Analysis

Adama NDOUR

2024-07-16

## Load package

library(tidyverse)  
library(readxl)  
library(paletteer)  
library(meta) # meta-analysis  
library(nnet) # Multinomial logit model  
library(GGally) # plot model coefficients  
library(recipes)  
library(recipeselectors)  
library(embed) # encoding  
library(report) # report statistical results  
library(stargazer) # formatting statistical results  
library(kableExtra) # format table  
library(DHARMa) # plot residuals of a linear model

## Load the data

df <- read\_excel("uav\_review\_data.xlsx")

## Overview of the data

#str(df)

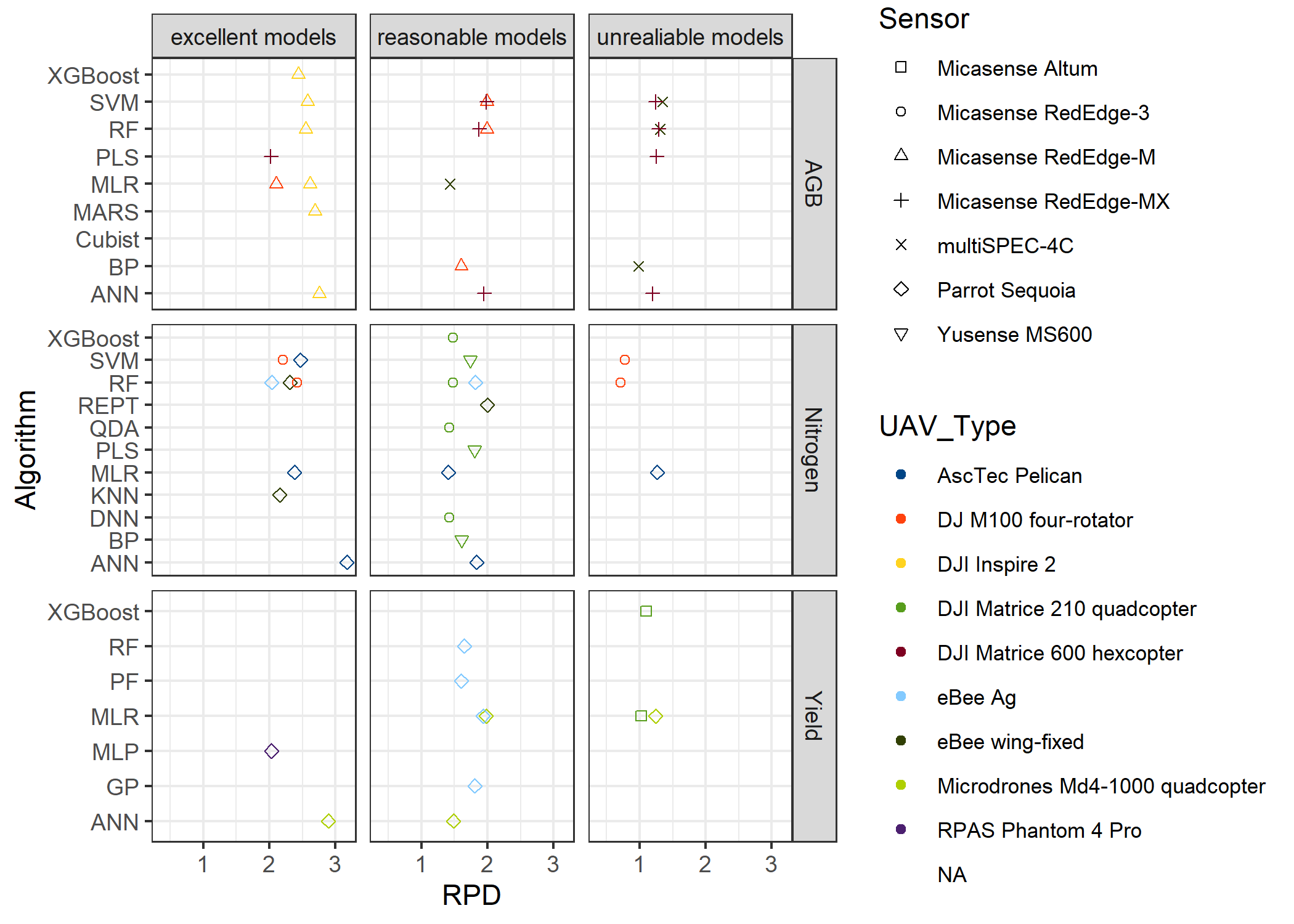
## Data manipulation: create a model class variable

df <- df %>% mutate(  
 Model\_Class = case\_when(  
 RPD < 1.4 ~ "unrealiable models",  
 RPD >= 1.4 & RPD < 2 ~ "reasonable models",  
 RPD >=2 ~ "excellent models"  
 )  
)  
  
# Remove special characters  
df$Sensor <- str\_replace\_all(df$Sensor, "\r", " ")  
df$Sensor <- str\_replace\_all(df$Sensor, "\n", " ")  
df$Sensor <- str\_replace\_all(df$Sensor, "\\s+", " ")

## Exploratory Data Analysis (EDA)

### Which UAV platform maximize the performance of ML models

df\_trait <- df %>% filter(Problem=="trait estimation")  
#df\_trait %>% group\_by(Crop,Algorithm)  
n\_shape\_var <- length(unique(df\_trait$Sensor))  
p<-df\_trait %>%   
 group\_by(DOI, Trait, UAV\_Type, Sensor, Algorithm,Model\_Class) %>%   
 summarise(RPD=mean(RPD)) %>%   
 ggplot(aes(y=Algorithm, x=RPD, color=UAV\_Type)) +  
 geom\_point(aes(shape=Sensor))+  
 scale\_shape\_manual(values = 0:n\_shape\_var) +  
 scale\_color\_paletteer\_d("ggthemes::calc")+  
 facet\_grid(Trait~ Model\_Class, scales = "free\_y")+  
 theme\_bw()+  
 theme(  
 legend.text = element\_text(size = 8.5)  
 )  
ggsave("output/figure1.png",plot = p,dpi = 300)  
p

