



# SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

## Highly Customized Graphs Using ODS Graphics

Warren F. Kuhfeld  
SAS Institute Inc.

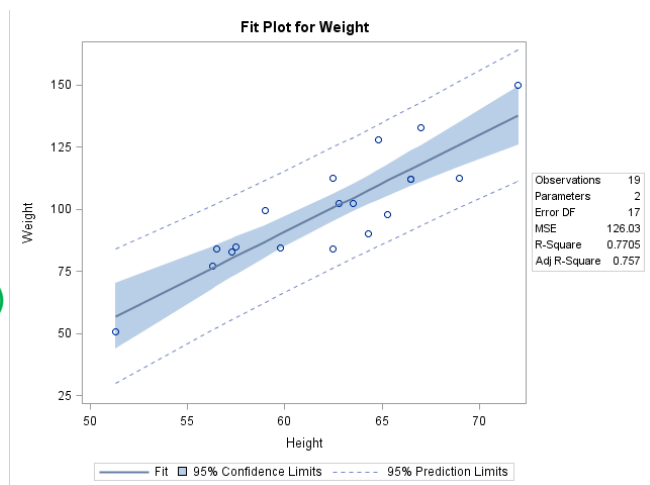
#SASGF 

1

## Graph Components

```
ods graphics on;
proc reg data=sashelp.class;
  model weight = height;
quit;
```

- Graph template (can be modified)
- Style template (can be modified or in SAS 9.4, overridden)
- Data object (can be output)
- Dynamic variables

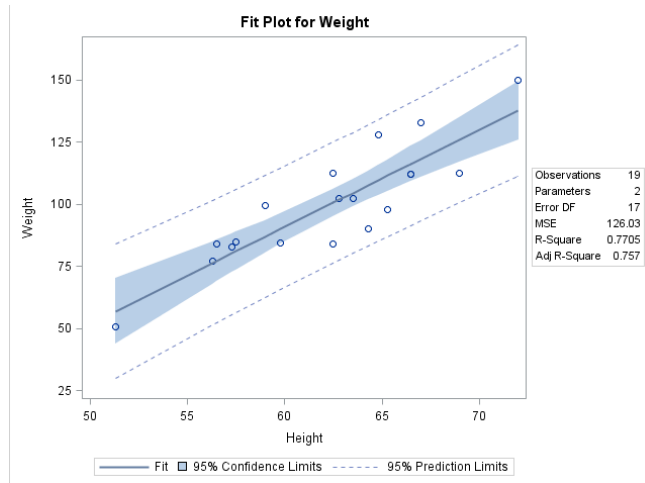


You can use an SG procedure (SGPLOT, SGPanel, or SGSCATTER) to make a graph from raw data or from the output data object from a graph.

You can then use SG Annotation to customize the graph.

# Dynamic Variables

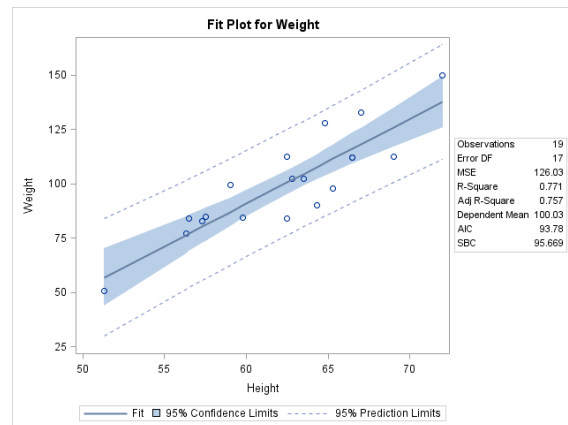
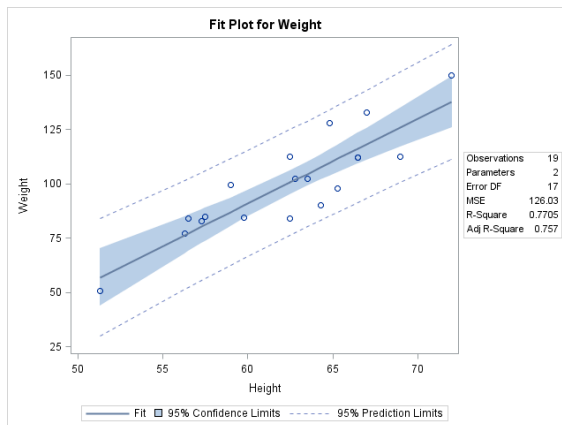
Name	Value
_SHOWCLM	1
_SHOWCLI	1
_SHOWSTATS	1
_NSTATSCOLS	2
_SHOWNOBS	1
_NOBS	19
_SHOWNPARM	1
_NPARM	2
_SHOWEDF	1
_EDF	17
_SHOWMSE	1
_MSE	126.02868962
_SHOWRSQUARE	1
_RSQUARE	0.7705068427
_SHOWADJR SQ	1
_ADJR SQ	0.7570072452
_TITLE	Fit Plot
_DEPNAME	Weight
_CONFLIMITS	95% Confidence Limits
_PREDLIMITS	95% Prediction Limits
_XVAR	_INDEPVAR1



```
dynamic _TITLE _DEPNAME _XVAR _SHOWSTATS;
entrytitle _TITLE " for " _DEPNAME;
BANDPLOT ... x=_XVAR ...
SCATTERPLOT ... x=_XVAR ...
BANDPLOT ... x=_XVAR ...
SCATTERPLOT ... x=_XVAR ...
SERIESPLOT ... x=_XVAR ...
```

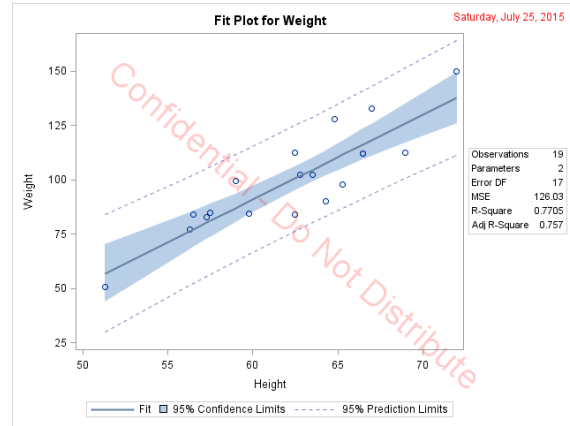
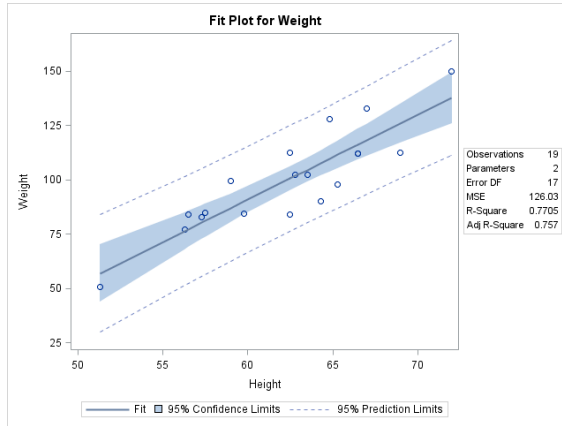
3

# Customized Dynamic Variables



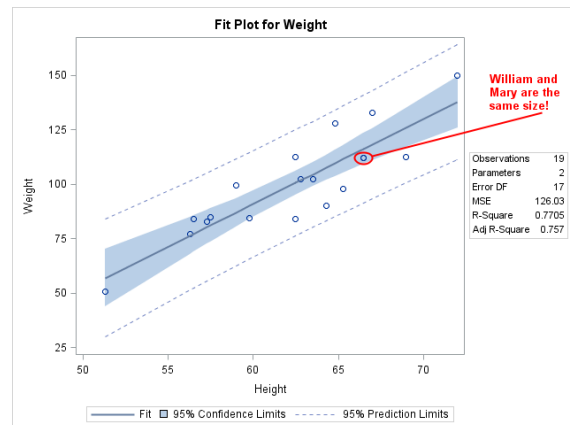
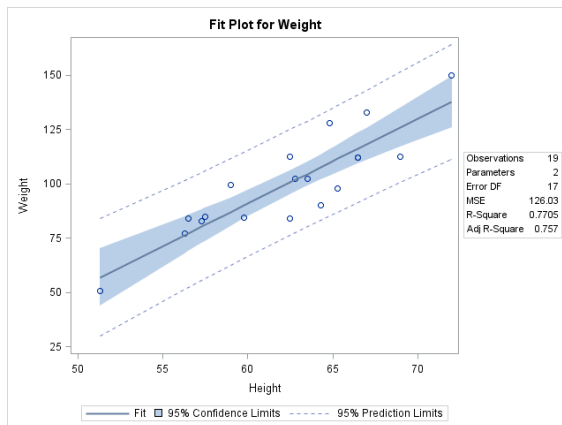
4

## SG Annotation: Text and Watermark



5

## SG Annotation: Oval, Line, and Text



6

## ODS Document

```
ods document name=MyDoc (write);
proc reg data=sashelp.class;
  ods select fitplot;
  model weight=height;
quit;
ods document close;

proc document name=MyDoc;
  list / levels=all;
quit;
```

