Waterhemp PRE BS Fomesafen DR Counts

Felipe Faleco

10/19/2022

READ DATA

```
data <- read.csv("data_PRE_BS_fome_2.csv")

str(data)
```

```
## 'data.frame':    128 obs. of  7 variables:
##  $ run    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ pop    : chr  "BS" "BS" "BS" "BS" ...
##  $ herb   : chr  "fome" "fome" "fome" "fome" ...
##  $ rate   : int  0 0 0 0 33 33 33 33 66 66 ...
##  $ rep    : int  1 2 3 4 1 2 3 4 1 2 ...
##  $ count  : int  10 16 0 0 22 0 13 0 5 0 ...
##  $ biomass: num  0.13 0.524 0 0 0.829 0 0.324 0 0.053 0 ...
```

```
fun <- function(x){
  c( mean = mean(x), VAR = var(x), SD = sd(x), SE = std.error(x), n = length(x))
}

summary.BS <- summaryBy(cbind(count, biomass) ~ rate + pop, data = subset(data, pop=="BS"),
          FUN=fun)


summary.A66 <- summaryBy(cbind(count, biomass) ~ rate + pop, data = subset(data, pop=="A66"),
          FUN=fun)
```

MODEL LL.4

```
model.W1.3<- drm(count~ rate, pop, fct = W1.3(), data = data, type = "continuous")


modelFit(model.W1.3) # higher the p-value, better.
```

```
## Lack-of-fit test
##
##             ModelDf    RSS Df F value p value
## ANOVA         112 3564.6
## DRC model     122 3700.4 10  0.4265  0.9310
```

```
summary(model.W1.3)
```

```
##
## Model fitted: Weibull (type 1) with lower limit at 0 (3 parms)
##
## Parameter estimates:
##
##         Estimate Std. Error t-value   p-value
## b:BS     1.80403    0.73192  2.4648 0.0151007 *
## b:A66    1.35074    3.51952  0.3838 0.7018061
## d:BS    11.74563    1.42987  8.2145 2.587e-13 ***
## d:A66   14.50917    1.94630  7.4547 1.441e-11 ***
## e:BS   182.36063   47.24309  3.8600 0.0001827 ***
## e:A66   21.49313   26.90537  0.7988 0.4259348
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error:
##
##  5.507348 (122 degrees of freedom)
```

```
# c: lower limit (models with negative lower limit are biologically impossible. E.g., negative biomass,

# d: upper limit (models where the response variable is in %, upper limit bigger than 100% are biologic

# b: slope around e (inflection point)

# e: e (inflection point). Represents the dose producing a response halfway between the u and l paramet
```

MODEL SELECTION

```
mselect(model.W1.3,  list(L.3(), L.4(), L.5(), LL.2(), LL.3(), LL.4(), LL.5(), W1.3(), W1.4(), W2.4(),
```

```
## Error in optim(startVec, opfct, hessian = TRUE, method = optMethod, control = list(maxit = maxIt,  :
##   non-finite finite-difference value [1]
## Error in optim(startVec, opfct, hessian = TRUE, method = optMethod, control = list(maxit = maxIt,  :
##   non-finite finite-difference value [5]
## Error in optim(startVec, opfct, hessian = TRUE, method = optMethod, control = list(maxit = maxIt,  :
##   non-finite finite-difference value [7]
## Error in optim(startVec, opfct, hessian = TRUE, method = optMethod, control = list(maxit = maxIt,  :
##   non-finite finite-difference value [8]
## Error in optim(startVec, opfct, hessian = TRUE, method = optMethod, control = list(maxit = maxIt,  :
##   non-finite finite-difference value [2]
```

```
##          logLik       IC Lack of fit  Res var
## W1.3  -396.9302 807.8604   0.9310491 30.33088
## W1.3  -396.9302 807.8604   0.9310491 30.33088
## L.3   -397.2532 808.5065   0.8965394 30.48436
## W2.4  -396.4818 810.9636   0.8998626 30.62110
## L.4   -397.3199 812.6399   0.7579154 31.02475
## LL.5  -396.6932 815.3865   0.6986021 31.24316
## L.5   -398.1899 818.3797   0.3692459 31.98238
## Cubic -416.7116 843.4232          NA 40.64967
## Quad  -422.3492 852.6983          NA 44.03766
## Lin   -429.5032 865.0063          NA 48.85506
## LL.2         NA       NA          NA       NA
## LL.3         NA       NA          NA       NA
## LL.4         NA       NA          NA       NA
## W1.4         NA       NA          NA       NA
## baro5        NA       NA          NA       NA
```

```
#LL.3 better option here (lower IC, which actually is AIC (default)). W1.3 could also be used, but sinc



#model.LL3<- drm(count ~ rate, pop, fct = LL.3(),  data = data, type = "continuous")



#modelFit(model.LL3)

#summary(model.LL3) # NO parameter c (lower limit). It means lower limit was set to zero.


#anova(model.LL3, model.LL4) # p-value > 0.05 means that there is no difference between models. Therefo
```

ED50 AND ED 90

```
ED(model.W1.3, c(50, 90), type = "relative")
```

```
##
## Estimated effective doses
##
##          Estimate Std. Error
## e:A66:50   16.385     31.834
## e:A66:90   39.853     20.686
## e:BS:50   148.833     38.329
## e:BS:90   289.543    100.117
```

```
# Estimated doses required for ED50, and ED90. Estimation of ED90 is statistically less accurate and of
```

ED 50 RI AND PARAMETERS STUDENT T-TEST COMPARISONS

```
EDcomp(model.W1.3, c(50, 50), type = "relative", interval = "none", reverse = TRUE)
```

```
##
## Estimated ratios of effect doses
##
##               Estimate Std. Error  t-value  p-value
## BS/A66:50/50  9.08326   17.80173  0.45407  0.65058
```

```
EDcomp(model.W1.3, c(90, 90), type = "relative", interval = "none", reverse = TRUE)
```

```
##
## Estimated ratios of effect doses
##
##               Estimate Std. Error t-value p-value
## BS/A66:90/90  7.26526    4.53129 1.38267 0.16929
```

```
# Comparing b parameter (slope)
compParm(model.W1.3, strVal = "b", operator = "-")
```

```
##
## Comparison of parameter ’b’
##
##         Estimate Std. Error t-value p-value
## BS-A66  0.45329    3.59482  0.1261  0.8999
```

```
# Comparing d parameter (upper limit)
compParm(model.W1.3, strVal = "d", operator = "-")
```

```
##
## Comparison of parameter ’d’
##
##         Estimate Std. Error t-value p-value
## BS-A66  -2.7635     2.4151 -1.1443  0.2547
```

```
# Comparing e parameter (ED50)
compParm(model.W1.3, strVal = "e", operator = "-") #here comparing the difference, and not the ratio li
```

```
##
## Comparison of parameter ’e’
##
##         Estimate Std. Error t-value  p-value
## BS-A66  160.868     54.367  2.9589 0.003709 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

#SAVE TIFF