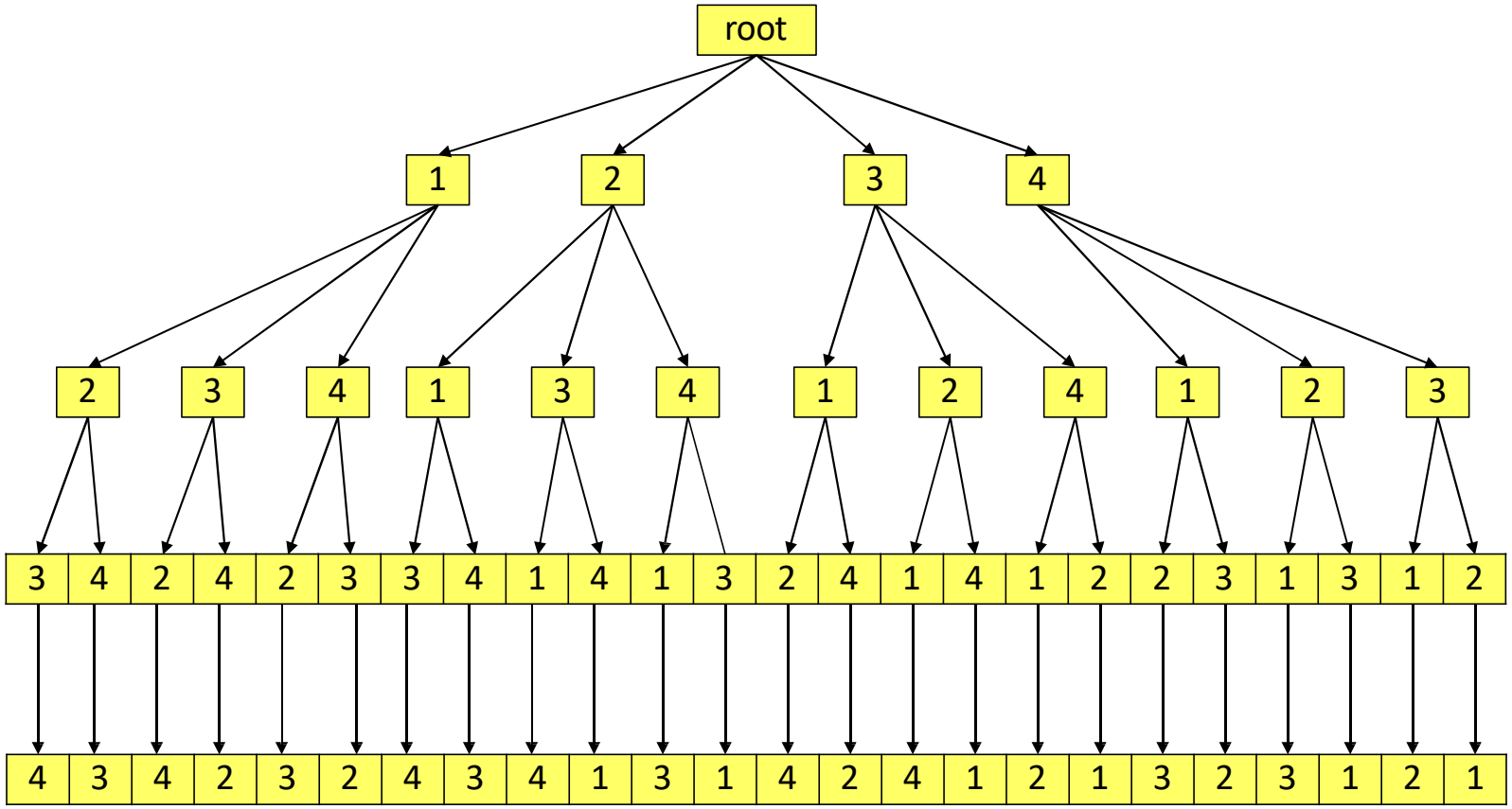
- Search strategy: search the power set of the feature set to find the optimum feature subset
  - Exhaustive search: the order of the search space is $O(2^m)$
  - Search strategy to reduce the size of the search space
    - Sequential Forward Selection (SFS)
    - Sequential Backward Selection (SBS)
    - Beam search
    - Simulated annealing

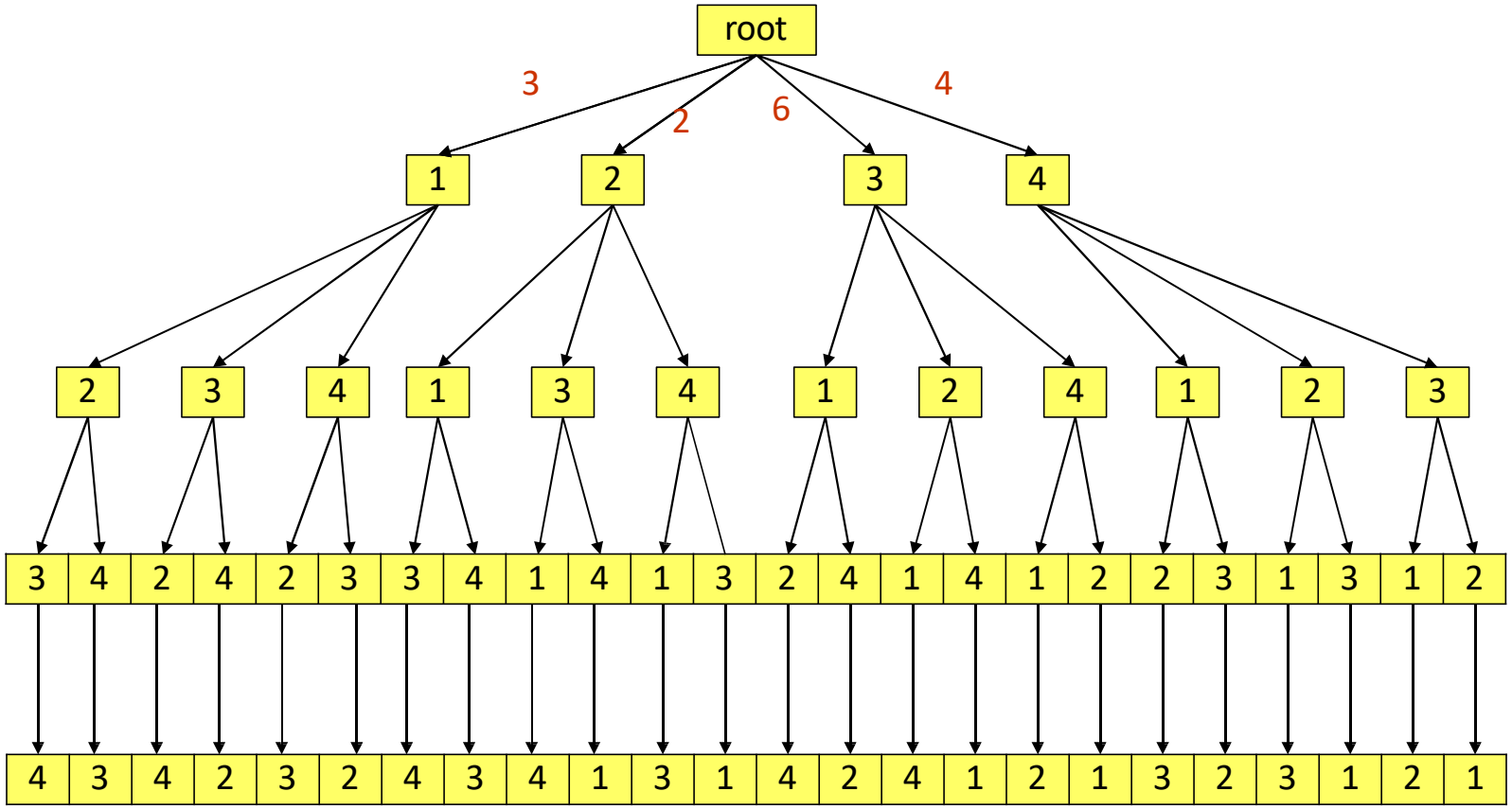
Search Strategies (2)

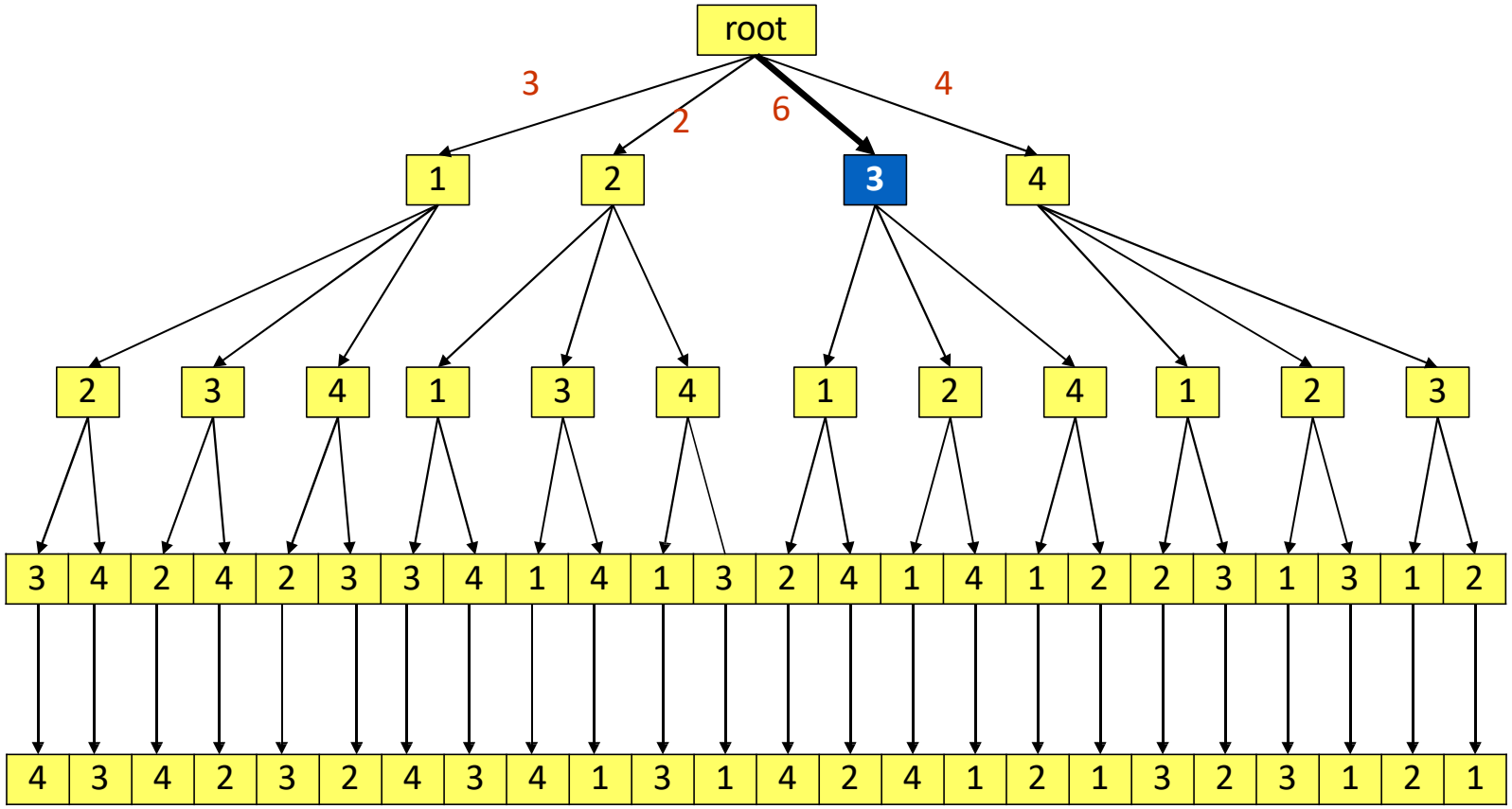
- Sequential Forward Selection (SFS)
  1. Start with empty set: Y← {}
  2. Select the next best feature that maximizes the objective function of the selected features

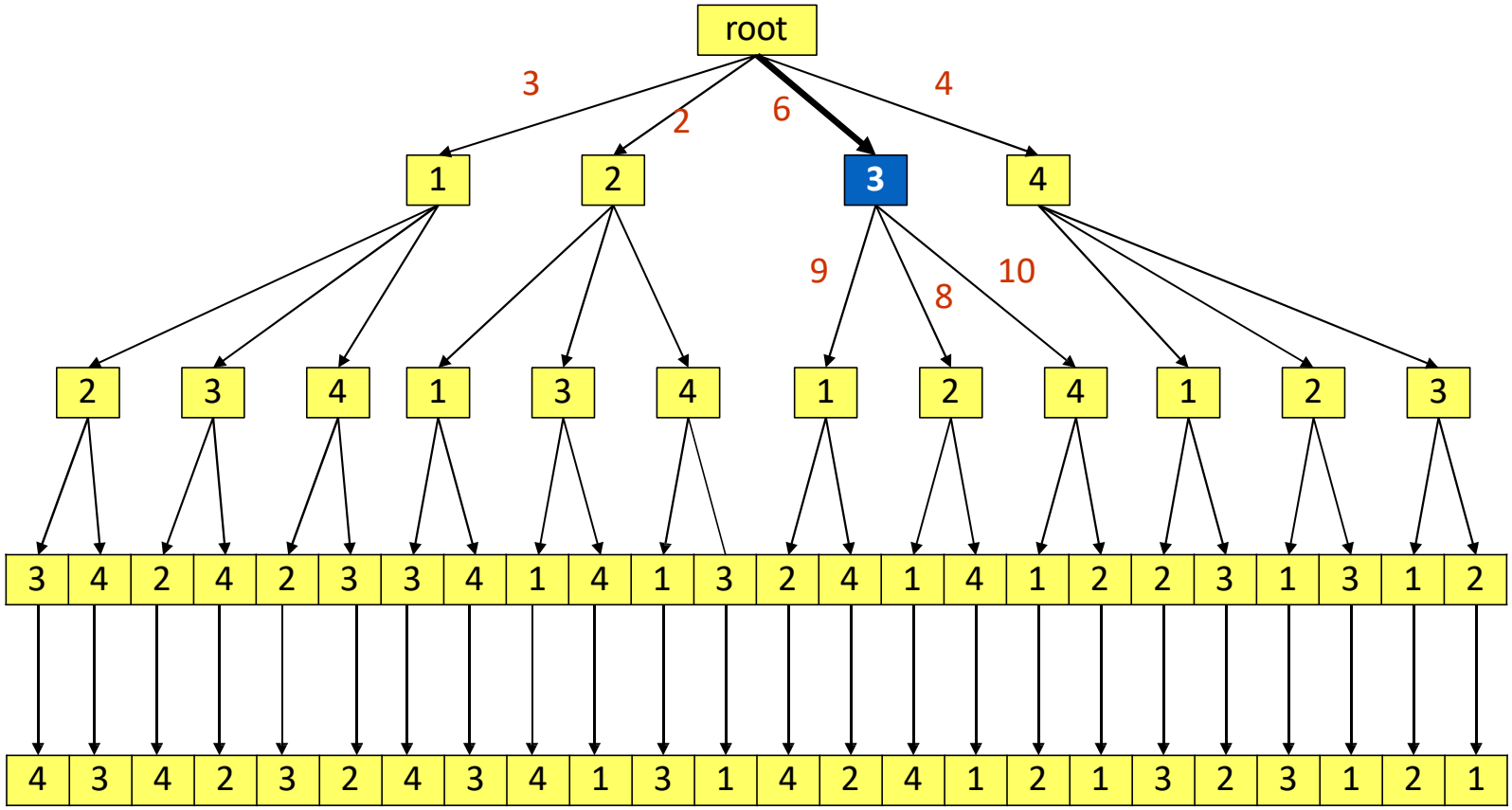
$$z \leftarrow \underset{x \notin Y}{\operatorname{argmax}}[h(Y+\{x\})]$$

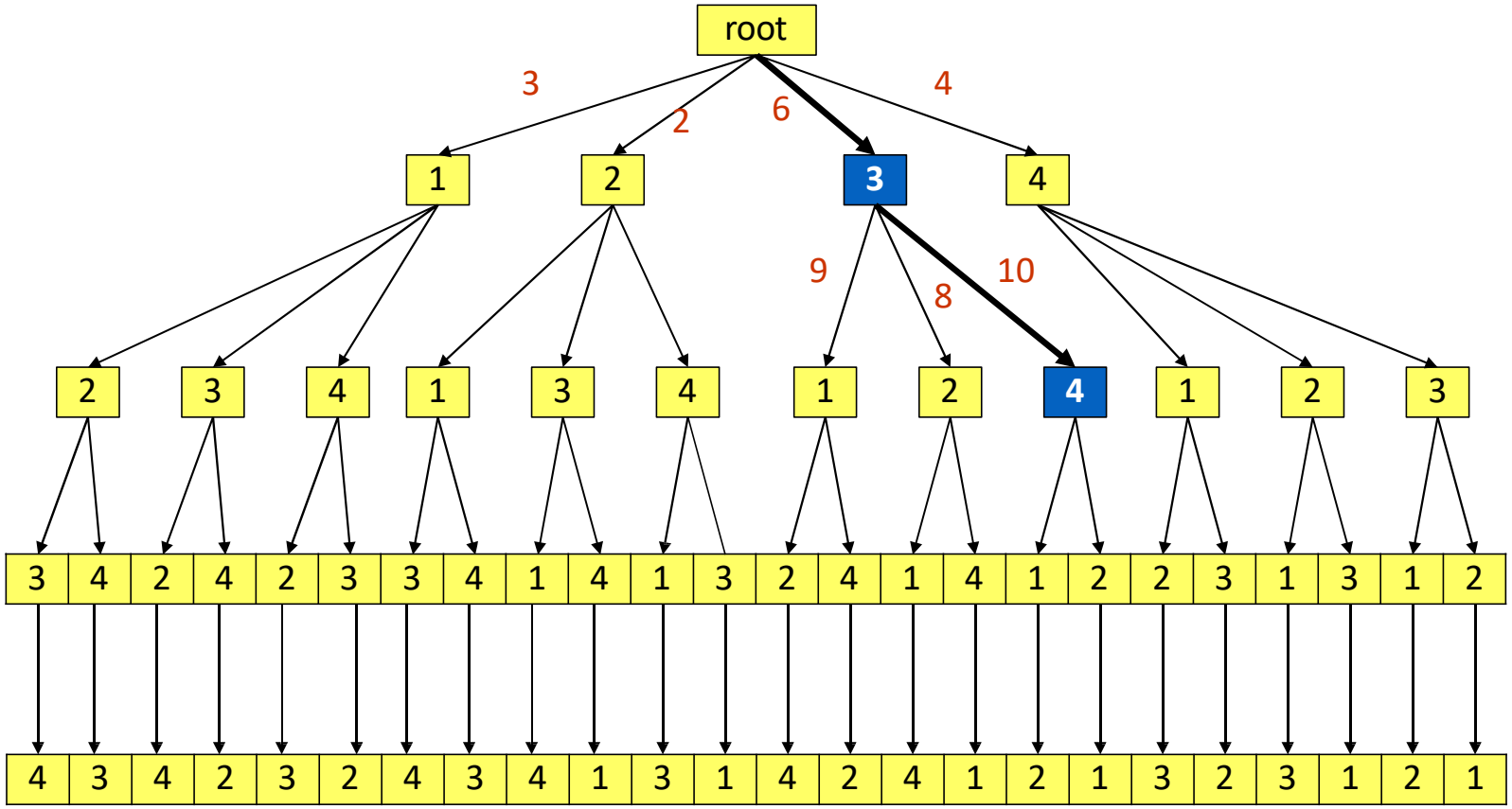
  3. Update Y: $Y \leftarrow Y+\{z\}$
  4. Go to 2
- Example:
  - Select the best feature subset among $\mathbf{X=\{x_1,x_2,x_3,x_4\}}$
  - Objective function
- $\mathbf{h=3x_1+2x_2+6x_3+4x_4-2x_1x_2-4x_1x_2x_3-7x_1x_2x_3x_4}$

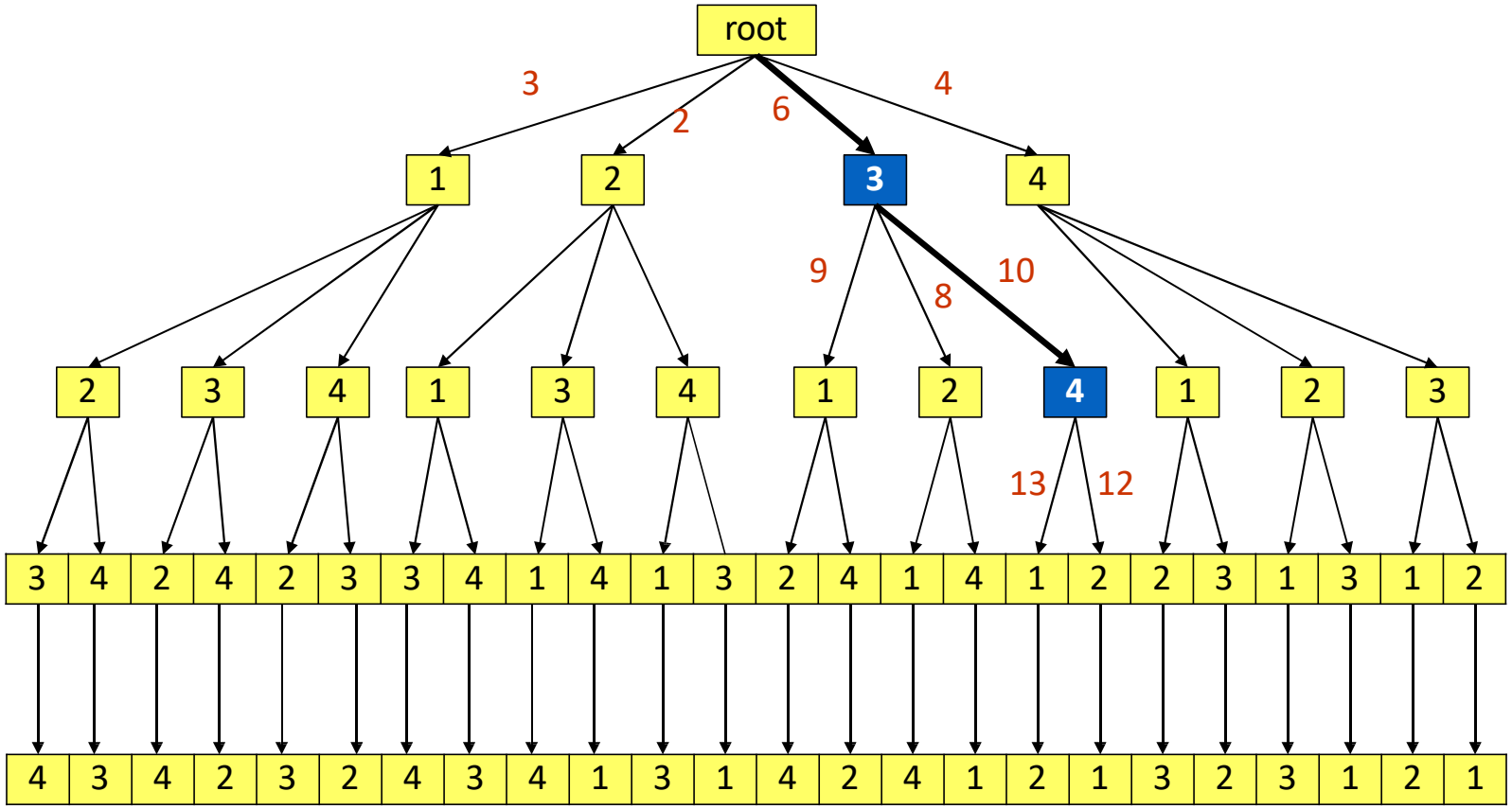

root
1
2
3
4
2 3 4 1 3 4 1 2 4 1 2 3
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1

