Project 2

Lukas Leindals

15/1/2020

Table of Contents

# Data

## location yield DGT olsenP  
## 1 001 70.90984 32.300000 4.0  
## 2 001 73.89208 32.300000 4.0  
## 3 001 77.61317 32.300000 4.0  
## 4 001 75.38128 32.300000 4.0  
## 5 002 62.94290 32.800000 4.0  
## 6 002 63.04568 32.800000 4.0  
## 7 002 56.47671 32.800000 4.0  
## 8 002 58.45712 32.800000 4.0  
## 9 003 89.90965 159.186314 8.3  
## 10 003 84.31793 159.186314 8.3  
## 11 003 91.05487 159.186314 8.3  
## 12 003 86.86807 159.186314 8.3  
## 13 004 72.10570 22.183105 2.4  
## 14 004 72.06431 22.183105 2.4  
## 15 004 72.80023 22.183105 2.4  
## 16 004 73.01493 22.183105 2.4  
## 17 006 83.99737 48.973421 2.2  
## 18 006 85.91062 48.973421 2.2  
## 19 006 84.23344 48.973421 2.2  
## 20 006 74.80201 48.973421 2.2  
## 21 007 64.80943 115.474782 4.8  
## 22 007 66.22915 115.474782 4.8  
## 23 007 67.04873 115.474782 4.8  
## 24 007 68.97321 115.474782 4.8  
## 25 008 58.68690 102.440168 7.4  
## 26 008 60.93848 102.440168 7.4  
## 27 008 65.03955 102.440168 7.4  
## 28 008 66.97715 102.440168 7.4  
## 29 010 52.22216 42.132332 2.5  
## 30 010 56.47037 42.132332 2.5  
## 31 010 55.17329 42.132332 2.5  
## 32 010 57.63942 42.132332 2.5  
## 33 011 25.03687 3.072097 2.0  
## 34 011 NA 3.072097 2.0  
## 35 011 23.00496 3.072097 2.0  
## 36 011 NA 3.072097 2.0

## Summaries

cat("Structure")

## Structure

str(data\_fos)

## 'data.frame': 36 obs. of 4 variables:  
## $ location: Factor w/ 9 levels "001","002","003",..: 1 1 1 1 2 2 2 2 3 3 ...  
## $ yield : num 70.9 73.9 77.6 75.4 62.9 ...  
## $ DGT : num 32.3 32.3 32.3 32.3 32.8 ...  
## $ olsenP : num 4 4 4 4 4 4 4 4 8.3 8.3 ...

cat("Summary")

## Summary

summary(data\_fos)

## location yield DGT olsenP   
## 001 : 4 Min. :23.00 Min. : 3.072 Min. :2.000   
## 002 : 4 1st Qu.:59.25 1st Qu.: 32.300 1st Qu.:2.400   
## 003 : 4 Median :68.01 Median : 42.132 Median :4.000   
## 004 : 4 Mean :67.59 Mean : 62.062 Mean :4.178   
## 006 : 4 3rd Qu.:75.24 3rd Qu.:102.440 3rd Qu.:4.800   
## 007 : 4 Max. :91.05 Max. :159.186 Max. :8.300   
## (Other):12 NA's :2

cat("Data")

## Data

data\_fos

## location yield DGT olsenP  
## 1 001 70.90984 32.300000 4.0  
## 2 001 73.89208 32.300000 4.0  
## 3 001 77.61317 32.300000 4.0  
## 4 001 75.38128 32.300000 4.0  
## 5 002 62.94290 32.800000 4.0  
## 6 002 63.04568 32.800000 4.0  
## 7 002 56.47671 32.800000 4.0  
## 8 002 58.45712 32.800000 4.0  
## 9 003 89.90965 159.186314 8.3  
## 10 003 84.31793 159.186314 8.3  
## 11 003 91.05487 159.186314 8.3  
## 12 003 86.86807 159.186314 8.3  
## 13 004 72.10570 22.183105 2.4  
## 14 004 72.06431 22.183105 2.4  
## 15 004 72.80023 22.183105 2.4  
## 16 004 73.01493 22.183105 2.4  
## 17 006 83.99737 48.973421 2.2  
## 18 006 85.91062 48.973421 2.2  
## 19 006 84.23344 48.973421 2.2  
## 20 006 74.80201 48.973421 2.2  
## 21 007 64.80943 115.474782 4.8  
## 22 007 66.22915 115.474782 4.8  
## 23 007 67.04873 115.474782 4.8  
## 24 007 68.97321 115.474782 4.8  
## 25 008 58.68690 102.440168 7.4  
## 26 008 60.93848 102.440168 7.4  
## 27 008 65.03955 102.440168 7.4  
## 28 008 66.97715 102.440168 7.4  
## 29 010 52.22216 42.132332 2.5  
## 30 010 56.47037 42.132332 2.5  
## 31 010 55.17329 42.132332 2.5  
## 32 010 57.63942 42.132332 2.5  
## 33 011 25.03687 3.072097 2.0  
## 34 011 NA 3.072097 2.0  
## 35 011 23.00496 3.072097 2.0  
## 36 011 NA 3.072097 2.0

