3

Exploring Data

The previous chapter addressed high-level data issues that are important in the knowledge discovery process. This chapter provides an introduction to **data exploration**, which is a preliminary investigation of the data in order to better understand its specific characteristics. Data exploration can aid in selecting the appropriate preprocessing and data analysis techniques. It can even address some of the questions typically answered by data mining. For example, patterns can sometimes be found by visually inspecting the data. Also, some of the techniques used in data exploration, such as visualization, can be used to understand and interpret data mining results.

This chapter covers three major topics: summary statistics, visualization, and On-Line Analytical Processing (OLAP). Summary statistics, such as the mean and standard deviation of a set of values, and visualization techniques, such as histograms and scatter plots, are standard methods that are widely employed for data exploration. OLAP, which is a more recent development, consists of a set of techniques for exploring multidimensional arrays of values. OLAP-related analysis functions focus on various ways to create summary data tables from a multidimensional data array. These techniques include aggregating data either across various dimensions or across various attribute values. For instance, if we are given sales information reported according to product, location, and date, OLAP techniques can be used to create a summary that describes the sales activity at a particular location by month and product category.

The topics covered in this chapter have considerable overlap with the area known as **Exploratory Data Analysis** (EDA), which was created in the 1970s by the prominent statistician, John Tukey. This chapter, like EDA, places a heavy emphasis on visualization. Unlike EDA, this chapter does not include topics such as cluster analysis or anomaly detection. There are two reasons for this. First, data mining views descriptive data analysis techniques as an end in themselves, whereas statistics, from which EDA originated, tends to view hypothesis-based testing as the final goal. Second, cluster analysis and anomaly detection are large areas and require full chapters for an in-depth discussion. Hence, cluster analysis is covered in Chapters 8 and 9, while anomaly detection is discussed in Chapter 10.

3.1 The Iris Data Set

In the following discussion, we will often refer to the Iris data set that is available from the University of California at Irvine (UCI) Machine Learning Repository. It consists of information on 150 Iris flowers, 50 each from one of three Iris species: Setosa, Versicolour, and Virginica. Each flower is characterized by five attributes:

1. sepal length in centimeters
2. sepal width in centimeters
3. petal length in centimeters
4. petal width in centimeters
5. class (Setosa, Versicolour, Virginica)

The sepals of a flower are the outer structures that protect the more fragile parts of the flower, such as the petals. In many flowers, the sepals are green, and only the petals are colorful. For Irises, however, the sepals are also colorful. As illustrated by the picture of a Virginica Iris in Figure 3.1, the sepals of an Iris are larger than the petals and are drooping, while the petals are upright.

3.2 Summary Statistics

Summary statistics are quantities, such as the mean and standard deviation, that capture various characteristics of a potentially large set of values with a single number or a small set of numbers. Everyday examples of summary statistics are the average household income or the fraction of college students who complete an undergraduate degree in four years. Indeed, for many people, summary statistics are the most visible manifestation of statistics. We will concentrate on summary statistics for the values of a single attribute, but will provide a brief description of some multivariate summary statistics.

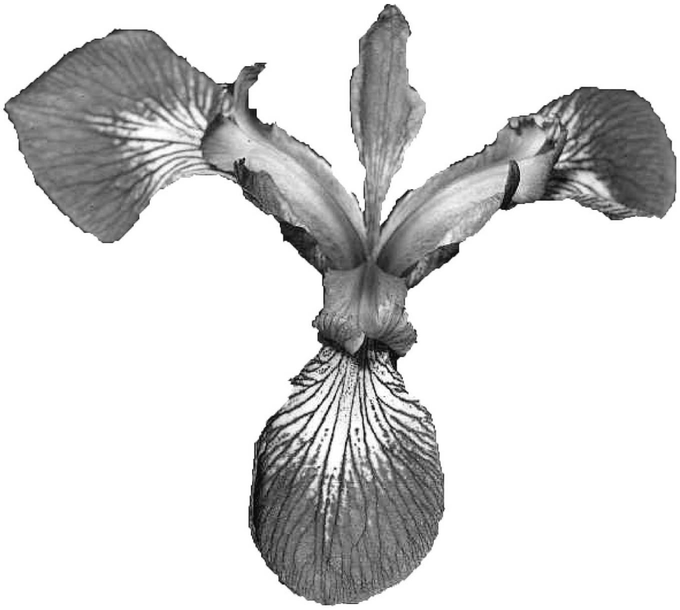


Figure 3.1. Picture of Iris Virginica. Robert H. Mohlenbrock @ USDA-NRCS PLANTS Database/ USDA NRCS. 1995. Northeast wetland flora: Field office guide to plant species. Northeast National Technical Center, Chester, PA. Background removed.

This section considers only the descriptive nature of summary statistics. However, as described in Appendix C, statistics views data as arising from an underlying statistical process that is characterized by various parameters, and some of the summary statistics discussed here can be viewed as estimates of statistical parameters of the underlying distribution that generated the data.

3.2.1 Frequencies and the Mode

Given a set of unordered categorical values, there is not much that can be done to further characterize the values except to compute the frequency with which each value occurs for a particular set of data. Given a categorical attribute x, which can take values $\{v_1, \ldots, v_i, \ldots v_k\}$ and a set of m objects, the frequency of a value v_i is defined as

$$\text{frequency}(v_i) = \frac{\text{number of objects with attribute value } v_i}{m}. \tag{3.1}$$

The **mode** of a categorical attribute is the value that has the highest frequency.

Example 3.1. Consider a set of students who have an attribute, *class*, which can take values from the set $\{freshman$, *sophomore*, $junior$, *senior*$\}$. Table 3.1 shows the number of students for each value of the *class* attribute. The mode of the *class* attribute is $freshman$, with a frequency of 0.33. This may indicate dropouts due to attrition or a larger than usual freshman class.

Table 3.1. Class size for students in a hypothetical college.

Class	Size	Frequency
freshman	200	0.33
sophomore	160	0.27
junior	130	0.22
senior	110	0.18

■

Categorical attributes often, but not always, have a small number of values, and consequently, the mode and frequencies of these values can be interesting and useful. Notice, though, that for the Iris data set and the *class* attribute, the three types of flower all have the same frequency, and therefore, the notion of a mode is not interesting.

For continuous data, the mode, as currently defined, is often not useful because a single value may not occur more than once. Nonetheless, in some cases, the mode may indicate important information about the nature of the values or the presence of missing values. For example, the heights of 20 people measured to the nearest millimeter will typically not repeat, but if the heights are measured to the nearest tenth of a meter, then some people may have the same height. Also, if a unique value is used to indicate a missing value, then this value will often show up as the mode.

3.2.2 Percentiles

For ordered data, it is more useful to consider the **percentiles** of a set of values. In particular, given an ordinal or continuous attribute x and a number p between 0 and 100, the p^{th} percentile x_p is a value of x such that p% of the observed values of x are less than x_p. For instance, the 50^{th} percentile is the value $x_{50\%}$ such that 50% of all values of x are less than $x_{50\%}$. Table 3.2 shows the percentiles for the four quantitative attributes of the Iris data set.

Table 3.2. Percentiles for sepal length, sepal width, petal length, and petal width. (All values are in centimeters.)

Percentile	Sepal Length	Sepal Width	Petal Length	Petal Width
0	4.3	2.0	1.0	0.1
10	4.8	2.5	1.4	0.2
20	5.0	2.7	1.5	0.2
30	5.2	2.8	1.7	0.4
40	5.6	3.0	3.9	1.2
50	5.8	3.0	4.4	1.3
60	6.1	3.1	4.6	1.5
70	6.3	3.2	5.0	1.8
80	6.6	3.4	5.4	1.9
90	6.9	3.6	5.8	2.2
100	7.9	4.4	6.9	2.5

Example 3.2. The percentiles, $x_{0\%}, x_{10\%}, \ldots, x_{90\%}, x_{100\%}$ of the integers from 1 to 10 are, in order, the following: 1.0, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5, 10.0. By tradition, $x_{0\%} = \min(x)$ and $x_{100\%} = \max(x)$. ■

3.2.3 Measures of Location: Mean and Median

For continuous data, two of the most widely used summary statistics are the **mean** and **median**, which are measures of the *location* of a set of values. Consider a set of m objects and an attribute x. Let $\{x_1, \ldots, x_m\}$ be the attribute values of x for these m objects. As a concrete example, these values might be the heights of m children. Let $\{x_{(1)}, \ldots, x_{(m)}\}$ represent the values of x after they have been sorted in non-decreasing order. Thus, $x_{(1)} = \min(x)$ and $x_{(m)} = \max(x)$. Then, the mean and median are defined as follows:

$$\text{mean}(x) = \overline{x} = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{3.2}$$

$$\text{median}(x) = \begin{cases} x_{(r+1)} & \text{if } m \text{ is odd, i.e., } m = 2r+1 \\ \frac{1}{2}(x_{(r)} + x_{(r+1)}) & \text{if } m \text{ is even, i.e., } m = 2r \end{cases} \tag{3.3}$$

To summarize, the median is the middle value if there are an odd number of values, and the average of the two middle values if the number of values is even. Thus, for seven values, the median is $x_{(4)}$, while for ten values, the median is $\frac{1}{2}(x_{(5)} + x_{(6)})$.

Although the mean is sometimes interpreted as the middle of a set of values, this is only correct if the values are distributed in a symmetric manner. If the distribution of values is skewed, then the median is a better indicator of the middle. Also, the mean is sensitive to the presence of outliers. For data with outliers, the median again provides a more robust estimate of the middle of a set of values.

To overcome problems with the traditional definition of a mean, the notion of a **trimmed mean** is sometimes used. A percentage p between 0 and 100 is specified, the top and bottom $(p/2)\%$ of the data is thrown out, and the mean is then calculated in the normal way. The median is a trimmed mean with $p = 100\%$, while the standard mean corresponds to $p = 0\%$.

Example 3.3. Consider the set of values $\{1, 2, 3, 4, 5, 90\}$. The mean of these values is 17.5, while the median is 3.5. The trimmed mean with $p = 40\%$ is also 3.5. ■

Example 3.4. The means, medians, and trimmed means ($p = 20\%$) of the four quantitative attributes of the Iris data are given in Table 3.3. The three measures of location have similar values except for the attribute *petal length*.

Table 3.3. Means and medians for sepal length, sepal width, petal length, and petal width. (All values are in centimeters.)

Measure	Sepal Length	Sepal Width	Petal Length	Petal Width
mean	5.84	3.05	3.76	1.20
median	5.80	3.00	4.35	1.30
trimmed mean (20%)	5.79	3.02	3.72	1.12

■

3.2.4 Measures of Spread: Range and Variance

Another set of commonly used summary statistics for continuous data are those that measure the dispersion or spread of a set of values. Such measures indicate if the attribute values are widely spread out or if they are relatively concentrated around a single point such as the mean.

The simplest measure of spread is the **range**, which, given an attribute x with a set of m values $\{x_1, \ldots, x_m\}$, is defined as

$$\text{range}(x) = \max(x) - \min(x) = x_{(m)} - x_{(1)}. \tag{3.4}$$

Table 3.4. Range, standard deviation (std), absolute average difference (AAD), median absolute difference (MAD), and interquartile range (IQR) for sepal length, sepal width, petal length, and petal width. (All values are in centimeters.)

