DATA MINING LECTURE 2

What is data?

What is Data Mining?

- Data mining is the use of efficient techniques for the analysis of very large collections of data and the extraction of useful and possibly unexpected patterns in data.

- "Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data analyst" (Hand, Mannila, Smyth)

- "Data mining is the discovery of models for data" (Rajaraman, Ullman)
  - We can have the following types of models
    - Models that explain the data (e.g., a single function)
    - Models that predict the future data instances.
    - Models that summarize the data
    - Models the extract the most prominent features of the data.

Why do we need data mining?

- Really huge amounts of complex data generated from multiple sources and interconnected in different ways
  - Scientific data from different disciplines
    - Weather, astronomy, physics, biological microarrays, genomics
  - Huge text collections
    - The Web, scientific articles, news, tweets, facebook postings.
  - Transaction data
    - Retail store records, credit card records
  - Behavioral data
    - Mobile phone data, query logs, browsing behavior, ad clicks
  - Networked data
    - The Web, Social Networks, IM networks, email network, biological networks.
  - All these types of data can be combined in many ways
    - Facebook has a network, text, images, user behavior, ad transactions.
- We need to analyze this data to extract knowledge
  - Knowledge can be used for commercial or scientific purposes.
  - Our solutions should scale to the size of the data

What is Data?

- Collection of data objects and their attributes
- An attribute is a property or characteristic of an object
  - Examples: name, date of birth, height, occupation.
  - Attribute is also known as variable, field, characteristic, or feature
- For each object the attributes take some values.
- The collection of attribute-value pairs describes a specific object
  - Object is also known as record, point, case, sample, entity, or instance

Attributes

Objects

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Size (n): Number of objects
Dimensionality (d): Number of attributes
Sparsity: Number of populated object-attribute pairs

Types of Attributes

- There are different types of attributes
  - Numeric
    - Examples: dates, temperature, time, length, value, count.
    - Discrete (counts) vs Continuous (temperature)
    - Special case: Binary/Boolean attributes (yes/no, exists/not exists)
  - Categorical
    - Examples: eye color, zip codes, strings, rankings (e.g, good, fair, bad), height in {tall, medium, short}
    - Nominal (no order or comparison) vs Ordinal (order but not comparable)

Numeric Relational Data

- If data objects have the same fixed set of numeric attributes, then the data objects can be thought of as points/vectors in a multi-dimensional space, where each dimension represents a distinct attribute

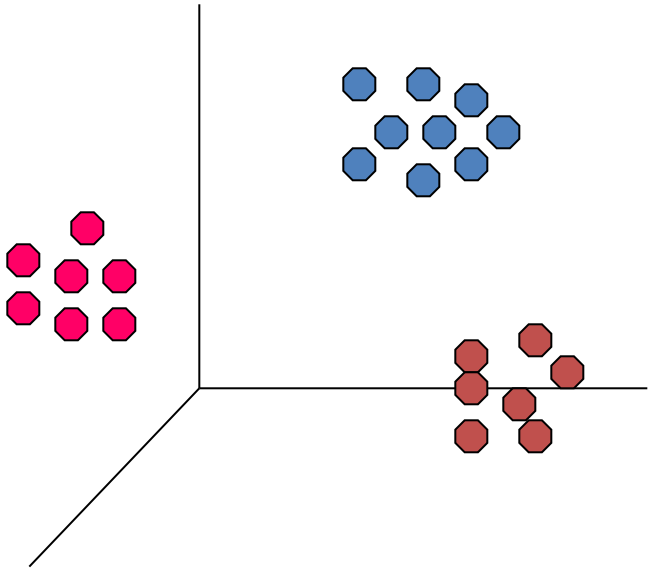
- Such data set can be represented by an n-by-d data matrix, where there are n rows, one for each object, and d columns, one for each attribute

Temperature	Humidity	Pressure
30	0.8	90
32	0.5	80
24	0.3	95

Numeric data

- Thinking of numeric data as points or vectors is very convenient

- For small dimensions we can plot the data
- We can use geometric analogues to define concepts like distance or similarity
- We can use linear algebra to process the data matrix



Categorical Relational Data

- Data that consists of a collection of records, each of which consists of a fixed set of categorical attributes

