UNIT 1

What is data, and why is it important?

Data – a collection of facts (numbers, words, measurements, observations, etc) that has been translated into a form that computers can process

Data is of much importance now a days

1) Data helps to make better decisions

- Finding new customers
- Increasing customer retention
- Improving customer service
- Better managing marketing efforts
- Tracking social media interaction
- Predicting sales trends

2) Data helps you solve problems

Ex After experiencing a slow sales month or watching a poor-performing marketing campaign, how do you pinpoint what went wrong? Tracking and reviewing data from business processes helps you uncover performance breakdowns so you can better understand each part of the process and know which steps need to be fixed and which are performing well.

3) Data helps you understand performance

One example: Say you have a top-performing sales rep who you send most leads to. However, when you delve into the data it shows that she closes deals at a lower rate than one of your other sales reps who receives fewer leads but closes deals at a higher percentage. (**In fact, here's how you can easily track sales rep performance**.) Certainly some performance data that can affect how you portion out leads--which can lead to revenue increase. Performance data provides the clarity needed for better results.

4) Data helps you improve processes

Data helps you understand and improve business processes so you can reduce wasted money and time. Every company feels the effects of waste. It depletes resources, squanders time, and ultimately impacts the bottom line.

For example, bad advertising decisions can be one of the greatest wastes of resources in a company. With data showing how different marketing channels are performing, however, you can see which ones offer the greatest ROI and focus on those. Or you could dig into why other channels are not performing as well and work to improve their performance. This would allow you budget to generate more leads without having to increase the advertising spend.

5) Data helps you understand consumers

TYPES OF DATA

1. Structured data

Structured data is data whose elements are addressable for effective analysis. It has been organized into a formatted repository that is typically a database. It concerns all data which can be stored in database SQL in a table with rows and columns. They have relational keys and can easily be mapped into pre-designed fields. Today, those data are most processed in the development and simplest way to manage information. Example: Relational data.

2. **Semi-Structured data**

   Semi-structured data is information that does not reside in a relational database but that have some organizational properties that make it easier to analyze. With some process, you can store them in the relation database (it could be very hard for some kind of semi-structured data), but Semi-structured exist to ease space. Example: XML data.

3. **Unstructured data**

   Unstructured data is a data that is which is not organized in a pre-defined manner or does not have a pre-defined data model, thus it is not a good fit for a mainstream relational database. So for Unstructured data, there are alternative platforms for storing and managing, it is increasingly prevalent in IT systems and is used by organizations in a variety of business intelligence and analytics applications. Example: Word, PDF, Text, Media logs.

UNSTRUCTURED DATA IS CRITICAL

Notable fact: almost all information we used to operate with is unstructured: emails, articles, or business-related data like customer interactions. Unstructured data can be extremely different: extracted from a human language with NLP (Natural Language Processing), gained thru various sensors, scrapped from the Internet, acquired from NoSQL databases, etc.

As the majority of information we can access is unstructured, the benefits of unstructured data analysis are obvious. It can bring many useful insights and ideas on how to improve the performance of the company or a specific service.If we want a machine to process the data, so the first step is to make it “understandable” for computers. We should build a bridge between human understanding and computer processing. It means that most often human operator processes required data manually and translates it to the format suitable for machine processing.One of the main problems with qualitative data analysis, however, is that standard databases like Excel or SQL require a certain structure. Unfortunately, unstructured data lacks this structure and traditional ORM (object-relational mapping) software can’t process it properly to fill the database.But it doesn’t mean that we should forget about this kind of information and lose valuable insights. When you sift the unstructured data, you get details that allow seeing the full picture of what’s going on.The information you receive after the analysis can become the cornerstone of a successful business strategy since it usually contains essential nuances about customer behavior or current trends. Let’s take a look at an example.

<u>Sources of data</u>

- **Personal data**

Personal data is anything that is specific to you. It covers your demographics, your location, your email address and other identifying factors. It’s usually in the news when it gets leaked (like the Ashley Madison scandal) or is being used in a controversial way (when Uber worked out who was having an affair).Lots of different companies collect your personal data (especially social media sites), anytime you have to put in your email address or credit card details you are giving away your personal data. Often they’ll use that data to provide you with personalized suggestions to keep you engaged. Facebook for example uses your personal information to suggest content you might like to see based on what other people similar to you like.

In addition, personal data is aggregated (to depersonalize it somewhat) and then sold to other companies, mostly for advertising and competitive research purposes. That’s one of the ways you get targeted ads and content from companies you’ve never even heard of.

- **Transactional data**

Transactional data is anything that requires an action to collect. You might click on an ad, make a purchase, visit a certain web page, etc.

Pretty much every website you visit collects transactional data of some kind, either through Google Analytics, another 3rd party system or their own internal data capture system.

Transactional data is incredibly important for businesses because it helps them to expose variability and optimize their operations for the highest quality results. By examining large amounts of data, it is possible to uncover hidden patterns and correlations. These patterns can create competitive advantages, and result in business benefits like more effective marketing and increased revenue.

- **Web data**

Web data is a collective term which refers to any type of data you might pull from the internet, whether to study for research purposes or otherwise. That might be data on what your competitors are selling, published government data, football scores, etc. It’s a catchall for anything you can find on the web that is public facing (ie not stored in some internal database). Studying this data can be very informative, especially when communicated well to management.

Web data is important because it’s one of the major ways businesses can access information that isn’t generated by themselves. When creating quality business models and making important BI decisions, businesses need information on what is happening internally and externally within their organization and what is happening in the wider market.

Web data can be used to monitor competitors, track potential customers, keep track of channel partners, generate eads, build apps, and much more. It’s uses are still being discovered as the technology for turning unstructured data into structured data improves.

Web data can be collected by writing web scrapers to collect it, using a scraping tool, or by paying a third party to do the scraping for you. A web scraper is a computer program that takes a URL as an input and pulls the data out in a structured format – usually a JSON feed or CSV.

- **Sensor data**

Sensor data is produced by objects and is often referred to as the Internet of Things. It covers everything from your smartwatch measuring your heart rate to a building with external sensors that measure the weather.

So far, sensor data has mostly been used to help optimize processes. For example, AirAsia saved $30-50 million by using GE sensors and technology to help reduce operating costs and increase aircraft usage. By measuring what is happening around them, machines can make smart changes to increase productivity and alert people when they are in need of maintenance.

Inaccurate Data

Before we can assess data correctness we need to understand the various ways inaccurate values get into databases. There are many sources of data inaccuracies, and each contributes its own part to the total data quality problem. Understanding these sources will demonstrate the need for a comprehensive program of assessment, monitoring, and improvement. Having highly accurate data requires attention to all sources of inaccuracies and appropriate responses and tools for each.

There are four general areas where inaccuracies occur. The first three cause inaccuracies in data within the databases, whereas the fourth area causes inaccuracies in the information products produced from the data. If you roll up all potential sources of errors, the interesting conclusion is that the most important use of the data (corporate decision making) is made on the rendition of data that has the most inaccuracies.