# nas <- which(is.na(data\_fos$yield))   
# data\_fos[unique(c(nas, nas+1, nas-1)), ]  
  
#xtable::xtable(summary(data\_fos, maxsum = 9)) #converts to latex

## Handle NA’s

Uses KNN imputation and looks at the k nearest neighbors, the missing value is then replaced with the average of the values for the nearest neighbors

data\_fos <- DMwR::knnImputation(data\_fos,k=2, meth = "Median")

## Registered S3 method overwritten by 'xts':  
## method from  
## as.zoo.xts zoo

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

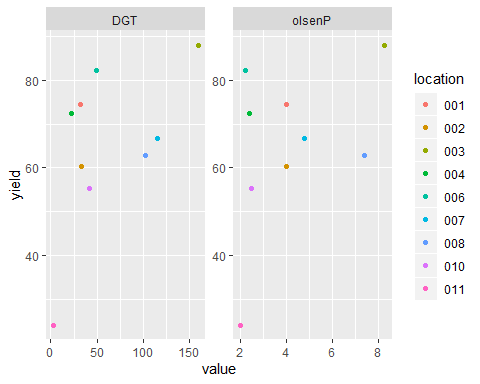
## grouped data set

groups the data, so that there is only one yield for each combination of location, DGT and olsenP, which is the mean. THis is done as these three factors are strongly dependent of each other.

data\_grouped <- data\_fos %>% dplyr::group\_by(location, DGT, olsenP) %>% dplyr::summarise\_each(mean)  
data\_grouped

## # A tibble: 9 x 4  
## # Groups: location, DGT [9]  
## location DGT olsenP yield  
## <fct> <dbl> <dbl> <dbl>  
## 1 001 32.3 4 74.4  
## 2 002 32.8 4 60.2  
## 3 003 159. 8.3 88.0  
## 4 004 22.2 2.4 72.5  
## 5 006 49.0 2.2 82.2  
## 6 007 115. 4.8 66.8  
## 7 008 102. 7.4 62.9  
## 8 010 42.1 2.5 55.4  
## 9 011 3.07 2 24.0

ggplot(tidyr::gather(data\_grouped, "key", "value", -c(location, yield)), aes(value, yield, col = location)) +  
 geom\_point() +  
 facet\_wrap("key", scales = "free")

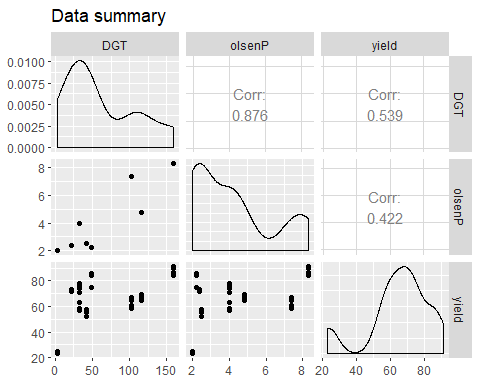


## Plots

Comparison plot

GGally::ggpairs(data\_fos, columns = c("DGT", "olsenP", "yield"), title = "Data summary")

