# NeuroTools.stgen

```python
import pylab
import NeuroTools.signals as signals
```

```python
import NeuroTools.stgen as stgen
```

```python
sg = stgen.StGen()
duration = 10000.
rate_independent = 100. #Hz
rate_shared = 20. #Hz, 20 % correlation
st1 = sg.poisson_generator(rate=rate_independent, t_stop = duration)
print "Spiketrain 1:"
```

```
Spiketrain 1:
```

```python
print "mean rate: %f" % st1.mean_rate()
```

```
mean rate: 98.300000
```

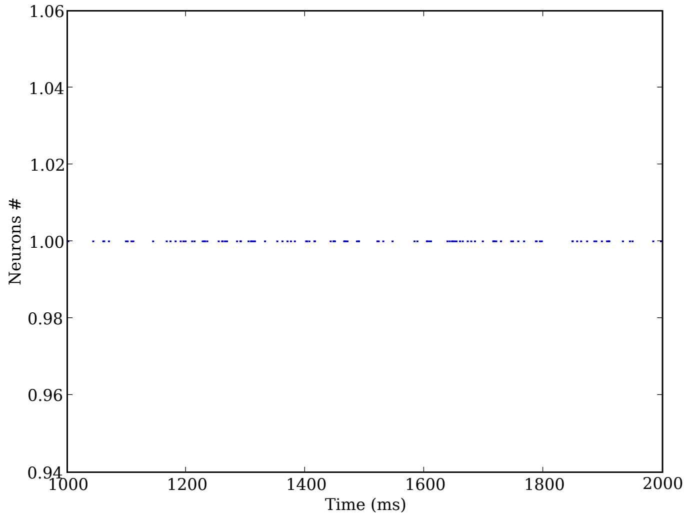
```python
print "coefficient of variation: %f" % st1.cv_isi()
```

```
coefficient of variation: 1.049050
```

```python
print "fano factor: %f" % st1.fano_factor_isi()
```

```
fano factor: 11.202128
```

```python
st1.raster_plot(display=True)
pylab.xlim(1000.,2000.)
pylab.show()
```



```

st2 = sg.poisson_generator(rate=rate_independent, t_stop = duration)
print ”Spiketrain 2:”
```

```
Spiketrain 2:
```

```
print ”mean rate: %f” % st2.mean_rate()
```

```
mean rate: 97.700000
```

```
print ”coefficient of variation: %f” % st2.cv_isi()
```

```
coefficient of variation: 1.072666
```

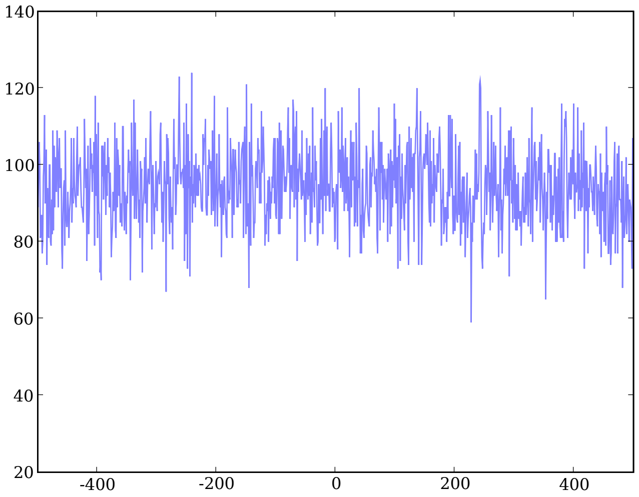
```
print ”fano factor: %f” % st2.fano_factor_isi()
```

```
fano factor: 11.787678
```

cross correlation

```
import numpy
pylab.figure()
```

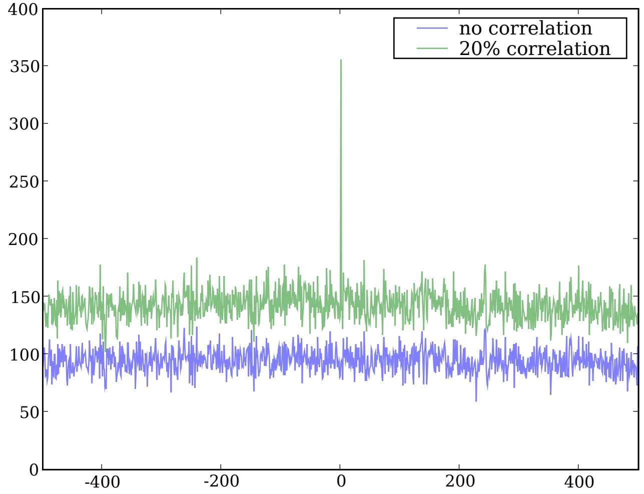
```python
cc = numpy.correlate(st1.time_histogram(time_bin = 1.),
                     st2.time_histogram(time_bin = 1.),
                     mode = 'same')
time_axis = numpy.linspace(cc.shape[0]/-2., cc.shape[0]/2, cc.shape[0])

pylab.plot(time_axis , cc, label = 'no correlation', alpha = 0.5)
pylab.xlim(-500.,500.)
pylab.show()
```



