

Actividad 11: Apocalipsis Zombie

Reyna Cornelio

30 de Abril 2016

Los zombies son una figura popular en la cultura del entretenimiento, se les asocia con brotes o epidemias causadas por virus generados por laboratorios. En esta actividad se modelará un ataque zombie basado en las películas populares. Se introduce un modelo básico para la infección zombi, en la cual se determinan los parámetros que se tomarán en cuenta. Se simularán varios escenarios, para ello se modificará el código ya existente.

Para que puedan surgir los zombies es necesario que un individuo susceptible sea mordido por un individuo infectado, que deja una herida abierta. La herida creada por el zombie tiene la saliva del zombie. Este fluido corporal se mezcla con la sangre, por lo tanto infecta al individuo (previamente susceptible).

Modelo Básico

En este modelo se tomarán en cuenta:

- Las personas susceptibles (S)
- Zombies (Z)
- Personas que murieron por causas naturales (R)

El código que se utilizó para modelar este ataque es el siguiente:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
plt.ion()
plt.rcParams['figure.figsize'] = 10, 8
```

```

P = 0 # birth rate
d = 0.0001 # natural death percent (per day)
B = 0.0095 # transmission percent (per day)
G = 0.0001 # resurect percent (per day)
A = 0.0001 # destroy percent (per day)

# solve the system dy/dt = f(y, t)
def f(y, t):
    Si = y[0]
    Zi = y[1]
    Ri = y[2]
    # the model equations (see Munz et al. 2009)
    f0 = P - B*Si*Zi - d*Si
    f1 = B*Si*Zi + G*Ri - A*Si*Zi
    f2 = d*Si + A*Si*Zi - G*Ri
    return [f0, f1, f2]

# initial conditions
S0 = 500. # initial population
Z0 = 0 # initial zombie population
R0 = 0.05*S0 # 1% of initial pop is dead
y0 = [S0, Z0, R0]
t = np.linspace(0, 5., 1000) # time grid

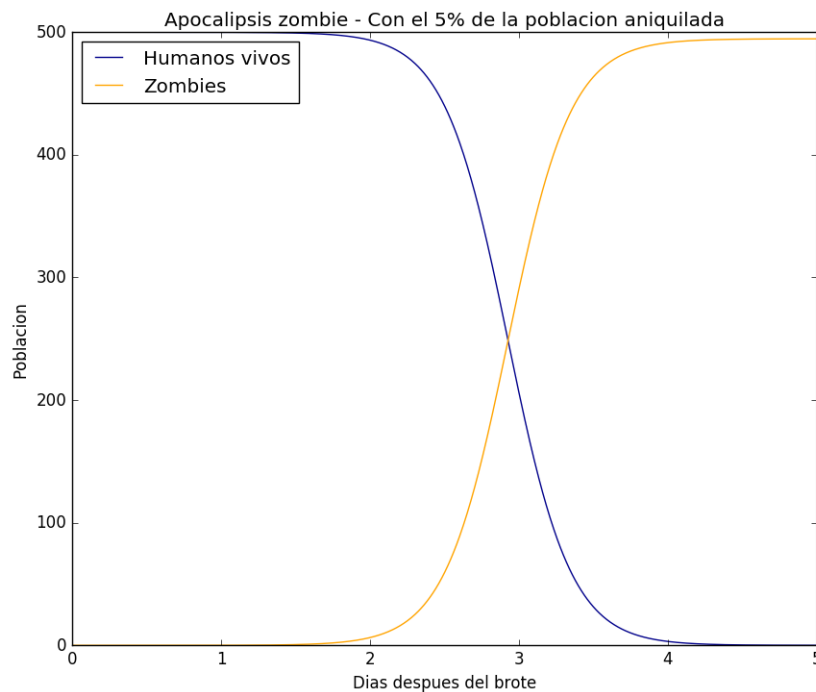
# solve the DEs
soln = odeint(f, y0, t)
S = soln[:, 0]
Z = soln[:, 1]
R = soln[:, 2]

# plot results
plt.figure()
plt.plot(t, S, "Darkblue", label='Humanos vivos')
plt.plot(t, Z, "Orange", label='Zombies')
plt.xlabel('Dias despues del brote')
plt.ylabel('Poblacion')
plt.title('Apocalipsis zombie - Con el 5% de la poblacion aniquilada')

```

