Module 2.
Preliminary Data Exploration and Data Preparation

Methodology Training
EXL Decision Analytics

Year 2013

GO NEXT. NOW.

Objectives and Scope

Course Goals

- Introduction to EXL DA Methodology
- Provide a structured overview of concepts relating to preliminary data exploration and data preparation as applied under EXL DA methodology
- Explain various alternative techniques of outlier treatment and missing value imputation
- Provide helpful “tricks of the trade”

Beyond the Scope of this Training

- Comprehensive coaching on Data Analysis and Preparation
- Technique-specific algorithms (unless required as part of methodology explanation)

Self Study Goals

- In-depth research on SAS implementation of explained techniques
- Innovations and new techniques related to methodology
- Discussion on advanced concepts can be taken up offline

EXL Decision Analytics Methodology Snapshot

We apply a set of highly effective tools, techniques and best practices for the end-to-end model development cycle

Stage 1	Preliminary Data Exploration	Univariate Analysis (EDD*) Modeling and Validation Split Bivariate Analysis	**These stages demand lot of manual effort in analyzing and understanding each and every variable**
Stage 2	Data Preparation	Outlier Treatment Missing Imputation Roll Ups and Data Merge	
Stage 3	Variable Creation	Dummy Variable Creation Binning and Banding Transformations Interactions and Groupings	**These stages require business sense and out-of-box thinking for brainstorming on creating hypothesis-based variables and dropping redundant features**
Stage 4	Variable Reduction	Variable Clustering Inter-Correlation Analysis Variance Inflation Factor Test	
Stage 5	Modeling	Modeling Technique Selection Model Improvements Ensemble	**These stages require good knowledge of statistical techniques for providing high-end quality solutions**
Stage 6	Validation and Stabilization	In-Sample Validation Out-of-Time Validation Bootstrapping Coefficient Blasting	

* Extended Data Dictionary

Table of Contents

Appendix

Chapter 1: Preliminary Data Exploration

Data Understanding and Prerequisites for Data Preparation

1.1 Data Collection Process

1. Identify Data Needs

Understand problem statement

2. Data Mapping

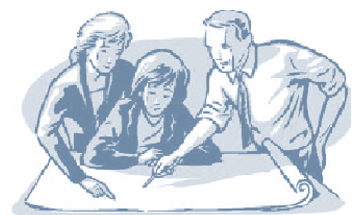
Interview clients

	A	B	C	D	E	F	G
1							
2							
3							

Study data layouts

Check data samples

3. Plan Data Request



Keep in mind population to be covered and alternative data sources

4. Send Data Request

List the data requirements clearly and precisely

5. Quality Check

Do data sanity checks and preliminary data analysis

Refer Appendix A.6 for details

1.2 Data Dictionary

A comprehensive data dictionary should be maintained and updated as and when any new information is gathered.

USE: It can go a long way in helping us understand the data better. For instance, it can help us to revisit old information and see what our initial hypothesis was and how it is changing with the new updated information.

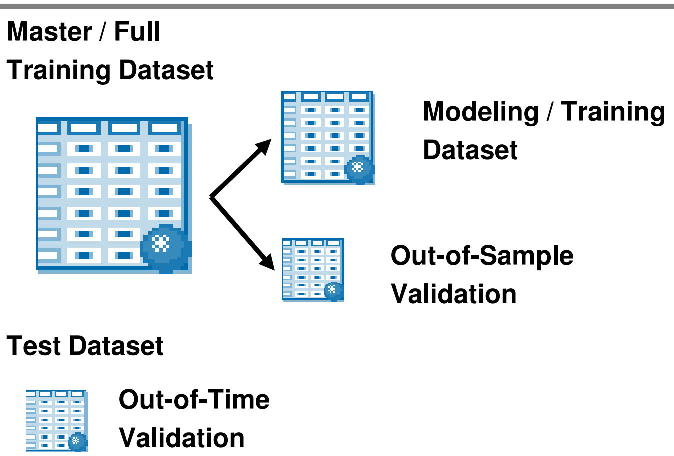
THINGS TO INCLUDE IN THE DATA DICTIONARY:

- Meaning of all Potential Predictors:
  - Maintain labels of as many variables as possible
  - If possible, one should also try to capture the business sense of these variables
  - Wherever things are not clear, it should be noted down so that it can be clarified with the client later on
- Clear Definition of Unique Identifier and its Meaning:
  - Ascertain the level at which data is to be rolled up / down. For instance,
    - **Individual** level
    - **Individual** x **Account** level
    - **Individual** x **Month** level
    - **Individual** x **Account** x **Month** level, etc.
  - Identify unique key of every dataset. Few examples below:
    - **Payment data** may be at **transaction level**
    - **Demographic data** at **individual level**
    - **Census data** at **zip code level**
- Dependent Variable Definition and Meaning: This is a very crucial step in modeling exercise as wrong definition can lead to completely wrong conclusions. In absence of a clear definition at this stage, it may be defined later after some actual data analysis.
- Variable Classification: If not already given, one should always try and classify the variables like
  - **Demographic variables**, e.g. age, gender
  - **Performance variables**, e.g. spend, number of transactions
  - **Credit Attributes**, e.g. total credit line, FICO score
  - **Census level**, e.g. population, location attributes such as income levels

1.3 Modeling and Validation Split

To start the modeling process, there is a need to create modeling and validation datasets. Validation dataset helps validate the performance of the model which is built using the modeling dataset. A poor performance on validation dataset would imply that the model is not robust.

Diagram Scale: Based on 60:40 split assumption



Data Preparation stage helps us create the master dataset for modeling and validation. Note that apart from out-of-sample validation, a fully independent out-of-time validation sample is also necessary to test the robustness of the model.

Step 1: Before we start the modeling process, we need to define and create the modeling population. From the data that is shared by the client, depending upon the scope of the analysis, an assessment of the required data (a certain amount of history, a certain length of future for prediction, quality of data, etc.), list down the defining criterion for eligible population.

Step 2: Split the final eligible population into parts – modeling dataset (also called training dataset) and validation datasets. This can be done using

- a random assessment (60:40 split or 80:20 split); or
- specific splitting criterion (based on time/segments)

It is important to validate our model for performance on data which was not used to build the model, but is the expected data that will be encountered in live environment – and hence, the need for validation dataset.

1.4 Univariate Analysis through EDD

The EDD, or Extended Data Dictionary macro produces a summary of the variables present in a dataset. It is a comprehensive and complete view of all variables with the following information being present.

