

```
1 % Hypergeometric Exact Expectations for large N, full freq spectrum 230310
2 function [HypExp] = HypExactExp230607(parg,Narg,narg);
3 % x=numbers detected in narg sampled from Narg with true parg
4 q=1-parg; % Prop alt allele
5 qN=round(q*Narg); pN=round(parg*Narg);
6 % Dim for cumulator
7 HypExp = zeros(1,(narg+1));
8
9 for x1=1:narg+1; % loop thru bins, +1 so index<>0
10     x=x1-1; notx=narg-x; % actual numbers
11     notx1=notx+1; %for index, never zero
12     %Dimensions for cumulators
13     qNpNdetect=ones(1,narg);
14     qNundetected=zeros(1,notx);
15     pNdetect=zeros(1,x);
16     nN=zeros(1,narg);
17     nNDenom=ones(1,narg);
18
19     for det1=1:x; % numerator for Hyp Prob - p part
20         det=det1-1; % actual numbers
21         pNdetect(det1)=(pN-det)/(x-det);
22         if pNdetect(det1)==Inf; pNdetect(det1)=1; end; %Div(0!)trap
23     end; % end numerator loop - p part
24
25     for undet1=1:notx; % numerator for Hyp Prob - q part
26         undet=undet1-1; % actual numbers
27         qNundetected(undet1)=(qN-undet)/(notx-undet);
28         if qNundetected(undet1)==Inf; qNundetected(undet1)=1; end; %Div(0!)trap
29     end; % end numerator loop - q part
30
31     % concat numerator, large to small, traps for all target or non-target
32     if notx==narg;
33         qNpNdetect(1:notx)=fliplr(qNundetected);
34     elseif x==narg;
35         qNpNdetect(1:x)=fliplr(pNdetect);
36     else
37         qNpNdetect(1:notx)=fliplr(qNundetected);
38         qNpNdetect(notx+1:narg)=fliplr(pNdetect);
39     end; % end concatenation
40
41     for nn=1:narg; %denominator for hyp prob
42         nN(nn)=(Narg-nn+1)/(narg-nn+1);
43     end; % end denom loop
44     nNDenom(1:narg)=fliplr(nN(1:narg)); % large to small, then unity
45
46     HypElements=qNpNdetect(:)./nNDenom(:); % column vector, narg rows
47     HypExp(x1)=prod(HypElements); %Exp for(Narg,parg,detect)
48 end; % end main loop
49 % end function
50 end
```