Determining the repeatability of turtle behaviour

R code to calculate the among-individual repeatabilty of the turtles behaviours ((active defensive behaviours (Aggression), shell emergence time (Shell), time of initial movement (Start), total time spent moving (Move)).

Packages

```
library(rptR)
library(ggplot2)
library(car)
library(multcomp)
library(lmtest)
library(lme4)
library(Hmisc)
library(writexl)
library(dplyr)
library(lmerTest)
library(optimx)
library(PerformanceAnalytics)
library(effects)
library(ggeffects)
library(splines)
library(glmtoolbox)
library(afex)
library(nloptr)
library(dfoptim)
library(psych)
library(ordinal)
library(ggpubr)
```

Upload the dataset to use

```
Behaviour<-read.csv("C:/Users/sebas/Desktop/Masters Work/Masters Work/Stats/B
in.Shell.600/Combined Repeatability Dataset.600.csv")
```

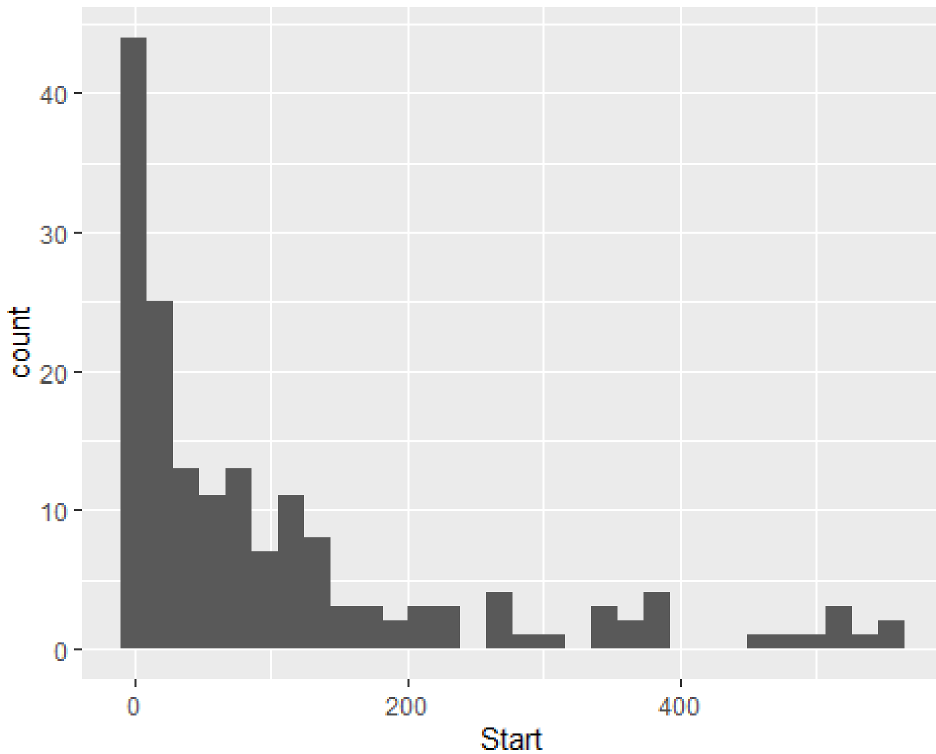
Preparing data

```
Behaviour$Shell <- as.numeric(Behaviour$Shell)
Behaviour$Aggression <- as.numeric(Behaviour$Aggression)
Behaviour$Start <- as.numeric(Behaviour$Start)
Behaviour$Move <- as.numeric(Behaviour$Move)
Behaviour$BIN.Shell <- as.numeric(Behaviour$BIN.Shell)
```

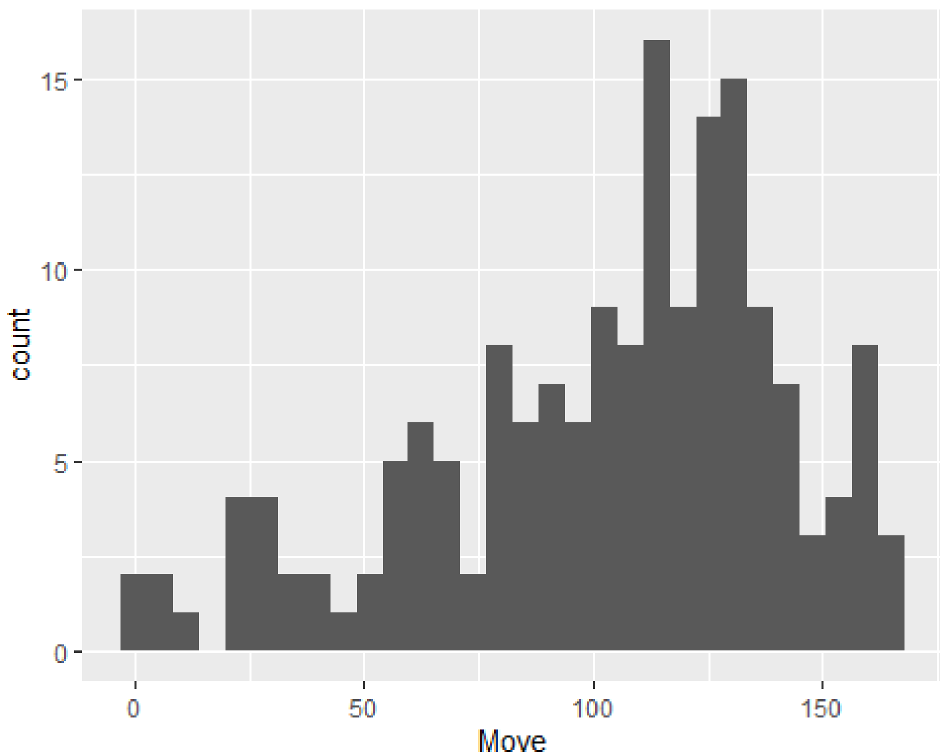
Plot data

I am doing this step to make sure I use the right data distribution in my models.

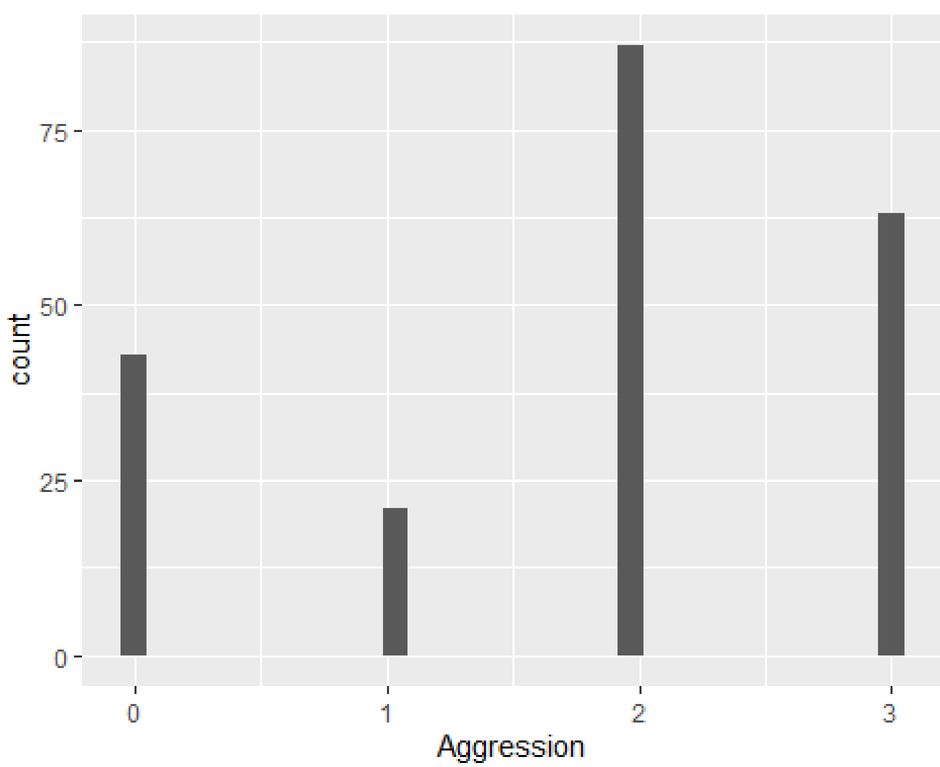
```
ggplot(Behaviour, aes(x=Start)) + geom_histogram()
```



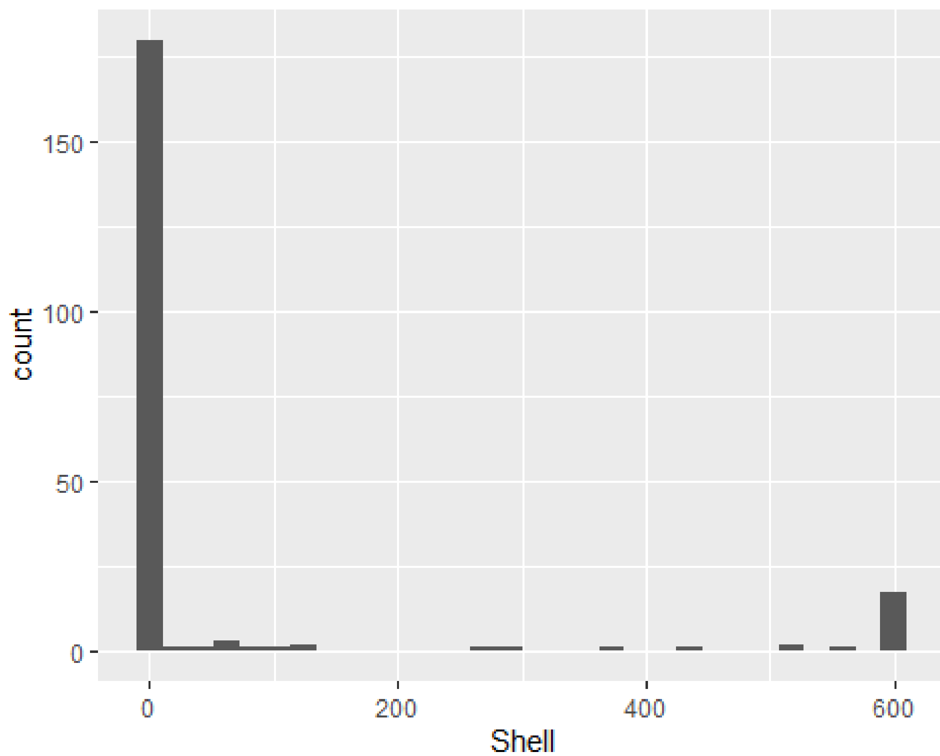

```
ggplot(Behaviour, aes(x=Move)) + geom_histogram()
```




```
ggplot(Behaviour, aes(x=Aggression)) + geom_histogram()
```




```
ggplot(Behaviour, aes(x=Shell)) + geom_histogram()
```



Time of initial movement (Start) has a skewed distribution so I will try log transforming it.

