# Developing a model to describe bacterial growth

Nojoud Almohanna

11/05/2021

## Description

One important dynamical process in Microbiology is the growth of populations of organisms like bacteria. They reproduce through binary fission, which is a form of asexual reproduction, where single cells divide asexually into two cells, subsequently the two cells divide to form four cells, and so on. The time required for a cell to mature, and divide is approximately the same for any two cells. The increase in a bacterial population is described by a discrete dynamical system model. The goal of this project is to understand and analyze bacterial growth.

## Data

The Bacterial growth data below obtained from a V. natriegens (a marine bacterium) experiment which aimed to examine the growth rate of V. natriegens in a laboratory.The goal is to build discrete dynamical system models describing its population growth. For more details see https://mathinsight.org/bacteria_growth_initial_model.

The bacteria population size was recorded every 16 minutes.

```
# Load and read the dataset
bacden <- read.csv("~/Desktop/CS 510/MidtermProject/ Bacterial density data.csv")
head(bacden)
```

```
##   Time Population.Density
## 1    0               0.022
## 2   16               0.036
## 3   32               0.060
## 4   48               0.101
## 5   64               0.169
## 6   80               0.266
```

```
# Load the required packages
library(growthrates)
```

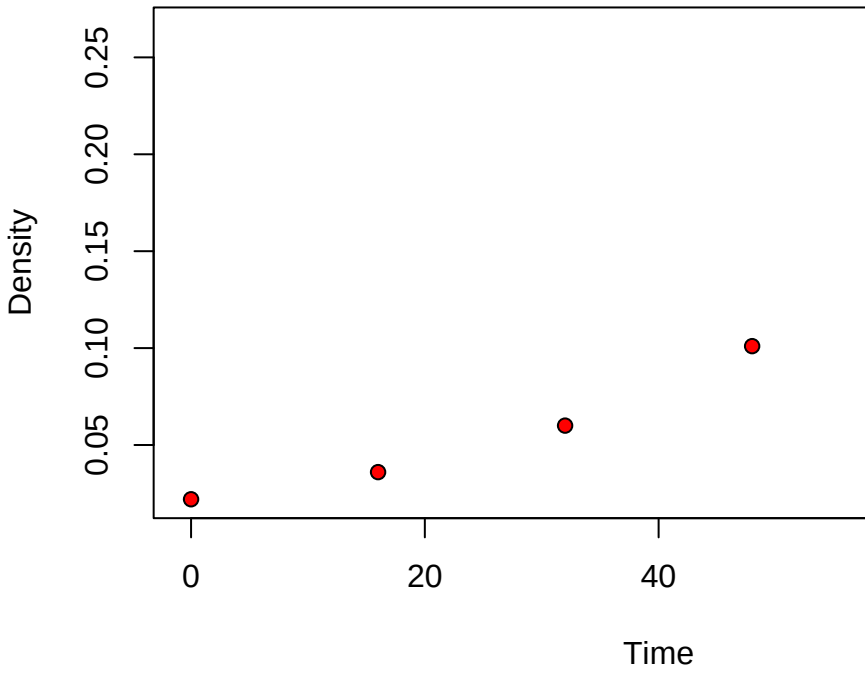
```
## Loading required package: lattice
```

```
## Loading required package: deSolve
```

```
library(ggplot2)
library(tibble)
library(biogrowth)
```

### Exploratory Data Analysis

```
plot(x = bacden$Time, y = bacden$Population.Density,col = 'black', bg = 'red', xlab="Time", ylab="Densi
```




