Screening and Transformation of Continuous Predictors for Logistic Regression

Bruce Lund

Magnify Analytic Solutions,
a Division of Marketing Associates, LLC
Detroit, MI

Goals of this Talk

There are two main goals in this talk.

I. To describe a process of screening a multitude of continuous predictors for PROC LOGISTIC by a SAS® Macro %LOGIT_CONTINUOUS.

II. To explain and discuss the FSP (Function Selection Procedure) of Royston and Sauerbrei.

a) FSP is focused on the candidate predictors that pass %LOGIT_CONTINUOUS screening

b) FSP processes one variable at a time.

c) FSP selects a final transformation or eliminates the variable from further review.

Broad Outline of Logistic Modeling

Here is how **Parts I and II** fit into a broad outline.

- Data Preparation (e.g. resolving "missing" values)
- EDA: tabulations and summarizations
- Screening of predictor variables (dozens or hundreds):
  - **Discrete / nominal**: Information value, c-statistic, chi-square
  - **Continuous:** PART I of talk
- Transforming of predictor variables:
  - **Discrete / nominal**: Binning and weight-of-evidence coding
  - **Continuous:** PART II of talk
- Exploring interactions / multi-collinearity checking
- Modeling

What Constitutes a Continuous Predictor?

No clear-cut definition of "continuous" X.

Maybe:

… where it's "**hard**" to start to do **binning** and **weight-of-evidence (WOE)** coding

… when many values of X occur only for 1 observation

Part I: Overview

%LOGIT_CONTINUOUS provides an **EDA** tool for inspecting **each predictor** as well as a **collection of transformations** of the predictor for, at least, minimal predictive power in order to justify further investigation.

If the number of candidate predictors is only a few, an exhaustive approach of fitting the predictor and transformations of the predictor by PROC LOGISTIC provides a simple, direct solution.

However, each PROC LOGISTIC requires a pass of the training data set.

%LOGIT_CONTINUOUS uses a "**short-cut**" provided by a connection between **2-group t-test** and **logistic regression**.

The training data set is passed only 3 times.

Part I: The Starting Point

Applied Logistic Regression 3rd ed. by Hosmer, Lemeshow, and Sturdivant (2013) discusses the connection between discriminant analysis and logistic regression (see p. 21 and p. 91.)

This led to the idea of mass-screening of potential logistic predictors by using PROC TTEST.

PROC TTEST is at the center of %LOGIT_CONTINUOUS

Very Quick Summary of Logistic Regression

Here is the 1-variable logistic regression model:

$$P(Y=1) = \exp(\beta_0 + \beta_1 X_1) / (1 + \exp(\beta_0 + \beta_1 X_1))$$

The β_1 (or X_1) is "significant" if its **(Wald) Chi-Square** (1 d.f.) is "large".

Examples: $P(\chi 2 > 6.6) = 1\%$ or $P(\chi 2 > 3.8) = 5\%$

Analysis of Maximum Likelihood Estimates						
Parameter	DF		Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	β_0 =	0.2369	0.0319	55.1351	<.0001
X1	1	β_1 =	-0.0894	0.032	**7.7964**	0.0052

To find that $\chi 2$ = **7.7964** the PROC LOGISTIC had to be run on the data set. No closed-end formula exists.

2-Group t-Test & Logistic Regression

But there is a estimate for the **Wald chi-sq.** for β_1 which does not require running PROC LOGISTIC.

It is given by: t^2 where .....

$t = (\ _0 - \ _1)\ \mathrm{sqrt}(1/n_0 + 1/n_1) / S_p$ ...

This is the **t-statistic** * for difference of 2 group (population) means.

$_0$ and $_1$ are the sample means for groups: Y=0, Y=1.

S_p is sample pooled std. dev.

Also interesting: An estimate for β_1 is $b_{1D} = (\ _0 - \ _1) / S_p^2$

"**D**" is for discriminant analysis – the model behind the t-statistic idea.

See Appendix A of paper for statistical background.

(*) under null hypothesis that: $\mu_0 = \mu_1$

Assumptions on X

Approximation of Wald chi-square of X by $\mathbf{t^2}$ is best when …

➔ X is normal over each of the 2 populations of Y values (0 and 1) with means μ_j for j = 0, 1 and common std dev σ

Assumption of common std dev σ is why S_p (pooled stdev) is used in

$$\mathbf{t} = (\ _0 - \ _1)\ \mathrm{sqrt}(1/n_0 + 1/n_1) / S_p$$

Using S_p is only appropriate if we have a common std dev σ

A way to screen hundreds of X's …

We want to compute t^2 for hundred's of X's: X1 - X100

This is exactly what PROC TTEST does … for any number of predictors in the VAR statement …

PROC TTEST Code

```
ods output TTests = TT;
proc ttest data=example plots=none;
  class Y;
  var X1 X2 X3 X4 X5 X6;
```

Need ODS OUTPUT = Ttests = TT

```
data TT; set TT;  /* 2 rows per X */
  chisq_D = tValue**2;
  label Probt = "Prob ChiSq";
  where method = "Pooled";
```

Probt gives "2-tail prob." for tValue. Same value as "right-tail" prob. for chisq_D.