Initial Data Entry

Most people assume that data inaccuracies are always the result of entering the wrong data at the beginning. This is certainly a major source of data inaccuracies but not the only source. Inaccurate data creation can be the result of mistakes, can result from flawed data entry processes, can be deliberate, or can be the result of system errors. By looking at our systems through these topics, you can gain insight into whether systems are designed to invite inaccurate data or are designed to promote accurate data.

Data Entry Mistakes

The most common source of a data inaccuracy is that the person entering the data just plain makes a mistake. You intend to enter blue but enter bleu instead; you hit the wrong entry on a select list; you put a correct value in the wrong field. Much of operational data originates from a person. People make mistakes; we make them all the time. It is doubtful that anyone could fill out a hundred-field form without making at least one mistake.

A real-world example involves an automobile damage claims database in which the COLOR field was entered as text. Examination of the content of this field yielded 13 different spellings for the word beige. Some of these mistakes were the result of typos. Others were just that the entry person did not know how to spell the word. In some of the latter cases, they thought they knew how to spell the word, whereas in others they were just not able or willing to look it up.

Flawed Data Entry Processes

A lot of data entry begins with a form. A person completes a form either on a piece of paper or on a computer screen. Form design has a lot to do with the amount of inaccurate data that ends up in the database. Form design should begin with a basic understanding of quality issues in order to avoid many of the mistakes commonly seen. For example, having someone select from a list of valid values instead of typing in a value can eliminate the misspellings previously cited.

Another common problem is having fields on the form that are confusing to the user. This often leads them to enter wrong information. The field itself may be confusing to the user. If it is a field that is not commonly understood, or if the database definition is unconventional, the form needs to provide assistance in guiding the user through entry of values into the field. Sometimes the confusion is in the way the field is described in its identifying text or in its positioning on the form. Form design should always be subjected to rigorous quality testing to find the fields a normal user would have difficulty in knowing what to enter.

Data entry windows should have instructions available as HELP functions and should be user friendly in handling errors. Frustration in using a form can lead to deliberate mistakes that corrupt the database.

Forms are better completed by a trained entry person than by a one-time user. This is because the entry person can be taught how things should be entered, can become proficient in using the form mechanisms, and can be given feedback to improve the efficiency and accuracy of the data. A one-time user is always uncertain about what they are supposed to do on the form. Unfortunately, our society is moving by way of the Internet toward eliminating the middle person in the process and having end users complete forms directly. This places a much higher demand on quality form design.

The data entry process includes more than the forms that are filled out. It also includes the process that surrounds it. Forms are completed at a specific point or points in a process. Sometimes we have forms that are required to be completed when not all information is known or easily obtained at that point in the process. This will inevitably lead to quality problems.

An example of a data entry process I helped design a number of years ago for military repair personnel is very instructive of the types of problems that can occur in data collection. The U.S. Navy has a database that collects detailed information on the repair and routine maintenance performed on all aircraft and on all major components of every ship. This database is intended to be used for a variety of reasons, from negotiating contracts with suppliers, to validating warranties, to designing new aircraft and ships.

When an aircraft carrier is in a combat situation, such as in Kuwait and Afghanistan, repairs are being made frequently. The repair crews are working around the clock and under a great deal of pressure to deal with a lot of situations that come up unexpectedly. Completing forms is the least of their concerns. They have a tendency to fix things and do the paperwork later. The amount of undocumented work piles up during the day, to be completed when a spare moment is available. By then the repair person has forgotten some of the work done or the details of some of the work and certainly is in a hurry to get it done and out of the way.

Another part of this problem comes in when the data is actually entered from the forms. The forms are coming out of a hectic, very messy environment. Some of the forms are torn; some have oil or other substances on them. The writing is often difficult to decipher. The person who created it is probably not available and probably would not remember much about it if available.

A database built from this system will have many inaccuracies in it. Many of the inaccuracies will be missing information or valid but wrong information. An innovative solution that involves wireless, handheld devices and employs voice recognition technology would vastly improve the completeness and accuracy of this database. I hope the U.S. Navy has made considerable improvements in the data collection processes for this application since I left. I trust they have.

The Null Problem

A special problem occurs in data entry when the information called for is not available. A data element has a value, an indicator that the value is not known, or an indicator that no value exists (or is applicable) for this element in this record. Have you ever seen an entry screen that had room for a value and two indicator boxes you could use for the case where there is no value? I haven't. Most form designs either mandate that a value be provided or allow it to be left blank. If left blank, you do not know the difference between value-not-known and no-value-applies.

When the form requires that an entry be available and the entry person does not have the value, there is a strong tendency to "fake it" by putting a wrong, but acceptable, value into the field. This is even unintentionally encouraged for selection lists that have a default value in the field to start with.

It would be better form design to introduce the notion of NOT KNOWN or NOT APPLICABLE for data elements that are not crucial to the transaction being processed. This would at least allow the entry people to enter accurately what they know and the users of the data to understand what is going on in the data.

It would make sense in some cases to allow the initial entry of data to record NOT KNOWN values and have the system trigger subsequent activities that would collect and update these fields after the fact. This is far better than having people enter false information or leaving something blank and not knowing if a value exists for the field or not.

An example of a data element that may be NOT KNOWN or NOT APPLICABLE is a driver's license number. If the field is left blank, you cannot tell if it was not known at the point of entry or whether the person it applies to does not have a driver's license. Failure to handle the possibility of information not being available at the time of entry and failure to allow for options to express what you do know about a value leads to many inaccuracies in data.

Deliberate Errors

Deliberate errors are those that occur when the person enters a wrong value on purpose. There are three reasons they do this:

- They do not know the correct information.
- They do not want you to know the correct information.
- They get a benefit from entering the wrong information.

Do Not Know Correct Information

Not knowing the correct information occurs when the form requires a value for a field and the person wants or needs to complete the form but does not know the value to use. The form will not be complete without a value. The person does not believe the value is important to the transaction, at least not relative to what they are trying to do. The result is that they make up a value, enter the information, and go on.

Usually the information is not important to completing the transaction but may be important to other database users later on. For example, asking and requiring a value for the license plate number of your car when registering for a hotel has no effect on getting registered. However, it may be important when you leave your lights on and they need to find out whose car it is.

Do Not Wish To Give The Correct Information

The second source of deliberate errors is caused by the person providing the data not wanting to give the correct information. This is becoming a more and more common occurrence with data coming off the Internet and the emergence of CRM applications. Every company wants a database on all of their customers in order to tailor marketing programs. However, they end up with a lot of incorrect data in their databases because the information they ask people for is more than people are willing to provide or is perceived to be an invasion of privacy.

Examples of fields that people will lie about are age, height, weight, driver's license number, home phone number, marital status, annual income, and education level. People even lie about their name if it can get the result they want from the form without putting in their correct name. A common name appearing in many marketing databases is Mickey Mouse.

