

## Model 3: Basic SIR with a covariate deathrate

Katie Montovan, Aaron King, Eli Goldwyn & Julie Blackwood

2025-1-22

This is a modification of the SIR model (model 2) to show how you can include a covariate. We bring in a file containing rates of death from the disease. We assume that the only death is happening from the illness. The covariate captures the hospitals getting better at treating the illness.

We assume that you have already read the chapter and worked through the “Model 1 SI” and “Model 2 SIR” r codes and understand them reasonably well.

#General Setup

```
options(  
  tidyverse.quiet=TRUE,  
  pomp_archive_dir="results",  
  stringsAsFactors=FALSE  
)  
library(tidyverse)  
library(pomp)  
#stopifnot(packageVersion("pomp")>="6.0.4")  
library(doParallel)  
library(doRNG)  
library(ggplot2)  
library(dplyr)  
  
library(readr) #this library is only used to load the csv file.  
simuldata <-read_csv("https://raw.githubusercontent.com/kmontovan/pomp/refs/heads/main/SimulatedPompiti")
```

##Import covariate table and make it into a pomp covariate table For this example we are going to assume that we have information on how the deathrate from the illness is changing as medical professionals learn how to treat the illness better. We also are adding death to our model (which was not included before).

The covar table could contain estimates for every day of our model, but even if it did, we would need a way to fill in for the Euler timesteps. So we set the **order** to either connect the points we have with a line (like you see in the graph), or **constant** which holds the value constant until the next time at which there is a specified value.

```
covardata <-read_csv("https://raw.githubusercontent.com/kmontovan/pomp/refs/heads/main/Covariate.csv")  
covardata
```

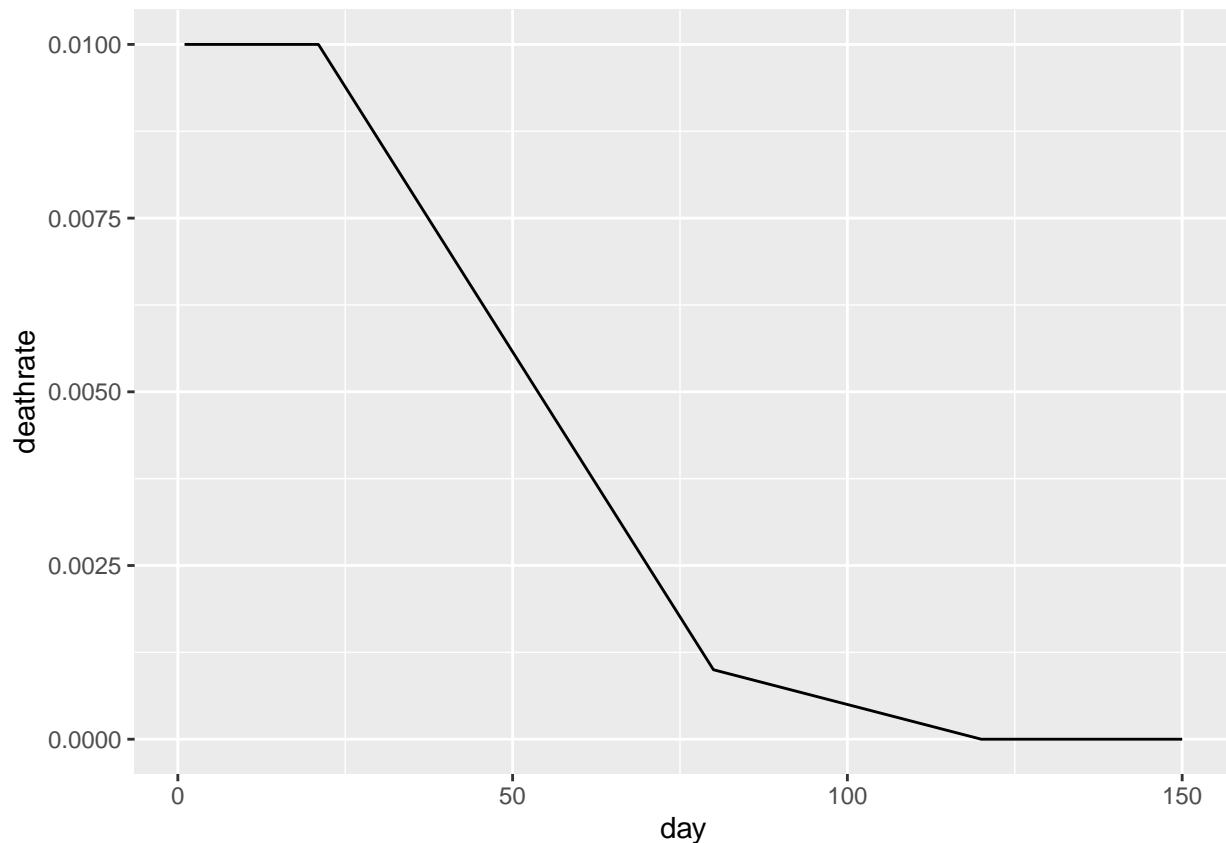
```
## # A tibble: 5 x 2  
##   day deathrate  
##   <dbl>     <dbl>  
## 1     1     0.01  
## 2    21     0.01  
## 3    80    0.001  
## 4   120     0  
## 5   150     0
```

```

#Save the covariate as a covariate_table:
covar=covariate_table(
  t=covardata$day,
  deathrate=covardata$deathrate,
  times="t",
  order = c("linear")
  #tells the program how to interpolate between data points.
  #options are linear or constant
)

ggplot(covardata,aes(x=day,y=deathrate))+
  geom_line()+
  ylim(c(0,.01))

