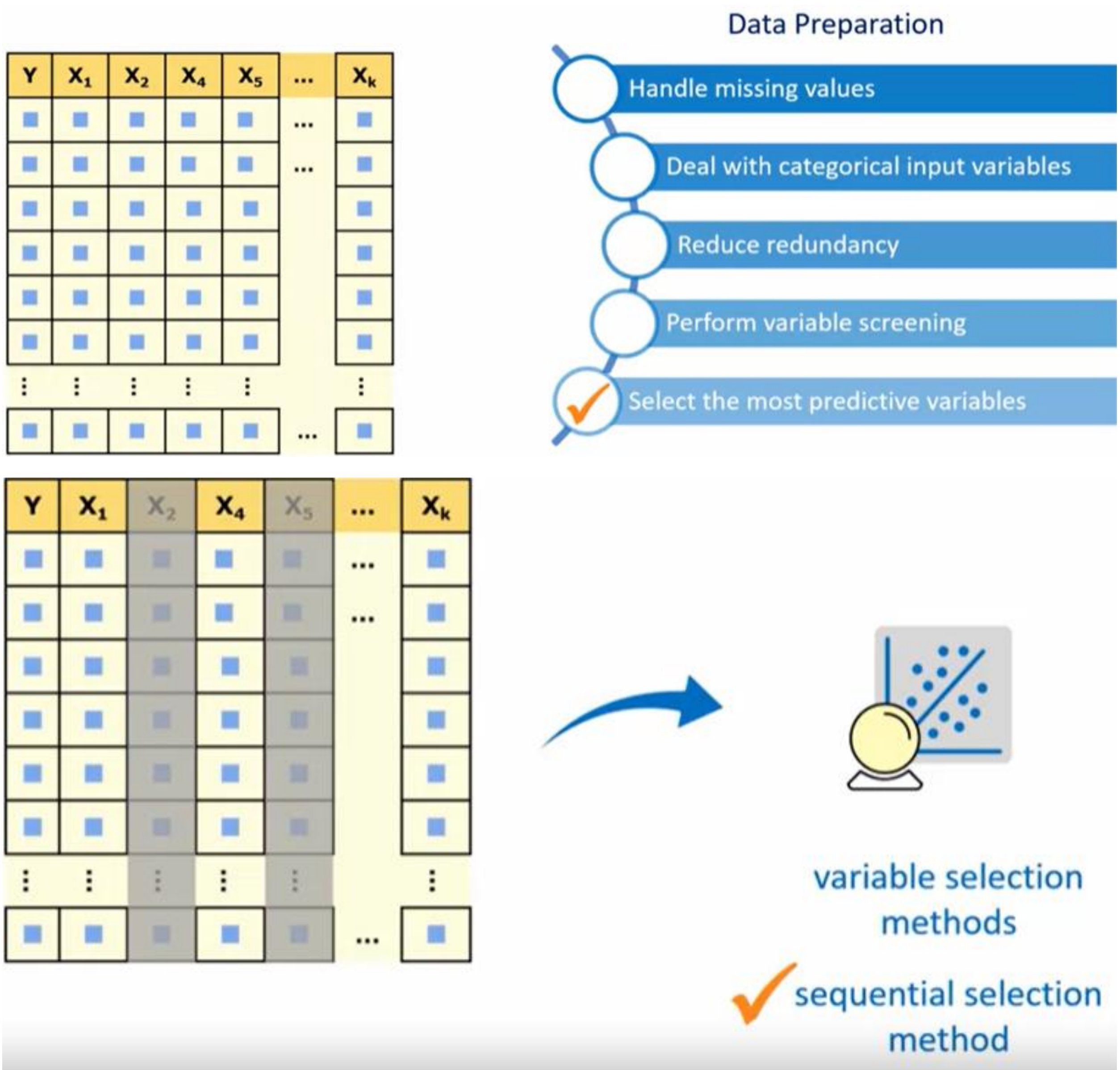
SBA Statistical Business Analyst using SAS

SBA3 Predictive Modeling with Logistic Regression

W4B Selecting Variables Sequentially

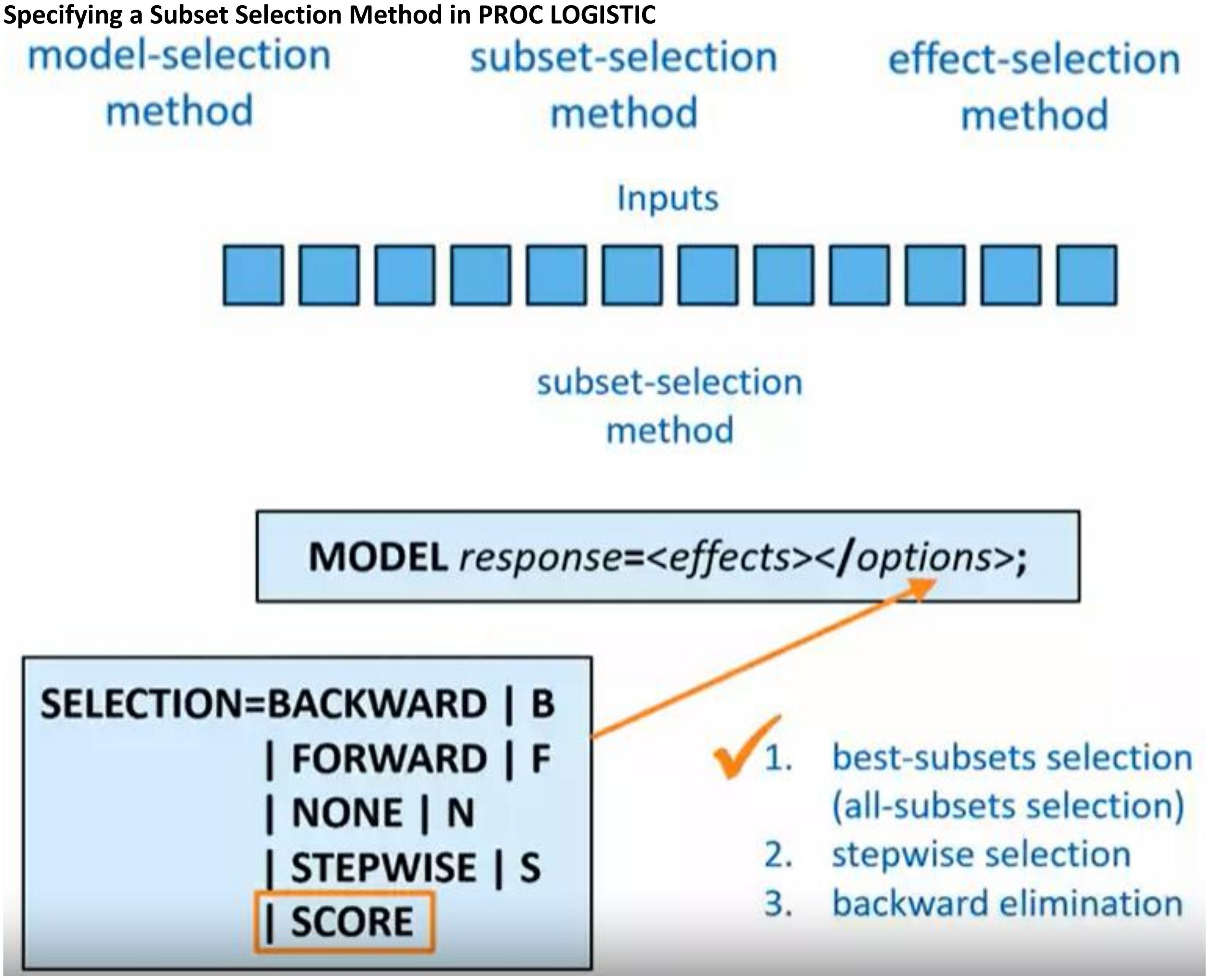
Introduction

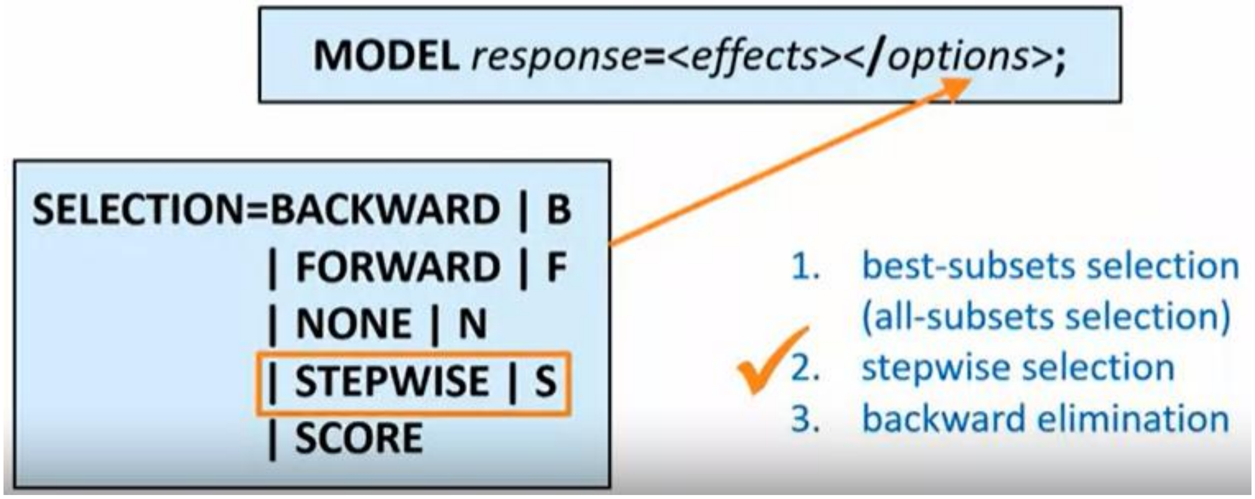


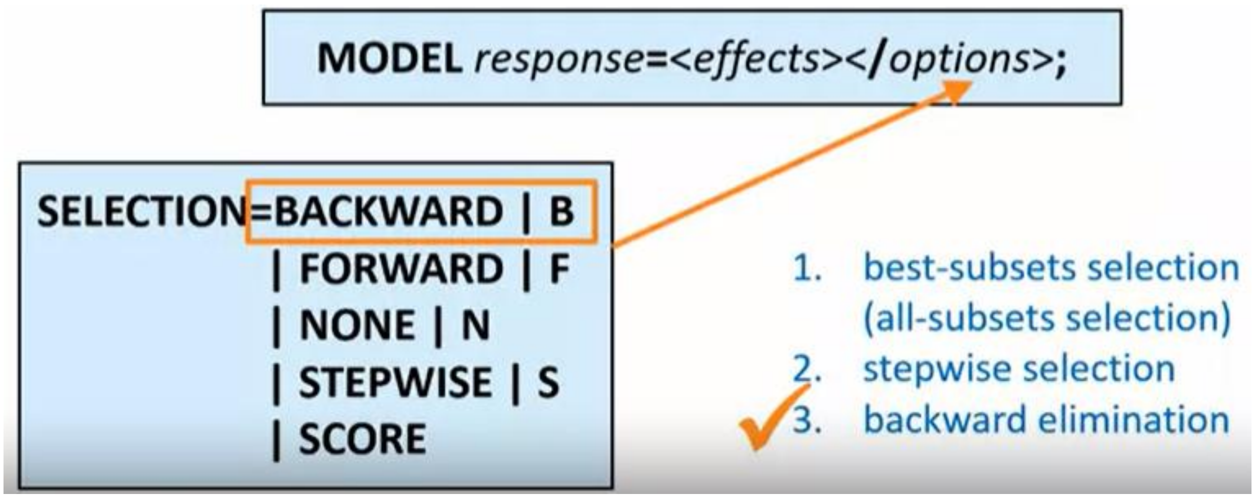
In this topic, you learn to do the following:

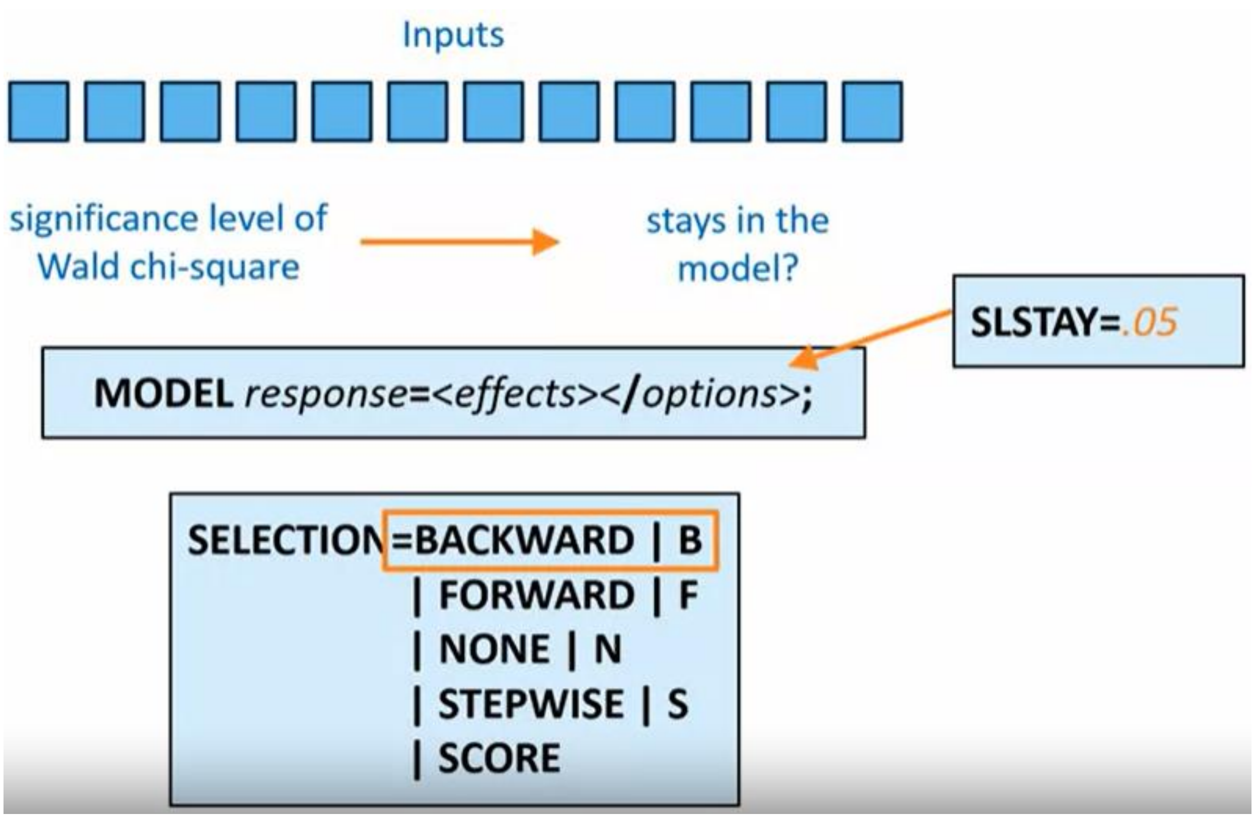
- describe the most commonly used subset selection methods in PROC LOGISTIC
- use a subset selection method to detect interactions
- use a subset selection method to find the subset of variables that minimizes Schwarz's Bayesian criterion

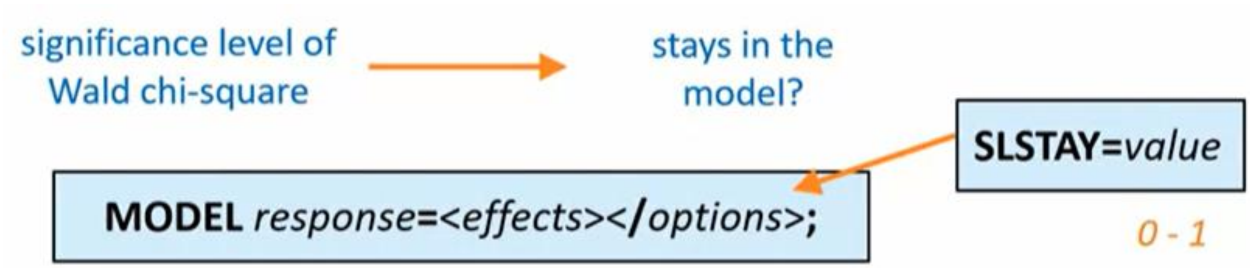
Specifying a Subset Selection Method in PROC LOGISTIC

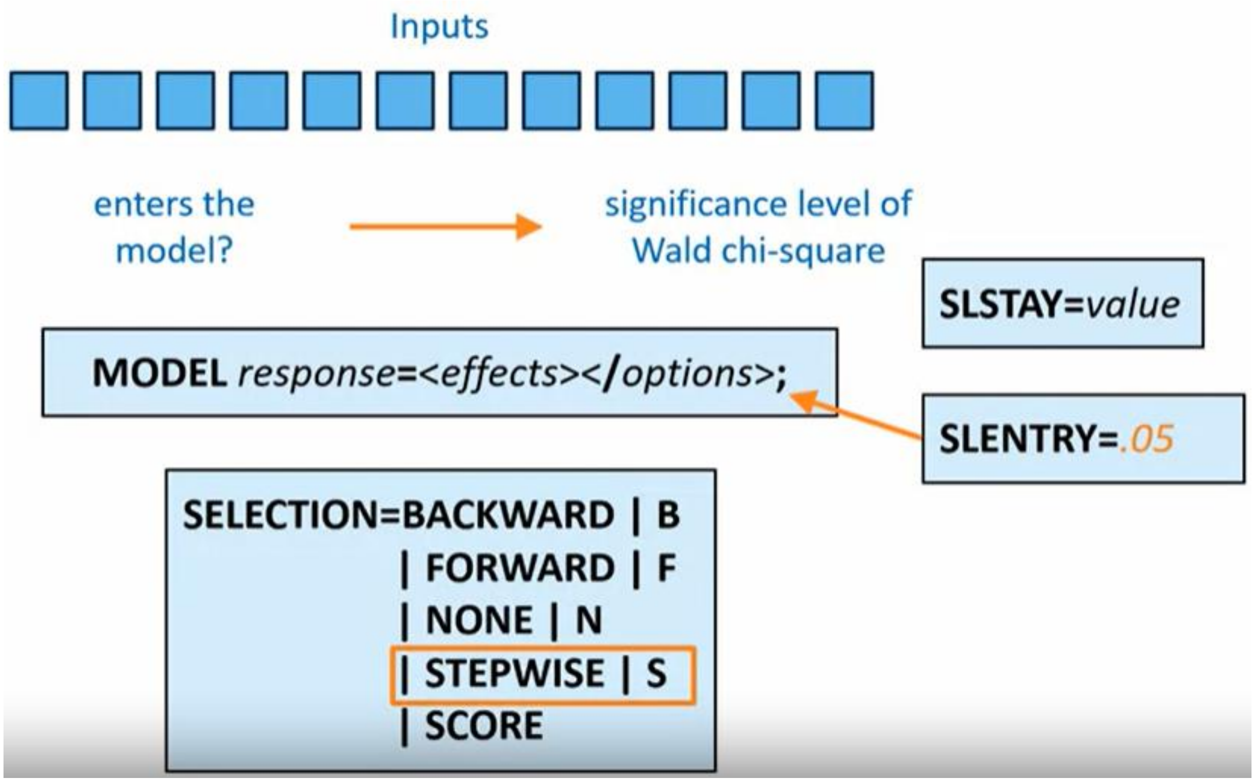


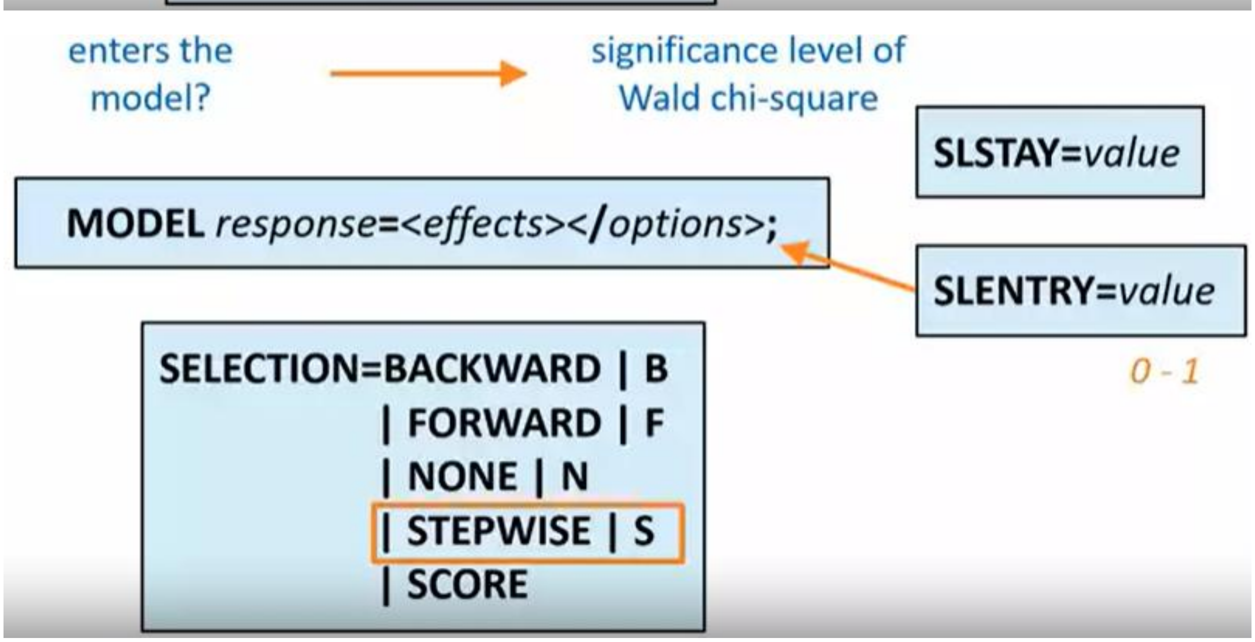

MODEL response=<effects></options>;
SELECTION=BACKWARD | B
| FORWARD | F
| NONE | N
| STEPWISE | S
| SCORE
1. best-subsets selection (all-subsets selection)
2. stepwise selection
3. backward elimination


MODEL response=<effects></options>;
SELECTION=BACKWARD | B
| FORWARD | F
| NONE | N
| STEPWISE | S
| SCORE
1. best-subsets selection (all-subsets selection)
2. stepwise selection
3. backward elimination

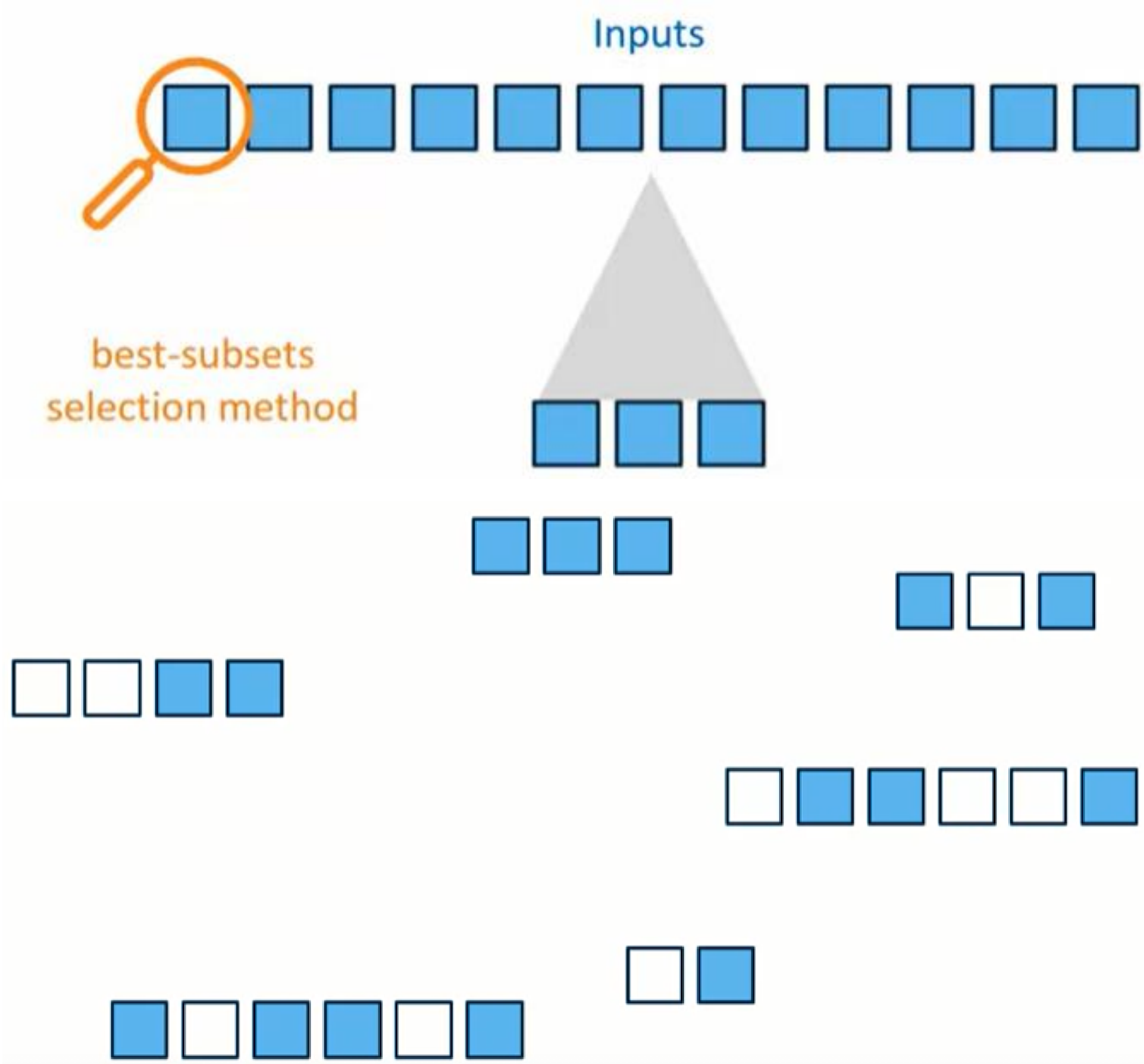

Inputs
significance level of
Wald chi-square
stays in the
model?
SLSTAY=.05
MODEL response=<effects></options>;
SELECTION=BACKWARD | B
| FORWARD | F
| NONE | N
| STEPWISE | S
| SCORE

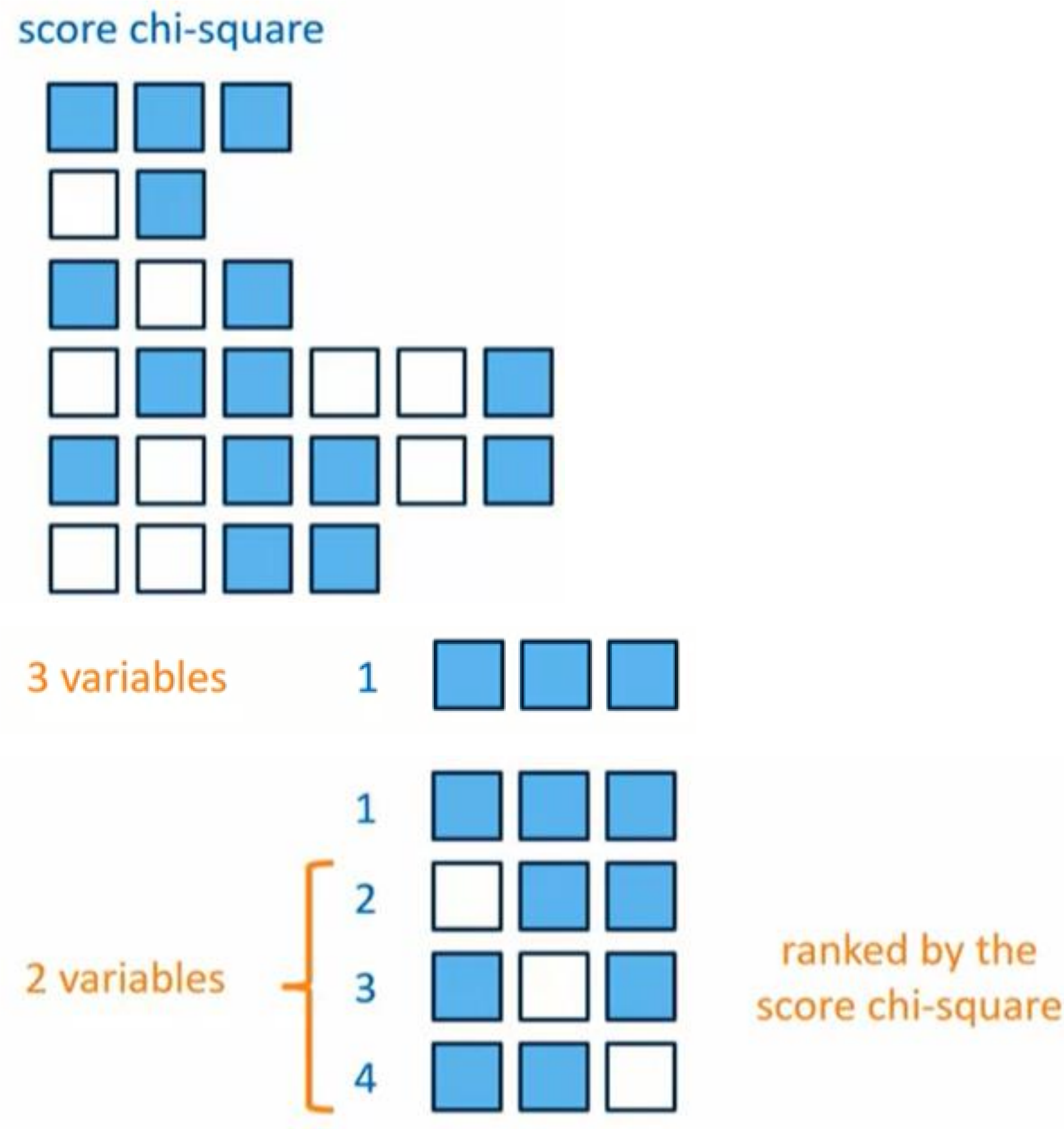

significance level of
Wald chi-square
stays in the
model?
SLSTAY=value
MODEL response=<effects></options>;
0 - 1

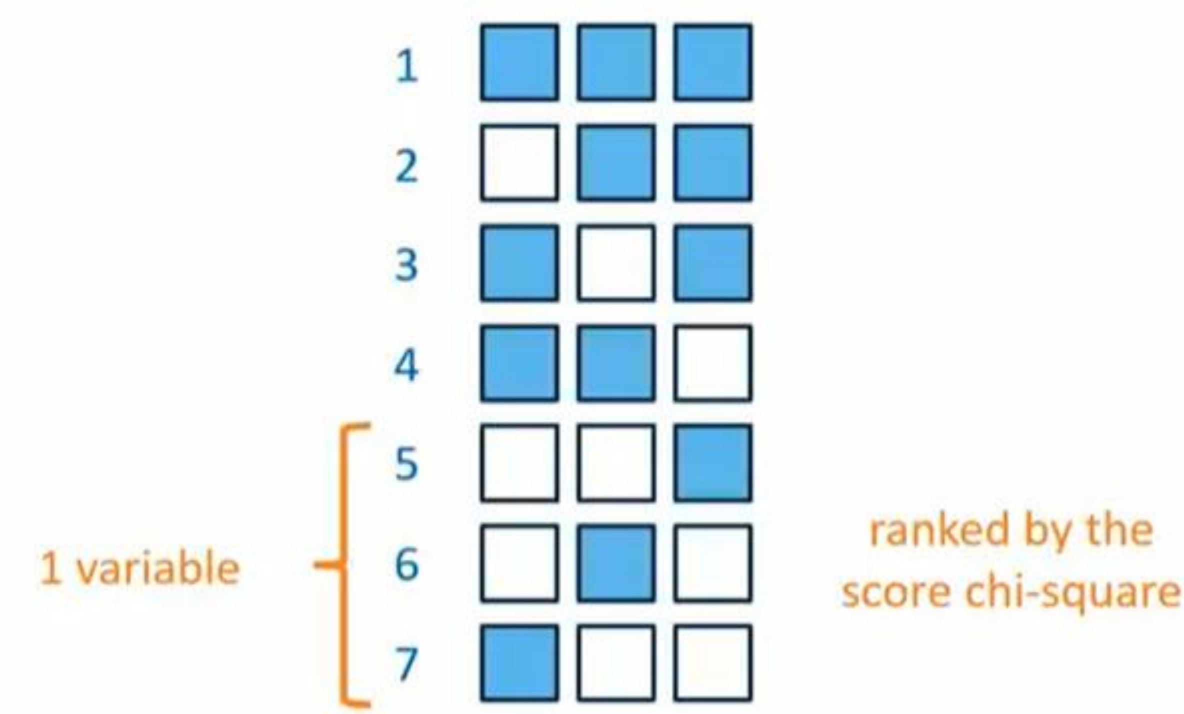

Inputs
enters the model?
significance level of Wald chi-square
SLSTAY=value
MODEL response=<effects></options>;
SLENTRY=.05
SELECTION=BACKWARD | B
| FORWARD | F
| NONE | N
| STEPWISE | S
| SCORE

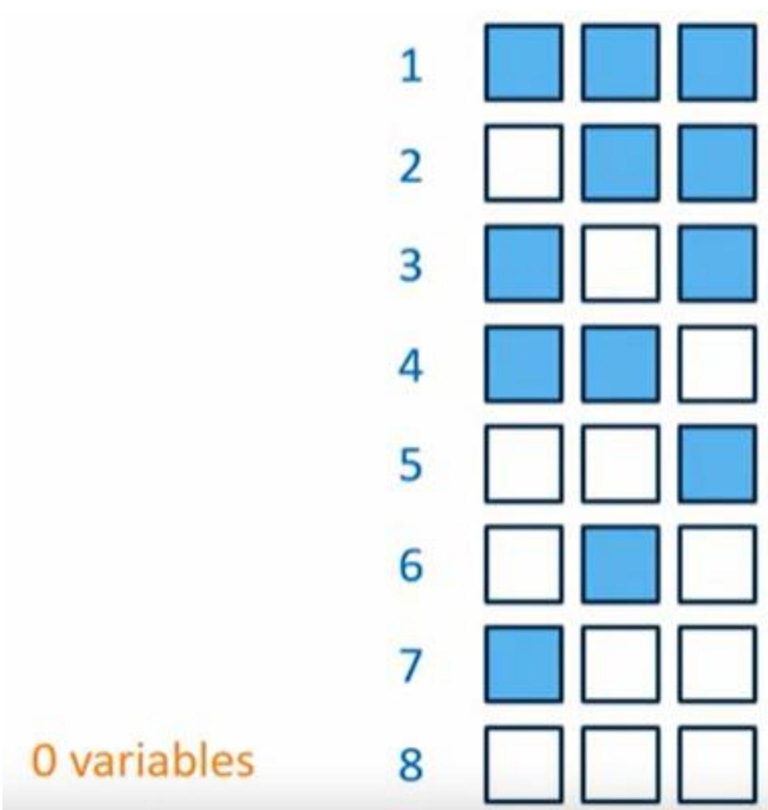

enters the model?
significance level of Wald chi-square
SLSTAY=value
MODEL response=<effects></options>;
SLENTRY=value
0 - 1
SELECTION=BACKWARD | B
| FORWARD | F
| NONE | N
| STEPWISE | S
| SCORE

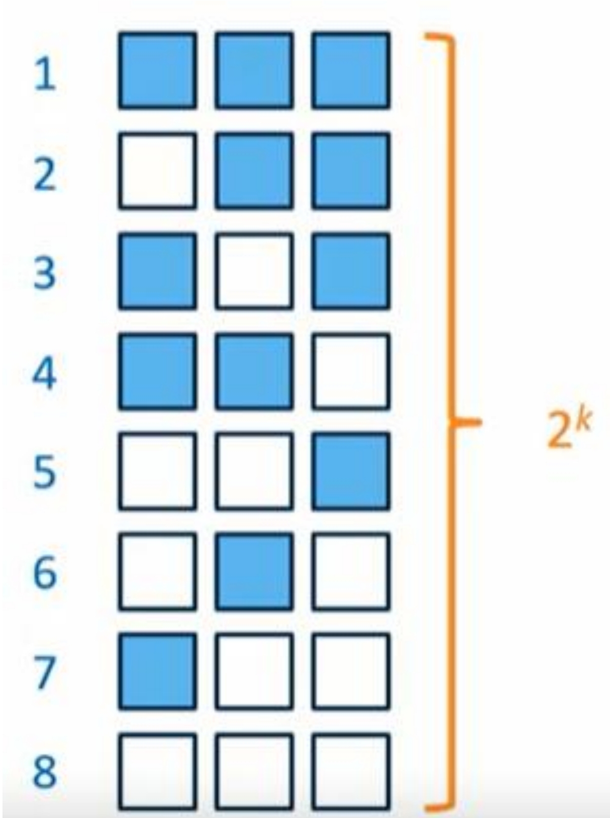
Best-Subsets Selection



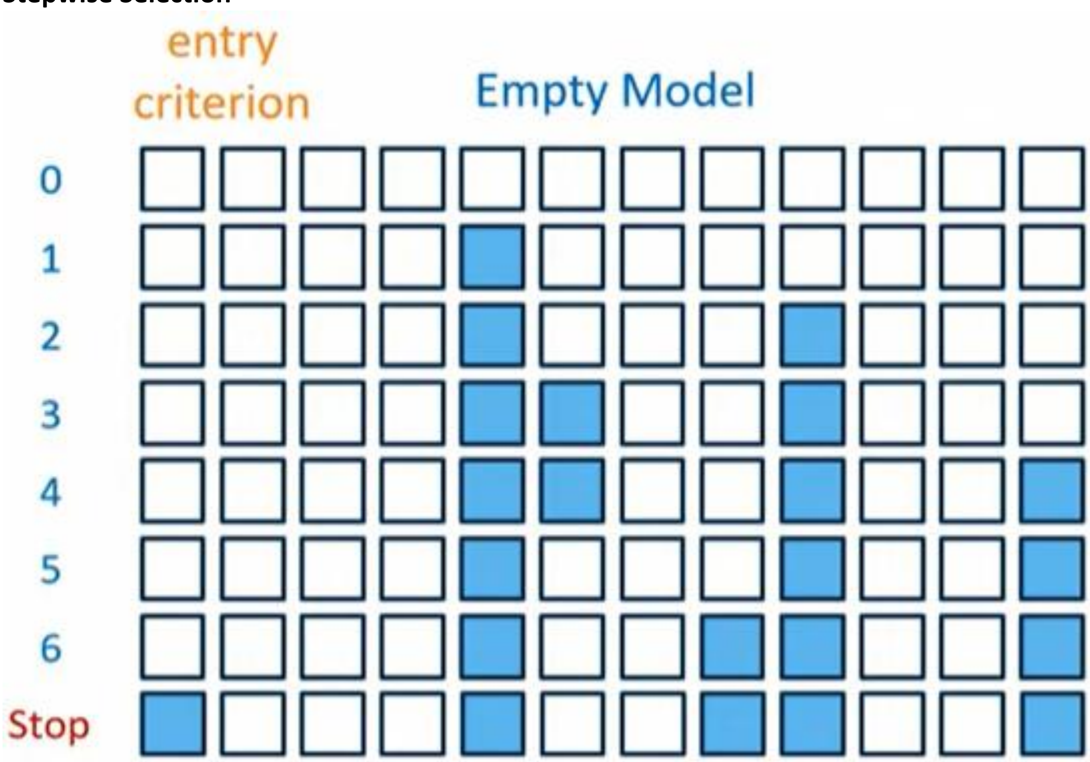

score chi-square
3 variables
1
1
2
2 variables
3
4
ranked by the
score chi-square

