Database Concepts

8th Edition

David M. Kroenke • David J. Auer

Scott L. Vandenberg • Robert C. Yoder

Online Appendix J

Business Intelligence Systems

VP Editorial Director: Andrew Gilfillan
Senior Portfolio Manager: Samantha Lewis
Content Development Team Lead: Laura Burgess
Program Monitor: Ann Pulido/SPi Global
Editorial Assistant: Madeline Houpt
Product Marketing Manager: Kaylee Carlson
Project Manager: Katrina Ostler/Cenveo® Publisher Services
Text Designer: Cenveo® Publisher Services
Interior design: Stock-Asso/Shutterstock; Faysal Shutterstock
Cover Designer: Brian Malloy/Cenveo® Publisher Services
Cover Art: Artwork by Donna R. Auer
Full-Service Project Management: Cenveo® Publisher Services
Composition: Cenveo® Publisher Services
Printer/Binder: Courier/Kendallville
Cover Printer: Lehigh-Phoenix Color/Hagerstown
Text Font: 10/12 Simoncini Garamond Std.

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on the appropriate page within text.

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability, whether express, implied or statutory, fitness for a particular purpose, title and non-infringement. In no event shall Microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time. Partial screen shots may be viewed in full within the software version specified.

Microsoft® Windows®, and Microsoft Office® are registered trademarks of the Microsoft Corporation in the U.S.A. and other countries. This book is not sponsored or endorsed by or affiliated with the Microsoft Corporation.

MySQL®, the MySQL Command Line Client®, the MySQL Workbench®, and the MySQL Connector/ODBC® are registered trademarks of Sun Microsystems, Inc./Oracle Corporation. Screenshots and icons reprinted with permission of Oracle Corporation. This book is not sponsored or endorsed by or affiliated with Oracle Corporation.

Oracle Database XE 2016 by Oracle Corporation. Reprinted with permission.

PHP is copyright The PHP Group 1999–2012, and is used under the terms of the PHP Public License v3.01 available at **http://www.php.net/license/3_01.txt**. This book is not sponsored or endorsed by or affiliated with The PHP Group.

Copyright © 2017, 2015, 2013, 2011 by Pearson Education, Inc., 221 River Street, Hoboken, New Jersey 07030. All rights reserved. Manufactured in the United States of America. This publication is protected by Copyright, and permission should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. To obtain permission(s) to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, 221 River Street, Hoboken, New Jersey 07030.

Many of the designations by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Library of Congress Cataloging-in-Publication Data

Kroenke, David M., 1948- author. | Auer, David J., author.
Database concepts / David M. Kroenke, David J. Auer, Western Washington University, Scott L. Vandenberg, Siena College, Robert C. Yoder, Siena College.
Eighth edition. | Hoboken, New Jersey : Pearson, [2017] | Includes index.
LCCN 2016048321| ISBN 013460153X | ISBN 9780134601533
LCSH: Database management. | Relational databases.
LCC QA76.9.D3 K736 2017 | DDC 005.74--dc23
LC record available at https://lccn.loc.gov/2016048321

Appendix J — 10 9 8 7 6 5 4 3 2 1

Appendix Objectives

- Learn the basic concepts of business intelligence (BI) systems
- Learn the basic concepts of data warehouses and data marts
- Learn the basic concepts of reporting systems
- Learn the basic concepts of data mining
- Learn the basic concepts of market basket analysis
- Learn the basic concepts of decision trees

What Is the Purpose of This Appendix?

In Chapter 8, we discussed Big Data, dimensional databases, and data warehouses in depth. We introduced business intelligence (BI) systems and learned that they can be categorized as reporting systems and data mining systems. We then explored online analytical processing (OLAP) systems, which are a type of BI reporting system.

This appendix takes a more thorough look at BI systems. It recaps some of the material covered in Chapter 8 to provide a context for the new material, and it logically should be studied after the Chapter 8 section on online analytical processing (OLAP) on pages 503–507.

Business Intelligence Systems

As discussed in Chapter 8, **business intelligence (BI) systems** are information systems that assist managers and other professionals in the analysis of current and past activities and in the prediction of future events. Unlike transaction processing systems, they do not support operational activities, such as the recording and processing of orders. Instead, BI systems are used to support management assessment, analysis, planning, control, and, ultimately, decision making.

Reporting Systems and Data Mining Applications

BI systems fall into two broad categories: reporting systems and data mining applications. **Reporting systems** sort, filter, group, and make elementary calculations on operational data. **Data mining applications**, in contrast, perform sophisticated analyses on data, analyses that usually involve complex statistical and mathematical processing. The characteristics of BI applications are summarized in Figure J-1.

Reporting Systems

Reporting systems filter, sort, group, and make simple calculations. All reporting analyses can be performed using standard SQL, although extensions to SQL, such as those used for **online analytical processing (OLAP)**, are sometimes used to ease the task of report production.

Reporting systems summarize the current status of business activities and compare that status with past or predicted future activities. Report delivery is crucial. Reports must be delivered to the proper users on a timely basis in the appropriate format. For example, reports may be delivered on paper, via a Web browser, or in some other format.

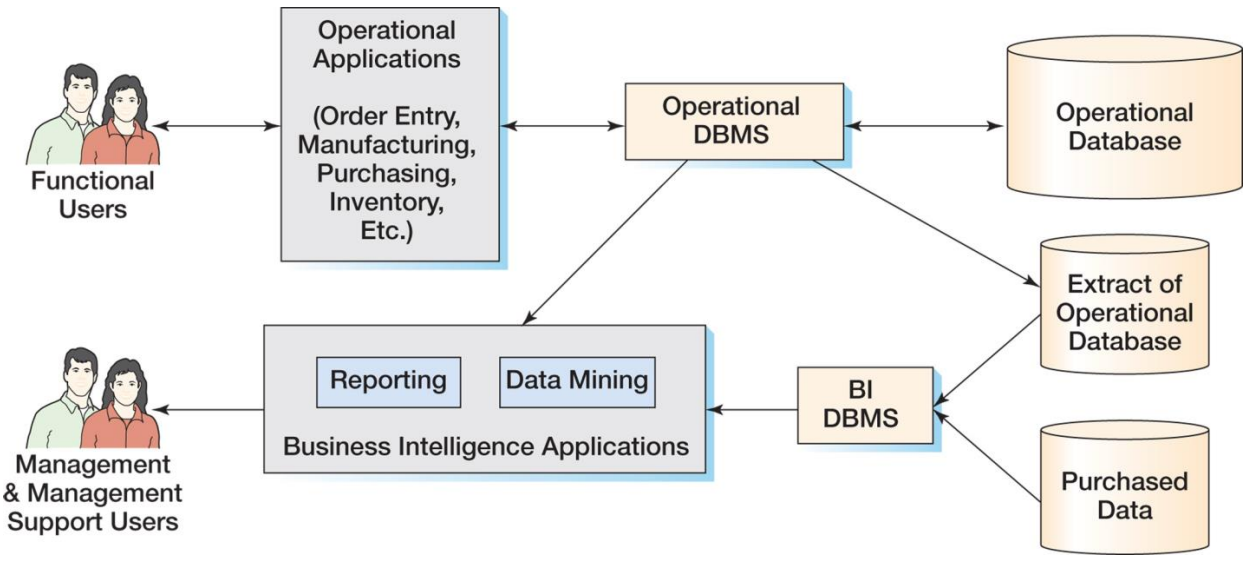


Figure J-1 — Characteristics of Business Intelligence Systems

Data Mining Applications

Data mining applications use sophisticated statistical and mathematical techniques to perform what-if analyses, to make predictions, and to facilitate decision making. For example, data mining techniques can analyze past cell phone usage and predict which customers are likely to switch to a competing phone company. Or data mining can be used to analyze past loan behavior to determine which customers are most (or least) likely to default on a loan.

Report delivery is not as important for data mining systems as it is for reporting systems. First, most data mining applications have only a few users, and those users have sophisticated computer skills. Second, the results of a data mining analysis are usually incorporated into some other report, analysis, or information system. In the case of cell phone usage, the characteristics of customers who are in danger of switching to another company may be given to the sales department for action. Or the parameters of an equation for determining the likelihood of a loan default may be incorporated into a loan approval application.

The Components of a Data Warehouse

A **data warehouse** is a database system that has data, programs, and personnel that specialize in the preparation of data for BI processing. In the context of Figure J-1, this involves creating portions of the "BI DBMS." Figure J-2 shows the components of the basic data warehouse architecture. Data are read from operational databases by the **extract, transform, and load (ETL) system**. The ETL system then cleans and prepares the data for BI processing. This can be a complex process.

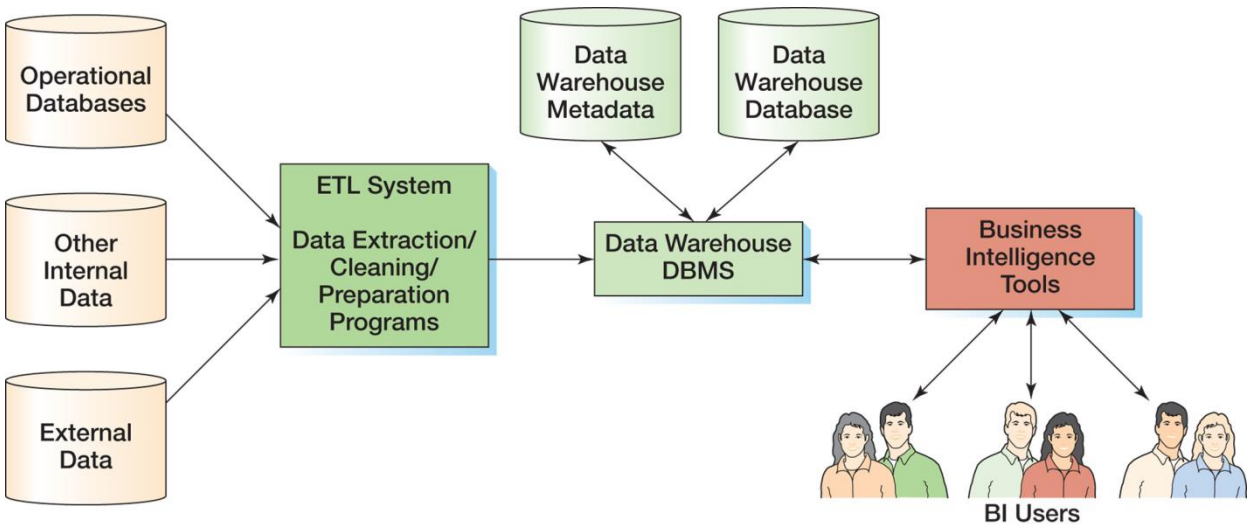


Figure J-2 — Components of a Data Warehouse

First, operational data often cannot be directly loaded into BI applications. Some of the problems of using operational data for BI processing include:

- "Dirty data" (for example, problematic data such as a value of "G" for customer gender, a value of "213" for customer age, a value of "999-999-9999" for a U.S. phone number, or a part color of "gren")
- Missing values
- Inconsistent data (for example, data that have changed, such as a customer's phone number or address, or using both "Yes" and "y" to mean the same thing)
- Nonintegrated data (for example, data from two or more sources that need to be combined for BI use)
- Incorrect format (for example, data that are gathered such that there are either too many data or not enough data, such as time measures in either seconds or hours when they are needed in minutes for BI use)
- Too much data (for example, an excess of columns [attributes], rows [records], or both)

Second, data may need to be changed or transformed for use in a data warehouse. For example, the operational systems may store data about countries using standard two-letter country codes, such as US (United States) and CA (Canada). However, applications using the data warehouse may need to use the country names in full. Thus, the data transformation **{CountryCode → CountryName}** will be needed before the data can be loaded into the data warehouse.

When the data are prepared for use, the ETL system loads the data into the data warehouse database. The extracted data are stored in a data warehouse database using a data warehouse DBMS, which may be from a different vendor than the organization's operational DBMS. For example, an organization might use Oracle for its operational processing but use SQL Server for its data warehouse.

By The Way

Problematic operational data that have been cleaned in the ETL system can also be used to update the operational system to fix the original data problems.

Metadata concerning the data's source, format, assumptions, constraints, and other facts are kept in a **data warehouse metadata database**. The data warehouse DBMS provides extracts of its data to BI tools, such as data mining programs.

Data Warehouses and Data Marts

As shown in Figure J-1, some BI applications read and process operational data directly from the operational database. Although this is possible for simple reporting systems and small databases, such direct reading of operational data is not feasible for more complex applications or larger databases.

Using operational data in BI applications can have cost, performance, and other limitations as outlined below:

- Querying data for BI applications can place a substantial burden on the DBMS and unacceptably slow the performance of operational applications.
- The creation and maintenance of BI systems require application programs, facilities, and expertise that are not normally available from operations.
- Operational data have problems that limit their use for BI applications.

Therefore, larger organizations usually process a separate data warehouse database constructed from an extract of one or more operational databases. You can think of a data warehouse as a distributor in a supply chain. The data warehouse takes data from the data manufacturers (operational systems and purchased data), cleans and processes them, and places the data on the shelves, so to speak, of the data warehouse. The people who work in a data warehouse are experts at data management, data cleaning, data transformation, and the like. However, they are not usually experts in a given business function.

A **data mart** is a collection of data that is smaller than the collection in the data warehouse that addresses a specific component or functional area of the business. A data mart is like a retail store in a supply chain. Users of the data mart obtain data from the data warehouse that pertain to their business function. Such users do not have the data management expertise that data warehouse employees have, but they are knowledgeable analysts for a given business function.

Figure J-3 illustrates these relationships using an example with three data marts (in general, a data warehouse can support any number of data marts). In this example, the data warehouse takes data from the data producers and distributes the data to three data marts. One data mart analyzes *click-stream data* for the purpose of designing Web pages. A second data mart analyzes *store sales data* and determines which products tend to be purchased together to use for training sales staff.

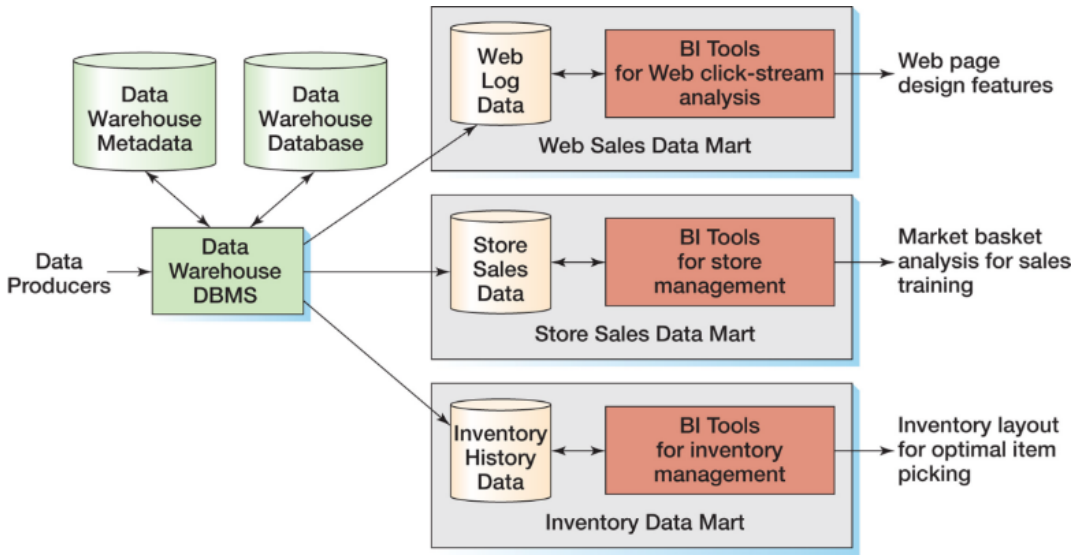


Figure J-3 — Data Warehouses and Data Marts

A third data mart analyzes *customer order data* to assist with reducing labor when picking up items at the warehouse. (Companies such as Amazon.com go to great lengths to organize their warehouses to reduce picking expenses.)

When the data mart structure shown in Figure J-3 is combined with the data warehouse architecture shown in Figure J-2, the combined system is known as an **enterprise data warehouse (EDW) architecture**. In this configuration, the data warehouse maintains all enterprise BI data and acts as the authoritative source for data extracts provided to the data marts. The data marts receive all their data from the data warehouse—they do not add or maintain any additional data.

Of course, it is expensive to create, staff, and operate data warehouses and data marts, and only large organizations with deep pockets can afford to operate a system such as an EDW. Smaller organizations operate subsets of such systems. For example, they may have just a single data mart for analyzing marketing and product promotion data.

Data Warehouses and Dimensional Databases

As discussed in Chapter 8, data warehouse databases are built using a **dimensional database** design. This design typically uses a star schema, as shown in Figure J-4.

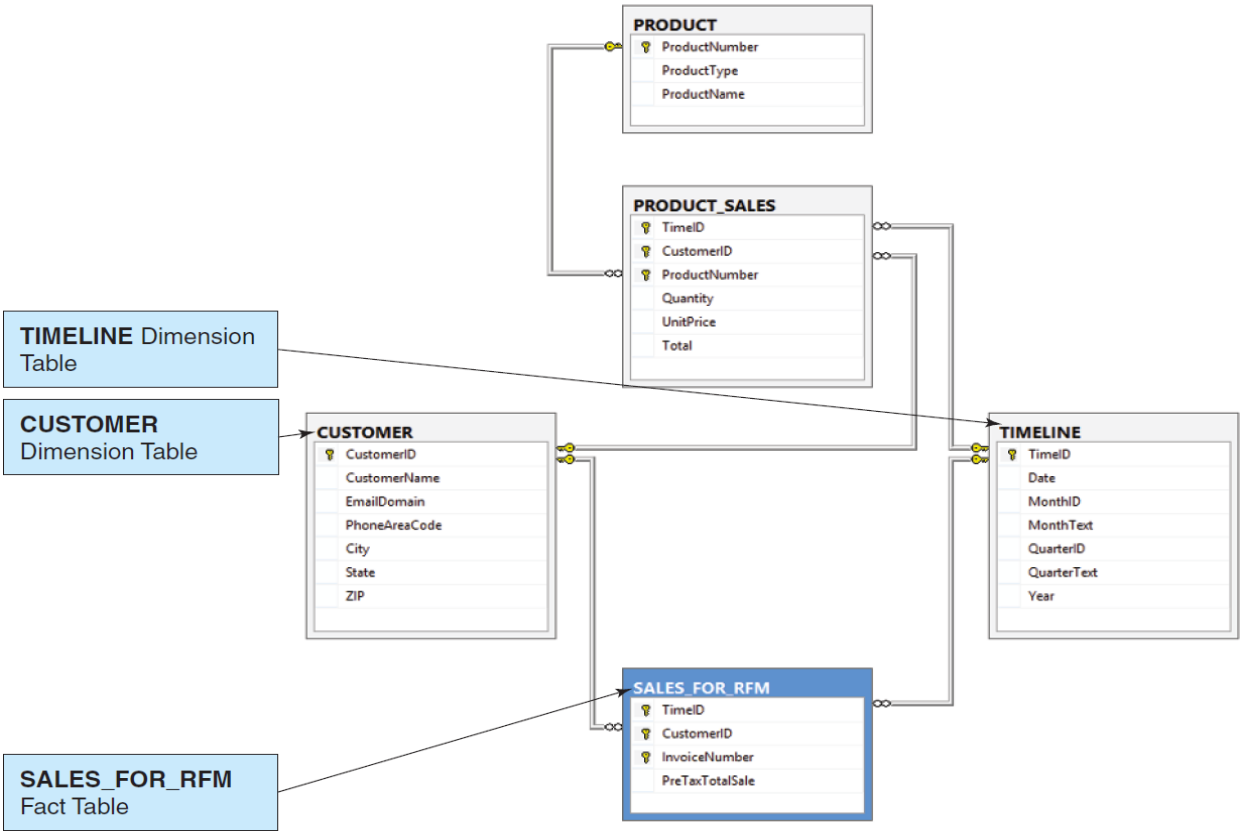


Figure J-4 — Dimensional Databases and the Star Schema

In a dimensional database, **fact tables** (PRODUCT_SALES and SALES_FOR_RFM in the diagram) are linked to **dimension tables** (TIMELINE, PRODUCT, and CUSTOMER in the diagram). A fact table contains facts and some measures of those facts. Each fact is associated with multiple dimensions. Fact tables are on the "many" side of one-to-many relationships.

Reporting Systems

The purpose of a reporting system is to create meaningful information from disparate data sources and to deliver that information to the proper users on a timely basis. Unlike data mining, which uses sophisticated statistical techniques, reporting systems create information by using the basic operations of sorting, filtering, grouping, and making simple calculations.

It is easier to understand reporting systems if you are familiar with a typical report, so let us take a look at a typical reporting problem: RFM analysis.

RFM Analysis

RFM analysis analyzes and ranks customers according to their purchasing patterns. It is a simple customer classification technique that considers how *recently* (**R**) a customer has ordered, how *frequently* (**F**) a customer orders, and *how much money* (**M**) the customer spends per order.

To produce an RFM score, we need only two things: customer data and sales data for each purchase (including the date of the sale and the total amount of the sale) made by each customer. If you look at the SALES_FOR_RFM table and its associated CUSTOMER and TIMELINE dimension tables in Figure J-4, you see that we have exactly those data: The SALES_FOR_RFM table is the starting point for RFM analysis in the HSD-DW BI system, as developed in Chapter 8. Although we will not do it here, RFM analysis can be done using SQL statements and a table such as SALES_FOR_RFM.[1]

To calculate an **R score**, you first sort the customer purchase records by the date of the most recent (R) purchase—note that *only* the most recent purchase for each customer is used in this calculation. In a common form of this analysis, the customers are then divided into five groups, and a score of 1 to 5 is given to customers in each group. The 20 percent of the customers having the most recent orders are given an R score of 1, the 20 percent of the customers having the next most recent orders are given an R score of 2, and so forth, down to the last 20 percent, who are given an R score of 5.

To calculate an **F score**, you re-sort the customers on the basis of how frequently they order. As before, the customers are again divided into five groups. The 20 percent of the customers who order most frequently are given an F score of 1, the next 20 percent most frequently ordering customers are given a score of 2, and so forth, down to the least frequently ordering customers, who are given an F score of 5.

[1] For a full discussion of RFM analysis using SQL statements, see David M. Kroenke and David J. Auer, *Database Processing: Fundamentals, Design, and Implementation*, 14th edition (Upper Saddle River, NJ: Prentice Hall, 2016).

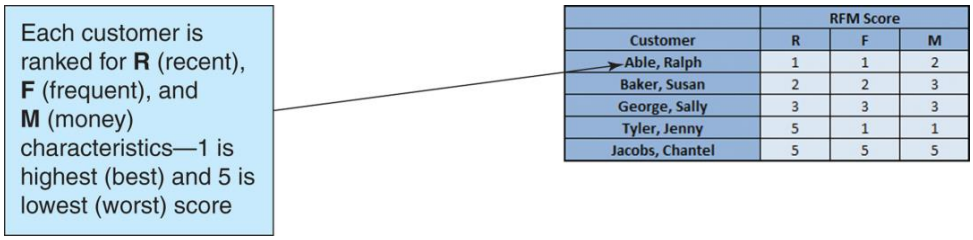


Figure J-5 — The RFM Score Report

To calculate an **M score**, you re-sort the customers according to the average amounts spent on their orders. The 20 percent who have placed the largest orders are given an M score of 1, the next 20 percent are given an M score of 2, and so forth, down to the 20 percent who spend the least, who are given an M score of 5.

Figure J-5 shows sample RFM data for Heather Sweeney Designs. (Note that these data have *not* been calculated and are for illustrative purposes only.) The first customer, Ralph Able, has a score of {1 1 2}, which means that he has ordered recently and orders frequently. His M score of 2 indicates, however, that he does not order the most expensive goods. From these scores, the salespeople can surmise that Ralph is a good customer who may be open to purchasing more expensive goods or higher quantities of goods.

Susan Baker is above average in terms of how recently she has shopped and how frequently she shops, but her purchases are average in value. Sally George is truly in the middle. Based on Jenny Tyler's scores, she has not ordered in some time, but in the past, when she did order, she ordered frequently, and her orders were of the highest monetary value. These data suggest that Jenny may be going to another vendor. Someone from the sales team should contact her immediately. However, no one on the sales team should be talking to Chantel Jacobs. She has not ordered for some time, she does not order frequently, and when she does order, she only buys inexpensive items and not many of them.

Reporting System Components

Figure J-6 shows the major components of a reporting system. Data from disparate data sources are read and processed. As shown, reporting systems can obtain data from operational databases, data warehouses, and data marts.

A reporting system maintains a database of reporting metadata. The metadata describe reports, users, groups, roles, events, and other entities involved in the reporting activity. The reporting system uses the metadata to prepare and deliver appropriate reports to the proper users in the correct format on a timely basis.

As shown in Figure J-6, reports can be prepared in a variety of media or formats. Figure J-7 lists report characteristics, which we describe in more detail next.

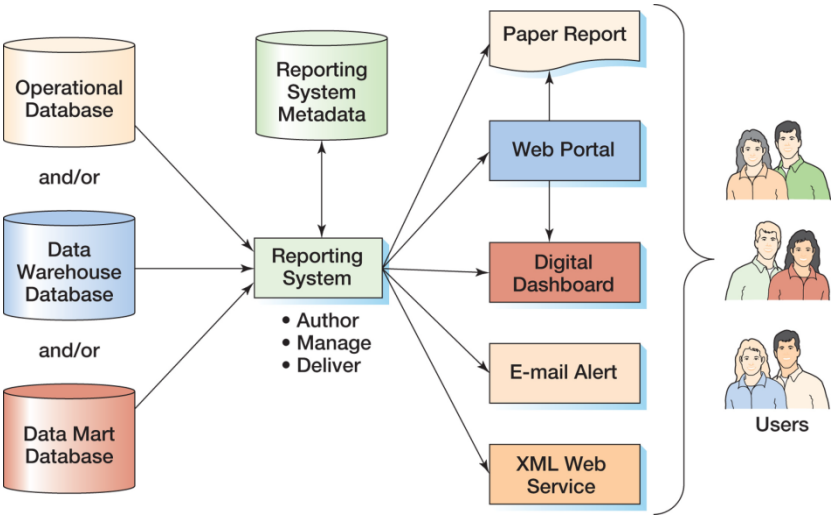


Figure J-6 — Components of a Reporting System

Report Types

Some reports are **static reports**. They are prepared once from the underlying data, and they do not change. A report of the past year's sales, for example, is a static report. Other reports are **dynamic reports**—at the time of their creation, the reporting system reads the latest, most current data and generates the report using those fresh data. Reports on today's sales and on current stock prices are dynamic reports.

Query reports are prepared in response to information entered by users. Google web search is an example of a reporting system that uses query reports: You enter the keywords you want to search for, and Google's reporting system searches its database and generates a response that is custom-built to your query and possibly your location or other factors. Within a specific organization, such as Heather Sweeney Designs, a query report could be generated to show current inventory levels.

Type	Media	Mode
Static	Paper	Push
Dynamic	Web portal	Pull
Query	Digital dashboard	
Online analytical processing (OLAP)	Email/alert	
	XML Web service and application specific	

Figure J-7 — Report Characteristics

The user would enter item numbers, and the reporting system would respond with inventory levels of those items. In terms of the reporting system, **OLAP reports** enable the user to dynamically change the report grouping structures. We discuss OLAP in detail in Chapter 8.

Report Media

As illustrated in Figure J-6 and summarized in Figure J-7, reports are delivered via many different channels. Some reports are printed on paper or its electronic equivalents, such as PDF format. Other reports are delivered via **Web portals**. An organization might place a sales report on the sales department's Web portal and a report on customers serviced on the customer service department's Web portal.

A **digital dashboard** is an electronic display that is customized for a particular user. Companies such as Google, MSN, and Yahoo! offer digital dashboard services that you might have seen or used. Users can define the content they want to see—say, a local weather forecast, a list of stock prices, and a list of news sources—and the vendor constructs a customized display for each user. Such pages are called, for example, myhomemsn.com and My Yahoo. Other dashboards are designed specifically for organizations. Executives at a manufacturing organization, for example, might have a dashboard that shows up-to-the-minute production and sales activities.

Reports can also be delivered via **alerts**. Users can indicate that they want to be notified of news and events by email or cell phone. "Smart cell phones" such as the iPhone and those using the Android operating system are capable of displaying Web pages and can use digital dashboards.

Finally, reports can be delivered to other information systems. The modern way to do this is to publish reports via XML Web Services, as discussed in Chapter 7. This style of reporting is particularly useful for interorganizational information systems, such as supply chain management.

Report Modes

The final report characteristic summarized in Figure J-7 is the report mode. A **push report** is sent to users based on a predetermined schedule. Users receive the report without any activity on their part. In contrast, users must request a **pull report**. To obtain a pull report, a user goes to a Web portal or digital dashboard and clicks a link or button to cause the reporting system to produce and deliver the report.

Report System Functions

As shown in Figure J-6, report systems serve three functions: report management, report authoring, and report delivery. **Report management** consists of defining who receives what reports, when, and by what means. Most report management systems enable the report system administrator to define user accounts and user groups and to assign users to one or more groups. For example, all the salespeople would be assigned to the Sales group, all upper-level management would be assigned to the Executive group, and so forth. All these objects and assignments are stored in the reporting system metadata shown in Figure J-6.

Report authoring involves connecting to the required data sources, creating the report structure, and formatting the report. Reports created using a report authoring system are then assigned to groups and users. Assigning reports to groups saves the administrator work; when a report is created, changed, or removed, the administrator need only change the report assignments of the group, and all the users in the group will inherit the changes. The report assignment metadata not only includes the user or group and the reports assigned but also indicates the format of the report that should be sent to the user, the channel by which the report will be delivered, and whether the report is to be pushed or pulled. If it is to be pushed, the administrator declares whether the report is to be generated on a regular schedule or as an alert based on a specific event in a database.

The **report delivery** function of a reporting system pushes reports or allows them to be pulled based on the report management metadata. Reports can be delivered by hand or via an email server, a Web portal, XML Web Services, or other program-specific means. The report delivery system uses the operating system and other program security components to ensure that only authorized users receive authorized reports, and it also ensures that push reports are produced at appropriate times.

For query reports, the report delivery system serves as an intermediary between the user and the report generator. It receives a user query request, such as the item numbers in an inventory query, passes the query request to the report generator, receives the resulting report, and delivers the report to the user.

OLAP

OLAP, which is discussed in detail in Chapter 8, provides the ability to sum, count, average, and perform other simple arithmetic operations on groups of data. OLAP systems produce OLAP reports. An OLAP report is also called an **OLAP cube**. This is a reference to the dimensional data model discussed in Chapter 8, and some OLAP products show OLAP displays using three axes, like a geometric cube. The remarkable characteristic of an OLAP report is that it is dynamic: The format of an OLAP report can be changed by the viewer, hence the term *online* in the name online analytical processing.

Data Mining

Instead of the basic calculations, filtering, sorting, and grouping used in reporting applications, data mining involves the application of sophisticated mathematical and statistical techniques to find patterns and relationships that can be used to classify data and predict future outcomes. As shown in Figure J-8, data mining represents the convergence of several methodologies. Data mining techniques have emerged from the statistical and mathematics disciplines and from the artificial intelligence and machine-learning communities. In fact, data mining terminology embraces an odd combination of terms used by these different disciplines.

Data mining techniques take advantage of developments for processing enormous databases that have emerged in the past two decades. Of course, all these data would not have been generated were it not for fast and inexpensive computers, and without such computers, results from the new techniques would be impossible to produce in a reasonable timeframe.

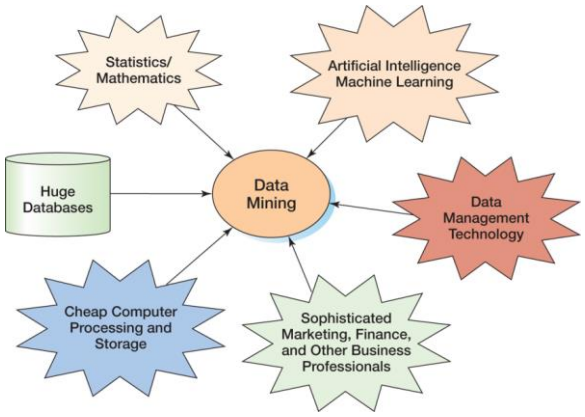


Figure J-8 — Convergence of Disciplines for Data Mining

Most data mining techniques are sophisticated and difficult to use. However, such techniques are valuable to organizations, and some business professionals, especially those in finance and marketing, have developed expertise in their use. Almost all data mining techniques require specialized software. Popular data mining products are Enterprise Miner from SAS Corporation, SPSS Modeler from IBM, and HPE Vertica from Hewlett-Packard. However, there is a movement to make data mining available to more users. For example, Microsoft has created the Microsoft SQL Server Data Mining Add-ins for Office.[2] With this add-in, data stored in Microsoft Excel are sent to SQL Server Analysis Services for processing, and the results are returned to Microsoft Excel for display. Oracle also offers data mining functionality via the "Oracle Advanced Analytics" option, with a GUI interface as part of SQL Developer.

Data mining techniques fall into two broad categories: unsupervised and supervised.

Unsupervised Data Mining

When using **unsupervised data mining** techniques, analysts do not create a model or hypothesis prior to beginning the analysis. Instead, the data mining technique is applied to the data and results are observed. After the analysis, explanations and hypotheses are created to explain the patterns found.

One commonly used unsupervised technique is **cluster analysis**. With cluster analysis, statistical techniques are used to identify groups of entities that have similar characteristics. A common use for cluster analysis is to find customer groups in (1) order data and (2) customer demographic data. For example, Heather Sweeney Designs could use cluster analysis to determine which groups of customers are associated with the purchase of specific products. For example, a cluster analysis could be created using the same HSD-DW data table developed in Chapter 8 to create the OLAP reports. In this case, the cluster analysis tool might indicate that there are different sales patterns for the Dallas area and the non-Dallas area. For example, sales of specific videos may differ markedly in between the two clusters.

[2] The Microsoft SQL Server Data Mining Add-ins for Office are available at **http://msdn.microsoft.com/en-us/library/dn282373.aspx** . Note, however, that these add-ins require a version of SQL Server with SQL Server Analysis Services. In addition, they do not yet work, as of December 2016, with Excel 2016.

Market basket analysis is another form of unsupervised data mining. As this is one of the most common and important forms of data mining, we will describe it in more detail next.

Market Basket Analysis

Data mining techniques are usually complex. However, **market basket analysis** is a data mining technique that can be readily implemented with pure SQL. All the major data mining products have features and functions to perform market basket analysis. Market basket analysis is also known as **association rules**.

Suppose that you run a diving shop, and one day you realize that one of your salespeople is much better than others at up-selling your customers. Any of your sales associates can fill a customer's order, but this particular salesperson is especially able to sell customers items in addition to those for which they ask. One day you ask him how he does it.

"It's simple," he says. "I just ask myself, 'What is the next product they'll want to buy?' If someone buys a dive computer, I don't try to sell her fins. If she's buying a dive computer, she's already a diver, and she already has fins. But, look, these dive computer displays are hard to read. A better mask makes it easier to read the display and get the full benefits from the dive computer." Thus, the market basket analysis might include an association rule that says, "If a customer buys a dive computer, then that customer will also buy a mask." Clearly not all customers buying a dive computer will also buy a mask, of course, so the market basket analysis will need to determine the likelihood that this will occur.

Market basket analysis is a data mining technique for determining such patterns and rules. A market basket analysis shows the products that customers tend to purchase at the same time. Several different statistical techniques can be used to generate a market basket analysis. Here we discuss a technique that involves, as implied above, conditional probabilities.

Figure J-9 shows hypothetical data from 1,000 transactions at a dive shop. The first row of numbers (shaded blue) under each column is the total number of transactions that include the product in that column. For example, the 270 near the top of the Mask column means that 270 of the 1,000 transactions include the purchase of a mask. The 120 under Dive Computer means that 120 of the 1,000 purchase transactions included a dive computer.

