

Single fit and overlay

h qin

June 9, July 6-11, 2017

20171002 Add AIC The log-likelihood values calculated by FlexSurv and my own routine return values with 10x differences. This is an old problem. see <http://hongqinlab.blogspot.com/2014/07/compare-binomial-gompertz-and-weibull.html> It seems FlexSurv used the loglikelihood value from optim(). When I switch to optim returned \$value, the results are comparable.

20170424. Major bug found. file names and strains names are inconsistent.

20160711-> Fitting RLS dataset merged by strains.

20160706-0711. Finished batch fitting of all individual RLS data sets. The fitting results showed that 'n' are often in the range of [5,7], though it is very noisy.

```
rm(list=ls())
#setwd("~/github/0.network.aging.prj.bmc/0a.rls.fitting")
#setwd("~/github/bmc_netwk_aging_manuscript/R1/0.nat.rls.fitting")
library('flexsurv')

## Loading required package: survival
source("../lifespan.r")

##### log likelihood function, R-fixed 2-parameter binomial mortality rate model
# http://hongqinlab.blogspot.com/2013/12/binomial-mortality-model.html
#  $m = R ( 1 + t/t_0 )^{-(n-1)}$ 
#  $s = \exp( (R t_0/n) * (1 - (1+t/t_0)^n ) )$ 

llh.binomialMortality.single.run.NoR0 <- function( Rt0n, lifespan, debug=0 ) {
  I = R0;
  t0 = Rt0n[2]; n=Rt0n[3];
  my.data = lifespan[!is.na(lifespan)];
  log_s = (I * t0 /n )*(1 - (1 + my.data/t0)^n);
  log_m = log(I) + (n-1) * log(1 + my.data/t0 );
  my.lh = sum(log_s) + sum(log_m);
  if(debug) { print (c(I, Rt0n )); } #trace the convergence
  ret = - my.lh # because optim seems to maximize
}

llh.binomialMortality.single.run.FixedN <- function( Rt0n, lifespan, debug=0 ) {
  I=Rt0n[1]; t0 = Rt0n[2]; #n=Rt0n[3]; n defined outside
  my.data = lifespan[!is.na(lifespan)];
  log_s = (I * t0 /n )*(1 - (1 + my.data/t0)^n);
  log_m = log(I) + (n-1) * log(1 + my.data/t0 );
  my.lh = sum(log_s) + sum(log_m);
  if(debug) { print (c(I, Rt0n )); } #trace the convergence
  ret = - my.lh # because optim seems to maximize
}
```

Parse strains from files

```
set.seed(20170101)
```

Take files from natural isolates

```
my.strains=c("101S", "M1-2", "M13", "M14", "M2-8", "M22", "M32", "M34", "M5", "M8", "RM112N", "S288c", "SGU57", "Y")

report = data.frame(cbind(my.strains))
report$samplesize = NA; report$R=NA; report$t0=NA; report$n=NA; report$G=NA; report$longfilename=NA;
```

Now, fit all RLS data sets by strains

```
for( i in 1:length(report[,1])){
  #for( i in 3:4){
    filename = paste("rls/", my.strains[i], ".csv", sep=' ');
    report$longfilename[i] = filename;
    tb = read.csv( filename, header =F)

    report$samplesize[i] = length(tb[,1])

    GompFlex = flexsurvreg(formula = Surv(tb[,1]) ~ 1, dist = 'gompertz')
    WeibFlex = flexsurvreg(formula = Surv(tb[,1]) ~ 1, dist = 'weibull')

    report$avgLS[i] = mean(tb[,1])
    report$stdLS[i] = sd(tb[,1])
    report$CV[i] = report$stdLS[i] / report$avgLS[i]

    report$GompGFlex[i] = GompFlex$res[1,1]
    report$GompRFlex[i] = GompFlex$res[2,1]
    report$GompLogLikFlex[i] = round(GompFlex$loglik, 1)
    report$GompAICFlex[i] = round(GompFlex$AIC)

    report$WeibShapeFlex[i] = WeibFlex$res[1,1]
    report$WeibRateFlex[i] = WeibFlex$res[2,1]
    report$WeibLogLikFlex[i] = round(WeibFlex$loglik, 1)
    report$WeibAICFlex[i] = round(WeibFlex$AIC)

    #set initial values
    Rhat = report$GompRFlex[i]; # 'i' was missing. a bug costed HQ a whole afternoon.
    Ghat = report$GompGFlex[i];
    nhathat = 8;
    t0= (nhathat-1)/Ghat;

    if( is.na( match(my.strains[i], c('M8' )) ) ) {
      fitBinom = optim ( c(Rhat, t0, nhathat), llh.binomialMortality.single.run,
        lifespan=tb[,1],
        #method='SANN') #SANN needs control
        method="L-BFGS-B",
        lower=c(1E-4, 1, 2), upper=c(0.1,200,100) );
      report[i, c("R", "t0", "n")] = fitBinom$par[1:3]
      report$BinomialAIC[i] = 2*3 + 2*fitBinom$value
    }
```

```

} else { #M8, fix n
  print(my.strains[i])
  #llh.binomialMortality.single.run.FixedN <- function( RtOn, lifespan, debug=0 ) {
  n=11.5;
  fitBinom = optim ( c(Rhat, t0, nhathat), llh.binomialMortality.single.run.FixedN,
                    lifespan=tb[,1],
                    #method='SANN') #SANN needs control
                    method="L-BFGS-B",
                    lower=c(1E-5, 1, 2), upper=c(0.1,200,100) );
  report[i, c("R", "t0", "n")] = c(fitBinom$par[1:2], n)
  report$BinomialAIC[i] = 2*3 + 2*fitBinom$value
}
report$G[i] = (report$n[i] - 1)/report$t0[i]

#AIC
# llh.binomialMortality.single.run <- function( RtOn, lifespan, debug=0 )
#report$BinomialAICqin[i] = 2*3 + 2*llh.binomialMortality.single.run( report[i, c("R", "t0", "n")], l

#llh.G.single.run <- function( IG, lifespan ) {
#report$GompAICqin[i] = 2*2 + 2* llh.G.single.run(report[i,c("GompRFlex", "GompGFlex")], lifespan=tb[,1],
#  # dgompertz <- function(x, shape, rate=1, log=FALSE)
#report$GompAICdgompertz[i] = 2*2 - 2* sum(dgompertz(tb[,1],shape=report$GompGFlex[i], rate=report$GompRFlex[i])
#report$GompAICdgompertz2[i] = 2*2 - 2* sum(dgompertz(tb[,1],shape=report$GompRFlex[i], rate=report$GompGFlex[i])
}

```

```
## [1] "M8"
```

Show the results

```

#report[ grep("tBY", report$strains), ]
report

```

##	my.strains	samplesize	R	t0	n	G
## 1	101S	85	0.0019760147	52.08681	9.174228	0.15693470
## 2	M1-2	54	0.0027664362	56.71641	9.475570	0.14943770
## 3	M13	70	0.0030071379	56.81672	9.648416	0.15221605
## 4	M14	60	0.0024869119	76.42752	9.180570	0.10703698
## 5	M2-8	105	0.0036633868	60.10209	10.068263	0.15088099
## 6	M22	60	0.0025096036	64.81929	9.400784	0.12960313
## 7	M32	60	0.0022090312	49.67204	9.556550	0.17226088
## 8	M34	58	0.0021231171	44.49124	8.940977	0.17848407
## 9	M5	166	0.0035875496	104.60572	9.903523	0.08511506
## 10	M8	60	0.0002276221	45.20362	11.500000	0.23228229
## 11	RM112N	59	0.0022367518	78.29925	8.184119	0.09175208
## 12	S288c	41	0.0053611878	80.43622	9.959994	0.11139252
## 13	SGU57	58	0.0067256427	78.02415	9.696555	0.11145978
## 14	YPS128	69	0.0024378049	58.96767	8.260993	0.12313514
## 15	YPS163	130	0.0019151177	52.24318	8.437491	0.14236291
##	longfilename	avgLS	stdLS	CV	GompGFlex	GompRFlex
## 1	rls/101S.csv	31.34118	7.512772	0.2397093	0.13407292	0.0012654562
## 2	rls/M1-2.csv	27.83333	9.201620	0.3305971	0.12310344	0.0024494207
## 3	rls/M13.csv	26.54286	9.142488	0.3444425	0.12284897	0.0029261133
## 4	rls/M14.csv	36.55000	12.804164	0.3503191	0.09148573	0.0019510502
## 5	rls/M2-8.csv	24.80952	8.133614	0.3278424	0.11608390	0.0043612883

```
## 6      rls/M22.csv 31.83333 10.271182 0.3226549 0.10778993 0.0021454528
## 7      rls/M32.csv 27.96667  6.888868 0.2463242 0.14043277 0.0017384578
## 8      rls/M34.csv 27.01724  8.206740 0.3037593 0.15692797 0.0012896468
## 9      rls/M5.csv  36.62651 12.938747 0.3532618 0.06684902 0.0041472501
## 10     rls/M8.csv  34.93333  6.905823 0.1976858 0.15888831 0.0003653141
## 11 rls/RM112N.csv 44.06780 13.006450 0.2951464 0.08938574 0.0010363304
## 12 rls/S288c.csv  26.26829 10.254327 0.3903690 0.08686882 0.0064512047
## 13 rls/SGU57.csv  23.86207 10.538898 0.4416590 0.08956763 0.0076748634
## 14 rls/YPS128.csv 35.00000  9.719598 0.2777028 0.11866014 0.0011041546
## 15 rls/YPS163.csv 34.43077  8.591449 0.2495282 0.13387276 0.0007889455
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 1          -295.0           594      4.778139      34.15561      -291.0
## 2          -193.6           391      3.538533      30.92356      -195.6
## 3          -250.1           504      3.114056      29.26548      -259.1
## 4          -233.7           471      3.361407      40.80155      -236.8
## 5          -373.3           751      3.371375      27.61417      -368.5
## 6          -222.9           450      3.596392      35.37118      -223.9
## 7          -205.6           415      4.419104      30.62828      -201.4
## 8          -198.0           400      3.985745      29.77062      -203.7
## 9          -670.1          1344      3.098392      40.98709      -658.3
## 10         -201.9           408      5.872213      37.70008      -200.1
## 11         -232.0           468      4.038465      48.63895      -233.7
## 12         -154.2           312      2.792964      29.42942      -153.3
## 13         -216.5           437      2.455703      26.85910      -218.2
## 14         -252.2           508      4.294122      38.52990      -253.0
## 15         -459.6           923      4.825655      37.62671      -460.2
##      WeibAICFlex BinomialAIC
## 1           586      601.3715
## 2           395      394.9773
## 3           522      508.0275
## 4           478      476.7300
## 5           741      749.3801
## 6           452      453.6004
## 7           407      417.9426
## 8           411      408.2894
## 9          1321     1339.8746
## 10          404      409.6617
## 11          471      478.0430
## 12          311      313.4152
## 13          440      439.2137
## 14          510      519.8934
## 15          924      946.1981
```

```
summary(report[, c("avgLS", "t0", "n")])
```

```
##      avgLS           t0           n
## Min.   :23.86   Min.   : 44.49   Min.   : 8.184
## 1st Qu.:26.78   1st Qu.: 52.16   1st Qu.: 9.058
## Median :31.34   Median : 58.97   Median : 9.476
## Mean   :31.27   Mean   : 63.93   Mean    : 9.426
## 3rd Qu.:34.97   3rd Qu.: 77.23   3rd Qu.: 9.800
## Max.   :44.07   Max.   :104.61   Max.    :11.500
```

#Overlay the binomial aging model with observation. see <http://hongqinlab.blogspot.com/2013/12/binomial-mortality-model.html> $m = R \left(1 + t/t_0 \right)^{(n-1)}$ $s = \exp \left((R \ t_0/n) * (1 - (1+t/t_0)^n) \right)$

```

for( i in 1:length(my.strains)){
#for( i in 3:4){
  filename = paste("rls/", my.strains[i], ".csv", sep='' );
  tb = read.csv( filename, header =F)

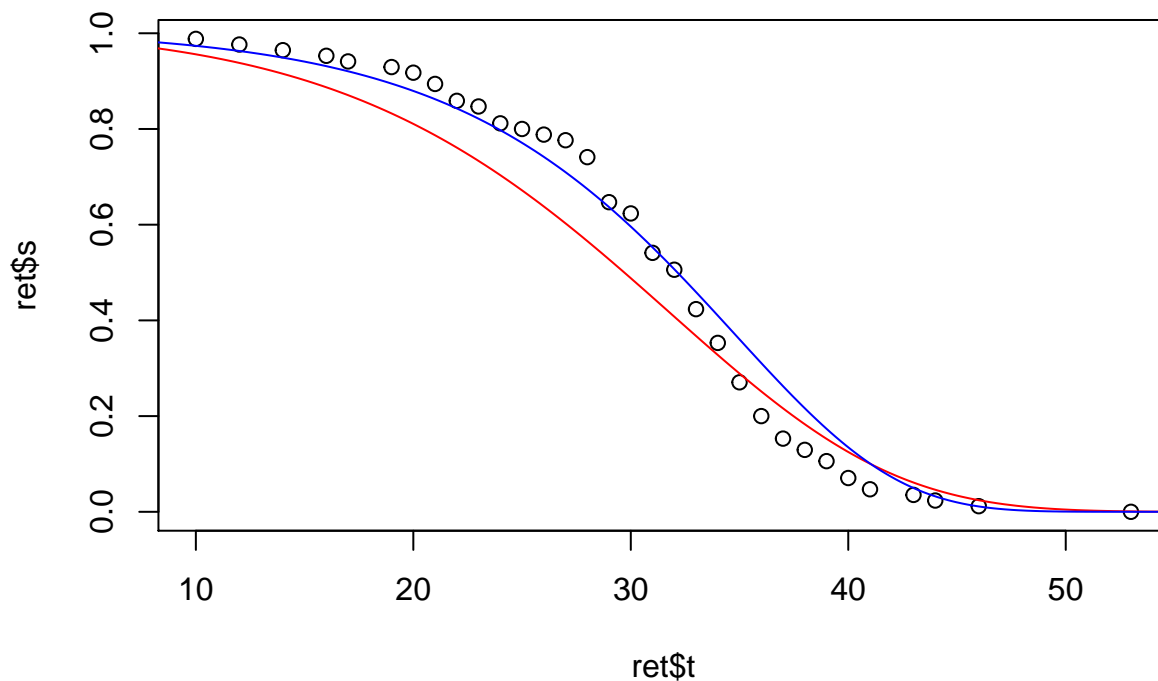
  ret = calculate.s(tb[,1])
  plot( ret$s ~ ret$t, main=my.strains[i]);
  print (report[i, ]);

  #overlay binomial aging viability
  print (report[i, c("R", "t0", "n", "G")] );
  t = seq(0,max(ret$t*1.1), 0.1);
  # s = exp( (R t0/n)*(1 - (1+t/t0)^n ) )
  s = exp( (report$R[i]* report$t0[i]/report$n[i])*(1 - (1+t/report$t0[i])^report$n[i] ) ) );
  lines(s ~ t, col='red')

  #overlay gompertz viability
  s.g = G.s( c(report$GompRFlex[i], report$GompGFlex[i], 0), t )
  lines(s.g ~ t, col='blue')
}

```

101S



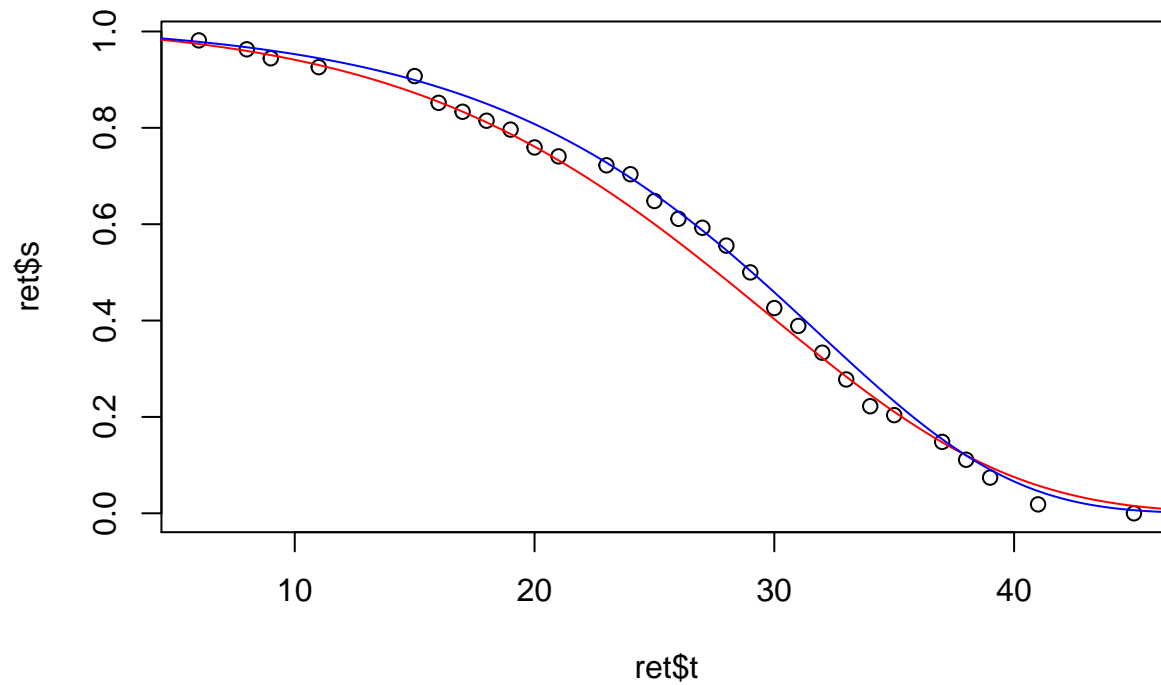
```

## my.strains samplesize      R      t0      n      G
## 1      101S          85 0.001976015 52.08681 9.174228 0.1569347
## longfilename  avgLS   stdLS      CV GompGFlex  GompRFlex
## 1 rls/101S.csv 31.34118 7.512772 0.2397093 0.1340729 0.001265456
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 1      -295          594      4.778139      34.15561      -291
## WeibAICFlex BinomialAIC
## 1      586      601.3715

```

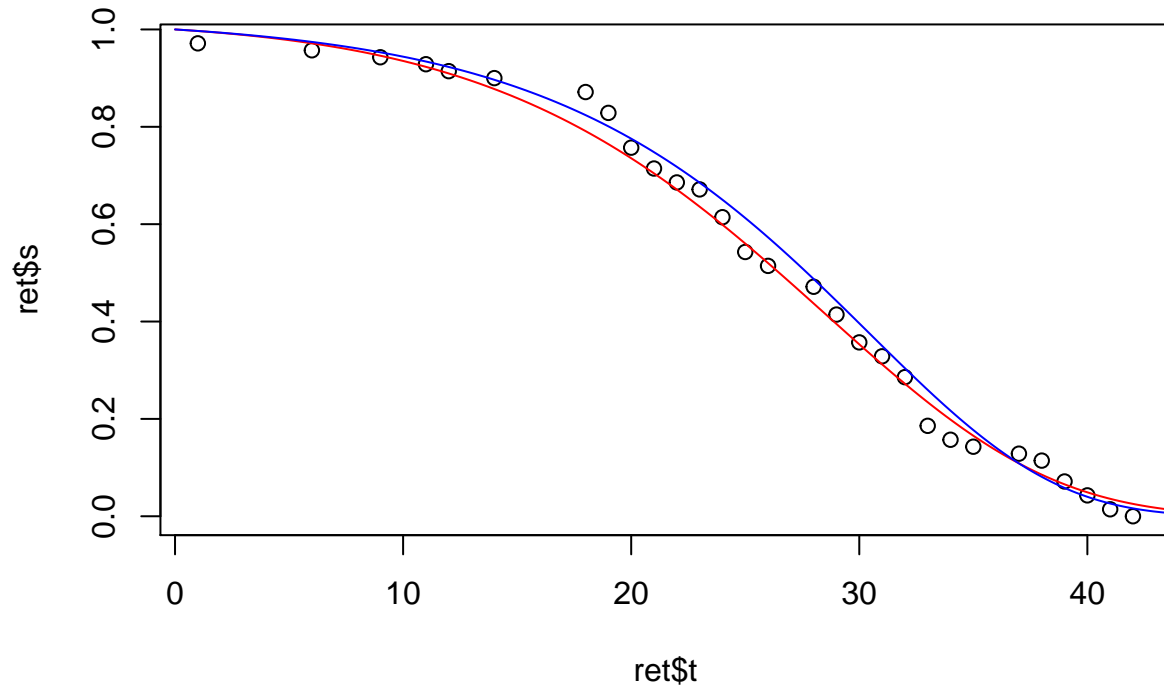
```
##          R      t0      n      G
## 1 0.001976015 52.08681 9.174228 0.1569347
```

M1-2



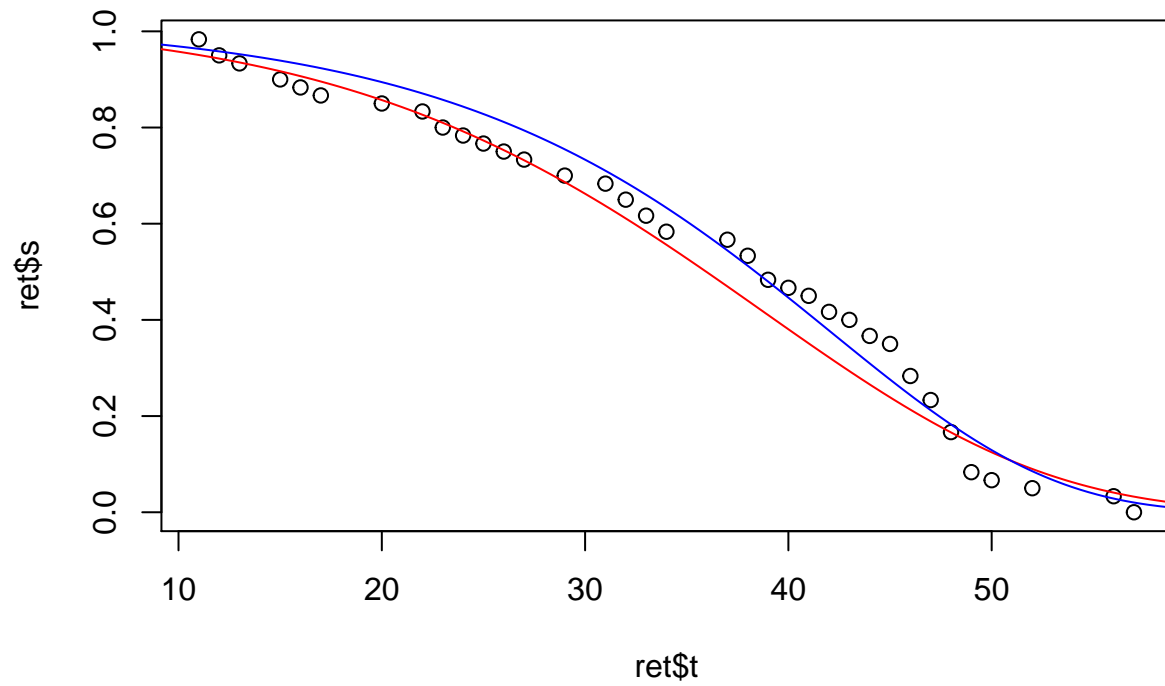
```
## my.strains samplesize      R      t0      n      G
## 2      M1-2          54 0.002766436 56.71641 9.47557 0.1494377
## longfilename  avgLS  stdLS      CV GompGFlex  GompRFlex
## 2 rls/M1-2.csv 27.83333 9.20162 0.3305971 0.1231034 0.002449421
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 2      -193.6      391      3.538533      30.92356      -195.6
## WeibAICFlex BinomialAIC
## 2      395      394.9773
##          R      t0      n      G
## 2 0.002766436 56.71641 9.47557 0.1494377
```

M13



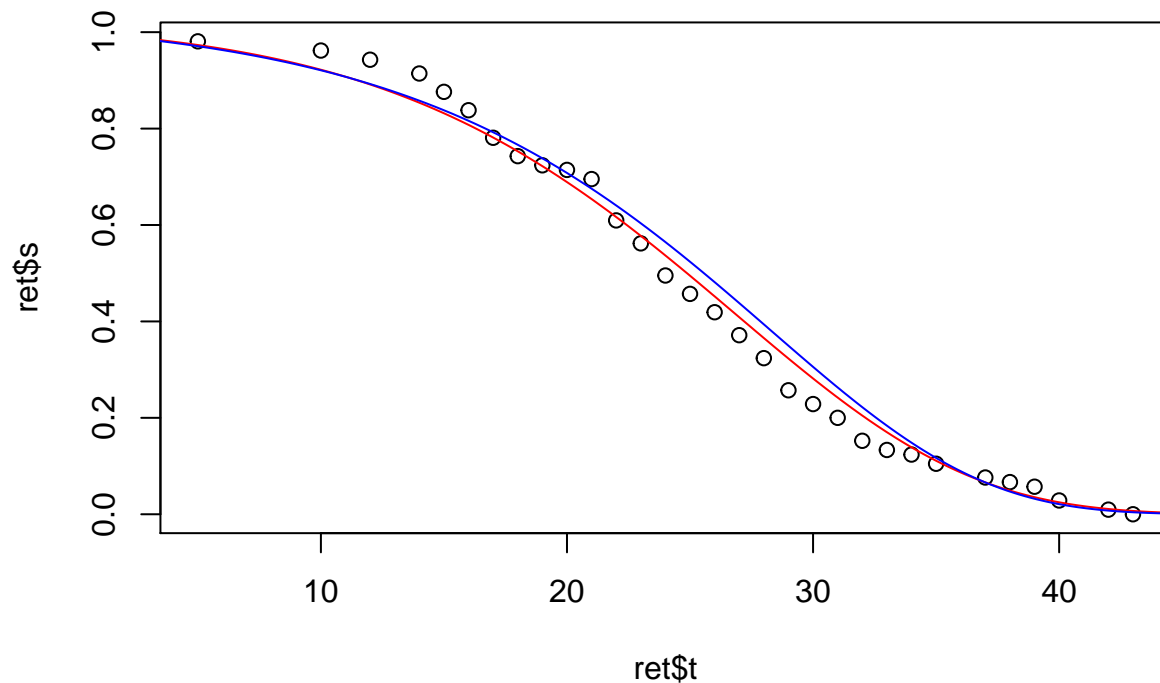
```
## my.strains samplesize      R      t0      n      G
## 3      M13           70 0.003007138 56.81672 9.648416 0.1522161
## longfilename  avgLS    stdLS      CV GompGFlex  GompRFlex
## 3  rls/M13.csv 26.54286 9.142488 0.3444425 0.122849 0.002926113
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 3      -250.1      504      3.114056      29.26548      -259.1
## WeibAICFlex BinomialAIC
## 3      522      508.0275
##      R      t0      n      G
## 3 0.003007138 56.81672 9.648416 0.1522161
```

M14



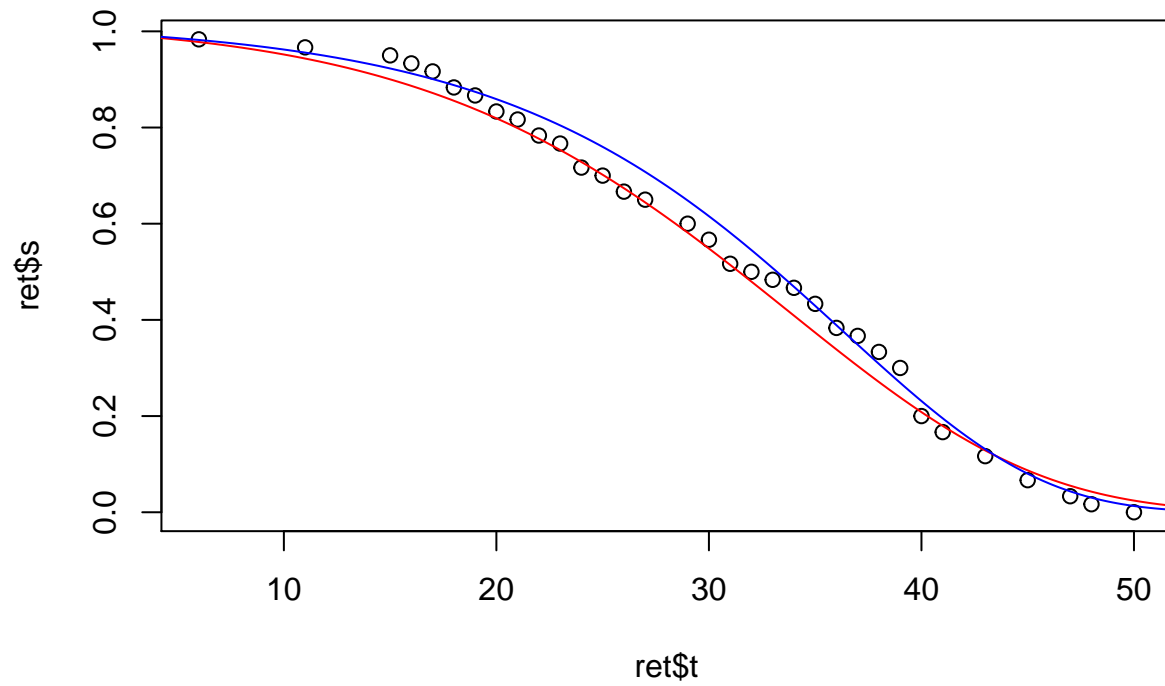
```
## my.strains samplesize      R      t0      n      G longfilename
## 4      M14           60 0.002486912 76.42752 9.18057 0.107037 rls/M14.csv
## avgLS   stdLS       CV GompGFlex GompRFlex GompLogLikFlex
## 4 36.55 12.80416 0.3503191 0.09148573 0.00195105      -233.7
## GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex WeibAICFlex
## 4      471      3.361407      40.80155      -236.8      478
## BinomialAIC
## 4      476.73
##      R      t0      n      G
## 4 0.002486912 76.42752 9.18057 0.107037
```


M2-8



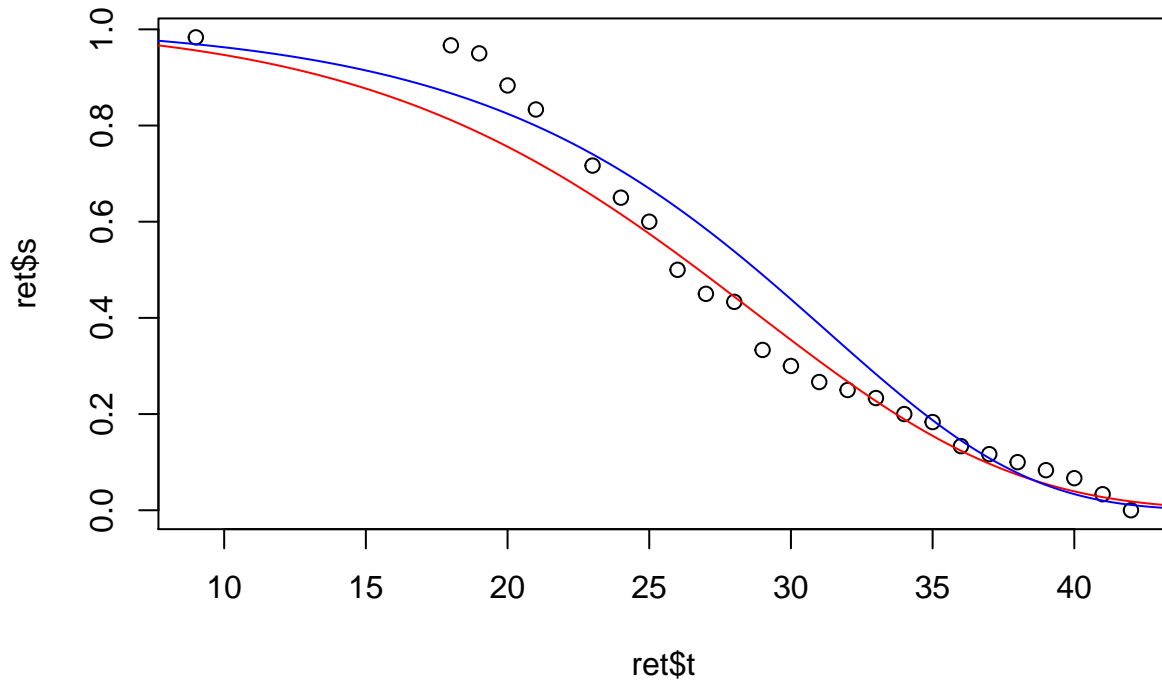
```
## my.strains samplesize      R      t0      n      G
## 5      M2-8          105 0.003663387 60.10209 10.06826 0.150881
## longfilename  avgLS    stdLS      CV GompGFlex  GompRFlex
## 5 rls/M2-8.csv 24.80952 8.133614 0.3278424 0.1160839 0.004361288
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 5      -373.3      751      3.371375      27.61417      -368.5
## WeibAICFlex BinomialAIC
## 5      741      749.3801
##      R      t0      n      G
## 5 0.003663387 60.10209 10.06826 0.150881
```

M22



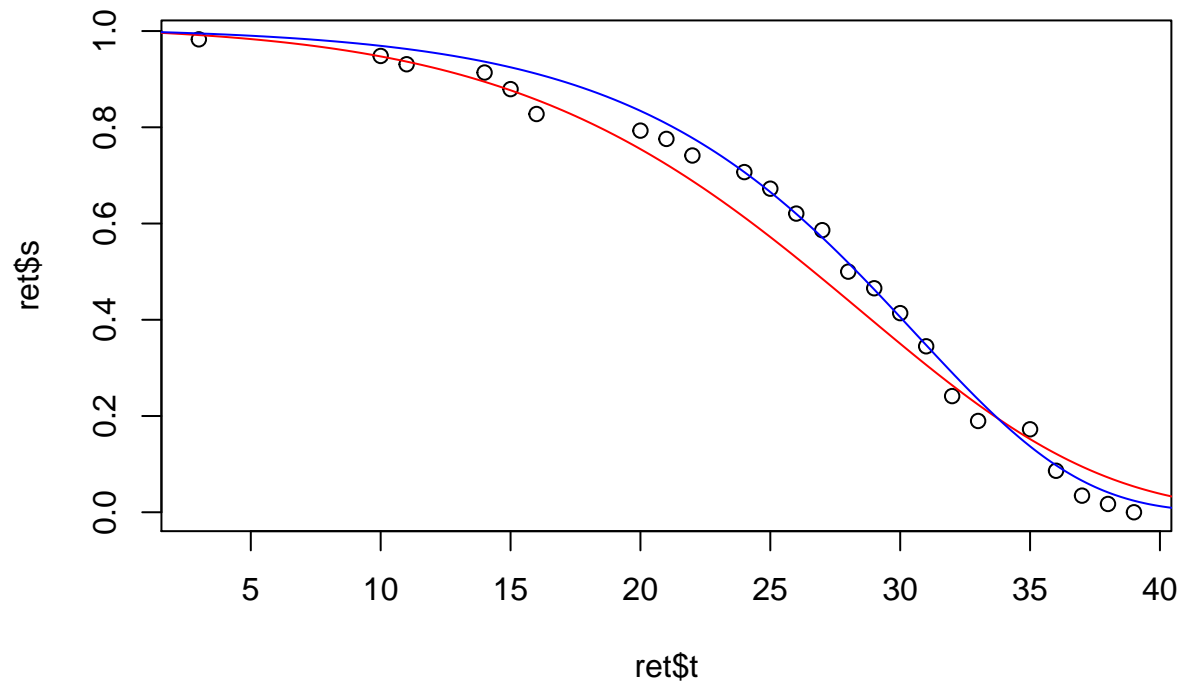
```
## my.strains samplesize      R      t0      n      G
## 6      M22           60 0.002509604 64.81929 9.400784 0.1296031
## longfilename  avgLS    stdLS      CV GompGFlex  GompRFlex
## 6  rls/M22.csv 31.83333 10.27118 0.3226549 0.1077899 0.002145453
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 6      -222.9      450      3.596392      35.37118      -223.9
## WeibAICFlex BinomialAIC
## 6      452      453.6004
##      R      t0      n      G
## 6 0.002509604 64.81929 9.400784 0.1296031
```

M32



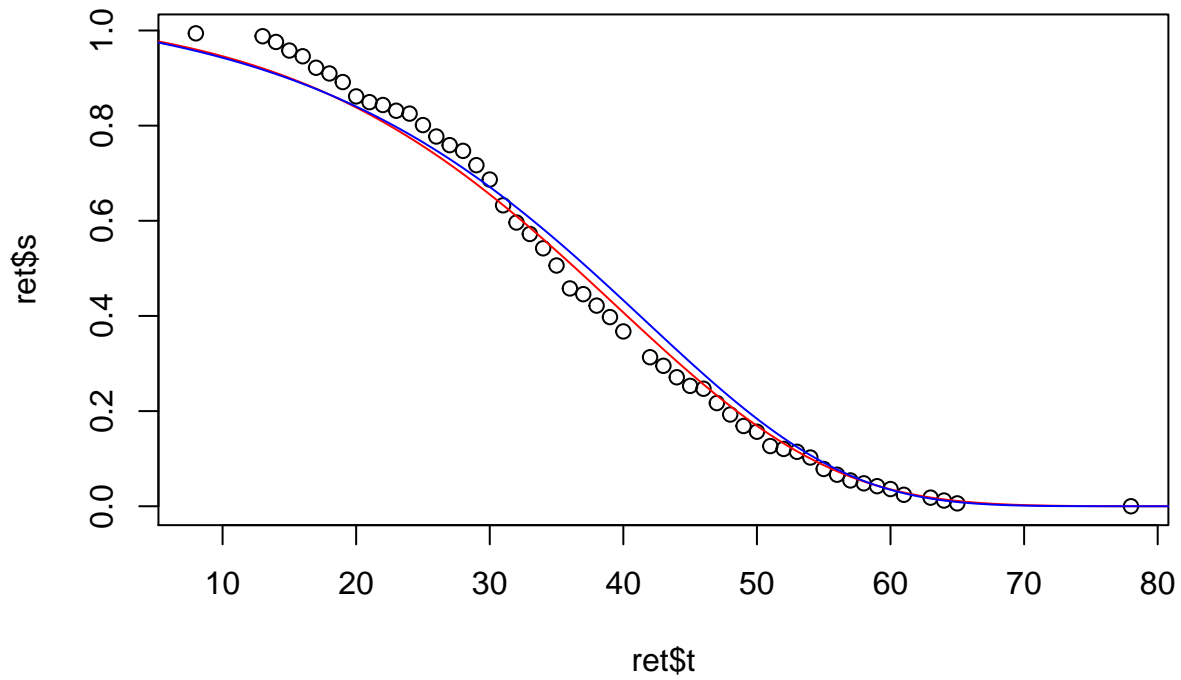
```
## my.strains samplesize      R      t0      n      G
## 7      M32           60 0.002209031 49.67204 9.55655 0.1722609
## longfilename  avgLS    stdLS      CV GompGFlex  GompRFlex
## 7  rls/M32.csv 27.96667 6.888868 0.2463242 0.1404328 0.001738458
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 7      -205.6      415      4.419104      30.62828      -201.4
## WeibAICFlex BinomialAIC
## 7      407      417.9426
##      R      t0      n      G
## 7 0.002209031 49.67204 9.55655 0.1722609
```

M34



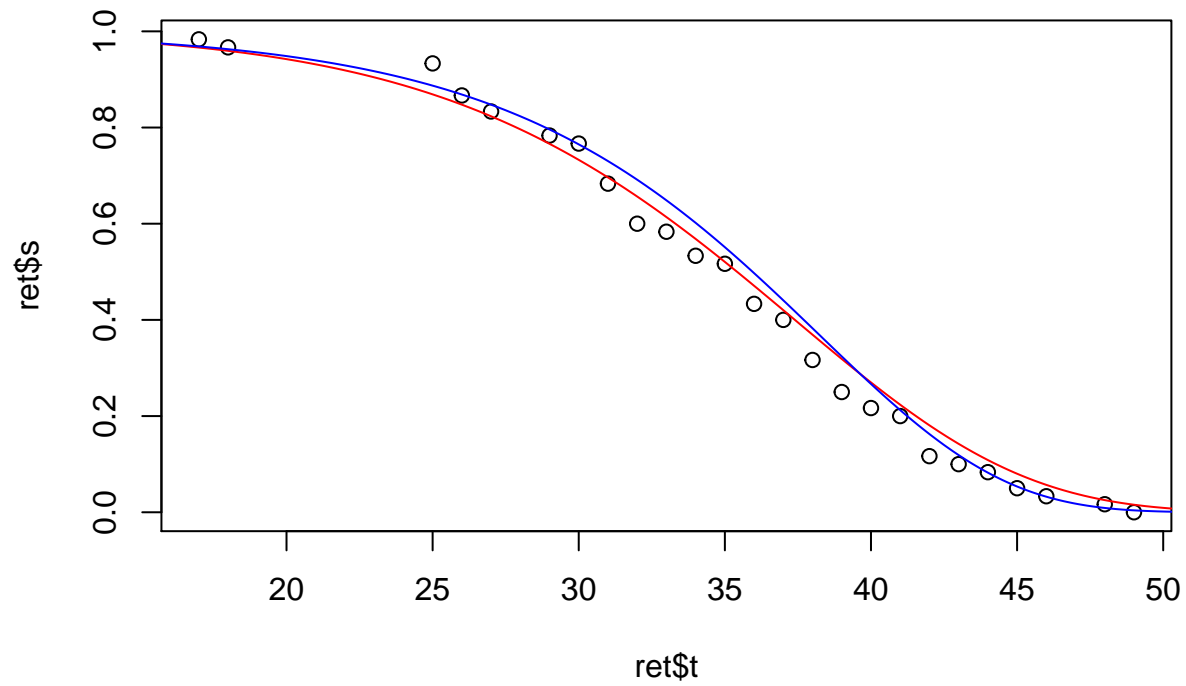
```
## my.strains samplesize      R      t0      n      G
## 8      M34           58 0.002123117 44.49124 8.940977 0.1784841
## longfilename  avgLS  stdLS      CV GompGFlex  GompRFlex
## 8  rls/M34.csv 27.01724 8.20674 0.3037593 0.156928 0.001289647
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 8      -198      400      3.985745      29.77062      -203.7
## WeibAICFlex BinomialAIC
## 8      411      408.2894
##      R      t0      n      G
## 8 0.002123117 44.49124 8.940977 0.1784841
```

M5



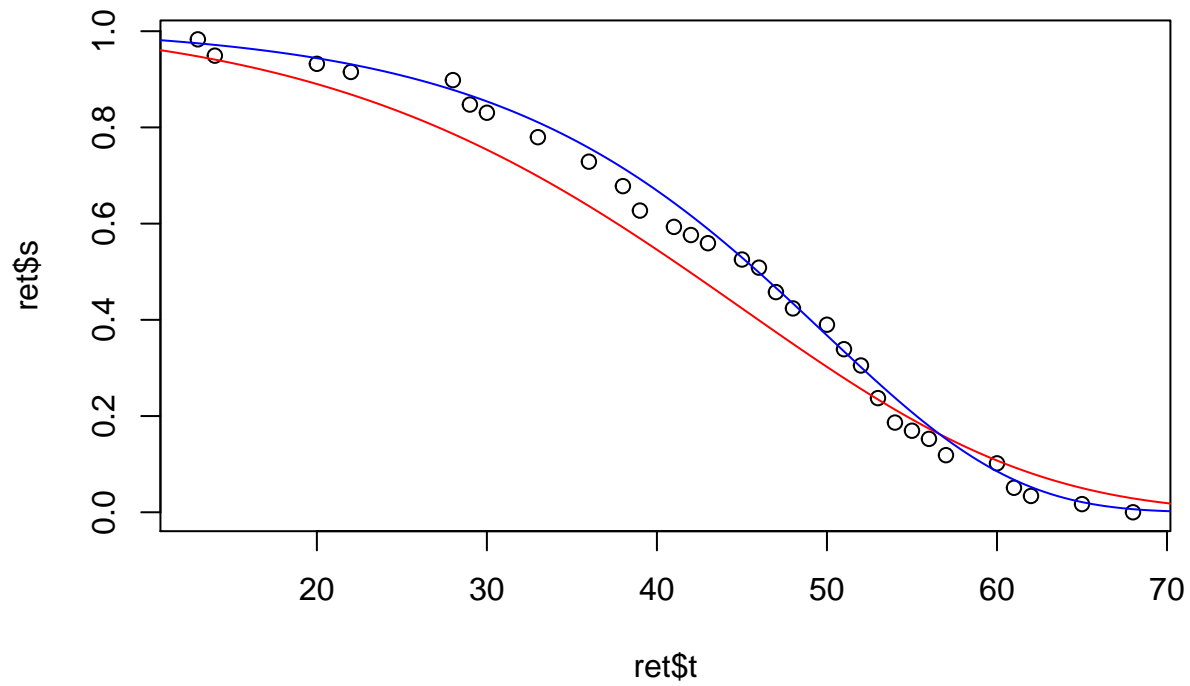
```
## my.strains samplesize      R      t0      n      G
## 9      M5          166 0.00358755 104.6057 9.903523 0.08511506
## longfilename  avgLS    stdLS      CV  GompGFlex  GompRFlex
## 9  rls/M5.csv 36.62651 12.93875 0.3532618 0.06684902 0.00414725
## GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 9      -670.1      1344      3.098392      40.98709      -658.3
## WeibAICFlex BinomialAIC
## 9      1321      1339.875
##      R      t0      n      G
## 9 0.00358755 104.6057 9.903523 0.08511506
```

M8



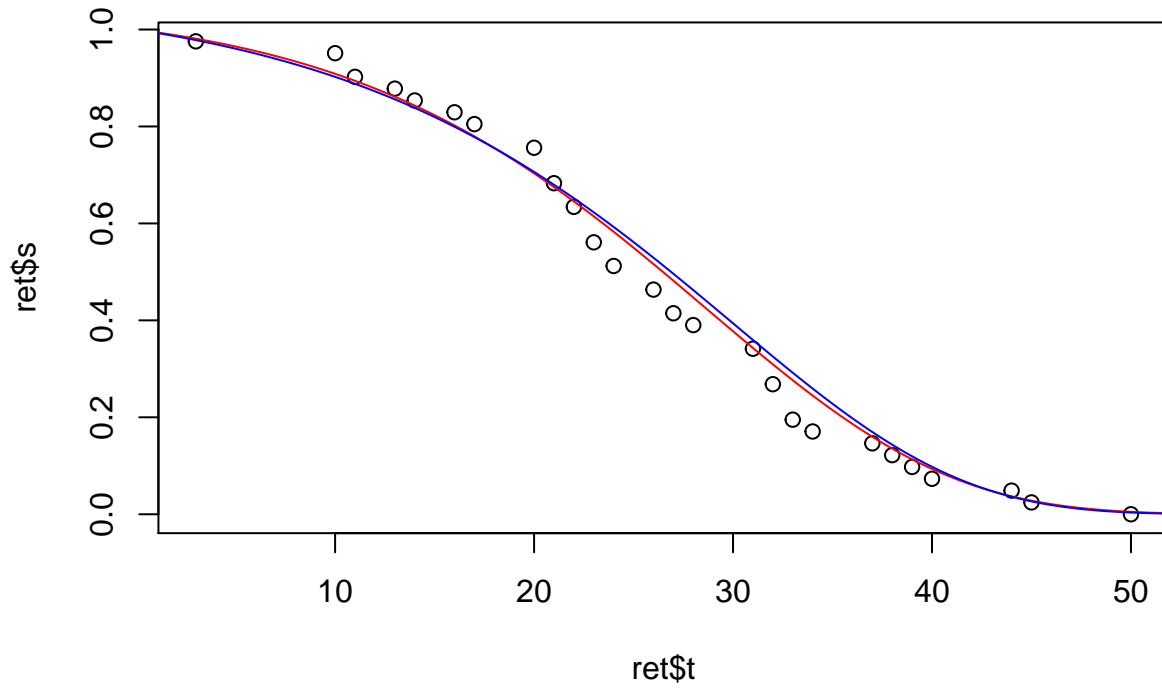
```
## my.strains samplesize      R      t0      n      G longfilename
## 10      M8           60 0.0002276221 45.20362 11.5 0.2322823 rls/M8.csv
##      avgLS      stdLS      CV GompGFlex      GompRFlex GompLogLikFlex
## 10 34.93333 6.905823 0.1976858 0.1588883 0.0003653141      -201.9
##      GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex WeibAICFlex
## 10      408      5.872213      37.70008      -200.1      404
##      BinomialAIC
## 10      409.6617
##      R      t0      n      G
## 10 0.0002276221 45.20362 11.5 0.2322823
```

RM112N



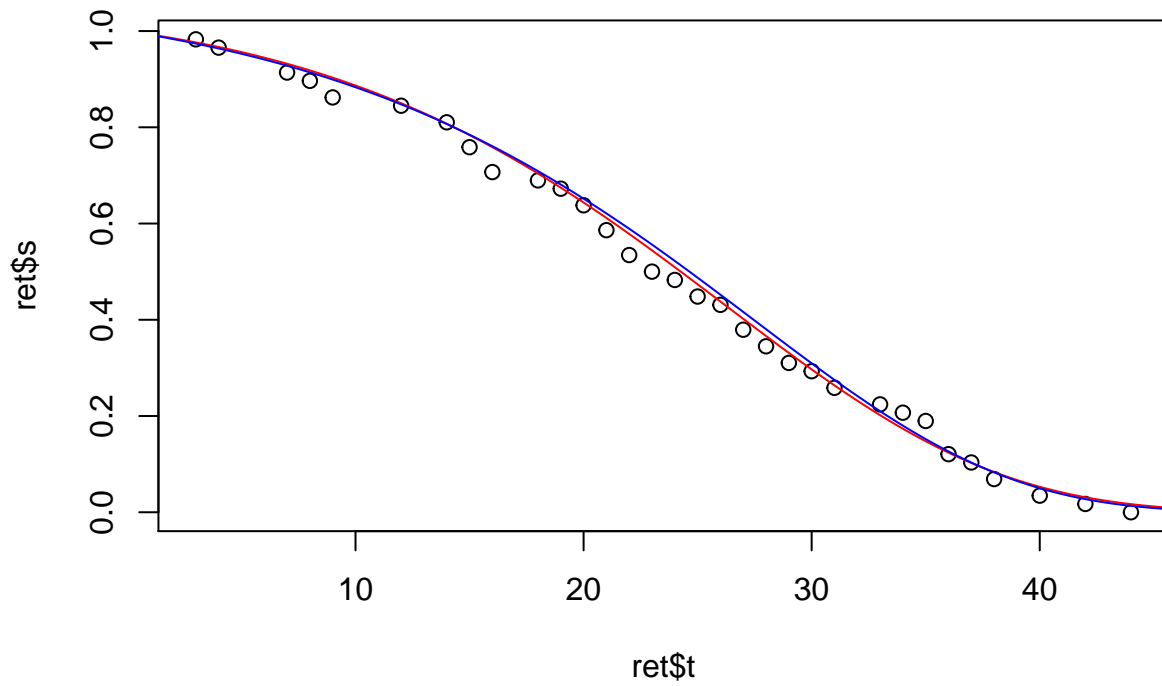
```
## my.strains samplesize      R      t0      n      G
## 11  RM112N           59 0.002236752 78.29925 8.184119 0.09175208
##      longfilename  avgLS   stdLS      CV  GompGFlex  GompRFlex
## 11 rls/RM112N.csv 44.0678 13.00645 0.2951464 0.08938574 0.00103633
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 11      -232          468      4.038465      48.63895      -233.7
##      WeibAICFlex BinomialAIC
## 11      471      478.043
##      R      t0      n      G
## 11 0.002236752 78.29925 8.184119 0.09175208
```

S288c



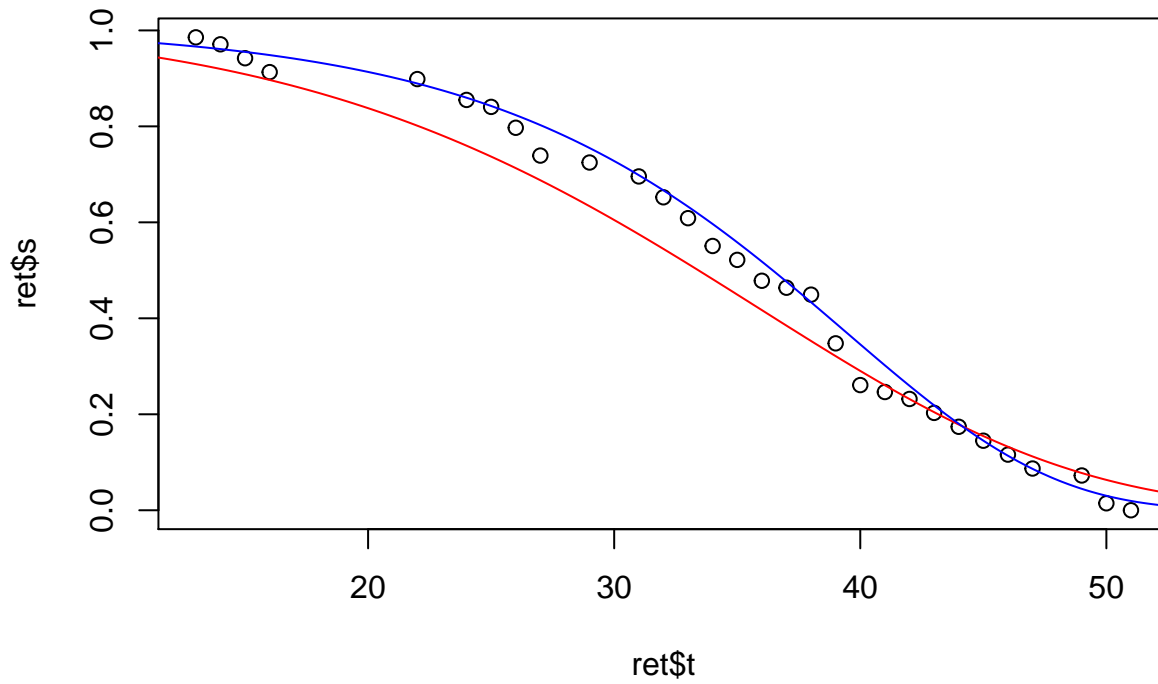
```
## my.strains samplesize      R      t0      n      G
## 12      S288c          41 0.005361188 80.43622 9.959994 0.1113925
##      longfilename    avgLS    stdLS      CV GompGFlex  GompRFlex
## 12 rls/S288c.csv 26.26829 10.25433 0.390369 0.08686882 0.006451205
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 12      -154.2          312      2.792964      29.42942      -153.3
##      WeibAICFlex BinomialAIC
## 12          311      313.4152
##      R      t0      n      G
## 12 0.005361188 80.43622 9.959994 0.1113925
```


SGU57



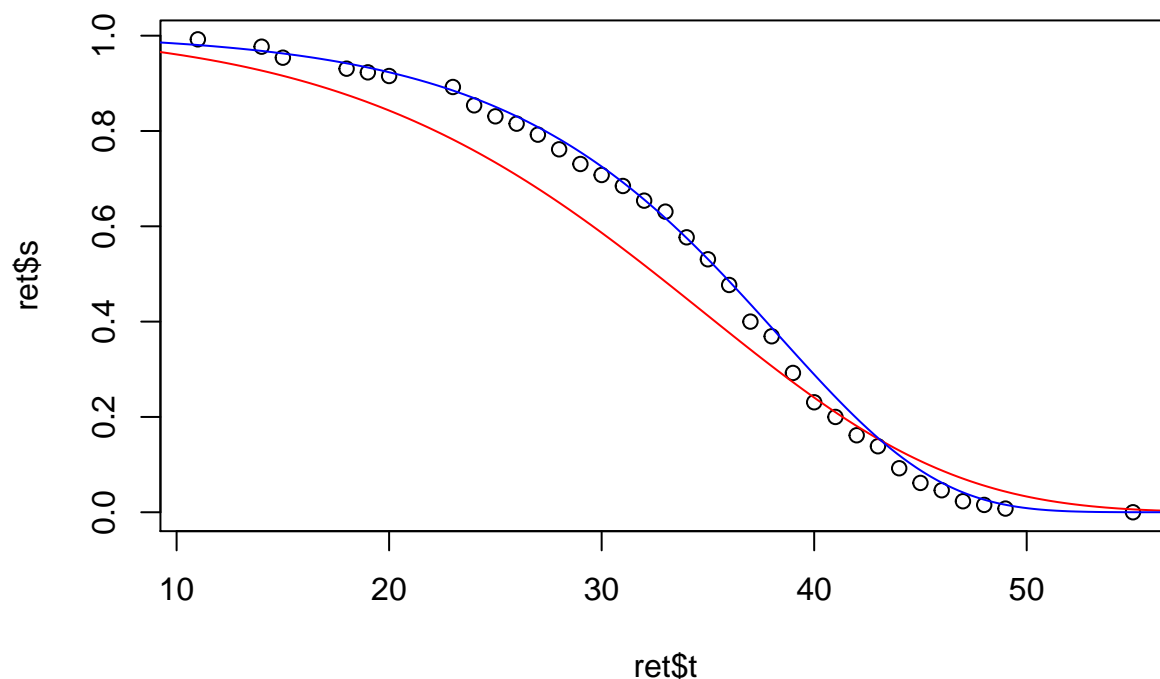
```
## my.strains samplesize      R      t0      n      G
## 13      SGU57          58 0.006725643 78.02415 9.696555 0.1114598
##      longfilename  avgLS  stdLS      CV GompGFlex  GompRFlex
## 13 rls/SGU57.csv 23.86207 10.5389 0.441659 0.08956763 0.007674863
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 13      -216.5          437      2.455703      26.8591      -218.2
##      WeibAICFlex BinomialAIC
## 13      440      439.2137
##      R      t0      n      G
## 13 0.006725643 78.02415 9.696555 0.1114598
```

YPS128



```
##      my.strains samplesize      R      t0      n      G
## 14      YPS128           69 0.002437805 58.96767 8.260993 0.1231351
##      longfilename avgLS      stdLS      CV GompGFlex  GompRFlex
## 14 rls/YPS128.csv    35 9.719598 0.2777028 0.1186601 0.001104155
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 14      -252.2          508      4.294122      38.5299      -253
##      WeibAICFlex BinomialAIC
## 14      510      519.8934
##      R      t0      n      G
## 14 0.002437805 58.96767 8.260993 0.1231351
```

YPS163



```
##      my.strains samplesize      R      t0      n      G
## 15      YPS163          130 0.001915118 52.24318 8.437491 0.1423629
##      longfilename    avgLS    stdLS      CV GompGFlex  GompRFlex
## 15 rls/YPS163.csv 34.43077 8.591449 0.2495282 0.1338728 0.0007889455
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 15      -459.6          923      4.825655      37.62671      -460.2
##      WeibAICFlex BinomialAIC
## 15          924      946.1981
##      R      t0      n      G
## 15 0.001915118 52.24318 8.437491 0.1423629
```

#show the report

report

```
##      my.strains samplesize      R      t0      n      G
## 1      101S          85 0.0019760147 52.08681 9.174228 0.15693470
## 2      M1-2          54 0.0027664362 56.71641 9.475570 0.14943770
## 3      M13          70 0.0030071379 56.81672 9.648416 0.15221605
## 4      M14          60 0.0024869119 76.42752 9.180570 0.10703698
## 5      M2-8         105 0.0036633868 60.10209 10.068263 0.15088099
## 6      M22          60 0.0025096036 64.81929 9.400784 0.12960313
## 7      M32          60 0.0022090312 49.67204 9.556550 0.17226088
## 8      M34          58 0.0021231171 44.49124 8.940977 0.17848407
## 9      M5          166 0.0035875496 104.60572 9.903523 0.08511506
## 10     M8          60 0.0002276221 45.20362 11.500000 0.23228229
## 11     RM112N       59 0.0022367518 78.29925 8.184119 0.09175208
## 12     S288c        41 0.0053611878 80.43622 9.959994 0.11139252
## 13     SGU57        58 0.0067256427 78.02415 9.696555 0.11145978
## 14     YPS128       69 0.0024378049 58.96767 8.260993 0.12313514
## 15     YPS163      130 0.0019151177 52.24318 8.437491 0.14236291
```

```
##      longfilename      avgLS      stdLS      CV      GompGFlex      GompRFlex
## 1      rls/101S.csv 31.34118  7.512772 0.2397093 0.13407292 0.0012654562
## 2      rls/M1-2.csv 27.83333  9.201620 0.3305971 0.12310344 0.0024494207
## 3      rls/M13.csv 26.54286  9.142488 0.3444425 0.12284897 0.0029261133
## 4      rls/M14.csv 36.55000 12.804164 0.3503191 0.09148573 0.0019510502
## 5      rls/M2-8.csv 24.80952  8.133614 0.3278424 0.11608390 0.0043612883
## 6      rls/M22.csv 31.83333 10.271182 0.3226549 0.10778993 0.0021454528
## 7      rls/M32.csv 27.96667  6.888868 0.2463242 0.14043277 0.0017384578
## 8      rls/M34.csv 27.01724  8.206740 0.3037593 0.15692797 0.0012896468
## 9      rls/M5.csv 36.62651 12.938747 0.3532618 0.06684902 0.0041472501
## 10     rls/M8.csv 34.93333  6.905823 0.1976858 0.15888831 0.0003653141
## 11 rls/RM112N.csv 44.06780 13.006450 0.2951464 0.08938574 0.0010363304
## 12 rls/S288c.csv 26.26829 10.254327 0.3903690 0.08686882 0.0064512047
## 13 rls/SGU57.csv 23.86207 10.538898 0.4416590 0.08956763 0.0076748634
## 14 rls/YPS128.csv 35.00000  9.719598 0.2777028 0.11866014 0.0011041546
## 15 rls/YPS163.csv 34.43077  8.591449 0.2495282 0.13387276 0.0007889455
##      GompLogLikFlex GompAICFlex WeibShapeFlex WeibRateFlex WeibLogLikFlex
## 1      -295.0          594      4.778139      34.15561      -291.0
## 2      -193.6          391      3.538533      30.92356      -195.6
## 3      -250.1          504      3.114056      29.26548      -259.1
## 4      -233.7          471      3.361407      40.80155      -236.8
## 5      -373.3          751      3.371375      27.61417      -368.5
## 6      -222.9          450      3.596392      35.37118      -223.9
## 7      -205.6          415      4.419104      30.62828      -201.4
## 8      -198.0          400      3.985745      29.77062      -203.7
## 9      -670.1        1344      3.098392      40.98709      -658.3
## 10     -201.9          408      5.872213      37.70008      -200.1
## 11     -232.0          468      4.038465      48.63895      -233.7
## 12     -154.2          312      2.792964      29.42942      -153.3
## 13     -216.5          437      2.455703      26.85910      -218.2
## 14     -252.2          508      4.294122      38.52990      -253.0
## 15     -459.6          923      4.825655      37.62671      -460.2
##      WeibAICFlex BinomialAIC
## 1          586      601.3715
## 2          395      394.9773
## 3          522      508.0275
## 4          478      476.7300
## 5          741      749.3801
## 6          452      453.6004
## 7          407      417.9426
## 8          411      408.2894
## 9         1321     1339.8746
## 10         404      409.6617
## 11         471      478.0430
## 12         311      313.4152
## 13         440      439.2137
## 14         510      519.8934
## 15         924      946.1981
```

Calculate lambda based on $t_0 = (1-p)/(p \text{ Lambda})$ So, $1/\text{lambda} = t_0 * p / (1-p)$

```
p = 0.7
report$One.over.lambdaP07 = report$t0 * p / (1-p)
report[, c("t0", "One.over.lambdaP07")]
```

```
##           t0 One.over.lambdaP07
## 1  52.08681      121.5359
## 2  56.71641      132.3383
## 3  56.81672      132.5723
## 4  76.42752      178.3309
## 5  60.10209      140.2382
## 6  64.81929      151.2450
## 7  49.67204      115.9014
## 8  44.49124      103.8129
## 9 104.60572      244.0800
##10  45.20362      105.4751
##11  78.29925      182.6982
##12  80.43622      187.6845
##13  78.02415      182.0564
##14  58.96767      137.5912
##15  52.24318      121.9008
```

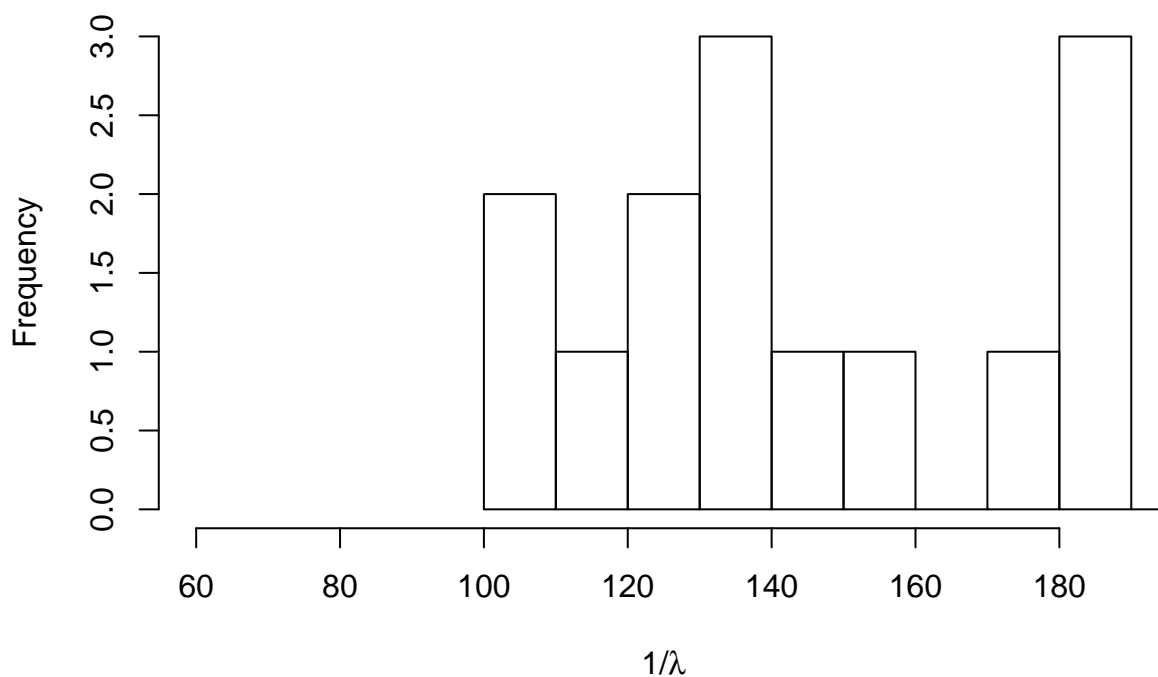
```
summary( report[, c("t0", "One.over.lambdaP07")])
```

```
##           t0           One.over.lambdaP07
## Min.      : 44.49      Min.       :103.8
## 1st Qu.: 52.16      1st Qu.:121.7
## Median : 58.97      Median :137.6
## Mean      : 63.93      Mean       :149.2
## 3rd Qu.: 77.23      3rd Qu.:180.2
## Max.      :104.61     Max.       :244.1
```

Histogram of $1/\lambda$ for $p=0.7$

```
hist(report$One.over.lambdaP07, br=20, xlim=c(60,190), xlab=expression(paste("1/",lambda) ), main=expression(paste("Histogram of 1/",lambda))))
```

Histogram of $1/\lambda$ when $p=0.7$



```
p = 0.9
report$One.over.lambdaP09 = report$t0 * p / (1-p)
report[, c("t0", "One.over.lambdaP09")]
```

```
##           t0 One.over.lambdaP09
## 1  52.08681      468.7813
## 2  56.71641      510.4477
## 3  56.81672      511.3504
## 4  76.42752      687.8476
## 5  60.10209      540.9188
## 6  64.81929      583.3737
## 7  49.67204      447.0484
## 8  44.49124      400.4211
## 9 104.60572      941.4515
##10  45.20362      406.8326
##11  78.29925      704.6932
##12  80.43622      723.9260
##13  78.02415      702.2174
##14  58.96767      530.7091
##15  52.24318      470.1886
```

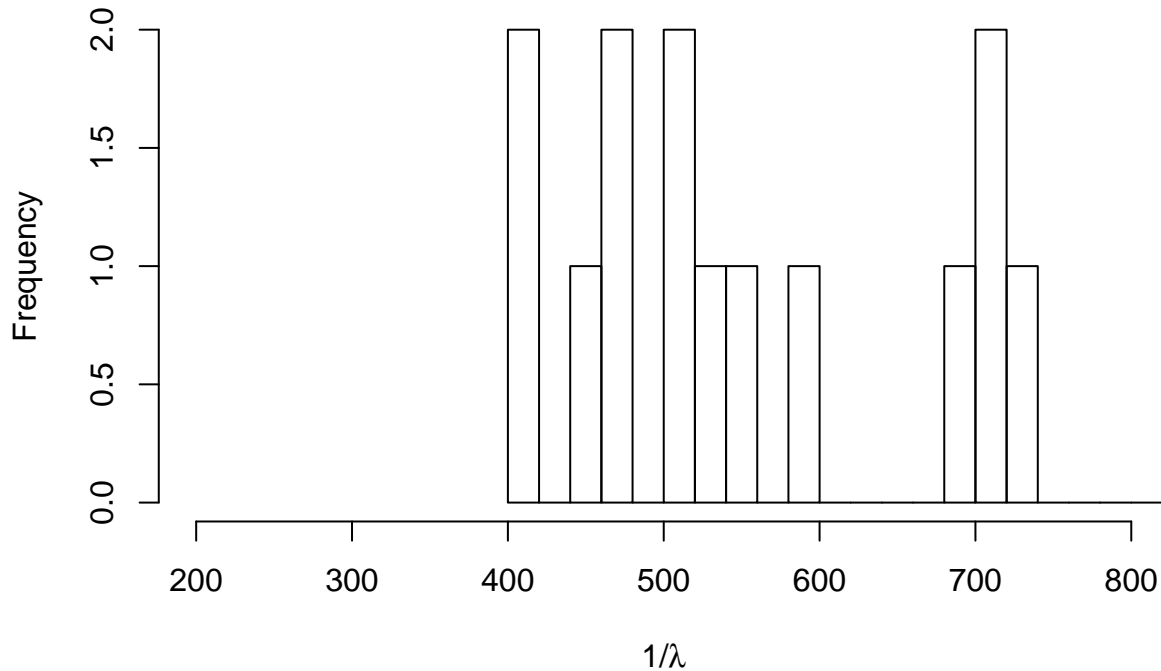
```
summary( report[, c("t0", "One.over.lambdaP09")] )
```

```
##           t0           One.over.lambdaP09
## Min.      : 44.49      Min.      :400.4
## 1st Qu.: 52.16      1st Qu.:469.5
## Median : 58.97      Median :530.7
## Mean    : 63.93      Mean    :575.3
## 3rd Qu.: 77.23      3rd Qu.:695.0
## Max.    :104.61      Max.     :941.5
```

Histogram 1/lambda, p=0.9

```
hist(report$One.over.lambdaP09, br=20, xlim=c(200,800), xlab=expression(paste("1/",lambda) ), main=expr
```

Histogram of $1/\lambda$ when $p=0.9$



Explorative analysis

```
# my.strains=c("101S", "BY4743", "M1-2", "M13", "M14", "M2-8", "M22", "M32", "M34", "M5", "M8", "RM112N", "S288c",
#my.strains=c("101S", "M1-2", "M13", "M14", "M2-8", "M22", "M32", "M34", "M5", "M8", "RM112N", "S288c", "SGU57", "
report2 = report
summary(lm(log10(report2$GompRFlex) ~ report2$GompGFlex))

##
## Call:
## lm(formula = log10(report2$GompRFlex) ~ report2$GompGFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50796 -0.22010 -0.03155  0.20504  0.36320
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.6976     0.3278  -5.179  0.000177 ***
## report2$GompGFlex -8.7148     2.7614  -3.156  0.007585 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2789 on 13 degrees of freedom
## Multiple R-squared:  0.4338, Adjusted R-squared:  0.3902
## F-statistic:  9.96 on 1 and 13 DF,  p-value: 0.007585
```

```
summary(lm(log10(report2$R) ~ report2$GompGFlex))
```

```
##
## Call:
## lm(formula = log10(report2$R) ~ report2$GompGFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68974 -0.11029  0.05577  0.15224  0.26476
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.7169     0.3007  -5.710 7.18e-05 ***
## report2$GompGFlex -7.7799     2.5335  -3.071  0.00893 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2559 on 13 degrees of freedom
## Multiple R-squared:  0.4204, Adjusted R-squared:  0.3758
## F-statistic:  9.43 on 1 and 13 DF,  p-value: 0.008935
```

```
summary(lm(log10(report2$R) ~ report2$G)) #G from t0 and n
```

```
##
## Call:
## lm(formula = log10(report2$R) ~ report2$G)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43986 -0.13556  0.02285  0.17206  0.26760
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.7359     0.2302  -7.543 4.23e-06 ***
## report2$G     -6.3156     1.5944  -3.961  0.00163 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2262 on 13 degrees of freedom
## Multiple R-squared:  0.5469, Adjusted R-squared:  0.512
## F-statistic: 15.69 on 1 and 13 DF,  p-value: 0.001627
```

```
summary(lm(report2$GompGFlex ~ report2$G)) #good agreement btween GFlex and G from binomial modeling
```

```
##
## Call:
## lm(formula = report2$GompGFlex ~ report2$G)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.017814 -0.007406 -0.001218  0.005985  0.016283
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.02400     0.01092   2.198  0.0467 *
```



```
## report2$G      0.65739      0.07566      8.689 8.95e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01074 on 13 degrees of freedom
## Multiple R-squared:  0.8531, Adjusted R-squared:  0.8418
## F-statistic: 75.49 on 1 and 13 DF,  p-value: 8.954e-07
summary(lm(log10(report2$R) ~ report2$t0)) #G from t0 and n

##
## Call:
## lm(formula = log10(report2$R) ~ report2$t0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.83042 -0.05424  0.05629  0.15827  0.29893
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.282258   0.302783 -10.840    7e-08 ***
## report2$t0   0.010395   0.004595   2.262    0.0414 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2847 on 13 degrees of freedom
## Multiple R-squared:  0.2825, Adjusted R-squared:  0.2273
## F-statistic: 5.119 on 1 and 13 DF,  p-value: 0.04145
summary(lm(log10(report2$R) ~ report2$n)) #G from t0 and n

##
## Call:
## lm(formula = log10(report2$R) ~ report2$n)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.71779 -0.14748 -0.01871  0.18607  0.48557
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.2214      0.9494  -1.286   0.221
## report2$n    -0.1481      0.1004  -1.476   0.164
##
## Residual standard error: 0.3111 on 13 degrees of freedom
## Multiple R-squared:  0.1435, Adjusted R-squared:  0.07765
## F-statistic: 2.179 on 1 and 13 DF,  p-value: 0.1637
```

Strehler-Mildvan correlation in natural isolates

```
plot( log10(report2$R) ~ report2$G, pch=19, col='red', xlab='G', ylab='log10(R)',
      xlim=c(0.07, 0.22), ylim=c(-2.9, -2.1),
      main="Strehler-Mildvan correlation in yeast wild isolates")
```

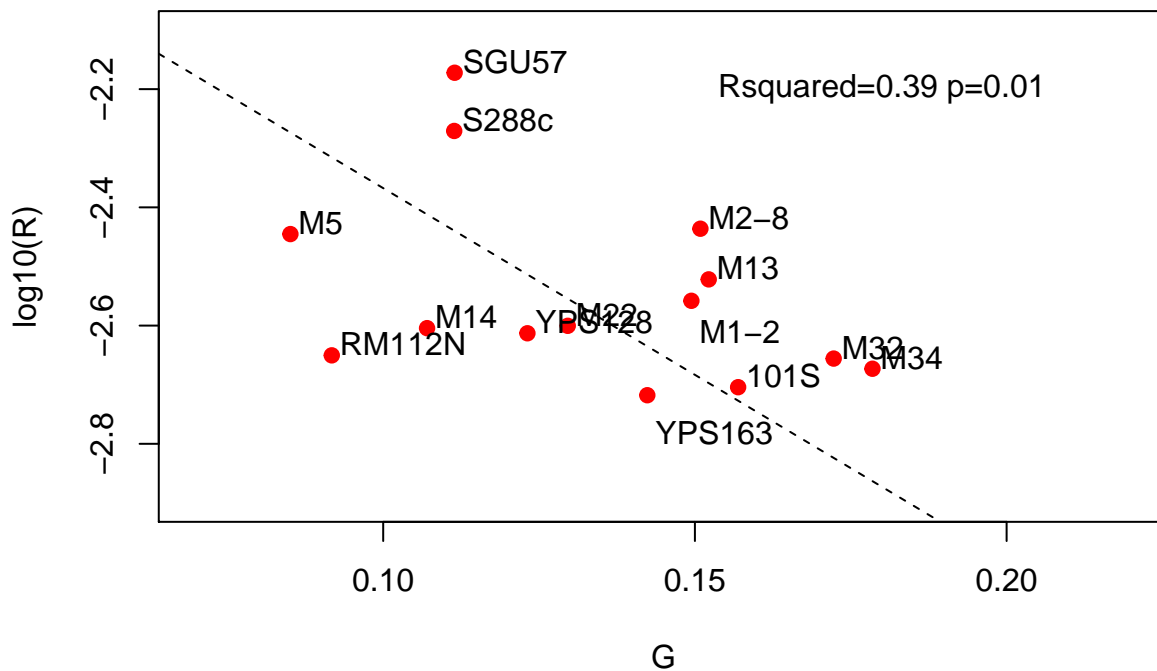
```

m = lm(log10(report2$R) ~ report2$G)
abline(m, col='black', lty=2);
text(0.18, -2.2, "Rsquared=0.39 p=0.01")

my.x = report2$G +0.0013 ; my.y = log10(report2$R) ;
names(my.x) = report2$my.strains; names(my.y) = report2$my.strains;
my.y[c("YPS163", "M1-2" )]=c(-2.8, -2.63)
text(my.x,my.y, report2$my.strains, adj=c(0,0) )

```

Strehler–Mildvan correlation in yeast wild isolates



```
summary(lm(log10(report2$R) ~ report2$G)) #G from t0 and n
```

```

##
## Call:
## lm(formula = log10(report2$R) ~ report2$G)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43986 -0.13556  0.02285  0.17206  0.26760
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.7359     0.2302  -7.543 4.23e-06 ***
## report2$G    -6.3156     1.5944  -3.961 0.00163 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2262 on 13 degrees of freedom
## Multiple R-squared:  0.5469, Adjusted R-squared:  0.512
## F-statistic: 15.69 on 1 and 13 DF, p-value: 0.001627

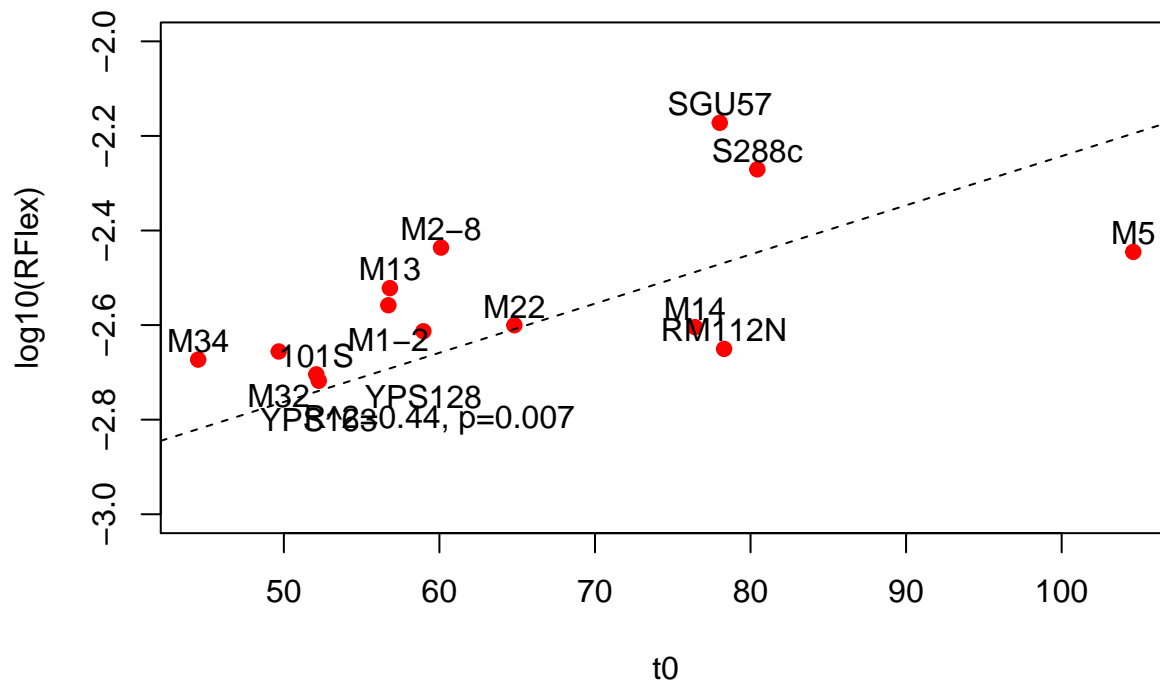
```

```
summary(lm(log10(report2$GompRFlex) ~ report2$GompGFlex)) #G from t0 and n
```

```
##
## Call:
## lm(formula = log10(report2$GompRFlex) ~ report2$GompGFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50796 -0.22010 -0.03155  0.20504  0.36320
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.6976     0.3278  -5.179 0.000177 ***
## report2$GompGFlex -8.7148     2.7614  -3.156 0.007585 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2789 on 13 degrees of freedom
## Multiple R-squared:  0.4338, Adjusted R-squared:  0.3902
## F-statistic:  9.96 on 1 and 13 DF,  p-value: 0.007585
```

plot log10(R) ~ t0

```
plot( log10(report2$R) ~ report2$t0, col='red', pch=19, xlab='t0', ylab='log10(RFlex)', ylim=c(-3,-2))
my.x = report2$t0; my.y = log10(report2$R) + 0.04;
names(my.x) = report2$my.strains; names(my.y) = report2$my.strains;
my.y[c("M32", "YPS163", "YPS128", "M1-2" )]=c(-2.75, -2.8, -2.75, -2.63)
text(my.x,my.y, report2$my.strains )
m = lm( log10(report2$R) ~ report2$t0 )
abline (m, col='black', lty=2);
text( 60, -2.8, "R^2=0.44, p=0.007")
```



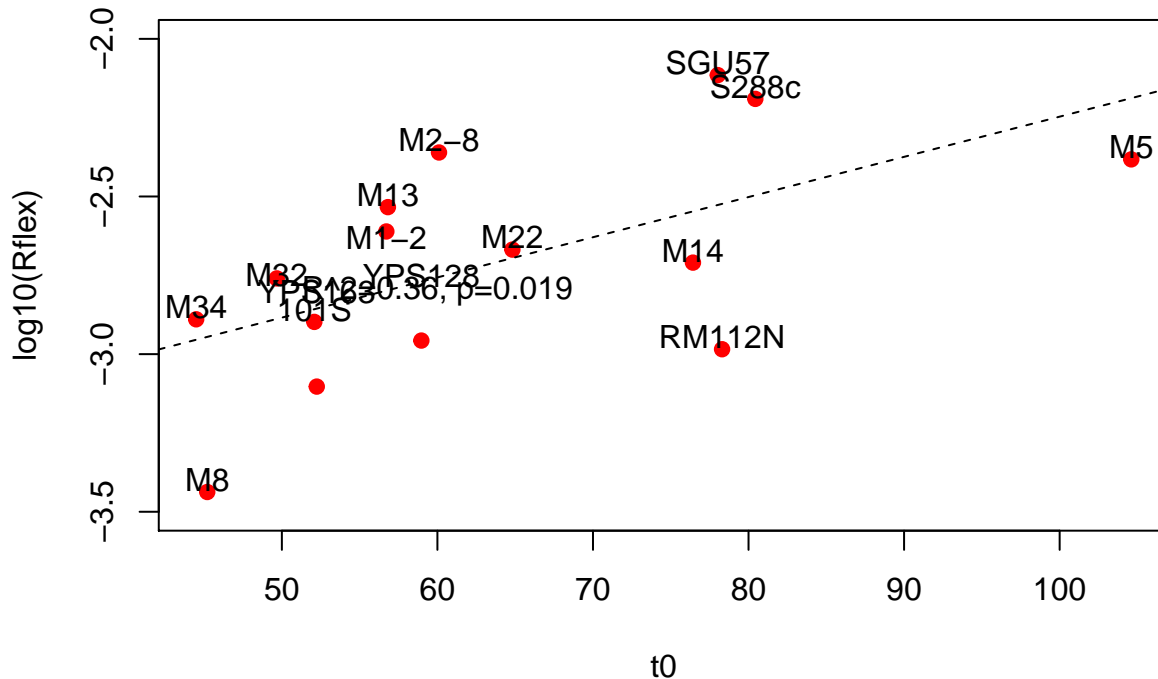
```
summary(m)
```

```
##
## Call:
## lm(formula = log10(report2$R) ~ report2$t0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.83042 -0.05424  0.05629  0.15827  0.29893
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.282258   0.302783  -10.840   7e-08 ***
## report2$t0   0.010395   0.004595   2.262   0.0414 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2847 on 13 degrees of freedom
## Multiple R-squared:  0.2825, Adjusted R-squared:  0.2273
## F-statistic: 5.119 on 1 and 13 DF,  p-value: 0.04145
```

plot $\log_{10}(\text{GompRFlex}) \sim t_0$

```
plot( log10(report2$GompRFlex) ~ report2$t0, col='red', pch=19, xlab='t0', ylab='log10(Rflex)', ylim=c(
my.x = report2$t0; my.y = log10(report2$GompRFlex) + 0.04;
names(my.x) = report2$my.strains; names(my.y) = report2$my.strains;
my.y[c("M32", "YPS163", "YPS128", "M1-2" )]=c(-2.75, -2.8, -2.75, -2.63)
text(my.x,my.y, report2$my.strains )
m = lm( log10(report2$GompRFlex) ~ report2$t0 )
abline (m, col='black', lty=2);
```

```
text( 60, -2.8, "R^2=0.36, p=0.019")
```



```
summary(m)
```

```
##
## Call:
## lm(formula = log10(report2$GompRFlex) ~ report2$t0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49200 -0.19063  0.02679  0.22561  0.41201
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.521584   0.317954 -11.076 5.43e-08 ***
## report2$t0   0.012748   0.004825   2.642  0.0203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.299 on 13 degrees of freedom
## Multiple R-squared:  0.3494, Adjusted R-squared:  0.2993
## F-statistic: 6.981 on 1 and 13 DF,  p-value: 0.02031
```

Could t_0 be the cause of logR-G correlation?

Partial regression on $G \sim \log(R) + t_0$.

```
summary(lm( log10(report2$GompRFlex) ~ report2$t0 + report2$GompGFlex )) #poor
```

```
##
## Call:
```

```
## lm(formula = log10(report2$GompRFlex) ~ report2$t0 + report2$GompGFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52445 -0.24582  0.05386  0.21803  0.34448
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.06152    2.29600   0.027   0.979
## report2$t0      -0.01329    0.01716  -0.774   0.454
## report2$GompGFlex -16.57147   10.52694  -1.574   0.141
##
## Residual standard error: 0.2833 on 12 degrees of freedom
## Multiple R-squared:  0.4607, Adjusted R-squared:  0.3709
## F-statistic: 5.126 on 2 and 12 DF,  p-value: 0.02459
summary(lm( log10(report2$GompGFlex) ~ report2$t0 + report2$GompRFlex ))#good
```

```
##
## Call:
## lm(formula = log10(report2$GompGFlex) ~ report2$t0 + report2$GompRFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.015355 -0.009478 -0.002220  0.002442  0.029280
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.5449827   0.0175062  -31.131 7.59e-13 ***
## report2$t0     -0.0062053   0.0003103  -19.998 1.40e-10 ***
## report2$GompRFlex -2.4328248   2.4033759   -1.012   0.331
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01578 on 12 degrees of freedom
## Multiple R-squared:  0.9813, Adjusted R-squared:  0.9782
## F-statistic: 314.6 on 2 and 12 DF,  p-value: 4.295e-11
#summary(lm( report2$G ~ log10(report2$R) + report2$t0 ))
#summary(lm( report2$G ~ report2$t0 + log10(report2$R) ))
summary(lm( report2$GompGFlex ~ report2$t0 + log10(report2$R) ))
```

```
##
## Call:
## lm(formula = report2$GompGFlex ~ report2$t0 + log10(report2$R))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.0067028 -0.0031232 -0.0002119  0.0006029  0.0129237
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.1643320   0.0208832   7.869 4.45e-06 ***
## report2$t0      -0.0014067   0.0001181  -11.914 5.24e-08 ***
## log10(report2$R) -0.0158091   0.0060373   -2.619  0.0224 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.006197 on 12 degrees of freedom
## Multiple R-squared:  0.9548, Adjusted R-squared:  0.9473
## F-statistic: 126.8 on 2 and 12 DF,  p-value: 8.506e-09
summary(lm( report2$GompGFlex ~ report2$t0 + log10(report2$GompRFlex)  ))

##
## Call:
## lm(formula = report2$GompGFlex ~ report2$t0 + log10(report2$GompRFlex))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.008587 -0.004575 -0.001389  0.001878  0.012961
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.1798481   0.0242995    7.401 8.26e-06 ***
## report2$t0       -0.0014394   0.0001415  -10.172 2.98e-07 ***
## log10(report2$GompRFlex) -0.0103287   0.0065612   -1.574  0.141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007073 on 12 degrees of freedom
## Multiple R-squared:  0.9412, Adjusted R-squared:  0.9313
## F-statistic: 95.96 on 2 and 12 DF,  p-value: 4.152e-08
```

Mediation test on Gflex \leftarrow t0 \leftarrow RFlex

Hong thinks this gives positive result.

```
library(mediation)

## Loading required package: MASS
## Loading required package: Matrix
## Loading required package: mvtnorm
## Loading required package: sandwich
## mediation: Causal Mediation Analysis
## Version: 4.4.6
set.seed(20170801)
report2$log10GompRFlex = log10(report2$GompRFlex)
med.fit = lm(t0 ~ log10GompRFlex, data=report2)
summary(med.fit)

##
## Call:
## lm(formula = t0 ~ log10GompRFlex, data = report2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -14.424 -10.842 -0.823 2.130 31.787
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    138.11      28.30   4.880 0.000301 ***
## log10GompRFlex  27.41      10.37   2.642 0.020314 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.86 on 13 degrees of freedom
## Multiple R-squared:  0.3494, Adjusted R-squared:  0.2993
## F-statistic: 6.981 on 1 and 13 DF, p-value: 0.02031
```

```
out.fit = lm(GompGFlex ~ t0 + log10GompRFlex, data=report2)
summary(out.fit)
```

```
##
## Call:
## lm(formula = GompGFlex ~ t0 + log10GompRFlex, data = report2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.008587 -0.004575 -0.001389  0.001878  0.012961
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.1798481  0.0242995   7.401 8.26e-06 ***
## t0             -0.0014394  0.0001415 -10.172 2.98e-07 ***
## log10GompRFlex -0.0103287  0.0065612  -1.574  0.141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007073 on 12 degrees of freedom
## Multiple R-squared:  0.9412, Adjusted R-squared:  0.9313
## F-statistic: 95.96 on 2 and 12 DF, p-value: 4.152e-08
```

```
med.out <- mediate(med.fit, out.fit, treat = "log10GompRFlex", mediator = "t0", robustSE = TRUE, sims =
summary(med.out)
```

```
##
## Causal Mediation Analysis
##
## Quasi-Bayesian Confidence Intervals
##
##             Estimate 95% CI Lower 95% CI Upper p-value
## ACME             -0.04304    -0.08032    -0.01 <2e-16 ***
## ADE              -0.00788    -0.02637     0.01  0.42
## Total Effect     -0.05092    -0.08303    -0.02 <2e-16 ***
## Prop. Mediated    0.85491     0.43297     1.24 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 15
##
##
```



```
## Simulations: 100
```

Mediation test 2 on Rflex \leftarrow t0 \leftarrow GFlex

Hong thinks this is negative result.

```
med.fit = lm(t0 ~ GompGFlex, data=report2)
summary(med.fit)
```

```
##
## Call:
## lm(formula = t0 ~ GompGFlex, data = report2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.8384 -2.9110 -1.2414 -0.1378  11.7384
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   132.397      5.381   24.60 2.74e-12 ***
## GompGFlex    -591.331     45.339  -13.04 7.65e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.579 on 13 degrees of freedom
## Multiple R-squared:  0.929, Adjusted R-squared:  0.9235
## F-statistic: 170.1 on 1 and 13 DF,  p-value: 7.647e-09
```

```
out.fit = lm(log10GompRFlex ~ t0 + GompGFlex, data=report2)
summary(out.fit)
```

```
##
## Call:
## lm(formula = log10GompRFlex ~ t0 + GompGFlex, data = report2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52445 -0.24582  0.05386  0.21803  0.34448
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.06152     2.29600   0.027   0.979
## t0             -0.01329     0.01716  -0.774   0.454
## GompGFlex     -16.57147    10.52694  -1.574   0.141
##
## Residual standard error: 0.2833 on 12 degrees of freedom
## Multiple R-squared:  0.4607, Adjusted R-squared:  0.3709
## F-statistic: 5.126 on 2 and 12 DF,  p-value: 0.02459
```

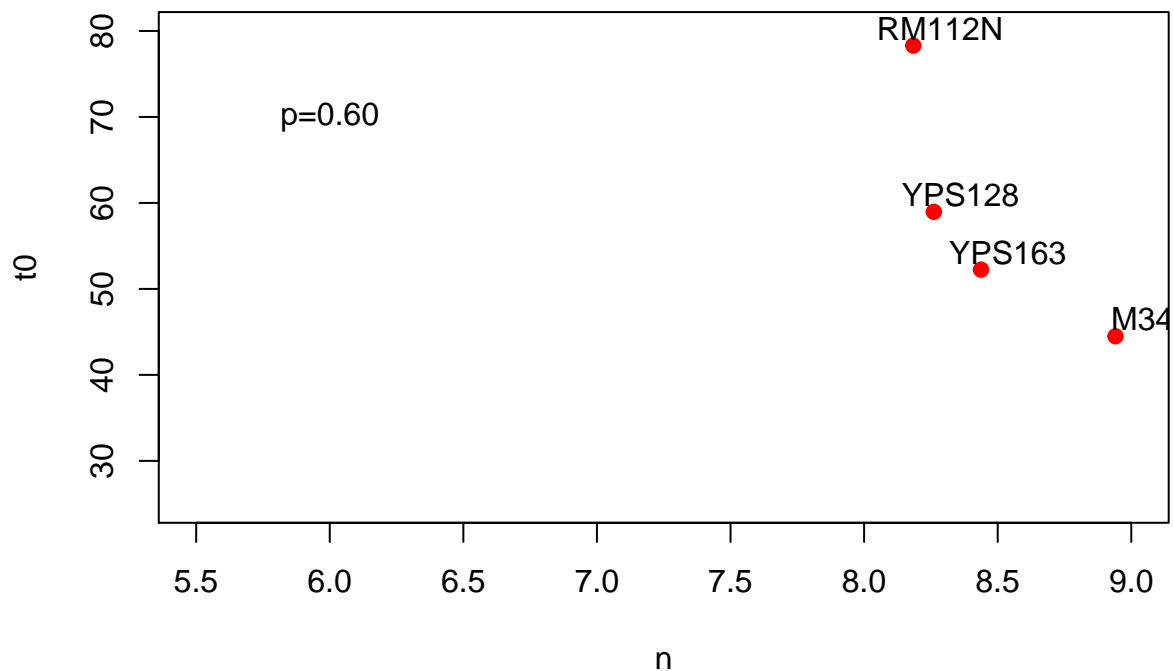
```
med.out <- mediate(med.fit, out.fit, treat = "GompGFlex", mediator = "t0", robustSE = TRUE, sims = 100)
summary(med.out)
```

```
##
## Causal Mediation Analysis
##
```

```
## Quasi-Bayesian Confidence Intervals
##
##           Estimate 95% CI Lower 95% CI Upper p-value
## ACME           6.639    -14.853     32.45    0.76
## ADE            -15.286    -45.947      8.47    0.18
## Total Effect    -8.648    -18.553     -0.33    0.04 *
## Prop. Mediated  -0.584    -15.157      3.15    0.76
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 15
##
##
## Simulations: 100
```

Plot initial virtula life ~ n (average interactions)

```
plot( report2$t0 ~ report2$n, col='red', xlim=c(5.5, 9), ylim=c(25, 80), pch=19, xlab='n', ylab='t0')
my.x = report2$n + 0.1; my.y = report2$t0 + 2;
names(my.x) = report2$my.strains; names(my.y) = report2$my.strains;
my.y[c("M14", "SGU57", "M32", "M13" )]=c(52, 52, 33, 38)
text(my.x,my.y, report2$my.strains )
text(6, 70, "p=0.60")
```



```
summary(lm(log10(report2$t0) ~ report2$n)) #G from t0 and n
```

```
##
## Call:
## lm(formula = log10(report2$t0) ~ report2$n)
##
## Residuals:
```

```
##      Min      1Q   Median      3Q      Max
## -0.14902 -0.08106 -0.03243  0.08902  0.23039
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.87284    0.33465   5.596 8.68e-05 ***
## report2$n    -0.00845    0.03538  -0.239   0.815
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1096 on 13 degrees of freedom
## Multiple R-squared:  0.004369, Adjusted R-squared:  -0.07222
## F-statistic: 0.05705 on 1 and 13 DF, p-value: 0.8149

Gompert versus Weibull? AIC: smaller is better (for information loss)
report$BestModel = ifelse(report$GompAICFlex < report$WeibAICFlex, "Gomp", "Weib")
report$BestModel = ifelse(abs(report$GompAICFlex - report$WeibAICFlex)<2, "<2", report$BestModel)
```

CV ~ Gomp and Weibull? How does noises influence likelihood of Gompertz and Weibull fitting?

```
summary(lm(report$CV ~ report$BestModel ))

##
## Call:
## lm(formula = report$CV ~ report$BestModel)
##
## Residuals:
##      Min      1Q   Median      3Q      Max
## -0.07528 -0.03570 -0.01063  0.03596  0.10837
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.31995    0.04309   7.426 7.99e-06 ***
## report$BestModelGomp  0.01334    0.04817   0.277   0.787
## report$BestModelWeib -0.04698    0.05098  -0.922   0.375
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06093 on 12 degrees of freedom
## Multiple R-squared:  0.2032, Adjusted R-squared:  0.07043
## F-statistic:  1.53 on 2 and 12 DF, p-value: 0.2559

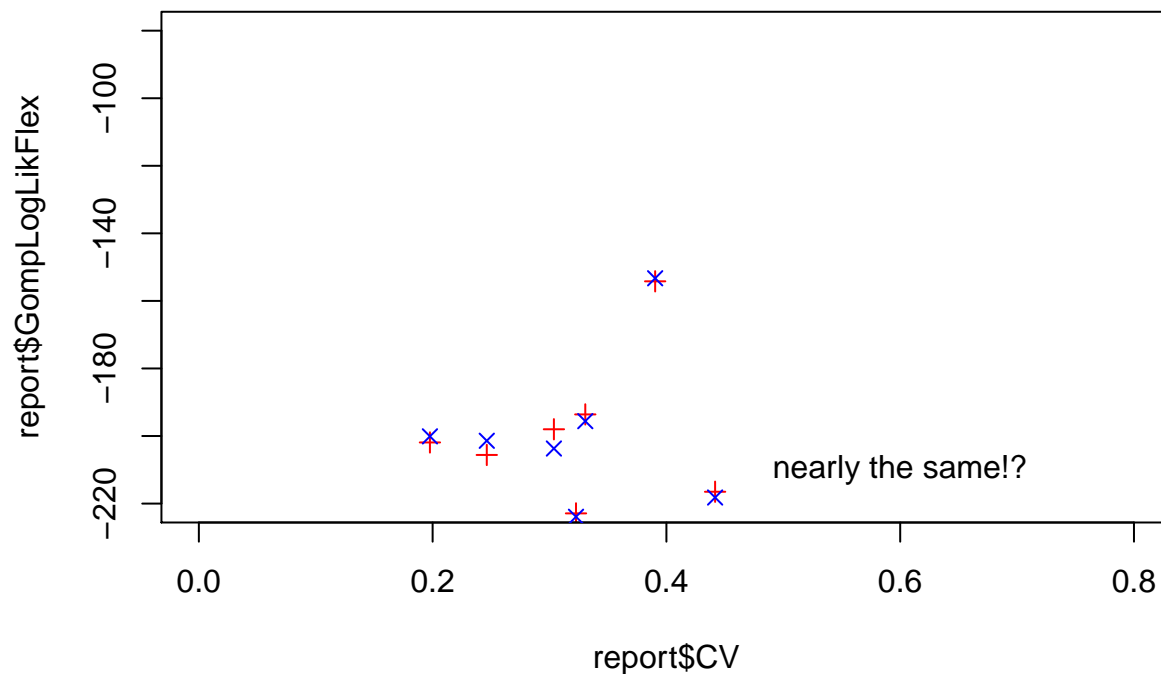
summary(lm(report$CV ~ report$WeibLogLikFlex ))

##
## Call:
## lm(formula = report$CV ~ report$WeibLogLikFlex)
##
## Residuals:
##      Min      1Q   Median      3Q      Max
## -0.11400 -0.04749  0.01106  0.03587  0.13004
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.124e-01  4.099e-02   7.623 3.78e-06 ***
## report$WeibLogLikFlex 3.708e-06  1.347e-04   0.028   0.978
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06558 on 13 degrees of freedom
## Multiple R-squared:  5.83e-05,    Adjusted R-squared:  -0.07686
## F-statistic: 0.000758 on 1 and 13 DF,  p-value: 0.9785
summary(lm(report$CV ~ report$GompLogLikFlex  ))

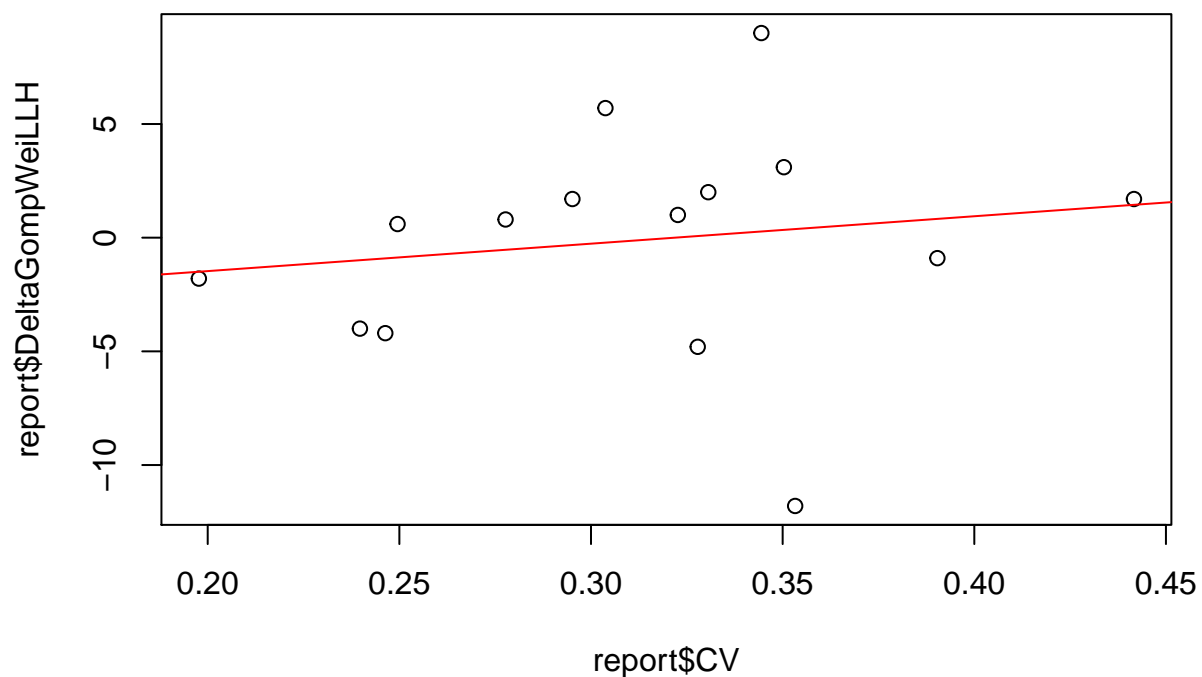
##
## Call:
## lm(formula = report$CV ~ report$GompLogLikFlex)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.11419 -0.04729  0.01091  0.03576  0.12988
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.131e-01  4.023e-02   7.785 3.01e-06 ***
## report$GompLogLikFlex 6.258e-06  1.316e-04   0.048   0.963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06558 on 13 degrees of freedom
## Multiple R-squared:  0.0001739,    Adjusted R-squared:  -0.07674
## F-statistic: 0.002261 on 1 and 13 DF,  p-value: 0.9628
summary(lm(report$CV ~ (report$GompLogLikFlex - report$WeibLogLikFlex)))

##
## Call:
## lm(formula = report$CV ~ (report$GompLogLikFlex - report$WeibLogLikFlex))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.11419 -0.04729  0.01091  0.03576  0.12988
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.131e-01  4.023e-02   7.785 3.01e-06 ***
## report$GompLogLikFlex 6.258e-06  1.316e-04   0.048   0.963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06558 on 13 degrees of freedom
## Multiple R-squared:  0.0001739,    Adjusted R-squared:  -0.07674
## F-statistic: 0.002261 on 1 and 13 DF,  p-value: 0.9628
plot( report$GompLogLikFlex ~ report$CV, col="red", pch=3, xlim=c(0, 0.8), ylim=c(-220, -80))
points( report$CV, report$WeibLogLikFlex, col="blue", pch=4)
m1 = lm( report$GompLogLikFlex ~ report$CV)
m2 = lm( report$WeibLogLikFlex ~ report$CV)
abline( m1, col="red", lty=2)
abline( m2, col='blue', lty=1)
text(0.6, -210, "nearly the same!?")
```



The QIN-RLS data suggested that noisy system signal prefer Gompertz model, based on GG01 theory. Notice that CV measures distrubition of system signals and are different from white noises (residues)

```
report$DeltaGompWeiLLH = report$GompLogLikFlex - report$WeibLogLikFlex
plot( report$DeltaGompWeiLLH ~ report$CV )
m3 = lm( report$DeltaGompWeiLLH ~ report$CV)
abline(m3, col='red')
```



TODO: Calculate the white noises (fitting errors) using the fitting residues for Gompertz and Weibull models.

```
# report$residues ??
```

Output

```
#write.csv(report2, file = 'sandbox/_report_qinlab_natural_isolates_rls.csv', row.names = FALSE)
```