Scarlett Royle, 201245875

26/10/2018

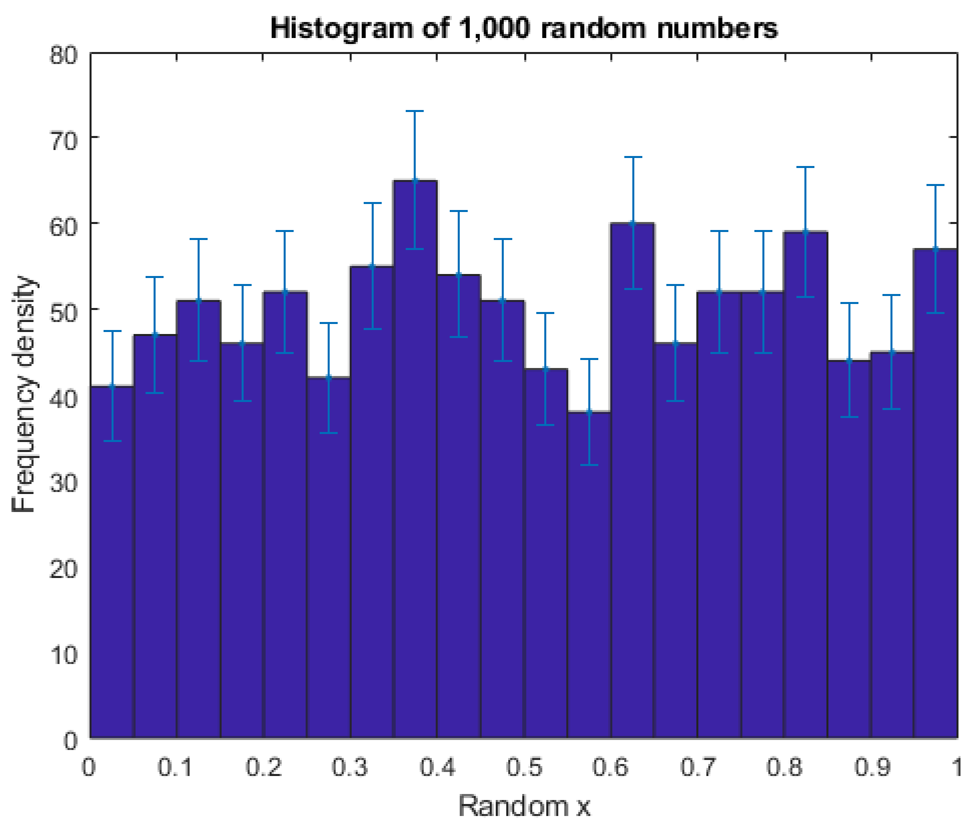
PHYS205: Problem set 3

Question 1a

For this question, I generated sets of random numbers, plotted their sums on histograms and analysed the distributions in order to demonstrate Central Limit Theorem.

```
x = rand(1000, 1); %1000 random numbers, 0-1

figure
hist(x, 20); %20 bins
hold on
[countsx, binsx] = hist(x, 20);
abs_unc = sqrt(countsx); %absolute uncetainties array
errorbar(binsx, countsx, abs_unc, abs_unc,'.'); %plotting errorbars: x, y, +ve, -ve unc.
xlabel('Random x');
ylabel('Frequency density');
title('Histogram of 1,000 random numbers');
hold off
```

