

A meta-analysis of sex differences in animal personality: no evidence for greater male variability

Lauren Harrison, Daniel Noble & Michael Jennions

last updated: 16/09/2021

Table of Contents

Supplementary Material - Data checks, analysis and meta-a models.....	2
Data Set-up.....	2
Load packages	2
Organising data for analysis	3
Calculating effect sizes	4
means - SMD using Hedges' g and variability - lnCVR.....	4
Data Checks.....	4
mean-variance relationship in our dataset	4
Checking for outliers and removing weird effect sizes	6
Flip signs of effects for SMD.....	10
Prepare the phylogenetic trees	10
Sensitivity analysis - checking score data.....	12
Meta-Analysis Models.....	13
Intercept-only MLMA models	13
I2 estimates of heterogeneity - intercept models	13
Personality trait MLMR models.....	15
Personality trait x SSD MLMR models	22
SSD subset models.....	27
NEED TO DROP:.....	27
Mammals.....	28
Birds	35
Fish.....	37
Inverts	41
Multiple testing.....	45
Exploratory analyses.....	47
mating system	47

age	48
population	49
study environment.....	50
study type	50
Sensitivity analyses - Dependency matrix models.....	58
Intercept-only models with D matrices	59
Personality trait models with D matrices	60
Personality * SSD models.....	61
Sensitivity analyses - Publication Bias	62
Exploratory analyses - Heterogamety and taxo group.....	63
Plots.....	64
Orchard plots of effect sizes from personality trait models.....	64
lnCVR	64
SMD.....	65
Phylogenetic trees with heatmaps	69
bird tree.....	70
mammals.....	72
reptiles	75
fish.....	77
inverts.....	79

Supplementary Material - Data checks, analysis and meta-a models

Data Set-up

Load packages

```
# Clear work space
# rm(list=ls())

# Install CRAN packages
library("pacman")

## Warning: package 'pacman' was built under R version 3.5.2
```

```

# Install orchard plot and metaAidR packages from GitHub
devtools::install_github("itchyshin/orchard_plot", subdir = "orchaRd",
force = TRUE, build_vignettes = TRUE);

devtools::install_github("daniellnoble/metaAidR");

pacman::p_load(knitr, metafor, dplyr, kableExtra, tidyverse, rotl, phytools,
GGally, R.rsp, patchwork, devtools, robumeta, ape, geiger, phytools,
phangorn, rlist, orchaRd, metaAidR, corrplot, stringr)

# set working directory
# setwd("~/Documents/GitHub/sex_meta/")

# Source our own functions
source("../R/func.R")

# Set the rerun object to FALSE so that you don't need to re-run all models
again. Some take quite a lot of time to run. If FALSE, it will just re-load
saved output.

rerun_models = FALSE

```

Organising data for analysis

```

## load our original pers dataset and our dataset with all of our sexual
selection info
pers <- read.csv("../data/pers_data.csv", stringsAsFactors = FALSE)
bodysize <- read.csv("../data/bodysize_SSD.csv", stringsAsFactors =
FALSE)

## Merge the two by spp_names columns
pers <- merge(x = pers,
              y = bodysize[,c("species_name", "SSD_index",
"mating_system")],
              by="species_name", all.x=TRUE, no.dups = TRUE)

## Select only the relevant columns to make life easier
pers_new <- pers %>%
  select(study_ID, year, species_name, SSD_index, taxo_group,
data_type, personality_trait, male_n, male_mean_conv,
male_sd_conv, female_n, female_mean_conv, female_sd_conv, depend,
directionality, spp_name_phylo, mating_system,
age, population, study_environment, study_type, measurement_type)

## Add in observation level random effect (metafor doesn't do this, need to
do it manually)

```

```
pers_new <- pers_new %>%
  group_by(taxo_group) %>%
  mutate(obs = 1:length(study_ID))
```

Calculating effect sizes

means - SMD using Hedges' g and variability - lnCVR

```
## SMD (Hedges' g)
pers_new <- escalc(measure = "SMD",
  n1i = male_n, n2i = female_n,
  m1i = male_mean_conv, m2i = female_mean_conv,
  sd1i = male_sd_conv, sd2i = female_sd_conv, data =
pers_new, var.names=c("SMD_yi", "SMD_vi"), append = TRUE)

## lnCVR
pers_new <- escalc(measure = "CVR",
  n2i = female_n, n1i = male_n,
  m2i = female_mean_conv, m1i = male_mean_conv,
  sd2i = female_sd_conv, sd1i = male_sd_conv, data =
pers_new, var.names=c("CVR_yi", "CVR_vi"))

# we have some NAs where one or both sexes have a value of 0 for either
mean or SD. Will be easiest to just remove these.

# Exclude NAs
pers_new <- pers_new %>%
  filter(!is.na(CVR_yi), !is.na(SMD_yi))

dim(pers_new) # check they've been removed with no issues
```

Data Checks

mean-variance relationship in our dataset

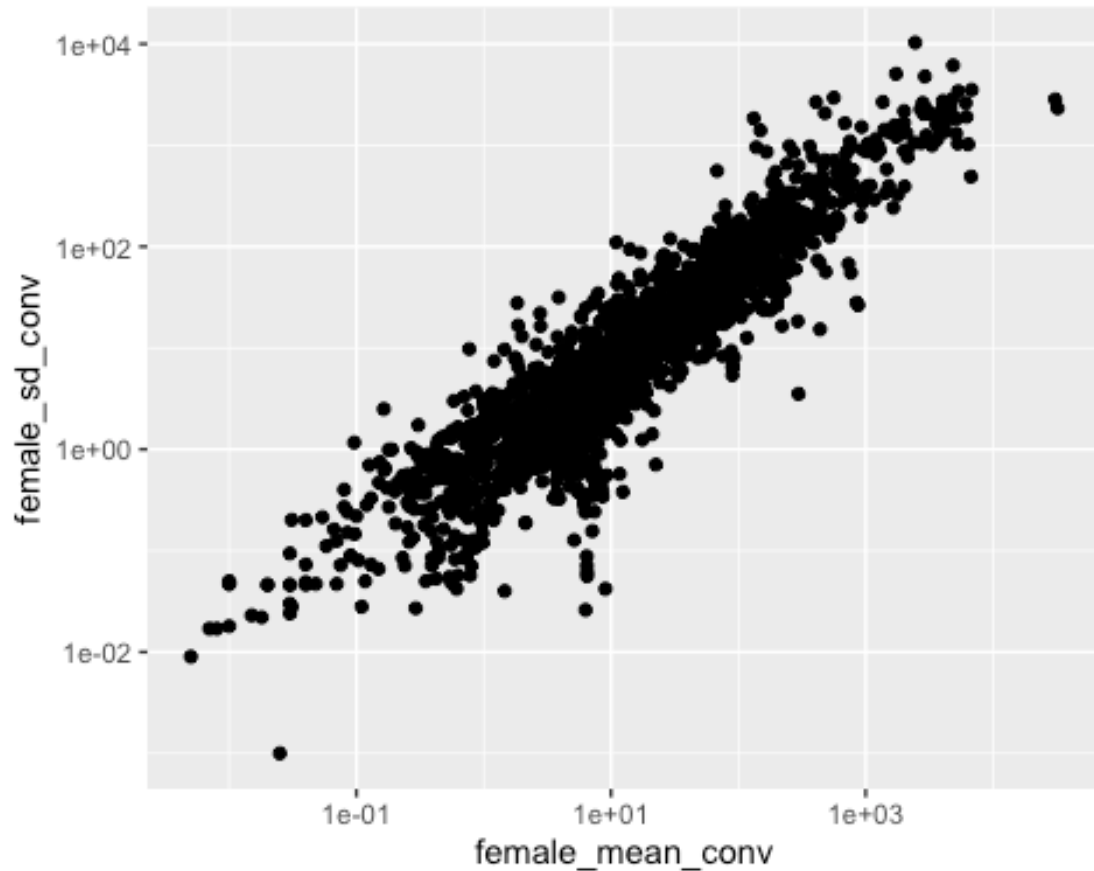
Looking at the strength of the correlation between the mean and SD to check that using lnCVR as a measure of variability is valid. If the mean and SD are NOT strongly correlated then using lnCVR is pointless.

```
# females and males seperately because they are in different columns

# use ggplot to make a scatterplot of females
fem <- ggplot(pers_new, aes(x = female_mean_conv, y = female_sd_conv)) +
  geom_point()

# on log scale
```

```
fem + scale_x_continuous(trans = 'log10') + scale_y_continuous(trans = 'log10')
```



```
# mean and SD on Log scale to calculate correlation
```

```
logfemale_mean <- log(pers_new$female_mean_conv)
```

```
logfemale_SD <- log(pers_new$female_sd_conv)
```

```
# correlation between mean and SD
```

```
cor(logfemale_mean, logfemale_SD) #0.91
```

```
## [1] 0.9182668
```

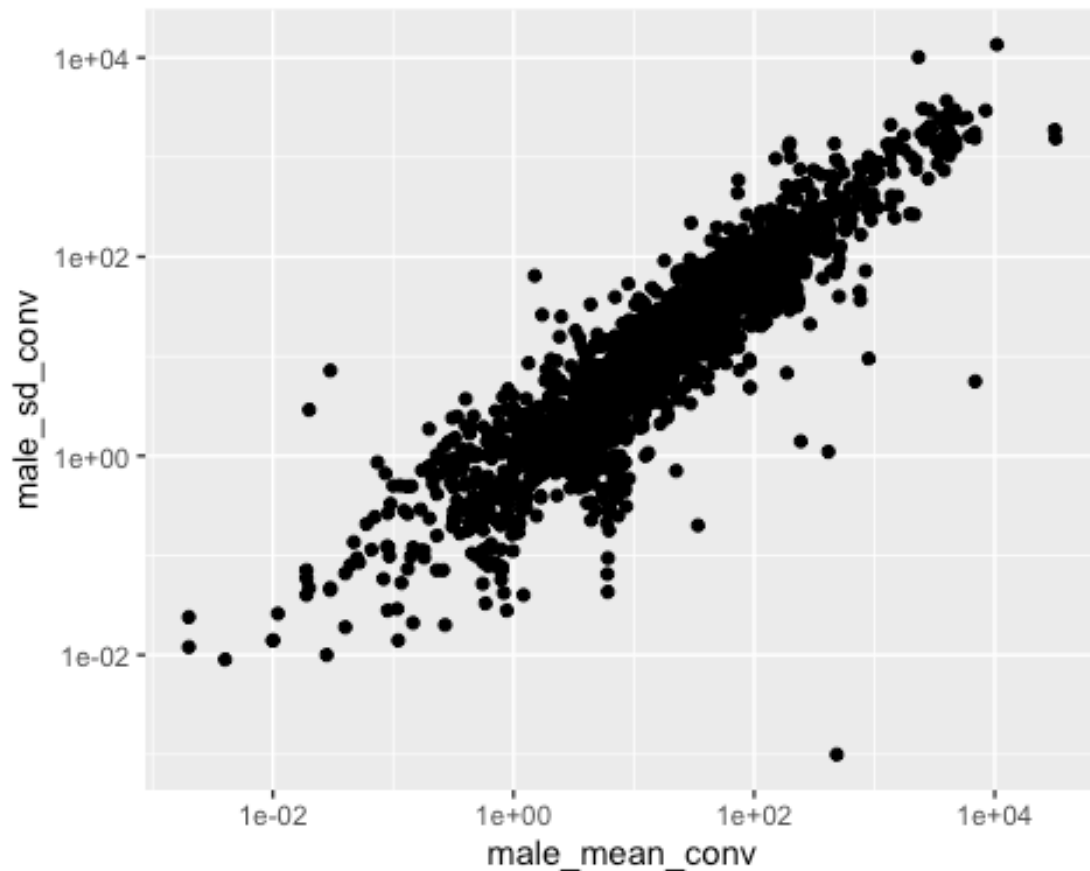
```
# Males
```

```
# use ggplot to make a scatterplot of females
```

```
male <- ggplot(pers_new, aes(x = male_mean_conv, y = male_sd_conv)) +  
geom_point()
```

```
# on Log scale
```

```
male + scale_x_continuous(trans = 'log10') + scale_y_continuous(trans = 'log10')
```



```
# mean and SD on log scale to calculate correlation
logmale_mean <- log(pers_new$male_mean_conv)
logmale_SD <- log(pers_new$male_sd_conv)
```

```
# correlation between mean and SD
cor(logmale_mean, logmale_SD) #0.90
```

```
## [1] 0.9071225
```

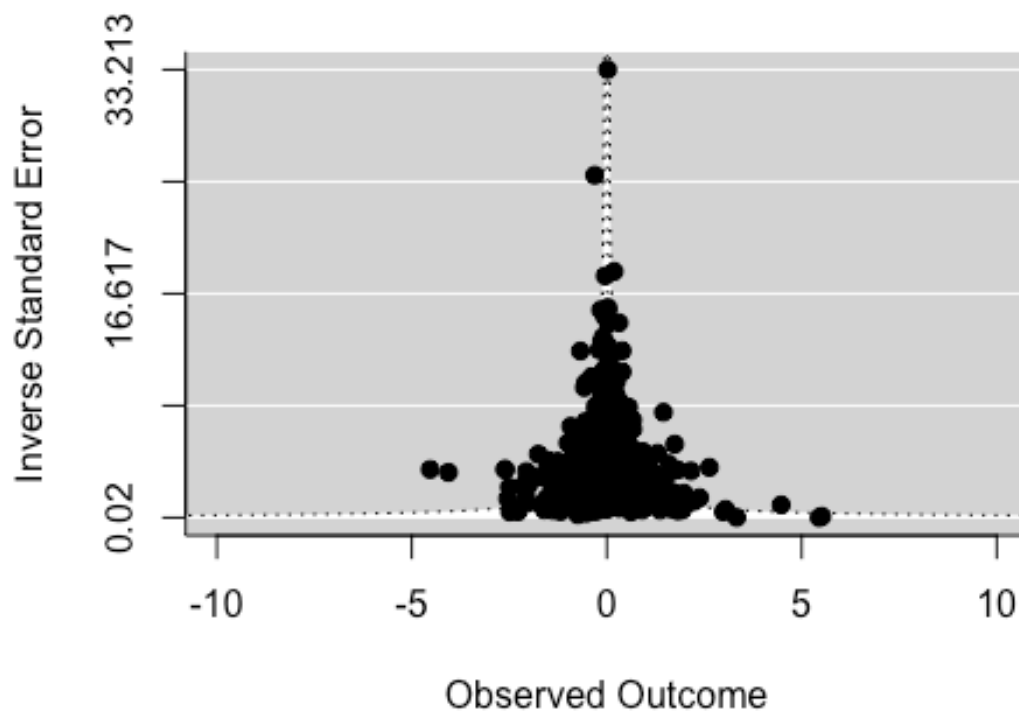
Checking for outliers and removing weird effect sizes

This is an important data checking step - here we can identify whether data has been entered or reported incorrectly (i.e. outliers)

First, let's look at the funnel plots for lnCVR and SMD. NOTE: these funnels have had our 2 big outliers already removed.

```
#funnel plot for lnCVR
```

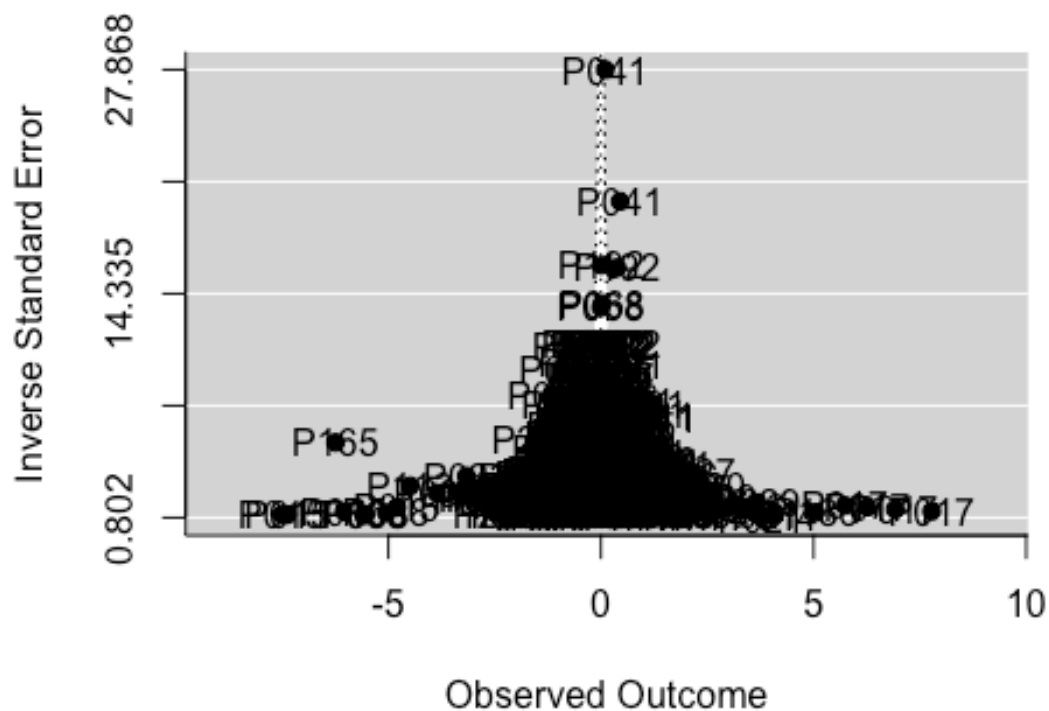
```
funnel(x = pers_new$CVR_yi, vi = pers_new$CVR_vi, yaxis="seinv", xlim =
c(-10, 10))
```



SMD

#funnel plot for SMD

```
funnel(x = pers_new$SMD_yi, vi = pers_new$SMD_vi, yaxis="seinv")
text(as.character(pers_new$study_ID), x = pers_new$SMD_yi, y =
1/sqrt(pers_new$SMD_vi))
```



Removing outliers

```
pers_new %>%
  filter(study_ID == "P015" & SMD_yi < -15) # P015 has 1 large effect size,
remove?

# filter out that large effect size
pers_new <- pers_new %>%
  filter(!study_ID == "P015" | !obs == "509")

dim(pers_new)

# checking SMD outliers - inverse SE > 14
funnel(x = pers_new$SMD_yi, vi = pers_new$SMD_vi, yaxis="seinv")
text(as.character(pers_new$obs), x = pers_new$SMD_yi, y =
1/sqrt(pers_new$SMD_vi), offset = 0.8)

# checking lnCVR outliers
funnel(x = pers_new$CVR_yi, vi = pers_new$CVR_vi, yaxis="seinv", xlim =
c(-10, 10))
text(as.character(pers_new$obs), x = pers_new$CVR_yi, y =
1/sqrt(pers_new$CVR_vi), offset = 0.8)
```



```

# Some measures are more physiological/not personality than personality, so
# probably wise to remove these before we run the models:
# P029 - obs 22, 23, 32
# P084 - obs 59, 62, 63, 65, 68, 70, 71, 72, 74
# P060 - obs 216, 217
# P211 - obs 230, 245
# P117 - obs 397, 393, 402, 414
# P197 - obs 541, 544, 546, 547
# P069 - obs 669, 672, 673, 682, 683, 684, 686, 694

# remove these by study
pers_new <- pers_new %>% filter(!study_ID == "P029" | !obs %in% c("21",
"25", "26", "28", "31"))

pers_new <- pers_new %>% filter(!study_ID == "P084" | !obs %in% c("59",
"62", "63", "65", "68", "70", "71", "72", "74"))

pers_new <- pers_new %>% filter(!study_ID == "P060" | !obs %in% c("216",
"217"))

pers_new <- pers_new %>% filter(!study_ID == "P211" | !obs %in% c("230",
"245"))

pers_new <- pers_new %>% filter(!study_ID == "P117" | !obs %in% c("397",
"393", "402", "414"))

pers_new <- pers_new %>% filter(!study_ID == "P197" | !obs %in% c("541",
"544", "546", "547"))

pers_new <- pers_new %>% filter(!study_ID == "P197" | !obs %in% c("669",
"672", "673", "682", "683", "684", "686", "694"))
pers_new <- pers_new %>% filter(!study_ID == "P041" | !obs %in% c("120",
"124"))

dim(pers_new) # check they've been removed without issue

# after checking the data, there are a few effect sizes that might be
# driving weird results so let's drop them and see

pers_new <- pers_new %>% filter(!study_ID == "P100" | !obs == "519") #
big outlier

