Data for Data Mining

Dr. Akinul Islam Jony

Assistant Professor

Dept. of Computer Science
Faculty of Science & Technology
American International University – Bangladesh

akinul@aiub.edu

Outline

- Knowledge about Data
- Data, Information and Knowledge
- Standard Formulation of Data
- Types of Variable/Attribute
- Data Preparation
- Data Cleaning
- Missing Values

Objectives & Outcomes

- To understand the knowledge about Data
- Understand the difference between data, information and knowledge
- To get familiar with the standard formulation of data
- To get familiar with different types of variable/attribute
- Get to know about data preparation and data cleaning
- To deal with Missing Values

Knowledge about Data

- It's tempting to jump straight into data mining, but first, we need to get the data ready. This involves having a closer look at *attributes* and data *values*.
- Real-world data are typically noisy, enormous in volume (often several gigabytes or more) and may originate from a hodgepodge of heterogenous sources.
- Knowledge about your data is useful for data *preprocessing*, the first major task of the data mining process.

Knowledge about Data

- To have the knowledge about your data, you will want to know the following:
  - What are the types of attributes or fields that make up your data?
  - What kind of values does each attribute have?
  - Which attributes are discrete, and which are continuous-valued?
  - What do the data look like?
  - How are the values distributed?
  - Are there ways we can visualize the data to get a better sense of it all?
  - Can we spot any outliers?
  - Can we measure the similarity of some data objects with respect to others?
  - Gaining such insight into the data will help with the subsequent analysis.

Data & Information & Knowledge

- Data
  - a "given" or a fact that represents something in real world
  - raw materials, can be processed, structured or unstructured
  - Data are elements of analysis
- Information
  - Data that have meaning in context
  - Data related
  - Data after manipulation
- Knowledge
  - familiarity, awareness and understanding of someone or something
  - acquired through experience or learning
  - it is a concept mainly for humans unlike data and information.

Knowledge is not information and information is not data.
Knowledge is derived from information in the same way information is derived from data.

Standard Formulation of Data

- Data Objects
  - Datasets are made up of data objects.
  - A data object represents an entity—
    - in a sales database, the objects may be customers, store items, and sales;
    - in a medical database, the objects may be patients;
    - in a university database, the objects may be students, professors, and courses.
  - Data objects are typically described by a number of *variables* (also called *attributes)*.
  - Data objects can also be referred to as *samples*, *examples*, *instances*, *data points*, or *objects*.
  - If the data objects are stored in a database, they are *data tuples*. That is, the *rows* of a database correspond to the *data objects*, and the *columns* correspond to the *attributes*.

Standard Formulation of Data

- Attribute/Variable
  - Each object is described by a number of *variables* that correspond to its properties.
  - In data mining variables are often called *attributes*. For examples, eye color of a person.
  - *Categorical* (*qualitative*) variables take on values that are names or labels. For example, the color of a ball (such as red, green, blue).
  - *Continuous* (*quantitative*) variables are numerical. They represent a measurable quantity. when we speak of the population of a city, we are talking about the number of people in the city.

Standard Formulation of Data

- Instance/Record
  - The set of variable values corresponding to each of the objects is called a *record* or (more commonly) an *instance*.

- Dataset
  - The complete set of data available to us for an application is called a *dataset*. A dataset is often depicted as a *table*, with each *row* representing an *instance*. Each *column* contains the value of one of the *variables* (*attributes*) for each of the *instances*.

Standard Formulation of Data

- Class attribute
  - This dataset is an example of *labelled* data, where one attribute is given special significance and the aim is to predict its value. This attribute is often called by a standard name '*class*'.
  - When there is no such significant attribute, we call the data *unlabelled*.

