Etas_approximations

Francesco Serafini

06/07/2021

Temporal ETAS model

Here, we investigate various methodology to approximate a temporal ETAS model. A temporal ETAS model is a marked point process model, specifically an Hawkes process, with conditional intesity given by:

$$\lambda(t)|\mathcal{H}_t = \mu + K \sum_{i:t_i<t} \exp(\alpha(m_i - M_0)) \frac{1}{(t - t_i + c)^p}$$

where $\mathcal{H}_t = \{(t_i, m_i)\}$ is the history of the process. The model, as we present it, has 5 parameters μ, K, α, c, p which have to be non-negative, except for p which has to be greater than 1.

To be free from any constraint we are going to consider a different parametrization, specifically $\mu = \exp\theta_1$, $K = \exp\theta_2$, $\alpha = \exp\theta_3$, $c = \exp\theta_4$, $p - 1 = \exp\theta_5$. Now the parameters θ_i are free from any constraint. The conditional intensity that we are going to consider is

$$\lambda(t)|\mathcal{H}_t = \exp(\theta_1) + \exp(\theta_2) \sum_{i:t_i<t} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big) \frac{1}{(t - t_i + \exp\theta_4)^{1+\exp\theta_5}}$$

Parametrized in this way, the conditional intensity is an increasing function of $\theta_1, \theta_2, \theta_3$, and a decreasing function of θ_4, θ_5.

The expected number of points in (T_1, T_2) by the model is given by:

$$\Lambda(T_1, T_2) = \int_{T_1}^{T_2} \lambda(t)dt = \int_{T_1}^{T_2} \left(\exp(\theta_1) + \exp(\theta_2) \sum_{i:t_i<t} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big) \frac{1}{(t - t_i + \exp\theta_4)^{1+\exp\theta_5}}\right) dt$$

$$= \exp(\theta_1)(T_2 - T_1) + \exp(\theta_2) \int_{T_1}^{T_2} \sum_{i:t_i\in\mathcal{H}_t} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big) \frac{1}{(t - t_i + \exp\theta_4)^{1+\exp\theta_5}} \mathbb{I}(t_i < t)dt$$

$$= \exp(\theta_1)(T_2 - T_1) + \exp(\theta_2)\Big(\sum_{i:t_i<T_1} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big) \int_{T_1}^{T_2} \frac{1}{(t - t_i + \exp\theta_4)^{1+\exp\theta_5}} \mathbb{I}(t_i < t)dt +$$

$$\sum_{i:t_i\geq T_1} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big) \int_{T_1}^{T_2} \frac{1}{(t - t_i + \exp\theta_4)^{1+\exp\theta_5}} \mathbb{I}(t_i < t)dt\Big)$$

Where, assuming $T_1 > t_i$

$$\int_{T_1}^{T_2} (t - t_i + \exp\theta_4)^{-1-\exp\theta_5} dt = -\frac{(t - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Bigg|_{T_1}^{T_2}$$

$$= \frac{(T_1 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5} - \frac{(T_2 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}$$

While

$$\int_{t_i}^{T_2} (t - t_i + \exp\theta_4)^{-1-\exp\theta_5} dt = \frac{\exp(\theta_4)^{-\exp\theta_5}}{\exp\theta_5} - \frac{(T_2 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}$$

Which leads to

$$\Lambda(T_1, T_2) = \int_{T_1}^{T_2} \lambda(t)dt =$$

$$= \exp(\theta_1)(T_2 - T_1) + \exp(\theta_2)\Bigg(\sum_{i:t_i<T_1} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big)\Big(\frac{(T_1 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5} -$$

$$\frac{(T_2 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Big) + \sum_{i:t_i\geq T_1} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big)\Big(\frac{(\exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5} -$$

$$\frac{(T_2 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Big)\Bigg)$$

This is the intensity and the expected number of points of the process without considering the marks. The effect of the mark is cosidered only inside the triggering function.

The expression of the number of points seen above is the most general case for which we are interested in the expected number of points in an interval which has observations inside it and before it. The first summation is the contribution to the expected number of points given by the *past* (observations s.t. $t_i < T_1$) while the second summation is the contribution given by the *present* (observations $t_i > T_1$). In practice, these cases rarely happears together. Indeed, if we are interested in the likelihood of the model then $T_1 < t_i$ and we have only the second summation. If we are intersted in predicting the future given the past then $T_1 > t_i$ and we have only the first summation.

We have than if $T_1 > t_i$ for any $t_i \in \mathcal{H}_t$

$$\Lambda(T_1, T_2) = \exp(\theta_1)(T_2 - T_1) + \exp(\theta_2) \sum_{i:t_i\in\mathcal{H}_t} \exp\Big(\exp(\theta_3)(m_i - M_0)\Big)\Big(\frac{(T_1 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5} - \frac{(T_2 - t_i + \exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Big)$$

If $t_i \geq T_1$ for any $t_i \in \mathcal{H}_t$

$$\Lambda(T_1,T_2) = \exp(\theta_1)(T_2-T_1)+\exp(\theta_2)\sum_{i:t_i\in\mathcal{H}_t}\exp\Big(\exp(\theta_3)(m_i-M_0)\Big)\Big(\frac{(\exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}-\frac{(T_2-t_i+\exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Big)$$

From here we can see that, in the last case, the expected number of points is an increasing function of $\theta_1, \theta_2, \theta_3$ because $m_i - M_0 \geq 0$, and a decreasing function of θ_4, θ_5.

Given a set of N observations $\mathcal{H}_t = \{(t_i, m_i), i = 1, ..., N\}$ such that $t_i \in [T_1, T_2]$ for any $i = 1, ..., N$ the likelihood of the model is given by:

$$\mathcal{L}(\boldsymbol{\theta}) = -\Lambda(T_1,T_2) + \sum_{i=1}^{N}\log\lambda(t_i)$$

$$= -\Bigg(\exp(\theta_1)(T_2-T_1)+\exp(\theta_2)\sum_{i:t_i\in\mathcal{H}_t}\exp\Big(\exp(\theta_3)(m_i-M_0)\Big)\Big(\frac{(\exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}-\frac{(T_2-t_i+\exp\theta_4)^{-\exp\theta_5}}{\exp\theta_5}\Big)\Bigg)+$$

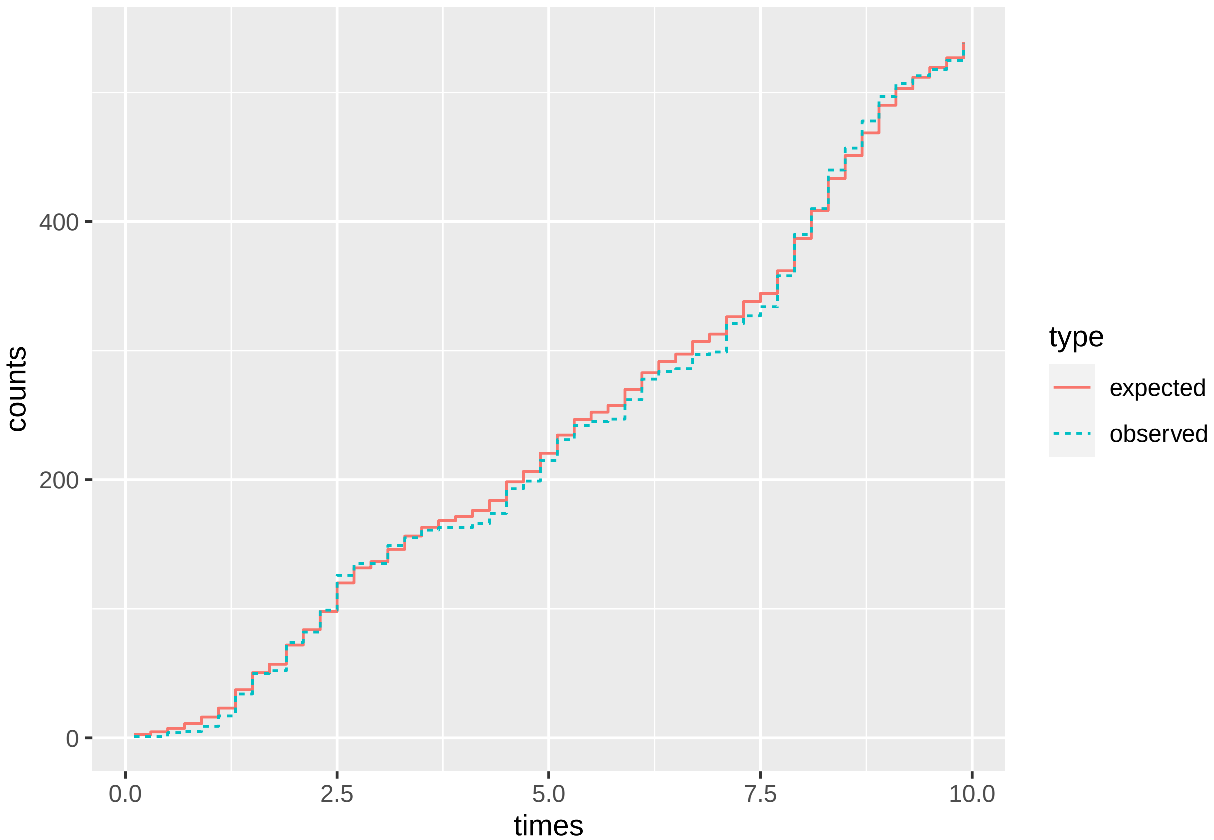
$$\sum_{i=1}^{N}\log\Big(\exp(\theta_1)+\exp(\theta_2)\sum_{j:t_j<t_i}\exp\Big(\exp(\theta_3)(m_j-M_0)\Big)\frac{1}{(t_i-t_j+\exp\theta_4)^{1+\exp\theta_5}}\Big)$$

```r
source('ETAS_utils.R')
```

```
## Loading required package: viridisLite

## Loading required package: sp

## Loading required package: Matrix

## Loading required package: foreach

## Loading required package: parallel

## This is INLA_21.02.23 built 2021-03-15 10:11:24 UTC.
##  - See www.r-inla.org/contact-us for how to get help.
##  - To enable PARDISO sparse library; see inla.pardiso()
##  - Save 273.9Mb of storage running 'inla.prune()'

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

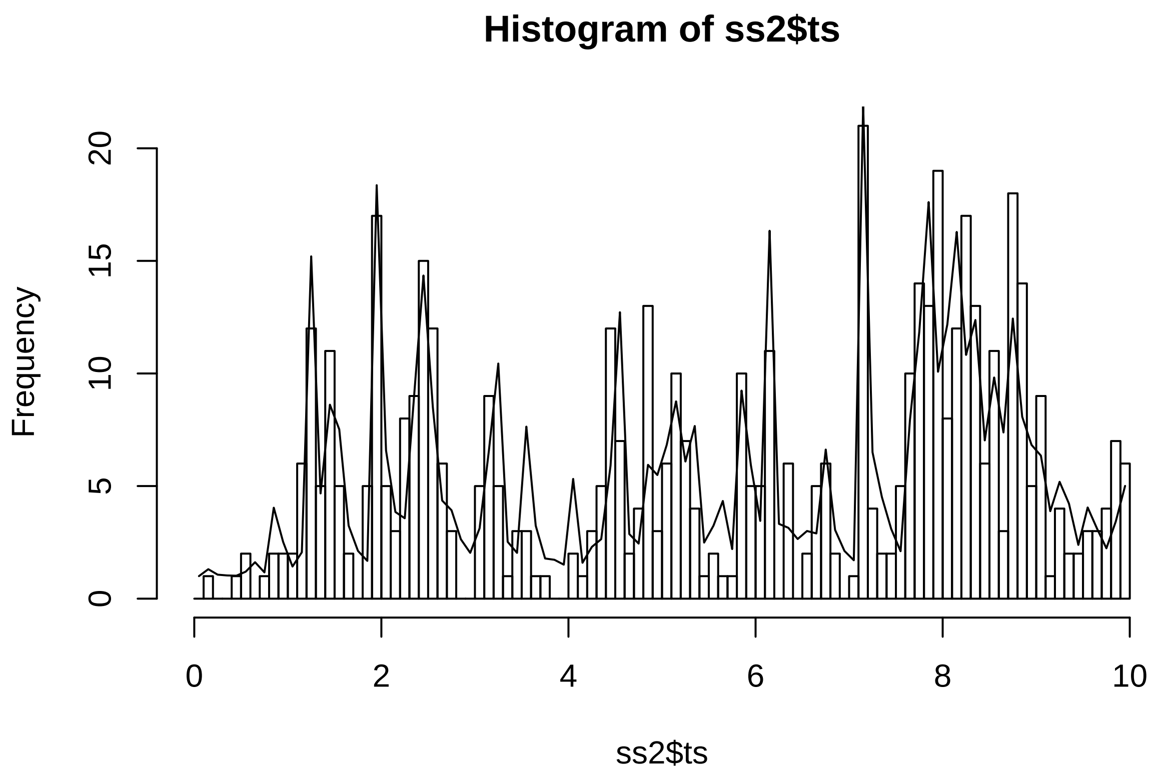
```
## The following object is masked from 'package:inlabru':
##
##     like

##
## Attaching package: 'metR'

## The following object is masked from 'package:INLA':
##
##     f

##
## Attaching package: 'matrixStats'

## The following object is masked from 'package:dplyr':
##
##     count
```

```
t.parms <- log(c(10, 0.8*(0.01^0.5)*0.5, 0.2, 0.01, 0.5))
Tlim = 10
# ss2 <- sample.ETAS.unnorm(t.parms, beta.par = 2.3, M0 = 2.5, Tlim)
# save(ss2, file = 'sample.etas.RData')

load('sample.etas.RData')
toplot.cumcounts(Tlim, t.parms, ss2, M0 = 2.5, by.s = 0.2)
```




