Paper SA01-2012

Methods for Interaction Detection in Predictive Modeling Using SAS

Doug Thompson, PhD, Blue Cross Blue Shield of IL, NM, OK & TX, Chicago, IL

ABSTRACT

Analysts typically consider combinations of variables ("interactions") when building a predictive model. If the target is Y and there is an interaction between two predictors X1 and X2, it means that the relationship between X1 and Y differs depending on the value of X2. SAS® provides several easy-to-use methods to detect interactions. One method is to consider all possible n-way interactions using multiplicative interaction terms. For example, this can be done in the MODEL statement of PROC LOGISTIC using "|" between potentially interactive variables and "@n" to specify the number of variables that can be involved in an interaction (e.g., "@2" refers to 2-way interactions). Another approach that has been recommended is to use decision trees to identify interactions; SAS Enterprise Miner enables interaction detection using this method. This paper presents a case study illustrating the use of different methods for interaction detection (i.e., all possible n-way interactions, interactions detected via decision trees, and a combination of the two) when building a predictive model in SAS.

INTRODUCTION

When creating a predictive model, analysts typically consider combinations of variables ("interactions") as potential predictors. If the target is Y and there is an interaction between two predictors X1 and X2, it means that the relationship between X1 and Y differs depending on the value of X2.

There are several methods that can be used to detect interactions in SAS. One method is to consider all possible n-way interactions using multiplicative interaction terms. SAS/STAT provides easy-to-use tools for doing this. For example, this can be done in the MODEL statement of PROC LOGISTIC using "|" between potentially interactive predictors and "@n" to specify the number of predictors that can be involved in an interaction (e.g., "@2" refers to 2-way interactions). In a logistic regression model, with a binary dependent variable Y and potential predictors X1, X2 and X3, the MODEL statement of PROC LOGISTIC might be: "model Y = X1|X2|X3 @2 / selection=stepwise". Then interactions X1*X2, X1*X3 and X2*X3 would all be considered, and retained in the model if they met the fit criteria specified in the stepwise algorithm. Although this method is easy to execute in SAS/STAT, a downside of the method is that the analysis can be slow when many interaction terms are considered.

A second approach to interaction detection that has been recommended is decision trees. This approach is used for interaction detection in the Rapid Predictive Modeler procedure of Enterprise Miner. A decision tree is a set of rules for dividing a set of observations into distinct subgroups. The first rule is the one that best splits the entire set of observations into subgroups, such that the resulting subgroups are as pure as possible with respect to the dependent variable (assuming a binary dependent variable coded 0 or 1, this means that the resulting subgroups are as different as possible from each other with respect to the proportion of 1's). Then another rule is applied to the resulting subgroups, splitting them into further subgroups that are as pure as possible with respect to the dependent variable. This process continues until pre-specified stopping criteria are met, which are sometimes based on the minimum allowed number of observations in a subgroup. The final resulting subgroups are called "leaves". A leaf that is defined by multiple variables may represent an interaction. For example, a leaf may be defined by the rule "an observation is in Leaf 1 if X1=1 and X2<56". Because leaves are constructed by identifying subgroups of observations that are as pure as possible with respect to the dependent variable, this essentially means that Y depends on both X1 and X2. However, a leaf defined by two predictors X1 and X2 does not necessarily point to an interaction between X1 and X2. It means that X1 and X2 are both important for the prediction of Y, but this is consistent with either main effects of X1 and X2 and no X1*X2 interaction, or an interaction between X1 and X2 (meaning that the relationship between X1 and Y is different at different levels of X2). An example of this difference is presented below.

The decision tree method of interaction detection requires SAS Enterprise Miner, unless the analyst is willing to program a decision tree algorithm in BASE/SAS, or use some other software that does decision tree analyses such

as R or Weka. An upside is that decision trees can detect complex non-linear associations; downsides are that the analyst needs to have Enterprise Miner available, program a decision tree algorithm or use various non-SAS decision tree algorithms in the analysis. Given the high cost of Enterprise Miner, it is an important practical question whether decision trees offer a substantial improvement over n-way interaction detection methods available with the SAS/STAT regression procedures.

