

## Preparing Output from Statistical Procedures for Publication, Part 1: PROC REG to APA Format

Vincent DelGobbo, SAS Institute Inc.; Peter L. Flom, Peter Flom Consulting

### ABSTRACT

Many scientific and academic journals require that statistical tables be created in a specific format, with one of the most common formats being that of the American Psychological Association (APA). The APA publishes a substantial guide book to writing and formatting papers, including an extensive section on creating tables (Nichol 2010). However, the output generated by SAS<sup>®</sup> statistical procedures does not match this style. This paper discusses techniques to change the SAS procedure output to match the APA guidelines using the SAS Output Delivery System (ODS).

### INTRODUCTION

The APA provides guidelines for creating tables for many types of statistical analyses that can be performed using SAS. Space does not permit explaining how to produce tables for all types of analyses, and this paper focuses on one of the most common statistical methods, linear regression. The APA format for regression analysis results is shown in Figure 1 (Nichol 2010).

Table X

*Regression Analysis Summary for Parent Variables Predicting Children's Phonemic Awareness*

Variable	<i>B</i>	95% CI	$\beta$	<i>t</i>	<i>p</i>
Parental education level	1.84	[-9.06, 15.78]	.53	3.51	.001
Parental literacy	0.75	[0.79, 2.89]	.18	1.51	.136
Parents' own reading	0.37	[-0.71, 1.44]	.06	0.68	.497
Reading to child	4.60	[1.25, 7.95]	.25	2.75	.008
Socioeconomic status	-0.20	[-2.07, 1.67]	-.03	-0.21	.831

*Note.*  $R^2 = .58$  ( $N = 68$ ,  $p < .001$ ). CI = confidence interval for *B*.

**Figure 1. APA-style Table Layout for Regression Analysis Results**

Although regression analysis can be performed using the REG, GLM, MIXED and GLIMMIX procedures, this paper focuses on creating the table in Figure 1 using PROC REG. You can create this table using word processing software such as Microsoft Word, but copying and pasting the result values from the REG procedure output to Microsoft Word is both laborious and prone to error. Instead, we programmatically create a table that closely matches the APA style and requires minimal editing.

### OUTPUT DELIVERY SYSTEM (ODS) BASICS

ODS is capable of creating files that can be opened by a wide variety of third-party applications, including word processing programs. ODS is the part of Base SAS<sup>®</sup> that enables you to generate different types of output from your procedure code. An *ODS destination* controls the type of output that is generated (HTML, RTF, PDF, and so on). An *ODS style* controls all aspects of the appearance of the output, and Base SAS is shipped with more than 50 different styles.

The general format for using ODS is as follows:

```
ods destination-name file='file-name.ext' style=style-name ... ;
  * Your SAS procedure code here;
ods destination-name close;
```

To generate Rich Text Format (RTF) output from your procedure code using the JOURNAL style supplied by SAS, use the following ODS statements:

```
❶ ods _all_ close;

❷ ods rtf file='file-name.rtf' style=Journal;
  * Your SAS procedure code here;

❸ ods rtf close;
```

The first ODS statement (❶) closes all destinations that are open, because we want to generate only RTF output.

The second ODS statement (❷) uses the RTF destination to generate the RTF output and then stores the output in a file. The STYLE option controls the appearance of the output, such as the font and color scheme. To see a list of ODS styles that are available for use at your site, submit the following SAS code:

```
ods _all_ close;
ods listing;
proc template;
  list styles;
run; quit;
```

To find the SAS code that generates sample output for the ODS styles available on your system, click the **Full Code** tab in SAS Sample 36900 (SAS Institute Inc. 2009).

The third ODS statement (❸) closes the RTF destination and releases the RTF file so that it can be opened with another application.

You create other types of output by changing the destination name in the second (❷) and third (❸) ODS statements.

## INITIAL SAS CODE

We use the BASEBALL SAS table and PROC REG to regress salary on number of hits, runs, runs batted in, walks in the past season, and the number of lifetime hits. First we modify the BASEBALL table by dividing the independent variables by 100 to change the magnitudes of the parameter estimates and confidence limits. This action results in fewer zeros in the final output without changing the meaning of the results. The ODS RTF destination and the ODS JOURNAL style create an RTF file that you can incorporate into word processing programs.

```
data baseball;
  set sashelp.baseball;
  hitsc  = nhits/100;
  runsc  = nruns/100;
  rbic   = nrbi/100;
  bbc    = nbb/100;
  crhitc = crhits/100;
run;
```

```

* Close all open ODS destinations;

ods _all_ close;

options nocenter nodate nonumber;

title1 'Table [your table number here]';
title2 '[your table title here]';

ods rtf file='APA.rtf' bodytitle_aux style=Journal;

proc reg data=baseball plots=none;
  id name team league;
  model logSalary = hitsc runsc rbic bbc yrmajor crhitsc/stb clb;
  label hitsc = 'Hits in 1986, in hundreds'
        runsc = 'Runs in 1986, in hundreds'
        rbic = 'RBI in 1986, in hundreds'
        bbc = 'BB 1986, in hundreds'
        crhitsc = 'Career hits, in hundreds';
run; quit;

ods rtf close;

```

Because the APA guidelines require standardized parameter estimates and their corresponding confidence limits, we use the STB and CLB options in the MODEL statement.

The results of executing this code are shown in [Figure 2](#).

The Parameter Estimates table contains all the information that is needed to create the table in [Figure 1](#) when we omit the first row of data, and also omit the columns labeled **Variable**, **DF**, and **Standard Error**. The footnote is constructed by using the R-square value (0.5867) from the Fit Statistics table, the number of observations used by PROC REG (263) from the Number of Observations table, and the  $p$ -value ( $<.001$ ) from the ANOVA table.

The remainder of this paper describes the following general steps to construct the output in [Figure 1](#):

1. Create SAS tables that contain results from the execution of PROC REG.
2. Use data from these SAS tables to create macro variables used in the footnote.
3. Reformat the Parameter Estimates table to more closely match the desired output.
4. Create an ODS style that matches the desired output.
5. Print the reformatted Parameter Estimates table using the PRINT procedure and ODS style overrides.
6. Copy-and-paste the SAS output into your word processing program.