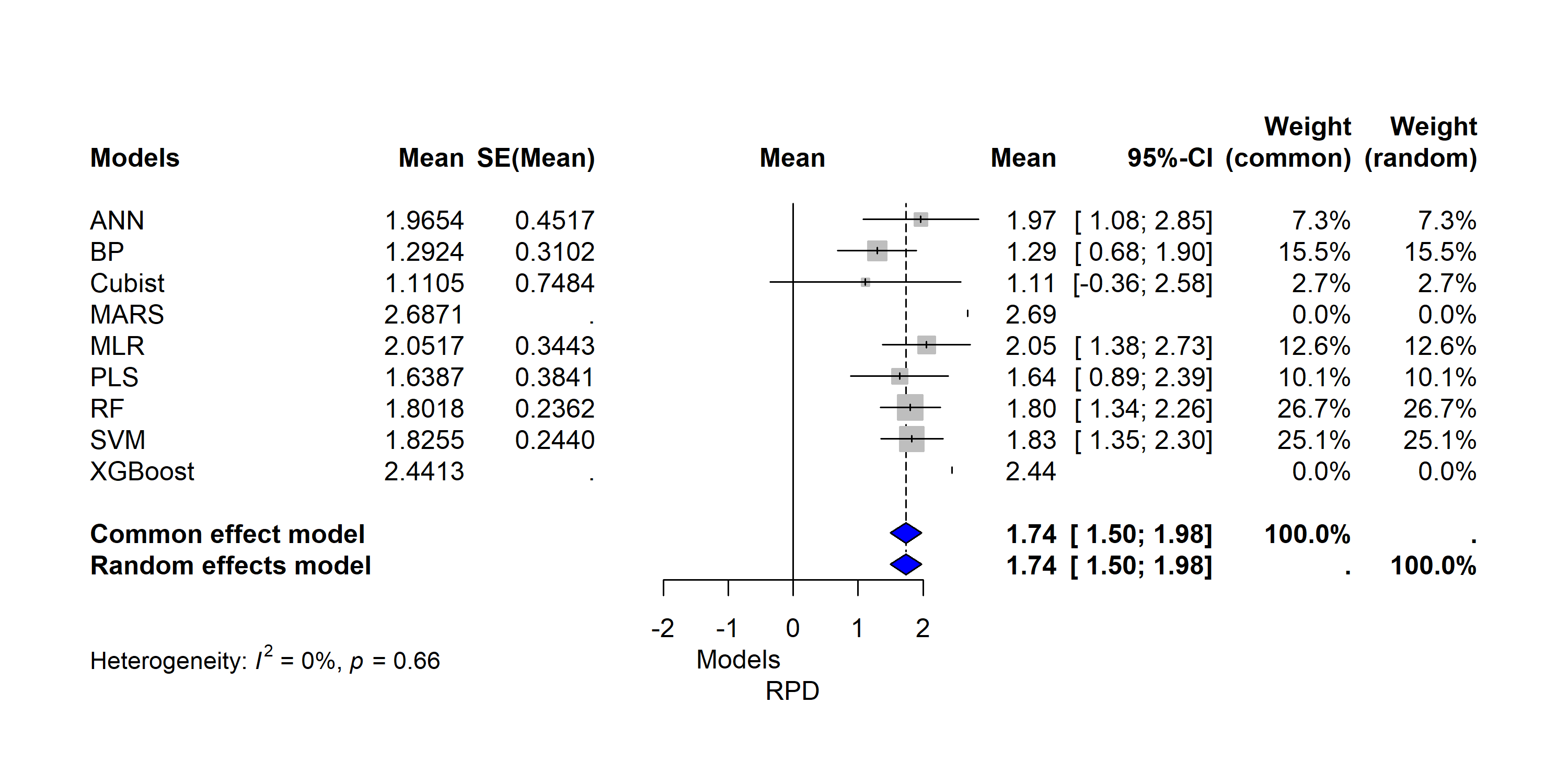


### Forest plot for the biomass

# Install and load necessary packages  
# Load necessary libraries  
  
# Example RPD data  
biomass\_data <- df\_trait %>% filter(Trait=="AGB")  
  
# Calculate summary statistics  
biomass\_rpd\_summary <- biomass\_data %>%  
 group\_by(Algorithm) %>%  
 summarize(  
 mean\_RPD = mean(RPD),  
 sd\_RPD = sd(RPD),  
 n = n(),  
 SEM\_RPD = sd\_RPD / sqrt(n),  
 CI\_Lower = mean\_RPD - 1.96 \* SEM\_RPD,  
 CI\_Upper = mean\_RPD + 1.96 \* SEM\_RPD  
 )  
  
# Print the summary  
print(biomass\_rpd\_summary)

## # A tibble: 9 × 7  
## Algorithm mean\_RPD sd\_RPD n SEM\_RPD CI\_Lower CI\_Upper  
## <chr> <dbl> <dbl> <int> <dbl> <dbl> <dbl>  
## 1 ANN 1.97 0.782 3 0.452 1.08 2.85  
## 2 BP 1.29 0.439 2 0.310 0.684 1.90  
## 3 Cubist 1.11 1.06 2 0.748 -0.356 2.58  
## 4 MARS 2.69 NA 1 NA NA NA   
## 5 MLR 2.05 0.596 3 0.344 1.38 2.73  
## 6 PLS 1.64 0.543 2 0.384 0.886 2.39  
## 7 RF 1.80 0.528 5 0.236 1.34 2.26  
## 8 SVM 1.83 0.546 5 0.244 1.35 2.30  
## 9 XGBoost 2.44 NA 1 NA NA NA

# Combine data for all models  
biomass\_meta\_combined <- metagen(  
 TE = biomass\_rpd\_summary$mean\_RPD,  
 lower = biomass\_rpd\_summary$CI\_Lower,  
 upper = biomass\_rpd\_summary$CI\_Upper,  
 studlab = biomass\_rpd\_summary$Algorithm,  
 sm = "Mean"  
)  
  
# Forest plot for all models  
# png(file = "output/forestplot\_biomass.png", width = 10, height = 5, res = 300, units = "in")  
forest(biomass\_meta\_combined,  
 main = "Forest Plot of RPD for All Models of Biomass Estimation",  
 xlab = "RPD",  
 label.left = "Models",  
 studlab = biomass\_rpd\_summary$Algorithm,  
 print.tau2 = FALSE,  
 col.diamond = "blue",  
 col.predict = "red",  
 leftlabs = c("Models", "Mean", "SE(Mean)"))

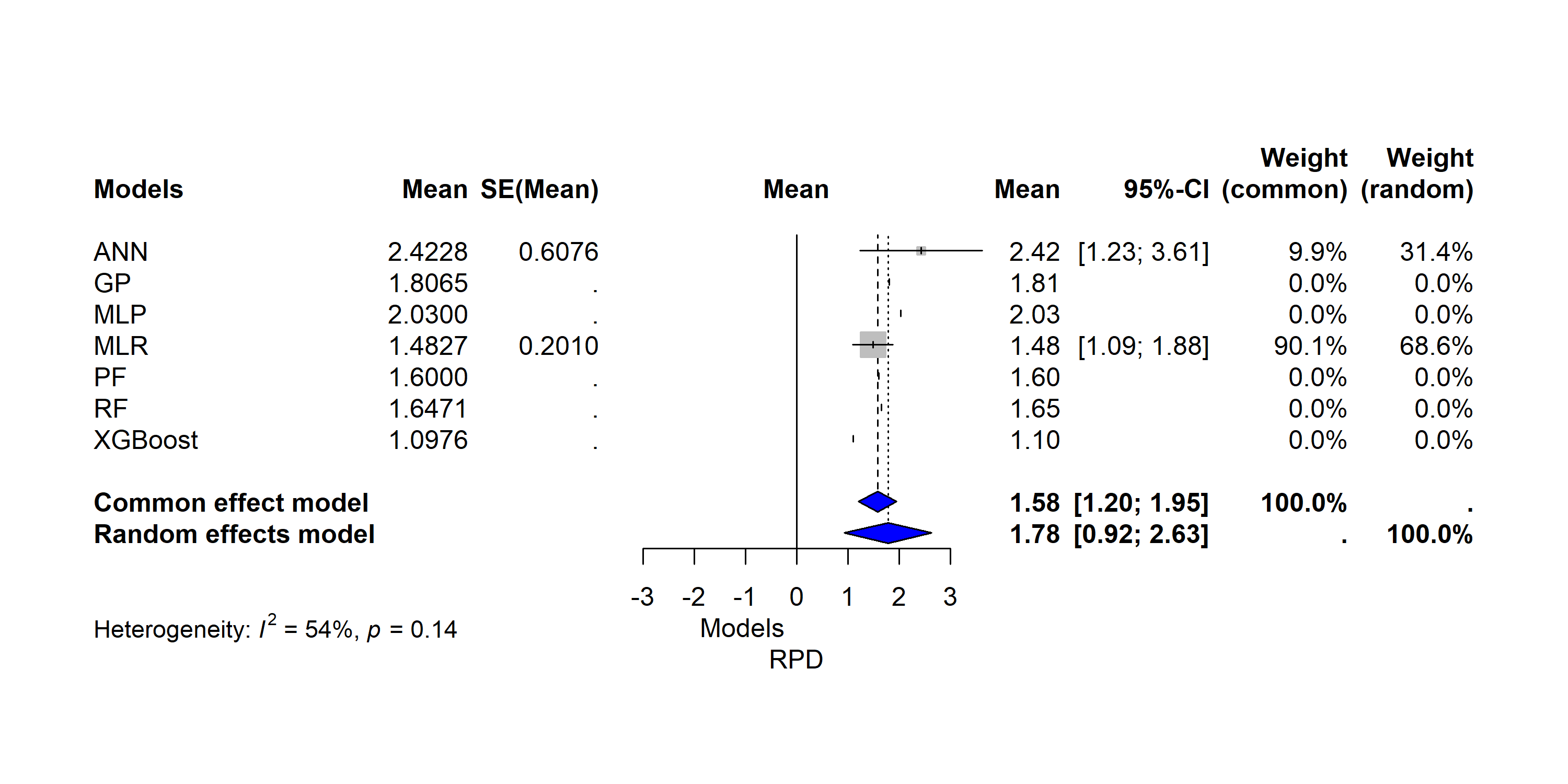


### Forest plot for the yield

# Install and load necessary packages  
# Load necessary libraries  
  
# Example RPD data  
yield\_data <- df\_trait %>% filter(Trait=="Yield")  
  
# Calculate summary statistics  
yield\_rpd\_summary <- yield\_data %>%  
 group\_by(Algorithm) %>%  
 summarize(  
 mean\_RPD = mean(RPD),  
 sd\_RPD = sd(RPD),  
 n = n(),  
 SEM\_RPD = sd\_RPD / sqrt(n),  
 CI\_Lower = mean\_RPD - 1.96 \* SEM\_RPD,  
 CI\_Upper = mean\_RPD + 1.96 \* SEM\_RPD  
 )  
  