Row 1: stdev is "Pooled"
Row 2: "Satterthwaite" (groups have **unequal** std dev.)

t^2 best approx. **Wald chi-square** when X is normal & groups have **equal** std dev.

tValue = t = $(\ _0 - \ _1)/(S_p \text{ sqrt}(1/n_0 + /n_1))$

Finally, Pooled and Satterthwaite give very similar values for t^2 anyway

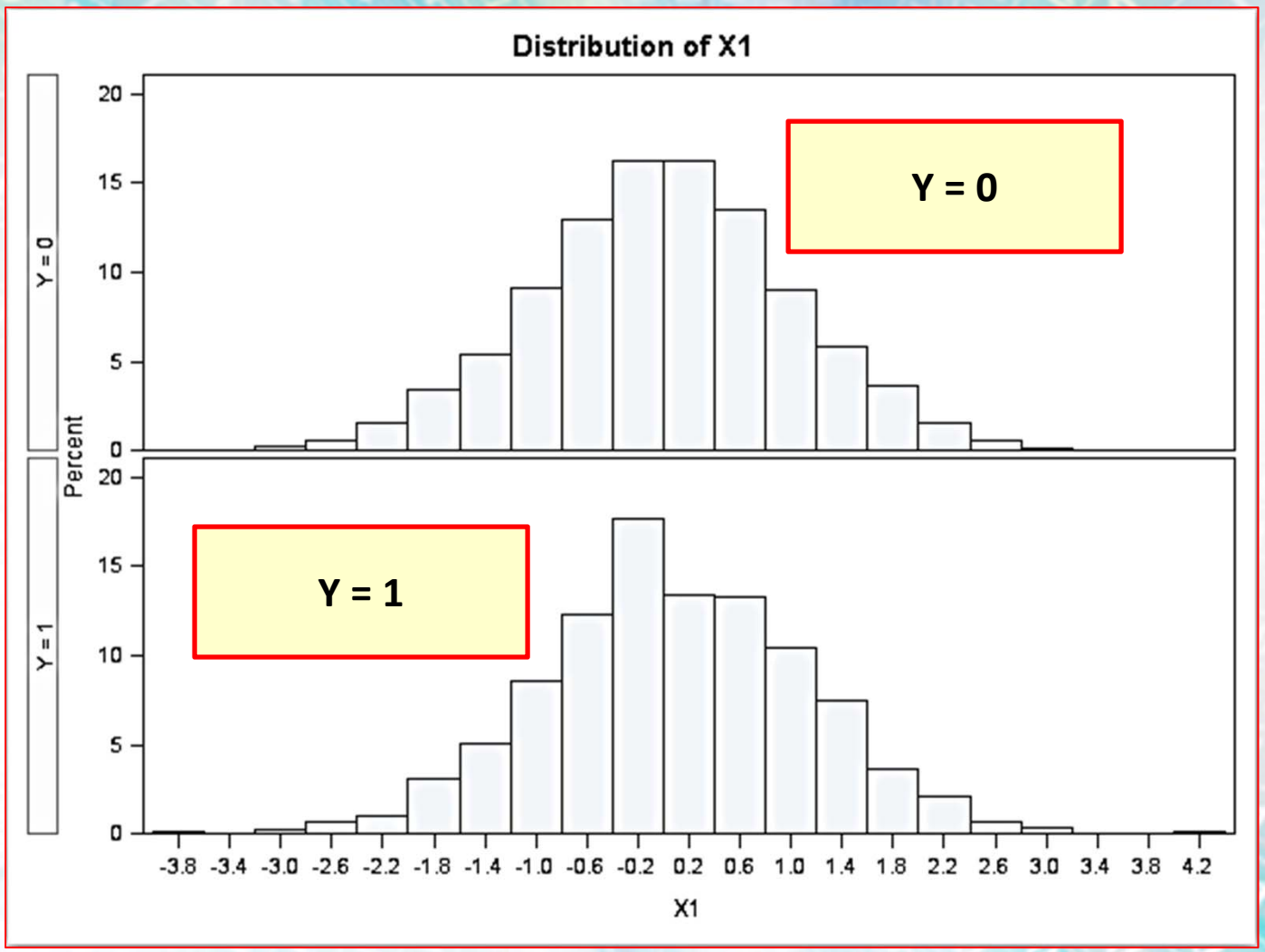
Example: "X1"

```
data Example1;
do i = 1 to 4000;
  Y = (ranuni(12345) < .45);
  if Y = 1 then X1 = rannor(12345) + 0.1;
   else if Y = 0 then X1 = rannor(12345);
   output;
  end;
run;
```

The distribution **X1** for the two groups meet the assumptions of normality with equal standard deviations (=1)

The t^2 and the Wald chi-square for **X1** from logistic regression should be nearly identical.

Example: "X1"


Distribution of X1
Y = 0
Y = 1
Percent
20
15
10
5
0
-3.8 -3.4 -3.0 -2.6 -2.2 -1.8 -1.4 -1.0 -0.6 -0.2 0.2 0.6 1.0 1.4 1.8 2.2 2.6 3.0 3.4 3.8 4.2
X1

Example: Chi-squares for "X1"

Prob ChiSq for β_1 from PROC TTEST (t^2)
Prob ChiSq for β_1 from PROC LOGISTIC (Wald)

	t-Test	Logistic	Prob ChiSq t-Test	Prob ChiSq Logistic
X1 Coeff. =	0.07420	0.07422	1.94%	1.95%
Chi-Sq=	5.467	5.455		
Satterthwaite=	5.423			

Example: "X2"

