Chapter 4
Data Preprocessing for Credit Risk Modeling

Data is the key ingredient for any credit risk model (Baesens 2014). Thus, it is vital to thoroughly consider and list all data sources of potential interest and relevance before modeling credit risk parameters such as probability of default (PD), loss given default (LGD), or exposure at default (EAD). Large experiments as well as a broad experience in different fields indicate that when it comes to data, bigger is better (Junqué de Fortuny, Martens, and Provost 2013). However, real-life credit risk data can be and typically is dirty because of inconsistencies, incompleteness, duplication, merging, and many other problems. Throughout the modeling steps, various data-filtering mechanisms will be applied to clean up and reduce the data to a manageable and relevant size. Worth mentioning here is the garbage in, garbage out (GIGO) principle, which essentially states that messy data will yield messy analytical models. It is of utmost importance that every data preprocessing step is carefully justified, carried out, validated, and documented before proceeding with further analysis; even the slightest mistake can make the data totally unusable for further analysis and the results invalid. In what follows, we will elaborate on the most important data preprocessing steps that should be considered during a credit risk modeling exercise. The activities discussed will apply to PD, LGD, and EAD modeling.

TYPES OF DATA SOURCES

First, let us have a closer look at what data to gather. Data can originate from a variety of different sources and provide different types of information that might be useful for the purpose of credit risk modeling, as will be further discussed in this section. The provided mixed discussion of different sources and types of data concerns a *broad*, *nonexhaustive*, and *non–mutually exclusive* categorization. We discuss the most prominent data sources and types of information available in a typical organization, but clearly not all possible data sources and types of information. Furthermore, some overlap may exist between the enlisted categories.

Transactional data is a first important source of data. It consists of structured and detailed information capturing the key characteristics of a customer transaction (e.g., cash transfer, installment payment). It is usually stored in massive online transaction processing (OLTP) relational databases. This data can also be summarized over longer time horizons by aggregating it into averages, absolute or relative trends, maximum or minimum values, and so forth.

Contractual, subscription, or account data may complement transactional data. Contractual data includes information about the type of product (e.g., loan) combined with customer characteristics. Examples of subscription data are the start date of the relationship, characteristics of a subscription such as type of services or products delivered, levels of

service, cost of service, product guarantees, and insurances. The moment when a customer subscribes to a service offers a unique opportunity for the organization to get to know the customer—unique in the sense that it may be the only time when a direct contact exists between the bank and the customer, either in person, over the phone, or online—and as such it offers the opportunity to gather additional information that is nonessential to the contract but may be useful for credit risk modeling. Such information is typically stored in an account management or customer relationship management (CRM) database.

Subscription data may also be a source of **sociodemographic information**, since subscription or registration typically requires identification. Examples of socioeconomic characteristics of a population consisting of customers are *age*, *gender*, *marital status*, *income level*, *education level*, *occupation*, and *religion*. Although not very advanced or complex measures, sociodemographic information may significantly relate to credit risk behavior. For instance, it appears that both gender and age are very often related to an individual's likelihood to default: Women and older individuals are less likely to default than men and younger customers. Similar characteristics can also be defined when the basic entities for which default is to be detected do not concern individuals but instead companies or organizations. In such a setting one rather speaks of slow-moving data dimensions, factual data, or static characteristics. Examples include the address, year of foundation, industrial sector, and activity type. These do not change over time at all, or do not change as often as do other characteristics such as turnover, solvency, number of employees, and so on. These latter variables are examples of what we will call behavioral information. Several data sources may be consulted for retrieving sociodemographic or factual data, including subscription data sources as discussed previously, as well as data poolers, survey data, and publicly available data sources as discussed next.

In recent times, **data poolers** have increased in importance in the credit risk modeling industry. Examples are Experian, Equifax, CIFAS, Dun & Bradstreet, Thomson Reuters, and so on. The core business of these companies is to gather data (e.g., sociodemographic information) in particular settings or for particular purposes (e.g., credit risk assessment, fraud detection, and marketing) and sell it to interested customers looking to enrich or extend their data sources. In addition to selling data, these data poolers typically also build predictive models themselves and sell the outputs of these models as risk scores. This is a common practice in credit risk; for instance, in the United States the FICO score is a credit score ranging between 300 and 850 provided by the three most important credit data poolers or credit bureaus: Experian, Equifax, and TransUnion.[1] Many financial institutions as well as commercial vendors that give credit to customers use these FICO scores either as their final internal model to assess creditworthiness or to benchmark it against an internally developed credit scorecard to better understand the weaknesses of the latter.

Surveys are another source of data, and this information is gathered via offline methods such as mail, or via online modes including telephone, website, and social media interactions (e.g., Facebook, LinkedIn, or Twitter). Surveys may aim at gathering sociodemographic data, but also behavioral information.

Behavioral information concerns any information describing the behavior of an individual or an entity in the particular context under study. Such data is also called fast-moving data or dynamic characteristics. Examples of behavioral variables include information with regard to preferences of customers, usage information, frequencies of events, and trend variables. When dealing with organizations, examples of behavioral characteristics or dynamic characteristics are *turnover*, *solvency*, or *number of employees*. Marketing data results from monitoring the impact of marketing actions on the target population, and concerns a particular type of behavioral information.

Also, **unstructured data** embedded in text documents (e.g., e-mails, web pages, claim forms) or multimedia content can be interesting to analyze. However, these sources typically require extensive preprocessing before they can be successfully included in a credit risk modeling exercise. Analyzing textual data is the goal of a particular branch of analytics (i.e., text analytics). Given the high level of specialization involved, this book does not provide an extensive discussion of text mining techniques. For more information on this topic, you could consult academic textbooks on the subject (Chakraborty, Murali, and Satish 2013; Miner et al. 2012).

A second type of unstructured information is **contextual or network information**, meaning the context of a particular entity. An example of such contextual information concerns relations of a particular type that exist between an entity and other entities of the same or a different type. An example in credit risk modeling could be liquidity dependencies between corporate counterparts. Taking into account these complex network relationships allows us to model system risk whereby the default of one company may create a knock-on effect in the network of interconnected companies.

Another important source of data is **qualitative, expert-based data**. An expert is a person with a substantial amount of subject matter expertise within a particular setting (e.g., credit portfolio manager, brand manager). The expertise stems from both common sense and business experience, and it is important to elicit this knowledge as much as possible before the credit risk model building exercise commences. It will allow for steering the modeling in the right direction and interpreting the analytical results from the right perspective. A popular example of applying expert-based validation is checking the univariate signs of a regression model. For instance, an example already discussed relates to the observation that a higher age often results in a lower credit risk. Consequently, a negative sign is expected when including age in a default risk model yielding the probability of an individual being a defaulter. If this turns out not to be the case (e.g., due to bad data quality or multicollinearity), the expert or business user will not be tempted to use the credit risk model at all, since it contradicts prior expectations.

A final source of information concerns **publicly available** data sources, which can provide **external information**. This is contextual information that is not related to a particular entity, such as macroeconomic data (e.g., gross domestic product [GDP], inflation, unemployment). By enriching the data set with such information, you may see, for example, how the model and the model outputs vary as a function of the state of the economy. This information can then be used to calibrate or stress test the credit risk model.

Also, social media data from publicly available sources like Facebook, Twitter, or LinkedIn can be an important source of information. However, you need to be careful when both gathering and using such data and ensure that local and international privacy regulations are respected at all times.

MERGING DATA SOURCES

Building a credit risk model typically requires or presumes the data to be presented in a single table containing and representing all the data in a structured manner. A structured data table allows straightforward processing and analysis.

The rows of a data table typically represent the basic entities to which the analysis applies (e.g., customers, companies, and countries). The rows are referred to as instances, observations, or lines. The columns in the data table contain information about the basic entities. Many synonyms are used to denote the columns of the data table, such as (explanatory) variables, fields, characteristics, attributes, indicators, or features.

In order to construct the aggregated, nonnormalized data table to facilitate further analysis, often several normalized source data tables have to be merged. Merging tables involves selecting information from different tables related to an individual entity, and copying it to the aggregated data table. The individual entity can be recognized and selected in the different tables by making use of *keys*, which are attributes that have been included in the table exactly to allow identifying and relating observations from different source tables pertaining to the same entity. Exhibit 4.1 illustrates the process of merging two tables (i.e., transaction data and customer data) into a single nonnormalized data table by making use of the key attribute *ID*, which allows connecting observations in the transactions table with observations in the customer table. The same approach can be taken to merge as many tables as required, but clearly, the more tables that are merged, the more duplicate data might be included in the resulting table.

Transactions		
ID	Date	Amount
XWV	2/01/2015	52 €
XWV	6/02/2015	21 €
XWV	3/03/2015	13 €
BBC	17/02/2015	45 €
BBC	1/03/2015	75 €
VVQ	2/03/2015	56 €

Customer data		
ID	Age	Start date
XWV	31	1/01/2015
BBC	49	10/02/2015
VVQ	21	15/02/2015

Non-normalized data table				
ID	Date	Amount	Age	Start date
XWV	2/01/2015	52 €	31	1/01/2015
XWV	6/02/2015	21 €	31	1/01/2015
XWV	3/03/2015	13 €	31	1/01/2015
BBC	17/02/2015	45 €	49	10/02/2015
BBC	1/03/2015	75 €	49	10/02/2015
VVQ	2/03/2015	56 €	21	15/02/2015

Exhibit 4.1 Aggregating Normalized Data Tables into a Non-normalized Data Table

When merging data tables, it is crucial that no errors occur, so some checks should be applied to control the resulting table and to make sure that all information is correctly integrated.

Data sets can be merged in SAS after sorting by the key variable using the DATA/MERGE command. Other ways to combine data sets in Base SAS include concatenating (DATA/SET command), appending (PROC APPEND), and updating (DATA/UPDATE command).

SAMPLING

We already discussed sampling in Chapter 3, Exploratory Data Analysis. Within the context of building scorecards, the aim of sampling is to take a subset of historical data (e.g., past customers), and use that to build the credit risk model. A first obvious question that comes to mind concerns the need for sampling. With the availability of high-performance computing facilities (e.g., grid and cloud computing), you could also try to directly analyze the full data set. However, a key requirement for a good sample is that it should be representative for the future entities on which the credit risk model will be run. Hence, the timing aspect becomes important since customers of today are more similar to customers of tomorrow than customers of yesterday are. Choosing the optimal time window of the sample involves a trade-off between lots of data (and hence a more robust credit risk model) and recent data (which may be more representative). The sample should also be taken from a representative business period to produce a picture of the target population that is as accurate as possible.

It speaks for itself that sampling bias should be avoided as much as possible. However, this is not always that straightforward. Let's take the example of credit scoring. Assume you want to build an application scorecard to score mortgage applications. The future population then consists of all customers who knock on the door of the bank and apply for a mortgage—the so-called through-the-door (TTD) population. You then need a subset of the historical through-the-door population to build an analytical model. However, in the past the bank was already

applying a credit policy (either expert based or based on a previous analytical model). This implies that the historical TTD population has two subsets: the customers who were accepted with the old policy, and the ones who were rejected (see Exhibit 4.2). Obviously, for the latter, we don't know the target value since they were never granted the credit. When building a sample, you can then only make use of the accepts, which clearly implies a bias. Procedures to deal with reject inference have been suggested in the literature (Thomas, Edelman, and Crook 2002). We discuss some of these in Chapter 5 on credit scoring.

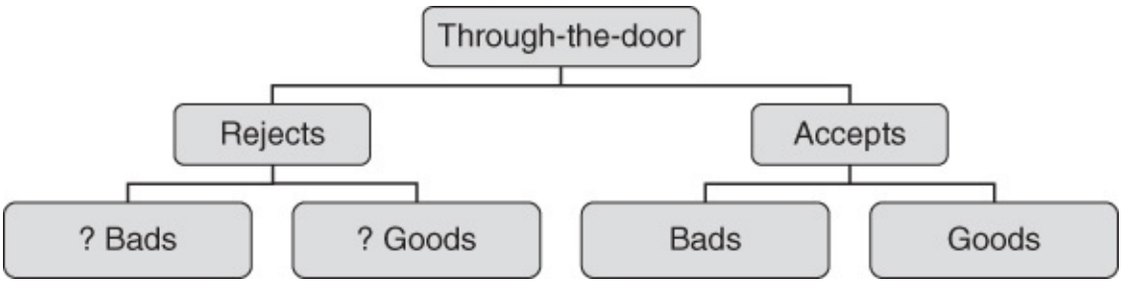


Exhibit 4.2 The Reject Inference Problem in Credit Scoring

Another potential bias concerns seasonality effects. Since every month may in fact deviate from *normal*, if normal is defined as average, it could make sense to build separate models for different months, or for *homogeneous* time frames. This is a rather complex and demanding solution from an operational perspective, since multiple models have to be developed, run, maintained, and monitored. Alternatively, a sample may be gathered by sampling observations over a period covering a full business cycle. Then only a single model has to be developed, run, maintained, and monitored, which may possibly come at a cost of reduced performance since it's less tailored to a particular time frame. However, this approach will be less complex and costly to operate.

In stratified sampling, a sample is taken according to predefined strata. In a default risk context, data sets are typically very skewed (e.g., 99 percent nondefaulters and 1 percent defaulters). When stratifying according to the target default indicator, the sample will contain exactly the same percentages of default and nondefault customers as in the original data. Additional stratification can be applied on predictor variables as well, for instance in order for the number of observations across different industry categories to closely resemble the real industry category distribution. However, as long as no large deviations exist with respect to the sample and observed distribution of predictor variables, it will usually be sufficient to limit stratification to the target variable.

In Base SAS, PROC SURVEYSELECT can be used to create a stratified sample as follows:

```
PROC SORT DATA=data.hmeq;
   BY bad;
RUN;
PROC SURVEYSELECT DATA=data.hmeq
METHOD=SRS N=1000 SEED=12345 OUT=data.mySample;
STRATA bad / ALLOC=PROP;
RUN;
```

Before a stratified sample can be created, the data needs to be sorted on the stratification variable using PROC SORT. The options of PROC SURVEYSELECT can be explained as follows:

- METHOD=SRS: Applies simple random sampling.
- N=1000: Creates a sample of 1,000 observations.
- SEED=12345: The seed needed to randomly sample the observations.
- OUT=data.mySample: The result will be stored in the data set mySample.
- STRATA bad / ALLOC=PROP: The stratification variable is bad and the allocation method used is proportional.

