Marwin B. Alejo 2020-20221 EE214_Module2-LabEx4

Table of Contents

*

I. Joint Distribution

```
N=2000; a=1; b=0;
x=a.*randn(N,1)+b;
y=a+(b-a).*rand(N,1);
figure; scatterhist(x,y); title('Fig.1: Histogram and Scatterplot of x and
 y');
figure; histogram2(x,y); title('Fig.2: Joint Histogram of x,y'); xlabel('x
 histogram'); ylabel('y historgram'); zlabel('Pr(x|y)')
```

Fig.1: Histogram and Scatterplot of x and y

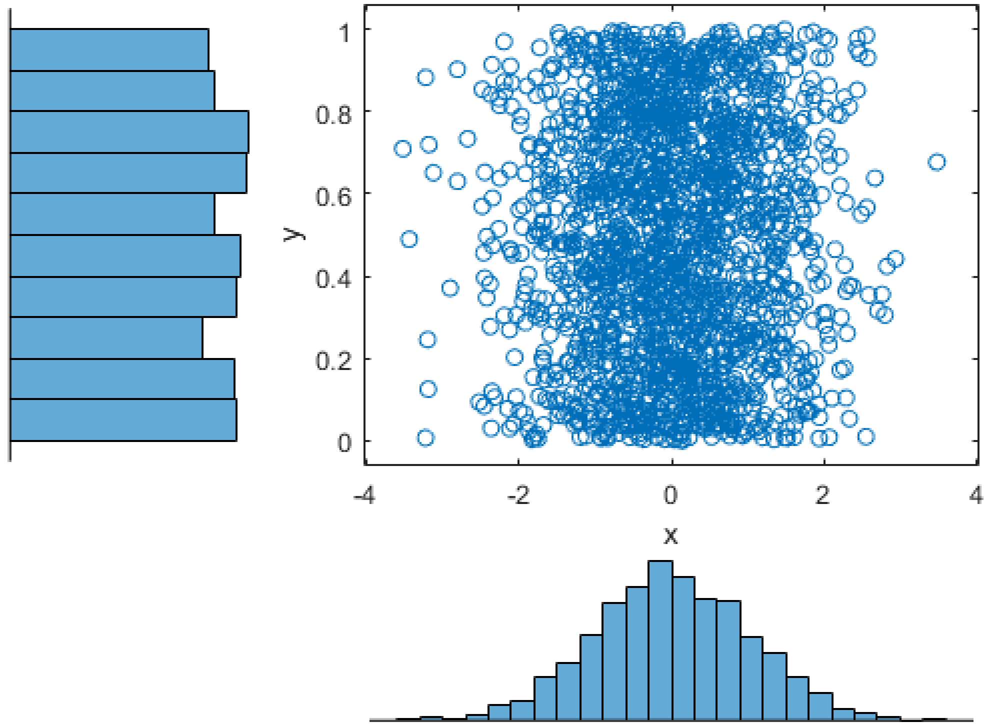
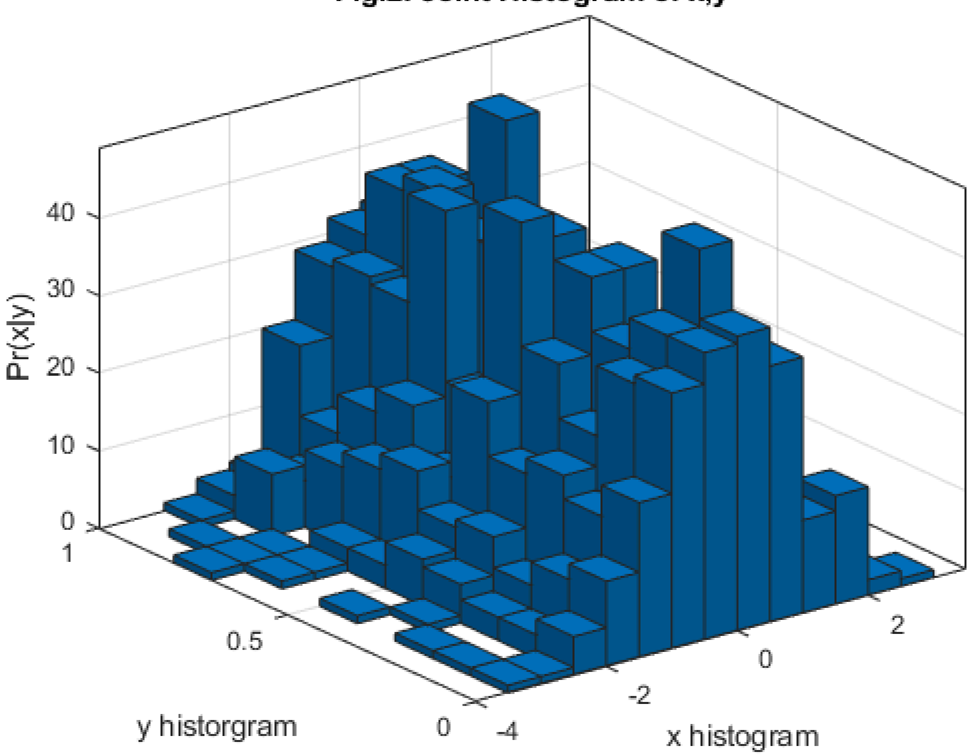


