

GrowthRate_BacterialIsolates

Aied Garcia

2/14/2022

```
#install.packages("growthcurver", "nlme", "bbmle", "optimx", "tidyverse")
```

```
library("ggplot2")  
library("nlme")  
library("bbmle")
```

```
## Loading required package: stats4
```

```
library("optimx")
```

```
##  
## Attaching package: 'optimx'
```

```
## The following object is masked from 'package:nlme':  
##  
## coef<-
```

```
require("tidyverse")
```

```
## Loading required package: tidyverse
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v tibble  3.1.7    v dplyr   1.0.9  
## v tidyr   1.2.0    v stringr 1.4.0  
## v readr   2.1.2    v forcats 0.5.1  
## v purrr   0.3.4
```

```
## -- Conflicts ----- tidyverse_conflicts() --  
## x dplyr::collapse() masks nlme::collapse()  
## x dplyr::filter()   masks stats::filter()  
## x dplyr::lag()      masks stats::lag()  
## x dplyr::slice()    masks bbmle::slice()
```

```
library("tidyr")  
library("growthcurver")
```

```
read.synergy <- function(input = " ", sep = ","){
  temp <- readLines(input, warn = F)
  t0 <- which(grepl("0:00:00", temp))[1]
  tf <- which(grepl("Results", temp)) - 2
  data.in <- read.csv2(input, sep = ",", header = F)
  names <- as.character(data.in[t0 - 1, ])
  names[2] <- "Temp"
  data.out <- data.in[t0:tf, ]
  colnames(data.out) <- names
  data.out$Time <- as.character(data.out$Time)
  t.h <- as.numeric(lapply(strsplit(data.out$Time, "\\:"), "[", 1))
  t.m <- as.numeric(lapply(strsplit(data.out$Time, "\\:"), "[", 2))
  data.out$Time <- round(t.h + t.m/60, 2)
  for (i in 1:dim(data.out)[2]){
    if (is.numeric(data.out[,i]) == FALSE){data.out[,i] = as.numeric(data.out[,i])}
  }
  return(data.out)
}
```

```
BacCP <- read.synergy("~/Desktop/Peralta_48well_WRC283M_WRC281MF.csv")
```

```
BacCP$CPMean_MF <- rowMeans(BacCP[,c("D2", "D4", "D6")])
BacCP$CPMean_M <- rowMeans(BacCP[,c("B2", "B4", "B6")])
```

```
dim(BacCP)
```

```
## [1] 49 52
```

```
BacCP_graph <- BacCP[c(1,51:52)]
```

```
BacCP_graph_long <- pivot_longer(BacCP_graph, cols=2:3, names_to = "Treatment", values_to = "meanGrowth")
```

```
str(BacCP_graph_long)
```

```
## tibble [98 x 3] (S3: tbl_df/tbl/data.frame)
## $ Time      : num [1:98] 0 0 0.5 0.5 1 1 1.5 1.5 2 2 ...
## $ Treatment : chr [1:98] "CPMean_MF" "CPMean_M" "CPMean_MF" "CPMean_M" ...
## $ meanGrowth: num [1:98] 0.141 0.141 0.143 0.142 0.144 ...
```

```
BacCP_graph_long$Treatment <- as.factor(BacCP_graph_long$Treatment)
```

```
summary(BacCP_graph)
```

```
##      Time      CPMean_MF      CPMean_M
## Min.   : 0      Min.   :0.1407      Min.   :0.1410
## 1st Qu.: 6      1st Qu.:0.1847      1st Qu.:0.1787
## Median :12      Median :0.3123      Median :0.3383
## Mean   :12      Mean   :0.2899      Mean   :0.2822
## 3rd Qu.:18      3rd Qu.:0.3917      3rd Qu.:0.3607
## Max.   :24      Max.   :0.3990      Max.   :0.3757
```

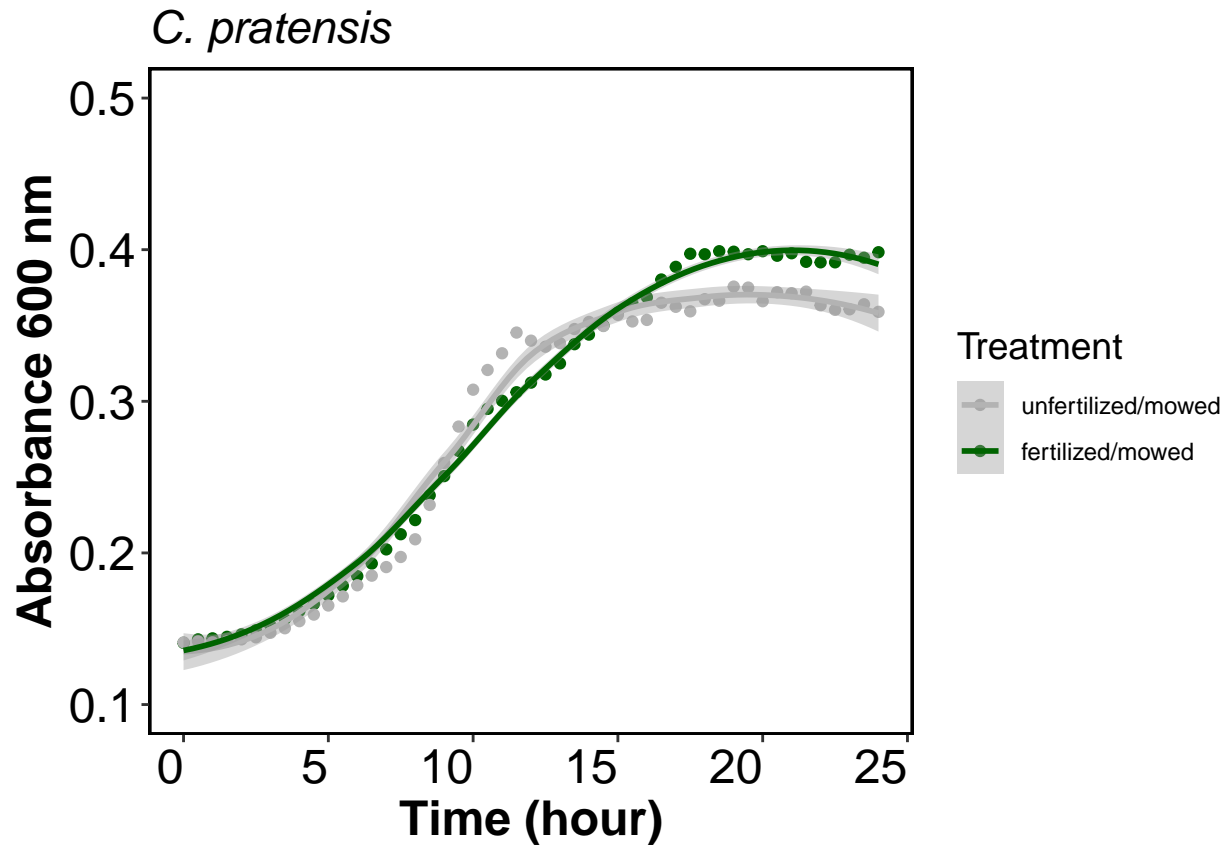
```
summary(BacCP_graph_long)
```

```
##      Time      Treatment    meanGrowth
##  Min.   : 0    CPMean_M :49    Min.     :0.1407
## 1st Qu.: 6    CPMean_MF:49    1st Qu.:0.1802
## Median :12                    Median  :0.3283
## Mean   :12                    Mean    :0.2861
## 3rd Qu.:18                    3rd Qu.:0.3658
## Max.   :24                    Max.    :0.3990
```

```
CP_graph <- ggplot(BacCP_graph_long, aes(x = Time, y = meanGrowth, color=Treatment)) +
  geom_point() + geom_smooth() + theme_bw() +
  scale_color_manual(name="Treatment", values=c("gray70", "darkgreen"), labels = c("unfertilized/mowed"
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_rect(colour = "black", fill=NA, size=1)) +
  theme(axis.text.x = element_text(size=18,colour = "black", hjust=1),
        axis.title.x = element_text(size=18,face="bold")) +
  theme(axis.text.y = element_text(size=18,colour = "black"),
        axis.title.y = element_text(size=18,face="bold")) +
  theme(strip.text.x = element_text(size = 18,face="bold"),strip.text.y = element_text(size = 18))+
  coord_cartesian(ylim = c(0.1,0.5))+
  ggtitle("C. pratensis") +
  theme(axis.text=element_blank(), title=element_text(size=rel(1.25))) +
  theme(plot.title = element_text(face = "italic")) +
  xlab("Time (hour)") + ylab("Absorbance 600 nm")
```

```
CP_graph
```

```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



```
str(BacCP_graph)
```

```
## 'data.frame':  49 obs. of  3 variables:
##  $ Time      : num  0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 ...
##  $ CPMean_MF: num  0.141 0.143 0.144 0.145 0.146 ...
##  $ CPMean_M  : num  0.141 0.142 0.141 0.142 0.143 ...
```

