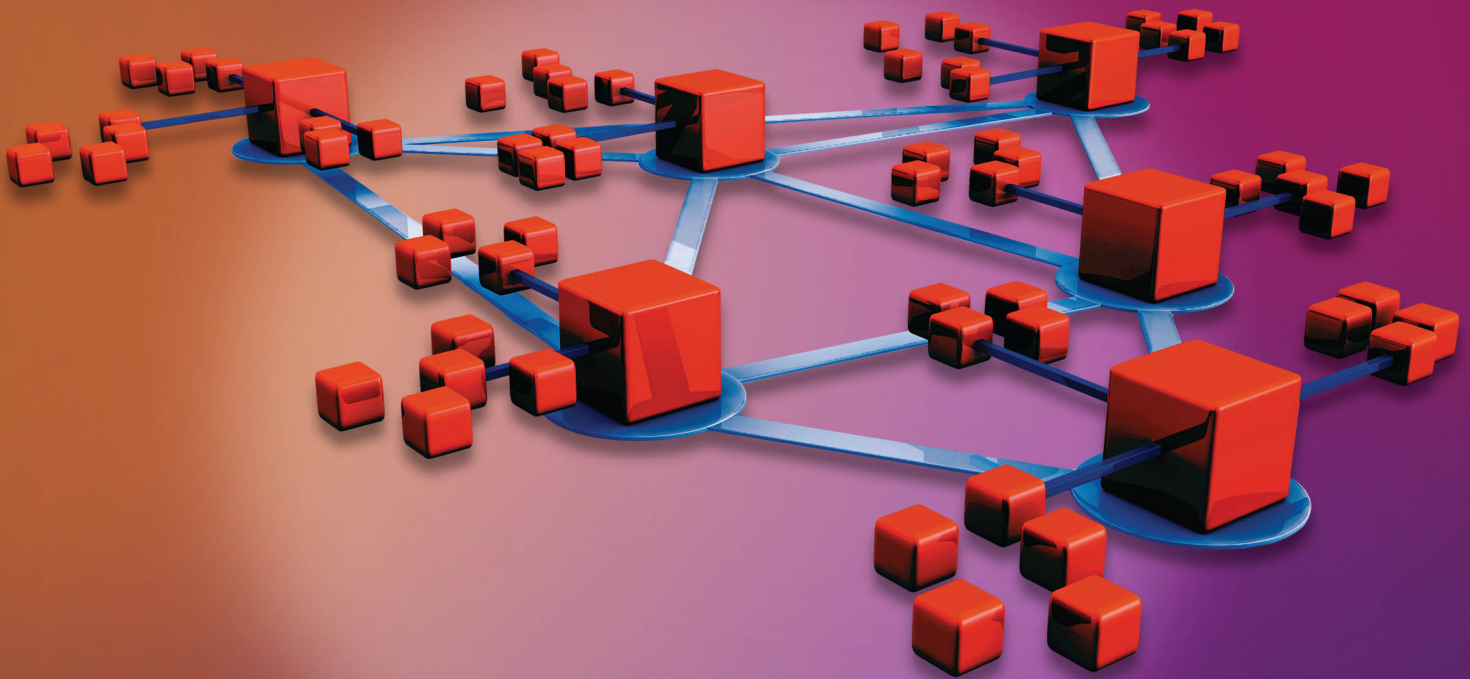


Predictive Modeling with SAS[®] Enterprise Miner[™]

Practical Solutions for Business Applications

Third Edition



Kattamuri S. Sarma, PhD

The correct bibliographic citation for this manual is as follows: Sarma, Kattamuri S., Ph.D. 2017. *Predictive Modeling with SAS® Enterprise Miner™: Practical Solutions for Business Applications, Third Edition*. Cary, NC: SAS Institute Inc.

Predictive Modeling with SAS® Enterprise Miner™: Practical Solutions for Business Applications, Third Edition

Copyright © 2017, SAS Institute Inc., Cary, NC, USA

ISBN 978-1-62960-264-6 (Hard copy)

ISBN 978-1-63526-038-0 (EPUB)

ISBN 978-1-63526-039-7 (MOBI)

ISBN 978-1-63526-040-3 (PDF)

All Rights Reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

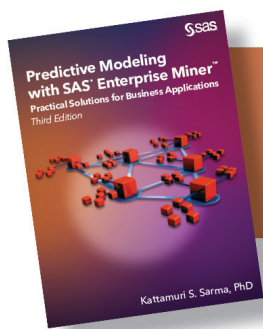
SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

July 2017

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to <http://support.sas.com/thirdpartylicenses>.



An Introduction to Predictive Modeling with SAS® Enterprise Miner™: Practical Solutions for Business Applications, Third Edition. Full book available for purchase [here](#).

Contents

About This Book	xi
About The Author	xiii
Chapter 1: Research Strategy.....	1
1.1 Introduction	1
1.2 Types of Inputs.....	2
1.2.1 Measurement Scales for Variables.....	2
1.2.2 Predictive Models with Textual Data.....	2
1.3 Defining the Target	2
1.3.1 Predicting Response to Direct Mail.....	2
1.3.2 Predicting Risk in the Auto Insurance Industry.....	4
1.3.3 Predicting Rate Sensitivity of Bank Deposit Products	5
1.3.4 Predicting Customer Attrition	7
1.3.5 Predicting a Nominal Categorical (Unordered Polychotomous) Target.....	8
1.4 Sources of Modeling Data.....	10
1.4.1 Comparability between the Sample and Target Universe.....	10
1.4.2 Observation Weights.....	10
1.5 Pre-Processing the Data	10
1.5.1 Data Cleaning Before Launching SAS Enterprise Miner	11
1.5.2 Data Cleaning After Launching SAS Enterprise Miner	11
1.6 Alternative Modeling Strategies	12
1.6.1 Regression with a Moderate Number of Input Variables	12
1.6.2 Regression with a Large Number of Input Variables.....	13
1.7 Notes	13
Chapter 2: Getting Started with Predictive Modeling.....	15
2.1 Introduction	16
2.2 Opening SAS Enterprise Miner 14.1	16
2.3 Creating a New Project in SAS Enterprise Miner 14.1	16
2.4 The SAS Enterprise Miner Window	17
2.5 Creating a SAS Data Source.....	18
2.6 Creating a Process Flow Diagram	27
2.7 Sample Nodes	27
2.7.1 Input Data Node.....	27
2.7.2 Data Partition Node	29
2.7.3 Filter Node.....	29
2.7.4 File Import Node	33

2.7.5 Time Series Nodes	36
2.7.6 Merge Node.....	50
2.7.7 Append Node	53
2.8 Tools for Initial Data Exploration.....	56
2.8.1 Stat Explore Node.....	57
2.8.2 MultiPlot Node	64
2.8.3 Graph Explore Node.....	67
2.8.4 Variable Clustering Node.....	73
2.8.5 Cluster Node	82
2.8.6 Variable Selection Node.....	85
2.9 Tools for Data Modification	94
2.9.1 Drop Node	94
2.9.2 Replacement Node.....	95
2.9.3 Impute Node.....	98
2.9.4 Interactive Binning Node	99
2.9.5 Principal Components Node	106
2.9.6 Transform Variables Node.....	112
2.10 Utility Nodes	120
2.10.1 SAS Code Node	120
2.11 Appendix to Chapter 2.....	126
2.11.1 The Type, the Measurement Scale, and the Number of Levels of a Variable	126
2.11.2 Eigenvalues, Eigenvectors, and Principal Components.....	129
2.11.3 Cramer's V.....	132
2.11.4 Calculation of Chi-Square Statistic and Cramer's V for a Continuous Input.....	133
2.12 Exercises.....	135
Notes	137
Chapter 3: Variable Selection and Transformation of Variables	139
3.1 Introduction	139
3.2 Variable Selection	140
3.2.1 Continuous Target with Numeric Interval-scaled Inputs (Case 1)	140
3.2.2 Continuous Target with Nominal-Categorical Inputs (Case 2).....	147
3.2.3 Binary Target with Numeric Interval-scaled Inputs (Case 3)	153
3.2.4 Binary Target with Nominal-scaled Categorical Inputs (Case 4)	158
3.3 Variable Selection Using the Variable Clustering Node.....	162
3.3.1 Selection of the Best Variable from Each Cluster.....	164
3.3.2 Selecting the Cluster Components.....	174
3.4 Variable Selection Using the Decision Tree Node.....	176
3.5 Transformation of Variables	179
3.5.1 Transform Variables Node.....	179
3.5.2 Transformation before Variable Selection	181
3.5.3 Transformation after Variable Selection	183
3.5.4 Passing More Than One Type of Transformation for Each Interval Input to the Next Node.....	185

3.5.5 Saving and Exporting the Code Generated by the Transform Variables Node.....	189
3.6 Summary	190
3.7 Appendix to Chapter 3.....	190
3.7.1 Changing the Measurement Scale of a Variable in a Data Source	190
3.7.2 SAS Code for Comparing Grouped Categorical Variables with the Ungrouped Variables.....	192
Exercises.....	192
Note	193
Chapter 4: Building Decision Tree Models to Predict Response and Risk....	195
4.1 Introduction	196
4.2 An Overview of the Tree Methodology in SAS® Enterprise Miner™	196
4.2.1 Decision Trees	196
4.2.2 Decision Tree Models	196
4.2.3 Decision Tree Models vs. Logistic Regression Models	198
4.2.4 Applying the Decision Tree Model to Prospect Data	198
4.2.5 Calculation of the Worth of a Tree.....	199
4.2.6 Roles of the Training and Validation Data in the Development of a Decision Tree....	201
4.2.7 Regression Tree	202
4.3 Development of the Tree in SAS Enterprise Miner	202
4.3.1 Growing an Initial Tree.....	202
4.3.2 P-value Adjustment Options	209
4.3.3 Controlling Tree Growth: Stopping Rules.....	211
4.3.3.1 Controlling Tree Growth through the Split Size Property	211
4.3.4 Pruning: Selecting the Right-Sized Tree Using Validation Data.....	211
4.3.5 Step-by-Step Illustration of Growing and Pruning a Tree.....	213
4.3.6 Average Profit vs. Total Profit for Comparing Trees of Different Sizes.....	218
4.3.7 Accuracy /Misclassification Criterion in Selecting the Right-sized Tree (Classification of Records and Nodes by Maximizing Accuracy)	218
4.3.8 Assessment of a Tree or Sub-tree Using Average Square Error	220
4.3.9 Selection of the Right-sized Tree	220
4.4 Decision Tree Model to Predict Response to Direct Marketing	221
4.4.1 Testing Model Performance with a Test Data Set	230
4.4.2 Applying the Decision Tree Model to Score a Data Set	231
4.5 Developing a Regression Tree Model to Predict Risk	236
4.5.1 Summary of the Regression Tree Model to Predict Risk.....	243
4.6 Developing Decision Trees Interactively	244
4.6.1 Interactively Modifying an Existing Decision Tree.....	244
4.6.3 Developing the Maximal Tree in Interactive Mode	266
4.7 Summary	269
4.8 Appendix to Chapter 4.....	270
4.8.1 Pearson's Chi-Square Test.....	270
4.8.2 Calculation of Impurity Reduction using Gini Index	271
4.8.3 Calculation of Impurity Reduction/Information Gain using Entropy	272
4.8.4 Adjusting the Predicted Probabilities for Over-sampling	274