Listing of: \Work.Mydoc\		
Order by: Insertion		
Number of levels: All		
Obs	Path	Type
1	\Reg#1	Dir
2	\Reg#1\MODEL1#1	Dir
3	\Reg#1\MODEL1#1\ObswiseStats#1	Dir
4	\Reg#1\MODEL1#1\ObswiseStats#1\Weight#1	Dir
5	\Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1	Graph

```
proc document name=MyDoc;
  replay \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1;
quit;

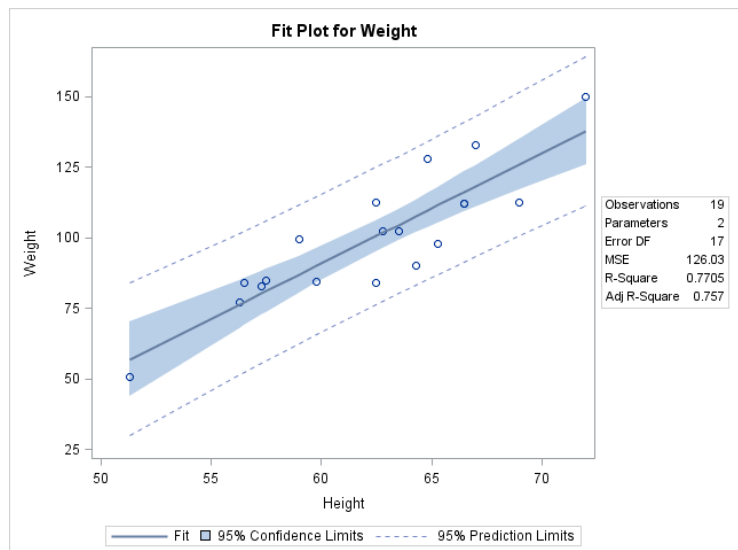
proc document name=MyDoc;
  ods exclude dynamics;
  ods output dynamics=dynamics;
  obdynam \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1;
quit;
```

7

## Replay from an ODS Document

```
ods document name=MyDoc (write);
proc reg data=sashelp.class;
  ods select fitplot;
  model weight=height;
quit;
ods document close;

proc document name=MyDoc;
  list / levels=all;
quit;
```



```
proc document name=MyDoc;
  replay \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1;
quit;
```

8

## Dynamic Variables

```
ods document name=MyDoc (write);
proc reg data=sashelp.class;
  ods select fitplot;
  model weight=height;
quit;
ods document close;

proc document name=MyDoc;
  list / levels=all;
quit;

proc document name=MyDoc;
  ods exclude dynamics;
  ods output dynamics=dynamics;
  obdynam \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1;
quit;

proc print;
  where label1 in ( . . . );
run;
```

Obs	Label1	cValue1	nValue1	Label
6	_SHOWNOBS	1	1.000000	Data
7	_NOBS	19	19.000000	Data
10	_SHOWNPARM	1	1.000000	Data
11	_NPARM	2	2.000000	Data
12	_SHOWEDF	1	1.000000	Data
13	_EDF	17	17.000000	Data
14	_SHOWMSE	1	1.000000	Data
15	_MSE	126.02868962	126.028690	Data
16	_SHOWRSQUARE	1	1.000000	Data
17	_RSQUARE	0.7705068427	0.770507	Data
18	_SHOWADJRSQ	1	1.000000	Data
19	_ADJRSQ	0.7570072452	0.757007	Data
26	_SHOWAIC	0	0	Data
27	_AIC	93.780394884	93.780395	Data
42	_BYTITLE_		.	Data
43	_BYLINE_		.	Data
44	_BYFOOTNOTE_		.	Data
45	_TITLE	Fit Plot	.	Data
46	_DEPNAME	Weight	.	Data
47	_DEPLABEL	Weight	.	Data
48	_SHORTYLABEL	Weight	.	Data
49	_SHORTXLABEL	Height	.	Data
50	_CONFLIMITS	95% Confidence Limits	.	Data
51	_PREDLIMITS	95% Prediction Limits	.	Data

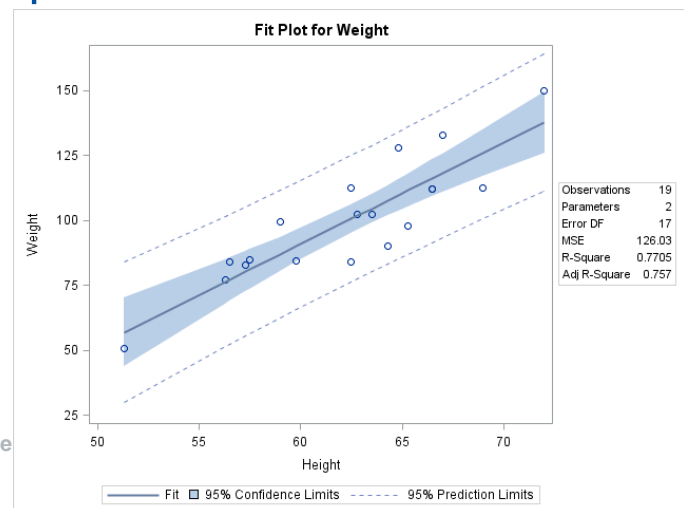
## Replay with Dynamics Specified

```
ods document name=MyDoc (write);
proc reg data=sashelp.class;
  ods select fitplot;
  model weight=height;
quit;
ods document close;

proc document name=MyDoc;
  list / levels=all;
quit;

proc document name=MyDoc;
  ods exclude dynamics;
  ods output dynamics=dynamics;
  obdynam \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1;
quit;

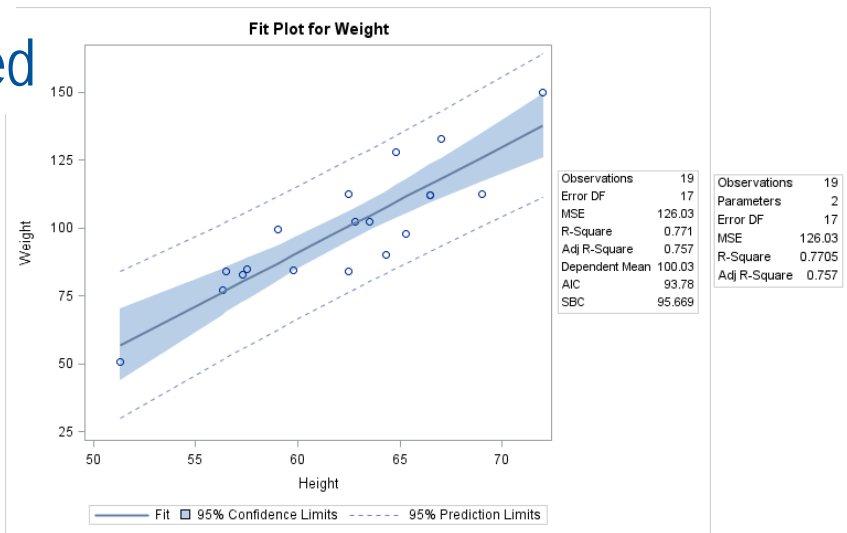
proc document name=MyDoc;
  replay \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1 /
    dynamdata=dynamics;
quit;
```



## Dynamics Modified

```
data dynamics2;
  set dynamics;
  if label1 = '_SHOWNPARM'
    then nvalue1 = 0;
  if label1 = '_SHOWAIC'
    then nvalue1 = 1;
  if label1 = '_SHOWSBC'
    then nvalue1 = 1;
  if label1 = '_SHOWDEPMEAN'
    then nvalue1 = 1;
  if label1 in ('_RSQUARE'
    '_ADJR SQ')
    then nvalue1 =
      round(nvalue1, 0.001);
run;
```

```
proc document name=MyDoc;
  replay \Reg#1\MODEL1#1\ObswiseStats#1\Weight#1\FitPlot#1 /
    dynamdata=dynamics2;
quit;
```



11

## Naïve Rendering

```
proc reg data=sashelp.class;
  ods output fitplot=fp;
  model weight = height;
quit;

proc sgrender data=fp
  template=Stat.REG.Graphics.Fit;
run;
```

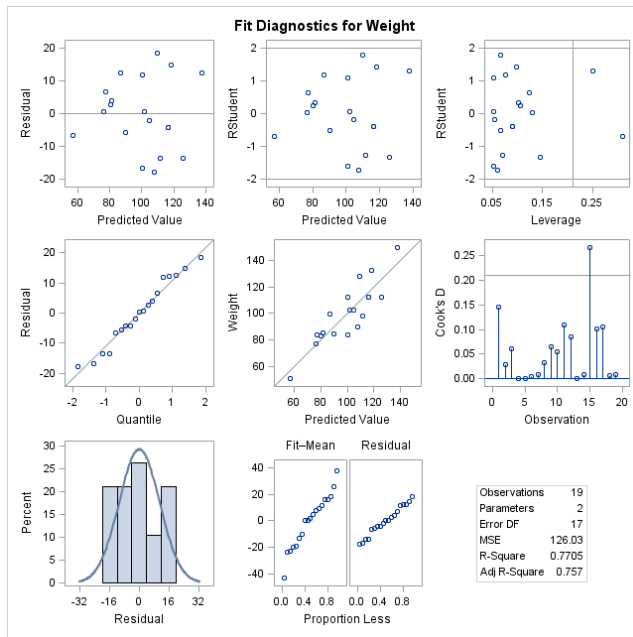


WARNING: The SCATTERPLOT statement will not be drawn because one or more of the required arguments were not supplied.

