Supplement 6 for Effect of Ebola Disease Progression on Transmission and Control in Liberia

Model algorithm

Constructing the matrix

For each individual, we defined a row vector comprising of the exposure day, an incubation period and the symptoms period. Every element in the vector represents a single day. The date of exposure was marked with a " -1", and the symptoms period was assigned the values of the relative infectiousness, $f_i^S(t)$. An example of such a vector:

$(-1, 0, 0, 0, 0, 0, 0, 0, 0, 208.14, 559.764, 1169.51, 1964.58, 2800.02, 1804.77, 1736.56, 1201.36, 352.745)$

Visually, this may be represented as shown in Supplement 9.

This vector was then embedded onto the time axis using the dates of the symptomatic periods, according to the data. The relative infectiousness is a function of the viral load, $g\left(V_i^S(t)\right)$, combined with the individuals' contacts, $C_i(t)$, and is presented in the Main text, Eq. (2).

If the number of contacts of an individual i was known, it was used to parameterize C_i. When the number of contacts was unknown, we sampled from a probability distribution, $f(C)$, constructed from the contact-tracing data: $C_i \sim f(C)$ (Supplement 2).

In each iteration of the stochastic simulation, we sampled values for $RR^{Late\ Phase}/RR^{Early\ Phase}$. Using Eq. (2) in Supplement 3, we calculated r, and from the relation $g(V^S(t)) = r^{Log(V^S(t))}$, we assigned each individual with their viral load contribution to the relative infectiousness. In the stochastic simulation, the variables were sampled from distributions (Main text, Table 1). A vector was assigned to each individual, and embedding the vectors in time yielded a matrix (Supplement 5).

Assigning transmission

Using the above matrix, each exposed case was probabilistically assigned with the contact from whom they were infected. In each day, the numbers of exposed and symptomatic individuals were in the same column in the matrix. Symptomatic individuals were considered potential infectors, and were probabilistically assigned as the source of transmission to their infected contact based on their relative infectiousness, $f_i^S(t)$ (see Supplement 10).

We used the relative infectiousness to determine probabilities of being the infector, normalizing over all possible infectors of that day (Main text, Equation (1)).

Testing intervention strategies

In every iteration the overall R_0 was calculated, and at the end of each of the simulations, the probability of disease elimination was calculated:

$$Pr(R_0 < 1) = \frac{N^{(R_0<1)}}{N}, \quad (1)$$

where $N^{(R_0<1)}$ denotes the number of iterations in which $R_0 < 1$ was obtained, and N is the overall number of iterations.

Quarantine of infected individuals

Two parameters were varied over this analysis: 1) number of days that have passed since symptoms onset and until recovery/death. 2) the coverage of population on which the policy is successfully implemented.

For every iteration, upon successful quarantine of individuals, all of the individuals' vectors were set to zero, and the R_0 was calculated (Main text, equation (3)). This procedure was repeated for each day of the timeframe considered (Supplement 7). To vary the coverage of quarantine, the corresponding proportion of the population's individuals' vectors were set to zero. This was repeated with different values of coverage, implemented by sampling a random number, $r \sim Uniform(0,1)$, and setting to zero the individuals' vector only if $r < C$, with C representing the tested coverage, varying from half the population ($C = 0.5$) to full coverage ($C = 1$). This stochastic process was conducted separately for the non-survivors (Main text, Figure 2A) and for all infected individuals (Supplement 7).

Reducing number of contacts

Reducing the contacts of the infected individuals was implemented by decreasing the probability of being selected as the infector. Eq. (1) in the Main text transforms into:

$$\hat{P}_{i,j}(t) = I_j(t)\frac{\gamma\, f_i^s(t)}{\gamma \sum_{k=1}^{n} f_k^s(t) + (1-\gamma)}, \qquad (2)$$

where γ is the proportion of contacts, $0 < \gamma < 1$, with $\gamma = 1$ implying no intervention, and $\gamma = 0$ is complete isolation of the individual from all contacts. This re-normalization lowers the chance of an individual to be chosen as the infector, which is effectively the same as reducing the number of contacts (Eq. (2), Main text).

Legends

Supplement 9. Disease Progression for an Infected Individual
Supplement 10. Assigning Transmission

