Simple model for Lombardia

Laura Balasso

10/24/2020

Model for Lombardia from august

```
data_lombardia <- get_model_data(data_it, 'Lombardia', initial_date = as.Date('2020-07-30'))

p_delay <- get_delay_distribution()

nonzero_days_l <- which(data_lombardia$total != 0)

stan_data_l <- list(N = nrow(data_lombardia),
                    conv_gt = get_gt_convolution(nrow(data_lombardia)),
                    length_delay = length(p_delay),
                    p_delay = p_delay,
                    exposures = exposures_from_total(data_lombardia$total),
                    N_nonzero = length(nonzero_days_l),
                    nonzero_positives = data_lombardia$positive[nonzero_days_l],
                    nonzero_days = nonzero_days_l
)


compiled_model <- stan_model('rt_model.stan')
```

```
## Trying to compile a simple C file
```

```
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -mmacosx-version-min=10.13 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG   -I
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/incl
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util/
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util/
## namespace Eigen {
##                 ^
##                 ;
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/incl
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/Core:96:10: fa
## #include <complex>
##          ^~~~~~~~
## 3 errors generated.
```

```
## make: *** [foo.o] Error 1
```

```
fit_model_lomb <- sampling(compiled_model, data=stan_data_l, iter = 2000)
```

```
##
## SAMPLING FOR MODEL 'rt_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.00135 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 13.5 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 59.6069 seconds (Warm-up)
## Chain 1:                62.2776 seconds (Sampling)
## Chain 1:                121.884 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'rt_model' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0.000665 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 6.65 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 59.7916 seconds (Warm-up)
## Chain 2:                52.8263 seconds (Sampling)
## Chain 2:                112.618 seconds (Total)
## Chain 2:
##
```

```
## SAMPLING FOR MODEL 'rt_model' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0.000661 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 6.61 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 60.5923 seconds (Warm-up)
## Chain 3:                58.0508 seconds (Sampling)
## Chain 3:                118.643 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'rt_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0.000773 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 7.73 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 60.7073 seconds (Warm-up)
## Chain 4:                55.6345 seconds (Sampling)
## Chain 4:                116.342 seconds (Total)
## Chain 4:
```

```
print(fit_model_lomb, pars = 'r_t')
```

```
## Inference for Stan model: rt_model.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
```

```
##
##          mean se_mean   sd 2.5%  25%  50%  75% 97.5% n_eff Rhat
## r_t[1]  1.13    0.00 0.10 0.95 1.06 1.13 1.20  1.33  3081    1
## r_t[2]  1.13    0.00 0.09 0.96 1.06 1.13 1.19  1.32  2883    1
## r_t[3]  1.13    0.00 0.09 0.96 1.07 1.12 1.19  1.31  2743    1
## r_t[4]  1.13    0.00 0.08 0.97 1.07 1.13 1.18  1.30  2731    1
## r_t[5]  1.13    0.00 0.08 0.97 1.07 1.12 1.18  1.29  2802    1
## r_t[6]  1.12    0.00 0.08 0.98 1.07 1.12 1.17  1.27  2566    1
## r_t[7]  1.12    0.00 0.07 0.99 1.07 1.12 1.17  1.27  2715    1
## r_t[8]  1.12    0.00 0.07 0.99 1.07 1.12 1.16  1.25  2833    1
## r_t[9]  1.12    0.00 0.07 0.99 1.07 1.11 1.16  1.25  2962    1
## r_t[10] 1.11    0.00 0.06 0.99 1.07 1.11 1.16  1.24  2936    1
## r_t[11] 1.11    0.00 0.06 0.99 1.07 1.11 1.15  1.24  3042    1
## r_t[12] 1.11    0.00 0.06 1.00 1.07 1.11 1.15  1.23  3203    1
## r_t[13] 1.11    0.00 0.06 1.00 1.07 1.11 1.15  1.23  2958    1
## r_t[14] 1.11    0.00 0.06 1.00 1.07 1.10 1.15  1.23  2922    1
## r_t[15] 1.10    0.00 0.06 0.99 1.06 1.10 1.14  1.23  3592    1
## r_t[16] 1.10    0.00 0.06 0.99 1.06 1.10 1.14  1.22  3548    1
## r_t[17] 1.10    0.00 0.06 0.98 1.05 1.09 1.14  1.22  3503    1
## r_t[18] 1.09    0.00 0.06 0.98 1.05 1.09 1.13  1.22  3490    1
## r_t[19] 1.09    0.00 0.06 0.97 1.05 1.09 1.13  1.21  3367    1
## r_t[20] 1.08    0.00 0.06 0.97 1.04 1.08 1.12  1.20  3442    1
## r_t[21] 1.08    0.00 0.06 0.97 1.04 1.07 1.12  1.19  3323    1
## r_t[22] 1.07    0.00 0.06 0.96 1.03 1.07 1.11  1.19  3179    1
## r_t[23] 1.06    0.00 0.06 0.95 1.02 1.06 1.10  1.18  3133    1
## r_t[24] 1.05    0.00 0.06 0.94 1.01 1.05 1.09  1.17  2818    1
## r_t[25] 1.04    0.00 0.06 0.93 1.00 1.04 1.08  1.16  3285    1
## r_t[26] 1.03    0.00 0.06 0.92 0.99 1.03 1.06  1.14  2927    1
## r_t[27] 1.01    0.00 0.06 0.91 0.98 1.01 1.05  1.13  2949    1
## r_t[28] 1.00    0.00 0.05 0.90 0.97 1.00 1.04  1.12  2993    1
## r_t[29] 0.99    0.00 0.05 0.89 0.95 0.99 1.03  1.10  3095    1
## r_t[30] 0.98    0.00 0.05 0.88 0.95 0.98 1.02  1.09  2807    1
## r_t[31] 0.97    0.00 0.05 0.87 0.93 0.97 1.00  1.08  3019    1
## r_t[32] 0.96    0.00 0.05 0.86 0.92 0.96 0.99  1.06  3060    1
## r_t[33] 0.95    0.00 0.05 0.86 0.92 0.95 0.98  1.05  3238    1
## r_t[34] 0.94    0.00 0.05 0.85 0.91 0.94 0.98  1.04  3247    1
## r_t[35] 0.94    0.00 0.05 0.84 0.90 0.93 0.97  1.03  3397    1
## r_t[36] 0.93    0.00 0.05 0.84 0.90 0.93 0.96  1.03  3307    1
## r_t[37] 0.93    0.00 0.05 0.83 0.89 0.93 0.96  1.03  3113    1
## r_t[38] 0.92    0.00 0.05 0.83 