The problem with collecting data that is not directly required to complete the transaction is that the quality of these data elements tends to be low but is not immediately detected. It is only later, when you try to employ this data, that the inaccuracies show up and create problems.

Falsifying To Obtain A Benefit

The third case in which deliberate mistakes are made is where the entry person obtains an advantage in entering wrong data. Some examples from the real world illustrate this.

An automobile manufacturer receives claim forms for warranty repairs performed by dealers. Claims for some procedures are paid immediately, whereas claims for other procedures are paid in 60 days. The dealers figure out this scheme and deliberately lie about the procedures performed in order to get their money faster. The database incorrectly identifies the repairs made. Any attempt to use this database to determine failure rates would be a total failure. In fact, it was in attempts to use this data for this purpose that led to the discovery of the practice. It had been going on for years.

A bank gives branch bank employees a bonus for all new corporate accounts. A new division of a larger company opens an account with a local branch. If the bank employee determines that this is a sub-account of a larger, existing customer (the correct procedure), no bonus is paid upon opening the account. If, however, the account is opened as a new corporate customer (the wrong procedure), a bonus is paid.

An insurance company sells automobile insurance policies through independent insurance writers. In a metropolitan area, the insurance rate is determined by the Zip code of the applicant. The agents figure out that if they falsify the ZIP CODE field on the initial application for high-cost Zip codes, they can get the client on board at a lower rate. The transaction completes, the agent gets his commission, and the customer corrects the error when the renewal forms arrive a year later. The customer's rates subsequently go up as a result.

Data entry people are rated based on the number of documents entered per hour. They are not penalized for entering wrong information. This leads to a practice of entering data too fast, not attempting to resolve issues with input documents, and making up missing information. The operators who enter the poorest-quality data get the highest performance ratings.

All of these examples demonstrate that company policy can encourage people to deliberately falsify information in order to obtain a personal benefit.

- System Problems

Systems are too often blamed for mistakes when, after investigation, the mistakes turn out to be the result of a human error. Our computing systems have become enormously reliable over the years. However, database errors do occur because of system problems when the transaction systems are not properly designed.

Database systems have the notion of COMMIT. This means that changes to a database system resulting from an external transaction either get completely committed or completely rejected. Specific programming logic ensures that a partial transaction never occurs. In application designs, the user is generally made aware that a transaction has committed to the database.

In older systems, the transaction path from the person entering data to the database was very short. It usually consisted of a terminal passing information through a communications controller to a mainframe, where an application program made the database calls, performed a COMMIT, and sent a response back to the terminal. Terminals were either locally attached or accessed through an internal network.

Today, the transaction path can be very long and very complex. It is not unusual for an application to occur outside your corporation on a PC, over the Internet. The transaction flows through ISPs to an application server in your company. This server then passes messages to a database server, where the database calls are made. It is not unusual for multiple application servers to be in the path of the transaction. It is also not unusual for multiple companies to house application servers in the path. For example, Amazon passes transactions to other companies for "used book" orders.

The person entering the data is a nonprofessional, totally unfamiliar with the system paths. The paths themselves involve many parts, across many communication paths. If something goes wrong, such as a server going down, the person entering the information may not have any idea of whether the transaction occurred or not. If there is no procedure for them to find out, they often reenter the transaction, thinking it is not there, when in fact it is; or they do not reenter the transaction, thinking it happened, when in fact it did not. In one case, you have duplicate data; in the other, you have missing data.

More attention must be paid to transaction system design in this new, complex world we have created. We came pretty close to cleaning up transaction failures in older "short path" systems but are now returning to this problem with the newer "long path" systems.

In summary, there are plenty of ways data inaccuracies can occur when data is initially created. Errors that occur innocently tend to be random and are difficult to correct. Errors that are deliberate or are the result of poorly constructed processes tend to leave clues around that can be detected by analytical techniques.

Uses of data

1) Data in business

Data can help businesses better understand their customers, improve their advertising campaigns, personalize their content and improve their bottom lines. The advantages of data are many, but you can’t access these benefits without the proper data analytics tools and processes. While raw data has a lot of potential, you need data analytics to unlock the power to grow your business.

2)Data in healthcare

Data is extremely useful in the field of medical and healthcareDoctors are able to keep track about patient’s history, the link to which is only accessed by the patient and his particular physician.

The data of the patient is stored in the database safe and secure and as and when required by the doctor it can be viewed. Most of the medical devices are big data oriented. The use of data has gone to such an extent that doctors can check the person through the heart and temperature monitoring watch fitted on the patient’s hand and prescribe him with the following medicines

3) Data in media and entertainment

New business models are launched for different companies in the media and entertainment industry. The business model runs on collecting or creating the content, further to analyze it, then marketing and distribution of the content. As the rate of consumer’s search increases, there is a need for obtaining content at any moment, in whatever place in all formats on a variety of devices. We can runs through customer’s data along with observable data and gather even minute information to create a customer’s detailed profile. The benefits of big data in media and entertainment industry include forecasting what the target audience wants, planning optimization, expanding acquisition and retention; suggest content on demand and new

4) Data in Transportation

In latest times, enormous amounts of data from location-oriented social networks and high speed data from telecoms have influenced travel journeys.

5) Data in Banking

Banking is a very crucial sector. The security, privacy, management and maintenance of any banking system is a challenge. Data here is very beneficial and helps in the fraud detection in banking system. Using Big Data , we can searches all the illegal activities that has taken place and can identifies the misuse of credit and debit cards, business precision, customer statistics modification, public analytics for business.

Data Objects and Attribute Types

Data sets 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 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. In this section, we define attributes and look at the various attribute types.

ATTRIBUTE:

An attribute is a data field, representing a characteristic or feature of a data object.

The nouns attribute, dimension, feature, and variable are often used interchangeably in the literature.

The term dimension is commonly used in data warehousing. Machine learning literature tends to use the term feature, while statisticians prefer the term variable. For example, customer ID, name, and address.

Observed values for a given attribute are known as observations. The type of an attribute is determined by the set of possible values—nominal, binary, ordinal, or numeric—the attribute can have. In the following subsections, we introduce each type.

Nominal Attributes

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.

Examples: Nominal attributes. Suppose that hair color and marital status are two attributes describing person objects. In our application, 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 Attributes:

A binary attribute is a nominal attribute with only two categories or states: 0 or 1, where 0 typically means that the attribute is absent, and 1 means that it is present. Binary attributes are referred to as Boolean if the two states correspond to true and false.Examples: Binary attributes. Given the attribute smoker describing a patient object, 1 indicates that the patient smokes, while 0 indicates that the patient does not. Similarly, suppose the patient undergoes a medical test that has two possible outcomes. The attribute medical test is binary, where a value of 1 means the result of the test for the patient is positive, while 0 means the result is negative.