```



We have set the deathrate to start high, then decrease to low, then decrease to zero. In reality, it probably would not reach zero, but setting it to zero makes it easier to see in the simulations. So for our example we include zero. You can play with other covariate values and see if you can find others that have clear effects on the model results.

## Model setup

You can now include the covariates included in your covariate table in the csnipits of your model. Here we add death to the model and use the covariate **deathrate** to control how many people die from the illness.

```

sir_step <- Csnippet("
  double infections = rbinom(S,1-exp(-Beta*I*dt));

```

```

double death = rbinom(I,1-exp(-deathrate*dt));
double recoveries = rbinom(I-death,1-exp(-gamma*dt));
double loseimmunity = rbinom(R,1-exp(-r*dt));
S += -infections + loseimmunity;
I += infections - recoveries-death;
R += recoveries - loseimmunity;
NewI += infections;
")

sir_rinit <- Csnippet("
  S = nearbyint((1-phi)*N);
  I = nearbyint(phi*N);
  R = 0;
  NewI = 0;
")

si_dmeas <- Csnippet("
  lik = dnbinom_mu(NewCases,k,rho*NewI,give_log);
")

si_rmeas <- Csnippet("
  NewCases = rnbinom_mu(k,rho*NewI);
")

```

```

SIR_Model <-
  pomp(
    data=simuldata, #the timeseries data
    times="day", #timesteps in the simuldata dataset
    t0=0, #the time to start the simulations.
    rprocess=euler(
      sir_step,
      delta.t=1/4 # steps of <= 6hr
    ),
    rinit=sir_rinit,
    rmeasure=si_rmeas,
    statenames=c("S","I","R","NewI"),
    accumvars="NewI",
    covar=covar,
    paramnames=c("N","phi","Beta","gamma","r","rho","k")
  )

```

```

## Warning: in 'pomp': the supplied covariate times do not embrace the data times:
## covariates may be extrapolated.