Fig.2: Joint Histogram of x,y



Given that x contain normally distributed numbers and y with uniformly distributed numbers, it is expected that their scatter plot linearly or vertically moves closer to x=0 and instead of a circular-shaped scatterplot. Figure 1 proves this statement with x having a bell-shaped histogram with most of the samples are scattered at the mean '0' while y having samples distributed between 0 and 1.

Figure 2 on the other hand shows the joint histogram of x and y with its x-axis represent the histogram of x and the y-axis represent the histogram of y. The top-view of Figure 2 show the equivalent plot of the shown scatterplot in Figure 1. Hence, the joint histogram in Figure 2 is consistent with the scatterplot in Figure 1.

Note: scatterplot and its variant functions of MATLAB returns the marginal distribution of x,y.

yk=0.5 with 0.1 tolerance

```
% generate (xk,yk) with y0-delta-y<yk<y0+delta-y; y0=0.5 and delta-y=0.1; 0.4
 and 0.6
yk=zeros(sum(y>0.4 & y<0.6),1);xk=zeros(sum(y>0.4 & y<0.6),1);
for count=1:length(y)
    if (y(count)>0.4) && (y(count)<0.6)
        yk(count,1)=y(count); xk(count,1)=x(count);
    end
end

% remove cells containing 0
yk(yk(:,1)==0,:)=[]; xk(xk(:,1)==0,:)=[];

% plot histogram
figure; scatterhist(xk,yk); title('Fig.3: Scatter-Hist of P(x|y) w/ y0=0.5 &
 tlrnce=0.1');
figure; histogram2(xk,yk); title('Fig.4: Joint Histogram of P(x|y) w/ y0=0.5
 & tlrnce=0.1'); xlabel('x histogram'); ylabel('y historgram'); zlabel('Pr(x|
y)')
```

Fig.3: Scatter-Hist of P(x|y) w/ y0=0.5 & tlrnce=0.1

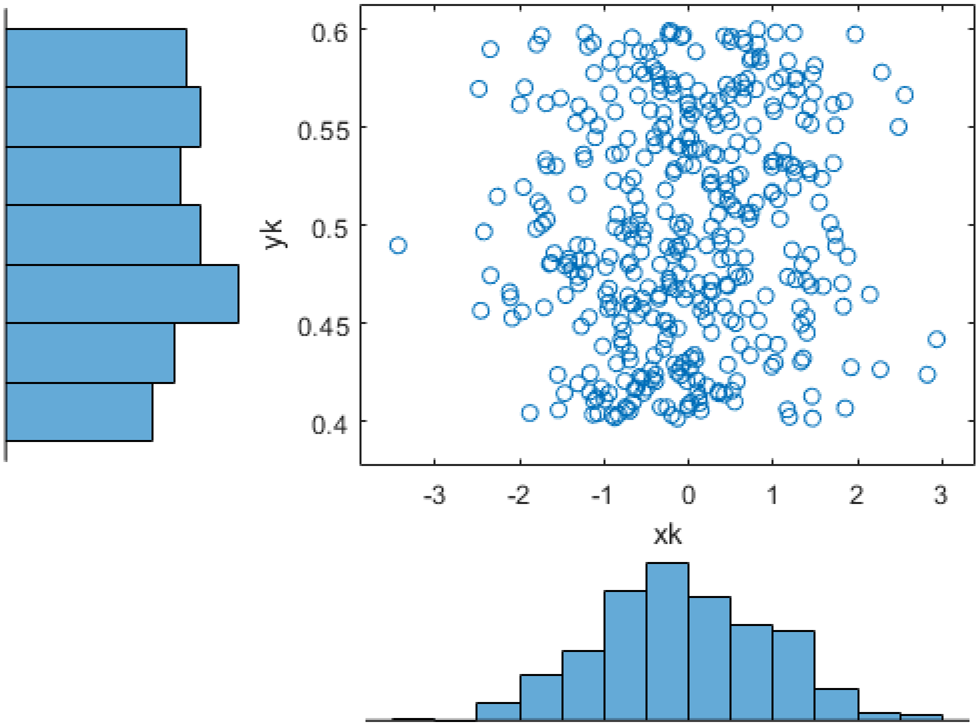
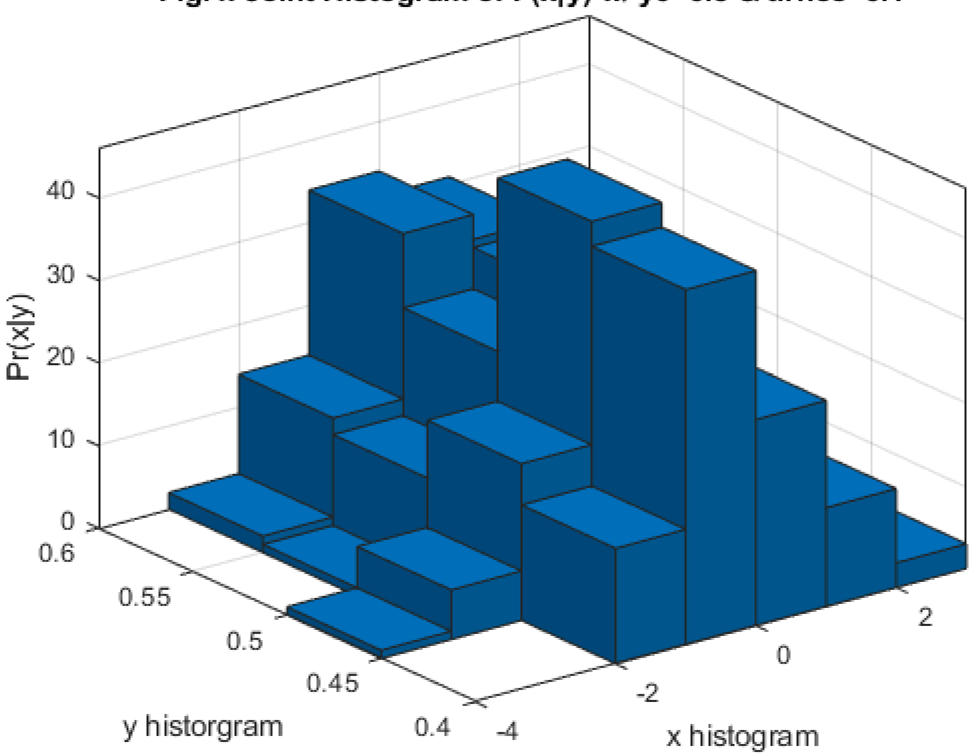


Fig.4: Joint Histogram of P(x|y) w/ y0=0.5 & tlrnce=0.1



Figures 3 and 4 show the histogram of the selected samples given y0=0.5 and a tolerance of 0.1 for P(x|y). The histogram or the marginal distribution of x in figures 3 and 4 differs from marginal distribution of x in figures 1 and 2 in a way that the number of samples are trimmed between the threshold of y>0.4 and y<0.6. Generally, the historgram or marginal distribution of both are still the same in terms of the distribution shape of both x, y, and the scatterplot (top-view).

yk=0.5 without tolerance

```
% code to generate (xk,yk) with yk=0.5
% yk=zeros(sum(y==0.5),1);xk=zeros(sum(y==0.5),1);
% for count=1:length(y)
%     if (y(count)==0.5)
%         yk(count,1)=y(count); xk(count,1)=x(count);
%     end
% end
%
% % remove cells containing 0
% yk(yk(:,1)==0,:)=[]; xk(xk(:,1)==0,:)=[];
%
% % plot histogram
% figure; scatterhist(xk,yk); title('Fig.5: Scatter-Hist of P(x|y) w/
 y0=0.5');
% figure; histogram2(xk,yk); title('Fig.6: Joint Histogram of P(x|y) w/
 y0=0.5'); xlabel('x histogram'); ylabel('y historgram'); zlabel('Pr(x|y)')
```