- Number of observations present (numobs)
- Number of observations missing (nmiss)
- Number of unique values for a variable
- Mean, standard deviation, minimum, maximum and percentile distribution of numeric variables (mean, stddev, min, p1, p5, p25, median, p75, p95, p99, max)
- Six-most frequently occurring and five-least frequently occurring values for character variables

Syntax:

```
LIBNAME catalog "<path of the catalogue>";
OPTIONS mstored sasmstore = catalog;
%EDD      (
          INLIB                = <Location of the input dataset>,
          INPUTDATA            = <Name of input dataset>,
          EDD_OUT_LOC_XLS      = <Name and location of the output XLS file>,
          OUTLIB               = <Location of the output dataset>,
          OUTDATA              = <Name of output dataset>,
          NUM_UNIQ             = <Option>*
          );
*NUM_UNIQ can either be Y or N depending on whether the # of unique values column is desired.
```

EDD OUTPUT ANALYSIS

Three Character Variables

39,007 Observations

Case of Single Unique Value for Entire Data

Mode for Character Variables

These may be (though not necessarily) "indicator variables".
Reason:
a. Number of Unique Values = 2; and
b. Minimum Value = 0; and
c. Maximum Value =1

Obs	name	label	type	var_length	n_pos	numobs	nmiss	unique	mean_or_top1	stddev_or_top2	min_or_top3	p1_or_top4	p5_or_top5	p25_or_top6	median_or_bot5	p75_or_bot4	p95_or_bot3	p99_or_bot2	max_or_bot1
1	NOTIFICATION_CD	Notification Code	char	1	1199	39007	0	2	N::37696	Y::1311									
2	POSTING_TYPE_CD	Posting Type Code	char	5	1092	39007	0	8	LS::25184	RP::7492	SB::5226	WU::494	WN::255	OC::210	WU::494	WN::255	OC::210	CO::142	FU::4
3	VEH_MAKE	Vehicle Make	char	55	888	39007	28646	32	::28646	Chev::4371	Ca::1669	GM::1479	Pon::955	Bui::777	SAAB::1	Mer::1	Maz::1	Lex::1	BMW::1
4	LAST_BID_AMT	Last Bid Amt.	num	8	80	39007	0	1	0	0	0	0	0	0	0	0	0	0	0
5	ind_ODOM_le_60K	ODOM <= 60K	num	8	808	39007	0	2	0.92	0.28	0	0	0	1	1	1	1	1	1
6	SOLD_IND	Sold Indicator	num	8	16	39007	0	2	0.27	0.44	0	0	0	0	0	1	1	1	1
7	VEH_MODEL_YR	Vehicle Model Year	num	8	0	39007	28646	8	2005.95	1.00	2002	2004	2004	2005	2006	2007	2007	2008	2009
8	SOLD_DT	Sold Date	num	8	8	39007	0	296	20085982	4689	2E+07	20080702	2E+07	2E+07	20090115	2E+07	2E+07	2E+07	2E+07
9	FIXED_PRICE_AMT	Fixed Price Amt.	num	8	64	39007	0	447	11297.36	8321.18	0	0	0	6700	11200	15800	26300	35300	8788800
10	FLOOR_PRICE_AMT	Floor Price Amt.	num	8	72	39007	0	496	14414.45	7103.27	900	4600	6300	9500	12800	17400	28800	36700	1754000
11	AQ_AMT	Acquisition Amt.	num	8	24	39007	0	8021	4084.26	7670.88	0	0	0	0	0	7442	20450	31469	5796040
12	WHOLESALE_VALUE	Wholesale Value	num	8	112	39007	0	18985	15747.56	7442.90	1000	5277	7148	10530	14046	18942	30807	38905	1770000
13	KEY	Key Variable	num	8	232	39007	0	39007	539288	299945	20	20744	56424	263410	560456	796925	992968	1060935	1073955

13 Variables

Ten Numeric Variables

Large number of missing values (i.e. low fill rate) for two variables.

Outliers

Number of Observations = Number of Unique Values

Therefore, dataset is unique at variable "KEY".

Variable takes value 0 for at least 50% data.

1.5 Bivariate Analysis

Unlike Univariate Analysis that involves standalone analysis of a variable (independent variable) distribution, the Bivariate profiling is a simultaneous analysis of two variables (a dependent and an independent variable)

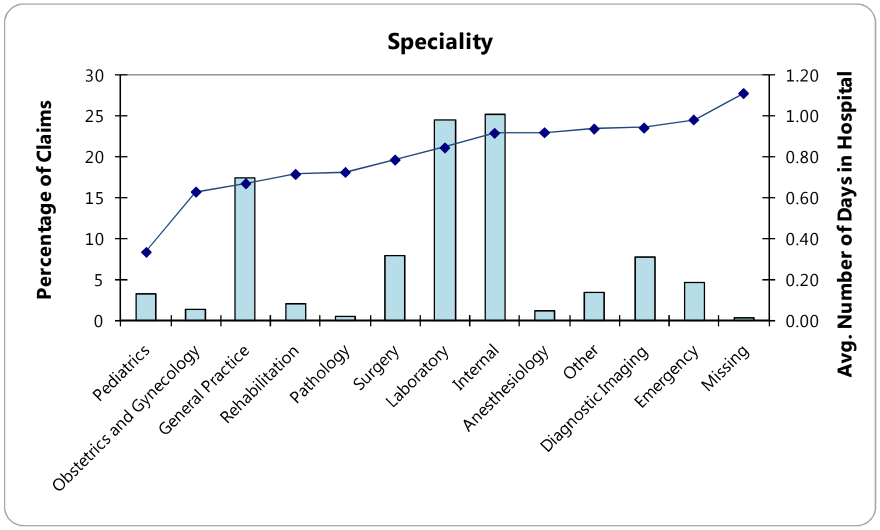
Bivariate profiling involves

- Dividing independent variable into different categories (or bins)
- Analyzing trend in sizing (percentage of records) across categories (or bins)
- Analyzing trend in mean value of dependent variable across categories (or bins)

Illustration:

Dependent Variable : *Number of Days Spent by Patient in Hospital in a Year*

Independent Variable : *Specialty of the Doctor*

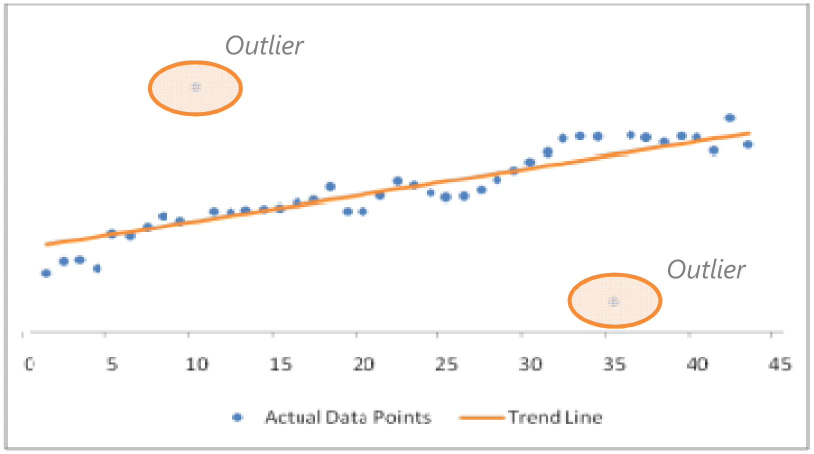


- On average, patients of **Pediatrics** spend very less number of days in hospital
- On the other hand, **Diagnostic Imaging** and **Emergency** cases spend longer time
- Majority of cases pertain to **Internal**, **Laboratory** and **General Practice** type

Chapter 2: Data Prep: Outlier Treatment

2. Outlier Treatment

An outlier is a single observation "far away" from rest of the data.



REASONS FOR OUTLIERS:

- Errors
  - Data errors
  - Sampling error
  - Standardization failure
  - Faulty distributional assumptions
  - Human Error
- Genuine Outliers

WHY DO WE CARE ABOUT OUTLIERS?

- Outliers are BAD
  - The presence of outliers can lead to inflated error rates and substantial distortions of results that can lead to wrong conclusions and inferences.
- Outliers are GOOD
  - The outliers can provide useful information in the data, for example, a spike in spend behavior of some customers may prove to be the deciding factor in marketing response campaigns. So care should be taken while dealing with outliers.