This paper presents a case study of using different methods for interaction detection when building a predictive model in SAS. The case study is a logistic regression model that would be fairly typical in marketing analytics. SAS/STAT is the primary analytic package. All of the methods can be implemented in SAS/STAT, with the exception that decision tree interaction detection uses SAS Enterprise Miner. This is a case study that illustrates findings in a single analytic situation; it is not guaranteed that similar findings will be present in every predictive modeling scenario.

METHOD

ILLUSTRATIVE DATA AND ANALYSIS SITUATION

The analyses are illustrated with a case study using data from the National Health and Nutrition Examination Survey (NHANES), 2003-‘04 and ‘05-‘06 waves combined. Assume that a campaign is being devised to market a new microbrew variety. The objective is to advertise to potential customers who are likely to drink beer; there is little value in advertising to individuals who are unlikely to drink beer. A predictive model is created to target likely beer drinkers.

The dependent variable is drank_beer, defined as drinking beer sometime during the past year (coded as 1=yes or 0=no, based on the NHANES food frequency questionnaire). Potential predictors included measures of demographics (race/ethnicity, age, gender, income, education, marriage status), energy intake, Body Mass Index (BMI) and pregnancy status. The predictors included a mix of continuous variables (age, energy intake, BMI) and binary variables (all others). The continuous variables differed in scale. Prior to building the model, the observations were randomly allocated to training and validation datasets (2/3 training sample, 1/3 validation or “holdout” sample). The variable splitwgt was defined as 1=training observation, .= validation observation. When splitwgt is included in a WEIGHT statement, PROC LOGISTIC builds the model using only the training observation but produces scores for all observations in the dataset (including the validation sample). Models were built using the training observations and evaluated using the validation sample. Prior to analysis, a small number of observations with missing data were excluded.

The analysis is applicable to any marketing scenario where the objective is to distribute the advertising message to likely consumers, and avoid unlikely consumers, for example, likely purchasers of automobiles, insurance policies, or telecommunications services.

Methods considered for interaction detection

1. Main effects only

This method considers only main effects and no interactions. The code is shown below.

```
proc logistic data=nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer = energy RIDAGEYR bmi wave0506 female mexam othhisp white afam
inc_lt25K inc_75Kplus some_college_or_more
married pregnant / selection=stepwise;
output out=scores1 prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms1;
run;
```

2. All possible 2-way interactions (no recode of continuous predictors)

This method considers all possible 2-way interactions among the predictors. Linear slopes are estimated for the continuous predictors (BMI, RIDAGEYR and energy).

```
proc logistic data=nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer = energy|RIDAGEYR|bmi|wave0506|female|mexam|othhisp|white|
afam|inc_lt25K|inc_75Kplus|some_college_or_more|married|pregnant @2 /
selection=stepwise;
output out=scores2 prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms2;
run;
```

