SimulationStudySummary

Xiang Ji

9/7/2017

This R markdown file summarizes Simulation Study results.

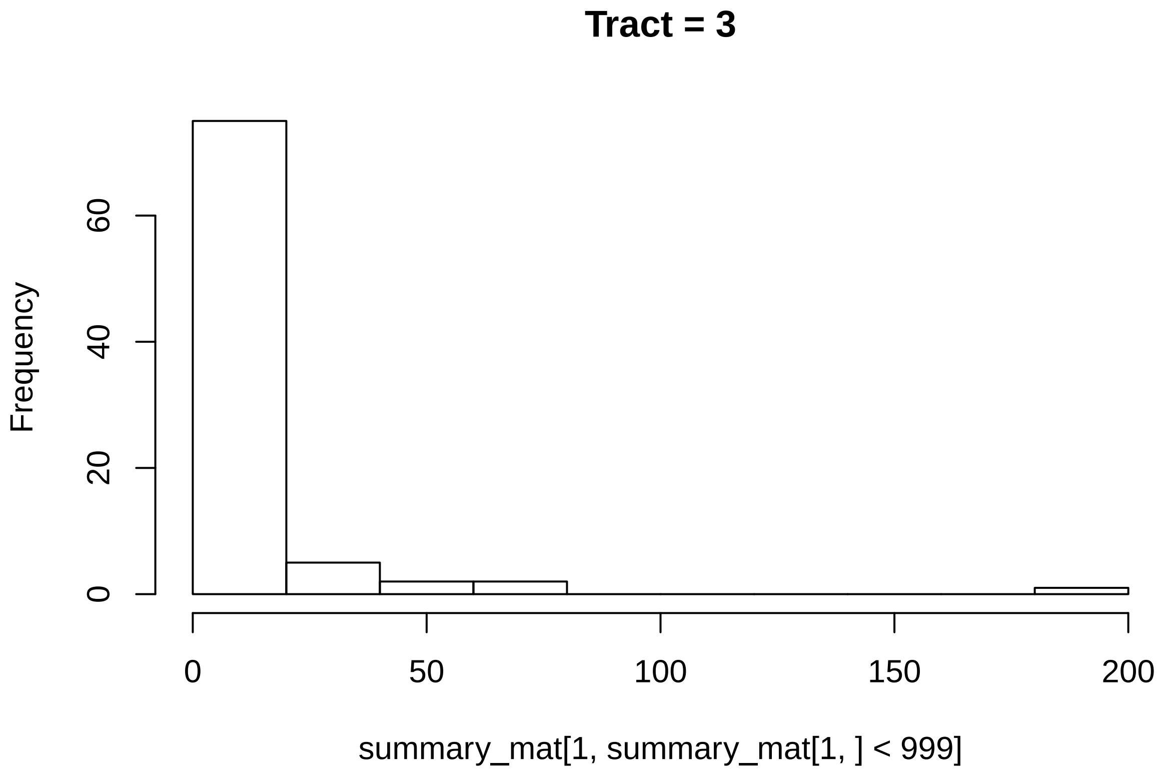
```r
rm(list=ls())  # clean up workspace
setwd("/Users/xji3/GitFolders/YeastIGCTract/SimulationStudy/")
filtered.pairs <- readLines('../Filtered_pairs.txt')

Tract.list <- c(3.0, 10.0, 50.0, 100.0, 200.0, 300.0, 400.0, 500.0)
# First read in HMM results
# from summary file
for(tract in Tract.list){
  hmm.tract.summary <- NULL
  for(sim in 1:100){
    hmm.summary <- paste("./summary/Tract_", toString(tract), '.0/sim_',
                         toString(sim), '/HMM_YDR418W_YEL054C_MG94_nonclock_sim_',
                         toString(sim), '_1D_summary.txt', sep = "")
    if (file.exists(hmm.summary)){
      all <- readLines(hmm.summary, n = -1)
      col.names <- paste("sim_", toString(sim), sep = "")
      row.names <- strsplit(all[length(all)], ' ')[[1]][-1]
      summary_mat <- as.matrix(read.table(hmm.summary,
                                          row.names = row.names,
                                          col.names = col.names))
      hmm.tract.summary <- cbind(hmm.tract.summary, summary_mat)
    }

  }
  assign(paste("HMM_Tract_", toString(tract), "_summary", sep = ""), hmm.tract.summary)
}

# from plots
for(tract in Tract.list){
  hmm.tract.plots <- NULL
  for(sim in 1:100){
    hmm.plot <- paste("./plot/Tract_", toString(tract), '.0/sim_',
                         toString(sim), '/HMM_YDR418W_YEL054C_lnL_sim_',
                         toString(sim), '_1D_surface.txt', sep = "")
    if (file.exists(hmm.plot)){
      lnL.surface <- read.table(hmm.plot)
      max.idx <- which.max(lnL.surface[, 2])
      new.summary <- matrix(c(3.0*exp(-lnL.surface[max.idx, 1]), lnL.surface[max.idx, 2]), 2, 1)
      rownames(new.summary) <- c("tract in nt", "lnL")
      colnames(new.summary) <- paste("sim_", toString(sim), sep = "")
      hmm.tract.plots <- cbind(hmm.tract.plots, new.summary)
    }
  }
  assign(paste("HMM_Tract_", toString(tract), "_plot", sep = ""), hmm.tract.plots)
}
```

```r
# Now read in PSJS summary results
for(tract in Tract.list){
  PSJS.tract.summary <- NULL
  for(sim in 1:100){
    PSJS.summary <- paste("./summary/Tract_", toString(tract), '.0/sim_',
                          toString(sim), '/PSJS_HKY_rv_sim_',
                          toString(sim), "_Tract_", toString(tract), '.0_summary.txt', sep = "")
    if (file.exists(PSJS.summary)){
      all <- readLines(PSJS.summary, n = -1)
      col.names <- paste("sim_", toString(sim), sep = "")
      row.names <- strsplit(all[length(all)], ' ')[[1]][-1]
      summary_mat <- as.matrix(read.table(PSJS.summary,
                                          row.names = row.names,
                                          col.names = col.names))
      PSJS.tract.summary <- cbind(PSJS.tract.summary, summary_mat)
    }

  }
  assign(paste("PSJS_Tract_", toString(tract), "_summary", sep = ""), PSJS.tract.summary)
}
```

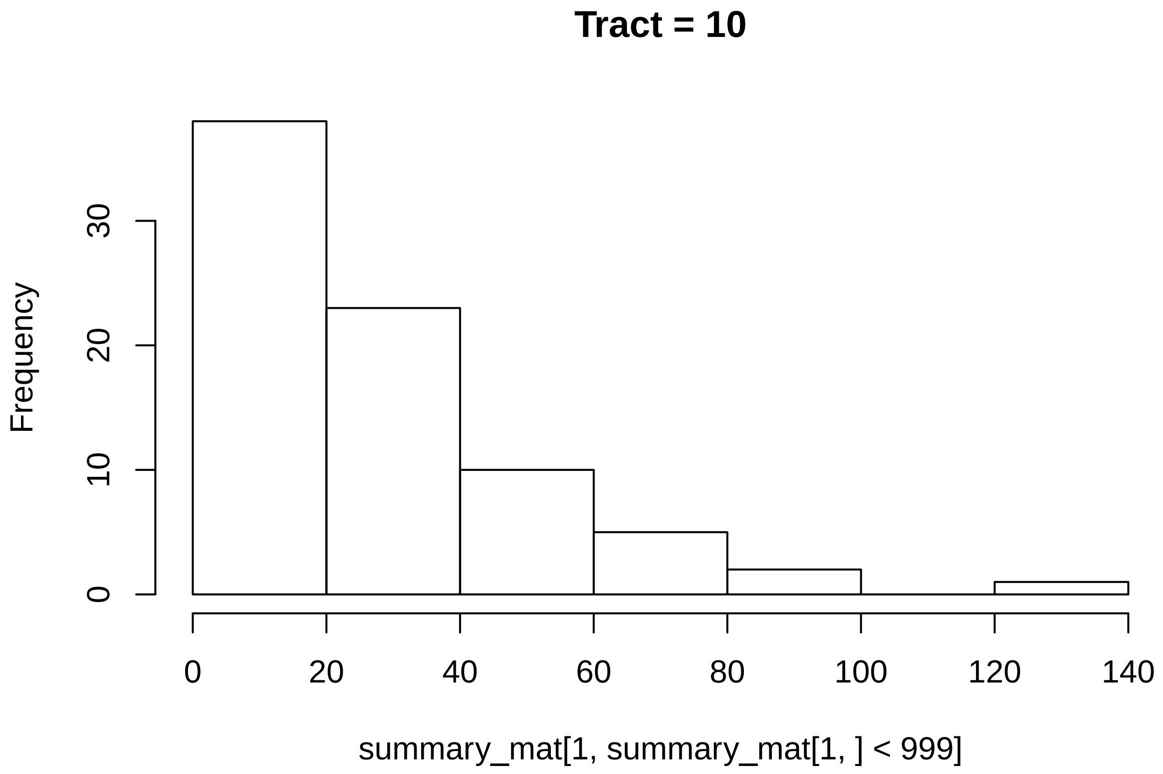
OK, Now show the performance summary

```r
# HMM results
for (tract in Tract.list){
  # show how many stuck at boundary 1000 nt first
  print(paste("Tract = ", toString(tract), sep = ""))
  summary_mat <- get(paste("HMM_Tract_", toString(tract), "_plot", sep = ""))

  # histogram of inferred tract length
  hist(summary_mat[1, summary_mat[1, ] < 999.], main = paste("Tract = ", toString(tract), sep = ""))
  print(paste("Among total 100 simulated data sets, ", toString(sum(summary_mat[1, ] > 999)),
              " datasets stuck at 1000", sep = ""))
  print(c("mean", mean(summary_mat[1, summary_mat[1, ] < 999.]),
          "sd", sd(summary_mat[1, summary_mat[1, ] < 999.])))
}
```

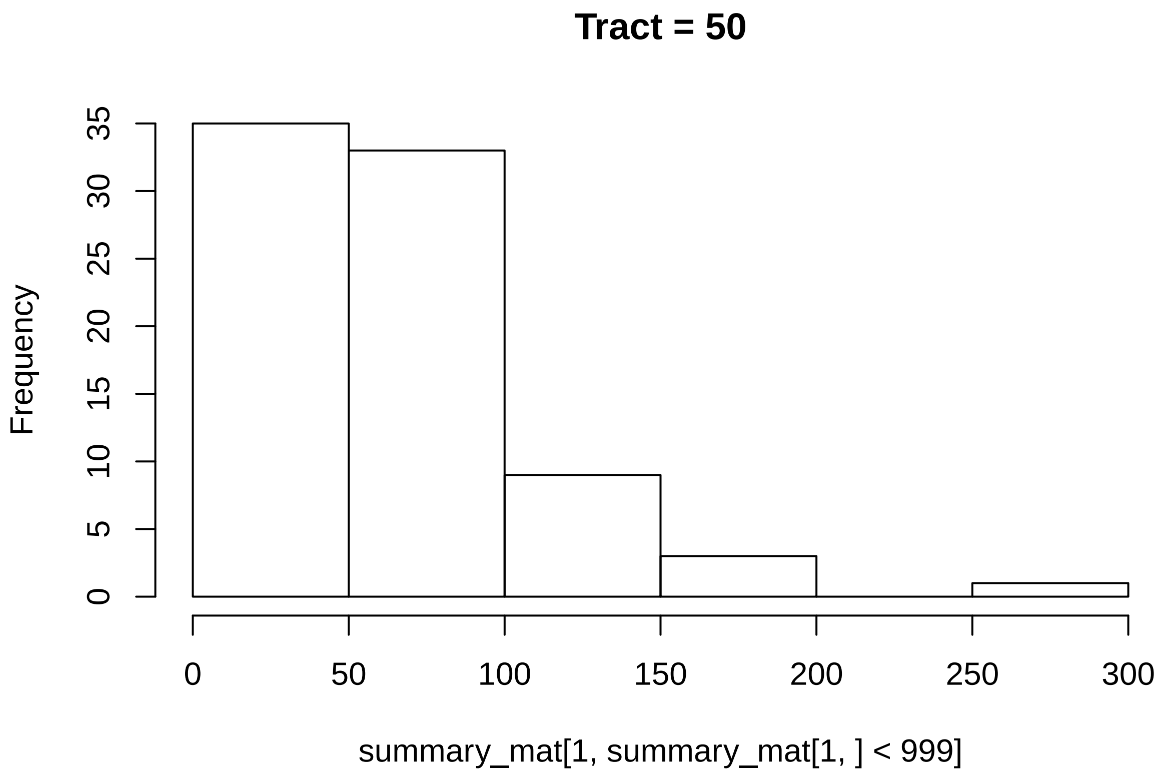
```
## [1] "Tract = 3"
```



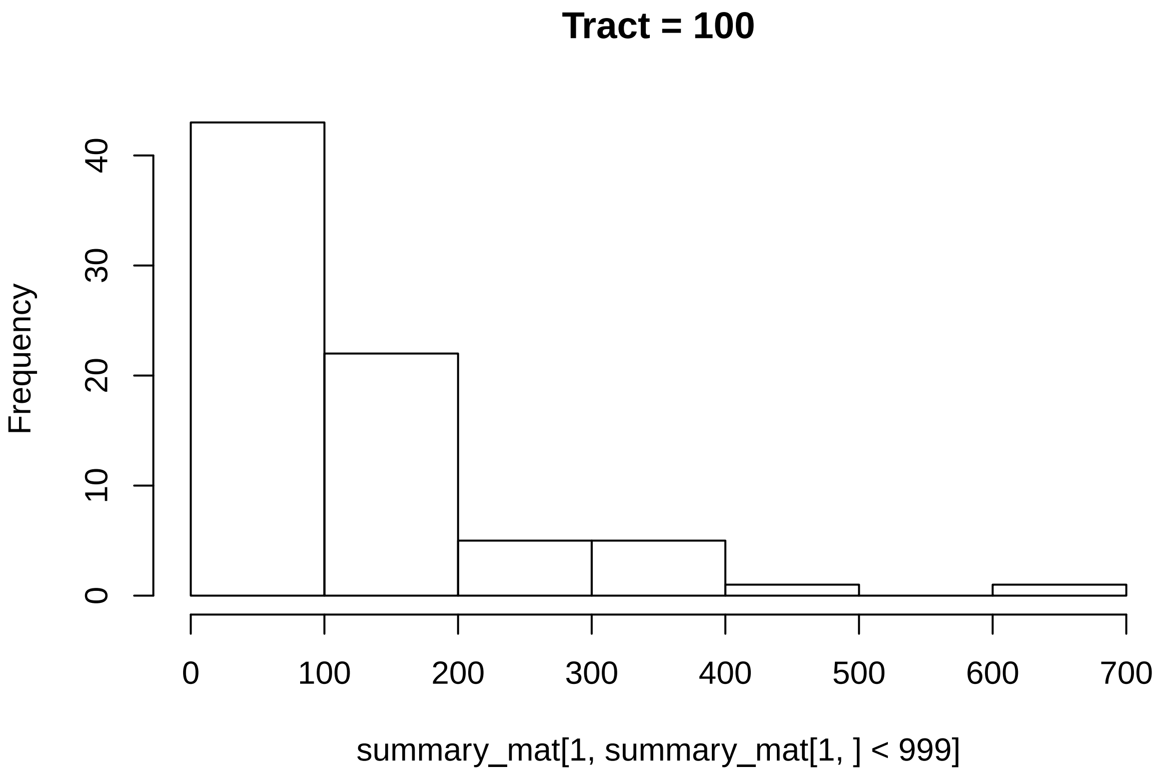
```
## [1] "Among total 100 simulated data sets, 15 datasets stuck at 1000"
## [1] "mean"              "11.6941176470561" "sd"
## [4] "23.409920327626"
## [1] "Tract = 10"
```