## LISTING THE ODS OUTPUT OBJECTS CREATED BY A SAS PROCEDURE

ODS output objects contain data resulting from the execution of SAS procedures. We use the following ODS TRACE statement (SAS Institute Inc. [2014d](#)) to list the output objects created by PROC REG:

```

ods trace on;

proc reg data=baseball plots=none;
  id ... ;
  model ... ;
  label ... ;
run; quit;

ods trace off;

```

The output shown in [Figure 3](#) is displayed in the SAS log after running our code.

**Table [your table number here]**  
**[your table title here]**

**The REG Procedure**  
**Model: MODEL1**  
**Dependent Variable: logSalary Log Salary**

---

Number of Observations Read	322	Number of Observations table
Number of Observations Used	263	
Number of Observations with Missing Values	59	

---

<i>Analysis of Variance</i>						ANOVA table
<i>Source</i>	<i>DF</i>	<i>Sum of Squares</i>	<i>Mean Square</i>	<i>F Value</i>	<i>Pr &gt; F</i>	
Model	6	121.53052	20.25509	60.56	<.0001	
Error	256	85.62322	0.33447			
Corrected Total	262	207.15373				

---

Root MSE	0.57833	R-Square	0.5867	Fit Statistics table
Dependent Mean	5.92722	Adj R-Sq	0.5770	
Coeff Var	9.75719			

---

<i>Parameter Estimates</i>										Parameter Estimates table
<i>Variable</i>	<i>Label</i>	<i>DF</i>	<i>Parameter Estimate</i>	<i>Standard Error</i>	<i>t Value</i>	<i>Pr &gt;  t </i>	<i>Standardized Estimate</i>	<i>95% Confidence Limits</i>		
Intercept	Intercept	1	4.14614	0.13612	30.46	<.0001	0	3.87808	4.41420	
hitsc	Hits in 1986, in hundreds	1	0.66260	0.21010	3.15	0.0018	0.32782	0.24886	1.07634	
runsc	Runs in 1986, in hundreds	1	0.01989	0.39849	0.05	0.9602	0.00561	-0.76484	0.80462	
rbic	RBI in 1986, in hundreds	1	0.12522	0.23506	0.53	0.5947	0.03621	-0.33768	0.58812	
bbc	BB 1986, in hundreds	1	0.67209	0.23931	2.81	0.0054	0.16256	0.20083	1.14334	
YrMajor	Years in the Major Leagues	1	0.07108	0.01925	3.69	0.0003	0.38293	0.03317	0.10899	
crhitsc	Career hits, in hundreds	1	0.02391	0.01457	1.64	0.1020	0.17449	-0.00478	0.05260	

**Figure 2. Initial REG Procedure Output**

```

Output Added:
-----
Name:          NObs
Label:         Number of Observations
Template:      Stat.Reg.NObs
Path:          Reg.MODEL1.Fit.logSalary.NObs
-----

Output Added:
-----
Name:          ANOVA
Label:         Analysis of Variance
Template:      Stat.REG.ANOVA
Path:          Reg.MODEL1.Fit.logSalary.ANOVA
-----

Output Added:
-----
Name:          FitStatistics
Label:         Fit Statistics
Template:      Stat.REG.FitStatistics
Path:          Reg.MODEL1.Fit.logSalary.FitStatistics
-----

Output Added:
-----
Name:          ParameterEstimates
Label:         Parameter Estimates
Template:      Stat.REG.ParameterEstimates
Path:          Reg.MODEL1.Fit.logSalary.ParameterEstimates
-----

```

**Figure 3. Output Objects Created by the REG Procedure**

These four output objects (NObs, ANOVA, FitStatistics, and ParameterEstimates) contain all the data that is used to create the four tables shown in [Figure 2](#). The output objects can be saved as SAS tables, and then the data can be used in subsequent steps in your SAS code.

## CREATING SAS TABLES FROM ODS OUTPUT OBJECTS

The ODS OUTPUT statement creates SAS tables from ODS output objects (SAS Institute Inc. [2014c](#)). Here is the general syntax:

```
ods output output-object-name1 = SAS-table-name1
           output-object-name2 = SAS-table-name2 ... ;

* Your SAS procedure code here;
```

Because we need to manipulate data from all the output objects in [Figure 3](#), our revised code is as follows:

```
ods _all_ close;

ods output Nobs=NumObs
           ANOVA=Anova
           FitStatistics=FitStats
           ParameterEstimates=ParamEst;

options nocenter nodate nonumber;

title1 'Table [your table number here]';
title2 '[your table title here]';

* No ODS RTF output desired;

proc reg data=baseball plots=none;
  id    ... ;
  model ... ;
  label ... ;
run; quit;
```

The ODS RTF statement is removed because we do not need the standard, formatted results from PROC REG. Instead, we reformat the data in the PARAMEST SAS table, display it using the PRINT procedure. We use data from the other SAS tables to construct the footnote.

## CREATING SAS MACRO VARIABLES FOR THE FOOTNOTE

We use a footnote to display explanatory information following the parameter estimates. The required information is retrieved from the NUMOBS, ANOVA, and FITSTATS SAS tables, and then stored in global macro variables that are used in the FOOTNOTE statement.

Figure 4 shows the NUMOBS, ANOVA, and FITSTATS SAS tables, with the values of interest circled.

