TB Vaccination

PDF Version available here.

0. Data creation

We construct three datasets: `peopleNoVax`, `peopleWithVax`, and `peopleTB`.

`peopleNoVax` is a dataset that spans 17 years. Each year, 10000 people are added to it. Each year, a person has a 5% annual risk of being infected with LTBI. Once a person is infected with LTBI, they have a 10% annual risk of being infected with ATBI.

`peopleWithVax` is a dataset that spans 17 years. Each year, 10000 people are added to it. For the first 10 years, a person has a 5% annual risk of being infected with LTBI. Subsequently they have a 2.5% annual risk of being infected with LTBI. Once a person is infected with LTBI, they have a 10% annual risk of being infected with ATBI.

`peopleTB` is a dataset that spans 17 years. Each year, 10000 people are added to it. Each year, a person has a 10% annual risk of being infected with ATBI.

```r
library(data.table)
library(pomp)
library(ggplot2)
library(rstan)

rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())

#everyone enters on jan 1
#results are for dec 31
newPeople <- data.table(id=1:10000,
                        year=2000,
                        age=1,
                        timeNoVax=0,
                        timeWithVax=0,
                        vaccinated=0,
                        ltbi=0,
                        atbi=0,
                        ageAtTB=0)

if(FALSE){
  file.remove("data/peopleNoVax.RDS")
  file.remove("data/peopleWithVax.RDS")
  file.remove("data/peopleTB.RDS")
  file.remove("results/noVaxLTBI.RDS")
  file.remove("results/withVaxLTBI.RDS")
  file.remove("results/ATBI.RDS")
  file.remove("results/noVaxATBI.RDS")
}

pomp::bake("data/peopleNoVax.RDS",{
  set.seed(4)
  peopleNoVax <- vector("list",length=17)
```

```r
  peopleNoVax[[1]] <- newPeople[1,][-1,]
  for(i in 1:17){
    # NO VACCINATIONS
    if(i!=1) peopleNoVax[[i]] <- copy(peopleNoVax[[i-1]])
    peopleNoVax[[i]][,age:=age+1]
    peopleNoVax[[i]] <- rbind(peopleNoVax[[i]],newPeople)
    peopleNoVax[[i]][,year:=2000+i]

    N <- nrow(peopleNoVax[[i]])
    peopleNoVax[[i]][sample(1:N,size=round(N*0.05)),ltbi:=1]
    peopleNoVax[[i]][,timeNoVax:=timeNoVax+1]

    peopleNoVax[[i]][,atbi:=0] # people with ATBI go to the doctor and get cured last year
    Nltbi <- sum(peopleNoVax[[i]]$ltbi==1 & peopleNoVax[[i]]$ageAtTB==0)
    peopleNoVax[[i]][sample(
      which(peopleNoVax[[i]]$ltbi==1 & peopleNoVax[[i]]$ageAtTB==0),
      size=round(Nltbi*0.10)),
      atbi:=1]
    peopleNoVax[[i]][atbi==1,ageAtTB:=age]
  }
  peopleNoVax
}) -> peopleNoVax

pomp::bake("data/peopleWithVax.RDS",{
  set.seed(4)
  peopleWithVax <- vector("list",length=17)
  peopleWithVax[[1]] <- newPeople[1,][-1,]
  for(i in 1:17){
    # WITH VACCINATIONS, BUT STOPPING THEM AFTERWARDS
    if(i!=1) peopleWithVax[[i]] <- copy(peopleWithVax[[i-1]])
    peopleWithVax[[i]][,age:=age+1]
    peopleWithVax[[i]] <- rbind(peopleWithVax[[i]],newPeople)
    peopleWithVax[[i]][,year:=2000+i]
    N <- nrow(peopleWithVax[[i]])
    if(i>=10){
      peopleWithVax[[i]][sample(1:N,size=round(N*0.050)),ltbi:=1]
      peopleWithVax[[i]][,timeNoVax:=timeNoVax+1]
    } else {
      peopleWithVax[[i]][sample(1:N,size=round(N*0.025)),ltbi:=1]
      peopleWithVax[[i]][,timeWithVax:=timeWithVax+1]
    }

    Nltbi <- sum(peopleWithVax[[i]]$ltbi==1 & peopleWithVax[[i]]$ageAtTB==0)
    peopleWithVax[[i]][atbi==1,ageAtTB:=age]
    peopleWithVax[[i]][,atbi:=0] # people with ATBI go to the doctor and get cured last year
    peopleWithVax[[i]][sample(
      which(peopleWithVax[[i]]$ltbi==1 & peopleWithVax[[i]]$ageAtTB==0),
      size=round(Nltbi*0.10)),
      atbi:=1]
  }
  peopleWithVax
}) -> peopleWithVax
```

```
pomp::bake("data/peopleTB.RDS",{
  set.seed(4)
  peopleTB <- vector("list",length=17)
  peopleTB[[1]] <- newPeople[1,][-1,]
  for(i in 1:17){
    # NO VACCINATIONS
    if(i!=1) peopleTB[[i]] <- copy(peopleTB[[i-1]])
    peopleTB[[i]][,age:=age+1]
    peopleTB[[i]] <- rbind(peopleTB[[i]],newPeople)
    peopleTB[[i]][,year:=2000+i]

    peopleTB[[i]][,timeNoVax:=timeNoVax+1]

    peopleTB[[i]][,atbi:=0] # people with ATBI go to the doctor and get cured last year
    N <- sum(peopleTB[[i]]$ageAtTB==0)
    peopleTB[[i]][sample(which(peopleTB[[i]]$ageAtTB==0),size=round(N*0.10)),atbi:=1]
    peopleTB[[i]][atbi==1,ageAtTB:=age]
  }
  peopleTB
}) -> peopleTB
```

1. Detecting LTBI without vaccination

We will try to model the annual risk of LTBI in the dataset `peopleNoVax`, where LTBI is only measured at the end of the dataset (i.e. after 17 years).

For a person i, who has spent T_i years at risk, their probability of not having LTBI is:

$$Pr(Y_i = 0|p, T_i) = (1 - p)^{T_i}$$

And their corresponding probability of having LTBI is:

$$Pr(Y_i = 1|p, T_i) = 1 - Pr(Y_i = 0|p, T_i)$$

Leaving us with a likelihood function of:

$$L(p|Y, T) = \Pi_{i=1}^{n} \left[1 - (1 - p)^{T_i} \right]^{y_i} \times \left[(1 - p)^{T_i} \right]^{1-y_i}$$

```
stanData <- copy(peopleNoVax[[17]])
stanData <- stanData[,.(R=.N),by=.(timeNoVax,timeWithVax,ltbi)]
data = list(N=nrow(stanData),
            R=stanData$R,
            y=stanData$ltbi*stanData$R,
            timeNoVax=stanData$timeNoVax)

stan_code = "
data {
  int<lower=0> N;
  int R[N];
  int y[N];
  vector[N] timeNoVax;
```

```
}
parameters {
  real<lower=0,upper=0.5> thetaNoVax;
}
model {
  thetaNoVax ~ beta(0.05, 1);

  for (n in 1:N){
    y[n] ~ binomial(R[n],1-((1-thetaNoVax)^timeNoVax[n]));
  }
}
"

pomp::bake("results/noVaxLTBI.RDS",{
  stan(model_code=stan_code,
            model_name="noVaxLTBI",
            data=data,
            iter=10000, chains=4, init=0, seed=4)
}) -> fit
summary(fit)$summary
```

```
##                    mean       se_mean          sd          2.5%
## thetaNoVax  4.994833e-02 2.374938e-06 0.0002040757  4.955082e-02
## lp__       -9.932467e+04 7.893606e-03 0.7160836299 -9.932671e+04
##                    25%          50%          75%         97.5%
## thetaNoVax  4.981124e-02  4.994849e-02  5.008541e-02  5.035190e-02
## lp__       -9.932484e+04 -9.932439e+04 -9.932421e+04 -9.932417e+04
##               n_eff      Rhat
## thetaNoVax 7383.766 1.000461
## lp__       8229.560 1.000168
```

```
plot(fit)
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```

4
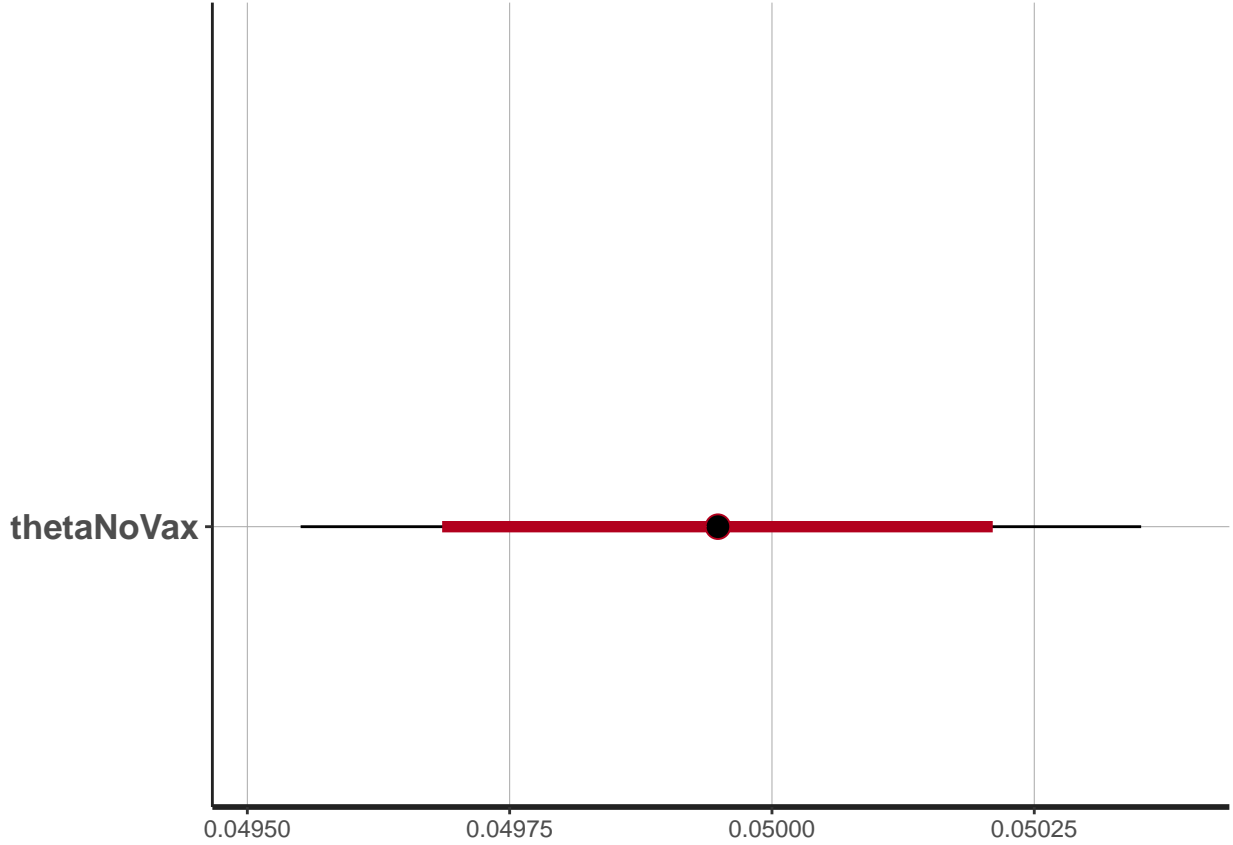
2. Detecting LTBI with vaccination

We will try to model the annual risk of LTBI in the dataset `peopleWithVax`, where LTBI is only measured at the end of the dataset (i.e. after 17 years), however, the risk of LTBI has changed due to the cessation of vaccination.

For a person i, who has spent TW_i years at risk when vaccinations were occurring, and TN_i years at risk when no vaccinations were occuring, their probability of not having LTBI is:

$$Pr(Y_i = 0 | p_w, p_n, TW_i, TN_i) = (1 - p_w)^{TW_i} \times (1 - p_n)^{TN_i}$$

And the probability of person i having LTBI is:

$$Pr(Y_i = 1 | p_w, p_n, TW_i, TN_i) = 1 - Pr(Y_i = 0 | p_w, p_n, TW_i, TN_i)$$

Leaving us with a likelihood function of:

$$L(p_w, p_n | Y, TW, TN) = \Pi_{i=1}^{n} \left[1 - (1 - p_w)^{TW_i} \times (1 - p_n)^{TN_i} \right]^{y_i} \times \left[(1 - p_w)^{TW_i} \times (1 - p_n)^{TN_i} \right]^{1 - y_i}$$

```
stanData <- copy(peopleWithVax[[17]])
stanData <- stanData[,.(R=.N),by=.(timeNoVax,timeWithVax,ltbi)]
data = list(N=nrow(stanData),
            R=stanData$R,
```

5

```
            y=stanData$ltbi*stanData$R,
            timeNoVax=stanData$timeNoVax,
            timeWithVax=stanData$timeWithVax
            )

stan_code = "
data {
  int<lower=0> N;
  int R[N];
  int y[N];
  vector[N] timeNoVax;
  vector[N] timeWithVax;
}
parameters {
  real<lower=0,upper=0.5> thetaNoVax;
  real<lower=0,upper=0.5> thetaWithVax;
}
model {
  thetaWithVax ~ beta(0.05, 1);
  thetaNoVax ~ beta(0.05, 1);

  for (n in 1:N){
    y[n] ~ binomial(R[n], 1-((1-thetaWithVax)^timeWithVax[n])*((1-thetaNoVax)^timeNoVax[n]));
  }
}
"

pomp::bake("results/withVaxLTBI.RDS",{
  stan(model_code=stan_code,
             model_name="withVaxLTBI",
             data=data,
             iter=10000, chains=4, init=0, seed=4)
}) -> fit
summary(fit)$summary
```

```
##                        mean       se_mean           sd          2.5%
## thetaNoVax     4.974044e-02 3.872300e-06 0.0003325474  4.909861e-02
## thetaWithVax   2.577528e-02 8.201603e-06 0.0006877307  2.441685e-02
## lp__          -9.842348e+04 1.116373e-02 0.9928729028 -9.842616e+04
##                         25%           50%           75%         97.5%
## thetaNoVax     4.951529e-02  4.974272e-02  4.996358e-02  5.039312e-02
## thetaWithVax   2.531198e-02  2.578225e-02  2.623945e-02  2.712795e-02
## lp__          -9.842387e+04 -9.842317e+04 -9.842277e+04 -9.842251e+04
##                     n_eff      Rhat
## thetaNoVax     7375.119 1.000426
## thetaWithVax   7031.360 1.000248
## lp__           7909.861 1.000069
```
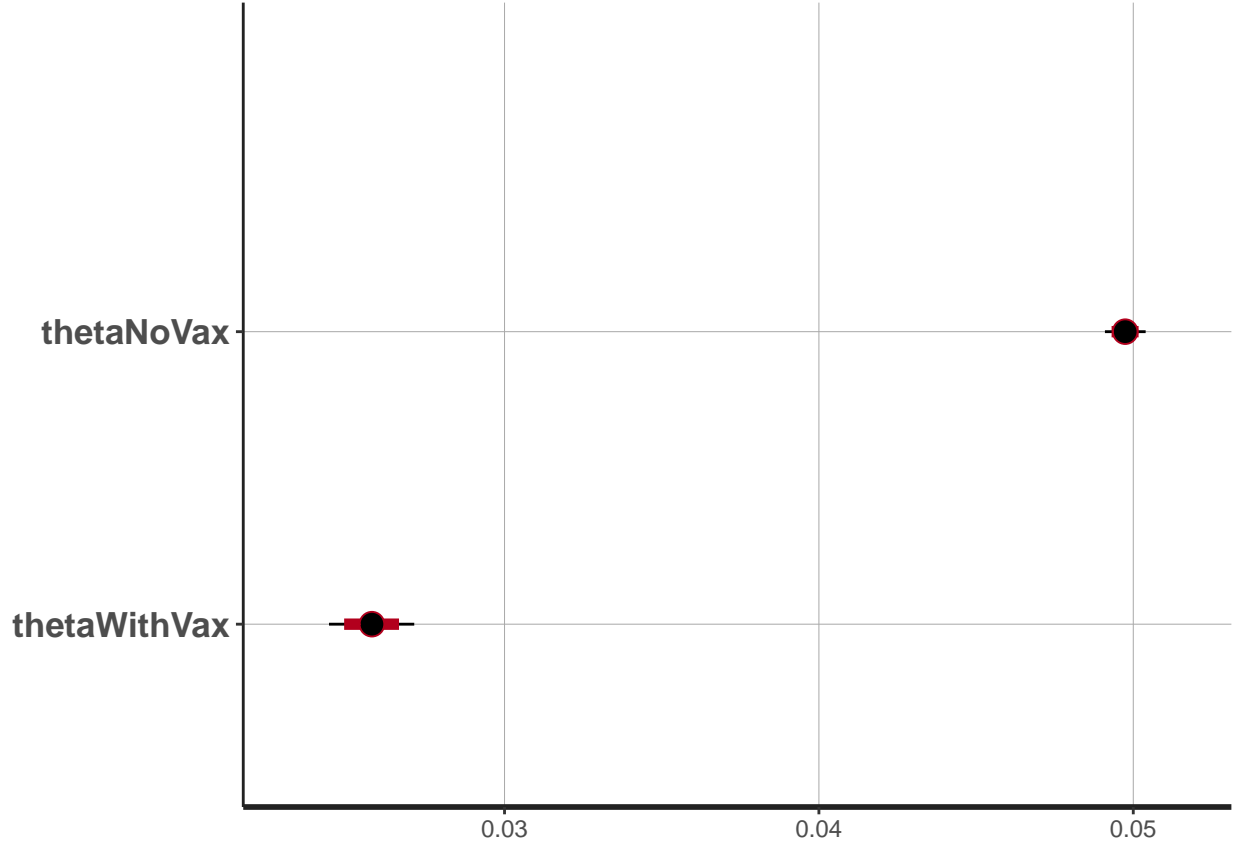
```
plot(fit)
```

