Information theoretic measures in predictive modeling

Sandeep Rajput
Infosys Corporation

June 11, 2017

Abstract

This document provides a very brief introduction to Information value, which is an information-theoretic measure used in predictive modeling.

1 Introduction

This discussion is limited to the binary classification modeling scenario, where the modeled or predicted variable (also referred to as *label*) is represented by Y which only takes the value 0 or 1. The objective is to create a model that returns a number between 0 and 1, with input X known as independent or predictor variable. It is desirable that the model output be small for the cases where Y is 0 and large for where Y is 1. This is mathematically represented as $\hat{y} = P(Y = 1 \mid X = x)$. Generally the events denoted by $Y = 1$ are of interest and relatively rare. For that reason, events or observations with $Y = 1$ are called **targets** and those with $Y = 0$ are called **non-targets**.

It is assumed that the model will be used to produce an output on observations where X **is known**, and Y **unknown**. This output will be used to make a decision, such as blocking a wire transfer request because the risk of fraud – as determined by the fraud model output – is too high.

For this discussion we assume that the *range* of X is spanned by disjoint sets a_k for the collection of such sets $\{a_k\}$ for $k = 1, 2, \ldots, K$ is a *complete cover*. In this paper we will refer to index k as the *level* of X for brevity sake. We will also use $p_k = P(Y = 1 \mid x = a_k)$ to define the model output for level k.

Since the goal of binary classification is to *discriminate* between two categories or classes, a level k is *informative* if it has the *odds* very different from unity **or** *log-odds* very different from zero. The *log-odds* in favor of $Y = 1$ are defined as the logarithm of the ratio $p_k/(1 - p_k)$

Through application of Bayes' theorem and substituting terms, we can obtain

$$\ln\left\{\frac{P(Y=1 \mid X \in a_k)}{P(Y=0 \mid X \in a_k)}\right\} = \ln\left(\frac{p_k}{1-p_k}\right) = \ln\left\{\frac{P(X =\in a_k \mid Y=1)}{P(X \in a_k \mid Y=0)}\right\} \underbrace{\frac{P(Y=1)}{P(Y=0)}}_{\text{Fixed}} \tag{1}$$

For a given population, $P(Y = 1)$ and $P(Y = 0)$ are fixed, and the predictive information is present only in the first term on RHS.

To simplify the notation, let the fraction of targets in level k be $t_k = P(X = a_k \mid Y = 1)$, and the fraction of non-targets be $z_k = P(X = a_k \mid Y = 0)$. These values can also be remembered as *column percentages*. For ease of exposition, let's assume $t_k, z_k > 0$ for all k

This reduces the equation to

$$\ln\frac{p_k}{1-p_k} = \ln\frac{t_k}{z_k} + \text{Const} \tag{2}$$

1.1 Weight of Evidence

The *weight of evidence* or WoE for level k is given as

$$w_k = \ln \frac{t_k}{z_k} = \ln t_k - \ln z_k \tag{3}$$

This value is 0 if and only if $z_k = t_k$ or the group k contains identical percentage of targets as non-targets. It follows that the level is *uninformative*. If the group contains relatively more targets than non-targets, $t_k > z_k$ and $w_k > 0$. Conversely $w_k < 0$ if $t_k < z_k$. Large absolute values of WoE have discriminatory information that can be used for predictive purposes.

1.2 Information Value

A large absolute value of WoE is informative; however its utility in predictive modeling depends on how many observations actually occur with that group. That is addressed by defining the *Information Value* of which there are many definitions. Here we provide a popular one that always produces positive values for each group. The reader may verify that the information value due to group k is $v_k > 0$ defined as

$$v_k = (t_k - z_k)w_k = (t_k - z_k)(\ln t_k - \ln z_k) \tag{4}$$

The overall information value due to the variable X is then

$$\mathrm{IV}(Y||X) = \sum_{k=1}^{K} v_k = \underbrace{-\sum_{k=1}^{K}(z_k \ln t_k + t_k \ln z_k)}_{\text{Cross-entropy}} - \underbrace{\Big(-\sum_{k=1}^{K} t_k \ln t_k\Big)}_{H_t} - \underbrace{\Big(-\sum_{k=1}^{K} z_k \ln z_k\Big)}_{H_z} \tag{5}$$