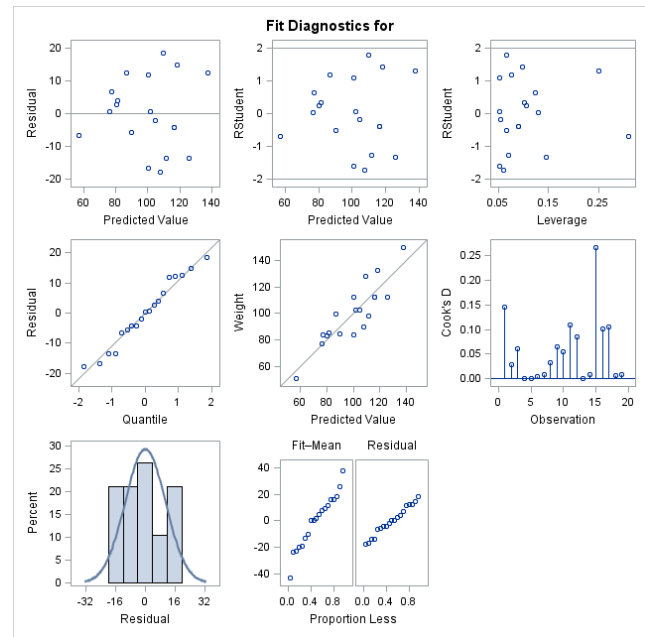
WARNING: The SERIESPLOT statement named 'Fit' will not be drawn because one or more of the required arguments were not supplied.

12

## Diagnostics Panel



## Naïve Rendering



## PROC SGRENDER Code with Dynamic Variables

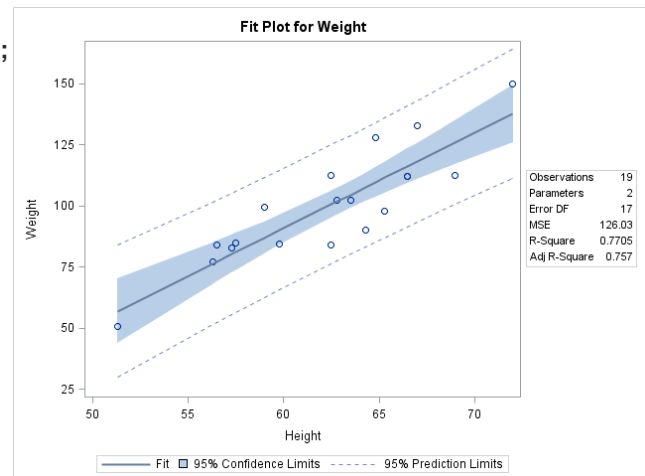
The next two slides show how to use CALL EXECUTE and a DATA step to generate this step:

```
proc sgrender data=fp template=Stat.REG.Graphics.Fit;
  dynamic _SHOWCLM = 1 _SHOWCLI = 1 _WEIGHT = 0 _SHOWSTATS = 1 _NSTATSCOLS = 2
  _SHOWNOBS = 1 _NOBS = 19 _SHOWTOTFREQ = 0 _TOTFREQ = 19 _SHOWNPARM = 1
  _NPARM = 2 _SHOWEDF = 1 _EDF = 17 _SHOWMSE = 1 _MSE = 126.02868962
  _SHOWRSQUARE = 1 _RSQUARE = 0.7705068427 _SHOWADJRSQ = 1 _ADJRSQ = 0.7570072452
  _SHOWSSE = 0 _SSE = 2142.4877235 _SHOWDEPMEAN = 0 _DEPMEAN = 100.02631579
  _SHOWCV = 0 _CV = 11.223296526 _SHOWAIC = 0 _AIC = 93.780394884 _SHOWBIC = 0
  _BIC = 96.223301459 _SHOWCP = 0 _CP = 2 _SHOWGMSEP = 0 _GMSEP = 140.9531397
  _SHOWJP = 0 _JP = 139.29486747 _SHOWPC = 0 _PC = 0.2834915472 _SHOWSBC = 0
  _SBC = 95.669272843 _SHOWSP = 0 _SP = 7.876793101 _TITLE = "Fit Plot"
  _DEPNAME = "Weight" _DEPLABEL = "Weight" _SHORTYLABEL = "Weight"
  _SHORTXLABEL = "Height" _CONFLIMITS = "95% Confidence Limits"
  _PREDLIMITS = "95% Prediction Limits" _XVAR = "INDEPVAR1";
run;
```

- Use CALL EXECUTE to generate PROC SGRENDER code to replay the graph and use the dynamic variables
- Programs that write programs
- The automatic variable \_infile\_ contains the contents of the input buffer

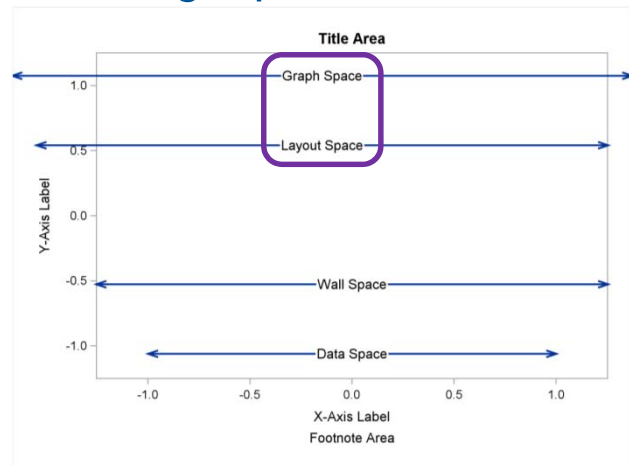
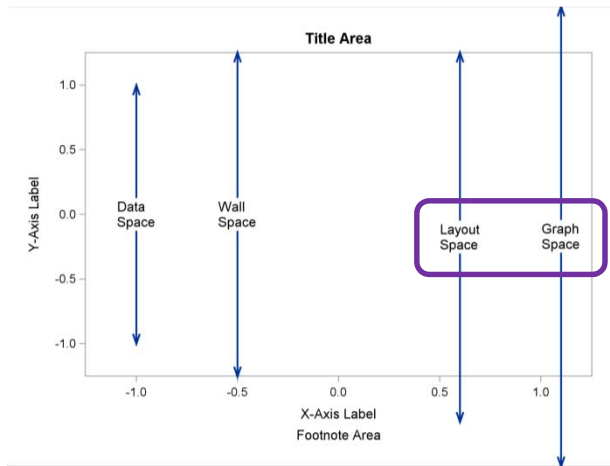
Obs	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83.0
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5
9	Jeffrey	M	13	62.5	84.0
10	John	M	12	59.0	99.5
11	Joyce	F	11	51.3	50.5
12	Judy	F	14	64.3	90.0
13	Louise	F	12	56.3	77.0
14	Mary	F	15	66.5	112.0
15	Philip	M	16	72.0	150.0
16	Robert	M	12	64.8	128.0
17	Ronald	M	15	67.0	133.0
18	Thomas	M	11	57.5	85.0
19	William	M	15	66.5	112.0

OPTIONS SOURCE;  
displays code





## Primer on SG Annotation Drawing Spaces



- Drawing spaces: 'DataPercent', 'DataPixel', 'DataValue', 'GraphPercent', 'GraphPixel', 'LayoutPercent', 'LayoutPixel', 'WallPercent', and 'WallPixel'
- 'LayoutPercent' is the default for the next PROC SGRENDER ('GraphPercent' for PROC SGPLOT)
- 'Graph...' and 'Layout...' work with an ANNOTATE statement in the BEGINGRAPH block
- 'Data...' and 'Wall...' produce: **WARNING: XSPACE= is invalid. Draw statement discarded**

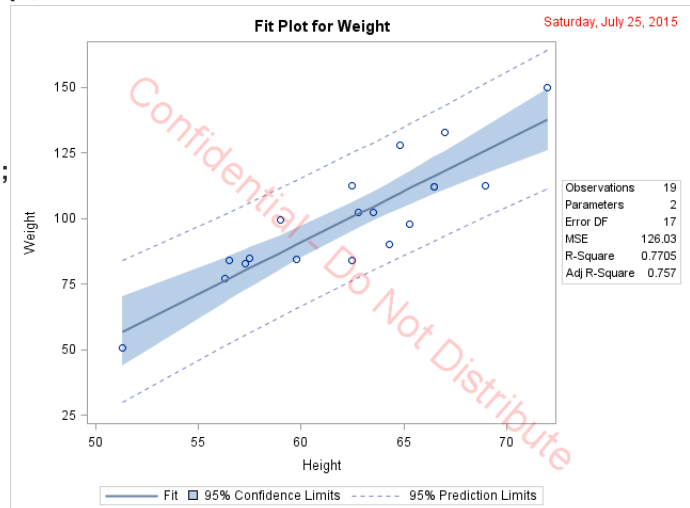
17

## Annotate the Layout Space

```
proc template;
  delete Stat.REG.Graphics.Fit / store=sasuser.templat;
  source Stat.REG.Graphics.Fit / file='temp.tmp';
quit;
```

```
data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1 then call execute('proc template;');
  call execute(_infile_);
  if lowercase(_infile_) = ' begingraph'
    then bg + 1;
  if bg and index(_infile_, ';') then do;
    bg = 0;
    call execute('annotate;');
  end;
run;
```

```
BeginGraph;
annotate;
```



