Supplementary material 1: The R code for fitting multiple survival curves (data for multiple seed lots) and hypothesis testing.

Dustin Wolkis, Angelino Carta, Shabnam Rezaei, Fiona R. Hay

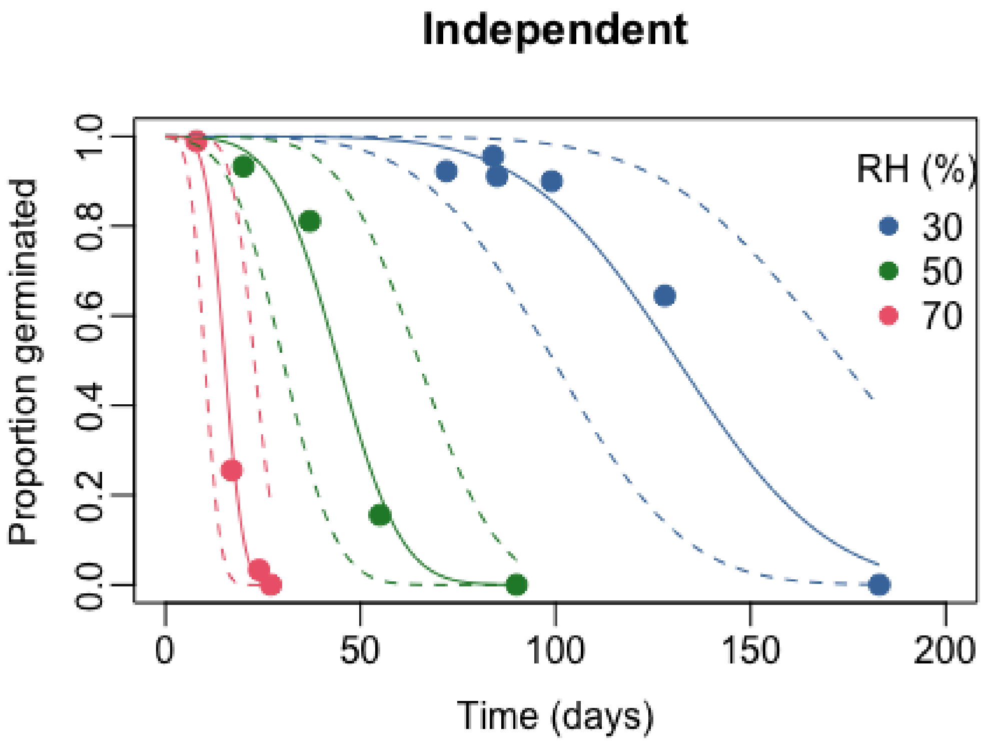
This supplementary material demontrates how to analyze seed longevity in R to fit the viability equation. Original data is from Rezaei *et al*. (2023).

```
# load libraries
library(tidyverse) # usefull for all manor of data manipulation and
visualizartion

## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.0     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors

library(MASS) # the dose.p() function used to calculate P50 and P85

##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select

library(knitr) # used to make tables in R Markdown

# set working directory
#setwd("my/file/path")



# Read in data
# brna short for the study organism, Brassica napus
```

```
brna <- read.csv("Rezaei_etal_2023_SSR.csv", header = TRUE)
glimpse(brna)

## Rows: 117
## Columns: 10
## $ Treatment    <chr> "Desorption 170", "Desorption 170", "Desorption 170",
"De…
## $ MC           <dbl> 9.141008, 9.141008, 9.141008, 9.141008, 8.000368,
8.00036…
## $ storage.RH   <dbl> 66.91, 66.91, 66.91, 66.91, 60.95, 60.95, 60.95,
60.95, 5…
## $ period       <int> 8, 17, 24, 27, 13, 24, 31, 41, 20, 37, 55, 90, 50,
59, 65…
## $ sown         <int> 90, 90, 90, 90, 87, 90, 90, 90, 90, 90, 90, 90, 90,
90, 9…
## $ germinated   <int> 89, 23, 3, 0, 83, 28, 10, 0, 84, 73, 14, 0, 73, 57,
51, 3…
## $ germ.percent <dbl> 98.888889, 25.555556, 3.333333, 0.000000, 95.402299,
31.1…
## $ sorption     <chr> "Desorption", "Desorption", "Desorption",
"Desorption", "…
## $ chamber.RH   <int> 70, 70, 70, 70, 60, 60, 60, 60, 50, 50, 50, 50, 40,
40, 4…
## $ cycle        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, …