```
plt.legend(loc=0)
```

La gráfica que se obtuvo:



Podemos observar que después de 3 días la población de humanos es la misma que la de zombies es decir aproximadamente 250 personas. Pero en cuestión de horas las personas que no están infectadas empiezan a disminuir.

Modo infección

En este modelo se incluye a la infección latente de los individuos infectados. En el cual hay un período de tiempo (aproximadamente 24 horas) después de que el ser humano susceptible es mordido antes se convierte en un zombi. De este modo se amplía el modelo básico para incluir el (más "realista") posibilidad de que un susceptible individuo se infecta antes de sucumbir a la zombificación. Esto es lo que es visto muy a menudo en las representaciones de la cultura pop de zombies ([2, 6, 8]).

```
#Modo infeccion latente  
# zombie apocalypse modeling
```

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
plt.ion()
plt.rcParams['figure.figsize'] = 10, 8

P = 0 # birth rate
d = 0.00010 # natural death percent (per day)
B = 0.009 # transmission percent (per day)
G = 0.0018 # resurect percent (per day)
A = 0.0001 # destroy percent (per day)
poin = 0.0015 # poblacion infectada

# solve the system dy/dt = f(y, t)
def f(y, t):
    Si = y[0]
    Ii = y[1]
    Zi = y[2]
    Ri = y[3]
    # the model equations (see Munz et al. 2009)
    f0 = P - B*Si*Zi - d*Si
    f1 = B*Si*Zi - poin*Ii - d*Ii
    f2 = poin*Ii + G*Ri - A*Si*Zi
    f3 = d*Si + d*Ii + A*Si*Zi - G*Ri
    return [f0, f1, f2, f3]

# condiciones iniciales
S0 = 1382. # población inicial
POI = 1 # población infectada pero no convertida
Z0 = 0 # initial zombie population
R0 = 0 # 1% of initial pop is dead
y0 = [S0, POI, Z0, R0]
t = np.linspace(0, 200, 10000) # time grid

# solve the DEs
soln = odeint(f, y0, t)
S = soln[:, 0]
I = soln[:, 1]

```

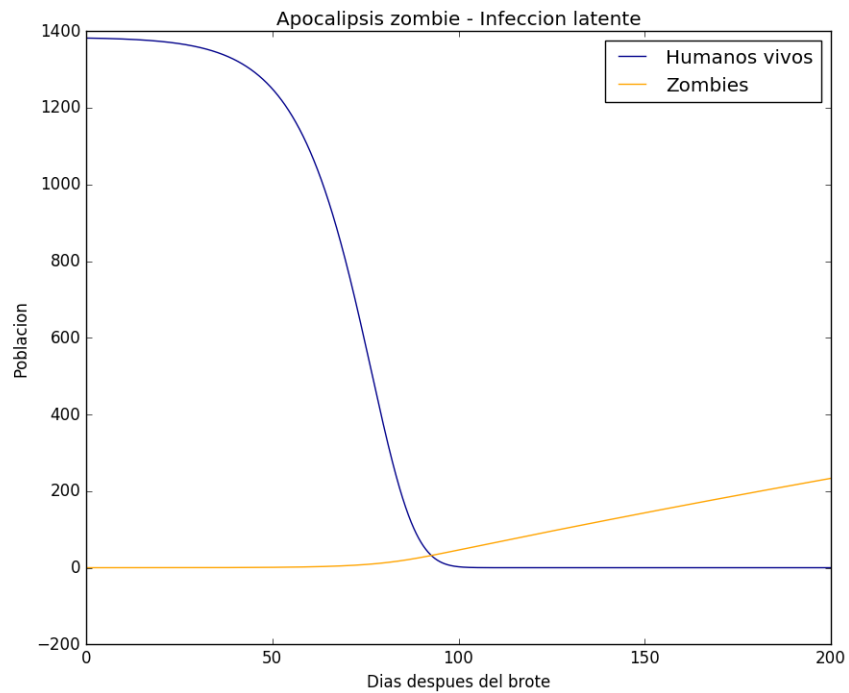
```

Z = soln[:, 2]
R = soln[:, 3]

# plot results
plt.figure()
plt.plot(t, S,"Darkblue", label='Humanos vivos')
plt.plot(t, Z,"Orange", label='Zombies')
plt.xlabel('Dias despues del brote')
plt.ylabel('Poblacion')
plt.title('Apocalipsis zombie - Infeccion latente')
plt.legend(loc=0)

```

La gráfica que se generó:



Se puede observar que después de 100 días la población de zombies esta aumentando pero la poblacion de humanos vivos se mantiene constante.