```
t.breaks <- seq(0, Tlim, by = 0.1)
hh <- hist(ss2$ts, breaks = t.breaks, plot = FALSE)
ll <- log.lambda.ETAS(hh$mids, t.parms, ss2, M0 = 2.5)
```

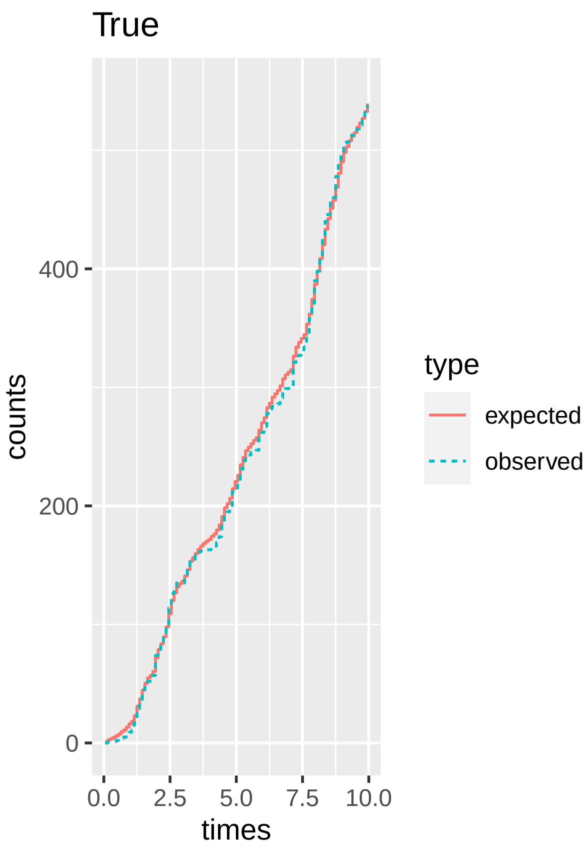
```r
plot(hh)
lines(hh$mids, exp(ll)*0.1)
```

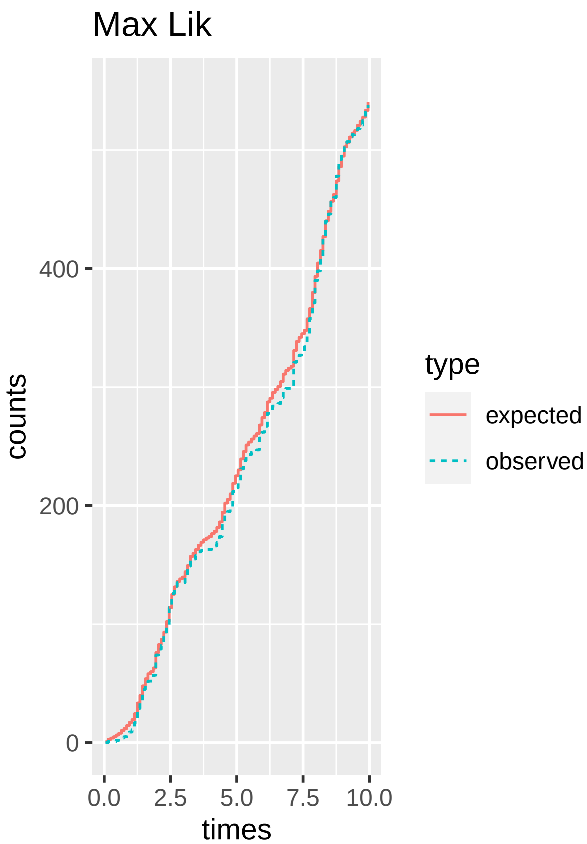



```r
ML.optim <- optim(c(0,0,0,0,0), ETAS.log.lik.toptim, Ht = ss2, M0 = 2.5, Tlim = 10)

rbind(ML = exp(ML.optim$par), True = exp(t.parms))
```

```
##          [,1]       [,2]         [,3]      [,4]      [,5]
## ML   10.55479 0.01415566 8.781532e-06 0.0146675 0.9483153
## True 10.00000 0.04000000 2.000000e-01 0.0100000 0.5000000
```

```r
pl.cs.ML <- toplot.cumcounts(10, ML.optim$par, ss2, 2.5, 0.1) + labs(title = 'Max Lik') +
  ylim(0,550)
pl.cs.true <- toplot.cumcounts(10, t.parms, ss2, 2.5, 0.1) + labs(title = 'True') +
  ylim(0,550)
multiplot(pl.cs.true, pl.cs.ML, cols = 2)
```

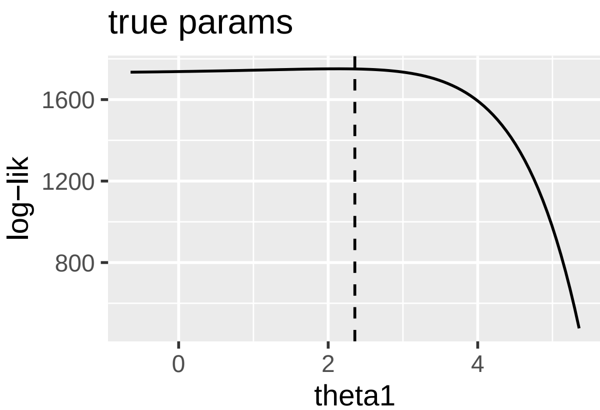


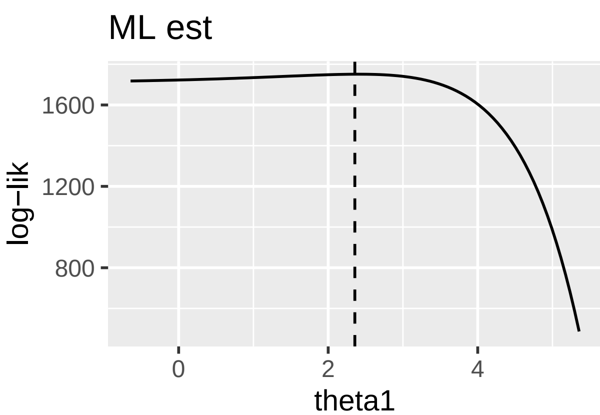



```r
# univariate likelihood analysis
toplot.univ.comp <- function(par.values, par.idx, par.name,
                             theta.par, ML.est, Ht, M0, Tlim){


  theta.m.true <- cbind(rep(t.parms[1], length(par.values)),
                        rep(t.parms[2], length(par.values)),
                        rep(t.parms[3], length(par.values)),
                        rep(t.parms[4], length(par.values)),
                        rep(t.parms[5], length(par.values)))

  theta.m.ML <- cbind(rep(ML.est[1], length(par.values)),
                      rep(ML.est[2], length(par.values)),
                      rep(ML.est[3], length(par.values)),
                      rep(ML.est[4], length(par.values)),
                      rep(ML.est[5], length(par.values)))

  theta.m.true[,par.idx] <- par.values
  theta.m.ML[,par.idx] <- par.values

  LL.true <- sapply(1:nrow(theta.m.true), function(i)
    ETAS.log.lik(theta.m.true[i,], Ht, M0, Tlim))

  LL.ML <- sapply(1:nrow(theta.m.ML), function(i)
    ETAS.log.lik(theta.m.ML[i,], Ht, M0, Tlim))

  ll.lim <- range(c(LL.true, LL.ML))
```

```
  df <- rbind(data.frame(x = par.values,
                         y = LL.ML,
                         typ = 'ml'),
              data.frame(x = par.values,
                         y = LL.true,
                         typ = 'true'))
  list(pl.true = ggplot(data.frame(x = par.values,
                                   y = LL.true), aes(x = x, y = y)) +
         geom_line() +
         geom_vline(xintercept = ML.est[par.idx], linetype = 2) +
         labs(title = 'true params') +
         ylab('log-lik') +
         xlab(par.name) +
         ylim(ll.lim),
       pl.ML = ggplot(data.frame(x = par.values,
                                 y = LL.ML), aes(x = x, y = y)) +
         geom_line() +
         geom_vline(xintercept = ML.est[par.idx], linetype = 2)+
         labs(title = 'ML est')+
         ylab('log-lik') +
         xlab(par.name) +
         ylim(ll.lim),
       df = df)

}
```

```
const <- 3
theta1.v <- seq(ML.optim$par[1] - const,
                ML.optim$par[1] + const,length.out = 100)
plotl.th1 <- toplot.univ.comp(theta1.v, 1, 'theta1', t.parms, ML.optim$par,
                          ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotl.th1, cols = 2)
```






```
##               x         y  typ
## 1   -0.64341996 1717.7388   ml
## 2   -0.58281390 1718.1271   ml
## 3   -0.52220784 1718.5293   ml
## 4   -0.46160178 1718.9460   ml
## 5   -0.40099572 1719.3776   ml
## 6   -0.34038966 1719.8248   ml
## 7   -0.27978360 1720.2878   ml
## 8   -0.21917753 1720.7672   ml
## 9   -0.15857147 1721.2634   ml
## 10  -0.09796541 1721.7769   ml
## 11  -0.03735935 1722.3081   ml
## 12   0.02324671 1722.8575   ml
## 13   0.08385277 1723.4253   ml
## 14   0.14445883 1724.0120   ml
## 15   0.20506489 1724.6179   ml
## 16   0.26567095 1725.2432   ml
## 17   0.32627701 1725.8881   ml
## 18   0.38688307 1726.5529   ml
## 19   0.44748913 1727.2376   ml
## 20   0.50809519 1727.9423   ml
## 21   0.56870125 1728.6668   ml
## 22   0.62930731 1729.4110   ml
## 23   0.68991337 1730.1747   ml
## 24   0.75051943 1730.9575   ml
## 25   0.81112550 1731.7588   ml
## 26   0.87173156 1732.5780   ml
```

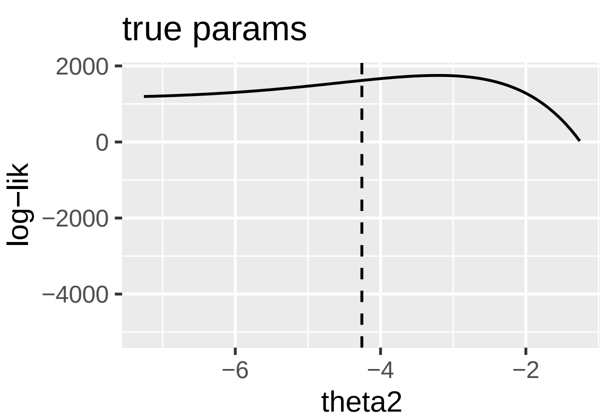
```
## 27   0.93233762 1733.4143   ml
## 28   0.99294368 1734.2666   ml
## 29   1.05354974 1735.1337   ml
## 30   1.11415580 1736.0143   ml
## 31   1.17476186 1736.9065   ml
## 32   1.23536792 1737.8084   ml
## 33   1.29597398 1738.7179   ml
## 34   1.35658004 1739.6323   ml
## 35   1.41718610 1740.5488   ml
## 36   1.47779216 1741.4639   ml
## 37   1.53839822 1742.3741   ml
## 38   1.59900428 1743.2752   ml
## 39   1.65961034 1744.1624   ml
## 40   1.72021640 1745.0306   ml
## 41   1.78082247 1745.8741   ml
## 42   1.84142853 1746.6865   ml
## 43   1.90203459 1747.4606   ml
## 44   1.96264065 1748.1887   ml
## 45   2.02324671 1748.8622   ml
## 46   2.08385277 1749.4717   ml
## 47   2.14445883 1750.0067   ml
## 48   2.20506489 1750.4560   ml
## 49   2.26567095 1750.8071   ml
## 50   2.32627701 1751.0465   ml
## 51   2.38688307 1751.1593   ml
## 52   2.44748913 1751.1294   ml
## 53   2.50809519 1750.9393   ml
## 54   2.56870125 1750.5699   ml
## 55   2.62930731 1750.0005   ml
## 56   2.68991337 1749.2086   ml
## 57   2.75051943 1748.1697   ml
## 58   2.81112550 1746.8577   ml
## 59   2.87173156 1745.2438   ml
## 60   2.93233762 1743.2973   ml
## 61   2.99294368 1740.9848   ml
## 62   3.05354974 1738.2703   ml
## 63   3.11415580 1735.1151   ml
## 64   3.17476186 1731.4773   ml
## 65   3.23536792 1727.3120   ml
## 66   3.29597398 1722.5707   ml
## 67   3.35658004 1717.2014   ml
## 68   3.41718610 1711.1479   ml
## 69   3.47779216 1704.3503   ml
## 70   3.53839822 1696.7440   ml
## 71   3.59900428 1688.2598   ml
## 72   3.65961034 1678.8234   ml
## 73   3.72021640 1668.3554   ml
## 74   3.78082247 1656.7706   ml
## 75   3.84142853 1643.9779   ml
## 76   3.90203459 1629.8798   ml
## 77   3.96264065 1614.3723   ml
## 78   4.02324671 1597.3438   ml
## 79   4.08385277 1578.6755   ml
## 80   4.14445883 1558.2404   ml
```

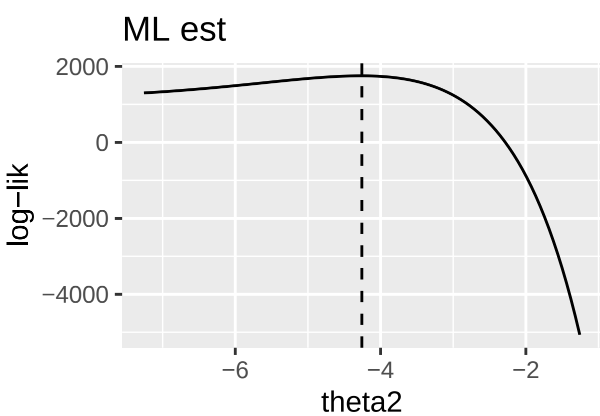
```
## 81   4.20506489 1535.9030   ml
## 82   4.26567095 1511.5189   ml
## 83   4.32627701 1484.9339   ml
## 84   4.38688307 1455.9839   ml
## 85   4.44748913 1424.4938   ml
## 86   4.50809519 1390.2774   ml
## 87   4.56870125 1353.1362   ml
## 88   4.62930731 1312.8592   ml
## 89   4.68991337 1269.2215   ml
## 90   4.75051943 1221.9842   ml
## 91   4.81112550 1170.8932   ml
## 92   4.87173156 1115.6781   ml
## 93   4.93233762 1056.0517   ml
## 94   4.99294368  991.7086   ml
## 95   5.05354974  922.3243   ml
## 96   5.11415580  847.5540   ml
## 97   5.17476186  767.0316   ml
## 98   5.23536792  680.3679   ml
## 99   5.29597398  587.1498   ml
## 100  5.35658004  486.9384   ml
## 101 -0.64341996 1734.4760 true
## 102 -0.58281390 1734.7197 true
## 103 -0.52220784 1734.9710 true
## 104 -0.46160178 1735.2299 true
## 105 -0.40099572 1735.4968 true
## 106 -0.34038966 1735.7719 true
## 107 -0.27978360 1736.0554 true
## 108 -0.21917753 1736.3475 true
## 109 -0.15857147 1736.6484 true
## 110 -0.09796541 1736.9583 true
## 111 -0.03735935 1737.2775 true
## 112  0.02324671 1737.6060 true
## 113  0.08385277 1737.9440 true
## 114  0.14445883 1738.2917 true
## 115  0.20506489 1738.6491 true
## 116  0.26567095 1739.0164 true
## 117  0.32627701 1739.3935 true
## 118  0.38688307 1739.7805 true
## 119  0.44748913 1740.1773 true
## 120  0.50809519 1740.5837 true
## 121  0.56870125 1740.9996 true
## 122  0.62930731 1741.4248 true
## 123  0.68991337 1741.8589 true
## 124  0.75051943 1742.3014 true
## 125  0.81112550 1742.7519 true
## 126  0.87173156 1743.2096 true
## 127  0.93233762 1743.6738 true
## 128  0.99294368 1744.1436 true
## 129  1.05354974 1744.6179 true
## 130  1.11415580 1745.0953 true
## 131  1.17476186 1745.5744 true
## 132  1.23536792 1746.0534 true
## 133  1.29597398 1746.5304 true
## 134  1.35658004 1747.0032 true
```

