Paper 133-2013

# Creating Clark Error Grid with SAS/GRAPH®, and the SAS/GRAPH® Annotate Facility, and the SAS® Macro Applications

Yongyin Wang, Ph.D., Medtronic Diabetes, Northridge, CA

John Shin, Ph.D. and MBA, Medtronic Diabetes, Northridge, CA

## ABSTRACT

Clarke Error Grid Analysis has been widely used in the accuracy quantification of blood glucose values obtained from continuous glucose monitoring (CGM) sensor against reference values from meter or YSI instruments. A vivid graphic presentation of clinical accuracy of CGM sensor data is preferred by statisticians and reviewers of regulatory agencies. SAS/GRAPH® Annotate facility is a powerful tool for customizing, enhancing, or changing the features of graphic outputs. Clarke Error Grid breaks down a scatterplot of estimated glucose values versus reference values into five zones: A, B, C, D, and E. This presentation demonstrates how to use SAS/GRAPH®, the SAS/GRAPH® Annotate facility, and SAS® macro applications together to create such Error Grid for clinical accuracy determination of CGM data against meter or YSI glucose values

## INTRODUCTION

It is an ever changing area where different matrices were developed and tried for the sensor accuracy analyses. As CGM technology is not good enough to replace blood glucose meter data, a value of CGM needs to be characterized and presented as clinically acceptable data. The Clarke Error Grid Analysis (EGA) developed in 1987 to quantify the clinical accuracy of patient current blood glucose estimates versus their Meter blood glucose values, or Meter blood glucose estimates versus a reference value. Eventually, EGA has been accepted as one of the gold standards for the accuracy determination of blood glucose meters. Today, EGA has been widely used in the accuracy quantification of CGM sensor data against reference values from a meter or YSI instrument. The Clarke error grid breaks down a scatterplot of estimated blood glucose versus reference blood glucose values into five zones: A, B, C, D and E (Figure 1). The data points in Zones A and B represent accurate (within 20% of the reference) or acceptable glucose results (outside of 20% but would not lead to inappropriate treatment). The data points in Zone C may lead to unnecessary correction and a poor outcome. The data points in Zone D indicate a dangerous failure to detect (hypoglycemia or hyperglycemia) and treat, and those in Zone E would lead to a confusing treatment of hypoglycemia for hyperglycemia or vice versa. The more values in Zones A and B, the more accurate the device in term of clinical utility. Using PROC GPLOT alone in SAS/ GRAPH® is unable to produce the desired graph (Figure 1). The Annotate facility included within SAS/GRAPH® acts as a bridge between the procedure selected by the user and the user's desire to customize the graphics output. This presentation demonstrates how to use SAS/GRAPH®, the SAS/GRAPH® Annotate facility, and SAS® macro applications together to create such error grid for clinical accuracy determination of CGM data against meter or YSI glucose values, with a focus on the usage of Annotate facility. This paper is written for those with little knowledge of annotate but with basic knowledge of SAS/Graph® and SAS® macro.

