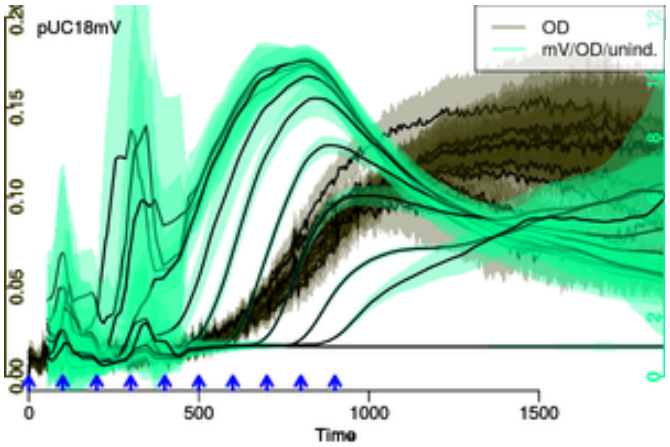
From Wells to Cells

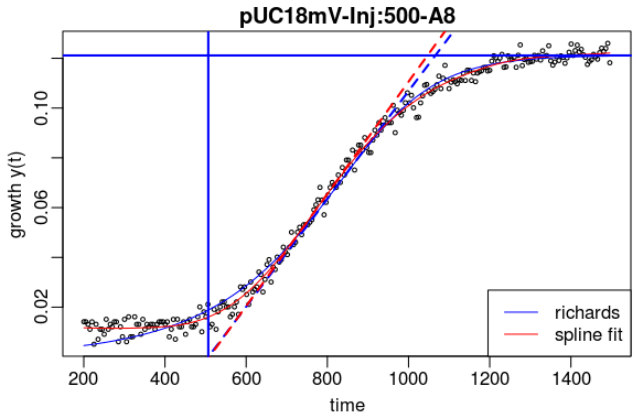
Growth vs. Gene Expression in Platereader Experiments

Rainer Machne

Plan

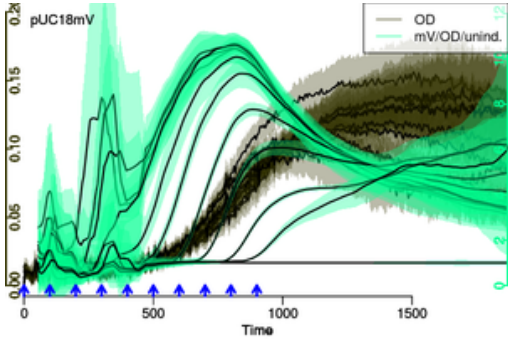
#	EXPERIMENT	THEORY
1	Using R packages: `platexpress`, `grofit`, etc.	The Basics: What is Exponential Growth?
2	The Experiment: View and Analyze Data	Statistics: Measurement Errors
3	The Result: Summary Plots	The Model: Growth vs. Expression Rates

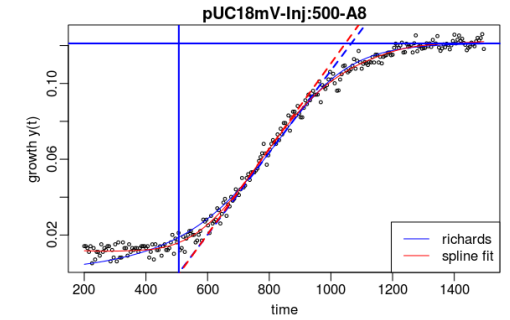


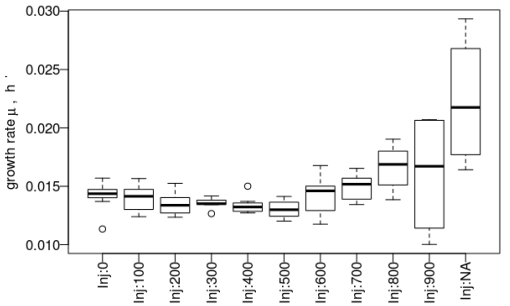


https://github.com/raim/platexpress

Goals: learn some R for data analysis - handling large data sets, &
gain better quantitative understanding of cell growth *VS.* gene regulation.







1. Use R to Analyze Growth & Expression Data
   - Installing R packages
   - Prepare & load data
2. Use R `base`
   - Linear regression with `lm(log(X) ~ time)`
   - Non-linear fit with `nls(X ~ X0*exp(mu*time))`
3. Use R packages
   - `grofit`, `growthcurver` or `growthrate`
   - Logistic, Gompertz, non-parametric growth models
4. Growth vs. Gene Expression
   - **Proteins/Cell: normalize fluorescence & compare**
   - Monod equation: growth vs. gene expression
   - **Towards the riboswitch model**

From Data Hell to Model Heaven : Why Platereader?

Tube vs. Platereader Cultures

1. Data Analysis
   - many replicates, many conditions
   - different statistics required
2. Endpoint vs. Kinetics
   - steady-state vs. dynamics

Are tube and platereader cultures comparable?
Why (not)?

From Data Hell to Model Heaven and back.

1. Inspect Data Files
   - Amend if required
2. Load & Inspect Data
   - Get overview, check data: roughly Ok?
   - Clean data: subtract blanks, cut time, rm outliers etc.
   - Inspect replicates: are observations replicated, what are our measurement errors?
3. Construct Your Analysis Pipeline
   - What do we need?
   - What were our assumptions, and are they justified?
   - Do we see additional unsuspected results?
4. Calculate Results & Statistics over Replicates

Do we have what we need for Monday?

Installing R Packages from `cran`, `bioconductor` & `github`

```
install.packages(c("grofit","growthcurver")) # at CRAN

source("https://bioconductor.org/biocLite.R") # at bioconductor
biocLite("cellGrowth")

install.packages("devtools") # R development tools
library(devtools)
install_github("raim/platexpress") # at github
```

https://github.com/raim/platexpress

Do you speak `git`? Try to install from your local copy:

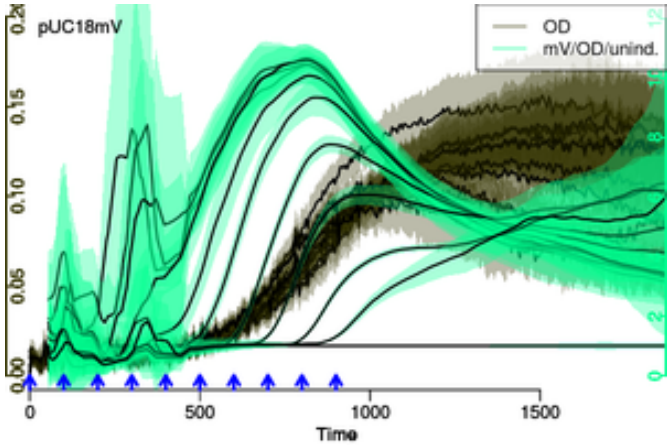
`git clone git@github.com:raim/platexpress.git`

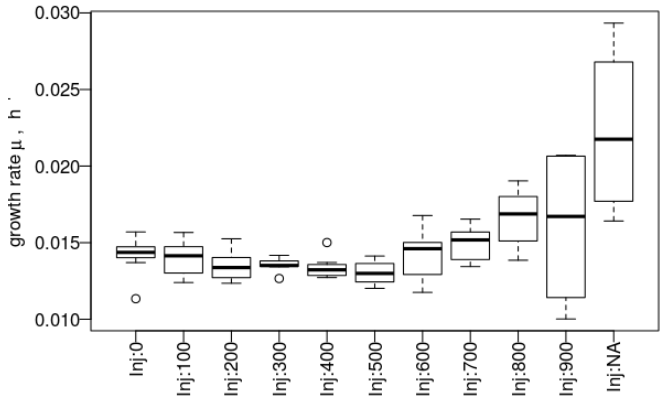
... and use `R CMD build` & `sudo R CMD INSTALL`

How to use a new R package?

```
## load the package & explore
library(platexpress)

?platexpress # VIEW HELP FILES
vignette("platexpress") # READ THE VIGNETTE
demo("demo_ap12") # RUN THE DEMO
getData # SEE WHAT A FUNCTION DOES: just type without brackets

## APPLY TO YOUR DATA:
setwd("~/work/hhu_2015/uebung_201612/cellgrowth_20161214/praktikum_201612/test2/")
plate <- readPlateMap("161130_Praktikum_RAJ11_Test_2_layout.csv")
files <- c("161130_Praktikum_RAJ11_Test_2.csv")
raw <- readPlateData(files, type="Synergy",skip=44,sep=";",
                     time.format="%H:%M:%S",time.conversion=1/3600)
viewPlate(raw)
```

Why Growth Rates?

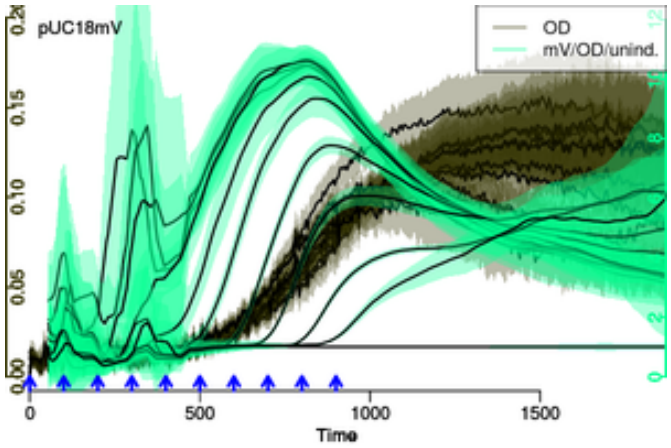


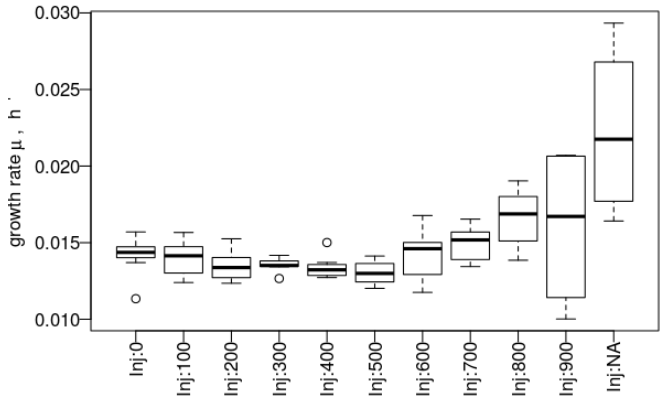


- Injection of IPTG into one column (7 wells + 1 blank) every 100 minutes.
- Slower response with later injections.

$\Rightarrow$ Faster growth rate μ with later or no induction.

Why Growth Rates?





- Injection of IPTG into one column (7 wells + 1 blank) every 100 minutes.
- Slower response with later injections.

$\Rightarrow$ Faster growth rate μ with later or no induction.

