

Oracle® Data Mining

Administrator's Guide

11g Release 2 (11.2)

E16807-07

July 2011

Oracle Data Mining Administrator's Guide, 11g Release 2 (11.2)

E16807-07

Copyright © 1996, 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Kathy L. Taylor

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documentation	vi
Where to Find Database Tuning Documentation	vi
Conventions	vi
What's New in Oracle Data Mining Administration?	ix
Oracle Database 11g Release 2 (11.2.0.3) Oracle Data Mining	ix
Oracle Database 11g Release 2 (11.2.0.2) New Features in Oracle Data Mining	ix
Oracle Database 11g Release 1 New Features in Oracle Data Mining	x
1 Getting Started with Oracle Data Mining	
Step One: Install the Database	1-1
Step Two: Verify the Data Mining Installation	1-2
Step Three: Create a Data Mining User	1-2
Step Four: If You Want to Use Oracle Data Miner	1-3
Step Five: If You Want to Use the Spreadsheet Add-In	1-3
Step Six: If You Want to Use the Sample Programs	1-4
2 Installing Oracle Data Mining	
Install Oracle Database	2-1
Verify the Data Mining Installation	2-9
Enable or Disable the Data Mining Option	2-9
Optionally Install Oracle Database Examples	2-10
Optionally Install Oracle Data Miner	2-11
Optionally Install the Spreadsheet Add-In	2-11
3 Connecting to an Oracle Database	
Connect Application Development Tools	3-1
Connect SQL*Plus	3-1
Connect SQL Developer	3-2
Connect Oracle Data Miner	3-3
Oracle Client and Oracle Net	3-3

Connect the Spreadsheet Add-In.....	3-12
4 Users and Privileges for Data Mining	
Create a Database User.....	4-1
Grant Data Mining Privileges.....	4-2
System Privileges for Data Mining.....	4-3
Object Privileges for Mining Models.....	4-4
5 Working With Mining Model Schema Objects	
Obtaining Information from the Data Dictionary.....	5-1
Data Definition Language for Mining Models.....	5-3
Adding a Comment to a Mining Model.....	5-4
Auditing Mining Models.....	5-4
Exporting and Importing Mining Models.....	5-6
Importing From PMML.....	5-11
6 Upgrading Oracle Data Mining	
Data Mining Upgrade Overview.....	6-1
Upgrade Using Database Upgrade Assistant.....	6-2
Upgrade Using Export/Import.....	6-2
Post Upgrade Steps.....	6-3
Downgrade Data Mining.....	6-4
7 The Data Mining Sample Programs	
Installation and Setup.....	7-1
The PL/SQL Programs.....	7-6
The Java Programs.....	7-8
The Sample Data.....	7-11
A Verifying Your Data Mining Installation	
Connect to the Database.....	A-1
Create a Script or Execute the Commands Individually.....	A-1
Command Output.....	A-3

Index

Preface

This manual explains how to install Oracle Data Mining and prepare the database for data mining activities.

The preface contains these topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documentation](#)
- [Where to Find Database Tuning Documentation](#)
- [Conventions](#)

Audience

This guide is intended primarily for users who want to install, configure, and use Oracle Data Mining on Microsoft Windows for data mining exercises and demonstrations.

Note for DBAs: DBAs managing production databases that support Oracle Data Mining should follow standard administrative practices as described in the *Oracle Database Administrator's Guide*.

Building data mining models and batch scoring of mining models tend to put a DSS-like workload on the system, and single-row scoring tends to put an OLTP-like workload on the system.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documentation

The documentation set for Oracle Data Mining is part of the Oracle Database 11g Release 2 (11.2) Online Documentation Library. The Oracle Data Mining documentation set consists of the following documents:

- *Oracle Data Mining Concepts*
- *Oracle Data Mining Application Developer's Guide*
- *Oracle Data Mining Java API Reference* (javadoc)

For detailed information about the Oracle Data Mining PL/SQL interface, see *Oracle Database PL/SQL Packages and Types Reference*.

For detailed information about the SQL data mining functions, see *Oracle Database SQL Language Reference*.

For an introduction to Oracle application development, see *Oracle Database 2 Day Developer's Guide*.

Where to Find Database Tuning Documentation

The Oracle Data Mining documentation set does not address the topic of database tuning for data mining. The *Oracle Database Performance Tuning Guide* is the best source of information for tuning Oracle Database (including when the goal is to invoke mining build/score operations).

The memory tuning parameters `PGA_AGGREGATE_TARGET` and `SGA_TARGET` can easily have the most impact on data mining. The correct setting of PGA is very important for model building routines, as well as complex queries and batch scoring. From a Data Mining perspective, the SGA is generally less of a concern, except that real-time scoring is enabled by having the model loaded into the shared cursor in the SGA, so this still should be sized accordingly.

See Also:

Oracle Database Performance Tuning Guide for getting started

Oracle Database Performance Tuning Guide for more detail on memory configuration

In addition, if parallel execution is desired, then there are `INIT.ORA` parameters that control this functionality.

See Also: Chapter "Using Parallel Execution" in *Oracle Database VLDB and Partitioning Guide* for information about parallel execution tuning

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.
<i>ORACLE_HOME</i>	<i>ORACLE_HOME</i> represents the directory in which Oracle Universal Installer installs an Oracle product. <i>ORACLE_HOME</i> contains subdirectories for Oracle software executables and network files.

What's New in Oracle Data Mining Administration?

This section summarizes the new features in Oracle Data Mining.

It contains the following topics:

- [Oracle Database 11g Release 2 \(11.2.0.3\) Oracle Data Mining](#)
- [Oracle Database 11g Release 2 \(11.2.0.2\) New Features in Oracle Data Mining](#)
- [Oracle Database 11g Release 1 New Features in Oracle Data Mining](#)

Oracle Database 11g Release 2 (11.2.0.3) Oracle Data Mining

- The Oracle Data Mining Java API is deprecated in this release.

Note: Oracle recommends that you not use deprecated features in new applications. Support for deprecated features is for backward compatibility only

- Oracle Data Mining supports a new release of Oracle Data Miner. The earlier release, Oracle Data Miner Classic, is still available for download on OTN, but it is no longer under active development.

To download Oracle Data Miner 11g Release 2, go to:

<http://www.oracle.com/technetwork/database/options/odm/dataminerworkflow-168677.html>

To download Oracle Data Miner Classic, go to:

<http://www.oracle.com/technetwork/database/options/odm/downloads/odminer-097463.html>

Oracle Database 11g Release 2 (11.2.0.2) New Features in Oracle Data Mining

In Oracle Data Mining 11g Release 2 (11.2.0.2), you can import externally-created data mining models when they are presented as valid PMML documents. The model must be of type `RegressionModel`, either linear regression or binary logistic regression.

PMML is an XML-based standard for representing data mining models.

The `IMPORT_MODEL` procedure in the `DBMS_DATA_MINING` package is overloaded with new syntax to support PMML import. When invoked with this syntax, the `IMPORT_MODEL` procedure will accept a PMML document and translate the information into an Oracle Data Mining model. This includes creating and populating model tables as well as `SYS` model metadata.

External models imported in this way will be automatically enabled for Exadata scoring offload.

See Also:

Oracle Database PL/SQL Packages and Types Reference for information about `DBMS_DATA_MINING.IMPORT_MODEL`

Oracle Data Mining Concepts for information about scoring in Oracle Exadata

Oracle Database 11g Release 1 New Features in Oracle Data Mining

The following features were introduced in Oracle Data Mining 11g Release 1 (11.1):

- [No DMSYS Schema](#)
- [New Database Privilege Required for Creating Models](#)
- [Enhanced Security](#)
- [Mining Models in the Oracle Data Dictionary](#)
- [Features Not Available in Oracle Data Mining 11g Release 1](#)
- [Features Deprecated in Oracle Data Mining 11g Release 1](#)

No DMSYS Schema

Oracle Data Mining 11g is more tightly integrated with Oracle Database than it was in previous releases. Data Mining metadata and PL/SQL packages have been migrated from `DMSYS` to `SYS`. The `DMSYS` schema does not exist in Oracle Database 11g fresh installations.

New Database Privilege Required for Creating Models

In addition to the privileges needed for data mining in previous releases, the `CREATE MINING MODEL` privilege is required for creating models. This privilege should be added to any accounts being upgraded to 11g.

Additional system and object privileges are available for controlling mining activities in 11g.

See Also: [Chapter 4, "Users and Privileges for Data Mining"](#)

Enhanced Security

Security features of Oracle Data Mining are significantly enhanced in 11g. Improved security for data mining has several aspects:

- Previously, Oracle Data Mining used `DMSYS` as its own repository schema. This necessitated the granting of advanced database privileges to `DMSYS`, a non-system account. In 11g, the Oracle Data Mining metadata repository is in `SYS`, where it is accessible only to the system `DBA`.
- The SQL auditing system can be used to track operations on mining model objects.

- New system and object privileges for mining model objects are introduced in 11g.

See Also:

"Auditing Mining Models" on page 5-4
 Chapter 4, "Users and Privileges for Data Mining"

Mining Models in the Oracle Data Dictionary

New catalog views for Data Mining are introduced in Oracle Database 11g:

- ALL/DBA/USER_MINING_MODELS provides information about mining models.
- ALL/DBA/USER_MINING_MODEL_ATTRIBUTES provides information about data columns used to build a mining model.
- ALL/DBA/USER_MINING_MODEL_SETTINGS provides information about configuration settings for mining models.

The ALL/DBA/USER_OBJECTS catalog view now identifies mining models.

See Also: "Obtaining Information from the Data Dictionary" in Chapter 5

Features Not Available in Oracle Data Mining 11g Release 1

The following features are not supported in Oracle Data Mining 11g:

- DMSYS schema.
- Database Configuration Assistant (DBCA) is no longer required for adding Oracle Data Mining metadata to a database. Oracle Data Mining is part of Oracle Database metadata in Oracle 11g.
- Oracle Data Mining Scoring Engine, a separately-installable database option in Oracle 10g, is not available in Oracle 11g. All functionality of the Data Mining Scoring Engine option is offered in the Data Mining option.
- Basic Local Alignment Search Tool (BLAST).

Features Deprecated in Oracle Data Mining 11g Release 1

The following features are deprecated in Oracle Data Mining 11g:

- Adaptive Bayes Network classification algorithm
- DM_USER_MODELS view is replaced by data dictionary views
- Several PL/SQL procedures have been deprecated.
 - GET_DEFAULT_SETTINGS
 Replaced with data dictionary views: USER/ALL/DBA_MINING_MODEL_SETTINGS
 - GET_MODEL_SETTINGS
 Replaced with data dictionary views: USER/ALL/DBA_MINING_MODEL_SETTINGS
 - GET_MODEL_SIGNATURE
 Replaced with data dictionary views: USER/ALL/DBA_MINING_MODEL_ATTRIBUTES

Note: Oracle recommends that you do not use deprecated features in new applications. Support for deprecated features is for backward compatibility only.

See Also: *Oracle Database PL/SQL Packages and Types Reference*

Getting Started with Oracle Data Mining

The steps described in this chapter explain how to install Oracle Data Mining locally on your Windows PC or laptop and start up the client interfaces: Oracle Data Miner and Oracle Spreadsheet Add-In for Predictive Analytics.

This chapter provides summary information. Details are provided in subsequent chapters of this manual.

Note: The minimum requirements for a working installation of Oracle Data Mining are:

- An installation of Oracle Database Enterprise Edition
 - A database user ID that has data mining privileges (see [Chapter 4](#))
-
-
- [Step One: Install the Database](#)
 - [Step Two: Verify the Data Mining Installation](#)
 - [Step Three: Create a Data Mining User](#)
 - [Step Four: If You Want to Use Oracle Data Miner](#)
 - [Step Five: If You Want to Use the Spreadsheet Add-In](#)
 - [Step Six: If You Want to Use the Sample Programs](#)

Step One: Install the Database

Perform a basic installation of Oracle Database Enterprise Edition and create a starter database.

1. In Windows Control Panel, Administrative Tools, choose **Services**. Stop any Oracle services that may be running on your computer.
2. From the Oracle Database installation directory, run `SETUP.EXE` to start Oracle Universal Installer. Follow the instructions as specified on the initial pages, then:
 - On the Select Installation Options page, choose **Create and Configure a Database**.
 - On the System Class page, choose **Desktop Class**.
 - On the Typical Install Configuration page, specify the installation directories, the database name, and the database system passwords.

3. The Installer performs prerequisite checks, displays summary information about the installation, and copies the Oracle executables and the starter database files to your computer.
4. The Installer configures Oracle Net to enable client connections.
5. The Installer starts Oracle Database Configuration Assistant to create the starter database.
6. Database Configuration Assistant displays summary information about the starter database. If you wish to use Oracle sample data or the Data Mining sample programs, unlock the `SH` account.
7. Exit the Installer. The starter database is now running, and Oracle Net is ready to accept client connections.
8. If necessary, restart the Oracle services that you stopped in step 1.

See: ["Install Oracle Database"](#) on page 2-1 for more detailed instructions

Step Two: Verify the Data Mining Installation

Once you have installed Oracle Database and created a starter database, you may wish to verify that Oracle Data Mining is functioning properly before proceeding further.

In Appendix A, you will find a series of SQL and PL/SQL commands that perform basic data mining operations. You can type these commands into SQL*Plus or SQL Developer to make sure that the database is enabled for data mining.

See: [Appendix A, "Verifying Your Data Mining Installation"](#)

Step Three: Create a Data Mining User

Create a database user with data mining privileges.

1. Log in to SQL*Plus or SQL Developer as the system user.
2. Execute the following `CREATE USER` statement. Specify a user name for `dmuser` and password for `dmpassword`.

```
CREATE USER dmuser IDENTIFIED BY dmpassword
        DEFAULT TABLESPACE users
        TEMPORARY TABLESPACE temp
        QUOTA UNLIMITED ON users;
Commit;
```

3. Execute these statements to grant data mining privileges to the user.

```
GRANT CREATE JOB TO dmuser;
GRANT CREATE MINING MODEL TO dmuser;           -- required for creating models
GRANT CREATE PROCEDURE TO dmuser;
GRANT CREATE SEQUENCE TO dmuser;
GRANT CREATE SESSION TO dmuser;
GRANT CREATE SYNONYM TO dmuser;
GRANT CREATE TABLE TO dmuser;
GRANT CREATE TYPE TO dmuser;
GRANT CREATE VIEW TO dmuser;
GRANT EXECUTE ON ctxsys.ctx_ddl TO dmuser;    -- required for text mining

GRANT SELECT ON data TO dmuser;              -- required for mining data that is
                                                not in your schema
```

4. If the user will need to import or export data mining models, grant this additional privilege.

```
GRANT CREATE ANY DIRECTORY TO dmuser;
```

For import/export operations, the user must create a directory object. The user may also require additional privileges. See "Exporting and Importing Mining Models" on page 5-6 for details.

See: Chapter 4, "Users and Privileges for Data Mining" for more detailed instructions

Step Four: If You Want to Use Oracle Data Miner

Oracle Data Miner, an extension to Oracle SQL Developer 3.0, is a graphical user interface to Oracle Data Mining. Oracle Data Miner uses a workflow paradigm to perform data mining tasks.

You can use Oracle Data Miner to explore data, build and evaluate multiple mining models, and apply the models to new data. By building workflows, you can capture and document the methodology you use to perform a range of mining tasks. You can save and share workflows.

To install and launch Oracle Data Miner:

1. Go to the following page on the Oracle Technology Network:

<http://www.oracle.com/technetwork/database/options/odm/dataminerworkflow-168677.html>
2. Follow the installation instructions provided on the page. The basic steps can be summarized as:
 - a. Install Oracle Database (also described in [Step One: Install the Database](#) earlier in this chapter).
 - b. Install Oracle SQL Developer 3.0, which includes Oracle Data Miner 11gR2.
 - c. Install the Oracle Data Miner repository.
3. For further instructions, use the Online Help provided by SQL Developer.
4. For additional assistance, use the Oracle By Example tutorials. Links to the tutorials are available with the installation instructions at the URL provided in Step 1.

Note: Oracle Data Miner Classic, the previous release of Oracle Data Mining, is still available for download on the Oracle Technology Network.

<http://www.oracle.com/technetwork/database/options/odm/downloads/index.html>

Oracle Data Miner Classic is no longer under active development.

Step Five: If You Want to Use the Spreadsheet Add-In

The Oracle Spreadsheet Add-In for Predictive Analytics lets you perform certain limited data mining operations in an Excel spreadsheet.

To install and launch the Spreadsheet Add-In:

1. Install Oracle Client and create a Net Service Name as described in "[Oracle Client and Oracle Net](#)" on page 3-3.
2. Download the Spreadsheet Add-In from the Oracle Technology Network.
<http://www.oracle.com/technetwork/database/options/odm/odm-pred-analytics-addin-092973.html>
3. Open the zip file and extract the contents to the Microsoft Office Library, typically:
C:\Program Files\Microsoft Office\Office\Library
4. Open Excel and click **Tools > Add-Ins**. Select **Oracle Predictive Analytics** from the Add-Ins dialog box. The OraclePA menu is added to the Excel toolbar.
5. From the Add-In menu in Excel, choose **Connect**.
6. In the Connect (Oracle Database) dialog:
 - Select the Net Service Name that you created.
 - Provide the user name and password of the data mining user.Click **Connect** to launch the Spreadsheet Add-In.

See: "[Optionally Install the Spreadsheet Add-In](#)" on page 2-11 for more detailed instructions

Step Six: If You Want to Use the Sample Programs

Follow these steps to install and configure the sample Data Mining programs:

1. Install Oracle Database Examples as described in "[Optionally Install Oracle Database Examples](#)" on page 2-10.
2. The sample programs are copied to the RDBMS\demo subdirectory of the Oracle home directory. You can find the PL/SQL programs by searching for dm*.sql.
3. Start SQL*Plus and connect to the database as the system user. Run the dmshgrants script. Specify the full path to the Oracle home directory and the name of the data mining user.

```
@ ORACLE_HOME\RDBMS\demo\dmshgrants dmuser
```

4. Connect to the database as the Data Mining user. Run the dmsh script.

```
CONNECT dmuser  
@ ORACLE_HOME\RDBMS\demo\dmsh  
COMMIT;
```

5. Once you have completed steps 1 through 4, you can run the PL/SQL programs.

See: [Chapter 7, "The Data Mining Sample Programs"](#) for more detailed instructions

Installing Oracle Data Mining

This chapter is intended for anyone who wishes to install Oracle Data Mining and its client interfaces on a personal computer or laptop running Microsoft Windows.

Note: For production installations of Oracle Database with Oracle Data Mining, refer to the installation instructions for your platform.

See the "Installing and Upgrading" page of the Oracle Database Online Documentation Library:

<http://www.oracle.com/pls/db112/homepage>

This chapter contains the following sections:

- [Install Oracle Database](#)
- [Verify the Data Mining Installation](#)
- [Enable or Disable the Data Mining Option](#)
- [Optionally Install Oracle Database Examples](#)
- [Optionally Install Oracle Data Miner](#)
- [Optionally Install the Spreadsheet Add-In](#)

Install Oracle Database

Oracle Data Mining is an option to Oracle Database Enterprise Edition. To perform data mining activities, you must be able to log on to an Oracle database, and your user ID must have the database privileges described in [Chapter 4](#).

The instructions in this section explain how to perform a basic installation of Oracle Database Enterprise Edition on Microsoft Windows. Instructions for creating a starter database are included.

Important: These instructions are not intended as a replacement for Oracle Database installation documentation. If you have questions, if you encounter problems during the installation, or if you already have Oracle components installed on your PC, refer to:

- *Oracle Database Installation Guide for Microsoft Windows*

For a high level introduction to Oracle Database installation and configuration, see:

- *Oracle Database 2 Day DBA Online Help*

Additional documentation is available on the "Installing and Upgrading" page of the Oracle Database 11g Online Documentation Library:

<http://www.oracle.com/pls/db112/homepage>

1. Before you begin the installation, ensure that your computer meets the system requirements described in *Oracle Database Installation Guide for Microsoft Windows*.

Note: If your computer uses Dynamic Host Configuration Protocol (DHCP), you may need to install a loopback adapter to assign a local IP address to your computer.

See *Oracle Database Installation Guide for Microsoft Windows* for instructions.

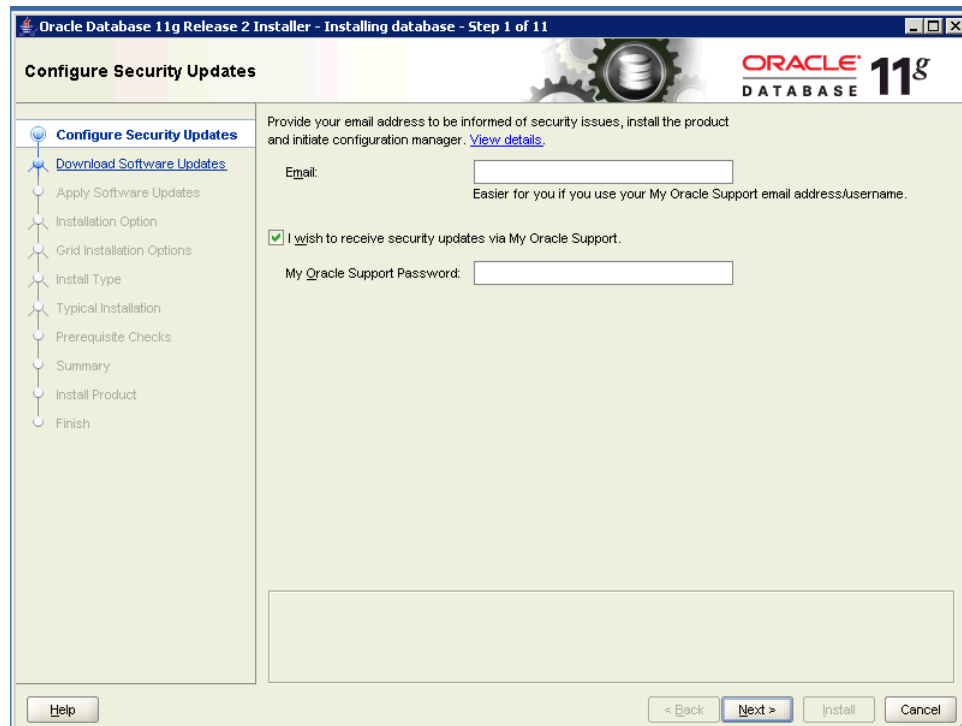
2. Log on to your computer as a member of the Administrator's group. If you are unsure if your user ID is in the Administrator's group, check **User Accounts** in Windows Control Panel.

3. Stop any Oracle services that may be running on your computer.

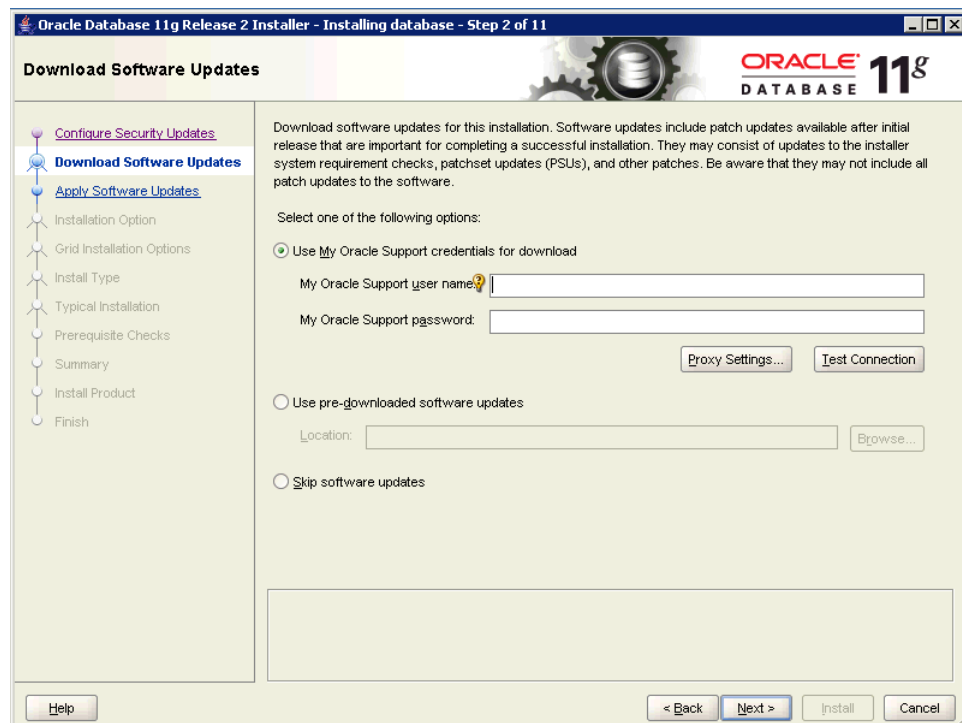
In Windows Control Panel, choose **Administrative Tools**, then **Services**. Find the service names that start with "Oracle". Choose **Stop** for each one.

Note: To stop an Oracle database, first stop the database service (`OracleServiceservice_name`), then stop the database listener (`OracleOraDb11g_homeTNSListener`).

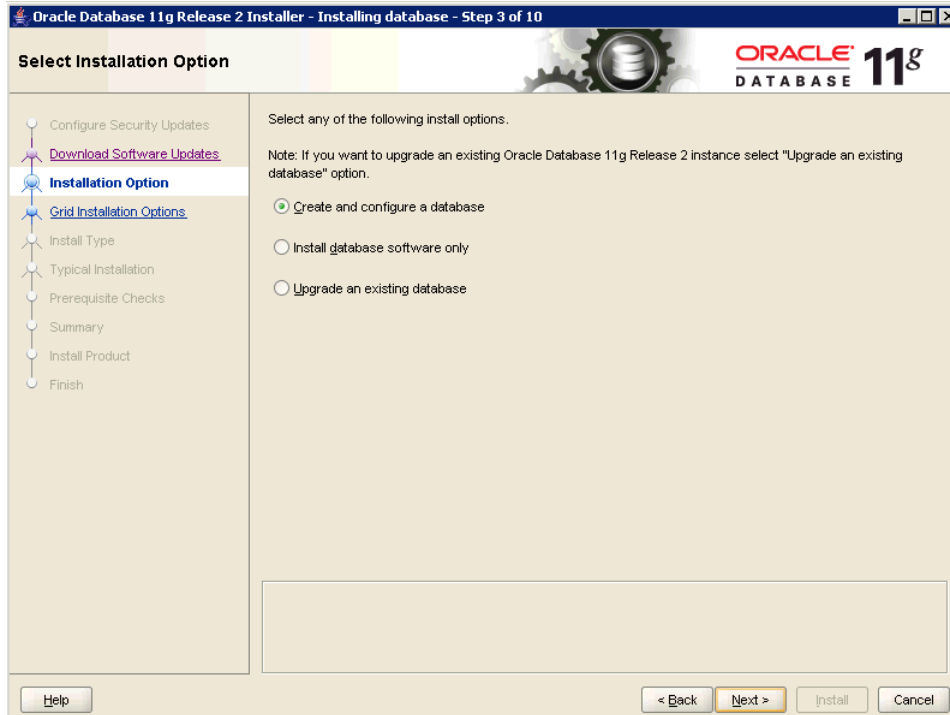
4. To start the installation, run `SETUP.EXE` from the Database installation directory. Oracle Universal Installer opens and displays the Configure Security Options page. Type your My Oracle Support email address. Click **Next**.



5. In the Download Software Updates box, type your My Oracle Support credentials if you want to automatically download software updates. Click Next.



6. On the Select Installation Option page, choose **Create and configure a database**.



Note: If you choose **Install database software only** instead of **Create and configure a database**, you may choose which database options you want to include in the installation. If you do not choose the Data Mining option during installation, you can enable it later. See ["Enable or Disable the Data Mining Option"](#) on page 2-9 for instructions.

7. On the System Class page, choose **Desktop Class** then **Next**.



8. On the Typical Install Configuration page, specify the following information about the installation:
 - **Oracle Base** — Specify a base directory for Oracle products on this computer. If this directory does not exist, the Installer will create it.

Oracle base is the root of the Oracle directory tree. It may contain multiple Oracle home directories. By convention, Oracle base is referred to as *ORACLE_BASE*.
 - **Software Location** — The Installer provides a Software Location under *ORACLE_BASE*. This will be the home directory for this installation of Oracle Database. If this directory does not exist, the Installer will create it. If the directory does exist, make sure that it is empty.
 - **Database File Location** — The Installer provides a Database File Location under *ORACLE_BASE*. This directory will hold the data files for the starter database. If this directory does not exist, the Installer will create it. If the directory does exist, make sure that it is empty.
 - **Database Edition** — Choose **Enterprise Edition**.
 - **Global Database Name** — Specify a unique name for the starter database.

The Global Database name is the fully-qualified name of the database: for example, *dbname.us.oracle.com*. Specify a unique name for *dbname* that is not more than eight characters.
 - **Administrative Password** — Specify a password for the database system accounts *SYS* and *SYSTEM*. The password should have at least eight characters and include both alphabetic and numeric characters.

You will have the opportunity to change the passwords at a later time.
 - **Confirm Password** — Re-enter the administrative password.

Oracle Database 11g Release 2 Installer - Installing database - Step 5 of 9

Typical Install Configuration

Perform full Database installation with basic configuration.

Oracle base:

Software location:

Database file location:

Database edition:

Character Set:

Global database name:

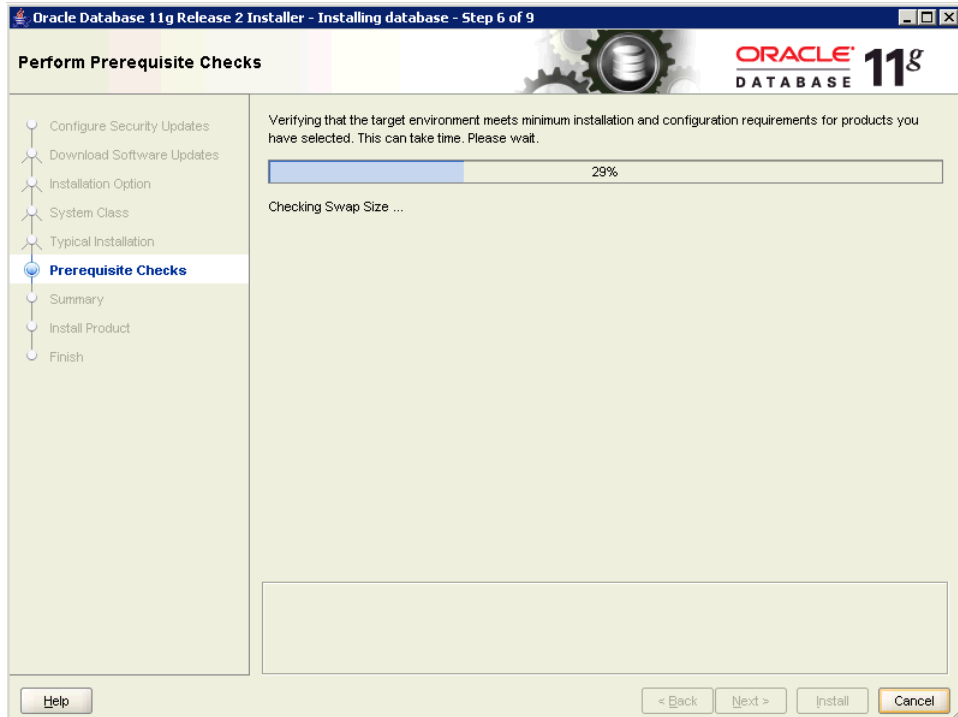
Administrative password:

Confirm Password:

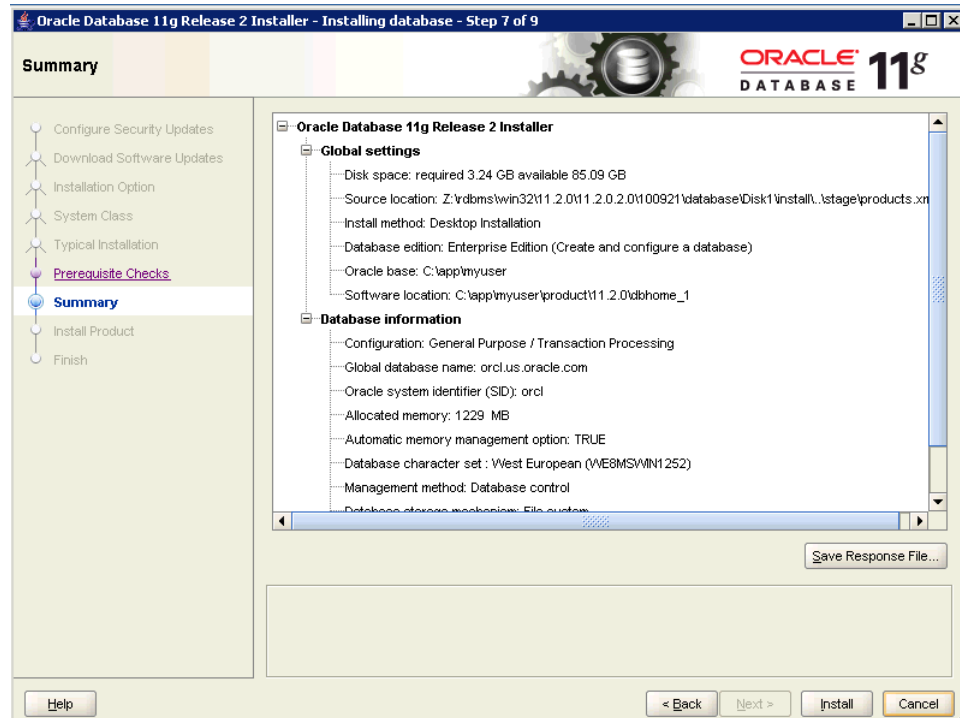
Note: To obtain information about any of the items on this page, drag the cursor from right to left over the item. A tooltip that describes the item will appear.

After you have provided the information, click **Next**.

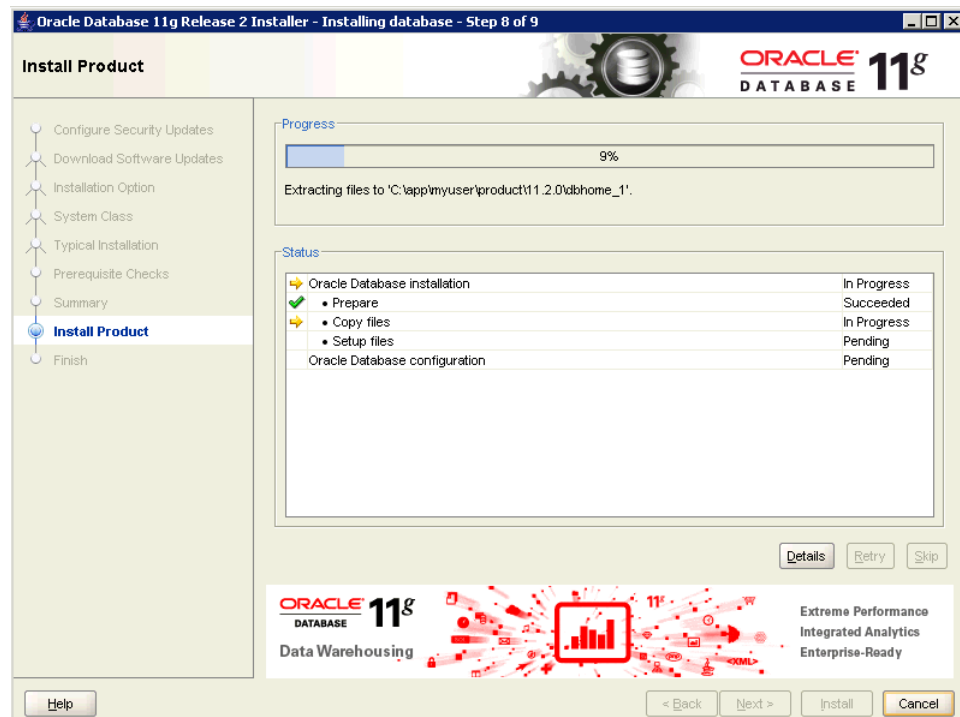
9. The Installer analyzes your computer to determine if you have the resources needed for the installation. If the prerequisite checks are successful, click **Next**. If not, click **Fix and Check Again**.



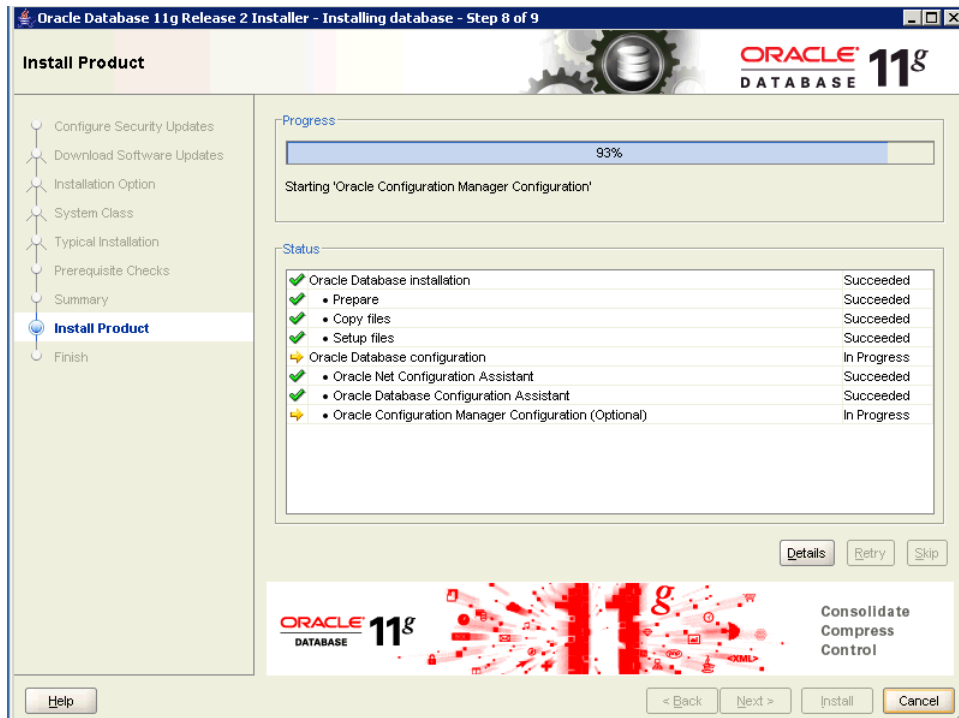
10. On the Summary page, click **Install**.



11. The Install Product page displays the progress of the installation.



12. Once the software has been copied to your computer, the Database Configuration Assistant creates and starts the starter database.

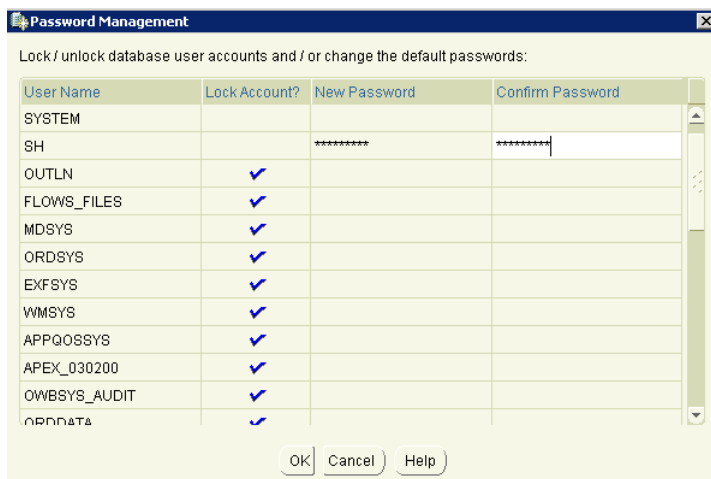


13. The Database Configuration Assistant displays summary information about the starter database.

At the bottom of the page is a note about passwords for the starter database. Choose Password Management if you wish to reset the administrative passwords or unlock any other accounts that are available in the database. By default the SYS and SYSTEM accounts are unlocked.

Note: If you wish to use Oracle sample data or the Oracle Data Mining sample programs, unlock the SH account.

See [Chapter 7](#) for information about the sample programs.



Click **OK** to return to the Database Configuration Assistant page. Click **OK** to return to Oracle Universal Installer.

14. On the Finish page, click **Close**.
15. Now the database is running and ready to accept connections.
16. If you stopped any Oracle services before running Oracle Universal Installer, you can restart them at this time.

In Windows Control Panel, choose **Administrative Tools**, then **Services**. Find the service names that start with "Oracle". Choose **Start** for each one.

Note: To restart an Oracle database, first restart the listener (OracleOraDb11g_homeTNSListener), then restart the database service (OracleServiceservice_name).

Verify the Data Mining Installation

If you wish to verify that Oracle Data Mining is present and functioning properly in the database, you can execute a few simple commands at the SQL command line. A set of commands is provided for this purpose in [Appendix A](#).

You can perform this verification without installing or downloading any additional software.

See: [Appendix A, "Verifying Your Data Mining Installation"](#)

Enable or Disable the Data Mining Option

The Data Mining option is enabled by default during installation of Oracle Database. However, you can choose to disable Data Mining and other database options when you install the database software. (See step 6 of [Install Oracle Database](#).)

If you wish to enable or disable a database option for an Oracle home, you can use the command-line utility `chopt`.

```
chopt [ enable | disable ] db_option_name
```

Set `db_option_name` to `dm` for the Data Mining option.

To enable the Data Mining option:

1. Shut down the database.

```
srvctl stop database -d db_name
```

2. Stop the database service, `OracleServiceSID`, using the Services program in Control Panel.

3. Execute these commands.

```
cd ORACLE_HOME/bin
chopt enable dm
```

4. Start the database service, `Oracle ServiceSID`, using the Services program in Control Panel.

5. Start up the database.

```
srvctl start database -d db_name
```

To disable the Data Mining option:

1. Shut down the database.

```
srvctl stop database -d db_name
```

2. Stop the database service, `OracleServiceSID`, using the Services program in Control Panel.

3. Execute these commands.

```
cd ORACLE_HOME/bin
chopt disable dm
```

4. Start the database service, `Oracle ServiceSID`, using the Services program in Control Panel.

5. Start up the database.

```
srvctl start database -d db_name
```

See Also: *Oracle Database Installation Guide for Microsoft Windows* for more information about `chopt`

Optionally Install Oracle Database Examples

The Data Mining sample programs are installed with Oracle Database Examples. The installation process copies the Data Mining sample programs, along with examples and demonstrations of other database features, to the `\RDBMS\demo` subdirectory of Oracle home.

Note: You only need to install Oracle Database Examples if you wish to use the sample programs. If you do not wish to install the sample programs, you can skip this section.

See Also:

Oracle Database Examples Installation Guide for complete installation instructions

[Chapter 7](#) for information about the Oracle Data Mining sample programs

To install Database Examples, perform these steps:

1. Ensure that your computer meets the system requirements described in *Oracle Database Examples Installation Guide*.
2. Stop any Oracle services that may be running on your computer.

In Windows Control Panel, choose **Administrative Tools**, then **Services**. Find the service names that start with "Oracle". Choose **Stop** for each one.

Note: To stop an Oracle database, first stop the database service (`OracleServiceservice_name`), then stop the database listener (`OracleOraDb11g_homeTNSListener`).

3. To start the installation, go to the Examples installation directory and run `SETUP.EXE`.

Oracle Universal Installer displays the **Specify Install Location** page. Specify the Oracle base and Oracle home provided in the database software installation (step 8).

Do not assume that the directories displayed by the Installer are correct.

Click **Next**.

Optionally Install Oracle Data Miner

Instructions for downloading and installing Oracle Data Miner are available at <http://www.oracle.com/technetwork/database/options/odm/dataminerworkflow-168677.html>. Additional details are available in the online help for Oracle Data Miner.

Note: Installation of Oracle Data Miner is not required. Oracle Data Mining supports a comprehensive SQL-based API and a Spreadsheet Add-In that supports limited data mining capabilities in Microsoft Excel.

See *Oracle Data Mining Application Developer's Guide* for an introduction to the SQL and PL/SQL interfaces to Oracle Data Mining.

See Also: [To install and launch Oracle Data Miner:](#) in [Chapter 1](#)

Optionally Install the Spreadsheet Add-In

The Oracle Spreadsheet Add-In for Predictive Analytics adds predictive analytics features to Microsoft Excel. Using simple "one click" data mining for PREDICT, EXPLAIN, PROFILE, and EXPORT/IMPORT operations, you can mine data stored in Excel or in an Oracle database.

Note: Installation of the Spreadsheet Add-In is not required. Oracle Data Mining supports a comprehensive SQL-based API as well as a full-featured graphical interface, Oracle Data Miner.

See *Oracle Data Mining Application Developer's Guide* for an introduction to the SQL and PL/SQL interfaces to Oracle Data Mining.

See Also: *Oracle Data Mining Concepts* for information about Oracle predictive analytics

Spreadsheet Add-In: Requirements

The Spreadsheet Add-In requires the following:

- Access to an instance of Oracle Database installed with the Data Mining option.
- A user ID with the privileges described in [Chapter 4](#).
- Microsoft Excel 2000 or later.
- Oracle Objects for OLE and Oracle Net Configuration Assistant.

To obtain these components, Oracle Client must be present on the computer where Excel and the Spreadsheet Add-In are installed. For Oracle Client installation instructions, see [Chapter 3](#).

- An Oracle Net Service Name for the database connection. Instructions for creating an Oracle Net Service Name are provided in [Chapter 3](#).

Spreadsheet Add-In: Install Steps

To install Oracle Spreadsheet Add-In for Predictive Analytics:

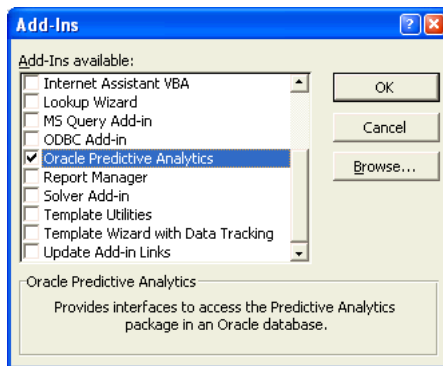
1. Ensure that your system meets the requirements described in "[Spreadsheet Add-In: Requirements](#)" on page 2-11.
2. Download the Spreadsheet Add-In from the Oracle Data Mining Web site at <http://www.oracle.com/technetwork/database/options/odm/odm-pred-analytics-addin-092973.html>.

3. Open the zip file and extract the files to the Microsoft Office Library directory, which has a path such as this one:

C:\Program Files\Microsoft Office\Office\Library

The zip file contains the following files:

- Predictive_Analytics.xla if you have Microsoft Excel 2000, 2002, or 2003
 - Predictive_Analytics_2007.xla if you have Microsoft Excel 2007
 - PAhelp.chm, the help file for the Spreadsheet Add-In
 - A readme (which you can copy to the directory of your choosing).
4. Open Excel and click **Tools > Add-Ins**. Select **Oracle Predictive Analytics** from the Add-Ins dialog box.



Note: If the Spreadsheet Add-In was previously installed on your computer, uncheck **Oracle Predictive Analytics** and click **OK** to close the Add-Ins dialog. Then reopen the dialog and check the **Oracle Predictive Analytics** box.

5. The OraclePA menu is added to the Excel toolbar.
6. Although the Add-In may use data stored in Excel or in Oracle Database, a connection to Oracle Database is required. For instructions, see "[Connect the Spreadsheet Add-In](#)" on page 3-12.

Start Data Mining in Excel

Although the Spreadsheet Add-In may use data stored in Excel or in Oracle Database, a connection to Oracle Database is always required. Once you have created the connection, you can use the `PREDICT`, `EXPLAIN`, `PROFILE`, and `export/import` operations provided by the Add-In.

To create the connection and install the Add-In in Excel, follow the instructions in "[Connect the Spreadsheet Add-In](#)" on page 3-12.

Note: The first time you use the Add-In, you will be prompted to provide the path to the Java executable. Specify the full path of `jdk\bin\java.exe` under Oracle home.

For the installation described in "[Install Oracle Database](#)" on page 2-1, the path would be:

```
C:\app\myuser\product\11.2.0\dbhome_1\jdk\bin\java.exe
```

Connecting to an Oracle Database

This chapter explains how to connect to a database to use Oracle Data Mining. It contains the following sections:

- [Connect Application Development Tools](#)
- [Connect SQL*Plus](#)
- [Connect SQL Developer](#)
- [Oracle Client and Oracle Net](#)
- [Connect the Spreadsheet Add-In](#)

Connect Application Development Tools

SQL*Plus and SQL Developer and other client tools are installed with Oracle Database. SQL*Plus is a command line tool for transmitting SQL and PL/SQL to the database. SQL Developer is a graphical development environment for SQL and PL/SQL.

You can use the Data Mining PL/SQL API in either SQL*Plus or SQL Developer. Within SQL Developer you can use both the Data Miner graphical tool and the Data Mining PL/SQL API. For information about the PL/SQL API, see *Oracle Data Mining Application Developer's Guide*.

If you have installed Oracle Database locally on your PC, you can follow these steps to start SQL*Plus or SQL Developer:

1. From the Windows Start menu, select **Oracle – OraDb11g_home*n***.

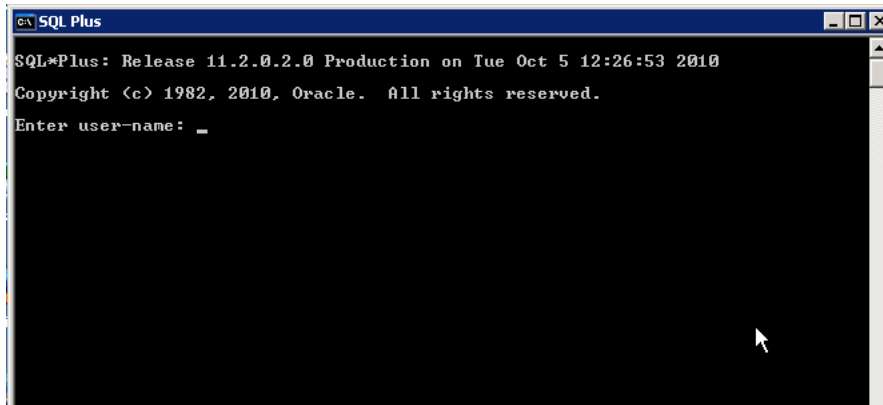
Note: If there are n database installations, they will appear as Oracle - OraDb11g_home1, Oracle - OraDb11g_home2, Oracle - OraDb11g_homen in the Start menu.

Each of these refers to a separate `ORACLE_HOME`. Be sure to select the appropriate one.

2. Choose **Application Development**.
3. Choose the name of the application.

Connect SQL*Plus

When SQL*Plus starts up it is already connected to the local database. Enter the name of a user ID that has data mining privileges, as described in [Chapter 4](#). You will be prompted to enter the password.



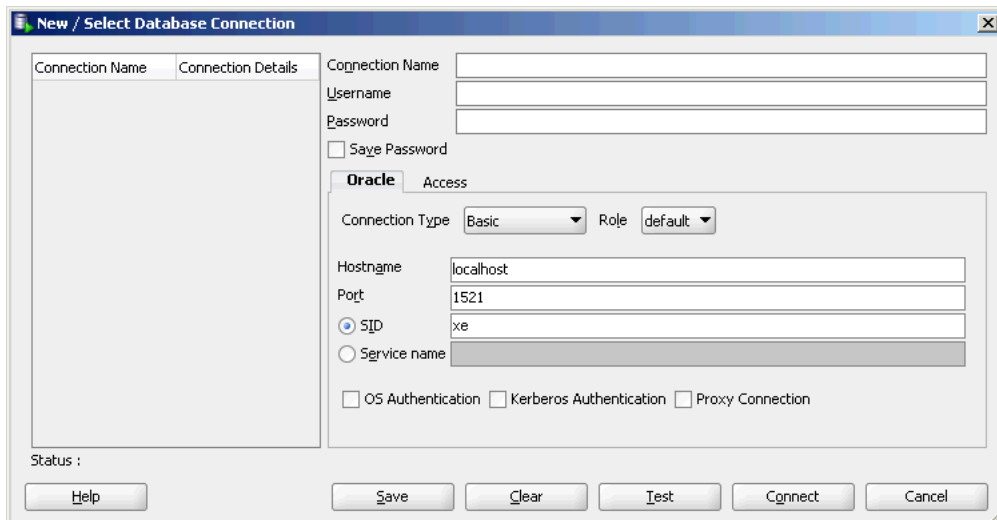
Connect SQL Developer

The first time you run SQL Developer, you may be prompted to provide the path to the Java executable. If so, specify the full path of `jdk\bin\java.exe` under the Oracle home directory for the database. For the installation described in [Chapter 2](#), the path would be:

```
C:\app\myuser\product\11.2.0\dbhome_1\jdk\bin\java.exe
```

To define a connection to the local database:

1. Right click **Connections** and choose **New**.
2. The New/Select Database Connection dialog appears.



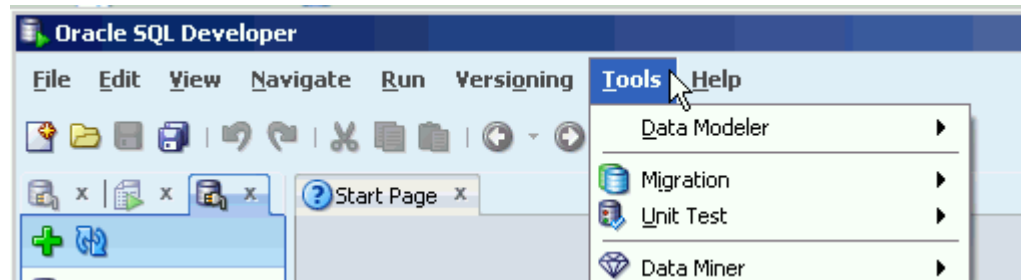
- For **Connection Name**, specify a meaningful name that you will remember.
- For **Username/Password**, provide a user name and password for the connection. The user must have data mining privileges as described in [Chapter 4](#).
- For **Hostname**, specify `localhost` to indicate that the database is hosted on the local computer.
- For **Port**, specify the port where the database will listen to requests from the client. The default port is 1521.

- For SID (Service Identifier), specify the simple database name. This is the first component of the global database name described in the installation instructions in [Chapter 2](#).

Click **Connect** to connect to the database using the specified credentials.

Connect Oracle Data Miner

Oracle Data Miner is a choice on the Tools menu of SQL Developer.



Oracle Data Miner requires an administrative (SYS) connection for installation. A separate connection to the database is required for each user that will use the tool. Instructions for creating these connections are provided in the Oracle Data Miner Help. For installation and configuration instructions, go to <http://www.oracle.com/technetwork/database/options/odm/dataminerworkflow-168677.html>.

Oracle Client and Oracle Net

To connect to an Oracle Database, a client application must identify the database and present appropriate credentials. The connection mechanism depends on the type of client and whether it is running locally or remotely. Some connections require the installation of Oracle Client software.

There are several methods for specifying login credentials. For example, a client may present login credentials at the time of the connection, as shown below. The system will prompt for a password and establish a connection.

```
SQL>CONNECT username @ "hostname[:port][/databasename]"
```

You may also identify a connection with a predefined Net Service Name.

Oracle Spreadsheet Add-In for Predictive Analytics, whether running locally or remotely, requires the installation of Oracle Client and the creation of a Net Service Name.

See Also: ["Create a Net Service Name"](#) on page 3-8

Note: SQL Developer and Oracle Data Miner do not require the installation of Oracle Client.

For details about Oracle Data Miner installation and configuration, see Oracle Data Miner 11g Release 2 "Getting Started" on the Oracle Technology Network:

<http://www.oracle.com/technetwork/database/options/odm/dataminerworkflow-168677.html>

Additional details are available in the online help for Oracle Data Miner.

Install Oracle Client

You must install Oracle Client if:

- You wish to use Oracle Spreadsheet Add-In for Predictive Analytics
- You do not have a local database and you wish to use SQL*Plus and other tools for connecting to a remote Oracle database.

Use the following steps to install Oracle Client on a Windows platform:

Note: The following instructions are not intended as a replacement for Oracle Client and Net Management documentation. For additional information, refer to:

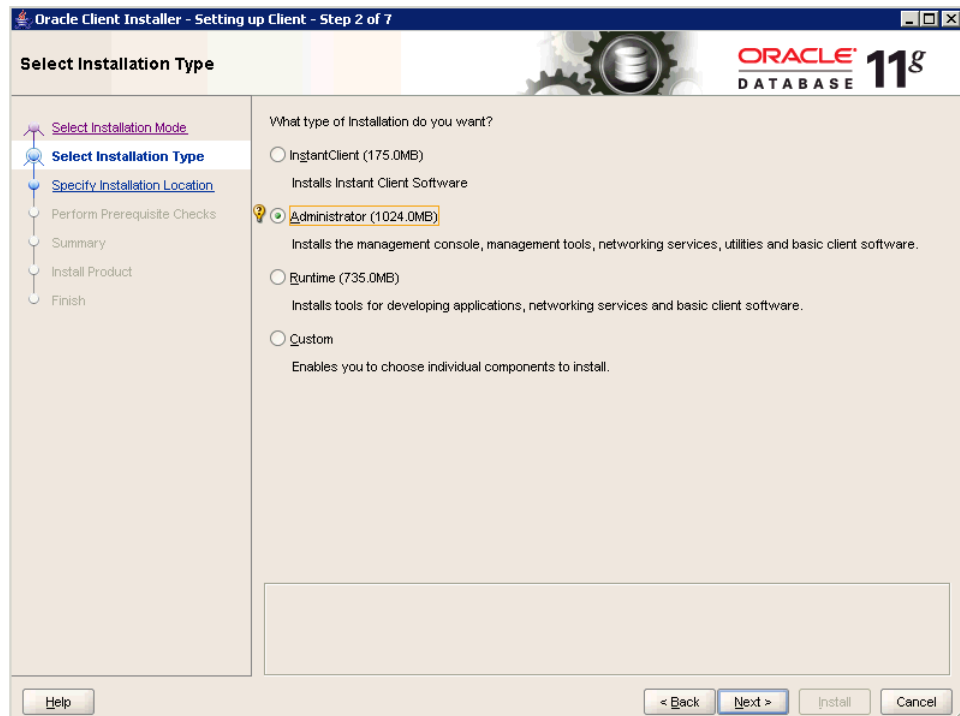
- *Oracle Database 2 Day DBA Online Help*
 - *Oracle Database Client Installation Guide for Microsoft Windows*
 - *Oracle Database Net Services Administrator's Guide*
-
-

1. Ensure that your computer meets the system requirements described in *Oracle Database Client Installation Guide for Microsoft Windows*.
2. Stop any Oracle services that may be running on your computer.

In Windows Control Panel, choose **Administrative Tools**, then **Services**. Find the service names that start with "Oracle". Choose **Stop** for each one.

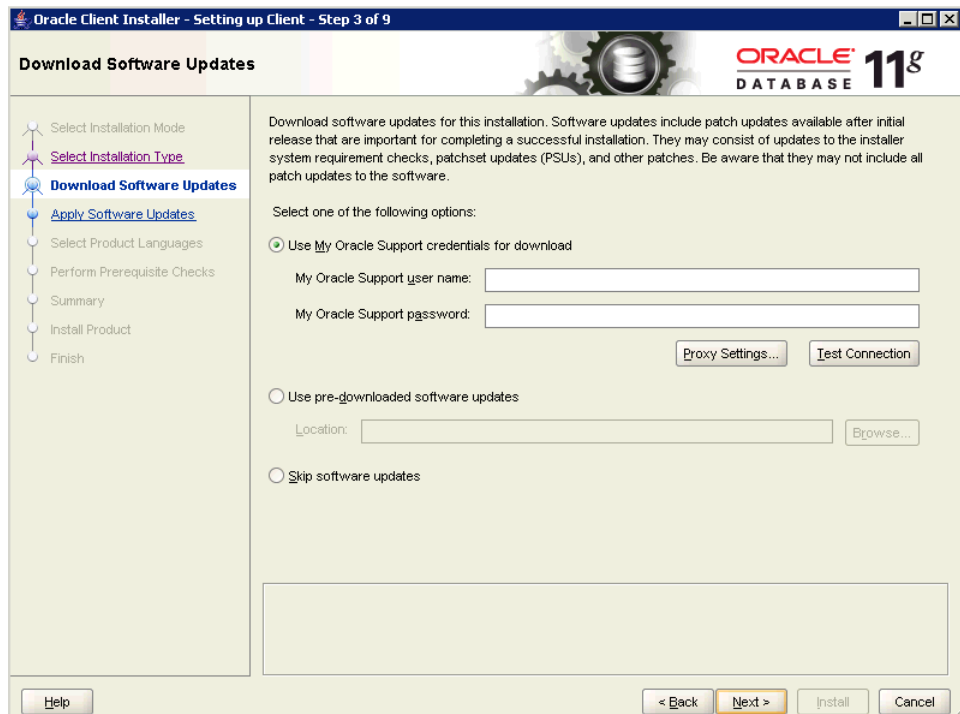
Note: To stop an Oracle database, first stop the database service (OracleServiceservice_name), then stop the database listener (OracleOraDb11g_homeTNSListener).

3. Log on to your computer as a member of the Administrator's group. If you are unsure if your user ID is in the Administrator's group, check **User Accounts** in Windows Control Panel.
4. To start the installation, go to the Client installation directory and run `SETUP.EXE` to start Oracle Universal Installer.
5. On the Select Installation Type page, choose **Administrator**.

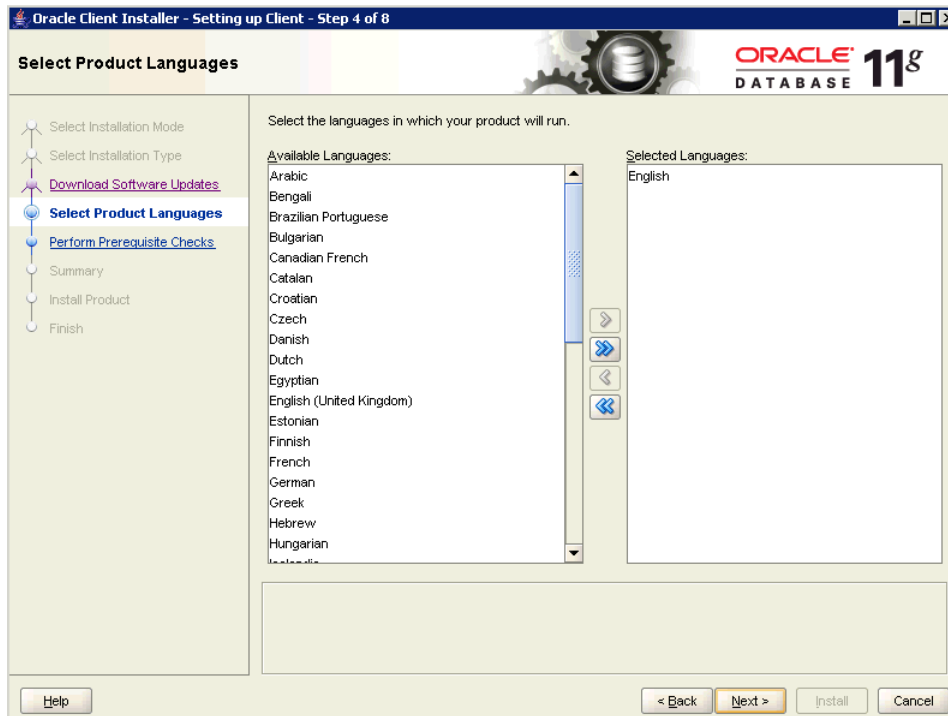


Click **Next**.

- In the Download Software Updates page, provide your My Oracle Support credentials if you want to automatically download software updates. Click **Next**.



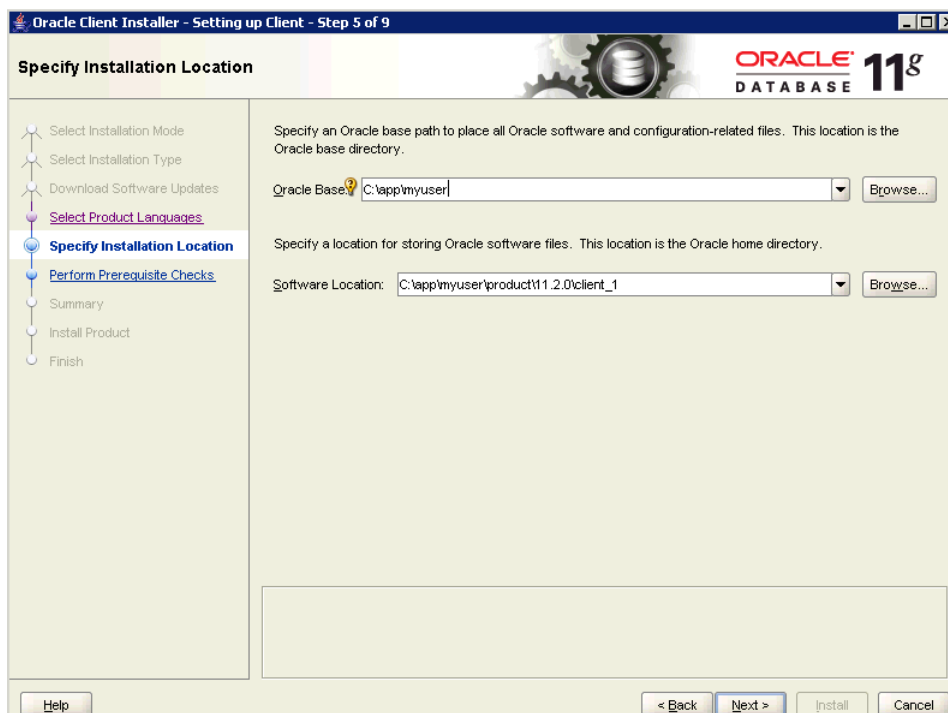
- On the Select Product Languages page, select the language for the Oracle Client user interface.



Click **Next**.

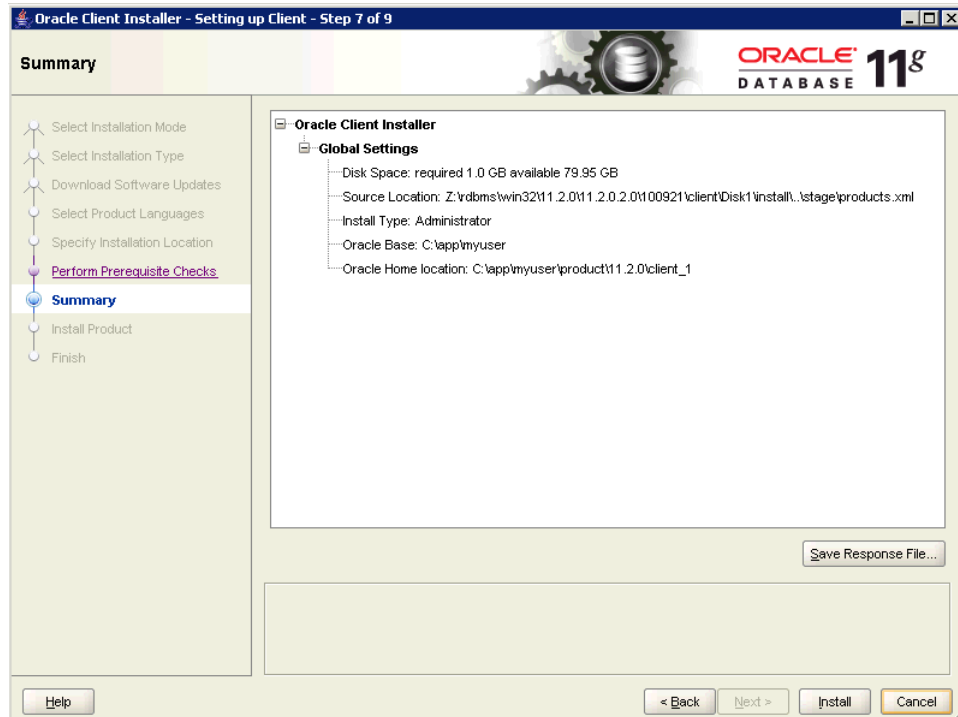
8. On the Specify Install Location page, provide the path of the Base directory in which you installed Oracle Database. (See "Install Oracle Database" step 8.)

The Installer provides a Software Location under Oracle Base. This will be the home directory of the Oracle client that you are installing (Oracle home). If this directory does not exist, the Installer will create it. If the directory does exist, make sure that it is empty.



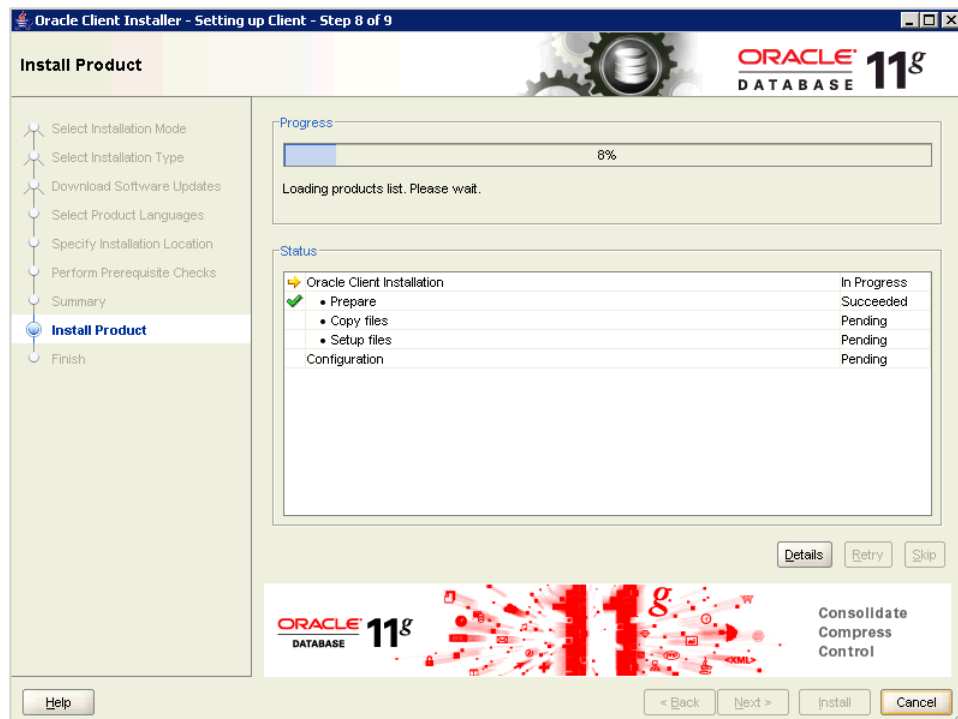
Click **Next**.

9. The Installer performs prerequisite checks, then displays the Summary page.



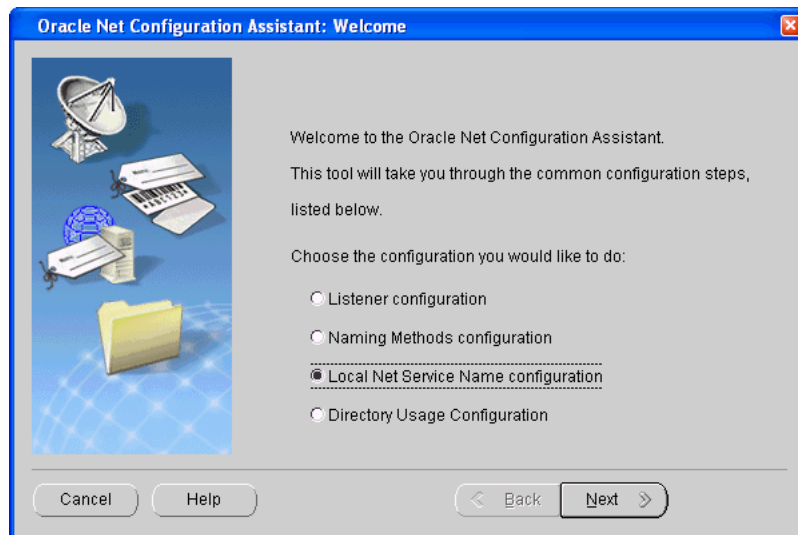
Click **Install** to complete the set up process and begin the installation.

10. On the Install Product page, the Installer displays a progress bar.



When the installation is successfully completed, click **Close**.

11. Oracle Net Configuration Assistant starts and displays the Welcome page.



Choose **Local Net Service Name Configuration**, then click **Next**.

Oracle Net Configuration Assistant determines that Oracle Client can use the Easy Connect naming method, as described in "[Oracle Client and Oracle Net](#)" on page 3-3. A Net Service Name is not needed. Simply click **Next**.

12. Oracle Net Configuration Assistant displays summary information about the connection configuration.

Click the **Password Management** button at the bottom of the page to unlock any schemas that you plan. If you wish, you can also reset the system passwords. If you will use the Oracle Data Mining sample programs and you did not unlock the SH schema when you installed the database, you should do so now. (See "[Install Oracle Database](#)".)

Click **Exit** to exit Oracle Net Configuration Assistant, then click **Finish** to complete the installation of Oracle Client.

13. On the End of Installation page, click **Exit** to exit Oracle Universal Installer.
14. If you stopped any Oracle services before running Oracle Universal Installer, you can restart them at this time.

In Windows Control Panel, choose **Administrative Tools**, then **Services**. Find the service names that start with "Oracle". Choose **Start** for each one.

Note: To restart an Oracle database, first restart the listener (OracleOraDb11g_homeTNSListener), then restart the database service (OracleServiceservice_name).

Create a Net Service Name

The instructions in this section explain how to create a Net Service Name. Follow these instructions if you wish to use Oracle Spreadsheet Add-In for Predictive Analytics.

Note: The following instructions are not intended as a replacement for Oracle Client and Net Management documentation. For additional information, refer to:

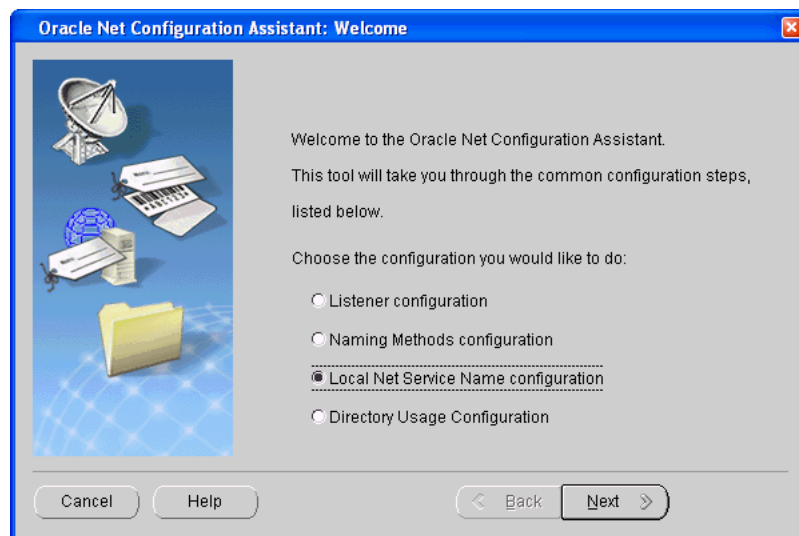
- *Oracle Database 2 Day DBA Online Help*
 - *Oracle Database Client Installation Guide for Microsoft Windows*
 - *Oracle Database Net Services Administrator's Guide*
-

To create a Net Service Name:

1. From the Windows Start menu, select **Oracle – OraClient11g_homen**.

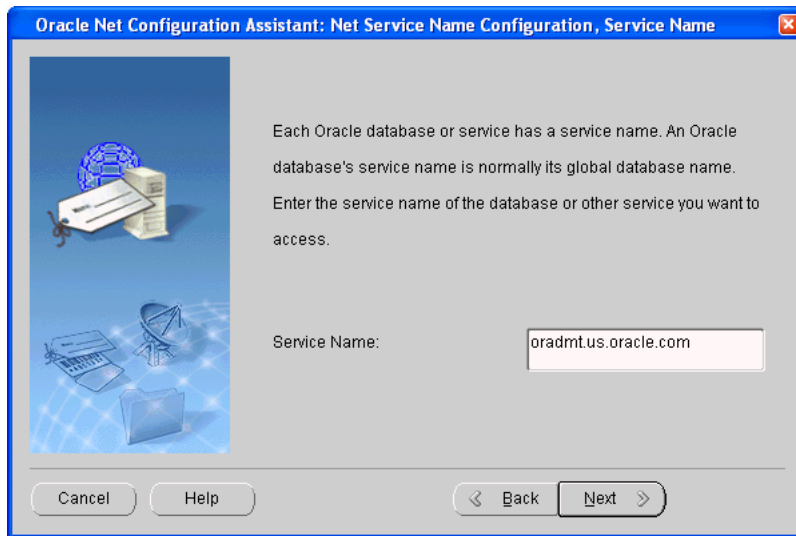
Note: If there are n installations of Oracle Client, they will appear as Oracle - OraClient11g_home1, Oracle - OraClient11g_home2, ..., Oracle - OraClient11g_homen in the Start menu. Be sure to select the appropriate one.

2. Choose **Configuration and Migration Tools**.
3. Choose **Net Configuration Assistant**.
4. Oracle Net Configuration Assistant displays the Welcome page.



Choose **Local Net Service Name configuration**, then **Next**.

5. On the next page, choose **Add** to add a new Net Service Name. Then click **Next**.
6. Provide the Global Database Name in the Service Name text box. (See "[Install Oracle Database](#)" on page 2-1.)

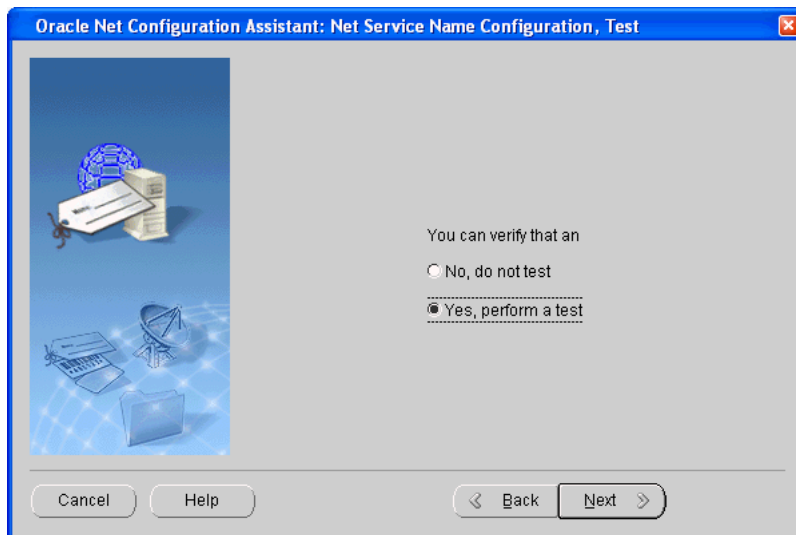


Choose **Next**.

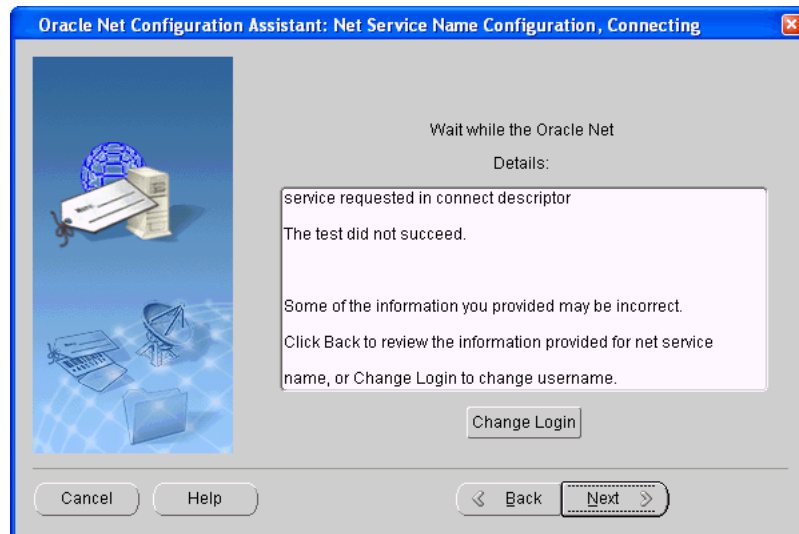
7. On the next page specify the communication protocol. The default is TCP.

Choose **Next**.

8. On the next page, choose **Yes, perform a test** to test the new connection.

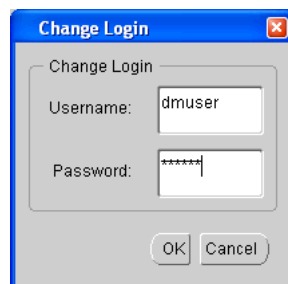


The connection will probably fail until you provide a new user ID and password.

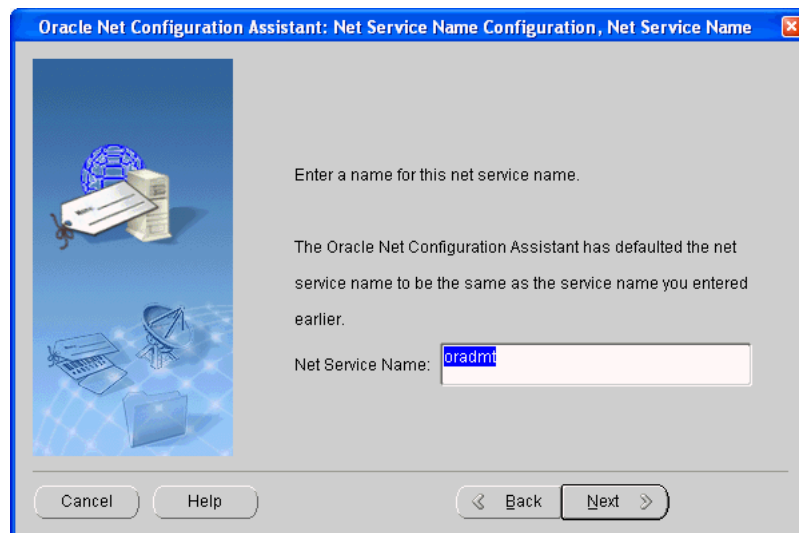


Choose **Change Login**.

Provide the user ID and password of an account that has data mining privileges on the host.



9. If the user name and password were the only problem with the connection, Oracle Net Configuration Assistant makes a successful test connection.



Choose **Next**.

- Oracle Net Configuration Assistant reports that the connection configuration was successful. When you click **Next**, you are returned to the Welcome page, where you have the opportunity to perform additional configuration.

Click **Finish** to exit Oracle Net Configuration Assistant.

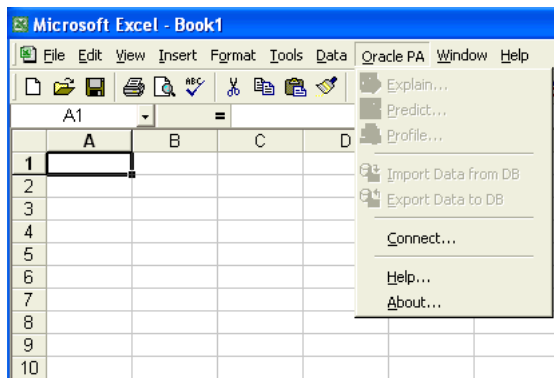
Connect the Spreadsheet Add-In

Oracle Spreadsheet Add-In for Predictive Analytics implements predictive analytics ("one-click" data mining) within Microsoft Excel. You can use the Spreadsheet Add-In to analyze data stored in Excel or in Oracle and to import/export data mining models to/from Oracle Database. Whether the data is stored in Excel or in the database, the Spreadsheet Add-In requires a database connection.