Mathematica 9.0 code

```
Impact of Ebola disease progression on transmission and control in Liberia

Initialization
init[]:=
(
SetDirectory[NotebookDirectory[]];
days=DateRange[{2014,5,1},{2014,12,1}];
)(*init*)

Data
(*In "id" vector, "ext" represents cases from WHO. Numbers are coded cases collected by
LMoHSW*)

id={"ext1","ext2","ext76","ext4","ext5","ext82","ext11","ext12","ext13","ext14","ext88","ext14
8","ext15","ext89","ext16","ext17","ext91","ext18","ext92","ext149","ext19","ext93","ext20","e
xt94","ext150","ext21","ext95","ext151","ext22","ext96","ext23","ext152","ext153","ext154","ex
t24","ext27","ext102","ext28","ext103","ext155","ext29","ext30","ext31","ext106","ext156","ext
32","ext107","ext157","ext33","ext108","ext158","ext34","ext109","ext35","ext110","ext36","ext
111","ext37","ext112","ext38","ext113","ext39","ext114","ext40","ext115","ext159","ext41","ext
116","ext160","ext161","ext42","ext117","ext162","ext163","ext164","ext43","ext118","ext165","
ext166","ext167","ext44","ext119","ext168","ext169","ext170","ext45","ext120","ext171","ext46"
,"ext121","ext47","ext122","ext172","ext48","ext123","ext173","ext49","ext124","ext174","ext17
5","ext176","ext50","ext125","ext177","ext178","ext179","ext51","ext126","ext180","ext181","ex
t182","ext183","ext52","ext127","ext184","ext185","ext186","ext187","ext53","ext128","ext188",
"ext189","ext190","ext191","ext54","ext129","ext55","ext130","ext192","ext56","ext131","ext193
","ext194","ext195","ext196","ext197","ext198","ext199","ext57",311.`,"ext132","ext200","ext20
1","ext202","ext203","ext204","ext205","ext206","ext58",1.`,2.`,134.`,217.`,225.`,254.`,257.`,
265.`,307.`,136.`,203.`,219.`,297.`,298.`,86.`,230.`,250.`,251.`,256.`,271.`,306.`,259.`,264.`
,267.`,272.`,273.`,274.`,277.`,294.`,302.`,310.`,252.`,255.`,261.`,266.`,295.`,170.`,209.`,212
.`,253.`,263.`,269.`,282.`,283.`,284.`,285.`,286.`,289.`,290.`,194.`,206.`,226.`,228.`,233.`,2
34.`,237.`,262.`,268.`,270.`,288.`,2565.`,185.`,208.`,229.`,235.`,236.`,275.`,87.`,89.`,91.`,1
47.`,178.`,181.`,186.`,187.`,190.`,196.`,211.`,216.`,222.`,26.`,98.`,99.`,111.`,115.`,120.`,13
9.`,142.`,163.`,164.`,165.`,169.`,172.`,174.`,184.`,192.`,193.`,8.`,29.`,61.`,83.`,95.`,97.`,1
55.`,166.`,179.`,180.`,200.`,204.`,205.`,207.`,28.`,101.`,103.`,109.`,117.`,173.`,188.`,199.`,
201.`,202.`,77.`,104.`,116.`,145.`,148.`,157.`,158.`,168.`,171.`,175.`,177.`,182.`,183.`,195.`
,197.`,198.`,41.`,62.`,113.`,140.`,144.`,146.`,149.`,150.`,151.`,156.`,161.`,162.`,167.`,27.`,
31.`,44.`,106.`,121.`,125.`,128.`,130.`,131.`,132.`,133.`,138.`,143.`,1092.`,1093.`,22.`,47.`,
49.`,58.`,59.`,60.`,63.`,64.`,67.`,82.`,88.`,90.`,92.`,96.`,100.`,102.`,108.`,112.`,114.`,119.
`,127.`,141.`,7.`,42.`,43.`,46.`,52.`,54.`,56.`,57.`,70.`,71.`,74.`,75.`,78.`,84.`,85.`,93.`,1
05.`,107.`,110.`,118.`,159.`,2566.`,12.`,39.`,48.`,50.`,65.`,68.`,69.`,72.`,76.`,79.`,80.`,81.
`,11.`,13.`,21.`,33.`,34.`,45.`,55.`,73.`,9.`,10.`,14.`,15.`,16.`,17.`,18.`,19.`,24.`,25.`,32.
`,35.`,36.`,37.`,38.`,40.`,23.`,"ext133","ext507","ext508","ext509","ext510","ext511","ext512"
,"ext513","ext514","ext59","ext555","ext556","ext557","ext558","ext559","ext560","ext561","ext
562","ext563","ext134","ext515","ext516","ext517","ext518","ext519","ext520","ext521","ext522"
,"ext60","ext564","ext565","ext566","ext567","ext568","ext569","ext570","ext571","ext572","ext
135","ext523","ext524","ext525","ext526","ext527","ext528","ext529","ext530","ext61","ext573",
"ext574","ext575","ext576","ext577","ext578","ext579","ext580","ext581","ext136","ext531","ext
532","ext533","ext534","ext535","ext536","ext537","ext538","ext62","ext582","ext583","ext584",
"ext585","ext586","ext587","ext588","ext589","ext590","ext591","ext137","ext539","ext540","ext
541","ext542","ext543","ext544","ext545","ext546","ext63","ext592","ext593","ext594","ext595",
"ext596","ext597","ext598","ext599","ext600","ext601","ext138","ext547","ext548","ext549","ext
550","ext551","ext552","ext553","ext554","ext64","ext602","ext603","ext604","ext605","ext606",
"ext607","ext608","ext609","ext610","ext611","ext139","ext207","ext208","ext209","ext210","ext
211","ext212","ext213","ext214","ext215","ext216","ext217","ext218","ext219","ext220","ext221"
,"ext222","ext223","ext224","ext225","ext226","ext227","ext228","ext229","ext230","ext231","ex
t232","ext233","ext234","ext235","ext236","ext237","ext238","ext239","ext65","ext240","ext241"
,"ext242","ext243","ext244","ext245","ext246","ext66","ext247","ext248","ext249","ext250","ext
251","ext252","ext253","ext254","ext255","ext256","ext67","ext257","ext258","ext259","ext260",
"ext261","ext262","ext263","ext264","ext265","ext266","ext267","ext268","ext269","ext270","ext
271","ext272","ext68","ext143","ext273","ext274","ext275","ext276","ext277","ext278","ext279",
"ext280","ext281","ext282","ext283","ext284","ext285","ext286","ext287","ext288","ext289","ext
290","ext69","ext291","ext292","ext293","ext294","ext70","ext145","ext295","ext296","ext297","
ext298","ext299","ext300","ext301","ext302","ext303","ext304","ext305","ext306","ext307","ext3
08","ext309","ext310","ext311","ext312","ext313","ext314","ext315","ext316","ext317","ext318",
