Model Validation and coefficient calculation

Marc Pérez

2025-05-22

## Context

To explore, whether the proposed mechanisms and experiments to assess their dynamic, in a first step the Treatment levels and for all sites were analyzed. The experiments were conducted as displayed in the original paper of Flossmann & Richter with adjustments according to developments in technique and available equipment of the soil laboratory. Instead of the CAL-method, the Olsen-method was used to measure and estimate the quantity of P.

## Model of P-release after Flossman & Richter

The constant denotes the amount of semi-labile P and was originally estimated as . Subsequently the DE is solved exactly, since the soil is as mixed with deionized water, it was assumed that

## Exact Solution

for we receive:

If we set we receive:

## Linearization

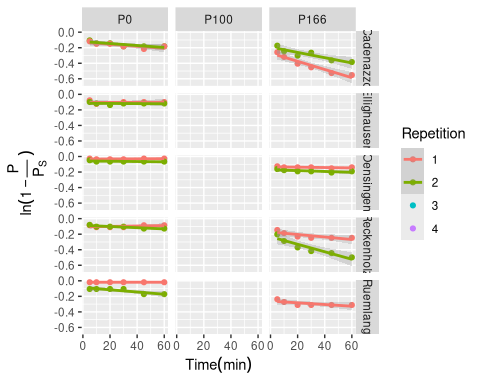
Now we linearize the DE, so that a linear model can be employed to test the relation and estimate the parameters of interest:

Given ,

## Setup and preparation of dataset

Now we can see, whether our linearized model displays a linear relation.

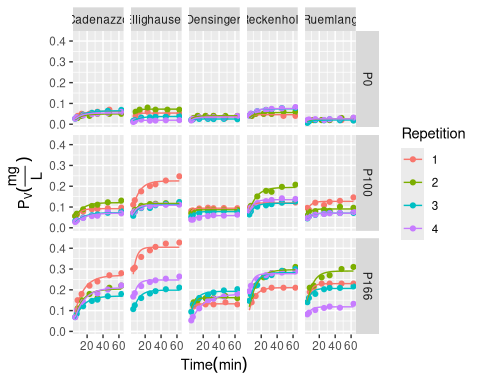
Call:  
 Model: Y1 ~ t.min. | uid   
 Data: d[d$Repetition == 1 | d$Repetition == 2, ]   
  
