```
rois = 10;
myrow = 1:10;
% myrow =  [1     2     3     4     5     6     7     8     9    10]


looks_matrix = zeros(rois,rois);
entropy_matix = zeros(rois,rois);
row_total = zeros(rois);
col_total = zeros(rois);
mysize = 0;

%turn the list of ROIs to a transition matrix;
for j=2:1:length(myrow)
    mysize = mysize+1;
    from = myrow(j-1);
    to = myrow(j);
    looks_matrix(from,to) = looks_matrix(from,to)+1;
end

num_looks_matrix = looks_matrix;
looks_matrix=looks_matrix./mysize;

%get the entropy of each cell in the matrix
for R=1:1:(size(looks_matrix,1))
    for C=1:1:(size(looks_matrix,2))
        entropy_matrix(R,C) = looks_matrix(R,C) * log2(1/
looks_matrix(R,C));
    end;
end;


%get the row and colum totals
columntotals = sum(looks_matrix);
rowtotals = sum(looks_matrix,2);

for ct = 1:1:rois
    colent(ct) = columntotals(ct) * log2(1/columntotals(ct));
    rowent(ct) = rowtotals(ct) * log2(1/rowtotals(ct));
end;

colenttotal = sum(colent(isnan(colent)==0));
rowenttotal = sum(rowent(isnan(rowent)==0));

correction = (colenttotal + rowenttotal)/2;
cellenttotal = sum(sum(entropy_matrix(isnan(entropy_matrix)==0)));
entropytotal = 1-((colenttotal + rowenttotal - cellenttotal)/
correction);

myent = entropytotal;
```

Published with MATLAB® R2018a