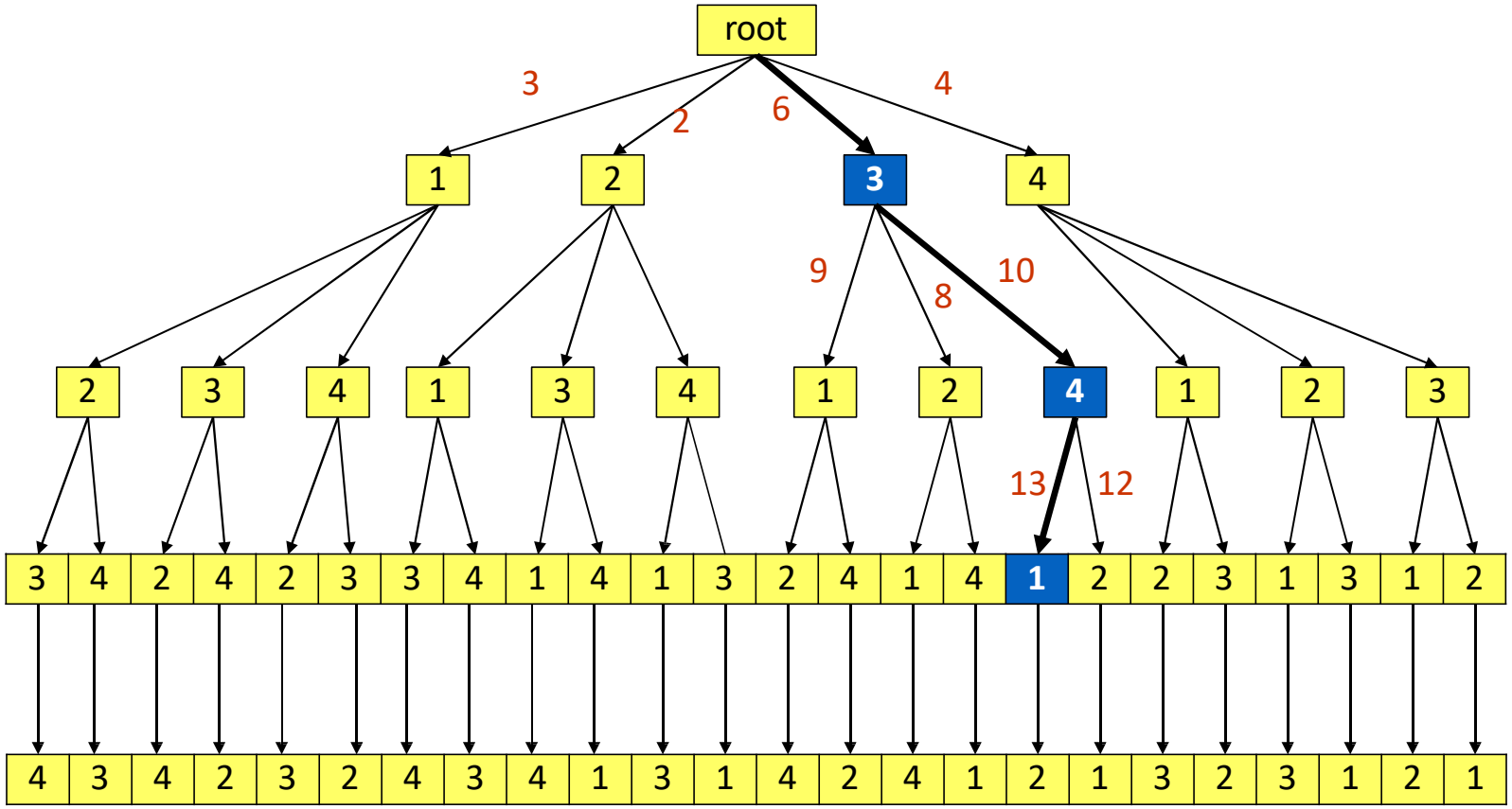

root
3
2
6
4
1
2
3
4
2 3 4 1 3 4 1 2 4 1 2 3
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1

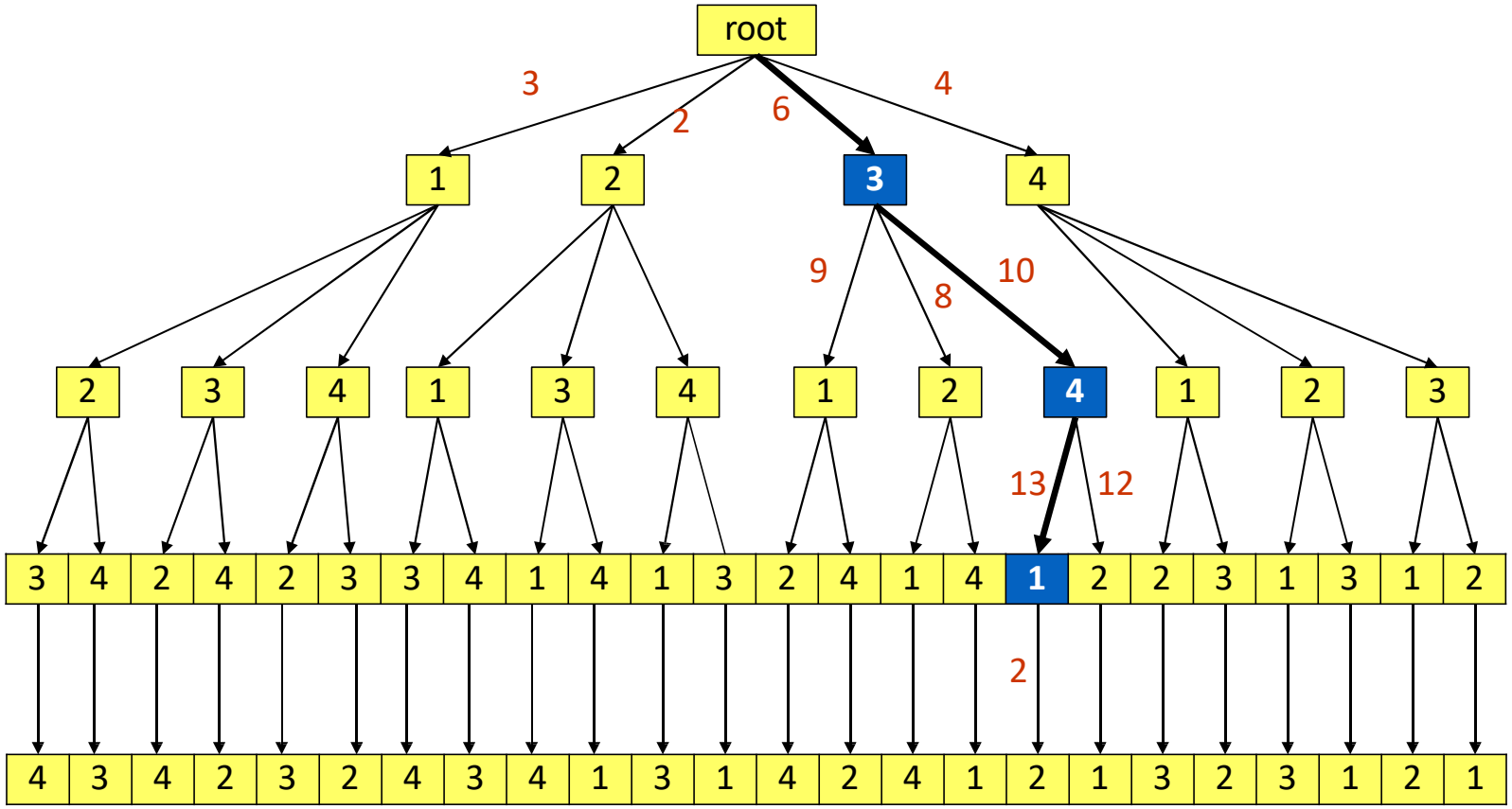

root
3
2
6
4
1
2
3
4
2 3 4 1 3 4 1 2 4 1 2 3
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1


root
3
2
6
4
1
2
3
4
9
8
10
2 3 4 1 3 4 1 2 4 1 2 3
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1


root
3
2
6
4
1
2
3
4
9
8
10
2 3 4 1 3 4 1 2 4 1 2 3
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1


root
3
2
6
4
1
2
3
4
9
8
10
2
3
4
1
3
4
1
2
4
1
2
3
13
12
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1


root
3
2
6
4
1
2
3
4
9
8
10
2
3
4
1
3
4
1
2
4
1
2
3
13
12
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1


root
3
2
6
4
1
2
3
4
9
8
10
2
3
4
1
3
4
1
2
4
1
2
3
13
12
3 4 2 4 2 3 3 4 1 4 1 3 2 4 1 4 1 2 2 3 1 3 1 2
2
4 3 4 2 3 2 4 3 4 1 3 1 4 2 4 1 2 1 3 2 3 1 2 1

Search Strategies (3)

- SFS performs best when the optimal subset has a small number of features
- When the search is near the empty set, a large number of states can be potentially evaluated
- Towards the full set, the region examined by SFS is narrower since most of the features have already been selected

Lambton College
School of Computer Studies

Lambton College

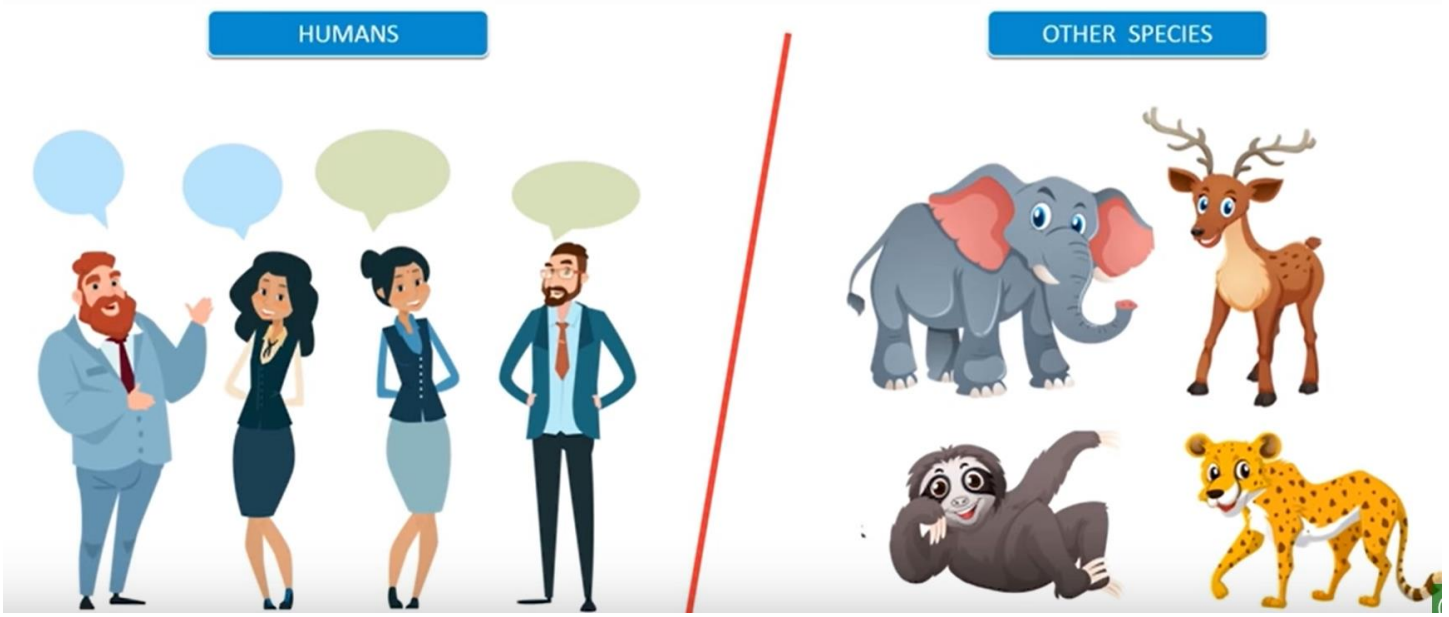
Lecture 3

CBD-3335 Data Mining and Analysis

Topics

- Natural / Human language
- What is NLP?
- Applications of NLP
- NLP components
- Text processing

Human language



Natural Language Processing (NLP)

- Part of computer science and Artificial Intelligent which deals with human language.

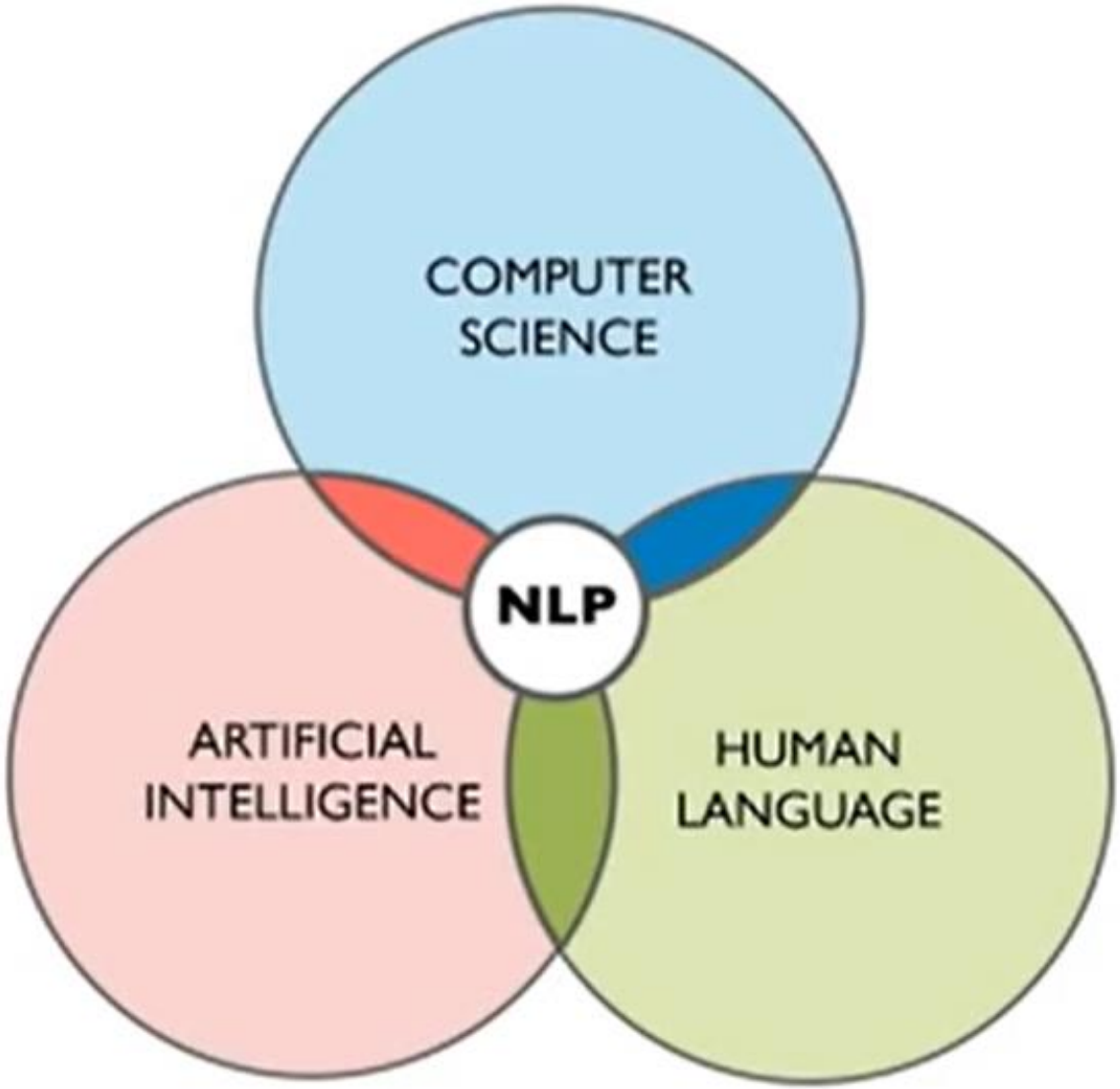


image: Edureka

NLP Applications

Sentimental Analysis

Chatbot

Speech Recognition

Machine Translation

Spell Checking

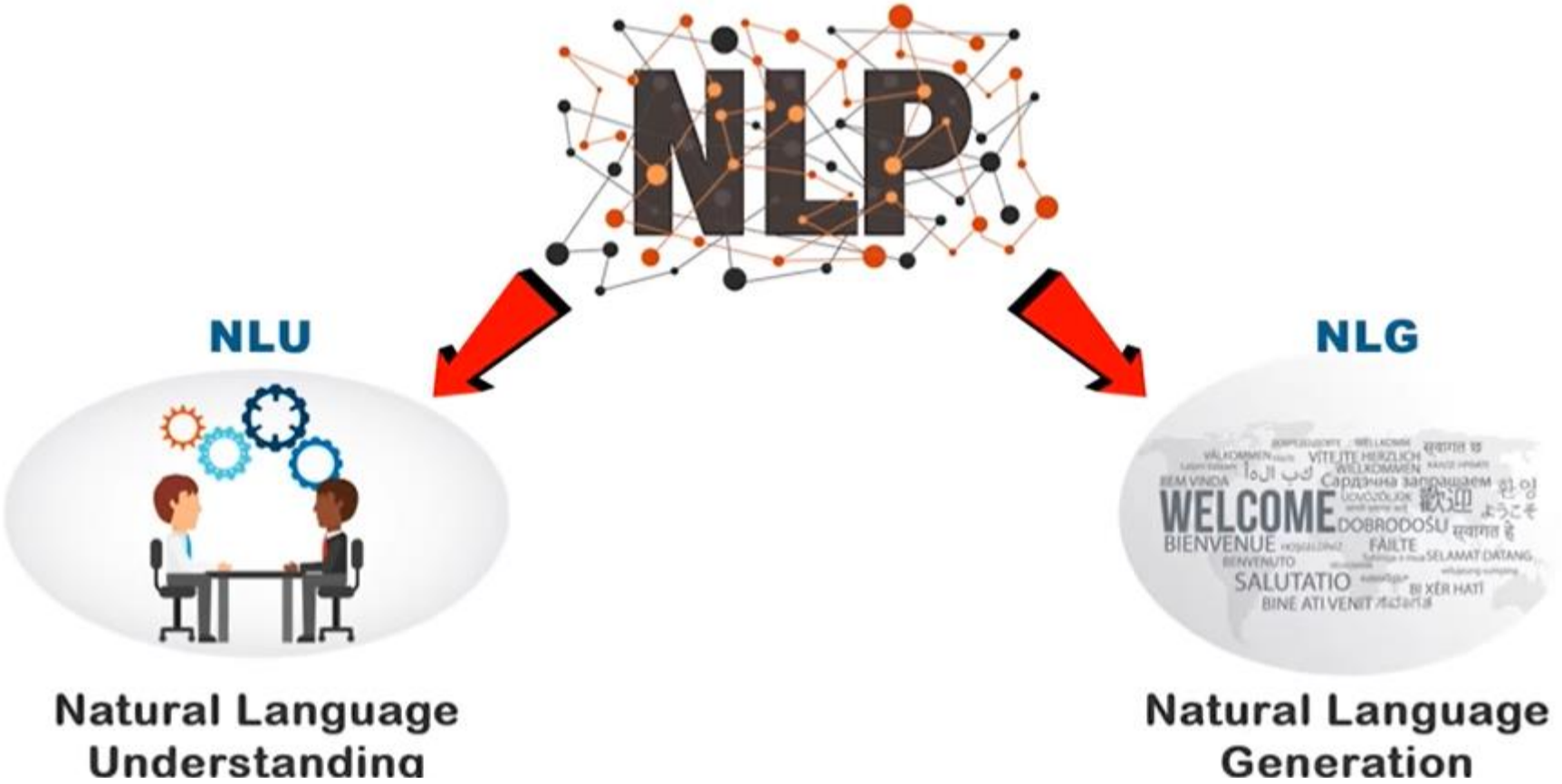
Information Extraction

Keyword Search

Advertisement Matching

image: Edureka

NLP Components



NLU Natural Language Understanding

NLG Natural Language Generation

Text Data

Usually natural language text (human generated)

Unstructured or semi-structured

Features are tokens of text (characters, words, n-grams, sentences, part of speech, named entities, semantics, and so on) terms

Data set: Corpora (corpus)

Text Data Representation

- Lexical
  - Character
  - Words
  - Phrases
  - Part-of-speech tags

- Syntactic
  - Taxonomies
  - Vector-space model
  - Language models

Character Level

- Sequences of characters are extracted
- A document is represented by a frequency distribution of sequences
  - Each character sequence of length 1, 2, 3, … represents a feature with its frequency
- Good and bad sides
  - It is very robust since avoids language morphology: useful for language identification
  - It captures simple patterns on character level: useful for spam and plagiarism detection
  - For deeper semantic tasks, the representation is too weak

Word level

- The most common representation: bag of words

- Pre-processing process
  - Tokenization
  - Converting to lower case?
  - Stemming?
  - Stop-word reduction?
  - Band-pass filtering?
  - Document length normalization

Relations between Words (1)

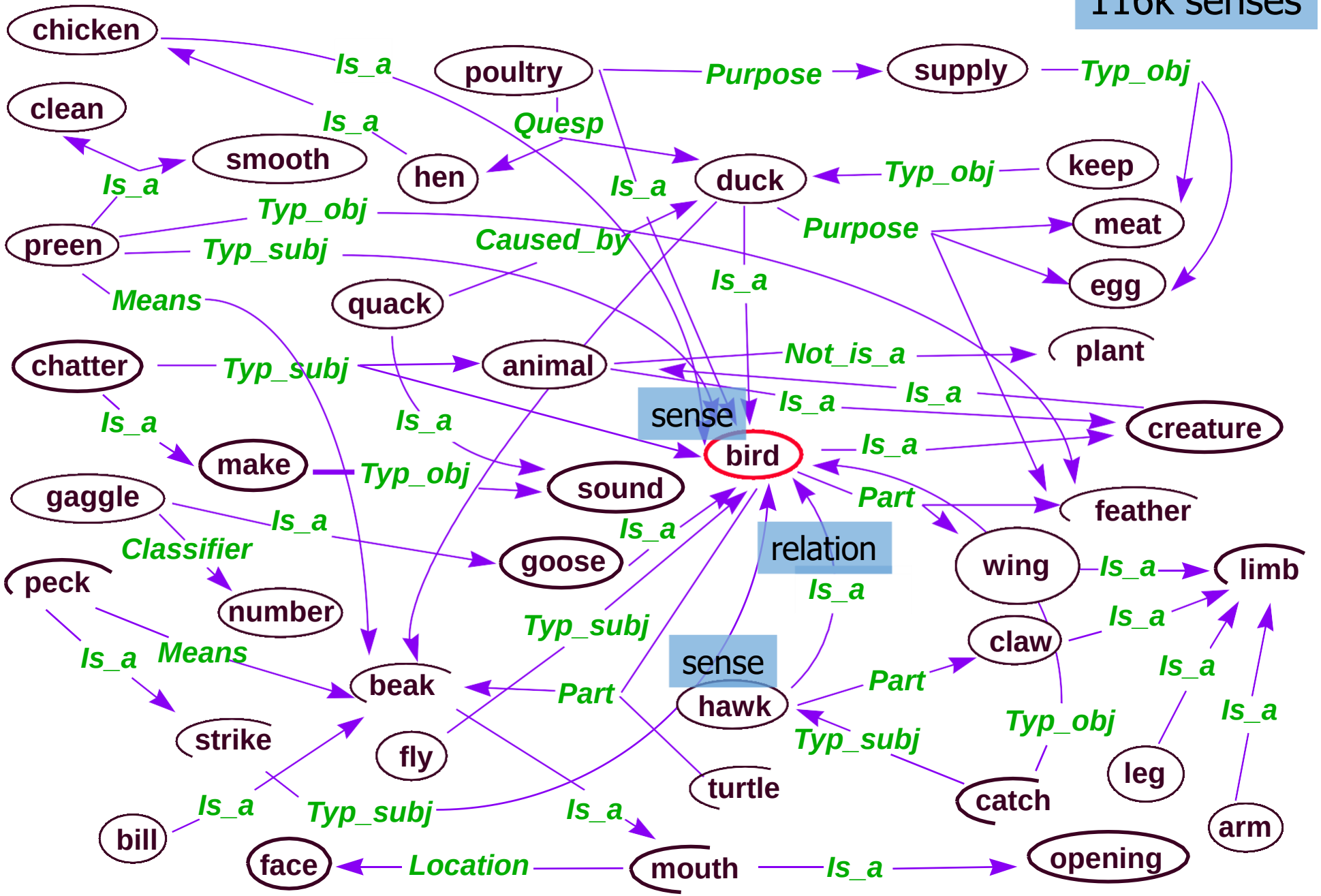
- Local relations
  - Co-occurrence and co-location: two word often appear together in documents (or in a specific category of documents)
  - Whole document
  - A sequence or in a sliding window
  - If we see “united” in a document what is the probability that the next word is “nations” and what is the probability of “states”?
  - We can learn from lots of example documents. Class information also helps.