```

## Simulations

```

SIR_Model |>
  simulate(
    params=c(N=1000, #population size
             phi = 0.004, #number of infections at time 0
             Beta=0.0024, # rate of spread of illness - per year
             gamma=0.1, #recovery rate
             r = 0.01, #rate of individuals loosing immunity

```

```

    rho=0.8, #fraction of new infections that are recorded
    k=5 ), #changes the variability of reporting accuracy
  nsim=10,
  format="data.frame",
  include.data=TRUE
) -> sims

```

```

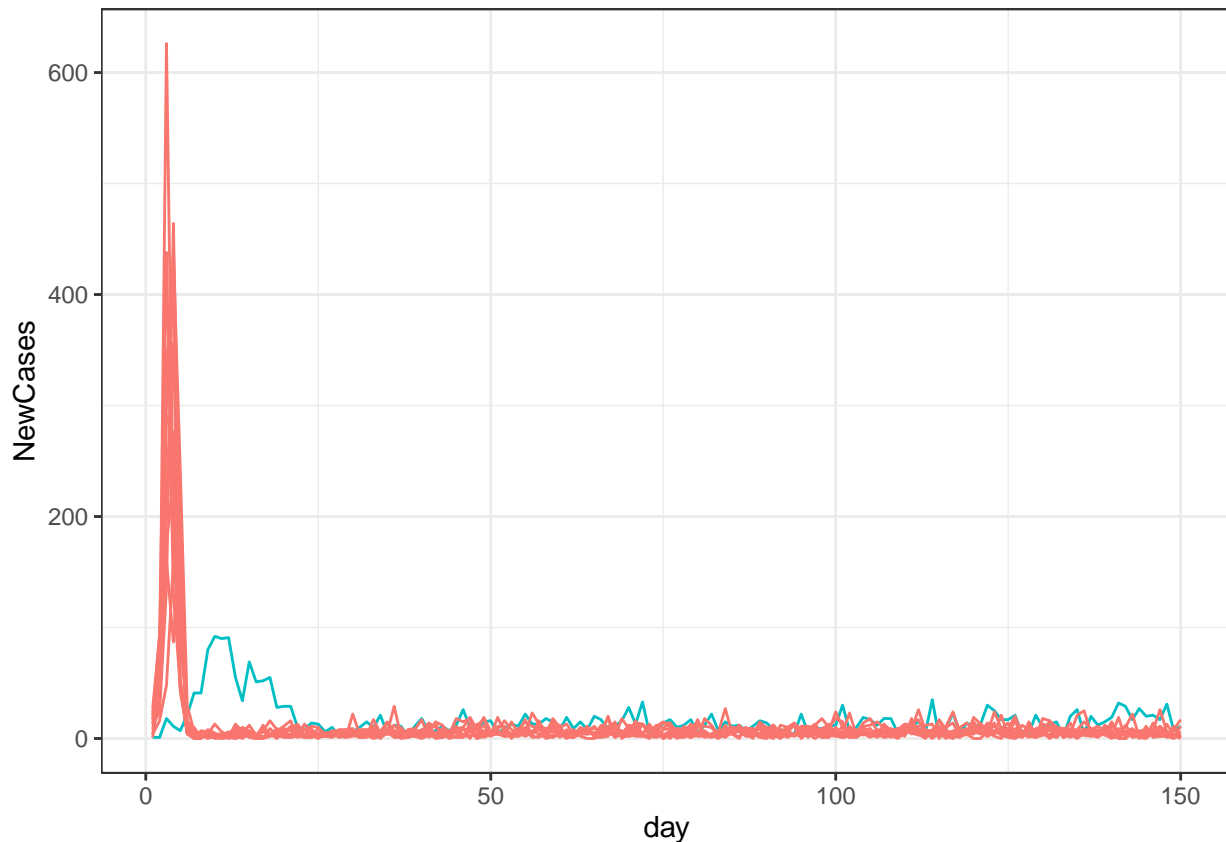
## Warning: in 'table_lookup': extrapolating at 0.000000e+00.
## Warning: in 'table_lookup': extrapolating at 0.000000e+00.
## Warning: in 'table_lookup': extrapolating at 2.500000e-01.
## Warning: in 'table_lookup': extrapolating at 5.000000e-01.
## Warning: in 'table_lookup': extrapolating at 7.500000e-01.

```

```

sims |>
  ggplot(aes(x=day,y=NewCases,group=.id,color=.id=="data"))+
  geom_line()+
  guides(color="none")+
  theme_bw()

```



Sometimes it can help to plot the modeled populations so that we can understand a bit better what we are seeing in the sampling. Here the susceptible population is in blue, the infected population is in red, and the recovered group is in green.