4.8.5 Expected Profits Using Unadjusted Probabilities	275
4.8.6 Expected Profits Using Adjusted Probabilities	275
4.9 Exercises.....	275
Notes	277
Chapter 5: Neural Network Models to Predict Response and Risk.....	279
5.1 Introduction	280
5.1.1 Target Variables for the Models.....	280
5.1.2 Neural Network Node Details.....	281
5.2 General Example of a Neural Network Model.....	281
5.2.1 Input Layer	282
5.2.2 Hidden Layers	283
5.2.3 Output Layer or Target Layer.....	288
5.2.4 Activation Function of the Output Layer	289
5.3 Estimation of Weights in a Neural Network Model.....	290
5.4 Neural Network Model to Predict Response	291
5.4.1 Setting the Neural Network Node Properties.....	293
5.4.2 Assessing the Predictive Performance of the Estimated Model.....	297
5.4.3 Receiver Operating Characteristic (ROC) Charts	300
5.4.4 How Did the Neural Network Node Pick the Optimum Weights for This Model?.....	303
5.4.5 Scoring a Data Set Using the Neural Network Model	305
5.4.6 Score Code	308
5.5 Neural Network Model to Predict Loss Frequency in Auto Insurance	308
5.5.1 Loss Frequency as an Ordinal Target	309
5.5.1.1 Target Layer Combination and Activation Functions	311
5.5.3 Classification of Risks for Rate Setting in Auto Insurance with Predicted Probabilities	321
5.6 Alternative Specifications of the Neural Networks	322
5.6.1 A Multilayer Perceptron (MLP) Neural Network.....	322
5.6.2 Radial Basis Function (RBF) Neural Network.....	324
5.7 Comparison of Alternative Built-in Architectures of the Neural Network Node	330
5.7.1 Multilayer Perceptron (MLP) Network.....	332
5.7.2 Ordinary Radial Basis Function with Equal Heights and Widths (ORBFEQ)	333
5.7.3 Ordinary Radial Basis Function with Equal Heights and Unequal Widths (ORBFUN).....	335
5.7.4 Normalized Radial Basis Function with Equal Widths and Heights (NRBFEQ).....	338
5.7.5 Normalized Radial Basis Function with Equal Heights and Unequal Widths (NRBFEH).....	340
5.7.6 Normalized Radial Basis Function with Equal Widths and Unequal Heights (NRBFEW).....	343
5.7.7 Normalized Radial Basis Function with Equal Volumes (NRBFEV)	346
5.7.8 Normalized Radial Basis Function with Unequal Widths and Heights (NRBFUN).....	348
5.7.9 User-Specified Architectures.....	351
5.8 AutoNeural Node.....	354
5.9 DMNeural Node.....	356
5.10 Dmine Regression Node	358

5.11 Comparing the Models Generated by DMNeural, AutoNeural, and Dmine Regression Nodes	360
5.12 Summary	362
5.13 Appendix to Chapter 5	363
5.14 Exercises	365
Notes	367
Chapter 6: Regression Models	369
6.1 Introduction	369
6.2 What Types of Models Can Be Developed Using the Regression Node?	369
6.2.1 Models with a Binary Target	369
6.2.2 Models with an Ordinal Target	373
6.2.3 Models with a Nominal (Unordered) Target	379
6.2.4 Models with Continuous Targets	383
6.3 An Overview of Some Properties of the Regression Node	383
6.3.1 Regression Type Property	384
6.3.2 Link Function Property	384
6.3.3 Selection Model Property	386
6.3.4 Selection Criterion Property	403
6.4 Business Applications	415
6.4.1 Logistic Regression for Predicting Response to a Mail Campaign	417
6.4.2 Regression for a Continuous Target	431
6.5 Summary	442
6.6 Appendix to Chapter 6	443
6.6.1 SAS Code	443
6.6.2 Examples of the selection criteria when the Model Selection property set to Forward.	447
6.7 Exercises	451
Notes	452
Chapter 7: Comparison and Combination of Different Models	453
7.1 Introduction	453
7.2 Models for Binary Targets: An Example of Predicting Attrition	454
7.2.1 Logistic Regression for Predicting Attrition	456
7.2.2 Decision Tree Model for Predicting Attrition	458
7.2.3 A Neural Network Model for Predicting Attrition	460
7.3 Models for Ordinal Targets: An Example of Predicting the Risk of Accident Risk	464
7.3.1 Lift Charts and Capture Rates for Models with Ordinal Targets	465
7.3.2 Logistic Regression with Proportional Odds for Predicting Risk in Auto Insurance	466
7.3.3 Decision Tree Model for Predicting Risk in Auto Insurance	469
7.3.4 Neural Network Model for Predicting Risk in Auto Insurance	473
7.4 Comparison of All Three Accident Risk Models	476
7.5 Boosting and Combining Predictive Models	476
7.5.1 Gradient Boosting	477
7.5.2 Stochastic Gradient Boosting	479
7.5.3 An Illustration of Boosting Using the Gradient Boosting Node	479

7.5.4 The Ensemble Node	482
7.5.5 Comparing the Gradient Boosting and Ensemble Methods of Combining Models ...	485
7.6 Appendix to Chapter 7	486
7.6.1 Least Squares Loss	486
7.6.2 Least Absolute Deviation Loss.....	486
7.6.3 Huber-M Loss	487
7.6.4 Logit Loss	487
7.7 Exercises.....	488
Note	488
Chapter 8: Customer Profitability	489
8.1 Introduction	489
8.2 Acquisition Cost	491
8.3 Cost of Default	492
8.5 Profit	493
8.6 The Optimum Cutoff Point	495
8.7 Alternative Scenarios of Response and Risk	496
8.8 Customer Lifetime Value	496
8.9 Suggestions for Extending Results	497
Note	497
Chapter 9: Introduction to Predictive Modeling with Textual Data	499
9.1 Introduction	499
9.1.1 Quantifying Textual Data: A Simplified Example.....	500
9.1.2 Dimension Reduction and Latent Semantic Indexing	503
9.1.3 Summary of the Steps in Quantifying Textual Information	506
9.2 Retrieving Documents from the World Wide Web	507
9.2.1 The %TMFILTER Macro.....	507
9.3 Creating a SAS Data Set from Text Files	509
9.4 The Text Import Node.....	512
9.5 Creating a Data Source for Text Mining	514
9.6 Text Parsing Node.....	516
9.7 Text Filter Node	521
9.7.1 Frequency Weighting	521
9.7.2 Term Weighting	521
9.7.3 Adjusted Frequencies	521
9.7.4 Frequency Weighting Methods	521
9.7.5 Term Weighting Methods	523
9.8 Text Topic Node	528
9.8.1 Developing a Predictive Equation Using the Output Data Set Created by the Text Topic Node	533
9.9 Text Cluster Node	534
9.9.1 Hierarchical Clustering	535
9.9.2 Expectation-Maximization (EM) Clustering	536
9.9.3 Using the Text Cluster Node	542

9.10 Exercises.....	546
Notes	546
Index	547

About This Book

What Does This Book Cover?

The book shows how to rapidly develop and test predictive models using SAS® Enterprise Miner™. Topics include Logistic Regression, Regression, Decision Trees, Neural Networks, Variable Clustering, Observation-Clustering, Data Imputation, Binning, Data Exploration, Variable Selection, Variable Transformation, Modeling Binary and continuous targets, Analysis of textual data, Eigenvalues, Eigenvectors and principal components, Gradient Boosting, Ensemble, Time Series Data Preparation, Time Series Dimension Reduction, Time Series Similarity and importing external data into SAS Enterprise Miner. The book demonstrates various methods using simple examples and shows how to apply them to real-world business data using SAS Enterprise Miner. It integrates theoretical explanations with the computations done by various SAS nodes. The examples include manual computations with simple examples as well computations done using SAS code with real data sets from different businesses.

Support Vector Machines and Association rules are not covered in this book.

Is This Book for You?

If you are a business analyst, a student trying to learn predictive modeling using SAS Enterprise Miner, a data scientist who wants process data efficiently and build predictive models, this book is for you. If you want to learn how to select key variables, test a variety of models quickly and develop robust predictive models in a short period of time using SAS Enterprise Miner, this book gives you step-by-step guidance with simple explanation of the procedures and the underlying theory.

What Are the Prerequisites for This Book?

- Elementary algebra and basic training (equivalent to one to two semesters of course work) in statistics covering inference, hypothesis testing, probability and regression
- Experience with Base SAS® software and some understanding of simple SAS macros and macro variables.

What's New in This Edition?

The book is updated to the latest version of SAS Enterprise Miner. The time series section is enhanced. Time Series Exponential Smoothing, Time Series Correlation, Time Series Dimension Reduction and Time Series Similarity nodes are added. Examples of calculating the information gain of node splits using Gini index and Entropy measures are included. More examples are added to describe the process of model selection in the regression node.

What Should You Know about the Examples?

Realistic business examples are used. You need SAS Enterprise Miner so that you can read the book and try the examples simultaneously.

Software Used to Develop the Book's Content

SAS Enterprise Miner

Example Code and Data

You can access the example code and data for this book by linking to its author page at <https://support.sas.com/authors>.

Output and Graphics

Almost all the graphics are generated by SAS Enterprise Miner. A few graphs are generated by SAS/GRAPH® Software.

Where Are the Exercise Solutions?

Exercise solutions are posted on the author page at <https://support.sas.com/authors>.

We Want to Hear from You

SAS Press books are written *by* SAS Users *for* SAS Users. We welcome your participation in their development and your feedback on SAS Press books that you are using. Please visit <https://support.sas.com/publishing> to do the following:

- Sign up to review a book
- Recommend a topic
- Request authoring information
- Provide feedback on a book

Do you have questions about a SAS Press book that you are reading? Contact the author through saspress@sas.com or https://support.sas.com/author_feedback.