In short, outliers are important and hence should not be ignored.

TECHNIQUES FOR OUTLIER DETECTION / TREATMENT:

- Capping and Flooring Technique
- Exponential Smoothing Technique
- Sigma Approach
- Robust Regression Technique
- Mahalanobis Distance Technique

2.1 Capping and Flooring Technique

OUTLIER DETECTION/TREATMENT TECHNIQUES

A. Capping and Flooring Technique

B. Exponential Smoothing Technique

C. Sigma Approach

D. Robust Regression Technique

E. Mahalanobis Distance Technique

Technique Description	In this technique, the outliers are identified and treated based upon the values of P99 and P1.
Outlier Identification	**Outlier** is defined as the value falling out by 'x' times P99 or 'y' times P1. **Note**: x and y are the factors which can take value any integer value as required by the data distribution and as decided based on the application.
Outlier Treatment	▪ **Capping** - All values falling higher than 'x' times P99, are *capped* at the value "*x * P99*". ▪ **Flooring** – All values less than 'y' times P1, are *floored* at the value "*y * P1*". **Note**: - Values are capped at x * P99 when P99 and PMax both are positive. - Values are floored at y * P1 when P1 and PMin both are negative.
Advantages	▪ Easy to understand & implement ▪ Run time is less
Disadvantages	▪ Distribution of the data is not taken into account while identifying the outliers ▪ Rank order is not maintained

2.2 Exponential Smoothing Technique

OUTLIER DETECTION/TREATMENT TECHNIQUES

A. Capping and Flooring Technique

B. **Exponential Smoothing Technique**

C. Sigma Approach

D. Robust Regression Technique

E. Mahalanobis Distance Technique

Technique Description	In this technique, the curve between P95 to P99 is extrapolated beyond P99, to identify the values falling above the curve. The values falling outside the curve are outliers and are treated according to some functions depending upon the boundary conditions.
Outlier Identification	Curve between P95 and P99 is extrapolated beyond P99. The values between P99 and PMax which fall outside this curve are termed as **outliers**. **Note**: Similar approach is followed for values between PMin and P1.
Outlier Treatment	Based upon the boundary condition a specific function is used to treat the outlier and maintain the rank order.
Advantages	▪ Rank order is maintained ▪ Distribution of data is taken into account while identifying the outliers
Disadvantages	▪ Functions involved in treating the outliers are quite complex

2.3 Sigma Approach

OUTLIER DETECTION/TREATMENT TECHNIQUES

A. Capping and Flooring Technique

B. Exponential Smoothing Technique

C. Sigma Approach

D. Robust Regression Technique

E. Mahalanobis Distance Technique

Technique Description	In this technique, the outliers are identified and treated based upon the values of mean and standard deviation.
Outlier Identification	Outlier is defined as the value falling out of mean '+' or '-' 'x' times sigma (standard deviation) **Note**: x is the factor which can take value any integer value as required by the data distribution and as decided specifically.
Outlier Treatment	▪ **Capping** - All values falling higher than mean *plus* 'x' times sigma, are *capped* at the value "*mean + x * sigma*". ▪ **Flooring** – All values less than mean *minus* 'x' times sigma, are *floored* at the value "*mean – x * sigma*".
Advantages	▪ Easy to understand & implement
Disadvantages	▪ Rank order is not maintained ▪ This method works best only when variables follow a normal distribution

2.4 Robust Regression Technique

OUTLIER DETECTION/TREATMENT TECHNIQUES

A. Capping and Flooring Technique

B. Exponential Smoothing Technique

C. Sigma Approach

D. Robust Regression Technique

E. Mahalanobis Distance Technique

Technique Description	This technique involves running regression repeatedly to identify outliers by assigning weights to the observations. The weights are on the basis of the prediction error (residual) in different iterations. **Note**: Higher residual means lower weight.
Outlier Identification	▪ Weights are assigned to each observation, based on the normalized residual value ▪ High breakdown value method is used --- it is the measure of the contamination in the data that an estimation can withstand and still maintain its robustness.
Outlier Treatment	The outliers are ignored (deleted) while making the model.
Advantages	▪ Effect of outliers on model performance is minimized
Disadvantages	▪ Computation of robust estimates is resource intensive ▪ Ignoring outliers may result in loss of data/information

2.5 Mahalanobis Distance Technique

OUTLIER DETECTION/TREATMENT TECHNIQUES

A. Capping and Flooring Technique

B. Exponential Smoothing Technique

C. Sigma Approach

D. Robust Regression Technique

E. Mahalanobis Distance Technique

Technique Description	In this technique, the outliers are identified by the magnitude of '*Mahalanobis'* or statistical distance from the origin. Weights are given to each observation as the inverse of '*Mahalanobis'* distance.
Outlier Identification	The observations with extremely low weights can be considered as **outliers**.
Outlier Treatment	Weighted regression is run to take into account the effect of weight given to each observation. Greater the '*Mahalanobis*' distance, lesser is the weight of that observation & hence lesser the contribution of that observation in final model.
Advantages	▪ Effect of outliers on model performance is minimized
Disadvantages	▪ Small change in data distribution could lead to more than normal deterioration of the model performance ▪ Complexity in calculation of '*Mahalanobis*' distance for weighted regression

2.6 Summary

Comparison Summary

Technique / *Description*	A. Capping and Flooring Technique	B. Exponential Smoothing Technique	C. Sigma Approach	D. Robust Regression Technique	E. Mahalanobis Distance Technique
Outlier Identification	▪ Multiples of P99 and P1	▪ Extrapolation of distribution	▪ Multiple of standard deviation from mean	▪ Assigning weights to observations	▪ Mahalanobis distance from origin
Outlier Treatment	Capping and flooring at multiples of P99 and P1	Boundary condition of exponential function	Capping and flooring at multiples of standard deviation from mean	Outliers ignored in modeling	Outliers given lower weights
Advantages	▪ Easy to understand and implement ▪ Run time is less	▪ Rank ordering ▪ Considers distribution	▪ Easy to understand and implement	▪ Effect of outliers minimized	▪ Effect of outliers minimized
Disadvantages	▪ No Rank ordering ▪ Distribution independent	▪ Complex exponential function	▪ No Rank ordering ▪ Best for normal distribution	▪ Data/ information loss ▪ Computationally complex	▪ Over-dependency on distribution ▪ Complex technique

In general, it has been observed:

✓ **Sigma Approach** comes out to be the best technique in case of **Logistic Regression**

✓ **Mahalanobis Distance Technique** is best for **Linear Regression**

Note: Logistic regression corresponds to binary dependent variable;
Linear regression is run to model a continuous dependent variable.

Exercise

Exercise 1. For the given 100 records, _DEPVAR_ is the dependent variable and X1 is a predictor. Analyze distribution of X1, identify outliers, treat them using all 5 techniques and compare the results.

