# Too Many Choices! The IMPORTANCE of Rank-Ordering Independent Variables Prior to Exploratory Data Analysis

Gregg Weldon, *Sigma Analytics & Consulting, Inc.*

*Data Analysts and Statisticians are constantly looking for more data and new data sources in order to increase their modeling capabilities. There are occasions, however, where the sheer volume of available variables slows the modeling process to a crawl. In these cases, the analyst can benefit from a simple tool that will rank all independent variables by an "importance" factor, allowing variables with low relative importance in predicting the dependent variable to be dropped from consideration. A new, streamlined list of independent variables is then available for exploratory data analysis. The method presented here is a quick, simple way to narrow the list of candidate variables to a manageable number.*

## INTRODUCTION

As the data warehousing capabilities of organizations continue to grow, the sheer volume of data increases exponentially. In the past, history on existing customers was limited to the number and speed of the company's data coders. Information on potential customers was almost non-existent. Today, companies large and small collect and archive enough data to make a typical baseball statistician green with envy. Many times, this data is carefully archived, never to be seen again. However, once an analysis project begins, this data is unleashed in a massive torrent. Ten years ago, a statistician may have had 25 to 50 variables with which to work when building a risk or marketing model. Today, there can literally be thousands of candidate independent variables. The amount of time it takes for a statistician to format, run frequencies, and analyze which of these variables may have a relationship with the dependent variable can be crippling. The key, then, is to find a tool which rank-orders all candidate variables prior to analysis. This way, the analyst can take the top 50 or 100 ranking independent variables and focus data analysis to them. Ideally, this method will provide 98%+ of the solution that a full analysis will provide, at a fraction of the time.

The value of an Importance Ratio program is that practically nothing needs to be done to the data prior to running it. Raw data goes in one end and a list of the top variables comes out the other. Only these top variables are then formatted, analyzed, and (possibly) used in the model.

## MISSING VALUES

One question that arises at the early stage of the Importance Ratio program is how to handle Missing values. Because the various SAS procedures such as PROC MEANS and PROC RANK ignore missing values, the statistician needs to decide if they should indeed be ignored or if the fact that they're missing adds something to the overall variable. How missing values are handled will affect the Importance Ratio, so this can be a significant decision. They may be ignored. Or, an array can convert all missing values to a "-1", "9999999", or any other value. It is also possible to make the missing values equal to the mean, median, or mode of that particular variable. Although an important consideration, the various methodologies on handling missing values are beyond the scope of this paper.

## CREATING IMPORTANCE

The Importance Ratio program utilizes some very basic SAS® procedures. First, we want to take each independent variable and split it up into a handful of (fairly) even groups. As a default, we typically use 10 groups, but it could be more or less, depending on the amount and robustness of the data.

```
Proc rank data=XXXXXXX
                out=testrank
(keep=&varname &vardep bracket)
                groups=10 ties=high;
        ranks bracket;
        var &varname;
        run;
```

In this case, &varname refers to a macro containing all of the independent variables, &vardep is the dependent variable, and bracket represents the 10 groups making up the variable. As an example, AGEOTD (Age of Oldest Trade in Months) was run through this portion of the program. The dependent variable for this project was BAD :

| Bracket | AGEOTD Min | AGEOTD Max | BAD Mean |
|---|---|---|---|
| 0 | 4 | 20 | 0.84 |
| 1 | 21 | 32 | 0.83 |
| 2 | 33 | 45 | 0.78 |
| 3 | 46 | 58 | 0.79 |
| 4 | 59 | 71 | 0.79 |
| 5 | 72 | 83 | 0.80 |
| 6 | 84 | 98 | 0.74 |
| 7 | 99 | 129 | 0.67 |
| 8 | 130 | 177 | 0.67 |
| 9 | 178 | 604 | 0.66 |

As you can see, AGEOTD has somewhat of a trend. Those applicants with a high AGEOTD (those that were on the books the longest) have a lower BAD rate than those with a low AGEOTD.

