Advancing in R
Nonlinear models

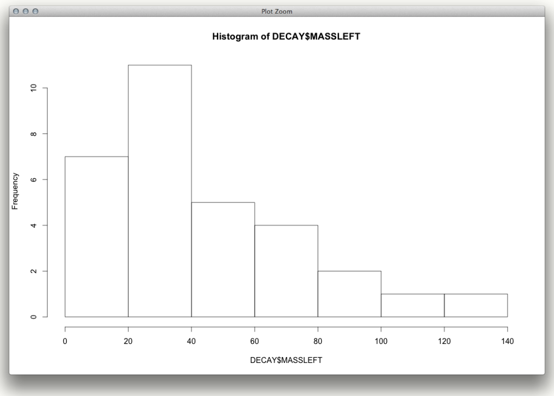
Transformations, polynomials and nonlinear least squares

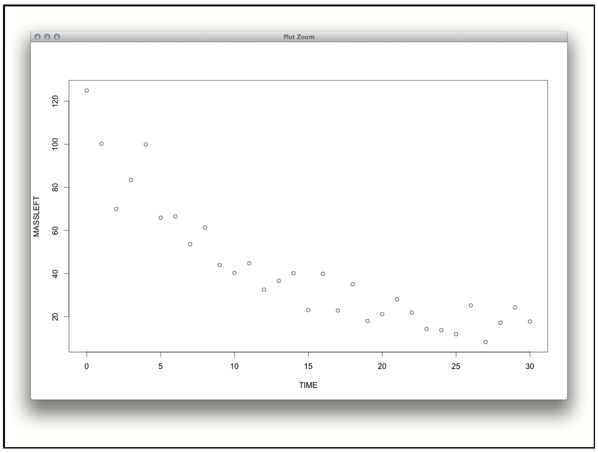
Outline: nonlinear regression

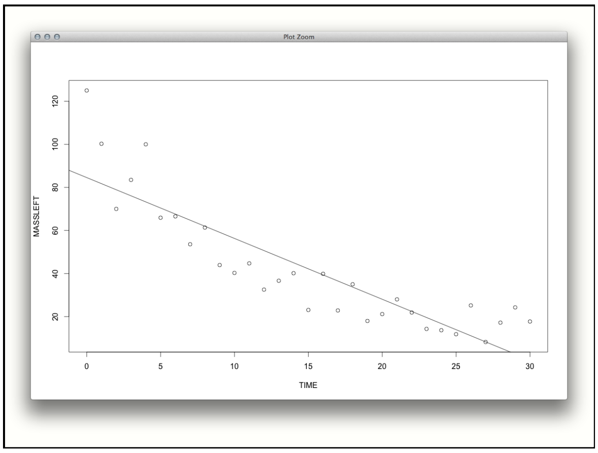
- Transformations
  - linearizing variables to straighten out relationships
- Polynomials
  - "linear" models with higher order terms that capture curvature in responses
- Nonlinear least squares
  - fitting custom-made curves

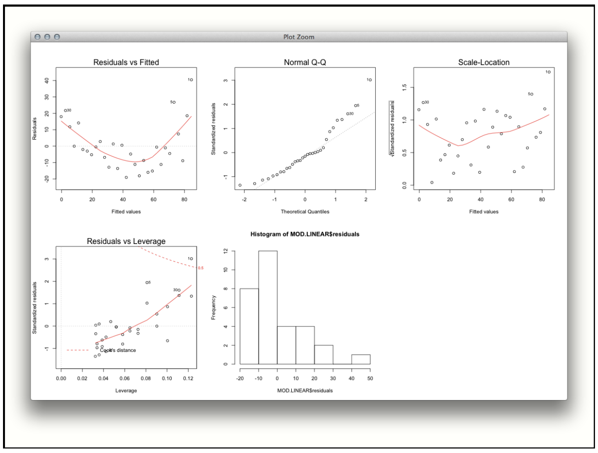
Some considerations

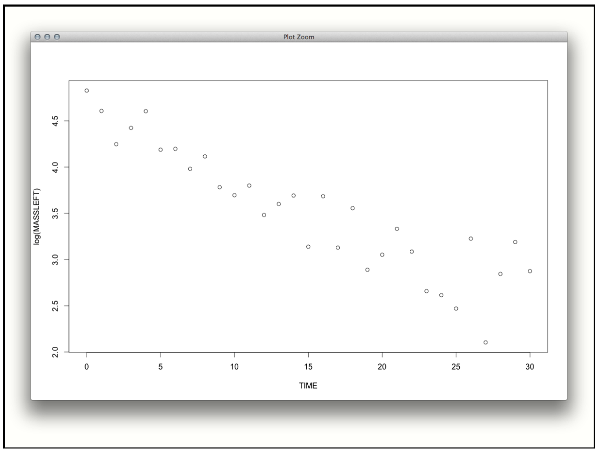
- Modelling assumptions
  - Homogeneity of variance?
- Theory of the relationship
  - E.g., for growth or decay or survival there there are existing functional models
- Parsimony
  - Straight lines preferred over curves
  - Fewer parameters the better
- Starting values?

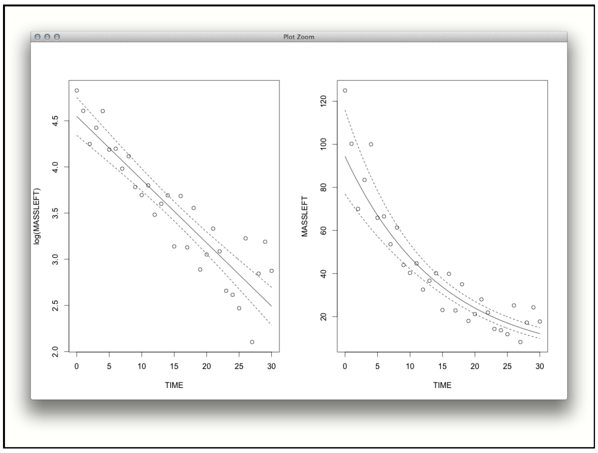


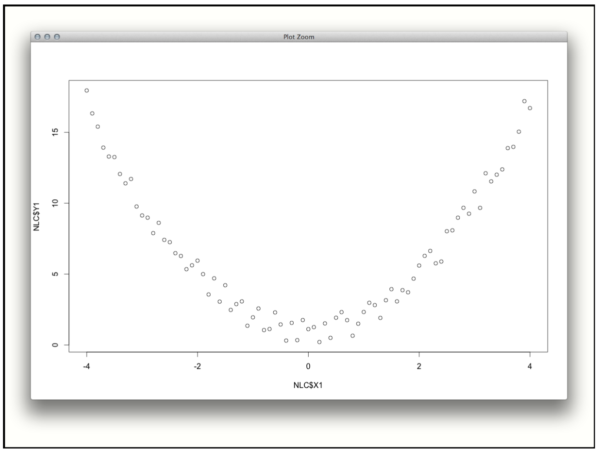

Plot Zoom
MASSLEFT
TIME

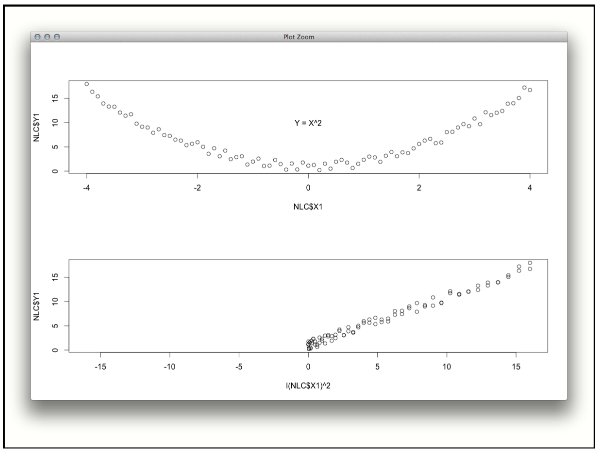

Plot Zoom
MASSLEFT
TIME

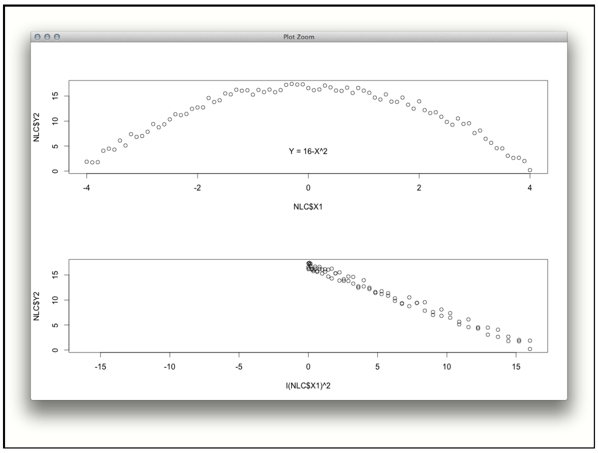

Plot Zoom
Residuals vs Fitted
Normal Q-Q
Scale-Location
Residuals vs Leverage
Histogram of MOD.LINEAR$residuals
Fitted values
Theoretical Quantiles
Leverage
MOD.LINEAR$residuals
Frequency

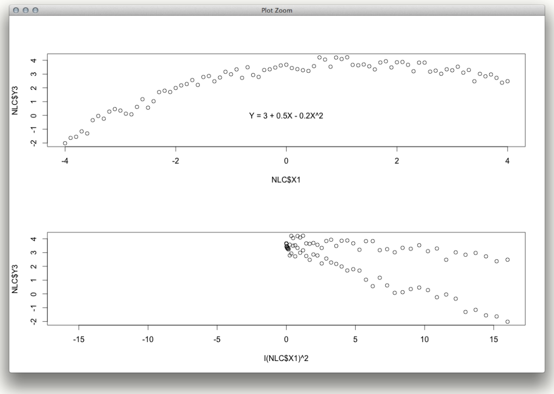

Plot Zoom
log(MASSLEFT)
TIME


Plot Zoom
log(MASSLEFT)
TIME
MASSLEFT
TIME


Plot Zoom
NLC$Y1
NLC$X1


Plot Zoom
NLC$Y1
Y = X^2
NLC$X1
NLC$Y1
I(NLC$X1)^2


Plot Zoom
NLC$Y2
Y = 16-X^2
NLC$X1
NLC$Y2
I(NLC$X1)^2




```
> MOD.POLYS<-lm(Y3~poly(X1,2),data=NLC)
> summary(MOD.POLYS)

Call:
lm(formula = Y3 ~ poly(X1, 2), data = NLC)

Residuals:
     Min       1Q   Median       3Q      Max
-0.52325 -0.23905  0.02902  0.25626  0.41930

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.46832    0.03237   76.25   <2e-16 ***
poly(X1, 2)1 10.43650    0.29134   35.82   <2e-16 ***
poly(X1, 2)2 -8.87316    0.29134  -30.46   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2913 on 78 degrees of freedom
Multiple R-squared: 0.9659,   Adjusted R-squared: 0.965
F-statistic:  1105 on 2 and 78 DF,  p-value: < 2.2e-16
```

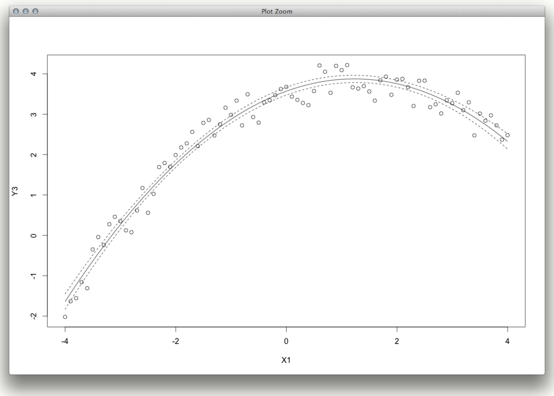


Table 9.1 Illustrative set of useful non-linear functions with corresponding R model fitting syntax. Some examples also illustrate corresponding self-starting functions. Note that this is a non-exhaustive set.

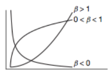
Function	Preview

Concave/convex functions

Power ($y = \alpha x^{\beta}$)

Used to describe a large range of physical and biological trends including allometric scaling relationships (e.g. Kleiber's law) and inverse square laws (e.g. Newtonian gravity). α defines the scale of the y-axis and β defines the magnitude and polatity of the rate of change and thus the degree of curvature

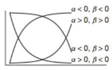
```
> nls(DV~a*IV^b, dataset, start=list(a=1,
    b=0.1))
```



Exponential ($y = \alpha e^{\beta x}$)

Models non-asymptotic growth and decay. α defines the scale of the y-axis and increasing magnitude of β increases the curvature of the curve.

```
> nls(DV~a*exp(b*IV), dataset, start=list(a=1,
    b=0.1))
```

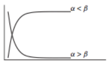


Aymptotic functions

Asymptotic exponential ($y = \alpha + (\beta - \alpha)e^{-e^{\gamma}x}$)

Used to describe general asymptotic relationships.
Equivelent to the more simple $y = a - be^{-cx}$ when $a = \alpha$, $b = \beta - \alpha$ and $c = e^{\gamma}$

α - y value of horizontal asymptote. β - value of y when $x = 0$.
γ - natural log of rate of curvature

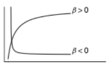



```
> nls(DV~a+b*exp(c*x),dataset,start=list(a=1,
    b=-1,c=-1))
> nls(DV~SSasymp(IV,a,b,c), dataset)
```

Michaelis-Menten ($y = \frac{\alpha x}{\beta + x}$)

Used to relate rates of enzymatic reactions to substrate concentrations

$\alpha - y$ value of horizontal asymptote. β (*Mechaelis parameter*) - value of x at which half the asymptotic response is obtained.