```

Flip signs of effects for SMD

The directional meaning of effect sizes vary depending on the specific units and trait being measured. The data has a directionality column that tells one if the meaning should be reversed (1) or left the same.

```
pers_new$directionality <- ifelse(is.na(pers_new$directionality), 0, 1)

pers_new$SMD_yi_flip <- ifelse(pers_new$directionality == 1,
pers_new$SMD_yi*(-1), pers_new$SMD_yi)
```

Prepare the phylogenetic trees

We constructed separate phylogenetic trees for each taxonomic group. The tree for birds was constructed using BirdTree.org, the rest were constructed using TimeTree.org. We'll use these trees for multi-level meta-analytic models throughout the analysis.

```
# Find all tree file names
tree_files <- paste0("./trees/", list.files("./trees"))[-1]

# Bird tree has been constructed already, just need to get trees for the
rest of the taxo groups
trees <- lapply(tree_files, function(x) read.tree(x))
names <- gsub("~/trees/", "", tree_files)
names(trees) <- names

# Plot the trees and see how they look
par(mfrow = c(1,5), mar = c(1,1,1,1))
lapply(trees, function(x) plot(x, cex = 1))
```

```

Cereopsis novaehollandiae
Cyprinus carpio
Danio rerio
Neogobius melanostomus
Gasterosteus aculeatus
Betta splendens
Pelmatochromis nigrofasciatus
Oreochromis tanganicae
Neolamprologus caudopunctatus
Neolamprologus pulcher
Hypsophrys nicaraguensis
Amphiphys citrinellus
Amatitlania nigrofasciata
Aphanius iberus
Aphanius fasciatus
Poecilia reticulata
Brachyrhaphis episcopi
Brachyrhaphis rhabdophora
Gambusia affinis
Gambusia nicaraguensis
Xiphophorus birchmanni
Xiphophorus furzeri
Heterodontus francisci

Euprymna scolopes
Eris militaris
Tegenaria atrica
Parasteatoda tepidariorum
Zygiella x-notata
Acheta domestica
Gryllus bimaculatus
Diploptera punctata
Stagmomantis limbata
Mantis religiosa
Gerris marginatus
Aquarius remigis
Pararge aegeria
Drosophila melanogaster
Drosophila prolongata
Drosophila kurseongensis
Sitophilus granarius
Chrysolina hyperici
Callosobruchus maculatus
Callosobruchus chinensis
Tenebrio molitor
Tribolium confusum
Tribolium castaneum
Carabus nemoralis
Pterostichus niger
Percus politus
Leistus spinibarbis
Ischnura elegans
Palaemon kadiakensis
Panopeus herbstii
Acanthocyclus hassleri
Carcinus maenas
Eriocheir sinensis
Homarus gammarus
Cherax destructor
Orconectes virilis
Procambarus clarkii
Gammarus fossarum
Cirolana rugicauda
Sphaeromides virei
Cymodoce tattersalli
Asellus aquaticus

Pan troglodytes
Chlorocebus pygerythrus
Macaca arctoides
Macaca assamensis
Theropithecus gelada
Ateles hybridus
Ateles geoffroyi
Callithrix geoffroyi
Saguinus oedipus
Plecturocebus moloch
Microcebus murinus
Ictidomys tridecemlineatus
Tamas striatus
Tamas sibiricus
Pteromys volans
Sciurus vulgaris
Nannospalax ehrenbergi
Eospalax smithii
Peromyscus maniculatus
Peromyscus californicus
Lasiopodomys mandarinus
Microtus guentheri
Microtus arvalis
Microtus ochrogaster
Phodopus sungorus
Mus musculus
Mus spicilegus
Rhabdomys pumilio
Rhabdomys dilectus
Micromys minutus
Rattus norvegicus
Cavia aperea
Tadarida teniotis
Sus scrofa
Capreolus capreolus
Steno bredanensis
Tursiops aduncus
Tursiops truncatus
Ailuropoda melanoleuca
Mustela lutreola
Neovison vison
Canis latrans
Puma concolor
Crocuta crocuta
Macropus giganteus
Macropus agilis

Terrapene carolina
Lampropholis delicata
Amphibolurus muricatus
Liolaemus nitidus
Thamnophis sirtalis
Pantherophis obsoletus
Rhinella marina
Lissotriton boscai
Ensatina eschscholtzii
Plethodon cinereus

```

Checking trees to ensure we only include species in the current dataset

Check that they are ultrametric

```
lapply(trees, function(x) is.ultrametric(x))
```

Check that ALL names in the phylogeny are also in the data

```
taxa_data_list <- split(pers_new, pers_new$taxo_group)
```

```
other_groups <- map(x = taxa_data_list,
                    y = trees,
                    function(x,y) tree_checks(x,y, "spp_name_phylo",
type = "checks"))
```

Print out each taxon group

```
for(i in colnames(other_groups)){
  print(i)
  print(other_groups[,i] )
}
```

Now to prune trees so that we get tree names that match with species in data

```

pruned_trees <- mapply(x = taxa_data_list,
                      y = trees,
                      function(x,y) tree_checks(x,y, "spp_name_phylo",
type = "prune"))

# Check that this has been done correctly
re_checks <- mapply(x = taxa_data_list,
                  y = pruned_trees,
                  function(x,y) tree_checks(x,y, "spp_name_phylo",
type = "checks"))

for(i in colnames(re_checks)){
  print(i)
  print(re_checks[,i] )
}

# Extract the phylogenetic correlation matrices
phylo_vcv <- lapply(pruned_trees, function(x) vcv(x, corr = TRUE)) #
these matrices are used in the meta-a models

```

Sensitivity analysis - checking score data

Before we begin, we need to run a sensitivity analysis to see if score data is ok to use. With these models, we are just including score as a moderator term to compare with the rest of the dataset (some of which has already been transformed, we just can't do that with scores). Model summaries are also presented in Supplementary Table S2.

Our score sensitivity model:

```

# model:
sensitivity_mod1_score <- meta_model_fits(pers_new, phylo_vcv, type =
"score")

# Extract the SMD and lnCVR results
smd_mods_score <- sensitivity_mod1_score["SMD",]
lnCVR_mods_score <- sensitivity_mod1_score["lnCVR",] # inverts
significant

# Because invert score data is significantly different, we need to remove
these effect sizes before running our models

# filter out invert scores from pers dataset
pers_new <- pers_new %>%
  filter(score != "score" | taxo_group !=
"invertebrate")

dim(pers_new)

```

```
# How many effect sizes, unique studies and different species are we left with?
data.frame(pers_new %>%
  summarise(n = n(), studies =
length(unique(study_ID)), species = length(unique(spp_name_phylo))))
```

Meta-Analysis Models

Let's run the first bunch of models on the whole dataset. We'll start off with intercept-only multi-level meta-analytic models, then move to multi-level meta-regression models (personality traits, and SSD). The functions in `func.R` should be consulted to see precisely what models are being fit across the taxonomic groups.

Intercept-only MLMA models

Complete model summaries are also presented in Supplementary Table S14.

```
# First we will fit our MLMA intercept only models, across each taxo group.

# we can use this function to just read the saved model output instead of re-running the model, which takes a while
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMA_models <- meta_model_fits(pers_new, phylo_vcv, type = "int")
  saveRDS(MLMA_models, "./output/MLMA_models_int")
}else{
  MLMA_models <- readRDS("./output/MLMA_models_int")
}

# View model results
split_taxa <- split(pers_new, pers_new$taxo_group)

smd_mods <- MLMA_models["SMD",]

lnCVR_mods <- MLMA_models["lnCVR",]
```

I2 estimates of heterogeneity - intercept models

Study_ID is the between study heterogeneity, Phylo tells us if there is a phylogenetic signal and the strength of that signal. Total I2 is testing how much heterogeneity we have beyond sampling variance

```

# From these models we can get I2 estimates:
birds_smd = I2(smd_mods[[1]], v = split_taxa[[1]]$SMD_vi, phylo =
"spp_name_phylo", obs = "obs")

birds_CVR = I2(lnCVR_mods[[1]], v = split_taxa[[1]]$CVR_vi, phylo =
"spp_name_phylo", obs = "obs")

fish_smd = I2(smd_mods[[2]], v = split_taxa[[2]]$SMD_vi, phylo =
"spp_name_phylo", obs = "obs")

fish_CVR = I2(lnCVR_mods[[2]], v = split_taxa[[2]]$CVR_vi, phylo =
"spp_name_phylo", obs = "obs")

invert_smd = I2(smd_mods[[3]], v = split_taxa[[3]]$SMD_vi, phylo =
"spp_name_phylo", obs = "obs")

invert_CVR = I2(lnCVR_mods[[3]], v = split_taxa[[3]]$CVR_vi, phylo =
"spp_name_phylo", obs = "obs")

mammal_smd = I2(smd_mods[[4]], v = split_taxa[[4]]$SMD_vi, phylo =
"spp_name_phylo", obs = "obs")

mammal_CVR = I2(lnCVR_mods[[4]], v = split_taxa[[4]]$CVR_vi, phylo =
"spp_name_phylo", obs = "obs")

reptile_smd = I2(smd_mods[[5]], v = split_taxa[[5]]$SMD_vi, phylo =
"spp_name_phylo", obs = "obs")

reptile_CVR = I2(lnCVR_mods[[5]], v = split_taxa[[5]]$CVR_vi, phylo =
"spp_name_phylo", obs = "obs")

# Now that we have our list of models, we can extract the estimates, CIs and
prediction intervals
MLMA_estimates_SMD <- plyr::ldply(lapply(smd_mods, function(x)
print(mod_results(x, mod = "Int"))))
MLMA_estimates_SMD

##           .id      name      estimate      lowerCL      upperCL      lowerPR
upperPR
## 1           bird Intrcpt -0.11192796 -0.3499999 0.1261440 -1.8619378
1.6380819
## 2           fish Intrcpt  0.17680902 -0.2270172 0.5806353 -1.5472144
1.9008324
## 3  invertebrate Intrcpt  0.25283343 -0.1153844 0.6210513 -2.1177023
2.6233692
## 4           mammal Intrcpt  0.06839391 -0.2982885 0.4350763 -1.1341419
1.2709297

```

```
## 5      reptilia Intrcpt  0.06381860 -0.1145932 0.2422304 -0.5626372
0.6902744

MLMA_estimates_lnCVR <- plyr::ldply(lapply(lnCVR_mods, function(x)
print(mod_results(x, mod = "Int"))))
MLMA_estimates_lnCVR
```

	.id	name	estimate	lowerCL	upperCL	lowerPR
## 1	bird	Intrcpt	-0.138856029	-0.63322342	0.35551136	-1.88286875
## 2	fish	Intrcpt	-0.004621130	-0.08649739	0.07725513	-0.74702714
## 3	invertebrate	Intrcpt	-0.008634225	-0.10911299	0.09184454	-0.73732713
## 4	mammal	Intrcpt	0.055549381	-0.20674669	0.31784545	-0.65499594
## 5	reptilia	Intrcpt	0.037232224	-0.04236748	0.11683193	-0.04136392

```
##      upperPR
## 1 1.6051567
## 2 0.7377849
## 3 0.7200587
## 4 0.7660947
## 5 0.1158284
```

Extract p-values from these models to use later when adjusting them for multiple testing:

```
# taking p-values from models for False Discovery Rate p-value adjustment
p.SMD_intercept <- unlist(lapply(smd_mods, function(x) x$pval))
p.SMD_intercept
```

	bird	fish	invertebrate	mammal	reptilia
##	0.3560563	0.3900624	0.1778479	0.7143065	0.4793203

```
p.lnCVR_intercept <- unlist(lapply(lnCVR_mods, function(x) x$pval))
p.lnCVR_intercept
```

	bird	fish	invertebrate	mammal	reptilia
##	0.5812731	0.9117446	0.8659517	0.6776659	0.3554154

Personality trait MLMR models

These models include personality trait type as a moderator. Please note that we estimate the mean for each of the categorical levels because we are not really interested in whether the means differ, but whether or not males and females differ in any of these traits.

Complete model summaries are presented in Supplementary Table S15.

```
# we can just reload saved model outputs here to save time
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMR_models_pers_trait <- meta_model_fits(pers_new, phylo_vcv, type =
"pers")
  saveRDS(MLMR_models_pers_trait, "./output/MLMR_models_pers_trait")
}
```

```

} else{
  MLMR_models_pers_trait <- readRDS("./output/MLMR_models_pers_trait")
}

# Extract the SMD and lnCVR results
smd_mods_pers <- MLMR_models_pers_trait["SMD",]

lnCVR_mods_pers <- MLMR_models_pers_trait["lnCVR",]

# these model objects are used to make the orchard plots shown in Figures
2-6

```

Get prediction intervals for personality trait models:

```

# Get the combined estimates from them all
MLMA_estimates_SMD_pers <- plyr::ldply(lapply(smd_mods_pers, function(x)
print(mod_results(x, mod = "personality_trait"))))
MLMA_estimates_SMD_pers

```

	.id	name	estimate	lowerCL	upperCL	lowerPR
## 1	bird	Activity	-0.082630938	-0.3731716	0.2079097	-1.8686217
## 2	bird	Aggression	-0.150395862	-0.4380609	0.1372692	-1.9359235
## 3	bird	Boldness	-0.141981559	-0.3986038	0.1146406	-1.9227956
## 4	bird	Exploration	0.061120004	-0.2107229	0.3329629	-1.7219395
## 5	bird	Sociality	-0.589437560	-1.0564775	-0.1223977	-2.4122911
## 6	fish	Activity	0.153275118	-0.3114801	0.6180303	-1.6341205
## 7	fish	Aggression	0.252603144	-0.1984714	0.7036777	-1.5313017
## 8	fish	Boldness	0.083577441	-0.3561599	0.5233148	-1.6975086
## 9	fish	Exploration	0.265681594	-0.1976033	0.7289665	-1.5213342
## 10	fish	Sociality	0.149525877	-0.3372741	0.6363258	-1.6436989
## 11	invertebrate	Activity	0.250548279	-0.1462921	0.6473887	-2.1358328
## 12	invertebrate	Aggression	0.420366327	-0.2762252	1.1169578	-2.0333404
## 13	invertebrate	Boldness	0.242813292	-0.1483189	0.6339455	-2.1426306
## 14	invertebrate	Exploration	0.039752886	-0.4098280	0.4893337	-2.3559092
## 15	invertebrate	Sociality	0.413467493	-0.2272736	1.0542085	-2.0250603
## 16	mammal	Activity	-0.163065307	-0.5680824	0.2419518	-1.3909083
## 17	mammal	Aggression	0.091224955	-0.2978548	0.4803048	-1.1314722
## 18	mammal	Boldness	0.129883912	-0.2305105	0.4902783	-1.0840218
## 19	mammal	Exploration	0.026387251	-0.3500445	0.4028190	-1.1923585
## 20	mammal	Sociality	0.065770946	-0.3292890	0.4608309	-1.1588354
## 21	reptilia	Activity	-0.045918474	-0.4728193	0.3809823	-0.7480079
## 22	reptilia	Aggression	-0.127350973	-0.4621377	0.2074358	-0.7789961
## 23	reptilia	Boldness	0.106563010	-0.1369802	0.3501062	-0.5044050
## 24	reptilia	Exploration	0.251195715	0.0136709	0.4887205	-0.3574618
## 25	reptilia	Sociality	0.004637977	-0.5570016	0.5662775	-0.7843910
##	upperPR					
## 1	1.7033599					
## 2	1.6351318					
## 3	1.6388325					
## 4	1.8441795					


```
## 5 1.2334160
## 6 1.9406707
## 7 2.0365079
## 8 1.8646635
## 9 2.0526973
## 10 1.9427507
## 11 2.6369293
## 12 2.8740730
## 13 2.6282572
## 14 2.4354150
## 15 2.8519953
## 16 1.0647777
## 17 1.3139221
## 18 1.3437897
## 19 1.2451330
## 20 1.2903773
## 21 0.6561710
## 22 0.5242942
## 23 0.7175310
## 24 0.8598532
## 25 0.7936670
```

```
MLMA_estimates_lnCVR_pers <- plyr::ldply(lapply(lnCVR_mods_pers,
function(x) print(mod_results(x, mod = "personality_trait"))))
MLMA_estimates_lnCVR_pers
```

##	.id	name	estimate	lowerCL	upperCL
lowerPR					
## 1	bird	Activity	0.06247105	-0.223810303	0.34875240 -
1.547893585					
## 2	bird	Aggression	-0.07509972	-0.388723954	0.23852451 -
1.690523480					
## 3	bird	Boldness	0.04002236	-0.177573785	0.25761851 -
1.559615558					
## 4	bird	Exploration	-0.29821007	-0.550498369	-0.04592177 -
1.902909359					
## 5	bird	Sociality	0.14671033	-0.358195237	0.65161591 -
1.516228788					
## 6	fish	Activity	-0.01057724	-0.164934481	0.14378000 -
0.744939105					
## 7	fish	Aggression	-0.12076668	-0.259837252	0.01830390 -
0.852083418					
## 8	fish	Boldness	0.02281042	-0.082594775	0.12821562 -
0.702885678					
## 9	fish	Exploration	-0.03757428	-0.191214061	0.11606550 -
0.771786428					
## 10	fish	Sociality	0.25131850	0.039101979	0.46353502 -
0.497274393					
## 11	invertebrate	Activity	-0.04431023	-0.187795258	0.09917481 -
0.786780181					

```

## 12 invertebrate Aggression 0.03920221 -0.272921753 0.35132617 -
0.753041918
## 13 invertebrate Boldness -0.02254238 -0.149602850 0.10451808 -
0.762031219
## 14 invertebrate Exploration 0.12140566 -0.069729521 0.31254083 -
0.631664152
## 15 invertebrate Sociality 0.10933426 -0.267988928 0.48665745 -
0.710630523
## 16 mammal Activity 0.10970622 -0.183407387 0.40281983 -
0.617702167
## 17 mammal Aggression 0.11346377 -0.183447978 0.41037551 -
0.615477836
## 18 mammal Boldness 0.03065931 -0.237003686 0.29832231 -
0.686907726
## 19 mammal Exploration 0.05050685 -0.231448720 0.33246242 -
0.672493506
## 20 mammal Sociality 0.04407016 -0.251715474 0.33985579 -
0.684415138
## 21 reptilia Activity -0.09756095 -0.378433808 0.18331191 -
0.374657513
## 22 reptilia Aggression 0.15707735 -0.007912971 0.32206767 -
0.005694702
## 23 reptilia Boldness 0.08308063 -0.044997677 0.21115893 -
0.043275684
## 24 reptilia Exploration -0.08675909 -0.237272780 0.06375460 -
0.235249147
## 25 reptilia Sociality 0.03223093 -0.530881924 0.59534378 -
0.523310952
## upperPR
## 1 1.67283568
## 2 1.54032404
## 3 1.63966028
## 4 1.30648922
## 5 1.80964946
## 6 0.72378463
## 7 0.61055007
## 8 0.74850653
## 9 0.69663786
## 10 0.99991139
## 11 0.69815973
## 12 0.83144634
## 13 0.71694645
## 14 0.87447546
## 15 0.92929904
## 16 0.83711461
## 17 0.84240537
## 18 0.74822634
## 19 0.77350721
## 20 0.77255546
## 21 0.17953562

```

```
## 22 0.31984940
## 23 0.20943694
## 24 0.06173097
## 25 0.58777281

# Add in n and k to these dataframes
n_k<- pers_new %>%
  group_by(taxo_group, personality_trait) %>%
  summarise(n = n(), spp = length(unique(spp_name_phylo)), k =
length(unique(study_ID)))

# Summary of model estimates with number of studies, species and effect
sizes included
MLMA_estimates_SMD_pers <- data.frame(MLMA_estimates_SMD_pers, n_k[,c("n",
"spp", "k")])
MLMA_estimates_SMD_pers
```

	.id	name	estimate	lowerCL	upperCL	lowerPR
## 1	bird	Activity	-0.082630938	-0.3731716	0.2079097	-1.8686217
## 2	bird	Aggression	-0.150395862	-0.4380609	0.1372692	-1.9359235
## 3	bird	Boldness	-0.141981559	-0.3986038	0.1146406	-1.9227956
## 4	bird	Exploration	0.061120004	-0.2107229	0.3329629	-1.7219395
## 5	bird	Sociality	-0.589437560	-1.0564775	-0.1223977	-2.4122911
## 6	fish	Activity	0.153275118	-0.3114801	0.6180303	-1.6341205
## 7	fish	Aggression	0.252603144	-0.1984714	0.7036777	-1.5313017
## 8	fish	Boldness	0.083577441	-0.3561599	0.5233148	-1.6975086
## 9	fish	Exploration	0.265681594	-0.1976033	0.7289665	-1.5213342
## 10	fish	Sociality	0.149525877	-0.3372741	0.6363258	-1.6436989
## 11	invertebrate	Activity	0.250548279	-0.1462921	0.6473887	-2.1358328
## 12	invertebrate	Aggression	0.420366327	-0.2762252	1.1169578	-2.0333404
## 13	invertebrate	Boldness	0.242813292	-0.1483189	0.6339455	-2.1426306
## 14	invertebrate	Exploration	0.039752886	-0.4098280	0.4893337	-2.3559092
## 15	invertebrate	Sociality	0.413467493	-0.2272736	1.0542085	-2.0250603
## 16	mammal	Activity	-0.163065307	-0.5680824	0.2419518	-1.3909083
## 17	mammal	Aggression	0.091224955	-0.2978548	0.4803048	-1.1314722
## 18	mammal	Boldness	0.129883912	-0.2305105	0.4902783	-1.0840218
## 19	mammal	Exploration	0.026387251	-0.3500445	0.4028190	-1.1923585
## 20	mammal	Sociality	0.065770946	-0.3292890	0.4608309	-1.1588354
## 21	reptilia	Activity	-0.045918474	-0.4728193	0.3809823	-0.7480079
## 22	reptilia	Aggression	-0.127350973	-0.4621377	0.2074358	-0.7789961
## 23	reptilia	Boldness	0.106563010	-0.1369802	0.3501062	-0.5044050
## 24	reptilia	Exploration	0.251195715	0.0136709	0.4887205	-0.3574618
## 25	reptilia	Sociality	0.004637977	-0.5570016	0.5662775	-0.7843910
##	upperPR	n	spp	k		
## 1	1.7033599	60	9	14		
## 2	1.6351318	50	10	11		
## 3	1.6388325	262	96	24		
## 4	1.8441795	77	9	15		
## 5	1.2334160	31	2	3		
## 6	1.9406707	92	5	9		

```
## 7 2.0365079 95 14 17
## 8 1.8646635 173 13 24
## 9 2.0526973 103 7 10
## 10 1.9427507 27 6 7
## 11 2.6369293 166 17 18
## 12 2.8740730 33 6 5
## 13 2.6282572 164 23 23
## 14 2.4354150 54 6 7
## 15 2.8519953 6 1 1
## 16 1.0647777 83 12 14
## 17 1.3139221 87 14 16
## 18 1.3437897 193 27 27
## 19 1.2451330 213 16 19
## 20 1.2903773 98 10 12
## 21 0.6561710 5 3 3
## 22 0.5242942 30 2 2
## 23 0.7175310 25 3 4
## 24 0.8598532 32 4 5
## 25 0.7936670 3 2 2
```

```
MLMA_estimates_lnCVR_pers <- data.frame(MLMA_estimates_lnCVR_pers,
n_k[,c("n", "spp", "k")])
MLMA_estimates_lnCVR_pers
```

	.id	name	estimate	lowerCL	upperCL
lowerPR					
## 1	bird	Activity	0.06247105	-0.223810303	0.34875240 -
1.547893585					
## 2	bird	Aggression	-0.07509972	-0.388723954	0.23852451 -
1.690523480					
## 3	bird	Boldness	0.04002236	-0.177573785	0.25761851 -
1.559615558					
## 4	bird	Exploration	-0.29821007	-0.550498369	-0.04592177 -
1.902909359					
## 5	bird	Sociality	0.14671033	-0.358195237	0.65161591 -
1.516228788					
## 6	fish	Activity	-0.01057724	-0.164934481	0.14378000 -
0.744939105					
## 7	fish	Aggression	-0.12076668	-0.259837252	0.01830390 -
0.852083418					
## 8	fish	Boldness	0.02281042	-0.082594775	0.12821562 -
0.702885678					
## 9	fish	Exploration	-0.03757428	-0.191214061	0.11606550 -
0.771786428					
## 10	fish	Sociality	0.25131850	0.039101979	0.46353502 -
0.497274393					
## 11	invertebrate	Activity	-0.04431023	-0.187795258	0.09917481 -
0.786780181					
## 12	invertebrate	Aggression	0.03920221	-0.272921753	0.35132617 -
0.753041918					

```

## 13 invertebrate Boldness -0.02254238 -0.149602850 0.10451808 -
0.762031219
## 14 invertebrate Exploration 0.12140566 -0.069729521 0.31254083 -
0.631664152
## 15 invertebrate Sociality 0.10933426 -0.267988928 0.48665745 -
0.710630523
## 16 mammal Activity 0.10970622 -0.183407387 0.40281983 -
0.617702167
## 17 mammal Aggression 0.11346377 -0.183447978 0.41037551 -
0.615477836
## 18 mammal Boldness 0.03065931 -0.237003686 0.29832231 -
0.686907726
## 19 mammal Exploration 0.05050685 -0.231448720 0.33246242 -
0.672493506
## 20 mammal Sociality 0.04407016 -0.251715474 0.33985579 -
0.684415138
## 21 reptilia Activity -0.09756095 -0.378433808 0.18331191 -
0.374657513
## 22 reptilia Aggression 0.15707735 -0.007912971 0.32206767 -
0.005694702
## 23 reptilia Boldness 0.08308063 -0.044997677 0.21115893 -
0.043275684
## 24 reptilia Exploration -0.08675909 -0.237272780 0.06375460 -
0.235249147
## 25 reptilia Sociality 0.03223093 -0.530881924 0.59534378 -
0.523310952
## upperPR n spp k
## 1 1.67283568 60 9 14
## 2 1.54032404 50 10 11
## 3 1.63966028 262 96 24
## 4 1.30648922 77 9 15
## 5 1.80964946 31 2 3
## 6 0.72378463 92 5 9
## 7 0.61055007 95 14 17
## 8 0.74850653 173 13 24
## 9 0.69663786 103 7 10
## 10 0.99991139 27 6 7
## 11 0.69815973 166 17 18
## 12 0.83144634 33 6 5
## 13 0.71694645 164 23 23
## 14 0.87447546 54 6 7
## 15 0.92929904 6 1 1
## 16 0.83711461 83 12 14
## 17 0.84240537 87 14 16
## 18 0.74822634 193 27 27
## 19 0.77350721 213 16 19
## 20 0.77255546 98 10 12
## 21 0.17953562 5 3 3
## 22 0.31984940 30 2 2
## 23 0.20943694 25 3 4

```

```
## 24 0.06173097 32 4 5
## 25 0.58777281 3 2 2
```

Extract p-values from models for multiple testing adjustment later:

```
# extract p-values for multiple testing
p.SMD_pers <- unlist(lapply(smd_mods_pers, function(x) x$pval))
p.SMD_pers

##      bird1      bird2      bird3      bird4      bird5
## 0.57653039 0.30479363 0.27751708 0.65883812 0.01348632
##      fish1      fish2      fish3      fish4      fish5
## 0.51728620 0.27173344 0.70897900 0.26038453 0.54643814
## invertebrate1 invertebrate2 invertebrate3 invertebrate4 invertebrate5
## 0.21528997 0.23621794 0.22305017 0.86210097 0.20535070
##      mammal1      mammal2      mammal3      mammal4      mammal5
## 0.42949385 0.64539838 0.47941582 0.89056692 0.74385107
##      reptilia1      reptilia2      reptilia3      reptilia4      reptilia5
## 0.83127102 0.45179068 0.38700906 0.03843513 0.98694696

p.lnCVR_pers <- unlist(lapply(lnCVR_mods_pers, function(x) x$pval))
p.lnCVR_pers

##      bird1      bird2      bird3      bird4      bird5
## 0.66827246 0.63819433 0.71794905 0.02062033 0.56829592
##      fish1      fish2      fish3      fish4      fish5
## 0.89295129 0.08859994 0.67087141 0.63106746 0.02038126
## invertebrate1 invertebrate2 invertebrate3 invertebrate4 invertebrate5
## 0.54416559 0.80512014 0.72746311 0.21252880 0.56927261
##      mammal1      mammal2      mammal3      mammal4      mammal5
## 0.46265634 0.45330729 0.82211806 0.72515472 0.76995584
##      reptilia1      reptilia2      reptilia3      reptilia4      reptilia5
## 0.49192601 0.06178802 0.20080617 0.25517952 0.90971983
```

Personality trait x SSD MLMR models

Now let's look at how SSD interacts with personality trait type. Here we are not estimating an intercept either, so each intercept varies by trait category and each slope as well. Note that there are lots of warnings, but these are the result of many levels not being present in taxa groups. We chose not to scale SSD_index because it is easier (and biologically relevant) to interpret SSD when it is 0 (when males and females are the same size), and when SSD is positive (when males are larger than females).

Model summaries are presented in Supplementary Table S17.

```
# again, we can just reload our saved model output here
rerun_models == FALSE

## [1] TRUE
```

```

    if(rerun_models == TRUE){
      MLMR_models_pers_SSD <- meta_model_fits(pers_new, phylo_vcv, type =
"pers_SSD")
      saveRDS(MLMR_models_pers_SSD, "./output/MLMR_models_pers_SSD")
    } else{
      MLMR_models_pers_SSD <- readRDS("./output/MLMR_models_pers_SSD")
    }

# Extract the SMD and lnCVR results
smd_mods_pers_SSD <- MLMR_models_pers_SSD["SMD",]

lnCVR_mods_pers_SSD <- MLMR_models_pers_SSD["lnCVR",]

```

Get the prediction intervals for our interaction models:

```

# extract estimates using modified function in func.R file:
# SMD
MLMA_estimates_SMD_SSD <- plyr::ldply(lapply(smd_mods_pers_SSD, function(x)
  print(mod_results_new(x, mod_cat = "personality_trait", mod_cont =
"SSD_index", type = "zero"))))

```

```

MLMA_estimates_SMD_SSD

##           .id           name      estimate      lowerCL      upperCL
## 1         bird      Aggression -0.151036532 -0.46852121 0.16644814
## 2         bird      Boldness  -0.214280247 -0.50506697 0.07650648
## 3         bird      Exploration 0.074227347 -0.23570163 0.38415632
## 4         bird      Sociality  -1.226866406 -2.91329598 0.45956317
## 5         bird Aggression:SSD_index 1.157852238 -1.77508072 4.09078520
## 6         bird  Boldness:SSD_index -0.848875167 -2.27108269 0.57333236
## 7         bird Exploration:SSD_index -1.321422783 -2.83439647 0.19155090
## 8         bird  Sociality:SSD_index -3.911491684 -12.70365135 4.88066798
## 9         fish      Aggression 0.103585582 -0.23823551 0.44540667
## 10        fish      Boldness 0.140413174 -0.17920769 0.46003403
## 11        fish      Exploration 0.272900300 -0.07480852 0.62060912
## 12        fish      Sociality -0.022130752 -0.41855338 0.37429188
## 13        fish Aggression:SSD_index 2.389762942 -0.45150502 5.23103090
## 14        fish  Boldness:SSD_index 1.286741330 -1.19395415 3.76743681
## 15        fish Exploration:SSD_index 0.607258807 -2.06820241 3.28272002
## 16        fish  Sociality:SSD_index -2.088881283 -5.26466879 1.08690623
## 17 invertibrate      Aggression 0.834112113 -0.25274144 1.92096567
## 18 invertibrate      Boldness 0.197107566 -0.23590070 0.63011583
## 19 invertibrate      Exploration -0.049461574 -0.67591471 0.57699156
## 20 invertibrate      Sociality 0.257291051 -0.39825935 0.91284145
## 21 invertibrate Aggression:SSD_index 0.680138870 -3.87195421 5.23223195
## 22 invertibrate  Boldness:SSD_index 0.915400058 -0.22367183 2.05447195
## 23 invertibrate Exploration:SSD_index 0.654641914 -1.74659478 3.05587861
## 24        mammal      Aggression -0.068848442 -0.64585730 0.50816041
## 25        mammal      Boldness -0.004157273 -0.59355508 0.58524054

```

## 26	mammal	Exploration	-0.153246169	-0.73241582	0.42592348
## 27	mammal	Sociality	0.066790898	-0.56353783	0.69711962
## 28	mammal	Aggression:SSD_index	2.607701753	1.05525461	4.16014890
## 29	mammal	Boldness:SSD_index	2.126650354	0.80742684	3.44587387
## 30	mammal	Exploration:SSD_index	2.374650037	1.04169176	3.70760831
## 31	mammal	Sociality:SSD_index	1.789427351	0.27966189	3.29919282
## 32	reptilia	Aggression	-0.066728598	-0.26994608	0.13648888
## 33	reptilia	Boldness	-0.054772986	-0.40164148	0.29209551
## 34	reptilia	Exploration	0.459170599	0.20789126	0.71044993
## 35	reptilia	Sociality	-0.022112770	-0.81539374	0.77116820
## 36	reptilia	Aggression:SSD_index	4.070920104	0.11347450	8.02836571
## 37	reptilia	Boldness:SSD_index	1.393904655	-2.56056843	5.34837774
## 38	reptilia	Exploration:SSD_index	3.196194595	-0.57224573	6.96463492
## 39	reptilia	Sociality:SSD_index	2.747115585	-2.91125356	8.40548473
##	lowerPR	upperPR			
## 1	-2.00561231	1.7035393			
## 2	-2.06447260	1.6359121			
## 3	-1.77906993	1.9275246			
## 4	-3.71336988	1.2596371			
## 5	-2.29768687	4.6133913			
## 6	-3.16432956	1.4665792			
## 7	-3.69370948	1.0508639			
## 8	-12.89150986	5.0685265			
## 9	-1.54548590	1.7526571			
## 10	-1.50420003	1.7850264			
## 11	-1.37740165	1.9232022			
## 12	-1.68337908	1.6391176			
## 13	-0.87755993	5.6570858			
## 14	-1.67238780	4.2458705			
## 15	-2.51695114	3.7314688			
## 16	-5.65093426	1.4731717			
## 17	-1.83590711	3.5041313			
## 18	-2.27983574	2.6740509			
## 19	-2.56743606	2.4685129			
## 20	-2.26807988	2.7826620			
## 21	-4.48409422	5.8443720			
## 22	-1.77629769	3.6070978			
## 23	-2.76788488	4.0771687			
## 24	-1.47455435	1.3368575			
## 25	-1.41499379	1.4066792			
## 26	-1.55984042	1.2533481			
## 27	-1.36162927	1.4952111			
## 28	0.59445511	4.6209484			
## 29	0.28724269	3.9660580			
## 30	0.52536703	4.2239330			
## 31	-0.19109320	3.7699479			
## 32	-0.56457573	0.4311185			
## 33	-0.62650034	0.5169544			
## 34	-0.06015177	0.9784930			
## 35	-0.93636061	0.8921351			


```
## 36 0.08746304 8.0543772
## 37 -2.58659930 5.3744086
## 38 -0.59955261 6.9919418
## 39 -2.92947633 8.4237075
```

```
# LnCVR
```

```
MLMA_estimates_lnCVR_SSD <- plyr::ldply(lapply(lnCVR_mods_pers_SSD,
function(x)
  print(mod_results_new(x, mod_cat = "personality_trait", mod_cont =
"SSD_index", type = "zero"))))
```

```
MLMA_estimates_lnCVR_SSD
```

##	.id	name	estimate	lowerCL	upperCL
## 1	bird	Aggression	-0.140533040	-0.50369345	0.22262737
## 2	bird	Boldness	0.041902624	-0.22089894	0.30470418
## 3	bird	Exploration	-0.423453401	-0.72575953	-0.12114727
## 4	bird	Sociality	1.657798939	-0.47785027	3.79344814
## 5	bird	Aggression:SSD_index	0.229910712	-3.09632946	3.55615089
## 6	bird	Boldness:SSD_index	1.444566031	-0.29662027	3.18575233
## 7	bird	Exploration:SSD_index	2.360671927	0.44152412	4.27981974
## 8	bird	Sociality:SSD_index	9.159821298	-1.91573451	20.23537710
## 9	fish	Aggression	-0.129070115	-0.27194461	0.01380438
## 10	fish	Boldness	0.005436362	-0.10297982	0.11385254
## 11	fish	Exploration	-0.008212732	-0.18136948	0.16494401
## 12	fish	Sociality	0.087624529	-0.16112582	0.33637487
## 13	fish	Aggression:SSD_index	0.530238759	-1.68577830	2.74625582
## 14	fish	Boldness:SSD_index	0.605617604	-1.48753113	2.69876634
## 15	fish	Exploration:SSD_index	1.031331626	-1.22079150	3.28345475
## 16	fish	Sociality:SSD_index	-1.236517607	-3.78677304	1.31373783
## 17	invertebrate	Aggression	-0.203555549	-0.58086635	0.17375525
## 18	invertebrate	Boldness	0.001279023	-0.14241642	0.14497446
## 19	invertebrate	Exploration	0.059489195	-0.25144755	0.37042594
## 20	invertebrate	Sociality	0.128494477	-0.25716983	0.51415878
## 21	invertebrate	Aggression:SSD_index	1.053333652	-0.28691645	2.39358376
## 22	invertebrate	Boldness:SSD_index	-0.019558502	-0.70714726	0.66803025
## 23	invertebrate	Exploration:SSD_index	-0.552874790	-2.17236028	1.06661070
## 24	mammal	Aggression	0.116404800	-0.20920302	0.44201262
## 25	mammal	Boldness	0.050279755	-0.28084944	0.38140895
## 26	mammal	Exploration	0.030879566	-0.29354193	0.35530106
## 27	mammal	Sociality	0.121987951	-0.23596112	0.47993703
## 28	mammal	Aggression:SSD_index	-0.138782253	-1.23324702	0.95568251
## 29	mammal	Boldness:SSD_index	-0.160814667	-1.01376179	0.69213246
## 30	mammal	Exploration:SSD_index	-0.002060956	-0.88930493	0.88518301
## 31	mammal	Sociality:SSD_index	-0.357422340	-1.35708312	0.64223844
## 32	reptilia	Aggression	0.280341067	0.09692606	0.46375608
## 33	reptilia	Boldness	0.041552023	-0.19703884	0.28014288
## 34	reptilia	Exploration	-0.019207567	-0.19364071	0.15522557
## 35	reptilia	Sociality	0.033598276	-1.39712811	1.46432466
## 36	reptilia	Aggression:SSD_index	-1.255533966	-7.56757841	5.05651047

```

## 37      reptilia      Boldness:SSD_index  2.075132144 -3.85381727  8.00408156
## 38      reptilia Exploration:SSD_index  2.934481218 -2.94935384  8.81831627
## 39      reptilia  Sociality:SSD_index  2.311721057 -6.17723397 10.80067608
##          lowerPR      upperPR
## 1  -1.83260874  1.5515427
## 2  -1.63150687  1.7153121
## 3  -2.10351997  1.2566132
## 4  -1.04261440  4.3582123
## 5  -3.48426596  3.9440874
## 6  -0.95605148  3.8451835
## 7  -0.17198735  4.8933312
## 8  -2.03835583 20.3579984
## 9  -0.85781158  0.5996714
## 10 -0.71733959  0.7282123
## 11 -0.74349109  0.7270656
## 12 -0.66903110  0.8442802
## 13 -1.79814751  2.8586250
## 14 -1.60615152  2.8173867
## 15 -1.33144419  3.3941074
## 16 -3.88499902  1.4119638
## 17 -1.02495455  0.6178435
## 18 -0.74234807  0.7449061
## 19 -0.73361517  0.8525936
## 20 -0.69677507  0.9537640
## 21 -0.47264250  2.5793098
## 22 -1.02211083  0.9829938
## 23 -2.32912537  1.2233758
## 24 -0.63758614  0.8703957
## 25 -0.70611196  0.8066715
## 26 -0.72259982  0.7843590
## 27 -0.64652322  0.8904991
## 28 -1.42732213  1.1497576
## 29 -1.25168619  0.9300569
## 30 -1.11995350  1.1158316
## 31 -1.56647303  0.8516283
## 32  0.09686176  0.4638204
## 33 -0.19708827  0.2801923
## 34 -0.19370831  0.1552932
## 35 -1.39713636  1.4643329
## 36 -7.56758027  5.0565123
## 37 -3.85381926  8.0040835
## 38 -2.94935584  8.8183183
## 39 -6.17723536 10.8006775

```

```

# Table to get species numbers, no. studies and no. effect sizes:
data.frame(pers_new %>%
  group_by(taxo_group, personality_trait) %>%
  filter(!is.na(SSD_index))%>%
  summarise(n = n(), N_spp = length(unique(spp_name_phylo)), N_studies =
length(unique(study_ID))))

```

##	taxo_group	personality_trait	n	N_spp	N_studies
## 1	bird	activity	60	9	14
## 2	bird	aggression	41	8	9
## 3	bird	boldness	234	78	21
## 4	bird	exploration	77	9	15
## 5	bird	sociality	31	2	3
## 6	fish	activity	92	5	9
## 7	fish	aggression	93	13	16
## 8	fish	boldness	171	12	23
## 9	fish	exploration	101	6	9
## 10	fish	sociality	27	6	7
## 11	invertebrate	activity	165	16	18
## 12	invertebrate	aggression	32	5	5
## 13	invertebrate	boldness	164	23	23
## 14	invertebrate	exploration	54	6	7
## 15	invertebrate	sociality	6	1	1
## 16	mammal	activity	83	12	14
## 17	mammal	aggression	85	13	15
## 18	mammal	boldness	163	26	26
## 19	mammal	exploration	213	16	19
## 20	mammal	sociality	89	9	11
## 21	reptilia	activity	5	3	3
## 22	reptilia	aggression	30	2	2
## 23	reptilia	boldness	25	3	4
## 24	reptilia	exploration	32	4	5
## 25	reptilia	sociality	3	2	2

SSD subset models

Because we aren't really interested in how each trait type differs from each other, we need to run our SSD models on subsets of the data where we can get the mean estimates for individual trait types and for SSD. Model summaries are presented in Supplementary Table S16.

NOTE: Since we are conducting our meta-regression at the species level (the level at which we can assume effect sizes are independent), any personality trait with fewer than 10 species needs to be dropped to look at interactions between SSD and personality. Having a minimum of 10 studies etc. is the rule of thumb for meta-regressions (e.g. see Borenstein et al Intro to Meta-A)

NEED TO DROP:

1. ALL REPTILES - GONE, NOT ENOUGH SPECIES
2. BIRDS - EVERYTHING BUT BOLDNESS
3. FISH - ACTIVITY, EXPLORATION & SOCIALITY
4. INVERTS - SOCIALITY, EXPLORATION & AGGRESSION
5. MAMMALS - SOCIALITY

Mammals

SSD for activity, boldness, aggression and exploration.

1. MAMMALS

First, we need to subset our pers dataset by taxo group to drop the unwanted levels.

a. activity

```
pers_new_mammal_activity <- as.data.frame(pers_new %>%  
  filter(personality_trait == "activity") %>%  
  filter(taxo_group == "mammal"))
```

b. boldness

```
pers_new_mammal_boldness <- as.data.frame(pers_new %>%  
  filter(personality_trait == "boldness") %>%  
  filter(taxo_group == "mammal"))
```

c. aggression

```
pers_new_mammal_aggression <- as.data.frame(pers_new %>%  
  filter(personality_trait == "aggression") %>%  
  filter(taxo_group == "mammal"))
```

d. exploration

```
pers_new_mammal_exploration <- as.data.frame(pers_new %>%  
  filter(personality_trait == "exploration") %>%  
  filter(taxo_group == "mammal"))
```

Extract the phylogenetic correlation matrices

```
phylo_vcv_mammal <- phylo_vcv[[4]]
```

Activity:

```
  # a. activity  
  #SMD  
  MLMR_mods_pers_SSD_mammal_activity_SMD <- rma.mv(SMD_yi_flip ~ SSD_index,  
V = SMD_vi,  
                                                    random = list(~1|study_ID,  
~1|spp_name_phylo, ~1|obs),  
                                                    R =  
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),  
                                                    test = "t", data =  
pers_new_mammal_activity)  
## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |  
:  
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which  
there  
## are no data.  
  
  MLMR_mods_pers_SSD_mammal_activity_SMD  
  
##  
## Multivariate Meta-Analysis Model (k = 83; method: REML)
```

```
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.1000  0.3163    14    no      study_ID    no
## sigma^2.2  2.8075  1.6755    12    no  spp_name_phylo  yes
## sigma^2.3  0.1914  0.4375    83    no      obs      no
##
## Test for Residual Heterogeneity:
## QE(df = 81) = 320.2151, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 81) = 5.0697, p-val = 0.0271
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb      ci.ub
## intrcpt      0.5040  1.2531   0.4022  81   0.6886  -1.9892   2.9972
## SSD_index    -2.2054  0.9795  -2.2516  81   0.0271  -4.1543  -0.2565 *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#LnCVR
MLMR_mods_pers_SSD_mammal_activity_lncvr <- rma.mv(CVR_yi ~ SSD_index, V
= CVR_vi,
                                                    random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                                    R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                                    test = "t", data =
pers_new_mammal_activity)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

MLMR_mods_pers_SSD_mammal_activity_lncvr

##
## Multivariate Meta-Analysis Model (k = 83; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0293  0.1710    14    no      study_ID    no
## sigma^2.2  0.0001  0.0079    12    no  spp_name_phylo  yes
## sigma^2.3  0.0607  0.2465    83    no      obs      no
##
```

```
## Test for Residual Heterogeneity:
## QE(df = 81) = 146.2596, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 81) = 0.1324, p-val = 0.7169
##
## Model Results:
##
##           estimate      se    tval  df    pval    ci.lb    ci.ub
## intrcpt      0.0518  0.0988  0.5236  81  0.6020  -0.1449  0.2484
## SSD_index    0.1248  0.3430  0.3638  81  0.7169  -0.5577  0.8073
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Boldness:

```
# b. boldness
#SMD
MLMR_mods_pers_SSD_mammal_bold_SMD <- rma.mv(SMD_yi_flip ~ SSD_index, V =
SMD_vi,
                                             random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                             R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                             test = "t", data =
pers_new_mammal_boldness)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_mammal_bold_SMD

##
## Multivariate Meta-Analysis Model (k = 163; method: REML)
##
## Variance Components:
##
##           estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0088  0.0938    26    no      study_ID    no
## sigma^2.2  0.0000  0.0050    26    no  spp_name_phylo  yes
## sigma^2.3  0.1707  0.4132   163    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 161) = 405.7659, p-val < .0001
##
```

```

## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 161) = 1.3101, p-val = 0.2541
##
## Model Results:
##
##           estimate      se      tval   df    pval    ci.lb    ci.ub
## intrcpt      0.0739  0.0774   0.9547  161   0.3412  -0.0789   0.2267
## SSD_index  -0.1686  0.1473  -1.1446  161   0.2541  -0.4594   0.1223
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

  #LnCVR
  MLMR_mods_pers_SSD_mammal_bold_lncvr <- rma.mv(CVR_yi ~ SSD_index, V =
CVR_vi,
                                                random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                                R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                                test = "t", data =
pers_new_mammal_boldness)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

  MLMR_mods_pers_SSD_mammal_bold_lncvr

##
## Multivariate Meta-Analysis Model (k = 163; method: REML)
##
## Variance Components:
##
##           estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0029  0.0543    26    no      study_ID    no
## sigma^2.2  0.0000  0.0027    26    no  spp_name_phylo  yes
## sigma^2.3  0.0211  0.1452   163    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 161) = 177.7499, p-val = 0.1737
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 161) = 2.1066, p-val = 0.1486
##
## Model Results:

```

```
##
##          estimate      se    tval   df    pval    ci.lb    ci.ub
## intrcpt      0.0114  0.0513  0.2230  161  0.8238  -0.0899  0.1128
## SSD_index    0.1316  0.0907  1.4514  161  0.1486  -0.0475  0.3106
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Aggression:

```
# c. aggression
#SMD
MLMR_mods_pers_SSD_mammal_aggression_SMD <- rma.mv(SMD_yi_flip ~
SSD_index, V = SMD_vi,
                                                    random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                                    R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                                    test = "t", data =
pers_new_mammal_aggression)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_mammal_aggression_SMD

##
## Multivariate Meta-Analysis Model (k = 85; method: REML)
##
## Variance Components:
##
##          estim    sqrt  nlvls  fixed          factor    R
## sigma^2.1  0.0000  0.0039    15    no          study_ID  no
## sigma^2.2  0.6852  0.8277    13    no  spp_name_phylo  yes
## sigma^2.3  0.1430  0.3781    85    no             obs   no
##
## Test for Residual Heterogeneity:
## QE(df = 83) = 312.3189, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 83) = 4.2481, p-val = 0.0424
##
## Model Results:
##
##          estimate      se    tval   df    pval    ci.lb    ci.ub
## intrcpt     -0.1036  0.6014  -0.1722  83  0.8637  -1.2997  1.0926
```



```
## SSD_index    1.4134  0.6858   2.0611  83  0.0424   0.0495  2.7774  *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

    #LnCVR
    MLMR_mods_pers_SSD_mammal_aggression_lncvr <- rma.mv(CVR_yi ~ SSD_index,
V = CVR_vi,
                                random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                test = "t", data =
pers_new_mammal_aggression)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

    MLMR_mods_pers_SSD_mammal_aggression_lncvr

##
## Multivariate Meta-Analysis Model (k = 85; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.1790  0.4231    15    no      study_ID    no
## sigma^2.2  0.0000  0.0052    13    no  spp_name_phylo  yes
## sigma^2.3  0.1539  0.3922    85    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 83) = 202.3514, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 83) = 0.0118, p-val = 0.9138
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb      ci.ub
## intrcpt      0.0992  0.1534   0.6467  83  0.5196  -0.2059  0.4042
## SSD_index  -0.0756  0.6959  -0.1086  83  0.9138  -1.4596  1.3084
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Exploration:

```
# d. exploration
#SMD
MLMR_mods_pers_SSD_mammal_explore_SMD <- rma.mv(SMD_yi_flip ~ SSD_index,
V = SMD_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
test = "t", data =
pers_new_mammal_exploration)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

MLMR_mods_pers_SSD_mammal_explore_SMD

##
## Multivariate Meta-Analysis Model (k = 213; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0504  0.2244    19    no      study_ID    no
## sigma^2.2  0.0000  0.0048    16    no  spp_name_phylo  yes
## sigma^2.3  0.1331  0.3649   213    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 211) = 658.4587, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 211) = 0.0351, p-val = 0.8516
##
## Model Results:
##
##      estimate      se      tval   df      pval      ci.lb      ci.ub
## intrcpt      -0.0016  0.0914  -0.0173  211   0.9862   -0.1817   0.1786
## SSD_index     -0.0522  0.2786  -0.1873  211   0.8516   -0.6015   0.4971
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#LnCVR
MLMR_mods_pers_SSD_mammal_explore_lncvr <- rma.mv(CVR_yi ~ SSD_index, V =
CVR_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
```

```

R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
      test = "t", data =
pers_new_mammal_exploration)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

      MLMR_mods_pers_SSD_mammal_explore_lncvr

##
## Multivariate Meta-Analysis Model (k = 213; method: REML)
##
## Variance Components:
##
##           estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0198  0.1406    19    no      study_ID    no
## sigma^2.2  0.0265  0.1628    16    no  spp_name_phylo  yes
## sigma^2.3  0.0323  0.1799   213    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 211) = 361.1620, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 211) = 0.2658, p-val = 0.6067
##
## Model Results:
##
##           estimate      se      tval   df      pval      ci.lb      ci.ub
## intrcpt      -0.0595  0.1507  -0.3951  211   0.6932   -0.3566   0.2375
## SSD_index     0.1324  0.2567   0.5156  211   0.6067   -0.3737   0.6384
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Birds

SSD for boldness only.

```

# 2. BIRDS

# subset dataset
pers_new_bird <- as.data.frame(pers_new %>%
  filter(personality_trait == "boldness" & taxo_group == "bird"))

# phylo_vcv birds only
phylo_vcv_bird <- phylo_vcv[[1]]

```

Boldness:

```
# SMD
MLMR_mods_pers_SSD_bird_SMD <- rma.mv(SMD_yi_flip ~ SSD_index, V =
SMD_vi,
                                     random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                     R =
list(spp_name_phylo=phylo_vcv_bird), control=list(optimizer="optim"),
                                     test = "t", data = pers_new_bird)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_bird_SMD

##
## Multivariate Meta-Analysis Model (k = 234; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  1.9496  1.3963    21    no      study_ID   no
## sigma^2.2   0.0001  0.0074    78    no  spp_name_phylo  yes
## sigma^2.3   0.0925  0.3042   234    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 232) = 1579.6588, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 232) = 0.1117, p-val = 0.7385
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb  ci.ub
## intrcpt      -0.2211  0.3139  -0.7043 232  0.4819  -0.8397  0.3974
## SSD_index     -0.2015  0.6028  -0.3342 232  0.7385  -1.3890  0.9861
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# LnCVR
MLMR_mods_pers_SSD_bird_lncvr <- rma.mv(CVR_yi ~ SSD_index, V = CVR_vi,
                                     random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                     R =
```

```
list(spp_name_phylo=phylo_vcv_bird), control=list(optimizer="optim"),
      test = "t", data = pers_new_bird)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_bird_lncvr

##
## Multivariate Meta-Analysis Model (k = 234; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0000  0.0003    21    no      study_ID    no
## sigma^2.2  0.0029  0.0537    78    no  spp_name_phylo  yes
## sigma^2.3  0.0000  0.0003   234    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 232) = 244.9667, p-val = 0.2670
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 232) = 0.6126, p-val = 0.4346
##
## Model Results:
##
##      estimate      se    tval   df    pval    ci.lb    ci.ub
## intrcpt      0.0428  0.0361  1.1851  232  0.2372  -0.0283  0.1139
## SSD_index    0.1023  0.1307  0.7827  232  0.4346  -0.1553  0.3599
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fish

SSD for aggression and boldness.

3. FISH

```
# subset by trait type
# a. aggression
pers_new_fish_aggression <- as.data.frame(pers_new %>%
  filter(personality_trait == "aggression") %>%
  filter(taxo_group == "fish"))
```

```
# b. boldness
pers_new_fish_bold <- as.data.frame(pers_new %>%
  filter(personality_trait == "boldness") %>%
  filter(taxo_group == "fish"))

# phylo
phylo_vcv_fish <- phylo_vcv[[2]]
```

Aggression:

```
# a. aggression
# SMD
MLMR_mods_pers SSD_fish_aggression_SMD <- rma.mv(SMD_yi_flip ~
SSD_index, V = SMD_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
test = "t", data =
pers_new_fish_aggression)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers SSD_fish_aggression_SMD

##
## Multivariate Meta-Analysis Model (k = 93; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0194  0.1395    16    no      study_ID    no
## sigma^2.2  0.3329  0.5770    13    no  spp_name_phylo  yes
## sigma^2.3  0.1704  0.4128    93    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 91) = 334.1728, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 91) = 0.2301, p-val = 0.6326
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb      ci.ub
## intrcpt      -0.1643  0.3987  -0.4120  91  0.6813  -0.9562  0.6277
```

```

## SSD_index    0.2659  0.5544  0.4797  91  0.6326  -0.8352  1.3671
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# LnCVR
MLMR_mods_pers_SSD_fish_aggression_lncvr <- rma.mv(CVR_yi ~ SSD_index,
V = CVR_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
test = "t", data =
pers_new_fish_aggression)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_fish_aggression_lncvr

##
## Multivariate Meta-Analysis Model (k = 93; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0210  0.1450    16    no      study_ID    no
## sigma^2.2  0.0000  0.0021    13    no  spp_name_phylo  yes
## sigma^2.3  0.0000  0.0012    93    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 91) = 68.2701, p-val = 0.9640
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 91) = 0.1495, p-val = 0.6999
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb      ci.ub
## intrcpt    -0.1163  0.0597  -1.9483  91  0.0545   -0.2348   0.0023 .
## SSD_index   -0.1323  0.3423  -0.3866  91  0.6999   -0.8122   0.5476
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Boldness:

```
# SMD
MLMR_mods_pers_SSD_fish_bold_SMD <- rma.mv(SMD_yi_flip ~ SSD_index, V =
SMD_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
test = "t", data =
pers_new_fish_bold)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_fish_bold_SMD

##
## Multivariate Meta-Analysis Model (k = 171; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.1717  0.4143    23    no      study_ID    no
## sigma^2.2  0.0279  0.1671    12    no  spp_name_phylo  yes
## sigma^2.3  0.1634  0.4042   171    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 169) = 614.1157, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 169) = 0.3196, p-val = 0.5726
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb  ci.ub
## intrcpt      0.1883  0.1764   1.0678 169   0.2871  -0.1599  0.5366
## SSD_index    -0.2571  0.4548  -0.5653 169   0.5726  -1.1550  0.6408
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# LnCVR
MLMR_mods_pers_SSD_fish_bold_lncvr <- rma.mv(CVR_yi ~ SSD_index, V =
CVR_vi,
random = list(~1|study_ID,
```



```

~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
                                test = "t", data =
pers_new_fish_bold)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

      MLMR_mods_pers_SSD_fish_bold_lncvr

##
## Multivariate Meta-Analysis Model (k = 171; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0445  0.2109    23    no      study_ID    no
## sigma^2.2  0.0184  0.1356    12    no  spp_name_phylo  yes
## sigma^2.3  0.0881  0.2968   171    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 169) = 395.3375, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 169) = 0.3068, p-val = 0.5804
##
## Model Results:
##
##      estimate      se      tval  df      pval      ci.lb  ci.ub
## intrcpt      0.0063  0.1310   0.0482 169   0.9616  -0.2523  0.2649
## SSD_index  -0.1507  0.2721  -0.5539 169   0.5804  -0.6878  0.3864
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Inverts

SSD for activity and boldness.

```

# 4. INVERTS
# subset dataset
# a. activity
invert_activity <- as.data.frame(pers_new %>%

```

```

filter(personality_trait == "activity") %>%
filter(taxo_group == "invertebrate"))
# b. boldness
invert_bold <- as.data.frame(pers_new %>%
filter(personality_trait == "boldness") %>%
filter(taxo_group == "invertebrate"))

# phylo
phylo_vcv_invert <- phylo_vcv[[3]]

```

Activity:

```

# rerun models
# a. activity
# SMD
MLMR_mods_pers_SSD_invert_activity_SMD <- rma.mv(SMD_yi_flip ~
SSD_index, V = SMD_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_invert), control=list(optimizer="optim"),
test = "t", data = invert_activity)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning: Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_invert_activity_SMD

##
## Multivariate Meta-Analysis Model (k = 165; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  2.1874  1.4790    18    no      study_ID    no
## sigma^2.2  0.0001  0.0104    16    no  spp_name_phylo  yes
## sigma^2.3  0.1562  0.3952   165    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 163) = 1081.7241, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 163) = 0.7140, p-val = 0.3993
##
## Model Results:
##

```

```
##           estimate      se      tval   df      pval      ci.lb      ci.ub
## intrcpt         0.3479  0.3670   0.9480  163   0.3445  -0.3767   1.0725
## SSD_index      -0.6862  0.8120  -0.8450  163   0.3993  -2.2896   0.9173
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# LnCVR
MLMR_mods_pers_SSD_invert_activity_lncvr <- rma.mv(CVR_yi ~ SSD_index,
V = CVR_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_invert), control=list(optimizer="optim"),
test = "t", data =
invert_activity)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## Rows with NAs omitted from model fitting.

MLMR_mods_pers_SSD_invert_activity_lncvr

##
## Multivariate Meta-Analysis Model (k = 165; method: REML)
##
## Variance Components:
##
##           estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.1193  0.3454    18    no      study_ID    no
## sigma^2.2  0.0000  0.0035    16    no  spp_name_phylo  yes
## sigma^2.3  0.0591  0.2431   165    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 163) = 486.7410, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 163) = 0.4392, p-val = 0.5085
##
## Model Results:
##
##           estimate      se      tval   df      pval      ci.lb      ci.ub
## intrcpt        -0.0278  0.1074  -0.2589  163   0.7960  -0.2400   0.1843
## SSD_index       0.2685  0.4052   0.6627  163   0.5085  -0.5315   1.0685
##
```

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

Boldness:

```

# SMD
MLMR_mods_pers_SSD_invert_bold_SMD <- rma.mv(SMD_yi_flip ~ SSD_index, V
= SMD_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_invert), control=list(optimizer="optim"),
test = "t", data = invert_bold)

## Warning in rma.mv(SMD_yi_flip ~ SSD_index, V = SMD_vi, random = list(~1 |
:
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

MLMR_mods_pers_SSD_invert_bold_SMD

##
## Multivariate Meta-Analysis Model (k = 164; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0822  0.2867    23    no      study_ID    no
## sigma^2.2  0.0000  0.0020    23    no  spp_name_phylo  yes
## sigma^2.3  0.0650  0.2550   164    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 162) = 513.4222, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 162) = 0.1533, p-val = 0.6959
##
## Model Results:
##
##      estimate      se    tval    df    pval    ci.lb    ci.ub
## intrcpt      0.0985  0.0823  1.1967  162  0.2332  -0.0640  0.2611
## SSD_index    0.1313  0.3354  0.3915  162  0.6959  -0.5310  0.7936
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# LnCVR
MLMR_mods_pers_SSD_invert_bold_lncvr <- rma.mv(CVR_yi ~ SSD_index, V =
CVR_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_invert), control=list(optimizer="optim"),
test = "t", data = invert_bold)

```

```

~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylo_vcv_invert), control=list(optimizer="optim"),
                                test = "t", data = invert_bold)

## Warning in rma.mv(CVR_yi ~ SSD_index, V = CVR_vi, random = list(~1 |
study_ID, :
## There are rows/columns in the 'R' matrix for 'spp_name_phylo' for which
there
## are no data.

      MLMR_mods_pers_SSD_invert_bold_lncvr

##
## Multivariate Meta-Analysis Model (k = 164; method: REML)
##
## Variance Components:
##
##           estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0383   0.1957    23    no      study_ID    no
## sigma^2.2  0.0000   0.0015    23    no  spp_name_phylo  yes
## sigma^2.3  0.0378   0.1945   164    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 162) = 380.7049, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 162) = 0.0013, p-val = 0.9716
##
## Model Results:
##
##           estimate      se      tval   df      pval      ci.lb      ci.ub
## intrcpt      -0.0145  0.0607  -0.2384  162   0.8119  -0.1344   0.1055
## SSD_index    -0.0089  0.2503  -0.0356  162   0.9716  -0.5032   0.4854
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Multiple testing

Now we can extract the p-values from our intercept models, personality trait models, and SSD subset models to adjust p-values using the false discovery rate method. This method uses the `p.adjust` function to adjust p-values to account for multiple testing.

```

## Extract p-values from SSD subset models (reported in main text)

# list
p.SMD_SSD <- c(0.69, 0.03, 0.86, 0.04, 0.34, 0.25, 0.99, 0.85, 0.48, 0.74,
0.68, 0.63, 0.29, 0.57, 0.34, 0.40, 0.23, 0.70)
p.lnCVR_SSD <- c(0.60, 0.72, 0.52, 0.91, 0.82, 0.15, 0.69, 0.61, 0.24,

```

0.43, 0.05, 0.70, 0.96, 0.58, 0.80, 0.51, 0.81, 0.97)

p adjustment on our 3 hypothesis-testing models

#SMD

`p.adjust(p = c(p.SMD_intercept, p.SMD_pers, p.SMD_SSD), method = "fdr")`

```
##      bird      fish  invertebrate      mammal      reptilia
## 0.8533333 0.8533333 0.8533333 0.8708500 0.8533333
##      bird1      bird2      bird3      bird4      bird5
## 0.8708500 0.8533333 0.8533333 0.8708500 0.4800000
##      fish1      fish2      fish3      fish4      fish5
## 0.8708500 0.8533333 0.8708500 0.8533333 0.8708500
## invertebrate1 invertebrate2 invertebrate3 invertebrate4 invertebrate5
## 0.8533333 0.8533333 0.8533333 0.9195744 0.8533333
##      mammal1      mammal2      mammal3      mammal4      mammal5
## 0.8533333 0.8708500 0.8533333 0.9292872 0.8708500
##      reptilia1      reptilia2      reptilia3      reptilia4      reptilia5
## 0.9195744 0.8533333 0.8533333 0.4800000 0.9900000
##
## 0.8708500 0.4800000 0.9195744 0.4800000 0.8533333
##
## 0.8533333 0.9900000 0.9195744 0.8533333 0.8708500
##
## 0.8708500 0.8708500 0.8533333 0.8708500 0.8533333
##
## 0.8533333 0.8533333 0.8708500
```

#Lncvr

`p.adjust(p = c(p.lnCVR_intercept, p.lnCVR_pers, p.lnCVR_SSD), method = "fdr")`

```
##      bird      fish  invertebrate      mammal      reptilia
## 0.9513857 0.9513857 0.9513857 0.9513857 0.9513857
##      bird1      bird2      bird3      bird4      bird5
## 0.9513857 0.9513857 0.9513857 0.4948879 0.9513857
##      fish1      fish2      fish3      fish4      fish5
## 0.9513857 0.8505594 0.9513857 0.9513857 0.4948879
## invertebrate1 invertebrate2 invertebrate3 invertebrate4 invertebrate5
## 0.9513857 0.9513857 0.9513857 0.9513857 0.9513857
##      mammal1      mammal2      mammal3      mammal4      mammal5
## 0.9513857 0.9513857 0.9513857 0.9513857 0.9513857
##      reptilia1      reptilia2      reptilia3      reptilia4      reptilia5
## 0.9513857 0.7414563 0.9513857 0.9513857 0.9513857
##
## 0.9513857 0.9513857 0.9513857 0.9513857 0.9513857
##
## 0.9513857 0.9513857 0.9513857 0.9513857 0.9513857
##
## 0.7414563 0.9513857 0.9700000 0.9513857 0.9513857
```

```
##
##      0.9513857      0.9513857      0.9700000

# these p-values are in the order presented in tables, so easy to replace
old p-values with new ones
```

Exploratory analyses

We collected some additional information from the literature (mating system) and from studies that we expected would influence sex differences. These analyses are strictly exploratory and just compare categorical moderator terms.

mating system

Do effect sizes from monogamous or multiply-mating species differ? Model summaries presented in Supplementary Table S3.

```
# what have we got to work with?
pers_new %>%
  group_by(taxo_group, mating_system) %>%
  filter(!is.na(mating_system))%>%
  summarise(n = n(), studies = length(unique(study_ID)), species =
length(unique(spp_name_phylo))) # make a table of numbers

## `summarise()` has grouped output by 'taxo_group'. You can override using
the `.groups` argument.

## # A tibble: 10 × 5
## # Groups:   taxo_group [5]
##   taxo_group mating_system      n studies species
##   <chr>      <chr>      <int> <int> <int>
## 1 bird      monogamy      370    43    92
## 2 bird      multiple mating  107     9    12
## 3 fish      monogamy       65     8     5
## 4 fish      multiple mating  411    34    15
## 5 invertebrate monogamy       22     3     3
## 6 invertebrate multiple mating  369    35    29
## 7 mammal     monogamy      105     9     8
## 8 mammal     multiple mating  517    52    33
## 9 reptilia   monogamy        2     1     1
## 10 reptilia  multiple mating   53     7     6

# reload model output
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMR_models_pers_mating_system <- meta_model_fits(pers_new, phylo_vcv,
type = "pers_mate")
```

```

    saveRDS(MLMR_models_pers_mating_system,
"./output/MLMR_models_pers_mating_system")
  } else{
    MLMR_models_pers_mating_system <-
readRDS("./output/MLMR_models_pers_mating_system")
  }

# Extract the SMD and lnCVR results
smd_mods_mating_system <- MLMR_models_pers_mating_system["SMD",]

lnCVR_mods_mating_system <- MLMR_models_pers_mating_system["lnCVR",]

```

age

Do effect sizes from adults (sexually mature) or juveniles differ? Model summaries presented in Supplementary Table S4

```

# make a table
data.frame(pers_new %>%
  group_by(taxo_group, age) %>%
  summarise(n= n(), N_spp = length(unique(spp_name_phylo)), N_studies =
length(unique(study_ID))))

## `summarise()` has grouped output by 'taxo_group'. You can override using
the `.groups` argument.

##      taxo_group    age    n N_spp N_studies
## 1         bird  adult 323   105         43
## 2         bird juvenile 157    10         13
## 3         fish  adult 483    22         43
## 4         fish juvenile   7     3          3
## 5  invertebrate  adult 384    36         37
## 6  invertebrate juvenile  39     3          3
## 7         mammal  adult 470    38         48
## 8         mammal juvenile 204    18         19
## 9        reptilia  adult  93     9         10
## 10        reptilia juvenile   2     1          1

# reload model output:
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMR_models_pers_age <- meta_model_fits(pers_new, phylo_vcv, type =
"age")
  saveRDS(MLMR_models_pers_age, "./output/MLMR_models_pers_age")
} else{
  MLMR_models_pers_age <- readRDS("./output/MLMR_models_pers_age")
}

```



```
# Extract the SMD and lnCVR results
smd_mods_pers_age <- MLMR_models_pers_age["SMD",]

lnCVR_mods_pers_age <- MLMR_models_pers_age["lnCVR",]
```

population

Do effect sizes from wild animals or lab animals differ? Model summaries presented in Supplementary Table S5.

```
# table
data.frame(pers_new %>%
  group_by(taxo_group, population) %>%
  summarise(n = n(), N_spp = length(unique(spp_name_phylo)), N_studies =
length(unique(study_ID))))

## `summarise()` has grouped output by 'taxo_group'. You can override using
the `.groups` argument.

##      taxo_group population    n N_spp N_studies
## 1      bird      field 263    100      34
## 2      bird      lab 217      9      16
## 3      fish      field 189     13      17
## 4      fish      lab 301     12      28
## 5 invertebrate      field 176     24      21
## 6 invertebrate      lab 247     13      17
## 7      mammal      field 181     23      26
## 8      mammal      lab 493     26      38
## 9      reptilia      field  81      9      10
## 10     reptilia      lab  14      2       2

# reload model output:
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMR_models_pers_pop <- meta_model_fits(pers_new, phylo_vcv, type =
"pop")
  saveRDS(MLMR_models_pers_pop, "./output/MLMR_models_pers_pop")
} else{
  MLMR_models_pers_pop <- readRDS("./output/MLMR_models_pers_pop")
}

# Extract the SMD and lnCVR results
smd_mods_pers_pop <- MLMR_models_pers_pop["SMD",]

lnCVR_mods_pers_pop <- MLMR_models_pers_pop["lnCVR",]
```

study environment

Do effect sizes collected in the wild or the lab differ? Model summaries presented in Supplementary Table S6.

```
# table
data.frame(pers_new %>%
  group_by(taxo_group, study_environment) %>%
  summarise(n = n(), N_spp = length(unique(spp_name_phylo)), N_studies =
length(unique(study_ID))))

## `summarise()` has grouped output by 'taxo_group'. You can override using
the `.groups` argument.

##      taxo_group study_environment    n N_spp N_studies
## 1      bird      field    224   100      29
## 2      bird      lab     256    11      22
## 3      fish      field     68     5       5
## 4      fish      lab    422    17      39
## 5 invertebrate  field     14     4       1
## 6 invertebrate  lab    409    35      38
## 7      mammal    field    115    18      22
## 8      mammal    lab    559    30      40
## 9      reptilia  field     5     2       2
## 10     reptilia  lab     90     8       9

# reload model output:
rerun_models == FALSE

## [1] TRUE

if(rerun_models == TRUE){
  MLMR_models_pers_environ <- meta_model_fits(pers_new, phylo_vcv, type =
"environ")
  saveRDS(MLMR_models_pers_environ, "./output/MLMR_models_pers_environ")
} else{
  MLMR_models_pers_environ <- readRDS("./output/MLMR_models_pers_environ")
}

# Extract the SMD and lnCVR results
smd_mods_pers_enviro <- MLMR_models_pers_environ["SMD",]

lnCVR_mods_pers_enviro <- MLMR_models_pers_environ["lnCVR",]
```

study type

Do effect sizes from observational or experimental study design differ? Model summaries presented in Supplementary Table S7.

```
# Let's see what we have to work with
data.frame(pers_new %>%
```

```

group_by(taxo_group, study_type) %>%
  summarise(N_spp = length(unique(spp_name_phylo)), N_studies =
length(unique(study_ID)))

## `summarise()` has grouped output by 'taxo_group'. You can override using
the `.groups` argument.

##      taxo_group  study_type N_spp N_studies
## 1      bird experimental   17      32
## 2      bird  observation   94      18
## 3      fish experimental   19      41
## 4      fish  observation    3       3
## 5 invertebrate experimental   37      38
## 6      mammal experimental   37      47
## 7      mammal  observation   12      14
## 8      reptilia experimental    8       9
## 9      reptilia  observation    2       2

  # inverts only have experimental observations, so need to exclude inverts
from this analysis
  # because our phylo_vcv matrix is in a list that is hard to drop elements
from, let's just run each model individually

# 1. Mammals
# Subset data
pers_new_mammal <- as.data.frame(pers_new %>%
  filter(taxo_group == "mammal"))

# Run models with just study type as moderator:
#SMD
MLMR_mods_pers_studytype_mammal_SMD <- rma.mv(SMD_yi_flip ~ study_type, V
= SMD_vi,
                                              random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                              R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                              test = "t", data =
pers_new_mammal)

MLMR_mods_pers_studytype_mammal_SMD

##
## Multivariate Meta-Analysis Model (k = 674; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.1051  0.3242   61    no      study_ID   no
## sigma^2.2  0.0094  0.0971   45    no  spp_name_phylo  yes
## sigma^2.3  0.1570  0.3963  674    no          obs    no

```

```

##
## Test for Residual Heterogeneity:
## QE(df = 672) = 2198.5222, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 672) = 9.6851, p-val = 0.0019
##
## Model Results:
##
##               estimate      se      tval   df    pval    ci.lb
ci.ub
## intrcpt          -0.0067  0.0913  -0.0734  672   0.9415  -0.1860
0.1726
## study_typeobservation    0.4092  0.1315   3.1121  672   0.0019   0.1510
0.6674
##
## intrcpt
## study_typeobservation  **
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

  #LnCVR
  MLMR_mods_pers_studytype_mammal_lncvr <- rma.mv(CVR_yi ~ study_type, V =
CVR_vi,
                                                    random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                                    R =
list(spp_name_phylo=phylo_vcv_mammal), control=list(optimizer="optim"),
                                                    test = "t", data =
pers_new_mammal)

  MLMR_mods_pers_studytype_mammal_lncvr

##
## Multivariate Meta-Analysis Model (k = 674; method: REML)
##
## Variance Components:
##
##               estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0356  0.1888    61    no      study_ID    no
## sigma^2.2  0.0436  0.2088    45    no  spp_name_phylo  yes
## sigma^2.3  0.0338  0.1837   674    no              obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 672) = 1061.6294, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 672) = 0.5661, p-val = 0.4521
##

```

```

## Model Results:
##
##              estimate      se    tval   df    pval    ci.lb
ci.ub
## intrcpt          0.0298  0.1361  0.2188  672  0.8269 -0.2375
0.2970
## study_typeobservation  0.0771  0.1024  0.7524  672  0.4521 -0.1240
0.2781
##
## intrcpt
## study_typeobservation
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# 2. BIRDS
# subset dataset
pers_new_bird <- as.data.frame(pers_new %>%
  filter(taxo_group == "bird"))

# rerun models
#SMD
MLMR_mods_pers_studytype_bird_SMD <- rma.mv(SMD_yi_flip ~ study_type,
V = SMD_vi,
                                random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylo_vcv_bird), control=list(optimizer="optim"),
                                test = "t", data =
pers_new_bird)

MLMR_mods_pers_studytype_bird_SMD

##
## Multivariate Meta-Analysis Model (k = 480; method: REML)
##
## Variance Components:
##
##      estim    sqrt  nlvls  fixed      factor    R
## sigma^2.1  0.6650  0.8155    50    no      study_ID  no
## sigma^2.2  0.0000  0.0038   106    no  spp_name_phylo  yes
## sigma^2.3  0.1203  0.3469   480    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 478) = 2378.8387, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 478) = 1.0068, p-val = 0.3162
##
## Model Results:

```

```

##
##              estimate      se      tval   df    pval    ci.lb
ci.ub
## intrcpt              -0.0210  0.1515  -0.1389  478   0.8896  -0.3187
0.2766
## study_typeobservation -0.2540  0.2532  -1.0034  478   0.3162  -0.7515
0.2434
##
## intrcpt
## study_typeobservation
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#LnCVR
MLMR_mods_pers_studytype_bird_lncvr <- rma.mv(CVR_yi ~ study_type, V =
CVR_vi,
                                             random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                             R =
list(spp_name_phylo=phylo_vcv_bird), control=list(optimizer="optim"),
                                             test = "t", data =
pers_new_bird)

MLMR_mods_pers_studytype_bird_lncvr

##
## Multivariate Meta-Analysis Model (k = 480; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.2537  0.5036    50    no      study_ID    no
## sigma^2.2  0.0001  0.0086   106    no  spp_name_phylo  yes
## sigma^2.3  0.3766  0.6137   480    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 478) = 3186.9869, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 478) = 1.7076, p-val = 0.1919
##
## Model Results:
##
##              estimate      se      tval   df    pval    ci.lb
ci.ub
## intrcpt              0.0436  0.1109   0.3929  478   0.6946  -0.1744
0.2615
## study_typeobservation -0.2451  0.1876  -1.3067  478   0.1919  -0.6138
0.1235

```

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

# 3. FISH
# subset dataset
pers_new_fish <- as.data.frame(pers_new %>%
  filter(taxo_group == "fish"))

# rerun models
#SMD
MLMR_mods_pers_studytype_fish_SMD <- rma.mv(SMD_yi_flip ~ study_type, V
= SMD_vi,
                                             random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                             R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
                                             test = "t", data = pers_new_fish)

MLMR_mods_pers_studytype_fish_SMD

##
## Multivariate Meta-Analysis Model (k = 490; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.5997  0.7744    44    no      study_ID    no
## sigma^2.2  0.0440  0.2098    22    no  spp_name_phylo  yes
## sigma^2.3  0.1091  0.3303   490    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 488) = 1523.5807, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 488) = 0.0188, p-val = 0.8911
##
## Model Results:
##
##              estimate      se      tval   df    pval    ci.lb
ci.ub
## intrcpt              0.1825  0.2106   0.8669  488  0.3864  -0.2312
0.5962
## study_typeobservation -0.0657  0.4798  -0.1369  488  0.8911  -1.0085
0.8771
##
## intrcpt
```

```

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

      #LnCVR
      MLMR_mods_pers_studytype_fish_lncvr <- rma.mv(CVR_yi ~ study_type, V =
CVR_vi,
                                                    random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                                    R =
list(spp_name_phylo=phylo_vcv_fish), control=list(optimizer="optim"),
                                                    test = "t", data = pers_new_fish)

      MLMR_mods_pers_studytype_fish_lncvr

##
## Multivariate Meta-Analysis Model (k = 490; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0352  0.1875    44     no      study_ID    no
## sigma^2.2  0.0017  0.0408    22     no  spp_name_phylo  yes
## sigma^2.3  0.1094  0.3307   490     no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 488) = 1122.6615, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 488) = 0.0722, p-val = 0.7882
##
## Model Results:
##
##      estimate      se      tval    df    pval    ci.lb
ci.ub
## intrcpt          -0.0025  0.0536  -0.0460  488  0.9633  -0.1078
0.1029
## study_typeobservation  -0.0399  0.1486  -0.2687  488  0.7882  -0.3319
0.2520
##
## intrcpt
## study_typeobservation
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# 4. Reptiles
# subset dataset
pers_new_reptile <- as.data.frame(pers_new %>%

```



```

    filter(taxo_group == "reptilia"))

# phylo
phylo_vcv_reptile <- phylo_vcv[[5]]

# rerun models
#SMD
MLMR_mods_pers_studytype_rep_SMD <- rma.mv(SMD_yi_flip ~ study_type, V
= SMD_vi,
                                             random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                             R =
list(spp_name_phylo=phylo_vcv_reptile), control=list(optimizer="optim"),
                                             test = "t", data =
pers_new_reptile)

MLMR_mods_pers_studytype_rep_SMD

##
## Multivariate Meta-Analysis Model (k = 95; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0000  0.0008    11    no      study_ID    no
## sigma^2.2  0.0730  0.2702    10    no  spp_name_phylo  yes
## sigma^2.3  0.0426  0.2063    95    no           obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 159.2776, p-val < .0001
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 93) = 3.4462, p-val = 0.0666
##
## Model Results:
##
##              estimate      se      tval  df      pval      ci.lb
ci.ub
## intrcpt              0.1334  0.1531   0.8715  93   0.3857  -0.1706
0.4374
## study_typeobservation -0.5085  0.2739  -1.8564  93   0.0666  -1.0524
0.0354
##
## intrcpt
## study_typeobservation .
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

#LnCVR
MLMR_mods_pers_studytype_rep_lncvr <- rma.mv(CVR_yi ~ study_type, V =
CVR_vi,
random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
R =
list(spp_name_phylo=phylo_vcv_reptile), control=list(optimizer="optim"),
test = "t", data =
pers_new_reptile)

MLMR_mods_pers_studytype_rep_lncvr

##
## Multivariate Meta-Analysis Model (k = 95; method: REML)
##
## Variance Components:
##
##      estim      sqrt  nlvls  fixed      factor      R
## sigma^2.1  0.0000  0.0003    11    no      study_ID    no
## sigma^2.2  0.0000  0.0012    10    no  spp_name_phylo  yes
## sigma^2.3  0.0000  0.0003    95    no          obs    no
##
## Test for Residual Heterogeneity:
## QE(df = 93) = 58.0505, p-val = 0.9983
##
## Test of Moderators (coefficient 2):
## F(df1 = 1, df2 = 93) = 0.1931, p-val = 0.6613
##
## Model Results:
##
##              estimate      se      tval  df      pval      ci.lb
ci.ub
## intrcpt              0.0427  0.0420   1.0173  93   0.3116  -0.0407
0.1261
## study_typeobservation -0.0621  0.1413  -0.4395  93   0.6613  -0.3426
0.2184
##
## intrcpt
## study_typeobservation
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Sensitivity analyses - Dependency matrix models

We need to refit our 3 main models accounting for any dependency resulting from the same traits measured on the same animals (likely a big source of non-independence) and any other shared covariance. We added the D matrices to the residual variance matrix as

opposed to the sampling covariance. We chose to set 3 different levels of dependency (ρ): 0.3, 0.5 and 0.8.

Model summaries are also presented in Supplementary Tables S8-S13.

```
# Create the dependency matrices; try 3 levels of rho = 0.3, 0.5, 0.8

pers_new <- data.frame(pers_new %>%
  group_by(taxo_group) %>%
  mutate(depend_n = paste0(study_ID, "_", depend)))

split_taxa <- split(pers_new, pers_new$taxo_group)

# 0.3 rho:
D_matrices_0.3 <- lapply(split_taxa, function(x) make_VCV_matrix(x, V =
x$SMD_vi, cluster = "depend_n",
                                                                    obs =
"obs", type = "cor", rho = 0.3))
# 0.5 rho:
D_matrices_0.5 <- lapply(split_taxa, function(x) make_VCV_matrix(x, V =
x$SMD_vi, cluster = "depend_n",
                                                                    obs =
"obs", type = "cor", rho = 0.5))
# 0.8 rho:
D_matrices_0.8 <- lapply(split_taxa, function(x) make_VCV_matrix(x, V =
x$SMD_vi, cluster = "depend_n",
                                                                    obs =
"obs", type = "cor", rho = 0.8))
```

Intercept-only models with D matrices

Model output is presented in Supplementary Tables S8-S10 in the Supporting Information.

```
# 1. Intercept only models
# rho = 0.3
int_0.3 <- fit_int_MLMAmodel_D(pers_new, phylo_vcv, D_matrices_0.3)

smd_mods_D_0.3 <- int_0.3[["SMD"]]
lnCVR_mods_D_0.3 <- int_0.3[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_D_0.3 <- plyr::ldply(lapply(smd_mods_D_0.3,
function(x) print(mod_results(x, mod = "Int"))))
MLMA_estimates_lnCVR_D_0.3 <- plyr::ldply(lapply(lnCVR_mods_D_0.3,
function(x) print(mod_results(x, mod =
"Int"))))

# rho = 0.5
int_0.5 <- fit_int_MLMAmodel_D(pers_new, phylo_vcv, D_matrices_0.5)
```

```

smd_mods_D_0.5 <- int_0.5[["SMD"]]
lnCVR_mods_D_0.5 <- int_0.5[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_D_0.5 <- plyr::ldply(lapply(smd_mods_D_0.5,
function(x) print(mod_results(x, mod = "Int"))))
MLMA_estimates_lnCVR_D_0.5 <- plyr::ldply(lapply(lnCVR_mods_D_0.5,
function(x) print(mod_results(x, mod =
"Int"))))

# rho = 0.8
int_0.8 <- fit_int_MLMAmodel_D(pers_new, phylo_vcv, D_matrices_0.8)

smd_mods_D_0.8 <- int_0.8[["SMD"]]
lnCVR_mods_D_0.8 <- int_0.8[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_D_0.8 <- plyr::ldply(lapply(smd_mods_D_0.8,
function(x) print(mod_results(x, mod = "Int"))))
MLMA_estimates_lnCVR_D_0.8 <- plyr::ldply(lapply(lnCVR_mods_D_0.8,
function(x) print(mod_results(x, mod =
"Int"))))

```

Personality trait models with D matrices

Model output is presented in Supplementary Tables S11-S13 in the Supporting Information.

2. Personality Trait models

```

# rho = 0.3
pers_0.3 <- fit_int_MLMAmodel_D_pers(pers_new, phylo_vcv, D_matrices_0.3)

smd_mods_D_pers_0.3 <- pers_0.3[["SMD"]]
lnCVR_mods_D_pers_0.3 <- pers_0.3[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_pers_D_0.3 <- plyr::ldply(lapply(smd_mods_D_pers_0.3,
function(x) print(mod_results(x, mod =
"personality_trait"))))
MLMA_estimates_lnCVR_pers_D_0.3 <-
plyr::ldply(lapply(lnCVR_mods_D_pers_0.3,
function(x) print(mod_results(x, mod =
"personality_trait"))))

# rho = 0.5
pers_0.5 <- fit_int_MLMAmodel_D_pers(pers_new, phylo_vcv, D_matrices_0.5)

smd_mods_D_pers_0.5 <- pers_0.5[["SMD"]]

```

```

lnCVR_mods_D_pers_0.5 <- pers_0.5[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_pers_D_0.5 <- plyr::ldply(lapply(smd_mods_D_pers_0.5,
function(x) print(mod_results(x, mod =
"personality_trait"))))
MLMA_estimates_lnCVR_pers_D_0.5 <-
plyr::ldply(lapply(lnCVR_mods_D_pers_0.5,
function(x) print(mod_results(x, mod =
"personality_trait"))))

# rho = 0.8
pers_0.8 <- fit_int_MLMAmodel_D_pers(pers_new, phylo_vcv, D_matrices_0.8)

smd_mods_D_pers_0.8 <- pers_0.8[["SMD"]]
lnCVR_mods_D_pers_0.8 <- pers_0.8[["lnCVR"]]

# prediction intervals
MLMA_estimates_SMD_pers_D_0.8 <- plyr::ldply(lapply(smd_mods_D_pers_0.8,
function(x) print(mod_results(x, mod =
"personality_trait"))))
MLMA_estimates_lnCVR_pers_D_0.8 <-
plyr::ldply(lapply(lnCVR_mods_D_pers_0.8,
function(x) print(mod_results(x, mod =
"personality_trait"))))

```

Personality * SSD models

These models were just to check since we don't really interpret the interaction models.

```

# 3. Pers Trait * SSD models
# just use the full interaction models here since this is just a check
# won't bother with prediction intervals here since these models aren't
really for interpretation

```

```

# rho = 0.3
ssd_0.3 <- fit_int_MLMAmodel_D_pers_ssd(pers_new, phylo_vcv,
D_matrices_0.3)

split_taxa <- split(pers_new, pers_new$taxo_group)
smd_mods_D_pers_ssd_0.3 <- ssd_0.3[["SMD"]]
lnCVR_mods_D_pers_ssd_0.3 <- ssd_0.3[["lnCVR"]]

# rho = 0.5
ssd_0.5 <- fit_int_MLMAmodel_D_pers_ssd(pers_new, phylo_vcv,
D_matrices_0.5)

split_taxa <- split(pers_new, pers_new$taxo_group)
smd_mods_D_pers_ssd_0.5 <- ssd_0.5[["SMD"]]

```

```

lnCVR_mods_D_pers_ssd_0.5 <- ssd_0.5[["lnCVR"]]

# rho = 0.8
ssd_0.8 <- fit_int_MLMAmodel_D_pers_ssd(pers_new, phylo_vcv,
D_matrices_0.8)

split_taxa <- split(pers_new, pers_new$taxo_group)
smd_mods_D_pers_ssd_0.8 <- ssd_0.8[["SMD"]]
lnCVR_mods_D_pers_ssd_0.8 <- ssd_0.8[["lnCVR"]]

```

Sensitivity analyses - Publication Bias

We can use: 1) funnel plots to look for asymmetry across all effect sizes for both SMD and lnCVR, and 2) Egger's test which performs a regression test on our funnel plots ... but is not useful when there is high heterogeneity NOT caused by publication bias (which is the case for our data).

Since our data has very high heterogeneity, we instead included the inverse of the 'effective sample size' as a moderator term in our full model (personality trait model) to see if study precision is driving effect size patterns. The logic here is that studies with low or high precision can have a significant influence and so including precision as a moderator will allow us to see if precision is significant (and which direction). See Nakagawa et al. 2021 for more info (reference in main text).

Model summaries are presented in Supplementary Table S18.

```

### NEW METHOD OF PUBLICATION BIAS FROM NAKAGAWA ET AL 2021 - PREPRINT

# calculating the inverse of the 'effective sample size' to account for
# unbalanced sampling
pers_new$inv_n_tilda <- with(pers_new, ((female_n +
male_n)/(female_n*male_n)))
pers_new$sqrt_inv_n_tilda <- with(pers_new, (sqrt(inv_n_tilda))) # use
# this in the model

if(rerun_models == TRUE){
  MLMR_models_pers_pubbias <- meta_model_fits(pers_new, phylo_vcv, type =
"pubbias")
  saveRDS(MLMR_models_pers_pubbias, "./output/MLMR_models_pers_pubbias")
} else{
  MLMR_models_pers_pubbias <- readRDS("./output/MLMR_models_pers_pubbias")
}

# Extract the SMD and lnCVR results
smd_mods_pubbias <- MLMR_models_pers_pubbias["SMD",]

lnCVR_mods_pubbias <- MLMR_models_pers_pubbias["lnCVR",]

```

Exploratory analyses - Heterogamety and taxo group

There was a trend for male mammals to be more variable than females and female birds to be more variable than males, for some of the five personality traits. To better compare the direction of these effect sizes we decided post hoc to conduct an exploratory analysis with personality trait type and taxonomic group as moderator terms to compare birds and mammals (males homogametic or heterogametic, respectively). To do this, we first combined the bird and mammal phylo correlation matrices together (assuming no phylo heritability across the taxo groups - since phylo did not really explain heterogeneity it shouldn't matter). We then created an interaction MLMR model with personality trait * taxa (no intercept) to get slope estimates for each of the traits for mammals and birds separately.

From this model, we then compared each of the five traits for birds and mammals using a post hoc Tukey pairwise comparison to test whether birds and mammals were significantly different from each other.

Model summaries are presented in Supplementary Table S19.

```
# install packages to make diagonal matrix and to make multiple comparisons
library(multcomp)
library(Matrix)

# Create block diag phylogeny
phylogeny <- Matrix::bdiag(phylo_vcv_bird, phylo_vcv_mammal) # use this as
the phylo vcv in the model

# needs to have colnames for use in random effects model
dimnames(phylogeny) <- Map(c, dimnames(phylo_vcv_bird),
dimnames(phylo_vcv_mammal))

# only include bird and mammal data
pers_new_contrast <- as.data.frame(pers_new %>%
                                     filter(taxo_group == "mammal" |
taxo_group == "bird") %>%
                                     mutate(sp_pers =
interaction(personality_trait, taxo_group)))

# 1. intercept only model
contrast_birdmammal_lncvr_int <- rma.mv(CVR_yi ~ taxo_group, V = CVR_vi,
                                     random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                     R =
list(spp_name_phylo=phylogeny), control=list(optimizer="optim"),
                                     test = "t", data =
pers_new_contrast)

# 2. personality trait model
```

```

# creating the model - with pers trait and taxo group as mods

#lnCVR model only
contrast_birdmammal_lncvr <- rma.mv(CVR_yi ~
personality_trait*taxo_group, V = CVR_vi,
                                random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylogeny), control=list(optimizer="optim"),
                                test = "t", data =
pers_new_contrast)

# model with interaction only to check output of model above
contrast_birdmammal_lncvr_2 <- rma.mv(CVR_yi ~ sp_pers -1, V = CVR_vi,
                                random = list(~1|study_ID,
~1|spp_name_phylo, ~1|obs),
                                R =
list(spp_name_phylo=phylogeny), control=list(optimizer="optim"),
                                test = "t", data =
pers_new_contrast)

# multiple comparison using Tukey test
summary(glht(contrast_birdmammal_lncvr, linfct = cbind(contrMat(rep(1:10),
type = "Tukey"))), test=adjusted("fdr"))

# here we are only interested in the comparisons between mammals and birds,
so: 1-6 (activity), 2-7 (aggression), 3-8 (boldness), 4-9 (exploration), and
5-10 (sociality)

```

Plots

Orchard plots of effect sizes from personality trait models

These plots use the orchaRd package to generate pretty plots where each effect size (k) is a point on the plot.

lnCVR

```

# create objects of each of the models first

# lnCVR

# Bird lnCVR
bird_lncvr <- orchard_plot(lnCVR_mods_pers[[1]], mod = "personality_trait",
xlab = "log Coefficient of Variance (lnCVR)", angle = 45, alpha = 0.5,
transfm = "none")

# Fish lnCVR

```



```

fish_lncvr <- orchard_plot(lncvr_mods_pers[[2]], mod = "personality_trait",
xlab = "log Coefficient of Variance (lnCVR)", angle = 45, alpha = 0.5,
transfm = "none")
# Invert lnCVR
invert_lncvr<- orchard_plot(lncvr_mods_pers[[3]], mod =
"personality_trait", xlab = "log Coefficient of Variance (lnCVR)", angle =
45, alpha = 0.5, transfm = "none")
# Mammal lnCVR
mammal_lncvr <- orchard_plot(lncvr_mods_pers[[4]], mod =
"personality_trait", xlab = "log Coefficient of Variance (lnCVR)", angle =
45, alpha = 0.5, transfm = "none")
# Reptile lnCVR
reptile_lncvr <- orchard_plot(lncvr_mods_pers[[5]], mod =
"personality_trait", xlab = "log Coefficient of Variance (lnCVR)", angle =
45, alpha = 0.5, transfm = "none")

```

SMD

```

# Bird SMD
bird_SMD <- orchard_plot(smd_mods_pers[[1]], mod = "personality_trait",
xlab = "Standardised mean difference", angle = 45, alpha = 0.5, transfm =
"none")
# Fish SMD
fish_SMD <- orchard_plot(smd_mods_pers[[2]], mod = "personality_trait",
xlab = "Standardised mean difference", angle = 45, alpha = 0.5, transfm =
"none")
# Invert SMD
invert_SMD<- orchard_plot(smd_mods_pers[[3]], mod = "personality_trait",
xlab = "Standardised mean difference", angle = 45, alpha = 0.5, transfm =
"none")
# Mammal SMD
mammal_SMD <- orchard_plot(smd_mods_pers[[4]], mod = "personality_trait",
xlab = "Standardised mean difference", angle = 45, alpha = 0.5, transfm =
"none")
# Reptile SMD
reptile_SMD <- orchard_plot(smd_mods_pers[[5]], mod = "personality_trait",
xlab = "Standardised mean difference", angle = 45, alpha = 0.5, transfm =
"none")

```

Putting the SMD and lnCVR plots together

Endotherms:

```

# window size for orchard plots
# the precision guides on the plots are a bit ugly, collect them to the
side and crop them out

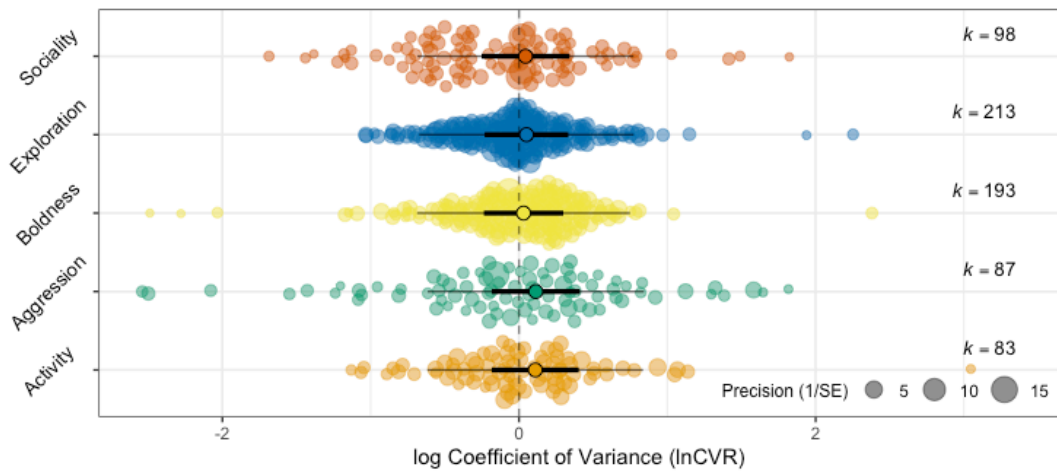
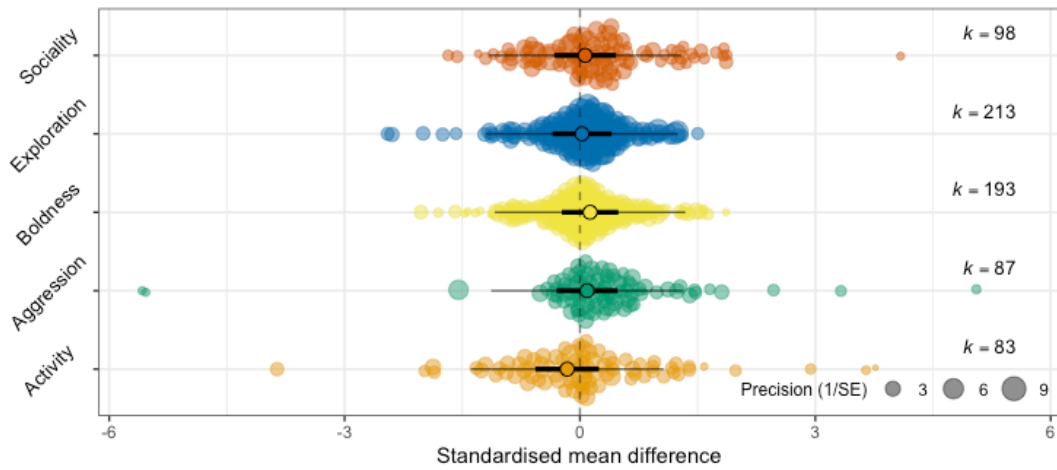
```

```
## Mammals
```

```
# dev.new(width=8,height=7,noRStudioGD = TRUE)
```

```
# place plots together
```

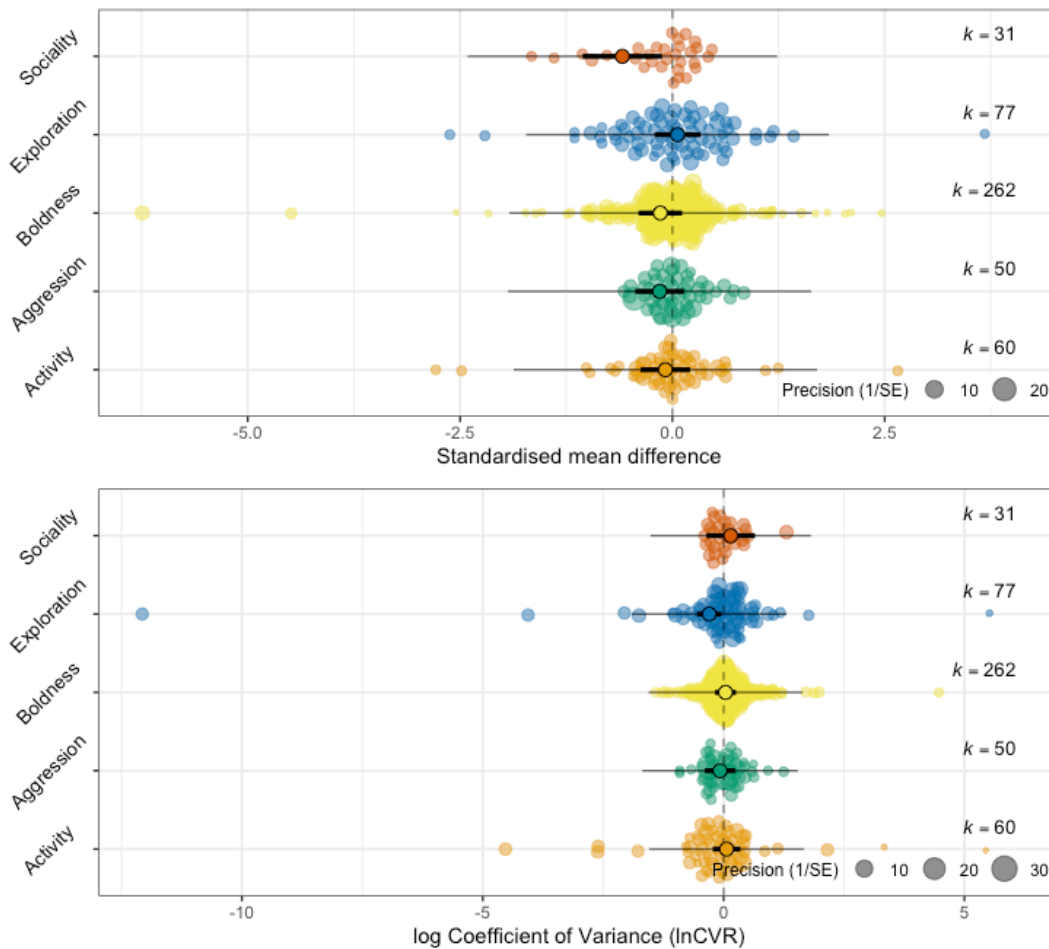
```
mammal_SMD / mammal_lncvr
```



```
# ggsave("./figs/finished figs/mammal_effects.tiff", width = 8, height = 7, units = "in") #save image
```

```
## Birds
```

```
bird_SMD / bird_lncvr
```



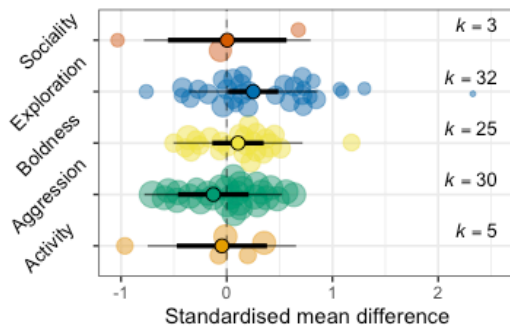
```
# ggsave("../figs/finished figs/bird_effects.tiff", width = 8, height = 7,
units = "in") #save image
```

Ectotherms:

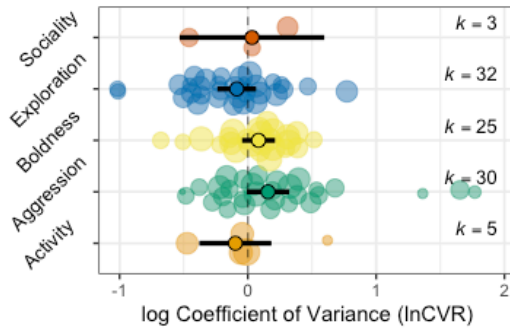
```
# window size a bit smaller for these guys
# dev.new(width=8,height=5,noRStudioGD = TRUE)
```

Reptiles and amphibians

```
reptile_SMD / reptile_lncvr / plot_layout(guides = 'collect')
```



Precision (1/SE) ● 2.0 ● 2.5 ● 3.0 ● 3.5 ● 4.0

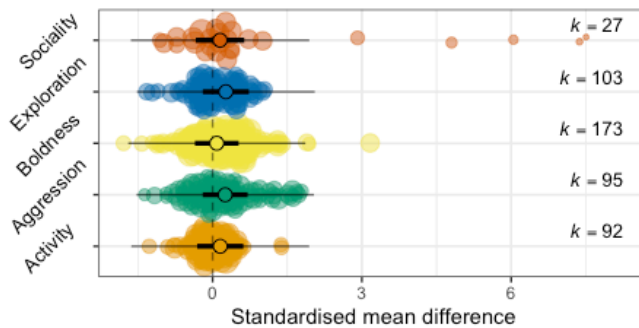


Precision (1/SE) ● 1 ● 2 ● 3 ● 4

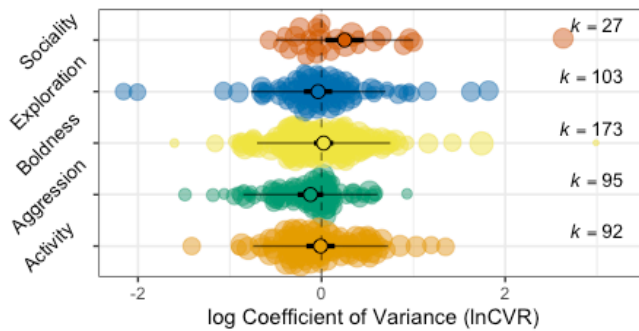
```
# ggsave("~/Documents/GitHub/sex_meta/figs/finished figs/rep_effects.tiff",
width = 8, height = 5, units = "in")
```

```
## Fish
```

```
fish_SMD / fish_lncvr / plot_layout(guides = 'collect')
```



Precision (1/SE) ● 1 ● 3 ● 5 ● 7
● 2 ● 4 ● 6

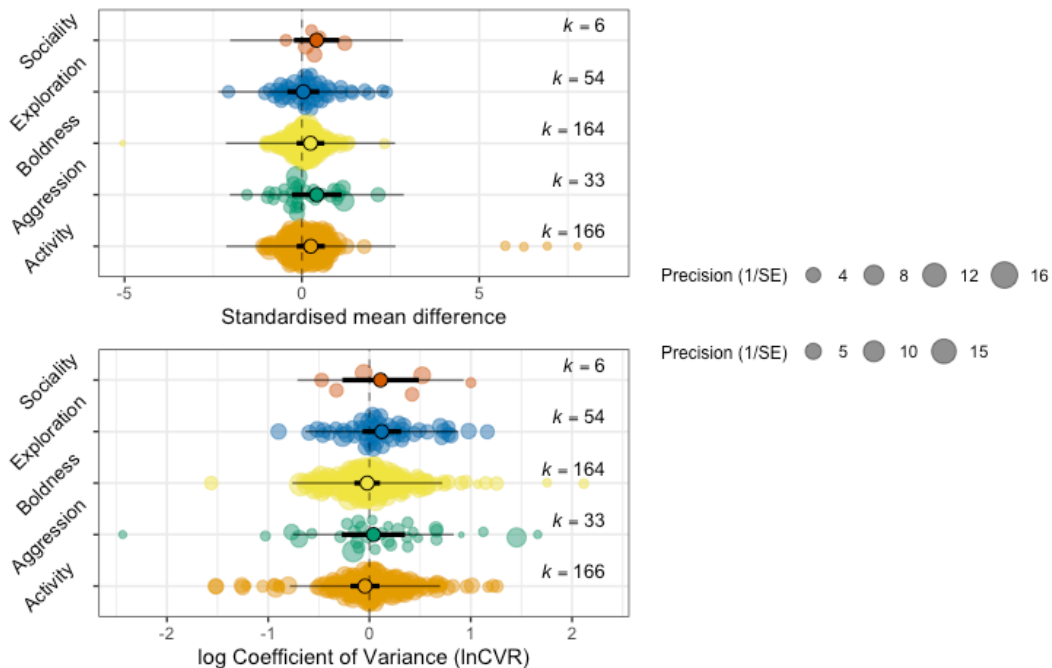


Precision (1/SE) ● 2 ● 4 ● 6

```
# ggsave("~/Documents/GitHub/sex_meta/figs/finished figs/fish_effects.tiff",
width = 8, height = 5, units = "in")
```

```
## Invertebrates
```

```
invert_SMD / invert_lncvr / plot_layout(guides = 'collect')
```



```
# ggsave("~/Documents/GitHub/sex_meta/figs/finished
figs/invert_effects.tiff", width = 8, height = 5, units = "in")
```

The precision guides will get cropped out when joining the orchard plots and phylogenies together.

Phylogenetic trees with heatmaps

Using ggtree to plot lots of complex data onto phylogenetic trees see:

<https://guangchuangyu.github.io/ggtree-book/chapter-ggtree.html> for more information about using ggtree

```
# install ggtree using this method:
source("https://bioconductor.org/biocLite.R")
BiocManager::install("ggtree")
```

```
##
```

```
## The downloaded binary packages are in
```

```
##
```

```
/var/folders/0b/pxghylq157gfhs1vrzdp2gc0000gq/T//RtmppmgALc/downloaded_packages
```

```
library(ggtree)
```

```
# Load organised SSD data using figs_data.csv
figs_data <- read.csv("./data/figs_data.csv", stringsAsFactors = FALSE)
```

bird tree

```
# subset dataset to include only birds
bird_data <- as.data.frame(figs_data %>%
  filter(taxo_group == "bird"))

# setting up the basic tree structure
# Load tree
birdtree <- read.tree("./trees/bird_species.nwk")

# prune tree to get rid of species we no longer have data for
pruned.birdtree <- drop.tip(birdtree, setdiff(birdtree$tip.label,
bird_data$spp_name_phylo))

# remove underscores from tip labels
pruned.birdtree$tip.label = gsub("_", " ", pruned.birdtree$tip.label)

# remove underscores from species name in our species data list
bird_data$spp_name_phylo = gsub("_", " ", bird_data$spp_name_phylo)

# set row names
row.names(bird_data) <- bird_data$spp_name_phylo

# define objects for the plot
species <- pruned.birdtree$tip.label

rownames(bird_data) <- pruned.birdtree$tip.label

# set window size
# dev.new(width=8, height=8, noRStudioGD = TRUE) #opens quartz window of set
size

# now need to make a matrix of effect sizes (n) for each species for each
personality trait to add to our plot!
# subset dataset
pers_bird <- as.data.frame(pers_new %>%
  filter(taxo_group == "bird"))

# make this a matrix-style dataframe
pers_bird <- data.frame(pers_bird %>%
  group_by(spp_name_phylo, personality_trait) %>%
  summarise(n = n()))
```

`summarise()` has grouped output by 'spp_name_phylo'. You can override using the `.groups` argument.

```
# remove underscores species names
pers_bird$spp_name_phylo = gsub("_", " ", pers_bird$spp_name_phylo)

# create matrix
pers_bird <- data.frame(pers_bird %>%
  spread(personality_trait, n, fill = 0))

# set species name as row name for matrix
row.names(pers_bird) <- pers_bird$spp_name_phylo

pers_bird <- pers_bird[,2:6]

# matrix
birds_matrix <- data.matrix(pers_bird)

# FINAL TREE
# making the tree
p_b1 <- ggtree(pruned.birdtree, size = 0.3, layout = 'circular',
  branch.length = 'none') %<+% bird_data +
  xlim(-40, NA) +
  geom_tippoint(aes(color=SSD_index)) +
  scale_color_gradient2(midpoint = 0, low = "red3", mid = "seashell2", high =
"deepskyblue2") +
  geom_tiplab2(size = 2.2, offset = 4, colour = "black", fontface = "italic")
+
  theme(legend.position = 'right')

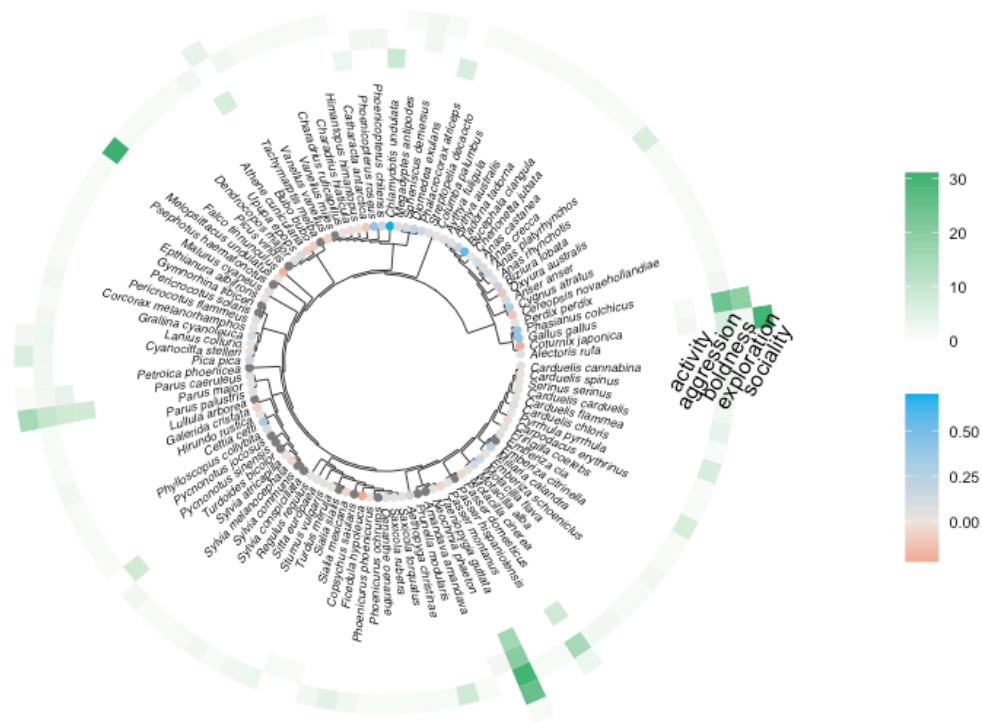
## Warning: `data_frame()` was deprecated in tibble 1.1.0.
## Please use `tibble()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
generated.

## Warning: `mutate_()` was deprecated in dplyr 0.7.0.
## Please use `mutate()` instead.
## See vignette('programming') for more help
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
generated.

# adding heatmap of traits
p_b2 <- gheatmap(p_b1, birds_matrix, offset=68, width=2, low = "white", high
= "mediumseagreen", color=NULL,
  colnames=T, colnames_angle = 60, colnames_offset_y = .1,
colnames_offset_x = .2) +
  theme(plot.tag = element_text(size = 2, face = "bold"),
```

```
legend.text = element_text(size = 8))
```

p_b2



```
# ggsave("./figs/finished figs/birdphylo.tiff", p_b2, width=8, height = 8,
units = "in")
```

mammals

```
# subset dataset to include only mammals
mammal_data <- as.data.frame(figs_data %>%
  filter(taxo_group == "mammal"))
```

```
# setting up the basic tree structure
```

```
# Load tree, set node colours
mammaltree <- read.tree("./trees/mammal_species.nwk")
```



```

# prune tree to get rid of species we no longer have data for
pruned.mammaltree <- drop.tip(mammaltree, setdiff(mammaltree$tip.label,
mammal_data$spp_name_phylo))

# remove underscores from tip labels
pruned.mammaltree$tip.label = gsub("_", " ", pruned.mammaltree$tip.label)

# set rownames for labelling tips
rownames(mammal_data) <- pruned.mammaltree$tip.label

# remove underscores from species name from mammal dataset
mammal_data$spp_name_phylo = gsub("_", " ", mammal_data$spp_name_phylo)

# set row names
row.names(mammal_data) <- mammal_data$spp_name_phylo

# set window
# dev.new(width=8,height=7,noRStudioGD = TRUE)

# make a matrix of effect sizes (n) for each species for each personality
trait to add to our plot!
# subset dataset
pers_mammal <- as.data.frame(pers_new %>%
filter(taxo_group == "mammal"))

# make this a matrix-style dataframe
pers_mammal <- data.frame(pers_mammal %>%
group_by(spp_name_phylo, personality_trait) %>%
summarise(n = n()))

## `summarise()` has grouped output by 'spp_name_phylo'. You can override
using the `.groups` argument.

# remove underscores species names
pers_mammal$spp_name_phylo = gsub("_", " ", pers_mammal$spp_name_phylo)

pers_mammal <- data.frame(pers_mammal %>%
spread(personality_trait, n, fill = 0))

row.names(pers_mammal) <- pers_mammal$spp_name_phylo

pers_mammal <- pers_mammal[,2:6]

# matrix
mammal_matrix <- data.matrix(pers_mammal)

# making the tree
p_m1 <- ggtree(pruned.mammaltree, size = 0.3, layout = 'circular',

```

```

branch.length = 'none') %<+% mammal_data +
  geom_tippoint(aes(color=SSD_index)) +
  scale_color_gradient2(midpoint = 0, low = "red3", mid = "seashell2", high =
"deepskyblue2") +
  geom_tiplab2(size = 2.5, offset = 2, colour = "black", fontface = "italic")
+
  theme(legend.position = 'right')

# adding heatmap of traits
p_m2 <- gheatmap(p_m1, mammal_matrix, offset=32, width=1.3, low = "white",
high = "mediumseagreen", color=NULL,
               colnames=T, colnames_angle = 60, colnames_offset_y = .1,
colnames_offset_x = .2) +
  theme(plot.tag = element_text(size = 9, face = "bold"),
        legend.text = element_text(size = 8))

p_m2

```



```

# ggsave("./figs/finished figs/mammalphylo.tiff", p_m2, width=8, height = 7,
units = "in")

```

reptiles

```
# subset dataset to include only reptiles
rep_data <- as.data.frame(figs_data %>%
  filter(taxo_group == "reptilia"))

row.names(rep_data) <- rep_data$spp_name_phylo

# setting up the basic tree structure

# load tree, set node colours
reptree <- read.tree("./trees/reptile_species.nwk")

# prune tree to get rid of species we no longer have data for
pruned.reptree <- drop.tip(reptree, setdiff(reptree$tip.label,
rep_data$spp_name_phylo))

# remove underscores from tip labels
pruned.reptree$tip.label = gsub("_", " ", pruned.reptree$tip.label)

# set rownames for labelling tips
rownames(rep_data) <- pruned.reptree$tip.label

# remove underscores from species name from mammal dataset
rep_data$spp_name_phylo = gsub("_", " ", rep_data$spp_name_phylo)

# set window size
#dev.new(width=7,height=5,noRStudioGD = TRUE)

# tree structure
p3 <- ggtree(pruned.reptree, branch.length='none', size = 0.3,
layout='circular') %<+% rep_data +
  geom_tippoint(aes(color=SSD_index)) +
  scale_color_gradient2(midpoint = 0, low = "red3", mid = "seashell2", high =
"deepskyblue2") +
  geom_tiplab2(align=T, linetype=NA, size=2.5, offset=4, hjust=0, colour =
"black", fontface = "italic")

# make a matrix of effect sizes (n) for each species for each personality
trait to add to our plot!
# subset dataset
pers_rep <- as.data.frame(pers_new %>%
  filter(taxo_group == "reptilia"))

# make this a matrix-style dataframe
pers_rep <- data.frame(pers_rep %>%
  group_by(spp_name_phylo, personality_trait) %>%
  summarise(n = n()))
```

`summarise()` has grouped output by 'spp_name_phylo'. You can override using the `.groups` argument.

```
# remove underscores from species name from mammal dataset
pers_rep$spp_name_phylo = gsub("_", " ", pers_rep$spp_name_phylo)

pers_rep <- data.frame(pers_rep %>%
  spread(personality_trait, n, fill = 0))

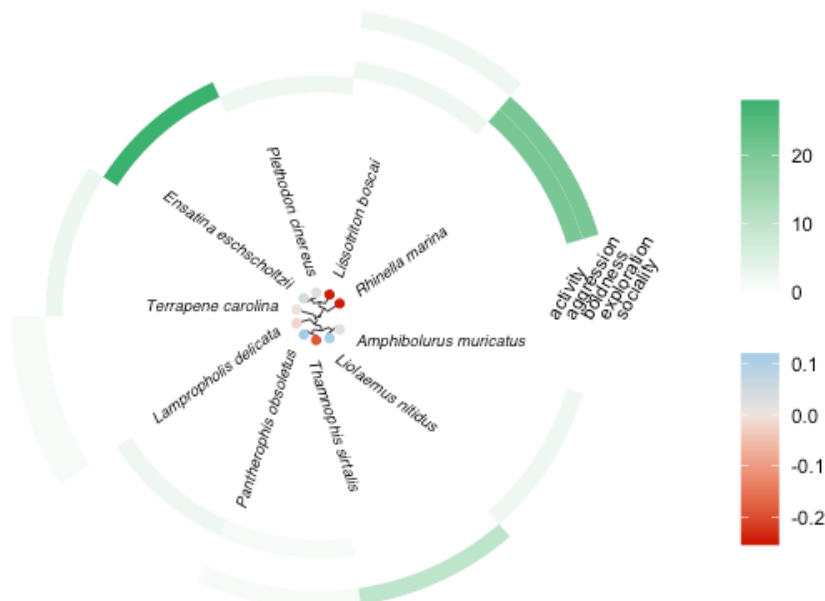
row.names(pers_rep) <- pers_rep$spp_name_phylo

pers_rep <- pers_rep[,2:6]

# matrix
rep_matrix <- data.matrix(pers_rep)

# add the heatmap data to our plot
rep_plot <- gheatmap(p3, rep_matrix, offset = 40, width = 3.5,
  low = "white", high = "mediumseagreen", color=NULL,
  colnames_position="top",
  colnames_angle=60, colnames_offset_y = 0,
  hjust=0, font.size=3) #just not aligning properly

rep_plot
```



```
# ggsave("./figs/finished figs/repphylo.tiff", rep_plot, width=7, height = 5,  
units = "in")
```

fish

```
# subset dataset to include only fish  
fish_data <- as.data.frame(figs_data %>%  
  filter(taxo_group == "fish"))  
  
# window size  
#dev.new(width=8,height=6,noRStudioGD = TRUE)  
  
# setting up the basic tree structure  
  
# Load tree  
fishtree <- read.tree("./trees/fish_species.nwk")  
  
# prune tree to get rid of species we no longer have data for  
pruned.fishtree <- drop.tip(fishtree, setdiff(fishtree$tip.label,  
fish_data$spp_name_phylo))  
  
# remove underscores from tip labels  
pruned.fishtree$tip.label = gsub("_", " ", pruned.fishtree$tip.label)  
  
# set rownames for labelling tips  
rownames(fish_data) <- pruned.fishtree$tip.label  
  
# remove underscores from species name from fish dataset  
fish_data$spp_name_phylo = gsub("_", " ", fish_data$spp_name_phylo)  
  
rownames(fish_data) <- fish_data$spp_name_phylo  
  
# make a matrix of effect sizes (n) for each species for each personality  
trait to add to our plot!  
# subset dataset  
pers_fish <- as.data.frame(pers_new %>%  
  filter(taxo_group == "fish"))  
  
# make this a matrix-style dataframe  
pers_fish <- data.frame(pers_fish %>%  
  group_by(spp_name_phylo, personality_trait) %>%  
  summarise(n = n()))  
  
## `summarise()` has grouped output by 'spp_name_phylo'. You can override  
using the `.groups` argument.  
  
# remove underscores from tip labels  
pers_fish$spp_name_phylo = gsub("_", " ", pers_fish$spp_name_phylo)  
  
pers_fish <- data.frame(pers_fish %>%
```

```

        spread(personality_trait, n, fill = 0))

    row.names(pers_fish) <- pers_fish$spp_name_phylo

    pers_fish <- pers_fish[,2:6]

    # matrix
    fish_matrix <- data.matrix(pers_fish)

    # FINAL TREE
    p_f1 <- ggtree(pruned.fishtree, size = 0.3, layout = 'circular',
branch.length = 'none') %<+% fish_data +
    xlim(-30, NA) +
    geom_tippoint(aes(color=SSD_index)) +
    scale_color_gradient2(midpoint = 0, low = "red3", mid = "seashell2", high =
"deepskyblue2") +
    geom_tiplab2(size = 2.5, offset = 6, colour = "black", fontface = "italic")
+
    theme(legend.position = 'right')

    # add the heatmap data to our plot
    fish_plot2 <- gheatmap(p_f1, fish_matrix, offset = 170, width = 5.5,
        low = "white", high = "mediumseagreen", color=NULL,
        colnames_position="bottom",
        colnames_angle=60,
        hjust=0, font.size=3)

fish_plot2

```



```
# ggsave("./figs/finished figs/fishphylo.tiff", fish_plot2, width=8, height = 6, units = "in")
```

inverts

```
# subset dataset to include only inverts
invert_data <- as.data.frame(figs_data %>%
  filter(taxo_group == "invertebrate"))

# setting up the basic tree structure

# Load tree, set node colours
inverttree <- read.tree("./trees/invert_species.nwk")

# prune tree to get rid of species we no longer have data for
pruned.inverttree <- drop.tip(inverttree, setdiff(inverttree$tip.label,
invert_data$spp_name_phylo))

# remove underscores from tip labels
pruned.inverttree$tip.label = gsub("_", " ", pruned.inverttree$tip.label)

# remove underscores from dataset and fix row names
invert_data$spp_name_phylo = gsub("_", " ", invert_data$spp_name_phylo)

row.names(invert_data) <- invert_data$spp_name_phylo
```

```

# set rownames for labelling tips
rownames(invert_data) <- pruned.inverttree$tip.label

# dev.new(width=8,height=6,noRStudioGD = TRUE)

# tree structure (cladogram, circular)
p5 <- ggtree(pruned.inverttree, branch.length='none', layout='circular')
%<+% invert_data +
  geom_tippoint(aes(color=SSD_index)) +
  scale_color_gradient2(midpoint = 0, low = "red3", mid = "seashell2", high =
"deepskyblue2") +
  geom_tiplab2(align=T, linetype=NA, size=2.2, offset=2, fontface = "italic")
+
  theme(legend.position = "right")

# make a matrix of effect sizes (n) for each species for each personality
trait to add to our plot!
  # subset dataset
pers_invert <- as.data.frame(pers_new %>%
  filter(taxo_group == "invertebrate"))

# make this a matrix-style dataframe
pers_invert <- data.frame(pers_invert %>%
  group_by(spp_name_phylo, personality_trait) %>%
  summarise(n = n()))

## `summarise()` has grouped output by 'spp_name_phylo'. You can override
using the `.groups` argument.

# remove underscores from dataset and fix row names
pers_invert$spp_name_phylo = gsub("_", " ", pers_invert$spp_name_phylo)

pers_invert <- data.frame(pers_invert %>%
  spread(personality_trait, n, fill = 0))

rownames(pers_invert) <- pers_invert$spp_name_phylo

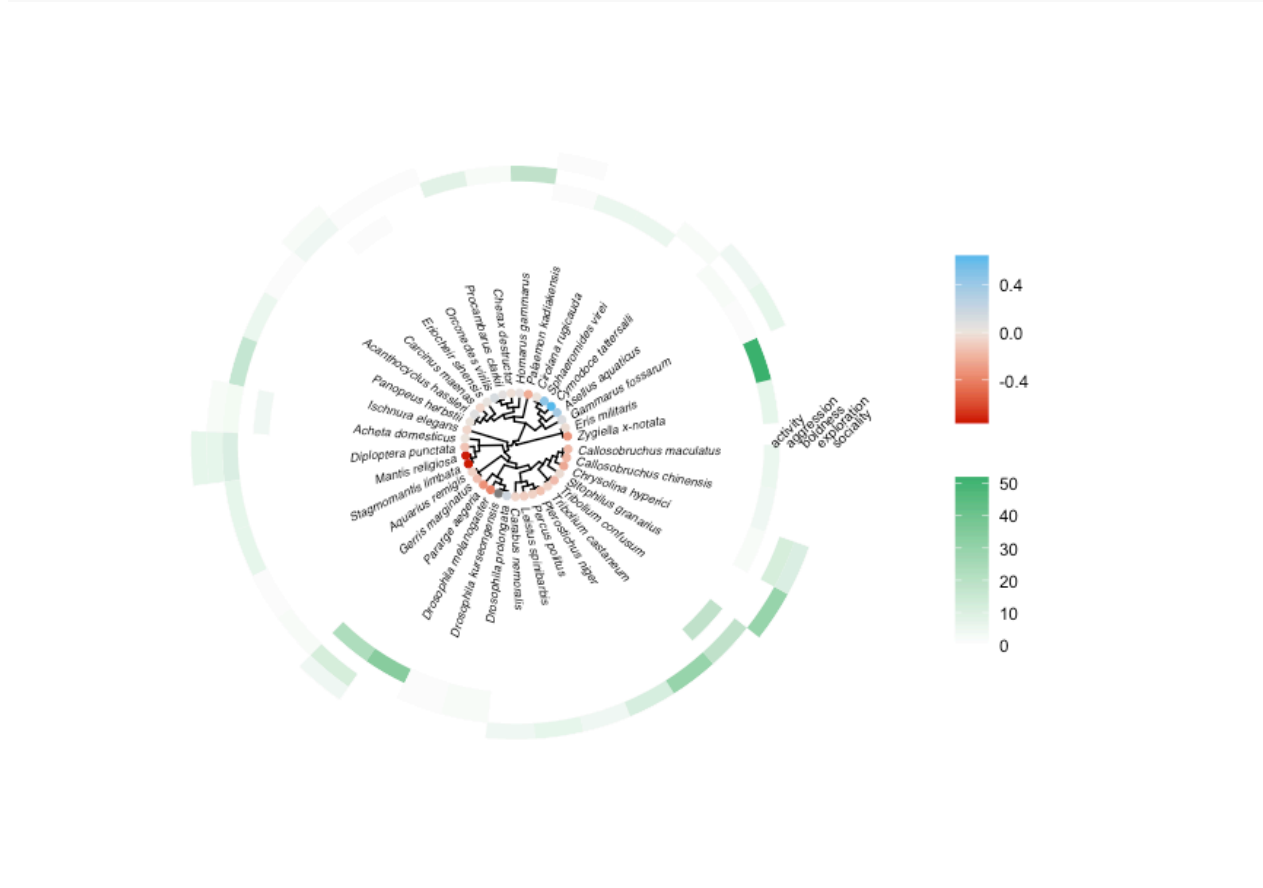
pers_invert <- pers_invert[,2:6]

# matrix
invert_matrix <- data.matrix(pers_invert)

# add the heatmap data to our plot
invertplot <- gheatmap(p5, invert_matrix, offset = 40, width = 1.5,
  low = "white", high = "mediumseagreen", color=NULL,
  colnames_position="bottom",
  colnames_angle=45, colnames_offset_y = 0,
  hjust=0, font.size=2.5)

```


invertplot



```
# save plot
# ggsave("./figs/finished figs/invertphylo.tiff", invertplot, width=8, height
= 6, units = "in")
```

These plots were edited together outside of R with the addition of creative commons animal silhouettes from PhyloPic to create Figures 2-6. Figure 1, the PRISMA diagram, was created using sankeymatic.com