The information expressed above is interesting, but doesn't allow for direct comparison to the many other independent variables. Our next procedure is to take the information from the above output and summarize it.

```
Proc freq data=testrank;
Table bracket*&vardep / out=freqout;
        Run;
```

This output gives the count and percentage for each bracket by Bad (0=Not BAD, 1=BAD) and looks like the following:

| Bracket | Bad | Count | Percent |
|---|---|---|---|
| 0 | 0 | 176 | 1.61 |
| 0 | 1 | 916 | 8.38 |
| 1 | 0 | 176 | 1.61 |
| 1 | 1 | 866 | 7.92 |
| 2 | 0 | 240 | 2.19 |
| 2 | 1 | 869 | 7.95 |
| …more lines… | | | |
| 9 | 0 | 375 | 3.43 |
| 9 | 1 | 721 | 6.59 |

As you'll notice, this is the same information as above, just in a different format. The next step will use this data to begin creating variables necessary for calculating the Importance Ratio.

```
Proc means data=freqout;
   Var count;
   By bracket;
   Output out=meanout sum=bsum;
Run;
```

The new variable "bsum" is the total number of observations within each bracket.

Finally, the variables are brought together to create an Importance Ratio, that shows the relative importance of each independent variable.

```
Data importance (keep=varname imptance);
Merge freqout (keep=bracket &vardep
        count)
        Meanout (keep=bracket bsum)
        End=done;
   By bracket;
   Numer+(-count*log(count/bsum));
   Denom+bsum;
   If done then do;
    Imptance=-log(numer/denom +
              0.00000000000001);
   Varname="&varname";
   Output;
```

```
   End;
Run;
```

The formula acts as a first-order indicator of a specific variable's ability to predict the dependent variable in a bivariate setting. "Importance", then, is a univariate measurement of the amount of information that a specific independent variable tells about the dependent variable. Typical output looks like this:

| Obs | Imptance | Varname |
|---|---|---|
| 1 | 1.31350 | t4906x |
| 2 | 1.30877 | totrat |
| 3 | 1.30122 | ageotd |
| 4 | 1.29808 | inq006 |

…more lines….

This output gives the Importance Ratio of each independent variable. It's also helpful to sort the variables by Importance Ratio for ease of use. Another helpful tip is to have the names of the top XX variables saved into a text file. This will give the analyst a ready-made list of variables to plug into any programs used for exploratory data analysis.

## ADVANTAGES

As noted earlier, this program's greatest advantage is its ability to rank-order all independent variables by importance to the dependent variable with a minimal amount of pre-processing. The program runs quickly (depending on the sample size and number of independent variables) and allows as many or as few independent variables to "survive" and move on to exploratory data analysis as the analyst wishes. Because the Importance Ratio is a relative measure, data can be run weighted or unweighted. In practice, models built using this method have been just as predictive as models built after analyzing all potential variables.

## LIMITATIONS

There are several limitations to the program. The primary one is that the process doesn't take into account the correlation between independent variables. For example, below is a list of the top variables for a recent risk project, ranked by Importance Ratio:

# of Trades Ever 30+ DPD
# of Trades 30+ DPD in the Last 6 Months
# of Trades Currently 30+ DPD
# of Trades Ever 30 DPD But Never Worse
# of Trades 30+ DPD in the Last 3 Months

…etc.

Obviously, being 30 days late is a strong indicator of future delinquency! Unfortunately, if the statistician is planning on working with just the top 50 variables and they are all this closely related, any resulting model would have a high level of multicollinearity. Of the above variables, only 1 or 2 could be used in a model and still maintain a reasonable level of correlation. This causes variables that are also important to predicting behavior but have lower Importance Ratios to be ignored. For instance, age of oldest trade could be an important variable in a predictive model. It is also not correlated with 30 day delinquencies. However, its Importance Ratio may be farther down the list because 30 day delinquency is SO important. Relying on just the top XX variables from the Importance Ratio program could cause many potentially strong variables to be missed.

One way of correcting for this would be to go through the Importance Ratio list and manually add variables that are lower on the list but, because of their lack of correlation with the top variables, could add valuable information to the model being built. This method works well when the independent variable list is known and understood by the statistician. A group of 150 credit bureau variables that are used on a regular basis to build risk models for credit unions, for example, is perfect for this method.