The last two terms are the Shannon entropy[1] of the distributions of targets and non-targets respectively, whereas the first summation defines the *cross-entropy* between $\{z_k\}$ and $\{t_k\}$ in a symmetric fashion. If there is a lot of *discordance* between $\{t_k\}$ and $\{z_k\}$, the cross-entropy is high. Information Value (IV) as defined above is never negative. It is zero if and only if $z_k = t_k$ for all k. The more distinct $\{t_k\}$ and $\{z_k\}$ distributions, the higher the information value. The higher the IV, the more predictive the variable.

1.3 How to use Information Value?

Information Value (IV) is a valuable tool for several purposes explained below

Pre-screen variables Information Value is a great tool to rank order variables. Variables having IV less than 0.05 rarely turn out to be predictive, whereas a variable having IV greater than 1.0 often does. However, usually a very large IV such as 5.0 or greater signifies an error or that the problem in question is rather trivial.

Condense variable groups Information Value can also be used to choose levels or groups within a variable or to combine multiple levels for greater robustness unaccompanied by a large decrease in IV. This is the central concept behind Decision Trees and in a fashion behind CHAID, CART and other such algorithms that deal well with mixed data (not only numeric values).

[1] Shannon entropy is a measure of the *randomness* in the distribution of a random variable. It is highest for a random uniform distribution and tends to zero where only one value is possible (degenerate distributions)

Present Evidence WoE and IV values together make a useful combination to display the preliminary results of variable discovery to a non-technical audience. With some skill on part of the data scientist, the non-technical audience can appreciate how a certain level for a certain variable is highly predictive or not very predictive. That allows them to confirm the findings or raise serious doubts early on in the model development process. In any case, such interaction allows data scientists to take qualitative input from the business or operations functions long before the model development is concluded.

1.4 Best Practices

1.4.1 Technical

If there are many groupings containing only targets or only non-targets, Information value is overstated. IV is a reliable measure if every grouping contains at least 1% of both classes or at least 5 of each class – whichever is **greater**. This requirement avoids learning small artifacts of the available dataset that won't generalize.

1.4.2 Beware of Default choices

Information Value is easy to code and most data science toolkits provide the means to do so. However, the reader is advised to pay attention to the grouping of individual variables before computing the IV and not let the software packages do the default grouping that is sometimes deceptive, unintuitive and often unoptimal. That is particularly avoidable when strong domain knowledge is available.

1.4.3 Avoid Imputation

If the value of `NULL` or `-99999` has high discriminatory power and for a good reason (matching record not found in the master database, say), simply imputing with zero, mean or median will lose the discriminatory power and lead to a large decrease in IV – without the data scientist realizing these behind-the-scenes default choices. For this reason, the variable *groupings* should be abstracted out as a configuration in cases where good domain knowledge is available or where compliance and governance requirements necessitate transparency.

2 Relatioship with Chi-square test

The Chi-square (also known as χ^2) test is also used to compare count data or frequencies. Using the same notation as in section 1, the task is to compare the set of observed counts $\{O_k\}$ against the expected counts $\{E_k\}$ and determine whether the observed counts are unlikely to have been genreated due to pure luck. In layman terms, that's called statistical significance testing that generates a *p-value*. If that *p-value* is small enough, we take that to mean that the observed counts are different from the expected values.

The test statistic is given as

$$c = \sum_k^K \frac{(O_k - E_k)^2}{E_k} \tag{6}$$

The test statistic follows the chi-square distribution with $K - 1$ degrees of freedom or is distributed as $c \sim \chi^2_{K-1}$, which is the distribution under the Null Hypothesis[2]. The corresponding *p-value* is defined as the complement of the cumulative distribution of the test-statistic at c – or the fraction of values no less than c.