We can now use PROC FREQ to contrast the bad distribution of the sample with the original data as follows:

```
PROC FREQ DATA=data.hmeq;
TABLES bad;
RUN;
PROC FREQ DATA=data.mySample;
TABLES bad;
RUN;
```

For the original data HMEQ, this gives the result shown in Exhibit 4.3.

Exhibit 4.3 The FREQ Procedure

The FREQ Procedure				
BAD	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
0	4771	80.05	4771	80.05
1	1189	19.95	5960	100.00

For the sample mySample, this gives the result shown in Exhibit 4.4.

Exhibit 4.4 The FREQ Procedure

The FREQ Procedure				
BAD	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
0	801	80.10	801	80.10
1	199	19.90	1000	100.00

It can be seen that, thanks to the stratification, the percentage of defaults is similar in the sample and the original data.

In SAS Enterprise Miner, a sample can be created using the Sample node from the Sample tab (see Exhibit 4.5). Various sample methods are supported such as First N, Stratify, Random, Cluster, and Systematic. In case a class variable is present (e.g., in case of modeling default

risk), the default setting is to stratify based on the class variable. The random seed is used for randomly selecting the observations. The same random seed will result in the same random sample. The size of the sample can be specified as the number of observations or as a percentage of the total number of observations.

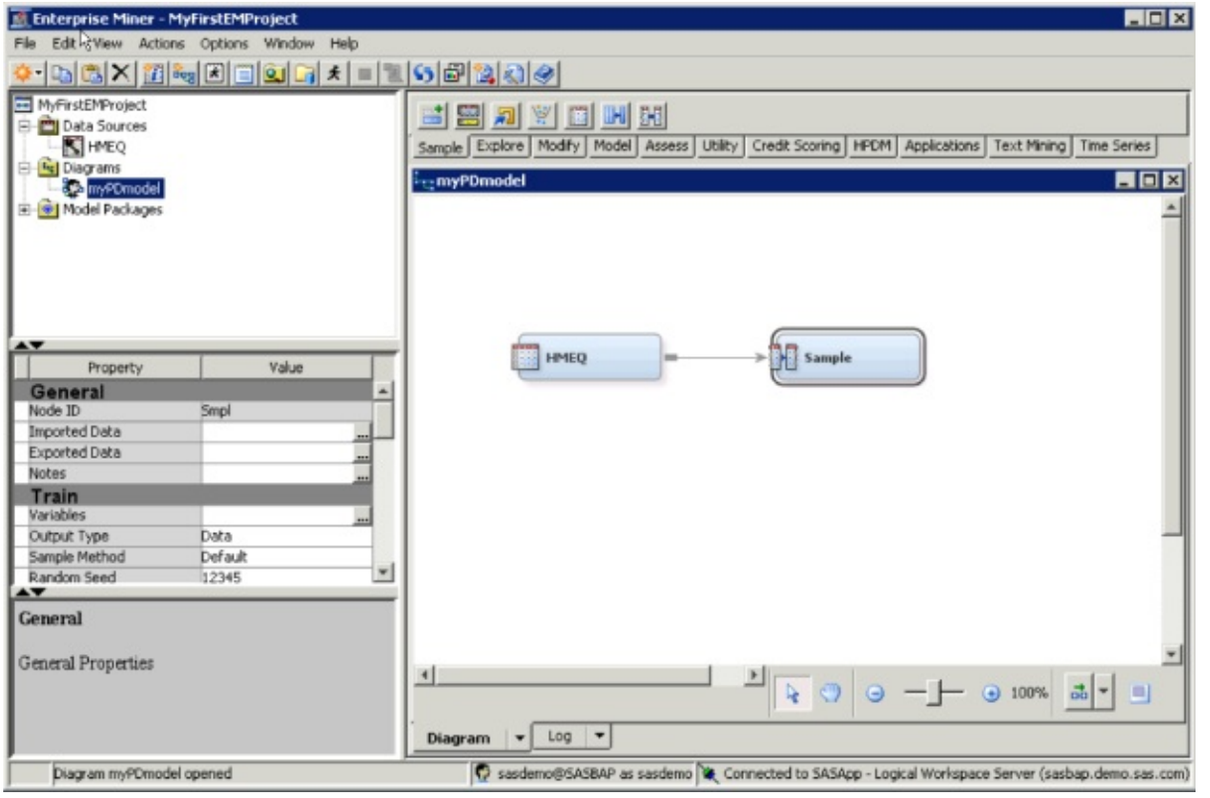


Exhibit 4.5 Sampling in SAS Enterprise Miner

TYPES OF DATA ELEMENTS

It is important to appropriately consider the different types of data elements at the start of each credit risk modeling exercise. The following types of data elements can be considered:

- Continuous data

  These are data elements that are defined on an interval that can be either limited or unlimited.

  A distinction is sometimes made between continuous data with and without a natural zero value; they are respectively referred to as ratio data (e.g., amounts) and interval data (e.g., temperature in degrees Celsius or Fahrenheit). In the latter case, the interval between values is interpretable. However, it does not make sense to take the ratio of two values and make statements such as "It is double or twice as hot as last month." Most continuous data in a credit risk modeling setting concerns ratio data, since we are often dealing with amounts.

  Examples: loan-to-value (LTV) ratio; income; balance on checking/savings account.

- Categorical data
  - Nominal. These are data elements that can only take on a limited set of values with no meaningful ordering in between.

    Examples: payment type; industry sector.
  - Ordinal. These are data elements that can only take on a limited set of values with a meaningful ordering in between.

    Examples: credit rating; age coded as young, middle-aged, or old.
  - Binary. These are data elements that can only take on one of two values.

    Examples: default (yes/no); employed (yes/no).

Appropriately distinguishing between these different data elements is of key importance to start the analysis when importing the data into SAS. For example, if marital status were incorrectly specified as a continuous data element, then SAS would calculate its mean, standard deviation, and so on, which is obviously meaningless and may perturb the analysis. In SAS Enterprise Miner, the type of variable can be specified by setting the measurement level as Binary, Interval, Nominal, Ordinal, or Unary (see Exhibit 4.6). Note that the last level represents a variable with only one value, which is meaningless for modeling.

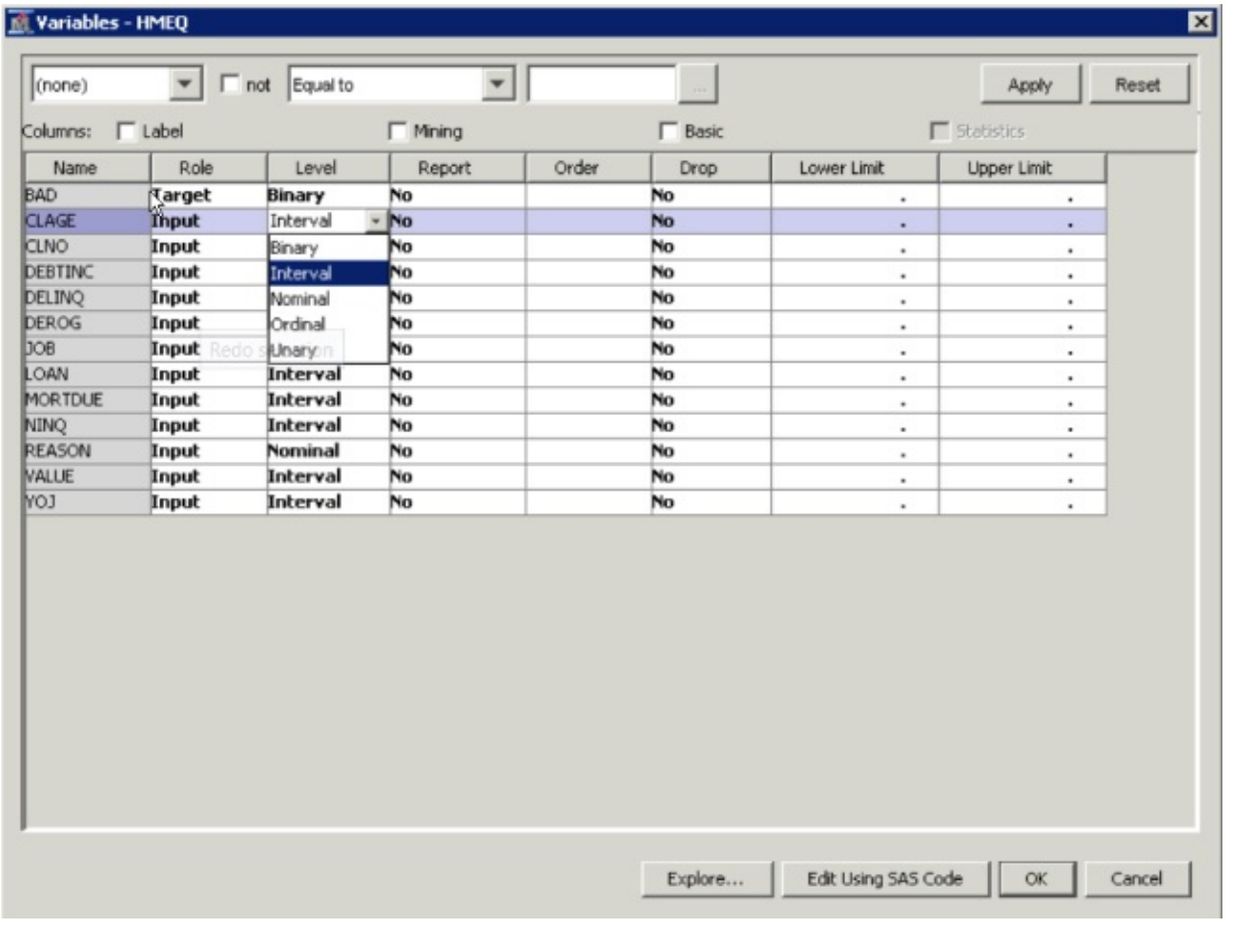


Exhibit 4.6 Setting the Measurement Level of Variables in SAS Enterprise Miner

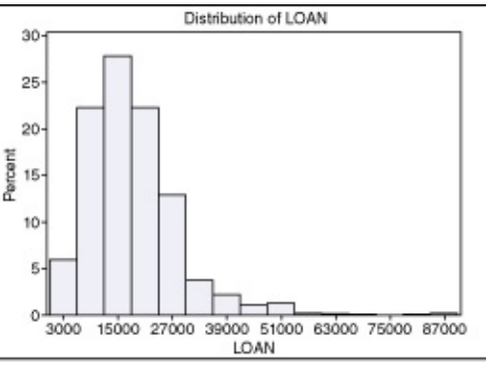
VISUAL DATA EXPLORATION AND EXPLORATORY STATISTICAL ANALYSIS

Visual data exploration is a very important step in getting to know your data in an informal way. It allows you to gain some initial insights into the data that can then be usefully adopted throughout the modeling stage. Different plots/graphs can be useful here. Bar charts represent the frequency of each of the values (either absolute or relative) as bars. A bar chart or histogram provides an easy way to visualize the central tendency and to determine the variability or spread of the data. It also allows you to contrast the observed data with standard known distributions (e.g., the normal distribution). A pie chart represents a variable's distribution as a pie, whereby each section represents the percentage taken by each value of the variable. The total of all pie slices is equal to 100 percent.

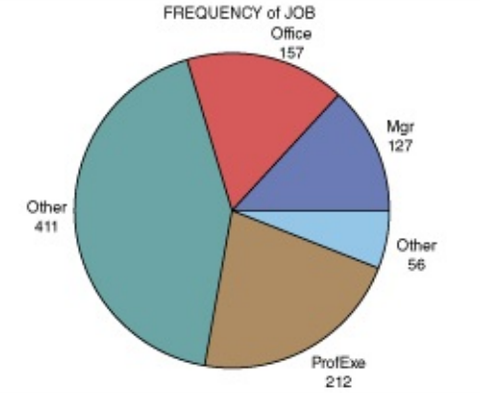
Other handy visual tools are scatter plots. Plots such as these allow you to visualize one variable against another to see whether there are any correlation patterns in the data. Also, online analytical processing (OLAP)-based multidimensional data analysis can be usefully adopted to explore patterns in the data.

Exhibit 4.7 illustrates how to create histograms, pie charts, and scatter plots in Base SAS using PROC UNIVARIATE, PROC GCHART, and PROC GPLOT.

```
/** histogram for loan variable**/
PROC UNIVARIATE DATA=data.mySample;
VAR loan;
HISTOGRAM;
RUN;
```




```
/** pie chart for job variable**/
PROC GCHART DATA=data.mySample;
  PIE job;
  RUN;
```




```
/** scatter plot of loan versus
mortdue**/
PROC GPLOT DATA=data.mySample;
PLOT loan*mortdue;
RUN;
```

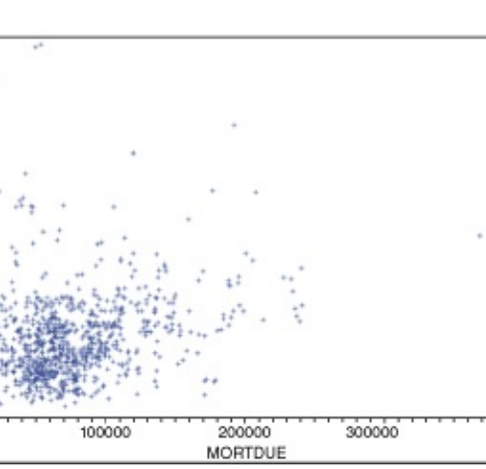


Exhibit 4.7 Plots in Base SAS

In SAS Enterprise Miner, the MultiPlot node from the Explore tab is a handy node to visually explore your data (see Exhibit 4.8). It allows you to create both histograms and scatter plots for each of the variables. Exhibit 4.9 displays the histogram of job status versus good/bad status.