```
Bac_CP_M <- BacCP_graph[,2]
```

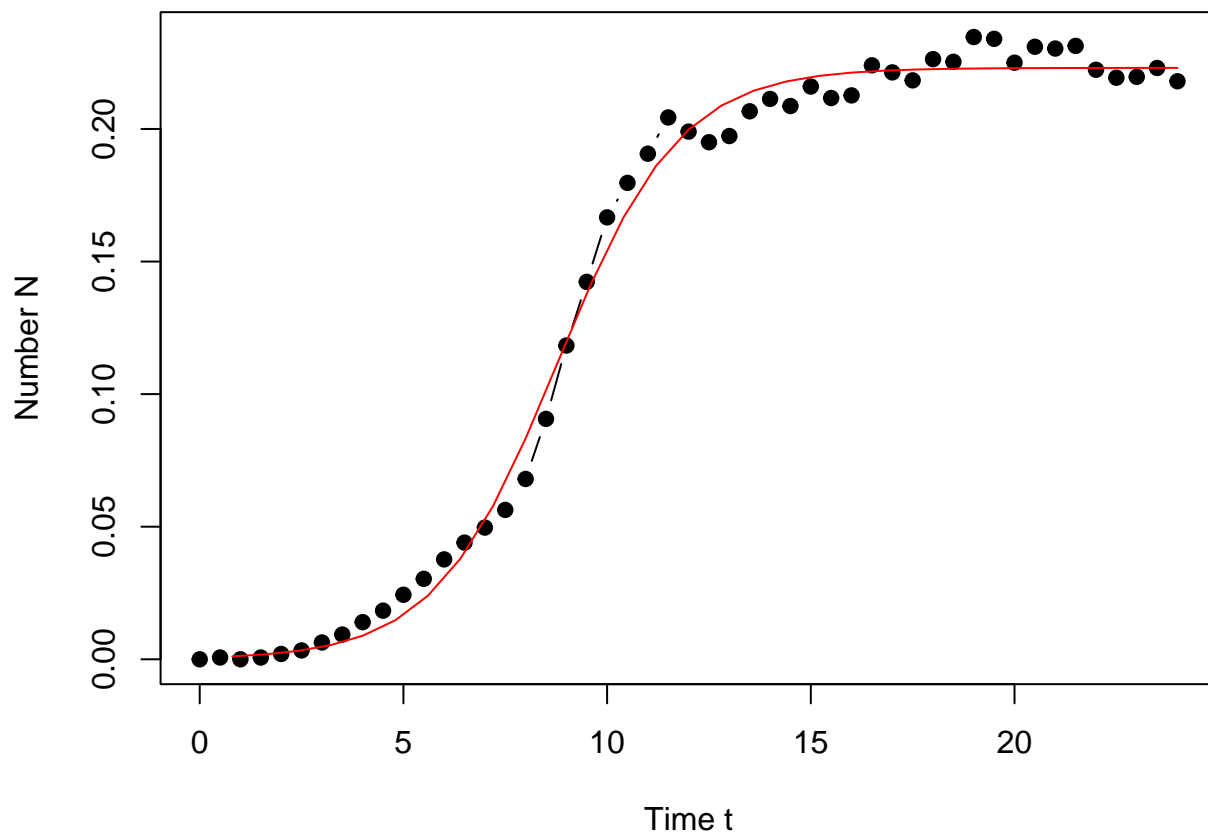
```
Bac_CP_MF <- BacCP_graph[,3]
```

```
GC_CP_Mfit <- SummarizeGrowth(Bac_CP_M$Time, Bac_CP_M$CPMean_M)
```

```
GC_CP_Mfit
```

```
## Fit data to K / (1 + ((K - NO) / NO) * exp(-r * t)):
##      K    NO  r
##  val: 0.223 0.001 0.667
##  Residual standard error: 0.007242805 on 46 degrees of freedom
##
## Other useful metrics:
##  DT 1 / DT  auc_l  auc_e
##  1.04  9.6e-01 3.39   3.41
```

