

# The %MktBlock Macro

---

## Introduction

The %MktBlock autocall macro blocks a choice design or an ordinary linear experimental design. The %MktBlock macro tries to create a blocking factor that is uncorrelated with every attribute of every alternative. That is, it tries to make a factor that is orthogonal to all the attributes of all the alternatives.

The input design can be in one of two formats: a choice design or a linear arrangement. A choice design has one row for each alternative of each choice set and one column for each of the attributes. The %ChoiceEff and %MktRoll macros generate this kind of design. A linear arrangement is an intermediate step in preparing some choice designs. The linear arrangement has one row for each choice set and one column for each attribute of each alternative. The %MktEx macro generates linear arrangements. The output from the %MktBlock macro is a data set that contains the design, with runs or choice sets nested within blocks.

---

## %MktBlock Macro Syntax

**%MktBlock**(NBLOCKS=*n*, <, *optional arguments*>)

### Required Argument

**NBLOCKS**=*n*

specifies the number of blocks to create. NBLOCKS=1 reports information about the design.

### Optional Arguments

**ALT**=*variable*

specifies the variable that indexes the alternatives. If this variable is in the input data set, it is excluded from the factor list. By default, ALT=Alt.

**BLOCK**=*variable*

specifies the variable that indexes the blocks. If this variable is in the input data set, it is excluded from the factor list. By default, BLOCK=Block.

**DATA**=*SAS-data-set*

specifies either the choice design or the linear arrangement. The choice design has one row for each alternative of each choice set and one column for each attribute. For choice designs, you must also specify the NALTS= argument. The linear arrangement has one row for each choice set and one column for each attribute of each alternative. By default, the macro uses the last data set that is created.

**FACTORS=***variable-list*

**VAR=***variable-list*

specifies the factors in the design. By default, all numeric variables are used, except variables whose names match those that are specified in the BLOCK=, SET=, ALT=, and ID= arguments. (By default, the variables Block, Set, Run, and Alt are excluded from the factor list.)

**ID=***variable-list*

specifies the DATA= data set variables to copy to the output data set.

**INITBLOCK=***variable*

specifies the name of the variable in the data set that is used as the initial blocking variable for the first iteration.

**LIST=***r*

lists canonical correlations larger than LIST=*r*. The default is  $r = 0.316 \approx \sqrt{r^2} = 0.1$ .

**MAXITER=***n*

**ITER=***n*

specifies the number of times to try to block the design, starting with a different random blocking. By default, the macro tries five random starts, iteratively refines each until *D*-efficiency stops improving, and selects the blocking that has the best *D*-efficiency.

**NALTS=***n*

specifies the number of alternatives in each choice set. If you input a choice design, you must specify NALTS=; otherwise the macro assumes that you are inputting a linear arrangement.

**NEXT=***n*

specifies how far into the design to go to look for the next exchange. NEXT=1 requests that the macro try exchanging the level for each run with the level for the next run and all other runs. NEXT=2 requests that the macro consider exchanges with half of the other runs, which makes the algorithm run more quickly. The macro considers exchanging the level for run *i* with run *i* + 1, and then uses the NEXT= argument value to find the next potential exchanges. You can also specify other values, including nonintegers. For example, NEXT=1.5 requests that the macro consider exchanging observation 1 with observations 2, 4, 5, 7, 8, 10, 11, and so on. The macro tends to find a slightly better blocking variable when you specify smaller values, but at a cost of much slower run time.

**OPTIONS=***options-list*

specifies binary options. By default, no binary options are specified. You can specify the following value:

**NOSORT**

does not sort the design into blocks. This is useful anytime you want the order of the observations in the output data set to match the order of the observations in the input data set.

**OUT=***SAS-data-set*

specifies the output data set that contains the block numbers. By default, OUT=Blocked.

**OUTR=***SAS-data-set*

specifies the randomized output data set if you want the design to be randomly sorted within blocks.

**PRINT=***print-options*

specifies both the %MktBlock and %MktEval macro display options, which control the display of the results. By default, PRINT=NORMAL. You can specify one or more of the following values:

**ALL**

displays all output.

**CORR**

displays canonical correlations.

**BLOCK**

displays canonical correlations within blocks.

