Tests: Sampling and likelihoods

We first set the stage.

```
library(ggplot2)
source('../R/HMM.R')
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'expm'
```

```
## The following object is masked from 'package:Matrix':
##
##     expm
```

```
set.seed(1723)
```

Step 1: sampling

We can't at the moment work with $L \approx 10^9, \rho \approx 10^{-8}, \theta \approx 10^{-6}$ since such a large set of x, y values won't fit neatly into memory. We therefore instead decrease L by a factor of 10^4 and increase ρ and θ by a corresponding factor; guaranteeing that the expected total number of events should remain the same.

```
rho <- 1e-4
theta <- 1e-2
M <- 10
L <- 1e5
sim <- sample_XandY(rho,theta,M,L)
X <- sim$X
Y <- sim$Y
rm(sim) #halve memmory use
```

Step 2: rudimentary analysis:

2.1 Basic exploration

```
print(count_changepoints(X))
```

```
## [1] 2
```

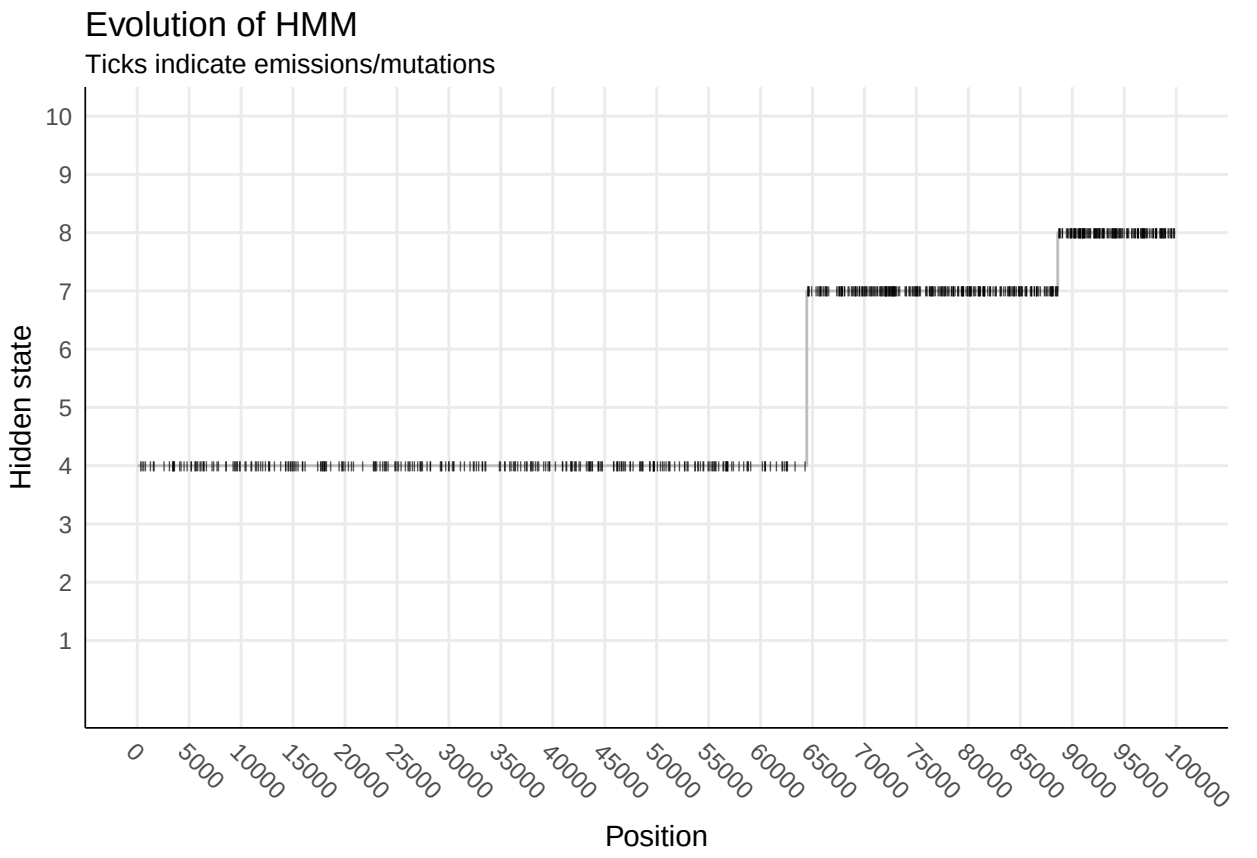
```
print(sum(Y))
```

```
## [1] 689
```