0.89 0.92 0.96  1.02  2945    1
## r_t[39] 0.92    0.00 0.05 0.83 0.89 0.92 0.95  1.03  2485    1
## r_t[40] 0.92    0.00 0.05 0.82 0.89 0.92 0.95  1.02  3030    1
## r_t[41] 0.92    0.00 0.05 0.82 0.89 0.92 0.96  1.03  2960    1
## r_t[42] 0.92    0.00 0.05 0.82 0.89 0.92 0.96  1.02  2988    1
## r_t[43] 0.93    0.00 0.05 0.83 0.89 0.93 0.96  1.03  2994    1
## r_t[44] 0.94    0.00 0.05 0.83 0.90 0.93 0.97  1.04  2920    1
## r_t[45] 0.94    0.00 0.05 0.85 0.91 0.94 0.98  1.05  2868    1
## r_t[46] 0.95    0.00 0.05 0.85 0.92 0.95 0.99  1.06  3109    1
## r_t[47] 0.97    0.00 0.05 0.87 0.93 0.97 1.00  1.08  2966    1
## r_t[48] 0.98    0.00 0.05 0.88 0.95 0.98 1.02  1.10  3047    1
## r_t[49] 1.00    0.00 0.05 0.90 0.97 1.00 1.04  1.12  2678    1
## r_t[50] 1.03    0.00 0.06 0.92 0.99 1.02 1.06  1.14  2991    1
## r_t[51] 1.05    0.00 0.06 0.94 1.01 1.05 1.09  1.16  2740    1
## r_t[52] 1.08    0.00 0.06 0.97 1.04 1.08 1.12  1.20  2879    1
```

```
## r_t[53] 1.11    0.00 0.06 1.00 1.07 1.11 1.15  1.23  3223    1
## r_t[54] 1.14    0.00 0.06 1.03 1.10 1.14 1.18  1.27  3411    1
## r_t[55] 1.18    0.00 0.06 1.06 1.14 1.18 1.22  1.31  3245    1
## r_t[56] 1.22    0.00 0.06 1.10 1.17 1.22 1.26  1.35  3181    1
## r_t[57] 1.26    0.00 0.07 1.13 1.21 1.26 1.30  1.39  3126    1
## r_t[58] 1.30    0.00 0.07 1.17 1.25 1.30 1.34  1.44  3156    1
## r_t[59] 1.34    0.00 0.07 1.21 1.29 1.34 1.39  1.49  2986    1
## r_t[60] 1.38    0.00 0.07 1.24 1.33 1.38 1.43  1.53  3108    1
## r_t[61] 1.42    0.00 0.08 1.28 1.37 1.42 1.47  1.58  3073    1
## r_t[62] 1.46    0.00 0.08 1.31 1.41 1.46 1.52  1.62  2877    1
## r_t[63] 1.50    0.00 0.08 1.35 1.45 1.50 1.56  1.67  2813    1
## r_t[64] 1.54    0.00 0.08 1.38 1.48 1.54 1.60  1.70  3035    1
## r_t[65] 1.57    0.00 0.09 1.41 1.51 1.57 1.63  1.75  3029    1
## r_t[66] 1.60    0.00 0.09 1.44 1.54 1.60 1.66  1.78  3077    1
## r_t[67] 1.63    0.00 0.09 1.45 1.57 1.63 1.69  1.80  3074    1
## r_t[68] 1.65    0.00 0.09 1.48 1.59 1.65 1.71  1.82  3125    1
## r_t[69] 1.66    0.00 0.09 1.49 1.60 1.66 1.72  1.84  3398    1
## r_t[70] 1.67    0.00 0.09 1.50 1.61 1.67 1.73  1.85  3700    1
## r_t[71] 1.68    0.00 0.09 1.50 1.61 1.67 1.74  1.86  3642    1
## r_t[72] 1.68    0.00 0.09 1.50 1.62 1.67 1.74  1.87  3652    1
## r_t[73] 1.67    0.00 0.10 1.49 1.61 1.67 1.73  1.87  3188    1
## r_t[74] 1.67    0.00 0.10 1.49 1.60 1.67 1.73  1.86  2936    1
## r_t[75] 1.66    0.00 0.10 1.47 1.60 1.66 1.73  1.86  2739    1
## r_t[76] 1.66    0.00 0.10 1.46 1.58 1.65 1.72  1.87  2799    1
## r_t[77] 1.65    0.00 0.11 1.44 1.57 1.65 1.73  1.88  2578    1
## r_t[78] 1.64    0.00 0.12 1.42 1.56 1.64 1.72  1.89  2455    1
## r_t[79] 1.64    0.00 0.13 1.40 1.55 1.64 1.73  1.90  2220    1
## r_t[80] 1.63    0.00 0.14 1.38 1.54 1.63 1.72  1.91  2121    1
## r_t[81] 1.63    0.00 0.15 1.35 1.53 1.63 1.73  1.93  2008    1
## r_t[82] 1.63    0.00 0.16 1.33 1.52 1.63 1.74  1.95  1949    1
## r_t[83] 1.63    0.00 0.16 1.31 1.52 1.63 1.74  1.97  1864    1
## r_t[84] 1.63    0.00 0.17 1.30 1.51 1.62 1.75  1.99  1845    1
## r_t[85] 1.63    0.00 0.18 1.28 1.50 1.62 1.75  2.00  1832    1
## r_t[86] 1.63    0.00 0.19 1.26 1.50 1.62 1.75  2.02  1844    1
## r_t[87] 1.63    0.00 0.20 1.26 1.49 1.62 1.76  2.04  1870    1
## r_t[88] 1.63    0.00 0.21 1.24 1.49 1.62 1.77  2.08  1881    1
## r_t[89] 1.63    0.00 0.22 1.22 1.48 1.62 1.77  2.10  1909    1
## r_t[90] 1.63    0.01 0.22 1.21 1.48 1.62 1.78  2.11  1923    1
## r_t[91] 1.63    0.01 0.23 1.21 1.47 1.63 1.79  2.11  1971    1
## r_t[92] 1.64    0.01 0.24 1.20 1.47 1.62 1.79  2.13  1955    1
## r_t[93] 1.64    0.01 0.25 1.19 1.47 1.63 1.80  2.15  2015    1
## r_t[94] 1.64    0.01 0.25 1.18 1.46 1.62 1.80  2.18  2022    1
## r_t[95] 1.64    0.01 0.26 1.17 1.46 1.62 1.80  2.20  2093    1
##
## Samples were drawn using NUTS(diag_e) at Sat Oct 24 20:29:52 2020.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
print(fit_model_lomb, pars = 'eta')
```

```
## Inference for Stan model: rt_model.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
```

```
##            mean se_mean     sd    2.5%     25%     50%     75%   97.5% n_eff
## eta[1]    76.20    0.16   8.59   60.76   70.24   75.83   81.47   95.13  2774
## eta[2]    61.11    0.12   6.34   49.55   56.76   60.87   64.99   74.94  2828
## eta[3]    68.38    0.12   6.62   56.32   63.88   68.12   72.49   82.76  2900
## eta[4]    64.50    0.10   5.72   54.08   60.64   64.32   68.07   76.97  3037
## eta[5]    36.66    0.05   3.00   31.14   34.63   36.53   38.54   43.16  3201
## eta[6]    52.66    0.07   4.04   45.21   49.93   52.44   55.16   61.50  3374
## eta[7]    87.56    0.10   6.28   76.07   83.32   87.22   91.54  100.80  3628
## eta[8]    87.80    0.10   5.99   76.83   83.71   87.51   91.60  100.47  3867
## eta[9]    81.71    0.08   5.37   71.99   78.04   81.51   85.17   93.07  4075
## eta[10]   66.87    0.07   4.27   59.13   63.90   66.70   69.66   76.00  4185
## eta[11]   69.35    0.07   4.36   61.25   66.34   69.22   72.21   78.61  4217
## eta[12]  102.04    0.10   6.38   90.17   97.55  101.76  106.26  115.15  4146
## eta[13]   50.75    0.05   3.17   44.80   48.52   50.63   52.83   57.32  4013
## eta[14]   91.18    0.09   5.70   80.55   87.17   90.98   94.92  103.04  3856
## eta[15]   84.00    0.09   5.26   74.13   80.39   83.82   87.47   95.00  3693
## eta[16]   89.55    0.09   5.60   79.19   85.76   89.38   93.29  101.08  3561
## eta[17]   86.08    0.09   5.38   76.13   82.43   85.89   89.61   97.03  3442
## eta[18]   60.90    0.07   3.80   53.91   58.29   60.77   63.39   68.54  3342
## eta[19]   53.02    0.06   3.30   46.90   50.78   52.91   55.16   