Consider a comparison of two counts at 5 levels. The distribution under null hypothesis is χ^2_4 or chi-square with 4 degrees of freedom. If the test statistic is 6.4574, the *p-value* is 0.1675, which is not small enough to be evidence for distributions being different. If the test statistic is twice as much, or 12.9148, the *p-value* comes to 0.010117 which is small enough for us to serve as evidence that observed counts are significantly different than expected counts.

2.1 Chi-square test for variable importance

For the binary classification task discussed above, chi-square test can be used to determine the importance of a variable in discriminating *targets* from *non-targets*. We can assume the count of non-targets as the *expected* and scale them so their total matches the total count of targets, so both counts sum up to the same number – let's call it N.

In that case we can write $E_k = N\, z_k$ and $O_k = N\, t_k$. With this the test statistic reduces to:

$$c = \sum_k^K \frac{(O_k - E_k)^2}{E_k} = \sum_k^K \frac{N(t_k - z_k)^2}{z_k} \tag{7}$$

2.2 Notes and Observations

The test statistic scales linearly with the total count N. With large enough N and $N \gg K$, even small differences will be found statistically significant. The example in the previous section illustrates that clearly, and this is a major known flaw of using a parametric test such as the chi-square test.

Writing the test statistic slightly differently to contrast it with Information value, we have

$$c = N \sum_k^K (t_k - z_k) \left(\frac{t_k}{z_k} - 1 \right) \tag{8}$$

whereas the information value is defined as

$$IV(Y||X) = \sum_k^K (t_k - z_k) \ln \frac{t_k}{z_k} \tag{9}$$

[2]Null Hypothesis posits that there is no difference between observed and expected counts

Other than the factor of N, the only difference is how $t_k - z_k$ is weighed. The weighting in Information value is *symmetric*, meaning the value does not change when we swap $\{t_k\}$ and $\{z_k\}$. The weighting factor in the test statistic does not have that property. Instead, it assigns great importance to levels where $t_k \gg z_k$ but little importance to levels where $t_k \ll z_k$.

The purpose of the chi-square statistic is allow hypothesis testing and not create a consistent metric or statistic for the population in general. As long as the chi-squre test is significant, a value of 100 or 1,000 or 10,000 rarely translates to better predictive power. But with Information Value a larger value often translates to better predictive power.

Chi-square test or statistic rarely reveals *which level* has predictive information, unlike *weight of evidence* or IV for that variable level.

3 Computing Information Value with Software Packages

Listing 1 contains R code to compute the Information Value as defined in this paper. This R code does not require any libraries other than what's present in **base R** distribution. The code is prolix by design to keep the logic flow clear. Usage and output are shown in Listing 2.

```r
# ------------------------------------------------------------------------------ #
#  Computes Information Value of a variable with respect to a binary variable
#
#  Usage:  iv.out <- infValue( x, y, verb )
#  =======
#
#       x  [ Vector  ] : Contains "predictor" or "input" variable
#       y  [ Vector  ] : Contains INTEGER binary "label" of 0 or 1
#     verb [ Boolean ] : If TRUE, details are printed to STDOUT
#
#  NOTES:
#  =======
#  1) The output is the Information Value of x in predicting y
#  2) x and y MUST BE the same length
#  3) Information Value is defined as \sum_{i}{ (f_i - g_i ) (\ln f_i - \ln g_i) }
#     where f_i and g_i are the distributions of x for y values of 0 and 1
#  4) Groupings with zero counts for y values of 0 or 1 are ignored
#  5) NA values are ignored ( which is the default for table() in R )
#
#         Author: Infosys Data Science Practice
#        Version: 0.5.2
#   Last Updated: 2017-06-25 17:56 PDT by SR
# ------------------------------------------------------------------------------ #
infValue <- function( x, y, verb )
{
    table.raw              <- table( Grouping=x, y )
    colnames( table.raw )  <- c( 'Non-Targets', 'Targets' )

    table.cpct             <- prop.table(table.raw,2);               #  Column percentages
    colnames(table.cpct)   <- c('Pct. Non-Targets','Pct. Targets')

    #   Compute Weight of Evidence (WoE) and IV contributions
    # ------------------------------------------------------------------
    temp.woe     <- ifelse(table.cpct[,1] * table.cpct[,2] > 0, log(table.cpct[,2]/table.cpct[,1]),
     ↪  0.0);
    table.final  <- cbind( table.raw, WoE=temp.woe  );
    table.final  <- cbind( table.final, PctClassDiff=(table.cpct[,2]-table.cpct[,1]) );
    table.final  <- cbind( table.final, IVcontrib=table.final[,'PctClassDiff']*table.final[,'WoE'] );

    iv.out       <- sum( table.final[,'IVcontrib'] );    #  Compute the IV

    if( verb ) {
        print( table.final );
        print( "-----------------------" )
        print(paste("IV = ", iv.out, sep=" "));

    }
    iv.out
}
```