Coefficients:  
 (Intercept)   
 Estimate Std. Error t value Pr(>|t|)  
Cadenazzo\_P0\_1 -0.12891945 0.01537006 -8.387702 4.332766e-12  
Cadenazzo\_P0\_2 -0.12037045 0.01537006 -7.831491 4.433395e-11  
Cadenazzo\_P100\_1 NA NA NA NA  
Cadenazzo\_P100\_2 NA NA NA NA  
Cadenazzo\_P166\_1 -0.26932199 0.01537006 -17.522512 6.499702e-27  
Cadenazzo\_P166\_2 -0.19243796 0.01537006 -12.520316 2.550625e-19  
Ellighausen\_P0\_1 -0.10464296 0.01537006 -6.808236 3.136905e-09  
Ellighausen\_P0\_2 -0.11438112 0.01537006 -7.441815 2.257472e-10  
Ellighausen\_P100\_1 NA NA NA NA  
Ellighausen\_P100\_2 NA NA NA NA  
Ellighausen\_P166\_1 NA NA NA NA  
Oensingen\_P0\_1 -0.03432646 0.01537006 -2.233333 2.882091e-02  
Oensingen\_P0\_2 -0.05745952 0.01537006 -3.738407 3.819350e-04  
Oensingen\_P100\_1 NA NA NA NA  
Oensingen\_P100\_2 NA NA NA NA  
Oensingen\_P166\_1 -0.13275856 0.01537006 -8.637481 1.527196e-12  
Oensingen\_P166\_2 -0.17051390 0.01537006 -11.093902 6.616653e-17  
Reckenholz\_P0\_1 -0.10545869 0.01537006 -6.861308 2.519112e-09  
Reckenholz\_P0\_2 -0.08557888 0.01537006 -5.567897 4.753375e-07  
Reckenholz\_P100\_1 NA NA NA NA  
Reckenholz\_P100\_2 NA NA NA NA  
Reckenholz\_P166\_1 -0.17172348 0.01537006 -11.172600 4.839473e-17  
Reckenholz\_P166\_2 -0.23296391 0.01537006 -15.156998 1.712692e-23  
Ruemlang\_P0\_1 -0.01851905 0.01537006 -1.204878 2.324269e-01  
Ruemlang\_P0\_2 -0.08675331 0.01537006 -5.644307 3.515958e-07  
Ruemlang\_P100\_1 NA NA NA NA  
Ruemlang\_P100\_2 NA NA NA NA  
Ruemlang\_P166\_1 -0.26153690 0.01537006 -17.016002 3.315417e-26  
Ruemlang\_P166\_2 NA NA NA NA  
 t.min.   
 Estimate Std. Error t value Pr(>|t|)  
Cadenazzo\_P0\_1 -1.318800e-03 0.0004483906 -2.941186e+00 4.466020e-03  
Cadenazzo\_P0\_2 -1.272378e-03 0.0004483906 -2.837654e+00 5.984783e-03  
Cadenazzo\_P100\_1 NA NA NA NA  
Cadenazzo\_P100\_2 NA NA NA NA  
Cadenazzo\_P166\_1 -5.270369e-03 0.0004483906 -1.175397e+01 4.905164e-18  
Cadenazzo\_P166\_2 -3.394812e-03 0.0004483906 -7.571105e+00 1.316077e-10  
Ellighausen\_P0\_1 4.952586e-05 0.0004483906 1.104525e-01 9.123759e-01  
Ellighausen\_P0\_2 -1.260933e-04 0.0004483906 -2.812130e-01 7.794010e-01  
Ellighausen\_P100\_1 NA NA NA NA  
Ellighausen\_P100\_2 NA NA NA NA  
Ellighausen\_P166\_1 NA NA NA NA  
Oensingen\_P0\_1 1.049070e-04 0.0004483906 2.339634e-01 8.157164e-01  
Oensingen\_P0\_2 -1.837559e-04 0.0004483906 -4.098121e-01 6.832320e-01  
Oensingen\_P100\_1 NA NA NA NA  
Oensingen\_P100\_2 NA NA NA NA  
Oensingen\_P166\_1 -2.320568e-04 0.0004483906 -5.175327e-01 6.064639e-01  
Oensingen\_P166\_2 -5.531502e-04 0.0004483906 -1.233635e+00 2.215861e-01  
Reckenholz\_P0\_1 2.780943e-04 0.0004483906 6.202053e-01 5.371956e-01  
Reckenholz\_P0\_2 -7.752286e-04 0.0004483906 -1.728914e+00 8.836252e-02  
Reckenholz\_P100\_1 NA NA NA NA  
Reckenholz\_P100\_2 NA NA NA NA  
Reckenholz\_P166\_1 -1.609218e-03 0.0004483906 -3.588876e+00 6.216266e-04  
Reckenholz\_P166\_2 -4.831330e-03 0.0004483906 -1.077482e+01 2.367928e-16  
Ruemlang\_P0\_1 8.878899e-20 0.0004483906 1.980171e-16 1.000000e+00  
Ruemlang\_P0\_2 -1.438957e-03 0.0004483906 -3.209160e+00 2.032261e-03  
Ruemlang\_P100\_1 NA NA NA NA  
Ruemlang\_P100\_2 NA NA NA NA  
Ruemlang\_P166\_1 -1.090605e-03 0.0004483906 -2.432266e+00 1.764226e-02  
Ruemlang\_P166\_2 NA NA NA NA  
  
Residual standard error: 0.02119011 on 68 degrees of freedom



If the parameter for the plateau could be estimated directly by using a non-linear non-least-squares model, we could omit the Olsen-measurement in the future.