[Hint: For using SAS macros, refer Appendix A.1 – A.5]

	A	B	C
1	ID	_DEPVAR_	X1
2	101	10	565
3	102	6	866
4	103	4	371
5	104	2	568
6	105	3	788
7	106	9	709
8	107	3	153
9	108	10	314
10	109	4	909
11	110	3	467
12	111	5	578
13	112	1	687
14	113	8	260
15	114	9	891
16	115	3	584
17	116	10	768
18	117	8	110
19	118	5	893
20	119	5	619
21	120	1	170
22	121	8	96
23	122	4	376
24	123	10	936
25	124	3	769
26	125	4	93

	A	B	C
1	ID	_DEPVAR_	X1
27	126	7	784
28	127	5	225
29	128	10	470
30	129	6	322
31	130	3	763
32	131	8	541
33	132	1	814
34	133	8	172
35	134	7	770
36	135	3	948
37	136	5	935
38	137	4	764
39	138	1	357
40	139	8	458
41	140	7	931
42	141	4	258
43	142	9	630
44	143	8	659
45	144	2	92
46	145	8	146
47	146	10	439
48	147	9	751
49	148	3	114
50	149	4	324
51	150	0	530

	A	B	C
1	ID	_DEPVAR_	X1
52	151	8	35
53	152	4	499
54	153	2	806
55	154	9	495
56	155	4	136
57	156	10	819
58	157	1	292
59	158	8	437
60	159	6	262
61	160	2	156
62	161	7	270
63	162	2	912
64	163	9	600
65	164	6	791
66	165	0	204
67	166	7	894
68	167	2	234
69	168	5	12000
70	169	4	701
71	170	7	596
72	171	1	293
73	172	5	656
74	173	9	81
75	174	1	180
76	175	2	182

	A	B	C
1	ID	_DEPVAR_	X1
77	176	10	549
78	177	1	903
79	178	4	40
80	179	4	769
81	180	1	512
82	181	9	368
83	182	2	820
84	183	6	563
85	184	5	746
86	185	6	357
87	186	0	383
88	187	0	521
89	188	8	954
90	189	8	413
91	190	3	461
92	191	1	373
93	192	4	169
94	193	4	218
95	194	9	341
96	195	10	203
97	196	0	23
98	197	8	65
99	198	8	647
100	199	10	489
101	200	5	441

Chapter 3: Data Prep: Missing Value Imputation

3. Missing Value Imputation

MVI is a process of replacing missing values of a variable with the best possible estimates.

Missing Identification

Missing structure has to be identified for all variables.

Variable Type	Missing Value
Numeric	.
Character	<blank>

Along with above cases, both numeric and character variables can take invalid values which should be converted to missing. e.g.

Variable	Invalid Value
Income	9999999
City	XX

Impact of Missing Data

Most multivariate analysis techniques, especially regression, drop all observations with missing values.

Solution

Missing Value Imputation – replace missing values with estimates.

Word of Caution

An incorrect imputation can result in an incorrect estimation / prediction.

There are a variety of techniques for missing value imputation; but these should be considered more as scenario-specific than just being a set of pure alternative choices.

Missing Value Imputation Techniques

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. DNI
K. Multiple Imputation

3.1 Impute Missing Values with ZERO

A B C D
E F G H
I J K L

MVI TECHNIQUES

A. **Impute Missing Values with ZERO**

B. Impute Missing Values with MEDIAN

C. Impute Missing Values with MEAN

D. Impute Missing Values with MODE

E. Information based Segmentation

F. Non-Missing Dummy Creation

G. Imputation and Non-Missing Dummy Creation

H. Impute based on Bivariate Graphs

I. Impute using Regression on other Non-Missing Predictors

J. Impute using CART

K. DNI

L. Multiple Imputation

Technique Description

Impute missing values with ZERO.

Execution

- Run EDD
- Identify numeric variables with missing values
- Check if there are same number of missing values for a particular type of variables (whose source dataset is a subset of other variable source datasets)
- If it makes sense, impute missing values with 0

Example

Consider a case where

- Modeling dataset has 1 million records
- Debt history is populated for those in debt (400K records)

Variable	#Obs.	#Missing	Data Source
age	1,000,000	0	Demographics
ind_female	1,000,000	0	Demographics
ind_payment_due	1,000,000	600,000	Debt History
due_amt	1,000,000	600,000	Debt History

It makes sense to impute 600K missing records with ZERO.

Application & Evaluation Levers

Applicable for variables whose missing values should actually have been taking ZERO value anyway.

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	High
Time Involved	Low
Coverage	High

3.2 Impute Missing Values with MEDIAN

MVI TECHNIQUES

A. Impute Missing Values with ZERO

B. Impute Missing Values with MEDIAN

C. Impute Missing Values with MEAN

D. Impute Missing Values with MODE

E. Information based Segmentation

F. Non-Missing Dummy Creation

G. Imputation and Non-Missing Dummy Creation

H. Impute based on Bivariate Graphs

I. Impute using Regression on other Non-Missing Predictors

J. Impute using CART

K. DNI

L. Multiple Imputation

Technique Description

Impute missing values with MEDIAN.

Execution

- Run EDD
- Identify numeric continuous variables with missing values
- Check variable distribution
- Identify variables with highly skewed distribution
- If it makes sense, impute missing values with median value

Example

Consider a variable with following distribution:

Variable XYZ			
N	800	NMISS	15
MEAN	355	MEDIAN	51
MIN	0	MAX	10,000
P1	2	P99	5,000
P5	5	P95	4,750
P10	9	P90	88
P25	26	P75	75

Variable distribution is highly skewed. Extremely large values for ~5% data are distorting the mean value, thereby making it significantly different from the median value. Here, it makes sense to impute 15 missing records with MEDIAN.

Application & Evaluation Levers

Applicable for continuous variables whose distribution is highly skewed.

Production Ready	High
Easy to Understand	High
Business Implication	Medium
Approximation	Medium
Time Involved	Low
Coverage	High

3.3 Impute Missing Values with MEAN

MVI TECHNIQUES

A. Impute Missing Values with ZERO

B. Impute Missing Values with MEDIAN

C. Impute Missing Values with MEAN

D. Impute Missing Values with MODE

E. Information based Segmentation

F. Non-Missing Dummy Creation

G. Imputation and Non-Missing Dummy Creation

H. Impute based on Bivariate Graphs

I. Impute using Regression on other Non-Missing Predictors

J. Impute using CART

K. DNI

L. Multiple Imputation

Technique Description

Impute missing values with MEAN.

Execution

- Run EDD
- Identify numeric continuous variables with missing values
- Check variable distribution
- Identify variables with evenly distributed values
- If it makes sense, impute missing values with mean value

Example

Consider time series data (daily price data for 1 year):

DAILY_CLOSING_PRICE			
N	246	NMISS	10
MEAN	4,340	MEDIAN	4,492
MIN	0	MAX	6,288
P1	2,607	P99	6,273
P5	2,710	P95	5,861
P10	2,919	P90	5,272
P25	3,903	P75	4,958

Variable values are quite evenly distributed. Mean and median values are close. 10 missing records can be safely imputed with MEAN.

***NOTE**: MEDIAN can also be used for imputation in this case; but MEAN value would be easier to interpret.*

Application & Evaluation Levers

Applicable for continuous variables whose values are not skewed, but somewhat evenly distributed.

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	Medium
Time Involved	Low
Coverage	High

3.4 Impute Missing Values with MODE

MVI TECHNIQUES

A. Impute Missing Values with ZERO

B. Impute Missing Values with MEDIAN

C. Impute Missing Values with MEAN

D. Impute Missing Values with MODE

E. Information based Segmentation

F. Non-Missing Dummy Creation

G. Imputation and Non-Missing Dummy Creation

H. Impute based on Bivariate Graphs

I. Impute using Regression on other Non-Missing Predictors

J. Impute using CART

K. DNI

L. Multiple Imputation

Technique Description

Impute missing values with MODE.