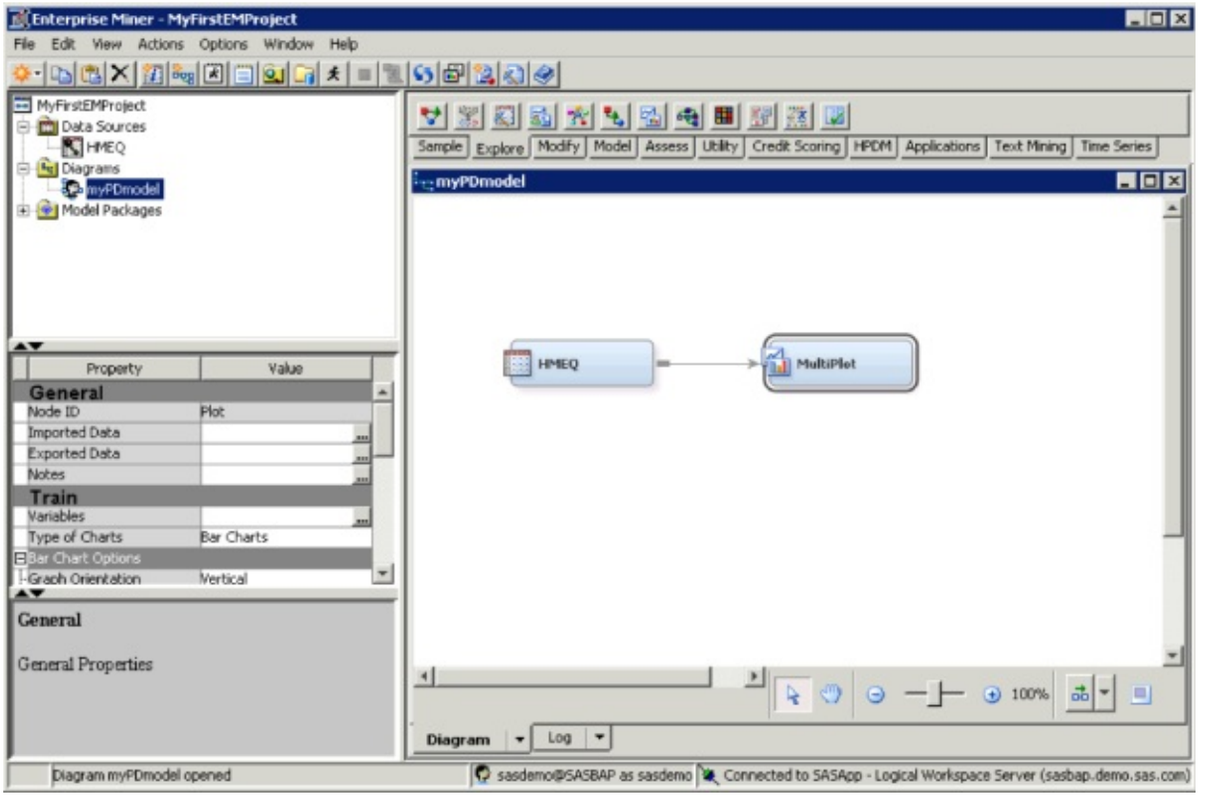


Exhibit 4.8 The Multiplot Node in SAS Enterprise Miner

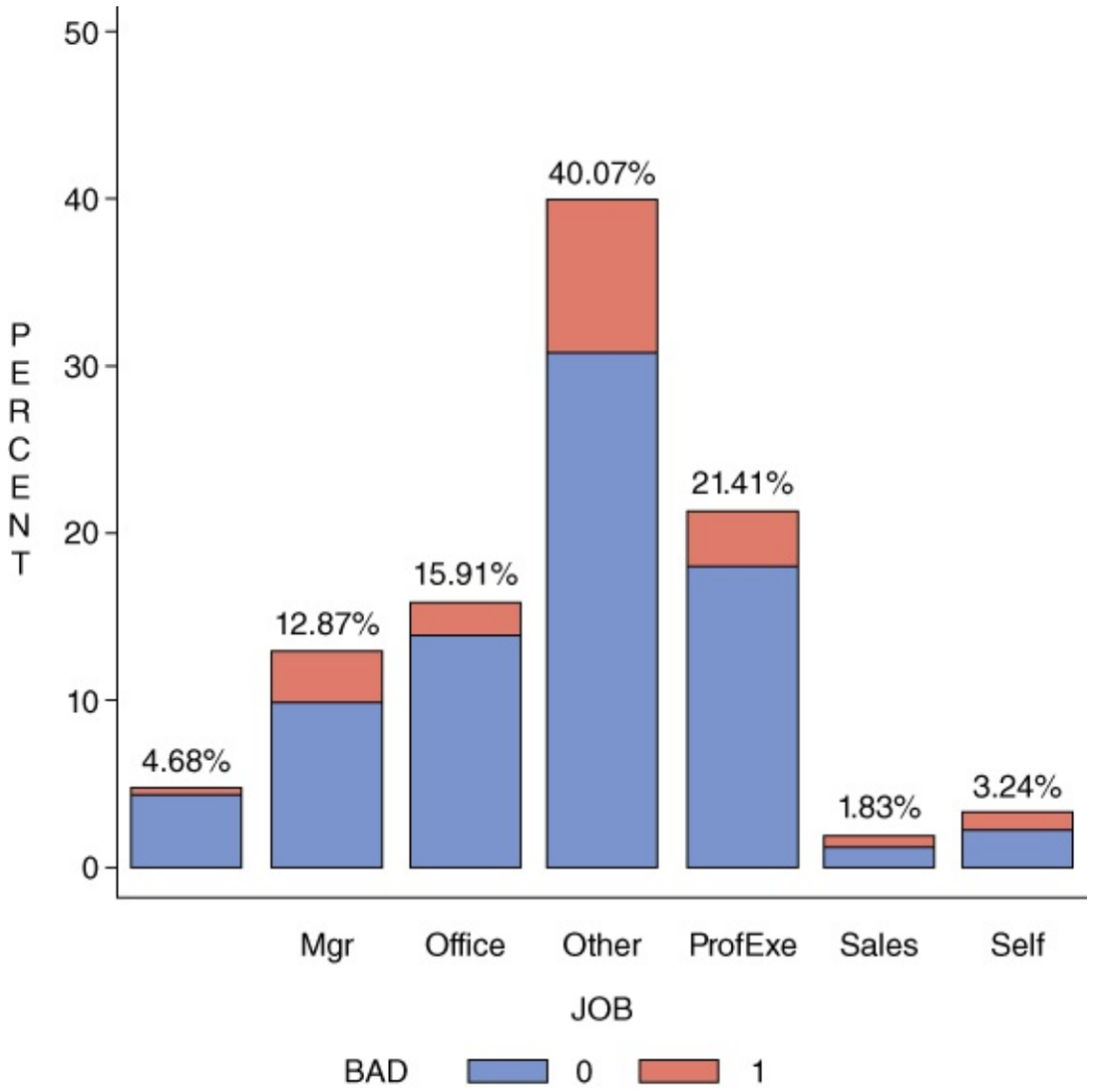


Exhibit 4.9 Histogram of Job Status versus Good/Bad Status

DESCRIPTIVE STATISTICS

In addition to the preparatory visual data exploration, several descriptive statistics might be calculated that provide basic insight or *feeling* for the data. Plenty of descriptive statistics exist that all summarize or provide information with respect to a particular characteristic of the data, and therefore descriptive statistics should be assessed together (i.e., in support and completion of each other). We have already discussed most of these in the exploratory data analysis chapter, and quickly refresh them here.

Basic descriptive statistics are the mean and median value of continuous variables, with the median value being less sensitive to extreme values (i.e., outliers), but not providing as much information with respect to the full distribution. Complementary to the mean value, the variation or the standard deviation provides insight with respect to how much the data is spread around the mean value. Likewise, percentile values such as the 10th, 25th, 75th, and 90th percentile provide further information with respect to the distribution and as a complement to the median value.

Specific descriptive statistics exist to express the symmetry or asymmetry of a distribution, such as the *skewness* measure, as well as the peakedness or flatness of a distribution (e.g., the *kurtosis* measure). However, the exact values of these measures are likely a bit harder to interpret than the value of the mean and the standard deviation, for instance. This limits their practical use. Instead, you could more easily assess these aspects by inspecting visual plots of the distributions of the involved variables.

When dealing with categorical variables, instead of the median and the mean value you may calculate the mode, which is the most frequently occurring value. In other words, the mode is the most typical value for the variable at hand. The mode is not necessarily unique, since multiple values can result in the same maximum frequency.

Descriptive statistics can be calculated in Base SAS using PROC UNIVARIATE as follows:

```
PROC UNIVARIATE DATA=data.mySample;
VAR loan;
RUN;
```

The results are displayed in Exhibit 4.10.

The UNIVARIATE Procedure			
Variable: LOAN			
Moments			
N	1000	**Sum Weights**	1000
Mean	17898.3	**Sum Observations**	17898300
Std Deviation	10331.742	**Variance**	106744892
Skewness	**1.92510314**	**Kurtosis**	**7.2790685**
Uncorrected SS	4.26987E11	**Corrected SS**	**1.06638E11**
Coeff Variation	**57.7247111**	**Std Error Mean**	**326.718368**

Basic Statistical Measures			
Location		Variability	
Mean	17898.30	**Std Deviation**	10332
Median	16000.00	**Variance**	106744892
Mode	15000.00	**Range**	87200
		Interquartile Range	11750

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	**t**	54.78204	**Pr > |t|**	<.0001
Sign	**M**	500	**Pr >= |M|**	<.0001
Signed Rank	**S**	250250	**Pr >= |S|**	<.0001

Quantiles (Definition 5)	
Level	**Quantile**
100% Max	89200
99%	51900
95%	36350
90%	28250
75% Q3	22500
50% Median	16000
25% Q1	10750
10%	7200
5%	5600
1%	3800
0% Min	2000

Extreme Observations			
Lowest		Highest	
Value	**Obs**	**Value**	**Obs**
2000	803	62900	797
2000	802	69700	798
2400	804	80300	799
3000	805	88500	800
3000	1	89200	801

Exhibit 4.10 Results of PROC Univariate

In SAS Enterprise Miner, the StatExplore node from the Explore tab can be used to calculate the descriptive statistics (see Exhibit 4.11). This node will calculate various descriptive statistics for each of the variables (see Exhibit 4.12) and also report these for each of the target classes individually (see Exhibit 4.13).

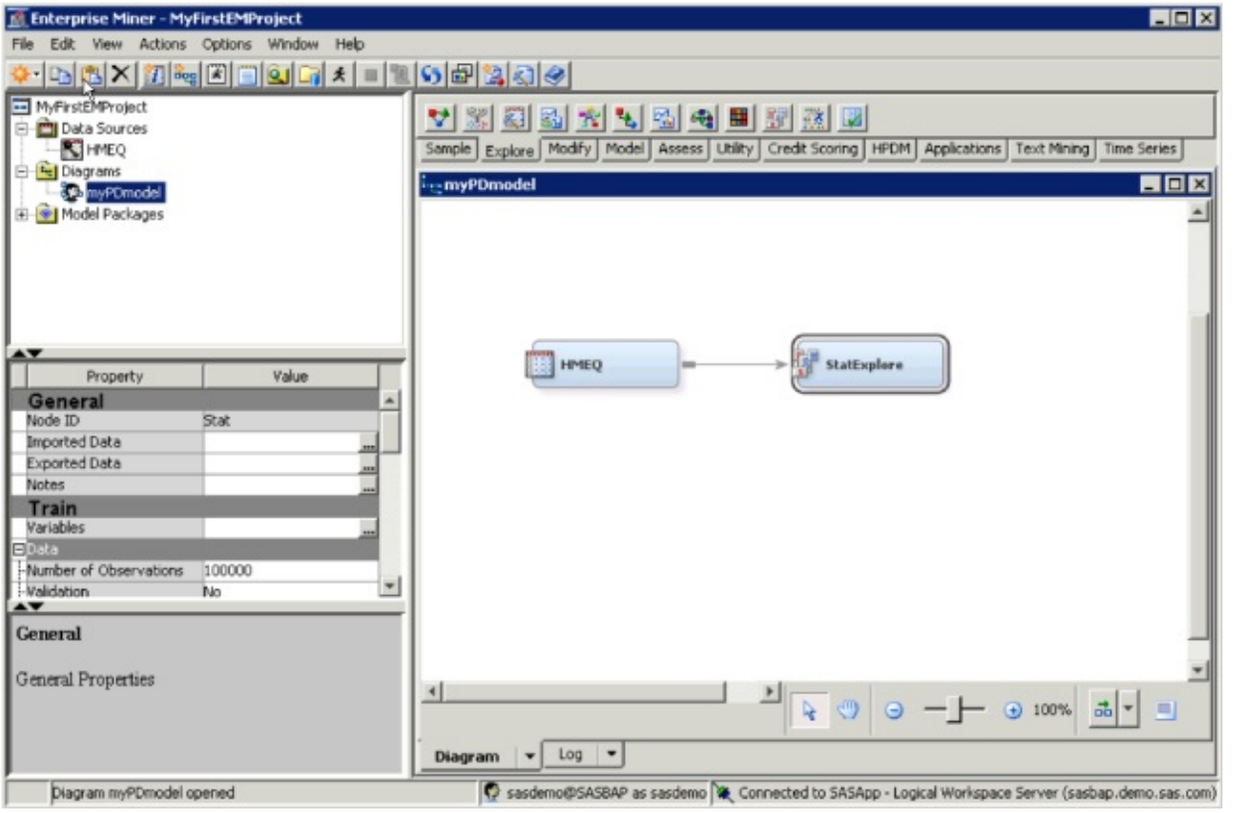


Exhibit 4.11 The StatExplore Node in SAS Enterprise Miner

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
CLAGE	INPUT	179.7663	85.81009	5652	308	0	173.4667	1168.234	1.343412	7.599549
CLNO	INPUT	21.2961	10.13893	5738	222	0	20	71	0.775052	1.157673
DEBTINC	INPUT	33.77992	8.601746	4693	1267	0.524499	34.81696	203.3121	2.852353	50.50404
DELINQ	INPUT	0.449442	1.127266	5380	580	0	0	15	4.02315	23.56545
DEROG	INPUT	0.25457	0.846047	5252	708	0	0	10	5.32087	36.87276
LOAN	INPUT	18607.97	11207.48	5960	0	1100	16300	89900	2.023781	6.93259
MORTDUE	INPUT	73760.82	44457.61	5442	518	2063	65017	399550	1.814481	6.481866
NINQ	INPUT	1.186055	1.728675	5450	510	0	1	17	2.621984	9.786507
VALUE	INPUT	101776	57385.78	5848	112	8000	89231	855909	3.053344	24.3628
YOJ	INPUT	8.922268	7.573982	5445	515	0	7	41	0.98846	0.372072

Exhibit 4.12 Descriptive Statistics for the HMEQ Data Set

Data Role=TRAIN Variable=CLAGE

Target	Target Level	Median	Missing	Non Missing	Minimum	Maximum	Mean	Standard Deviation	Skewness	Kurtosis
BAD	0	180.2841	230	4541	0.486711	649.7471	187.0024	84.46522	0.901459	2.191917
BAD	1	132.8667	78	1111	0	1168.234	150.1902	84.95229	3.480899	35.27536
OVERALL		173.4667	308	5652	0	1168.234	179.7663	85.81009	1.343412	7.599549

Exhibit 4.13 Class Conditional Descriptive Statistics for the HMEQ Data Set

MISSING VALUES

Missing values can occur for various reasons. The information can be nonapplicable. For example, when modeling the amount of loss for obligors, then this information is available only for the defaulters and not for the nondefaulters since it is not applicable there. The information can also be undisclosed, such as when a customer decides not to disclose his or her income for privacy reasons. Missing data can also originate because of an error during merging (e.g.,

typos in name or ID).