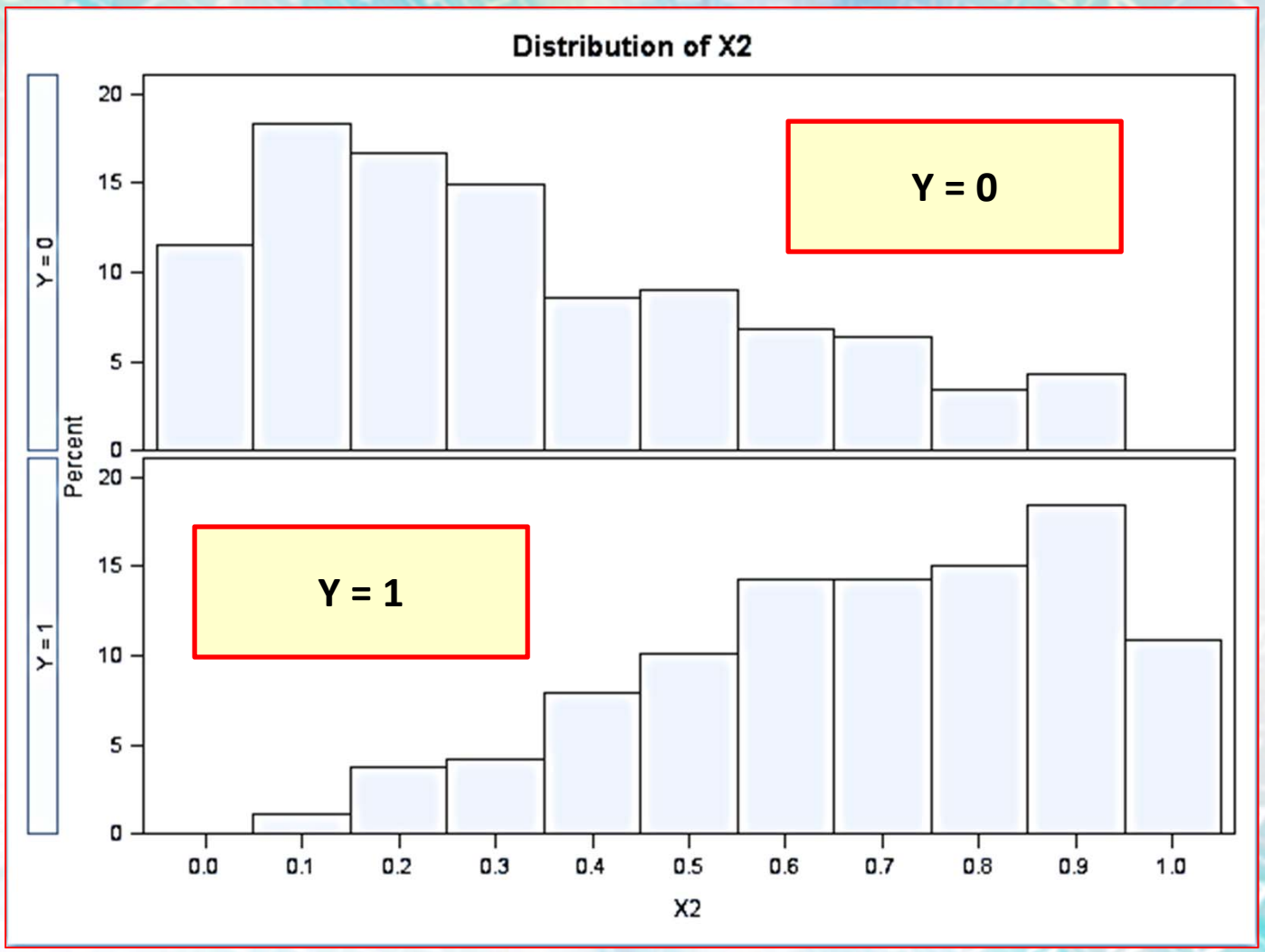
Data Set "Example2" has 500 observations

```
data Example2;
do i = 1 to 500;
  X2 = ranuni(12341);
  Y = floor(X2 + ranuni(12341));
  output;
  end;
run;
```

As will be seen on the next slide, **X2** is not normal over Y=0 or Y=1.

The t^2 and the Wald chi-square for **X2** from logistic regression are not be close in value. **But both are significant**.

Example: "X2"


Distribution of X2
Y = 0
Y = 1
Percent
20
15
10
5
0
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
X2

Example: Chi-squares for "X2"

Prob ChiSq for β_1 from PROC TTEST (t^2)
Prob ChiSq for β_1 from PROC LOGISTIC (Wald)

	t-Test	Logistic	Prob ChiSq t-Test	Prob ChiSq Logistic
X2 Coeff. =	6.2743	5.3491	<.01%	<.01%
Chi-Sq=	275.397	132.515		
Satterthwaite=	271.409			

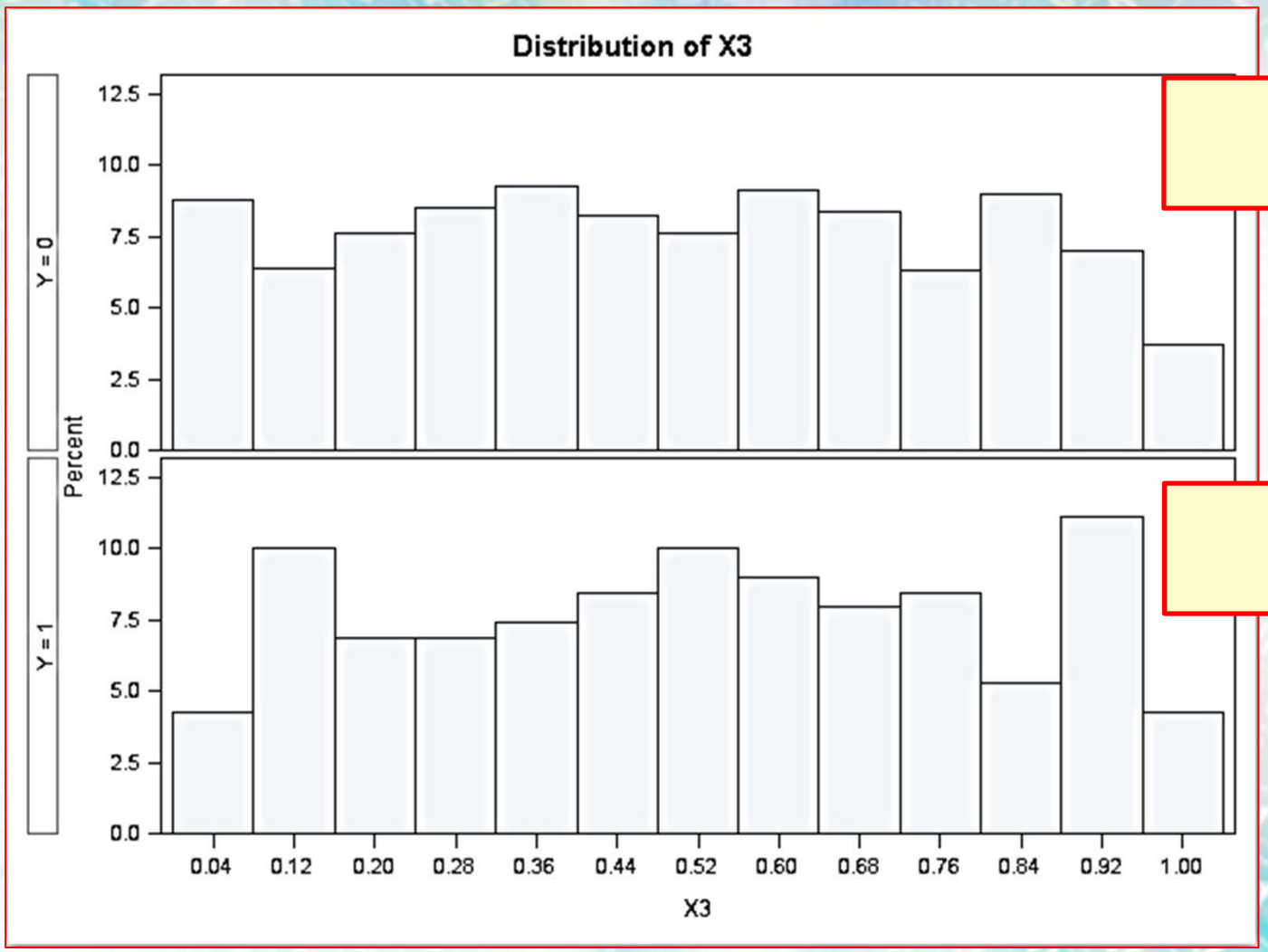
Example: "X3"