Execution

- Run EDD
- Identify character variables with missing values
- If number of missing values is not very large, impute missing values with mode

Example

Consider a character variable "STATE_CD" with say "six" unique values.

Mode: Value with Highest Frequency

OCCURRENCE	STATE	STATE_CD	FREQUENCY
top 1	Indiana	"IN"	2,400
top 2	New York	"NY"	1,400
top 3	New Jersey	"NJ"	1,000
bottom 3	Arizona	"AZ"	800
bottom 2	Texas	"TX"	250
bottom 1	**Missing Value**	**" "**	**150**
Total number of Observations			**6,000**

150 missing records may be imputed with mode value "IN".

Application & Evaluation Levers

Applicable for character variables whose fill rate is high enough (i.e. whose number of missing records is not very large).

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	Medium
Time Involved	Low
Coverage	High

3.5 Information based Segmentation

MVI TECHNIQUES

A. Impute Missing Values with ZERO

B. Impute Missing Values with MEDIAN

C. Impute Missing Values with MEAN

D. Impute Missing Values with MODE

E. Information based Segmentation

F. Non-Missing Dummy Creation

G. Imputation and Non-Missing Dummy Creation

H. Impute based on Bivariate Graphs

I. Impute using Regression on other Non-Missing Predictors

J. Impute using CART

K. DNI

L. Multiple Imputation

Technique Description

Use a classifier to segment population with high missing rate on a set of variables.

Execution

- Identify classifier by business logic or by iterative analysis
- Segment population into sub populations based on classifier
- Resulting sub populations should have high fill rate
- Impute with mean/median/zero (as applicable)

Example

Low Fill Rate (orange)

High Fill Rate (blue)

All Accounts — 150 Accounts

Var.	N	NMISS	Fill Rate
A	150	90	40.0%
B	150	105	30.0%
C	150	50	66.7%

Individual Liable — 100 Accounts

Var.	N	NMISS	Fill Rate
A	100	88	12.0%
B	100	100	0.0%
C	100	2	98.0%

Corporate Liable — 50 Accounts

Var.	N	NMISS	Fill Rate
A	50	2	96.0%
B	50	5	90.0%
C	50	48	4.0%

There is a case for segment level modeling. Instead of dropping all three, **C** can be used in '**Individual Liable**' segment model, while **A** and **B** can be used in the other one.

Application & Evaluation Levers

Applicable where correct imputation is highly dependent on value of a classifier.

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	Low
Time Involved	High
Coverage	High

3.6 Non-Missing Dummy Creation

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. Impute using CART
K. DNI
L. Multiple Imputation

Technique Description

Create a binary variable identifying non-missing vs. missing data.

Execution

- Run EDD
- Identify variables with high percentage of missing data
- Write "automatic-code" in spreadsheet or using Block-copy in Ultra Edit
- Keep dummies for modeling and drop original variables

Example

Consider the following case:

BALANCE			
N	100	NMISS	70
MEAN	320	MEDIAN	250
MIN	0	MAX	2,000
P1	100	P99	1,050
P5	130	P95	900
P10	140	P90	700
P25	175	P75	450

Action A:

Create an indicator variable for non-missing Balance Amount.

Syntax:

i_NMI_Balance = (Balance ne .);

Action B:

Drop variable "BALANCE" from modeling dataset.

Application & Evaluation Levers

Applicable when missing rate is high and the variable does not have a very strong correlation with dependent variable.

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	Low
Time Involved	Low
Coverage	High

3.7 Non-Missing Dummy Creation

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. Impute using CART
K. DNI
L. Multiple Imputation

Technique Description

Create a non-missing dummy and also retain original variable.

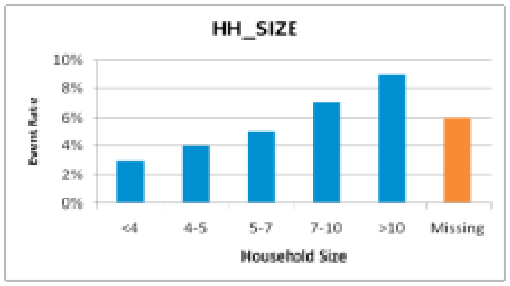
Execution

- Run EDD
- Run bivariate graphs to identify variables with high correlation with dependent variable
- Create non-missing dummies
- Impute missing with Mean/Median (as applicable)
- Use **both** imputed value & non-missing dummy as predictors

Example

Consider the following case:

HH_SIZE			
N	100	NMISS	70
MEAN	5.5	MEDIAN	5
MIN	1	MAX	100
P1	2	P99	11
P5	3	P95	10
P10	3	P90	8
P25	4	P75	6



Non-Missing Indicator Creation:

i_NMI_HH_SIZE = (HH_SIZE ne .);

Missing Imputation by Median:

If (HH_SIZE eq .) then HH_SIZE = 5;

Application & Evaluation Levers

Applicable when missing rate is high and the variable has a quite high correlation with the dependent variable.

Production Ready	High
Easy to Understand	High
Business Implication	Medium
Approximation	Low
Time Involved	Medium
Coverage	Medium

3.8 Impute based on Bivariate Graphs

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. Impute using CART
K. DNI
L. Multiple Imputation

Technique Description

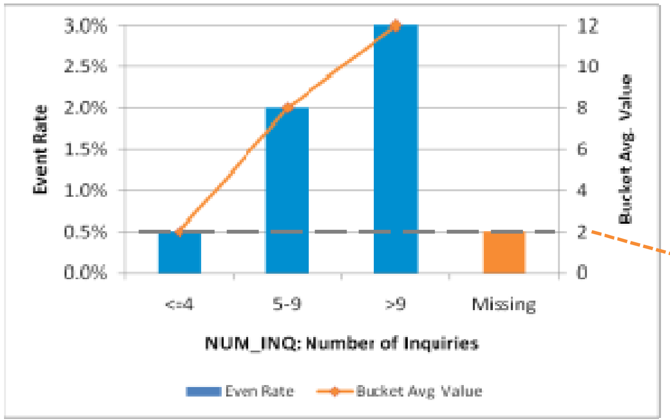
Impute missing value with an average value of the bucket having the same dependent variable event rate or average value as the missing bucket.

Execution

- Run bivariate analysis
- Identify variables with high correlation with dep. variable
- Identify if dep. variable event rate or average value on missing bucket is in the same range as some other bucket
- Impute missing with the average value in that bucket

Example

Missing Value Imputation: *If (NUM_INQ eq .) then NUM_INQ = 2;*



Average Value of Dependent Variable for 'Missing Value' bucket and '<=4' bucket is same.

Application & Evaluation Levers

Applicable when variable has a high correlation with dep. variable and the avg. (mean or median) dependent value for the missing bucket is uniquely similar to that for some other bucket.

