Introduction to Data Science

CS 5665
Utah State University
Department of Computer Science
Instructor: Prof. Kyumin Lee

Project Teams

- Sahit Katragadda, Sree Vidya Susarla, Shivakesh Reddy Annepally
- Arshdeep Singh, Venkatesh Kadali, Rohit Gopalan
- Sidhant Chatterjee, Sirisha Rani Deekonda
- Jake Felzien, Mike Larsen, and Yancy Knight
- Bhagyashree, Meiling, Anuj
- Mounika, Akhil, Pravallika
- Prakhar Amlathe, Amitesh Mahajan, Vaibhav Sahu

So far 20 students

Practicum

- Nick & Amitesh: Storm – Nov 10
- Karun: Tensorflow – Nov 17

HW1

- https://usu.instructure.com/courses/431417/assignments/2117107
- Due date: September 22

Previous Class…

Measuring Data Similarity and Dissimilarity
➔ Nominal (Binary), Ordinal and Numeric (Quantitative)

Measuring Distance of Numeric Data
➔ Manhattan Distance and Euclidean Distance

Measuring Data Similarity and Dissimilarity*

*(These are mainly for understanding the relationship between objects with multiple attributes)

Cosine Similarity

- A **document** can be represented by thousands of attributes, each recording the *frequency* of a particular word (such as keywords) or phrase in the document.

Document	*team*	*coach*	*hockey*	*baseball*	*soccer*	*penalty*	*score*	*win*	*loss*	*season*
Document1	5	0	3	0	2	0	0	2	0	0
Document2	3	0	2	0	1	1	0	1	0	1
Document3	0	7	0	2	1	0	0	3	0	0
Document4	0	1	0	0	1	2	2	0	3	0

- Other vector objects: gene features in micro-arrays, …
- Applications: information retrieval, biologic taxonomy, gene feature mapping, …
- Cosine measure: If d_1 and d_2 are two vectors (e.g., term-frequency vectors), then

  $\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1||\ ||d_2||$,

  where • indicates vector dot product, $||d||$: the length of vector d

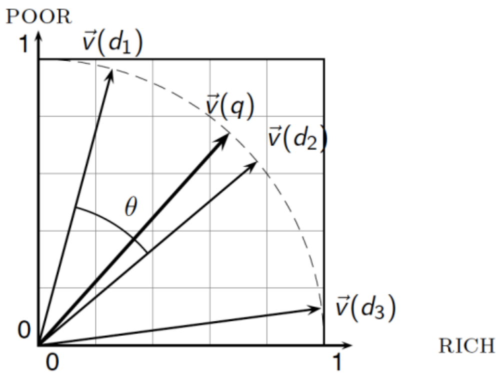
cosine(document1,document2)

Dot product

$$\cos(\vec{d}_1, \vec{d}_2) = \frac{\vec{d}_1 \bullet \vec{d}_2}{|\vec{d}_1||\vec{d}_2|} = \frac{\sum_{i=1}^{|V|} d_{1i} d_{2i}}{\sqrt{\sum_{i=1}^{|V|} d_{1i}^2}\sqrt{\sum_{i=1}^{|V|} d_{2i}^2}}$$

$\cos(d_1, d_2)$ is the cosine similarity of d_1 and d_2 … or, equivalently, the cosine of the angle between d_1 and d_2.

Cosine similarity illustrated



Example: Cosine Similarity

$$\cos(\vec{d}_1, \vec{d}_2) = \frac{\vec{d}_1 \bullet \vec{d}_2}{|\vec{d}_1||\vec{d}_2|} = \frac{\sum_{i=1}^{|V|} d_{1i} d_{2i}}{\sqrt{\sum_{i=1}^{|V|} d_{1i}^2}\sqrt{\sum_{i=1}^{|V|} d_{2i}^2}}$$

- Ex: Find the **similarity** between documents 1 and 2.

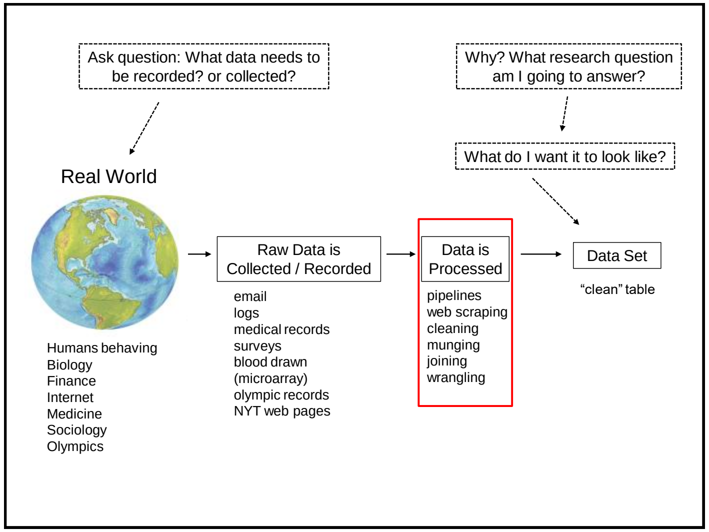
d_1 = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)
d_2 = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)