18

## Annotate the Layout Space

```
data anno;
  length Label $ 40;
  Function = 'Text';   Label = 'Saturday, July 25, 2015';
  Width    = 100;     x1 = 100; y1 = 104;
  Anchor   = 'Right'; TextColor = 'Red'; TextSize = 9;
  output;
```

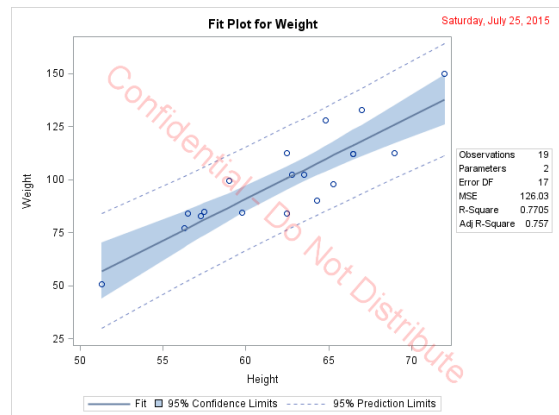
```
Label = 'Confidential - Do Not Distribute';
Width = 150;     x1 = 50; y1 = 50;
Anchor = 'Center';
Transparency = 0.8; TextSize = 28; Rotate = -41;
output;
run;
```

```
proc print;
run;
```

Default coordinates are percentages of the layout area ('LayoutPercent')

Obs	Label	Function	Width	x1	y1	Anchor	TextColor	TextSize	Transparency	Rotate
1	Saturday, July 25, 2015	Text	100	100	104	Right	Red	9	.	.
2	Confidential - Do Not Distribute	Text	150	50	50	Center	Red	28	0.8	-41

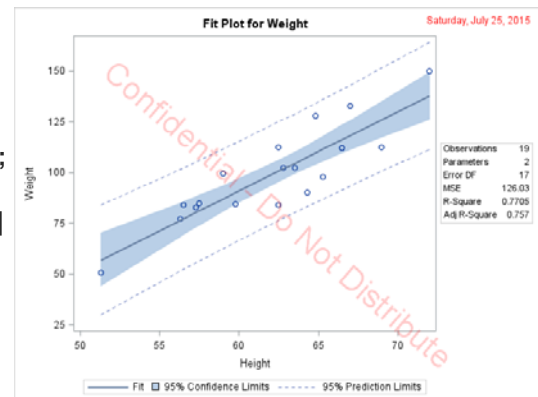
19



## Annotate the Layout Space

```
data _null_;
  set dynamics(where=(label1 ne '___NOBS___')) end=eof;
  if _n_ = 1 then do;
    call execute('proc sgrender data=fp sganno=anno ' ||
      'template=Stat.REG.Graphics.Fit;');
    call execute('dynamic');
  end;
  if cvalue1 ne '' then
    call execute(catx(' ', label1, '=',
      ifc(n(nvalue1), cvalue1, quote(trim(cvalue1)))));
  if eof then call execute('; run;');
run;
```

```
proc template;
  delete Stat.REG.Graphics.Fit / store=sasuser.template;
quit;
```



20

## Annotate the Data Space

```

define statgraph Stat.Reg.Graphics.Fit;
  notes "Fit Plot";
  dynamic ...;
  BeginGraph;
    entrytitle halign=left textattrs=GRAPHVALUETEXT _MODELLABEL halign=center
      textattrs=GRAPHTITLETEXT _TITLE " for " _DEPNAME;
    layout Overlay / yaxisopts=(label=_DEPLABEL shortlabel=_SHORTYLABEL)
      xaxisopts=(shortlabel=_SHORTXLABEL);
    ...
    annotate / id="a";
    SCATTERPLOT y=DEPVAR x=_XVAR / markerattrs=GRAPHDATADEFAULT primary=
      true rolename=( _tip1=OBSERVATION _id1=ID1 _id2=ID2 _id3=ID3 _id4=
        ID4 _id5=ID5) tip=(y x _tip1 _id1 _id2 _id3 _id4 _id5);
    SERIESPLOT y=PREDICTEDVALUE x=_XVAR / lineattrs=GRAPHFIT connectorder=
      xaxis name="Fit" LegendLabel="Fit";
    ...
  endlayout;
  ...
EndGraph;
end;

```

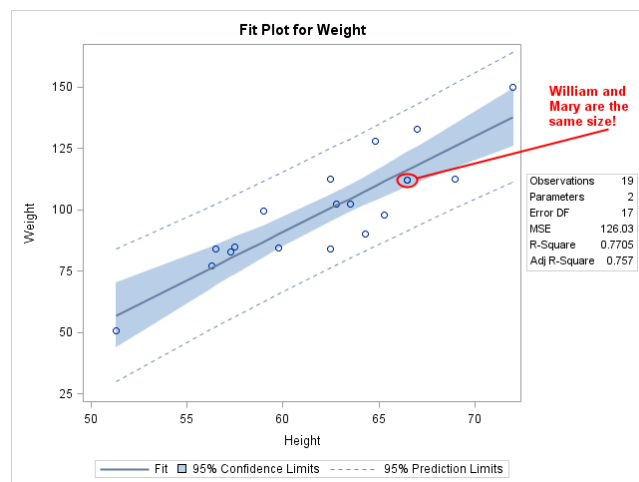
21

## Annotate the Data Space

```

data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1
    then call execute('proc template;');
  if left(_infile_) = 'SCATTERPLOT' then call execute('annotate / id="a";');
  call execute(_infile_);
run;

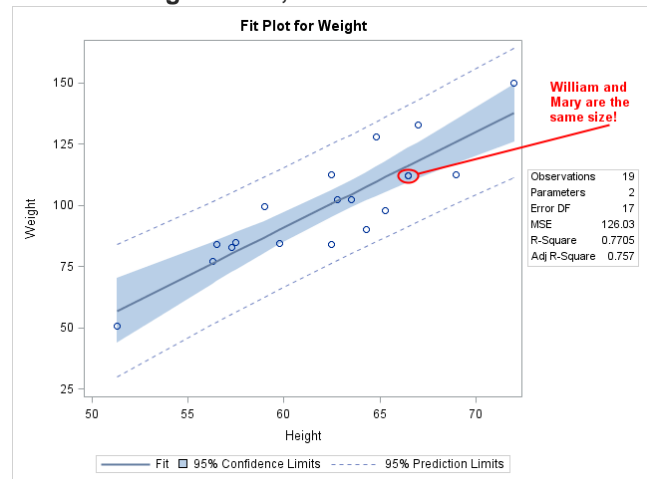
```



22

## Annotate the Data Space

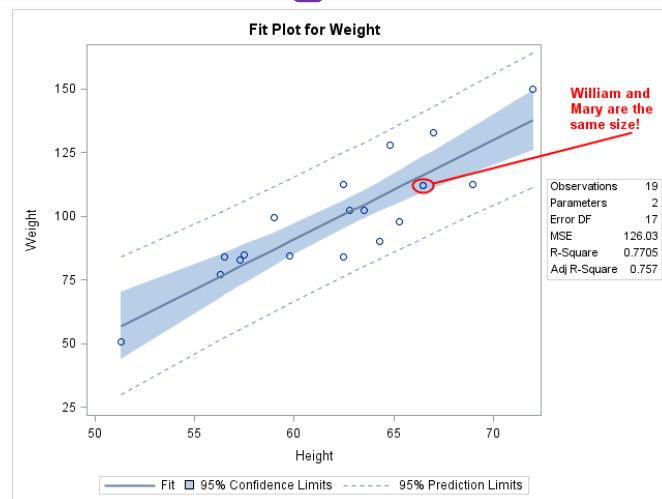
```
data anno(drop=name sex age);
  set sashelp.class(where=(name='William')) rename=(height=x1 weight=y1);
  retain DrawSpace 'DataValue' Function 'Oval' HeightUnit WidthUnit 'Data'
    Height 5 Width 1 ID 'a' TextColor LineColor 'Red' TextWeight 'Bold';
  output;
  function = 'Line';
  x1 + 0.5;
  y1 + 1;
  x2 = x1 + 10;
  y2 = y1 + 20;
  output;
  x1 = x2 - 1;
  y1 = y2 + 9;
  function = 'Text';
  Label = 'William and Mary are the same size!';
  Anchor = 'Center';
  Width = 6;
  output;
run;
```



23

## SG Annotation Data Set

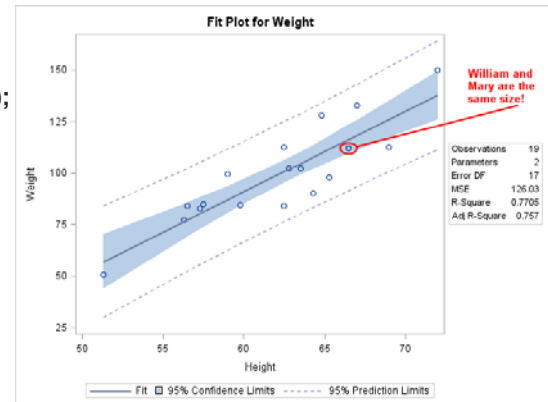
x1	y1	DrawSpace	Function	HeightUnit	WidthUnit	Height	Width	ID	TextColor	LineColor	TextWeight	x2	y2	Label	Anchor
66.5	112	DataValue	Oval	Data	Data	5	1	a	Red	Red	Bold	.	.		
67.0	113	DataValue	Line	Data	Data	5	1	a	Red	Red	Bold	77	133		
76.0	142	DataValue	Text	Data	Data	5	6	a	Red	Red	Bold	77	133	William and Mary are the same size!	Center