Measure	Sepal Length	Sepal Width	Petal Length	Petal Width
range	3.6	2.4	5.9	2.4
std	0.8	0.4	1.8	0.8
AAD	0.7	0.3	1.6	0.6
MAD	0.7	0.3	1.2	0.7
IQR	1.3	0.5	3.5	1.5

Although the range identifies the maximum spread, it can be misleading if most of the values are concentrated in a narrow band of values, but there are also a relatively small number of more extreme values. Hence, the **variance** is preferred as a measure of spread. The variance of the (observed) values of an attribute x is typically written as s_x^2 and is defined below. The **standard deviation**, which is the square root of the variance, is written as s_x and has the same units as x.

$$\text{variance}(x) = s_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \overline{x})^2 \tag{3.5}$$

The mean can be distorted by outliers, and since the variance is computed using the mean, it is also sensitive to outliers. Indeed, the variance is particularly sensitive to outliers since it uses the squared difference between the mean and other values. As a result, more robust estimates of the spread of a set of values are often used. Following are the definitions of three such measures: the **absolute average deviation** (AAD), the **median absolute deviation** (MAD), and the **interquartile range** (IQR). Table 3.4 shows these measures for the Iris data set.

$$\text{AAD}(x) = \frac{1}{m}\sum_{i=1}^{m}|x_i - \overline{x}| \tag{3.6}$$

$$\text{MAD}(x) = median\Big(\{|x_1 - \overline{x}|, \ldots, |x_m - \overline{x}|\}\Big) \tag{3.7}$$

$$\text{interquartile range}(x) = x_{75\%} - x_{25\%} \tag{3.8}$$

3.2.5 Multivariate Summary Statistics

Measures of location for data that consists of several attributes (multivariate data) can be obtained by computing the mean or median separately for each attribute. Thus, given a data set the mean of the data objects, $\overline{\mathbf{x}}$, is given by

$$\overline{\mathbf{x}} = (\overline{x_1}, \ldots, \overline{x_n}), \tag{3.9}$$

where $\overline{x_i}$ is the mean of the i^{th} attribute x_i.

For multivariate data, the spread of each attribute can be computed independently of the other attributes using any of the approaches described in Section 3.2.4. However, for data with continuous variables, the spread of the data is most commonly captured by the **covariance matrix S**, whose ij^{th} entry s_{ij} is the covariance of the i^{th} and j^{th} attributes of the data. Thus, if x_i and x_j are the i^{th} and j^{th} attributes, then

$$s_{ij} = \text{covariance}(x_i, x_j). \tag{3.10}$$

In turn, covariance(x_i, x_j) is given by

$$\text{covariance}(x_i, x_j) = \frac{1}{m-1}\sum_{k=1}^{m}(x_{ki} - \overline{x_i})(x_{kj} - \overline{x_j}), \tag{3.11}$$

where x_{ki} and x_{kj} are the values of the i^{th} and j^{th} attributes for the k^{th} object. Notice that covariance(x_i, x_i) = variance(x_i). Thus, the covariance matrix has the variances of the attributes along the diagonal.

The covariance of two attributes is a measure of the degree to which two attributes vary together and depends on the magnitudes of the variables. A value near 0 indicates that two attributes do not have a (linear) relationship, but it is not possible to judge the degree of relationship between two variables by looking only at the value of the covariance. Because the correlation of two attributes immediately gives an indication of how strongly two attributes are (linearly) related, correlation is preferred to covariance for data exploration. (Also see the discussion of correlation in Section 2.4.5.) The ij^{th} entry of the **correlation matrix R**, is the correlation between the i^{th} and j^{th} attributes of the data. If x_i and x_j are the i^{th} and j^{th} attributes, then

$$r_{ij} = \text{correlation}(x_i, x_j) = \frac{\text{covariance}(x_i, x_j)}{s_i s_j}, \tag{3.12}$$

where s_i and s_j are the variances of x_i and x_j, respectively. The diagonal entries of $\mathbf{R}$ are correlation$(x_i, x_i) = 1$, while the other entries are between -1 and 1. It is also useful to consider correlation matrices that contain the pairwise correlations of objects instead of attributes.

3.2.6 Other Ways to Summarize the Data

There are, of course, other types of summary statistics. For instance, the **skewness** of a set of values measures the degree to which the values are symmetrically distributed around the mean. There are also other characteristics of the data that are not easy to measure quantitatively, such as whether the distribution of values is multimodal; i.e., the data has multiple "bumps" where most of the values are concentrated. In many cases, however, the most effective approach to understanding the more complicated or subtle aspects of how the values of an attribute are distributed, is to view the values graphically in the form of a histogram. (Histograms are discussed in the next section.)

3.3 Visualization

Data visualization is the display of information in a graphic or tabular format. Successful visualization requires that the data (information) be converted into a visual format so that the characteristics of the data and the relationships among data items or attributes can be analyzed or reported. The goal of visualization is the interpretation of the visualized information by a person and the formation of a mental model of the information.

In everyday life, visual techniques such as graphs and tables are often the preferred approach used to explain the weather, the economy, and the results of political elections. Likewise, while algorithmic or mathematical approaches are often emphasized in most technical disciplines—data mining included—visual techniques can play a key role in data analysis. In fact, sometimes the use of visualization techniques in data mining is referred to as **visual data mining**.

3.3.1 Motivations for Visualization

The overriding motivation for using visualization is that people can quickly absorb large amounts of visual information and find patterns in it. Consider Figure 3.2, which shows the Sea Surface Temperature (SST) in degrees Celsius for July, 1982. This picture summarizes the information from approximately 250,000 numbers and is readily interpreted in a few seconds. For example, it

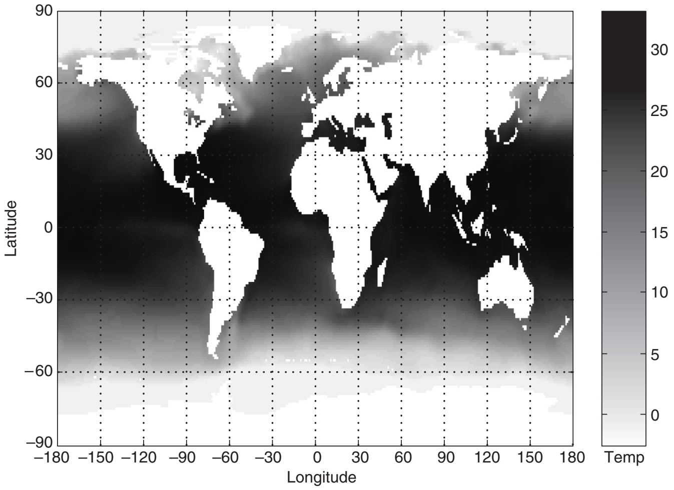


Figure 3.2. Sea Surface Temperature (SST) for July, 1982.

is easy to see that the ocean temperature is highest at the equator and lowest at the poles.

Another general motivation for visualization is to make use of the domain knowledge that is "locked up in people's heads." While the use of domain knowledge is an important task in data mining, it is often difficult or impossible to fully utilize such knowledge in statistical or algorithmic tools. In some cases, an analysis can be performed using non-visual tools, and then the results presented visually for evaluation by the domain expert. In other cases, having a domain specialist examine visualizations of the data may be the best way of finding patterns of interest since, by using domain knowledge, a person can often quickly eliminate many uninteresting patterns and direct the focus to the patterns that are important.

3.3.2 General Concepts

This section explores some of the general concepts related to visualization, in particular, general approaches for visualizing the data and its attributes. A number of visualization techniques are mentioned briefly and will be described in more detail when we discuss specific approaches later on. We assume that the reader is familiar with line graphs, bar charts, and scatter plots.

Representation: Mapping Data to Graphical Elements

The first step in visualization is the mapping of information to a visual format; i.e., mapping the objects, attributes, and relationships in a set of information to visual objects, attributes, and relationships. That is, data objects, their attributes, and the relationships among data objects are translated into graphical elements such as points, lines, shapes, and colors.

Objects are usually represented in one of three ways. First, if only a single categorical attribute of the object is being considered, then objects are often lumped into categories based on the value of that attribute, and these categories are displayed as an entry in a table or an area on a screen. (Examples shown later in this chapter are a cross-tabulation table and a bar chart.) Second, if an object has multiple attributes, then the object can be displayed as a row (or column) of a table or as a line on a graph. Finally, an object is often interpreted as a point in two- or three-dimensional space, where graphically, the point might be represented by a geometric figure, such as a circle, cross, or box.

For attributes, the representation depends on the type of attribute, i.e., nominal, ordinal, or continuous (interval or ratio). Ordinal and continuous attributes can be mapped to continuous, ordered graphical features such as location along the x, y, or z axes; intensity; color; or size (diameter, width, height, etc.). For categorical attributes, each category can be mapped to a distinct position, color, shape, orientation, embellishment, or column in a table. However, for nominal attributes, whose values are unordered, care should be taken when using graphical features, such as color and position that have an inherent ordering associated with their values. In other words, the graphical elements used to represent the ordinal values often have an order, but ordinal values do not.

The representation of relationships via graphical elements occurs either explicitly or implicitly. For graph data, the standard graph representation—a set of nodes with links between the nodes—is normally used. If the nodes (data objects) or links (relationships) have attributes or characteristics of their own, then this is represented graphically. To illustrate, if the nodes are cities and the links are highways, then the diameter of the nodes might represent population, while the width of the links might represent the volume of traffic.

In most cases, though, mapping objects and attributes to graphical elements implicitly maps the relationships in the data to relationships among graphical elements. To illustrate, if the data object represents a physical object that has a location, such as a city, then the relative positions of the graphical objects corresponding to the data objects tend to naturally preserve the actual relative positions of the objects. Likewise, if there are two or three continuous attributes that are taken as the coordinates of the data points, then the resulting plot often gives considerable insight into the relationships of the attributes and the data points because data points that are visually close to each other have similar values for their attributes.

In general, it is difficult to ensure that a mapping of objects and attributes will result in the relationships being mapped to easily observed relationships among graphical elements. Indeed, this is one of the most challenging aspects of visualization. In any given set of data, there are many implicit relationships, and hence, a key challenge of visualization is to choose a technique that makes the relationships of interest easily observable.

Arrangement

As discussed earlier, the proper choice of visual representation of objects and attributes is essential for good visualization. The arrangement of items within the visual display is also crucial. We illustrate this with two examples.

Example 3.5. This example illustrates the importance of rearranging a table of data. In Table 3.5, which shows nine objects with six binary attributes, there is no clear relationship between objects and attributes, at least at first glance. If the rows and columns of this table are permuted, however, as shown in Table 3.6, then it is clear that there are really only two types of objects in the table—one that has all ones for the first three attributes and one that has only ones for the last three attributes. ■

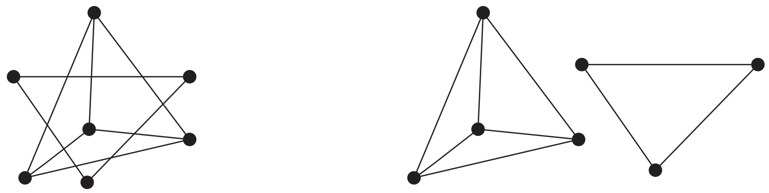
Table 3.5. A table of nine objects (rows) with six binary attributes (columns).

	1	2	3	4	5	6
1	0	1	0	1	1	0
2	1	0	1	0	0	1
3	0	1	0	1	1	0
4	1	0	1	0	0	1
5	0	1	0	1	1	0
6	1	0	1	0	0	1
7	0	1	0	1	1	0
8	1	0	1	0	0	1
9	0	1	0	1	1	0

Table 3.6. A table of nine objects (rows) with six binary attributes (columns) permuted so that the relationships of the rows and columns are clear.

	6	1	3	2	5	4
4	1	1	1	0	0	0
2	1	1	1	0	0	0
6	1	1	1	0	0	0
8	1	1	1	0	0	0
5	0	0	0	1	1	1
3	0	0	0	1	1	1
9	0	0	0	1	1	1
1	0	0	0	1	1	1
7	0	0	0	1	1	1

Example 3.6. Consider Figure 3.3(a), which shows a visualization of a graph. If the connected components of the graph are separated, as in Figure 3.3(b), then the relationships between nodes and graphs become much simpler to understand. ■



(a) Original view of a graph.

(b) Uncoupled view of connected components of the graph.

Figure 3.3. Two visualizations of a graph.

Selection

Another key concept in visualization is **selection**, which is the elimination or the de-emphasis of certain objects and attributes. Specifically, while data objects that only have a few dimensions can often be mapped to a two- or three-dimensional graphical representation in a straightforward way, there is no completely satisfactory and general approach to represent data with many attributes. Likewise, if there are many data objects, then visualizing all the objects can result in a display that is too crowded. If there are many attributes and many objects, then the situation is even more challenging.

The most common approach to handling many attributes is to choose a subset of attributes—usually two—for display. If the dimensionality is not too high, a matrix of bivariate (two-attribute) plots can be constructed for simultaneous viewing. (Figure 3.16 shows a matrix of scatter plots for the pairs of attributes of the Iris data set.) Alternatively, a visualization program can automatically show a series of two-dimensional plots, in which the sequence is user directed or based on some predefined strategy. The hope is that visualizing a collection of two-dimensional plots will provide a more complete view of the data.