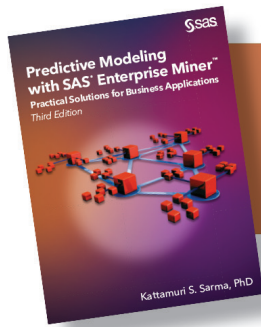
SAS has many resources to help you find answers and expand your knowledge. If you need additional help, see our list of resources: <https://support.sas.com/publishing>.

About The Author



Kattamuri S. Sarma, PhD, is an economist and statistician with 30 years of experience in American business, including stints with IBM and AT&T. He is the founder and president of Ecostat Research Corp., a consulting firm specializing in predictive modeling and forecasting. Over the years, Dr. Sarma has developed predictive models for the banking, insurance, telecommunication, and technology industries. He has been a SAS user since 1992, and he has extensive experience with multivariate statistical methods, econometrics, decision trees, and data mining with neural networks. The author of numerous professional papers and publications, Dr. Sarma is a SAS Certified Professional and a SAS Alliance Partner. He received his bachelor's degree in mathematics and his master's degree in economic statistics from universities in India. Dr. Sarma received his PhD in economics from the University of Pennsylvania, where he worked under the supervision of Nobel Laureate Lawrence R. Klein.

Learn more about this author by visiting his author page at support.sas.com/sarma. There you can download free book excerpts, access example code and data, read the latest reviews, get updates, and more.



An Introduction to Predictive Modeling with SAS® Enterprise Miner™: Practical Solutions for Business Applications, Third Edition. Full book available for purchase [here](#).

Chapter 1: Research Strategy

1.1 Introduction.....	1
1.2 Types of Inputs.....	2
1.2.1 Measurement Scales for Variables	2
1.2.2 Predictive Models with Textual Data	2
1.3 Defining the Target	2
1.3.1 Predicting Response to Direct Mail	2
1.3.2 Predicting Risk in the Auto Insurance Industry	4
1.3.3 Predicting Rate Sensitivity of Bank Deposit Products	5
1.3.4 Predicting Customer Attrition	7
1.3.5 Predicting a Nominal Categorical (Unordered Polychotomous) Target	8
1.4 Sources of Modeling Data	10
1.4.1 Comparability between the Sample and Target Universe	10
1.4.2 Observation Weights	10
1.5 Pre-Processing the Data	10
1.5.1 Data Cleaning Before Launching SAS Enterprise Miner	11
1.5.2 Data Cleaning After Launching SAS Enterprise Miner	11
1.6 Alternative Modeling Strategies	12
1.6.1 Regression with a Moderate Number of Input Variables.....	12
1.6.2 Regression with a Large Number of Input Variables.....	13
1.7 Notes.....	13

1.1 Introduction

This chapter discusses the planning and organization of a predictive modeling project. Planning involves tasks such as these:

- defining and measuring the target variable in accordance with the business question
- collecting the data
- comparing the distributions of key variables between the modeling data set and the target population to verify that the sample adequately represents the target population
- defining sampling weights if necessary
- performing data-cleaning tasks that need to be done prior to launching SAS® Enterprise Miner™

Alternative strategies for developing predictive models using SAS Enterprise Miner are discussed at the end of this chapter.

1.2 Types of Inputs

In predictive models one can use different types of data. The common data types are *numeric* and *character*. The *measurement scales*, which are referred to as *levels* in SAS® Enterprise Miner™, of the variables of a predictive modeling data set are defined in Section 1.2.1. The use of textual inputs in predictive modeling is discussed in Section 1.2.2.

1.2.1 Measurement Scales for Variables

I will first define the measurement scales for variables that are used in this book. In general, I have tried to follow the definitions given by Alan Agresti:

- A *categorical variable* is one for which the measurement scale consists of a set of categories.
- Categorical variables for which levels (categories) do not have a natural ordering are called *nominal*.
- Categorical variables that do have a natural ordering of their levels are called *ordinal*.
- An *interval variable* is one that has numerical distances between any two levels of the scale.¹
- A *binary variable* is one that takes only two values such as “1” and “0”, “M” and “F”, etc.

According to the above definitions, the variables INCOME and AGE in Tables 1.1 to 1.5 and BAL_AFTER in Table 1.3 are interval-scaled variables. Because the variable RESP in Table 1.1 is categorical and has only two levels, it is called a binary variable. The variable LOSSFRQ in Table 1.2 is ordinal. (In SAS Enterprise Miner, you can change its measurement scale to interval, but I have left it as ordinal.) The variables PRIORPR and NEXTPR in Table 1.5 are nominal.

Interval-scaled variables are sometimes called *continuous*. Continuous variables are treated as interval variables. Therefore, I use the terms *interval-scaled* and *continuous* interchangeably.

I also use the terms ordered polychotomous variables and ordinal variables interchangeably. Similarly, I use the terms unordered polychotomous variables and nominal variables interchangeably.

1.2.2 Predictive Models with Textual Data

Textual data can be used for developing predictive models. To develop predictive models from Textual data, one has to first convert the textual data into a numeric form. The Textual data is first arranged into tabular form, where each row of the table contains one full document. Some examples of textual data and methods of converting textual data into numeric form are discussed in Chapter 9.

1.3 Defining the Target

The first step in any data mining project is to define and measure the target variable to be predicted by the model that emerges from your analysis of the data. This section presents examples of this step applied to five different business questions.

1.3.1 Predicting Response to Direct Mail

In this example, a hypothetical auto insurance company wants to acquire customers through direct mail. The company wants to minimize mailing costs by targeting only the most responsive customers. Therefore, the company decides to use a response model. The target variable for this model is RESP, and it is binary, taking the value of 1 for response and 0 for no response.

Table 1.1 shows a simplified version of a data set used for modeling the binary target response (RESP).

Table 1.1

CUSTOMER	AGE	INCOME	STATUS	PC	NC	RESP
1	25	\$45,000	S	1	1	0
2	45	\$61,000	MC	1	2	1
3	54	.	MC	1	3	0
4	32	\$24,000	MNC	0	4	0
5	43	\$31,000	MC	0	5	0
6	56	\$23,456	MC	1	6	1
7	78	.	W	0	7	0
8	6	\$100,256	D	1	1	0
9	26	\$345,678	MNC	1	2	1
10	32	\$100,211	S	0	3	0
11	51	\$21,312	MC	1	4	0
12	31	\$83,456		0	5	1
13	23	\$24,234	MNC	1	1	0
14	47	\$43,566	MC	0	3	0
15	77	\$12,002	MC	1	4	1
16	83	\$32,454	W	1	5	0
17	25	\$61,345	S	0	6	0
18	32	\$76,123	MC	1	7	0
19	52	\$25,324		1	8	0
20	32	\$31,886	MNC	0	1	0
21	23	\$78,345	S	1	8	0
22	80	\$61,234	MNC	1	2	0
23	123	\$76,876	S	1	4	0
24	45	\$24,002		3	5	0

In Table 1.1 the variables AGE, INCOME, STATUS, PC, and NC are input variables (or explanatory variables). AGE and INCOME are numeric and, although they could theoretically be considered continuous, it is simply more practical to treat them as interval-scaled variables.

The variable STATUS is categorical and nominal-scaled. The categories of this variable are S if the customer is single and never married, MC if married with children, MNC if married without children, W if widowed, and D if divorced.

The variable PC is numeric and binary. It indicates whether the customers own a personal computer, taking the value 1 if they do and 0 if not. The variable NC represents the number of credit cards the customers own. You can decide whether this variable is ordinal or interval scaled.

The target variable is RESP and takes the value 1 if the customer responded, for example, to a mailing campaign, and 0 otherwise. A binary target can be either numeric or character; I could have recorded a response as Y instead of 1, and a non-response as N instead of 0, with virtually no implications for the form of the final equation.

Note that there are some extreme values in the table. For example, one customer's age is recorded as 6. This is obviously a recording error, and the age should be corrected to show the actual value, if possible. Income has missing values that are shown as dots, while the nominal variable STATUS has missing values that are represented by blanks. The **Impute** node of SAS Enterprise Miner can be used to impute such missing values. See Chapters 2, 6, and 7 for details.

1.3.2 Predicting Risk in the Auto Insurance Industry

The auto insurance company wants to examine its customer data and classify its customers into different risk groups. The objective is to align the premiums it is charging with the risk rates of its customers. If high-risk customers are charged low premiums, the loss ratios will be too high and the company will be driven out of business. If low-risk customers are charged disproportionately high rates, then the company will lose customers to its competitors. By accurately assessing the risk profiles of its customers, the company hopes to set customers' insurance premiums at an optimum level consistent with risk. A risk model is needed to assign a risk score to each existing customer.

In a risk model, *loss frequency* can be used as the target variable. Loss frequency is calculated as the number of losses due to accidents per *car-year*, where car-year is equal to the time since the auto insurance policy went into effect, expressed in years, multiplied by the number of cars covered by the policy. Loss frequency can be treated as either a continuous (interval-scaled) variable or a discrete (ordinal) variable that classifies each customer's losses into a limited number of bins. (See Chapters 5 and 7 for details about bins.) For purposes of illustration, I model loss frequency as a continuous variable in Chapter 4 and as a discrete ordinal variable in Chapters 5 and 7. The loss frequency considered here is the loss arising from an accident in which the customer was "at fault," so it could also be referred to as "at-fault accident frequency". I use *loss frequency*, *claim frequency*, and *accident frequency* interchangeably.

Table 1.2 shows what the modeling data set might look like for developing a model with loss frequency as an ordinal target.

Table 1.2

CUSTOMER	AGE	INCOME	NPRVIO	LOSSFRQ
1	25	\$45,000	0	0
2	45	\$61,000	1	1
3	54	.	2	0
4	32	\$24,000	3	3
5	43	\$31,000	4	0
6	56	\$23,456	0	0
7	78	.	1	2
8	6	\$100,256	3	2
9	26	\$345,678	4	1
10	32	\$100,211	5	3
11	51	\$21,312	3	2
12	31	.	1	1
13	23	\$24,234	0	0
14	47	\$43,566	1	0
15	77	\$12,002	0	0
16	83	\$32,454	0	2
17	25	\$61,345	1	1
18	32	\$76,123	1	0
19	52	\$25,324	3	1
20	32	\$31,886	1	0
21	23	\$78,345	3	3
22	80	\$61,234	2	0
23	123	\$76,876	2	0
24	45	\$24,002	1	1

The target variable is LOSSFRQ, which represents the accidents per car-year incurred by a customer over a period of time. This variable is discussed in more detail in subsequent chapters in this book. For now it is sufficient to note that it is an ordinal variable that takes on values of 0, 1, 2, and 3. The input variables are AGE, INCOME, and NPRVIO. The variable NPRVIO represents the number of previous violations a customer had before he purchased the insurance policy.

