```
function [ signalPerformance ] = signalPerformance4_avg( threshold_dBA, idxCombinations, signal_dBA  )
% This function compares the signal to the requirement

% Initializing signalPerformance variable
signalPerformance = zeros( 1, size( idxCombinations, 1 ) );

%looping through all combinations of 4 bands to see if any pass the 4-band threshold requirement
for ink = 1 : size( idxCombinations, 1 )

     idx = idxCombinations( ink, idxCombinations( ink, : ) > 0 ); % Of the combinations checking the "ink" row to make sure all indices are greater than 0.
                                                                   % They should always be greater than 0!Check may not be needed

     for lead = 1 : size( signal_dBA, 1 )

         if length( threshold_dBA ) > size( idxCombinations, 2 ) % If the size of the threshold (requirement) is larger than the size of the combinations (again, should always be true).
             thresholdExceedance = signal_dBA( lead, idx ) - threshold_dBA( idx ) ; % The threshold is the minimum SPL levels for the 4-band requirement
         else
             thresholdExceedance = signal_dBA( lead, idx ) - threshold_dBA ;
         end


         % Bands Spanned, Number of Bands, Number of Bands Detectable
         numBandsDetected = length( thresholdExceedance( thresholdExceedance >= 0 ) ); % Since signal-threshold, looking for values that are greater than or equal to 0
         if numBandsDetected == 4 %4 bands
             signalPerformance( lead, ink ) = 1;
         else
             signalPerformance( lead, ink ) = 0;
         end

     end

end
```