"ext319","ext320","ext321","ext322","ext323","ext324","ext325","ext326","ext327","ext328","ext
329","ext330","ext331","ext332","ext333","ext334","ext335","ext336","ext337","ext338","ext339"
,"ext340","ext341","ext342","ext343","ext344","ext345","ext346","ext347","ext348","ext349","ex
t350","ext351","ext352","ext353","ext354","ext355","ext356","ext357","ext358","ext359","ext360
","ext361","ext362","ext71","ext146","ext363","ext364","ext365","ext366","ext367","ext368","ex
t369","ext370","ext371","ext372","ext373","ext374","ext375","ext376","ext377","ext378","ext379
```

","ext380","ext381","ext382","ext383","ext384","ext385","ext386","ext387","ext388","ext389","e
xt390","ext391","ext392","ext393","ext394","ext395","ext396","ext397","ext398","ext399","ext40
0","ext401","ext402","ext403","ext404","ext405","ext406","ext407","ext408","ext409","ext410","
ext411","ext412","ext413","ext414","ext415","ext416","ext417","ext418","ext419","ext420","ext4
21","ext422","ext423","ext424","ext425","ext426","ext427","ext428","ext429","ext430","ext431",
"ext432","ext72","ext147","ext433","ext434","ext435","ext436","ext437","ext438","ext439","ext4
40","ext441","ext442","ext443","ext444","ext445","ext446","ext447","ext448","ext449","ext450",
"ext451","ext452","ext453","ext454","ext455","ext456","ext457","ext458","ext459","ext460","ext
461","ext462","ext463","ext464","ext465","ext466","ext467","ext468","ext469","ext470","ext471"
,"ext472","ext473","ext474","ext475","ext476","ext477","ext478","ext479","ext480","ext481","ex
t482","ext483","ext484","ext485","ext486","ext487","ext488","ext489","ext490","ext491","ext492
","ext493","ext494","ext495","ext496","ext497","ext498","ext499","ext500","ext501","ext502","e
xt503","ext504","ext505","ext506","ext73","ext612","ext613","ext614","ext615","ext616","ext617
","ext618","ext619","ext620","ext621","ext622","ext623","ext624","ext625","ext626","ext627","e
xt628","ext629","ext630","ext631","ext632","ext633","ext634","ext635","ext636","ext637","ext63
8","ext639","ext640","ext641","ext642","ext643","ext644","ext645","ext646","ext647","ext648","
ext649","ext650","ext651","ext652","ext653","ext654","ext655","ext656","ext657","ext658","ext6
59","ext660","ext661","ext662","ext663","ext664","ext665","ext666","ext667","ext668","ext669",
"ext670","ext671","ext672","ext673","ext674","ext675","ext676","ext677","ext678","ext679","ext
680","ext681","ext682","ext683","ext684","ext685","ext686","ext687","ext688","ext689","ext690"
,"ext691","ext692","ext693","ext694","ext695","ext696","ext697","ext698","ext699","ext700","ex
t701","ext702","ext703","ext704","ext705","ext706","ext707","ext708","ext709","ext710","ext711
","ext712","ext713","ext714","ext715","ext716","ext717","ext718","ext719","ext720","ext721","e
xt722","ext723","ext724","ext725","ext726","ext727","ext728","ext729","ext730","ext731","ext73
2","ext733","ext734","ext735","ext736","ext737","ext738","ext739","ext740","ext741","ext742","
ext743","ext744","ext745","ext746","ext747","ext748","ext749","ext750","ext751","ext752","ext7
53","ext754","ext755","ext756","ext757","ext758","ext759","ext760","ext761","ext762","ext763",
"ext764","ext765","ext766","ext767","ext768","ext769","ext770","ext771","ext772","ext773","ext
774","ext775","ext776","ext777","ext778","ext779","ext780","ext781","ext782","ext783","ext784"
,"ext785","ext786","ext787","ext788","ext789","ext790","ext791","ext792","ext793","ext794","ex
t795","ext796","ext797","ext798","ext799","ext800","ext801","ext802","ext803","ext804","ext805
","ext806","ext807","ext808","ext809","ext810","ext811","ext812","ext813","ext814","ext815","e
xt816","ext817","ext818","ext819","ext820","ext821","ext822","ext823","ext824","ext825","ext82
6","ext827","ext828","ext829","ext830","ext831","ext832","ext833","ext834","ext835","ext836","
ext837","ext838","ext839","ext840","ext841","ext842","ext843","ext844","ext845","ext846","ext8
47","ext848","ext849","ext850","ext851","ext852","ext853","ext854","ext855","ext856","ext857",
"ext858","ext859","ext860","ext861","ext862","ext863","ext864","ext865","ext866","ext867","ext
868","ext869","ext870","ext871","ext872","ext873","ext874","ext875","ext876","ext877","ext878"
,"ext879","ext880","ext881","ext882","ext883","ext884","ext885","ext886","ext887","ext888","ex
t889","ext890","ext891","ext892","ext893","ext894","ext895","ext896","ext897","ext898","ext899
","ext900","ext901","ext902","ext903","ext904","ext905","ext906","ext907","ext908","ext909","e
xt910","ext911","ext912","ext913","ext914","ext915","ext916","ext917","ext918","ext919","ext92
0","ext921","ext922","ext923","ext924","ext925","ext926","ext927","ext928","ext929","ext930","
ext931","ext932","ext933","ext934","ext935","ext936","ext937","ext938","ext939","ext940","ext9
41","ext942","ext943","ext944","ext945","ext946","ext947","ext948","ext949","ext950","ext951",
"ext952","ext953","ext954","ext955","ext956","ext957","ext958","ext959","ext960","ext961","ext
962","ext963","ext964","ext965","ext966","ext967","ext968","ext969","ext970","ext971","ext972"
,"ext973","ext974","ext975","ext976","ext977","ext978","ext979","ext980","ext981","ext982","ex
t983","ext984","ext985","ext986","ext987"};