3. All possible 2-way interactions (binary split of continuous predictors)

Method 2 estimated linear slopes for the continuous predictors. A possible improvement is to split the predictors into categories prior to estimating the interactions. This enables non-linear patterns of association between a continuous predictor and the dependent variable to be detected. To gauge the possible lift from using continuous predictors split into categories, interactions after splitting the predictors into categories were tested. Two and ten splits were considered. Two-category splits provides a baseline, while 10-category splits enables a more sensitive representation of possible non-linear patterns of association between the continuous predictors and the dependent variable. The macro categorize_continuous uses PROC RANK to break the continuous predictors into n equal size categories, passed to the macro using levels=n. A set of dummy variables is created to represent the resulting dichotomous predictor variable and entered into the MODEL statement of PROC LOGISTIC.

```
options mprint;
%macro categorize_continuous(levels=2);
%let levels_minus_1=%eval(&levels-1);
%let ranked = r_energy r_RIDAGEYR r_bmi;

* Use class variables for continuous;
* dummy coded ranks;
proc rank data=nhanes_0306_extract out=rnk_nhanes_0306_extract groups=&levels;
ranks r_energy r_RIDAGEYR r_bmi;
var energy RIDAGEYR bmi;
run;

data rnk_nhanes_0306_extract;
set rnk_nhanes_0306_extract;

%do i=1 %to 3;
%let curr_var = %scan(&ranked,&i);
%do j=0 %to &levels_minus_1;
&curr_var.&j=(&curr_var=&j);
%end;
%end;

run;

proc logistic data=rnk_nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer =
%do k=0 %to &levels_minus_1;
r_energy&k|r_RIDAGEYR&k|r_bmi&k|
%end;
wave0506|female|mexam|othhisp|white|afam|inc_lt25K|inc_75Kplus|
some_college_or_more|married|pregnant @2 / selection=stepwise;
output out=scores prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms;
run;
%mend categorize_continuous;
```

```
%categorize_continuous(levels=2);
```

4. All possible 2-way interactions (10 category split of continuous predictors)

This is the same as Method 3 except that continuous predictors are broken into 10 equal size categories instead of 2. The code is exactly the same as in Method 3, except for a different macro call: %***categorize_continuous***(levels=**10**);

5. Stepwise with decision tree leaves, no other interactions

Method 5 used decision tree leaves to represent interactions. The leaves were terminal nodes from a set of decision tree analyses conducted using SAS Enterprise Miner (EM). To conduct decision tree analyses, the first step was to import the training sample data into EM. Then decision trees were built varying the following analytic options:

Tree A. Maximum branches = 2, maximum depth 2, accept all other defaults (misclassification rate, training sample = 0.349).

Tree B. Maximum branches = 5, maximum depth 2, accept all other defaults (misclassification rate, training sample = 0.349; Tree B turned out to be identical to Tree A).

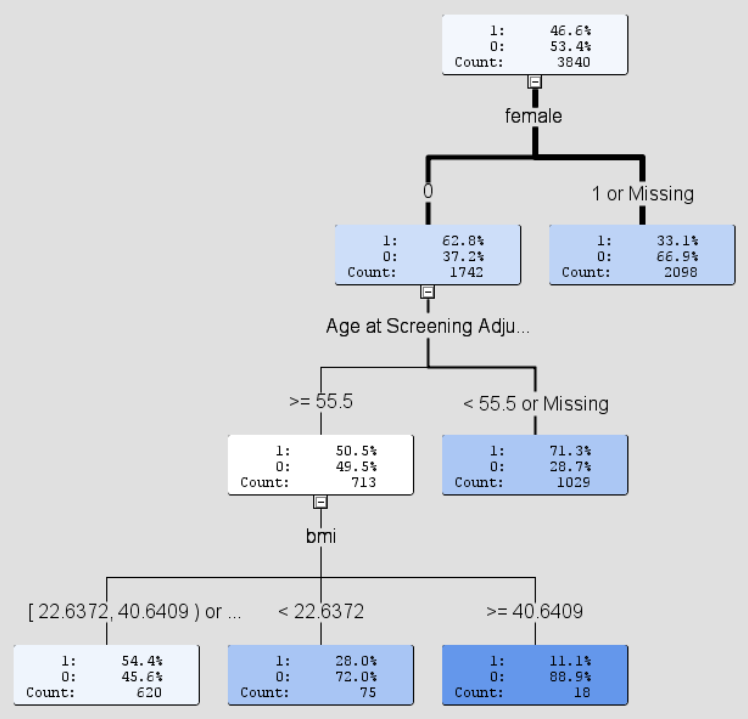
Tree C. Maximum branches = 2, maximum depth 3, accept all other defaults (misclassification rate, training sample = 0.341).