Obs	Model	Dependent	Label	N	NObsRead	NObsUsed	SumFreqsRead	SumFreqsUsed	NObsMiss
1	MODEL1	logSalary	Number of Observations Read	322	322	263	322	263	59
2	MODEL1	logSalary	Number of Observations Used	263	322	263	322	263	59
3	MODEL1	logSalary	Number of Observations with Missing Values	59	322	263	322	263	59

Obs	Model	Dependent	Source	DF	SS	MS	FValue	ProbF
1	MODEL1	logSalary	Model	6	121.53052	20.25509	60.56	<.0001
2	MODEL1	logSalary	Error	256	85.62322	0.33447	—	—
3	MODEL1	logSalary	Corrected Total	262	207.15373	—	—	—

Obs	Model	Dependent	Label1	cValue1	nValue1	Label2	cValue2	nValue2
1	MODEL1	logSalary	Root MSE	0.57833	0.578330	R-Square	0.5867	0.586668
2	MODEL1	logSalary	Dependent Mean	5.92722	5.927222	Adj R-Sq	0.5770	0.576981
3	MODEL1	logSalary	Coeff Var	9.75719	9.757187			0

**Figure 4. The NUMOBS, ANOVA, and FITSTATS SAS Tables**

The following SYMPUTX routine creates the global macro variables (SAS Institute Inc. 2014a):

```
ods output ... ;

proc reg ... ; run; quit;

* Create macro variables used in the footnote;

data _null_;
  set NumObs;
  if (_N_ eq 1);
  call symputx('N_OBS_USED', put(NObsUsed, best.));
run;

data _null_;
  set Anova;
  if (_N_ eq 1);
  call symputx('PROBF', put(ProbF, pvalue6.3));
run;

data _null_;
  set FitStats;
  if (_N_ eq 1);
  call symputx('R2', put(nValue2, 10.2));
run;
```

The macro variables are now used in the FOOTNOTE statement, as shown:

```
footnote "Note. R2 = &R2 "
        "(N = &N_OBS_USED, "
        "p &PROBF). "
        'CI = confidence interval for B.';
```

## REFORMATTING THE PARAMETER ESTIMATES TABLE

Figure 5 shows the PARAMEST SAS table, with the columns of interest circled.

Obs	Model	Dependent	Variable	DF	Estimate	StdE	tValue	Probt	StandardizedEst	LowerCL	UpperCL	Label
1	MODEL1	logSalary	Intercept	1	4.14614	0.13612	30.46	<.0001	0	-3.87808	4.41420	Intercept
2	MODEL1	logSalary	hitsc	1	0.66260	0.21010	3.15	0.0018	0.32782	0.24886	1.07634	Hits in 1986, in hundreds
3	MODEL1	logSalary	runsc	1	0.01989	0.39849	0.05	0.9602	0.00561	-0.76484	0.80462	Runs in 1986, in hundreds
4	MODEL1	logSalary	rbic	1	0.12522	0.23506	0.53	0.5947	0.03621	-0.33768	0.58812	RBI in 1986, in hundreds
5	MODEL1	logSalary	bbc	1	0.67209	0.23931	2.81	0.0054	0.16256	0.20083	1.14334	BB 1986, in hundreds
6	MODEL1	logSalary	YrMajor	1	0.07108	0.01925	3.69	0.0003	0.38293	0.03317	0.10899	Years in the Major Leagues
7	MODEL1	logSalary	crhitsc	1	0.02391	0.01457	1.64	0.1020	0.17449	-0.00478	0.05260	Career hits, in hundreds

**Figure 5. The PARAMEST SAS Table**

We use a subsetting IF statement to omit the first row of data because the intercept is not part of the APA-formatted table in [Figure 1](#).

If a column in the input table for PROC REG does not have a column label, then the column name is used for **Label**. The APA guidelines require that the lower and upper confidence levels are displayed as a range in a single column (Figure 1). We create the CIRANGE variable to display the two values enclosed in brackets.

```
* Create the final table;

data Final;
  set ParamEst;
  keep Label Estimate CIRange StandardizedEst tValue Probt;
  length CIRange $15;

  if (upcase(Variable) ne 'INTERCEPT');

  if (compress(Label) eq '') then do;
    Label = Variable;
  end;

  CIRange = '[' || strip(put(LowerCL, 10.2)) ||
            ', ' || strip(put(UpperCL, 10.2)) || ']';

run;
```

The FINAL SAS table can be displayed using the PRINT procedure.

## INTERMEDIATE RESULTS

Here is our SAS code after implementing the modifications and including the PRINT procedure code:

```
data baseball; set sashelp.baseball; ...; run;

ods _all_ close;

ods output Nobs=NumObs
           ANOVA=Anova
           FitStatistics=FitStats
           ParameterEstimates=ParamEst;
```



```

options ... ;

title1 'Table [your table number here]';
title2 '[your table title here]';

proc reg data=baseball plots=none;
  id    ... ;
  model ... ;
  label ... ;
run; quit;

ods rtf file='APA.rtf' bodytitle_aux style=Journal;

* Create macro variables used in the footnote;

data _null_; set NumObs; ... run;

data _null_; set Anova; ... run;

data _null_; set FitStats; ... run;

data Final; set ParamEst; ... run;

footnote "Note. R2 = &R2 "
        "(N = &N_OBS_USED, "
        "p &PROBF). "
        'CI = confidence interval for B.';

proc print data=Final noobs label;
  var Label Estimate CIRange StandardizedEst tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label          = 'Variable'
         Estimate      = 'B'
         CIRange       = '95% CI'
         StandardizedEst = 'b'
         tValue        = 't'
         Probt         = 'p';
run; quit;

ods rtf close;

```

The results of executing this code are shown in [Figure 6](#). The format of these results does not exactly match [Figure 1](#), the differences being:

1. The font for the text displaying the table number should be normal weight, non-italic.
2. The table title text should be displayed using a normal weight font.
3. Column headings are not centered.
4. The letter **b** is used for a column heading instead of lower case beta ( $\beta$ ).
5. The **Variable**, **95% CI**, and **b** column headings should not be displayed using an italic font.
6. The footnote text should be displayed using a normal weight font, with only some text in italics.
7. The R2 should be  $R^2$ .

