

%MktBSize Macro

The %MktBSize autocall macro suggests sizes for balanced incomplete block designs (BIBDs). The sizes that it reports are sizes that meet necessary but not sufficient conditions for the existence of a BIBD, so a BIBD might not exist for every size reported. In the following example, a list of designs with 12 treatments, between 4 and 8 treatments per block, and between 12 and 30 blocks are requested:

```
%mktbsize(t=12, k=4 to 8, b=12 to 30)
```

The results are as follows:

t	k	b	r	Lambda	n
Number of Treatments	Block Size	Number of Blocks	Treatment Frequency	Pairwise Frequencies	Total Sample Size
12	6	22	11	5	132

You can use this information to create a BIBD with the %MktBIBD macro as follows:

```
%mktbibd(t=12, k=6, b=22, seed=104)
```

There is no guarantee that %MktBIBD will find a BIBD for any specification, but in this case it does, and it finds the following design:

Balanced Incomplete Block Design

x1	x2	x3	x4	x5	x6
7	9	3	10	12	2
5	2	4	6	10	7
5	10	12	9	11	8
7	9	11	12	1	4
12	3	4	5	9	10
10	6	9	11	2	1
3	8	7	1	9	6
2	10	12	6	8	1
9	7	6	4	8	5
1	12	5	7	6	11
12	1	2	3	5	7
8	4	1	10	7	12
6	2	9	8	12	4
4	7	11	3	10	6
11	5	1	2	4	9
11	3	2	8	7	9
6	11	8	5	3	12

3	6	10	9	1	5
2	4	6	12	11	3
1	8	10	11	4	3
4	1	3	2	5	8
10	5	8	7	2	11

The design has $b=22$ blocks (rows), $k=6$ treatments per block (columns), and $t=12$ treatments (the entries are the integers 1 to 12). Each of the $t=12$ treatments occurs $r=11$ times, and each treatment occurs in a block with every other treatment $\lambda=5$ times.

The following step creates a list of over 50 designs:

```
%mktbsize(t=5 to 20, k=3 to t - 1, b=t to 30)
```

Some of the results are as follows:

t	k	b	r	Lambda	n
Number of Treatments	Block Size	Number of Blocks	Treatment Frequency	Pairwise Frequencies	Total Sample Size
5	3	10	6	3	30
5	4	5	4	3	20
6	3	10	5	2	30
6	4	15	10	6	60
6	5	6	5	4	30

Note that by default, `maxreps=1` (the maximum number of replications is 1), so for example, the design $t=5$, $k=3$, $b=20$ is not listed since it consists of two replications of $t=5$, $k=3$, $b=10$, which is listed. Also note that b , the number of blocks, was specified so that it is never less than the number of treatments. Furthermore, k , the block size (number of treatments per block), is set to always be less than the number of treatments. Even more complicated expressions are permitted. For example, to limit the number of treatments per block to no more than half of the number of treatments, you could specify the following:

```
%mktbsize(t=2 to 10, k=2 to 0.5 * t, b=t to 10)
```

The results are as follows:

t	k	b	r	Lambda	n
Number of Treatments	Block Size	Number of Blocks	Treatment Frequency	Pairwise Frequencies	Total Sample Size
4	2	6	3	1	12
5	2	10	4	1	20
6	3	10	5	2	30
7	3	7	3	1	21

To limit the number of blocks as a function of the number of treatments, you could specify the following:

```
%mktbsize(t=2 to 10, k=2 to t - 1, b=t to 2 * t)
```

However, if you want to limit the number of treatments as a function of the number of blocks, you need to use the `order=` option to ensure that the number of blocks loop comes first, for example, as follows:

```
%mktbsize(b=2 to 10, t=2 to 0.5 * b, k=2 to t - 1, order=btk)
```

The macro reports sizes in which $r = b \times k/t$ and $l = r \times (k - 1)/(t - 1)$ are integers, $2 \leq k < t$, and $b \geq t$. When $r = b \times k/t$ and $l = r \times (k - 1)/(t - 1)$ are integers, and $k = t$ and $b \geq t$, then a complete block design might be possible. This is a necessary but not sufficient condition for the existence of a complete block design. When $r = b \times k/t$ and $l = r \times (k - 1)/(t - 1)$ are integers, and $k < t$ and $b \geq t$, then a balanced incomplete block design might be possible. This is a necessary but not sufficient condition for the existence of a BIBD. When you specify `options=ubd` and $r = b \times k/t$ is an integer, then unbalanced block design sizes are reported as well. For example, if you want a design with $t=20$ treatments and a block size of 6, you can run the following to find out how many blocks you need:

```
%mktbsize(t=20, k=6, options=ubd)
```

The results are as follows:

t	k	b	r	Lambda	n
Number of Treatments	Block Size	Number of Blocks	Treatment Frequency	Pairwise Frequencies	Total Sample Size
20	6	10	3	0.79	60

Then the %MktBIBD macro can be used to find a design where each treatment occurs 3 times, but the treatments do not appear together an equal number of times, for example, as follows:

```
%mktbibd(t=20, k=6, b=10, seed=104)
```