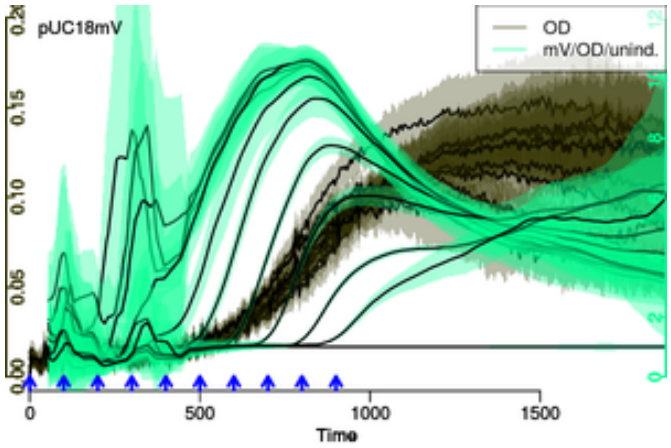
$$\mu = k \frac{\text{ribosomes}}{\text{proteins}}$$

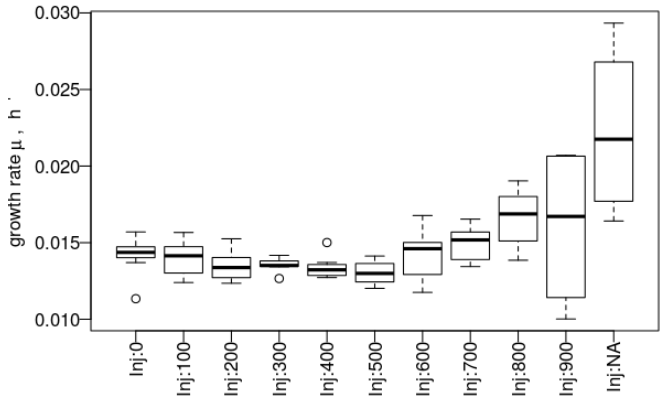
Schaechter, Maaloe & Kjeldgaard, J Gen Microbiol **1958**: *Dependency on medium and temperature of cell size and chemical composition during **balanced growth** of Salmonella typhimurium.*

Koch, Can J Microbiol 1988: ***Why can't a cell grow infinitely fast?***

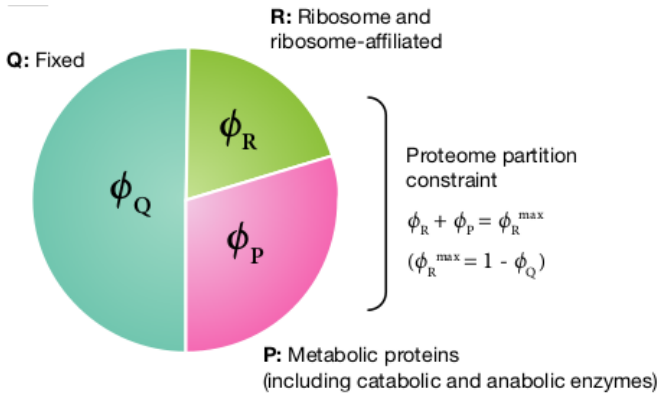
Neidhardt, J Bacteriol 1999: *Bacterial growth: constant obsession with dN/dt.*

Why Growth Rates?





$$\mu = k \frac{\text{ribosomes}}{\text{proteins}}$$

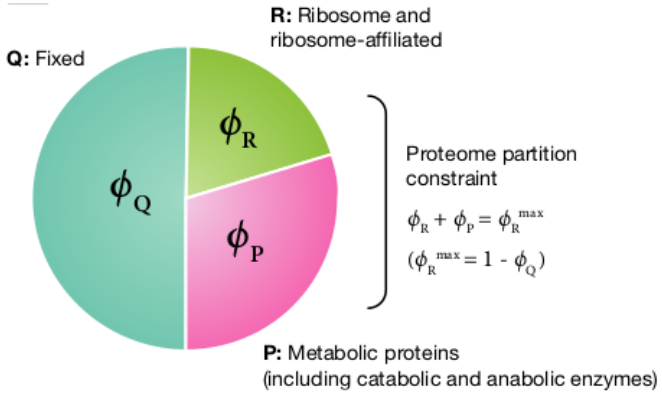


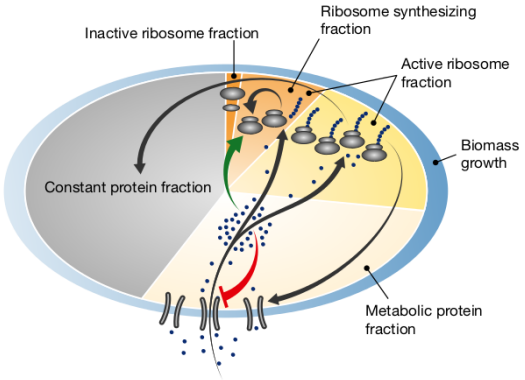
Brauer *et al.*, Mol Biol Cell 2008: *Coordination of growth rate, cell cycle, stress response, and metabolic activity in yeast.*

Slavov *et al.*, Mol Biol Cell 2011: *Coupling among growth rate response, metabolic cycle, and cell division cycle in yeast.*

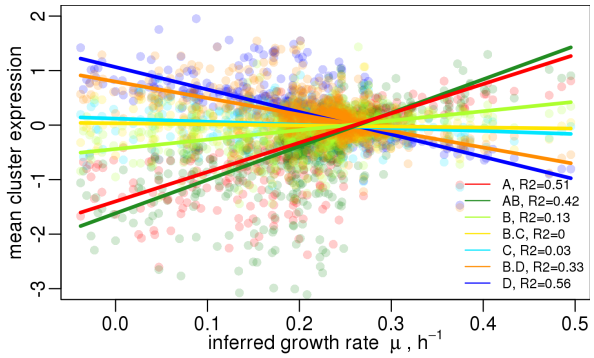
Scott *et al.*, Science 2010: *Interdependence of cell growth and gene expression: origins and consequences.*

Weiße *et al.*, PNAS 2015: *Mechanistic links between cellular trade-offs, gene expression, and growth.*

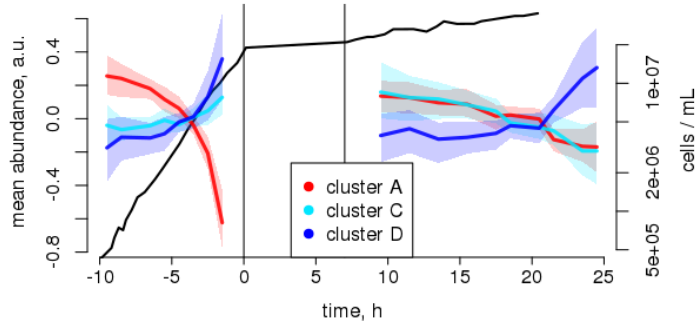




Scott *et al.*, Mol Syst Biol 2014: *Emergence of robust growth laws from optimal regulation of ribosome synthesis.*



Expression of large gene groups correlates with μ.

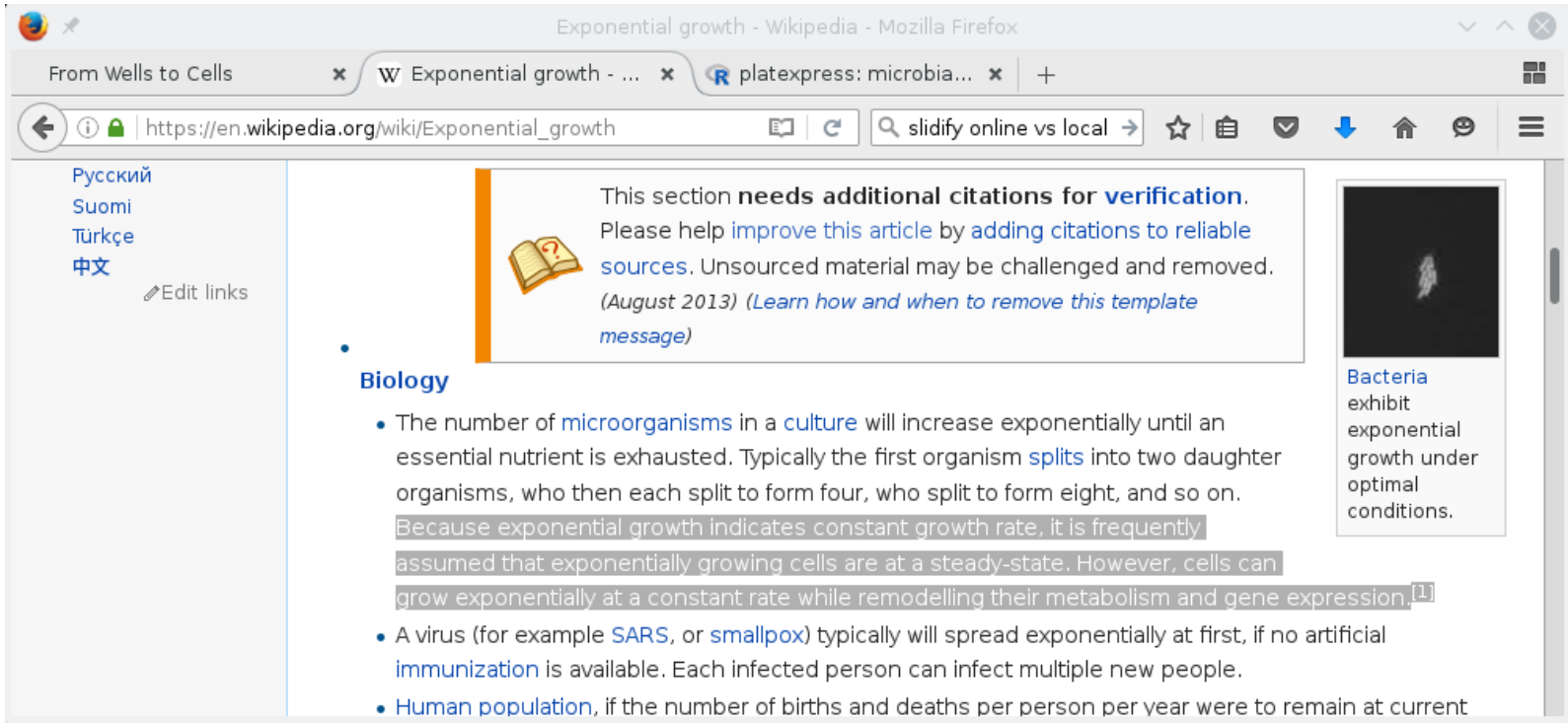


Even at constant μ cells are not in steady-state!

Slavov *et al.*, Cell Rep 2014: *Constant growth rate can be supported by decreasing energy flux and increasing aerobic glycolysis.*

The concept of "balanced growth" is flawed, yet a central assumption of many quantitative models.

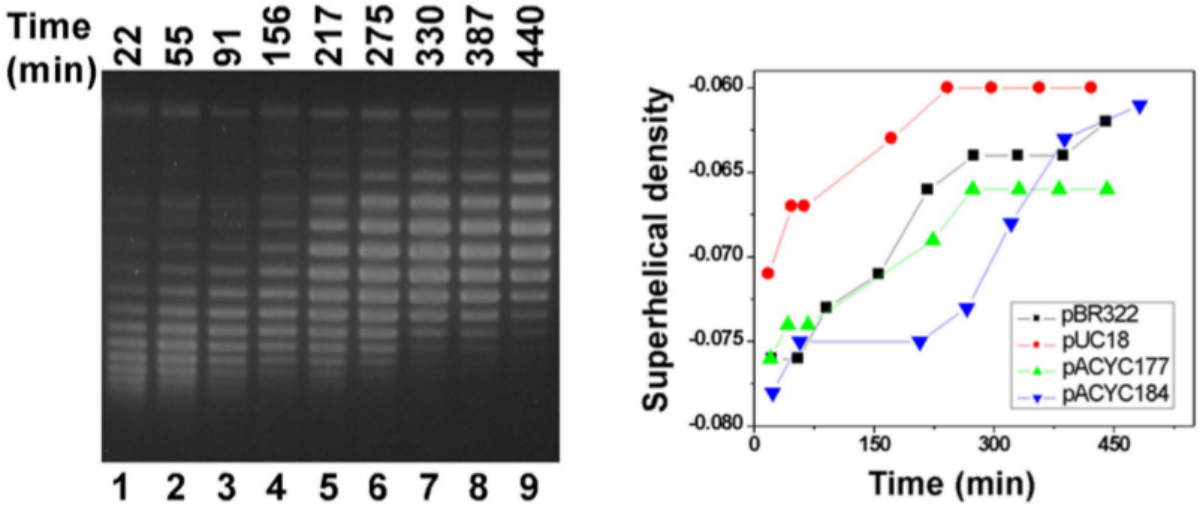
Cited at Wikipedia:



. . that's how important it is (or perhaps it was his dad :p).

[1] Slavov *et al.*, Cell Rep 2014: *Constant growth rate can be supported by decreasing energy flux and increasing aerobic glycolysis.*

DNA Supercoiling Varies during Exponential Phase (constant μ).