Production Ready	High
Easy to Understand	High
Business Implication	High
Approximation	Medium
Time Involved	Medium
Coverage	Low

3.9 Impute using Regression

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. **Impute using Regression on other Non-Missing Predictors**
J. Impute using CART
K. DNI
L. Multiple Imputation

Technique Description

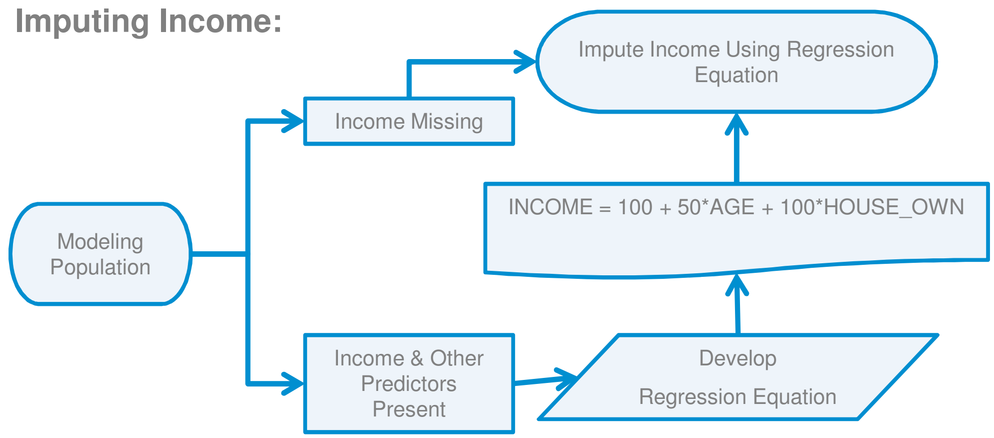
Develop a linear regression model with non-missing records to predict the value of missing variable.

Execution

- Run EDD
- Identify variables which are always present and will always be present in production
- Build a regression model on the non missing population to predict the missing variable
- Validate signs of estimated parameters using business logic
- Impute the missing variable using regression equation

Example

Imputing Income:



Application & Evaluation Levers

Applicable for variables that have reliable predictive relationship with other consistently available independent variables.

Production Ready	Medium
Easy to Understand	Medium
Business Implication	Medium
Approximation	Medium
Time Involved	High
Coverage	Low

3.10 Impute using CART

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. **Impute using CART**
K. DNI
L. Multiple Imputation

Technique Description

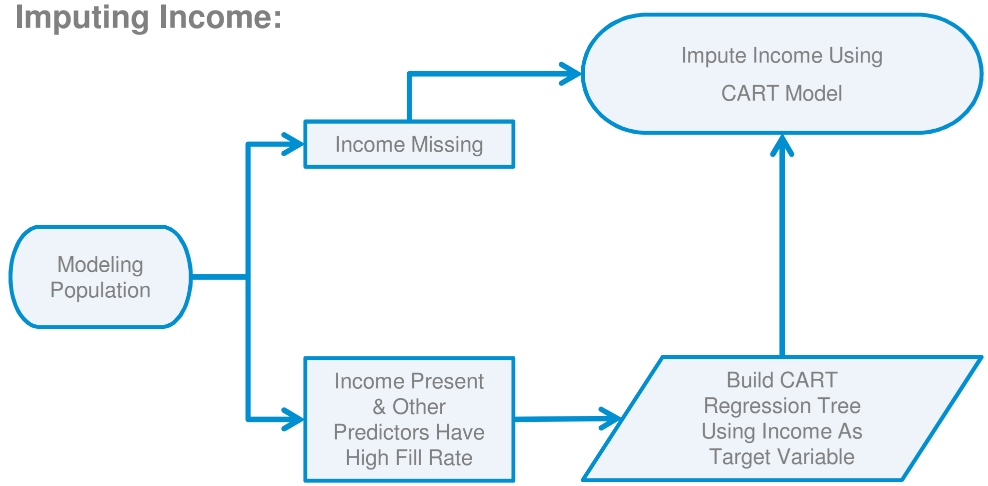
Develop a CART classification/regression tree with non-missing records to predict the value of missing variable.

Execution

- Run EDD
- Identify variables with high fill rate (say, more than 80%)
- Build a CART model on the non missing population using selected variables to predict the missing variable
- Impute the missing variable using CART model

Example

Imputing Income:



Application & Evaluation Levers

Applicable for variables that have reliable predictive relationship with other independent variables.

Production Ready	Medium
Easy to Understand	Medium
Business Implication	Medium
Approximation	Medium
Time Involved	High
Coverage	Medium

3.11 Do Not Impute

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. Impute using CART
K. DNI
L. Multiple Imputation

Technique Description

Do Not Impute (DNI) - Drop variables or drop observations.

Execution

- Drop variables which have a very low coverage and on which any other technique is not applicable
- Drop observations that have any missing value present

Example

To Predict: *Probability of Employee Attrition*

- "Employee's favorite color" has high missing rate. Also, it doesn't seem to be an important variable for this model
- "Months since last promotion" has few missing values, but this variable seems one among the essential variables

Variable	#Obs.	#Missing	Missing Rate
FAV_COLOR	20,000	15,000	75.00%
MNTHS_SINCE_LAST_PROM	20,000	20	0.01%

Action A: Drop **FAV_COLOR**
Action B: Drop 20 records where **MNTHS_SINCE_LAST_PROM** is missing

Application & Evaluation Levers

Apply to variables with low coverage that are not essential to the model; **apply to records** if they are missing values for essential variables; **apply to records** with missing values for many independent variables.

Production Ready	Low
Easy to Understand	Medium
Business Implication	Low
Approximation	High
Time Involved	Low
Coverage	Low

3.12 Multiple Imputation

MVI TECHNIQUES

A. Impute Missing Values with ZERO
B. Impute Missing Values with MEDIAN
C. Impute Missing Values with MEAN
D. Impute Missing Values with MODE
E. Information based Segmentation
F. Non-Missing Dummy Creation
G. Imputation and Non-Missing Dummy Creation
H. Impute based on Bivariate Graphs
I. Impute using Regression on other Non-Missing Predictors
J. Impute using CART
K. DNI
L. Multiple Imputation

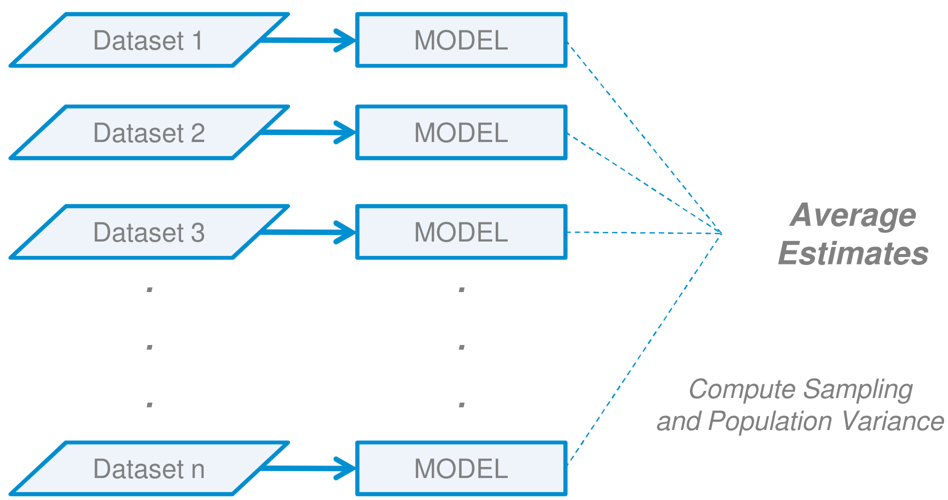
Technique Description

Prepare multiple imputed datasets using different valid imputation strategies.