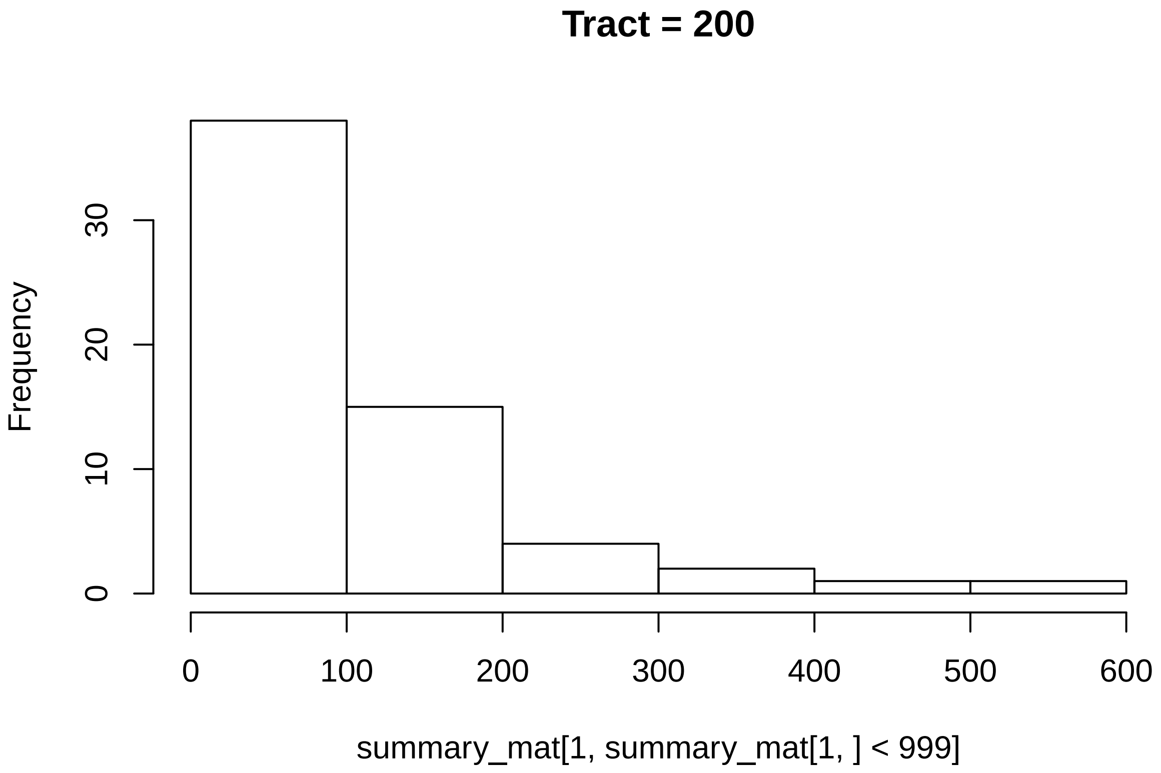
```
## [1] "Among total 100 simulated data sets, 21 datasets stuck at 1000"
## [1] "mean"             "28.5316455696352" "sd"
## [4] "24.4245753935622"
## [1] "Tract = 50"
```



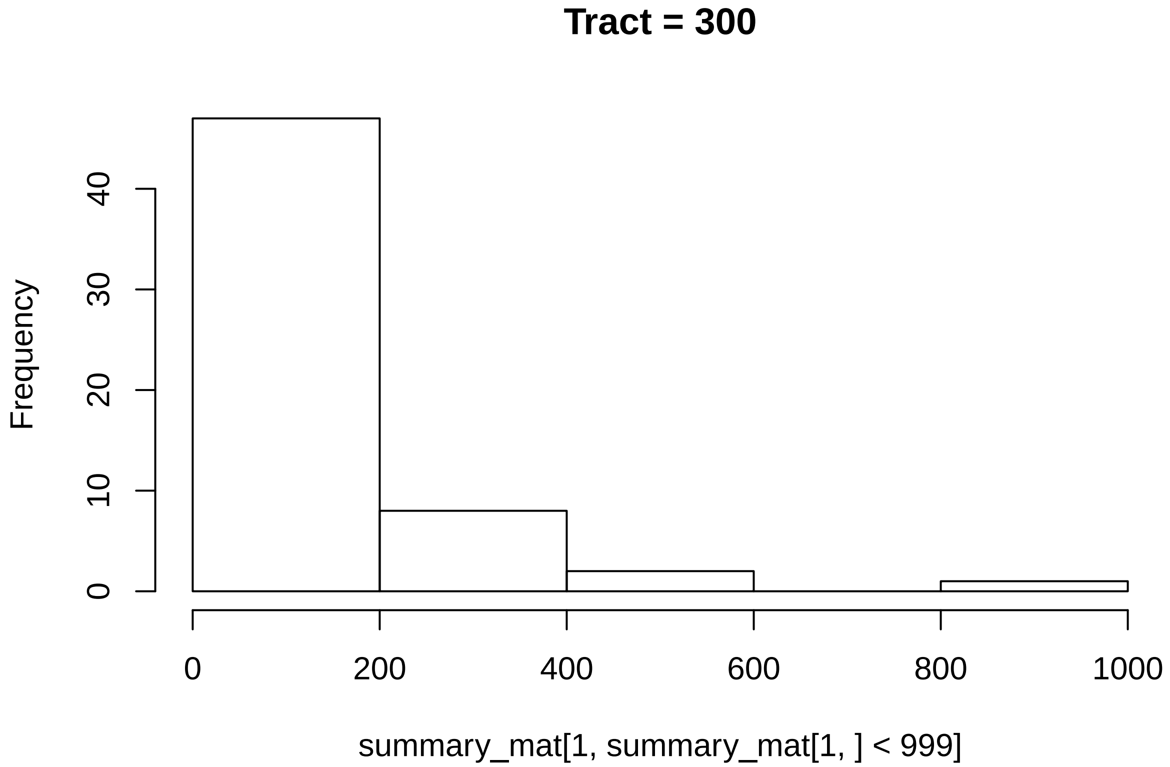
```
## [1] "Among total 100 simulated data sets, 19 datasets stuck at 1000"
## [1] "mean"             "67.0864197531772" "sd"
## [4] "50.0167965615616"
## [1] "Tract = 100"
```



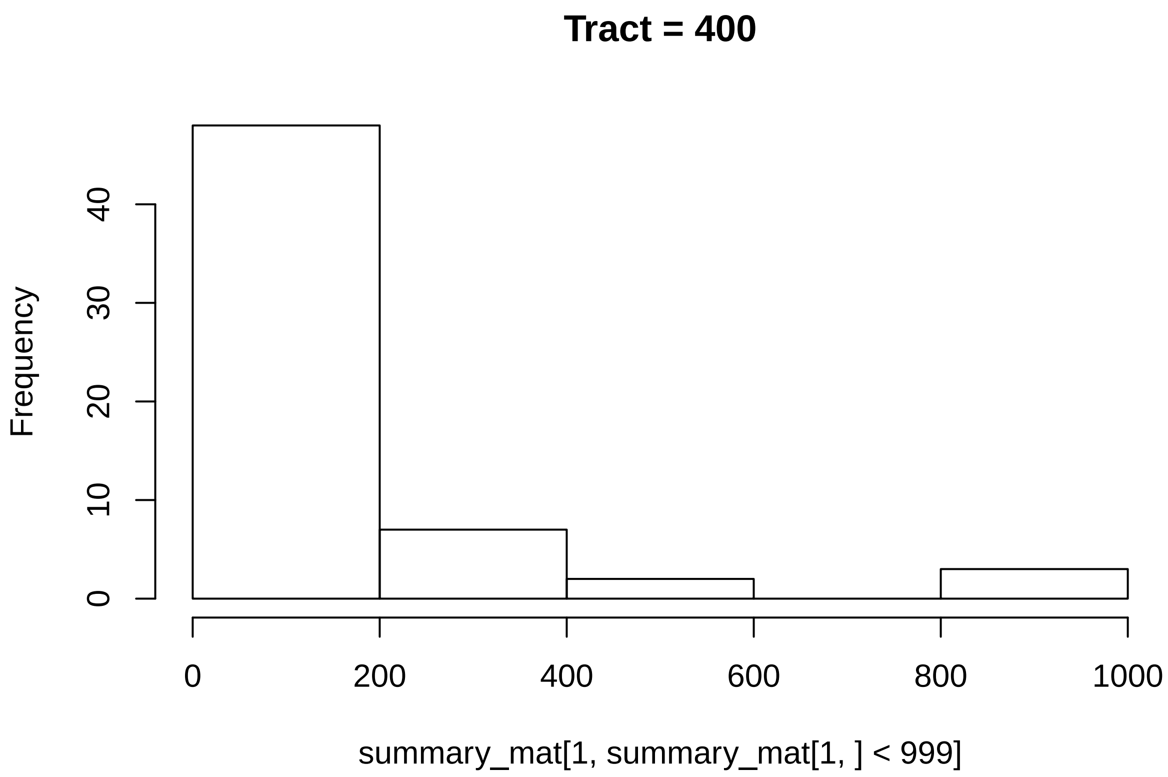
```
## [1] "Among total 100 simulated data sets, 23 datasets stuck at 1000"
## [1] "mean"             "121.597402597374" "sd"
## [4] "111.189768998766"
## [1] "Tract = 200"
```



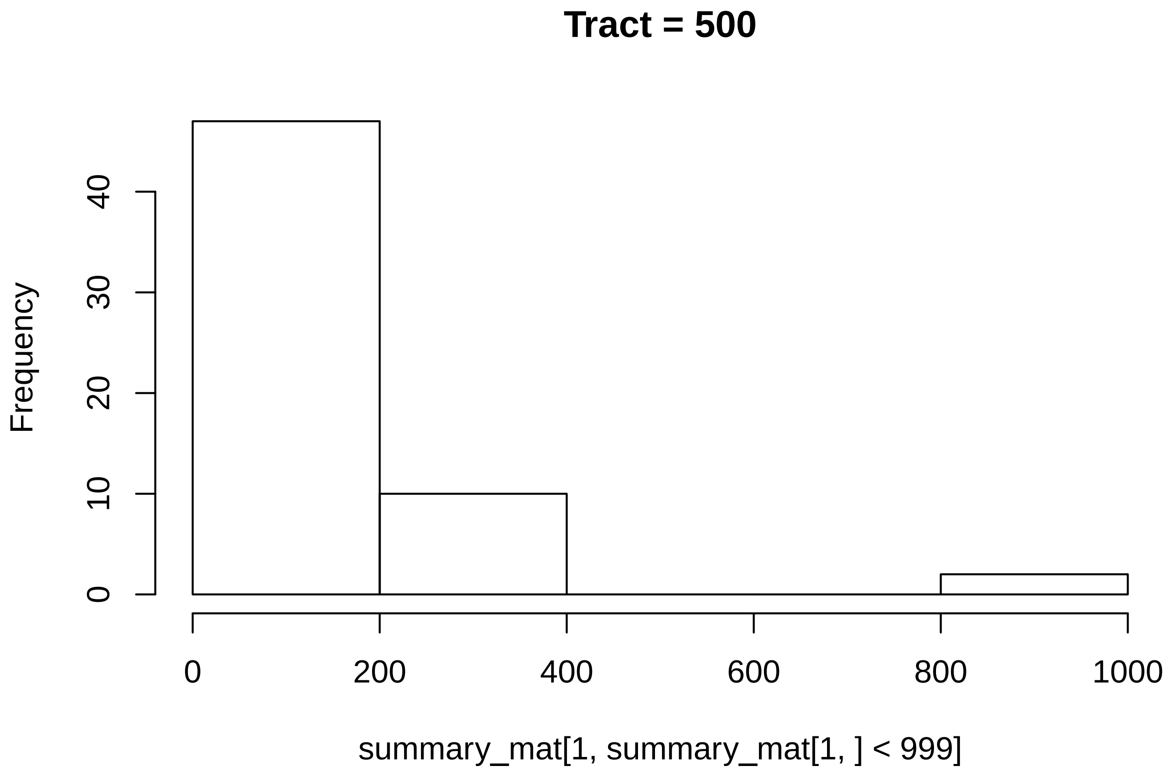
```
## [1] "Among total 100 simulated data sets, 38 datasets stuck at 1000"
## [1] "mean"             "111.590163934384" "sd"
## [4] "109.950955285812"
## [1] "Tract = 300"
```



```
## [1] "Among total 100 simulated data sets, 42 datasets stuck at 1000"
## [1] "mean"             "124.293103448246" "sd"
## [4] "157.010001761835"
## [1] "Tract = 400"
```

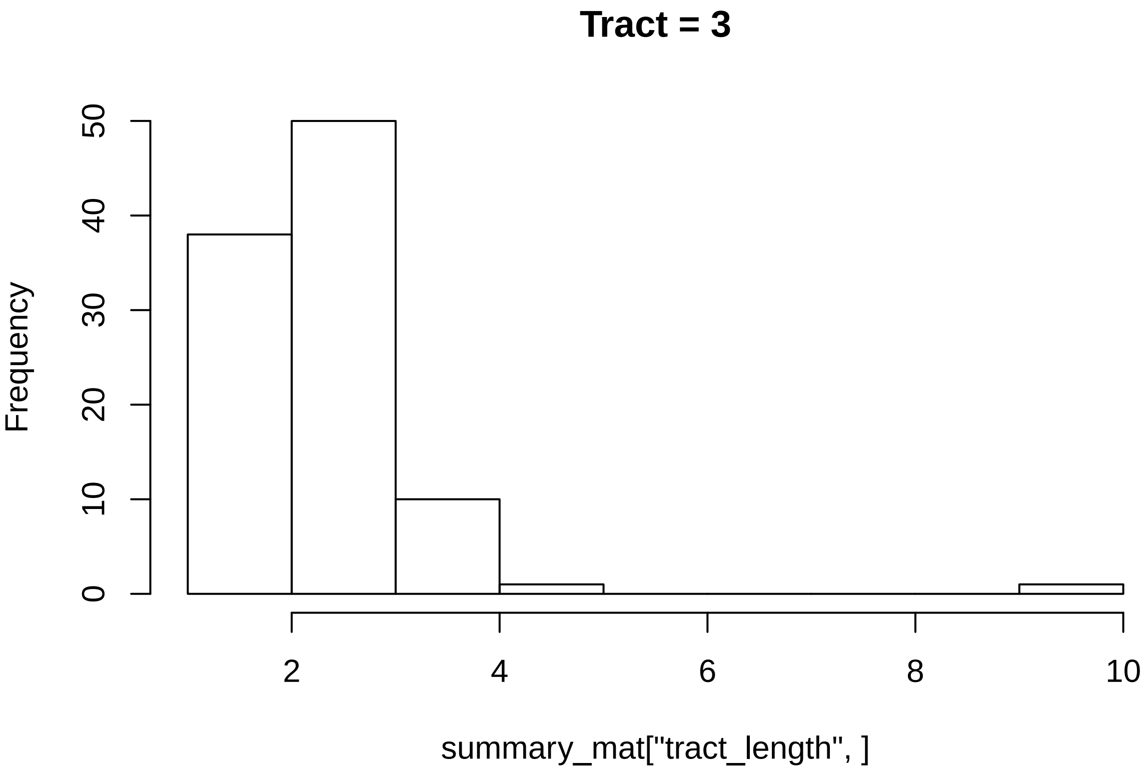


```
## [1] "Among total 100 simulated data sets, 40 datasets stuck at 1000"
## [1] "mean"             "153.333333333315" "sd"
## [4] "219.73201449035"
## [1] "Tract = 500"
```

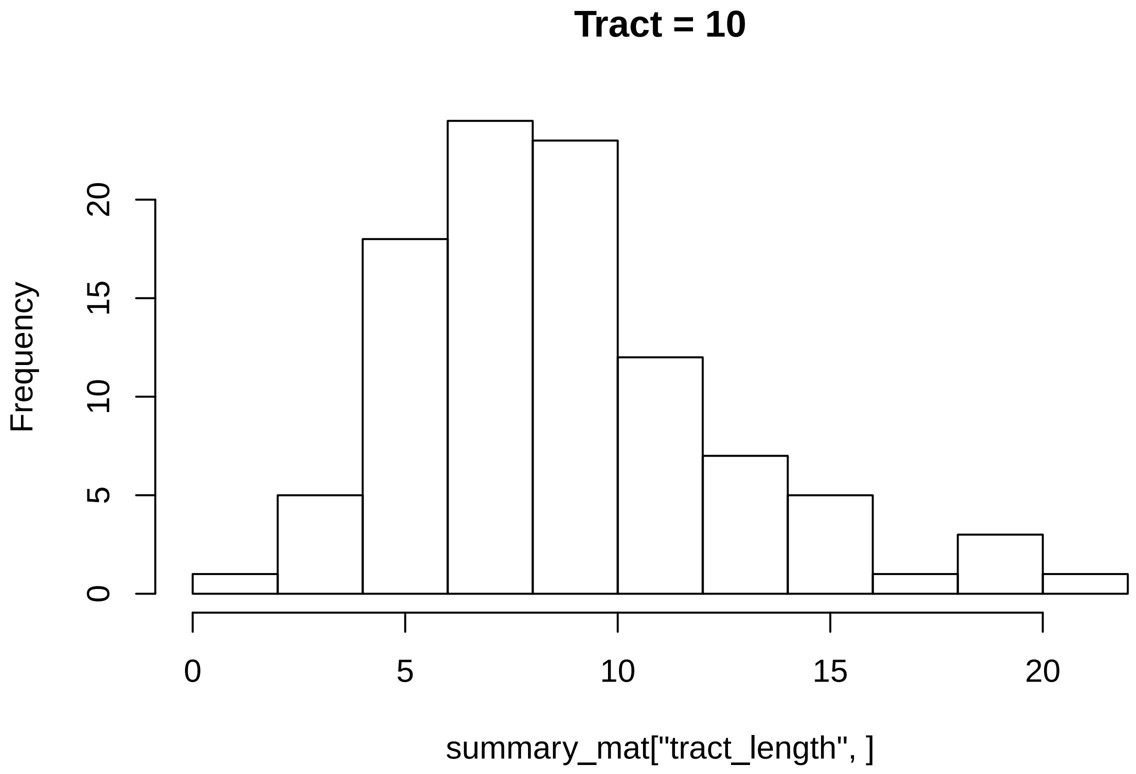


```
## [1] "Among total 100 simulated data sets, 41 datasets stuck at 1000"
## [1] "mean"             "136.30508474575"  "sd"
## [4] "187.058284687408"
```