Ordinal Attributes

An ordinal attribute is an attribute with possible values that have a meaningful order or ranking among them, but the magnitude between successive values is not known.

Examples: Ordinal attributes. Suppose that 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.

$+ \quad - \quad +$

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*. Professional ranks can be enumerated in a sequential order: for example, *assistant*, *associate*, and *full* for professors, and *private, private first class, specialist, corporal, and sergeant* for army ranks.

If integers are used, they represent computer codes for the categories, as opposed to measurable quantities (e.g., 0 for *small* drink size, 1 for *medium*, and 2 for *large*). In the following subsec- tion we look at numeric attributes, which provide *quantitative* measurements of an object.

Numeric Attributes

A **numeric attribute** is *quantitative*; that is, it is a measurable quantity, represented in integer or real values. Numeric attributes can be *interval-scaled* or *ratio-scaled*.

Interval-Scaled Attributes

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.

Examples Interval-scaled attributes. 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. Calendar dates are another example. For instance, the years 2002 and 2010 are eight years apart.

Ratio-Scaled Attributes

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.

Examples: Ratio-scaled attributes. Unlike temperatures in Celsius and Fahrenheit, the Kelvin (K) temperature scale has what is considered a true zero-point ($0^{\circ}K = -273.15^{\circ}C$): It is the point at which the particles that comprise matter have zero kinetic energy.

Discrete versus Continuous Attribute

In our presentation, we have organized attributes into nominal, binary, ordinal, and numeric types. There are many ways to organize attribute types. The types are not mutually exclusive. Classification algorithms developed from the field of machine learning often talk of attributes as being either *discrete* or *continuous*. Each type may be processed differently. A **discrete attribute** has a finite or countably infinite set of values, which may or may not be represented as integers. The attributes *hair color*, *smoker*, *medical test*, and *drink size* each have a finite number of values, and so are discrete.

If an attribute is not discrete, it is **continuous**. The terms *numeric attribute* and *continuous attribute* are often used interchangeably in the literature. (This can be confusing because, in the classic sense, continuous values are real numbers, whereas numeric val- ues can be either integers or real numbers.) In practice, real values are represented using a finite number of digits. Continuous attributes are typically represented as floating-point variables.

Basic Statistical Descriptions of Data

For data preprocessing to be successful, it is essential to have an overall picture of your data. Basic statistical descriptions can be used to identify properties of the data and highlight which data values should be treated as noise or outliers.

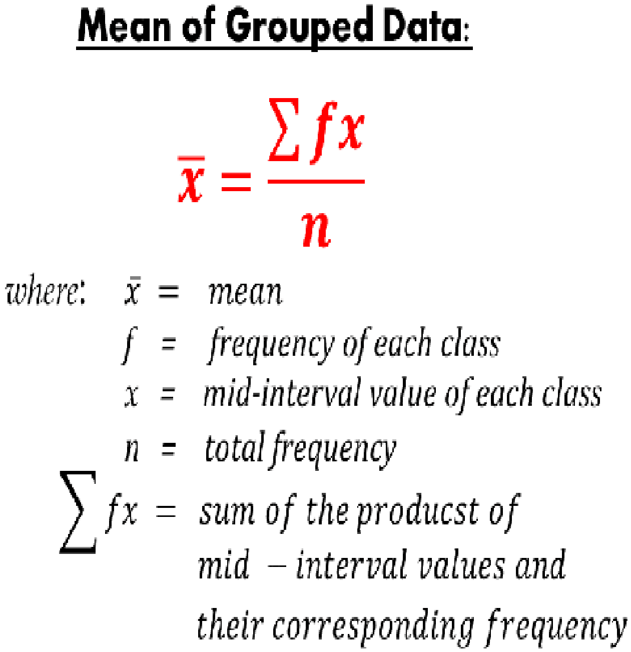
This section discusses three areas of basic statistical descriptions. We start with *measures of central tendency* which measure the location of the middle or center of a data distribution. Intuitively speaking, given an attribute, where do most of its values fall? In particular, we discuss the mean, median, mode, and midrange.

In addition to assessing the central tendency of our data set, we also would like to have an idea of the *dispersion of the data*. That is, how are the data spread out? The most common data dispersion measures are the *range*, *quartiles*, and *interquartile range*; the *five-number summary* and *boxplots*; and the *variance* and *standard deviation* of the data These measures are useful for identifying outliers. Other popular displays of data summaries and distributions include *quantile plots*, *quantile–quantile plots*, *histograms*, and *scatter plots*.

Measuring the Central Tendency: Mean, Median, and Mode

In this section, we look at various ways to measure the central tendency of data. Suppose that we have some attribute X, like *salary*, which has been recorded for a set of objects. Let $x_1,x_2,\ldots,x_N$ be the set of N observed values or *observations* for X. Here, these val- ues may also be referred to as the data set (for X). If we were to plot the observations for *salary*, where would most of the values fall? This gives us an idea of the central ten- dency of the data. Measures of central tendency include the mean, median, mode, and midrange.

The most common and effective numeric measure of the "center" of a set of data is the *(arithmetic) mean*. Let $x_1,x_2,\ldots,x_N$ be a set of N values or *observations*, such as for some numeric attribute X, like *salary*. The **mean** of this set of values is



Mean of Grouped Data:

$$\bar{x} = \frac{\sum fx}{n}$$

where: $\bar{x}$ = mean

f = frequency of each class

x = mid-interval value of each class

n = total frequency

$\sum fx$ = sum of the producst of mid − interval values and their corresponding frequency

Median: The <u>median</u> is the **middle number** in a data set. To find the median, list your data points in ascending order and then find the middle number. The middle number in this set is 28 as there are 4 numbers below it and 4 numbers above:

23, 24, 26, 26, *28*, 29, 30, 31, 33

Note: If you have an even set of numbers, average the middle two to find the median. For example, the median of this set of numbers is 28.5 (28 + 29 / 2).

23, 24, 26, 26, *28*, *29*, 30, 31, 33, 34

MODE: The mode is another measure of central tendency. The **mode** for a set of data is the value that occurs most frequently in the set. Therefore, it can be determined for qualita- tive and quantitative attributes. It is possible for the greatest frequency to correspond to several different values, which results in more than one mode. Data sets with one, two, or three modes are respectively called **unimodal**, **bimodal**, and **trimodal**. In general, a data set with two or more modes is **multimodal**. At the other extreme, if each data value occurs only once, then there is no mode.