This condition is possible when there are elements equal to 0.5 in array y. If this happens, only the samples that suffice the condition y==0.5 will be generated and plotted. The x and y curves will still be the same only that the samples plotted in the curve are those that suffices the given condition (similar to above plots but with fewer samples or bars).

```
% P.S. this condition rarely happen to randomly generated numbers, hehe.
```

II. Central Limit Theorem

```
% read BPSYS.txt
text = textread('BPSYS.txt');
```

compute and display mean and standard deviation of BPSYS.txt

Mean of BPSYS.txt

```
mean(text)
```

```
ans =

  125.0069
```

Standard deviation of BPSYS.txt

```
std(text)
```

```
ans =

   18.4117
```

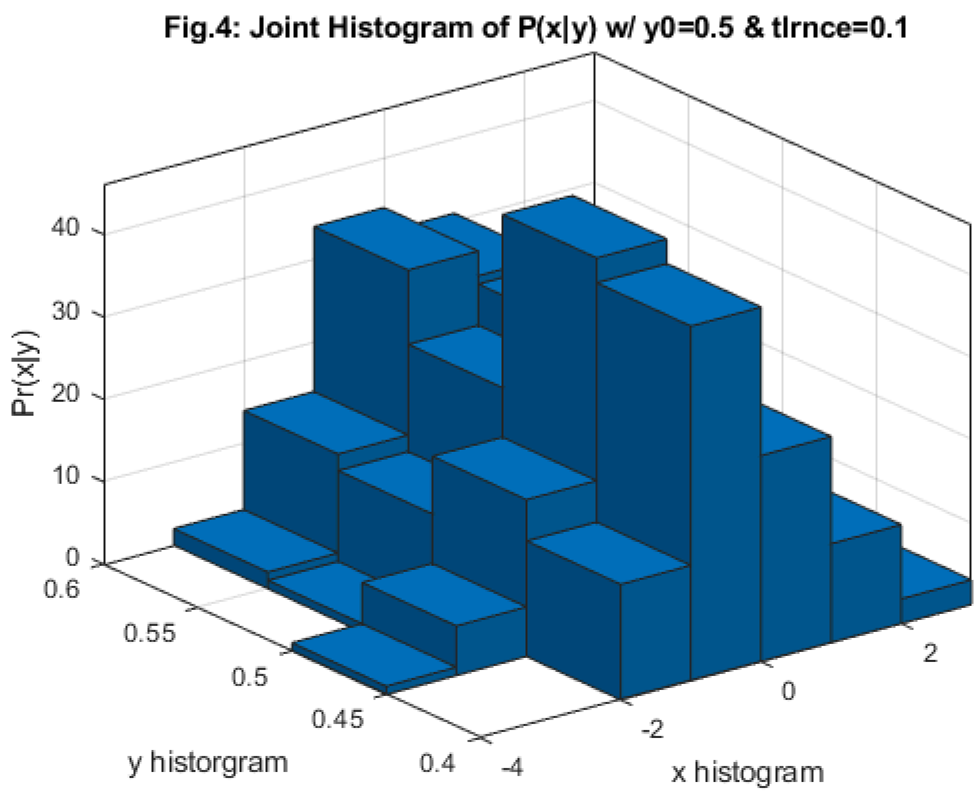
Random variable of BPSYS.txt may be computed using ksdensity() fxn.

