Classic skyline method

Example

Suppose $t_k = \{100, 100, 20, 100\}$.
Then the classic skyline method gives

$$(N_e)_k = \frac{1}{2} C_k^2 t_k$$

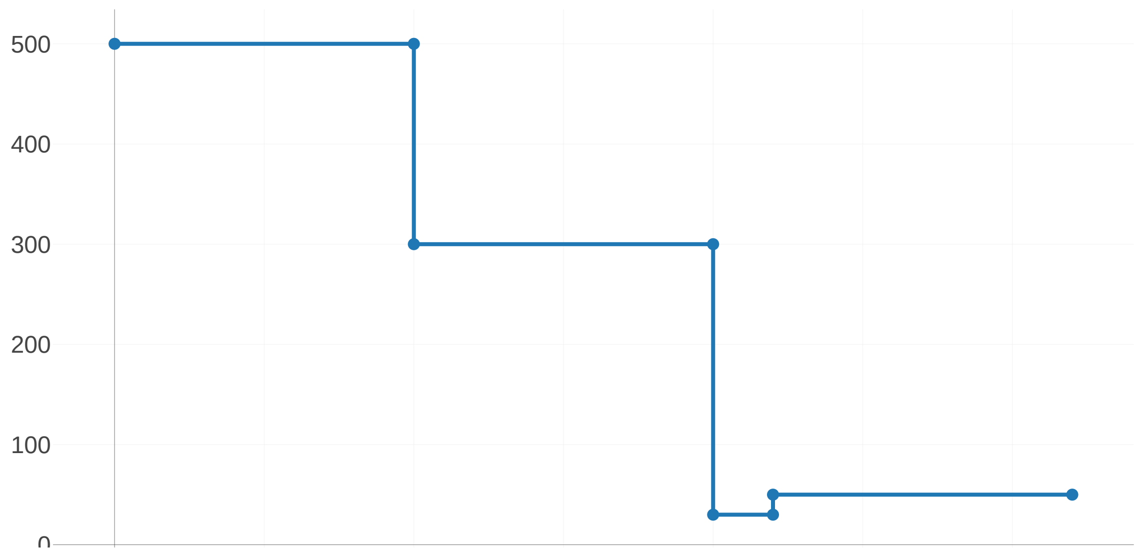
and we have

```
from math import comb as choose

def classic_skyline_N_e(coalescence_times):
        return list(map(lambda k, t: (1/2)*choose(k, 2)*t, reversed(range(1,
len(coalescence_times)+2)),  coalescence_times))

print(classic_skyline_N_e([100, 100, 20, 100]))
```

If we print the graph of the population size against the time, we get

```
from matplotlib import pyplot as plt
times_between_coalescence = [100, 100, 20, 100]
times = [sum(times_between_coalescence[0:i]) for i in range(1,
len(times_between_coalescence)+1)]
population_sizes = classic_skyline_N_e(times_between_coalescence)
print(times)
print(population_sizes)
```

and that's why it's called "skyline" (I guess, THE NAME [MANHATTAN PLOT] WAS ALREADY OCCUPIED)




0
0
50
100
150
200
250
300