```
## 135  1.41718610 1747.4692 true
## 136  1.47779216 1747.9254 true
## 137  1.53839822 1748.3687 true
## 138  1.59900428 1748.7953 true
## 139  1.65961034 1749.2012 true
## 140  1.72021640 1749.5818 true
## 141  1.78082247 1749.9318 true
## 142  1.84142853 1750.2457 true
## 143  1.90203459 1750.5169 true
## 144  1.96264065 1750.7386 true
## 145  2.02324671 1750.9028 true
## 146  2.08385277 1751.0010 true
## 147  2.14445883 1751.0237 true
## 148  2.20506489 1750.9603 true
## 149  2.26567095 1750.7994 true
## 150  2.32627701 1750.5283 true
## 151  2.38688307 1750.1332 true
## 152  2.44748913 1749.5989 true
## 153  2.50809519 1748.9089 true
## 154  2.56870125 1748.0450 true
## 155  2.62930731 1746.9875 true
## 156  2.68991337 1745.7149 true
## 157  2.75051943 1744.2037 true
## 158  2.81112550 1742.4286 true
## 159  2.87173156 1740.3620 true
## 160  2.93233762 1737.9738 true
## 161  2.99294368 1735.2317 true
## 162  3.05354974 1732.1005 true
## 163  3.11415580 1728.5422 true
## 164  3.17476186 1724.5158 true
## 165  3.23536792 1719.9771 true
## 166  3.29597398 1714.8782 true
## 167  3.35658004 1709.1677 true
## 168  3.41718610 1702.7901 true
## 169  3.47779216 1695.6859 true
## 170  3.53839822 1687.7909 true
## 171  3.59900428 1679.0363 true
## 172  3.65961034 1669.3482 true
## 173  3.72021640 1658.6473 true
## 174  3.78082247 1646.8486 true
## 175  3.84142853 1633.8611 true
## 176  3.90203459 1619.5874 true
## 177  3.96264065 1603.9231 true
## 178  4.02324671 1586.7570 true
## 179  4.08385277 1567.9697 true
## 180  4.14445883 1547.4341 true
## 181  4.20506489 1525.0144 true
## 182  4.26567095 1500.5656 true
## 183  4.32627701 1473.9333 true
## 184  4.38688307 1444.9526 true
## 185  4.44748913 1413.4482 true
## 186  4.50809519 1379.2329 true
## 187  4.56870125 1342.1078 true
## 188  4.62930731 1301.8610 true
```

```
## 189  4.68991337 1258.2671 true
## 190  4.75051943 1211.0862 true
## 191  4.81112550 1160.0635 true
## 192  4.87173156 1104.9278 true
## 193  4.93233762 1045.3911 true
## 194  4.99294368  981.1471 true
## 195  5.05354974  911.8705 true
## 196  5.11415580  837.2157 true
## 197  5.17476186  756.8156 true
## 198  5.23536792  670.2804 true
## 199  5.29597398  577.1961 true
## 200  5.35658004  477.1230 true
```

```
# theta2
const <- 3
theta2.v <- seq(ML.optim$par[2] - const,
                ML.optim$par[2] + const,length.out = 100)
plotl.th2 <- toplot.univ.comp(theta2.v, 2, 'theta2', t.parms, ML.optim$par,
                          ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotl.th2, cols = 2)
```






```
##              x          y  typ
## 1   -7.257641  1300.54477   ml
## 2   -7.197035  1307.34157   ml
## 3   -7.136429  1314.39182   ml
## 4   -7.075823  1321.69796   ml
## 5   -7.015217  1329.26181   ml
```

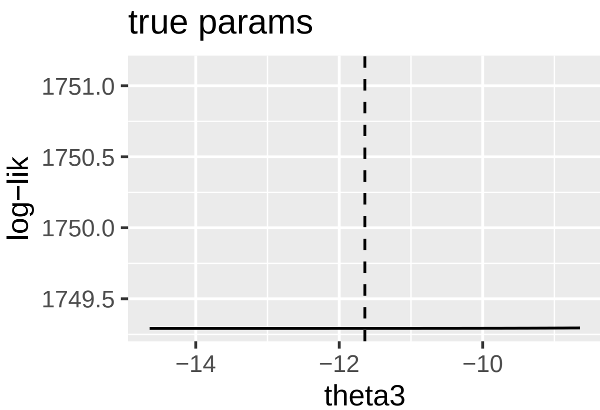
```
## 6   -6.954610  1337.08454   ml
## 7   -6.894004  1345.16663   ml
## 8   -6.833398  1353.50780   ml
## 9   -6.772792  1362.10696   ml
## 10  -6.712186  1370.96219   ml
## 11  -6.651580  1380.07065   ml
## 12  -6.590974  1389.42853   ml
## 13  -6.530368  1399.03106   ml
## 14  -6.469762  1408.87238   ml
## 15  -6.409156  1418.94555   ml
## 16  -6.348550  1429.24246   ml
## 17  -6.287944  1439.75382   ml
## 18  -6.227338  1450.46908   ml
## 19  -6.166732  1461.37640   ml
## 20  -6.106126  1472.46259   ml
## 21  -6.045520  1483.71306   ml
## 22  -5.984914  1495.11180   ml
## 23  -5.924307  1506.64127   ml
## 24  -5.863701  1518.28242   ml
## 25  -5.803095  1530.01455   ml
## 26  -5.742489  1541.81535   ml
## 27  -5.681883  1553.66078   ml
## 28  -5.621277  1565.52500   ml
## 29  -5.560671  1577.38037   ml
## 30  -5.500065  1589.19731   ml
## 31  -5.439459  1600.94428   ml
## 32  -5.378853  1612.58765   ml
## 33  -5.318247  1624.09170   ml
## 34  -5.257641  1635.41843   ml
## 35  -5.197035  1646.52753   ml
## 36  -5.136429  1657.37629   ml
## 37  -5.075823  1667.91942   ml
## 38  -5.015217  1678.10900   ml
## 39  -4.954610  1687.89430   ml
## 40  -4.894004  1697.22169   ml
## 41  -4.833398  1706.03446   ml
## 42  -4.772792  1714.27263   ml
## 43  -4.712186  1721.87287   ml
## 44  -4.651580  1728.76822   ml
## 45  -4.590974  1734.88794   ml
## 46  -4.530368  1740.15727   ml
## 47  -4.469762  1744.49723   ml
## 48  -4.409156  1747.82436   ml
## 49  -4.348550  1750.05043   ml
## 50  -4.287944  1751.08221   ml
## 51  -4.227338  1750.82114   ml
## 52  -4.166732  1749.16298   ml
## 53  -4.106126  1745.99754   ml
## 54  -4.045520  1741.20822   ml
## 55  -3.984914  1734.67171   ml
## 56  -3.924307  1726.25750   ml
## 57  -3.863701  1715.82748   ml
## 58  -3.803095  1703.23544   ml
## 59  -3.742489  1688.32657   ml
```

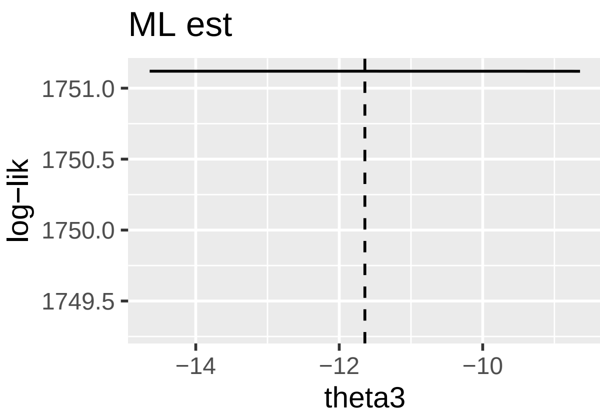
```
## 60  -3.681883  1670.93694   ml
## 61  -3.621277  1650.89290   ml
## 62  -3.560671  1628.01051   ml
## 63  -3.500065  1602.09484   ml
## 64  -3.439459  1572.93931   ml
## 65  -3.378853  1540.32497   ml
## 66  -3.318247  1504.01969   ml
## 67  -3.257641  1463.77734   ml
## 68  -3.197035  1419.33694   ml
## 69  -3.136429  1370.42168   ml
## 70  -3.075823  1316.73792   ml
## 71  -3.015217  1257.97418   ml
## 72  -2.954610  1193.79998   ml
## 73  -2.894004  1123.86465   ml
## 74  -2.833398  1047.79604   ml
## 75  -2.772792   965.19922   ml
## 76  -2.712186   875.65499   ml
## 77  -2.651580   778.71838   ml
## 78  -2.590974   673.91704   ml
## 79  -2.530368   560.74951   ml
## 80  -2.469762   438.68336   ml
## 81  -2.409156   307.15329   ml
## 82  -2.348550   165.55906   ml
## 83  -2.287944    13.26327   ml
## 84  -2.227338  -150.41097   ml
## 85  -2.166732  -326.18248   ml
## 86  -2.106126  -514.81469   ml
## 87  -2.045520  -717.11839   ml
## 88  -1.984914  -933.95475   ml
## 89  -1.924307 -1166.23841   ml
## 90  -1.863701 -1414.94093   ml
## 91  -1.803095 -1681.09424   ml
## 92  -1.742489 -1965.79450   ml
## 93  -1.681883 -2270.20612   ml
## 94  -1.621277 -2595.56599   ml
## 95  -1.560671 -2943.18806   ml
## 96  -1.500065 -3314.46812   ml
## 97  -1.439459 -3710.88898   ml
## 98  -1.378853 -4134.02587   ml
## 99  -1.318247 -4585.55224   ml
## 100 -1.257641 -5067.24588   ml
## 101 -7.257641  1195.78391 true
## 102 -7.197035  1199.01927 true
## 103 -7.136429  1202.42024 true
## 104 -7.075823  1205.99335 true
## 105 -7.015217  1209.74518 true
## 106 -6.954610  1213.68238 true
## 107 -6.894004  1217.81158 true
## 108 -6.833398  1222.13941 true
## 109 -6.772792  1226.67245 true
## 110 -6.712186  1231.41720 true
## 111 -6.651580  1236.38007 true
## 112 -6.590974  1241.56732 true
## 113 -6.530368  1246.98500 true
```

```
## 114 -6.469762  1252.63898 true
## 115 -6.409156  1258.53484 true
## 116 -6.348550  1264.67785 true
## 117 -6.287944  1271.07294 true
## 118 -6.227338  1277.72462 true
## 119 -6.166732  1284.63696 true
## 120 -6.106126  1291.81352 true
## 121 -6.045520  1299.25732 true
## 122 -5.984914  1306.97075 true
## 123 -5.924307  1314.95557 true
## 124 -5.863701  1323.21279 true
## 125 -5.803095  1331.74269 true
## 126 -5.742489  1340.54468 true
## 127 -5.681883  1349.61735 true
## 128 -5.621277  1358.95830 true
## 129 -5.560671  1368.56419 true
## 130 -5.500065  1378.43062 true
## 131 -5.439459  1388.55209 true
## 132 -5.378853  1398.92198 true
## 133 -5.318247  1409.53244 true
## 134 -5.257641  1420.37440 true
## 135 -5.197035  1431.43748 true
## 136 -5.136429  1442.70993 true
## 137 -5.075823  1454.17863 true
## 138 -5.015217  1465.82897 true
## 139 -4.954610  1477.64487 true
## 140 -4.894004  1489.60868 true
## 141 -4.833398  1501.70112 true
## 142 -4.772792  1513.90129 true
## 143 -4.712186  1526.18652 true
## 144 -4.651580  1538.53241 true
## 145 -4.590974  1550.91269 true
## 146 -4.530368  1563.29921 true
## 147 -4.469762  1575.66184 true
## 148 -4.409156  1587.96842 true
## 149 -4.348550  1600.18468 true
## 150 -4.287944  1612.27415 true
## 151 -4.227338  1624.19807 true
## 152 -4.166732  1635.91534 true
## 153 -4.106126  1647.38235 true
## 154 -4.045520  1658.55293 true
## 155 -3.984914  1669.37822 true
## 156 -3.924307  1679.80651 true
## 157 -3.863701  1689.78318 true
## 158 -3.803095  1699.25045 true
## 159 -3.742489  1708.14734 true
## 160 -3.681883  1716.40939 true
## 161 -3.621277  1723.96857 true
## 162 -3.560671  1730.75303 true
## 163 -3.500065  1736.68689 true
## 164 -3.439459  1741.69006 true
## 165 -3.378853  1745.67793 true
## 166 -3.318247  1748.56117 true
## 167 -3.257641  1750.24542 true
```