LG: our nls is very sensitive to moderatly high Pv.mg.L at small time points. Since the … disolves already before we start measureing, we will add 3 min to our time-measurement.

Res <- nlsList(Pv.mg.L. ~ PS \* (1 - exp(-k \* (t.dt))) | uid, d[, c("Pv.mg.L.", "uid", "t.dt")], start=list(PS=0.1,k=0.2))  
# summary(Res)  
# d$nls\_pred <- predict(Res)  
  
# Extract coefficients from the nlsList results  
nls\_coefs <- coef(Res)  
nls\_coefs$uid <- rownames(nls\_coefs)  
  
# Merge coefficients back to the main dataset  
d\_plot <- merge(d, nls\_coefs, by = "uid")  
  
# Most straightforward approach - create curves manually  
time\_seq <- seq(min(d$t.dt, na.rm = TRUE), max(d$t.dt, na.rm = TRUE), length.out = 100)  
  
# Create prediction data  
pred\_data <- d\_plot %>%  
 select(uid, Site, Treatment, Repetition, PS, k) %>%  
 distinct() %>%  
 crossing(t.dt = time\_seq) %>%  
 mutate(pred\_Pv = PS \* (1 - exp(-k \* (t.dt))))  
  
# Final plot  
p1 <- ggplot() +  
 geom\_point(data = d\_plot, aes(y = Pv.mg.L., x = t.dt, col = Repetition)) +  
 geom\_line(data = pred\_data, aes(x = t.dt, y = pred\_Pv, col = Repetition), size = 0.5) +  
 facet\_grid(Treatment ~ Site) +  
 labs(x = TeX("$Time (min)$"),  
 y = TeX("$P\_{V}(\\frac{mg}{L})$")); suppressWarnings(print(p1))

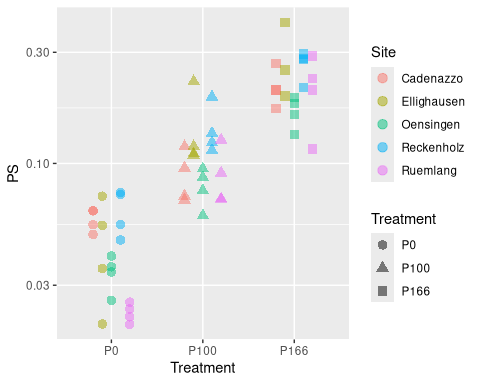


d$ui <- interaction(d$Site, d$Treatment)  
  
nlme.coef.avg <- list()  
nlme.coef <- list()  
for (lvl in levels(d$ui)){  
 d.tmp <- subset(d, ui == lvl)  
 # first get nlsList coefs for comparison only (unused)  
 temp\_nls <- coef(nlsList(Pv.mg.L. ~ PS \* (1 - exp(-k \* t.dt)) | uid,   
 d.tmp[, c("Pv.mg.L.", "uid", "t.dt")],   
 start = list(PS = 0.1, k = 0.2)))  
 nlsList\_coefs <- c(apply(temp\_nls, 2, \(x) c(mean=mean(x), sd=sd(x))))  
 names(nlsList\_coefs) <- c("PS.mean", "PS.sd", "k.mean", "k.sd")  
  
 # now do the real thing  
 model4 <- nlme(Pv.mg.L. ~ PS \* (1 - exp(-k \* t.dt)),  
 fixed = PS + k ~ 1,  
 random = PS + k ~ 1 | uid,  
 data = d.tmp[, c("Pv.mg.L.", "uid", "t.dt")],  
 start = c(PS = 0.05, k = 0.12),  
 control = nlmeControl(maxIter = 200))  
 coef(model4)  
 fixef <- model4$coefficients$fixed  
 ranefs <- ranef(model4)  
 colnames(ranefs) <- paste0("ranef\_",colnames(ranefs))  
 nlme.coef[[lvl]] <- cbind(coef(model4), ranefs, Rep=1:nrow(ranef(model4)), ui=lvl, Site=d.tmp[1, "Site"], Treatment=d.tmp[1, "Treatment"], uid = rownames(coef(model4)))  
 nlme.coef.avg[[lvl]] <- data.frame(PS=fixef["PS"], k=fixef["k"], ui=lvl, Site=d.tmp[1, "Site"], Treatment=d.tmp[1, "Treatment"], uid = d.tmp$uid)  
}  
  
nlme.coef.avg <- do.call(rbind, nlme.coef.avg)  
# folgendes datenset wollen wir benutzen um ihn mit dem Boden zu kombinieren  
nlme.coef <- do.call(rbind, nlme.coef)