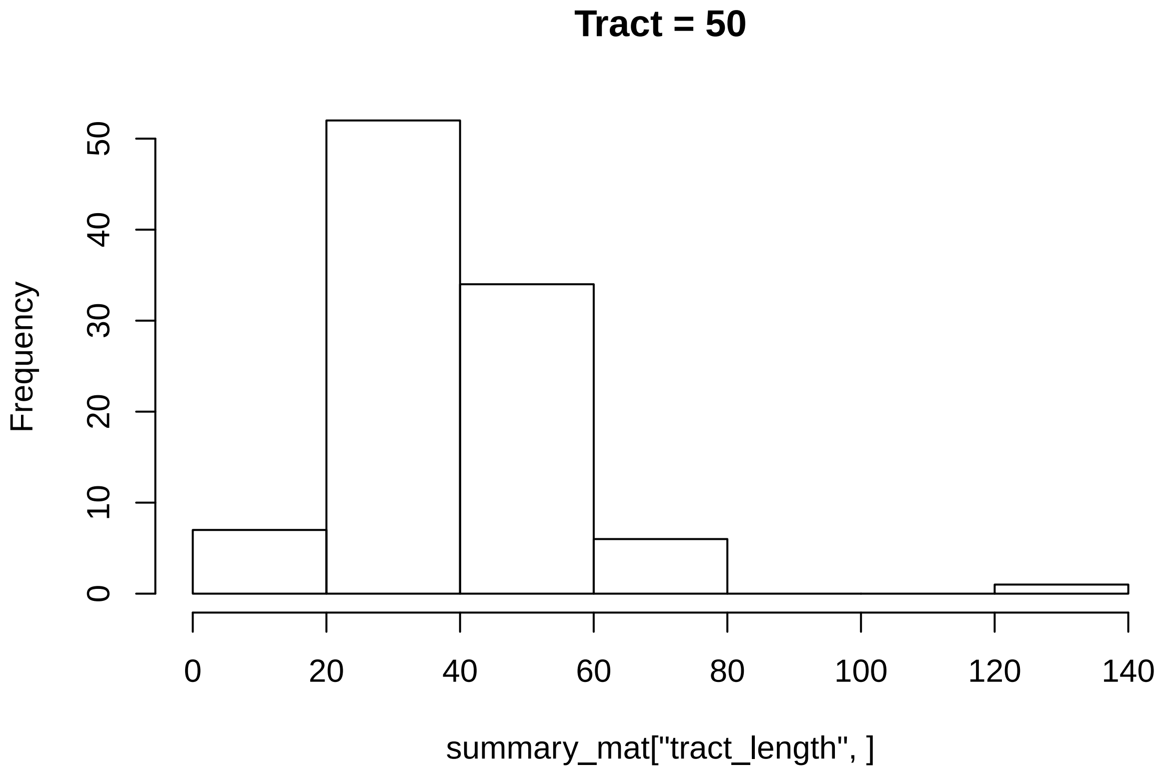
```
# PSJS results
for(tract in Tract.list){
  summary_mat <- get(paste("PSJS_Tract_", toString(tract), "_summary", sep = ""))
  # histogram of inferred tract length
  hist(summary_mat["tract_length", ], main = paste("Tract = ", toString(tract), sep = ""))
  print(c("mean", mean(summary_mat["tract_length", ]),
          "sd", sd(summary_mat["tract_length", ])))
}
```



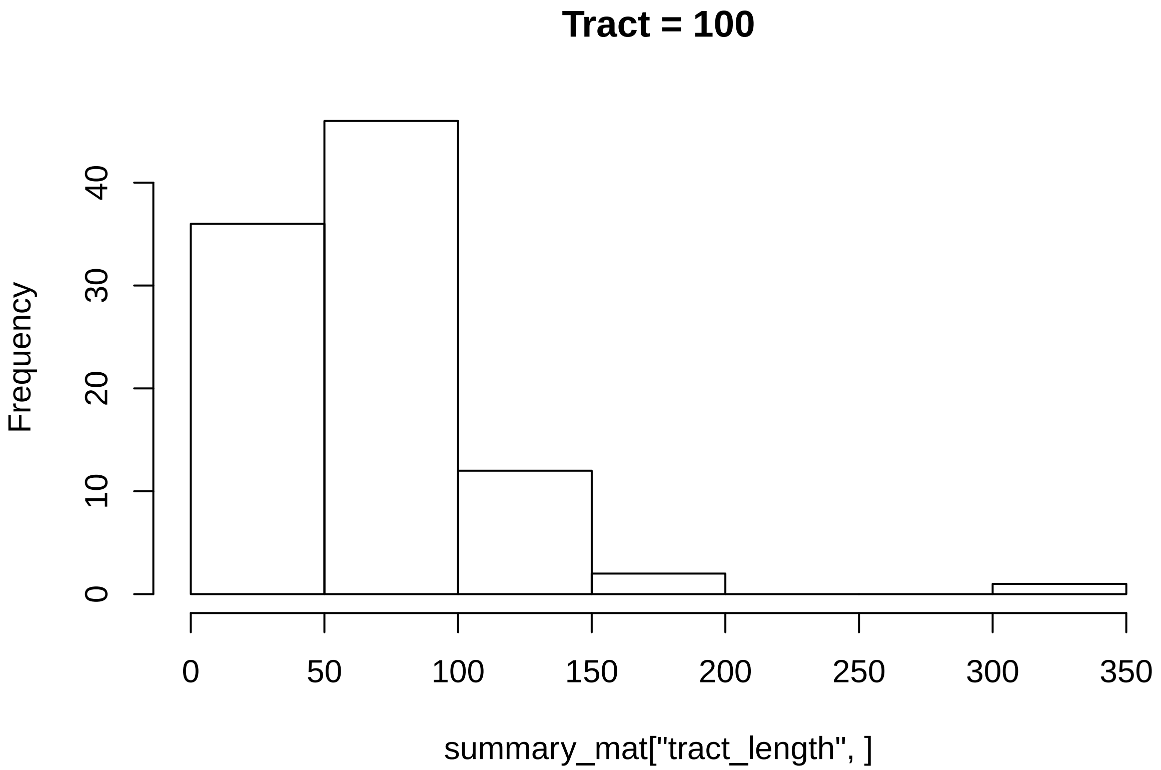
```
## [1] "mean"             "2.29718013607812" "sd"
## [4] "1.02874535788003"
```



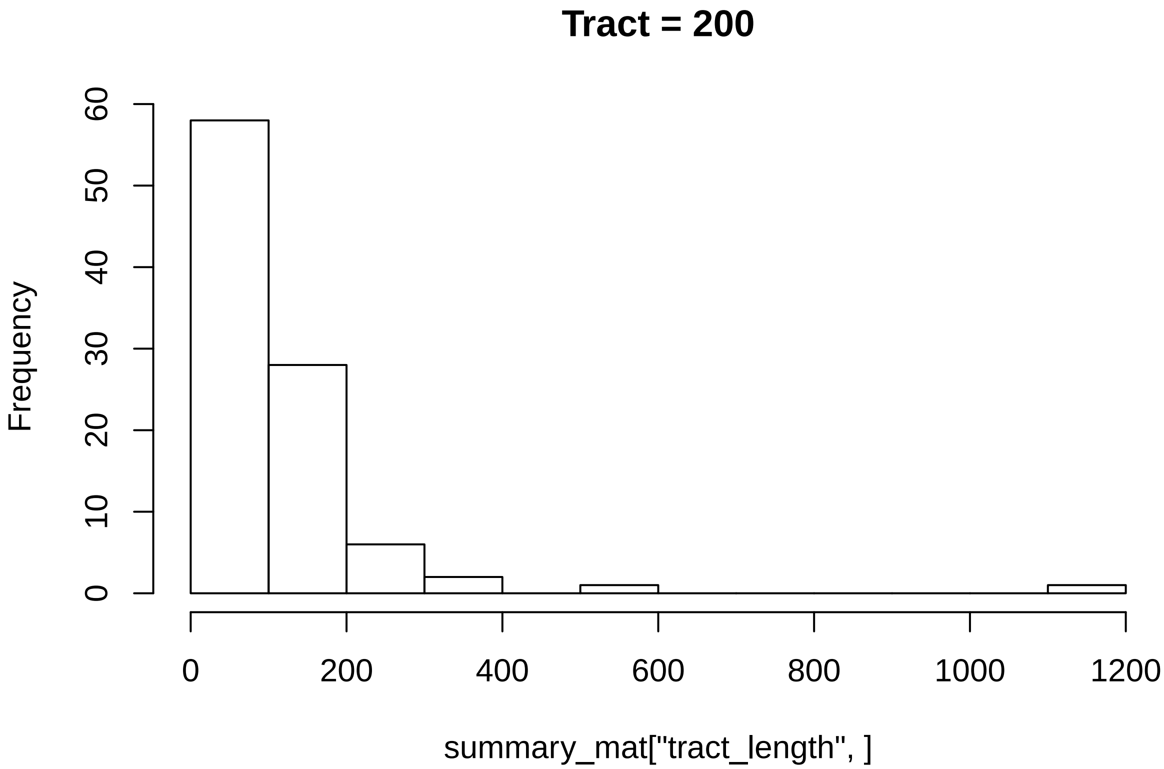
```
## [1] "mean"             "8.72620746076138" "sd"
## [4] "3.72714198660433"
```



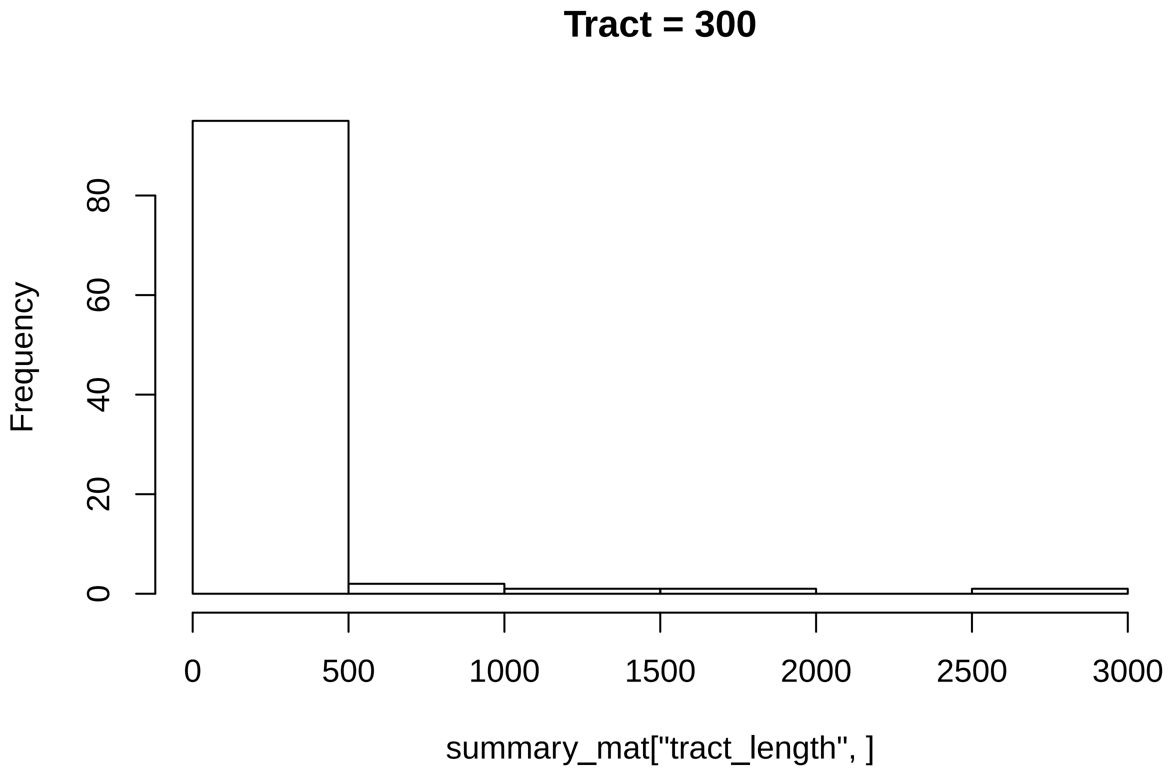
```
## [1] "mean"             "38.1001820929933" "sd"
## [4] "16.207953647501"
```



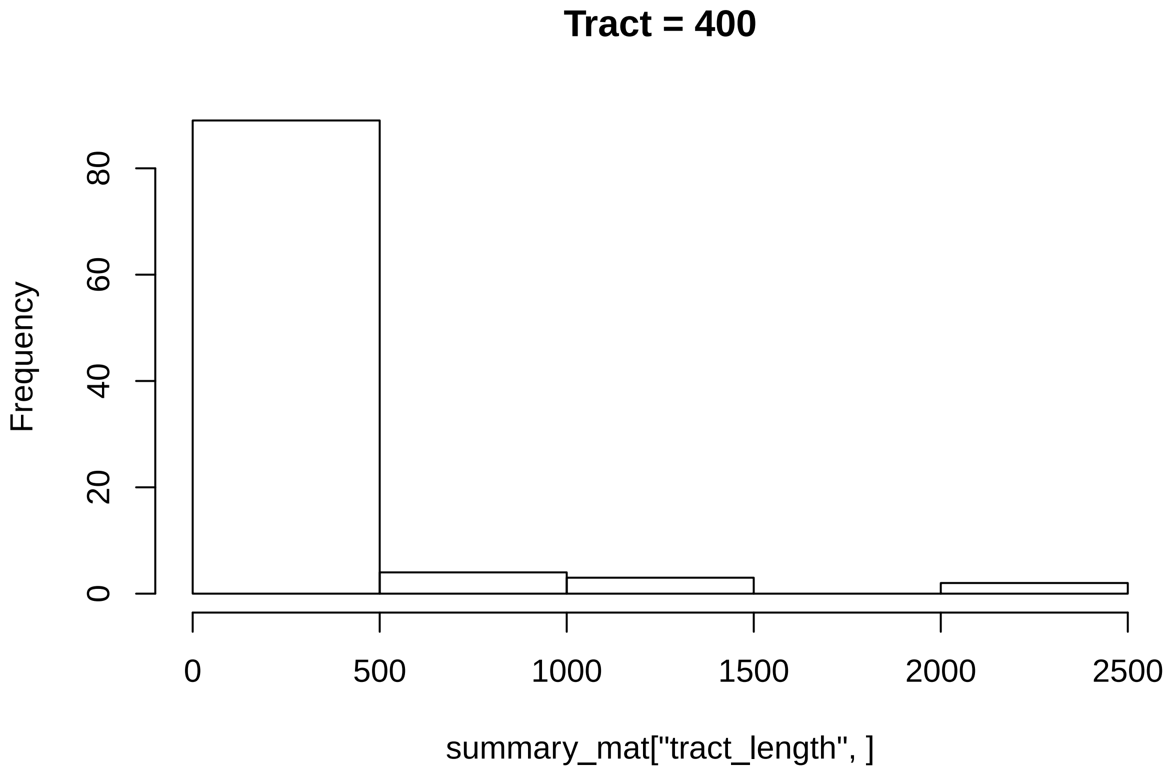
```
## [1] "mean"             "66.9946146602955" "sd"
## [4] "40.527169109434"
```