59.59  3274
## eta[20]   71.87    0.08   4.44   63.54   68.84   71.72   74.75   80.79  3244
## eta[21]  118.67    0.13   7.26  104.93  113.68  118.47  123.44  133.32  3232
## eta[22]  184.55    0.20  11.18  163.19  176.81  184.41  191.99  207.01  3243
## eta[23]  145.66    0.15   8.75  128.94  139.52  145.48  151.51  163.38  3186
## eta[24]  178.96    0.19  10.63  159.15  171.47  178.61  186.06  200.42  3185
## eta[25]  191.13    0.20  11.24  170.28  183.29  190.86  198.74  213.45  3225
## eta[26]  109.32    0.11   6.36   97.54  104.86  109.14  113.65  121.85  3321
## eta[27]  141.18    0.14   8.15  125.90  135.59  140.88  146.78  157.56  3400
## eta[28]  238.64    0.23  13.66  212.70  229.25  238.08  247.93  266.27  3428
## eta[29]  260.39    0.25  14.82  232.17  250.16  259.56  270.55  290.00  3442
## eta[30]  287.12    0.28  16.25  255.98  276.02  286.23  298.03  320.02  3445
## eta[31]  272.83    0.26  15.40  243.72  262.36  272.10  283.07  304.37  3436
## eta[32]  187.77    0.18  10.57  167.77  180.58  187.27  194.77  209.46  3416
## eta[33]  143.82    0.14   8.09  128.57  138.27  143.48  149.06  160.44  3385
## eta[34]  235.56    0.23  13.24  210.70  226.45  235.07  244.11  262.64  3346
## eta[35]  247.05    0.24  13.90  221.18  237.53  246.34  256.21  275.31  3304
## eta[36]  202.27    0.20  11.38  180.98  194.48  201.78  209.84  225.62  3242
## eta[37]  389.55    0.39  21.91  348.25  374.42  388.56  404.06  434.97  3190
## eta[38]  330.83    0.33  18.58  296.02  317.95  329.99  343.26  368.88  3174
## eta[39]  169.46    0.17   9.52  151.70  162.90  169.04  175.77  189.04  3168
## eta[40]  125.70    0.13   7.06  112.56  120.81  125.46  130.41  140.17  3168
## eta[41]  284.04    0.28  15.95  254.10  272.71  283.50  294.57  316.48  3172
## eta[42]  288.50    0.29  16.22  257.71  277.09  287.94  299.15  321.36  3182
## eta[43]  231.94    0.23  13.06  207.30  223.00  231.43  240.61  258.17  3197
## eta[44]  236.88    0.24  13.39  211.64  227.72  236.45  245.62  263.76  3216
## eta[45]  214.62    0.21  12.19  192.10  206.19  214.29  222.39  238.92  3236
## eta[46]  165.24    0.17   9.44  148.05  158.66  165.03  171.34  184.36  3225
## eta[47]  101.03    0.10   5.81   90.62   96.97  100.81  104.81  112.87  3187
## eta[48]  244.90    0.25  14.19  219.10  235.26  244.43  254.26  273.83  3146
## eta[49]  223.55    0.23  13.05  199.54  214.86  223.08  232.19  250.32  3154
## eta[50]  270.95    0.28  15.96  241.86  260.02  270.30  281.49  304.05  3181
## eta[51]  209.66    0.22  12.47  187.00  201.05  209.13  217.75  235.33  3187
## eta[52]  271.14    0.29  16.28  241.05  259.93  270.46  281.51  304.82  3133
## eta[53]  187.37    0.20  11.37  166.36  179.41  186.85  194.57  211.03  3078
```

```
## eta[54]  126.32    0.14   7.74  112.17  121.00  125.88  131.16  142.61  3025
## eta[55]  190.50    0.22  11.80  168.88  182.23  189.82  198.04  215.62  2981
## eta[56]  299.39    0.34  18.73  264.95  286.05  298.37  311.46  338.83  2948
## eta[57]  280.54    0.32  17.55  248.27  268.04  279.59  291.85  317.49  2948
## eta[58]  275.02    0.32  17.36  243.55  262.98  274.00  286.28  311.97  2925
## eta[59]  265.88    0.31  16.90  235.21  254.06  264.79  276.96  300.67  2912
## eta[60]  239.05    0.28  15.27  211.49  228.31  238.08  248.70  270.63  2907
## eta[61]  119.64    0.14   7.67  105.75  114.22  119.11  124.49  135.45  2913
## eta[62]  218.82    0.26  14.07  193.48  208.81  217.90  227.88  248.41  2922
## eta[63]  315.96    0.38  20.35  279.38  301.66  314.73  329.04  358.14  2931
## eta[64]  442.20    0.52  28.49  390.81  422.37  440.47  460.40  501.91  2946
## eta[65]  381.15    0.45  24.53  337.31  364.30  379.70  396.55  433.03  2962
## eta[66]  390.88    0.46  25.10  345.33  373.59  389.40  406.91  444.28  2977
## eta[67]  332.66    0.39  21.30  294.00  318.23  331.69  346.25  378.46  2989
## eta[68]  197.96    0.23  12.63  174.52  189.41  197.60  206.01  224.78  3002
## eta[69]  430.07    0.50  27.31  379.69  411.43  429.18  447.51  487.56  3018
## eta[70]  636.45    0.73  40.19  562.20  609.24  635.44  662.60  720.17  3041
## eta[71]  717.97    0.81  45.03  636.23  686.98  716.48  747.14  812.08  3081
## eta[72]  921.27    1.02  57.36  817.31  881.92  919.68  958.19 1040.37  3151
## eta[73]  911.97    0.99  56.38  809.26  873.01  911.25  947.72 1027.84  3267
## eta[74]  687.89    0.72  42.28  610.06  658.41  686.90  715.53  774.76  3446
## eta[75]  681.99    0.69  41.79  605.10  652.20  680.74  709.64  768.51  3699
## eta[76]  933.38    0.90  57.33  825.43  892.79  931.78  971.64 1051.76  4023
## eta[77] 1750.73    1.64 108.77 1544.82 1674.73 1746.34 1823.28 1971.44  4385
## eta[78] 2173.95    2.05 138.46 1915.46 2077.36 2167.86 2265.70 2454.04  4572
## eta[79] 2269.26    2.25 150.78 1990.04 2164.51 2259.87 2368.55 2573.59  4507
## eta[80] 2390.56    2.59 169.06 2081.28 2274.75 2379.15 2500.74 2738.96  4256
## eta[81] 2826.57    3.48 217.08 2428.81 2675.30 2810.95 2966.81 3280.29  3894
## eta[82] 1474.38    2.11 125.11 1247.45 1387.50 1464.81 1554.47 1739.46  3522
## eta[83] 2435.87    4.15 231.28 2020.95 2273.84 2419.87 2582.09 2937.66  3105
## eta[84] 4525.87    9.25 484.42 3671.53 4179.99 4485.87 4831.76 5577.66  2743
## eta[85] 4920.88   11.97 595.43 3885.49 4498.21 4866.11 5295.96 6225.96  2476
##         Rhat
## eta[1]     1
## eta[2]     1
## eta[3]     1
## eta[4]     1
## eta[5]     1
## eta[6]     1
## eta[7]     1
## eta[8]     1
## eta[9]     1
## eta[10]    1
## eta[11]    1
## eta[12]    1
## eta[13]    1
## eta[14]    1
## eta[15]    1
## eta[16]    1
## eta[17]    1
## eta[18]    1
## eta[19]    1
## eta[20]    1
## eta[21]    1
```