**Table [your table number here]  
[your table title here]**

<i>Variable</i>	<i>B</i>	<i>95% CI</i>	<i>b</i>	<i>t</i>	<i>p</i>
Hits in 1986, in hundreds	0.66	[0.25, 1.08]	0.33	3.15	0.002
Runs in 1986, in hundreds	0.02	[-0.76, 0.80]	0.01	0.05	0.960
RBI in 1986, in hundreds	0.13	[-0.34, 0.59]	0.04	0.53	0.595
BB 1986, in hundreds	0.67	[0.20, 1.14]	0.16	2.81	0.005
Years in the Major Leagues	0.07	[0.03, 0.11]	0.38	3.69	<.001
Career hits, in hundreds	0.02	[-0.00, 0.05]	0.17	1.64	0.102

**Note.**  $R^2 = 0.59$  ( $N = 263$ ,  $p < .001$ ). *CI* = confidence interval for *B*.

**Figure 6. Intermediate Results in RTF Format**

Note that the **-0.00** in the lower CI for career hits indicates that the true result is between 0.000 and -0.0049, and is rounded to 0. Thus, it agrees with the *p*-value being above 0.05.

You can stop here and copy and paste these results into your word processing program and then correct the differences. Or you can continue to the following sections to learn how to use ODS styles and style overrides to make the changes, eliminating the need to hand-edit the output.

## CREATING AND USING AN ODS STYLE

The JOURNAL style produces results that are similar to those of [Figure 1](#), but some changes to this style are needed to address the issues listed in the previous section.

A style contains *style elements*, each of which controls a particular part of the output. For example, a style element named HEADER controls the appearance of column headings. Style elements consist of collections of *style attributes*, such as the background color and font size. We need to change the attributes of some style elements to match the APA guidelines.

## DETERMINING THE STYLE ELEMENTS AND ATTRIBUTES IN YOUR OUTPUT

We must determine the style elements and attributes used by the JOURNAL style, and then create a new style based on JOURNAL and named APA, with the appropriate attributes.

Use the TAGSETS.STYLE\_POPUP destination to determine the attributes of style elements. The tagset creates an HTML file that, when viewed using the SAS Results window or the Microsoft Internet Explorer Web browser, displays style element information in popup windows (SAS Institute Inc. [2014g](#)).

```
ods _all_ close;

ods tagsets.style_popup file='APA.htm' style=Journal;

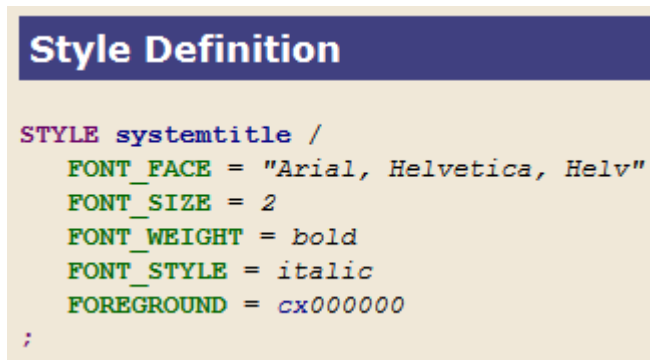
title1 'Table [your table number here]';
title2 '[your table title here]';

footnote "Note. R2 = &R2 "
        "(N = &N_OBS_USED, "
        "p &PROBF). "
        'CI = confidence interval for B.';

proc print data=Final noobs label;
  var Label Estimate CIRange StandardizedEst tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label          = 'Variable'
         Estimate      = 'B'
         CIRange       = '95% CI'
         StandardizedEst = 'b'
         tValue        = 't'
         Probt         = 'p';
run; quit;

ods tagsets.style_popup close;
```

Click on **Table [your table number here]** to display the name and attributes of the style element used for this title line (Figure 7).



```
Style Definition

STYLE systemtitle /
  FONT_FACE = "Arial, Helvetica, Helv"
  FONT_SIZE = 2
  FONT_WEIGHT = bold
  FONT_STYLE = italic
  FOREGROUND = cx000000
;
```

**Figure 7. The SYSTEMTITLE Style Element and Its Attributes**

The SYSTEMTITLE style element of our new style must specify normal weight, non-italic text to match the APA guidelines. You accomplish this by specifying **medium** for the **FONT\_WEIGHT** attribute, and **roman** for the **FONT\_STYLE** attribute. The ODS documentation lists style attributes and their values (SAS Institute Inc. [2014e](#)).

Table 1 lists the existing style elements that must be changed to match the results of [Figure 1](#).

Style Element	Used by
SYSTEMTITLE	TITLE1 statement
SYSTEMTITLE2	TITLE2 statement
SYSTEMFOOTER	FOOTNOTE statement
HEADER	All column headings

**Table 1. Existing Style Elements and Their Usage**

We also need to create two new style elements to control the appearance of the **Variable, 95% CI**, and **b** column headings of [Figure 6](#).

### CREATING THE APA STYLE

The TEMPLATE procedure supplied with Base SAS creates and modifies ODS styles (SAS Institute Inc. [2014f](#)). Although you can use the TEMPLATE procedure to create a new style without copying an existing style, it's usually easier to copy an existing style that is close to what you want, and then make modifications to the copy.

The code below creates the user-defined APA style by copying the JOURNAL style supplied by SAS:

```
proc template;
  define style styles.APA;
    parent = styles.Journal;
  end;
run; quit;
```

Because this TEMPLATE procedure code does not contain statements that change any style elements, the APA style is an exact copy of the JOURNAL style. That is, the APA style *inherits* all of the style elements and attributes of the *parent* style, JOURNAL.

The PROC TEMPLATE CLASS statement changes the attributes of an existing style element, and the STYLE statement creates new style elements (SAS Institute Inc. [2014f](#)). We use the CLASS statement to modify the style elements shown in Table 1 and the STYLE statement to create two new style elements named HEADER\_ROMAN and HEADER\_SYMBOL.

Here is the CLASS statement general format:

```
class existing-style-element-name /
  style-attribute-specifications;
```

Because the SYSTEMTITLE style element of our new style must have normal weight, non-italic text, we use this CLASS statement:

```
class systemtitle /
  font_weight = medium
  font_style = roman;
```

All other attributes of the SYSTEMTITLE style element remain unchanged from their inherited values.

Table 2 lists the style attributes used in the CLASS statement for the existing style elements to match the APA guidelines.