Some analytical techniques (e.g., decision trees) can deal directly with missing values. Other techniques need some additional preprocessing. The following are the most popular schemes that deal with missing values (Little and Rubin 2002, 408):

- **Replace (impute)**. This implies replacing the missing value with a known value. For example, consider the example in Exhibit 4.14. You could impute the missing credit bureau scores with the average or median of the known values. For marital status the mode can then be used. You could also apply regression-based imputation whereby a regression model is estimated to model a target variable (e.g., credit bureau score) based on the other information available (e.g., age, income). The latter is more sophisticated, although the added value from an empirical viewpoint (e.g., in terms of model performance) is questionable.

ID	Age	Income	Marital Status	Credit Bureau Score	Default
1	34	1,800	?	620	Yes
2	28	1,200	Single	?	No
3	22	1,000	Single	?	No
4	60	2,200	Widowed	700	Yes
5	58	2,000	Married	?	No
6	44	?	?	?	No
7	22	1,200	Single	?	No
8	26	1,500	Married	350	No
9	34	?	Single	?	Yes
10	50	2,100	Divorced	?	No

Exhibit 4.14 Dealing with Missing Values

- **Delete**. This is the most straightforward option and consists of deleting observations or variables with lots of missing values. This of course assumes that information is missing at random and has no meaningful interpretation and/or relationship to the target.
- **Keep**. Missing values can be meaningful. For example, if a customer did not disclose his or her income because he or she is currently unemployed, this fact may have a relationship with default and needs to be considered as a separate category.

As a practical way of working, you can first start by statistically testing whether missing information is related to the target variable (using, e.g., a chi-square test, discussed later). If yes, then you can adopt the keep strategy and make a special category for it. If not, depending on the number of observations available, you can decide to either delete or impute.

In Base SAS, PROC STANDARD can be used to replace missing values. Consider the following statement:

```
PROC STANDARD DATA =data.mysample REPLACE OUT=data.mysamplenomissing;
RUN;
```

This will replace all missing values of the continuous variables with the mean. The missing values for the categorical variables will remain untreated. Note that PROC STANDARD cannot be used to impute with the median or other values. In this case, this should be manually programmed using a set of data manipulation statements.

In SAS Enterprise Miner, missing values can be treated using the Impute node from the Modify tab (see Exhibit 4.15). This node provides separate treatment options for the class and interval variables. In our example, the missing categorical variables will be replaced with the mode (indicated by Default Input Method=Count in the Class Variables section), and missing interval variables will be replaced by the median (indicated by Default Input Method=Median in the Interval Variables section).

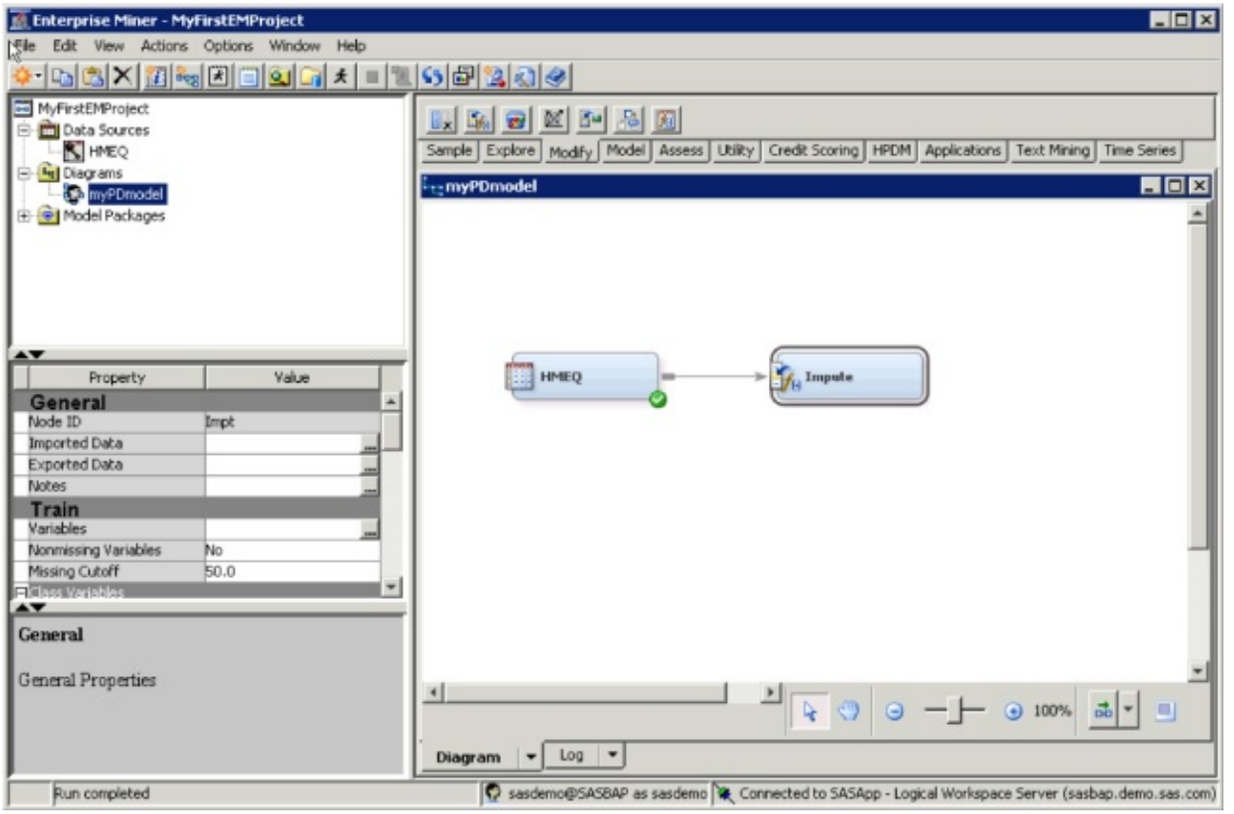


Exhibit 4.15 The Impute Node in SAS Enterprise Miner

OUTLIER DETECTION AND TREATMENT

Outliers are extreme observations that are very dissimilar to the rest of the population. Actually, two types of outliers can be considered:

1. Valid observations (e.g., salary of boss is $1 million)
2. Invalid observations (e.g., age is 300 years)

Both are univariate outliers in the sense that they are outlying on one dimension. However,

outliers can be hidden in unidimensional views of the data. Multivariate outliers are observations that are outlying in multiple dimensions. Exhibit 4.16 gives an example of two outlying observations considering the dimensions of both income and age.

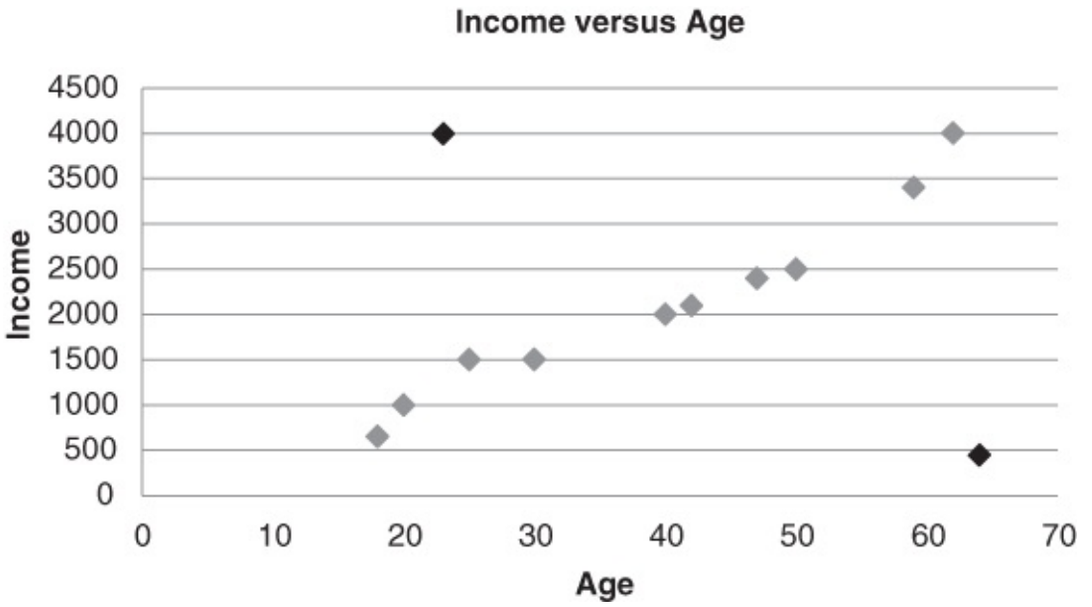


Exhibit 4.16 Multivariate Outliers

Two important steps in dealing with outliers are detection and treatment. A first obvious check for outliers is to calculate the minimum and maximum values for each of the data elements. Various graphical tools can be used to detect outliers. Histograms are a first example. Exhibit 4.17 presents an example of a distribution for age whereby the circled areas clearly represent outliers.

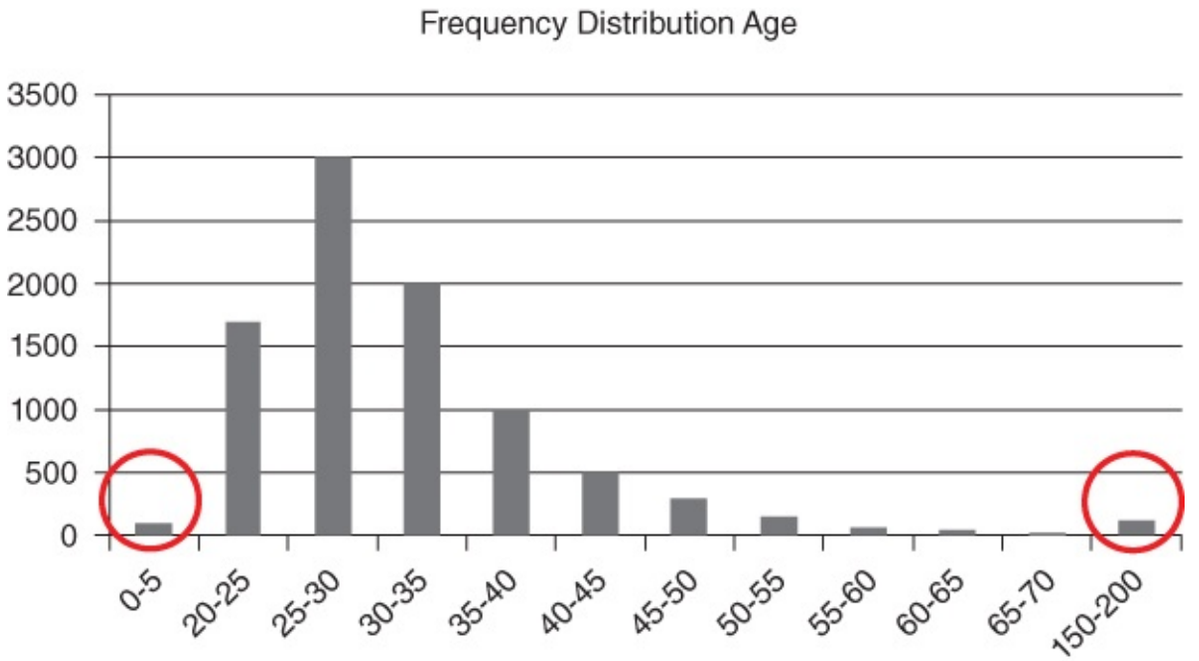


Exhibit 4.17 Histogram for Outlier Detection

Box plots (see Chapter 3, Exploratory Data Analysis) can also be used to detect outliers.

Another way is to calculate z-scores, measuring how many standard deviations an observation lies away from the mean, as follows:

$$z_i = \frac{x_i - \overline{x_i}}{s},$$

whereby $\overline{x_i}$ represents the average of the variable and *s* its standard deviation. An example is given in Exhibit 4.18. Note that by definition the z-scores will have a mean of zero and standard deviation of one.

ID	Age	z-Score
1	30	(30 – 40)/10 = –1
2	50	(50 – 40)/10 = +1
3	10	(10 – 40)/10 = –3
4	40	(40 – 40)/10 = 0
5	60	(60 – 40)/10 = +2
6	80	(80 – 40)/10 = +4
. . .	. . .	. . .
	$\overline{x_i}$ = **40** ***s*** **= 10**	$\overline{x_i}$ = **0** ***s*** **= 1**

Exhibit 4.18 z-Scores for Outlier Detection

A practical rule of thumb then defines outliers when the absolute value of the z-score |z| is bigger than 3. Note that the z-score method relies on the normal distribution.

The preceding methods all focus on univariate outliers. Multivariate outliers can be detected by fitting regression lines and inspecting the observations with large errors (e.g., using a residual plot). Alternative methods are clustering or calculating the Mahalanobis distance. Note, however, that although potentially useful, multivariate outlier detection is typically not considered in many modeling exercises due to the typically marginal impact on model performance.