The technique of selecting a pair (or small number) of attributes is a type of dimensionality reduction, and there are many more sophisticated dimensionality reduction techniques that can be employed, e.g., principal components analysis (PCA). Consult Appendices A (Linear Algebra) and B (Dimensionality Reduction) for more information.

When the number of data points is high, e.g., more than a few hundred, or if the range of the data is large, it is difficult to display enough information about each object. Some data points can obscure other data points, or a data object may not occupy enough pixels to allow its features to be clearly displayed. For example, the shape of an object cannot be used to encode a characteristic of that object if there is only one pixel available to display it. In these situations, it is useful to be able to eliminate some of the objects, either by zooming in on a particular region of the data or by taking a sample of the data points.

3.3.3 Techniques

Visualization techniques are often specialized to the type of data being analyzed. Indeed, new visualization techniques and approaches, as well as specialized variations of existing approaches, are being continuously created, typically in response to new kinds of data and visualization tasks.

Despite this specialization and the ad hoc nature of visualization, there are some generic ways to classify visualization techniques. One such classification is based on the number of attributes involved (1, 2, 3, or many) or whether the data has some special characteristic, such as a hierarchical or graph structure. Visualization methods can also be classified according to the type of attributes involved. Yet another classification is based on the type of application: scientific, statistical, or information visualization. The following discussion will use three categories: visualization of a small number of attributes, visualization of data with spatial and/or temporal attributes, and visualization of data with many attributes.

Most of the visualization techniques discussed here can be found in a wide variety of mathematical and statistical packages, some of which are freely available. There are also a number of data sets that are freely available on the World Wide Web. Readers are encouraged to try these visualization techniques as they proceed through the following sections.

Visualizing Small Numbers of Attributes

This section examines techniques for visualizing data with respect to a small number of attributes. Some of these techniques, such as histograms, give insight into the distribution of the observed values for a single attribute. Other techniques, such as scatter plots, are intended to display the relationships between the values of two attributes.

Stem and Leaf Plots Stem and leaf plots can be used to provide insight into the distribution of one-dimensional integer or continuous data. (We will assume integer data initially, and then explain how stem and leaf plots can be applied to continuous data.) For the simplest type of stem and leaf plot, we split the values into groups, where each group contains those values that are the same except for the last digit. Each group becomes a stem, while the last digits of a group are the leaves. Hence, if the values are two-digit integers, e.g., 35, 36, 42, and 51, then the stems will be the high-order digits, e.g., 3, 4, and 5, while the leaves are the low-order digits, e.g., 1, 2, 5, and 6. By plotting the stems vertically and leaves horizontally, we can provide a visual representation of the distribution of the data.

Example 3.7. The set of integers shown in Figure 3.4 is the sepal length in centimeters (multiplied by 10 to make the values integers) taken from the Iris data set. For convenience, the values have also been sorted.

The stem and leaf plot for this data is shown in Figure 3.5. Each number in Figure 3.4 is first put into one of the vertical groups—4, 5, 6, or 7—according to its ten's digit. Its last digit is then placed to the right of the colon. Often, especially if the amount of data is larger, it is desirable to split the stems. For example, instead of placing all values whose ten's digit is 4 in the same "bucket," the stem 4 is repeated twice; all values 40–44 are put in the bucket corresponding to the first stem and all values 45–49 are put in the bucket corresponding to the second stem. This approach is shown in the stem and leaf plot of Figure 3.6. Other variations are also possible. ■

Histograms Stem and leaf plots are a type of **histogram**, a plot that displays the distribution of values for attributes by dividing the possible values into bins and showing the number of objects that fall into each bin. For categorical data, each value is a bin. If this results in too many values, then values are combined in some way. For continuous attributes, the range of values is divided into bins—typically, but not necessarily, of equal width—and the values in each bin are counted.

Once the counts are available for each bin, a **bar plot** is constructed such that each bin is represented by one bar and the area of each bar is proportional to the number of values (objects) that fall into the corresponding range. If all intervals are of equal width, then all bars are the same width and the height of a bar is proportional to the number of values in the corresponding bin.

```
43 44 44 44 45 46 46 46 46 47 47 48 48 48 48 48 49 49 49 49 49 49 50
50 50 50 50 50 50 50 50 50 51 51 51 51 51 51 51 51 51 52 52 52 52 53
54 54 54 54 54 54 55 55 55 55 55 55 55 56 56 56 56 56 56 57 57 57 57
57 57 57 57 58 58 58 58 58 58 58 59 59 59 60 60 60 60 60 60 61 61 61
61 61 61 62 62 62 62 63 63 63 63 63 63 63 63 63 64 64 64 64 64 64 64
65 65 65 65 65 66 66 67 67 67 67 67 67 67 67 68 68 68 69 69 69 69 70
71 72 72 72 73 74 76 77 77 77 77 79
```

Figure 3.4. Sepal length data from the Iris data set.

```
4 : 34444566667788888999999
5 : 0000000000111111111222234444445555555666666777777778888888999
6 : 000000111111222233333333344444445555566777777778889999
7 : 0122234677779
```

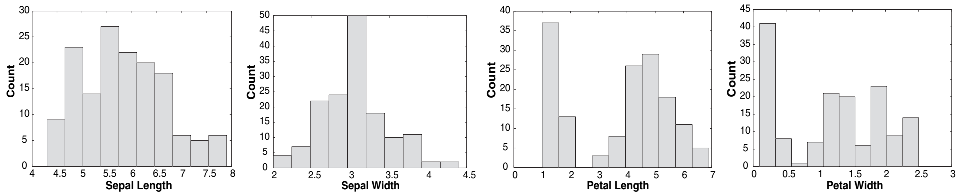
Figure 3.5. Stem and leaf plot for the sepal length from the Iris data set.

```
4 : 3444
4 : 566667788888999999
5 : 000000000011111111122223444444
5 : 5555555666666777777778888888999
6 : 00000011111122223333333334444444
6 : 5555566777777778889999
7 : 0122234
7 : 677779
```

Figure 3.6. Stem and leaf plot for the sepal length from the Iris data set when buckets corresponding to digits are split.

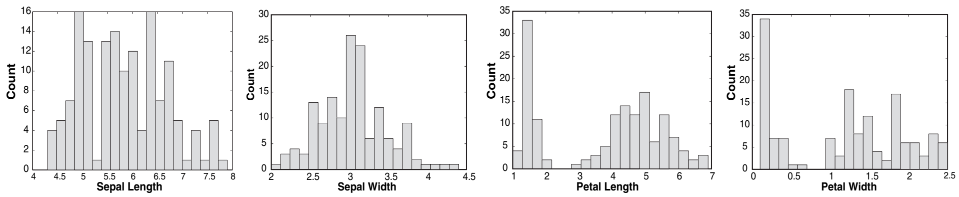
Example 3.8. Figure 3.7 shows histograms (with 10 bins) for sepal length, sepal width, petal length, and petal width. Since the shape of a histogram can depend on the number of bins, histograms for the same data, but with 20 bins, are shown in Figure 3.8. ■

There are variations of the histogram plot. A **relative (frequency) histogram** replaces the count by the relative frequency. However, this is just a change in scale of the y axis, and the shape of the histogram does not change. Another common variation, especially for unordered categorical data, is the **Pareto histogram**, which is the same as a normal histogram except that the categories are sorted by count so that the count is decreasing from left to right.



(a) Sepal length. (b) Sepal width. (c) Petal length. (d) Petal width.

Figure 3.7. Histograms of four Iris attributes (10 bins).



(a) Sepal length. (b) Sepal width. (c) Petal length. (d) Petal width.

Figure 3.8. Histograms of four Iris attributes (20 bins).

Two-Dimensional Histograms Two-dimensional histograms are also possible. Each attribute is divided into intervals and the two sets of intervals define two-dimensional rectangles of values.

Example 3.9. Figure 3.9 shows a two-dimensional histogram of petal length and petal width. Because each attribute is split into three bins, there are nine rectangular two-dimensional bins. The height of each rectangular bar indicates the number of objects (flowers in this case) that fall into each bin. Most of the flowers fall into only three of the bins—those along the diagonal. It is not possible to see this by looking at the one-dimensional distributions. ■

While two-dimensional histograms can be used to discover interesting facts about how the values of two attributes co-occur, they are visually more complicated. For instance, it is easy to imagine a situation in which some of the columns are hidden by others.

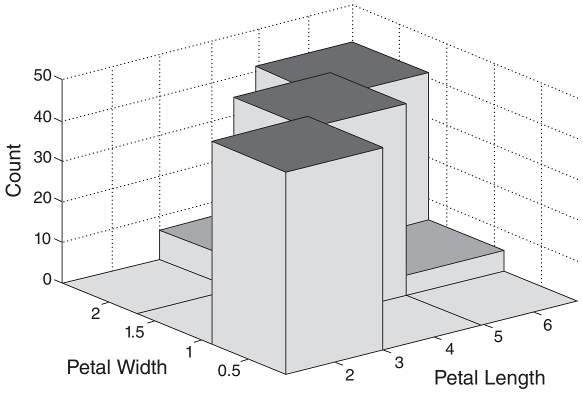


Figure 3.9. Two-dimensional histogram of petal length and width in the Iris data set.

Box Plots Box plots are another method for showing the distribution of the values of a single numerical attribute. Figure 3.10 shows a labeled box plot for sepal length. The lower and upper ends of the box indicate the 25^{th} and 75^{th} percentiles, respectively, while the line inside the box indicates the value of the 50^{th} percentile. The top and bottom lines of the **tails** indicate the 10^{th} and 90^{th} percentiles. Outliers are shown by "+" marks. Box plots are relatively compact, and thus, many of them can be shown on the same plot. Simplified versions of the box plot, which take less space, can also be used.

Example 3.10. The box plots for the first four attributes of the Iris data set are shown in Figure 3.11. Box plots can also be used to compare how attributes vary between different classes of objects, as shown in Figure 3.12. ■

Pie Chart A pie chart is similar to a histogram, but is typically used with categorical attributes that have a relatively small number of values. Instead of showing the relative frequency of different values with the area or height of a bar, as in a histogram, a pie chart uses the relative area of a circle to indicate relative frequency. Although pie charts are common in popular articles, they are used less frequently in technical publications because the size of relative areas can be hard to judge. Histograms are preferred for technical work.

Example 3.11. Figure 3.13 displays a pie chart that shows the distribution of Iris species in the Iris data set. In this case, all three flower types have the same frequency. ■

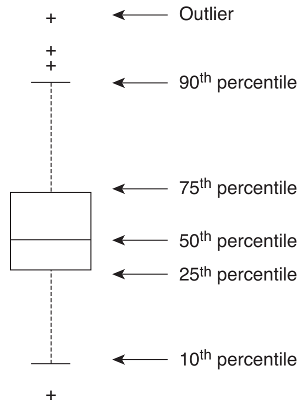


Figure 3.10. Description of box plot for sepal length.

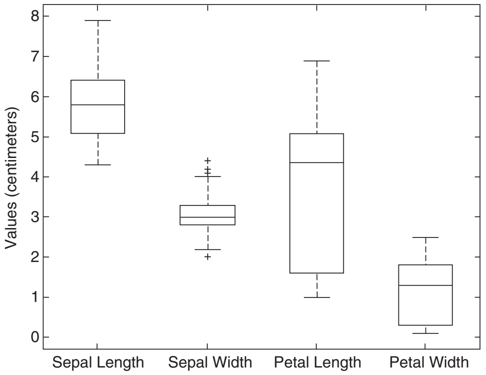
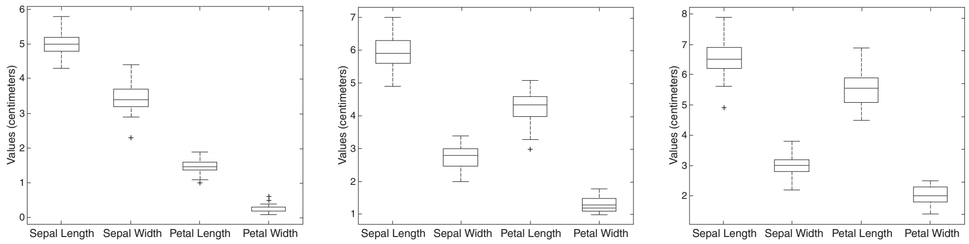


Figure 3.11. Box plot for Iris attributes.



