CMPE362 - HOMEWORK 1

Question 1:

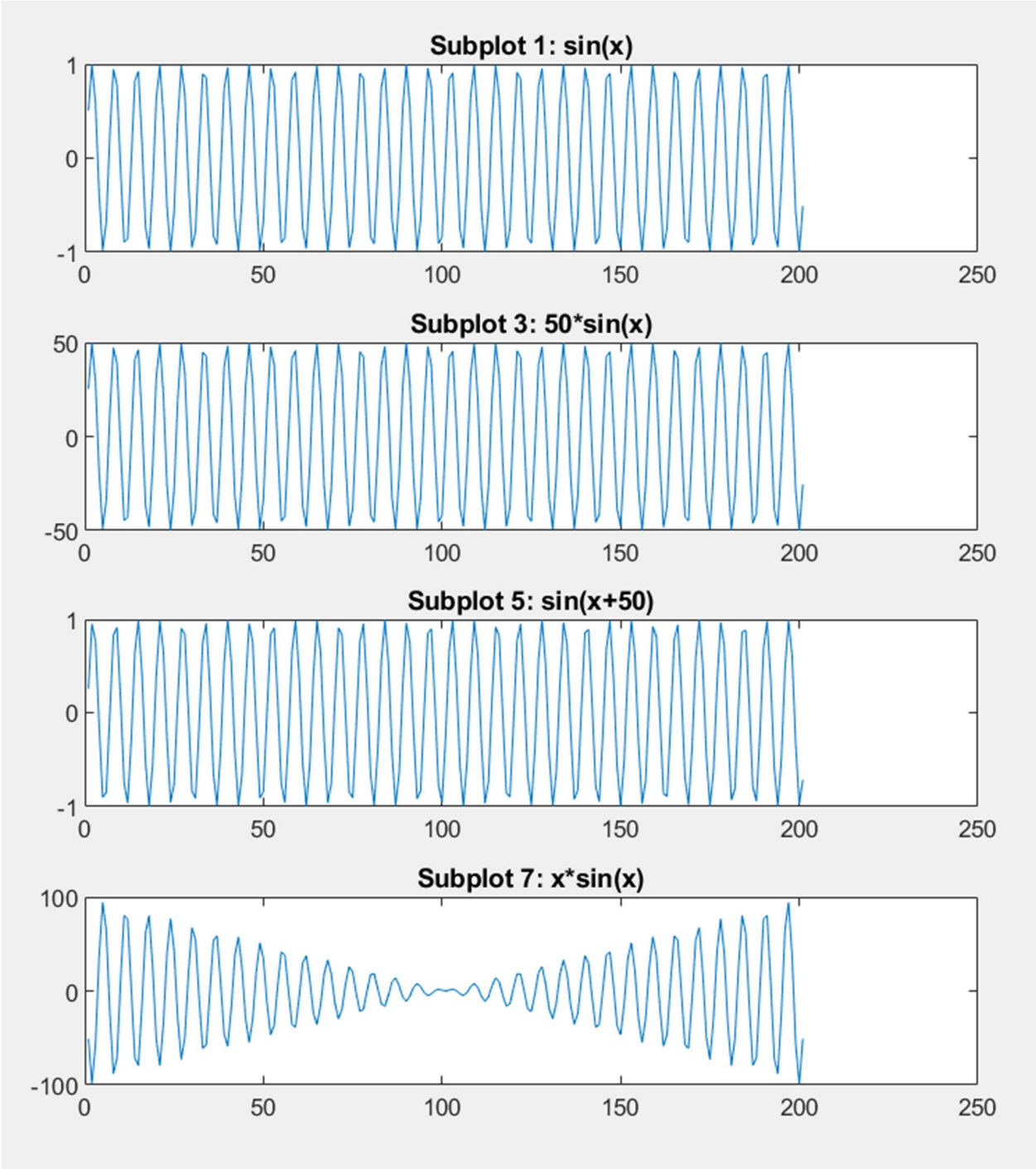
Code Snippet:

```
% ------------------------ Question 1 ------------------------

 x = -100:1:+100;

 subplot(4,2,1)
 y1 = sin(x);
 plot(y1)
 title('Subplot 1: sin(x)')

subplot(4,2,2);
y2 = sin(50*x);
plot(y2)
title('Subplot 2: sin(50*x)')


subplot(4,2,3);
y3 = 50.*sin(x);
plot(y3)
title('Subplot 3: 50*sin(x)')

subplot(4,2,4);
y4 = sin(x)+50;
plot(y4)
title('Subplot 4: sin(x)+50')

subplot(4,2,5);
y5 = sin(x+50);
plot(y5)
title('Subplot 5: sin(x+50)')

subplot(4,2,6);
y6 = 50.*sin(50*x);
plot(y6)
title('Subplot 6: 50*sin(50*x)')

subplot(4,2,7);
y7 = x.*sin(x);
plot(y7)
title('Subplot 7: x*sin(x)')

subplot(4,2,8);
y8 = sin(x)./x;
plot(y8)
title('Subplot 8: sin(x)/x')
```

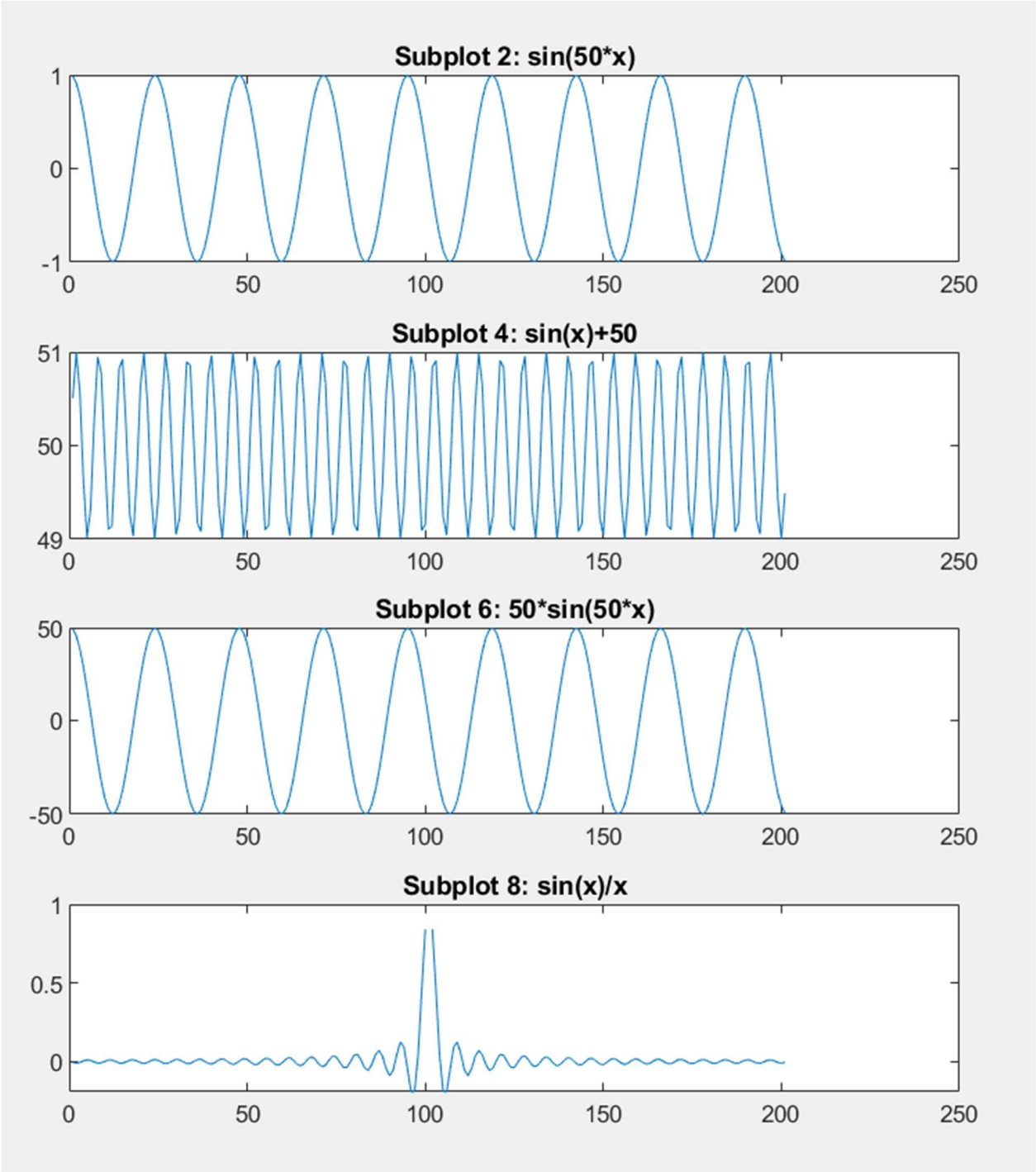
What did I learn?

I learned how adding a number to sin and multiplying sin with a number affect sin function. The differences are so clear in the figures below.

Figures (Q1)


Subplot 1: sin(x)
1
0
-1
0
50
100
150
200
250
Subplot 3: 50*sin(x)
50
0
-50
0
50
100
150
200
250
Subplot 5: sin(x+50)
1
0
-1
0
50
100
150
200
250
Subplot 7: x*sin(x)
100
0
-100
0
50
100
150
200
250

Figures (Q1):


Subplot 2: sin(50*x)
1
0
-1
0
50
100
150
200
250
Subplot 4: sin(x)+50
51
50
49
0
50
100
150
200
250
Subplot 6: 50*sin(50*x)
50
0
-50
0
50
100
150
200
250
Subplot 8: sin(x)/x
1
0.5
0
0
50
100
150
200
250

Question 2:

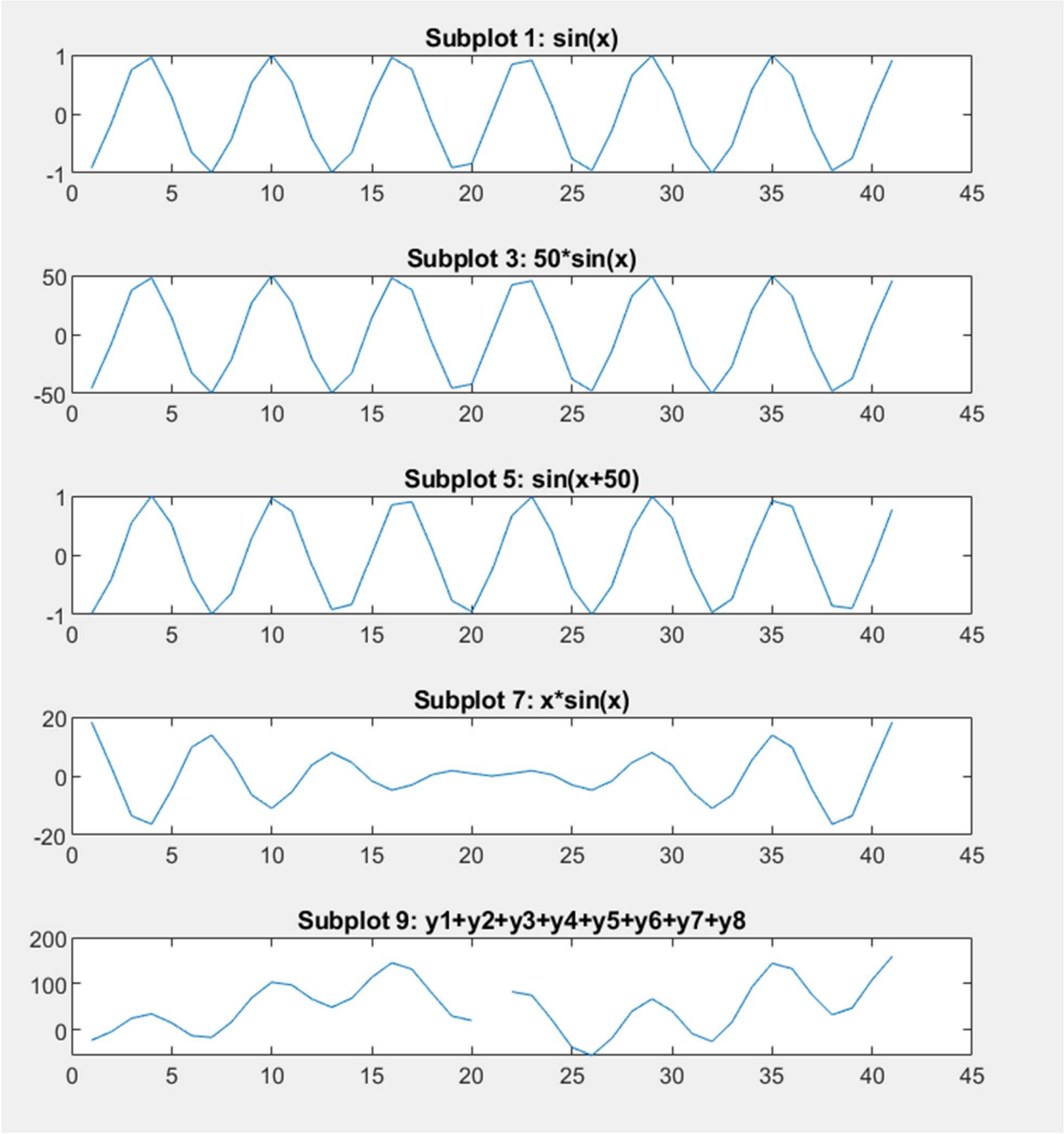
Code Snippet:

```
% ------------------------ Question 2 ------------------------

 x = -20:1:+20;

 subplot(5,2,1)
 y1 = sin(x);
 plot(y1)
 title('Subplot 1: sin(x)')

 subplot(5,2,2);
 y2 = sin(50*x);
 plot(y2)
 title('Subplot 2: sin(50*x)')

 subplot(5,2,3);
 y3 = 50.*sin(x);
 plot(y3)
 title('Subplot 3: 50*sin(x)')

 subplot(5,2,4);
 y4 = sin(x)+50;
 plot(y4)
 title('Subplot 4: sin(x)+50')

 subplot(5,2,5);
 y5 = sin(x+50);
 plot(y5)
 title('Subplot 5: sin(x+50)')

 subplot(5,2,6);
 y6 = 50.*sin(50*x);
 plot(y6)
 title('Subplot 6: 50*sin(50*x)')

 subplot(5,2,7);
 y7 = x.*sin(x);
 plot(y7)
 title('Subplot 7: x*sin(x)')

 subplot(5,2,8);
 y8 = sin(x)./x;
 plot(y8)
 title('Subplot 8: sin(x)/x')

 subplot(5,2,9);
 y9 = y1+y2+y3+y4+y5+y6+y7+y8;
 plot(y9)
 title('Subplot 9: y1+y2+y3+y4+y5+y6+y7+y8')
```

