How to Replace Missing Values in SAS

<https://sasexamplecode.com/replace-missing-values-in-sas/>

*/\* CREATE A DATA SET \*/*

**data** work.ds\_missing\_values;

infile datalines dlm=',';

input group $ var1 var2 var3;

datalines;

A, **1**, **2**, **3**

A, ., **2**, **5**

A, **3**, **4**, .

A, **8**, ., **8**

B, ., ., **1**

B, ., **2**, **6**

B, **3**, **1**, **8**

C, **9**, **4**, .

C, ., ., **7**

;

**run**;

## Replace Missing Values with Zeros

**Using COALESCE**

*/\* REPLACE WITH 0 - USING COALESCE \*/*

**data** work.ds\_no\_missing\_values;

set work.ds\_missing\_values;

var1 = coalesce(var1,**0**);

var2 = coalesce(var2,**0**);

var3 = coalesce(var3,**0**);

**run**;

*/\* REPLACE WITH 0 - USING ARRAY \*/*

**data** work.ds\_no\_missing\_values;

set work.ds\_missing\_values;

array num\_array \_numeric\_;

do over num\_array;

if missing(num\_array) then num\_array = **0**;

end;

**run**;

*/\* REPLACE WITH 0 - PROC STDIZE \*/*

**proc stdize** **data**=work.ds\_missing\_values

out=work.ds\_no\_missing\_values

reponly missing=**0**;

**run**;

*/\* REPLACE WITH MEAN (OF GROUP) - PROC STDIZE\*/*

**proc stdize** **data**=work.ds\_missing\_values

out=work.ds\_no\_missing\_values

reponly method=mean;

by group;

**run**;

*/\* REPLACE WITH MIN (OF GROUP) \*/*

**proc means** **data**=work.ds\_missing\_values;

by group;

output out=work.ds\_min min= /autoname;

**run**;

**data** work.ds\_no\_missing\_values;

merge work.ds\_missing\_values (in=a)

work.ds\_min (in=b);

by group;

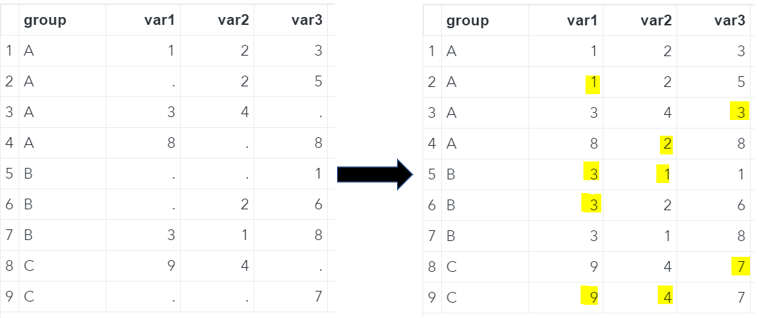
if missing(var1) then var1 = var1\_Min;

if missing(var2) then var2 = var2\_Min;

if missing(var3) then var3 = var3\_Min;

keep group var1 var2 var3;

**run**;

Missing Values Replaced With The **Group’s Minimum**

To replace the missing values with the minimum value of the complete column instead of the minimum of the group, simply remove the by statement in the MEANS procedure. However, you can’t use the merge statement as in the previous example. Because we removed the by statement from the MEANS procedure, need to use a join in the SQL procedure.

*/\* REPLACE WITH MIN (OVERALL) \*/*

**proc means** **data**=work.ds\_missing\_values;

output out=work.ds\_min min= /autoname;

**run**;

**proc sql**;

create table work.ds\_no\_missing\_values as

select group,

coalesce(var1, var1\_Min) as var1,

coalesce(var2, var2\_Min) as var2,

coalesce(var3, var3\_Min) as var3

from work.ds\_missing\_values, work.ds\_min;

**quit**;

*/\* REPLACE WITH MIN (OVERALL) \*/*

**proc means** **data**=work.ds\_missing\_values;

output out=work.ds\_min min= /autoname;

**run**;

**proc sql**;

create table work.ds\_no\_missing\_values as

select group,

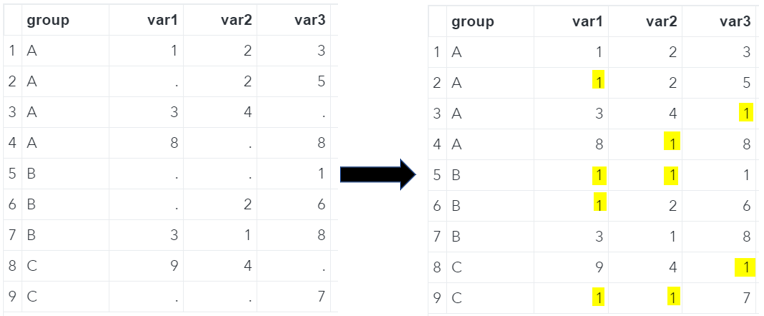
coalesce(var1, var1\_Min) as var1,

coalesce(var2, var2\_Min) as var2,

coalesce(var3, var3\_Min) as var3

from work.ds\_missing\_values, work.ds\_min;

**quit**;

Missing Values Replaced With The **Overall Minimum**

You can easily replace missing values with the maximum of a group using the code below.