24

## Annotate the Data Space

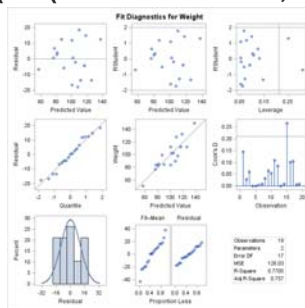
```
data _null_;
  set dynamics(where=(label1 ne '___NOBS___')) end=eof;
  if _n_ = 1 then do;
    call execute('proc sgrender data=fp sganno=anno ' ||
      'template=Stat.Reg.Graphics.Fit;');
    call execute('dynamic');
  end;
  if cvalue1 ne '' then
    call execute(catx(' ', label1, '=',
      ifc(n(nvalue1), cvalue1, quote(trim(cvalue1)))));
  if eof then call execute('; run;');
run;
```



25

## Multiple Overlays

```
data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1 then call execute('proc template;');
  call execute(_infile_);
  if index(lowercase(_infile_), ' layout overlay ')
    then lo + 1;
  if lo and index(_infile_, ';') then do;
    lo = 0;
    lonum + 1;
    call execute(catt('annotate / id="LO', lonum, '"'););
  end;
run;
```



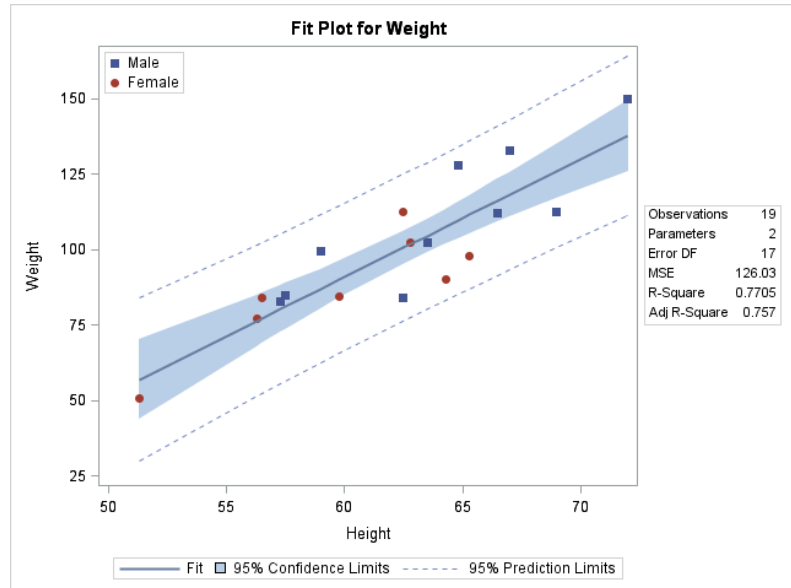
```
define statgraph Stat.Reg.Graphics.DiagnosticsPanel;
  ...
  BeginGraph / designheight=defaultDesignWidth;
  layout lattice / columns=3 rows=3 ...;
  layout overlay / xaxisopts=(shortlabel='Predicted');
  annotate / id="LO1";
  ...
  endlayout;
  layout overlay / xaxisopts=(shortlabel='Predicted');
  annotate / id="LO2";
  ...
  endlayout;
  layout overlay / xaxisopts=(label='Leverage' offsetmax=0.05)
    yaxisopts=(offsetmin=0.05 offsetmax=0.05);
  annotate / id="LO3";
  ...
  endlayout;
  layout overlay / yaxisopts=(label='Residual'
    shortlabel='Resid')
    xaxisopts=(label='Quantile');
  annotate / id="LO4";
  ...
  endlayout;
  ...
  EndGraph;
end;
```

26

# Template Modification and Style Overrides

```
proc template;
  delete Stat.REG.Graphics.Fit /
    store=sasuser.templat;
  source Stat.REG.Graphics.Fit /
    file='temp.tmp';
quit;

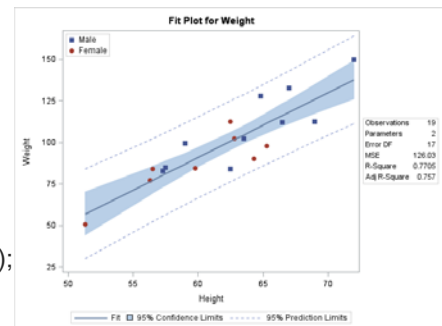
proc format;
  value $sex 'M' = 'Male' 'F' = 'Female';
run;
```



27

## Template Modification

```
data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1 then call execute('proc template;');
  if left(_infile_) = 'SCATTERPLOT y=DEPVAR' then do;
    _infile_ = tranwrd(_infile_, 'markerattrs=GRAPHDATADEFAULT', ' ');
    _infile_ = tranwrd(_infile_, '/', ' / group=id1 name="sc"');
  end;
  if left(_infile_) = 'BeginGraph' then
    _infile_ = 'BeginGraph / attrpriority=none datasymbols=(squarefilled circlefilled)';
  call execute(_infile_);
  if left(_infile_) = 'SCATTERPLOT y=DEPVAR' then
    call execute('discretelegend "sc" / location=inside across=1 autoalign=(topleft)');
run;
```



```
proc reg data=sashelp.class;
  ods output fitplot=fp;
  model weight=height;
  id sex;
  format sex $sex.;
quit;
```

AttrPriority=None enables markers of different shapes

The sex variable is in the data object and is named id1

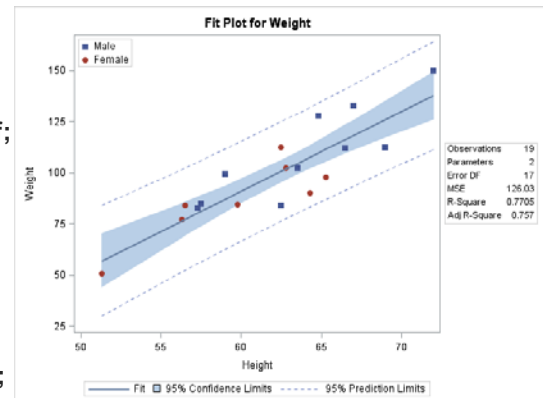
UpperCLMean	LowerCLMean	_INDEPVAR1	PredictedValue	UpperCL	LowerCL	DepVar	id1
135.071	116.942	69.0	126.006	151.367	100.646	112.5	Male
85.630	68.907	56.5	77.268	102.386	52.150	84.0	Female

28

## Data Object Modification – What if Sex had not been there?

```
data both(drop=height weight rename=(sex=id1));
  merge sashelp.class(keep=height weight sex) fp;
  if height ne _indepvar1 or weight ne depvar then put _all_;
  format sex $sex.;
run;
```

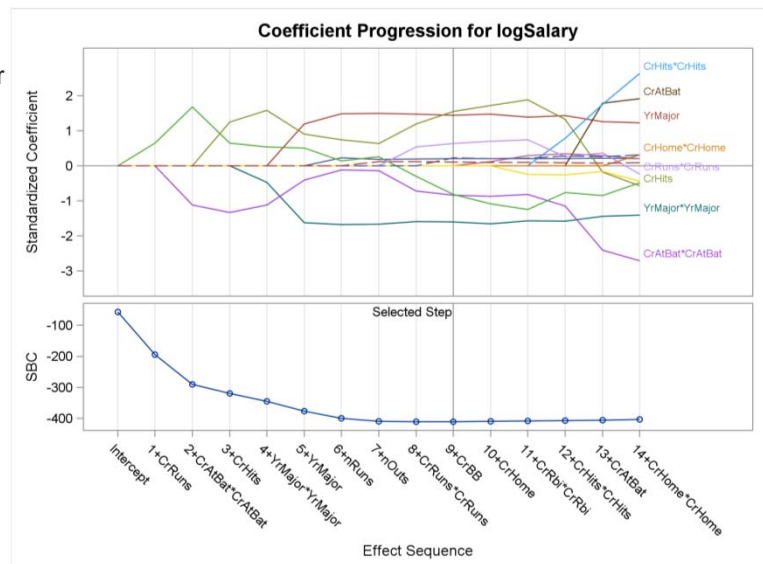
```
data _null_;
  set dynamics(where=(label1 ne '___NOBS___')) end=eof;
  if _n_ = 1 then do;
    call execute('proc sgrender data=both;');
    call execute('template=Stat.REG.Graphics.Fit;');
    call execute('dynamic;');
  end;
  if cvalue1 ne '' then
    call execute(catx(' ', label1, '=',
      ifc(n(nvalue1), cvalue1, quote(trim(cvalue1)))));
  if eof then call execute('; run;');
run;
```



