

Want to Conquer the Fear of Annotation?

Start Using the NOTE Statement

Madhuri Aswale, Inventive Health Clinical, Pune, Maharashtra (India)

ABSTRACT

The capabilities of #BYVAL in plotting graphs are exceptional for graphical display and can be made even more efficient and effective when used with the NOTE statement, Unlike TITLE and FOOTNOTE, NOTE is not frequently used in SAS code for creating and modifying graphs.

The use of both NOTE and #BYVAL cohesively allows the programmer to write better code and to embellish graphs with essential information.

The purpose of this paper is to explore the effective use of the NOTE statement to add text to maps, plots, charts, and text slides, resulting in better control of content, appearance, and placement of the text, including color, size, font, and alignment, in addition to allowing for underline and box drawing around text and straight lines on the output.

INTRODUCTION

This paper will discuss the uses of the NOTE statement with sample graphs, and real time examples.

Both beginners and experienced programmers will benefit from the examples by recognizing the wide application and increasing the understanding simple NOTE statement.

Most programmers increasing their understanding of the using annotations for enhancing their graphical outputs but not many have explored the easier way of using the combination of #BYVAL in not just the TITLE and FOOTNOTE statements but also the NOTE statement.

A NOTE statement in combination with #BYVAL is easier to program when compared to annotations especially when we have to make simple enhancements like: adding text to maps, drawing box around text, highlighting the values on graphs etc.

DECONSTRUCTING THE NOTE STATEMENT

NOTE statements are local, not global and they must be defined within a procedure or RUN-group with which they are used.

They remain in effect for the duration of the procedure which includes the NOTE statements in any of its RUN-groups, or until you end your SAS session.

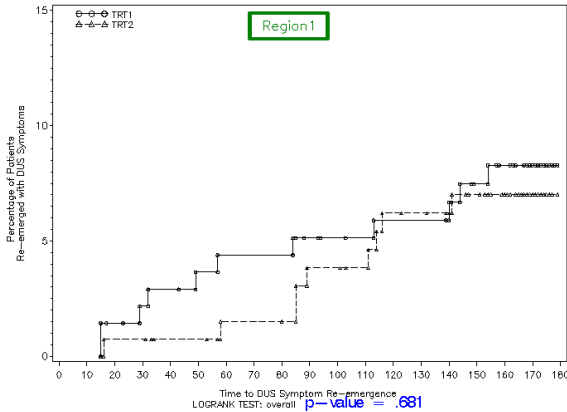
Through the following examples, we will display how we can achieve all of the above and try to understand how to use the NOTE statement programmatically.

Want to Conquer the Fear of Annotation? Start Using Note Statement.

Figure 1 - Kaplan-Meier graph by Region using combination of #BYVAL in a NOTE statement

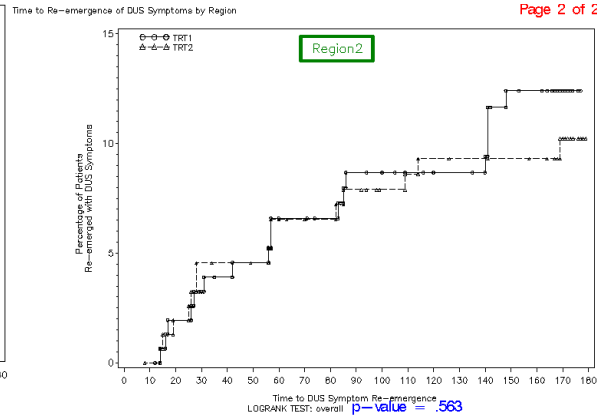
Time to Re-emergence of DUS Symptoms by Region

Page 1 of 2



Abbreviations: DUS = disease under study; TRT1 = treatment 1; TRT2 = treatment 2;
(a) Median time to DUS Symptom re-emergence are obtain using Kaplan-Meier estimate.

Page 2 of 2



Abbreviations: DUS = disease under study; TRT1 = treatment 1; TRT2 = treatment 2;
(a) Median time to DUS Symptom re-emergence are obtain using Kaplan-Meier estimate.

In the above example you can see how we have used the NOTE statement to generate all of the following:

- 1) **Titles** : Appearance, including color

Syntax:

```
* Generate Title Using NOTE and Title Statement *;
TITLE1 j=1 h=2 "Time to Re-emergence of DUS Symptoms by Region";
NOTE move=(90,98) h=3 j=r colour=red font=swiss '#BYVAL(newpage)';
```

- 2) **Page X of Y** : Placement of the text

Syntax:

```
* Generate Page X of Y Using NOTE Statement *;
NOTE j=c h=3 colour = green box=3 move = (50,90) " #BYVAL(region1) ";
```

- 3) **P-value** : Control of content,size, font, and alignment

Syntax:

```
* Generate P-vale Using NOTE Statement *;
NOTE j=c h=2 move = (40,16) "LOGRANK TEST: overall" font=swiss height=3
colour = blue "p-value = #BYVAL(pval) ";

* Generate Footnote Using NOTE Statement *;
NOTE move = (1,10)j=1 h=2 "Abbreviations: DUS = disease under study;
TRT1 = treatment 1; TRT2 = treatment 2; ";
NOTE move = (1,8) j=1 h=2 "(a) Median time to DUS Symptom re-emergence are
obtain using Kaplan-Meier estimate.";
```

NOTE statements can automatically include the BY variable name, BY variable values, or BY lines in the text they produce. To insert BY variable information into the text strings used by these statements, use the #BYVAR, #BYVAL, and #BYLINE substitution options.