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

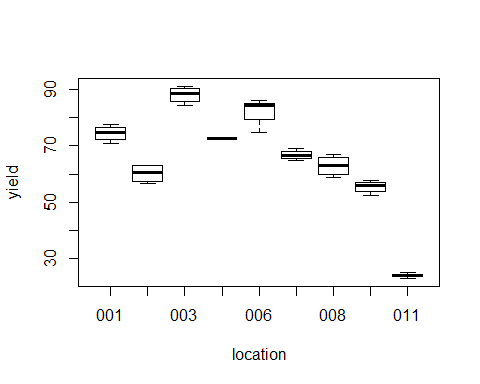


ggsave("data\_plot.jpg")

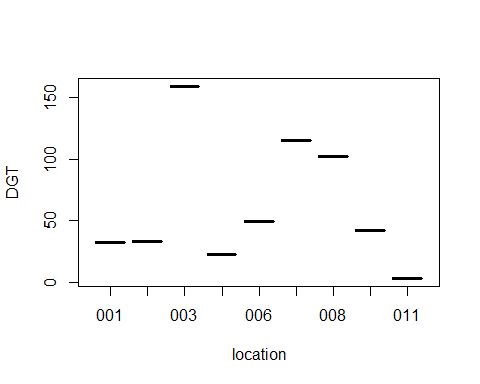
## Saving 5 x 4 in image

Boxplots

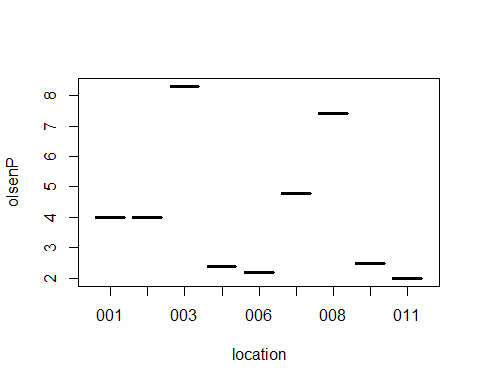
#par(mfrow = c(3,1))  
boxplot(yield ~location, data = data\_fos)



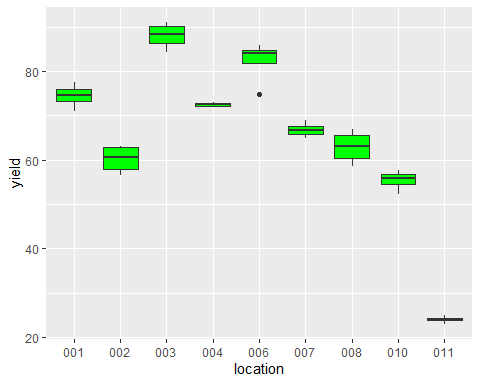
boxplot(DGT ~ location, data = data\_fos)



boxplot(olsenP ~ location, data = data\_fos)



data\_fos %>%   
 tidyr::gather("key", "value", -location) %>%  
 dplyr::filter(key == "yield") %>%   
 dplyr::rename(yield = value) %>%   
 ggplot(aes(x = location, y = yield)) +   
 geom\_boxplot(fill = 'green')

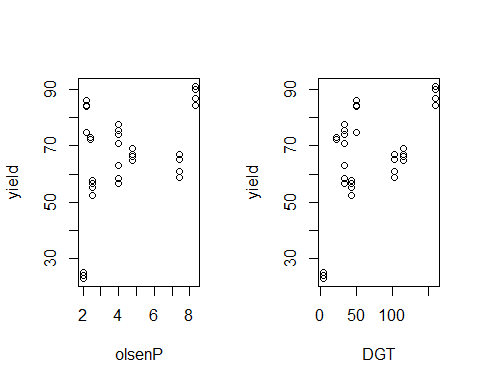


ggsave("boxplot\_yield\_location.jpg")

## Saving 5 x 4 in image

Scatter plots

par(mfrow = c(1,2))  
plot(data = data\_fos , yield ~ olsenP)  
plot(data = data\_fos , yield ~ DGT)



# OlsenP vs. DGT

## ANOVA

lm1 <- lm(data = data\_fos, formula = yield ~ DGT)  
anova(lm1)

## Analysis of Variance Table  
##   
## Response: yield  
## Df Sum Sq Mean Sq F value Pr(>F)   
## DGT 1 3298.0 3298.0 13.958 0.0006852 \*\*\*  
## Residuals 34 8033.5 236.3   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(lm1)