Relations between Words (2)

- Semantic relations among words (WordNet)
  - **Homonomy**: same form, but different meaning (e.g. bank: river bank, financial institution)
  - **Polysemy**: same form, related meaning (e.g. bank: blood bank, financial institution)
  - **Synonymy**: different form, same meaning (e.g. singer, vocalist)
  - **Hyponymy**: one word denotes a subclass of another (e.g. breakfast, meal)

WordNet example

26 relations
116k senses

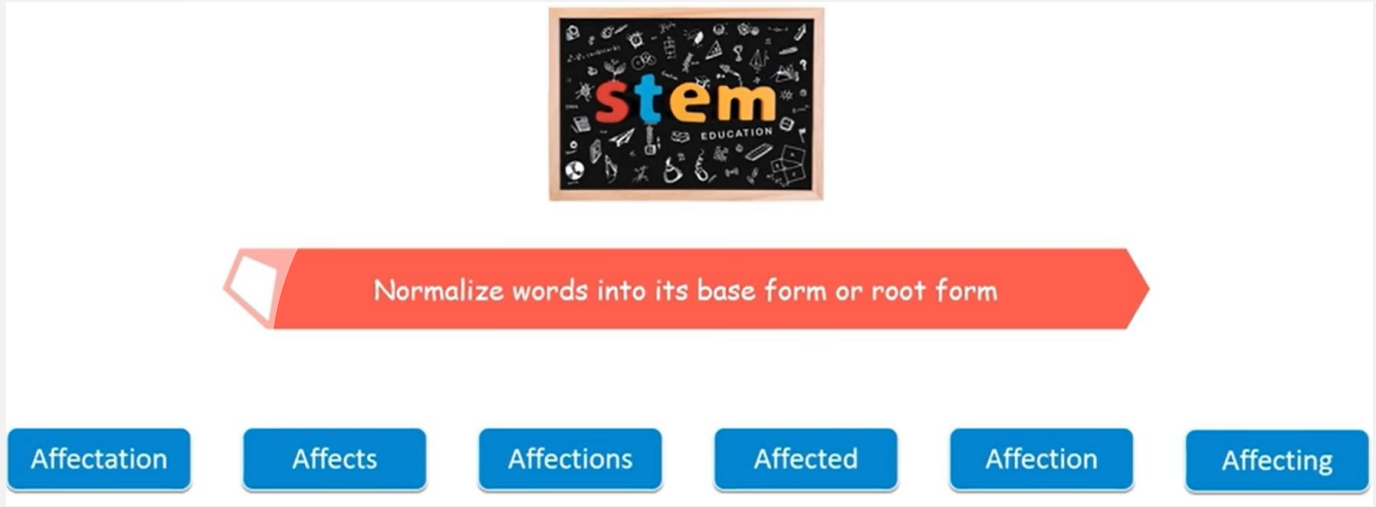


Stop-words

- Stop-words are words that from non-linguistic view do not carry information
  - They have mainly functional role
  - Usually we remove them to help the methods to perform better

- Stop words are language dependent – examples:
  - **English**: A, ABOUT, ABOVE, ACROSS, AFTER, AGAIN, AGAINST, ALL, ALMOST, ALONE, ALONG, ALREADY
  - **Dutch**: de, en, van, ik, te, dat, die, in, een, hij, het, niet, zijn, is, was, op, aan, met, als, voor, had, er, maar, om, hem, dan, zou, of, wat, mijn, men, dit, zo, ...

Stemming


stem
EDUCATION
Normalize words into its base form or root form
Affectation
Affects
Affections
Affected
Affection
Affecting
Affect

Stemming (2)

- For English, we usually use Porter stemmer http://www.tartarus.org/~martin/PorterStemmer/
- Example cascade rules used in English Porter stemmer
  - ATIONAL -> ATE relational -> relate
  - TIONAL -> TION conditional -> condition
  - ENCI -> ENCE valenci -> valence
  - ANCI -> ANCE hesitanci -> hesitance
  - IZER -> IZE digitizer -> digitize
  - ABLI -> ABLE conformabli -> conformable
  - ALLI -> AL radicalli -> radical
  - ENTLI -> ENT differentli -> different
  - ELI -> E vileli -> vile
  - OUSLI -> OUS analogousli -> analogous

Phrase Level

- Instead of having just single words we can deal with phrases
- We use two types of phrases:
  - Phrases as frequent continuous word sequences
  - Phrases as frequent non- continuous word sequences
  - Both types of phrases could be identified by simple dynamic programming algorithm
- The main effect of using phrases is to more precisely identify senses

Google N-Gram

- Google N-gram
- Some statistics of the corpus:
  - File sizes: approx. 24 GB compressed (gzip'ed) text files
  - Number of tokens: 1,024,908,267,229
  - Number of sentences: 95,119,665,584
  - Number of unigrams: 13,588,391
  - Number of bigrams: 314,843,401
  - Number of trigrams: 977,069,902
  - Number of fourgrams: 1,313,818,354
  - Number of fivegrams: 1,176,470,663

Part-of-Speech level

- introduces word-types to differentiate words functions
  - For text-analysis part-of-speech information is used mainly for “information extraction” where we are interested in e.g. named entities which are “noun phrases”
  - Another possible use is for feature reduction
  - it is known that nouns carry most of the information in text documents
- Online part-of-speech tagger

Part-of-Speech Table

part of speech	function or "job"	example words	example sentences
Verb	action or state	(to) be, have, do, like, work, sing, can, must	EnglishClub.com **is** a web site. I **like** EnglishClub.com.
Noun	thing or person	pen, dog, work, music, town, London, teacher, John	This is my **dog**. He lives in my **house**. We live in **London**.
Adjective	describes a noun	a/an, the, 69, some, good, big, red, well, interesting	My dog is **big**. I like **big** dogs.
Adverb	describes a verb, adjective or adverb	quickly, silently, well, badly, very, really	My dog eats **quickly**. When he is **very** hungry, he eats **really** quickly.
Pronoun	replaces a noun	I, you, he, she, some	Tara is Indian. **She** is beautiful.
Preposition	links a noun to another word	to, at, after, on, but	We went **to** school **on** Monday.
Conjunction	joins clauses or sentences or words	and, but, when	I like dogs **and** I like cats. I like cats **and** dogs. I like dogs **but** I don't like cats.
Interjection	short exclamation, sometimes inserted into a sentence	oh!, ouch!, hi!, well	**Ouch**! That hurts! **Hi**! How are you? **Well**, I don't know.

Part-of-Speech examples

verb
Stop!

noun	verb
John	works.

noun	verb	verb
John	is	working.

pronoun	verb	noun
She	loves	animals.

noun	verb	adjective	noun
Animals	like	kind	people.

noun	verb	noun	adverb
Tara	speaks	English	well.

noun	verb	adjective	noun
Tara	speaks	good	English.

pronoun	verb	preposition	adjective	noun	adverb
She	ran	to	the	station	quickly.

pron.	verb	adj.	noun	conjunction	pron.	verb	pron.
She	likes	big	snakes	but	I	hate	them.

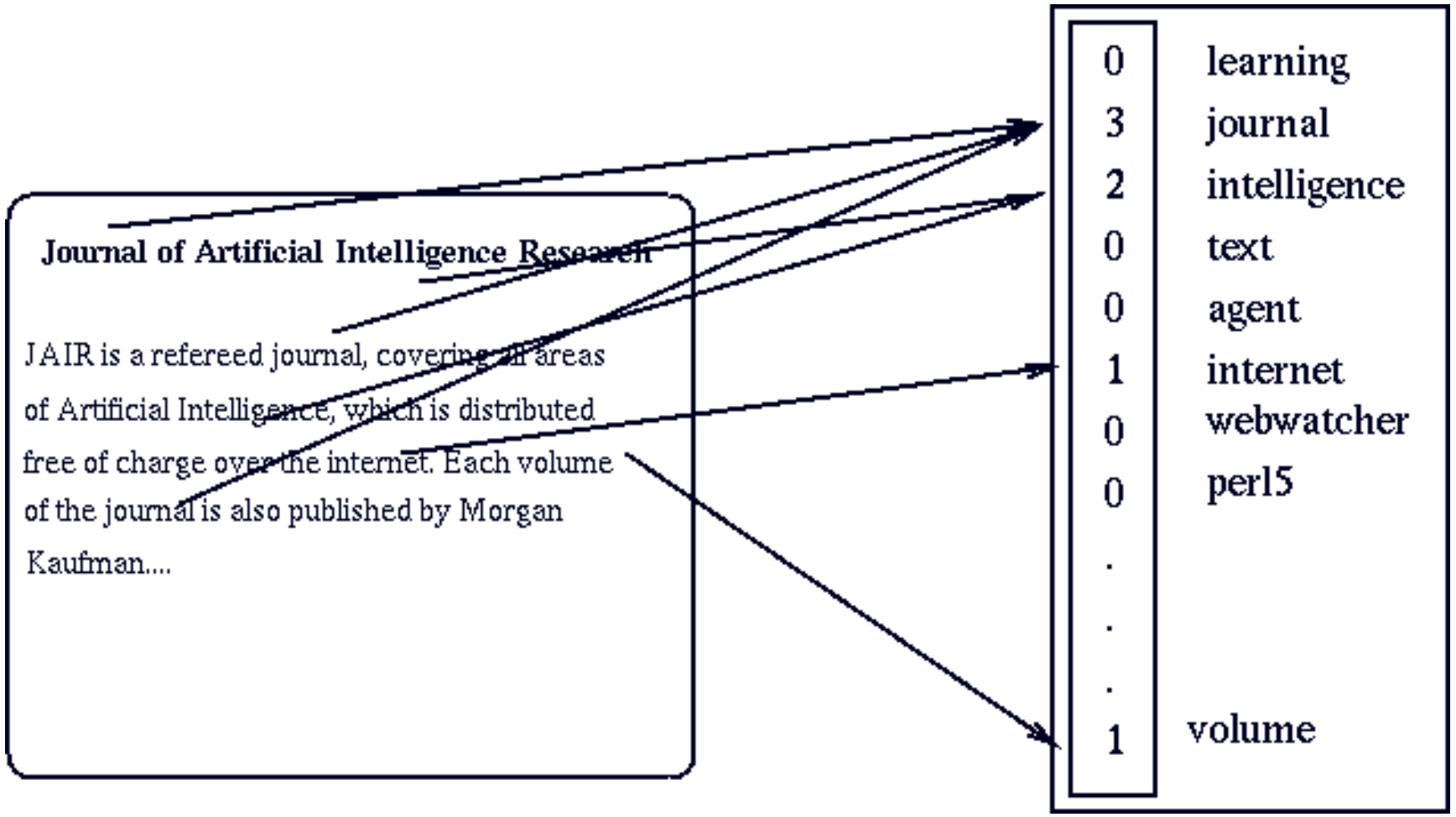
Here is a sentence that contains every part of speech:

interjection	pron.	conj.	adj.	noun	verb	prep.	noun	adverb
Well,	she	and	young	John	walk	to	school	slowly.

Vector-space model level

- The most common way to represent documents is Vector Space Model
  - first to transform them into **sparse numeric vectors** and
  - then deal with them with **linear algebra operations**
- We ignore the linguistic structure within the text (word order will be lost)
- It is called “structural curse” because ignoring the linguistic structure doesn’t harm efficiency of solutions in many problems
- This representation is also called “Bag-Of-Words”
- Typical tasks on vector-space-model are classification, clustering, visualization etc.

Bag-of-words document representation



Bag-of-words matching

How single stars lost their companions
Space Daily - Sep 15, 2011
by Staff Writers Not all stars are loners. In our home galaxy, the Milky Way, about half of all stars have a companion and travel through space in a binary system. But explaining why some stars are in double or even triple systems while others are ...

Coupled stars break up for the single life
Astronomy Now Online - Gemma Lavender - Sep 16, 2011
Why some stars prefer to be single, while others are either paired up or in trios, could have been answered by a team of astronomers at the Max-Planck-Institute for Radio astronomy and the University of Bonn with the help of sophisticated computer ...

In both stories

4	stars	2
1	single	2
1	triple	1

why
some
others
while

have are in
of or a the

Bag-of-words matching

How single stars lost their companions
Space Daily - Sep 15, 2011
by Staff Writers Not all stars are loners. In our home galaxy, the Milky Way, about half of all stars have a companion and travel through space in a binary system. But explaining why some stars are in double or even triple systems while others are ...

Coupled stars break up for the single life
Astronomy Now Online - Gemma Lavender - Sep 16, 2011
Why some stars prefer to be single, while others are either paired up or in trios, could have been answered by a team of astronomers at the Max-Planck-Institute for Radio astronomy and the University of Bonn with the help of sophisticated computer ...

Only in story 1

about
binary
companion
double
even
explaining
galaxy
half
home
loners
lost
milky
space
system
through
travel
our
way

In both stories

4	stars	2
1	single	2
1	triple	1

why
some
others
while

have are in
of or a the

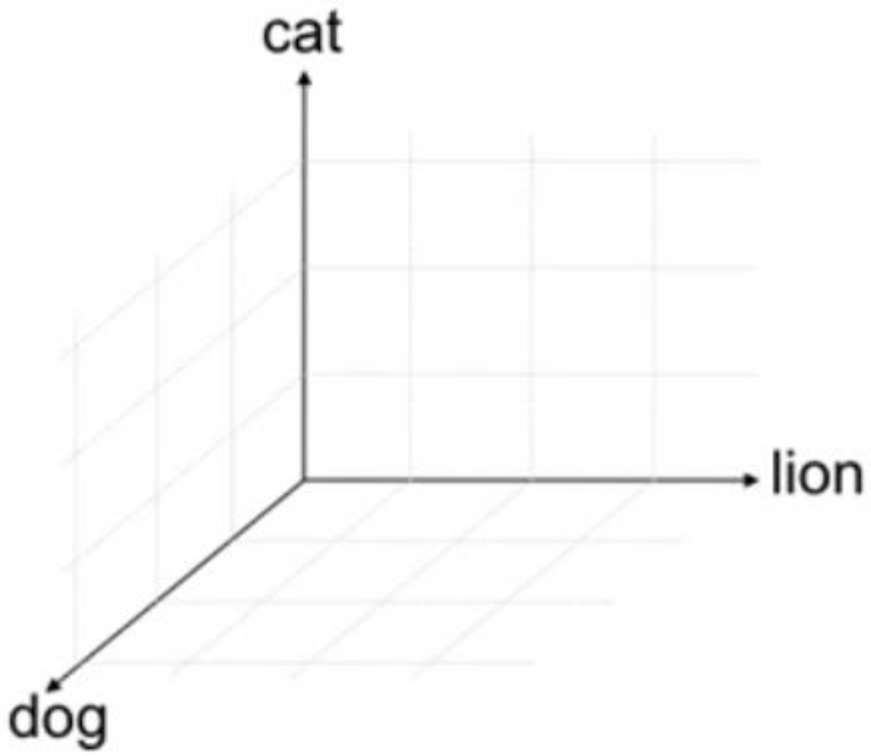
Only in story 2

answered
been
break
could
either
institute
max
planck
radio
team
astronomy
bonn
computer
coupled
help
life
paired
prefer
sophisticated
university

How to combine all this into a similarity measure?

Bag-of-words& Vector Space

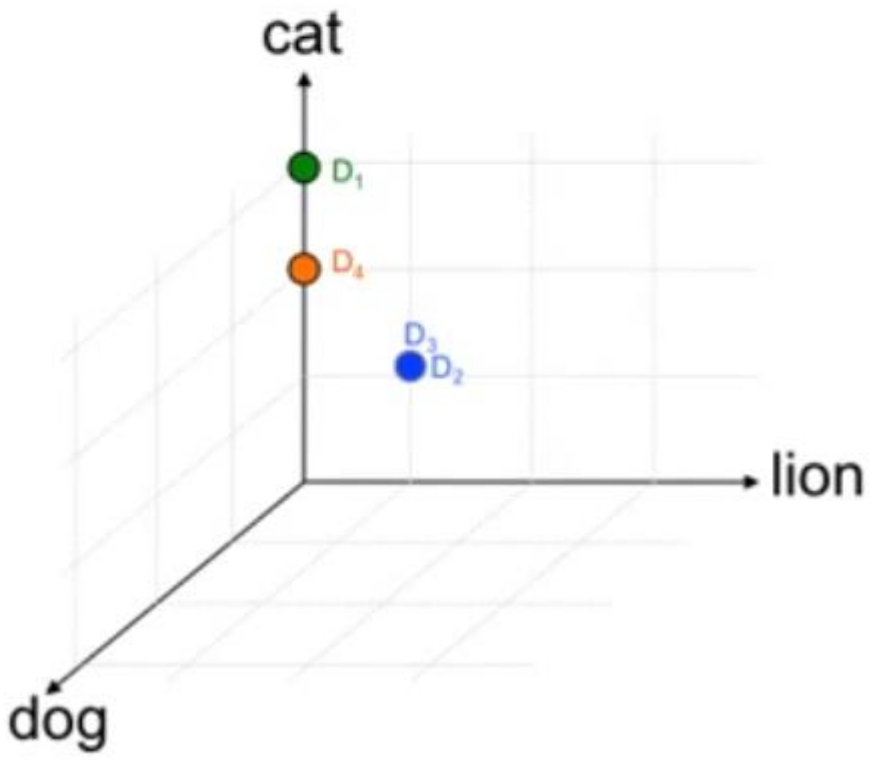
- Separate dimension for each distinct word
  - words = coordinate vectors
  - value along dimension “cat” ~ number of times “cat” occurs



Bag-of-words & Vector Space

- Separate dimension for each distinct word
  - words = coordinate vectors
  - value along dimension “cat” ~ number of times “cat” occurs

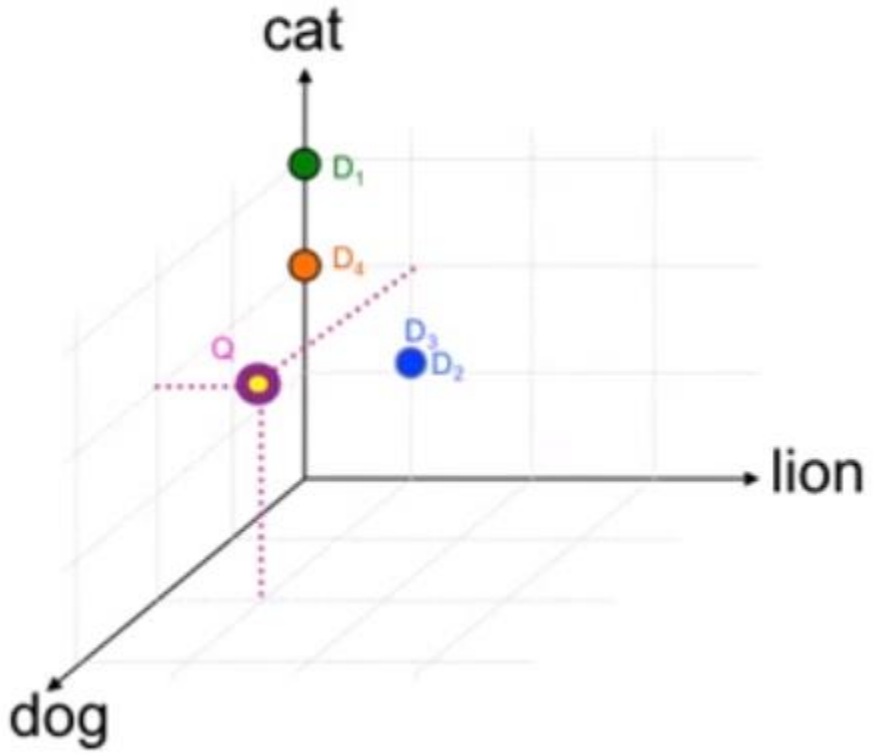
- D_1: “cat cat cat” → (0,3,0)
- D_2: “cat lion” → (0,1,1)
- D_3: “lion cat” → (0,1,1)
- D_4: “cat cat” → (0,2,0)



Bag-of-words & Vector Space

- Separate dimension for each distinct word
  - words = coordinate vectors
  - value along dimension “cat” ~ number of times “cat” occurs

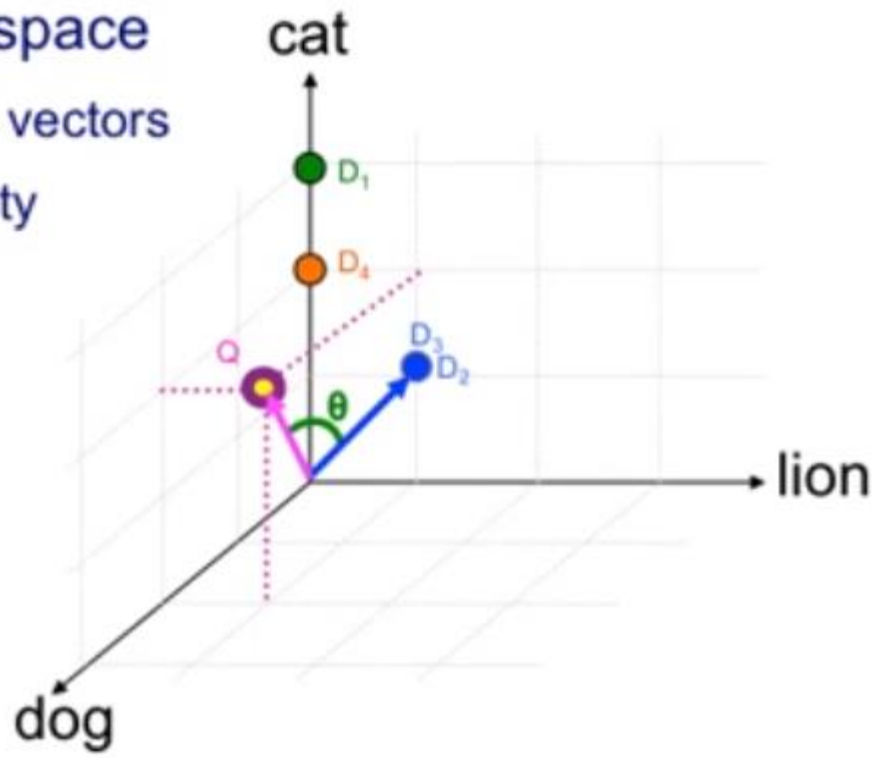
- D_1: “cat cat cat” → (0,3,0)
- D_2: “cat lion” → (0,1,1)
- D_3: “lion cat” → (0,1,1)
- D_4: “cat cat” → (0,2,0)
  - Q: cat cat lion dog dog → (2,2,1)



Bag-of-words & Vector Space

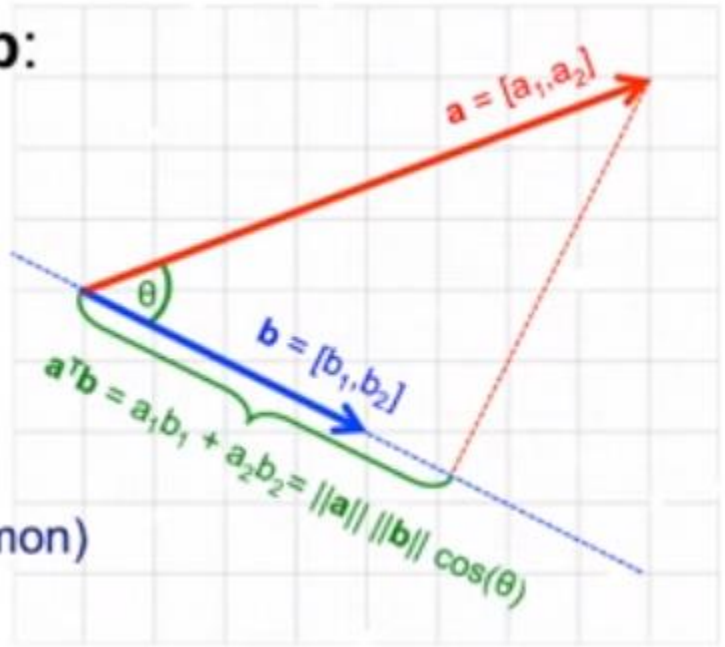
- Separate dimension for each distinct word
  - words = coordinate vectors
  - value along dimension "cat" ~ number of times "cat" occurs
- Comparing documents to queries
  - distance between points in space
    - Euclidean, or angle between vectors
    - usually expressed as similarity

- D_1: "cat cat cat" → (0,3,0)
- D_2: "cat lion" → (0,1,1)
- D_3: "lion cat" → (0,1,1)
- D_4: "cat cat" → (0,2,0)
  - Q: cat cat lion dog dog → (2,2,1)



Dot product

- Similarity of document vectors **a** and **b**:
  - $\mathbf{a} = [a_1\ a_2\ \ldots\ a_d]$, $\mathbf{b} = [b_1\ b_2\ \ldots\ b_d]$
  - $\mathbf{a}^T\mathbf{b} = a_1b_1 + \ldots + a_db_d = \Sigma_i\ a_ib_i$
- Geometrically:
  - length of projection of **a** onto **b**
    - highest if **a**,**b** point in the same direction
    - zero if **a**,**b** are orthogonal (no words in common)



Word (term) Weighting

- In the bag-of-words representation each word is represented as a separate variable having numeric weight (importance)
- The most popular weighting schema is normalized word frequency TFIDF:

$$tfidf(w) = tf \cdot \log\left(\frac{N}{df(w)}\right)$$

- tf(w) – term frequency (number of word occurrences in a document)
- df(w) – document frequency (number of documents containing the word)
- N – number of all documents
- tfIdf(w) – relative importance of the word in the document

The word is more important if it appears several times in a target document

The word is more important if it appears in less documents

TF.IDF

Example: Suppose we have three documents

D1="mid term1 marks"

D2="mid term2 marks"

D3="total"

Term Frequency

	mid	term1	marks	term2	total
D1	1	1	1	0	0
D2	1	0	1	1	0
D3	0	0	0	0	1
Term Doc freq	2	1	2	1	1

TF.IDF

Inverse Document Frequency can be calculated by

$$idf(t, D) = \log_2 \frac{N}{df}$$

Where N = 3, total 3 documents

df = frequency of term in all documents.