Tree D. Maximum branches = 5, maximum depth 3, accept all other defaults (misclassification rate, training sample = 0.337).

Tree E. Interactive tree, allowing only splits where –log(p)>=3 (misclassification rate, training sample = 0.349).

The five trees exhibited only minor differences in accuracy as indicated by misclassification rate on the training sample. Results for the best fitting tree, Tree D, are shown in Figure 1.

Figure 1. Results for the best fitting decision tree (Tree D).



All terminal nodes (leaves) from all trees were retained. Table 1 below describes the entire set of leaves. Leaves that were already represented by binary split predictors (e.g., female=0 vs. other, which is the top right terminal node that appears in Figure 1) were not coded as a leaf because this would be redundant. Across all trees, there were 8 distinct leaves, labeled "leaf1" through "leaf8" subsequently in this paper. After conducting decision tree analyses in SAS/EM as described above, terminal nodes were coded in the following dataset and the resulting leaves were used as inputs to logistic regression analyses conducted in SAS/STAT as shown below.

```
* Code nodes from decision tree analysis;
data nhanes_0306_extract;
set nhanes_0306_extract;
leaf1=(female=0 and ridageyr<55.5);
leaf2=(female=0 and ridageyr>=55.5 and bmi ne . and bmi<22.6372);
leaf3=(female=0 and ridageyr>=55.5 and (bmi=. or bmi>=22.6372));
leaf4=(female=0 and ridageyr>=55.5 and bmi ne . and bmi>40.6409);
leaf5=(female=0 and ridageyr>=55.5 and (bmi=. or 22.6372<=bmi<40.6409));
leaf6=(female in(.,1) and ridageyr<55.5);
leaf7=(female in(.,1) and ridageyr>=55.5 and some_college_or_more=1);
leaf8=(female in(.,1) and ridageyr>=55.5 and some_college_or_more<1);
run;

proc logistic data=nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer =
leaf1 leaf2 leaf3 leaf4 leaf5 leaf6 leaf7 leaf8
energy RIDAGEYR bmi wave0506 female mexam othhisp white afam          inc_lt25K
inc_75Kplus some_college_or_more married pregnant
/ include=8 selection=stepwise;
output out=scores5 prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms5;
run;
```

6. Stepwise with decision tree leaves, allow other 2-way interactions

This method estimates all possible 2-way interactions plus the potential interactions identified using decision trees.

```
proc logistic data=nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer =
leaf1 leaf2 leaf3 leaf4 leaf5 leaf6 leaf7 leaf8
energy|RIDAGEYR|bmi|wave0506|female|mexam|othhisp|white|afam|
inc_lt25K|inc_75Kplus|some_college_or_more|married|pregnant @2
/ selection=stepwise;
output out=scores6 prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms6;
run;
```

7. Stepwise with all possible 2-way interactions (no recode of continuous predictors), then allow decision tree leaves to enter

This model takes the model selected in Method 2, forces that model in via the include= option in the MODEL statement of PROC LOGISTIC, and finally allows decision trees to enter the model.

```
proc logistic data=nhanes_0306_extract descending namelen=100;
weight splitwgt;
model drank_beer =
Energy
RIDAGEYR
Energy*RIDAGEYR
bmi
```

```
female
mexam
afam
inc_lt25K
inc_75Kplus
afam*inc_75Kplus
some_college_or_more
Energy*some_college_or_more
afam*some_college_or_more
married
RIDAGEYR*married
afam*married
pregnant
mexam*pregnant
leaf1 leaf2 leaf3 leaf4 leaf5 leaf6 leaf7 leaf8
/ include=18 selection=stepwise;
output out=scores7 prob=prob predprobs=crossvalidate;
ods output parameterestimates=parms7;
run;
```

FIT STATISTICS CONSIDERED

To assess the fit of models resulting from Methods 1 through 7, fit statistics were computed using the observations in the validation sample. Two common fit statistics were used, the C-statistic (also known as area under the ROC curve or AUC) and percent of target cases in the top decile, i.e., the percent of all observations where drank_beer=1 that were caught in the top 10% of observations by model score. The C-statistic ranges from 0 to 1; the closer to 1, the better the model fit. A model where C=0.5 is no better than random guessing (in other words, a model that predicts well should have C>0.5). The percent of target cases in the top decile ranges from 0% to 100%; the closer to 100% the better, and 10% indicates a model that is no better than random guessing (in other words, a model that predicts well should have values of >10% for this fit statistic).