Some analytical techniques (e.g., decision trees, neural networks, support vector machines [SVMs]) are fairly robust with respect to outliers. Others (e.g., linear/logistic regression) are more sensitive to them. Various schemes exist to deal with outliers. It highly depends upon whether the outlier represents a valid or an invalid observation. For invalid observations (e.g., age is 300 years), you could treat the outlier as a missing value using any of the schemes discussed in the previous section. For valid observations (e.g., income is $1 million), other schemes are needed. A popular scheme is truncation/capping/winsorizing. You hereby impose both a lower limit and an upper limit on a variable and any values below/above are brought back to these limits. The limits can be calculated using the z-scores (see Exhibit 4.19) or the interquartile range (IQR) (which is more robust than the z-scores), as follows (see Van Gestel

and Baesens 2009):

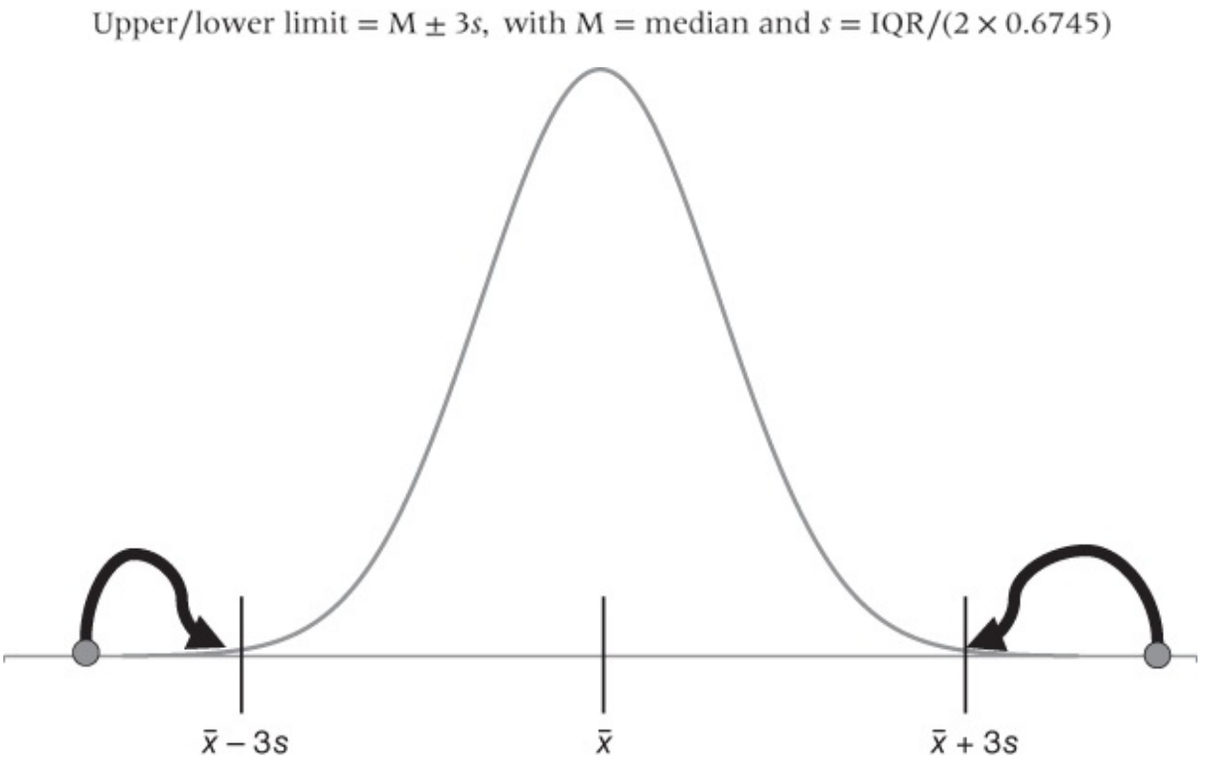


Exhibit 4.19 Using the z-Scores for Truncation

A sigmoid transformation ranging between 0 and 1 can also be used for capping, as follows:

$$f(x) = \frac{1}{1 + e^{-x}}$$

Also, expert-based limits based on business knowledge and/or experience can be imposed.

An important remark concerning outliers is the fact that not all invalid values are outlying, and as such may go unnoted if not explicitly looked into. For instance, a clear issue exists when observing customers with values *gender = male* and *pregnant = yes*. Which value is invalid, either the value for gender or the value for pregnant, cannot be determined, but neither value is outlying and therefore such a conflict will not be noted by the analyst unless some explicit precautions are taken. In order to detect particular invalid combinations, you may construct a set of rules that are formulated based on expert knowledge and experience, which is applied to the data to check and alert for issues. In this particular context a network representation of the variables may be of use to construct the rule set and reason upon relationships that exist between the different variables, with links representing constraints that apply to the combination of variable values and resulting in rules added to the rule set.

In Base SAS, we can filter outliers based on the z-scores by using PROC STANDARD as follows:

```
PROC STANDARD DATA =data.mysample MEAN=0 STD=1 OUT=data.zscores;
VAR clage clno debtinc delinq derog loan mortdue ninq value yoj;
RUN;
```

The preceding statement will calculate the z-scores for each of the continuous variables. We can then filter out the outliers by requiring that all z-scores should be less than 3 in absolute value, as follows:

```
DATA data.filteredsample;
SET data.zscores;
WHERE ABS(clage) < 3 & ABS(clno) <3  & ABS(debtinc) < 3 & ABS(delinq)<3
  & ABS(derog) < 3
& ABS(loan) < 3 & ABS(mortdue)< 3 & ABS(ninq)<3 & ABS(value) < 3 & ABS(yoj)
 < 3;
RUN;
```

The resulting data set, filteredsample, will have 890 observations; in other words, 110 observations have been removed.

In SAS Enterprise Miner, the Filter node from the Sample tab can be used to filter or remove the outliers (see Exhibit 4.20). In our example, we will remove observations with class values that occur less than 1 percent for class variables with fewer than 25 values. We will also remove observations with interval variables that are more than 3 standard deviations away from the mean (the latter can be specified in the Tuning Parameters property).

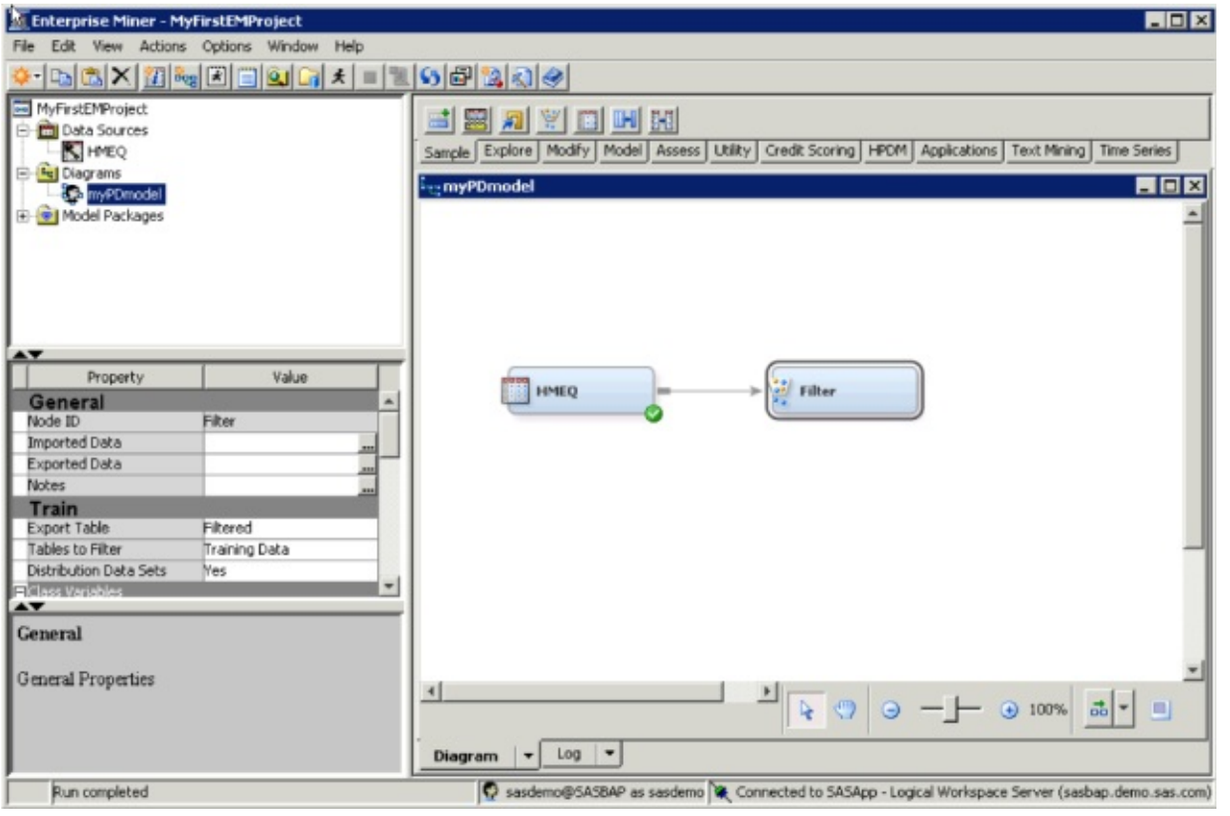


Exhibit 4.20 The Filter Node in SAS Enterprise Miner

The Replacement node from the Modify tab can be used for outlier truncation (see Exhibit 4.21). In the example, interval variables will be truncated to the mean ± 3 standard deviations (specified by the Default Limits method and Cutoff Values properties in the Interval Variables section). For class variables, a replacement editor is provided where the user can manually enter the new values for each of the class variables.

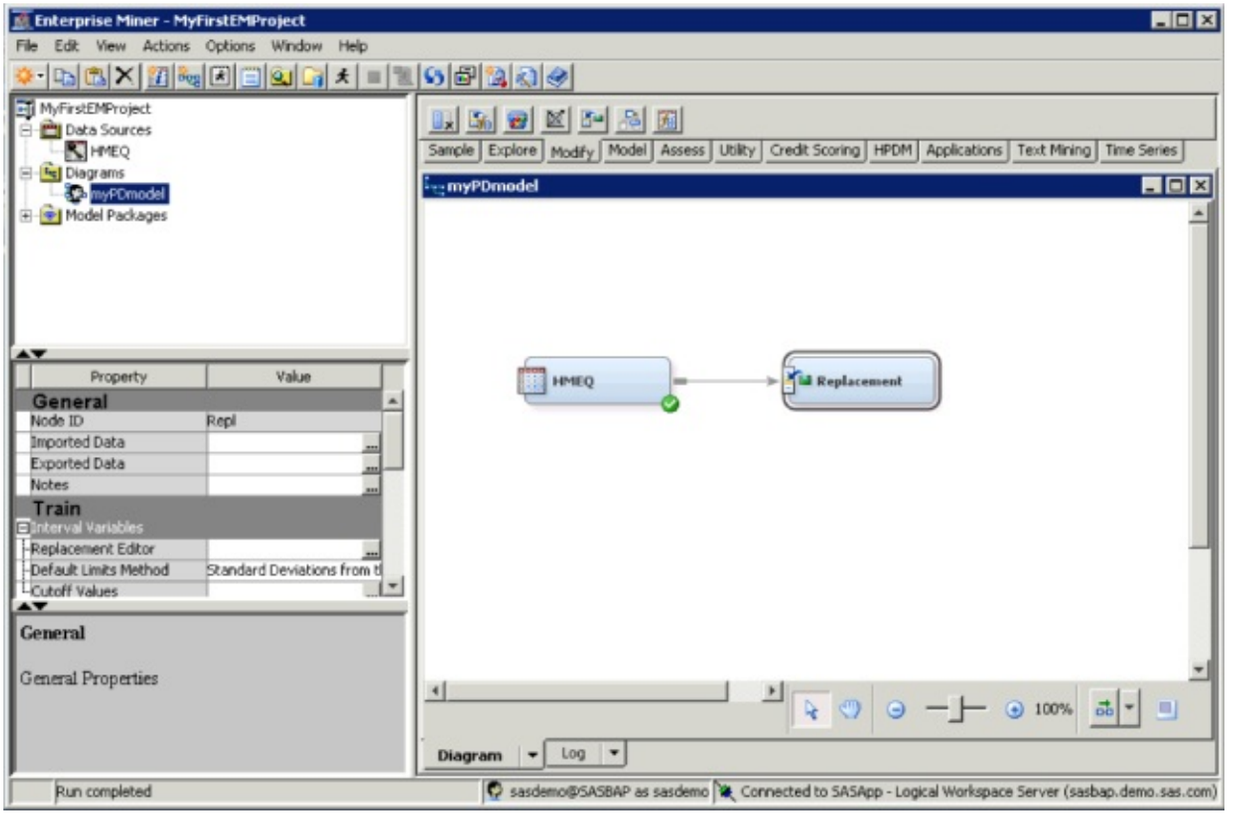


Exhibit 4.21 The Replacement Node in SAS Enterprise Miner

STANDARDIZING DATA

Standardizing data is a data preprocessing activity targeted at scaling variables to a similar range. Consider, for example, two variables gender (coded as 0/1) and income (ranging between zero and $1 million). When building logistic regression models using both information elements, the coefficient for income might become very small. Hence, it could make sense to bring them back to a similar scale. The following standardization procedures could be adopted:

- Min/max standardization
  - $x_{new} = \frac{x_{old} - \min(x_{old})}{\max(x_{old}) - \min(x_{old})}(newmax - newmin) + newmin$, whereby *newmax* and *newmin* are the newly imposed maximum and minimum (e.g., 1 and 0).
- z-score standardization
  - Calculate the z-scores (see the previous section).
- Decimal scaling
  - Dividing by a power of 10, as follows: $x_{new} = \frac{x_{old}}{10^n}$, with *n* the number of digits of the maximum absolute value.

Again note that standardization is especially useful for regression-based approaches but is not needed for some approaches like decision trees.

As we already illustrated in the section on outliers, the z-scores can be calculated using PROC STANDARD in Base SAS.

In SAS Enterprise Miner, the Transform Variables node from the Modify tab can be used to standardize or transform the data. In our example, all interval variables will be standardized to zero mean and unit standard deviation by calculating the z-scores (specified by the Interval Inputs=Standardize property in the Default Methods section).