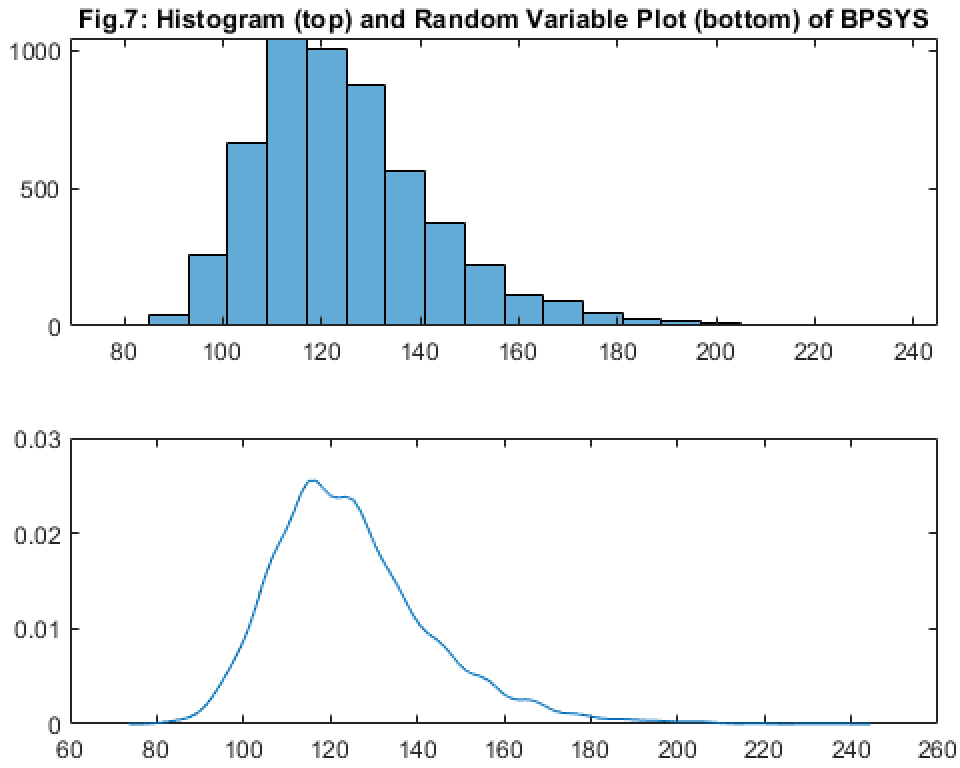
```
[f,ii]=ksdensity(text);
f


f =

  Columns 1 through 7

    0.0000    0.0000    0.0000    0.0000    0.0001    0.0002    0.0003

  Columns 8 through 14

    0.0005    0.0007    0.0011    0.0017    0.0027    0.0039    0.0052

  Columns 15 through 21

    0.0066    0.0082    0.0100    0.0124    0.0150    0.0172    0.0188

  Columns 22 through 28

    0.0203    0.0222    0.0242    0.0255    0.0255    0.0247    0.0239

  Columns 29 through 35

    0.0237    0.0238    0.0235    0.0223    0.0204    0.0185    0.0170

  Columns 36 through 42

    0.0157    0.0143    0.0128    0.0113    0.0102    0.0095    0.0089

  Columns 43 through 49

    0.0082    0.0073    0.0064    0.0056    0.0052    0.0050    0.0046

  Columns 50 through 56

    0.0039    0.0032    0.0027    0.0025    0.0026    0.0025    0.0021

  Columns 57 through 63

    0.0017    0.0013    0.0012    0.0011    0.0011    0.0009    0.0007

  Columns 64 through 70

    0.0006    0.0005    0.0005    0.0005    0.0005    0.0004    0.0004

  Columns 71 through 77
```

```
   0.0004    0.0003    0.0003    0.0002    0.0002    0.0002    0.0002

  Columns 78 through 84

   0.0002    0.0002    0.0001    0.0001    0.0000    0.0000    0.0001

  Columns 85 through 91

   0.0001    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000

  Columns 92 through 98

   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000

  Columns 99 through 100

   0.0000    0.0000
```

plot the histogram and random variable

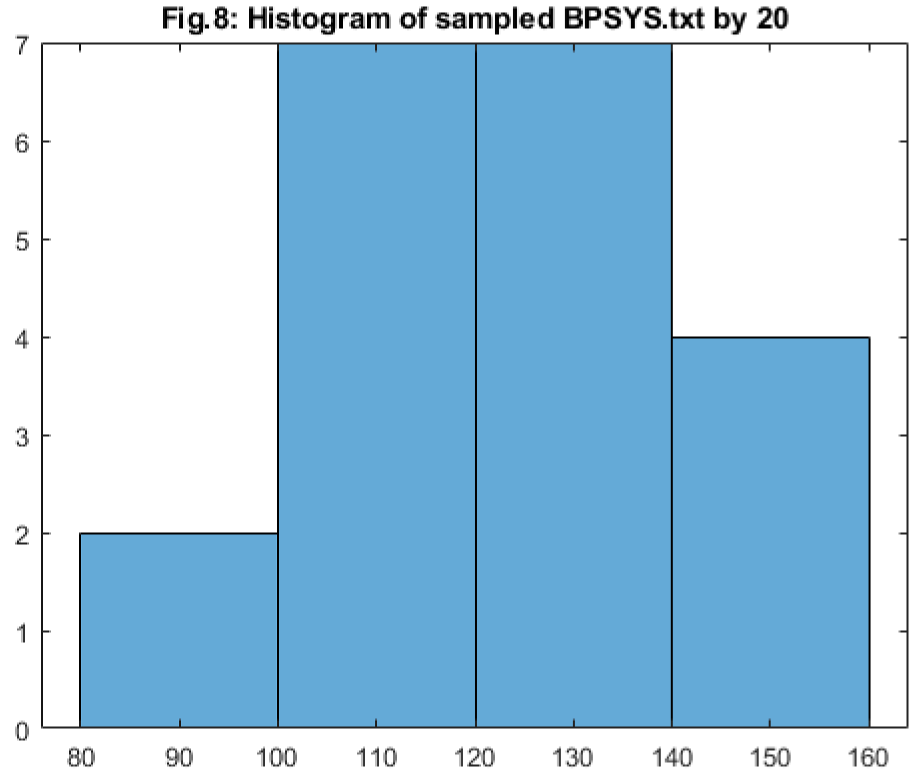
```
figure; subplot(2,1,1); histogram(text,20); title('Fig.7: Histogram (top) and
 Random Variable Plot (bottom) of BPSYS'); subplot(2,1,2); ksdensity(text);
```