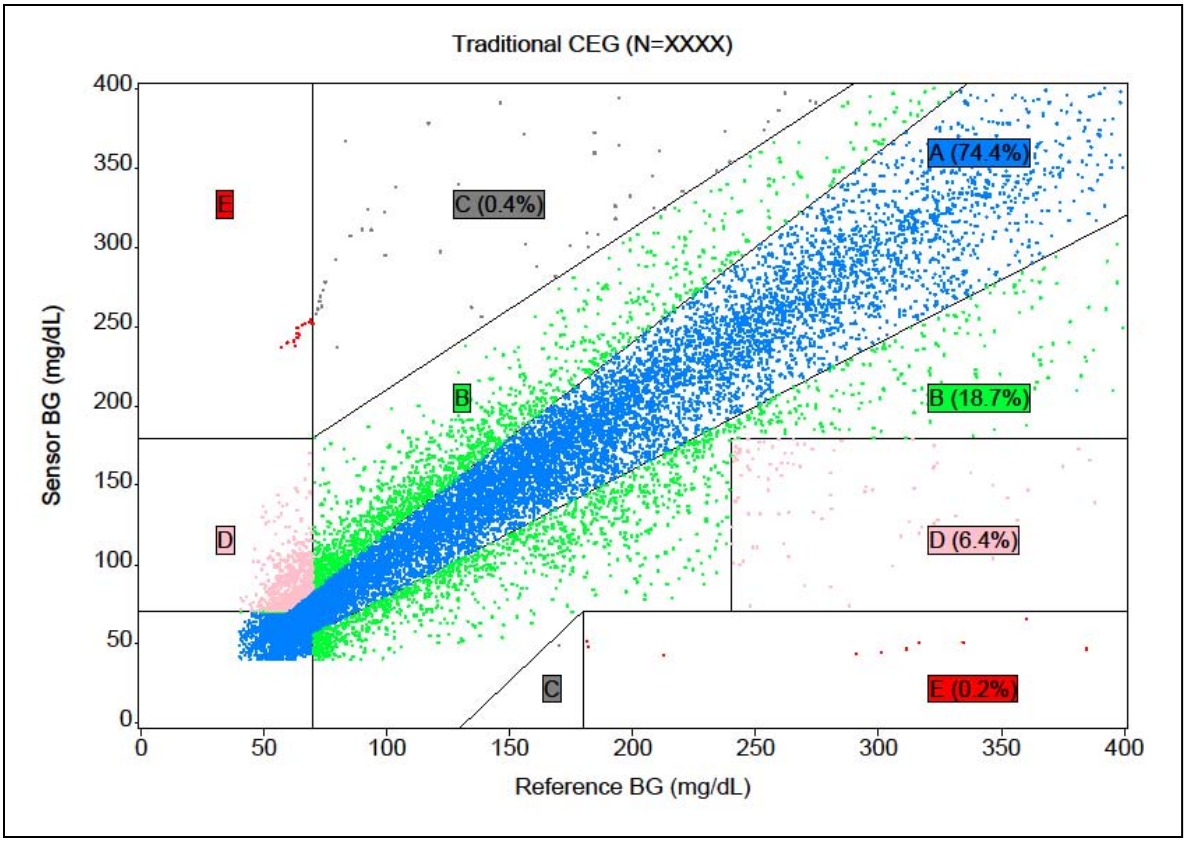



**Figure 1. The Sample Graph of Clarke Error Grid**

## THE STEPS TO CREATE CLARKE ERROR GRID

The creation of Clarke error grid consists of the following steps: 1) prepare the blood glucose data by calculating the bias and absolute relative errors (ARE); 2) assign the data points into five zones based on the calculated ARE and two predefined line equations; 3) apply PROC GPLOT to create scatterplot only, 4) draw lines and labels on the scatterplot using annotate facility, 5) wrap the codes in a macro so the same codes can be used for different studies.

### STEP 1: BLOOD GLUCOSE DATA PREPARATION

The data containing subject id, time stamp, sensor glucose value and reference glucose value obtained from Meter or YSI instruments were processed to calculate the bias and absolute relative errors (ARE, ARE= abs(Bias)/Ref*100).

**Table 1. The Sample Paired Data Points between Sensor Glucose and Reference Glucose Values**

| Subject ID | Time Stamp | Sensor Glucose Value | Reference Glucose value | Bias | ARE |
|---|---|---|---|---|---|
| 001001 | 30NOV2011:07:23:00 | 123 | 113 | 10 | 8.8 |
| 001001 | 30NOV2011:07:38:00 | 111 | 109 | 2 | 1.8 |
| 001001 | 30NOV2011:07:53:00 | 96 | 97.5 | -1.5 | 1.5 |
| 001001 | 30NOV2011:08:08:00 | 103 | 90.1 | 12.9 | 14.3 |
| 001001 | 30NOV2011:08:18:00 | 94 | 86.2 | 7.8 | 9 |

| | | | | | |
|---|---|---|---|---|---|
| 001001 | 30NOV2011:08:33:00 | 79 | 79.35 | -0.3 | 0.4 |
| 001001 | 30NOV2011:08:48:00 | 70 | 75.95 | -6 | 7.8 |
| 001001 | 30NOV2011:08:53:00 | 67 | 71.85 | -4.9 | 6.8 |
| 001001 | 30NOV2011:08:58:00 | 65 | 72.25 | -7.3 | 10 |
| 001001 | 30NOV2011:09:03:00 | 62 | 68.55 | -6.6 | 9.6 |
| 001001 | 30NOV2011:09:08:00 | 60 | 66.75 | -6.8 | 10.1 |
| 001001 | 30NOV2011:09:13:00 | 58 | 65.7 | -7.7 | 11.7 |
| 001001 | 30NOV2011:09:18:00 | 56 | 64.95 | -9 | 13.8 |
| 001001 | 30NOV2011:09:23:00 | 55 | 65.55 | -10.6 | 16.1 |
| 001001 | 30NOV2011:09:28:00 | 55 | 61.35 | -6.4 | 10.4 |
| 001001 | 30NOV2011:09:33:00 | 55 | 59.5 | -4.5 | 7.6 |
| 001001 | 30NOV2011:09:38:00 | 55 | 60.7 | -5.7 | 9.4 |
| 001001 | 30NOV2011:09:43:00 | 56 | 57.3 | -1.3 | 2.3 |
| 001001 | 30NOV2011:09:48:00 | 56 | 56.4 | -0.4 | 0.7 |
| 001001 | 30NOV2011:09:53:00 | 52 | 57.2 | -5.2 | 9.1 |
| 001001 | 30NOV2011:09:58:00 | 49 | 56.95 | -8 | 14 |
| 001001 | 30NOV2011:10:03:00 | 48 | 53.8 | -5.8 | 10.8 |
| 001001 | 30NOV2011:10:08:00 | 47 | 52.9 | -5.9 | 11.2 |
| 001001 | 30NOV2011:10:13:00 | 47 | 52.75 | -5.8 | 10.9 |
| 001001 | 30NOV2011:10:18:00 | 47 | 56.1 | -9.1 | 16.2 |
| 001001 | 30NOV2011:10:23:00 | 49 | 62.45 | -13.5 | 21.5 |

