I have generated sin wave signal with time=100 and let the Amplitude as default which is equal to 1. After that, I added noise to it ,then try to remove the noise using Moving Average filter. np.random.uniform used here to generate constant or regular shape of noise.

```python
import numpy as np
import matplotlib.pyplot as plt
#=================================================
Time=np.arange(0,100,0.01)
Amplitude=np.sin(Time)
Noise = np.random.uniform(0.0, 1.0,10000)+Amplitude
#======================Moving Average Smooth======
#
def MA (window):
    counter=0
    Smooth=[]
#    window= input('Enter window = ')
#    window = int(window)
    orginal_window=window
    size=(np.size(Noise))-(window-1)

    while (counter< size):
       s=0.0
       pointer=counter
       while (pointer <window):
            value=Noise[pointer]
            s+=value
            pointer+=1
       s/=orginal_window
       Smooth.append(s)
       window+=1
       counter+=1

    Smooth=np.array(Smooth)
#--------------------Change the start value to 1------------
    c=0
    while(c<np.size(Smooth)):
          Smooth[c]=Smooth[c]/Smooth[0]
          c=c+1
    b=0
    while(b<np.size(Noise)):
          Noise[b]=Noise[b]/Noise[0]
          b=b+1

    Smooth.resize(10000)
    return Smooth
```

```python
#==========================Original Sin wave==========================
plt.figure(1)
plt.plot(Time,Amplitude,'sienna',label='Original Signal',linewidth='3')
plt.xlabel('Time')
plt.ylabel('Amplitude')
plt.title('Sin wave')
plt.grid(axis='both')
plt.xlim(0,100)
plt.legend(loc='upper right')
#####=============================Noise=================================
plt.figure(2)
plt.plot(Time,Noise,'sienna',label='Signal with noise',linewidth='3')
plt.xlabel('Time')
plt.ylabel('Amplitude ')
plt.title('Signal with noise')
plt.grid(axis='both')
plt.xlim(0,100)
plt.legend(loc='upper right')
##==================================MA waves=======================================
plt.figure(3)
plt.plot(Time,MA(1),'indianred',label='Window=1')
plt.plot(Time,MA(10),'darkslateblue',label='Window=15',linewidth='3')
plt.plot(Time,MA(70),'limegreen',label='Window=70',linewidth='3')
plt.plot(Time,MA(150),'orange',label='Window=150',linewidth='3')
plt.plot(Time,MA(370),'cornflowerblue',label='Window=370',linewidth='3')
plt.xlabel('Time')
plt.ylabel('Amplitude ')
plt.title('Smoothed signal with MA filter')
plt.grid(axis='both')
plt.xlim(0,100)
plt.legend(loc='upper right')
#
##==========================Best MA waves=======================================
plt.figure(4)
plt.plot(Time,MA(370),'cornflowerblue',label='Smoothed with best window',linewidth='3')
plt.xlabel('Time')
plt.ylabel('Amplitude ')
plt.title('Best smoothed')
plt.grid(axis='both')
plt.xlim(0,100)
plt.legend(loc='upper right')
```