# Print the summary  
print(yield\_rpd\_summary)

## # A tibble: 7 × 7  
## Algorithm mean\_RPD sd\_RPD n SEM\_RPD CI\_Lower CI\_Upper  
## <chr> <dbl> <dbl> <int> <dbl> <dbl> <dbl>  
## 1 ANN 2.42 1.05 3 0.608 1.23 3.61  
## 2 GP 1.81 NA 1 NA NA NA   
## 3 MLP 2.03 NA 1 NA NA NA   
## 4 MLR 1.48 0.449 5 0.201 1.09 1.88  
## 5 PF 1.6 NA 1 NA NA NA   
## 6 RF 1.65 NA 1 NA NA NA   
## 7 XGBoost 1.10 NA 1 NA NA NA

# Combine data for all models  
yield\_meta\_combined <- metagen(  
 TE = yield\_rpd\_summary$mean\_RPD,  
 lower = yield\_rpd\_summary$CI\_Lower,  
 upper = yield\_rpd\_summary$CI\_Upper,  
 studlab = yield\_rpd\_summary$Algorithm,  
 sm = "Mean"  
)  
  
# Forest plot for all models  
# png(file = "output/forestplot\_yield.png", width = 10, height = 5, res = 300, units = "in")  
forest(yield\_meta\_combined,  
 main = "Forest Plot of RPD for All Models for Yield Estimation",  
 xlab = "RPD",  
 label.left = "Models",  
 studlab = yield\_rpd\_summary$Algorithm,  
 print.tau2 = FALSE,  
 col.diamond = "blue",  
 col.predict = "red",  
 leftlabs = c("Models", "Mean", "SE(Mean)"))

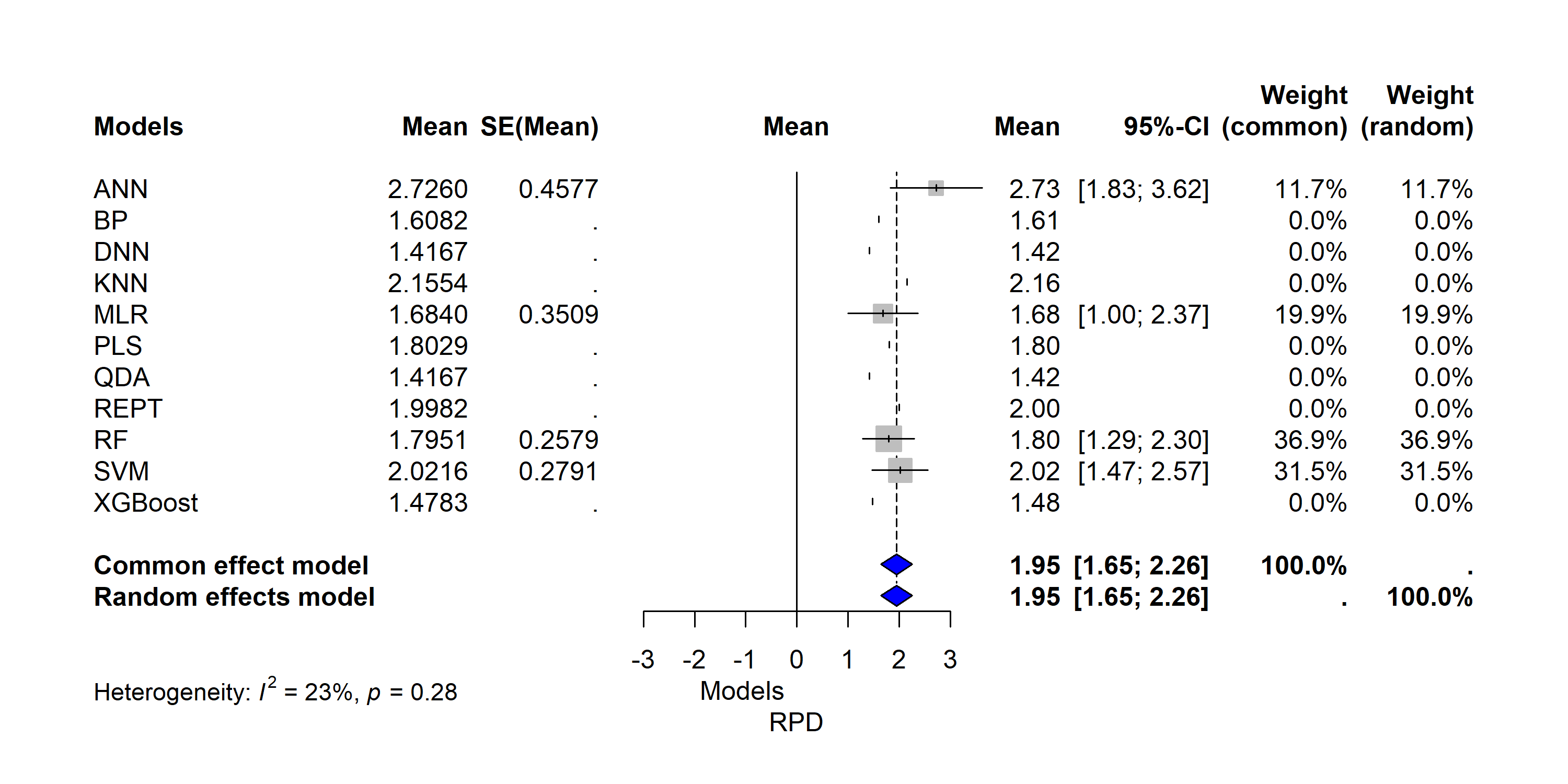


### Forest plot for the nitrogen

# Install and load necessary packages  
# Load necessary libraries  
library(meta)  
  
# Example RPD data  
nitrogen\_data <- df\_trait %>% filter(Trait=="Nitrogen")  
  
# Calculate summary statistics  
nitrogen\_rpd\_summary <- nitrogen\_data %>%  
 group\_by(Algorithm) %>%  
 summarize(  
 mean\_RPD = mean(RPD),  
 sd\_RPD = sd(RPD),  
 n = n(),  
 SEM\_RPD = sd\_RPD / sqrt(n),  
 CI\_Lower = mean\_RPD - 1.96 \* SEM\_RPD,  
 CI\_Upper = mean\_RPD + 1.96 \* SEM\_RPD  
 )  
  
# Print the summary  
print(nitrogen\_rpd\_summary)

## # A tibble: 11 × 7  
## Algorithm mean\_RPD sd\_RPD n SEM\_RPD CI\_Lower CI\_Upper  
## <chr> <dbl> <dbl> <int> <dbl> <dbl> <dbl>  
## 1 ANN 2.73 0.793 3 0.458 1.83 3.62  
## 2 BP 1.61 NA 1 NA NA NA   
## 3 DNN 1.42 NA 1 NA NA NA   
## 4 KNN 2.16 NA 1 NA NA NA   
## 5 MLR 1.68 0.608 3 0.351 0.996 2.37  
## 6 PLS 1.80 NA 1 NA NA NA   
## 7 QDA 1.42 NA 1 NA NA NA   
## 8 REPT 2.00 NA 1 NA NA NA   
## 9 RF 1.80 0.632 6 0.258 1.29 2.30  
## 10 SVM 2.02 0.684 6 0.279 1.47 2.57  
## 11 XGBoost 1.48 NA 1 NA NA NA

