Analyzing a set of growth data measured by plate reader

This is a documentation for how to read a large set of growth data and extract important physiological parameters.

For practice, below I show typical data of cell growth obtained from a plate reader. The first three columns show the number of cycle, time of measurement (second) and temperature. The rest of columns (A1 to A6) show OD600 readings at each well. Here the data are shwon until 10 cycles, but they actually continue up to 88 cycles.

library(knitr)  
# read csv file  
df <- read.csv("plate\_reader\_growth\_sample.csv", header = TRUE, check.names=FALSE)  
kable(df[1:10, 1:9], caption = "raw growth data")

raw growth data

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cycle Nr. | Time | Temp. | A1 | A2 | A3 | A4 | A5 | A6 |
| 1 | 0.000 | 37.0 | 0.0919 | 0.0862 | 0.0875 | 0.0886 | 0.0874 | 0.0918 |
| 2 | 899.077 | 37.0 | 0.0920 | 0.0860 | 0.0874 | 0.0884 | 0.0870 | 0.0915 |
| 3 | 1799.026 | 36.8 | 0.0907 | 0.0861 | 0.0875 | 0.0885 | 0.0872 | 0.0917 |
| 4 | 2698.979 | 37.1 | 0.0917 | 0.0863 | 0.0875 | 0.0885 | 0.0872 | 0.0919 |
| 5 | 3598.935 | 37.0 | 0.0913 | 0.0864 | 0.0877 | 0.0887 | 0.0873 | 0.0923 |
| 6 | 4498.889 | 37.1 | 0.0921 | 0.0866 | 0.0877 | 0.0889 | 0.0873 | 0.0926 |
| 7 | 5398.841 | 37.0 | 0.0925 | 0.0868 | 0.0878 | 0.0890 | 0.0875 | 0.0930 |
| 8 | 6298.796 | 36.8 | 0.0933 | 0.0870 | 0.0880 | 0.0891 | 0.0875 | 0.0936 |
| 9 | 7198.759 | 37.1 | 0.0940 | 0.0871 | 0.0880 | 0.0892 | 0.0875 | 0.0943 |
| 10 | 8098.712 | 37.1 | 0.0952 | 0.0875 | 0.0882 | 0.0893 | 0.0876 | 0.0951 |

Frist, I start with cleaning up for the raw data table in the follwoing order: 1) extract columns only for time & OD600 readings (required for producing growth curves) 2) change time from sec to hr  
3) extract background values.

After these process, the table looks like below.

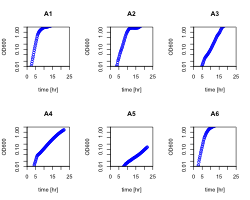
#remove cycle number and Temp  
df1 <- subset(df, select = -c(`Cycle Nr.`, Temp.))  
  
#change time from sec to hr  
df1$Time <- df1$Time/60/60   
  
#extract background and calibrate the vlues comparable to with 1cm cuvvete OD spec  
for (i in seq(2, ncol(df1))) {  
 df1[,i] <- (df1[,i]-df1[1,i])\*2.14 #substruct background OD and calibrate the values comparable to 1cm cuvvete OD spec  
 for (j in seq(1, nrow(df1))) {  
 if (df1[j,i] > 0) { #extract values only larger than 0 (cause problems to make log plots)  
 df1[j,i] <- df1[j,i]  
 }  
 else {  
 df1[j,i] <- NA  
 }  
 }  
}  
  
kable(df1[1:10, 1:7], caption = "raw growth data")

raw growth data

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time | A1 | A2 | A3 | A4 | A5 | A6 |
| 0.0000000 | NA | NA | NA | NA | NA | NA |
| 0.2497436 | 0.000214 | NA | NA | NA | NA | NA |
| 0.4997294 | NA | NA | NA | NA | NA | NA |
| 0.7497164 | NA | 0.000214 | NA | NA | NA | 0.000214 |
| 0.9997042 | NA | 0.000428 | 0.000428 | 0.000214 | NA | 0.001070 |
| 1.2496914 | 0.000428 | 0.000856 | 0.000428 | 0.000642 | NA | 0.001712 |
| 1.4996781 | 0.001284 | 0.001284 | 0.000642 | 0.000856 | 0.000214 | 0.002568 |
| 1.7496656 | 0.002996 | 0.001712 | 0.001070 | 0.001070 | 0.000214 | 0.003852 |
| 1.9996553 | 0.004494 | 0.001926 | 0.001070 | 0.001284 | 0.000214 | 0.005350 |
| 2.2496422 | 0.007062 | 0.002782 | 0.001498 | 0.001498 | 0.000428 | 0.007062 |

Next, based on the cleaned data table, produce growth curves as below.