# citation for dataset;
# Rezaei, S., Buitink, J. & Hay, F.R. (2023) Contrasting seed moisture
sorption
# behaviour between two species and the implication for seed longevity.
# Seed Science Research, 1-9. https://doi.org/10.1017/ S0960258523000156

# Tell R which variables are factors
brna <- brna |>
  mutate(Treatment = as.factor(Treatment),
         sorption = as.factor(sorption),
         chamber.RH = as.factor(chamber.RH),
         cycle = as.factor(cycle))
glimpse(brna)

## Rows: 117
## Columns: 10
## $ Treatment    <fct> Desorption 170, Desorption 170, Desorption 170,
Desorptio…
## $ MC           <dbl> 9.141008, 9.141008, 9.141008, 9.141008, 8.000368,
8.00036…
## $ storage.RH   <dbl> 66.91, 66.91, 66.91, 66.91, 60.95, 60.95, 60.95,
60.95, 5…
## $ period       <int> 8, 17, 24, 27, 13, 24, 31, 41, 20, 37, 55, 90, 50,
59, 65…
```

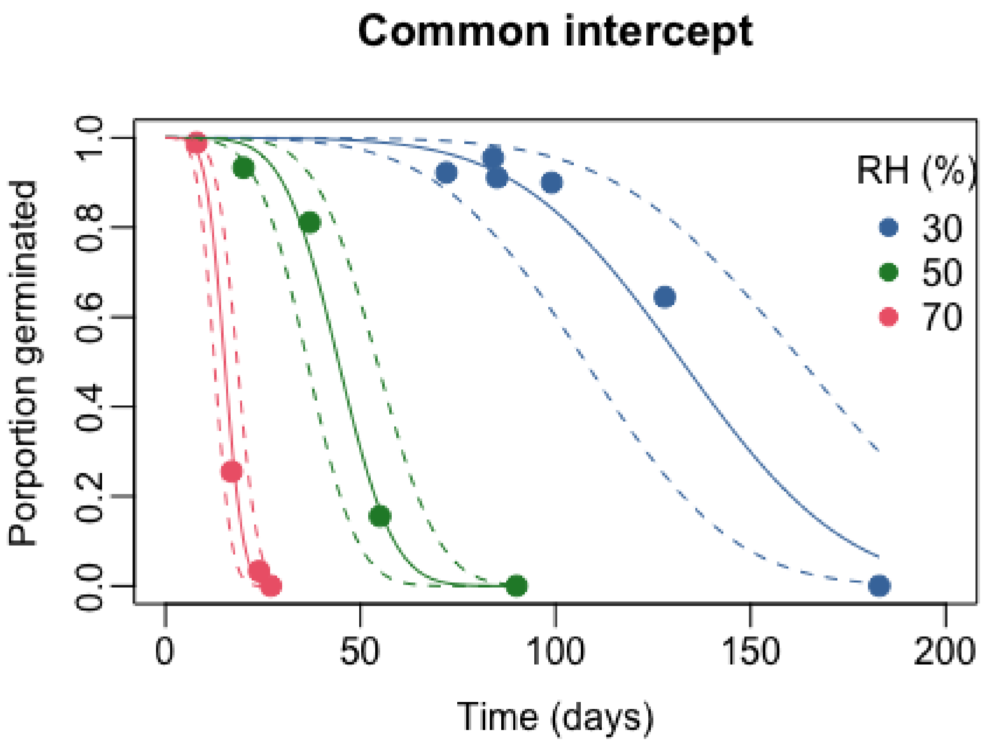
```
## $ sown         <int> 90, 90, 90, 90, 87, 90, 90, 90, 90, 90, 90, 90, 90,
90, 9…
## $ germinated   <int> 89, 23, 3, 0, 83, 28, 10, 0, 84, 73, 14, 0, 73, 57,
51, 3…
## $ germ.percent <dbl> 98.888889, 25.555556, 3.333333, 0.000000, 95.402299,
31.1…
## $ sorption     <fct> Desorption, Desorption, Desorption, Desorption,
Desorptio…
## $ chamber.RH   <fct> 70, 70, 70, 70, 60, 60, 60, 60, 50, 50, 50, 50, 40,
40, 4…
## $ cycle        <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, …

### analysis and plotting with three different Treatments ###

# subset desorption, cycle 1, and 3 RHs (30, 50, 70%)
des<-droplevels(subset(brna, sorption == "Desorption" &  cycle == "1"))
des<-droplevels(subset(des, chamber.RH == "30" |chamber.RH == "50" |
chamber.RH == "70"))

### Independent (independent Ki and -sigma^-1)
independent <- glm(formula = cbind(germinated, sown-germinated) ~ Treatment +
Treatment:period -1 , family = binomial(link = "probit"),  data = des)
summary(independent)

##
## Call:
## glm(formula = cbind(germinated, sown - germinated) ~ Treatment +
##     Treatment:period - 1, family = binomial(link = "probit"),
##     data = des)
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## TreatmentDesorption 130         4.315944   0.281077  15.355   <2e-16 ***
## TreatmentDesorption 150         3.433957   0.344167   9.978   <2e-16 ***
## TreatmentDesorption 170         3.978369   0.427873   9.298   <2e-16 ***
## TreatmentDesorption 130:period -0.032867   0.002419 -13.585   <2e-16 ***
## TreatmentDesorption 150:period -0.077701   0.007848  -9.900   <2e-16 ***
## TreatmentDesorption 170:period -0.261960   0.025678 -10.202   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1086.437  on 14  degrees of freedom
## Residual deviance:   36.411  on  8  degrees of freedom
## AIC: 90.036
##
## Number of Fisher Scoring iterations: 6
```

```
#generating coefficient and confidence intervals for plotting
coefficients.for.plotting1<- cbind(summary(independent)$coef[,c(1)],
confint(independent, level = 0.95)) # level can be modified. Because few