### Visualizing population density versus time

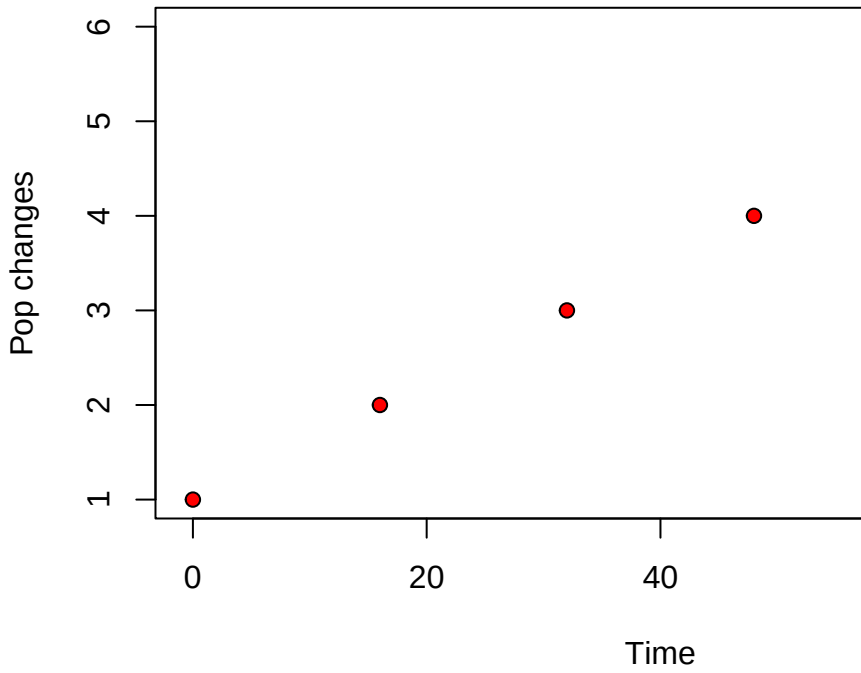
It's clear that the density is increasing faster as the time progresses.

```
counts = bacden$Population.Density
l = counts[-1]-counts[-length(counts)]
c <- round(l, 3)
c
```

### Calculating population changes

```
## [1] 0.014 0.024 0.041 0.068 0.097
```

```
plot(x=bacden$Time,y=c(1:length(bacden$Time)), col = 'black', bg = 'red', xlab="Time", ylab="Pop changes
```




**Visualizing population change versus time**

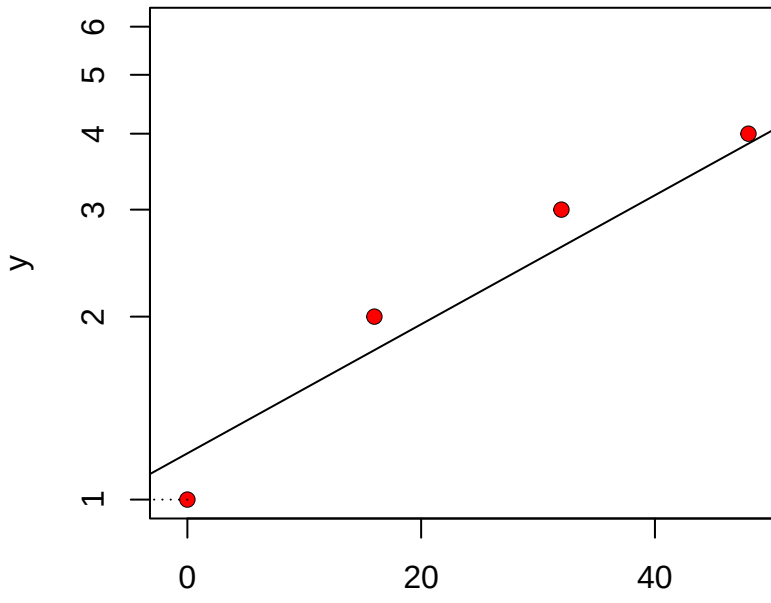
The population change increases as a function of time.

## Fitting a linear model to population change versus density

```
# Print out the summary
fit <- fit_easylinear(bacden$Time, c(1:length(bacden$Time)))
summary(fit)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##        1        2        3        4        5
## -0.17509  0.12685  0.14111  0.03759 -0.13047
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.175094   0.130458   1.342  0.27208
## x           0.024450   0.003329   7.345  0.00521 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1684 on 3 degrees of freedom
## Multiple R-squared:  0.9473, Adjusted R-squared:  0.9298
## F-statistic: 53.95 on 1 and 3 DF,  p-value: 0.005214
```

```
plot(fit, log = "y") #log-scale
```




**Plotting population change versus population size**

```
coef(fit) # exponential growth parameters
```

```
##          y0       y0_lm       mumax         lag
##  1.00000000  1.19135790  0.02445014 -7.16125635
```

```
deviance(fit)  # residual sum of squares
```

```
## [1] 0.08509659
```

```
rsquared(fit)  # coefficient of determination
```

```
##        r2
## 0.9473246
```

## Predicting the growth under dynamic conditions

In predictive microbiology mathematical models are usually divided in primary and secondary models.