**DESIGN**

displays a blocked design.

**FREQS**

displays a long frequencies list.

**LIST**

displays a list of big canonical correlations.

**NONZERO**

is the same as ORDERED but sets LIST=1e-6.

**NOPRINT**

suppresses output.

**NORMAL**

is equivalent to specifying CORR LIST SUMM DESIGN NOTE.

**NOTE**

displays a blocking note.

**ORDERED**

is the same as LIST but displays a list that is ordered by variable names.

**SHORT**

is equivalent to specifying CORR SUMM NOTE.

**SUMM**

displays frequency summaries.

**RIDGE=***n*

specifies the value to add to the diagonal of  $\mathbf{X}'\mathbf{X}$  to make it nonsingular. Specify RIDGE=0 to use a generalized inverse instead of ridging. By default, RIDGE=0.01.

**SEED=***n*

specifies the random number seed. By default, SEED=0, and clock time is used to make the random number seed.

**SET=variable**

specifies the variable that indexes the choice sets. When the NALTS= argument is specified, the default value is Set; otherwise the default value is Run. If this variable is in the input data set, it is excluded from the factor list.

---

## Help Option

You can specify either of the following to display the option names and simple examples of the macro syntax:

```
%mktblock(help)
%mktblock(?)
```

## %MktBlock Macro Notes

This macro specifies **options nonotes** throughout most of its execution. If you want to see all the notes, submit the following statement before running the macro:

```
%let mktopts = notes;
```

To see the macro version, submit the following statement before running the macro:

```
%let mktopts = version;
```

---

## Example: Blocking a Linear Arrangement

When a choice design is too large to show all choice sets to each subject, the design can be blocked and a block of choice sets can be shown to each subject. For example, if you have 27 choice sets, instead of showing each subject 27 sets, you can create three blocks and show three groups of subjects 9 choice sets each. The following statements create a linear version of a choice design by using the %MktEx macro, and then block the design by using the %MktBlock macro:

```
%mktex(3 ** 6, n=27, seed=350)
```

By default, the %MktEx macro saves a randomized version of the design that it creates in a data set named Randomized:

```
%mktblock(data=Randomized, nblocks=3, seed=377, maxiter=50)
```

The DATA= argument specifies that the design to be blocked is contained in the data set Randomized. The NBLOCKS= argument creates three blocks. The MAXITER= argument requests 50 attempts to block the design with different random starts. The SEED= argument ensures reproducibility.

Figure 1 shows that the blocking factor is uncorrelated with all the factors in the design. This output comes from the %MktEval macro, which is called by the %MktBlock macro.

### Figure 1 %MktBlock Macro Output

**Canonical Correlations Between the Factors**  
There are 0 Canonical Correlations Greater Than 0.316

	Block	x1	x2	x3	x4	x5	x6
	<b>Block</b>	1	0	0	0	0	0
	<b>x1</b>	0	1	0	0	0	0
	<b>x2</b>	0	0	1	0	0	0
	<b>x3</b>	0	0	0	1	0	0
	<b>x4</b>	0	0	0	0	1	0
	<b>x5</b>	0	0	0	0	0	1
	<b>x6</b>	0	0	0	0	0	0

**Summary of Frequencies**  
**There are 0 Canonical Correlations Greater Than 0.316**

	Frequencies
Block	999
x1	999
x2	999
x3	999
x4	999
x5	999
x6	999
Block x1	3333333333
Block x2	3333333333
Block x3	3333333333
Block x4	3333333333
Block x5	3333333333
Block x6	3333333333
x1 x2	3333333333
x1 x3	3333333333
x1 x4	3333333333
x1 x5	3333333333
x1 x6	3333333333
x2 x3	3333333333
x2 x4	3333333333
x2 x5	3333333333
x2 x6	3333333333
x3 x4	3333333333
x3 x5	3333333333
x3 x6	3333333333
x4 x5	3333333333
x4 x6	3333333333
x5 x6	3333333333
N-Way	11111111111111111111 11111111

Figure 2 displays the blocked linear arrangement (three blocks of nine choice sets).