```
## ci_level: 0.8 (80% intervals)

## outer_level: 0.95 (95% intervals)
```

3. Detecting straight ATBI without vaccination

We will try to model the annual risk of ATBI in the dataset `peopleTB`, where ATBI is measured at the end of each year. This is an example of survival analysis, as the people who were not infected with ATBI will have a right-censored time at the end of the followup.

Standard survival analysis definitions apply here. The probability of an individual's surviving (i.e. not acquiring ATBI) till time t is given by the survivor function:

$$S(t) = 1 - F(t) = P(T > t)$$

The likelihood function for parameters λ, given data D is:

$$L(\lambda|D) = \Pi_{i=1}^{n} f(y_i|\lambda)_i^v \times S(y_i|\lambda)^{(1-v_i)}$$

Where $v_i = 1$ if person i is uncensored (0 if censored), and f is the density distribution for y_i (time until ATBI).

In our dataset, for person i, the probability that they will get ATBI after y_i years is distributed as a geometric distribution:

$$f(Y_i = y_i|p) = (1-p)^{(y_i-1)} \times p$$

The survival function can thus be defined as:

$$S(y_i|p) = (1-p)^{y_i}$$

```r
cases <- list()
for(i in 1:17){
  cases[[i]] <- peopleTB[[i]][atbi==1]
}
cases <- rbindlist(cases)
cases[,timeNoVax:=timeNoVax+1]
controls <- peopleTB[[17]][ageAtTB==0]

cases <- cases[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]
controls <- controls[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]

stanData <- rbind(controls,cases)
#stanData <- cases
data = list(N=nrow(stanData),
            R=stanData$R,
            y=stanData$atbi*stanData$R,
            timeNoVax=stanData$timeNoVax)

stan_code = "
data {
  int<lower=0> N;
  int timeNoVax[N];
  int R[N];
  int y[N];
}
parameters {
  real<lower=0,upper=0.5> atbiNoVax;
}
model {
  atbiNoVax ~ beta(0.05, 1);

  {
    int replicates;
    int personTime;
    real prob_no_atbi;
    real prob_atbi_in_year;

    for (n in 1:N){
      prob_no_atbi = (1-atbiNoVax)^timeNoVax[n];
      prob_atbi_in_year = (1-atbiNoVax)^(timeNoVax[n]-1)*atbiNoVax;
      replicates = R[n];

      if( y[n]==0 ){
        target += R[n]*log(prob_no_atbi);
      } else {
        target += binomial_lpmf(y[n] | R[n], prob_atbi_in_year);
      }

    }
  }
}
```

```
"

pomp::bake("results/ATBI.RDS",{
  stan(model_code=stan_code,
            model_name="ATBI",
            data=data,
            iter=10000, chains=4, init=0, seed=4)
}) -> fit
summary(fit)$summary
```

```
##                  mean       se_mean          sd          2.5%
## atbiNoVax  9.091132e-02 3.130571e-06 0.0002812406  9.036228e-02
## lp__      -3.183761e+05 7.733701e-03 0.7176592334 -3.183781e+05
##                   25%          50%          75%          97.5%    n_eff
## atbiNoVax  9.072289e-02  9.090863e-02  9.109960e-02  9.146753e-02 8070.655
## lp__      -3.183762e+05 -3.183758e+05 -3.183756e+05 -3.183756e+05 8611.163
##              Rhat
## atbiNoVax 1.0002705
## lp__      0.9998968
```

```
plot(fit)
```

ci_level: 0.8 (80% intervals)

outer_level: 0.95 (95% intervals)
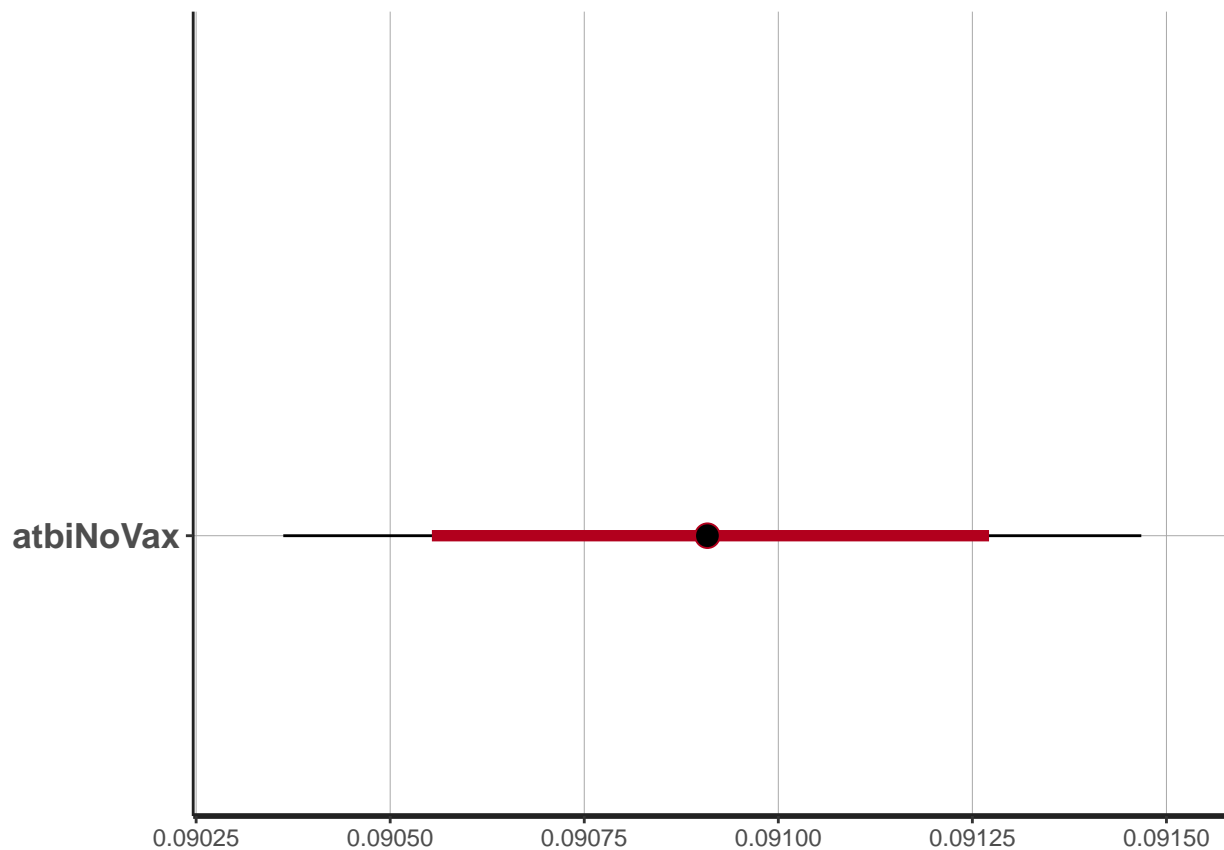

```

## 4. Detecting ATBI following LTBI without vaccination (prespecifying ATBI risk)

Using the dataset `peopleNoVax`, we try to estimate the annual risk of LTBI and ATBI, given that we can only observe ATBI in the year that it occurs. This is an extension of the previous models.

In our dataset, for person $i$, the probability that they will get LTBI after $l_i$ years is distributed as a geometric distribution:

$$f(L_i = l_i | p_L) = (1 - p_L)^{(l_i - 1)} \times p_L$$

where $l_i = 1, \ldots, \infty$.

In our dataset, for person $i$, the probability that they will get ATBI $a_i$ years after being infected with LTBI (where the year of LTBI acquirement counts as a year) is distributed as a geometric distribution:

$$f(A_i = a_i | p_A) = (1 - p_A)^{(a_i - 1)} \times p_A$$

where $a_i = 1, \ldots, \infty$.

