

Results - Plots and Analyzes (GLMs and GAMs)

2022-12-06

Analysis of the summary data (without landscape)

0. Imported the data and define directory:

```
datao <- read.table(file = "~/Documents/phd_project/Chapter_2/output/final_simulations/output.txt", sep = ", ", header = T)
datap <- read.table(file = "~/Documents/phd_project/Chapter_2/output/final_simulations/sim_parameters.txt", sep = ", ", header = T)
data = merge(x = datao, y = datap, by = "run_number",
             all.x = TRUE)
```

1. Create a data_frame with the summary data, the means to plot the boxplot figure with the general data, settlement rate, mean habitat quality in settlement (q), mean energetic condition in settlement (en), total distance moved and euclidean distance moved:

```
summary(data_means)
```

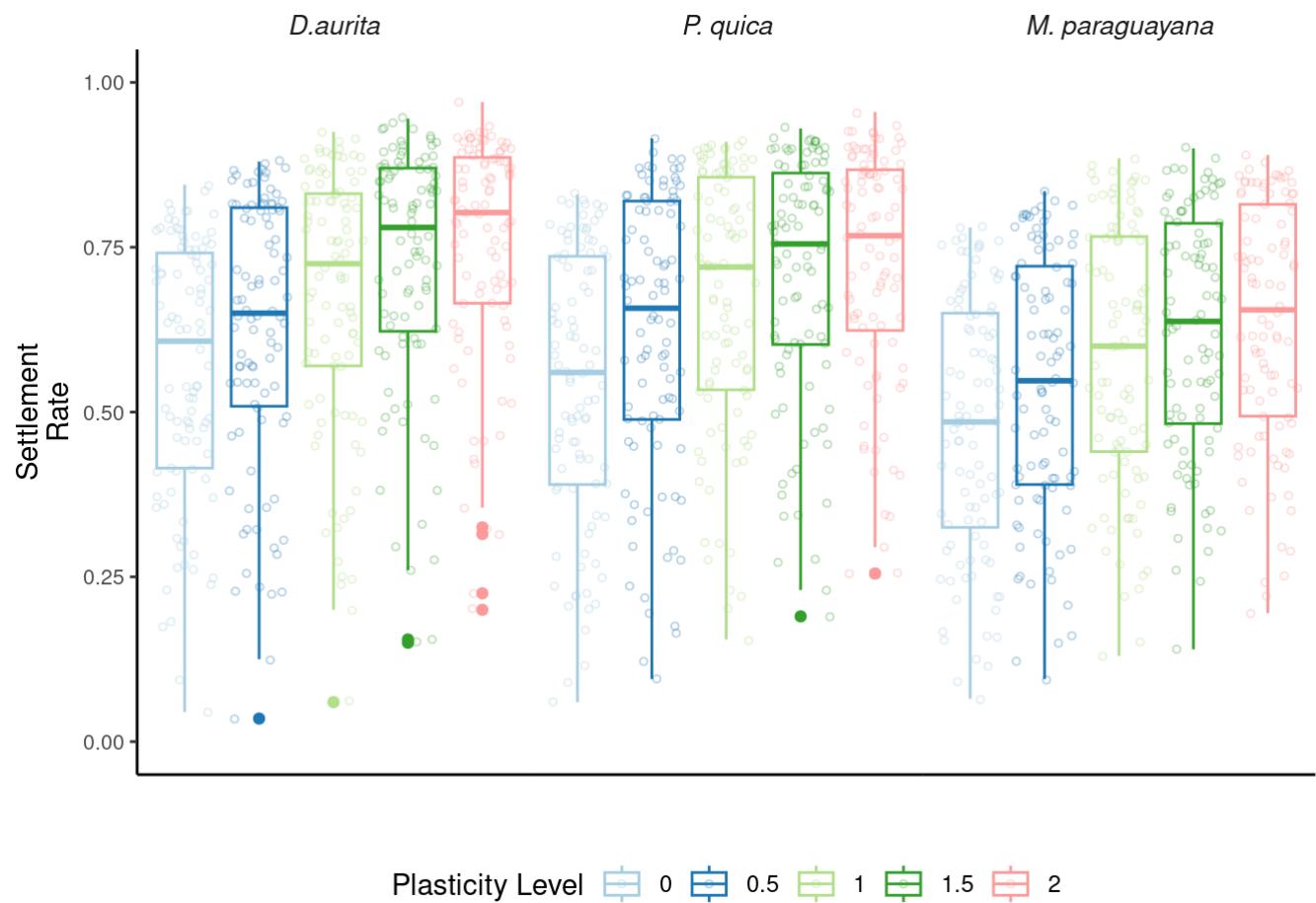
```
##   run_number      behaviour     hab_amount    clumpiness
## Min.   : 1.00   Min.   :0.0   Min.   : 5.00   Min.   : 7.00
## 1st Qu.: 75.75  1st Qu.:0.5   1st Qu.:20.00  1st Qu.:33.75
## Median :151.50  Median :1.0   Median :36.00  Median :57.00
## Mean   :150.92  Mean   :1.0   Mean   :36.58  Mean   :53.86
## 3rd Qu.:226.25 3rd Qu.:1.5   3rd Qu.:51.25  3rd Qu.:71.00
## Max.   :299.00  Max.   :2.0   Max.   :69.00  Max.   :99.00
##   initial_individuals specie      n       meanQ      meanen
## Min.   :1000        DA:480   Min.   : 7   Min.   :0.1153  Min.   :0.1378
## 1st Qu.:1000        PQ:480   1st Qu.: 97  1st Qu.:0.3313  1st Qu.:0.3903
## Median :1000        MP:480   Median :133   Median :0.4226  Median :0.5156
## Mean   :1000        Mean   :126   Mean   :0.4172  Mean   :0.4935
## 3rd Qu.:1000        3rd Qu.:162   3rd Qu.:0.5019  3rd Qu.:0.6039
## Max.   :1000        Max.   :194   Max.   :0.6943  Max.   :0.7799
##   meandt      meanld      rate
## Min.   :1.016   Min.   :0.2819  Min.   :0.035
## 1st Qu.:1.296  1st Qu.:0.4954  1st Qu.:0.485
## Median :1.652  Median :0.5982  Median :0.665
## Mean   :1.877  Mean   :0.6283  Mean   :0.630
## 3rd Qu.:2.267  3rd Qu.:0.7315  3rd Qu.:0.810
## Max.   :4.208  Max.   :1.3660  Max.   :0.970
```

2. Perfome GLM with gaussian distribution for all variables and ANOVA to analyze them:

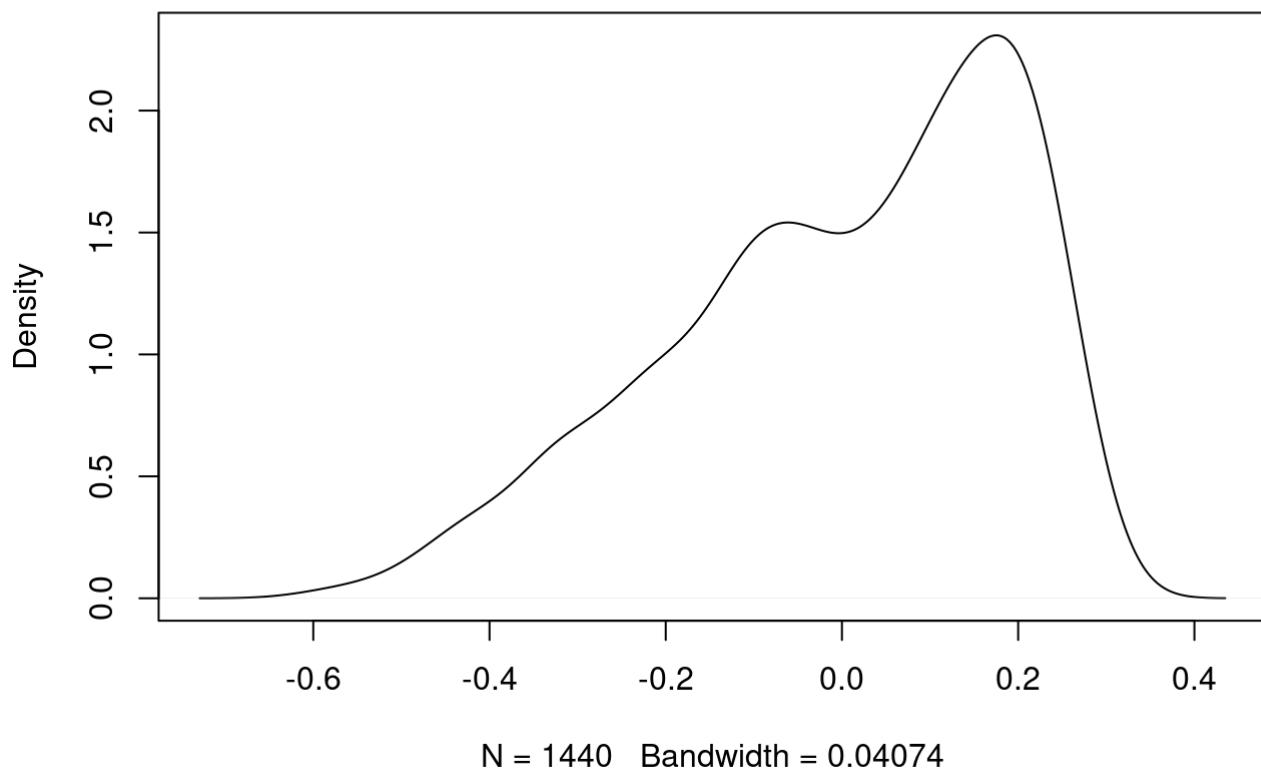
```
#settlement rate
m_set <- glm(rate ~ behaviour + specie + behaviour*specie, data=data_means,family =
"gaussian")
#habitat quality
m_q <- glm(meanQ ~ behaviour + specie + behaviour*specie, data=data_means,family = "g
aussian")
#energetic condition
m_en <- glm(meanden ~ behaviour + specie + behaviour*specie, data=data_means,family =
"gaussian")
#total distance
m_dt <- glm(meandt ~ behaviour + specie + behaviour*specie, data=data_means,family =
"gaussian")
#euclidean distance
m_ld <- glm(meanld ~ behaviour + specie + behaviour*specie, data=data_means,family =
"gaussian")
```

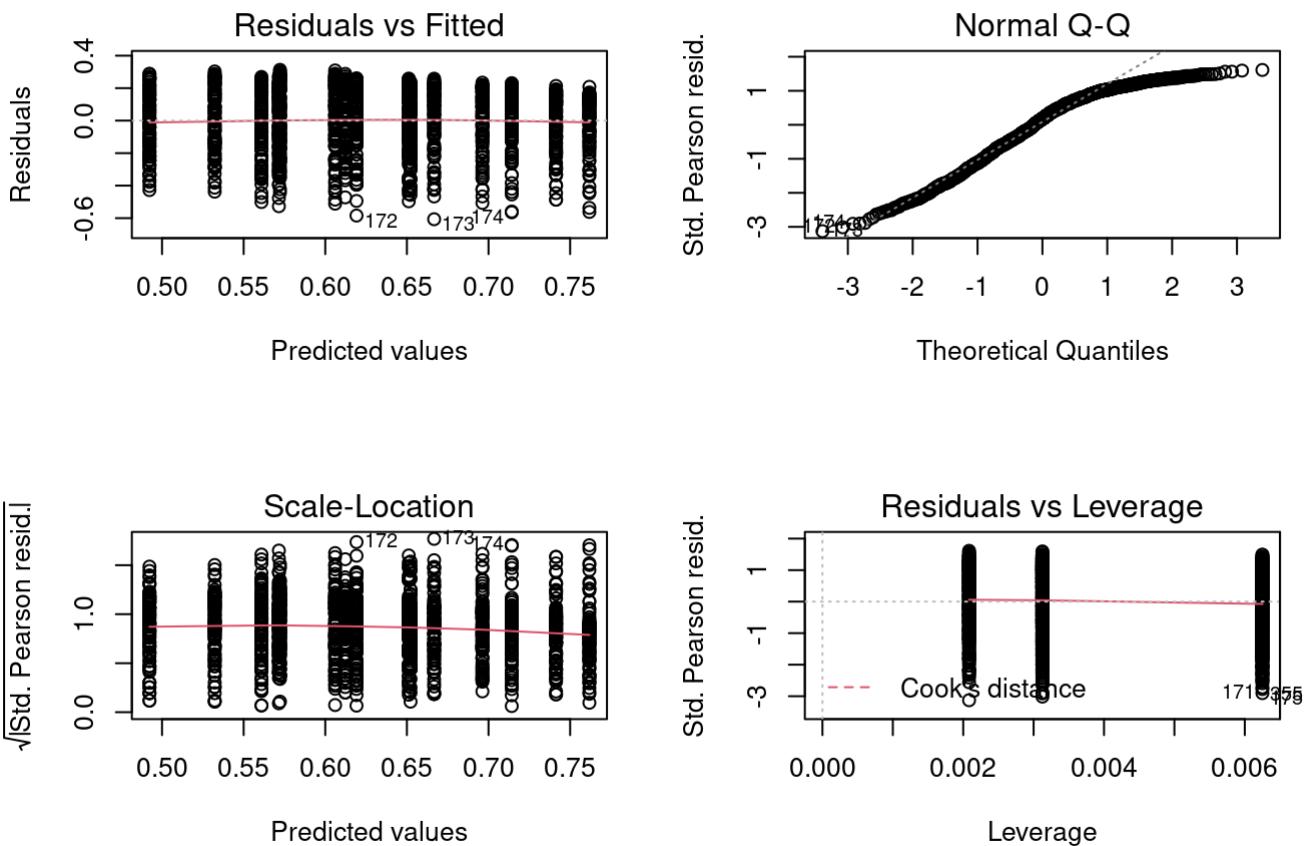
3. Print plots residuals, summary and anova outputs for each model:

Settlement Rate



density.default(x = resid(m_set, type = "response"))



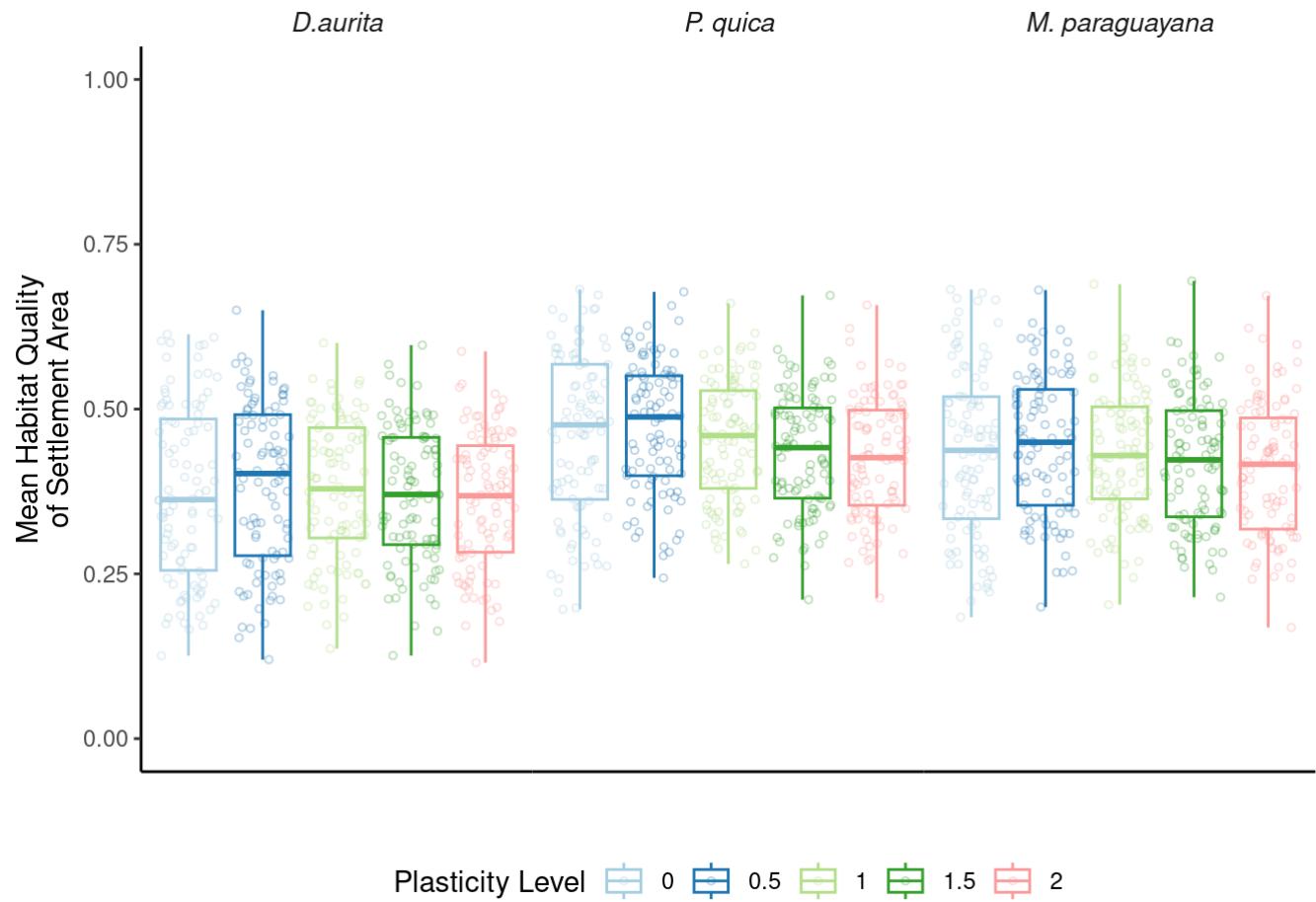


```

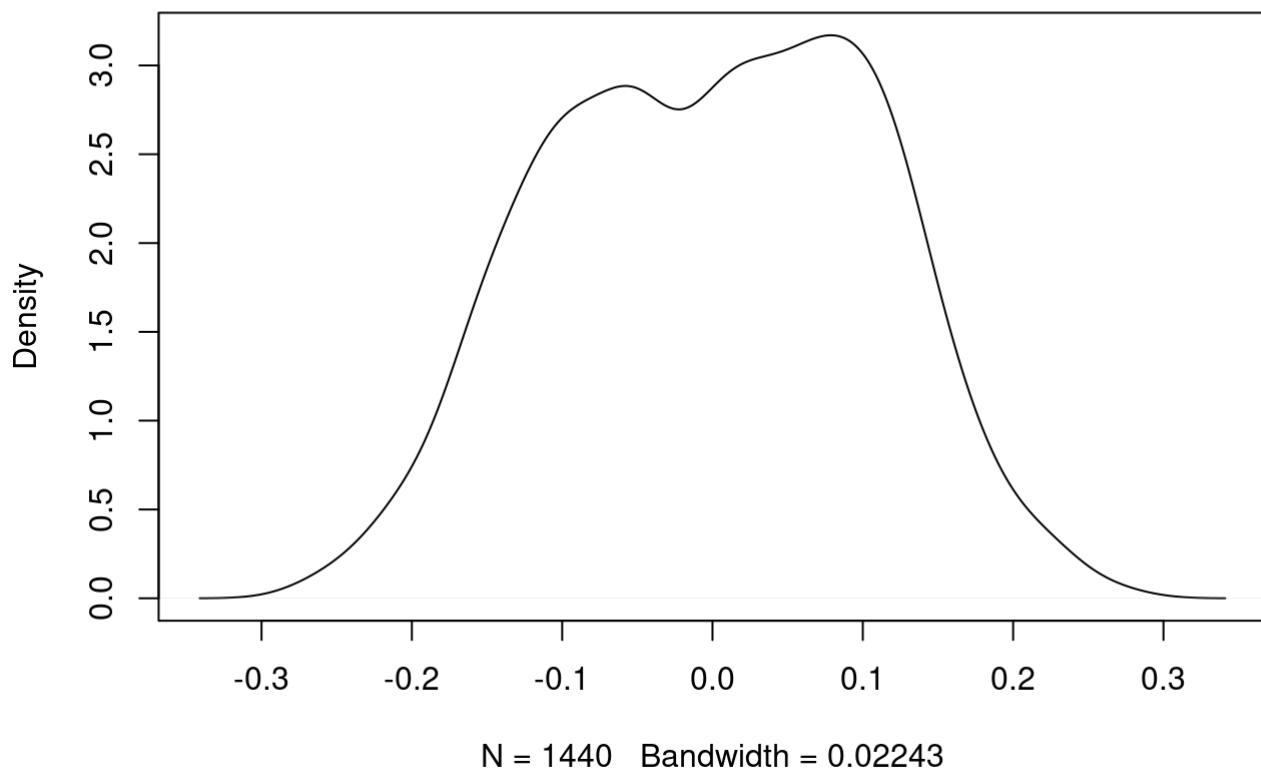
## 
## Call:
## glm(formula = rate ~ behaviour + specie + behaviour * specie,
##      family = "gaussian", data = data_means)
## 
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max 
## -0.60669 -0.12916  0.03408  0.16272  0.31274 
## 
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            0.571583  0.015353 37.231 < 2e-16 ***
## behaviour             0.095104  0.012535  7.587 5.86e-14 ***
## speciePQ              -0.010667  0.021712 -0.491  0.62330    
## specieMP              -0.079292  0.021712 -3.652  0.00027 ***
## behaviour:speciePQ   -0.004875  0.017728 -0.275  0.78336    
## behaviour:specieMP   -0.015135  0.017728 -0.854  0.39337    
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 0.03771193)
## 
## Null deviance: 62.199  on 1439  degrees of freedom
## Residual deviance: 54.079  on 1434  degrees of freedom
## AIC: -625.47
## 
## Number of Fisher Scoring iterations: 2

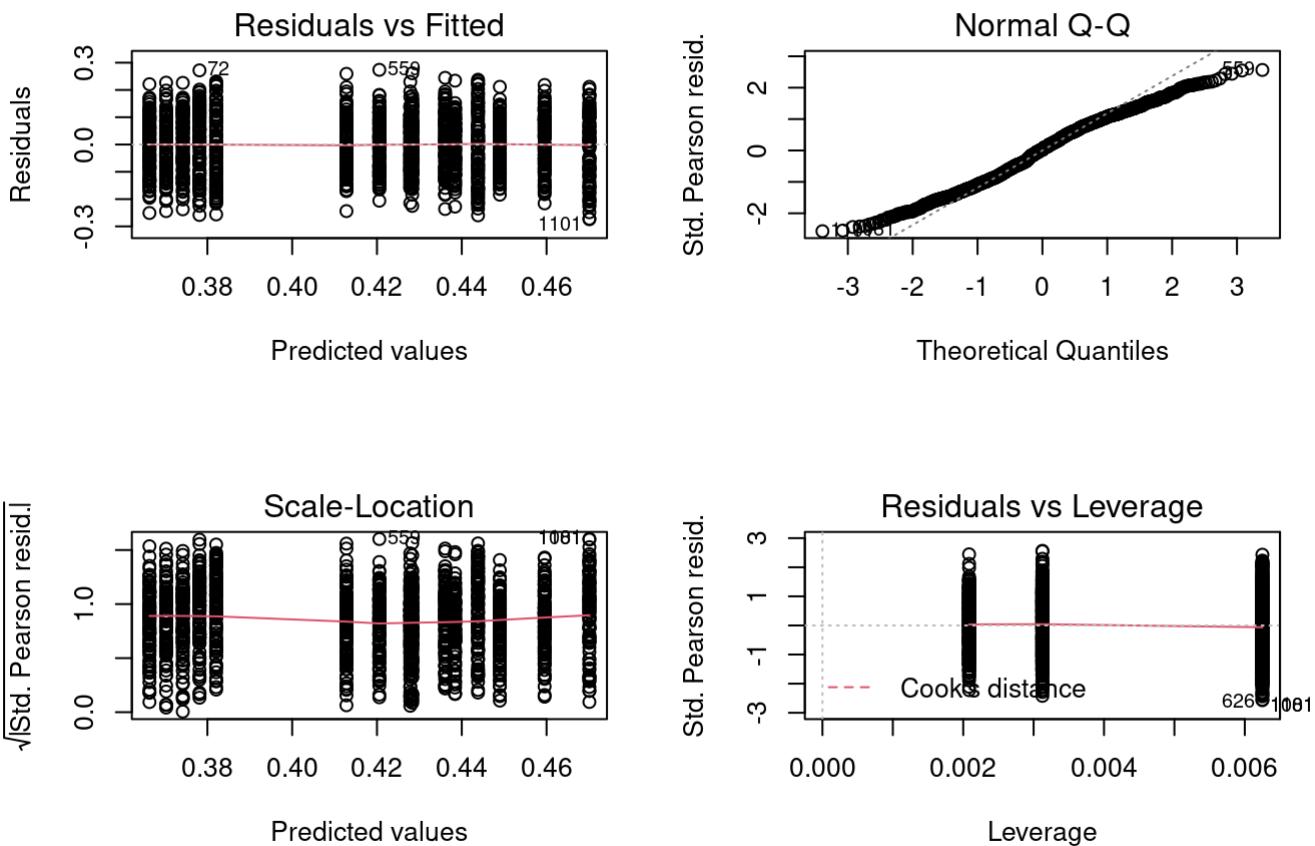
```

Habitat Quality



density.default(x = resid(m_q, type = "response"))



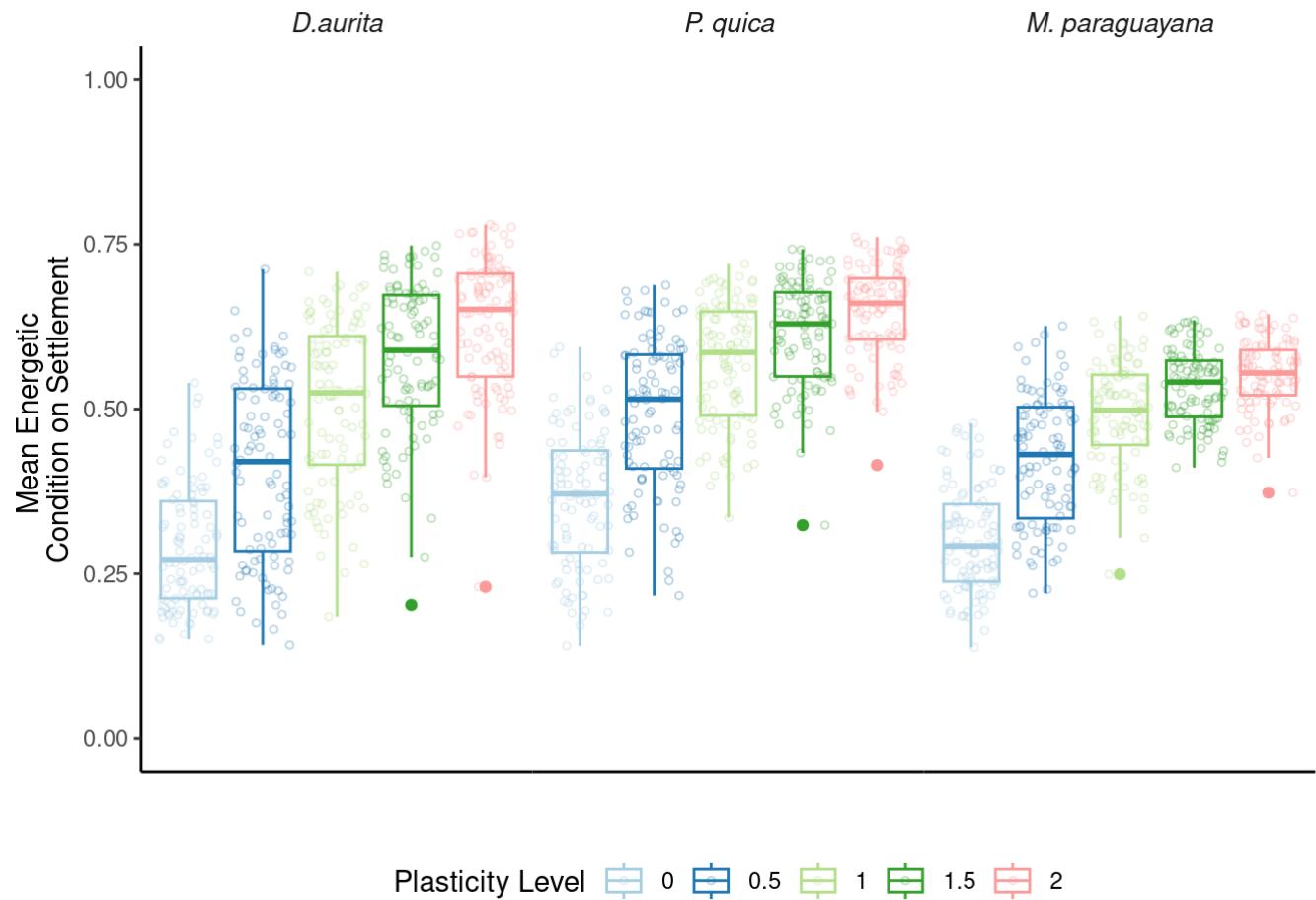


```

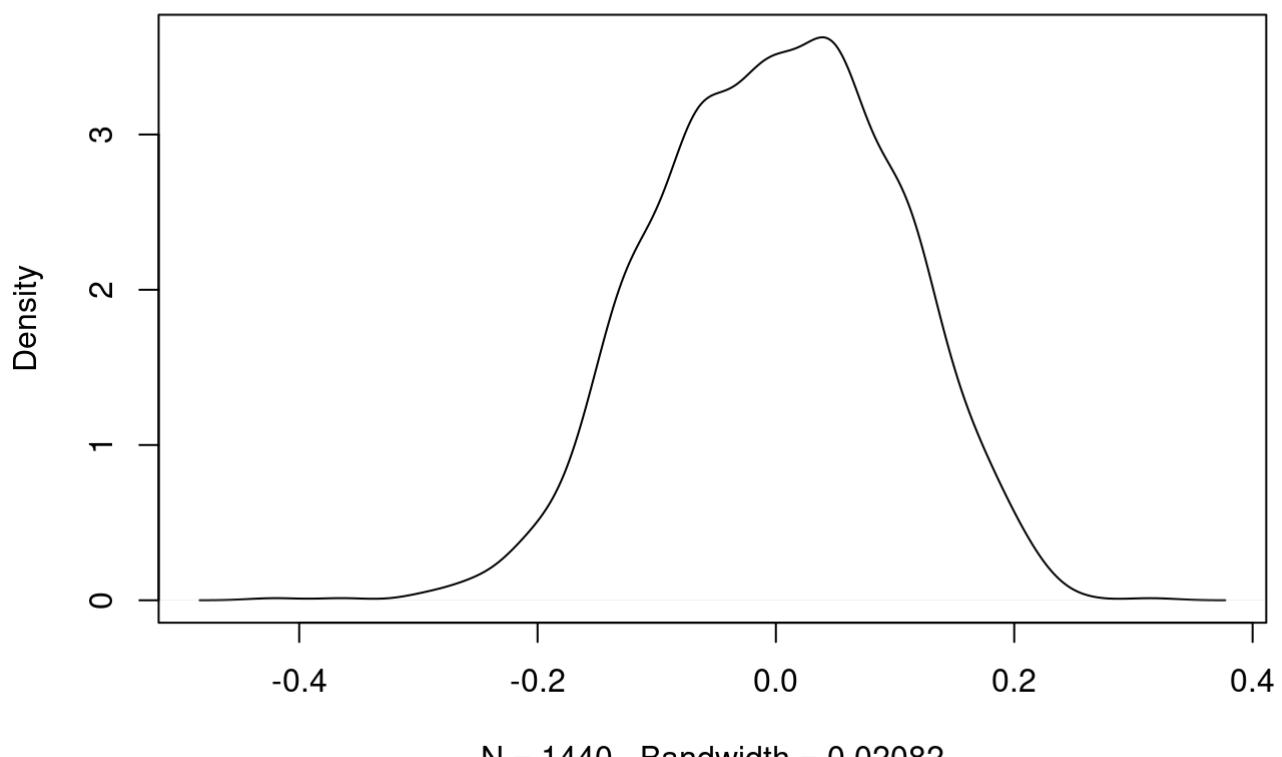
## 
## Call:
## glm(formula = meanQ ~ behaviour + specie + behaviour * specie,
##      family = "gaussian", data = data_means)
## 
## Deviance Residuals:
##      Min        1Q     Median        3Q       Max 
## -0.273847 -0.085304   0.003124   0.085456   0.273791 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.382071  0.008452 45.207 < 2e-16 ***
## behaviour   -0.007869  0.006901 -1.140   0.254    
## speciePQ    0.088029  0.011952  7.365 2.98e-13 ***
## specieMP    0.061763  0.011952  5.167 2.71e-07 ***
## behaviour:speciePQ -0.013254  0.009759 -1.358   0.175    
## behaviour:specieMP -0.007649  0.009759 -0.784   0.433    
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 0.01142866)
## 
## Null deviance: 18.000 on 1439 degrees of freedom
## Residual deviance: 16.389 on 1434 degrees of freedom
## AIC: -2344.6
## 
## Number of Fisher Scoring iterations: 2

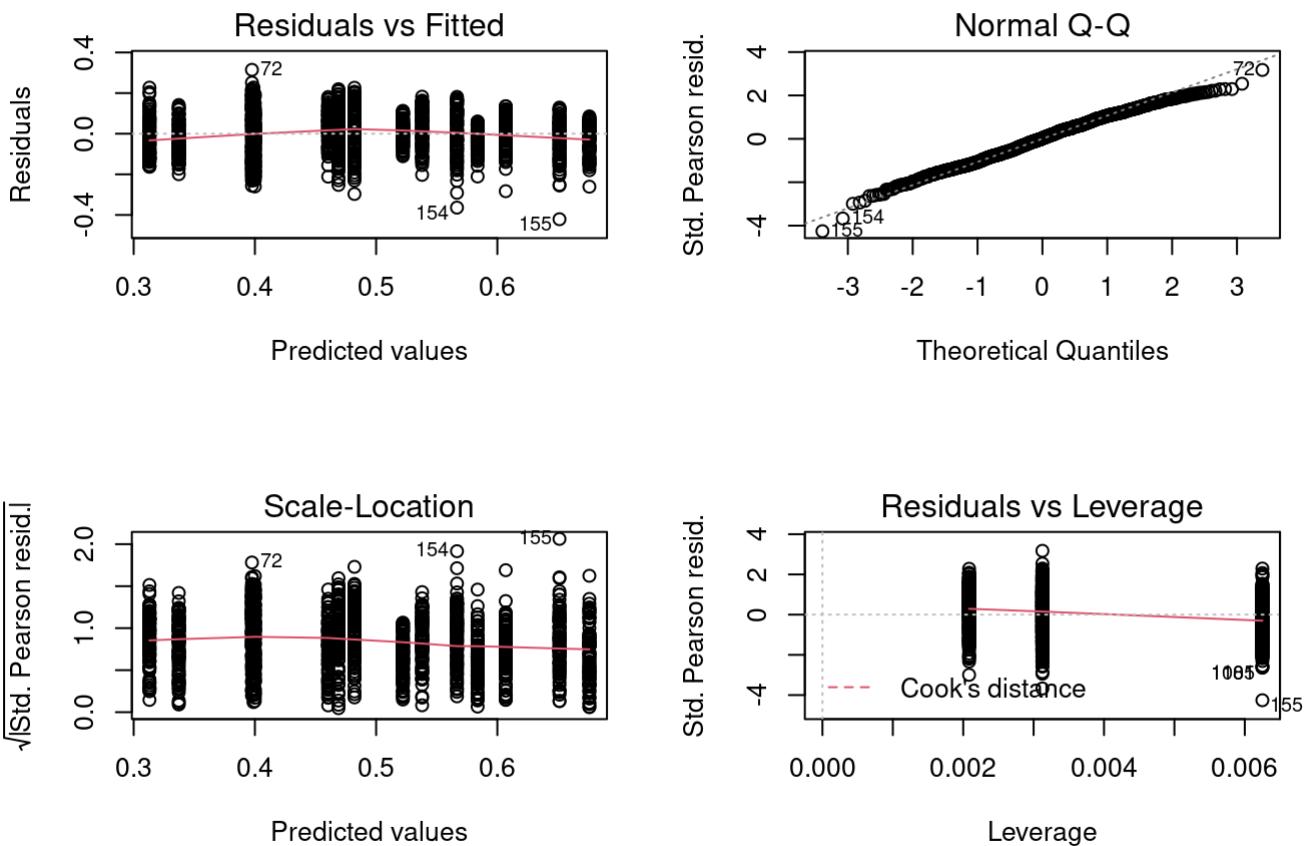
```

Energetic Conditions



density.default(x = resid(m_en, type = "response"))



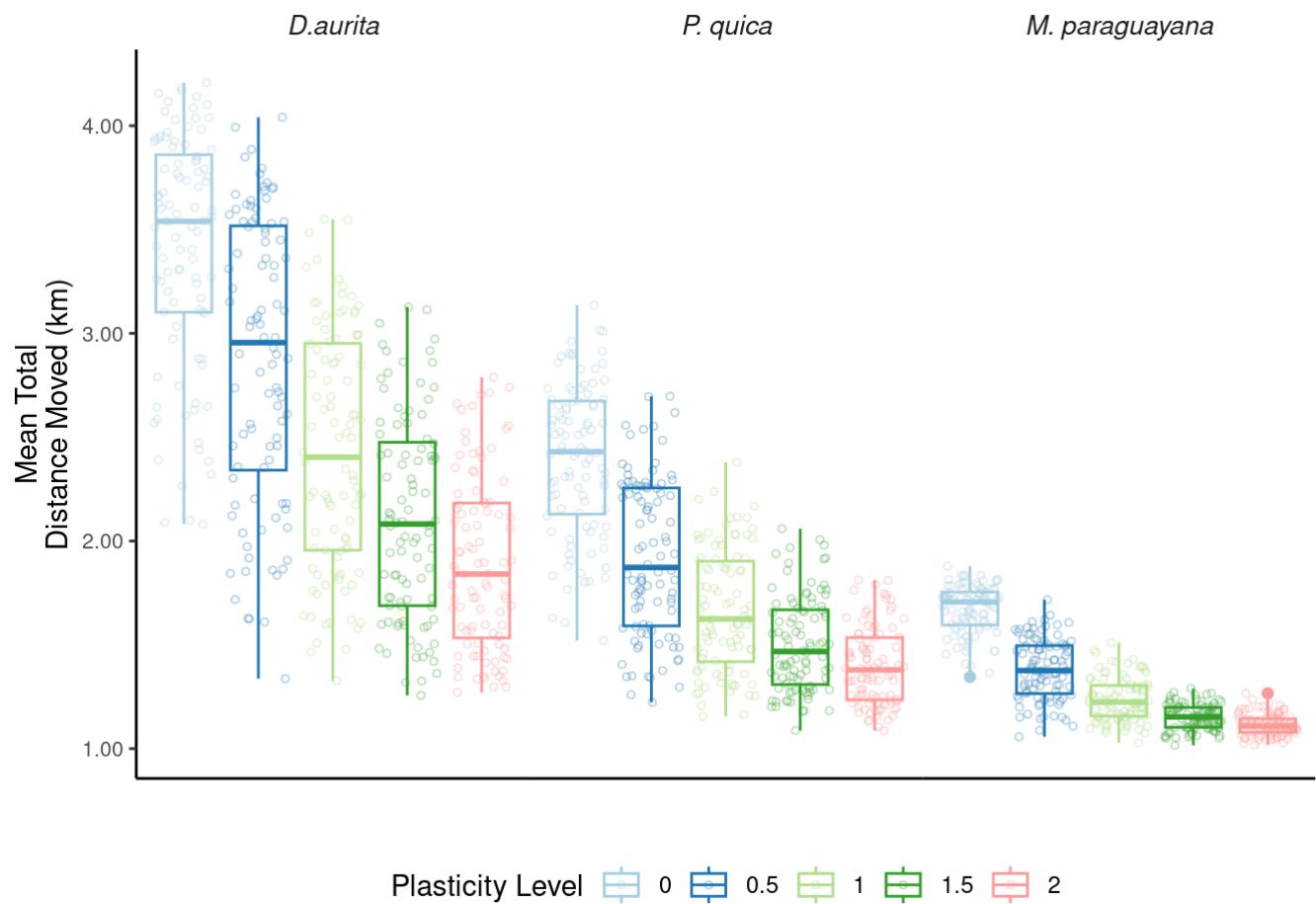


```

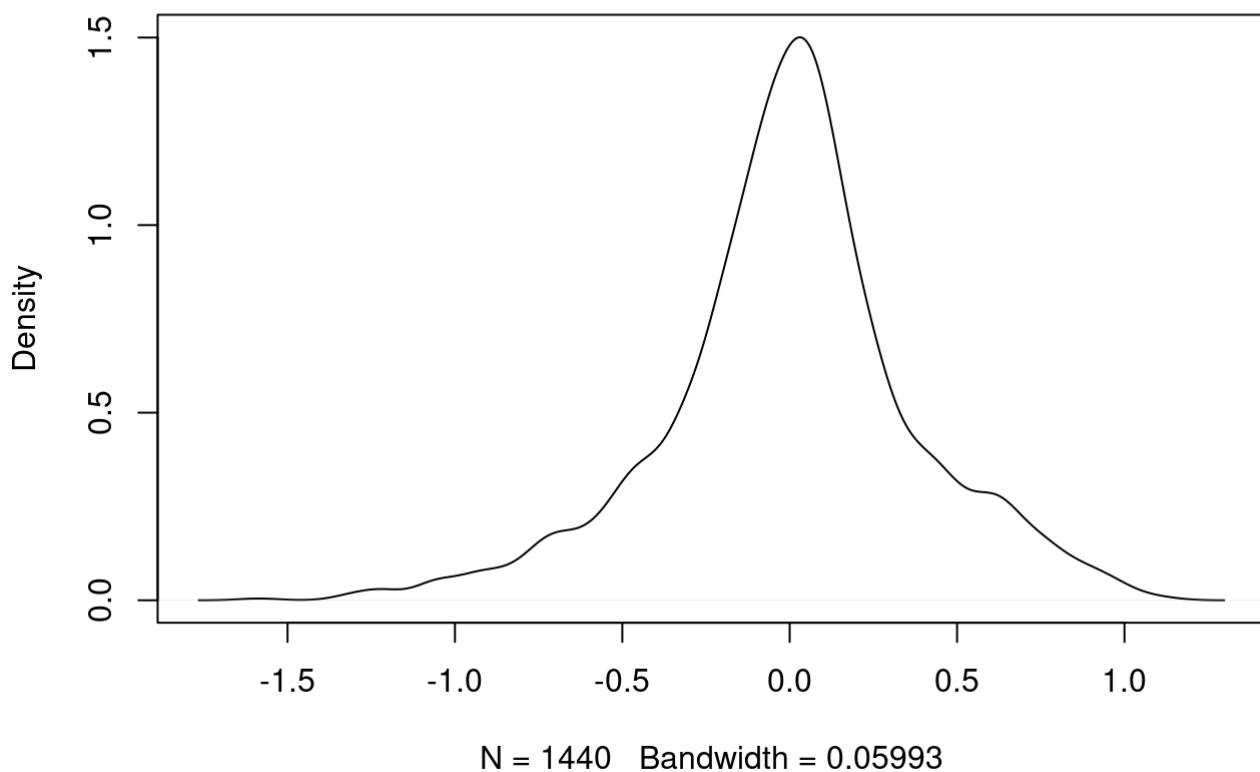
## 
## Call:
##   glm(formula = meanen ~ behaviour + specie + behaviour * specie,
##        family = "gaussian", data = data_means)
## 
## Deviance Residuals:
##       Min      1Q  Median      3Q     Max 
## -0.42104 -0.07096  0.00233  0.07174  0.31456 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.313005  0.007847 39.888 < 2e-16 ***
## behaviour   0.169125  0.006407 26.396 < 2e-16 ***
## speciePQ    0.086941  0.011097  7.834 9.12e-15 ***
## specieMP    0.024197  0.011097  2.180 0.029393 *  
## behaviour:speciePQ -0.031138  0.009061 -3.436 0.000606 *** 
## behaviour:specieMP -0.045868  0.009061 -5.062 4.68e-07 *** 
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## (Dispersion parameter for gaussian family taken to be 0.009852343)
## 
## Null deviance: 30.743 on 1439 degrees of freedom
## Residual deviance: 14.128 on 1434 degrees of freedom
## AIC: -2558.3
## 
## Number of Fisher Scoring iterations: 2

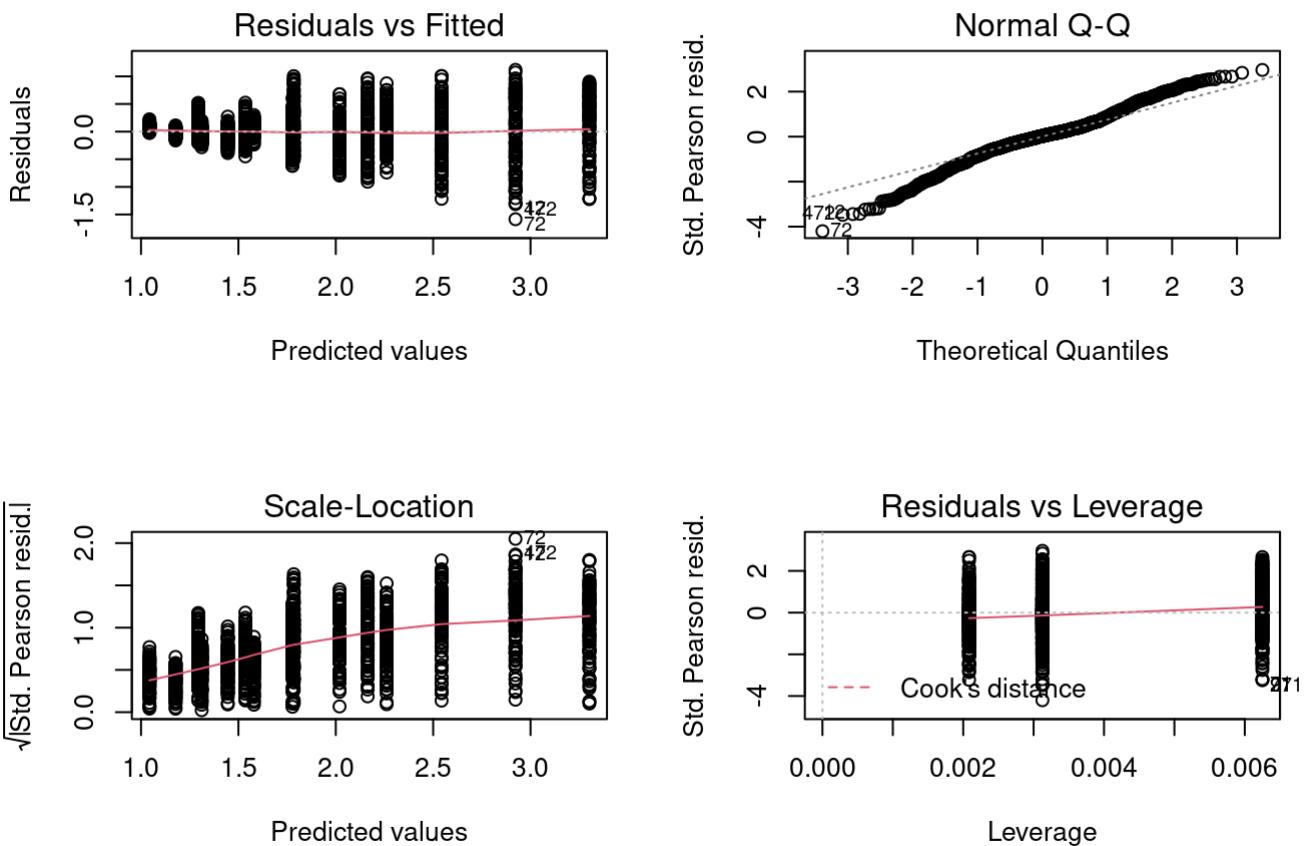
```

Distance Total



density.default(x = resid(m_dt, type = "response"))



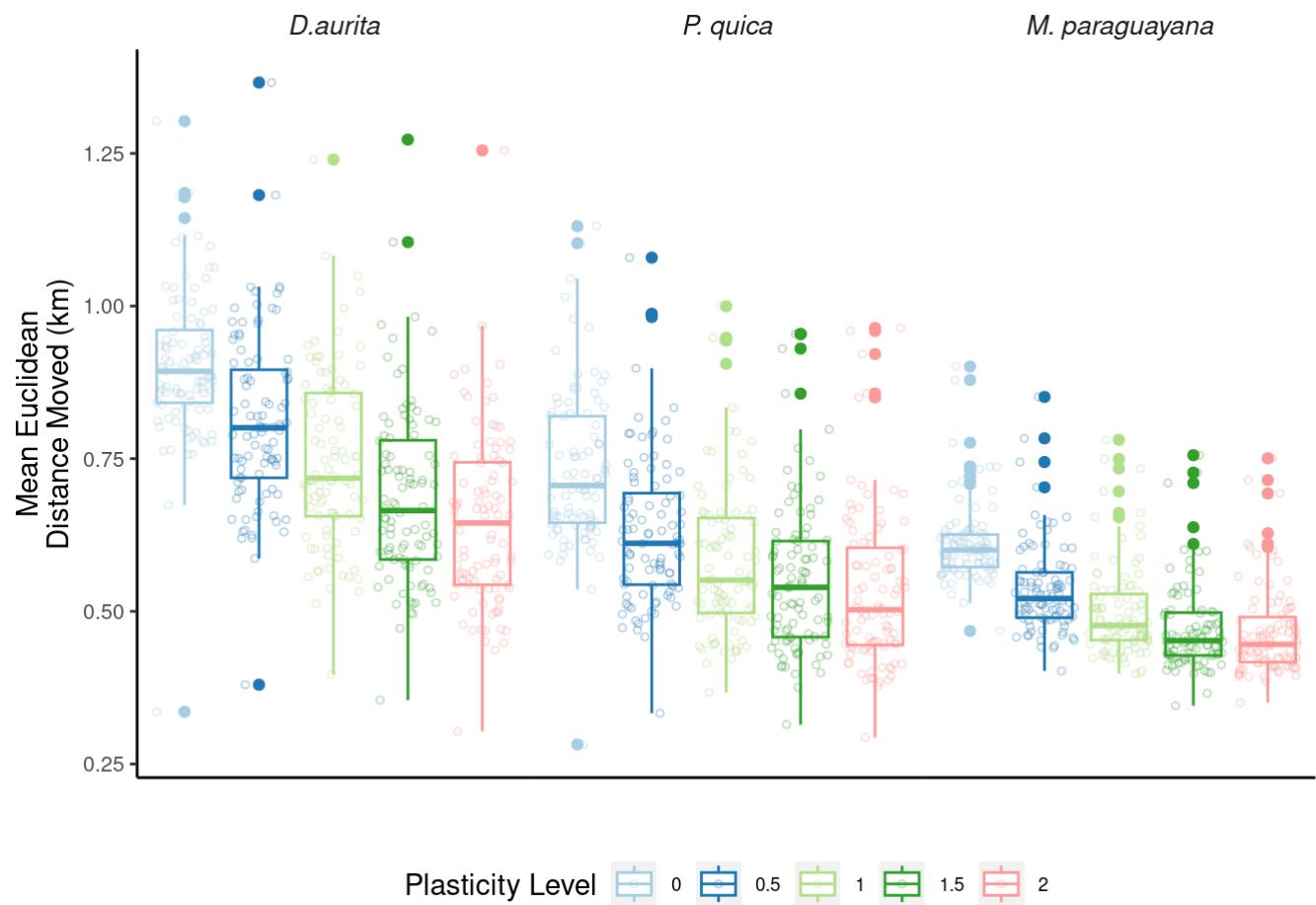


```

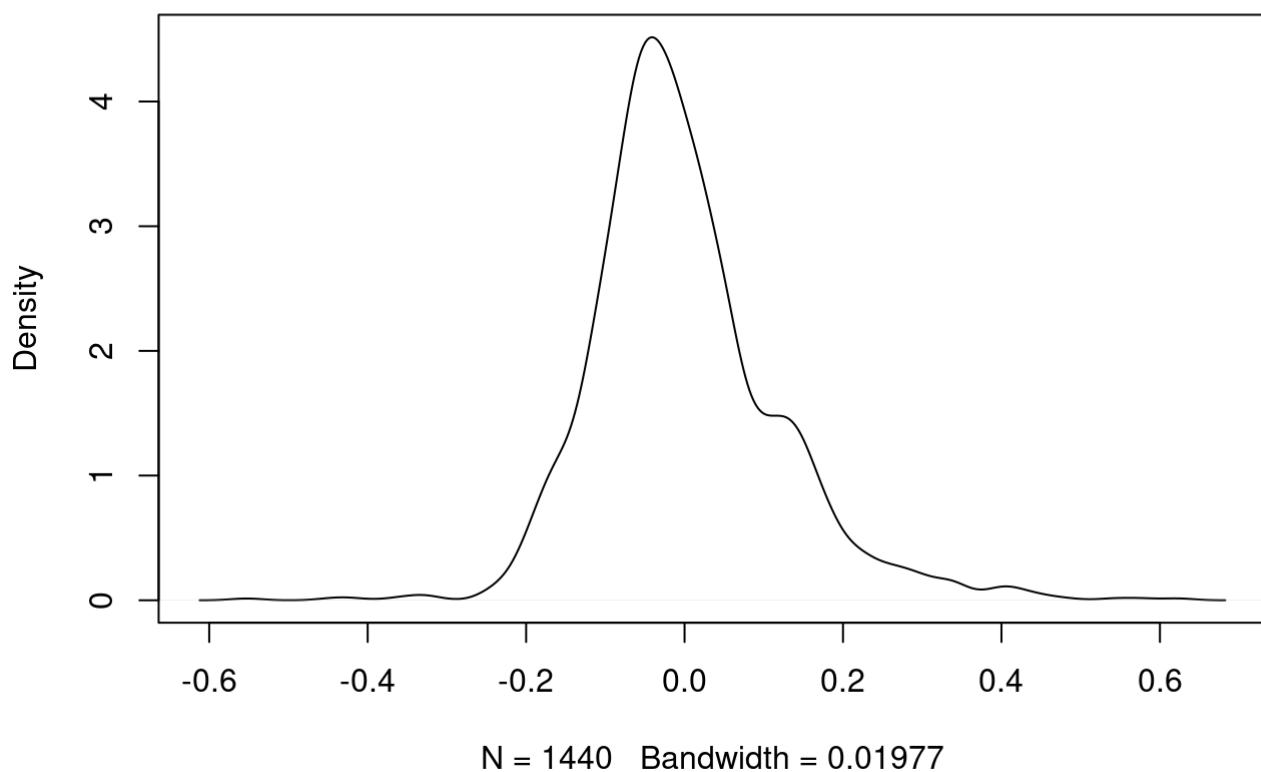
## 
## Call:
## glm(formula = meandt ~ behaviour + specie + behaviour * specie,
##      family = "gaussian", data = data_means)
## 
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max 
## -1.58563 -0.19023  0.01084  0.19188  1.11817 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 3.30222  0.02988 110.521 < 2e-16 ***
## behaviour   -0.75903  0.02440 -31.113 < 2e-16 ***
## speciePQ    -1.04114  0.04225 -24.640 < 2e-16 ***
## specieMP    -1.72273  0.04225 -40.770 < 2e-16 ***
## behaviour:speciePQ 0.27536  0.03450   7.981 2.94e-15 ***
## behaviour:specieMP 0.49088  0.03450  14.228 < 2e-16 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 0.1428368)
## 
## Null deviance: 787.87 on 1439 degrees of freedom
## Residual deviance: 204.83 on 1434 degrees of freedom
## AIC: 1292.2
## 
## Number of Fisher Scoring iterations: 2

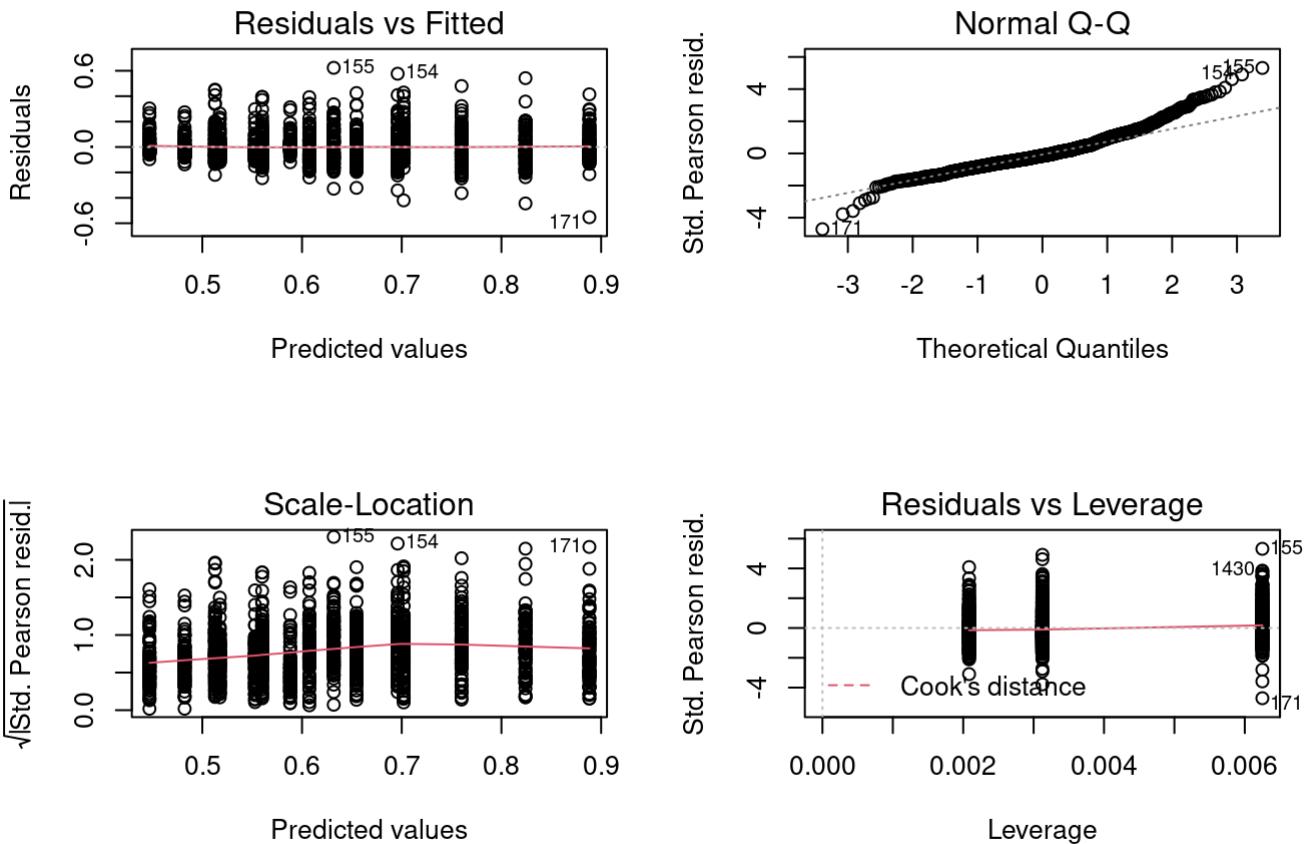
```

Euclidean Distance



density.default(x = resid(m_Id, type = "response"))





```

## 
## Call:
## glm(formula = meanld ~ behaviour + specie + behaviour * specie,
##      family = "gaussian", data = data_means)
## 
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max 
## -0.55272 -0.07120 -0.01687  0.05485  0.62311 
## 
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             0.888210  0.009295 95.556 < 2e-16 ***
## behaviour            -0.128199  0.007589 -16.892 < 2e-16 ***
## speciePQ              -0.186266  0.013145 -14.170 < 2e-16 ***
## specieMP              -0.300078  0.013145 -22.828 < 2e-16 ***
## behaviour:speciePQ   0.033592  0.010733   3.130  0.00179 ** 
## behaviour:specieMP   0.057625  0.010733   5.369 9.23e-08 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 0.01382406)
## 
## Null deviance: 41.536 on 1439 degrees of freedom
## Residual deviance: 19.824 on 1434 degrees of freedom
## AIC: -2070.6
## 
## Number of Fisher Scoring iterations: 2

```

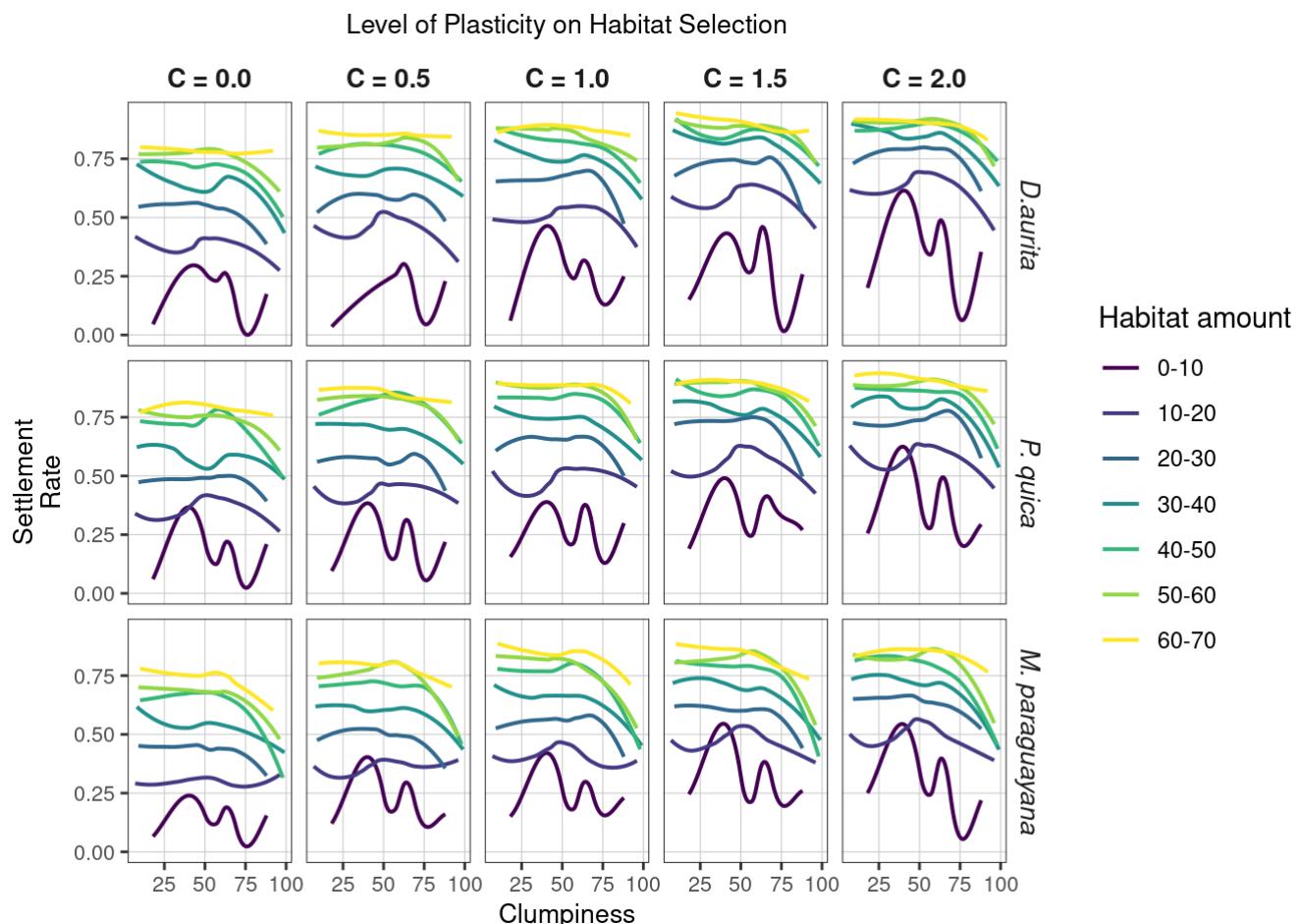
Analysis of landscape effects for all variables

1. To investigate the interaction effects between habitat amount and landscape, I generate scatterplots for each variable, and a GAM model to adjust the relationship:

Settlement Rate

For this, I summarized for each run with 1000 individuals, how many settled. The simulation ends only when all individuals settled or died, so the number of remaining individual per run is the number of settlement. Then I calculated a rate.

*For the manuscript figure, to avoid too many information on the plot, we will keep only the plots for $C = 0.0$ and $C = 2.0$ and reduce the number of lines, and keep only the values of fragmentation threshold, such as 0-10, 20-30, 40-50, 50-70.



And then, I perfomed a GAM model:

GAM MODEL

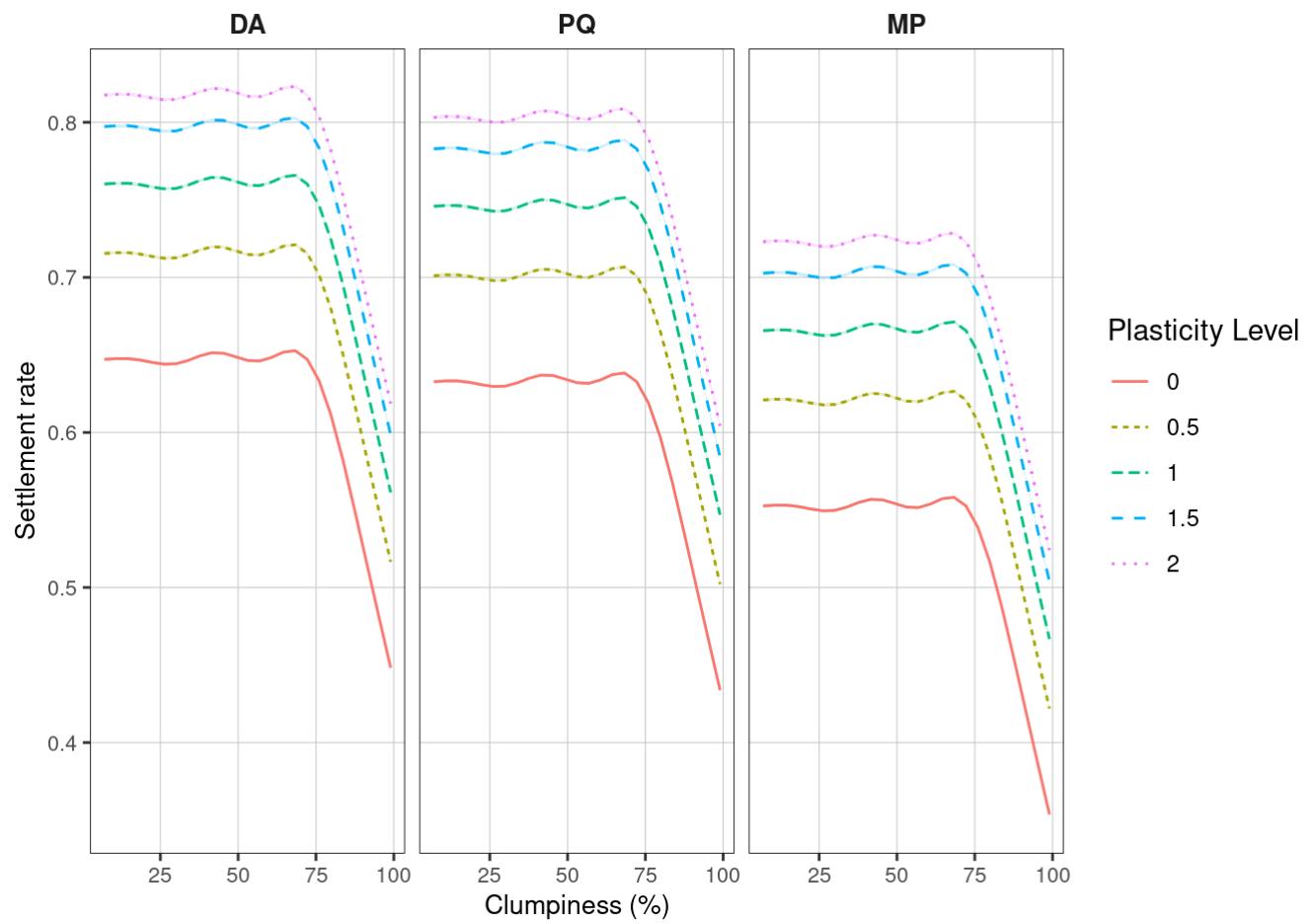
```
library(tidymv)
```

```
## tidymv will be deprecated. Users are recommended
##      to check out the in-progress replacement tidygam
##      (https://github.com/stefanocoretta/tidygam).
```

```
gam_set <- gam(srate~s(clumpiness)+s(hab_amount)+behaviour+specie,data=data_land, family=gaussian, select=T) # Fits the gam
summary(gam_set)
```

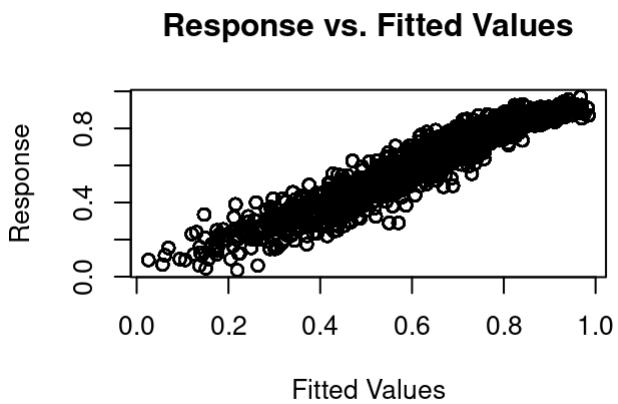
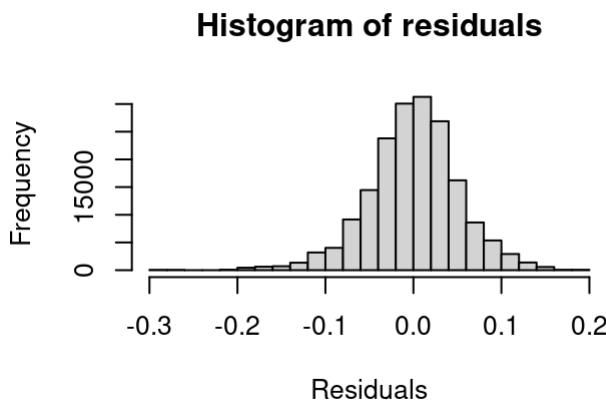
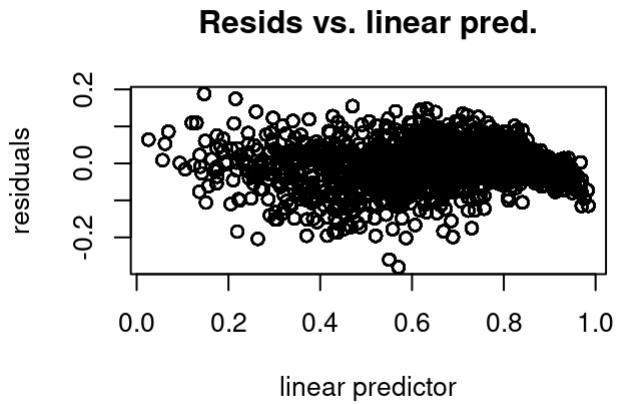
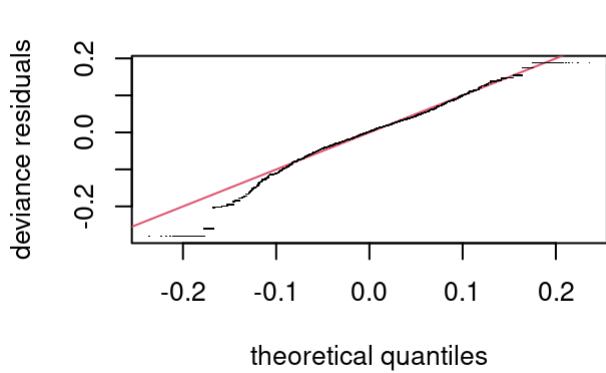
```
## 
## Family: gaussian
## Link function: identity
##
## Formula:
## srate ~ s(clumpiness) + s(hab_amount) + behaviour + specie
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.6256184 0.0003404 1838.1   <2e-16 ***
## behaviour0.5 0.0683338 0.0004094   166.9   <2e-16 ***
## behaviour1   0.1130796 0.0004023   281.1   <2e-16 ***
## behaviour1.5 0.1501398 0.0003970   378.2   <2e-16 ***
## behaviour2   0.1704334 0.0003945   432.1   <2e-16 ***
## speciePQ     -0.0143958 0.0002920   -49.3   <2e-16 ***
## specieMP     -0.0945638 0.0003022   -313.0   <2e-16 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df      F p-value    
## s(clumpiness) 8.821      9 17334   <2e-16 ***
## s(hab_amount) 9.000      9 162805   <2e-16 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.911  Deviance explained = 91.1%
## GCV = 0.0026969  Scale est. = 0.0026965 n = 181449
```

```
#plot(gam_set)
```



- GAM check:

```
par(mfrow=c(2,2))
gam.check(gam_set)
```



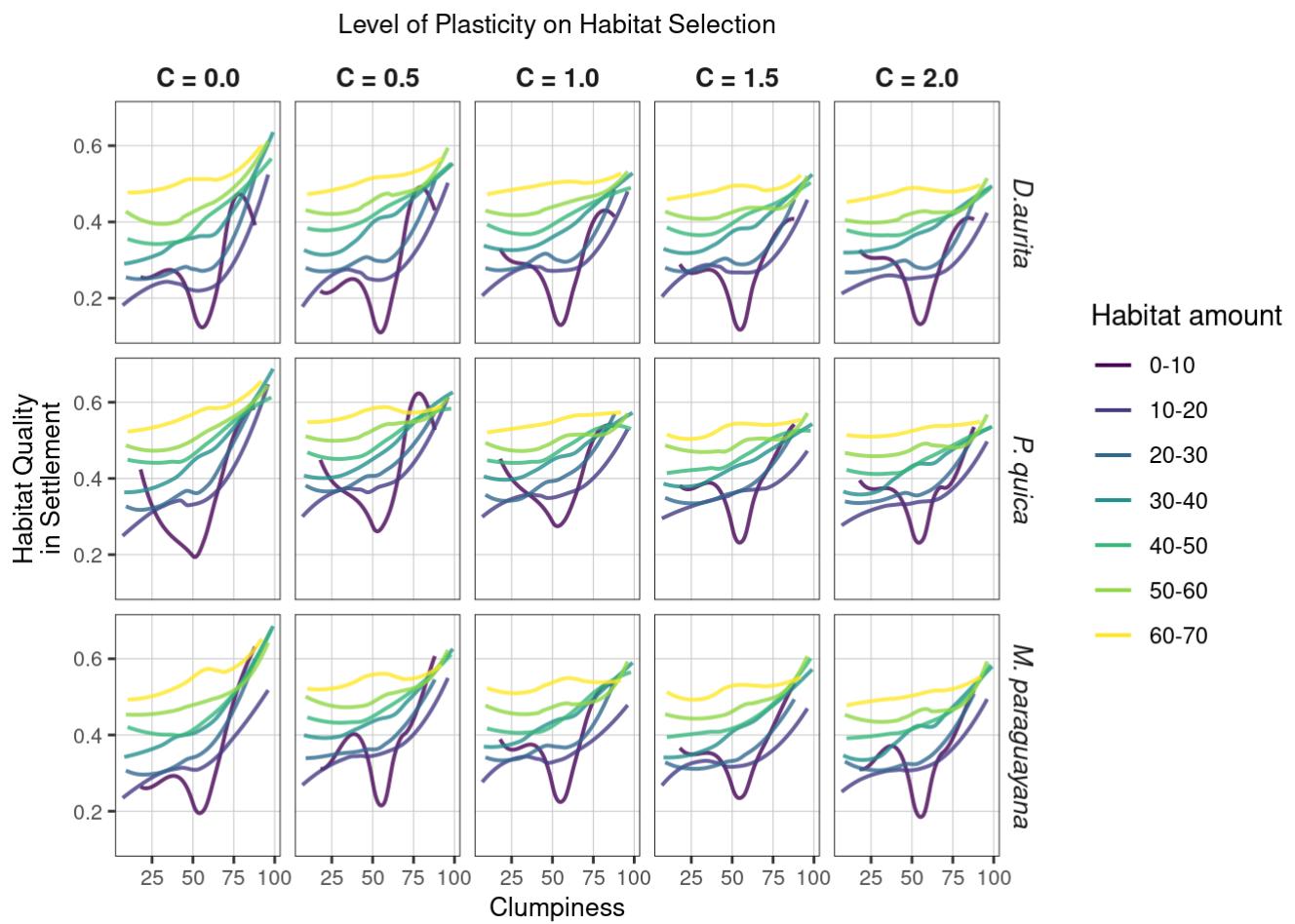
```

## 
## Method: GCV  Optimizer: magic
## Smoothing parameter selection converged after 28 iterations.
## The RMS GCV score gradient at convergence was 4.248702e-08 .
## The Hessian was positive definite.
## Model rank =  25 / 25
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##           k'  edf k-index p-value
## s(clumpiness) 9.00 8.82    0.79 <2e-16 ***
## s(hab_amount) 9.00 9.00    0.85 <2e-16 ***
## ...
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

For the others variables, I did in two different ways: 1. I didn't calculate the mean again, I used all points for all individuals in the plot, and categorize habitat amount to generate smooth lines that helped to see the pattern. 2. I calculated the means and plot the points. And then, I perfomed a GAM for each variable with all data, not the means.

Habitat Quality



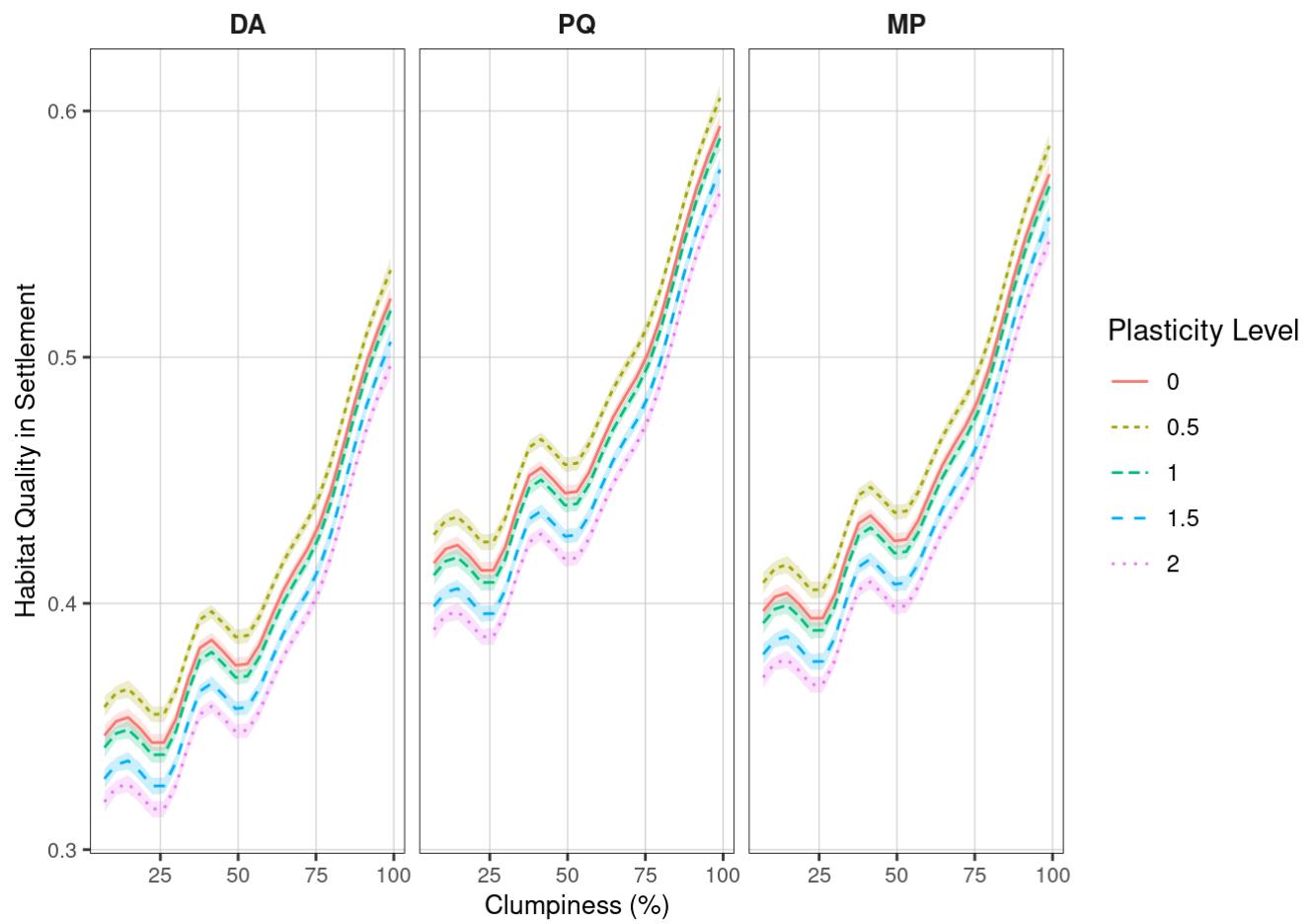
GAM MODEL

```
gam_q <- gam(Hab_quality_area~s(clumpiness)+s(hab_amount)+behaviour+specie,data=data,
family=gaussian, select=T) # Fits the gam
#plot(gam_q)
summary(gam_q)
```

```

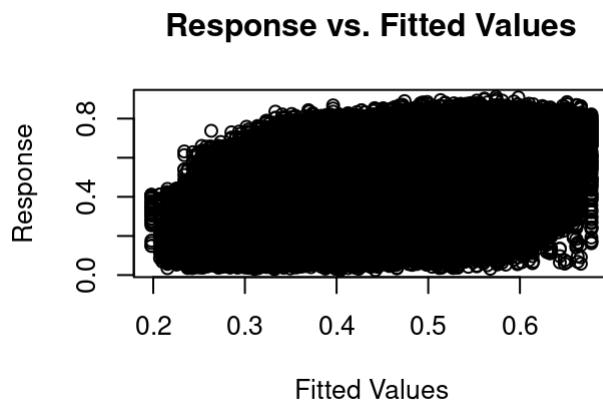
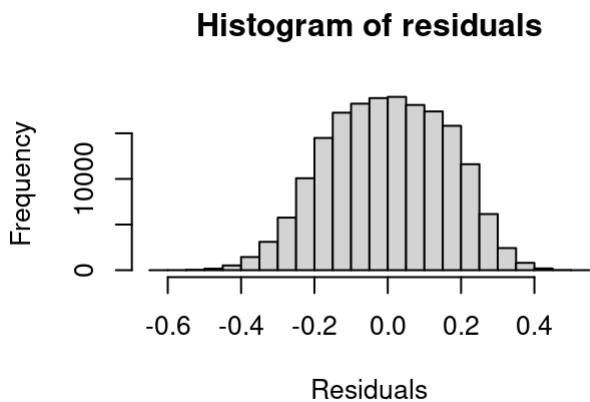
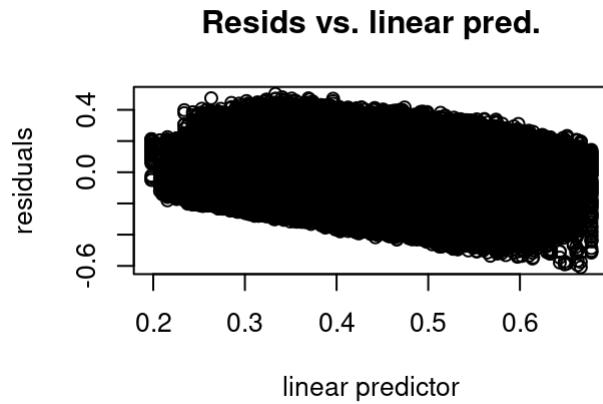
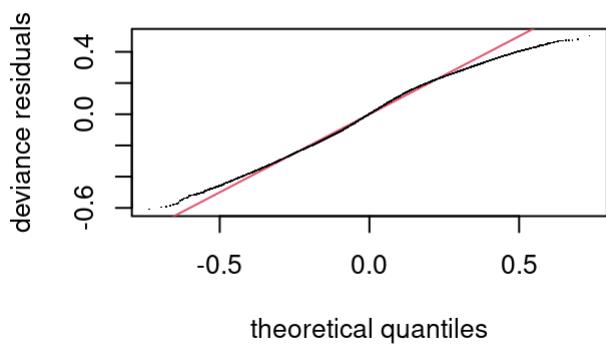
## 
## Family: gaussian
## Link function: identity
##
## Formula:
## Hab_quality_area ~ s(clumpiness) + s(hab_amount) + behaviour +
##   specie
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 0.4016233 0.0010598 378.965 < 2e-16 ***
## behaviour0.5 0.0115326 0.0012747  9.047 < 2e-16 ***
## behaviour1   -0.0049380 0.0012526 -3.942 8.08e-05 ***
## behaviour1.5 -0.0175998 0.0012363 -14.236 < 2e-16 ***
## behaviour2   -0.0269496 0.0012283 -21.941 < 2e-16 ***
## speciePQ      0.0699691 0.0009093  76.945 < 2e-16 ***
## specieMP      0.0505287 0.0009409  53.704 < 2e-16 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df    F p-value    
## s(clumpiness) 8.925     9 1752 <2e-16 ***
## s(hab_amount)  8.300     9 3472 <2e-16 ***
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.257  Deviance explained = 25.7%
## GCV = 0.026149  Scale est. = 0.026146 n = 181449

```



- GAM check:

```
par(mfrow=c(2,2))
gam.check(gam_q)
```

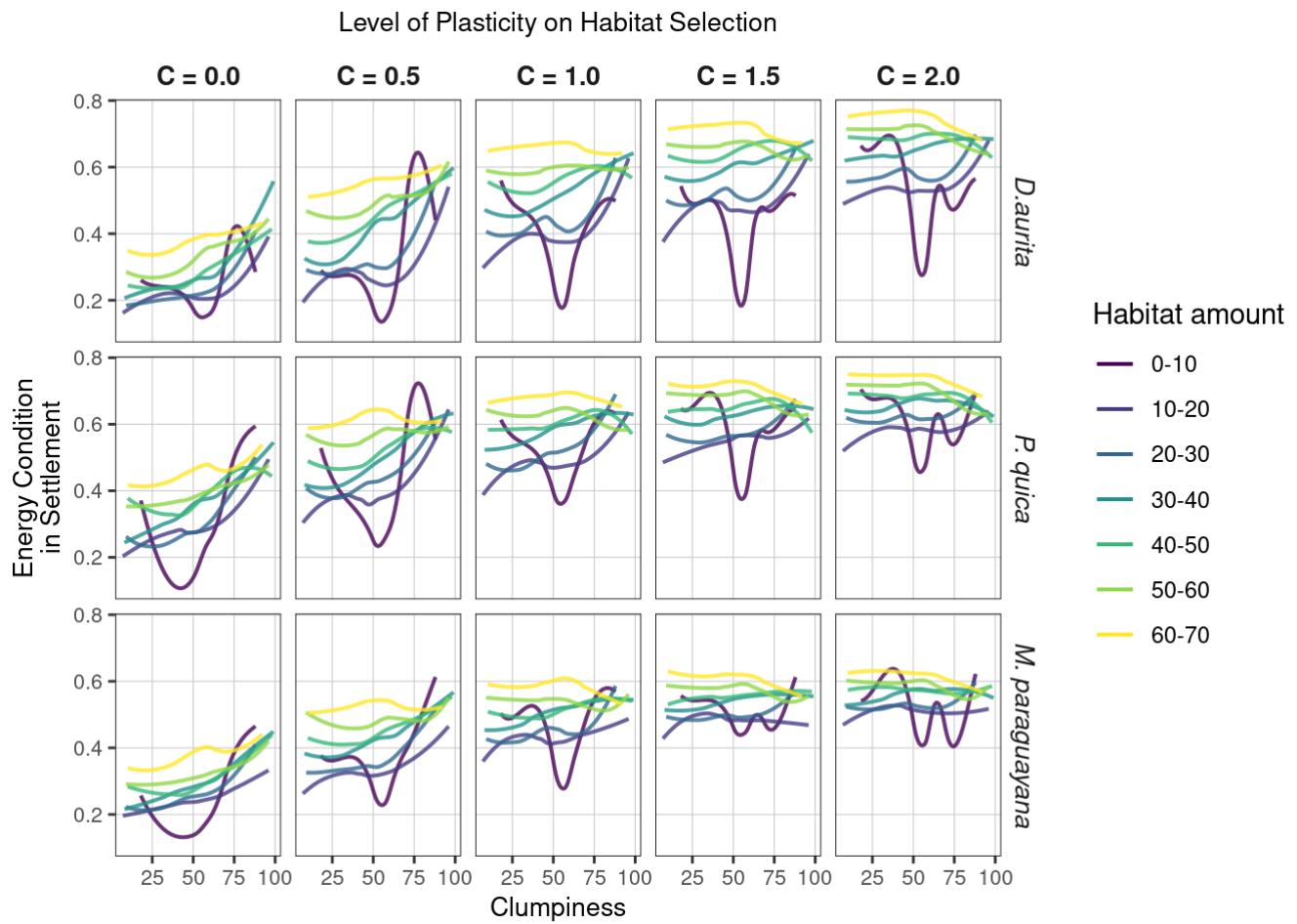


```

## 
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 10 iterations.
## The RMS GCV score gradient at convergence was 4.621066e-08 .
## The Hessian was not positive definite.
## Model rank =  25 / 25
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##           k'  edf k-index p-value
## s(clumpiness) 9.00 8.93    0.99    0.23
## s(hab_amount) 9.00 8.30    0.98    0.14

```

Energetic Conditions



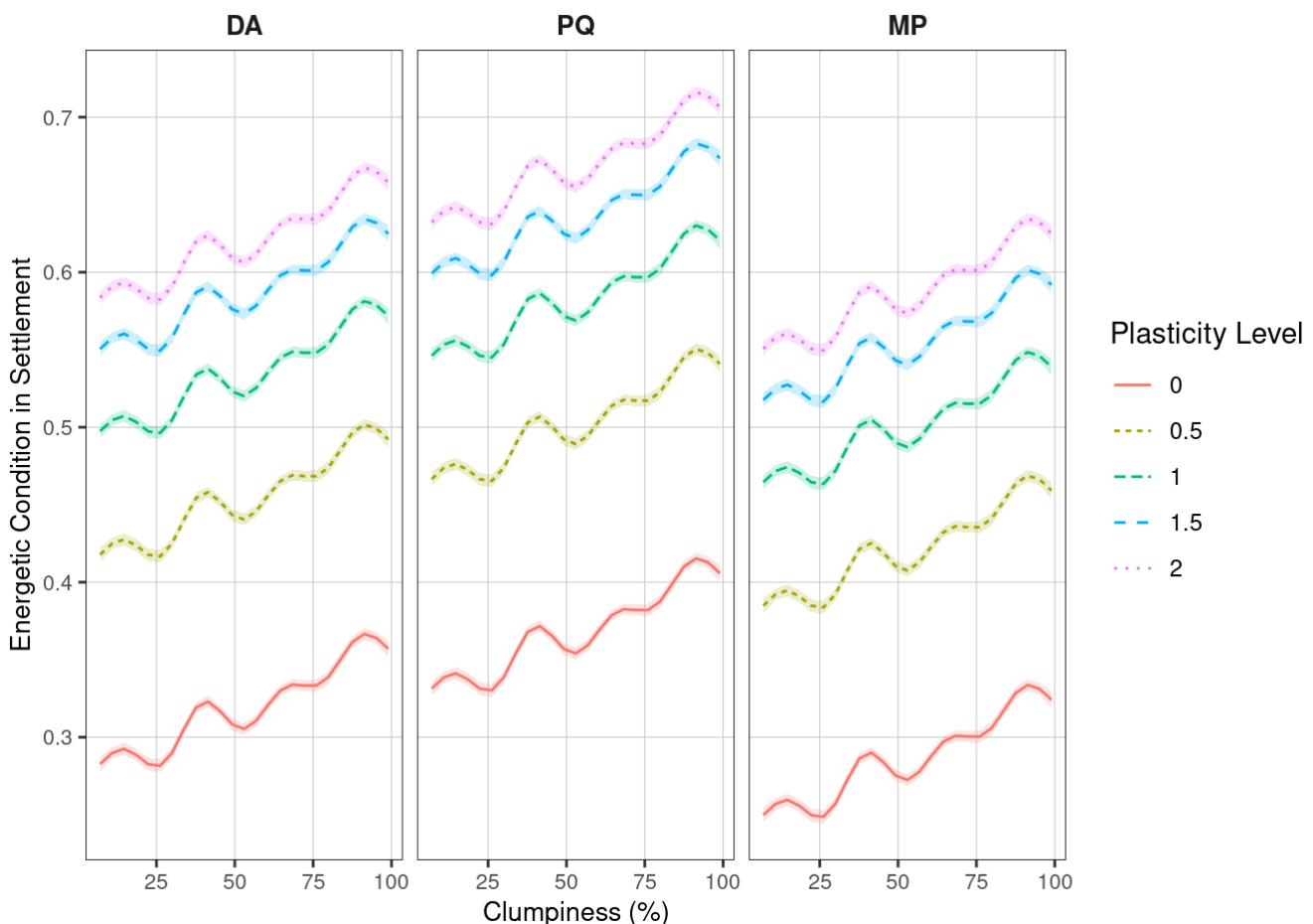
GAM MODEL

```
gam_en <- gam(final_En~s(clumpiness)+s(hab_amount)+behaviour+specie, data=data, family=gaussian, select=T) # Fits the gam
#plot(gam_en)
summary(gam_en)
```

```

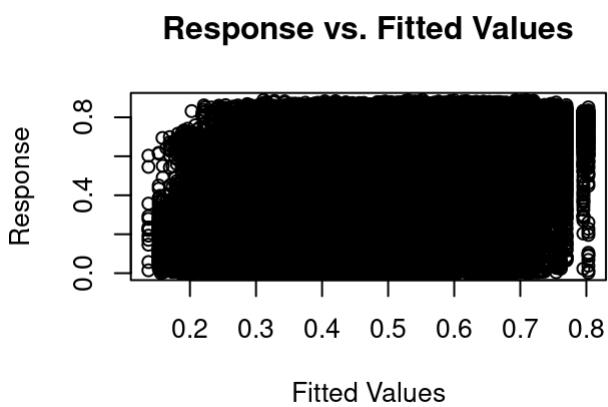
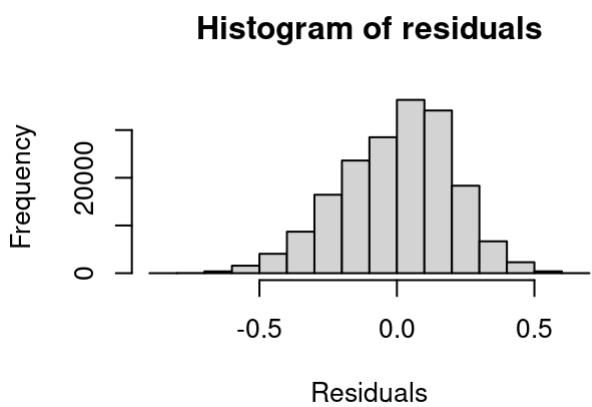
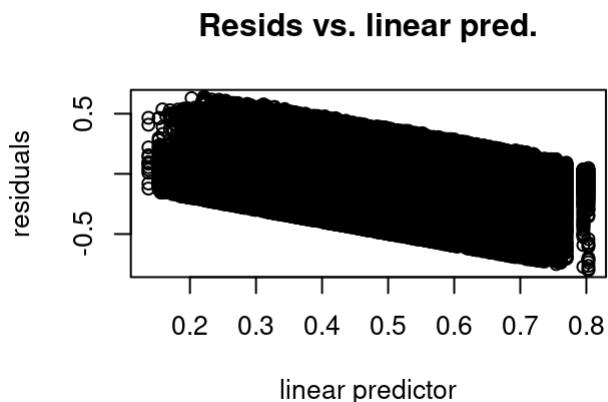
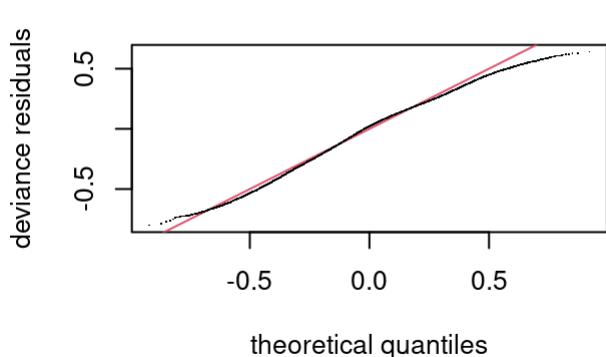
## 
## Family: gaussian
## Link function: identity
##
## Formula:
## final_En ~ s(clumpiness) + s(hab_amount) + behaviour + specie
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.320455  0.001327 241.58 <2e-16 ***
## behaviour0.5 0.135112  0.001596  84.68 <2e-16 ***
## behaviour1   0.214758  0.001568 136.98 <2e-16 ***
## behaviour1.5 0.267803  0.001547 173.06 <2e-16 ***
## behaviour2   0.300894  0.001537 195.72 <2e-16 ***
## speciePQ     0.048720  0.001138  42.80 <2e-16 ***
## specieMP     -0.032908 0.001178 -27.95 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df      F p-value
## s(clumpiness) 7.992      9 282.5 <2e-16 ***
## s(hab_amount)  7.932      9 1887.2 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.283  Deviance explained = 28.3%
## GCV = 0.040967  Scale est. = 0.040962 n = 181449

```



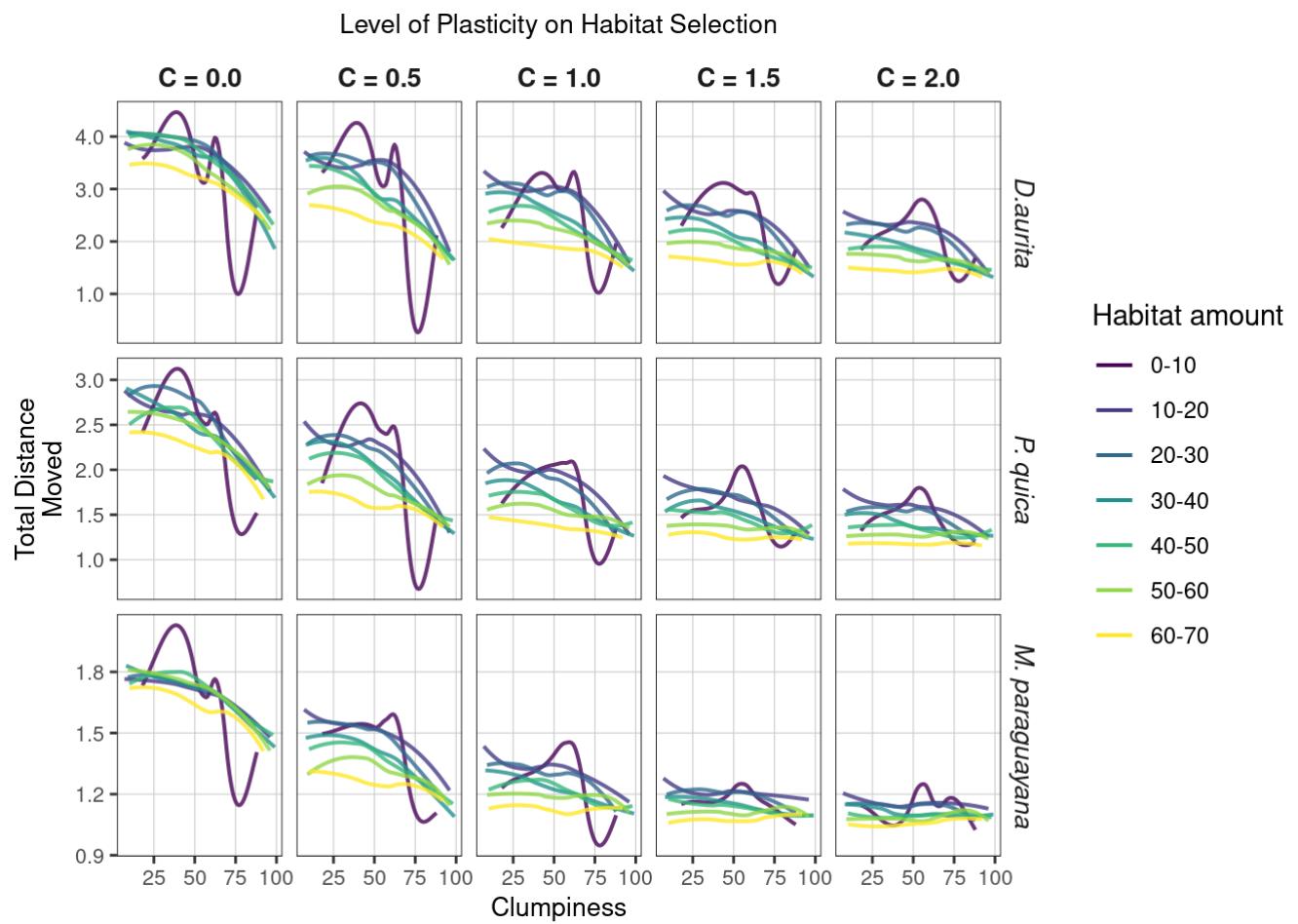
- GAM check:

```
par(mfrow=c(2,2))
gam.check(gam_en)
```



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 12 iterations.
## The RMS GCV score gradient at convergence was 4.240772e-08 .
## The Hessian was positive definite.
## Model rank =  25 / 25
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'  edf k-index p-value
## s(clumpiness) 9.00 7.99    0.99   0.31
## s(hab_amount) 9.00 7.93    1.00   0.47
```

Distance Total



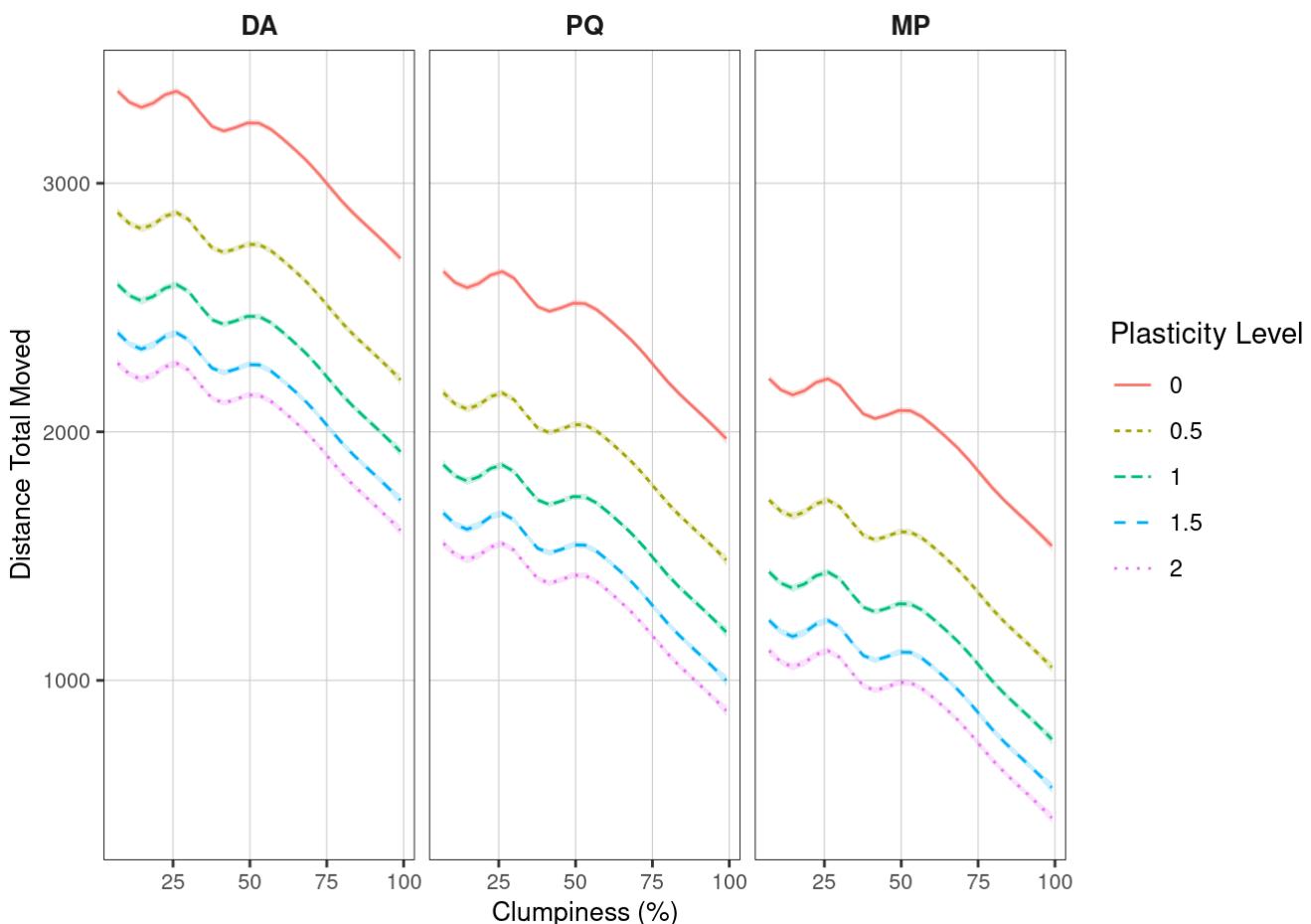
GAM MODEL

```
gam_dt <- gam(total_dist~s(clumpiness)+s(hab_amount)+behaviour+specie,data=data, family=gaussian, select=T) # Fits the gam
#plot(gam_dt)
summary(gam_dt)
```

```

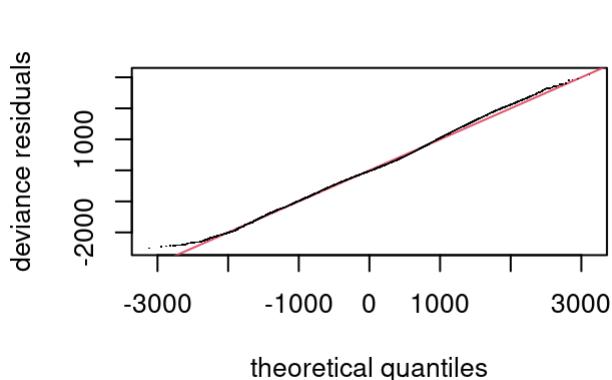
## 
## Family: gaussian
## Link function: identity
##
## Formula:
## total_dist ~ s(clumpiness) + s(hab_amount) + behaviour + specie
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3142.697    4.491  699.71 <2e-16 ***
## behaviour0.5 -488.321   5.402  -90.39 <2e-16 ***
## behaviour1    -777.796   5.308 -146.52 <2e-16 ***
## behaviour1.5  -972.262   5.239 -185.56 <2e-16 ***
## behaviour2    -1094.143   5.206 -210.19 <2e-16 ***
## speciePQ      -725.286   3.854 -188.20 <2e-16 ***
## specieMP      -1156.693   3.987 -290.09 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df F p-value
## s(clumpiness) 8.947     9 1433 <2e-16 ***
## s(hab_amount) 8.938     9 1215 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.479  Deviance explained = 47.9%
## GCV = 4.6965e+05  Scale est. = 4.6959e+05 n = 181449

```

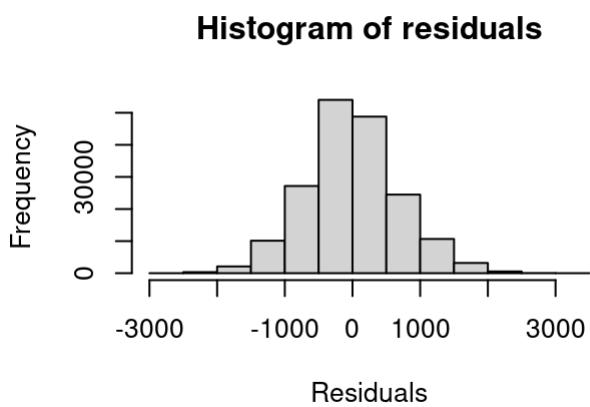
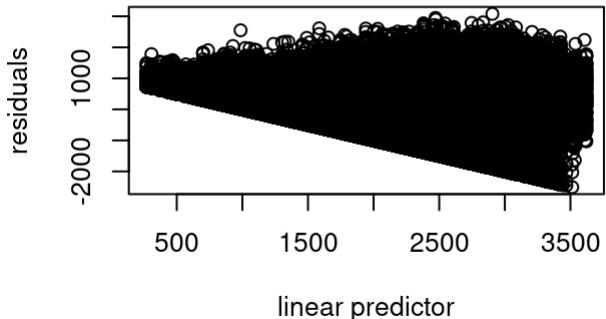


- GAM check:

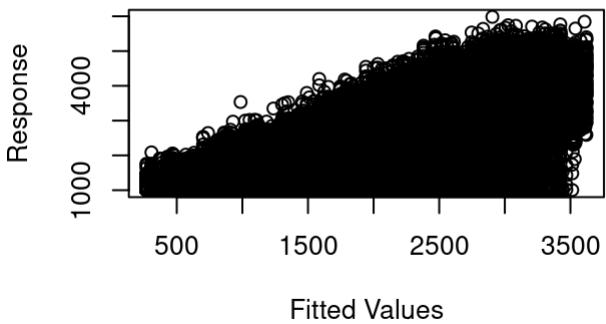
```
par(mfrow=c(2,2))
gam.check(gam_dt)
```



Resids vs. linear pred.

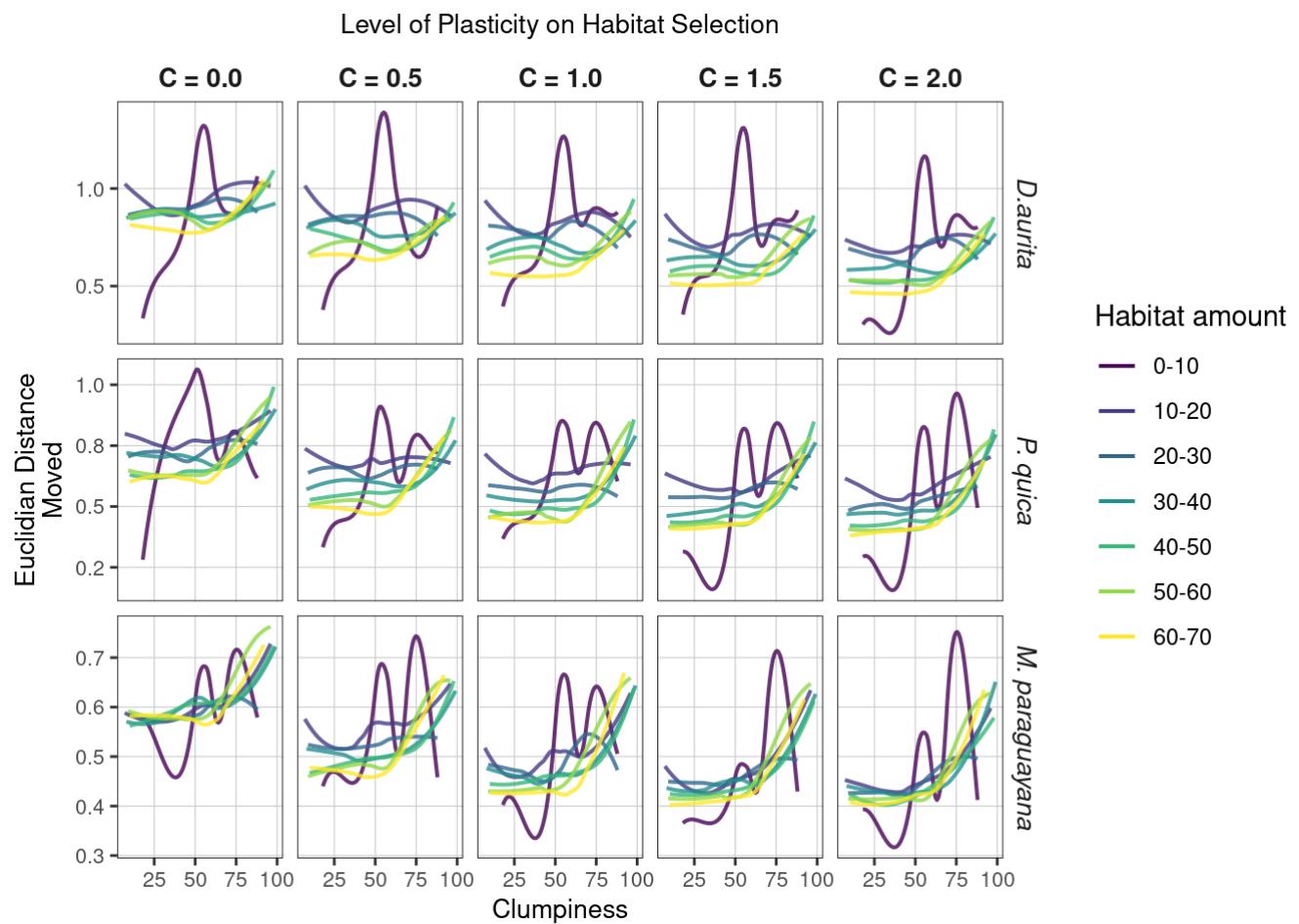


Response vs. Fitted Values



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 31 iterations.
## The RMS GCV score gradient at convergence was 0.05339854 .
## The Hessian was positive definite.
## Model rank =  25 / 25
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'    edf k-index p-value
## s(clumpiness) 9.00 8.95    0.98   0.04 *
## s(hab_amount) 9.00 8.94    0.97   0.02 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Euclidean Distance



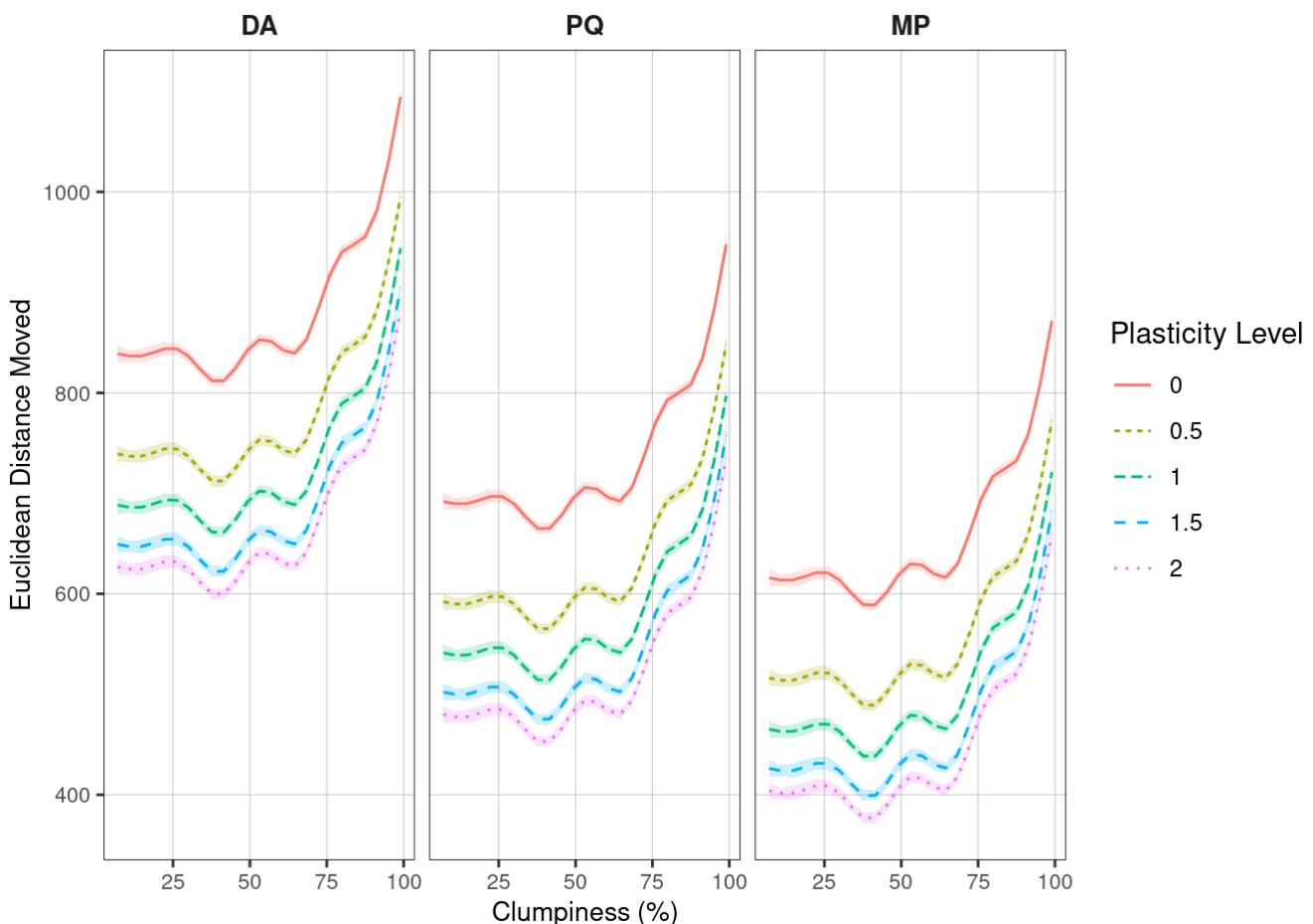
GAM MODEL

```
gam_ld <- gam(linear_dist~s(clumpiness)+s(hab_amount)+behaviour+specie,data=data, family=gaussian, select=T) # Fits the gam
#plot(gam_ld)
summary(gam_ld)
```

```

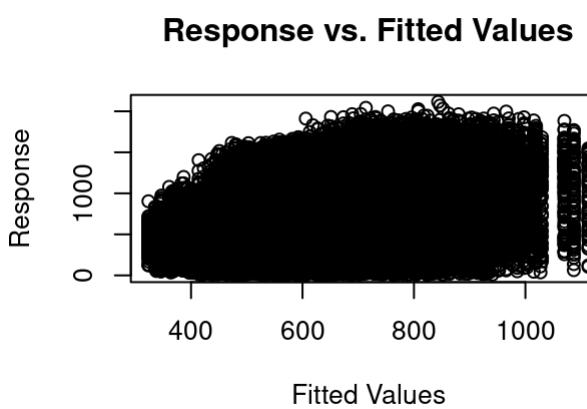
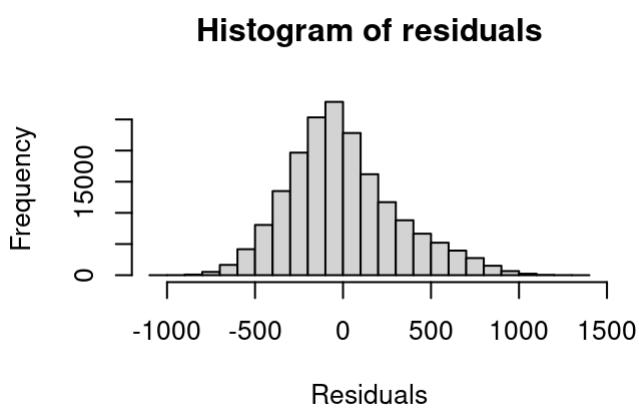
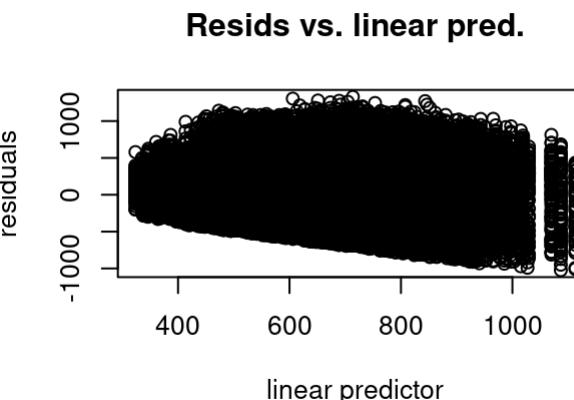
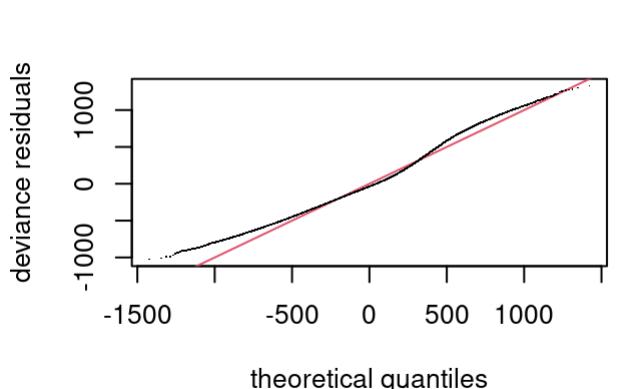
## 
## Family: gaussian
## Link function: identity
##
## Formula:
## linear_dist ~ s(clumpiness) + s(hab_amount) + behaviour + specie
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 869.365   2.051   423.95 <2e-16 ***
## behaviour0.5 -99.747   2.467  -40.44 <2e-16 ***
## behaviour1    -150.673  2.424  -62.17 <2e-16 ***
## behaviour1.5  -189.710  2.392  -79.31 <2e-16 ***
## behaviour2    -212.050  2.377  -89.22 <2e-16 ***
## speciePQ      -147.043  1.759  -83.57 <2e-16 ***
## specieMP      -223.029  1.820  -122.51 <2e-16 ***
## ... 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df     F p-value    
## s(clumpiness) 8.941     9 770.6 <2e-16 ***
## s(hab_amount)  8.940     9 430.4 <2e-16 *** 
## ... 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.166  Deviance explained = 16.7%
## GCV =  97899  Scale est. = 97885 n = 181449

```



- GAM check:

```
par(mfrow=c(2,2))
gam.check(gam_ld)
```



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 33 iterations.
## The RMS GCV score gradient at convergence was 0.01222095 .
## The Hessian was positive definite.
## Model rank =  25 / 25
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##          k'  edf k-index p-value
## s(clumpiness) 9.00 8.94    0.99    0.32
## s(hab_amount) 9.00 8.94    1.03    0.97
```

Analysis of landscape effects on habitat quality

1. Categorize habitat amount and clumpiness to generate density plots:

```

data <- data %>%
  mutate(cat_land=cut(hab_amount, breaks=c(0,10,20,30,40,50,60,70),
                     labels=c("0-10","10-20","20-30","30-40","40-50","50-60","60-70"),
                     include.lowest = TRUE)) %>%
  mutate(cat_frag=cut(clumpiness, breaks=c(0,20,40,60,80,100),
                     labels=c("0-20","20-40","40-60","60-80","80-100"),
                     include.lowest = TRUE))
data$specie= factor(data$specie, levels = c("DA","PQ","MP"))
data$behaviour <- factor(data$behaviour)

```

2. Generate plots:

