

Using Darwin

Release 3.0.1

Thinking Machines Corporation

The information in this document is subject to change without notice and should not be construed as a commitment by Thinking Machines Corporation. Thinking Machines reserves the right to make changes to any product described herein.

Although the information in this document has been reviewed and is believed to be reliable, Thinking Machines Corporation assumes no liability for errors in this document. Thinking Machines does not assume any liability arising from the application or use of any information or product described herein.

Thinking Machines[®] and Darwin[®] are registered trademarks of Thinking Machines Corporation.

Note: “Darwin” is a registered trademark of Thinking Machines Corporation in the United States.

“Darwin” is a registered trademark of Science in Finance Ltd. in the United Kingdom. Therefore “Darwin” is not available from Thinking Machines Corporation in the United Kingdom. In the United Kingdom,

Thinking Machines Corporation sells its product under the name “LoyaltyStream.”

HP-UX and HP-UX 10.20 are trademarks of Hewlett-Packard Company.

INFORMIX is a trademark of Informix Software, Inc.

InstallShield is a trademark of InstallShield Corporation.

INTERSOLV is a trademark of INTERSOLV, Inc.

Microsoft, Windows, Windows NT, and Windows 95 are trademarks of Microsoft Corporation.

Oracle is a trademark of Oracle Corporation.

Open Windows is a trademark of Sun Microsystems, Inc.

Sun, Solaris, Sun Ultra, Ultra, and Sun Workstation are trademarks of Sun Microsystems, Inc.

All SPARC trademarks are used under license and are trademarks or registered trademarks of

SPARC International, Inc., in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through

X/Open Company, Ltd.

The X Window System is a trademark of the Massachusetts Institute of Technology.

All other product or service names mentioned herein are trademarks or service marks of their respective owners.

Copyright © 1998 by Thinking Machines Corporation. All rights reserved.

Thinking Machines Corporation
16 New England Executive Park
Burlington, Massachusetts 01803
Tel: 781-238-3400 • Fax: 781-238-3420

Contents

About This Manual	vii
Customer Support	xi
1 Before Using Darwin	1
1.1 Before You Start	1
1.2 The Darwin Client and Server	2
2 Overview of the Darwin Data Mining Process	5
2.1 Creating or Opening a Project	5
2.2 Creating or Opening a Dataset and Transforming Datasets	6
2.3 Building a Model	6
2.4 Analyzing the Results	6
2.5 The Data Mining Process for Each Model Type	7
3 The Darwin User Interface	11
3.1 Starting and Ending a Darwin Session	11
3.1.1 Starting a Darwin Session	11
3.1.2 Ending a Darwin Session	13
3.2 The Main Darwin Window	13
3.2.1 The Menu Bar	14
3.2.2 The Toolbar	16
3.2.3 Darwin Workspace	18
3.3 About the Darwin User Interface	18
3.3.1 Customizing Darwin	18
3.3.2 Dialog Boxes and MDI Windows	19
3.3.3 Command Output	19
3.3.4 Naming Objects	20

4 Project Menu	21
4.1 New Project	21
4.2 Open Project	22
4.3 Close	23
4.4 Save	23
4.5 Deleting a Project	23
4.6 New File	24
4.7 Export Table	25
4.8 Display	25
4.9 Graph	26
4.10 Print	27
4.11 Print Preview	28
4.12 Database Connect	28
4.13 Database Disconnect	29
4.14 Stop	29
4.15 Exit	30
5 Edit Menu	31
6 View Menu	33
6.1 Modeling Wizard	33
6.2 Evaluation Wizard	36
6.3 Workflow	38
6.4 Customize	39
6.5 Status Bar, Workspace, Folders, Darwin Logo	39
6.6 Refresh	39
7 Dataset Menu	41
7.1 Create	41
7.1.1 Import Text File	41
7.1.2 Database	42
7.2 Export	43
7.3 Transform	44
7.3.1 Append	45
7.3.2 Explode	45

7.3.3 Merge	45
7.3.4 Missing	46
7.3.5 Normalize	47
7.3.6 Project	47
7.3.7 Randomize	48
7.3.8 Range	48
7.3.9 Replace	49
7.3.10 Sample	50
7.3.11 Select	50
7.3.12 Set Form	50
7.3.13 Split	51
8 Model Menu	53
8.1 Tree Model	53
8.1.1 Create Tree Model	54
8.1.2 Test Tree Model	54
8.1.3 Predict with Tree Model	56
8.2 Net Model	57
8.2.1 Create Net Model	57
8.2.2 Test Net Model	59
8.2.3 Predict with Net Model	60
8.2.4 Continue Training Net Model	61
8.2.5 Perturbation	62
8.3 Match Model	63
8.3.1 Create Match Model	63
8.3.2 Test Match Model	64
8.3.3 Predict with Match Model	65
8.4 Copy	66
8.5 Deleting a Model	66
9 Analysis Menu	67
9.1 Evaluate	68
9.2 Summarize	69
9.3 Frequencies	70
9.4 Performance	71
9.5 Lift	72

10 Options Menu	75
10.1 Advanced	75
10.1.1 Project	76
10.1.2 Datasets	76
10.1.3 Tree	77
10.1.4 Net Build	78
10.1.5 Net Train	79
10.1.6 Match	81
10.1.7 Analysis	81
10.1.8 Setup	82
10.2 Macro	83
10.3 Code Generation	84
11 Window Menu	85
12 Help Menu	87
13 UNIX Utilities	89
13.1 Automatically Creating a Descriptor File	89
13.1.1 Darwin Descriptor Files	89
13.1.2 How to Create Descriptor Files	90
13.1.3 Using darwinDG	90
13.1.4 The darwinDG Command	91
13.1.5 Examples of Using darwinDG	93
13.1.6 How darwinDG Creates Descriptor Files	94
13.2 Converting between Darwin and SAS File Formats	95
13.2.1 Darwin Datasets	95
13.2.2 Using the Conversion Tools	96
13.2.3 Conversions Using darwin2sas and sas2darwin	96
13.2.4 Conversion Examples	97
13.2.5 Conversions Using DBMS/COPY	98
13.2.6 Implementing Custom Conversion Applications	103
A Glossary	105
Index	115

About This Manual

Objectives

Using Darwin describes the Darwin user interface and how to use it. It is a “how-to” book; it tells you which menus to pull down, what commands to click, etc., to execute commands in a Darwin data mining project.

Its companion volume, *Darwin Reference*, introduces data mining concepts, describes the Darwin tools, and explains the process of data mining with Darwin, from formulating the initial business questions through working with the data to developing models and analyzing results. *Darwin Reference* contains background and conceptual material related to the commands and operations performed with Darwin.

Intended Audience

The two books are intended for all users of Darwin software. They assume some familiarity with data mining and basic statistical techniques.

Revision Information

Using Darwin and *Darwin Reference* are new manuals. They are derived from the *Darwin User's Guide*, which accompanied all earlier releases of Darwin. That volume contains both background and theoretical material about data mining and Darwin as well as detailed description of the Darwin user interface and explanations of how to use it. With this release, the two topics are treated in separate manuals: background and theoretical material is contained in *Darwin Reference*, and the mechanics of the user interface are contained in *Using Darwin*.

Organization

Using Darwin is organized as follows:

Chapter 1 Before Using Darwin

Reminds you of what you need to have done before you start, and describes the Darwin client-server design.

- Chapter 2 Overview of the Data Mining Process**
Describes in general terms the steps in a data mining operation.
- Chapter 3 The Darwin User Interface**
Describes how to start and end a Darwin session, and describes the components of the main Darwin window.
- Chapter 4 Project Menu**
Describes the commands on the **Project** menu.
- Chapter 5 Edit Menu**
Describes the commands on the **Edit** menu.
- Chapter 6 View Menu**
Describes the commands on the **View** menu.
- Chapter 7 Dataset Menu**
Describes the commands on the **Dataset** menu.
- Chapter 8 Model Menu**
Describes the commands on the **Model** menu.
- Chapter 9 Analysis Menu**
Describes the commands on the **Analysis** menu.
- Chapter 10 Options Menu**
Describes the commands on the **Options** menu.
- Chapter 11 Window Menu**
Describes the commands on the **Window** menu.
- Chapter 12 Help Menu**
Describes the commands on the **Help** menu.
- Chapter 13 UNIX Operations**
Describes two UNIX commands: Creating a descriptor file automatically and converting to and from SAS file format.
- Appendix A Glossary**
Lists terminology relevant to Darwin and to data mining.

Hardcopy Documentation

The complete Darwin documentation set includes

- *Using Darwin* (this volume). Describes the user interface and provides detailed instructions for using it.
- *Darwin Reference* (companion volume to *Using Darwin*). Introduces data mining and Darwin, provides background and theoretical material on datasets, Darwin tools, and analyses.
- *Darwin Release Notes, Release 3.0.1*. Describes the release and documents any problems or bugs in the software. There are separate release notes for Solaris and for HP-UX.
- For Solaris system administrators: *Darwin Installation and Administration, Release 3.0.1 for Solaris*.
- For HP-UX system administrators: *Darwin Installation and Administration, Release 3.0.1 for HP-UX*.

Darwin Online Help

Darwin includes extensive online help that can be summoned from a list of contents and from Help buttons or the F1 key on dialog windows.

Online Documentation

All the Darwin documentation is available in HTML format at a password-protected site for customers only, accessible from our Web site at <http://www.think.com>. You can get the password from your system administrator.

To view these files, you will need

- Netscape 2.x or later, or
- Internet Explorer 3.01 or later

Notation Conventions

The table below displays the notation conventions observed in this manual.

Convention	Meaning
Boldface	Darwin commands, menu names, and menu items.
Project -> New File	Indicates the path for a command, e.g., on the Project menu, click the New File command.
<code>typewriter</code>	Data fields and values, special characters, etc., examples of files, data, and so on.
<i>italics</i>	Argument names and placeholders in command formats.
% user input system output	In interactive examples, user input is shown in bold typewriter; system output is shown in regular typewriter.

Customer Support

Thinking Machines Customer Support encourages customers to report problems with Darwin and to suggest improvements in our products.

Customer support is available Monday through Friday from 9 a.m. to 5 p.m., Eastern Standard Time.

When reporting a problem, please provide as much information as possible to help us identify it. Please record and pass along as much of the following information as possible:

- The dataset that you used and its characteristics (size, number of fields and records, etc.)
- How the dataset was created (flat file, ORACLE, INFORMIX, versions, etc.)
- The type of model(s) that you used
- Where you were in Darwin (the folder you were in and the menu you were using)
- The three or four steps you performed just prior to the error
- Any error messages generated by either Darwin or the operating system

Because much of this information is collected automatically in a Darwin macro (command log), we recommend that you turn on command logging using the **Macro** selection of the **Options** menu and forward the log to Thinking Machines when you report a bug.

You should also record and communicate the following information:

- The hardware platform
- The release of UNIX that you are using
- The hardware platform that the client is running on
- The version of Windows NT or Windows 95 that you are running
- The mode you were operating in (serial or parallel)
- If parallel, the number of processors you were running on

Please contact Thinking Machines' home office customer support staff:

Internet Electronic Mail:	customer-support@think.com
U.S. Mail:	Thinking Machines Corporation Customer Support 16 New England Executive Park Burlington, Massachusetts 01803
Telephone:	
Within North America	1-800-677-1110 Monday – Friday, 9 a.m. – 5 p.m. EST
Outside North America	+1-781-238-3400 Monday – Friday, 9 a.m. – 5 p.m. EST
Fax:	1-781-238-3420

1 Before Using Darwin

This book describes the Darwin user interface and the mechanics of using it to execute the commands for a Darwin data mining project. It is a “how-to” book — it tells you which menus to pull down, which command buttons to click, which text boxes to fill in with what information, etc., to accomplish a given objective.

Darwin includes extensive online help that can be summoned from a list of contents and from Help buttons or the F1 key on dialog windows.

Introductory material, background information, conceptual matters, and the like are provided in the companion volume, *Darwin Reference*. The two manuals contain many cross-references to each other. You will probably want to have both at hand whenever you are working in Darwin.

This chapter is organized as follows:

- a reminder of what you need to have done before you start a Darwin session
- a description and diagram of the Darwin client and server

1.1 Before You Start

Before you are ready to start using Darwin, there is a lot of preparatory work that you must have done in order to obtain meaningful results from your data mining project.

In broad terms, this means you have formulated your question carefully and appropriately, you have located the relevant data, and you have prepared that data appropriately and made it accessible to a Darwin server by having transferred it to your UNIX system. These topics are addressed in the first few chapters of *Darwin Reference*.

Do not underestimate the importance of formulating your question and preparing your data properly. Time spent on these activities is time well spent, as anyone who has skimmed on them can attest.

1.2 The Darwin Client and Server

Darwin's design is a *client-server* design. The Darwin client runs on a PC running Windows 95 or Windows NT. You interact with the client to execute commands and requests for operations that the client sends to the server. The server performs all data mining operations and sends the results to the client, which displays them.

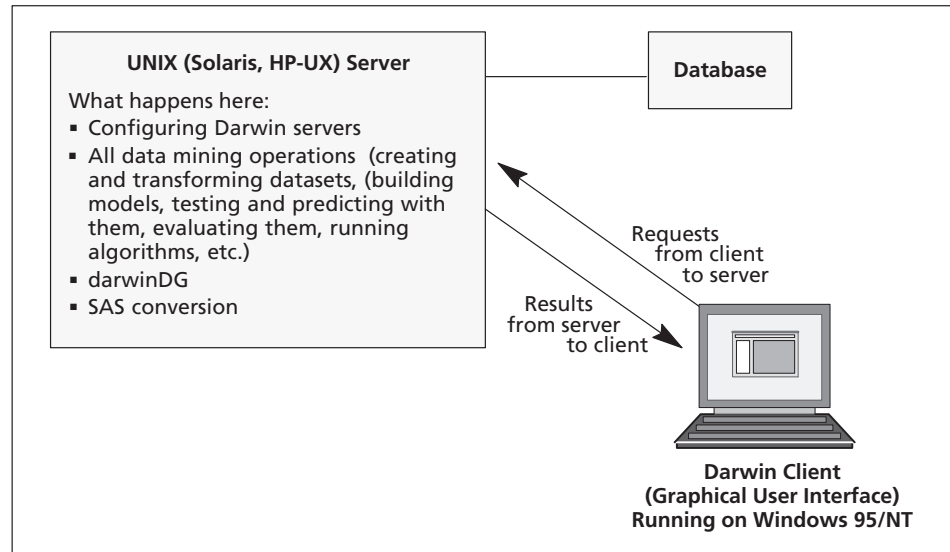


Figure 1. Darwin client and server.

All Darwin servers run on UNIX. There are two basic server types: *serial* and *parallel*. Serial servers run either on a single-processor system or on a single node of a multiprocessor system. Parallel servers run on two or more nodes of a multiprocessor system.

Each system that runs Darwin at your site must have at least one configured Darwin server but may have more than one.

Your system administrator defines the Darwin servers at your site, using the command `darwinconfig`. As part of this process, the system administrator includes a one-line description of each server. This information is displayed in the Description text box of the login window. Decide which server (configuration) among those listed is best suited to your needs, and click on its name to select it.

Note: The term *server* can refer to three different things:

- A *Darwin server* is a *configuration* created by your system administrator.
- In the context of Darwin's client-server architecture, *server* also refers to the part of Darwin where data mining algorithms run.
- The term *server* also refers to the physical machine on which programs or operating systems run.

2 Overview of the Darwin Data Mining Process

This chapter describes in general terms the steps in a Darwin data mining operation. These steps are outlined in more detail in *Darwin Reference*, section 1.3.

The steps are these:

- create or open a project
- create or open a dataset and perform the appropriate transformations
- build a model
- analyze results

These steps are diagrammed in section 2.5 for the three different model types.

2.1 Creating or Opening a Project

All Darwin work takes place within a *project*. After logging in and choosing a server, the first thing you must do is create or open a project. Existing projects are listed in the **Workspace** of the main Darwin window (see section 3.2.3). Until you select or create a project, most of the menu and toolbar items are unavailable.

The project that you create or open becomes the *current* project. All existing projects are listed in the **Workspace**; the current project's name appears in bold.

After you have used Darwin once, Darwin always opens with a project identified as the current project; this is the project you last worked in.

(All Darwin project files reside on UNIX. You access them through your PC, using the Darwin pull-down menus and commands.)

2.2 Creating or Opening a Dataset and Transforming Datasets

Once you select or create a project, you can open an existing dataset or create a new dataset from a dataset descriptor file and a text file or from a database.

The name of the current dataset has a colored icon in the **Workspace** listing, and is the one whose name appears in dialog windows as the source dataset for performing transformations or analysis. You can specify a different dataset by selecting its name from the drop-down list on the dialog window.

Typically, you perform a series of transformations on your dataset to, for example, randomize the records and split it into subsets for creating a model, testing it, and doing a practice prediction with it.

A transformation command always creates a new dataset, which is automatically included in the project; its name appears in the **Workspace** listing.

2.3 Building a Model

Once you have created the appropriate datasets and performed the appropriate transformations, you can create a model that you'll train and test and then use to predict the value of the target variable.

Typically, you will use one of the three subsets of your dataset to create and train the model, and another subset to test the model. During training, the model "learns" how the target variable's value is determined by the other variables. In the practice prediction phase, you use the trained model to predict the value of the target variable in the third subset dataset, which of course already has actual values in the target field. In the next phase of your project, you'll compare predicted with actual values to get an idea of how accurate your model is.

2.4 Analyzing the Results

The next step is to evaluate your model's performance, using the commands on the **Analysis** menu. You'll compare the model's predictions in the target variable with the variable's actual value. There are several different analyses available to you.

The process is likely to be iterative; you and Darwin keep adjusting parameters in order to improve the model's performance, i.e., to increase the accuracy of its predictions.

2.5 The Data Mining Process for Each Model Type

This section presents diagrams of the model-building process for trees, nets, and match models. At the most general level, the process is similar for all three model types, though the details differ. The process can also be accomplished using the Darwin Wizards.

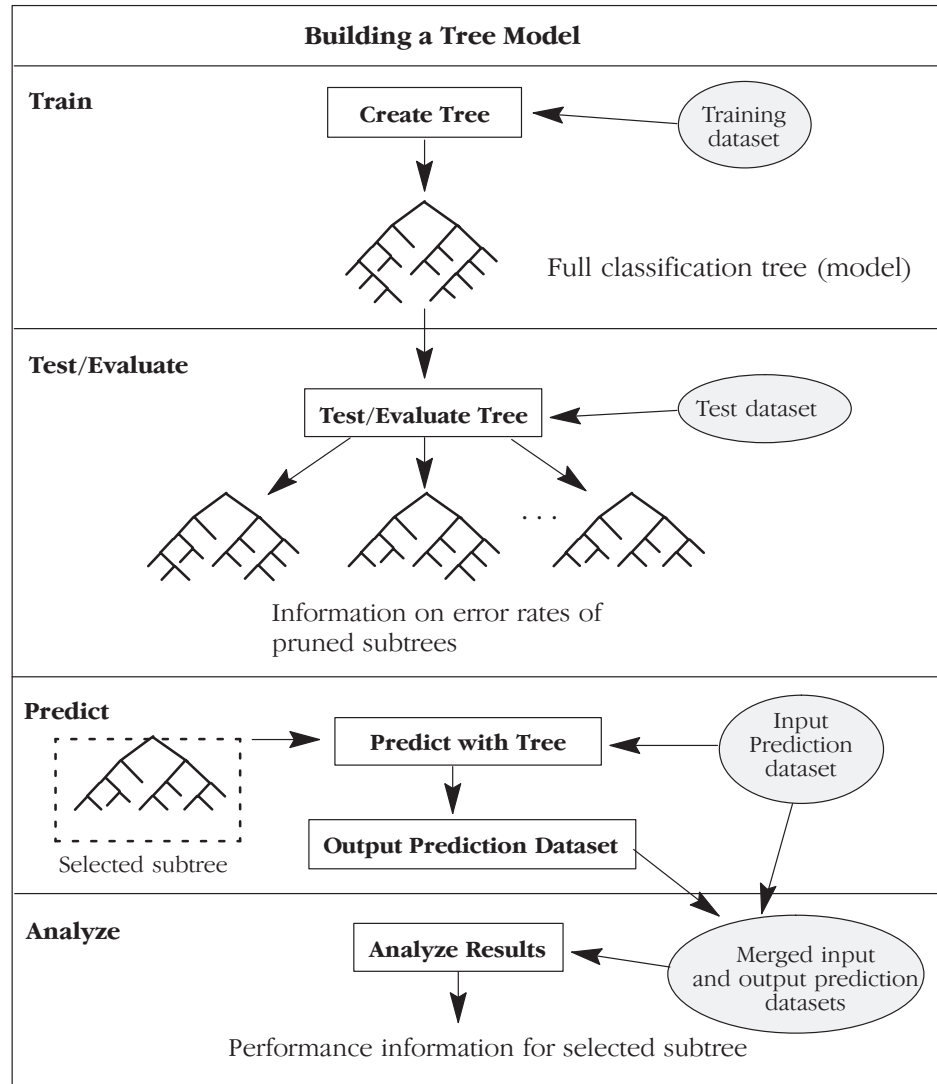


Figure 2. Building a tree model.

Figure 2 illustrates building a tree model, also known as a classification and regression tree (C&RT). Tree models develop a set of rules that split the data into ever smaller branches, and then “prune” the tree improve the accuracy of its predictions.

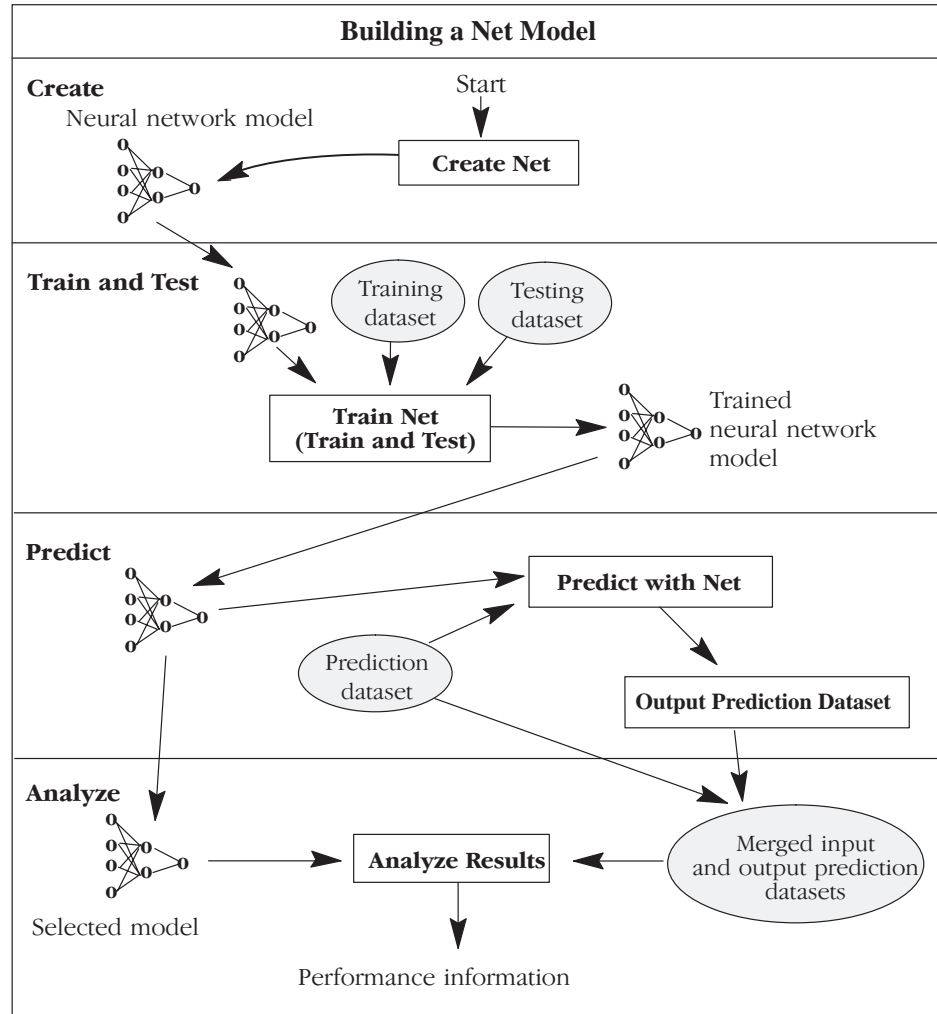


Figure 3. Building a net model (using Train and Test).

Figure 3 illustrates building a neural net model. Neural nets simulate the interconnectedness of pathways in the brain. This diagram illustrates building the net using the train and test option, which can use two datasets or two parts of a single dataset for training and testing.

You create nets in two steps. In the first step, you define the structure of the net. In the second, you train the net. Training a net involves passing the data through the net a specified number of times, with Darwin adjusting the weights after each pass.

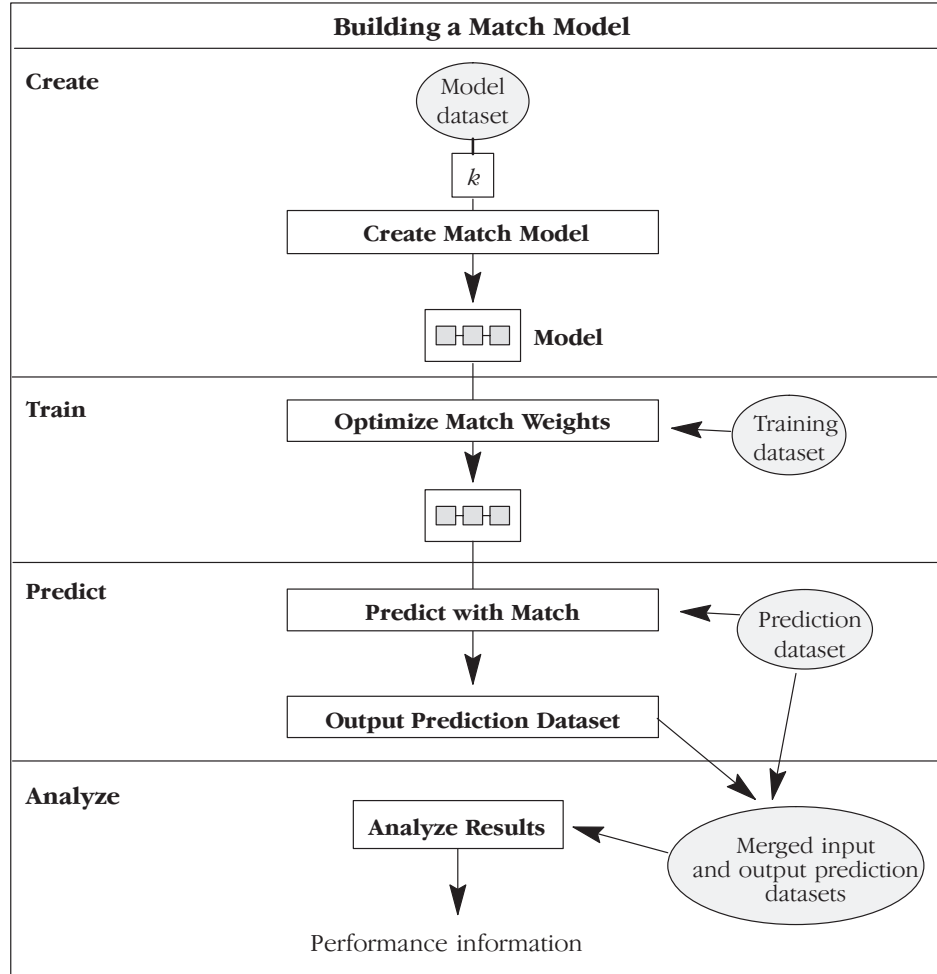


Figure 4. Building a match model.

Figure 4 illustrates building a match model. Match models are based on memory-based reasoning, using a k -nearest-neighbor algorithm. This is something like the notion that “birds of a feather flock together.” Match models are very good at discovering island patterns in data.

3 The Darwin User Interface

This chapter describes

- how to start and end a Darwin session (section 3.1).
- the main Darwin window (section 3.2)
- customizing the Darwin user interface, Darwin command output, the two types of windows, and characters permitted in Darwin names (section 3.3)

3.1 Starting and Ending a Darwin Session

3.1.1 Starting a Darwin Session

Assuming Darwin is properly installed on your system and that at least one Darwin server is configured and running, you can start a Darwin session by selecting it from the **Start** menu, under **Programs**.

The Darwin **Welcome** screen appears, and after a moment is replaced by a **Login** dialog window:

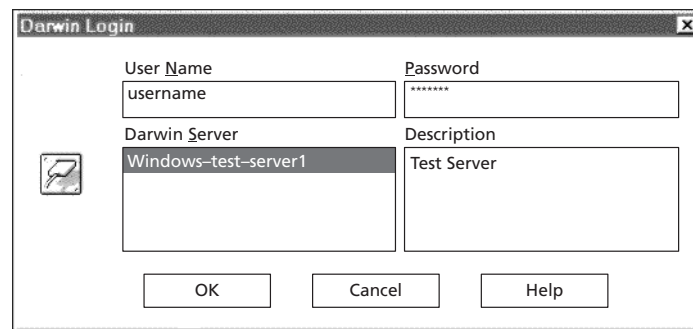


Figure 5. Darwin login window.

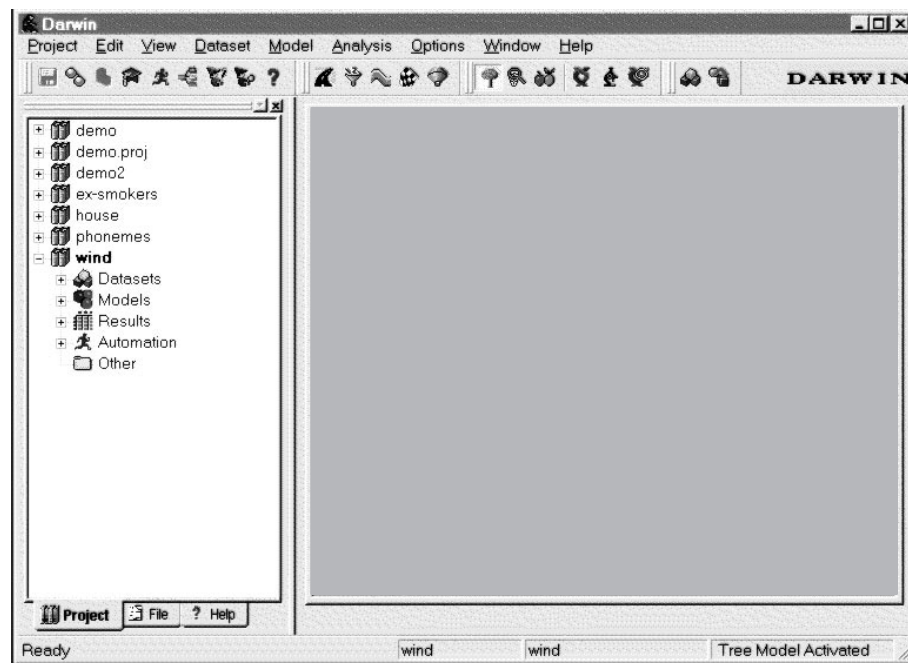
The **Login** dialog prompts you for the following information:

- **User Name:** Enter the user name for your account on the server.
- **Password:** Enter the password for your account on the server (it appears as a string of asterisks). This password may or may not be the same password that you use to log on to your PC.
- **Darwin Server:** Click the name of the Darwin server you want to connect to. To the right of the server name is a brief description of it.

Then, to start Darwin,

- Click **OK** or press the ENTER key

The main Darwin window then appears:



Note: If the attempt to connect to a server fails, check these possible reasons:

- The Darwin license(s) on the server you chose is/are already in use.
- The server itself is not currently available; choose another server.
- You entered an incorrect user name or password.

Any of these conditions leaves you in the login window, so that you can either select a different server or try a new connection. If you prefer to try connecting again at another time, click the **Cancel** button.

If all Darwin licenses on the server you chose are already in use, see your system administrator.

3.1.2 Ending a Darwin Session

From within a Darwin session, the best way to end the session and exit Darwin is by clicking **Exit** on the **Project** menu, because it gives you a last chance to save unsaved objects.

In a typical Darwin session, you create many objects, some of which you save, and most of which you do not save. If, before ending the session, you change your mind about some of the objects you did not save, and decide that you may want them in a future session after all, ending the session with the **Exit** command gives you another opportunity to save them.

3.2 The Main Darwin Window

The main Darwin window, in its default configuration, contains the following elements (see figure 6) :

- **Title bar**, which displays the window's name, **Darwin**, plus the standard Control Menu button and, at the right, the standard Windows 95 Minimize, Maximize/Restore, and Close buttons.
- **Menu bar**, which contains the menus of Darwin commands, each with its pull-down submenu; see section 3.2.1.
- **Toolbar**, containing groups of icons for the most frequently used commands; see section 3.2.2.
- **Workspace**, which has three tabbed pages, providing name spaces for projects, a backward-compatible file system, and **Help**. Select a tab by clicking one of the buttons on the small toolbar under the **Workspace**. The **Workspace** is described in more detail in section 3.2.3.
- **Document area**, the largest area of the window, where all the dialog boxes and MDI windows display results, data, files, etc. (see section 3.3.2 for a description of the two kinds of windows).

- **Status bar**, at the bottom, which displays at the left messages that indicate Darwin's progress while it executes commands, and messages that tell when Darwin is ready to accept a new command. At the right, the status bar displays the names of active projects, models, and files.

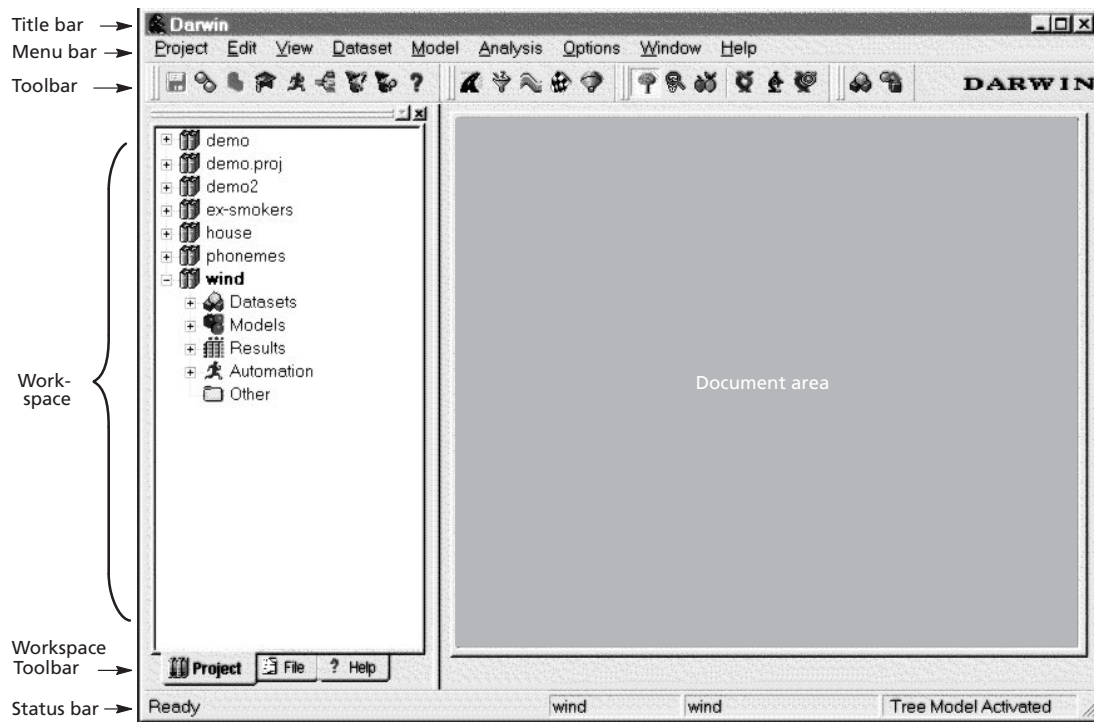


Figure 6. The Darwin window with main components identified.

3.2.1 The Menu Bar

Click a menu name to view its contents; then click the command that you want.

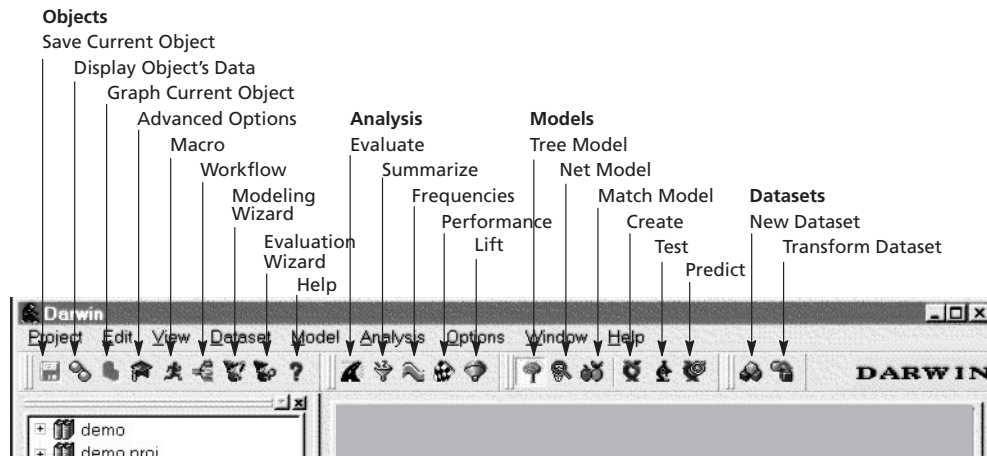
The Darwin menu bar, in its default configuration, contains the following menus:

- **Project:** The commands on the **Project** menu allow you to create new projects, open existing projects, close, save, create a new Darwin file, export a table, display, graph, print, connect to and disconnect from a database, stop an executing command, and exit Darwin. **Project** menu commands are described in chapter 4.

- **Edit:** These commands allow you to undo, cut, copy, and paste in text files. See chapter 5.
- **View:** The **View** menu contains the **Wizard** commands, the **Workflow** command, and the **Refresh** command. This menu also contains commands that let you change the appearance of elements of the Darwin window. See chapter 6.
- **Dataset:** These commands let you create, export, and transform datasets. See chapter 7; see also *Darwin Reference*, chapters 4 and 5.
- **Model:** These commands let you create, copy, and test models and use them to predict. See chapter 8; see also *Darwin Reference*, chapters 6, 7, 8, and 9.
- **Analysis:** These commands let you perform various analyses of models. See chapter 9; see also *Darwin Reference*, chapter 10.
- **Options:** This menu gives you access to advanced options, macros, and code generation. See chapter 10; see also *Darwin Reference*, chapters 6, 7, 8, and 9.
- **Window:** This menu lets you cascade or tile dialog windows, and arrange icons. See chapter 11.
- **Help:** This menu displays a list of help contents, context-sensitive help, Intro to Darwin, and About Darwin. See chapter 12.

3.2.2 The Toolbar

The Darwin toolbar groups related icons for some of the commands most frequently used. The figure below shows the default configuration.





You can change the order and placement of these icons, you can add or remove these and other icons, with the **Customize** menu (see section 6.4). The listing below reflects the default arrangement of icons, from left to right.


- Object icons:



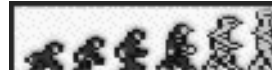
- Save Current Object
- Display Object's Data
- Graph Current Object
- Advanced Options for Current Object (same as Options menu, Advanced)
- Macro (same as Options menu, Macro)
- Workflow
- Modeling Wizard
- Evaluation Wizard
- Help

- Analysis icons: 
 - Evaluate (same as Analysis menu, Evaluate)
 - Summarize (same as Analysis menu, Summarize)
 - Frequencies (same as Analysis menu, Frequencies)
 - Performance (same as Analysis menu, Performance)
 - Lift (same as Analysis menu, Lift)

- Model icons: 
 - Select Tree Model mode (same as Model menu, Tree)
 - Select Net Model mode (same as Model menu, Net)
 - Select Match Model mode (same as Model menu, Match)
 - Create Model (same as Model menu, Create)
 - Test Model (same as Model menu, Test)
 - Predict with Model (same as Model menu, Predict)

- Dataset Icons: 
 - New Dataset (same as Dataset menu, Create)
 - Transform Dataset (same as Dataset menu, Transform)

The right side of the toolbar contains the **DARWIN** icon, which indicates, through animation, when Darwin is executing a command.



Important: The animation indicates that the server is busy. Interrupting the server can sometimes produce unexpected results, so wait for it to finish before trying to do something else in Darwin.

You can turn off the animation by clicking the **View** menu and then clicking **Darwin Logo** to uncheck it and hide the logo.

3.2.3 Darwin Workspace

The **Workspace** is made up of three tabbed pages:

- **Project:** Contains a listing of all project directories visible to the user. The current project's name appears in bold; its icon appears in color.

The tree control for the **Project** tab has the following categories: datasets, models, results, automation (macros), and other. As different items are created, they are inserted in the appropriate category.

Double-click an item to bring up a dialog box or an MDI window that displays its attributes and data.

- **File:** Contains a listing of all project files visible to the user, listed by their UNIX names (i.e., these are the filenames as UNIX sees them, whereas the names shown in the **Project** tab are “nicknames”).

The tree control for the **File** tab has the following categories: datasets, descriptors, models, logs, tables, text, costs, priors, SQL script, weights, and other. As items are created, they are inserted in the appropriate category.

Double-click an item to bring up a dialog box or an MDI window that displays its attributes and data.

- **Help:** Contains a listing of the **Help** contents.

To bring up a help topic, double-click the icon by the topic's name. See chapter 12 for details.

For each of the tabs listed above, you can use the ALT key with the arrow keys to move up and down the list and open and close the tree control.

Also, for each of the tabs listed above, you can select an item and press the right mouse button to bring up a context menu from which you may choose a command.