# Combine data for all models  
nitrogen\_meta\_combined <- metagen(  
 TE = nitrogen\_rpd\_summary$mean\_RPD,  
 lower = nitrogen\_rpd\_summary$CI\_Lower,  
 upper = nitrogen\_rpd\_summary$CI\_Upper,  
 studlab = nitrogen\_rpd\_summary$Algorithm,  
 sm = "Mean"  
)  
  
# Forest plot for all models  
#png(file = "output/forestplot\_nitrogen.png", width = 10, height = 5, res = 300, units = "in")  
forest(nitrogen\_meta\_combined,  
 main = "Forest Plot of RPD for All Models of nitrogen Estimation",  
 xlab = "RPD",  
 label.left = "Models",  
 studlab = nitrogen\_rpd\_summary$Algorithm,  
 print.tau2 = FALSE,  
 comb.random = FALSE,  
 col.diamond = "blue",  
 col.predict = "red",  
 leftlabs = c("Models", "Mean", "SE(Mean)"))



## Multivariate Linear Regression: Key drivers of ML model performance

# Recode RPD variable to convert to factor predictors  
df\_trait\_model <- df\_trait %>%   
 mutate(  
 RPD\_rec = recode\_factor(Model\_Class,   
 "unrealiable models" = "Bad",  
 "reasonable models" = "Reliable",  
 "excellent models" = "Excellent"),  
 Crop = as\_factor(Crop),  
 Stage = as\_factor(Stage),  
 Trait = as\_factor(Trait),  
 UAV\_Type = as\_factor(UAV\_Type),  
 Sensor = as\_factor(Sensor),  
 Band = as\_factor(Band),  
 Algorithm = as\_factor(Algorithm)  
 )  
df\_trait\_model <- df\_trait\_model %>%   
 select(RPD, Crop, Stage, Trait, UAV\_Type, Sensor, Band, Altitude\_m, Algorithm)

# Drop Na  
df\_trait\_model <- df\_trait\_model %>% drop\_na(UAV\_Type)

### Biomass

# Example data  
biomass\_model\_data <- df\_trait\_model %>% filter(Trait=="AGB")  
  
# Remove trait  
biomass\_model\_data <- biomass\_model\_data %>% select(-Trait)  
  
# Standardize predictor variables  
recipe <- recipe(RPD ~ ., data=biomass\_model\_data) %>%  
 # convert string to factor  
 #step\_string2factor(all\_nominal()) %>%  
 # remove no variance predictors   
 #recipes::step\_nzv(all\_nominal()) %>%  
 # factor to dummy variables  
 #step\_dummy(all\_nominal(), one\_hot=T) %>%  
 step\_lencode\_mixed(all\_nominal\_predictors() , outcome=vars(RPD)) %>%  
 # remove non-variance variables  
 step\_nzv(where(is.numeric)) %>%  
 #step\_dummy(all\_nominal\_predictors(), one\_hot=F) %>% # Convert categorical variables to dummy variables  
 prep()

## boundary (singular) fit: see help('isSingular')

# juice recipe  
biomass\_model\_data\_final <- juice(recipe)  
  
# Fit the multinomial logistic regression model  
biomass\_model <- lm(RPD ~. , data = biomass\_model\_data)

### Yield

# Example data  
yield\_model\_data <- df\_trait\_model %>% filter(Trait=="Yield")  
  
# Remove trait  
yield\_model\_data <- yield\_model\_data %>% select(-Trait)  
  
# Standardize predictor variables  
recipe <- recipe(RPD ~ ., data=yield\_model\_data) %>%  
 # convert string to factor  
 #step\_string2factor(all\_nominal()) %>%  
 # remove no variance predictors   
 #recipes::step\_nzv(all\_nominal()) %>%  
 # factor to dummy variables  
 #step\_dummy(all\_nominal(), one\_hot=T) %>%  
 step\_lencode\_mixed(all\_nominal\_predictors() , outcome=vars(RPD)) %>%  
 # remove non-variance variables  
 step\_nzv(where(is.numeric)) %>%  
 #step\_dummy(all\_nominal\_predictors(), one\_hot=F) %>% # Convert categorical variables to dummy variables  
 prep()

## boundary (singular) fit: see help('isSingular')  
## boundary (singular) fit: see help('isSingular')

# juice recipe  
yield\_model\_data\_final <- juice(recipe)  
  
# Fit the multinomial logistic regression model  
yield\_model <- lm(RPD ~. , data = yield\_model\_data)

### Nitrogen

# Example data  
nitrogen\_model\_data <- df\_trait\_model %>% filter(Trait=="Nitrogen")  
  
# Remove trait  
nitrogen\_model\_data <- nitrogen\_model\_data %>% select(-Trait)  
  
# Standardize predictor variables  
recipe <- recipe(RPD ~ ., data=nitrogen\_model\_data) %>%  
 # convert string to factor  
 #step\_string2factor(all\_nominal()) %>%  
 # remove no variance predictors   
 #recipes::step\_nzv(all\_nominal()) %>%  
 # factor to dummy variables  
 #step\_dummy(all\_nominal(), one\_hot=T) %>%  
 step\_lencode\_mixed(all\_nominal\_predictors() , outcome=vars(RPD)) %>%  
 # remove non-variance variables  
 step\_nzv(where(is.numeric)) %>%  
 #step\_dummy(all\_nominal\_predictors(), one\_hot=F) %>% # Convert categorical variables to dummy variables  
 prep()

## boundary (singular) fit: see help('isSingular')

# juice recipe  
nitrogen\_model\_data\_final <- juice(recipe)  
  
# Fit the multinomial logistic regression model  
nitrogen\_model <- lm(RPD ~. , data = nitrogen\_model\_data)

## Report results

### Biomass

* Model

report\_model(biomass\_model)

## linear model (estimated using OLS) to predict RPD with Crop, Stage, UAV\_Type, Sensor, Band, Altitude\_m and Algorithm (formula: RPD ~ Crop + Stage + UAV\_Type + Sensor + Band + Altitude\_m + Algorithm)

* Performance

report\_performance(biomass\_model)

## The model explains a statistically significant and substantial proportion of  
## variance (R2 = 0.99, F(11, 10) = 154.51, p < .001, adj. R2 = 0.99)

* Parameters

report\_parameters(biomass\_model)