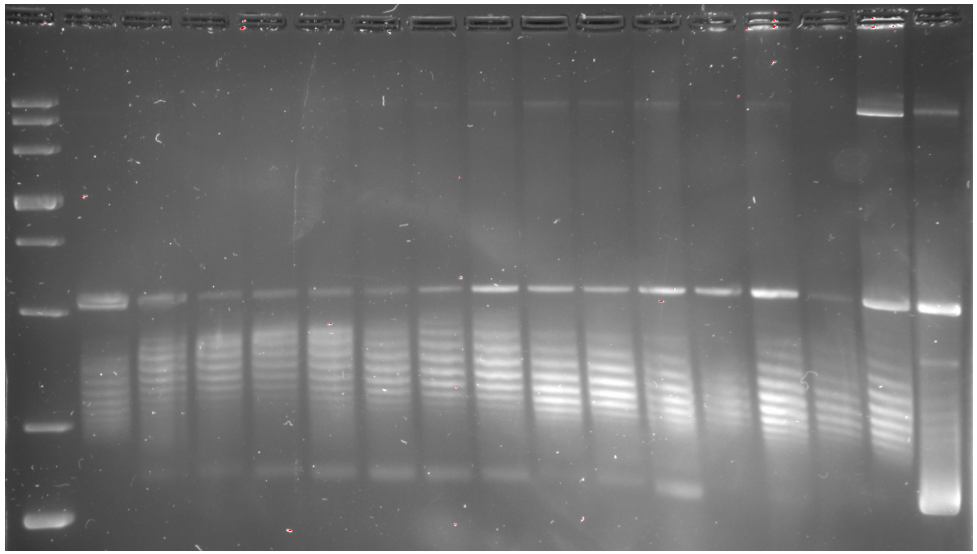


Interested in a bachelor/master project?

- Working on `platexpress` and/or quantitative models of growth vs. gene expression.
- At the interface of three projects: Ribonets, Coil-seq & Yeast Oscillations.

Fulcrand *et al.*, Sci Rep 2016: *DNA supercoiling, a critical signal regulating the basal expression of the lac operon in Escherichia coli.*

DNA Supercoiling Varies during Exponential Phase (constant μ).



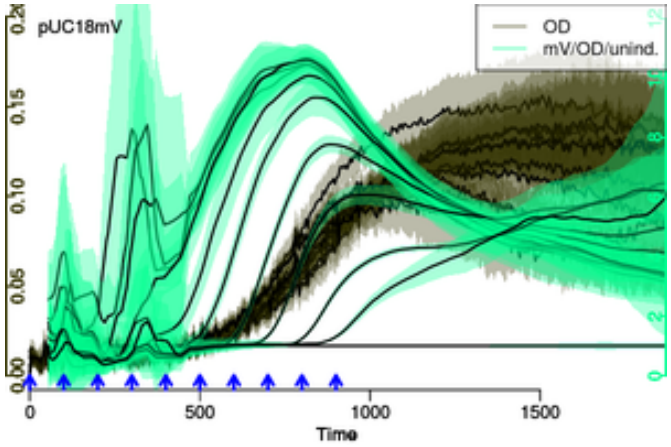
⇐ © Kai, today

Interested in a bachelor/master project?

- Working on `platexpress` and/or quantitative models of growth vs. gene expression.
- At the interface of three projects: Ribonets, Coil-seq & Yeast Oscillations.

Fulcrand *et al.*, Sci Rep 2016: *DNA supercoiling, a critical signal regulating the basal expression of the lac operon in Escherichia coli.*

Growth & Gene Expression in *E. coli* : exponential growth

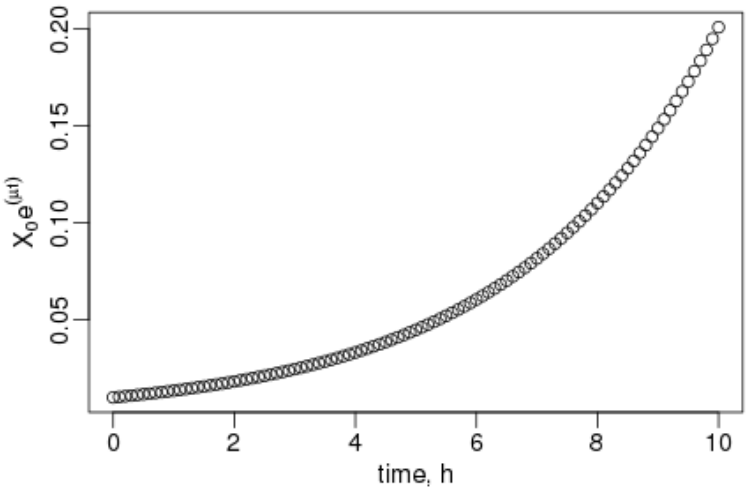


$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \mu X(t)$$

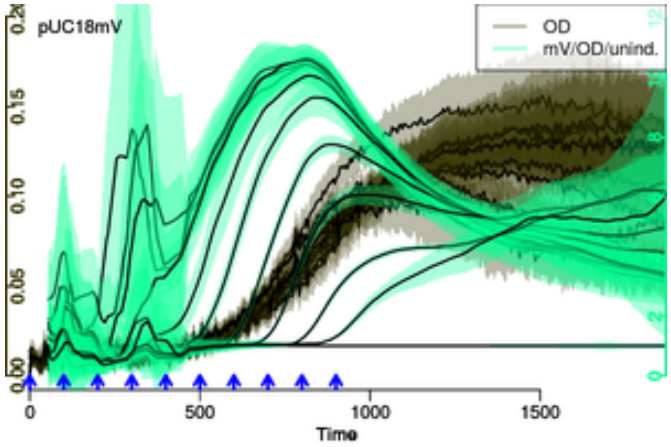
$$X(t) = X(0)e^{\mu t}$$

Can you derive the formula for exponential growth?

```
time <- seq(0,10,0.1) # hours
mu <- 0.3 # specific growth rate, hour^-1
x0 <- 0.01 # the inoculum: cell density, cells liter^-1
xt <- x0 * exp(mu*time)
par(mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(time, xt,
     xlab="time, h",ylab=expression(X[0]*e^(mu*t)))
```



Growth & Gene Expression in *E. coli* : growth rate



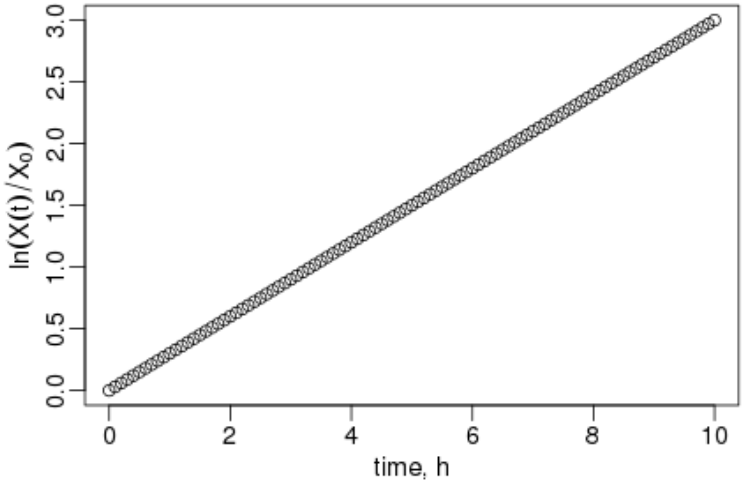
$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \mu X(t)$$

$$X(t) = X(0)e^{\mu t}$$

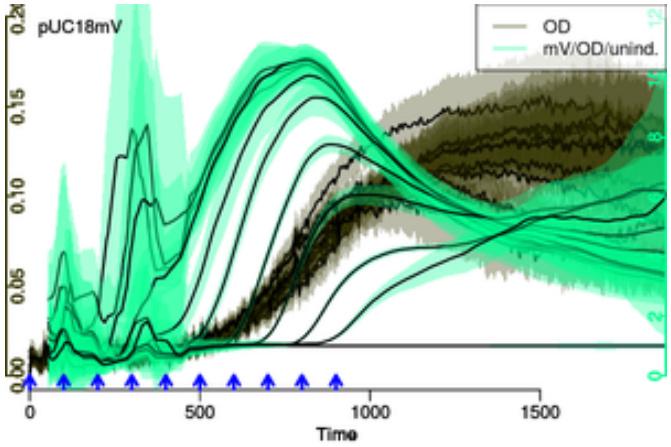
$$\ln \frac{X(t)}{X(0)} = \mu t$$

Which important parameter can we calculate from here?

```
time <- seq(0,10,0.1) # hours
mu <- 0.3 # specific growth rate, hour^-1
x0 <- 0.01 # the inoculum: cell density, cells liter^-1
xt <- x0 * exp(mu*time)
par(mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(time, log(xt/x0),
     xlab="time, h",ylab=expression(ln(X(t)/X[0])))
```



Growth & Gene Expression in *E. coli* : doubling time



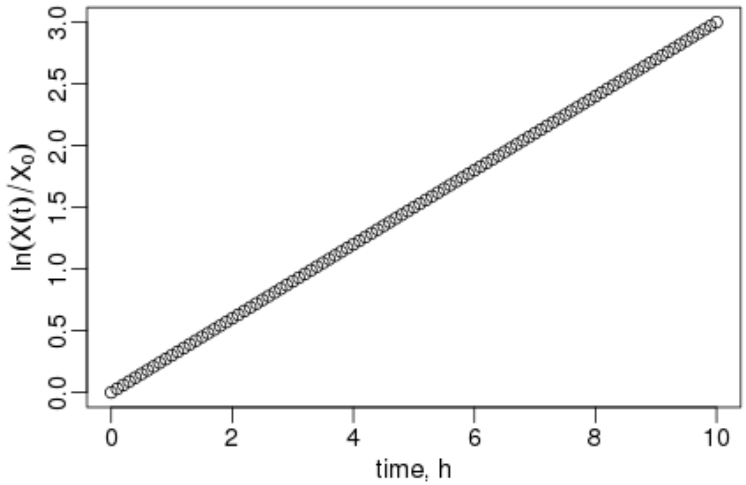
$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \mu X(t)$$

$$X(t) = X(0)e^{\mu t}$$

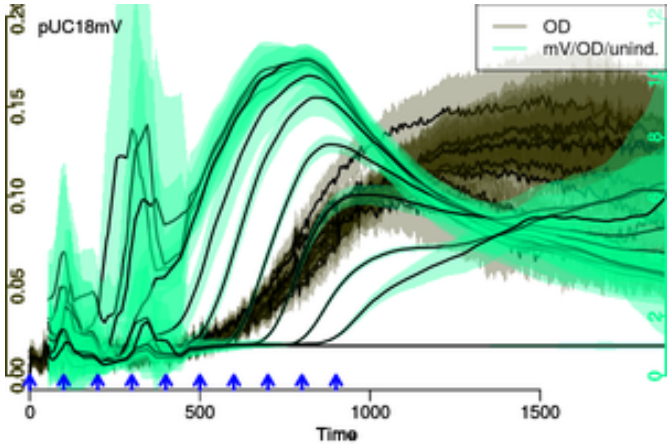
$$\frac{\ln 2}{\mu} = t_D$$

t_D is the average CULTURE doubling time.
In which cases is it also the average CELL doubling time?

```
time <- seq(0,10,0.1) # hours
mu <- 0.3 # specific growth rate, hour^-1
x0 <- 0.01 # the inoculum: cell density, cells liter^-1
xt <- x0 * exp(mu*time)
par(mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(time, log(xt/x0),
     xlab="time, h",ylab=expression(ln(X(t)/X[0])))
```



Growth & Gene Expression in *E. coli* : growth rate

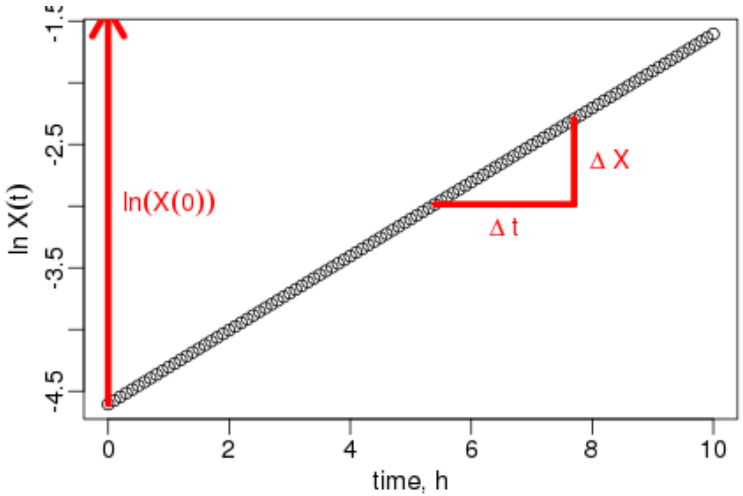



```
par(mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(time, log(xt), xlab="time, h",ylab=expression(ln~X(t)))
x1 <- .05; idx1 <- which(abs(xt-x1)==min(abs(xt-x1)))
x2 <- .1; idx2 <- which(abs(xt-x2)==min(abs(xt-x2)))
lines(x=time[c(idx1,idx2)], y=log(xt[c(idx1,idx1)]),col=2,lwd=5)
text(time[idx2],mean(log(xt[c(idx1,idx2)])),expression(Delta~X),pos=4,col=2)
lines(x=time[c(idx2,idx2)], y=log(xt[c(idx1,idx2)]),col=2,lwd=5)
text(mean(time[c(idx1,idx2)]),log(xt[idx1]),expression(Delta~t),pos=1,col=2)
arrows(x0=0,x1=0,y0=log(xt[1]),y1=-1.4,col=2,lwd=5);
text(x=0,y=-3,expression(ln(X(0))),pos=4,col=2)
```