29

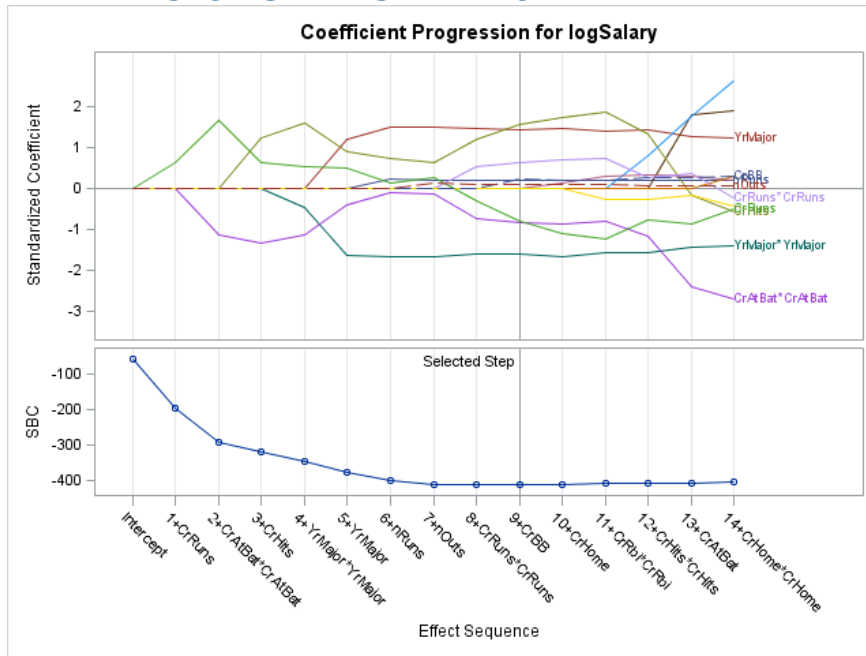
## PROC GLMSELECT

```
proc glmselect data=sashelp.baseball plots=coefficients;
  class league division;
  model logSalary = nAtBat nHits nHome
    nRuns nRBI nBB yrMajor|yrMajor
    crAtBat|crAtBat crHits|crHits
    crHome|crHome crRuns|crRuns
    crRbi|crRbi crBB|crBB league
    division nOuts nAssts nError /
    selection=forward(stop=AICC
      choose=SBC);
run;
```

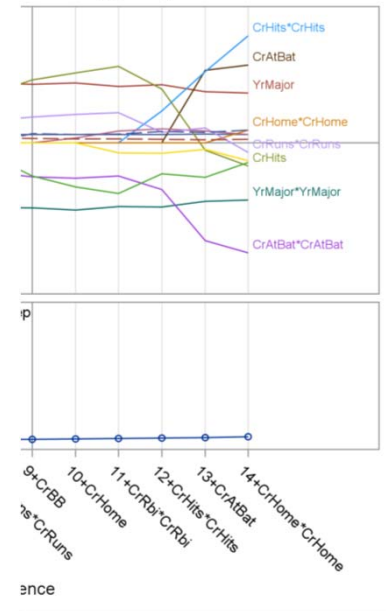


30

# PROC GLMSELECT



on for logSalary



## Create ODS Document and Data Object

```
ods document name=MyDoc (write);
proc glmselect data=sashelp.baseball plots=coefficients;
  ods select CoefficientPanel;
  ods output CoefficientPanel=cp;
  class league division;
  model logSalary = nAtBat nHits nHome nRuns nRBI nBB yrMajor|yrMajor
    crAtBat|crAtBat crHits|crHits crHome|crHome crRuns|crRuns
    crRbi|crRbi crBB|crBB league division nOuts nAssts nError /
    selection=forward(stop=AICC choose=SBC);
run;
ods document close;

proc document name=MyDoc;
  list / levels=all;
quit;
```

Listing of: \Work.Mydoc\		
Order by: Insertion		
Number of levels: All		
Obs	Path	Type
1	\GLMSelect#1	Dir
2	\GLMSelect#1\Summary#1	Dir
3	\GLMSelect#1\Summary#1\CoefficientPanel#1	Graph



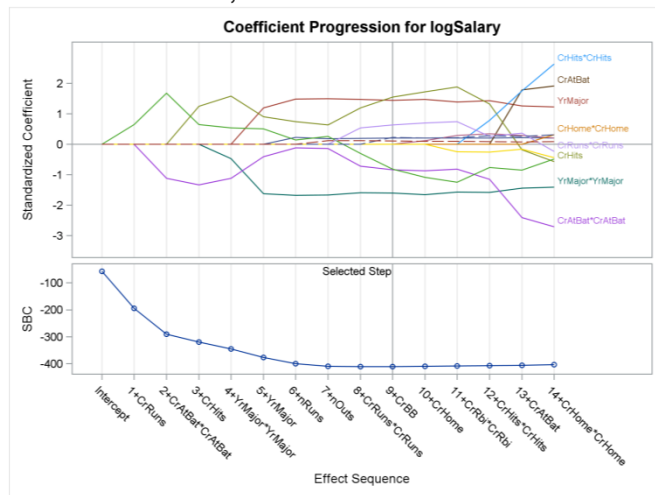
## Dynamic Variables and Step Chosen

```
proc document name=MyDoc;
  ods exclude dynamics;
  ods output dynamics=dynamics;
  obdynam \GLMSelect#1\Summary#1\CoefficientPanel#1;
quit;
```

```
data _null_;
  set dynamics;
  if label1 = '_CHOSENVALUE'
    then call symputx('cv', cvalue1);
run;

%put &cv;
```

9+CrBB



33

## Chosen Parameters and Label of the Last Step

```
data chosen(keep=parameter rename=(parameter=parm));
  set cp(where=(steplabel = "&cv"));
  if standardizedest ne 0;
run;
```

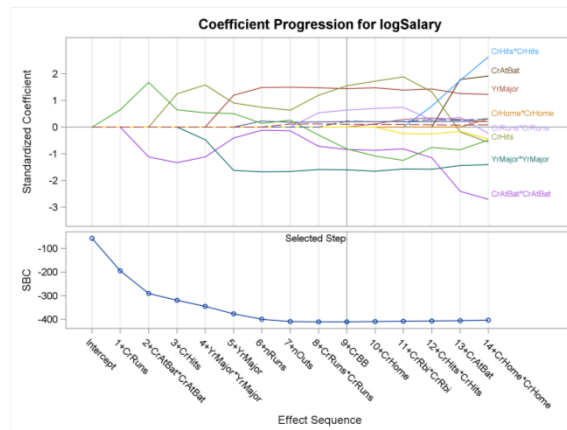
&cv: 9+CrBB

```
proc print;
run;

data _null_;
  set cp;
  call symputx('last', steplabel);
run;

%put &last;
```

14+CrHome\*CrHome



Obs	parm
1	nRuns
2	YrMajor
3	YrMajor*YrMajor
4	CrAtBat*CrAtBat
5	CrHits
6	CrRuns
7	CrRuns*CrRuns
8	CrBB
9	nOuts

34

## Modify Data Object

```
data cp2;
  set cp;
  rhslabelyvalue = .;
  if steplabel = "&last" then do i = 1 to nob;
    set chosen point=i nob=nob;
    if parm eq parameter then rhslabelyvalue = standardizedest;
  end;
run;
```

&last: 14+CrHome\*CrHome

```
proc print;
  where steplabel = "&last";
run;
```

Obs	parm
1	nRuns
2	YrMajor
3	YrMajor*YrMajor
4	CrAtBat*CrAtBat
5	CrHits
6	CrRuns
7	CrRuns*CrRuns
8	CrBB
9	nOuts

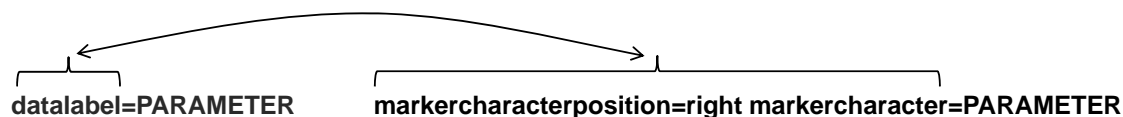
Obs	StandardizedEst	STEPLABEL	CurveNumber	CurveIndex	Step	Parameter	RhsLabelYValue	SBC
197	0.20733	14+CrHome*CrHome	1	1	14	nRuns	0.20733	-402.904
198	1.22708	14+CrHome*CrHome	2	2	14	YrMajor	1.22708	.
199	-1.41214	14+CrHome*CrHome	3	3	14	YrMajor*YrMajor	-1.41214	.
200	1.91095	14+CrHome*CrHome	4	4	14	CrAtBat	.	.
201	-2.71058	14+CrHome*CrHome	5	5	14	CrAtBat*CrAtBat	-2.71058	.
202	-0.57227	14+CrHome*CrHome	6	6	14	CrHits	-0.57227	.
203	2.63635	14+CrHome*CrHome	7	7	14	CrHits*CrHits	.	.
204	0.20877	14+CrHome*CrHome	8	8	14	CrHome	.	.
205	0.31144	14+CrHome*CrHome	9	9	14	CrHome*CrHome	.	.
206	-0.48425	14+CrHome*CrHome	10	10	14	CrRuns	-0.48425	.
207	-0.23971	14+CrHome*CrHome	11	11	14	CrRuns*CrRuns	-0.23971	.
208	-0.44132	14+CrHome*CrHome	12	12	14	CrRbi*CrRbi	.	.
209	0.30819	14+CrHome*CrHome	13	13	14	CrBB	0.30819	.
210	0.08364	14+CrHome*CrHome	14	14	14	nOuts	0.08364	.