SoftEng	ARIN	HCI	CSA	Project	Class
A	B	A	B	B	Second
A	B	B	B	B	Second
B	A	A	B	A	Second
A	A	A	A	B	First
A	A	B	B	A	First
B	A	A	B	B	Second
.....	.....	.....	.....	.....	.....
A	A	B	A	B	First

The *Degrees* Dataset

Types of Variable/Attribute

- In general, there are many types of variable that can be used to measure the properties of an object.
- A lack of understanding of the differences between the various types can lead to problems with any form of data analysis.
- At least six main types of variable can be distinguished:
  - Nominal Variables
  - Binary Variables
  - Ordinal Variables
  - Integer Variables
  - Interval-scaled Variables
  - Ratio-scaled Variables

Nominal Variables

- Nominal means “relating to names”. The values of a nominal attribute are symbols or names of things. Each value represents some kind of category, code, or state, and so nominal attributes are also referred to as categorical. The values do not have any meaningful order.
- A variable used to put objects into categories, e.g., the name or color of an object.
- A nominal variable may be numerical in form, but the numerical values have no mathematical interpretation. For example, we might label 10 people as numbers 1, 2, 3, ..., 10, but any arithmetic with such values, e.g., 1 + 2 = 3 would be meaningless. They are simply labels.
- A *classification* can be viewed as a nominal variable which has been designated as of particular importance.

Nominal Variables

- For example, *hair color* and *marital status* are two attributes describing *person* objects.
  - Here, possible values for hair color are black, brown, blond, red, auburn, gray, and white.
  - The attribute marital status can take on the values single, married, divorced, and widowed.
  - Both *hair color* and *marital status* are nominal attributes.
- Another example of a nominal attribute is *occupation*, with the values teacher, dentist, programmer, farmer, and so on.

Binary Variables

- A binary variable is a special case of a nominal variable that takes only two possible values: true or false, 1 or 0 etc.
- Binary attributes are referred to as Boolean if the two states correspond to *true* and *false*.
- For example, the attribute *smoker* describing a *patient* object, 1 indicates that the patient smokes, while 0 indicates that the patient does not.
- A binary attribute is *symmetric* if both of its states are equally valuable and carry the same weight; that is, there is no preference on which outcome should be coded as 0 or 1. One such example could be the attribute gender having the states male and female.
- A binary attribute is *asymmetric* if the outcomes of the states are not equally important, such as the positive and negative outcomes of a medical test for HIV. By convention, we code the most important outcome, which is usually the rarest one, by 1 (e.g., HIV positive) and the other by 0 (e.g., HIV negative).

Ordinal Variables

- Ordinal variables are similar to nominal variables, except that an ordinal variable has values that can be arranged in a meaningful order/rank, e.g., small, medium, large. But the magnitude between successive values is not known.
- For example, the attribute *drink size* corresponds to the size of drinks available at a fast-food restaurant. This nominal attribute has three possible values: *small*, *medium*, and *large*. The values have a meaningful sequence (which corresponds to increasing drink size); however, we cannot tell from the values how much bigger, say, a *medium* is than a *large*.
- Other examples of ordinal attributes include *grade* (e.g., A+, A, A-, B+, and so on) and *professional rank* (e.g., assistant, associate, and full for professor).

Integer/Numeric Variables

- An integer/numeric attribute is quantitative; that is, it is a measurable quantity, represented in integer or real values.
- Integer variables are ones that take values that are genuine integers, for example ‘number of children’. Unlike nominal variables that are numerical in form, arithmetic with integer variables is meaningful (1 child + 2 children = 3 children etc.).
- Numeric attributes can be interval-scaled or ratio-scaled.

Interval-scaled Variables

- Interval-scaled variables are variables that take numerical values which are measured at equal intervals from a zero point or origin. However, the origin does not imply a true absence of the measured characteristic.
- Interval-scaled attributes are measured on a scale of equal-size units. The values of interval-scaled attributes have order and can be positive, 0, or negative.
- Thus, in addition to providing a ranking of values, such attributes allow us to compare and quantify the difference between values.