##   
## Call:  
## lm(formula = yield ~ DGT, data = data\_fos)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -30.5574 -8.7287 -0.4578 14.6196 23.3168   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 52.95788 4.15290 12.752 1.64e-14 \*\*\*  
## DGT 0.19676 0.05267 3.736 0.000685 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 15.37 on 34 degrees of freedom  
## Multiple R-squared: 0.291, Adjusted R-squared: 0.2702   
## F-statistic: 13.96 on 1 and 34 DF, p-value: 0.0006852

lm2 <- lm(data = data\_fos, formula = yield ~ olsenP)  
anova(lm2)

## Analysis of Variance Table  
##   
## Response: yield  
## Df Sum Sq Mean Sq F value Pr(>F)   
## olsenP 1 2021.4 2021.38 7.382 0.01029 \*  
## Residuals 34 9310.1 273.83   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(lm2)

##   
## Call:  
## lm(formula = yield ~ olsenP, data = data\_fos)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -34.651 -7.389 -0.677 12.005 27.565   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 50.756 5.979 8.489 6.49e-10 \*\*\*  
## olsenP 3.450 1.270 2.717 0.0103 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 16.55 on 34 degrees of freedom  
## Multiple R-squared: 0.1784, Adjusted R-squared: 0.1542   
## F-statistic: 7.382 on 1 and 34 DF, p-value: 0.01029

lm3 <- lm(data = data\_fos, formula = yield ~ location)  
anova(lm3)

## Analysis of Variance Table  
##   
## Response: yield  
## Df Sum Sq Mean Sq F value Pr(>F)   
## location 8 11100.6 1387.58 162.29 < 2.2e-16 \*\*\*  
## Residuals 27 230.8 8.55   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(lm3)

##   
## Call:  
## lm(formula = yield ~ location, data = data\_fos)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -7.4339 -1.3205 0.1418 2.0304 4.0666   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 74.449 1.462 50.923 < 2e-16 \*\*\*  
## location002 -14.218 2.068 -6.877 2.18e-07 \*\*\*  
## location003 13.589 2.068 6.572 4.76e-07 \*\*\*  
## location004 -1.953 2.068 -0.944 0.353300   
## location006 7.787 2.068 3.766 0.000819 \*\*\*  
## location007 -7.684 2.068 -3.716 0.000933 \*\*\*  
## location008 -11.539 2.068 -5.581 6.43e-06 \*\*\*  
## location010 -19.073 2.068 -9.225 7.79e-10 \*\*\*  
## location011 -50.428 2.068 -24.390 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.924 on 27 degrees of freedom  
## Multiple R-squared: 0.9796, Adjusted R-squared: 0.9736   
## F-statistic: 162.3 on 8 and 27 DF, p-value: < 2.2e-16

phos.model.DGT <- nls(yield ~ alfa \* DGT/(beta + DGT) , data = data\_fos,  
start = list(alfa = 90 , beta = 1))  
  
phos.model.olsenP <- nls(yield ~ alfa \* olsenP/(beta + olsenP) , data = data\_fos,  
start = list(alfa = 90 , beta = 1))  
  
  
  
summary(phos.model.DGT)

##   
## Formula: yield ~ alfa \* DGT/(beta + DGT)  
##   
## Parameters:  
## Estimate Std. Error t value Pr(>|t|)   
## alfa 78.114 3.187 24.506 < 2e-16 \*\*\*  
## beta 5.494 1.578 3.481 0.00139 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 10.58 on 34 degrees of freedom  
##   
## Number of iterations to convergence: 7   
## Achieved convergence tolerance: 3.552e-06

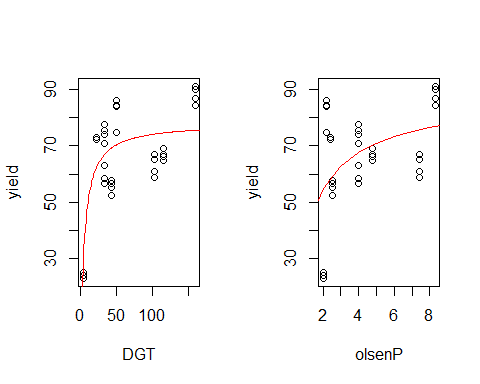
summary(phos.model.olsenP)