RESULTS

Descriptive statistics for the predictors considered are shown in Table 1. All percentages are column percentages (for example, 54.6% of the total sample was female, 39.3% of the beer drinkers were female and 68.0% of the non-beer-drinkers were female).

Table 1. Mean or % of potential predictor variables by beer consumption (drank beer sometime in the past year vs. did not drink beer in the past year) and in total.

Variable name	Description	Drank beer (n=2,703)	Did not drink beer (n=3,074)	Total (n=5,777)
Energy	24-hour energy intake (kcal)	2,360.4	1,950.6	2,142.4
RIDAGEYR	Age (years)	46.1	52.5	49.5
bmi	Body mass index (BMI)	28.1	29.2	28.7
wave0506	NHANES wave = 05-06	48.6%	46.9%	47.7%
female	Female	39.3%	68.0%	54.6%
mexam	Mexican America	18.7%	17.6%	18.1%
othhisp	Other Hispanic	3.1%	2.6%	2.9%
white	Non-Hispanic Caucasian	57.7%	53.3%	55.3%
afam	African American	16.4%	22.1%	19.4%
inc_lt25K	Annual income < $25K	22.0%	32.8%	27.8%
inc_75Kplus	Annual income $75K+	27.0%	18.2%	22.3%
some_college_or_more	Some college education or more	56.5%	46.0%	50.9%
married	Married	57.0%	58.8%	58.0%
pregnant	Pregnant	4.1%	7.3%	5.8%
leaf1	Male & age < 56	40.9%	14.9%	27.1%
leaf2	Male & age >=56 & BMI < 22.6	1.6%	2.6%	2.1%
leaf3	Male & age >=56 & BMI >= 22.6	18.2%	14.5%	16.2%
leaf4	Male & age >=56 & BMI > 40.6	0.1%	0.7%	0.4%
leaf5	Male & age >=56 & (26.6<=BMI<40.6)	18.1%	13.8%	15.8%
leaf6	Female & age < 56	30.6%	38.6%	34.8%
leaf7	Female & age >=56 & some college+	4.6%	10.8%	7.9%
leaf8	Female & age >=56 & <college	4.1%	18.6%	11.8%

Fit statistics from models using Methods 1 through 7 are shown in Table 2.

Table 2. Fit statistics for model selection Methods 1 through 7 for the training sample and validation sample.

Method	Training sample		Validation sample	
	C-statistic	% target in top decile	C-statistic	% target in top decile
1. Main effects only	0.729	16.8%	0.732	16.8%
2. All possible 2-way interactions (no recode of continuous predictors)	0.735	17.2%	0.735	17.2%
3. All possible 2-way interactions (binary split of continuous predictors)	0.725	17.1%	0.731	16.7%
4. All possible 2-way interactions (10 category split of continuous predictors)	0.737	17.2%	0.726	16.5%
5. Stepwise with decision tree leaves, no other interactions	0.737	17.2%	0.726	16.9%
6. Stepwise with decision tree leaves, allow other 2-way interactions	0.739	17.0%	0.729	17.3%
7. Stepwise with all possible 2-way interactions (no recode of continuous predictors); then allow decision tree leaves to enter	0.743	17.3%	0.733	17.2%

All models fit reasonably well and were markedly better than random guessing (C>0.5, % of target observations in the top decile >10%). Based on the c-statistic, the best method was Method 2 for the validation sample (Method 7 was close). Based on % of target observations in the top decile, the best method was Method 6 for validation, although Methods 2 and 7 were close. Creating categorical predictors always resulted in a worse-fitting solution compared with estimating linear slopes of continuous predictors, regardless of whether 2- or 10-category splits where used. Allowing 2-way interactions in addition to tree leaves resulted in a better solution than using decision tree leaves alone.