Data Set "Example3" has 500 observations

```
data Example3;
do i = 1 to 1000;
  X3 = ranuni(12345);
  Y = (ranuni(12345) < .2);
  output;
  end;
run;
```

X3 is uniform [0, 1] for both Y=0 and Y=1.

Example: "X3"



Y = 0

Y = 1

Example: Chi-squares for "X3"

Prob ChiSq for β_1 from PROC TTEST (t^2)
Prob ChiSq for β_1 from PROC LOGISTIC (Wald)

	t-Test	Logistic	Prob ChiSq t-Test	Prob ChiSq Logistic
X3 Coeff. =	0.35467	0.35550	21.29%	21.29%
Chi-Sq=	1.554	1.552		
Satterthwaite=	1.566			

The t^2 and the Wald chi-square for **X3** from logistic regression are close in value. **And both are in-significant**.

Conclusions

Three examples do not prove a general principle. But

- The risk in judging a truly significant predictor as insignificant (false negative) by using the t-test approach appears to be low based on these and other simulations.
- ***Applied Logistic Regression*** by Hosmer, Lemeshow, and Sturdivant (2013): "if X has approx. normal distribution for both of groups, t-test is a good guide for screening a predictor for logistic regression."

Let's proceed to discuss %LOGIT_CONTINUOUS:

- Creates 11 transformation of a predictor X
- Uses t-test idea to determine significance
- Processes any number of X and with only 3 passes of the data set.

%LOGIT_CONTINUOUS(example, Y, X1 – X50)

Step 1: **translate** each X, if min X < 1, so that min X = +1

Step 2: Create 11 **transformations** (including X)

- **8 monotonic** (monotonic since X >= 1)

  X^p where p is in S = {-2, -1, -0.5, 0, 1, 0.5, 2, 3} where "0" denotes log(x). Includes original X.

- **3 quadratic**:

  $(X\text{-median})^2$, $(X\text{-p25})^2$, $(X\text{-p75})^2$ where median, p25, and p75 are respectively the 50th, 25th, and 75th percentiles for X.

Step 3: Compute **CHI-SQ**s for X and transforms (using square of t-stat)

Three Passes of Data: PROC MEANS, DATA step, PROC TTEST

The X^p are the "fractional polynomials".

%LOGIT_CONTINUOUS example

%LOGIT_CONTINUOUS(example, Y, X1-X50)

Only showing **X1** …

Transform	Coeff est. of β_1	ChiSq	Prob ChiSq		(Not Given by %LOGIT_CONTINUOUS) Prob Wald ChiSq
x**3	0.001	6.793	0.919%		0.932%
(x-p25)**2	0.010	6.424	1.130%		1.166%
x**2	0.089	6.268	1.233%		1.243%
linear	**0.369**	**5.467**	**1.943%**		1.951%
x**0.5	0.368	4.945	2.622%		2.632%
log(x)	0.042	4.325	3.761%		3.775%
x**-0.5	-1.410	3.593	5.811%		5.836%
x**-1	-1.281	2.739	9.798%		9.851%
(x-p50)**2	-1.688	1.958	16.179%		16.237%
x**-2	-0.021	0.881	34.809%		35.036%
(x-p75)**2	0.022	0.331	56.510%		56.639%

%LOGIT_CONTINUOUS example

%LOGIT_CONTINUOUS(example, Y, X1-X50)

Only showing **X1** …

Transform	Coeff est. of β_1	ChiSq	Prob ChiSq		(Not Given by %LOGIT_CONTINUOUS) Prob Wald ChiSq
x3**	**0.001**	**6.793**	**0.919%**		0.932%
(x-p25)**2	0.010	6.424	1.130%		1.166%
x**2	0.089	6.268	1.233%		1.243%
linear	0.369	5.467	1.943%		1.951%

Continue with X1? Guidelines might be:

Prob ChiSq < 1.0% …. vs. 0.919%

ChiSq > 10.0 … vs. 6.793

We will continue with X1 -- although borderline, at best.

Can Translations improve %LOGIT_CONTINUOUS?

A different "best transform" of X might be found after a small translation of X.

```
data Example4;
do i = 1 to 500;
        X4 = ranuni(12341) + 1;
        Y = floor(X4 + ranuni(12341));
X4_1 = X4 + 1;
        output;
        end;
run;
%LOGIT_CONTINUOUS(translation, Y, X4 X4_1);
```

Best: $X^{-0.5}$

Best: X^{-1}

… Probably Not

Could a "good" X be found that would otherwise be missed if **translations** were added to %LOGIT_CONTINUOUS to form **translation-transforms** combinations?