We can put these together, and say that for person $i$, the probability that they will get ATBI after $y_i$ years is equal to $l_i + a_i$ and is distributed as a geometric distribution:

$$f(Y_i = y_i | p_L, p_A) = \Sigma_{t=1}^{y_i} f(L_i = t | p_L) \times f(A_i = y_i - t + 1 | p_A)$$

$$f(Y_i = y_i | p_L, p_A) = \Sigma_{t=1}^{y_i} (1 - p_L)^{(t-1)} \times p_L \times (1 - p_A)^{(y_i - t)} \times p_A$$

The survival function can thus be defined as:

$$S(y_i | p_L, p_A) = 1 - \Sigma_{t=1}^{y_i} f(Y_i = t | p_L, p_A)$$

We can also see the distribution of Y using simulations:

```
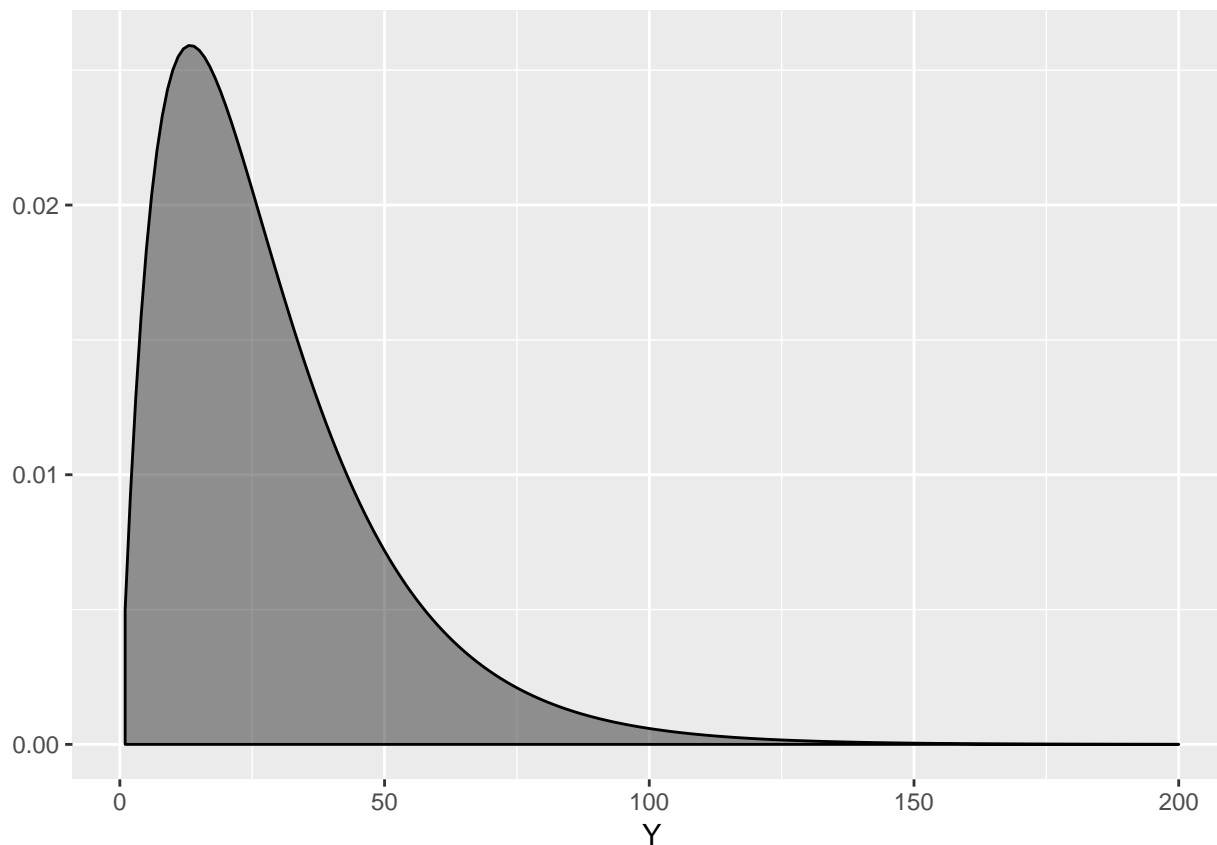pl <- 0.05
pa <- 0.10
TS <- ts <- 200

f <- c() # ATBI IN FIRST 10 YRS
f[1]=0
for(i in 1:TS){
  ts <- i
  t <- 1:(ts)
  f[i] <- sum(((1-pl)^(t-1))*pl*((1-pa)^(ts-t))*pa)
}

pd <- data.frame(Y=1:TS,p=f)
q <- ggplot(pd,aes(x=Y,ymax=p,ymin=0))
q <- q + geom_ribbon(alpha=0.5,colour="black")
q
```

We also note that it sums to 1:

```
sum(f)
```

```
## [1] 0.9999334
```

We can now try to fit this using stan, where we hardcode the ATBI risk at $p_A = 0.10$:

```
#file.remove("results/noVaxATBI.RDS")
cases <- list()
for(i in 1:17){
  cases[[i]] <- peopleNoVax[[i]][atbi==1]
}
cases <- rbindlist(cases)
controls <- peopleNoVax[[17]][ageAtTB==0]

cases <- cases[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]
controls <- controls[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]

stanData <- rbind(controls,cases)
print(stanData)
```

```
##      timeNoVax timeWithVax atbi    R
## 1:          17           0    0 6428
## 2:          16           0    0 6724
## 3:          15           0    0 6926
## 4:          14           0    0 7196
## 5:          13           0    0 7532
```

```
##  6:          12               0     0 7642
##  7:          11               0     0 8011
##  8:          10               0     0 8288
##  9:           9               0     0 8497
## 10:           8               0     0 8701
## 11:           7               0     0 8974
## 12:           6               0     0 9145
## 13:           5               0     0 9417
## 14:           4               0     0 9556
## 15:           3               0     0 9750
## 16:           2               0     0 9846
## 17:           1               0     0 9955
## 18:           1               0     1  812
## 19:           2               0     1 1464
## 20:           3               0     1 1923
## 21:           4               0     1 2203
## 22:           5               0     1 2381
## 23:           6               0     1 2438
## 24:           7               0     1 2505
## 25:           8               0     1 2296
## 26:           9               0     1 2151
## 27:          10               0     1 2061
## 28:          11               0     1 1818
## 29:          12               0     1 1537
## 30:          13               0     1 1343
## 31:          14               0     1  990
## 32:          15               0     1  779
## 33:          16               0     1  488
## 34:          17               0     1  223
##      timeNoVax timeWithVax atbi     R
```

```r
#stanData <- cases
data = list(N=nrow(stanData),
            R=stanData$R,
            ATBIstatus=stanData$atbi,
            timeNoVax=stanData$timeNoVax)

stan_code = "
data {
  int<lower=0> N;
  int timeNoVax[N];
  int R[N];
  int ATBIstatus[N];
}
parameters {
  real<lower=0,upper=0.5> ltbiNoVax;
}
model {
real ps;
  ltbiNoVax ~ beta(0.05, 1);

  {
    real atbiNoVax;
    int replicates;
```