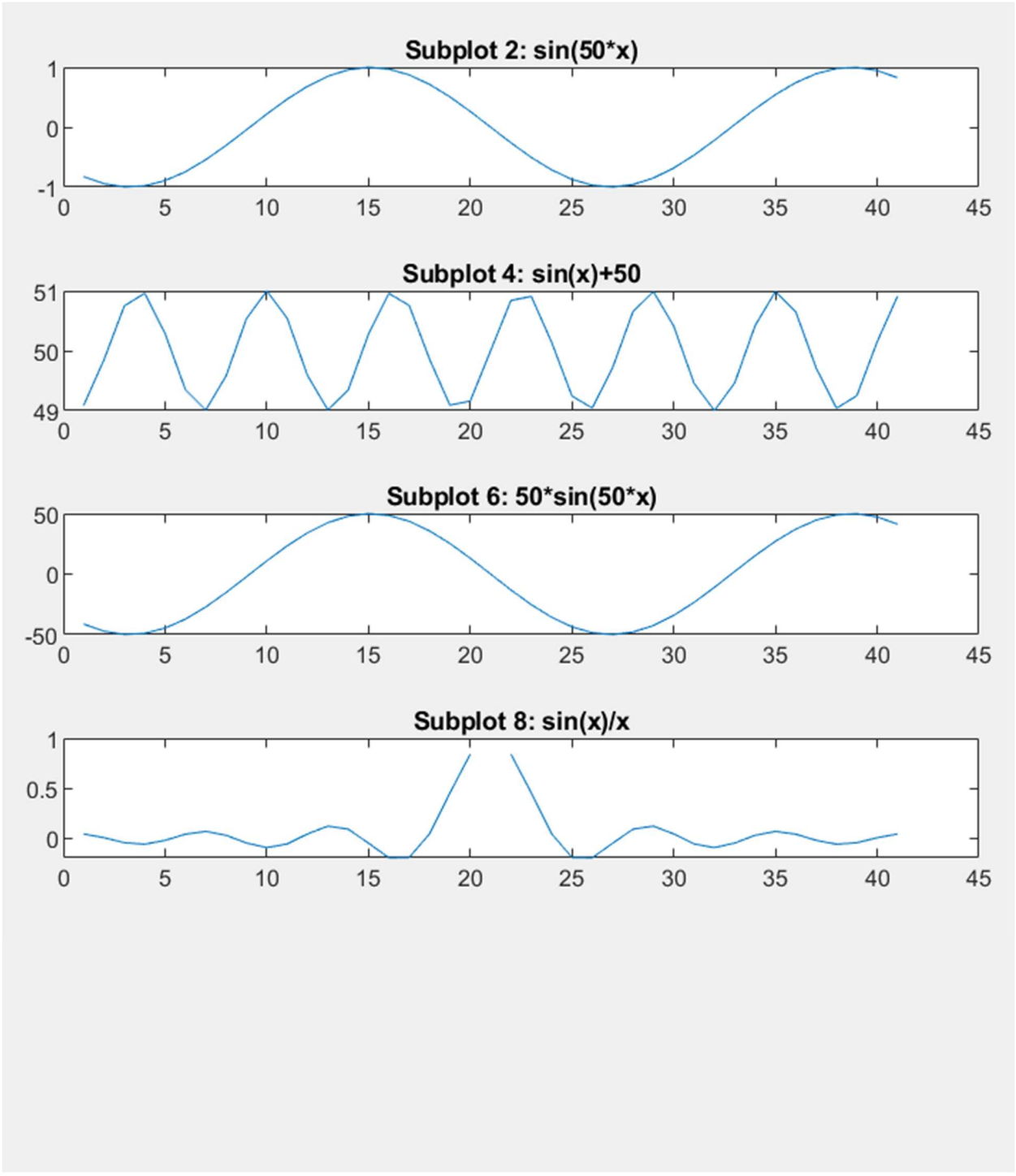
What did I learn?

This was not much different than the first question. I just reinforced what I had learned in Q1.

Figures (Q2):


Subplot 1: sin(x)
1
0
-1
0 5 10 15 20 25 30 35 40 45
Subplot 3: 50*sin(x)
50
0
-50
0 5 10 15 20 25 30 35 40 45
Subplot 5: sin(x+50)
1
0
-1
0 5 10 15 20 25 30 35 40 45
Subplot 7: x*sin(x)
20
0
-20
0 5 10 15 20 25 30 35 40 45
Subplot 9: y1+y2+y3+y4+y5+y6+y7+y8
200
100
0
0 5 10 15 20 25 30 35 40 45

Figures (Q2):


Subplot 2: sin(50*x)
1
0
-1
0 5 10 15 20 25 30 35 40 45
Subplot 4: sin(x)+50
51
50
49
0 5 10 15 20 25 30 35 40 45
Subplot 6: 50*sin(50*x)
50
0
-50
0 5 10 15 20 25 30 35 40 45
Subplot 8: sin(x)/x
1
0.5
0
0 5 10 15 20 25 30 35 40 45

Questions 3:

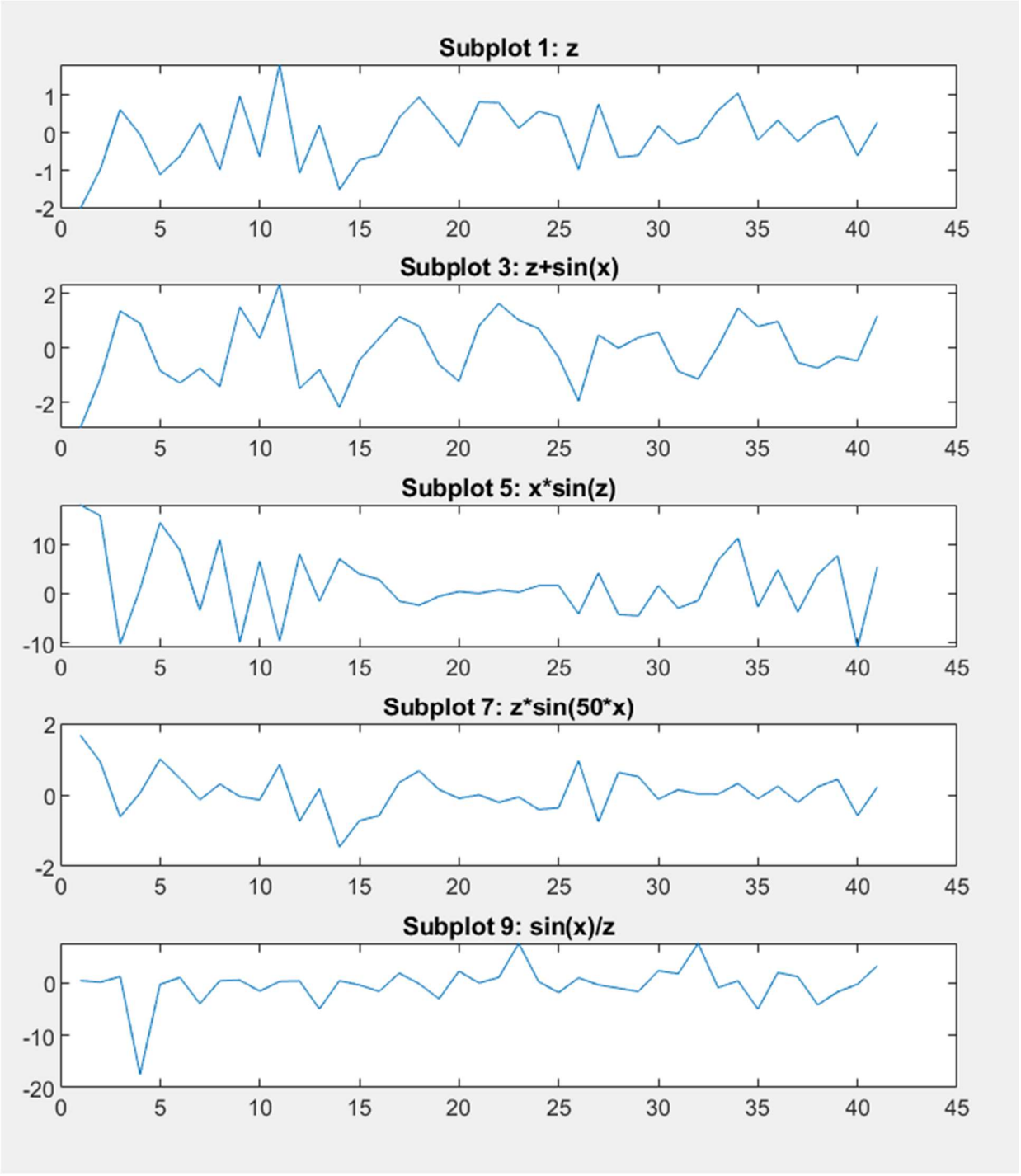
Code Snippet:

```
% ------------------------ Question 3 ------------------------

  z = randn(1, 41);

  subplot(5,2,1)
  y10 = z;
  plot(y10)
  title('Subplot 1: z')

  subplot(5,2,2);
  y11 = z + x;
  plot(y11)
  title('Subplot 2: z+x')

  subplot(5,2,3);
  y12 = z + sin(x);
  plot(y12)
  title('Subplot 3: z+sin(x)')

  subplot(5,2,4);
  y13 = z.*sin(x);
  plot(y13)
  title('Subplot 4: z*sin(x)')

  subplot(5,2,5);
  y14 = x.*sin(z);
  plot(y14)
  title('Subplot 5: x*sin(z)')

  subplot(5,2,6);
  y15 = sin(x+z);
  plot(y15)
  title('Subplot 6: sin(x+z)')

  subplot(5,2,7);
  y16 = z.*sin(50*x);
  plot(y16)
  title('Subplot 7: z*sin(50*x)')

  subplot(5,2,8);
  y17 = sin(x+50*z);
  plot(y17)
  title('Subplot 8: sin(x+50*z)')

  subplot(5,2,9);
  y18 = sin(x)./z;
  plot(y18)
  title('Subplot 9: sin(x)/z')

  subplot(5,2,10);
  y19 = y11+y12+y13+y14+y15+y16+y17+y18;
  plot(y19)
  title('Subplot 10: y11+y12+y13+y14+y15+y16+y17+y18')
```