datapoints, intervals are large. Could use eg 75.

## Waiting for profiling to be done...

plot(des$period, des$germinated/des$sown, xlab="Time (days)",
ylab="Proportion germinated", xlim=c(0,200), ylim=c(0,1), mgp=c(2,.5,0),
pch=19, cex=1.2,
col=c("#4477AA","#228833","#EE6677")[unclass(des$Treatment)])
curve(pnorm(coefficients.for.plotting1[1] + coefficients.for.plotting1[4]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=1,lwd=1, col = "#4477AA")
curve(pnorm(coefficients.for.plotting1[2]  +
coefficients.for.plotting1[5]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=1,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting1[3]  +
coefficients.for.plotting1[6]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=1,lwd=1, col = "#EE6677")

### If confidence intervals are to be shown see below code ###

curve(pnorm(coefficients.for.plotting1[1,2] +
coefficients.for.plotting1[4,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")
curve(pnorm(coefficients.for.plotting1[1,3] +
coefficients.for.plotting1[4,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")

curve(pnorm(coefficients.for.plotting1[2,2] +
coefficients.for.plotting1[5,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=2,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting1[2,3] +
coefficients.for.plotting1[5,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=2,lwd=1, col = "#228833")

curve(pnorm(coefficients.for.plotting1[3,2] +
coefficients.for.plotting1[6,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")
curve(pnorm(coefficients.for.plotting1[3,3] +
coefficients.for.plotting1[6,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")

legend(175, 1, title = "RH (%)",
rev(unique(des$chamber.RH)),col=c("#4477AA","#228833","#EE6677"), pch=19, bty
= "n")
title("Independent")
```



```
#dev.off()


###Common intercept (Common Ki)
com.inter <- glm(formula = cbind(germinated, sown-germinated) ~
Treatment:period , family = binomial(link = "probit"),  data = des)
summary(com.inter)

##
## Call:
## glm(formula = cbind(germinated, sown - germinated) ~ Treatment:period,
##     family = binomial(link = "probit"), data = des)
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       3.981964   0.193851   20.54   <2e-16 ***
## TreatmentDesorption 130:period -0.030063   0.001709  -17.59   <2e-16 ***
## TreatmentDesorption 150:period -0.089767   0.004793  -18.73   <2e-16 ***
## TreatmentDesorption 170:period -0.262164   0.013179  -19.89   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1079.71  on 13  degrees of freedom
## Residual deviance:   40.39  on 10  degrees of freedom
## AIC: 90.015
##
## Number of Fisher Scoring iterations: 6

# generating coefficient and confidence intervals for plotting
coefficients.for.plotting2<- cbind(summary(com.inter)$coef[,c(1)],
confint(com.inter, level = 0.95)) # level can be modified. Because few data
points, intervals are large. Could use eg 75.