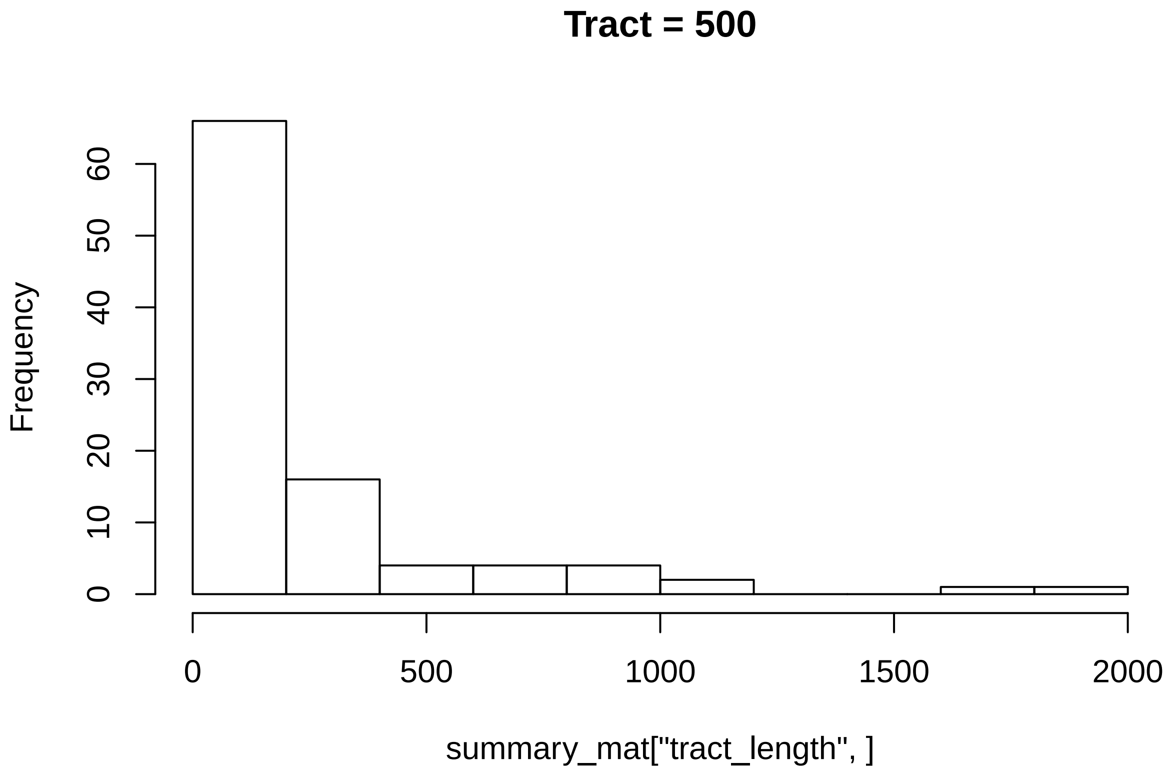
```
## [1] "mean"             "115.573878487051" "sd"
## [4] "132.756671098129"
```



```
## [1] "mean"             "189.995520518941" "sd"
## [4] "356.357994056025"
```

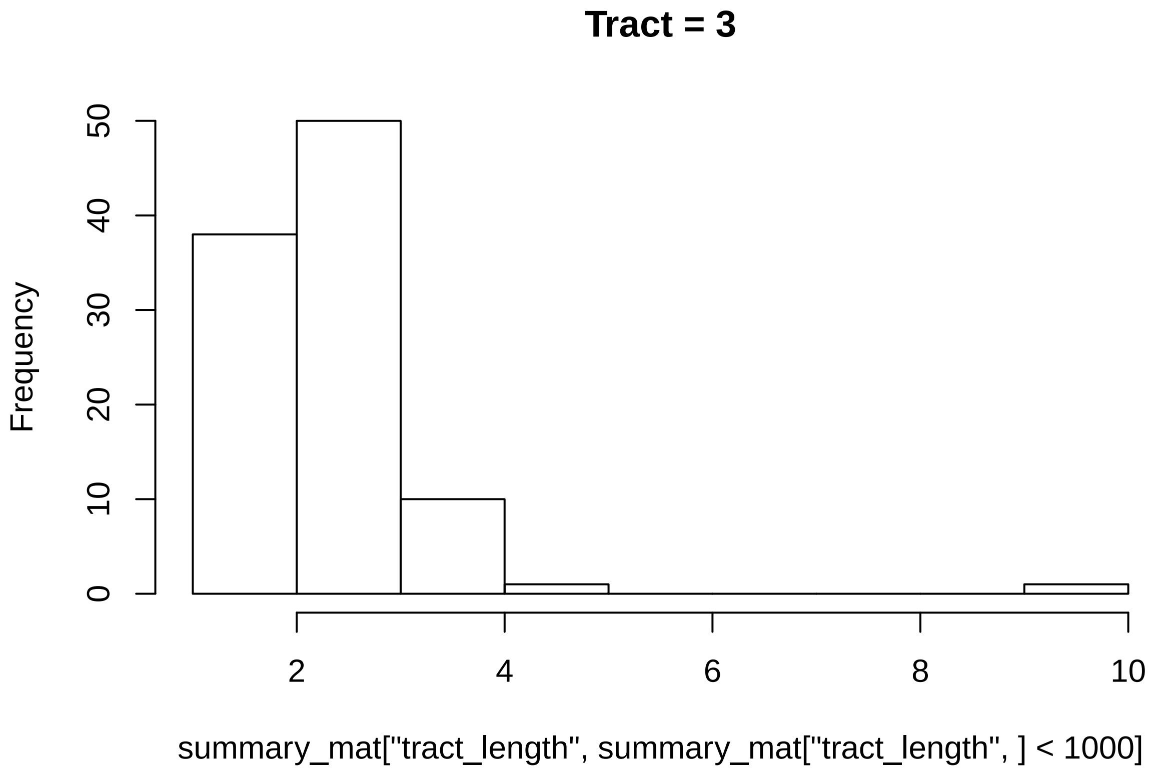


```
## [1] "mean"             "217.14487788415"  "sd"
## [4] "377.699962735285"
```

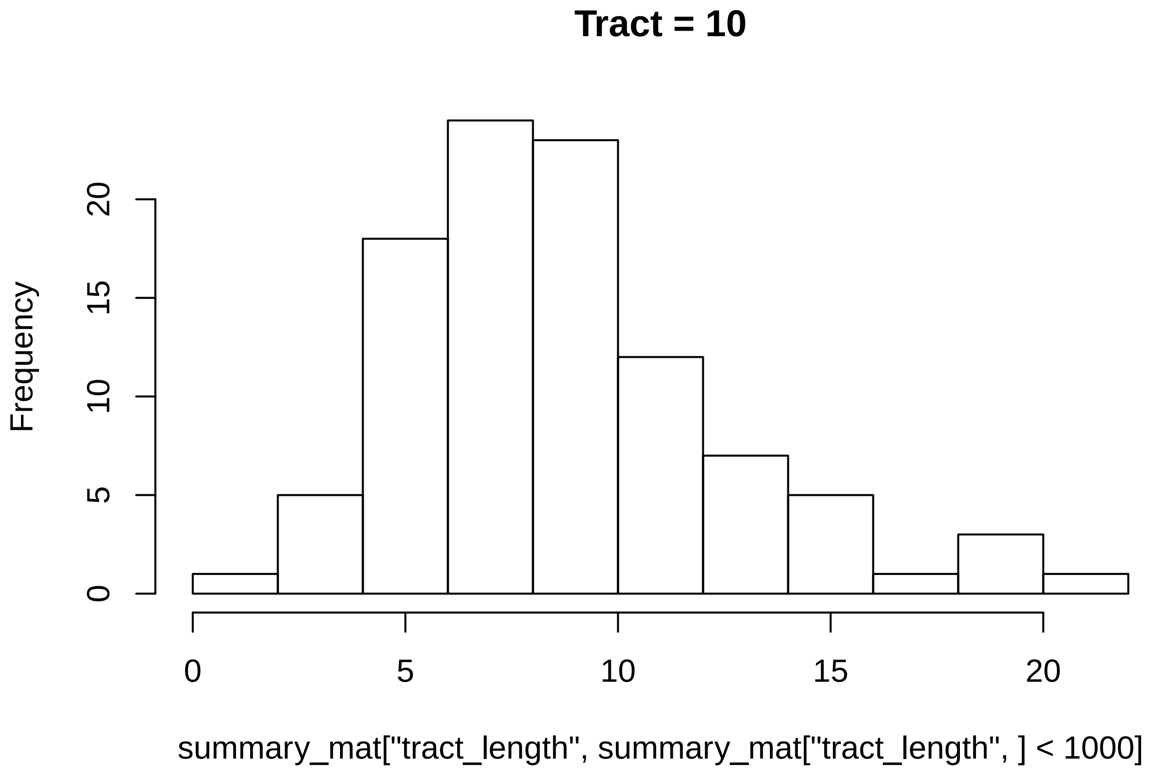


```
## [1] "mean"             "240.064448783542" "sd"
## [4] "348.396527453917"
```

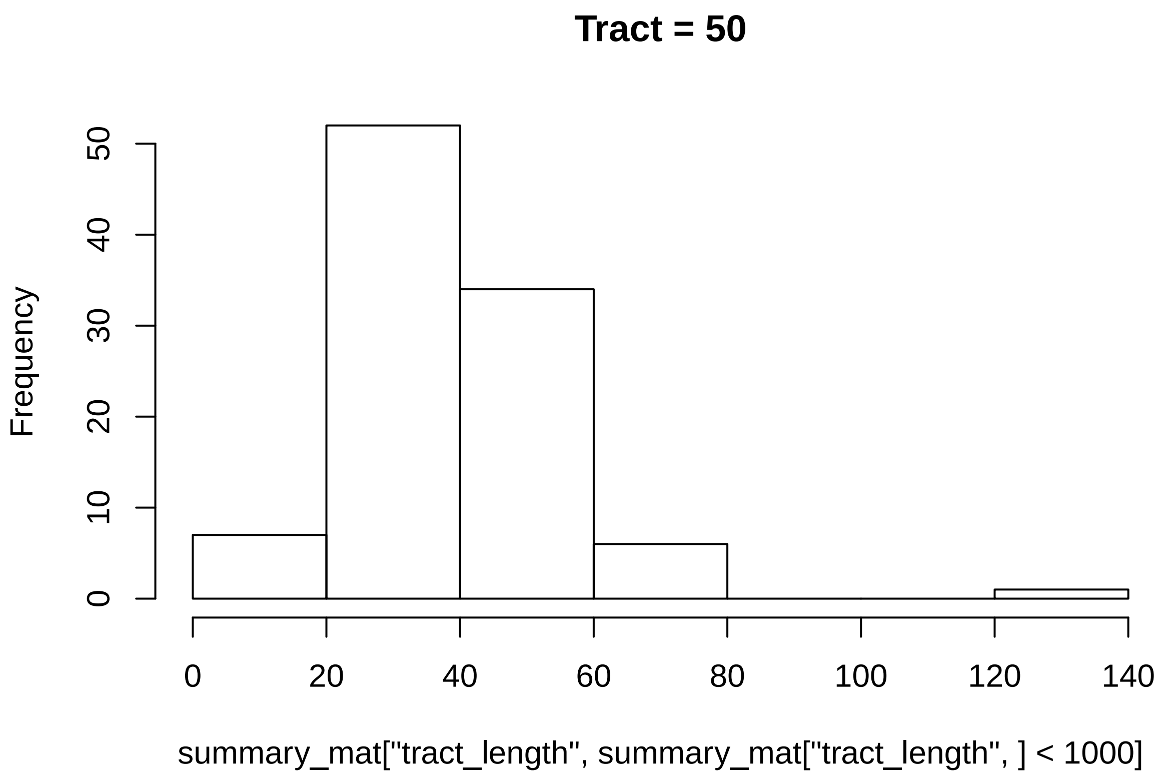
```
# exclude suspiciously long inferred tract length, plot again
for(tract in Tract.list){
  summary_mat <- get(paste("PSJS_Tract_", toString(tract), "_summary", sep = ""))
  # histogram of inferred tract length
  hist(summary_mat["tract_length", summary_mat["tract_length", ] < 1000.0], main = paste("Tract = ", toS
  print(c("mean", mean(summary_mat["tract_length", summary_mat["tract_length", ] < 1000.0]),
          "sd", sd(summary_mat["tract_length", summary_mat["tract_length", ] < 1000.0])))
}
```



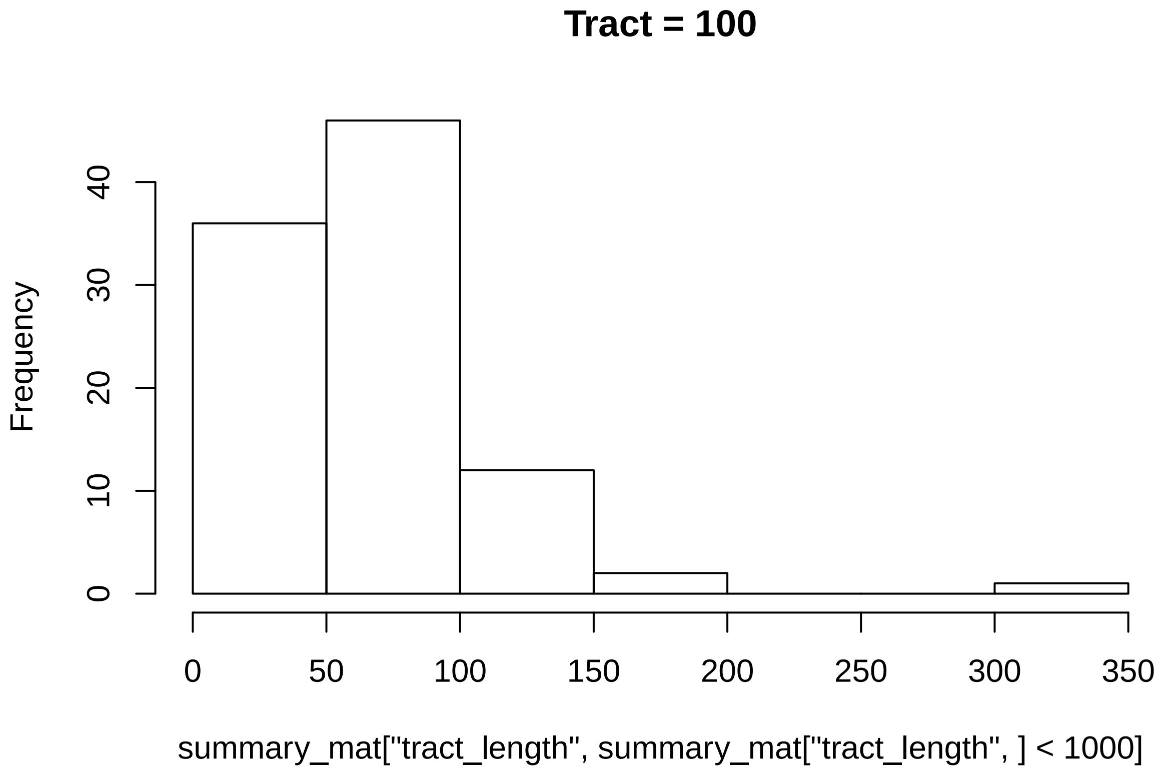
```
## [1] "mean"             "2.29718013607812" "sd"
## [4] "1.02874535788003"
```



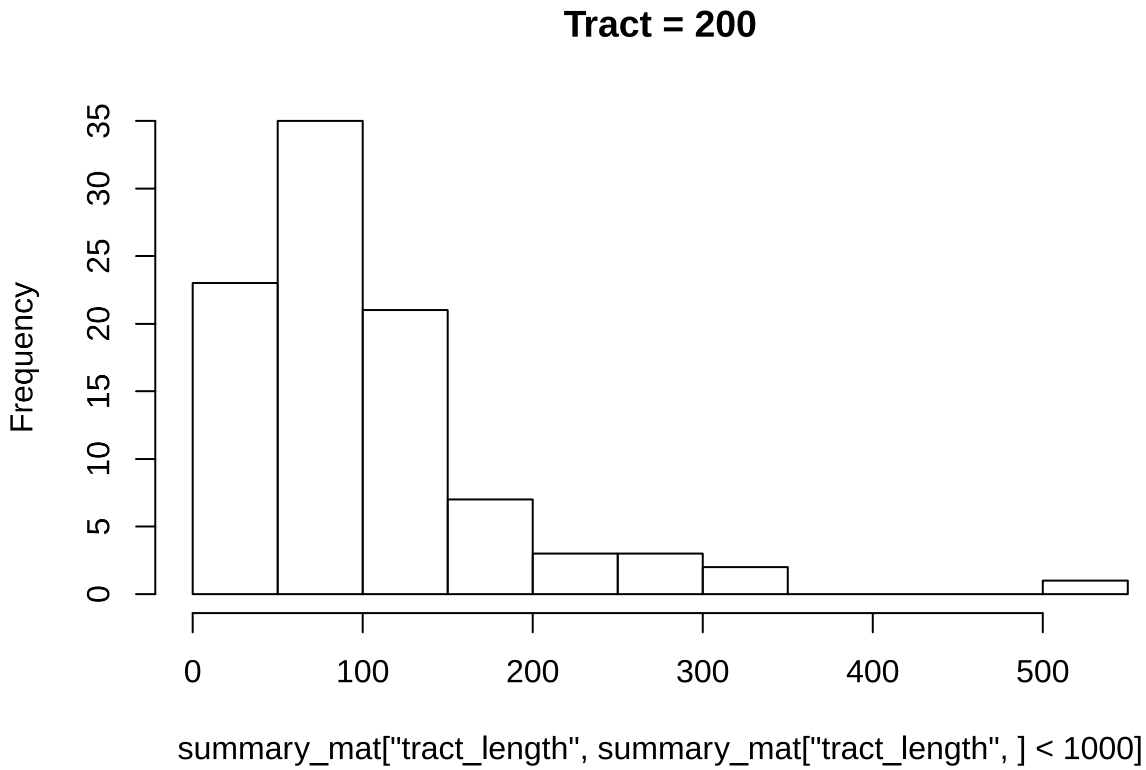
```
## [1] "mean"             "8.72620746076138" "sd"
## [4] "3.72714198660433"
```