For **mid** keyword

df is 2.

N is 3.

$$= \log_2 \frac{3}{2}$$

idf	0.585	1.585	0.585	1.585	1.585

TF.IDF

Then TF.IDF can be calculated by multiplying Term frequency by Inverse document frequency

For **mid** keyword in **document D1**

tf*idf = 1 * 0.585 = 0.585

tf.idf	mid	term1	marks	term2	total
D1	0.585	1.585	0.585	0	0
D2	0.585	0	0.585	1.585	0
D3	0	0	0	0	1.585

Text Data Example (1)

- TRUMP MAKES BID FOR CONTROL OF RESORTS Casino owner and real estate Donald Trump has offered to acquire all Class B common shares of Resorts International Inc, a spokesman for Trump said. The estate of late Resorts chairman James M. Crosby owns 340,783 of the 752,297 Class B shares. Resorts also has about 6,432,000 Class A common shares outstanding. Each Class B share has 100 times the voting power of a Class A share, giving the Class B stock about 93 pct of Resorts' voting power.

Original text

- [RESORTS:0.624] [CLASS:0.487] [TRUMP:0.367] [VOTING:0.171] [ESTATE:0.166] [POWER:0.134] [CROSBY:0.134] [CASINO:0.119] [DEVELOPER:0.118] [SHARES:0.117] [OWNER:0.102] [DONALD:0.097] [COMMON:0.093] [GIVING:0.081] [OWNS:0.080] [MAKES:0.078] [TIMES:0.075] [SHARE:0.072] [JAMES:0.070] [REAL:0.068] [CONTROL:0.065] [ACQUIRE:0.064] [OFFERED:0.063] [BID:0.063] [LATE:0.062] [OUTSTANDING:0.056] [SPOKESMAN:0.049] [CHAIRMAN:0.049] [INTERNATIONAL:0.041] [STOCK:0.035] [YORK:0.035] [PCT:0.022] [MARCH:0.011]

Bag-of-Words representation (high dimensional sparse vector)

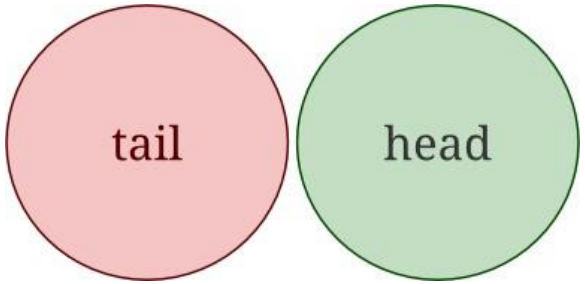
Information Retrieval (IR) Performance Metrics

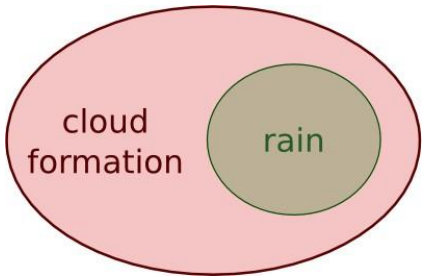
Recall : Number of relevant documents retrieved by a search / Total number of existing relevant documents

Precision: Number of relevant documents retrieved by a search / Total number of documents retrieved by that search

F-score: Single measure that combines precision and recall to compare different result sets

Understanding Conditional Probability (1)



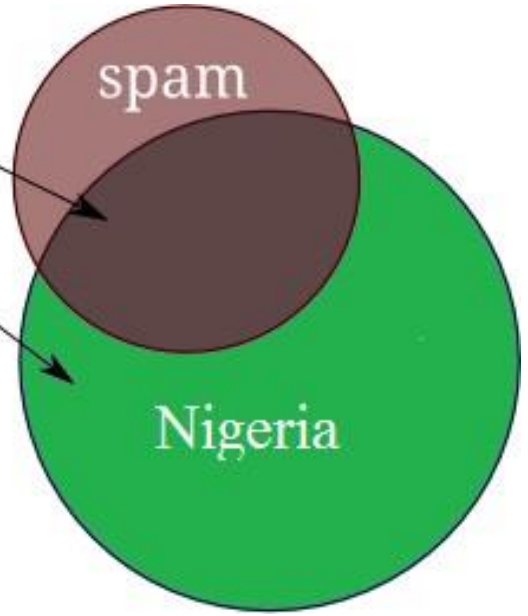


Understanding Conditional Probability (2)

- Conditional probability: to measure the degree of dependence

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$= \frac{P(B|A)P(A)}{P(B)}$$



Understanding Conditional Probability (3)

- The ***conditional probability*** of an event *B* is the probability that the event will occur given the knowledge that an event *A* has already occurred.
- This probability is written *P(B|A)*, notation for the *probability of B given A*.
- If events *A* and *B* are *independent* (where event *A* has no effect on the probability of event *B*), the conditional probability of event *B* given event *A* is simply the probability of event *B*, that is *P(B)*.

Understanding Conditional Probability (4)

- If events *A* and *B* are not independent, then the probability of the *intersection of A and B* (the probability that both events occur) is defined by *P(A and B) = P(A)P(B|A).*
- From this definition, the conditional probability *P(B|A)* is easily obtained by dividing by *P(A)*:

$$P(B|A) = \frac{P(A \text{ and } B)}{P(A)}$$

Which one is given? Conditional or joint probability?

Example (1)

- In a card game, suppose a player needs to draw two cards of the same suit in order to win. Of the 52 cards, there are 13 cards in each suit. Suppose first the player draws a heart. Now the player wishes to draw a second heart.
- ?

Example (1)

- In a card game, suppose a player needs to draw two cards of the same suit in order to win. Of the 52 cards, there are 13 cards in each suit. Suppose first the player draws a heart. Now the player wishes to draw a second heart.
- Since one heart has already been chosen, there are now 12 hearts remaining in a deck of 51 cards. So the conditional probability *P(Draw second heart|First card a heart)* = 12/51.
- Joint probability is given

Example (2)

- Suppose an individual applying to a college determines that he has an 80% chance of being accepted, and he knows that dormitory housing will only be provided for 60% of all of the accepted students. The chance of the student being accepted *and* receiving dormitory housing is defined by
- *?*

Example (2)

- Suppose an individual applying to a college determines that he has an 80% chance of being accepted, and he knows that dormitory housing will only be provided for 60% of all of the accepted students. The chance of the student being accepted *and* receiving dormitory housing is defined by
- *P(Accepted and Dormitory Housing) = P(Dormitory Housing|Accepted)P(Accepted)* = (0.60)*(0.80) = 0.48
- Cond. Probability is given

Question

- A jar contains black and white marbles. Two marbles are chosen without replacement. The probability of selecting a black marble and then a white marble is 0.34, and the probability of selecting a black marble on the first draw is 0.47. What is the probability of selecting a white marble on the second draw, given that the first marble drawn was black?

- The probability that it is Friday and that a student is absent is 0.03. Since there are 5 school days in a week, the probability that it is Friday is 0.2. What is the probability that a student is absent given that today is Friday?

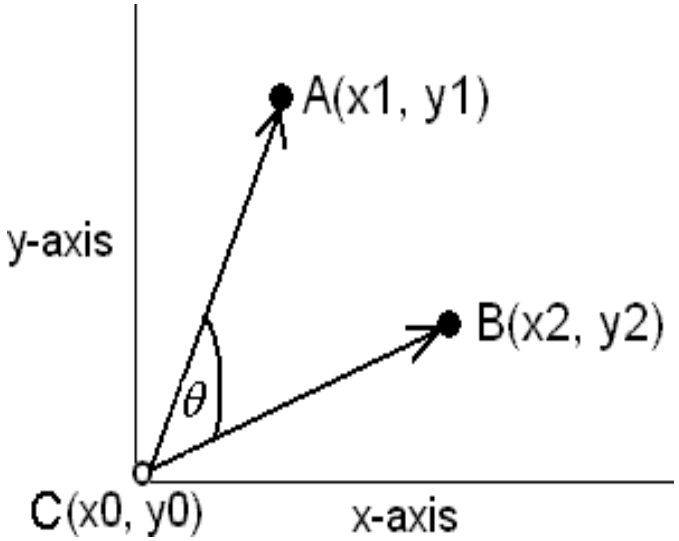
- P(absent/Friday) = p(Friday and absent)/p(absent) =0.03/0.2

Similarity Metrics in TextMining

- D_i: a document vector
- Document (and query) vectors are normalized
- Similarity: a real number between 0 and 1
- Applications:
  - Search and information retrieval: Similarity between documents and queries
  - Distance between two documents (vectors) in document clustering algorithms

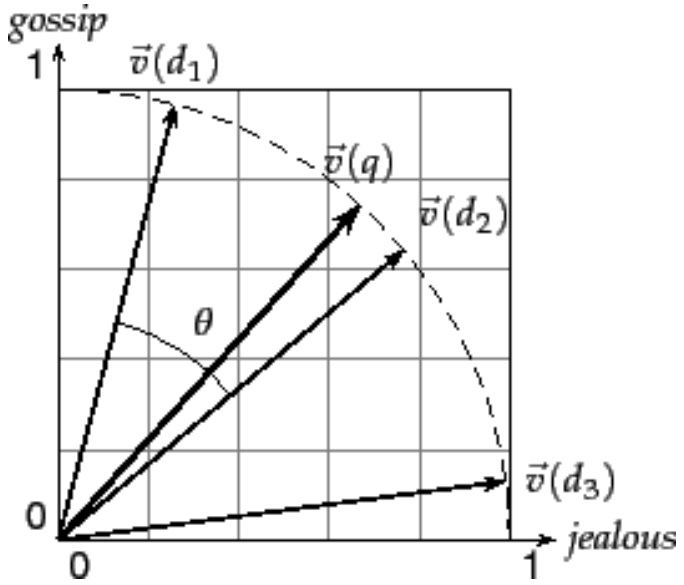
Cosine Similarity

- Cosine similarity is a measure of similarity between two vectors by measuring the cosine of the angle between them.
- The result of the Cosine function is equal to 1 when the angle is 0, and it is less than 1 when the angle is of any other value.
- As the angle between the vectors shortens, the cosine angle approaches 1, meaning that the two vectors are getting closer and the similarity of whatever is represented by the vectors increases.



$$Sim(D_1, D_2) = \frac{\sum x_{1i} x_{2i}}{\sqrt{\sum_j x_j^2}\sqrt{\sum_k x_k^2}}$$

Document (Length) Normalization



$$\text{sim}(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{|\vec{V}(d_1)||\vec{V}(d_2)|},$$

KL Divergence

- Kullback-Leibler (KL) divergence measures how much one probability distribution is different from another.

$$KL(p, q) = \sum_x p(x) \log \frac{p(x)}{q(x)}.$$

- Is KL-divergence a metric?
- Applications: can be used for scoring terms, language models, etc
- Rarely we use for estimating distance between documents

Text Mining Tools

- Install and explore Python NLTK toolkit for text mining

Lambton College
School of Computer Studies

Lambton College

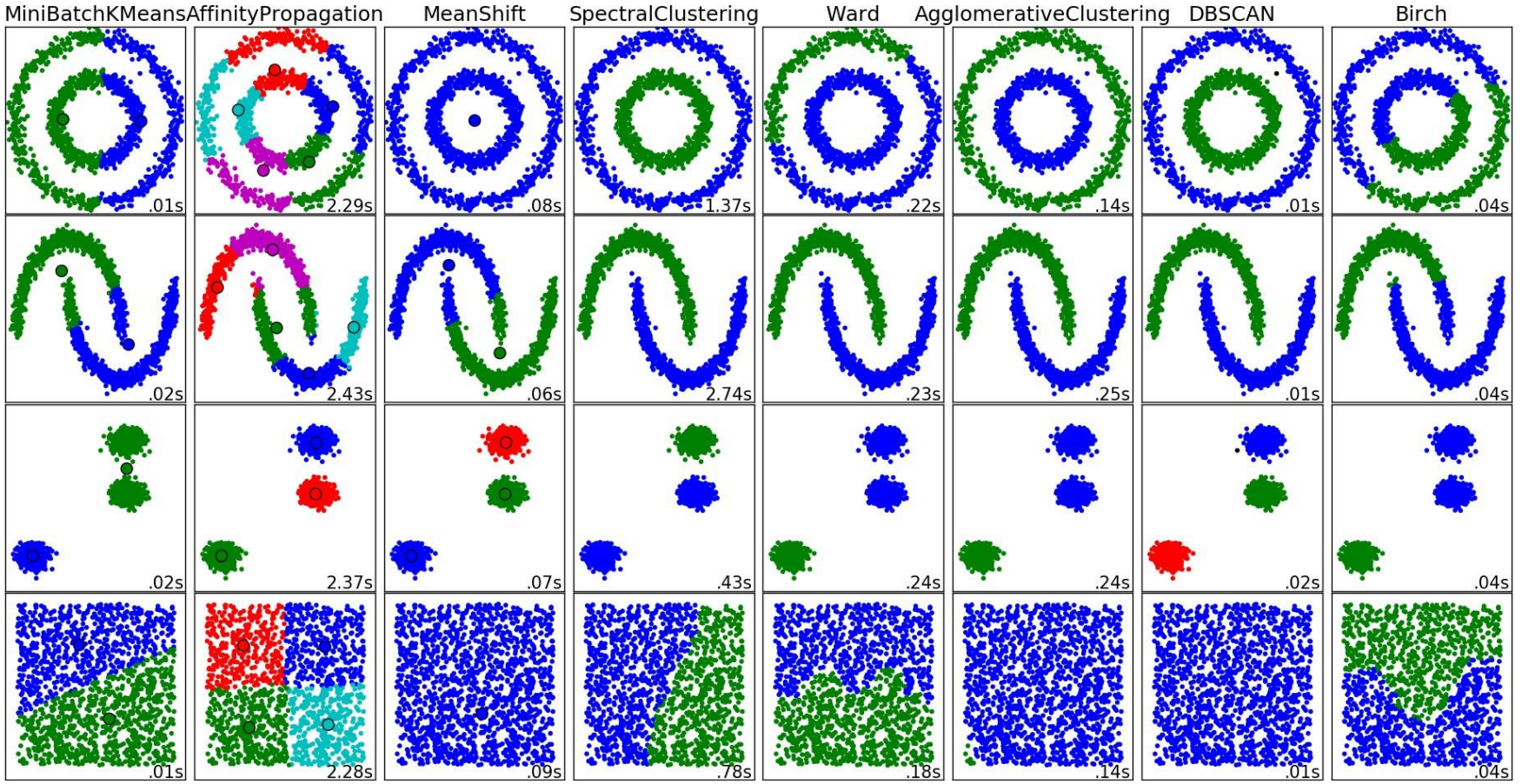
Lecture 5

CBD-3335 Data Mining and Analysis

Topics

- Unsupervised Learning
- Taxonomy of Clustering
- Applications of NLP
- NLP components
- Text processing

Unsupervised learning (Clustering)



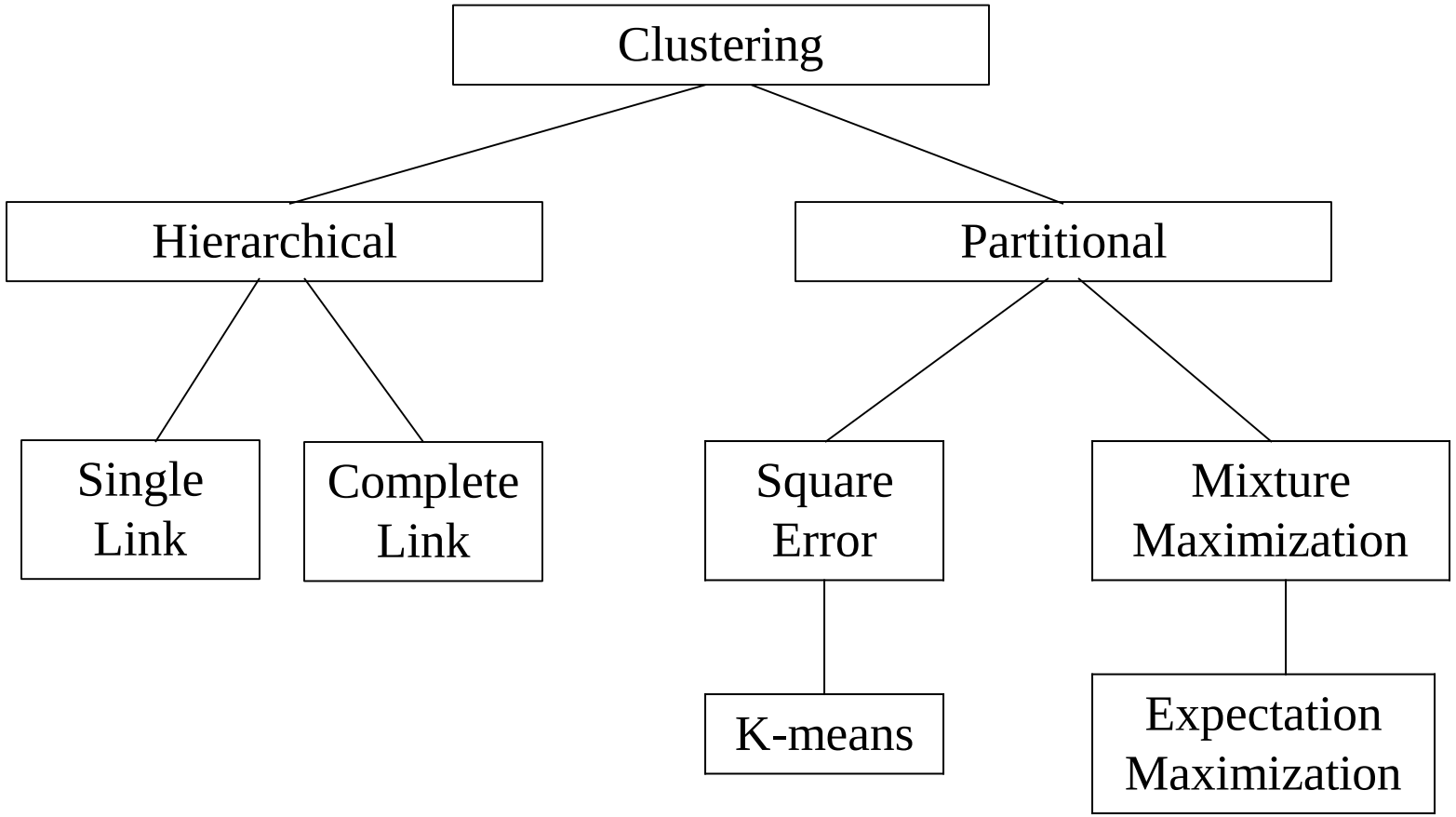
What is clustering?

- A way of grouping data samples that are ***similar*** in some way - according to some criteria that you pick

- A form of ***unsupervised learning*** – you generally don't have examples demonstrating how the data *should* be grouped together

- So, it's a method of ***data exploration*** – a way of looking for patterns or structure in the data that are of interest

Taxonomy of Clustering Methods



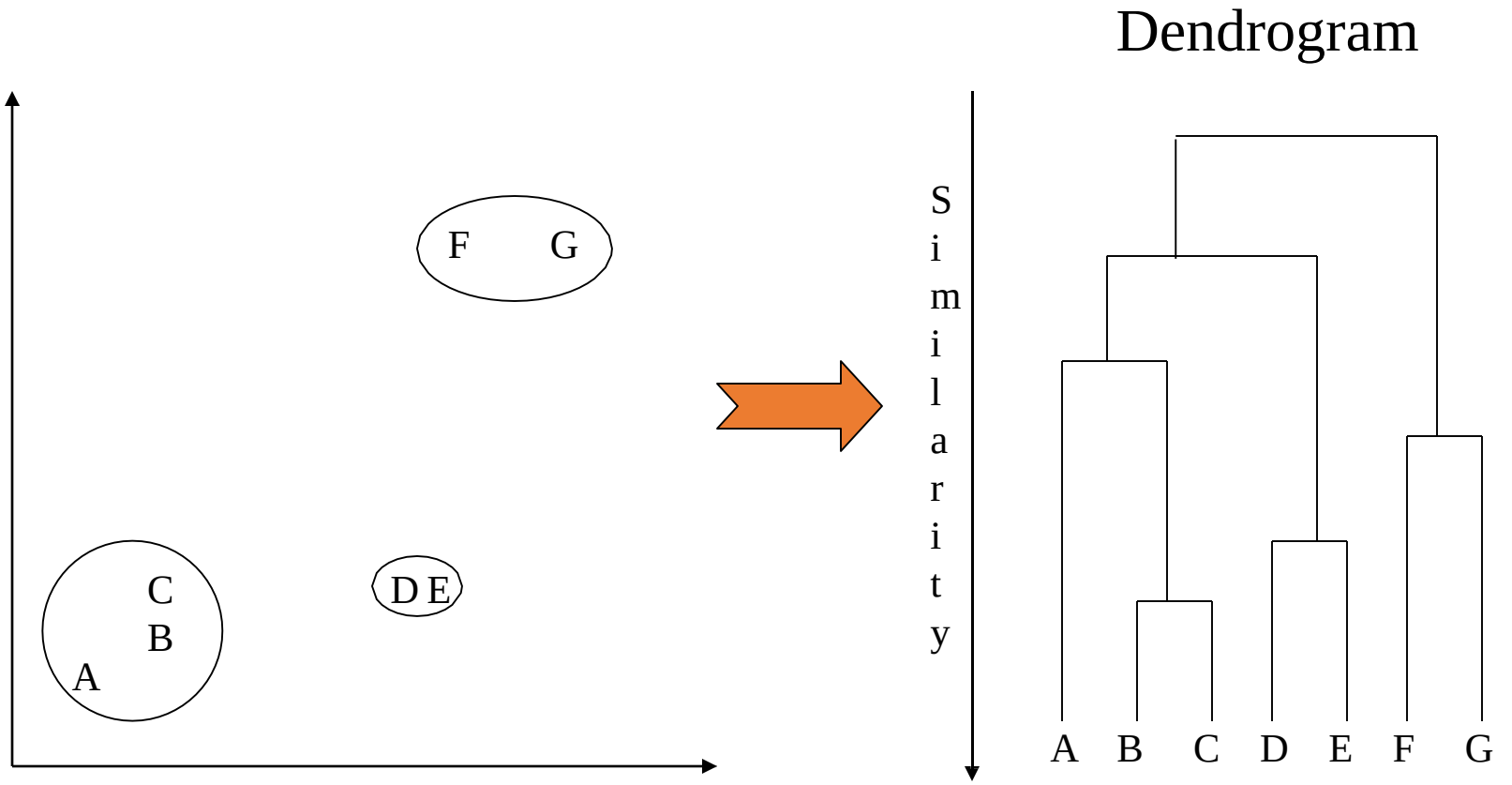
Direct Clustering Method

- *Direct clustering* methods require a specification of the number of clusters, *k*, desired.
- A *clustering evaluation function* assigns a real-value quality measure to a clustering.
- The number of clusters can be determined automatically by explicitly generating clusterings for multiple values of *k* and choosing the best result according to a clustering evaluation function.