ID Number	Zip Code	Marital Status	Income Bracket
1129842	45221	Single	High
2342345	45223	Married	Low
1234542	45221	Divorced	High
1243535	45224	Single	Medium

Mixed Relational Data

- Data that consists of a collection of records, each of which consists of a fixed set of both numeric and categorical attributes

ID Number	Zip Code	Age	Marital Status	Income	Income Bracket
1129842	45221	55	Single	250000	High
2342345	45223	25	Married	30000	Low
1234542	45221	45	Divorced	200000	High
1243535	45224	43	Single	150000	Medium

Mixed Relational Data

- Data that consists of a collection of records, each of which consists of a fixed set of both numeric and categorical attributes

ID Number	Zip Code	Age	Marital Status	Income	Income Bracket	Refund
1129842	45221	55	Single	250000	High	No
2342345	45223	25	Married	30000	Low	Yes
1234542	45221	45	Divorced	200000	High	No
1243535	45224	43	Single	150000	Medium	No

Mixed Relational Data

- Data that consists of a collection of records, each of which consists of a fixed set of both numeric and categorical attributes

ID Number	Zip Code	Age	Marital Status	Income	Income Bracket	Refund
1129842	45221	55	Single	250000	High	0
2342345	45223	25	Married	30000	Low	1
1234542	45221	45	Divorced	200000	High	0
1243535	45224	43	Single	150000	Medium	0

Boolean attributes can be thought as both numeric and categorical
When appearing together with other attributes they make more sense as categorical
They are often represented as numeric though

Physical data storage

- Tab or Comma separated files (TSV/CSV), Excel sheets, relational tables
  - Assumes a strict schema and relatively dense data
- Flat file with triplets (record id, attribute, attribute value)
  - A very flexible data format, allows multiple values for the same attribute (e.g., phone number)
- JSON, XML format
  - Standards for data description that are more flexible than relational tables
  - There exist parsers for reading such data.

Examples

JSON EXAMPLE – Record of a person

```
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [],
  "spouse": null
}
```

XML EXAMPLE – Record of a person

```
<person>
  <firstName>John</firstName>
  <lastName>Smith</lastName>
  <age>25</age>
  <address>
    <streetAddress>21 2nd
Street</streetAddress>
    <city>New York</city>
    <state>NY</state>
    <postalCode>10021</postalCode>
  </address>
  <phoneNumbers>
    <phoneNumber>
      <type>home</type>
      <number>212 555-1234</number>
    </phoneNumber>
    <phoneNumber>
      <type>fax</type>
      <number>646 555-4567</number>
    </phoneNumber>
  </phoneNumbers>
  <gender>
    <type>male</type>
  </gender>
</person>
```

Set data

- Each record is a set of items from a space of possible items
- Example: Transaction data
  - Also called market-basket data

TID	Items
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Set data

- Each record is a set of items from a space of possible items
- Example: Document data
  - Also called bag-of-words representation

Doc Id	Words
1	the, dog, followed, the, cat
2	the, cat, chased, the, cat
3	the, man, walked, the, dog

Vector representation of market-basket data

- Market-basket data can be represented, or thought of, as numeric vector data
  - The vector is defined over the set of all possible items
  - The values are binary (the item appears or not in the set)

TID	Items
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

TID	Bread	Coke	Milk	Beer	Diaper
1	1	1	1	0	0
2	1	0	0	1	0
3	0	1	1	1	1
4	1	0	1	1	1
5	0	1	1	0	1

Sparsity: Most entries are zero. Most baskets contain few items

Vector representation of document data

- Document data can be represented, or thought of, as numeric vector data
  - The vector is defined over the set of all possible words
  - The values are the counts (number of times a word appears in the document)

Doc Id	Words
1	the, dog, follows, the, cat
2	the, cat, chases, the, cat
3	the, man, walks, the, dog

Doc Id	the	dog	follows	cat	chases	man	walks
1	2	1	1	1	0	0	0
2	2	0	0	2	1	0	0
3	1	1	0	0	0	1	1