```
## 168 -3.197035  1750.63100 true
## 169 -3.136429  1749.61260 true
## 170 -3.075823  1747.07891 true
## 171 -3.015217  1742.91228 true
## 172 -2.954610  1736.98833 true
## 173 -2.894004  1729.17553 true
## 174 -2.833398  1719.33476 true
## 175 -2.772792  1707.31881 true
## 176 -2.712186  1692.97192 true
## 177 -2.651580  1676.12921 true
## 178 -2.590974  1656.61614 true
## 179 -2.530368  1634.24784 true
## 180 -2.469762  1608.82855 true
## 181 -2.409156  1580.15083 true
## 182 -2.348550  1547.99492 true
## 183 -2.287944  1512.12788 true
## 184 -2.227338  1472.30282 true
## 185 -2.166732  1428.25797 true
## 186 -2.106126  1379.71578 true
## 187 -2.045520  1326.38188 true
## 188 -1.984914  1267.94405 true
## 189 -1.924307  1204.07107 true
## 190 -1.863701  1134.41154 true
## 191 -1.803095  1058.59260 true
## 192 -1.742489   976.21854 true
## 193 -1.681883   886.86943 true
## 194 -1.621277   790.09951 true
## 195 -1.560671   685.43564 true
## 196 -1.500065   572.37550 true
## 197 -1.439459   450.38583 true
## 198 -1.378853   318.90040 true
## 199 -1.318247   177.31802 true
## 200 -1.257641    25.00027 true
```

```r
# theta3
const <- 3
theta3.v <- seq(ML.optim$par[3] - const,
                ML.optim$par[3] + const,length.out = 100)
plotl.th3 <- toplot.univ.comp(theta3.v, 3, 'theta3', t.parms, ML.optim$par,
                          ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotl.th3, cols = 2)
```






```
##              x        y  typ
## 1   -14.642860 1751.120   ml
## 2   -14.582254 1751.120   ml
## 3   -14.521648 1751.120   ml
## 4   -14.461041 1751.120   ml
## 5   -14.400435 1751.120   ml
## 6   -14.339829 1751.120   ml
## 7   -14.279223 1751.120   ml
## 8   -14.218617 1751.120   ml
## 9   -14.158011 1751.120   ml
## 10  -14.097405 1751.120   ml
## 11  -14.036799 1751.120   ml
## 12  -13.976193 1751.120   ml
## 13  -13.915587 1751.120   ml
## 14  -13.854981 1751.120   ml
## 15  -13.794375 1751.120   ml
## 16  -13.733769 1751.120   ml
## 17  -13.673163 1751.120   ml
## 18  -13.612557 1751.120   ml
## 19  -13.551951 1751.120   ml
## 20  -13.491344 1751.120   ml
## 21  -13.430738 1751.120   ml
## 22  -13.370132 1751.120   ml
## 23  -13.309526 1751.120   ml
## 24  -13.248920 1751.120   ml
## 25  -13.188314 1751.120   ml
## 26  -13.127708 1751.120   ml
```

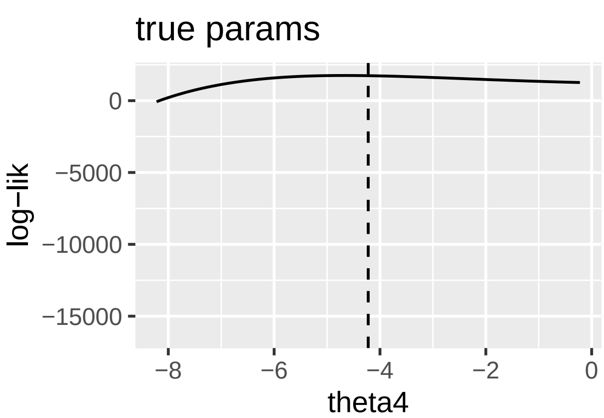
```
## 27  -13.067102 1751.120   ml
## 28  -13.006496 1751.120   ml
## 29  -12.945890 1751.120   ml
## 30  -12.885284 1751.120   ml
## 31  -12.824678 1751.120   ml
## 32  -12.764072 1751.120   ml
## 33  -12.703466 1751.120   ml
## 34  -12.642860 1751.120   ml
## 35  -12.582254 1751.120   ml
## 36  -12.521648 1751.120   ml
## 37  -12.461041 1751.120   ml
## 38  -12.400435 1751.120   ml
## 39  -12.339829 1751.120   ml
## 40  -12.279223 1751.120   ml
## 41  -12.218617 1751.120   ml
## 42  -12.158011 1751.120   ml
## 43  -12.097405 1751.120   ml
## 44  -12.036799 1751.120   ml
## 45  -11.976193 1751.120   ml
## 46  -11.915587 1751.120   ml
## 47  -11.854981 1751.120   ml
## 48  -11.794375 1751.120   ml
## 49  -11.733769 1751.120   ml
## 50  -11.673163 1751.120   ml
## 51  -11.612557 1751.120   ml
## 52  -11.551951 1751.120   ml
## 53  -11.491344 1751.120   ml
## 54  -11.430738 1751.120   ml
## 55  -11.370132 1751.120   ml
## 56  -11.309526 1751.120   ml
## 57  -11.248920 1751.120   ml
## 58  -11.188314 1751.120   ml
## 59  -11.127708 1751.120   ml
## 60  -11.067102 1751.120   ml
## 61  -11.006496 1751.120   ml
## 62  -10.945890 1751.120   ml
## 63  -10.885284 1751.120   ml
## 64  -10.824678 1751.120   ml
## 65  -10.764072 1751.120   ml
## 66  -10.703466 1751.120   ml
## 67  -10.642860 1751.120   ml
## 68  -10.582254 1751.120   ml
## 69  -10.521648 1751.120   ml
## 70  -10.461041 1751.120   ml
## 71  -10.400435 1751.120   ml
## 72  -10.339829 1751.120   ml
## 73  -10.279223 1751.120   ml
## 74  -10.218617 1751.120   ml
## 75  -10.158011 1751.120   ml
## 76  -10.097405 1751.120   ml
## 77  -10.036799 1751.120   ml
## 78   -9.976193 1751.120   ml
## 79   -9.915587 1751.120   ml
## 80   -9.854981 1751.120   ml
```

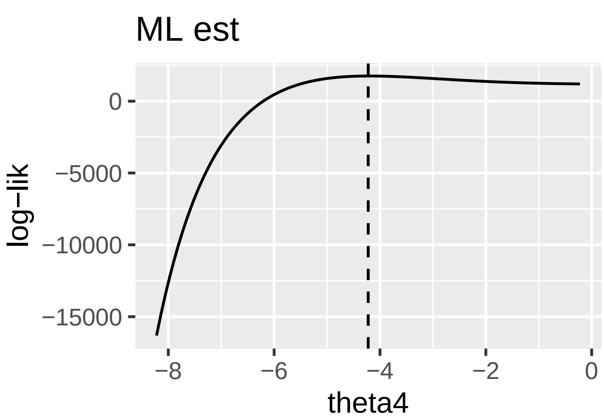
```
## 81   -9.794375 1751.120   ml
## 82   -9.733769 1751.120   ml
## 83   -9.673163 1751.120   ml
## 84   -9.612557 1751.120   ml
## 85   -9.551951 1751.120   ml
## 86   -9.491344 1751.119   ml
## 87   -9.430738 1751.119   ml
## 88   -9.370132 1751.119   ml
## 89   -9.309526 1751.119   ml
## 90   -9.248920 1751.119   ml
## 91   -9.188314 1751.119   ml
## 92   -9.127708 1751.119   ml
## 93   -9.067102 1751.119   ml
## 94   -9.006496 1751.119   ml
## 95   -8.945890 1751.119   ml
## 96   -8.885284 1751.119   ml
## 97   -8.824678 1751.119   ml
## 98   -8.764072 1751.119   ml
## 99   -8.703466 1751.119   ml
## 100  -8.642860 1751.119   ml
## 101 -14.642860 1749.293 true
## 102 -14.582254 1749.293 true
## 103 -14.521648 1749.293 true
## 104 -14.461041 1749.293 true
## 105 -14.400435 1749.293 true
## 106 -14.339829 1749.293 true
## 107 -14.279223 1749.293 true
## 108 -14.218617 1749.293 true
## 109 -14.158011 1749.293 true
## 110 -14.097405 1749.293 true
## 111 -14.036799 1749.293 true
## 112 -13.976193 1749.293 true
## 113 -13.915587 1749.293 true
## 114 -13.854981 1749.293 true
## 115 -13.794375 1749.293 true
## 116 -13.733769 1749.293 true
## 117 -13.673163 1749.293 true
## 118 -13.612557 1749.293 true
## 119 -13.551951 1749.293 true
## 120 -13.491344 1749.293 true
## 121 -13.430738 1749.293 true
## 122 -13.370132 1749.293 true
## 123 -13.309526 1749.293 true
## 124 -13.248920 1749.293 true
## 125 -13.188314 1749.293 true
## 126 -13.127708 1749.293 true
## 127 -13.067102 1749.293 true
## 128 -13.006496 1749.293 true
## 129 -12.945890 1749.293 true
## 130 -12.885284 1749.293 true
## 131 -12.824678 1749.293 true
## 132 -12.764072 1749.293 true
## 133 -12.703466 1749.293 true
## 134 -12.642860 1749.293 true
```

```
## 135 -12.582254 1749.293 true
## 136 -12.521648 1749.293 true
## 137 -12.461041 1749.293 true
## 138 -12.400435 1749.293 true
## 139 -12.339829 1749.293 true
## 140 -12.279223 1749.293 true
## 141 -12.218617 1749.293 true
## 142 -12.158011 1749.293 true
## 143 -12.097405 1749.293 true
## 144 -12.036799 1749.293 true
## 145 -11.976193 1749.293 true
## 146 -11.915587 1749.293 true
## 147 -11.854981 1749.293 true
## 148 -11.794375 1749.293 true
## 149 -11.733769 1749.293 true
## 150 -11.673163 1749.293 true
## 151 -11.612557 1749.293 true
## 152 -11.551951 1749.293 true
## 153 -11.491344 1749.293 true
## 154 -11.430738 1749.293 true
## 155 -11.370132 1749.293 true
## 156 -11.309526 1749.293 true
## 157 -11.248920 1749.293 true
## 158 -11.188314 1749.293 true
## 159 -11.127708 1749.293 true
## 160 -11.067102 1749.293 true
## 161 -11.006496 1749.293 true
## 162 -10.945890 1749.293 true
## 163 -10.885284 1749.293 true
## 164 -10.824678 1749.293 true
## 165 -10.764072 1749.293 true
## 166 -10.703466 1749.293 true
## 167 -10.642860 1749.293 true
## 168 -10.582254 1749.293 true
## 169 -10.521648 1749.293 true
## 170 -10.461041 1749.293 true
## 171 -10.400435 1749.293 true
## 172 -10.339829 1749.293 true
## 173 -10.279223 1749.293 true
## 174 -10.218617 1749.293 true
## 175 -10.158011 1749.293 true
## 176 -10.097405 1749.293 true
## 177 -10.036799 1749.293 true
## 178  -9.976193 1749.293 true
## 179  -9.915587 1749.293 true
## 180  -9.854981 1749.293 true
## 181  -9.794375 1749.293 true
## 182  -9.733769 1749.293 true
## 183  -9.673163 1749.293 true
## 184  -9.612557 1749.294 true
## 185  -9.551951 1749.294 true
## 186  -9.491344 1749.294 true
## 187  -9.430738 1749.294 true
## 188  -9.370132 1749.294 true
```

```
## 189  -9.309526 1749.294 true
## 190  -9.248920 1749.294 true
## 191  -9.188314 1749.294 true
## 192  -9.127708 1749.294 true
## 193  -9.067102 1749.294 true
## 194  -9.006496 1749.294 true
## 195  -8.945890 1749.294 true
## 196  -8.885284 1749.295 true
## 197  -8.824678 1749.295 true
## 198  -8.764072 1749.295 true
## 199  -8.703466 1749.295 true
## 200  -8.642860 1749.295 true
```

```
# theta4
const <- 4
theta4.v <- seq(ML.optim$par[4] - const,
                ML.optim$par[4] + const,length.out = 100)
plotl.th4 <- toplot.univ.comp(theta4.v, 4, 'theta4', t.parms, ML.optim$par,
                          ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotl.th4, cols = 2)
```






