Poisson generator:

[Basic Concepts of the Poisson Process (probabilitycourse.com)](https://www.probabilitycourse.com/chapter11/11_1_2_basic_concepts_of_the_poisson_process.php)

Interarrival times = e hoch lambda, daher für Y oft generieren, und dann cumsum.

Für Z: e hoch u – I gibt uns wann es fired

**expovariate()** is an inbuilt method of the random module. It is used to return a random floating point number with [exponential distribution](https://en.wikipedia.org/wiki/Exponential_distribution).

Checken ob poissonGenerator inputPoissonGenerator sinn macht!

Macht keinen sinn! Ich sollte das während der simulation machen, nicht davor, und auch nicht so wie jetzt, sondern mit einer wahrscheinlichkeit, ob das neuron in diesem timestep feuert.

Das schaut gut aus: [Neuron\_Poisson (tu-chemnitz.de)](https://www.tu-chemnitz.de/informatik/KI/scripts/ws0910/Neuron_Poisson.pdf) … der link und die pobc VO agreen, dass fire rate( 25Hz) \* dt reichen um zu predicten obs in dem timestep an spike gibt. (da prob dass mehr als 1 spike passiert quasi 0 ist, da dt klein.

Firing rate 20hz

20 \* 0.001 seconds = 0.02 = 2% chance => etwa alle 2 \*5 \* 10 = 100 alle 50ms? Ein spike? 1000ms/50ms = 20 spikes in einer sekunde

Ich habe die inputspike generation geändert auf during simulation, rennt viel schneller jetzt

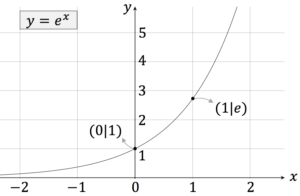
# WEIGHTS:

Meine weights sind jetzt zwischen 0 und 5 und c is 10 oder so. Daraus folgt positives membrane potential

Im paper mit c = 1 werden alle weights zwingend negativ. Und in der realität liegt das membrane potential bei -70mV!! Also is es eh plausibel und je negativer es wird desto eher wird ein aktionspotential ausgelöst. Die schwellenspannung bei der ausgelöst wird ist -50mV, das heißt es muss positiver werden um zu feuern.

Paper sagt dass die weights auf log(p) hinstreben. P ist die wahrscheinlichkeit dass Y neuron innnerhalb von sigma gefeurt hat vor dem Z neuron. Log(1) = 0 … log(0.01) = -2

Weiters sagt es bei (3) dass p(z fires) proportional zu exp(Uk – I) ist.



Was passen würde, da bei u = 0 die feuerwahrscheinlichkeit 1 wäre. Und bei negativen werten kleiner 1. Und dann sagt es, dass rk = exp(U – I) ist, was passt!!!!!!

Das muss ich alles so probieren, und vorallem müsste ich dann nichtmal die weights beschränken. Außer mby auf -1000 oder so. Eher -5 oder so! da log(0.00001) = -5

OMG in weight offsets and positive weights steht das!ß1?!?!?!?1 wtf wie dumm bin ich.

# Training of shape network:

Weights von angle network sind fixed.

Shapes sind zur zeit nicht random gedreht, aber könnte ich adaptieren

TODO: update zugriff auf ZSpikes around line 153 for calc of ZTilde, da es jetzt eine dimension mehr hat.