Unlikely …

The maximum of the chi-square for X and its 10 transformations is not greatly affected by translations. Recall example on prior slide:

X4: chi-sq. = 281.23 X4_1: chi-sq. = 281.47 X4_80: chi-sq. = 275.89

Note: X (linear) and (X-median)^2, (X-p25)^2, (X-p75)^2 are not affected by translations.

Part II: The Starting Point

Multivariate Model-building by Royston and Sauerbrei (2008) discusses the Function Selection Procedure (**FSP**).

Our Broad Outline:

- Screening of predictor variables (dozens or hundreds):
  - **Discrete / nominal**: Information value, c-statistic, chi-square
  - **Continuous:** PART I of talk
- Transforming of predictor variables:
  - **Discrete / nominal**: Binning and weight-of-evidence
  - **Continuous:** **PART II of talk**

FSP was developed by Royston, Altman, Sauerbrei, others in the mid 1990's

Part II: Function Selection Procedure

First, translate (if needed) to make X be positive.

Fractional Polynomials (FP) are used to find a transform for X:

$X^{\mathbf{p}}$ for **p** in S = {-2, -1, -0.5, 0, 0.5, 1, 2, 3} where "0" denotes log(x)

FP1 refers to collection of functions formed by selection of one X^p. There are 8 FP1 functions

$g(X,p) = \beta_0 + \beta_1 X^p$ ← 8

FP2 refers to collection of functions formed by selection of two X^p. There are 36 FP2 functions

$G(X,p_1,p_2) = \beta_0 + \beta_1 X^{p1} + \beta_2 X^{p2}$ $\quad p_1 \neq p_2$ ← 28

$G(X,p_1,p_1) = \beta_0 + \beta_1 X^{p1} + \beta_2 X^{p1} \log(X)$ $\quad p_1 = p_2$ ← 8

Part II: Function Selection Procedure

Recall: p is in S = {-2, -1, -0.5, 0, 0.5, 1, 2, 3} where 0 means LOG

The 44 (8 FP1) and (36 FP2) are the possible transformations that FSP can select for X.

The FP1 are monotonic transforms of X (since X is positive)

Example:

$$g(X,-2) = \beta_0 + \beta_1 X^{-2}$$

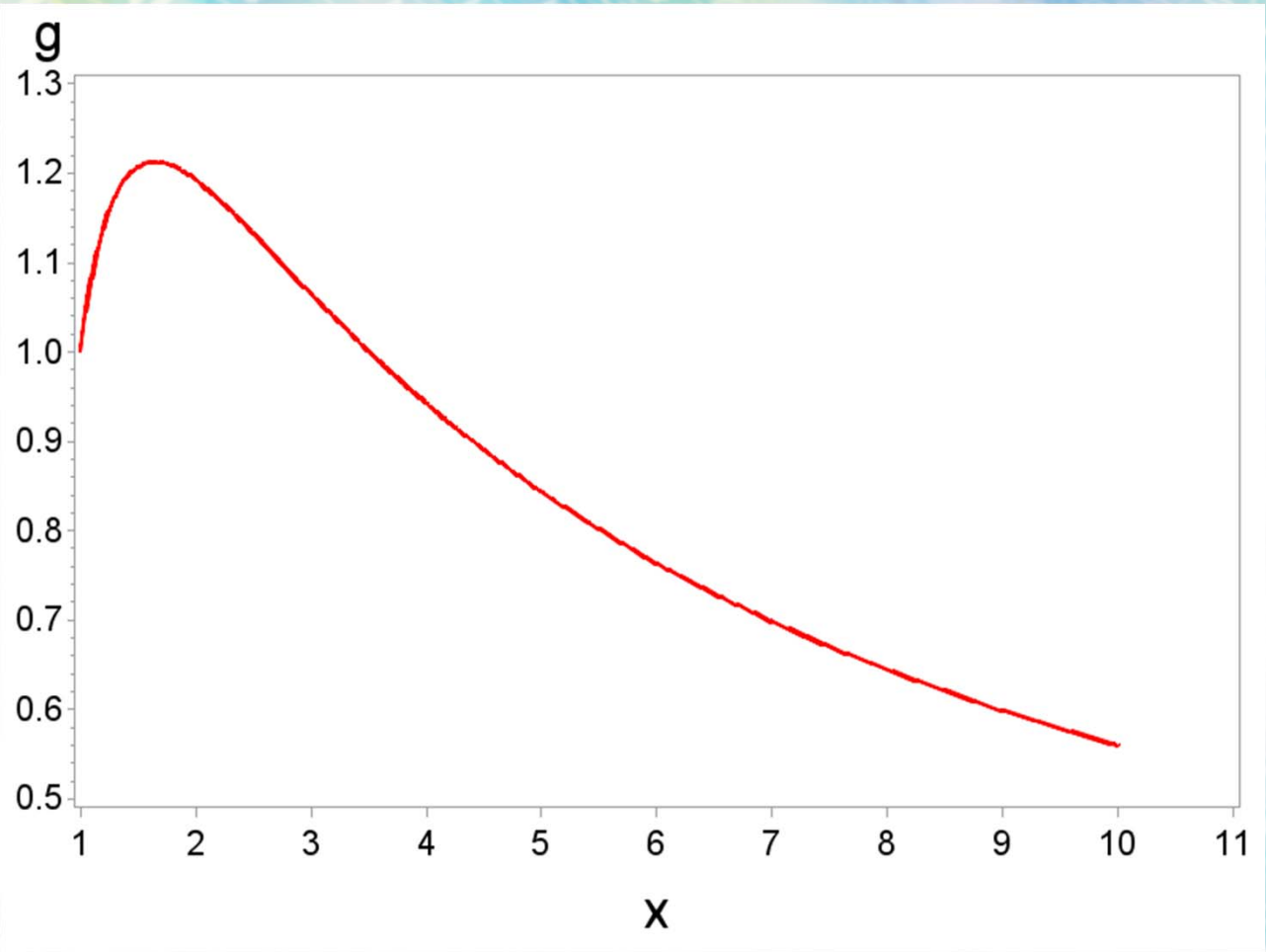
The FP2 transforms can produce non-monotonic curves.

