rmd_notebook

2022-10-16

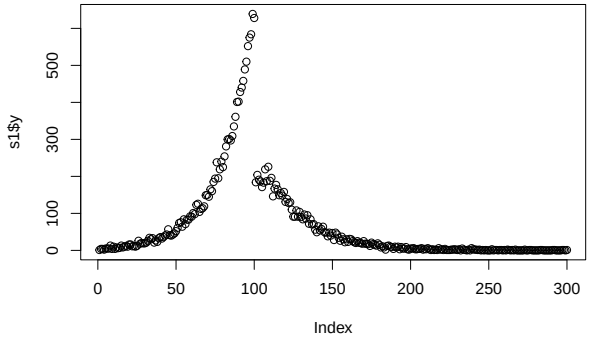
```
library(tidyverse)
library(dplyr)
library(EpiEstim)
library(zoo)
source("constant/constant.R")
```

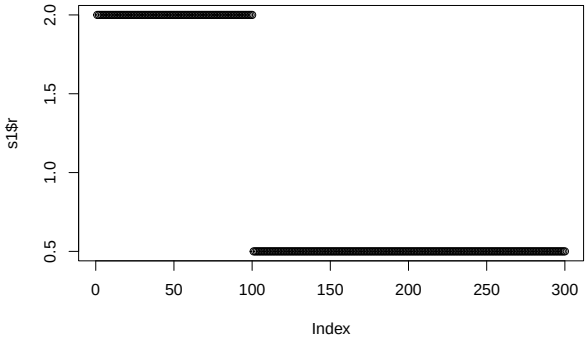
Table of content

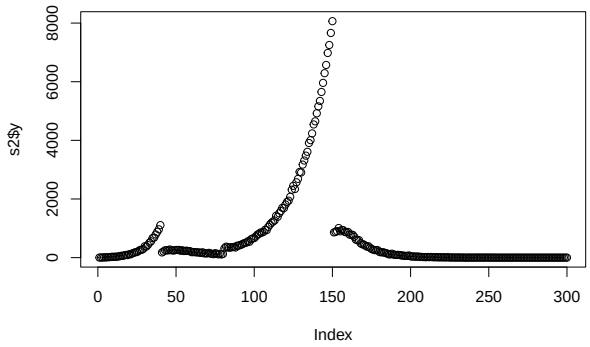
Synthetic datasets - Part 1

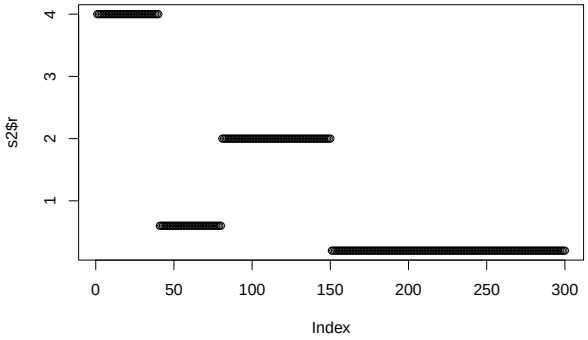
1. 2A, 2B from Epifilter, R_t are step functions.

Synthetic datasets are generated by first specify R_t, then generating cases using renewal equations.





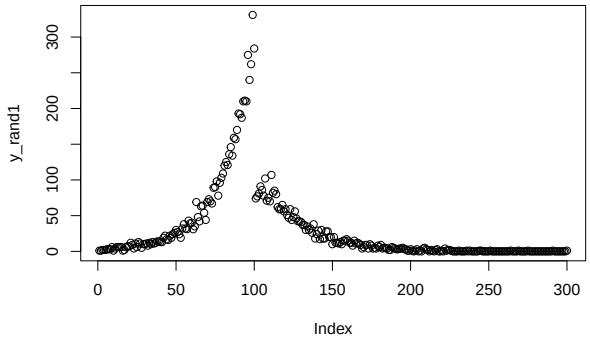


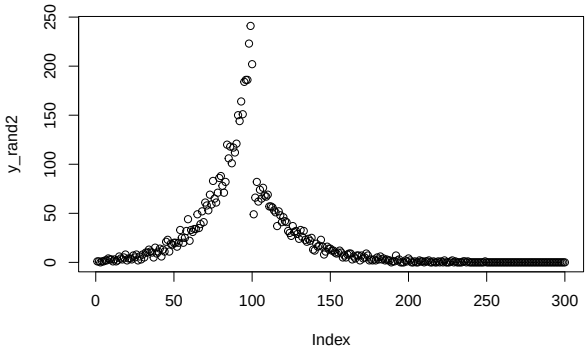


Discussion

Question 1: data generated resembles each other, but differ by magnitude

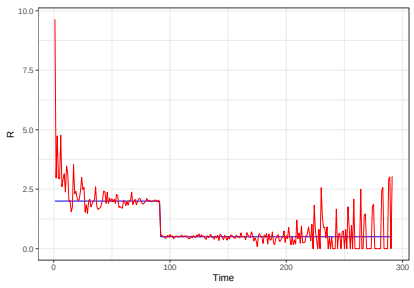
Answer 1: randomness accumulated, an increase in count at day t, makes a larger value possible for day t+1. Maybe the renewal equation is too flexible