## - The intercept is statistically significant and positive (beta = 1.42, 95% CI [1.32, 1.51], t(10) = 31.82, p < .001; Std. beta = -0.81, 95% CI [-0.99, -0.63])  
## - The effect of Crop [oats] is statistically significant and positive (beta = 0.58, 95% CI [0.47, 0.70], t(10) = 11.32, p < .001; Std. beta = 1.03, 95% CI [0.83, 1.24])  
## - The effect of Crop [strawberry] is statistically significant and positive (beta = 1.25, 95% CI [1.15, 1.36], t(10) = 26.53, p < .001; Std. beta = 2.23, 95% CI [2.04, 2.41])  
## - The effect of Stage [pre-heading] is statistically significant and negative (beta = -0.71, 95% CI [-0.81, -0.61], t(10) = -16.11, p < .001; Std. beta = -1.26, 95% CI [-1.44, -1.09])  
## - The effect of Stage [post-heading] is statistically significant and positive (beta = 0.66, 95% CI [0.56, 0.75], t(10) = 14.87, p < .001; Std. beta = 1.17, 95% CI [0.99, 1.34])  
## - The effect of Stage [maturity] is statistically non-significant and negative (beta = -0.09, 95% CI [-0.19, 0.02], t(10) = -1.80, p = 0.103; Std. beta = -0.15, 95% CI [-0.34, 0.04])  
## - The effect of UAV Type [DJ M100 four-rotator] is statistically significant and negative (beta = -0.45, 95% CI [-0.58, -0.32], t(10) = -7.69, p < .001; Std. beta = -0.80, 95% CI [-1.03, -0.57])  
## - The effect of UAV Type [DJI Matrice 600 hexcopter] is statistically non-significant and negative (beta = -0.06, 95% CI [-0.17, 0.04], t(10) = -1.30, p = 0.222; Std. beta = -0.11, 95% CI [-0.30, 0.08])  
## - The effect of UAV Type [DJI Inspire 2] is statistically non-significant and negative (beta = -2.91e-03, 95% CI [-0.15, 0.15], t(10) = -0.04, p = 0.966; Std. beta = -5.17e-03, 95% CI [-0.27, 0.26])  
## - The effect of Sensor [Micasense RedEdge-M] is statistically non-significant and negative (beta = -0.02, 95% CI [-0.15, 0.11], t(10) = -0.32, p = 0.754; Std. beta = -0.03, 95% CI [-0.26, 0.20])  
## - The effect of Sensor [Micasense RedEdge-MX] is statistically significant and negative (beta = -0.23, 95% CI [-0.40, -0.06], t(10) = -2.94, p = 0.015; Std. beta = -0.40, 95% CI [-0.71, -0.10])  
## - The effect of Band [B, G, R, RE, NIR] is statistically non-significant and positive (beta = 0.02, 95% CI [-0.15, 0.19], t(10) = 0.24, p = 0.816; Std. beta = 0.03, 95% CI [-0.27, 0.34])  
## - The effect of Altitude m is statistically significant and positive (beta = 1.42, 95% CI [1.32, 1.51], t(10) = 31.82, p < .001; Std. beta = -0.81, 95% CI [-0.99, -0.63])  
## - The effect of Algorithm [RF] is statistically significant and positive (beta = 0.58, 95% CI [0.47, 0.70], t(10) = 11.32, p < .001; Std. beta = 1.03, 95% CI [0.83, 1.24])  
## - The effect of Algorithm [BP] is statistically significant and positive (beta = 1.25, 95% CI [1.15, 1.36], t(10) = 26.53, p < .001; Std. beta = 2.23, 95% CI [2.04, 2.41])  
## - The effect of Algorithm [SVM] is statistically significant and negative (beta = -0.71, 95% CI [-0.81, -0.61], t(10) = -16.11, p < .001; Std. beta = -1.26, 95% CI [-1.44, -1.09])  
## - The effect of Algorithm [PLS] is statistically significant and positive (beta = 0.66, 95% CI [0.56, 0.75], t(10) = 14.87, p < .001; Std. beta = 1.17, 95% CI [0.99, 1.34])  
## - The effect of Algorithm [ANN] is statistically non-significant and negative (beta = -0.09, 95% CI [-0.19, 0.02], t(10) = -1.80, p = 0.103; Std. beta = -0.15, 95% CI [-0.34, 0.04])  
## - The effect of Algorithm [XGBoost] is statistically significant and negative (beta = -0.45, 95% CI [-0.58, -0.32], t(10) = -7.69, p < .001; Std. beta = -0.80, 95% CI [-1.03, -0.57])  
## - The effect of Algorithm [MARS] is statistically non-significant and negative (beta = -0.06, 95% CI [-0.17, 0.04], t(10) = -1.30, p = 0.222; Std. beta = -0.11, 95% CI [-0.30, 0.08])

* Summary

library(flextable)

##   
## Attaching package: 'flextable'

## The following objects are masked from 'package:kableExtra':  
##   
## as\_image, footnote

## The following object is masked from 'package:purrr':  
##   
## compose

library(officer)

##   
## Attaching package: 'officer'

## The following object is masked from 'package:readxl':  
##   
## read\_xlsx

#stargazer(biomass\_model, type = "text")  
#sjPlot::tab\_model(biomass\_model, show.p = T, show.ci = T)  
# Create a summary of the model  
biomass\_model\_summary <- summary(biomass\_model)  
  
# Extract coefficients  
biomass\_coefficients <- as.data.frame(biomass\_model\_summary$coefficients)  
  
# Create a beautiful table  
  
# Create a flextable object  
# ft <- flextable(biomass\_coefficients)  
#   
# # Customize the flextable  
# ft <- theme\_vanilla(ft)  
# ft <- autofit(ft)  
# ft <- set\_caption(ft, caption = "Customized Sample Table")  
#   
# # Additional styling  
# ft <- bold(ft, part = "header")  
# ft <- bg(ft, part = "header", bg = "lightblue")  
# ft <- color(ft, part = "header", color = "white")  
# ft <- border\_remove(ft)  
# ft <- border\_outer(ft, border = fp\_border(color = "black", width = 1))  
# ft <- border\_inner\_h(ft, border = fp\_border(color = "gray", width = 0.5))  
# ft <- border\_inner\_v(ft, border = fp\_border(color = "gray", width = 0.5))  
#   
# # Display the flextable  
# ft  
# Create beautiful table  
kable(biomass\_coefficients, format = "simple")

|  | Estimate | Std. Error | t value | Pr(>|t|) |
| --- | --- | --- | --- | --- |
| (Intercept) | 1.4153405 | 0.0444809 | 31.8190615 | 0.0000000 |
| Cropoats | 0.5817703 | 0.0514031 | 11.3178006 | 0.0000005 |
| Cropstrawberry | 1.2532661 | 0.0472377 | 26.5310557 | 0.0000000 |
| Stagepre-heading | -0.7109344 | 0.0441242 | -16.1121231 | 0.0000000 |
| UAV\_TypeDJ M100 four-rotator | 0.6559266 | 0.0441242 | 14.8654636 | 0.0000000 |
| AlgorithmRF | -0.0858758 | 0.0478097 | -1.7962018 | 0.1026854 |
| AlgorithmBP | -0.4508564 | 0.0586409 | -7.6884255 | 0.0000166 |
| AlgorithmSVM | -0.0622463 | 0.0478097 | -1.3019601 | 0.2221203 |
| AlgorithmPLS | -0.0029105 | 0.0674633 | -0.0431415 | 0.9664380 |
| AlgorithmANN | -0.0186130 | 0.0577721 | -0.3221787 | 0.7539515 |
| AlgorithmXGBoost | -0.2273210 | 0.0772480 | -2.9427423 | 0.0147163 |
| AlgorithmMARS | 0.0184683 | 0.0772480 | 0.2390774 | 0.8158741 |

### Yield

* Model

report\_model(yield\_model)

## linear model (estimated using OLS) to predict RPD with Crop, Stage, UAV\_Type, Sensor, Band, Altitude\_m and Algorithm (formula: RPD ~ Crop + Stage + UAV\_Type + Sensor + Band + Altitude\_m + Algorithm)

* Performance

report\_performance(yield\_model)

## The model explains a statistically not significant and substantial proportion  
## of variance (R2 = 0.58, F(10, 2) = 0.28, p = 0.934, adj. R2 = -1.52)

* Parameters

report\_parameters(yield\_model)

