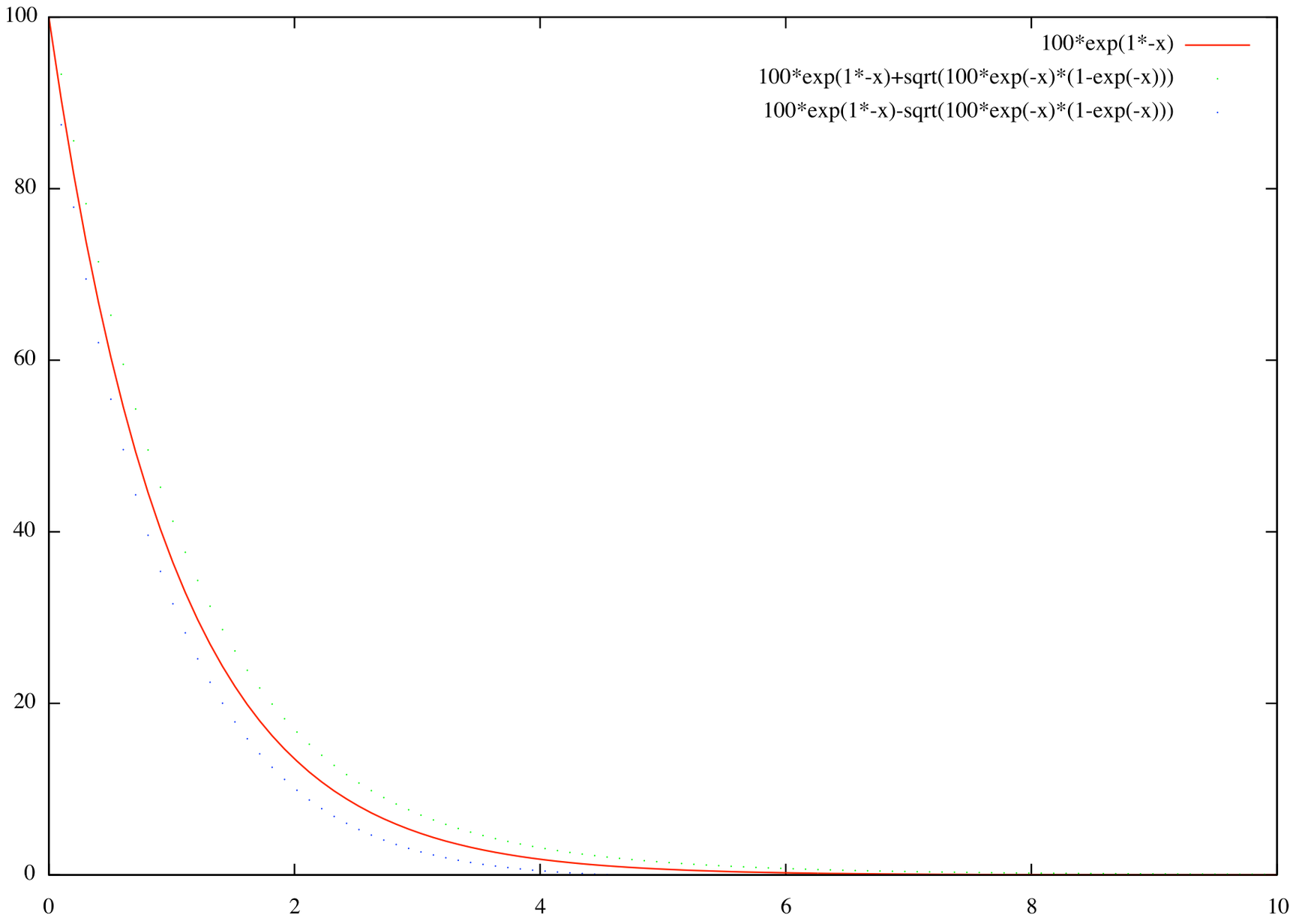
exponential decay




```
gnuplot> set xrange [0:10]
gnuplot> plot 100*exp(1*-x)
```

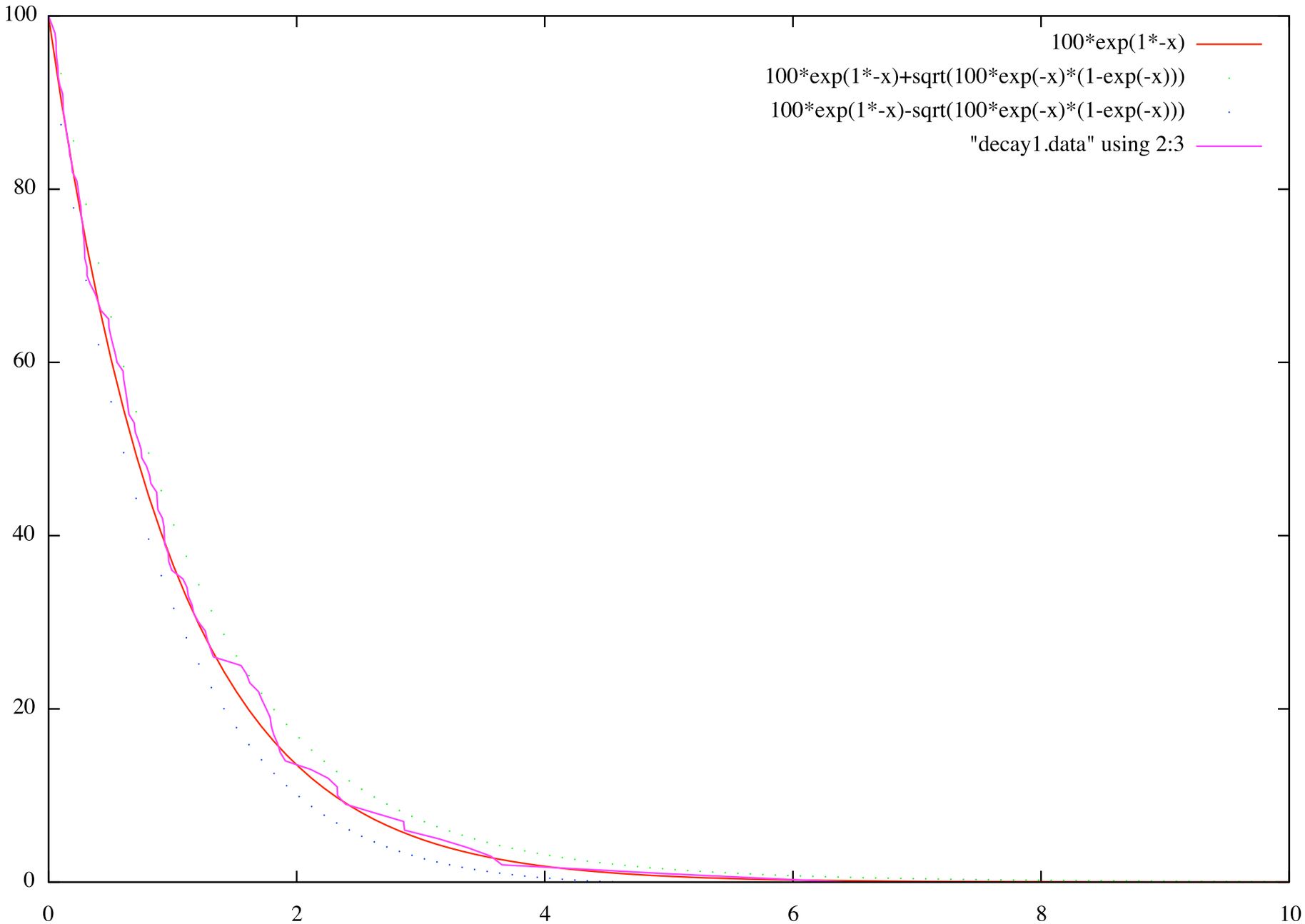
decay.py

```
import random

# initial state
NA = 100

# rate constants
k = 1.0

rescale = 1

# initialization
t = 0.0 # wall clock
na = NA * rescale # number of As
a = k * na # reaction activity
f = open('/Users/russ/Dropbox/Documents/decay'+str(rescale)+'.data','w')
f.write('# event time A\n' + '0 ' + str(t) + ' ' + str(na) + '\n')

# loop
for i in range(NA * rescale): # number of events to simulate
    t = t + random.expovariate(a) # time advance
    na = na - 1 # state change
    a = k * na # new activity
    f.write(str(i+1) + ' ' + str(t) + ' ' + str(na) + '\n')

# cleanup
f.close()
```

decay1.data

```
# event time A
0 0.0 100
1 0.0271041638862 99
2 0.053044345264 98
3 0.0612656855322 97
4 0.0620117104019 96
5 0.067354047165 95
6 0.0761948311196 94
7 0.078333802917 93
8 0.090634193252 92
9 0.114577990151 91
10 0.116857835680 90
...
```

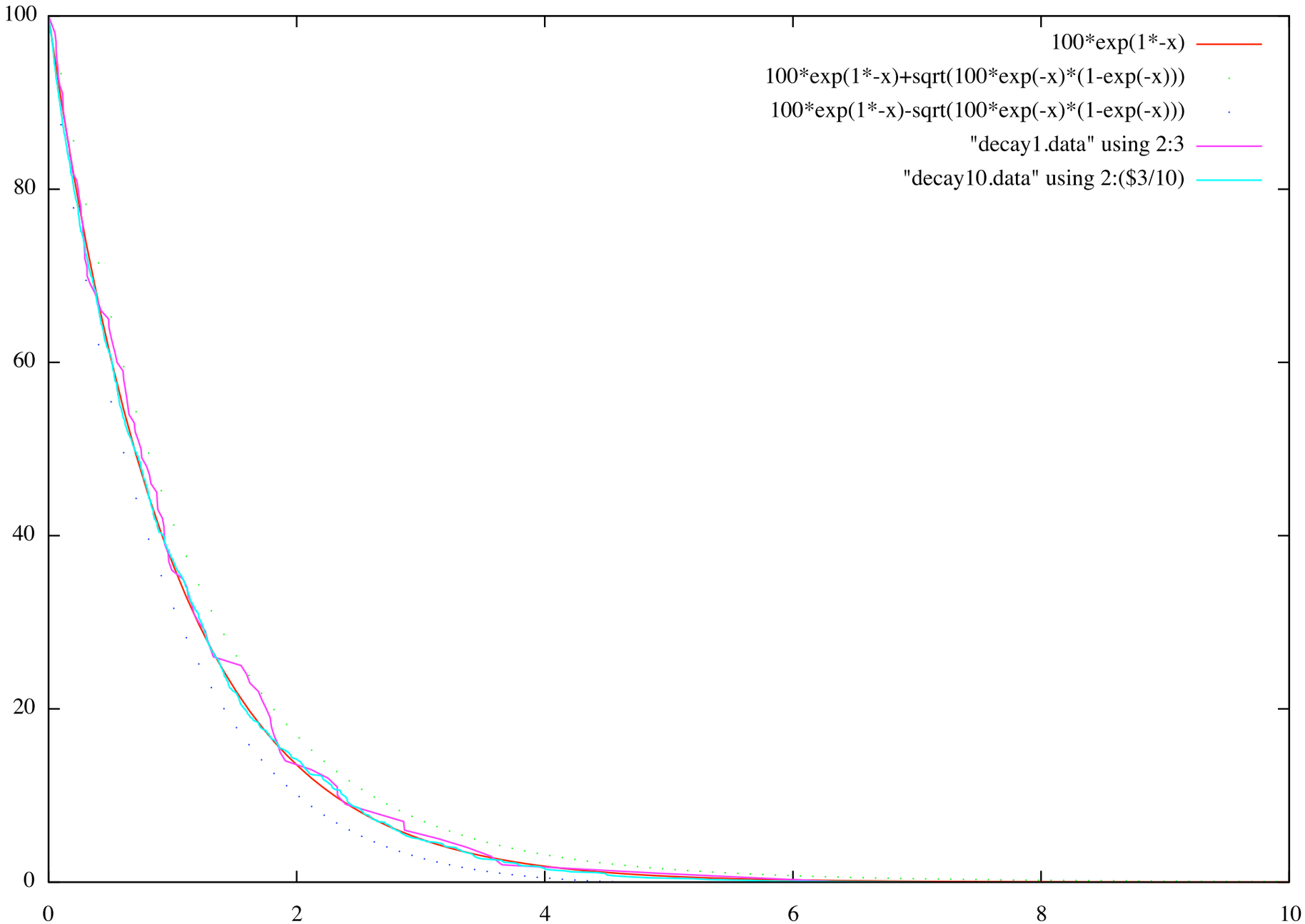
gnuplot> **plot “decay1.data” using 2:3 with lines**



gnuplot> **replot “decay1.data” using 2:3 with lines**

decay.py

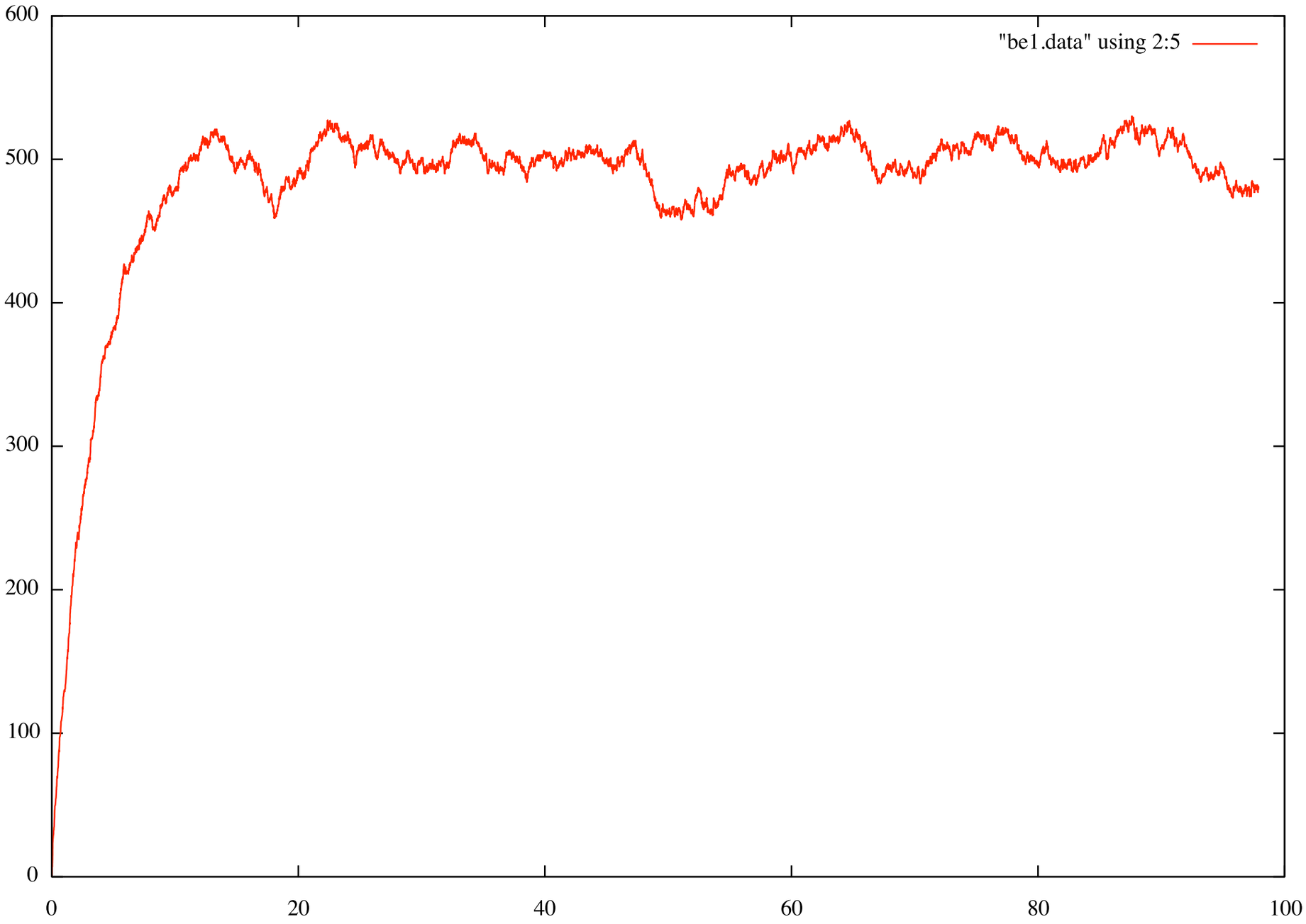
```
import random

# initial state
NA = 100

# rate constants
k = 1.0

rescale = 10

# initialization
t = 0.0 # wall clock
na = NA * rescale # number of As
a = k * na # reaction activity
f = open('/Users/russ/Dropbox/Documents/decay'+str(rescale)+'.data','w')
f.write('# event time A\n' + '0 ' + str(t) + ' ' + str(na) + '\n')

# loop
for i in range(NA * rescale): # number of events to simulate
    t = t + random.expovariate(a) # time advance
    na = na - 1 # state change
    a = k * na # new activity
    f.write(str(i+1) + ' ' + str(t) + ' ' + str(na) + '\n')

# cleanup
f.close()
```



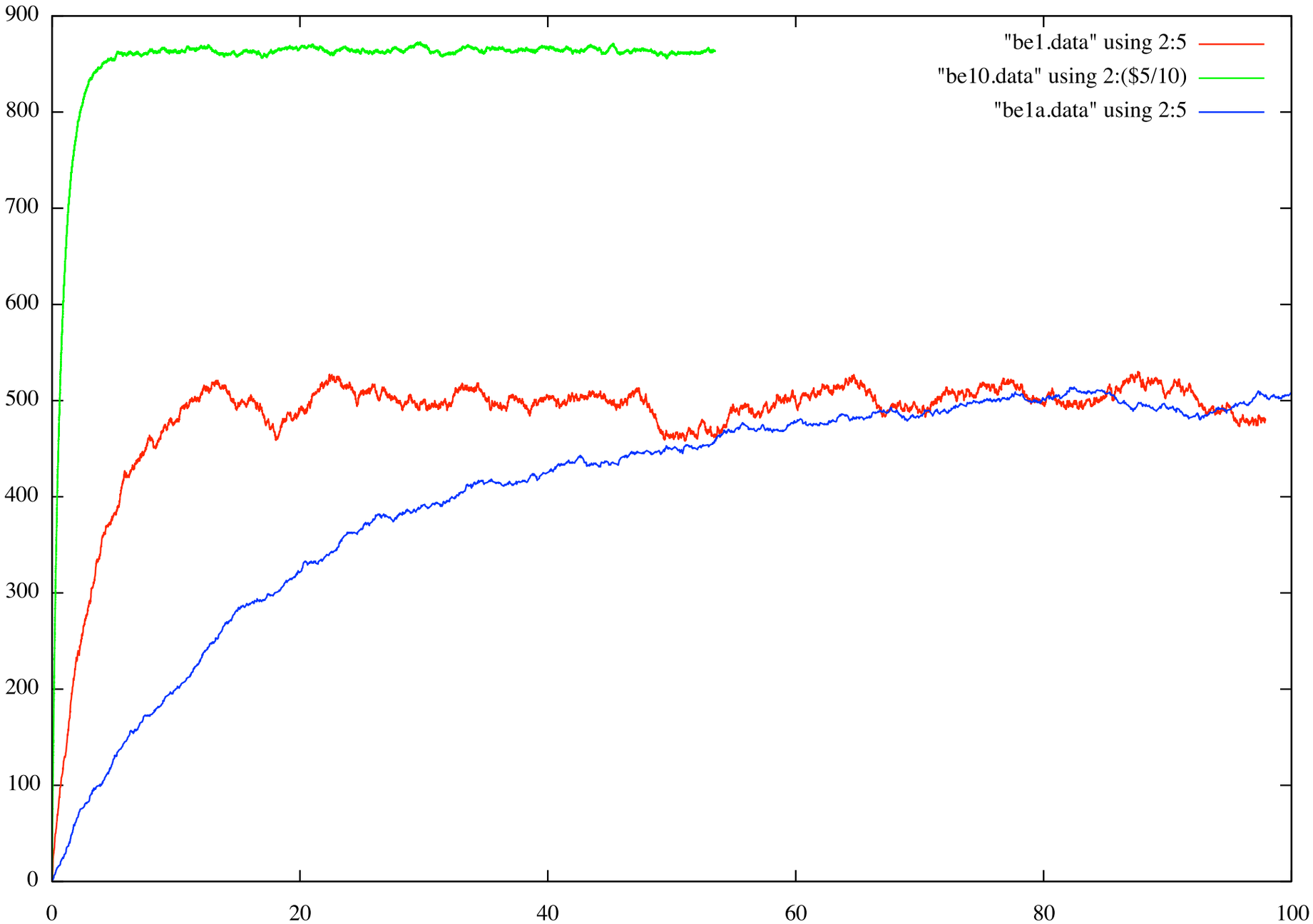
gnuplot> **replot “decay10.data” using 2:($3/10) with lines**

be.py

```
import random

# the variables
NA = 1000
NB = 1500
NAB = 0

# default rate constants
KF = 0.0001
KB = 0.1