$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \mu X(t)$$

$$X(t) = X(0)e^{\mu t}$$

$$\ln(X(t)) = \mu t + \ln(X(0))$$



Load Your Data

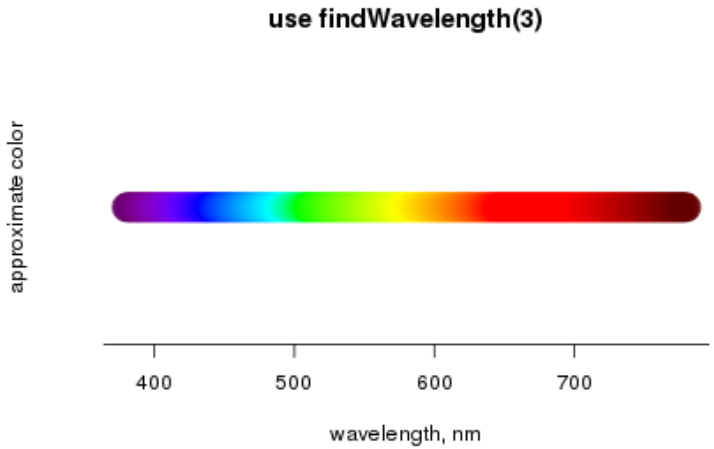
```
library(platexpress)

dpath <- "~/work/hhu_2015/uebung_201612/Praktikum-M4452_20161207/ecoli_ts_20161014"

plate <-readPlateMap(file.path(dpath,"20161014_platemap.csv"), fsep=";",
                     fields=c("strain","IPTG","blank"))
files <- c("20161014_20161014 IPTG mVenus Injection  1_Absorbance.CSV",
           "20161014_20161014 IPTG mVenus Injection  1_Fluorescence.CSV")
raw <- readPlateData(file.path(dpath,files), type="BMG", time.conversion=1/60)
```

```
## Parsing file ~/work/hhu_2015/uebung_201612/Praktikum-M4452_20161207/ecoli_ts_20161014/20161014_20161014 IPTG mVenus Injection
##  found data 584
## Parsing file ~/work/hhu_2015/uebung_201612/Praktikum-M4452_20161207/ecoli_ts_20161014/20161014_20161014 IPTG mVenus Injection
##  found data 485/Em520
## Interpolating all data to a single master time.
```

```
## Warning in listAverage(data, "time"): time : max. SD within timepoint is 70.5 % of median difference between time points.
```

What does the `warning` mean?

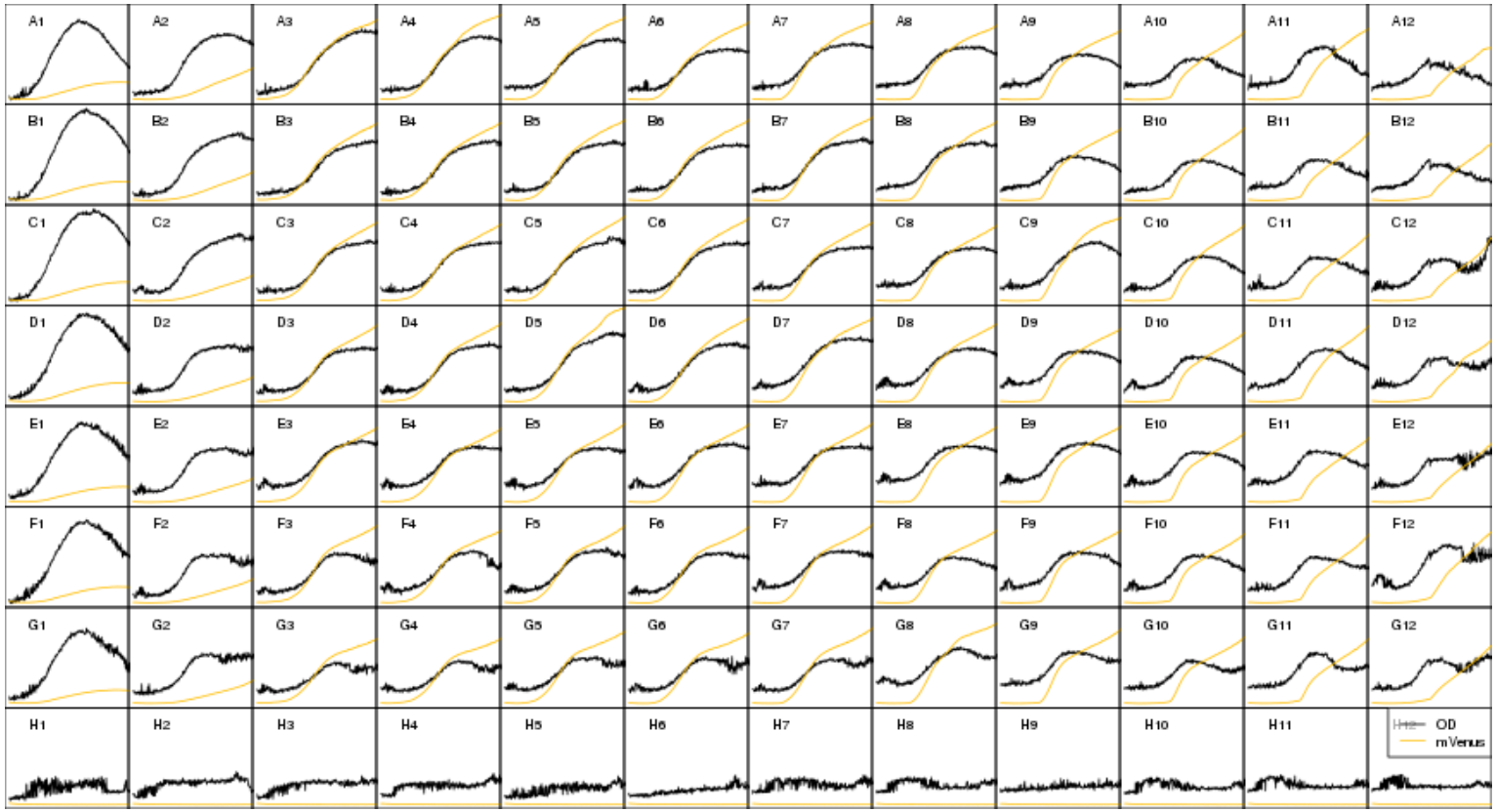
```
showSpectrum() # try: findWavelength(3)
```




```
## re-name and color data
raw <- prettyData(raw, dids=c(OD="584",mVenus="485/Em520"),
                  colors=c("#000000",wavelength2RGB(600)))
```

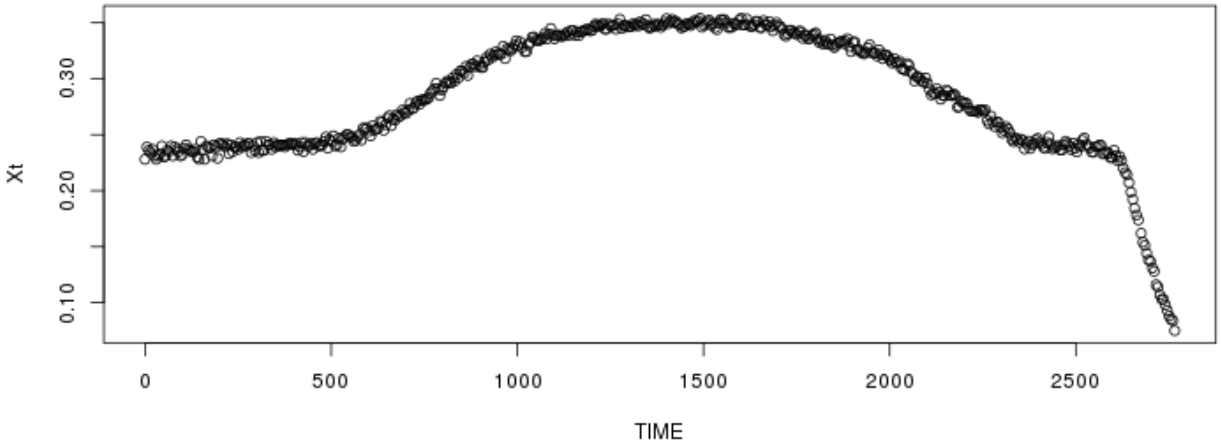
```
vp <- viewPlate(raw,xlim=c(0,1800),xscale=TRUE)
```

```
## x-axis: Time
## plotting OD;mVenus
```

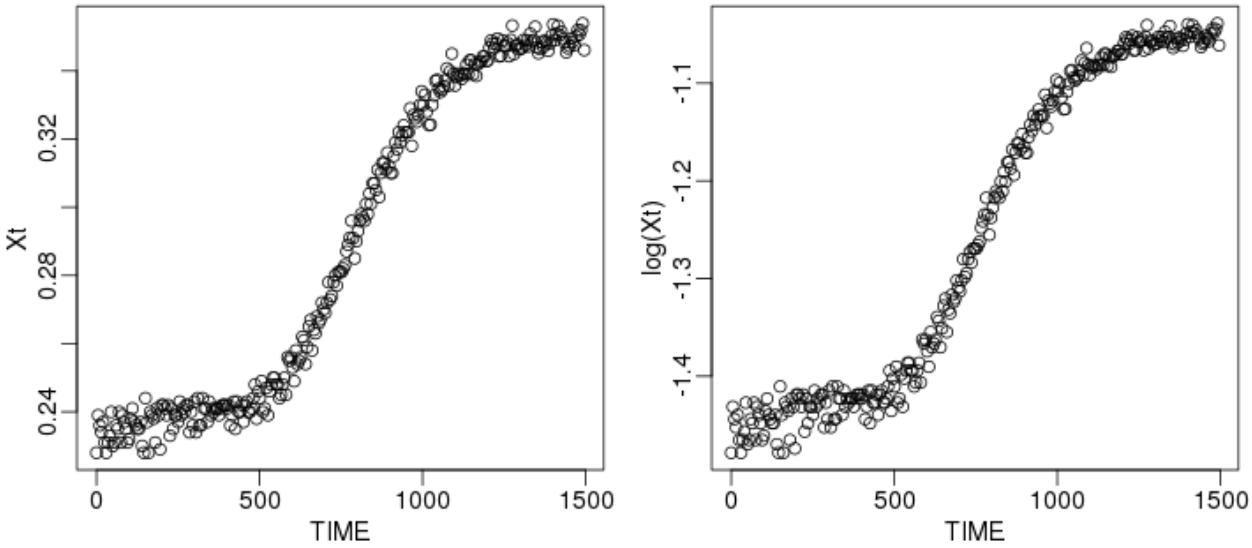


Linear and Non-linear Regression: analyzing a single data set

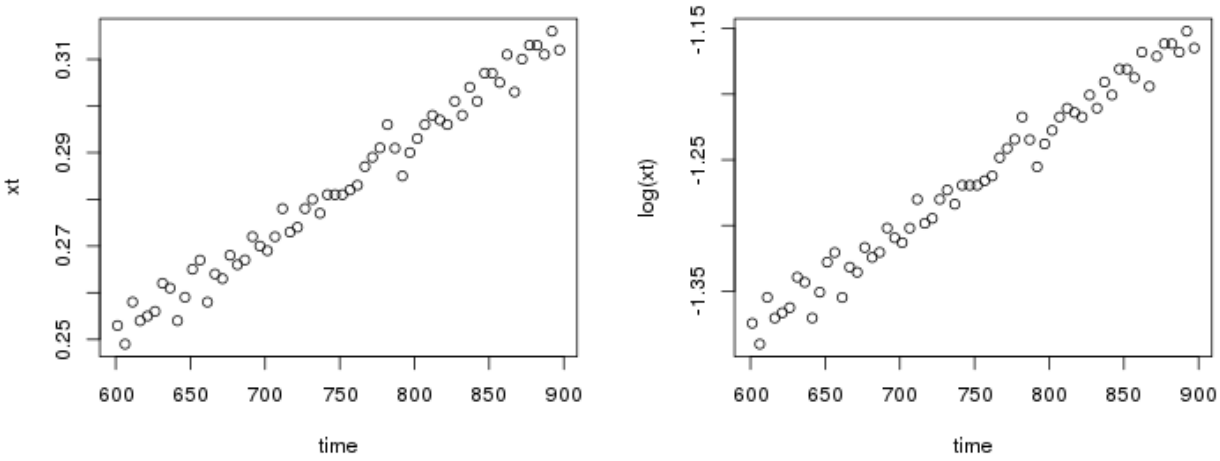
```
## GET A SINGLE DATASET
od <- getData(raw,"OD") # what is `od` ?
TIME <- raw$Time        # what does the $ do?
Xt <- od[,"A8"]
plot(TIME, Xt)
```



```
## cut to growth range
rng <- TIME<1500   # what is `rng` ?
Xt <- Xt[rng]
TIME <- TIME[rng]

## look at data
par(mfcol=c(1,2), mai=c(.75,.75,.1,.1), mgp=c(1.5,.5,0), cex=1.2)
plot(TIME, Xt)
plot(TIME, log(Xt)) # log it - default `log` in R is the natural logarithm, ln
```