35

## Modify Template

```
proc template;
  delete Stat.GLMSELECT.Graphics.CoefficientPanel;
  source Stat.GLMSELECT.Graphics.CoefficientPanel / file='temp.tmp';
quit;
```

```
data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1 then call execute('proc template;');
  if index(_infile_, 'datalabel=PARAMETER') then
    _infile_ = tranwrd(_infile_, 'datalabel',
      'markercharacterposition=right markercharacter');
  call execute(_infile_);
run;
```



Markers



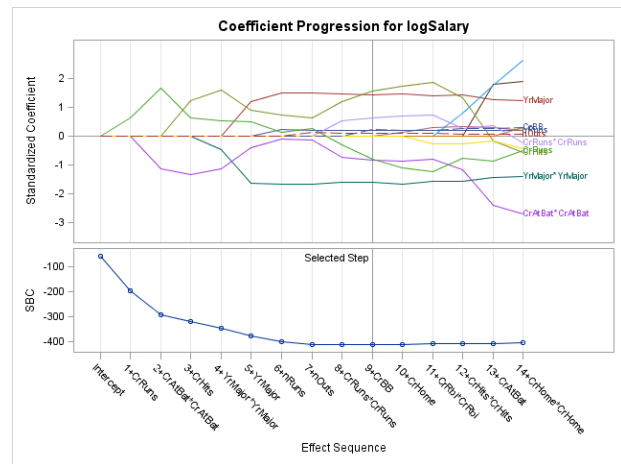
Labels



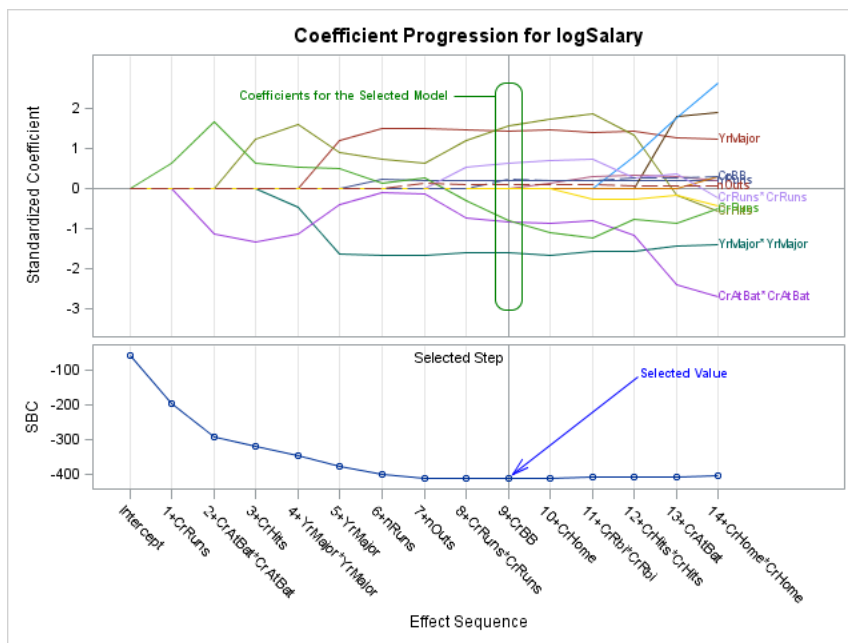
36

## Make Graph

```
data _null_;
  set dynamics(where=(label1 ne '___NOBS___')) end=eof;
  if _n_ = 1 then do;
    call execute('proc sgrender data=cp2');
    call execute('template=Stat.GLMSELECT.Graphics.CoefficientPanel;');
    call execute('dynamic');
  end;
  if cvalue1 ne '' then
    call execute(catx(' ', label1, '=',
                      ifc(n(nvalue1), cvalue1,
                          quote(trim(cvalue1)))));
  if eof then call execute('; run;');
run;
```



## PROC GLMSELECT with Annotation



38

## Modify the Template

```
data _null_;
  infile 'temp.tmp';
  input;
  if _n_ = 1 then call execute('proc template;');
  if index(_infile_, 'datalabel=PARAMETER') then
    _infile_ = tranwrd(_infile_, 'datalabel',
      'markercharacterposition=right markercharacter');
  call execute(_infile_);
  if index(lowercase(_infile_), ' layout overlay ') then lo + 1;
  if lo and index(_infile_, ';') then do;
    lo = 0;
    lonum + 1;
    call execute(catt('annotate / id="LO', lonum, ";"));
  end;
end;
run;
```

```
define statgraph Stat.GLMSelect.Graphics.CoefficientPanel;
  ...
  BeginGraph;
  layout lattice ...;
  layout overlay ...;
  annotate / id="LO1";
  ...
  endlayout;
  if (_SHOWPVAL = 1)
    layout overlay ...;
    annotate / id="LO2";
    ...
  endlayout;
  else
    layout overlay ...;
    annotate / id="LO3";
    ...
  endlayout;
  endif;
  endlayout;
  ...
EndGraph;
end;
```

39

## Annotation Data Set

```
data anno;
  length ID $ 3 Function $ 9 Label $ 40;
  retain x1space y1space x2space y2space 'DataPercent' Direction 'In';
  length Anchor $ 10 xC1 xC2 $ 20;
  retain Scale 1e-12 Width 100 WidthUnit 'Data' CornerRadius 0.8
    TextSize 7 TextWeight 'Bold'
    LineThickness 1.2 DiscreteOffset -0.3 LineColor 'Green';

  ID      = 'LO1';      Function = 'Text';
  Anchor  = 'Right';    TextColor = 'Green';
  x1      = 55;         y1      = 94;
  Label   = 'Coefficients for the Selected Model';      output;

  Function = 'Line';    x1      = .;
  x1space  = 'DataValue'; x2space = x1space;
  xc1      = '9+CrBB';  xc2      = '8+CrRuns*CrRuns';
  y1      = 94;         y2      = 94;      output;

  Function = 'Rectangle'; y1space = 'WallPercent';
  Anchor   = 'BottomLeft'; y1      = 10;
  Height   = 80;         Width    = 0.6;      output;

  ID      = 'LO3';      Width    = 100;
  Function = 'Text';    Label     = 'Selected Value';
  x1space  = 'DataPercent'; y1space = x1space;
  Anchor   = 'Left';    TextColor = 'Blue';
  x1      = 86;         y1      = 84;      output;

  Function = 'Arrow';    LineColor = 'Blue';
  x1space  = 'DataValue'; x2space  = x1space;
  xc1      = '9+CrBB';  xc2      = '12+CrHits*CrHits';
  y1      = 4;         y2      = 83;
  DiscreteOffset = .1;  x1      = .;      output;
run;
```

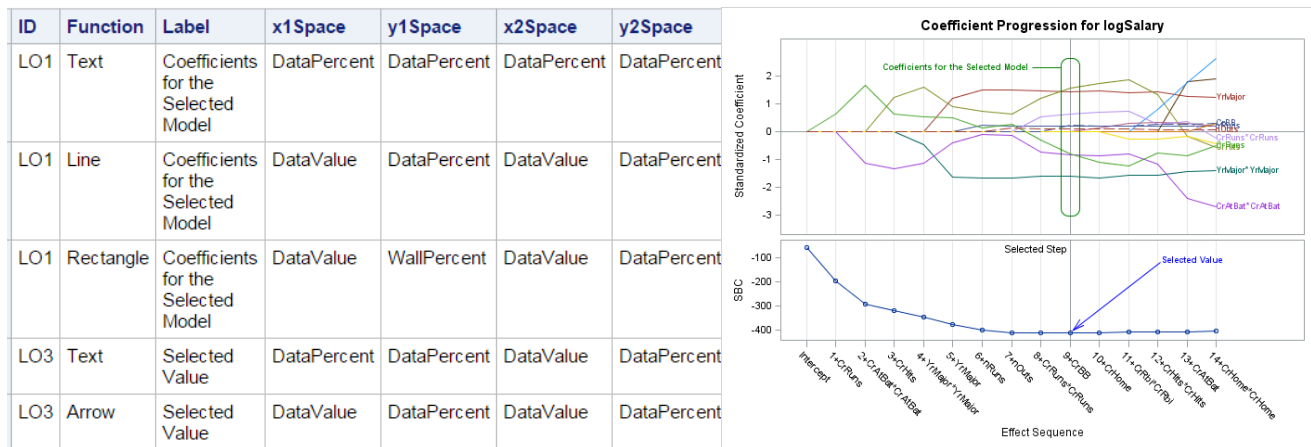
40

Created by:

- RETAIN statement
- Assignment statements

## Annotation Data Set

Obs	ID	Function	Label	x1Space	y1Space	x2Space	y2Space	Direction	Anchor	xC1	xC2	Scale	Width	WidthUnit	CornerRadius	TextSize	TextWeight	LineThickness	DiscreteOffset	LineColor	TextColor	x1	y1	x2	Height
1	LO1	Text	Coefficients for the Selected Model	DataPercent	DataPercent	DataPercent	DataPercent	In	Right			1E-12	100.0	Data	0.8	7	Bold	1.2	-0.3	Green	Green	55	94	.	.
2	LO1	Line	Coefficients for the Selected Model	DataValue	DataPercent	DataValue	DataPercent	In	Right	9+CrBB	8+CrRuns*CrRuns	1E-12	100.0	Data	0.8	7	Bold	1.2	-0.3	Green	Green	94	94	.	.
3	LO1	Rectangle	Coefficients for the Selected Model	DataValue	WallPercent	DataValue	DataPercent	In	BottomLeft	9+CrBB	8+CrRuns*CrRuns	1E-12	0.8	Data	0.8	7	Bold	1.2	-0.3	Green	Green	10	94	80	
4	LO3	Text	Selected Value	DataPercent	DataPercent	DataValue	DataPercent	In	Left	9+CrBB	8+CrRuns*CrRuns	1E-12	100.0	Data	0.8	7	Bold	1.2	-0.3	Green	Blue	86	84	94	80
5	LO3	Arrow	Selected Value	DataValue	DataPercent	DataValue	DataPercent	In	Left	9+CrBB	12+CrHits*CrHits	1E-12	100.0	Data	0.8	7	Bold	1.2	0.1	Blue	Blue	4	83	80	



