Basic SIR fitting

March 20, 2017

1 Harbin

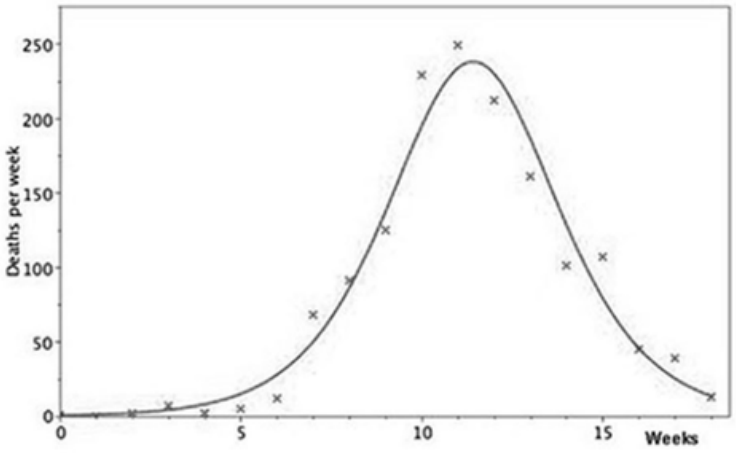


Figure 1: Unnumbered figure (p. 102) from Dietz (2009) showing the Harbin epidemic.

Figure 1 shows a Kermack-Mckendrick model fit to Harbin plague data. Based on the equations (1) and estimates ("$x_0 = 2985$, $\mathcal{R}_0 = 2.00$ and a mean infectious period of 11 days") that Dietz (2009) provides, we can compare how Kermack-Mckendrick model fit differs from SIR model fit based on maximum likelihood estimation.

$$\begin{aligned} \frac{dz}{dt} &= \frac{\gamma x_0}{2\mathcal{R}_0^2} c_1 \text{sech}^2(c_1\gamma t - c_2), \\ c_1 &= \sqrt{(\mathcal{R}_0 - 1)^2 + \frac{2\mathcal{R}_0^2}{x_0}} \\ c_2 &= \tanh^{-1}\left(\frac{\mathcal{R}_0 - 1}{c_1}\right). \end{aligned} \tag{1}$$

We note that the original equation provided by Dietz (2009) contains a typo. $c_1\gamma t$ after sech^2 in the first equation should be corrected to $c_1\gamma t/2$ (Kermack and McKendrick, 1927).

First, load the pacakge:

```
library(fitsir)
```

Since `fitsir` package lazy loads harbin data, `data(harbin)` syntax is unnecessary.

```
head(harbin)
```

```
##   week Deaths
## 1    2      2
## 2    3      7
## 3    4      2
## 4    5      6
## 5    6     12
## 6    7     68
```

Then, we transform the parameters provided by Dietz (2009) into *unconstrained parameters* (`log.beta, log.gamma, log.N, logit.i`) so that they can be used as starting parameters for MLE. Although `fitsir` expects a dataframe with column names `times` and `count`, we can specify a time column and a count column with `tcol` and `icol` arguments.

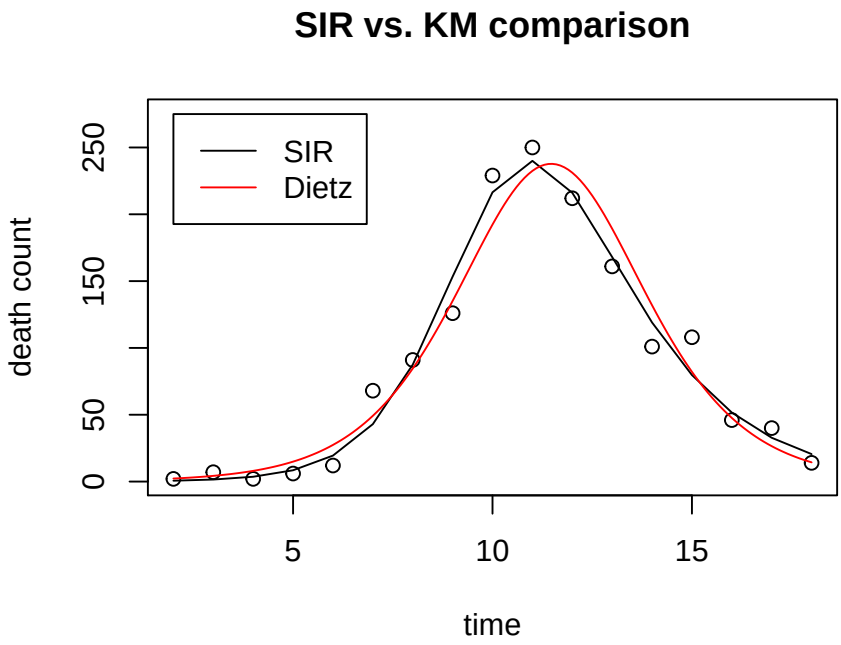
```
dietz_harbin <- c(x0=2985,rzero=2,gamma=7/11)
dietz_lpars <- with(as.list(dietz_harbin),
      c(log.beta=log(rzero*gamma),
        log.gamma=log(gamma),
        log.N=log(x0),
        logit.i=qlogis(1e-3)))
(ff <- fitsir(harbin, start=dietz_lpars, type="death",
              tcol="week", icol="Deaths", method="BFGS"))
```

```
##
## Call:
## mle2(minuslogl = objfun, start = start, method = method, data = dataarg,
##     vecpar = TRUE, gr = gradfun, control = control)
##
## Coefficients:
##   log.beta  log.gamma      log.N    logit.i
##  0.4868478 -0.2708639  7.4966582 -8.1274230
##
## Log-likelihood: -68.27
```