inject correlation into st1 and st2

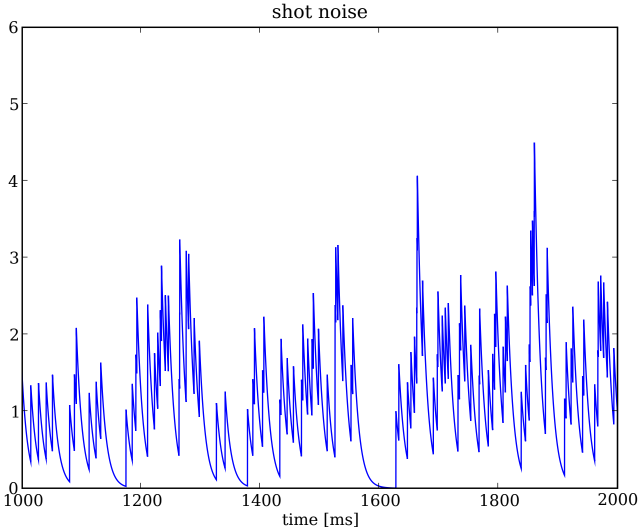
```python
st3 = sg.poisson_generator(rate = rate_shared , t_stop = duration)
st1.merge(st3)
st2.merge(st3)

cc2 = numpy.correlate(st1.time_histogram(time_bin = 1.),
                      st2.time_histogram(time_bin = 1.),
                      mode = 'same')

```

```
pylab.plot(time_axis, cc2, label = '20% correlation', alpha = 0.5)
pylab.legend()
pylab.xlim(-500.,500.)
pylab.show()
```



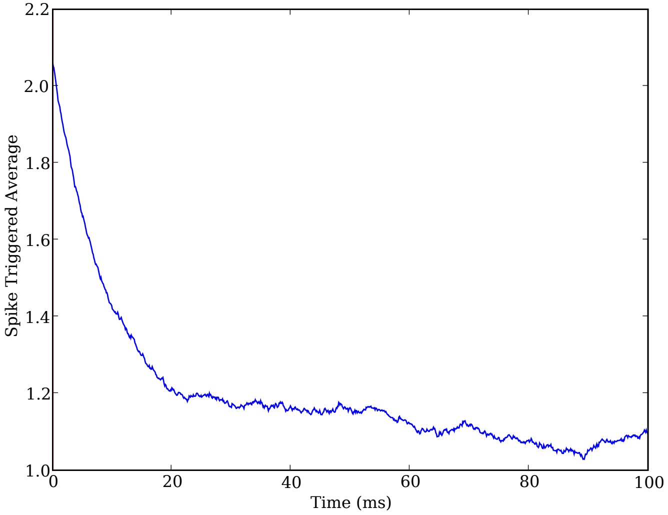
generate shot noise from st1

```
st1 = sg.poisson_generator(rate=rate_independent, t_stop = duration)
st1_shot = stgen.shotnoise_fromspikes(st1,q = 1.0,tau = 10.,t_start = st1.t_start,t_stop = st1.t_stop,array=True)
f = pylab.figure()
pylab.plot(st1_shot[1],st1_shot[0])
f.gca().set_title('shot noise')
f.gca().set_xlabel('time [ms]')
pylab.xlim(1000.,2000.)
pylab.show()
```



spike triggered average reveals the original shot-noise kernel

```
st1_shot = stgen.shotnoise_fromspikes(st1,q = 1.0,tau = 10.,t_start = st1.t_start,t_stop = st1.
    t_stop)
sta = st1_shot.event_triggered_average(st1, display = True, average = True)
pylab.show()
```


2.2
2.0
1.8
1.6
1.4
1.2
1.0
Spike Triggered Average
0
20
40
60
80
100
Time (ms)