Execution

- Make multiple imputation of missing data (3-10) sets
- Estimate Separate Models on all imputed datasets
- PROC MI/PROC MI ANALYZE

Example



Application & Evaluation Levers

Apply when more than one approach may be valid and these may generate quite different final model results.

Production Ready	Low
Easy to Understand	Low
Business Implication	Low
Approximation	Low
Time Involved	High
Coverage	Medium

Exercise

Exercise 2. For the given 50 records, _DEPVAR_ is the dependent variable, X1 and X2 are predictors. Impute missing values of X2 based on

1. Distribution of X2 only
2. Distribution of X2 and _DEPVAR_
3. Distribution of X2, _DEPVAR_ and X1

	A	B	C	D
1	**ID**	**_DEPVAR_**	**X1**	**X2**
2	101	0	287	0
3	102	1	596	
4	103	0	885	168
5	104	0	109	424
6	105	0	671	232
7	106	0	699	306
8	107	0	287	402
9	108	0	420	271
10	109	1	529	498
11	110	1	534	495
12	111	0	917	126
13	112	0	112	249
14	113	0	297	394
15	114	1	338	423
16	115	1	137	444
17	116	1	664	
18	117	0	363	398
19	118	0	758	332
20	119	0	429	208
21	120	1	190	
22	121	0	435	41
23	122	0	727	130
24	123	0	761	14
25	124	0	142	370
26	125	0	242	188

	A	B	C	D
1	**ID**	**_DEPVAR_**	**X1**	**X2**
27	126	0	798	191
28	127	1	590	
29	128	1	207	105
30	129	0	622	403
31	130	0	954	
32	131	1	802	120
33	132	1	652	336
34	133	0	960	162
35	134	0	384	197
36	135	0	721	19
37	136	0	648	165
38	137	0	824	366
39	138	0	944	393
40	139	0	823	471
41	140	1	807	
42	141	0	440	56
43	142	1	446	465
44	143	0	358	194
45	144	0	847	398
46	145	0	646	381
47	146	1	700	165
48	147	0	702	302
49	148	0	717	500
50	149	0	424	481
51	150	1	949	434

Chapter 4: Post Outlier Treatment and Imputation

4.1 Identify Non Usable Variables

Even at this early stage one can identify certain variables which can be deemed as '**non-usable for modeling purpose**'. This way we can reduce the dimension of the dataset. Some logics that can be applied are as follows:

- **Variables with a single unique value throughout the dataset:** By definition, such variables have zero explanatory power and hence are irrelevant for any analysis. These variables are usually flags like merge indicators.

- **ID Variables**: Such variables may be needed in the dataset for observation tagging. However, they should NOT be used as predictors in the model.

- **Variables with very low fill rates:**

  Case I: Variable, in question, is defined over a specific segment only. This segment may be used to subset the modeling dataset for developing segment-specific models. In such a case, the same variable is usable for one segment; while non-usable for the other.

  Case II: Missing value may signify something; and may be associated with a meaningful value.

  Case III: Variable fill rate is less than even 50% but there is a strong business case for its inclusion. In this case, the appropriate technique of missing value imputation should be applied.

  Case IV: If none of the above cases holds, some minimum fill rate cut-off may be put for dataset dimension reduction. According to standard modeling conventions, any variable with fill rate lower than 50% is not included in the model. This cut-off for fill rate can be set higher or lower depending on how well populated is the data received.

- Variables which cannot be used because of **implementation issues** should be dropped.

- Certain variable like Gender, Ethnicity which cannot be used due to **regulatory issues** (depending upon the business problem in context) should also be dropped.

4.2 Reformat Variables

Categorical and continuous variables are treated differently in most of the analysis like CART, Logistic Regression, Bivariate analysis (as continuous variables would require binning and banding whereas categorical won't). Hence, it's always advisable to separate out possible categorical variables from the continuous ones.

Few points to remember

- Look at EDD to check variable format. However, it is possible that variable format is not correct in data itself. Variable format type column in EDD can't help in such exceptions.
- Check number of unique values. Numerical variables taking only 10-15 unique value may be treated as categorical. It's a subjective call, depending on the variable and its expected use in model.
- Apply business sense before treating variables as continuous / categorical

A numeric variable should never be converted to a categorical variable if the values have ordered meaning, even if the number of its unique values is just 3 or 4.

4.3 Immediate Next Steps

Some Pointers as Immediate Next Steps

- Redundant variables should be dropped. Such variables don't add any extra information. To identify such variables, variable reduction techniques can be used like variable clustering.
- Few predictors may be highly correlated. In such a situation, coefficient estimates may change erratically in response to small changes in the data. This problem of 'multicollinearity' should be taken care of.
- In case of categorical dependent variable, event rate needs to be looked upon. If event rate is too low, it may create a problem in developing a robust model. The modeler may need to do oversampling.

These concepts (as components of data analysis and modeling) would be covered in more detail in the next module.

Appendix

A.1. Macro Call: Capping and Flooring

Capping and Flooring Macro (X times) Syntax

```
%OUTL_TREATMENT_X_TIMES(
                    EDD_LOC_LIB        = <Library where EDD in form of SAS dataset is located>,
                    EDD_LOC_DATASET    = <Name of EDD SAS Dataset>,
                    LIB_IN             = <Library of input dataset>,
                    DATA_IN            = <Name of input dataset>,
                    LIB_OUT            = <Library of output dataset(outlier treated dataset)>,
                    DATA_OUT           = <Name of output dataset i.e.  Outlier treated dataset>,
                    NO_TREAT_VARLIST   = <VARLIST for no outlier treatment (separated by space)>,
                    UNIQUE_ID          = <Unique identifier of input dataset>,
                    X_TIMES_CAP        = <Outlier factor on P99 side>,
                    X_TIMES_FLOOR      = <Outlier factor on P1 side>
                    );
```

This macro detects and treats the outlier values using P99 and P1 for numeric variables.

FLOORING

CASES	LOGIC				ILLUSTRATION (Let X = 5)		
	MIN	P1	OUTLIER	TREATMENT	MIN	P1	TREATMENT
CASE I	< 0	< 0	Any value < X * P1	Floor at X * P1	- 200	- 10	Floor at -50
CASE II	< 0	= 0	Any value < - X	Floor at - X	- 200	0	Floor at -5
CASE III	< 0	> 0	Any value < P1 – (X * P1)	Floor at P1 – (X * P1)	- 200	10	Floor at -40
CASE IV	> 0	> 0	Any value < P1 / X	Floor at P1 / X	1	10	Floor at 2

CAPPING

CASES	LOGIC				ILLUSTRATION (Let X = 5)		
	P99	MAX	OUTLIER	TREATMENT	P99	MAX	TREATMENT
CASE I	> 0	> 0	Any value > X * P99	Cap at X * P99	10	200	Cap at 50
CASE II	= 0	> 0	Any value > X	Cap at X	0	200	Cap at 5
CASE III	< 0	> 0	Any value > P99 - (X * P99)	Cap at P99 - (X * P99)	- 10	200	Cap at 40
CASE IV	< 0	< 0	Any value > P99 / X	Cap at P99 / X	- 10	- 1	Cap at -2

A.2. Macro Call: Exponential Smoothing

Exponential Smoothing Syntax

```
%OUTL_TREATMENT_EXP_SMOOTH
           (
           EDD_LOC_LIB       = <Library where EDD in form of SAS dataset is located>,
           EDD_LOC_DATASET   = <Name of EDD SAS Dataset>,
           LIB_IN            = <Library of input dataset>,
           DATA_IN           = <Name of input dataset>,
           LIB_OUT           = <Library of output dataset(outlier treated dataset)>,
           DATA_OUT          = <Name of output dataset i.e.  Outlier treated dataset>,
           NO_TREAT_VARLIST  = <VARLIST for no outlier treatment (separated by space)>,
           UNIQUE_ID         = <Unique identifier of input dataset>,
           X_TIMES_CAP       = <Outlier factor on P99 side>,
           X_TIMES_FLOOR     = <Outlier factor on P1 side>
           );
```

This macro detects and treats the outlier values using exponential smoothing technique.