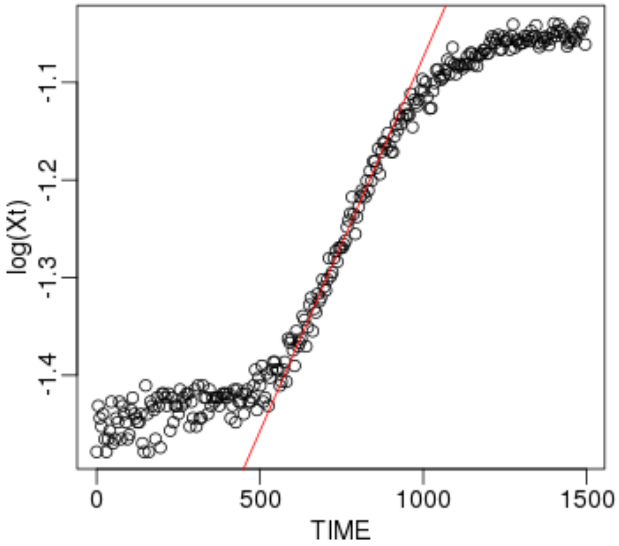
```
## cut to linear range of growth
rng <- TIME>600 & TIME < 900
xt <- Xt[rng]
time <- TIME[rng]

## look again at data
par(mfcol=c(1,2))
plot(time, xt)
plot(time, log(xt))
```



```
## DO LINEAR REGRESSION
## ln(X(t)) = mu * t + ln(X(0))
lfit <- lm(log(xt) ~ time) # what is `~` ?

## check quality of fit
summary(lfit)
```

```
##
## Call:
## lm(formula = log(xt) ~ time)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.021694 -0.006298 -0.001451  0.006707  0.023853
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.840e+00  1.122e-02  -164.0   <2e-16 ***
## time         7.663e-04  1.488e-05    51.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01001 on 58 degrees of freedom
## Multiple R-squared:  0.9786, Adjusted R-squared:  0.9782
## F-statistic:  2652 on 1 and 58 DF,  p-value: < 2.2e-16
```

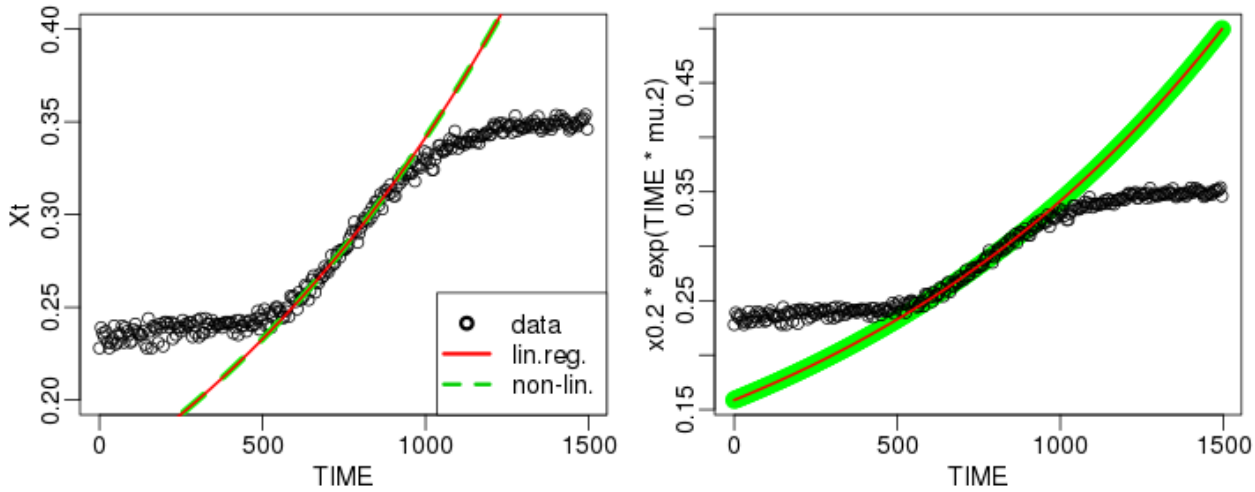
```
## get parameters from linear regression
x0.1 <- exp(coefficients(lfit)[1]) # e^ln(X(0)) = ?
mu.1 <- coefficients(lfit)[2] # mu

## plot
par(mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(TIME,log(Xt))
lines(TIME, mu.1*TIME + log(x0.1), col="red")
```



```
## DO NON-LINEAR REGRESSION, using
## the results of the linear fit as initial parameter guesses
dat <- data.frame(time=time, xt=xt)
start <- list(mu=mu.1,x0=x0.1)
nlfit <- nls(xt ~ x0*exp(mu*time),data=dat,start=start) # TRY MORE COMPLEX MODELS, see a few slides ahead

## check quality of fit
summary(nlfit)
```

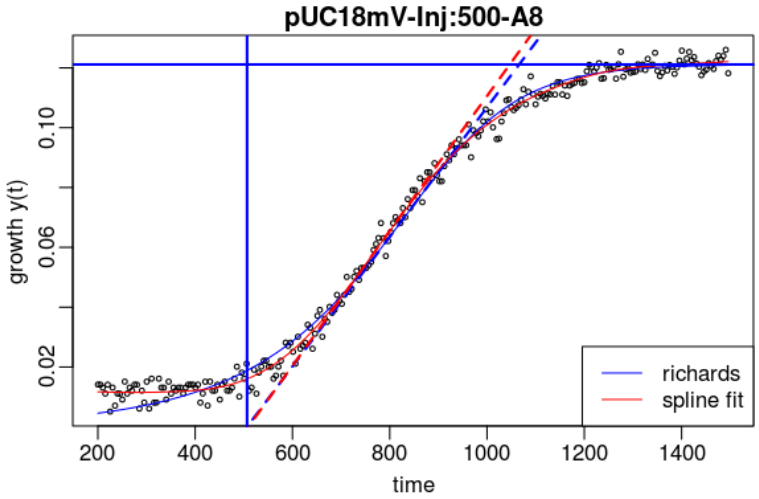
```
##
## Formula: xt ~ x0 * exp(mu * time)
##
## Parameters:
##     Estimate Std. Error t value Pr(>|t|)
## mu 7.660e-04  1.474e-05   51.99   <2e-16 ***
## x0 1.588e-01  1.792e-03   88.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.002792 on 58 degrees of freedom
##
## Number of iterations to convergence: 1
## Achieved convergence tolerance: 2.42e-06
```

```
## get parameters & plot results
mu.2 <- coefficients(nlfit)[1]
x0.2 <- coefficients(nlfit)[2]
par(mfcol=c(1,2),mai=c(.75,.75,.1,.1),mgp=c(1.5,.5,0),cex=1.2)
plot(TIME,Xt,ylim=c(0.2,.4))
lines(TIME, x0.2 * exp(TIME*mu.2),col="green", lty=2,lwd=5)
lines(TIME, x0.1 * exp(TIME*mu.1),col="red",lwd=2)
legend("bottomright",legend=c("data","lin.reg.","non-lin."),
       col=c(1,2,3),pch=c(1,NA,NA),lty=c(NA,1,2),lwd=3)
plot(TIME, x0.2 * exp(TIME*mu.2),col="green", lty=2,lwd=5)
points(TIME,Xt)
lines(TIME, x0.1 * exp(TIME*mu.1),col="red",lwd=2)
```



How many linear phases can you detect in your experiments? Why are there several?

Fitting Growth Models



- Initial Cell Density: $X(0)$
- Lag Phase: λ
- Exponential Phase: growth rate μ
- Stationary Phase: *capacity* A

https://cran.r-project.org/package=grofit

- Logistic Equation:
$$X(t) = \frac{A}{1+e^{\frac{4\mu}{A}(\lambda-t)+2}}$$
- Gompertz:
$$X(t) = Ae^{-e^{\frac{\mu e}{A}(\lambda-t)+1}}$$
- Modified Gompertz:
$$X(t) = Ae^{-e^{\frac{\mu e}{A}(\lambda-t)+1}} + Ae^{\alpha(t-t_{shift})}$$
- Richard's generalized logistic model:
$$X(t) = A(1+\nu e^{1+\nu+\frac{\mu}{A}(1+\nu)^{1+\frac{1}{\nu}}(\lambda-t)})^{-\frac{1}{\nu}}$$

as implemented in R package `grofit`

Try these equations with `nls`.
Do we need packages if `nls` works well?

29/52

Prepare Data: blanks, cuts, etc.

```
raw2 <- cutData(raw, rng=c(0,1550), mid="Time")
raw3 <- correctBlanks(raw2, plate,dids="OD",max.mid=1500)
```

```
## blanking OD
## blanking TRUE : 84 wells, using 12 blank wells
## OD
##  time bin: 1 1 - 309 skipping 10 bins at 1500
##  blank: 0.2435858298247
```

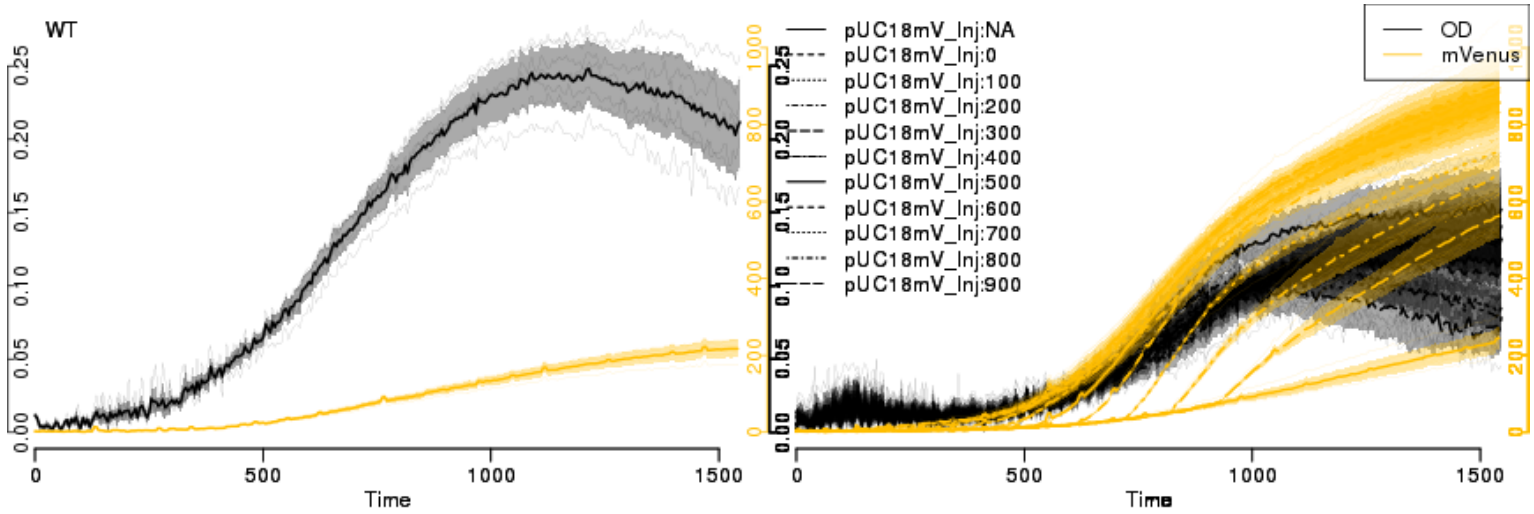
```
raw4 <- correctBlanks(raw3,plate,dids="mVenus",by=c("strain","IPTG"),
                      mbins=length(raw$Time)/5,verb=FALSE)
data <- adjustBase(raw4, base=0, add.fraction=.001,
                   wells=unlist(groups),xlim=c(1,which(raw$Time>1500)[1]),
                   each=TRUE, verb=FALSE) # set verb to TRUE
```

What happens in `correctBlanks` and `cutData`?