Transform inital time of movement

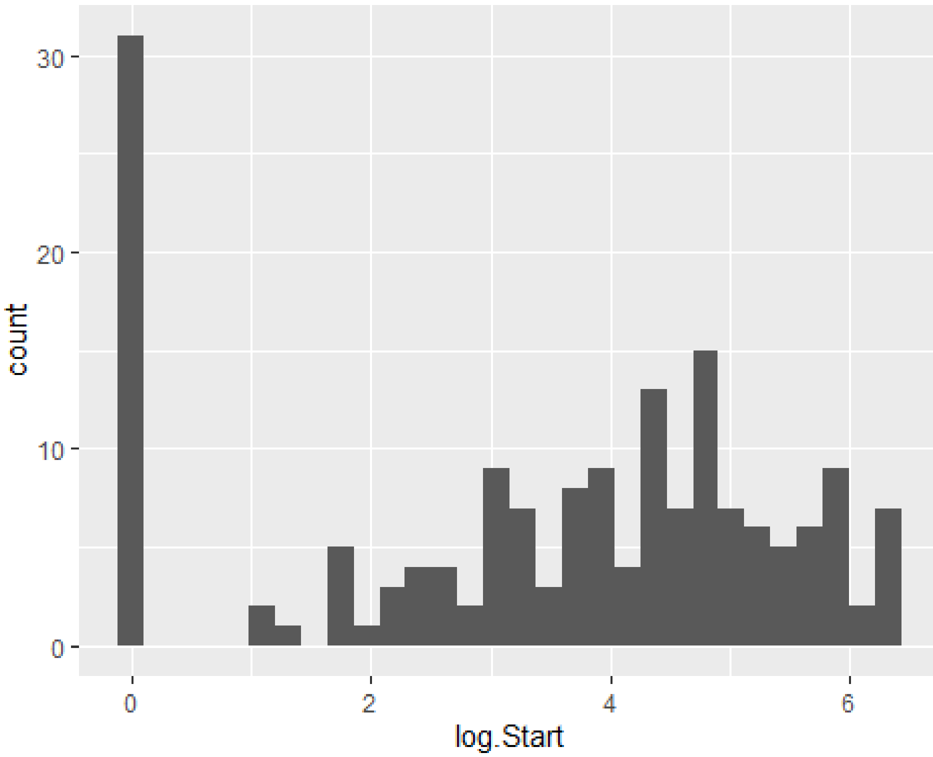
```
Behaviour$log.Start <- log(Behaviour$Start + 1)
```

Prepare log(x+1) transformed version of initial time of movement (log.Start)

```
Behaviour$log.Start <- as.numeric(Behaviour$log.Start)
```

Plot log(x+1) transformed version of initial time of movement

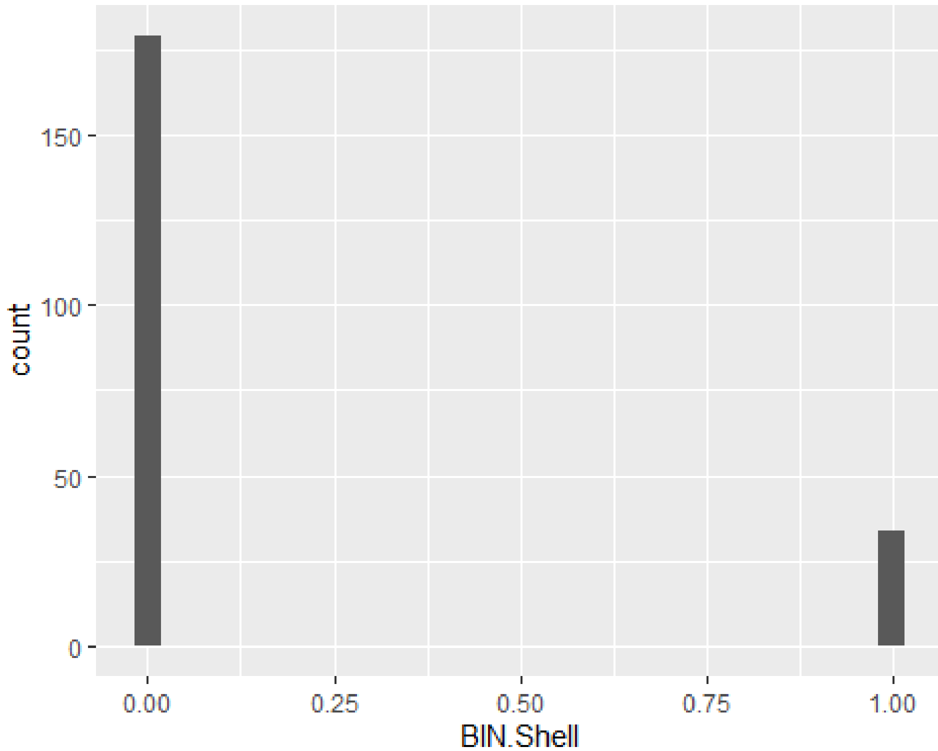
```
ggplot(Behaviour, aes(x=log.Start)) + geom_histogram()
```



The log(x+1) transformed version of time of initial movement is still somewhat skewed, so I will run my analysis on both versions for comparison. Shell emergence time (Shell) is also skewed, so I will plot the binary version of it.

Plot binary shell emergence time (BIN.Shell)

```
ggplot(Behaviour, aes(x=BIN.Shell)) + geom_histogram()
```



Calculation of sources of variance and repeatability estimates

For my study I set the bootstrap value to 1000, but for this R Code, I set the value to 10 to reduce the calculation time for some of the models, especially for the adjusted estimations.This does not affect the value of the repeatability estimates or p-values, but it does affect the values for the standard errors and 95% confidence intervals

Time of initial movement (Start)

Unadjusted

For the unadjusted estimations, I only included turtle ID (Code) as a random effect.

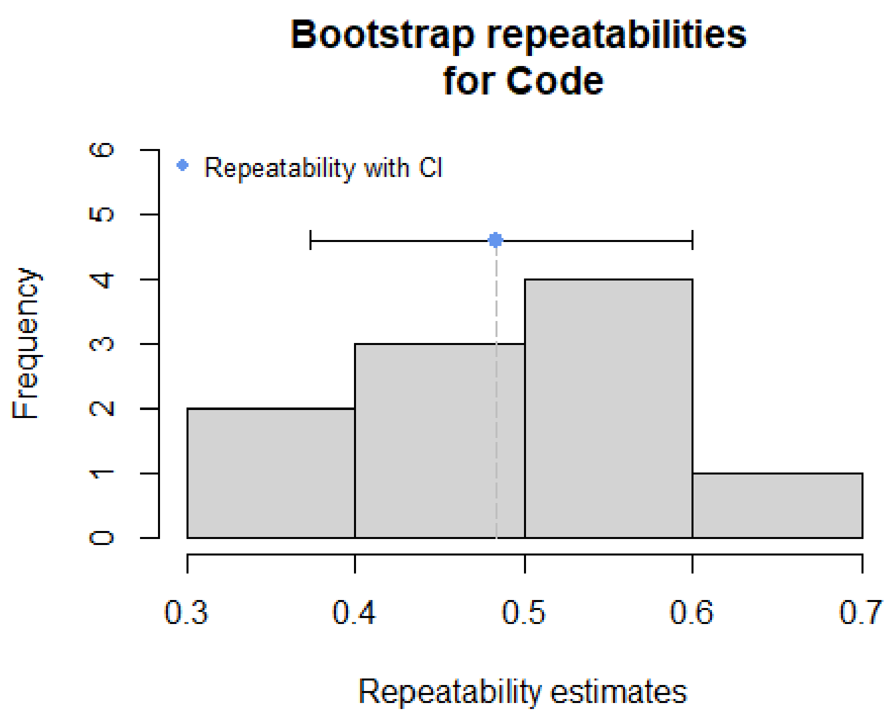
Repeatability

```
rep.Start <- rpt(Start ~ (1 | Code), grname = "Code", data = Behaviour, datat
ype = "Gaussian", nboot = 10, npermut = 0)
```

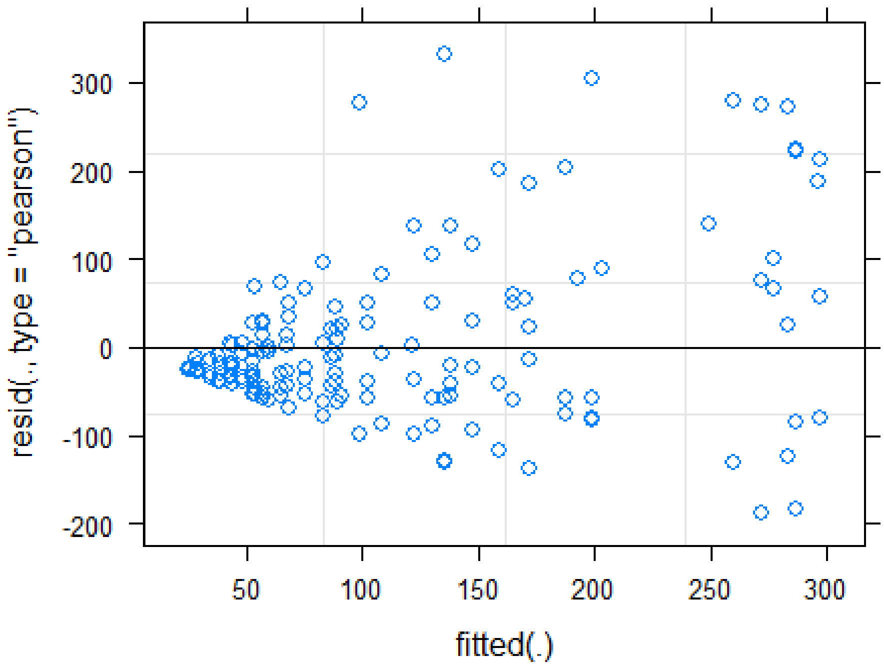
Analysis

```
summary(rep.Start)

##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Start ~ (1 | Code), grname = "Code", data = Behaviour
, datatype = "Gaussian", nboot = 10, npermut = 0)
```

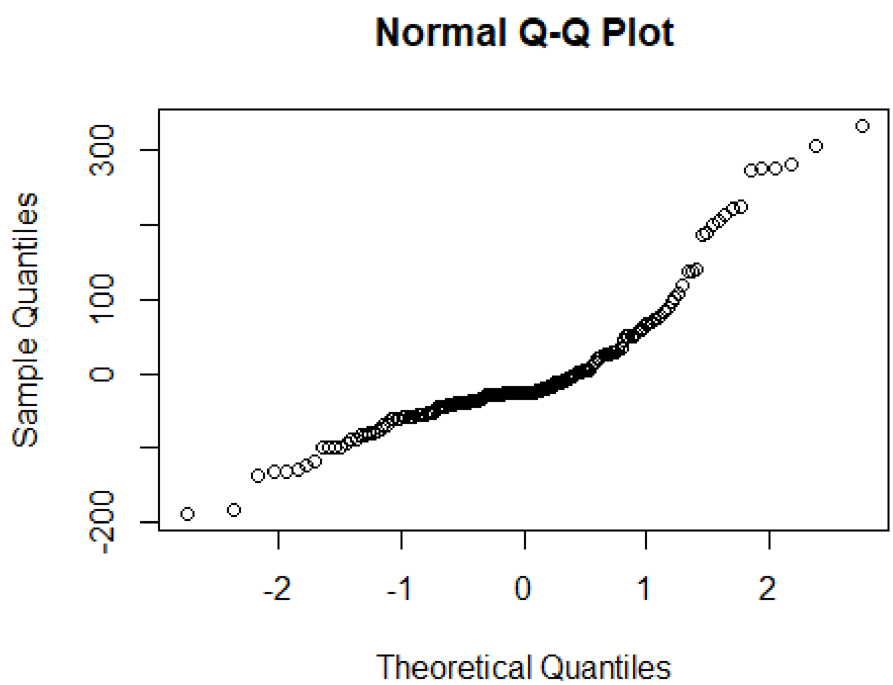
```
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.484 0.0771  0.373  0.599       NA      0
##
## Bootstrapping and Permutation test:
##             N    Mean Median   2.5%  97.5%
## boot       10   0.488  0.498  0.373  0.599
## permut      1      NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -1062.632
## logLik red. model = -1077.491
## D  = 29.7, df = 1, P = 2.5e-08
##
## ----------------------------------------

plot(rep.Start)
```




```
plot(rep.Start$mod)
```




```
qqnorm(resid(rep.Start$mod))
```




```
print(rep.Start)
```

```
##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.484
## SE = 0.077
## CI = [0.373, 0.599]
## P  = 2.5e-08 [LRT]
##      NA [Permutation]
```

Adjusted

For the adjusted estimations, I also included turtle sex (Sex) and day of testing (Day) as fixed effects, and site identity (Site) as a random effect.