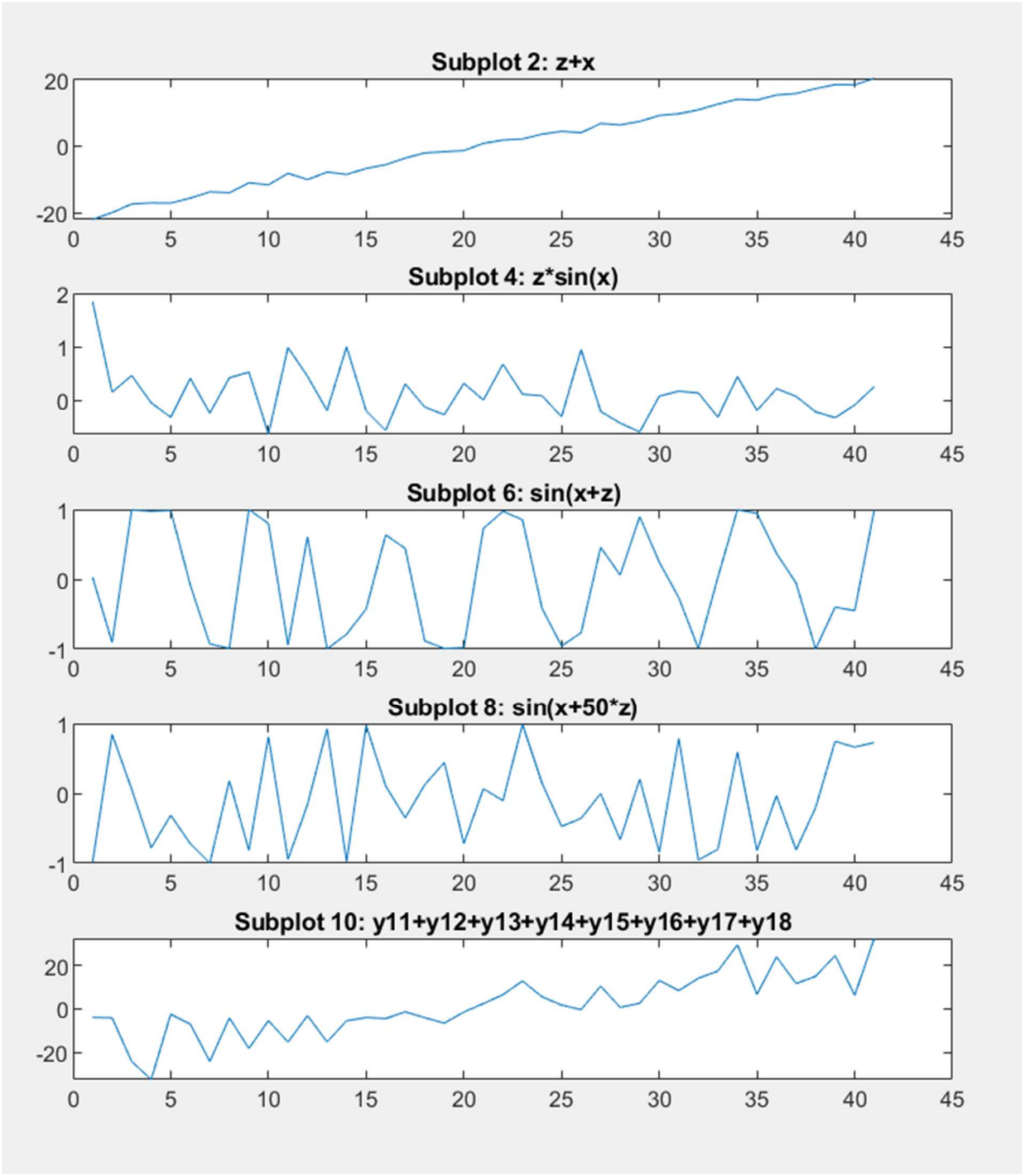
What did I learn?

I learned that how adding a Gaussian random variable to sin and multiplying sin with a Gaussian random variable affect the behavior of sin function.