Some of the results are as follows:

Treatment by Treatment Frequencies

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	3	1	0	0	2	1	1	1	0	1	1	1	0	2	0	1	0	1	1	1
2		3	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0	2	1
3			3	1	1	1	1	0	1	0	1	0	1	1	1	2	2	1	1	0
4				3	0	1	1	1	0	1	1	1	1	0	1	1	1	1	2	0
5					3	1	1	1	1	1	0	1	1	1	0	2	0	0	1	0
6						3	0	0	1	1	1	0	1	1	1	1	0	1	1	1
7							3	1	0	0	2	2	1	1	1	0	1	0	1	0
8								3	1	1	1	1	2	1	0	1	1	0	0	1
9									3	1	0	1	2	0	1	1	1	1	0	2
10										3	0	1	0	1	2	1	0	1	1	1
11											3	1	1	2	1	0	1	1	0	1
12												3	1	0	1	0	0	1	1	1
13													3	0	0	1	1	0	0	1
14														3	1	1	1	1	0	1
15															3	0	1	1	1	1
16																3	1	1	1	0
17																	3	1	1	1
18																		3	1	2
19																			3	0
20																				3

%MktBSize Macro Options

The following options can be used with the %MktBIBD macro:

Option	Description
<code>help</code>	(positional) “help” or “?” displays syntax summary
<code>b=do-list</code>	number of blocks (alias for <code>nsets=</code>)
<code>k=do-list</code>	block size (alias for <code>setsize=</code>)
<code>maxreps=n</code>	maximum number of replications
<code>nattrs=do-list</code>	number of attributes (alias for <code>t=</code>)
<code>nsets=do-list</code>	number of sets (alias for <code>b=</code>)
<code>order=order-list</code>	order of the loops
<code>options=nocheck</code>	suppress checking <i>b</i> , <i>t</i> , and <i>k</i>
<code>options=ubd</code>	lifts the balance restriction on the design
<code>out=SAS-data-set</code>	output data set design list
<code>setsize=do-list</code>	set size (alias for <code>k=</code>)
<code>t=do-list</code>	number of treatments (alias for <code>nattrs=</code>)

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

```
%mktbsize(help)
%mktbsize(?)
```

The **k=** or **setsize=**, and the **t=** or **nattrs=** options must be specified.

b= *do-list*

nsets= *do-list*

specifies the number of blocks. In a partial-profile design, this is the number of profiles. In a MaxDiff design, this is the number of sets. Specify either an integer or a list of integers in the SAS *do-list* syntax. The default is **b=2** to 500. The **nsets=** and **b=** options are aliases.

k= *do-list*

setsize= *do-list*

specifies the block size, or the number of treatments in each block. In a partial-profile or MaxDiff design, this is the number of attributes or messages shown at one time in each set. Specify either an integer or a list of integers in the SAS *do-list* syntax. The **setsize=** and **k=** options are aliases. This option (in one of its two forms) must be specified.

maxreps= *n*

specifies the maximum number of replications. The default is **maxreps=1**. By default, this option prevents the %MktBSize macro from reporting designs of size $2b$, $3b$, and so on after it has found a size with b blocks.

options= *options-list*

specifies binary options. By default, none of these options are specified. Specify one or more of the following values after **options=**.

nocheck

by default, certain checks are performed on b , t , and k . Specify **options=nocheck** to turn them off. This lets you make some creative expressions that otherwise would not be permitted.

ubd

lifts the balance restriction on the design. Results are reported when $r = b \times k/t$ is in integer but $l = r \times (k - 1)/(t - 1)$ might or might not be an integer. Use this option when you want to see sizes where every treatment can occur equally often, but the pairwise frequencies can be unequal. The listing can contain both sizes where a BIBD might be possible (λ , the expected pairwise frequency, is an integer) and sizes where a BIBD is not possible (λ is not an integer). You might use this option, for example, when the block design is being used to make a partial-profile design.

order= *tkb | tbk | btk | bkt | kbt | ktb*

specifies the order of the loops, the default is **tkb**, *t* then *k* then *b*. If you specify expressions in **t=**, **b=**, or **k=**, you might need some other ordering. For example, if you specify something like **t = 2 to 0.5 * b**, then you must specify **order=bkt** or any other ordering that defines *b* before *t*. Alternatively, you can specify this option just to change the default ordering of the results.

out= *SAS-data-set*

specifies the output data set with the list of potential design sizes. The default is **out=bibd**.

t= *do-list*

nattrs= *do-list*

specifies the number of treatments. In a partial-profile or MaxDiff design, this is the total number of attributes or messages. Specify either an integer or a list of integers in the SAS *do-list* syntax. The **nattrs=** and **t=** options are aliases. This option (in one of its two forms) must be specified. When the **nattrs=** option is specified, the output will use the word “Attribute” rather than “Treatment” and “Set” rather than “Block”.

%MktBSize Macro Notes

This macro specifies **options nonotes** throughout most of its execution. If you want to see all of the notes, submit the statement **%let mktopts = notes;** before running the macro. To see the macro version, submit the statement **%let mktopts = version;** before running the macro.