For the above graph, including the values of variables region, page number and P-value in the BY statement allows us to use these values dynamically within a procedure in NOTE statement

The following values are going to change for each region, so first we have to calculate all these values using below algorithm

- 1) P-value - First find out p-value for each region using log rank test.
- 2) Page Number - Find out total number of regions and assign a unique number sequentially for each region
- 3) There should be one variable which contains the name of each distinct region.

Once we derive all of the real-time information about each region, we then display it dynamically in the graph using the code below:

```
*****Step 1: Find out total number of region from data *****;
PROC SQL NOPRINT;
SELECT count(distinct region1) into :textcn
FROM graph_data_n;
QUIT;

%let textcn = &textcn;

*****Step 2: Assign a unique number sequentially for each region *****;
PROC SORT data = graph_data_nnodupkey
OUT = graph_data_n2;
by region1;
RUN;

DATA graph_data_n2;
SET graph_data_n2;
cnt = _n_;
newpage = 'Page ' || compress(put(cnt,8.0)) || ' of ' || "&textcn";
KEEP region1 newpage;
RUN;

*****Step 3 : graph_data_n : Contains graph data to be plot
graph_data_n2 : Contains region name and page number in X of Y form
temp_b : Contains P-value of each region *****;

DATA graph_data_n;
MERGE graph_data_n graph_data_n2 temp_b;
by region1;
RUN;

***** Step 4 : Plot graph Using Proc Gplot *****;

PROC GPLOT data=graph_data_n ;
PLOT per*time =trt / legend=legend1 haxis=axis1 vaxis=axis2 ;
BY region1 newpage pval;

ods escapechar="^";

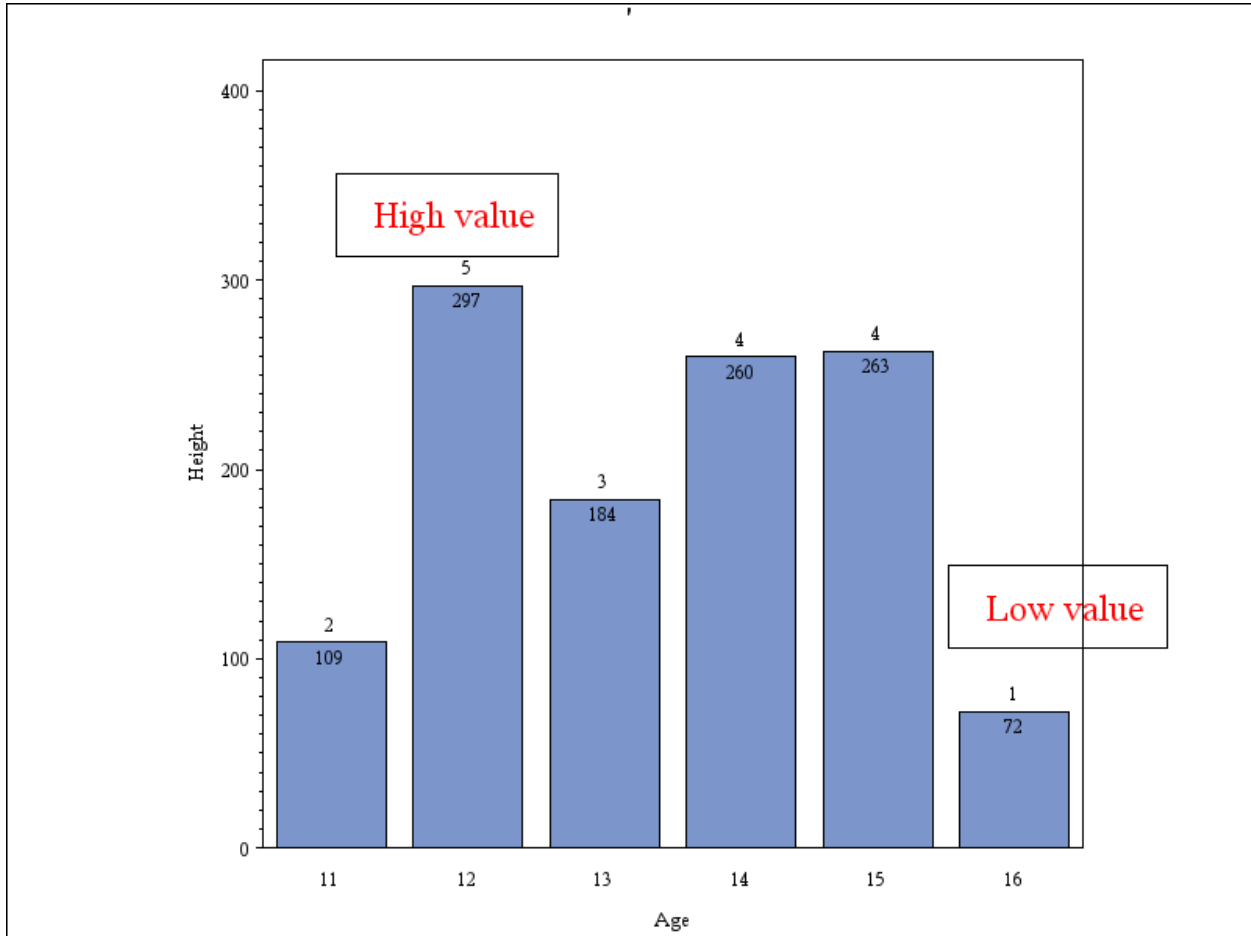
***** Step 5 : Generate Title Using NOTE and Title Statement *****;
TITLE1 j=1 h=2 "Time to Re-emergence of DUS Symptoms by Region";
NOTE move=(90,98) h=3 j=r colour=red font=swiss'#BYVAL(newpage)';

***** Step 6 : Generate Page X of Y Using NOTE Statement *****;
NOTE j=c h=3 colour = green box=3 move = (50,90) " #BYVAL(region1) ";

***** Step 7 : Generate P-value Using NOTE Statement *****;
NOTE j=c h=2 move = (40,16) LSPACE = 3 "LOGRANK TEST: overall" font=swiss
height=3 colour = blue
" p-value = #BYVAL(pval) ";

***** Step 8 : Generate Footnote Using NOTE Statement *****;
NOTE move = (1,10)j=1 h=2 "Abbreviations: DUS = disease under study; TRT1 =
treatment 1; TRT2 = treatment 2; ";
NOTE move = (1,8) j=1 h=2 "(a) Median time to DUS Symptom re-emergence are
obtain using Kaplan-Meier estimate.";
RUN;
QUIT;
```