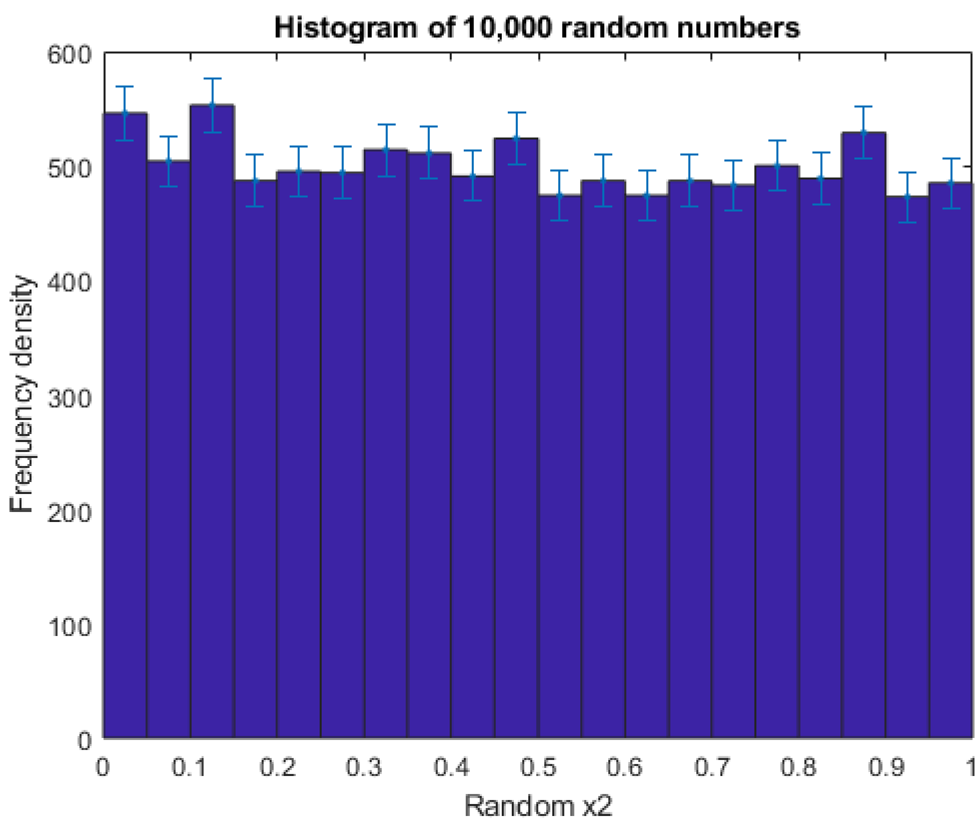


Histogram 1: Histogram of 1,000 random x values between 0 and 1 and their frequency

By increasing the number of random numbers, N, it is expected that the absolute uncertainty ($= \sqrt{N}$) will increase and the relative uncertainty ($= \frac{1}{\sqrt{N}}$) will decrease. This can be demonstrated by comparing the relative sizees of the errors on Histogram 1, where N = 1,000, with those on a histogram where N = 10,000 (*Histogram 2):*

```
x2 = rand(10000, 1); %10,000 random numbers, 0-1

figure
hist(x2, 20); %20 bins
hold on
[countsx2, binsx2] = hist(x2, 20); %assigning the counts and the bin numbers to variables
abs_uncx2 = sqrt(countsx2); %absolute uncetainties array
errorbar(binsx2, countsx2, abs_uncx2, abs_uncx2,'.'); %plotting errorbars: x, y, +ve, -ve
% unc.
xlabel('Random x2');
ylabel('Frequency density');
title('Histogram of 10,000 random numbers');
hold off
```



Histogram 2: Histogram of 10,000 random x values between 0 and 1 and their frequency

It can be seen that, relative to the bin height, the errors on Histogram 2 are smaller than on Histogram 1, showing that for increased N, the relative uncertainty is smaller. We can also demonstrate this by calculating the mean relative uncertainty of each dataset:

```
rel_uncx = abs_unc ./ countsx; %relative uncertainties array for x
```

```
rel_uncx2 = abs_uncx2 ./ countsx2; %for x2

mean_rel_uncs = [mean(rel_uncx); mean(rel_uncx2)]; %average relative uncertainties for
% x and x2
DataSet = {'x'; 'x2'};
N = {'1,000'; '10,000'};

unc_table = table(DataSet, N, mean_rel_uncs);
disp(unc_table)
```

DataSet	N	mean_rel_uncs
'x'	'1,000'	0.14243
'x2'	'10,000'	0.044754

Table 1: Table of mean relative uncertainties for x and x2 where N = 1,000 and N = 10,000 respectively