## Waiting for profiling to be done...

# plot
plot(des$period, des$germinated/des$sown, xlab="Time (days)",
ylab="Porportion germinated", xlim=c(0,200), ylim=c(0,1), mgp=c(2,.5,0),
pch=19, cex=1.2,
col=c("#4477AA","#228833","#EE6677")[unclass(des$Treatment)])

curve(pnorm(coefficients.for.plotting2[1] + coefficients.for.plotting2[2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=1,lwd=1, col = "#4477AA")
curve(pnorm(coefficients.for.plotting2[1]  +
coefficients.for.plotting2[3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=1,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting2[1]  +
coefficients.for.plotting2[4]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=1,lwd=1, col = "#EE6677")

### If confidence intervals are to be shown see below code ###

curve(pnorm(coefficients.for.plotting2[1,2] +
coefficients.for.plotting2[2,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")
curve(pnorm(coefficients.for.plotting2[1,3] +
coefficients.for.plotting2[2,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")
```

```
curve(pnorm(coefficients.for.plotting2[1,2] +
coefficients.for.plotting2[3,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
      add=TRUE, lty=2,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting2[1,3] +
coefficients.for.plotting2[3,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=2,lwd=1, col = "#228833")

curve(pnorm(coefficients.for.plotting2[1,2] +
coefficients.for.plotting2[4,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")
curve(pnorm(coefficients.for.plotting2[1,3] +
coefficients.for.plotting2[4,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")

legend(175, 1, title = "RH (%)",
rev(unique(des$chamber.RH)),col=c("#4477AA","#228833","#EE6677"), pch=19, bty
= "n")
title("Common intercept")
```



```
#dev.off()


#  Table of coefficients for common intercept



###Common slope (common -sigma^-1)
com.slope <- glm(formula = cbind(germinated, sown-germinated) ~ Treatment +
period -1, family = binomial(link = "probit"),  data = des)
summary(com.slope)
```

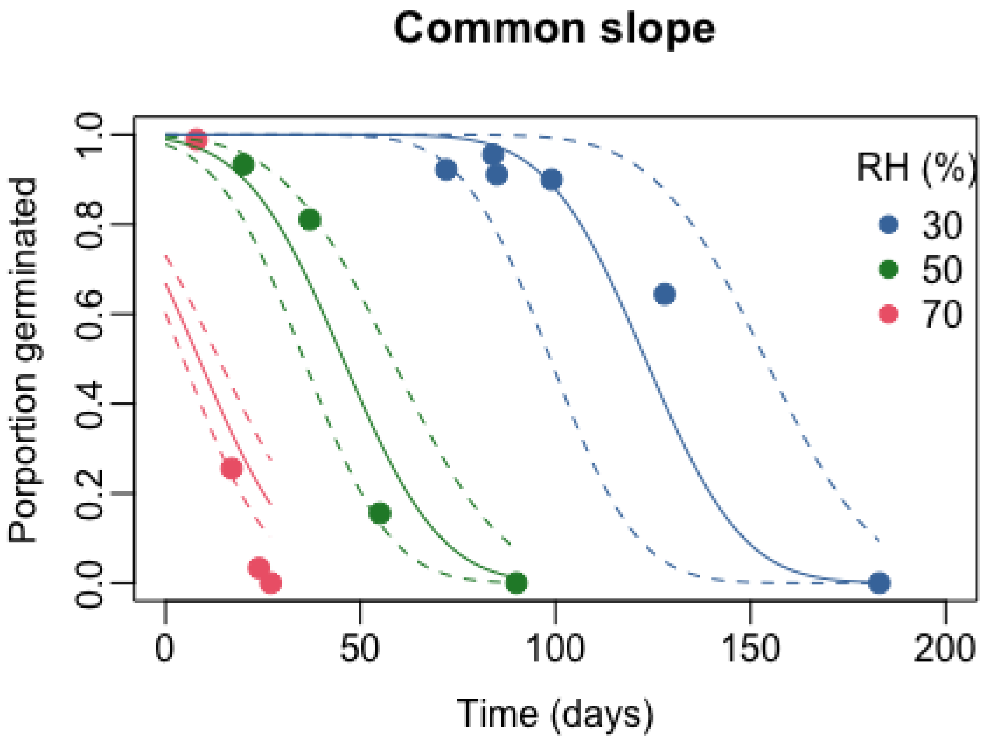
```
##
## Call:
## glm(formula = cbind(germinated, sown - germinated) ~ Treatment +
##     period - 1, family = binomial(link = "probit"), data = des)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## TreatmentDesorption 130  6.24593    0.35392  17.648  < 2e-16 ***
## TreatmentDesorption 150  2.31196    0.15725  14.702  < 2e-16 ***
## TreatmentDesorption 170  0.43695    0.09078   4.813 1.48e-06 ***
## period                  -0.05077    0.00312 -16.270  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1086.44  on 14  degrees of freedom
## Residual deviance:  255.08  on 10  degrees of freedom
## AIC: 304.71
##
## Number of Fisher Scoring iterations: 6
```

```
#generating coefficient and confidence intervals for plotting
coefficients.for.plotting3<- cbind(summary(com.slope)$coef[,c(1)],
confint(com.slope, level = 0.95)) # level can be modified. Because few data
points, intervals are large. Could use eg 75.