What does `adjustBase` do, and when could we need it?

Get Replicate Groups

```
groups <- getGroups(plate, c("strain"), verb=FALSE) # SET TO TRUE!
groups2 <- getGroups(plate, c("strain","IPTG"), verb=FALSE)
viewGroups(data, groups=groups, groups2=groups2, verb=FALSE)
```



What are the lines and areas?

What is the structure of the `groups` item? Can you make your own groupings?

Try different groupings and parameters to `viewGroups`.

Use Package grofit to Fit Growth Data

```
raw2 <- cutData(raw,rng=c(200,1500))
grodat <- data2grofit(raw2, did="OD", plate=plate,
                      wells=groups[["pUC18mV"]])
library(grofit)
fitparams <- grofit.2.control(interactive=FALSE, plot=TRUE) # SET ALL TO TRUE!!
pdf("growthrates.pdf")
fits <- gcFit.2(time=grodat$time, data=grodat$data, control=fitparams)
```

```
##
##
## = 1. growth curve ================================
## ------------------------------------------------
## --> Try to fit model logistic
```

```
## ....... OK
```

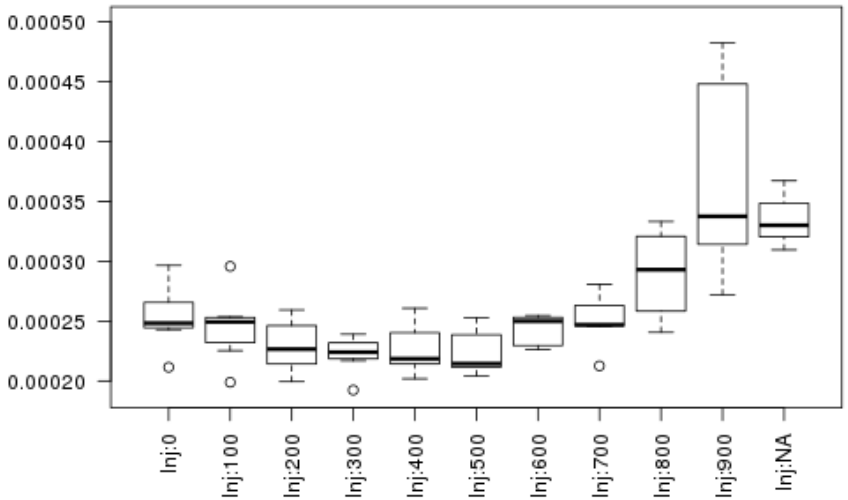
```
## --> Try to fit model richards
```



```
## ....... ERROR in nls(). For further information see help(gcFitModel)
```

Use Package `grofit` to Fit Growth Data

```
table <- grofitGetParameters(fits, p=c("AddId","mu.spline"))
boxplot(table[,"mu.spline"] ~ table[,"AddId"], las=2, ylim=c(1.9e-4,5e-4))
```



Try different well groups instead of `table[,"AddId"]`.
Are the fits Ok? How to get more information on the fits?

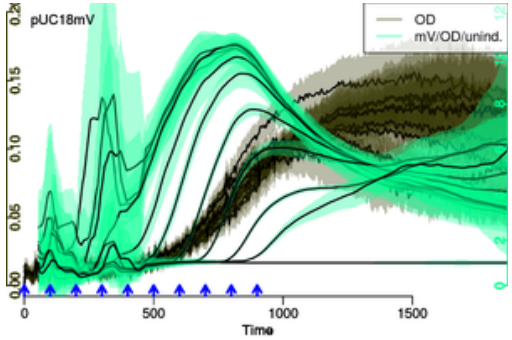
Gene Expression

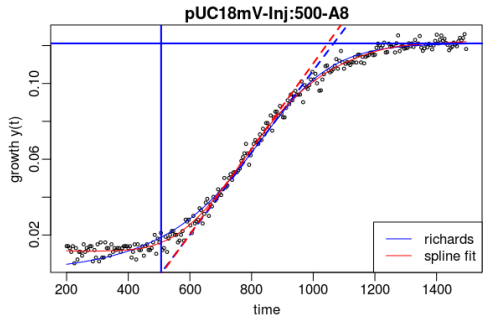
- Measured: total fluorescence per well
- Wanted: protein level per cell!

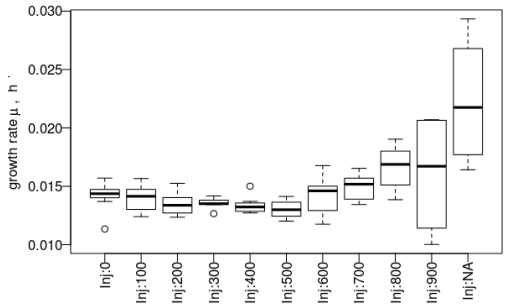
Assumptions:

1. Fluorescence is linear with protein level
2. OD is linear with cell number

⇒ **Fluor./OD ~ proteins/cell**

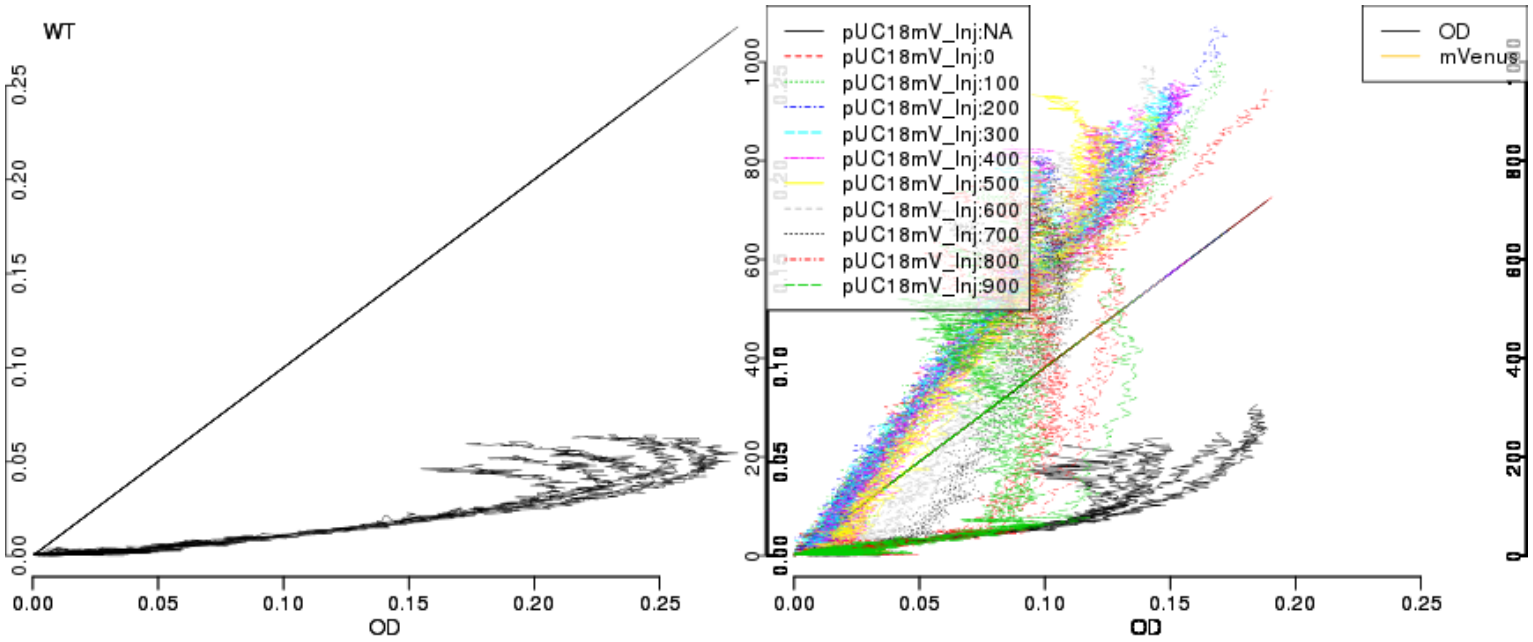






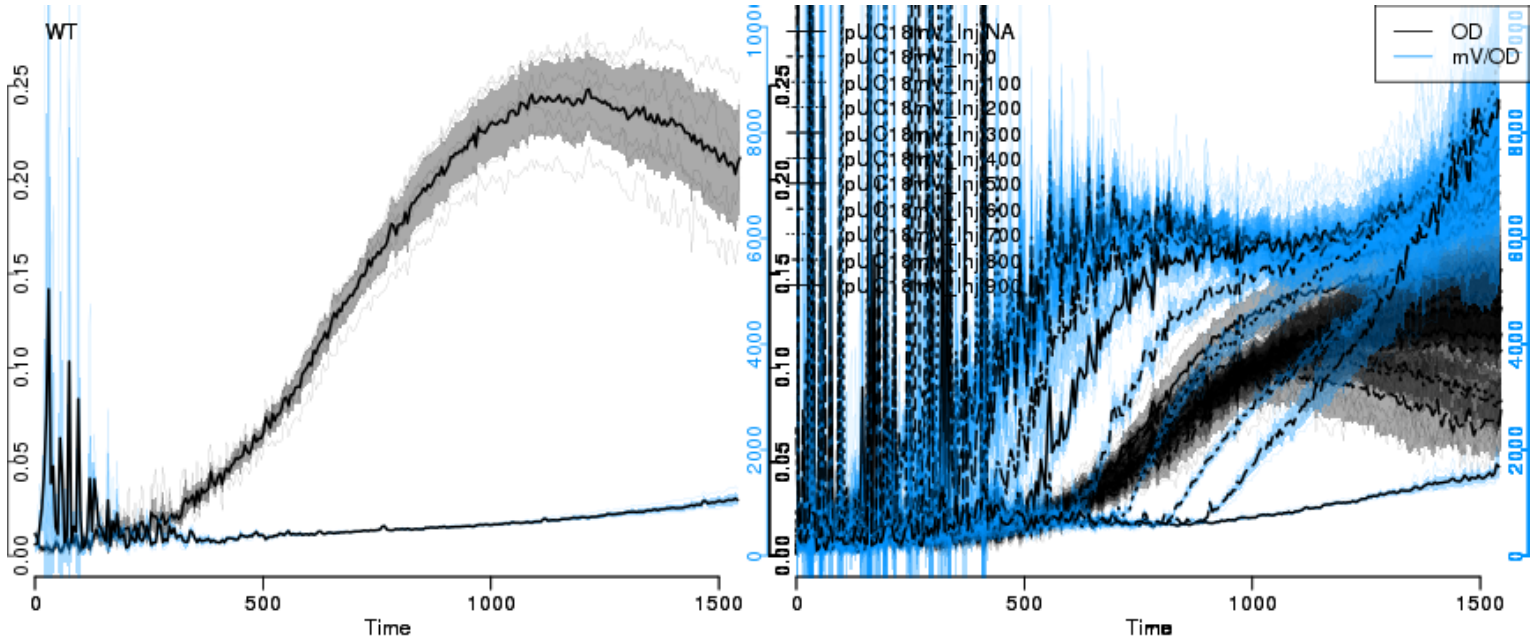
Gene Expression: Normalized Fluorescence - FL/OD

```
viewGroups(data, groups=groups, groups2=groups2, xid="OD", lwd.orig=.5, verb=FALSE)
```