12

```
    int personTime;
    real prob_atbi_in_year;
    real survival;
    real p[201];

    atbiNoVax=0.10;

    for(q in 1:17){
      prob_atbi_in_year=0;
      for(t in 1:q){
        prob_atbi_in_year = prob_atbi_in_year + ((1-ltbiNoVax)^(t-1))*ltbiNoVax*((1-atbiNoVax)^(q-t))*a
      }
      p[q]=prob_atbi_in_year;
    }

    for (n in 1:N){
      personTime = timeNoVax[n];
      replicates = R[n];

      if( ATBIstatus[n]==1 ){
        target += replicates*log(p[personTime]);
      } else {
        survival=0;
        for(q in 1:personTime){
          survival=survival+p[q];
        }
        target += replicates*log(1-survival);
      }
    }
  }
}

"

pomp::bake("results/noVaxATBI_fixed_ATBI.RDS",{
  stan(model_code=stan_code,
            model_name="noVaxATBI_fixed_ATBI",
            data=data,
            iter=2000, chains=4, init=0, seed=4)
}) -> fit
summary(fit)$summary
```

```
##                   mean      se_mean           sd           2.5%
## ltbiNoVax  5.003396e-02 7.643335e-06 0.0003261038   4.939284e-02
## lp__      -1.316050e+05 1.707662e-02 0.7256806503 -1.316071e+05
##                    25%          50%          75%        97.5%    n_eff
## ltbiNoVax  4.982046e-02  5.002861e-02  5.024081e-02  5.072224e-02 1820.313
## lp__      -1.316051e+05 -1.316047e+05 -1.316045e+05 -1.316045e+05 1805.873
##                  Rhat
## ltbiNoVax 1.001902
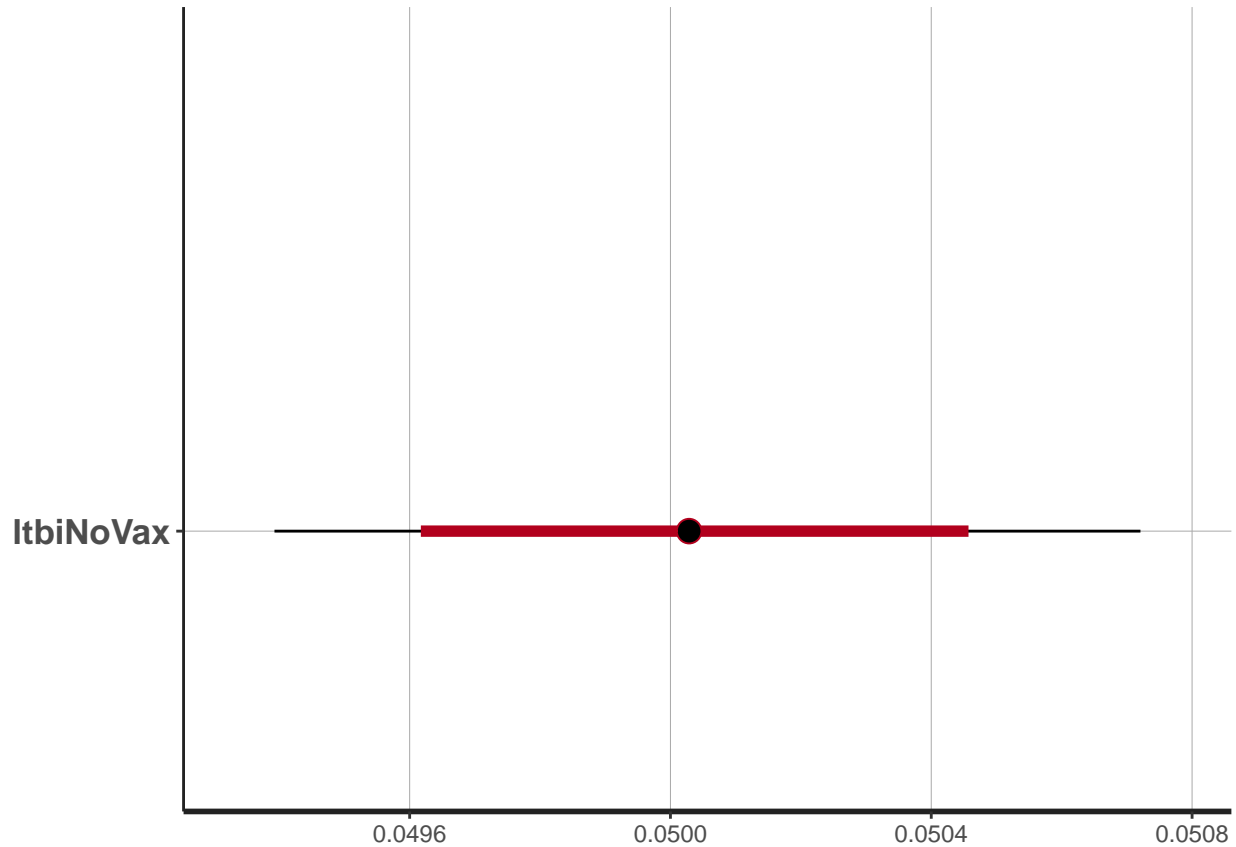## lp__      1.000804
```

```r
plot(fit)
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```


This meets our expected output of `ltbiNoVax=0.05`

5. Detecting ATBI following LTBI without vaccination (estimating ATBI risk)

This is the same as the previous model, however, we attempt to estimate p_A.

```r
#file.remove("results/noVaxATBI.RDS")
cases <- list()
for(i in 1:17){
  cases[[i]] <- peopleNoVax[[i]][atbi==1]
}
cases <- rbindlist(cases)
controls <- peopleNoVax[[17]][ageAtTB==0]

cases <- cases[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]
controls <- controls[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]

stanData <- rbind(controls,cases)
print(stanData)
```

```
##      timeNoVax timeWithVax atbi    R
```

```
##  1:          17           0     0 6428
##  2:          16           0     0 6724
##  3:          15           0     0 6926
##  4:          14           0     0 7196
##  5:          13           0     0 7532
##  6:          12           0     0 7642
##  7:          11           0     0 8011
##  8:          10           0     0 8288
##  9:           9           0     0 8497
## 10:           8           0     0 8701
## 11:           7           0     0 8974
## 12:           6           0     0 9145
## 13:           5           0     0 9417
## 14:           4           0     0 9556
## 15:           3           0     0 9750
## 16:           2           0     0 9846
## 17:           1           0     0 9955
## 18:           1           0     1  812
## 19:           2           0     1 1464
## 20:           3           0     1 1923
## 21:           4           0     1 2203
## 22:           5           0     1 2381
## 23:           6           0     1 2438
## 24:           7           0     1 2505
## 25:           8           0     1 2296
## 26:           9           0     1 2151
## 27:          10           0     1 2061
## 28:          11           0     1 1818
## 29:          12           0     1 1537
## 30:          13           0     1 1343
## 31:          14           0     1  990
## 32:          15           0     1  779
## 33:          16           0     1  488
## 34:          17           0     1  223
##      timeNoVax timeWithVax atbi    R
```

```r
#stanData <- cases
data = list(N=nrow(stanData),
            R=stanData$R,
            ATBIstatus=stanData$atbi,
            timeNoVax=stanData$timeNoVax)

stan_code = "
data {
  int<lower=0> N;
  int timeNoVax[N];
  int R[N];
  int ATBIstatus[N];
}
parameters {
  real<lower=0,upper=0.5> ltbiNoVax;
  real<lower=0,upper=0.5> atbiNoVax;
}
model {
```

```
real ps;
  ltbiNoVax ~ beta(0.05, 1);
  atbiNoVax ~ normal(0.1, 0.05);

  {
    int replicates;
    int personTime;
    real prob_atbi_in_year;
    real survival;
    real p[201];

    for(q in 1:17){
      prob_atbi_in_year=0;
      for(t in 1:q){
        prob_atbi_in_year = prob_atbi_in_year + ((1-ltbiNoVax)^(t-1))*ltbiNoVax*((1-atbiNoVax)^(q-t))*a
      }
      p[q]=prob_atbi_in_year;
    }

    for (n in 1:N){
      personTime = timeNoVax[n];
      replicates = R[n];

      if( ATBIstatus[n]==1 ){
        target += replicates*log(p[personTime]);
      } else {
        survival=0;
        for(q in 1:personTime){
          survival=survival+p[q];
        }
        target += replicates*log(1-survival);
      }
    }
  }
}

"

pomp::bake("results/noVaxATBI.RDS",{
  stan(model_code=stan_code,
           model_name="noVaxATBI",
           data=data,
           iter=2000, chains=4, init=0, seed=4)
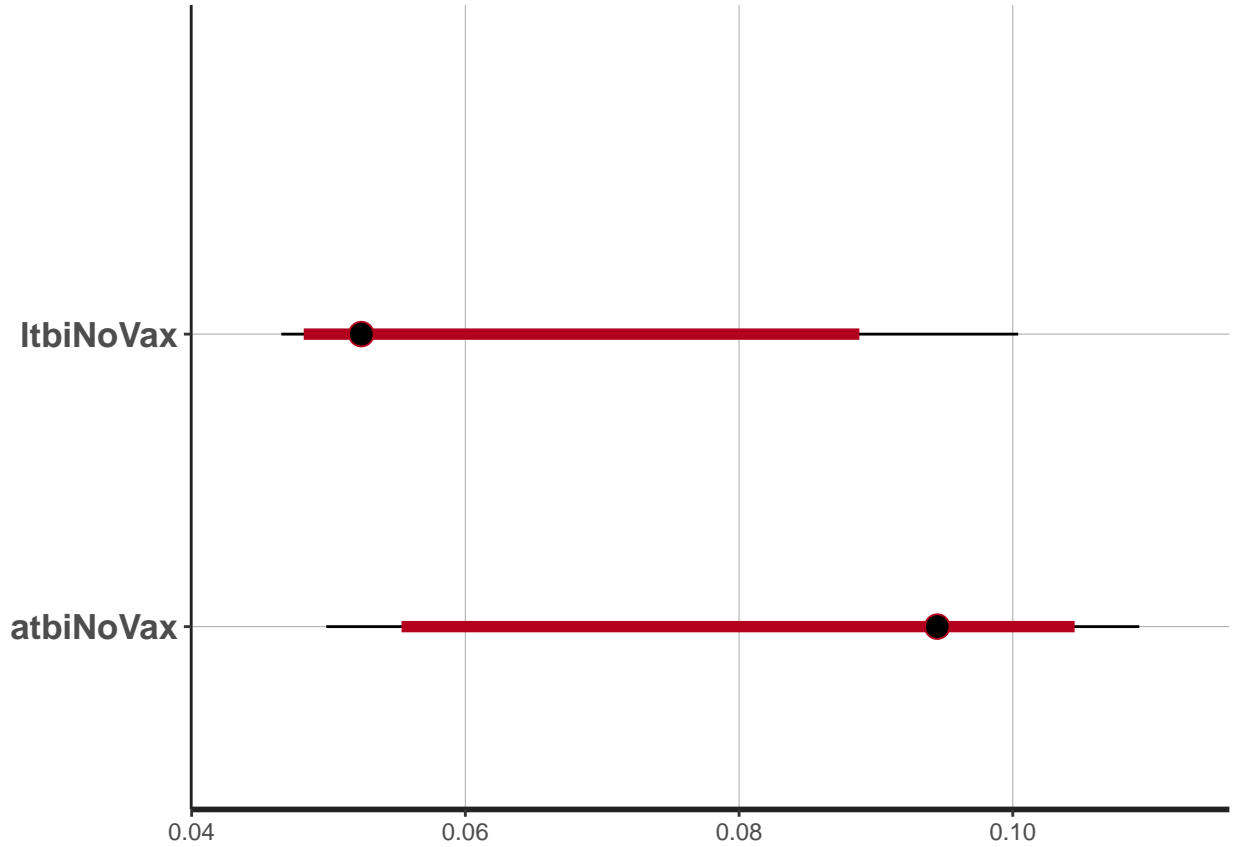}) -> fit
summary(fit)$summary