Style Element	Used by	Style Attribute Specification
SYSTEMTITLE	TITLE1 statement	font_weight = medium font_style = roman
SYSTEMTITLE2	TITLE2 statement	font_style = italic
SYSTEMFOOTER	FOOTNOTE statement	font_weight = medium font_style = roman
HEADER	All column headings	just = center font_style = italic

**Table 2. Modified APA Style Elements and Their Attributes**

New style elements are created using the STYLE statement, which follows this general format:

```
style new-style-element-name from existing-style-element-name /
  style-attribute-specifications;
```

New style elements have all of the style attributes of the existing element, plus any attributes specified by **style-attribute-specifications**.

We need to create two new style elements, HEADER\_ROMAN and HEADER\_SYMBOL, to control the appearance of some column headings (Table 3). Both style elements should have the same attributes as the existing HEADER style element, except for the differences noted in Table 3.

Style Element	Used by	Style Attribute Specification
HEADER_ROMAN	<b>Label &amp; 95% CI</b> column headings	font_style = roman
HEADER_SYMBOL	<b>b</b> column heading	font_style = roman font_face = Symbol

**Table 3. New APA Style Elements and Their Attributes**

The HEADER\_SYMBOL style element is applied to the column heading of the **b** column in [Figure 6](#) because a lower case b is rendered as β in the Symbol font. Refer to [Table A - 1](#) in the appendix for additional information about the Symbol font.

Here is the complete code to create the APA style:

```

*;
* Create the APA ODS style and store in WORK. Use
* a permanent library in production code.
*;

❶ ods path (prepend) work.tmplmst(update);

proc template;
❷ define style styles.APA;
  parent = styles.Journal;

❸ class systemtitle /
  font_weight=medium
  font_style=roman;
```

```

class systemtitle2 /
  font_weight=medium
  font_style=italic;

class systemfooter /
  font_weight=medium
  font_style=roman;

class header /
  just=center
  font_style=italic;

❷ style header_roman from header /
  font_style=roman;

style header_symbol from header /
  font_style=roman
  font_face=Symbol;
end;
run; quit;

```

The ODS PATH statement (❶) specifies the locations of, and the order in which to search for, ODS tagsets and styles. Notice that the access mode for **work.tmplmst** is specified as "update" and it is first in the search path as a result of the PREPEND option. Because ODS searches the path in the order given, and the access mode for **work.tmplmst** is "update", PROC TEMPLATE stores the style in a file named "tmplmst.sas7bitm" in the directory that is associated with the Work library.

Submit this code to display the ODS search path:

```
ods path show;
```

The APA style is created, and all of the style attributes of the parent JOURNAL style are inherited by APA (❷). The CLASS statements (❸) change the inherited attributes of some existing style elements and the STYLE statements (❹) create new style elements.

The APA style can now be used with any SAS procedure.

Although you can temporarily store the style in the Work library, it is more efficient to create it once in a separate program, storing it in a permanent library. You can reference it in other SAS programs using this code:

```

libname mystyle 'path-to-stored-style' access=read;

ods path (prepend) mystyle.tmplmst(read);

```

## USING THE APA STYLE

Starting with the code from the [Intermediate Results](#) section, use the APA style instead of JOURNAL:

```

data baseball; set sashelp.baseball; ...; run;

ods _all_ close;

ods output ... ;

options ... ;

```

```

title1 ... ;
title2 ... ;

proc reg; ... ; run; quit;

ods rtf file='APA.rtf' bodytitle_aux style=APA;

* Create macro variables used in the footnote;

data _null_; set NumObs; ... run;

data _null_; set Anova; ... run;

data _null_; set FitStats; ... run;

data Final; set ParamEst; ... run;

footnote "Note. R2 = &R2 "
        "(N = &N_OBS_USED, "
        "p &PROBF). "
        'CI = confidence interval for B.';

proc print data=Final noobs label;
  var Label Estimate CIRange StandardizedEst tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label      = 'Variable'
         Estimate   = 'B'
         CIRange    = '95% CI'
         StandardizedEst = 'b'
         tValue     = 't'
         Probt      = 'p';
run; quit;

ods rtf close;

```

All column heading text in [Figure 8](#) is displayed with an italic font because `font_style = italic` is specified in the HEADER style element definition ([Table 2](#)). Changing the HEADER style element, as we have done in the APA style, affects all column headings. Thus, you should carefully consider the impact of changing style elements that are part of a style supplied by SAS.

Table [your table number here]  
 [your table title here]

<i>Variable</i>	<i>B</i>	<i>95% CI</i>	<i>b</i>	<i>t</i>	<i>p</i>
Hits in 1986, in hundreds	0.66	[0.25, 1.08]	0.33	3.15	0.002
Runs in 1986, in hundreds	0.02	[-0.76, 0.80]	0.01	0.05	0.960
RBI in 1986, in hundreds	0.13	[-0.34, 0.59]	0.04	0.53	0.595
BB 1986, in hundreds	0.67	[0.20, 1.14]	0.16	2.81	0.005
Years in the Major Leagues	0.07	[0.03, 0.11]	0.38	3.69	<.001
Career hits, in hundreds	0.02	[-0.00, 0.05]	0.17	1.64	0.102

Note. R2 = 0.59 (N = 263, p <.001). CI = confidence interval for B.

### Figure 8. Results Using the APA Style

The HEADER\_ROMAN and HEADER\_SYMBOL style elements are not used for column headings because they are user-defined elements unknown to ODS. We must use an ODS style override to instruct ODS to use the HEADER\_ROMAN style element for the **Variable** and **95% CI** column headings, and HEADER\_SYMBOL for the **b** column heading.

You can alter the appearance specific parts of your PRINT, REPORT, and TABULATE procedure output by using style overrides. These specific parts of your SAS output are called *locations*. [Figure 9](#) shows the locations of the PRINT procedure output (SAS Institute Inc. 2008).

The HEADER location controls the appearance of column headings. By default ODS applies the HEADER style element to this location.