- **Primary models** describe the variation of the population size against time.
- **Secondary models** describe the relationship between the parameters of the primary model and the environmental conditions.

The **biogrowth** package can be used for simulating growth under dynamic environmental conditions using **predict_dynamic_growth()** function. This function combines primary and secondary growth models. It has 7 arguments:

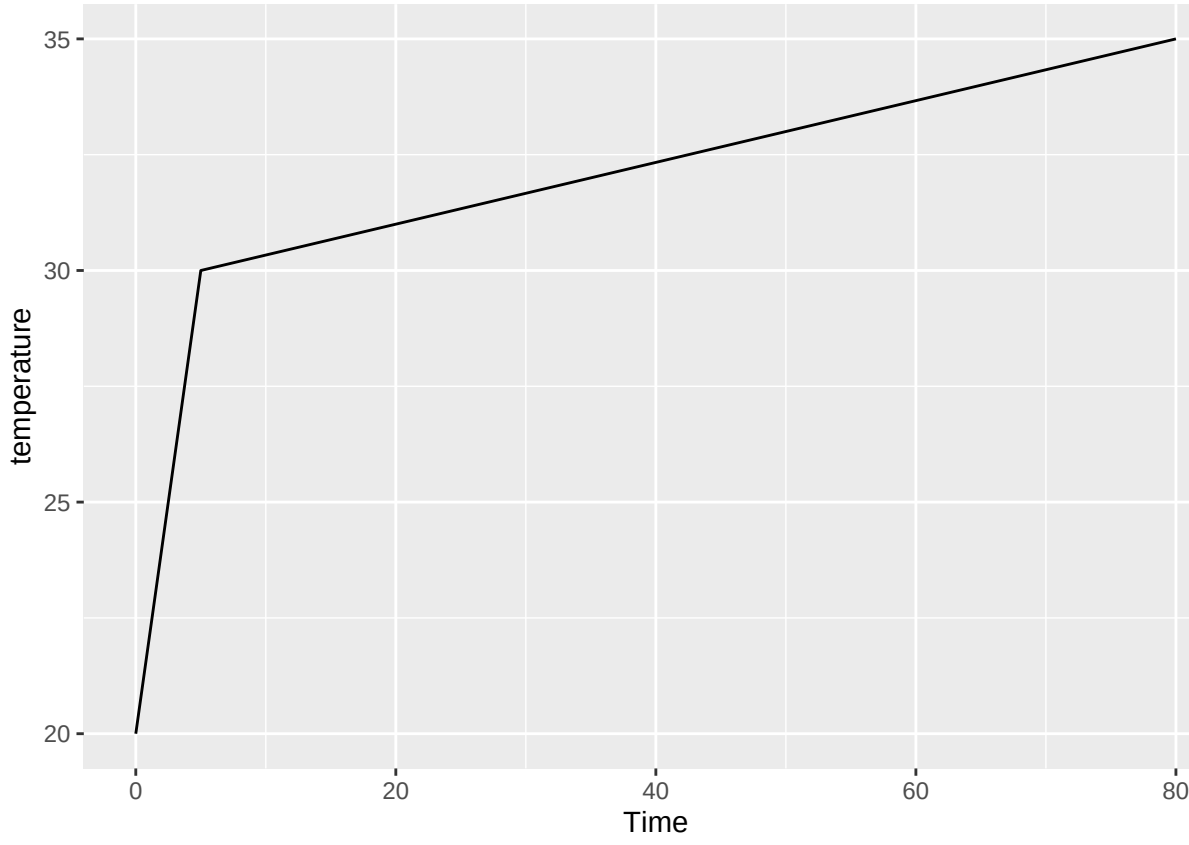
- times: Numeric vector of time points for the calculations.
- env_conditions: A tibble describing the variation of the environmental conditions.

- primary_pars: A named list describing the model parameters of the primary model.
- secondary_models: A nested list defining the secondary model(s).
- ...: Additional arguments passed to the numeric solver of the differential equation.
- check: Whether to do some validity checks of model parameters (TRUE by default).
- formula: A one-sided formula describing the x variable in env_conditions.

I will consider two environmental factors: *temperature* and *pH*, defined using a tibble.

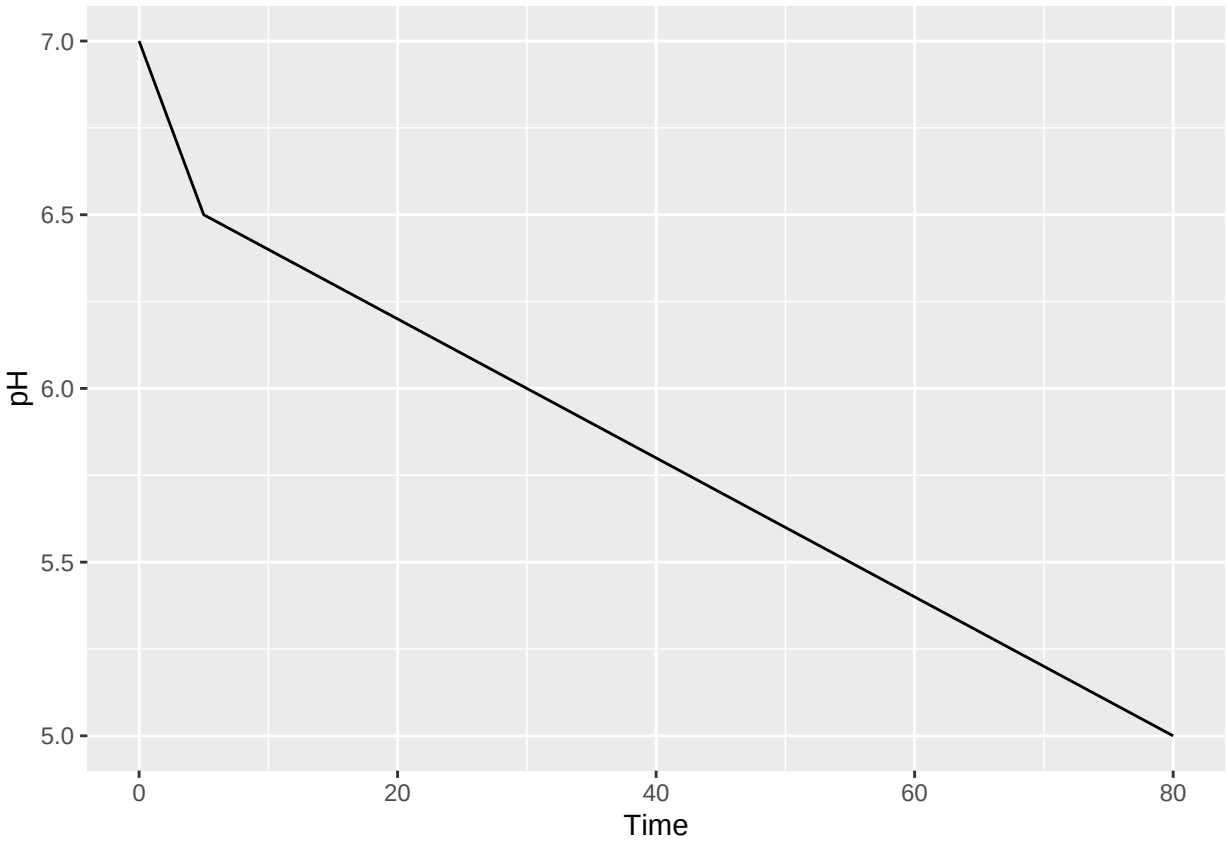
```
my_conditions <- tibble(Time=c(0, 5, 80),
                         temperature = c(20, 30, 35),
                        pH = c(7, 6.5, 5))
```

```
ggplot(my_conditions) + geom_line(aes(x = Time, y = temperature))
```