Example:

$$G(X,-1,-1) = \beta_0 + \beta_1 X^{-1} + \beta_2 X^{-1} \log(X)$$

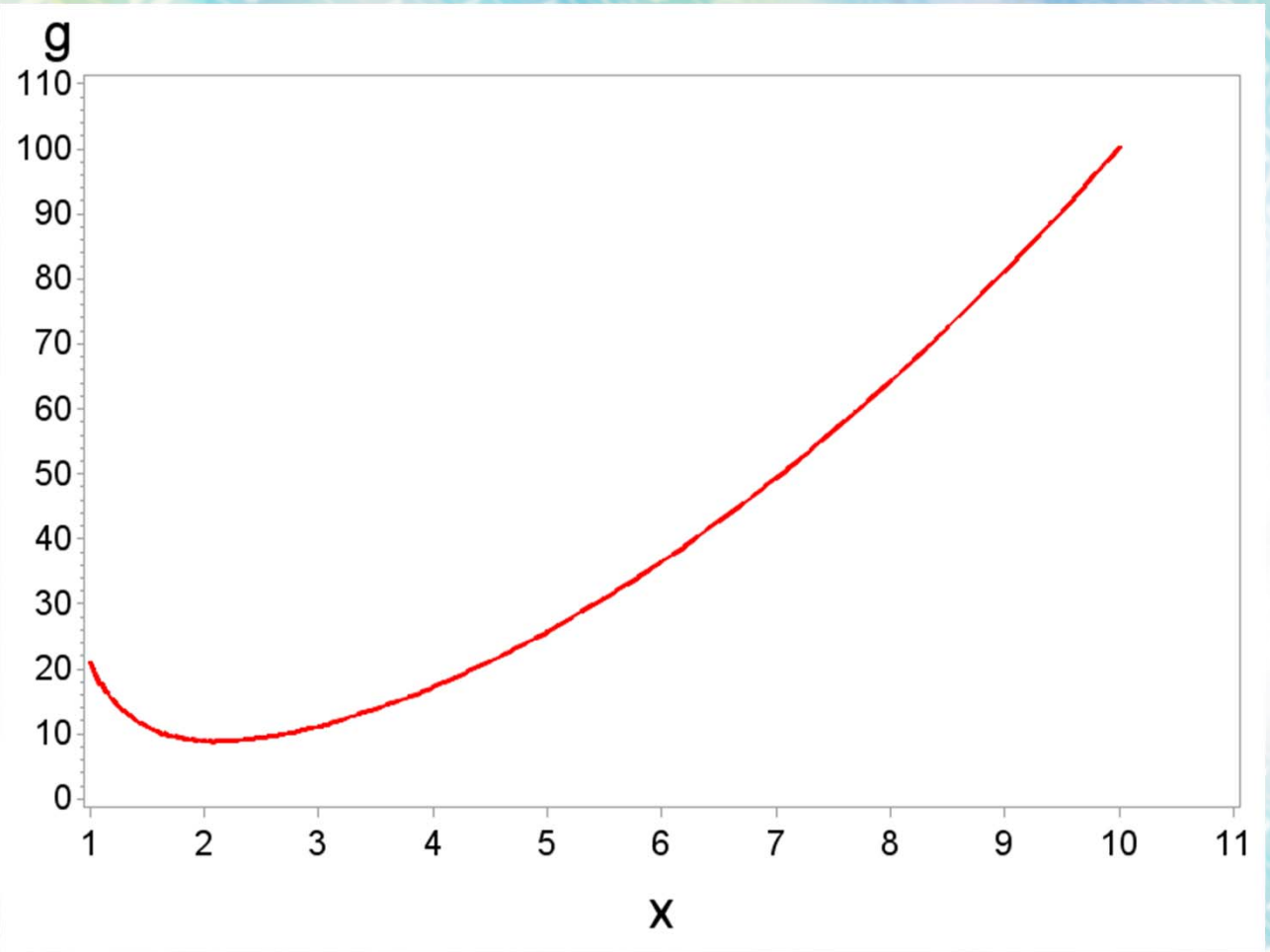
See graphs on next slides.

Non-Monotonic FP2 Functions



$$G(X,-1,-1) = X^{-1} + 2\,X^{-1}\log(X)$$

Non-Monotonic FP2 Functions



$$G(X,+2,-2) = X^2 + 20\ X^{-2}$$

Function Selection Procedure

FSP SAS macro MFP8 at: http://portal.uni-freiburg.de/imbi/mfp

(Also can download R and Stata versions)

MFP8 performs the Function Selection Procedure on a predictor X

Given X and binary Y for logistic regression, **MFP8** macro does this:

➔ Finds the FP1 and FP2 with **best log likelihoods**

- This is done by an exhaustive search. All 44 Logistic Models are run. (PROC LOGISTIC is run a total of 47 times.)

➔ Performs a 3-step test to determine the **statistical significance** of the FP1 and FP2 solutions.

- Using these tests a final transformation is selected.

Note: User must pre-translate X, if needed, to be positive.

Function Selection Procedure

Test Stat = { - 2Log(L)$_{restricted}$ } - { - 2log(L)$_{full}$ } ~ Chi-Sq.

1. Performs a 4 d.f. test at "α" level of best FP2 against the null model. If test is not significant, drop X and stop, else continue.
2. Performs a 3 d.f. test at "α" of best FP2 against a X (linear). If test is not significant, stop (final model is linear), else continue.
3. Performs a 2 d.f. test at "α" of best FP2 vs. best FP1. If test is significant, the final model is the FP2, otherwise it is the FP1.

See Royston and Sauerbrei's book for a DISCUSSION OF DEGREES OF FREEDOM in these tests (d.f. = **4, 3, 2**).