1.3.3 Predicting Rate Sensitivity of Bank Deposit Products

In order to assess customers' sensitivity to an increase in the interest rate on a savings account, a bank may conduct price tests. Suppose one such test involves offering a higher rate for a fixed period of time, called the *promotion window*.

In order to assess customer sensitivity to a rate increase, it is possible to fit three types of models to the data generated by the experiment:

- a response model to predict the probability of response
- a short-term demand model to predict the expected change in deposits during the promotion period
- a long-term demand model to predict the increase in the level of deposits beyond the promotion period

The target variable for the response model is binary: response or no response. The target variable for the short-term demand model is the increase in savings deposits during the promotion period net² of any concomitant declines in other accounts. The target variable for the long-term demand model is the amount of the increase remaining in customers' bank accounts after the promotion period. In the case of this model, the promotion window for analysis has to be clearly defined, and only customer transactions that have occurred prior to the promotion window should be included as inputs in the modeling sample.

Table 1.3 shows what the data set looks like for modeling a continuous target.

Table 1.3

CUSTOMER	AGE	INCOME	B_JAN	B_FEB	B_MAR	B_APR	BAL_AFTER
1	25	\$45,000	\$4,000	\$4,230	\$4,400	\$4,900	\$5,900
2	45	\$61,000	\$5,000	\$4,000	\$3,000	\$0	\$2,000
3	54	.	\$1,200	\$1,100	\$3,000	\$100	\$200
4	32	\$24,000	\$5,234	\$345	\$5,678	\$78	\$878
5	43	\$31,000	\$4,000	\$4,232	\$4,100	\$4,700	\$4,950
6	56	\$23,456	\$2,000	\$4,000	\$3,000	\$20	\$1,000
7	78	.	\$1,200	\$1,100	\$3,000	\$100	\$1,300
8	6	\$100,256	\$5,234	\$345	\$5,678	\$78	\$1,088
9	26	\$345,678	\$3,435	\$4,674	\$678	\$80,000	\$80,000
10	32	\$100,211	\$787	\$4,230	\$4,400	\$4,900	\$5,900
11	51	\$21,312	\$8,750	\$7,800	\$3,456	\$50	\$10,000
12	31	.	\$5,000	\$4,000	\$3,000	\$100	\$4,000
13	23	\$24,234	\$4,000	\$4,230	\$4,400	\$4,376	\$5,900
14	47	\$43,566	\$4,674	\$678	\$800	\$7,890	\$8,890
15	77	\$12,002	\$5,234	\$345	\$5,678	\$78	\$1,078
16	83	\$32,454	\$4,000	\$4,230	\$4,400	\$4,900	\$5,900
17	25	\$61,345	\$2,000	\$4,000	\$3,000	\$120	\$1,000
18	32	\$76,123	\$1,200	\$1,100	\$3,000	\$100	\$1,100
19	52	\$25,324	\$5,234	\$345	\$5,678	\$78	\$1,078
20	32	\$31,886	\$3,435	\$4,674	\$678	\$8,000	\$9,000
21	23	\$78,345	\$787	\$4,230	\$4,400	\$4,900	\$5,900
22	80	\$61,234	\$8,780	\$7,800	\$3,456	\$0	\$100
23	123	\$76,876	\$5,000	\$4,000	\$3,000	\$250	\$1,034
24	45	\$24,002	\$4,300	\$4,200	\$4,400	\$4,900	\$7,245

The data set shown in Table 1.3 represents an attempt by a hypothetical bank to induce its customers to increase their savings deposits by increasing the interest paid to them by a predetermined number of basis points. This increased interest rate was offered (let us assume) in May 2006. Customer deposits were then recorded at the end of May 2006 and stored in the data set shown in Table 1.3 under the variable name BAL_AFTER. The bank would like to know what type of customer is likely to increase her savings balances the most in response to a future incentive of the same amount. The target variable for this is the dollar amount of change in balances from a point before the promotion period to a point after the promotion period. The target variable is continuous. The inputs, or explanatory variables, are AGE, INCOME, B_JAN, B_FEB, B_MAR, and B_APR. The variables B_JAN, B_FEB, B_MAR, and B_APR refer to customers' balances in all their accounts at the end of January, February, March, and April of 2006, respectively.

1.3.4 Predicting Customer Attrition

In banking, attrition may mean a customer closing a savings account, a checking account, or an investment account. In a model to predict attrition, the target variable can be either binary or continuous. For example, if a bank wants to identify customers who are likely to terminate their accounts at *any* time within a pre-defined interval of time in the future, it is possible to model attrition as a binary target. However, if the bank is interested in predicting the *specific* time at which the customer is likely to “attrit,” then it is better to model attrition as a continuous target—time to attrition.

In this example, attrition is modeled as a binary target. When you model attrition using a binary target, you must define a performance window during which you observe the occurrence or non-occurrence of the event. If a customer attrited during the performance window, the record shows 1 for the event and 0 otherwise.

Any customer transactions (deposits, withdrawals, and transfers of funds) that are used as inputs for developing the model should take place during the period prior to the performance window. The *inputs window* during which the transactions are observed, the *performance window* during which the event is observed, and the *operational lag*, which is the time delay in acquiring the inputs, are discussed in detail in Chapter 7 where an attrition model is developed.

Table 1.4 shows what the data set looks like for modeling customer attrition.

Table 1.4

CUSTOMER	AGE	INCOME	B_JAN	B_FEB	B_MAR	B_APR	ATTR
1	25	\$45,000	\$4,000	\$4,230	\$4,400	\$4,900	0
2	45	\$61,000	\$5,000	\$4,000	\$3,000	\$0	1
3	54	.	\$1,200	\$1,100	\$3,000	\$100	0
4	32	\$24,000	\$5,234	\$345	\$5,678	\$78	0
5	43	\$31,000	\$4,000	\$4,232	\$4,100	\$4,700	0
6	56	\$23,456	\$2,000	\$4,000	\$3,000	\$20	1
7	78	.	\$1,200	\$1,100	\$3,000	\$100	0
8	6	\$100,256	\$5,234	\$345	\$5,678	\$78	0
9	26	\$345,678	\$3,435	\$4,674	\$678	\$80,000	1
10	32	\$100,211	\$787	\$4,230	\$4,400	\$4,900	0
11	51	\$21,312	\$8,750	\$7,800	\$3,456	\$50	1
12	31	.	\$5,000	\$4,000	\$3,000	\$100	1
13	23	\$24,234	\$4,000	\$4,230	\$4,400	\$4,376	0
14	47	\$43,566	\$4,674	\$678	\$800	\$7,890	0
15	77	\$12,002	\$5,234	\$345	\$5,678	\$78	1
16	83	\$32,454	\$4,000	\$4,230	\$4,400	\$4,900	0
17	25	\$61,345	\$2,000	\$4,000	\$3,000	\$120	0
18	32	\$76,123	\$1,200	\$1,100	\$3,000	\$100	0
19	52	\$25,324	\$5,234	\$345	\$5,678	\$78	0
20	32	\$31,886	\$3,435	\$4,674	\$678	\$8,000	0
21	23	\$78,345	\$787	\$4,230	\$4,400	\$4,900	0
22	80	\$61,234	\$8,780	\$7,800	\$3,456	\$0	0
23	123	\$76,876	\$5,000	\$4,000	\$3,000	\$250	0
24	45	\$24,002	\$4,300	\$4,200	\$4,400	\$4,900	0

In the data set shown in Table 1.4, the variable ATTR represents the customer attrition observed during the performance window, consisting of the months of June, July, and August of 2006. The target variable takes the value of 1 if a customer attrits during the performance window and 0 otherwise. Table 1.4 shows the input variables for the model. They are AGE, INCOME, B_JAN, B_FEB, B_MAR, and B_APR. The variables B_JAN, B_FEB, B_MAR, and B_APR refer to customers' balances for all of their accounts at the end of January, February, March, and April of 2006, respectively.

1.3.5 Predicting a Nominal Categorical (Unordered Polychotomous) Target

Assume that a hypothetical bank wants to predict, based on the products a customer currently owns and other characteristics, which product the customer is likely to purchase next. For example, a customer may currently have a savings account and a checking account, and the bank would like to know if the customer is likely to open an investment account or an IRA, or take out a mortgage. The target variable for this situation is nominal. Models with nominal targets are also used by market researchers who need to understand consumer preferences for different products or brands. Chapter 6 shows some examples of models with nominal targets.

Table 1.5 shows what a data set might look like for modeling a nominal categorical target.

Table 1.5

CUSTOMER	AGE	INCOME	PRIORPR	NEXTPR
1	25	\$45,000	A	X
2	45	\$61,000	B	Z
3	54		C	Y
4	32	\$24,000	A	X
5	43	\$31,000	B	Z
6	56	\$23,456	C	Z
7	78		C	Z
8	6	\$100,256	A	X
9	26	\$345,678	AB	X
10	32	\$100,211	CD	Z
11	51	\$21,312	AC	Y
12	31		AB	X
13	23	\$24,234	CD	Z
14	47	\$43,566	D	Z
15	77	\$12,002	E	Z
16	83	\$32,454	A	X
17	25	\$61,345	B	X
18	32	\$76,123	A	Z
19	52	\$25,324	A	Y
20	32	\$31,886	C	X
21	23	\$78,345	D	Z
22	80	\$61,234	A	Z
23	123	\$76,876	B	Z
24	45	\$24,002	D	X

In Table 1.5, the input data includes the variable PRIORPR, which indicates the product or products owned by the customer of a hypothetical bank at the beginning of the performance window. The *performance window*, defined in the same way as in Section 1.3.4, is the time period during which a customer's purchases are observed. Given that a customer owned certain products at the beginning of the performance window, we observe the next product that the customer purchased during the performance window and indicate it by the variable NEXTPR.

For each customer, the value for the variable PRIORPR indicates the product that was owned by the customer at the beginning of the performance window. The letter A might stand for a savings account, B might stand for a certificate of deposit, etc. Similarly, the value for the variable NEXTPR indicates the first product purchased by a customer during the performance window. For example, if the customer owned product B at the beginning of the performance window and purchased products X and Z, in that order,

during the performance window, then the variable NEXTPR takes the value X. If the customer purchased Z and X, in that order, the variable NEXTPR takes the value Z, and the variable PRIORPR takes the value B on the customer's record.

