Weight of Evidence (WoE)
and
Information Value (IV)

Contents

Handling Categorical Variables In Statistical Models

- In classification or regression problems independent variables can either be continuous or categorical. Categorical variable can have limited number of categories or many categories as shown in the following table.

Type	Example
Variable with limited categories	Age Groups: Below 15 = 1, 15-25 = 2, Above 25 = 3
Variables with large number of categories	City or Country will have many levels

- Generally, when independent variables in statistical models are categorical, they are replaced and represented by Dummy Variables.
- If there are k categorical variables, then they are represented by k-1 Dummy Variables.

Weight of Evidence (WoE)

- Weight of Evidence (WoE) estimates the **predictive power of an independent variable in relation to the dependent variable.**
- WoE is originally used in credit risk analytics, as the method of separation of "good" and "bad" customers (Non-defaulters: Y=0 and Defaulters: Y=1)
- WoE is defined as

$$\ln\left(\frac{\text{Distribution of Good}_i}{\text{Distribution of Bad}_i}\right)$$

- Here, Distribution of Good **is the proportion of good customers in a category to total good customers**. Similarly, Distribution of Bad **is the proportion of bad customers in a category to total bad customers**.
- Using WoE ,we can assign continuous value for each category. For instance, if there are 50 cities then there will be 50 WoE values.

Get an Edge!

Some thumb rules related to Weight of Evidence

- Each category (bin) should have at least 5% of the observations.
- Each category (bin) should be non-zero for both non-events and events.
- The WoE should be distinct for each category. Similar groups should be aggregated.
- The WoE should be monotonic, i.e. either growing or decreasing with the groupings (Not applicable when groups are for character strings).
- Missing values are binned separately.

Information Value (IV)

- Information Value (IV) is a highly useful tool for variable selection.
- The concept has its roots in entropy in information theory.
- IV of an independent variable **expresses the amount of diagnostic information of that variable for separating the Goods from the Bads.**
- IV is calculated as

$$\sum \left(\text{Distribution of Good}_i - \text{Distribution of Bad}_i\right) \times \ln\left(\frac{\text{Distribution of Good}_i}{\text{Distribution of Bad}_i}\right)$$

- IV helps in ranking variables based on their importance.

Weight of Evidence

Information Value (IV)

By convention, information values can be interpreted as follows:

Value	Predictive
< 0.02	Not useful for prediction
0.02 to 0.1	Weak predictor
0.1 to 0.3	Medium predictor
0.3 to 0.5	Strong predictor
> 0.5	Suspicious predictive power

Case Study – Predicting Loan Defaulters

Background

- The bank possesses demographic and transactional data of its loan customers. If the bank has a robust model to predict defaulters it can undertake better resource allocation.

Objective

- To predict whether the customer applying for the loan will be a defaulter.

Available Information

- Sample size is 700
- Independent Variables: Age group, Town, Years at current address, Years at current employer, Debt to Income Ratio, Credit to Debit ratio, Other Debts
- Dependent Variables: Defaulter (=1 if defaulter, 0 otherwise)

Data Snapshot

BANK LOAN WOE-IV

Independent Variables

Dependent Variable

SN	AGE	TOWN	EMPLOY	ADDRESS	DEBTINC	CREDDEBT	OTHDEBT	DEFAULTER
1	3	Mumbai	17	12	9.3	11.36	5.01	1

Observations

Column	Description	Type	Measurement	Possible Values
SN	Serial Number	Numeric	-	-
AGE	Age Groups	Categorical	1(<28 years),2(28-40 years),3(>40 years)	3
TOWN	Customer Belonging to Which Town	Categorical	Mumbai, Delhi,etc..	15
EMPLOY	Number of years customer working at current employer	Continuous	-	Positive value
ADDRESS	Number of years customer staying at current address	Continuous	-	Positive value
DEBTINC	Debt to Income Ratio	Continuous	-	Positive value
CREDDEBT	Credit to Debit Ratio	Continuous	-	Positive value
OTHDEBT	Other Debt	Continuous	-	Positive value
DEFAULTER	Whether customer defaulted on loan	Binary	1(Defaulter), 0(Non-Defaulter)	2

WoE and IV in R

```
# Import the data
```

```
data<-read.csv("BANK LOAN WOE-IV.csv",header=T)
head(data)
str(data)
```

```
# Convert AGE to Factor
```

```
data$AGE<-as.factor(data$AGE)
```

- ❑ **read.csv**() is used to import csv file.
- ❑ **str**() shows class and levels of variables in the data.
- ❑ AGE is actually a categorical variable but represented numerically. We will convert it to factor using **as.factor**() and then calculate WoE and IV for AGE.