Modo cuarentena

Con el fin de contener el brote, se decide modelar los efectos de la cuarentena parcial de zombis. En este modelo, se supone que los individuos en cuarentena se eliminan de la población y no puede infectar a nuevas personas siempre que se mantengan en cuarentena.

```
#Modo cuarentena
# zombie apocalypse modeling
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
plt.ion()
plt.rcParams['figure.figsize'] = 10, 8

P = 0 # birth rate
d = 0.0001 # natural death percent (per day)
B = 0.0095 # transmission percent (per day)
G = 0.0001 # resurect percent (per day)
A = 0.0001 # destroy percent (per day)
poin = 0.3 #poblacion infectada
cuar = 0.0005 #cuarentena
poz= 0.082 #producción de zombies
dead= 0.015 #muertes

# solve the system dy/dt = f(y, t)
def f(y, t):
    Si = y[0]
    Ii = y[1]
    Zi = y[2]
    Ri = y[3]
    Qi = y[4]
    # the model equations (see Munz et al. 2009)
    f0 = P - B*Si*Zi - d*Si
    f1 = B*Si*Zi - poin*Ii - d*Ii - cuar*Ii
    f2 = poin*Ii + G*Ri - A*Si*Zi - poz*Zi
    f3 = d*Si + d*Ii + A*Si*Zi - G*Ri + dead* Qi
    f4 = cuar*Ii + poz*Zi - dead*Qi
```

```

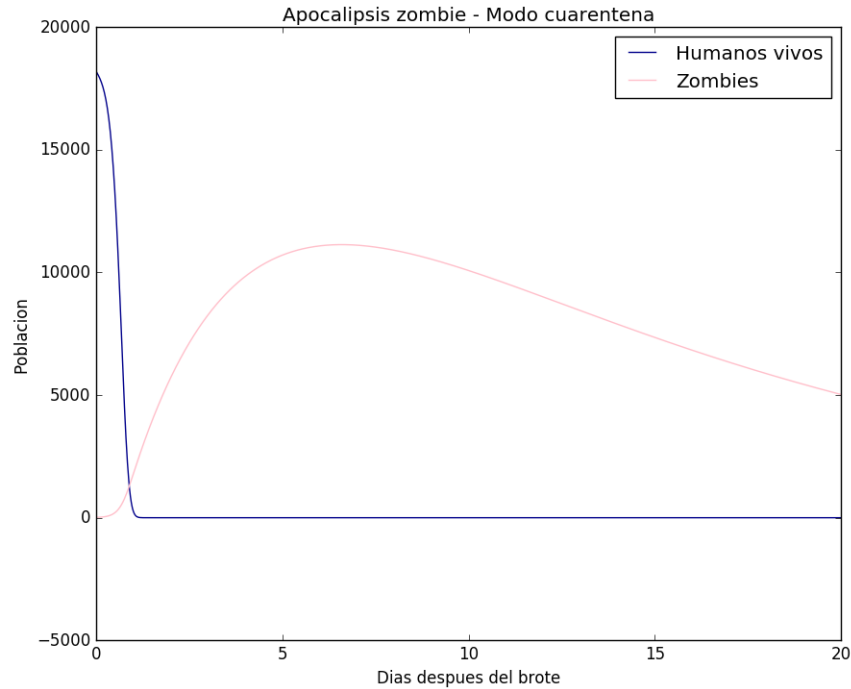
    return [f0, f1, f2, f3, f4]

# initial conditions
S0 = 18200. # initial population
I0 = 150
Z0 = 15 # initial zombie population
Q0 = 130
R0 = 0 # 1% of initial pop is dead
y0 = [S0, I0, Z0, R0, Q0]
t = np.linspace(0, 20, 1000) # time grid

# solve the DEs
soln = odeint(f, y0, t)
S = soln[:, 0]
I = soln[:, 1]
Z = soln[:, 2]
R = soln[:, 3]
Q = soln[:, 4]

# plot results
plt.figure()
plt.plot(t, S, "Darkblue", label='Humanos vivos')
plt.plot(t, Z, "pink", label='Zombies')
plt.xlabel('Dias despues del brote')
plt.ylabel('Poblacion')
plt.title('Apocalipsis zombie - Modo cuarentena')
plt.legend(loc=0)