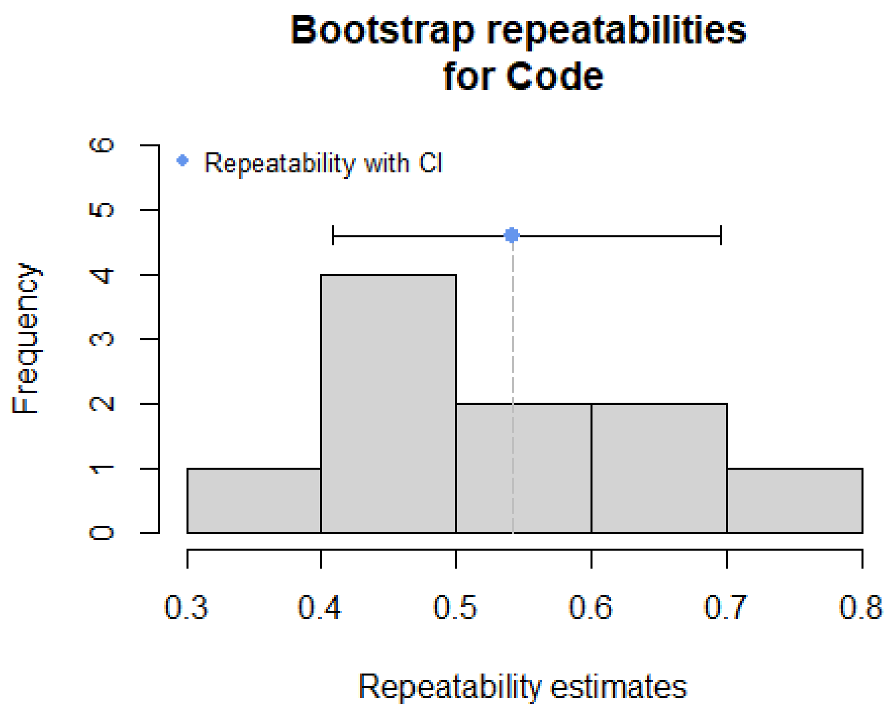
Repeatability

```
rep.fix.Start <- rpt(Start ~ Day + Sex + (1 | Code) + (1 | Site), grname = "C
ode", data = Behaviour, datatype = "Gaussian",
                            nboot = 10, npermut = 0)
```

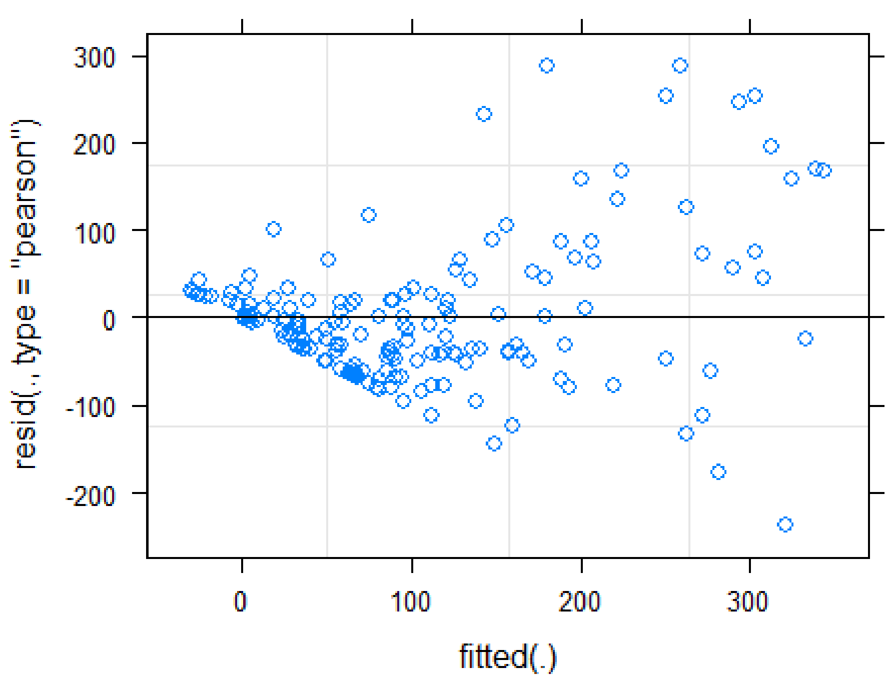
Analysis

```
summary(rep.fix.Start)

##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Start ~ Day + Sex + (1 | Code) + (1 | Site), grname =
"Code", data = Behaviour, datatype = "Gaussian", nboot = 10, npermut = 0)
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.542 0.0971  0.409  0.696       NA      0
##
## Bootstrapping and Permutation test:
##             N    Mean Median   2.5%  97.5%
## boot       10   0.533  0.512  0.409  0.696
## permut      1      NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -1042.618
## logLik red. model = -1060.074
## D  = 34.9, df = 1, P = 1.73e-09
##
## ----------------------------------------
```

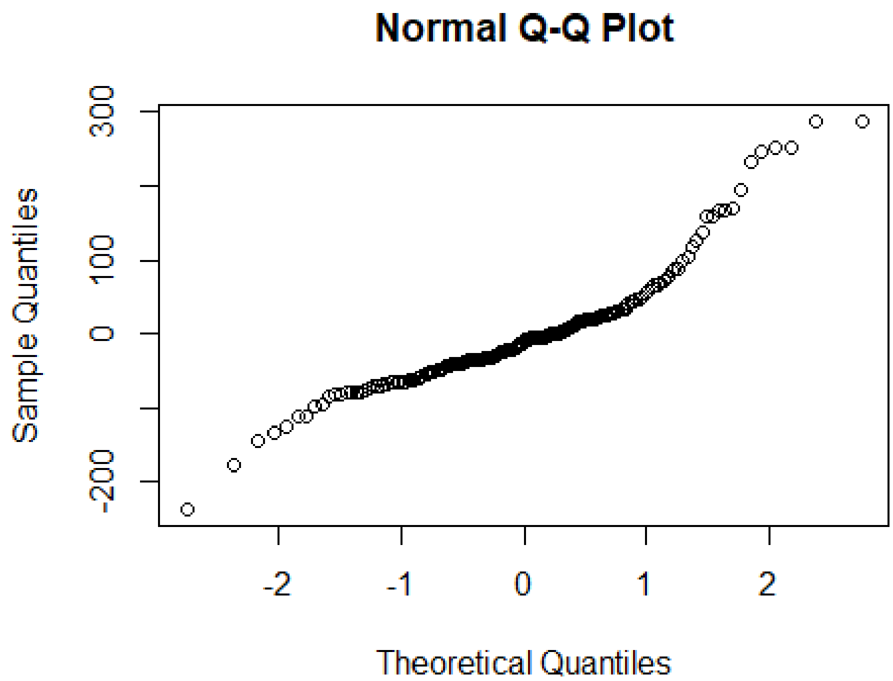
```
plot(rep.fix.Start)
```




```
plot(rep.fix.Start$mod)
```



```
qqnorm(resid(rep.fix.Start$mod))
```




```
print(rep.fix.Start)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.542
## SE = 0.097
## CI = [0.409, 0.696]
## P  = 1.73e-09 [LRT]
##      NA [Permutation]
```

Log(x+1) transformed version ot time of initial movement (log.Start)

Unadjusted

For the unadjusted estimations, I only included turtle ID (Code) as a random effect.

Repeatability

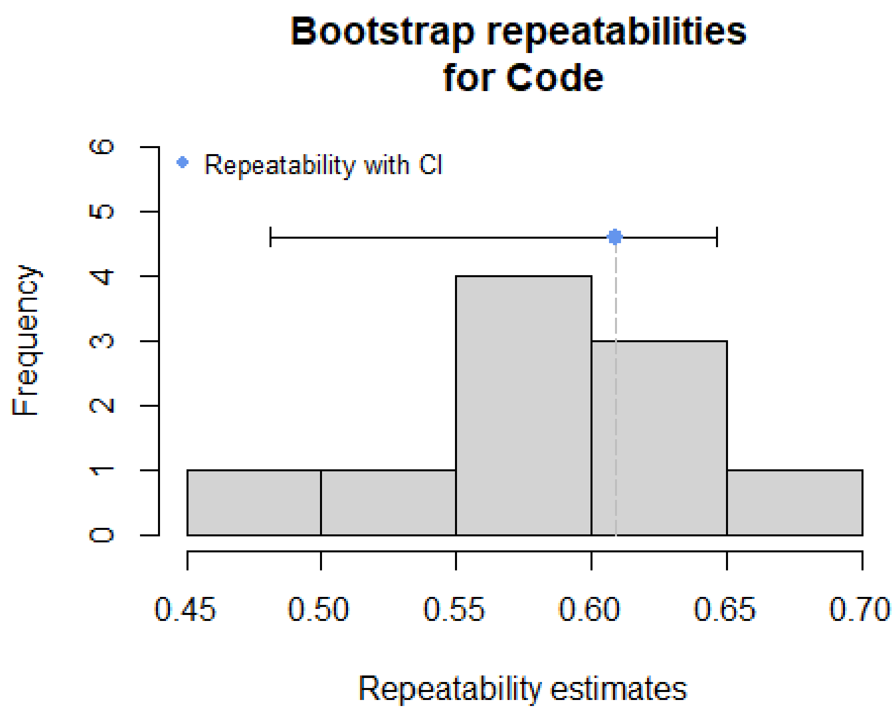
```
rep.log.Start <- rpt(log.Start ~ (1 | Code), grname = "Code", data = Behaviou
r, datatype = "Gaussian", nboot = 10, npermut = 0)
```

Analysis

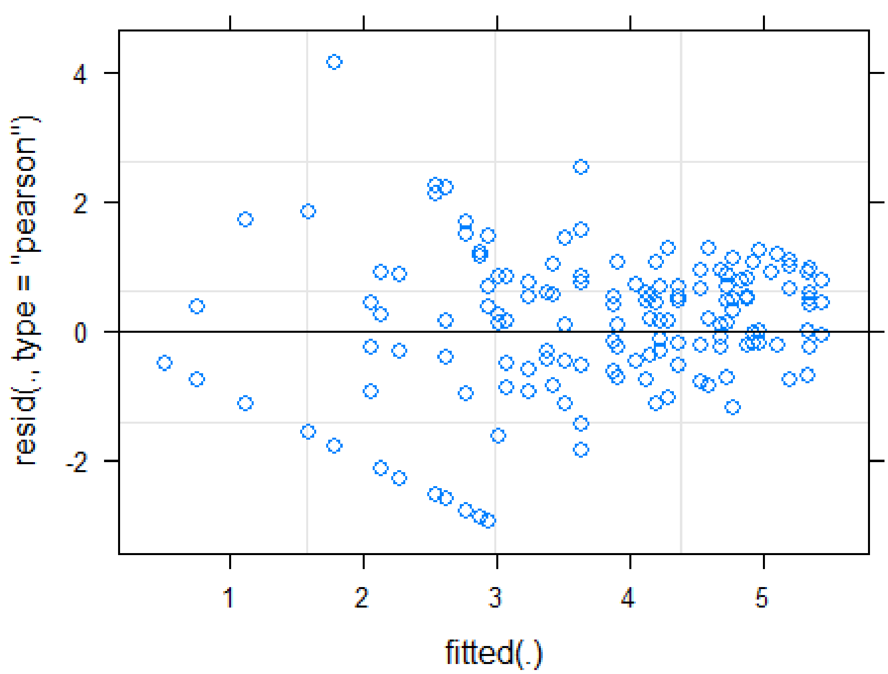
```
summary(rep.log.Start)
```

```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = log.Start ~ (1 | Code), grname = "Code", data = Behav
iour, datatype = "Gaussian", nboot = 10, npermut = 0)
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.609 0.0523  0.481  0.646       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.581  0.582  0.481  0.646
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -325.158
## logLik red. model = -358.4637
## D  = 66.6, df = 1, P = 1.65e-16
##
## ----------------------------------------
```