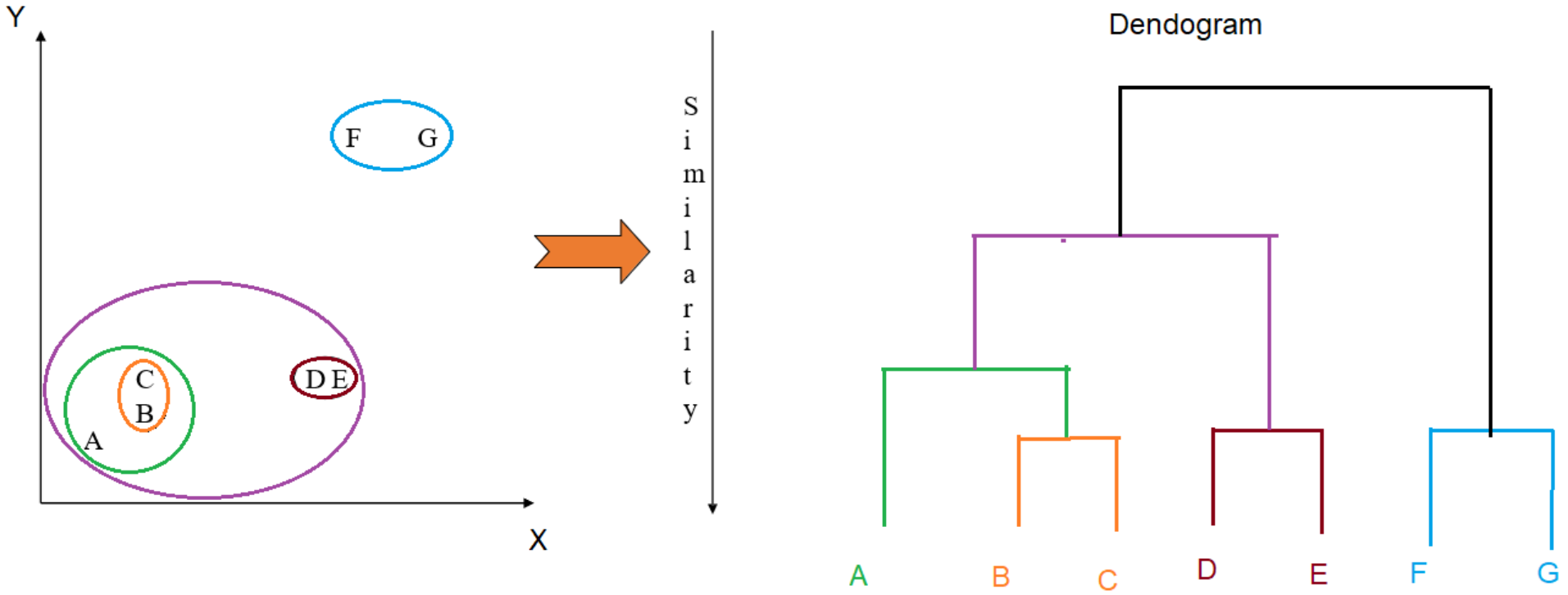
Technique Characteristics

- Agglomerative vs Divisive
  - *Agglomerative*: each instance is its own cluster and the algorithm merges clusters
  - *Divisive*: begins with all instances in one cluster and divides it up
- Hard vs Fuzzy
  - Hard clustering assigns each instance to one cluster whereas in fuzzy clustering assigns degree of membership

Hierarchical Clustering



Hierarchical Clustering



Similarity: Distance between the points

DENDROGRAM

- **A** tree like structure which represents hierarchical technique.
  - ✓ Leaf- Individual.
  - ✓ Root – One cluster.
- A cluster at level 1, is the merger of its child cluster at level i + 1.

Hierarchical Agglomerative Clustering (HAC)

- Assumes a *similarity function* for determining the similarity of two instances.
- Starts with all instances in a separate cluster and then repeatedly joins the two clusters that are most similar until there is only one cluster.
- The history of merging forms a binary tree or hierarchy.

HAC Algorithm

Start with all instances in their own cluster.
Until there is only one cluster:
    Among the current clusters, determine the two clusters, c_i and c_j, that are most similar.
    Replace c_i and c_j with a single cluster $c_i \cup c_j$

Hierarchical Algorithms

- Single-link
  - Distance between two clusters is estimated by *minimum* of distances between all instances
  - Produces (sometimes too) large clusters (chaining)

- Complete-link
  - Distance between two clusters is estimated by *maximum* of all distances between instances in the clusters
  - Tightly bound, compact clusters
  - Often more useful in practice

Single Link Agglomerative Clustering

- Use minimum similarity of pairs:

$$sim(c_i, c_j) = \min_{x \in c_i, y \in c_j} sim(x, y)$$

- Can result in "long and thin" clusters due to *chaining effect*.
  - Appropriate in some domains, such as clustering islands.

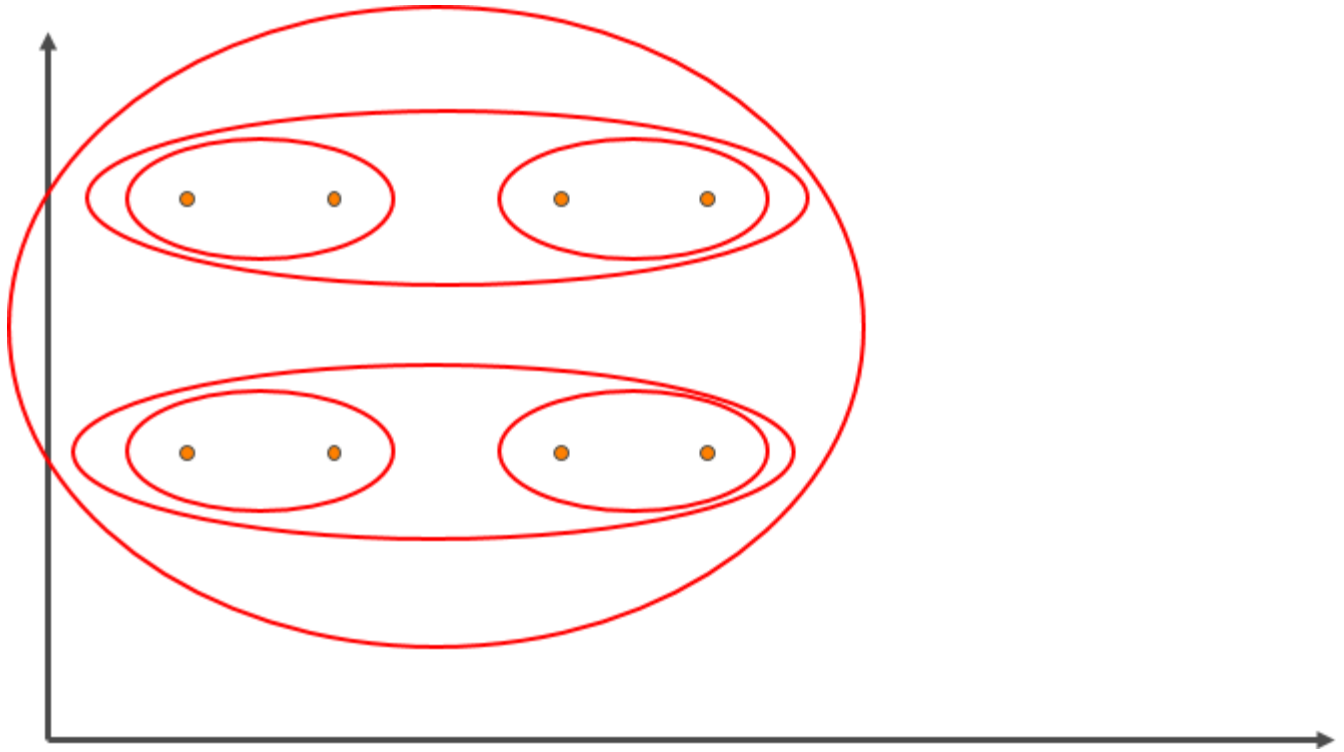
Complete Link Agglomerative Clustering

- Use maximum similarity of pairs:

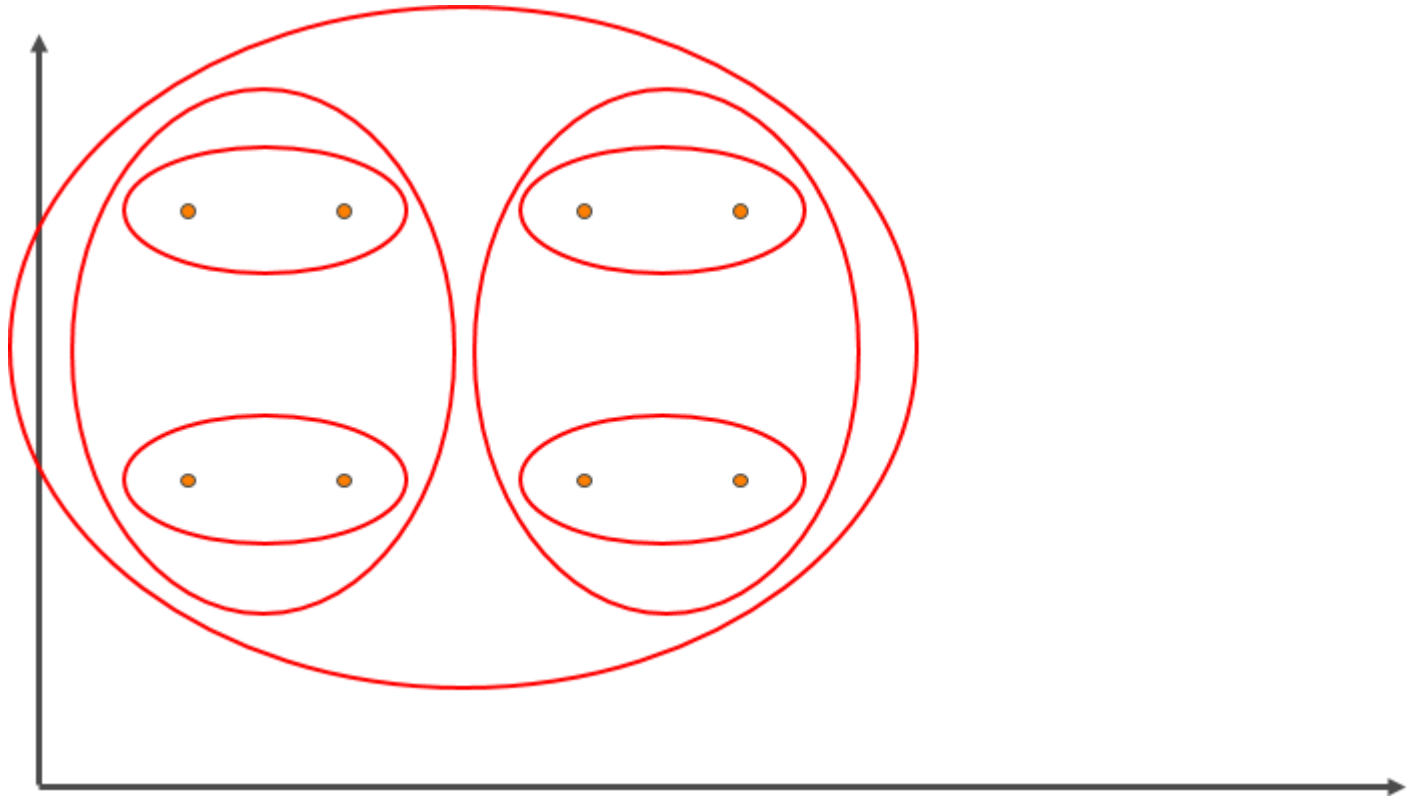
$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

- Makes more “tight,” spherical clusters that are typically preferable.

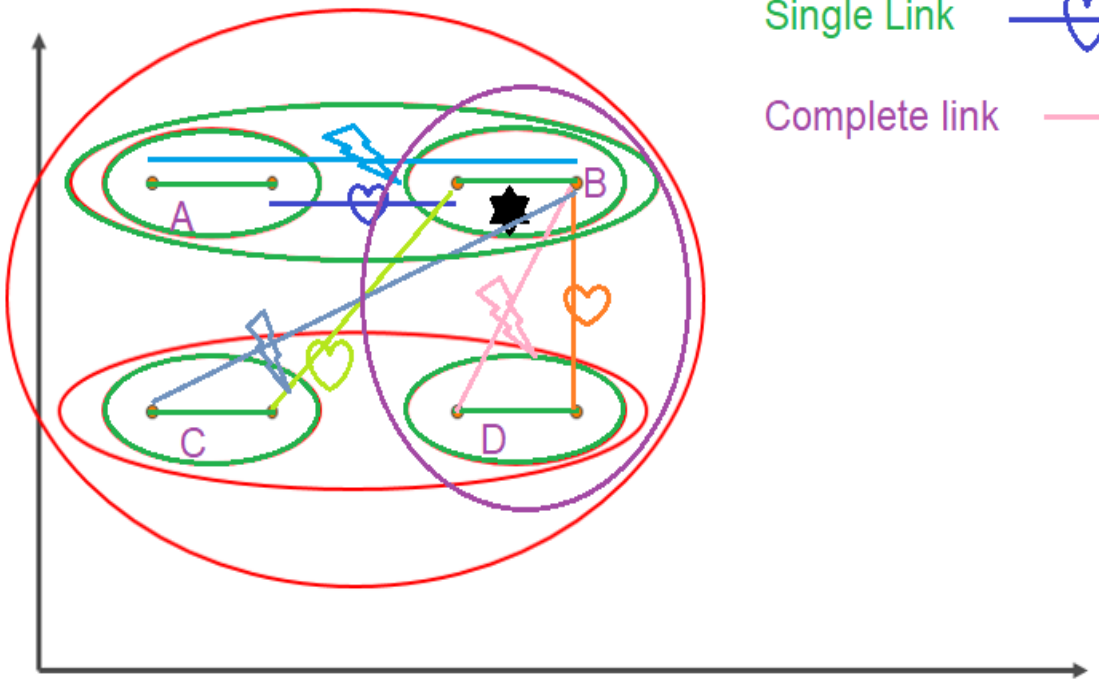
Single Link Example



Complete Link Agglomerative Clustering



Link Agglomerative Clustering



Single Link

Complete link

Distance: should always minimum for new cluster formation

Single link
A-B: Blue heart
B-C: Green heart
B-D: Orange heart

Complte Link
A-B: Blue Bolt
B-C: Grey bolt
B-D: Pink bolt

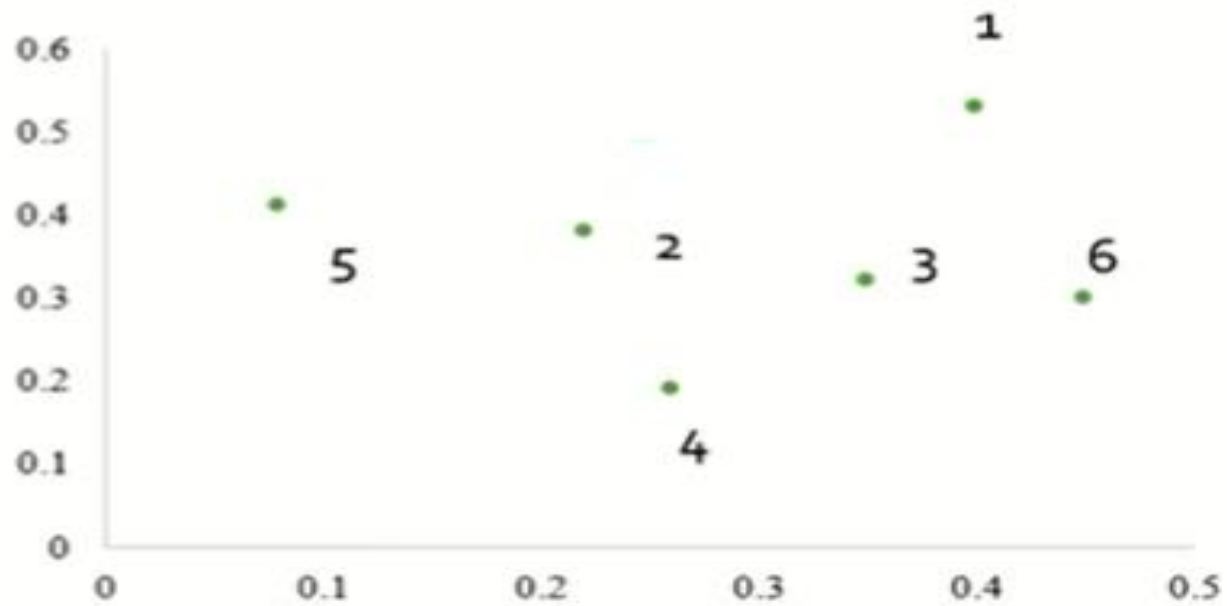
Single-link Clustering

- Find the clusters using single link technique. Use Euclidean distance, and draw the dendrogram.

	X	Y
P1	0.40	0.53
P2	0.22	0.38
P3	0.35	0.32
P4	0.26	0.19
P5	0.08	0.41
P6	0.45	0.30

Single-link Clustering

	X	Y
P1	0.40	0.53
P2	0.22	0.38
P3	0.35	0.32
P4	0.26	0.19
P5	0.08	0.41
P6	0.45	0.30



Single-link Clustering

- Calculate Euclidean distance, create the distance matrix.

$$\text{Distance } [(x,y), (a,b)] = \sqrt{(x-a)^2+(x-b)^2}$$

$$\text{Distance (P1,P2)} = \sqrt{(0.40-0.22)^2+(0.53-0.38)^2}$$

$$\text{(0.40,0.53), (0.22,0.38)} = \sqrt{(0.18)^2+(0.15)^2}$$

$$= \sqrt{0.0324 + 0.0225}$$

$$= \sqrt{0.0549}$$

$$= 0.23$$

Single-link Clustering

- The distance matrix is

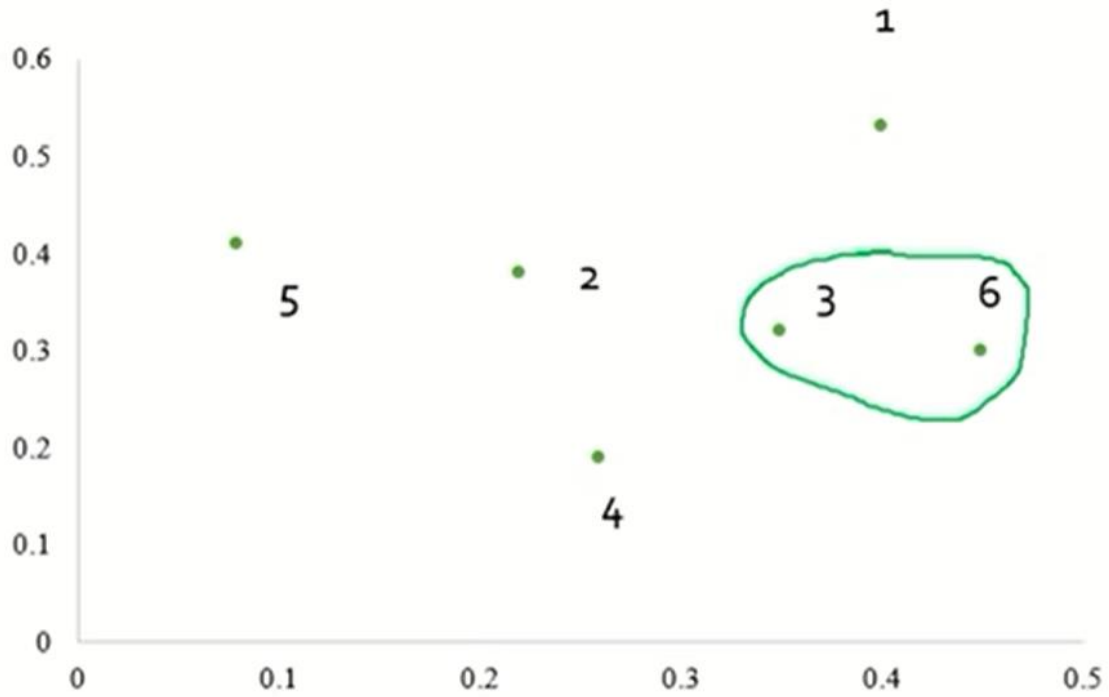
	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.23	0				
P3	0.22	0.15	0			
P4	0.37	0.20	0.15	0		
P5	0.34	0.14	0.28	0.29	0	
P6	0.23	0.25	0.11	0.22	0.39	0

Single-link Clustering

- The distance matrix is

	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.24	0				
P3	0.22	0.15	0			
P4	0.37	0.20	0.15	0		
P5	0.34	0.14	0.28	0.29	0	
P6	0.23	0.25	0.11	0.22	0.39	0

Single-link Clustering



Single-link Clustering

- To update the distance matrix MIN[dist(P3,P6),P1)]
- MIN(dist(P3,P1), (P6,P1))

  = min[(0.22,0.23)]

  = 0.22

- To update the distance matrix MIN[dist(P3,P6),P2)]
- MIN(dist(P3,P2), (P6,P2))

  = min[(0.15,0.25)]

  = 0.15

Single-link Clustering

- The distance matrix is

	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.23	0				
P3	0.22	0.15	0			
P4	0.37	0.20	0.15	0		
P5	0.34	0.14	0.28	0.29	0	
P6	0.23	0.25	0.11	0.22	0.39	0

Single-link Clustering

- The distance matrix is

	P1	P2	P3	P4	P5	P6
P1	0					
P2	0.23	0				
P3	0.22	0.15	0			
P4	0.37	0.20	0.15	0		
P5	0.34	0.14	0.28	0.29	0	
P6	0.23	0.25	0.11	0.22	0.39	0

- The updated distance matrix for cluster P3, P6

	P1	P2	P3,P6	P4	P5
P1	0				
P2	0.23	0			
P3,P6	0.22	0.15	0		
P4	0.37	0.20	0.15	0	
P5	0.34	0.14	0.28	0.29	0

Single-link Clustering

- The updated distance matrix for cluster P3, P6

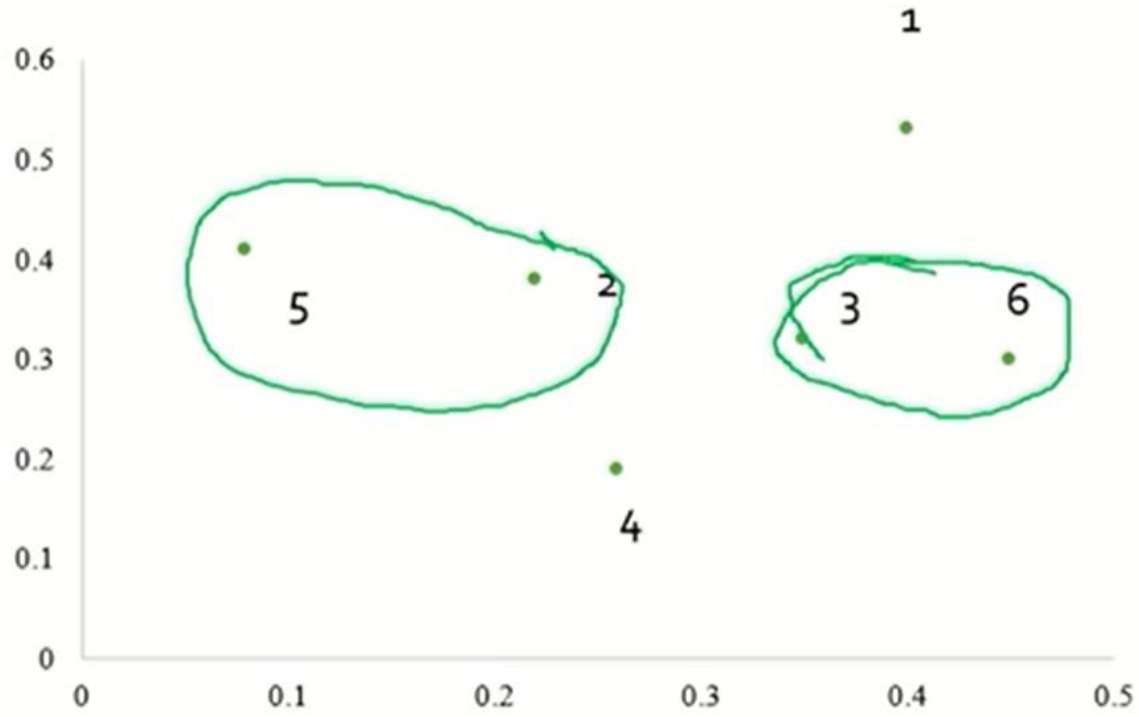
	P1	P2	P3,P6	P4	P5
P1	0				
P2	0.23	0			
P3,P6	0.22	0.15	0		
P4	0.37	0.20	0.15	0	
P5	0.34	0.14	0.28	0.29	0

