PhUSE 2015

Paper AD05

Statistical Analysis Reporting Tool (StART) From Concept to Table Production

David Hartman, UCB Biosciences, Raleigh NC, USA

ABSTRACT

The idea of producing of tables for statistical analysis has been around for decades. Along the way there have been many attempts at writing programs that will produce tables in as short of amount of time as possible with the fewest amounts of resources. The Statistical Analysis Reporting Tool (StART) is a program written in SAS® that has achieved this goal. It is totally data independent, and with its modular design StART can produce an almost endless variety of publication ready displays. In order to understand how StART is able to perform this task it is important to understand the concept of table production. For example, what exactly are the components of a table? What are the minimum requirements needed to produce a table? What are the various ways a table can be presented? This paper explains these concepts and illustrates how StART utilizes them to produce statistical displays.

INTRODUCTION

Table production can be a long and labor intensive process for the development and the validation programmer. Often times the programs that are written are very specific to a study, type of data, and layout of the table. It is a lot of work to both develop and maintain these programs. The programmers at UCB Biosciences have written a program called the "Statistical Analysis Reporting Tool" otherwise known as StART. It is intended to reduce the time and effort that goes into table production and thereby deliver them in a timelier manner.

In order to understand how to use StART you must first understand the concept of a table. Exactly what are the components of a table? What are the various ways the results of a statistical analysis can be displayed? This paper will provide answers to these questions and demonstrate by example how StART utilizes this information to produce a table as defined by the user.

THE CONCEPT

Table production in simple terms is the organizing of statistical results on discrete and continuous data in an easy to read format from which conclusions regarding the data can be made. The statistics performed on data is often either descriptive such as mean, minimum or maximum or the more complex modeling for such things as p-values and confidence intervals (see section on available statistics provided by StART).

THE COMPONENTS

A typical statistical table is made up of 3 primary components; 1) column header, 2) row text, and 3) main body.

Column Header

	Treatment 1	Treatment 2	Treatment 3
Systolic Blood Pressure (mmHg))			
Mean	101.7	103.2	102.7
Min	77	71	80
Max	123	125	127
Diastolic Blood Pressure (mmHg)			
Mean	62.0	62.6	62.8
Min	34	38	42
Max	79	81	86

Row Text

Main Body

The column header will typically contain the grouping of the output. The row text will typically contain the variable(s) on which the statistics will be performed as well as the statistics. The main body of a table is simply the intersection of the column header and row text.

In a typical table these components are defined by 3 sets of information 1) a classifying variable, 2) statistical variable, and 3) statistics.:

1. Classifying Variables (ex. armcd)
2. Statistic Variables (ex. paramcd)
3. Statistics (ex. min, mean, max)

1.

	Treatment 1	Treatment 2	Treatment 3
Systolic Blood Pressure (mmHg)			
Mean	101.7	103.2	102.7
Min	77	71	80
Max	123	125	127
Diastolic Blood Pressure (mmHg)			
Mean	62.0	62.6	62.8
Min	34	38	42
Max	79	81	86

2.

3.

The combination of one or more classifying variables, statistical variables, and statistics provides a wide variety of displays.

THE LAYOUT

By defining the column and row dimensions a great many different types of tables can be created. Through the remainder of this paper you will see examples of the different table displays and how they are defined with StART.

START CALL TO COL AND ROW PARAMETERS

The 2 primary StART parameters (and associated key words) that control what analysis is to be performed and how it is to be displayed are COL for the column dimension and ROW for the row dimension.

col<row>= **classvar<statvar>** *variable_name1 variable_name(n)*
*<# **label** variable_name1="label" / variable_name(n)="label" >*
*<# **format** variable_name1=format_name / variable_name(n)=format_name >*
*<# **where** variable= >*
< **neq** = variable_name1 variable_name(n) >*
< **totals** variable_name1 variable_name(n)*
*# **label** variable1="label" / variablen="label" >*
*# **value** variable1=first<last> / variablen=first<last>*
< **group** variable1=(value1,value2,value(n)=group_value) # **label** variable1="label">*
< **subclass** variable1 # **label** variable1="label" # where >*
< **stats** stats1 stats(n) # **label** stats1="label" / statsn="label"*
*# **format** stats1="format"*
*# **denom**=variable to be used as denominator in calculation of %*
*# **censor**=censor variable(n)*
*# **effect**=list of variables*
*# **strata**=list of variables*
*# **basevar**=base variable*
*# **baseval**=base value*
*# **pboval**=placebo value*
*# **sitevar**=site ID variable*

These sets of parameters are used in the defining of statistical models.