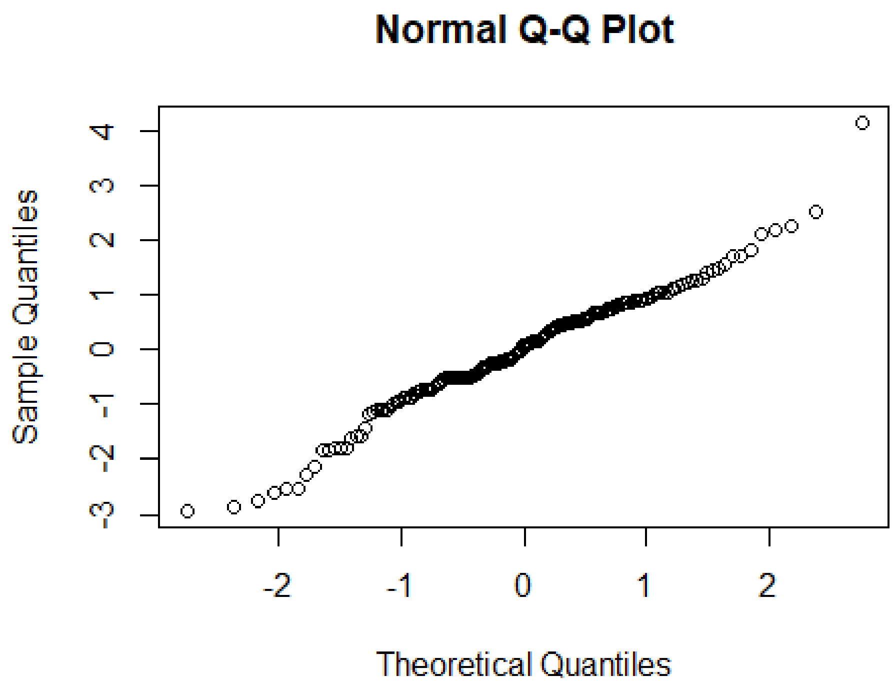
```
plot(rep.log.Start)
```




```
plot(rep.log.Start$mod)
```




```
qqnorm(resid(rep.log.Start$mod))
```




```
print(rep.log.Start)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.609
## SE = 0.052
## CI = [0.481, 0.646]
## P  = 1.65e-16 [LRT]
##      NA [Permutation]
```

Adjusted

For the adjusted estimations, I also included turtle sex (Sex) and day of testing (Day) as fixed effects, and site identity (Site) as a random effect.

Repeatability

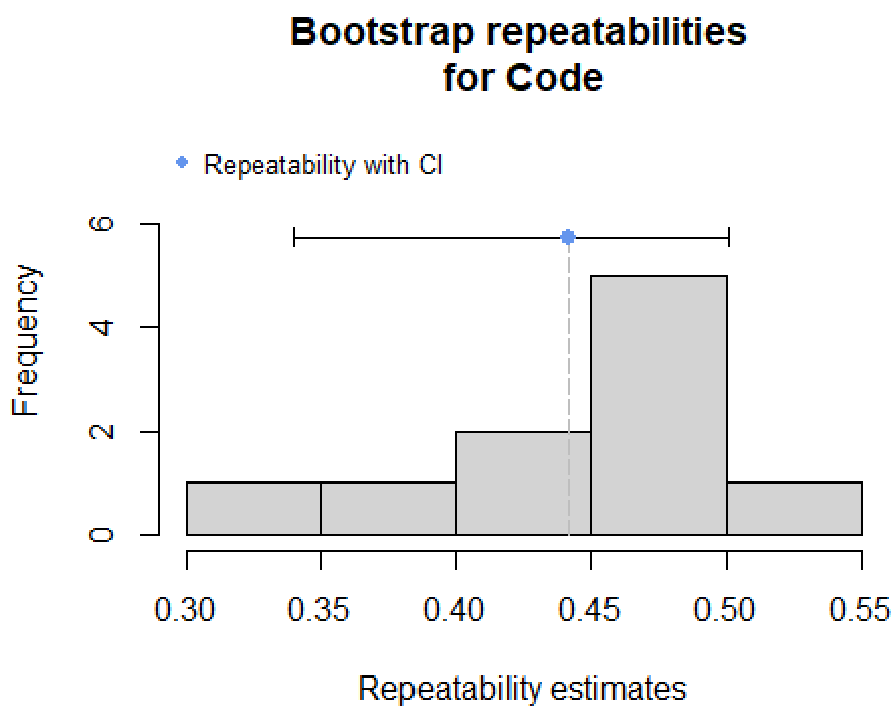
```
rep.fix.log.Start <- rpt(log.Start ~ Day + Sex + (1 | Code) + (1 | Site), grn
ame = "Code", data = Behaviour, datatype = "Gaussian",
                         nboot = 10, npermut = 0)
```

Analysis

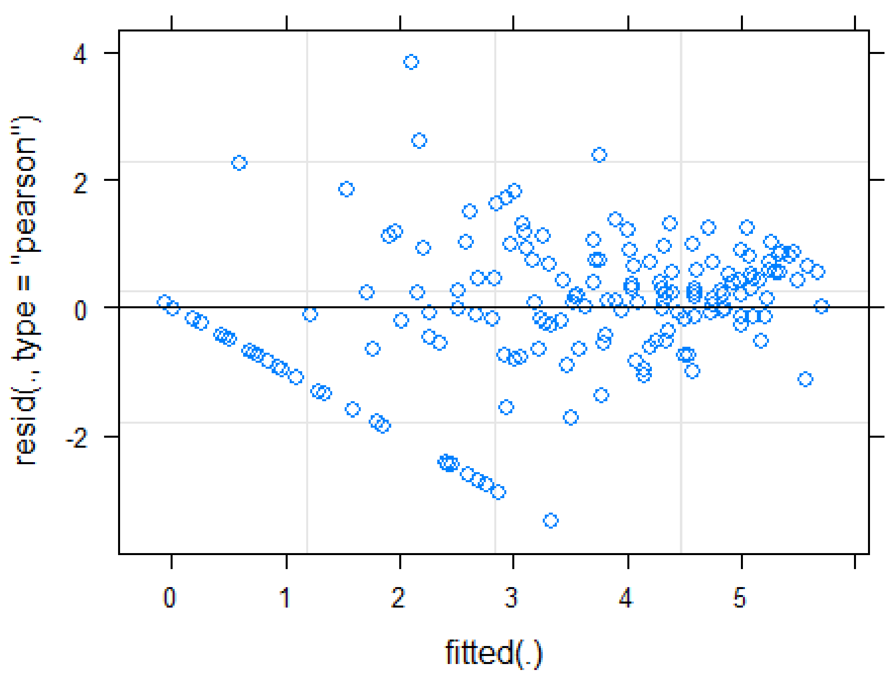
```
summary(rep.fix.log.Start)
```

```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = log.Start ~ Day + Sex + (1 | Code) + (1 | Site), grna
me = "Code", data = Behaviour, datatype = "Gaussian", nboot = 10, npermut = 0
)
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.442 0.0566   0.34  0.501       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.447  0.467   0.34  0.501
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -320.7227
## logLik red. model = -343.6035
## D  = 45.8, df = 1, P = 6.68e-12
##
## ----------------------------------------
```

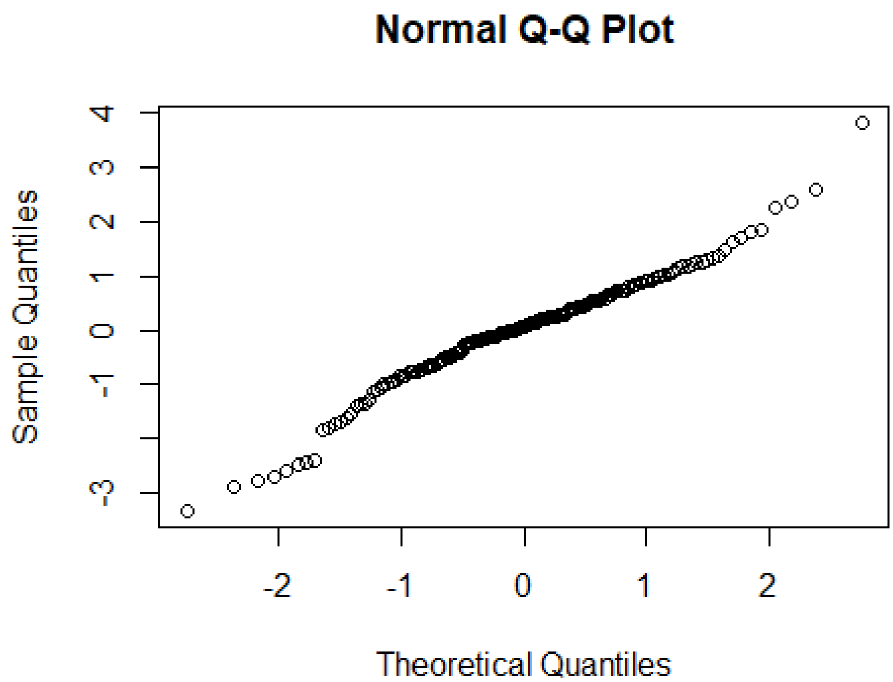
```
plot(rep.fix.log.Start)
```




```
plot(rep.fix.log.Start$mod)
```




```
qqnorm(resid(rep.fix.log.Start$mod))
```




```
print(rep.fix.log.Start)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.442
## SE = 0.057
## CI = [0.34, 0.501]
## P  = 6.68e-12 [LRT]
##      NA [Permutation]
```

Total time spent moving (Move)

Unadjusted

For the unadjusted estimations, I only include turtle ID (Code) as a random effect.

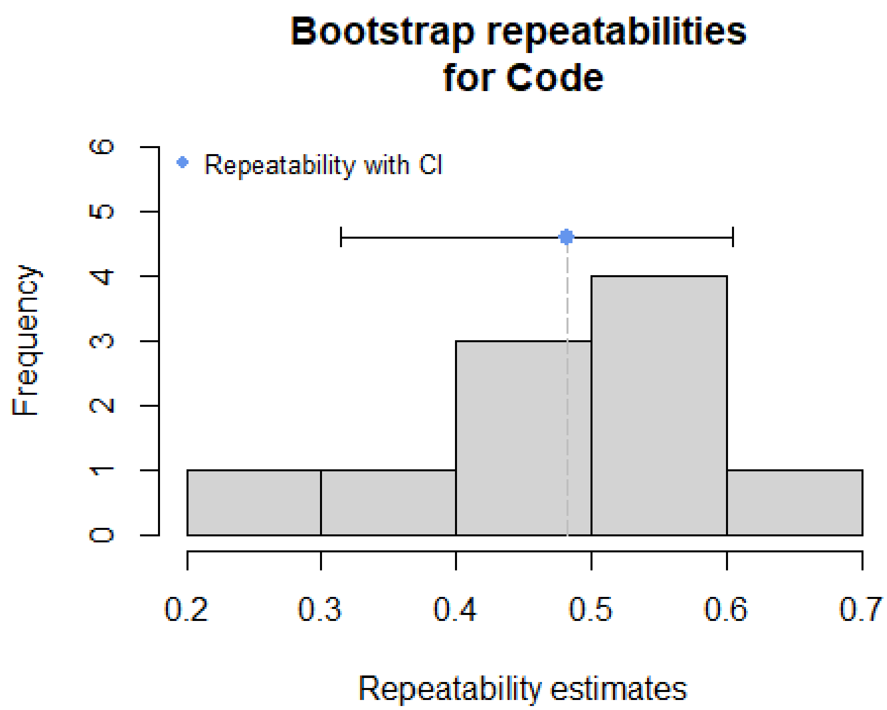
Repeatability

```
rep.Move <- rpt(Move ~ (1 | Code), grname = "Code", data = Behaviour, datatyp
e = "Gaussian", nboot = 10, npermut = 0)
```