Figures (Q3):



Figures (Q3):


Subplot 2: z+x
Subplot 4: z*sin(x)
Subplot 6: sin(x+z)
Subplot 8: sin(x+50*z)
Subplot 10: y11+y12+y13+y14+y15+y16+y17+y18

Questions 4:

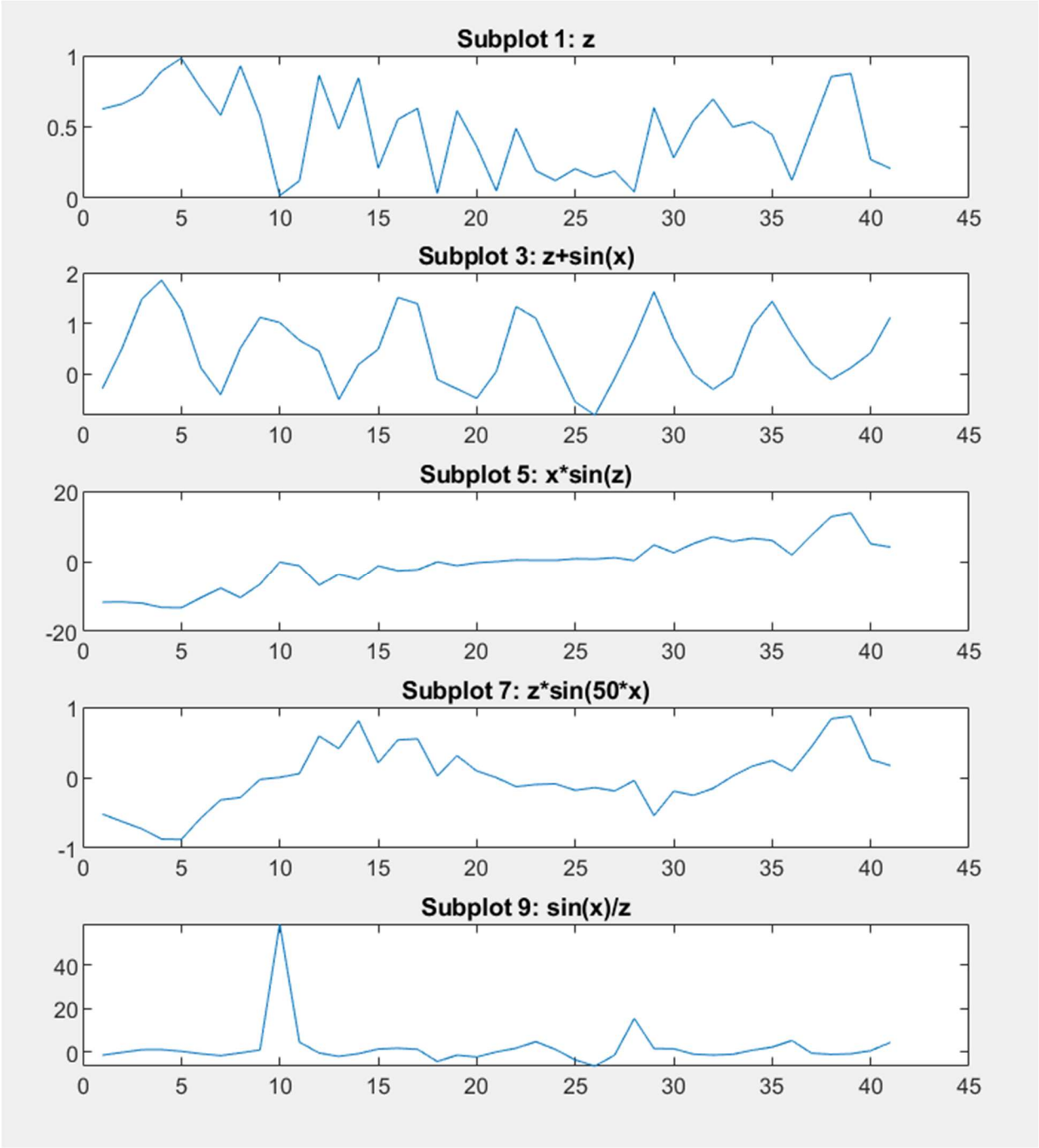
Code Snippet:

```
% ------------------------ Question 4 ------------------------

  z = rand(1,41);

  subplot(5,2,1)
  y20 = z;
  plot(y20)
  title('Subplot 1: z')

  subplot(5,2,2);
  y21 = z + x;
  plot(y21)
  title('Subplot 2: z+x')

  subplot(5,2,3);
  y22 = z + sin(x);
  plot(y22)
  title('Subplot 3: z+sin(x)')

  subplot(5,2,4);
  y23 = z.*sin(x);
  plot(y23)
  title('Subplot 4: z*sin(x)')

  subplot(5,2,5);
  y24 = x.*sin(z);
  plot(y24)
  title('Subplot 5: x*sin(z)')

  subplot(5,2,6);
  y25 = sin(x+z);
  plot(y25)
  title('Subplot 6: sin(x+z)')

  subplot(5,2,7);
  y26 = z.*sin(50*x);
  plot(y26)
  title('Subplot 7: z*sin(50*x)')

  subplot(5,2,8);
  y27 = sin(x+50*z);
  plot(y27)
  title('Subplot 8: sin(x+50*z)')

  subplot(5,2,9);
  y28 = sin(x)./z;
  plot(y28)
  title('Subplot 9: sin(x)/z')

  subplot(5,2,10);
  y29 = y21+y22+y23+y24+y25+y26+y27+y28;
  plot(y29)
  title('Subplot 10: y21+y22+y23+y24+y25+y26+y27+y28')
```

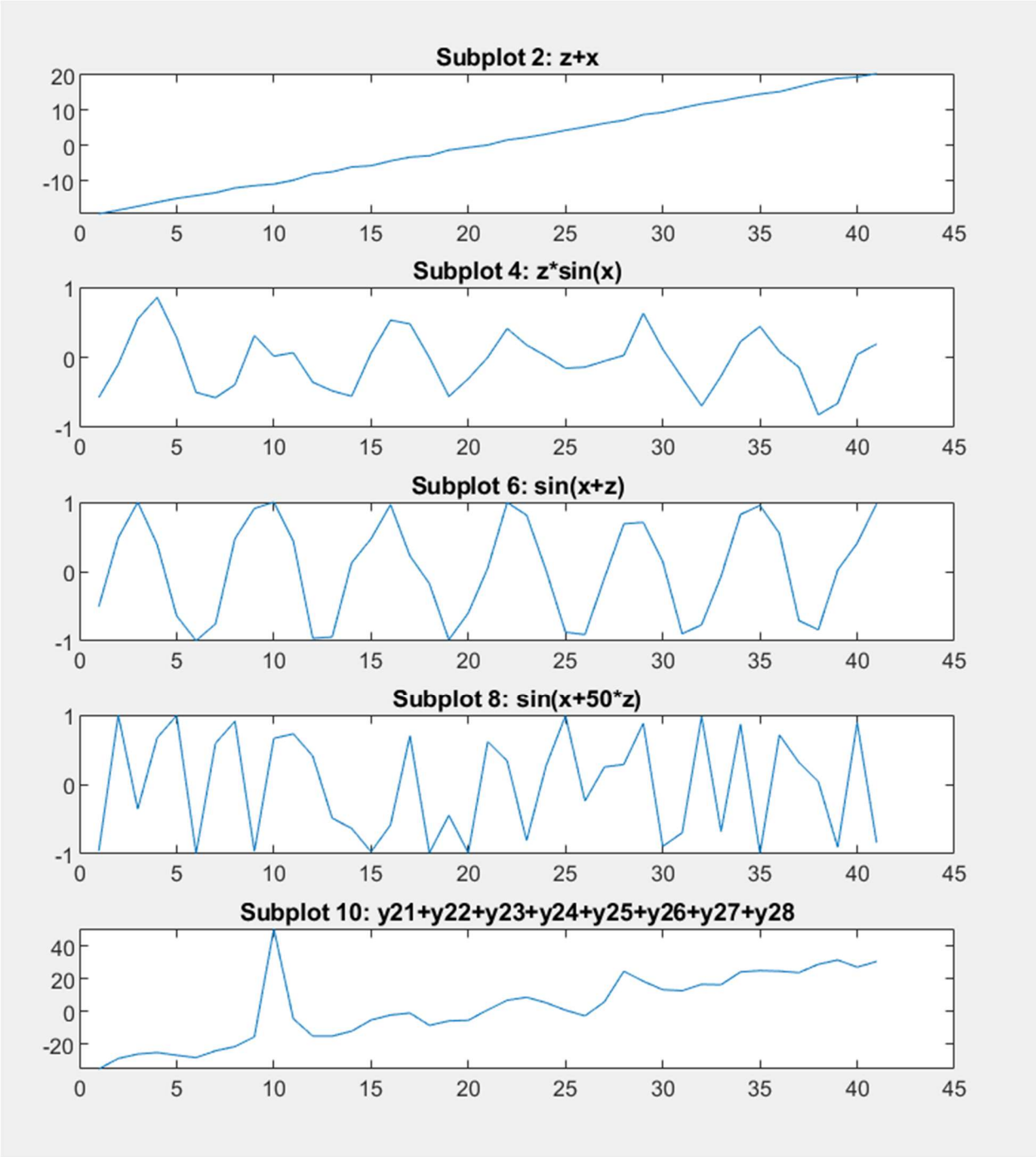
What did I learn?