**STEP 2: ASSIGEMNT OF BLOOD GLUCOSE DATA POINTS INTO 5 DIFFERNET ERROR GRID ZONES**

After calculating the ARE, the glucose data points will be assigned into 5 categories, i.e. zones A, B, C, D and E, based on the ARE and two linear equations (see code below).

```
Data plot_ds;
    length zone $2.;
    set &ds1;

    line_eq1=(7/5)*(_RFBG_-130);  *used for identifying Zone C;
    line_eq2=_RFBG_+110;          *used for identifying Zone C;

    if (are<=20) or (_RFBG_<70 and _SBG_<70)                                then A=1;
    else if (_RFBG_<=70 and _SBG_>=180) or (_RFBG_>=180 and _SBG_<=70)    then E=1;
    else if (_RFBG_<70 and 70<_SBG_<180) or (_RFBG_>240 and 70<_SBG_<180) then D=1;
    else if (130<=_RFBG_<=180 and _SBG_<line_eq1) or (_RFBG_>70 and _SBG_>180 and
                                                    _SBG_ > line_eq2) then C=1;
                                                                      else B=1;
    if A=1 then zone='A';
    if E=1 then zone='E';
    if D=1 then zone='D';
    if C=1 then zone='C';
    if B=1 then zone='B';

    if _SBG_~=. and _RFBG_~=.;

    label  zone='Clark Error Grid Zone';

    drop A B C D E line_eq1 line_eq2 are bias;

Run;
```

**Table 2. The Sample Paired Data Points Assigned into Different 5 Zones**

| Subject ID | Time Stamp | Sensor Glucose Value | Meter Blood Glucose | Zone |
|---|---|---|---|---|
| 001001 | 30NOV2011:07:23:00 | 123 | 113 | A |
| 001001 | 30NOV2011:10:58:00 | 75 | 60.25 | D |
| 001001 | 30NOV2011:11:03:00 | 76 | 63 | D |
| 001001 | 30NOV2011:15:18:00 | 289 | 202.5 | B |
| 001001 | 30NOV2011:15:33:00 | 256 | 195 | B |
| 001001 | 30NOV2011:15:48:00 | 245 | 187.5 | B |
| 001001 | 30NOV2011:16:03:00 | 230 | 165.5 | B |
| 001001 | 30NOV2011:16:18:00 | 200 | 163.5 | B |
| 001001 | 30NOV2011:16:33:00 | 182 | 156.5 | A |
| 001004 | 09DEC2011:11:32:00 | 162 | 132 | B |
| 001004 | 09DEC2011:11:47:00 | 144 | 123.5 | A |
| 001004 | 09DEC2011:12:02:00 | 133 | 114 | A |
| 001004 | 12DEC2011:09:02:00 | 208 | 165 | B |
| 001004 | 12DEC2011:09:17:00 | 173 | 128.5 | B |
| 001004 | 12DEC2011:09:32:00 | 145 | 110 | B |
| 001004 | 12DEC2011:09:47:00 | 130 | 104 | B |
| 001007 | 04JAN2012:11:52:00 | 258 | 71.4 | C |
| 001007 | 04JAN2012:11:57:00 | 254 | 69.05 | E |

| | | | | |
|---|---|---|---|---|
| 001007 | 04JAN2012:13:37:00 | 76 | 60.15 | D |
| 001007 | 04JAN2012:13:42:00 | 77 | 59.85 | D |
| 001007 | 04JAN2012:13:47:00 | 77 | 60.95 | D |
| 001007 | 04JAN2012:13:52:00 | 78 | 68.2 | A |
| 001007 | 04JAN2012:13:57:00 | 81 | 76.9 | A |
| 001007 | 04JAN2012:14:12:00 | 107 | 137 | B |
| 001007 | 04JAN2012:14:27:00 | 151 | 195.5 | B |
| 001007 | 04JAN2012:14:42:00 | 179 | 232.5 | B |
| 001007 | 04JAN2012:14:57:00 | 206 | 252 | A |
| 001007 | 04JAN2012:15:12:00 | 224 | 280 | A |