(a) Setosa. (b) Versicolour. (c) Virginica.

Figure 3.12. Box plots of attributes by Iris species.

Percentile Plots and Empirical Cumulative Distribution Functions A type of diagram that shows the distribution of the data more quantitatively is the plot of an empirical cumulative distribution function. While this type of plot may sound complicated, the concept is straightforward. For each value of a statistical distribution, a **cumulative distribution function** (CDF) shows the probability that a point is less than that value. For each observed value, an **empirical cumulative distribution function** (ECDF) shows the fraction of points that are less than this value. Since the number of points is finite, the empirical cumulative distribution function is a step function.

Example 3.12. Figure 3.14 shows the ECDFs of the Iris attributes. The percentiles of an attribute provide similar information. Figure 3.15 shows the **percentile plots** of the four continuous attributes of the Iris data set from

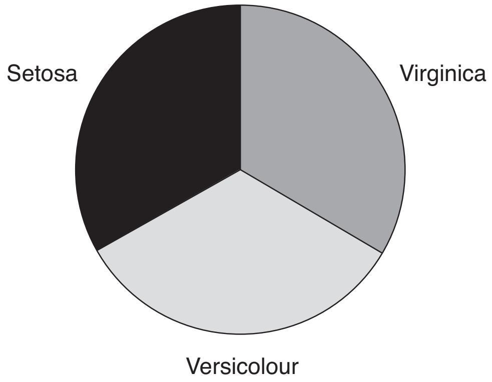
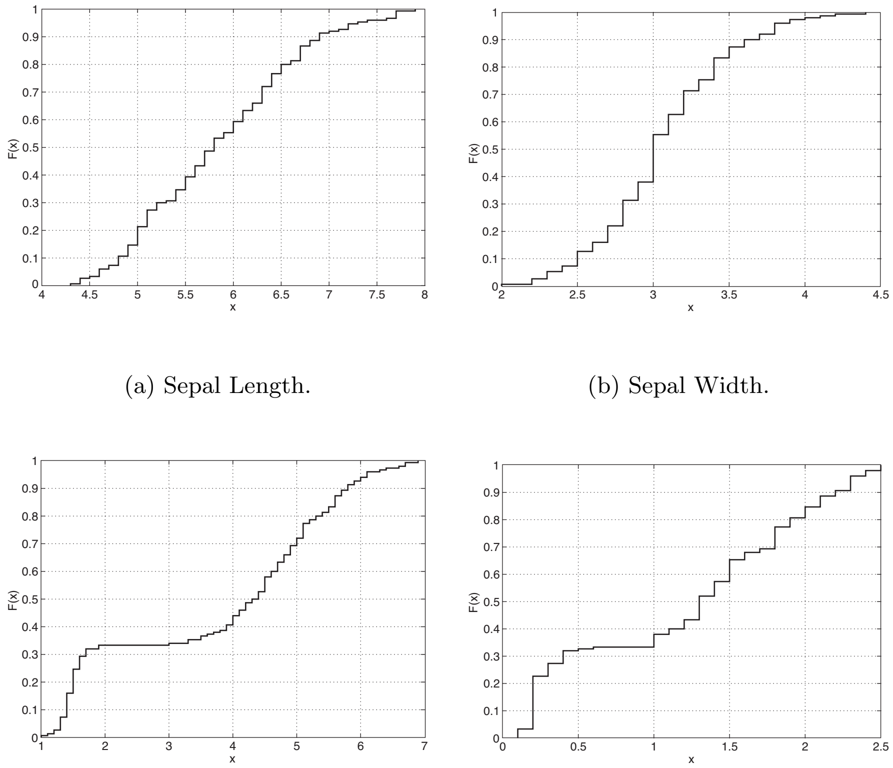


Figure 3.13. Distribution of the types of Iris flowers.

Table 3.2. The reader should compare these figures with the histograms given in Figures 3.7 and 3.8. ■

Scatter Plots Most people are familiar with scatter plots to some extent, and they were used in Section 2.4.5 to illustrate linear correlation. Each data object is plotted as a point in the plane using the values of the two attributes as x and y coordinates. It is assumed that the attributes are either integer- or real-valued.

Example 3.13. Figure 3.16 shows a scatter plot for each pair of attributes of the Iris data set. The different species of Iris are indicated by different markers. The arrangement of the scatter plots of pairs of attributes in this type of tabular format, which is known as a **scatter plot matrix**, provides an organized way to examine a number of scatter plots simultaneously. ■



(a) Sepal Length.

(b) Sepal Width.

(c) Petal Length.

(d) Petal Width.

Figure 3.14. Empirical CDFs of four Iris attributes.

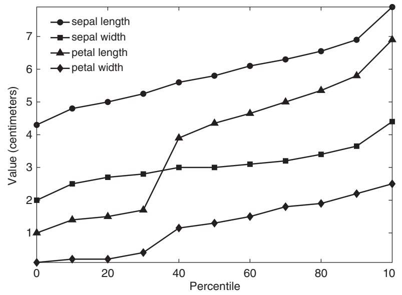


Figure 3.15. Percentile plots for sepal length, sepal width, petal length, and petal width.

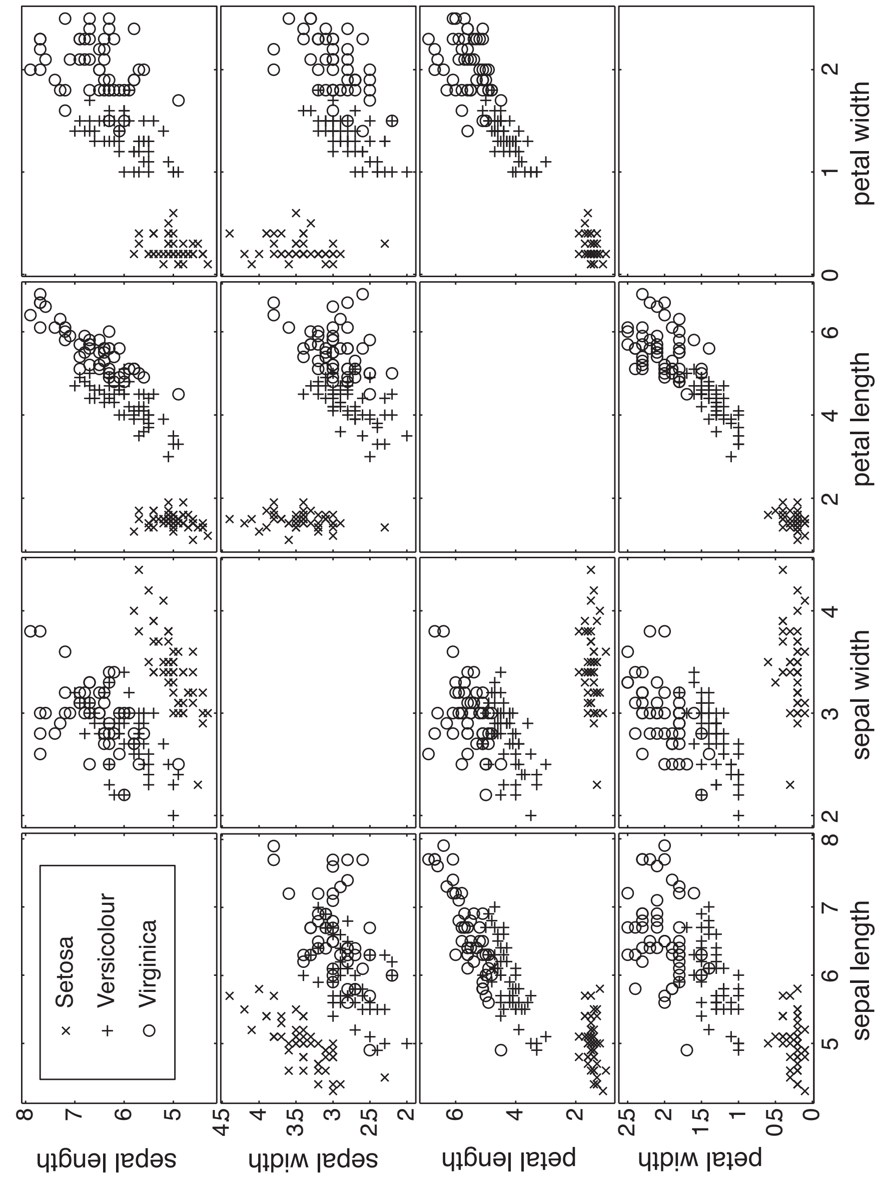


Figure 3.16. Matrix of scatter plots for the Iris data set.

There are two main uses for scatter plots. First, they graphically show the relationship between two attributes. In Section 2.4.5, we saw how scatter plots could be used to judge the degree of linear correlation. (See Figure 2.17.) Scatter plots can also be used to detect non-linear relationships, either directly or by using a scatter plot of the transformed attributes.

Second, when class labels are available, they can be used to investigate the degree to which two attributes separate the classes. If is possible to draw a line (or a more complicated curve) that divides the plane defined by the two attributes into separate regions that contain mostly objects of one class, then it is possible to construct an accurate classifier based on the specified pair of attributes. If not, then more attributes or more sophisticated methods are needed to build a classifier. In Figure 3.16, many of the pairs of attributes (for example, petal width and petal length) provide a moderate separation of the Iris species.

Example 3.14. There are two separate approaches for displaying three attributes of a data set with a scatter plot. First, each object can be displayed according to the values of three, instead of two attributes. Figure 3.17 shows a three-dimensional scatter plot for three attributes in the Iris data set. Second, one of the attributes can be associated with some characteristic of the marker, such as its size, color, or shape. Figure 3.18 shows a plot of three attributes of the Iris data set, where one of the attributes, sepal width, is mapped to the size of the marker. ■

Extending Two- and Three-Dimensional Plots As illustrated by Figure 3.18, two- or three-dimensional plots can be extended to represent a few additional attributes. For example, scatter plots can display up to three additional attributes using color or shading, size, and shape, allowing five or six dimensions to be represented. There is a need for caution, however. As the complexity of a visual representation of the data increases, it becomes harder for the intended audience to interpret the information. There is no benefit in packing six dimensions' worth of information into a two- or three-dimensional plot, if doing so makes it impossible to understand.

Visualizing Spatio-temporal Data

Data often has spatial or temporal attributes. For instance, the data may consist of a set of observations on a spatial grid, such as observations of pressure on the surface of the Earth or the modeled temperature at various grid points in the simulation of a physical object. These observations can also

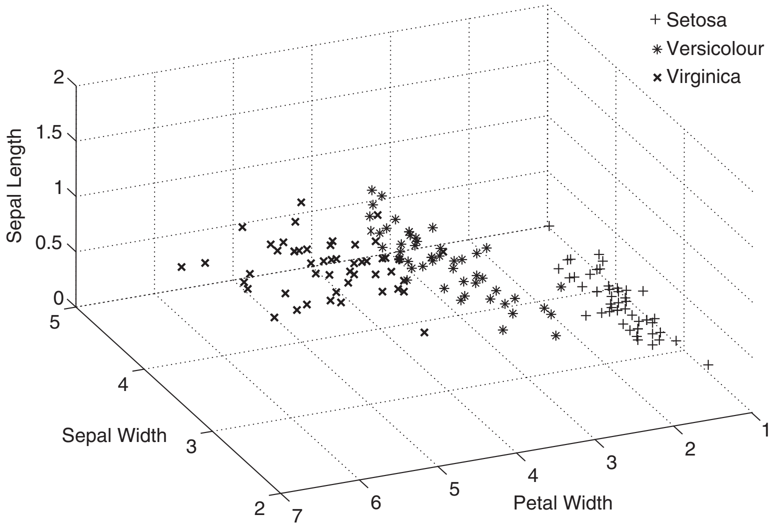


Figure 3.17. Three-dimensional scatter plot of sepal width, sepal length, and petal width.