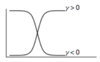



```
> nls(DV~(a*IV)/(b*IV), dataset,
    start=list(a=1, b=1))
> nls(DV~SSmicmen(IV,a,b), dataset)
```

Sigmoidal

Logistic ($y = \frac{\alpha}{1+e^{(\beta - x)/\gamma}}$)

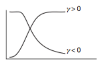
Used to describe binary responses (presence/absence, alive/dead, etc) relationships.
α - horizontal asymptote (typically 1). β - value of x at which half the asymptotic response is obtained (inflection point).
γ - determines the steepness at inflection.




```
> nls(DV~a/(1+exp((b-IV)/c)), dataset,
    start=list(a=1,b=1,c=.1))
> nls(DV~SSlogis(IV,a,b,c), dataset)
```

Weibull ($y = \alpha - \beta e^{-(e^{\gamma}x^{\delta})}$)

Describes the kinetics of many enzymes. Used to relate rates of enzymatic reactions to substrate concentrations
α - right side horizontal asymptote. β - rate of vertical change.
γ - natural log of rate of curvature. δ - power to raise x.

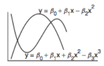



```
> nls(DV~a - b*exp(-exp(c)*IV^d), dataset,
    start=list(a=1, b=1, c=1, d=1))
> nls(DV~SSweibull(IV,a,b,c,d), dataset)
```

Peaks and/or valleys

Polynomials

Describes the kinetics of many enzymes. Used to relate rates of enzymatic reactions to substrate concentrations




```
> lm(DV~ IV + I(IV^2) + I(IV^3), dataset)
> lm(DV poly(IV, 3), dataset)
```

Nonlinear least squares

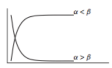
- Fit pre-defined functions
- Parameters estimated by iteratively changing them to minimize SS or maximize likelihood
- Often need realistic starting values
  - Educated guesses
  - Try several alternatives to check sensitivity of results
- Hypothesis testing not straightforward

Aymptotic functions

Asymptotic exponential ($y = \alpha + (\beta - \alpha)e^{-e^{\gamma}x}$)

Used to describe general asymptotic relationships.
Equivelent to the more simple $y = a - be^{-cx}$ when $a = \alpha$, $b = \beta - \alpha$ and $c = e^{\gamma}$

α - y value of horizontal asymptote. β - value of y when $x = 0$.
γ - natural log of rate of curvature

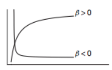


```
> nls(DV~a+b*exp(c*x),dataset,start=list(a=1,
    b=-1,c=-1))
> nls(DV~SSasymp(IV,a,b,c), dataset)
```

Michaelis-Menten ($y = \frac{\alpha x}{\beta + x}$)

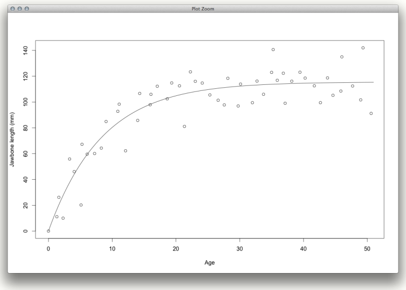
Used to relate rates of enzymatic reactions to substrate concentrations

$\alpha - y$ value of horizontal asymptote. β (*Mechaelis parameter*) - value of x at which half the asymptotic response is obtained.



```
> nls(DV~(a*IV)/(b*IV), dataset,
    start=list(a=1, b=1))
> nls(DV~SSmicmen(IV,a,b), dataset)
```

```
> MOD.ASYMP<-nls(BONE~a-b*exp(-c*AGE),
data=DEER, start=list(a=120,b=110,c=0.064))
```



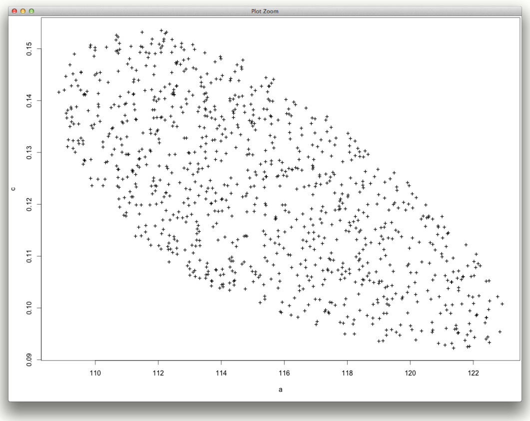
```
> summary(MOD.ASYMP)

Formula: BONE ~ a - b * exp(-c * AGE)

Parameters:
  Estimate Std. Error t value Pr(>|t|)
a 115.2528     2.9139   39.55  < 2e-16 ***
b 118.6875     7.8925   15.04  < 2e-16 ***
c   0.1235     0.0171    7.22 2.44e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
'.' 0.1 ' ' 1

Residual standard error: 13.21 on 51 degrees of
freedom

Number of iterations to convergence: 5
Achieved convergence tolerance: 2.405e-06
```



Suggested reading:

- Ch. 9 in Logan
- Chs. 10, 20 in Crawley *The R Book*