**Temperature plot**

```
ggplot(my_conditions) + geom_line(aes(x = Time, y = pH))
```




**PH plot**

```
# For dynamic conditions,biogrowth uses the Baranyi growth model as primary model.

my_primary <- list(mu_opt = .9, # the maximum specific growth rate
             Nmax = 1e8, # the maximum growth rate
             N0 = 0.022, # the initial population size
             Q0 = 1e-3)
```

```
sec_temperature <- list(model = "Zwietering",
                        xmin = 25,
                        xopt = 35,
                        n = 1)
```

```
sec_pH <- list(model = "CPM",
               xmin = 5.5,
               xopt = 6.5,
               xmax = 7.5,
               n = 2)
```

```
my_secondary <- list(
    temperature = sec_temperature,
    pH = sec_pH
    )
```

```
my_times <- seq(0, 50, length = 1000)
```

```
dynamic_prediction <- predict_dynamic_growth(my_times,
                        my_conditions, my_primary,
                        my_secondary,
                        formula = . ~ Time)
```

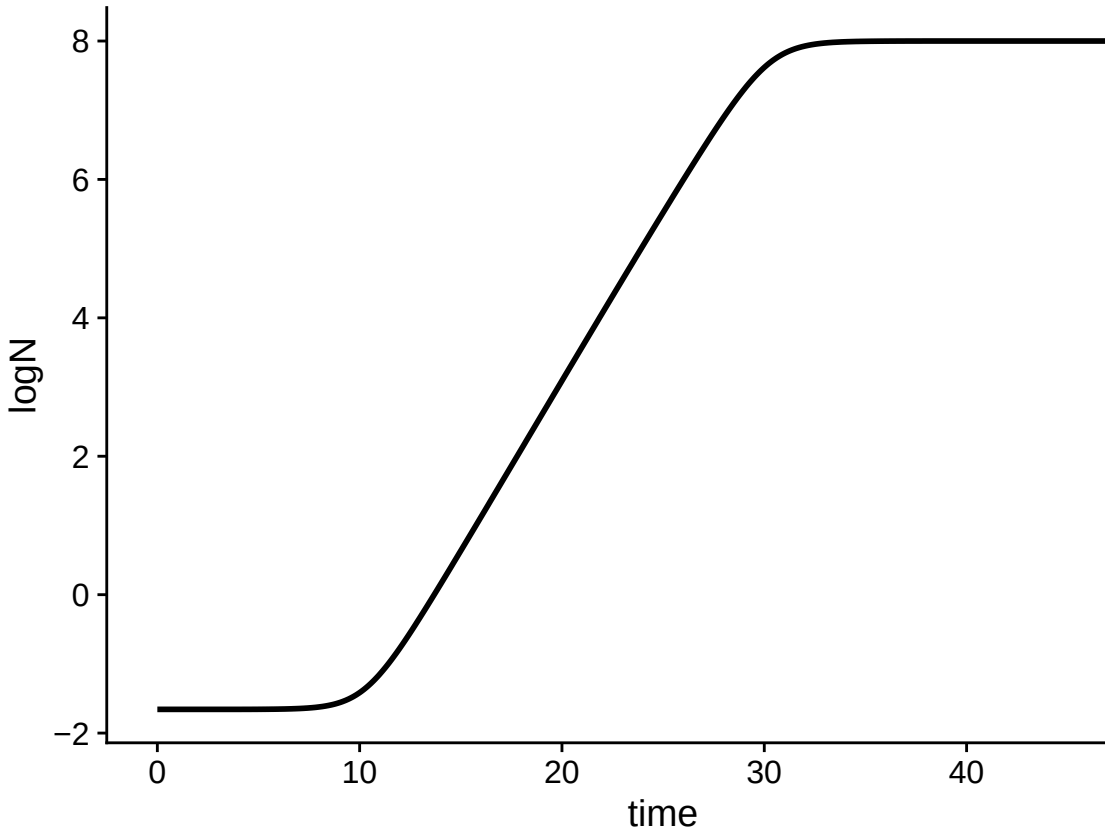
**Define model parameters**

```
# The results of the simulation
dynamic_prediction$simulation
```