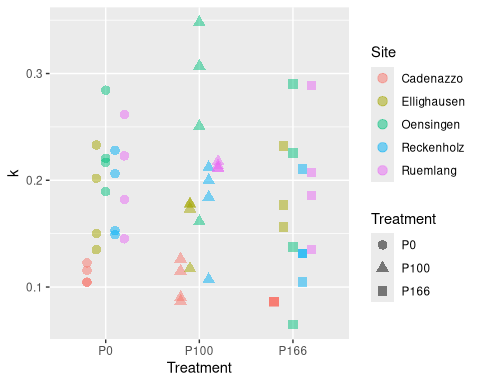
LG: hier machen wir folgendes:

1. Visualisiere Daten
2. for k\*PS use sqrt-scale
3. Erkenne, dass keine offenslichtichen verletzuungen für ein lineares modell vorhanden sind
4. fitte ordinary linear squares model, with Treatment as the factor of interest and Site as covariate (analougous to paired t-test and equivalent to anova with Site as block factor)
5. Perform a classical Type II anova (using the car::Anova function)
6. Perform (post-hoc) TukeyHSD test (using multcomp package)

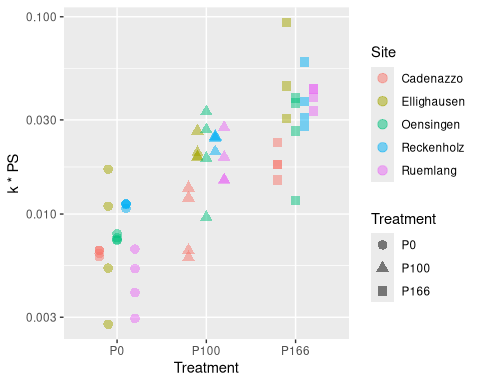
points <- geom\_point(position=position\_dodge(width=0.5), size = 3, alpha = 0.5)  
  
ggplot(nlme.coef, aes(y=PS , x=Treatment, col=Site, pch=Treatment)) + points + scale\_y\_log10()



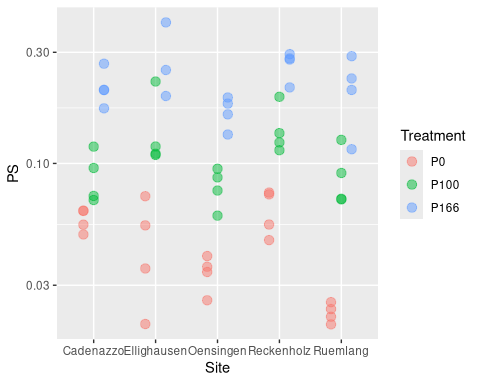
ggplot(nlme.coef, aes(y=k , x=Treatment, col=Site, pch=Treatment)) + points



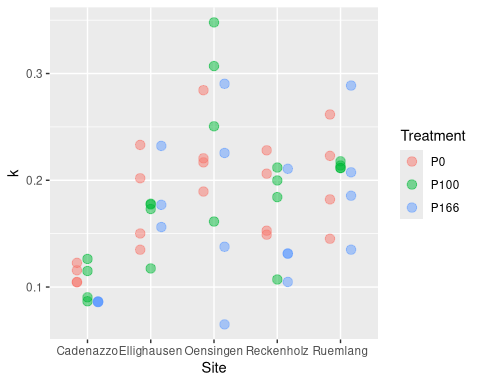
ggplot(nlme.coef, aes(y=k\*PS, x=Treatment, col=Site, pch=Treatment)) + points + scale\_y\_log10()



ggplot(nlme.coef, aes(y=PS , x=Site, col=Treatment)) + points + scale\_y\_log10()



ggplot(nlme.coef, aes(y=k , x=Site, col=Treatment)) + points



ggplot(nlme.coef, aes(y=k\*PS, x=Site, col=Treatment)) + points + scale\_y\_log10()