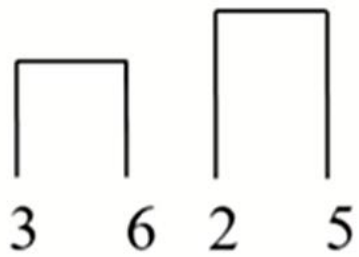
Single-link Clustering

- The distance matrix is

	P1	P2	P3,P6	P4	P5
P1	0				
P2	0.23	0			
P3,P6	0.22	0.15	0		
P4	0.37	0.20	0.15	0	
P5	0.34	0.14	0.28	0.29	0

Single-link Clustering



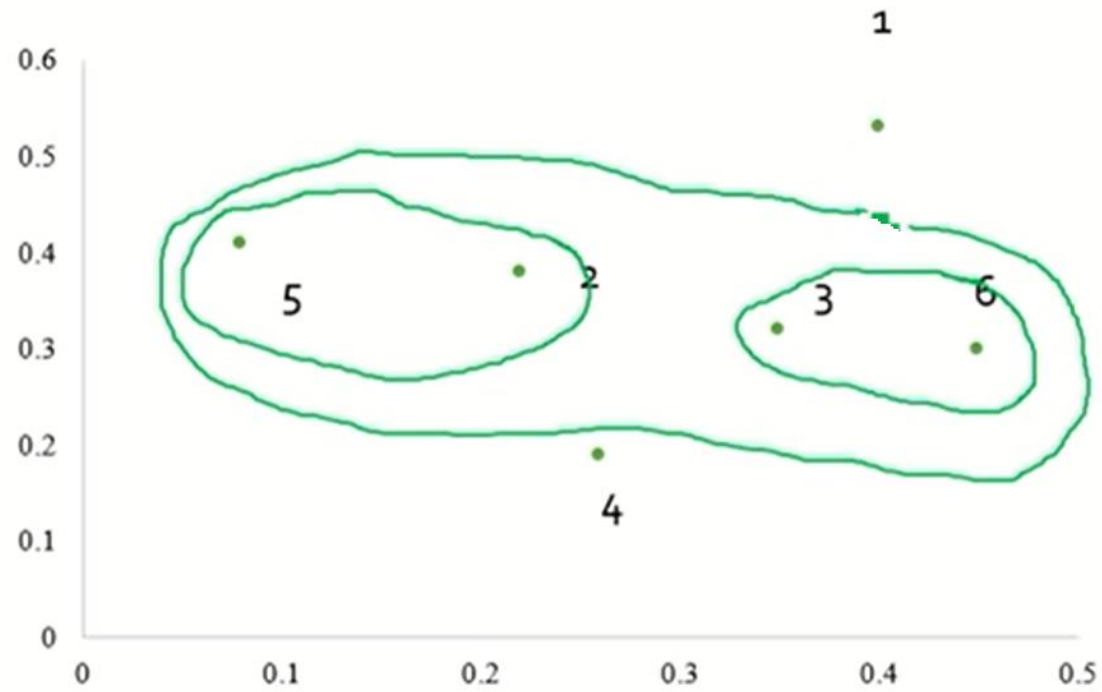


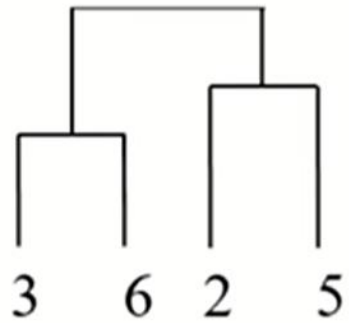
Single-link Clustering

- The distance matrix is

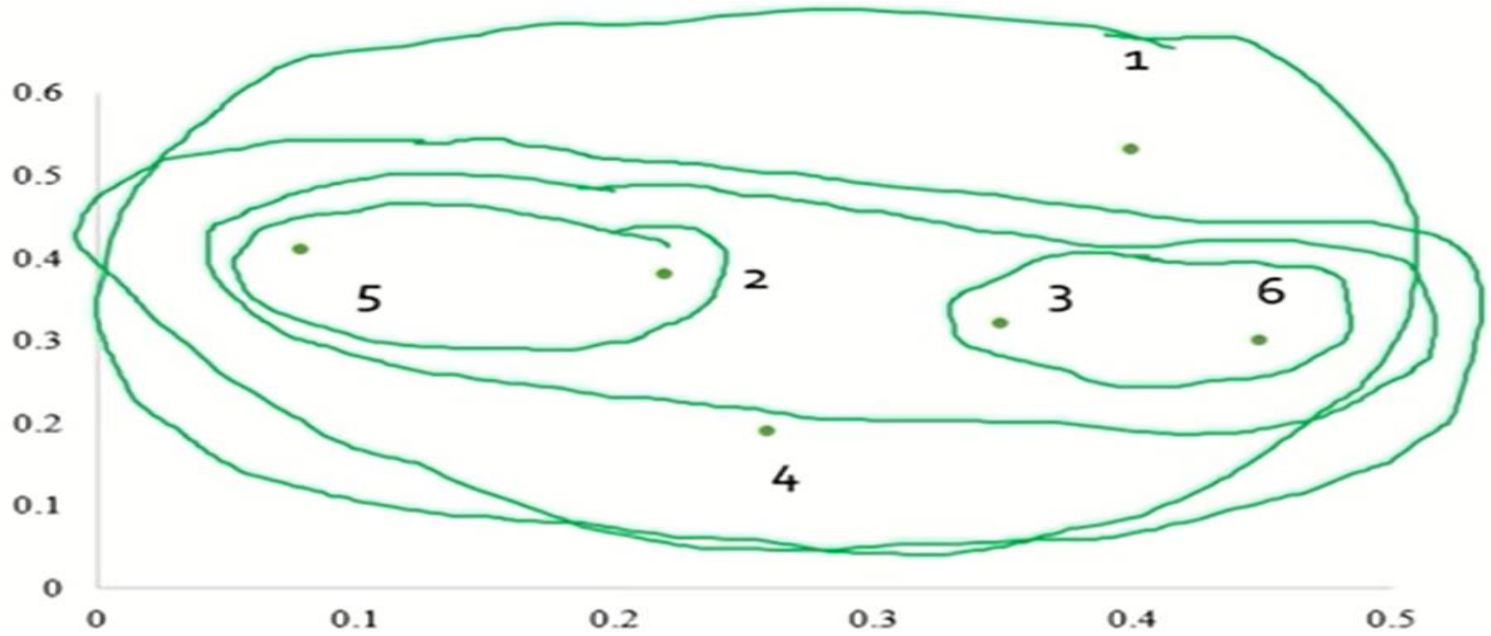
	P1	P2,P5	P3,P6	P4
P1	0			
P2,P5	0.23	0		
P3,P6	0.22	0.15	0	
P4	0.37	0.20	0.15	0

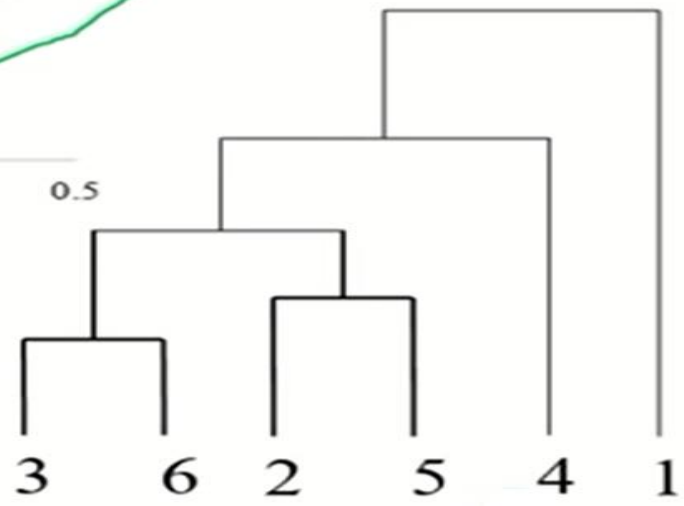
Single-link Clustering





Single-link Clustering





Computational Complexity

- In the first iteration, all HAC methods need to compute similarity of all pairs of n individual instances which is $O(n^2)$.
- In each of the subsequent $n-2$ merging iterations, it must compute the distance between the most recently created cluster and all other existing clusters.

Partitional Clustering

- Output a single partition of the data into clusters
- Good for large data sets
- Determining the number of clusters is a major challenge

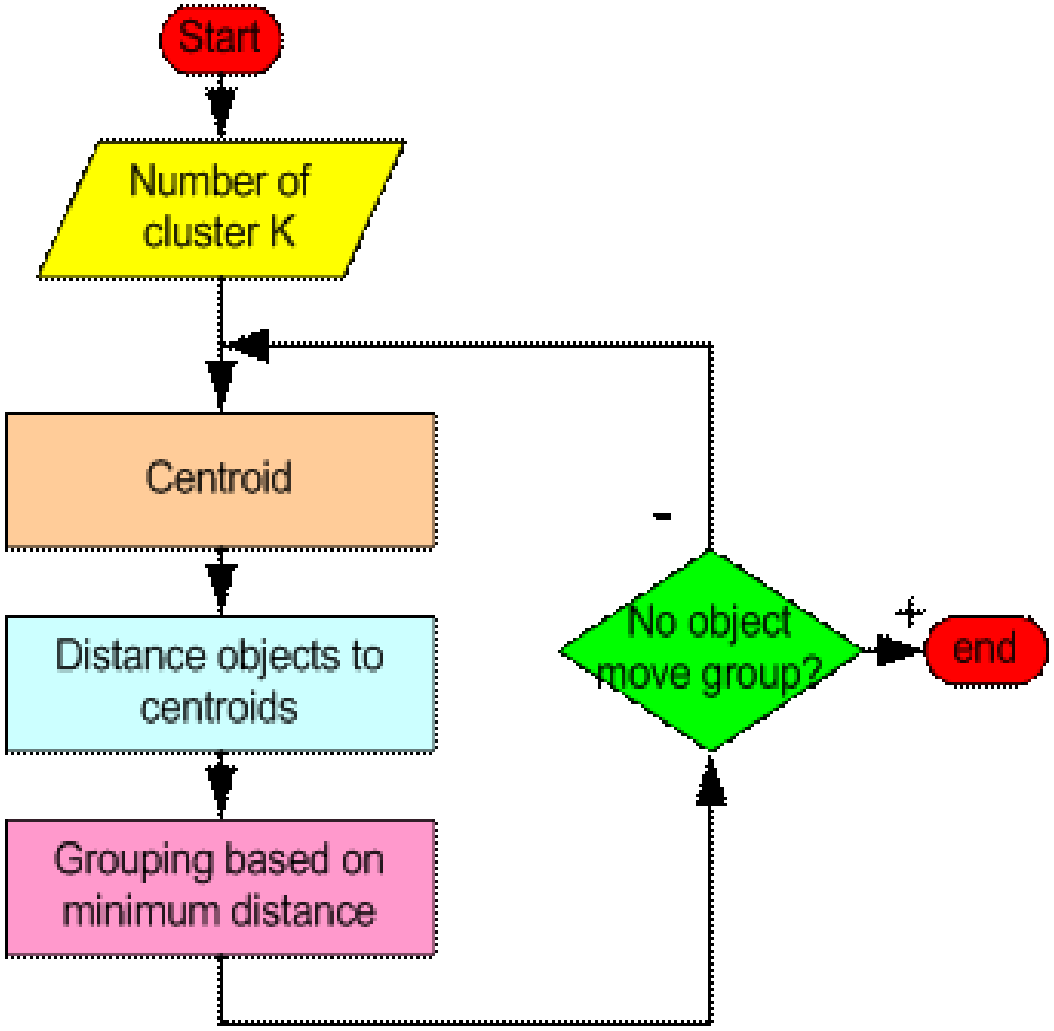
K-Means

- Assumes instances are real-valued vectors.
- Clusters based on *centroids*, *center of gravity*, or mean of points in a cluster, *c*:

$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

- Reassignment of instances to clusters is based on distance to the current cluster centroids.

How the K-Mean Clustering algorithm works?



Distance Metrics

- Euclidian distance (L_2 norm):

$$L_2(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

- L_1 norm:

$$L_1(x, y) = \sum_{i=1}^{m} |x_i - y_i|$$

- Cosine Similarity (transform to a distance by subtracting from 1):

$$1 - \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| \cdot |\vec{y}|}$$

K-Means Algorithm

- Let d be the distance measure between instances.
- Select k random instances $\{s_1, s_2, \ldots s_k\}$ as seeds.
- Until clustering converges or other stopping criterion:

  For each instance x_i:

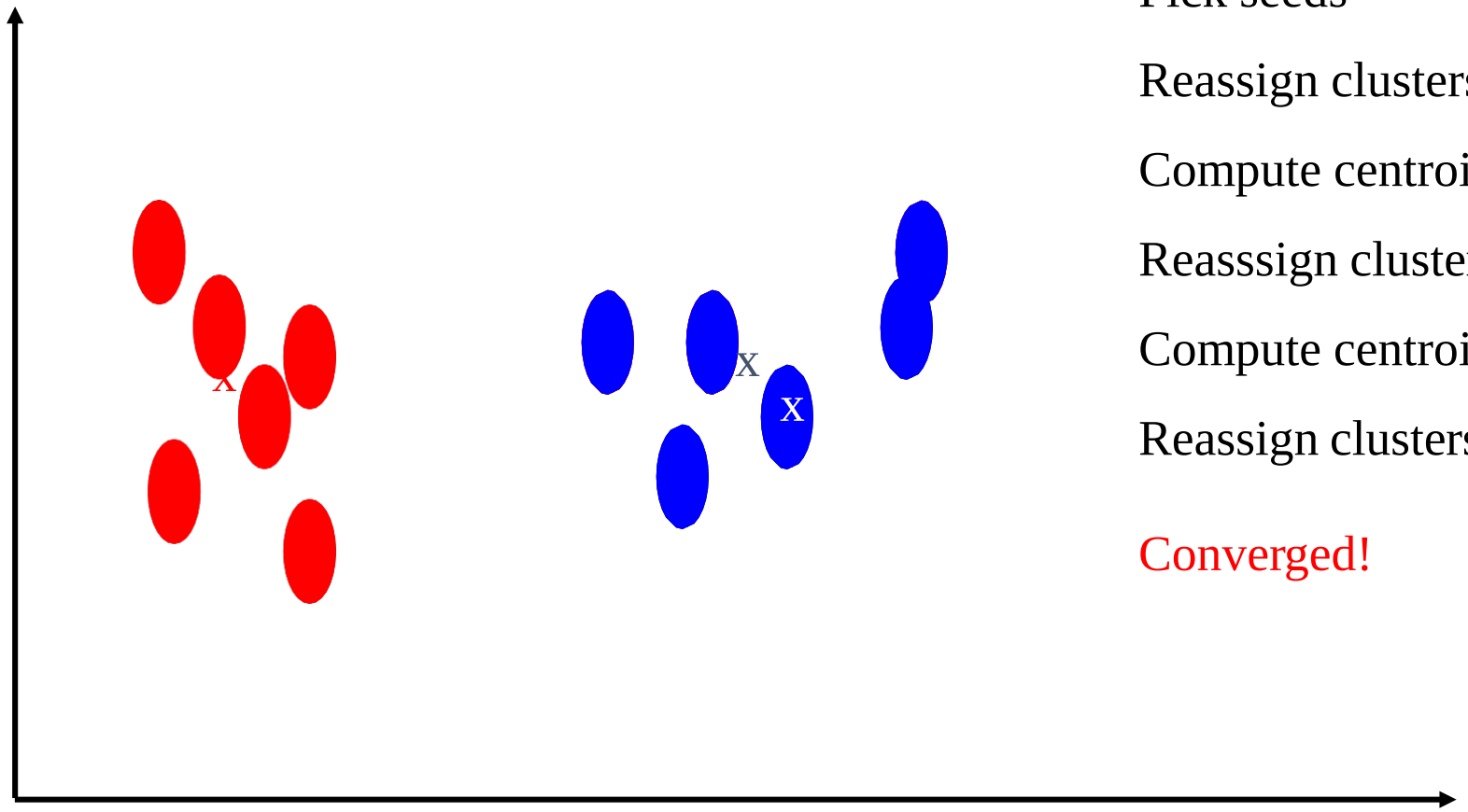
    Assign x_i to the cluster c_j such that $d(x_i, s_j)$ is minimal.

  (*Update the seeds to the centroid of each cluster*)

  For each cluster c_j

    $s_j = \mu(c_j)$

K Means Example
(K=2)



Pick seeds

Reassign clusters

Compute centroids

Reasssign clusters

Compute centroids

Reassign clusters

Converged!

Time Complexity

- Assume computing distance between two instances is O(*m*) where *m* is the dimensionality of the vectors.
- Reassigning clusters: O(*kn*) distance computations, or O(*knm*).
- Computing centroids: Each instance vector is added once to some centroid: O(*nm*).
- Assume these two steps are each done once for *I* iterations: O(*Iknm*).
- Linear in all relevant factors, assuming a fixed number of iterations, more efficient than $O(n^2)$ HAC.

K-Means Objective

- The objective of k-means is to minimize the total sum of the squared distance of every point to its corresponding cluster centroid.

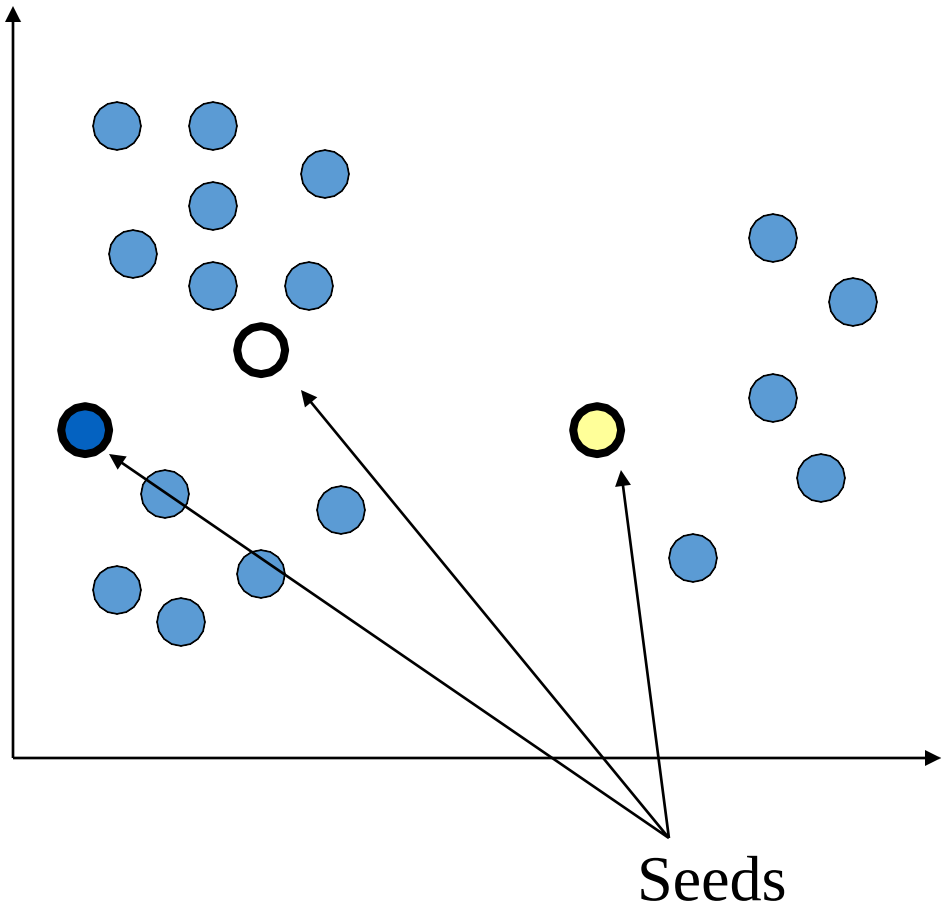
$$\sum_{l=1}^{K} \sum_{x_i \in X_l} \| x_i - \mu_l \|^2$$

- Finding the global optimum is NP-hard.
- The k-means algorithm is guaranteed to converge a local optimum.

Seed Choice

- Results can vary based on random seed selection.
- Some seeds can result in poor convergence rate, or convergence to sub-optimal clusterings.
- Select good seeds using a heuristic or the results of another method.