**Simulation**

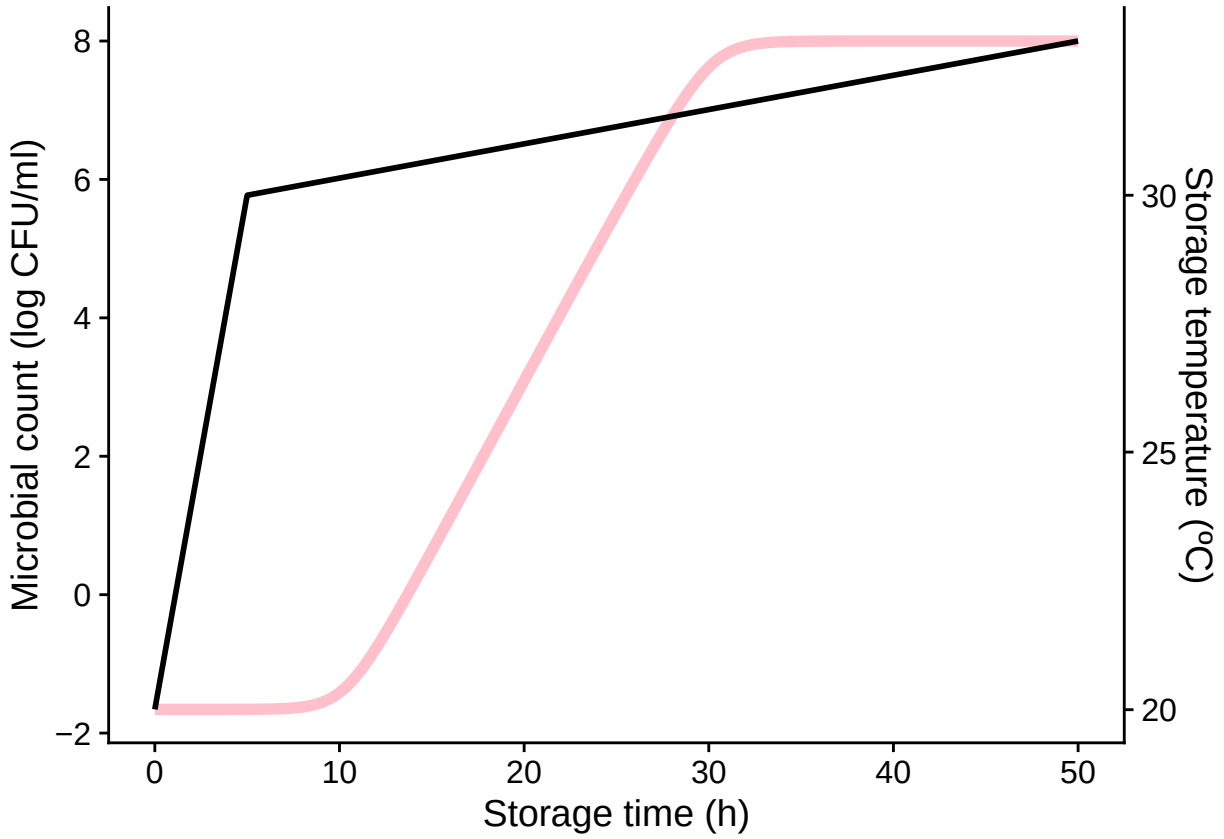
```
## # A tibble: 1,000 x 4
##      time     Q     N  logN
##     <dbl> <dbl> <dbl> <dbl>
##  1 0      0.001 0.022 -1.66
##  2 0.0501 0.001 0.022 -1.66
##  3 0.100  0.001 0.022 -1.66
##  4 0.150  0.001 0.022 -1.66
##  5 0.200  0.001 0.022 -1.66
##  6 0.250  0.001 0.022 -1.66
##  7 0.300  0.001 0.022 -1.66
##  8 0.350  0.001 0.022 -1.66
##  9 0.400  0.001 0.022 -1.66
## 10 0.450  0.001 0.022 -1.66
## # ... with 990 more rows
```

```
plot(dynamic_prediction)
```




**Visualize the simulation**

```
# plot some environmental factor with add_factor
plot(dynamic_prediction,
     add_factor = "temperature",
     label_y1 = "Microbial count (log CFU/ml)",
     label_y2 = "Storage temperature (ºC)",
     line_col = "pink",
     line_size = 2, line_type2 = 1) + xlab("Storage time (h)")
```




```
# the time required to reach 6 log CFU/ml in the dynamic prediction
time_to_logcount(dynamic_prediction, 6)
```

**Predicting the time to reach a given population size**

```
## [1] 26.01065
```