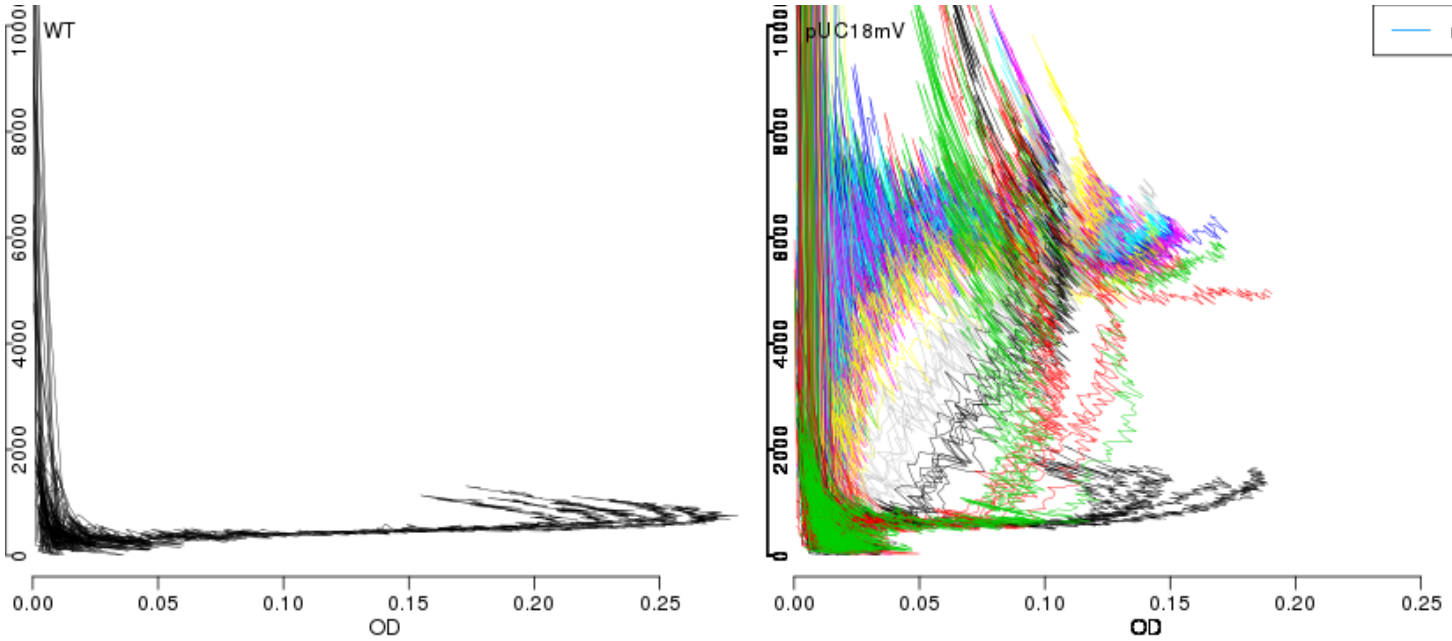
Gene Expression: Normalized Fluorescence - FL/OD

```
fl <- getData(data,"mVenus")
od <- getData(data,"OD")
data <- addData(data, ID="mV/OD", dat=fl/od, col="#0095FF")
viewGroups(data, groups=groups, groups2=groups2,
           dids=c("OD","mV/OD"),ylims=list("mV/OD"=c(0,1e4)),emphasize.mean=T,verb=F)
```



Gene Expression: Normalized Fluorescence - FL/OD

```
viewGroups(data, groups=groups, groups2=groups2, dids="mV/OD", xid="OD", lwd.orig=.5,
           ylim=c(0,1e4), g2.legend=FALSE, verb=FALSE)
```

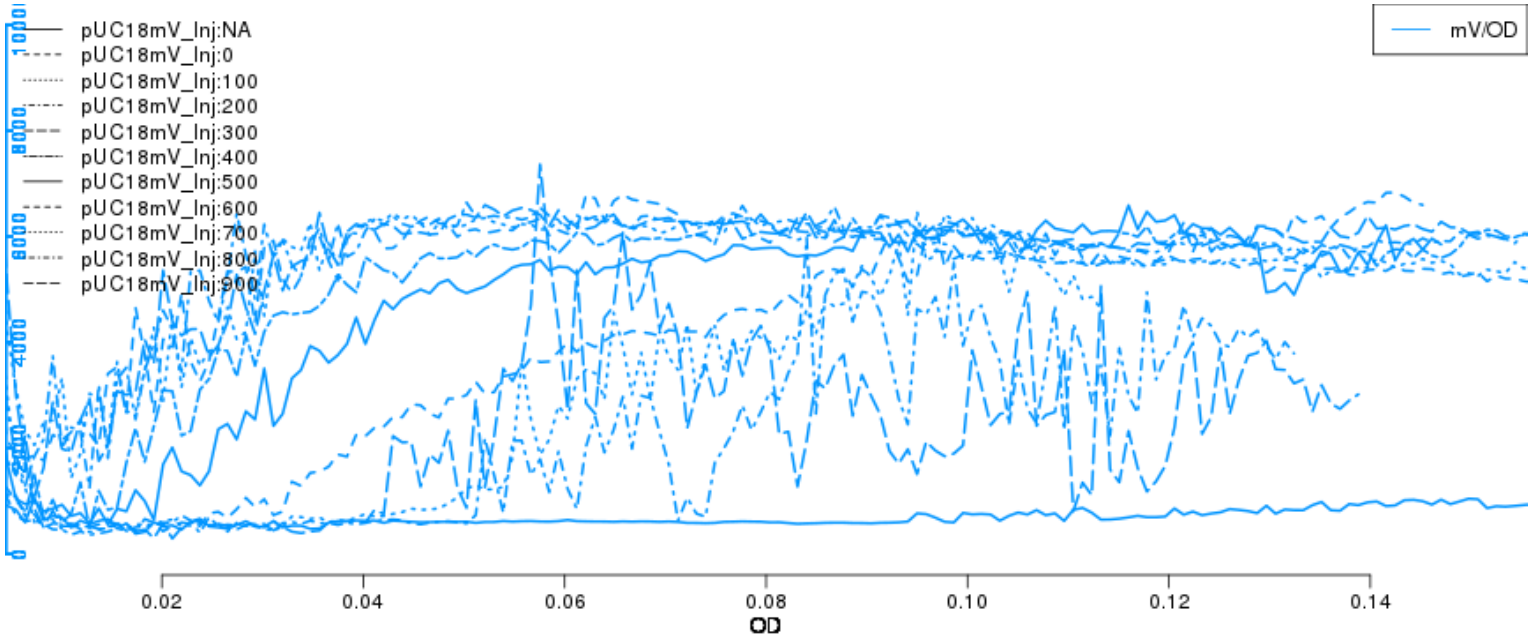


What could the slope $\frac{mV}{OD} \sim OD$ (where it's linear) mean?

Can we use it?

Gene Expression: Normalized Fluorescence - Interpolate to OD

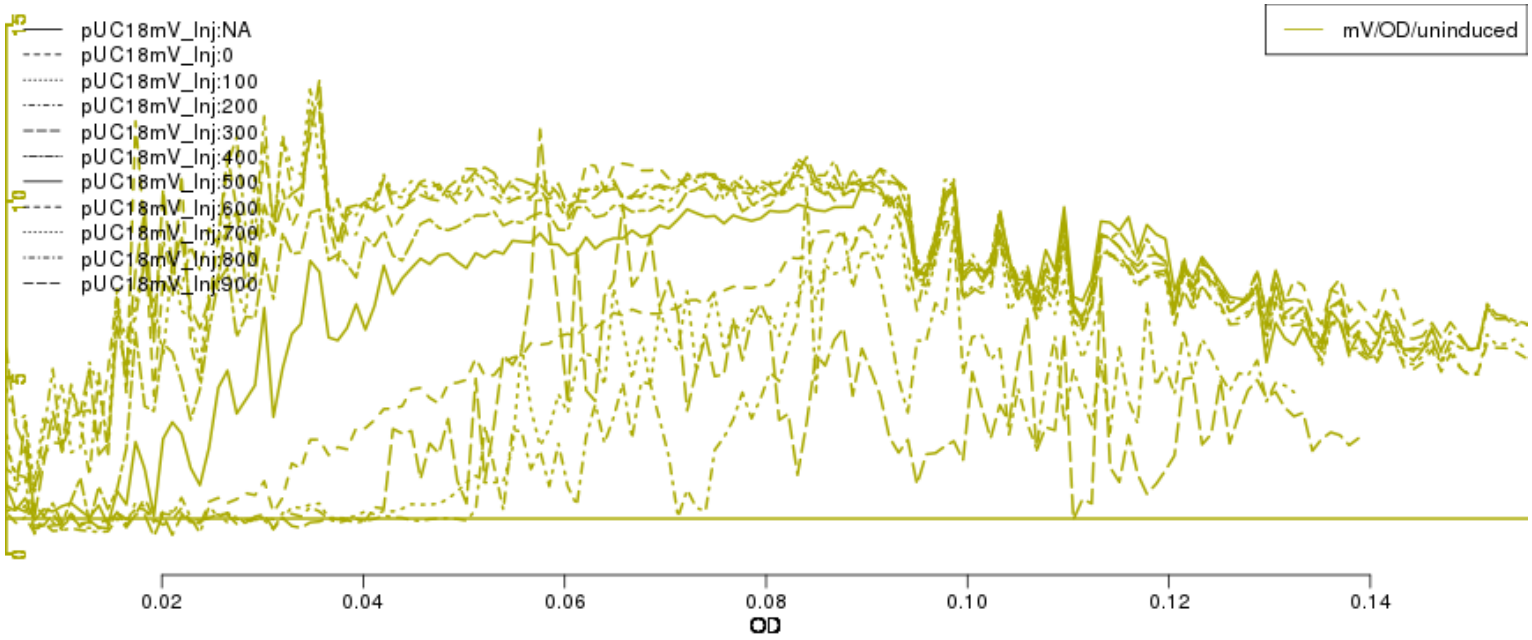
```
od.data <- interpolatePlateData(data,"OD")
viewGroups(od.data, groups=groups[2], groups2=groups2,xlim=c(.01,.15),
           ylims=list("mV/OD"=c(0,1e4)),dids="mV/OD", show.ci=F,lwd.orig=0, verb=F)
```