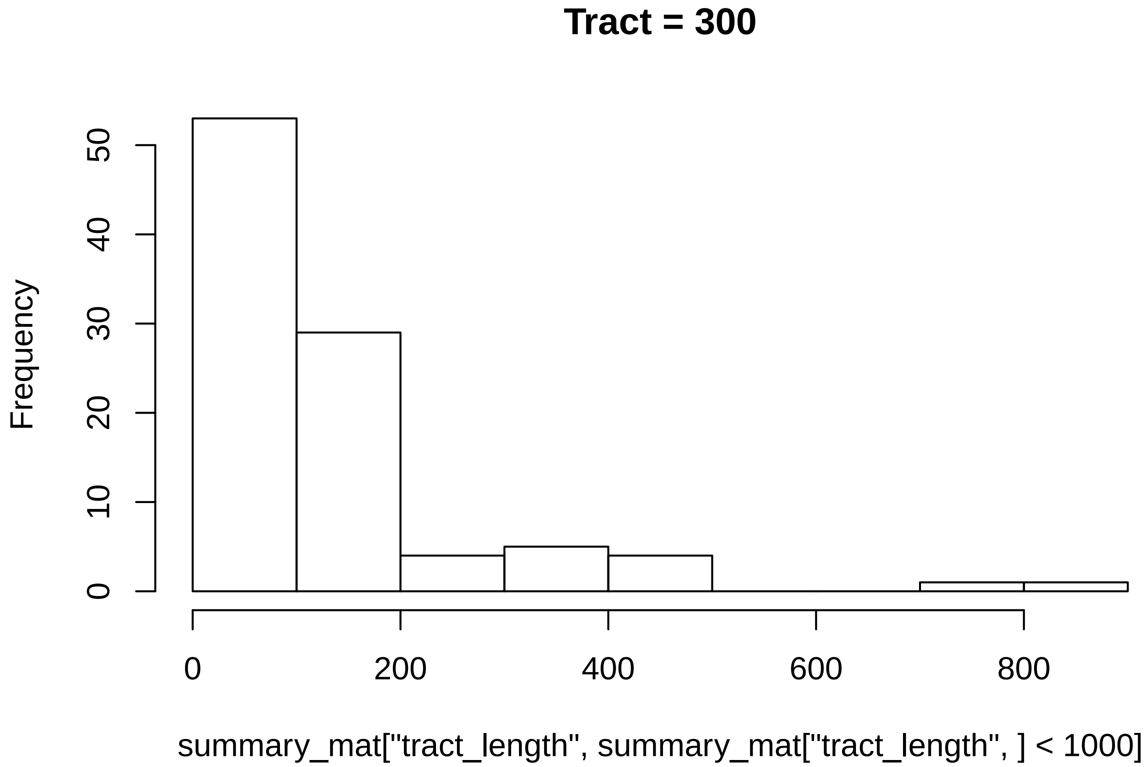
```
## [1] "mean"             "38.1001820929933" "sd"
## [4] "16.207953647501"
```



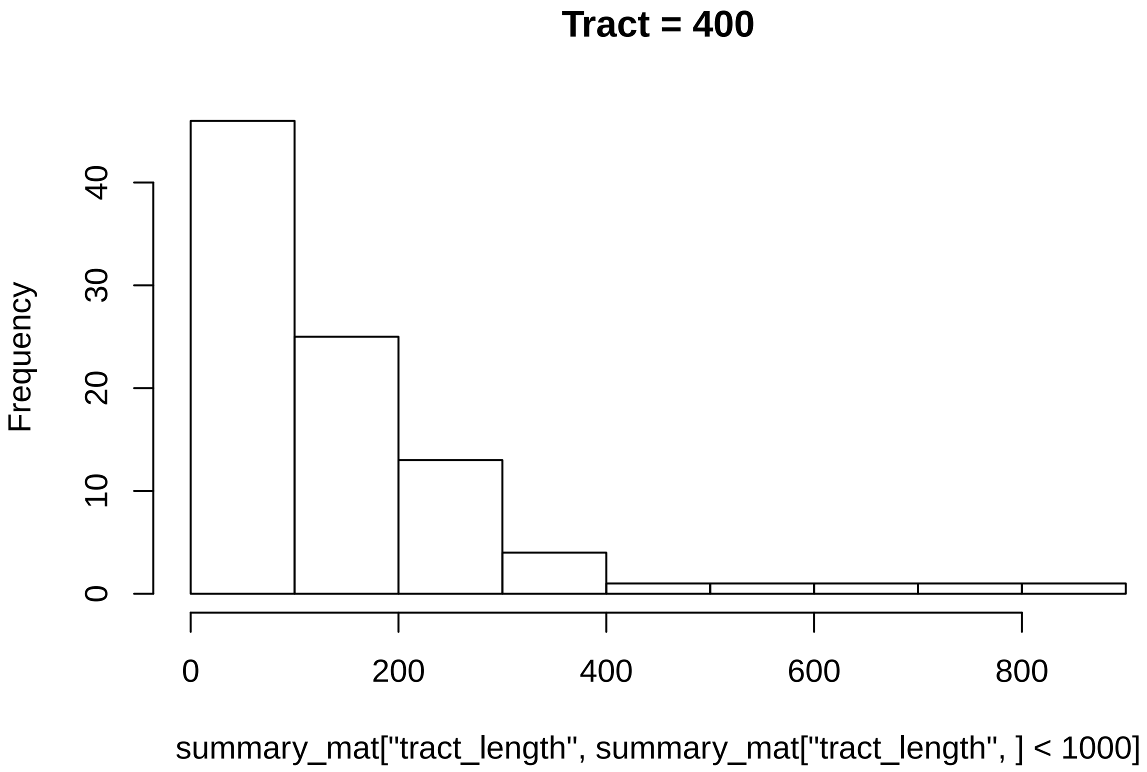
```
## [1] "mean"             "66.9946146602955" "sd"
## [4] "40.527169109434"
```



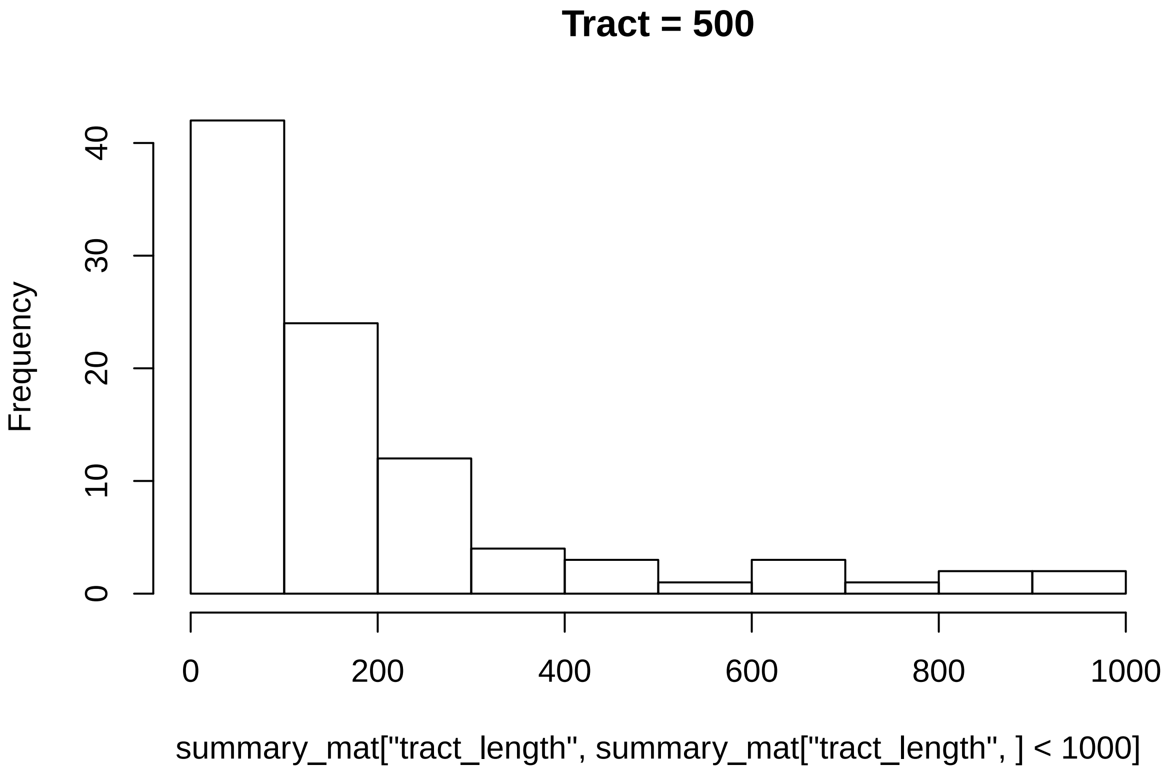
```
## [1] "mean"             "105.044108496939" "sd"
## [4] "83.9910122074207"
```



```
## [1] "mean"             "135.843591947529" "sd"
## [4] "146.132608739573"
```



```
## [1] "mean"             "142.57022420328"  "sd"
## [4] "154.570518796298"
```

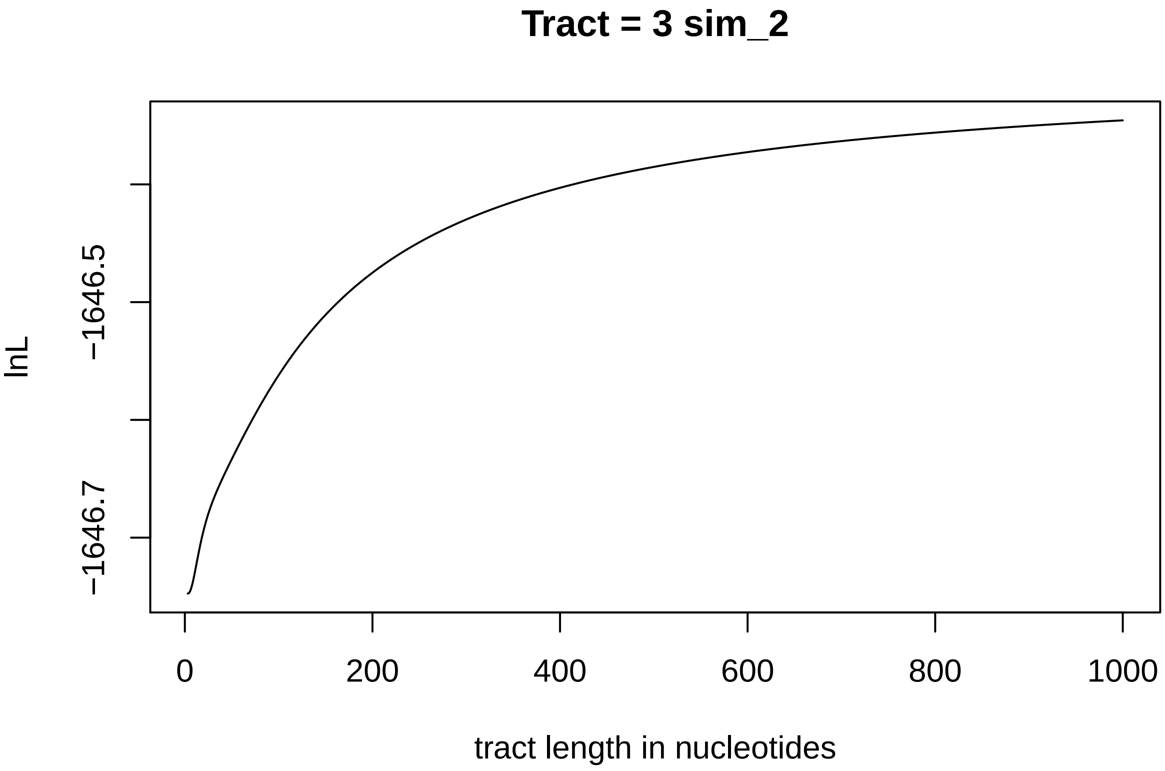


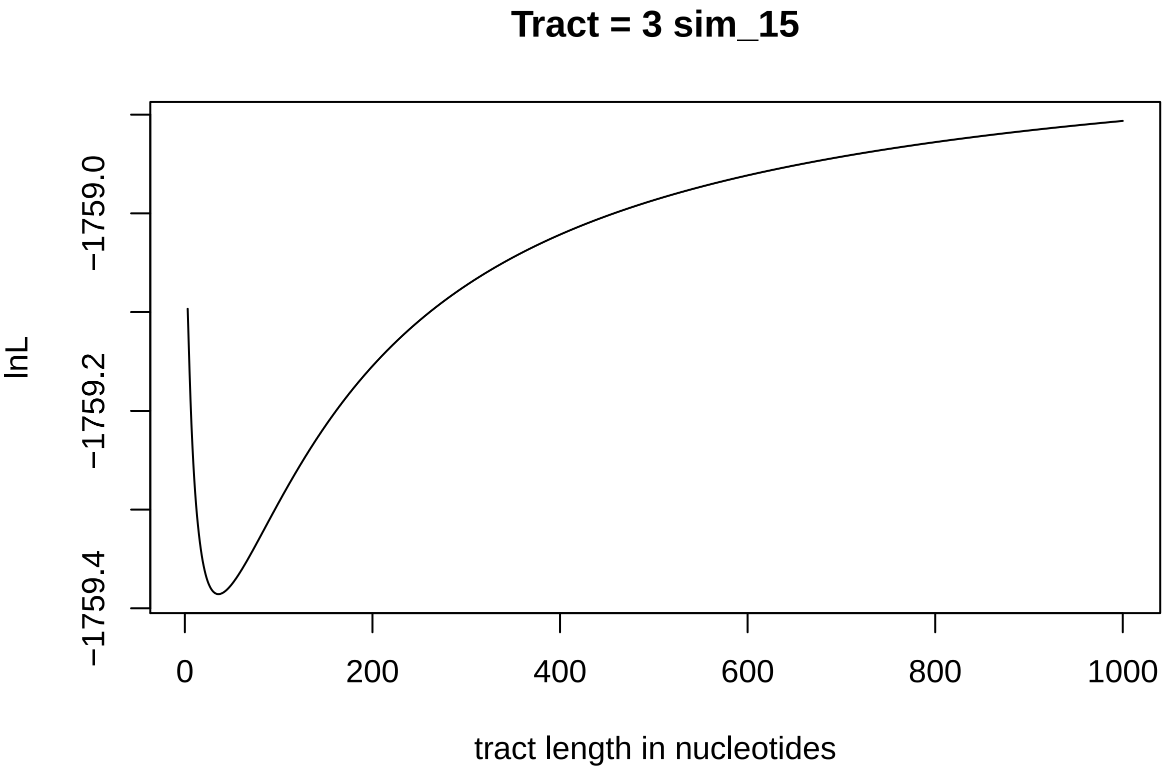
```
## [1] "mean"             "187.222502177302" "sd"
## [4] "223.924381481332"
```

A plot of HMM surface that infer tract length at boundary from each tract length condition