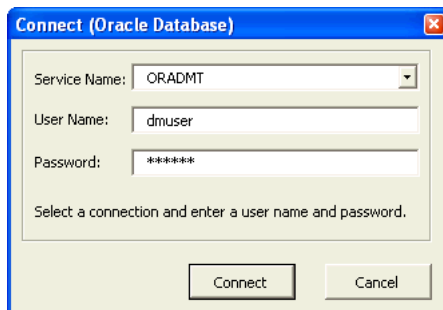
The Spreadsheet Add-In requires Oracle Client. It uses a Net Service Name to connect to the database. Even if the database is local, Oracle Client and a Net Service Name are required.

To connect to Oracle, follow these steps:

- Install Oracle Client on your computer as described in ["Install Oracle Client"](#) on page 3-4.
- Create a Net Service Name as described in ["Create a Net Service Name"](#) on page 3-8.
- Add the Add-In to Excel as described in ["Optionally Install the Spreadsheet Add-In"](#) on page 2-11.
- From the Add-In menu, choose **Connect**.



- The Connect (Oracle Database) dialog appears.



From the list of predefined Net Service Names, choose the connection you want to use. Provide a user name and password and click **Connect** to activate the Spreadsheet Add-In.

See Also: ["Optionally Install the Spreadsheet Add-In"](#) on page 2-11

Users and Privileges for Data Mining

This chapter explains how to create data mining users and control their mining activities through the use of database privileges.

See Also: *Oracle Database 2 Day + Security Guide* for information about Oracle Database security

This chapter contains the following topics:

- [Create a Database User](#)
- [Grant Data Mining Privileges](#)
- [System Privileges for Data Mining](#)
- [Object Privileges for Mining Models](#)

Create a Database User

To create a database user, you must be logged in as a system administrator. You can use any tool that transmits SQL to the database, for example Database Control, SQL*Plus, or SQL Developer.

Example: Create a Database User in SQL*Plus

To create a database user in SQL*Plus:

1. From the Windows Start menu, select the Oracle home of the local database.

Note: If there are n database installations, they will appear as Oracle – OraDb11g_home1, Oracle – OraDb11g_home2, Oracle – OraDb11g_homen in the Start menu.

Each of these refers to a separate *ORACLE_HOME*. Be sure to select the appropriate one.

2. Choose **Application Development**.
3. Choose **SQL*Plus**.
4. Log in with system privileges.

```
Enter user-name: sys / as sysdba
Enter password: password
```

- To create a user named `dmuser`, type these commands. Specify a password of your choosing.

```
CREATE USER dmuser IDENTIFIED BY password
      DEFAULT TABLESPACE USERS
      TEMPORARY TABLESPACE TEMP
      QUOTA UNLIMITED ON USERS;
Commit;
```

Specify the `USERS` and `TEMP` tablespaces. These tablespaces are included in the starter database that was created during the installation, and they are typically included in a database by default. (See ["Install Oracle Database"](#) on page 2-1). The `USERS` and `TEMP` tablespaces are generally assigned to users that generate examples and demonstrations.

Note: Tablespaces for Data Mining users should be assigned according to standard DBA practices, depending on system load and system resources.

- To login as `dmuser`, type the following.

```
CONNECT dmuser
Enter password: password
```

See Also:

Oracle Database 2 Day DBA Online Help for an introduction to creating database users

Oracle Database SQL Language Reference for the complete syntax of the `CREATE USER` statement

Grant Data Mining Privileges

A set of basic privileges is required for data mining. Some may not be required for all mining activities, however it is prudent to grant them all as a group. The `GRANT` statements in [Example 4-1](#) grant these privileges to `dmuser`. You could execute these statements in `SQL*Plus` after logging in with system privileges.

Example 4-1 Privileges Required for Data Mining

```
GRANT CREATE JOB TO dmuser;
GRANT CREATE MINING MODEL TO dmuser;           -- required for creating models
GRANT CREATE PROCEDURE TO dmuser;
GRANT CREATE SEQUENCE TO dmuser;
GRANT CREATE SESSION TO dmuser;
GRANT CREATE SYNONYM TO dmuser;
GRANT CREATE TABLE TO dmuser;
GRANT CREATE TYPE TO dmuser;
GRANT CREATE VIEW TO dmuser;
GRANT EXECUTE ON ctxsys.ctx_ddl TO dmuser;     -- required for text mining
GRANT SELECT ON data TO dmuser;               -- required for data
                                                that is not in your schema
```

Additional access rights are required for exporting and importing mining models, as described in ["Exporting and Importing Mining Models"](#) on page 5-6.

Additional system and object privileges described in "[System Privileges for Data Mining](#)" and "[Object Privileges for Mining Models](#)" can be used to enable or restrict particular mining activities.

Note: To create a user for the Data Mining sample programs, you need to run two configuration scripts as described in "[Installation and Setup](#)" on page 7-1.

System Privileges for Data Mining

A **system privilege** confers the right to perform a particular action in the database or to perform an action on a type of schema objects. For example, the privileges to create tablespaces and to delete the rows of any table in a database are system privileges.

To grant a system privilege, you must either have been granted the system privilege with the `ADMIN OPTION` or have been granted the `GRANT ANY PRIVILEGE` system privilege.

The system privileges listed in [Table 4-1](#) control operations on mining models.

Table 4-1 System Privileges for Data Mining

System Privilege	Allows you to
CREATE MINING MODEL	Create mining models in your own schema.
CREATE ANY MINING MODEL	Create mining models in any schema.
ALTER ANY MINING MODEL	Change the name or cost matrix of any mining model in any schema.
DROP ANY MINING MODEL	Drop any mining model in any schema.
SELECT ANY MINING MODEL	Apply (score) a mining model in any schema, also view model details in any schema.
COMMENT ANY MINING MODEL	Add a comment to any mining model in any schema. (See " Adding a Comment to a Mining Model " on page 5-4.)
AUDIT ANY	Generate an audit trail for any mining model (or any object) in any schema. (See " Auditing Mining Models " on page 5-4.)

Example: Grant System Privileges for Data Mining

The following statements allow `dmuser` to score models and view model details in any schema as long as `SELECT` access has been granted on the data. However, `dmuser` can only create models in the `dmuser` schema.

```
GRANT CREATE MINING MODEL TO dmuser;
GRANT SELECT ANY MINING MODEL TO dmuser;
```

The following statement revokes the privilege of scoring or viewing model details in other schemas. When this statement is executed, `dmuser` can only perform data mining activities in the `dmuser` schema.

```
REVOKE SELECT ANY MINING MODEL FROM dmuser;
```

Object Privileges for Mining Models

An object privilege confers the right to perform a particular action on a specific schema object. For example, the privilege to delete rows from the `SH.PRODUCTS` table is an example of an object privilege.

A user automatically has all object privileges for schema objects in the user's own schema. A user can grant object privilege on objects in the user's own schema to other users or roles.

The object privileges listed in [Table 4–2](#) control operations on specific mining models.

Table 4–2 Object Privileges for Mining Models

Object Privilege	Allows you to
ALTER MINING MODEL	Change the name or cost matrix of the specified mining model object.
SELECT MINING MODEL	Apply (score) and view model details of the specified mining model object.

Example: Grant Object Privileges on Mining Models

The following statements allow `dmuser` to apply the model `testmodel` to the `sales` table, specifying different cost matrixes with each apply. The user `dmuser` can also rename the model `testmodel`. The `testmodel` model and `sales` table are in the `sh` schema, not in the `dmuser` schema.

```
GRANT SELECT ON MINING MODEL sh.testmodel TO dmuser;
GRANT ALTER ON MINING MODEL sh.testmodel TO dmuser;
GRANT SELECT ON sh.sales TO dmuser;
```

This statement prevents `dmuser` from renaming or changing the cost matrix of `testmodel`. However, `dmuser` can still apply `testmodel` to the `sales` table.

```
REVOKE ALTER ON MINING MODEL sh.testmodel FROM dmuser;
```

See Also: [Chapter 5](#) for information on mining model schema objects

Working With Mining Model Schema Objects

In this chapter, you will learn how to retrieve information about mining models from the data dictionary and perform various operations on mining models.

See Also: [Chapter 4, "Users and Privileges for Data Mining"](#)

This chapter contains the following topics:

- [Obtaining Information from the Data Dictionary](#)
- [Data Definition Language for Mining Models](#)
- [Adding a Comment to a Mining Model](#)
- [Auditing Mining Models](#)
- [Exporting and Importing Mining Models](#)
- [Importing From PMML](#)

Obtaining Information from the Data Dictionary

Mining models are database schema objects. They can be queried in the ALL, DBA, and USER data dictionary views.

The data dictionary views in [Table 5–1](#) reveal information about mining models created by Oracle Data Mining.

Table 5–1 Mining Model Data Dictionary Views

ALL_ Views	DBA_ Views	USER_ Views
ALL_MINING_MODELS	DBA_MINING_MODELS	USER_MINING_MODELS
ALL_MINING_MODEL_ATTRIBUTES	DBA_MINING_MODEL_ATTRIBUTES	USER_MINING_MODEL_ATTRIBUTES
ALL_MINING_MODEL_SETTINGS	DBA_MINING_MODEL_SETTINGS	USER_MINING_MODEL_SETTINGS

Obtaining Information about Mining Models

You can query the ALL_MINING_MODELS data dictionary view to obtain information about all accessible mining model objects.

```
SQL> DESCRIBE all_mining_models
```

```
Name                                     Null?    Type
-----
OWNER                                     NOT NULL VARCHAR2 (30)
MODEL_NAME                               NOT NULL VARCHAR2 (30)
MINING_FUNCTION                           VARCHAR2 (30)
```

ALGORITHM	VARCHAR2 (30)
CREATION_DATE	NOT NULL DATE
BUILD_DURATION	NUMBER
MODEL_SIZE	NUMBER
COMMENTS	VARCHAR2 (4000)

The information in the COMMENTS column is generated by a SQL COMMENT statement. The comment provides descriptive information about the model. If no comment was generated by SQL COMMENT, the COMMENTS column is empty. See ["Adding a Comment to a Mining Model"](#) on page 5-4.

See Also:

Oracle Data Mining Concepts for information about mining functions and algorithms

Oracle Data Mining Application Developer's Guide for more information about ALL_MINING_MODELS

Obtaining Information about Mining Model Attributes

You can query the ALL_MINING_MODEL_ATTRIBUTES data dictionary view to obtain information about all accessible mining model attributes.

```
SQL> DESCRIBE all_mining_model_attributes
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2 (30)
MODEL_NAME	NOT NULL	VARCHAR2 (30)
ATTRIBUTE_NAME	NOT NULL	VARCHAR2 (30)
ATTRIBUTE_TYPE		VARCHAR2 (11)
DATA_TYPE		VARCHAR2 (12)
DATA_LENGTH		NUMBER
DATA_PRECISION		NUMBER
DATA_SCALE		NUMBER
USAGE_TYPE		VARCHAR2 (8)
TARGET		VARCHAR2 (3)

Note: The attributes listed in this data dictionary view are the physical columns in the build data that were used to construct the model. Some or all of these columns should be present for scoring. These **data attributes** are referred to as the **model signature**.

The term *attribute* is more accurately used to designate the numericals and categoricals derived from the data attributes for manipulation by the algorithm. These **model attributes** may or may not correspond to data attributes, depending on transformations and on whether or not the column is nested. The model attributes can be viewed in the model details returned by the GET_MODEL_DETAILS functions in the DBMS_DATA_MINING package.

See Also:

Oracle Data Mining Application Developer's Guide for more information about attributes

Oracle Data Mining Application Developer's Guide for more information about ALL_MINING_MODEL_ATTRIBUTES

Oracle Database PL/SQL Packages and Types Reference for details about the GET_MODEL_DETAILS functions

Obtaining Information about Mining Model Settings

You can query the ALL_MINING_MODEL_SETTINGS data dictionary view to obtain information about all accessible mining model settings.

```
SQL> DESCRIBE all_mining_model_settings
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2 (30)
MODEL_NAME	NOT NULL	VARCHAR2 (30)
SETTING_NAME	NOT NULL	VARCHAR2 (30)
SETTING_VALUE		VARCHAR2 (4000)
SETTING_TYPE		VARCHAR2 (7)

Model settings are specified in a settings table that is used in the model build. The settings all have default values. The defaults are used when they are not overridden by settings specified in the settings table, or when there is no settings table.

See Also:

Oracle Database PL/SQL Packages and Types Reference for more information about model settings

Oracle Data Mining Application Developer's Guide for more information about ALL_MINING_MODEL_SETTINGS

Data Definition Language for Mining Models

The DDL operations for mining models are described in [Table 5-2](#). The operations are procedures in the DBMS_DATA_MINING PL/SQL package.

Table 5-2 DDL for Mining Models

DDL	DBMS_DATA_MINING Procedure	Description
Add cost matrix	ADD_COST_MATRIX	Adds a cost matrix to a classification model
Alter reverse expression	ALTER_REVERSE_EXPRESSION	Alters the reverse transformation expression associated with a model
Create model	CREATE_MODEL	Creates a model
Drop model	DROP_MODEL	Drops a model
Remove cost matrix	REMOVE_COST_MATRIX	Removes the cost matrix from a classification model
Rename model	RENAME_MODEL	Renames a model

See: *Oracle Database PL/SQL Packages and Types Reference* for details

Adding a Comment to a Mining Model

Comments can be used to associate descriptive information with a database object. You can associate a comment with a mining model using a SQL `COMMENT` statement.

```
COMMENT ON MINING MODEL schema_name.model_name IS string;
```

Note: To add a comment to a model in another schema, you must have the `COMMENT ANY MODEL` system privilege.

See [Table 4–1, "System Privileges for Data Mining"](#).

To drop a comment, set it to the empty `''` string.

The following statement adds a comment to the model `DT_SH_CLAS_SAMPLE` in your own schema.

```
SQL> COMMENT ON MINING MODEL dt_sh_clas_sample IS
      'Decision Tree model predicts promotion response';
```

You can view the comment by querying the catalog view `USER_MINING_MODELS`.

```
SQL> COLUMN comments FORMAT a22
SQL> SELECT model_name, mining_function, algorithm, comments FROM user_mining_models;
```

MODEL_NAME	MINING_FUNCTION	ALGORITHM	COMMENTS
DT_SH_CLAS_SAMPLE	CLASSIFICATION	DECISION_TREE	Decision Tree model predicts promotion response

To drop this comment from the database, issue the following statement:

```
SQL> COMMENT ON MINING MODEL dt_sh_clas_sample '';
```

Auditing Mining Models

The SQL auditing system is a DBA tool for tracking operations on database objects in a production environment. Data mining models can be tracked with the SQL auditing system.

Note: To audit a mining model in another schema, you must have the `AUDIT ANY` system privilege.

See [Table 4–1, "System Privileges for Data Mining"](#).

Enabling Auditing in the Database

DBAs use auditing to monitor activity in the database. Audit records can be generated to monitor various database operations. They include information such as the operation that was audited, the user performing the operation, and the date and time of the operation.

Audit records can be stored in either a data dictionary table, called the database audit trail, or in operating system files, called an operating system audit trail.

The database initialization parameter `AUDIT_TRAIL` controls auditing capabilities in the database. To enable auditing, set `AUDIT_TRAIL` to one of the following:

```
DB
DB, EXTENDED
```

OS
XML
XML, EXTENDED

Set `AUDIT_TRAIL` to `NONE` to prevent auditing information from being recorded. By default, `AUDIT_TRAIL` is set to `DB`.

See Also: *Oracle Database Security Guide* for details about the `AUDIT_TRAIL` settings

Opening an Audit Trail on Mining Models

Use the SQL `AUDIT` statement to open an auditing trail on a data mining model.

```
AUDIT {operation|ALL} ON MINING MODEL schema_name.model_name;
```

You can track the following operations on mining models.

Audit Operation	Description
AUDIT	Generate an audit trail for a mining model
COMMENT	Add a comment to a mining model
GRANT	Give permission to a user to access the model
RENAME	Change the name of the model
SELECT	Apply the model or view its signature.

For example, this statement generates an audit trail for all `GRANT` operations on the model `NB_SH_CLAS_SAMPLE` in the `DMUSER` schema.

```
SQL> AUDIT GRANT ON MINING MODEL dmuser.nb_sh_clas_sample;
```

This statement generates an audit trail for all operations on the same model.

```
SQL> AUDIT GRANT,AUDIT,COMMENT,RENAME,SELECT
      ON MINING MODEL dmuser.nb_sh_clas_sample;
```

You can refine the criteria for auditing with the following additional semantics.

```
AUDIT {operation|ALL} ON MINING MODEL schema_name.model_name
      [BY [SESSION|ACCESS]]
      [WHENEVER [NOT] SUCCESSFUL];
```

Closing the Audit Trail

Use the `NOAUDIT` statement to stop one or more auditing operations previously enabled by the `AUDIT` statement.

```
NOAUDIT {operation| ALL} ON MINING MODEL model_name
      [WHENEVER [NOT] SUCCESSFUL];
```

Viewing the Audit Trail

For each audited operation, Oracle Database produces an audit record containing:

- The name of the user performing the operation
- The type of operation
- The object involved in the operation

- The date and time of the operation

Several data dictionary views present auditing information. Some examples are:

- `DBA_AUDIT_OBJECT` displays audit trail records for all objects in the database.
- `USER_AUDIT_OBJECT` displays audit trail records for all objects accessible to the current user
- `DBA_OBJ_AUDIT_OPTS` describes auditing options for all objects in the database.
- `USER_OBJ_AUDIT_OPTS` describes auditing options for all objects owned by the current user.

Note: The Oracle Database auditing system is a powerful, highly configurable tool for tracking operations on schema objects. Refer to the following manuals for more information:

- *Oracle Database Security Guide* for a comprehensive discussion of database auditing
 - *Oracle Database SQL Language Reference* for documentation of the `AUDIT` and `NOAUDIT` statements
 - *Oracle Database Reference* for documentation of the `AUDIT_TRAIL` initialization parameter and the data dictionary views for querying the database audit trail
-
-

Exporting and Importing Mining Models

You can export data mining models to flat files to back up work in progress or to move models to a different instance of Oracle Database Enterprise Edition (such as from a development database to a production database). All methods for exporting and importing models are based in Oracle Data Pump technology.

Using Oracle Data Pump

Oracle Data Pump consists of two command-line clients and two PL/SQL APIs. The command-line clients, `EXPDP` and `IMPDP`, provide an easy-to-use interface to the Data Pump export and import utilities. `EXPDP` and `IMPDP` can be used to export/import entire schemas or databases.

The Data Pump export utility writes the schema objects, including the tables and metadata that constitute mining models, to a dump file set. The Data Pump import utility retrieves the schema objects, including the model tables and metadata, from the dump file set and restores them in the target database.

`EXPDP` and `IMPDP` *cannot* be used to export/import individual mining models.

See Also: *Oracle Database Utilities* for information about Oracle Data Pump and the `expdp` and `impdp` utilities

Using `EXPORT_MODEL` and `IMPORT_MODEL`

The `DBMS_DATA_MINING` package includes the `EXPORT_MODEL` and `IMPORT_MODEL` procedures for exporting and importing individual mining models. `EXPORT_MODEL` and `IMPORT_MODEL` use the export and import facilities of Oracle Data Pump.

See Also: *Oracle Database PL/SQL Packages and Types Reference* for detailed information about the export and import procedures in the `DBMS_DATA_MINING` package

Database Privileges for Export/Import

Your options for exporting and importing mining models depend on your access rights in the database:

- A DBA can use `EXPDP` to export a full database and `IMPDP` to import a full database. All mining models in the database are included in the export/import.
- A DBA or individual user can use `EXPDP` to export a schema and `IMPDP` to import a schema. All mining models in the schema are included in the export/import.
- A DBA or individual user can use `DBMS_DATA_MINING.EXPORT_MODEL` to export individual models and `DBMS_DATA_MINING.IMPORT_MODEL` to import individual models. These procedures can export/import a single mining model, all the mining models in a schema, or mining models that match specific criteria.

By default, `EXPORT_MODEL` and `IMPORT_MODEL` support export and import within the same schema. You may need special privileges in the database to import models into a different schema. These privileges are granted by the `EXP_FULL_DATABASE` and `IMP_FULL_DATABASE` roles, which are only available to privileged users (such as `SYS` or a user with the `DBA` role). You do not need these roles to export/import models within your own schema.

To import models, you must have the same database privileges as the user who created the dump file set. Otherwise, a DBA with full system privileges must import the models.

Directory Object for EXPORT_MODEL and IMPORT_MODEL

`EXPORT_MODEL` and `IMPORT_MODEL` use a directory object to identify the location of the dump file set. A directory object is a logical name in the database for a physical directory on the host computer.

To export Data Mining models, you must have write access to the directory object and to the file system directory that it represents. To import Data Mining models, you must have read access to the directory object and to the file system directory. Also, the database itself must have access to file system directory.

You must have the `CREATE ANY DIRECTORY` privilege to create directory objects.