*/\* REPLACE WITH MAX (OF GROUP) \*/*

**proc means** **data**=work.ds\_missing\_values;

by group;

output out=work.ds\_max max= /autoname;

**run**;

**data** work.ds\_no\_missing\_values;

merge work.ds\_missing\_values (in=a)

work.ds\_max (in=b);

by group;

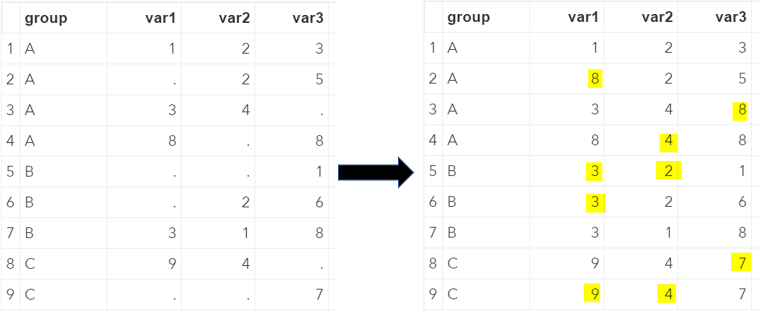
if missing(var1) then var1 = var1\_Max;

if missing(var2) then var2 = var2\_Max;

if missing(var3) then var3 = var3\_Max;

keep group var1 var2 var3;

**run**;

Missing Values Replaced With The **Group’s Maximum**

**Do you know?**[**5 Ways to Find the Minimum Value of a Column**](https://sasexamplecode.com/how-to-find-the-minimum-value-of-a-variable-by-group-in-sas/)

Finally, to replace missing values with the column maximum instead of the group maximum, you can use the example code below.

*/\* REPLACE WITH MAX (OVERALL) \*/*

**proc means** **data**=work.ds\_missing\_values;

output out=work.ds\_max max= /autoname;

**run**;

**proc sql**;

create table work.ds\_no\_missing\_values as

select group,

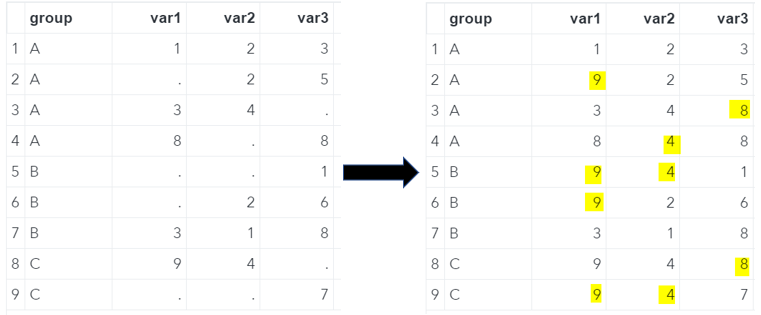
coalesce(var1, var1\_Max) as var1,

coalesce(var2, var2\_Max) as var2,

coalesce(var3, var3\_Max) as var3

from work.ds\_missing\_values, work.ds\_max;

**quit**;

Missing Values Replaced With The **Overall Maximum**

## Replace Missing Values with the Previous Non-Missing Value

Finally, we discuss how to replace missing values in SAS with the previous non-missing value. To do so we use the UPDATE statement.

### Using UPDATE

The UPDATE statement isn’t frequently used. However, to replace missing values with the last non-missing value it is very useful. The code snippet below shows the correct syntax. We will replace the missing values with the previous non-missing value from a specific group.

*/\* REPLACE WITH THE LAST NON-MISSING VALUE (OF GROUP) \*/*

**data** work.ds\_no\_missing\_values;

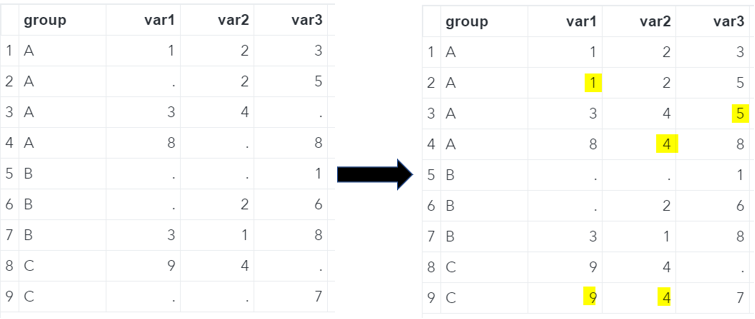
update work.ds\_missing\_values (obs=**0**) work.ds\_missing\_values;

by group;

output;

**run**;

The image below shows the result of the SAS code above. As you can see, the code correctly replaced the missing values with the last non-missing value by the variable group. A cell remains blank if all previous values of that specific group are missing. For example, in group B and variable one.

Missing Values Replaced With **Last Non-Missing Value**

#### SIMILAR POSTS

### [How to Count the Number of Missing Values in SAS](https://sasexamplecode.com/how-to-count-the-number-of-missing-values-in-sas/)

[10/01/2021](https://sasexamplecode.com/how-to-count-the-number-of-missing-values-in-sas/) [SAS Example Code](https://sasexamplecode.com/author/KempeSAS/)

.