datasample n=20 of BPSYS.txt

```
sampleData = datasample(text,20);
figure; histogram(sampleData); title('Fig.8: Histogram of sampled BPSYS.txt by
 20');
```



Mean of sampleData

```
mean(sampleData)
```

```
ans =

  121.8000
```

Standard deviation of sampleData

```
std(sampleData)
```

```
ans =

   15.6124
```

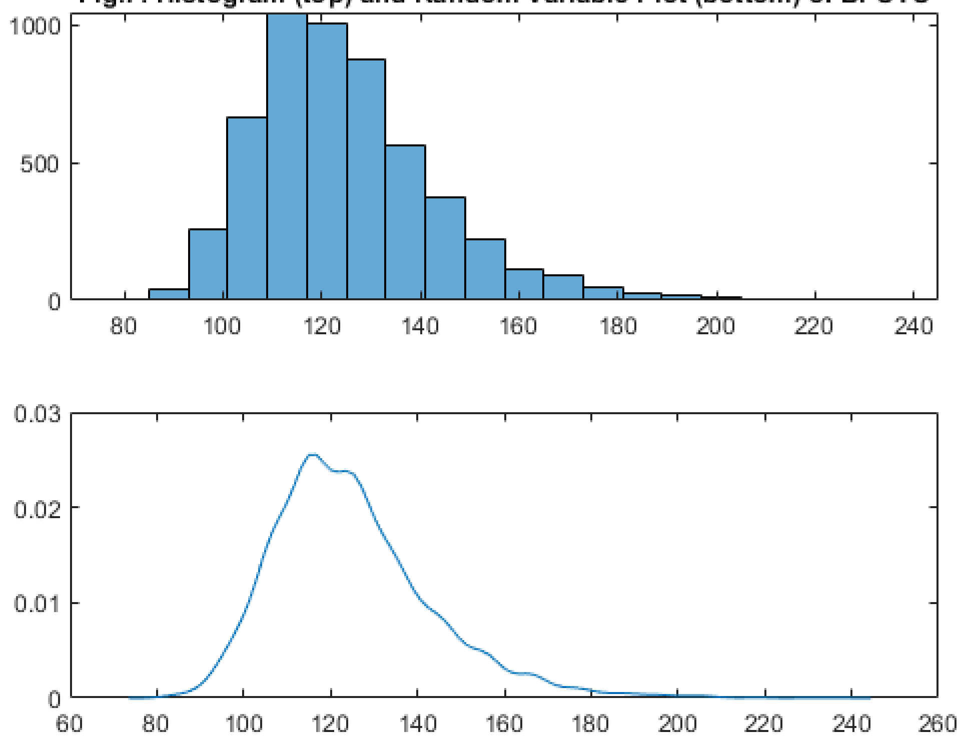
Figure 8 shows the histogram of BPSYS.txt population sampled by n=20. It yields a mean of 128.90 and standard deviation of 25.7966. The plot is expected to be theoretically as is since the sample is not properly distributed normally or uniformly. The yielded mean represent the highest peak of the generated sample which is within the bar sample of 120-140 with a std of ~30.

n=20 N=100

Independent and identically distributed (iid) random variables

```
n = 20; N = 100;
iid=zeros(n,N); % initialize iid mtx
for ctr=1:N
    iid(:,ctr) = datasample(text,n);
end

% mean of iid
iid = iid.';
iidMean = mean(iid);
iidMean = iidMean.';

% histogram of iidMean
figure; histogram(iidMean); title('Hitogram of average iid')
```