Generating a basic plot:

```
plt <- ggplot(data = data.frame(Index = 1:length(X), X = X, Y = Y)) +
  geom_line(mapping = aes(x = Index, y = X), linetype = 1, color = 'gray') +
  geom_point(mapping = aes(x = Index, y = Y*X), size = 1, alpha = Y, shape = '|') +
  scale_y_continuous(breaks = round(seq(1,M,length.out = min(M,M))), limits = c(0,M))+
  scale_x_continuous(breaks = round(seq(0,L,length.out = min(21,L+1))), limits = c(0,L))+
  theme_minimal(base_size = 11) +
  theme(panel.grid.minor = element_blank(),
```

```
        axis.text.x = element_text(angle = -45,vjust = 1, hjust = 0),
        axis.line = element_line(size = 0.3, linetype = "solid")) +
  ggtitle('Evolution of HMM', subtitle = 'Ticks indicate emissions/mutations') +
  xlab('Position') +
  ylab('Hidden state')
print(plt)
```



2.2 Maximum likelihood estimation (of ρ)

We examine the recombination rate across two windows: one with a recombination point in it, and one without.

```
win1 <- 1:1e4
win2 <- 6e4:7e4

#see what is going on in each window
print(c(count_changepoints(X[win1]), sum(Y[win1])))
```

```
## [1]  0 43
```

```
print(c(count_changepoints(X[win2]), sum(Y[win2])))
```

```
## [1]  1 62
```

```
theta_global_estimate <- sum(Y == 1)/L #theta-value to be used in ML-estimation
ML_rho_1 <- compute_ML_rho(Y[win1],M,theta = theta_global_estimate)
ML_rho_2 <- compute_ML_rho(Y[win2],M,theta = theta_global_estimate)
```

We plot the likelihood functions in either window

```
print(ML_rho_1)
```

```
## [1] 2.143783e-14
```

```
print(ML_rho_2)
```

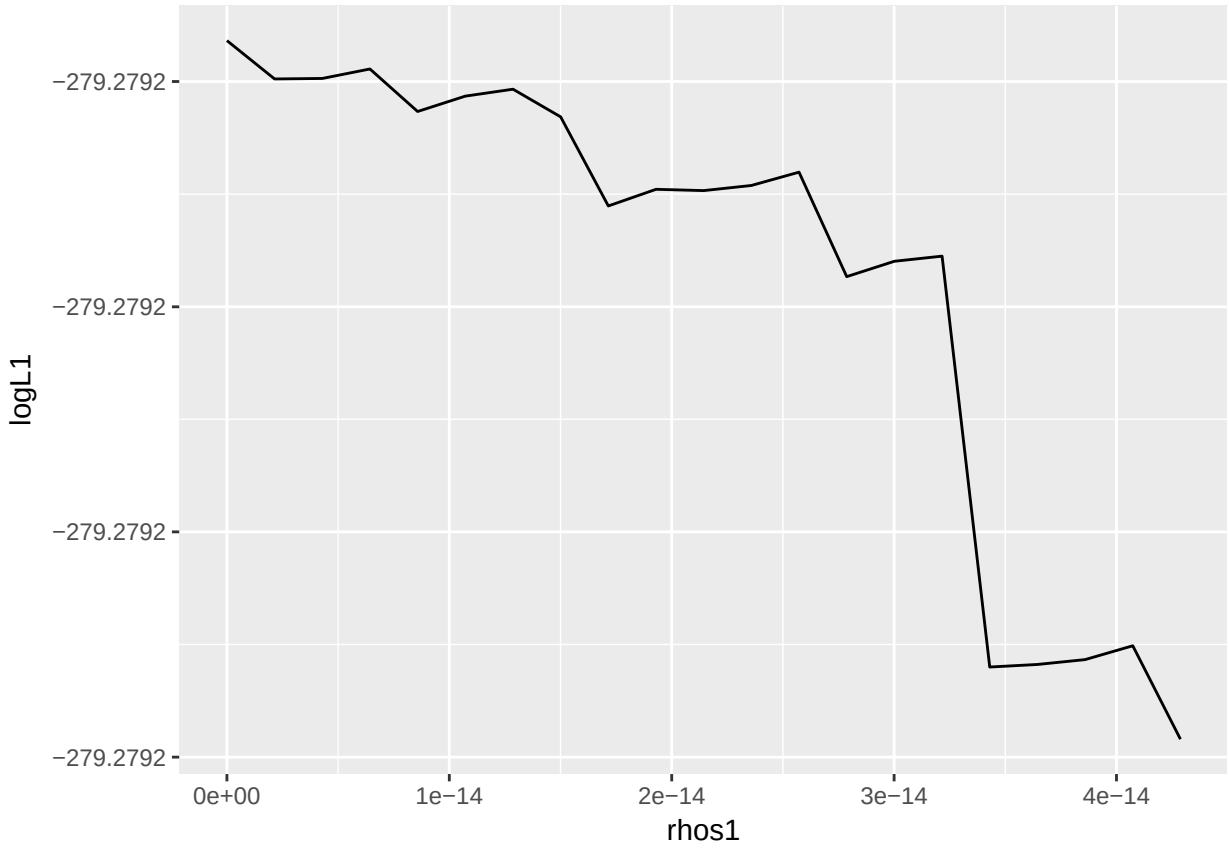
```
## [1] 0.04015143
```

```
rhos1 <- seq(0, 2*ML_rho_1,length.out = 21)
rhos2 <- seq(0, 2*ML_rho_2,length.out = 21)

f1 <- function(x) log_likelihood_Y(Y[win1],M,rho = x, theta = theta_global_estimate)
f2 <- function(x) log_likelihood_Y(Y[win2],M,rho = x, theta = theta_global_estimate)

logL1 <- sapply(X = rhos1, FUN = f1)
logL2 <- sapply(X = rhos2, FUN = f2)
```