WoE and IV in R

```
# Output:
```

```
> data<-read.csv(file.choose())
> str(data)
'data.frame':   700 obs. of  9 variables:
 $ SN      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ AGE     : int  3 1 2 3 1 3 2 3 1 2 ...
 $ TOWN    : Factor w/ 15 levels "Ahmedabad","Bengaluru",..: 12 4 2 5 1 3 10 15 14 7 ...
 $ EMPLOY  : int  17 10 15 15 2 5 20 12 3 0 ...
 $ ADDRESS : int  12 6 14 14 0 5 9 11 4 13 ...
 $ DEBTINC : num  9.3 17.3 5.5 2.9 17.3 10.2 30.6 3.6 24.4 19.7 ...
 $ CREDDEBT : num  11.36 1.36 0.86 2.66 1.79 ...
 $ OTHDEBT : num  5.01 4 2.17 0.82 3.06 ...
 $ DEFAULTER: int  1 0 0 0 1 0 0 0 1 0 ...
> data$AGE<-as.factor(data$AGE)
> str(data)
'data.frame':   700 obs. of  9 variables:
 $ SN      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ AGE     : Factor w/ 3 levels "1","2","3": 3 1 2 3 1 3 2 3 1 2 ...
 $ TOWN    : Factor w/ 15 levels "Ahmedabad","Bengaluru",..: 12 4 2 5 1 3 10 15 14 7 ...
 $ EMPLOY  : int  17 10 15 15 2 5 20 12 3 0 ...
 $ ADDRESS : int  12 6 14 14 0 5 9 11 4 13 ...
 $ DEBTINC : num  9.3 17.3 5.5 2.9 17.3 10.2 30.6 3.6 24.4 19.7 ...
 $ CREDDEBT : num  11.36 1.36 0.86 2.66 1.79 ...
 $ OTHDEBT : num  5.01 4 2.17 0.82 3.06 ...
 $ DEFAULTER: int  1 0 0 0 1 0 0 0 1 0 ...
```

Interpretation:

- Age initially as integer is converted to factor with 3 levels 1,2,3.

WoE and IV in R

```
# Install and load package "Information"
```

```
install.packages("Information")
library(Information)
```

- The **Information** package is designed to perform exploratory data analysis and variable screening for binary classification models using WOE and IV. The package is specifically designed to perform data exploration by producing easy-to-read tables and graphs.

```
# Changing Binary Values for Defaulter
```

```
data$DEFAULTERNEW = 1-data$DEFAULTER
```

- Packages for computing WoE and IV consider binary value 0 to be ‘bad’ and 1 to be ‘good’. However, in our data, (and as a general practice) 1 represents occurrence of an event and 0 otherwise. It is imperative to remember this before calculating WoE and IV tables.

WoE and IV in R

```
# Calculate WoE and IV
```

```
IV <- create_infotables(data=data, y="DEFAULTERNEW")
```

- create_infotable generates WoE and IV for all variables in the data except dependent variable which is specified as **"y="**.

```
# Get WoE and IV values for 'AGE' variable
```

```
woe_age<-as.data.frame(IV$Tables$AGE)
woe_age
```

- **create_infotables()** returns WOE tables as data.frames, and a data.frame with IV values for all predictive variables.
- IV$Tables$'predictor variable name' is created to store WoE and IV values in a dataframe which are used for further analysis.

WoE and IV in R

```
# Output :
```

```
> woe_age
  AGE   N   Percent        WOE          IV
1   1 242 0.3457143 -0.4430480 0.07452269
2   2 284 0.4057143  0.2577412 0.09978166
3   3 174 0.2485714  0.3051780 0.12120615
```

Interpretation:

- Output table contains categories of the variable, count and percent of observations for each category, WoE and IV values.

Appending WoE Values to Original Data

```
# Check the type of key variable before merging
```

```
str(woe_age)
woe_age$AGE<-as.factor(woe_age$AGE)
str(woe_age)
```

- **‘Age’** is the common variable in the original data and WoE data.
- Type of ‘Age’ in both data should be common for merging datasets.]

```
# Output :
```

```
> str(woe_age)
'data.frame':	3 obs. of  5 variables:
 $ AGE    : chr  "1" "2" "3"
 $ N      : num  242 284 174
 $ Percent: num  0.346 0.406 0.249
 $ WOE    : num  -0.443 0.258 0.305
 $ IV     : num  0.0745 0.0998 0.1212
> woe_age$AGE<-as.factor(woe_age$AGE)
> str(woe_age)
'data.frame':	3 obs. of  5 variables:
 $ AGE    : Factor w/ 3 levels "1","2","3": 1 2 3
 $ N      : num  242 284 174
 $ Percent: num  0.346 0.406 0.249
 $ WOE    : num  -0.443 0.258 0.305
 $ IV     : num  0.0745 0.0998 0.1212
```