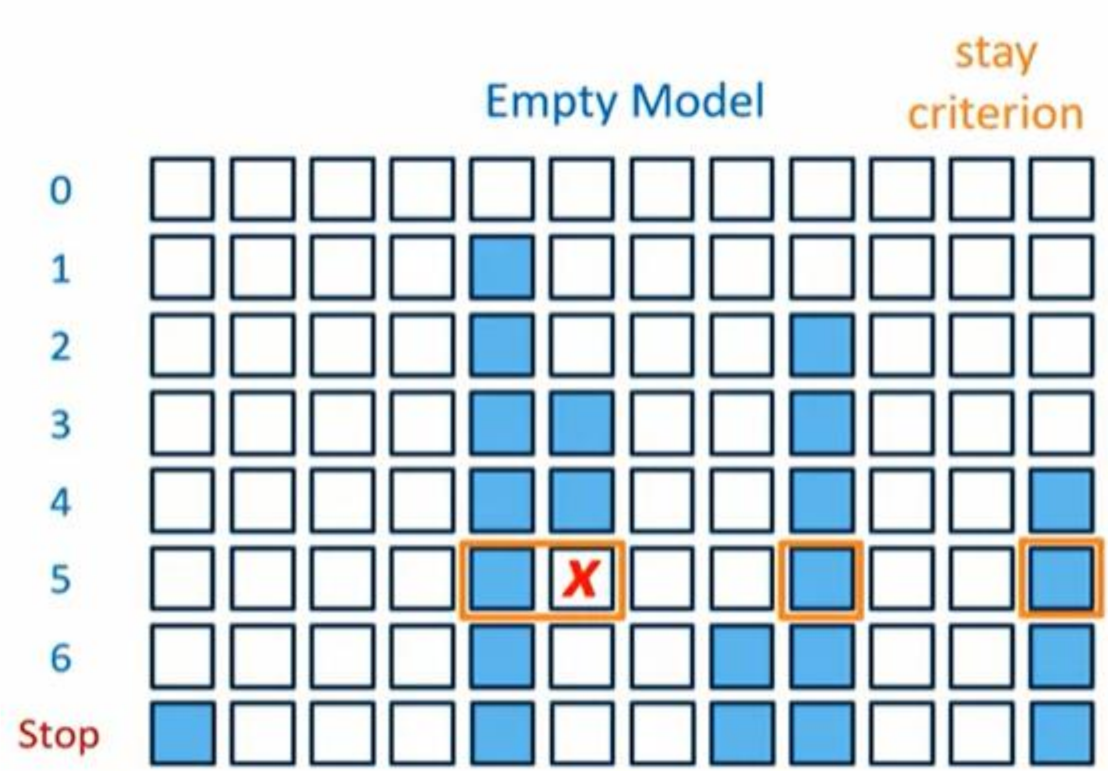

1
2
3
4
5
6
7
1 variable
ranked by the score chi-square


1
2
3
4
5
6
7
8
0 variables

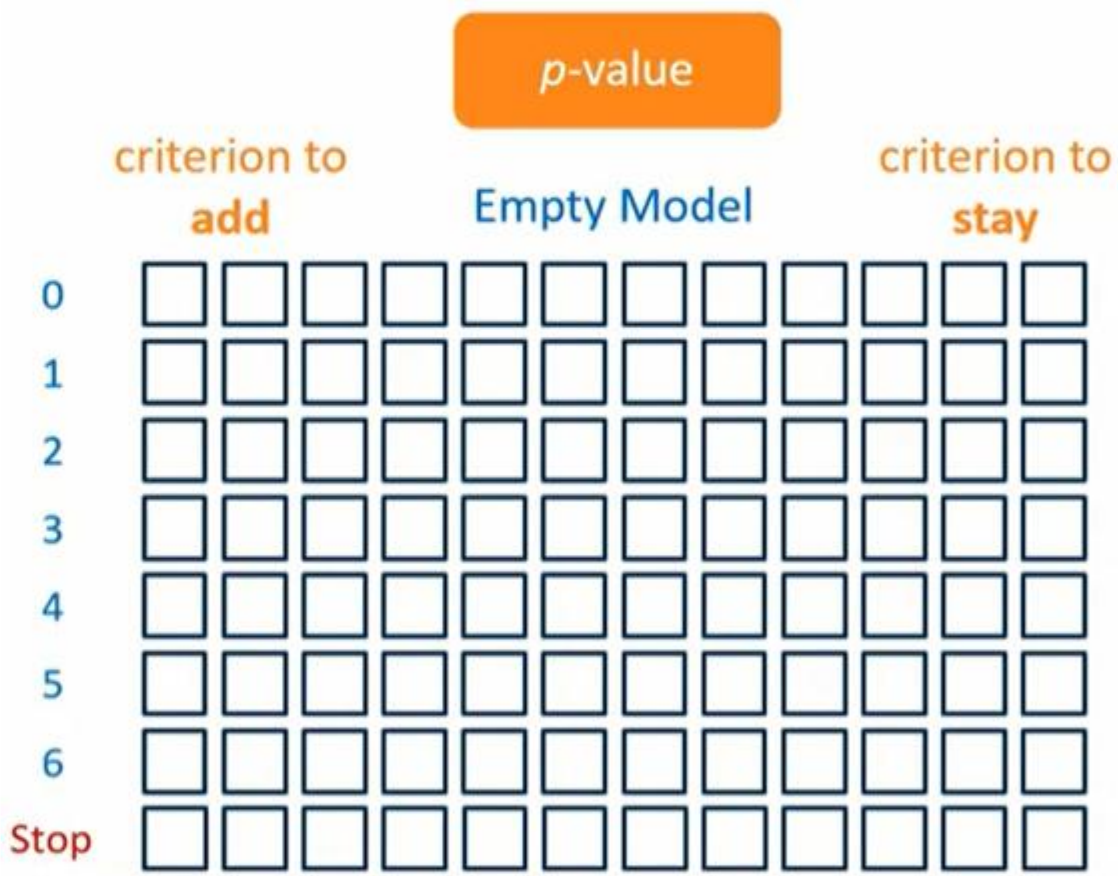


Stepwise Selection

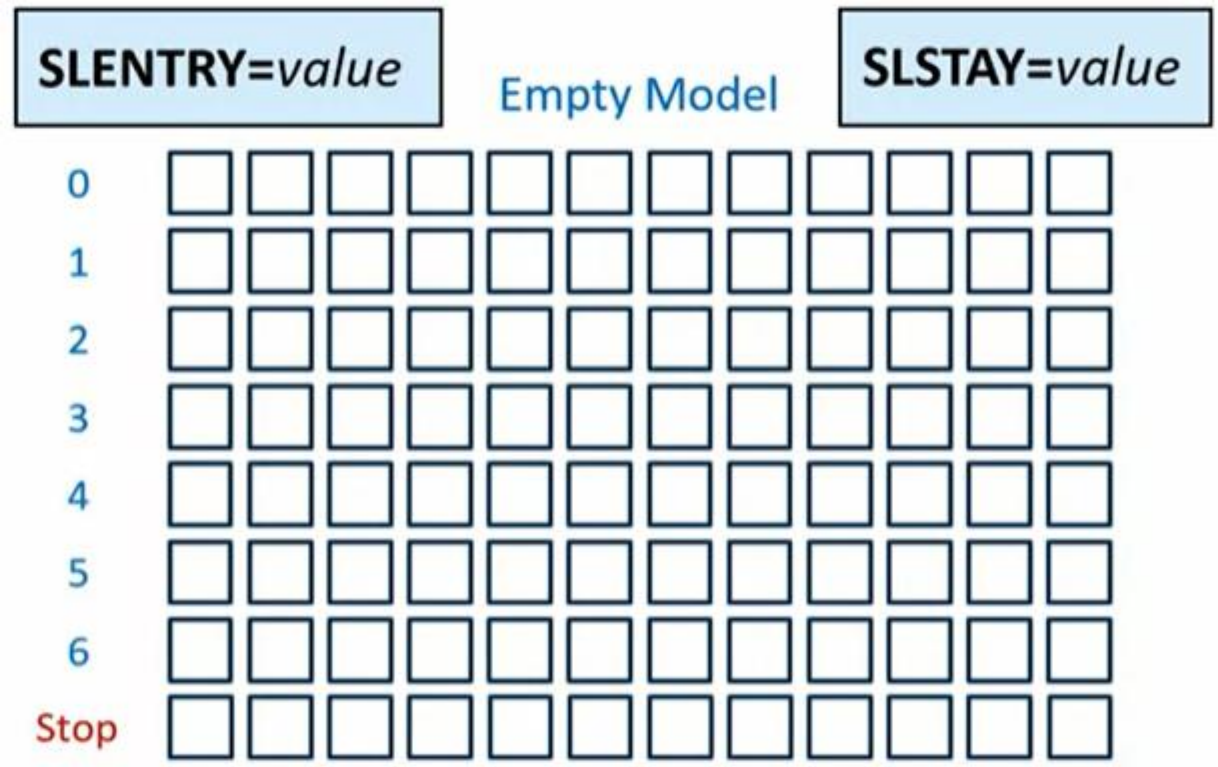


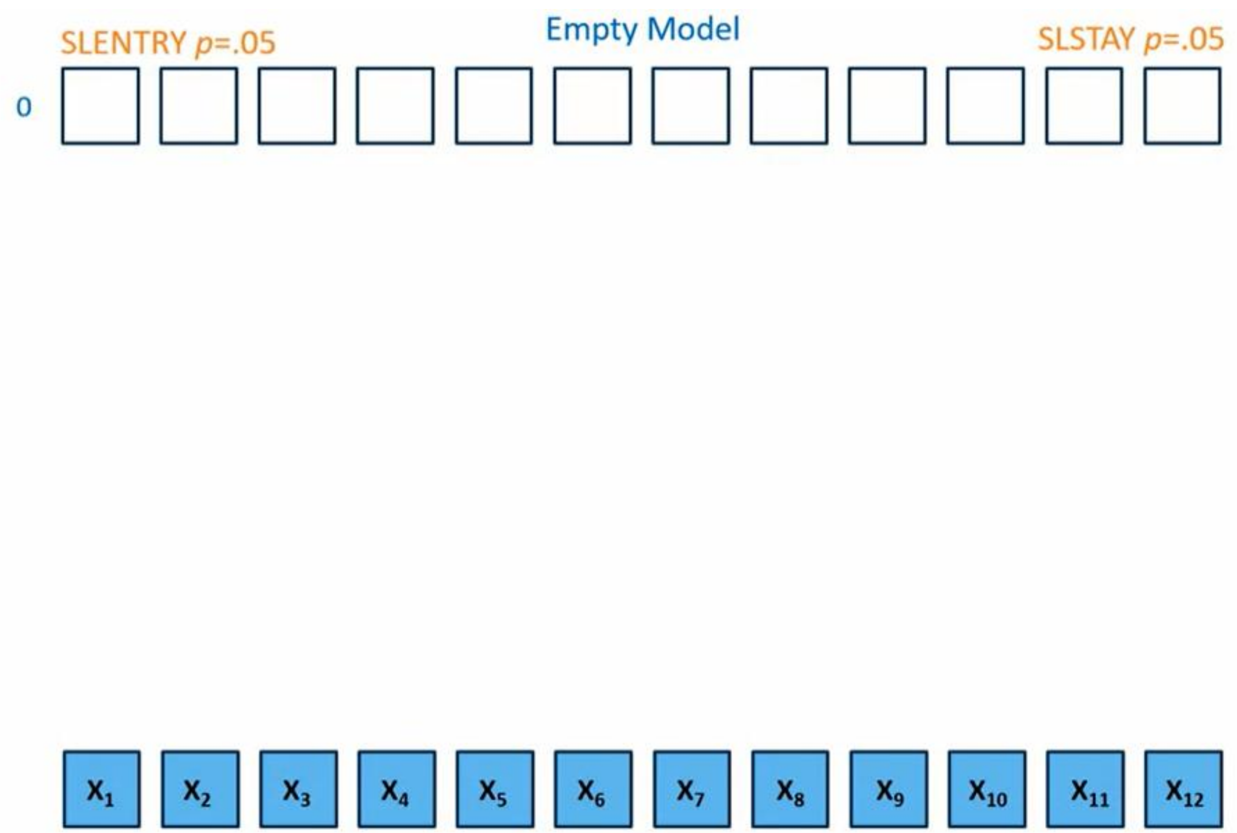


Empty Model
stay
criterion
0
1
2
3
4
5
6
Stop
X

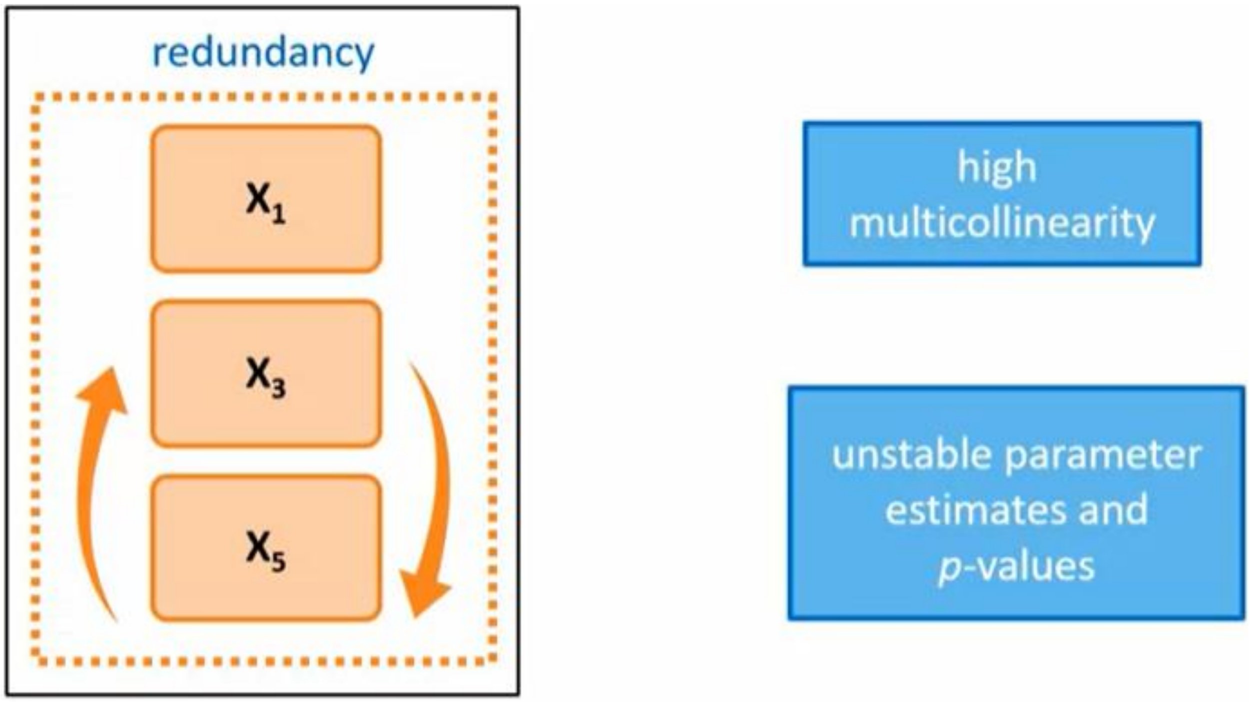


p-value
criterion to
add
Empty Model
criterion to
stay
0
1
2
3
4
5
6
Stop

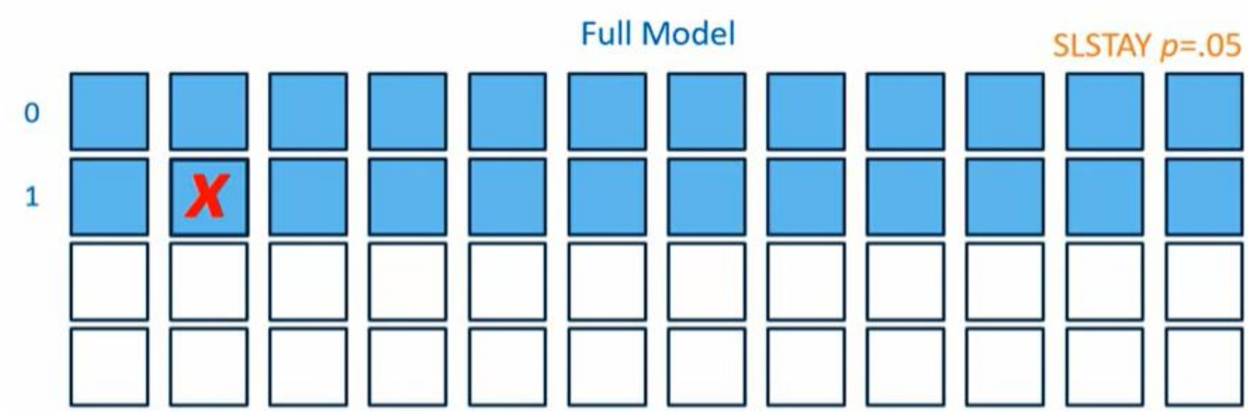

SLENTRY=value
Empty Model
SLSTAY=value
0
1
2
3
4
5
6
Stop


SLENTRY p=.05
Empty Model
SLSTAY p=.05
0
X1
X2
X3
X4
X5
X6
X7
X8
X9
X10
X11
X12


SLENTRY p=.05
SLSTAY p=.05
0
1
2
3
4
5
X_5
X_6
X_9
X_{12}
X_1
X_2
X_3
X_4
X_7
X_8
X_{10}
X_{11}
6
Stop



Backward Elimination



Full Model

SLSTAY p=.05

0	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}
1	X_1		X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}
2	X_1		X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}		X_{12}
3	X_1		X_3		X_5	X_6	X_7	X_8	X_9	X_{10}		X_{12}
4	X_1		X_3		X_5	X_6	X_7	X_8	X_9			X_{12}
5	X_1		X_3		X_5	X_6		X_8	X_9			X_{12}
6	X_1				X_5	X_6		X_8	X_9			X_{12}
Stop	X_1				X_5			X_8	X_9			X_{12}

less inclined to:

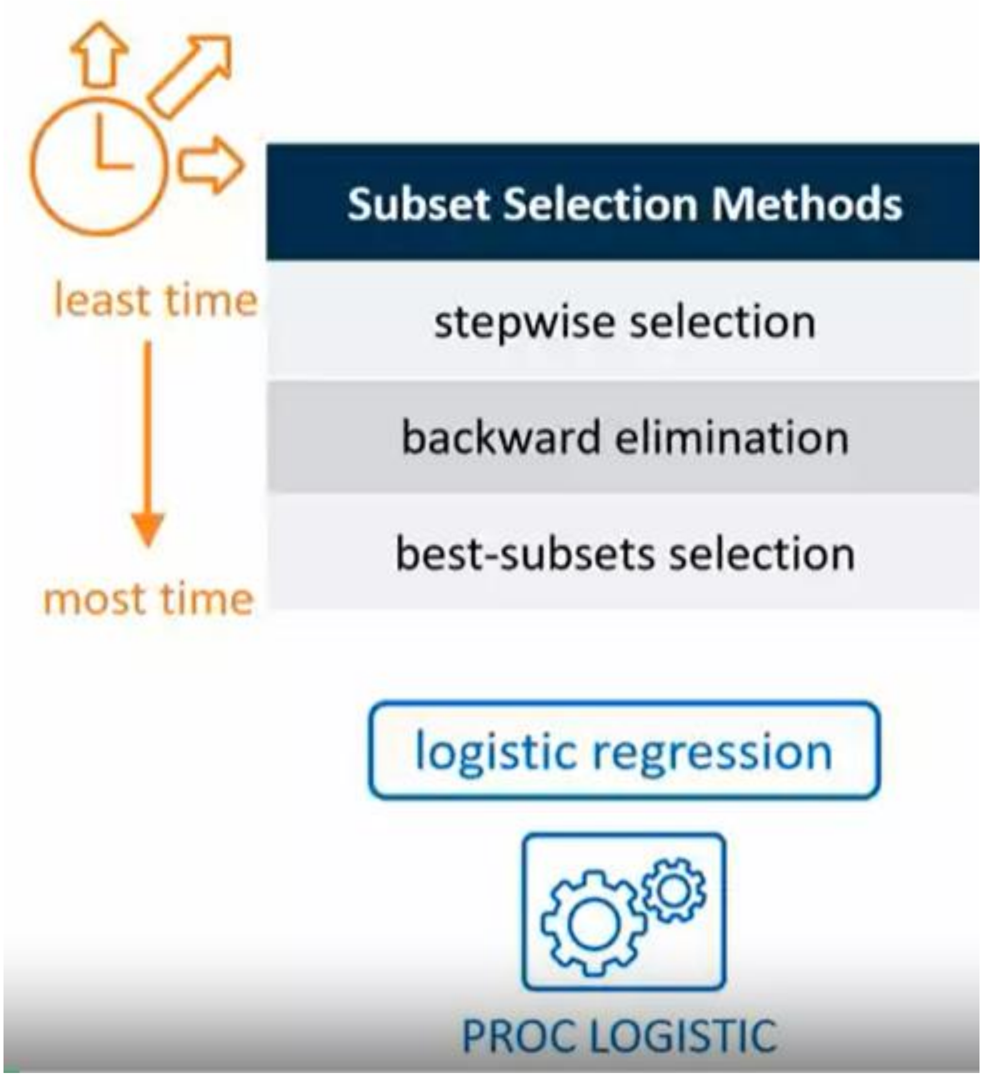
- exclude important inputs
- include spurious inputs

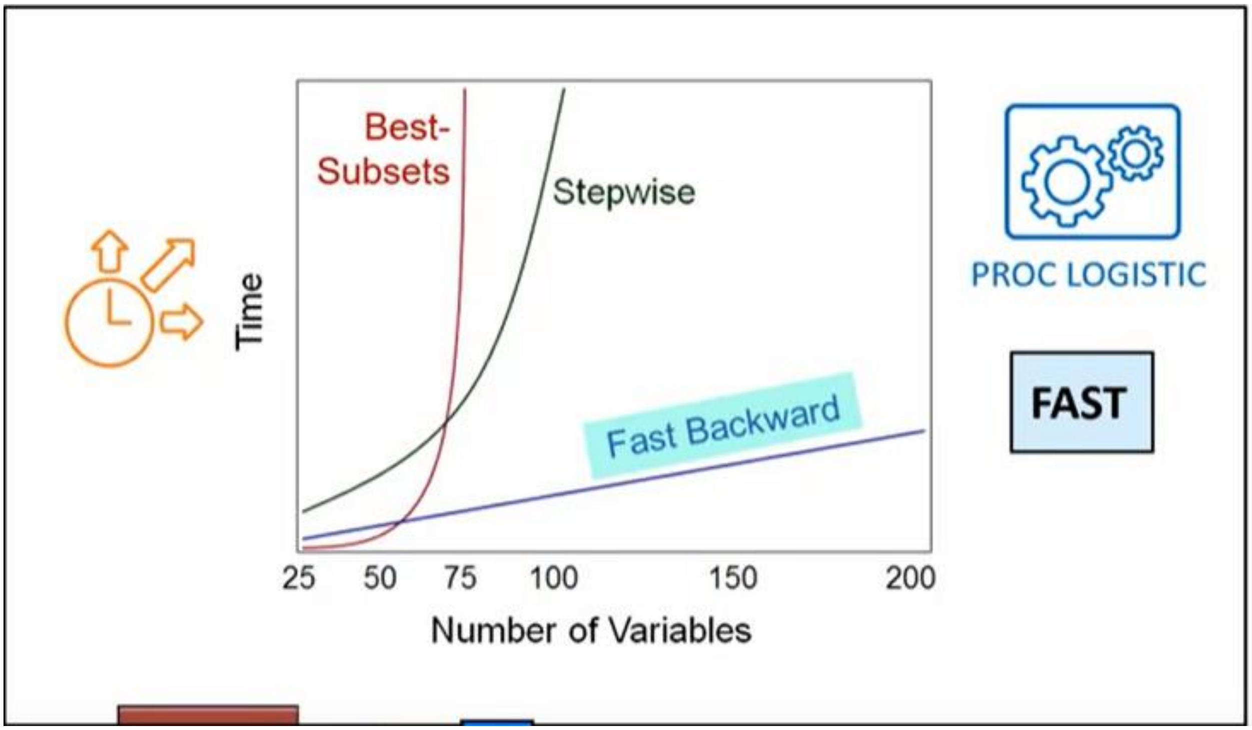
quasi-complete separation

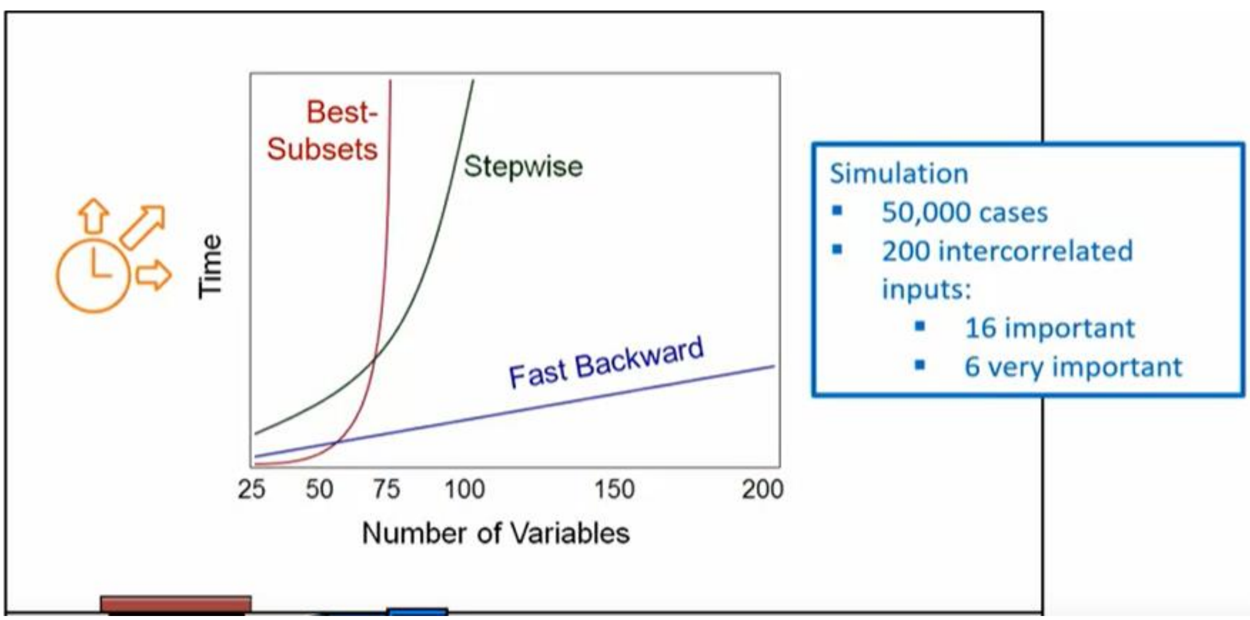
inputs cannot be re-added

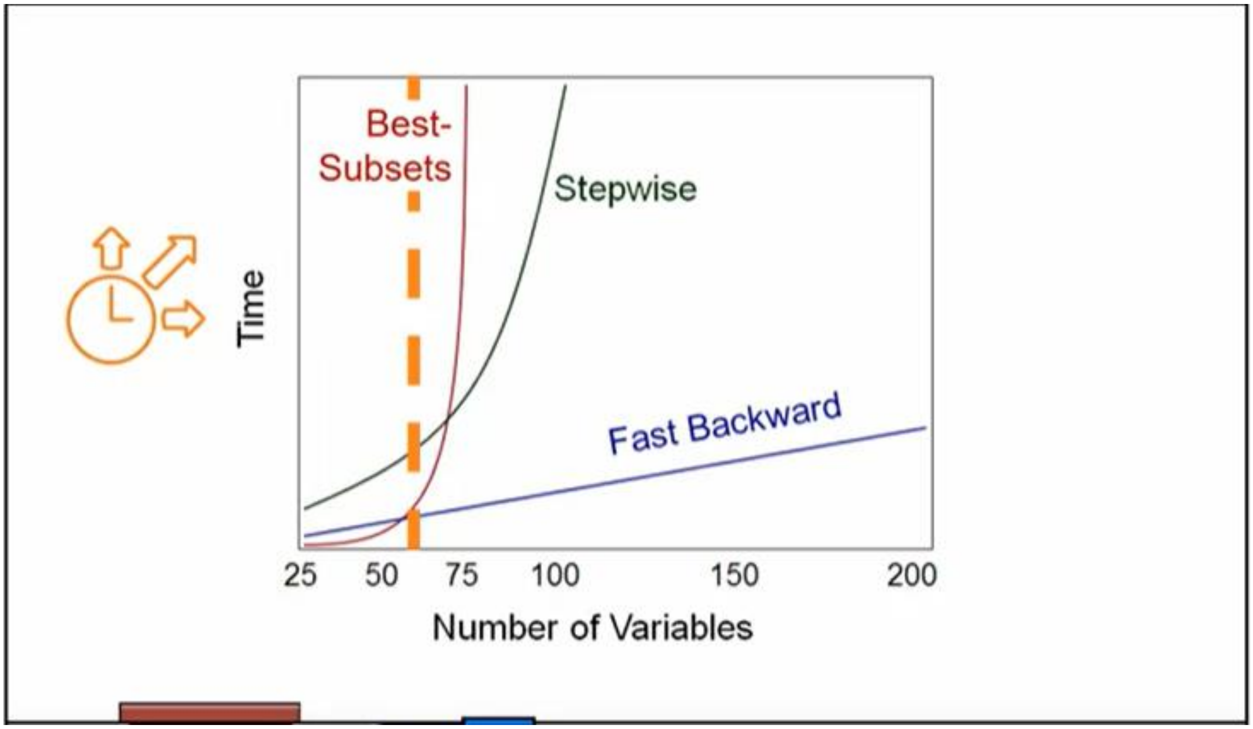
multicollinearity

Scalability of the Subset Selection Methods in PROC LOGISTIC

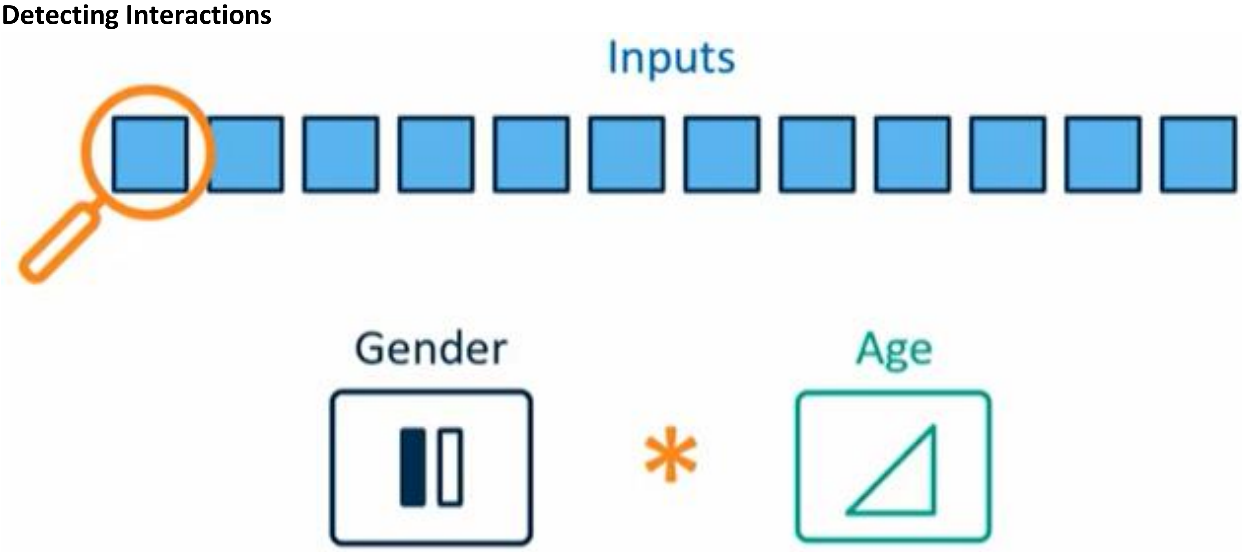


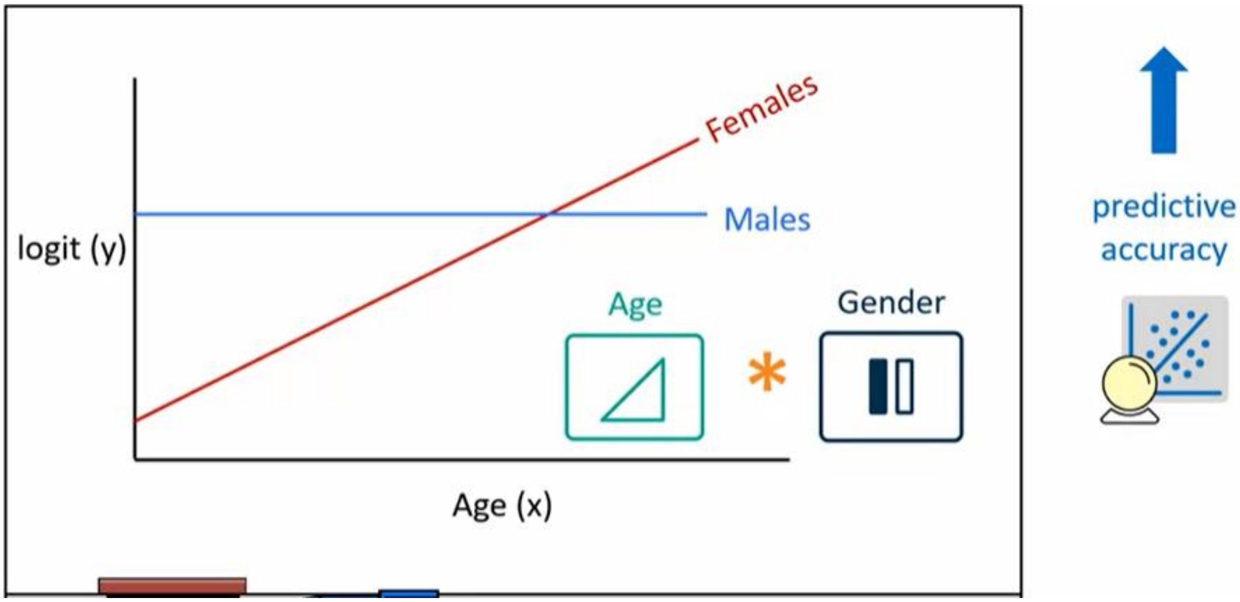

Best-
Subsets
Stepwise
Time
Fast Backward
25
50
75
100
150
200
Number of Variables
PROC LOGISTIC
FAST