Predictors selected and parameter estimates for the solutions that tended to exhibit best predictive performance, i.e., Methods 2 and 7, are shown in Tables 3 through 5 below.

Table 3. Parameter estimates, standard errors and p-values (null hypothesis is parameter estimate =0) for the model selected by Method 2.

Variable	Estimate	SE	p-value
Intercept	2.4197	0.3497	<0.0001
Energy	<0.0001	0.0001	0.7917
RIDAGEYR	-0.0369	0.0055	<0.0001
Energy*RIDAGEYR	<0.0001	<0.0001	0.0474
bmi	-0.0179	0.0055	0.0012
female	-1.1900	0.0801	<0.0001
mexam	0.2935	0.1032	0.0045
afam	-0.2695	0.1519	0.0759
inc_lt25K	-0.2539	0.0878	0.0038
inc_75Kplus	0.3215	0.1026	0.0017
afam*inc_75Kplus	-0.6136	0.2483	0.0135
some_college_or_more	0.6553	0.1827	0.0003
Energy*some_college_or_more	-0.0002	0.0001	0.0316
afam*some_college_or_more	-0.4627	0.1861	0.0129
married	-0.9103	0.2212	<0.0001
RIDAGEYR*married	0.0109	0.0041	0.0073
afam*married	0.3874	0.1910	0.0425
pregnant	-0.3270	0.1776	0.0656
mexam*pregnant	-1.3035	0.4148	0.0017

Table 4. Parameter estimates, standard errors and p-values (null hypothesis is parameter estimate =0) for the model selected by Method 6.

Variable	Estimate	SE	p-value
Intercept	0.6150	0.3459	0.0755
leaf1	1.4788	0.1827	<0.0001
leaf5	1.2501	0.1445	<0.0001
leaf6	0.3661	0.1799	0.0418
leaf8	-0.4730	0.1760	0.0072
Energy	0.0002	<0.0001	0.0002
RIDAGEYR	-0.0174	0.0043	0.0001
bmi	-0.0218	0.0056	0.0001
afam	-0.3846	0.1020	0.0002
inc_lt25K	-0.2767	0.0872	0.0015
inc_75Kplus	0.3424	0.0985	0.0005
afam*inc_75Kplus	-0.6087	0.2346	0.0095
married	-0.8374	0.2147	0.0001
RIDAGEYR*married	0.0110	0.0041	0.0076
pregnant	-0.5729	0.1648	0.0005

Table 5. Parameter estimates, standard errors and p-values (null hypothesis is parameter estimate =0) for the model selected by Method 7.

Variable	Estimate	SE	p-value
Intercept	2.1581	0.3739	<0.0001
Energy	0.0001	0.0001	0.5404
RIDAGEYR	-0.0288	0.0061	<0.0001
Energy*RIDAGEYR	0.0000	0.0000	0.1132
bmi	-0.0200	0.0057	0.0004
female	-1.5108	0.1204	<0.0001
mexam	0.2819	0.1039	0.0067
afam	-0.2926	0.1523	0.0547
inc_lt25K	-0.2709	0.0884	0.0022
inc_75Kplus	0.2926	0.1033	0.0046
afam*inc_75Kplus	-0.6091	0.2493	0.0146
some_college_or_more	0.6449	0.1839	0.0005
Energy*some_college_or_more	-0.0002	0.0001	0.0356
afam*some_college_or_more	-0.4620	0.1868	0.0134
married	-0.9536	0.2217	<0.0001
RIDAGEYR*married	0.0115	0.0041	0.0051
afam*married	0.3952	0.1917	0.0393
pregnant	-0.3324	0.1786	0.0628
mexam*pregnant	-1.2956	0.4154	0.0018
leaf2	-1.3581	0.2771	<0.0001
leaf4	-2.1964	0.7609	0.0039
leaf6	0.4125	0.1418	0.0036