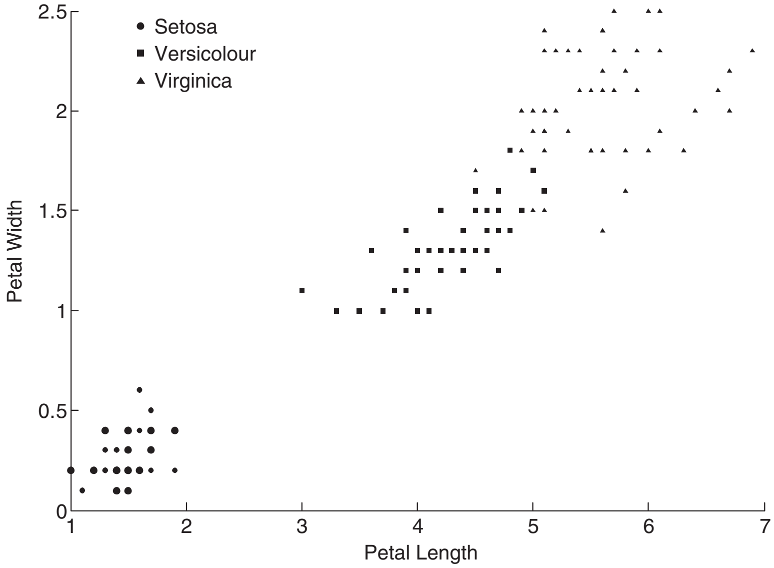


Figure 3.18. Scatter plot of petal length versus petal width, with the size of the marker indicating sepal width.

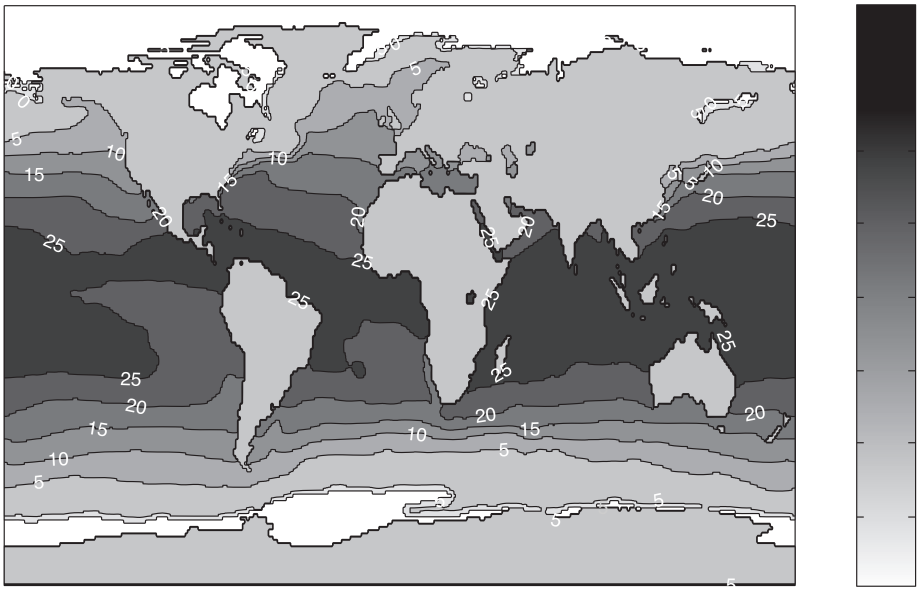


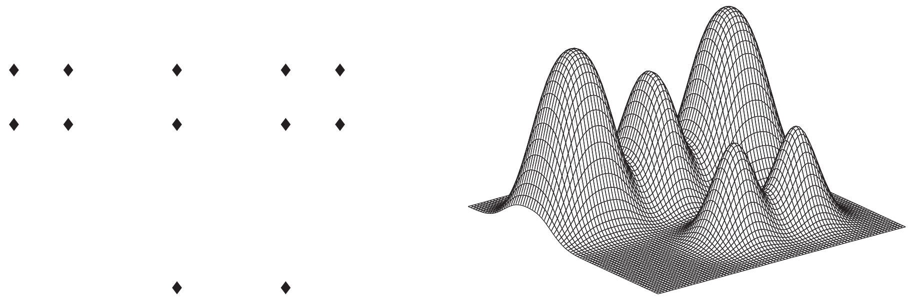
Figure 3.19. Contour plot of SST for December 1998.

be made at various points in time. In addition, data may have only a temporal component, such as time series data that gives the daily prices of stocks.

Contour Plots For some three-dimensional data, two attributes specify a position in a plane, while the third has a continuous value, such as temperature or elevation. A useful visualization for such data is a **contour plot**, which breaks the plane into separate regions where the values of the third attribute (temperature, elevation) are roughly the same. A common example of a contour plot is a contour map that shows the elevation of land locations.

Example 3.15. Figure 3.19 shows a contour plot of the average sea surface temperature (SST) for December 1998. The land is arbitrarily set to have a temperature of 0°C. In many contour maps, such as that of Figure 3.19, the **contour lines** that separate two regions are labeled with the value used to separate the regions. For clarity, some of these labels have been deleted. ■

Surface Plots Like contour plots, **surface plots** use two attributes for the x and y coordinates. The third attribute is used to indicate the height above



(a) Set of 12 points.

(b) Overall density function—surface plot.

Figure 3.20. Density of a set of 12 points.

the plane defined by the first two attributes. While such graphs can be useful, they require that a value of the third attribute be defined for all combinations of values for the first two attributes, at least over some range. Also, if the surface is too irregular, then it can be difficult to see all the information, unless the plot is viewed interactively. Thus, surface plots are often used to describe mathematical functions or physical surfaces that vary in a relatively smooth manner.

Example 3.16. Figure 3.20 shows a surface plot of the density around a set of 12 points. This example is further discussed in Section 9.3.3. ■

Vector Field Plots In some data, a characteristic may have both a magnitude and a direction associated with it. For example, consider the flow of a substance or the change of density with location. In these situations, it can be useful to have a plot that displays both direction and magnitude. This type of plot is known as a **vector plot**.

Example 3.17. Figure 3.21 shows a contour plot of the density of the two smaller density peaks from Figure 3.20(b), annotated with the density gradient vectors. ■

Lower-Dimensional Slices Consider a spatio-temporal data set that records some quantity, such as temperature or pressure, at various locations over time. Such a data set has four dimensions and cannot be easily displayed by the types of plots that we have described so far. However, separate “slices” of the data

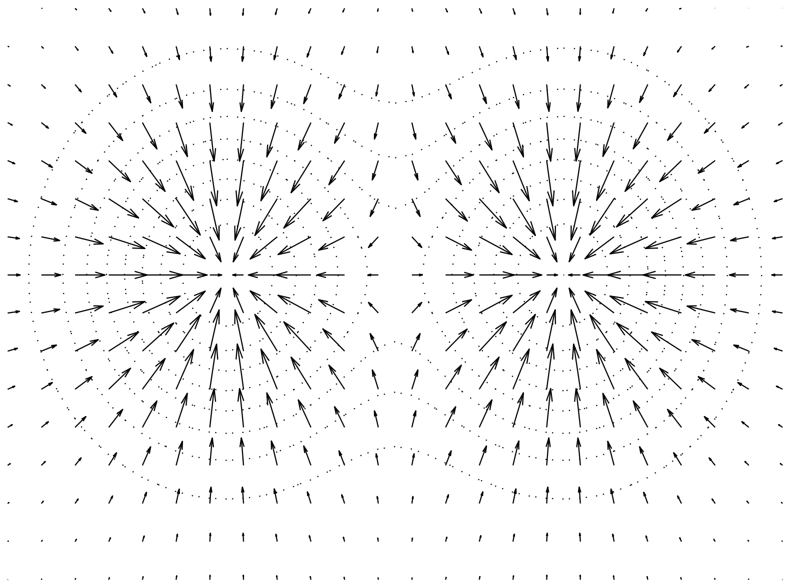


Figure 3.21. Vector plot of the gradient (change) in density for the bottom two density peaks of Figure 3.20.

can be displayed by showing a set of plots, one for each month. By examining the change in a particular area from one month to another, it is possible to notice changes that occur, including those that may be due to seasonal factors.

Example 3.18. The underlying data set for this example consists of the average monthly sea level pressure (SLP) from 1982 to 1999 on a 2.5° by 2.5° latitude-longitude grid. The twelve monthly plots of pressure for one year are shown in Figure 3.22. In this example, we are interested in slices for a particular month in the year 1982. More generally, we can consider slices of the data along any arbitrary dimension. ■

Animation Another approach to dealing with slices of data, whether or not time is involved, is to employ animation. The idea is to display successive two-dimensional slices of the data. The human visual system is well suited to detecting visual changes and can often notice changes that might be difficult to detect in another manner. Despite the visual appeal of animation, a set of still plots, such as those of Figure 3.22, can be more useful since this type of visualization allows the information to be studied in arbitrary order and for arbitrary amounts of time.

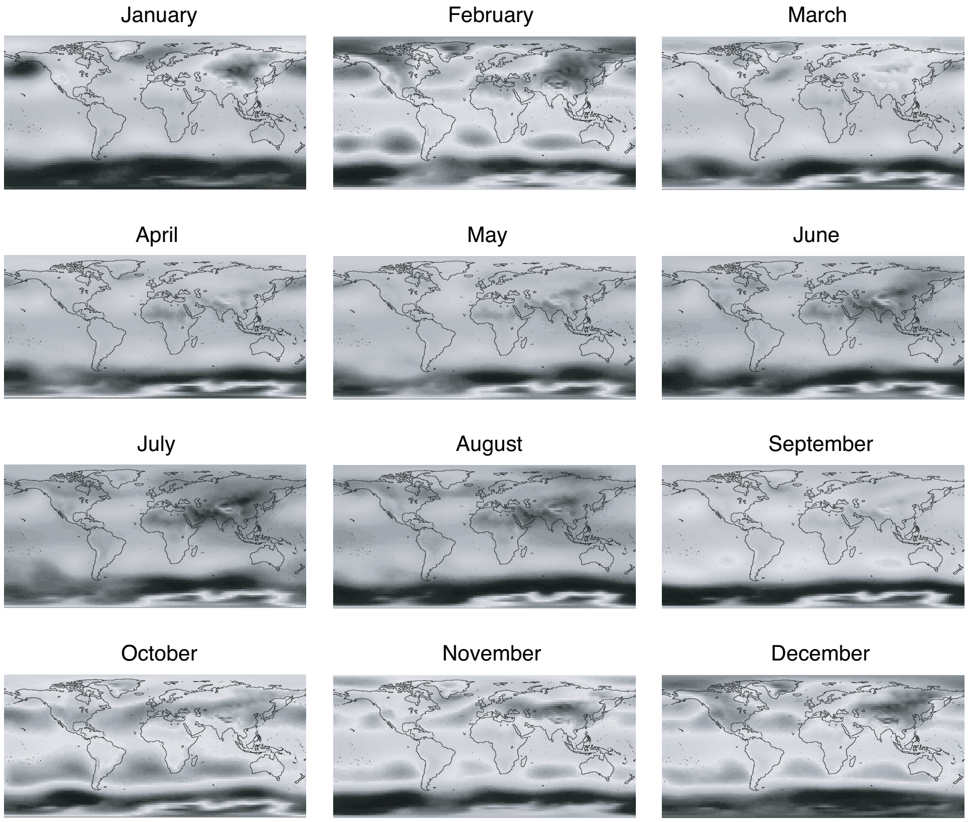


Figure 3.22. Monthly plots of sea level pressure over the 12 months of 1982.

3.3.4 Visualizing Higher-Dimensional Data

This section considers visualization techniques that can display more than the handful of dimensions that can be observed with the techniques just discussed. However, even these techniques are somewhat limited in that they only show some aspects of the data.

Matrices An image can be regarded as a rectangular array of pixels, where each pixel is characterized by its color and brightness. A data matrix is a rectangular array of values. Thus, a data matrix can be visualized as an image by associating each entry of the data matrix with a pixel in the image. The brightness or color of the pixel is determined by the value of the corresponding entry of the matrix.

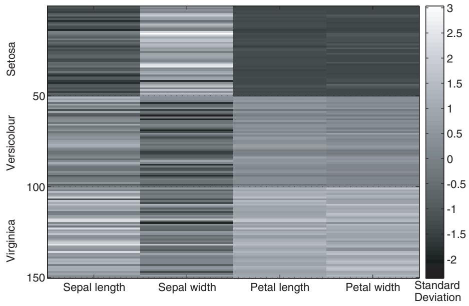


Figure 3.23. Plot of the Iris data matrix where columns have been standardized to have a mean of 0 and standard deviation of 1.