I learned that how adding a uniformly distributed random variable to sin and multiplying sin with a uniformly distributed random variable affect the behavior of sin function.

Figures (Q4):



Figures (Q4):

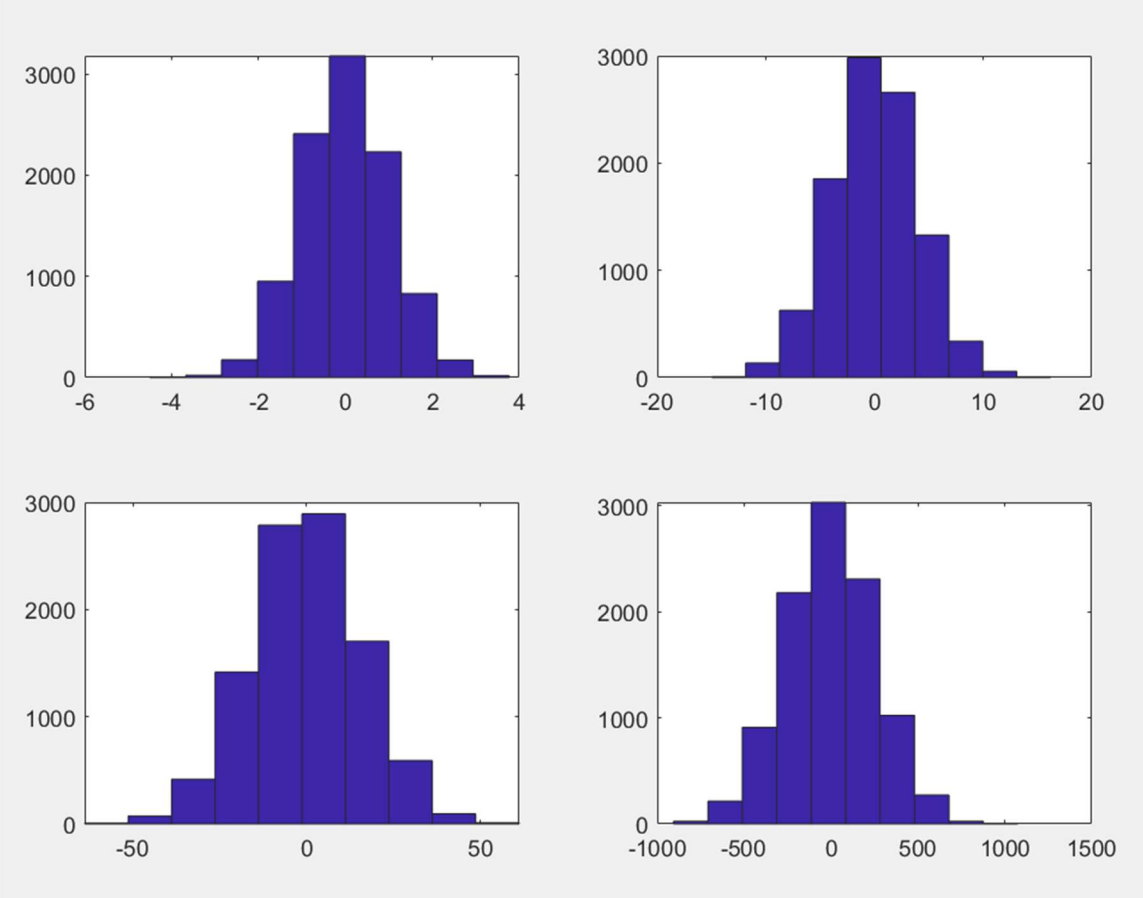

Subplot 2: z+x
Subplot 4: z*sin(x)
Subplot 6: sin(x+z)
Subplot 8: sin(x+50*z)
Subplot 10: y21+y22+y23+y24+y25+y26+y27+y28

Questions 5:

Code Snippet:

```
% ------------------------ Question 5 ------------------------

r1 = 0 + 1.*randn(10000,1);
r2 = 0 + 4.*randn(10000,1);
r3 = 0 + 16.*randn(10000,1);
r4 = 0 + 256.*randn(10000,1);

 figure
 subplot(2,2,1)
hist(r1)
subplot(2,2,2)
hist(r2)
subplot(2,2,3)
hist(r3)
subplot(2,2,4)
hist(r4)
```

Figures (Q5):



What did I learn?

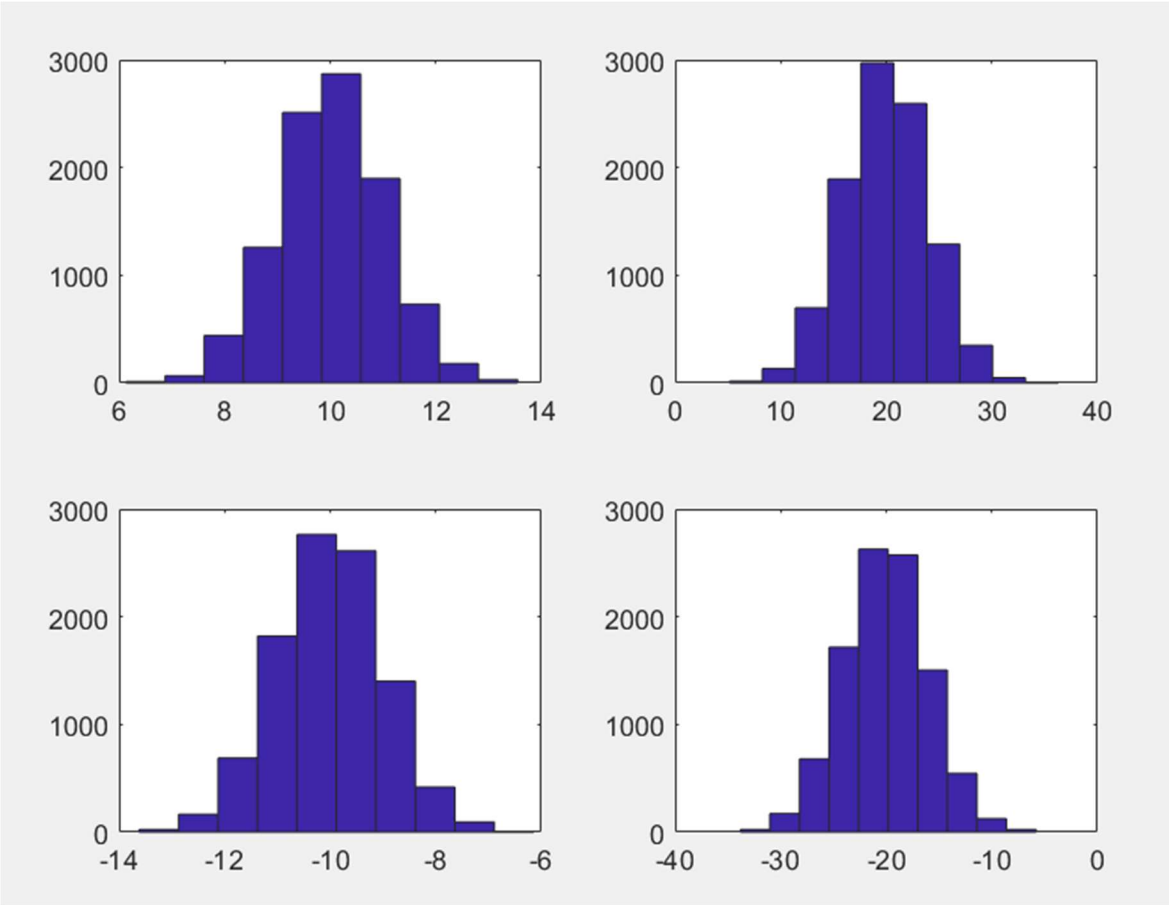
I learned how to transform Gaussian distribution to any normal distribution.