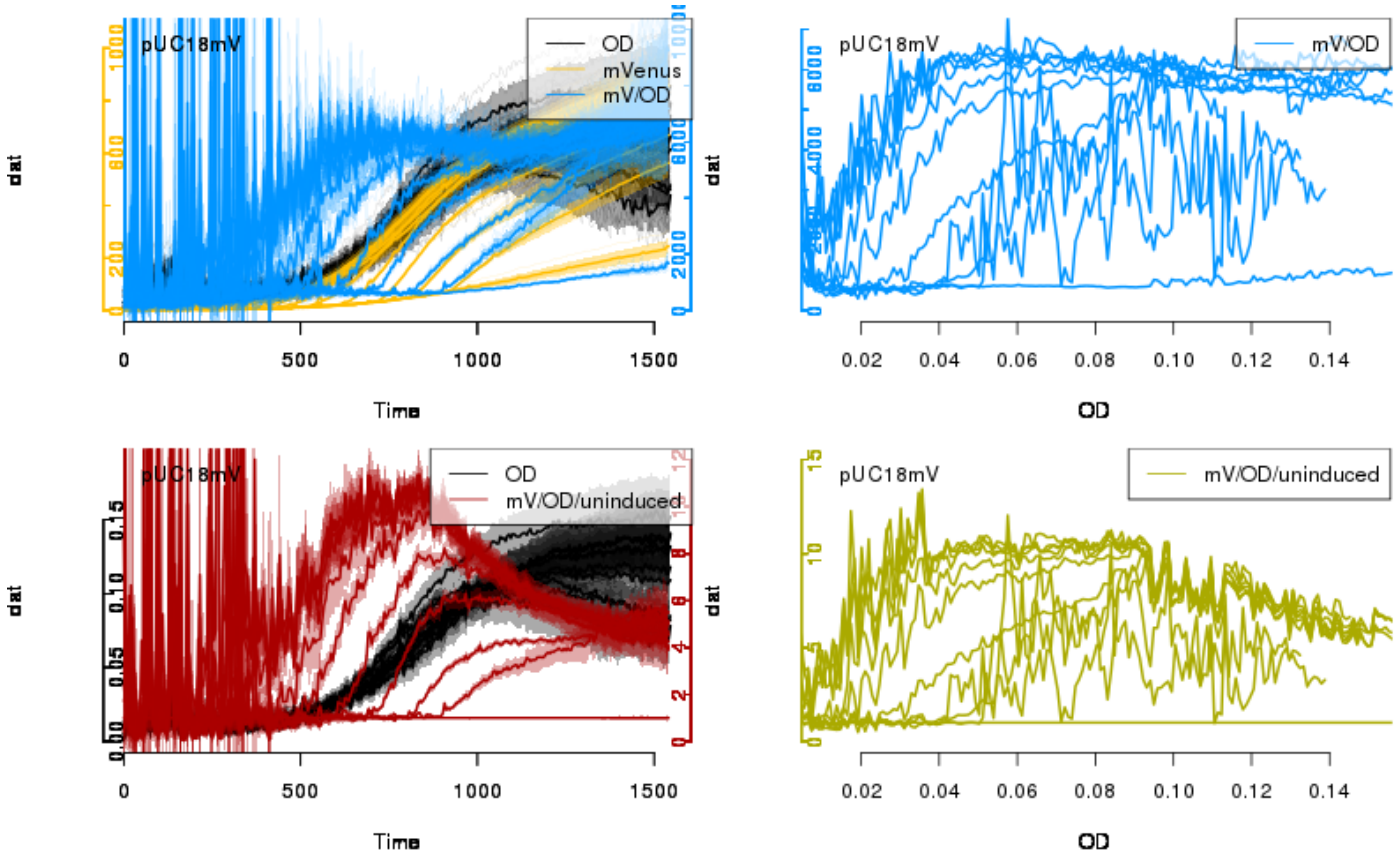
Can we smooth upstream data to get rid of the noise?

Gene Expression: The Fold Change

```
flod <- getData(od.data,"mV/OD")
uninduced <- rowMeans(flod[, groups2[["pUC18mV_Inj:NA"]]],na.rm=TRUE)
od.data <- addData(od.data, ID="mV/OD/uninduced", dat=flod/uninduced, col="#AAAA00")
```



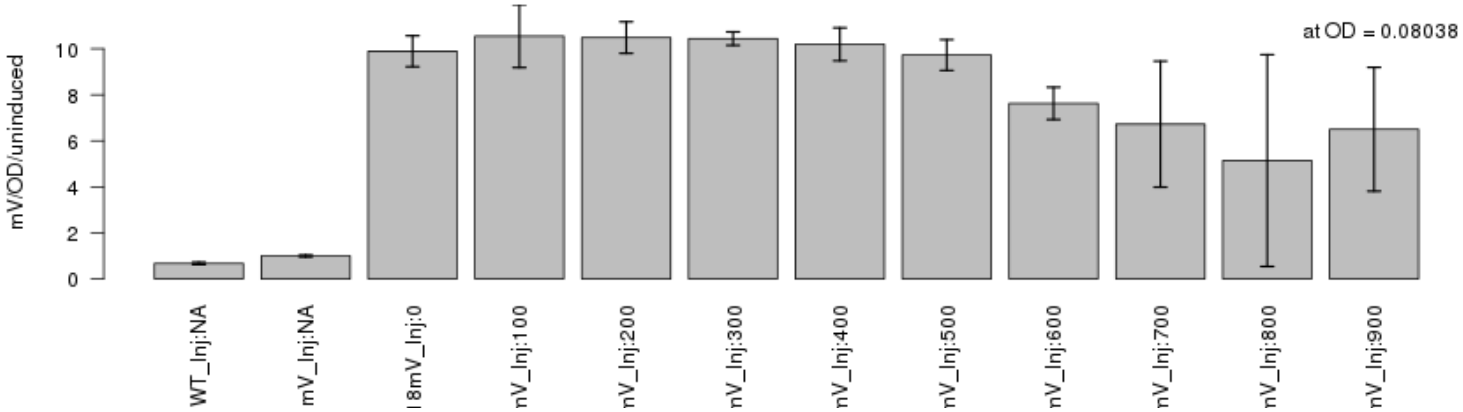
Gene Expression: Data Normalization



What's wrong with `mV/OD/uninduced` vs. `time` w/o interpolation?

Gene Expression: Data Normalization - Result

```
results <- boxData(od.data, did="mV/OD/uninduced", rng=.08, groups=groups2, type="bar")
```



Extracts values in given range; plots by groups, and returns values for each well.

What are the error bars?

See `?cutData` on how to obtain values in a given range directly.

Or try yourself with base R.

Gene Expression: Data Normalization - Result

```
library(platexpress)
head(results)
```

```
##   well     group mV/OD/uninduced_0.08038
## 1   A1 WT_Inj:NA                0.6856046
## 2   B1 WT_Inj:NA                0.6510172
## 3   C1 WT_Inj:NA                0.6200460
## 4   D1 WT_Inj:NA                0.7815149
## 5   E1 WT_Inj:NA                0.6994579
## 6   F1 WT_Inj:NA                0.6037127
```

Calculate confidence intervals and error bars yourself.

```
imgtxt <- matrix(results[,1], nrow=7, ncol=12) # MATRIX THAT LOOKS LIKE PLATE
imgdat <- matrix(results[,3], nrow=7, ncol=12)
tmp <- t(apply(imgdat, 2, rev)) # WHAT HAPPENS HERE?
par(mai=c(.5,.5,.01,.01))
image(x=1:12, y=1:7, z=log(tmp), axes=FALSE, ylab=NA, xlab=NA)
text(x=rep(1:12,7), y=rep(7:1,each=12), paste(t(imgtxt),":\n",t(round(imgdat,2))))
axis(1,at=1:12); axis(2, at=1:7, labels=toupper(letters[7:1]), las=2)
```

In which order does `image` plot rows and columns of a matrix?

Can you spot any systematic plate effects?

hide: pixels in the first row, from left to right

Gene Expression - Summary

1. Growth rate μ matters!
2. Fluorescence per OD as a measure of proteins per cell.
   - Note noise accumulation at low values. **Why?**
3. Interpolate data to a common OD.
4. Calculate fold-ratio to control.
   - Be careful; what is your control?
5. Get data at a given OD, or search maximum, etc.

What other measures would be interesting or could be required for a quantitative model?

From Wells to Cells

Conversion of Carbon&Energy Source S to biomass X:

$$S \longrightarrow y\ X + (1\text{-}y)\ P$$

What are substrate S and product P in our experiments?

Catalyzed by X:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu X$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\frac{1}{y}\,\mu X$$

$$\mu = \mu_{max}\,\frac{S}{S+K}$$

How can we estimate the yield y? Does y have units? Which?
What is the steady state of the system?

From Wells to Cells

Continuous Culture - Dilution Rate ϕ

$$\frac{\mathrm{d}X}{\mathrm{d}t} = (\mu - \phi)X$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = (S_{in} - S)\phi - \frac{1}{y}\mu X$$

$$\mu = \mu_{max}\frac{S}{S + K}$$

What is the steady state of above system?
What is the feedback structure of growth?
Can above system show more complex dynamics, such as oscillations or multistablity?

From Wells to Cells: Growth

Fluorescent Protein F, total culture concentration:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu X$$
$$\frac{\mathrm{d}F}{\mathrm{d}t} = K(?) - \delta F$$

Intracellular concentration:

$$f = \frac{F}{XV_c}$$

What is K, must be $\sim X$, right ?
How can we map F from total culture to intracellular concentration f ?
Calculate $\frac{\mathrm{d}f}{\mathrm{d}t}$.

From Wells to Cells : Growth

$$
\begin{aligned}
f &= \frac{F}{XV_c} \;;\; v_f = XV_c \\
\dot{f} &= \dot{F}\,\frac{1}{XV_c} - \dot{X}\,\frac{F}{X}\,\frac{1}{XV_c} \\
&= \dot{F}\,\frac{1}{v_f} - \dot{X}\,\frac{F}{X}\,\frac{1}{v_f} \\
&= (K - \delta f\, v_f)\,\frac{1}{v_f} - \mu X\,\frac{f v_f}{X}\,\frac{1}{v_f} \\
&= \frac{K}{v_f} - (\delta + \mu) f \\
K &= kXV_c
\end{aligned}
$$

What is XV_c ?

Note the term $\delta + \mu$. What is it, and where did the μ come from?

From Wells to Cells : Growth

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu X$$
$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\frac{1}{y}\,\mu X$$
$$\frac{\mathrm{d}f}{\mathrm{d}t} = k - (\delta + \mu) f$$

- Add activation of f production by an inducer and modulation by a riboswitch r.
- Add expression of both from a plasmid (which itself increases exponentially within cells).
- And how could we account for the effect of induced gene expression on growth rate?

. . . and find appropriate parameters (back in data hell),
start e.g. at `bionumbers`.

PARAMETER	UNIT	VALUE	DESCRIPTION	SOURCE
y	g DCW / g glucose	0.5	yield factor $\frac{\Delta X}{-\Delta S}$	bionumbers
V_c	µm3	1	average cell volume	bionumbers
C_c	fg/µm3	242∓43	cell carbon content	bionumbers
DCW/OD	(g/L)/OD	0.36	g DCW per OD600	bionumbers
cells/OD	(cells/mL)/OD	5.9-21 1e8	cells per OD600	bionumbers
μ_{max}	1/h		maximal growth rate	estimate from exponential phase
K	mol/L		S where $\mu = \mu_{max}/2$	estimate from data
k	mol/(L*h)		protein expression rate	search literature/databases
δ	1/h		protein degradation rate	search literature/databases

From Wells to Cells : Riboswitch

$$\frac{\mathrm{d}sRNA}{\mathrm{d}t} = CP_s(TetR, aTc) - (\mu + \delta_s)sRNA$$
$$\frac{\mathrm{d}mRNA}{\mathrm{d}t} = CP_m(LacI, IPTG) - (\mu + \delta_m)mRNA$$
$$\frac{\mathrm{d}GFP}{\mathrm{d}t} = (r_0 mRNA + r_1 sRNA :: mRNA)\,\frac{m}{m + \mu + \delta_g} - (\mu + \delta_g)GFP$$

with transcriptional activities:

$$P_m(LacI, IPTG) = P_m^0\,\frac{1 + \frac{1}{f_{lac}}\left(\frac{LacI}{K_{lac}(1+\frac{IPTG}{K_{IPTG}})}\right)^{n_{lac}}}{1 + \left(\frac{LacI}{K_{lac}(1+\frac{IPTG}{K_{IPTG}})}\right)^{n_{lac}}}$$

$$P_s(TetR, aTc) = P_s^0\,\frac{1 + \frac{1}{f_{tet}}\left(\frac{TetR}{K_{tet}(1+\frac{aTc}{K_{aTc}})}\right)^{n_{tet}}}{1 + \left(\frac{TetR}{K_{tet}(1+\frac{aTc}{K_{aTc}})}\right)^{n_{tet}}}$$

What and where is sRNA::mRNA?

What are all the parameters? Can you derive equations for the steady state?

From Wells to Cells : Riboswitch

Rodrigo *et al.* 2012 - Steady State Solution:

$$GFP_{ss}(IPTG, aTc) = F_0 \frac{1 + f_1(\frac{IPTG}{K_1})^{n_1} + f_2(\frac{aTc}{K_2})^{n_2} + f_1 f_{sRNA}(\frac{IPTG}{K_1})^{n_1}(\frac{aTc}{K_2})^{n_2}}{1 + (\frac{IPTG}{K_1})^{n_1} + (\frac{aTc}{K_2})^{n_2} + (\frac{IPTG}{K_1})^{n_1}(\frac{aTc}{K_2})^{n_2}}$$

What are these parameters>? See Supplement of Rodrigo *et al.* 2012

Plot this function in R or `python`, using the concentration ranges in your experiments: can you use `image`? Can R do 3D plots?

Is the steady state assumption justified? Can we combine it with our cell growth model?