FSP Report for X2 (translated so min(X2) = +1)

Function	p1	p2	Deviance	Achieved α	D.F.
Null	.	.	691.098	0.000%	4
Linear	.	.	489.001	10.658%	3
First Degree FP1	-2	.	483.540	72.451%	2
Second Degree FP2	-2	3	482.896		

Deviance
= - 2*Log(Likelihood) of the model

1. Best FP2 is $G(X2, -2, +2) = \beta_0 + \beta_1 X2^{-2} + \beta_2 X2^{3}$
2. Best FP1 is $g(X2,3) = \beta_0 + \beta_1 X2^{-2}$
3. Test 1 at α_0 = 5%. DO NOT drop X2 from consideration

- $\chi 2$ = 691.098 – 482.896 = 208.202
- 1- Prob($\chi 2$(208.202,**4**))= **0.000%** < 5% … significant

4. Now Test 2: 10.658% > 5% … not significant at α_0 = 5%
   - Therefore Accept X2 (LINEAR)

Test 2: FP2 vs. LINEAR. If not significant, accept LINEAR.

FSP Solutions for X2

Linear = -7.9343 + 5.3491* X2

P(Y=1) = exp(Linear) / (1 + exp(Linear))

or

Log(P(Y=1)/(P(Y=0)) = Linear = -7.9343 + 5.3491* X2

➔ How do Log(P(Y=1)/(P(Y=0)) v. Linear plot across X2?

See Next Slide …

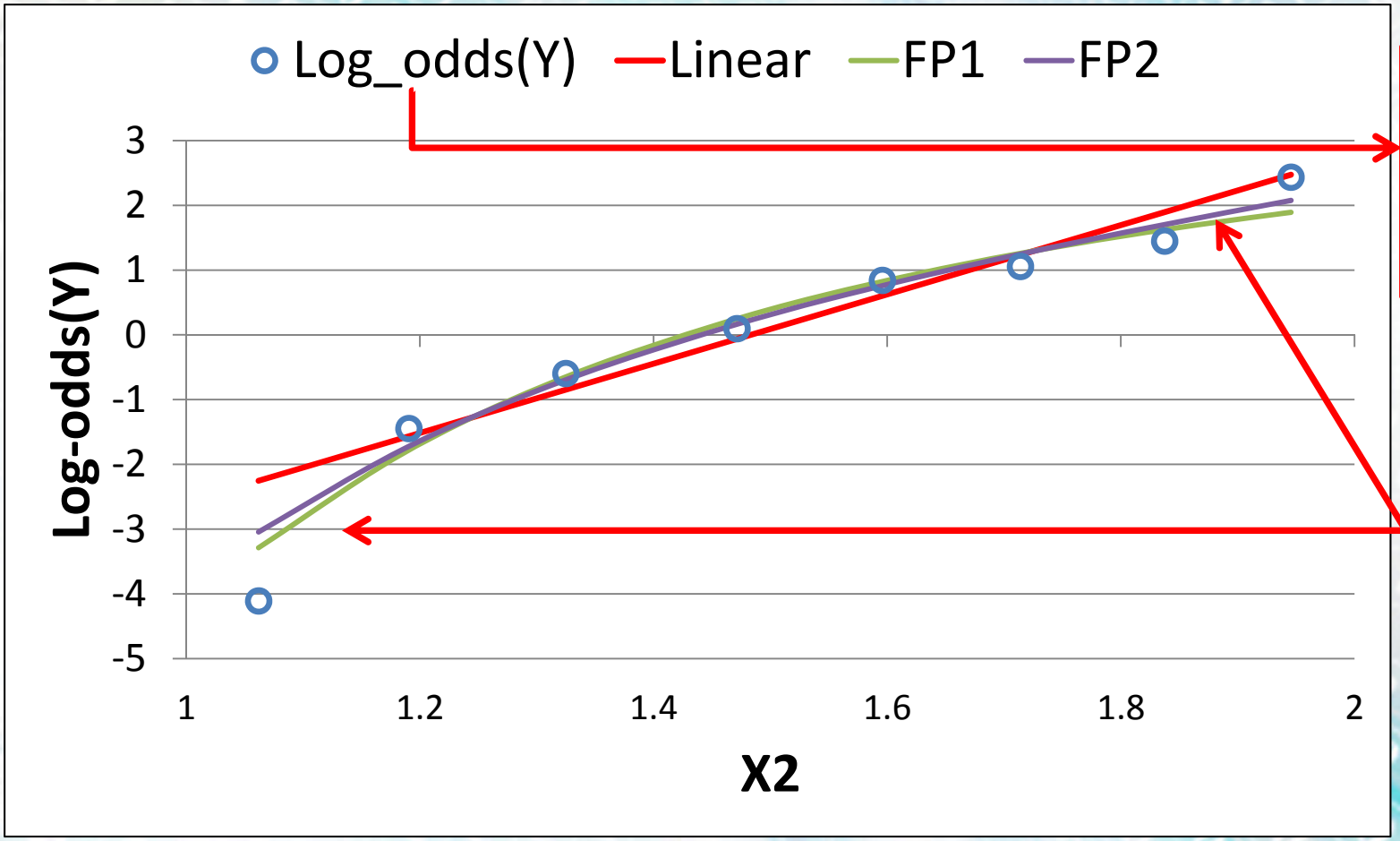
Similarly:

FP1 = 4.0901 + -8.3200 * $X2^{-2}$

FP2 = 2.9666 + -6.9522 * $X2^{-2}$ + 0.1284 * $X2^{3}$

FSP Graphic for X2

8 ranks of X2. Linear, FP1, FP2 computed at median of X2 within rank.