Example: Mode. The data from Example are bimodal. The two modes are $52,000 and $70,000. ■

For unimodal numeric data that are moderately skewed (asymmetrical), we have the following empirical relation:

$$mean - mode \approx 3 \times (mean - median).$$

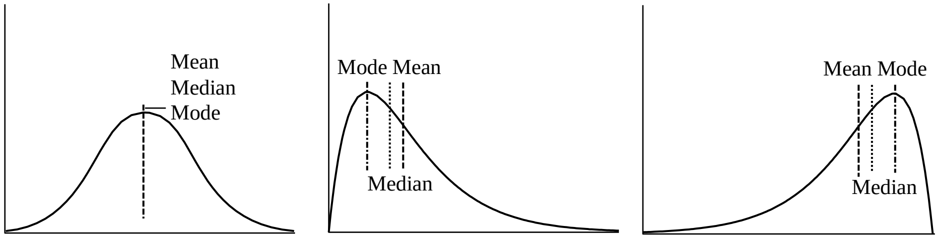
This implies that the mode for unimodal frequency curves that are moderately skewed can easily be approximated if the mean and median values are known.

The **midrange** can also be used to assess the central tendency of a numeric data set. It is the average of the largest and smallest values in the set. This measure is easy to compute using the SQL aggregate functions, max() and min().

Example :Midrange. The midrange of the data of Example 2.6 is $\frac{30,000+110,000}{2}$ = $70,000. ■

In a unimodal frequency curve with perfect **symmetric** data distribution, the mean, median, and mode are all at the same center value, as shown in Figure 2.1(a).

Data in most real applications are not symmetric. They may instead be either **posi- tively skewed**, where the mode occurs at a value that is smaller than the median (Figure2.1b), or **negatively skewed**, where the mode occurs at a value greater than the median (Figure 2.1c).



(a) Symmetric data **(b)** Positively skewed data **(c)** Negatively skewed data

Figure Mean, median, and mode of symmetric versus positively and negatively skewed data.

Measuring the Dispersion of Data: Range, Quartiles, Variance, Standard Deviation, and Interquartile Range

We now look at measures to assess the dispersion or spread of numeric data. The measures include range, quantiles, quartiles, percentiles, and the interquartile range. The five-number summary, which can be displayed as a boxplot, is useful in identifying outliers. Variance and standard deviation also indicate the spread of a data distribution.

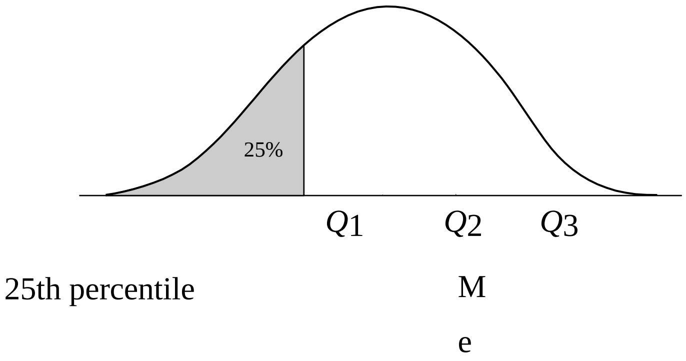
Range, Quartiles, and Interquartile Range

To start off, let's study the *range*, *quantiles*, *quartiles*, *percentiles*, and the *interquartile range* as measures of data dispersion.

Let $x_1, x_2, \ldots, x_N$ be a set of observations for some numeric attribute, X. The **range** of the set is the difference between the largest (max()) and smallest (min()) values.

Suppose that the data for attribute X are sorted in increasing numeric order. Imagine that we can pick certain data points so as to split the data distribution into equal-size consecutive sets, as in Figure 2.2. These data points are called *quantiles*. **Quantiles** are points taken at regular intervals of a data distribution, dividing it into essentially equal- size consecutive sets. (We say "essentially" because there may not be data values of *X* that divide the data into exactly equal-sized subsets. For readability, we will refer to them as equal.) The *k*th *q-quantile* for a given data distribution is the value *x* such that at most k/q of the data values are less than *x* and at most $(q \quad k)/q$ of the data values are more than *x*, where *k* is an integer such that $0 < k < q$. There are q 1 q-quantiles.

The 2-quantile is the data point dividing the lower and upper halves of the data distribution. It corresponds to the median. The 4-quantiles are the three data points that split the data distribution into four equal parts; each part represents one-fourth of the data distribution. They are more commonly referred to as **quartiles**. The 100-quantiles are more commonly referred to as **percentiles**; they divide the data distribution into 100 equal-sized consecutive sets. The median, quartiles, and percentiles are the most widely used forms of quantiles.



A plot of the data distribution for some attribute *X*. The quantiles plotted are quartiles. The three quartiles divide the distribution into four equal-size consecutive subsets. The second quartile corresponds to the median.

The quartiles give an indication of a distribution's center, spread, and shape. The **first quartile**, denoted by Q_1, is the 25th percentile. It cuts off the lowest 25% of the data. The **third quartile**, denoted by Q_3, is the 75th percentile—it cuts off the lowest 75% (or highest 25%) of the data. The second quartile is the 50th percentile. As the median, it gives the center of the data distribution.

The distance between the first and third quartiles is a simple measure of spread that gives the range covered by the middle half of the data. This distance is called the **interquartile range** (**IQR**) and is defined as $IQR = Q_3 - Q_1$.

Example : Interquartile range. The quartiles are the three values that split the sorted data set into four equal parts. The data of Example 2.6 contain 12 observations, already sorted in increasing order. Thus, the quartiles for this data are the third, sixth, and ninth val- ues, respectively, in the sorted list. Therefore, Q_1 \$47,000 and Q_3 is \$63,000. Thus, the interquartile range is IQR 63 47 \$16,000. (Note that the sixth value is a median, \$52,000, although this data set has two medians since the number of data values is even.) ■

Five-Number Summary, Boxplots, and Outliers

No single numeric measure of spread (e.g., *IQR*) is very useful for describing skewed distributions. Have a look at the symmetric and skewed data distributions of Figure. In the symmetric distribution, the median (and other measures of central tendency) splits the data into equal-size halves. This does not occur for skewed distributions. Therefore, it is more informative to also provide the two quartiles Q_1 and Q_3, along with the median. A common rule of thumb for identifying suspected **outliers** is to single out values falling at least 1.5 *IQR* above the third quartile or below the first quartile.

Because Q_1, the median, and Q_3 together contain no information about the end- points (e.g., tails) of the data, a fuller summary of the shape of a distribution can be obtained by providing the lowest and highest data values as well. This is known as the *five-number summary*. The **five-number summary** of a distribution consists of the median (Q_2), the quartiles Q_1 and Q_3, and the smallest and largest individual obser- vations, written in the order of *Minimum*, Q_1, *Median*, Q_3, *Maximum*.

Boxplots are a popular way of visualizing a distribution. A boxplot incorporates the five-number summary as follows:

Typically, the ends of the box are at the quartiles so that the box length is the interquartile range.

The median is marked by a line within the box.

Two lines (called *whiskers*) outside the box extend to the smallest (*Minimum*) and largest (*Maximum*) observations.