1.4 Sources of Modeling Data

There are two different scenarios by which data becomes available for modeling. For example, consider a marketing campaign. In the first scenario, the data is based on an experiment carried out by conducting a marketing campaign on a well-designed sample of customers drawn from the target population. In the second scenario, the data is a sample drawn from the results of a past marketing campaign and not from the target population. While the latter scenario is clearly less desirable, it is often necessary to make do with whatever data is available. In such cases, you can make some adjustments through observation weights to compensate for the lack of perfect compatibility between the modeling sample and the target population.

In either case, for modeling purposes, the file with the marketing campaign results is appended to data on customer characteristics and customer transactions. Although transaction data is not always available, these tend to be key drivers for predicting the attrition event.

1.4.1 Comparability between the Sample and Target Universe

Before launching a modeling project, you must verify that the sample is a good representation of the target universe. You can do this by comparing the distributions of some key variables in the sample and the target universe. For example, if the key characteristics are age and income, then you should compare the age and income distribution between the sample and the target universe.

1.4.2 Observation Weights

If the distributions of key characteristics in the sample and the target population are different, sometimes observation weights are used to correct for any bias. In order to detect the difference between the target population and the sample, you must have some prior knowledge of the target population. Assuming that age and income are the key characteristics, you can derive the weights as follows: Divide income into, let's say, four groups and age into, say, three groups. Suppose that the target universe has N_{ij} people in the i^{th} age group and j^{th} income group, and assume that the sample has n_{ij} people in the same age-income group. In addition, suppose the total number of people in the target population is N , and the total number of people in the sample is n . In this case, the appropriate observation weight is $(N_{ij} / N) / (n_{ij} / n)$ for the individual in the i^{th} age group and j^{th} income group in the sample. You should construct these observation weights and include them for each record in the modeling sample prior to launching SAS Enterprise Miner, in effect creating an additional variable in your data set. In SAS Enterprise Miner, you assign the role of **Frequency** to this variable in order for the modeling tools to consider these weights in estimating the models. This situation inevitably arises when you do not have a scientific sample drawn from the target population, which is very often the case.

However, another source of bias is often deliberately introduced. This bias is due to over-sampling of rare events. For example, in response modeling, if the response rate is very low, you must include all the responders available and only a random fraction of non-responders. The bias introduced by such over-sampling is corrected by adjusting the predicted probabilities with prior probabilities. These techniques are discussed in Section 4.8.2.

1.5 Pre-Processing the Data

Pre-processing has several purposes:

- eliminate obviously irrelevant data elements, e.g., name, social security number, street address, etc., that clearly have no effect on the target variable

- convert the data to an appropriate measurement scale, especially converting categorical (nominal scaled) data to interval scaled when appropriate
- eliminate variables with highly skewed distributions
- eliminate inputs which are really target variables disguised as inputs
- impute missing values

Although you can do many cleaning tasks within SAS Enterprise Miner, there are some that you should do prior to launching SAS Enterprise Miner.

1.5.1 Data Cleaning Before Launching SAS Enterprise Miner

Data vendors sometimes treat interval-scaled variables, such as birth date or income, as character variables. If a variable such as birth date is entered as a character variable, it is treated by SAS Enterprise Miner as a categorical variable with many categories. To avoid such a situation, it is better to derive a numeric variable from the character variable and then drop the original character variable from your data set.

Similarly, income is sometimes represented as a character variable. The character A may stand for \$20K (\$20,000), B for \$30K, etc. To convert the income variable to an ordinal or interval scale, it is best to create a new version of the income variable in which all the values are numeric, and then eliminate the character version of income.

Another situation which requires data cleaning that cannot be done within SAS Enterprise Miner arises when the target variable is disguised as an input variable. For example, a financial institution wants to model customer attrition in its brokerage accounts. The model needs to predict the probability of attrition during a time interval of three months in the future. The institution decides to develop the model based on actual attrition during a performance window of three months. The objective is to predict attritions based on customers' demographic and income profiles, and balance activity in their brokerage accounts prior to the window. The binary target variable takes the value of 1 if the customer attrits and 0 otherwise. If a customer's balance in his brokerage account is 0 for two consecutive months, then he is considered an attriter, and the target value is set to 1. If the data set includes both the target variable (attrition/no attrition) and the balances during the performance window, then the account balances may be inadvertently treated as input variables. To prevent this, inputs which are really target variables disguised as input variables should be removed before launching SAS Enterprise Miner.

1.5.2 Data Cleaning After Launching SAS Enterprise Miner

Display 1.1 shows an example of a variable that is highly skewed. The variable is MS, which indicates the marital status of a customer. The variable RESP represents customer response to mail. It takes the value of 1 if a customer responds, and 0 otherwise. In this hypothetical sample, there are only 100 customers with marital status M (married), and 2900 with S (single). None of the married customers are responders. An unusual situation such as this may cause the marital status variable to play a much more significant role in the predictive model than is really warranted, because the model tends to infer that all the married customers were non-responders because they were married. The real reason there were no responders among them is simply that there were so few married customers in the sample.

Display 1.1

The FREQ Procedure				
Frequency Percent Row Pct Col Pct	Table of MS by Resp			
	MS(Marital Status)	Resp		
		0	1	Total
M		100	0	100
		3.33	0.00	3.33
		100.00	0.00	
		3.42	0.00	
S		2826	74	2900
		94.20	2.47	96.67
		97.45	2.55	
		96.58	100.00	
Total		2926	74	3000
		97.53	2.47	100.00

These kinds of variables can produce spurious results if used in the model. You can identify these variables using the **StatExplore** node, set their roles to Rejected in the **Input Data** node, and drop them from the table using the **Drop** node.

The **Filter** node can be used for eliminating observations with extreme values, although I do not recommend elimination of observations. Correcting them or capping them instead might be better, in order to avoid introducing any bias into the model parameters. The **Impute** node offers a variety of methods for imputing missing values. These nodes are discussed in the next chapter. Imputing missing values is necessary when you use **Regression** or **Neural Network** nodes.

1.6 Alternative Modeling Strategies

The choice of modeling strategy depends on the modeling tool and the number of inputs under consideration for modeling. Here are examples of two possible strategies when using the **Regression** node.

1.6.1 Regression with a Moderate Number of Input Variables

Pre-process the data:

- Eliminate obviously irrelevant variables.
- Convert nominal-scaled inputs with too many levels to numeric interval-scaled inputs, if appropriate.
- Create composite variables (such as average balance in a savings account during the six months prior to a promotion campaign) from the original variables if necessary. This can also be done with SAS Enterprise Miner using the **SAS Code** node.

Next, use SAS Enterprise Miner to perform these tasks:

- Impute missing values.
- Transform the input variables.

- Partition the modeling data set into train, validate, and test (when the available data is large enough) samples. Partitioning can be done prior to imputation and transformation, because SAS Enterprise Miner automatically applies these to all parts of the data.
- Run the **Regression** node with the Stepwise option.

1.6.2 Regression with a Large Number of Input Variables

Pre-process the data:

- Eliminate obviously irrelevant variables.
- Convert nominal-scaled inputs with too many levels to numeric interval-scaled inputs, if appropriate.
- Combine variables if necessary.

Next, use SAS Enterprise Miner to perform these tasks:

- Impute missing values.
- Make a preliminary variable selection. (Note: This step is not included in Section 1.6.1.)
- Group categorical variables (collapse levels).
- Transform interval-scaled inputs.
- Partition the data set into train, validate, and test samples.
- Run the **Regression** node with the Stepwise option.

The steps given in Sections 1.6.1 and 1.6.2 are only two of many possibilities. For example, one can use the **Decision Tree** node to make a variable selection and create dummy variables to then use in the **Regression** node.

1.7 Notes

1. Alan Agresti, *Categorical Data Analysis* (New York, NY: John Wiley & Sons, 1990), 2.
2. If a customer increased savings deposits by \$100 but decreased checking deposits by \$20, then the net increase is \$80. Here, *net* means *excluding*.

Index

A

- accuracy criterion 218–220
- acquisition cost 491–492
- activation functions
 - about 283, 370
 - output layer 289–290
 - target layer 311–315
- Add value 351
- adjusted frequencies 521
- adjusted probabilities, expected profits using 275
- AIC (Akaike Information Criterion) 406–407
- Append node 53–56, 137
- Arc Tanget function 284, 286
- Architecture property
 - about 362
 - MLP setting 290
 - Neural Network node 323, 324, 326, 336, 341, 346, 347
 - NRBFUN network 347
 - Regression node 460
- architectures
 - alternative built-in 330–354
 - of neural networks 362
 - user-specified 351–354
- Assessment Measure property 200, 213, 218–219, 224, 458–460, 469–473
- attrition, predicting 454–464
- auto insurance industry, predicting risk in 4–5
- AutoNeural node 354–355, 360–362, 363
- Average method 483
- average profit, vs. total profit for comparing tree size 218
- average squared error 200, 220

B

- Backward Elimination method
 - about 386
 - when target is binary 386–389
 - when target is continuous 389–392
- bank deposit products, predicting rate sensitivity of 5–7
- β , as vector of coefficients 370
- bin
 - See groups
- binary split search, splitting nodes using 202–203
- binary targets
 - Backward Elimination method with 386–389
 - Forward Selection method with 393–395
 - models for 454–464

- with nominal-scaled categorical inputs 158–162
- with numeric interval-scaled inputs 153–158
- regression models with 369–373
- stepwise selection method with 398–400
- binning transformations 113
- Bonferroni Adjustment property 209–210
- Boolean retrieval method 501
- branch 196
- bucket 113
 - See also groups
- business applications
 - logistic regression for predicting mail campaign response 417–431
 - of regression models 415–442

C

- calculating
 - Chi-Square statistic for continuous input 133–135
 - cluster components 75–76
 - Cramer's V for continuous input 132–133
 - eigenvectors 130–131
 - misclassification rate/accuracy rate 218–220
 - principal components 131–132
 - residuals 478–479
 - validation profits 216–218
 - worth of a tree 199–201
 - worth of splits 203–209
- categorical variables 2, 192
- child nodes 196
- Chi-Square
 - calculating for continuous input 133–135
 - criterion for 154–158, 160–162
 - selection method 87
 - statistic 58, 59
 - test for 270–274
- Chi-Square property, StatExplore node 134
- class inputs, transformations of 116
- Class Inputs property, Transform Variables node 116, 188, 438
- class interval
 - See groups
- Class Levels Count Threshold property 21, 22, 127, 128, 190, 292, 309
- Cloglog 385
- Cluster Algorithm property 534–535
- Cluster node 56, 82–85, 533
- Cluster Variable Role property, Cluster node 83, 85
- Clustering Source property, Variable Clustering node 75

- clusters and clustering
 - assigning variables to 76
 - EM (Expectation-Maximization) 535–541, 543–544
 - hierarchical 534–535
 - selecting components 174–176
 - selecting variables for 164–174
 - Code Editor property, SAS Code node 122–124
 - combination functions 283, 311–315
 - combining
 - groups 104–106
 - models 453–487
 - predictive models 476–486
 - comparing
 - alternative built-in architectures of neural networks 330–354
 - categorical variables with ungrouped variables 192
 - gradient boosting and ensemble methods 485–486
 - models 453–487
 - models generated by DMNeural, AutoNeural, and Dmine Regression nodes 360–362
 - samples and targets 10
 - Complementary Log-Log link (Cloglog) 385
 - continuous input, calculating Chi-Square and Cramer's V for 133–135
 - continuous targets
 - Backward Elimination method with 389–392
 - with Forward Selection method 395–397
 - with nominal-categorical inputs 147–153
 - with numeric interval-scaled inputs 140–147
 - regression for 431–442
 - regression models with 383
 - stepwise selection method with 400–403
 - Correlations property, StatExplore node 63
 - Cosine function 284
 - cost of default 492–493
 - Cramer's V 60–61, 132–133
 - Cross Validation Error 411
 - Cross Validation Misclassification rate 411–412
 - Cross Validation Profit/Loss criterion 414–415
 - customer attrition, predicting 7–8
 - customer lifetime value 496
 - customer profitability
 - about 489–491
 - acquisition cost 491–492
 - alternative scenarios of response and risk 496
 - cost of default 492–493
 - customer lifetime value 496
 - extending results 497
 - optimum cut-off point 495–496
 - profit 493–495
 - revenue 493
 - Cutoff Cumulative property, Principal Components node 109
 - Cutoff Value property
 - Replacement node 96
 - Transform Variables node 116
 - cut-off values 300
- ## D
- data
 - applying decision tree models to prospect 198
 - pre-processing 10–12
 - data cleaning 11–12
 - data matrix 502–503, 505–506
 - Data Mining the Web* (Markov and Larose) 541
 - data modification, nodes for
 - See also* Transform Variables node
 - Drop 12, 94–95
 - Impute 12, 98–99, 179–180, 417, 456
 - Interactive Binning 99–106
 - Principal Components 106–112
 - Replacement 95–98
 - Data Options dialog box 67–68
 - Data Partition node
 - about 29, 30, 291, 293
 - loss frequency as an ordinal target 311
 - Partitioning Method property 29, 456
 - property settings 223
 - Regression node 417, 431
 - variable selection 170
 - variable transformation 179–180
 - Data Set Allocations property, Data Partition node 29
 - data sets
 - applying decision tree models to score 231–235
 - creating from text files 509–512
 - scoring using Neural Network models 305–308
 - scoring with models 319–321
 - Data Source property, Input Data node 27–28
 - data sources
 - changing measurement scale of variables in 190–191
 - creating 18–27, 38–41, 514–516
 - creating for text mining 514–516
 - creating for transaction data 38–41**
 - decision 197, 200
 - decision tree models
 - about 196–198
 - accuracy/misclassification criterion 218–220
 - adjusting predicted possibilities for over-sampling 274–275
 - applying to prospect data 198
 - assessing using Average Square Error 220
 - average profit vs. total profit 218
 - binary split searches 202–203
 - calculating worth of trees 199–201
 - compared with logistic regression models 198
 - controlling growth of trees 211
 - developing regression tree model to predict risk 236–244

- exercises 275–276
 - impurity reduction 209
 - measuring worth of splits 203–209
 - Pearson's Chi-square test 270–274
 - for predicting attrition 458–460
 - predicting response to direct marketing with 221–235
 - for predicting risk in auto insurance 469–473
 - pruning trees 211–213
 - p*-value adjustment options 209–211
 - regression tree 202
 - roles of training and validation data in 201–202
 - in SAS Enterprise Miner 202–221
 - selecting size of trees 220–221
 - Decision Tree node
 - See also* decision trees
 - about 139–140, 190
 - bins in 115
 - building decision tree models 221
 - Interactive property 245, 252, 266–269
 - Leaf Role property 177
 - logistic regression 439
 - in process flow 157
 - regression models 458–460, 464–475
 - Regression node 416
 - variable selection in 143
 - variable selection using 176–179
 - decision trees, growing 269
 - Decision Weights tab 24–25
 - Decisions property 213
 - Decisions tab 25
 - Default Filtering Method property, Filter node 30
 - Default Input Method property, Impute node 97
 - Default Limits Method property, Replacement node 96
 - default methods 116–118
 - degree of separation 204–206
 - depth adjustment 210–211
 - depth multiplier 210
 - Diagram Workspace 18
 - dimension reduction 503–506
 - direct mail, predicting response to 2–4
 - direct marketing, predicting response to 221–235
 - DMDB procedure 93–94
 - Dmine Regression node 358–360, 360–362, 363
 - DMNeural node 356–358, 360–362, 363
 - documents, retrieving from World Wide Web 507–508
 - document-term matrix 502–503
 - Drop from Tables property, Drop node 95
 - Drop node 12, 94–95
- E**
- EHRadial value 351
 - Eigenvalue Source property 108
 - eigenvalues 75, 129–132
 - eigenvectors 129–132
 - Elliot function 284, 287
 - EM (Expectation-Maximization) clustering 535–541, 543–544
 - Ensemble node 454, 476, 482–484, 485–486
 - Entropy 206–207
 - Entry Significance Level property, Regression node
 - Forward Selection method 393–395, 395–397
 - regression models 432, 456
 - Stepwise Selection method 400–403
 - EQRadial value 351
 - EQSlopes value 351
 - error function 290
 - EVRadial value 351
 - EWRadial value 351
 - exercises
 - decision tree models 275–276
 - models, combining 488
 - neural network models 365–367
 - predictive modeling 135–136
 - regression models 451–452
 - textual data, predictive modeling with 545
 - variable selection 192–193
 - Expectation-Maximization (EM) clustering 535–541, 543–544
 - expected losses 497
 - expected lossfrq 466
 - explanatory variables 196, 282
 - Exported Data property
 - Input Data node 121
 - Time Series node 41–45
- F**
- false positive fraction 300
 - File Import node 33–36
 - Filter node 12, 29–33, 525
 - Filter Viewer property 527
 - fine tuning 29
 - Forward Selection method
 - about 393
 - when target is binary 393–395
 - when target is continuous 395–397
 - frequency
 - about 10
 - adjusted 521
 - FREQUENCY procedure 61–62
 - frequency weighting 521–522
 - Frequency Weighting property 525
- G**
- Gauss function 284
 - Gini Cutoff property, Interactive Binning node 100
 - Gini Impurity Index 206
 - gradient boosting 477–479
 - Gradient Boosting node 476, 479–481, 485–486
 - GraphExplore node 56, 57, 67–72

groups

See also leaf nodes

combining 104–106

splitting 101–103

H

Help Panel 1

Hidden Layer Activation Function property 281,
324, 326, 332, 351

Hidden Layer Combination Function property 281,
324, 325, 332, 351–354

hidden layers 283–288

Hide property

Regression node 421

Transform Variables node 119, 185, 188

transforming variables 188

Hide Rejected Variables property, Variable Selection
node 142

hierarchical clustering 534–535

Huber-M Loss 487

Hyperbolic Tangent function 283–288

I

Identity link 385

Import File property, File Import node 34

Imported Data property, SAS Code node 121

impurity reduction

about 60

as measure of goodness of splits 206

when target is continuous 209

Impute node 12, 98–99, 179–180, 417, 456

Include Class Variables property, Variable Clustering
node 163

initial data exploration, nodes for

about 56–57

Cluster 56, 82–85, 533

Graph Explore 56, 57, 67–72

MultiPlot 56, 57, 64–67, 416

Stat Explore 12, 56, 57–64, 94–95, 134, 416

Variable Clustering 56, 73–82, 139–140, 162–
176, 190, 407

Variable Selection 56, 85–94, 179–180, 181–
185, 188, 190, 416

input 130, 196

Input Data node

about 12, 291, 292

building decision tree models 221–222, 231

Data Source property 27–28

Exported Data property 121

loss frequency as an ordinal target 309–311

in process flow 119

regression models 456, 482

scoring datasets 319

transforming variables 179–180

Input Data Source node 384, 417, 431

input layer 282

Input Standardization property 352

input variables, regression with large number of 13

inputs window 7

Interactive Binning node 99–106

Interactive Binning property, Interactive Binning
node 100–101

Interactive property, Decision Tree node 245, 252,
266–269

Interactive Selection property, Principal Components
node 110

intermediate nodes 155

Interval Criterion property 203, 209

interval inputs, transformations for 113–115

Interval Inputs property

Merge node 50–53, 52

Regression node 421

Transform Variables node 113, 116, 181, 185,
188

interval variables 2

Interval Variables property

Filter node 31–32

StatExplore node 58, 63, 134

inverse link function 370

K

KeepHierarchies property, Variable Clustering node
76

L

Larose, D.T.

Data Mining the Web 541

latent semantic indexing 503–506

leaf nodes 196, 269

See also terminal nodes

Leaf Role property 461

Decision Tree node 177

Regression node 439

Leaf Size property 211, 425

Leaf Variable property 461

Least Absolute Deviation Loss 486

Least Squares Loss 486

lift 201

lift charts 465–466

Linear Combination function 352

Linear Regression 384

Linear value 351

link function 369

Link Function property, Regression node 373, 383,
384–385, 404

Logistic function 284–285

logistic regression

about 384

for predicting attrition 456–458

for predicting mail campaign response 417–431

with proportional odds 466–469

logistic regression models, vs. decision tree models 198
 Logit link 370, 385
 Logit Loss 487
 logworth 204–205
 loss frequency 4, 236, 280, 308–321

M

marginal profit 494
 marginal revenue 494
 Markov, Z.
 Data Mining the Web 541
 maximal tree 201, 216–218
 Maximum Clusters property, Variable Clustering node 73
 Maximum Depth property 211, 425
 Maximum Eigenvalue property, Variable Clustering node 73, 75
 Maximum method 483
 Maximum Number of Steps property 400–403
 maximum posterior probability/accuracy, classifying nodes by 219
 measurement scale 2, 126–128
 measurement scale, of variables 190–191
 Menu Bar 17
 Merge node 50–53, 185–188
 Merging property, Transform Variables node 187
 Metadata Advisor Options window 21
 Method property 100, 213, 425
 methods
 Average 483
 Backward Elimination 386–392
 Boolean retrieval 501
 Chi-Square selection 87
 default 116–118
 frequency weighting 521–522
 Maximum 483
 R-Square selection 86–87
 term weighting 522–527
 Minimum Chi-Square property, Variable Selection node 87
 Minimum property, Cluster node 84
 Minimum R-Square property, Variable Selection node 88, 142–143
 misclassification criterion 200, 218–220
 MLP (Multilayer Perception) neural network 322–324, 332–333
 Model Comparison node
 assessing predictive performance of estimated models 297–300
 building decision tree models 230–231, 269
 comparing alternative built-in architectures 330
 in process flow 175
 Regression node 417, 427, 429
 variable selection 177
 Model Selection Criterion property 291, 292, 315, 460

Model Selection property 456
 modeling data, sources of 10
 modeling strategies, alternative 12–13
 models
 See also neural network models
 for binary targets 454–464
 combining 488
 comparing and combining 453–487
 for ordinal targets 464–475
 Multilayer Perception (MLP) neural network 322–324, 332–333
 Multiple Method property, Transform Variable node 188–189
 MultiPlot node 56, 57, 64–67, 416

N

Network property, Neural Network node 293, 311, 326
 neural network models
 about 280
 alternative specifications of 322–330
 AutoNeural node 360–362
 comparing alternative built-in architectures of
 Neural Network node 330–354
 Dmine Regression node 358–360, 360–362
 DMNeural node 356–358, 360–362
 estimating weights in 290–291
 exercises 365–367
 general example of 281–290
 nodes for 281
 for predicting attrition 460–464
 predicting loss frequency in auto insurance 308–321
 for predicting risk in auto insurance 473–475
 scoring data sets using 305–308
 target variables for 280
 Neural Network node
 about 281, 363
 Architecture property 323, 324, 326, 336, 341, 346, 352
 loss frequency as an ordinal target 309–311
 Model Selection Criterion property 315
 Multilayer Perceptron (MLP) neural networks 322–324
 Normalized Radial Basis Function with Equal Volumes (NRBFV) 346–348
 Normalized Radial Basis Function with Equal Widths and Heights (NRBFHQ) 338–340
 Ordinary Radial Basis Function with Equal Heights and Unequal Widths (ORBFUN) 333–335
 Radial Basis Function neural networks in 324–330
 regression models 464–475
 score ranks in Results window 315
 scoring datasets 319

- selecting optimal weights 303–305
 - setting properties of 293–297
 - target layer combination and activation functions 311–315
 - neural networks
 - about 363
 - alternative specifications of 322–330
 - comparing alternative built-in architectures in 330–354
 - node definitions 201
 - Node (Tool) group tabs 17
 - Node ID property, Transform Variables node 186, 187
 - nodes
 - See also* Data Partition node
 - See also* Decision Tree node
 - See also* Input Data node
 - See also* Model Comparison node
 - See also* Neural Network node
 - See also* Regression node
 - See also* SAS Code node
 - See also* terminal nodes
 - See also* Transform Variables node
 - See also* Variable Clustering node
 - See also* Variable Selection node
 - Append 53–56, 137
 - AutoNeural node 354–355, 360–362, 363
 - child 196
 - classifying by maximum posterior probability/accuracy 219
 - Cluster 56, 82–85, 533
 - for data modification 94–120
 - Dmine Regression 358–360, 360–362, 363
 - DMNeural 356–358, 360–362, 363
 - Drop 12, 94–95
 - Ensemble 454, 476, 482–484, 485–486
 - File Import 33–36
 - Filter 12, 29–33, 527
 - Gradient Boosting 476, 479–481, 485–486
 - GraphExplore 56, 57, 67–72
 - Impute 12, 98–99, 179–180, 417, 456
 - for initial data exploration 56–94
 - Input Data Source 384, 417, 431
 - Interactive Binning 99–106
 - intermediate 154–155
 - leaf 196, 269
 - Merge 50–53, 185–188
 - MultiPlot 5, 56, 64–67, 416
 - for neural network models 281
 - parent 196
 - Principal Components 106–112
 - Replacement 95–98
 - responder 218
 - Root 155, 196, 257–266
 - sample 27–56
 - Score 232, 308, 319
 - splitting using binary split search 202–203
 - StatExplore 12, 56, 57–64, 94–95, 134, 416
 - Stochastic Boosting 454
 - terminal 154, 196
 - Text Cluster 533, 534–535, 541–544
 - Text Filter 506
 - Text Filtering 521–527, 522
 - Text Import 506, 512–514
 - Text Parsing 506, 516–520, 522, 527–533
 - Text Topic 527–533
 - Time Series 36–50
 - Transformation 438
 - utility 120–126
 - nominal categorical (unordered polychotomous) target, predicting 8–10
 - Nominal Criterion property 203, 207
 - nominal (unordered) target, regression models with 379–383
 - nominal-categorical inputs, continuous target with 147–153
 - nominal-scaled categorical inputs, binary target with 18–162
 - non-responders 270
 - NRBFEQ (Normalized Radial Basis Function with Equal Widths and Heights) 338–340
 - NRBFEV (Normalized Radial Basis Function with Equal Volumes) 346–348
 - NRBFEW (Normalized Radial Basis Function with Equal Widths and Unequal Heights) 343–345
 - NRBFUN (Normalized Radial Basis Function with Unequal Widths and Heights) 346–351
 - Number of Bins property
 - about 155
 - StatExplore node 58
 - Variable Selection node 87
 - Number of Hidden Units property 322–323, 331, 460, 473–475
 - number of levels, of variables 126–128
 - numeric interval-scaled inputs
 - binary target with 153–158
 - continuous target with 140–147
- O**
- observation weights 10
 - observed proportions 196
 - Offset Value property, Transform Variables node 113
 - opening SAS Enterprise Miner 12.1 16
 - operational lag 7
 - optimal binning 50, 113–115
 - optimal tree 201
 - Optimization property 294
 - optimum cut-off point 495–496
 - ORBFEQ (Ordinary Radial Basis Function with Equal Heights and Widths) 333–335

ORBFUN (Ordinary Radial Basis Function with Equal heights and Unequal Widths) 333–335

ORBFUN (Ordinary Radial with Unequal Widths) 325–326

ordered polychotomous targets
See ordinal targets

Ordinal Criterion property 203, 207

ordinal targets
 loss frequency as 309–311
 models for 464–475
 regression models with 373–379

original segment 196

output data sets
 created by Time Series node 45–47
 developing predictive equations created by Text Topic node 532–533

output layer 288–289

overriding default methods 116–118

over-sampling, adjusting predicted probabilities for 274–275

P

p weights 327

parent nodes 196

Partitioning Method property, Data Partition node 29, 456

Pearson Correlations property, StatExplore node 63

Pearson's Chi-square test 270–274

percentage of ranked data (n%) 201

performance window 7, 9

posterior probability
 about 196
 for leaf nodes from training data 215
 of non-response 229
 of response 229

Posterior Probability property 483, 484

Predicted Values property 483

predicting
See also neural network models
 attrition 454–464
 customer attrition 7–8
 loss frequency in auto insurance with Neural Network model 308–321
 nominal categorical (unordered polychotomous) target 8–10
 rate sensitivity of bank deposit products 5–7
 response to direct mail 2–4
 response to direct marketing 221–235
 risk in auto insurance industry 4–5
 risk of accident risk 464–475
 risk with regression tree models 236–244

predictive equations, developing using output data set created by Text Topic node 532–533

predictive modeling
See also textual data, predictive modeling with
 about 16

boosting 476–486

combining 476–486

creating new projects in SAS Enterprise Miner 12.1 16–17

creating process flow diagrams 27

creating SAS data sources 18–27

eigenvalues 75, 129–132

eigenvectors 129–132

exercises 135–136

measurement scale 126–128

nodes for data modification 94–120

nodes for initial data exploration 56–94

number of levels of variable 126–128

opening SAS Enterprise Miner 12.1 16

principal components 129–132

sample nodes 27–56

SAS Enterprise Miner window 17–18

type of variable 126–128

utility nodes 120–126

Preliminary Maximum property, Cluster node 84

pre-processing data 10–12

principal components 129–132

Principal Components node 106–112

Prior Probabilities tab 24

probabilities, adjusted 275

Probit link 385

process flow diagrams 27, 41

profit 493–495
See also customer profitability
See also validation profit
 average vs. total 218
 marginal 494

Profit/Loss criterion 413–414

Project Panel 18

projects, creating in SAS Enterprise Miner 12.1 16–17

promotion window 5

properties
See also specific properties
 of Neural Network node 293–297
 of Regression node 383–415

Properties Panel 18

Proportional Odds model 466–469

pruning trees 201, 211

p-value 59

P-value adjustment options
 Bonferroni Adjustment property 209–210
 depth adjustment 210–211
 Leaf Size property 211
 Threshold Significance Level property 211

Q

quantifying textual data 500–503, 506–507

quantile 113

R

rate sensitivity, predicting of bank deposit products 5–7

RBF (Radial Basis Function) neural network 324–330

Receiver Operating Characteristic (ROC) charts 300–303

recursive partitioning 154, 196

regression

- for continuous targets 431–442
- with large number of input variables 1

regression models

- about 369
- with binary targets 369–373
- business applications 415–442
- exercises 451–452
- Regression node properties 383–415
- types of models developed using 369–383

Regression node

- See also* regression models
- about 12, 13
- Architecture property 460
- Chi-Square criterion 161
- Data Partition node 417, 431
- Decision Tree node 416
- Entry Significance Level property 393–395, 395–397, 400–403, 432, 456
- Hide property 421
- Interval Inputs property 421
- Leaf Role property 439
- Link Function property 373, 383, 384–385, 404
- predictive modeling 532–533
- in process flow 90, 106, 111, 119, 146, 153, 156, 163, 168–169
- properties of 383–415
- regression models 456, 458–460, 464–475
- Regression Type property 373, 383, 384, 404
- Reject property 421
- R-Square criterion 154, 159–160
- Selection Model property 168, 393, 395–397, 404, 425, 467, 532–533
- testing significance of dummy variables 116
- testing variables and transformations 50, 53
- Transform Variables node 416
- transforming variables 185, 186, 187, 188
- Variable Clustering node 407
- variable selection 170, 172, 174, 175, 177, 178
- variable selection in 143
- Variable Selection property 439, 461
- Variables property 111, 146, 156–157

regression tree 202

Regression Type property, Regression node 373, 383, 384, 404

Reject property

- Regression node 421
- Transform Variables node 119, 185, 188
- transforming variables 188