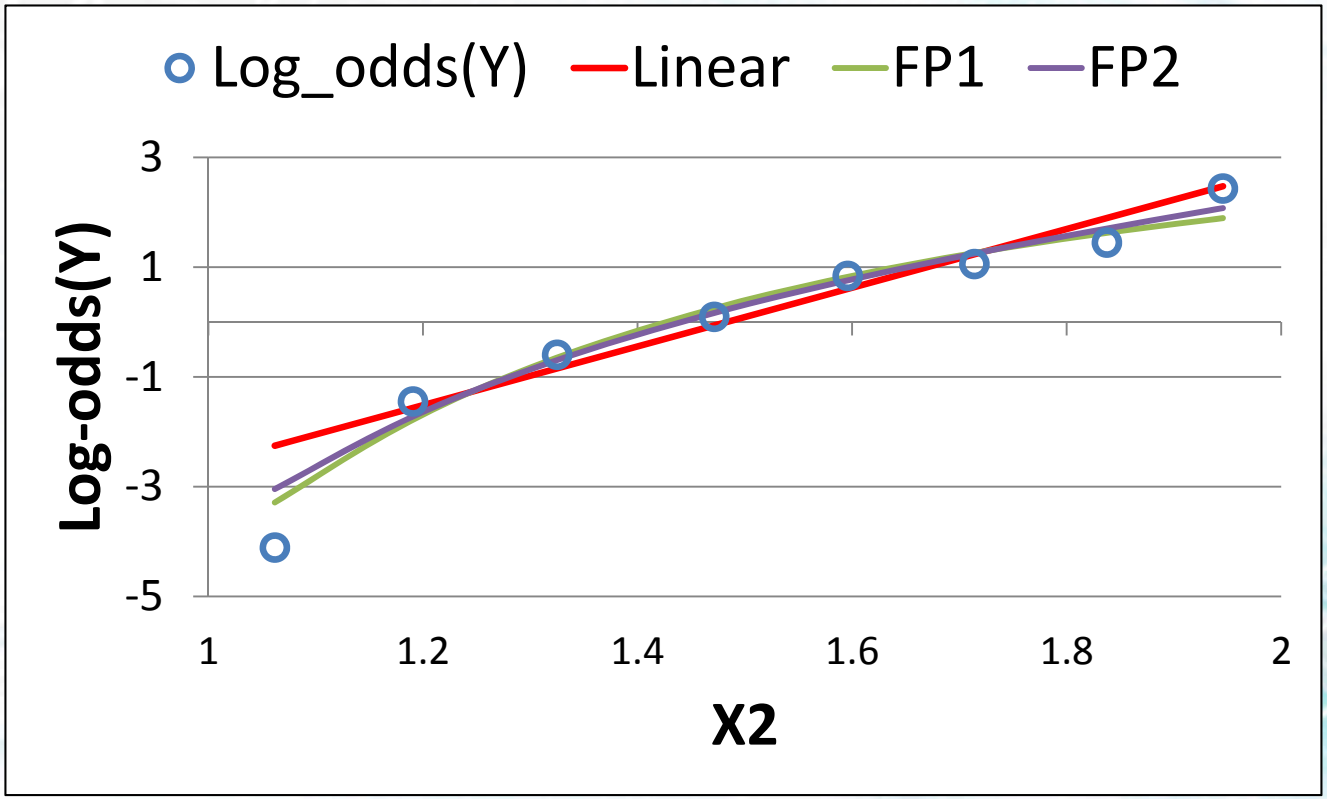


Log-odds(Y) is average of Y within rank.

Visually, FP1 & FP2 give much better fit than Linear.

FSP Graphic for X2

The conservative nature of the FSP 3-step testing rejected FP1 and FP2 at 5% and selected Linear.



Do FP1 & FP2 over-fit the data?

I would use FP1. What would you do?

SAS Macros for FSP

I wondered if the 47 PROC LOGISTIC runs could be reduced.

I wrote an FSP macro that runs PROC LOGISTIC 36 times (the minimum possible, I think).

It runs the 36 FP2 solution candidates through PROC LOGISTIC and picks up the required information about 8 FP1 solution candidates in the process.

For a while I thought I had an FSP macro with 8 PROC LOGISTIC runs. Then I saw it can have a non-optimal FP2.

BUT: if many predictors and a large dataset, this 8 PROC LOGISTIC idea might have merit (it has ~ 17% of run time). See next slide …

SAS Macros for FSP

PROC LOGISTIC; MODEL Y = &Var<k> / SELECTION = FORWARD INCLUDE=**1** START=**1** STOP=**2** SLE=**1**;

- All possible FP2 pairs have a chance to be selected
- But selection of the 2nd variable in a pair by FORWARD, to add to the 1st variable forced in by INCLUDE=1, may be sub-optimal
- Reason: 2nd variable is selected by best Wald χ2 not by LL.

E.g. Consider &Var1. We force in X^{-2} and perhaps Wald picks X^{-1} but best LL is given by X^3. Such examples exist.

Var1=	X^{-2}	X^{-1}	$X^{-.5}$	$X^{.5}$	X	X^2	X^3	Log(X)	X^{-2} Log(X)
Var2=		X^{-1}	$X^{-.5}$	$X^{.5}$	X	X^2	X^3	Log(X)	X^{-1} Log(X)
Var3=			$X^{-.5}$	$X^{.5}$	X	X^2	X^3	Log(X)	$X^{-.5}$ Log(X)
Var4=				$X^{.5}$	X	X^2	X^3	Log(X)	$X^{.5}$ Log(X)
Var5=					X	X^2	X^3	Log(X)	X Log(X)
Var6=						X^2	X^3	Log(X)	X^2 Log(X)
Var7=							X^3	Log(X)	X^3 Log(X)
Var8=								Log(X)	Log(X) Log(X)

Sum Up

1. t-test idea appears to be effective at screening continuous predictor X for power in predicting binary Y
2. %LOGIT_CONTINUOUS implements t-test idea for X and 10 transforms.
   a) These transforms include monotonic and quadratic
      i. Are additional transforms needed?
   b) More simulations to explore chance of false negatives
   c) Translations appear not to add predictive power
3. FSP provides an effective means to select final transform.
   a) Three-step tests for significance
   b) One variable at a time … slow process if many X
   c) Use 8 PROC LOGISTIC method for mass processing X1-X100?
      i. Need to understand freq. / magnitude of sub-optimal FP2.

Contact Information

Name:	Bruce Lund
Enterprise	Magnify Analytic Services, a division of Marketing Associates, LLC
Address:	777 Woodward Ave, Suite 500
City, State ZIP:	Detroit, MI, 48226
E-mail:	blund@marketingassociates.com blund_data@mi.rr.com