**STEP 3: CREATION OF SCATRERPLOT USING PROC GPLOT**

After the data points are categorized into 5 different zones; then PROC GPLOT is to be used to create scatterplot (Figure 2) only using code below. So far the things are very simple and straightforward.

```
Goptions
    reset=all
    device=PDFC
    gsfname=out
    gsfmode=append
    reset=global
    gunit=pct
    noborder
    colors=(black red Blue green)
    cback=white
    ftitle=swissb
    htitle=5
    htext=3
    ftext='Helvetica' ;

    axis1 label=(j=c h=3 "Reference BG (mg/dL)" ) order=0 to 400 by 50  offset=(,);
    axis2 label=(j=c h=3 a=90 "Sensor BG (mg/dL)") order=0 to 400 by 50 offset=(,);

    symbol1 c=CX00FFFF  i=none h=0.5  v =dot POINTLABEL= none;* RGB color system--Blue;
    symbol2 c=CX00FF33  i=none h=0.5  v =dot POINTLABEL= none;* RGB color system--Green;
    symbol3 c=CXCCCCCC  i=none h=0.5  v =dot POINTLABEL= none;* RGB color system--Gray;
    symbol4 c=Red       i=none h=0.5  v =dot POINTLABEL= none;
    symbol5 c=Pink      i=none h=0.5  v =dot POINTLABEL= none;

PROC GPLOT data= plot_ds ;
    Plot _SBG_ * _RFBG_ =zone /hminor=0 vminor=0 SKIPMISS
                               haxis=axis1 vaxis=axis2  NOlegend;
Run;
```

