E177, N-D Arrays
Array Assignment
http://jagger.me.berkeley.edu/~pack/e177

Copyright 2005-2008, Andy Packard. This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/2.0/ or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Multidimensional Assignment
Let A and R be multidimensional arrays. Consider
nA = ndims(A); szA = size(A)
nR = ndims(R); szR = size(R)
A(idx1,idx2,...,idxM) = R
if szR equals
[length(idx1) length(idx2) ... length(idxM)]
then the assignment takes place. Moreover, for
k1<=szR(1), k2<=szR(2),... kM<=szR(M)
the values of
A(idx1(k1),idx2(k2),...,idxM(kM))
R(k1,k2,...,kM)
are equal. If M<nA, and
max(idx1)>szA(1), max(idx2)>szA(2),... max(idxM)>szA(M),
then the size of A will be unchanged

Multidimensional Assignment (growing LHS)
Let A and R be multidimensional arrays. Consider
nA = ndims(A); szA = size(A)
nR = ndims(R); szR = size(R)
A(idx1,idx2,...,idxM) = R
if szR equals
[length(idx1) length(idx2) ... length(idxM)]
then the assignment takes place. Moreover, for
k1<=szR(1), k2<=szR(2),... kM<=szR(M)
the values of
A(idx1(k1),idx2(k2),...,idxM(kM))
R(k1,k2,...,kM)
are equal. If M>=nA, and any of
max(idx1)>szA(1), max(idx2)>szA(2),... max(idxM)>szA(M),
then A will automatically grow to appropriate size.

Multidimensional Assignment (singleton dims)
The built-in command squeeze removes singleton dimensions from an array
size(squeeze(rand(4,5,1,3,1,7))) % [4 5 3 7]
size(squeeze(rand(1,4,5))) % [4 5]
size(squeeze(rand(1,4))) % [1 4]
size(squeeze(rand(4,1))) % [4 1]
Now, again let A and R be multidimensional arrays. Consider
nA = ndims(A); szA = size(A)
nR = ndims(R); szR = size(R)
A(idx1,idx2,...,idxM) = R
if size(squeeze(R)) equals
size(squeeze(A(idx1,idx2,...,idxM)))
then the assignment takes place. In other words, dimensions of R and length(idx) do not have to line up if the differences are due to singletons.

Multidimensional Assignment (examples w/ doubles)
A = randn(3,4,2,5);
The error here, several exceptions...
A(:,:,,[1 4],[2 3]) = ones(3,2,1,3);
sz(A)
B = randn(4,5,6,7);
B(:,:,,[13 19 41]) = reshape(1:60,[4 5 3]);
sz(B)
C = randn(5,5,5);
C(:,:,,[13]) = A(:,:,2,4,9 [8]);
sz(C)
D = randn(4,5,6,7);
D(:,:,,[1 4],[1 5],[5 6],[2 4]) = rand(8,2)
F = randn(3,4,5);
F(:,:,,[3 4],[1 6]) = ones(2,3,7);
F([3 4],45) = pi; % error, unambiguous growth

Multidimensional Assignment (singletons, examples)
A = randn(4,5,6,7);
A(:,:,,[14 15 41]) = ones(4,5,1,3);
A(:,:,,[14 15 41]) = ones(4,5,1,3);
A(:,:,,[14 15 41]) = ones(4,5,1,3);
A(:,:,,[14 15 41]) = rand(4,1,5,1,3);
A(:,:,3,[14 15 41]) = rand(4,3);
A(:,:,3,[14 15 41]) = rand(4,3);
### Multidimensional Assignment (2-d)

Let A and R be multidimensional arrays. Consider:

\[
\begin{align*}
\text{nA} &= \text{ndims}(A); \quad \text{szA} = \text{size}(A) \\
\text{nR} &= \text{ndims}(R); \quad \text{szR} = \text{size}(R) \\
A(\text{idx1, idx2}) &= R
\end{align*}
\]

If \(\text{nA} = 2\) and \(\text{nR} = 2\) and each side represents a vector (row or column), then regardless whether the sides line up (i.e., one may be row, one may be column), the assignment takes place.

**Example**

\[
\begin{align*}
A &= \text{randn}(5, 6); \\
A(:,3) &= \text{ones}(1,5); \quad \text{5-by-1, 1-by-5}
\end{align*}
\]

Note `squeeze` doesn’t make these equal.

### Multidimensional Assignment (LHS single idx)

Let A and R be multidimensional arrays. Consider:

\[
\begin{align*}
\text{nA} &= \text{ndims}(A); \quad \text{szA} = \text{size}(A) \\
\text{nR} &= \text{ndims}(R); \quad \text{szR} = \text{size}(R) \\
A(\text{idx1}) &= R
\end{align*}
\]

If \(\text{prod(szR)}\) equals \(\text{length(idx1)}\) then the assignment takes place, drawing elements from R in linear order. In other words, for \(k1 <= \text{prod(szR(1))}\) the values \(A(\text{idx1(k1)})\) and \(R(k1)\) are equal.

**Examples**

\[
\begin{align*}
A &= \text{randn}(4,3,5,6); \\
A([3 5 7 10 100 1 76 85]) &= \text{rand}(2,2); \\
A([3 5 7 10 100 1 76 85]) &= 1:8;
\end{align*}
\]