status={"Dead","Dead","Alive","Dead","Dead","Alive","Dead","Dead","Dead","Dead","Alive","Dead"
,"Dead","Alive","Dead","Dead","Alive","Dead","Alive","Dead","Dead","Alive","Dead","Alive","Dea
d","Dead","Alive","Dead","Dead","Alive","Dead","Dead","Dead","Dead","Dead","Alive","Dead","Dea
d","Alive","Dead","Dead","Dead","Dead","Alive","Dead","Dead","Alive","Dead","Dead","Alive","De
ad","Dead","Alive","Dead","Alive","Dead","Alive","Dead","Alive","Dead","Alive","Dead","Alive",
"Dead","Alive","Dead","Dead","Alive","Dead","Dead","Dead","Alive","Alive","Dead","Dead","Dead"
,"Alive","Alive","Dead","Dead","Dead","Alive","Alive","Dead","Dead","Dead","Alive","Dead","Dea
d","Alive","Dead","Alive","Dead","Dead","Alive","Dead","Dead","Alive","Alive","Dead","Dead","D
ead","Alive","Alive","Dead","Dead","Dead","Alive","Alive","Alive","Dead","Dead","Dead","Alive"
,"Alive","Alive","Dead","Dead","Dead","Alive","Alive","Alive","Dead","Dead","Dead","Alive","De
ad","Alive","Dead","Dead","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","
Dead","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Dead"
,"Dead","Alive","Alive","Alive","Alive","Dead","Alive","Dead","Alive","Dead","Dead","Dead","Al
ive","Dead","Dead","Dead","Dead","Alive","Dead","Dead","Dead","Dead","Dead","Alive","Alive","A
live","Dead","Dead","Dead","Alive","Dead","Alive","Alive","Alive","Alive","Dead","Alive","Aliv
e","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Alive","Aliv
e","Alive","Dead","Alive","Alive","Dead","Dead","Alive","Alive","Dead","Alive","Dead","Alive",
"Alive","Alive","Dead","Alive","Alive","Dead","Dead","Alive","Alive","Dead","Alive","Alive","A
live","Dead","Alive","Dead","Dead","Dead","Alive","Alive","Dead","Alive","Dead","Dead","Alive"
,"Alive","Alive","Dead","Dead","Dead","Alive","Dead","Alive","Dead","Dead","Dead","Alive","Ali
ve","Alive","Alive","Dead","Alive","Dead","Alive","Dead","Alive","Dead","Dead","Alive","Alive"
,"Dead","Alive","Dead","Dead","Dead","Dead","Alive","Alive","Dead","Dead","Dead","Alive","Aliv
e","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Alive","Alive","Dead","Dead","
Dead","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Dead","Alive","Dead","Alive","Ali
ve","Alive","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Dead","Alive","Alive","Aliv
e","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","A
live","Alive","Dead","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","

```
Alive","Alive","Alive","Dead","Alive","Alive","Alive","Alive","Dead","Alive","Alive","Alive","
Dead","Alive","Dead","Alive","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","D
ead","Alive","Dead","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Alive","Dead
","Alive","Dead","Dead","Dead","Alive","Alive","Dead","Alive","Alive","Alive","Dead","Dead","D
ead","Dead","Dead","Dead","Dead","Alive","Dead","Alive","Alive","Alive","Alive","Alive","Alive
","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead
","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Dead"
,"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Ali
ve","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","De
ad","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive","Alive","Alive","Al
ive","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","
Dead","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","D
ead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Al
ive","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","
Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","D
ead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead
","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","De
ad","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Aliv
e","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Al
ive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","
Alive","Alive","Alive","Alive","Alive","Alive","Dead","Alive","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","De
ad","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive","Alive","Alive","Alive","
Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive"
,"Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Ali
ve","Alive","Alive","Dead","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","De
ad","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead"
,"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","D
ead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","A
live","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive",
"Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive
","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead
","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","
Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Aliv
e","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Al
ive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","
Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive","Alive",
"Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive
","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Ali
ve","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","D
ead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead
","Dead","Dead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive"
,"Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Aliv
e","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Al
ive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","
Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Dead","Dead","
Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive
e","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Al
ive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Dead","Dead","Dead","Dead
","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Alive","Alive","Alive","Alive","A
live","Alive","Alive","Alive","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","
Dead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","
Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dea
d","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead","Dead",
"Dead","Dead","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive",
"Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive","Alive
","Alive","Alive"};

numContacts={1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1
.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`,1.`
,1.`,1.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2
.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,2.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`
,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3.`,3
.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`,4.`
,4.`,4.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5.`,5
```

```
.`,5.`,5.`,6.`,6.`,6.`,6.`,6.`,6.`,6.`,6.`,6.`,6.`,7.`,7.`,7.`,7.`,7.`,7.`,7.`,7.`,7.`,7.`,7.`
,7.`,7.`,7.`,8.`,8.`,8.`,8.`,8.`,8.`,8.`,8.`,8.`,9.`,9.`,9.`,9.`,9.`,10.`,10.`,10.`,10.`,10.`,
10.`,10.`,10.`,10.`,10.`,11.`,11.`,11.`,11.`,11.`,11.`,11.`,11.`,12.`,12.`,12.`,12.`,12.`
,12.`,12.`,13.`,13.`,13.`,14.`,14.`,16.`,18.`,18.`,18.`,19.`,22.`,23.`,26.`,27.`,28.`,31.`,34.
`};

infectionDate={{2014,6,9},{2014,6,10},{2014,6,10},{2014,6,12},{2014,6,13},{2014,6,16},{2014,6,
19},{2014,6,20},{2014,6,21},{2014,6,22},{2014,6,22},{2014,6,23},{2014,6,23},{2014,6,23},{2014,
6,24},{2014,6,25},{2014,6,25},{2014,6,26},{2014,6,26},{2014,6,27},{2014,6,27},{2014,6,27},{201
4,6,28},{2014,6,28},{2014,6,29},{2014,6,29},{2014,6,29},{2014,6,30},{2014,6,30},{2014,6,30},{2
014,7,1},{2014,7,2},{2014,7,2},{2014,7,2},{2014,7,2},{2014,7,5},{2014,7,6},{2014,7,6},{2014,7,
7},{2014,7,7},{2014,7,7},{2014,7,8},{2014,7,9},{2014,7,10},{2014,7,10},{2014,7,10},{2014,7,11}
,{2014,7,11},{2014,7,11},{2014,7,12},{2014,7,12},{2014,7,12},{2014,7,13},{2014,7,13},{2014,7,1
4},{2014,7,14},{2014,7,15},{2014,7,15},{2014,7,16},{2014,7,16},{2014,7,17},{2014,7,17},{2014,7
,18},{2014,7,18},{2014,7,19},{2014,7,19},{2014,7,19},{2014,7,20},{2014,7,20},{2014,7,20},{2014
,7,20},{2014,7,21},{2014,7,21},{2014,7,21},{2014,7,21},{2014,7,21},{2014,7,21},{2014,7,22},{201
4,7,22},{2014,7,22},{2014,7,22},{2014,7,23},{2014,7,23},{2014,7,23},{2014,7,23},{2014,7,23},{
2014,7,24},{2014,7,24},{2014,7,24},{2014,7,25},{2014,7,25},{2014,7,26},{2014,7,26},{2014,7,26}
,{2014,7,27},{2014,7,27},{2014,7,27},{2014,7,28},{2014,7,28},{2014,7,28},{2014,7,28},{2014,7,2
8},{2014,7,29},{2014,7,29},{2014,7,29},{2014,7,29},{2014,7,29},{2014,7,30},{2014,7,30},{2014,7
,30},{2014,7,30},{2014,7,30},{2014,7,30},{2014,7,31},{2014,7,31},{2014,7,31},{2014,7,31},{2014
,7,31},{2014,7,31},{2014,8,1},{2014,8,1},{2014,8,1},{2014,8,1},{2014,8,1},{2014,8,1},{2014,8,2
},{2014,8,2},{2014,8,3},{2014,8,3},{2014,8,3},{2014,8,4},{2014,8,4},{2014,8,4},{2014,8,4},{201
4,8,4},{2014,8,4},{2014,8,4},{2014,8,4},{2014,8,4},{2014,8,5},{2014,8,5},{2014,8,5},{2014,8,5}
,{2014,8,5},{2014,8,5},{2014,8,5},{2014,8,5},{2014,8,5},{2014,8,5},{2014,8,6},{2014,8,6},{2014
,8,6},{2014,8,6},{2014,8,6},{2014,8,6},{2014,8,6},{2014,8,6},{2014,8,7},{2014,8,7},
{2014,8,7},{2014,8,7},{2014,8,7},{2014,8,8},{2014,8,8},{2014,8,8},{2014,8,8},{2014,8,8},{2014,
8,8},{2014,8,8},{2014,8,9},{2014,8,9},{2014,8,9},{2014,8,9},{2014,8,9},{2014,8,9},{2014,8,9},{
2014,8,9},{2014,8,9},{2014,8,9},{2014,8,10},{2014,8,10},{2014,8,10},{2014,8,10},{2014,8,10},{2
014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},
{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,11},{2014,8,12},{2014,8,12},{2014,8,12
},{2014,8,12},{2014,8,12},{2014,8,12},{2014,8,12},{2014,8,12},{2014,8,12},{2014,8,12},{2014,8,
12},{2014,8,12},{2014,8,13},{2014,8,13},{2014,8,13},{2014,8,13},{2014,8,13},{2014,8,13},{2014,
8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{201
4,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,14},{2014,8,15},{2014,8,15},{2014,8,15},{2
014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},
{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,15},{2014,8,16},{2014,8,16
},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,
16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,16},{2014,8,17},{2014,8,17},{2014,
8,17},{2014,8,17},{2014,8,17},{2014,8,17},{2014,8,17},{2014,8,17},{2014,8,17},{2014,8,18},{201
4,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2
014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,18},{2014,8,19},
{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19
},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,19},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,
20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,
8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,20},{2014,8,21},{2014,8,21},{2014,8,21},{201
4,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2
014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},{2014,8,21},
{2014,8,21},{2014,8,21},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22
},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,
22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,8,22},{2014,
8,22},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{201
4,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,23},{2014,8,24},{2014,8,24},{2014,8,24},{2
014,8,24},{2014,8,24},{2014,8,24},{2014,8,24},{2014,8,24},{2014,8,25},{2014,8,25},{2014,8,25},
{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25
},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,25},{2014,8,26},{2014,8,26},{2014,8,
26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,
8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{2014,8,26},{201
4,8,26},{2014,8,26},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2
014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},
{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,27},{2014,8,28},{2014,8,28},{2014,8,28
},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,
28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,8,28},{2014,
8,28},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{201
4,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2
014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,29},{2014,8,30},{2014,8,30},{2014,8,30},
{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30
},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,30},{2014,8,
30},{2014,8,30},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,
8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{201
4,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,8,31},{2014,9,1},{2014,9,1},{201
4,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1}
,{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014
,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},
{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,1},{2014,9,2},{2014,9,2},{2014,9,2},{2014,
9,2},{2014,9,2},{2014,9,2},{2014,9,2},{2014,9,2},{2014,9,2},{2014,9,2},{2014,9,3},{2014,9,3},{
2014,9,3},{2014,9,3},{2014,9,3},{2014,9,3},{2014,9,3},{2014,9,3},{2014,9,3},{2014,9,3},{2014,9
```