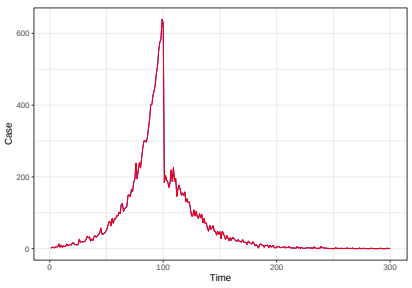


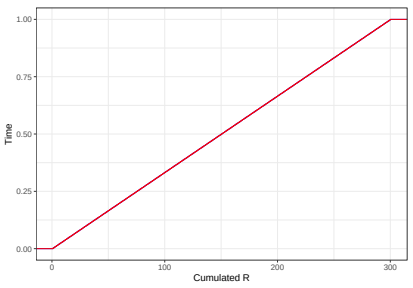


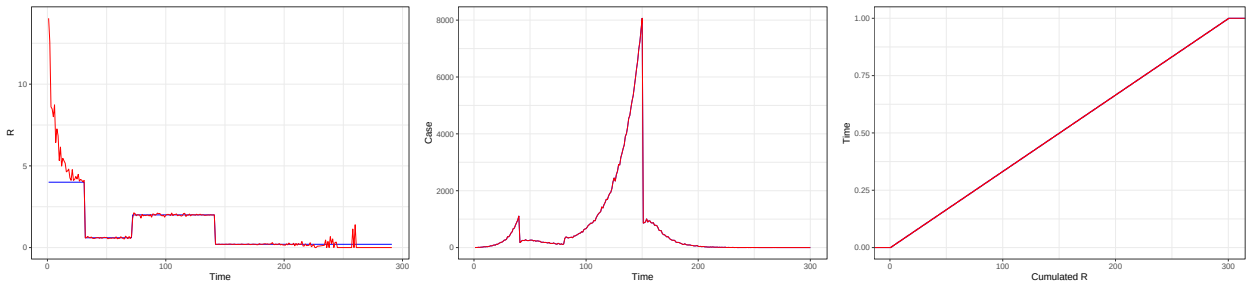
MLE estimate

Let $I_t \sim Poisson(\lambda_t), \lambda_t = \sum_{a=1}^{t} I_{t-a} w_a$, then the MLE estimate $R_{MLE} = \frac{I_t}{\lambda_t}$.









Discussion

Using R_{MLE} results in a perfect one-day ahead prediction. However, comparing with the true R_t, R_{MLE} is not smooth, especially in the beginning and at the end. In real life, it is impossible to see R_t changes this much on a daily basis. Two things we could do to avoid this

1. Smooth R_t when estimating. This is what are focusing on. The question that comes with doing this is how much to smooth R_t
2. Smooth I_t by 1) taking rolling average, 2) deconvolving it with other distributions. Lots of methods deconvolves case counts with reporting delay and incubation period. This is more difficult to achieve compare to 1. above because it needs a reporting delay and incubation period distribution.