```
##             x            y  typ
## 1   -8.2221213 -16314.17423   ml
## 2   -8.1413132 -14879.93686   ml
## 3   -8.0605052 -13552.45160   ml
## 4   -7.9796971 -12323.88830   ml
## 5   -7.8988890 -11186.99572   ml
```

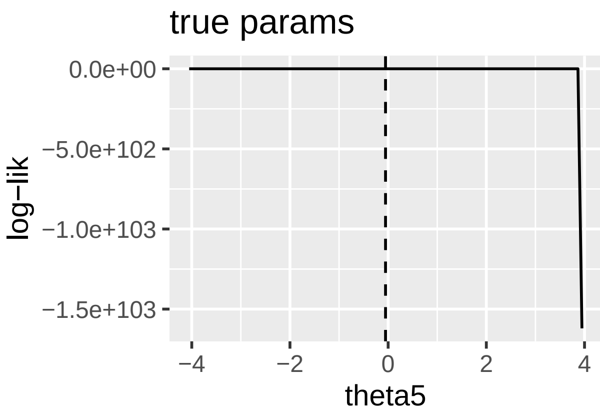
```
## 6   -7.8180809 -10135.05884   ml
## 7   -7.7372728  -9161.85924   ml
## 8   -7.6564648  -8261.63841   ml
## 9   -7.5756567  -7429.06377   ml
## 10  -7.4948486  -6659.19718   ml
## 11  -7.4140405  -5947.46577   ml
## 12  -7.3332324  -5289.63493   ml
## 13  -7.2524244  -4681.78321   ml
## 14  -7.1716163  -4120.27916   ml
## 15  -7.0908082  -3601.75982   ml
## 16  -7.0100001  -3123.11079   ml
## 17  -6.9291920  -2681.44772   ml
## 18  -6.8483839  -2274.09927   ml
## 19  -6.7675759  -1898.59121   ml
## 20  -6.6867678  -1552.63170   ml
## 21  -6.6059597  -1234.09773   ml
## 22  -6.5251516   -941.02242   ml
## 23  -6.4443435   -671.58339   ml
## 24  -6.3635355   -424.09186   ml
## 25  -6.2827274   -196.98264   ml
## 26  -6.2019193     11.19522   ml
## 27  -6.1211112    201.78702   ml
## 28  -6.0403031    376.04046   ml
## 29  -5.9594951    535.11299   ml
## 30  -5.8786870    680.07873   ml
## 31  -5.7978789    811.93482   ml
## 32  -5.7170708    931.60728   ml
## 33  -5.6362627   1039.95652   ml
## 34  -5.5554547   1137.78236   ml
## 35  -5.4746466   1225.82873   ml
## 36  -5.3938385   1304.78799   ml
## 37  -5.3130304   1375.30494   ml
## 38  -5.2322223   1437.98055   ml
## 39  -5.1514142   1493.37536   ml
## 40  -5.0706062   1542.01267   ml
## 41  -4.9897981   1584.38146   ml
## 42  -4.9089900   1620.93910   ml
## 43  -4.8281819   1652.11381   ml
## 44  -4.7473738   1678.30695   ml
## 45  -4.6665658   1699.89513   ml
## 46  -4.5857577   1717.23215   ml
## 47  -4.5049496   1730.65068   ml
## 48  -4.4241415   1740.46391   ml
## 49  -4.3433334   1746.96698   ml
## 50  -4.2625254   1750.43825   ml
## 51  -4.1817173   1751.14050   ml
## 52  -4.1009092   1749.32196   ml
## 53  -4.0201011   1745.21722   ml
## 54  -3.9392930   1739.04807   ml
## 55  -3.8584850   1731.02427   ml
## 56  -3.7776769   1721.34412   ml
## 57  -3.6968688   1710.19511   ml
## 58  -3.6160607   1697.75441   ml
## 59  -3.5352526   1684.18934   ml
```

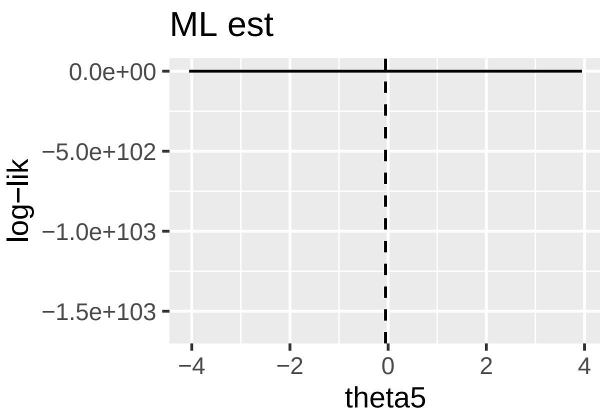
```
## 60  -3.4544446   1669.65778   ml
## 61  -3.3736365   1654.30856   ml
## 62  -3.2928284   1638.28179   ml
## 63  -3.2120203   1621.70916   ml
## 64  -3.1312122   1604.71424   ml
## 65  -3.0504041   1587.41269   ml
## 66  -2.9695961   1569.91245   ml
## 67  -2.8887880   1552.31398   ml
## 68  -2.8079799   1534.71033   ml
## 69  -2.7271718   1517.18731   ml
## 70  -2.6463637   1499.82355   ml
## 71  -2.5655557   1482.69061   ml
## 72  -2.4847476   1465.85308   ml
## 73  -2.4039395   1449.36860   ml
## 74  -2.3231314   1433.28804   ml
## 75  -2.2423233   1417.65556   ml
## 76  -2.1615153   1402.50876   ml
## 77  -2.0807072   1387.87893   ml
## 78  -1.9998991   1373.79123   ml
## 79  -1.9190910   1360.26497   ml
## 80  -1.8382829   1347.31395   ml
## 81  -1.7574749   1334.94683   ml
## 82  -1.6766668   1323.16746   ml
## 83  -1.5958587   1311.97540   ml
## 84  -1.5150506   1301.36626   ml
## 85  -1.4342425   1291.33221   ml
## 86  -1.3534345   1281.86241   ml
## 87  -1.2726264   1272.94344   ml
## 88  -1.1918183   1264.55971   ml
## 89  -1.1110102   1256.69388   ml
## 90  -1.0302021   1249.32722   ml
## 91  -0.9493940   1242.43994   ml
## 92  -0.8685860   1236.01149   ml
## 93  -0.7877779   1230.02086   ml
## 94  -0.7069698   1224.44679   ml
## 95  -0.6261617   1219.26799   ml
## 96  -0.5453536   1214.46332   ml
## 97  -0.4645456   1210.01194   ml
## 98  -0.3837375   1205.89341   ml
## 99  -0.3029294   1202.08785   ml
## 100 -0.2221213   1198.57597   ml
## 101 -8.2221213    -73.59652 true
## 102 -8.1413132     34.64001 true
## 103 -8.0605052    138.11334 true
## 104 -7.9796971    236.99201 true
## 105 -7.8988890    331.43754 true
## 106 -7.8180809    421.60473 true
## 107 -7.7372728    507.64201 true
## 108 -7.6564648    589.69172 true
## 109 -7.5756567    667.89039 true
## 110 -7.4948486    742.36906 true
## 111 -7.4140405    813.25353 true
## 112 -7.3332324    880.66465 true
## 113 -7.2524244    944.71854 true
```

```
## 114 -7.1716163   1005.52694 true
## 115 -7.0908082   1063.19734 true
## 116 -7.0100001   1117.83331 true
## 117 -6.9291920   1169.53470 true
## 118 -6.8483839   1218.39787 true
## 119 -6.7675759   1264.51590 true
## 120 -6.6867678   1307.97880 true
## 121 -6.6059597   1348.87372 true
## 122 -6.5251516   1387.28511 true
## 123 -6.4443435   1423.29496 true
## 124 -6.3635355   1456.98290 true
## 125 -6.2827274   1488.42641 true
## 126 -6.2019193   1517.70094 true
## 127 -6.1211112   1544.88007 true
## 128 -6.0403031   1570.03563 true
## 129 -5.9594951   1593.23780 true
## 130 -5.8786870   1614.55525 true
## 131 -5.7978789   1634.05520 true
## 132 -5.7170708   1651.80352 true
## 133 -5.6362627   1667.86478 true
## 134 -5.5554547   1682.30235 true
## 135 -5.4746466   1695.17842 true
## 136 -5.3938385   1706.55403 true
## 137 -5.3130304   1716.48912 true
## 138 -5.2322223   1725.04257 true
## 139 -5.1514142   1732.27216 true
## 140 -5.0706062   1738.23462 true
## 141 -4.9897981   1742.98564 true
## 142 -4.9089900   1746.57980 true
## 143 -4.8281819   1749.07063 true
## 144 -4.7473738   1750.51057 true
## 145 -4.6665658   1750.95095 true
## 146 -4.5857577   1750.44197 true
## 147 -4.5049496   1749.03268 true
## 148 -4.4241415   1746.77099 true
## 149 -4.3433334   1743.70361 true
## 150 -4.2625254   1739.87606 true
## 151 -4.1817173   1735.33266 true
## 152 -4.1009092   1730.11650 true
## 153 -4.0201011   1724.26946 true
## 154 -3.9392930   1717.83217 true
## 155 -3.8584850   1710.84406 true
## 156 -3.7776769   1703.34334 true
## 157 -3.6968688   1695.36701 true
## 158 -3.6160607   1686.95092 true
## 159 -3.5352526   1678.12972 true
## 160 -3.4544446   1668.93694 true
## 161 -3.3736365   1659.40503 true
## 162 -3.2928284   1649.56532 true
## 163 -3.2120203   1639.44811 true
## 164 -3.1312122   1629.08270 true
## 165 -3.0504041   1618.49737 true
## 166 -2.9695961   1607.71945 true
## 167 -2.8887880   1596.77533 true
```

```
## 168 -2.8079799   1585.69047 true
## 169 -2.7271718   1574.48942 true
## 170 -2.6463637   1563.19584 true
## 171 -2.5655557   1551.83247 true
## 172 -2.4847476   1540.42121 true
## 173 -2.4039395   1528.98304 true
## 174 -2.3231314   1517.53807 true
## 175 -2.2423233   1506.10554 true
## 176 -2.1615153   1494.70380 true
## 177 -2.0807072   1483.35033 true
## 178 -1.9998991   1472.06174 true
## 179 -1.9190910   1460.85379 true
## 180 -1.8382829   1449.74139 true
## 181 -1.7574749   1438.73864 true
## 182 -1.6766668   1427.85881 true
## 183 -1.5958587   1417.11441 true
## 184 -1.5150506   1406.51718 true
## 185 -1.4342425   1396.07812 true
## 186 -1.3534345   1385.80754 true
## 187 -1.2726264   1375.71507 true
## 188 -1.1918183   1365.80969 true
## 189 -1.1110102   1356.09974 true
## 190 -1.0302021   1346.59295 true
## 191 -0.9493940   1337.29645 true
## 192 -0.8685860   1328.21678 true
## 193 -0.7877779   1319.35991 true
## 194 -0.7069698   1310.73121 true
## 195 -0.6261617   1302.33551 true
## 196 -0.5453536   1294.17703 true
## 197 -0.4645456   1286.25941 true
## 198 -0.3837375   1278.58570 true
## 199 -0.3029294   1271.15835 true
## 200 -0.2221213   1263.97921 true
```

```r
# theta5
const <- 4
theta5.v <- seq(ML.optim$par[5] - const,
                ML.optim$par[5] + const,length.out = 100)
plotl.th5 <- toplot.univ.comp(theta5.v, 5, 'theta5', t.parms, ML.optim$par,
                          ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotl.th5, cols = 2)
```






```
##               x            y  typ
## 1   -4.05306828  1.349580e+03   ml
## 2   -3.97226020  1.349973e+03   ml
## 3   -3.89145212  1.350400e+03   ml
## 4   -3.81064404  1.350865e+03   ml
## 5   -3.72983596  1.351371e+03   ml
## 6   -3.64902788  1.351923e+03   ml
## 7   -3.56821980  1.352524e+03   ml
## 8   -3.48741172  1.353179e+03   ml
## 9   -3.40660364  1.353893e+03   ml
## 10  -3.32579556  1.354673e+03   ml
## 11  -3.24498748  1.355524e+03   ml
## 12  -3.16417939  1.356454e+03   ml
## 13  -3.08337131  1.357469e+03   ml
## 14  -3.00256323  1.358581e+03   ml
## 15  -2.92175515  1.359796e+03   ml
## 16  -2.84094707  1.361128e+03   ml
## 17  -2.76013899  1.362588e+03   ml
## 18  -2.67933091  1.364189e+03   ml
## 19  -2.59852283  1.365946e+03   ml
## 20  -2.51771475  1.367878e+03   ml
## 21  -2.43690667  1.370003e+03   ml
## 22  -2.35609859  1.372343e+03   ml
## 23  -2.27529051  1.374923e+03   ml
## 24  -2.19448242  1.377771e+03   ml
## 25  -2.11367434  1.380919e+03   ml
## 26  -2.03286626  1.384403e+03   ml
```

```
## 27  -1.95205818   1.388264e+03   ml
## 28  -1.87125010   1.392548e+03   ml
## 29  -1.79044202   1.397311e+03   ml
## 30  -1.70963394   1.402612e+03   ml
## 31  -1.62882586   1.408522e+03   ml
## 32  -1.54801778   1.415121e+03   ml
## 33  -1.46720970   1.422499e+03   ml
## 34  -1.38640162   1.430761e+03   ml
## 35  -1.30559354   1.440024e+03   ml
## 36  -1.22478546   1.450420e+03   ml
## 37  -1.14397737   1.462097e+03   ml
## 38  -1.06316929   1.475218e+03   ml
## 39  -0.98236121   1.489959e+03   ml
## 40  -0.90155313   1.506503e+03   ml
## 41  -0.82074505   1.525031e+03   ml
## 42  -0.73993697   1.545707e+03   ml
## 43  -0.65912889   1.568645e+03   ml
## 44  -0.57832081   1.593867e+03   ml
## 45  -0.49751273   1.621222e+03   ml
## 46  -0.41670465   1.650267e+03   ml
## 47  -0.33589657   1.680071e+03   ml
## 48  -0.25508849   1.708901e+03   ml
## 49  -0.17428040   1.733713e+03   ml
## 50  -0.09347232   1.749313e+03   ml
## 51  -0.01266424   1.746949e+03   ml
## 52   0.06814384   1.711867e+03   ml
## 53   0.14895192   1.618972e+03   ml
## 54   0.22976000   1.424860e+03   ml
## 55   0.31056808   1.052674e+03   ml
## 56   0.39137616   3.623072e+02   ml
## 57   0.47218424  -9.105104e+02   ml
## 58   0.55299232  -3.283863e+03   ml
## 59   0.63380040  -7.817862e+03   ml
## 60   0.71460848  -1.678616e+04   ml
## 61   0.79541657  -3.532200e+04   ml
## 62   0.87622465  -7.568118e+04   ml
## 63   0.95703273  -1.689588e+05   ml
## 64   1.03784081  -3.994372e+05   ml
## 65   1.11864889  -1.012569e+06   ml
## 66   1.19945697  -2.781191e+06   ml
## 67   1.28026505  -8.354082e+06   ml
## 68   1.36107313  -2.768653e+07   ml
## 69   1.44188121  -1.021417e+08   ml
## 70   1.52268929  -4.233863e+08   ml
## 71   1.60349737  -1.991464e+09   ml
## 72   1.68430545  -1.074351e+10   ml
## 73   1.76511353  -6.724710e+10   ml
## 74   1.84592162  -4.945263e+11   ml
## 75   1.92672970  -4.330966e+12   ml
## 76   2.00753778  -4.584018e+13   ml
## 77   2.08834586  -5.957965e+14   ml
## 78   2.16915394  -9.674901e+15   ml
## 79   2.24996202  -2.000003e+17   ml
## 80   2.33077010  -5.371233e+18   ml
```