The following table layouts illustrate the flexibility of StART and demonstrate how the COL and ROW parameters and their associated key words are used to create these tables. For an explanation of the delimiters see section on START DELIMITERS below.

MINIMUM REQUIREMENT TO BUILDING A TABLE

The minimum requirement for a table is to have at least one statistical variable and one statistic. The statistical variable has to be defined in either the row or column dimension and the statistic must be defined in the dimension opposite of the statistical variable.

	Gender	
	Male	Female
N	368	396

Column: statistic variable
Row: statistic

Call to StART:

COL=statvar sex ,
ROW=stats n ,

By default StART displays the label of all classifying and statistical variables as is seen in this example.

ADDING A CLASSIFYING VARIABLE

	Treatment		
	Treatment 1	Treatment 2	Placebo
Gender n(%)			
Male	102(40.3)	133(53.2)	133(51.0)
Female	151(59.7)	117(46.8)	128(49.0)

Column: classifying variable
Row: statistic variable, statistics

Call to StART:

COL=classvar armcd,
ROW=statvar sexn # label sexn=Gender n(%) * stats n(pct) # denom=armcd ,

Refer to StART Delimiter section for explanation of delimiters.

TABLE TRANSPOSED

Simply by changing COL to ROW and ROW to COL virtually any table can be transposed.

	Gender	
	Male	Female
	n(%)	n(%)
Treatment		
Treatment 1	102(40.3)	151(59.7)
Treatment 2	133(53.2)	117(46.8)
Placebo	133(51.0)	128(49.0)

Column: statistic variable, statistics
Row: classifying variable

Call to StART:

COL=statvar sexn * stats n(pct) # denom=armcd ,
ROW=classvar armcd,

CLASSIFYING VARIABLE AND STATISTIC VARIABLE IN ONE DIMENSION, STATISTICS IN THE OTHER DIMENSION

	n(%)
Treatment	
Treatment 1	
Gender	
Male	102(40.3)
Female	151(59.7)
Treatment 2	
Gender	
Male	133(53.2)
Female	117(46.8)
Placebo	
Gender	
Male	133(51.0)
Female	128(49.0)

Column: statistics
Row: classifying variable and statistic variable

Call to StART:

COL= stats n(pct) # denom=armcd ,
ROW=classvar armcd | statvar sexn,

MULTIPLE STATISTIC VARIABLES IN ONE DIMENSION, STATISTICS IN THE OTHER DIMENSION

	n(pct)
Gender	
Male	102(40.3)
Female	151(59.7)
Race	
AMERICAN INDIAN OR ALASKA NATIVE	8(3.2)
ASIAN	32(12.6)
BLACK OR AFRICAN AMERICAN	8(3.2)
OTHER	21(8.3)
WHITE	182(71.9)
XMissing	2(0.8)

Column: statistics
Row: 2 statistic variables

Call to StART:

COL= stats n(pct) # denom=armcd ,
ROW=statvar sexn race,

A table like this is only possible if the 2 statistic variables are providing the same set of statistics.

MULTIPLE CLASSIFYING VARIABLES IN ONE DIMENSION, STATISTIC VARIABLE AND STATISTICS IN THE OTHER DIMENSION

	Gender			
	Male		Female	
	N	(%)	N	(%)
Treatment				
Treatment 1				
Age Group 1 (N)				
<18	3	(50.0)	3	(50.0)
18-<65	92	(39.0)	144	(61.0)
>=65	7	(63.6)	4	(36.4)
Treatment 2				
Age Group 1 (N)				
<18	7	(100)	0	
18-<65	123	(51.9)	114	(48.1)
>=65	3	(50.0)	3	(50.0)
Placebo				
Age Group 1 (N)				
<18	4	(66.7)	2	(33.3)
18-<65	127	(50.8)	123	(49.2)
>=65	2	(40.0)	3	(60.0)

Column: statistic variable, statistics
Row: 2 classifying variables

Call to StART:

COL= statvar sexn * stats n(pct) # denom=armcd –agegrp1n,
ROW= classvar armcd agegrp1n,

ONE CLASSIFYING VARIABLE IN ONE DIMENSION, MULTIPLE STATISTIC VARIABLES AND STATISTICS IN THE OTHER DIMENSION

	Treatment		
	Treatment 1	Treatment 2	Placebo
Age (Years)			
Nobs	253	250	261
Mean(STD)	39.1(13.4)	39.8(12.8)	39.8(12.5)
Median	39.0	40.0	39.0
Min-Max	16.0-80.0	16.0-73.0	16.0-77.0
Age Category (Years) n(%)			
<18	6(2.4)	7(2.8)	6(2.3)
18-<65	236(93.3)	237(94.8)	250(95.8)
>=65	11(4.3)	6(2.4)	5(1.9)

Column: classifying variable
Row: 2 statistic variables, statistics

Call to StART:

COL= classvar armcd,
ROW=statvar age * stats n(pct) # denom=armcd |
statvar agegr1n * stats nobs mean(std) median min-max,

TOTALS IN ONE DIMENSION

	Treatment 1 N=253	Treatment 2 N=250	Total Treatment N=503	Placebo N=261	Total N=764
Age (Years)					
Nobs	253	250	503	261	764
Mean(STD)	39.1(13.4)	39.8(12.8)	39.4(13.1)	39.8(12.5)	39.5(12.9)
Median	39.0	40.0	39.0	39.0	39.0
Min-Max	16.0-80.0	16.0-73.0	16.0-80.0	16.0-77.0	16.0-80.0
Age Category (Years) n(%)					
<18	6(2.4)	7(2.8)	13(2.6)	6(2.3)	19(2.5)
18-<65	236(93.3)	237(94.8)	473(94.0)	250(95.8)	723(94.6)
>=65	11(4.3)	6(2.4)	17(3.4)	5(1.9)	22(2.9)

Column: classifying variable, total of a group of columns and total of all columns
Row: 2 statistic variables, statistics

Call to StART:

```
COL=classvar armcd # label armcd=nolabel
                * neq=armcd
                * group armcd=(TRT1,TRT2=TRTZZZ) # label armcd=Total Treatment
                * totals armcd  # label armcd=Total,
ROW=statvar age * stats n(pct) # denom=armcd |
      statvar agegr1n * stats nobs mean(std) median min-max,
```

TOTALS IN TWO DIMENSIONS

	Treatment 1 N=253		Treatment 2 N=250		Placebo N=261		Total Subjects N=764	
	n(%)	#	n(%)	#	n(%)	#	n(%)	#
Total Patients with AE's	5(2.0)	8	7(2.8)	9	7(2.7)	7	19(2.5)	24
Blood and lymphatic system disorders	5(2.0)	8	7(2.8)	9	7(2.7)	7	19(2.5)	24
Anaemia	1(0.4)	1	4(1.6)	4	2(0.8)	2	7(0.9)	7
Eosinophilia	0(0.0)	0	3(1.2)	4	3(1.1)	3	6(0.8)	7
Iron deficiency anaemia	0(0.0)	0	0(0.0)	0	1(0.4)	1	1(0.1)	1
Leukopenia	2(0.8)	2	0(0.0)	0	0(0.0)	0	2(0.3)	2
Neutropenia	3(1.2)	3	0(0.0)	0	1(0.4)	1	4(0.5)	4
Thrombocytopenia	0(0.0)	0	1(0.4)	1	0(0.0)	0	1(0.1)	1

Column: classifying variable, total of all columns
Row: classifying variable, statistic variable, statistics, total of all rows under a classifying variable and total of all rows

Call to StART:

```
COL=classvar armcd # label armcd=nolabel
            * stats n(pct) nobs # denom=ad.adsl.armcd
            * neq=ad.adsl.armcd
            * totals armcd # label armcd=Total Subjects,
ROW=classvar aebodsys # label aebodsys=nolabel
            * totals grndtot # label grndtot=Total Patients with AE%nrbquote(')s |
      statvar aedecod  # label aedecod=nolabel
          * totals aedecod # label aedecod=nolabel,
```

BY VARIABLES, PAGE AND COLUMN

Intensity: MILD

MedDRA (Version 9.0) System Organ Class Preferred Term	Drug Relation ship	Treatment 1 N=45 n	(%)	[#]	Treatment 2 N=24 n	(%)	[#]
Total Patients with AE's	Any	0			1	(4.2)	[1]
	Related	0			0		
	Not Related	0			1	(4.2)	[1]
CARDIAC DISORDERS	Any	0			1	(4.2)	[1]
	Related	0			0		
	Not Related	0			1	(4.2)	[1]
Sinus arrhythmia	Any	0			1	(4.2)	[1]
	Related	0			0		
	Not Related	0			1	(4.2)	[1]

By variables: page and column
Column: classifying variable, total of all columns
Row: classifying variable, statistic variable, statistics, total of all rows under a classifying variable and total of all rows

Call to StART:

BYVARS= aesitsl aereln # label aesitsl=Intensity: / aereln=Drug~Relation~ship
* position aesitsl=page / aereln=col
COL=classvar armcd # label armcd=nolabel
* stats n(pct) # denom=ad.adsl.armcd
* neq=ad.adsl.armcd ,
ROW=classvar aebodsys # label aebodsys=nolabel
* totals grndtot # label grndtot=Total Patients with AE%nrbquote(')s |
statvar aedecod # label aedecod=nolabel
* totals aedecod # label aedecod=nolabel,

SUBJECT LISTING, STATISTIC VARIABLE, AND STATISTICS

Subject Number	Relative time post infusion start (hours) 0	1	1.5	3	5	11	24	48	72	96
00003	BLQ	3.3	3.6	3.73	3.3	2.92	2.75	2.06	1.84	1.57
00008	BLQ	2.96	3.01	2.93	2.81	2.61	2.27	1.84	1.74	1.48
00019	BLQ	1.79	2.01	1.93	1.88	1.71	1.51	1.2	1.09	0.902
00023	BLQ	3.26	3.18	3.1	2.87	2.7	2.28	1.94	1.67	1.48
N	4	4	4	4	4	4	4	4	4	4
GeoMean	0.000	2.748	2.885	2.844	2.660	2.436	2.153	1.724	1.554	1.327
95% CI LL	0.000	1.733	1.931	1.827	1.810	1.661	1.440	1.166	1.061	0.879
95% CI UL	0.000	4.358	4.310	4.427	3.909	3.571	3.220	2.548	2.275	2.004
geoCV(%)	0.00	29.60	25.63	28.37	24.56	24.39	25.70	24.95	24.31	26.33
Mean	0.000	2.828	2.950	2.923	2.715	2.485	2.203	1.760	1.585	1.358
SD	0.0000	0.7081	0.6739	0.7458	0.5979	0.5328	0.5131	0.3840	0.3373	0.3069
Median	0.000	3.110	3.095	3.015	2.840	2.655	2.275	1.890	1.705	1.480
Min	0.000	1.790	2.010	1.930	1.880	1.710	1.510	1.200	1.090	0.902
Max	0.000	3.300	3.600	3.730	3.300	2.920	2.750	2.060	1.840	1.570

Column: statistic variable, subclass variable
Row: subject variable, statistics

Call to StART:

COL=statvar aval # label aval=nolabel
* subclass avisitn
# label avisitn=Relative time post infusion start (hours)
# where avisitn lt **10,**
ROW=subjvar subjrnd # label subjrnd=nolabel
dispvar subjrnd=avalc |
stats n gmmean gmlclm gmuclm gmcv mean std median min max
# label n=n / std=SD
# format mean=**6.3** / std=**8.4** / median=**6.3** / min=**6.3** / max=**6.3** /
gmmean=**6.3** / gmlclm=**6.3** / gmuclm=**6.3** / gmcv=**5.2**

THE TOOL

StART is tool designed to produce tables in a data independent and free form way in order to minimize the time and effort that goes into table production. This tool can be completely independent and run on any platform or it can be tied into a platform having a standard folder structure. StART is made up of the following 3 primary modules:

1. **PARSER** – evaluates the information provided by the user and reassembles that information for the new 2 parameters.
2. **GENSTATS** – generates the statistics and produces the dataset that will eventually go into the table
3. **GENRTP** – receives the table mapping information from PARSER and the statistics generated by GENSTATS to produce the table.