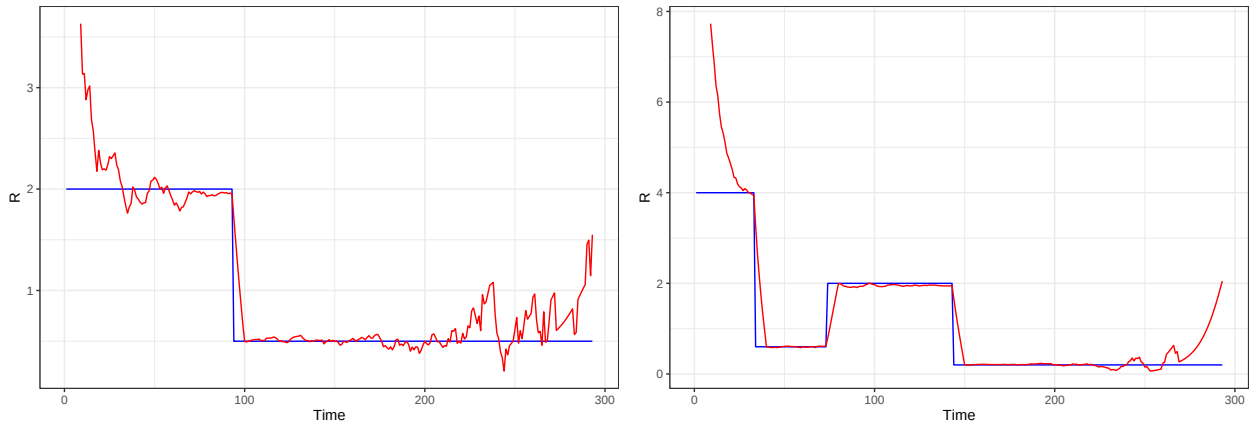
EpiEstim

EpiEstim uses Bayes rule

$$P(R_1^t|I_1^t, w) = P(I_1^t|R_1^t, w_t)P(R_1^t)$$

where $I_1^t|R_1^t, w_t \sim Pois(R_t \sum_{a=1}^{t} I_{t-a} w_a)$, and R_1^t $Gamma(a, b)$. Note here that R's are independent from each other, and has common parameters a, b. In practice, a, b are pre-specified

```
## Warning: Removed 8 row(s) containing missing values (geom_path).
## Removed 8 row(s) containing missing values (geom_path).
```



Discussion

We see that EpiEstim is estimating R_t way better than the MLE estimate in the beginning and the end of the true R_t. However, it is slightly worse than the MLE estimate in the overall fit.

EpiEstim represents dependency between dates by assigning a R_t value for a period of time, as opposed to one for each day. This results in a delay of estimation (Luis et al.), meaning that the model responses slower with changes of R_t.

TODO

EpiEstim allows uncertainty in serial interval distribution. This is done by providing a variance for the mean and sd of the serial interval distribution. Then 1000 pairs of (mean, sd), and are sampled and are run.

Penalized Least Square - Part 1

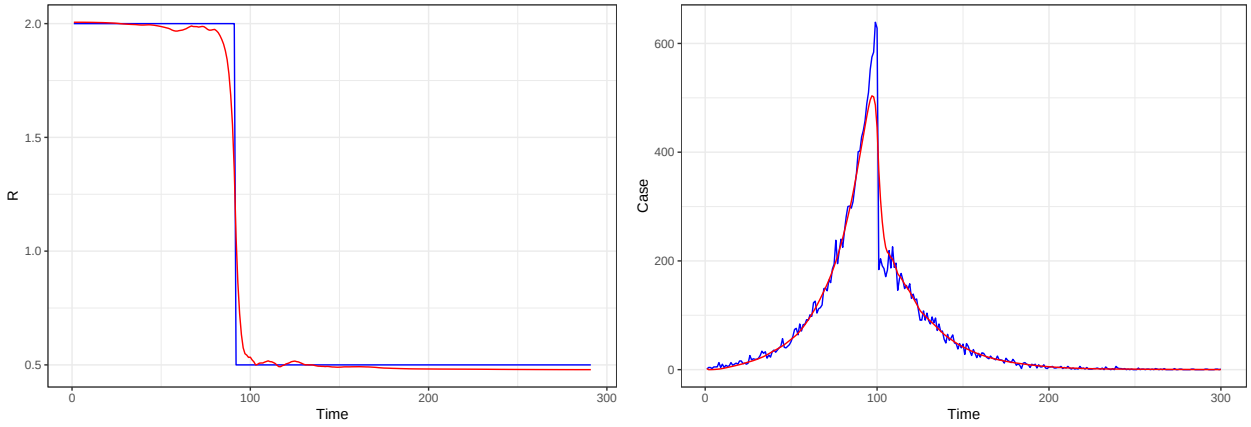
As mentioned above, we could smooth R_t by running penalized least squares by minimizing the equation below