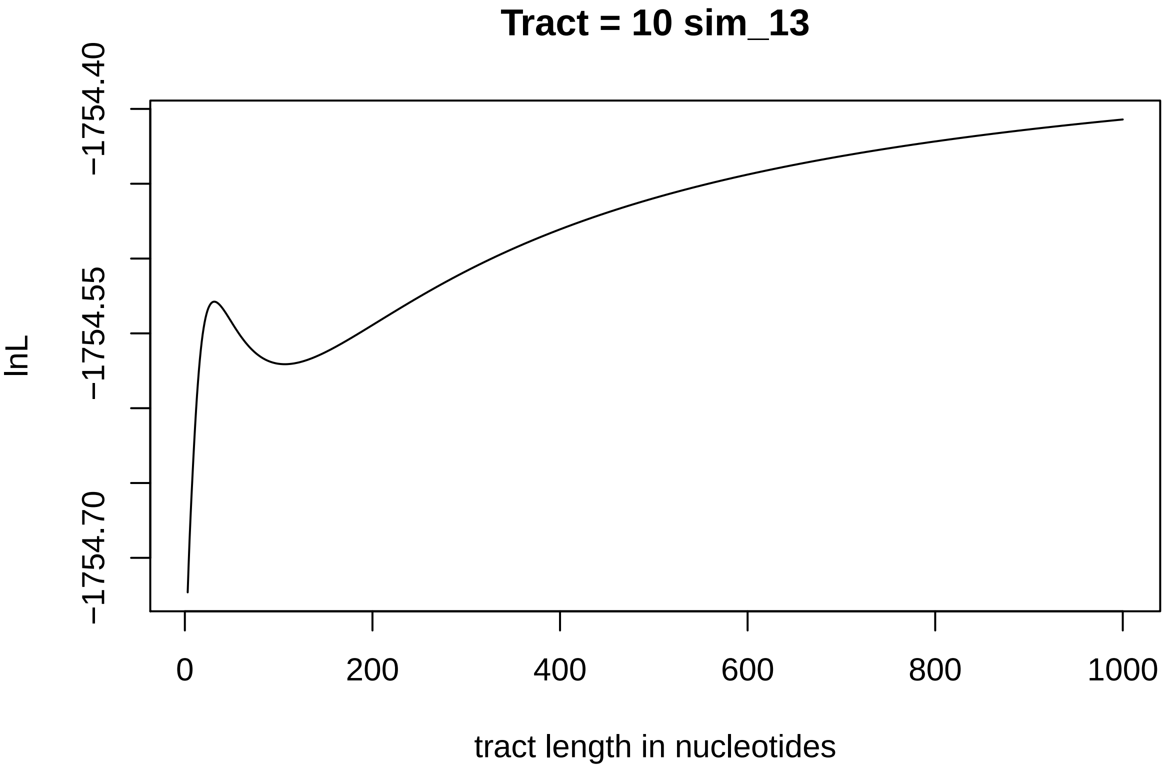
```
# plot the first two datasets that HMM inferred tract length stuck at boundary of 1000.0
for (tract in Tract.list){
  print(paste("Tract = ", toString(tract), sep = ""))
  summary_mat <- get(paste("HMM_Tract_", toString(tract), "_plot", sep = ""))
  count = 0
  for(iter in 1:99){
    if(summary_mat[1, iter] > 999. & count < 2){
      to.plot <- read.table(paste("./plot/Tract_", toString(tract), ".0/", colnames(summary_mat)[iter],
                                  "/HMM_YDR418W_YEL054C_lnL_", colnames(summary_mat)[iter],
                                  "_1D_surface.txt", sep = ""))
      plot(3.0*exp(-to.plot[, 1]), to.plot[, 2], main = paste("Tract = ", toString(tract), " ",colnames
           ylab = "lnL")
      count = count + 1
    }
  }
}
```

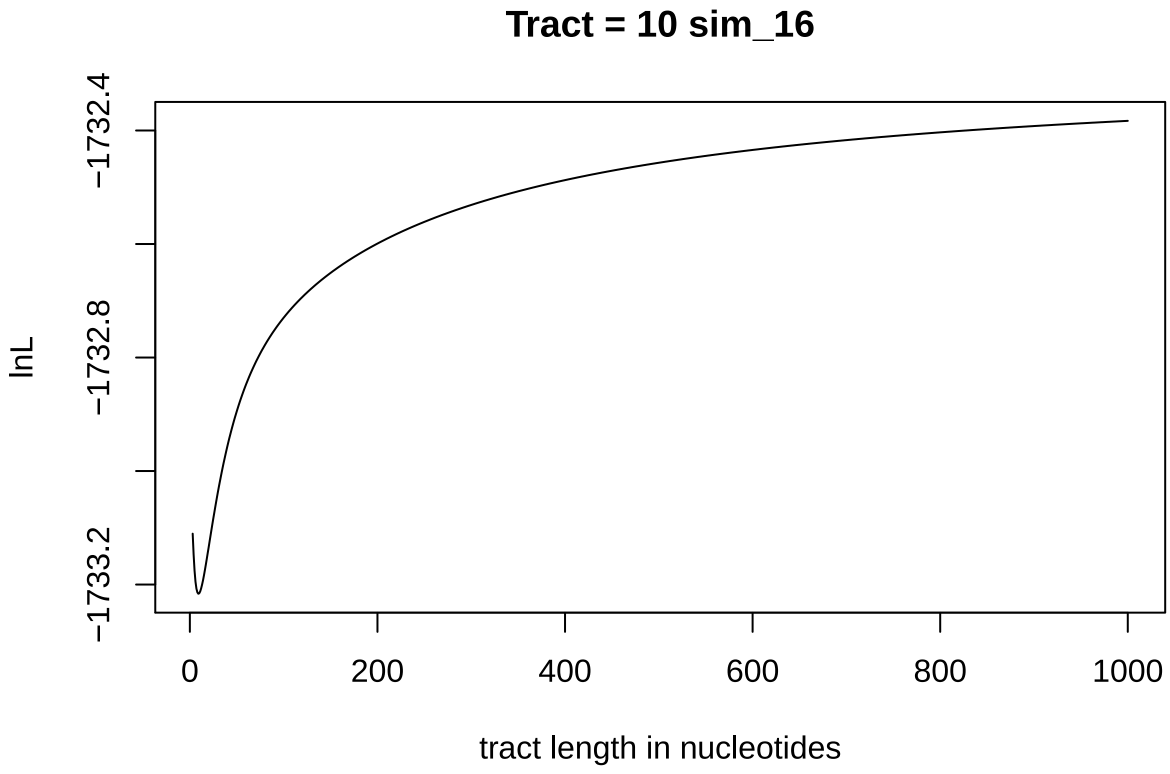
```
## [1] "Tract = 3"
```



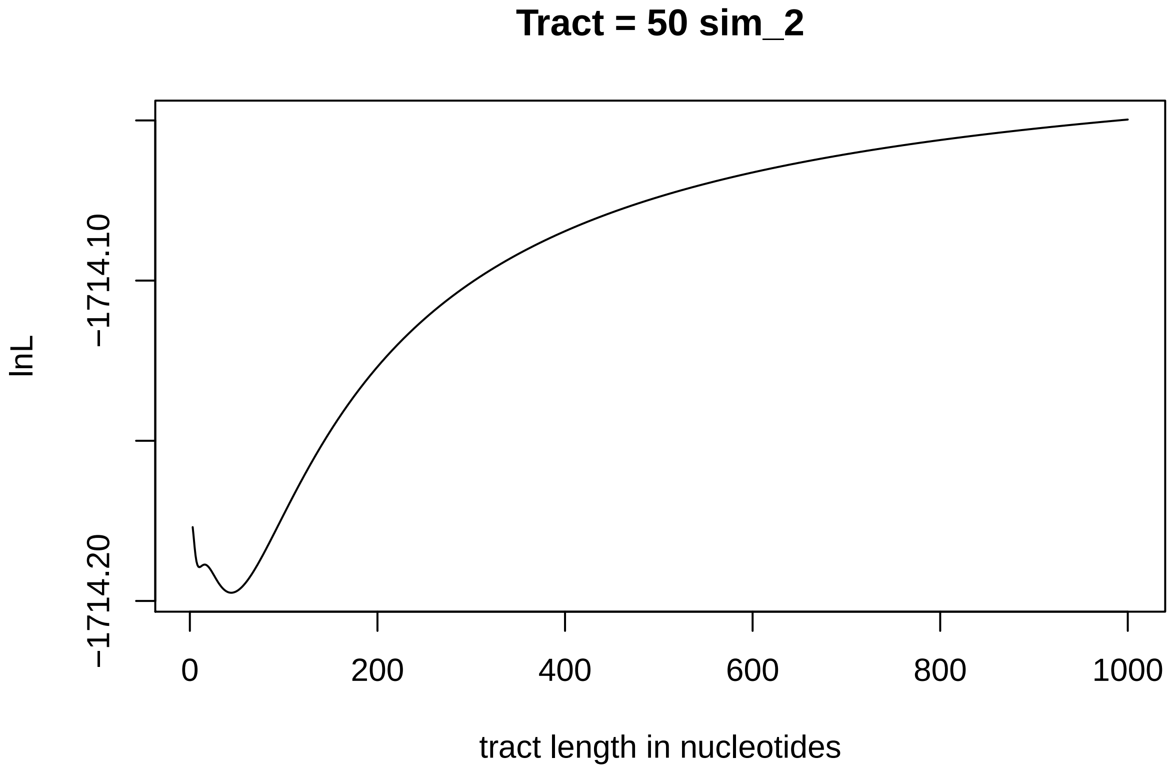

Tract = 3 sim_15
lnL
−1759.0
−1759.2
−1759.4
0
200
400
600
800
1000
tract length in nucleotides

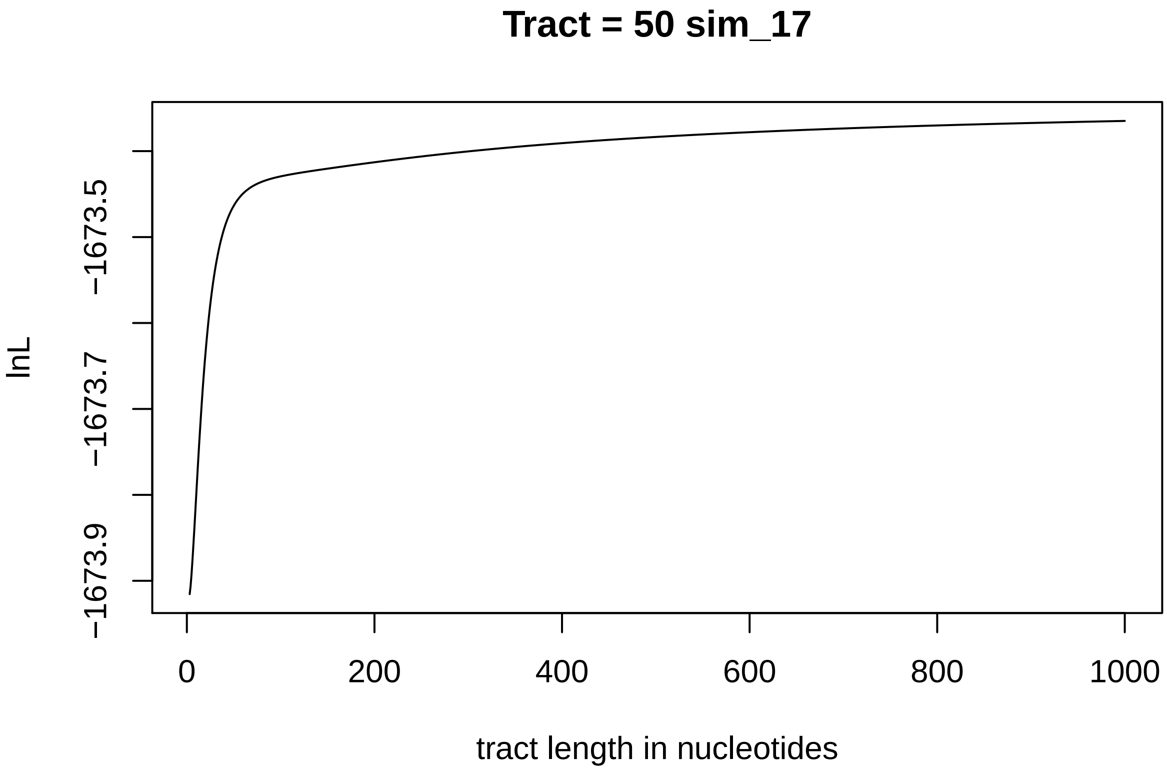
```
## [1] "Tract = 10"
```


Tract = 10 sim_13
lnL
−1754.40
−1754.55
−1754.70
0
200
400
600
800
1000
tract length in nucleotides


Tract = 10 sim_16
lnL
−1732.4
−1732.8
−1733.2
0
200
400
600
800
1000
tract length in nucleotides

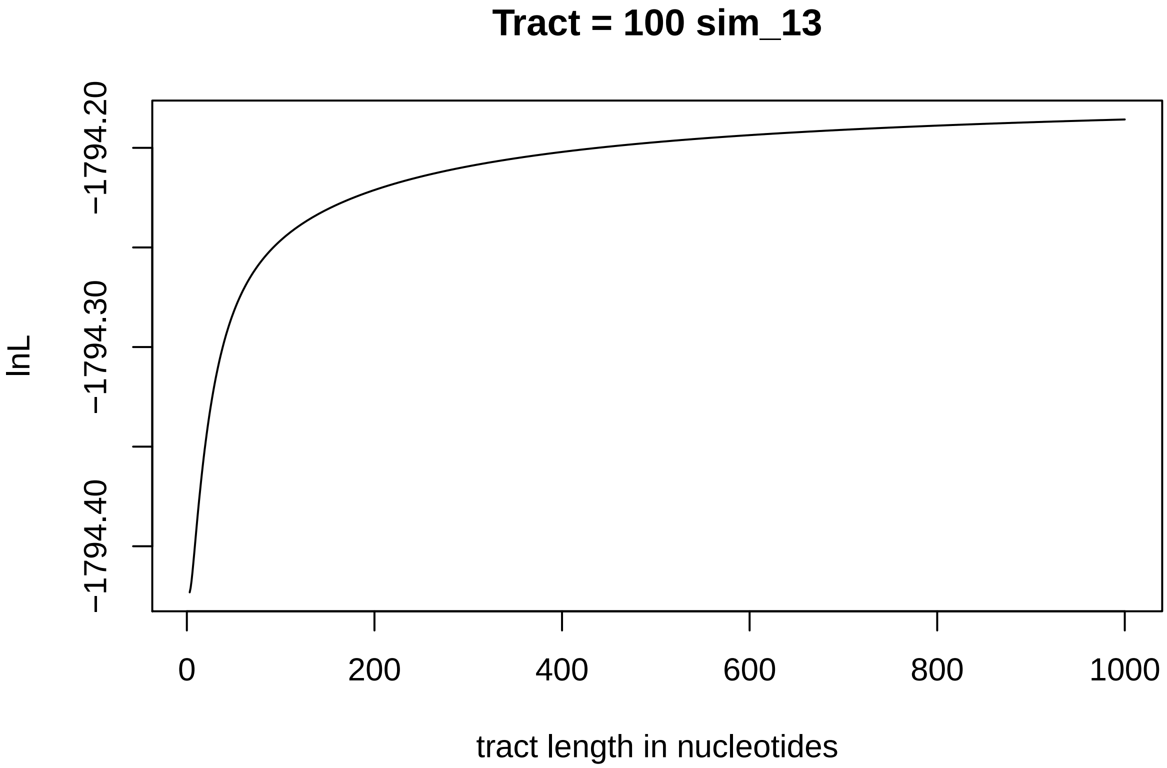
```
## [1] "Tract = 50"
```


Tract = 50 sim_2
lnL
−1714.10
−1714.20
0
200
400
600
800
1000
tract length in nucleotides

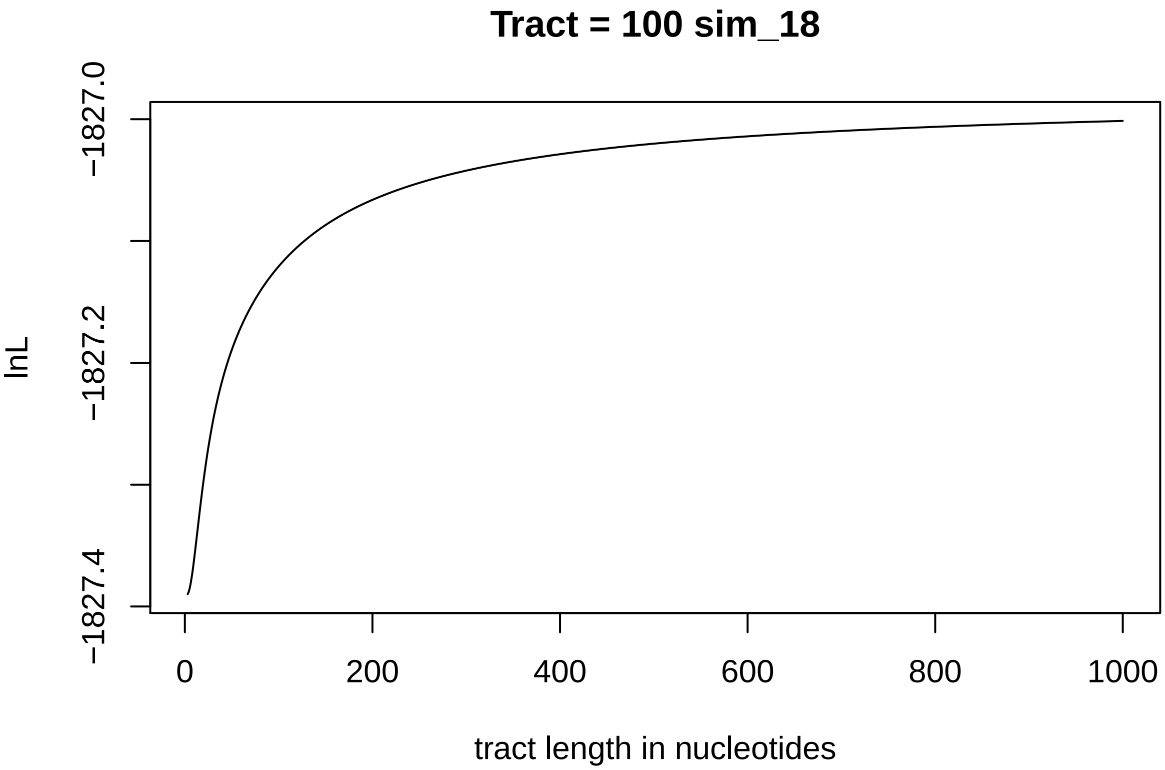


Tract = 50 sim_17

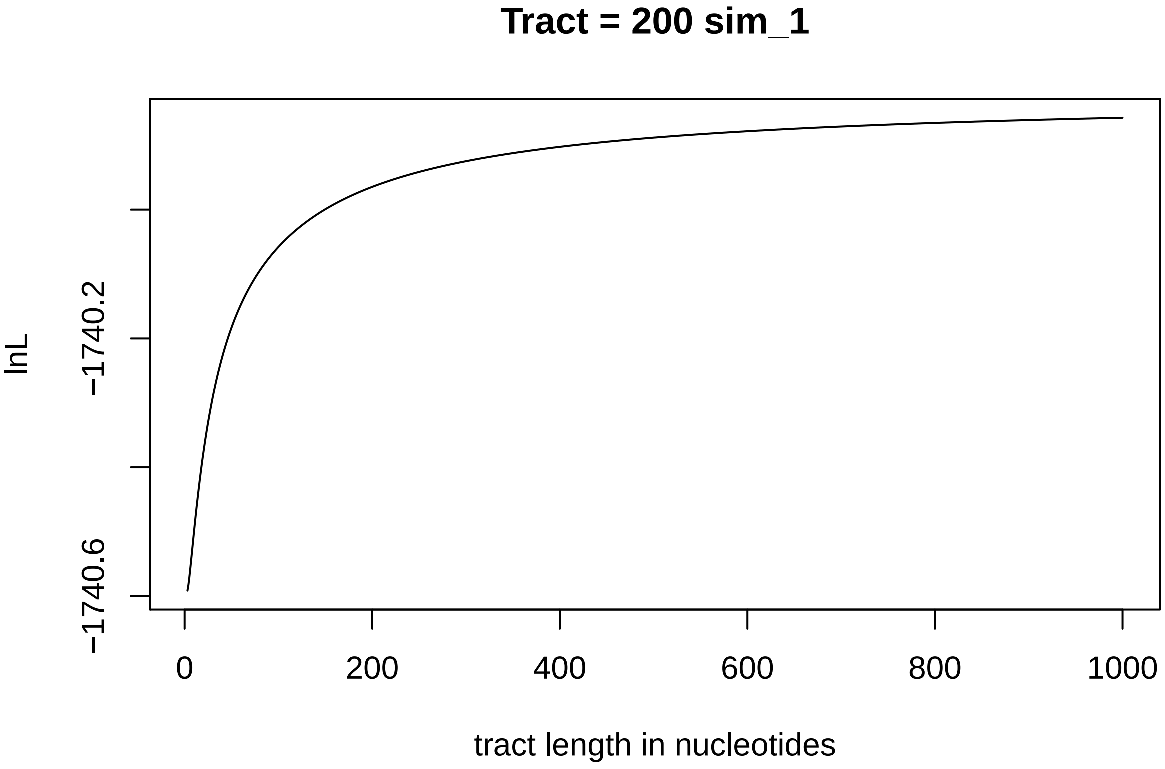
```
## [1] "Tract = 100"
```

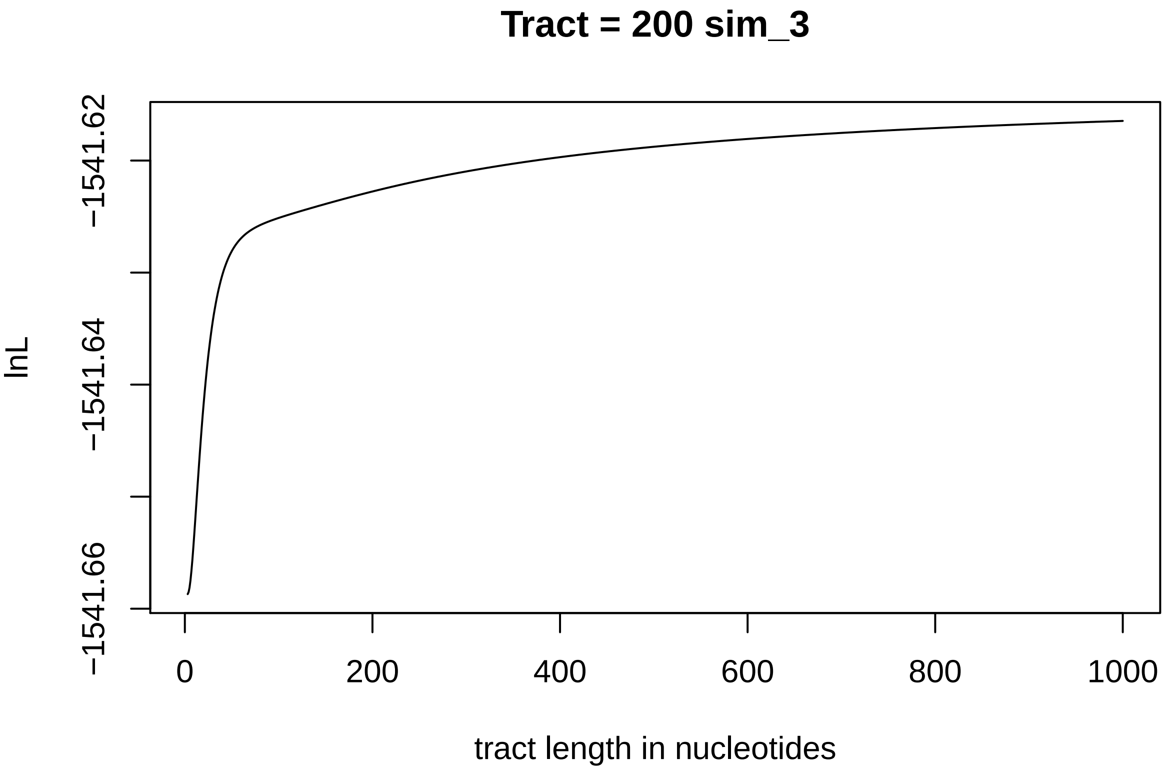


Tract = 100 sim_13

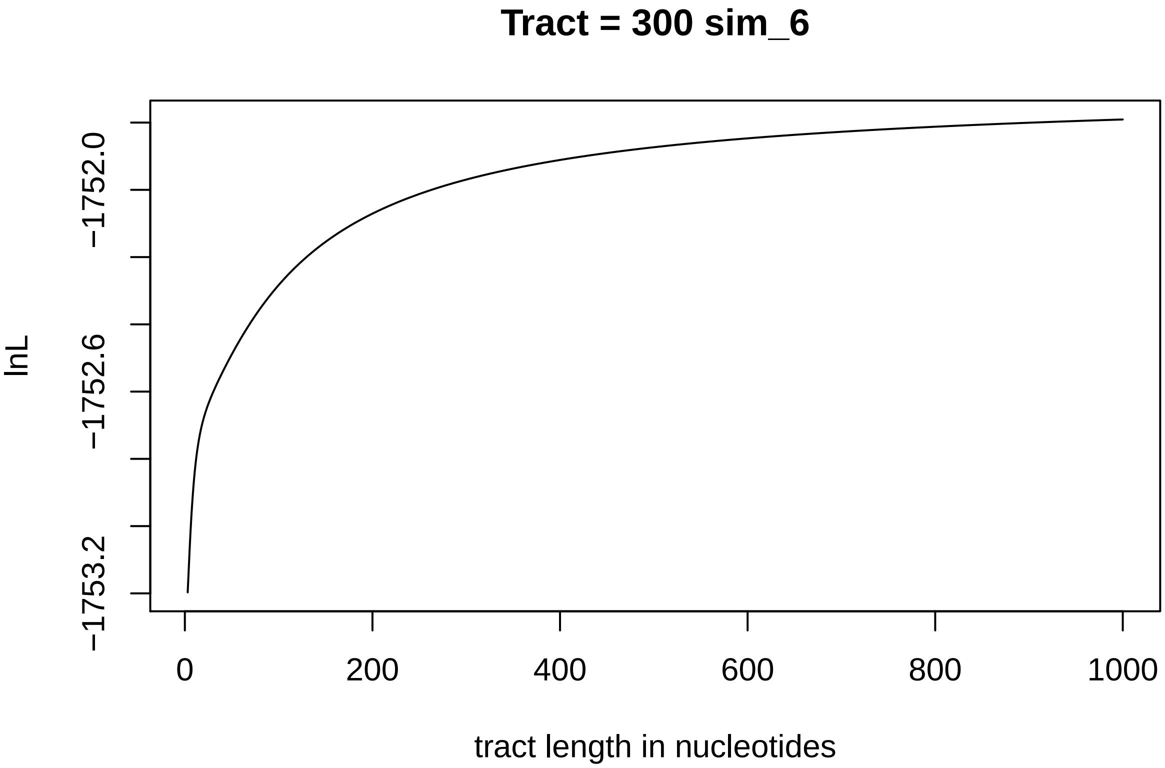

Tract = 100 sim_18
lnL
−1827.0
−1827.2
−1827.4
0
200
400
600
800
1000
tract length in nucleotides

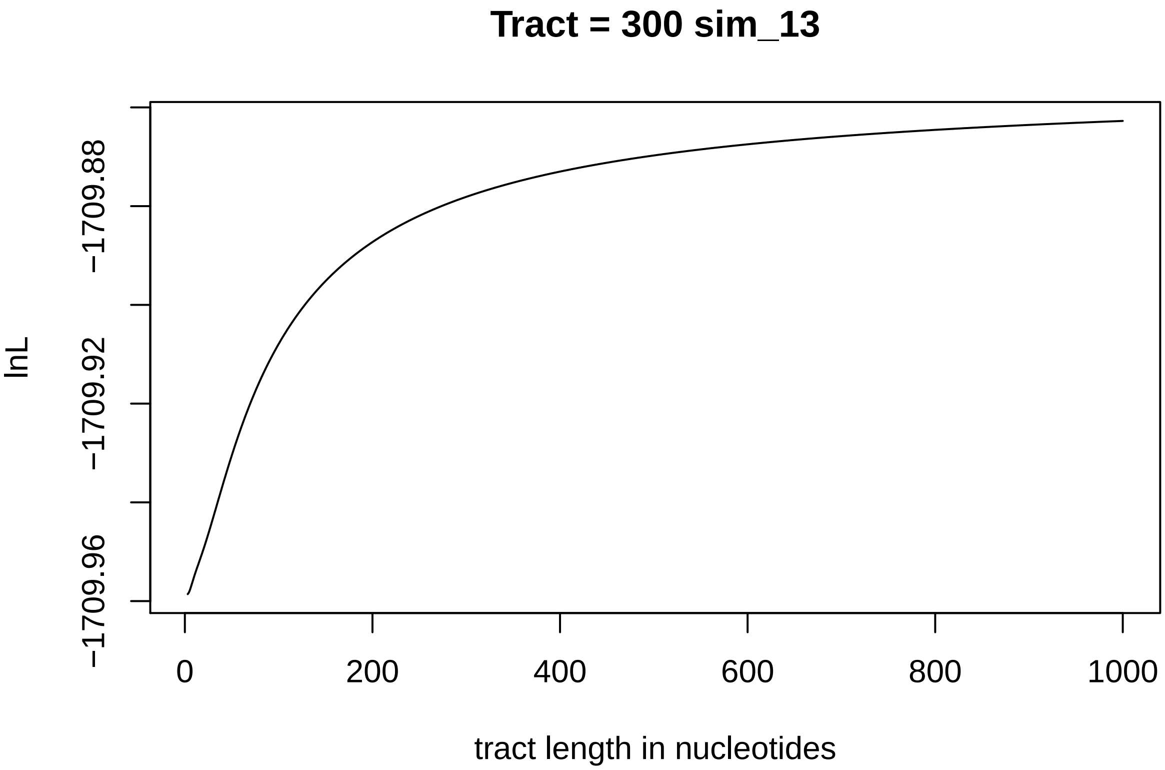
```
## [1] "Tract = 200"
```


Tract = 200 sim_1
lnL
−1740.2
−1740.6
0
200
400
600
800
1000
tract length in nucleotides

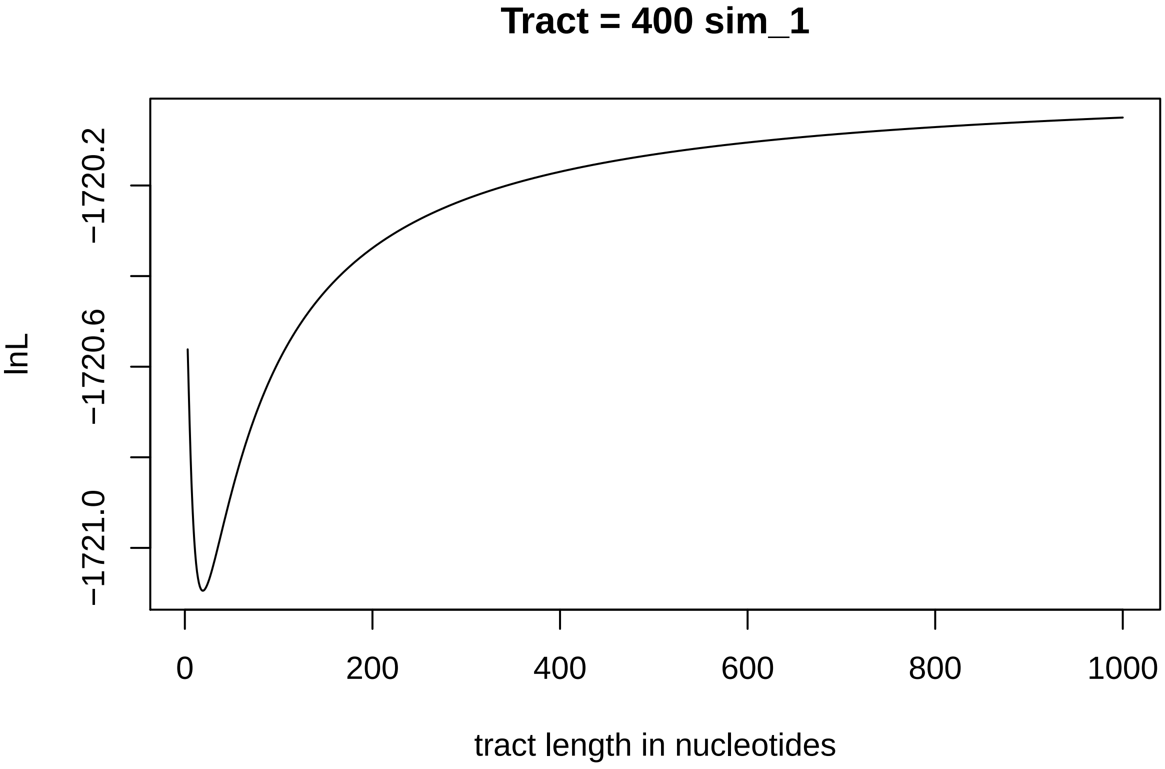

Tract = 200 sim_3
lnL
−1541.62
−1541.64
−1541.66
0
200
400
600
800
1000
tract length in nucleotides

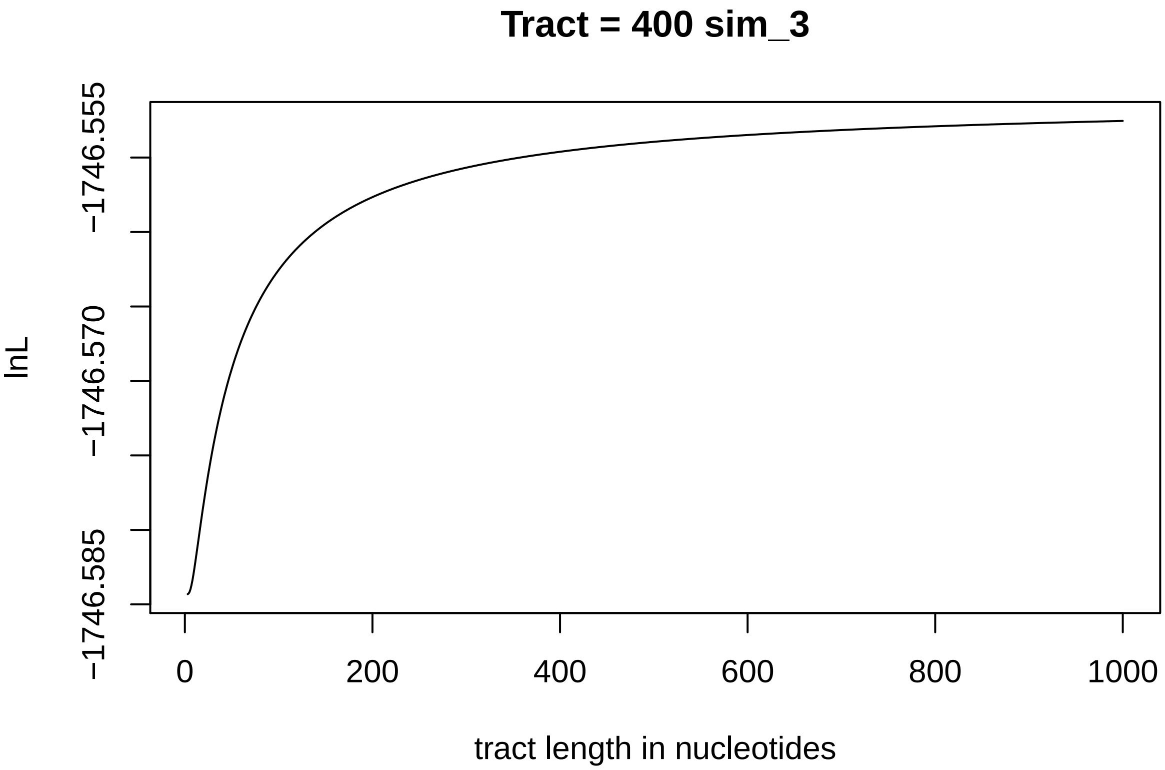
```
## [1] "Tract = 300"
```


Tract = 300 sim_6
lnL
−1752.0
−1752.6
−1753.2
0
200
400
600
800
1000
tract length in nucleotides

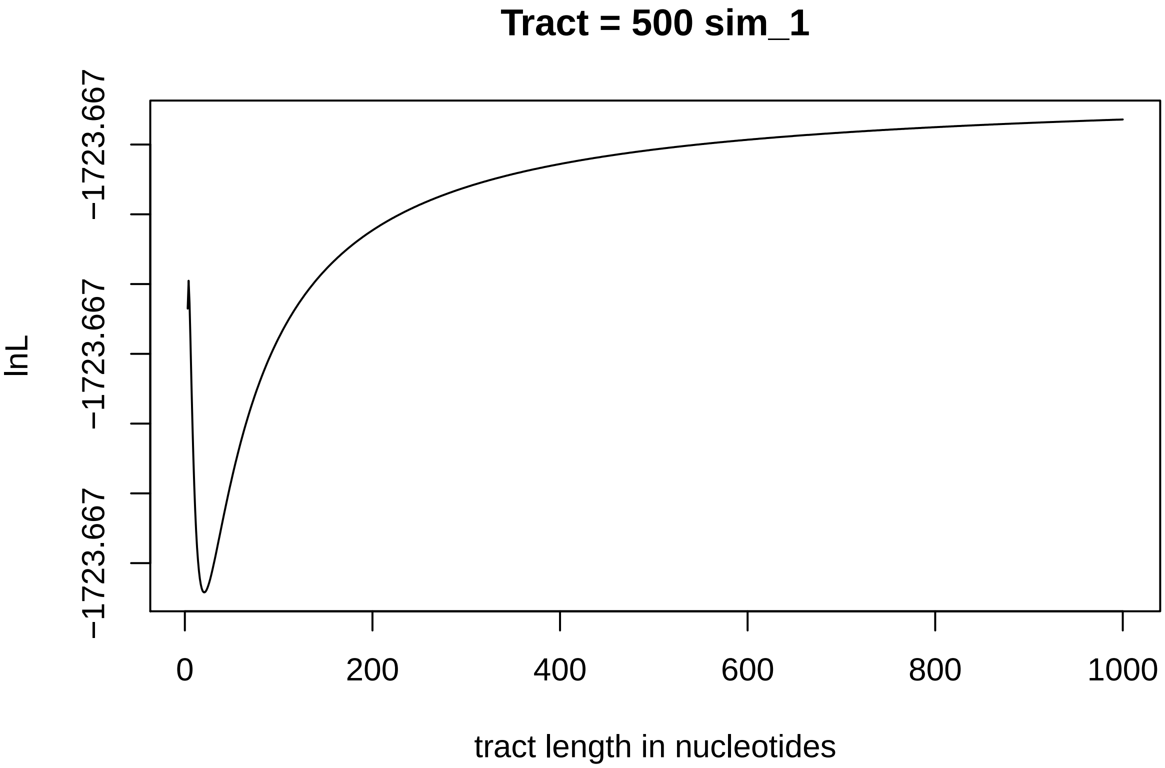

Tract = 300 sim_13
lnL
−1709.88
−1709.92
−1709.96
0
200
400
600
800
1000
tract length in nucleotides

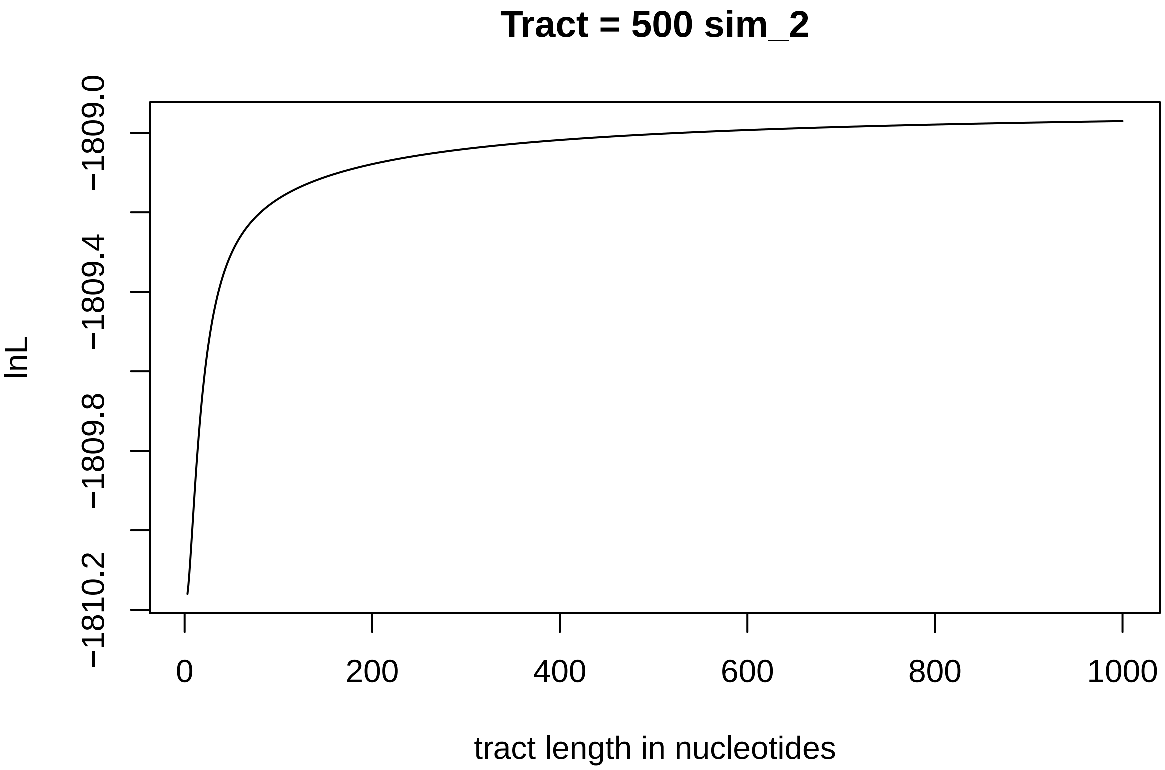
```
## [1] "Tract = 400"
```


Tract = 400 sim_1
lnL
−1720.2
−1720.6
−1721.0
0
200
400
600
800
1000
tract length in nucleotides


Tract = 400 sim_3
lnL
−1746.555
−1746.570
−1746.585
0
200
400
600
800
1000
tract length in nucleotides

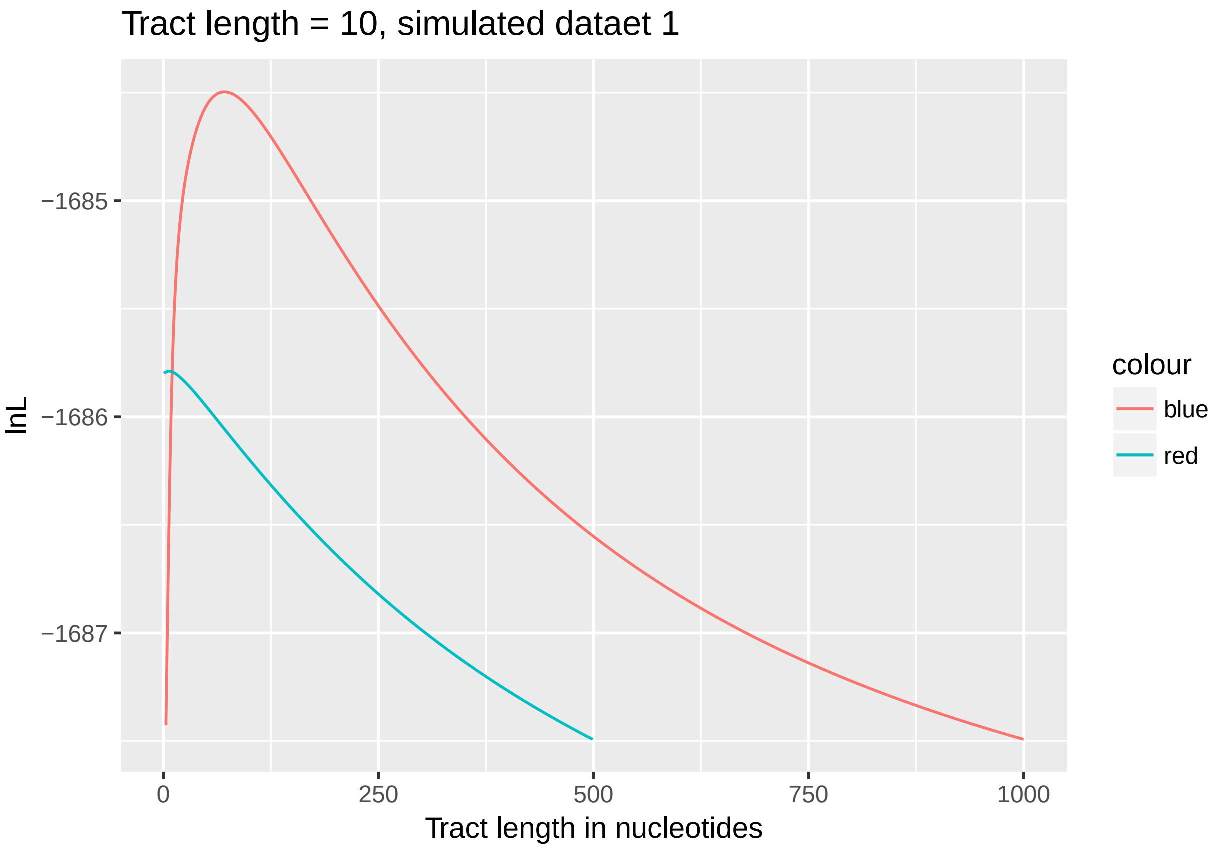
```
## [1] "Tract = 500"
```


Tract = 500 sim_1
lnL
−1723.667
−1723.667
−1723.667
0
200
400
600
800
1000
tract length in nucleotides

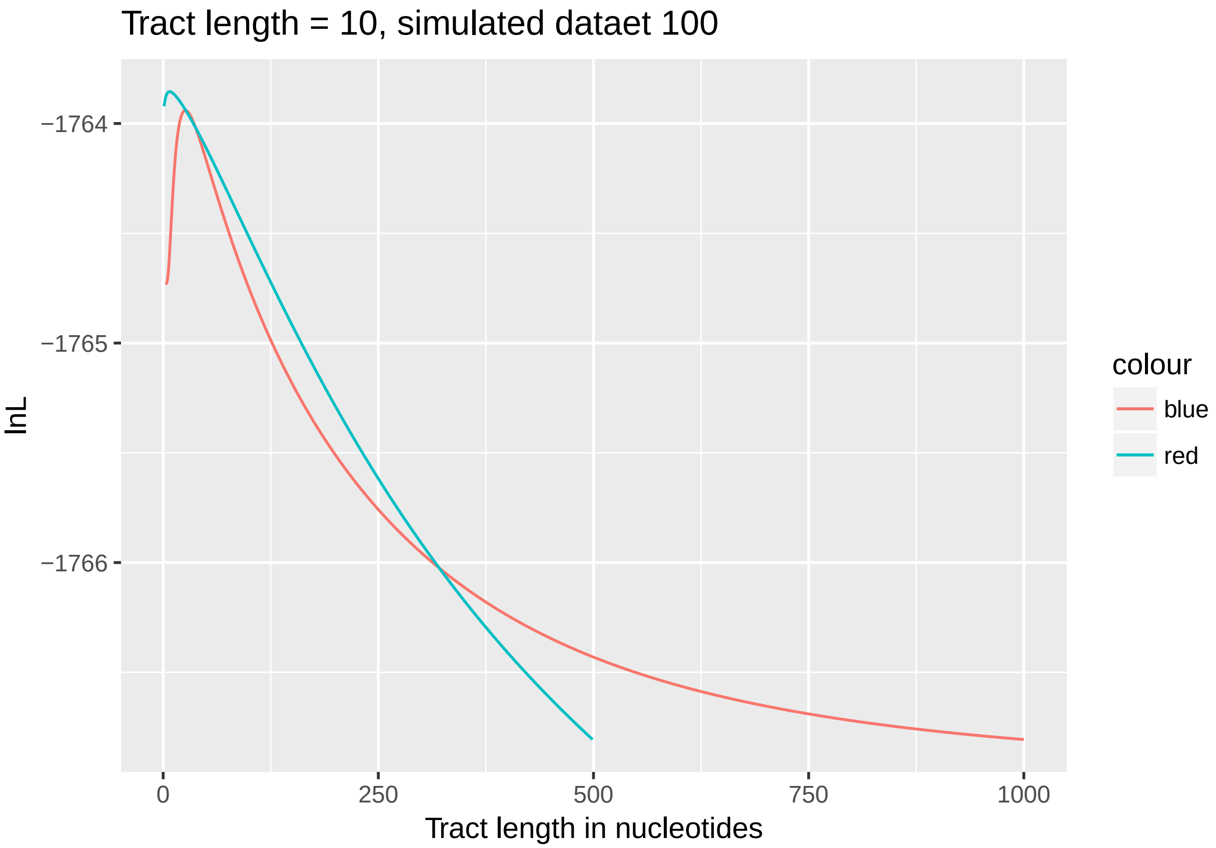


Now plot the two plots of lnL

```
library(ggplot2)
# Tract length = 10
hmm.plot <- read.table("./plot/Tract_10.0/sim_1/HMM_YDR418W_YEL054C_lnL_sim_1_1D_surface.txt")
PSJS.plot <- read.table("./plot/Tract_10.0/sim_1/PSJS_HKY_rv_sim_1_Tract_10.0_lnL_1D_surface.txt")
ggplot(mapping = aes(x = 3.0*exp(-hmm.plot[,1]), y = hmm.plot[, 2], colour = "blue")) + geom_line() +
  geom_line(aes(x = exp(-PSJS.plot[,1]), y = PSJS.plot[, 2]/488 + min(hmm.plot[, 2]) - min(PSJS.plot[,
              colour = "red")) +
  xlab("Tract length in nucleotides") +
  ylab("lnL") +
  ggtitle("Tract length = 10, simulated dataet 1")
```



```
hmm.plot <- read.table("./plot/Tract_10.0/sim_100/HMM_YDR418W_YEL054C_lnL_sim_100_1D_surface.txt")
PSJS.plot <- read.table("./plot/Tract_10.0/sim_100/PSJS_HKY_rv_sim_100_Tract_10.0_lnL_1D_surface.txt")
ggplot(mapping = aes(x = 3.0*exp(-hmm.plot[,1]), y = hmm.plot[, 2], colour = "blue")) + geom_line() +
  geom_line(aes(x = exp(-PSJS.plot[,1]), y = PSJS.plot[, 2]/488 + min(hmm.plot[, 2]) - min(PSJS.plot[, 2
                colour = "red")) +
  xlab("Tract length in nucleotides") +
  ylab("lnL") +
  ggtitle("Tract length = 10, simulated dataet 100")
```



save workspace now

```
save.image("./SimulationStudy.RData")
```