##                     mean     se_mean         sd        2.5%          25%
## ltbiNoVax  5.814597e-02 0.001954316 0.01497471  4.655418e-02  4.991536e-02
## atbiNoVax  8.948722e-02 0.002072485 0.01659243  4.984678e-02  8.687115e-02
## lp__      -1.315497e+05 0.060167291 1.03708044 -1.315522e+05 -1.315502e+05
##                    50%          75%         97.5%    n_eff      Rhat
## ltbiNoVax  5.239780e-02  5.636854e-02  1.003905e-01  58.71207 1.022357
## atbiNoVax  9.447974e-02  1.001260e-01  1.092432e-01  64.09692 1.017786
## lp__      -1.315494e+05 -1.315489e+05 -1.315485e+05 297.10090 1.005172
```

```
plot(fit)
```

```
## ci_level: 0.8 (80% intervals)
## outer_level: 0.95 (95% intervals)
```


Unfortunately, this does not meet our expected output of `ltbiNoVax=0.05` and `atbiNoVax=0.025`

6. Detecting ATBI following LTBI with vaccination (prespecifying ATBI risk)

Using the dataset `peopleWithVax`, we try to estimate the annual risk of LTBI and ATBI, given that we can only observe ATBI in the year that it occurs. This is an extension of the previous models.

In our dataset, for person i, whose time at risk began in year s_i, with vaccination ceasing in year v (i.e. no vaccination occurring in year V_i), the probability that they will get LTBI after l_i years is distributed as a geometric distribution:

For $v - s_i \leq 0 < l_i$:

$$f(L_i = l_i | p_{LN}, p_{LW}) = (1 - p_{LN})^{(l_i - 1)} \times p_{LN}$$

For $0 < l_i < v - s_i$

$$f(L_i = l_i | p_{LN}, p_{LW}) = (1 - p_{LW})^{(l_i - 1)} \times p_{LW}$$

For $0 < v - s_i \leq l_i$ $s_i < v$ and $s_i + l_i \geq v$:

$$f(L_i = l_i|p_{LN}, p_{LW}) = (1 - p_{LW})^{(S_i-1)} \times (1 - p_{LN})^{(l_i-1)} \times p_{LN}$$

To simplify things, we can define years spent during vaccination as $tw_i = max(min(v - s_i, l_i), 0)$, and years spent without vaccination as $tn_i = (l_i - max(v_i - s_i, 0)) \times I(v - s_i \leq l_i)$.

Scenario 1: $s_i = 2001$, $v = 2010$, $l_i = 1$:

$tw_i = max(min(2010 - 2001, 1), 0) = max(min(9, 1), 0) = max(1, 0) = 1$

$tn_i = (1 - max(2010 - 2001, 0)) \times I(2010 - 2001 \leq 1) = (1 - max(9, 0)) \times I(9 \leq 1) = 0$

Scenario 2: $s_i = 2001$, $v = 2010$, $l_i = 9$:

$tw_i = max(min(2010 - 2001, 9), 0) = max(min(9, 9), 0) = max(9, 0) = 9$

$tn_i = (9 - max(2010 - 2001, 0)) \times I(2010 - 2001 \leq 9) = (9 - max(9, 0)) \times I(9 \leq 9) = 0$

Scenario 3: $s_i = 2001$, $v = 2010$, $l_i = 10$:

$tw_i = max(min(2010 - 2001, 10), 0) = max(min(9, 10), 0) = max(9, 0) = 9$

$tn_i = (10 - max(2010 - 2001, 0)) \times I(2010 - 2001 \leq 10) = (10 - max(9, 0)) \times I(9 \leq 10) = 1$

Scenario 4: $s_i = 2011$, $v = 2010$, $l_i = 10$:

$tw_i = max(min(2010 - 2011, 10), 0) = max(min(-1, 10), 0) = max(-1, 0) = 0$

$tn_i = (10 - max(2010 - 2011, 0)) \times I(2010 - 2011 \leq 10) = (10 - max(-1, 0)) \times I(-1 \leq 10) = 10$

Showing that tw_i and tn_i are correctly specified, we can now rewrite $f(L_i = l_i|p_{LN}, p_{LW})$:

$$f(L_i = l_i|p_{LN}, p_{LW}) = (1 - p_{LW})^{(TW_i-1) \times I(TW_i>0)} \times (1 - p_{LN})^{(TN_i-1) \times I(TN_i>0)} \times q$$

where

$$q = (1 - p_{LW})^{I(TW_i>0 \ \& \ TN_i>0)} \times p_{LW}^{I(TW_i>0 \ \& \ TN_i=0)} \times p_{LN}^{I(TN_i>0)}$$

```
plw <- 0.025
pln <- 0.05
pa <- 0.10

s <- 2001
v <- 2010

TS <- 100
f <- c()
for(l in 1:TS){
  tw <- max(min(v-s,l),0)
  tn <- (l-max(v-s,0))*I(v-s<l)
  q <- 1
  q <- q * (1-plw)^I(tw>0 & tn>0)
  q <- q * plw^(tw>0 & tn==0)
  q <- q * pln^(tn>0)

  f[l] <- q
  if(tw>0) f[l] <- f[l] * (1-plw)^(tw-1)
  if(tn>0) f[l] <- f[l]*(1-pln)^(tn-1)
}
```

```
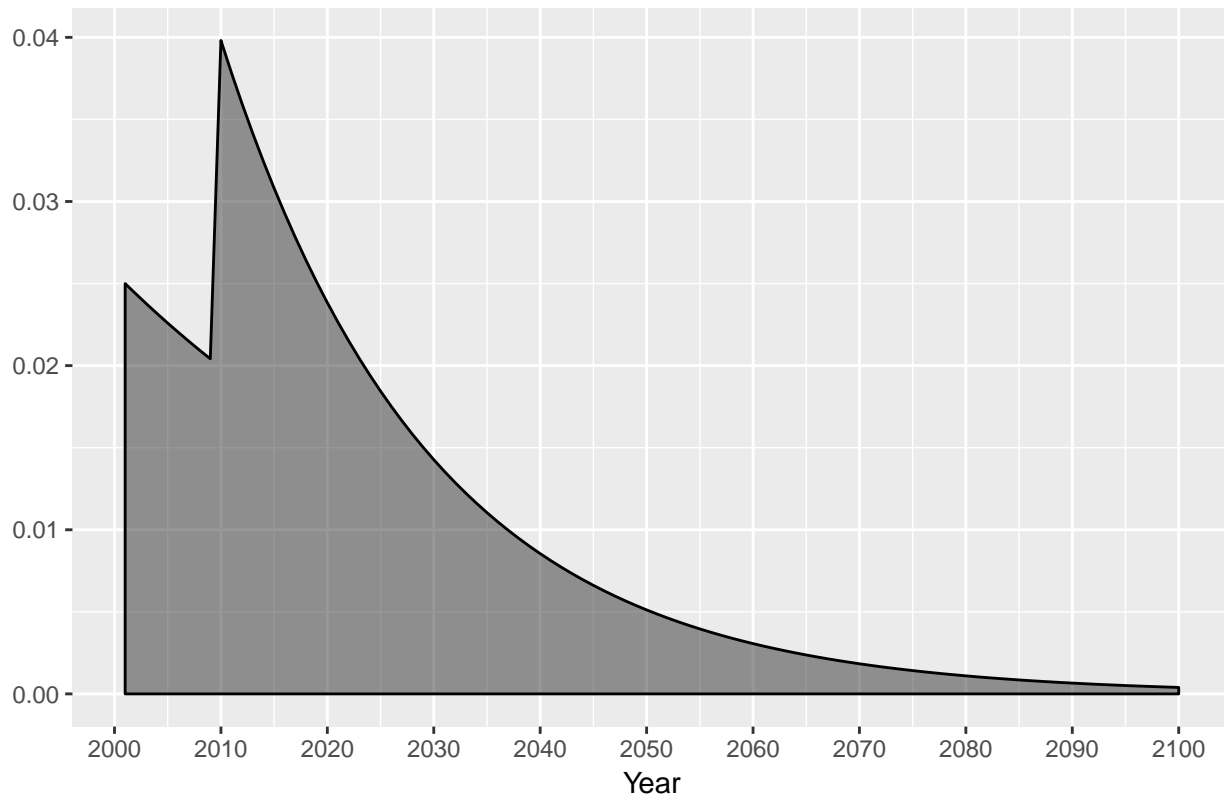sum(f)
```

```
## [1] 0.9925202
```

```
pd <- data.frame(Y=1:TS+s-1,p=f)
q <- ggplot(pd,aes(x=Y,ymax=p,ymin=0))
q <- q + geom_ribbon(alpha=0.5,colour="black")
q <- q + scale_x_continuous("Year",breaks=seq(2000,2100,10))
q <- q + labs(title=sprintf("PDF of year of LTBI infection for a person born in %s, vaccination ceasing
q
```

PDF of year of LTBI infection for a person born in 2001, vaccination ceasing i