# k PS macht von der interpretation her Sinn  
# aber PS ist log-normal verteilt  
  
  
fit.PS <- lm(log(PS) ~ Treatment + Site, nlme.coef)  
fit.k <- lm(k ~ Treatment + Site, nlme.coef)  
fit.kPS <- lm(I(log(k\*PS)) ~ Treatment + Site, nlme.coef)  
  
  
Anova(fit.PS)

Anova Table (Type II tests)  
  
Response: log(PS)  
 Sum Sq Df F value Pr(>F)   
Treatment 27.6260 2 154.7655 < 2.2e-16 \*\*\*  
Site 3.0383 4 8.5104 2.324e-05 \*\*\*  
Residuals 4.6411 52   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(glht(fit.PS, mcp(Treatment = "Tukey")))

Simultaneous Tests for General Linear Hypotheses  
  
Multiple Comparisons of Means: Tukey Contrasts  
  
  
Fit: lm(formula = log(PS) ~ Treatment + Site, data = nlme.coef)  
  
Linear Hypotheses:  
 Estimate Std. Error t value Pr(>|t|)   
P100 - P0 == 0 0.91948 0.09447 9.733 <1e-09 \*\*\*  
P166 - P0 == 0 1.68127 0.09580 17.550 <1e-09 \*\*\*  
P166 - P100 == 0 0.76179 0.09580 7.952 <1e-09 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
(Adjusted p values reported -- single-step method)

# Fazit: PS wird von treatment stark beeinfluss, k eher nicht (dafür von site)  
Anova(fit.k)

Anova Table (Type II tests)  
  
Response: k  
 Sum Sq Df F value Pr(>F)   
Treatment 0.007374 2 1.6124 0.2092   
Site 0.108427 4 11.8547 6.442e-07 \*\*\*  
Residuals 0.118902 52   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(glht(fit.k, mcp(Treatment = "Tukey")))

Simultaneous Tests for General Linear Hypotheses  
  
Multiple Comparisons of Means: Tukey Contrasts  
  
  
Fit: lm(formula = k ~ Treatment + Site, data = nlme.coef)  
  
Linear Hypotheses:  
 Estimate Std. Error t value Pr(>|t|)  
P100 - P0 == 0 0.003111 0.015121 0.206 0.977  
P166 - P0 == 0 -0.022243 0.015334 -1.451 0.323  
P166 - P100 == 0 -0.025354 0.015334 -1.653 0.233  
(Adjusted p values reported -- single-step method)

Anova(fit.kPS)

Anova Table (Type II tests)  
  
Response: I(log(k \* PS))  
 Sum Sq Df F value Pr(>F)   
Treatment 22.4177 2 68.5970 2.609e-15 \*\*\*  
Site 3.9298 4 6.0124 0.0004703 \*\*\*  
Residuals 8.4969 52   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(glht(fit.kPS, mcp(Treatment = "Tukey")))

Simultaneous Tests for General Linear Hypotheses  
  
Multiple Comparisons of Means: Tukey Contrasts  
  
  
Fit: lm(formula = I(log(k \* PS)) ~ Treatment + Site, data = nlme.coef)  
  
Linear Hypotheses:  
 Estimate Std. Error t value Pr(>|t|)   
P100 - P0 == 0 0.9127 0.1278 7.140 < 1e-04 \*\*\*  
P166 - P0 == 0 1.5035 0.1296 11.599 < 1e-04 \*\*\*  
P166 - P100 == 0 0.5908 0.1296 4.558 0.000116 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
(Adjusted p values reported -- single-step method)

