

# AMATH 536 Problem Set 3

Cade Ballew #2120804

May 6, 2022

## 1 Problem 1

Assume that there are  $n_i$  resistance mutations providing resistance to drug  $i$ , and that the point mutation rate is  $\mu$ . We will further assume that there are no mutations that provide resistance to more than one of the drugs. Process is initiated at time 0 with a single cell sensitive to all drugs. All cells in the process have birth rate  $b$  and death rate  $d$ , with  $b > d$ .

### 1.1 Part a

To calculate the expected number of cells resistant to all  $n$  drugs at time  $t$  arising through a single path  $0 \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow n$ , we first define

$$\alpha_j = \begin{cases} (1 - \mu n_{j+1})b - d, & j = 0, \dots, n-1 \\ b - d, & j = n \end{cases}$$

as the growth rate for type  $j$  cells and

$$\beta_j = \mu b n_j$$

to be the mutation rate for type  $j$  cells from type  $j-1$  cells,  $j = 1, \dots, n$ . If  $X_k(t)$  is the expected number of cells resistant to first  $k$  drugs, then we have the system of ODEs

$$\begin{aligned} X'_0(t) &= \alpha_0 X_0(t) \\ X'_j(t) &= \alpha_j X_j(t) + \beta_j X_{j-1}(t), \quad j = 1, \dots, n. \end{aligned}$$

For an initial condition, assume that we start with 1 cell with no mutations at time  $t = 0$ , so  $X_0(0) = 1$  and  $X_j(0) = 0$  for  $j = 1, \dots, n$ . We tackle this system by means of the Laplace transform. Namely, we use the fact that

$$\mathcal{L}[f'(t)](s) = s\mathcal{L}[f](s) - f(0^+)$$

and denote  $\mathcal{L}[X_j(t)] = V_j(s)$  to find that our system becomes

$$\begin{aligned} sV_0(s) - 1 &= \alpha_0 V_0(s) \\ sV_j(s) &= \alpha_j V_j(s) + \beta_j V_{j-1}(s), \quad j = 1, \dots, n \end{aligned}$$

which a recurrence relation that is easily solvable. Namely,

$$\begin{aligned} V_0(s) &= \frac{1}{s - \alpha_0} \\ V_j(s) &= \frac{\beta_j V_{j-1}(s)}{s - \alpha_j}, \quad j = 1, \dots, n \end{aligned}$$

which can be recursively found to have form

$$V_j(s) = \frac{1}{s - \alpha_0} \prod_{k=1}^j \frac{\beta_k}{s - \alpha_k}$$

for  $j = 1, \dots, n$ . If we adopt the convention that  $\beta_0 = 1$ , this can be written as

$$V_j(s) = \prod_{k=0}^j \frac{\beta_k}{s - \alpha_k}$$

for all  $j = 0, \dots, n$ . To compute the inverse transform, we use the general partial fraction decomposition formula to write

$$\prod_{k=0}^j \frac{1}{s - \alpha_k} = \sum_{k=0}^j \frac{1}{Q'(\alpha_k)} \frac{1}{s - \alpha_k}$$

where

$$Q(s) = \prod_{\ell=0}^j (s - \alpha_\ell),$$

so

$$Q'(s) = \sum_{k=0}^j \left( \prod_{\ell=0, \ell \neq k}^j (s - \alpha_\ell) \right)$$

and

$$Q'(\alpha_k) = \prod_{\ell=0, \ell \neq k}^j (\alpha_k - \alpha_\ell).$$

Thus,

$$V_j(s) = \left( \prod_{k=0}^j \beta_k \right) \sum_{k=0}^j \frac{1}{(s - \alpha_k) \prod_{\ell=0, \ell \neq k}^j (\alpha_k - \alpha_\ell)}.$$

Now, use that in general

$$\mathcal{L}^{-1} \left[ \frac{1}{s - d} \right] = e^{dt}$$

for a constant  $d$  and use the linearity of the Laplace transform to get that

$$X_j(t) = \left( \prod_{k=0}^j \beta_k \right) \sum_{k=0}^j \frac{e^{\alpha_k t}}{\prod_{\ell=0, \ell \neq k}^j (\alpha_k - \alpha_\ell)}.$$

