**MATLAB Code.**

% This Matlab code calculates p-values for averages of lifespans of mice

% treated with multiple compounds with the null hypothesis being that

% these mice come from a control population. The control population data

% must be in the file 'IndCont.ext' as a column of lifespans of individual

% control mice, in days.

%

% The program assumes that there are 15 mice treated by each compound.

% This constant is set up in the beginning of the file. It also assumes that no

% mouse lived longer than 1500 days.

%

% We start by computing the distribution of the averages of one compound

% under the null hypothesis, by taking successive convolutions of distributions

% of the lifespans of individual control mice. The Matlab function conv() is

% utilized for this purpose.

%

% The program then asks for the maximum number of compounds MaxNumComp

% that one wishes to consider. It calculates the distributions of averages of

% multiple compounds under the null hypothesis assumption by taking successive

% convolutions with the distributions of single compound.

%

% Finally, the program interactively asks for actual experimental data for

% averages of multiple compounds and outputs the p-values, i.e. the probabilities

% of observed or more extreme values appearing under the null hypothesis.

%

% Written by Lev Borisov, 2013.

MicePerComp = 15;

load IndCont.ext % loading the data from the file

data=IndCont;

NumMice = length(data);

h1 = zeros(1,1500); % generating initial histogram h1

for k = [1:NumMice],

 h1(data(k)) = h1(data(k))+1;

end

for i = [1:1500],

 h1(i)=h1(i)/NumMice;

end

h2 = h1; % generating histogram of one compound sums h2

for k = [1:MicePerComp-1],

 h2 = conv(h2,h1);

end

% There are MicePerComp mice for each compound. Due to technical reasons

% it is shifted down by MicePerComp-1 (the arrays start at one, not 0).

% This issue is addressed in the subsequent code.

h3 = zeros(1,1500); % generating histogram of one compound averages h3

for i=[1:1499\*MicePerComp+1],

 j = round((i+MicePerComp-1)/MicePerComp);

 h3(j) = h3(j) + h2(i);

end

MaxNumComp = input('Please input the maximum number of compounds: ');

AllPvalues = zeros(MaxNumComp,1500); % initialize the p-values array

h4 = h3; % initialize the histogram of sums of NumComp compounds h4

for NumComp = [1:MaxNumComp],

 fprintf('%i\n',NumComp); % print the number of compounds being considered

 h5 = zeros(1,1500); % generate the histogram of averages of NumComp compounds h5

 for i=[1:1499\*NumComp+1],

 j=round((i+NumComp-1)/NumComp);

 h5(j)=h5(j) + h4(i);

 end

 pvalue = 0; % save the pvalues of averages into AllPvalues

 for i1=[1:1500],

 pvalue = pvalue + h5(i1);

 AllPvalues(NumComp,i1) = pvalue;

 end

 h4 = conv(h4,h3); % generate the histogram of sums of one more component

end

fprintf('P-values generated\n');

while NumComp>0, % initially, NumComp=MaxNumComp>0, later may be entered 0

 results = input('Please enter [number of compounds (0 to exit), average]: ');

 NumComp = results(1);

 Ave = results(2);

 if NumComp > 0,

 p1 = AllPvalues(NumComp,round(Ave)); % find the probability of smaller or equal average

 p2 = 1-AllPvalues(NumComp,round(Ave)-1); % find the probability of larger or equal average

 if p1 > p2,

 fprintf('Increases life span, p-value = %s\n',p2);

 end

 if p2 >= p1,

 fprintf('Decreases life span, p-value = %s\n',p1);

 end

 end

end