Results:

1. for PS Treatment explains a lot, and site not so much. c.f. plot for a monotonous relationship
2. for k, the Treatment seems to be little relevant

# new Data set, that gives info about Soil  
allP <- tryCatch(readRDS("./data/all\_P.rds"))  
if (inherits(d, "try-error")){  
 allP <- tryCatch(readRDS("~/Documents/Master Thesis/Master-Thesis-P-kinetics/data/all\_P.rds"))  
}  
allP$rep <- allP$rep %>% as.roman() %>% as.integer()  
allP$uid <- paste(allP$location,allP$treatment\_ID,as.character(allP$rep),sep = "\_")

# 1. merge this with nlme.coef  
nlme.coef$kPS <- nlme.coef$k \* nlme.coef$PS  
  
  
nlme.coef.mrg <- merge(nlme.coef,allP[allP$year>=2017,],by = "uid")  
# add log-transformed versions  
nlme.coef.mrg$kPS\_log <- log(nlme.coef.mrg$kPS)  
nlme.coef.mrg$PS\_log <- log(nlme.coef.mrg$PS)  
nlme.coef.mrg$soil\_0\_20\_P\_AAE10\_log <- log(nlme.coef.mrg$soil\_0\_20\_P\_AAE10)  
nlme.coef.mrg$soil\_0\_20\_P\_CO2\_log <- log(nlme.coef.mrg$soil\_0\_20\_P\_CO2)  
  
nrow(nlme.coef.mrg)

[1] 471

names(nlme.coef.mrg)