In this case, BFGS method has been used because using sensitivity equations allows for more accurate computation of the Hessian matrix.

We can plot `fitsir` objects using `plot` function to see whether this fit is good or not (`plot(ff)`). Here, we plot SIR fit along with Dietz fit to compare how they differ:

```
plot(ff, main="SIR vs. KM comparison")
times <- with(as.list(harbin), seq(min(week), max(week), by = 0.1))
dpKM <- with(as.list(dietz_harbin),
        {
           c1 <- sqrt((rzero-1)^2+2*rzero^2/x0)
           c2 <- atanh((rzero-1)/c1)
           gamma*x0/(2*rzero^2)*c1*
               (1/cosh(c1*gamma*times/2-c2))^2
        })
lines(times,dpKM, col = 2)
legend(x=2, y=275, legend=c("SIR","Dietz"), col=c("black", "red"), lty = 1)
```



Apart from the differences in the trajectories, we note that the Kermack-Mckendrick equation models the instantaneous change in the number of recovered individuals (dR/dt) whereas `fitsir` fits are based on the actual number of individuals that recovered during a given time interval ($R(\tau_{n+1}) - R(\tau_n)$).

We can also use the `summary` method provided by the `fitsir` package to see the summarized parameters:

```
summary(ff)

## Maximum likelihood estimation
##
## Call:
## mle2(minuslogl = objfun, start = start, method = method, data = dataarg,
##     vecpar = TRUE, gr = gradfun, control = control)
##
## Coefficients:
##                     R0          r     infper         i0         I0
## Estimate   2.1334e+00 8.6446e-01 1.3111e+00 2.9524e-04 5.3203e-01
## Std. Error 5.4710e-01 1.0344e-01 4.9281e-01 1.2155e-04 2.6501e-01
##                     S0       N
## Estimate   1.8015e+03 1802.01
## Std. Error 2.6259e+02  262.77
##
## -2 log L: 136.5431
```

MLE returns slightly higher $\mathcal{R}_0$ and longer infectious period but lower population size.

In fact, this is not the best fit. `fitsir` provides three ways of dealing with overdispersion (quasipossion, NB1, NB2) and in this case, using NB1 error function fits better (higher Log-likelihood) than using any of the provided error functions. First, to explore how these fits differ, we define a new data frame, namely `harbin2`, to avoid using `tcol` and `icol` arguments:

```
harbin2 <- setNames(harbin, c("times", "count"))
```

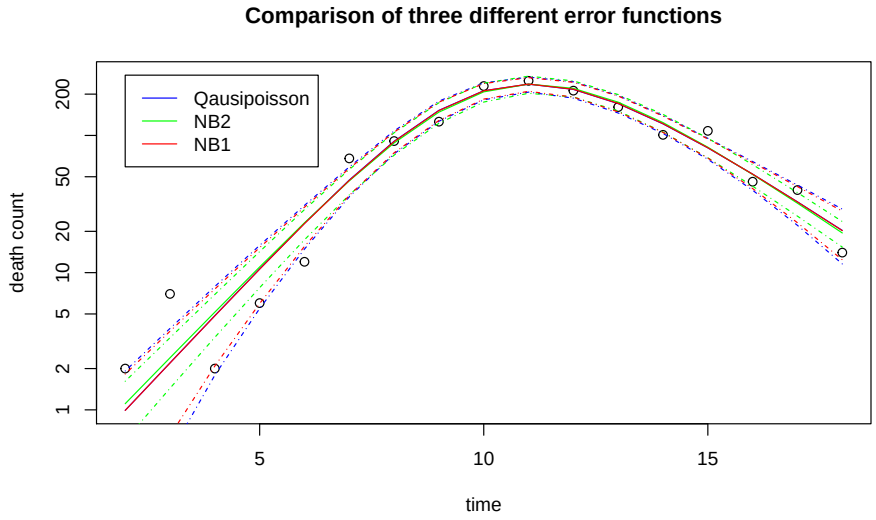
Then, we can fit:

```
ff2 <- fitsir(harbin2, dist="quasipoisson", type="death", method="BFGS")
ff3 <- fitsir(harbin2, dist="nbinom", type="death")
ff4 <- fitsir(harbin2, dist="nbinom1", type="death", hessian.opts=list(r=6))
```

For `nbinom1`, `hessian.opts=list(r=6)` was used because default Hessian calculation is not stable.