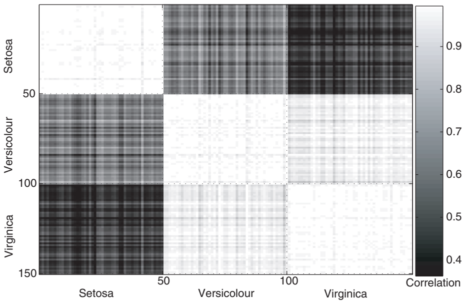


Figure 3.24. Plot of the Iris correlation matrix.

There are some important practical considerations when visualizing a data matrix. If class labels are known, then it is useful to reorder the data matrix so that all objects of a class are together. This makes it easier, for example, to detect if all objects in a class have similar attribute values for some attributes. If different attributes have different ranges, then the attributes are often standardized to have a mean of zero and a standard deviation of 1. This prevents the attribute with the largest magnitude values from visually dominating the plot.

Example 3.19. Figure 3.23 shows the standardized data matrix for the Iris data set. The first 50 rows represent Iris flowers of the species Setosa, the next 50 Versicolour, and the last 50 Virginica. The Setosa flowers have petal width and length well below the average, while the Versicolour flowers have petal width and length around average. The Virginica flowers have petal width and length above average. ■

It can also be useful to look for structure in the plot of a proximity matrix for a set of data objects. Again, it is useful to sort the rows and columns of the similarity matrix (when class labels are known) so that all the objects of a class are together. This allows a visual evaluation of the cohesiveness of each class and its separation from other classes.

Example 3.20. Figure 3.24 shows the correlation matrix for the Iris data set. Again, the rows and columns are organized so that all the flowers of a particular species are together. The flowers in each group are most similar to each other, but Versicolour and Virginica are more similar to one another than to Setosa. ■

If class labels are not known, various techniques (matrix reordering and seriation) can be used to rearrange the rows and columns of the similarity matrix so that groups of highly similar objects and attributes are together and can be visually identified. Effectively, this is a simple kind of clustering. See Section 8.5.3 for a discussion of how a proximity matrix can be used to investigate the cluster structure of data.

Parallel Coordinates Parallel coordinates have one coordinate axis for each attribute, but the different axes are parallel to one other instead of perpendicular, as is traditional. Furthermore, an object is represented as a line instead of as a point. Specifically, the value of each attribute of an object is mapped to a point on the coordinate axis associated with that attribute, and these points are then connected to form the line that represents the object.

It might be feared that this would yield quite a mess. However, in many cases, objects tend to fall into a small number of groups, where the points in each group have similar values for their attributes. If so, and if the number of data objects is not too large, then the resulting parallel coordinates plot can reveal interesting patterns.

Example 3.21. Figure 3.25 shows a parallel coordinates plot of the four numerical attributes of the Iris data set. The lines representing objects of different classes are distinguished by their shading and the use of three different line styles—solid, dotted, and dashed. The parallel coordinates plot shows that the classes are reasonably well separated for petal width and petal length, but less well separated for sepal length and sepal width. Figure 3.26 is another parallel coordinates plot of the same data, but with a different ordering of the axes. ■

One of the drawbacks of parallel coordinates is that the detection of patterns in such a plot may depend on the order. For instance, if lines cross a lot, the picture can become confusing, and thus, it can be desirable to order the coordinate axes to obtain sequences of axes with less crossover. Compare Figure 3.26, where sepal width (the attribute that is most mixed) is at the left of the figure, to Figure 3.25, where this attribute is in the middle.

Star Coordinates and Chernoff Faces

Another approach to displaying multidimensional data is to encode objects as **glyphs** or **icons**—symbols that impart information non-verbally. More

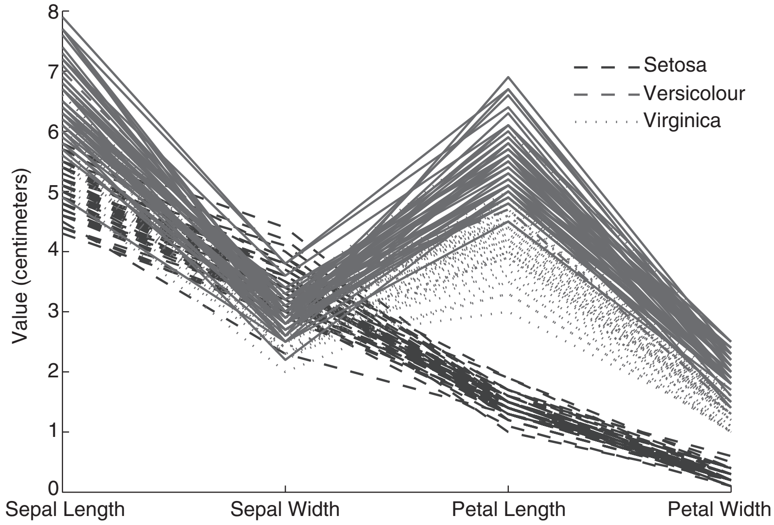


Figure 3.25. A parallel coordinates plot of the four Iris attributes.

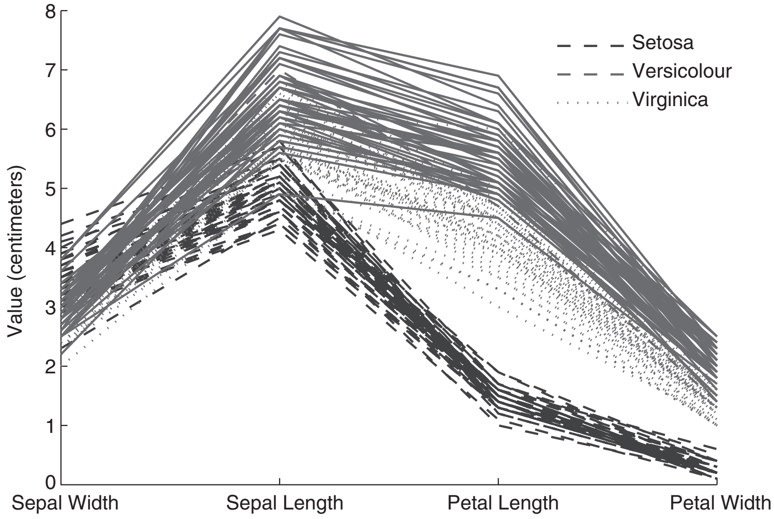


Figure 3.26. A parallel coordinates plot of the four Iris attributes with the attributes reordered to emphasize similarities and dissimilarities of groups.

specifically, each attribute of an object is mapped to a particular feature of a glyph, so that the value of the attribute determines the exact nature of the feature. Thus, at a glance, we can distinguish how two objects differ.

Star coordinates are one example of this approach. This technique uses one axis for each attribute. These axes all radiate from a center point, like the spokes of a wheel, and are evenly spaced. Typically, all the attribute values are mapped to the range [0,1].

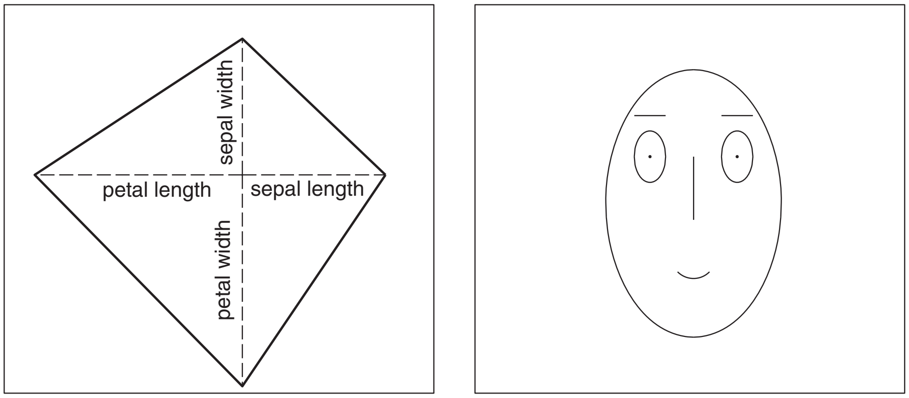
An object is mapped onto this star-shaped set of axes using the following process: Each attribute value of the object is converted to a fraction that represents its distance between the minimum and maximum values of the attribute. This fraction is mapped to a point on the axis corresponding to this attribute. Each point is connected with a line segment to the point on the axis preceding or following its own axis; this forms a polygon. The size and shape of this polygon gives a visual description of the attribute values of the object. For ease of interpretation, a separate set of axes is used for each object. In other words, each object is mapped to a polygon. An example of a star coordinates plot of flower 150 is given in Figure 3.27(a).

It is also possible to map the values of features to those of more familiar objects, such as faces. This technique is named **Chernoff faces** for its creator, Herman Chernoff. In this technique, each attribute is associated with a specific feature of a face, and the attribute value is used to determine the way that the facial feature is expressed. Thus, the shape of the face may become more elongated as the value of the corresponding data feature increases. An example of a Chernoff face for flower 150 is given in Figure 3.27(b).

The program that we used to make this face mapped the features to the four features listed below. Other features of the face, such as width between the eyes and length of the mouth, are given default values.

Data Feature	Facial Feature
sepal length	size of face
sepal width	forehead/jaw relative arc length
petal length	shape of forehead
petal width	shape of jaw

Example 3.22. A more extensive illustration of these two approaches to viewing multidimensional data is provided by Figures 3.28 and 3.29, which shows the star and face plots, respectively, of 15 flowers from the Iris data set. The first 5 flowers are of species Setosa, the second 5 are Versicolour, and the last 5 are Virginica. ■



(a) Star graph of Iris 150.

(b) Chernoff face of Iris 150.

Figure 3.27. Star coordinates graph and Chernoff face of the 150^{th} flower of the Iris data set.

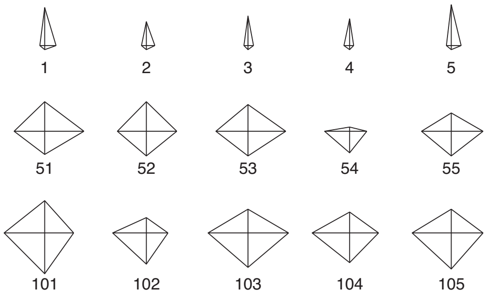


Figure 3.28. Plot of 15 Iris flowers using star coordinates.

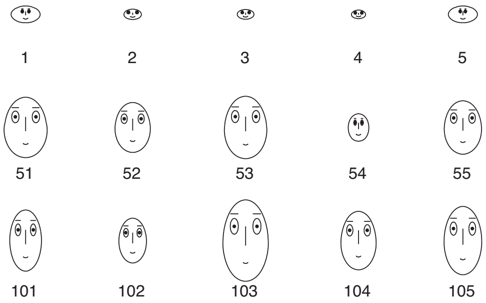


Figure 3.29. A plot of 15 Iris flowers using Chernoff faces.

Despite the visual appeal of these sorts of diagrams, they do not scale well, and thus, they are of limited use for many data mining problems. Nonetheless, they may still be of use as a means to quickly compare small sets of objects that have been selected by other techniques.

3.3.5 Do's and Don'ts

To conclude this section on visualization, we provide a short list of visualization do's and don'ts. While these guidelines incorporate a lot of visualization wisdom, they should not be followed blindly. As always, guidelines are no substitute for thoughtful consideration of the problem at hand.

ACCENT Principles The following are the *ACCENT* principles for effective graphical display put forth by D. A. Burn (as adapted by Michael Friendly):

Apprehension Ability to correctly perceive relations among variables. Does the graph maximize apprehension of the relations among variables?

Clarity Ability to visually distinguish all the elements of a graph. Are the most important elements or relations visually most prominent?

Consistency Ability to interpret a graph based on similarity to previous graphs. Are the elements, symbol shapes, and colors consistent with their use in previous graphs?

Efficiency Ability to portray a possibly complex relation in as simple a way as possible. Are the elements of the graph economically used? Is the graph easy to interpret?

Necessity The need for the graph, and the graphical elements. Is the graph a more useful way to represent the data than alternatives (table, text)? Are all the graph elements necessary to convey the relations?