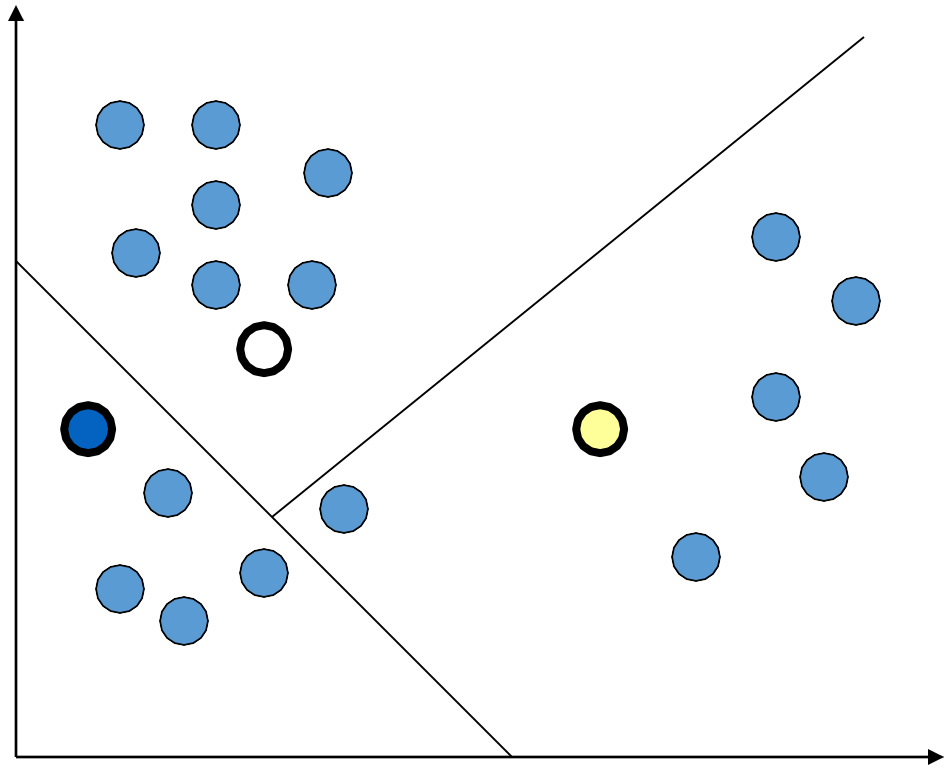
K-Means



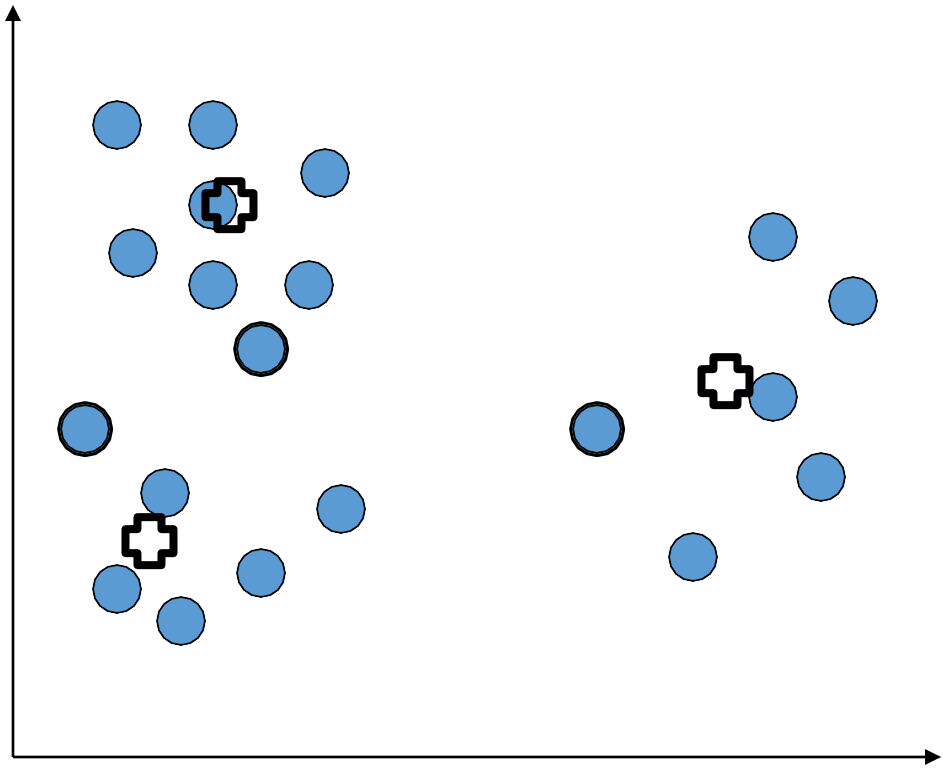
Predetermined number of clusters

Start with seed clusters of one element

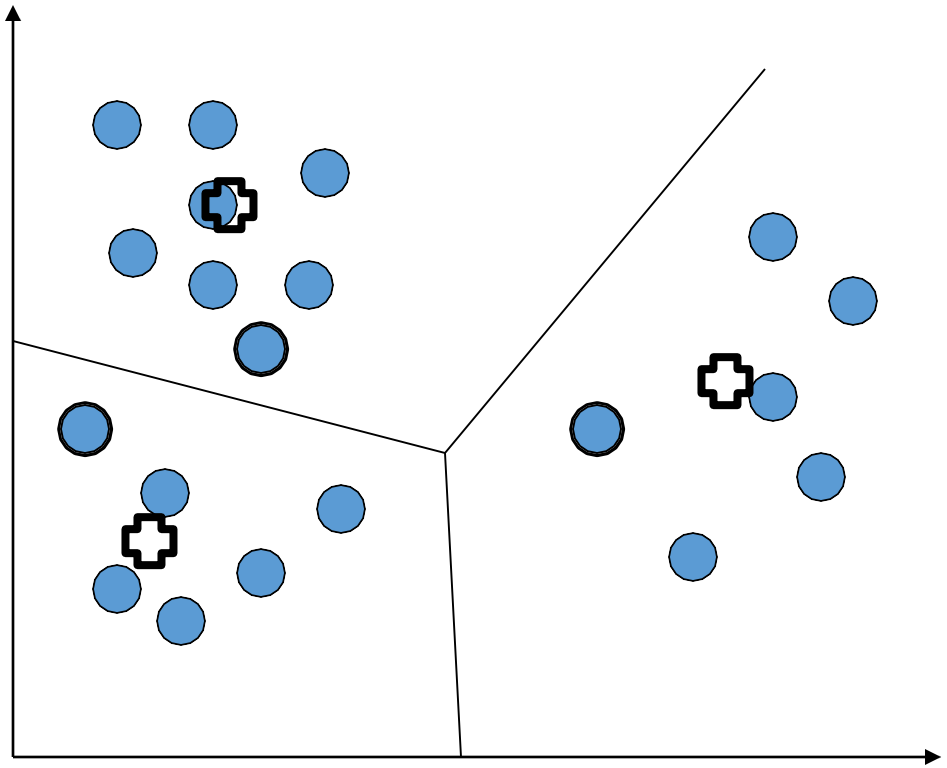
Assign Instances to Clusters



Find New Centroids



New Clusters



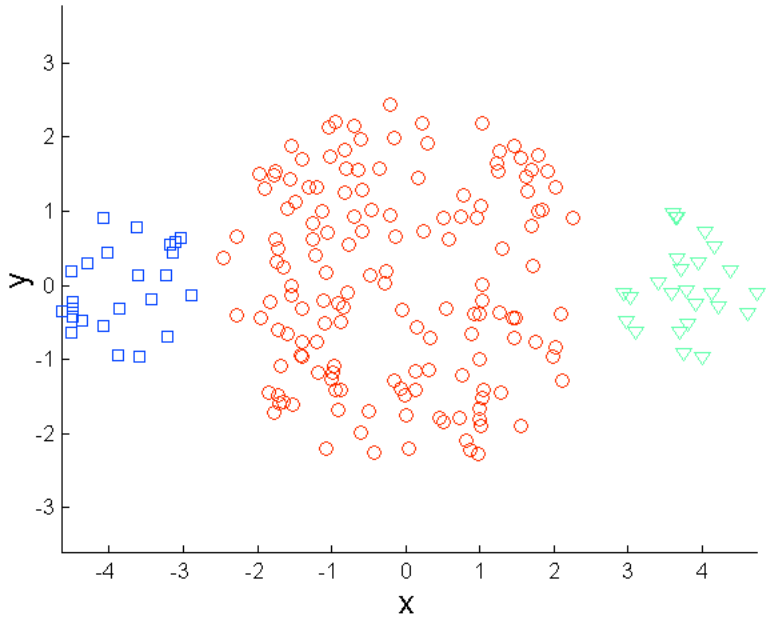
Discussion: k-means

- Applicable to fairly large data sets
- Sensitive to initial centers
  - Use other heuristics to find good initial centers
- Converges to a local optimum
- Specifying the number of centers very subjective

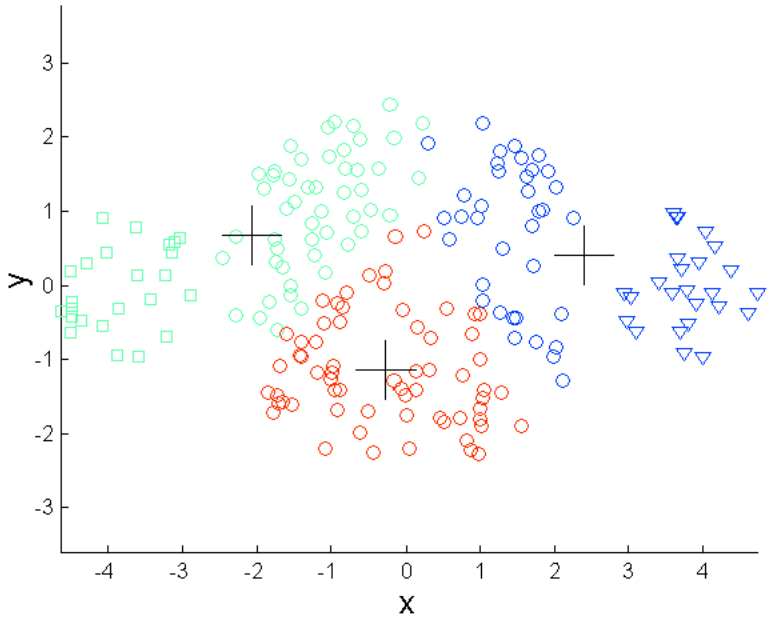
Limitations of K-means

- K-means has problems when clusters are of different
  - Sizes
  - Densities
  - Non-globular shapes
- K-means has problems when the data contains outliers.

Limitations of K-means: Differing Sizes

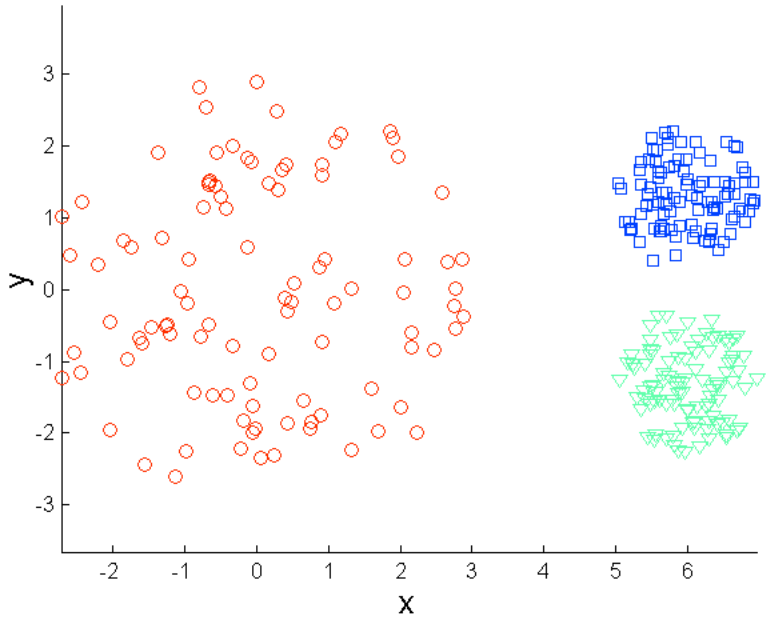


Original Points

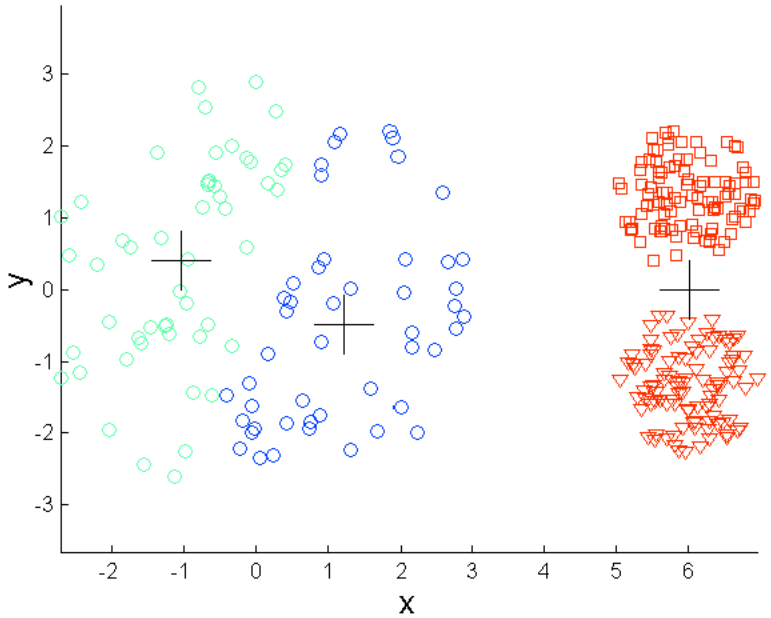


K-means (3 Clusters)

Limitations of K-means: Differing Density

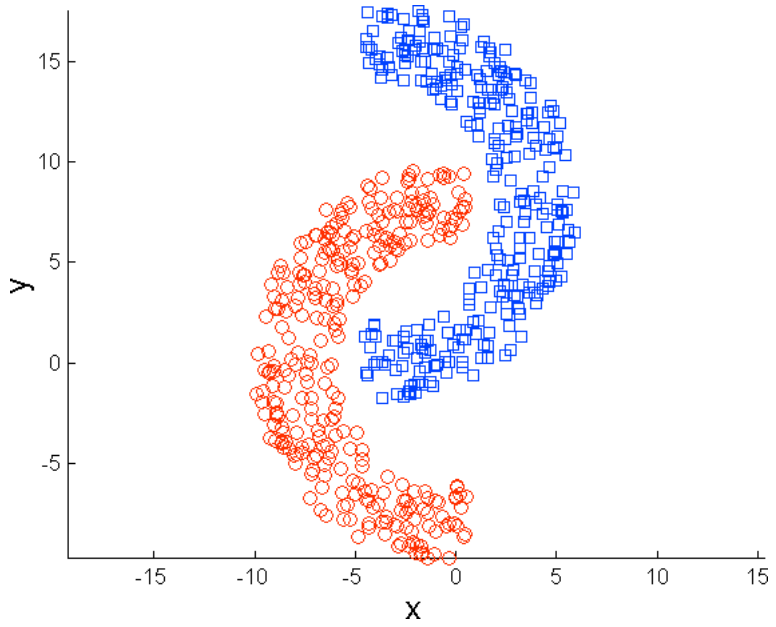


Original Points

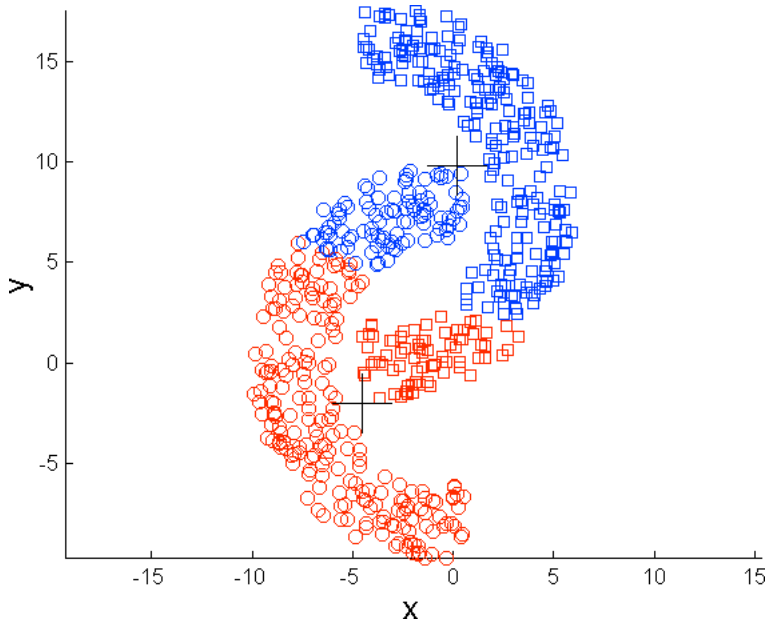


K-means (3 Clusters)

Limitations of K-means: Non-globular Shapes

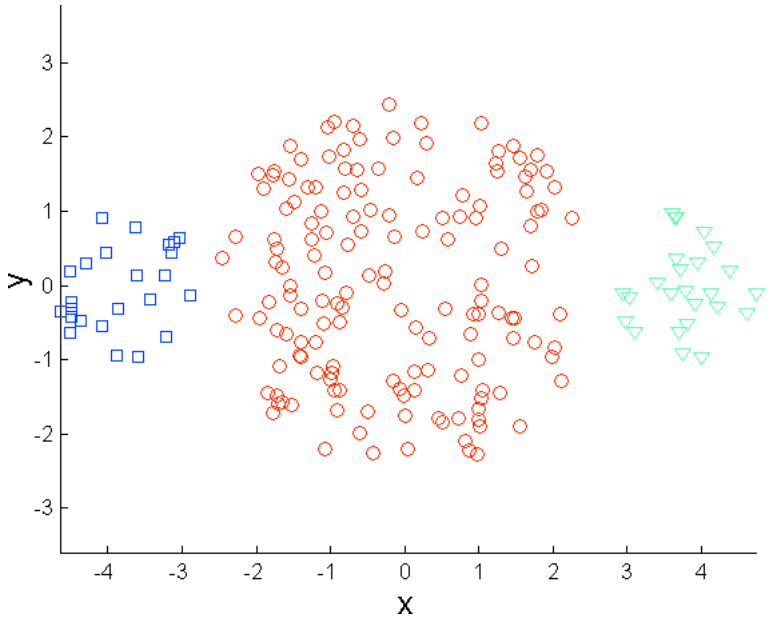


Original Points

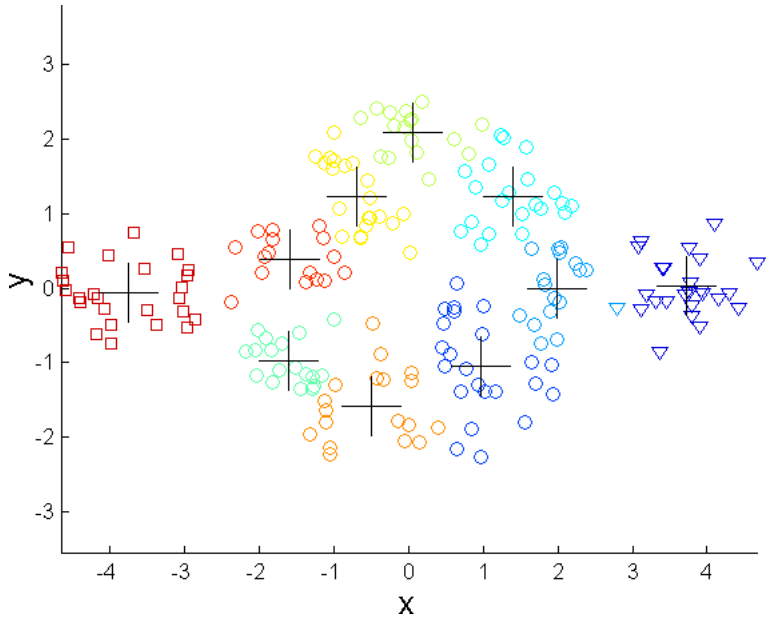


K-means (2 Clusters)

Overcoming K-means Limitations



Original Points

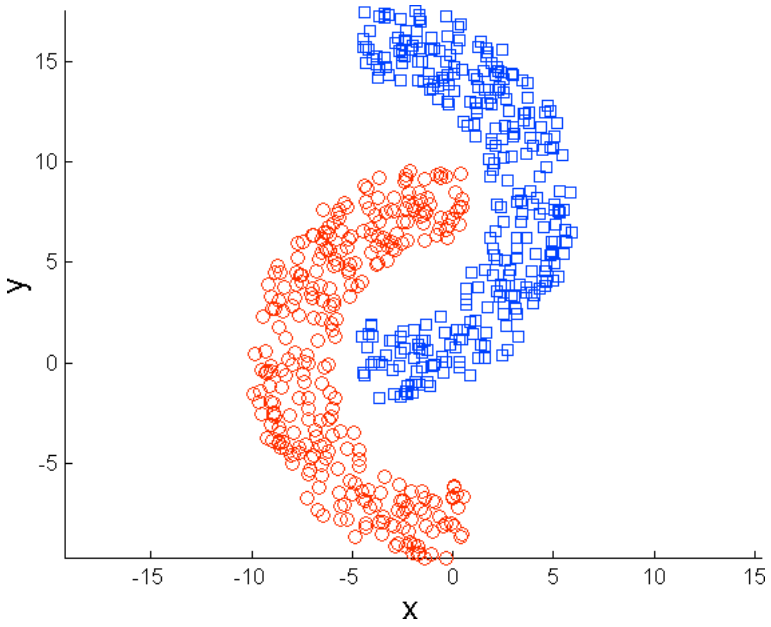


K-means Clusters

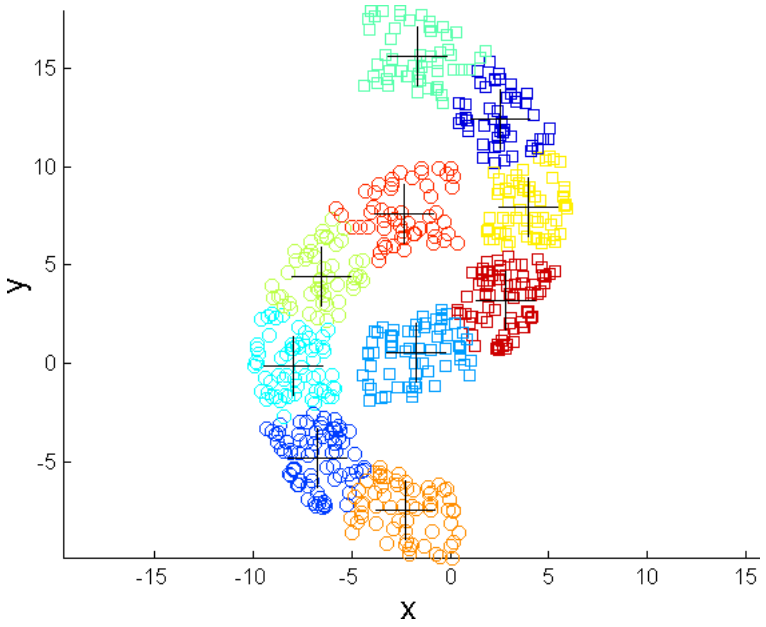
One solution is to use many clusters.

Find parts of clusters, but need to put together.

Overcoming K-means Limitations



Original Points



K-means Clusters

Example of K-means: k=2

1. Find two individuals furthest apart

Subject	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

2. Each individual is a center of each cluster

	Individual	Mean Vector (centroid)
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

3. Calculate the centroid

	Cluster 1		Cluster 2	
Step	Individual	Mean Vector (centroid)	Individual	Mean Vector (centroid)
1	1	(1.0, 1.0)	4	(5.0, 7.0)
2	1, 2	(1.2, 1.5)	4	(5.0, 7.0)
3	1, 2, 3	(1.8, 2.3)	4	(5.0, 7.0)
4	1, 2, 3	(1.8, 2.3)	4, 5	(4.2, 6.0)
5	1, 2, 3	(1.8, 2.3)	4, 5, 6	(4.3, 5.7)
6	1, 2, 3	(1.8, 2.3)	4, 5, 6, 7	(4.1, 5.4)

4. Regenerate the centroid

	Individual	Mean Vector (centroid)
Cluster 1	1, 2, 3	(1.8, 2.3)
Cluster 2	4, 5, 6, 7	(4.1, 5.4)

5. Calculate the distance between individuals and the new centroids

Individual	Distance to mean (centroid) of Cluster 1	Distance to mean (centroid) of Cluster 2
1	1.5	5.4
2	0.4	4.3
3	2.1	1.8
4	5.7	1.8
5	3.2	0.7
6	3.8	0.6
7	2.8	1.1

6. Regenerate the clusters

	Individual	Mean Vector (centroid)
Cluster 1	1, 2	(1.3, 1.5)
Cluster 2	3, 4, 5, 6, 7	(3.9, 5.1)

Lambton College
School of Computer Studies

Lambton College

Lecture 6

CBD-3335 Data Mining and Analysis

Topics

- Understanding Association Rule
- Taxonomy of Clustering
- Applications of NLP
- NLP components
- Text processing

Understanding Association Rule

- Analyzes and predicts customer behaviour.
- If/then statements.

Example

Bread => butter.

buys{ onions,potatoes} => buys {tomatoes}

Association Rule Components

Bread => butter[20 %,45%].

Bread : Ancedent.

Butter : consequent.

20% :Support

45% :confidence

Association Rule Components

A => B

- Support denotes probability that contains both A & B.
- Confidence denotes probability that a transaction containing A also contains B.

Association Rules
Example for market basket data

- Items={A,B,C,D,E,F}

Transaction-id	Items bought
10	A, B, D
20	A, C, D
30	A, D, E
40	B, E, F
50	B, C, D, E, F

Let min_sup = 60% (3)

min_conf = 50%

FP= {A:3, B:3, D:4, E:3, AD:3}

Association rules:

$A \rightarrow D$ (60%, 100%)

$D \rightarrow A$ (60%, 75%)

Types of Association Rule

- Single Dimensional

Bread => butter.

Dimension: buying.

- Multidimensional

With 2 or more predicates or dimensions.

Occupation(I.T),Age (>22) => buys(laptop)

Types of Association Rule

- Hybrid Dimensional ()

With Repetative predicates or dimensions.

Time(5’0 clock) ,buys (tea) => buys (biscuits).

Fields of Association Rule

- Web Usage Mining.
- Banking
- Bio Informatics.
- Market based Analysis.
- Credit/debit card analysis.
- Product clustering.
- Catalog design.

Algorithms for Association Rule

- Apriori Algorithm
- Elcat Algorithm
- F.P Growth Algorithm

Apriori

- The Apriori property:

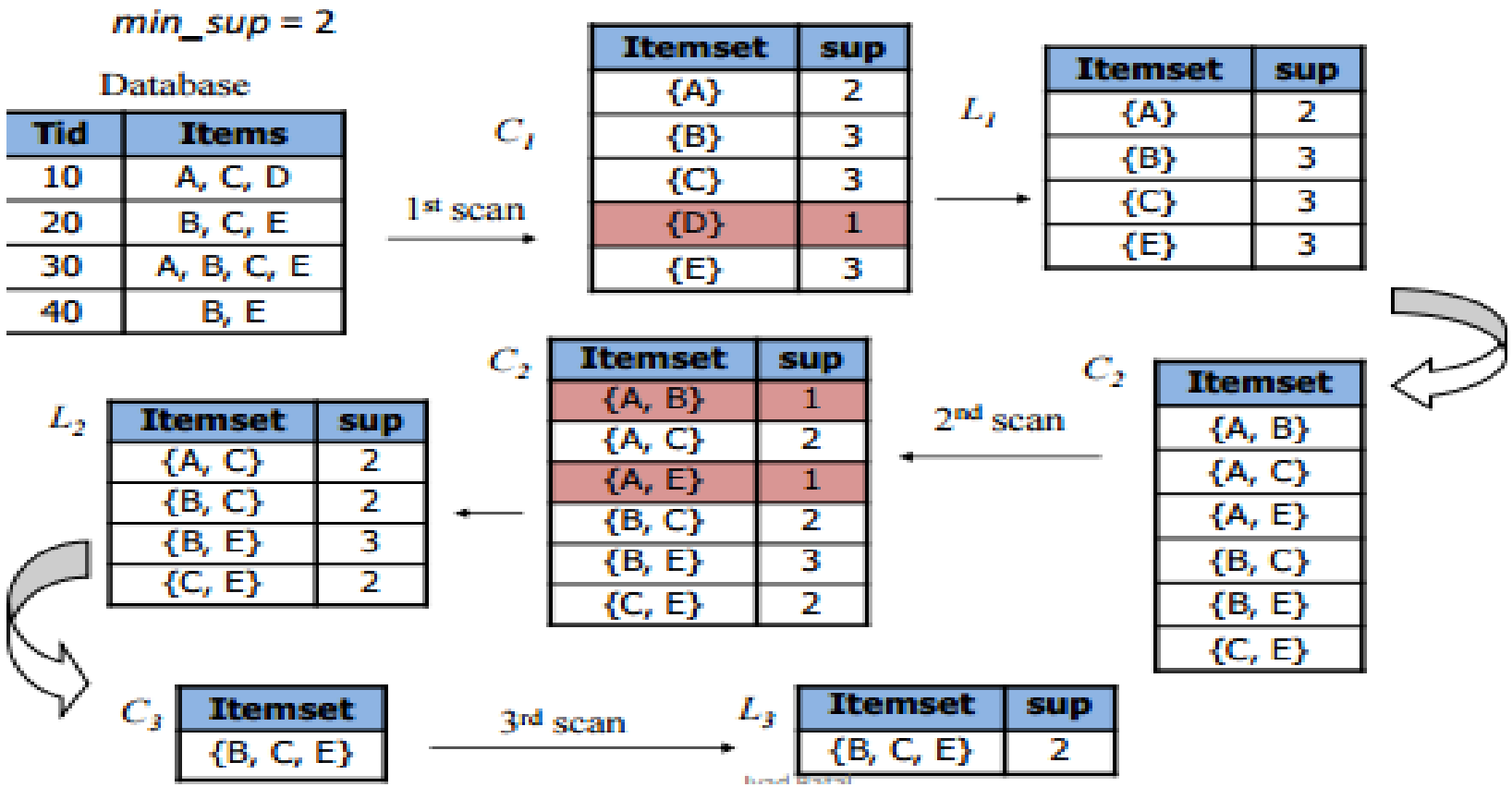
  – Any subset of a frequent pattern must be frequent.

  – If {beer, chips, nuts} is frequent, so is {beer, chips}, i.e., every transaction having {beer, chips, nuts} also contains {beer, chips}.

Apriori

- The Apriori procedures:
- Initially, scan DB once to get frequent 1-itemset
  - For each level k:
    - ✓ Generate length (k+1) candidates from length k frequent patterns
    - ✓ Scan DB and remove the infrequent candidates
  - Terminate when no candidate set can be generated

Apriori

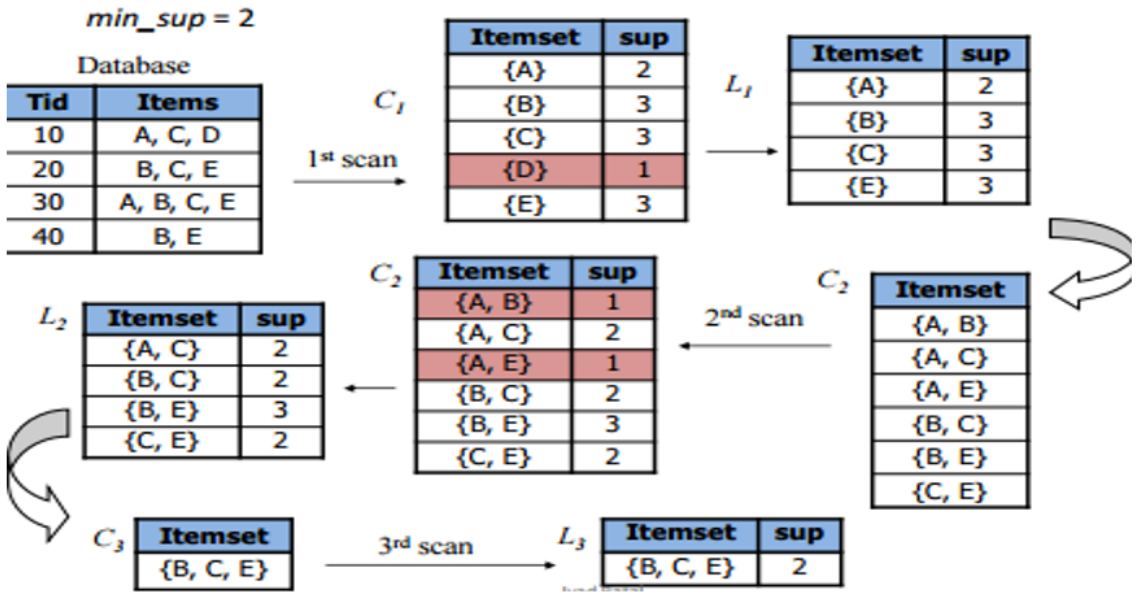

min_sup = 2
Database
Tid | Items
10 | A, C, D
20 | B, C, E
30 | A, B, C, E
40 | B, E
1st scan
C1
Itemset | sup
{A} | 2
{B} | 3
{C} | 3
{D} | 1
{E} | 3
L1
Itemset | sup
{A} | 2
{B} | 3
{C} | 3
{E} | 3
C2
Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}
2nd scan
C2
Itemset | sup
{A, B} | 1
{A, C} | 2
{A, E} | 1
{B, C} | 2
{B, E} | 3
{C, E} | 2
L2
Itemset | sup
{A, C} | 2
{B, C} | 2
{B, E} | 3
{C, E} | 2
C3
Itemset
{B, C, E}
3rd scan
L3
Itemset | sup
{B, C, E} | 2

Apriori

- Candidate generation: Assume we are generating k+1 candidates at level k
  - Step 1: self-joining two frequent k-patterns if they have the same k-1 prefix
  - Step 2: remove a candidate if it contains any infrequent kpattern.
- Example: L3={abc, abd, acd, ace, bcd}
  - Self-joining: L3*L3
    - abc and abd ⇒abcd
    - acd and ace ⇒acde
    - acde is removed because ade is not in L3
  - C4={abcd}

Apriori

Apriori



Exercise

min_sup = 2

TID	Items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Frequent Pattern Mining

- A Frequent pattern is a pattern (a set of items, subsequences, subgraphs, etc.) that occurs frequently in a data set.
- Motivation: Finding inherent regularities (associations) in data.
- Forms the foundation for many essential data mining tasks:
  - Association, correlation, and causality analysis
  - Classification: associative classification
  - Cluster analysis: frequent pattern-based clustering
  - …

Frequent Pattern Mining

- Association Rules and Frequent Patterns
- Frequent Pattern Mining Algorithms
  - Apriori
  - FP-growth
- Correlation Analysis
- Constraint-based Mining
- Using Frequent Patterns for Classification
  - Associative Classification (rule-based classification)
  - Frequent Pattern-based Classification

Frequent Pattern Mining

- An item (I) is:
  - For market basket data: I is an item in the store, e.g. milk.
  - For relational data: I is an attribute-value pair (numeric attributes should be discretized), e.g. salary=high, gender=male.
- A pattern (P) is a conjunction of items: P=I1 ∧I2 ∧… In (itemset)
- A pattern defines a group (subpopulation) of instances.
- Pattern P′ is subpattern of P if P′ ⊂P
- A rule R is A ⇒B where A and B are disjoint patterns.
  - Support(A ⇒B)=P(A n B)
  - Confidence(A ⇒B)=P(B|A)=posterior probability

Association Rules

- Framework: find all the rules that satisfy both a minimum support (min_sup) and a minimum confidence (min_conf) thresholds.
- Association rule mining:
- Find all frequent patterns (with support ≥ min_sup).
- Generate strong rules from the frequent patterns.
- The second step is straightforward:
  - For each frequent pattern p, generate all nonempty subsets.
  - For every non-empty subset s, output the rule s ⇒(p-s) if conf=sup(p)/sup(s) ≥ min_conf

FP-growth

The FP-growth algorithm:

- mining frequent patterns without candidate generation
- Compress a large database into a compact Frequent-Pattern tree (FPtree) structure
  – highly condensed, but complete for frequent pattern mining
  – avoid costly database scans
- Develop an efficient, FP-tree-based frequent pattern mining method
  –A divide-and-conquer methodology: decompose mining tasks into smaller ones
  – Avoid candidate generation: sub-database test only!

FP-growth: Constructing the FP-tree

TID	Items bought	(ordered) frequent items
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

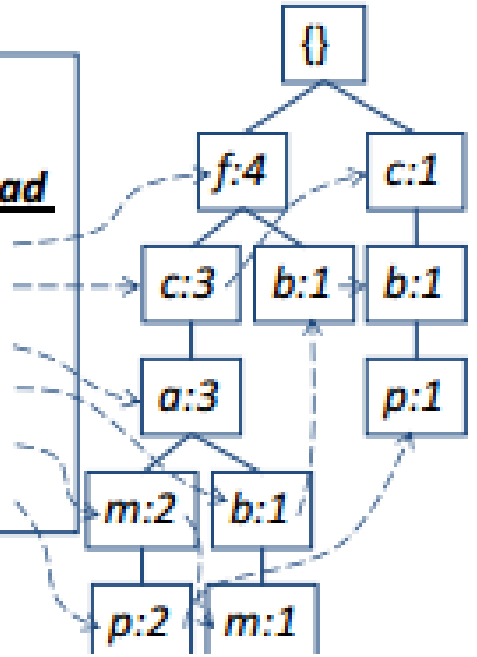
Steps:

1. Scan DB once, find frequent 1-itemset (single item pattern)
2. Order frequent items in frequency descending order
3. Scan DB again, construct FP-tree

Header Table

Item	*frequency*	*head*
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

min_sup = 3



FP-growth

- Method (divide-and-conquer)

–For each item, construct its conditional pattern-base, and then its conditional FP-tree.

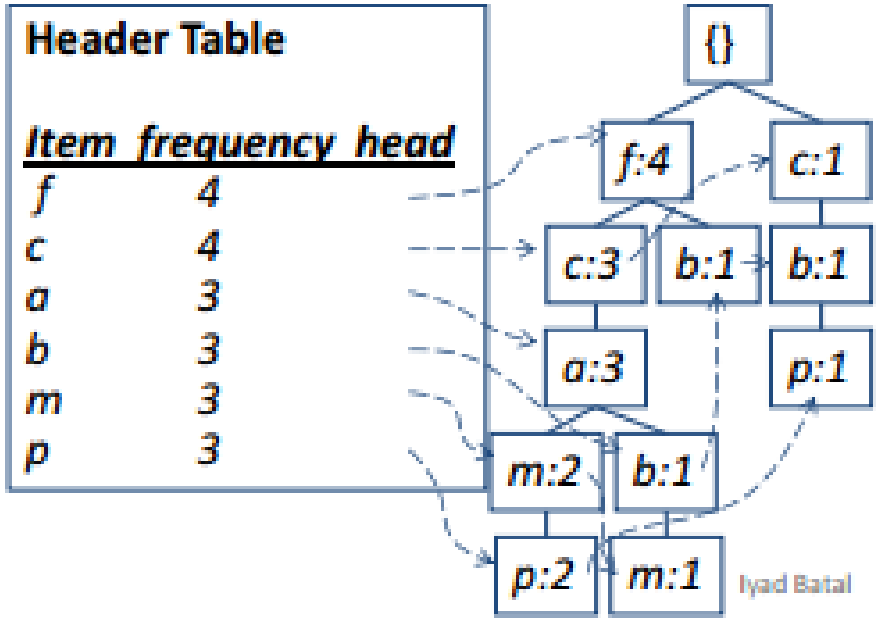
– Repeat the process on each newly created conditional FP-tree.

–Until the resulting FP-tree is empty, or it contains only one path (single path will generate all the combinations of its sub-paths, each of which is a frequent pattern)

FP-growth

Step 1: From FP-tree to Conditional Pattern Base

- Starting at the frequent header table in the FP-tree
- Traverse the FP-tree by following the link of each frequent item
- Accumulate all of transformed prefix paths of that item to form a conditional pattern base



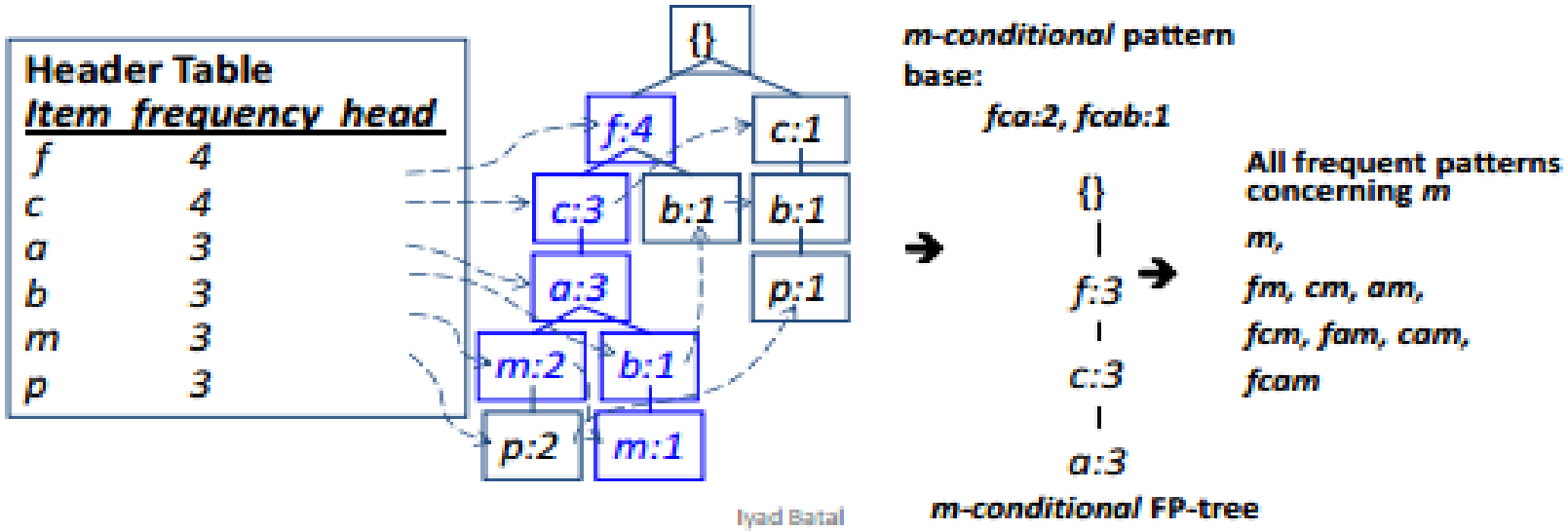
***Conditional* pattern bases**

item	*cond. pattern base*
c	*f:3*
a	*fc:3*
b	*fca:1, f:1, c:1*
m	*fca:2, fcab:1*
p	*fcam:2, cb:1*

FP-growth

Step 2: Construct Conditional FP-tree

- Start from the end of the list
- For each pattern-base
  - Accumulate the count for each item in the base
  - Construct the FP-tree for the frequent items of the pattern base
- Example: Here we are mining for pattern m, min_sup=3



Step-1

ItemSet	Support count
{I1}	6
{I2}	7
{I3}	6
{I4}	2
{I5}	2

Items bought

TID	Items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Minimum Support = 2

FP-growth

Step-2

ItemSet	Support count
{I2}	7
{I1}	6
{I3}	6
{I4}	2
{I5}	2

Minimum Support = 2

Items bought sorted

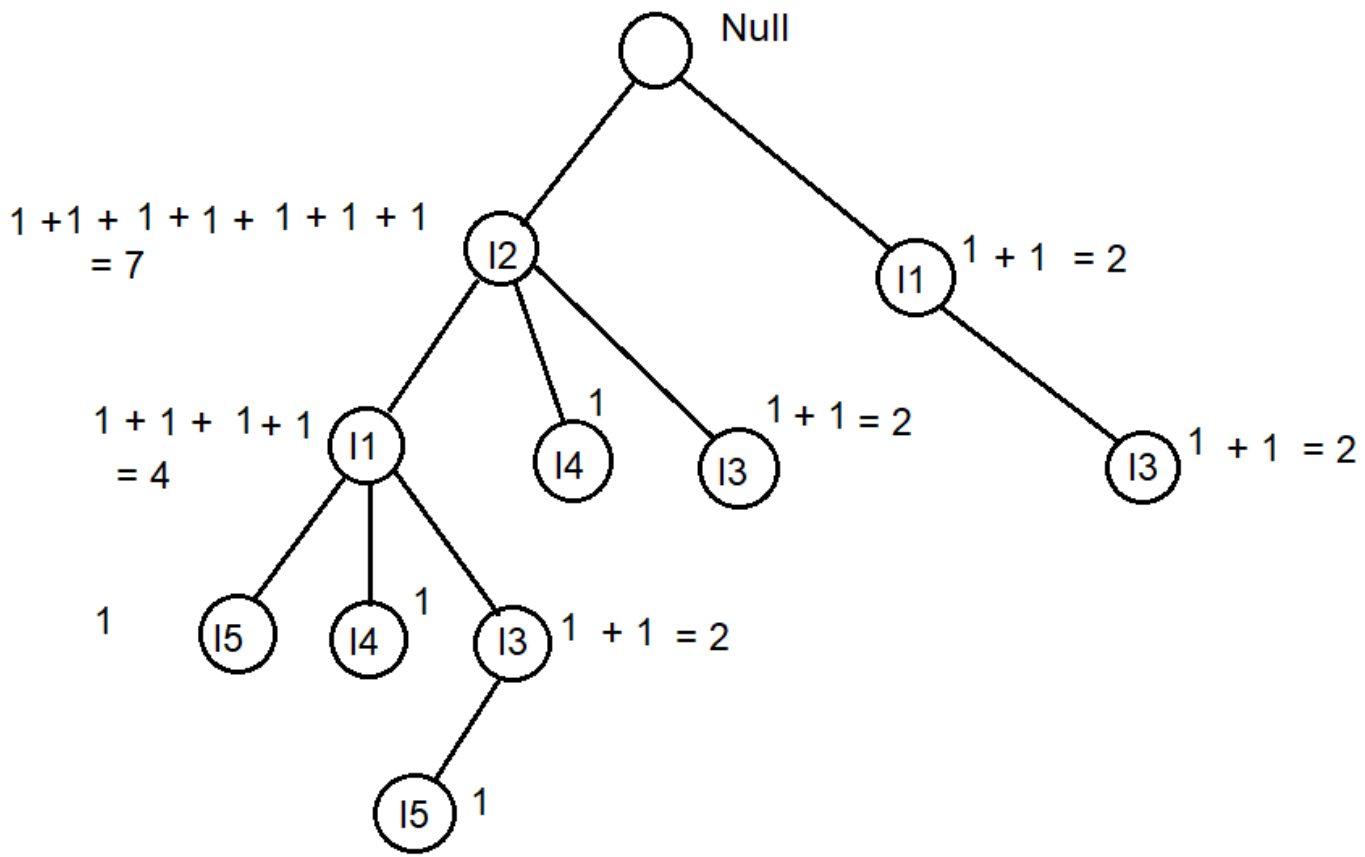
Step-3

TID	Items
T100	I2, I1, I5
T200	I2, I4
T300	I2, I3
T400	I2, I1, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I2, I1, I3, I5
T900	I2, I1, I3

FP-growth

FP-growth

Step-4



FP-growth

Step-5

Items	Conditional pattern base	Conditional FP tree	Frequent Pattern Generated
I5	{I2, I1: 1} {I2, I1, I3: 1}	{I2: 2} {I1: 2} {I3: 1}	{I2, I5: 2} {I1, I5: 2}, {I2, I1, I5: 2}
I4	{I2: 1} {I2, I1: 1}	{I2: 2} {I1: 1}	{I2, I4: 2}
I3	{I2: 2} {I2, I1: 2} {I1: 2}	<{I2: 4} {I1: 2} > {I1: 2}	{I2, I3: 4} {I1, I3: 4} {I2, I1, I3: 2}
I1	{I2: 4}	{I2: 4}	{I2, I1: 4}

Step-2

ItemSet	Support count
{I2}	7
{I1}	6
{I3}	6
{I4}	2
{I5}	2

Step-3

TID	Items
T100	I2, I1, I5
T200	I2, I4
T300	I2, I3
T400	I2, I1, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I2, I1, I3, I5
T900	I2, I1, I3

FP-growth

- FP-growth is faster than Apriori because:

– No candidate generation, no candidate test

– Use compact data structure

– Eliminate repeated database scan

–Basic operation is counting and FP-tree building (no pattern matching)

- Disadvantage: FP-tree may not fit in main memory!

Correlation analysis

- Association rule mining often generates a huge number of rules, but a majority of them either are redundant or do not reflect the true correlation relationship among data objects.

•Some strong association rules (based on support and confidence) can be misleading.

•Correlation analysis can reveal which strong association rules are interesting and useful.

Correlation analysis

- play basketball = > eat cereal [40%, 66.7%] is misleading

– The overall % of students eating cereal is 75% > 66.7%.

- play basketball = > not eat cereal [20%, 33.3%] is more accurate, although with lower support and confidence

Contingency table

	Basketball	Not basketball	Sum (row)
Cereal	2000 (40%)	1750 (35%)	3750 (75%)
Not cereal	1000 (20%)	250 (5%)	1250 (25%)
Sum(col.)	3000 (60%)	2000 (40%)	5000 (100%)

Associative classification

- Associative classification: build a rule-based classifier from association rules.
- This approach overcomes some limitations of greedy methods (e.g. decision-tree, sequential covering algorithms), which considers only one attribute at a time (found to be more accurate than C4.5).
- Build class association rules:
  - Association rules in general can have any number of items in the consequent.
  - Class association rules set the consequent to be the class label.
- Example: Age=youth ∧ Credit=OK ⇒ buys_computer=yes [sup=20%, conf=90%]

- Associative classification CBA
- CBA: Classification-Based Association
- Use the Apriori algorithm to mine the class association rules.
- Classification:

– Organize the rules according to their confidence and support.

– classify a new example x by the first rule satisfying x.

– Contains a default rule (with lowest precedence)

Frequent pattern-based classification

- The classification model is built in the feature space of single features as well as frequent patterns, i.e. map the data to a higher dimensional space.

•Feature combination can capture more underlying semantics than single features.

•Example: word phrases can improve the accuracy of document classification.

•FP-based classification been applied to many problems: – Graph classification – Time series classification – Protein classification – Text classification