As a program StART is written entirely in SAS®. This makes future enhancements and maintenance of this tool easy for anyone who is has medium to high level of experience as a SAS programmer and has experience in producing statistical displays. Approximately 75 percent of the code is written using the SAS Macro facility with all of the macro variables created being declared local. StART is made available to the users via compiled macro code that is stored in a macro catalog. This makes StART easy to copy into the individual study level folder as well as making the code secure (safe from any unauthorized modifications).

StART has several features that provide users total control over how a table is to be displayed. These features range from what data and statistics go into the table to how wide each individual column should be (providing the internal calculation is not adequate). Below is a list of some of the features currently available.

StART FEATURES

Feature	Description	Example
Modular design	StART is modular in its design which means that it is possible to go above and beyond the current (nearly 70) set of statistics already provided. Also, it is possible to customize the display beyond what is currently possible with this tool	
COL and ROW parameters interchangeable	The COL and ROW parameters are totally interchangeable. It is even possible to take a demographics table and rotate it simply by changing COL to ROW and ROW to COL	
Define labels for classifying and statistic variables	The labels for classifying and statistic variables can be defined in the call to StART. The labels can also be suppressed (turned off).	Define a new label: label armcd= Treatments Turn off a label: label armcd=nolabel
Define labels and formats for statistics	The labels and formats for statistics can be defined in the call to StART overriding the default labels and formats.	Stats n pct # label pct=% # format pct=4.1

Control column widths and number of lines on a page	Through the use of the UNIT,(in, cm, or %) column width parameters (TXTCOLW, STATCW, BYCOLW, and COLWDTH) users can override the column widths of the table defined by StART..	COLWDTH=col1=3/col2=2.75/col3=3.25
Processing of each statistical variable	Each statistical variable are processed individually. A single statistical variable can be processed in 2 or more ways (i.e. different subsets of the same variable) or, as in the case of a demographics table multiple different variables, both discrete and continuous can be processed. The order in which the variables are displayed depend on the order they are defined in either the COL or ROW parameter	
Subclass a statistic variable	Grouping an individual statistic variable.	STATVAR* stats nobs mean std median min max * subclass sexn # label sexn=Gender
Subsetting of data before generating statistics	Data can be subset in 3 different levels: 1. Input dataset 2. Statistic variable 3. Subclass (used to group a specific statistic variable) variable. This is often used in patient disposition tables	1. SUBSET=saffl eq 'Y' 2. STATVAR randfl # label randfl=Randomized # WHERE randfl=randfl in ('Y') 3. STATVAR termreas # label termreas=Primary reason for screen failure * SUBCLASS compstdy # label compstdy=Screen Failure # WHERE compstdy not in ('Y')
Subsetting of rows in a table	User can subset rows of a table based upon certain criteria. This is especially valuable when you want only those adverse events that exceed a certain threshold (typically 5%).	GEVAL=level=aesoc-aehlt-aedecod # stat=pct gt **5** # col=armcd in ('A','B','Z') # row=aereln ne **0**,
Sort table by specified column	The rows of a table can be sorted by whatever column desired. This is especially useful when sorting adverse events by descending frequency. In case of a tie then the adverse events are sorted alphabetically.	SORTBY=aessocl(order=descending, stat=nobs, col=col7)
Stats can be grouped into single cells or displayed separately	Stats can be displayed individually or grouped into a single cell using virtually any delimiter as defined by the user.	Examples of grouping two or more statistics into single cells: STATS min-max or Q1,Q2
Zero filling of table	Zero filling of a table can be done using the existing data and/or format catalog. The zero filling can also be done on a single variable or the full set of variables included in the table, both classifying and statistical variables. While the default zero filling value is 0, the user can fill the empty spaces with any numeric or text string including blanks.	ZEROFILL=_fmt_ / _data_ , ZEROVAL=pct=blank
Multiple output file types available	Ability to choose file type (i.e. .pdf, .rtf, or .lst)	
Table layout 2 ways	There is the option of outputting a table as landscape or portrait.	
Keeping dataset	The final dataset created just before the output of the table can be stored permanently.	KEEPDS=ad.t_demo,
Keeping resolved macro code	The resolved macro code created by StART can be stored. This code can be modified and or rerun when needed.	FILEPGM=run_t_demo.sas, PGMPATH=*path to program folder*