Note that in this example, every transaction involves 1 or 2 items among those listed in Figure J-9; those transactions with 2 different items will be counted in two columns of the blue row. Also note that some of the 1,000 transactions do not contain any of the five products listed in the table (e.g., somebody purchases a wet suit and nothing else).

You can use the numbers in the blue row to estimate the probability that a customer will purchase an item. Because 270 out of 1,000 transactions included a mask, you can estimate the likelihood that a customer will buy a mask to be 270/1,000, or .27. Similarly, the likelihood of a tank purchase is 200/1,000, or .2, and the likelihood of a fins purchase is 280/1,000, or .28.

1,000 Transactions	Mask	Tank	Fins	Weights	Dive Computer
	270	200	280	130	120
Mask	20	20	150	20	50
Tank	20	80	40	30	30
Fins	150	40	10	60	20
Weights	20	30	60	10	10
Dive Computer	50	30	20	10	5
No Additional Product	10	–	–	–	5

Support = P (A & B) Example: P (Fins & Mask) = 150 / 1000 = .15
Confidence = P (A | B) Example: P (Fins | Mask) = 150 / 270 = .55556
Lift = P (A | B)/ P (A) Example: P (Fins | Mask) / P (Fins) = .55556 / .28 = 1.98
Note: P (Mask | Fins) / P (Mask) = 150 / 280 / .27 = 1.98

Figure J-9 — A Market Basket Analysis Example

The next five rows in this table show the occurrences of transactions that involve two items. For example, the last column indicates that 50 transactions included both a dive computer and a mask, 30 transactions included a dive computer and a tank, 20 included a dive computer and fins, 10 included a dive computer and weights, 5 included a dive computer and another dive computer (meaning the customer bought two dive computers), and 5 transactions had a dive computer and no other product. Note the symmetry in the table: The numbers for (mask, fins) and (fins, mask) are the same, etc.

These data are interesting, but we can refine the analysis by computing additional factors. Marketing professionals define **support** as the probability that two items will be purchased together. From these data, the support for fins and mask is 150 out of 1,000, or .15. Similarly, the support for dive computer and mask (the combination cited earlier by a salesperson) is 50/1,000, or .05.

Confidence is defined as the probability of a customer buying one product, given that he or she is buying another product. The confidence of fins, given that the customer is purchasing a mask, is the number of purchases of fins and masks out of the number of purchases of masks. Thus, in this example, the confidence is 150 out of 270, or .55556. The confidence that a customer purchases a tank, given that the customer is purchasing fins, is 40 out of 280, or .14286. As another example, the confidence in the rule alluded to by our star salesperson ("If a customer buys a dive computer, then that customer will also buy a mask") is 50/120 = .41667, or roughly 42%.

Lift is defined as the ratio of confidence divided by the base probability of an item purchase. The lift for fins, given a mask, is the probability that a customer buys fins (given that the customer is purchasing a mask) divided by the overall probability that the customer buys fins. If the lift is greater than 1, then the probability of buying fins goes up when a customer buys a mask; if the lift is less than 1, the probability of buying fins goes down when a customer buys a mask.

For the data in Figure J-9, the lift for fins, given a mask purchase, is .55556/.28 or 1.98. This means that when someone purchases a mask, the likelihood he or she will also purchase fins almost doubles. The lift for fins, given a dive computer purchase, is 20/120 (the confidence of fins, given a dive computer) divided by .28, the probability that someone buys fins (280 of the 1,000 transactions involved fins). Therefore, 20/120 is .16667, and .16667/.28 is .59525. So the lift for fins, given purchase of a dive computer, is just under .6, meaning that when a customer buys a dive computer the likelihood that he or she will buy fins decreases. Finally, returning to our salesperson's example, the lift for a mask, given the purchase of a dive computer, is .41667 (the confidence in our "rule") divided by the overall likelihood of buying a mask, which is .27. This gives a lift of .41667/.27 = 1.5432, meaning that our salesperson's intuition was correct: Purchasing a dive computer increases the odds of a mask being purchased at the same time. Note that, as shown in the last line of Figure J-9, lift is symmetrical. If the lift of fins, given purchase of a mask, is 1.98, then the lift of a mask, given purchase of fins, is also 1.98.

Supervised Data Mining

When using **supervised data mining** techniques, data miners develop a model prior to the analysis and then apply statistical techniques to the data to estimate parameters of the model. For example, suppose that marketing experts at a communications company believe that the use of cell phone weekend minutes is determined by the age of the customer and the number of months the customer has had the cell phone account. A data mining analyst could then run a statistical analysis technique known as **regression analysis** to determine the coefficients of the equation of that model. A possible result is:

$$\text{CellPhoneWeekendMinutes} = 12 + (17.5 * \text{CustomerAge}) + (23.7 * \text{NumberMonthsOfAccount})$$

Considerable skill is required to interpret and adjust the quality of such a model: The software will create an equation, but whether the equation becomes a good predictor of future cell phone usage depends on a variety of factors beyond the scope of this book.

Three other popular supervised data mining techniques are decision tree analysis, logistic regression, and neural networks. **Decision tree analysis** classifies customers or other entities of interest into two or more groups, according to history. **Logistic regression** produces equations that offer probabilities that particular events will occur. Common applications of logistic regression are using donor characteristics to predict the likelihood of a donation in a given period and using customer characteristics to predict the likelihood that customers will switch to another vendor. **Neural networks** are complex statistical prediction techniques. The name is actually a misnomer—although there is some similarity between the structure of a neural network and a network of biological neurons, the similarity is only superficial. In data mining, neural networks are just a technique for creating very complex mathematical functions for making predictions.

Decision Trees

The potential for straightforward graphical representation makes decision trees possible to interpret without a lot of background and training. Constructing decision trees, on the other hand, requires sophisticated algorithms and enough experience to properly parameterize those algorithms. In this section, we will focus on the intuition behind decision tree structure and usage by examining an example that is further explored in the exercises.

A decision tree intuitively represents a set of rules that can be easily expressed in English or in SQL. As a simple example, consider the situation of a child deciding whether to read a particular book. Based on his or her experience, the child has determined some characteristics of books that are likely to make them either good or bad choices for the child to read. For example, the child may not like books that are too long and may like short books only if they have plenty of pictures. Without realizing it, that child is using a decision tree similar to that shown in Figure J-10.

The structure in Figure J-10 is referred to as a tree (in computing, we draw trees upside-down, with the root of the tree at the top of the diagram and the leaves at the bottom). Any tree, including decision trees, will have just one **root** and will have just one way to get from the root to any specific **leaf**. In this example, as with all decision trees, we begin with a single question asked at the root of the tree. We will represent questions by rectangles and decisions using ovals.

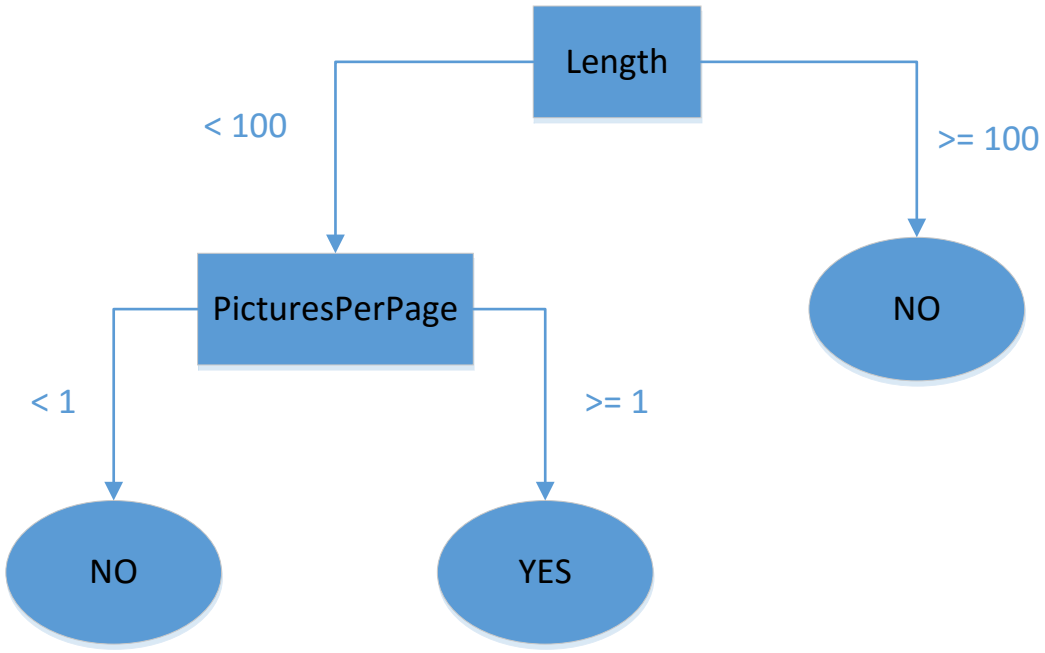


Figure J-10 — A Decision Tree Example

The question first asked by the child is whether the book is less than 100 pages long or at least 100 pages long. The root question of the tree contains the name of the attribute being examined, in this case the Length of the book. If the book is at least 100 pages long, then the questioning follows the arrow coming out of the right side of the root. At the end of that arrow is a leaf with the word *NO*, indicating that the child is not interested in reading this book. Note that the arrow is labeled with the value(s) of the Length attribute that will lead the questioning that way.