However, there are occasions when the analyst is working with hundreds (or thousands) of variables created by the client. Many of these variables may be unfamiliar to the analyst, as may be the client's particular industry or business

practices. In these cases, going beyond the top XX variables amounts to little more than guesswork. It's possible to find the best variables through "gut feel" or plain dumb luck, but not likely.

In this case, it may make sense to add a clustering routine to the Importance Ratio program. The Importance Ratio is calculated just as before, but the variables are then broken into a certain number of individual clusters, based on the independent variables' correlation with each other. This way, the statistician may take the top XX variables from each cluster. This would minimize the multicollinearity problem and allow variables from farther down the list to rise to the top.

Another limitation to the program is that the Importance Ratio doesn't vary a huge amount from the "best" variables to the "worst" variables. The top variables, as a group, will be much more likely to work in a model. However, the 100th variable may not be much "worse" than the 50th variable. If the top 50 variables are chosen, variable #51 could also work.

The fact that an independent variable has a high relative Importance Ratio is no guarantee that it will make it in the final model (although it is a good indicator). In the AGEOTD example earlier, the longer the time on the books, the lower the likelihood of going delinquent. Sometimes, however, a variable will demonstrate a trend that makes no sense intuitively. As always, business and common sense take precedence over "what the computer says". Variables that violate the laws of common sense should be dropped, no matter the Importance Ratio.

## SUMMARY

The increase in data is a trend that will continue for the foreseeable future. Projects will continue to get more complex as businesses search for better ways to compete with industry rivals. Industries not traditionally associated with statistical modeling will begin analyzing their data with great interest. In order to keep up with this crush of information, new procedures such as the Importance Ratio will need to be utilized. These methods can maintain a manageable, streamlined approach to data analysis, allowing analysts to continue to build models with a high level of confidence in their ability to predict behavior.

## APPENDIX

To further illustrate the inner workings of the Importance Ratio, below is an example from a recent project. The independent variable is TOTRAT (Ratio of Balance to High Credit for All Open Trades) and the dependent variable is BAD (90+ DPD).

## FREQOUT output:

| Obs | bracket | BAD | COUNT | PERCENT |
|---|---|---|---|---|
| 1 | 2 | 0 | 429 | 3.9225 |
| 2 | 2 | 1 | 2158 | 19.7312 |
| 3 | 3 | 0 | 221 | 2.0207 |
| 4 | 3 | 1 | 567 | 5.1842 |
| 5 | 4 | 0 | 742 | 6.7843 |
| 6 | 4 | 1 | 1351 | 12.3526 |

...more lines of output

## MEANOUT Output:

| Obs | bracket | _TYPE_ | _FREQ_ | bsum |
|---|---|---|---|---|
| 1 | 2 | 0 | 2 | 2587 |
| 2 | 3 | 0 | 2 | 788 |
| 3 | 4 | 0 | 2 | 2093 |
| 4 | 5 | 0 | 2 | 1094 |

...more lines of output

## FORMULA:

(-429 * log(429/2587)) = 770.826 / 2587

770.826 + (-2158 * log(2158/2587)) = 1162.108 / 5174

1162.108 + (-221 * log(221/788)) = 1443.073 / 5962

1443.073 + (-567 * log(567/788)) = 1629.695 / 6750

…more lines…

5777.793 + (-952 * log(952/1093)) =
5909.279 / 21874

IMPTANCE = -log(5909.279 / 21874 +
0.000000001) = **1.30877**

## ACKNOWLEDGEMENTS

SAS is a registered trademark of SAS Institute, Inc., in the USA and other countries. ® indicates USA registration.

## AUTHOR'S ADDRESS

The author may be contacted at:

Gregg Weldon
Sigma Analytics & Consulting, Inc.
7000 Central Parkway
Suite 330
Atlanta, GA 30328

(770) 804-1088
gregg.weldon@sigmaanalytics.com

Submission Number W2039