Style overrides can be specified in several ways, the two most common formats being:

- ❶ `style(location)=[style-attribute-name1=value1  
style-attribute-name2=value2 ...]`
- ❷ `style(location)=style-element-name`

The first format (❶) uses individual style attributes defined inline. For example, the following code alters some attributes of the column headings:

```
style(header)=[background=white font_size=10pt just=center]
```

Although this is the most commonly used format, it has some disadvantages. To use the same style override for different variables, you must apply it in multiple places, making your SAS code harder to read and maintain. And, if you want to use the style overrides in other SAS programs, you must copy the list of attribute name/value pairs to the new code.



table		
obsheader	header	header
obs	column	column
obs	column	column
obs	column	column
obs	column	column
obs	column	column
bylabel	total	total
grandtotal	grandtotal	grandtotal
n		

**Figure 9. Style Locations for the PRINT Procedure**

The second format (②) overcomes these problems by referencing a style element. Using this format involves creating a new style element as we have done with the APA style, setting the style attributes within the element, and then using the *style element* name in your style override. This results in code that is easier to read, maintain, and re-use.

To apply style overrides to the **Variable**, **95% CI**, and **b** column headings of [Figure 8](#), we isolate those columns by using multiple VAR statements. Then we apply the style overrides using the STYLE option:

```
proc print data=Final noobs label;
  var Label / style(header)=header_roman;
  var Estimate;
  var CIRange / style(header)=header_roman;
  var StandardizedEst / style(header)=header_symbol;
  var tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label          = 'Variable'
         Estimate      = 'B'
         CIRange       = '95% CI'
         StandardizedEst = 'b'
         tValue        = 't'
         Probt         = 'p';
run; quit;
```

A non-italic font is used for the text of the **Variable** and **95% CI** column headings, and  $\beta$  is displayed for the **b** column heading text:

Table [your table number here]  
[your table title here]

Variable	<i>B</i>	95% CI	$\beta$	<i>t</i>	<i>p</i>
Hits in 1986, in hundreds	0.66	[0.25, 1.08]	0.33	3.15	0.002
Runs in 1986, in hundreds	0.02	[-0.76, 0.80]	0.01	0.05	0.960
RBI in 1986, in hundreds	0.13	[-0.34, 0.59]	0.04	0.53	0.595
BB 1986, in hundreds	0.67	[0.20, 1.14]	0.16	2.81	0.005
Years in the Major Leagues	0.07	[0.03, 0.11]	0.38	3.69	<.001
Career hits, in hundreds	0.02	[-0.00, 0.05]	0.17	1.64	0.102

Note.  $R^2 = 0.59$  ( $N = 263$ ,  $p < .001$ ). CI = confidence interval for *B*.

**Figure 10. Results Using the APA Style and Style Overrides**

Our code addresses all of the issues listed in the [Intermediate Results](#) section, except that some text in the footnote should be displayed in an italic font.

### APPLYING ODS INLINE FORMATTING TO THE FOOTNOTE TEXT

The text strings **Note.**, **N**, **p**, and **B** in our footnote must be displayed using an italic font, and **R<sup>2</sup>** rendered as  $R^2$ , to conform to the APA guidelines. ODS inline formatting is used to make all these changes by inserting formatting instructions in the footnote text (SAS Institute Inc. 2014b).

The ODS ESCAPECHAR statement defines a special character to indicate that inline formatting operations follow. We use ^ for an escape character because it is not used in our footnote text:

```
ods escapechar '^';
```

Here is the general syntax for applying style attributes to your text string:

```
^{style [style-attribute-specifications] your-text-here}
```

Inline formatting instructions start with the escape character and are enclosed within braces. We apply the FONT\_STYLE attribute within brackets to display portions of the footnote in italic text:

```
footnote " ^{style [font_style=italic]Note. R2} = &R2 "
         " (^{style [font_style=italic]N} = &N_OBS_USED, "
         " ^{style [font_style=italic]p} &PROBF) . "
         'CI = confidence interval for ^{style [font_style=italic]B} .';
```

Inline style functions can be used in addition to style attributes. Here is the general syntax:

```
^{function-name function-arguments}
```

The SUPER function accepts one argument, the text string to superscript. Our final footnote statement is:

```
footnote " ^{style [font_style=italic]Note. R^{super 2}} = &R2 "
        " (^{style [font_style=italic]N} = &N_OBS_USED, "
        " ^{style [font_style=italic]p} &PROBF). "
        'CI = confidence interval for ^{style [font_style=italic]B}.';
```

The results of running the final code (Figure 11) match the APA guidelines shown in [Figure 1](#):

Table [your table number here]  
[your table title here]

Variable	<i>B</i>	95% CI	$\beta$	<i>t</i>	<i>p</i>
Hits in 1986, in hundreds	0.66	[0.25, 1.08]	0.33	3.15	0.002
Runs in 1986, in hundreds	0.02	[-0.76, 0.80]	0.01	0.05	0.960
RBI in 1986, in hundreds	0.13	[-0.34, 0.59]	0.04	0.53	0.595
BB 1986, in hundreds	0.67	[0.20, 1.14]	0.16	2.81	0.005
Years in the Major Leagues	0.07	[0.03, 0.11]	0.38	3.69	<.001
Career hits, in hundreds	0.02	[-0.00, 0.05]	0.17	1.64	0.102

Note.  $R^2 = 0.59$  ( $N = 263$ ,  $p < .001$ ). CI = confidence interval for *B*.

**Figure 11. Final Results in RTF Format**

You can copy-and-paste these results into your word processing program. A copy of the complete code to create the output in Figure 11 is in the section [The Final SAS Code](#).

## ADAPTING THE TECHNIQUE FOR OTHER USES OF PROC REG

Although the code presented in the section [The Final SAS Code](#) is lengthy, it is easy to change it to meet your needs. Omit the code to change the magnitude of values in the BASEBALL SAS table and then find the comment in the code and follow the instructions.