Replacement Editor property, Replacement node 97

Replacement node 95–98

research strategy

- about 1
- alternative modeling strategies 12–13
- defining targets 2–10
- measurement scales for variables 2
- pre-processing data 10–12

residuals, calculating 478–479

responder node 218

responders 270

response

- See also* neural network models
- alternative scenarios of 496
- predicting to direct mail 2–4

revenue 493

risk

- See also* neural network models
- alternative scenarios of 496
- classifying for rate setting 321
- predicting in auto insurance industry 4–5

risk rate 490

ROC (Receiver Operating Characteristic) charts 300–303

Root node 155, 196, 257–266

R-Square criterion 154, 159–160

R-Square selection method 86–87

S

sample nodes

- Append 53–56, 137
- Data Partition 29, 30, 170, 179–180, 291, 293, 311, 417, 431
- File Import 33–36
- Filter 12, 29–33, 527
- Input Data 12, 27–28, 120, 121, 179–180, 221–222, 231, 291, 319, 456, 482
- Merge 50–53, 185–188
- Time Series 36–50

samples, compared with targets 10

SAS Code node

- about 12, 120–126
- building decision tree models 233
- logistic regression 434
- predictive modeling 518
- score ranks in Results window 317

SAS Enterprise Miner

- creating projects in 16–17
- data cleaning after launching 11–12
- data cleaning before launching 11
- developing decision trees in 202–221
- opening 16
- window 17–18

SAS Enterprise Miner: Reference Help 327, 517

SBC (Schwarz Bayesian Criterion) 408–409

Score node 232, 308, 319

- scoring
 - data sets using Neural Network models 305–308
 - datasets with models 319–321
 - showing ranks in Results window 315–318
 - Seasonal property, Time Series node 41–45
 - segments 196
 - See also* leaf nodes
 - Selection Criterion property, Regression node
 - about 403–406
 - Akaike Information Criteria (AIC) 406–407
 - Backward Elimination method 386–392
 - cross validation error 411
 - cross validation misclassification rate 411–412
 - Cross Validation Profit/Loss Criterion 414–415
 - Forward Selection method 393–395, 395–397
 - logistic regression 467
 - predictive modeling with textual data 532–533
 - Profit/Loss Criterion 413–414
 - regression models 423, 425, 438, 456
 - Schwarz Bayesian Criterion (SBC) 408–409
 - validation error 409–410
 - validation misclassification 410–411
 - Validation Profit/Loss Criterion 412–413
 - variable selection 168
 - Selection Default property 397
 - Selection Model property, Regression node 168, 393, 395–397, 404, 425, 467, 532–533
 - Selection Options property 398–400
 - sensitivity
 - See* true positive fraction
 - separation, degree of 204–206
 - Significance Level property 210, 404, 425
 - simple transformation 113
 - Sine function 284
 - Singular Value Decomposition (SVD) 503, 506
 - sources, of modeling data 10
 - Spearman Correlations property, StatExplore node 63
 - specificity
 - See* true positive fraction
 - split point, changing of nominal variables 246–257
 - Split Size property 211
 - splits, measuring worth of 203–209, 207–209
 - splitting
 - groups 101–103
 - nodes using binary split search 202–2037
 - process of 73
 - Splitting Rule Criterion property 269, 458–460, 461
 - splitting value 202
 - StatExplore node 12, 56, 57–64, 94–95, 134, 416
 - Status Bar 18
 - Stay Significance Level property 387, 388, 397, 400–403, 432
 - stepwise selection method
 - about 397
 - when target is binary 398–400
 - when target is continuous 400–403
 - Stochastic Boosting node 454
 - stochastic gradient boosting 479
 - Stop R-Square property, Variable Selection node 88, 143
 - Sub Tree Method property 224
 - sub-segments 196
 - Subtree Assessment Measure property 269
 - Subtree Method property 439, 458–460, 469–473
 - SVD (Singular Value Decomposition) 503, 506
 - SVD Resolution property 534
 - synthetic variables 289
- T**
- Tables to Filter property, Filter node 30
 - Target Activation Function property 465
 - target layer 288–289, 311–315
 - Target Layer Activation Function property 281, 324, 326, 352, 354
 - Target Layer Combination Function property 281, 311–312, 324, 326, 352, 354
 - Target Layer Error Function property 314–315, 324, 326
 - Target Model property, Variable Selection node 86, 92–93, 143, 159, 160
 - target variables, for neural network models 280
 - targets
 - See also* binary targets
 - See also* continuous targets
 - See also* ordinal targets
 - compared with samples 10
 - defining 2–10
 - maximizing relationship to 113–115
 - transformations of 116
 - Targets tab 23
 - Term Weight property 525
 - term weighting 521, 522–527
 - term-document matrix 500–501
 - terminal nodes 154, 196
 - test data
 - roles of in development of decision trees 202
 - testing model performance with 230–231
 - Test property, Data Partition node 29, 417
 - Text Cluster node 533, 534–535, 541–544
 - text files, creating SAS data sets from 509–512
 - Text Filter node 506, 521–527
 - Text Filtering node 552
 - Text Import node 506, 512–514
 - text mining, creating data sources for 514–516
 - Text Parsing node 506, 516–520, 527–533, 552
 - Text Topic node 527–533
 - textual data, predictive modeling with
 - about 499–500
 - creating data sources for text mining 514–516
 - creating SAS data sets from text files 509–512
 - dimension reduction 503–506
 - exercises 545
 - latent semantic indexing 503–506

- quantifying textual data 500–503
 - retrieving documents from World Wide Web 507–508
 - Text Cluster node 533, 534–535, 541–544
 - Text Filter node 506, 521–527
 - Text Import node 506, 514–516
 - Text Parsing node 506, 516–520, 527–533, 552
 - Text Topic node 527–533
 - Threshold Significance Level property 211
 - Time Series node 36–50
 - %TMFILTER macro 506, 507–508
 - Toolbar 17
 - Toolbar Shortcut Buttons 18
 - tools
 - See* nodes
 - total profit, vs. average profit for comparing tree size 218
 - training, of trees 198
 - training data
 - developing trees using 214–215
 - roles of in development of decision trees 201
 - training data set 269
 - Training property, Data Partition node 29, 417
 - transaction data
 - converting to time series 36–38
 - creating data sources for 38–41
 - transform variables, saving code generated by 189
 - Transform Variables node
 - See also* variable selection
 - about 112–120, 139–140, 190
 - Class Inputs property 438
 - Hide property 185, 188
 - Interval Inputs property 113, 116, 181, 185, 188
 - Merging property 187
 - Multiple Method property 188
 - Node ID property 186, 187
 - in process flow 137
 - Regression node 416
 - Reject property 185, 188
 - testing variables and transformations 50, 51–52, 53
 - transforming variables 181–185, 185–188, 186
 - transforming variables with 179–180
 - Variables property 116
 - Transformation node 438
 - transformations
 - after variable selection 183–185
 - binning 113
 - of class inputs 116
 - for interval inputs 113–115
 - multiple using Multiple Method property 188–189
 - passing more than one for each interval input 185–189
 - passing two types using Merge node 185–188
 - simple 113
 - of targets 116
 - before variable selection 181–182
 - of variables 179–189
 - TRANSPOSE procedure 520
 - Treat Missing as Level property
 - Interactive Binning node 100
 - Regression node 438
 - trees
 - about 196
 - assessing using Average Square Error 220
 - true positive fraction 300
- ## U
- unadjusted probabilities, expected profits using 275
 - ungrouped variables, compared with categorical variables 192
 - unordered (nominal) target, regression models with 379–383
 - Use AOV16 Variables property
 - Dmine Regression node 359
 - Variable Selection node 8, 93, 142, 144, 147
 - Use Group Variables property, Variable Selection node 88, 93, 144, 147–148
 - Use Selection Defaults property 387, 393, 395–397, 456
 - user-defined networks 354
 - user-specified architectures 351–354
 - utility nodes 120–126
- ## V
- validation accuracy 219
 - validation data
 - pruning trees using 211–213
 - roles of in development of decision trees 201
 - validation error 409–410
 - Validation Error criterion 456
 - validation misclassification 410–411
 - validation profit 201, 216–218
 - Validation Profit/Loss criterion 412–413
 - Validation property, Data Partition node 29, 417
 - variable clustering, using example data set 77–82
 - Variable Clustering node
 - about 56, 73–82, 139–140, 190
 - Include Class Variables property 163
 - Maximum Clusters property 73
 - Maximum Eigenvalue property 73, 75
 - Regression node 407
 - Variable Selection property 163
 - variable selection using 162–176
 - variable selection
 - See also* Transform Variables node
 - about 139–140
 - binary target with nominal-scaled categorical inputs 158–162
 - binary target with numeric interval-scaled inputs 153–158

- continuous target with nominal-categorical inputs 147–153
- continuous target with numeric interval-scaled inputs 140–147
- exercises 192–193
- transformation after 183–185
- transformation before 181–182
- using Decision Tree node 176–179
- using Variable Clustering node 162–176
- Variable Selection node
 - about 56, 85, 188, 190
 - Hide Rejected Variables property 142
 - Minimum R-Square property 88, 142–143
 - regression models 416
 - Stop R-Square property 88, 143
 - transforming variables 179–180, 181–185
- Variable Selection property
 - Regression node 439, 461
 - Variable Clustering node 163
- variables
 - assigning to clusters 76
 - categorical 2, 192
 - changing measurement scale of in data sources 190–191
 - explanatory 282
 - interval 2
 - measurement scale of 2, 126–128
 - number of levels of 126–128
 - selecting for clusters 164–174
 - synthetic 289
 - transformation of 179–189
 - types of 126–128
- Variables property
 - about 190
 - Drop node 95
 - File Import node 35
 - Impute node 9
 - Regression node 111, 146, 156–157
 - Transform Variables node 116
 - viewing properties 26
- variance
 - of inputs 130
 - proportion explained by cluster component 76–77
 - proportion of explained by principal components 132
- Variation Proportion property, Variable Clustering node 74
- Voting Posterior Probabilities property 483–484
- Voting...Average method 483–484
- Voting..Proportion method 484

W

- weights
 - estimating in neural network models 290–291
 - selecting for Neural Network node 303–305
- windows
 - Metadata Advisor Options 21
 - SAS Enterprise Miner 17–18
- World Wide Web, retrieving documents from 507–508

X

- XRadial value 351

Special Characters

- : (colon) 79
- , (comma) 26–28, 205
- (dash) 79
- = (equal sign) 79–81
- > (greater than) 79, 81
- < (less than) 79, 81
- () parentheses 28–29, 41–42
- . (period) 79, 159
- + (plus sign) 79
- _ (underscore) 79, 83
- xn sequency 282–284
- xz formatting sequences 287–289