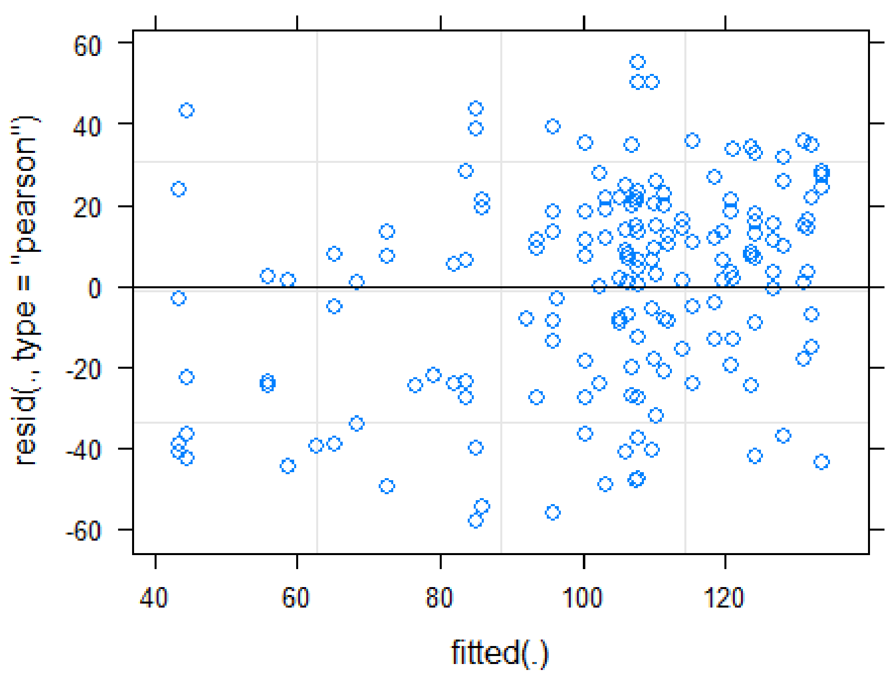
Analysis

```
summary(rep.Move)
```

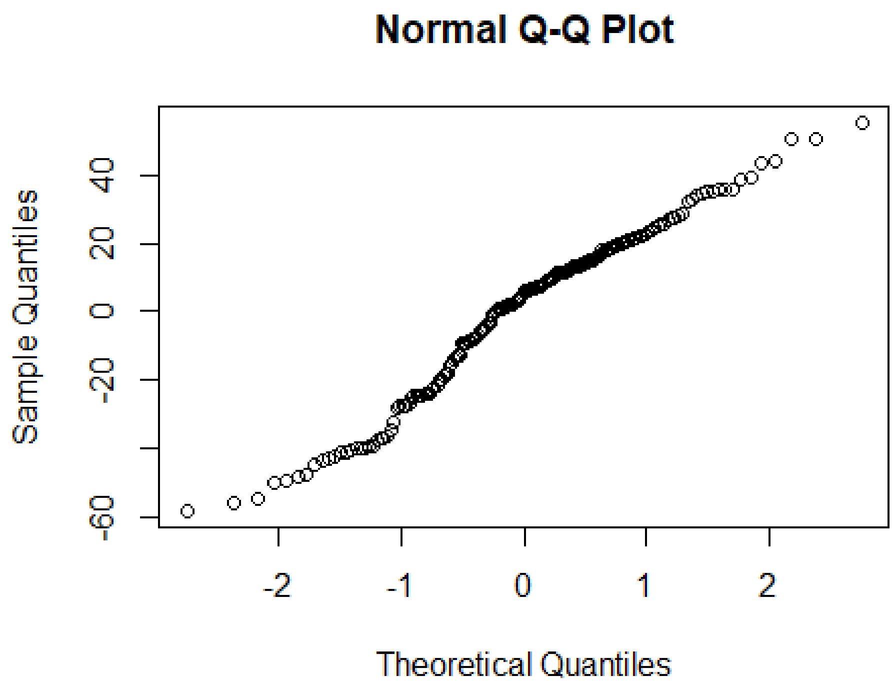
```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Move ~ (1 | Code), grname = "Code", data = Behaviour,
datatype = "Gaussian", nboot = 10, npermut = 0)
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.483 0.0978  0.314  0.605       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.471  0.493  0.314  0.605
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -845.3104
## logLik red. model = -862.5117
## D  = 34.4, df = 1, P = 2.24e-09
##
## ----------------------------------------

plot(rep.Move)
```




```
plot(rep.Move$mod)
```




```
qqnorm(resid(rep.Move$mod))
```




```
print(rep.Move)
```

```
##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.483
## SE = 0.098
## CI = [0.314, 0.605]
## P  = 2.24e-09 [LRT]
##      NA [Permutation]
```

Adjusted

For the adjusted estimations, I also included turtle sex (Sex) and day of testing (Day) as fixed effects, and site identity (Site) as a random effect.

Repeatability

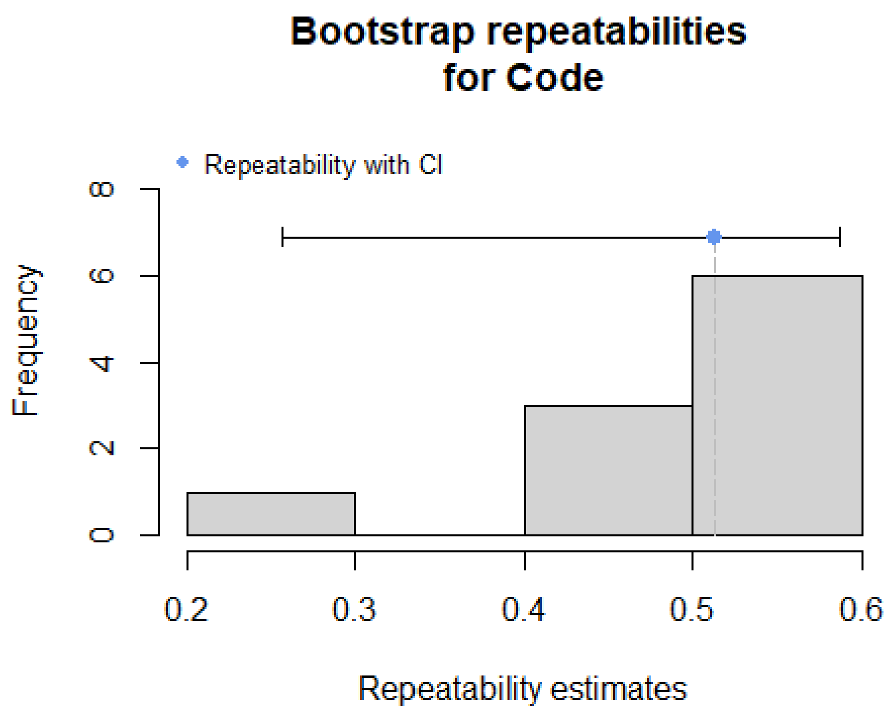
```
rep.fix.Move <- rpt(Move ~ Day + Sex + (1 | Code) + (1 | Site), grname = "Cod
e", data = Behaviour, datatype = "Gaussian",
                       nboot = 10, npermut = 0)
```

Analysis

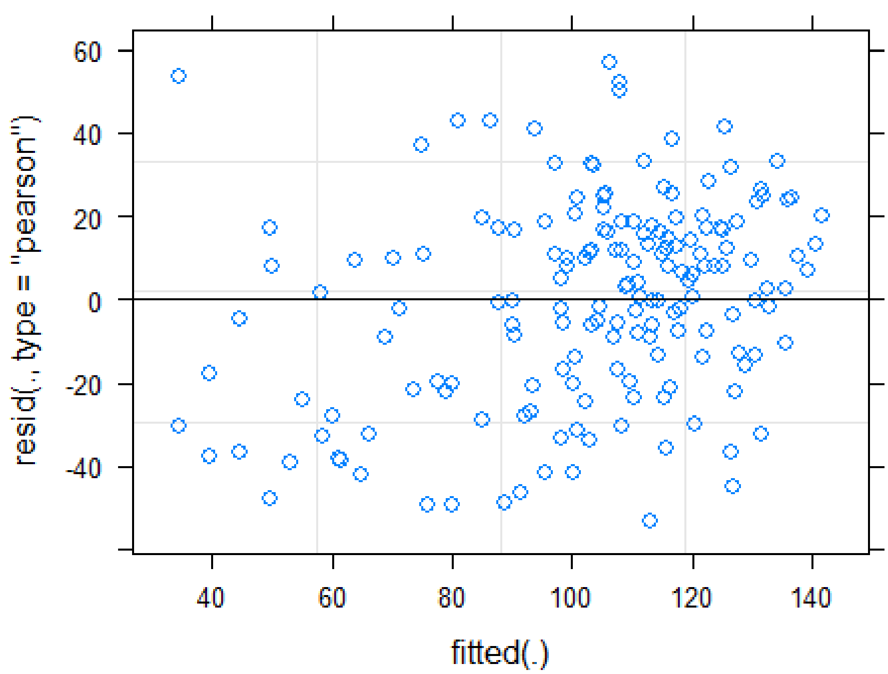
```
summary(rep.fix.Move)
```

```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Move ~ Day + Sex + (1 | Code) + (1 | Site), grname =
"Code", data = Behaviour, datatype = "Gaussian", nboot = 10, npermut = 0)
##
## Data: 170 observations
## ----------------------------------------
##
## Code (52 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.513   0.11  0.257  0.587       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.483  0.519  0.257  0.587
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -834.9996
## logLik red. model = -853.0628
## D  = 36.1, df = 1, P = 9.25e-10
##
## ----------------------------------------
```

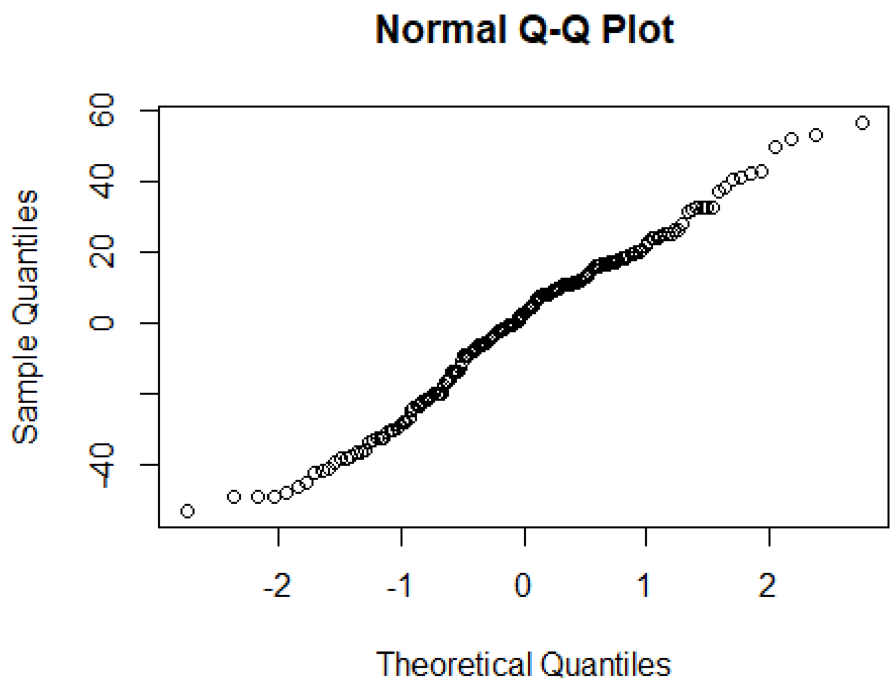
```
plot(rep.fix.Move)
```




```
plot(rep.fix.Move$mod)
```




```
qqnorm(resid(rep.fix.Move$mod))
```




```
print(rep.fix.Move)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.513
## SE = 0.11
## CI = [0.257, 0.587]
## P  = 9.25e-10 [LRT]
##      NA [Permutation]
```

Active defensive behaviours (Aggression)

Unadjusted

For the unadjusted estimations, I only included turtle ID (Code) as a random effect.