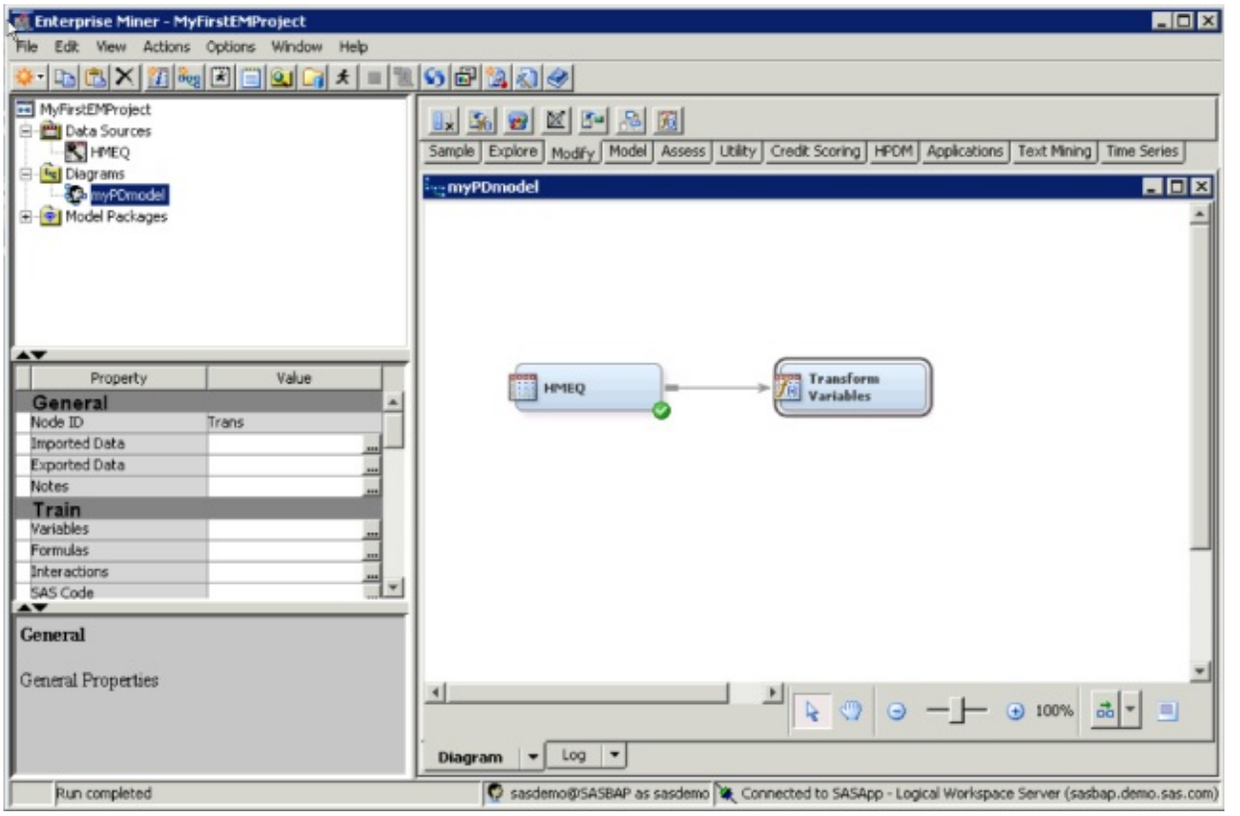


Exhibit 4.22 The Transform Variables Node in SAS Enterprise Miner

CATEGORIZATION

Categorization (also referred to as coarse classification, classing, grouping, or binning) can be done for various reasons. For categorical variables, it is needed to reduce the number of categories. Consider, for example, the variable *industry sector* having 50 different values. When this variable is put into a (logistic) regression model, you would need 49 dummy variables (50 – 1 because of the collinearity), which would necessitate the estimation of 49 parameters for only one variable. With categorization, you create categories of values such that fewer parameters will have to be estimated and a more robust model is obtained.

For continuous variables, categorization may also be beneficial. Consider, for example, the age variable and the observed default rate as depicted in Exhibit 4.23. Clearly, there is a nonmonotonous relationship between risk of default and age. If a nonlinear model (e.g., neural network, support vector machine) were used, then the nonlinearity could be perfectly modeled. However, if a regression model were used (which is typically more common because of its

interpretability), then since it can only fit on a line, it would miss out on the nonmonotonicity. By categorizing the variable into ranges, part of the nonmonotonicity can be taken into account in the regression. Hence, categorization of continuous variables can be useful to model nonlinear effects in linear models.

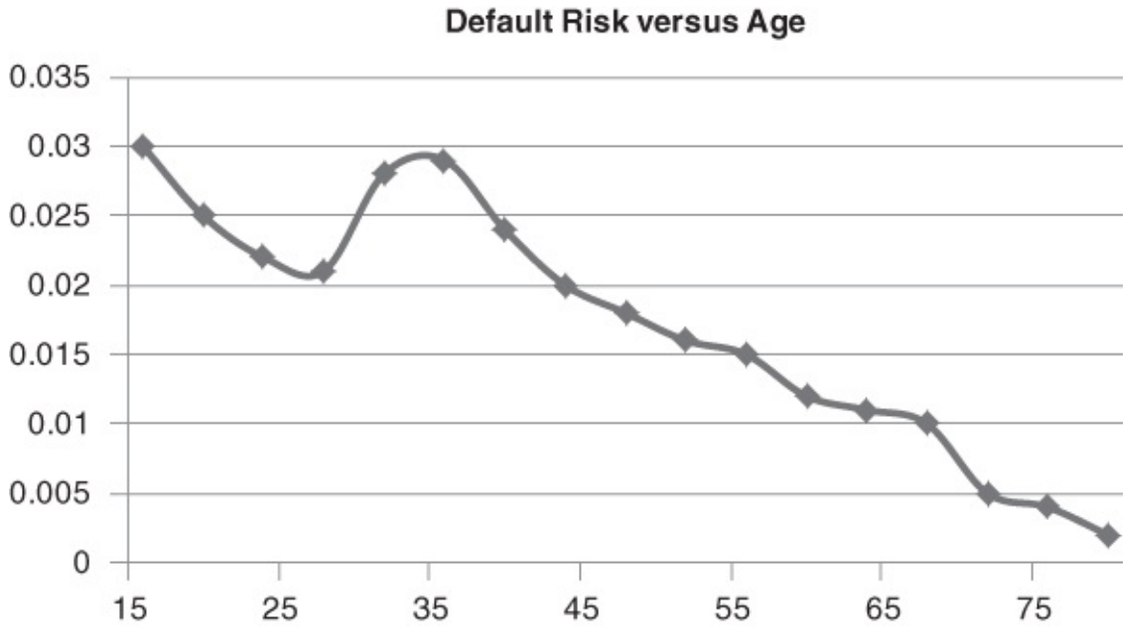


Exhibit 4.23 Default Risk versus Age

Various methods can be used to do categorization. Two very basic methods are equal-interval binning and equal-frequency binning. Consider, for example, the income values 1,000, 1,200, 1,300, 2,000, 1,800, 1,400. Equal-interval binning would create two bins with the same range (Bin 1: 1,000, 1,500, and Bin 2: 1,500, 2,000), whereas equal-frequency binning would create two bins with the same number of observations (Bin 1: 1,000, 1,200, 1,300, and Bin 2: 1,400, 1,800, 2,000). However, both methods are quite basic and do not take into account the target variable or default risk. We provide further examples in the context of rating class formation in the chapters on exploratory data analysis and discrete time hazard models.

Most analytics software tools have built-in facilities to perform categorization. A very handy and simple approach (available in Microsoft Excel) is to use pivot tables. Consider the example shown in Exhibit 4.24.

Customer ID	Age	Purpose of Loan	. . .	Default
C1	44	Travel		No
C2	20	House		No
C3	58	Car		Yes
C4	26	Travel		No
C5	30	Study		Yes
C6	32	Furniture		No
C7	48	House		Yes
C8	60	Travel		No
. . .	. . .	. . .		

Exhibit 4.24 Coarse Classifying the Purpose of Loan Variable

We can then construct a pivot table and calculate the odds as shown in Exhibit 4.25.

	Travel	Car	Furniture	House	Study	. . .
Good	1,000	2,000	3,000	100	5,000	
Bad	500	100	200	80	800	
Odds	2	20	15	1.25	6.25	

Exhibit 4.25 Pivot Table for Coarse Classifying the Purpose of Loan Variable

We can then categorize the values based on similar odds. One example would be category 1 (travel, house), category 2 (study), and category 3 (car, furniture).

Chi-square analysis is a more sophisticated way to perform categorization. Consider the example depicted in Exhibit 4.26 for coarse classifying a residential status variable.[2]

		Rent	Rent	With		No	
Attribute	Owner	Unfurnished	Furnished	Parents	Other	Answer	Total
Good	6,000	1,600	350	950	90	10	9,000
Bad	300	400	140	100	50	10	1,000
Good/Bad Odds	20:1	4:1	2.5:1	9.5:1	1.8:1	1:1	9:1

Exhibit 4.26 Coarse Classifying the Residential Status Variable

Suppose we want three categories and consider the following options:

- Option 1: owners; renters; others
- Option 2: owners; with parents; others

Both options can now be investigated using chi-square analysis. The purpose is to compare the

empirically observed with the independence frequencies. For Option 1, the empirically observed frequencies are depicted in Exhibit 4.27.

Attribute	Owners	Renters	Others	Total
Good	6,000	1,950	1,050	9,000
Bad	300	540	160	1,000
Total	6,300	2,490	1,210	10,000

Exhibit 4.27 Empirical Frequencies Option 1 for Coarse Classifying Residential Status

The independence frequencies can be calculated as follows. The number of good owners given that the odds are the same as in the whole population is $6{,}300/10{,}000 * 9{,}000/10{,}000 * 10{,}000 = 5{,}670$. We then obtain Exhibit 4.28.

Attribute	Owners	Renters	Others	Total
Good	5,670	2,241	1,089	9,000
Bad	630	249	121	1,000
Total	6,300	2,490	1,210	10,000

Exhibit 4.28 Independence Frequencies Option 1 for Coarse Classifying Residential Status

The more the numbers in both tables differ, the less independence and hence the more dependence and a better coarse classification. Formally, we can calculate the chi-square distance as follows:

$$\chi^2 = \frac{(6{,}000 - 5{,}670)^2}{5{,}670} + \frac{(300 - 630)^2}{630} + \frac{(1{,}950 - 2{,}241)^2}{2{,}241} + \frac{(540 - 249)^2}{249} + \frac{(1{,}050 - 1{,}089)^2}{1{,}089} + \frac{(160 - 121)^2}{121} = 583$$

Likewise, for option 2, the calculation becomes:

$$\chi^2 = \frac{(6{,}000 - 5{,}670)^2}{5{,}670} + \frac{(300 - 630)^2}{630} + \frac{(950 - 945)^2}{945} + \frac{(100 - 105)^2}{105} + \frac{(2{,}050 - 2{,}385)^2}{2{,}385} + \frac{(600 - 265)^2}{265} = 662$$

So, based on the chi-square values, option 2 is the better categorization. Note that formally you need to compare the value with a chi-square distribution with $k - 1$ degrees of freedom, with k the number of values of the characteristic.

In Base SAS, we can start by first defining the data set as follows:

```
DATA residence;
   INPUT default$ resstatus$ count;
   DATALINES;
```

```
good owner 6000
good rentunf 1600
good rentfurn 350
good withpar 950
good other 90
good noanswer 10
bad owner 300
bad rentunf 400
bad rentfurn 140
bad withpar 100
bad other 50
bad noanswer 10
;
```

We can now also create the data sets for both categorization options:

```
DATA coarse1;
   INPUT default$ resstatus$ count;
   DATALINES;
good owner 6000
good renter 1950
good other 1050
bad owner 300
bad renter 540
bad other 160
;
DATA coarse2;
   INPUT default$ resstatus$ count;
   DATALINES;
good owner 6000
good withpar 950
good other 2050
bad owner 300
bad withpar 100
bad other 600
;
```

We can now run PROC FREQ on both options as follows:

```
PROC FREQ DATA=coarse1;
   WEIGHT count;
   TABLES default*resstatus / CHISQ;
RUN;
```

This will give the output depicted in Exhibit 4.29.

The FREQ Procedure				
Table of default by resstatus				
default	resstatus			
	other	**owner**	**renter**	**Total**
Bad	160	300	540	1000
	1.60	3.00	5.40	10.00
	16.00	30.00	54.00	
	13.22	4.76	21.69	
Good	1050	6000	1950	9000
	10.50	60.00	19.50	90.00
	11.67	66.67	21.67	
	86.78	95.24	78.31	
Total	1210	6300	2490	10000
	12.10	63.00	24.90	100.00

The FREQ Procedure			
Statistics for Table of default by resstatus			
Statistic	DF	**Value**	Prob
Chi-Square	2	583.9019	<.0001
Likelihood Ratio Chi-Square	2	540.0817	<.0001
Mantel-Haenszel Chi-Square	1	199.5185	<.0001
Phi Coefficient		0.2416	
Contingency Coefficient		0.2349	
Cramer's V		0.2416	

Exhibit 4.29 Output for Categorization Option 1

We can now also examine Option 2 as follows:

```
PROC FREQ DATA=coarse2;
   WEIGHT count;
   TABLES default*resstatus / CHISQ;
RUN;
```

This will give the output depicted in Exhibit 4.30.

The FREQ Procedure				
Table of default by resstatus				
default	resstatus			
	other	owner	withpar	**Total**
Bad	600	300	100	1000
	6.00	3.00	1.00	10.00
	60.00	30.00	10.00	
	22.64	4.76	9.52	
Good	2050	6000	950	9000
	20.50	60.00	9.50	90.00
	22.78	66.67	10.56	
	77.36	95.24	90.48	
Total	2650	6300	1050	10000
	26.50	63.00	10.50	100.00

The FREQ Procedure			
Statistics for Table of default by resstatus			
Statistic	**DF**	**Value**	**Prob**
Chi-Square	2	662.8731	<.0001
Likelihood Ratio Chi-Square	2	594.0167	<.0001
Mantel-Haenszel Chi-Square	1	372.9141	<.0001
Phi Coefficient		0.2575	
Contingency Coefficient		0.2493	
Cramer's V		0.2575	

Exhibit 4.30 Output for Categorization Option 2

WEIGHTS OF EVIDENCE CODING

Categorization reduces the number of categories for categorical variables. For continuous variables, categorization will introduce new variables. Consider a regression model with age characteristics (four categories, so three parameters) and purpose characteristics (five categories, so four parameters). The model then looks like the following:

$$D = \beta_0 + \beta_1 \text{Age}_1 + \beta_2 \text{Age}_2 + \beta_3 \text{Age}_3 + \beta_4 \text{Purp}_1 + \beta_5 \text{Purp}_2 + \beta_6 \text{Purp}_3 + \beta_7 \text{Purp}_4$$

with $D = 1$ for defaulters, and 0 otherwise.