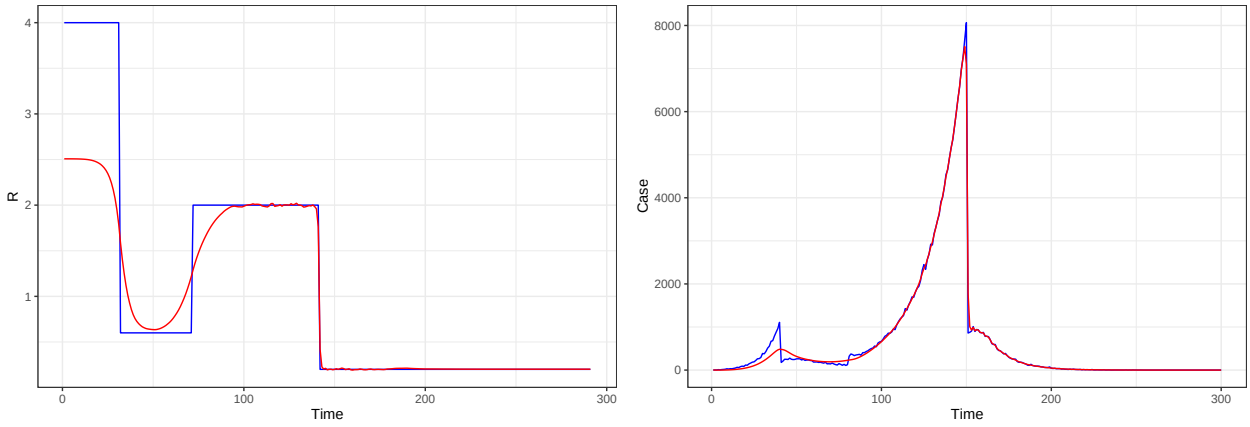
$$Loss(I_t - R_t * \sum_{a=1}^{t} I_{t-a} w_a) + \lambda * Penalty(R_t)$$

Here, the loss function is chosen to be the probability of observing I_t given the mean of the Poisson distribution is $R_t * \sum_{a=1}^{t} I_{t-a} w_a$. The simplest penalty to smooth R_t is the L-2 Norm. Under this configuration, the objective function is smooth and any optimization methods should work great

```
## [1] "Optimizing for s1 takes 1.5199830532074"
```

```
## [1] "Optimizing for s1 takes 1.44440793991089"
```





TODO

Penalty term should be chosen with cross-validation.

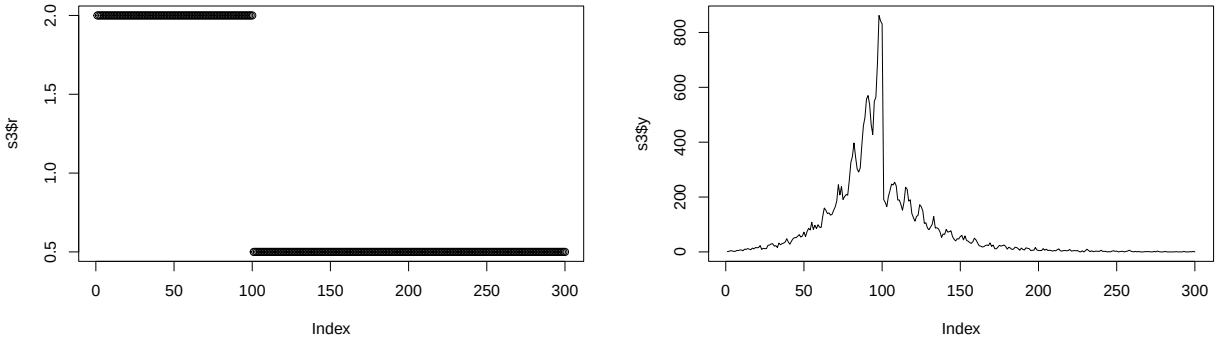
Synthetic datasets - Part 2

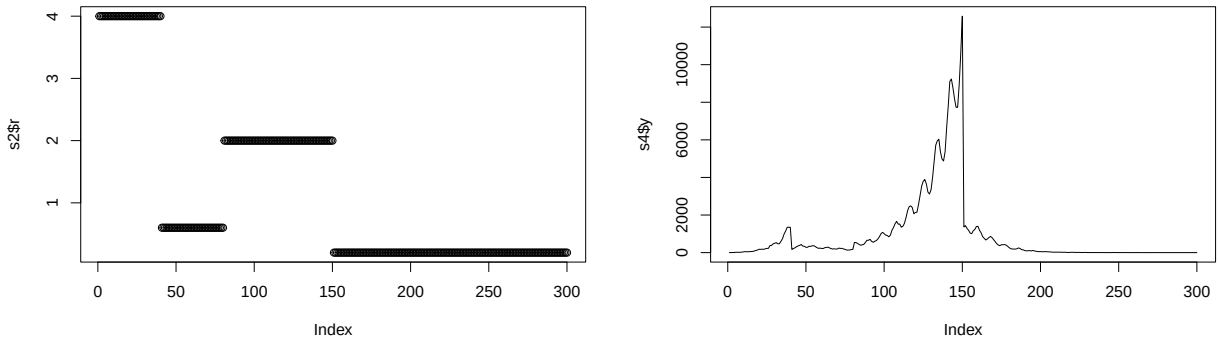
Now we see that a simple penalized least square is good enough for the synthetic dataset generated in the beginning. Here I will add two more features to make it more complicated