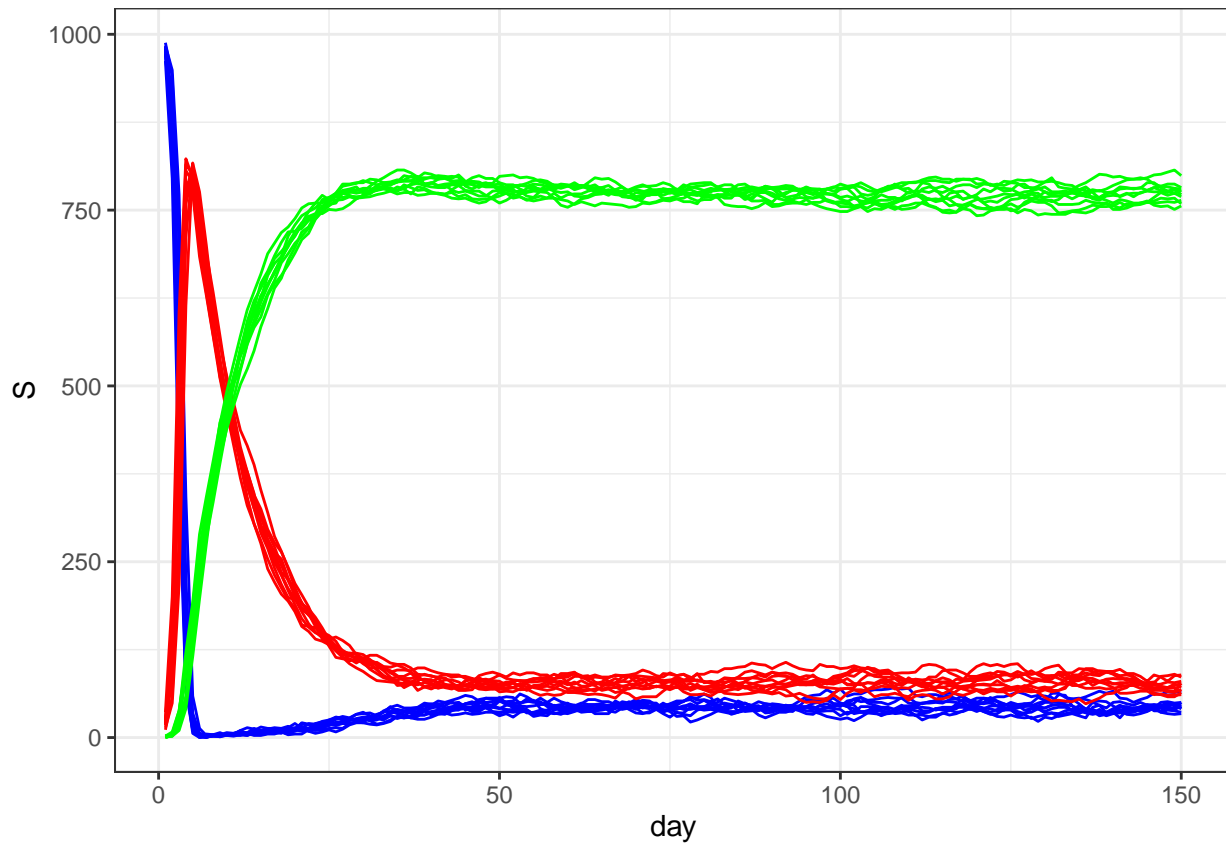
```

sims |>
  ggplot()+
  geom_line(aes(x=day,y=S,group=.id),color="blue")+
  geom_line(aes(x=day,y=I,group=.id),color="red")+

```

```
geom_line(aes(x=day,y=R,group=.id),color="green")+
guides(color="none")+
theme_bw()
```

