

Data Analysis in R

Regression - Sigmoid Curves

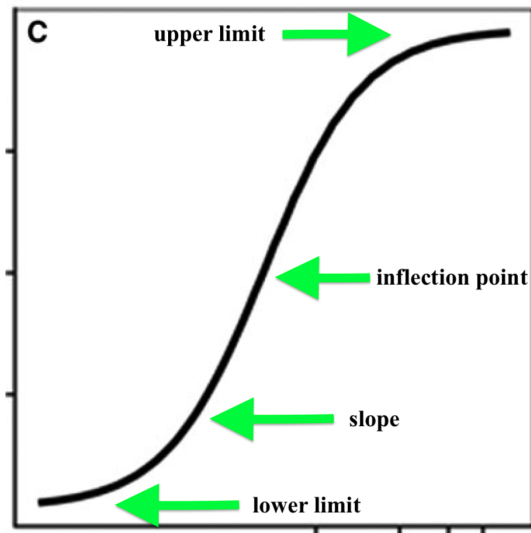
Maxwel Coura Oliveira, PhD

University of Wisconsin-Madison

Sigmoid curves

- It is a nonlinear regression
- It is a mathematical function having a characteristic “S”-shaped curve or sigmoid curve.
- It is a logistic function (e = the natural logarithm base (also known as Euler's number))

Curve and Parameters



Example

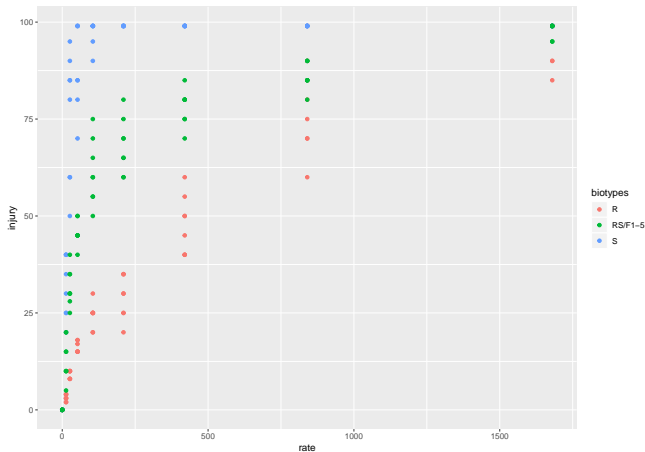
- A study performed to compare waterhemp R (resistant) and S (susceptible) populations to HPPD-inhibitor herbicides

```
data <- read_csv("Dose.csv")
```

```
## Parsed with column specification:
## cols(
##   biotypes = col_character(),
##   run = col_character(),
##   rate = col_double(),
##   injury = col_double()
## )
```

Plot the data

```
ggplot(data, aes(x=rate, y=injury, color=biotypes)) +  
  geom_point()
```



Model

- Example of **four-parameter log-logistic**:

Equation:

$$Y = c + \frac{d - c}{(1 + \exp[b(\log x - \log e)])}$$

where Y is response variable (%), x is the x -values (treatment). The parameter c is the lower limit, d is the upper limit (asymptote), b is the slope and the parameter e is the ED50 (effective x that causes 50% reduction in Y).

- There are multiple variations in sigmoid models. See **Knezevic et al. (2007)**

Function

- Package **drc** makes easier to work with sigmoid curves
- Use package **drc** and **tidyverse**
- Function **drm**

```
model <- drm(injury ~ rate, biotypes, fct = 14(), data=data)
summary(model)
```

```
##
## Model fitted: Log-logistic (ED50 as parameter) (4 parms)
##
## Parameter estimates:
##
##           Estimate Std. Error  t-value   p-value
## b:R          -0.76587    0.15099  -5.0722 8.814e-07 ***
## b:S          -2.57224    0.24120 -10.6643 < 2.2e-16 ***
## b:RS/F1-5     -0.82957    0.08032 -10.3282 < 2.2e-16 ***
## c:R           0.57180    2.38063   0.2402 0.810427
## c:S          -0.26795    2.27906  -0.1176 0.906522
## c:RS/F1-5     -1.10894    2.18517  -0.5075 0.612362
## d:R          189.78824   66.15590   2.8688 0.004554 **
## d:S           98.43896    1.05137  93.6296 < 2.2e-16 ***
## d:RS/F1-5    100.48984    4.13543  24.2997 < 2.2e-16 ***
## e:R          1587.54650  1378.60385   1.1516 0.250849
## e:S           17.16366    0.66135  25.9526 < 2.2e-16 ***
## e:RS/F1-5     71.93429   10.49998   6.8509 8.491e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model selection

- Model selection with `anova` function

```
model2 <- drm(injury ~ rate, biotypes, fct = l3(), data=data)
```

```
anova(model, model2)
```

```
##
## 1st model
## fct:      13()
## 2nd model
## fct:      14()