1. Cyclic effect: generate data with cyclic effect. TODO: add cyclic effect to a given data (same shape different magnitude, why? show for help)

For now, the cyclic effect is added based on the equation $ceiling(max(0, \frac{i[j]}{5} * sin(7j) + i[j]))$, with $i[j]$ being the case count at day j

2. Outliers that fall outside of 2 sd away from the predicted case counts



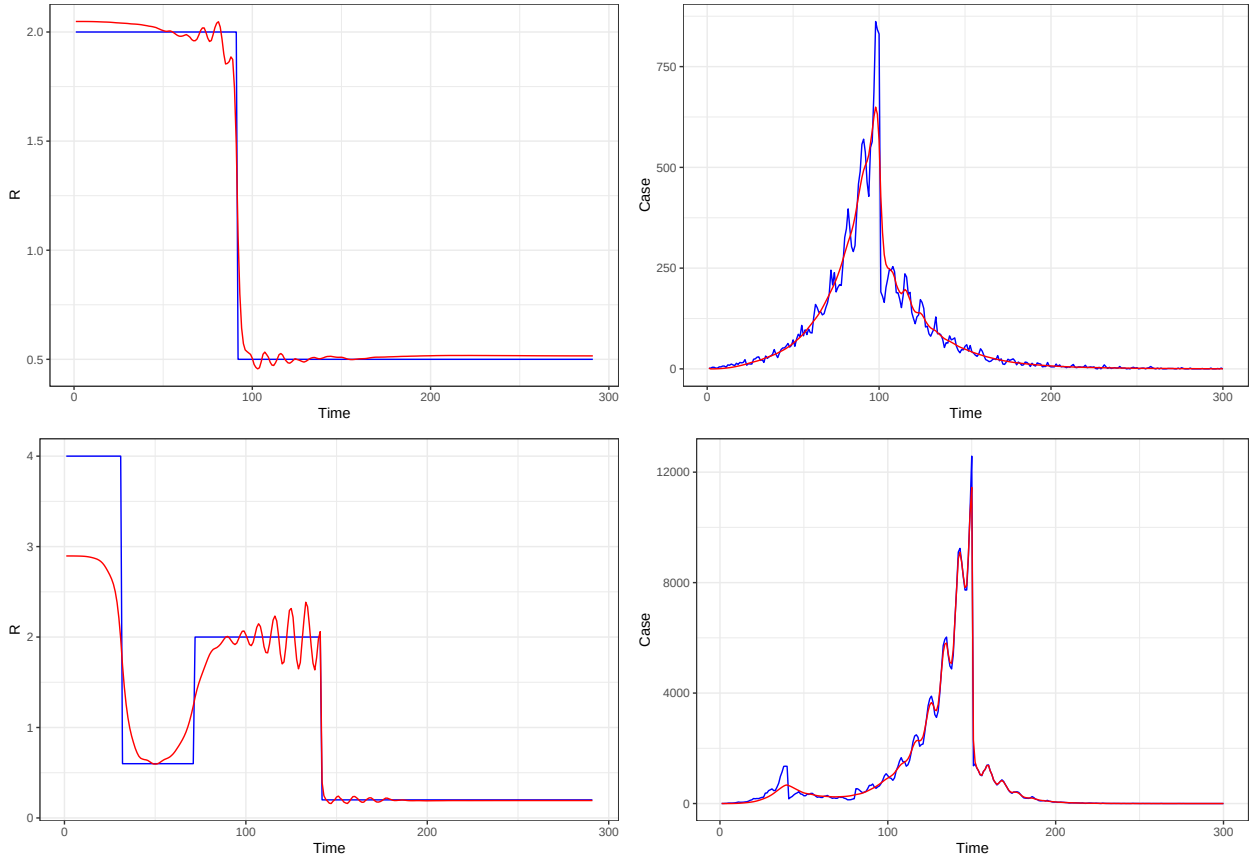


Now to see the fit of PLS with this newly generated data

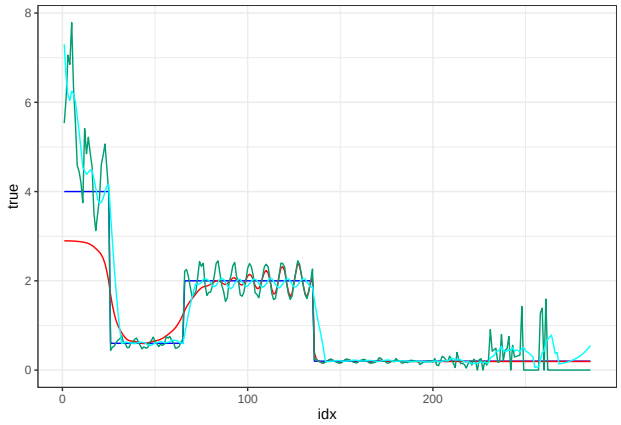
```
## [1] "Optimizing for s3 takes 1.2936840057373"
```

```
## [1] "Optimizing for s4 takes 1.25042796134949"
```

```
## Default config will estimate R on weekly sliding windows.
##     To change this change the t_start and t_end arguments.
```