The following SQL command creates, or re-creates if it already exists, a directory object named `dmtest`. The file system directory (in this example, `C:\oracle\product\11.1.0\dmining`) must already exist and have shared read/write access rights granted by the operating system.

```
CREATE OR REPLACE DIRECTORY dmtest AS 'c:\oracle\product\11.1.0\dmining';
```

This SQL command gives user `dmuser` both read and write access to `dmtest`.

```
GRANT READ,WRITE ON DIRECTORY dmtest TO dmuser;
```

Note: For security reasons, use caution in granting permissions.

See Also: *Oracle Database SQL Language Reference* for information about the `CREATE DIRECTORY` statement

Tablespace for IMPORT_MODEL

Mining models are stored in the default tablespace of the mining model owner, or in a tablespace to which the owner has access. The tablespace must also exist in the target database, and the target user must have access to it. If the tablespace does not exist in the target database, you must create it before importing the models.

For example, if the models were created in schema DMUSER and the default tablespace for DMUSER is USERS, then the USERS tablespace must exist in the target database. You can create the USERS tablespace and grant access to a target user with appropriate tablespace quota as follows.

```
connect / as sysdba;
create tablespace USERS datafile 'data_file_name' size 200M autoextend on;
alter user target_user quota unlimited on USERS;
```

Tables Created By EXPORT_MODEL and IMPORT_MODEL

The Data Mining export and import utilities create tables in the user's schema that are for internal use only:

- DM\$P_MODEL_EXPIMP_TEMP. Used for internal purposes during export and import, and provides a job history.
- DM\$P_MODEL_IMPORT_TEMP. Used only for internal purposes during import.
- DM\$P_MODEL_TABKEY_TEMP. Used only for internal purposes during export and import.

Do not alter these tables. However, you may drop them when no export or import job is running. The utilities will re-create them for the next job.

Examples: Exporting and Importing Mining Models

This example creates a dump file containing two models and imports the models from the dump file.

This example was generated on a Linux system. The directory object `mydir` identifies the path `/scratch/dumpfiles`. (See ["Directory Object for EXPORT_MODEL and IMPORT_MODEL"](#) on page 5-7.)

Exporting Models from the DMUSER Schema

In this example, the `dmuser` schema contains two mining models.

```
SQL> CONNECT dmuser
Enter password: password

SQL> SELECT model_name FROM user_mining_models;

MODEL_NAME
-----
GLMR_SH_REGR_SAMPLE
SVMC_SH_CLAS_SAMPLE
```

The following commands create the directory object `mydir` and export all models from `dmuser` to `mydir`.

```
SQL> CREATE OR REPLACE DIRECTORY mydir AS '/scratch/dumpfiles';
SQL> EXECUTE dbms_data_mining.export_model('all_dmuser_models.dmp', 'mydir');
```


An export or import creates a log file in the same directory as the dump file. Error messages are returned to the current output device (such as the screen), and the log file may provide additional information.

This sample export created two files in the `mydir` directory:

- A dump file named `ALL_DMUSER_MODELS01.DMP` (note the 2-digit suffix added to the name)
- A log file with the name `DMUSER_EXP_920.LOG`

For detailed information about the default names of files, see *Oracle Database PL/SQL Packages and Types Reference*.

You can view the log file using a system command or editor. You must know the path of the physical directory in order to locate the file.

`DMUSER_EXP_920.LOG` lists the two exported mining models and supporting objects .

```
Starting "DMUSER"."DMUSER_exp_17": DM_EXPIMP_JOB_ID=17
Estimate in progress using BLOCKS method...
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
Total estimation using BLOCKS method: 1.062 MB
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processing object type TABLE_EXPORT/TABLE/INDEX/INDEX
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Processing object type TABLE_EXPORT/TABLE/POST_INSTANCE/PROCACT_INSTANCE
. . exported "DMUSER"."DM$PIGLMR_SH_REGR_SAMPLE"          7.085 KB      14 rows
. . exported "DMUSER"."DM$PISVMC_SH_CLAS_SAMPLE"         7.507 KB      17 rows
. . exported "DMUSER"."DM$PCGLMR_SH_REGR_SAMPLE"        53.27 KB     2278 rows
. . exported "DMUSER"."DM$PAGLMR_SH_REGR_SAMPLE"        5.796 KB      24 rows
. . exported "DMUSER"."DM$PBGLMR_SH_REGR_SAMPLE"        15.76 KB      67 rows
. . exported "DMUSER"."DM$PDGLMR_SH_REGR_SAMPLE"         8 KB         66 rows
. . exported "DMUSER"."DM$PDSVMC_SH_CLAS_SAMPLE"        9.023 KB      88 rows
. . exported "DMUSER"."DM$PFGLMR_SH_REGR_SAMPLE"        5.656 KB      10 rows
. . exported "DMUSER"."DM$POSVMC_SH_CLAS_SAMPLE"        5.320 KB       8 rows
. . exported "DMUSER"."DM$PXSVMC_SH_CLAS_SAMPLE"        7.265 KB      77 rows
. . exported "DMUSER"."DM$PZSVMC_SH_CLAS_SAMPLE"        6.164 KB       1 rows
. . exported "DMUSER"."DM$P_MODEL_EXPIMP_TEMP"          5.921 KB       2 rows
Master table "DMUSER"."DMUSER_exp_17" successfully loaded/unloaded
*****
Dump file set for DMUSER.DMUSER_exp_17 is:
  /scratch/dumpfiles/all_dmuser_models01.dmp
Job "DMUSER"."DMUSER_exp_17" successfully completed at 10:30:19
```

Importing Models Into the Same Schema

The exported models still exist in `DMUSER`. In this example, we drop the models before importing from the dump file. An import will not overwrite an existing model with the same name.

```
SQL> EXECUTE dbms_data_mining.drop_model('GLMR_SH_REGR_SAMPLE');
SQL> EXECUTE dbms_data_mining.drop_model('SVMC_SH_CLAS_SAMPLE');
```

The following command restores all models from the dump file to the `DMUSER` schema.

```
SQL> EXECUTE dbms_data_mining.import_model('all_dmuser_models01.dmp', 'mydir');
```

Importing Models Into a Different Schema

A user with the necessary privileges can load the models from a dump file into a different schema. The target schema must have the same permissions and have access to the same tablespace as the schema from which the models were exported.

```
CREATE USER dmuser2 IDENTIFIED BY password
        default tablespace dmuser
        temporary tablespace temp
        quota unlimited on dmuser;
```

```
@ $ORACLE_HOME/RDBMS/demo/dmshgrants dmuser2
```

The import command, also executed as SYS, loads the two models into the DMUSER2 schema.

```
SQL> EXECUTE dbms_data_mining.import_model('all_dmuser_models01.dmp', 'mydir',
null, null, null, 'todmuser2', 'dmuser:dmuser2');
```

A parameter specifies TODMUSER2.LOG as the name of the log file; the .LOG extension is added automatically to the name. The log file shows the names of the imported models and supporting metadata.

```
Master table "SYS"."todmsur2" successfully loaded/unloaded
Starting "SYS"."todmsur4":  DM_EXPIMP_JOB_ID=21
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
. . imported "DMUSER2"."DM$PIGLMR_SH_REGR_SAMPLE"          7.085 KB      14 rows
. . imported "DMUSER2"."DM$PISVMC_SH_CLAS_SAMPLE"         7.507 KB      17 rows
. . imported "DMUSER2"."DM$PCGLMR_SH_REGR_SAMPLE"        53.27 KB     2278 rows
. . imported "DMUSER2"."DM$PAGLMR_SH_REGR_SAMPLE"         5.796 KB      24 rows
. . imported "DMUSER2"."DM$PBGLMR_SH_REGR_SAMPLE"        15.76 KB      67 rows
. . imported "DMUSER2"."DM$PDGLMR_SH_REGR_SAMPLE"         8 KB         66 rows
. . imported "DMUSER2"."DM$PDSVMC_SH_CLAS_SAMPLE"         9.023 KB      88 rows
. . imported "DMUSER2"."DM$PFGLMR_SH_REGR_SAMPLE"         5.656 KB      10 rows
. . imported "DMUSER2"."DM$POSVMC_SH_CLAS_SAMPLE"         5.320 KB       8 rows
. . imported "DMUSER2"."DM$PXVMC_SH_CLAS_SAMPLE"          7.265 KB      77 rows
. . imported "DMUSER2"."DM$PZVMC_SH_CLAS_SAMPLE"          6.164 KB       1 rows
. . imported "DMUSER2"."DM$P_MODEL_EXPIMP_TEMP"           5.921 KB       2 rows
Processing object type TABLE_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processing object type TABLE_EXPORT/TABLE/INDEX/INDEX
Processing object type TABLE_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Processing object type TABLE_EXPORT/TABLE/POST_INSTANCE/PROCACT_INSTANCE
Job "SYS"."todmsur2" successfully completed at 11:15:26
```

Exporting and Importing Specific Models

You can export/import a single model, a list of models, or a group of models that share certain characteristics. For example, the following command exports two models, NMF_SH_SAMPLE and SVMR_SH_REGR_SAMPLE to the directory specified by directory object EXP_OUT.

```
SQL>execute dbms_data_mining.export_model ( 'models2_out', 'EXP_OUT',
        'name in ('SVMR_SH_REGR_SAMPLE')');
```

See: *Oracle Database PL/SQL Packages and Types Reference* for more examples

Importing From PMML

This functionality is available starting with Oracle Database 11g Release 2 (11.2.0.2) Data Mining.

You can import a mining model represented in Predictive Model Markup Language (PMML). The model must be of type `RegressionModel`, either linear regression or binary logistic regression.

PMML is an XML-based standard specified by the Data Mining Group (<http://www.dmg.org>). Applications that are PMML-compliant can deploy PMML-compliant models that were created by any vendor. Oracle Data Mining supports the core features of PMML 3.1 for regression models.

See: *Oracle Database PL/SQL Packages and Types Reference* for more information about PMML import

Upgrading Oracle Data Mining

This chapter explains how to upgrade or downgrade Oracle Data Mining 11g Release 2 (11.2). It contains the following sections:

- [Data Mining Upgrade Overview](#)
- [Upgrade Using Database Upgrade Assistant](#)
- [Upgrade Using Export/Import](#)
- [Post Upgrade Steps](#)
- [Downgrade Data Mining](#)

Data Mining Upgrade Overview

Oracle Data Mining upgrade is fully integrated with the Oracle Database upgrade process. Whether you are upgrading from 11.1 release or from 10g releases, Data Mining models and metadata are upgraded automatically during the upgrade of Oracle Database.

To upgrade a database, you can use Database Upgrade Assistant (DBUA) or you can perform a manual upgrade using export/import utilities.

See: *Oracle Database Upgrade Guide* for complete database upgrade instructions

Data Mining Model Upgrade

Upgraded models will continue to work as they did in prior releases. New models that you create in the upgraded environment can make use of the new mining functionality introduced in the new release.

Note: Models created by the Oracle Data Mining PL/SQL API can be upgraded from 10.1 to 11g.

Models created by the Oracle Data Mining Java API *cannot* be upgraded from 10.1 to 11g. The 10.1 version of the Java API was no longer supported in Oracle Data Mining 10.2.

Model upgrade from 9.2 release to 11g releases is not supported.

Upgrading from 10g

In Oracle Data Mining 10g, Data Mining metadata and PL/SQL packages are stored in the DMSYS schema. In Oracle Data Mining 11g, DMSYS no longer exists; Data Mining metadata objects are stored in SYS.

During the upgrade from 10g to 11g, all Data Mining metadata objects and PL/SQL packages are migrated from DMSYS to SYS. After the upgrade, when you determine that there is no need to perform a downgrade, set the initialization parameter COMPATIBLE to 11.2 and drop the DMSYS schema and its associated objects from the upgraded database as follows:

```
SQL> CONNECT / AS sysdba;
SQL> DROP USER dmsys CASCADE;
SQL> DELETE FROM sys.exppkgact$ WHERE schema = 'DMSYS';
SQL> SELECT COUNT(*) FROM dba_synonyms WHERE table_owner = 'DMSYS';
```

If the result is non-zero rows, create and run a SQL script as follows:

```
SQL> set head off
SQL> spool directory_path/drop_dmsys_synonyms.sql
SQL> SELECT 'Drop public synonym ' || '"' || SYNONYM_NAME || '"'
        FROM DBA_SYNONYMS WHERE TABLE_OWNER = 'DMSYS';
SQL> spool off
SQL> @directory_path/drop_dmsys_synonyms.sql
```

Once DMSYS is removed, the SYS.DBA_REGISTRY view will no longer list Oracle Data Mining as a component.

After upgrading to 11g, you can no longer switch to the Data Mining Scoring Engine (DMSE). The Scoring Engine does not exist in 11g.

See Also: ["Oracle Database 11g Release 1 New Features in Oracle Data Mining"](#) on page ii-x

Upgrading from 11.1

If you upgrade from 11.1 release to 11.2, check for the existence of DMSYS. If the database was previously upgraded from 10g, DMSYS may still be present. If DMSYS is still present in the database, you should remove it.

Upgrade Using Database Upgrade Assistant

Oracle Database Upgrade Assistant provides a graphical user interface that guides you interactively through the upgrade process. To start the Upgrade Assistant:

1. Go to the Windows **Start** menu and choose the Oracle home directory.
2. Choose the **Configuration and Migration Tools** menu.
3. Launch the **Upgrade Assistant**.

On Linux platforms, run the DBUA utility to upgrade Oracle Database.

Upgrade Using Export/Import

If you wish, you can use a less automated approach to upgrading Oracle Data Mining. You can export the models created in a previous version of Oracle Database and import them into a new 11g Release 2 (11.2) database.

Export/Import Release 11.1 Data Mining Models

To export models from an 11.1 database to a dump file and import them into an 11.2 database, follow the instructions in ["Exporting and Importing Mining Models"](#) on page 5-6.

Export/Import Release 10g Data Mining Models

To export models from a 10g database to a dump file, follow the instructions in ["Exporting and Importing Mining Models"](#) on page 5-6. If any models were created by the 10.1 Java API, do not include them in the export; they are not supported in 11g.

Before importing the models from the dump file, run the DMEIDMSYS script to create the DMSYS schema in the 11g database.

```
SQL>CONNECT / as sysdba;
SQL>@ORACLE_HOME\RDBMS\admin\dmeidmsys.sql
SQL>EXIT;
```

Note: The TEMP tablespace must already exist in the 11g database. The DMEIDMSYS script uses the TEMP and SYSAUX tablespaces to create the DMSYS schema.

To import the dump file into the database:

```
%ORACLE_HOME\bin\impdp system\

```

If you shutdown the database before operating on the upgraded mining models, this will also flush the shared pool.

After the import is complete, you should drop the DMSYS schema from the database.

Post Upgrade Steps

After upgrading the database, check the upgrade log file and the SYS.DBA_REGISTRY view to ensure that the upgrade process completed successfully. Also check the DBA_MINING_MODELS view in the upgraded database. The newly upgraded mining models should be listed in this view.

After you have verified the upgrade and confirmed that there will be no need to downgrade, you should set the initialization parameter COMPATIBLE to 11.2.

Important: In Oracle 11g, the CREATE MINING MODEL privilege must be granted to Data Mining user accounts. This privilege is required for creating Data Mining models in 11g. Refer to [Chapter 4, "Users and Privileges for Data Mining"](#) for more information.

Downgrade Data Mining

Before downgrading the database back to the previous version, ensure that no 11g Release 2 (11.2) mining models were created in the upgraded database. Issue the following SQL statement in `SYS` to verify:

```
SQL>SELECT o.name FROM sys.model$ m, sys.obj$ o
        WHERE m.obj#=o.obj# AND m.version=2;
```

If there are any 11g Release 2 (11.2) mining models in the database, you must manually delete them using the `DBMS_DATA_MINING.DROP_MODEL` routine before downgrading the database. If you do not do this, the database downgrade process will be aborted. See *Oracle Database PL/SQL Packages and Types Reference* for the calling syntax of `DROP_MODEL`.

The Data Mining Sample Programs

You can learn a great deal about the Oracle Data Mining APIs from the Data Mining sample programs. The programs illustrate typical approaches to data preparation, algorithm selection, algorithm tuning, testing, and scoring. Each program creates a mining model in the database. All the programs include extensive inline comments to help you understand the code.

See Also: *Oracle Data Mining Application Developer's Guide* for information about the Data Mining APIs

Note: The Oracle Data Mining Java API is deprecated in this release.

The Java sample programs are still shipped, but Oracle recommends that you not use the Oracle Data Mining Java API in new applications. Support for deprecated features is for backward compatibility only

This chapter includes the following sections:

- [Installation and Setup](#)
- [The PL/SQL Programs](#)
- [The Java Programs](#)
- [The Sample Data](#)

Installation and Setup

The Data Mining sample programs are installed with Oracle Database Examples. They are also available for download from the Oracle Technology Network:

The programs require access to a database that includes the sample schemas. Before you can run the programs, you must run two configuration scripts to configure the data and assign the required privileges to your user ID.

Install the Sample Programs

Follow these steps to install the sample programs:

1. Install Oracle Database with the sample schemas, or obtain access to a database that includes the sample schemas.
 - If you followed the instructions in "[Install Oracle Database](#)" on page 2-1, the sample schemas are installed automatically in the starter database. Be sure to unlock the SH schema.

- If the database does not include the sample schemas, you can install them manually or by using Oracle Database Configuration Assistant. See *Oracle Database Sample Schemas* for instructions.
- 2. Determine whether or not Database Examples was installed with Oracle Database. Database Examples provides a set of sample programs that illustrate numerous features of Oracle Database, including Oracle Data Mining. The programs are loaded into the `RDBMS/demo` subdirectory of Oracle home.

If Database Examples was not installed, you can perform the installation by following the instructions in "[Optionally Install Oracle Database Examples](#)" on page 2-10. Alternatively, you can download the Data Mining sample programs from the Oracle Technology Network.

<http://www.oracle.com/technetwork/database/options/odm/index.html>

Run the Configuration Scripts

Follow these steps to configure the sample data and grant the necessary privileges to your data mining user ID.

1. Log in to SQL*Plus with system privileges.

```
Enter user-name: sys / as sysdba
Enter password: password
```

2. If you do not have a user ID for your data mining activities, you can create one by following the instructions in "[Example: Create a Database User in SQL*Plus](#)" on page 4-1.
3. Run `dmshgrants.sql` to grant data mining privileges and `SH` access to your user ID. Several tables in `SH` are used by the Data Mining sample programs. Specify the data mining user name as the parameter. Specify the full path of the Oracle home directory.

```
@ ORACLE_HOME\RDBMS\demo\dmshgrants dmuser
```

4. Now connect to the database as the Data Mining user.

```
CONNECT dmuser
Enter password: password
```

5. Run `dmsh.sql` to populate the schema of the Data Mining user with tables, views, and other objects needed by the sample programs. Specify the full path of the Oracle home directory.

```
@ ORACLE_HOME\RDBMS\demo\dmsh
COMMIT;
```

Locate the Sample Programs

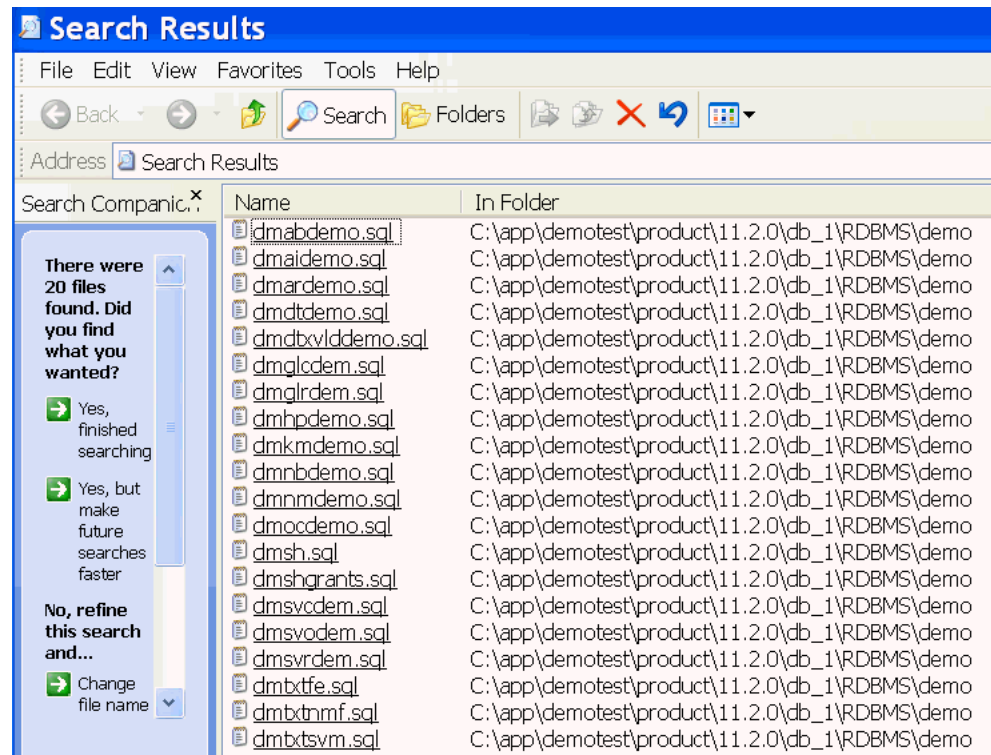
This section explains how to locate the sample programs if they were installed with Database Examples.

To locate the PL/SQL programs, navigate to the parent directory and search for the files that start with `dm` and end with `.sql`.

For example, if Database Examples was installed in Oracle home

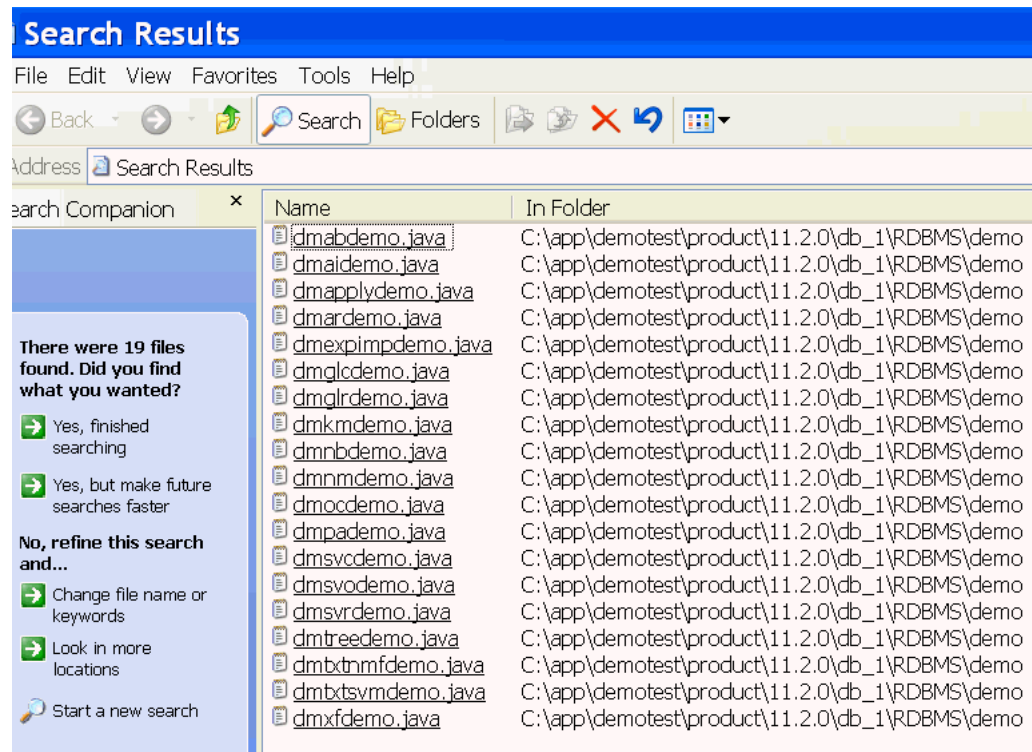
`C:\app\demotest\product\11.2.0\db_1\`, then navigate to

`C:\app\demotest\product\11.2.0\db_1\RDBMS\demo\` and use Windows Search to find the files named `dm*.sql`. Windows Search returns the list of Data Mining PL/SQL programs, as shown in [Figure 7-1](#).

Figure 7–1 The Data Mining Sample PL/SQL Programs

Note: The files listed in [Figure 7–1](#) include all the Data Mining PL/SQL programs. However, one of the files, `dmhpdemo.sql`, is *not* a Data Mining program.

Use Windows Search to find the files named `dm*.java` in the same directory. Windows Search returns the Data Mining Java programs, as shown in [Figure 7–2](#).

Figure 7–2 The Data Mining Sample Java Programs

Run the Sample Programs

You can run the sample programs as many times as you wish. The programs clean up the results of the previous run before executing the current run.

While the program is running, it displays the program code and the program output.

Run the PL/SQL Sample Programs

To run the PL/SQL programs:

1. Start SQL*Plus and log in as the Data Mining user.

```
Enter user-name: dmuser
Enter password: password
```

2. Run the program by specifying an *at* sign (@) followed by the fully-qualified path of the program. In the following example, replace *ORACLE_HOME* with the path of the Oracle home directory.

```
SQL>@ ORACLE_HOME\RDBMS\demo\dmnbdemo
```

This example executes the program `dmnbdemo.sql`, which creates a Naive Bayes model.

Prepare to Run the Java Programs

Before you can run the Java programs, you must set up your Java environment and compile the programs.

1. Check that the version of Java you are using is 1.5 or higher. You can execute the following in a command window to check the version of Java.

```
>java -version
```

2. Add `ORACLE_HOME\jdk\bin\` to your `PATH` variable before the paths of any other Java versions. `ORACLE_HOME` is the full path to the Oracle home directory.
3. Add the following Data Mining JAR files to your Windows `CLASSPATH`:

```
ORACLE_HOME\RDBMS\jlib\jdm.jar
ORACLE_HOME\RDBMS\jlib\ojdm_api.jar
ORACLE_HOME\RDBMS\jlib\xdb.jar
ORACLE_HOME\jdbc\lib\ojdbc5.jar
ORACLE_HOME\oc4j\j2ee\home\lib\connector.jar
ORACLE_HOME\jlib\orai18n.jar
ORACLE_HOME\jlib\orai18n-mapping.jar
ORACLE_HOME\lib\xmlparserv2.jar
```

4. Compile the programs listed in [Figure 7–2](#). To use the `JAVAC` executable, open a command window and go to `\RDBMS\demo` in Oracle home.

```
>javac program_name.java
```

For example:

```
>javac dmbdemo.java
```

If `JAVAC` is not found, then check the value of the `PATH` variable.

Run the Java Programs

You can run the Java programs from the operating system prompt with a command like this:

```
>java program_name host_name:port_number:database_identifier user password
```

List the Sample Models

The mining models created by the sample programs can be viewed with a query like the one shown in [Example 7–1](#).

Example 7–1 Sample Data Mining Models

```
SQL> SELECT model_name, mining_function, algorithm FROM user_mining_models
       ORDER BY model_name;
```

MODEL_NAME	MINING_FUNCTION	ALGORITHM
ABNMODEL_JDM	CLASSIFICATION	ADAPTIVE_BAYES_NETWORK
ABN_SH_CLAS_SAMPLE	CLASSIFICATION	ADAPTIVE_BAYES_NETWORK
AIMODEL_JDM	ATTRIBUTE_IMPORTANCE	MINIMUM_DESCRIPTION_LENGTH
AI_SH_SAMPLE	ATTRIBUTE_IMPORTANCE	MINIMUM_DESCRIPTION_LENGTH
APMODEL_JDM	CLASSIFICATION	NAIVE_BAYES
ARMODEL_JDM	ASSOCIATION_RULES	APRIORI_ASSOCIATION_RULES
AR_SH_SAMPLE	ASSOCIATION_RULES	APRIORI_ASSOCIATION_RULES
AR_SH_SAMPLE_STR_XNAL	ASSOCIATION_RULES	APRIORI_ASSOCIATION_RULES
AR_SH_SAMPLE_XNAL_SVAL	ASSOCIATION_RULES	APRIORI_ASSOCIATION_RULES
DT_SH_CLAS_SAMPLE	CLASSIFICATION	DECISION_TREE
GLMCMODEL_JDM	CLASSIFICATION	GENERALIZED_LINEAR_MODEL
GLMC_SH_CLAS_SAMPLE	CLASSIFICATION	GENERALIZED_LINEAR_MODEL
GLMRMODEL_JDM	REGRESSION	GENERALIZED_LINEAR_MODEL
GLMR_SH_REGR_SAMPLE	REGRESSION	GENERALIZED_LINEAR_MODEL
KMMODEL_JDM	CLUSTERING	KMEANS

KM_SH_CLUS_SAMPLE	CLUSTERING	KMEANS
NBEXPIMPMODEL_JDM	CLASSIFICATION	NAIVE_BAYES
NBMODEL_JDM	CLASSIFICATION	NAIVE_BAYES
NB_SH_CLAS_SAMPLE	CLASSIFICATION	NAIVE_BAYES
NMFMODEL_JDM	FEATURE_EXTRACTION	NONNEGATIVE_MATRIX_FACTOR
NMF_SH_SAMPLE	FEATURE_EXTRACTION	NONNEGATIVE_MATRIX_FACTOR
OCMODEL_JDM	CLUSTERING	O_CLUSTER
OC_SH_CLUS_SAMPLE	CLUSTERING	O_CLUSTER
SVMCMODEL_JDM	CLASSIFICATION	SUPPORT_VECTOR_MACHINES
SVMC_SH_CLAS_SAMPLE	CLASSIFICATION	SUPPORT_VECTOR_MACHINES
SVMOMODEL_JDM	CLASSIFICATION	SUPPORT_VECTOR_MACHINES
SVMO_SH_CLAS_SAMPLE	CLASSIFICATION	SUPPORT_VECTOR_MACHINES
SVMRMODEL_JDM	REGRESSION	SUPPORT_VECTOR_MACHINES
SVMR_SH_REGR_SAMPLE	REGRESSION	SUPPORT_VECTOR_MACHINES
TREEMODEL_JDM	CLASSIFICATION	DECISION_TREE
TXTNMFMODEL_JDM	FEATURE_EXTRACTION	NONNEGATIVE_MATRIX_FACTOR
TXTSVMODEL_JDM	CLASSIFICATION	SUPPORT_VECTOR_MACHINES
T_NMF_SAMPLE	FEATURE_EXTRACTION	NONNEGATIVE_MATRIX_FACTOR
T_SVM_CLAS_SAMPLE	CLASSIFICATION	SUPPORT_VECTOR_MACHINES

The model names distinguish the models created by the Java programs from those created by the PL/SQL programs. The models created by the Java programs have "_JDM" appended to the name.

The PL/SQL Programs

The PL/SQL sample programs illustrate the use of the `DBMS_DATA_MINING` package for creating models and the `DBMS_DATA_MINING_TRANSFORM` package for performing transformations on the mining data.

See Also:

- *Oracle Database PL/SQL Packages and Types Reference* for syntax of the PL/SQL API
- *Oracle Data Mining Application Developer's Guide* for information on the use of the APIs
- *Oracle Database SQL Language Reference* for syntax of the SQL functions for model scoring

PL/SQL Programs: Algorithms

The PL/SQL programs are presented by algorithm in [Table 7-1](#).

Table 7-1 Algorithms in PL/SQL Sample Programs

Program File	Algorithm	Mining Function or Task
<code>dmaidemo.sql</code>	Minimum Descriptor Length	Attribute Importance
<code>dmardemo.sql</code>	Apriori	Association
<code>dmtddemo.sql</code>	Decision Tree	Classification
<code>dmtdxvlddemo.sql</code>	Decision Tree (cross validation)	Classification
<code>dmglcdem.sql</code>	Binary Logistic Regression (GLM)	Classification
<code>dmglrdem.sql</code>	Multivariate Linear Regression (GLM)	Regression
<code>dmkmdemo.sql</code>	<i>k</i> -Means	Clustering

Table 7–1 (Cont.) Algorithms in PL/SQL Sample Programs

Program File	Algorithm	Mining Function or Task
dmbdemo.sql	Naive Bayes	Classification
dmmdemo.sql	Non-Negative Matrix Factorization	Feature Extraction
dmocdemo.sql	O-Cluster	Clustering
dmsvcdem.sql	Support Vector Machine	Classification
dmsvodem.sql	Support Vector Machine	Anomaly Detection
dmsvrдем.sql	Support Vector Machine	Regression
dmtxtfe.sql	Term extraction using Oracle Text	Text transformation for mining
dmtxtnmf.sql	Non-Negative Matrix Factorization	Text mining using NMF
dmtxtsvm.sql	Support Vector Machine	Text mining using SVM

PL/SQL Programs: Mining Functions

The PL/SQL sample programs are presented by mining function in [Table 7–2](#). For detailed descriptions of the sample programs, see the comments in the source code.

Table 7–2 Mining Functions of PL/SQL Sample Programs

Mining Function	Description
Classification	<p>The classification programs demonstrate various preprocessing techniques and perform the following steps:</p> <ul style="list-style-type: none"> ▪ Build a classification model using training data ▪ Display model details and settings ▪ Test the model by applying the model on the test data ▪ Compute test metrics, such as confusion matrix, lift, and ROC ▪ Apply the model on the scoring data ▪ Present apply results ▪ Present ranked apply results, influenced by a cost matrix <p>dmbdemo.sql illustrates Naive Bayes. dmdtdemo.sql illustrates Decision Tree. dmsvcdem.sql illustrates SVM classification. dmglcdem.sql illustrates GLM classification (binary logistic regression)</p> <p>The dmdtxvlddemo.sql program demonstrates cross-validation techniques for decision tree based-classification. With minor modifications, this program can be used to perform cross validation using other models/algorithms.</p>
Regression	<p>dmsvrдем.sql uses different test metrics, but otherwise performs most of the same steps used in the classification programs. Selected attributes of the input data are preprocessed (normalized).</p> <p>NOTE: dmsvrдем.sql illustrates the use of Automatic Data Preparation.</p> <p>dmglrdem.sql illustrates GLM regression (multivariate linear regression)</p>
Anomaly Detection	<p>dmsvodem.sql illustrates one-class SVM</p>
Association	<p>dmardemo.sql builds an association model and presents frequent itemsets and association rules as output.</p>

Table 7–2 (Cont.) Mining Functions of PL/SQL Sample Programs

Mining Function	Description
Clustering	<p><code>dmkmdemo.sql</code> (<i>k</i>-Means) and <code>dmocdemo.sql</code> (0-Cluster) build clustering models and present cluster details, such as rules, centroid, and histogram for each cluster as output. The models are scored, and the probabilities associated with each cluster are returned as output. Selected attributes of the input data are preprocessed.</p> <p>NOTE: <code>dmkmdemo.sql</code> illustrates the use of Automatic Data Preparation.</p>
Feature extraction	<p><code>dmnmdemo.sql</code> builds a feature extraction model and presents model details as the output. The model is scored, and each feature ID is associated with a probability. Selected attributes of the input data are preprocessed (normalized).</p>
Attribute importance	<p><code>dmaidemo.sql</code> builds an attribute importance model and presents a list of important attributes as the output of model details. Selected attributes of the input data are preprocessed (binned).</p>

PL/SQL Text Mining Programs

Three sample programs illustrate the process of text mining using PL/SQL. One program illustrates the preprocessing that is required to transform the text for mining. The other two programs build models that use the transformed text.

See Also:

- *Oracle Data Mining Concepts* for information about text mining with Oracle Data Mining
- *Oracle Data Mining Application Developer's Guide* for details about text transformation in PL/SQL

The PL/SQL sample text mining programs are:

- `dmtxtfe.sql` — Illustrates the process of term extraction that prepares text for mining
- `dmtxtnmf.sql` — Creates a text mining model using the Non-Negative Matrix Factorization algorithm
- `dmtxtsvm.sql` — Creates a text mining model using SVM classification

The Java Programs

The Java demos illustrate the features of the Oracle Data Mining Java API, which implements Oracle-specific extensions to the Java Data Mining (JDM) 1.0.1.1 standard. The sample Java programs demonstrate all the Data Mining algorithms as well as data transformation techniques, predictive analytics, export/import, and text mining.

Note: The Oracle Data Mining Java API is deprecated in this release.

Oracle recommends that you not use deprecated features in new applications. Support for deprecated features is for backward compatibility only

See Also: ■

- *Oracle Data Mining Application Developer's Guide* for information on the Oracle Data Mining Java API
- *Oracle Data Mining Java API Reference* (javadoc)

Java Programs: Algorithms

The Java programs are presented by algorithm in [Table 7–3](#).

Table 7–3 Algorithms in Java Sample Programs

Program File	Algorithm	Mining Function or Task
dmaidemo.java	Minimum Description Length	Attribute importance
dmapplydemo.java	Naive Bayes	Scoring methods
dmardemo.java	Apriori	Association
dmexpimpdemo.java	export/import	Model Export/Import
dmglcdemo.java	Binary Logistic Regression (GLM)	Classification
dmglrdemo.java	Multivariate Linear Regression (GLM)	Regression
dmkmdemo.java	<i>k</i> -Means	Clustering
dmnbdemo.java	Naive Bayes	Classification
dmnmdemo.java	Non-Negative Matrix Factorization	Feature extraction
dmocdemo.java	O-Cluster	Clustering
dmppademo.java	Automated predict and explain	Predictive Analytics
dmsvcdemo.java	Support Vector Machine	Classification
dmsvodemo.java	Support Vector Machine (one class)	Classification
dmsvrdemo.java	Support Vector Machine	Regression
dmtreedemo.java	Decision Tree	Classification
dmtxtnmfdemo.java	Non-Negative Matrix Factorization	Text mining with NMF
dmtxtsvmdemo.java	Support Vector Machine	Text mining with SVM classification
dmxfdemo.java	Binning, clipping, and normalization	Data Transformations

Java Programs: Mining Functions

The Java sample programs are presented by mining function in [Table 7–4](#). For detailed descriptions of the sample programs, see the comments in the source code.

Table 7–4 Mining Functions of the Java Sample Programs

Mining Function or Task	Description
Classification	<p>The classification programs demonstrate various preprocessing techniques and perform the following steps:</p> <ul style="list-style-type: none"> ■ Build a classification model using training data ■ Display model details and settings ■ Test the model by applying the model on the test data ■ Compute test metrics, such as confusion matrix, lift, and ROC ■ Apply the model on the scoring data ■ Present apply results ■ Present ranked apply results, influenced by a cost matrix <p>The <code>dmapplydemo.java</code> program demonstrates several ways of applying a Naive Bayes model.</p> <p><code>dmglcdemo.java</code> illustrates GLM classification (binary logistic regression)</p>
Regression	<p><code>dmsvrdemo.java</code> uses different test metrics, but otherwise performs most of the same steps used in the classification programs. Selected attributes of the input data are preprocessed (normalized).</p> <p><code>dmglrdemo.java</code> illustrates GLM regression (multivariate linear regression)</p>
Association	<p><code>dmardemo.java</code> builds an association model and presents frequent itemsets and association rules as output. Selected attributes of the input data are preprocessed (binned).</p>
Clustering	<p><code>dmkmdemo.java</code> (<i>k</i>-Means) and <code>dmocdemo.java</code> (0-Cluster) build clustering models and present cluster details, such as rules, centroid, and histogram for each cluster as output. The models are scored, and the probabilities associated with each cluster are returned as output. Selected attributes of the input data are preprocessed (normalized).</p>
Feature extraction	<p><code>dmnmdemo.java</code> builds a feature extraction model and presents model details as the output. The model is scored, and each feature ID is associated with a probability. Selected attributes of the input data are preprocessed (normalized).</p>
Attribute importance	<p><code>dmaidemo.java</code> builds an attribute importance model and presents a list of important attributes as the output of model details. Selected attributes of the input data are preprocessed (binned).</p>
Data transformations	<p><code>dmxfdemo.java</code> demonstrates binning, clipping, and normalization transformations.</p>
Predictive Analytics	<p><code>dmypademo.java</code> demonstrates PREDICT, EXPLAIN, and PROFILE functions.</p>
Model import/export	<p><code>dmexpimpdemo.java</code> builds a Naive Bayes model, exports it to a dump file, then imports it from the dump file.</p>

Java Text Mining Programs

Two Java programs illustrate the process of text mining. One builds a feature extraction model, the other builds a classification model.

See Also:

- *Oracle Data Mining Concepts* for information about text mining with Oracle Data Mining
- *Oracle Data Mining Application Developer's Guide* for details about text transformation in Java

The Java text mining programs both use the `dmtxtnmfdemo.java` interface to transform the text for mining. The programs are as follows:

- `dmtxtnmf.sql` — Creates a text mining model using the Non-Negative Matrix Factorization algorithm
- `dmtxtsvmdemo.java` — Creates a text mining model using SVM classification

The Sample Data

The `dmsh.sql` script creates views, tables, and indexes in the user's schema. The views define columns of customer data from tables in the `SH` schema. This data is used by the Data Mining sample programs. The tables reference the same columns in `SH`, but they include an extra `COMMENTS` column for text mining. The indexes are used to extract terms from the text in the `COMMENTS` column and build a nested table column.

Customer Data for Data Mining

Views in the data mining user's schema define columns of data from the `CUSTOMERS`, `SALES`, `PRODUCTS`, `COUNTRIES`, and `SUPPLEMENTARY_DEMOGRAPHICS` tables in the `SH` schema. You can list these views with the following SQL statements.

```
SQL>CONNECT dmuser
Enter password: password
SQL>SELECT view_name FROM user_views;
```

The views are listed in [Table 7-5](#).

Table 7-5 Views Used by the Data Mining Sample Programs

View Name	Description
<code>MINING_DATA_APPLY_STR_V</code>	Scoring data for o-cluster
<code>MINING_DATA_BUILD_STR_V</code>	Training data for o-cluster
<code>MINING_DATA_APPLY_V</code>	Scoring data for data mining (not text mining)
<code>MINING_DATA_BUILD_V</code>	Training data for data mining (not text mining)
<code>MINING_DATA_TEST_V</code>	Test data for data mining (not text mining)
<code>MARKET_BASKET_V</code>	Data for association rules
<code>MINING_DATA_ONE_CLASS_V</code>	Data for one-class SVM

You can see the references to tables in `SH` by listing the view definitions. The definition of the view `MINING_DATA_BUILD_V` is shown as follows.

```
SQL> set long 1000000
SQL> set longc 100000
SQL> set pagesize 100
SQL> SELECT text FROM all_views WHERE
      owner='dmuser3'AND view_name='mining_data_build_v';
```

```

SELECT a.CUST_ID, a.CUST_GENDER, 2003-a.CUST_YEAR_OF_BIRTH AGE,
       a.CUST_MARITAL_STATUS, c.COUNTRY_NAME, a.CUST_INCOME_LEVEL,
       b.EDUCATION, b.OCCUPATION, b.HOUSEHOLD_SIZE, b.YRS_RESIDENCE,
       b.AFFINITY_CARD, b.BULK_PACK_DISKETTES, b.FLAT_PANEL_MONITOR,
       b.HOME_THEATER_PACKAGE, b.BOOKKEEPING_APPLICATION,
       b.PRINTER_SUPPLIES, b.Y_BOX_GAMES, b.OS_DOC_SET_KANJI
FROM   sh.customers a,
       sh.supplementary_demographics b,
       sh.countries c
WHERE  a.CUST_ID = b.CUST_ID AND a.country_id = c.country_id
AND    a.cust_id between 101501 and 103000

```

The views are used to build, test, and score the sample models. Each view has a CUSTOMER_ID column, which is the case ID, and an AFFINITY_CARD column, which is the target used by the predictive models. Most of the views provide data for 1500 customers (1500 rows). The view used by the One-Class SVM model has data for 940 customers.

The columns of training data in the MINING_DATA_BUILD_V view are as follows.

```
SQL> DESCRIBE mining_data_build_v
```

CUST_ID	NOT NULL	NUMBER
CUST_GENDER	NOT NULL	CHAR(1)
AGE		NUMBER
CUST_MARITAL_STATUS		VARCHAR2(20)
COUNTRY_NAME	NOT NULL	VARCHAR2(40)
CUST_INCOME_LEVEL		VARCHAR2(30)
EDUCATION		VARCHAR2(21)
OCCUPATION		VARCHAR2(21)
HOUSEHOLD_SIZE		VARCHAR2(21)
YRS_RESIDENCE		NUMBER
AFFINITY_CARD		NUMBER(10)
BULK_PACK_DISKETTES		NUMBER(10)
FLAT_PANEL_MONITOR		NUMBER(10)
HOME_THEATER_PACKAGE		NUMBER(10)
BOOKKEEPING_APPLICATION		NUMBER(10)
PRINTER_SUPPLIES		NUMBER(10)
Y_BOX_GAMES		NUMBER(10)
OS_DOC_SET_KANJI		NUMBER(10)

Market Basket Data for Association Rules

The association demos use the MARKET_BASKET_V data set, which includes columns of products from the PRODUCTS table and the CUSTOMER_ID column from the CUSTOMERS table in SH. The columns of the MARKET_BASKET_V view are as follows.

```
SQL> DESCRIBE market_basket_v
```

CUST_ID	NOT NULL	NUMBER
EXTENSION_CABLE		NUMBER
FLAT_PANEL_MONITOR		NUMBER
CD_RW_HIGH_SPEED_5_PACK		NUMBER
ENVOY_256MB_40GB		NUMBER
ENVOY_AMBASSADOR		NUMBER
EXTERNAL_8X_CD_ROM		NUMBER
KEYBOARD_WRIST_REST		NUMBER
SM26273_BLACK_INK_CARTRIDGE		NUMBER
MOUSE_PAD		NUMBER
MULTIMEDIA_SPEAKERS_3INCH		NUMBER
OS_DOC_SET_ENGLISH		NUMBER

```
SIMM_16MB_PCMCIAII_CARD      NUMBER
STANDARD_MOUSE               NUMBER
```

Customer Data for Text Mining

The text mining demos use the same customer data from tables in SH, but they include either an extra text column or a collection type column. The collection type is a nested table of type DM_NESTED_NUMERICALS.

Tip: The process of extracting terms from a text column into a nested table column is described in *Oracle Data Mining Application Developer's Guide*.

You can list these tables with the following SQL statements.

```
SQL>CONNECT dmuser
Enter password: password
SQL>SELECT table_name FROM user_tables WHERE table_name LIKE '%MINING%';
```

The text mining tables are listed in [Table 7-6](#).

Table 7-6 Tables Used by the Text Mining Sample Programs

Table Name	Description
MINING_APPLY_NESTED_TEXT	Apply table with COMMENTS column as DM_NESTED_NUMERICALS
MINING_BUILD_NESTED_TEXT	Build table with COMMENTS column as DM_NESTED_NUMERICALS
MINING_TEST_NESTED_TEXT	Test table with COMMENTS column as DM_NESTED_NUMERICALS
MINING_APPLY_TEXT	Apply table with COMMENTS column as VARCHAR2 (4000)
MINING_BUILD_TEXT	Build table with COMMENTS column as VARCHAR2 (4000)
MINING_TEST_TEXT	Test table with COMMENTS column as VARCHAR2 (4000)

In the MINING_BUILD_TEXT, MINING_TEST_TEXT, and MINING_APPLY_TEXT tables, the COMMENTS column is of type VARCHAR2 (4000).

```
SQL> DESCRIBE mining_build_text
Name                               Null?    Type
-----
CUST_ID                             NOT NULL NUMBER
CUST_GENDER                          NOT NULL CHAR(1)
AGE                                   NUMBER
CUST_MARITAL_STATUS                  VARCHAR2(20)
COUNTRY_NAME                         NOT NULL VARCHAR2(40)
CUST_INCOME_LEVEL                    VARCHAR2(30)
EDUCATION                             VARCHAR2(21)
OCCUPATION                           VARCHAR2(21)
HOUSEHOLD_SIZE                       VARCHAR2(21)
YRS_RESIDENCE                        NUMBER
AFFINITY_CARD                        NUMBER(10)
BULK_PACK_DISKETTES                  NUMBER(10)
FLAT_PANEL_MONITOR                   NUMBER(10)
HOME_THEATER_PACKAGE                 NUMBER(10)
BOOKKEEPING_APPLICATION              NUMBER(10)
PRINTER_SUPPLIES                     NUMBER(10)
Y_BOX_GAMES                          NUMBER(10)
OS_DOC_SET_KANJI                     NUMBER(10)
COMMENTS                             VARCHAR2(4000)
```

In the MINING_*_NESTED_TEXT tables, the COMMENTS column is of type DM_NESTED_NUMERICALS.

```
SQL> DESCRIBE mining_build_nested_text
```

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_GENDER	NOT NULL	CHAR(1)
AGE		NUMBER
CUST_MARITAL_STATUS		VARCHAR2(20)
COUNTRY_NAME	NOT NULL	VARCHAR2(40)
CUST_INCOME_LEVEL		VARCHAR2(30)
EDUCATION		VARCHAR2(21)
OCCUPATION		VARCHAR2(21)
HOUSEHOLD_SIZE		VARCHAR2(21)
YRS_RESIDENCE		NUMBER
AFFINITY_CARD		NUMBER(10)
BULK_PACK_DISKETTES		NUMBER(10)
FLAT_PANEL_MONITOR		NUMBER(10)
HOME_THEATER_PACKAGE		NUMBER(10)
BOOKKEEPING_APPLICATION		NUMBER(10)
PRINTER_SUPPLIES		NUMBER(10)
Y_BOX_GAMES		NUMBER(10)
OS_DOC_SET_KANJI		NUMBER(10)
COMMENTS		DM_NESTED_NUMERICALS

Verifying Your Data Mining Installation

This appendix provides a set of SQL and PL/SQL commands that test whether or not the basic capabilities of data mining are present in the database. You can execute these commands directly in the database without downloading or installing any additional software.

Note: These commands create and score several data mining models. These models *do not* perform meaningful data mining. They simply show whether or not model creation and scoring operations are working properly.

This appendix includes the following topics:

- [Connect to the Database](#)
- [Create a Script or Execute the Commands Individually](#)
- [Command Output](#)

Connect to the Database

In SQL*Plus or SQL Developer, connect to the database as the system user. If you are not sure how to start these tools, see "[Connect Application Development Tools](#)" on page 3-1.

```
CONNECT sys / as sysdba
Enter password: <sys_password>
```

Insure that the `USERS` and `TEMP` tablespaces are present in the database. These tablespaces are typically included by default and are always included in the starter database created during installation (See "[Install Oracle Database](#)" on page 2-1). You can check with a simple query.

```
select tablespace_name from dba_tablespaces;
```

Create a Script or Execute the Commands Individually

You can create a script by copying the commands in [Example A-1](#) into a text file, saving the file with a `.SQL` extension, and invoking the script from the SQL command line.

```
SQL> @ script_name.SQL
```

Alternatively, you can copy each command from [Example A-1](#) to the SQL command line and execute it individually.

Example A-1 Commands To Verify Data Mining Installation

```
-----
-- create a data mining user
-----
CREATE USER dm IDENTIFIED BY <dm_password>
        DEFAULT TABLESPACE users
        TEMPORARY TABLESPACE temp
        QUOTA 20M ON users;

-- grant minimum privileges
GRANT create mining model TO dm;
GRANT create table TO dm;
GRANT create view TO dm;
GRANT create session TO dm;

-----
-- create a simple table for mining
-----
CONNECT dm/dm;
CREATE TABLE mt (p NUMBER, t NUMBER);
INSERT INTO mt VALUES (1,1);
INSERT INTO mt VALUES (2,2);
COMMIT;

-----
-- create models
-----
EXEC dbms_data_mining.create_model('MAI', 'ATTRIBUTE_IMPORTANCE', 'MT', null, 'T');
EXEC dbms_data_mining.create_model('MAR', 'ASSOCIATION', 'MT', 'T');
EXEC dbms_data_mining.create_model('MNB', 'CLASSIFICATION', 'MT', null, 'T');
EXEC dbms_data_mining.create_model('MSVM', 'REGRESSION', 'MT', null, 'T');
EXEC dbms_data_mining.create_model('MKM', 'CLUSTERING', 'MT', null);
EXEC dbms_data_mining.create_model('MNMF', 'FEATURE_EXTRACTION', 'MT', null);

-- check the models
SELECT model_name, algorithm FROM user_mining_models;

-----
-- score models
-----
SELECT p,t,PREDICTION(MNB USING *) pred_nb,
       p,t,prediction(MSVM USING *) pred_svm,
       p,t,cluster_id(MKM USING *) clus,
       p,t,feature_id(MNMF USING *) feat
FROM mt;

-----
-- drop the models and data
-----
EXEC dbms_data_mining.drop_model('MAI');
EXEC dbms_data_mining.drop_model('MAR');
EXEC dbms_data_mining.drop_model('MNB');
EXEC dbms_data_mining.drop_model('MSVM');
EXEC dbms_data_mining.drop_model('MKM');
EXEC dbms_data_mining.drop_model('MNMF');
```



```
DROP TABLE mt PURGE;
```

Command Output

The output of the commands in [Example A-1](#) is as follows.

```
User created.
```

```
Grant succeeded.
```

```
Grant succeeded.
```

```
Grant succeeded.
```

```
Grant succeeded.
```

```
Connected.
```

```
Table created.
```

```
1 row created.
```

```
1 row created.
```

```
Commit complete.
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
MODEL_NAME          ALGORITHM
-----
MAI                  MINIMUM_DESCRIPTION_LENGTH
MAR                  APRIORI_ASSOCIATION_RULES
MKM                  KMEANS
MNB                  NAIVE_BAYES
MNMF                 NONNEGATIVE_MATRIX_FACTOR
MSVM                 SUPPORT_VECTOR_MACHINES
```

```
6 rows selected.
```

```

      P      T  PRED_NB      P      T  PRED_SVM      P
-----
      T      CLUS      P      T      FEAT
-----
      1      1      1      1      1  1.48008198      1
      1      3      1      1      1
      2      2      2      2      2  1.51991802      2
      2      2      2      2      1
```

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

Table dropped.

A

Adaptive Bayes Network
 See deprecated features
ALL_MINING_MODEL_ATTRIBUTES, xi, 5-1
ALL_MINING_MODEL_SETTINGS, xi, 5-1, 5-3
ALL_MINING_MODELS, xi, 5-1
ALL_OBJECTS, xi
ALTER ANY MINING MODEL, 4-3
ALTER MINING MODEL, 4-4
anomaly detection, 7-7
Apriori, 7-6, 7-9
association, 7-7, 7-9, 7-10, 7-12
attribute importance, 7-6, 7-8, 7-9, 7-10
attributes, 5-2
AUDIT, 4-3, 5-4 to 5-6
AUDIT_TRAIL parameter, 5-4
Automatic Data Preparation, 7-7 to 7-8

B

binning, 7-9, 7-10
BLAST
 See desupported features

C

catalog views, xi, 5-1
chopt utility, 2-9
classification, 7-6, 7-7, 7-9, 7-10
CLASSPATH, 7-5
clipping, 7-9, 7-10
clustering, 7-7, 7-8, 7-9, 7-10
COMMENT, 4-3, 5-2, 5-5
COMMENT ANY MINING MODEL, 4-3
COMMENT ANY MODEL, 5-4
COMPATIBLE parameter, 6-3
connection name, 3-2
cost matrix, 4-4
CREATE ANY DIRECTORY, 5-7
CREATE ANY MINING MODEL, 4-3
CREATE MINING MODEL, x, 1-2, 4-2, 4-3

D

data dictionary views, xi, 5-1
Data Mining option, 2-1

 enable or disable, 2-9
data mining user, 2-1, 2-11, 3-1, 4-1
Database Companion
 See Database Examples
Database Configuration Assistant, xi, 1-2, 7-2
Database Examples
 installation, 2-10
 sample programs, 7-1
database file location, 2-5
database tuning, vi
Database Upgrade Assistant, 6-2
DBA
 database tuning, vi
 production database, v, 2-1
DBA_AUDIT_OBJECT, 5-6
DBA_MINING_MODEL_ATTRIBUTES, 5-1
DBA_MINING_MODEL_SETTINGS, 5-1
DBA_MINING_MODELS, 5-1, 6-3
DBA_OBJ_AUDIT_OPTS, 5-6
DBA_OBJECTS, xi
DBA_REGISTRY, 6-3
DBMS_DATA_MINING, 5-2, 7-6
DBMS_DATA_MINING_TRANSFORM, 7-6
DBMS_DATA_MINING.DROP_MODEL, 6-4
Decision Tree, 7-6, 7-9
demo programs
 See sample programs
deprecated features, ix, xi, 1, 8
desupported features, xi
DHCP, 2-2
directory objects, 5-7
DM\$P_MODEL_EXPIMP_TEMP table, 5-8
DM\$P_MODEL_IMPORT_TEMP table, 5-8
DM\$P_MODEL_TABKEY_TEMP table, 5-8
DM_USER_MODELS, xi
DMEIDMSYS, 6-3
dmshgrants.sql, 1-4, 5-10, 7-2
dmsh.sql, 7-2, 7-11
DMSYS, x, xi, 6-3
downgrading, 6-1, 6-4
DROP ANY MINING MODEL, 4-3
DSS, v

E

Enterprise Edition, 2-1, 2-5

EXPDP, 5-6
EXPLAIN, 2-11, 2-13, 7-10
exporting, 2-13, 4-2, 5-6 to 5-8, 6-2, 7-9, 7-10

F

feature extraction, 7-7, 7-8, 7-9, 7-10

G

Generalized Linear Models, 7-6, 7-7, 7-10
GET_DEFAULT_SETTINGS, xi
GET_MODEL_DETAILS, 5-2
GET_MODEL_SETTINGS, xi
GET_MODEL_SIGNATURE, xi
GLM
 See Generalized Linear Models
global database name, 2-5, 3-3

I

IMPDP, 5-6
importing, 2-13, 4-2, 5-6 to 5-9, 6-2, 7-9, 7-10
INIT.ORA, vi
installation
 Data Mining sample programs, 7-1
 Oracle Client, 3-4
 Oracle Data Miner, 2-11
 Oracle Database, 2-1
 Oracle Database Examples, 2-10
 Oracle Spreadsheet Add-In for Predictive Analytics, 2-11

J

Java API, ix, 6-1, 1, 8
 See also deprecated features
Java API sample programs, 7-8
Java API text mining, 7-10
JDM 1.0 standard, 7-8
JSR-73 specification, 7-8

K

k-Means, 7-6, 7-9

L

linear regression, 7-6, 7-7, 7-9
logistic regression, 7-6, 7-7, 7-9
loopback adapter, 2-2

M

market basket data, 7-12
memory tuning parameters, vi
Microsoft Excel, 1-4, 2-11, 2-12, 3-12
Microsoft Windows, 2-1
Minimum Description Length, 7-6, 7-9
mining models
 about, 5-1

adding a comment, 5-4
applying, 4-4
auditing, 5-4
changing the name, 4-4
listing, 5-1
object privileges, 4-4
 See also scoring
 settings, 5-3
 upgrading, 6-1
 viewing, 4-4
model details, 5-2
model signature, xi, 5-2

N

Naive Bayes, 7-7, 7-9
Net Configuration Assistant, 2-11, 3-8
Net Service Name, 2-12, 3-8
new features, ix
NMF
 See Non-Negative Matrix Factorization
Non-Negative Matrix Factorization, 7-7, 7-8, 7-9, 7-11
normalization, 7-9, 7-10

O

object privileges, 4-4
O-Cluster, 7-7, 7-9
OLTP, v
One-Class SVM, 7-7
Oracle base directory, 2-5, 2-11
Oracle By Example, 3
Oracle Client
 connect, 3-3
 install, 3-4
 required for Spreadsheet Add-In, 2-12
Oracle Data Miner, ix, 3, 3-1, 2, 3-3, 4
Oracle Data Pump, 5-6
Oracle Database
 Enterprise Edition, 2-1, 2-5
 install, 2-1
 local, 3-2
 remote, 3-3
Oracle home directory, 2-5, 2-10, 2-11, 2-13, 3-6, 7-2, 7-4, 7-5
Oracle Net, 3-3
Oracle Objects for OLE, 2-11
Oracle services, 2-2
Oracle Spreadsheet Add-In for Predictive Analytics
 install, 2-12
 Net Service Name, 3-3, 3-8
 start, 2-13
Oracle Text, 7-7
ORACLE_BASE, 2-5
ORACLE_HOME, vii

P

parallel execution, vi
performance, vi

PGA_AGGREGATE_TARGET, vi
PL/SQL API, 3-1, 6-1
PL/SQL sample programs, 7-6
PL/SQL text mining, 7-8
PMML, ix, 5-11
port number, 3-2
PREDICT, 2-11, 2-13, 7-10
predictive analytics, 2-11, 3-3, 7-8, 7-9, 7-10
privileges for data mining, x, xi, 2-1, 3-1, 4-1, 5-7, 7-1
privileges for exporting and importing, 5-7
privileges for running sample programs, 7-1, 7-2
production database, v, 2-1
PROFILE, 2-11, 2-13, 7-10

R

regression, 7-6, 7-7, 7-9, 7-10

S

sample mining models, 7-5
sample programs, 2-10, 7-1
 configuration scripts, 4-3, 7-1, 7-2
 data used by, 7-11
 find, 7-2
 install, 7-1
 Java, 7-4, 7-8
 PL/SQL, 7-6
 run, 7-4, 7-5
 user, 4-3, 7-1
sample schemas, 7-1
scoring
 batch, v, vi
 java programs, 7-9
 model signature, 5-2
 privileges for, 4-3
 real-time, vi
 single-row, v
Scoring Engine, 6-2
 See also desupported features
security, x, 4-1
SELECT ANY MINING MODEL, 4-3
SELECT MINING MODEL, 4-4
SGA_TARGET, vi
SH schema, 7-1, 7-2, 7-11
shared cursor, vi
SID, 3-3
software location
 See Oracle home directory
Spreadsheet Add-In
 See Oracle Spreadsheet Add-In for Predictive
 Analytics
 start, 2-13
SQL AUDIT, 5-4
SQL COMMENT, 5-4
SQL Developer, 3-1, 3-2, 4
SQL*Plus, 3-1
starter database., 2-5
Support Vector Machine, 7-7, 7-8, 7-9, 7-11
SVM

See Also One-Class SVM
See Support Vector Machine
SYS, x, 2-5
system privileges, 4-3

T

term extraction, 7-7, 7-8
text mining, 7-7, 7-8, 7-9, 7-10, 7-13
transformations, 7-9
tuning, vi

U

upgrading, 6-1
USER_AUDIT_OBJECT, 5-6
USER_MINING_MODEL_ATTRIBUTES, 5-1
USER_MINING_MODEL_SETTINGS, 5-1
USER_MINING_MODELS, 5-1
USER_OBJ_AUDIT_OPTS, 5-6
USER_OBJECTS, xi
users, 2-1, 3-2, 5-7, 7-1, 7-2