Finally, we can plug back in our coefficients for  $j = 1, \dots, n-1$  while noting that

$$\alpha_k - \alpha_\ell = (1 - \mu n_{k+1})b - d - ((1 - \mu n_{\ell+1})b - d) = b\mu(n_{\ell+1} - n_{k+1}).$$

for  $k, \ell = 0, \dots, n-1$ . Then,

$$\begin{aligned} X_j(t) &= \left( \prod_{k=1}^j \mu b n_k \right) \sum_{k=0}^j \frac{e^{((1-\mu n_{k+1})b-d)t}}{\prod_{\ell=0, \ell \neq k}^j b\mu(n_{\ell+1} - n_{k+1})} \\ &= \left( (\mu b)^j \prod_{k=1}^j n_k \right) \sum_{k=0}^j \frac{e^{((1-\mu n_{k+1})b-d)t}}{(\mu b)^j \prod_{\ell=0, \ell \neq k}^j (n_{\ell+1} - n_{k+1})} \\ &= \left( \prod_{k=1}^j n_k \right) \sum_{k=0}^j \frac{e^{((1-\mu n_{k+1})b-d)t}}{\prod_{\ell=0, \ell \neq k}^j (n_{\ell+1} - n_{k+1})}. \end{aligned}$$

When  $j = 0$ , we instead have simply that

$$X_0(t) = e^{((1-\mu n_1)b-d)t}$$

as expected. When  $j = n$ , it is perhaps easiest to simply adopt the convention that  $n_{n+1} = 0$  so that we may use our prior analysis to write

$$X_n(t) = \left( \prod_{k=1}^n n_k \right) \sum_{k=0}^n \frac{e^{((1-\mu n_{k+1})b-d)t}}{\prod_{\ell=0, \ell \neq k}^n (n_{\ell+1} - n_{k+1})}$$

as before. Note that this does have the implicit assumption that the  $n_k$ s are distinct.

## 1.2 Part b

To condition on the survival of the process, we note that the survival probability is  $1 - \frac{d}{b}$ , so we set

$$N = \frac{e^{(b-d)t}}{1 - \frac{d}{b}} = \frac{b}{b-d} e^{(b-d)t},$$

so we plug in

$$t = \frac{\log\left(\frac{rN}{b}\right)}{r}$$

if we set  $r = b-d$  and multiply our entire expression by the survival probability. Then,

$$\begin{aligned} X_n(N) &= \frac{b}{r} \left( \prod_{k=1}^n n_k \right) \sum_{k=0}^n \frac{e^{((1-\mu n_{k+1})b-d) \frac{\log\left(\frac{rN}{b}\right)}{r}}}{\prod_{\ell=0, \ell \neq k}^n (n_{\ell+1} - n_{k+1})} \\ &= \frac{b}{r} \left( \prod_{k=1}^n n_k \right) \sum_{k=0}^n \frac{\left(\frac{rN}{b}\right)^{((1-\mu n_{k+1})b-d)/r}}{\prod_{\ell=0, \ell \neq k}^n (n_{\ell+1} - n_{k+1})} \end{aligned}$$

where we have maintained our convention that  $n_{n+1} = 0$ .

### 1.3 Part c

Now, we start with our expression from part a so that we may Taylor expand

$$e^{((1-\mu n_{k+1})b-d)t} = e^{rt} \sum_{s=0}^{\infty} \frac{(-\mu n_{k+1}bt)^s}{s!},$$

so

$$\begin{aligned} X_n(t) &= \frac{b}{r} \left( \prod_{k=1}^n n_k \right) \sum_{k=0}^n \frac{e^{rt} \sum_{s=0}^{\infty} \frac{(-\mu n_{k+1}bt)^s}{s!}}{\prod_{\ell=0, \ell \neq k}^n (n_{\ell+1} - n_{k+1})} \\ &= \frac{b}{r} e^{rt} \left( \prod_{k=1}^n n_k \right) \sum_{k=0}^n \sum_{s=0}^{\infty} \frac{(\mu bt)^s}{s!} \frac{(-n_{k+1})^s}{\prod_{\ell=0, \ell \neq k}^n (n_{\ell+1} - n_{k+1})} \\ &= \frac{b}{r} e^{rt} \left( \prod_{k=1}^n n_k \right) \sum_{s=0}^{\infty} \frac{(\mu bt)^s}{s!} \sum_{k=1}^{n+1} \frac{(-n_k)^s}{\prod_{\ell=1, \ell \neq k}^{n+1} (n_{\ell} - n_k)} \\ &= \frac{b}{r} e^{rt} \left( \prod_{k=1}^n n_k \right) \sum_{s=0}^{\infty} \frac{(\mu bt)^s}{s!} \sum_{k=1}^n \frac{(-n_k)^s}{(-n_k) \prod_{\ell=1, \ell \neq k}^n (n_{\ell} - n_k)} \end{aligned}$$

since we have the convention that  $n_{n+1} = 0$ . Now, we can use the provided identity to find that

$$\sum_{k=1}^n \frac{(-n_k)^s}{n_k \prod_{\ell=1, \ell \neq k}^n (n_{\ell} - n_k)} = \begin{cases} \frac{1}{\prod_{\ell=1}^n n_{\ell}}, & s = 0 \\ 0, & 1 \leq s \leq n \\ -1, & s = n \end{cases}$$

which is nonzero only when  $s = 0, n$ . We then truncate all terms where  $s > n$  from our Taylor series which leaves us with

$$X_n(t) = \frac{b}{r} e^{rt} \left( \prod_{k=1}^n n_k \right) (-1) \left( \frac{1}{\prod_{\ell=1}^n n_{\ell}} + \frac{(\mu bt)^n}{n!} (-1) \right) = \frac{b}{r} e^{rt} \left( \left( \prod_{k=1}^n n_k \right) \frac{(\mu bt)^n}{n!} - 1 \right).$$

Plugging back in  $t = \frac{\log(\frac{rN}{b})}{r}$ ,

$$\begin{aligned} X_n(N) &= \frac{b}{r} e^{\log(\frac{rN}{b})} \left( \left( \prod_{k=1}^n n_k \right) \frac{(\mu b \frac{\log(\frac{rN}{b})}{r})^n}{n!} - 1 \right) \\ &= N \left( \left( \prod_{k=1}^n n_k \right) \frac{(\mu b \log(\frac{rN}{b}))^n}{r^n n!} - 1 \right). \end{aligned}$$

### 1.4 Part d

To compute the expected number of cells resistant to all  $n$  drugs,  $1, 2, \dots, n$  in a tumor containing  $N$  cells, we note that there are  $n!$  possible paths so we simply

multiply our answer from part c by  $n!$  to get

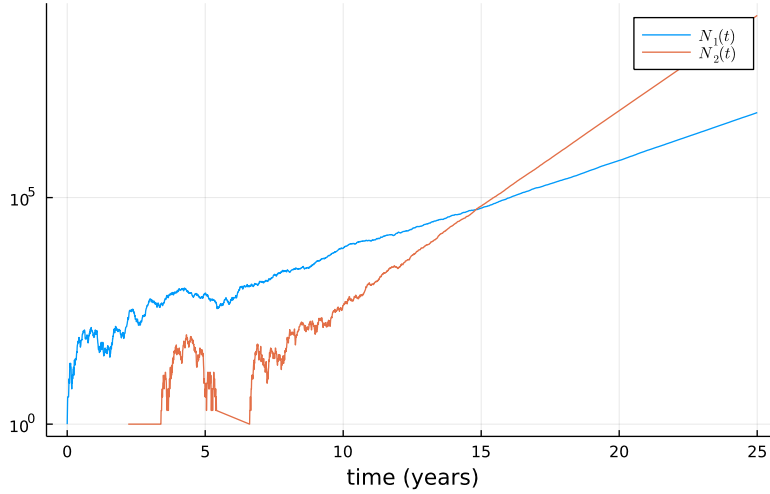
$$N \left( \left( \prod_{k=1}^n n_k \right) \left( \frac{\mu b \log \left( \frac{rN}{b} \right)}{r} \right)^n - n! \right).$$

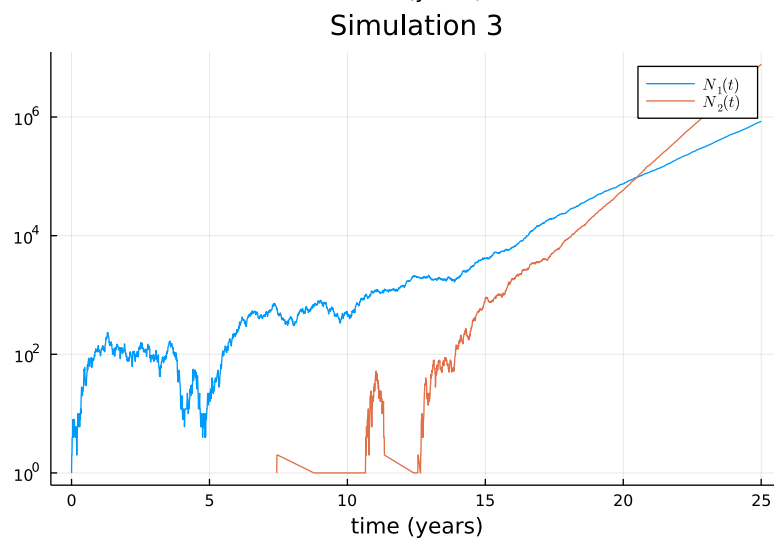
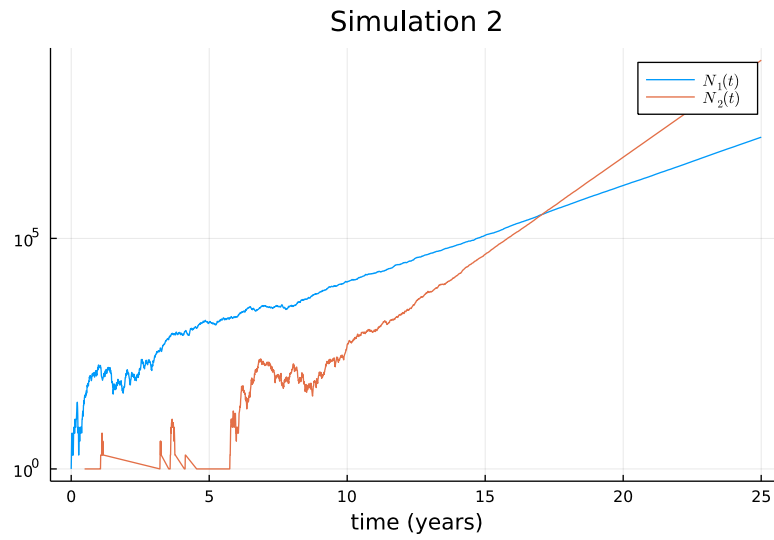
## 2 Problem 2

### 2.1 Part a

We simulate the discrete-time branching process from the paper "Accumulation of driver and passenger mutations during tumor progression" with only type 1 and type 2 cells by taking the parameter values  $u = 3.4 \times 10^{-5}$ ,  $T = 3$  days, and  $s = 0.4\%$  and letting  $d_j = \frac{1}{2}(1 - s)^j$ ,  $j = 1, 2$ . We sample births, deaths, and mutations at each timestep via the multinomial distribution in (S2). The following are three simulations in which the process does not die out over 25 years.

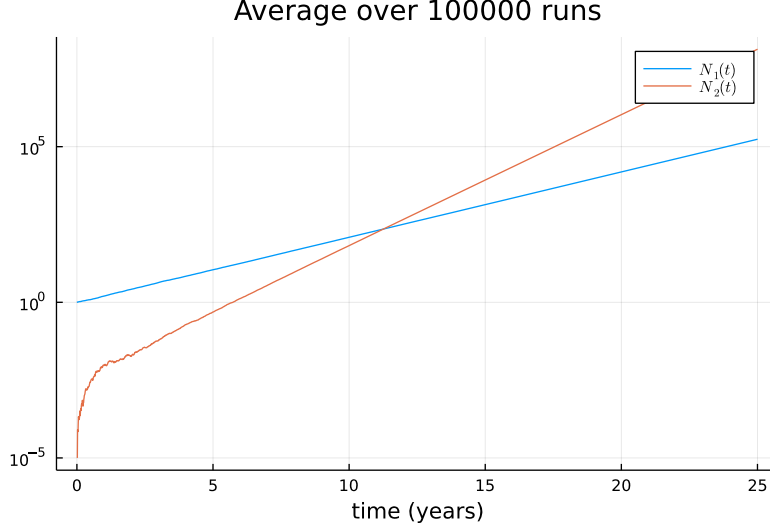
Simulation 1





## 2.2 Part b

We now run the simulation 100000 times and produce the following plot of the average number of cells of each time present at each timestep.



### 2.3 Part c

To derive the average numbers of type 1 and type 2 cells in the process analytically, we keep with the notation of the paper by letting  $u$  denote the mutation rate,  $b_1, b_2$  the respective birth rates, and  $d_1, d_2$  the respective death rates. If we let  $x_t$  denote the expected number of type 1 cells at timestep  $t$  (corresponding to time  $tT$ ) and let  $y_t$  denote the expected number of type 2 cells at timestep  $t$ , we have that

$$\begin{aligned} x_t &= x_{t-1} + b_1(1-u)x_{t-1} - d_1x_{t-1} \\ y_t &= y_{t-1} + b_2y_{t-1} - d_2y_{t-1} + b_1ux_{t-1}. \end{aligned}$$

Note that we have assumed that type 2 cells are still able to mutate. Then,

$$x_t = (1 + b_1(1-u) - d_1)x_{t-1},$$

so we have that

$$x_t = (1 + b_1(1-u) - d_1)^t$$

since  $x_0 = 1$ . Then, we find that

$$y_t = (1 + b_2 - d_2)y_{t-1} + b_1u(1 + b_1(1-u) - d_1)^{t-1}.$$

We can use Mathematica to solve this recurrence relationship (noting that  $y_0 = 0$ ) which gives that

$$y_t = \frac{b_1u((1 + b_2 - d_2)^t - (1 + b_1(1-u) - d_1)^t)}{(b_2 - d_2) - (b_1(1-u) - d_1)}.$$

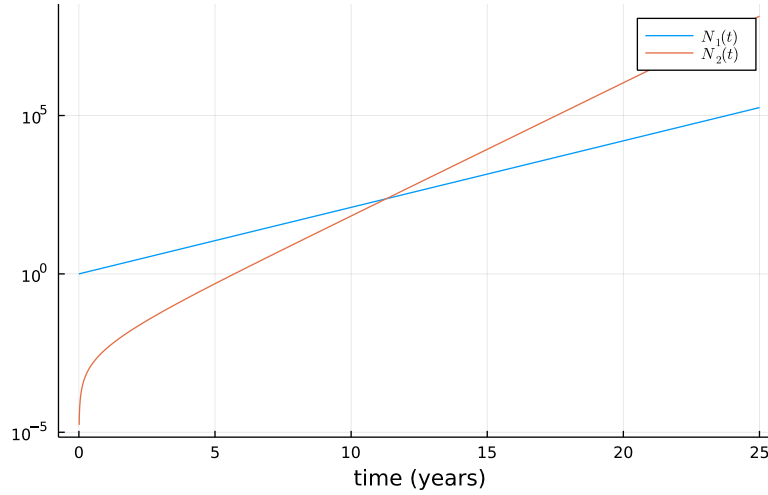
If we instead wish to have  $t$  be the true time, we can simply write

$$N_1(t) = (1 + b_1(1 - u) - d_1)^{t/T},$$

$$N_2(t) = \frac{b_1 u ((1 + b_2 - d_2)^{t/T} - (1 + b_1(1 - u) - d_1)^{t/T})}{(b_2 - d_2) - (b_1(1 - u) - d_1)}.$$

If we plot this, we observe the following

Average number of cells computed analytically



In general, this seems to match what we observed in our 100000 trials. Using more trials would likely smooth out the curve for  $N_2(t)$  to more closely match our analytical solution.

### 3 Appendix A

The following Julia code is used for problem 2.

```
using Plots, Distributions, Random, LaTeXStrings, Printf

u, T, s = 3.4e-5, 3/365, 0.004
keepExtinct = false
T_e = 25. #final time in years
Random.seed!(1234)

function simul(u, T, s, T_e, keepExtinct)
    n = convert{Int64, ceil(T_e/T)}

    x1 = zeros{Int64, n+1}
    x2 = zeros{Int64, n+1}
    t = zeros{n+1}
    x1[1], x2[1], t[1] = 1, 0, 0.

    d1 = (1-s)/2
    d2 = (1-s)^2/2
    b1 = 1-d1
    b2 = 1-d2
```



```

probs1 = [b1*(1-u), d1, b1*u]
probs2 = [b2, d2, 0]

for i = 2:n+1
    dist1 = Multinomial(x1[i-1], probs1)
    dist2 = Multinomial(x2[i-1], probs2)
    result1 = rand(dist1)
    result2 = rand(dist2)
    x1[i] = x1[i-1]+result1[1]-result1[2]
    x2[i] = x2[i-1]+result2[1]-result2[2]+result1[3]
    t[i] = t[i-1]+T

    if (x1[i]==0 && x2[i]==0) && keepExtinct == false
        return simul(u, T, s, Te, keepExtinct)
    end
end
return x1, x2, t
end

trial11, trial12, t = simul(u, T, s, Te, keepExtinct)
ind1 = trial11.>0
ind2 = trial12.>0
p1 = plot(t[ind1],trial11[ind1],label=L"N_1(t)", yaxis=:log)
plot!(t[ind2],trial12[ind2],label=L"N_2(t)")
xlabel! ("time (years)")
title! ("Simulation 1")
display(p1)
savefig(p1, "sim1.pdf")

trial21, trial22, t = simul(u, T, s, Te, keepExtinct)
ind1 = trial21.>0
ind2 = trial22.>0
p2 = plot(t[ind1],trial21[ind1],label=L"N_1(t)", yaxis=:log)
plot!(t[ind2],trial22[ind2],label=L"N_2(t)")
xlabel! ("time (years)")
title! ("Simulation 2")
display(p2)
savefig(p2, "sim2.pdf")

trial31, trial32, t = simul(u, T, s, Te, keepExtinct)
ind1 = trial31.>0
ind2 = trial32.>0
p3 = plot(t[ind1],trial31[ind1],label=L"N_1(t)", yaxis=:log)
plot!(t[ind2],trial32[ind2],label=L"N_2(t)")
xlabel! ("time (years)")
title! ("Simulation 3")
display(p3)
savefig(p3, "sim3.pdf")

#part b
function compute_avg(u, T, s, Te, numruns)
    n = convert{Int64,ceil(Te/T)}
    avg1 = zeros(n+1)
    avg2 = zeros(n+1)
    for i = 1:numruns
        x1, x2, t = simul(u, T, s, Te, true)
        avg1 .+= x1./numruns
        avg2 .+= x2./numruns
    end
    return avg1, avg2, t
end

numruns = 100000 #chosen number of runs
avg1, avg2, t = compute_avg(u, T, s, Te, numruns)
ind1 = avg1.>0
ind2 = avg2.>0
p4 = plot(t[ind1],avg1[ind1],label=L"N_1(t)", yaxis=:log)
plot!(t[ind2],avg2[ind2],label=L"N_2(t)")

```

```

xlabel! ("time (years)")
title! (@sprintf("Average over %i runs", numruns))
display(p4)
savefig(p4, "avg.pdf")

#part c
d1 = (1-s)/2
d2 = (1-s)^2/2
b1 = 1-d1
b2 = 1-d2
xt = t -> (1+b1*(1-u)-d1)^(round(t/T))
yt = t -> (b1*u*((1+b2-d2)^(round(t/T))-(1+b1*(1-u)-d1)^(round(t/T))))/(b2-d2)-(b1*(1-u)-d1)

ind1 = xt.(t).>0
ind2 = yt.(t).>0
p5 = plot(t[ind1], xt.(t[ind1]), label=L"N_1(t)", yaxis=:log)
plot!(t[ind2], yt.(t[ind2]), label=L"N_2(t)")
xlabel! ("time (years)")
title! ("Average number of cells computed analytically")
display(p5)
savefig(p5, "true_avg.pdf")

```

The following line of Mathematica code is used in problem 2 part c.

```

RSolve[{y[t + 1] == (1 + b - d) y[t] + au (1 + a (1 - u) - c),
y[0] == 0}, y[t], t]

```