Interval-scaled Variables

- For example, a temperature attribute is interval-scaled. Suppose that we have the outdoor temperature value for a number of different days, where each day is an object. By ordering the values, we obtain a ranking of the objects with respect to temperature. In addition, we can quantify the difference between values. For example, a temperature of 20° C is five degrees higher than a temperature of 15° C.
  - Temperatures in Celsius and Fahrenheit do not have a true zero-point, that is, neither 0 ° C nor 0 ° F indicates “no temperature.” Although we can compute the difference between temperature values, we cannot talk of one temperature value as being a multiple of another. Without a true zero, we cannot say, for instance, that 10 ° C is twice as warm as 5 ° C. That is, we cannot speak of the values in terms of ratios. The zero value does not imply ‘absence of temperature’.
- Calendar dates are another example. For instance, the years 2002 and 2010 are eight years apart. There is no true zero-point for calendar dates.

Ratio-scaled Variables

- Ratio-scaled variables are similar to interval-scaled variables except that the zero point does reflect the absence of the measured characteristic. For example, Kelvin temperature scale and molecular weight scale.
  - So, a temperature of 20 degrees Kelvin is twice one of 10 degrees Kelvin. A weight of 10 kg is twice one of 5 kg, a price of 100 dollars is twice a price of 50 dollars etc.
- A ratio-scaled attribute is a numeric attribute with an inherent zero-point. That is, if a measurement is ratio-scaled, we can speak of a value as being a multiple (or ratio) of another value.
- In addition, the values are ordered, and we can also compute the difference between values, as well as the mean, median, and mode.
- Other examples of ratio-scaled attributes include count attributes such as years of experience (e.g., the objects are employees) and number of words (e.g., the objects are documents). Additional examples include attributes to measure weight, height, latitude and longitude coordinates, and monetary quantities.

Categorical and Continuous Attributes

- Although the distinction between different categories of variable can be important in some cases, many practical data mining systems divide attributes into just two types:
  - **categorical** corresponding to nominal, binary and ordinal variables
  - **continuous** corresponding to integer, interval-scaled and ratio-scaled variables.

Ignore Attribute

- For many applications it is helpful to have a third category of attribute, the '**ignore**' attribute, corresponding to variables that are of no significance for the application.
- For example, the name of a patient in a hospital or the serial number of an instance, but which we do not wish to (or are unable to) delete from the dataset

Data Preparation

- For many applications, the data can simply be extracted from a database.
- However, for some applications the hardest task may be to get the data into a standard form in which it can be analyzed.
- For example, data values may have to be extracted from textual output generated by a fault logging system or (in a crime analysis application) extracted from transcripts of interviews with witnesses.
- The amount of effort required to do this may be considerable.

Data Cleaning

- Even when the data is in the standard form it cannot be assumed that it is error free.
- In real-world datasets erroneous values can be recorded for a variety of reasons, including measurement errors, subjective judgements and malfunctioning or misuse of automatic recording equipment.
- Erroneous values can be divided into **noisy** values and **invalid** values.

Data Cleaning

- Noisy Value:
  - A noisy value to mean one that is valid for the dataset but is incorrectly recorded.
  - For example, the number **69.72** may accidentally be entered as **6.972**, or a categorical attribute value such as **brown** may accidentally be recorded as another of the possible values, such as **blue**.
- Invalid Value:
  - A far smaller problem arises with noisy values that are invalid for the dataset, such as **69.7X** for **6.972** or **bbrown** for **brown**. These kind of values will consider as invalid values, not noise.
  - An invalid value can easily be detected and either corrected or rejected.

Data Cleaning

- In attempting to ‘clean up’ data it is helpful to have a range of software tools available, especially to give an overall visual impression of the data, when some anomalous values or unexpected concentrations of values may stand out.
- However, in the absence of special software, even some very basic analysis of the values of variables may be helpful. Simply sorting the values into ascending order (which for fairly small datasets can be accomplished using just a standard spreadsheet) may reveal unexpected results.

Data Cleaning: Example

- A numerical variable may only take six different values, all widely separated. It would probably be best to treat this as a categorical variable rather than a continuous one.
- All the values of a variable may be identical. The variable should be treated as an ‘ignore’ attribute.
- All the values of a variable except one may be identical. It is then necessary to decide whether the one different value is an error or a significantly different value. In the latter case the variable should be treated as a categorical attribute with just two values.