Some 2-way interactions were selected by all three methods, specifically afam*inc_75Kplus and RIDAGEYR*married. These interactions are graphed using validation sample observations in Figures 2 and 3 below. Although a linear slope was estimated for RIDAGEYR (age in years), Figure 3 uses a median split of RIDAGYR for graphing purposes. Among African Americans, beer drinking was unrelated to income, whereas for other ethnicities, beer drinking was much more frequent among higher-income individuals. For younger individuals, beer drinking was more frequent among the unmarried, whereas beer drinking was more frequent among the married for older individuals.

Figure 2. Percent of individuals drinking beer by ethnicity and income.

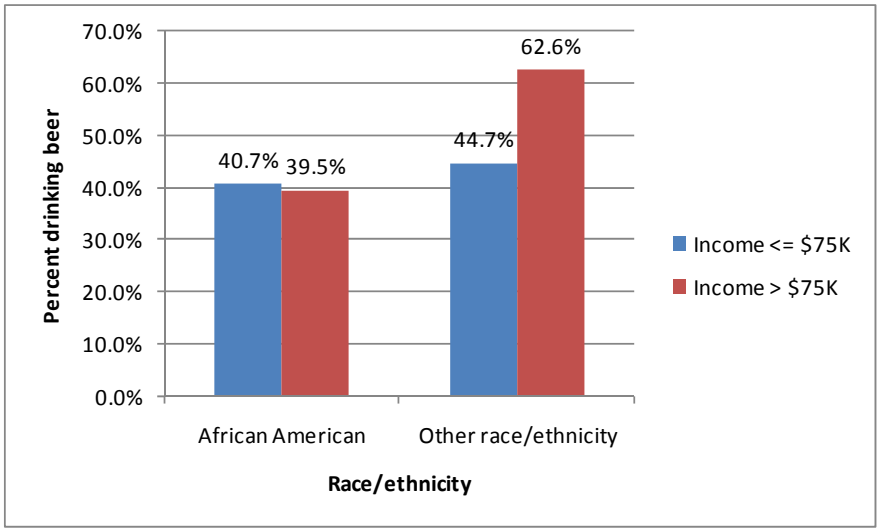
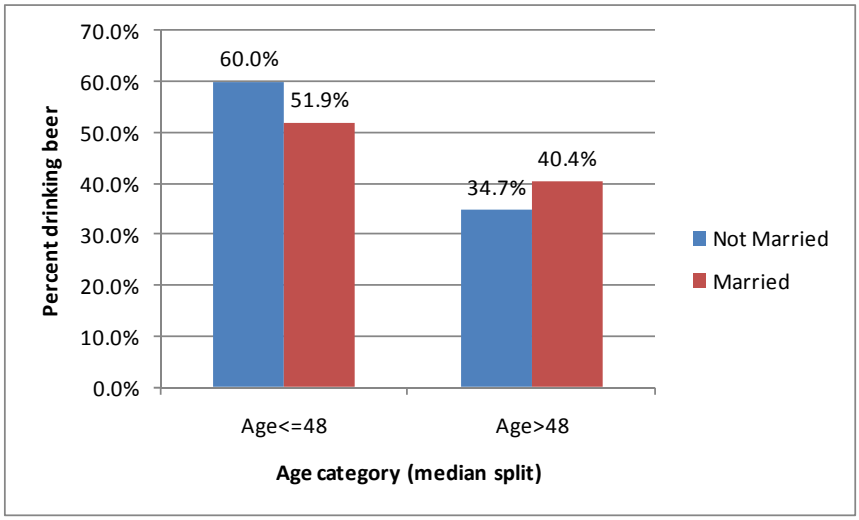


Figure 3. Percent of individuals drinking beer by age and marriage status.



As noted above, tree leaves may or may not represent interactions. For example, Leaf1 was uncovered in the decision tree analyses. It is a combination of gender and age<56. It entered as a significant predictor in the model selected by Method 6. However, Leaf1 represents main effects of gender and age<56, but there is no interaction. This is shown in the following analysis.

```
data trainset;
set nhanes_0306_extract(where=(role='T'));
age_lt56=(ridageyr<55.5);
age_gt56=(ridageyr>=55.5);
bmi_lt23=(bmi<22.6372);
run;

proc logistic data=trainset descending namelen=100;
model drank_beer = female age_lt56;
run;
proc logistic data=trainset descending namelen=100;
model drank_beer = female age_lt56 leaf1;
run;
```

In both logistic regressions, gender and age<56 are significant predictors of drinking beer (p<0.0001), but leaf1 does not add anything to the prediction (p=0.57 in the second logistic regression; note that leaf1 is equivalent to female*age_lt56). Why was there a tree leaf, but no interaction? The reason becomes clear based on the data in Table 6.

Table 6. Percent of individuals drinking beer by age and gender.

	Age	
Gender	<56	≥56
Female	40.2%	20.2%
Male	71.3%	50.5%

First the tree splits by gender (rates of drinking beer appear much higher among males), then the tree splits by age among the males (there is a 20.8% difference between younger vs. older males in beer drinking rates, as opposed to a 20.0% difference between younger vs. older females). These two splits result in Leaf1, but these splits indicate main effects only, not interactions.

However, Leaf2 (defined by gender, age>56 and BMI<22.6) is a decision tree leaf that represents a real interaction, as illustrated in the logistic regressions below.

```
proc logistic data=trainset descending namelen=100;
model drank_beer =
female age_gt56 bmi_lt23;
run;
proc logistic data=trainset descending namelen=100;
model drank_beer =
female age_gt56 bmi_lt23 leaf2;
run;
```

In the second logistic regression, leaf2 (which is equivalent to female*age_gt56*bmi_lt23) was highly significant ($p<0.0001$), and the C-statistics differed between the two logistic regressions (C-statistics =0.70 vs. 0.71, respectively). When looking at means by group, it appears that the interaction is in part driven by a difference in beer drinking rates among the older respondents (age≥56), by BMI and gender. Specifically, among the older respondents, males with lower BMI (<22.6) were much less likely to drink beer (28%, vs. 53% for older males with BMI≥22.6), whereas females with lower BMI were slightly more likely to drink beer (21%, vs. 20% for older females with BMI≥22.6).

CONCLUSION

This paper illustrated two possible approaches to detect interactions in predictive models: n-way multiplicative interactions and decisions trees. These approaches were examined separately and in combination in the scenarios illustrated in this paper.

In the illustrative scenarios, n-way multiplicative interactions and decision trees both pointed to useful interactions. The n-way multiplicative interaction detection method performed best based on the holdout c-statistic (and it was nearly the best in the holdout decile analysis), raising questions as to whether decision trees -- and the tools required to create them -- are really necessary for effective interaction detection in predictive models. However, good models did result from combining the two methods of interaction detection -- a combination of the two approaches, Method 6, performed best in the holdout decile analysis. Therefore, possibly the optimal approach is to combine the two approaches for interaction detection in predictive models. A definitive answer would require further research.

When using the decision tree method to detect interactions, it is worthwhile to keep in mind that tree leaves may point to either main effects or interactions.

The results reported in this paper were a case study. It is unknown to what extent the results reported here will generalize to other situations.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Doug Thompson, PhD
Senior Director, Analytics Consulting
Blue Cross Blue Shield of IL, NM, OK & TX
300 E Randolph
Chicago, IL 60601
(312) 653-5371
doug_thompson@bcbsil.com