Figure 1 Code for signal generation, visualization and smoothing

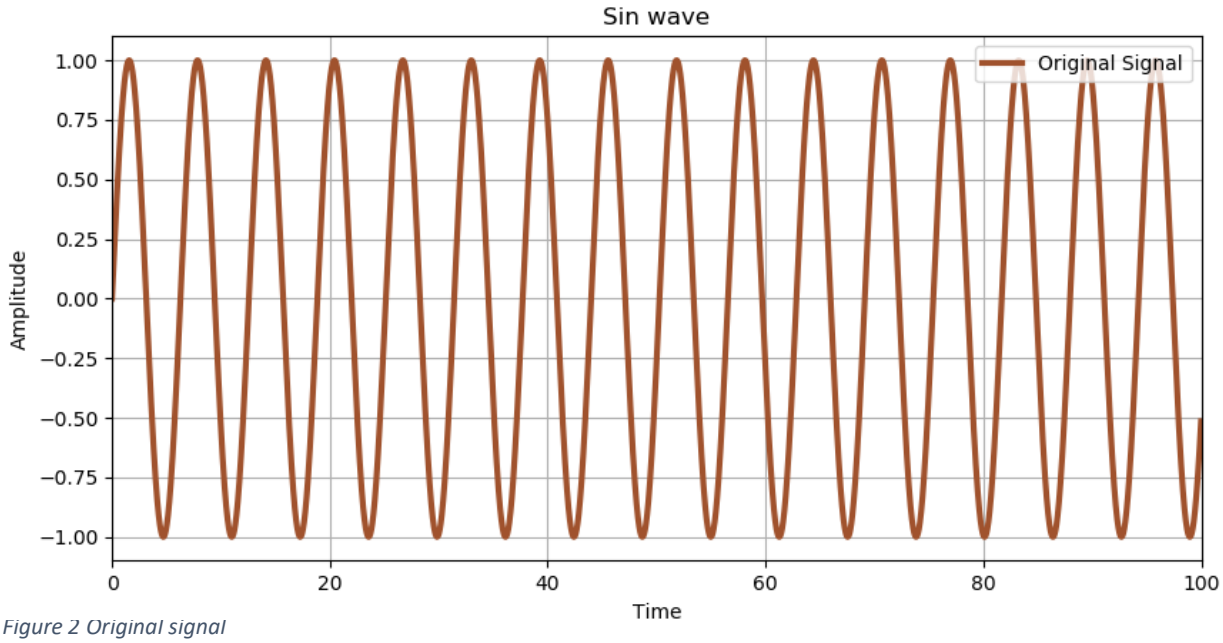

Figure 2 Original signal

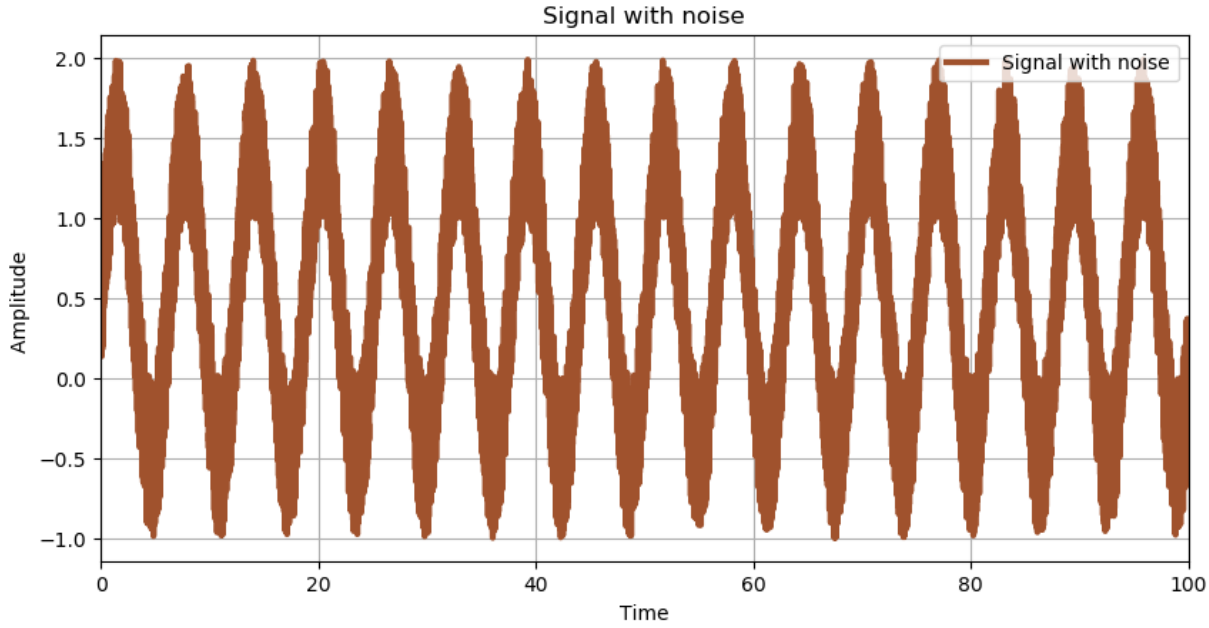

Figure 3 Signal with noise

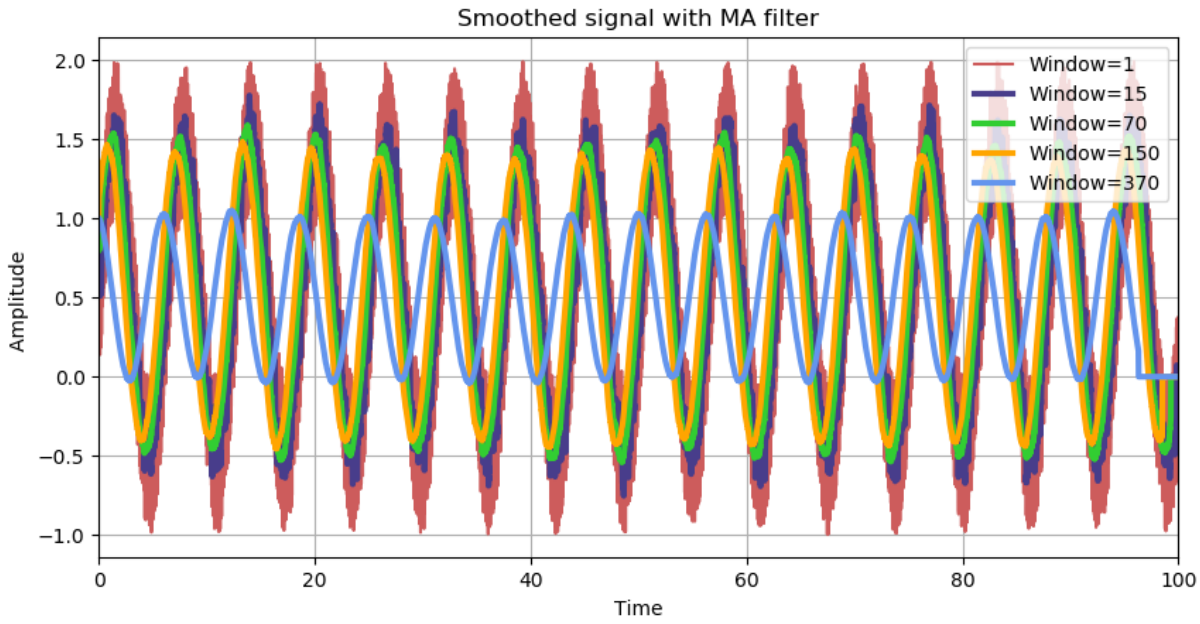

Figure 4 Moving Average filter with 5 windows

Attempts to remove or smooth the noise to become as much as the original signal using Moving Average filter algorithm. I have tried 5 windows to pick the best one. By looking to the figure we can see that when the window size is bigger the noise become less and it's become closer to the original shape. The best window size with less noise here is 370. When it's plotted alone it become more clear that it's look like the original one.

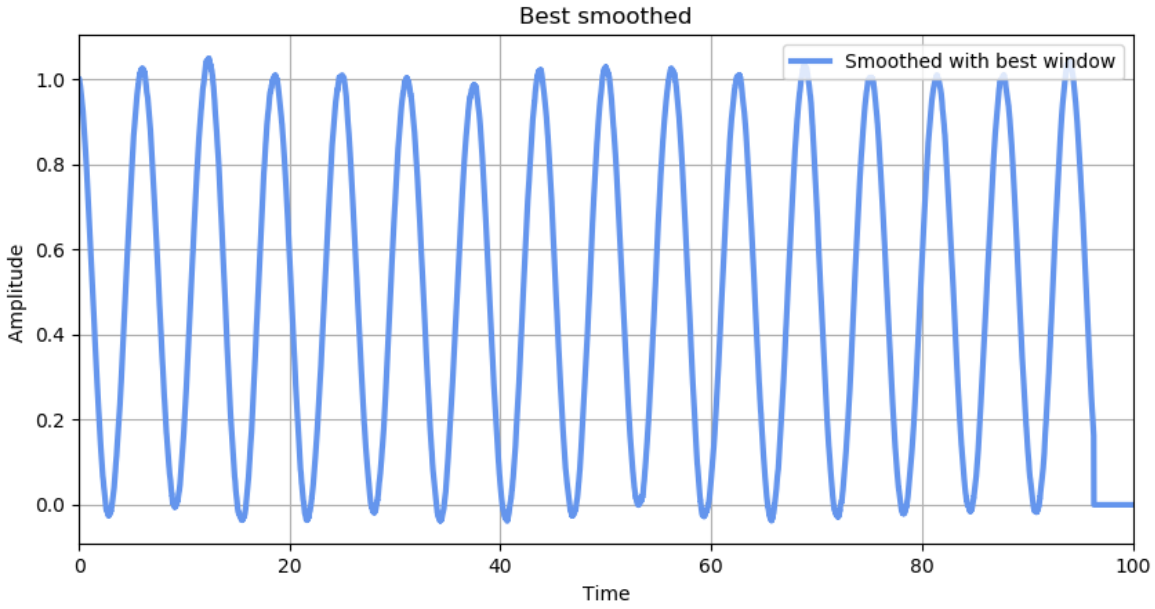

Figure 5 Best window size