It doesn't seem very good. Now to have all three methods together side by side. Blue, true R_t, red, R_t estimated using PLS, R_t, the MLE estimate



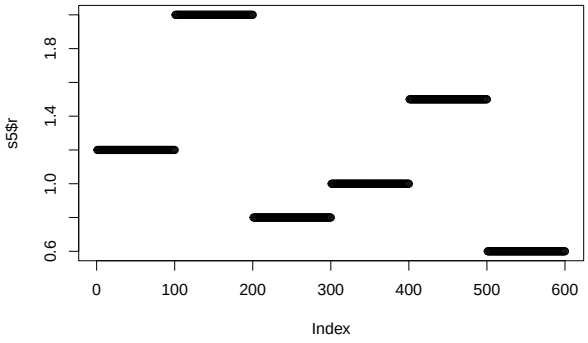
Discussion

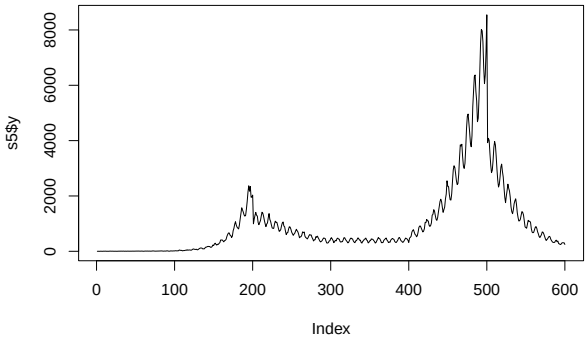
For both papers on PLS, (Luis et al. EpiInvert, and Pascal et al. nonsmooth PLS), they only uses the covid case, with MLE as baseline. They show that PLS is a smoother estimate of MLE. Both PLS and MLE methods underperforms in the beginning, but PLS shows correct estimates at the end of the pandemic.

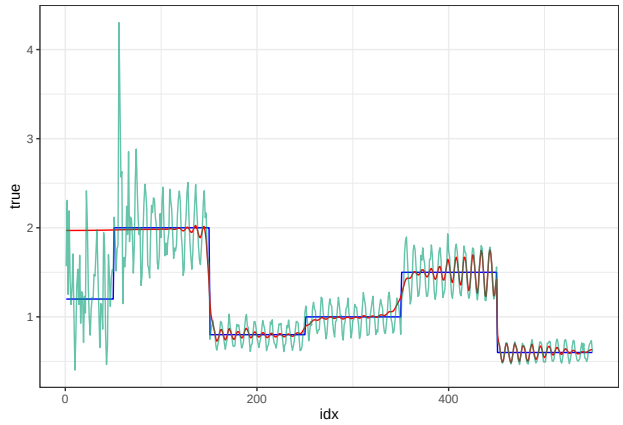
```
## Default config will estimate R on weekly sliding windows.
##     To change this change the t_start and t_end arguments.
```

```
## Warning in estimate_R_func(incid = incid, method = method, si_sample = si_sample, : You're estimating
##             posterior CV.
```

```
## [1] "Optimizing for s5 takes 4.73461413383484"
```







The added cyclic effect is the following. I allow the amplitude to change also with the number of case count, which should be logical in a real world setting.

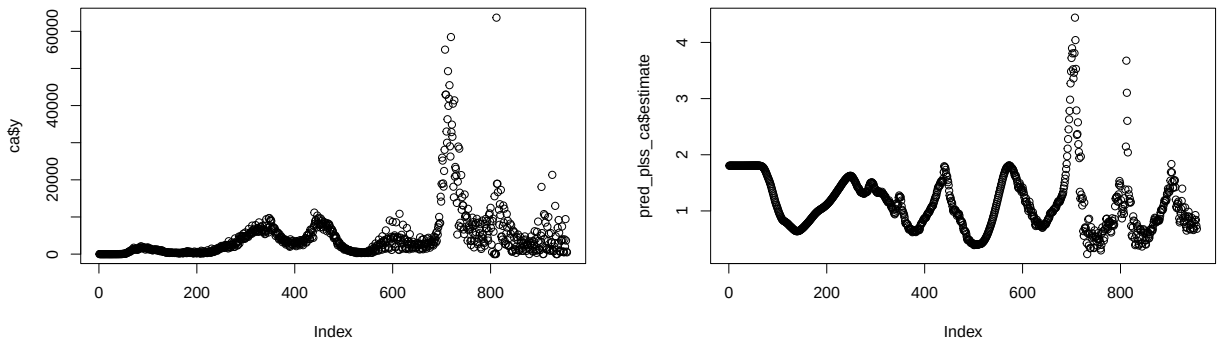
$$ceiling(max(0, \frac{i[j]}{5} * sin(7j) + i[j]))$$