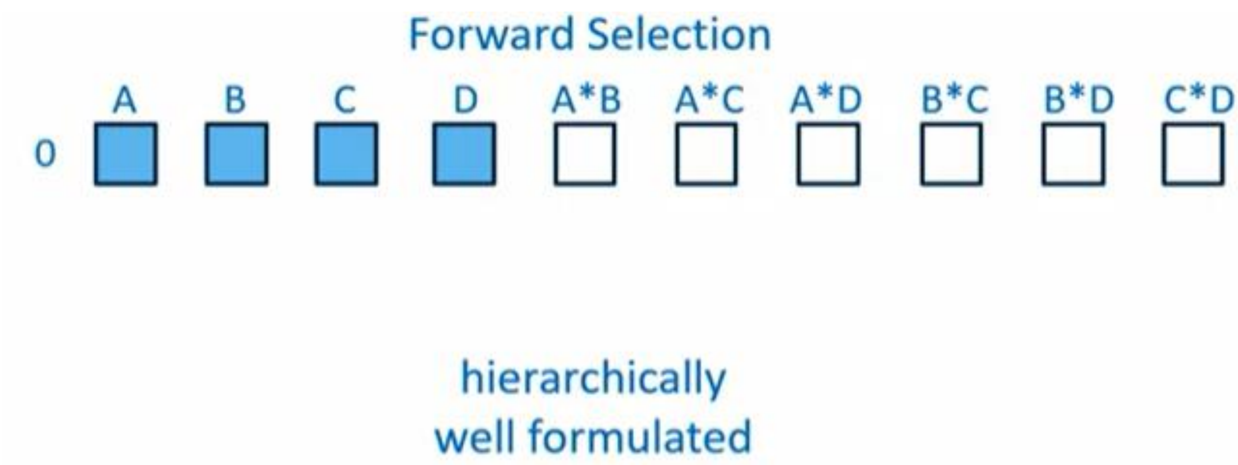

Best-
Subsets
Stepwise
Time
Fast Backward
25
50
75
100
150
200
Number of Variables
Simulation
50,000 cases
200 intercorrelated inputs:
16 important
6 very important

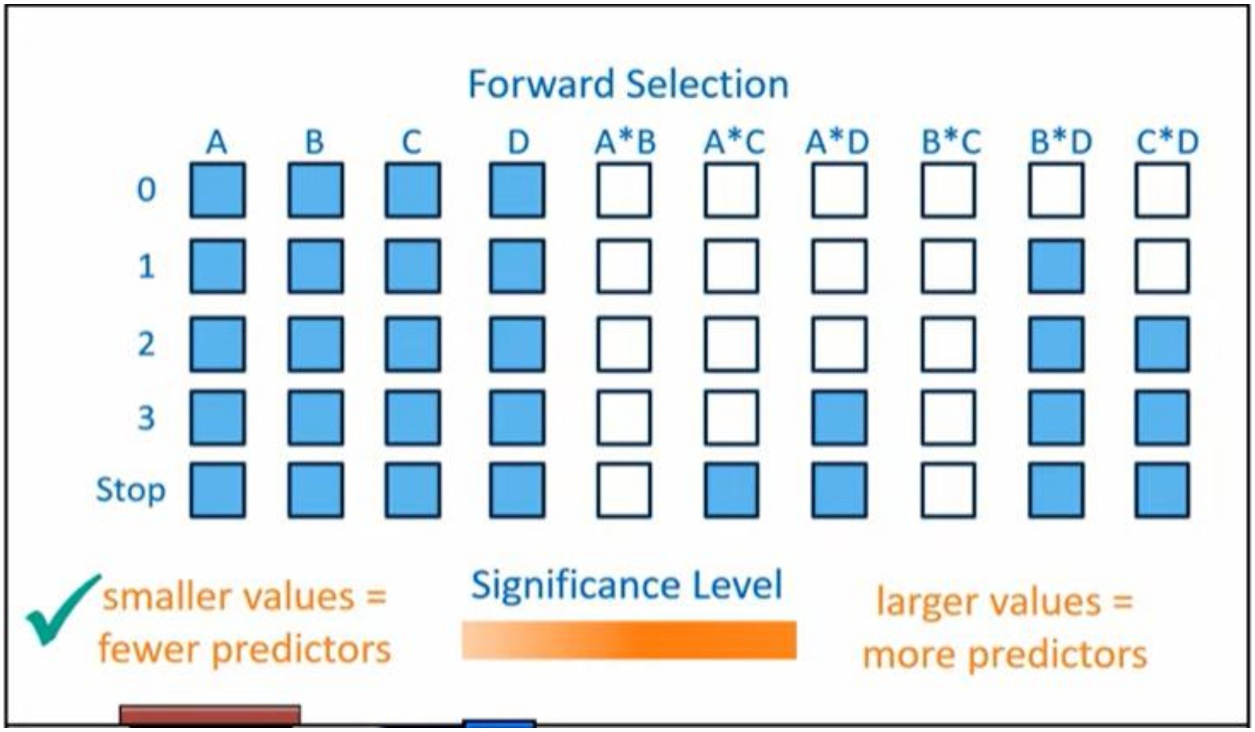


Detecting Interactions

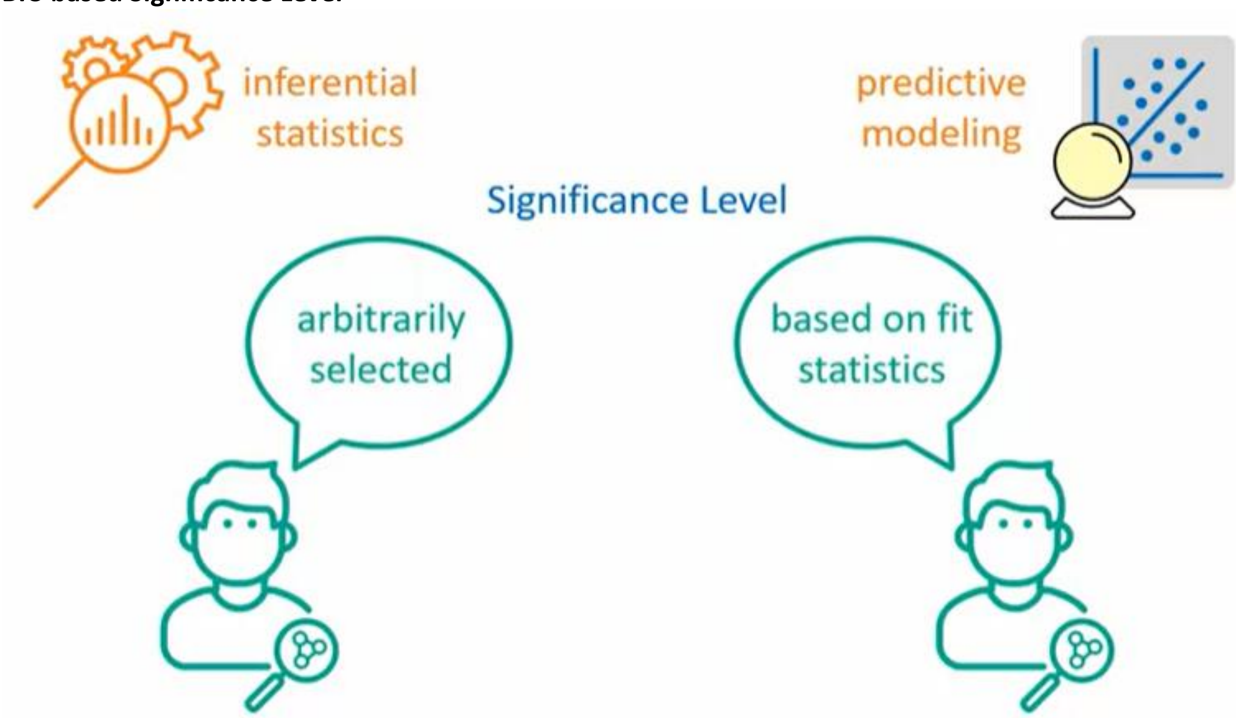


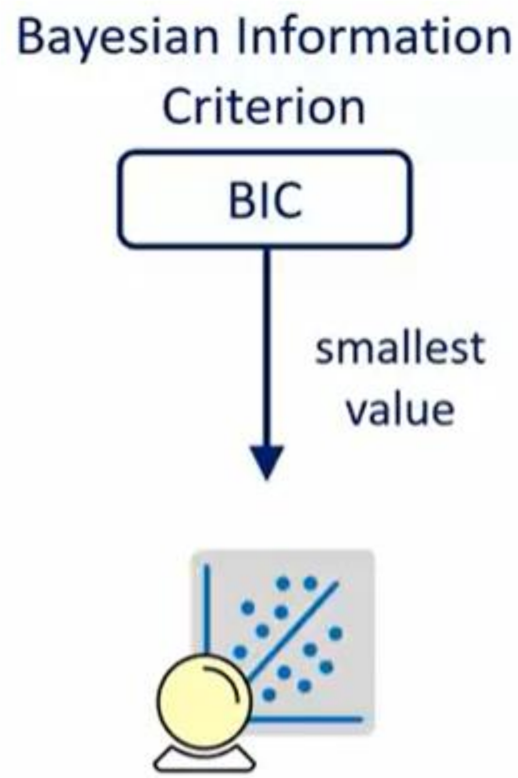

Females
Males
logit (y)
Age
Gender
*
Age (x)
predictive
accuracy


Forward Selection
A
B
C
D
A*B
A*C
A*D
B*C
B*D
C*D
0
hierarchically
well formulated



BIC-based Significance Level





$$BIC = -2\ln(likelihood) + (\#\ parameters) * \ln(n)$$

number of observations

$$BIC_{p+1} < BIC_p$$

$$-2\ln(L_{p+1}) + (p+1) * \ln(n) < -2\ln(L_p) + p * \ln(n)$$

$$\ln(n) < 2(\ln(L_{p+1}) - \ln(L_p))$$

$$"p - value" < 1 - F_{\chi^2}(\ln(n))$$

$$BIC = -2\ln(likelihood) + (\#\,parameters) * \ln(n)$$

$$BIC_{p+1} < BIC_p$$

number of parameters

$$-2\ln(L_{p+1}) + (p+1) * \ln(n) < -2\ln(L_p) + p * \ln(n)$$

$$\ln(n) < 2(\ln(L_{p+1}) - \ln(L_p))$$

$$"\,p - value\," < 1 - F_{\chi^2}(\ln(n))$$

$$BIC = -2\ln(likelihood) + (\#\,parameters) * \ln(n)$$

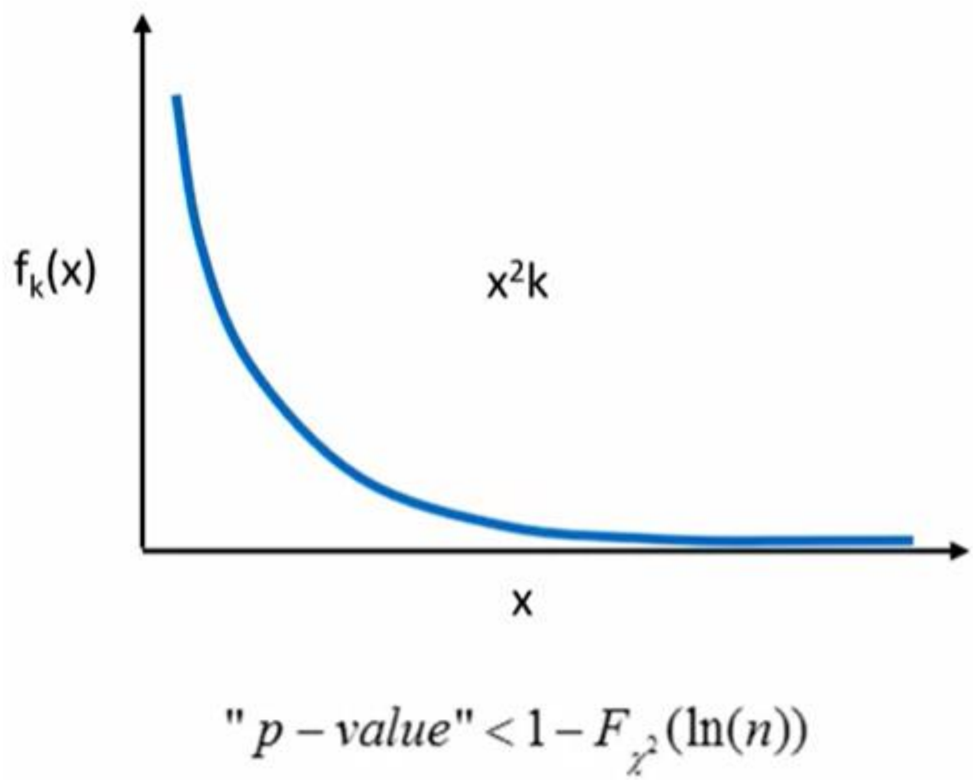
$$BIC_{p+1} < BIC_p$$

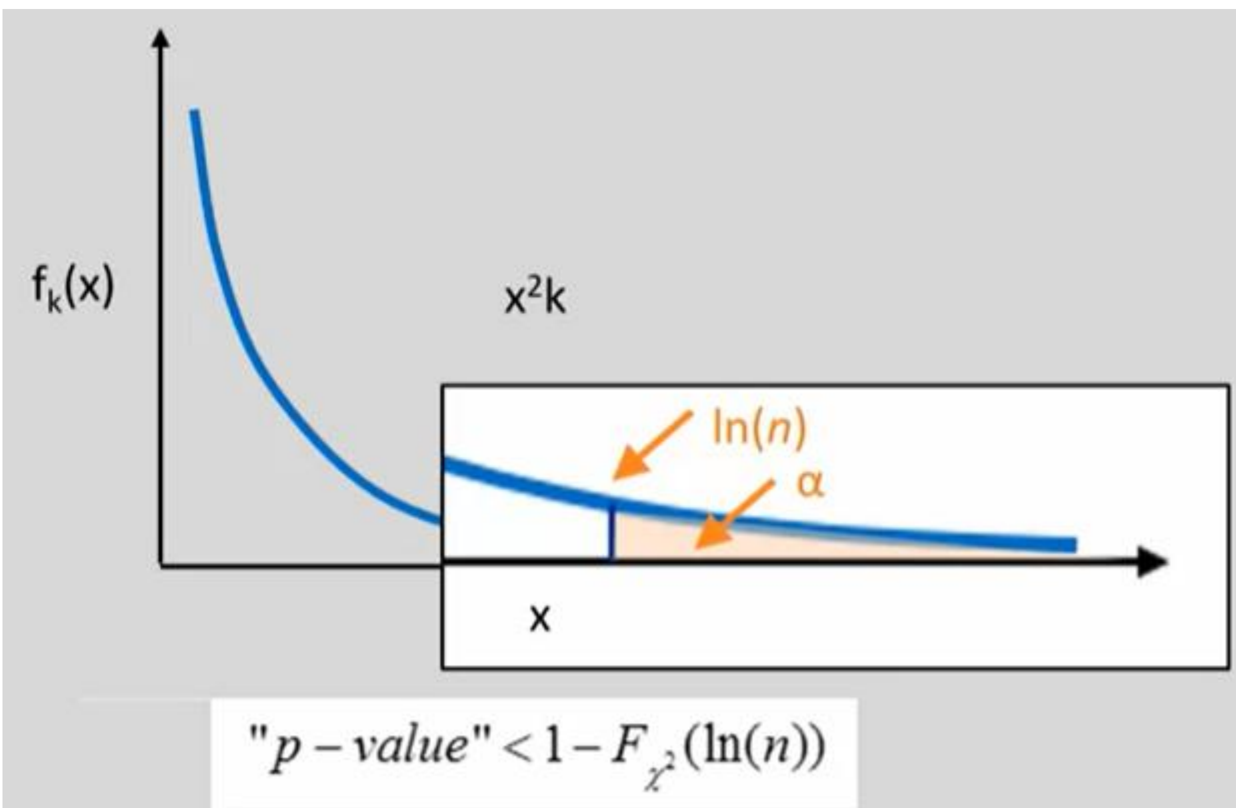
$$-2\ln(L_{p+1}) + (p+1) * \ln(n) < -2\ln(L_p) + p * \ln(n)$$

$$\ln(n) < 2(\ln(L_{p+1}) - \ln(L_p))$$

likelihood ratio test

$$"\,p - value\," < 1 - F_{\chi^2}(\ln(n))$$


$f_k(x)$
x^2k
x
$"p-value" < 1 - F_{\chi^2}(\ln(n))$


$f_k(x)$
x^2k
$\ln(n)$
α
x
$"p-value" < 1 - F_{\chi^2}(\ln(n))$

Demo Detecting Interactions

* Compute a BIC-based significance level using the sample size for *n*.

* Use the forward selection method to detect important 2-factor interactions among the screened variables and **Res**.

pmlr03d08.sas

```
title1 "P-Value for Entry and Retention";

%global sl;
proc sql;
   select 1-probchi(log(sum(ins ge 0)),1) into :sl
   from work.train_imputed_swoe_bins;
quit;
```

P-Value for Entry and Retention

0.001586

```
title1 "Interaction Detection using Forward Selection";
proc logistic data=work.train_imputed_swoe_bins;
   class res (param=ref ref='S');
   model ins(event='1')= &screened res
               SavBal|Dep|DDA|CD|Sav|CC|ATM|MM|branch_swoe|Phone|IRA|
               IRABal|B_DDABal|ATMAmt|ILS|POS|NSF|CCPurc|SDB|DepAmt|
               CCBal|Inv|InArea|Age|CashBk|MICRScor|Income|res @2 / incl
         selection=forward slentry=&sl;
run;
```

```
include=28 clodds=pl
```

Number of Observations Read	21512
Number of Observations Used	21512

Response Profile		
Ordered Value	Ins	Total Frequency
1	0	14061
2	1	7451

Probability modeled is Ins=1.

Forward Selection Procedure

Class Level Information			
Class	Value	Design Variables	
Res	R	1	0
	S	0	0
	U	0	1

Intercept SavBal Dep DDA CD Sav CC ATM MM branch_swoe Phone IRA IRABal B_DDABal ATMAmt ILS POS NSF CCPurc SDB DepAmt CCBal Inv InArea Age CashBk MICRScor Income Res

Step 1. Effect SavBal*B_DDABal entered:

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
1030.7985	383	<.0001

Step 2. Effect SavBal*DDA entered:

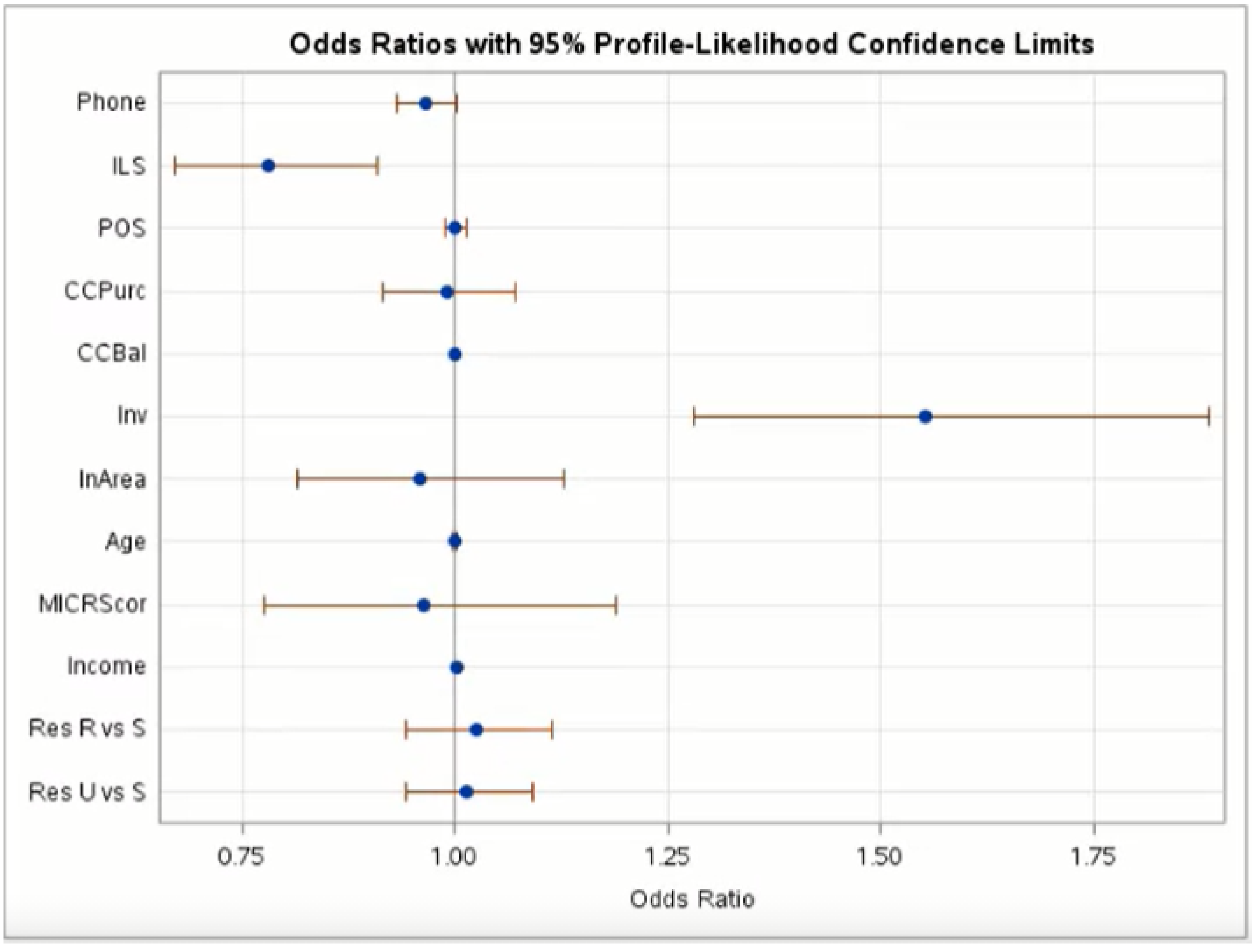
Step 3. Effect MM*B_DDABal entered:

Step	Effect Entered	DF	Number In	Score Chi-Square	Pr > ChiSq	Variable Label
Summary of Forward Selection						
1	SavBal*B_DDABal	1	29	342.3565	<.0001	
2	SavBal*DDA	1	30	75.6237	<.0001	
3	MM*B_DDABal	1	31	61.2113	<.0001	
4	branch_swoe*ATMAmt	1	32	55.0544	<.0001	
5	Sav*B_DDABal	1	33	46.3236	<.0001	
6	ATMAmt*DepAmt	1	34	36.9443	<.0001	
7	SavBal*SDB	1	35	28.9771	<.0001	
8	SavBal*ATMAmt	1	36	24.4441	<.0001	
9	B_DDABal*ATMAmt	1	37	28.2743	<.0001	
10	SavBal*IRA	1	38	18.1867	<.0001	
11	SavBal*MM	1	39	18.2860	<.0001	
12	SavBal*CC	1	40	17.6232	<.0001	
13	Sav*NSF	1	41	14.3527	0.0002	
14	DDA*ATMAmt	1	42	14.8869	0.0001	
15	Dep*ATM	1	43	14.5252	0.0001	
16	IRA*B_DDABal	1	44	13.9868	0.0002	
17	CD*MM	1	45	12.9265	0.0003	
18	MM*IRABal	1	46	11.9803	0.0005	
19	CD*Sav	1	47	10.8910	0.0010	
20	B_DDABal*CashBk	1	48	10.3711	0.0013	
21	Sav*CC	1	49	10.1994	0.0014	

Type 3 Analysis of Effects			
Effect	DF	Wald Chi-Square	Pr > ChiSq
SavBal	1	299.0089	<.0001
Dep	1	3.2378	0.0720
DDA	1	7.9296	0.0049
CD	1	219.7287	<.0001
Sav	1	121.8246	<.0001
CC	1	6.1556	0.0131
ATM	1	0.2451	0.6205
MM	1	142.9057	<.0001
branch_swoe	1	129.4792	<.0001
Phone	1	3.3792	0.0660
IRA	1	29.0627	<.0001
IRABal	1	6.3858	0.0115

Analysis of Maximum Likelihood Estimates						
Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept		1	-2.0116	0.1406	204.7488	<.0001
SavBal		1	0.000167	9.646E-6	299.0089	<.0001
Dep		1	-0.0457	0.0254	3.2378	0.0720
DDA		1	-0.1794	0.0637	7.9296	0.0049
CD		1	1.1361	0.0766	219.7287	<.0001
Sav		1	1.0100	0.0915	121.8246	<.0001
CC		1	0.1248	0.0503	6.1556	0.0131
ATM		1	0.0314	0.0633	0.2451	0.6205
MM		1	1.7606	0.1473	142.9057	<.0001

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	78.9	Somers' D	0.578
Percent Discordant	21.1	Gamma	0.578
Percent Tied	0.0	Tau-a	0.262
Pairs	104768511	c	0.789



/* Run this code before demo l3d8 */

/* === */

/* Lesson 1, Section 1: l1d1.sas

Demonstration: Examining the Code for Generating

Descriptive Statistics and Frequency Tables */

/* === */

```
data work.develop;
   set pmlr.develop;
run;

%global inputs;
```

```
%let inputs=ACCTAGE DDA DDABAL DEP DEPAMT CASHBK
        CHECKS DIRDEP NSF NSFAMT PHONE TELLER
        SAV SAVBAL ATM ATMAMT POS POSAMT CD
        CDBAL IRA IRABAL LOC LOCBAL INV
        INVBAL ILS ILSBAL MM MMBAL MMCRED MTG
        MTGBAL CC CCBAL CCPURC SDB INCOME
        HMOWN LORES HMVAL AGE CRSCORE MOVED
        INAREA;

proc means data=work.develop n nmiss mean min max;
  var &inputs;
run;

proc freq data=work.develop;
  tables ins branch res;
run;

/* ===================================================== */
/* Lesson 1, Section 2: l1d2.sas
  Demonstration: Splitting the Data   */
/* ===================================================== */


/* Sort the data by the target in preparation for stratified sampling. */

proc sort data=work.develop out=work.develop_sort;
  by ins;
run;
```

```
/* The SURVEYSELECT procedure will perform stratified sampling
  on any variable in the STRATA statement. The OUTALL option
  specifies that you want a flag appended to the file to
  indicate selected records, not simply a file comprised
  of the selected records. */

proc surveyselect noprint data=work.develop_sort
              samprate=.6667 stratumseed=restore
              out=work.develop_sample
              seed=44444 outall;
  strata ins;
run;

/* Verify stratification. */

proc freq data=work.develop_sample;
  tables ins*selected;
run;

/* Create training and validation data sets. */

data work.train(drop=selected SelectionProb SamplingWeight)
    work.valid(drop=selected SelectionProb SamplingWeight);
  set work.develop_sample;
  if selected then output work.train;
  else output work.valid;
run;
```

```
/* ====================================================== */
/* Lesson 2, Section 1: l2d1.sas
   Demonstration: Fitting a Basic Logistic
   Regression Model, Parts 1 and 2                    */
/* ====================================================== */

title1 "Logistic Regression Model for the Variable Annuity Data Set";
proc logistic data=work.train
          plots(only maxpoints=none)=(effect(clband x=(ddabal depamt checks res))
          oddsratio (type=horizontalstat));
   class res (param=ref ref='S') dda (param=ref ref='0');
   model ins(event='1')=dda ddabal dep depamt
           cashbk checks res / stb clodds=pl;
   units ddabal=1000 depamt=1000 / default=1;
   oddsratio 'Comparisons of Residential Classification' res / diff=all cl=pl;
   effectplot slicefit(sliceby=dda x=ddabal) / noobs;
   effectplot slicefit(sliceby=dda x=depamt) / noobs;
run;
title1;



/* ====================================================== */
/* Lesson 2, Section 1: l2d2.sas
   Demonstration: Scoring New Cases                   */
/* ====================================================== */

/* Score a new data set with one run of the LOGISTIC procedure with the
   SCORE statement.  */
```

```
proc logistic data=work.train noprint;
  class res (param=ref ref='S');
  model ins(event='1')= res dda ddabal dep depamt cashbk checks;
  score data = pmlr.new out=work.scored1;
run;

title1 "Predicted Probabilities from Scored Data Set";
proc print data=work.scored1(obs=10);
  var p_1 dda ddabal dep depamt cashbk checks res;
run;

title1 "Mean of Predicted Probabilities from Scored Data Set";
proc means data=work.scored1 mean nolabels;
  var p_1;
run;

/* Score a new data set with the OUTMODEL= amd INMODEL= options */

proc logistic data=work.train outmodel=work.scoredata noprint;
  class res (param=ref ref='S');
  model ins(event='1')= res dda ddabal dep depamt cashbk checks;
run;

proc logistic inmodel=work.scoredata noprint;
  score data = pmlr.new out=work.scored2;
run;

title1 "Predicted Probabilities from Scored Data Set";
```

```
proc print data=work.scored2(obs=10);
  var p_1 dda ddabal dep depamt cashbk checks res;
run;

/* Score a new data set with the CODE Statement  */

proc logistic data=work.train noprint;
  class res (param=ref ref='S');
  model ins(event='1')= res dda ddabal dep depamt cashbk checks;
  code file="&PMLRfolder/pmlr_score.txt";
run;

data work.scored3;
  set pmlr.new;
  %include "&PMLRfolder/pmlr_score.txt";
run;

title1 "Predicted Probabilities from Scored Data Set";
proc print data=work.scored3(obs=10);
  var p_ins1 dda ddabal dep depamt cashbk checks res;
run;
title1 ;



/* ===================================================== */
/* Lesson 2, Section 2: l2d3.sas
  Demonstration: Correcting for Oversampling           */
/* ===================================================== */
```

```
/* Specify the prior probability to correct for oversampling.  */
%global pi1;
%let pi1=.02;

/* Correct predicted probabilities */

proc logistic data=work.train noprint;
  class res (param=ref ref='S');
  model ins(event='1')=dda ddabal dep depamt cashbk checks res;
  score data=pmlr.new out=work.scored4 priorevent=&pi1;
run;

title1 "Adjusted Predicted Probabilities from Scored Data Set";
proc print data=work.scored4(obs=10);
  var p_1 dda ddabal dep depamt cashbk checks res;
run;

title1 "Mean of Adjusted Predicted Probabilities from Scored Data Set";
proc means data=work.scored4 mean nolabels;
  var p_1;
run;
title1 ;

/* Correct probabilities in the Score Code */

proc logistic data=work.train noprint;
  class res (param=ref ref='S');
  model ins(event='1')=dda ddabal dep depamt cashbk checks res;
  /* File suffix "txt" is used so you can view the file */
```

```
  /* with a native text editor. SAS prefers "sas", but  */
  /* when specified as a filename, SAS does not care.   */
  code file="&PMLRfolder/pmlr_score_adj.txt";
run;

%global rho1;
proc SQL noprint;
  select mean(INS) into :rho1
  from work.train;
quit;

data new;
  set pmlr.new;
  off=log(((1-&pi1)*&rho1)/(&pi1*(1-&rho1)));
run;

data work.scored5;
  set work.new;
  %include "&PMLRfolder/pmlr_score_adj.txt";
  eta=log(p_ins1/p_ins0) - off;
  prob=1/(1+exp(-eta));
run;

title1 "Adjusted Predicted Probabilities from Scored Data Set";
proc print data=scored5(obs=10);
  var prob dda ddabal dep depamt cashbk checks res;
run;
title1 ;
```

```
/* ===================================================== */
/* Lesson 3, Section 1: l3d1.sas
   Demonstration: Imputing Missing Values
/* ===================================================== */

title1 "Variables with Missing Values";
proc print data=work.train(obs=15);
  var ccbal ccpurc income hmown;
run;
title1 ;

/* Create missing indicators */
data work.train_mi(drop=i);
  set work.train;
  /* name the missing indicator variables */
  array mi{*} MIAcctAg MIPhone MIPOS MIPOSAmt
          MIInv MIInvBal MICC MICCBal
          MICCPurc MIIncome MIHMOwn MILORes
          MIHMVal MIAge MICRScor;
  /* select variables with missing values */
  array x{*} acctage phone pos posamt
         inv invbal cc ccbal
         ccpurc income hmown lores
         hmval age crscore;
  do i=1 to dim(mi);
    mi{i}=(x{i}=.);
    nummiss+mi{i};
  end;
```

```
run;

/* Impute missing values with the median */
proc stdize data=work.train_mi reponly method=median out=work.train_imputed;
  var &inputs;
run;

title1 "Imputed Values with Missing Indicators";
proc print data=work.train_imputed(obs=12);
  var ccbal miccbal ccpurc miccpurc income miincome hmown mihmown nummiss;
run;
title1;



/* ===================================================== */
/* Lesson 3, Section 2: l3d2a.sas
  Demonstration: Collapsing the Levels of a
  Nominal Input, Part 1                          */
/* ===================================================== */

proc means data=work.train_imputed noprint nway;
  class branch;
  var ins;
  output out=work.level mean=prop;
run;

title1 "Proportion of Events by Level";
proc print data=work.level;
```

```
run;

/* Use ODS to output the ClusterHistory output object into a data set
  named "cluster." */

ods output clusterhistory=work.cluster;

proc cluster data=work.level method=ward outtree=work.fortree
     plots=(dendrogram(vertical height=rsq));
  freq _freq_;
  var prop;
  id branch;
run;




/* ===================================================== */
/* Lesson 3, Section 2: l3d2b.sas
  Demonstration: Collapsing the Levels of a
  Nominal Input, Part 2                    */
/* ===================================================== */

/* Use the FREQ procedure to get the Pearson Chi^2 statistic of the
  full BRANCH*INS table. */

proc freq data=work.train_imputed noprint;
  tables branch*ins / chisq;
  output out=work.chi(keep=_pchi_) chisq;
run;
```

```
/* Use a one-to-many merge to put the Chi^2 statistic onto the clustering
  results. Calculate a (log) p-value for each level of clustering. */

data work.cutoff;
  if _n_=1 then set work.chi;
  set work.cluster;
  chisquare=_pchi_*rsquared;
  degfree=numberofclusters-1;
  logpvalue=logsdf('CHISQ',chisquare,degfree);
run;

/* Plot the log p-values against number of clusters. */

title1 "Plot of the Log of the P-Value by Number of Clusters";
proc sgplot data=work.cutoff;
  scatter y=logpvalue x=numberofclusters
        / markerattrs=(color=blue symbol=circlefilled);
  xaxis label="Number of Clusters";
  yaxis label="Log of P-Value" min=-120 max=-85;
run;
title1 ;

/* Create a macro variable (&ncl) that contains the number of clusters
  associated with the minimum log p-value. */

proc sql;
  select NumberOfClusters into :ncl
  from work.cutoff
```

```
  having logpvalue=min(logpvalue);
quit;

proc tree data=work.fortree nclusters=&ncl out=work.clus noprint;
  id branch;
run;

proc sort data=work.clus;
  by clusname;
run;

title1 "Levels of Branch by Cluster";
proc print data=work.clus;
  by clusname;
  id clusname;
run;
title1 ;

/* The DATA Step creates the scoring code to assign the branches to a cluster. */

filename brclus "&PMLRfolder/branch_clus.sas";

data _null_;
  file brclus;
  set work.clus end=last;
  if _n_=1 then put "select (branch);";
  put "  when ('" branch +(-1) "') branch_clus = '" cluster +(-1) "';";
  if last then do;
    put "  otherwise branch_clus = 'U';" / "end;";
```

```
  end;
run;

data work.train_imputed_greenacre;
  set work.train_imputed;
  %include brclus / source2;
run;



/* ========================================================== */
/* Lesson 3, Section 2: l3d3.sas
  Demonstration: Computing the Smoothed Weight of Evidence  */
/* ========================================================== */

/* Rho1 is the proportion of events in the training data set. */
%global rho1;
proc sql noprint;
  select mean(ins) into :rho1
  from work.train_imputed;
run;

/* The output data set from PROC MEANS will have the number of
  observations and events for each level of branch. */

proc means data=work.train_imputed sum nway noprint;
  class branch;
  var ins;
  output out=work.counts sum=events;
run;
```

```
/* The DATA Step creates the scoring code that assigns each branch to
   a value of the smoothed weight of evidence. */


filename brswoe "&PMLRfolder/swoe_branch.sas";

data _null_;
   file brswoe;
   set work.counts end=last;
   logit=log((events + &rho1*24)/(_FREQ_ - events + (1-&rho1)*24));
   if _n_=1 then put "select (branch);" ;
   put "  when ('" branch +(-1) "') branch_swoe = " logit ";" ;
   if last then do;
   logit=log(&rho1/(1-&rho1));
   put "  otherwise branch_swoe = " logit ";" / "end;";
   end;
run;

data work.train_imputed_swoe;
   set work.train_imputed;
   %include brswoe / source2;
run;



/* =========================================================== */
/* Lesson 3, Section 3: l3d4.sas
   Demonstration: Reducing Redundancy by Clustering Variables */
```

```
/* ========================================================= */

/* Use the ODS OUTPUT statement to generate data sets based on the variable
  clustering results and the clustering summary. */

ods select none;
ods output clusterquality=work.summary
        rsquare=work.clusters;

proc varclus data=work.train_imputed_swoe maxeigen=.7 hi;
  var &inputs branch_swoe miacctag
      miphone mipos miposamt miinv
      miinvbal micc miccbal miccpurc
      miincome mihmown milores mihmval
      miage micrscor;
run;
ods select all;

/* Use the CALL SYMPUT function to create a macro variable:&NVAR =
  the number of of clusters. This is also the number of variables
  in the analysis, going forward. */

%global nvar;
data _null_;
  set work.summary;
  call symput('nvar',compress(NumberOfClusters));
run;

title1 "Variables by Cluster";
```

```
proc print data=work.clusters noobs label split='*';
  where NumberOfClusters=&nvar;
  var Cluster Variable RSquareRatio VariableLabel;
  label RSquareRatio="1 - RSquare*Ratio";
run;
title1 ;

title1 "Variation Explained by Clusters";
proc print data=work.summary label;
run;

/* Choose a representative from each cluster.  */
%global reduced;
%let reduced=branch_swoe MIINCOME Dep CCBal MM Income ILS POS NSF CD
        DDA LOC Age Inv InArea AcctAge Moved CRScore MICRScor
        IRABal MIAcctAg SavBal CashBk DDABal SDB InvBal CCPurc
        ATMAmt Sav CC Phone HMOwn DepAmt IRA MTG ATM LORes;



/* ===================================================== */
/* Lesson 3, Section 4:  l3d5a.sas
  Demonstration: Performing Variable Screening, Part 1  */
/* ===================================================== */

ods select none;
ods output spearmancorr=work.spearman
        hoeffdingcorr=work.hoeffding;
```

```
proc corr data=work.train_imputed_swoe spearman hoeffding;
  var ins;
  with &reduced;
run;

ods select all;

proc sort data=work.spearman;
   by variable;
run;

proc sort data=work.hoeffding;
   by variable;
run;

data work.correlations;
  merge work.spearman(rename=(ins=scorr pins=spvalue))
        work.hoeffding(rename=(ins=hcorr pins=hpvalue));
  by variable;
  scorr_abs=abs(scorr);
  hcorr_abs=abs(hcorr);
run;

proc rank data=work.correlations out=work.correlations1 descending;
   var scorr_abs hcorr_abs;
   ranks ranksp rankho;
run;

proc sort data=work.correlations1;
```

```
  by ranksp;
run;

title1 "Rank of Spearman Correlations and Hoeffding Correlations";
proc print data=work.correlations1 label split='*';
  var variable ranksp rankho scorr spvalue hcorr hpvalue;
  label ranksp ='Spearman rank*of variables'
        scorr  ='Spearman Correlation'
        spvalue='Spearman p-value'
        rankho ='Hoeffding rank*of variables'
        hcorr  ='Hoeffding Correlation'
        hpvalue='Hoeffding p-value';
run;



/* ===================================================== */
/* Lesson 3, Section 4: l3d5b.sas
  Demonstration: Performing Variable Screening, Part 2  */
/* ===================================================== */

/* Find values for reference lines */
%global vref href;
proc sql noprint;
  select min(ranksp) into :vref
  from (select ranksp
  from work.correlations1
  having spvalue > .5);
```

```
  select min(rankho) into :href
  from (select rankho
  from work.correlations1
  having hpvalue > .5);
quit;

/* Plot variable names, Hoeffding ranks, and Spearman ranks. */

title1 "Scatter Plot of the Ranks of Spearman vs. Hoeffding";
proc sgplot data=work.correlations1;
  refline &vref / axis=y;
  refline &href / axis=x;
  scatter y=ranksp x=rankho / datalabel=variable;
  yaxis label="Rank of Spearman";
  xaxis label="Rank of Hoeffding";
run;
title1 ;

%global screened;
%let screened=SavBal Dep DDA CD Sav CC ATM MM branch_swoe Phone IRA IRABal
         DDABal ATMAmt ILS POS NSF CCPurc SDB DepAmt CCBal Inv InArea
         Age CashBk MICRScor Income;


/* ======================================================= */
/* Lesson 3, Section 4: l3d6.sas
  Demonstration: Creating Empirical Logit Plots        */
/* ======================================================= */
```

```
%global var;
%let var=DDABal;

/* Group the data by the variable of interest in order to create
   empirical logit plots.   */

proc rank data=work.train_imputed_swoe groups=100 out=work.ranks;
   var &var;
   ranks bin;
run;

title1 "Checking Account Balance by Bin";
proc print data=work.ranks(obs=10);
   var &var bin;
run;

/* The data set BINS will contain:INS=the count of successes in each bin,
   _FREQ_=the count of trials in each bin, DDABAL=the avg DDABAL in each bin. */

proc means data=work.ranks noprint nway;
   class bin;
   var ins &var;
   output out=work.bins sum(ins)=ins mean(&var)=&var;
run;

title1 "Number of Observations, Events, and Average Checking Account Balance by Bin";
proc print data=work.bins(obs=10);
run;
```

```
/* Calculate the empirical logit */

data work.bins;
  set work.bins;
  elogit=log((ins+(sqrt(_FREQ_ )/2))/
        ( _FREQ_ -ins+(sqrt(_FREQ_ )/2)));
run;

title1 "Empirical Logit against &var";
proc sgplot data=work.bins;
  reg y=elogit x=&var /
     curvelabel="Linear Relationship?"
     curvelabelloc=outside
     lineattrs=(color=ligr);
  series y=elogit x=&var;
run;

title1 "Empirical Logit against Binned &var";
proc sgplot data=work.bins;
  reg y=elogit x=bin /
     curvelabel="Linear Relationship?"
     curvelabelloc=outside
     lineattrs=(color=ligr);
  series y=elogit x=bin;
run;

/* ====================================================== */
```

```
/* Lesson 3, Section 4: l3d7a.sas
   Demonstration: Accommodating a Nonlinear Relationship,
   Part 1                              */
/* ===================================================== */

title1 "Checking Account Balance and INS by Checking Account";
proc means data=work.train_imputed_swoe mean median min max;
   class dda;
   var ddabal ins;
run;

/* A possible remedy for that non-linearity is to replace the logical
   imputation of 0 for non-DDA customers with the mean. */

%global mean;
proc sql noprint;
   select mean(ddabal) into :mean
   from work.train_imputed_swoe where dda;
quit;

data work.train_imputed_swoe_dda;
   set work.train_imputed_swoe;
   if not dda then ddabal=&mean;
run;

/* Create new logit plots */
%global var;
%let var=DDABal;
```

```
proc rank data=work.train_imputed_swoe_dda groups=100 out=work.ranks;
  var &var;
  ranks bin;
run;

proc means data=work.ranks noprint nway;
  class bin;
  var ins &var;
  output out=work.bins sum(ins)=ins mean(&var)=&var;
run;

/* Calculate the empirical logit */
data work.bins;
  set work.bins;
  elogit=log((ins+(sqrt(_FREQ_ )/2))/
        ( _FREQ_ -ins+(sqrt(_FREQ_ )/2)));
run;

title1 "Empirical Logit against &var";
proc sgplot data=work.bins;
  reg y=elogit x=&var /
     curvelabel="Linear Relationship?"
    curvelabelloc=outside
    lineattrs=(color=ligr);
  series y=elogit x=&var;
run;

title1 "Empirical Logit against Binned &var";
proc sgplot data=work.bins;
```

```
  reg y=elogit x=bin /
     curvelabel="Linear Relationship?"
     curvelabelloc=outside
     lineattrs=(color=ligr);
  series y=elogit x=bin;
run;



/* ====================================================== */
/* Lesson 3, Section 4: l3d7b.sas
   Demonstration: Accommodating a Nonlinear Relationship,
   Part 2                              */
/* ====================================================== */

/* Using the binned values of DDABal may make for a more linear
   relationship between the input and the target. The following code
   creates DATA step code to bin DDABal, yielding a new predictor, B_DDABal.  */

/* Rank the observations. */

proc rank data=work.train_imputed_swoe_dda groups=100 out=work.ranks;
  var ddabal;
  ranks bin;
run;

/* Save the endpoints of each bin */

proc means data=work.ranks noprint nway;
```

```
  class bin;
  var ddabal;
  output out=endpts max=max;
run;

title1 "Checking Account Balance Endpoints";
proc print data=work.endpts(obs=10);
run;

/* Write the code to assign individuals to bins according to the DDABal. */

filename rank "&PMLRfolder/rank.sas";

data _null_;
  file rank;
  set work.endpts end=last;
  if _n_=1 then put "select;";
  if not last then do;
    put "  when (ddabal <= " max ") B_DDABal =" bin ";";
  end;
  else if last then do;
    put "  otherwise B_DDABal =" bin ";" / "end;";
  end;
run;

/* Use the code. */

data work.train_imputed_swoe_bins;
  set work.train_imputed_swoe_dda;
```

```
  %include rank / source;
run;

title1 "Minimum and Maximum Checking Account Balance by Bin";
proc means data=work.train_imputed_swoe_bins min max;
  class B_DDABal;
  var DDABal;
run;
title1 ;

/* Switch the binned DDABal (B_DDABal) for the originally scaled
  DDABal input in the list of potential inputs. */
%global screened;
%let screened=SavBal Dep DDA CD Sav CC ATM MM branch_swoe Phone IRA
        IRABal B_DDABal ATMAmt ILS POS NSF CCPurc SDB DepAmt
        CCBal Inv InArea Age CashBk MICRScor Income;

/* ===================================================== */
/* Lesson 3, Section 5: l3d8a.sas
  Demonstration: Detecting Interactions
  [m643_5_m; derived from pmlr03d08.sas]          */
/* ===================================================== */

title1 "P-Value for Entry and Retention";

%global sl;
proc sql;
  select 1-probchi(log(sum(ins ge 0)),1) into :sl
  from work.train_imputed_swoe_bins;
```

```
quit;

title1 "Interaction Detection using Forward Selection";
proc logistic data=work.train_imputed_swoe_bins;
   class res (param=ref ref='S');
   model ins(event='1')= &screened res
            SavBal|Dep|DDA|CD|Sav|CC|ATM|MM|branch_swoe|Phone|IRA|
            IRABal|B_DDABal|ATMAmt|ILS|POS|NSF|CCPurc|SDB|DepAmt|
            CCBal|Inv|InArea|Age|CashBk|MICRScor|Income|res @2 / include=28 clodds=pl
      selection=forward slentry=&sl;
run;
```

```
/* Run this code before doing practice l3p7 */

/* ===================================================== */
/* Lesson 1, Practice 1
   Practice: Exploring the Veterans' Organization Data
   Used in the Practices                          */
/* ===================================================== */

data pmlr.pva(drop=control_number
              MONTHS_SINCE_LAST_PROM_RESP
              FILE_AVG_GIFT
              FILE_CARD_GIFT);
   set pmlr.pva_raw_data;
   STATUS_FL=RECENCY_STATUS_96NK in("F","L");
   STATUS_ES=RECENCY_STATUS_96NK in("E","S");
   home01=(HOME_OWNER="H");
   nses1=(SES="1");
   nses3=(SES="3");
   nses4=(SES="4");
   nses_=(SES="?");
   nurbr=(URBANICITY="R");
   nurbu=(URBANICITY="U");
   nurbs=(URBANICITY="S");
   nurbt=(URBANICITY="T");
   nurb_=(URBANICITY="?");
run;

proc contents data=pmlr.pva;
run;
```

```
proc means data=pmlr.pva mean nmiss max min;
  var _numeric_;
run;

proc freq data=pmlr.pva nlevels;
  tables _character_;
run;



/* ===================================================== */
/* Lesson 1, Practice 2
  Practice: Splitting the Data                    */
/* ===================================================== */

proc sort data=pmlr.pva out=work.pva_sort;
  by target_b;
run;

proc surveyselect noprint data=work.pva_sort
            samprate=0.5 out=pva_sample seed=27513
            outall stratumseed=restore;
  strata target_b;
run;

data pmlr.pva_train(drop=selected SelectionProb SamplingWeight)
    pmlr.pva_valid(drop=selected SelectionProb SamplingWeight);
  set work.pva_sample;
  if selected then output pmlr.pva_train;
```

```
  else output pmlr.pva_valid;
run;



/* ===================================================== */
/* Lesson 2, Practice 1
  Practice: Fitting a Logistic Regression Model        */
/* ===================================================== */

/* Modifications for your SAS software:
  ---------------------------------------------------
  (Optional) To avoid a warning in the log about the
  suppression of plots that have more than 5000
  observations, you can add the MAXPOINTS= option
  to the PROC LOGISTIC statement like this:
  plots(maxpoints=none only). Omitting the
  MAXPOINTS= option does not affect the results
  of the practices in this course.
*/

%global ex_pi1;
%let ex_pi1=0.05;

title1 "Logistic Regression Model of the Veterans' Organization Data";
proc logistic data=pmlr.pva_train plots(only)=
          (effect(clband x=(pep_star recent_avg_gift_amt
          frequency_status_97nk)) oddsratio (type=horizontalstat));
  class pep_star (param=ref ref='0');
  model target_b(event='1')=pep_star recent_avg_gift_amt
```

```
            frequency_status_97nk / clodds=pl;
  effectplot slicefit(sliceby=pep_star x=recent_avg_gift_amt) / noobs;
  effectplot slicefit(sliceby=pep_star x=frequency_status_97nk) / noobs;
  score data=pmlr.pva_train out=work.scopva_train priorevent=&ex_pi1;
run;

title1 "Adjusted Predicted Probabilities of the Veteran's Organization Data";
proc print data=work.scopva_train(obs=10);
  var p_1 pep_star recent_avg_gift_amt frequency_status_97nk;

run;
title;



/* ===================================================== */
/* Lesson 3, Practice 1
  Practice: Imputing Missing Values                  */
/* ===================================================== */

data pmlr.pva_train_mi(drop=i);
  set pmlr.pva_train;
  /* name the missing indicator variables */
  array mi{*} mi_DONOR_AGE mi_INCOME_GROUP
          mi_WEALTH_RATING;
  /* select variables with missing values */
  array x{*} DONOR_AGE INCOME_GROUP WEALTH_RATING;
  do i=1 to dim(mi);
    mi{i}=(x{i}=.);
    nummiss+mi{i};
```

```
  end;
run;

proc rank data=pmlr.pva_train_mi out=work.pva_train_rank
        groups=3;
  var recent_response_prop recent_avg_gift_amt;
  ranks grp_resp grp_amt;
run;

proc sort data=work.pva_train_rank out=work.pva_train_rank_sort;
  by grp_resp grp_amt;
run;

proc stdize data=work.pva_train_rank_sort method=median
         reponly out=pmlr.pva_train_imputed;
  by grp_resp grp_amt;
  var DONOR_AGE INCOME_GROUP WEALTH_RATING;
run;

options nolabel;
proc means data=pmlr.pva_train_imputed median;
  class grp_resp grp_amt;
  var DONOR_AGE INCOME_GROUP WEALTH_RATING;
run;
options label;


/* ===================================================== */
/* Lesson 3, Practice 2
```

```
  Practice: Collapsing the Levels of a Nominal Input

  Note: After you submit this code, a note in the log
  indicates that argument 3 to the LOGSDF function
  is invalid. You can ignore this note; it is not
  important for this analysis. The note pertains
  to the situation in which the number of clusters is 1.
  In this case, the degrees of freedom is 0 (degrees of
  freedom is equal to the number of clusters minus 1) and
  the mathematical operation cannot be performed in the
  LOGSDF function. Therefore, the log of the p-value is
  set to missing.                          */
/* ===================================================== */

proc means data=pmlr.pva_train_imputed noprint nway;
  class cluster_code;
  var target_b;
  output out=work.level mean=prop;
run;

ods output clusterhistory=work.cluster;

proc cluster data=work.level method=ward
         outtree=work.fortree
         plots=(dendrogram(horizontal height=rsq));
  freq _freq_;
  var prop;
  id cluster_code;
run;
```

```
proc freq data=pmlr.pva_train_imputed noprint;
  tables cluster_code*target_b / chisq;
  output out=work.chi(keep=_pchi_) chisq;
run;

data work.cutoff;
  if _n_=1 then set work.chi;
  set cluster;
  chisquare=_pchi_*rsquared;
  degfree=numberofclusters-1;
  logpvalue=logsdf('CHISQ',chisquare,degfree);
run;

title1 "Plot of the Log of the P-Value by Number of Clusters";
proc sgplot data=work.cutoff;
  scatter y=logpvalue x=numberofclusters
        / markerattrs=(color=blue symbol=circlefilled);
  xaxis label="Number of Clusters";
  yaxis label="Log of P-Value" min=-40 max=0;
run;

title1;

%global ncl;

proc sql;
  select NumberOfClusters into :ncl
  from work.cutoff
```

```
  having logpvalue=min(logpvalue);
quit;

proc tree data=work.fortree nclusters=&ncl
        out=work.clus noprint;
  id cluster_code;
run;

proc sort data=work.clus;
  by clusname;
run;

title1 "Cluster Assignments";
proc print data=work.clus;
  by clusname;
  id clusname;
run;

filename clcode "&PMLRfolder/cluster_code.sas";

data _null_;
  file clcode;
  set work.clus end=last;
  if _n_=1 then put "select (cluster_code);";
  put "  when ('" cluster_code +(-1) "')
        cluster_clus='" cluster +(-1) "';";
  if last then do;
    put "  otherwise cluster_clus='U';" / "end;";
  end;
```

```
run;

data pmlr.pva_train_imputed_clus;
  set pmlr.pva_train_imputed;
  %include clcode;
run;



/* ===================================================== */
/* Lesson 3, Practice 3
  Practice: Computing the Smoothed Weight of Evidence   */
/* ===================================================== */

%global rho1_ex;
proc sql noprint;
  select mean(target_b) into :rho1_ex
  from pmlr.pva_train_imputed;
run;

proc means data=pmlr.pva_train_imputed
        sum nway noprint;
  class cluster_code;
  var target_b;
  output out=work.counts sum=events;
run;

filename clswoe "&PMLRfolder/swoe_cluster.sas";

data _null_;
```

```
  file clswoe;
  set work.counts end=last;
    logit=log((events + &rho1_ex*24)/
           (_FREQ_ - events + (1-&rho1_ex)*24));
  if _n_=1 then put "select (cluster_code);" ;
  put "  when ('" cluster_code +(-1) "') cluster_swoe=" logit ";" ;
  if last then do;
    logit=log(&rho1_ex/(1-&rho1_ex));
    put "  otherwise cluster_swoe=" logit ";" / "end;";
  end;
run;

data pmlr.pva_train_imputed_swoe;
  set pmlr.pva_train_imputed;
  %include clswoe;
run;

title;

proc print data=pmlr.pva_train_imputed_swoe(obs=1);
  where cluster_code = "01";
  var cluster_code cluster_swoe;
run;



/* ====================================================== */
/* Lesson 3, Practice 4
  Practice: Reducing Redundancy by Clustering Variables */
/* ====================================================== */
```

```
/*Note: If you run this code in 32-bit SAS, the variable
  assignments to clusters might vary from what is shown
  in the results in this course. This discrepancy does
  not affect the results of the remaining practices in
  this course.
*/

%let ex_inputs= MONTHS_SINCE_ORIGIN
DONOR_AGE IN_HOUSE INCOME_GROUP PUBLISHED_PHONE
MOR_HIT_RATE WEALTH_RATING MEDIAN_HOME_VALUE
MEDIAN_HOUSEHOLD_INCOME PCT_OWNER_OCCUPIED
PER_CAPITA_INCOME PCT_MALE_MILITARY
PCT_MALE_VETERANS PCT_VIETNAM_VETERANS
PCT_WWII_VETERANS PEP_STAR RECENT_STAR_STATUS
FREQUENCY_STATUS_97NK RECENT_RESPONSE_PROP
RECENT_AVG_GIFT_AMT RECENT_CARD_RESPONSE_PROP
RECENT_AVG_CARD_GIFT_AMT RECENT_RESPONSE_COUNT
RECENT_CARD_RESPONSE_COUNT LIFETIME_CARD_PROM
LIFETIME_PROM LIFETIME_GIFT_AMOUNT
LIFETIME_GIFT_COUNT LIFETIME_AVG_GIFT_AMT
LIFETIME_GIFT_RANGE LIFETIME_MAX_GIFT_AMT
LIFETIME_MIN_GIFT_AMT LAST_GIFT_AMT
CARD_PROM_12 NUMBER_PROM_12 MONTHS_SINCE_LAST_GIFT
MONTHS_SINCE_FIRST_GIFT STATUS_FL STATUS_ES
home01 nses1 nses3 nses4 nses_ nurbr nurbu nurbs
nurbt nurb_;

ods select none;
```

```
ods output clusterquality=work.summary
        rsquare=work.clusters;

proc varclus data=pmlr.pva_train_imputed_swoe
         hi maxeigen=0.70;
  var &ex_inputs mi_DONOR_AGE mi_INCOME_GROUP
     mi_WEALTH_RATING cluster_swoe;
run;

ods select all;

data _null_;
  set work.summary;
  call symput('nvar',compress(NumberOfClusters));
run;

title1 "Variables by Cluster";
proc print data=work.clusters noobs label split='*';
  where NumberOfClusters=&nvar;
  var Cluster Variable RSquareRatio;
  label RSquareRatio="1 - RSquare*Ratio";
run;

title1 "Variation Explained by Clusters";
proc print data=work.summary label;
run;
title1 ;
```

```
/* ======================================================= */
/* Lesson 3, Practice 5
  Practice: Performing Variable Screening             */
/* ======================================================= */

%let ex_reduced=
LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE
FREQUENCY_STATUS_97NK MONTHS_SINCE_LAST_GIFT  nses_
mi_DONOR_AGE PCT_MALE_VETERANS PCT_MALE_MILITARY
PCT_WWII_VETERANS LIFETIME_AVG_GIFT_AMT cluster_swoe
PEP_STAR nurbu nurbt home01 nurbr DONOR_AGE STATUS_FL
MOR_HIT_RATE nses4 INCOME_GROUP RECENT_STAR_STATUS IN_HOUSE
WEALTH_RATING PUBLISHED_PHONE PCT_OWNER_OCCUPIED nurbs;

ods select none;
ods output spearmancorr=work.spearman
         hoeffdingcorr=work.hoeffding;

proc corr data=pmlr.pva_train_imputed_swoe
        spearman hoeffding;
  var target_b;
  with &ex_reduced;
run;

ods select all;

proc sort data=work.spearman;
   by variable;
run;
```

```
proc sort data=work.hoeffding;
   by variable;
run;

data work.correlations;
  attrib variable length=$32;
  merge work.spearman(rename=
        (target_b=scorr ptarget_b=spvalue))
        work.hoeffding
        (rename=(target_b=hcorr ptarget_b=hpvalue));
  by variable;
  scorr_abs=abs(scorr);
  hcorr_abs=abs(hcorr);
run;

proc rank data=work.correlations
         out=work.correlations1 descending;
   var scorr_abs hcorr_abs;
   ranks ranksp rankho;
run;

proc sort data=work.correlations1;
  by ranksp;
run;

title1 "Rank of Spearman Correlations and Hoeffding Correlations";
proc print data=work.correlations1 label split='*';
  var variable ranksp rankho scorr spvalue hcorr hpvalue;
```

```
  label ranksp='Spearman rank*of variables'
      scorr='Spearman Correlation'
      spvalue='Spearman p-value'
      rankho='Hoeffding rank*of variables'
      hcorr='Hoeffding Correlation'
      hpvalue='Hoeffding p-value';
run;

%global vref href;
proc sql noprint;
  select min(ranksp) into :vref
  from (select ranksp
      from work.correlations1
      having spvalue > .5);
  select min(rankho) into :href
  from (select rankho
      from work.correlations1
      having hpvalue > .5);
quit;

title1 "Scatter Plot of the Ranks of Spearman vs. Hoeffding";
proc sgplot data=work.correlations1;
  refline &vref / axis=y;
  refline &href / axis=x;
  scatter y=ranksp x=rankho / datalabel=variable;
  yaxis label="Rank of Spearman";
  xaxis label="Rank of Hoeffding";
run;
```

```
/* ======================================================= */
/* Lesson 3, Practice 6
   Practice: Creating Empirical Logit Plots           */
/* ======================================================= */

%global var;
%let var=LAST_GIFT_AMT;

proc rank data=pmlr.pva_train_imputed_swoe
        groups=20 out=work.ranks;
   var &var;
   ranks bin;
run;

proc means data=work.ranks noprint nway;
   class bin;
   var target_b &var;
   output out=work.bins sum(target_b)=target_b
        mean(&var)=&var;
run;

data work.bins;
   set work.bins;
   elogit=log((target_b+(sqrt(_FREQ_ )/2))/
        ( _FREQ_ -target_b+(sqrt(_FREQ_ )/2)));
run;
```

```
title1 "Empirical Logit against &var";
proc sgplot data=work.bins;
  reg y=elogit x=&var /
     curvelabel="Linear Relationship?"
     curvelabelloc=outside
     lineattrs=(color=ligr);
  series y=elogit x=&var;
run;
title1;

title1 "Empirical Logit against Binned &var";
proc sgplot data=work.bins;
  reg y=elogit x=bin /
     curvelabel="Linear Relationship?"
     curvelabelloc=outside
     lineattrs=(color=ligr);
  series y=elogit x=bin;
run;
title1;

/* Practice: l3p7.sas step 1 */
%global ex_screened;
%let ex_screened=LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE
          FREQUENCY_STATUS_97NK MONTHS_SINCE_LAST_GIFT  nses_
          mi_DONOR_AGE PCT_MALE_VETERANS PCT_MALE_MILITARY
          PCT_WWII_VETERANS LIFETIME_AVG_GIFT_AMT cluster_swoe
          PEP_STAR nurbu nurbt home01 nurbr DONOR_AGE STATUS_FL
          MOR_HIT_RATE nses4 INCOME_GROUP RECENT_STAR_STATUS
          IN_HOUSE WEALTH_RATING nurbs;
```

```
/* Solution for l3p7 */

/* step 2 */

%global ex_screened;
%let ex_screened=LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE
        FREQUENCY_STATUS_97NK MONTHS_SINCE_LAST_GIFT  nses_
        mi_DONOR_AGE PCT_MALE_VETERANS PCT_MALE_MILITARY
        PCT_WWII_VETERANS LIFETIME_AVG_GIFT_AMT cluster_swoe
        PEP_STAR nurbu nurbt home01 nurbr DONOR_AGE STATUS_FL
        MOR_HIT_RATE nses4 INCOME_GROUP RECENT_STAR_STATUS
        IN_HOUSE WEALTH_RATING nurbs;


/* step 3 */

%global sl;

title1 "P-Value for Entry and Retention";
proc sql;
  select 1-probchi(log(sum(target_b ge 0)),1) into :sl
  from pmlr.pva_train_imputed_swoe;
quit;
title1;


/* step 4 */

title1 "Interaction Detection using Forward Selection";
```

```
proc logistic data=pmlr.pva_train_imputed_swoe namelen=50;
  model target_b(event='1')= &ex_screened
      LIFETIME_GIFT_COUNT|LAST_GIFT_AMT|MEDIAN_HOME_VALUE|
      FREQUENCY_STATUS_97NK|MONTHS_SINCE_LAST_GIFT|nses_|
      mi_DONOR_AGE|PCT_MALE_VETERANS|PCT_MALE_MILITARY|
      PCT_WWII_VETERANS|LIFETIME_AVG_GIFT_AMT|cluster_swoe|
      PEP_STAR|nurbu|nurbt|home01|nurbr|DONOR_AGE|STATUS_FL|
      MOR_HIT_RATE|nses4|INCOME_GROUP|RECENT_STAR_STATUS|
      IN_HOUSE|WEALTH_RATING|nurbs @2 / include=26 clodds=pl
     selection=forward slentry=&sl;
run;
title1;
```

P-Value for Entry and Retention

0.002449

Interaction Detection using Forward Selection

The LOGISTIC Procedure

Model Information	
Data Set	PMLR.PVA_TRAIN_IMPUTED_SWOE
Response Variable	TARGET_B
Number of Response Levels	2
Model	binary logit
Optimization Technique	Fisher's scoring

Number of Observations Read	9687
Number of Observations Used	9687

Response Profile		
Ordered Value	TARGET_B	Total Frequency
1	0	7265
2	1	2422

Probability modeled is TARGET_B=1.

Forward Selection Procedure

The following effects will be included in each model:

Intercept LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE FREQUENCY_STATUS_97NK MONTHS_SINCE_LAST_GIFT nses_ mi_DONOR_AGE PCT_MALE_VETERANS PCT_MALE_MILITARY PCT_WWII_VETERANS LIFETIME_AVG_GIFT_AMT cluster_swoe PEP_STAR nurbu nurbt home01 nurbr DONOR_AGE STATUS_FL MOR_HIT_RATE nses4 INCOME_GROUP RECENT_STAR_STATUS IN_HOUSE WEALTH_RATING nurbs

Step 0. The INCLUDE effects were entered.

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	10897.230	10551.519
SC	10904.409	10745.340
-2 Log L	10895.230	10497.519

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	397.7113	26	<.0001
Score	395.1719	26	<.0001
Wald	378.4073	26	<.0001

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
325.7358	307	0.2212

Step 1. Effect LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT entered:

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	10897.230	10538.651
SC	10904.409	10739.650
-2 Log L	10895.230	10482.651

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	412.5793	27	<.0001
Score	409.9007	27	<.0001
Wald	390.7876	27	<.0001

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
302.6965	306	0.5426

Step 2. Effect LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS entered:

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	10897.230	10529.687
SC	10904.409	10737.865
-2 Log L	10895.230	10471.687

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	423.5434	28	<.0001
Score	420.3558	28	<.0001
Wald	398.3173	28	<.0001

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
290.1709	305	0.7202

Step 3. Effect LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT entered:

Model Convergence Status
Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	10897.230	10521.304
SC	10904.409	10736.660
-2 Log L	10895.230	10461.304

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	433.9266	29	<.0001
Score	436.1775	29	<.0001
Wald	408.5931	29	<.0001

Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
282.3477	304	0.8086

Note: No (additional) effects met the 0.002449 significance level for entry into the model.

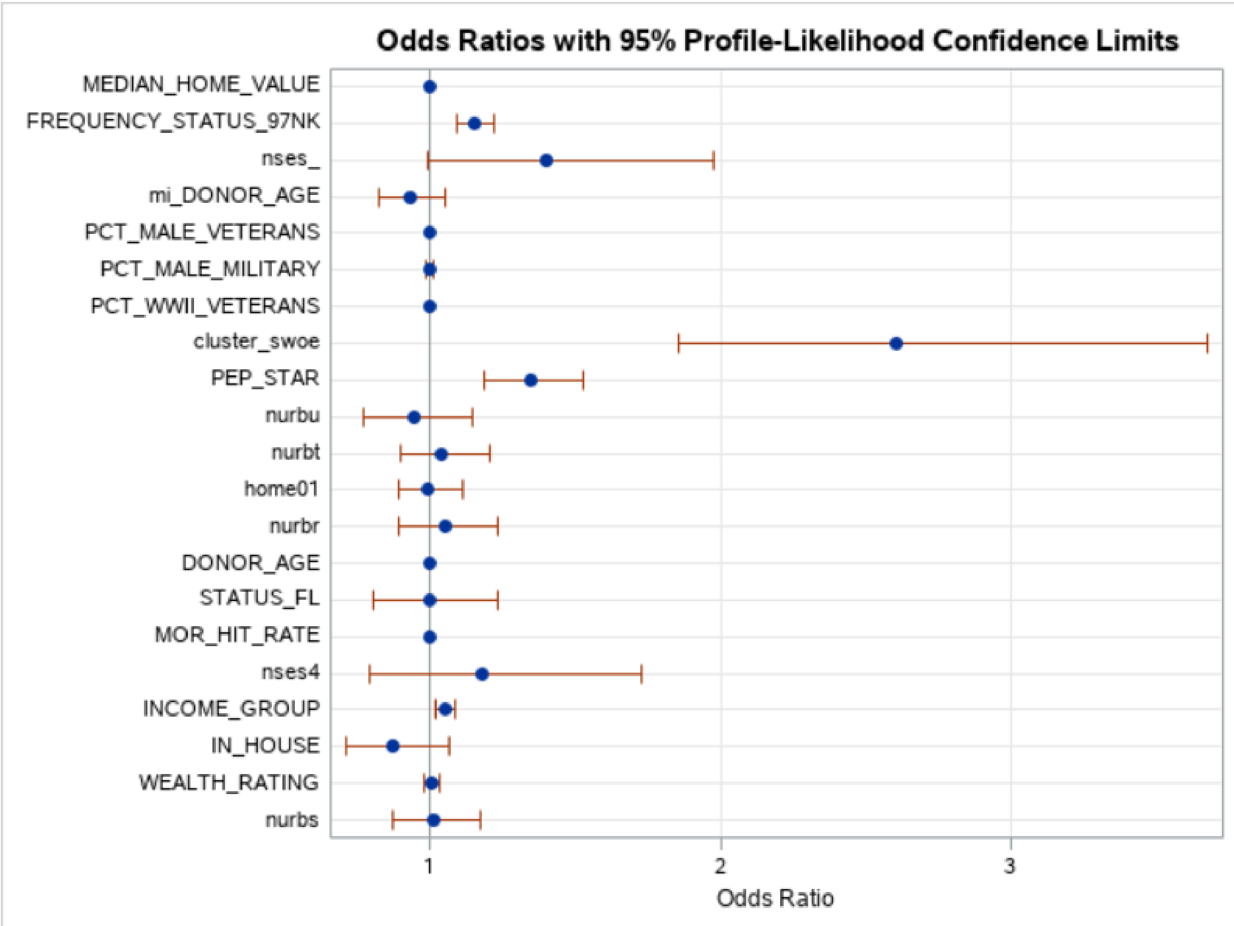
Summary of Forward Selection					
Step	Effect Entered	DF	Number In	Score Chi-Square	Pr > ChiSq
1	LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT	1	27	23.1980	<.0001
2	LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS	1	28	12.8362	0.0003
3	LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT	1	29	10.1872	0.0014

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-0.6647	0.3584	3.4393	0.0637
LIFETIME_GIFT_COUNT	1	0.0363	0.0108	11.2376	0.0008
LAST_GIFT_AMT	1	-0.0157	0.00429	13.4230	0.0002
MEDIAN_HOME_VALUE	1	0.000100	0.000029	11.8486	0.0006
FREQUENCY_STATUS_97NK	1	0.1466	0.0284	26.6538	<.0001
MONTHS_SINCE_LAST_GIFT	1	-0.00729	0.0105	0.4868	0.4854
nses_	1	0.3404	0.1750	3.7837	0.0518
mi_DONOR_AGE	1	-0.0669	0.0619	1.1678	0.2798
PCT_MALE_VETERANS	1	-0.00100	0.00257	0.1506	0.6979
PCT_MALE_MILITARY	1	0.00231	0.00526	0.1924	0.6609
PCT_WWII_VETERANS	1	0.00257	0.00157	2.6770	0.1018
LIFETIME_AVG_GIFT_AMT	1	-0.0165	0.00656	6.3696	0.0116
cluster_swoe	1	0.9581	0.1744	30.1629	<.0001

PEP_STAR	1	0.2983	0.0650	21.0944	<.0001
nurbu	1	-0.0562	0.1005	0.3132	0.5757
nurbt	1	0.0424	0.0747	0.3220	0.5704
home01	1	-0.00248	0.0555	0.0020	0.9644
nurbr	1	0.0509	0.0820	0.3851	0.5349
DONOR_AGE	1	0.00380	0.00175	4.7392	0.0295
STATUS_FL	1	0.00191	0.1080	0.0003	0.9859
MOR_HIT_RATE	1	0.00179	0.00254	0.4976	0.4806
nses4	1	0.1677	0.1982	0.7158	0.3975
INCOME_GROUP	1	0.0538	0.0165	10.6590	0.0011
RECENT_STAR_STATUS	1	-0.0529	0.0193	7.5163	0.0061
IN_HOUSE	1	-0.1349	0.1027	1.7252	0.1890
WEALTH_RATING	1	0.00982	0.0128	0.5870	0.4436
nurbs	1	0.0127	0.0748	0.0289	0.8650
LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT	1	-0.00193	0.000608	10.0893	0.0015
LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT	1	0.000243	0.000060	16.1677	<.0001
LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS	1	0.00375	0.00125	8.9715	0.0027

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	63.5	Somers' D	0.269
Percent Discordant	36.5	Gamma	0.269
Percent Tied	0.0	Tau-a	0.101
Pairs	17595830	c	0.635

Odds Ratio Estimates and Profile-Likelihood Confidence Intervals				
Effect	Unit	Estimate	95% Confidence Limits	
MEDIAN_HOME_VALUE	1.0000	1.000	1.000	1.000
FREQUENCY_STATUS_97NK	1.0000	1.158	1.095	1.224
nses_	1.0000	1.406	0.994	1.975
mi_DONOR_AGE	1.0000	0.935	0.828	1.056
PCT_MALE_VETERANS	1.0000	0.999	0.994	1.004
PCT_MALE_MILITARY	1.0000	1.002	0.991	1.012
PCT_WWII_VETERANS	1.0000	1.003	0.999	1.006
cluster_swoe	1.0000	2.607	1.853	3.672
PEP_STAR	1.0000	1.348	1.186	1.531
nurbu	1.0000	0.945	0.776	1.150
nurbt	1.0000	1.043	0.901	1.208
home01	1.0000	0.998	0.895	1.112
nurbr	1.0000	1.052	0.896	1.236
DONOR_AGE	1.0000	1.004	1.000	1.007
STATUS_FL	1.0000	1.002	0.809	1.235
MOR_HIT_RATE	1.0000	1.002	0.997	1.007
nses4	1.0000	1.183	0.794	1.730
INCOME_GROUP	1.0000	1.055	1.022	1.090
IN_HOUSE	1.0000	0.874	0.713	1.067
WEALTH_RATING	1.0000	1.010	0.985	1.036
nurbs	1.0000	1.013	0.875	1.173


Odds Ratios with 95% Profile-Likelihood Confidence Limits
MEDIAN_HOME_VALUE
FREQUENCY_STATUS_97NK
nses_
mi_DONOR_AGE
PCT_MALE_VETERANS
PCT_MALE_MILITARY
PCT_WWII_VETERANS
cluster_swoe
PEP_STAR
nurbu
nurbt
home01
nurbr
DONOR_AGE
STATUS_FL
MOR_HIT_RATE
nses4
INCOME_GROUP
IN_HOUSE
WEALTH_RATING
nurbs
1
2
3
Odds Ratio

Demo Using Backward Elimination to Subset Variables

```
title1 "Backward Selection for Variable Annuity Data Set";
proc logistic data=work.train_imputed_swoe_bins;
   class res (param=ref ref='S');
   model ins(event='1')= &screened res SavBal*B_DDABal MM*B_DDABal
                  branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
                  SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt
                  SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM
                  IRA*B_DDABal CD*MM MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC
                 / clodds=pl selection=backward slstay=&sl hier=single fast;
run;
```

Number of Observations Read	21512
Number of Observations Used	21512

Response Profile

Ordered Value	Ins	Total Frequency
1	0	14061
2	1	7451

Probability modeled is Ins=1.

Intercept SavBal Dep DDA CD Sav CC ATM MM branch_swoe Phone IRA IRABal B_DDABal ATMAmt ILS POS NSF CCPurc SDB DepAmt CCBal Inv InArea Age CashBk MICRScor Income Res SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt Sav*B_DDABal SavBal*SDB SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal CD*MM MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC

Step 1. Fast Backward Elimination:

Analysis of Effects Removed by Fast Backward Elimination						
Effect Removed	Chi-Square	DF	Pr > ChiSq	Residual Chi-Square	DF	Pr > Residual ChiSq
POS	0.0017	1	0.9673	0.0017	1	0.9673
Res	0.3178	2	0.8531	0.3195	3	0.9563
CCPurc	0.0713	1	0.7895	0.3907	4	0.9832
MICRScor	0.1243	1	0.7244	0.5151	5	0.9916
Age	0.2614	1	0.6091	0.7765	6	0.9927
InArea	0.2819	1	0.5954	1.0584	7	0.9938
Income	1.0045	1	0.3162	2.0629	8	0.9790
Phone	3.4393	1	0.0637	5.5023	9	0.7885
CCBal	4.2769	1	0.0386	9.7791	10	0.4601
MM*IRABal	8.6585	1	0.0033	18.4377	11	0.0720
IRABal	0.0702	1	0.7911	18.5078	12	0.1011

Summary of Backward Elimination						
Step	Effect Removed	DF	Number In	Wald Chi-Square	Pr > ChiSq	Variable Label
1	POS	1	48	0.0017	0.9673	Number Point of Sale
1	Res	2	47	0.3178	0.8531	Area Classification
1	CCPurc	1	46	0.0713	0.7895	Credit Card Purchases
1	MICRScor	1	45	0.1243	0.7244	
1	Age	1	44	0.2614	0.6091	Age
1	InArea	1	43	0.2819	0.5954	Local Address
1	Income	1	42	1.0045	0.3162	Income
1	Phone	1	41	3.4393	0.0637	Number Telephone Banking
1	CCBal	1	40	4.2769	0.0386	Credit Card Balance
1	MM*IRABal	1	39	8.6585	0.0033	
1	IRABal	1	38	0.0702	0.7911	IRA Balance

Type 3 Analysis of Effects			
Effect	DF	Wald Chi-Square	Pr > ChiSq
SavBal	1	298.8926	<.0001
Dep	1	4.2465	0.0393
DDA	1	8.3313	0.0039
CD	1	222.9731	<.0001
Sav	1	121.9672	<.0001
CC	1	5.1065	0.0238
ATM	1	0.2032	0.6522
MM	1	142.6023	<.0001
branch_swoe	1	128.1414	<.0001
IRA	1	32.2143	<.0001
B_DDABal	1	738.0455	<.0001
ATMAmt	1	44.5548	<.0001
ILS	1	11.1663	0.0008
NSF	1	45.1865	<.0001
SDB	1	6.6264	0.0100
DepAmt	1	1.5137	0.2186
Inv	1	18.6934	<.0001
CashBk	1	10.0010	0.0016
SavBal*B_DDABal	1	193.7508	<.0001
MM*B_DDABal	1	50.6903	<.0001
branch_swoe*ATMAmt	1	51.3337	<.0001
Sav*B_DDABal	1	61.9298	<.0001
SavBal*SDB	1	11.5998	0.0007
SavBal*DDA	1	29.7241	<.0001
ATMAmt*DepAmt	1	24.2998	<.0001

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	-2.0561	0.0996	426.2679	<.0001
SavBal	1	0.000166	9.605E-6	298.8926	<.0001
Dep	1	-0.0517	0.0251	4.2465	0.0393
DDA	1	-0.1810	0.0627	8.3313	0.0039
CD	1	1.1427	0.0765	222.9731	<.0001
Sav	1	1.0092	0.0914	121.9672	<.0001
CC	1	0.1103	0.0488	5.1065	0.0238
ATM	1	0.0280	0.0622	0.2032	0.6522
MM	1	1.7524	0.1467	142.6023	<.0001
branch_swoe	1	0.9132	0.0807	128.1414	<.0001
IRA	1	1.1104	0.1956	32.2143	<.0001
B_DDABal	1	0.0284	0.00104	738.0455	<.0001
ATMAmt	1	0.000199	0.000030	44.5548	<.0001

Sav*NSF	1	-0.4483	0.1276	12.3484	0.0004
DDA*ATMAmt	1	0.000078	0.000019	17.2617	<.0001
Dep*ATM	1	-0.1075	0.0272	15.6329	<.0001
IRA*B_DDABal	1	-0.0107	0.00290	13.7202	0.0002
CD*MM	1	-0.5033	0.1202	17.5219	<.0001
CD*Sav	1	-0.3183	0.0952	11.1855	0.0008
B_DDABal*CashBk	1	0.0185	0.00584	10.0421	0.0015
Sav*CC	1	0.2346	0.0700	11.2318	0.0008

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	78.9	Somers' D	0.577
Percent Discordant	21.1	Gamma	0.578
Percent Tied	0.0	Tau-a	0.261
Pairs	104768511	c	0.789

SavBal*B_DDABal	1	-1.67E-6	1.201E-7	193.7508	<.0001

```
/* ===================================================== */
/* Lesson 3, Section 5: l3d8b.sas
   Demonstration: Using Backward Elimination to Subset the
   Variables
   [m643_5_n; derived from pmlr03d08.sas]           */
/* ===================================================== */

title1 "Backward Selection for Variable Annuity Data Set";
proc logistic data=work.train_imputed_swoe_bins;
   class res (param=ref ref='S');
   model ins(event='1')= &screened res SavBal*B_DDABal MM*B_DDABal
            branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
            SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt
            SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM
            IRA*B_DDABal CD*MM MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC
           / clodds=pl
      selection=backward slstay=&sl hier=single fast;
run;
```

Question 3.10

Consider the following MODEL statement in PROC LOGISTIC:

model ins(event='1') = SavBal|Age|IRABal @2;

Which effects does it model?

SavBal, **Age**, **IRABal**, **SavBal*Age**, **SavBal*IRABal**, and **Age*IRABal**

The bar notation with @2 constructs a model with all the main effects and the two-factor interactions.

```
/* Solution for l3p8 */

/* step 2 */

title1 "Backward Selection for Variable Annuity Data Set";
proc logistic data=pmlr.pva_train_imputed_swoe namelen=50;
  model target_b(event='1')= &ex_screened
      LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT
      LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS
      LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT
      / clodds=pl selection=backward slstay=&sl hier=single
      fast;
run;
title1;
```

Number of Observations Read	9687
Number of Observations Used	9687

Response Profile		
Ordered Value	TARGET_B	Total Frequency
1	0	7265
2	1	2422

Probability modeled is TARGET_B=1.

Backward Elimination Procedure

Step 0. The following effects were entered:

Intercept LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE FREQUENCY_STATUS_97NK MONTHS_SINCE_LAST_GIFT nses_ mi_DONOR_AGE PCT_MALE_VETERANS PCT_MALE_MILITARY PCT_WWII_VETERANS LIFETIME_AVG_GIFT_AMT cluster_swoe PEP_STAR nurbu nurbt home01 nurbr DONOR_AGE STATUS_FL MOR_HIT_RATE nses4 INCOME_GROUP RECENT_STAR_STATUS IN_HOUSE WEALTH_RATING nurbs LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT

Step 1. Fast Backward Elimination:

Analysis of Effects Removed by Fast Backward Elimination						
Effect Removed	Chi-Square	DF	Pr > ChiSq	Residual Chi-Square	DF	Pr > Residual ChiSq
STATUS_FL STATUS_FL	0.0003	1	0.9859	0.0003	1	0.9859
home01 home01	0.0020	1	0.9645	0.0023	2	0.9989
nurbs nurbs	0.0289	1	0.8650	0.0312	3	0.9985
PCT_MALE_VETERANS PCT_MALE_VETE	0.1600	1	0.6892	0.1912	4	0.9957
PCT_MALE_MILITARY PCT_MALE_MILI	0.1767	1	0.6742	0.3679	5	0.9962
nurbt nurbt	0.3333	1	0.5637	0.7012	6	0.9945
nurbr nurbr	0.1894	1	0.6634	0.8906	7	0.9964
MOR_HIT_RATE MOR_HIT_RATE	0.4537	1	0.5006	1.3443	8	0.9950
WEALTH_RATING WEALTH_RATING	0.4980	1	0.4804	1.8423	9	0.9937
nses4 nses4	0.6128	1	0.4337	2.4551	10	0.9915
nurbu nurbu	0.4164	1	0.5187	2.8715	11	0.9923
mi_DONOR_AGE mi_DONOR_AGE	1.4611	1	0.2268	4.3327	12	0.9767
IN_HOUSE IN_HOUSE	1.5824	1	0.2084	5.9150	13	0.9492
PCT_WWII_VETERANS PCT_WWII_VETE	2.2087	1	0.1372	8.1237	14	0.8828
nses_ nses_	2.4131	1	0.1203	10.5368	15	0.7847
DONOR_AGE DONOR_AGE	5.1482	1	0.0233	15.6851	16	0.4751
LIFETIME_GIFT_COUNT*MONTHS_SINCELIFETIME_GIFT	8.4930	1	0.0036	24.1781	17	0.1147
LIFETIME_GIFT_COUNT LIFETIME_GIFT	1.6454	1	0.1996	25.8235	18	0.1039
LIFETIME_AVG_GIFT_AMT*RECENT_STALIFETIME_AVG_	8.6003	1	0.0034	34.4238	19	0.0164
RECENT_STAR_STATUS RECENT_STAR_S	0.2137	1	0.6439	34.6375	20	0.0221

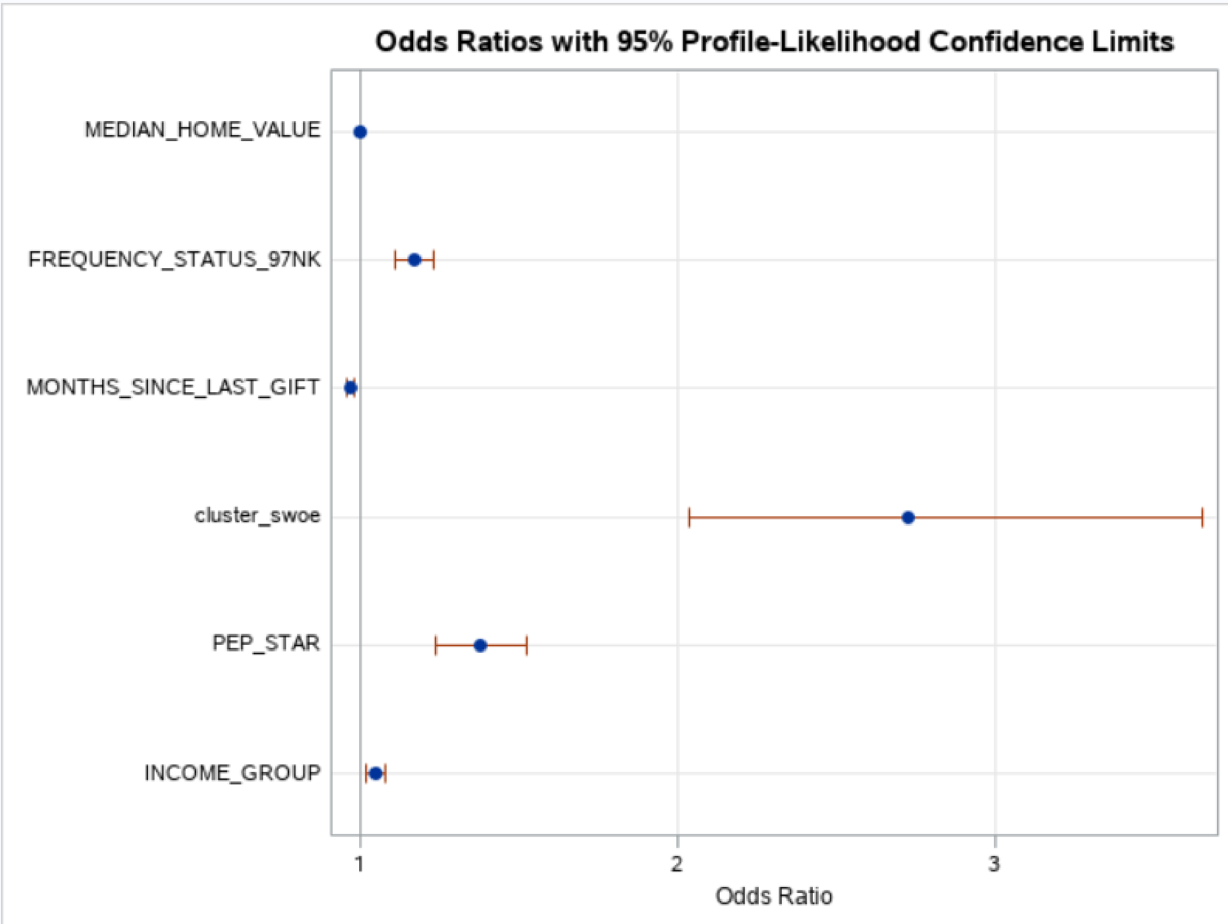
Residual Chi-Square Test		
Chi-Square	DF	Pr > ChiSq
39.1330	20	0.0064

Summary of Backward Elimination					
Step	Effect Removed	DF	Number In	Wald Chi-Square	Pr > ChiSq
1	STATUS_FL	1	28	0.0003	0.9859
1	home01	1	27	0.0020	0.9645
1	nurbs	1	26	0.0289	0.8650
1	PCT_MALE_VETERANS	1	25	0.1600	0.6892
1	PCT_MALE_MILITARY	1	24	0.1767	0.6742
1	nurbt	1	23	0.3333	0.5637
1	nurbr	1	22	0.1894	0.6634
1	MOR_HIT_RATE	1	21	0.4537	0.5006
1	WEALTH_RATING	1	20	0.4980	0.4804
1	nses4	1	19	0.6128	0.4337
1	nurbu	1	18	0.4164	0.5187
1	mi_DONOR_AGE	1	17	1.4611	0.2268
1	IN_HOUSE	1	16	1.5824	0.2084
1	PCT_WWII_VETERANS	1	15	2.2087	0.1372
1	nses_	1	14	2.4131	0.1203
1	DONOR_AGE	1	13	5.1482	0.0233
1	LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT	1	12	8.4930	0.0036
1	LIFETIME_GIFT_COUNT	1	11	1.6454	0.1996
1	LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS	1	10	8.6003	0.0034
1	RECENT_STAR_STATUS	1	9	0.2137	0.6439

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	0.1482	0.2498	0.3520	0.5530
LAST_GIFT_AMT	1	-0.0139	0.00426	10.6085	0.0011
MEDIAN_HOME_VALUE	1	0.000095	0.000026	13.4329	0.0002
FREQUENCY_STATUS_97NK	1	0.1565	0.0258	36.6681	<.0001
MONTHS_SINCE_LAST_GIFT	1	-0.0334	0.00620	29.1079	<.0001
LIFETIME_AVG_GIFT_AMT	1	-0.0138	0.00615	5.0581	0.0245
cluster_swoe	1	1.0032	0.1492	45.1805	<.0001
PEP_STAR	1	0.3191	0.0533	35.8039	<.0001
INCOME_GROUP	1	0.0484	0.0154	9.8491	0.0017
LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT	1	0.000227	0.000061	13.9023	0.0002

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	63.1	Somers' D	0.262
Percent Discordant	36.9	Gamma	0.262
Percent Tied	0.0	Tau-a	0.098
Pairs	17595830	c	0.631

Odds Ratio Estimates and Profile-Likelihood Confidence Intervals				
Effect	Unit	Estimate	95% Confidence Limits	
MEDIAN_HOME_VALUE	1.0000	1.000	1.000	1.000
FREQUENCY_STATUS_97NK	1.0000	1.169	1.112	1.230
MONTHS_SINCE_LAST_GIFT	1.0000	0.967	0.955	0.979
cluster_swoe	1.0000	2.727	2.037	3.656
PEP_STAR	1.0000	1.376	1.239	1.528
INCOME_GROUP	1.0000	1.050	1.018	1.082



Demo Displaying Odds Ratio for Variables Involved in Interactions

SavBal*B_DDABal	1	-1.67E-6	1.213E-7	189.7509	<.0001

```
title1 "Candidate Model for Variable Annuity Data Set";
ods select OddsRatiosPL;
proc logistic data=work.train_imputed_swoe_bins;
   model ins(event='1')= SavBal Dep DDA CD Sav CC ATM MM branch_swoe IRA I
                         DepAmt Inv SavBal*B_DDABal MM*B_DDABal branch_sw
                         SavBal*SDB SavBal*DDA AtmAmt*DepAmt B_DDABAL*ATM
                         SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM I
                         CD*MM CD*Sav Sav*CC / clodds=pl;
   oddsratio B_DDABAL / at(savbal=0, 1211, 52299) cl=pl;
run;
```

Candidate Model for Variable Annuity Data Set

The LOGISTIC Procedure

Odds Ratio Estimates and Profile-Likelihood Confidence Intervals			
Odds Ratio	**Estimate**	**95% Confidence Limits**	
B_DDABal at SavBal=0 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	1.020	1.019	1.021
B_DDABal at SavBal=1211 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	1.018	1.017	1.019
B_DDABal at SavBal=52299 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	0.935	0.923	0.946

```
/* ===================================================== */
/* Lesson 3, Section 5: l3d8c.sas
   Demonstration: Displaying Odds Ratios for Variables
   Involved in Interactions
   [m643_5_o; derived from pmlr03d08.sas]            */
/* ===================================================== */

title1 "Candidate Model for Variable Annuity Data Set";
ods select OddsRatiosPL;
proc logistic data=work.train_imputed_swoe_bins;
   model ins(event='1')= SavBal Dep DDA CD Sav CC ATM MM branch_swoe IRA B_DDABal
                  ATMAmt ILS NSF SDB
                  DepAmt Inv SavBal*B_DDABal MM*B_DDABal
                  branch_swoe*ATMAmt Sav*B_DDABal
                  SavBal*SDB SavBal*DDA AtmAmt*DepAmt B_DDABAL*ATMAmt SavBal*IRA
                  SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal
                  CD*MM CD*Sav Sav*CC / clodds=pl;
   oddsratio B_DDABAL / at(savbal=0, 1211, 52299) cl=pl;
run;
```

Candidate Model for Variable Annuity Data Set

The LOGISTIC Procedure

Odds Ratio Estimates and Profile-Likelihood Confidence Intervals			
Odds Ratio	Estimate	95% Confidence Limits	
B_DDABal at SavBal=0 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	1.020	1.019	1.021
B_DDABal at SavBal=1211 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	1.018	1.017	1.019
B_DDABal at SavBal=52299 Sav=0.4684 MM=0.1137 IRA=0.0529 ATMAmt=1252.6	0.935	0.923	0.946

Demo Creating an Interaction Plot

```
/*----  MACRO INTERACT  ----*\
Reserved data set names: work.percentiles
                         work.plot
\*-------------------------*/
%macro interact(data=,target=,event=,inputs=,var1=,var2=,mean_inputs=);

proc logistic data=&data noprint;
   model &target(event="&event")= &inputs;
   code file="&PMLRfolder\interaction.txt";
run;

proc univariate data=&data noprint;
   var &var1 &var2;
   output out=work.percentiles pctlpts=5 25 50 75 95 pctlpre=&var1._p &var2._p;
run;
```

```
data _null_;
   set work.percentiles;
   call symput("&var1._p5",&var1._p5);
   call symput("&var1._p25",&var1._p25);
   call symput("&var1._p50",&var1._p50);
   call symput("&var1._p75",&var1._p75);
   call symput("&var1._p95",&var1._p95);
   call symput("&var2._p5",&var2._p5);
   call symput("&var2._p25",&var2._p25);
   call symput("&var2._p50",&var2._p50);
   call symput("&var2._p75",&var2._p75);
   call symput("&var2._p95",&var2._p95);
run;
```