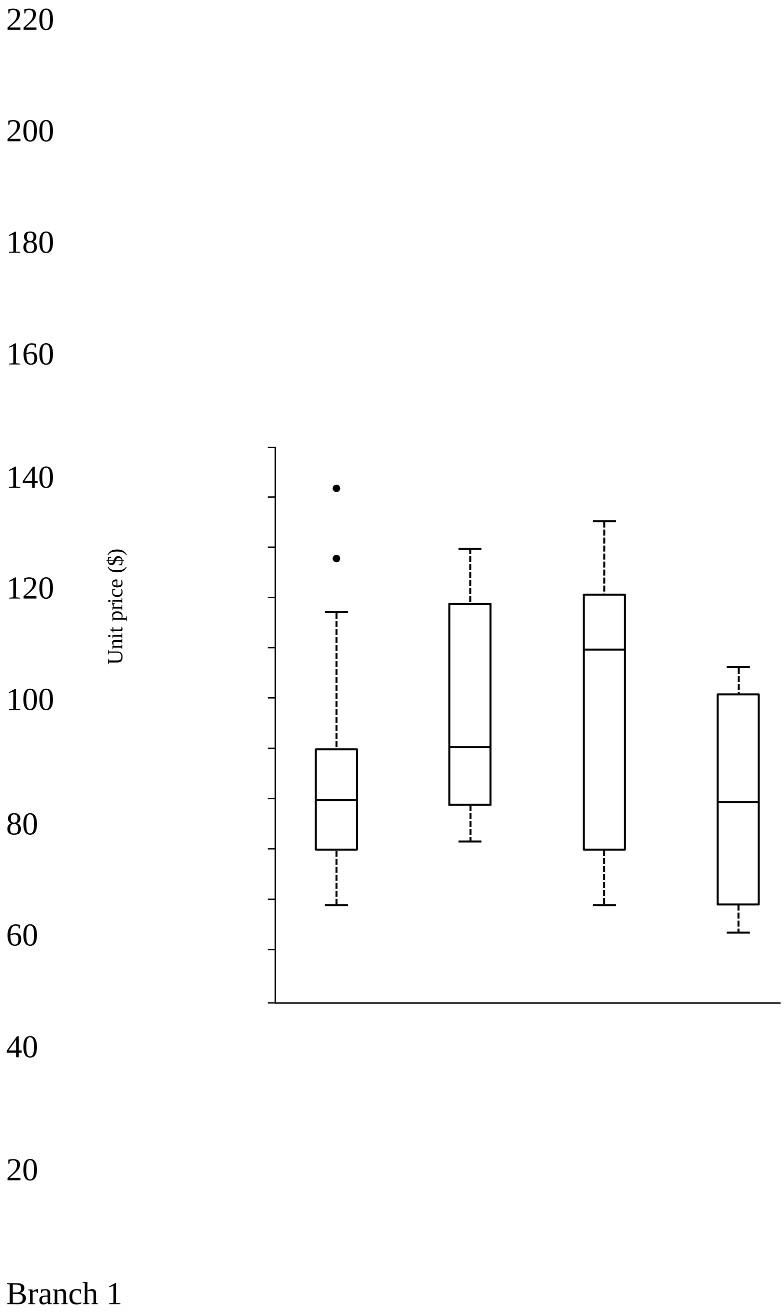

220
200
180
160
140
120
100
80
60
40
20
Unit price ($)
Branch 1

Figure Boxplot for the unit price data for items sold at four branches of *AllElectronics* during a given time period.

When dealing with a moderate number of observations, it is worthwhile to plot potential outliers individually. To do this in a boxplot, the whiskers are extended to the extreme low and high observations *only if* these values are less than 1.5 *IQR* beyond the quartiles. Otherwise, the whiskers terminate at the most extreme observations occurring within 1.5 *IQR* of the quartiles. The remaining cases are plotted individually. Boxplots can be used in the comparisons of several sets of compatible data.

Example **Boxplot.** Figure shows boxplots for unit price data for items sold at four branches of *AllElectronics* during a given time period. For branch 1, we see that the median price of items sold is \$80, Q_1 is \$60, and Q_3 is \$100. Notice that two outlying observations for this branch were plotted individually, as their values of 175 and 202 are more than 1.5 times the IQR here of 40. ■

Boxplots can be computed in $O(n \log n)$ time. Approximate boxplots can be com- puted in linear or sublinear time depending on the quality guarantee required.

Variance and Standard Deviation

Variance and standard deviation are measures of data dispersion. They indicate how spread out a data distribution is. A low standard deviation means that the data observations tend to be very close to the mean, while a high standard deviation indicates that the data are spread out over a large range of values.

where $\bar{x}$ is the mean value of the observations, as defined in Eq. (2.1). The **standard deviation**, σ, of the observations is the square root of the variance, σ^2.

- **Example Variance and standard deviation.** In Example 2.6, we found $\bar{x} = \$58{,}000$ using Eq. (2.1) for the mean. To determine the variance and standard deviation of the data from that example, we set $N = 12$ and use Eq. (2.6) to obtain

$$\sigma^2 = \frac{1}{12}(30^2 + 36^2 + 47^2 \ldots + 110^2) - 58^2$$

$$\approx 379.17$$

$$\sigma \approx \sqrt{379.17} \approx 19.47. \qquad \blacksquare$$

The basic properties of the standard deviation, σ, as a measure of spread are as follows:

σ measures spread about the mean and should be considered only when the mean is chosen as the measure of center.

$\sigma = 0$ only when there is no spread, that is, when all observations have the same value. Otherwise, $\sigma > 0$.

Importantly, an observation is unlikely to be more than several standard deviations away from the mean. Mathematically, using Chebyshev's inequality, it can be shown that at least $\left(1 - \frac{1}{k^2}\right) \times 100\%$ of the observations are no more than k standard deviations from the mean. Therefore, the standard deviation is a good indicator of the spread of a data set.

The computation of the variance and standard deviation is scalable in large databases.

Graphic Displays of Basic Statistical Descriptions of Data

In this section, we study graphic displays of basic statistical descriptions. These include *quantile plots*, *quantile–quantile plots*, *histograms*, and *scatter plots*. Such graphs are help- ful for the visual inspection of data, which is useful for data preprocessing. The first three of these show univariate distributions (i.e., data for one attribute), while scatter plots show bivariate distributions (i.e., involving two attributes).

Quantile Plot

In this and the following subsections, we cover common graphic displays of data distributions. A **quantile plot** is a simple and effective way to have a first look at a univariate data distribution. First, it displays all of the data for the given attribute to assess both the overall behavior and unusual occurrences). Second, it plots quantile information (see Section 2.2.2). Let x_i, for $i = 1$ to N, be the data sorted in increasing order so that x_1 is the smallest observation and x_N is the largest for some ordinal or numeric attribute X. Each observation, x_i, is paired with a percentage, f_i, which indicates

that approximately $f_i \times 100\%$ of the data are below the value, x_i. We say "approximately" because there may not be a value with exactly a fraction, f_i, of the data below x_i. Note that the 0.25 percentile corresponds to quartile Q_1, the 0.50 percentile is the median, and the 0.75 percentile is Q_3.