## - The intercept is statistically non-significant and positive (beta = 2.03, 95% CI [-2.50, 6.56], t(2) = 1.93, p = 0.194; Std. beta = 0.41, 95% CI [-6.42, 7.24])  
## - The effect of Crop [rice] is statistically non-significant and negative (beta = -1.01, 95% CI [-7.41, 5.40], t(2) = -0.68, p = 0.568; Std. beta = -1.52, 95% CI [-11.18, 8.14])  
## - The effect of Crop [grapevine] is statistically non-significant and negative (beta = -0.05, 95% CI [-6.46, 6.35], t(2) = -0.04, p = 0.975; Std. beta = -0.08, 95% CI [-9.73, 9.58])  
## - The effect of Crop [wheat] is statistically non-significant and negative (beta = -0.10, 95% CI [-6.50, 6.30], t(2) = -0.07, p = 0.953; Std. beta = -0.15, 95% CI [-9.81, 9.51])  
## - The effect of Stage [booting] is statistically non-significant and negative (beta = -0.61, 95% CI [-7.01, 5.79], t(2) = -0.41, p = 0.722; Std. beta = -0.92, 95% CI [-10.57, 8.74])  
## - The effect of Stage [fruit set] is statistically non-significant and negative (beta = -0.87, 95% CI [-7.27, 5.54], t(2) = -0.58, p = 0.620; Std. beta = -1.30, 95% CI [-10.96, 8.35])  
## - The effect of Stage [veraison] is statistically non-significant and negative (beta = -0.33, 95% CI [-6.73, 6.07], t(2) = -0.22, p = 0.845; Std. beta = -0.50, 95% CI [-10.16, 9.16])  
## - The effect of Stage [after harvest] is statistically non-significant and positive (beta = 0.05, 95% CI [-6.36, 6.45], t(2) = 0.03, p = 0.978; Std. beta = 0.07, 95% CI [-9.59, 9.73])  
## - The effect of Stage [flowering] is statistically non-significant and positive (beta = 0.44, 95% CI [-4.78, 5.67], t(2) = 0.37, p = 0.749; Std. beta = 0.67, 95% CI [-7.21, 8.56])  
## - The effect of Stage [grain filling] is statistically non-significant and positive (beta = 0.07, 95% CI [-6.33, 6.48], t(2) = 0.05, p = 0.964; Std. beta = 0.11, 95% CI [-9.54, 9.77])  
## - The effect of UAV Type [DJI Matrice 210 quadcopter] is statistically non-significant and negative (beta = -0.12, 95% CI [-6.53, 6.28], t(2) = -0.08, p = 0.941; Std. beta = -0.19, 95% CI [-9.84, 9.47])  
## - The effect of UAV Type [Microdrones Md4-1000 quadcopter] is statistically non-significant and positive (beta = 2.03, 95% CI [-2.50, 6.56], t(2) = 1.93, p = 0.194; Std. beta = 0.41, 95% CI [-6.42, 7.24])  
## - The effect of UAV Type [eBee Ag] is statistically non-significant and negative (beta = -1.01, 95% CI [-7.41, 5.40], t(2) = -0.68, p = 0.568; Std. beta = -1.52, 95% CI [-11.18, 8.14])  
## - The effect of Sensor [Micasense Altum] is statistically non-significant and negative (beta = -0.05, 95% CI [-6.46, 6.35], t(2) = -0.04, p = 0.975; Std. beta = -0.08, 95% CI [-9.73, 9.58])  
## - The effect of Band [G, R, RE, NIR, TIR] is statistically non-significant and negative (beta = -0.10, 95% CI [-6.50, 6.30], t(2) = -0.07, p = 0.953; Std. beta = -0.15, 95% CI [-9.81, 9.51])  
## - The effect of Altitude m is statistically non-significant and negative (beta = -0.61, 95% CI [-7.01, 5.79], t(2) = -0.41, p = 0.722; Std. beta = -0.92, 95% CI [-10.57, 8.74])  
## - The effect of Algorithm [RF] is statistically non-significant and negative (beta = -0.87, 95% CI [-7.27, 5.54], t(2) = -0.58, p = 0.620; Std. beta = -1.30, 95% CI [-10.96, 8.35])  
## - The effect of Algorithm [ANN] is statistically non-significant and negative (beta = -0.33, 95% CI [-6.73, 6.07], t(2) = -0.22, p = 0.845; Std. beta = -0.50, 95% CI [-10.16, 9.16])  
## - The effect of Algorithm [XGBoost] is statistically non-significant and positive (beta = 0.05, 95% CI [-6.36, 6.45], t(2) = 0.03, p = 0.978; Std. beta = 0.07, 95% CI [-9.59, 9.73])  
## - The effect of Algorithm [MLP] is statistically non-significant and positive (beta = 0.44, 95% CI [-4.78, 5.67], t(2) = 0.37, p = 0.749; Std. beta = 0.67, 95% CI [-7.21, 8.56])  
## - The effect of Algorithm [GP] is statistically non-significant and positive (beta = 0.07, 95% CI [-6.33, 6.48], t(2) = 0.05, p = 0.964; Std. beta = 0.11, 95% CI [-9.54, 9.77])  
## - The effect of Algorithm [PF] is statistically non-significant and negative (beta = -0.12, 95% CI [-6.53, 6.28], t(2) = -0.08, p = 0.941; Std. beta = -0.19, 95% CI [-9.84, 9.47])

* Summary

#stargazer(yield\_model, type = "text")  
#sjPlot::tab\_model(yield\_model, show.p = T, show.ci = T)  
# Create a summary of the model  
yield\_model\_summary <- summary(yield\_model)  
  
# Extract coefficients  
yield\_coefficients <- as.data.frame(yield\_model\_summary$coefficients)  
  
# Create a beautiful table  
kable(yield\_coefficients, format = "simple")

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Estimate** | **Std. Error** | **t value** | **Pr(>|t|)** |
| (Intercept) | 2.03 | 1.05 | 1.93 | 0.19 |
| Croprice | -1.01 | 1.49 | -0.68 | 0.57 |
| Cropgrapevine | -0.05 | 1.49 | -0.04 | 0.98 |
| Cropwheat | -0.10 | 1.49 | -0.07 | 0.95 |
| Stagefruit set | -0.61 | 1.49 | -0.41 | 0.72 |
| Stageveraison | -0.87 | 1.49 | -0.58 | 0.62 |
| Stageflowering | -0.33 | 1.49 | -0.22 | 0.84 |
| AlgorithmRF | 0.05 | 1.49 | 0.03 | 0.98 |
| AlgorithmANN | 0.45 | 1.22 | 0.37 | 0.75 |
| AlgorithmXGBoost | 0.07 | 1.49 | 0.05 | 0.96 |
| AlgorithmGP | -0.12 | 1.49 | -0.08 | 0.94 |

### Nitrogen

* Model

report\_model(nitrogen\_model)

## linear model (estimated using OLS) to predict RPD with Crop, Stage, UAV\_Type, Sensor, Band, Altitude\_m and Algorithm (formula: RPD ~ Crop + Stage + UAV\_Type + Sensor + Band + Altitude\_m + Algorithm)

* Performance

report\_performance(nitrogen\_model)

## The model explains a statistically not significant and substantial proportion  
## of variance (R2 = 0.90, F(19, 5) = 2.43, p = 0.166, adj. R2 = 0.53)

* Parameters

report\_parameters(nitrogen\_model)