```
## 81   2.41157818  -1.915765e+20   ml
## 82   2.49238626  -9.294093e+21   ml
## 83   2.57319434  -6.293787e+23   ml
## 84   2.65400242  -6.118538e+25   ml
## 85   2.73481050  -8.802989e+27   ml
## 86   2.81561859  -1.937261e+30   ml
## 87   2.89642667  -6.758584e+32   ml
## 88   2.97723475  -3.885752e+35   ml
## 89   3.05804283  -3.839771e+38   ml
## 90   3.13885091  -6.825627e+41   ml
## 91   3.21965899  -2.293236e+45   ml
## 92   3.30046707  -1.536359e+49   ml
## 93   3.38127515  -2.175207e+53   ml
## 94   3.46208323  -6.931427e+57   ml
## 95   3.54289131  -5.322439e+62   ml
## 96   3.62369939  -1.060503e+68   ml
## 97   3.70450747  -5.941268e+73   ml
## 98   3.78531556  -1.020937e+80   ml
## 99   3.86612364  -5.913400e+86   ml
## 100  3.94693172  -1.278818e+94   ml
## 101 -4.05306828   1.560143e+03 true
## 102 -3.97226020   1.560791e+03 true
## 103 -3.89145212   1.561494e+03 true
## 104 -3.81064404   1.562257e+03 true
## 105 -3.72983596   1.563087e+03 true
## 106 -3.64902788   1.563989e+03 true
## 107 -3.56821980   1.564969e+03 true
## 108 -3.48741172   1.566034e+03 true
## 109 -3.40660364   1.567193e+03 true
## 110 -3.32579556   1.568453e+03 true
## 111 -3.24498748   1.569824e+03 true
## 112 -3.16417939   1.571315e+03 true
## 113 -3.08337131   1.572938e+03 true
## 114 -3.00256323   1.574704e+03 true
## 115 -2.92175515   1.576628e+03 true
## 116 -2.84094707   1.578722e+03 true
## 117 -2.76013899   1.581003e+03 true
## 118 -2.67933091   1.583489e+03 true
## 119 -2.59852283   1.586197e+03 true
## 120 -2.51771475   1.589149e+03 true
## 121 -2.43690667   1.592367e+03 true
## 122 -2.35609859   1.595875e+03 true
## 123 -2.27529051   1.599698e+03 true
## 124 -2.19448242   1.603867e+03 true
## 125 -2.11367434   1.608410e+03 true
## 126 -2.03286626   1.613360e+03 true
## 127 -1.95205818   1.618750e+03 true
## 128 -1.87125010   1.624616e+03 true
## 129 -1.79044202   1.630991e+03 true
## 130 -1.70963394   1.637909e+03 true
## 131 -1.62882586   1.645401e+03 true
## 132 -1.54801778   1.653491e+03 true
## 133 -1.46720970   1.662194e+03 true
## 134 -1.38640162   1.671505e+03 true
```

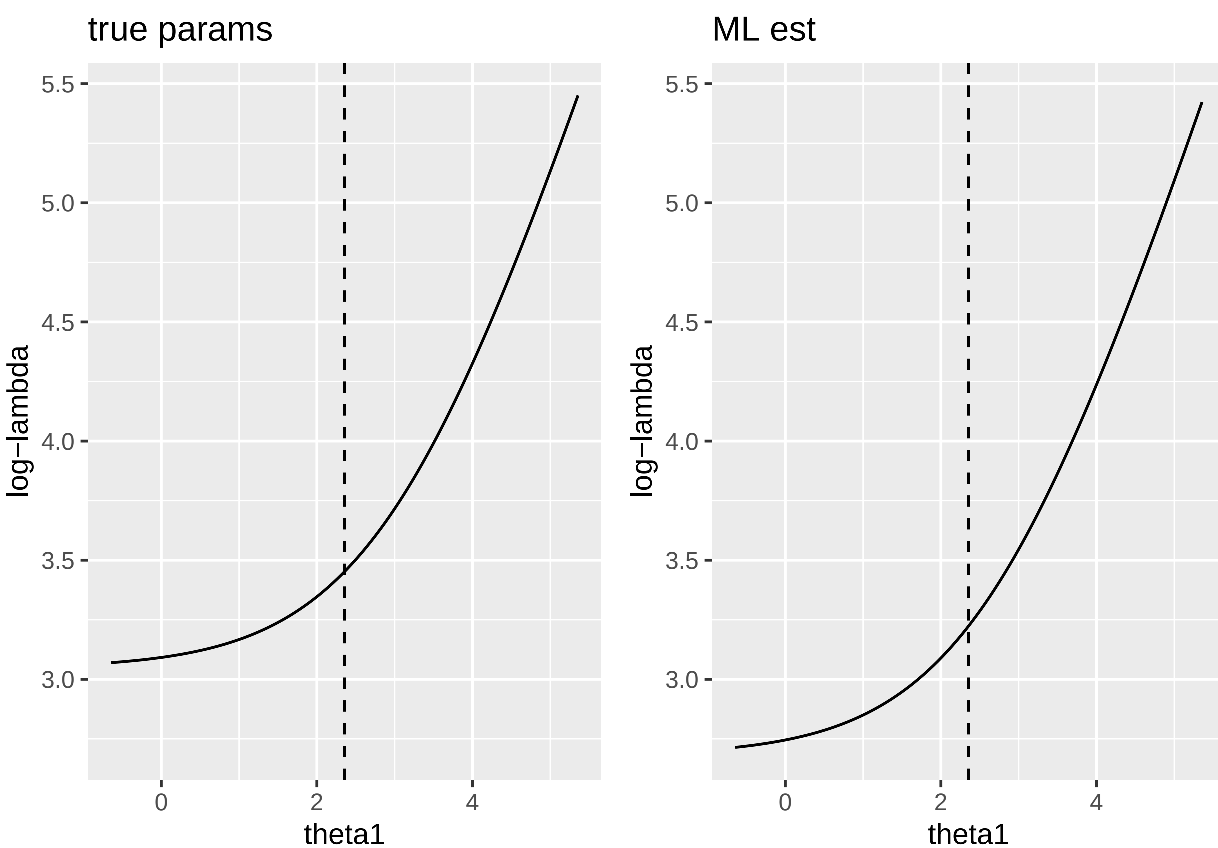
```
## 135 -1.30559354   1.681396e+03 true
## 136 -1.22478546   1.691799e+03 true
## 137 -1.14397737   1.702588e+03 true
## 138 -1.06316929   1.713551e+03 true
## 139 -0.98236121   1.724350e+03 true
## 140 -0.90155313   1.734463e+03 true
## 141 -0.82074505   1.743099e+03 true
## 142 -0.73993697   1.749072e+03 true
## 143 -0.65912889   1.750610e+03 true
## 144 -0.57832081   1.745077e+03 true
## 145 -0.49751273   1.728528e+03 true
## 146 -0.41670465   1.695038e+03 true
## 147 -0.33589657   1.635625e+03 true
## 148 -0.25508849   1.536505e+03 true
## 149 -0.17428040   1.376218e+03 true
## 150 -0.09347232   1.120739e+03 true
## 151 -0.01266424   7.149823e+02 true
## 152  0.06814384   6.757582e+01 true
## 153  0.14895192  -9.772256e+02 true
## 154  0.22976000  -2.693095e+03 true
## 155  0.31056808  -5.577366e+03 true
## 156  0.39137616  -1.056778e+04 true
## 157  0.47218424  -1.950518e+04 true
## 158  0.55299232  -3.616554e+04 true
## 159  0.63380040  -6.867307e+04 true
## 160  0.71460848  -1.354387e+05 true
## 161  0.79541657  -2.806050e+05 true
## 162  0.87622465  -6.166935e+05 true
## 163  0.95703273  -1.450256e+06 true
## 164  1.03784081  -3.679044e+06 true
## 165  1.11864889  -1.014697e+07 true
## 166  1.19945697  -3.066572e+07 true
## 167  1.28026505  -1.023780e+08 true
## 168  1.36107313  -3.808198e+08 true
## 169  1.44188121  -1.592865e+09 true
## 170  1.52268929  -7.566348e+09 true
## 171  1.60349737  -4.125687e+10 true
## 172  1.68430545  -2.612476e+11 true
## 173  1.76511353  -1.945453e+12 true
## 174  1.84592162  -1.727144e+13 true
## 175  1.92672970  -1.855243e+14 true
## 176  2.00753778  -2.450211e+15 true
## 177  2.08834586  -4.048450e+16 true
## 178  2.16915394  -8.527948e+17 true
## 179  2.24996202  -2.337478e+19 true
## 180  2.33077010  -8.523577e+20 true
## 181  2.41157818  -4.235467e+22 true
## 182  2.49238626  -2.943729e+24 true
## 183  2.57319434  -2.943578e+26 true
## 184  2.65400242  -4.366470e+28 true
## 185  2.73481050  -9.932946e+30 true
## 186  2.81561859  -3.592085e+33 true
## 187  2.89642667  -2.147236e+36 true
## 188  2.97723475  -2.213336e+39 true
```

```
## 189  3.05804283  -4.118749e+42 true
## 190  3.13885091  -1.454205e+46 true
## 191  3.21965899  -1.028105e+50 true
## 192  3.30046707  -1.543050e+54 true
## 193  3.38127515  -5.238044e+58 true
## 194  3.46208323  -4.307604e+63 true
## 195  3.54289131  -9.245318e+68 true
## 196  3.62369939  -5.614243e+74 true
## 197  3.70450747  -1.052831e+81 true
## 198  3.78531556  -6.704079e+87 true
## 199  3.86612364  -1.606627e+95 true
## 200  3.94693172 -1.620058e+103 true
```

```
# univariate likelihood analysis
toplot.loglambda.comp <- function(par.values, par.idx, par.name,
                             theta.par, ML.est, Ht, M0, Tlim, tt = 5){


  theta.m.true <- cbind(rep(t.parms[1], length(par.values)),
                        rep(t.parms[2], length(par.values)),
                        rep(t.parms[3], length(par.values)),
                        rep(t.parms[4], length(par.values)),
                        rep(t.parms[5], length(par.values)))

  theta.m.ML <- cbind(rep(ML.est[1], length(par.values)),
                      rep(ML.est[2], length(par.values)),
                      rep(ML.est[3], length(par.values)),
                      rep(ML.est[4], length(par.values)),
                      rep(ML.est[5], length(par.values)))

  theta.m.true[,par.idx] <- par.values
  theta.m.ML[,par.idx] <- par.values

  LL.true <- sapply(1:nrow(theta.m.true), function(i)
    log.lambda.ETAS.single(tt, theta.m.true[i,], Ht, M0))

  LL.ML <- sapply(1:nrow(theta.m.ML), function(i)
    log.lambda.ETAS.single(tt, theta.m.ML[i,], Ht, M0))

  ll.lim <- range(c(LL.true, LL.ML))

  df <- rbind(data.frame(x = par.values,
                         y = LL.ML,
                         typ = 'ml'),
              data.frame(x = par.values,
                         y = LL.true,
                         typ = 'true'))

  list(pl.true = ggplot(data.frame(x = par.values,
                                   y = LL.true), aes(x = x, y = y)) +
         geom_line() +
         geom_vline(xintercept = ML.est[par.idx], linetype = 2) +
         labs(title = 'true params') +
         ylab('log-lambda') +
```

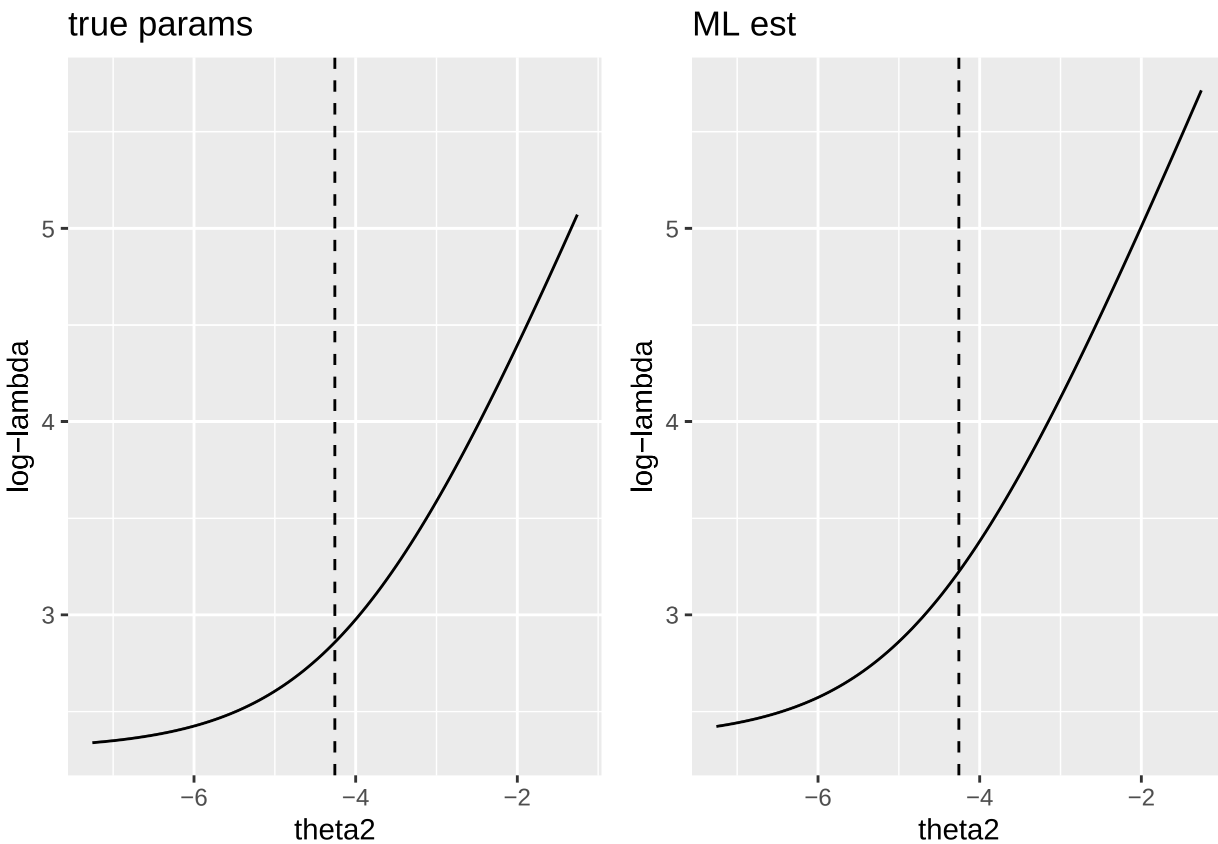
```
        xlab(par.name) +
        ylim(ll.lim),
      pl.ML = ggplot(data.frame(x = par.values,
                                y = LL.ML), aes(x = x, y = y)) +
        geom_line() +
        geom_vline(xintercept = ML.est[par.idx], linetype = 2)+
        labs(title = 'ML est')+
        ylab('log-lambda') +
        xlab(par.name) +
        ylim(ll.lim),
      df = df)
}
```