```

```
## Waiting for profiling to be done...
```

```
# plot
plot(des$period, des$germinated/des$sown, xlab="Time (days)",
ylab="Porportion germinated", xlim=c(0,200), ylim=c(0,1), mgp=c(2,.5,0),
pch=19, cex=1.2,
col=c("#4477AA","#228833","#EE6677")[unclass(des$Treatment)])

curve(pnorm(coefficients.for.plotting3[1] + coefficients.for.plotting3[4]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=1,lwd=1, col = "#4477AA")
```

```
curve(pnorm(coefficients.for.plotting3[2]  +
coefficients.for.plotting3[4]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=1,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting3[3]  +
coefficients.for.plotting3[4]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=1,lwd=1, col = "#EE6677")

### If confidence intervals are to be shown see below code ###

curve(pnorm(coefficients.for.plotting3[1,2] +
coefficients.for.plotting3[4,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")
curve(pnorm(coefficients.for.plotting3[1,3] +
coefficients.for.plotting3[4,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 130")$period),
add=TRUE, lty=2,lwd=1, col = "#4477AA")

curve(pnorm(coefficients.for.plotting3[2,2] +
coefficients.for.plotting3[4,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
      add=TRUE, lty=2,lwd=1, col = "#228833")
curve(pnorm(coefficients.for.plotting3[2,3] +
coefficients.for.plotting3[4,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 150")$period),
add=TRUE, lty=2,lwd=1, col = "#228833")

curve(pnorm(coefficients.for.plotting3[3,2] +
coefficients.for.plotting3[4,2]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")
curve(pnorm(coefficients.for.plotting3[3,3] +
coefficients.for.plotting3[4,3]*x),
      from = 0, to = max(subset(des, Treatment=="Desorption 170")$period),
add=TRUE, lty=2,lwd=1, col = "#EE6677")

legend(175, 1, title = "RH (%)",
rev(unique(des$chamber.RH)),col=c("#4477AA","#228833","#EE6677"), pch=19, bty
= "n")
title("Common slope")
```

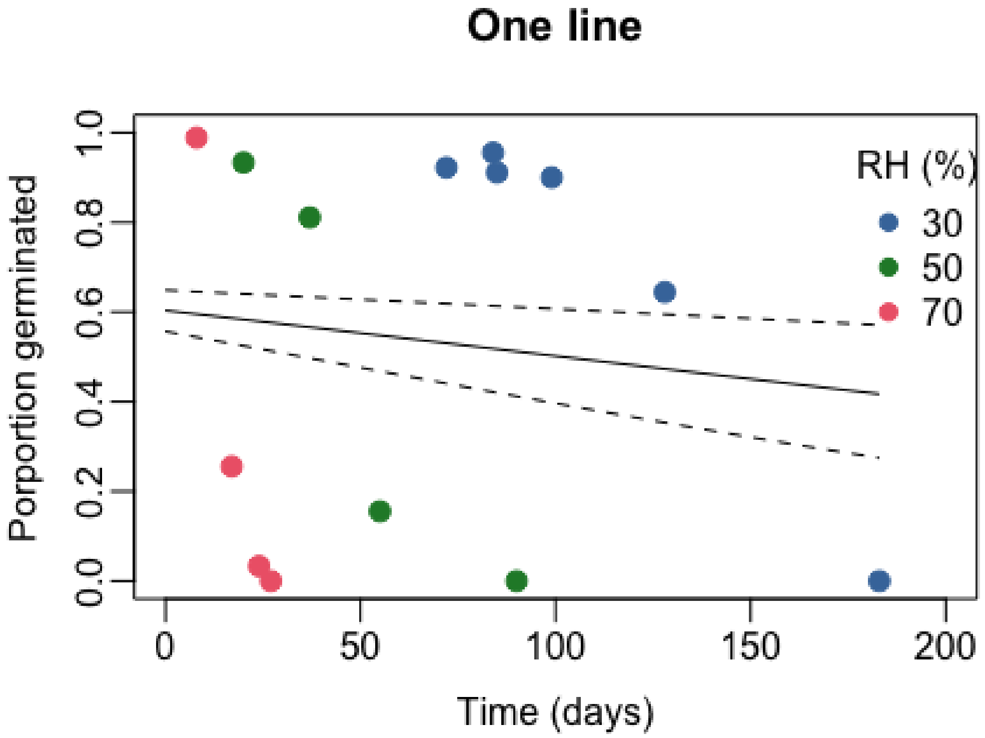


```
#dev.off()


## one line
one.line <- glm(formula = cbind(germinated, sown-germinated) ~ period ,
family = binomial(link = "probit"), data = des)
summary(one.line)

##
## Call:
## glm(formula = cbind(germinated, sown - germinated) ~ period,
##     family = binomial(link = "probit"), data = des)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.262001   0.060829   4.307 1.65e-05 ***
## period      -0.002573   0.000743  -3.464 0.000533 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1079.7  on 13  degrees of freedom
## Residual deviance: 1067.8  on 12  degrees of freedom
## AIC: 1113.4
##
## Number of Fisher Scoring iterations: 5
```