**Figure 2** %MktBlock Macro Output

Block=1							
Block	Run	x1	x2	x3	x4	x5	x6
1	1	2	1	2	2	1	1
	2	3	2	1	1	3	2
	3	1	3	3	3	2	3
	4	2	3	1	3	1	2
	5	1	1	1	2	2	2
	6	2	2	3	1	1	3
	7	3	1	3	2	3	3
	8	3	3	2	3	3	1
	9	1	2	2	1	2	1

  

Block	Run	x1	x2	x3	x4	x5	x6
2	1	1	3	3	1	3	2
	2	3	3	2	1	1	3
	3	2	2	3	2	2	2
	4	3	2	1	2	1	1
	5	2	1	2	3	2	3
	6	3	1	3	3	1	2
	7	2	3	1	1	2	1
	8	1	2	2	2	3	3
	9	1	1	1	3	3	1

  

Block	Run	x1	x2	x3	x4	x5	x6
3	1	3	2	1	3	2	3
	2	3	1	3	1	2	1
	3	1	3	3	2	1	1
	4	2	2	3	3	3	1
	5	2	1	2	1	3	2
	6	1	2	2	3	1	2
	7	3	3	2	2	2	2
	8	2	3	1	2	3	3
	9	1	1	1	1	1	3

In the linear version of the design, there is one row for each choice set, and all the attributes of all the alternatives are in the same row.

---

## Example 2: Blocking a Choice Design

This example creates and blocks a choice design that has two blocks of nine choice sets. The following statements create a choice design by using the %MktEx, %MktRoll, and %ChoiceEff macros:

```
%mktex(3 ** 6, n=3**6)
%mktroll(design=design, key=2 3, out=out)
%choicEff(data=out, model=class(x1-x3 / sta), nsets=18,
          nalts=2, seed=151, options=nodups relative, beta=zero)
```

By default, the %ChoiceEff macro saves the choice design in a data set named Best. The following statement blocks the design by using the %MktBlock macro:

```
%mktblock(data=best, nalts=2, nblocks=2, factors=x1-x3, seed=472)
```

The DATA= argument specifies that the design to be blocked is contained in the data set Best. The NALTS= argument specifies that each choice set contains two alternatives, and by implication, that the input data set is a choice design and not a linear arrangement. The NBLOCKS= argument creates two blocks. The FACTORS= argument specifies that the factors in the design are represented by the variables X1, X2, and X3. The SEED= argument ensures reproducibility.

Figure 3 shows the canonical correlations between the factors. The variable names in the output are composed of Alt, the alternative number, and the factor name. Because there are two alternatives, each composed of three factors plus one blocking variable, a  $7 \times 7$  correlation matrix is reported. The results indicate that the design and the blocking are not as good as in the previous example.

**Figure 3** %MktBlock Macro Output

**Canonical Correlations Between the Factors**  
**There are 9 Canonical Correlations Greater Than 0.316**

	Block	Alt1_x1	Alt1_x2	Alt1_x3	Alt2_x1	Alt2_x2	Alt2_x3
Block	1	0.25	0.14	0.14	0.15	0	0.14
Alt1_x1	0.25	1	0.45	0.36	0.69	0.35	0.31
Alt1_x2	0.14	0.45	1	0.12	0.21	0.58	0.33
Alt1_x3	0.14	0.36	0.12	1	0.31	0.43	0.52
Alt2_x1	0.15	0.69	0.21	0.31	1	0.56	0.21
Alt2_x2	0	0.35	0.58	0.43	0.56	1	0.29
Alt2_x3	0.14	0.31	0.33	0.52	0.21	0.29	1

**Canonical Correlations > 0.316 Between the Factors**  
**There are 9 Canonical Correlations Greater Than 0.316**

	r	r Square
Alt1_x1 Alt2_x1	0.69	0.48
Alt1_x2 Alt2_x2	0.58	0.34
Alt2_x1 Alt2_x2	0.56	0.31
Alt1_x3 Alt2_x3	0.52	0.27
Alt1_x1 Alt1_x2	0.45	0.20
Alt1_x3 Alt2_x2	0.43	0.18
Alt1_x1 Alt1_x3	0.36	0.13
Alt1_x1 Alt2_x2	0.35	0.13
Alt1_x2 Alt2_x3	0.33	0.11