Fig.7: Histogram (top) and Random Variable Plot (bottom) of BPSYS



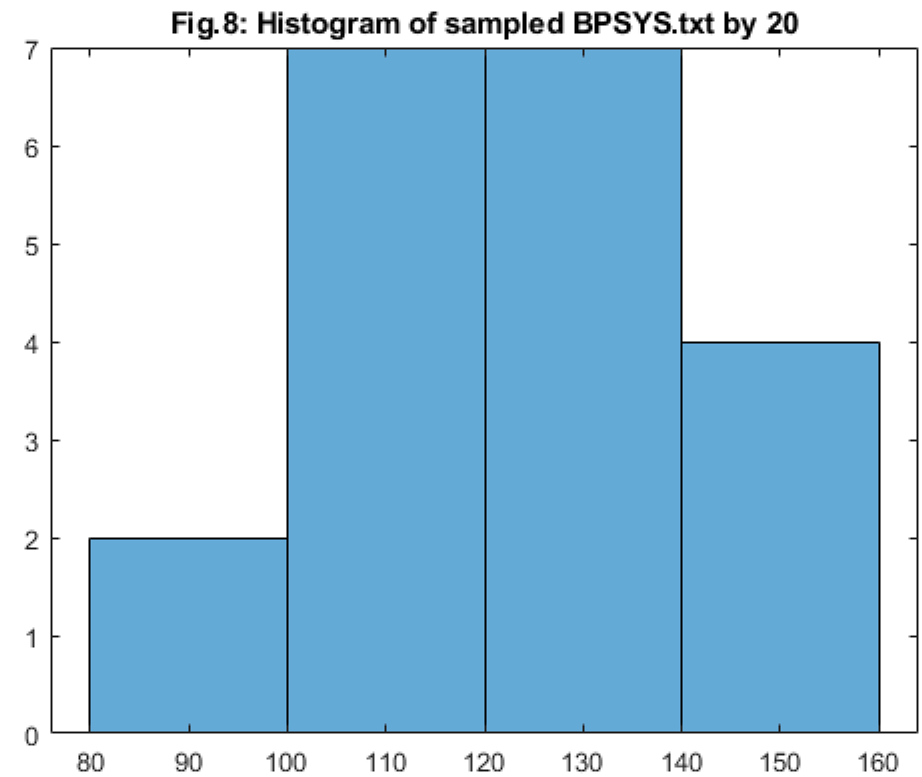
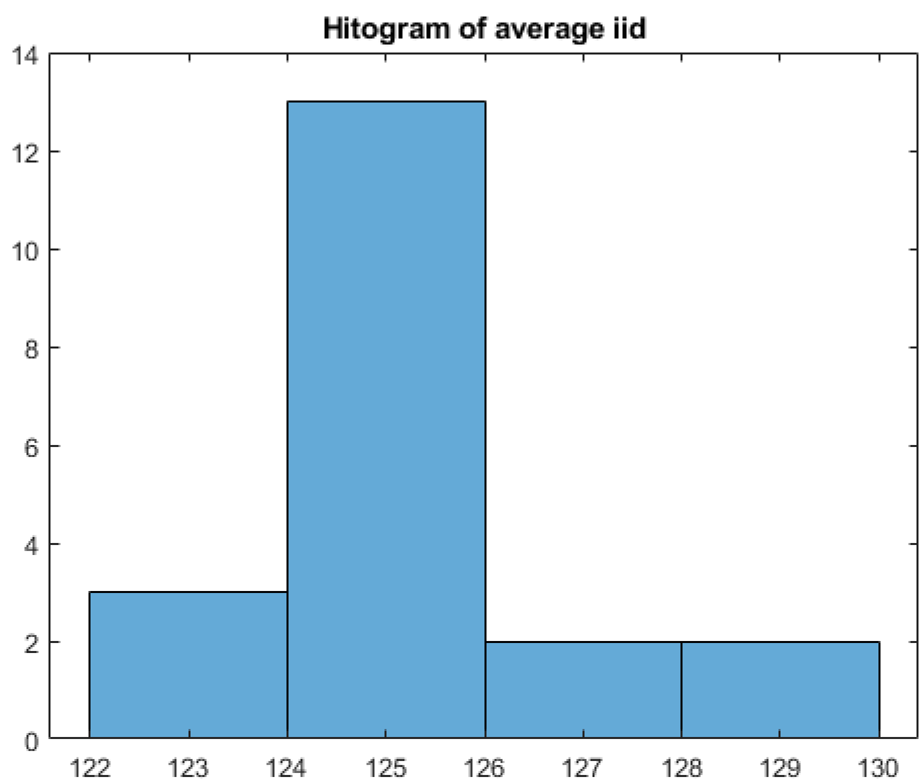