##   
## Formula: yield ~ alfa \* olsenP/(beta + olsenP)  
##   
## Parameters:  
## Estimate Std. Error t value Pr(>|t|)   
## alfa 87.8750 10.5829 8.303 1.08e-09 \*\*\*  
## beta 1.2073 0.5749 2.100 0.0432 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 16.33 on 34 degrees of freedom  
##   
## Number of iterations to convergence: 3   
## Achieved convergence tolerance: 5.377e-06

# lm\_august <- nls(yield ~ a \* olsenP/(b+olsenP), data = data\_fos, start = list(b = max(data\_fos$yield)/2, a = max(data\_fos$yield)))  
# anova(lm\_august)  
# summary(lm\_august)  
# coef(lm\_august)  
# lm\_august  
# phos.model.olsenP

### Plots of model coefficients

par(mfrow=(c(1,2)))  
alfa1 <- coef(phos.model.DGT)[1]  
beta1 <- coef(phos.model.DGT)[2]  
alfa2 <- coef(phos.model.olsenP)[1]  
beta2 <- coef(phos.model.olsenP)[2]  
plot(data = data\_fos , yield ~ DGT)  
lines(x<-c(1:200),(alfa1 \* x )/(beta1 + x),col='red')  
plot(data = data\_fos , yield ~ olsenP)  
lines(x<-c(1:200),(alfa2 \* x )/(beta2 + x),col='red')



## Predicting yield

Test errorrate med leave one out cross validation

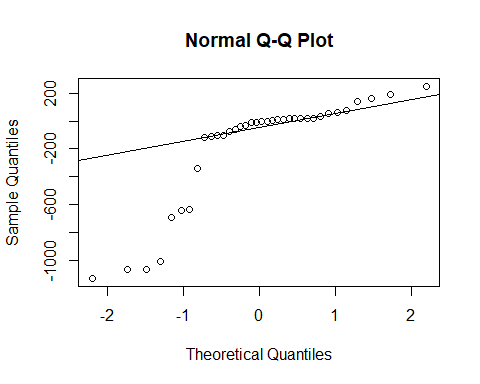
data\_CV <- data\_fos  
  
N <- nrow(data\_CV)  
s <- sample(N)  
K\_fold <- N  
  
  
accuracy <- c()  
random\_index <- split(sample(c(1:N),N), c(1:K\_fold))  
loss <- c()  
for (i in c(1:K\_fold)){  
 f <- c(1:K\_fold)[-i]  
 train\_index <- Reduce(c,random\_index[f])  
 test\_index <- Reduce(c,random\_index[i])  
  
 train\_set <- data\_CV[train\_index,]  
 test\_set <- data\_CV[test\_index,]  
 model\_cv\_DGT <- phos.model.DGT\_perm <- nls(yield ~ alfa \* DGT/(beta + DGT) , data = train\_set,  
 start = list(alfa = 90 , beta = 1))  
   
 model\_cv\_olsenP <- phos.model.DGT\_perm <- nls(yield ~ alfa \* olsenP/(beta + olsenP) , data = train\_set,  
 start = list(alfa = 90 , beta = 1))  
   
 pred\_DGT <- predict(model\_cv\_DGT, test\_set)  
 pred\_olsenP <- predict(model\_cv\_olsenP, test\_set)  
   
 error\_DGT <- sum((test\_set$yield - pred\_DGT)^2)  
 error\_olsenP <- sum((test\_set$yield - pred\_olsenP)^2)  
 loss <- rbind(loss,c(error\_DGT, error\_olsenP))  
   
}  
Z <- loss[,1] - loss[,2]  
t.test(Z)

##   
## One Sample t-test  
##   
## data: Z  
## t = -2.6943, df = 35, p-value = 0.01076  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## -301.90065 -42.44333  
## sample estimates:  
## mean of x   
## -172.172

t.test(loss[,1], loss[,2], paired =T)

##   
## Paired t-test  
##   
## data: loss[, 1] and loss[, 2]  
## t = -2.6943, df = 35, p-value = 0.01076  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -301.90065 -42.44333  
## sample estimates:  
## mean of the differences   
## -172.172