**Figure 3** *continued*

**Summary of Frequencies**  
**There are 9 Canonical Correlations Greater Than 0.316**  
**\* - Indicates Unequal Frequencies**

	Frequencies
Block	9 9
* Alt1_x1	9 8 1
* Alt1_x2	6 5 7
* Alt1_x3	6 7 5
* Alt2_x1	3 4 1 1
Alt2_x2	6 6 6
* Alt2_x3	6 5 7
* Block Alt1_x1	4 4 1 5 4 0
* Block Alt1_x2	3 2 4 3 3 3
* Block Alt1_x3	3 4 2 3 3 3
* Block Alt2_x1	2 2 5 1 2 6
Block Alt2_x2	3 3 3 3 3 3
* Block Alt2_x3	3 2 4 3 3 3
* Alt1_x1 Alt1_x2	3 4 2 3 1 4 0 0 1
* Alt1_x1 Alt1_x3	3 4 2 3 2 3 0 1 0
* Alt1_x1 Alt2_x1	0 4 5 2 0 6 1 0 0
* Alt1_x1 Alt2_x2	3 3 3 2 3 3 1 0 0
* Alt1_x1 Alt2_x3	3 3 3 3 2 3 0 0 1
* Alt1_x2 Alt1_x3	2 2 2 2 2 1 2 3 2
* Alt1_x2 Alt2_x1	1 2 3 1 1 3 1 1 5
* Alt1_x2 Alt2_x2	0 3 3 2 0 3 4 3 0
* Alt1_x2 Alt2_x3	2 2 2 2 2 1 2 1 4
* Alt1_x3 Alt2_x1	1 2 3 2 1 4 0 1 4
* Alt1_x3 Alt2_x2	2 3 1 2 1 4 2 2 1
* Alt1_x3 Alt2_x3	0 3 3 3 0 4 3 2 0
* Alt2_x1 Alt2_x2	1 0 2 1 3 0 4 3 4
* Alt2_x1 Alt2_x3	1 1 1 2 1 1 3 3 5
* Alt2_x2 Alt2_x3	1 2 3 3 1 2 2 2 2
N-Way	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Because the input data set is a choice design, the results shown in Figure 4 are in choice design format; there is one row for each alternative of each choice set.

**Figure 4** %MktBlock Macro Output

Block=1 Set=1					
Block	Set	Alt	x1	x2	x3
1	1	1	2	3	1
		2	3	1	3
Block	Set	Alt	x1	x2	x3
1	2	1	2	3	3
		2	3	2	1



**Figure 4** *continued*

Block	Set	Alt	x1	x2	x3
1	3	1	2	1	2
		2	1	3	1

Block	Set	Alt	x1	x2	x3
1	4	1	1	3	2
		2	2	2	1

Block	Set	Alt	x1	x2	x3
1	5	1	1	2	1
		2	3	1	2

Block	Set	Alt	x1	x2	x3
1	6	1	1	2	2
		2	3	3	3

Block	Set	Alt	x1	x2	x3
1	7	1	3	3	2
		2	1	1	3

Block	Set	Alt	x1	x2	x3
1	8	1	2	1	3
		2	3	3	2

Block	Set	Alt	x1	x2	x3
1	9	1	1	1	1
		2	2	2	3

Block	Set	Alt	x1	x2	x3
2	1	1	2	3	1
		2	3	2	3

Block	Set	Alt	x1	x2	x3
2	2	1	2	2	1
		2	1	3	2

Block	Set	Alt	x1	x2	x3
2	3	1	1	2	3
		2	2	1	1

Block	Set	Alt	x1	x2	x3
2	4	1	2	1	3
		2	3	2	1

Block	Set	Alt	x1	x2	x3
2	5	1	1	1	1
		2	2	2	2

**Figure 4** *continued*

Block	Set	Alt	x1	x2	x3
2	6	1	1	2	2
		2	3	3	1

Block	Set	Alt	x1	x2	x3
2	7	1	2	3	2
		2	3	1	3

Block	Set	Alt	x1	x2	x3
2	8	1	1	1	2
		2	3	3	3

Block	Set	Alt	x1	x2	x3
2	9	1	1	3	3
		2	3	1	2