Again, we can plot these three fits to compare:

```
plot(ff2, level=0.95, col.traj="blue", col.conf="blue", log="y", main="Comparison of three d
plot(ff3, level=0.95, add=TRUE, col.traj="green", col.conf="green")
plot(ff4, level=0.95, add=TRUE, col.traj="red", col.conf="red")
legend(x=2, y=275, legend=c("Qausipoisson","NB2", "NB1"), col=c("blue", "green", "red"), lty
```

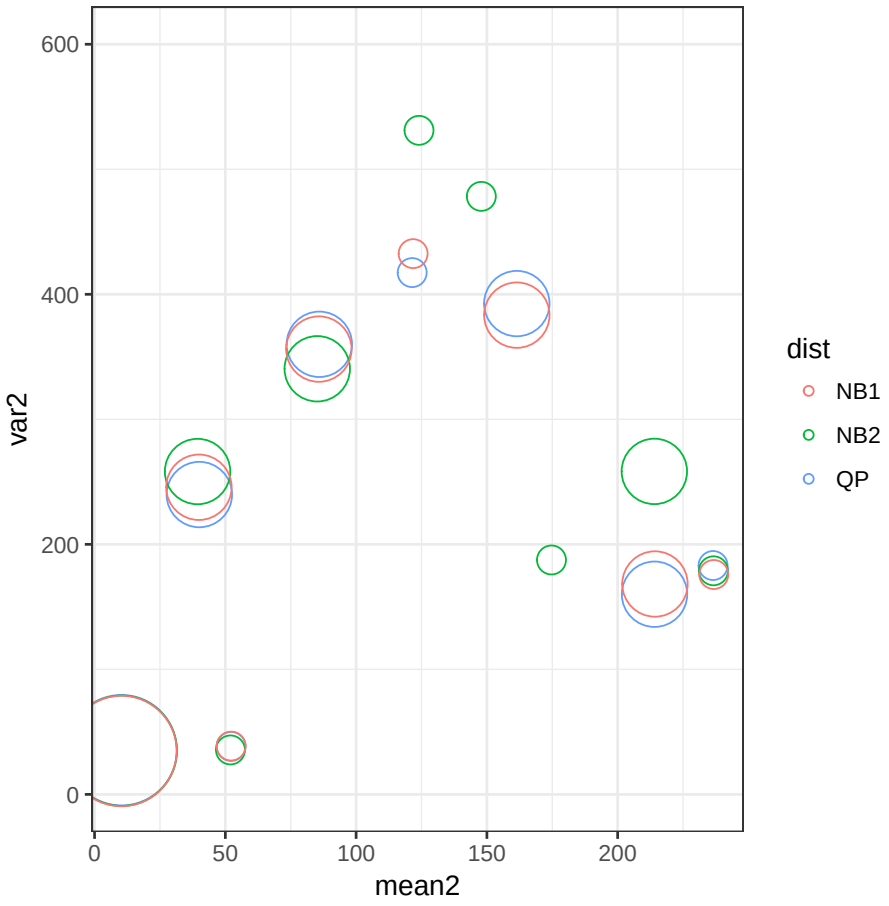



```
library(dplyr)
library(ggplot2); theme_set(theme_bw())
mvrel <- function(fit, data) {
    mean <- SIR.detsim(data$times, coef(fit, "trans"), type="death")
    data.frame(
        mean=mean,
        var=(data$count-mean)^2
    )
}
level <- seq(0, 300, by = 25)

mvfun <- . %>%
    mvrel(harbin2) %>%
    mutate(group=cut(mean, breaks=level)) %>%
    group_by(group) %>%
    summarise(mean2 = mean(mean), var2=mean(var), n=length(var))

mvtot <- list(QP=mvfun(ff2), NB2=mvfun(ff3), NB1=mvfun(ff4)) %>%
    bind_rows(.id="dist")

ggplot(mvtot, aes(mean2, var2)) +
    geom_point(aes(size=n, col=dist), pch=1) +
    scale_y_continuous(lim=c(0, 600)) +
    scale_size_continuous(range = c(5, 20), guide=FALSE)
```



References

Dietz, K. (2009, April). Epidemics: the fitting of the first dynamic models to data. *Journal of Contemporary Mathematical Analysis 44*(2), 97–104.

Kermack, W. O. and A. G. McKendrick (1927). A contribution to the mathematical theory of epidemics. In *Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences*, Volume 115, pp. 700–721. The Royal Society.