,3},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2
014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,4},{2014,9,
4},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{20
14,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,5
},{2014,9,5},{2014,9,5},{2014,9,5},{2014,9,6},{2014,9,6},{2014,9,6},{2014,9,6},{2014,9,6},{201
4,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7}
,{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014
,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},
{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,
9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{
2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9
,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2
014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,7},{2014,9,
7},{2014,9,7},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{20
14,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8
},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{201
4,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8}
,{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014
,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},
{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,
9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{
2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,8},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9
,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2
014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,
9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{20
14,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9
},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{201
4,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9}
,{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014
,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},
{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,9},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2
014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},
{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10
},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,
10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,
9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{201
4,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2
014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},
{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10
},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,10},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,
11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,
9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{201
4,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2
014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},
{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11
},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,
11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,
9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{2014,9,11},{201
4,9,11},{2014,9,11},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2
014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},
{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12
},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,
12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,
9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{2014,9,12},{201
4,9,12},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2
014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},
{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13
},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,
13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,
9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{2014,9,13},{201
4,9,13},{2014,9,13},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2
014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},
{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14
},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,
14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,
9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{2014,9,14},{201
4,9,14},{2014,9,14},{2014,9,14},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2
014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},
{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15},{2014,9,15
},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,
16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,
9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,16},{2014,9,17},{2014,9,17},{201
4,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2
014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},
{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17
},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,
17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,

```
9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{2014,9,17},{201
4,9,17},{2014,9,17},{2014,9,17},{2014,9,17}};

Viral loads - RNA copies
fatalMean={4.5,7.2,7.9,8.3,8.3,8.5,8.2,8.2,7.4,5.9,6.8,5.7,4.9,4};
nonFatalMean={5.2,4.8,6.5,6.6,6.1,6.2,5.2,5.8,5.1,4.1,4.6,4.9,4,4};

fatalSD={0.7,0.8,0.6,0.2,0.3,0.2,0.4,0.3,0.6,0.9,0.9,0.4,0.4,0.4};
nonFatalSD={0.4,1,0.4,0.6,0.4,0.5,0.8,0.7,0.7,0.2,0.1,0.1,0.1,0.1};

Constructing "disease vector"
constructSickVector[]:=
(
scaledViralLoadDead=viralLoadDead[[1;;sickLengthDead]];
scaledViralLoadAlive=viralLoadAlive[[1;;sickLengthAlive]];

contactCalibrationDead=Table[1,{i,1,sickLengthDead}];
contactCalibrationDead[[1;;sickLengthDead-stageCutoff]]=contactFirstStageDead;
contactCalibrationDead[[sickLengthDead-stageCutoff+1;;sickLengthDead]]=contactSecondStageDead;

contactCalibrationAlive=Table[1,{i,1,sickLengthAlive}];
contactCalibrationAlive[[1;;sickLengthDead-stageCutoff-1]]=contactFirstStageAlive;
contactCalibrationAlive[[sickLengthDead-stageCutoff;;sickLengthDead]]=contactSecondStageAlive;

sickVectorDead=Join[{-
1},Table[0,{i,1,incubationDead}],scaledViralLoadDead*contactCalibrationDead];
sickVectorAlive=Join[{-
1},Table[0,{i,1,incubationAlive}],scaledViralLoadAlive*contactCalibrationAlive];
)(*constructSickVector*)

Constructing matrix
constructMatrix[]:=
(
ebolaMatrix =
Table[0,{i,1,Length[infectionDate]},{j,1,Length[days]+maxIncubationPeriod+maxSickPeriod+1}];
(*Initalize zero matrix*)
symptomDateIndex=Table[0,{i,1,Length[infectionDate]}]; (*For later use*)
For[i=1,i<=Length[infectionDate],i++,