## - The intercept is statistically non-significant and positive (beta = 1.46, 95% CI [-0.35, 3.26], t(5) = 2.07, p = 0.093; Std. beta = -0.74, 95% CI [-3.62, 2.14])  
## - The effect of Crop [rice] is statistically non-significant and positive (beta = 0.63, 95% CI [-1.14, 2.40], t(5) = 0.91, p = 0.405; Std. beta = 1.00, 95% CI [-1.82, 3.81])  
## - The effect of Crop [grapevine] is statistically non-significant and negative (beta = -0.83, 95% CI [-2.40, 0.73], t(5) = -1.37, p = 0.229; Std. beta = -1.33, 95% CI [-3.82, 1.16])  
## - The effect of Crop [wheat] is statistically non-significant and negative (beta = -0.28, 95% CI [-1.84, 1.29], t(5) = -0.46, p = 0.666; Std. beta = -0.44, 95% CI [-2.94, 2.05])  
## - The effect of Crop [tea] is statistically non-significant and negative (beta = -0.50, 95% CI [-2.42, 1.41], t(5) = -0.68, p = 0.529; Std. beta = -0.80, 95% CI [-3.85, 2.25])  
## - The effect of Crop [corn] is statistically significant and negative (beta = -1.53, 95% CI [-3.00, -0.07], t(5) = -2.69, p = 0.043; Std. beta = -2.44, 95% CI [-4.77, -0.11])  
## - The effect of Stage [late vegetative] is statistically non-significant and negative (beta = -0.22, 95% CI [-1.78, 1.35], t(5) = -0.36, p = 0.735; Std. beta = -0.35, 95% CI [-2.84, 2.14])  
## - The effect of Stage [booting] is statistically non-significant and negative (beta = -0.32, 95% CI [-1.23, 0.58], t(5) = -0.92, p = 0.400; Std. beta = -0.51, 95% CI [-1.95, 0.92])  
## - The effect of Stage [flowering] is statistically non-significant and negative (beta = -0.87, 95% CI [-1.78, 0.03], t(5) = -2.48, p = 0.056; Std. beta = -1.39, 95% CI [-2.83, 0.05])  
## - The effect of Stage [vegetative] is statistically significant and positive (beta = 1.57, 95% CI [0.46, 2.68], t(5) = 3.64, p = 0.015; Std. beta = 2.50, 95% CI [0.73, 4.26])  
## - The effect of Stage [reproductive] is statistically non-significant and positive (beta = 0.86, 95% CI [-0.57, 2.29], t(5) = 1.54, p = 0.184; Std. beta = 1.36, 95% CI [-0.91, 3.64])  
## - The effect of Stage [ripening] is statistically non-significant and positive (beta = 0.66, 95% CI [-1.15, 2.46], t(5) = 0.93, p = 0.394; Std. beta = 1.04, 95% CI [-1.83, 3.92])  
## - The effect of Stage [new shoot growth] is statistically non-significant and positive (beta = 0.79, 95% CI [-0.12, 1.69], t(5) = 2.23, p = 0.076; Std. beta = 1.25, 95% CI [-0.19, 2.69])  
## - The effect of Stage [leaf development] is statistically non-significant and positive (beta = 0.85, 95% CI [-0.96, 2.66], t(5) = 1.21, p = 0.281; Std. beta = 1.35, 95% CI [-1.52, 4.23])  
## - The effect of Stage [stem elongation] is statistically significant and positive (beta = 1.04, 95% CI [0.14, 1.95], t(5) = 2.96, p = 0.031; Std. beta = 1.66, 95% CI [0.22, 3.10])  
## - The effect of UAV Type [DJ M100 four-rotator] is statistically non-significant and positive (beta = 0.86, 95% CI [-1.26, 2.98], t(5) = 1.04, p = 0.346; Std. beta = 1.36, 95% CI [-2.01, 4.74])  
## - The effect of UAV Type [DJI Matrice 210 quadcopter] is statistically non-significant and positive (beta = 0.80, 95% CI [-1.32, 2.91], t(5) = 0.96, p = 0.379; Std. beta = 1.27, 95% CI [-2.11, 4.64])  
## - The effect of UAV Type [eBee Ag] is statistically non-significant and positive (beta = 0.80, 95% CI [-1.32, 2.91], t(5) = 0.96, p = 0.379; Std. beta = 1.27, 95% CI [-2.11, 4.64])  
## - The effect of UAV Type [AscTec Pelican] is statistically non-significant and positive (beta = 0.54, 95% CI [-1.58, 2.66], t(5) = 0.66, p = 0.540; Std. beta = 0.86, 95% CI [-2.51, 4.24])  
## - The effect of Sensor [Yusense MS600] is statistically non-significant and positive (beta = 0.70, 95% CI [-1.42, 2.82], t(5) = 0.85, p = 0.435; Std. beta = 1.11, 95% CI [-2.26, 4.49])  
## - The effect of Sensor [Micasense RedEdge-3] is statistically non-significant and positive (beta = 1.46, 95% CI [-0.35, 3.26], t(5) = 2.07, p = 0.093; Std. beta = -0.74, 95% CI [-3.62, 2.14])  
## - The effect of Band [B, G, R, RE, NIR] is statistically non-significant and positive (beta = 0.63, 95% CI [-1.14, 2.40], t(5) = 0.91, p = 0.405; Std. beta = 1.00, 95% CI [-1.82, 3.81])  
## - The effect of Altitude m is statistically non-significant and negative (beta = -0.83, 95% CI [-2.40, 0.73], t(5) = -1.37, p = 0.229; Std. beta = -1.33, 95% CI [-3.82, 1.16])  
## - The effect of Algorithm [RF] is statistically non-significant and negative (beta = -0.28, 95% CI [-1.84, 1.29], t(5) = -0.46, p = 0.666; Std. beta = -0.44, 95% CI [-2.94, 2.05])  
## - The effect of Algorithm [BP] is statistically non-significant and negative (beta = -0.50, 95% CI [-2.42, 1.41], t(5) = -0.68, p = 0.529; Std. beta = -0.80, 95% CI [-3.85, 2.25])  
## - The effect of Algorithm [SVM] is statistically significant and negative (beta = -1.53, 95% CI [-3.00, -0.07], t(5) = -2.69, p = 0.043; Std. beta = -2.44, 95% CI [-4.77, -0.11])  
## - The effect of Algorithm [PLS] is statistically non-significant and negative (beta = -0.22, 95% CI [-1.78, 1.35], t(5) = -0.36, p = 0.735; Std. beta = -0.35, 95% CI [-2.84, 2.14])  
## - The effect of Algorithm [ANN] is statistically non-significant and negative (beta = -0.32, 95% CI [-1.23, 0.58], t(5) = -0.92, p = 0.400; Std. beta = -0.51, 95% CI [-1.95, 0.92])  
## - The effect of Algorithm [XGBoost] is statistically non-significant and negative (beta = -0.87, 95% CI [-1.78, 0.03], t(5) = -2.48, p = 0.056; Std. beta = -1.39, 95% CI [-2.83, 0.05])  
## - The effect of Algorithm [QDA] is statistically significant and positive (beta = 1.57, 95% CI [0.46, 2.68], t(5) = 3.64, p = 0.015; Std. beta = 2.50, 95% CI [0.73, 4.26])  
## - The effect of Algorithm [DNN] is statistically non-significant and positive (beta = 0.86, 95% CI [-0.57, 2.29], t(5) = 1.54, p = 0.184; Std. beta = 1.36, 95% CI [-0.91, 3.64])  
## - The effect of Algorithm [REPT] is statistically non-significant and positive (beta = 0.66, 95% CI [-1.15, 2.46], t(5) = 0.93, p = 0.394; Std. beta = 1.04, 95% CI [-1.83, 3.92])  
## - The effect of Algorithm [KNN] is statistically non-significant and positive (beta = 0.79, 95% CI [-0.12, 1.69], t(5) = 2.23, p = 0.076; Std. beta = 1.25, 95% CI [-0.19, 2.69])

* Summary

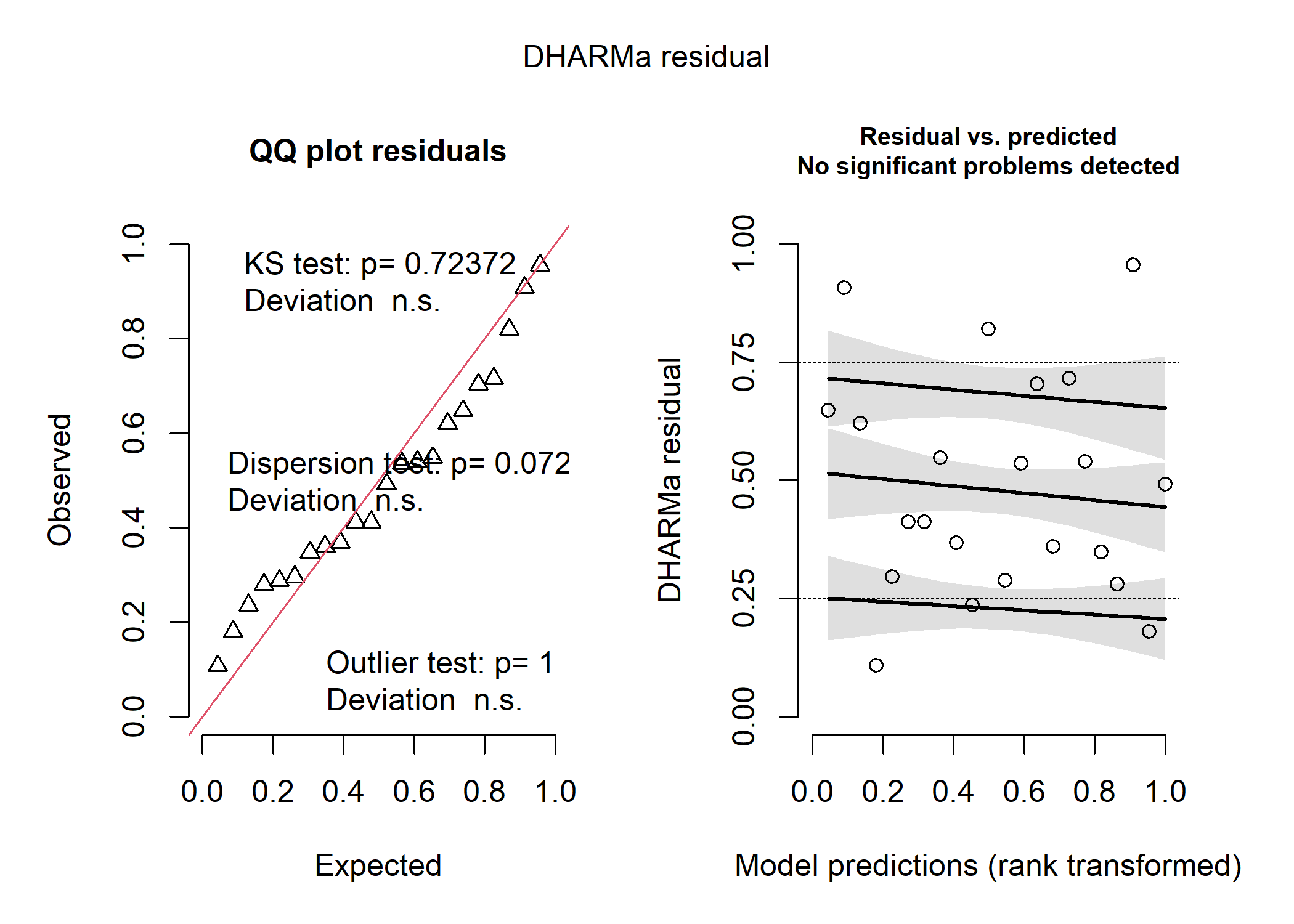
#stargazer(nitrogen\_model, type = "text")  
#sjPlot::tab\_model(nitrogen\_model, show.p = T, show.ci = T)  
# Create a summary of the model  
nitrogen\_model\_summary <- summary(nitrogen\_model)  
  