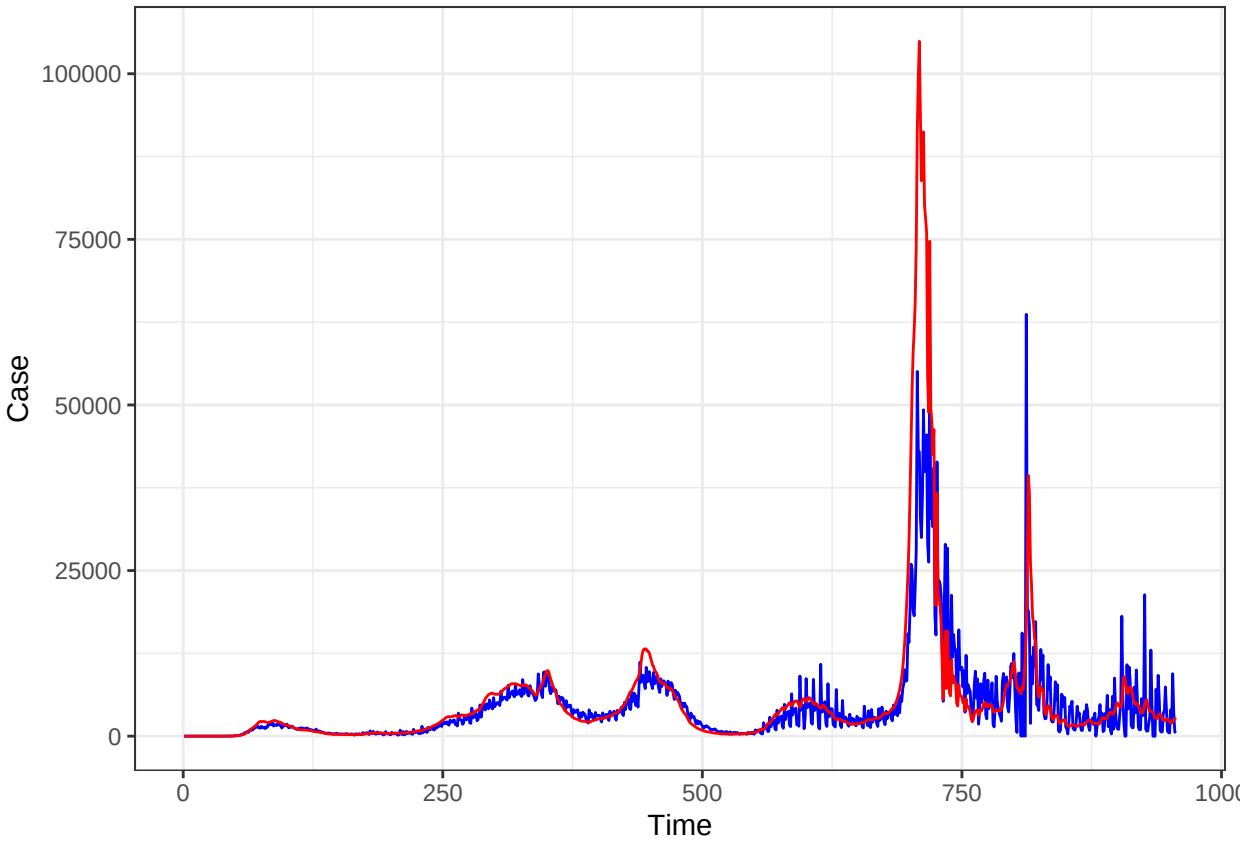
Methods on covid data in Canada

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## [1] "Optimizing for ca takes 10.0734570026398"
```





Penalized Least Square - Part 2

Notice the fit to the Canadian data above.

Outliers:

There is an outlier point at day around 800 that is way above its neighbors. Pascal et al. model the outliers by model it in the renewal models, i.e., $I_t \sim Pois(R_t * \sum_{a=1}^{t} I_{t-a} w_a + o_t)$, and later penalize the size of the outliers using L1 penalty. With sudden outliers like this, it is more justified to use Pascal et al.'s method to model the outliers

Smoothness:

1. We have applied one single penalty to the model $\lambda_r = 30$ and saw that it counters the weekly trend really well around day 600. However, the penalty is soon to be not enough for the peak case at 700, and the period around day 730. This happens when the magnitude is too big, or the weekly trend effect is too big.
2. Penalty of the smoothness is by an L-2 penalty, which doesn't encourage the consecutive R_t to be exactly the same.

Specialized ADMM algorithm by Ramdas and Tibshirani linked by Daniel from last email solves problems of the following form

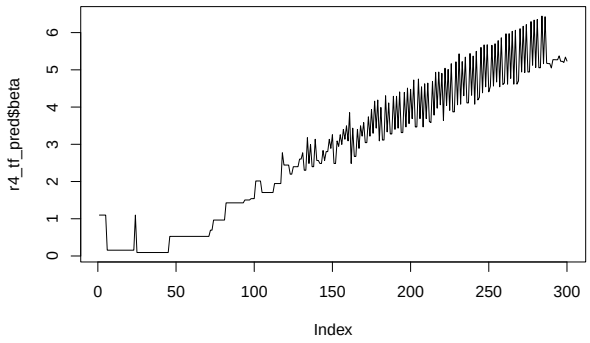
$$\hat{\beta} = argmin_{\beta \in R} \frac{1}{2}(y - X\beta)^2 + \lambda |D^{k+1}\beta|$$

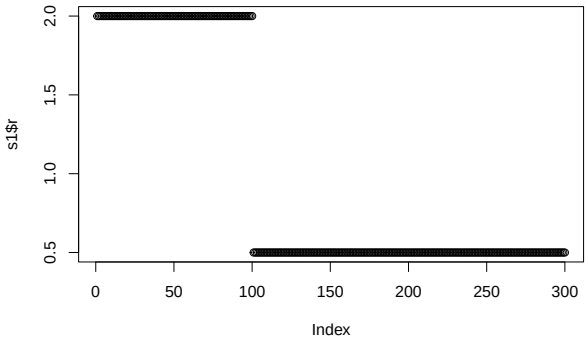
where D is the difference matrix, and k is the number of differences to take. Running R code gives the following estimation of R_t. Left hand side is the predicted R_t, right hand side is the actual R_t

```
library(glmgen)

r4_tf_pred = trendfilter(x=s1_iwt, y = s1$y, k = 0, lambda=5, family="poisson")
plot(r4_tf_pred$beta, type="l")

plot(s1$r)
```





TODO

1. Make penalty depends on case count
2. Either make specialized ADMM work, or implement proximal gradient with line search

Discussion on Serial Interval Distribution

Serial interval distribution is an estimate of the generation interval distribution. Serial interval distribution is the delay between the onset of primary and secondary cases, where the generation interval distribution is the infection of the primary and secondary cases.