[1] "uid" "PS"   
 [3] "k" "ranef\_PS"   
 [5] "ranef\_k" "Rep"   
 [7] "ui" "Site"   
 [9] "Treatment" "kPS"   
 [11] "location" "LtE\_name"   
 [13] "site" "WGS84\_N"   
 [15] "WGS84\_E" "year"   
 [17] "year\_fac" "year\_sampling"   
 [19] "cropclass" "crop"   
 [21] "crop\_abr" "nested\_trial"   
 [23] "rep" "block"   
 [25] "treatment\_ID" "treatment\_serie"   
 [27] "fieldplot" "plot\_nr"   
 [29] "plot\_nr\_serie" "sow\_date"   
 [31] "harv1\_date" "harv2\_date"   
 [33] "harv3\_date" "harv4\_date"   
 [35] "harv5\_date" "fert\_N\_tot"   
 [37] "fert\_P\_tot" "fert\_K\_tot"   
 [39] "fert\_Mg\_tot" "fert\_Ca\_tot"   
 [41] "soil\_0\_20\_Corg" "soil\_0\_20\_clay"   
 [43] "soil\_0\_20\_silt" "soil\_0\_20\_pH\_H2O"   
 [45] "soil\_0\_20\_P\_test" "soil\_0\_20\_K\_test"   
 [47] "soil\_0\_20\_Mg\_test" "soil\_0\_20\_P\_AAE10"   
 [49] "soil\_0\_20\_K\_AAE10" "soil\_0\_20\_Mg\_AAE10"   
 [51] "soil\_0\_20\_Ca\_AAE10" "soil\_0\_20\_P\_H2O10"   
 [53] "soil\_0\_20\_K\_H2O10" "soil\_0\_20\_Mg\_H2O10"   
 [55] "soil\_0\_20\_Ca\_H2O10" "soil\_0\_20\_presample\_lime"   
 [57] "soil\_0\_20\_humus" "rollMean\_soil\_0\_20\_Corg"   
 [59] "rollMean\_soil\_0\_20\_clay" "rollMean\_soil\_0\_20\_silt"   
 [61] "rollMean\_soil\_0\_20\_pH\_H2O" "rollMean\_soil\_0\_20\_P\_test"   
 [63] "rollMean\_soil\_0\_20\_K\_test" "rollMean\_soil\_0\_20\_Mg\_test"   
 [65] "rollMean\_soil\_0\_20\_P\_AAE10" "rollMean\_soil\_0\_20\_K\_AAE10"   
 [67] "rollMean\_soil\_0\_20\_Mg\_AAE10" "rollMean\_soil\_0\_20\_Ca\_AAE10"   
 [69] "rollMean\_soil\_0\_20\_P\_H2O10" "rollMean\_soil\_0\_20\_K\_H2O10"   
 [71] "rollMean\_soil\_0\_20\_Mg\_H2O10" "rollMean\_soil\_0\_20\_Ca\_H2O10"   
 [73] "rollMean\_soil\_0\_20\_presample\_lime" "rollMean\_soil\_0\_20\_humus"   
 [75] "soil\_0\_20\_K\_CO2" "soil\_0\_20\_P\_CO2"   
 [77] "soil\_0\_20\_Mg\_CaCl2" "annual\_yield\_mp\_DM"   
 [79] "annual\_yield\_bp\_DM" "annual\_total\_biomass\_maincrop\_DM"   
 [81] "harv1\_mp\_Nuptake" "harv1\_mp\_Puptake"   
 [83] "harv1\_mp\_Kuptake" "harv1\_mp\_Cauptake"   
 [85] "harv1\_mp\_Mguptake" "harv2\_mp\_Nuptake"   
 [87] "harv2\_mp\_Puptake" "harv2\_mp\_Kuptake"   
 [89] "harv2\_mp\_Cauptake" "harv2\_mp\_Mguptake"   
 [91] "harv3\_mp\_Nuptake" "harv3\_mp\_Puptake"   
 [93] "harv3\_mp\_Kuptake" "harv3\_mp\_Cauptake"   
 [95] "harv3\_mp\_Mguptake" "harv4\_mp\_Nuptake"   
 [97] "harv4\_mp\_Puptake" "harv4\_mp\_Kuptake"   
 [99] "harv4\_mp\_Cauptake" "harv4\_mp\_Mguptake"   
[101] "harv5\_mp\_Nuptake" "harv5\_mp\_Puptake"   
[103] "harv5\_mp\_Kuptake" "harv5\_mp\_Cauptake"   
[105] "harv5\_mp\_Mguptake" "harv6\_mp\_Nuptake"   
[107] "harv6\_mp\_Puptake" "harv6\_mp\_Kuptake"   
[109] "harv6\_mp\_Cauptake" "harv6\_mp\_Mguptake"   
[111] "harv1\_bp1\_Nuptake" "harv1\_bp1\_Puptake"   
[113] "harv1\_bp1\_Kuptake" "harv1\_bp1\_Cauptake"   
[115] "harv1\_bp1\_Mguptake" "harv1\_bp2\_Nuptake"   
[117] "harv1\_bp2\_Puptake" "harv1\_bp2\_Kuptake"   
[119] "harv1\_bp2\_Cauptake" "harv1\_bp2\_Mguptake"   
[121] "annual\_N\_uptake" "annual\_P\_uptake"   
[123] "annual\_K\_uptake" "annual\_Ca\_uptake"   
[125] "annual\_Mg\_uptake" "annual\_N\_balance"   
[127] "annual\_P\_balance" "annual\_K\_balance"   
[129] "annual\_Ca\_balance" "annual\_Mg\_balance"   
[131] "ref\_yield\_dm" "Ymain\_rel"   
[133] "stn" "anavg\_temp"   
[135] "ansum\_prec" "ansum\_sun"   
[137] "juvdev\_temp" "juvdev\_prec"   
[139] "juvdev\_sun" "kPS\_log"   
[141] "PS\_log" "soil\_0\_20\_P\_AAE10\_log"   
[143] "soil\_0\_20\_P\_CO2\_log"

nrow(allP[allP$year==2017,])

[1] 96

nrow(allP[allP$year==2018,])

[1] 144

RES$nlme.coef.mrg <- nlme.coef.mrg  
RES$nlme.coef.avg <- nlme.coef.avg  
RES$data <- d  
  
  
saveRDS(RES, file = "./data/RES.rds")