index=Position[days,infectionDate[[i]]][[1,1]];
If[status[[i]]=="Dead",
ebolaMatrix[[i]][[index+2-
Length[sickVectorsForMatrix[[i]]];;index+1]]=sickVectorsForMatrix[[i]];
symptomDateIndex[[i]]=index+2-sickLengthForMatrix[[i]],
ebolaMatrix[[i]][[index-2-incubationAliveForMatrix[[i]];;index-
2+sickLengthForMatrix[[i]]]]=sickVectorsForMatrix[[i]];
symptomDateIndex[[i]]=index-1;
];
];

infectedByDay=Table[0,{i,1,Length[days]}]; (*Overall infected every day*)
For[j=1,j<=Length[days],j++,
infectedByDay[[j]]=-1*Total[Cases[Transpose[ebolaMatrix][[j]],x_/;x<0]];
];

rMatrixTEMP=ebolaMatrix; (*Zeroizing negative elements*)
For[i=1,i<=Length[ebolaMatrix],i++,
For[j=1,j<=Length[ebolaMatrix[[1]]],j++,
If[ebolaMatrix[[i,j]]<0,
rMatrixTEMP[[i,j]]=0
];
];
];

rMatrix= Table[0,{i,1,Length[infectionDate]},{j,1,Length[days]}]; (*Initalize zero matrix*)
For[j=1,j<=Length[days],j++,
If[Count[Transpose[rMatrixTEMP][[j]],x_/;x>0]==0,
Return,
If[coverage!= 0&&RandomReal[]<coverage,
probSelected=RandomChoice[Join[{(1-
propContacts)},propContacts/Total[Transpose[rMatrixTEMP][[j]]]*Transpose[rMatrixTEMP][[j]]]-
>Join[{-1},Range[Length[Transpose[rMatrixTEMP][[j]]]],infectedByDay[[j]]],
probSelected=RandomChoice[Transpose[rMatrixTEMP][[j]]-
>Range[Length[Transpose[rMatrixTEMP][[j]]]],infectedByDay[[j]]];
];
For[k=1,k<=Length[probSelected],k++,
If[probSelected[[k]]==-1,
Return,
```

```
rMatrix[[probSelected[[k]],j]]+=1
];
];
];
];

cropTop=1;
cropBottom=Length[ebolaMatrix];(*Counted from top*)
cropLeft=Min[Table[Position[ebolaMatrix[[row]],-1],{row,1,Length[id]}]];
cropRight=Max[Table[Position[ebolaMatrix[[row]],-1],{row,1,Length[id]}]];(*Counted from left*)

rMatrixCrop=rMatrix[[cropTop;;cropBottom,cropLeft;;cropRight]];

rPerDayAll={};
rPerDayDead={};
rPerDayAlive={};
rDead={};
rAlive={};

For[i=cropTop,i<=cropBottom,i++,
If[cropLeft<=
symptomDateIndex[[i]]&&symptomDateIndex[[i]]+sickLengthForMatrix[[i]]<=cropRight,
rPerDayAll=Insert[rPerDayAll,rMatrix[[i,symptomDateIndex[[i]];;symptomDateIndex[[i]]+sickLengt
hForMatrix[[i]]-1]],-1];
If[status[[i]]=="Dead",
rPerDayDead=Insert[rPerDayDead,rMatrix[[i,symptomDateIndex[[i]];;symptomDateIndex[[i]]+sickLen
gthForMatrix[[i]]-1]],-1];
rDead=Join[rDead,{Total[rMatrix[[i,symptomDateIndex[[i]];;symptomDateIndex[[i]]+sickLengthForM
atrix[[i]]-1]]]}],
rPerDayAlive=Insert[rPerDayAlive,rMatrix[[i,symptomDateIndex[[i]];;symptomDateIndex[[i]]+sickL
engthForMatrix[[i]]-1]],-1];
rAlive=Join[rAlive,{Total[rMatrix[[i,symptomDateIndex[[i]];;symptomDateIndex[[i]]+sickLengthFo
rMatrix[[i]]-1]]]}]
];
];
];

rAll=Join[rAlive,rDead];
rPerDayAll=Flatten[rPerDayAll,{2}];
rPerDayDead=Flatten[rPerDayDead,{2}];
rPerDayAlive=Flatten[rPerDayAlive,{2}];
)(*constructMatrix*)
Constructing "visual" matrix
constructVisualMatrix[]:=
(
visualEbolaMatrixTEMP=ebolaMatrix[[cropTop;;cropBottom,cropLeft;;cropRight+maxIncubationPeriod
+maxSickPeriod]];

(*indexOfPeopleInfected=Table[0,{day,1,Length[visualEbolaMatrixTEMP[[1]]]}];*)
visualEbolaMatrix=Table[0,{day,1,Length[visualEbolaMatrixTEMP[[1]]]}];

For[day=1,day<=Length[visualEbolaMatrixTEMP[[1]]],day++,
indexOfPeopleInfected=Flatten[Position[Transpose[visualEbolaMatrixTEMP][[day]],-1]];
visualEbolaMatrix[[day]]=Total[visualEbolaMatrixTEMP[[indexOfPeopleInfected]]];
If[visualEbolaMatrix[[day]]==0,
visualEbolaMatrix[[day]]=Table[0,{i,1,Length[visualEbolaMatrixTEMP[[1]]]}];
];
];

cutBottom=Length[visualEbolaMatrix];
While[visualEbolaMatrix[[cutBottom]]== Table[0,{r,1,Length[visualEbolaMatrix[[cutBottom]]]}],
cutBottom-=1
];

cutRight=Length[Transpose[visualEbolaMatrix]];
While[Transpose[visualEbolaMatrix][[cutRight]]==
Table[0,{r,1,Length[Transpose[visualEbolaMatrix][[cutRight]]]}],
cutRight-=1
];

visualEbolaMatrixCropped=visualEbolaMatrix[[1;;cutBottom,1;;cutRight]];
plotVisualMatrix=MatrixPlot[visualEbolaMatrixCropped,ImageSize->Large];

totalVisualInfections=Table[0,{j,1,Length[Transpose[visualEbolaMatrixCropped]]}];
totalVisualViralLoad=Table[0,{j,1,Length[Transpose[visualEbolaMatrixCropped]]}];

For[j=1,j<=Length[Transpose[visualEbolaMatrixCropped]],j++,
```

```
totalVisualInfections[[j]]=Total[Cases[Transpose[visualEbolaMatrixCropped][[j]],x_/;x<0]];
totalVisualViralLoad[[j]]=Total[Cases[Transpose[visualEbolaMatrixCropped][[j]],x_/;0<x]];
];

visualLines={totalVisualViralLoad,Table[0,{j,1,Length[Transpose[visualEbolaMatrixCropped]]}],t
otalVisualInfections};

plotVisualLines=MatrixPlot[visualLines,ImageSize->Large];
)(*constructVisualMatrix*)

Relative risk calibration
generateR[viralLoadIN_]:=
(
RcoefOUT=0;
While[RcoefOUT<=1,
Switch[viralLoadIN,
1,
meal=Max[1,Exp[RandomVariate[NormalDistribution[Log[1.2],0.415]]]];
RRmeal=Max[meal,Exp[RandomVariate[NormalDistribution[Log[2.2],0.31]]]]/meal;
RcoefOUT=RRmeal^(1/((52*Sum[fatalMean[[i]],{i,5,10}]/6+3*Sum[nonFatalMean[[i]],{i,5,10}]/6)/55
-((52*Sum[fatalMean[[i]],{i,1,4}]/4+3*Sum[nonFatalMean[[i]],{i,1,4}]/4)/55)));
2,
talk=Max[1,Exp[RandomVariate[NormalDistribution[Log[0.7],0.525]]]];
RRtalk=Max[talk,Exp[RandomVariate[NormalDistribution[Log[3.9],0.58]]]]/talk;
RcoefOUT=RRtalk^(1/((52*Sum[fatalMean[[i]],{i,5,10}]/6+3*Sum[nonFatalMean[[i]],{i,5,10}]/6)/55
-((52*Sum[fatalMean[[i]],{i,1,4}]/4+3*Sum[nonFatalMean[[i]],{i,1,4}]/4)/55)));
3,
bed=Max[1,Exp[RandomVariate[NormalDistribution[Log[1.3],0.34]]]];
RRbed=Max[bed,Exp[RandomVariate[NormalDistribution[Log[2.2],0.33]]]]/bed;
RcoefOUT=RRbed^(1/((52*Sum[fatalMean[[i]],{i,5,10}]/6+3*Sum[nonFatalMean[[i]],{i,5,10}]/6)/55-
((52*Sum[fatalMean[[i]],{i,1,4}]/4+3*Sum[nonFatalMean[[i]],{i,1,4}]/4)/55)));
];
];
RcoefOUT
)(*generateR*)

Analysis
Defining distributions
define[iterationsIN_]:=
(
numOfIterations=iterationsIN;
maxSickPeriod=14;
minSickPeriod=5;

maxIncubationPeriod=15;
minIncubationPeriod=5;

incubationDist=Round[RandomVariate[TriangularDistribution[{minIncubationPeriod-
0.5,maxIncubationPeriod+0.5},8],{numOfIterations,Length[id]}]];
sickPeriodDist=Round[RandomVariate[TriangularDistribution[{minSickPeriod-
0.5,maxSickPeriod+0.5},8],{numOfIterations,Length[id]}]];

contactFirstStageDist=Round[RandomVariate[HistogramDistribution[numContacts],{numOfIterations,
Length[id]}]];
contactSecondStageDist=Round[RandomVariate[TruncatedDistribution[{0.5,5.5},HistogramDistributi
on[numContacts]],{numOfIterations,Length[id]}]];
stageCutoffDist=Round[RandomVariate[UniformDistribution[{0.5,5.5}],{numOfIterations,Length[id]
}]];

Rcoef=Table[generateR[viralLoad],{i,1,numOfIterations}];

viralLoadFatalDist=