```r
#generating coefficient and confidence intervals for plotting
coefficients.for.plotting4<- cbind(summary(one.line)$coef[,c(1)],
confint(one.line, level = 0.95)) # level can be modified. Because few data
points, intervals are large. Could use eg 75.
```

```
## Waiting for profiling to be done...
```

```r
# plot
plot(des$period, des$germinated/des$sown, xlab="Time (days)",
ylab="Porportion germinated", xlim=c(0,200), ylim=c(0,1), mgp=c(2,.5,0),
pch=19, cex=1.2,
col=c("#4477AA","#228833","#EE6677")[unclass(des$Treatment)])

curve(pnorm(coefficients.for.plotting4[1] + coefficients.for.plotting4[2]*x),
      from = 0, to = max(des$period), add=TRUE, lty=1,lwd=1)

### If confidence intervals are to be shown see below code ###
curve(pnorm(coefficients.for.plotting4[1,2] +
coefficients.for.plotting4[2,2]*x),
      from = 0, to = max(des$period), add=TRUE, lty=2,lwd=1)
curve(pnorm(coefficients.for.plotting4[1,3] +
coefficients.for.plotting4[2,3]*x),
      from = 0, to = max(des$period), add=TRUE, lty=2,lwd=1)

legend(175, 1, title = "RH (%)",
rev(unique(des$chamber.RH)),col=c("#4477AA","#228833","#EE6677"), pch=19, bty
= "n")
title("One line")
```



```
#dev.off()

# Table of coefficients for one line model
table.oneline<-
signif(as.data.frame(summary(one.line)$coefficients[,c(1,2,4)]), 4)
table.oneline[2,1]<- signif(-1/table.oneline[2,1], 4)

# dose.p
table.oneline[3,1]<- signif(dose.p(one.line, cf = 1:2, p = 0.5), 4)
table.oneline[3,2]<- signif(unname(attributes(dose.p(one.line, cf = 1:2, p =
0.5))$SE[, 1]), 4)
table.oneline[4,1]<- signif(dose.p(one.line, cf = 1:2, p = 0.85), 4)
table.oneline[4,2]<- signif(unname(attributes(dose.p(one.line, cf = 1:2, p =
0.85))$SE[, 1]), 4)

row.names(table.oneline)<-c("Ki", "sigma", "p50", "p85")
table.oneline$`Std. Error` <- round(table.oneline$`Std. Error`, 4)

table.oneline[3,3] <- "-"
table.oneline[4,3] <- "-"
table.oneline$AIC<- "-"
table.oneline$AIC[1]<- round(summary(one.line)$aic, 4)
```

```
# Call the table
knitr::kable(table.oneline)
```

	Estimate	Std. Error	Pr(>|z|)	AIC
Ki	0.262	0.0608	1.653e-05	1113.4349
sigma	388.700	0.0007	0.000533	-
p50	101.800	17.1400	-	-
p85	-301.000	107.0000	-	-

```
### Compare models ###

# Create deviance table
dev.table<-data.frame(
  Model = c("independent", "Common.intercept", "Common.slope", "One.line"),
  Deviance = c(independent$deviance, com.inter$deviance, com.slope$deviance,
one.line$deviance),
  df = c(independent$df.residual,
com.inter$df.residual,com.slope$df.residual, one.line$df.residual),
  AIC = c(independent$aic, com.inter$aic, com.slope$aic, one.line$aic)
)