Repeatability

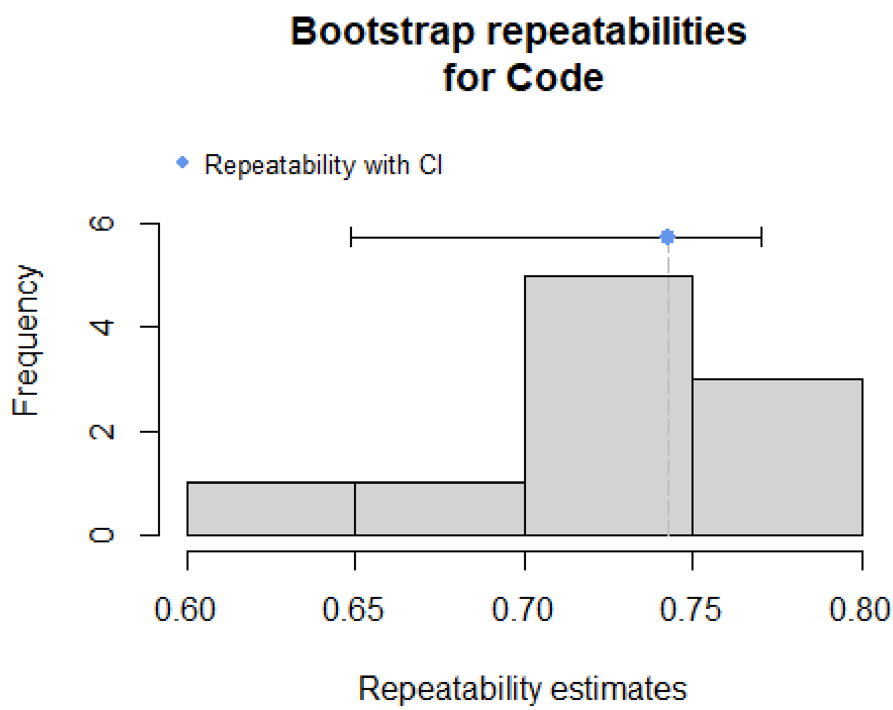
```
rep.Aggression <- rpt(Aggression ~ (1 | Code), grname = "Code", data = Behavi
our, datatype = "Gaussian", nboot = 10, npermut = 0)
```

Analysis

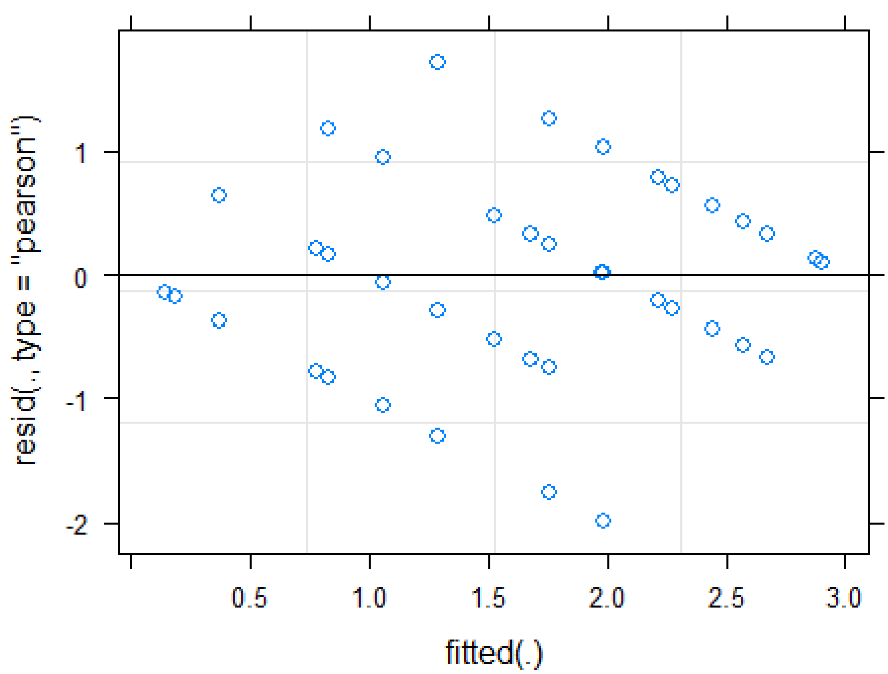
```
summary(rep.Aggression)
```

```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Aggression ~ (1 | Code), grname = "Code", data = Beha
viour, datatype = "Gaussian", nboot = 10, npermut = 0)
##
## Data: 214 observations
## ----------------------------------------
##
## Code (58 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.743 0.0424  0.648   0.77       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.724  0.736  0.648   0.77
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -246.7537
## logLik red. model = -318.9777
## D  = 144, df = 1, P = 1.42e-33
##
## ----------------------------------------
```

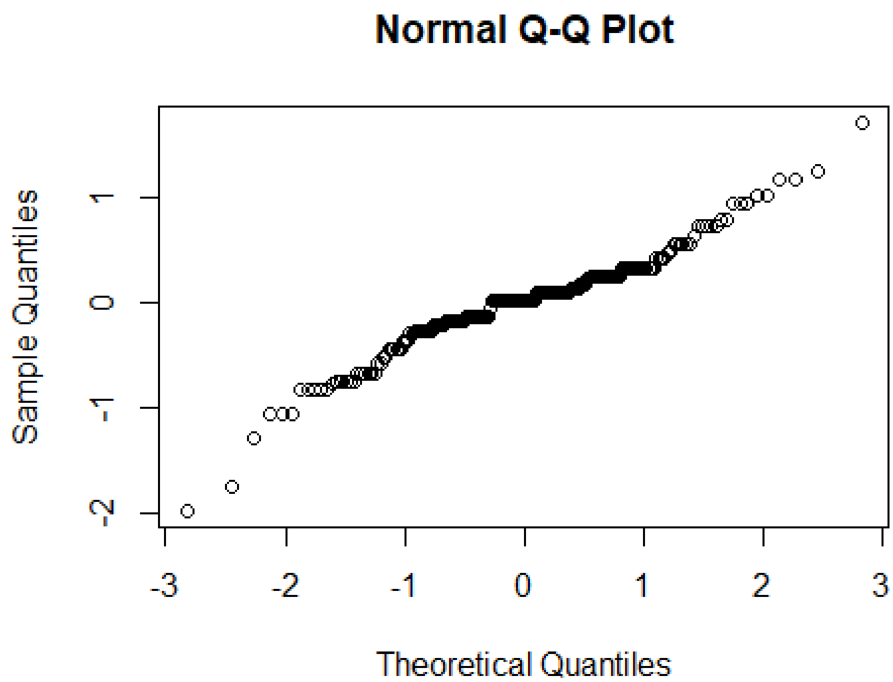
```
plot(rep.Aggression)
```




```
plot(rep.Aggression$mod)
```




```
qqnorm(resid(rep.Aggression$mod))
```




```
print(rep.Aggression)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.743
## SE = 0.042
## CI = [0.648, 0.77]
## P  = 1.42e-33 [LRT]
##      NA [Permutation]
```

Adjusted

For the adjusted estimations, I also included turtle sex (Sex) and day of testing (Day) as fixed effects, and site identity (Site) as a random effect.

Repeatability

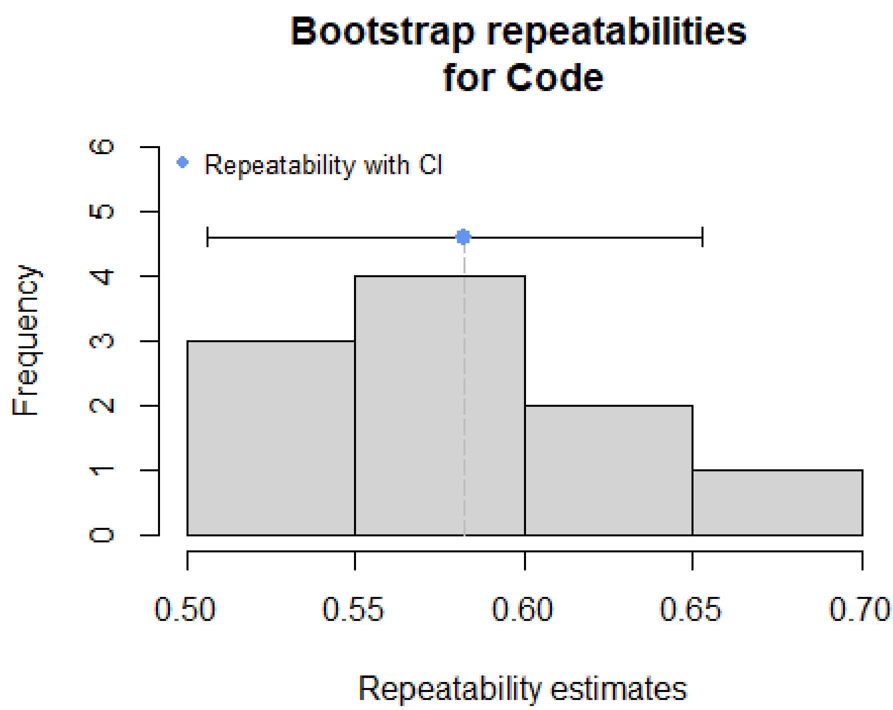
```
rep.fix.Aggression <- rpt(Aggression ~ Day + Sex + (1 | Site) + (1 | Code), g
rname = "Code", data = Behaviour, datatype = "Gaussian",
                              nboot = 10, npermut = 0)
```

Analysis

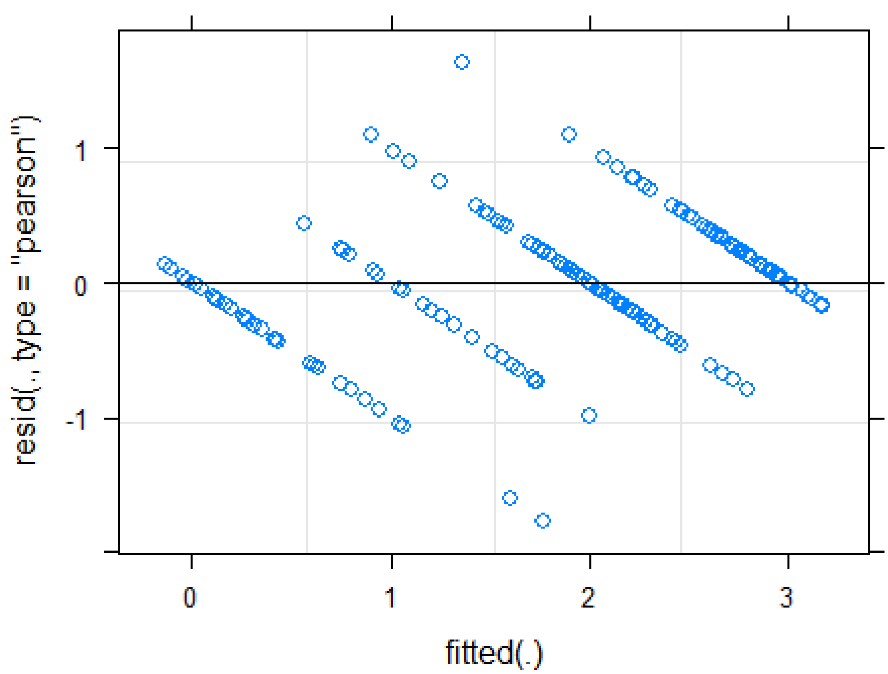
```
summary(rep.fix.Aggression)
```

```
##
## Repeatability estimation using the lmm method
##
## Call = rpt(formula = Aggression ~ Day + Sex + (1 | Site) + (1 | Code), grn
ame = "Code", data = Behaviour, datatype = "Gaussian", nboot = 10, npermut =
0)
##
## Data: 214 observations
## ----------------------------------------
##
## Code (58 groups)
##
## Repeatability estimation overview:
##       R     SE   2.5%  97.5% P_permut  LRT_P
##   0.582 0.0497  0.506  0.653       NA      0
##
## Bootstrapping and Permutation test:
##             N   Mean Median   2.5%  97.5%
## boot       10  0.571  0.555  0.506  0.653
## permut      1     NA     NA     NA     NA
##
## Likelihood ratio test:
## logLik full model = -234.9387
## logLik red. model = -288.0011
## D  = 106, df = 1, P = 3.46e-25
##
## ----------------------------------------
```

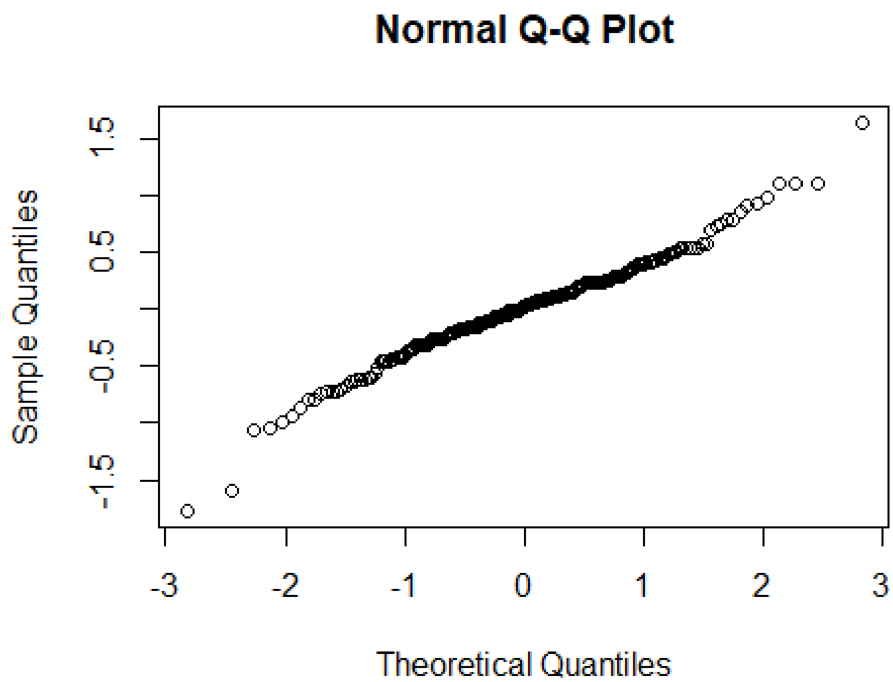
```
plot(rep.fix.Aggression)
```




```
plot(rep.fix.Aggression$mod)
```




```
qqnorm(resid(rep.fix.Aggression$mod))
```




```
print(rep.fix.Aggression)

##
##
## Repeatability estimation using the lmm method
##
## Repeatability for Code
## R  = 0.582
## SE = 0.05
## CI = [0.506, 0.653]
## P  = 3.46e-25 [LRT]
##      NA [Permutation]
```

Shell emergence time (binary) (BIN.Shell)

Unadjusted

For the unadjusted estimations, I only included turtle ID (Code) as a random effect.

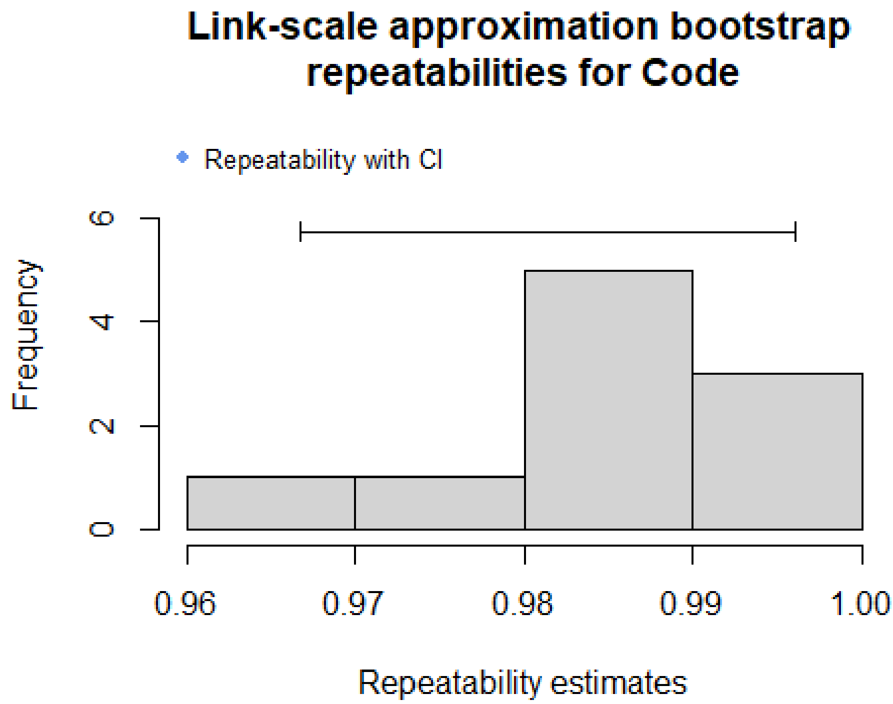
Repeatability

```
rep.Bin.Shell <- rpt(BIN.Shell ~ (1 | Code), grname = "Code", data = Behaviou
r, datatype = "Binary", nboot = 10, npermut = 0)
```

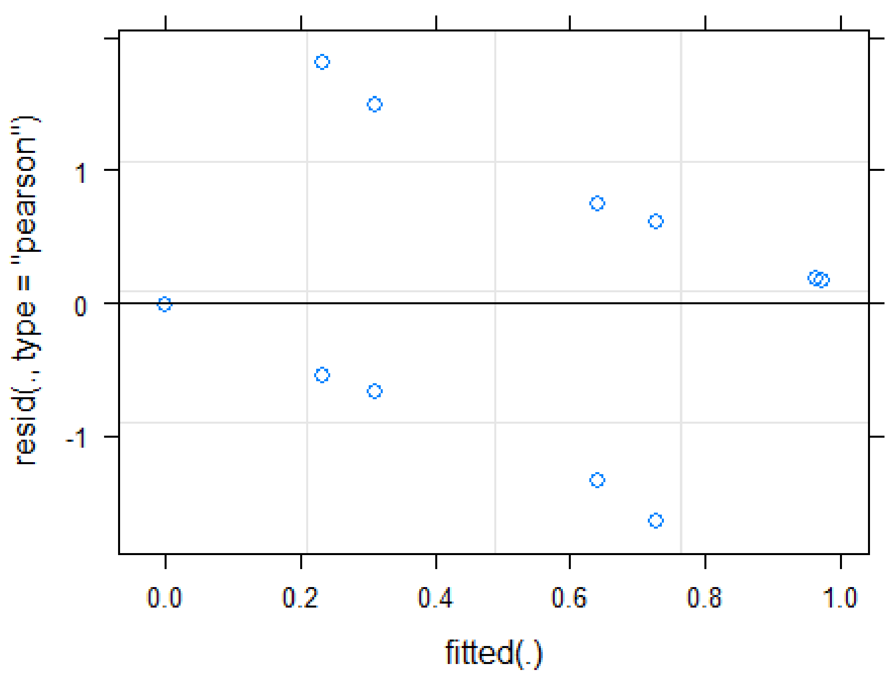
Analysis

```
summary(rep.Bin.Shell)
```

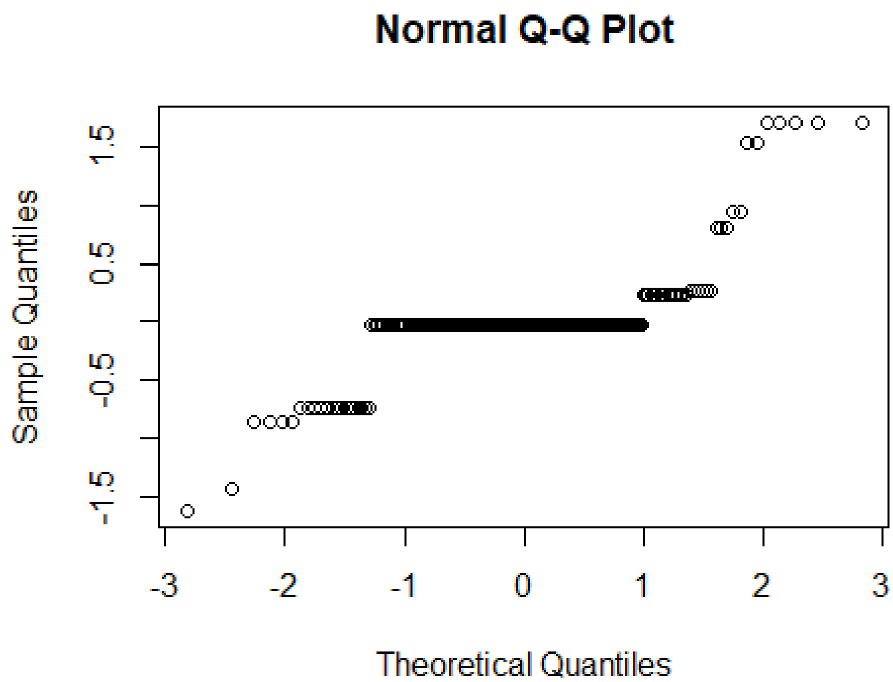
```
##
## Repeatability estimation using glmer method
##
## Call = rpt(formula = BIN.Shell ~ (1 | Code), grname = "Code", data = Behav
iour, datatype = "Binary", nboot = 10, npermut = 0)
##
## Data: 213 observations
## ----------------------------------------
##
## Code (58 groups)
##
## Repeatability estimation overview:
##           R      SE   2.5%  97.5% P_permut
## Org   0.675 0.04176  0.545  0.664       NA
## Link  0.939 0.00969  0.967  0.996       NA
##
##
## Bootstrapping:
##           N   Mean Median   2.5%  97.5%
## Org      10  0.597  0.591  0.545  0.664
## Link     10  0.984  0.984  0.967  0.996
##
## Permutation test:
##           N   Mean Median   2.5%  97.5% P_permut
## Org       1     NA     NA     NA     NA       NA
## Link      1     NA     NA     NA     NA       NA
##
## Likelihood ratio test:
## logLik full model = -55.21043
## logLik red. model = -93.51691
## D  = 76.6, df = 1, P = 1.04e-18
##
## ----------------------------------------

plot(rep.Bin.Shell)
```




```
plot(rep.Bin.Shell$mod)
```




```
qqnorm(resid(rep.Bin.Shell$mod))
```




```
print(rep.Bin.Shell)

##
##
## Repeatability estimation using the glmm method and logit link
##
## Repeatability for Code
## --------------------------------
## Link-scale approximation:
## R  = 0.939
## SE = 0.01
## CI = [0.967, 0.996]
## P  = 1.04e-18 [LRT]
##      NA [Permutation]
##
## Original-scale approximation:
## R  = 0.675
## SE = 0.042
## CI = [0.545, 0.664]
## P  = 1.04e-18 [LRT]
##      NA [Permutation]
```

Adjusted

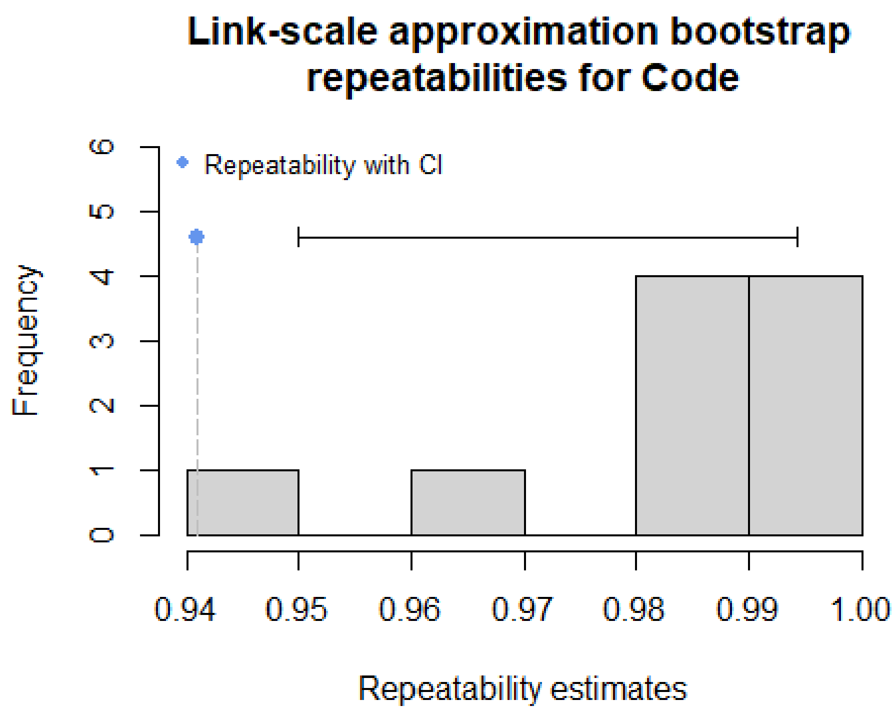
For the adjusted estimations, I also included turtle sex (Sex) and day of testing (Day) as fixed effects, and site identity (Site) as a random effect.

Repeatability

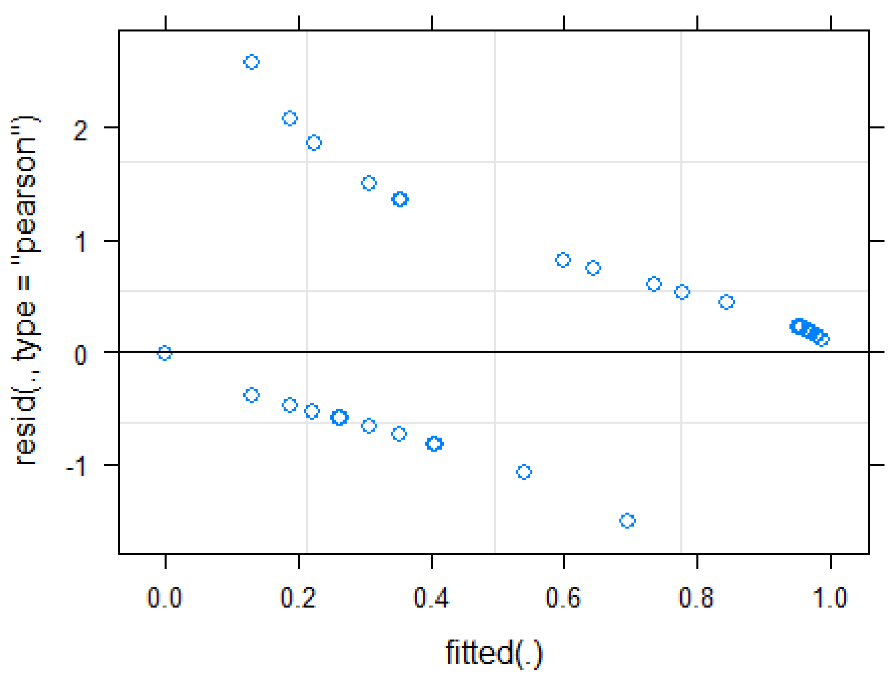
```
rep.fix.Bin.Shell <- rpt(BIN.Shell ~ Day + Sex + (1 | Code) + (1 | Site), grn
ame = "Code", data = Behaviour, datatype = "Binary",
                            nboot = 10, npermut = 0)
```

Analysis

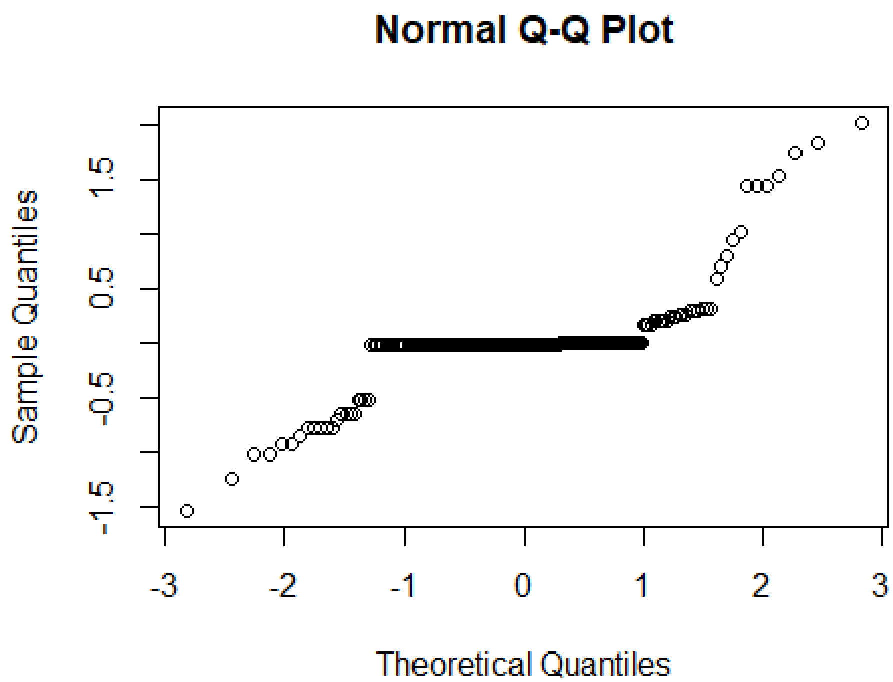
```
summary(rep.fix.Bin.Shell)

##
## Repeatability estimation using glmer method
##
## Call = rpt(formula = BIN.Shell ~ Day + Sex + (1 | Code) + (1 | Site), grna
me = "Code", data = Behaviour, datatype = "Binary", nboot = 10, npermut = 0)
##
## Data: 213 observations
## ----------------------------------------
##
## Code (58 groups)
##
## Repeatability estimation overview:
##            R     SE   2.5%  97.5% P_permut
## Org    0.677 0.0789  0.472  0.699       NA
## Link   0.941 0.0155  0.950  0.994       NA
##
##
## Bootstrapping:
##            N   Mean Median   2.5%  97.5%
## Org       10  0.581  0.589  0.472  0.699
## Link      10  0.982  0.989  0.950  0.994
##
## Permutation test:
##            N   Mean Median   2.5%  97.5% P_permut
## Org        1     NA     NA     NA     NA       NA
## Link       1     NA     NA     NA     NA       NA
##
## Likelihood ratio test:
## logLik full model = -54.50538
## logLik red. model = -87.67746
## D  = 66.3, df = 1, P = 1.89e-16
##
## ----------------------------------------

plot(rep.fix.Bin.Shell)
```




```
plot(rep.fix.Bin.Shell$mod)
```




```
qqnorm(resid(rep.fix.Bin.Shell$mod))
```



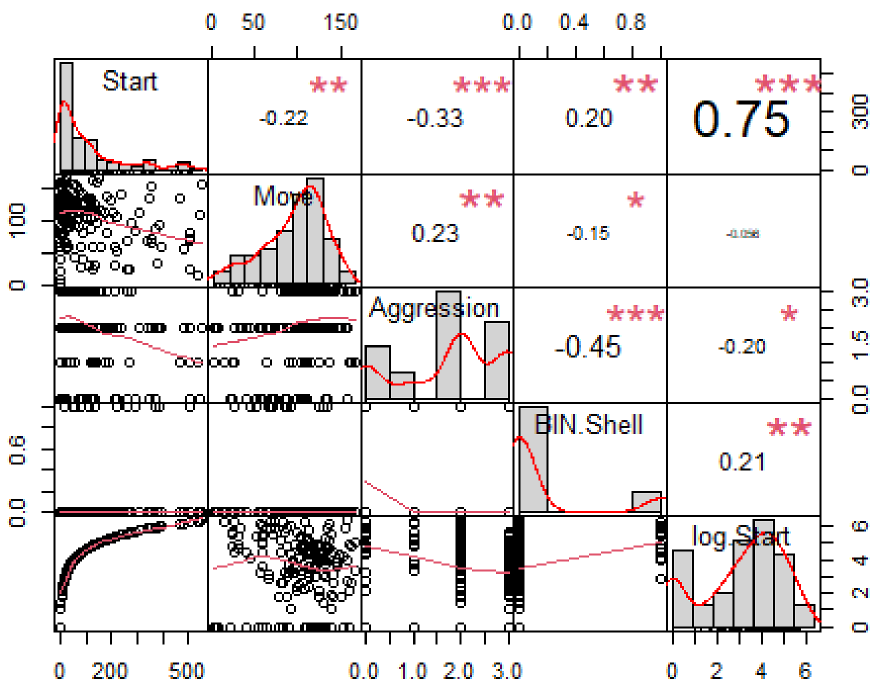

```
print(rep.fix.Bin.Shell)

##
##
## Repeatability estimation using the glmm method and logit link
##
## Repeatability for Code
## --------------------------------
## Link-scale approximation:
## R  = 0.941
## SE = 0.016
## CI = [0.95, 0.994]
## P  = 1.89e-16 [LRT]
##      NA [Permutation]
##
## Original-scale approximation:
## R  = 0.677
## SE = 0.079
## CI = [0.472, 0.699]
## P  = 1.89e-16 [LRT]
##      NA [Permutation]
```

Correlations between behaviors

Calculation of the Pearson and Spearman correlations coefficients between behaviors

Visualization of the correlations

```
cor.behaviour <- Behaviour[, c(7,8,9,10,12)]
chart.Correlation(cor.behaviour, histogram=TRUE, pch=19)
```



Creation of the correlation tables

Pearson correlation coefficients

```
table.corr.pearson <- rcorr(as.matrix(cor.behaviour), type="pearson")
table.rcorr.pearson <- table.corr.pearson$r # pearson correlation coefficient
s
table.p.pearson <- table.corr.pearson$P # p value of the correlations
table.rcorr.pearson

##                  Start        Move Aggression  BIN.Shell   log.Start
## Start       1.0000000 -0.21939548 -0.3324922  0.2044127  0.74975899
## Move       -0.2193955  1.00000000  0.2280770 -0.1508184 -0.05567449
## Aggression -0.3324922  0.22807697  1.0000000 -0.4545812 -0.19691759
## BIN.Shell   0.2044127 -0.15081844 -0.4545812  1.0000000  0.20961530
## log.Start   0.7497590 -0.05567449 -0.1969176  0.2096153  1.00000000

table.p.pearson
```

```
##                  Start        Move   Aggression    BIN.Shell   log.Start
## Start               NA 0.004045790 9.428660e-06 7.498693e-03 0.000000000
## Move       4.045790e-03          NA 2.777485e-03 4.962667e-02 0.470841622
## Aggression 9.428660e-06 0.002777485           NA 2.938094e-12 0.010058843
## BIN.Shell  7.498693e-03 0.049626670 2.938094e-12           NA 0.006080079
## log.Start  0.000000e+00 0.470841622 1.005884e-02 6.080079e-03          NA
```

Spearman correlation coefficients

```
table.corr.spearman <- rcorr(as.matrix(cor.behaviour), type="spearman")
table.rcorr.spearman <- table.corr.spearman$r # spearman correlation coeffici
ents
table.p.spearman <- table.corr.spearman$P # p value of the correlations
table.rcorr.spearman
```

```
##                Start       Move Aggression  BIN.Shell  log.Start
## Start       1.0000000 -0.1381580 -0.2442427  0.2203949  1.0000000
## Move       -0.1381580  1.0000000  0.2096954 -0.1617061 -0.1381580
## Aggression -0.2442427  0.2096954  1.0000000 -0.4235177 -0.2442427
## BIN.Shell   0.2203949 -0.1617061 -0.4235177  1.0000000  0.2203949
## log.Start   1.0000000 -0.1381580 -0.2442427  0.2203949  1.0000000
```

```
table.p.spearman
```

```
##                 Start        Move   Aggression    BIN.Shell   log.Start
## Start              NA 0.072383880 1.328297e-03 3.877059e-03 0.000000000
## Move       0.072383880          NA 6.060241e-03 3.514192e-02 0.072383880
## Aggression 0.001328297 0.006060241           NA 1.112337e-10 0.001328297
## BIN.Shell  0.003877059 0.035141916 1.112337e-10           NA 0.003877059
## log.Start  0.000000000 0.072383880 1.328297e-03 3.877059e-03          NA
```