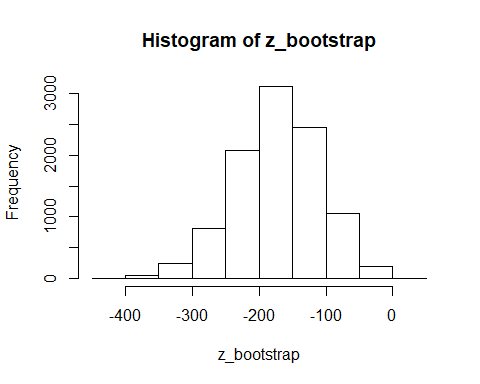
qqnorm(Z)  
qqline(Z)



### Bootstrap - loss

Difference in performance with Michaelis Menten fitted to DGT vs olsenP Non parametric bootstrapping. Sampler fra forskel i loss af de to modeller. Udregner mean af disse forskelle og bestemmer konfidens intervaller.

z\_bootstrap <- apply( replicate(10000, sample(Z, replace = T)), 2, mean)  
hist(z\_bootstrap)



alfa <- 0.05  
quantile(z\_bootstrap, c(alfa/2,1-alfa/2))

## 2.5% 97.5%   
## -304.2633 -55.3594

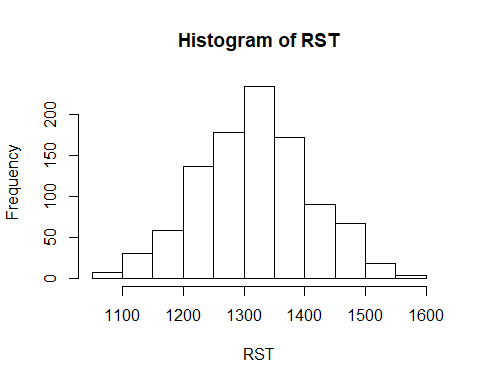
### Rank Sum Test

Difference in performance with Michaelis Menten fitted to DGT vs olsenP Concatenates loss af DGT og olsenP. Bestemmers størrelseordenen af denne vektor og finder rank sum for DGT. Derefter laves permutationer af vektor med 1-72 og rank sum af de 36 første bestemmes og sammenlignes med rank sum for DGT

Z\_RST <- order(c(loss[,1], loss[,2]))  
DGT\_RS <- sum(Z\_RST[1:36])  
RST <- c()  
for (i in 1:1000){  
 RST\_sample <- sample(72)  
 RST <- c(RST,sum(RST\_sample[1:36]))  
}  
p\_val <- 2\*mean(DGT\_RS>RST)  
p\_val

## [1] 1.076

hist(RST)



Z\_RST

## [1] 41 69 67 55 26 64 6 25 59 72 28 42 61 30 14 2 44 5 24 68 32 70 12 19 56  
## [26] 71 22 23 20 58 35 48 4 21 27 8 39 16 7 29 3 47 43 40 33 11 31 13 9 57  
## [51] 63 36 15 53 52 17 65 37 49 18 10 1 34 46 38 45 51 54 62 66 50 60

# Influence of phosphorous

## Permutation test

n <- 500  
results\_DGT <- c()  
results\_olsenP <- c()  
for (i in 1:n){  
 permutation <- sample(nrow(data\_fos))  
   
 phos.model.DGT\_perm <- nls(yield[permutation] ~ alfa \* DGT/(beta + DGT) , data = data\_fos,  
 start = list(alfa = 90 , beta = 1))  
  
 phos.model.olsenP\_perm <- nls(yield[permutation] ~ alfa \* olsenP/(beta + olsenP) , data = data\_fos,  
 start = list(alfa = 90 , beta = 1))  
   
 results\_DGT <- rbind(results\_DGT, coef(phos.model.DGT\_perm))  
 results\_olsenP <- rbind(results\_olsenP, coef(phos.model.olsenP\_perm))  
}  
sum(beta1 < results\_DGT[,2])

## [1] 0

mean(results\_DGT[,2])

## [1] 0.06547645

alfa1

## alfa   
## 78.11392

beta1

## beta   
## 5.493647