Listing 1: R code to compute Information Value

```
> computeIV( df.local$DepDelay, df.local$target, TRUE )
>  # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
>  # ~~   Begin output
>  # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
>           Non-Targets Targets         WoE  PctClassDiff    IVcontrib
>  0001-0559        5028     232 -1.15570155 -0.0079048428 9.135639e-03
>  0600-0659       34401    1423 -1.26497966 -0.0566566742 7.166954e-02
>  0700-0759       31956    1988 -0.85689220 -0.0422004632 3.616125e-02
>  0800-0859       31355    2495 -0.61074637 -0.0328835439 2.008350e-02
>  ...
[truncated]
>  [1] "IV for variable X is =  0.216039970974521"
>  # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
>  # ~~   End output
>  # ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Listing 2: Usage and output of `computeIV` module

4 Extension to Regression Problems

So far the discussion has focused on **classification** problems where the predicted quantity takes very few unique values. Information Value derives from Information theory, and is a simplification of the quantity called **mutual information** that can be understood as a measure of *general* association between two random variables.

We assumed earlier that the *range* of X is spanned by disjoint sets a_k such that the collection of sets $\{a_k\}$ for $k = 1, 2, \ldots, K$ is a *complete cover*. We refer to index k as the *level* of X for brevity sake. Now let us also assume that the *range* of Y is spanned by disjoint sets b_j such that $\{b_j\}$ for $j = 1, 2, \ldots, J$ is a complete cover of Y.

Let us define $p_{j,k} = P(y \in b_j, x \in a_k)$, $p_k = P(x \in a_k)$ and $p_j = P(y \in b_j)$. The mutual information then is defined as

$$I(X;Y) = \sum_{j=1,k=1}^{J,K} p_{j,k} \ln\left(\frac{p_{j,k}}{p_j p_k}\right) \tag{10}$$

Consider the logarithm term in the sum. Since $p_{j,k} \geq p_j p_k$ and $p_{j,k} \geq 0$, the contribution of each term will be at least zero. The contribution will be exactly one only when $p_{j,k} = p_j p_k$, or when knowing $x \in a_k$ does nothing to reduce uncertainty about the value of $y \in b_j$. On the other extreme, when X and Y are identical we have $p_{j,k} = p_j = p_k$. In that case mutual information will be the Shannon Entropy of X or Y, or equal to $H(X) = -\sum_k^K p_k \ln p_k$.

4.1 Using Mutual Information Values

A large value of mutual information indicates a stronger association between X and Y, and hence it can be used to gauge relative importance of predictors in a regression setting. Note that unlike linear correlation measures such as Pearson correlation, mutual information makes no assumptions about the relationship of X and Y or the need of X and Y to be strictly numeric. It is also much more robust to extreme values due to grouping into finite number of levels. Those groupings are often determined based on the marginal distribution of X and Y.

Shannon Entropy has a known bias for the number of unique levels K or J, where a large value of K results in a larger value of entropy on the same data set. In practical applications, two approaches are most common.

1. Fix K and J so I(X;Y) computed on a dataset are comparable within
2. Normalize Shannon Entropy by dividing it with the maximum possible value – i.e. $\ln K$ or $\ln J$ so entropy is from 0 to 1