Ex4_Ornstein_Uhlenbeck

November 20, 2019

```python
[5]: import matplotlib.pyplot as plt
     import numpy as np
     from pandas.plotting import autocorrelation_plot
     plt.rcParams['figure.figsize'] = [10, 5]
```

```python
[6]: #Probably the implemantation is WRONG!
     #Initialize parameters and generate 1000 random points from N(0,1) g works from
      ↪0 up to 1.2
     s=2
     g=0.8
     positions = [0]
     random_p = np.random.normal(0,1,1000)
```

```python
[7]: #positions
     for r in random_p:
         mv = -1 if r<0 else 1
         value = positions[-1]*(1-g) + s*mv
         positions.append(value)
```

```python
[8]: plt.plot(positions)
```
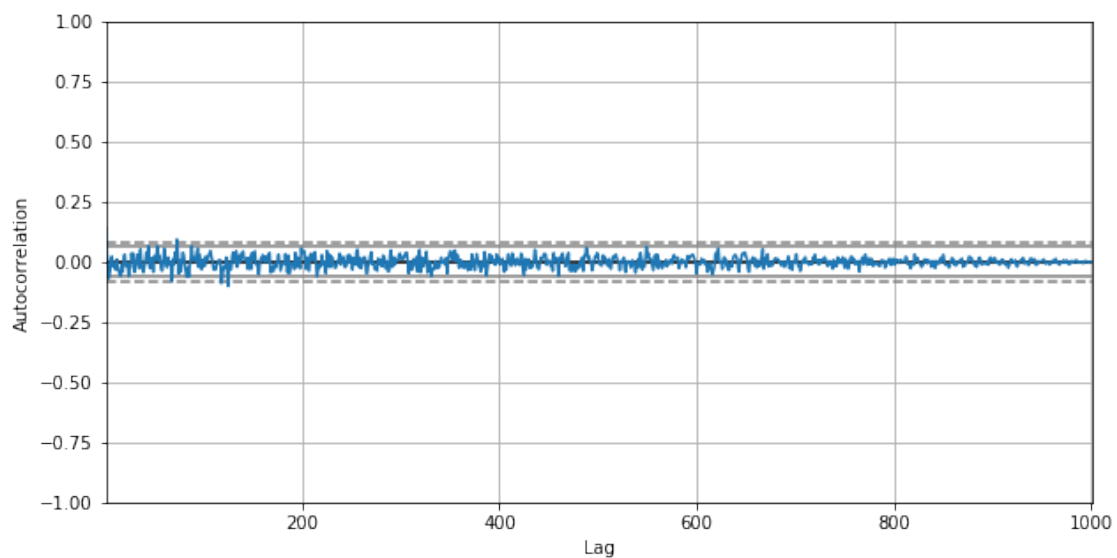
```
[8]: [<matplotlib.lines.Line2D at 0x7efdb91a0208>]
```
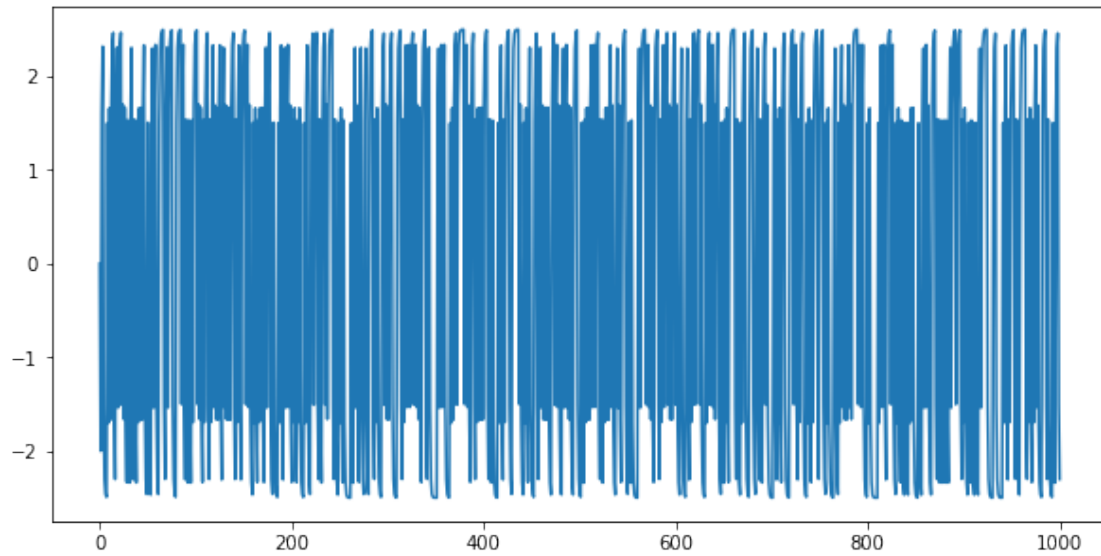
```
[9]: autocorrelation_plot(positions)
```

```
[9]: <matplotlib.axes._subplots.AxesSubplot at 0x7efdb9155550>
```



```
[10]: #calc differences
      diff1 = []
      #calc differences with absolute values
      diff2 = []
```
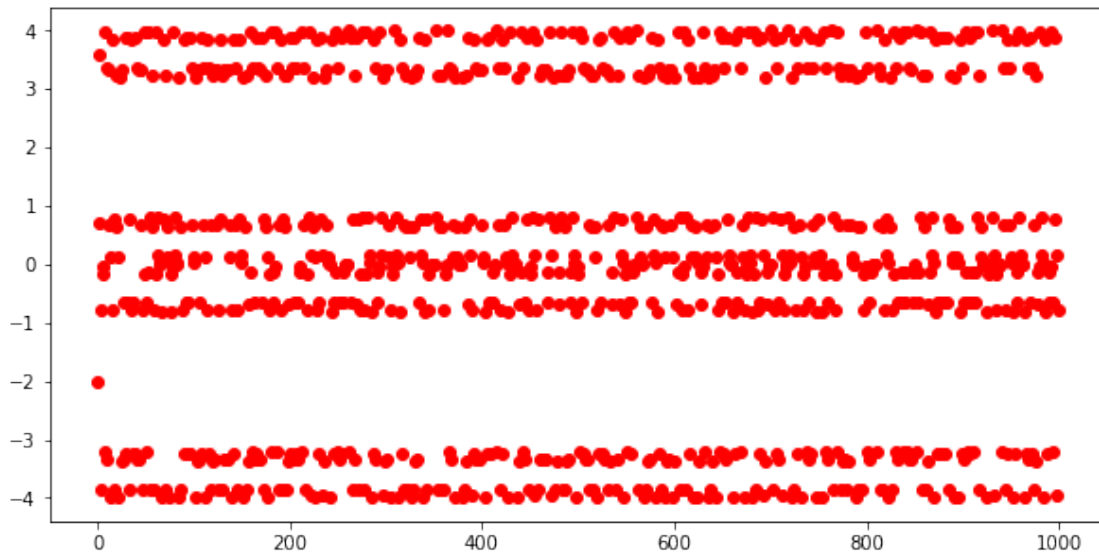
```
for i in range(1,len(positions)):
    value = positions[i] - positions[i-1]
    diff1.append(value)
for i in range(1,len(positions)):
    value = np.abs(positions[i] - positions[i-1])
    diff2.append(value)
```