Truthfulness Ability to determine the true value represented by any graphical element by its magnitude relative to the implicit or explicit scale. Are the graph elements accurately positioned and scaled?

Tufte's Guidelines Edward R. Tufte has also enumerated the following principles for graphical excellence:

- Graphical excellence is the well-designed presentation of interesting data—a matter of *substance*, of *statistics*, and of *design*.

- Graphical excellence consists of complex ideas communicated with clarity, precision, and efficiency.

- Graphical excellence is that which gives to the viewer the greatest number of ideas in the shortest time with the least ink in the smallest space.

- Graphical excellence is nearly always multivariate.

- And graphical excellence requires telling the truth about the data.

3.4 OLAP and Multidimensional Data Analysis

In this section, we investigate the techniques and insights that come from viewing data sets as multidimensional arrays. A number of database systems support such a viewpoint, most notably, On-Line Analytical Processing (OLAP) systems. Indeed, some of the terminology and capabilities of OLAP systems have made their way into spreadsheet programs that are used by millions of people. OLAP systems also have a strong focus on the interactive analysis of data and typically provide extensive capabilities for visualizing the data and generating summary statistics. For these reasons, our approach to multidimensional data analysis will be based on the terminology and concepts common to OLAP systems.

3.4.1 Representing Iris Data as a Multidimensional Array

Most data sets can be represented as a table, where each row is an object and each column is an attribute. In many cases, it is also possible to view the data as a multidimensional array. We illustrate this approach by representing the Iris data set as a multidimensional array.

Table 3.7 was created by discretizing the petal length and petal width attributes to have values of *low*, *medium*, and *high* and then counting the number of flowers from the Iris data set that have particular combinations of petal width, petal length, and species type. (For petal width, the categories *low*, *medium*, and *high* correspond to the intervals [0, 0.75), [0.75, 1.75), [1.75, ∞), respectively. For petal length, the categories *low*, *medium*, and *high* correspond to the intervals [0, 2.5), [2.5, 5), [5, ∞), respectively.) Empty combinations—those combinations that do not correspond to at least one flower—are not shown.

The data can be organized as a multidimensional array with three dimensions corresponding to petal width, petal length, and species type, as

Table 3.7. Number of flowers having a particular combination of petal width, petal length, and species type.

Petal Length	Petal Width	Species Type	Count
low	low	Setosa	46
low	medium	Setosa	2
medium	low	Setosa	2
medium	medium	Versicolour	43
medium	high	Versicolour	3
medium	high	Virginica	3
high	medium	Versicolour	2
high	medium	Virginica	3
high	high	Versicolour	2
high	high	Virginica	44

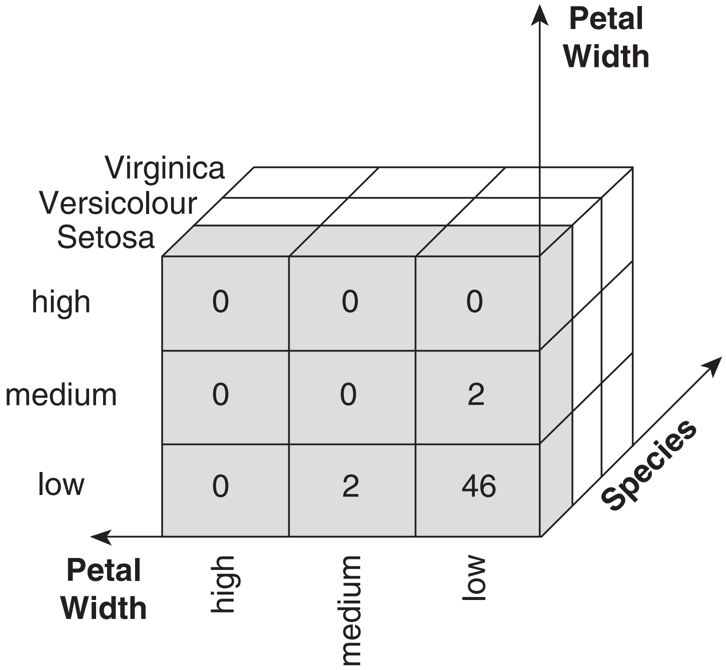


Figure 3.30. A multidimensional data representation for the Iris data set.

Table 3.8. Cross-tabulation of flowers according to petal length and width for flowers of the Setosa species.

		Width		
		low	medium	high
Length	low	46	2	0
	medium	2	0	0
	high	0	0	0

Table 3.9. Cross-tabulation of flowers according to petal length and width for flowers of the Versicolour species.

		Width		
		low	medium	high
Length	low	0	0	0
	medium	0	43	3
	high	0	2	2

Table 3.10. Cross-tabulation of flowers according to petal length and width for flowers of the Virginica species.

		Width		
		low	medium	high
Length	low	0	0	0
	medium	0	0	3
	high	0	3	44

illustrated in Figure 3.30. For clarity, slices of this array are shown as a set of three two-dimensional tables, one for each species—see Tables 3.8, 3.9, and 3.10. The information contained in both Table 3.7 and Figure 3.30 is the same. However, in the multidimensional representation shown in Figure 3.30 (and Tables 3.8, 3.9, and 3.10), the values of the attributes—petal width, petal length, and species type—are array indices.

What is important are the insights can be gained by looking at data from a multidimensional viewpoint. Tables 3.8, 3.9, and 3.10 show that each species of Iris is characterized by a different combination of values of petal length and width. Setosa flowers have low width and length, Versicolour flowers have medium width and length, and Virginica flowers have high width and length.

3.4.2 Multidimensional Data: The General Case

The previous section gave a specific example of using a multidimensional approach to represent and analyze a familiar data set. Here we describe the general approach in more detail.

The starting point is usually a tabular representation of the data, such as that of Table 3.7, which is called a **fact table**. Two steps are necessary in order to represent data as a multidimensional array: identification of the dimensions and identification of an attribute that is the focus of the analysis. The dimensions are categorical attributes or, as in the previous example, continuous attributes that have been converted to categorical attributes. The values of an attribute serve as indices into the array for the dimension corresponding to the attribute, and the number of attribute values is the size of that dimension. In the previous example, each attribute had three possible values, and thus, each dimension was of size three and could be indexed by three values. This produced a 3 × 3 × 3 multidimensional array.

Each combination of attribute values (one value for each different attribute) defines a cell of the multidimensional array. To illustrate using the previous example, if petal length = *low*, petal width = *medium*, and species = Setosa, a specific cell containing the value 2 is identified. That is, there are only two flowers in the data set that have the specified attribute values. Notice that each row (object) of the data set in Table 3.7 corresponds to a cell in the multidimensional array.

The contents of each cell represents the value of a **target quantity** (target variable or attribute) that we are interested in analyzing. In the Iris example, the target quantity is the *number of flowers* whose petal width and length fall within certain limits. The target attribute is quantitative because a key goal of multidimensional data analysis is to look aggregate quantities, such as totals or averages.

The following summarizes the procedure for creating a multidimensional data representation from a data set represented in tabular form. First, identify the categorical attributes to be used as the dimensions and a quantitative attribute to be used as the target of the analysis. Each row (object) in the table is mapped to a cell of the multidimensional array. The indices of the cell are specified by the values of the attributes that were selected as dimensions, while the value of the cell is the value of the target attribute. Cells not defined by the data are assumed to have a value of 0.

Example 3.23. To further illustrate the ideas just discussed, we present a more traditional example involving the sale of products.The fact table for this example is given by Table 3.11. The dimensions of the multidimensional representation are the *product ID*, *location*, and *date* attributes, while the target attribute is the *revenue*. Figure 3.31 shows the multidimensional representation of this data set. This larger and more complicated data set will be used to illustrate additional concepts of multidimensional data analysis. ■

Table 3.11. Sales revenue of products (in dollars) for various locations and times.

Product ID	Location	Date	Revenue
⋮	⋮	⋮	⋮
1	Minneapolis	Oct. 18, 2004	$250
1	Chicago	Oct. 18, 2004	$79
⋮	⋮	⋮	
1	Paris	Oct. 18, 2004	301
⋮	⋮	⋮	⋮
27	Minneapolis	Oct. 18, 2004	$2,321
27	Chicago	Oct. 18, 2004	$3,278
⋮	⋮	⋮	
27	Paris	Oct. 18, 2004	$1,325
⋮	⋮	⋮	⋮

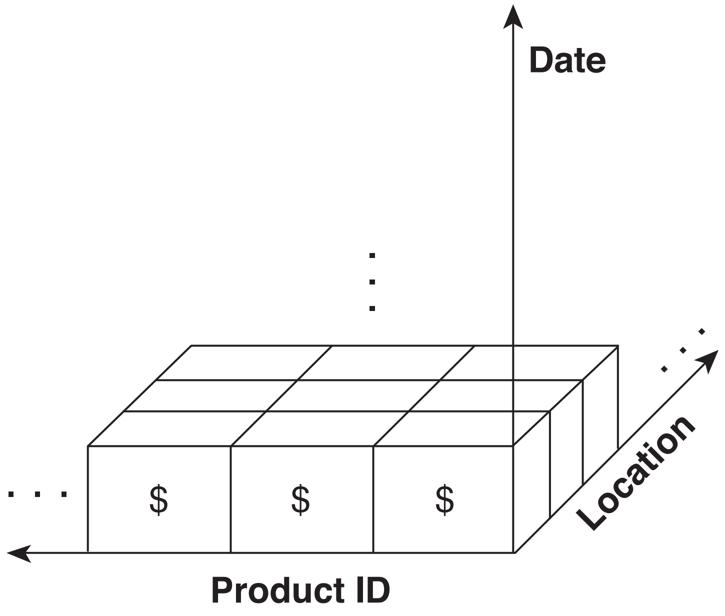


Figure 3.31. Multidimensional data representation for sales data.

3.4.3 Analyzing Multidimensional Data

In this section, we describe different multidimensional analysis techniques. In particular, we discuss the creation of data cubes, and related operations, such as slicing, dicing, dimensionality reduction, roll-up, and drill down.

Data Cubes: Computing Aggregate Quantities

A key motivation for taking a multidimensional viewpoint of data is the importance of aggregating data in various ways. In the sales example, we might wish to find the total sales revenue for a specific year and a specific product. Or we might wish to see the yearly sales revenue for each location across all products. Computing aggregate totals involves fixing specific values for some of the attributes that are being used as dimensions and then summing over all possible values for the attributes that make up the remaining dimensions. There are other types of aggregate quantities that are also of interest, but for simplicity, this discussion will use totals (sums).

Table 3.12 shows the result of summing over all locations for various combinations of date and product. For simplicity, assume that all the dates are within one year. If there are 365 days in a year and 1000 products, then Table 3.12 has 365,000 entries (totals), one for each product-data pair. We could also specify the store location and date and sum over products, or specify the location and product and sum over all dates.

Table 3.13 shows the **marginal totals** of Table 3.12. These totals are the result of further summing over either dates or products. In Table 3.13, the total sales revenue due to product 1, which is obtained by summing across row 1 (over all dates), is $370,000. The total sales revenue on January 1, 2004, which is obtained by summing down column 1 (over all products), is $527,362. The total sales revenue, which is obtained by summing over all rows and columns (all times and products) is $227,352,127. All of these totals are for all locations because the entries of Table 3.13 include all locations.

A key point of this example is that there are a number of different totals (aggregates) that can be computed for a multidimensional array, depending on how many attributes we sum over. Assume that there are n dimensions and that the i^{th} dimension (attribute) has s_i possible values. There are n different ways to sum only over a single attribute. If we sum over dimension j, then we obtain $s_1 * \cdots * s_{j-1} * s_{j+1} * \cdots * s_n$ totals, one for each possible combination of attribute values of the $n-1$ other attributes (dimensions). The totals that result from summing over one attribute form a multidimensional array of $n-1$ dimensions and there are n such arrays of totals. In the sales example, there

Table 3.12. Totals that result from summing over all locations for a fixed time and product.

	date			
product ID	Jan 1, 2004	Jan 2, 2004	…	Dec 31, 2004
1	$1,001	$987	…	$891
⋮	⋮			⋮
27	$10,265	$10,225	…	$9,325
⋮	⋮			⋮

Table 3.13. Table 3.12 with marginal totals.

	date				
product ID	Jan 1, 2004	Jan 2, 2004	…	Dec 31, 2004	total
1	$1,001	$987	…	$891	$370,000
⋮	⋮			⋮	⋮
27	$10,265	$10,225	…	$9,325	$3,800,020
⋮	⋮			⋮	⋮
total	$527,362	$532,953	…	$631,221	$227,352,127

are three sets of totals that result from summing over only one dimension and each set of totals can be displayed as a two-dimensional table.

