```
% Produces:
    % vORIG = X% of the length is 1's (active).
    % v1-v5 = vectors identical to vORIG, except with 20% of the data reset
    % back to 0.
% Length MUST be multiples of 10.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [vORIG v1 v2 v3 v4 v5 v6 v7 v8 v9 v10 v11 v12 v13 v14 v15 v16 v17 v18 v19

% Generating binary vector with 10% of the length being 1's (active units).
n = length*percent;
vORIG = zeros(1, length);
vORIG(randperm(numel(vORIG), n)) = 1;

% Reset v1 - v5 of 20% of original data to 0.
m = length*0.2;

v1 = vORIG;
v2 = vORIG;
v3 = vORIG;
v4 = vORIG;
v5 = vORIG;
v6 = vORIG;
v7 = vORIG;
v8 = vORIG;
v9 = vORIG;
v10 = vORIG;

v1(randperm(numel(v1), m)) = 0;
v2(randperm(numel(v2), m)) = 0;
v3(randperm(numel(v3), m)) = 0;
v4(randperm(numel(v4), m)) = 0;
v5(randperm(numel(v5), m)) = 0;
v6(randperm(numel(v6), m)) = 0;
v7(randperm(numel(v7), m)) = 0;
v8(randperm(numel(v8), m)) = 0;
v9(randperm(numel(v9), m)) = 0;
v10(randperm(numel(v10), m)) = 0;

k = length*0.1

v11 = vORIG;
v12 = vORIG;
v13 = vORIG;
v14 = vORIG;
v15 = vORIG;
v16 = vORIG;
v17 = vORIG;
v18 = vORIG;
v19 = vORIG;
v20 = vORIG;
```

```
v21 = vORIG;

v11(randperm(numel(v11), k)) = 0;
v12(randperm(numel(v12), k)) = 0;
v13(randperm(numel(v13), k)) = 0;
v14(randperm(numel(v14), k)) = 0;
v15(randperm(numel(v15), k)) = 0;
v16(randperm(numel(v16), k)) = 0;
v17(randperm(numel(v17), k)) = 0;
v18(randperm(numel(v18), k)) = 0;
v19(randperm(numel(v19), k)) = 0;
v20(randperm(numel(v20), k)) = 0;
v21(randperm(numel(v21), k)) = 0;



end
```

```
Error using patternGen (line 12)
Not enough input arguments.
```

Published with MATLAB® R2014b