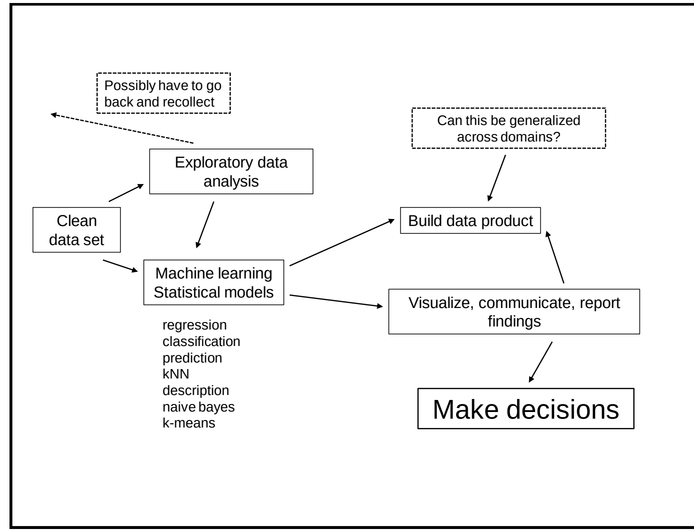
$d_1 \bullet d_2$ = 5*3+0*0+3*2+0*0+2*1+0*1+0*1+2*1+0*0+0*1 = 25
$||d_1||$ = (5*5+0*0+3*3+0*0+2*2+0*0+0*0+2*2+0*0+0*0)$^{0.5}$=(42)$^{0.5}$ = 6.481
$||d_2||$ = (3*3+0*0+2*2+0*0+1*1+1*1+0*0+1*1+0*0+1*1)$^{0.5}$=(17)$^{0.5}$ = 4.12
cos(d_1, d_2) = 0.94

Summary

- Data attribute types: nominal, binary, ordinal, interval-scaled, ratio-scaled
- Many types of data sets, e.g., numerical, text, graph, Web, image.
- Gain insight into the data by:
  - Basic statistical data description: central tendency, dispersion, graphical displays
  - Data visualization: map data onto graphical primitives
  - Measure data similarity
- Above steps are the beginning of data preprocessing
- Many methods have been developed but still an active area of research
- Read sections 1 and 2 in Data Mining Concepts and Techniques

Data Science: The Context





Data Preprocessing: Overview

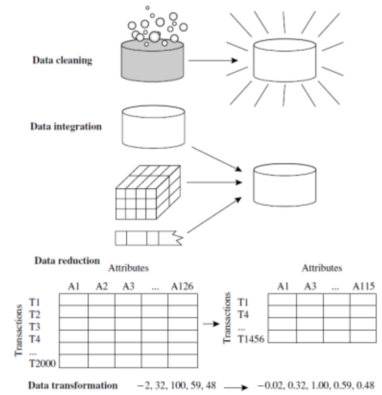
Why Preprocess the Data?

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

Major Tasks in Data Preprocessing

- **Data cleaning**
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- **Data integration**
  - Integration of multiple databases/data sources, or files
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

Forms of Data Preprocessing



Data Cleaning

Data Cleaning

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

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

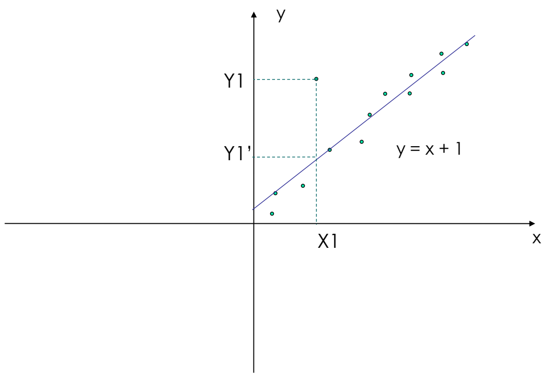
Noisy Data

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

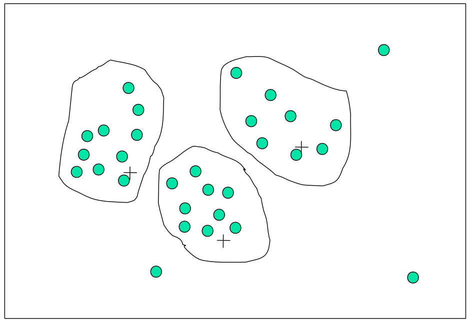
How to Handle Noisy Data?

- **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)

Regression



Cluster Analysis



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

Data Integration

Data Integration

- **Data integration**:
  - Combines data from multiple sources into a coherent store
- Entity identification problem:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton, Cust-id = Cust-#
- 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 and covariance analysis
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Correlation Analysis (Nominal Data)

- X^2 (chi-square) test

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

- The larger the X^2 value, the more likely the variables are related
- The cells that contribute the most to the X^2 value are those whose actual count is very different from the expected count
- Expected frequency of (A_i, B_j), which can be calculated as

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

- Correlation does not imply causality
  - # of hospitals and # of car-theft in a city are correlated
  - Both are causally linked to the third variable: population

Chi-Square Calculation: An Example

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

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

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

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

Correlation Analysis (Numeric Data)

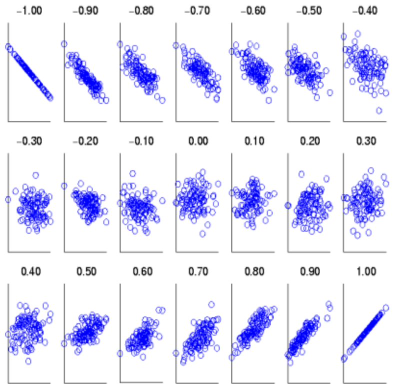
- Correlation coefficient (also called Pearson's product moment coefficient)

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

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

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

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.