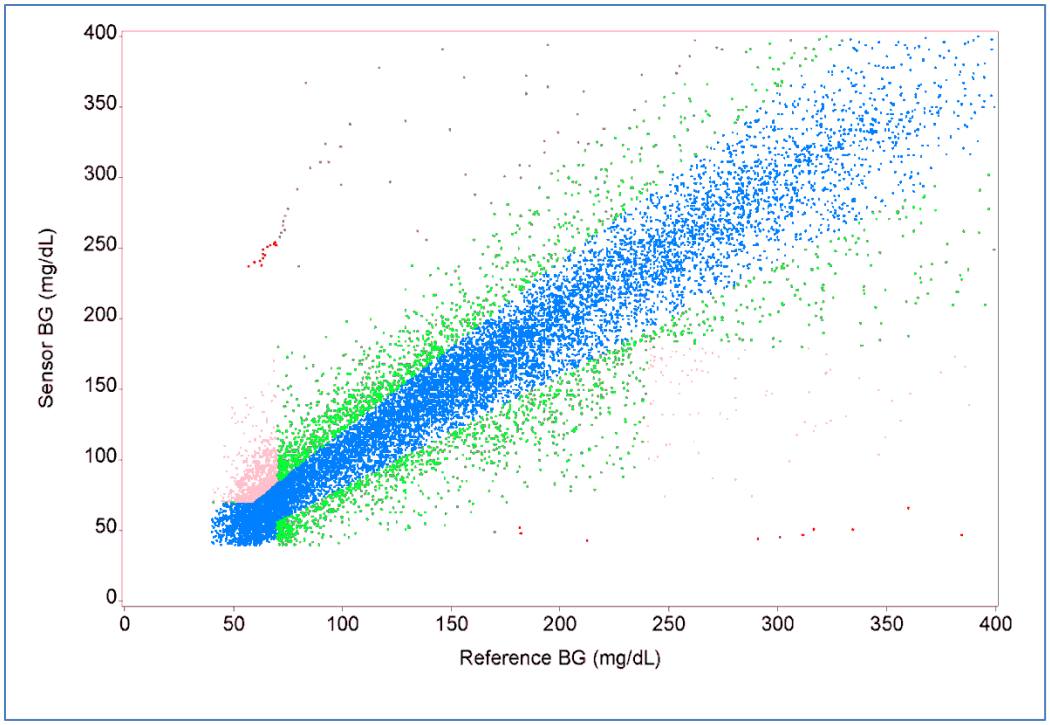



**Figure 2. The Scatterplot Produced by PROC GPLOT**

## STEP 4: DRAWING LINES AND LABELS ON THE SCATRERPLOT USING ANNOATE FACILITY

In order to create the Clarke error grid, we need to draw lines and labels to clearly separate all data points into each zone on the scatterplot created, we need to use SAS/GRAPH® annotate facility to achieve this purpose. The power of the Annotate facility is accessed through the use of a specialized data set, called SAS Annotate dataset. When using this data set, Annotate facility looks for variables with specific pre-defined names and attributes, and the values taken on by these variables and let Annotate know what your intentions are. Annotate dataset is a bridge between the user-selected procedure and the user's desire to customize the graphics output. Once the specialized SAS annotate dataset is created, the user selected procedure can execute the Annotate commands by including the following option in you SAS/GRAPH code: /ANNOTATE=<annotate file name> (like example shown below).

```
proc gplot data= plot_ds ;   Plot SBG * RefBG =zone  /Anno = anno_ds ; run;
```

Therefore, the creation of SAS Annotate dataset is a critical step. Here we are going into details on how to create SAS Annotate dataset. SAS Annotate dataset consists of a series of pre-defined variables and the assignable values. Those predefined variables can be classified into three categories, what to do, where to do and how to do. Those predefined variables and their functional meanings are listed in Table 3.

**Table 3. List of Predefined Annotate Variables and Their Usage**

| Category | Pre-defined Variables | Description |
|---|---|---|
| What to do | FUNCTION | Specifies the Annotate drawing action (also known as annotate commands). |
| | | |
| Where to do | XSYS | The coordinate system for the X variable. |

| | | |
|---|---|---|
| | YSYS | The coordinate system for the Y variable. |
| | ZSYS | The coordinate system for the Z variable (for three-dimensional graphs). |
| | X | The numeric horizontal coordinate. |
| | Y | The numeric vertical coordinate. |
| | Z | For three-dimensional graphs, specifies the coordinate for the 3rd dimension. |
| | POSITION | Placement/alignment of text. |
| | | |
| How to do | ANGLE | Angle of text label or start angle for a pie slice. |
| | COLOR | Color of graphics item. |
| | HSYS | The type of units for the size (height) variable. Defines the coordinate system and area of the output used by the SIZE variable to display the Annotate graphics. |
| | LINE | Line type of graphics item. |
| | ROTATE | Angle of individual characters in a text string or the sweep of a pie slice. |
| | SIZE | Size of the graphics item. Specific to the function. For example, size is the height of the character for a label function. |
| | STYLE | Font/pattern of a graphics item. |
| | TEXT | Text to use in a label, symbol, or comment. |
| | WHEN | Determines if Annotate command is executed (B)efore or (A)fter the graph. |

The common available values (also known as annotate commands) for predefined annotate variable 'FUNCTION' includes LABEL, MOVE, DRAW, COMMENT, PIE, and SYMBOL etc. Their functional meanings are shown in Table 4.

**Table4. The common available values (commands) for predefined annotate variable 'Function'**

| FUNCTION | DESCRIPTION |
|---|---|
| BAR | Draws a rectangle from the current position to a specified position |
| COMMENT | As a documentation aid, allows you to insert a comment into the SAS Annotate file |
| DRAW | Draws a line from the current position to a specified position. |
| LABEL | Draws text. |
| MOVE | Moves to a specific point. |
| PIE | Draws a pie slice, circle or arc. |
| POLY | Specifies the starting point of a polygon. |
| POLYCONT | Continues drawing the polygon. |
| SYMBOL | Draws a symbol. |

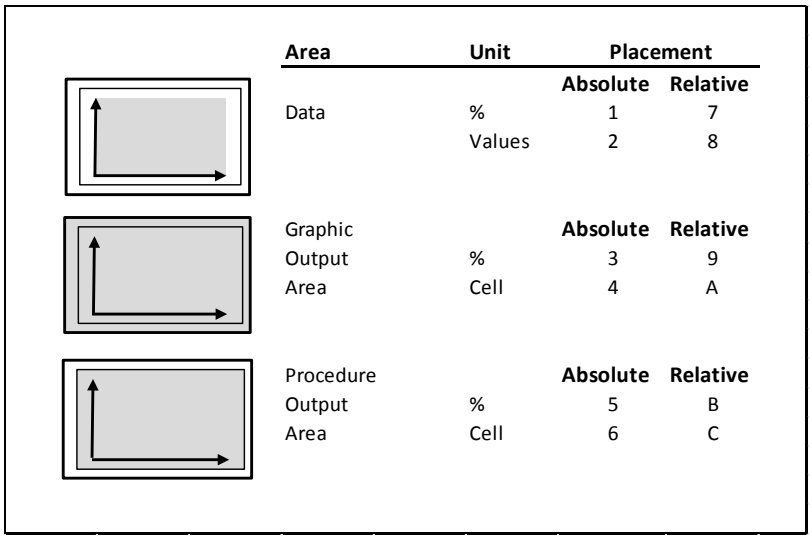
When SAS Annotate reads the value of the variable 'Function', it will know what your intentions are, but it still does not know where to do it. Therefore you need to provide values for those variables in the where-to-do category (Table 3). In order to tell SAS Annotate facility where to do, you need to specify what coordinate system to use and what the numeric horizontal coordinate is and what the numeric vertical coordinate is in a two-dimensional coordinate system. The coordinate system is determined by predefined variables XSYS, YSYS, and ZSYS. The available values for XSYS, YSYS, and ZSYS are 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, and C. The correct coordinate system code is determined by the absolute or relative placement, the drawing area, and unit used (Figure 3). Absolute placement specifies the exact location for a graphics element in the graphics output. Relative placement specifies the location with respect to another graphics element in the graphics output. The variables XSYS, YSYS, and ZSYS can have different values. Here are a few examples showing the meanings of the variables XSYS and YSYS with different assigned values.