```
# theta1
# const <- 4
# theta1.v <- seq(ML.optim$par[4] - const,
#                  ML.optim$par[4] + const,length.out = 100)
plotllambda.th1 <- toplot.loglambda.comp(theta1.v, 1, 'theta1',
                                    theta.par = t.parms,
                                    ML.est = ML.optim$par,
                                    ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotllambda.th1[1:2], cols = 2)
```

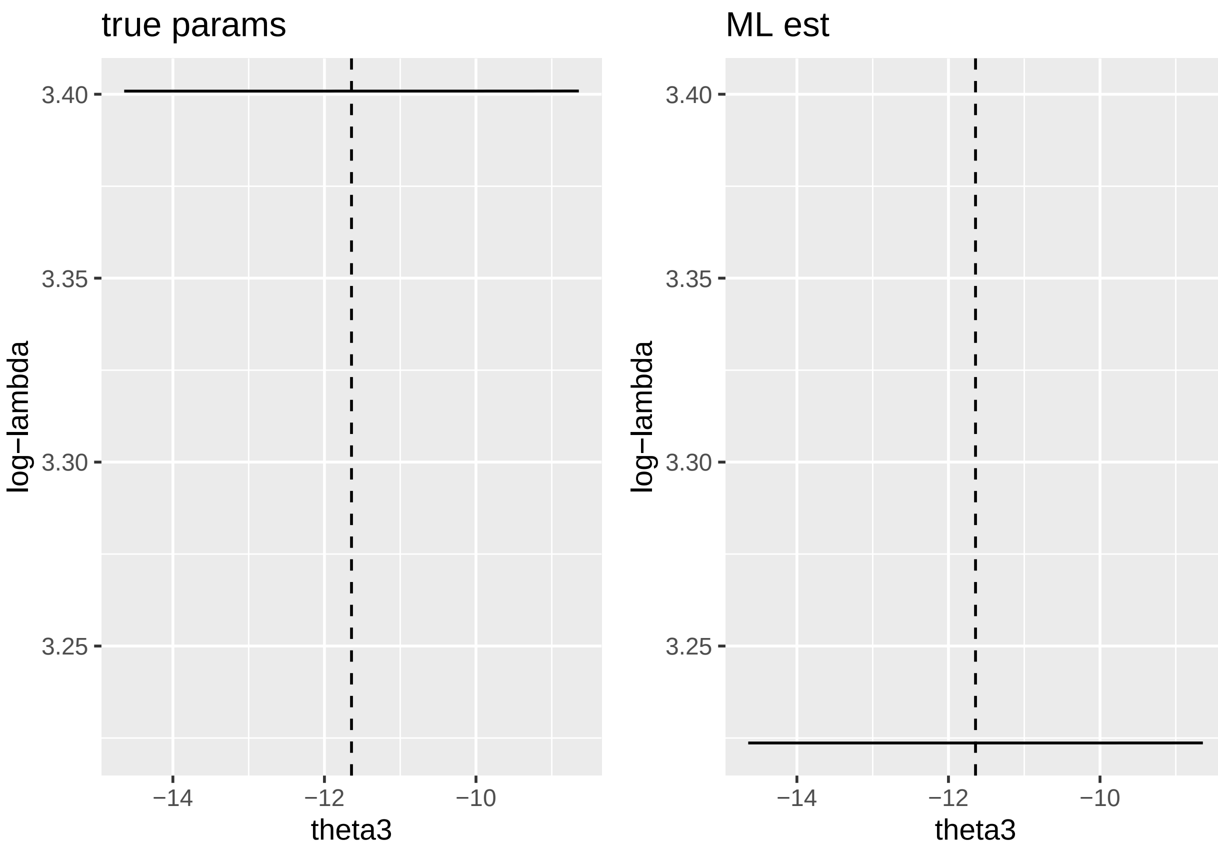



```
plotllambda.th2 <- toplot.loglambda.comp(theta2.v, 2, 'theta2',
                                    theta.par = t.parms,
                                    ML.est = ML.optim$par,
```

```r
                                      ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotllambda.th2[1:2], cols = 2)
```

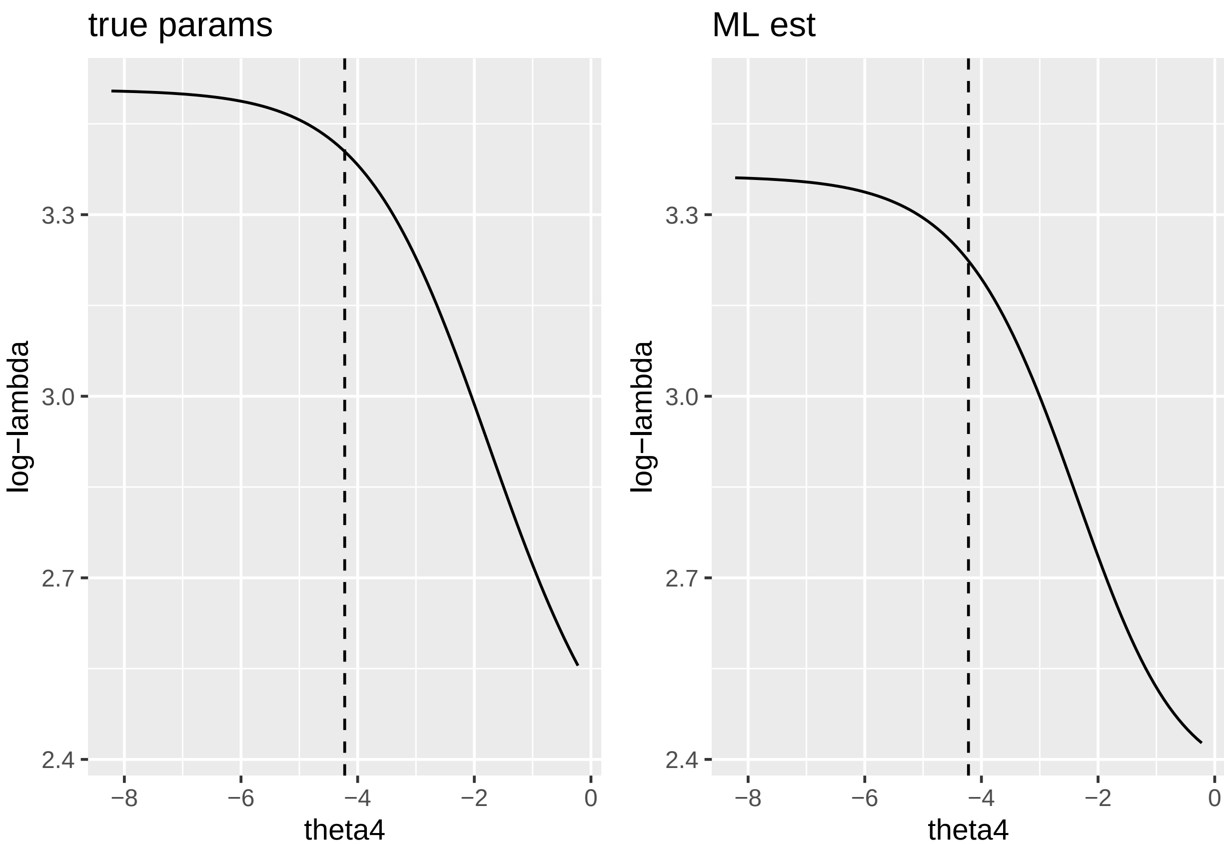



```r
plotllambda.th3 <- toplot.loglambda.comp(theta3.v, 3, 'theta3',
                                      theta.par = t.parms,
                                      ML.est = ML.optim$par,
                                      ss2, M0 = 2.5, Tlim = 10)


multiplot(plotlist = plotllambda.th3[1:2], cols = 2)
```

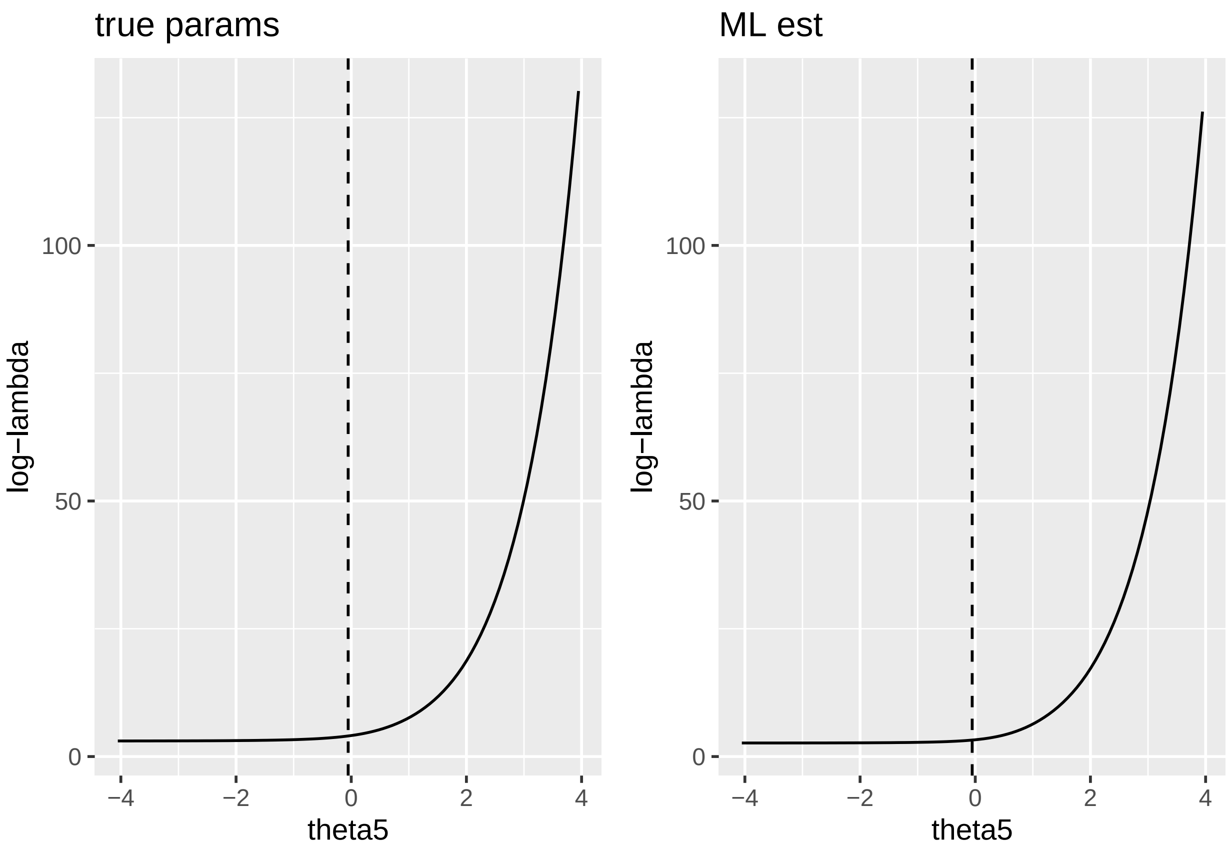



```
#plot(plotllambda.th3$df$x[plotllambda.th3$df$typ == 'ml'],
#     plotllambda.th3$df$y[plotllambda.th3$df$typ == 'ml'])
```

```
plotllambda.th4 <- toplot.loglambda.comp(theta4.v, 4, 'theta4',
                                    theta.par = t.parms,
                                    ML.est = ML.optim$par,
                                    ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotllambda.th4[1:2], cols = 2)
```



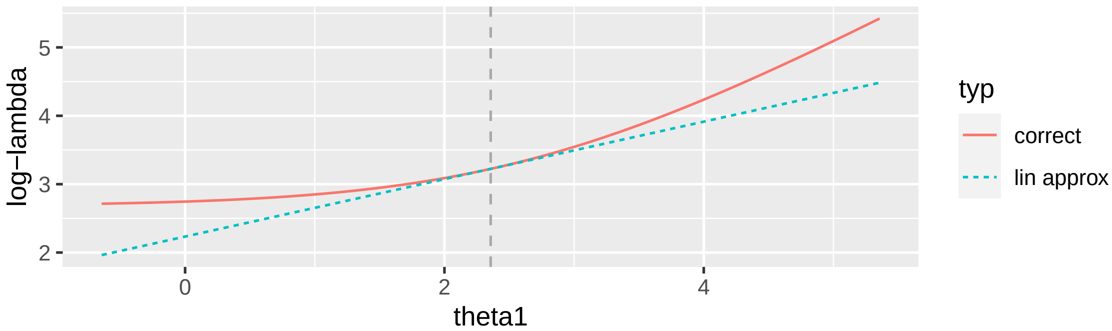