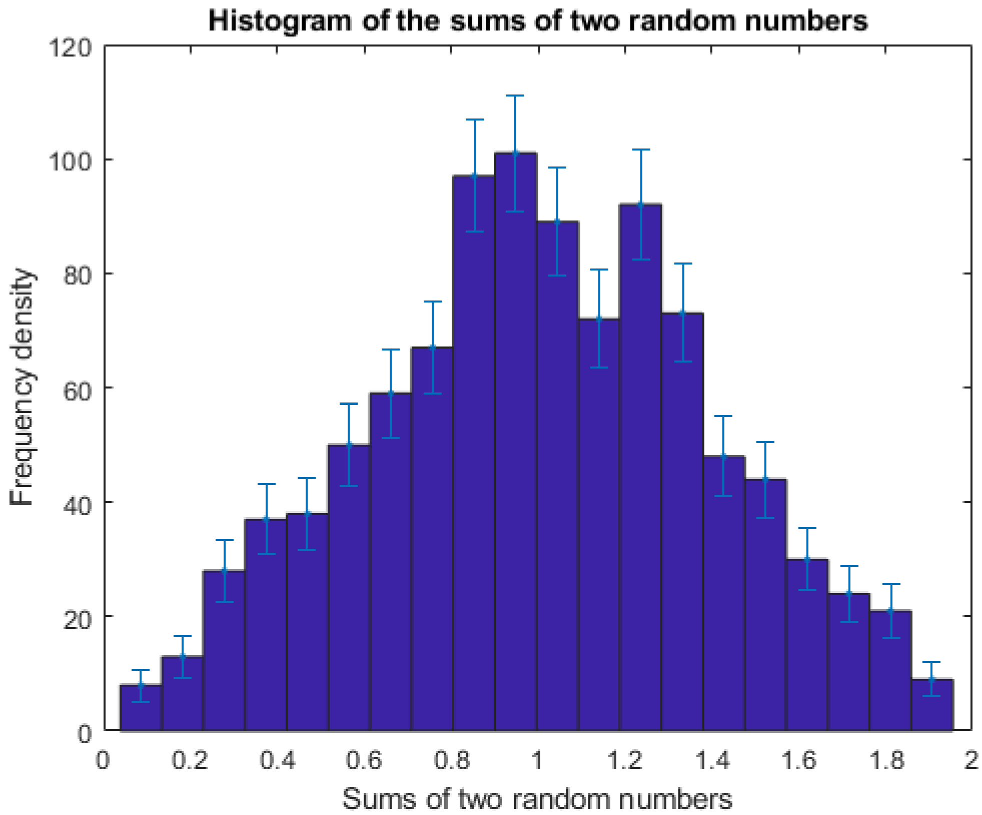
The average relative uncertainty for x2 (N = 10,000 numbers) is smaller than that for x (N = 1,000 numbers), showing that as N increases, the relative uncertainty decreases.

Question 1b

For the following two sections, a function was created to generate sets (of a given size, of a given number) of random numbers and sum them. This was called 'randomSetSum().'

The function can be seen in *Appendix 1* at the end of this report (MATLAB Live editor requires that functions come at the end of the document).

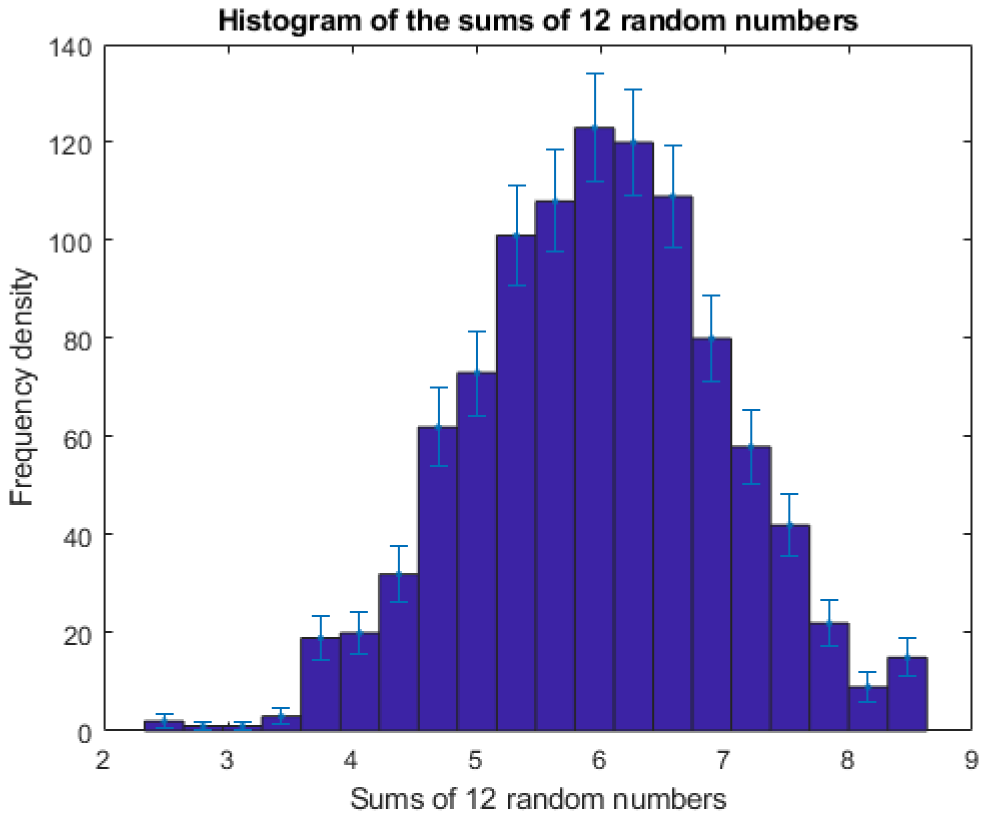
```
[~, sum1] = randomSetSum(1000, 2); %1000 pairs of random numbers (10 sets of 2 numbers)
figure
hist(sum1, 20);
hold on
[counts1, bins1] = hist(sum1, 20);
abs_unc1 = sqrt(counts1); %absolute uncertainties
errorbar(bins1, counts1, abs_unc1, abs_unc1,'.'); %plotting errorbars
xlabel('Sums of two random numbers');
ylabel('Frequency density');
title('Histogram of the sums of two random numbers');
hold off
```



Histogram 3: Histogram of 1,000 sums of pairs of random numbers and their frequency

Question 1c

```
[~, sum2] = randomSetSum(1000, 12); %1000 sets of 12 random numbers. The sum of each set
% stored as 'sum2'
figure
hist(sum2, 20);
hold on
[counts2, bins2] = hist(sum2, 20);
abs_unc2 = sqrt(counts2); %absolute uncertainties
errorbar(bins2, counts2, abs_unc2, abs_unc2, '.'); %plotting errorbars
xlabel('Sums of 12 random numbers');
ylabel('Frequency density');
title('Histogram of the sums of 12 random numbers');
hold off
```



Histogram 4: Histogram of the sums of 1,000 sets of 12 random numbers and their frequency

Question 1d

```
Mean = [mean(x); mean(sum1); mean(sum2)];
Variance = [var(x); var(sum1); var(sum2)];
Standard_deviation = [std(x); std(sum1); std(sum2)];
Data_set = {'Single (x)'; 'Pairs'; 'Sets of 12'};

tableResults = table(Data_set, Mean, Variance, Standard_deviation);
disp(tableResults)
```

Data_set	Mean	Variance	Standard_deviation
'Single (x)'	0.50667	0.08122	0.28499
'Pairs'	1.0036	0.15797	0.39745
'Sets of 12'	5.9796	1.04	1.0198

Table 2: Table of results for each data set showing the mean, variance and standard deviation of each

Question 1e

The shapes of the distribution become more peaked as the number of numbers being summed, N, increases; they appear more Gaussian. As can be seen from Table 2, the standard deviation approaches 1 as N increases.

Central limit theorem establishes that, when independent random variables are added, their sum tends towards a normal distribution. As the number being added increases, the standard deviation tends towards 1 and the spread of the curve decreases. This agrees with the observations above.

In the physical sciences, this means that the larger the data set collected, the smaller the uncertainty.

Question 2:

'Generate 6 random numbers from 1-49 inclusive for your lottery ticket. Rember that you can't use the same lottery number twice in one draw. Order these smallest to largest.'

```
q2 = randperm(49, 6); %a row vector of 6 random numbers, 1-49 inclusive, of unique
% values (no repetitions)
q2 = sort(q2); %'sort' function sorts the numbers, smallest to largest
disp('Your lottery numbers are: ')
```

```
Your lottery numbers are:
```

```
disp(q2)
```

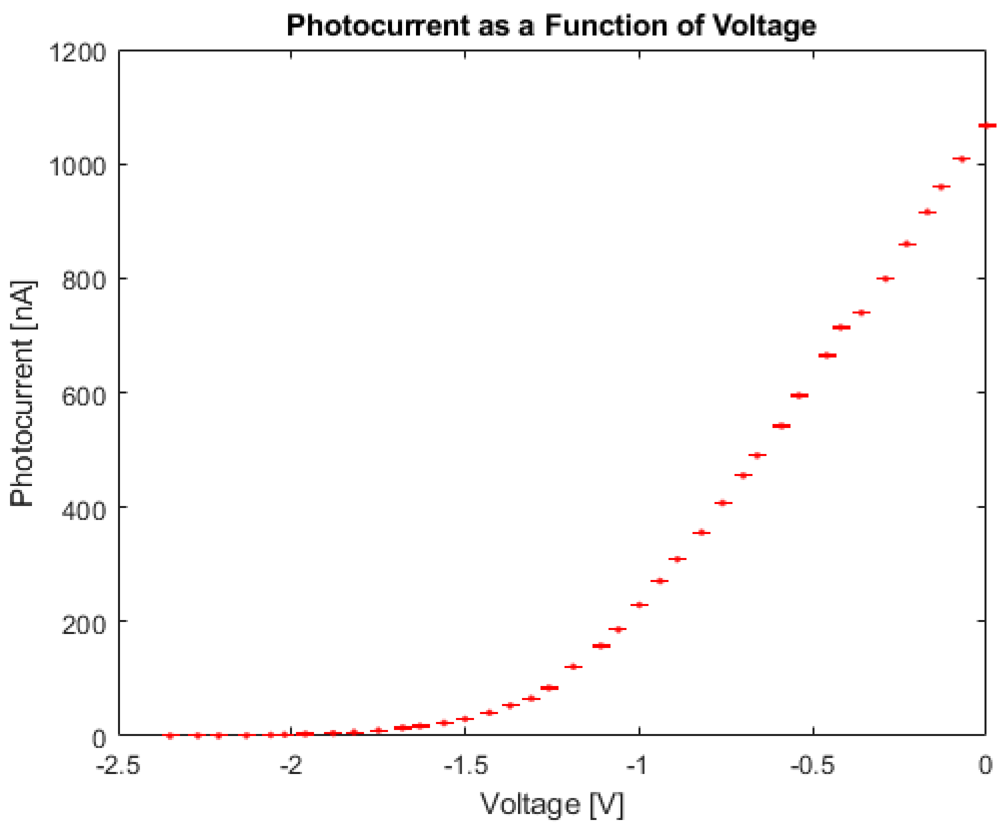
```
     6    13    14    16    22    28
```

Question 3

'Plot the data for photocurrent vs voltage and the uncertainty in voltage given in the Plank_raw.csv file.'

```
plank_data = csvread('Plank_raw.csv');
V = plank_data(:,1); %voltage
Ip = plank_data(:,2); %photocurrent
dIp = plank_data(:,3); %unc. in photocurrent
n = length(V); %number of entries

figure
plot(V, Ip, '.r'); %plotting the plank data
hold on
ylim([0, 1200]) %setting a limit on the y axis
errorbar(V, Ip, dIp, '.r') %plotting the errorbars. Very small error in y (0.5) not
% visible
xlabel('Voltage [V]');
ylabel('Photocurrent [nA]');
title('Photocurrent as a Function of Voltage');
```



Plot 1: Plot of photocurrent vs voltage for the Plank data

Appendix 1

This function was used to generate sets of random numbers of a given size in Question 1

```
function [random, column_sum] = randomSetSum(sets, numbers)
%gives a random number matrix of 'sets' sets of 'numbers' numbers ('random') and sums
%each set ('column_sum')
random = rand(numbers, sets); %rows, columns. Numbers between 0-1 by default
column_sum = sum(random); %the sums of each set
end
```