XSYS='1' and YSYS='1' -- '**Absolute** (Placement) **Data** (Drawing Area) **Percentage** (Unit)' is to place the point according to the percentages of the horizontal and vertical axes plotted. In this system, a point (50, 50) will be put in the place where x=50 and y=50.

XSYS='2' and YSYS='2' -- '**Absolute** (Placement) **Data** (Drawing Area) **Value** (Unit)' is to place the point according to the values of the horizontal and vertical axes plotted. In this system, a point (50, 50) will be put in the place where x=50% of x-axis plotted and y=50% of y-axis plotted.

XSYS='2' and YSYS='1' -- '**Absolute** (Placement) **Data** (Drawing Area) **Value** (Unit)' for x-axis is to place the point according to the values of the horizontal x axis; and '**Absolute** (Placement) **Data** (Drawing Area) **Percentage** (Unit)' for y axis to place the point according to the percentages of the vertical axis plotted. In this system, a point (50, 50) will be put in the place where x=50% of x-axis plotted and y=50.

XSYS='3' and YSYS='3' -- '**Absolute** (Placement) **Graphics Output Area** (Drawing Area) **Percentage** (Unit)' is to place the point according to the percentages of the entire graphics area, measured from the lower left corner. In this system, a point (50, 50) will be put in the place where x=50% of horizontal length of entire graphic output area and y=50% of vertical length of entire graphic output area.



| Area | Unit | Placement | |
|---|---|---|---|
| | | **Absolute** | **Relative** |
| Data | % | 1 | 7 |
| | Values | 2 | 8 |
| Graphic | | **Absolute** | **Relative** |
| Output | % | 3 | 9 |
| Area | Cell | 4 | A |
| Procedure | | **Absolute** | **Relative** |
| Output | % | 5 | B |
| Area | Cell | 6 | C |

**Figure 3. The three important components of the Annotate coordinate systems**

The partial sample code below shows you how to draw 2 forward slash lines, from points (70,180) to (290, 400), from points (58.3,70) to (333, 400), and 2 horizontal lines, from points (0,180) to (70, 180), and from points (240,180) to (400, 180), within the different coordinate system; and how to draw 3 labels within a box. The CBOARDER variable is used to control the color of border line of a box. The CBOX variable is used to control the background color of a box. The meanings of all other pre-defined variables can be found in Tables 3 and 4. The Annotate dataset created by this sample code are shown in Table 5.

Imagine if we run the complete annotate dataset only; it would produce a graph like the one in Figure 4.

If the created Annotate dataset is included as an option (see code below) in a user selected procedure, like PROC GPLOT, then it will combine Figure 2 and 4 together and produce the desired Clarke error grid in Figure 1.

```
DATA ANNO_DS;

 *declare variables;
 Length Function style color $8 text $25;

 *2 forward slash lines;
 *(70,180) to (290, 400);
 Function='Move'; xsys='2'; ysys='2'; x=70; y=180;  output;
 Function='Draw'; xsys='2'; ysys='2'; x=290; y=400; color='black'; size=0.8; line=1; output;
 *(58.3,70) to (333, 400);
 Function='Move'; xsys='2'; ysys='2'; x=58.3; y=70; output;
 Function='Draw'; xsys='2'; ysys='2'; x=333; y=400; color='black'; size=0.8; line=1; output;

 * 2 horizontal lines;
 *(0,180) to (70, 180);
 Function='Move'; xsys='1'; ysys='1'; x=0; y=45;    output;
 Function='Draw'; xsys='1'; ysys='1'; x=17.5; y=45; color='black'; size=0.8; line=1; output;
 *(240,180) to (400, 180);
 Function='Move'; xsys='1'; ysys='1'; x=60; y=45;  output;
 Function='Draw'; xsys='1'; ysys='1'; x=100; y=45; color='black'; size=0.8; line=1; output;

 Function="Label"; xsys='1'; ysys='1'; x=32; y=52; position="6"; text="B"; Style="Arial";
       color='Black'; CBORDER='CTEXT'; CBOX='CX00FF33'; when='after'; Size=2; output;
 Function="Label"; xsys='1'; ysys='1'; x=41; y=7;  position="6"; text="C"; Style="Arial";
       color='Black'; CBORDER='CTEXT'; CBOX='CXCCCCCC'; when='after'; Size=2; output;
 Function="Label"; xsys='1'; ysys='1'; x=8; y=30;  position="6"; text="D"; Style="Arial";
       color='Black'; CBORDER='CTEXT'; CBOX='RED';      when='after'; Size=2; output;

Run;

PROC GPLOT data= plot_ds;  Plot SBG * RefBG =zone /anno = anno_ds; Run;
```

**Table 5. The Sample Annotate Datasets (Grouped by Same Category)**

| What to Do | Where to Do | | | | | How to Do | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| function | xsys | ysys | x | y | position | style | color | text | size | line | CBORDER | CBOX | when |
| Move | 2 | 2 | 70 | 180 | | | black | | 0.8 | 1 | | | |
| Draw | 2 | 2 | 290 | 400 | | | black | | 0.8 | 1 | | | |
| Move | 2 | 2 | 58.3 | 70 | | | black | | 0.8 | 1 | | | |
| Draw | 2 | 2 | 333 | 400 | | | black | | 0.8 | 1 | | | |
| Move | 1 | 1 | 0 | 45 | | | black | | 0.8 | 1 | | | |
| Draw | 1 | 1 | 17.5 | 45 | | | black | | 0.8 | 1 | | | |
| Move | 1 | 1 | 60 | 45 | | | black | | 0.8 | 1 | | | |
| Draw | 1 | 1 | 100 | 45 | | | black | | 0.8 | 1 | | | |
| Label | 1 | 1 | 32 | 52 | 6 | Arial | Black | B | 2 | 1 | CTEXT | CX00FF33 | after |
| Label | 1 | 1 | 41 | 7 | 6 | Arial | Black | C | 2 | 1 | CTEXT | CXCCCCCC | after |
| Label | 1 | 1 | 8 | 30 | 6 | Arial | Black | D | 2 | 1 | CTEXT | RED | after |

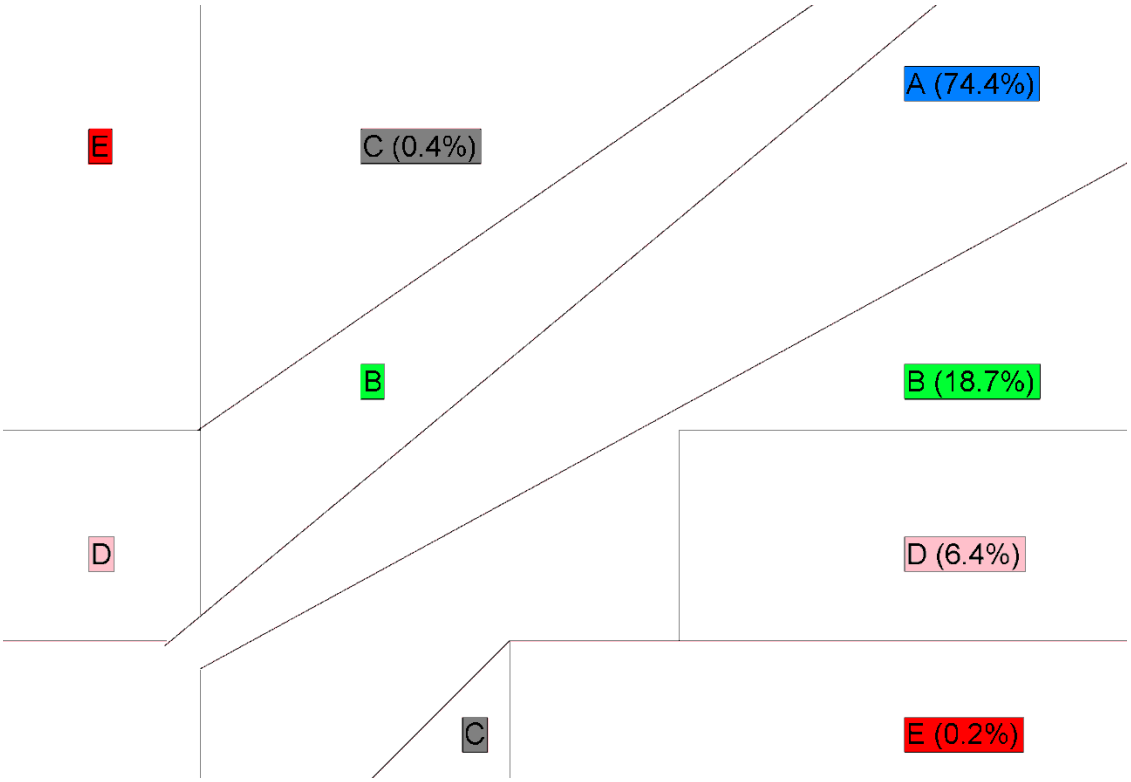



**Figure 4. The sample graph drawn by Annotate facility Only.**

### STEP 5: WRAPPING THE CODES INTO A MACRO

All the codes are then wrapped into a macro (like sample below) so the same codes can be used for different studies when the same Clarke error graphs are needed.

```
%CEG_Graph_Creator (

    directory     =     C:\CEP\Pump\Combine\Intermediate,
    dataset_name  =     Paired_SGVYSI
    graph_name    =     Pump_CEG_Abstr,
    SBG           =     SGV,
    MBG           =     Shifted_YSI
 );
```

## CONCLUSION

SAS/Graph Annotate facility is a powerful tool allowing you to draw any text or object on any graphical output. The Annotate Facility can be used to put text or symbols anywhere on the graph, control text color, font, and size, draw line segments of any length or thickness, draw polygons of any style, size, or shape, or otherwise enhance your graph. Annotate dataset is a bridge between the user-selected procedure and the user's intention to customize the graphics output.

Annotate can so far be used with the following SAS/GRAPH procedures: PROC GANNO (displaying the output from Annotate data sets), PROC GCHART (producing histograms and charts), RPOC GCONTOUR (creating contour plots), GMAP (using coordinate data sets to produce maps), PROC GPLOT (producing scatter plots), PROC GPRINT (displaying printed output) and PROC GSLIDE (creating panels).

## REFERENCES


- Carpenter, Arthur L. 2006. “Data Driven Annotations: An Introduction to SAS/GRAPH’s® Annotate Facility.” Proceedings of the Thirty-First Annual SAS Users Group International Conference. Paper 108-31. Cary, NC: SAS Institute Inc.
- Clarke William, 2005, The Original Clarke Error Grid Analysis (EGA), Diabetes Technology and Therapeutics 7:776-779, New Rochelle, NY, Mary Ann Liebert, Inc.
- David Mink and David Pasta, 2006, Improving Your Graphics Using SAS/GRAPH® Annotate Facility. SUGI 31 Proceedings, San Francisco, California, March 26-29, Cary, NC: SAS Institute Inc.
- SAS Institute, 2009. Base SAS® 9.2 Procedures Guide. Cary, NC: SAS Institute Inc.
- SAS Institute, About Annotate Graphics, SAS/GRAPH(R) 9.2: Reference, Second Edition, Cary, NC: SAS Institute Inc.
  2010, http://support.sas.com/documentation/cdl/en/graphref/63022/HTML/default/viewer.htm#annodata-creating-grelem.htm
- SAS Institute, About POSITION Variable, SAS/GRAPH(R) 9.2: Reference, Second Edition, Cary, NC: SAS Institute Inc.
  2010, http://support.sas.com/documentation/cdl/en/graphref/63022/HTML/default/viewer.htm#annotate_position.htm
- SAS Institute, 2013, SAS 9.3 Online Help Documents, Cary, NC, SAS Institute Inc.


## CONTACT INFORMATION

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


| | |
|---|---|
| Name: | Yongyin Wang |
| Enterprise: | Medtronic Diabetes |
| Address: | 18000 Devonshire Street |
| City, State ZIP: | Northridge, CA 91325 |
| Work Phone: | 818-576-3259 |
| Fax: | 818-576-6250 |
| E-mail: | Yongyin.Wang@medtronic.com |
| Web: | http://www.medtronic.com |

| | |
|---|---|
| Name: | John Shin |
| Enterprise: | Medtronic Diabetes |
| Address: | 18000 Devonshire Street |
| City, State ZIP: | Northridge, CA 91325 |
| Work Phone: | 818-576-4019 |
| Fax: | 818-576- 6648 |
| E-mail: | John.Shin@medtronic.com |
| Web: | http://www.medtronic.com |