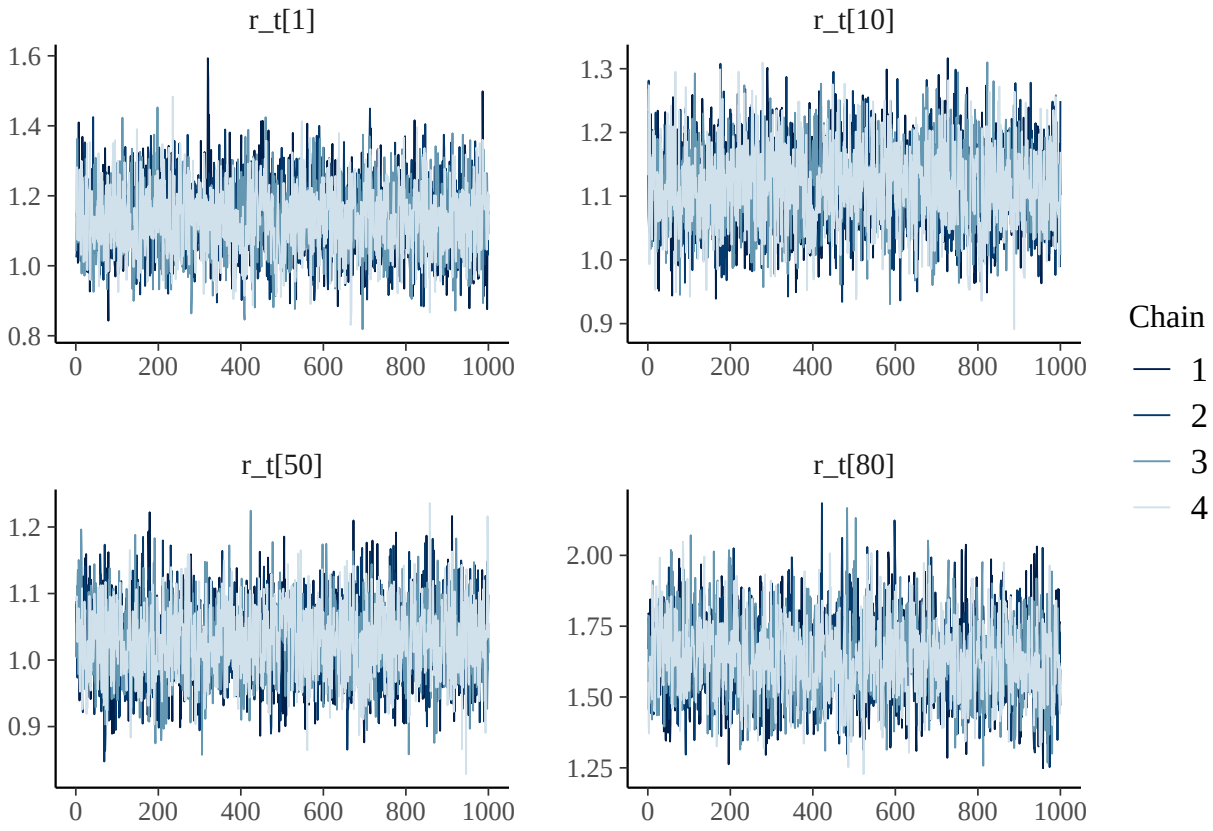
```
## eta[22]    1
## eta[23]    1
## eta[24]    1
## eta[25]    1
## eta[26]    1
## eta[27]    1
## eta[28]    1
## eta[29]    1
## eta[30]    1
## eta[31]    1
## eta[32]    1
## eta[33]    1
## eta[34]    1
## eta[35]    1
## eta[36]    1
## eta[37]    1
## eta[38]    1
## eta[39]    1
## eta[40]    1
## eta[41]    1
## eta[42]    1
## eta[43]    1
## eta[44]    1
## eta[45]    1
## eta[46]    1
## eta[47]    1
## eta[48]    1
## eta[49]    1
## eta[50]    1
## eta[51]    1
## eta[52]    1
## eta[53]    1
## eta[54]    1
## eta[55]    1
## eta[56]    1
## eta[57]    1
## eta[58]    1
## eta[59]    1
## eta[60]    1
## eta[61]    1
## eta[62]    1
## eta[63]    1
## eta[64]    1
## eta[65]    1
## eta[66]    1
## eta[67]    1
## eta[68]    1
## eta[69]    1
## eta[70]    1
## eta[71]    1
## eta[72]    1
## eta[73]    1
## eta[74]    1
## eta[75]    1
```

```
## eta[76]    1
## eta[77]    1
## eta[78]    1
## eta[79]    1
## eta[80]    1
## eta[81]    1
## eta[82]    1
## eta[83]    1
## eta[84]    1
## eta[85]    1
##
## Samples were drawn using NUTS(diag_e) at Sat Oct 24 20:29:52 2020.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Trace plots

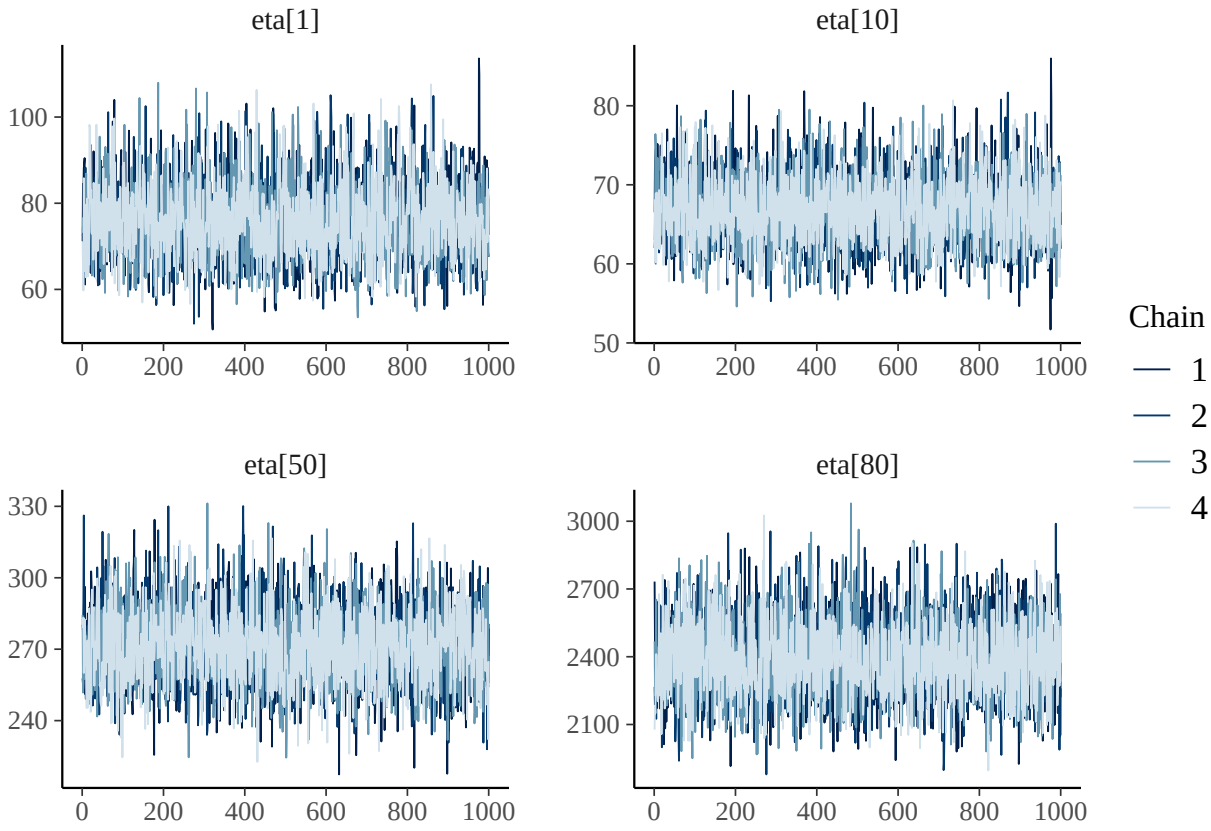
```
mcmc_trace(as.array(fit_model_lomb, pars = c('r_t[1]', 'r_t[10]', 'r_t[50]', 'r_t[80]')),
           np = nuts_params(fit_model_lomb)
           )
```