```
plotllambda.th5 <- toplot.loglambda.comp(theta5.v, 5, 'theta5',
                                  theta.par = t.parms,
                                  ML.est = ML.optim$par,
                                  ss2, M0 = 2.5, Tlim = 10)

multiplot(plotlist = plotllambda.th5[1:2], cols = 2)
```

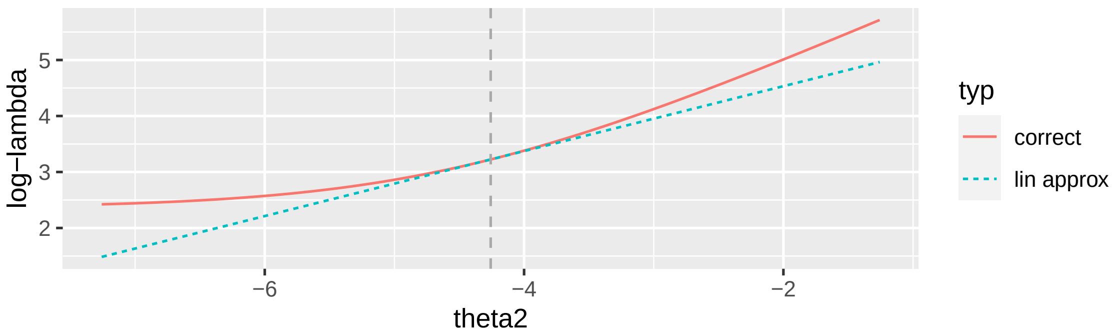


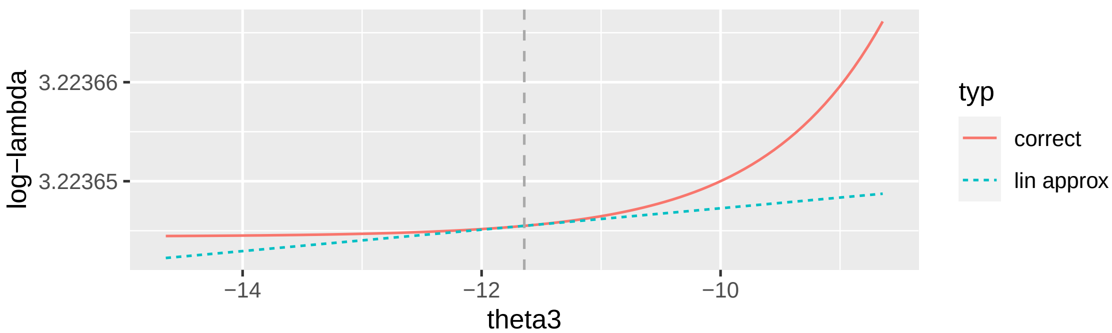

```
lin.loglambda.th1 <-
  toplot.loglambda.lin.comp(theta1.v, 1, 'theta1', ML.optim$par[1],
                            ML.optim$par, ss2, M0 = 2.5, Tlim = 10,
                            derFUN.list = list(log.lambda.der.th1,
                                               log.lambda.der.th2,
                                               log.lambda.der.th3,
                                               log.lambda.der.th4,
                                               log.lambda.der.th5))


lin.loglambda.th2 <-
  toplot.loglambda.lin.comp(theta2.v, 2, 'theta2', ML.optim$par[2],
                            ML.optim$par, ss2, M0 = 2.5, Tlim = 10,
                            derFUN.list = list(log.lambda.der.th1,
                                               log.lambda.der.th2,
                                               log.lambda.der.th3,
                                               log.lambda.der.th4,
                                               log.lambda.der.th5))


lin.loglambda.th3 <-
  toplot.loglambda.lin.comp(theta3.v, 3, 'theta3', ML.optim$par[3],
                            ML.optim$par, ss2, M0 = 2.5, Tlim = 10,
                            derFUN.list = list(log.lambda.der.th1,
                                               log.lambda.der.th2,
                                               log.lambda.der.th3,
                                               log.lambda.der.th4,
```

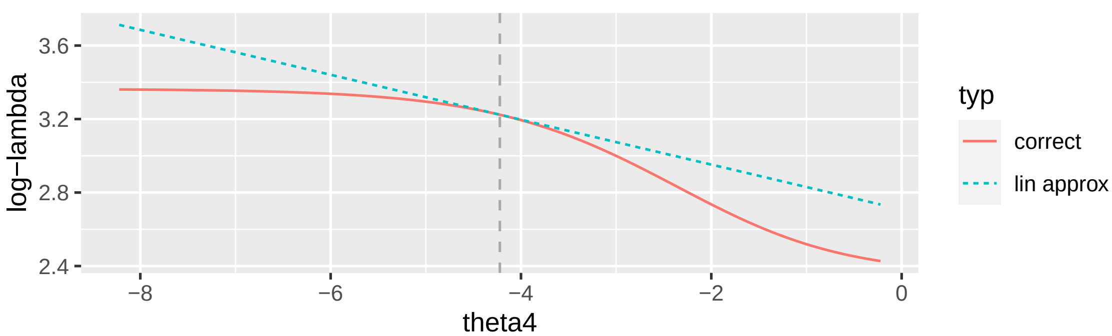
```
                                          log.lambda.der.th5))


lin.loglambda.th4 <-
  toplot.loglambda.lin.comp(theta4.v, 4, 'theta4', ML.optim$par[4],
                           ML.optim$par, ss2, M0 = 2.5, Tlim = 10,
                           derFUN.list = list(log.lambda.der.th1,
                                              log.lambda.der.th2,
                                              log.lambda.der.th3,
                                              log.lambda.der.th4,
                                              log.lambda.der.th5))


lin.loglambda.th5 <-
  toplot.loglambda.lin.comp(theta5.v, 5, 'theta5', ML.optim$par[5],
                           ML.optim$par, ss2, M0 = 2.5, Tlim = 10,
                           derFUN.list = list(log.lambda.der.th1,
                                              log.lambda.der.th2,
                                              log.lambda.der.th3,
                                              log.lambda.der.th4,
                                              log.lambda.der.th5))
```

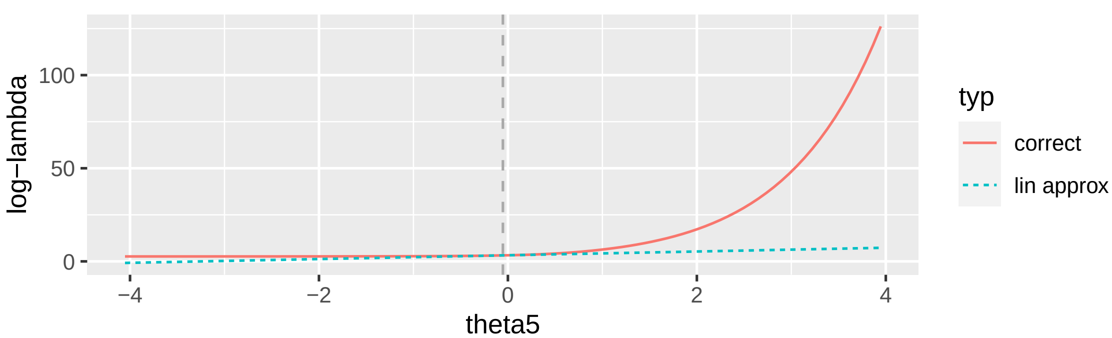
```
multiplot(lin.loglambda.th1$pl,
          lin.loglambda.th2$pl,
          lin.loglambda.th3$pl,
          lin.loglambda.th4$pl,
          lin.loglambda.th5$pl)
```


log−lambda
5
4
3
2
0
2
4
theta1
typ
correct
lin approx


log−lambda
5
4
3
2
−6
−4
−2
theta2
typ
correct
lin approx


log−lambda
3.22366
3.22365
−14
−12
−10
theta3
typ
correct
lin approx


log−lambda
3.6
3.2
2.8
2.4
−8
−6
−4
−2
0
theta4
typ
correct
lin approx


log−lambda
100
50
0
−4
−2
0
2
4
theta5
typ
correct
lin approx

```r
pl.theta1 <- toplot.loglik.lin.comp(theta1.v, 1, 'theta1', ML.optim$par, ss2, M0 = 2.5,
                       Tlim, ML.optim$par,
                       by.s = 0.1)
```

```
## [1] 100
```

```r
pl.theta2 <- toplot.loglik.lin.comp(theta2.v, 2, 'theta2', ML.optim$par, ss2, M0 = 2.5,
                       Tlim, ML.optim$par,
                       by.s = 0.1)
```

```
## [1] 100
```

```r
pl.theta3 <- toplot.loglik.lin.comp(theta3.v, 3, 'theta3', ML.optim$par, ss2, M0 = 2.5,
                       Tlim, ML.optim$par,
                       by.s = 0.1)
```

```
## [1] 100
```

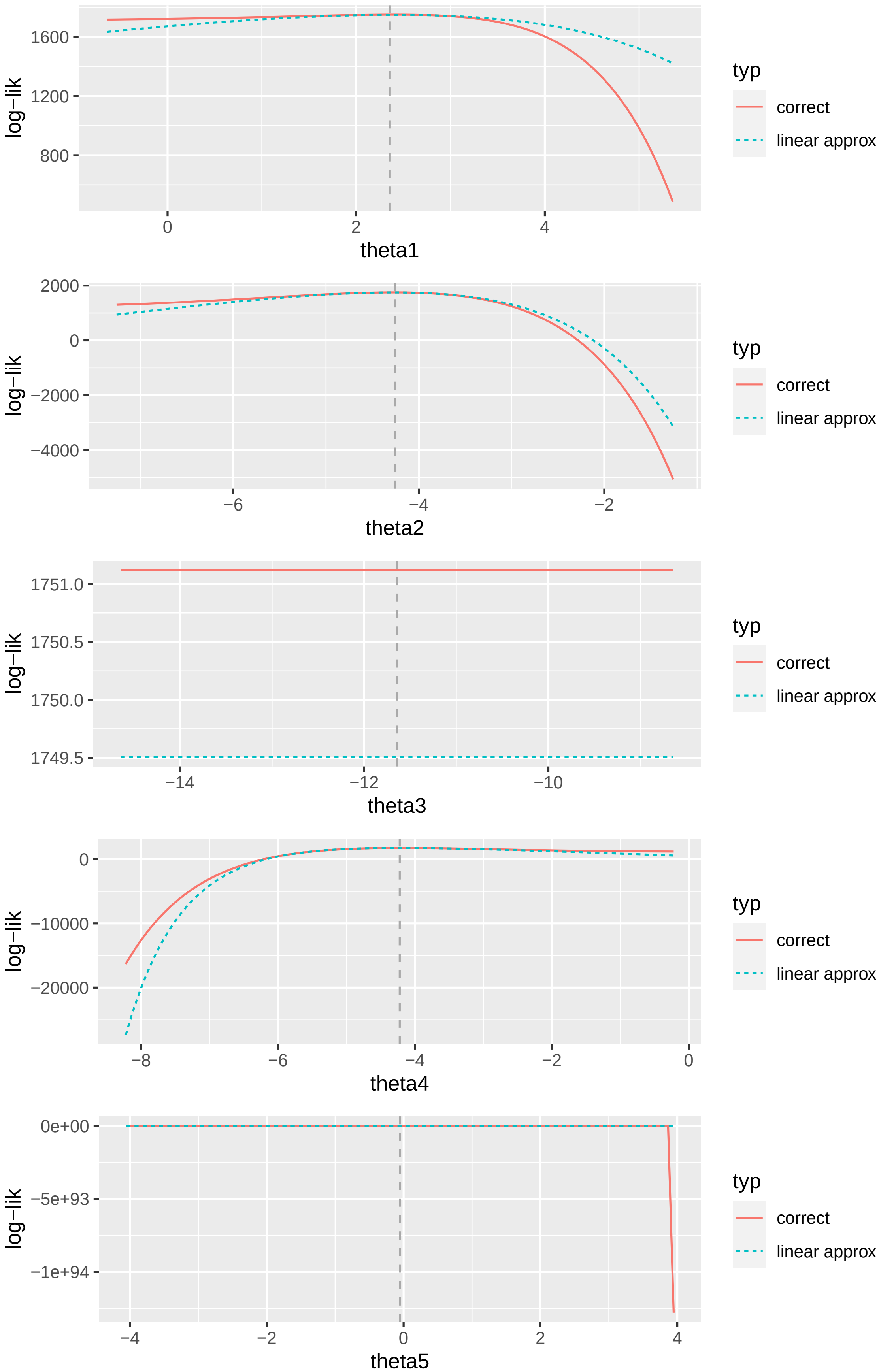
```r
pl.theta4 <- toplot.loglik.lin.comp(theta4.v, 4, 'theta4', ML.optim$par, ss2, M0 = 2.5,
                       Tlim, ML.optim$par,
                       by.s = 0.1)
```

```
## [1] 100
```

```r
pl.theta5 <- toplot.loglik.lin.comp(theta5.v, 5, 'theta5', ML.optim$par, ss2, M0 = 2.5,
                       Tlim, ML.optim$par,
                       by.s = 0.1)
```

```
## [1] 100
```

```r
multiplot(pl.theta1,
          pl.theta2,
          pl.theta3,
          pl.theta4,
          pl.theta5)
```


log−lik
1600
1200
800
0
2
4
theta1
typ
correct
linear approx
2000
0
−2000
−4000
−6
−4
−2
theta2
typ
correct
linear approx
1751.0
1750.5
1750.0
1749.5
−14
−12
−10
theta3
typ
correct
linear approx
0
−10000
−20000
−8
−6
−4
−2
0
theta4
typ
correct
linear approx
0e+00
−5e+93
−1e+94
−4
−2
0
2
4
theta5
typ
correct
linear approx