Estimating time of origin of 2019-nCoV assuming a star phylogeny

Erik Volz e.volz@imperial.ac.uk

January 24, 2020

Following du Plessis & Pybus, we can estimate the TMRCA of a star-phylogeny with heterochronous sampling and given a molecular clock rate: http://virological.org/uploads/short-url/egpzsjtErKV4EIwOrYhy0qhgik1.pdf NOTE: There are a lot of assumptions baked into this which also apply to this analysis.

Here I modify this approach slightly by considering the likelihood of each branch of a star phylogeny. This is based on the idea that each branch is an independent realization of a Poisson process with different rates. This is the product of Poisson densities for each branch in the star:

$$p(\mathcal{G}|\omega, s, x_{1:n}, t_{1:n}, t_0) = \prod_{i=1}^{n} \frac{e^{-s\omega(t_i - t_0)}(s\omega(t_i - t_0))^{x_i}}{x_i!}$$

Where

- $p(\mathcal{G})$ is the probability density of a genealogy (in this case a star),
- ω is the rate of substitutions per site per year,
- s is the genome length,
- x is a vector giving the number of SNPs observed in each sample
- t is the vector of sample times
- and, t_0 is the time of the root

These are the data obtained from du Plessis & Pybus:

```
library( lubridate)
d = data.frame(
 date = ymd( c("2020/01/16", "2020/01/17", "2019/12/30", "2019/12/30", "2019/12/30", "2019/12/30"
     , "2019/12/30", "2020/01/13", "2020/01/08", "2019/12/30", "2019/12/30", "2019/12/30", "2019/12/24"
     , "2019/12/26", "2019/12/30", "2019/12/30", "2019/12/30", "2020/01/17", "2020/01/15", "2020/01/14")
 , snps = c(2, 0, 1, 2, 2, 0, 2, 0, 0, 0, 2, 0, 3, 0, 0, 1, 0, 1, 1, 3)
 , t = c(2020.04098360656, 2020.04371584699, 2019.99452054795, 2019.99452054795
     , 2019.99452054795, 2019.99452054795, 2019.99452054795, 2020.03278688525, 2020.01912568306
     , 2019.99452054795, 2019.99452054795, 2019.99452054795, 2019.97808219178, 2019.98356164384
     , 2019.99452054795, 2019.99452054795, 2019.99452054795, 2020.04371584699, 2020.03825136612
     , 2020.03551912568)
)
d
```

```
##          date snps        t
## 1  2020-01-16    2 2020.041
## 2  2020-01-17    0 2020.044
## 3  2019-12-30    1 2019.995
## 4  2019-12-30    2 2019.995
## 5  2019-12-30    2 2019.995
## 6  2019-12-30    0 2019.995
## 7  2019-12-30    2 2019.995
## 8  2020-01-13    0 2020.033
## 9  2020-01-08    0 2020.019
## 10 2019-12-30    0 2019.995
## 11 2019-12-30    2 2019.995
```

```
## 12 2019-12-30    0 2019.995
## 13 2019-12-24    3 2019.978
## 14 2019-12-26    0 2019.984
## 15 2019-12-30    0 2019.995
## 16 2019-12-30    1 2019.995
## 17 2019-12-30    0 2019.995
## 18 2020-01-17    1 2020.044
## 19 2020-01-15    1 2020.038
## 20 2020-01-14    3 2020.036
```

Here, I define two likelihood functions. The first is identical to what was defined by du Plessis & Pybus. The second uses the branch-wise model.

```
#' @param tmrca numeric
#' @param rate scalar or vector or rates in units of subst / genome / year
loglik_treelength <- function( tmrca, rate  ){
    blen <- d$t - tmrca
    if ( any (  blen < 0  ))
        return ( NA )
    trelen <- sum( blen )
    mean( dpois( sum( d$snps ), lambda =  trelen * rate, log = TRUE ) )
}

#' @param tmrca numeric
#' @param rate scalar or vector or rates in units of subst / genome / year
loglik_branch <- function( tmrca, rate  ){
    blen <- d$t - tmrca
    if ( any (  blen < 0  ))
        return ( NA )

    mean( sapply( rate, function(r) sum( dpois(  d$snps, lambda = r*blen , log = TRUE ) ) ) )
}
```

