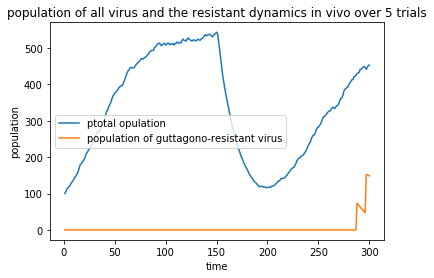
**experiment report**

# Problem 4

**First Results:** In the first picture, the results for the blue line is roughly consistently with the evidence. Before 150 time step, i.e the point where patient start to take a drug, viruses grow quickly and nearly stop growing In 100 time step. It shows that the population density is one of the effect to the growth of virus.

But the orange line is not consistent with the evidence. The population of resistant viruses keep zero until 280 time step.



**Guess:** the error will be either in the function that calculates the population or in the reproduce function that administrative the reproduction.

**Next:** design a repeatable experiment for the simplest version of the same problem

**Experiment:**

set the parameters as below :1 trial, 10 time steps, 10 initial viruses, 100 max population, 0.5 mutProb

Print: print the population every time step, print the resistance of the offspring reproduced by original viruses every time step. If there 10 viruses in after clearance in a time step, there will be roughly (probability\*10) offspring reproduced.

**Results:** we can see that even though the resistant viruses were created through mutation, but the population didn’t grow, So the range of error is narrowed down to the function that calculate the population of the resistant viruses.

In order to check it clearly without 干扰, so we delete the function of clearing the viruses(comment that out), and check the how it run

|  |
| --- |
| timestep:0,total population after clearance: 10 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:1,total population after clearance: 12 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:2,total population after clearance: 14 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} True  timestep:3,total population after clearance: 15 resistant pop:0  timestep:4,total population after clearance: 15 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:5,total population after clearance: 18 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:6,total population after clearance: 20 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} True  timestep:7,total population after clearance: 21 resistant pop:0  timestep:8,total population after clearance: 19 resistant pop:0  timestep:9,total population after clearance: 18 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} True |

**Results(following):**

obviously, the number of reproduced offspring is consistent with the total population growth. So far, except the calculation of the population resistant viruses, there are no other error from the given evidence.

|  |
| --- |
| timestep:0,total population after clearance: 10 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:1,total population after clearance: 12 resistant pop:0  timestep:2,total population after clearance: 12 resistant pop:0  timestep:3,total population after clearance: 12 resistant pop:0  timestep:4,total population after clearance: 12 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:5,total population after clearance: 14 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} True  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:6,total population after clearance: 17 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:7,total population after clearance: 18 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} True  timestep:8,total population after clearance: 19 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} False  timestep:9,total population after clearance: 21 resistant pop:0  the resistance to guttagonol of offspring after reproduce:{0} True  the resistance to guttagonol of offspring after reproduce:{0} False  the resistance to guttagonol of offspring after reproduce:{0} True |

**Dig**

Eventually, 定位出错误（记得还有一个单词）located the error

the line of code marked in red is the source of error

isResist=True # this should be reset in every check of the virus

|  |
| --- |
| def getResistPop(self, drugResist):  """  Get the population of virus particles resistant to the drugs listed in  drugResist.  drugResist: Which drug resistances to include in the population (a list  of strings - e.g. ['guttagonol'] or ['guttagonol', 'grimpex'])  returns: the population of viruses (an integer) with resistances to all  drugs in the drugResist list.  """  # TODO  popResisV=0  isResist=True # this should be reset in every check of the virus  for virus in self.viruses:  # if type(virus)==ResistantVirus:  # ??? is it right  # the instance of superclass would not have the attribute of  # its subclass, but vice-versa, i.e. if a statement is instance.attribute,  # then it will directly override that of superclass, no matter how many arguments of the method  for drug in drugResist:  isResist=isResist and virus.getResistance(drug)  if isResist: popResisV+=1  # print(isResist,popResisV)  return popResisV |

**Then:**

Modify the code, then run It again, results is shown below, one is smaller version, one is bigger version.

what the hell

We won’t panic, think, whoops, forget to uncomment out the codes that execute clearance function for the viruses.

|  |
| --- |
|  |

**Finally,** v1.0

uncomment that out and run it

|  |
| --- |
|  |

**Finally v1.1**

Reset all the parameters to the original bigger version and run it

|  |
| --- |
| maxBirthProb,clearProb,resistances,muProb=0.1,0.05,{'guttagonol':False},0.005  timesteps,numViruses,maxPop=300,100,1000  the first one only run 1 times, which is sort of jaggy, the second one smoother over 10 trials |

**For fun ,** we run it without administering guttagonol to the patient

|  |
| --- |
|  |

# problem 5

0,75,150,300 time steps before administering guttagonol to the patient

|  |
| --- |
|  |

# problem5

|  |
| --- |
| Histogram plotting for preSteps 0  Histogram plotting done for preSteps 0  Histogram plotting for preSteps 75  Histogram plotting done for preSteps 75  Histogram plotting for preSteps 150  Histogram plotting done for preSteps 150  Histogram plotting for preSteps 300  Histogram plotting done for preSteps 300  time using: 112.20539927482605 |

|  |
| --- |
| Histogram plotting for preSteps 0  Histogram plotting done for preSteps 0  Histogram plotting for preSteps 75  Histogram plotting done for preSteps 75  Histogram plotting for preSteps 150  Histogram plotting done for preSteps 150  Histogram plotting for preSteps 300  Histogram plotting done for preSteps 300  time using: 167.0388786792755 |

# problem6(75 middle time steps is the best)

|  |
| --- |
| for 30 patients  Histogram plotting for middle timesteps 300  Histogram plotting done for middle timesteps 300  Histogram plotting for middle timesteps 150  Histogram plotting done for middle timesteps 150  Histogram plotting for middle timesteps 75  Histogram plotting done for middle timesteps 75  Histogram plotting for middle timesteps 0  Histogram plotting done for middle timesteps 0  time using: 30.40410017967224 |

|  |
| --- |
| for 300 patients  Histogram plotting for middle timesteps 300  Histogram plotting done for middle timesteps 300  Histogram plotting for middle timesteps 150  Histogram plotting done for middle timesteps 150  Histogram plotting for middle timesteps 75  Histogram plotting done for middle timesteps 75  Histogram plotting for middle timesteps 0  Histogram plotting done for middle timesteps 0  time using: 315.30966567993164 |

# problem7

|  |
| --- |
| preSteps=150  postSteps=850  total=1000 |

|  |
| --- |
| preSteps=150  postSteps=850  total=1000 |

# problem8

forget to take drugs

add some code using random and probability to override the effect of prescription in method ResistantVirus.reproduce()