If the answer to the question at the root is that the book is less than 100 pages long, then another question is required in order to make the final decision. In this case, we follow the arrow to the left of the root (labeled "< 100") to arrive at the second question. This question is based on how many pictures the book has. If the book has at least one picture per page, then the child wants to read the book. This corresponds to following the arrow coming from the right side of the PicturesPerPage rectangle. This arrow is labeled, as elsewhere, with the conditions under which it will be followed (PicturesPerPage >= 1). On the other hand, if the book has less than one picture per page, on average, then the left arrow will be followed from the PicturesPerPage rectangle and the decision will be to not read the book.

Where did this decision tree come from? It is based on the child's experience reading and examining other books. In general, a decision tree is created in two main phases. In the **training phase**, the decision tree is constructed based on data found, for example, in a data warehouse. These data will include the proper classification (answer) for each record. In the child's books example, the data would include such things as "*War and Peace* is over 100 pages long, it has no pictures, and I did not like it" or "*Green Eggs and Ham* is less than 100 pages long, it has many pictures, and I did like it." These are the training data for the decision tree.

The exact process used to build a decision tree can be very complicated. There are many algorithms for creating decision trees, and they need to determine which attribute should be used for each question and which values of each attribute should be used to guide the questioning to the next level down in the tree. We will not discuss these algorithms further.

The second phase of decision tree creation is called the **testing phase**. In this phase, we give the tree some "new" data points for which we already know the answer and judge its results. This may cause us to alter the decision tree in various ways. After the testing phase is complete, the tree is ready to be deployed and used to make future decisions. It can, of course, always be refined more in the future as we learn how well it continues to classify the data. For example, as the child grows up, he or she may learn to like longer books a little better and could then change the "100" values in the decision tree to "200."

A decision tree represents a set of rules that are used to make a decision about a record that represents a sample or an event. Ideally, this tree will represent a short series of simple questions. The rectangles and ovals in our trees are called **nodes**. Each rectangular node in the decision tree is a question, and each oval-shaped node is a final answer. In our first example in Figure J-10, the questions all have only two possible outcomes, but in general each question can have any number of possible results (at least two, of course). Starting at the root, the answer to the question determines which node to visit next.

This process is continued until a leaf is reached. The full series of questions from the root to the leaf represents a classification rule. For example, in the case of *Green Eggs and Ham* mentioned above, the classification rule used is:

Length < 100 and PicturesPerPage >= 1

To illustrate the difficulties of constructing an accurate, efficient decision tree, consider the tree in Figure J-11, which solves the same problem as that in Figure J-10. The decision tree in Figure J-11 will come to the same decisions about books as will that in Figure J-10, but for some books this decision will now take longer to arrive at. In particular, long books with many pictures will now require two questions to rule out rather than one. Another disadvantage of this decision tree will be explored in the exercises.

Now that we have seen a basic example and introduced the terminology, we will consider a more complete example and illustrate how it might appear in Oracle Data Miner. In addition, we have not yet considered the accuracy of our decision trees: How well do they truly make decisions for both the training and testing data, and thus how well do we expect them to perform on future data? Consider the problem of deciding whether to go outside and ice skate on a nearby lake. From past experience, we have the following data and decisions from the past to guide us (the training data), based on observations of weather conditions (sunny or cloudy), temperature, and the number of days that ice fishing has been observed so far this season:

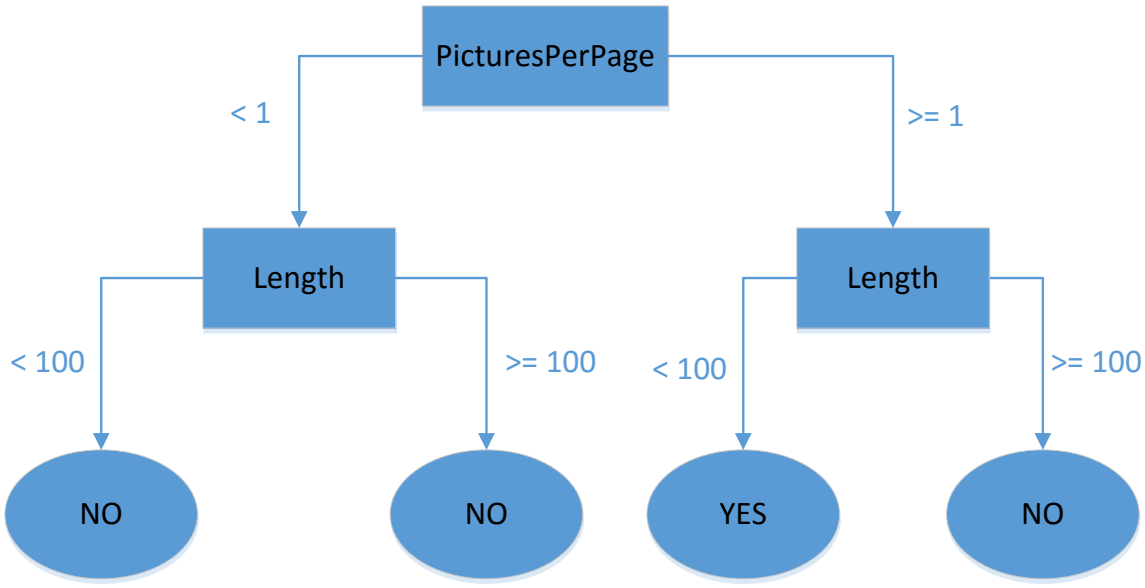
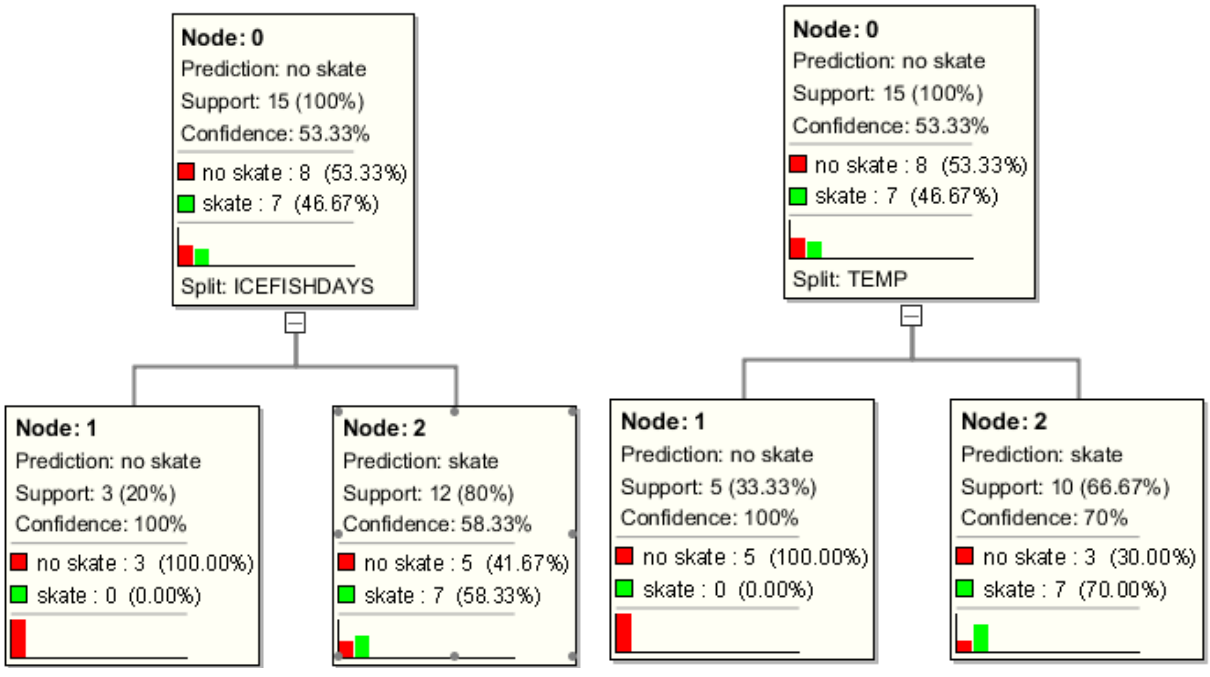


Figure J-11 — A Second Decision Tree Example

	WEATHER	TEMP	ICEFISHDAYS	CLASS
1	cloudy	32	2	no skate
2	sunny	32	17	skate
3	cloudy	10	15	no skate
4	sunny	7	28	skate
5	cloudy	-5	38	no skate
6	sunny	26	23	skate
7	sunny	-7	12	no skate
8	cloudy	26	20	skate
9	sunny	17	3	no skate
10	sunny	-3	19	no skate
11	cloudy	35	35	skate
12	sunny	-10	32	no skate
13	cloudy	15	13	skate
14	sunny	-6	4	no skate
15	cloudy	27	10	skate

Our decision (the "class" attribute) is either to skate or not skate, depending on various combinations of weather, temperature, and ice fishing duration. We want to build a decision tree to help us make the decision in the future. Figure J-12 shows two small decision trees created by Oracle Data Miner to classify the data as "skate" or "no skate." Both trees consist of just one question and two leaves (note that Oracle Data Miner uses rectangles for both question nodes and leaf, or decision, nodes). Figure J-12(a) bases its first question on the ICEFISHDAYS attribute, with the left leaf corresponding to



(a) First Question Based on ICEFISHDAYS (b) First Question Based on TEMP

Figure J-12 — Oracle Data Miner Decision Trees for the Ice Skating Example

ICEFISHDAYS <= 7 and the right leaf (labeled Node 2) corresponding to ICEFISHDAYS > 7. Note that each leaf node contains a support and confidence for the rule that led to that node. Node 1 represents all three records with ICEFISHDAYS <= 7, and all of those (confidence 100%) are labeled "no skate." Node 2, however, is not very good at predicting the outcome: It predicts that all 12 of the other training records correspond to "skate," when in fact only 7 of the 12 do (giving us 58.33% confidence in this rule). In Figure J-12(b), Oracle Data Miner has chosen a different first question, this time based on TEMP, with Node 1 corresponding to TEMP <= 2. This decision tree does a better job at accurately determining the decision, but both these trees are based on a very small amount of data. A more accurate decision tree for the skating data set is shown in Figure J-13, using the same notation as in Figures J-10 and J-11.

A more realistic decision tree generated by Oracle Data Miner, based on a larger data set,[3] is shown in Figure J-14. In this figure, we see part of a decision tree that does a very good job deciding whether a

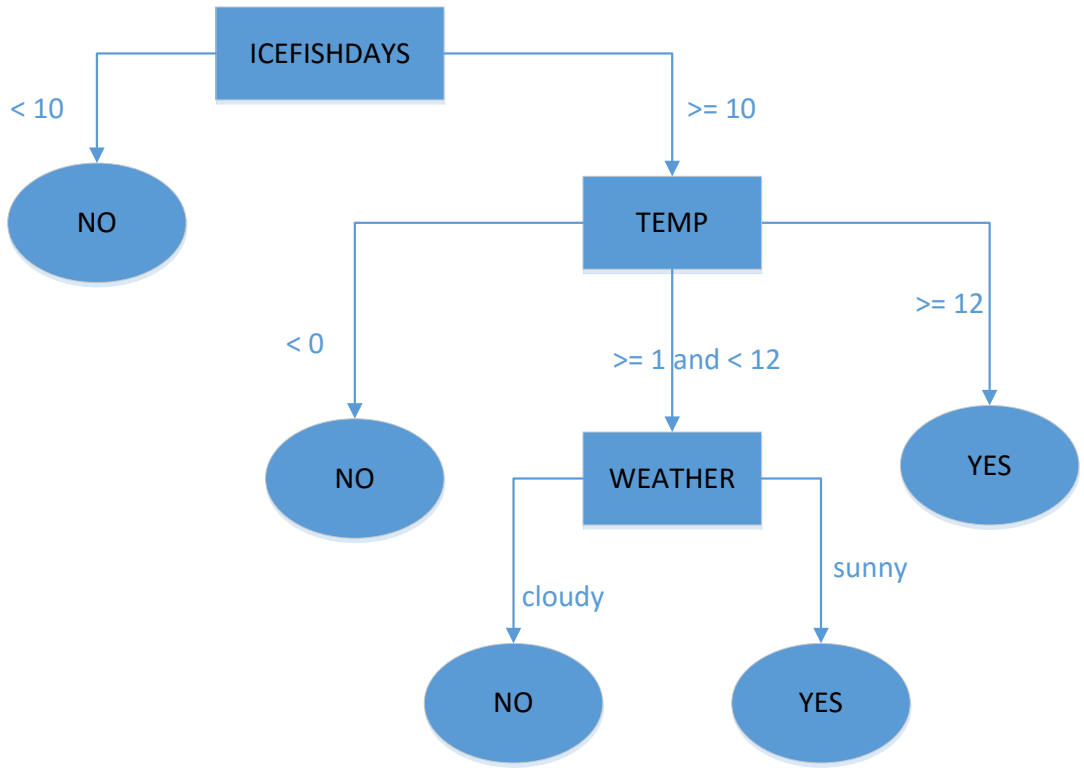


Figure J-13 — A Decision Tree for the Skating Data Set

[3] Poisonous mushroom identification data set obtained from the UCI Machine Learning Repository: M. Lichman, *UCI Machine Learning Repository, http://archive.ics.uci.edu/ml*. Irvine, CA: University of California, School of Information and Computer Science (accessed November 2016).

mushroom is poisonous (p) or edible (e) based on some of its characteristics. The rule for Node 8 (a leaf node) is displayed at the bottom of the figure, and it displays the series of questions/answers (based on the mushroom's odor, spore color, and bruising) that led to that leaf node, which represents 44 poisonous mushrooms and 0 edible mushrooms. Thus, if you examine a mushroom that has those characteristics, you can assume (with confidence 1.0 = 100%) it is poisonous.

Summary

Business intelligence (BI) systems assist managers and other professionals in the analysis of current and past activities and in the prediction of future events. BI applications are of two major types: reporting

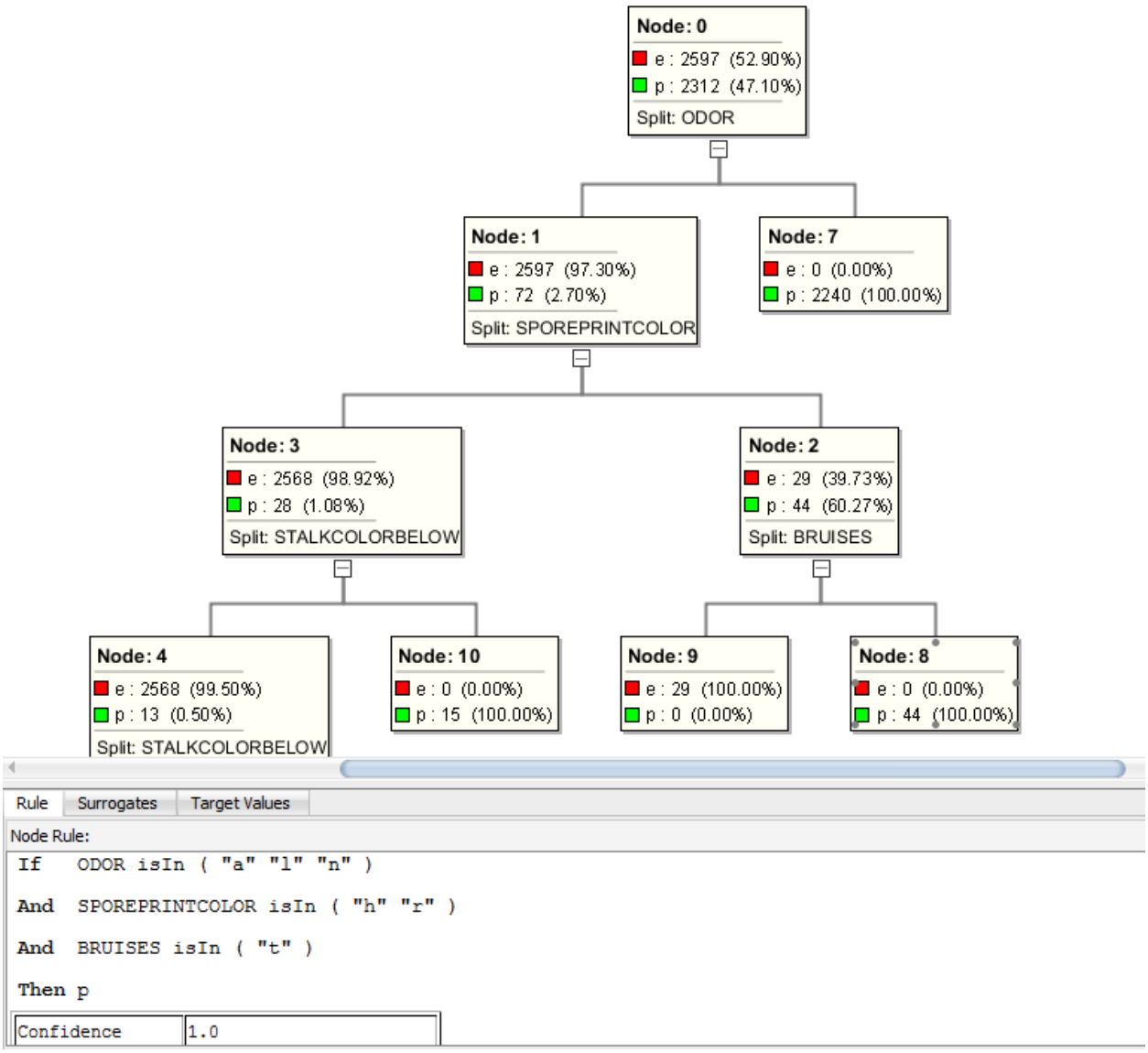


Figure J-14 — Oracle Data Miner Partial Decision Tree and Rule for Poisonous Mushroom Data

applications and data mining applications. Reporting applications make elementary calculations on data; data mining applications use sophisticated mathematical and statistical techniques.

BI applications obtain data from three sources: operational databases, extracts of operational databases, and purchased data. A BI system sometimes has its own DBMS, which may or may not be the operational DBMS.

Direct reading of operational databases is not feasible for any but the smallest and simplest BI applications and databases—for several reasons. Querying operational data can unacceptably slow the performance of operational systems, operational data have problems that limit their usefulness for BI applications, and BI system creation and maintenance require programs, facilities, and expertise that are normally not available for an operational database.

Operational data may have problems. Because of the problems with operational data, many organizations have chosen to create and staff data warehouses and data marts. Extract, transform, and load (ETL) systems are used to extract data from operational systems; transform the data and load them into data warehouses; and maintain metadata that describe the source, format, assumptions, and constraints about the data. A data mart is a collection of data that is smaller than that held in a data warehouse and that addresses a particular component or functional area of the business. In Figure J-3, the enterprise data warehouse distributes data to three smaller data marts, each of which services the needs of a different aspect of the business.

The purpose of a reporting system is to create meaningful information from disparate data sources and to deliver that information to the proper users on a timely basis. Reports are produced by sorting, filtering, grouping, and making simple calculations on the data. RFM analysis is a typical reporting application. Customers are grouped and classified according to how recently they have placed an order (R), how frequently they order (F), and how much money (M) they spend on orders. The result of an RFM analysis is three scores. In a typical analysis, the scores range from 1 to 5. An RFM score of {1 1 4} indicates that the customer has purchased recently, purchases frequently, and does not purchase large-dollar items. An RFM report can be produced using SQL statements.

For RFM data to add value to an organization, an RFM report must be prepared and delivered to the appropriate users. The components of a modern reporting system are shown in Figure J-6. Reporting systems maintain metadata that support the three basic report functions: authoring, managing, and delivering reports. The metadata include information about users, user groups, and reports and data about which users are to receive which reports, in what medium, and when. As shown in Figure J-7, reports vary by type, media, and mode.

Online analytical processing (OLAP) reporting applications, discussed in detail in Chapter 8, enable users to dynamically restructure reports. A fact table contains facts and some measures of those facts. Each fact is associated with multiple dimensions. An OLAP report, or OLAP cube, is an arrangement of measures and dimensions.

Data mining is the application of mathematical and statistical techniques to find patterns and relationships and to classify records and predict outcomes based on the data. Data mining has arisen in recent years because of the confluence of factors shown in Figure J-8.

With unsupervised data mining, analysts do not create models or hypotheses prior to the analysis. Rather, results are explained after the analysis is performed. With supervised techniques, hypotheses are formed and tested before the analysis. Six popular data mining techniques are cluster analysis, market basket analysis, regression analysis, decision tree analysis, logistic regression, and neural networks. The first two of those are unsupervised techniques.

Market basket analysis, or association rules, can be used to determine which sets of products are likely to be sold at the same time. According to market basket analysis terminology, the support for two products is the frequency with which they appear together in transactions. Confidence is the conditional probability that one item will be purchased, given that another item is being purchased. Lift is confidence divided by the base probability that an item will be purchased.

Decision trees can be used to predict future decisions based on the results of past decisions. By asking a proper series of questions about a new data point or record, it is possible to reach a decision about how to classify that data point. These questions can be organized as a tree, with every new data point being asked the same question initially and subsequent questions based on the responses to the previous questions. Decision trees can be used to determine levels of risk in the insurance industry or in medical diagnosis.

KEY TERMS

alert	**association rules**
business intelligence (BI) system	**cluster analysis**
confidence	**data mart**
data mining application	**data warehouse**
data warehouse metadata database	**decision tree analysis**
digital dashboard	**dimension table**
dimensional database	**dynamic report**
enterprise data warehouse (EDW) architecture	**extract, transform, and load (ETL) system**
F score	**fact table**
leaf	**lift**

logistic regression	**M score**
market basket analysis	**neural network**
node	**OLAP cube**
OLAP report	**online analytical processing (OLAP)**
pull report	**push report**
query report	**R score**
regression analysis	**report authoring**
report delivery	**report management**
reporting system	**RFM analysis**
root	**static report**
supervised data mining	**support**
testing phase	**training phase**
unsupervised data mining	**Web portal**

REVIEW QUESTIONS

J.1 What are BI systems?

J.2 How do BI systems differ from transaction processing systems?

J.3 Name and describe the two main categories of BI systems.

J.4 What are the three sources of data for BI systems?

J.5 Summarize the problems with operational databases that limit their usefulness for BI applications.

J.6 What is an ETL system, and what functions does it perform?

J.7 What problems in operational data create the need to clean data before loading the data into a data warehouse?

J.8 What does it mean to transform data? Give an example other than the ones used in this book.

J.9 Why are data warehouses necessary?

J.10 Give examples of data warehouse metadata.

J.11 Explain the difference between a data warehouse and a data mart. Give an example other than the ones used in this book.

J.12 What is the enterprise data warehouse (EDW) architecture?

J.13 State the purpose of a reporting system.

J.14 In RFM analysis, what do the letters *RFM* stand for?

J.15 Describe, in general terms, how to perform an RFM analysis.

J.16 Explain the characteristics of customers that have the following RFM scores:

{1 1 5}, {1 5 1}, {5 5 5}, {2 5 5}, {5 1 2}, {1 1 3}

J.17 Name and describe the purpose of the major components of a reporting system.

J.18 What are the major functions of a reporting system?

J.19 Summarize the types of reports described in this chapter.

J.20 Describe the various media used to deliver reports.

J.21 Summarize the modes of reports described in this chapter.

J.22 Describe the major tasks in report management. Explain the role of report metadata in report management.

J.23 Name three tasks of report authoring.

J.24 Describe the major tasks in report delivery.

J.25 What does OLAP stand for?

J.26 Define *data mining*.

J.27 Explain the difference between unsupervised and supervised data mining.

J.28 Name five popular data mining techniques.

EXERCISES

Use the data in Figure J-9 to answer questions J.29 through J.35.

J.29 What is the probability that someone will buy a tank?

J.30 What is the support for buying a tank and fins? What is the support for buying two tanks?

J.31 What is the confidence for fins, given that a tank has been purchased?

J.32 What is the confidence for a second tank, given that a tank has been purchased?

J.33 What is the lift for fins, given that a tank has been purchased?

J.34 What is the lift for a second tank, given that a tank has been purchased?

J.35 How many transactions are there (among the 1,000) that involve none of the five products mentioned in the table (mask, fins, tanks, dive computer, and weights)?

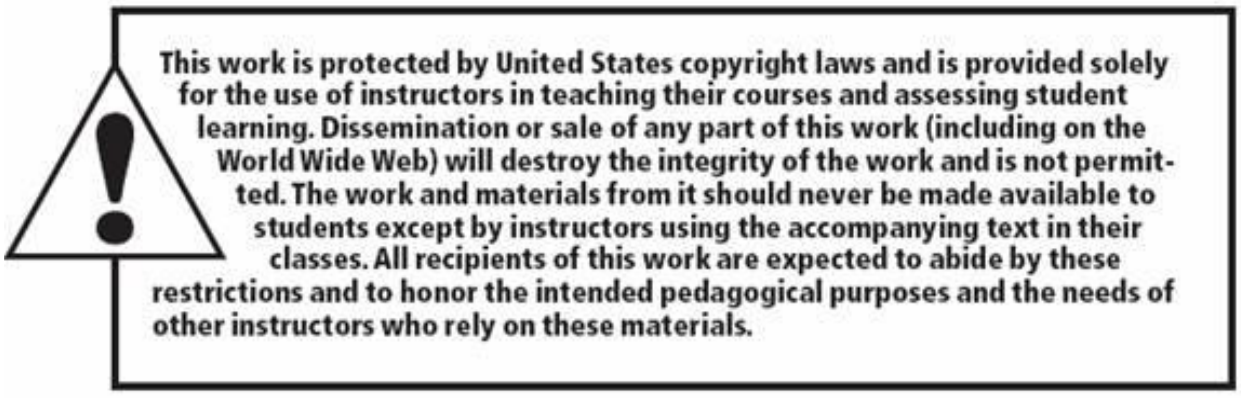
J.36 How could you improve the decision tree in Figure J-11 to be more efficient?

Use the decision tree in Figure J-13 to answer questions J.37 through J.39.

J.37 Would the new data point (record) (cloudy, -3, 16) be classified as “skate” or “no skate”? Which nodes (questions) in the tree would be asked of this new record?

J.38 Would the new data point (record) (sunny, 5, 22) be classified as “skate” or “no skate”? Which nodes (questions) in the tree would be asked of this new record?

J.39 Draw a different decision tree, based on the same data, by basing the second question on a different attribute. Does your tree ask more or fewer questions, on average, to categorize a new point when compared to the tree presented in the text? Does your tree have higher or lower accuracies for its decisions?


This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work (including on the World Wide Web) will destroy the integrity of the work and is not permitted. The work and materials from it should never be made available to students except by instructors using the accompanying text in their classes. All recipients of this work are expected to abide by these restrictions and to honor the intended pedagogical purposes and the needs of other instructors who rely on these materials.