```
## No divergences to plot.
```



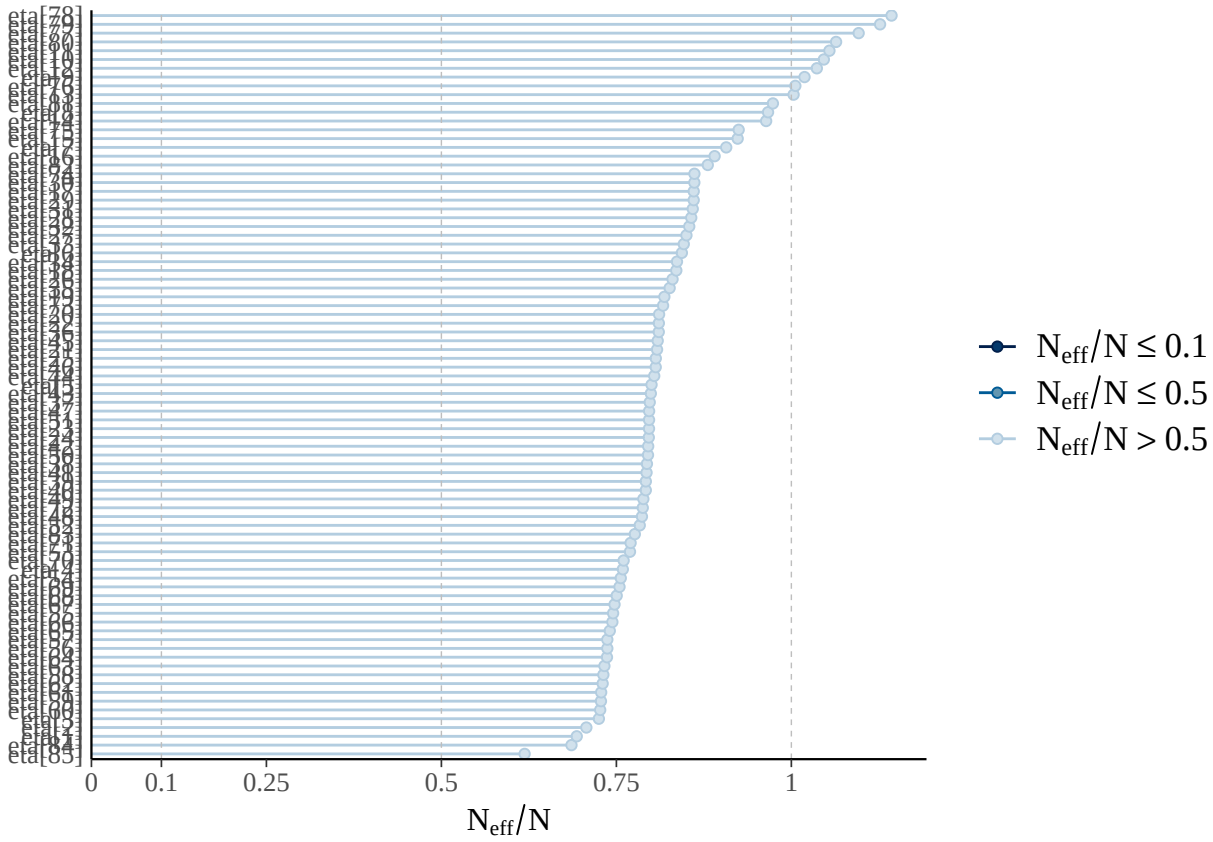

```
mcmc_trace(as.array(fit_model_lomb, pars = c('eta[1]', 'eta[10]', 'eta[50]', 'eta[80]')),
           np = nuts_params(fit_model_lomb)
)
```

```