THE AVAILABLE STATISTICS

Statistic	Label	Default Format
CSS	CSS	8.1
CV	CV	8.3
DF	DF	8.0
LCLM	CLL-xx%	8.0
MAX	Max	8.1
MEAN	Mean	8.0
MEDIAN	Median	8.0
MIN	Min	8.1
MODE	Mode	8.0
MSIGN	M(sign)	8.0
N	N	8.0
NMISS	NMISS	8.3
NOBS	NOBS	8.1
NORMAL	W(normality)	8.4
PCT	Pct	8.4
PROBS	P(signed rank)	8.4
PROBM	P(sign test)	8.1
PROBN	P(normality)	8.1
PRT	P(t-test)	8.1
P1	P1%	8.1
P5	P5%	8.1
P10	P10%	8.1
P95	P95%	8.1
P99	P99%	8.0
QRANGE	P75%-P25%	8.3
RANGE	Range	8.1
SIGNRANK	S(signed rank)	8.1
SKEWNESS	Skewness	8.1
STD	STD	8.1
STDERR	SEM	8.1
SUM	Sum	8.1
T	t-statistic	8.1
UCLM	CLU-xx%	8.1
USS	USS	8.1
VAR	Var	8.1

All statistics are generated from PROC UNIVARIATE with the exception of N and PCT which are derived using PROC SQL (this enables frequency and percent to be performed on both character and numeric variables).

START DELIMITERS

The following are delimiters used in the ROW and COL parameters that separate the different levels of information provided by the user to define the layout of the table.

Delimiter	Description	Example
|	First level. Separates variables in a ROW or COL definition.	row=statvar age | statvar agegr1n
*	Second level. Separates tasks within first level.	row=statvar age * stats n(pct) |
#	Third level. Separates tasks within second level.	col=classvar armcd # label armcd=nolabel
/	Fourth level. Separates tasks within third level.	row=statvar age height # label age=Age (years) / height=Height (cm)

CONCLUSION

Statistical tables, as a concept, are simply that of columns and rows which when intersected gives the results of the statistics performed on the data. StART utilizes this concept through the use of 2 primary parameters, COL and ROW, which enables users to define the layout of a table in a wide variety of possibilities.

The examples given in the LAYOUT section of this paper illustrate how the COL and ROW parameters are interchangeable and the various combinations of CLASSVAR, STATVAR, and STATS and the data independence gives StART its flexibility. With a basic understanding of the concept and components of a table as well as a basic understand of SAS® the length of time needed to become proficient in the use of StART is minimal. In addition, due to its modular design StART is scalable enabling it to expand its capabilities in terms of the types of statistics as well as the type table layouts beyond what is currently available.

StART is a program written entirely in SAS® making it easy to perform future enhancements and maintenance of this tool. In addition, the use of this tool for producing tables significantly reduces the amount of code needed to be written by programmers which has the potential of reducing the turnaround time from the creating of the analysis datasets to the delivering of the tables for publications and regulatory authorities. Since this is a validated tool, the amount of resources needed for validation of tables is also significantly reduced because the programmer performing the validation can also use StART ensuring that the information that is provided by the table developer is producing the table as defined according to the specifications. Therefore, because of its flexibility, ease of maintenance, and ease of use, StART is a valuable tool in the creating of tables for CSRs, publications, submission packages, and any adhoc analysis.

ACKNOWLEDGMENTS

I wish to thank the following for their guidance and support through the development and validation of this tool:

Brian Mabe, Manager, Head Statistical Programming Group, US
Giuseppe Di Monaco, Senior Statistical Programmer

CONTACT INFORMATION

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

David Hartman
UCB Biosciences
P.O. Box 110167
Research Triangle Park, NC 27709 USA
Work Phone: +1 919-767-2597
Fax: +1 919-767-2573
Email: david.hartman@ucb.com