Let

$$f_i = \frac{i-0.5}{N}.$$

These numbers increase in equal steps of $1/N$, ranging from $\frac{1}{2N}$ (which is slightly

Figure A q-q plot for unit price data from two *AllElectronics* branches.

data, which is plotted against the *(i* – 0.5)*/M* quantile of the *x* data. This computation typically involves interpolation.

- **Example Quantile–quantile plot.** Figure shows a quantile–quantile plot for *unit* *price* data of items sold at two branches of *AllElectronics* during a given time period. Each point cor- responds to the same quantile for each data set and shows the unit price of items sold at branch 1 versus branch 2 for that quantile. (To aid in comparison, the straight line represents the case where, for each given quantile, the unit price at each branch is the same. The darker points correspond to the data for Q_1, the median, and Q_3, respectively.)

We see, for example, that at Q_1, the unit price of items sold at branch 1 was slightly less than that at branch 2. In other words, 25% of items sold at branch 1 were less than or equal to \$60, while 25% of items sold at branch 2 were less than or equal to \$64. At the 50th percentile (marked by the median, which is also Q_2), we see that 50% of items sold at branch 1 were less than \$78, while 50% of items at branch 2 were less than \$85. In general, we note that there is a shift in the distribution of branch 1 with respect to branch 2 in that the unit prices of items sold at branch 1 tend to be lower than those at branch 2.

Histograms

Histograms (or **frequency histograms**) are at least a century old and are widely used. "Histos" means pole or mast, and "gram" means chart, so a histogram is a chart of poles. Plotting histograms is a gr̄aphical methōd for summarizing the distribution of a given attribute, *X*. If *X* is nominal, such as *automobile model* or *item type*, then a pole or vertical bar is drawn for each known value of *X*. The height of the bar indicates the frequency (i.e., count) of that *X* value. The resulting graph is more commonly known as a **bar chart**.

If *X* is numeric, the term *histogram* is preferred. The range of values for *X* is partitioned into disjoint consecutive subranges. The subranges, referred to as *buckets* or *bins*, are disjoint subsets of the data distribution for *X*. The range of a bucket is known as the **width**. Typically, the buckets are of equal width. For example, a *price* attribute with a value range of \$1 to \$200 (rounded up to the nearest dollar) can be partitioned into subranges 1 to 20, 21 to 40, 41 to 60, and so on. For each subrange, a bar is drawn with a height that represents the total count of items observed within the subrange. Histograms and partitioning rules are further discussed in Chapter 3 on data reduction.

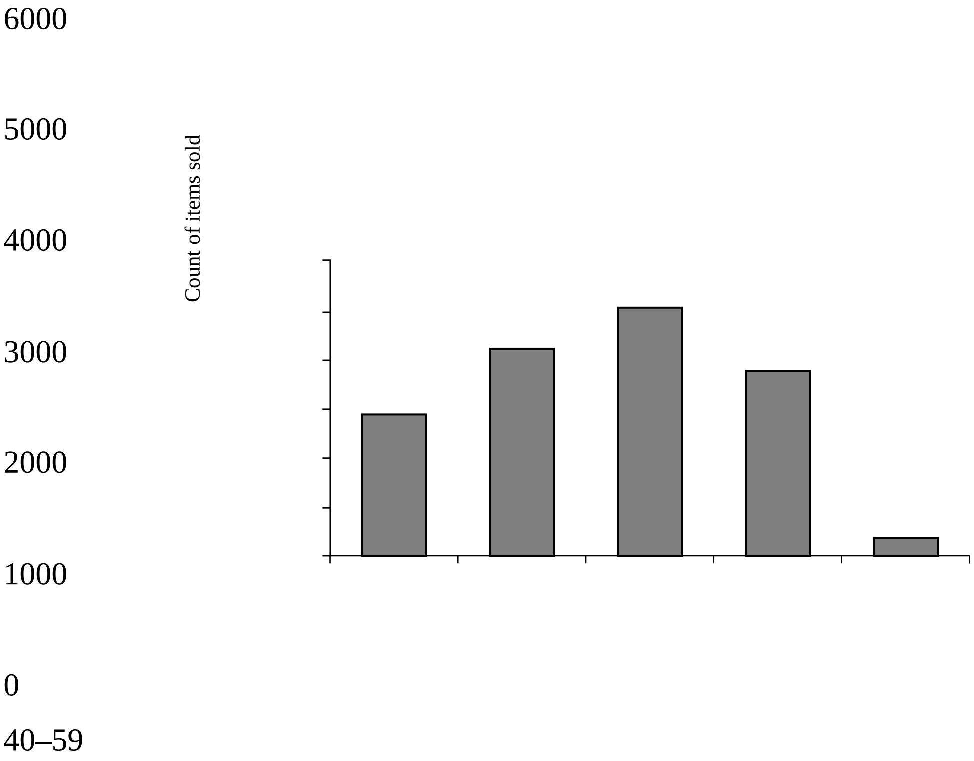
- **Example Histogram.** shows a histogram for the data set of where buckets (or bins) are defined by equal-width ranges representing \$20 increments and the frequency is the count of items sold.

Although histograms are widely used, they may not be as effective as the quantile plot, q-q plot, and boxplot methods in comparing groups of univariate observations.

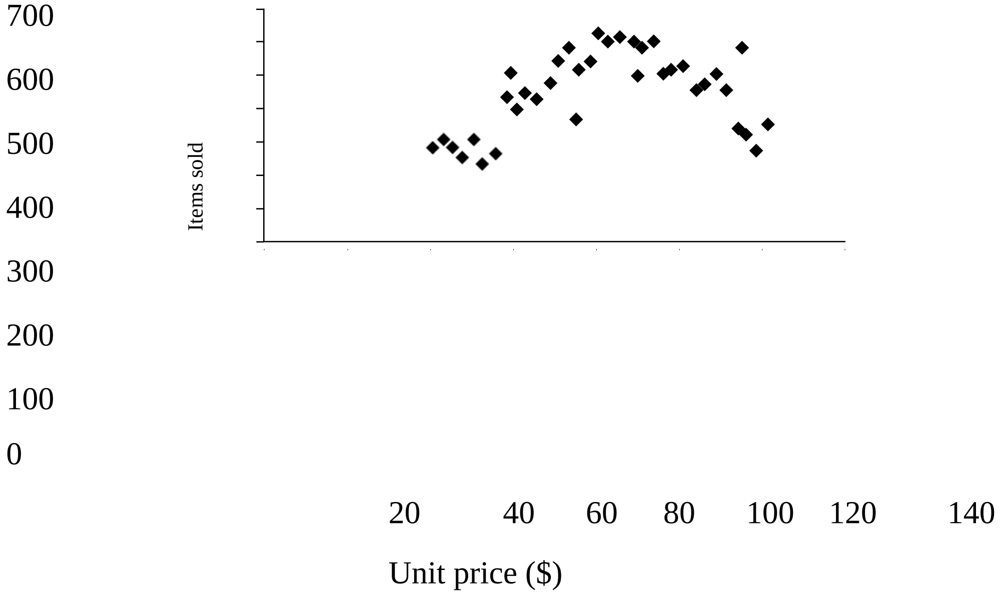
Scatter Plots and Data Correlation

A **scatter plot** is one of the most effective graphical methods for determining if there appears to be a relationship, pattern, or trend between two numeric attributes. To con- struct a scatter plot, each pair of values is treated as a pair of coordinates in an algebraic sense and plotted as points in the plane. Figure 2.7 shows a scatter plot for the set of data in Table 2.1.