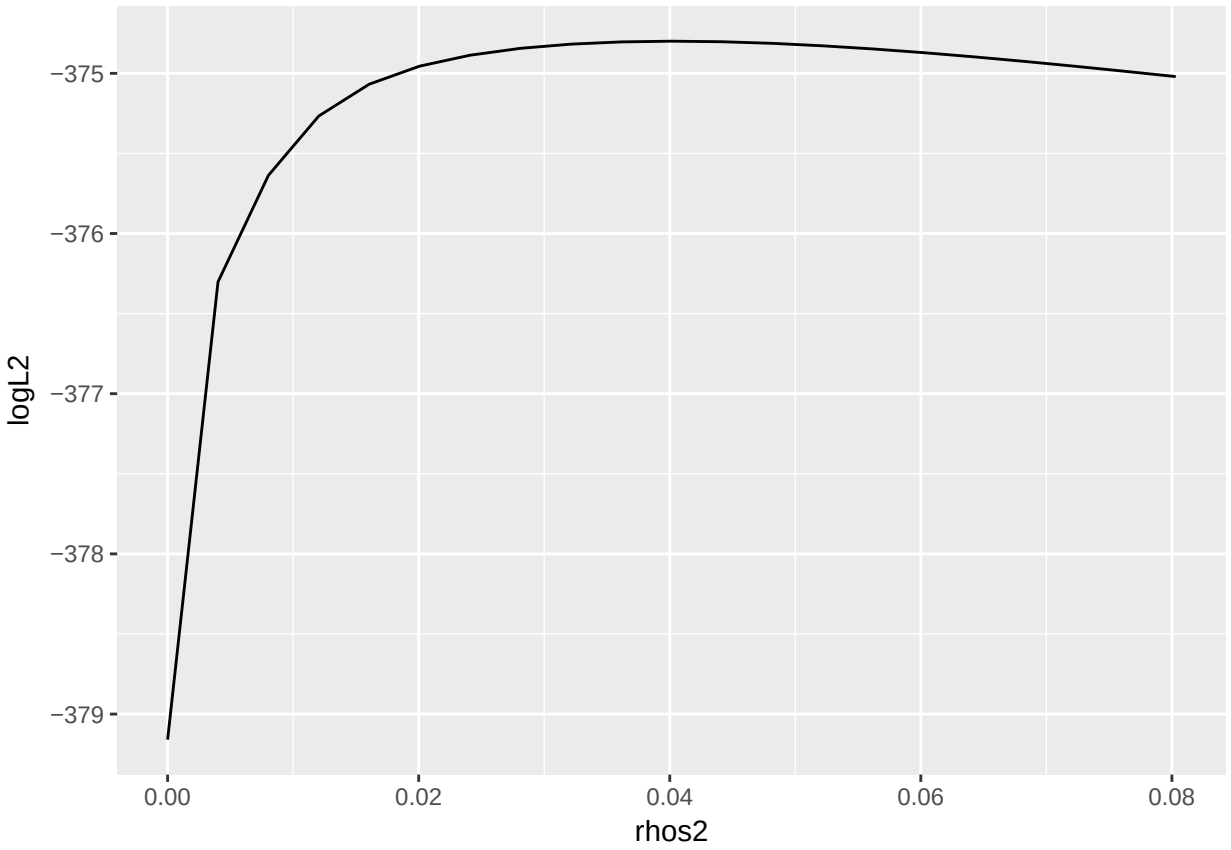
```
pl_lik1 <- qplot(x = rhos1, y = logL1, geom = 'line')
print(pl_lik1)
```



Here, we seem to be dealing with an optimum close to 0. Note on the y-axis that the likelihood is rather flat here. The difference between the highest and lowest values of the log-likelihood evaluated in this plot is $\approx 1.6 \cdot 10^{-10}$.

for thew interval containing a change-point, things look much better.

```
pl_lik2 <- qplot(x = rhos2, y = logL2, geom = 'line')
print(pl_lik2)
```



Note: the important thing here, is that we are indeed able to get quite different estimates of the maximum likelihood in both windows: the estimated values $\hat{\rho}_{\text{window 1}} = 2.6 \cdot 10^{-10}$ and $\hat{\rho}_{\text{window 2}} = 4.0 \cdot 10^{-2}$ *differ by 12 orders of magnitude*