This code fragment illustrates what you should change to adapt this technique for your PROC REG code:

```
*;
* Specify the title text, PROC REG code, and your path and filename.
* No other changes are needed.
*;

options nocenter nodate nonumber;

title1 'Table [your table number here]';
title2 '[your table title here]';
```

```

proc reg data=baseball plots=none;
  id    name team league;
  model logSalary = hitsc runsc rbic bbc yrmajor crhitsc/stb clb;
  label hitsc    = 'Hits in 1986, in hundreds'
        runsc    = 'Runs in 1986, in hundreds'
        rbic     = 'RBI in 1986, in hundreds'
        bbc      = 'BB 1986, in hundreds'
        crhitsc  = 'Career hits, in hundreds';
run; quit;

```

```
ods rtf file='APA.rtf' bodytitle_aux style=APA;
```

```

*;
* Do not edit anything below here, unless you changed the
* ODS destination.  If so, then be sure to close the destination
* at the end.
*;

```

## THE FINAL SAS CODE

Here is the complete SAS code used to create the output in [Figure 11](#):

```

* Change the magnitude of some values;

data baseball;
  set sashelp.baseball;
  hitsc    = nhits/100;
  runsc    = nruns/100;
  rbic     = nrbi/100;
  bbc      = nbb/100;
  crhitsc  = crhits/100;
run;

* Close all open ODS destinations;

ods _all_ close;

*;
* Create the APA ODS style and store in WORK.  Use
* a permanent library in production code.
*;

ods path (prepend) work.tmplmst(update);

proc template;
  define style styles.APA;
    parent = styles.Journal;

    class systemtitle /
      font_weight=medium
      font_style=roman;

    class systemtitle2 /
      font_weight=medium
      font_style=italic;
  end;
run;

```

```

class systemfooter /
  font_weight=medium
  font_style=roman;

class header /
  just=center
  font_style=italic;

style header_roman from header /
  font_style=roman;

style header_symbol from header /
  font_style=roman
  font_face=Symbol;
end;
run; quit;

* Save the ODS output objects as SAS tables;

ods output Nobs=NumObs
           ANOVA=Anova
           FitStatistics=FitStats
           ParameterEstimates=ParamEst;

*;
* Specify the title text, PROC REG code, and your path and filename.
* No other changes are needed.
*;

options nocenter nodate nonumber;

title1 'Table [your table number here]';
title2 '[your table title here]';

proc reg data=baseball plots=none;
  id name team league;
  model logSalary = hitsc runsc rbic bbc yrmajor crhitsc/stb clb;
  label hitsc = 'Hits in 1986, in hundreds'
        runsc = 'Runs in 1986, in hundreds'
        rbic = 'RBI in 1986, in hundreds'
        bbc = 'BB 1986, in hundreds'
        crhitsc = 'Career hits, in hundreds';
run; quit;

ods rtf file='APA.rtf' bodytitle_aux style=APA;

```

```

*;
* Do not edit anything below here, unless you changed the
* ODS destination. If so, then be sure to close the destination
* at the end.
*;

* Create macro variables used in the footnote;

data _null_;
  set NumObs;
  if (_N_ eq 1);
  call symputx('N_OBS_USED', put(NObsUsed, best.));
run;

data _null_;
  set Anova;
  if (_N_ eq 1);
  call symputx('PROBF', put(ProbF, pvalue6.3));
run;

data _null_;
  set FitStats;
  if (_N_ eq 1);
  call symputx('R2', put(nValue2, 10.2));
run;

* Create the final table;

data Final;
  set ParamEst;
  keep Label Estimate CIRange StandardizedEst tValue Probt;
  if (upcase(Variable) ne 'INTERCEPT');

  if (compress(Label) eq '') then do;
    Label = Variable;
  end;

  CIRange = '[' || strip(put(LowerCL, 10.2)) ||
            ', ' || strip(put(UpperCL, 10.2)) || ']';
run;

* Set the ODS escape character for inline formatting;

ods escapechar '^';

* Apply inline formatting to control the footnote appearance;

footnote "Note. R2 = &R2 "
         "(N = &N_OBS_USED, "
         "p = &PROBF). ""
         "CI = confidence interval for B.";

```

```

* Use ODS style overrides to control procedure output appearance;

proc print data=Final noobs label;
  var Label / style(header)=header_roman;
  var Estimate;
  var CIRange / style(header)=header_roman;
  var StandardizedEst / style(header)=header_symbol;
  var tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label          = 'Variable'
        Estimate      = 'B'
        CIRange       = '95% CI'
        StandardizedEst = 'b'
        tValue        = 't'
        Probt         = 'p';
run; quit;

ods rtf close;

```

Here is the code to create the output in the [Intermediate Results](#) section ([Figure 6](#)):

```

* Change the magnitude of some values;

data baseball;
  set sashelp.baseball;
  hitsc  = nhits/100;
  runsc  = nruns/100;
  rbic   = nrbi/100;
  bbc    = nbb/100;
  crhitsc = crhits/100;
run;

* Close all open ODS destinations;

ods _all_ close;

* Save the ODS output objects as SAS tables;

ods output Nobs=NumObs
          ANOVA=Anova
          FitStatistics=FitStats
          ParameterEstimates=ParamEst;

*;
* Specify the title text, PROC REG code, and your path and filename.
* No other changes are needed.
*;

options nocenter nodate nonumber;

title1 'Table [your table number here]';
title2 '[your table title here]';

```

```

proc reg data=baseball plots=none;
  id   name team league;
  model logSalary = hitsc runsc rbic bbc yrmajor crhitsc/stb clb;
  label hitsc   = 'Hits in 1986, in hundreds'
        runsc   = 'Runs in 1986, in hundreds'
        rbic    = 'RBI in 1986, in hundreds'
        bbc     = 'BB 1986, in hundreds'
        crhitsc = 'Career hits, in hundreds';
run; quit;

ods rtf file = 'APA.rtf' bodytitle_aux style=Journal;

*;
* Do not edit anything below here, unless you changed the
* ODS destination.  If so, then be sure to close the destination
* at the end.
*;

* Create macro variables used in the footnote;

data _null_;
  set NumObs;
  if (_N_ eq 1);
  call symputx('N_OBS_USED', put(NObsUsed, best.));
run;

data _null_;
  set Anova;
  if (_N_ eq 1);
  call symputx('PROBF', put(ProbF, pvalue6.3));
run;

data _null_;
  set FitStats;
  if (_N_ eq 1);
  call symputx('R2', put(nValue2, 10.2));
run;

* Create the final table;

data Final;
  set ParamEst;
  keep Label Estimate CIRange StandardizedEst tValue Probt;
  if (upcase(Variable) ne 'INTERCEPT');

  if (compress(Label) eq '') then do;
    Label = Variable;
end;

  CIRange = '[' || strip(put(LowerCL, 10.2)) ||
            ', ' || strip(put(UpperCL, 10.2)) || ']';
run;