# Call the table.
knitr::kable(dev.table, row.names = FALSE)
```

Model	Deviance	df	AIC
independent	36.41096	8	90.03587
Common.intercept	40.39037	10	90.01528
Common.slope	255.08134	10	304.70624
One.line	1067.81000	12	1113.43490

```
# Here we demonstrate pairwise comparisons of common slope and common
intercept with  independent and one line.
# However, in practice this would only be done with the models that have the
lowest and closest in their AIC
# (e.g. independent and common intercept in this data presented).

# Common intercept vs Independent
df.change1<-com.inter$df.residual - independent$df.residual
df.change1

## [1] 2

F1<-((com.inter$deviance-
independent$deviance)/df.change1)/(independent$deviance/independent$df.residu
al)
F1

## [1] 0.437166
```

```
F1prob<-pf(F1, df.change1, independent$df.residual, lower.tail = FALSE) #
note the lower.tail = False is not default
F1prob<-format(signif(F1prob, 4), nsmall = 4)
F1prob

## [1] "0.6604"

F1sum<-c(df.change1, F1, F1prob)
F1sum

## [1] "2"                  "0.437166049341884" "0.6604"

# Common slope vs Independent
df.change2<-com.slope$df.residual - independent$df.residual
df.change2

## [1] 2

F2<-((com.slope$deviance-
independent$deviance)/df.change2)/(independent$deviance/independent$df.residu
al)
F2

## [1] 24.02248

F2prob<-pf(F2, df.change2, independent$df.residual, lower.tail = FALSE) #
note the lower.tail = False is not default
F2prob<-format(signif(F2prob, 4), nsmall = 4)
F2prob

## [1] "0.0004152"

F2sum<-c(df.change2, F2, F2prob)
F2sum

## [1] "2"                 "24.0224765318435" "0.0004152"

# Common intercept vs One line
df.change3<-one.line$df.residual - com.inter$df.residual
df.change3

## [1] 2

F3<-((one.line$deviance-
com.inter$deviance)/df.change3)/(com.inter$deviance/com.inter$df.residual)
F3

## [1] 127.1862

F3prob<-pf(F3, df.change3, one.line$df.residual, lower.tail = FALSE) # note
the lower.tail = False is not default
F3prob<-format(signif(F3prob, 4), nsmall = 4)
F3prob
```

```
## [1] "8.359e-09"

F3sum<-c(df.change3, F3, F3prob)
F3sum

## [1] "2"               "127.18620508823" "8.359e-09"

# Common slope vs One line
df.change4<-one.line$df.residual - com.slope$df.residual
df.change4

## [1] 2

F4<-((one.line$deviance-
com.slope$deviance)/df.change4)/(com.slope$deviance/com.slope$df.residual)
F4

## [1] 15.93077

F4prob<-pf(F4, df.change4, com.slope$df.residual, lower.tail = FALSE) # note
the lower.tail = False is not default
F4prob<-format(signif(F4prob, 4), nsmall = 4)
F4prob

## [1] "0.0007779"

F4sum<-c(df.change4, F4, F4prob)
F4sum

## [1] "2"                "15.9307746494971" "0.0007779"

# Combine into one summary table. Report in appendices
# Most important value is F probability
# If F probability > alpha, you can accept the model
# In this case there's no significant difference between common intercept vs
independent
F.table1<-data.frame(Model.comparison = c("Common intercept vs Independent",
                                          "Common slope vs Independent",
                                          "Common intercept vs One line",
                                          "Common slope vs One line "),