Questions 6:

Code Snippet:

```
% ------------------------ Question 6 ------------------------

r5 = 10 + 1.*randn(10000,1);
r6 = 20 + 4.*randn(10000,1);
r7 = -10 + 1.*randn(10000,1);
r8 = -20 + 4.*randn(10000,1);

 figure
 subplot(2,2,1)
hist(r5)
subplot(2,2,2)
hist(r6)
subplot(2,2,3)
hist(r7)
subplot(2,2,4)
hist(r8)
```

Figures (Q6):



What did I learn?

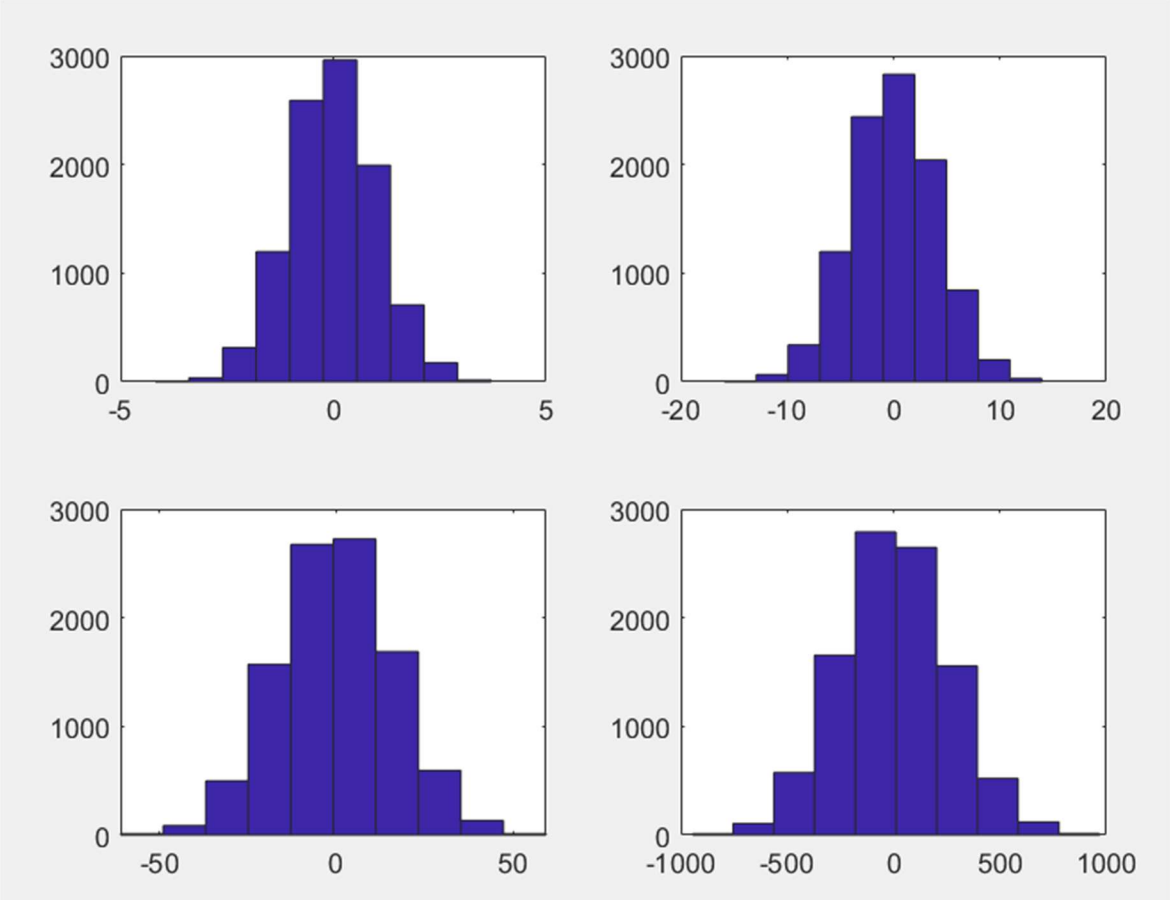
I learned how to transform Gaussian distribution to any normal distribution.

Questions 7:

Code Snippet:

```
% ------------------------ Question 7 ------------------------

r11 = 0 + sqrt(2)*1*erfinv(2*rand(10000,1)-1);
r21 = 0 + sqrt(2)*4*erfinv(2*rand(10000,1)-1);
r31 = 0 + sqrt(2)*16*erfinv(2*rand(10000,1)-1);
r41 = 0 + sqrt(2)*256*erfinv(2*rand(10000,1)-1);

 figure
 subplot(2,2,1)
hist(r11)
subplot(2,2,2)
hist(r21)
subplot(2,2,3)
hist(r31)
subplot(2,2,4)
hist(r41)
```

Figures (Q7):



What did I learn?

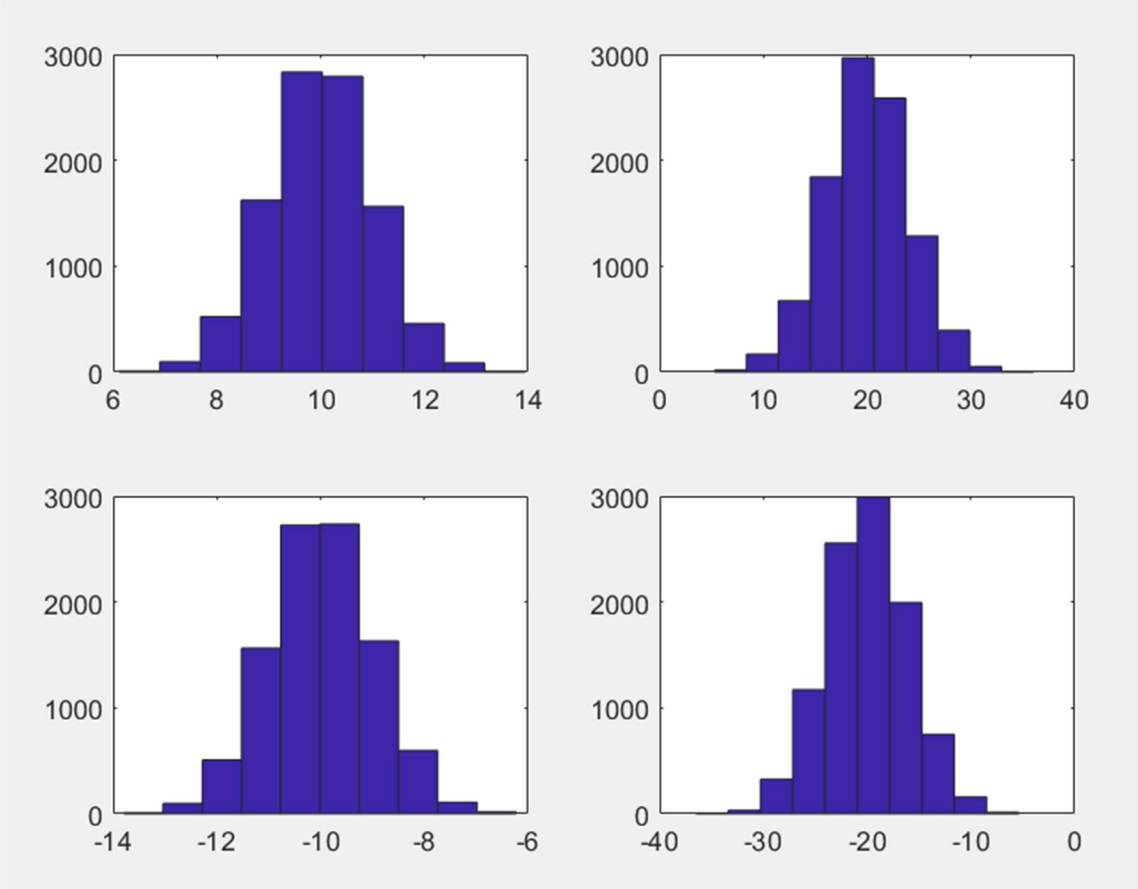
I learned how to transform uniform distribution to any normal distribution.