3.3 About the Darwin User Interface

3.3.1 Customizing Darwin

You can modify the appearance of the Darwin user interface: You can hide or display toolbar groups, disable or enable the “Cool Look,” disable or enable tooltips, and display large or small buttons. You can also add icons to or remove them from the toolbar groups or rearrange them. See section 6.4 for details.

You can also change the default Darwin behavior regarding naming and renaming, saving objects and files, editing descriptor files and macros, and displaying **Workflow**. See section 10.1.8 for details.

3.3.2 Dialog Boxes and MDI Windows

The Darwin user interface uses two kinds of windows: dialog boxes and MDI (multiple-document interface) windows.

Certain operations bring up dialog boxes, and others bring up MDI windows:

- **Dialog Boxes:** Operations that bring up dialog boxes are those that must be completed or canceled before other operations are possible, e.g., a **Create** command, or an **Advanced Options** command. Dialog boxes can be moved anywhere on your screen.
- **MDI Windows:** Operations that bring up MDI windows are those that can coexist with other operations, i. e., they do not have to be completed before other operations can be started. Examples are **Dataset** commands and **Analysis** commands. You can move MDI windows around only within the document area. MDI windows also have labeled folder tabs at the bottom of the document area.

3.3.3 Command Output

Output from a Darwin command can take several forms:

- Commands that create, open, or transform a dataset or model produce a dataset or model that exists as a memory object. By default, Darwin prompts you about saving these objects. (You can change this behavior; see section 10.1.8.)
- Prediction commands produce datasets as their output.
- Commands that evaluate models produce summaries of results as output (spreadsheets). The summaries are displayed on your PC; you can also have a summary
 - saved to a file in your project directory
 - printed to a printer accessible from your desktop
 - printed to a file

- Commands that display information do so in summary or tabular form (spreadsheets). In many cases, you can plot the information to one of several choices of graphic displays with the **Graph** command (see section 4.9).

Numbers are displayed to four decimal digits. Note that the calculations that create these numbers are carried out at whatever precision is available. This occasionally produces results that may be puzzling at first glance. For example, the error rate reported while training a net may appear to remain stable over several iterations, yet in fact be slowly declining. Similarly, the minimum and maximum values of a variable may appear to be equal, but a small standard deviation will be reported. This indicates a variation that is smaller than can be seen with only four decimal places.

- The **Code Generation** command produces a program in C++, C, or Java (plus related files for C and C++).

3.3.4 Naming Objects

Darwin checks the validity of names of objects, i.e., projects, models, datasets, or input and output files. A name is valid only if it consists of characters from the following sets:

- numerals zero through nine (0 - 9)
- lowercase alphabet (a - z)
- uppercase alphabet (A - Z)

The following characters are *not* permitted in the names of Darwin objects:

. , : ; < > # @ ! ? | [] { } () ~ % ^ & * " ' \ /

Control key characters are not allowed as input to Darwin dialogs. If you accidentally enter a control key character, Darwin beeps and removes whatever input you entered. Re-enter the input without the control characters.

4 Project Menu

The **Project** menu contains the following commands:

- **New Project**
- **Open Project**
- **Close**
- **Save**
- **New File**
- **Export Table**
- **Display**
- **Graph**
- **Print**
- **Print Preview**
- **Database Connect**
- **Database Disconnect**
- **Stop**
- **Exit**

4.1 New Project

Click **Project -> New Project** or
Press **CTRL-N**

Creates a new project, which then becomes your current project. In most cases, creating a project is the first thing you must do the first time you use Darwin.

The **New Project** dialog prompts you for the following information:

- **Project:**
 - **Name:** Enter a name for the project. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.
 - **Description:** Description of the project (optional).
 - **Shared:** Click the check box to indicate that the project is to be shared. The system administrator specifies (during configuration) whether sharing is to be an option on each server. This check box is greyed out for servers on which sharing is not an option.

- **In Server:** (This information is displayed by the system)
 - **Name:** Name of the server on which the project is to reside; this will be the name of the server you are logged on to.
 - **Description:** A brief description of the server identified in **Name**.
 - **Distributed:** If you are logged on to a single-node server, this check box will be greyed out. If you are logged on to a multinode SMP, it will be enabled. The default is serial; click the check box to indicate that the project is to be distributed.

After you have supplied the necessary information, click **OK** or press ENTER to create the project. You can also **Cancel** the command or click the **Advanced** button, which brings up the **Advanced Options, Project** page.

The **Project** page of the **Advanced Options** dialog is a bookkeeping and record-keeping page. It provides a place to store the information about a given project and any notes you may wish to make. See section 10.1.1 for details.

4.2 Open Project

Click **Project -> Open** or
Press **CTRL-O**

Opens an existing project, which then becomes your current project. The previous project is automatically closed. You can use this command to change from one project to another.

The **Open Project** dialog displays a list of projects from which to choose.

To select a project, click its name and click the **Select** button, or double-click the name. The new current project's name then appears in bold in the **Workspace**.

Another way to open a project or change projects is to left-click the project's name in the **Workspace** listing, and then right-click to bring up a context menu. On the context menu, click **Set as Active Project**. The new current project's name then appears in bold in the **Workspace** listing.

4.3 Close

Click **Project -> Close**

Closes the currently selected project. Windows associated with the project close, but the project continues as the current project until you select a different project as your current project.

4.4 Save

Click **Project -> Save** or
Press **CTRL-S**

Allows you to save an object. The **Save** dialog prompts you for the following information:

- **Name of object:** Enter a name. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.
- **Object type:** The type of object is identified here.
- **Description of object:** Enter any description you want.

After you have supplied the needed information, click **Yes** to save the object. You can also click **Cancel** to cancel the command or **Advanced** to bring up **Advanced Options** for **Setup** (see section 10.1.8).

4.5 Deleting a Project

To delete a project:

- In the **Workspace** listing, right-click on the project name to bring up a context menu, then click **Delete**. A dialog appears and asks whether you really want to delete the named project.

If the project is not empty, Darwin will notify you.

4.6 New File

Click **Project -> New File**

The **New File** command lets you create a new Darwin file and copy a file to the Darwin server. Note that the **Edit** menu's commands are available when you are creating a Darwin file; you can use them to copy, cut, and paste text (and undo).

Darwin File: Click the **Darwin File** option button and then click the name of the kind of file you want to create. The types of files you can create with this command are

- descriptor
- data
- log
- priors
- costs
- weights
- SQL script
- notes
- reports

Click the **Darwin File** option button and then click the name of the kind of file you want to create.

For example, to create a descriptor file,

- Click **Darwin File**.
- Click **Descriptor**.
- Click **OK**; a template for a descriptor file appears in the document area.
- Edit the template as appropriate for your purposes. See *Darwin Reference*, section 4.5, for complete information about descriptor files.

To save a file, you can use the **Project** menu's **Save** command, or, when you are using the **New File** dialog and have finished editing the template and close the dialog, a **Save As** dialog appears and gives you the option of saving the file. Darwin files are saved into the current project's directory.

Copy to Darwin Server: To copy a simple text file from your PC to the project directory on the UNIX side,

- Click the **Copy to Darwin Server** button.

- Click **OK**. A browser appears, from which you can select any *simple* text file (for example, a file created with something like NotePad, but not with something like Microsoft Word).

4.7 Export Table

Click **Project -> Export Table**

This command is available if there is a table displayed in the document area. **Export Table** saves a table to your desktop or to another location that you specify. When you click this command, a **Save As** dialog appears and prompts you for the following information:

- **Save in:** This field displays the folder for the file. The default location is your desktop; click the down-arrow at the right to select a different location. The large text box below displays the items in the location specified.
- **File name:** Enter a name for the file.
- **Save as type:** Select the file type.

4.8 Display

Click **Project -> Display** or
Click the **Display** icon (above) or
Press **F2**

There are several ways to display an item:

- With the item's name selected in the **Workspace** listing,
 - Click **Project -> Display** or
 - Click the **Display** icon on the toolbar or
 - Press **F2**
- Double-click the item's name in the **Workspace** listing.

What is displayed depends on the type of item. If any of the following is selected, the item itself is displayed:

- dataset
- descriptor file

- text file
- results file

If a model is selected, **Display** brings up a model properties sheet, which has between two and four tabs: Project, model (Tree, Net, or Match), Object, and Parent:

- The **Project** tab displays basic information about the project — its name, the name of the project leader, phone number, a description of the business problem, the project's objectives, and whether the project is shared.
- The model tab is labeled **Tree**, **Net**, or **Match**, and displays information specific to the model, e.g., its name, the name of the file as known to the server, and basic properties specific to the model, including values of parameters on **Advanced Options** dialogs.
- The **Object** tab displays the model's name, filename as known to the server, a description, the name of the project, and object's type. The **Object Properties** section displays the date created, date last used, storage (serial or distributed), parent, precedent (target + field), whether it is in memory, whether it is open, and the command that created it.
- The **Parent** tab displays the object's name, filename as known to the server, a description, the name of the project, and the object's type. The **Object Properties** section displays the date created, date last used, storage (serial or distributed), parent, precedent (target + field), whether it is in memory, whether it is open, and the command used to create it.

4.9 Graph



Click **Project -> Graph** or
Click the **Graph** icon (above) or
Press **CTRL-G**

The **Graph** command calls up Microsoft Excel to plot the data of an object that has been processed in this project. The **Chart** dialog window that appears requires the following information:

- **Object:**
 - **Name:** If a name is preselected, that name appears.
 - **Type:** Type of object, e.g., dataset.

- **Select Fields to Plot:**
 - **X-axis:** Enter the name of the x axis field.
 - **Y-axis:** Enter the name of the y axis field.
- **Select Chart Parameters:**
 - **Type:** Select the type of chart. Choices are area, bar (default), column, line, pie, doughnut, radar, XY-scatter, combination, 3-D area, 3-D bar, 3-D column, 3-D line, 3-D pie, 3-D surface.
 - **Number:** Indicate the number of elements; defaults vary by type of chart.
 - **Title:** Supply the title you want to appear on the chart.
 - **Label X:** Label for the x -axis.
 - **Label Y:** Label for the y -axis.
 - **Subtitle:** Provides for a subtitle.
 - **Data Series in:** Indicate whether rows or columns.

After you have supplied the necessary information, click **OK** to create the chart or **Cancel** to cancel the command.

Note: Creating a graph is often a fairly slow operation — the data is loaded into Excel, and Excel then creates the graph.

4.10 Print

Click **Project -> Print** or
Press **CTRL-P**

The **Print** command is available when there is something printable in the document area. If there is more than one object in the document area, it prints the object that is active (the object you most recently did something with).

Clicking **Print** brings up the **Print** dialog window, which displays the standard Windows printing options: printer, print range, number of copies, and properties (page setup, advanced). When you are ready, click **OK** to print, or **Cancel** to cancel the command.

4.11 Print Preview

Click **Project -> Print Preview**

The **Print Preview** command displays a preview of the printout of the active object displayed in the document area.

You can zoom in and out, page through the document, display it as single pages or in two-page spreads. If you decide to send the object to the printer, you can do so directly from the **Print Preview** dialog by clicking **Print**, which brings up the same dialog that is produced by the **Print** command (section 4.10).

If there is more than one object in the document area, it prints the object that is active (the object you most recently did something with).

4.12 Database Connect

Click **Project -> Database Connect**

You can create Darwin datasets by running SQL scripts on databases, which requires that you be connected to a database from within a Darwin session. To connect to a database, use the **Project** menu's **Database Connect** command.

The **Database Connect** dialog prompts you for the following information:

- **Name:** Enter your user name, which may or may not be the same as your user name for your PC or the Darwin server (see example below)
- **Password:** This is the password for the database, which may or may not be the same as your password for your PC or the Darwin server. For example, to connect to Oracle, you might type in the following:
Username: *myname*
Password: *mypassword*
- **Database:** The database names displayed are read in from your `odbc.ini` file. These are datasource names that point to databases. Click a database name; then click **OK**.

You are now connected to the selected database. On the **Project** menu, note that the **Database Connect** command is no longer available, and the **Database Disconnect** command is now available.

You may have only one database connection open at a time. Therefore, you must close one database connection before you open another.

See *Darwin Reference*, section 4.6, for general information about using databases, and see *Darwin Installation and Administration* for server configuration requirements for allowing connections to databases.

4.13 Database Disconnect

To end a database connection, click the **Database Disconnect** command.

A dialog appears to ask whether you really want to disconnect, and gives you two options: **Yes** and **No**.

If you click **Yes**, you are disconnected from the database, and the **Database Connect** command is again available on the **Project** menu.

If you have an active dataset that was created from the database, you must save it before disconnecting.

If you do not disconnect from a database before you exit Darwin, Darwin closes the connection automatically.

Note: If you do not close all datasets created using a database product before you try to disconnect from the database, you get the error message *Database in Use*. This message appears because of the way that a database supporting ODBC creates datasets. When such a product creates a new dataset, it does so by creating a pointer; the product does not retrieve data at the same time. If you try to close a connection to the database when there is an active pointer, you get the *Database in Use* message.

Therefore, before disconnecting, you must copy any datasets created from a database to a Darwin dataset and then close the original dataset that was created directly from the database.

4.14 Stop

Click **Project -> Stop**

This command interrupts the building of model. Some aspects of a build cease immediately; others complete the cycle or process they are in before stopping.

Important: **Project -> Stop** is the safe way to stop a model-building command that is running. Closing the window by clicking the generic Windows Close button can crash Darwin.

4.15 Exit

Click **Project -> Exit**

Terminates your Darwin session. You can exit Darwin by clicking the generic Windows close button, but the **Project** menu's **Exit** command is the best way to end a Darwin session, because Darwin then displays a list of any unsaved objects and gives you the option of saving them.

5 Edit Menu

The **Edit** menu contains the following generic Windows Edit commands:

- **Cut**
- **Copy**
- **Paste**

These commands are useful with text files, for example, when creating Darwin files with the **Project -> New File** command.

These commands do not work with material other than text; i.e., you cannot use the **Edit -> Copy** command to copy a model. (Use the **Model -> Copy** command instead, described in section 8.4.)

6 View Menu

The commands on the **View** menu include the two Darwin wizards as well as commands that let you change the appearance of elements on the Darwin window:

- **Modeling Wizard**
- **Evaluation Wizard**
- **Workflow**
- **Customize**
- **Status Bar**
- **Workspace**
- **Folders**
- **Darwin Logo**
- **Refresh**

6.1 Modeling Wizard



Click **View -> Modeling Wizard** or
Click the **Modeling Wizard** icon (above) or
Press **CTRL-W**

The Darwin **Modeling Wizard** provides a quick way to create a model with the minimum input from you: the input dataset, a target field, the model type, and a name for the model. The **Modeling Wizard** performs the basic steps in preparing data, building a model (without any optimization), and predicting with a model.

For a user new to Darwin, the wizard provides a “one-button” approach — it creates the model after you have supplied the basic information. For the more experienced user, the wizard lets you set up almost any of the model-building parameters and then run through all the steps.

The **Modeling Wizard** consists of four dialog windows that are displayed sequentially. There are **Back**, **Next**, **Finish**, **Cancel**, and **Help** command buttons on every dialog. On each (including the first, where you provide the basic information), you can click **Finish** and Darwin will create the model. You must complete the first step; the others are optional.

6.1.1 Step 1: Create Model

The **Create Model** dialog requires the following information:

- **Input Dataset:** Select the appropriate input dataset for the model.
- **Target field:** Select one target field for the model.
- **Model type:** Select the model type: **Tree**, **Net**, or **Match**.
- **Model Name:** Specify the name for the model. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.

When you have entered a name for the model, the **Next** and **Finish** buttons become available.

You can either:

- Click **Finish** and have the **Modeling Wizard** complete the next steps and create the model, or
- Click **Next** to go to the next step.

6.1.2 Step 2: Input Dataset

On the **Input Dataset** page, you specify how to create the three subsets of the input dataset for the training, testing, and practice prediction phases.

The default (indicated by the checked box at the top) is to have the Wizard split the input dataset into three subsets. If you have already split the input dataset yourself into three subsets, click this box to uncheck it and cancel the default.

If you go with the default, the left side of the page (**Dataset Split**) is enabled. It provides for the following settings:

- **Randomize:** By default, this box is checked, which means the input dataset will be randomized before it is split into three subsets.
- **Percentage of dataset:** By default, the randomized input dataset is split into thirds. You can use the slider bars to indicate different percentages.

- **Fields to be used in creating the model:** By default, all fields in the dataset appear in selected mode. To exclude a field, hold the CTRL key down and click the name of the field to be excluded.

If you do not go with the default, i.e., if you created three subsets of your randomized input dataset before invoking the **Modeling Wizard**, click the checkbox at the top to cancel the default. The right side of the page is then enabled. Specify the names of the three subset datasets in the three boxes (Train dataset, Test dataset, and Predict dataset).

At this point, you can click **Finish** and have the **Modeling Wizard** create the model, or you can click **Next** to go to the next step.

6.1.3 Step 3: Mode

In this step, you ask the Wizard to create the model in one of three modes:

- Create a model very rapidly, although it may not be very accurate.
- Create a more accurate model, which usually requires more time (the default).
- Create the best possible model, which takes the most time.

You can at this point click **Finish** and have the **Modeling Wizard** create the model, or you can click **Next** to go to the next step.

6.1.4 Last Step: Finish

This last dialog shows you the default names of the output dataset(s) and the error results; you can accept the default names or change them, as you like. These files will be used by the **Evaluation Wizard**.

This page also has an **Advanced** button, which gives you access to **Advanced Options** for the type of model that you are creating.

The checked checkbox indicates that the **Modeling Wizard** output is, by default, saved. Click the checkbox to uncheck it and cancel the default.

Click **Finish** to have the Modeling Wizard create the model.

When the Wizard has finished building the model, the model's name appears in the **Workspace** listing.

If you chose not to save the object and later change your mind (before exiting Darwin), you can still save it. Click its name in the **Workspace** listing and right-click to bring up a context menu; then click **Save**.

The next step is to evaluate the model using the **Evaluation Wizard** (section 6.2).

6.2 Evaluation Wizard

Click **View -> Evaluation Wizard** or
Click the **Evaluation Wizard** icon (above) or
Press **CTRL-U**

The Darwin **Evaluation Wizard** provides a quick way to perform an initial evaluation of a model. You can use the **Evaluation Wizard** to evaluate a model you created by yourself or one created using the **Modeling Wizard**.

The **Evaluation Wizard** consists of dialog boxes. Each has **Back**, **Next**, **Finish**, and **Cancel** buttons. On either dialog, you can click **Finish** and Darwin will proceed.

6.2.1 Step 1: Input

The **Evaluation Wizard Input** dialog requires the following input:

- **Model to Evaluate:** Select the model to be used in the analysis.
- **Analysis Dataset:** By default, this field contains the name of the correct dataset, i.e., the analysis dataset created by the **Modeling Wizard**.
If you are evaluating a model you created yourself, enter the name of the merged dataset, i.e., the output of the **Merge** transform (see section 7.3.3), which merged the input and output datasets from the prediction phase.
- **Target Field:** By default, the field contains the name of the correct target field.
- **Target Value:** Enter the target value. (A target value is needed for the **Lift** computation. If you do not want a **Lift** computation, you can leave this box blank, but go to the next dialog (click the **Next** button) and click the **Lift** check box to deselect it.)

- **Model Evaluation Error Results:** By default, this field contains the name of the error file created by the Modeling Wizard, modelWizerror. You can select a different error results file if you like.
- **Dataset for Sensitivity Analysis:** You can select a different dataset for **Sensitivity** analysis if you like.

You can at this point click **Finish** and have the **Evaluation Wizard** proceed with the evaluation and analysis, or you can click **Next** and go to the next (and last) step, where you can choose different kinds of results.

6.2.2 Last Step: Results

The **Output Results** dialog provides five check boxes that you use to indicate the kind of results you would like to see:

- **Sensitivity** (see *Darwin Reference*, section 10.6).
- **Error Results**
- **Tree Rules** (for tree models only; see *Darwin Reference*, section 7.2.3).
- **Performance Matrix** (see *Darwin Reference*, section 10.4).
- **Lift** (see *Darwin Reference*, section 10.5). If you select **Lift** calculation, you can specify values for **Profit** (default is 1) and **Cost** (default is -1). **Note:** **Lift** is disabled if the target field is ordered.

Click the **Advanced** button to see or change the settings of any **Advanced Options** for **Analysis**.

By default, all relevant check boxes are checked. To deselect any option, click its check box.

Then click **Finish** to have the Wizard complete the evaluation.

6.3 Workflow

Click **View -> Workflow** or
 Click the **Workflow** icon (above) or
 Press **CTRL-F**

Workflow produces a diagram of the steps and procedures you have undertaken within a given project in your current Darwin session. The diagram lets you see where you are, where you have been, and where you are going in the process. Darwin creates the diagram from the commands you execute.

When you click **Workflow**, the diagram appears in the document area, and a small control panel appears above the main Darwin window.



The three icons on the left of the control panel let you display three different versions of the **Workflow**: an enhanced view (the default, with the most detail), a “standard” view, and a view that includes objects only.

