Prediction Model for Human and Mosquito Populations in Malaria Afflicted Regions

By Team Seven-Up: Esther Ryu, Grace Lo, Zach Miller, Percy Adams, Jacob Harless, Temilola Aderibigbe, Apoorv Palkar

July 13, 2017
Basic Model Assumptions -> System Simplification

- Infected mosquitoes always die (ie. no recovery)

- No new people/mosquitoes move into a population

- The malaria death rate, human death rate, immune rate, recovery rate, etc. are assumed to be the same all year long

- Simplifies system by modeling both the number of sick/healthy/immune villagers over time and the number of healthy/infected mosquitoes over time
def Healthy_Villagers ( i, hV, sV, brV, drV, rrV, M, iM, brfM, iV ) :
    hV[i+1] = hV[i] + (((hV[i] + iV[i] + sV[i]) * brV) - (hV[i] * drV) - (hV[i] * brfM * (iM[i] / max(M[i],1))))

def Sick_Villagers ( i, hV, sV, rrV, irV, drV, M, iM, brfM, midrV ,rV):
    sV[i+1] = sV[i] + ((brfM * hV[i] * (iM[i] / max(M[i],1))) - (sV[i] * rrV) - (sV[i] * irV) - (sV[i] * (drV+midrV)) + (rV[i]*.17))

def Immune_Villagers (i, iV, sV, irV, drV ):
    iV[i+1] = iV[i] + ((sV[i] * irV) - (iV[i] * drV))
def Healthy_Mosquitoes (i, V, sV, M, hM, brM, drM, brfM ):
    hM[i+1] = hM[i] + ((M[i] * brM) - (hM[i] * drM) - ((sV[i] / max((V[i]+1),1)) * hM[i] * brfM))

def Infected_Mosquitoes (i, V, sV, hM, iM, drM, brfM ):
    iM[i+1] = iM[i] + (((sV[i] / max((V[i]+1),1)) * hM[i] * brfM) - (iM[i] * drM))
200 days, 500 villagers, 1 sick, 1000 healthy mosquitoes
1400 days, 500 villagers, 1 sick, 1000 healthy mosquitoes
Extra Model Additions: Relapse (Code)

```python
89 def Healthy_Villagers ( i, hV, sV, brV, drV, rrV, M, iM, brfM, iV ) :
90     hV[i+1] = hV[i] + (((hV[i] + iV[i] + sV[i]) * brV) - (hV[i] * drV) - (hV[i] * brfM * (iM[i] / max(M[i],1))))
91
92 def Rem_Villagers (i, sV, rrV, drV, rerV):
93     rV[i+1] = rV[i] + ((sV[i] * rrV) - (rV[i]*rerV) - (rV[i] * drV))
94
95 def Sick_Villagers ( i, hV, sV, rrV, irV, drV, M, iM, brfM, midrV ,rV):
96     sV[i+1] = sV[i] + ((brfM * hV[i] * (iM[i] / max(M[i],1))) - (sV[i] * rrV) - (sV[i] * irV) - (sV[i] * (drV+midrV)) + (rV[i]*17))
```
Extra Model Additions: Relapse (Graphs)
Extra Model Additions: Seasonal Effect

```python
for i in range(N):
    brM = (math.cos((t[i] * (2 * math.pi)) / (365 + 1)) * (0.02 / 2))
    drM = (math.cos((t[i] * (2 * math.pi)) / (365 + 1)) * (0.022 / 2))
    t[i+1] = t[i] + time_step
```
Extra Model Additions: Seasonal Effect

With effect

Without effect
Future Steps

Using a differential equation changes runtime from $O(n)$ to $O(1)$

Test with larger populations -> Using HPC to accelerate the calculations of using larger datasets

Zika: would have to consider birth rate in infected patients, as Zika can be spread through pregnancy.

Can also be sexually transmitted