Despite having only two characteristics, the model still needs eight parameters to be estimated. It would be handy to have a monotonic transformation $f(.)$ such that our model could be rewritten as follows:

$$D = \beta_0 + \beta_1 f(\text{Age}_1, \text{Age}_2, \text{Age}_3) + \beta_2 f(\text{Purp}_1, \text{Purp}_2, \text{Purp}_3, \text{Purp}_4)$$

The transformation should have a monotonically increasing or decreasing relationship with D. Weights of evidence (WOE) coding is one example of a transformation that can be used for this purpose. This is illustrated in Exhibit 4.31.

Age	Count	Distribution of Count	Goods	Distribution of Goods	Bads	Distribution of Bads	WOE
Missing	50	2.50%	42	2.33%	8	4.12%	–57.28%
18–22	200	10.00%	152	8.42%	48	24.74%	–107.83%
23–26	300	15.00%	246	13.62%	54	27.84%	–71.47%
27–29	450	22.50%	405	22.43%	45	23.20%	–3.38%
30–35	500	25.00%	475	26.30%	25	12.89%	71.34%
36–43	350	17.50%	339	18.77%	11	5.67%	119.71%
44+	150	7.50%	147	8.14%	3	1.55%	166.08%
	2,000		1,806		194		

Exhibit 4.31 Calculating Weights of Evidence (WOE)

The WOE is calculated as: ln(Dist Good/Dist Bad). Because of the logarithmic transformation, a positive WOE means Dist Good > Dist Bad; a negative WOE means Dist Good < Dist Bad. WOE coding thus implements a transformation that is monotonically related to the target variable.

The model can then be reformulated as follows:

$$D = \beta_0 + \beta_1 \text{WOE}_{\text{age}} + \beta_2 \text{WOE}_{\text{purpose}}$$

This gives a more concise model than the model we started this section with. However, note that the interpretability of the model becomes somewhat less straightforward when WOE variables are being used.

The technique can be programmed separately using a set of data manipulation statements, or alternatively PROC HPBIN can be used (we will illustrate this in the chapter on time-discrete hazard models).

The best and easiest way to perform categorization and weights of evidence coding in SAS Enterprise Miner is by using the Interactive Grouping node from the Credit Scoring tab (Exhibit 4.32). The latter is an extension to Enterprise Miner offering four nodes tailored to building credit scorecards. The Interactive Grouping node has various properties that can be

inspected in the properties panel. As with many nodes in Enterprise Miner, the default settings usually work well. After the node has run, you can interactively inspect the results by clicking the button next to Interactive Grouping in the properties panel. This will give the output shown in Exhibit 4.33. Here you can see the variables listed, their measurement level, and their calculated role, which we will discuss in the next section. If you now click the Groupings tab in the upper left corner, you will obtain the output of Exhibit 4.34. In the upper left corner, you can see the variable distribution and how it has been categorized. The upper right corner depicts the corresponding weights of evidence plot. The lower left corner shows the variable values. It is possible to adjust the grouping by selecting multiple values using SHIFT followed by right-click to decide which group the selected values should be assigned to. Obviously, all the plots will then change as well. The lower right corner shows some variable statistics, which will be discussed in the next section.

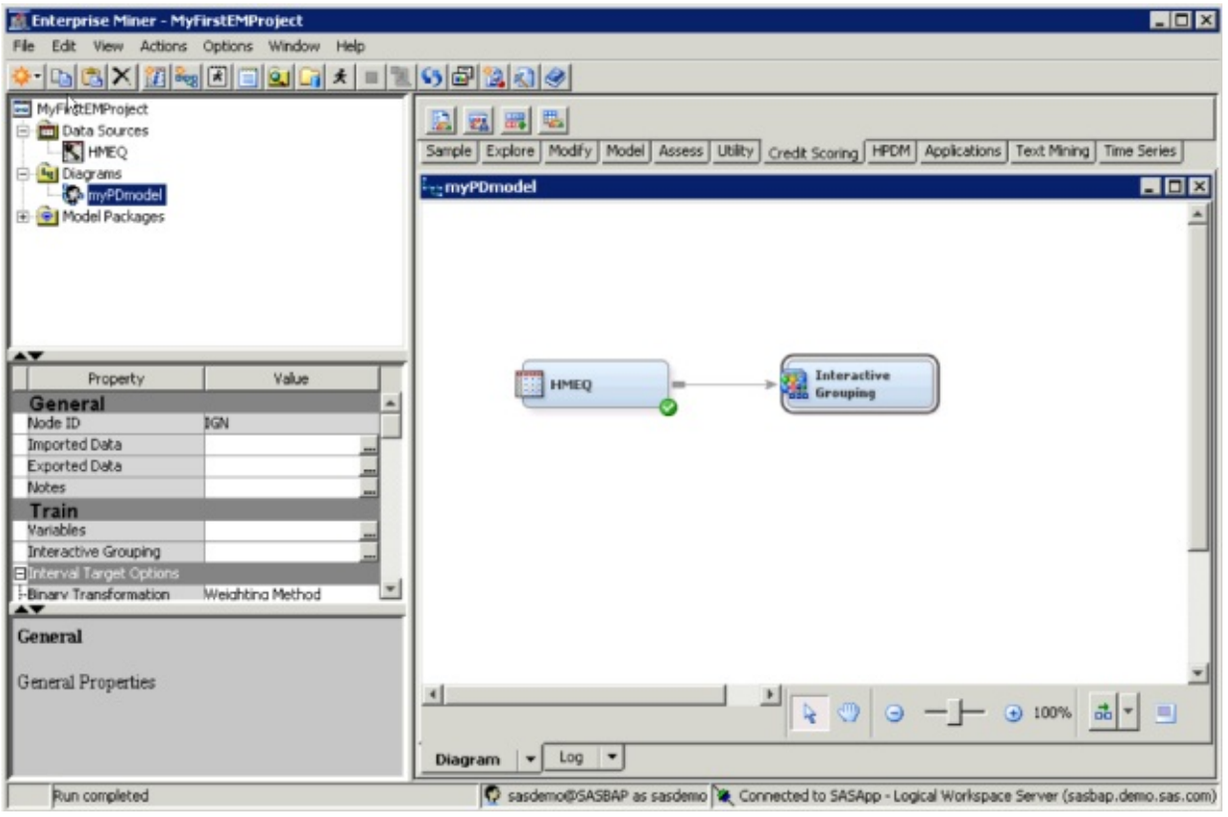
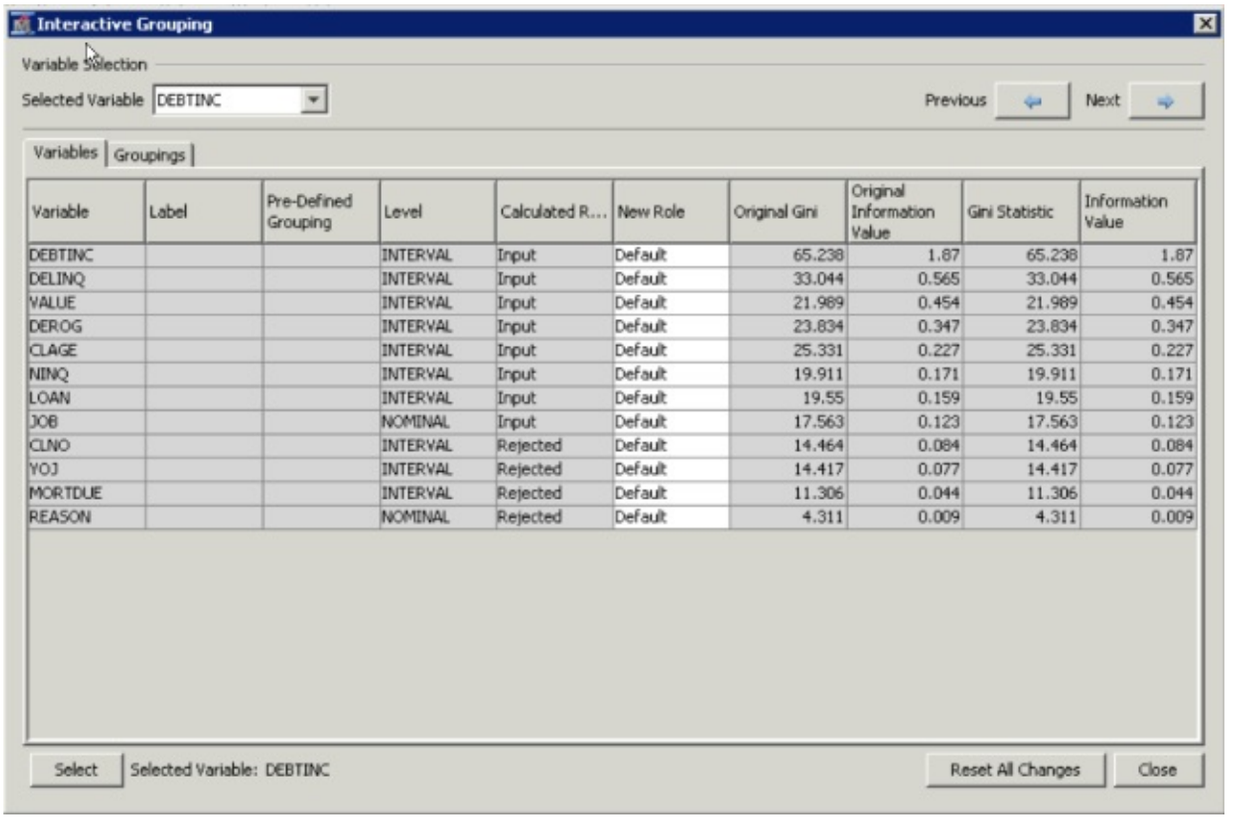
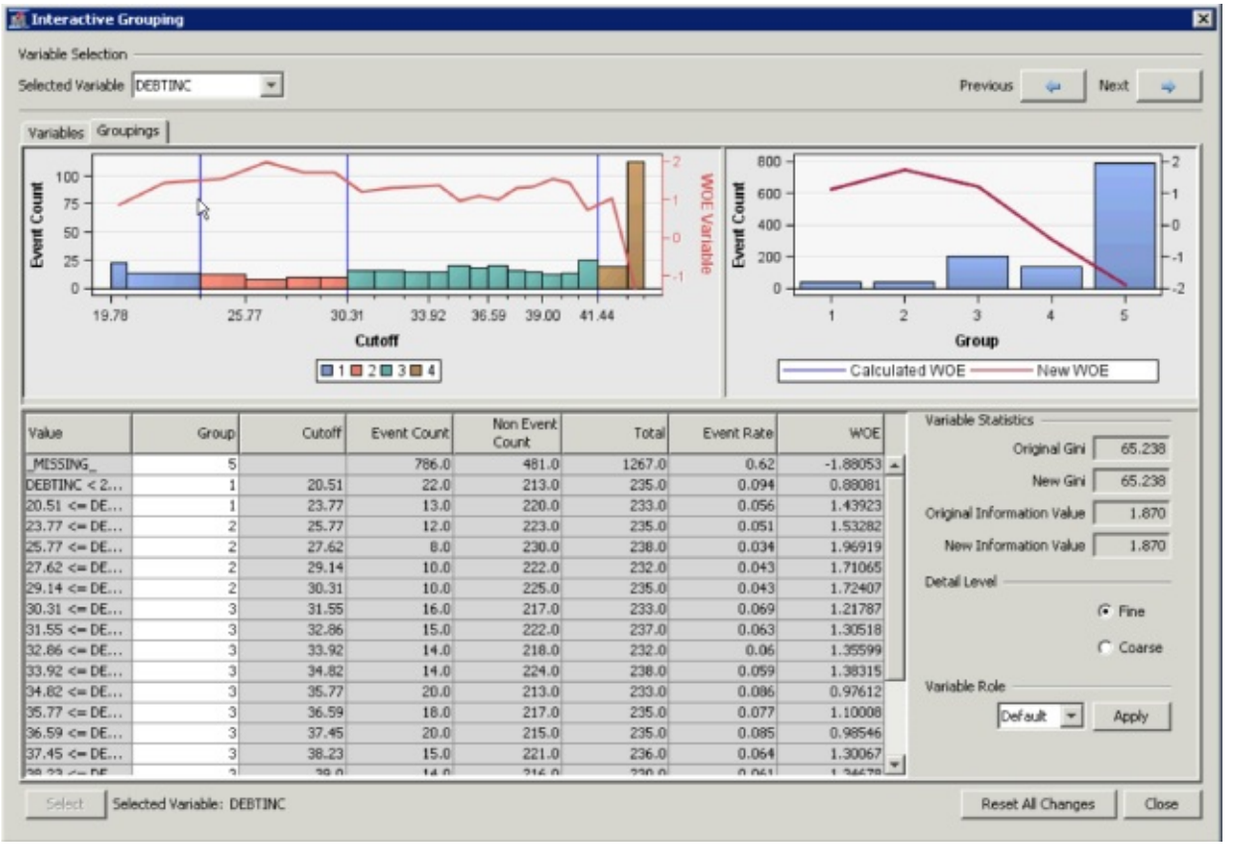


Exhibit 4.32 The Interactive Grouping Node in SAS Enterprise Miner



Variable	Label	Pre-Defined Grouping	Level	Calculated R...	New Role	Original Gini	Original Information Value	Gini Statistic	Information Value
DEBTINC			INTERVAL	Input	Default	65.238	1.87	65.238	1.87
DELINQ			INTERVAL	Input	Default	33.044	0.565	33.044	0.565
VALUE			INTERVAL	Input	Default	21.989	0.454	21.989	0.454
DEROG			INTERVAL	Input	Default	23.834	0.347	23.834	0.347
CLAGE			INTERVAL	Input	Default	25.331	0.227	25.331	0.227
NINQ			INTERVAL	Input	Default	19.911	0.171	19.911	0.171
LOAN			INTERVAL	Input	Default	19.55	0.159	19.55	0.159
JOB			NOMINAL	Input	Default	17.563	0.123	17.563	0.123
CLNO			INTERVAL	Rejected	Default	14.464	0.084	14.464	0.084
YOJ			INTERVAL	Rejected	Default	14.417	0.077	14.417	0.077
MORTDUE			INTERVAL	Rejected	Default	11.306	0.044	11.306	0.044
REASON			NOMINAL	Rejected	Default	4.311	0.009	4.311	0.009