```
plw <- 0.025
pln <- 0.05
pa <- 0.10

s <- 2020
v <- 2010

TS <- 100
f <- c()
for(l in 1:TS){
  tw <- max(min(v-s,l),0)
  tn <- (l-max(v-s,0))*I(v-s<l)
  q <- 1
  q <- q * (1-plw)^I(tw>0 & tn>0)
  q <- q * plw^(tw>0 & tn==0)
```

```
    q <- q * pln^(tn>0)

    f[l] <- q
    if(tw>0) f[l] <- f[l] * (1-plw)^(tw-1)
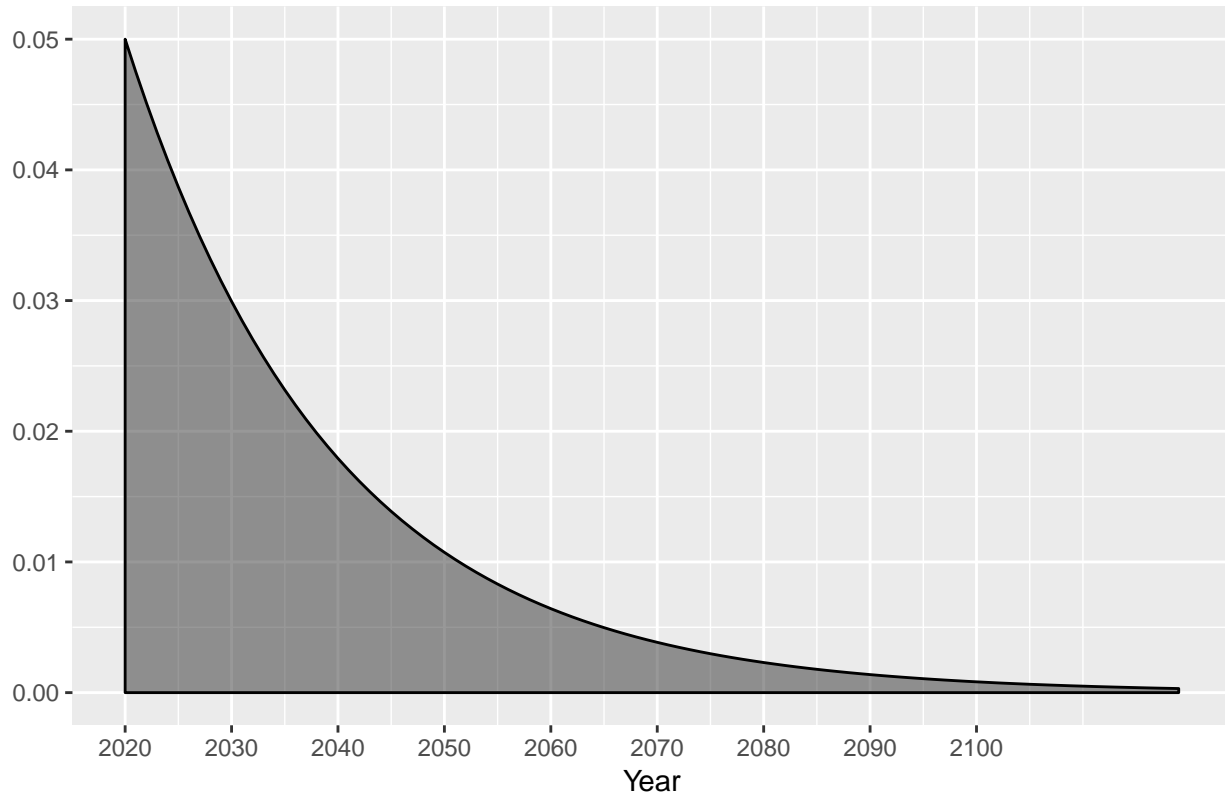    if(tn>0) f[l] <- f[l]*(1-pln)^(tn-1)
}

sum(f)
```

```
## [1] 0.9940795
```

```
pd <- data.frame(Y=1:TS+s-1,p=f)
q <- ggplot(pd,aes(x=Y,ymax=p,ymin=0))
q <- q + geom_ribbon(alpha=0.5,colour="black")
q <- q + scale_x_continuous("Year",breaks=seq(2000,2100,10))
q <- q + labs(title=sprintf("PDF of year of LTBI infection for a person born in %s, vaccination ceasing
q
```


PDF of year of LTBI infection for a person born in 2020, vaccination ceasing i

$$f(L_i = l_i | p_{LN}, p_{LW}) = (1 - p_{LW})^{(S_i - 1)} \times (1 - p_{LN})^{(l_i - 1)} \times p_{LN}$$

where $l_i = 1, \ldots, \infty$.

In our dataset, for person i, the probability that they will get ATBI a_i years after being infected with LTBI (where the year of LTBI acquirement counts as a year) is distributed as a geometric distribution:

$$f(A_i = a_i | p_A) = (1 - p_A)^{(a_i - 1)} \times p_A$$

where $a_i = 1, \ldots, \infty$.

We can put these together, and say that for person i, the probability that they will get ATBI after y_i years is equal to $l_i + a_i$ and is distributed as a geometric distribution:

$$f(Y_i = y_i | p_L, p_A) = \Sigma_{t=1}^{y_i} f(L_i = t | p_L) \times f(A_i = y_i - t + 1 | p_A)$$

$$f(Y_i = y_i | p_L, p_A) = \Sigma_{t=1}^{y_i} (1 - p_L)^{(t-1)} \times p_L \times (1 - p_A)^{(y_i - t)} \times p_A$$

The survival function can thus be defined as:

$$S(y_i | p_L, p_A) = 1 - \Sigma_{t=1}^{y_i} f(Y_i = t | p_L, p_A)$$

We can also see the distribution of Y using simulations:

```
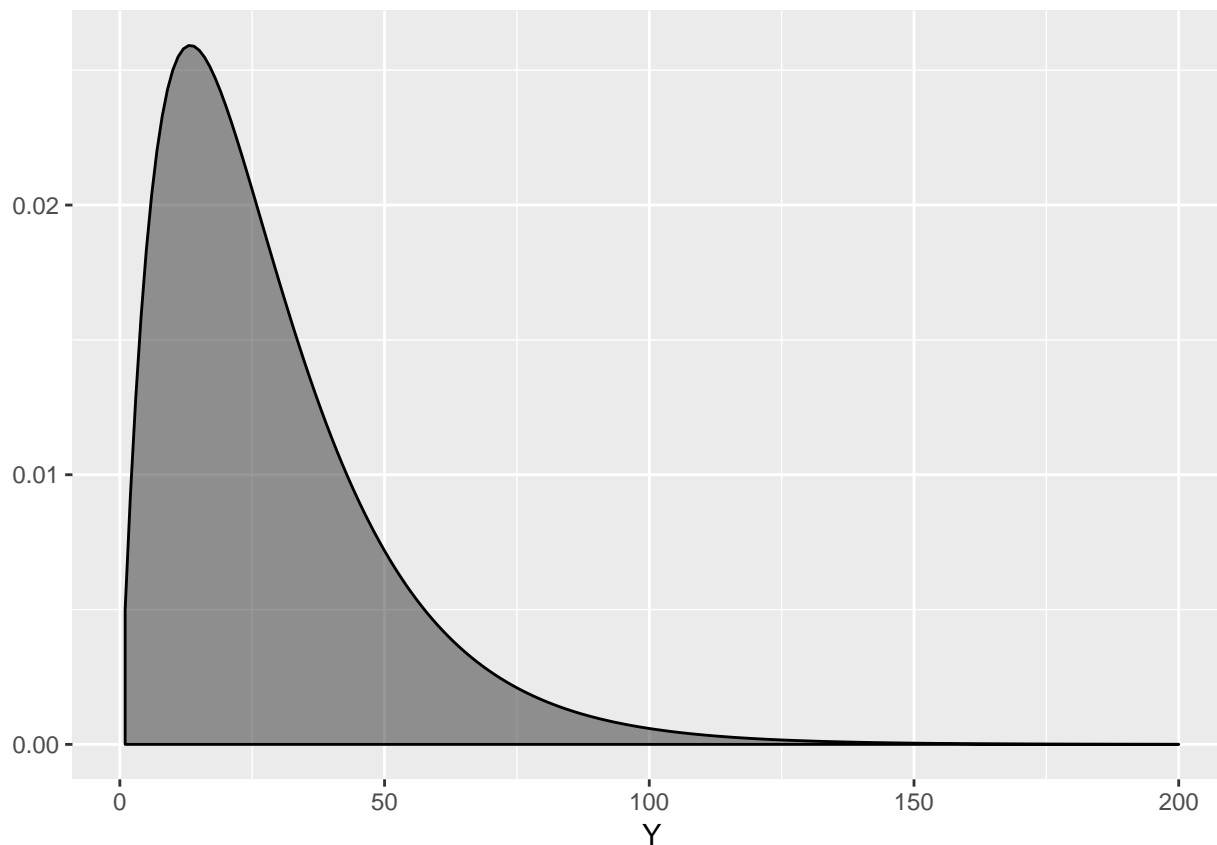pl <- 0.05
pa <- 0.10
TS <- ts <- 200