[11]: ```
plt.plot(diff1,'ro')
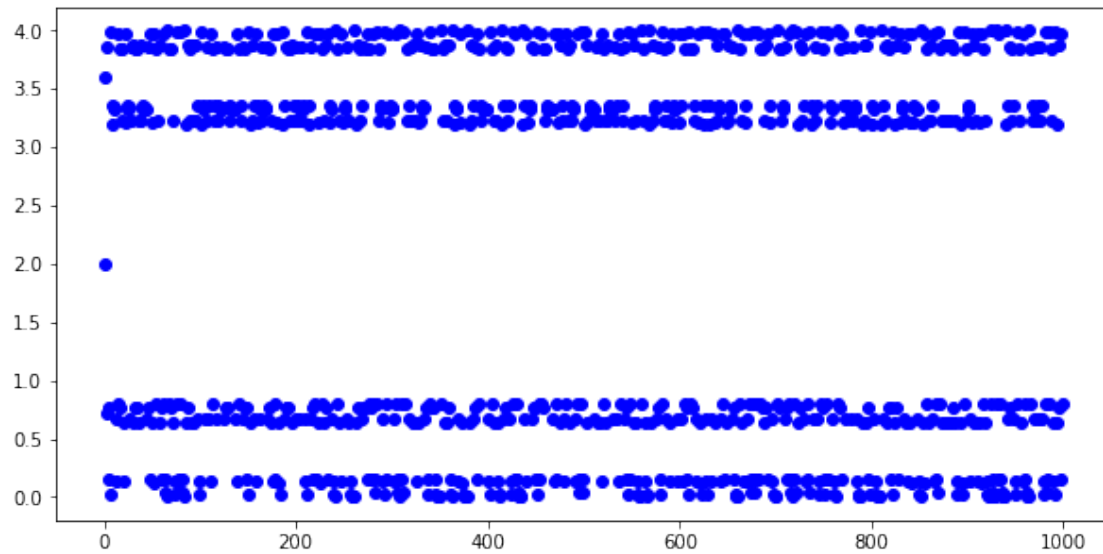```

[11]: [<matplotlib.lines.Line2D at 0x7efdb9094160>]

[12]: ```
plt.plot(diff2,'bo')
#calculate s
#s = E[diff2]
np.mean(diff2)
```

[12]: 2.096763755311913

```
[13]:  #g = sqrt(var(diff)/var(positions))
       np.sqrt(np.var(diff2)/np.var(positions))
```

```
[13]:  0.7984944976780886
```

```
[14]:  autocorrelation_plot(diff2)
```

```
[14]:  <matplotlib.axes._subplots.AxesSubplot at 0x7efdb8fbb240>
```


4