```



```

footnote "Note. R2 = &R2 "
        "(N = &N_OBS_USED, "
        "p &PROBF). "
        'CI = confidence interval for B.';

proc print data=Final noobs label;
  var Label Estimate CIRange StandardizedEst tValue Probt;
  format Estimate StandardizedEst tValue 10.2
         Probt pvalue6.3;
  label Label      = 'Variable'
        Estimate   = 'B'
        CIRange    = '95% CI'
        StandardizedEst = 'b'
        tValue     = 't'
        Probt      = 'p';
run; quit;

ods rtf close;

```

## CONCLUSION

SAS provides a rich set of procedures to perform statistical analyses, but the output from the procedures cannot be used directly in journal articles. By using DATA Step code and the SAS Output Delivery System (ODS), you can create elaborate publication-ready output.

## APPENDIX

The following table shows the Symbol font for Microsoft Windows systems:

Keyboard Characters	Symbol Font Characters	Keyboard Characters	Symbol Font Characters
a, A	$\alpha$ , A	n, N	$\nu$ , N
b, B	$\beta$ , B	o, O	$\omicron$ , O
c, C	$\chi$ , X	p, P	$\pi$ , $\Pi$
d, D	$\delta$ , $\Delta$	q, Q	$\theta$ , $\Theta$
e, E	$\epsilon$ , E	r, R	$\rho$ , P
f, F	$\phi$ , $\Phi$	s, S	$\sigma$ , $\Sigma$
g, G	$\gamma$ , $\Gamma$	t, T	$\tau$ , T
h, H	$\eta$ , H	u, U	$\upsilon$ , Y
l, I	I, I	v, V	$\varpi$ , $\varsigma$
j, J	$\varphi$ , $\vartheta$	w, W	$\omega$ , $\Omega$
k, K	$\kappa$ , K	x, X	$\xi$ , $\Xi$
l, L	$\lambda$ , $\Lambda$	y, Y	$\psi$ , $\Psi$
m, M	$\mu$ , M	z, Z	$\zeta$ , Z

**Table A - 1. The Symbol Font**

## REFERENCES

Nichol, Adelheid A. M., and Pexman, Penny M. 2010. *Presenting Your Findings*, Washington, DC: American Psychological Association.

SAS Institute Inc. 2008. "SAS® 9 Reporting Procedure Styles Tip Sheet". Available at <http://support.sas.com/rnd/base/ods/scratch/reporting-styles-tips.pdf>.

SAS Institute Inc. 2009. "Sample 36900: Instructions for viewing all of the style templates that are shipped with SAS®". Available at <http://support.sas.com/kb/36/900.html>.

SAS Institute Inc. 2014a. "CALL SYMPUTX Routine". *SAS® 9.4 Macro Language: Reference, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/mcrolref/64754/HTML/default/viewer.htm#p1fa0ay5pzz9yun1mvqxv8ipzd4d.htm>.

SAS Institute Inc. 2014b. "ODS ESCAPECHAR Statement". *SAS® 9.4 Output Delivery System: User's Guide, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsug/66611/HTML/default/viewer.htm#p11xia2ltavr8ln17srq8vn4rnqc.htm>.

SAS Institute Inc. 2014c. "ODS Output Statement". *SAS® 9.4 Output Delivery System: User's Guide, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsug/66611/HTML/default/viewer.htm#p0oxrbinw6fjuwn1x23gam6dntyd.htm>.

SAS Institute Inc. 2014d. "ODS Trace Statement". *SAS® 9.4 Output Delivery System: User's Guide, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsug/66611/HTML/default/viewer.htm#p1fpt3uuow90o3n155hs7bp1mo7f.htm>.

SAS Institute Inc. 2014e. "Style Attributes Detailed Information". *SAS® 9.4 Output Delivery System: User's Guide, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsug/67325/HTML/default/viewer.htm#p0xi2cygmfk0wkn1ei625zq5r488.htm>.

SAS Institute Inc. 2014f. "TEMPLATE Procedure: Creating a Style Template". *SAS® 9.4 Output Delivery System: Procedures Guide, Second Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsproc/67384/HTML/default/viewer.htm#p0cgtsockm3ei3n1uqw9v4mbswkh.htm>.

SAS Institute Inc. 2014g. "Uses of the Results Window". *SAS® 9.4 Language Reference: Concepts, Fourth Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/lrcon/67227/HTML/default/viewer.htm#n1039zk8bk9aton1fmbm7z2wji3k.htm>.

## RECOMMENDED READING

SAS Institute Inc. 2014. "ODS RTF Statement". *SAS® 9.4 Output Delivery System: User's Guide, Third Edition*. Cary, NC: SAS Institute Inc. Available at <http://support.sas.com/documentation/cdl/en/odsug/67325/HTML/default/viewer.htm#p1vsv8ucnjzinn1wq5wrlp74mdb.htm>.

SAS Institute Inc. 2014. "The REG Procedure". *SAS/STAT® 13.2 User's Guide*. Cary, NC: SAS Institute Inc. Available at [http://support.sas.com/documentation/cdl/en/statug/67523/HTML/default/viewer.htm#statug\\_reg\\_toc.htm](http://support.sas.com/documentation/cdl/en/statug/67523/HTML/default/viewer.htm#statug_reg_toc.htm).

## CONTACT INFORMATION

Your comments and questions are valued and encouraged.

Vincent DelGobbo  
SAS Institute Inc.  
100 SAS Campus Drive  
Cary, NC 27513



[sasvcd@SAS.com](mailto:sasvcd@SAS.com)

<http://www.sas.com/reg/gen/corp/867226?page=Resources>

Peter L. Flom  
Peter Flom Consulting  
(917) 488-7176

[PeterFlomConsulting@mindspring.com](mailto:PeterFlomConsulting@mindspring.com)

<http://www.StatisticalAnalysisConsulting.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.