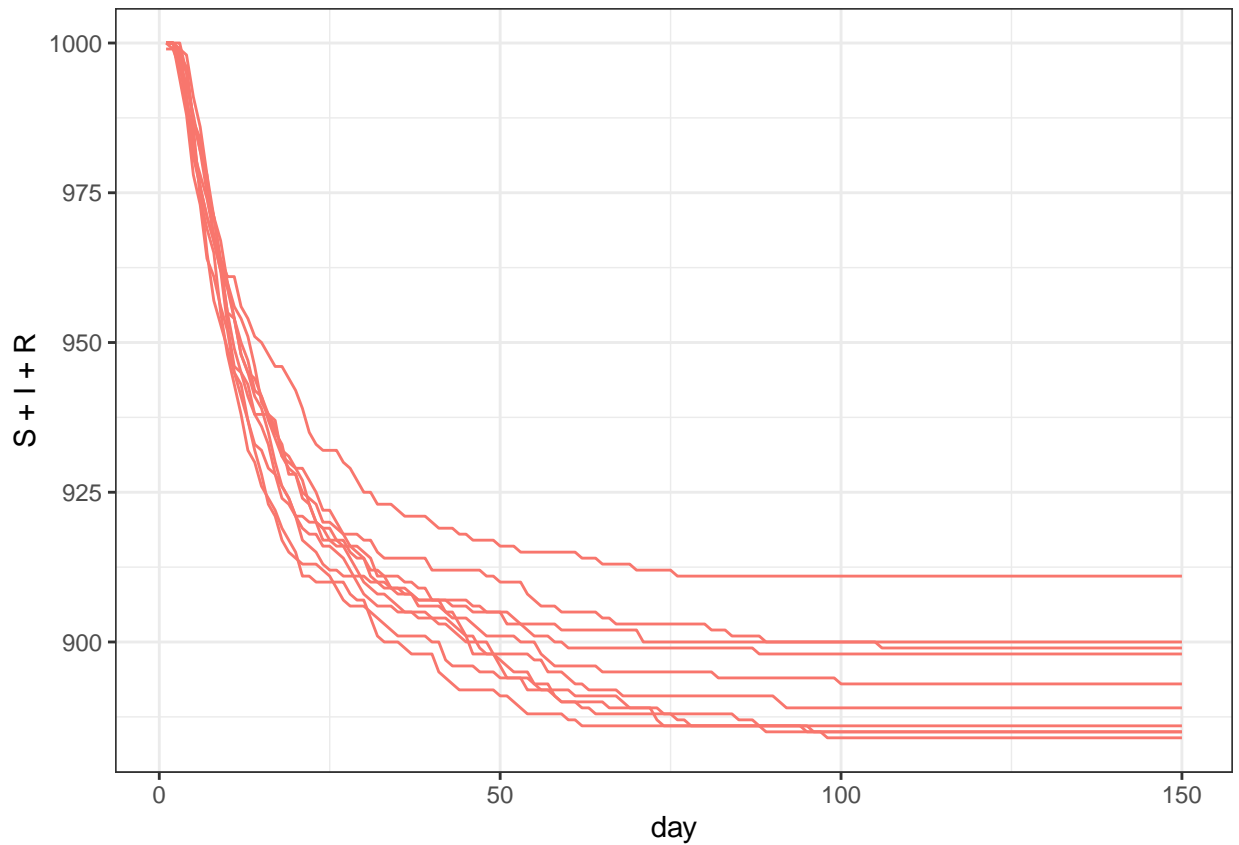
```
## Warning: Removed 150 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 150 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 150 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



We can see the effects of the covariate most clearly if we graph the total population ( $S+I+R$ ). Here we can see the population decline and level off.

```
sims |>
ggplot(aes(x=day,y=S+I+R,group=.id,color=.id=="data"))+
geom_line()+
guides(color="none")+
theme_bw()
```

```
## Warning: Removed 150 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



It is hard to see the effects of the early shifts in deathrate, but we can see that after day 120, the population stays constant in all of our modeled population. This is the result of the death rate being zero after that point.

To play with this you could try different values for the deathrates each day. Can you adjust them until you can clearly see the changes in the graph?