Exhibit 4.33 Results of the Interactive Grouping Node in SAS Enterprise Miner



Value	Group	Cutoff	Event Count	Non Event Count	Total	Event Rate	WOE
MISSING	5		786.0	481.0	1267.0	0.62	-1.88053
DEBTINC < 2...	1	20.51	22.0	213.0	235.0	0.094	0.88081
20.51 <= DE...	1	23.77	13.0	220.0	233.0	0.056	1.43923
23.77 <= DE...	2	25.77	12.0	223.0	235.0	0.051	1.53282
25.77 <= DE...	2	27.62	8.0	230.0	238.0	0.034	1.96919
27.62 <= DE...	2	29.14	10.0	222.0	232.0	0.043	1.71065
29.14 <= DE...	2	30.31	10.0	225.0	235.0	0.043	1.72407
30.31 <= DE...	3	31.55	16.0	217.0	233.0	0.069	1.21787
31.55 <= DE...	3	32.86	15.0	222.0	237.0	0.063	1.30518
32.86 <= DE...	3	33.92	14.0	218.0	232.0	0.06	1.35599
33.92 <= DE...	3	34.82	14.0	224.0	238.0	0.059	1.38315
34.82 <= DE...	3	35.77	20.0	213.0	233.0	0.086	0.97612
35.77 <= DE...	3	36.59	18.0	217.0	235.0	0.077	1.10008
36.59 <= DE...	3	37.45	20.0	215.0	235.0	0.085	0.98546
37.45 <= DE...	3	38.23	15.0	221.0	236.0	0.064	1.30067

Exhibit 4.34 Results of the Interactive Grouping Node and Groupings Tab in SAS Enterprise Miner

VARIABLE SELECTION

Many analytical modeling exercises start with a broad selection of variables, of which typically only a few actually contribute to the prediction of the target variable. Common application or behavioral scorecards often have between 10 and 15 variables. The key question is how to find these variables. Filters are a very handy variable selection mechanism. They work by measuring univariate correlations between each variable and the target. As such, they allow for a quick screening of which variables should be retained for further analysis. Various filter measures have been suggested in the literature. You can categorize them as depicted in Exhibit 4.35.

	Continuous Target	**Categorical Target**
	(e.g., LGD, EAD)	**(e.g., Default)**
Continuous variable	Pearson correlation	Fisher score
Categorical variable	Fisher score/ANOVA	Information value Cramer's V Gain/entropy

Exhibit 4.35 Filters for Variable Selection

The Pearson correlation ρ_P is calculated as follows:

$$\rho_P = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

It measures a linear dependency between two variables and always varies between –1 and +1. To apply it as a filter, you could select all variables for which the Pearson correlation is significantly different from 0 (according to the *p*-value), or, for example, the ones where $|\rho_P| > 0.50$.

The Fisher score can be calculated as follows:

$$\frac{|\bar{x}_{ND} - \bar{x}_D|}{\sqrt{s_{ND}^2 + s_D^2}},$$

whereby $\bar{x}_{ND}$ represents the average value of the variable for the nondefaulters (and $\bar{x}_D$ for the defaulters) and s_{ND}^2 (s_D^2) the corresponding variances. High values of the Fisher score indicate a predictive variable. To apply it as a filter, you could keep the top 10 percent. Note that the Fisher score may generalize to a well-known analysis of variance (ANOVA) when a variable has multiple categories.

The information value (IV) filter is based on weights of evidence and is calculated as follows:

$$IV = \sum_{i=1}^{k} (Dist\ Good_i - Dist\ Bad_i) * WOE_i$$

whereby *k* represents the number of categories of the variable. For the example presented in Exhibit 4.31 the calculation becomes as shown in Exhibit 4.36.

Age	Count	Distribution of Count	Goods	Distribution of Goods	Bads	Distribution of Bads	WOE	IV
Missing	50	2.50%	42	2.33%	8	4.12%	–57.28%	0.0103
18–22	200	10.00%	152	8.42%	48	24.74%	–107.83%	0.1760
23–26	300	15.00%	246	13.62%	54	27.84%	–71.47%	0.1016
27–29	450	22.50%	405	22.43%	45	23.20%	–3.38%	0.0003
30–35	500	25.00%	475	26.30%	25	12.89%	71.34%	0.0957
36–43	350	17.50%	339	18.77%	11	5.67%	119.71%	0.1568
44+	150	7.50%	147	8.14%	3	1.55%	166.08%	0.1095
Information Value								**0.6502**

Exhibit 4.36 Calculating the Information Value Filter Measure

The following rules of thumb apply for the information value:

- <0.02: unpredictive
- 0.02–0.1: weakly predictive
- 0.1–0.3: moderately predictive
- +0.3: strongly predictive

Note that the information value assumes that the variable has been categorized. It can also be used to adjust or steer the categorization so as to optimize the IV. Many software tools will provide interactive support to do this, whereby the modeler can adjust the categories and gauge the impact on the IV. To apply it as a filter, you can calculate the information value of all (categorical) variables and keep only those for which the IV > 0.1, or the top 10 percent.

Another filter measure based on chi-square analysis is Cramer's V. Consider the contingency table depicted in Exhibit 4.37 for employed/unemployed versus good/bad.

	Good	Bad	Total
Employed	500	100	600
Unemployed	300	100	400
Total	800	200	1,000

Exhibit 4.37 Contingency Table for Employment Status versus Good/Bad Customer

Similar to the example discussed in the section on categorization, the chi-square value for

independence can then be calculated as follows:

$$\chi^2 = \frac{(500-480)^2}{480} + \frac{(100-120)^2}{120} + \frac{(300-320)^2}{320} + \frac{(100-80)^2}{80} = 10.41$$

This follows a chi-square distribution with $k-1$ degrees of freedom, with k being the number of classes of the characteristic. The Cramer's V measure can then be calculated as follows:

$$\text{Cramer's V} = \sqrt{\frac{\chi^2}{n}} = 0.10$$

with n the number of observations in the data set. Cramer's V is always bounded between 0 and 1, and higher values indicate better predictive power. As a rule of thumb, a cutoff of 0.1 is commonly adopted. You can then again select all variables where Cramer's V is bigger than 0.1, or consider the top 10 percent. Note that the information value and Cramer's V typically consider the same characteristics as most important.

Filters are very handy, as they permit a reduction of the number of variables of the data set early in the analysis in a quick way. Their main drawback is that they work univariately and typically do not consider correlation between the variables individually. Hence, a follow-up variable selection step during the modeling phase will be necessary to further refine the set of variables.

It is worth mentioning that other criteria may play a role in selecting variables, such as regulatory compliance and privacy issues. Note that different regulations may apply in different geographical regions and should be checked. Also, operational issues could be considered. For example, trend variables could be very predictive but may require too much time to be computed in a real-time, online credit scoring environment.

In Base SAS, the Pearson correlation can be calculated using PROC CORR, whereas the Cramer's V is available in PROC FREQ (see Exhibit 4.29). The information value can be computed by PROC HPBIN.

In SAS Enterprise Miner, the Pearson correlation and Cramer's V can be obtained from the StatExplore node (see Exhibit 4.11). The information value can be obtained from the Interactive Grouping node. In the right-hand column of Exhibit 4.33, you can see the information value for each of the variables. If the information value is bigger than 0.1, the variable's calculated role is set to Input; otherwise it is set to Rejected. The cutoff value for the information value can be set in the properties of the Interactive Grouping node. The original information value is the information value obtained by SAS Enterprise Miner using its default settings. The information value in the last column is the information value obtained when the user has changed the groupings. The two values are identical when the user does not adjust the groupings (see also Exhibit 4.34).

SEGMENTATION

Sometimes the data is segmented before the credit risk modeling starts. A first reason for this could be strategic. Banks might want to adopt special strategies for specific segments of customers. Segmentation could also be motivated from an operational viewpoint. Some new customers must have separate models because the characteristics in the standard model do not make sense operationally for them. Segmentation could also be needed to take into account significant variable interactions. If one variable strongly interacts with a number of others, it might be sensible to segment according to this variable.

The segmentation can be done using the experience and knowledge of a business expert, or it could be based on the results of a clustering analysis.

Segmentation is a very useful preprocessing activity since you can now estimate different analytical models, each tailored to a specific segment. However, you need to be careful with it since by segmenting you will increase the number of analytical models to estimate, which will obviously also increase the production, monitoring, and maintenance costs.

DEFAULT DEFINITION

Credit scoring systems have been implemented for many years and since long before the Basel Capital Accords were introduced. Banks were using their own proprietary default definition. Roll-rate analysis can be used in order to gauge the stability of the default definition adopted. In roll-rate analysis, you investigate how customers already in payment arrears in one period move to a more or less severe default status in the next period. Exhibit 4.38 provides an example of roll-rate analysis. It can be seen that once customers are 90 days in payment arrears, most of them will keep this delinquency status for the next period and only a small minority will recover. Hence, using 90 days as a cutoff for the default definition seems a stable and viable option. Markov chains are essentially a more advanced approach of doing roll-rate analysis where you model the transition probabilities of moving from one default state to another during one period of time.

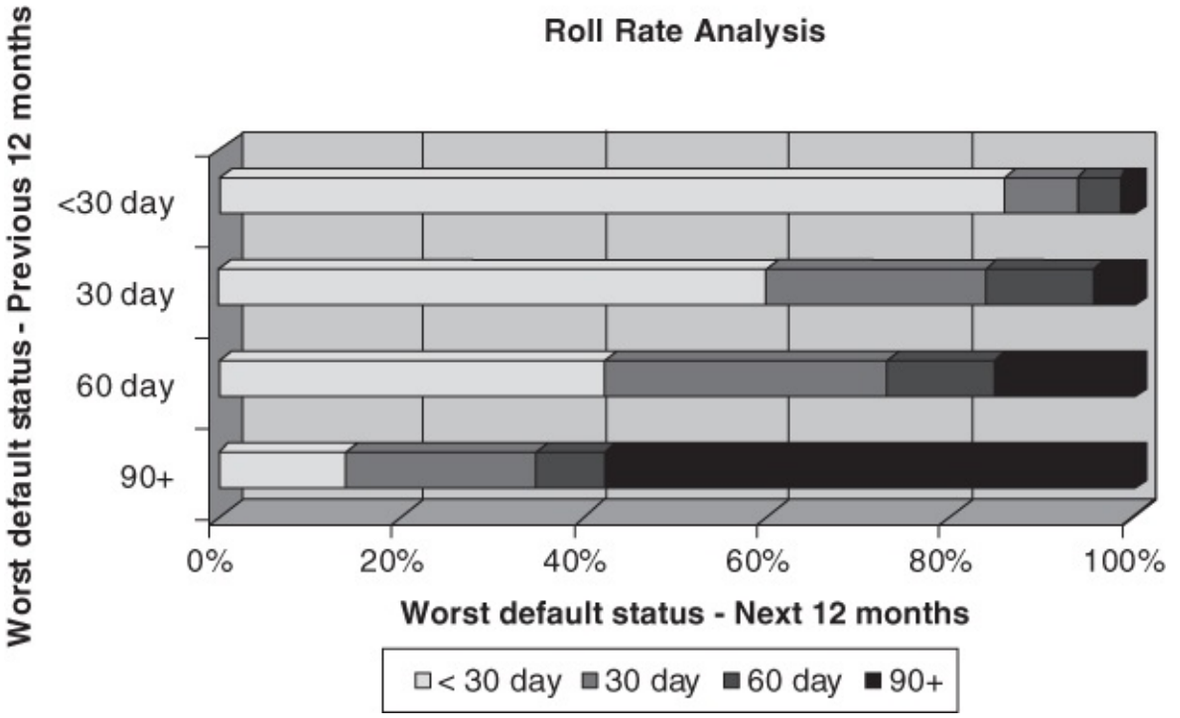


Exhibit 4.38 Roll-Rate Analysis

PRACTICE QUESTIONS

1. Discuss how the FICO score can be used by banks and other companies.
2. Why is the reject inference problem a sampling problem?
3. Contrast the benefits of categorization for the continuous versus categorical variables.
4. Consider the following data:

Characteristic	No Children	1 or 2 Children	3+ Children
Number of goods	1,500	2,200	300
Number of bads	500	300	200

   We are interested in the best coarse classification into two classes. One possibility is to split into no children versus children; another is two or fewer children versus three or more children. Find which of these splits is better using the chi-square statistic and PROC FREQ in SAS.

5. Preprocess the HMEQ data set in SAS Enterprise Miner as follows:
   - Draw a stratified sample of 10,000 observations (stratified based on the target).
   - Impute missing values by using the median for continuous variables and the mode for categorical variables.
   - Remove observations with outliers as follows: more than 3 standard deviations away from the mean for the continuous variables. For class variables with less than 20

different values, remove the values that occur in less than 5 percent of the observations.

- Use a StatExplore node to determine the five most predictive variables based on Cramer's V.

  Use an Interactive Grouping node to categorize all variables and code them using weights of evidence.

  Inspect the results. Which are the five most predictive variables according to the information value? Are these the same as according to Cramer's V?

NOTES

[1] FICO is an acronym for the Fair Isaac Corporation, the developers of the FICO score.

[2] The example is taken from Thomas, Edelman, and Crook (2002).

REFERENCES

Baesens, B. 2014. *Analytics in a Big Data World: The Essential Guide to Data Science and Its Applications*. Hoboken, NJ: John Wiley & Sons.

Chakraborty, G., P. Murali, and G. Satish. 2013. *Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS*. Cary, NC: SAS Institute.

Junqué de Fortuny, E., D. Martens, and F. Provost. 2013. "Predictive Modeling with Big Data: Is Bigger Really Better?" *Big Data* **1** (4): 215–226. doi:10.1089/big.2013.0037.

Little, R. J. A., and D. B. Rubin. 2002. *Statistical Analysis with Missing Data*. 2nd ed. New York: John Wiley & Sons.

Miner, G., J. Elder, A. Fast, T. Hill, B. Nisbet, and D. Delen. 2012. *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. Waltham, MA: Academic Press.

Thomas, L. C., D. B. Edelman, and J. N. Crook. 2002. *Credit Scoring and Its Applications*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Van Gestel, T., and B. Baesens. 2009. *Credit Risk Management: Basic Concepts: Financial Risk Components, Rating Analysis, Models, Economic and Regulatory Capital*. Oxford: Oxford University Press.