Figure 2 - Box Plot by Age using NOTE Statement:



In the above example you can see how we have used the NOTE statement to highlight key values by drawing a box around the minimum and maximum values

```
ODS RTF file="&outfile/test.rtf" ;
PROC SORT data=sashelp.class out=class;
BY age;
RUN;
PROCGCHART data=class;
NOTE box = 1 move=(32,38) h=2 j=r colour=red " High value";
NOTE box =1 move=(88,17) h=2 j=r colour=red " Low value";
VBAR age / sumvar=height discrete width=10 maxis=axis1 raxis=axis2
inside=sum outside=freq;
AXIS1 label=('Age');
AXIS2 label=(angle=90 'Height')
ORDER = (0 to 400 by 100);
RUN;
ODS RTF CLOSE;
```

Want to Conquer the Fear of Annotation? Start Using Note Statement.

DISCOVER MORE ON NOTE STATEMENT

Syntax

NOTE<*text-arguments(s)*>;

text-argument(s) can be one or more of these in any order:

'text-string'

text-options

text-options can be one or more options from any or all of the following categories:

1) Appearance options

COLOR=*color*

FONT=*font*

HEIGHT=*text-height*<*units*>

2) Placement and spacing options

JUSTIFY=LEFT | CENTER | RIGHT

LSPACE=*line-space*<*units*>

MOVE=(*x,y*)<*units*>

3) Baseline angling and character rotation options

ANGLE=*degrees*

LANGLE=*degrees*

ROTATE=*degrees*

4) boxing, underlining, and line drawing options

BCOLOR=*background-color*

BLANK=YES

BOX=1...4

BSPACE=*box-space*<*units*>

DRAW=(*x,y...x-n,y-n*)<*units*>

You can use above options for further controlling appearance, placement, spacing, baseline angling, character rotation, boxing, underlining, and line drawing options to beautify your graph.

CONCLUSION

Through this paper we want to showcase the multifaceted personality of a simple NOTE Statement in combination with #BYVAL and also highlight its ease of use while programming when embellishing graphs for simple control of content, appearance, placement of text including color, size and alignment.

REFERENCES

1. SAS OnlineDoc®documentation SAS Institute Inc. Cary, NC.

CONTACT INFORMATION

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

Name: Madhuri Aswale

Enterprise: inVentiv Health Clinical

Address: Commerzone, Building No. 4, 6th Floor

City,StateZIP: Yerwada, Pune, India - 411006

Work Phone: 91 20 30569240

E-mail:madhuri.aswale@inventivhealth.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.