```



Modo Tratamiento

Supongamos que somos capaces de producir rápidamente una cura para la 'zombie-ismo'. El tratamiento haría al zombie un individuo su forma humana de nuevo. Sin embargo, el nuevo ser humano volvería a ser susceptibles de convertirse en un zombi; Por lo tanto, nuestra cura no proporciona inmunidad.

```
#Modo tratamiento
# zombie apocalypse modeling
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
plt.ion()
plt.rcParams['figure.figsize'] = 10, 8
```

```
P = 0 # birth rate
```



```

d = 0.0001 # natural death percent (per day)
B = 0.0095 # transmission percent (per day)
G = 0.0001 # resurect percent (per day)
A = 0.0001 # destroy percent (per day)
poin = 0.03 #poblacion infectados
cur = 0.01 #poblacion curada

# solve the system dy/dt = f(y, t)
def f(y, t):
    Si = y[0]
    Ii = y[1]
    Zi = y[2]
    Ri = y[3]
    # the model equations (see Munz et al. 2009)
    f0 = P - B*Si*Zi - d*Si + cur*Zi
    f1 = B*Si*Zi - poin*Ii - d*Ii
    f2 = poin*Ii + G*Ri - A*Si*Zi - cur*Zi
    f3 = d*Si + d*Ii + A*Si*Zi - G*Ri
    return [f0, f1, f2, f3]

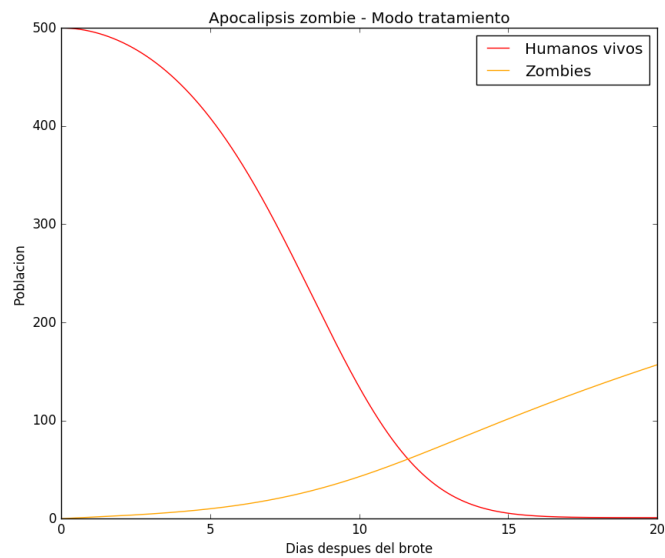
# initial conditions
S0 = 500. # initial population
I0 = 50
Z0 = 0 # initial zombie population
R0 = 0 # 1% of initial pop is dead
y0 = [S0, I0, Z0, R0]
t = np.linspace(0, 20, 1000) # time grid

# solve the DEs
soln = odeint(f, y0, t)
S = soln[:, 0]
I = soln[:, 1]
Z = soln[:, 2]
R = soln[:, 3]

# plot results
plt.figure()
plt.plot(t, S, "red", label='Humanos vivos')

```

```
plt.plot(t, Z, "Orange", label='Zombies')
plt.xlabel('Dias despues del brote')
plt.ylabel('Poblacion')
plt.title('Apocalipsis zombie - Modo tratamiento')
plt.legend(loc=0)
```



1 Bibliografia

http://scipy-cookbook.readthedocs.io/items/Zombie_Apocalypse_ODEINT.html