## No divergences to plot.
```

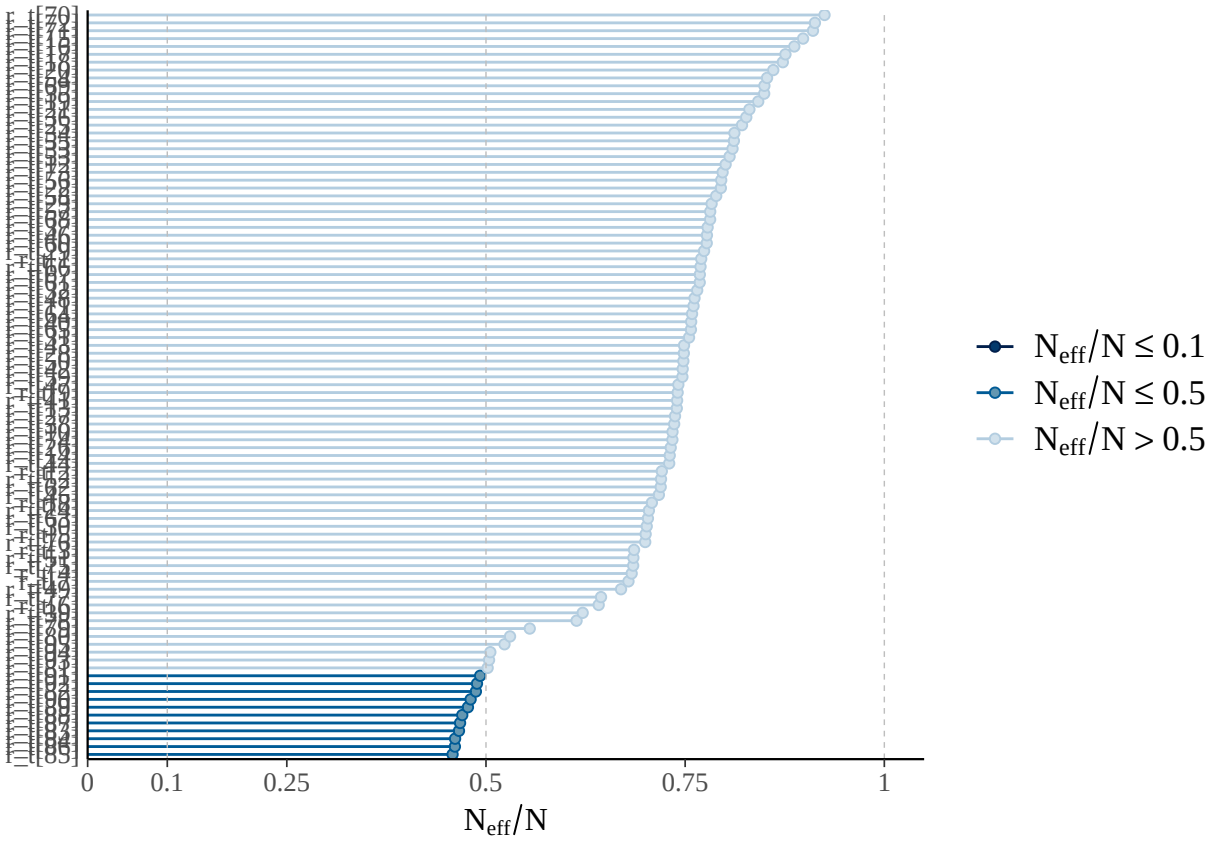


Effective sample size

```
ratios1 <- neff_ratio(fit_model_lomb, pars = c('eta'))
mcmc_neff(ratios1) + yaxis_text(hjust = 1)
```

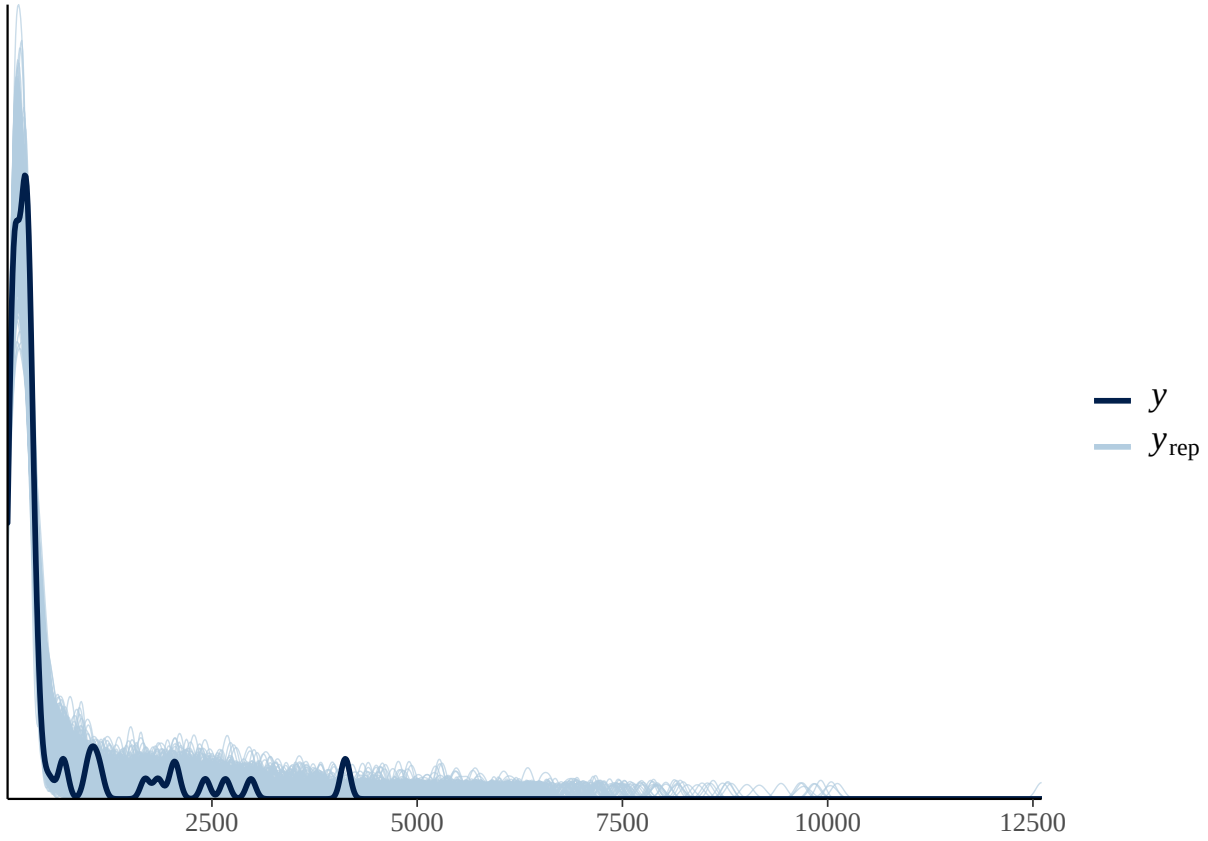