Fig.8: Histogram of sampled BPSYS.txt by 20
0
1
2
3
4
5
6
7
80
90
100
110
120
130
140
150
160


Hitogram of average iid
0
2
4
6
8
10
12
14
122
123
124
125
126
127
128
129
130

Mean of average iid

```
mean(iidMean)
```

```
ans =

  125.3010
```

standard deviation of iid

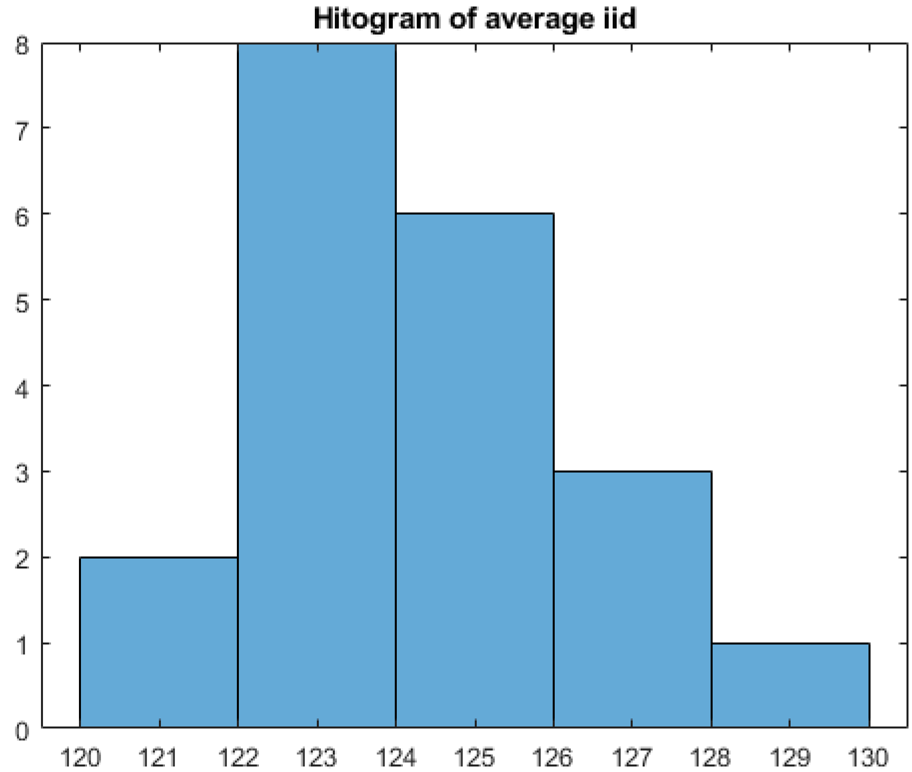
```
std(iidMean)
```

```
ans =

    1.5734
```

n=10 N=1000

Independent and identically distributed (iid) random variables

```
n = 20; N = 100;
iid=zeros(n,N); % initialize iid mtx
for ctr=1:N
    iid(:,ctr) = datasample(text,n);
end

% mean of iid
iid = iid.';
iidMean = mean(iid);
iidMean = iidMean.';

% histogram of iidMean
figure; histogram(iidMean); title('Hitogram of average iid')
```



Mean of average iid

```
mean(iidMean)
```

ans =

124.4140

standard deviation of iid

```
std(iidMean)
```

ans =

1.7840

Given the above results of section II - central limit theorem, the mean and standard deviation of the sampled BPSYS.txt by n for N-times are expected. The mean specifies the center of the largest sample while the standard deviation shows the step from the centroid of the sample bar to another. If the mean and std are in float value, the graph represent it by its round-off integer value. Nevertheless, the same values are ofcourse expected. Also, changing the values of n and N alter the values of mean and std which affect the histogram and the representation of the mean and std graphically. By technical observation, the samples are normally distributed.

Conclusion:

This exercise allowed me to explore joint probability and central limit theorem on application basis through MATLAB. In joint probability, I realized that given a random variable defined in a probability space, a general probability distribution indicates the chnace that each sample will fall within a [articular range or set of individual values specified for a variable. In central limit theorem, it states that there is a population with mean mu and stadard deviation sigma, and given these parameters the distribution becomes equivalently normal.

Published with MATLAB® R2021b