If we sum over two dimensions (perhaps starting with one of the arrays of totals obtained by summing over one dimension), then we will obtain a multidimensional array of totals with $n - 2$ dimensions. There will be $\binom{n}{2}$ distinct arrays of such totals. For the sales examples, there will be $\binom{3}{2} = 3$ arrays of totals that result from summing over location and product, location and time, or product and time. In general, summing over k dimensions yields $\binom{n}{k}$ arrays of totals, each with dimension $n - k$.

A multidimensional representation of the data, together with all possible totals (aggregates), is known as a **data cube**. Despite the name, the size of each dimension—the number of attribute values—does not need to be equal. Also, a data cube may have either more or fewer than three dimensions. More importantly, a data cube is a generalization of what is known in statistical terminology as a **cross-tabulation**. If marginal totals were added, Tables 3.8, 3.9, or 3.10 would be typical examples of cross tabulations.

Dimensionality Reduction and Pivoting

The aggregation described in the last section can be viewed as a form of **dimensionality reduction**. Specifically, the j^{th} dimension is eliminated by summing over it. Conceptually, this collapses each "column" of cells in the j^{th} dimension into a single cell. For both the sales and Iris examples, aggregating over one dimension reduces the dimensionality of the data from 3 to 2. If s_j is the number of possible values of the j^{th} dimension, the number of cells is reduced by a factor of s_j. Exercise 17 on page 155 asks the reader to explore the difference between this type of dimensionality reduction and that of PCA.

Pivoting refers to aggregating over all dimensions except two. The result is a two-dimensional cross tabulation with the two specified dimensions as the only remaining dimensions. Table 3.13 is an example of pivoting on date and product.

Slicing and Dicing

These two colorful names refer to rather straightforward operations. **Slicing** is selecting a group of cells from the entire multidimensional array by specifying a specific value for one or more dimensions. Tables 3.8, 3.9, and 3.10 are three slices from the Iris set that were obtained by specifying three separate values for the species dimension. **Dicing** involves selecting a subset of cells by specifying a range of attribute values. This is equivalent to defining a subarray from the complete array. In practice, both operations can also be accompanied by aggregation over some dimensions.

Roll-Up and Drill-Down

In Chapter 2, attribute values were regarded as being "atomic" in some sense. However, this is not always the case. In particular, each date has a number of properties associated with it such as the year, month, and week. The data can also be identified as belonging to a particular business quarter, or if the application relates to education, a school quarter or semester. A location also has various properties: continent, country, state (province, etc.), and city. Products can also be divided into various categories, such as clothing, electronics, and furniture.

Often these categories can be organized as a hierarchical tree or lattice. For instance, years consist of months or weeks, both of which consist of days. Locations can be divided into nations, which contain states (or other units of local government), which in turn contain cities. Likewise, any category of products can be further subdivided. For example, the product category, furniture, can be subdivided into the subcategories, chairs, tables, sofas, etc.

This hierarchical structure gives rise to the roll-up and drill-down operations. To illustrate, starting with the original sales data, which is a multidimensional array with entries for each date, we can aggregate (**roll up**) the sales across all the dates in a month. Conversely, given a representation of the data where the time dimension is broken into months, we might want to split the monthly sales totals (**drill down**) into daily sales totals. Of course, this requires that the underlying sales data be available at a daily granularity.

Thus, roll-up and drill-down operations are related to aggregation. Notice, however, that they differ from the aggregation operations discussed until now in that they aggregate cells within a dimension, not across the entire dimension.

3.4.4 Final Comments on Multidimensional Data Analysis

Multidimensional data analysis, in the sense implied by OLAP and related systems, consists of viewing the data as a multidimensional array and aggregating data in order to better analyze the structure of the data. For the Iris data, the differences in petal width and length are clearly shown by such an analysis. The analysis of business data, such as sales data, can also reveal many interesting patterns, such as profitable (or unprofitable) stores or products.

As mentioned, there are various types of database systems that support the analysis of multidimensional data. Some of these systems are based on relational databases and are known as ROLAP systems. More specialized database systems that specifically employ a multidimensional data representation as their fundamental data model have also been designed. Such systems are known as MOLAP systems. In addition to these types of systems, statistical databases (SDBs) have been developed to store and analyze various types of statistical data, e.g., census and public health data, that are collected by governments or other large organizations. References to OLAP and SDBs are provided in the bibliographic notes.

3.5 Bibliographic Notes

Summary statistics are discussed in detail in most introductory statistics books, such as [109]. References for exploratory data analysis are the classic text by Tukey [121] and the book by Velleman and Hoaglin [122].

The basic visualization techniques are readily available, being an integral part of most spreadsheets (Microsoft EXCEL [112]), statistics programs (SAS

[116], SPSS [119], R [113], and S-PLUS [115]), and mathematics software (MATLAB [111] and Mathematica [110]). Most of the graphics in this chapter were generated using MATLAB. The statistics package R is freely available as an open source software package from the R project.

The literature on visualization is extensive, covering many fields and many decades. One of the classics of the field is the book by Tufte [120]. The book by Spence [118], which strongly influenced the visualization portion of this chapter, is a useful reference for information visualization—both principles and techniques. This book also provides a thorough discussion of many dynamic visualization techniques that were not covered in this chapter. Two other books on visualization that may also be of interest are those by Card et al. [104] and Fayyad et al. [106].

Finally, there is a great deal of information available about data visualization on the World Wide Web. Since Web sites come and go frequently, the best strategy is a search using "information visualization," "data visualization," or "statistical graphics." However, we do want to single out for attention "The Gallery of Data Visualization," by Friendly [107]. The ACCENT Principles for effective graphical display as stated in this chapter can be found there, or as originally presented in the article by Burn [103].

There are a variety of graphical techniques that can be used to explore whether the distribution of the data is Gaussian or some other specified distribution. Also, there are plots that display whether the observed values are statistically significant in some sense. We have not covered any of these techniques here and refer the reader to the previously mentioned statistical and mathematical packages.

Multidimensional analysis has been around in a variety of forms for some time. One of the original papers was a white paper by Codd [105], the father of relational databases. The data cube was introduced by Gray et al. [108], who described various operations for creating and manipulating data cubes within a relational database framework. A comparison of statistical databases and OLAP is given by Shoshani [117]. Specific information on OLAP can be found in documentation from database vendors and many popular books. Many database textbooks also have general discussions of OLAP, often in the context of data warehousing. For example, see the text by Ramakrishnan and Gehrke [114].

Bibliography

[103] D. A. Burn. Designing Effective Statistical Graphs. In C. R. Rao, editor, *Handbook of Statistics 9*. Elsevier/North-Holland, Amsterdam, The Netherlands, September 1993.

[104] S. K. Card, J. D. MacKinlay, and B. Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers, San Francisco, CA, January 1999.

[105] E. F. Codd, S. B. Codd, and C. T. Smalley. Providing OLAP (On-line Analytical Processing) to User- Analysts: An IT Mandate. White Paper, E.F. Codd and Associates, 1993.

[106] U. M. Fayyad, G. G. Grinstein, and A. Wierse, editors. *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann Publishers, San Francisco, CA, September 2001.

[107] M. Friendly. Gallery of Data Visualization. http://www.math.yorku.ca/SCS/Gallery/, 2005.

[108] J. Gray, S. Chaudhuri, A. Bosworth, A. Layman, D. Reichart, M. Venkatrao, F. Pellow, and H. Pirahesh. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals. *Journal Data Mining and Knowledge Discovery*, 1(1): 29–53, 1997.

[109] B. W. Lindgren. *Statistical Theory*. CRC Press, January 1993.

[110] Mathematica 5.1. Wolfram Research, Inc. http://www.wolfram.com/, 2005.

[111] MATLAB 7.0. The MathWorks, Inc. http://www.mathworks.com, 2005.

[112] Microsoft Excel 2003. Microsoft, Inc. http://www.microsoft.com/, 2003.

[113] R: A language and environment for statistical computing and graphics. The R Project for Statistical Computing. http://www.r-project.org/, 2005.

[114] R. Ramakrishnan and J. Gehrke. *Database Management Systems*. McGraw-Hill, 3rd edition, August 2002.

[115] S-PLUS. Insightful Corporation. http://www.insightful.com, 2005.

[116] SAS: Statistical Analysis System. SAS Institute Inc. http://www.sas.com/, 2005.

[117] A. Shoshani. OLAP and statistical databases: similarities and differences. In *Proc. of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems*, pages 185–196. ACM Press, 1997.

[118] R. Spence. *Information Visualization*. ACM Press, New York, December 2000.

[119] SPSS: Statistical Package for the Social Sciences. SPSS, Inc. http://www.spss.com/, 2005.

[120] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, March 1986.

[121] J. W. Tukey. *Exploratory data analysis*. Addison-Wesley, 1977.

[122] P. Velleman and D. Hoaglin. *The ABC's of EDA: Applications, Basics, and Computing of Exploratory Data Analysis*. Duxbury, 1981.

3.6 Exercises

1. Obtain one of the data sets available at the UCI Machine Learning Repository and apply as many of the different visualization techniques described in the chapter as possible. The bibliographic notes and book Web site provide pointers to visualization software.

2. Identify at least two advantages and two disadvantages of using color to visually represent information.

3. What are the arrangement issues that arise with respect to three-dimensional plots?

4. Discuss the advantages and disadvantages of using sampling to reduce the number of data objects that need to be displayed. Would simple random sampling (without replacement) be a good approach to sampling? Why or why not?

5. Describe how you would create visualizations to display information that describes the following types of systems.

   (a) Computer networks. Be sure to include both the static aspects of the network, such as connectivity, and the dynamic aspects, such as traffic.

   (b) The distribution of specific plant and animal species around the world for a specific moment in time.

   (c) The use of computer resources, such as processor time, main memory, and disk, for a set of benchmark database programs.

   (d) The change in occupation of workers in a particular country over the last thirty years. Assume that you have yearly information about each person that also includes gender and level of education.

   Be sure to address the following issues:

   - **Representation.** How will you map objects, attributes, and relationships to visual elements?
   - **Arrangement.** Are there any special considerations that need to be taken into account with respect to how visual elements are displayed? Specific examples might be the choice of viewpoint, the use of transparency, or the separation of certain groups of objects.
   - **Selection.** How will you handle a large number of attributes and data objects?

6. Describe one advantage and one disadvantage of a stem and leaf plot with respect to a standard histogram.

7. How might you address the problem that a histogram depends on the number and location of the bins?

8. Describe how a box plot can give information about whether the value of an attribute is symmetrically distributed. What can you say about the symmetry of the distributions of the attributes shown in Figure 3.11?

9. Compare sepal length, sepal width, petal length, and petal width, using Figure 3.12.

10. Comment on the use of a box plot to explore a data set with four attributes: age, weight, height, and income.

11. Give a possible explanation as to why most of the values of petal length and width fall in the buckets along the diagonal in Figure 3.9.

12. Use Figures 3.14 and 3.15 to identify a characteristic shared by the petal width and petal length attributes.

13. Simple line plots, such as that displayed in Figure 2.12 on page 59, which shows two time series, can be used to effectively display high-dimensional data. For example, in Figure 2.12 it is easy to tell that the frequencies of the two time series are different. What characteristic of time series allows the effective visualization of high-dimensional data?

14. Describe the types of situations that produce sparse or dense data cubes. Illustrate with examples other than those used in the book.

15. How might you extend the notion of multidimensional data analysis so that the target variable is a qualitative variable? In other words, what sorts of summary statistics or data visualizations would be of interest?

16. Construct a data cube from Table 3.14. Is this a dense or sparse data cube? If it is sparse, identify the cells that are empty.

Table 3.14. Fact table for Exercise 16.

Product ID	Location ID	Number Sold
1	1	10
1	3	6
2	1	5
2	2	22

17. Discuss the differences between dimensionality reduction based on aggregation and dimensionality reduction based on techniques such as PCA and SVD.