Data Cleaning: Example

- There may be some values that are outside the normal range of the variable. For example, the values of a continuous attribute may all be in the range **200** to **5000** except for the highest three values which are **22654.8**, **38597** and **44625.7**. If the data values were entered by hand a reasonable guess is that the first and third of these abnormal values resulted from pressing the initial key twice by accident and the second one is the result of leaving out the decimal point. If the data were recorded automatically, it may be that the equipment malfunctioned. This may not be the case, but the values should certainly be investigated.
  - Alternatively, they may be **outliers**, i.e., genuine values that are significantly different from the others. The recognition of outliers and their significance may be the key to major discoveries, especially in fields such as medicine and physics, so we need to be careful before simply discarding them or adjusting them back to 'normal' values.

Missing Values

- In many real-world datasets data values are not recorded for all attributes.
- This can happen simply because there are some attributes that are not applicable for some instances (e.g., certain medical data may only be meaningful for female patients or patients over a certain age). The best approach here may be to divide the dataset into two (or more) parts, e.g., treating male and female patients separately.
- It can also happen that there are attribute values that should be recorded that are missing. This can occur for several reasons, for example
  - a malfunction of the equipment used to record the data
  - a data collection form to which additional fields were added after some data had been collected
  - information that could not be obtained, e.g., about a hospital patient.
- There are several possible strategies for dealing with missing values. Two of the most used are **Discard Instances** and **Replace by Most Frequent/Average Value**.

Discard Instances

- This is the simplest strategy: delete all instances where there is at least one missing value and use the remainder.
- This strategy is a very conservative one, which has the advantage of avoiding introducing any data errors. Its disadvantage is that discarding data may damage the reliability of the results derived from the data.
- Although it may be worth trying when the proportion of missing values is small, it is not recommended in general.
- It is clearly not usable when all or a high proportion of all the instances have missing values.

Replace by Most Frequent/Average Value

- A less cautious strategy is to estimate each of the missing values using the values that are present in the dataset.
- A straightforward but effective way of doing this for a categorical attribute is to use its most frequently occurring (non-missing) value.
  - This is easy to justify if the attribute values are very unbalanced. For example, if attribute X has possible values a, b and c which occur in proportions 80%, 15% and 5% respectively, it seems reasonable to estimate any missing values of attribute X by the value a. If the values are more evenly distributed, say in proportions 40%, 30% and 30%, the validity of this approach is much less clear.
- In the case of continuous attributes, it is likely that no specific numerical value will occur more than a small number of times. In this case the estimate used is generally the average value.

Reducing the Number of Attributes

- For some datasets there can be substantially more attributes.
- Many irrelevant attributes will place an unnecessary computational overhead on any data mining algorithm. At worst, they may cause the algorithm to give poor results.
- When the number of attributes becomes large, there is always a risk that the results obtained will have only superficial accuracy and will actually be less reliable than if only a small proportion of the attributes were used — a case of '**more means less**'.
- There are several ways in which the number of attributes (or 'features') can be reduced before a dataset is processed. The term **feature reduction** or **dimension reduction** is generally used for this process.

Repository of Datasets

- Most of the commercial datasets used by companies for data mining are — unsurprisingly — not available for others to use.
- However, there are a number of 'libraries' of datasets that are readily available for downloading from the World Wide Web free of charge by anyone.
  - The UCI (University of California at Irvine) Repository of Datasets
    - https://archive.ics.uci.edu/ml/index.php
  - Kaggle
    - https://www.kaggle.com/

Exercises

- The following information is held in an *employee* database.
  *Name, Date of Birth, Sex, Weight, Height, Marital Status, Number of Childre*n.
  What is the type of each variable?

- Give two ways of dealing with missing data values.

References

- Max Bramer, “Chapter 2: Data for Data Mining”, *Principles of Data Mining* (4th Edition).