Here I define some basic parameters for the analysis: genome length, the range of dates to examine, and the range of rates to consider. I will look at two scenarios:

1. Rates in teh range 0.00083-0.00109 subst/ site / year
    - This was used for the main result of du Plessis and Pybus and was based on the earlier study of Dudas
2. A somewhat broader range that encompasses rates seen for other coronaviruses: 0.005-0.001

```
s <- 29903 # genome length
# substitution rate (per genome)
rates1 <- s * seq( 8.3e-4, 0.00109, length = 1e3 )
rates2 <- s * seq( 5e-4, 0.00109, length = 1e3 )

daterange <- seq( as.Date( '2019-12-05'), as.Date( '2019-12-24') , by = .001)
xrange <- decimal_date( daterange )
```

This function computes the likelihoods over a range of dates, produces plots, and computes CIs:

```
plot_densities <- function( rates, ofn ='ncov2019-starTreeTRMCA.pdf' ){
    # likelihoods
    l_tl <- sapply( xrange, function(x) loglik_treelength( x, rates ))
    l_b <-  sapply( xrange, function(x) loglik_branch ( x, rates ))

    # convert to densiy
```

```
    d_tl <- exp( l_tl ) / sum( exp( na.omit( l_tl ) ))
    d_b <- exp( l_b ) / sum( exp( na.omit(l_b) ))

    # compute ci
    ci <- function(  dens  ){
        i <- c( max( which( cumsum( dens ) < .025 ))
          , min( which( cumsum( dens ) >.975 )) )
        daterange[ i ]
    }

    cat( 'Estimated TMRCA, tree length and branch model:\n' )
    print ( c( daterange[which.max(d_tl)], daterange[which.max(d_b)] ) )
    cat('CI for tree length model:\n')
    print( ci(  d_tl) )
    cat('CI for branch model:\n')
    print( ci(  d_b) )

    pdf(ofn)
    plot( daterange, d_tl, type = 'l', col = 'black' , ylab = 'Density', xlab = ''
        , main = 'Black: Tree length likelihood, Red: Branch length likelihood')
    abline( v = ci(  d_tl) , col = 'black',  lty =3 )
    lines( daterange, d_b, type = 'l' , col = 'red', lty = 1)
    abline( v = ci(  d_b) , col = 'red',  lty =3 )
    dev.off()
}
```

```
plot_densities( rates1, ofn = 'ncov2019-starTreeTRMCA-rates00083-001.png'  )
plot_densities( rates2, ofn = 'ncov2019-starTreeTRMCA-rates0005-001.png'  )
```

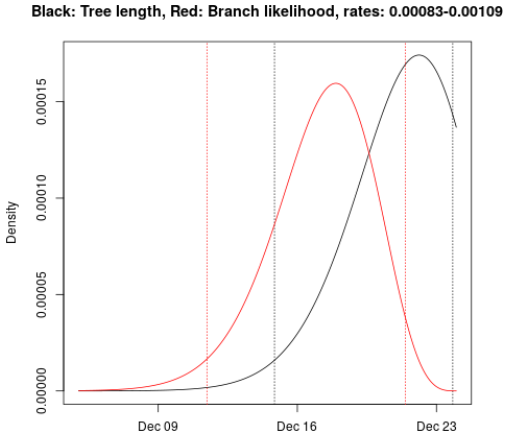
Here are the estimated dates in rate scenario 1:

```
Estimated TMRCA, tree length and branch model:
[1] "2019-12-22" "2019-12-17"
CI for tree length model:
[1] "2019-12-14" "2019-12-23"
CI for branch model:
[1] "2019-12-11" "2019-12-21"
```

Here are the estimated dates in rate scenario 2:

```
Estimated TMRCA, tree length and branch model:
[1] "2019-12-19" "2019-12-15"
CI for tree length model:
[1] "2019-12-11" "2019-12-23"
CI for branch model:
[1] "2019-12-08" "2019-12-20"
```

Here is the density for rate scenario 1 (black line similar to du Plessis & Pybus Fig 3)



Here is the density for scenario 2: