Rerun of Clements et al data original complete and corrected olfaction habitat and risk 1 as lowest

Anon

06/04/2022

————————– Supplementary information to ————————-

——————— META-ANALYSIS REVEALS AN EXTREME "DECLINE EFFECT" ——————— ——————— IN OCEAN ACIDIFICATION IMPACTS ON FISH BEHAVIOUR ———————-

——– Jeff C. Clements, Josefin Sundin, Timothy D. Clark, Fredrik Jutfelt ——–-

---

Get packages

```
library(pacman)

## Warning: package 'pacman' was built under R version 4.1.2

pacman::p_load(metafor, MCMCglmm, tidyverse, rotl, magrittr, kableExtra,
rmarkdown,gridExtra, psych, bindrcpp, pander)
library(BiocManager)

## Warning: package 'BiocManager' was built under R version 4.1.2

library(ggplot2)
library(viridis)

## Loading required package: viridisLite

library(patchwork)

## Warning: package 'patchwork' was built under R version 4.1.2
```

META-ANALYSIS - YEAR ONLINE - FULL DATASET

##attach dataset

```
decline<-read.csv(file.choose())
attach(decline)
```

#Note s5 data complete and corrected olfaction risk and habitat 1 as the lowest

```
head(decline)

##   obs study      authors year.online year.print if.at.pub X2017.if
if.group
## 1   1    a1 Munday et al        2009       2009     9.432    9.504
J
## 2   2    a1 Munday et al        2009       2009     9.432    9.504
J
## 3   3    a1 Munday et al        2009       2009     9.432    9.504
J
## 4   4    a1 Munday et al        2009       2009     9.432    9.504
J
## 5   5    a1 Munday et al        2009       2009     9.432    9.504
J
## 6   6    a1 Munday et al        2009       2009     9.432    9.504
J
##   avg.n            species climate cue cue.type life.stage ctrl.n
ctrl.mean
## 1    27 Amphiprion percula    Trop Yes  Habitat     Larvae     26
94.129
## 2    NA Amphiprion percula    Trop Yes  Habitat     Larvae     20
0.783
## 3    NA Amphiprion percula    Trop Yes  Habitat     Larvae     20
46.380
## 4    NA Amphiprion percula    Trop Yes  Habitat     Larvae     10
98.826
## 5    NA Amphiprion percula    Trop Yes      Kin     Larvae     30
0.912
## 6    NA Amphiprion percula    Trop Yes      Kin     Larvae     30
90.876
##     ctrl.sd oa.n oa.mean        oa.sd
## 1  2.993124   46  72.789 13.273019780
## 2  3.501682   46  83.562 10.621128750
## 3 14.002258   46  71.429 13.273019780
## 4  1.856257   16  88.258  7.044000000
## 5  2.996042   20  99.818  0.004472136
## 6  3.998375   20  99.818  0.004472136
```

##set factors

```
decline$year.online<-as.factor(decline$year.online)
decline$year.print<-as.factor(decline$year.print)
decline$obs<-as.factor(decline$obs)
decline$study<-as.factor(decline$study)
```

##view summary

```
summary(decline)

##       obs          study      authors           year.online   year.print
##  1      :  1   a3     :48   Length:224         2010   :48   2010   :60
```

```
##  2       :  1    a31     :28    Class  :character    2014    :34    2012    :44
##  3       :  1    a11     :21    Mode   :character    2011    :29    2014    :38
##  4       :  1    a9      :18                         2012    :29    2018    :27
##  5       :  1    a87     :16                         2018    :29    2013    :17
##  6       :  1    a2      :12                         2009    :19    2017    :14
##  (Other):218   (Other):81                           (Other):36    (Other):24
##    if.at.pub         X2017.if        if.group             avg.n
##  Min.   : 1.658   Min.   : 1.729   Length:224         Min.   :  8.00
##  1st Qu.: 3.483   1st Qu.: 2.766   Class :character   1st Qu.: 15.00
##  Median : 7.543   Median : 8.997   Mode  :character   Median : 20.00
##  Mean   : 7.419   Mean   : 8.009                      Mean   : 46.93
##  3rd Qu.: 9.771   3rd Qu.: 9.504                      3rd Qu.: 28.00
##  Max.   :19.181   Max.   :19.181                      Max.   :568.00
##                                                       NA's   :195
##    species            climate              cue              cue.type
##  Length:224         Length:224         Length:224         Length:224
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   life.stage            ctrl.n         ctrl.mean          ctrl.sd
##  Length:224         Min.   :  8.00   Min.   :-69.781   Min.   :  0.0001
##  Class :character   1st Qu.: 14.00   1st Qu.:  1.000   1st Qu.:  0.9112
##  Mode  :character   Median : 20.00   Median :  7.248   Median :  3.5543
##                     Mean   : 31.48   Mean   : 18.186   Mean   : 11.0884
##                     3rd Qu.: 29.00   3rd Qu.: 24.462   3rd Qu.: 12.7898
##                     Max.   :752.00   Max.   :122.938   Max.   :194.9300
##
##       oa.n           oa.mean           oa.sd
##  Min.   :  8.00   Min.   :-59.67   Min.   :  0.0001
##  1st Qu.: 14.00   1st Qu.:  1.00   1st Qu.:  1.0500
##  Median : 20.00   Median : 15.46   Median :  4.7842
##  Mean   : 31.72   Mean   : 31.85   Mean   : 13.3291
##  3rd Qu.: 29.00   3rd Qu.: 55.27   3rd Qu.: 18.4768
##  Max.   :755.00   Max.   :168.18   Max.   :222.0120
##
```

##subset by year

```
y2009 <- filter(decline, year.online == "2009")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2009$obs <- 1:nrow(y2009)
y2010 <- filter(decline, year.online == "2010")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2010$obs <- 1:nrow(y2010)
y2011 <- filter(decline, year.online == "2011")[,-
```

```
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2011$obs <- 1:nrow(y2011)
y2012 <- filter(decline, year.online == "2012")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2012$obs <- 1:nrow(y2012)
y2013 <- filter(decline, year.online == "2013")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2013$obs <- 1:nrow(y2013)
y2014 <- filter(decline, year.online == "2014")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2014$obs <- 1:nrow(y2014)
#y2015 <- filter(decline, year.online == "2015")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
#y2015$obs <- 1:nrow(y2015)
y2016 <- filter(decline, year.online == "2016")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2016$obs <- 1:nrow(y2016)
y2017 <- filter(decline, year.online == "2017")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2017$obs <- 1:nrow(y2017)
y2018 <- filter(decline, year.online == "2018")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2018$obs <- 1:nrow(y2018)
y2019 <- filter(decline, year.online == "2019")[,-
match(c("avg.n","cue.type","if.at.pub","X2017.if","if.group"), colnames
(decline))]
y2019$obs <- 1:nrow(y2019)
```

##comupte effect sizes for each year

```
lnRR2009 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
009,append=TRUE)
lnRR2010 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
010,append=TRUE)
lnRR2011 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
011,append=TRUE)
lnRR2012 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
012,append=TRUE)
```

```
lnRR2013 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
013,append=TRUE)
lnRR2014 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
014,append=TRUE)
#lnRR2015 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
015,append=TRUE)
lnRR2016 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
016,append=TRUE)
lnRR2017 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
017,append=TRUE)
lnRR2018 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
018,append=TRUE)
lnRR2019 <- escalc(measure= "ROM",
m1i=oa.mean,sd1i=oa.sd,n1i=oa.n,m2i=ctrl.mean,sd2i=ctrl.sd,n2i=ctrl.n,data=y2
019,append=TRUE)

# Note log(m1i/m21i) produced NAs for 2011, 2012, 2013, 2015, 2017
```

##remove NAs

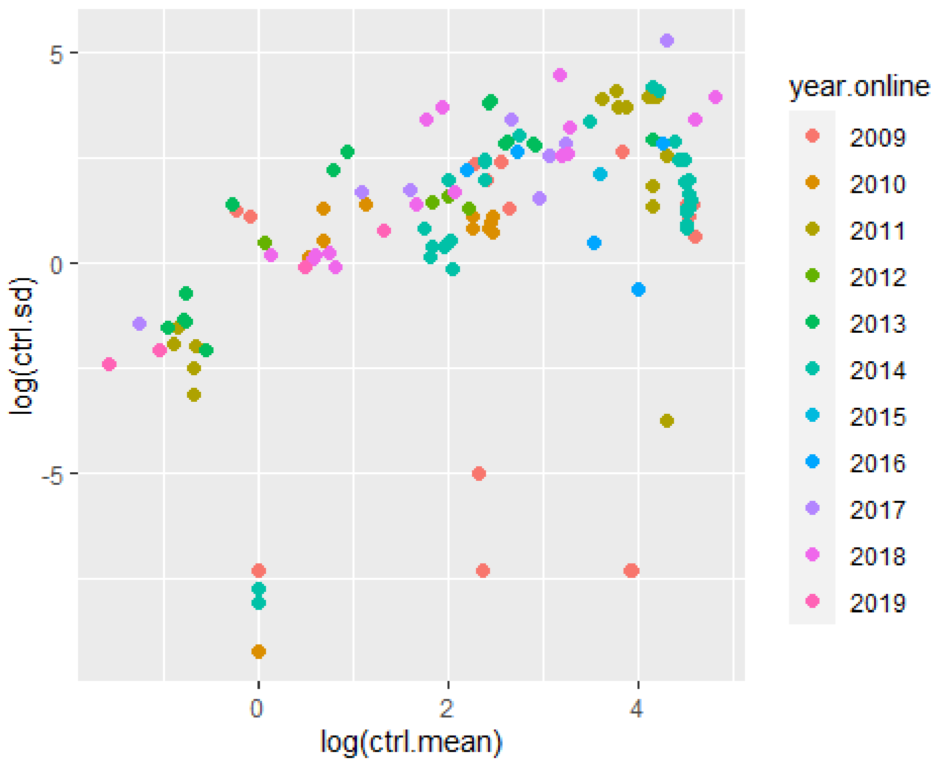
```
lnRR2009clean<-na.omit(lnRR2009)
lnRR2010clean<-na.omit(lnRR2010)
lnRR2011clean<-na.omit(lnRR2011)
lnRR2012clean<-na.omit(lnRR2012)
lnRR2013clean<-na.omit(lnRR2013)
lnRR2014clean<-na.omit(lnRR2014)
#lnRR2015clean<-na.omit(lnRR2015)
lnRR2016clean<-na.omit(lnRR2016)
lnRR2017clean<-na.omit(lnRR2017)
lnRR2018clean<-na.omit(lnRR2018)
lnRR2019clean<-na.omit(lnRR2019)
```

##view mean-variance relationship

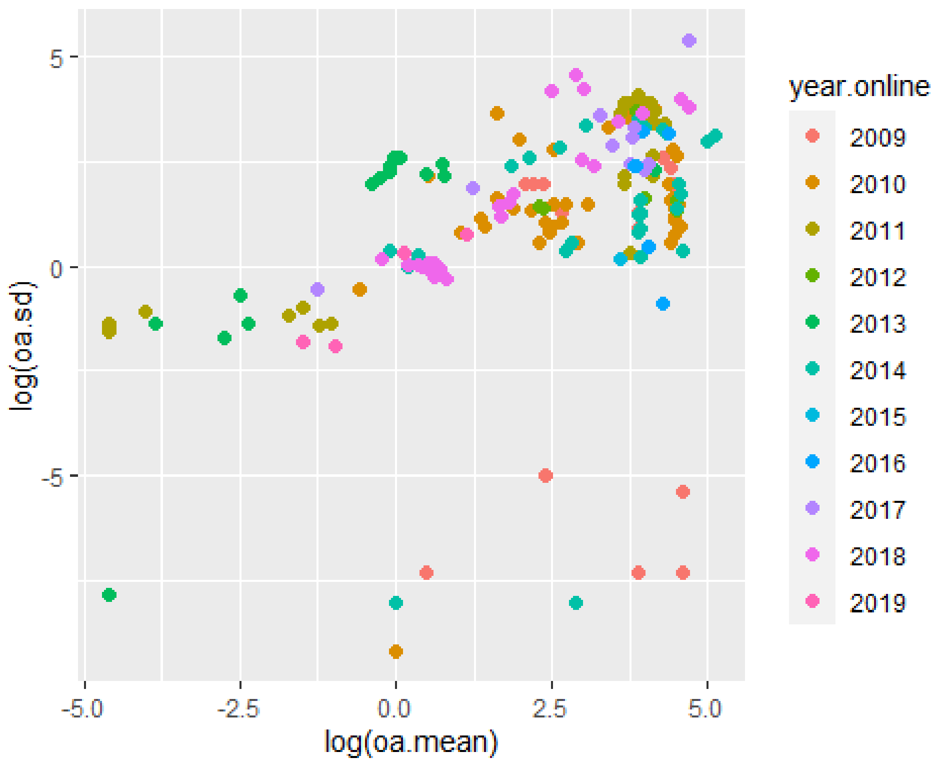
```
pp1<-ggplot(decline,aes(x=log(ctrl.mean),y=log(ctrl.sd),col=year.online))+
geom_point(size=2,na.rm = TRUE)
pp2<-ggplot(decline,aes(x=log(oa.mean),y=log(oa.sd),col=year.online))+
geom_point(size=2,na.rm = TRUE)
pp1 #Note added

## Warning in log(ctrl.mean): NaNs produced

## Warning in log(ctrl.mean): NaNs produced
```




```
pp2 #jNote added
## Warning in log(oa.mean): NaNs produced
## Warning in log(oa.mean): NaNs produced
```



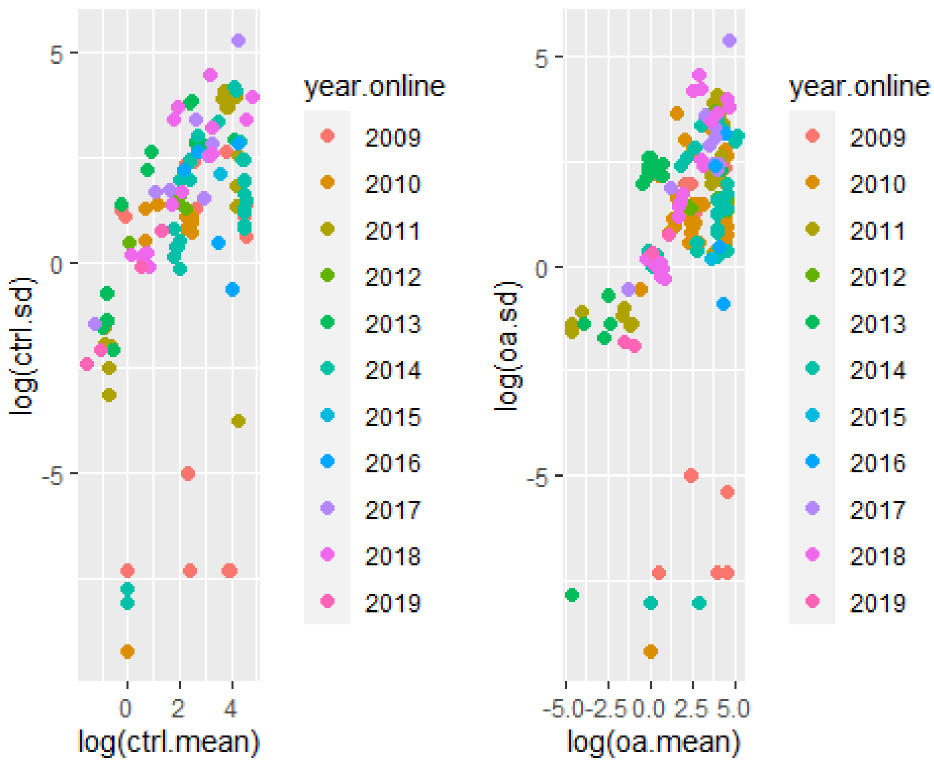

```
grid.arrange(pp1,pp2, nrow =1)

## Warning in log(ctrl.mean): NaNs produced

## Warning in log(ctrl.mean): NaNs produced

## Warning in log(oa.mean): NaNs produced

## Warning in log(oa.mean): NaNs produced
```




```
#Note there are a number of NAs created due the the log of a negative being
not computible
```

##look at lnRR by year

```
MLMA_2009_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2009)

## Warning: Ratio of largest to smallest sampling variance extremely large.
May not
## be able to obtain stable results.

summary(MLMA_2009_lnRR)

##
## Multivariate Meta-Analysis Model (k = 19; method: REML)
##
##    logLik  Deviance       AIC       BIC      AICc
## -39.8746   79.7493   83.7493   85.5300   84.5493
##
## Variance Components:
##
##              estim    sqrt  nlvls  fixed  factor
## sigma^2     4.7709  2.1842     19     no     obs
##
## Test for Heterogeneity:
## Q(df = 18) = 4124865078.0293, p-val < .0001
##
## Model Results:
```

```
##
## estimate      se    zval    pval   ci.lb   ci.ub
##   1.2647  0.5046  2.5062  0.0122  0.2757  2.2538  *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2010_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2010)
```

```
## Warning: Ratio of largest to smallest sampling variance extremely large.
May not
## be able to obtain stable results.
```

```
summary(MLMA_2010_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 48; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -91.7106  183.4213  187.4213  191.1216  187.6940
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    2.8052  1.6749     48     no     obs
##
## Test for Heterogeneity:
## Q(df = 47) = 1407483.9857, p-val < .0001
##
## Model Results:
##
## estimate      se    zval    pval   ci.lb   ci.ub
##   1.6112  0.2473  6.5150  <.0001  1.1265  2.0959  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2011_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2011)
summary(MLMA_2011_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 29; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -18.3957   36.7914   40.7914   43.4558   41.2714
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    0.0430  0.2073     29     no     obs
##
```

```
## Test for Heterogeneity:
## Q(df = 28) = 84.9825, p-val < .0001
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb    ci.ub
##  -0.2147  0.0617  -3.4785  0.0005  -0.3357  -0.0937  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2012_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2012)
summary(MLMA_2012_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 29; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -64.2157  128.4315  132.4315  135.0959  132.9115
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    2.9082  1.7054     29     no     obs
##
## Test for Heterogeneity:
## Q(df = 28) = 802.2060, p-val < .0001
##
## Model Results:
##
## estimate      se    zval    pval    ci.lb   ci.ub
##   0.1705  0.3777  0.4515  0.6517  -0.5697  0.9107
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2013_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2013)
summary(MLMA_2013_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 15; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -26.3805   52.7611   56.7611   58.0392   57.8520
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    1.3793  1.1744     15     no     obs
##
## Test for Heterogeneity:
```

```
## Q(df = 14) = 202.1515, p-val < .0001
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb    ci.ub
##  -1.7781  0.4521  -3.9329  <.0001  -2.6643  -0.8920  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2014_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2014)
summary(MLMA_2014_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 34; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -74.9609  149.9217  153.9217  156.9147  154.3217
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    5.4553  2.3357     34     no     obs
##
## Test for Heterogeneity:
## Q(df = 33) = 456372.2158, p-val < .0001
##
## Model Results:
##
## estimate      se    zval    pval    ci.lb   ci.ub
##   0.2464  0.4027  0.6119  0.5406  -0.5429  1.0358
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#MLMA_2015_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2015)
#summary(MLMA_2015_lnRR)
MLMA_2016_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2016)
summary(MLMA_2016_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 6; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
##  -7.0986   14.1971   18.1971   17.4160   24.1971
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    0.9608  0.9802      6     no     obs
##
```

```
## Test for Heterogeneity:
## Q(df = 5) = 551.9130, p-val < .0001
##
## Model Results:
##
## estimate      se    zval    pval    ci.lb   ci.ub
##   0.5459  0.4044  1.3501  0.1770  -0.2466  1.3385
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2017_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2017)
summary(MLMA_2017_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 10; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
##  -9.7965   19.5931   23.5931   23.9875   25.5931
##
## Variance Components:
##
##              estim    sqrt  nlvls  fixed  factor
## sigma^2     0.3714  0.6094     10     no     obs
##
## Test for Heterogeneity:
## Q(df = 9) = 86.2787, p-val < .0001
##
## Model Results:
##
## estimate      se    zval    pval   ci.lb   ci.ub
##   0.8269  0.2251  3.6738  0.0002  0.3858  1.2681  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2018_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2018)
summary(MLMA_2018_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 29; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -33.3855   66.7711   70.7711   73.4355   71.2511
##
## Variance Components:
##
##              estim    sqrt  nlvls  fixed  factor
## sigma^2     0.0943  0.3071     29     no     obs
##
## Test for Heterogeneity:
```

```
## Q(df = 28) = 94.8064, p-val < .0001
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb   ci.ub
##  -0.1216  0.0723  -1.6813  0.0927  -0.2633  0.0202  .
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MLMA_2019_lnRR <- rma.mv(yi ~ 1, V = vi, random=~1|obs, data=lnRR2019)
summary(MLMA_2019_lnRR)
```

```
##
## Multivariate Meta-Analysis Model (k = 4; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
##   0.1600   -0.3201    3.6799    1.8771   15.6799
##
## Variance Components:
##
##             estim    sqrt  nlvls  fixed  factor
## sigma^2    0.0000  0.0000      4     no     obs
##
## Test for Heterogeneity:
## Q(df = 3) = 1.1718, p-val = 0.7598
##
## Model Results:
##
## estimate      se     zval    pval    ci.lb   ci.ub
##  -0.0094  0.1357  -0.0696  0.9445  -0.2753  0.2564
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

##set prior

```
prior <- list(R=list(V = 1, nu =0.002), G = list(G = list(V=1, nu = 0.002)))
```

##run bayesian MLMA models

```
model_magnitude_bayes_2009 <- MCMCglmm(yi ~ 1, mev = lnRR2009clean$vi, random
= ~obs, data = lnRR2009clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2010 <- MCMCglmm(yi ~ 1, mev = lnRR2010clean$vi, random
= ~obs, data = lnRR2010clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2011 <- MCMCglmm(yi ~ 1, mev = lnRR2011clean$vi, random
= ~obs, data = lnRR2011clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2012 <- MCMCglmm(yi ~ 1, mev = lnRR2012clean$vi, random
```

```
= ~obs, data = lnRR2012clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2013 <- MCMCglmm(yi ~ 1, mev = lnRR2013clean$vi, random
= ~obs, data = lnRR2013clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2014 <- MCMCglmm(yi ~ 1, mev = lnRR2014clean$vi, random
= ~obs, data = lnRR2014clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
#model_magnitude_bayes_2015 <- MCMCglmm(yi ~ 1, mev = lnRR2015clean$vi,
random = ~obs, data = lnRR2015clean, prior = prior, burnin = 10000, nitt =
1000000, thin = 100, verbose = FALSE)
model_magnitude_bayes_2016 <- MCMCglmm(yi ~ 1, mev = lnRR2016clean$vi, random
= ~obs, data = lnRR2016clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2017 <- MCMCglmm(yi ~ 1, mev = lnRR2017clean$vi, random
= ~obs, data = lnRR2017clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2018 <- MCMCglmm(yi ~ 1, mev = lnRR2018clean$vi, random
= ~obs, data = lnRR2018clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
model_magnitude_bayes_2019 <- MCMCglmm(yi ~ 1, mev = lnRR2019clean$vi, random
= ~obs, data = lnRR2019clean, prior = prior, burnin = 10000, nitt = 1000000,
thin = 100, verbose = FALSE)
```

##get model summaries

```
summary(model_magnitude_bayes_2009)

##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 30.14551
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs     2.612 0.0002158    7.538     3619
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units     2.833 0.0002297    7.608     3377
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)    1.2618   0.1972   2.3481    10191 0.0216 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model_magnitude_bayes_2010)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 55.95079
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs     1.458 0.0002328    3.587     1595
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units     1.479 0.0001946    3.542     1660
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)     1.610    1.124    2.102     9900 <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model_magnitude_bayes_2011)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: -34.69787
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs   0.02702 0.0001686  0.08106     8401
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units   0.02736 0.0001958  0.08292     9067
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp   pMCMC
```

```
## (Intercept)  -0.21406 -0.34688 -0.07854     8921 0.00545 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model_magnitude_bayes_2012)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 33.5066
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs     1.642 0.0002374    4.565     2643
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units     1.632 0.0002128    4.543     2562
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)    0.1656  -0.5925   0.9548     9900  0.66
```

```
summary(model_magnitude_bayes_2013)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 13.25863
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs    0.8996 0.0001951    2.851     6536
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units    0.8971 0.000252    2.902     7306
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp    pMCMC
## (Intercept)   -1.7772  -2.6964  -0.7794     9900 0.000808 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model_magnitude_bayes_2014)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 52.42814
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs     2.872 0.0001745    7.353     1823
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units     2.999 0.0002072    7.523     1864
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)    0.2438  -0.5995   1.0414     9900 0.543
```

```
#summary(model_magnitude_bayes_2015)
summary(model_magnitude_bayes_2016)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: 2.944739
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs    0.8854 0.0002583    3.237     9381
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units    0.8601 0.000147    3.164     9900
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)    0.5423  -0.5045   1.6582     9180 0.254
```

```r
summary(model_magnitude_bayes_2017)
```

```
## 
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900 
## 
##  DIC: -0.2582832 
## 
##  G-structure:  ~obs
## 
##     post.mean  l-95% CI u-95% CI eff.samp
## obs    0.2626 0.0002499   0.8991     9480
## 
##  R-structure:  ~units
## 
##       post.mean  l-95% CI u-95% CI eff.samp
## units    0.2599 0.0001939   0.8564     8936
## 
##  Location effects: yi ~ 1 
## 
##             post.mean l-95% CI u-95% CI eff.samp   pMCMC   
## (Intercept)    0.8235   0.3264   1.3445     9900 0.00848 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model_magnitude_bayes_2018)
```

```
## 
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900 
## 
##  DIC: -32.96555 
## 
##  G-structure:  ~obs
## 
##     post.mean  l-95% CI u-95% CI eff.samp
## obs   0.06186 0.0001632   0.2262     8768
## 
##  R-structure:  ~units
## 
##       post.mean  l-95% CI u-95% CI eff.samp
## units   0.06072 0.0001726   0.2272     9126
## 
##  Location effects: yi ~ 1 
## 
##             post.mean l-95% CI u-95% CI eff.samp  pMCMC  
## (Intercept)   -0.1249  -0.3100   0.0317     9900 0.0954 .
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model_magnitude_bayes_2019)
```

```
##
##  Iterations = 10001:999901
##  Thinning interval  = 100
##  Sample size  = 9900
##
##  DIC: -10.31207
##
##  G-structure:  ~obs
##
##     post.mean  l-95% CI u-95% CI eff.samp
## obs    0.1859 0.0001657   0.3318     9900
##
##  R-structure:  ~units
##
##       post.mean  l-95% CI u-95% CI eff.samp
## units    0.1357 0.0001709   0.3269     9900
##
##  Location effects: yi ~ 1
##
##             post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)   -0.0433  -0.5270   0.4424     9900 0.854
```

##extract posteriors

```
sol2009 <- model_magnitude_bayes_2009$Sol
VCV2009 <- model_magnitude_bayes_2009$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2009$VCV))]
sol2010 <- model_magnitude_bayes_2010$Sol
VCV2010 <- model_magnitude_bayes_2010$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2010$VCV))]
sol2011 <- model_magnitude_bayes_2011$Sol
VCV2011 <- model_magnitude_bayes_2011$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2011$VCV))]
sol2012 <- model_magnitude_bayes_2012$Sol
VCV2012 <- model_magnitude_bayes_2012$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2012$VCV))]
sol2013 <- model_magnitude_bayes_2013$Sol
VCV2013 <- model_magnitude_bayes_2013$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2013$VCV))]
sol2014 <- model_magnitude_bayes_2014$Sol
VCV2014 <- model_magnitude_bayes_2014$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2014$VCV))]
#sol2015 <- model_magnitude_bayes_2015$Sol
#VCV2015 <- model_magnitude_bayes_2015$VCV[,-
match("sqrt(mev):sqrt(mev).meta", colnames(model_magnitude_bayes_2015$VCV))]
sol2016 <- model_magnitude_bayes_2016$Sol
```

```
VCV2016 <- model_magnitude_bayes_2016$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2016$VCV))]
sol2017 <- model_magnitude_bayes_2017$Sol
VCV2017 <- model_magnitude_bayes_2017$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2017$VCV))]
sol2018 <- model_magnitude_bayes_2018$Sol
VCV2018 <- model_magnitude_bayes_2018$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2018$VCV))]
sol2019 <- model_magnitude_bayes_2019$Sol
VCV2019 <- model_magnitude_bayes_2019$VCV[,-match("sqrt(mev):sqrt(mev).meta",
colnames(model_magnitude_bayes_2019$VCV))]
```

##get folded normal function

```
mu.fnorm <- function(mu, sigma){dnorm(mu, 0, sigma)*2*sigma^2 +
mu*(2*pnorm(mu, 0, sigma) -1)}
```

##get magnitude means + variance

```
magnitude_mean_2009 <- mu.fnorm(sol2009[,1], sqrt(rowSums(VCV2009)))
magnitude_2009_mean <- data.frame(mean_mag = mean(magnitude_mean_2009), L_CI
= HPDinterval(magnitude_mean_2009)[1], U_CI =
HPDinterval(magnitude_mean_2009)[2])
magnitude_mean_2010 <- mu.fnorm(sol2010[,1], sqrt(rowSums(VCV2010)))
magnitude_2010_mean <- data.frame(mean_mag = mean(magnitude_mean_2010), L_CI
= HPDinterval(magnitude_mean_2010)[1], U_CI =
HPDinterval(magnitude_mean_2010)[2])
magnitude_mean_2011 <- mu.fnorm(sol2011[,1], sqrt(rowSums(VCV2011)))
magnitude_2011_mean <- data.frame(mean_mag = mean(magnitude_mean_2011), L_CI
= HPDinterval(magnitude_mean_2011)[1], U_CI =
HPDinterval(magnitude_mean_2011)[2])
magnitude_mean_2012 <- mu.fnorm(sol2012[,1], sqrt(rowSums(VCV2012)))
magnitude_2012_mean <- data.frame(mean_mag = mean(magnitude_mean_2012), L_CI
= HPDinterval(magnitude_mean_2012)[1], U_CI =
HPDinterval(magnitude_mean_2012)[2])
magnitude_mean_2013 <- mu.fnorm(sol2013[,1], sqrt(rowSums(VCV2013)))
magnitude_2013_mean <- data.frame(mean_mag = mean(magnitude_mean_2013), L_CI
= HPDinterval(magnitude_mean_2013)[1], U_CI =
HPDinterval(magnitude_mean_2013)[2])
magnitude_mean_2014 <- mu.fnorm(sol2014[,1], sqrt(rowSums(VCV2014)))
magnitude_2014_mean <- data.frame(mean_mag = mean(magnitude_mean_2014), L_CI
= HPDinterval(magnitude_mean_2014)[1], U_CI =
HPDinterval(magnitude_mean_2014)[2])
#magnitude_mean_2015 <- mu.fnorm(sol2015[,1], sqrt(rowSums(VCV2015)))
#magnitude_2015_mean <- data.frame(mean_mag = mean(magnitude_mean_2015), L_CI
= HPDinterval(magnitude_mean_2015)[1], U_CI =
HPDinterval(magnitude_mean_2015)[2])
magnitude_mean_2016 <- mu.fnorm(sol2016[,1], sqrt(rowSums(VCV2016)))
magnitude_2016_mean <- data.frame(mean_mag = mean(magnitude_mean_2016), L_CI
= HPDinterval(magnitude_mean_2016)[1], U_CI =
HPDinterval(magnitude_mean_2016)[2])
```

```
magnitude_mean_2017 <- mu.fnorm(sol2017[,1], sqrt(rowSums(VCV2017)))
magnitude_2017_mean <- data.frame(mean_mag = mean(magnitude_mean_2017), L_CI
= HPDinterval(magnitude_mean_2017)[1], U_CI =
HPDinterval(magnitude_mean_2017)[2])
magnitude_mean_2018 <- mu.fnorm(sol2018[,1], sqrt(rowSums(VCV2018)))
magnitude_2018_mean <- data.frame(mean_mag = mean(magnitude_mean_2018), L_CI
= HPDinterval(magnitude_mean_2018)[1], U_CI =
HPDinterval(magnitude_mean_2018)[2])
magnitude_mean_2019 <- mu.fnorm(sol2019[,1], sqrt(rowSums(VCV2019)))
magnitude_2019_mean <- data.frame(mean_mag = mean(magnitude_mean_2019), L_CI
= HPDinterval(magnitude_mean_2019)[1], U_CI =
HPDinterval(magnitude_mean_2019)[2])
```

##view ES magnitudes and uncertainty

```
magnitude_2009_mean

##   mean_mag     L_CI     U_CI
## 1 2.149148 1.481714 2.927198

magnitude_2010_mean

##   mean_mag     L_CI     U_CI
## 1 1.938874 1.599878 2.328271

magnitude_2011_mean

##    mean_mag      L_CI      U_CI
## 1 0.2660906 0.1794605 0.3631261

magnitude_2012_mean

##   mean_mag     L_CI     U_CI
## 1 1.460949 1.003474 1.992196

magnitude_2013_mean

##   mean_mag     L_CI     U_CI
## 1 1.929857 1.150945 2.710492

magnitude_2014_mean

##   mean_mag     L_CI     U_CI
## 1  1.95539 1.483953 2.448358

#magnitude_2015_mean
magnitude_2016_mean

##   mean_mag      L_CI     U_CI
## 1  1.14367 0.5008537 2.053592

magnitude_2017_mean
```

```
##    mean_mag      L_CI     U_CI
## 1 0.9385116 0.5663024 1.350452
```

```
magnitude_2018_mean
```

```
##    mean_mag       L_CI      U_CI
## 1 0.2829174 0.07055759 0.5008954
```

```
magnitude_2019_mean
```

```
##    mean_mag       L_CI      U_CI
## 1 0.3119009 0.03263849 0.7566612
```

#Note Code below not in original. Made to allow a data frame to be constructed and plotted from.

```
magnitudedata = rbind(magnitude_2009_mean, magnitude_2010_mean,
magnitude_2011_mean, magnitude_2012_mean, magnitude_2013_mean,
magnitude_2014_mean, NA, magnitude_2016_mean, magnitude_2017_mean,
magnitude_2018_mean, magnitude_2019_mean)
```

```
yearlabel = c("2009", "2010", "2011", "2012", "2013", "2014", "2015", "2016",
"2017", "2018", "2019")
```

```
magnitudedata = cbind(yearlabel, magnitudedata )
magnitudedata
```

```
##    yearlabel  mean_mag       L_CI      U_CI
## 1       2009 2.1491484 1.48171404 2.9271977
## 2       2010 1.9388739 1.59987832 2.3282707
## 3       2011 0.2660906 0.17946050 0.3631261
## 4       2012 1.4609486 1.00347369 1.9921959
## 5       2013 1.9298566 1.15094518 2.7104917
## 6       2014 1.9553900 1.48395301 2.4483581
## 7       2015        NA         NA        NA
## 8       2016 1.1436702 0.50085366 2.0535923
## 9       2017 0.9385116 0.56630242 1.3504522
## 10      2018 0.2829174 0.07055759 0.5008954
## 11      2019 0.3119009 0.03263849 0.7566612
```

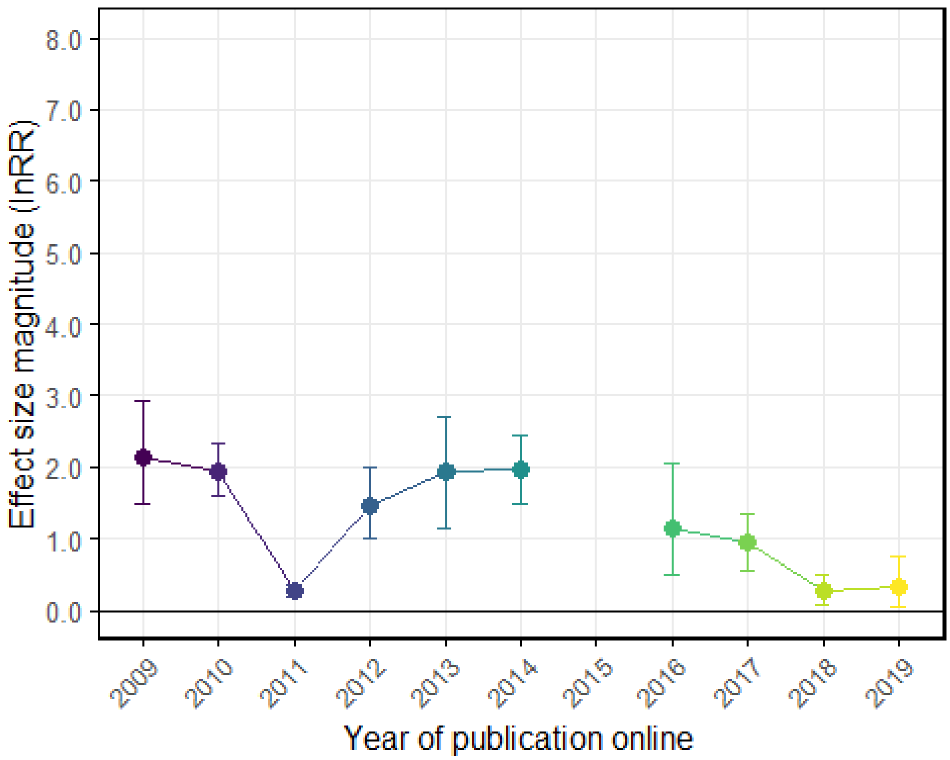
#Note Plot from above model was not included in the original code. This has been built from scratch. Should be the same as Fig 1B

```
Decline_magnitude<-ggplot(magnitudedata,aes(x=yearlabel, y=mean_mag,
colour=yearlabel))  + geom_line(aes(group=1)) +
scale_color_viridis(discrete=TRUE)+ geom_point(size=3) +  geom_errorbar(aes
(ymin = L_CI, ymax = U_CI), width=0.2) +  scale_y_continuous(breaks = seq(0,
15, by = 1), minor_breaks = NULL, limits=c(0,8),labels =
scales::number_format(accuracy = 0.1)) +  theme(legend.position = "none") +
xlab("Year of publication online") + ylab("Effect size magnitude (lnRR)")  +
theme_minimal(12) + theme(panel.border = element_rect(colour = "black",
fill=NA, size=1)) + theme(legend.position = "none") +
```

```
theme(axis.ticks=element_line()) + theme(axis.text.x = element_text(angle =
45, hjust=1))+ geom_hline(yintercept = 0)

Decline_magnitude

## Warning: Removed 1 rows containing missing values (geom_point).
```




```
ggsave(Decline_magnitude, filename = 'Decline magnitude original corrected
olfaction risk and habitat model 1 as lowest lnRR.png', device=png,  width =
4.2, height = 4.6, units = "in", res = 800)

## Warning: Removed 1 rows containing missing values (geom_point).
```

CREATE SCATTERPLOT FIGURE TO VISUALIZE MEAN EFFECT SIZE MAGNITUDE FOR EACH OBSERVATION OVER TIME (FIG 1A)

##attach dataset

```
decline_allobs<-read.csv(file.choose()) ##use dataset "S10 Data"
attach(decline_allobs)

## The following objects are masked from decline:
##
##     ctrl.n, obs, study, year.online, year.print

summary(decline_allobs)
```

```
##       obs            study              year.online     year.print  
##  Min.   :  1.0   Length:839         Min.   :2009   Min.   :2009  
##  1st Qu.:210.5   Class :character   1st Qu.:2012   1st Qu.:2013  
##  Median :420.0   Mode  :character   Median :2015   Median :2015  
##  Mean   :420.0                      Mean   :2015   Mean   :2015  
##  3rd Qu.:629.5                      3rd Qu.:2017   3rd Qu.:2018  
##  Max.   :839.0                      Max.   :2019   Max.   :2019  
##                                                                 
##      ctrl.n          lnrr.mag     
##  Min.   :  3.00   Min.   :0.0000  
##  1st Qu.: 10.00   1st Qu.:0.1858  
##  Median : 18.00   Median :0.7364  
##  Mean   : 28.77   Mean   :1.4228  
##  3rd Qu.: 30.00   3rd Qu.:2.2179  
##  Max.   :752.00   Max.   :5.5936  
##                   NA's   :615
```

##Create plot

#Note I had to fix the specifications of the scale continuous both x and y Original below not working: scale_x_continuous(breaks = round(seq(min(study$Year), max(study$Year), by = 1),1)) + scale_y_continuous(breaks = round(seq(min(study$lnrr), 15, by = 1),1)) #Note Code given did not produce a graph that looked exactly the same as in the figures of the paper. Changes to aesthetics were add to make it look the same as the published version.

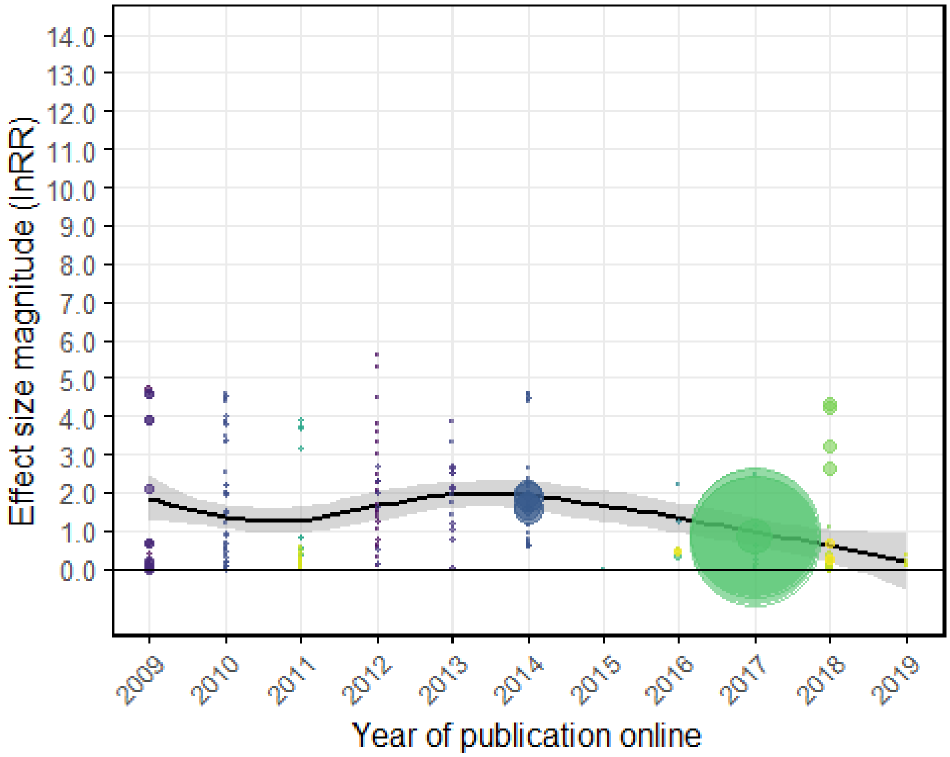
```
Decline_studies_loess<-ggplot(decline_allobs,aes(x=year.online, y=lnrr.mag,
color=study)) + geom_smooth(method="loess", se=TRUE, fullrange=TRUE,
level=0.95,color="black") + geom_point(size=ctrl.n*0.03,alpha=0.6) +
scale_size(range = c(1, 2), name="Sample size")+
scale_color_viridis(discrete=TRUE)+ xlab("Year of publication
online")+ylab("Effect size magnitude (lnRR)") + scale_x_continuous(breaks =
round(seq(min(2009), max(2019), by = 1),1)) + scale_y_continuous(breaks =
seq(0, 14, by = 1), minor_breaks = NULL, limits=c(-1,14),labels =
scales::number_format(accuracy = 0.1)) + theme_minimal(12) +
theme(legend.position = "none") + theme(panel.grid.minor = element_blank())
+ theme(panel.border = element_rect(colour = "black", fill=NA, size=1)) +
theme(legend.position = "none")+ theme(axis.ticks=element_line()) +
theme(axis.text.x = element_text(angle = 45, hjust=1))+ geom_hline(yintercept
= 0)

Decline_studies_loess

## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 615 rows containing non-finite values (stat_smooth).

## Warning: Removed 615 rows containing missing values (geom_point).
```




```
ggsave(Decline_studies_loess, filename = 'Decline magnitude original
corrected olfaction risk and habitat 1 as lowest lnRR.png', device=png,
width = 4.2, height = 4.6, units = "in", res = 800)

## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 615 rows containing non-finite values (stat_smooth).

## Warning: Removed 615 rows containing missing values (geom_point).
```