                     "DF change" = c(df.change1, df.change2, df.change3,
df.change4),
                     "F statistic" = c(F1, F2, F3, F4),
                     "F probability" = c(F1prob, F2prob, F3prob, F4prob))
# call the table
knitr::kable(F.table1)
```

Model.comparison	DF.change	F.statistic	F.probability
Common intercept vs Independent	2	0.437166	0.6604
Common slope vs Independent	2	24.022476	0.0004152
Common intercept vs One line	2	127.186205	8.359e-09

Model.comparison	DF.change	F.statistic	F.probability
Common slope vs One line	2	15.930775	0.0007779

```
rm(df.change1, df.change2, df.change3, df.change4, F1, F2, F3, F4, F1prob,
F2prob, F3prob, F4prob, F1sum, F2sum, F3sum, F4sum)
# N.B. Common line  = one line
# N.B. Independent

# Note: A quick way to view the deviance between two models is with function
anova()
# adding, test = test="Chisq", will output a p-value.
anova(independent, com.inter,  test="Chisq")

## Analysis of Deviance Table
##
## Model 1: cbind(germinated, sown - germinated) ~ Treatment +
Treatment:period -
##     1
## Model 2: cbind(germinated, sown - germinated) ~ Treatment:period
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         8     36.411
## 2        10     40.390 -2  -3.9794   0.1367

###  Table of coefficients
### Presented for only the selected model. In this case the independent
model.

table.independent<-
round(as.data.frame(summary(independent)$coefficients[,c(1,2,4)]), 3)
table.independent[c(4:6),1]<- signif(-1/table.independent[c(4:6),1], 3)
table.independent[c(4:6),2] <- table.independent[
c(4:6),2]*table.independent[ c(4:6),1]

# dose.p
table.independent[7,1]<- signif(dose.p(independent, cf = c(1,4), p = 0.5), 3)
table.independent[7,2]<- signif(unname(attributes(dose.p(independent, cf =
c(1,4), p = 0.5))$SE[, 1]), 3)
table.independent[8,1]<- signif(dose.p(independent, cf = c(2,5), p = 0.5), 3)
table.independent[8,2]<- signif(unname(attributes(dose.p(independent, cf =
c(2,5), p = 0.5))$SE[, 1]), 3)
table.independent[9,1]<- signif(dose.p(independent, cf = c(3,6), p = 0.5), 3)
table.independent[9,2]<- signif(unname(attributes(dose.p(independent, cf =
c(3,6), p = 0.5))$SE[, 1]), 3)

row.names(table.independent)<-c( "Ki-Desorption 130", "Ki-Desorption
150","Ki-Desorption 170","sigma-Desorption 130", "sigma-Desorption
150","sigma-Desorption 170", "p50-Desorption 130", "p50-Desorption 150","p50-
Desorption 170")
table.independent$`Std. Error` <- round(table.independent$`Std. Error`, 3)
```

```r
table.independent[c(7:9),3] <- "-"
table.independent$AIC<- "-"
table.independent$AIC[1]<- round(summary(independent)$aic, 3)
names(table.independent)<- c("Estimate", "SE", "P", "AIC")
table.independent$P <- ifelse(table.independent$P=="0", "<0.001",
table.independent$P)

# Call the table.
knitr::kable(table.independent)
```

	Estimate	SE	P	AIC
Ki-Desorption 130	4.316	0.281	<0.001	90.036
Ki-Desorption 150	3.434	0.344	<0.001	-
Ki-Desorption 170	3.978	0.428	<0.001	-
sigma-Desorption 130	30.300	0.061	<0.001	-
sigma-Desorption 150	12.800	0.102	<0.001	-
sigma-Desorption 170	3.820	0.099	<0.001	-
p50-Desorption 130	131.000	2.730	-	-
p50-Desorption 150	44.200	1.250	-	-
p50-Desorption 170	15.200	0.449	-	-