

```
# Hello World project
# python3
# پیاده سازی میانگین متحرک ساده در پایتون
# کتابخانه های مورد نیاز - numpy and matplotlib
```

```
import numpy as np
import matplotlib.pyplot as plt
```

```
# تولید داده مصنوعی برای پیاده سازی میانگین متحرک در پایتون
# اکنون با ایجاد و استفاده از یک تابع، ۵۰ داده مصنوعی تولید می شود:
```

```
def Seri(N):
    X = np.arange(0, N, 1)
    S = 0.9*np.sin(X/8) - 0.9*np.sin(2+X/5) +
    0.4*np.sin(4+X/2) - 2*X/N
    return S
```

```
N = 50 # Data Size
S = Seri(N) # Time Seri Data
```

```
# نمایش و رسم داده های مصنوعی تولید شده
# حالا باید کد نویسی لازم برای نمایش و رسم داده ها را انجام داد
```

```
plt.style.use('ggplot')
plt.plot(np.arange(1, N+1), S, label = 'Seri Data',
         c = 'b', linewidth = 0.8, marker = 'o',
         ms = 4, mfc = 'r', mec = 'r')
plt.xlabel('T')
```

```
plt.ylabel('Y')
```

```
plt.legend()
```

```
plt.show()
```

```
# محاسبه میانگین متحرک در پایتون
```

```
# حال باید طول پنجره را تعریف و مقدار میانگین متحرک را محاسبه کرد:
```

```
L = 3
```

```
n = np.size(S) - L + 1
```

```
M = np.zeros(n)
```

```
for i in range(0, n):
```

```
    M[i] = np.mean(S[i:i + L])
```

```
# تعریف تابع میانگین متحرک در پایتون
```

```
# حال می‌توان برای محاسبه میانگین متحرک، تابعی را به صورت زیر تعریف کرد .
```

```
def SMA(S, L):
```

```
    n = np.size(S) - L + 1
```

```
    M = np.zeros(n)
```

```
    for i in range(0, n):
```

```
        M[i] = np.mean(S[i:i + L])
```

```
    return M
```

```
# استفاده از تابع تعریف شده برای محاسبه میانگین متحرک در پایتون
```

```
# حال می‌توان از تابع فوق برای محاسبه میانگین متحرک داده‌ها استفاده کرد :
```

```
L = 3 # Window Size
M = SMA(S, L) # Simple Moving Average Values
```

تحلیل شیوه رفتار میانگین متحرک
در یک M و S حالا به منظور بررسی شیوه رفتار میانگین متحرک ، مقادیر توابع
: نمودار به همراه هم استفاده شده‌اند

```
plt.style.use('ggplot')
plt.plot(np.arange(1, N+1), S, label = 'Seri Data',
         c = 'b', linewidth = 0.8, marker = 'o', ms
         = 4, mfc = 'r', mec = 'r')
plt.plot(np.arange(L, N+1), M, label = 'Simple
Moving Average (L={})'.format(L),
         c = 'g', linewidth = 1.3)
plt.xlabel('T')
plt.ylabel('Y')
plt.legend()
plt.show()
```

تحلیل اثر طول پنجره در نمودار
می‌توان برای بررسی اثر طول پنجره در نمودار، میانگین متحرک‌هایی با طول
: را محاسبه و رسم کرد 3، 8، 21

```
M3 = SMA(S, 3) # SMA(3)
M8 = SMA(S, 8) # SMA(8)
M21 = SMA(S, 21) # SMA(21)
```

پس از محاسبه، برای رسم نمودار باید از کدهای زیر استفاده شود :

```

plt.style.use('ggplot')
plt.plot(np.arange(1, N+1), S, label = 'Seri Data',
         c = 'k', linewidth = 0.8, marker = 'o', ms
         = 4,
         mfc = 'crimson',mec = 'crimson')
plt.plot(np.arange(3, N+1), M3, label = 'SMA
(L=3)',
         linewidth = 1.3, c = 'teal')
plt.plot(np.arange(8, N+1), M8, label = 'SMA
(L=8)',
         linewidth = 1.3, c = 'salmon')
plt.plot(np.arange(21, N+1), M21, label = 'SMA
(L=21)',
         linewidth = 1.3, c = 'navy')
plt.xlabel('T')
plt.ylabel('Y')
plt.legend()
plt.show()

```

حذف روند از سری زمانی
کد مربوطه به صورت زیر است:

```
D = S[2:] - M3 # Detrended Data With SMA(3)
```

حالا برای رسم نمودار باید از کدهای زیر استفاده کرد :

```

plt.style.use('ggplot')
plt.subplot(2,1,1)
plt.plot(np.arange(1, N+1), S, label = 'Seri Data',
         c = 'k', linewidth = 0.8, marker = 'o', ms

```

```

= 4,
    mfc = 'crimson',mec = 'crimson')
plt.plot(np.arange(3, N+1), M3, label = 'SMA
(L=3)',
    linewidth = 1.3, c = 'teal')
plt.xlim(0,N+1)
plt.xlabel('T')
plt.ylabel('Y')
plt.legend()

plt.subplot(2,1,2)
plt.plot(np.arange(3, N+1), D, label = 'Deternded
Data With SMA (L=3)',
    linewidth = 1.3, c = 'teal')
plt.plot([3, N], [0, 0], label = 'Zero Line',
    linewidth = 1.3, c = 'k')
plt.xlim(0,N+1)
plt.xlabel('T')
plt.ylabel('Y')
plt.legend()
plt.show()

```

اولین آرگومان subplot باید توجه داشت که در زمان استفاده از
تعداد سطرها، دومین آرگومان تعداد ستون‌ها و آخرین آرگومان نیز
subplot شماره نمودار مورد نظر را تعیین می‌کند. همچنین، برای هر
xlabel, ylabel, legend باید به صورت جداگانه خصیصه‌های
را تعریف کرد xlim و