Table[Table[Table[RandomVariate[TruncatedDistribution[{0,Infinity},NormalDistribution[fatalMea
n[[i]],fatalSD[[i]]]]],{i,1,maxSickPeriod}],{k,1,Length[id]}],{j,1,numOfIterations}];viralLoad
NonFatalDist=
Table[Table[Table[RandomVariate[TruncatedDistribution[{0,Infinity},NormalDistribution[nonFatal
Mean[[i]],nonFatalSD[[i]]]]],{i,1,maxSickPeriod}],{k,1,Length[id]}],{j,1,numOfIterations}];
)(*define*)

Single draw from distributions
singleDraw:=
(
Rcoef=generateR[viralLoad];
define[1];
coverage=0;
propContacts=1;
```

```
sickVectorsForMatrix=Table[0,{i,1,Length[id]}];
sickLengthForMatrix=Table[0,{i,1,Length[id]}];
incubationAliveForMatrix=Table[0,{i,1,Length[id]}];
iter=1;

For[person=1,person<=Length[id],person++,

incubationDead=incubationDist[[iter,person]];
incubationAlive=incubationDead;

sickLengthDead =sickPeriodDist[[iter,person]];
sickLengthAlive =sickLengthDead;

contactFirstStageDead=contactFirstStageDist[[iter,person]];
contactSecondStageDead=contactSecondStageDist[[iter,person]];

contactFirstStageAlive=contactFirstStageDist[[iter,person]];
contactSecondStageAlive=contactFirstStageAlive;

stageCutoff=stageCutoffDist[[iter,person]]; (*Days counted from end*)

viralLoadDead=Rcoef[[iter]]^viralLoadFatalDist[[iter,person]];
viralLoadAlive=Rcoef[[iter]]^viralLoadNonFatalDist[[iter,person]];

constructSickVector[];

If[status[[person]]=="Dead",
sickVectorsForMatrix[[person]]=sickVectorDead;
sickLengthForMatrix[[person]]=sickLengthDead,
sickVectorsForMatrix[[person]]=sickVectorAlive;
sickLengthForMatrix[[person]]=sickLengthAlive;
incubationAliveForMatrix[[person]]=incubationAlive
];
];

constructMatrix[];

rAllSingle=rAll;
rDeadSingle=rDead;
rAliveSingle=rAlive;
)(*singleDraw*)

Sensitivity analysis (Iterations, Coverage, Proportion of contacts)
sensAnal[iterationsIN_,coverageIN_,propContactsIN_]:=
(
coverage=coverageIN;
propContacts=propContactsIN;

define[iterationsIN];

rAllSensitivity=Table[0,{i,1,numOfIterations}];
rDeadSensitivity=Table[0,{i,1,numOfIterations}];
rAliveSensitivity=Table[0,{i,1,numOfIterations}];
rPerDayAllSensitivity=Table[0,{i,1,numOfIterations}];
rPerDayDeadSensitivity=Table[0,{i,1,numOfIterations}];
rPerDayAliveSensitivity=Table[0,{i,1,numOfIterations}];
personPickedIndex=Table[{},{i,1,numOfIterations}];
personPickedRVector=Table[{},{i,1,numOfIterations}];

For[iter=1,iter<=numOfIterations,iter++,

sickVectorsForMatrix=Table[0,{i,1,Length[id]}];
sickLengthForMatrix=Table[0,{i,1,Length[id]}];
incubationAliveForMatrix=Table[0,{i,1,Length[id]}];

tracedCount=1;

For[person=1,person<=Length[id],person++,

incubationDead=incubationDist[[iter,person]];
incubationAlive=incubationDead;

sickLengthDead =sickPeriodDist[[iter,person]];
sickLengthAlive =sickLengthDead;

If[NumberQ[id[[person]]],
```

```
contactFirstStageDead=numContacts[[tracedCount]];
contactSecondStageDead=Round[RandomVariate[TruncatedDistribution[{0.5,Min[(numContacts[[traced
Count]]+0.5),5.5]},HistogramDistribution[numContacts]]]];
contactFirstStageAlive=numContacts[[tracedCount]];
contactSecondStageAlive=numContacts[[tracedCount]];

tracedCount+=1
,
contactFirstStageDead=contactFirstStageDist[[iter,person]];
contactSecondStageDead=contactSecondStageDist[[iter,person]];
contactFirstStageAlive=contactFirstStageDist[[iter,person]];
contactSecondStageAlive=contactFirstStageAlive;
];

stageCutoff=stageCutoffDist[[iter,person]]; (*Days counted from end*)

viralLoadDead=Rcoef[[iter]]^viralLoadFatalDist[[iter,person]];
viralLoadAlive=Rcoef[[iter]]^viralLoadNonFatalDist[[iter,person]];

constructSickVector[];

If[status[[person]]=="Dead",
sickVectorsForMatrix[[person]]=sickVectorDead;
sickLengthForMatrix[[person]]=sickLengthDead,
sickVectorsForMatrix[[person]]=sickVectorAlive;
sickLengthForMatrix[[person]]=sickLengthAlive;
incubationAliveForMatrix[[person]]=incubationAlive
];
];

constructMatrix[];

rand=RandomInteger[{cropTop,cropBottom}];
While[Not[cropLeft<= symptomDateIndex[[rand]]&&(symptomDateIndex[[rand]]+sickLengthAlive-1)<=
cropRight],
rand=RandomInteger[{cropTop,cropBottom}];
];

personPickedIndex[[iter]]=rand;
If[status[[rand]]=="Dead",
personPickedRVector[[iter]]=rMatrix[[rand,symptomDateIndex[[rand]];;symptomDateIndex[[rand]]+s
ickLengthDead-1]],
personPickedRVector[[iter]]=rMatrix[[rand,symptomDateIndex[[rand]];;symptomDateIndex[[rand]]+s
ickLengthAlive-1]];
];

rAllSensitivity[[iter]]=rAll;
rDeadSensitivity[[iter]]=rDead;
rAliveSensitivity[[iter]]=rAlive;
rPerDayAllSensitivity[[iter]]=rPerDayAll;
rPerDayDeadSensitivity[[iter]]=rPerDayDead;
rPerDayAliveSensitivity[[iter]]=rPerDayAlive;
];
)(*sensAnal*)

Secondary cases distribution (one person per iteration)
secondaryCases[]:=
(
personPickedDeadRVector={};
personPickedAliveRVector={};

For[i=1,i<=Length[personPickedIndex],i++,
If[status[[personPickedIndex[[i]]]]=="Dead",
personPickedDeadRVector=Insert[personPickedDeadRVector,personPickedRVector[[i]],-1],
personPickedAliveRVector=Insert[personPickedAliveRVector,personPickedRVector[[i]],-1]
];
];

Print["Overall ",Length[personPickedIndex]," picked. "];
Print["Survived: ",Length[personPickedAliveRVector],", Deceased:
",Length[personPickedDeadRVector]];

plotPersonPickedBothRVectorHist=Histogram[{Table[Total[personPickedDeadRVector[[i]]],{i,1,Leng
th[personPickedDeadRVector]}],Table[Total[personPickedAliveRVector[[i]]],{i,1,Length[personPic
kedAliveRVector]}]},
```

```
{1},"Probability",ChartStyle->{Blue,Red},ChartLegends-
>Placed[{"Deceased","Survived"},Top],AxesOrigin->{0,0},PlotRange->{{0,15},{0,1}}];
plotPersonPickedDeadRVectorHist=Histogram[Table[Total[personPickedDeadRVector[[i]]],{i,1,Lengt
h[personPickedDeadRVector]}],{1},"Probability",ChartStyle->Blue,PlotLabel->"Secondary cases -
Deceased",AxesOrigin->

{0,0},PlotRange->{{0,15},{0,1}}];
plotPersonPickedAliveRVectorHist=Histogram[Table[Total[personPickedAliveRVector[[i]]],{i,1,Len
gth[personPickedAliveRVector]}],{1},"Probability",ChartStyle->Red,PlotLabel->"Secondary cases
- Survived",AxesOrigin->

{0,0},PlotRange->{{0,15},{0,1}}];
plotSinglePersonAllHist=Histogram[Table[Total[personPickedRVector[[i]]],{i,1,Length[personPick
edRVector]}],{1},"Probability",ChartStyle->Green,PlotLabel->"All",AxesOrigin->{0,0},PlotRange-
>{{0,15},{0,1}}];

plotHistogramsSecondary=GraphicsGrid[{{plotPersonPickedBothRVectorHist,
plotPersonPickedAliveRVectorHist,plotPersonPickedDeadRVectorHist,plotSinglePersonAllHist}},Ima
geSize->1200];
)(*secondaryCases*)

Secondary cases per day (one person per iteration)