In this technique, the curve between P95 to P99 is extrapolated beyond P99, to identify the values falling above the curve. The values falling outside the curve are outliers and are treated according to some functions depending upon the boundary conditions.

Advantages

- Rank order is maintained
- Distribution of data is taken into account while identifying the outliers
- Run time is less

Disadvantages

- Functions involved in treating the outliers are quite complex

A.3. Macro Call: Sigma Approach

Sigma Approach Macro Syntax

```
%OUTL_TREATMENT_SIGMA
          (
          EDD_LOC_LIB       = <Library where EDD in form of SAS dataset is located>,
          EDD_LOC_DATASET   = <Name of EDD SAS Dataset>,
          LIB_IN            = <Library of input dataset>,
          DATA_IN           = <Name of input dataset>,
          LIB_OUT           = <Library of output dataset(outlier treated dataset)>,
          DATA_OUT          = <Name of output dataset i.e.  Outlier treated dataset>,
          NO_TREAT_VARLIST  = <VARLIST for no outlier treatment (separated by space)>,
          UNIQUE_ID         = <Unique identifier of input dataset>,
          X_TIMES_CAP       = <Outlier factor on P99 side>,
          X_TIMES_FLOOR     = <Outlier factor on P1 side>
          );
```

This macro detects and treats the outlier values using sigma approach. In this technique, the outliers are identified and treated based upon the values of mean and standard deviation (sigma). The macro uses simple boundary condition to check for the outliers.

Capping: Any value greater than (mean + X_TIMES_CAP * sigma) is an outlier
Imputed value = (mean + X_TIMES_CAP * sigma)

Flooring: Any value less than (mean – X_TIMES_FLOOR * sigma) is an outlier
Imputed value = (mean – X_TIMES_FLOOR * sigma)

Advantages

- Easy to understand & implement
- Run time is less

Disadvantages

- Rank order is not maintained
- This method works best only when variables follow a normal distribution

A.4. Macro Call: Robust Regression

Robust Regression Macro Syntax

```
%ROBUSTREG_OUTLIER
           (
           IN_DATA   = <Library and name of input dataset>,
           OUT_DATA1 = <Library and name of output dataset with info. on identified outlier records>,
           OUT_DATA2 = <Library and name of output dataset after removing outlier records>,
           OUT_DATA3 = <Library and name of output dataset with outlier records>,
           VAR_LIST  = <VARLIST for no outlier treatment (separated by space)>,
           DEP_VAR   = <Name of dependent variable>,
           UNIQUE_ID = <Unique identifier of input dataset>
           );
```

This macro detects and treats outliers by using ROBUSTREG procedure in SAS.

Outlier Identification:

- This technique involves running regression repeatedly to identify outliers by assigning weights to the observations. The weights are on the basis of the prediction error (residual) in different iterations. **Higher residual means lower weight.**

Outlier treatment:

- Observations with zero weight are marked as outliers and need to be removed from the data for any kind of analysis

Advantages

- Effect of outliers on model performance is minimized

Disadvantages

- Ignoring outliers may result in loss of data/information.
- Computation of robust estimates is resource intensive. It takes ~30 minutes for running on 600 variables but for 2000 variables it takes ~100hrs (4days)

A.5. Macro Call: Mahalanobis Distance

Mahalanobis Distance Macro Syntax

```
%OUTLIER_MD (
            IN_DATA      = <Library and name of input dataset>,
            VAR          = <VARLIST for no outlier treatment (separated by space)>,
            UNIQUE_ID    = <Unique identifier of input dataset>,
            OUT_DATA     = <Library and name of output dataset (dataset with weight for OUTLIERS)>
            );
```

This macro detects outliers using Mahalanobis distance approach. This macro creates a weight variable (OT_WT) which, when used in regression, reduces the effect of outliers.

In this technique, the outliers are identified by the magnitude of "Mahalanobis" or statistical distance from the origin. To each observation, weight is given as the inverse of "Mahalanobis" distance.

Outlier Identification

- The observation with extremely low weights can be considered to be outliers

Outlier Treatment

- Weighted regression is run to take into account the effect of weight given to each observation. (Greater the "Mahalanobis" distance, lesser is the weight of that observation & hence lesser the contribution of that observation in final model.)

Advantages

- Effect of outliers on model performance is minimized

Disadvantages

- A minor change in data distribution would lead to more than normal deterioration of the model performance
- Complexity in calculation of Mahalanobis distance for weighted regression

A.6. Data Collection Process: Details

Identify Data Needs

- Start with Business Question
- Determine data need for delivering desired outcome

Illustration:

Business Question:
How to match the most profitable credit product with each new customer?

Solution:
Use Credit & Payment History and Financial Statement data to predict account performance for different products.

Data Request:
A representative sample of customers from each product with usage and payment data for sufficient no. of months along with their credit score

Data Mapping

Become as familiar as possible with the data sources and their content

Data Mapping has basically three major components:

- Interview clients
- Obtain & study data layouts
- Obtain & evaluate data samples

Note:
The results of each step may require us to repeat one or more previous steps.

Plan Data Request

Assess Available Population Coverage

- Data availability constraints; viz. archives time span
- Population sizing by key characteristics like credit history
- Discuss with client unexpected size limitations

Assess Alternative Data Sources

- Choose between alternatives
- Ensure that link keys work between sources chosen

Plan to Optimize Client Resource Use

Minimize workload for client IT department; even if it makes more work at our end to link files, convert media, reformat etc.

Send Data Request

Be as specific as possible!

- Accurate file names
- Specify selection criteria with respect to actual field names and value formats (e.g. "Values of the field STATE_CD in the subset = (IN,MI)" rather than "Records from Indiana and Michigan")
- Specify required or acceptable file formats
- Give detailed randomization and/or stratification

Note:
In case of Account x Transaction level data, random sampling of records is not the same as random sampling of accounts.

- Prepare the driver file

Quality Check

Always examine results before acceptance!

For each data file received,

- Compare basic statistics (no. of records, no. of fields, range of values in each field) to expectations and resolve discrepancies
- Ensure that delimiters, file format and record format meet requirements
- Ensure that the data dictionary matches the file exactly
- Enter file into data inventory, recording basic information (file name, date received, file size, record length, SPOC)

While merging files,

- Watch out for identical merge-key field name with different meanings in two files
- Beware of the consequences of merging two datasets with few identically named non-key fields
- Specify a distinct output file for sorting

Thanks

For queries, contact Varun Aggarwal at Varun.Aggarwal@exlservice.com