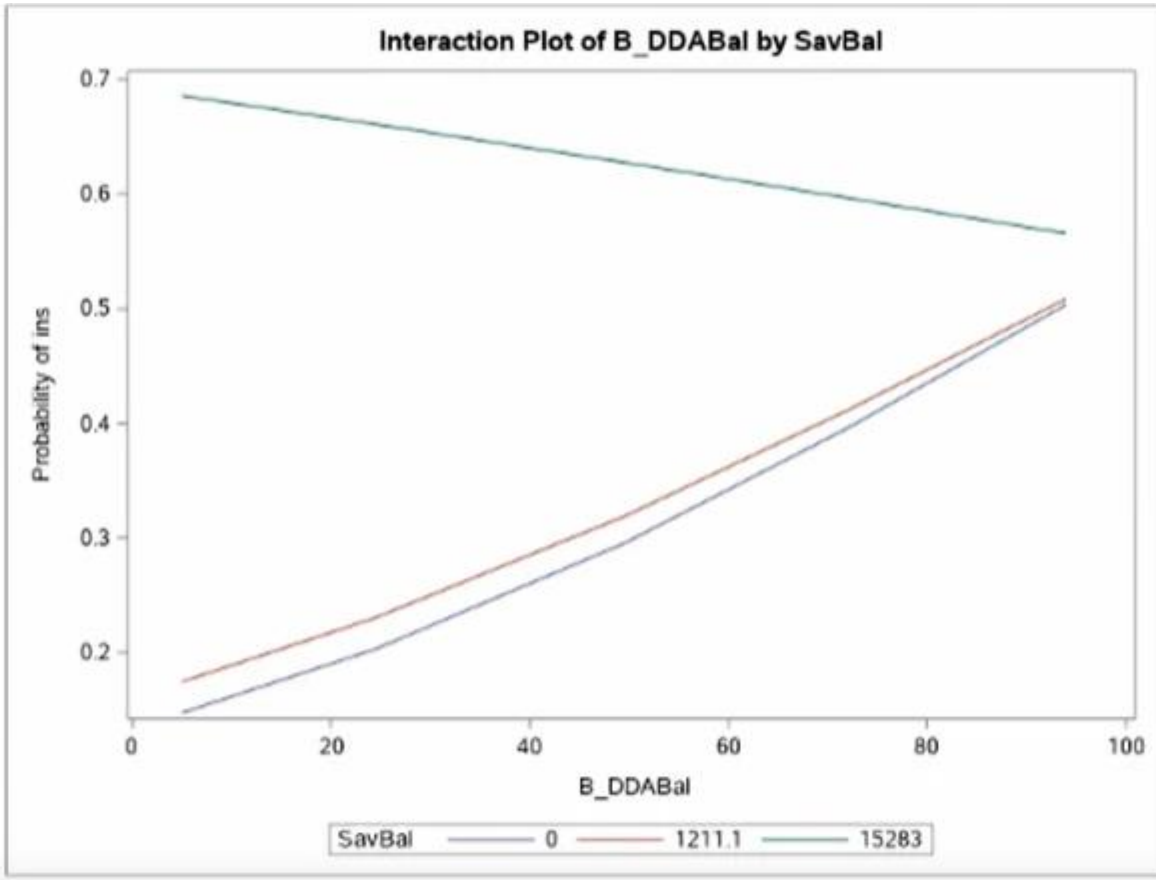
```
proc means data=&data noprint;
   var &mean_inputs;
   output out=work.plot mean=;
run;
```

```
data work.plot(drop=_type_ _freq_);
   set work.plot;
   do &var2=&&&var2._p5,&&&var2._p25,&&&var2._p50,&&&var2._p75,&&&var2._p95;
      do &var1=&&&var1._p5,&&&var1._p25,&&&var1._p50,&&&var1._p75,&&&var1._p95;
        %include "&PMLRfolder\interaction.txt";
         output;
     end;
   end;
run;
```

```
title1 "Interaction Plot of &var2 by &var1";
proc sgplot data=work.plot;
   series y=p_&target&event x=&var2 / group=&var1;
   yaxis label="Probability of &target";
run;

%mend interact;
```

```
%interact(data=train_imputed_swoe_bins,target=ins,event=1,inputs=SavBal Dep DDA CD
_imputed_swoe_bins,target=ins,event=1,inputs=SavBal Dep DDA CD Sav CC ATM MM
     branch_swoe IRA B_DDABal ATMAmt ILS NSF SDB
     DepAmt Inv SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt Sav*B_DDABal
     SavBal*SDB SavBal*DDA AtmAmt*DepAmt B_DDABAL*ATMAmt SavBal*ATMAmt SavBal*IRA
     SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal
     CD*MM CD*Sav B_DDABal*CashBk Sav*CC,var1=SavBal,var2=B_DDABal,
     mean_inputs=Dep CashBk DDA CD Sav CC ATM MM branch_swoe IRA
     ATMAmt ILS NSF SDB DepAmt Inv);
```




```
/* ======================================================= */
/* Lesson 3, Section 5: l3d8d.sas
  Demonstration: Creating an Interaction Plot
  [m643_5_r; derived from pmlr03d08.sas]              */
/* ======================================================= */


/*----  MACRO INTERACT  ----*\
Reserved data set names: work.percentiles
                  work.plot
\*--------------------------*/
%macro interact(data=,target=,event=,inputs=,var1=,var2=,mean_inputs=);


proc logistic data=&data noprint;
  model &target(event="&event")= &inputs;
```

```
  code file="&PMLRfolder/interaction.txt";
run;

proc univariate data=&data noprint;
  var &var1 &var2;
  output out=work.percentiles pctlpts=5 25 50 75 95 pctlpre=&var1._p &var2._p;
run;

data _null_;
  set work.percentiles;
  call symput("&var1._p5",&var1._p5);
  call symput("&var1._p25",&var1._p25);
  call symput("&var1._p50",&var1._p50);
  call symput("&var1._p75",&var1._p75);
  call symput("&var1._p95",&var1._p95);
  call symput("&var2._p5",&var2._p5);
  call symput("&var2._p25",&var2._p25);
  call symput("&var2._p50",&var2._p50);
  call symput("&var2._p75",&var2._p75);
  call symput("&var2._p95",&var2._p95);
run;

proc means data=&data noprint;
  var &mean_inputs;
  output out=work.plot mean=;
run;

data work.plot(drop=_type_ _freq_);
  set work.plot;
```

```
  do &var2=&&&var2._p5,&&&var2._p25,&&&var2._p50,&&&var2._p75,&&&var2._p95;
    do &var1=&&&var1._p5,&&&var1._p25,&&&var1._p50,&&&var1._p75,&&&var1._p95;
      %include "&PMLRfolder/interaction.txt";
       output;
    end;
  end;
run;

title1 "Interaction Plot of &var2 by &var1";
proc sgplot data=work.plot;
  series y=p_&target&event x=&var2 / group=&var1;
  yaxis label="Probability of &target";
run;

%mend interact;

%interact(data=train_imputed_swoe_bins,target=ins,event=1,
          inputs=SavBal Dep DDA CD Sav CC ATM MM branch_swoe
          IRA B_DDABal ATMAmt ILS NSF SDB DepAmt Inv
          SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt Sav*B_DDABal
          SavBal*SDB SavBal*DDA AtmAmt*DepAmt B_DDABAL*ATMAmt SavBal*IRA
          SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal
          CD*MM CD*Sav Sav*CC,var1=SavBal,var2=B_DDABal,mean_inputs=SavBal Dep
          DDA CD Sav CC ATM MM branch_swoe IRA B_DDABal ATMAmt ILS NSF SDB
          DepAmt Inv);
```

Demo Using the Best-Subsets Selection Method

```
data work.train_imputed_swoe_bins;
  set work.train_imputed_swoe_bins;
  resr=(res='R');
  resu=(res='U');
run;

/* Run best subsets */
title1 "Models Selected by Best Subsets Selection";
proc logistic data=work.train_imputed_swoe_bins;
   model ins(event='1')=&screened resr resu SavBal*B_DDABal MM*B_DDABal
                 branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
                 SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt
                 SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM
                 IRA*B_DDABal CD*MM MM*IRABal CD*Sav B_DDABal*CashBk Sav*C(
                 / selection=score best=1;
```

Number of Observations Read	21512
Number of Observations Used	21512

Response Profile		
Ordered Value	Ins	Total Frequency
1	0	14061
2	1	7451

Probability modeled is Ins=1.

Regression Models Selected by Score Criterion		
Number of Variables	Score Chi-Square	Variables Included in Model
1	1872.2915	B_DDABal
2	2490.9205	CD B_DDABal
3	2935.4526	SavBal CD B_DDABal
4	3240.0673	SavBal CD B_DDABal SavBal*B_DDABal
5	3501.6203	SavBal CD branch_swoe B_DDABal SavBal*B_DDABal
6	3731.8960	SavBal CD Sav MM branch_swoe B_DDABal
7	3952.2174	SavBal CD Sav MM branch_swoe B_DDABal SavBal*B_DDABal
8	4039.4342	SavBal Dep CD Sav MM branch_swoe B_DDABal SavBal*B_DDABal
9	4086.6501	SavBal Dep CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal
10	4129.3415	SavBal CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal DDA*ATMAmt Dep*ATM
11	4158.9269	SavBal CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal MM*B_DDABal DDA*ATMAmt Dep*ATM

10	4129.3415	SavBal CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal DDA*ATMAmt Dep*ATM

Demo Using Fit Statistics to Select a Model

```
/* The fitstat macro generates model fit statistics for the
   models selected in the all subsets selection. The macro
   variable IM is set equal to the variable names in the
   model_indx model while the macro variable IC is set
   equal to the number of variables in the model_indx model. */

%macro fitstat(data=,target=,event=,inputs=,best=,priorevent=);

ods select none;
ods output bestsubsets=work.score;

proc logistic data=&data namelen=50;
   model &target(event="&event")=&inputs / selection=score best=&best;
run;
```

```
proc sql noprint;
  select variablesinmodel into :inputs1 -
  from work.score;

  select NumberOfVariables into :ic1 -
  from work.score;
quit;

%let lastindx=&SQLOBS;

%do model_indx=1 %to &lastindx;

%let im=&&inputs&model_indx;
%let ic=&&ic&model_indx;
```

```
ods output scorefitstat=work.stat&ic;
proc logistic data=&data namelen=50;
  model &target(event="&event")=&im;
  score data=&data out=work.scored fitstat
        priorevent=&priorevent;
run;
```

```
proc datasets
   library=work
   nodetails
   nolist;
   delete scored;
run;
quit;
```

```
%end;

/* The data sets with the model fit statistics are
   concatenated and sorted by BIC. */

data work.modelfit;
   set work.stat1 - work.stat&lastindx;
   model=_n_;
run;

%mend fitstat;
```

```
%fitstat(data=train_imputed_swoe_bins,
```

```
=train_imputed_swoe_bins,target=ins,event=1,inputs=&screened resr resu
 SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
 SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt SavBal*IRA
 SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal CD*MM
 MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC,best=1,priorevent=0.02);
```

```
proc sort data=work.modelfit;
   by bic;
run;
```

```
title1 "Fit Statistics from Models selected from Best-Subsets";
ods select all;
proc print data=work.modelfit;
   var model auc aic bic misclass adjrsquare brierscore;
run;
```

Fit Statistics from Models selected from Best-Subsets

Obs	model	AUC	AIC	BIC	MisClass	AdjRSquare	BrierScore
1	35	0.788365	50279.66	50566.81	0.3398	0.354277	0.30715
2	30	0.787561	50321.71	50568.98	0.3400	0.352545	0.307317
3	33	0.788091	50299.66	50570.86	0.3399	0.353479	0.307291
4	36	0.788464	50275.9	50571.03	0.3398	0.354468	0.307156
5	28	0.787377	50340.06	50571.38	0.3399	0.3518	0.307333
6	27	0.787237	50348.13	50571.47	0.3396	0.351464	0.307294
7	37	0.788533	50271.74	50574.84	0.3397	0.354673	0.307166
8	34	0.788171	50295.93	50575.11	0.3399	0.353669	0.307297
9	32	0.788012	50313.5	50576.72	0.3399	0.352952	0.307405
10	26	0.787156	50364.17	50579.53	0.3397	0.350862	0.307379
11	39	0.788885	50261.01	50580.06	0.3398	0.355162	0.307148
12	38	0.78862	50270.52	50581.6	0.3398	0.35478	0.307158

```
%global selected;
proc sql;
   select VariablesInModel into :selected
   from work.score
   where numberofvariables=35;
quit;
```

Fit Statistics from Models selected from Best-Subsets

Variables Included in Model
SavBal DDA CD Sav MM branch_swoe IRA IRABal B_DDABal ATMAmt ILS NSF SDB CCBal Inv SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt Sav*B_DDABal SavBal*SDB SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal CD*MM MM*IRABal CD*Sav Sav*CC

```
/* ====================================================== */
/* Lesson 3, Section 5: l3d8e.sas
   Demonstration: Using the Best-Subsets Selection Method
   [m643_5_s; derived from pmlr03d08.sas]            */
/* ====================================================== */

data work.train_imputed_swoe_bins;
  set work.train_imputed_swoe_bins;
  resr=(res='R');
  resu=(res='U');
run;

/* Run best subsets */
title1 "Models Selected by Best Subsets Selection";
proc logistic data=work.train_imputed_swoe_bins;
   model ins(event='1')=&screened resr resu SavBal*B_DDABal MM*B_DDABal
            branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
            SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt
            SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM
            IRA*B_DDABal CD*MM MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC
            / selection=score best=1;

run;
```

Number of Observations Read	21512
Number of Observations Used	21512

Response Profile		
Ordered Value	Ins	Total Frequency
1	0	14061
2	1	7451

Probability modeled is Ins=1.

Regression Models Selected by Score Criterion

Number of Variables	Score Chi-Square	Variables Included in Model
1	1872.2915	B_DDABal
2	2490.9205	CD B_DDABal
3	2935.4526	SavBal CD B_DDABal
4	3240.0673	SavBal CD B_DDABal SavBal*B_DDABal
5	3501.6203	SavBal CD branch_swoe B_DDABal SavBal*B_DDABal
6	3731.8960	SavBal CD Sav MM branch_swoe B_DDABal
7	3952.2174	SavBal CD Sav MM branch_swoe B_DDABal SavBal*B_DDABal
8	4039.4342	SavBal Dep CD Sav MM branch_swoe B_DDABal SavBal*B_DDABal
9	4086.6501	SavBal Dep CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal
10	4129.3415	SavBal CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal DDA*ATMAmt Dep*ATM
11	4158.9269	SavBal CD Sav MM branch_swoe IRA B_DDABal SavBal*B_DDABal MM*B_DDABal DDA*ATMAmt Dep*ATM
12	4190.0309	SavBal CD Sav MM branch_swoe IRA B_DDABal ATMAmt SavBal*B_DDABal SavBal*SDB B_DDABal*ATMAmt Dep*ATM
13	4221.5081	SavBal CD Sav MM branch_swoe IRA B_DDABal ATMAmt SavBal*B_DDABal branch_swoe*ATMAmt SavBal*SDB B_DDABal*ATMAmt Dep*ATM
14	4252.2092	SavBal CD Sav MM branch_swoe IRA B_DDABal ATMAmt SavBal*B_DDABal branch_swoe*ATMAmt B_DDABal*ATMAmt SavBal*CC Dep*ATM Sav*CC
15	4276.0193	SavBal CD Sav MM branch_swoe IRA B_DDABal ATMAmt Inv SavBal*B_DDABal branch_swoe*ATMAmt B_DDABal*ATMAmt SavBal*CC Dep*ATM Sav*CC

```
/* ================================================= */
/* Lesson 3, Section 5: l3d8f.sas
   Demonstration: Using Fit Statistics to Select a Model
   [m643_5_L; derived from pmlr03d08.sas]          */
/* ================================================= */

/* The fitstat macro generates model fit statistics for the
   models selected in the all subsets selection. The macro
   variable IM is set equal to the variable names in the
   model_indx model while the macro variable IC is set
   equal to the number of variables in the model_indx model. */

%macro fitstat(data=,target=,event=,inputs=,best=,priorevent=);

ods select none;
```

```
ods output bestsubsets=work.score;

proc logistic data=&data namelen=50;
  model &target(event="&event")=&inputs / selection=score best=&best;
run;

/* The names and number of variables are transferred to macro
  variables using PROC SQL. */

proc sql noprint;
 select variablesinmodel into :inputs1 -
 from work.score;

 select NumberOfVariables into :ic1 -
 from work.score;
quit;

%let lastindx=&SQLOBS;

%do model_indx=1 %to &lastindx;

%let im=&&inputs&model_indx;
%let ic=&&ic&model_indx;

ods output scorefitstat=work.stat&ic ;
proc logistic data=&data namelen=50;
 model &target(event="&event")=&im;
 score data=&data out=work.scored fitstat
     priorevent=&priorevent;
```

```
run;

proc datasets
  library=work
  nodetails
  nolist;
  delete scored;
run;
quit;

%end;

/* The data sets with the model fit statistics are
  concatenated and sorted by BIC. */

data work.modelfit;
  set work.stat1 - work.stat&lastindx;
  model=_n_;
run;

%mend fitstat;

%fitstat(data=train_imputed_swoe_bins,target=ins,event=1,inputs=&screened resr resu
         SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt B_DDABal*Sav SavBal*SDB
         SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt SavBal*IRA
         SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal CD*MM
         MM*IRABal CD*Sav B_DDABal*CashBk Sav*CC,best=1,priorevent=0.02);

proc sort data=work.modelfit;
```

```
  by bic;
run;

title1 "Fit Statistics from Models selected from Best-Subsets";
ods select all;
proc print data=work.modelfit;
  var model auc aic bic misclass adjrsquare brierscore;
run;

%global selected;
proc sql;
  select VariablesInModel into :selected
  from work.score
  where numberofvariables=35;
quit;
```

Fit Statistics from Models selected from Best-Subsets

Obs	model	AUC	AIC	BIC	MisClass	AdjRSquare	BrierScore
1	35	0.788365	50279.66	50566.81	0.3398	0.354277	0.30715
2	30	0.787561	50321.71	50568.98	0.3400	0.352545	0.307317
3	33	0.788091	50299.66	50570.86	0.3399	0.353479	0.307291
4	36	0.788464	50275.9	50571.03	0.3398	0.354468	0.307156
5	28	0.787377	50340.06	50571.38	0.3399	0.3518	0.307333
6	27	0.787237	50348.13	50571.47	0.3396	0.351464	0.307294
7	37	0.788533	50271.74	50574.84	0.3397	0.354673	0.307166
8	34	0.788171	50295.93	50575.11	0.3399	0.353669	0.307297
9	32	0.788012	50313.5	50576.72	0.3399	0.352952	0.307405
10	26	0.787156	50364.17	50579.53	0.3397	0.350862	0.307379
11	39	0.788885	50261.01	50580.06	0.3398	0.355162	0.307148
12	38	0.78862	50270.52	50581.6	0.3398	0.35478	0.307158
13	29	0.78739	50342.95	50582.24	0.3401	0.35177	0.307446
14	25	0.786873	50377.32	50584.7	0.3397	0.350356	0.307381
15	41	0.78915	50251	50586.01	0.3397	0.355627	0.307095
16	40	0.788968	50259.86	50586.89	0.3398	0.355267	0.307139
17	31	0.787825	50334.21	50589.45	0.3401	0.352195	0.307527
18	24	0.785995	50391.72	50591.13	0.3396	0.349807	0.307081
19	42	0.789184	50251.19	50594.17	0.3397	0.355687	0.307086
20	23	0.785837	50409.67	50601.1	0.3398	0.34914	0.307214
21	43	0.789204	50252.82	50603.78	0.3397	0.3557	0.307084
22	44	0.78921	50254.79	50613.73	0.3397	0.355701	0.307088
23	45	0.789198	50256.66	50623.57	0.3398	0.355705	0.307087
24	22	0.785666	50442.77	50626.22	0.3399	0.347964	0.3074
25	46	0.789209	50258.16	50633.05	0.3397	0.355721	0.307084

Fit Statistics from Models selected from Best-Subsets

Variables Included in Model
SavBal DDA CD Sav MM branch_swoe IRA IRABal B_DDABal ATMAmt ILS NSF SDB CCBal Inv SavBal*B_DDABal MM*B_DDABal branch_swoe*ATMAmt Sav*B_DDABal SavBal*SDB SavBal*DDA ATMAmt*DepAmt B_DDABal*ATMAmt SavBal*ATMAmt SavBal*IRA SavBal*MM SavBal*CC Sav*NSF DDA*ATMAmt Dep*ATM IRA*B_DDABal CD*MM MM*IRABal CD*Sav Sav*CC

/* Practice: l3p9.sas step 1 */

%global ex_selected;

```
%macro fitstat(data=,target=,event=,inputs=,best=,priorevent=);

ods select none;
ods output bestsubsets=work.score;

proc logistic data=&data namelen=50;
  model &target(event="&event")=&inputs /
      selection=score best=&best;
run;

proc sql noprint;
 select variablesinmodel into :inputs1 -
 from work.score;
 select NumberOfVariables into :ic1 -
 from work.score;
quit;

%let lastindx=&SQLOBS;

%do model_indx=1 %to &lastindx;

%let im=&&inputs&model_indx;
%let ic=&&ic&model_indx;

ods output scorefitstat=work.stat&ic ;
proc logistic data=&data namelen=50;
 model &target(event="&event")=&im;
 score data=&data out=work.scored fitstat
     priorevent=&priorevent;
```

```
run;

proc datasets
  library=work
  nodetails
  nolist;
  delete scored;
run;
quit;

%end;

data work.modelfit;
  set work.stat1 - work.stat&lastindx;
  model=_n_;
run;

%mend fitstat;
/* Solution for l3p9 */

/* step 2 */

%global ex_selected;

%macro fitstat(data=,target=,event=,inputs=,best=,priorevent=);

ods select none;
ods output bestsubsets=work.score;
```

```
proc logistic data=&data namelen=50;
  model &target(event="&event")=&inputs /
      selection=score best=&best;
run;

proc sql noprint;
 select variablesinmodel into :inputs1 -
 from work.score;
 select NumberOfVariables into :ic1 -
 from work.score;
quit;

%let lastindx=&SQLOBS;

%do model_indx=1 %to &lastindx;

%let im=&&inputs&model_indx;
%let ic=&&ic&model_indx;

ods output scorefitstat=work.stat&ic ;
proc logistic data=&data namelen=50;
 model &target(event="&event")=&im;
 score data=&data out=work.scored fitstat
      priorevent=&priorevent;
run;

proc datasets
  library=work
  nodetails
```

```
  nolist;
  delete scored;
run;
quit;

%end;

data work.modelfit;
  set work.stat1 - work.stat&lastindx;
  model=_n_;
run;

%mend fitstat;



/* step 3 */

%fitstat(data=pmlr.pva_train_imputed_swoe,target=target_b,event=1,
      inputs=&ex_screened LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT
      LIFETIME_AVG_GIFT_AMT*RECENT_STAR_STATUS
      LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT,best=1,
      priorevent=0.05);

proc sort data=work.modelfit;
  by bic;
run;

title1 "Fit Statistics from Models selected from Best-Subsets";
ods select all;
```

```
proc print data=work.modelfit;
  var model auc aic bic misclass adjrsquare brierscore;
run;
title1;


/* step 4 */

proc sql;
  select VariablesInModel into :ex_selected
  from work.score
  where numberofvariables=9;
quit;
```

Fit Statistics from Models selected from Best-Subsets

Obs	model	AUC	AIC	BIC	MisClass	AdjRSquare	BrierScore
1	9	0.631652	14779.62	14851.4	0.2498	0.063068	0.222593
2	10	0.632213	14774.72	14853.68	0.2498	0.063921	0.222532
3	8	0.630349	14789.23	14853.84	0.2499	0.061631	0.222709
4	11	0.632442	14771.15	14857.29	0.2498	0.064609	0.222463
5	12	0.632987	14764.68	14858.01	0.2497	0.065654	0.222331
6	7	0.628417	14802.91	14860.34	0.2499	0.059688	0.222876
7	13	0.633555	14760.16	14860.66	0.2497	0.066458	0.222274
8	6	0.626502	14814.34	14864.59	0.2499	0.058021	0.223026
9	14	0.633561	14759.11	14866.79	0.2497	0.066835	0.222234
10	5	0.625815	14826.13	14869.2	0.2500	0.056308	0.223329
11	15	0.633848	14758.26	14873.11	0.2497	0.067187	0.222202
12	4	0.622972	14844.22	14880.11	0.2500	0.053807	0.223532
13	16	0.634088	14758.3	14880.33	0.2497	0.067428	0.222179
14	17	0.63403	14758.47	14887.69	0.2497	0.067653	0.222161
15	18	0.634047	14759.89	14896.28	0.2497	0.067725	0.222152
16	19	0.634239	14761.15	14904.72	0.2497	0.067816	0.22215
17	3	0.619008	14879.2	14907.92	0.2500	0.049189	0.223944
18	20	0.634277	14762.5	14913.25	0.2497	0.067897	0.22214
19	21	0.634355	14763.95	14921.88	0.2497	0.067964	0.222137
20	22	0.63435	14765.75	14930.86	0.2497	0.067988	0.222135
21	23	0.634421	14767.34	14939.63	0.2497	0.068039	0.222132
22	24	0.634453	14769.14	14948.61	0.2497	0.068063	0.222131
23	25	0.634478	14770.93	14957.57	0.2497	0.06809	0.222128
24	2	0.607981	14937.08	14958.62	0.2500	0.041674	0.224492
25	26	0.634514	14772.88	14966.7	0.2497	0.068096	0.222126
26	27	0.634635	14774.81	14975.81	0.2497	0.068104	0.222149
27	28	0.63464	14776.78	14984.96	0.2497	0.068107	0.222149
28	29	0.63465	14778.78	14994.14	0.2497	0.068108	0.222149
29	1	0.586151	15028.27	15042.63	0.2500	0.029887	0.225337

Variables Included in Model
LIFETIME_GIFT_COUNT LAST_GIFT_AMT MEDIAN_HOME_VALUE FREQUENCY_STATUS_97NK cluster_swoe PEP_STAR INCOME_GROUP LAST_GIFT_AMT*LIFETIME_AVG_GIFT_AMT LIFETIME_GIFT_COUNT*MONTHS_SINCE_LAST_GIFT

```
data work.pva(drop=CONTROL_NUMBER MONTHS_SINCE_LAST_PROM_RESP
          FILE_AVG_GIFT FILE_CARD_GIFT);
  set pmlr.pva_raw_data;
run;

title "Models Selected by Backward Selection";
proc logistic data=work.pva;
  model target_b(event='1')=recent_response_prop
      recent_avg_gift_amt lifetime_card_prom
      per_capita_income pct_male_veterans
      lifetime_gift_range pct_male_military
      / selection=backward;
run;
title;
```