```
plot(GC_CP_Mfit)
```

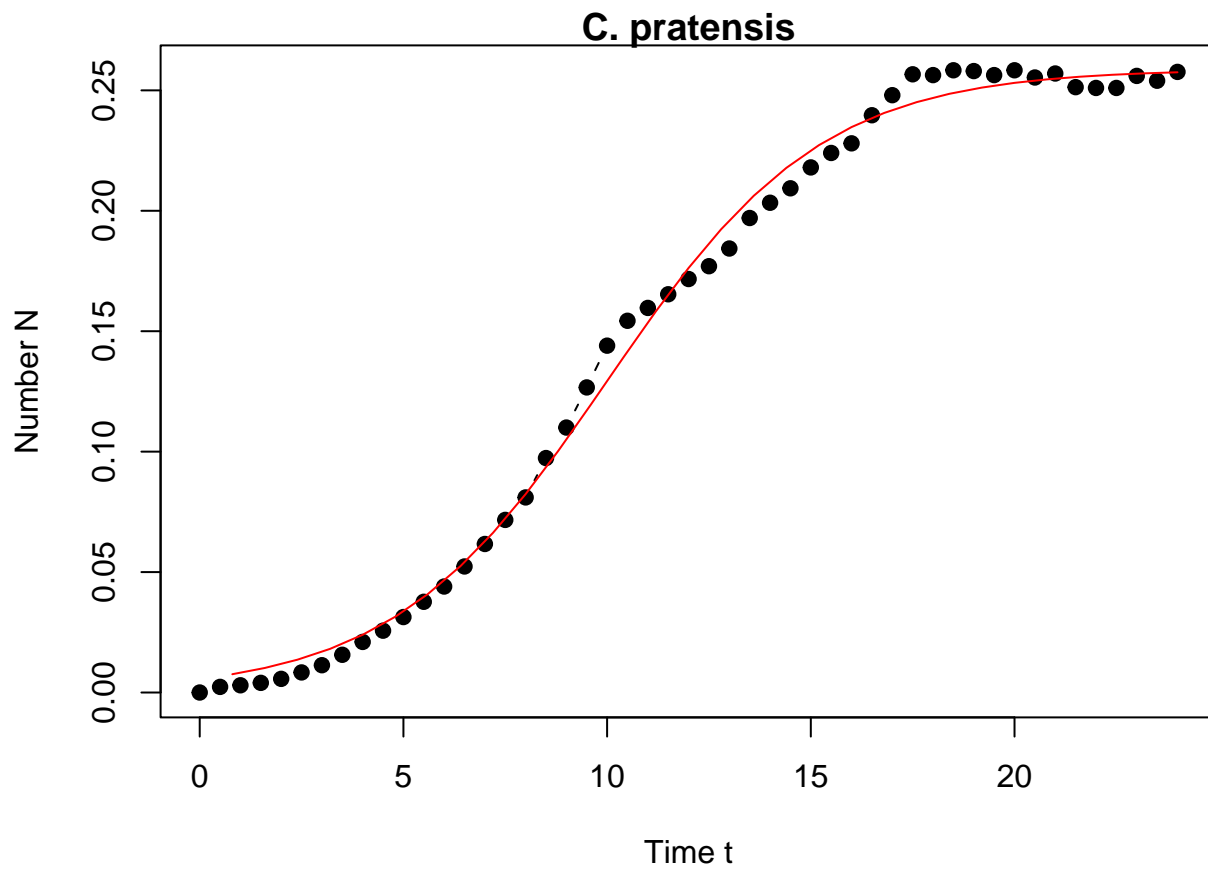


```
GC_CP_MFfit <- SummarizeGrowth(Bac_CP_MF$Time, Bac_CP_MF$CPMean_MF)
```

```
GC_CP_MFfit
```

```
## Fit data to  $K / (1 + ((K - N_0) / N_0) * \exp(-r * t))$ :
##      K   N0  r
## val: 0.259 0.006 0.38
## Residual standard error: 0.006581556 on 46 degrees of freedom
##
## Other useful metrics:
##   DT 1 / DT  auc_l  auc_e
##   1.82  5.5e-01 3.61   3.59
```

```
plot(GC_CP_MFfit, main="C. pratensis", main="Italics", col=c("red", "006600"))
```



```
GC_CP_MFfit$vals
```

```
## k      k_se      k_p n0  n0_se  n0_p
## 0.259    0.002    8e-60 0.006 0.001 3e-12
##
## r      r_se      r_p sigma  df  t_mid
## 0.38 0.011    8e-34 0.007 46 10.003
##
## t_gen    auc_l    auc_e
## 1.823    3.61    3.591
```

```
GC_CP_Mfit$vals
```

```
## k      k_se      k_p n0  n0_se  n0_p
## 0.223    0.002    8e-62 0.001 0 3e-04
##
## r      r_se      r_p sigma  df  t_mid
## 0.667    0.029    1e-26 0.007 46 8.776
##
## t_gen    auc_l    auc_e
## 1.04 3.395    3.405
```