## ANOVA table
##
##           ModelDf      RSS Df F value p value
## 2nd model      207 8601.6
## 1st model      204 8581.5  3  0.1594  0.9235
```


Model selection

- Model selection with `mselect` function

```
mselect(model, list(l3(), W2.3(), W1.3(), W1.4(), W2.4()),
        nested = TRUE,
        sorted = c("IC", "Res var", "Lack of fit", "no"),
        linreg = TRUE, icfct = AIC)
```

##	logLik	IC	Lack of fit	Res var
## W1.3	-700.7944	1421.589	4.775318e-03	40.18602
## l3	-704.4085	1428.817	5.981242e-04	41.55356
## W1.4	-701.9982	1429.996	5.962530e-04	41.23405
## l4	-704.1556	1434.311	1.514111e-04	42.06605
## W2.3	-714.2772	1448.554	1.023427e-06	45.52951
## W2.4	-713.6539	1453.308	2.263080e-07	45.93317
## Cubic	-1000.1991	2010.398	NA	627.60903
## Quad	-1013.6819	2035.364	NA	707.72264
## Lin	-1036.5831	2079.166	NA	870.80401

Model is selected

- The selected model is the log logistic **W1.3** (3-parameters)

```
model2 <- drm(injury ~ rate, biotypes, fct = W1.3(fixed=c(NA,NA,NA),  
names=c("slope", "upper", "infpoint")), data=data)
```

```
summary(model2)
```

```
##  
## Model fitted: Weibull (type 1) with lower limit at 0 (3 parms)  
##  
## Parameter estimates:  
##  
##           Estimate Std. Error t-value p-value  
## slope:R      -2.3535e-01 2.6164e-02 -8.9950 < 2.2e-16 ***  
## slope:S      -1.9653e+00 1.7035e-01 -11.5367 < 2.2e-16 ***  
## slope:RS/F1-5 -5.2790e-01 5.9584e-02 -8.8597 4.066e-16 ***  
## upper:R       4.7775e+02 1.1723e+02  4.0755 6.543e-05 ***  
## upper:S       9.8907e+01 1.0723e+00 92.2384 < 2.2e-16 ***  
## upper:RS/F1-5 1.0905e+02 6.0820e+00 17.9300 < 2.2e-16 ***  
## infpoint:R    1.2427e+04 1.0570e+04  1.1756  0.2411  
## infpoint:S    1.3903e+01 4.0950e-01 33.9504 < 2.2e-16 ***  
## infpoint:RS/F1-5 4.3291e+01 5.7682e+00  7.5051 1.790e-12 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error:  
##  
## 6.339245 (207 degrees of freedom)
```

Fixing the modeling problem

- There is a problem at my model. Let's fixed!

```
model3 <- drm(injury ~ rate, biotypes, fct = W1.3(fixed=c(NA, 100, NA),
names=c("slope", "upper", "infpnt")), data=data)
```

```
summary(model3)
```

```
##
## Model fitted: Weibull (type 1) with lower limit at 0 (2 parms)
##
## Parameter estimates:
##
##           Estimate Std. Error t-value  p-value
## slope:R      -0.758462   0.052162 -14.540 < 2.2e-16 ***
## slope:S      -1.880586   0.169438 -11.099 < 2.2e-16 ***
## slope:RS/F1-5 -0.628063   0.031594 -19.879 < 2.2e-16 ***
## infpoint:R    191.945010  12.690310  15.125 < 2.2e-16 ***
## infpoint:S     13.966846   0.501431  27.854 < 2.2e-16 ***
## infpoint:RS/F1-5 36.316981  2.292598  15.841 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error:
##
## 7.611254 (210 degrees of freedom)
```

Post-hoc analysis

- Usually researchers are looking for the the dose that causes 50% or 90% injury (e.g., control or biomass reduction) in the weed biotypes.
- The values is estimated with **ED** function

```
ED(model3, c(50,90), type="absolute", interval = "delta")
```

```
##  
## Estimated effective doses  
##  
##           Estimate Std. Error      Lower      Upper  
## e:R:50          311.20147    17.10679    277.47843    344.92452  
## e:R:90          3730.21779   656.15897   2436.71535   5023.72024  
## e:RS/F1-5:50      65.09487     3.80044     57.60296     72.58678  
## e:RS/F1-5:90    1306.76637   217.25708    878.48211   1735.05063  
## e:S:50           16.97224     0.55646     15.87528     18.06921  
## e:S:90           46.21559     4.55312     37.23992     55.19126
```

Test statistics

- The **EDcomp** function provides a pairwise t-test with the selected **ED** values

```
EDcomp(model3, c(50,50), type="absolute")
```

```
##  
## Estimated ratios of effect doses  
##  
##           Estimate Std. Error   t-value   p-value  
## R/RS/F1-5:50/50 4.7807e+00 3.8336e-01 9.8620e+00 2.1367e-19  
## R/S:50/50        1.8336e+01 1.1736e+00 1.4772e+01 1.2409e-34  
## RS/F1-5/S:50/50 3.8354e+00 2.5681e-01 1.1041e+01 6.0449e-23
```

Comparing parameters