# Extract coefficients  
nitrogen\_coefficients <- as.data.frame(nitrogen\_model\_summary$coefficients)  
  
# Create a beautiful table  
kable(nitrogen\_coefficients, format = "simple")

|  | Estimate | Std. Error | t value | Pr(>|t|) |
| --- | --- | --- | --- | --- |
| (Intercept) | 1.4562551 | 0.7030787 | 2.0712548 | 0.0930910 |
| Croprice | 0.6261553 | 0.6882754 | 0.9097452 | 0.4046769 |
| Cropgrapevine | -0.8349064 | 0.6088840 | -1.3712076 | 0.2286504 |
| Cropwheat | -0.2791642 | 0.6088840 | -0.4584850 | 0.6658432 |
| Croptea | -0.5036314 | 0.7457276 | -0.6753557 | 0.5294160 |
| Cropcorn | -1.5331134 | 0.5695588 | -2.6917560 | 0.0432078 |
| Stagebooting | -0.2176897 | 0.6088840 | -0.3575224 | 0.7352967 |
| Stagevegetative | -0.3232314 | 0.3515394 | -0.9194742 | 0.4000331 |
| Stagereproductive | -0.8720299 | 0.3515394 | -2.4806041 | 0.0557998 |
| Stageleaf development | 1.5685123 | 0.4305460 | 3.6430771 | 0.0148555 |
| AlgorithmRF | 0.8569121 | 0.5558325 | 1.5416733 | 0.1837903 |
| AlgorithmBP | 0.6555303 | 0.7030787 | 0.9323711 | 0.3939424 |
| AlgorithmSVM | 0.7850170 | 0.3515394 | 2.2330843 | 0.0758705 |
| AlgorithmPLS | 0.8503219 | 0.7030787 | 1.2094264 | 0.2805593 |
| AlgorithmANN | 1.0420579 | 0.3515394 | 2.9642710 | 0.0313638 |
| AlgorithmXGBoost | 0.8569121 | 0.8244329 | 1.0393959 | 0.3462479 |
| AlgorithmQDA | 0.7953179 | 0.8244329 | 0.9646849 | 0.3790085 |
| AlgorithmDNN | 0.7953179 | 0.8244329 | 0.9646849 | 0.3790085 |
| AlgorithmREPT | 0.5419882 | 0.8244329 | 0.6574073 | 0.5399727 |
| AlgorithmKNN | 0.6991215 | 0.8244329 | 0.8480029 | 0.4351410 |

## Plot the residuals

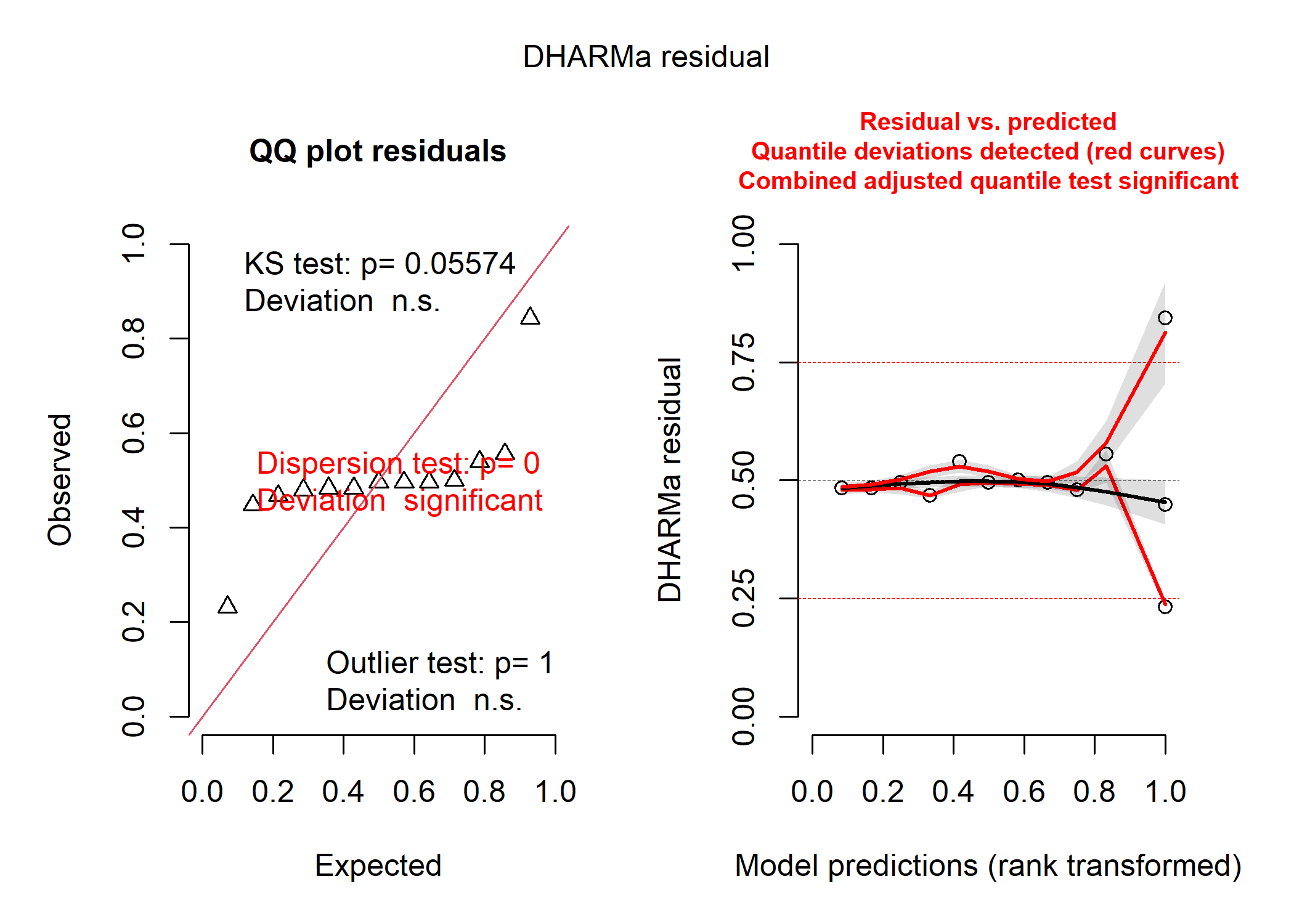
### Biomass

biomass\_simulationOutput <- simulateResiduals(fittedModel = biomass\_model)  
plot(biomass\_simulationOutput)



### Yield

yield\_simulationOutput <- simulateResiduals(fittedModel = yield\_model)  
plot(yield\_simulationOutput)



### Nitrogen

nitrogen\_simulationOutput <- simulateResiduals(fittedModel = nitrogen\_model)  
plot(nitrogen\_simulationOutput)