The two icons on the right (magnifying glasses) control zooming in and out:

- To zoom out on the diagram, click the left-hand magnifying glass (with a minus sign in the middle) on the control panel and then click the diagram. You move out successively with each click.
- To zoom back in, click the right-hand magnifying glass (with a plus sign in the middle) on the control panel and then click the diagram. You move in successively with each click.

A new menu item, **Workflow**, appears on the menu bar, between **Options** and **Window**.



The three views and the zooming in and out can be controlled from the **Workflow** menu as well as by the control panel.

The objects in the Workflow diagram have property sheets, which you can view by right-clicking the object’s icon.

6.4 Customize

Click **View -> Customize**

The **Customize** dialog window has two tabs: **Toolbars** and **Commands**.

- The **Toolbar** page lists the groups of icons on the toolbar (Project, Dataset, Model, Analysis, Object); click a name to uncheck it and hide the associated group of icons. You can also create a new group of icons, **Tools**.
Tools contains icons for **Refresh Workspace**, **Database Connect**, **Database Disconnect**, **Database Query**, **Code Generation**, and **Undo**. These icons are not by default activated. You can activate them if you wish.
- The **Toolbar** page also has check boxes to enable/disable tooltips, the “Cool Look,” and large buttons.
- The **Commands** page lets you custom-configure the toolbar icons, i.e., you can change or eliminate them as you wish, and group or arrange them in any way you like.

6.5 Status Bar, Workspace, Folders, Darwin Logo

By default, the status bar, Workspace, folders, and the Darwin logo are visible. Each has a check mark on the **View** menu. To hide an element, click its name to uncheck it.

6.6 Refresh

Click **View -> Refresh** or
Press **F5**

The **Refresh** command updates the client copies of any files that may have been changed on the server (UNIX).

7 Dataset Menu

The **Dataset** menu contains the following commands:

- **Create**
- **Export**
- **Transform**

There is no command for copying datasets. To copy a dataset, simply do a “drag-and-drop” with the dataset’s name — i.e., in the Workspace listing, left-click on a dataset name to select it, and hold the left mouse button down while you drag the selected dataset name to the new location.

See *Darwin Reference*, chapters 4 and 5, for complete information about datasets.

7.1 Create

Click **Dataset -> Create** or
Click the **New Dataset** icon (above) or
Press **CTRL-D**

You can create a new dataset from either a text file or from a database; the **Create Dataset** dialog window has a tab for each. If you are creating a dataset from a text file, click the **Import Text File** tab.


7.1.1 Import Text File

The **Create Dataset, Import Text File** tab, prompts you for the following information:

- **Name:** Enter a name for the dataset. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.
- **Description:** Description of the dataset (optional).

- **Text Data File:** The name of the source text file; see *Darwin Reference*, section 4.2.
- **Descriptors:** Name of dataset descriptor file; see *Darwin Reference*, section 4.5.

Before clicking **Create**, specify the separator (delimiter) the dataset uses:

- On the **Options** menu, click **Advanced** (or click the **Advanced Options** icon on the toolbar ) to bring up **Advanced Options** and click the **Datasets** tab.
- In the **Descriptor** section, in the **Separator** box, you'll see that the comma (",") is the default. If this is not the separator used in your dataset, click the down-arrow to view the list of possible separators, and click the appropriate separator.
- Click **OK**; the **Advanced Options** dialog disappears.

On the **Create Dataset** dialog, click the **Create** button. (The **Create** button is not available until you enter a name for the dataset in the **Name** edit box.)

A **Save As** dialog appears and asks whether you want to save this object. (By default, Darwin prompts you about saving a newly created object. To change this behavior, see section 10.1.)

When the dataset is created, its name appears in the **Workspace** listing of the current project, under Datasets, Created.

7.1.2 Database

If you are creating your dataset from a database, click the **Database** tab on the **Create Dataset** dialog window:

The **Create Dataset, Database** tab, prompts you for the following information:

- **Name:** Enter a name for the dataset. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.
- **Description:** Description of the dataset (optional).
- **Database Query: Use SQL Script File:**
 - With the SQL script file box *not* checked, you can enter a query in the edit box. For example,

```
select * from name_of_database_table
```

- With the SQL script file check box checked, Darwin displays the names of all SQL scripts available for the current project.
- To create an SQL file, see the **New File** command (section 4.6). If you have the **Create Dataset from Database** dialog open before you create the SQL script file, you'll need to close the **Create Dataset** dialog and open a new one in order to see the new SQL script file listed.

When all the information is entered, click the **Create** button.

A **Save As** dialog appears and asks whether you want to save this object. (By default, Darwin prompts you about saving a newly created object. To change this behavior, see section 10.1.8.)

When the dataset is created, its name is listed in the **Workspace** of the current project, under Datasets, Created.

See *Darwin Reference*, section 4.6, for information about working with databases.

7.2 Export

Click **Dataset -> Export**

The **Export** command reverses the process of creating a dataset, i.e., it creates a dataset descriptor file and a data (text) file from an existing dataset.

When you click **Export**, a **Save As** dialog appears to ask whether you want to save the object, and offers you a default name (which you can keep or change; see section 3.3.4 to learn what characters are permitted in the names of Darwin objects).

7.3 Transform

Click **Dataset -> Transform** or
Click the **Transform Dataset** icon (above) or
Press **CTRL-T**

The **Transform** commands create modifications (transformations) of a dataset. The output of each transform command is a new dataset. Often, you will want to perform a series of transformations in a particular order.

The **Transform** dialog window requires the following input:

- **Source Dataset:** Select the name of source dataset.
- **Transformations:** Select a transform command from the list by clicking its name.
- Then click the **Add** button.
For each transformation added, Darwin creates a tab on the dialog window where you provide information specific to that transformation.
- To remove a transform from the set, click its tab and then click **Delete**.

You can select several transformations, in any order, including repeating a transformation. If there are problems performing the specified transformations in the specified order, Darwin explains this to you.

Because most transformation datasets are needed only temporarily, they are, by default, not saved. The exception to this is the **Split** transformation, which is, by default, saved, because you use the output in building models.

If you want to save the output of a transformation that is not automatically saved, click the **Save** check box on that transformation's tab. The text box containing the suggested name for the output then becomes editable, and you can change the name. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

After you have selected and provided information specific to each transformation you want performed, click **Transform** to execute the commands. Darwin then performs all the transformations in the order indicated.

The sections below briefly define the transformations and tell you how to use the Darwin user interface to perform them. For more information about the dataset transformations themselves, see *Darwin Reference*, chapter 5.

7.3.1 Append

Creates a new dataset by appending the records of one dataset to those of another. (This is a top-to-bottom attachment, as compared to the side-to-side attachment of the **Merge** command.)

The datasets must have the same fields in the same order, and both must be either serial or distributed.

From the list displayed, select the dataset that you want appended to the source dataset.

7.3.2 Explode

Creates a series of binary fields from a single multiclass categorical field.

Specify the following:

- **Field:** Select the field whose values are to be exploded.
- **If value in field is:** Enter the value to be exploded. (**Note:** If the value is a character string, it must be enclosed in double quote marks, e.g., “1” or “one” or “one1” or “4”.)
- **Create field named:** Name of new binary field.

Example: Assume you have a field named **Color**, whose values are red, green, and blue. Use **Explode** to create three new binary fields, **Red**, **Green**, and **Blue**. The values in each of the new binary fields will be 1 or 0, for red/not-red, green/not-green, and blue/not-blue.

You can then use the **Project** transform to remove the field **Color**.

7.3.3 Merge


Combines two datasets to create a third, by attaching all fields from the records of one dataset to the records of another. (This is a side-to-side attachment, as compared to the top-to-bottom attachment of the **Append** command.)

The two source datasets must contain the same number of records. The datasets must *not* have fields with the same name; if there are duplicate names, an error message is posted.

Confirm that the **Source Dataset** box contains the name of the correct dataset. Then, from the list displayed in **Merge Source Dataset with**, select the dataset you want merged with the source dataset.

Example: As a general practice, you create a merged dataset with the input and output datasets of the prediction phase of building your model. You use this merged dataset with the **Evaluate**, **Performance**, and **Lift** analyses (these are commands on the **Analysis** menu; see chapter 9).

After you have created, tested, and performed a practice prediction with your model, the next step is to analyze the model's performance. For this step, you need a dataset that contains both the actual and the predicted values of the target field. You create this dataset by merging the input and output datasets of the prediction phase. Here are the details:

- Click **Dataset -> Transform**, or click the **Transform** icon  on the toolbar.
- On the **Transform** dialog, select **Merge**, and click **Add**.

On the **Merge** dialog,

- For **Source Dataset**, select the input dataset for the practice prediction, e.g., **dsPred**.
- For **Merge Source Dataset with**, select the output dataset from the practice prediction, e.g., **dsPred[predict]**.
- For the outcome of this transform, the **Suggested Name** is, for example, **dsPred[merge]**. If you want to change this name, click the **Save** check box to make the text box editable, and enter a name. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

The output of this transform has two fields more than the input datasets (tree models have three extra fields). The extra fields are the target field with predicted value, *targetfieldname__p*, and the relative error. For tree models, the additional column, **tree-node**, contains the number of the node that determined the prediction.

7.3.4 Missing

Creates a dataset with a new field whose value is 0 or 1, indicating that the value in a given field is present or missing, where 1 = missing and 0 = present. If it is more convenient to reverse these values, you can do so by clicking the **Missing value** box below the text boxes for **Field** and **New Field**; then 0 = missing and 1 = present.

In the **Field** list box, specify the field for which you want to determine whether values are present or missing. The **Type** (data type) of this field is indicated in a text box to the right.

The **New Field** box contains the name of the specified field with __m appended.

You can then use the **Replace** transform on the output of the **Missing** transform to substitute another value (e.g., the maximum, minimum, the average) for the fields with missing values.

7.3.5 Normalize

Creates a dataset in which all fields in the source dataset have been converted to double-precision floating-point values between 0.0 and 1.0.

7.3.6 Project

Project: Creates a dataset containing only specified fields from the old dataset.

- Confirm that the name of the correct dataset appears in the **Source Dataset** box.
- In the **Fields** box, all fields initially appear in a selected state.
 - To deselect a field, press CTRL and left-click the field's name. You can use CTRL-click to deselect multiple fields.
 - To deselect a *range* of fields, use CTRL-click for the first field, and SHIFT-CTRL-click for the last; the first, last, and all fields in between are deselected.
- Indicate by clicking the **Include** or **Exclude** option whether the fields you leave in a selected state are to be *included* or *excluded* from the output dataset.

Example Using Include: Given a dataset with fields F1 through F10, create a projected dataset containing fields F1 through F9:

- CTRL-click field F10 to deselect it.
- Select the **Include** option.

Fields F1 through F9 remain selected. Because you chose the **Include** option, the output dataset contains fields F1 through F9.

Example Using Exclude: Given the same dataset, containing fields F1 through F10, create a projected dataset containing only fields F3 and F8:

- CTRL-click field F3 and F8 to deselect them.
- Select the **Exclude** option.

All fields other than F3 and F8 remain selected. Because you chose the **Exclude** option, the output dataset contains only fields F3 and F8.

7.3.7 Randomize

Writes, in random order, all records from the source dataset into the new dataset.

The default random seed value is displayed; you can specify a different value if you wish.

Note: Because random access to ASCII text files is very slow, randomization of datasets based on such files is also slow. To improve performance, **Randomize** first makes a temporary copy of this type of dataset in virtual memory, space permitting, or creates a temporary distributed dataset file that it deletes when the operation is finished, then randomizes it. If there is no space for the temporary copy, Darwin generates an error message and does not perform the transformation. An alternative approach is to first **Save** the dataset, which creates a dataset file; then open that file and randomize it.

7.3.8 Range

Creates a new dataset from a subset of an existing dataset, using consecutive records from a specified starting point to a specified ending point.

Specify the range by either moving the slider bar or entering values in the edit boxes. Range values are exclusive.

For example, to divide a dataset into four equal subsets, you would run **Range** four times on the source dataset, specifying each range as follows:

- starting at 0 and ending at 25 percent
- starting at 26 and ending at 50 percent
- starting at 51 and ending at 75 percent
- starting at 76 and ending at 100 percent

For the second and subsequent transforms, you'll need to reselect the original dataset as the source dataset, because Darwin by default enters the output of the

last transform as the source dataset for the next transform. The assumptions guiding this default behavior are correct in most instances, i.e., transformations do build on each other, but in this particular case, it is not what you want.

7.3.9 Replace

Creates a dataset in which the values in a specified field are replaced by a different value.

Replace works by comparing a value you enter (**Current** value) to the value in a field (**Compared** field) using an arithmetic test (**Comparator**). If the test is successful (i.e., if the comparison statement is true), the value in the **Replaced** field is changed to the **New** value. The entries in **Replaced** and **Compared** may be the same field names, or different.

Under **Field Information**, specify the following:

- **Replaced:** The field whose original values are to be replaced.
- **Compared:** The field whose values are to be compared with a value you enter in the **Current** box.
- **Comparator:** The logical test to be used. Select one of the following: equal, not equal, lower than, greater than, lower or equal, greater or equal.

Under **Values**, specify the following:

- **Current:** The value to be compared with the value in the field specified in the **Compared** box.
- **New:** The new value that is to replace the original value.

(**Note:** If the value is a character string, it must be enclosed in double quote marks, e.g., “1” or “one” or “one1” or “4”.)

Examples:

- **Replace** lets you replace missing values. You perform **Replace** on the output of the **Missing** transform, and specify that missing values are to be replaced with a value you specify — you could specify, for example, the maximum, minimum, or average value for that field.
- You can also use **Replace** to accomplish binning, and reduce an ordered field to a multiclass or binary field. For example, if salary > 50,000, replace with 1; if salary ≤ 50,000, replace with 0.

To have Darwin replace missing values with a typical value, leave the **Current** and **New** fields blank. “Typical value” is the mean for floating point, rounded mean for integers, and mode for character strings.

Click **Precheck dataset** to have Darwin confirm that the target value you identified is legitimate and exists in the dataset.

7.3.10 Sample

Creates a new dataset containing a random selection of records from the original dataset. Specify the following:

- **Sample Rate Value:** Enter the percentage of the records to be included in the sample; default is 50%.
- **Random Seed Value:** Enter the random seed value. Default is 0.

7.3.11 Select

Creates a new dataset using only records in which a given field’s value meets a specified logical test.

Specify the following:

- **Field:** The field containing the value of interest.
- **Comparator:** The logical test to be used. Select one of the following: equal, not equal, lower than, greater than, lower or equal, or greater or equal.
- **Value:** The value to use in the text. (**Note:** if the value is a character string, it must be enclosed in double quote marks, e.g., “1” or “one” or “one1” or “4”.)

Example: If **Field** is age, **Comparator** is lower than, and **Value** is 37, the resulting dataset will contain all records in which the value in the age field is less than 37.

7.3.12 Set Form

Creates a new dataset in which the form of specified fields is changed from categorical to ordered and/or from ordered to categorical.

The field information displayed shows the fields and their current form. Click the check box of any field whose form you wish to change. Selected fields that are

ordered will be changed to categorical; selected fields that are categorical will be changed to ordered.

7.3.13 Split

Creates three subset datasets, each with a specified percentage of the records.

Specify the percentage of records to be used for a first, middle, and last subset. The default settings divide the dataset into thirds.

By default, the output of this transformation is saved. Because they are to be saved, you have the option of changing the names offered by Darwin, which are shown in the three text fields labeled **First**, **Middle**, and **Last**. You can change the default names to something more meaningful, such as, for example, **dsTrain**, **dsTest**, and **dsPred**. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

The option button at the right of the first text field indicates that this subset dataset is active (by default). This means that if you add another transformation after this one, this is the subset dataset on which it will be performed.

8

Model Menu



The **Model** menu contains the following commands:

- **Tree**
- **Net**
- **Match**
- **Create**
- **Copy**
- **Test**
- **Predict**

The first three commands select the type of model that is then used with the remaining four commands. For example, to create a Tree, click **Tree** and then click **Create**.

This chapter describes how to use these commands. For information about the models themselves, see *Darwin Reference*, chapters 6, 7, 8, and 9.

The information in this chapter is organized by model type (Tree, Net, Match); for each model type, the **Create**, **Test**, and **Predict** commands are described. The two remaining sections describe the **Copy** command and how to delete a model.

8.1 Tree Model



You can use the commands below to build a tree model. You can also build a tree model using the **Modeling Wizard** (see section 6.1).

The model-building process with **Tree** is diagrammed in figure 2 (page 7). For complete information about Darwin **Tree**, see *Darwin Reference*, chapters 6 and 7.

8.1.1 Create Tree Model

Click **Model** → **Tree** and **Create** or
 Click the **Tree** and **Create** icons (above) or
 Click **Model** → **Tree** and press **CTRL-L**

Creates and trains a new tree model. The dialog window prompts you for the following information:

- **Name:** Enter a name for the model. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.
- **Training Dataset:** Select the name of the dataset to be used in creating the new model, for example, **dsTrain**.
- **Target Field:** Click the name of the target field.

When you enter a name for the tree model, the **OK** button becomes available. When you click **OK**, you'll be prompted by the **Save As** dialog to save (or not) the object you are creating.

Darwin then creates the tree model, using default values for the various parameters that influence the model's performance. For access to these parameters, click the **Advanced** button. After you have built a few tree models, you may wish to experiment with setting the values for some of these parameters. The user interface for these parameters is described in section 10.1.3. See *Darwin Reference*, chapter 7, for more information.

When Darwin has built the model, its name appears in the **Workspace** listing. To view the model's properties, double-click its name in the **Workspace** listing.

8.1.2 Test Tree Model

Click **Model** → **Tree** and **Test** or
 Click the **Tree** and **Test** icons (above) or
 Click **Model** → **Tree** and press **CTRL-E**

Testing the tree model is the next step after creating it.

The **Test Tree Model** dialog prompts you for the following information:

- **Model:** Select the name of the model to be tested. The default is the current tree model.
- **Input Dataset:** Select the name of the dataset to be used in testing or evaluating the model, for example, **dsTest**.


There are several things you can do on the **Test Tree Model** dialog window: You can test/evaluate the model, you can perform a sensitivity analysis, you can view the rules by which the tree was constructed, and you can re prune the tree.

Note the two command buttons, **Evaluate** and **Reprune**, and the three tabs below them, **Evaluate**, **Sensitivity**, and **Rules**. The left command button's label changes to match the tab you have clicked. It comes up as **Evaluate**; if you click the **Sensitivity** tab, the command button changes to **Sensitivity**. If you click the **Rules** tab, it changes to **Rules**.

- **Evaluate:** To test/evaluate the model, click the **Evaluate** tab and specify the input dataset, e.g., **dsTest**. Then click the left command button, **Evaluate**. Results are displayed in a table, listing the subtrees, showing the number of nodes in each, and the relative error for each.
- **Sensitivity:** Click the **Sensitivity** tab, specify the subtree; then click the left command button (which has changed to **Sensitivity**). Sensitivity measures the relative importance of fields used in building a model. Results are displayed as a graph. For information about sensitivity analysis, see *Darwin Reference*, section 10.6.
- **Rules:** To see the rules by which the tree model makes a branching decision, click the **Rules** tab, specify the subtree; then click the left command button (which has changed to **Rules**). On the **Rules** dialog, indicate the subtree whose rules you wish to view. For information about the **Rules** command and how to read the output, see *Darwin Reference*, section 7.2.3.
- **Reprune:** Click the right command button to re prune the tree. For information about repruning, see *Darwin Reference*, sections 7.2.2 and 7.4.2.

If you re prune a tree model, Darwin uses the alternative prune function, i.e., the one that was not used in the initial pruning when creating the model. For example, assuming the initial pruning was done with the **cost** prune function (the default), Darwin would use the **gini** prune function for repruning. However, the **Advanced Options for Tree** still shows **cost** as the prune function, because this field refers only to **Create Tree**, not to **Reprune**.

To see or change settings for **Advanced Options for Tree**, click the **Advanced**

Options icon  and click the **Tree** tab. The user interface for these parameters is described in section 10.1.3. See *Darwin Reference*, chapter 7, for more information.

8.1.3 Predict with Tree Model

Click **Model** → **Tree** and **Predict** or
 Click the **Tree** and **Predict** icons (above) or
 Click **Model** → **Tree** and press **CTRL-R**

This is the third step in the process of building a model — after creating and testing, you do a practice prediction, using the third subset dataset. This is also the command you use on a brand new dataset — one whose values in the target field you do not already know.

The **Predict with Tree Model** dialog window requires the following input:

- **Name:** Select the name of the model. The default is the current tree model.
- **Subtree:** Indicate the number of the subtree you wish to use for the prediction.
- **Input Dataset:** Select the name of the input dataset, for example, **dsPred**.
- **Output Dataset:** Name of the output dataset. Darwin offers a default name, which you can accept or change. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.


When you click the **Predict** button, a **Save As** dialog appears and gives you the option of saving the object.

The output of this command, a new dataset, is then displayed, showing is then displayed, showing predicted values, the relative error, and the number of the tree node that determined the prediction.

Reprune: You can also reprune your tree model from this dialog window by clicking the **Reprune** button. For information about repruning, see *Darwin Reference*, sections 7.2.2 and 7.4.2.

If you reprune a tree model, Darwin uses the alternative prune function, i.e., the one that was not used in the initial pruning when creating the model. For example, assuming the initial pruning was done with the **cost** prune function (the default), Darwin would use the **gini** prune function for repruning. However, the **Advanced Options for Tree** will still show **cost** as the prune function, because this field refers only to **Grow Tree**, not to **Reprune**.

To see or change settings for **Advanced Options** for **Tree**, click the **Advanced**

Options icon  and click the **Tree** tab. The user interface for these parameters is described in section 10.1. See *Darwin Reference*, chapter 7, for more information.

Next Step: Merge

The next step is to evaluate and analyze the model's performance to see how accurate the predictions were, using the commands on the **Analysis** menu (see chapter 9). For this, you will need a dataset that contains both the actual and predicted values for the target field.

You create this dataset by merging the input and output datasets from this prediction step, using the **Merge** command (a **Transform**) (see section 7.3.3).

8.2 Net Model



You can use the commands below to build a net model. You can also build a net model using the **Modeling Wizard** (see section 6.1).

The model-building process with **Net** is diagrammed in figure 3 (page 8). For complete information about Darwin **Net**, see *Darwin Reference*, chapters 6 and 8.

8.2.1 Create Net Model



Click **Model** -> **Net** and **Create** or
Click the **Net** and **Create** icons (above) or
Click **Model** -> **Net** and press **CTRL-L**

Creates and trains a neural net model. The dialog window prompts you for the following information:

- **Name:** Enter a name for the model. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.
- **Training Dataset:** Select the name of dataset to be used in creating and training the new model, for example, **dsTrain**.
- **Target Field(s):** Click the name(s) of the target field(s). You can have more than one target field with a net model, although it's rare that this is wanted. (Do not confuse this with multiple target values, which all three types of models can handle.)

Net models work best when there are roughly equal numbers of target values in the dataset. If your source dataset is unbalanced, you may wish to perform the

appropriate transformations on the dataset so as to balance the number of target values. See the example of multiple transformations in section 5.3.1 of *Darwin Reference*.

When you enter a name for the net model, the **OK** button becomes available. When you click **OK**, you'll be prompted by the **Save As** dialog to save (or not) the object you are creating.

Darwin then creates the model, using default values for the various parameters that affect building and training the model. For access to these parameters, click the **Advanced** button (see sections 10.1.4 and 10.1.5).

When Darwin has created the model, a table appears, showing the relative training error at each iteration (the relative training error is usually much larger than the absolute error, so don't be alarmed). The name of the model also then appears in the **Workspace** listing. To view the model's properties, double-click its name in the **Workspace** listing.

After you have created a net model, the next step is either to test it or to predict with it, depending on whether testing was built into your creating and training phase. Testing can be part of creating and training the net if you select certain options on **Advanced Options, Net Train**:

- If on **Advanced Options, Net Train**, you used **simple Training** to train the net, the next step is to test it, as described **Test Net Model**, below.
- If on **Advanced Options, Net Train**, you selected **Test and Train** (the default) or **Cross-Validation** to train the net, skip the testing phase and go directly to **Predict with Net**.

Advanced Options: The parameters that influence the net model's performance are grouped in **Advanced Options** as having to do with building the model or training it. For access to these parameters, click the **Advanced** button, which takes you to **Advanced Options, Net Build**. From there, click the **Net Train** tab to see the parameters relevant to training the net.

After you have built a few net models, you may wish to experiment with setting values for some of these parameters. See *Darwin Reference*, chapter 8, for detailed information.

Note: On **Advanced Options, Net Train**, you can specify the number of iterations for training the net. If you specify a small number of iterations and later decide you want to continue training beyond that number, you can do so. See **Continue Training Net Model**, section 8.2.4, below.

8.2.2 Test Net Model



Click **Model** -> **Net** and **Test** or
Click the **Net** and **Create** icons (above) or
Click **Model** -> **Net** and press **CTRL-E**

Testing the net is the next step after creating and training it.

If you used **Train and Test** or **Cross-Validation** to train the net, you have already tested the net, and you can skip this step and go to **Predict with Net**, section 8.2.3.


Otherwise, you can try such utilities as **Sensitivity** or experiment with the parameters available on the **Advanced Options** dialog.

The **Test Net Model** dialog prompts you for the following information:

- **Model:** Select the name of the model to be tested. The default is the current net model.
- **Input Dataset:** Select the name of the dataset to be used in testing or evaluating the model, for example, **dsTest**.

Note the two command buttons, **Evaluate** and **Perturb**, and the two tabs below them, **Evaluate** and **Sensitivity**. The left command button's label changes to match the tab you have clicked. It comes up as **Evaluate**; if you click the **Sensitivity** tab, the command button changes to **Sensitivity**.

The **Test Net Model** dialog has three command options: **Evaluate**, **Sensitivity**, and **Perturb**:

- **Evaluate:** To test/evaluate the model, click the **Evaluate** tab and specify the input dataset, e.g., **dsTest**. Then click the left command button, **Evaluate**. Results are displayed in a table listing the evaluation error, the training error, cost, and the evaluation dataset size.
- **Sensitivity:** Click the **Sensitivity** tab, then click the left command button (which has changed from **Evaluate** to **Sensitivity**). **Sensitivity** measures the relative importance of fields used in building a model. For more information about **Sensitivity**, see *Darwin Reference*, section 10.6.
- **Perturb:** This command applies to retraining a net model. You might want to retrain if you have already trained and tested a net model and you would like to start over, using different starting weights. On the **Options** menu, click **Advanced** (or click the **Advanced** icon ) and click the

Net Train tab. Under **Re-Train**, indicate the degree of perturbation by selecting a value between 0 and 1. Zero gives you exactly the same weights to start with as before; 1 gives you a completely modified set of weights. See section 8.2.5 and *Darwin Reference*, section 8.4, for more information about perturbation and retraining a net.

If, after you have created and trained a net model, you wish to continue training that model, e.g., for more iterations than you initially specified, see **Continue Training Net Model** (section 8.2.4). If you want to continue training, but with slightly different weights, see **Perturbation** (section 8.2.5).

After you have tested the net model, the next step is as practice prediction, as described in **Predict with Net Model**.

8.2.3 Predict with Net Model



Click **Model -> Net** and **Predict** or
Click the **Net** and **Predict** icons (above) or
Click **Model -> Net** and press **CTRL-R**

This is the third step in building a net model — after creating, training, and testing (or training and testing together), you do a practice prediction, using the third subset dataset. The **Predict with Net** command is also the command you use on a new dataset — one whose values in the target field you do not already know.

The **Predict with Net Model** dialog requires the following input:

- **Name:** Select the name of the model. The default is the current net model.
- **Input Dataset:** Select the name of the input dataset, for example, **dsPred**.
- **Output Dataset:** Name of the output dataset. Darwin offers a default name, which you can accept or change. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.

When you click the **Predict** button, a **Save As** dialog appears and gives you the option of saving the object.

The output of this command is a new dataset, which is displayed, showing predicted values for each target field, the confidence of that prediction (expressed as a value between 0 and 1), and the probability. In cases in which the target field can take multiple values, Darwin specifies multiple probabilities.

Perturb: This command applies to retraining a net model. You might want to retrain if you have already trained and tested a net model and you would like to start over, using different starting weights. On the **Options** menu, click

Advanced (or click the **Advanced** icon ) and click the **Net Train** tab.

Under **Re-Train**, indicate the degree of perturbation by selecting a value between 0 and 1. Zero gives you exactly the same weights to start with as before; 1 gives you a completely modified set of weights. See section 8.2.5 and *Darwin Reference*, section 8.4, for more information about perturbation and retraining a net.

Next Step: Merge

The next step is to evaluate and analyze the model's performance to see how accurate the predictions were, using the commands on the **Analysis** menu (see chapter 9). For this, you will need a dataset that contains both the actual and predicted values for the target field.

You create this dataset by merging the input and output datasets from this prediction step, using the **Merge** command (a **Transform**) (see section 7.3.3).

8.2.4 Continue Training Net Model

There are circumstances under which you may decide you want to continue training a net model that you have already trained. This can be the case if, for example, you specified a small number of iterations for training (on the **Advanced Options** for **Net Train**) and at the end of training you see that the error and testing rates are still decreasing, which means the model had not yet reached its best state.

To continue training a net model:

- Click the **Create Net** and enter the same net model name. Remember to specify the target variable again.
- Click **Advanced Options, Net Train**, change the number of iterations, if desired, and click **OK**.
- **Advanced Options, Net Train**, goes away, and you are back at the **Create Net** dialog. If everything looks okay, click **OK**.
- Darwin prompts you with a message that says "Net already exists! Do you want to continue training?" Click **Yes** to continue training.
- Darwin prompts you to save or not; click **Yes** or **No**.
- Darwin starts training.

When the net has trained for the specified number of iterations, Darwin displays a table that shows the error rate(s) for this second training session.

If you compare the tables from the first and second training sessions, you will see that the first error rate(s) from the second session is/are the same as the last error rate(s) from the first session. Note that the numbering of iterations starts over with 0.

If you want to continue training in connection with the **Perturb** option, i.e., with different starting weights, see **Perturbation** (section 8.2.5).

8.2.5 Perturbation

The result of training a net is a set of weights that are used in the calculations that determine a prediction. The term for manually changing the trained weights is *perturb*. See *Darwin Reference*, section 8.4.2, for more information.

To continue training, but with perturbed weights (re-train):

1. Click **Advanced Options, Net Train**. In the **Re-Train** section, at the bottom, move the **Perturb** slider as far to the right as possible (100% perturbation), and click **OK**.
2. On the **Test Net Model** or the **Predict with Net Model** dialog, click **Perturb**.
3. Click **Model -> Net, Model -> Create**, and enter the name of the trained net in the **Name** box (and make any desired changes in **Advanced Options**). Remember to specify the target field.
4. At the message "Net already exists! Do you want to continue training?", click **Yes**.

To continue training keeping the current weights, skip steps 1 and 2.

To continue training, but with slightly perturbed weights:

1. Click **Advanced Options, Net Train**, and go to the **Re-Train** section. Move the **Perturb** slider 5% to 10% of the distance to the right.
2. On the **Test Net Model** or the **Predict with Net Model** dialog, click **Perturb**.
3. Click **Model -> Net, Model -> Create**, and enter the name of the trained net in the **Name** box (and make any desired changes in **Advanced Options**). Remember to specify the target field.
4. At the message "Net already exists! Do you want to continue training?", click **Yes**.

8.3 Match Model

You can use the commands below to build a match model. You can also build a match model using the **Modeling Wizard** (see section 6.1).

The model-building process with Darwin **Match** is diagrammed in figure 4 (page 9). For complete information about Darwin match models, see *Darwin Reference*, chapters 6 and 9.

8.3.1 Create Match Model

Click **Model** → **Match** and **Create** or
Click the **Match** and **Create** icons (above) or
Click **Model** → **Match** and press **CTRL-L**

Creates and trains a new match model. The **Create Match Model** dialog prompts you for the following information:

- **Name:** Enter a name for the model. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.
- **Training Dataset:** Select the name of dataset to be used as the model dataset (see figure 4, on page 9), for example, **dsTrain**.
- **Target Field:** Click the name of the target field.
- **Target Value:** Enter the positive target value if target field is categorical (This field is greyed out if the target field is ordered.)

When you enter a name for the match model, the **OK** button becomes available. When you click the **OK** button, you'll be prompted by the **Save As** dialog to save (or not) the object you are creating.

Darwin then creates the match model, using default values for the various parameters that influence the model's performance. For access to these parameters, click the **Advanced** button. After you have built a few match models, you may wish to experiment with setting the values for some of these parameters. The user interface for these parameters is described in section 10.1.6. See *Darwin Reference*, chapter 9, for more information about the parameters themselves.

When Darwin has built the model, its name appears in the **Workspace** listing. To view the model's properties, double-click its name in the **Workspace** listing.

After you have created the match model, the next step is to test it, as described below, in section 8.3.2.

8.3.2 Test Match Model



Click **Model** -> **Match** and **Test** or
 Click the **Match** and **Test** icons (above) or
 Click **Model** -> **Match** and press **CTRL-E**

Testing the match model is the next step after creating it.


The **Test Match Model** dialog prompts you for the following information:

- **Model:** Select the name of the model to be tested. The default is the current match model.
- **Input Dataset:** Select the name of the dataset to be used in testing the model, for example, **dsTest**.

There are several things you can do on the **Test Match Model** dialog: You can test/evaluate the model, perform a sensitivity analysis, view neighbors, and optimize the model:

Note the two command buttons, **Evaluate** and **Optimize**, and the three tabs below them, **Evaluate**, **Sensitivity**, and **Neighbors**. The left command button's label changes to match the tab you have clicked. It comes up as **Evaluate**; if you click the **Sensitivity** tab, the command button changes to **Sensitivity**. If you click the **Neighbors** tab, it changes to **Neighbors**.

- **Evaluate:** To test/evaluate the model, click the **Evaluate** tab and specify the input dataset, e.g., **dsTest**. Then click the left command button, **Evaluate**. Results are displayed in a table showing the error rate and the RMS (root mean square).
- **Sensitivity:** Click the **Sensitivity** tab, then click the left command button (which has changed from **Evaluate** to **Sensitivity**). **Sensitivity** measures the relative importance of fields used in building a model. Results are displayed as a graph. For more information about **Sensitivity**, see *Darwin Reference*, section 10.6.
- **Neighbors:** For information about the neighbors, click the **Neighbors** tab, then click the left command button (which has changed to **Neighbors**). Results are displayed in a table showing, for each neighbor, distance from the target and its target value. See *Darwin Reference*, section 9.3.2, for more information.
- **Optimize:** Click the **Optimize** button to have Darwin optimize the match model. Optimizing means adjusting the match weights to improve their predictive value. See *Darwin Reference*, section 9.3, for more information.

To see **Advanced Options for Match**, click **Advanced** on the **Options** menu, (or click the **Advanced Options** icon ) and then click the **Match** tab.

After you have tested the match model, the next step is a practice prediction, as described in **Predict with Match Model**, below.

8.3.3 Predict with Match Model



Click **Model** → **Match** and **Predict** or
Click the **Match** and **Predict** icons (above) or
Click **Model** → **Match** and press **CTRL-R**


This is the third step in building a model — after creating and testing, you do a practice prediction, using the third subset dataset. This is also the command you use on a new dataset — one whose values in the target field you do not already know.

The **Predict with Match Model** dialog requires the following input:

- **Name:** Select the name of the model. The default is the current match model.
- **Input Dataset:** Select the name of the input dataset, for example, **dsPred**.
- **Output Dataset:** Name of the output dataset. Darwin offers a default name, which you can accept or change. See section 3.3.4 to learn what characters are permitted in names of Darwin objects.

There are two command buttons: **Predict** and **Optimize**.

- **Predict:** When you click the **Predict** button, a **Save As** dialog window appears and gives you the option of saving the object.
- **Optimize:** You can also click **Optimize** to have Darwin adjust the match weights to improve their predictive value. See *Darwin Reference*, section 9.3.1, for more information.

To see **Advanced Options for Match**, click **Advanced** on the **Options** menu, (or click the **Advanced Options** icon ) and then click the **Match** tab.

The output of this command, a new dataset, is then displayed, showing for each record the predicted value and the level of confidence in that prediction, expressed as a value between 0 and 1.

Next Step: Merge

The next step is to evaluate and analyze the model's performance to see how accurate the predictions were, using the commands on the **Analysis** menu (see chapter 9). For this, you will need a dataset that contains both the actual and predicted values for the target field.

You create this dataset by merging the input and output datasets from this prediction step, using the **Merge** command (a **Transform**) (see section 7.3.3).

8.4 Copy

Click **Model -> Copy**

The **Copy** command lets you make a copy of a model (the **Copy** command on the **Edit** menu works only for text).

The **Copy Model** dialog prompts you for the following information:

- **Source Model:** Select the name of the model you wish to copy. The models listed are those in the current project.
- **Model Type:** Type of model (Tree, Net, or Match) appears here.
- **Copy Name:** Enter the name you wish to give to the copy. Darwin offers a default name, which you can keep or change. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

Click **OK** to proceed, or **Cancel** to cancel the command.

8.5 Deleting a Model

To delete a model:

- Right-click the model's name in the **Workspace** listing to bring up the context menu, and click **Delete**. A dialog window appears and asks whether you really want to delete the named model.

9

Analysis Menu




The commands on the **Analysis** menu provide different methods of analyzing the performance of Darwin models.

This chapter describes the mechanics of using the commands on the **Analysis** menu. See *Darwin Reference*, chapter 10, for more information about the analyses themselves.

The commands on the **Analysis** menu are as follows:

- **Evaluate**
- **Summarize**
- **Frequencies**
- **Performance**
- **Lift**

You can graph the results of any of these analyses by clicking **Project -> Graph** or by clicking the **Graph** icon  in the toolbar. See section 4.9 for details.

Three of the analysis commands, **Evaluate**, **Performance**, and **Lift**, work on a dataset that contains both the actual and predicted values for the target field. To create this dataset, merge the input and output prediction datasets of the prediction phase of building the model — i.e., the dataset that you used as input for the **Predict** command and the output of that command, which was a new dataset. Use the **Dataset** menu's **Transform** command **Merge** to combine these two datasets. See section 7.3.3 for details.

9.1 Evaluate

Click **Analysis -> Evaluate** or
Click the **Evaluate** icon (above) or
Press **CTRL-1**

The **Evaluate** command evaluates the results of predictions made by a Darwin model. Specifically, it compares the actual and predicted values in the target field.

Before using this command, merge the input and output datasets from the prediction phase of building a model. The merged dataset is the input dataset for this analysis. See **Merge** (section 7.3.3).

The **Evaluate** dialog window requires the following input:

- **Dataset:** Select the name of the merged dataset.
- **Actual:** Select the name of target field containing actual values.
- **Predicted:** Select the name of target field containing predicted values (typically, it's the name of the actual target field with `__p` appended.)
- **Add:** Click **Add** to bring the names of the actual and predicted target fields to the **Source** and **Target** window.
- **Table:** Darwin offers a name for the output file. You can accept the name or change it. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

After you have supplied the requested information, click **Evaluate**. A **Save As** dialog appears and asks whether you want to save the results, and offers a default name, which you can accept or change (see section 3.3.4).

Results are then displayed in a table showing, for each selected field, the following: NULLs; maximum, minimum, and absolute error; and RMS (root mean square) error for the predicted values.

See *Darwin Reference*, section 10.1, for more information about the **Evaluate** analysis.

9.2 Summarize

Click **Analysis -> Summarize** or
Click the **Evaluate** icon (above) or
Press **CTRL-2**

The **Summarize Data** command provides a statistical summary of the values taken by data in the specified fields of a given dataset.

The dialog prompts you for the following information:

- **Dataset:** Select the name of the source dataset. The default is the current dataset.
- **Fields:** Select the fields of interest. By default, all fields are selected.
 - To deselect a field, press CTRL and click the field. Use CTRL-click to deselect multiple fields.
 - To deselect a *range* of fields, use CTRL-click for the first field, and SHIFT-CTRL-click for the last; the first, last, and all fields in between are deselected.
- **Table:** Results are displayed in a table. Darwin offers a name for the table, which you can accept or change. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

After you have supplied the requested information, click **Summarize**. A **Save As** dialog appears and asks whether you want to save the results.

The results table is then displayed, showing, for each selected numerical field, the following: datatype; form (categorical or ordered); number of NULLs; maximum, minimum, and average value; average squared; and standard deviation. For string fields, the results displayed are datatype, form, and number of NULLs.

See *Darwin Reference*, section 10.2, for a more information about the **Summarize Data** analysis.

9.3 Frequencies



Click **Analysis -> Frequencies** or
Click the **Frequencies** icon (above) or
Press **CTRL-3**

The **Frequencies** analysis provides information on the frequency with which particular data values appear in a dataset. You can specify two fields, e.g., the actual target field and the predicted target field.

You can also use **Frequencies** to determine the distribution of values in a single field of the dataset. Click the **Get frequencies from Field 2** check box to disable reference to a second field.

The **Frequency** dialog requires the following input:

- **Dataset:** Select the name of the merged dataset.
- **Field 1:** Name of the actual target field.
- **Field 2:** Name of the predicted target field.
- **Table:** Darwin offers a name for the output file. You can accept the name or change it. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

After you have supplied the requested information, click **Count**. A **Save As** dialog appears and asks whether you want to save the results, and offers a default name, which you can accept or change (see section 3.3.4).

Results are then displayed in a table showing a frequency distribution for minimum and maximum error, NULLs, and each value in the target field. For a single field, the results displayed are minimum and maximum error and count.

Advanced Options for **Analysis** provides options relevant to **Frequencies** (see section 10.1.7). See *Darwin Reference*, section 10.3, for more information about the **Frequencies** analysis.

9.4 Performance

Click **Analysis** -> **Performance** or
Click the **Performance** icon (above) or
Press **CTRL-4**

The **Performance Matrix** command provides information on the frequency of particular data values in a dataset. It is most useful for models with a continuous target value.

Performance expects two fields as input. The fields may be manipulated through functions that compute the values.

Before using this command, merge the input and output datasets from the prediction phase of building a model. The merged dataset is the input dataset for this analysis. See **Merge** (section 7.3.3).

The **Performance** dialog requires the following input:

- **Dataset:** Select the name of the merged dataset.
- **Field 1:** Select the name of the first field (e.g., target field with actual values).
- **Function 1:** Select the function for the value in Field 1. Choices are:
 - Use Field 1 value
 - Use Field 2 value
 - Value of Field 2 minus Field 1
 - Absolute of Field 2 minus Field 1
 - Field 2 minus Field 1 over Field 1
 - Field 2 minus Field 1 over Field 2
- **Field 2:** Select the name of second field (e.g., target field with predicted values).
- **Function 2:** Select the function for the value in Field 2. Choices are the same as for **Function 1**.
- **Table:** Darwin offers a name for the output file. You can accept the name or change it. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.

After you have supplied the requested information, click **Performance**. A **Save As** dialog appears and asks whether you want to save the results, and offers a default name (see section 3.3.4).

Results are then displayed in a table showing a frequency distribution for minimum and maximum error, NULLs, and each value in the target field.

The **Advanced Options** page provides options relevant to this analysis. You can specify starting and stopping values, steps, and bins. Specify these values for two fields (see section 10.1.7). See *Darwin Reference*, section 10.4, for more information.

9.5 Lift

Click **Analysis** -> **Lift** or
Click the **Lift** icon (above) or
Press **CTRL-5**

Lift represents the benefit gained from a model's predictive ability. It can be thought of as the profits accruing from an accurately targeted campaign: for example, sending a direct mailing to only that 20% of customers who will provide 90% of the positive response. Lift tells you how much better you can do using the model compared to doing a random mailing.

Before using this command, merge the input and output datasets from the prediction phase of building a model. The merged dataset is the input dataset for this analysis. See **Merge** (section 7.3.3).

The **Lift** dialog requires the following information:

- **Dataset:** Select the name of the merged dataset.
- **Table:** Suggested name for the output table. You can accept this name or change it. See section 3.3.4 to learn what characters are permitted in the names of Darwin objects.
- **Actual Field:** Select the name of the target field with actual values.
- **Predicted Field:** Select the name of the target field with predicted values. This will be the name of the target field with `__p` appended.
- **Profit:** Enter a value that represents the potential revenue for a correct prediction. The default is 1.

- **Cost:** Enter a value that represents the potential cost of an incorrect prediction. The default is -1.
- **Target Value:** You must enter a value here; **Lift Computation** requires a target value.

The **Lift** button remains unavailable until you supply a target value. When you have supplied all the necessary input, click **Lift**. A **Save As** dialog appears and asks whether you want to save the results, and offers a default name (see section 3.3.4).

Results are then displayed in a table showing, for each quantile of the dataset, the following: lift, margin, ROI, confidence, and information about targets and non-targets. See *Darwin Reference*, section 10.5.3, for an explanation of how to read the output.

The **Advanced Options** page provides options relevant to **Lift**. You can specify the following: number of quantiles, target expansion value, and confidence. See section 10.1.7.

See *Darwin Reference*, section 10.5, for more information about lift analysis.

10 Options Menu

The **Option** menu contains the following commands:

- **Advanced**
- **Macro**
- **Code Generation**

10.1 Advanced

Click **Options -> Advanced** or
Click the **Advanced Options** icon (above) or
Press **CTRL-A**

The **Advanced Options** dialog window has eight tabbed pages:

- **Project**
- **Datasets**
- **Tree**
- **Net Build**
- **Net Train**
- **Match**
- **Analysis**
- **Setup**

10.1.1 Project

The **Project** tab of **Advanced Options** is a bookkeeping and recordkeeping dialog. It provides a place to store the following information about a given project:

- the project's name
- name of project leader
- phone number of the project leader
- business problem
- project's objectives
- whether the project is shared
- any notes you may wish to make

10.1.2 Datasets

The **Datasets** tab of **Advanced Options** is where you specify the separator (delimiter) used in the descriptor file and whether the dataset file is to be serial or distributed.

Descriptor:

- **Separator:** Select the separator used in the dataset.

Serial or Distributed: These fields are enabled if you are logged on to a multi-node system:

- **Name:** Name of dataset.
- **File:** Name of text file, with an extension that indicates whether the file is distributed or serial.
- **Serial/Distributed:** If the button is labeled **Serial**, clicking it changes the file to serial. If the button is labeled **Distributed**, clicking it changes the file to distributed. If the dataset already exists, you can use this selection to create a copy of it that is distributed. (You end up with, for example, **wind.ds** and **wind.ds[distributed]**.)

When you have finished, you can click **OK** to confirm your choices. Click **Cancel** to cancel your choices and dismiss the window. Click **Reset** to restore the default settings.

10.1.3 Tree

There are several parameters that influence the performance of a tree model. These parameters have to do with controlling the size of the tree and influencing the way the splits are made.

Thresholds:

- **Density:** Sets a lower limit on the percentage of records with each target value on a non-leaf node. Can be set to any number between 0 and 1. Default is 0.05. See *Darwin Reference*, section 7.3.1, for more information.
- **Max Nodes:** It is sometimes useful to set a limit on the maximum number of nodes the tree can contain in order to prevent early tree models from being overly large. The asterisk in this field means there is no default setting. See *Darwin Reference*, section 7.3.2, for more information.

The **Functions** and **Files** parameters apply only to categorical fields:

Functions: Select decrease and prune functions from options Darwin offers.

- **Decrease:** Determines how branching is done. Choices are **gini** (the default) and **entropy**. See *Darwin Reference*, section 7.4.1, for more information.
- **Prune:** Influences pruning decisions. Choices are **cost** (the default) and **gini**. See *Darwin Reference*, section 7.4.2, for more information.

If you re prune a tree model, Darwin uses whichever prune function was not used in the initial pruning when creating the tree. For example, assuming the initial pruning was done with the default prune function, **cost**, Darwin would use the **gini** prune function in re pruning. However, the **Advanced Options** for **Tree** will still show **cost** as the prune function, because this field refers only to **Create Tree**, not to **Reprune**.

Files: These are files you create using **Project -> New File**.

- **Priors:** A **Priors** file specifies the expected distribution of target values within the population to be predicted. Used only when actual distribution does not match distribution within the training dataset. (Training datasets may be skewed toward more equal distribution to improve training efficiency.) If no **priors** are given, Darwin assumes that the training dataset accurately reflects the distribution of values. See *Darwin Reference*, section 7.4.3, for more information.
- **Costs:** A **Costs** file contains a matrix that defines the costs of incorrect predictions. Rows are predicted values, *i*. Columns are actual values, *j*. **Cost** is the cost of predicting value *i* when the actual value is *j*. Correct

predictions must have a cost of 0. See *Darwin Reference*, section 7.4.4, for more information.

- **Use Costs to modify Priors:** If you want to use **Costs** to modify **Priors**, click this check box.

To create a **cost** or **priors** file, use the **Project** menu's **New File** command (section 4.6). Click **Darwin File**, and then select the type of file you wish to create. When you click **OK**, a template for that file appears in the document area. Modify it as necessary for your purposes.

When you have finished supplying the necessary information, click **OK** to confirm your choices. You can also click **Cancel** to cancel your choices and dismiss the window, or **Reset** to restore the default settings.

If you have come to the **Advanced Options** for **Tree** from the **Create Tree Model** dialog, clicking **OK** will dismiss the **Advanced Options** dialog and return you to the **Create Tree Model** dialog.

10.1.4 Net Build

The parameters on the **Net Build** dialog have to do with building the net model, i.e., designing its topology: the number of hidden layers, the number of units per layer, and the activation functions for the hidden and output layers.

The **Net Build** dialog is divided into sections. The top section, **Layers**, is divided into two subsections, **Hidden** and **Output**. Below the **Layers** section is **Weight**.

For the **Hidden** layer, specify the following (reading across):

- **Number:** Number of hidden layers (default is 1).
- **Units per Layer:** Number of nodes in the hidden layer. The default, indicated by an asterisk (*), is to have the same number of nodes as are in the input layer.
- **Activation Function:** Sigmoid is the default. Alternatives are Hypertangent and Linear.
- **Optimize:** Check the check box to have Darwin find the optimal size of the hidden layer, i.e., the number of units in the hidden layer. This is likely to take a while.

For the **Output** layer, specify the following (reading across):

- **Activation Function:** There are different activation functions for categorical and ordered data:
 - **Categorical:** Default is Sigmoid. Alternative is Hypertangent.
 - **Ordered:** Default is Linear. Alternatives are Sigmoid and Hypertangent.

For **Weight:**

- Weight is specified as a single number x , with values ranging from $-x$ to $+x$. As a starting point for training, Darwin provides a default set of random weights between -1 and 1 .

When you have finished supplying the necessary information, click **OK** to confirm your choices. You can also click **Cancel** to cancel your choices and dismiss the window, or **Reset** to restore the default settings.

See *Darwin Reference*, section 8.2, for more information.

Your next step is to specify how to train the net model.

10.1.5 Net Train

The **Net Train** dialog shows two sets of parameters, one for training the net (**Train**) and another for retraining the net (**Re-Train**).

For **Net Train**, the parameters are these:

- **Mode: Mode** refers to **Learning Mode**, which refers to the mode of learning for the net. There are three modes:
 - **Train and Test**, which trains and tests simultaneously, using either two datasets or two parts of one dataset.
 - **Cross-Validation**, which trains and tests with two datasets or two parts of one dataset, and then repeats the process, swapping the two parts. This is a good option when working with amounts of data that are too small to allow you to split a dataset into three subsets. You can use this training mode to test out different network structures to find the most promising model.
 - **Simple Training**, which trains the network for the specified number of iterations. You must then test the model.

- **Train-Test Ratio:** If you select Train and Test and use a single dataset, specify the proportion to use for training and for testing by specifying a number between (but not including) 0 and 1. The default ratio is 0.5, which means half the dataset will be used for training and half for testing.
- **Dataset:** If you select **Train and Test** and use a second dataset for testing, enter the name of the second dataset, e.g., **dsTest**.
- **Algorithm:** Specify the training algorithm: Conjugate gradient is the default. Alternatives are modified newton, steepest descent, backpropagation, and genetic algorithm.
- **Learn Rate:** If you choose back propagation or genetic as your training algorithm, specify the learning rate. Set a value from 0.01 to 0.05; default is 0.05.
- **Cost Function:** Square (the default), cross-entropy, or p -norm.
- Value of p -norm (if you elect p -norm as the cost function):
- **Iterations:** Specify the number of iterations the net is to train.

See *Darwin Reference*, section 8.3, for more information about these parameters.

Note: If, after you have created and trained a net model, you wish to continue training that model, e.g., for more iterations than you initially specified, see **Continue Training Net Model**.

For **Re-Train**, the parameters are these:

- **Perturb or Restore:** Select **Perturb** if you want to modify the starting weights, which means to start over and train the net again, but with different starting weights. Select **Restore** to restore the original weights.
- **Degree of Perturbation:** Specify a value between 0 and 1. Zero gives you exactly the same weights to start with as before; 1 gives you a completely modified set of weights.

When you have finished supplying the necessary information, click **OK** to confirm your choices. You can also click **Cancel** to cancel your choices and dismiss the window, or **Reset** to restore the default settings.

If you have selected to the **Advanced Options Net Build** or **Net Train** from the **Create Net Model** dialog, clicking **OK** will dismiss the **Advanced Options** dialog and return you to the **Create Net Model** dialog.

See *Darwin Reference*, section 8.4, for more information.

10.1.6 Match

There are several parameters that influence a match model's performance. After you have built a few match models, you may wish to experiment with setting the values for some of these parameters.

- **Neighbors:** This is k , the number of nearest neighbors to use when making a prediction. The default value is 2; you can specify a different value to see whether it produces better predictions. See Darwin Reference, section 9.3.2.
- **Weights:** You can specify an existing weights file, if you have created one (use the **Project** menu's **New File** command, section 4.6). Otherwise, Darwin creates a default weights file in which all fields are given equal weight. See Darwin Reference, section 9.3.1.
- **Bias:** A number between 0 and 1, representing the degree to which Darwin should bias predictions. The default is 0.5, which creates no bias. Negative bias grows stronger as the bias number approaches 0; positive bias grows stronger as the bias number approaches 1. See Darwin Reference, section 9.3.3.
- **Verify presence of positive values:** Click the check box to have Darwin verify that the target value exists in the dataset. The default is not to verify.

When you have finished supplying the necessary information, click **OK** to confirm your choices. You can also click **Cancel** to cancel your choices and dismiss the window, or **Reset** to restore the default settings.

If you have come to the **Advanced Options** for **Match** from the **Create Match Model** dialog, clicking **OK** will dismiss the **Advanced Options** dialog and return you to the **Create Match Model** dialog.

10.1.7 Analysis

The **Analysis** tab of the **Advanced Options** dialog lets you adjust the parameters of **Frequencies**, **Performance**, and **Lift**, as follows:

- For **Frequencies** and **Performance**, you can specify
 - starting and stopping values (for two fields)
 - steps (for two fields)
 - bins (for two fields)

See *Darwin Reference*, sections 10.3 and 10.4 for more information.

- For **Lift**, you can specify the following:
 - **Quantiles:** The number of quantiles; default is 10.
 - **Target Expansion:** A value representing the amount by which Darwin should correct for differences in the composition of the evaluation and prediction datasets. Expansion is calculated as the number of target records in the testing dataset divided by the expected (estimated) number of target records in the prediction dataset. By default, this field contains an asterisk (*), which means no correction.
 - **Confidence:** Specify the correct name for this field:
 - For a tree or match model or for a net model with a single target field, the default, `confidence`, is correct.
 - For a net model with more than one target field, enter `confidence_targetfieldname` where `targetfieldname` is the name of the target field.

When you have finished supplying the necessary information, click **OK** to confirm your choices and go to the next step. Click **Cancel** to cancel your choices and dismiss the window, or **Reset** to restore the default settings.

See *Darwin Reference*, section 10.5, for more information.

10.1.8 Setup

The **Setup** tab lets you specify the following options:

- **Names:** For names, you can specify whether to
 - Create names automatically
 - Enable renaming
- **Editing Text Files:** By default, editing is not enabled. Click the check box to enable editing.
- **Workflow:** Specify the way you want **Workflow** to display:
 - Enhanced display
 - Standard display
 - No actions
- **Save to a File:** Specify the way you want objects saved:
 - Always save objects
 - Prompt to save objects
 - Never save objects

To change the setting for an item, click its check box. Then, click **OK**. You can also click **Cancel** or **Reset**.

10.2 Macro

Click **Options -> Macro** or
Click the **Macro** icon (above) or
Press **CTRL-M**

With a macro (command log), you can keep track of and record the series of commands you issue and use the macro to re-execute those commands at a later time.

Macros are also useful for analyzing what goes wrong in a session, because you can more easily track bugs and other problems.

The **Macro** dialog requires the following input:

- In the top field, enter either a name for the new macro, if you are recording one (see section 3.3.4 to learn what characters are permitted in names of Darwin objects), or enter the name of an existing macro if you want to run one. The filename appears below the text box; this is the name you entered, with a `.log` suffix added.

When you have supplied all the required information:

- Click **Record** to record a new macro. The **Macro** dialog disappears. Then begin executing the series of commands or whatever it is you wanted to record. To stop recording, click the **Macro** icon again (or, on the **Options** menu, click **Macro**), and on the dialog, click **Stop**.
- Click **Run** to execute an existing macro.
- Click **Close** to cancel the command.
- Click **Contents** to view a macro's log file.

Notes:

- Macros are specific to a project. If you are looking for a particular macro, you need to know what project it belongs to.
- Do not try to change from one project to another while recording a macro. Close the macro first, and then start another macro.

- If you want connect to a database using a macro, connect to the database before starting or executing the macro. **Database Connect** commands are not saved in a macro.
- Similarly, **Exit** commands are not saved in macros. The presence of an **Exit** command in a macro causes an error.

10.3 Code Generation

Click **Options -> Code Generation** or
Press **F8**

This command is not available for all versions of Darwin.

The **Code Generation** command allows you to generate C, C++, or Java code for tree and net models so that you can include the models in application programs or embed them in Web browsers.

For C and C++, the **Code Generation** command creates code for the prediction function and header files. For Java, it creates only the code file (no header file is needed).

See *Darwin Reference*, chapter 11, for more information about this command.

The **Code Generation** dialog requires the following input:

- **Using:**
 - **Tree/Net Model:** Select the name of the tree or net model.
 - **Subtree:** For tree models, specify the number of the selected subtree.
 - **Language:** C++ (default), C, or Java.
 - **Function:** The name of the production function that is created.

- **Created Files:**
 - **Named:** Enter the name of the file to be created.
 - **Local:** Check for local files (files on your PC).
 - The last field displays a default directory path into which files with `.cpp` and `.hpp` extensions will be saved. If you wish to save them in a different directory, click **Change** to bring up a dialog that provides other options.

By default, these files are saved to your project directory on UNIX. For a C program, Darwin creates (in your project directory on UNIX) a file `name.c` for the program and `name.h` for the header file.

11 Window Menu

The **Window** menu provides support for handling Darwin's open windows and icons. Open windows are listed and numbered, and the active window is indicated by a check mark. If the window you want is buried, you can bring it to the front by clicking its name on this menu.



You can choose to display open windows in either cascade or tile format:

- **Cascade** (the default) arranges windows on top of one another, with each subsequent window being placed slightly below and to the right of the preceding one so that the title of each shows.
- **Tile** displays open windows so that all can be seen simultaneously.
- **Arrange Icons:** This command has to do with the placement of icons for minimized Darwin windows. If you have moved these icons around in the document area, you can realign them to the bottom of the document area by clicking **Arrange Icons**.

12 Help Menu

Click **Help -> Contents** or
Click the **Help** icon (above) or
Press **F1** for context-sensitive help

Darwin online help can be called up in several ways:

- Select the **Help** pull-down menu, which contains the following:
 - **Contents:** Displays a list of help topics.
 - **Context Sensitive:** Brings up help for the dialog window that is active.
 - **Intro to Darwin:** Brings up an introduction to Darwin and data mining.
 - **About Darwin:** Tells which release of the Darwin client and server you are running and the number of CPUs you are running on.
- Click the **Help** button on dialog boxes and other windows; this brings up help for that dialog or window.
- With the cursor on a dialog window, press the **F1** key to bring up help related to that dialog window.
- Click the **Help** icon  on the toolbar; this brings up context-sensitive help.
- In the **Workspace**, click the **Help** tab at the bottom, and double-click a topic name, i.e., an item preceded by the  icon.

Each Help window has a toolbar with the following icons, from left to right:

- **Back** (goes to the previous window).
- **Next** (goes to the next window).
- **Zoom** (left-click to zoom in, enlarging the type; after four clicks, Zoom resets).

- **Print** (prints out the help file).
- **TOC** (on some windows, there is a fifth icon, which goes to the Table of Contents).



13 UNIX Utilities

This chapter describes two UNIX operations:

- `darwinDG`, which automatically creates Darwin descriptor files
- the TMC utilities `sas2darwin` and `darwin2sas`, which use the third-party product DBMS/COPY to convert between Darwin and SAS file formats

13.1 Automatically Creating a Descriptor File

This section describes how to use the UNIX command `darwinDG` to automatically create Darwin descriptor files from data (text) files. You can then create a Darwin dataset from the text file and descriptor using the **Import Text File** tab of the **Create Dataset** command.

13.1.1 Darwin Descriptor Files

Dataset descriptor files contain information that allows Darwin to read and manipulate data records:

- the number of records in the dataset
- the number of fields in each record
- the name of each field (the identifier that you give the field for use within Darwin)
- the form of each field (categorical or ordered)
- the data type of each field

You create a Darwin dataset from a text file and its corresponding descriptor file.

For detailed information about descriptor files, see *Darwin Reference*, section 4.5.

13.1.2 How to Create Descriptor Files

There are two ways to create a descriptor file from a given text file:

- Create it “by hand” using a text editor or the **New File** command of the **Project** menu, following the format described in *Darwin Reference*, section 4.5.
- Use the UNIX command `darwinDG`, as described below.

13.1.3 Using `darwinDG`

The following command gives you a summary of how to use `darwinDG`:

```
darwinDG
```

Follow these steps to create a descriptor file using `darwinDG`:

1. Log in to the Darwin server as an ordinary user.
2. Copy the text file to the project directory where you intend to create the Darwin dataset and change to that directory.
3. Type the command

```
darwinDG dataFileName -delimiter separator
```

where *dataFileName* is the name of the text file whose descriptor you wish to create and *separator* is the separator in quotation marks or the word `tab` if the separator is the tab character. (Other options are described below.)

`darwinDG` reads the file and determines the following information:

- the number of records in the text file
- the number of fields in the text file
- the data type of each field
- the form of each field (categorical or ordered)

Note: `darwinDG` assumes that all records contain the same number of fields. The first record in the text file should contain the correct number of fields because the first record is used to determine the number of fields per record. If you specify the field names using the `-fieldNames fieldNameFile` option, then the number of elements in *fieldNameFile* is the number of fields per record. Any additional records are ignored.

If you have not specified a name for each field using the `-fieldNames` option, `darwinDG` assigns the names `f1`, ..., `fn` to the fields, where `n` is the number of distinct fields.

Using this information, `darwinDG` creates a valid Darwin descriptor file in the directory where the text file resides. (You can specify a different directory, if you wish.)

`darwinDG` does not need to process the entire text file to generate a descriptor. A sample of several thousand records will generate a usable descriptor rapidly; it can take hours to process all of a large text file.

Important: You should check the generated descriptor file to make sure that it accurately reflects your intentions; you should especially check that the assignment of the type and form of each field (categorical or ordered) is appropriate.

13.1.4 The `darwinDG` Command

The complete syntax of the `darwinDG` is as follows:

```
darwinDG dataFileName -delimiter separator
        [-outputFileName desfileName]
        [-outputFileDir desDirName]
        [-fieldNames fieldsFileName]
        [-maxUniqueForCategorical numValues]
        [-displayStatistics]
```

where

<i>dataFileName</i>	Name or pathname of the text file for which you wish to generate a descriptor file. If you do not specify a directory, the current directory is assumed. REQUIRED.
<i>separator</i>	Separator or delimiter for the data file, enclosed in quotation marks (e.g., if space is the delimiter, " ") or the word <code>tab</code> for the tab character. REQUIRED.
<i>desFileName</i>	Name of the descriptor file. <code>darwinDG</code> appends <code>.des</code> to this name. If you do not specify a name, the name of the dataset file is used. OPTIONAL.
<i>desDirName</i>	Directory where the descriptor file will reside. If you do not specify a name, the directory where the data file resides is used. OPTIONAL.

fieldsFileName Name or pathname of the file containing the names of the fields in the dataset. If you do not specify a directory, the current directory is assumed. The file should consist of a list of the field names with one field name per line.

If you do not specify the field names, `darwinDG` assumes that the field names are `f1, ... fn`, where `n` is the total number of fields found. OPTIONAL.

numValues `darwinDG` determines the number of unique values for each field. If an integer field contains fewer than *numValues* unique values, it is considered to be categorical; otherwise, it is considered to be ordered. The default value is 15. OPTIONAL.

If you specify the `-displayStatistics` option, `darwinDG` prints the following statistics to the command line:

Field	Name of the field assigned by <code>darwinDG</code> or given by the user
Type	Type of the field (integer, float, or string)
Form	Form of the field (ordered or categorical)
Nulls	Total number of nulls found for the field in records
%Nulls	Percentage of total records that are null
UniVals	Number of unique values found for the field (the cutoff for counting unique fields is 50)
Max	Maximum integer or float in the field; for strings, the maximum string length
Min	Minimum integer or float in the field; for strings, the minimum string length
Mean	Average of integer or float non-null values for field (blank for strings)

`-displayStatistics` is OPTIONAL.

13.1.5 Examples of Using darwinDG

This section provides two examples of using darwinDG.

Example 1

The following command creates a descriptor file for the text file `cars2.txt`. The delimiter is " " (space). The field names are in the file `fields` (which resides in the same directory as `cars2.txt`). The descriptor file will be created in the same directory as `cars2.txt`; the name of the descriptor file is `cars2.des`.

```
darwinDG car2.txt -delimiter " " -fieldNames fields
```

The file `fields` contains the field names, each on a separate line; for example,

```
revs_per_mile
manual_trans
fuel_tank_capacity
passenger_capacity
length
wheelbase
width
uturn_space
rear_seat_room
luggage_capacity
weight
domestic
```

Example 2

The following command creates a descriptor file for the text files `test.txt`. The delimiter is the tab character. The maximum number of unique values is 30; if an integer field contains fewer than 30 values, it is considered categorical. Statistics are printed. The descriptor file `test.des` is created in the same directory as `test.txt`.

```
darwinDG test.txt -delimiter tab -displayStatistics
-maxUniqueForCategorical 30
```

The output of this command is as follows:

```
4 records processed
Field  Type    Form  Nulls  %Nulls  UniVals  Max  Min  Mean
f1     int     c     0      0       4       346  56  217.5
f2     astring c     1      25      3       3    2
f3     float  o     1      25      3       3.45 0  1.15
f4     float  o     0      0       4       3.2  0.5  1.775
```

13.1.6 How darwinDG Creates Descriptor Files

darwinDG iterates over each record in the data file and determines the type of the field based on the basic rules described in this section. The type of a field is one of the following:

- integer, such as “1”, “123456”, “-23”, “+1234”
- float, such as “1234.345”, “-2.5”, “1e0”, “-1.23e-3”
- string (anything that is not integer or float)

A field’s type is the type of the most general value. For example, if the first $n-1$ values are integers and the n th value is a float, the type of the field is float. Here are the rules used to select the most general value, where “+” means “combined with” and “->” means “yields”:

Integer + Float	-> Float
Integer + String	-> String
Float + String	-> String
String + Any	-> String

A field containing a date value, such as “9/7/67” or “24Jun89”, is treated as a string and is given the type `astring` with length equal to the maximum length of the field values.

A field’s form (ordered or categorical) is assigned as follows:

- Integer fields with fewer than n unique values, where n is the value specified with the option `-maxUniqueForCategorical` of darwinDG, are categorical, otherwise they are ordered. The default value for `-maxUniqueForCategorical` of darwinDG is 15.
- Float fields are ordered.
- String fields are categorical.
- A field containing all nulls is treated as a categorical integer field.

13.2 Converting between Darwin and SAS File Formats

You can use the TMC utilities `sas2darwin` and `darwin2sas` to convert between SAS files and Darwin text file/descriptor file pairs. You can also use the third-party product DBMS/COPY to convert between SAS files and text file/descriptor file pairs. The TMC utilities and DBMS/COPY run on UNIX.

This section describes how to convert SAS files to and from text files using the conversion scripts or DBMS/COPY. You can export Darwin datasets as text files and create Darwin datasets from text file/descriptor file pairs.

For information about the hardware and software requirements of the conversion utility and how to install it, see *Darwin Installation and Administration*.

13.2.1 Darwin Datasets

Darwin can import and export datasets as text files. `sas2darwin` creates a text file and descriptor from a SAS file that you can then use to create a Darwin dataset, as described in chapter 4. You convert Darwin datasets to SAS files by exporting them as text files and then using `darwin2sas` to convert the text file to SAS format. You can also perform these conversions using DBMS/COPY directly.

The format of Darwin datasets is described in *Darwin Reference*, chapter 4, “Creating and Handling Datasets.” A Darwin dataset is created from two files, the text or data file and the descriptor file. The data file contains the records, with each line containing a set of fields separated by a delimiter. The descriptor file contains metadata that defines field names, types, lengths, etc.

Within Darwin you import a text file using the **Import Text File** tab of the **Dataset** menu’s **Create** command or the **New Dataset** icon. To export a dataset, use the **Dataset** menu’s **Export** command to create a text file and a descriptor. In each case you need to specify the file names and the field delimiter.

For information about creating and exporting Darwin datasets, see chapter 7.

There are two main issues to be aware of when converting datasets:

- When converting from Darwin, the field delimiter is not part of the metadata.
- When converting to Darwin, you will have to specify for each field whether that field is categorical or ordered.

Further information on implementing custom conversion applications is given below, in section 13.2.6, “Implementing Custom Conversion Applications.”

13.2.2 Using the Conversion Tools

There are two ways to convert SAS files to and from Darwin text files:

1. Use one of the Perl scripts, `sas2darwin` or `darwin2sas`.
2. Use DBMS/COPY directly.

DBMS/COPY helps with the basic conversion, but you need to provide additional information or do additional preprocessing and postprocessing as described below. The TMC utilities automate the additional work required with DBMS/COPY.

13.2.3 Conversions Using `darwin2sas` and `sas2darwin`

Two Perl scripts are included with Darwin. These scripts automate the use of DBMS/COPY. The script `sas2darwin` is used to convert a SAS dataset to a Darwin text file/descriptor file pair; you can then use Darwin to create a dataset. The script `darwin2sas` is used to convert in the opposite direction. To use these scripts you must have the scripts and DBMS/COPY installed and available in your path.

For information about the details of the conversion, including information about the SAS formats supported, see section 13.2.5 and Table 1. If you wish to implement custom conversions, see section 13.2.6.

If you use the TMC utilities, you do not have to run DBMS/COPY — the scripts themselves use DBMS/COPY to accomplish the conversion.

Notes:

- If you execute `sas2darwin` or `darwin2sas` from the directory in which DBMS//COPY resides, you will not have to reset the `PATH` environment variable.
- Be sure you are not logged in as root when you try to execute either `sas2darwin` or `darwin2sas`.

`sas2darwin` and `darwin2sas` Options

The scripts support the following options:

- `-v` — Turns on verbose mode
- `-help` — Prints the help file
- `-sastype type` — Defines the pseudotype for the SAS file

- `-c list` — Specify the form of the fields (which fields are ordered and which are categorical) where *list* is a string of digits and the letters *c* (for categorical) and *o* for ordered.

The digits signify when the form of the field changes. For example, `-c c5o9` means “make the first 4 fields categorical, fields 5 through 8 ordered, fields 9 and all remaining fields categorical.” `-c c5o9` is the same as `-c cccccoooooccc...`

- `-s separator` — Defines the separator for the Darwin dataset.

For more information about these options, their defaults, and how to use them, type

```
sas2darwin -help
```

Note: The scripts use links or copies of the files and hence require extra storage space.

The examples in section 13.2.4 illustrate the use of the scripts.

13.2.4 Conversion Examples

Conversion Examples: SAS to Darwin

This section provides three examples of converting a SAS file to a Darwin text file and a descriptor file.

Note: The scripts assume that the text file is named *datafile-name* and the descriptor file is named *datafile-name.des*. *datafile-name* should not have an extension (such as *.txt* or *.ascii*).

Hint: If you execute `sas2darwin` from the directory where `DBMS/COPY` resides, you will not have to reset the `PATH` environment variable.

```
sas2darwin sal.ssd01 sal
```

This example converts the SAS dataset `sal.ssd01` to the files `sal` (data) and `sal.des` (descriptor). All fields are assumed to be categorical.

```
sas2darwin -c oco sal.ssd01 sal sal.des
```

This example converts the dataset `sal.ssd01` to the files `sal` (data) and `sal.des` (descriptor). The `-c` option specifies that first field is ordered, the second field is categorical, and all subsequent fields are ordered.

```
sas2darwin -c c11o16 sal.ssd01 sal sal.des
```

This example converts the dataset `sal.ssd01` to the files `sal` (data) and `sal.des` (descriptor). The first ten fields (fields 1 through 10) are categorical, fields 11 through 15 are ordered, and all subsequent fields (16 to end) are categorical.

```
sas2darwin -sastype ssdsun salfile sal
```

This time the input file is `salfile` and is pseudo type `ssdsun`. The `sastype` option allows you to use files that do not use the proper extensions.

Conversion Examples: Darwin to SAS

This section provides two examples of converting a Darwin data file/descriptor file pair to a SAS file.

Note: The scripts assume that the Darwin dataset was exported to two files named *dataset-name* containing the data and *dataset-name.des* containing the dataset descriptor. *dataset-name* should not have an extension (such as `.txt` or `.ascii`).

Hint: If you execute `darwin2sas` from the directory where `DBMS/COPY` resides, you will not have to reset the `PATH` environment variable.

```
darwin2sas -s "," sal sal.des sal.ssd01
```

This example converts the Darwin data files `sal` (data) and `sal.des` (descriptor) to the SAS file `sal.ssd01`. The field separator is defined as `","`.

```
darwin2sas -s "," sal sal.ssd01
```

This example is equivalent to the previous one.

13.2.5 Conversions Using DBMS/COPY

`DBMS/COPY` is a product of Conceptual Software, Inc.; it is a conversion utility that translates data between software packages. Its purpose is to insulate you from the specifics of each data format. On UNIX there are two versions: the X Windows version `dbmscopy` and the command-line version `dbmsnox`. Note that the X Windows version can also be used from the command line.

This section provides basic information on `DBMS/COPY` and specific information regarding conversion of Darwin datasets. For detailed information about `DBMS/COPY`, see *DBMS/COPY for UNIX*, written by Conceptual Software, Inc.

In subsequent sections we concentrate on the conversion to and from a SAS dataset to be used with SAS 6.11 (or higher) on UNIX.

Conversions Between Darwin Datasets and SAS Files

Conversion from SAS to Darwin. DBMS/COPY can produce a text file containing the delimited records in the correct format for Darwin. You will have to build the descriptor yourself. (For information about descriptor files, see *Darwin Reference*, section 4.5, “Dataset Descriptors.”) The field names can be obtained either from the first line of the produced dataset or from a separate dictionary file created by DBMS/COPY. The dictionary file has an extension of `.dct` and also contains information about field type and length.

When building the Darwin descriptor file, you will also have to indicate for each field whether it is categorical or ordered.

Once you have the text file and the descriptor, use the **Dataset** menu’s **Create** command or the **New Dataset** icon to create a Darwin dataset.

Conversion from Darwin to SAS. To convert a Darwin dataset to a SAS file, first save the dataset as text using the **Dataset** menu’s **Export** command. DBMS/COPY can read the resulting Darwin text file. You will have to enter the metadata to create the SAS file.

SAS Dataset Formats

There are many different data formats used by SAS. DBMS/COPY handles these by a combination of file extensions and pseudo extensions. Table 1 shows which extension relates to which SAS data format.

DBMS/COPY requires file extensions as shown in Table 1 and in some cases you need to know the SAS type.

Interactive Conversion with DBMS/COPY

You can use DBMS/COPY either interactively or as a command. This section describes using DBMS/COPY interactively.

DBMS/COPY provides a graphical user interface to facilitate conversion of datasets.

Table 1. DBMS/COPY SAS File Types and Extensions.

SAS File Descriptor	File Extension	SAS Type
SAS for PC/DOS 6.04	.ssd	ssd
SAS for UNIX 6.09 and 6.11	.ssd01	ssdsun
SAS for UNIX DEC/Alpha 6.09	.ssd01	ssddec
SAS for UNIX DEC/Alpha 6.11	.ssd04	ssd04dec
SAS for Windows 6.08 and 6.10	.sd2	sd2
SAS Transport	.v5x	sasport5
SAS Transport V6 Compressed	.ssd02	sascomp
SAS Transport V6	.ssd02	sasport
SAS Xport Transport Engine	.v5x	sasxport

First start DBMS/COPY with `dbmscopy`.

Then proceed as follows:

- To begin a conversion, select the **Interactives -> Copy Database** menu option.

When performing a conversion, you will be asked to provide the input file name, the output file name, and the file type. For a SAS dataset, choose the file type (for example, SAS for UNIX 6.09 and 6.11) from the list in Table 1. Make sure that the file extension is `ssd01`. For the text file, choose a file type of ASCII (`*.dat`), and make sure that the extension is `.dat`.

The next step depends on the direction of the conversion (SAS to Darwin or Darwin to SAS) that you want to make. Steps specific to each conversion direction are explained below.

Copying from SAS to Darwin. After initiating a copy and defining the input data file, a dialog window appears that gives you the option of viewing the dataset or transforming the data.

- Click **OK** to continue.

Another dialog window appears, prompting you to define the output file:

- Define the output file as ASCII format with file extension `.dat`.
- Click on **OK**.

A dialog window then appears and prompts you to specify the field separator, end-of-line characters, etc.:

- Specify the comma (“,”) to separate fields; specify Line Feed as end-of-line character (these are the defaults).

You can also specify whether the first line should contain the field names. Note that the names are also stored in a dictionary file with a `.dct` extension.

- Click on **OK**.

A Transfer Verification Window then appears; it lists the equivalent batch commands (with option to save).

- Click on **Do-It!**

A Processing window then appears, and updates as the transform is performed.

- When 100% of records are processed, click on **Done**.

At this stage you will have a data file containing the records and a dictionary file containing metadata. You will have to convert the dictionary file to a Darwin descriptor file and then create a Darwin dataset using the **Dataset** menu's **Create** command or the **New Dataset** icon.

Converting from Darwin to SAS. The procedure is similar to the conversion from SAS to Darwin but this time you need to define the metadata (field name and type). Put the field names on line 1 of the file; this simplifies the process of building the dictionary (metadata).

Export the Darwin dataset, creating a text file and a descriptor file.

Start DBMS/COPY with `dbmscopy`.

The conversion process is as follows:

- From the menu, select **Interactives -> Copy Database**.
- Select the file with extension `.dat`.
- Select file type of ASCII (`*.dat`).
- Click on **OK**.

Then an “ASCII Input Format Options” window appears. In it, you set null values, date formats, delimiters, and specify whether field names appear in row 1. Do not select the “fixed format box.”

- Click on **OK**.

You are then presented with an “ASCII Dictionary Builder” tool with which you define the metadata. If you placed the field names on row 1 of the file, they will already be defined. DBMS/COPY asks for field lengths but seems to ignore them. You must define the types correctly. Note that DBMS uses `char` for fields of various lengths; Darwin uses `char` for fields of length 1 and `string` for fields of length greater than 1.

- Click on **OK**.

The power panel pops up, which allows data viewing, etc.

- Click on **OK**.

The output dataset window appears.

- Select “SAS for UNIX 6.09 and 6.11 (*.ssd01)” and set the filename.
- Click on **OK**.

The Transfer Verification window appears.

- Click on **Do-It!**

A processing window then appears, and updates as the transform is performed.

- When 100% of records are processed, click on **Done**.

At this point you have generated the converted SAS dataset.

If the field names are not supplied on line 1, the same steps apply but the dictionary builder requires that you enter the field names.

Conversions Using the DBMS/COPY Command Line Interface

You can also convert between Darwin and SAS formats using a command-line interface.

Copying from SAS to Darwin Command Line Interface. You can use either the `dbmscopy` or the `dbmsnox` version of DBMS/COPY to copy SAS files to Darwin data file/descriptor file pairs. For example, either of the following commands:

```
dbmscopy comp.ssdsun comp.ascii2
dbmsnox comp.ssdsun comp.ascii2
```

copies `comp.ssd01` to `comp.dat`. Note that you have to use the pseudo extensions. You still have to postprocess `comp.dat`.

Copying from Darwin to SAS Command Line Interface. Copying from Darwin to SAS is tricky since you must somehow define the metadata (the field name and type information). It is possible to convert with a command-line interface if you have a valid dictionary file from a previous conversion with the graphical user interface. If you have a valid dictionary file, then either of the following commands will accomplish the reverse transformation:

```
dbmscopy comp.ascii2 comp.ssdsun
dbmsnox comp.ascii2 comp.ssdsun
```

The safest approach is to place the field names on line 1 of the input file and use the dictionary builder within `dbmscopy` to define the metadata.

13.2.6 Implementing Custom Conversion Applications

This section provides information for those who wish to implement conversion applications between some external application and Darwin.

The formats of Darwin datasets, data files and descriptor files are described in *Darwin Reference*, chapter 4. The information below supplements that discussion.

Descriptor File Format

The following points relate to the format of the descriptor file:

- Darwin descriptors do not have to contain the line specifying the number of fields as this can be inferred from the number of field declaration lines.
- A “normalization” line can appear in the descriptor between the lines that define the number of records and fields.

Data Types

In addition to the types listed in *Darwin Reference*, chapter 4, the following types can be output by Darwin: `_string`, `_astring`, and `_estring`. The underscore prefix indicates that Darwin may have transformed the data — for example, by trimming strings to remove leading and trailing white space. Also note that `astring` can appear in a descriptor file.

Field Names

When converting from Darwin to another application, you may have to map field names to take account of any restrictions in the other application. For example, some applications will not allow a plus sign (“+”) in a field name.

A Glossary

Note: Italicized terms appear as entries elsewhere in this glossary.

- Analysis** menu Provides commands that evaluate predictions, summarize information about datasets, calculate frequencies, calculate performance, and calculate lift.
- attribute A *field* in a Darwin dataset.
- artificial neural network
See *neural network*.
- backpropagation The most popular form of *neural network*. Backpropagation networks are feed-forward multilayer networks that use a supervised training algorithm (backward propagation of errors) to adjust the connection weights.
- byte Eight bits.
- categorical (field) A *field* is categorical if it take on a finite number of unordered values; compare with *ordered (field)*.
- classification The act of predicting that a *record* belongs to a particular group. For example, in direct marketing, predicting that the record of one customer belongs to the group of records of people who would buy gourmet coffee. Classification is one of the two methods by which data mining makes predictions; the other is *forecasting*. The difference between the two is that classification predicts membership in sets, whereas forecasting predicts values within a series.
- Classification and Regression Trees (C&RT)
A computer software technique that finds rules for making predictions by repeatedly breaking up historical examples of data into ever-smaller subgroups.

client	The part of Darwin where the graphical user interface runs; compare with <i>server</i> . The Darwin client runs on a personal computer.
clustering	The process of grouping similar input patterns together using an unsupervised training algorithm.
code generation	Generating C, C++, or Java code for tree and net models so that the models can be included in application programs or embedded in Web applets. Not all versions of Darwin permit code generation. Use the Code Generation command of the Options menu to generate code.
command log	See <i>macro</i> .
continuous (field)	Same as <i>ordered</i> . A <i>field</i> is continuous if it takes continuous values; compare with <i>categorical (field)</i> .
darwin2sas	A UNIX command that converts Darwin datasets to SAS files.
darwinDG	A UNIX command that automatically generates a descriptor file from a data (text) file.
Darwin software	A set of integrated tools that support data mining by creating <i>neural networks</i> , <i>match models</i> , and <i>tree models</i> using all the information contained in very large databases.
database	A self-describing collection of integrated <i>records</i> .
data cleansing	A processing step during which missing or inaccurate data is replaced with valid values.
data mining	The process of applying intelligent algorithms, such as <i>artificial neural networks</i> or <i>C&RT</i> , to large collections of historical data to find patterns, predict trends such as customer behavior, and achieve accurate results that are not available using traditional methods. (See <i>traditional database processing</i> .)
dataset	In Darwin, a collection of <i>records</i> sharing a common format, with a <i>dataset descriptor</i> as their header. All data is stored in datasets. Datasets are created as objects in virtual memory and may be saved as files.
dataset descriptor	A file containing information that allows Darwin tools to read and manipulate the <i>records</i> in a Darwin <i>dataset</i> .

Dataset menu	Provides commands for creating, exporting, and transforming Darwin <i>datasets</i> .
data warehouse	A large database where an organization's historical data is kept for long-term online storage.
dependent (variable)	In general, a variable whose value is determined by one or more <i>independent variables</i> ; the field whose value is to be predicted in a Darwin model. Also known as <i>target field</i> or response variable.
descriptor file	The file associated with a Darwin dataset that specifies the name and data type for each <i>field</i> in the dataset. Creating a Darwin dataset requires a data (text) file and a descriptor.
distributed dataset	A dataset that is distributed across the processors of a multiprocessor machine.
distributed file	A file that consists of several components that reside on different processors in a multiprocessor system or a file that resides in a shared file space.
Edit menu	Provides commands for various text editing functions (undo, cut, copy, and paste).
Euclidean distance	The usual way of measuring distance between points in space based on the Pythagorean Theorem.
Evaluation Wizard	The Darwin <i>wizard</i> that guides you through the process of analyzing a model.
feed-forward	In a neural net, using the input values to calculate the output values, without using backpropagation.
field	The components of a <i>record</i> . Each field contains one or more items of data and becomes a variable for data analysis. A field is either <i>ordered</i> or <i>categorical</i> .
field form	In a dataset <i>descriptor</i> , a property of a field: either <i>categorical</i> or <i>ordered</i> .
forecasting	The method of predicting values via regression, and usually referred to as <i>regression</i> . Forecasting is one of the two methods by which data mining makes predictions; the other is <i>classification</i> . The distinction between the two is that forecasting predicts values within a series, whereas classification predicts membership in sets.

fuzzy logic	A method of reasoning that allows for partial or “fuzzy” descriptions of rules. For example, the truth of a proposition such as “Company X is a medium-sized company” might vary over a range of from “completely false” to “completely true.”
genetic algorithms	Techniques that use the principles of genetics and evolution to make increasingly accurate predictions about a given database.
gigabytes of data	2 ³⁰ (approximately one billion) bytes (a byte is a unit of information consisting of 8 bits).
Help menu	Provides access to the Darwin online help and to information about the Darwin version.
independent (variable)	In general, one or more variables that determine the value of the <i>dependent variable</i> . In a Darwin model, one of the fields used to predict the <i>target field</i> .
input (layer)	The layer of a neural network that consists of the fields used to calculate the output.
input (for a model)	The datasets used to train the model.
importance	The relative contribution of an input attribute to the prediction of a particular model. The importance/sensitivity analysis performed by Darwin analyses a predictor function and a dataset producing a table that contains the input variables and their relative ranking. See also <i>sensitivity</i> .
<i>k</i> -nearest-neighbor algorithm	An algorithm that predicts values for a field using the values of the <i>k</i> records in the model nearest to the prediction record; nearness is measured using Euclidean distance.
knowledge discovery in databases (KDD)	The extraction of previously unknown information from a database; <i>data mining</i> is one phase of the KDD process.
level of confidence	Degree of certainty of result.
macro	A command log containing the commands issued during a Darwin session. You can execute the macro to reissue all the commands. You turn macros on and off using the Macro command of the Options menu or clicking the Macro icon.

massively parallel processing (MPP)	A technology for doing parallel processing. MPPs can harness hundreds of processors to work together on a problem because of the way they are hooked together. This allows them to do <i>scalable computing</i> .
match model	A model created by Darwin using a <i>k-nearest neighbors algorithm</i> .
Match Model mode	The component of Darwin that creates models using a parallel weight-adjustable <i>k-nearest-neighbor algorithm</i> ; to create a match model, click the Select Match Model mode icon or click Match on the Model menu.
MBR	<i>Memory-based reasoning</i> .
megabyte	2 ²⁰ (approximately one million) bytes (a byte is a unit of information consisting of 8 bits).
memory-based reasoning	A technique for classifying records in a database by comparing them with similar records that are already classified.
menus	The lists of commands in the Darwin user interface.
Model menu	Permits you to select the type of model to create (tree, neural net, or match); provides commands to create, copy, test, and predict with Darwin models.
Modeling Wizard	The Darwin <i>wizard</i> that guides you through the process of creating a Darwin model.
multiclass	Refers to items, such as classification or prediction, that are associated with more than one class.
net model	Same as <i>neural net model</i> .
neural networks	Techniques that make predictions by analyzing the relationships among data elements in historical data. The name is derived from the fact that artificial neural networks are similar in structure to biological neural systems. Darwin uses this technique to create <i>neural net</i> models. (Also known as <i>artificial neural networks</i> .)
neural net model	A model created by Darwin using the techniques of neural networks. Also known as net model.

Net Model mode	The component of Darwin that creates models using a parallel implementation of feed-forward <i>neural networks</i> ; to create a neural net model, click the Select Net Model mode icon or click Net in the Models menu.
Open Database Connectivity (ODBC)	The relational database that Darwin can connect to and use to create datasets, described in <i>Microsoft ODBC 2.0 Programmer's Reference and SDK Guide</i> (Redmond, Washington: Microsoft Press, 1994).
Options menu	Permits you to set and change advanced options, turn on <i>macros</i> , and invoke <i>code generation</i> .
ordered (field)	A <i>field</i> is ordered if it takes ordered values; compare with <i>categorical (field)</i> . Ordered fields are sometimes referred to as "continuous fields."
output	The output of a model is the value or values being predicted (along with an indication of the confidence in the prediction).
parallel processing	A technology that allows a more efficient processing of information, just as mass production allowed a more efficient processing of manufactured goods. Rather than funnelling information through a single processor, like conventional "vector" computers, parallel computers send tasks through multiple processors simultaneously. Darwin supports <i>symmetric multiprocessors</i> , or SMPs.
perturbation	In the context of neural networks, refers to changing the weights (values).
platform	Computer hardware on which software runs; Darwin runs on Sun Microsystems and Hewlett Packard platforms.
prediction	In the data mining context, <i>prediction</i> refers to the use of information gained from some number of known values to estimate further values.
project	In Darwin, a collection of related datasets, models, and tables. All work in Darwin takes place in the context of a project. A project is either a directory or a directory that is also linked to a distributed directory.
Project menu	Contains the commands that manage projects, display reports and graphs, control database connectivity, and exit Darwin.

query (a database)	The act of retrieving information from a database, either as a list of <i>records</i> or as a summary of the information in the records.
record	An element of a database that groups together a set of named values called <i>fields</i> . For example, in a credit card database, each cardholder would have one record. The fields would probably include name, address, income, and amount of last payment.
regression	The method of predicting values via regression (sometimes referred to as <i>forecasting</i>). Regression is one of the two methods by which data mining makes predictions; the other is <i>classification</i> . The difference between the two is that regression predicts values within a series, whereas classification predicts membership in sets.
Relational Database Management System (RDBMS)	A type of database or database management system that stores information in tables and conducts searches by using data in specified columns of one table to find additional data in another table.
<code>sas2darwin</code>	A UNIX command that converts a SAS file to a Darwin text file and descriptor.
scalable computing	The property of a parallel computer that enables the user to build a more powerful machine by adding more processors and storage capacity. Scalable machine designs allow you to expand to work with problems that are hundreds of times larger while running exactly the same software.
sensitivity	The contribution to a model from each attribute. The importance/sensitivity analysis performed by Darwin analyses a predictor function and a dataset, producing a table that contains the input variables and their relative ranking. See also <i>importance</i> .
serial dataset	A dataset that is not distributed, that is, a dataset that resides on one machine or one system of a multiprocessor system.
serial file	A file that is not distributed, that is, a file that resides on one machine or one system of a multiprocessor system.
server	(1) The part of Darwin where data mining algorithms run; compare with <i>client</i> . Darwin servers run on UNIX. (2) One or

	more collections of executables, a daemon, and a configuration file to which users connect when starting Darwin. (3) The physical machine on which Darwin runs.
SQL	The standard language used to create, manipulate (including <i>query</i>), and control relational databases.
supervised learning	A training paradigm in which the neural network is presented with an input pattern and a desired output pattern. The desired output is compared with the neural network output, and the error information is used to adjust the connection weights; compare with <i>unsupervised learning</i> .
symmetric multiprocessing (SMP)	SMP is a technology for doing parallel processing.
target (field)	The <i>dependent variable</i> for the model; the field that the model predicts.
targeted marketing	The marketing of products to select groups of consumers that are more likely than average to be interested in the offer.
terabyte	2 ⁴⁰ (approximately one trillion) bytes or one million megabytes (a byte is a unit of information consisting of 8 bits).
traditional database processing	A statistical approach to analyzing databases that requires an analyst's intuition to devise hypotheses for predicting customer behavior, and then uses small samples of historical data to test the accuracy of the hypotheses. (See <i>data mining</i> .)
transformation	One of the operations performed on a Darwin dataset using the Transform operator to create a new dataset from an old one.
transformation dataset	A Darwin dataset that is created from an existing dataset using one of the <i>Transform commands</i> .
Transform commands	The Darwin commands used to create new datasets from an existing one by performing various operations such as normalization, projection, randomization, etc.
tree model	A Darwin model created using the <i>C&RT</i> algorithm.

Tree model mode	The component of Darwin that creates models using a parallel implementation of the <i>C&RT</i> decision tree algorithm; to create a tree mode, click the Set Tree Model mode icon or click Tree on the Model menu.
unsupervised learning	A training paradigm in which the neural network is presented with input data, and it self-organizes to cluster or segment the data by learning to recognize statistical similarities; compare with <i>supervised learning</i> .
user interface	The screens, commands, menus, buttons, dialogues, and functions through which a user communicates with a piece of software.
View menu	Provides commands to invoke the two wizards, view <i>workflow</i> , customize the Darwin workspace, and refresh the display.
Window menu	Provides commands to control the display of the Darwin window (cascade, tile, and arrange icons).
wizard	A program that helps you perform a task such as installing software or creating models. Darwin includes two wizards, the <i>Modeling Wizard</i> , which guides you through the creation of a Darwin model, and the <i>Evaluation Wizard</i> , which performs the evaluation of a Darwin model. You invoke the Darwin wizards from the View menu or by clicking the wizard's icon.
workflow	A graphical display of the work done in the current Darwin session. To display the workflow, click the Workflow icon. Once you've displayed the workflow, use the Workflow menu or the icons to manage the display (zoom in, zoom out, and change the view).
Workflow menu	Provides commands for examining the workflow (zoom in, zoom out) and specifying what is displayed. This menu appears only when Workflow is active.

Index

See also appendix A, “Glossary,” for an alphabetized list of terms and their definitions.

A

activation function, with net models, 78–80

Advanced Options

for building net models, 58, 78

for datasets, 76

for Frequency Count, 70, 81

for Lift, 73, 82

for match models, 81

for Performance, 72, 81

for projects, 76

for setup, 82

for training net models, 58, 60, 79–81

for tree models, 55–57, 77

menu, 75

algorithm, training (net models), 80

Analysis menu, 15, 67

analyzing results, 6, 67

animation, of Darwin icon, 17

Append command (Transform menu), 45

arrange icons option (Window menu), 85

B

backpropagation training algorithm, (nets), 80

before using Darwin, 1

bias, match models, 81

building a model, 6

busy server, 17

C

cascade option (Window menu), 85

characters, valid, in names, 20

client and server, 2

Close command (Project menu), 23

Code Generation command (Options menu),
84

output, 20

command log. *See* Macro command

command output, 19

comparator

in Replace command, 49

in Select command, 50

confidence, in Lift, 73

configuring a server, 2

conjugate gradient training algorithm, 80

connecting

to Darwin, 11

to databases, 28

context-sensitive help, 87

continue training, net models, 58, 61

control characters, 20

controlling tree size, max nodes, 77

converting SAS file formats, 95

Copy command (Edit menu), 31

Copy command (Model menu), 66

Copy to Darwin Server (New File command),
24

cost

file, tree models, 77

function, with net models, 80

pruning function, 77

to modify priors, 78

Create command (Dataset menu), 41–43

Create Dataset

from database, 42

from text file, 41

creating a dataset, 6, 41

creating a project, 5, 21

creating models, 6, 53

match models, 63

nets, 57

overview, 7–9

trees, 54

creating transformation datasets, 44

cross-entropy (cost function), 80

cross-validation, with net models, 58, 59, 79

CTRL-1 (Evaluate analysis), 68

CTRL-2 (Summarize analysis), 69

CTRL-3 (Frequency Count), 70
 CTRL-4 (Performance analysis), 71
 CTRL-5 (Lift analysis), 72
 CTRL-A (Advanced Options), 75
 CTRL-D (Create Dataset command), 41
 CTRL-E (Test model command), 54, 59, 64
 CTRL-F (Workflow command), 38
 CTRL-G (Graph command), 26
 CTRL-L (Create model command), 54, 57, 63
 CTRL-M (Macro command), 83
 CTRL-N (New Project command), 21
 CTRL-O (Open Project command), 22
 CTRL-P (Print command), 27
 CTRL-R (Predict with model command), 56, 60, 65
 CTRL-S (Save command), 23
 CTRL-T (Transform command), 44
 CTRL-U (Evaluation Wizard), 36
 CTRL-W (Modeling Wizard), 33
 Customize command (View menu), 39
 customizing Darwin, 18
 Cut command (Edit menu), 31

D

Darwin
 about (Help file), 87
 intro to (Help file), 87
 Darwin client and server, 2
 Darwin file, creating, 24
 Darwin servers, 2
 busy, 17
 logging in to, 12
 Darwin session
 ending, 13
 starting, 11
 Darwin window, 13
 darwin2sas, 96
 examples, 98
 darwinconfig command, 2
 darwindg command, 90
 data mining process
 diagrams of, 7–10
 overview of, 5
 data type, in Missing (Transform) command, 47
 data types
 in descriptor files, 89, 90
 in SAS conversion, 103
 Database Connect command (Project menu), 28
 Database Disconnect command (Project menu), 29
 databases, creating dataset from, 42
 Dataset menu, 15, 41

datasets
 descriptor file, template for, 24
 in custom conversion, 103
 transformation, 44
 decrease function, 77
 deleting a model, 66
 deleting a project, 23
 density threshold, 77
 descriptor files
 creating automatically, 89
 in custom conversion, 103
 dialog windows, 19
 Display command (Project menu), 25
 document area, 13, 14
 documentation, ix

E

Edit menu, 15, 31
 ending a Darwin session, 13, 30
 entropy, decrease function, 77
 Evaluate Prediction command (Analysis menu), 68
 Evaluation Wizard, 36
 Exit command (Project menu), 13, 30
 exiting the login window, 13
 Explode command (Transform menu), 45
 Export command (Datasets menu), 43
 Export Table command (Project menu), 25
 exporting models, 84

F

F1 (context-sensitive help), ix, 87
 F2 (Display command), 25
 F5 (Refresh command), 39
 F8 (Code Generation command), 84
 files, creating, 24
 form, changing, 50
 Frequency Count command (Analysis menu), 70

G

Generate Model Code command. *See* Code Generation command
 genetic training algorithm, (nets), 80
 gini
 decrease function, 77
 pruning function, 77
 Graph command (Project menu), 26
 graphical output, 20, 67
 growing a tree, 53

H

Help menu, 15, 87

I

icons

- arrange option (Window menu), 85
- list of, 16–18

Import Text File (creating dataset), 41

Intro to Darwin (Help file), 87

iterations, 80

L

layers, in net models, 78

learning mode, with net models, 79

learning rate, with net models, 80

Lift command (Analysis menu), 72

logging in to Darwin, 11

Login dialog window, 11

M

Macro command (Options menu), 83

margin, in Lift, 73

match models

- creating and training, 63
- creating and training models, 81
- overview (diagram), 9
- predicting with, 65
- testing, 64

max nodes, in tree models, 77

MDI windows, 19

memory objects, datasets or models as, 19

menu bar, 13, 14

Merge command (Transform menu), 45

Missing command (Transform menu), 46

missing values, 46

mode (learning), with net models, 79

Model menu, 15, 53

Modeling Wizard, 33

models

- creating, 6, 53, 57, 63
- match, 63
- net, 57
- tree, 53

modified newton training algorithm, (nets), 80

N

naming objects, 20

neighbors

- information about, 64
- number of, 81

net models

- continue training, 61
- creating, 57

layers, 78–80

overview (diagram), 8

perturbation, 62

predicting with, 60

testing, 59

neural networks. *See* net models

New Dataset icon, 41

New File command (Project menu), 24

New Project command, 21

Normalize command (Transform menu), 47

numbers, displayed to four decimal digits, 20

O

online documentation, ix

online help, ix, 87

Open Project command, 22

opening a dataset, 6

opening a project, 5

optimization

- for match models, 64, 65, 81
- for net models, 78
- for trees, 77

Options menu, 15, 75

output datasets, of prediction commands. *See*

output prediction datasets

output of commands, 19

output prediction dataset, 56, 60, 65

merged with input dataset, 67

overview of Darwin data mining process, 5

P

Paste command (Edit menu), 31

Performance Matrix command (Analysis menu), 71

perturbation, 59, 61, 62, 80

p-norm, 80

prediction

- using match models, 65
- with net models, 60
- with tree models, 56

prediction datasets, 19

merged with output, 45

output of Predict command, 56, 60, 65

use in analysis, 67

Print command (Project menu), 27

Print Preview command (Project menu), 28

priors file, in tree models, 77–79

Project command (Transform menu), 47

Project menu, 14, 21

pruning functions, with tree models, 77

R

Randomize command (Transform menu), 48

Range command (Transform menu), 48
 Refresh command (View menu), 39
 Replace command (Transform menu), 49
 Reprune Tree command, 55, 56
 restore (net weights), 80
 Re-Train, net models, 80
 ROI, in Lift, 73
 Rules, in tree models, 55

S

Sample command (Transform menu), 50
 SAS conversion utility
 Perl scripts, 96
 using, 95, 96
 SAS dataset formats, 99
 sas2darwin, 96
 examples, 97
 Save command (Project menu), 23
 saving objects, before exiting Darwin, 13
 Select command (Transform menu), 50
 Sensitivity
 with Evaluation Wizard, 37
 with match models, 64
 with net models, 59
 with tree models, 55
 servers, definitions of, 3
 Set Form command (Transform menu), 50
 Setup tab (Advanced Options), 82
 simple training, with net models, 79
 size of tree, controlling, 77
 Split command (Transform menu), 51
 square, cost function (nets), 80
 starting a Darwin session, 11
 status bar, 13, 14
 steepest descent training algorithm, (nets), 80
 Stop command (Project menu), 29
 Summarize Data command (Analysis menu),
 69

T

testing models
 match models, 64

 nets, 59
 trees, 54
 text files, creating datasets from, 41
 tile option (Window menu), 85
 title bar, 13, 14
 toolbar, 13, 14, 16
 train and test, with net models, 8, 59, 79–81
 train-test ratio (net models), 80
 Transform commands (Dataset menu), 44
 transformation datasets, creating, 44
 tree models
 controlling size of, 77
 creating and training, 54
 displaying rules, 55
 overview (diagram), 7
 predicting with, 56
 testing, 54

U

units per layer, net models, 78
 UNIX operations, 89

V

valid characters in names, 20
 View menu, 15, 33

W

weights
 creating a file, 24
 with match models, 64, 65, 81
 with net models, 62, 79, 80
 window, main Darwin, 13
 Window menu, 15, 85
 windows, dialog and MDI, 19
 wizards
 evaluation, 36
 modeling, 33
 Workflow comand (View menu), 38
 Workspace, 13, 14, 18