Appending WoE Values to Original Data

```
# Merging the datasets
```

```
leftjoin<-merge(data,woe_age,by="AGE", all.x = TRUE)
head(leftjoin)
```

- **merge()** with **all.x=TRUE** returns data with all rows from left table (here, data) and any rows with matching keys from the right table (here, woe_age).

```
# Output :
```

```
> head(leftjoin)
  AGE  SN      TOWN EMPLOY ADDRESS DEBTINC CREDDEBT OTHDEBT DEFAULTER DEFAULTERNEW   N   Percent       WOE         IV
1   1 523     Kochi      4       7     4.1     0.29    0.49         0            1 242 0.3457143 -0.443048 0.07452269
2   1 376     Kochi      1       4     2.5     0.13    0.29         0            1 242 0.3457143 -0.443048 0.07452269
3   1  39    Jaipur      1       8    17.1     1.34    2.77         1            0 242 0.3457143 -0.443048 0.07452269
4   1 201 Ahmedabad      3       7     4.1     0.26    0.52         0            1 242 0.3457143 -0.443048 0.07452269
5   1 245   Kolkata      3       4    13.3     1.60    3.05         0            1 242 0.3457143 -0.443048 0.07452269
6   1  46    Kanpur      0       1     6.8     0.15    0.94         0            1 242 0.3457143 -0.443048 0.07452269
```

Binary Logistic Model

```
# Binary Logistic Model with AGE as FACTOR
```

```
riskmodel1<-glm(DEFAULTER~AGE+EMPLOY+ADDRESS+DEBTINC+
                 CREDDEBT+OTHDEBT,
               family=binomial,data=data)

summary(riskmodel1)
```

```
# Binary Logistic Model using ‘WOE’ as a predictor instead of ‘AGE’
```

```
riskmodel2<-glm(DEFAULTER~WOE+EMPLOY+ADDRESS+DEBTINC+
                 CREDDEBT+OTHDEBT,
               family=binomial,data=leftjoin)

summary(riskmodel2)
```

Binary Logistic Model

Output :

```
> summary(riskmodel1)

Call:
glm(formula = DEFAULTER ~ AGE + EMPLOY + ADDRESS + DEBTINC +
    CREDDEBT + OTHDEBT, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3495  -0.6601  -0.2974   0.2509   2.8583

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.78821    0.26407  -2.985  0.00284 **
AGE2         0.25202    0.26651   0.946  0.34433
AGE3         0.62707    0.36056   1.739  0.08201 .
EMPLOY      -0.26172    0.03188  -8.211  < 2e-16 ***
ADDRESS     -0.09964    0.02234  -4.459 8.22e-06 ***
DEBTINC      0.08506    0.02212   3.845  0.00012 ***
CREDDEBT     0.56336    0.08877   6.347 2.20e-10 ***
OTHDEBT      0.02315    0.05709   0.405  0.68517
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 804.36  on 699  degrees of freedom
Residual deviance: 553.41  on 692  degrees of freedom
AIC: 569.41

Number of Fisher Scoring iterations: 6
```

Interpretation:

- Model with Age as factor creates 2 dummy variables.
- P values for both dummy variables are greater than 0.05. Therefore, the impact of AGE is statistically insignificant.

Binary Logistic Model

Output :

```
> summary(riskmodel2)

Call:
glm(formula = DEFAULTER ~ WOE + EMPLOY + ADDRESS + DEBTINC +
    CREDDEBT + OTHDEBT, family = binomial, data = leftjoin)

Deviance Residuals:
    Min       1Q    Median       3Q      Max
-2.3634  -0.6484  -0.3069   0.2472   2.9116

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.58887    0.28711  -2.051 0.040263 *
WOE          0.48221    0.36301   1.328 0.184048
EMPLOY      -0.26104    0.03187  -8.190 2.62e-16 ***
ADDRESS     -0.09535    0.02205  -4.325 1.53e-05 ***
DEBTINC      0.08242    0.02197   3.752 0.000176 ***
CREDDEBT     0.57151    0.08857   6.452 1.10e-10 ***
OTHDEBT      0.02922    0.05665   0.516 0.606014
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 804.36  on 699  degrees of freedom
Residual deviance: 554.64  on 693  degrees of freedom
AIC: 568.64

Number of Fisher Scoring iterations: 6
```

Interpretation:

- Model with WOE as a predictor gives one P value
- The P value for WOE is greater than 0.05. Therefore, the impact of AGE is statistically insignificant.