The scatter plot is a useful method for providing a first look at bivariate data to see clusters of points and outliers, or to explore the possibility of correlation relationships. Two attributes, *X*, and *Y* , are **correlated** if one attribute implies the other. Correlations can be positive, negative, or null (uncorrelated). Figure 2.8 shows examples of positive and negative correlations between two attributes. If the plotted points pattern slopes



A histogram for the Table data set.



A scatter plot for the Table data set.

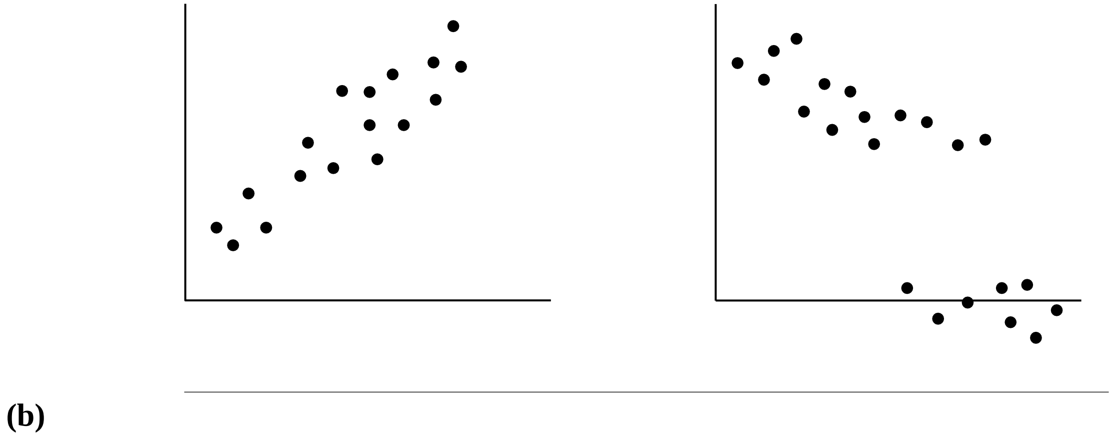


Figure Scatter plots can be used to find (a) positive or (b) negative correlations between attributes.

Data Visualization

Data visualization aims to communicate data clearly and effectively through graphical representation. Data visualization has been used extensively in many applications—for example, at work for reporting, managing business operations, and tracking progress of tasks

Pixel-Oriented Visualization Techniques

A simple way to visualize the value of a dimension is to use a pixel where the color of the pixel reflects the dimension's value. For a data set of m dimensions, **pixel-oriented techniques** create m windows on the screen, one for each dimension. The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows. The colors of the pixels reflect the corresponding values.

Inside a window, the data values are arranged in some global order shared by all windows. The global order may be obtained by sorting all data records in a way that's meaningful for the task at hand.

Geometric Projection Visualization Techniques

A drawback of pixel-oriented visualization techniques is that they cannot help us much in understanding the distribution of data in a multidimensional space. For example, they do not show whether there is a dense area in a multidimensional subspace. **Geometric projection techniques** help users find interesting projections of multidimensional data sets. The central challenge the geometric projection techniques try to address is how to visualize a high-dimensional space on a 2-D display.

A **scatter plot** displays 2-D data points using Cartesian coordinates. A third dimen- sion can be added using different colors or shapes to represent different data points. Figure 2.13 shows an example, where X and Y are two spatial attributes and the third dimension is represented by different shapes. Through this visualization, × we can see that points of types "+" and " " tend to be colocated. A 3-D scatter plot uses three axes in a Cartesian coordinate system.

Icon-Based Visualization Techniques

Icon-based visualization techniques use small icons to represent multidimensional data values. We look at two popular icon-based techniques: Chernoff faces and stick figures.

Chernoff faces were introduced in 1973 by statistician Herman Chernoff. They dis- play multidimensional data of up to 18 variables (or dimensions) as a cartoon human face . Chernoff faces help reveal trends in the data. Components of the face, such as the eyes, ears, mouth, and nose, represent values of the dimensions by their shape, size, placement, and orientation. For example, dimensions can be mapped to the following facial characteristics: eye size, eye spacing, nose length, nose width, mouth curvature, mouth width, mouth openness, pupil size, eyebrow slant, eye eccentricity, and head eccentricity.

Chernoff faces make use of the ability of the human mind to recognize small differences in facial characteristics and to assimilate many facial characteristics at once.

Hierarchical Visualization Techniques

The visualization techniques discussed so far focus on visualizing multiple dimensions simultaneously. However, for a large data set of high dimensionality, it would be diffi- cult to visualize all dimensions at the same time. **Hierarchical visualization techniques** partition all dimensions into subsets (i.e., subspaces). The subspaces are visualized in a hierarchical manner.

“Worlds-within-Worlds,” also known as n-Vision, is a representative hierarchical visualization method. Suppose we want to visualize a 6-D data set, where the dimensions are F, $X_1,\ldots,X_5$. We want to observe how dimension F changes with respect to the other dimensions. We can first fix the values of dimensions X_3, X_4, X_5 to some selected values, say, c_3, c_4, c_5. We can then visualize F, X_1, X_2 using a 3-D plot, called a world, as shown in Figure 2.19. The position of the origin of the inner world is located at the point (c_3, c_4, c_5) in the outer world, which is another 3-D plot using dimensions X_3, X_4, X_5. A user can interactively change, in the outer world, the location of the origin of the inner world. The user then views the resulting changes of the inner world. Moreover, a user can vary the dimensions used in the inner world and the outer world. Given more dimensions, more levels of worlds can be used, which is why the method is called “worlds-within- worlds.”

As another example of hierarchical visualization methods, **tree-maps** display hierarchical data as a set of nested rectangles.

Visualizing Complex Data and Relations

In early days, visualization techniques were mainly for numeric data. Recently, more and more non-numeric data, such as text and social networks, have become available. Visualizing and analyzing such data attracts a lot of interest.

There are many new visualization techniques dedicated to these kinds of data. For example, many people on the Web tag various objects such as pictures, blog entries, and product reviews. A **tag cloud** is a visualization of statistics of user-generated tags. Often, in a tag cloud, tags are listed alphabetically or in a user-preferred order. The importance of a tag is indicated by font size or color