Sparsity: Most entries are zero. Most documents contain few of the words

Physical data storage

- Usually set data is stored in flat files
  - One line per set

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
30 31 32
33 34 35
36 37 38 39 40 41 42 43 44 45 46
38 39 47 48
38 39 48 49 50 51 52 53 54 55 56 57 58
32 41 59 60 61 62
3 39 48
```

- I heard so many good things about this place so I was pretty juiced to try it. I'm from Cali and I heard Shake Shack is comparable to IN-N-OUT and I gotta say, Shake Shake wins hands down. Surprisingly, the line was short and we waited about 10 MIN. to order. I ordered a regular cheeseburger, fries and a black/white shake. So yummerz. I love the location too! It's in the middle of the city and the view is breathtaking. Definitely one of my favorite places to eat in NYC.
- I'm from California and I must say, Shake Shack is better than IN-N-OUT, all day, err'day.

Ordered Data

- Genomic sequence data

```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

- Data is a long ordered string

Ordered Data

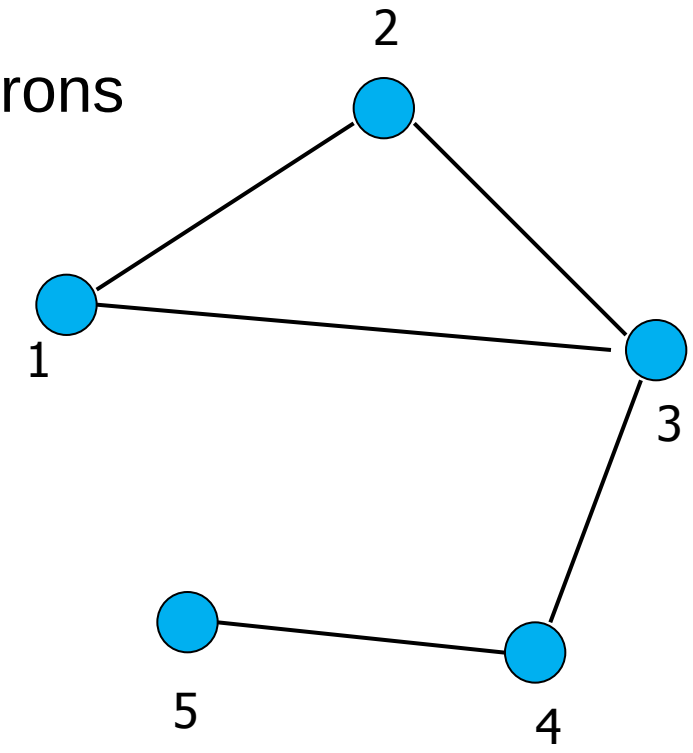
- Time series
  - Sequence of ordered (over “time”) numeric values.

Graph Data

- Graph data: a collection of entities and their pairwise relationships. Examples:
  - Web pages and hyperlinks
  - Facebook users and friendships
  - The connections between brain neurons

In this case the data consists of pairs:

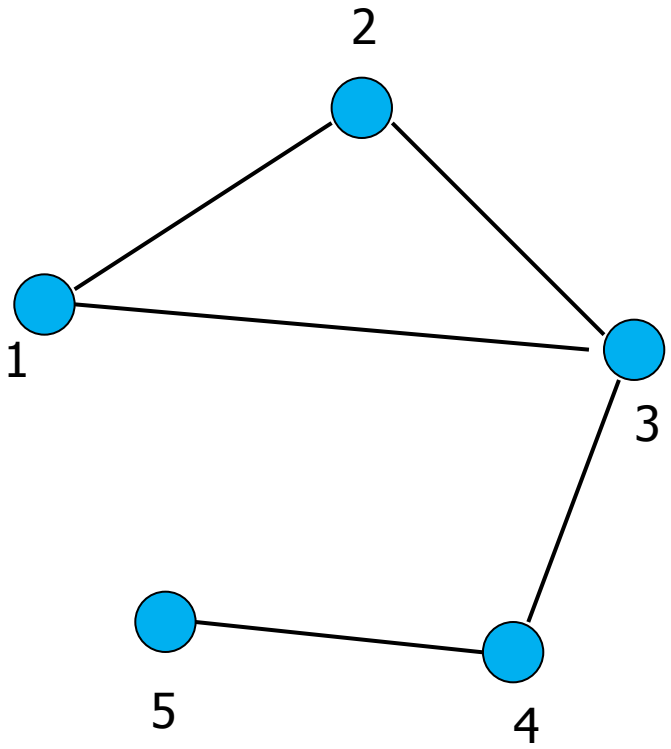
Who links to whom



Representation

- Adjacency matrix
  - Very sparse, very wasteful, but useful conceptually

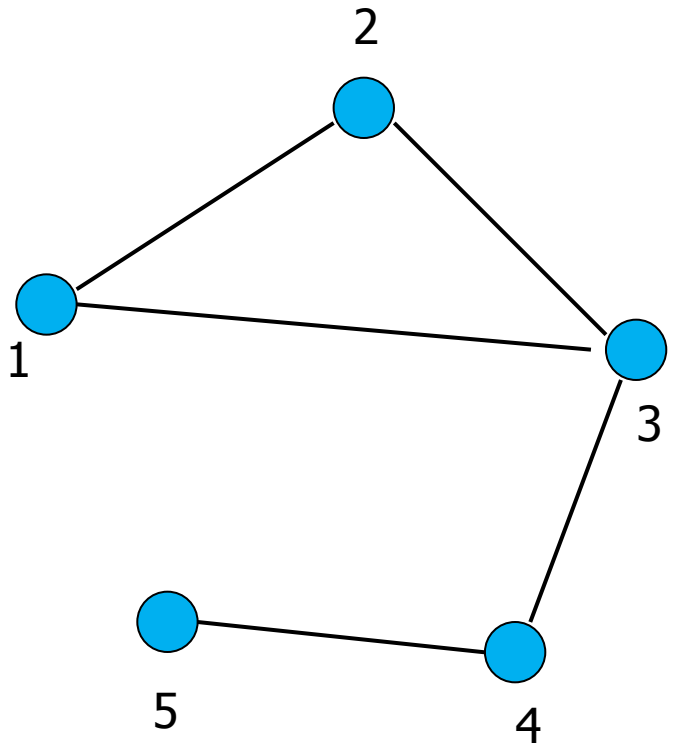
$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



Representation

- Adjacency list
  - Not so easy to maintain

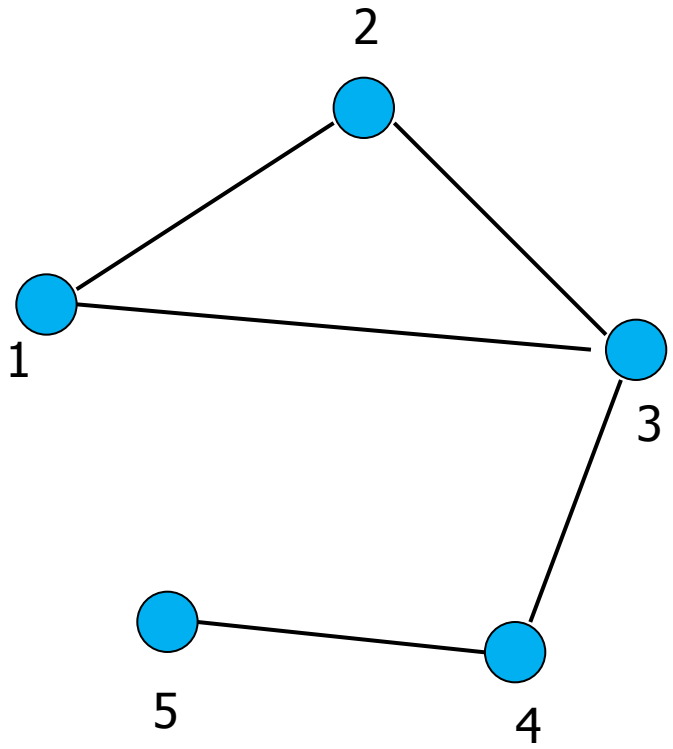
1: [2, 3]
2: [1, 3]
3: [1, 2, 4]
4: [3, 5]
5: [4]



Representation

- List of pairs
  - The simplest and most efficient representation

(1,2)
(2,3)
(1,3)
(3,4)
(4,5)



Types of data: summary

- Numeric data: Each object is a point in a multidimensional space
- Categorical data: Each object is a vector of categorical values
- Set data: Each object is a set of values (with or without counts)
  - Sets can also be represented as binary vectors, or vectors of counts
- Ordered sequences: Each object is an ordered sequence of values.
- Graph data: A collection of pairwise relationships