E = 10000 * rescale # number of events; this is a rule of thumb only
rescale = 1

# actual rate constants
kf = KF / rescale
kb = KB

# initial state
(na, nb, nab) = (rescale*NA, rescale*NB, rescale*NAB)
(af, ab) = (kf * na * nb, kb * nab)
t = 0.0
f = open('/Users/russ/Dropbox/Documents/be'+str(rescale)+'.data','w')
f.write('0 ' + str(t) + ' ' + str(na) + ' ' + str(nb) + ' ' + str(nab) + '\n')

# loop
for i in range(E):
    t = t + random.expovariate(af+ab) # time advance
    if random.uniform(0,af+ab) <= af: # forward reaction
        (af, ab) = (af - kf*(na+nb-1), ab+kb)
        (na, nb, nab) = (na-1, nb-1, nab+1)
    else: # backward reaction
        (af, ab) = (af + kf*(na+nb+1), ab-kb)
        (na, nb, nab) = (na+1, nb+1, nab-1)
    f.write(str(i+1) + ' ' + str(t) + ' ' + str(na) + ' ' + str(nb) + ' ' + str(nab) + '\n')

# cleanup
f.close()
```



gnuplot> **plot “be1.data” using 2:5 with lines**


900
800
700
600
500
400
300
200
100
0
0
20
40
60
80
100
"be1.data" using 2:5
"be10.data" using 2:($5/10)
"be1a.data" using 2:5