Questions 8:

Code Snippet:

```
% ------------------------ Question 8 ------------------------

r61 = 10 + sqrt(2)*1*erfinv(2*rand(10000,1)-1);
r71 = 20 + sqrt(2)*4*erfinv(2*rand(10000,1)-1);
r81 = -10 + sqrt(2)*1*erfinv(2*rand(10000,1)-1);
r91 = -20 + sqrt(2)*4*erfinv(2*rand(10000,1)-1);

 figure
 subplot(2,2,1)
 hist(r61)
 subplot(2,2,2)
 hist(r71)
 subplot(2,2,3)
 hist(r81)
 subplot(2,2,4)
 hist(r91)
```

Figures (Q8):



What did I learn?

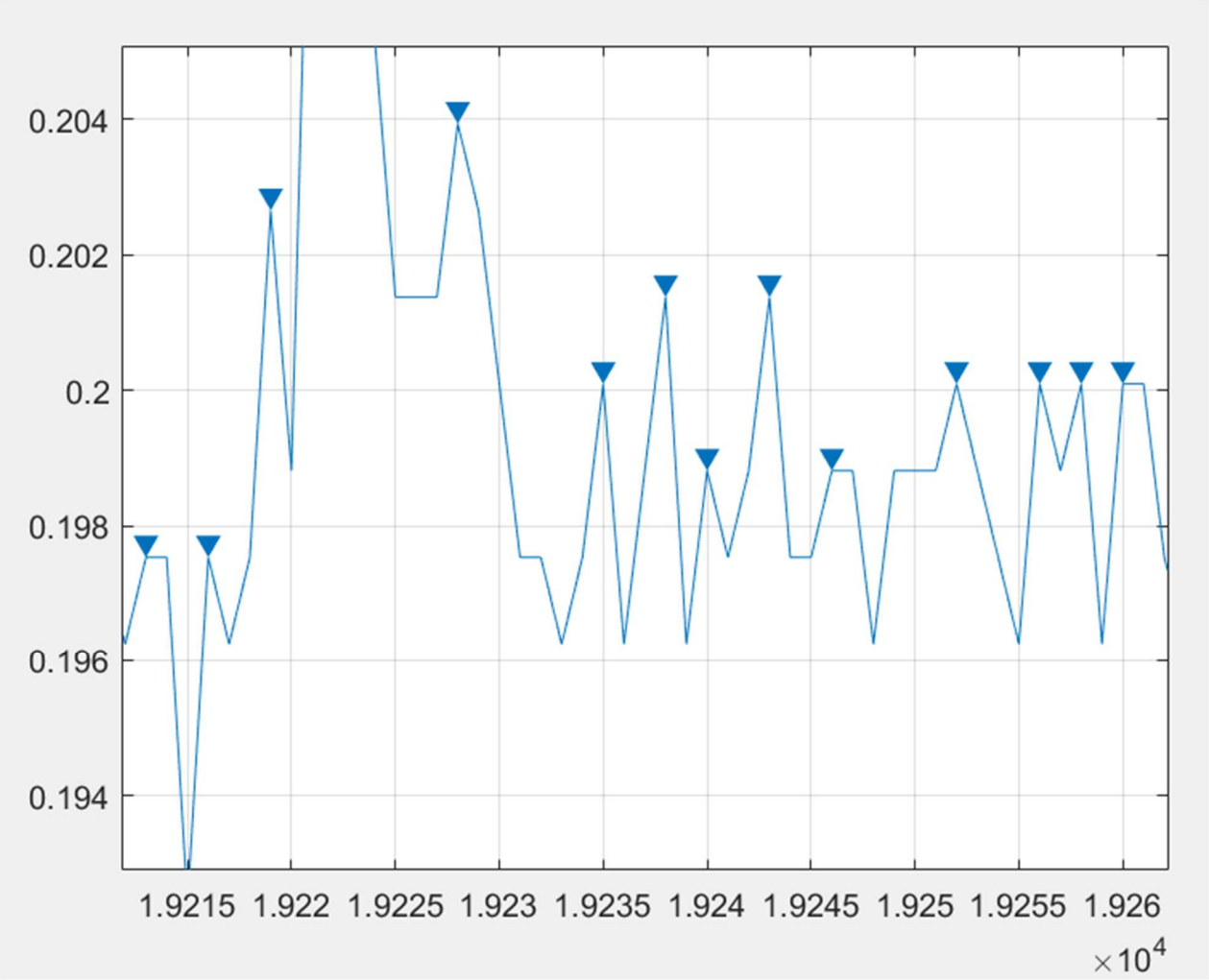
I learned how to transform uniform distribution to any normal distribution.

Questions 9:

Code Snippet:

```
% ------------------------ Question 9 ------------------------

M = csvread('exampleSignal.csv');

findpeaks(M)
```

Figures (Q9):



***Just a sample**

What did I learn?

I learned how to read a CSV file and find peaks in it. In addition to that I learned why findpeaks function does not show all the peaks. The reason is that the function just does not prefer to show. For example, if figure includes a flat part, findpeaks only show the first point of the flat part. This function accepts a point as a peak if it is larger or lower than its two neighbor point.

Questions 10:

Code Snippet:

```
% ------------------------ Question 10 ------------------------

lena = imread('lena.png');

lenaGray = rgb2gray(lena);

Mean = mean2(lenaGray)
SD = std2(lenaGray)

Max = max(lenaGray, [], 'all');
[xmax,ymax] = find(lenaGray==Max)

Min = min(lenaGray, [], 'all');
[xmin,ymin] = find(lenaGray==Min)
```

Output (Q10):

```
>> Mean = 124.0425

>> SD = 47.8556

>> xmax = 274  ymax = 396

>> xmin = 72 ymin = 4
```

What did I learn?

I learned how to calculate the mean, standard deviation, maximum and minimum of a matrix. In addition to that I learned how to read an image and transform it into a grayscale image.

Feedback:

This is my first-time using MATLAB. It has so many predefined functions and this make calculations easier. I was able to read an image into a matrix, then calculate its standard deviation with only two function calls. I think this is the biggest advantage of MATLAB over other languages, it makes the calculations a lot easier. However, as a disadvantage, since MATLAB includes so many functions and features it will take a while to learn and get used to it