f <- c() # ATBI IN FIRST 10 YRS
f[1]=0
for(i in 1:TS){
  ts <- i
  t <- 1:(ts)
  f[i] <- sum(((1-pl)^(t-1))*pl*((1-pa)^(ts-t))*pa)
}

pd <- data.frame(Y=1:TS,p=f)
q <- ggplot(pd,aes(x=Y,ymax=p,ymin=0))
q <- q + geom_ribbon(alpha=0.5,colour="black")
q
```

We also note that it sums to 1:

```
sum(f)
```

```
## [1] 0.9999334
```

We can now try to fit this using stan:

```
#file.remove("results/noVaxATBI.RDS")
cases <- list()
for(i in 1:17){
  cases[[i]] <- peopleWithVax[[i]][atbi==1]
}
cases <- rbindlist(cases)
controls <- peopleWithVax[[17]][ageAtTB==0]

cases <- cases[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]
controls <- controls[,.(R=.N),by=.(timeNoVax,timeWithVax,atbi)]

stanData <- rbind(controls,cases)
print(stanData)
```

```
##      timeNoVax timeWithVax atbi    R
##   1:         8           9    0 7336
##   2:         8           9    1  220
##   3:         8           8    0 7537
##   4:         8           8    1  264
##   5:         8           7    0 7684
```

22

```
##  ---
## 119:          8             4   1  248
## 120:          8             3   1  233
## 121:          8             2   1  276
## 122:          8             1   1  230
## 123:          8             0   1  251
```

```r
#stanData <- cases
data = list(N=nrow(stanData),
            R=stanData$R,
            ATBIstatus=stanData$atbi,
            timeNoVax=stanData$timeNoVax)

stan_code = "
data {
  int<lower=0> N;
  int timeNoVax[N];
  int timeWithVax[N];
  int R[N];
  int ATBIstatus[N];
}
parameters {
  real<lower=0,upper=0.5> ltbiNoVax;
}
model {
real ps;
  ltbiNoVax ~ beta(0.05, 1);

  {
    real atbiNoVax;
    int replicates;
    int personTime;
    real prob_atbi_in_year;
    real survival;
    real p[201];

    atbiNoVax=0.10;

    for(q in 1:17){
      prob_atbi_in_year=0;
      for(t in 1:q){
        prob_atbi_in_year = prob_atbi_in_year + ((1-ltbiNoVax)^(t-1))*ltbiNoVax*((1-atbiNoVax)^(q-t))*a
      }
      p[q]=prob_atbi_in_year;
    }

    for (n in 1:N){
      personTime = timeNoVax[n];
      replicates = R[n];

      if( ATBIstatus[n]==1 ){
        target += replicates*log(p[personTime]);
      } else {
        survival=0;
```

```
      for(q in 1:personTime){
        survival=survival+p[q];
      }
      target += replicates*log(1-survival);
    }
  }
}
}

"

pomp::bake("results/noVaxATBI.RDS",{
  stan(model_code=stan_code,
             model_name="noVaxATBI.RDS",
             data=data,
             iter=2000, chains=4, init=0, seed=4)
}) -> fit
summary(fit)$summary
```

```
##                    mean      se_mean          sd          2.5%           25%
## ltbiNoVax  5.814597e-02 0.001954316 0.01497471  4.655418e-02  4.991536e-02
## atbiNoVax  8.948722e-02 0.002072485 0.01659243  4.984678e-02  8.687115e-02
## lp__      -1.315497e+05 0.060167291 1.03708044 -1.315522e+05 -1.315502e+05
##                     50%          75%          97.5%      n_eff      Rhat
## ltbiNoVax  5.239780e-02  5.636854e-02  1.003905e-01   58.71207 1.022357
## atbiNoVax  9.447974e-02  1.001260e-01  1.092432e-01   64.09692 1.017786
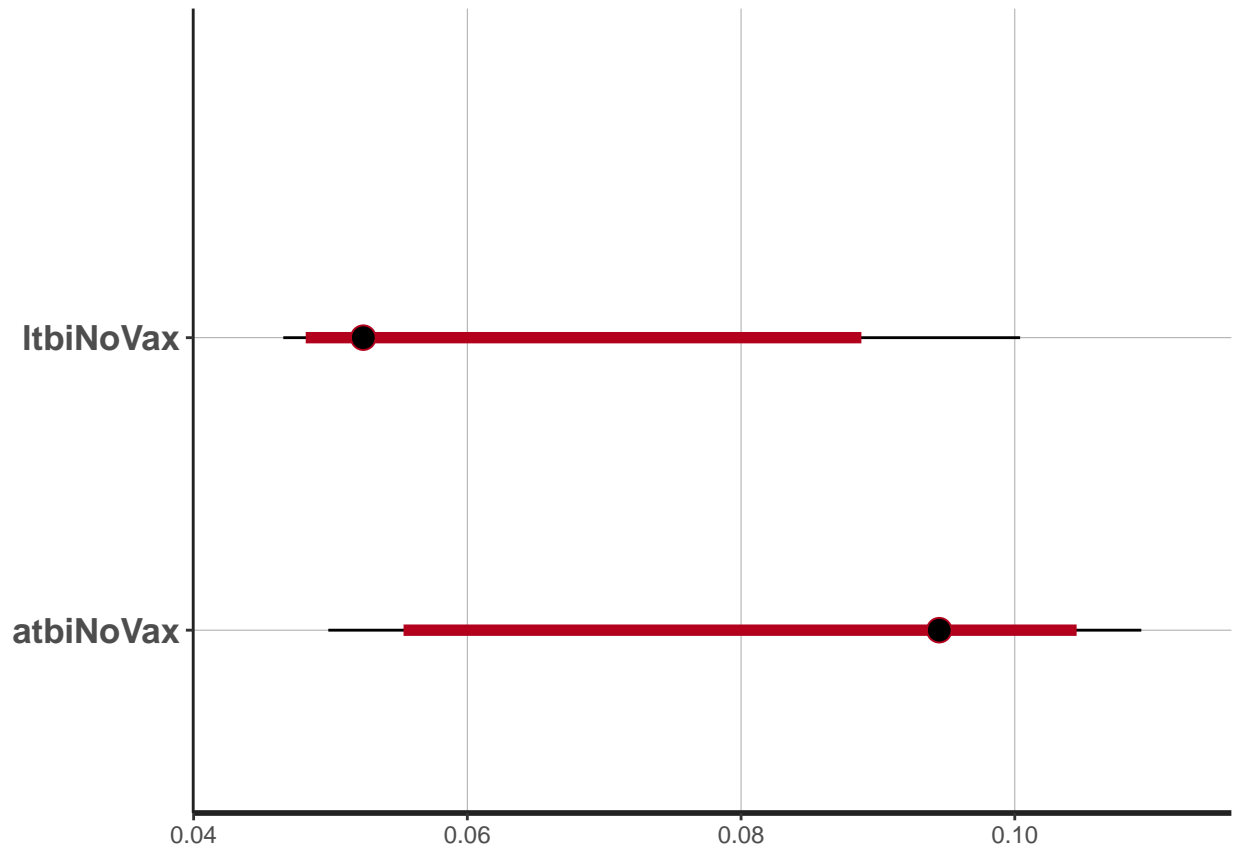## lp__      -1.315494e+05 -1.315489e+05 -1.315485e+05 297.10090 1.005172
```

```
plot(fit)
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```

This meets our expected output of `ltbiNoVax=0.05`