WoE of Character Variable

```
# WoE and IV for variable 'TOWN'
```

```
IV$Tables$TOWN
```

```
# Output :
```

```
> IV$Tables$TOWN
        TOWN  N    Percent         WOE         IV
1  Ahmedabad 34 0.04857143 -0.80216794 0.03627141
2  Bengaluru 35 0.05000000  0.17783860 0.03778456
3    Chennai 33 0.04714286 -0.20564760 0.03987340
4      Delhi 36 0.05142857  0.78599257 0.06552734
5  Hyderabad 40 0.05714286  0.34773764 0.07184911
6     Indore 39 0.05571429  0.16541608 0.07331250
7     Jaipur 40 0.05714286 -0.19125886 0.07549575
8     Kanpur 52 0.07428571  0.16541608 0.07744694
9      Kochi 66 0.09428571 -0.13284810 0.07916282
10   Kolkata 63 0.09000000 -0.27308888 0.08629521
11   Lucknow 51 0.07285714 -0.06669614 0.08662442
12    Mumbai 58 0.08285714 -0.24004903 0.09166334
13    Nagpur 59 0.08428571  0.12904844 0.09302324
14      Pune 50 0.07142857  0.22710965 0.09650390
15     Surat 44 0.06285714  0.46552068 0.10856864
```

WoE of Numeric Variable

```
# WoE and IV for variable 'EMPLOY'
IV$Tables$EMPLOY
```

- By default, **create_infotables()** categorizes numeric variable into 10 bins. You can change the number of bins by specifying bins='n' in the function.

```
# Output :
```

```
> IV$Tables$EMPLOY
    EMPLOY  N    Percent        WOE        IV
1    [0,0] 62 0.08857143 -1.1030952 0.1288816
2    [1,1] 49 0.07000000 -0.5817983 0.1555268
3    [2,3] 86 0.12285714 -0.9920367 0.2987787
4    [4,4] 47 0.06714286 -0.1811065 0.3010739
5    [5,6] 82 0.11714286  0.0277947 0.3011638
6    [7,8] 69 0.09857143  0.6239910 0.3336590
7   [9,10] 75 0.10714286  0.2663920 0.3407685
8  [11,13] 83 0.11857143  0.6449892 0.3822789
9  [14,17] 70 0.10000000  0.8750926 0.4424923
10 [18,31] 77 0.11000000  1.4323637 0.5922371
```

WoE of Numeric Variable

```
# WoE and IV for variable 'EMPLOY' with 3 bins
```

```
IV <- create_infotables(data=data, y="DEFAULTERNEW", bins = 3)
IV$Tables$EMPLOY
```

```
# Output :
```

```
> IV$Tables$EMPLOY
   EMPLOY   N   Percent        WOE        IV
1   [0,3] 197 0.2814286 -0.9267653 0.2845505
2   [4,9] 243 0.3471429  0.1441387 0.2915113
3 [10,31] 260 0.3714286  0.8898857 0.5217636
```

Information Value (IV) Interpretation

```
# Extracting IV for all predictor variables
IV <- create_infotables(data=data, y="DEFAULTERNEW")

IV_Value = data.frame(IV$Summary)
IV_Value
```

- **create_infotable** generates Tables and Summary objects.
- **Tables** object used earlier to extract WoE and IV for individual variables.
- **Summary** object contains IV for all predictor variables.

Information Value Interpretation

```
# Output :
```

```
> IV_Value
   Variable        IV
6   DEBTINC 0.7871927
4    EMPLOY 0.5922371
5   ADDRESS 0.3359295
7  CREDDEBT 0.2835522
8   OTHDEBT 0.1453887
2       AGE 0.1212061
3      TOWN 0.1085686
1        SN 0.0424855
9 DEFAULTER 0.0000000
```

Interpretation:

- We will not consider IV for SN and DEFAULTER as they are not the predictor variables.
- With the help of table '**IV values and its Predictive Power**' on slide number 8 we can say that,
  - Town and Age are weak predictor.
  - Othdebt, Creddebt, Address have medium predictive power.
  - Employ and Debtinc are strong predictor.

Quick Recap

In this session, we learnt how to **compute and use Weight of Evidence and Information Value:**

Weight of Evidence	• Tells the predictive power of an independent variable in relation to the dependent variable • $\ln((\text{Distribution of Good}_i)/(\text{Distribution of Bad}_i))$
Information Value	• Expresses the amount of diagnostic information of that variable for separating the Goods from the Bads • $\sum(\text{Distribution of Good}_i - \text{Distribution of Bad}_i) \times \text{WoE}$
Weight of Evidence and Information Value in R	• Package "**Information**" in R contains functions for calculating weights of evidence, information value. • Function **`create_infotables()`** generates Tables and Summary objects. • Tables object used to extract WoE and IV of individual variable . • Summary object gives list of variables and its corresponding IV.