```
ratios2 <- neff_ratio(fit_model_lomb, pars = c('r_t'))
mcmc_neff(ratios2) + yaxis_text(hjust = 1)
```

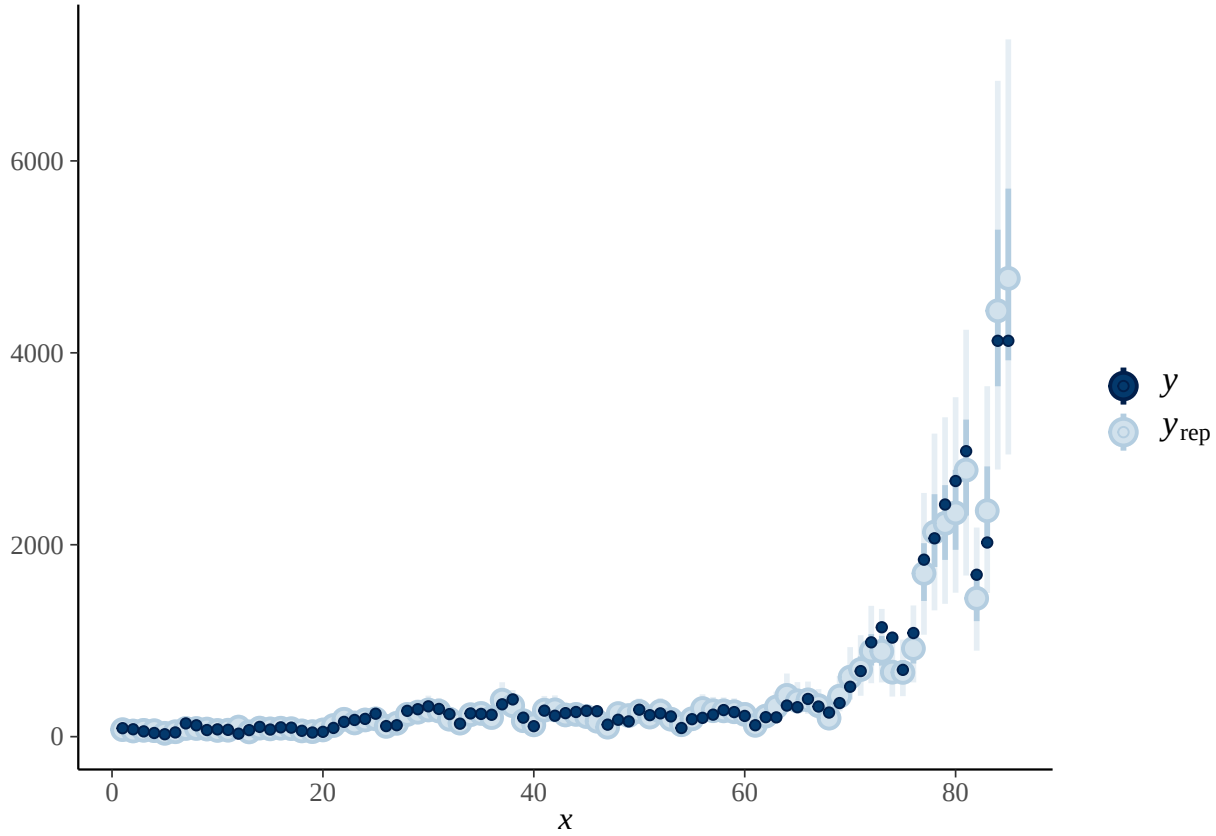


Posterior predictive check

```
y_rep <- as.matrix(fit_model_lomb, pars = "y_rep")
ppc_dens_overlay(y = data_lombardia$positive[nonzero_days_l], y_rep[1:1000, ])
```

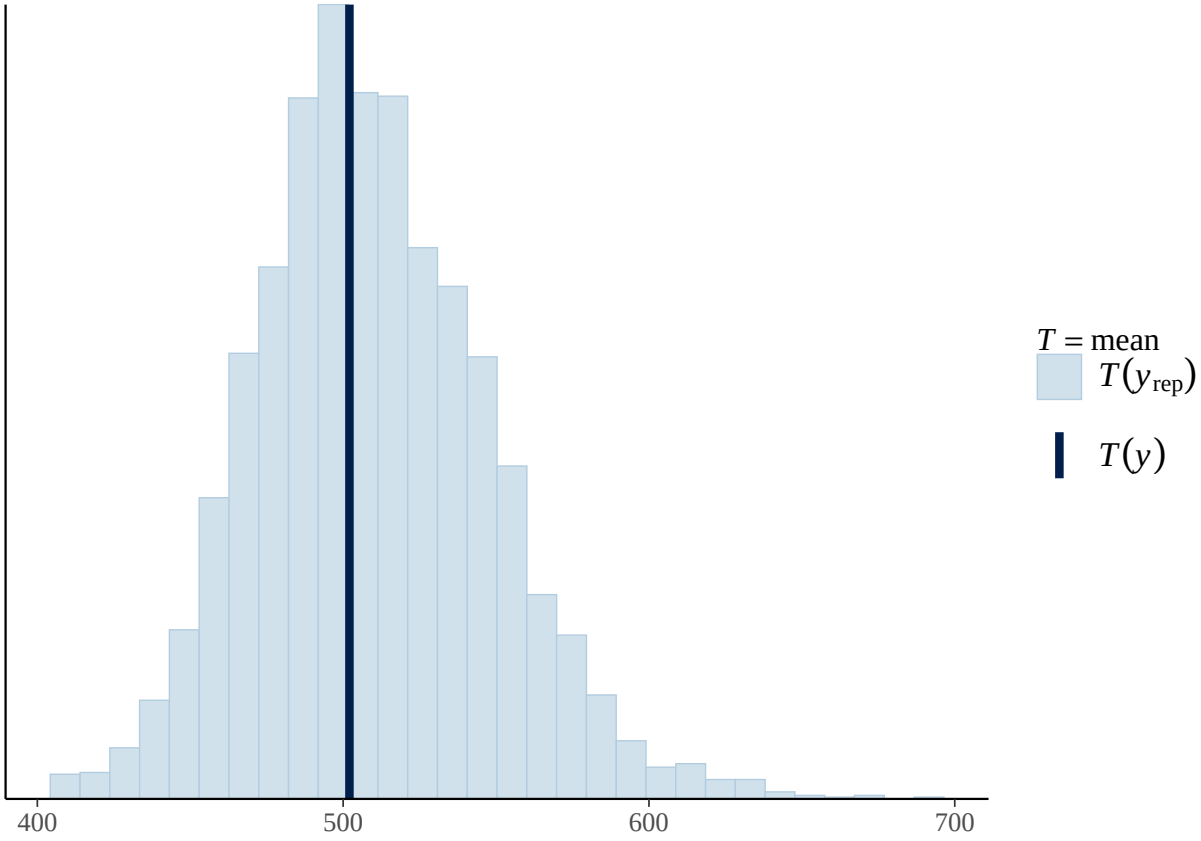