Serial interval distribution should change in time. This could to be done either 1) through constant contact tracing in real life, or 2) adding to the uncertainty of estimating R_t. Both are difficult to do.

Both EpiEstim and EpiNow2 allow uncertainty on Serial Interval Distribution. EpiNow2 does this by having a prior on its parametrization, EpiEstim does this by drawing samples from the sid uncertainty.

If we have the information of sid across time, we can have the parametrization of sid vary on a Markov chain.

Discussion: Frequentist vs Bayesian approach

1. Daniel said: If only cares about the predicted mean, why use so much computation power to calculate the full posterior?

- We could try faster Bayesian methods, such as sequential Monte Carlo, or turn into penalized least squares.

2. Smoothness:

- PLS add smoothness by putting more penalty on the smoothing term
- Bayesian methods add smoothness by giving a smaller variance for the transition probability (or a smaller prior for the variance)

3. Uncertainty of serial interval distribution

- Sample from the SID distribution, then estimate R_t, then imposing the uncertainty on top.
- Bayesian: Hierarchical Bayes, give prior to sid

4. Filtering vs Smoothing

- PLS is a smoothing method, as it uses data from day $1...T$ to estimate R_t at day $t, t < T$.
- Bayesian methods have the flexibility to do filtering, therefore achieving a more real-time estimation.

5. Outlier

- PLS by Pascal et al. models the outlier directly.
- DARt has another variable M, which allow smc to sample from a broader prior if M=1. It additionally model

$p(R_t|R_{t-1}, M_t) \sim N(R_{t-1}, \sigma_R^2)$ if $M_t = 0$
$p(R_t|R_{t-1}, M_t) \sim U[0, R_{t-1} + \Delta]$ if $M_t = 1$

6. What about variational Bayes methods, it turns Bayesian space exploration (posterior) style algorithm to optimization algorithm.

Big TODO

1. Review "Stochastic Variational Inference for Bayesian Time Series Models" by Johnson and Willsky.