secondaryCasesPerDay[]:=
(
personPickedDeadPerDay=Table[0,{i,1,Length[Flatten[personPickedDeadRVector,{2}]]}];
confIntDeadPerDay=Table[{0,0},{i,1,Length[Flatten[personPickedDeadRVector,{2}]]}];

For[i=1,i<=Length[personPickedDeadPerDay],i++,
personPickedDeadPerDay[[i]]=Mean[Flatten[personPickedDeadRVector,{2}][[i]]];
confIntDeadPerDay[[i,1]]=Sort[Flatten[personPickedDeadRVector,{2}][[i]]][[Round[0.975*Length[F
latten[personPickedDeadRVector,{2}][[i]]]]]];
confIntDeadPerDay[[i,2]]=Sort[Flatten[personPickedDeadRVector,{2}][[i]]][[Max[Round[0.025*Leng
th[Flatten[personPickedDeadRVector,{2}][[i]]],1]]];
];

personPickedAlivePerDay=Table[0,{i,1,Length[Flatten[personPickedAliveRVector,{2}]]}];
confIntAlivePerDay=Table[{0,0},{i,1,Length[Flatten[personPickedAliveRVector,{2}]]}];

For[i=1,i<=Length[personPickedAlivePerDay],i++,
personPickedAlivePerDay[[i]]=Mean[Flatten[personPickedAliveRVector,{2}][[i]]];
confIntAlivePerDay[[i,1]]=Sort[Flatten[personPickedAliveRVector,{2}][[i]]][[Round[0.975*Length
[Flatten[personPickedAliveRVector,{2}][[i]]]]]];
confIntAlivePerDay[[i,2]]=Sort[Flatten[personPickedAliveRVector,{2}][[i]]][[Max[Round[0.025*Le
ngth[Flatten[personPickedAliveRVector,{2}][[i]]],1]]];
];

Print["Secondary cases total - Deceased: ", N[Total[personPickedDeadPerDay]]];
Print["Secondary cases total - Survived: ", N[Total[personPickedAlivePerDay]]];

errorPerDayDeadHi=Table[confIntDeadPerDay[[i,1]],{i,1,Length[confIntDeadPerDay]}]-
personPickedDeadPerDay;
errorPerDayDeadLo=Table[confIntDeadPerDay[[i,2]],{i,1,Length[confIntDeadPerDay]}]-
personPickedDeadPerDay;

errorPerDayAliveHi=Table[confIntAlivePerDay[[i,1]],{i,1,Length[confIntAlivePerDay]}]-
personPickedAlivePerDay;
errorPerDayAliveLo=Table[confIntAlivePerDay[[i,2]],{i,1,Length[confIntAlivePerDay]}]-
personPickedAlivePerDay;

Needs["ErrorBarPlots`"];

confIntDead=Table[ErrorBar[{errorPerDayDeadLo[[i]],errorPerDayDeadHi[[i]]}],{i,1,Length[person
PickedDeadPerDay]}];
confIntAlive=Table[ErrorBar[{errorPerDayAliveLo[[i]],errorPerDayAliveHi[[i]]}],{i,1,Length[per
sonPickedAlivePerDay]}];

plotSecondaryPerDay=GraphicsGrid[{{ErrorListPlot[Table[{{i,personPickedAlivePerDay[[i]]},confI
ntAlive[[i]]},{i,1,Length[personPickedAlivePerDay]}],PlotLabel->"Secondary cases per day -
Survived",PlotRange->

{0,4},PlotStyle->Red,AxesOrigin-
>{0,0}],ErrorListPlot[Table[{{i,personPickedDeadPerDay[[i]]},confIntDead[[i]]},{i,1,Length[per
sonPickedDeadPerDay]}],PlotLabel->"Secondary cases per day - Deceased",PlotRange->

{0,4},PlotStyle->Blue,AxesOrigin->{0,0}]}},ImageSize->1000];
)(*secondaryCasesPerDay*)
```

```
Removing from all population
removeAll[]:=
(
rPerPersonAllSensitivity=Table[Flatten[rPerDayAllSensitivity[[i]],{2}],{i,1,Length[rPerDayAllS
ensitivity]}];

rUntilDayAll=Table[Table[0,{i,1,Length[rPerDayAllSensitivity]},{j,1,maxSickPeriod}],
{k,1,Length[coverageRemoveScenarios]}];

For[k=1,k<=Length[coverageRemoveScenarios],k++,
For[i=1,i<=Length[rPerPersonAllSensitivity],i++,
For[p=1,p<=Length[rPerPersonAllSensitivity[[i]]],p++,
rPerPersonAllSensitivity[[i]][[p]]=PadRight[rPerPersonAllSensitivity[[i]][[p]],maxSickPeriod,
{0}];
];
For[j=1,j<=maxSickPeriod,j++,
tempCopy=rPerPersonAllSensitivity[[i]];
For[p=1,p<=Length[rPerPersonAllSensitivity[[i]]],p++,
If[RandomReal[]<coverageRemoveScenarios[[k]],
tempCopy[[p]][[j;;Length[rPerPersonAllSensitivity[[i]][[p]]]]]=0;
If[j==maxSickPeriod,
tempCopy[[p]][[j]]=0;
];
];
];
rUntilDayAll[[k,i,j]]=Total[Mean[tempCopy]];
];
];
];

plotRemoveAll=GraphicsGrid[{Table[ListLinePlot[Table[Count[Flatten[rUntilDayAll[[k]],{2}][[j]]
,x_/;x<=1]/Length[Flatten[rUntilDayAll[[k]],{2}][[j]]],{j,1,maxSickPeriod}],
PlotRange->{0,1.1},AxesLabel->{"days"},PlotLabel->{"Removing from all, coverage:
",coverageRemoveScenarios[[k]]}],{k,1,Length[coverageRemoveScenarios]}]},ImageSize-> 1200];
)(*removeAll*)

Removing deceased from population
removeDead[]:=
(
rPerPersonDeadSensitivity=Table[Flatten[rPerDayDeadSensitivity[[i]],{2}],{i,1,Length[rPerDayDe
adSensitivity]}];
rPerPersonAliveSensitivity=Table[Flatten[rPerDayAliveSensitivity[[i]],{2}],{i,1,Length[rPerDay
AliveSensitivity]}];

rUntilDayDead=Table[Table[0,{i,1,Length[rPerDayDeadSensitivity]},{j,1,maxSickPeriod}],{k,1,Len
gth[coverageRemoveScenarios]}];

For[k=1,k<=Length[coverageRemoveScenarios],k++,
For[i=1,i<=Length[rPerPersonDeadSensitivity],i++,
For[p=1,p<=Length[rPerPersonDeadSensitivity[[i]]],p++,
rPerPersonDeadSensitivity[[i]][[p]]=PadRight[rPerPersonDeadSensitivity[[i]][[p]],maxSickPeriod
,{0}];
];
For[p=1,p<=Length[rPerPersonAliveSensitivity[[i]]],p++,
rPerPersonAliveSensitivity[[i]][[p]]=PadRight[rPerPersonAliveSensitivity[[i]][[p]],maxSickPeri
od,{0}];
];
For[j=1,j<=maxSickPeriod,j++,
tempCopy=rPerPersonDeadSensitivity[[i]];
For[p=1,p<=Length[rPerPersonDeadSensitivity[[i]]],p++,
If[RandomReal[]<coverageRemoveScenarios[[k]],
tempCopy[[p]][[j;;Length[rPerPersonDeadSensitivity[[i]][[p]]]]]=0;
If[j==maxSickPeriod,
tempCopy[[p]][[j]]=0;
];
];
];
tempSum=Length[tempCopy]+Length[rPerPersonAliveSensitivity[[i]]];
rUntilDayDead[[k,i,j]]=Total[Mean[tempCopy]]*Length[tempCopy]/tempSum+Total[Mean[rPerPersonAli
veSensitivity[[i]]]]*Length[rPerPersonAliveSensitivity[[i]]]/tempSum;
];
];
];

plotRemoveDead=GraphicsGrid[{Table[ListLinePlot[Table[Count[Flatten[rUntilDayDead[[k]],{2}][[j
]],x_/;x<=1]/Length[Flatten[rUntilDayDead[[k]],{2}][[j]]],{j,1,maxSickPeriod}],
```

```
PlotRange->{0,1.1},AxesLabel->{"days"},PlotLabel->{"Removing from deceased, coverage:
",coverageRemoveScenarios[[k]]}],{k,1,Length[coverageRemoveScenarios]}]},ImageSize-> 1200];
)(*removeDead*)

Reducing contacts (Iterations per scenario)
reduceContacts[iterUsed_]:=
(
probEradicate[]:=Count[Table[Mean[rAllSensitivity[[iii]]],{iii,1,Length[rAllSensitivity]}]
,x_/;x<=1]/Length[rAllSensitivity];

reduceContactsResults=Table[Table[0,{j,1,Length[contactScenarios]}],{i,1,Length[coverageContac
tsScenarios]}];

For[ii=1,ii<=Length[coverageContactsScenarios],ii++,
For[jj=1,jj<=Length[contactScenarios],jj++,
sensAnal[iterUsed,coverageContactsScenarios[[ii]],contactScenarios[[jj]]];
reduceContactsResults[[ii,jj]]=probEradicate[];
];
];

plotReduceContacts=Table[ListLinePlot[Transpose[{contactScenarios,reduceContactsResults[[ii]]}
],PlotRange->{0,1.1},AxesOrigin->{0,0},ImageSize->Medium,AxesLabel->{"% contacts"},
PlotLabel->{"Reducing contacts,coverage:
",coverageContactsScenarios[[ii]]}],{ii,1,Length[coverageContactsScenarios]}];
)(*reduceContacts*)
```