```
ppc_intervals(
  y = data_lombardia$positive[nonzero_days_l],
  yrep = y_rep
)
```




```
ppc_stat(y = stan_data_l$nonzero_positives, yrep = y_rep, stat = 'mean')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

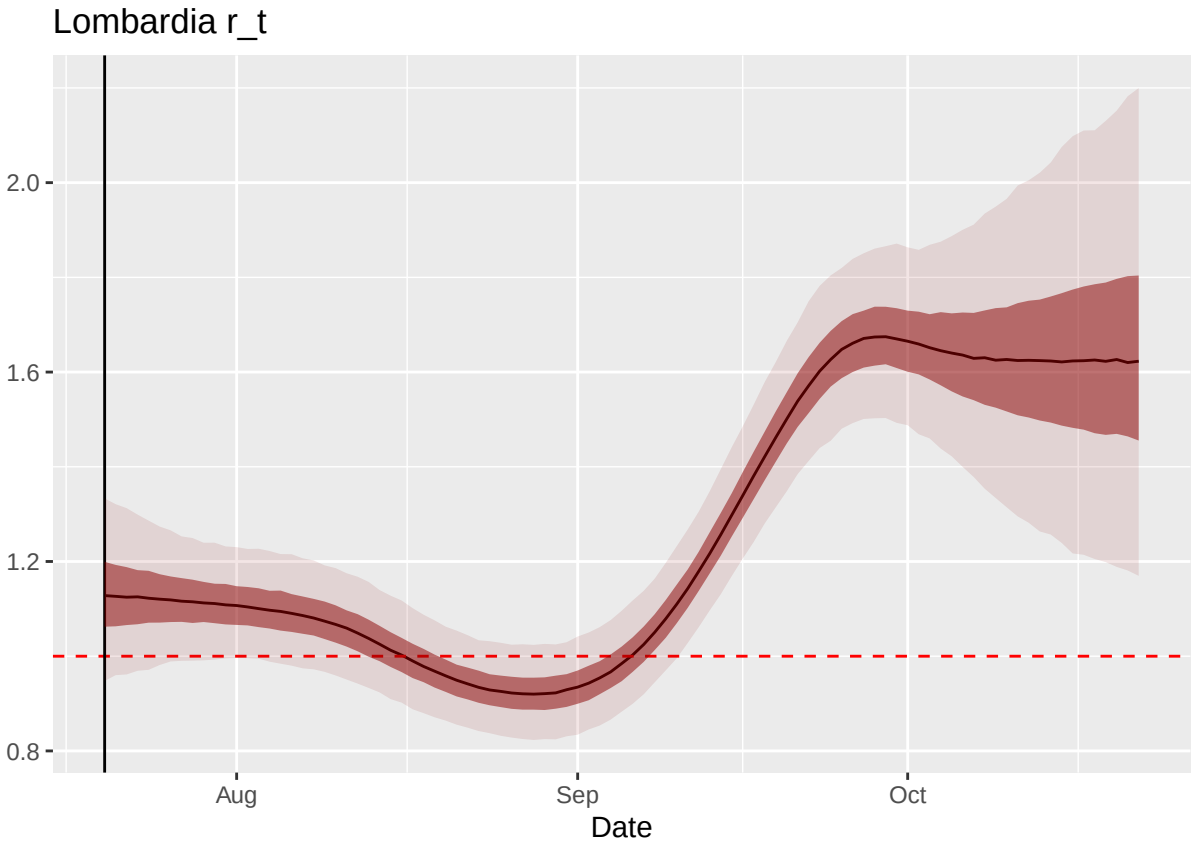


R_t curve

```
fit_summary_lomb <- summary(fit_model_lomb)

rt_idx <- which(rownames(fit_summary_lomb$summary) == 'r_t[1]')
medians_rt <- fit_summary_lomb$summary[rt_idx: (rt_idx + stan_data_l$N - 1), '50%']
min_rt_50_interval <- fit_summary_lomb$summary[rt_idx: (rt_idx + stan_data_l$N - 1), '25%']
max_rt_50_interval <- fit_summary_lomb$summary[rt_idx: (rt_idx + stan_data_l$N - 1), '75%']
min_rt_95_interval <- fit_summary_lomb$summary[rt_idx: (rt_idx + stan_data_l$N - 1), '2.5%']
max_rt_95_interval <- fit_summary_lomb$summary[rt_idx: (rt_idx + stan_data_l$N - 1), '97.5%']


ggplot(data = NULL, aes(x = data_lombardia$date, y = medians_rt)) +
  geom_line() +
  xlab('Date') +
  ylab('') +
  ggtitle( 'Lombardia r_t')+
  geom_hline(yintercept=1, linetype="dashed", color = "red") +
  geom_vline(xintercept = data_lombardia$date[1]) +
  geom_ribbon(aes(ymin = min_rt_50_interval, ymax = max_rt_50_interval), alpha= 0.5, fill = 'darkred') +
  geom_ribbon(aes(ymin = min_rt_95_interval, ymax = max_rt_95_interval), alpha= 0.1, fill = 'darkred')
```


Lombardia r_t
2.0
1.6
1.2
0.8
Aug
Sep
Oct
Date