# plot growth curves for all columns  
par(mfrow=c(2,3))  
par(new=FALSE)  
for (i in seq(2, ncol(df1))) {  
 plot(df1$Time, df1[,i], xlim=c(0,25), ylim=c(0.01,2), log="y", xaxs = "i", yaxs = "i" ,  
 xlab = "time [hr]", ylab = "OD600", cex.lab=1, cex.axis=1,  
 col="blue", cex=1, pch=1, main = colnames(df1[i]))   
 par(new=FALSE)  
}

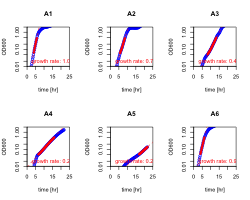


From my experience, the very high/low OD values have some uncertaintity. Below I extract reliable OD600 values only from 0.05 to 0.5.

df2<-df1  
for (i in seq(2, ncol(df1))) {  
 for (j in seq(1, nrow(df1))) {  
 if (df1[j,i] > 0.05 & df1[j,i] < 0.5 & is.na(df1[j,i])==FALSE) {  
 df2[j,i] <- df1[j,i]  
 }  
 else {  
 df2[j,i] <- NA  
 }  
 }  
}

Now overwrite these relaible OD readings onto the previous plots with red points and calculate growth rates in that range by exponential fitting.

# plot growth curves for all columns  
par(mfrow=c(2,3))  
par(new=FALSE)  
for (i in seq(2, ncol(df1))) {  
 plot(df1$Time, df1[,i], xlim=c(0,25), ylim=c(0.01,2), log="y", xaxs = "i", yaxs = "i" ,  
 xlab = "time [hr]", ylab = "OD600", cex.lab=1, cex.axis=1,  
 col="blue", cex=1, pch=1, main = colnames(df1[i]))   
 points(df2$Time, df2[,i], col="red", cex=1, pch=16)  
 abline(lm(log10(df2[,i]) ~ df2$Time), lty=2, col="red") # the base of line is log10  
 fit <- lm(log(df2[,i]) ~ df2$Time)  
 text(0, 0.02, paste(" growth rate:", fit$coefficients[2]), pos=4, col="red") # the base of slope is natural log  
 par(new=FALSE)  
}



The fitted growth rates (red lines) can be extracted as below.

output <- matrix(ncol=2, nrow=ncol(df1), head(c("plate number", "growth rate [hr-1]")))  
for (i in seq(2, ncol(df1))) {  
 fit <- lm(log(df2[,i]) ~ df2$Time)  
 output[i,1] <- colnames(df1[i])  
 output[i,2] <- fit$coefficients[2]  
}  
output <- data.frame(output)  
kable(output, caption = "raw growth data")

raw growth data

|  |  |
| --- | --- |
| X1 | X2 |
| plate number | growth rate [hr-1] |
| A1 | 1.09373879978018 |
| A2 | 0.724683516555484 |
| A3 | 0.446551698454079 |
| A4 | 0.250959998520243 |
| A5 | 0.20390634410275 |
| A6 | 0.958002642112856 |

Next, let’s think about how time derivative of OD600 (instataneous growth rate) changes. This is useful, for instance, to see at what OD600 cells cosume up nutrient and subsequently growth rate starts to decrease.

The instataneous growth rate is calcurated as below.

# first create NULL dataframe  
df\_iGR <- matrix(ncol=ncol(df1), nrow=nrow(df1))  
for (i in seq(2, ncol(df1))) {  
 for (j in seq(1, NROW(df1[,i]))) {  
 df\_iGR[j,i] <- (log(df1[j+1,i])-log(df1[j,i]))/(df1$Time[j+1]-df1$Time[j])  
 }  
}

Below are the line plots of how instataneous growth rate (iGR) changes over OD600.

par(mfrow=c(2,3))  
par(new=FALSE)  
for (i in seq(2, ncol(df1))) {  
 plot(df1[,i], df\_iGR[,i], ylim=c(0,1.2), xlim=c(0.01,2), xaxs = "i", yaxs = "i" , log = "x",  
 ylab = "iGR [hr-1]", xlab = "OD600", cex.lab=1.5, cex.axis=1.5,  
 col="blue", type = "l", main = colnames(df1[i]))   
 par(new=FALSE)  
}