41

## Create Graph

### Modifications:

- Data object
- Template
- Annotation

```

data _null_;
set dynamics(where=(label1 ne '___NOBS___')) end=eof;
if _n_ = 1 then do;
call execute('proc sgrender data=cp2 sganno=anno');
call execute('template=Stat.GLMSELECT.Graphics.CoefficientPanel;');
call execute('dynamic;');
end;
if cvalue1 ne '' then
call execute(catx(' ', label1, '=',
ifc(n(nvalue1), cvalue1, quote(trim(cvalue1)))));
if eof then call execute('; run;');
run;

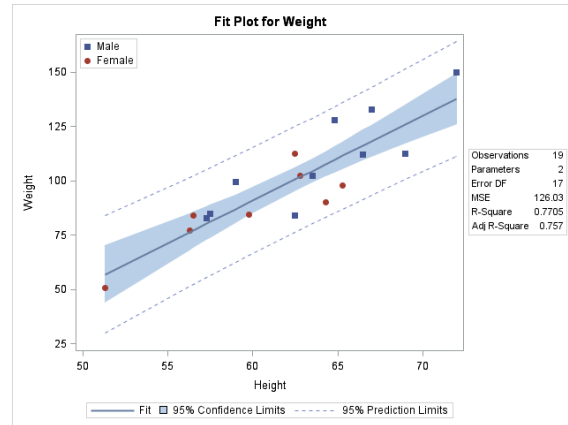
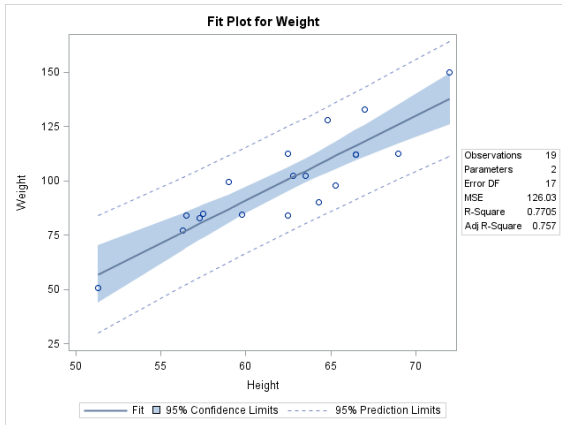
```

- We processed the dynamic variables but did not modify them
- The paper provides macros that make it easy to modify graphs

42

## Conclusions

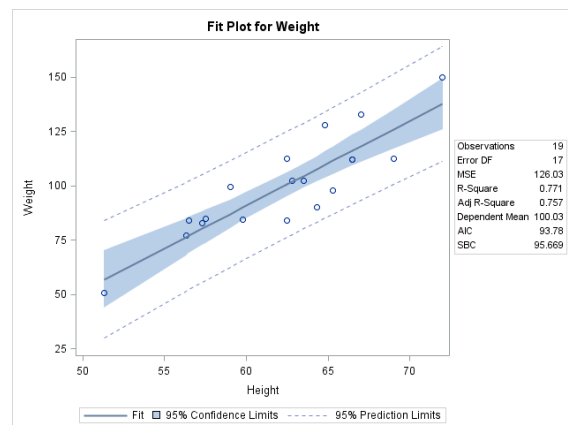
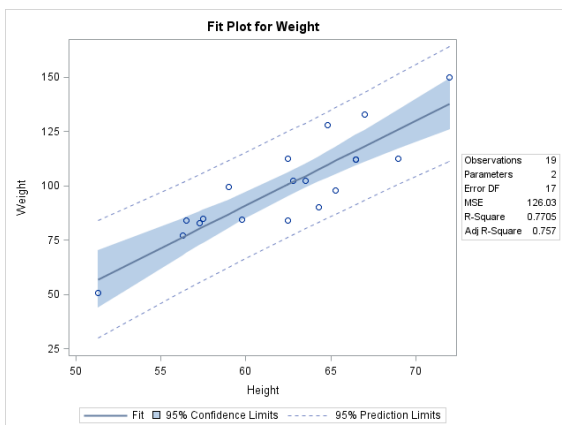
- You can modify graph templates
- Use newer appearance options (or modify styles)
  - In BEGINGRAPH in GTL or STYLEATTRS in SG procedures



43

## Conclusions

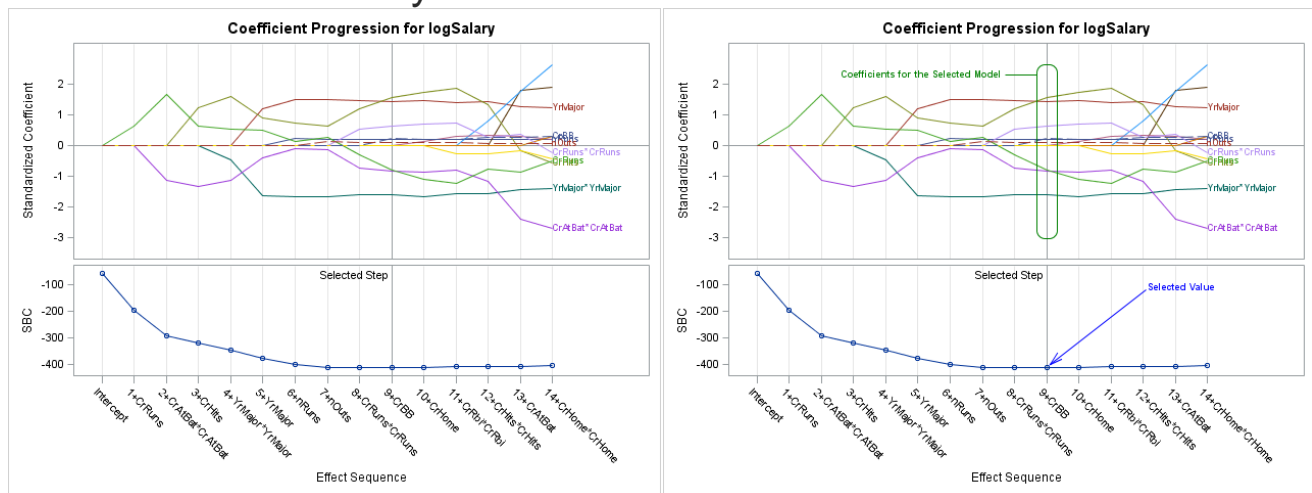
- You can capture, display, output, and modify dynamic variables; then you can recreate the graph



44

## Conclusions

- You can annotate single- and multiple-panel graphs
- You can do as many or as few modifications as you want and in any combination



## For More Information

- **Graphically Speaking SAS Blog**  
<http://blogs.sas.com/content/graphicallyspeaking/>
- **Advanced ODS Graphics Examples**  
<http://support.sas.com/documentation/prod-p/grstat/9.4/en/PDF/odsadv.pdf>
- SAS/STAT introductory chapters
- **Statistical Graphics in SAS**  
<https://support.sas.com/publishing/authors/kuhfeld.html>
- Books by Sanjay Matange and Dan Heath  
<http://support.sas.com/publishing/authors/matange.html>

**Free!**

**Free!**

**Free!**

# Advanced ODS Graphics Examples

## Axes

- Multiple Axes, Offsets, and Drop Lines
- Multiple Axes and Highlighted Points
- Multiple Axes, Axis Alignment, and Many Tick Labels
- Broken Axes
- Multiple Plots with Equated Axes

## Axis Tables

- Axis Table Example Using PROC REG
- Creating a Forest Plot Using PROC SGPLOT
- Stem-and-Leaf Plot with a Box Plot
- Axis Table Example Using PROC AUTOREG

## Annotation

- Replacing Tick Labels
- Understanding the Drawing Spaces
- Displaying Text in a Graph
- Drawing Lines
- Custom Markers, No Markers, and the Data Region
- Displaying Images in a Graph
- Lines, Circles, Ovals, Rectangles, and Other Shapes
- Watermarks
- Rotating Text
- Continuing Text
- Shape and Scale of Arrowheads
- Text Justification and Anchoring
- Selecting the X, X2, Y, and Y2 Axes
- Scaling Images
- Adding Links to Graphs
- SG Annotation Functions, Variables, and Their Values

## Bars, Lines, Curves, and Arrows

- Adverse Events Plot
- Attribute Maps
- Connecting Points with Lines, Arrows, and Curves

## Plots of Labeled Points

- Placing Labels in Scatter Plots
- Changing How Vectors Are Displayed

## Advanced Customization of Graphs That Analytical Procedures Produce

- Changing Dynamic Variables by Using the ODS Document
- Annotating Single-Panel Graphs That Analytical Procedures Produce
- Annotating Multiple-Panel Graphs That Analytical Procedures Produce

47



**SAS® GLOBAL FORUM 2016**  
 IMAGINE. CREATE. INNOVATE.

**Warren F. Kuhfeld**  
**SAS Institute Inc.**  
**Cary, NC, 27513**  
**(919) 531-7922**  
**Warren.Kuhfeld@sas.com**

**support.sas.com/statistics**



#SASGF