```
compParm(model3, "slope", "-")
```

```
##
## Comparison of parameter 'slope'
##
##           Estimate Std. Error t-value    p-value
## R-S          1.122124   0.177285   6.3295 1.465e-09 ***
## R-RS/F1-5    -0.130399   0.060984  -2.1382   0.03365 *
## S-RS/F1-5    -1.252523   0.172358  -7.2670 7.109e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
compParm(model3, "infpnt", "-")
```

```
##
## Comparison of parameter 'infpnt'
##
##           Estimate Std. Error t-value    p-value
## R-S          177.9782   12.7002 14.0138 < 2.2e-16 ***
## R-RS/F1-5    155.6280   12.8957 12.0682 < 2.2e-16 ***
## S-RS/F1-5    -22.3501    2.3468  -9.5237 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Confidence Intervals for model parameters

```
confint(model3)
```

##	2.5 %	97.5 %
## slope:R	-0.8612912	-0.6556333
## slope:S	-2.2146034	-1.5465691
## slope:RS/F1-5	-0.6903454	-0.5657815
## infpoint:R	166.9282868	216.9617332
## infpoint:S	12.9783618	14.9553304
## infpoint:RS/F1-5	31.7975258	40.8364371

Figure

```
plot(model3, col=c(1,2,3))
```

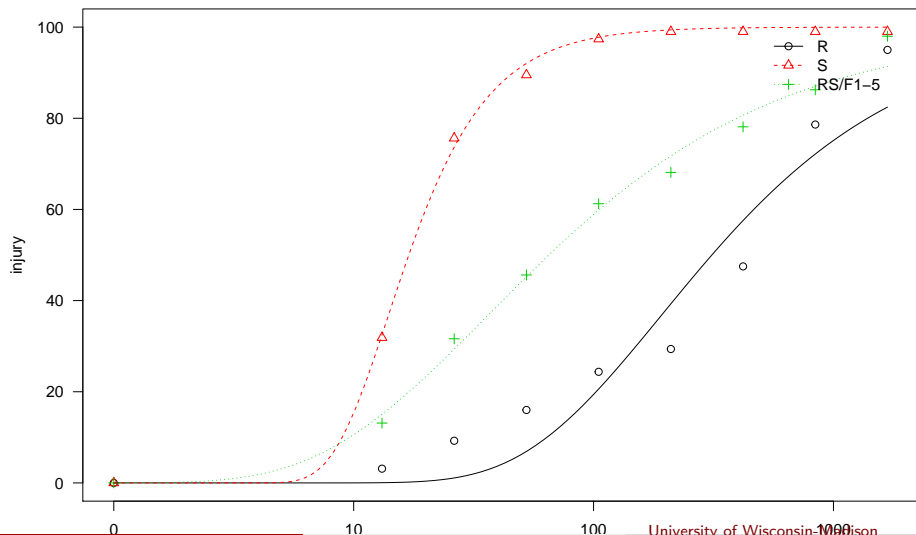


Figure using ggplot2

```
# new dose levels as support for the line
newdata <- expand.grid(rate=exp(seq(log(0.5), log(1680), length=1680)))
newdata1 <- data.frame(biotypes =c("R"), newdata)
newdata2 <- data.frame(biotypes =c("S"), newdata)
newdata3 <- data.frame(biotypes =c("RS/F1-5"), newdata)

nd <- rbind(newdata1, newdata2, newdata3)

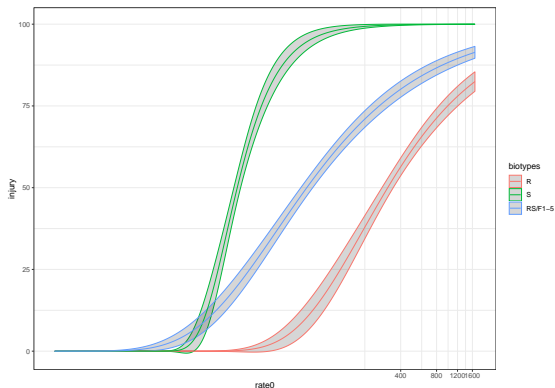
# new data with predictions
pm <- predict(model3, newdata=nd, interval="confidence")

# new data with predictions
nd$p <- pm[,1]
nd$pmin <- pm[,2]
nd$pmax <- pm[,3]

# need to shift conc == 0 a bit up, otherwise there are problems with coord_trans
data$rate0 <- data$rate
data$rate0[data$rate0 == 0] <- 0.5
```

Figure ggplot2

```
ggplot(data, aes(x=rate0, y=injury, color=biotypes)) + #geom_point() +  
  geom_ribbon(data=nd, aes(x=rate, y=p, ymin=pmin, ymax=pmax), alpha=0.2) +  
  geom_line(data=nd, aes(x=rate, y=p)) + theme_bw() +  
  coord_trans(x="log")
```



Other uses

- Critical time for weed removal
- Weed emergence