R-markdown

Statistical analysis, Jaric et al.

Using mice from different breeders fails to improve replicability of results

Initialization

```
setwd("F:/REP FAIL/Experiment B4/Analysis")

library(lmerTest)

## Warning: package 'lmerTest' was built under R version 3.6.3

## Loading required package: lme4

## Loading required package: Matrix

##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step

library(MuMIn)

## Warning: package 'MuMIn' was built under R version 3.6.3

library(MASS)
library(ggplot2)
library(car)

## Loading required package: carData

## Warning: package 'carData' was built under R version 3.6.3

library(dendextend)

##
## ---------------------
## Welcome to dendextend version 1.16.0
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
```

```
## Suggestions and bug-reports can be submitted at:
https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:
suppressPackageStartupMessages(library(dendextend))
## ----------------------

##
## Attaching package: 'dendextend'

## The following object is masked from 'package:stats':
##
##     cutree

library(metafor)

##
## Loading the 'metafor' package (version 3.0-2). For an
## introduction to the package please type: help(metafor)

##
## Attaching package: 'metafor'

## The following object is masked from 'package:car':
##
##     vif

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 3.6.3

library(effectsize)
library(pheatmap)

## Warning: package 'pheatmap' was built under R version 3.6.3

options(contrasts = c("contr.sum","contr.poly"))

sessionInfo()

## R version 3.6.2 (2019-12-12)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=German_Switzerland.1252  LC_CTYPE=German_Switzerland.1252
## [3] LC_MONETARY=German_Switzerland.1252 LC_NUMERIC=C
## [5] LC_TIME=German_Switzerland.1252
```

```
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] pheatmap_1.0.12   effectsize_0.8.3  gridExtra_2.3     metafor_3.0-2
##  [5] dendextend_1.16.0 car_3.0-12        carData_3.0-4     ggplot2_3.3.5
##  [9] MASS_7.3-51.4     MuMIn_1.43.17     lmerTest_3.1-3    lme4_1.1-27.1
## [13] Matrix_1.2-18
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.6          mvtnorm_1.1-1       lattice_0.20-38
##  [4] zoo_1.8-9           assertthat_0.2.1    digest_0.6.27
##  [7] utf8_1.2.1          R6_2.5.1            stats4_3.6.2
## [10] evaluate_0.17       coda_0.19-4         pillar_1.6.4
## [13] rlang_1.0.6         multcomp_1.4-20     rstudioapi_0.13
## [16] minqa_1.2.4         nloptr_1.2.2.2      rmarkdown_2.11
## [19] mathjaxr_1.4-0      splines_3.6.2       stringr_1.4.0
## [22] munsell_0.5.0       compiler_3.6.2      numDeriv_2016.8-1.1
## [25] xfun_0.34           pkgconfig_2.0.3     parameters_0.20.2
## [28] htmltools_0.5.1.1   insight_0.18.8      tidyselect_1.2.0
## [31] tibble_3.1.1        codetools_0.2-16    fansi_0.4.2
## [34] viridisLite_0.4.0   crayon_1.4.2        dplyr_1.0.6
## [37] withr_2.5.0         grid_3.6.2          nlme_3.1-160
## [40] xtable_1.8-4        gtable_0.3.0        lifecycle_1.0.3
## [43] DBI_1.1.1           magrittr_2.0.1      bayestestR_0.13.0
## [46] scales_1.2.1        datawizard_0.6.5    estimability_1.3
## [49] cli_3.4.1           stringi_1.6.1       viridis_0.6.2
## [52] ellipsis_0.3.2      generics_0.1.1      vctrs_0.3.8
## [55] sandwich_3.0-2      boot_1.3-23         TH.data_1.1-1
## [58] RColorBrewer_1.1-2  tools_3.6.2         glue_1.4.2
## [61] emmeans_1.7.4-1     survival_3.1-8      abind_1.4-5
## [64] yaml_2.2.1          colorspace_2.0-1    knitr_1.40
```

Functions

```
makeci<-function(meanvalue,sdvalue,n){
  ci <- qnorm(0.975)*sdvalue/sqrt(n)
  c(meanvalue,meanvalue-ci,meanvalue+ci)
}

HetVariances<-function(y){
out1<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 1, ])
out2<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 2, ])
out3<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 3, ])
out4<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 4, ])
```

```
out5<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 5, ])
out6<-leveneTest(eval(parse(text=y)) ~ LAB_CATEGORY, data=data[data$LAB_BATCH
== 6, ])
chisq=-
2*sum(log(c(out1$'Pr(>F)'[1],out2$'Pr(>F)'[1],out3$'Pr(>F)'[1],out4$'Pr(>F)'[
1],out5$'Pr(>F)'[1],out6$'Pr(>F)'[1])))
f<-c(out1$'F value'[1],out2$'F value'[1],out3$'F value'[1],out4$'F
value'[1],out5$'F value'[1],out6$'F value'[1])
p<-
c(out1$'Pr(>F)'[1],out2$'Pr(>F)'[1],out3$'Pr(>F)'[1],out4$'Pr(>F)'[1],out5$'P
r(>F)'[1],out6$'Pr(>F)'[1])
cp<-pchisq(chisq, df=12, lower.tail=FALSE)
outcome<-list(f,p,cp)
names(outcome)<-c("f-val","p-val","combined p-val")
outcome
}

sdo<-function(x){sd(na.omit(x))}

sdorat <- function(var) {
  c(sdo(var[data$LAB_BATCH == 1 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 1 & data$LAB_CATEGORY == "Hom"]),
    sdo(var[data$LAB_BATCH == 2 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 2 & data$LAB_CATEGORY == "Hom"]),
    sdo(var[data$LAB_BATCH == 3 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 3 & data$LAB_CATEGORY == "Hom"]),
    sdo(var[data$LAB_BATCH == 4 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 4 & data$LAB_CATEGORY == "Hom"]),
    sdo(var[data$LAB_BATCH == 5 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 5 & data$LAB_CATEGORY == "Hom"]),
    sdo(var[data$LAB_BATCH == 6 & data$LAB_CATEGORY ==
"Het"])/sdo(var[data$LAB_BATCH == 6 & data$LAB_CATEGORY == "Hom"]))
}

makeforestplots<-function(column){
  ds<-rbind(

c(mean(na.omit(s1[,column])),sd(na.omit(s1[,column])),length(na.omit(s1[,colu
mn])),var(na.omit(s1[,column]))),

c(mean(na.omit(s2[,column])),sd(na.omit(s2[,column])),length(na.omit(s2[,colu
mn])),var(na.omit(s2[,column]))),

c(mean(na.omit(s3[,column])),sd(na.omit(s3[,column])),length(na.omit(s3[,colu
mn])),var(na.omit(s3[,column]))),

c(mean(na.omit(s4[,column])),sd(na.omit(s4[,column])),length(na.omit(s4[,colu
mn])),var(na.omit(s4[,column]))),
```

```r
c(mean(na.omit(s5[,column])),sd(na.omit(s5[,column])),length(na.omit(s5[,colu
mn])),var(na.omit(s5[,column]))),

c(mean(na.omit(s6[,column])),sd(na.omit(s6[,column])),length(na.omit(s6[,colu
mn])),var(na.omit(s6[,column])))
  )

  dh<-rbind(

c(mean(na.omit(h1[,column])),sd(na.omit(h1[,column])),length(na.omit(h1[,colu
mn])),var(na.omit(h1[,column]))),

c(mean(na.omit(h2[,column])),sd(na.omit(h2[,column])),length(na.omit(h2[,colu
mn])),var(na.omit(h2[,column]))),

c(mean(na.omit(h3[,column])),sd(na.omit(h3[,column])),length(na.omit(h3[,colu
mn])),var(na.omit(h3[,column]))),

c(mean(na.omit(h4[,column])),sd(na.omit(h4[,column])),length(na.omit(h4[,colu
mn])),var(na.omit(h4[,column]))),

c(mean(na.omit(h5[,column])),sd(na.omit(h5[,column])),length(na.omit(h5[,colu
mn])),var(na.omit(h5[,column]))),

c(mean(na.omit(h6[,column])),sd(na.omit(h6[,column])),length(na.omit(h6[,colu
mn])),var(na.omit(h6[,column])))
  )

  meta<-rma.mv(c(ds[,1],dh[,1]),c(ds[,4],dh[,4]))
  metamean<-meta$beta[1]
  metalb<-meta$ci.lb
  metaub<-meta$ci.ub

  dfh<-data.frame(rbind(
    makeci(dh[1,1],dh[1,2],dh[1,3]),
    makeci(dh[2,1],dh[2,2],dh[2,3]),
    makeci(dh[3,1],dh[3,2],dh[3,3]),
    makeci(dh[4,1],dh[4,2],dh[4,3]),
    makeci(dh[5,1],dh[5,2],dh[5,3]),
    makeci(dh[6,1],dh[6,2],dh[6,3])
  ))
  names(dfh)<-c("effect","lower","upper")
  dfh$row_num <- seq.int(nrow(dfh))

  dfs<-data.frame(rbind(
    makeci(ds[1,1],ds[1,2],ds[1,3]),
    makeci(ds[2,1],ds[2,2],ds[2,3]),
    makeci(ds[3,1],ds[3,2],ds[3,3]),
    makeci(ds[4,1],ds[4,2],ds[4,3]),
```

```
    makeci(ds[5,1],ds[5,2],ds[5,3]),
    makeci(ds[6,1],ds[6,2],ds[6,3])
  ))
  names(dfs)<-c("effect","lower","upper")
  dfs$row_num <- seq.int(nrow(dfs))

  rangemin<-min(c(dfs[,2],dfh[,2]))-mean(dfs[,1])/50
  rangemax<-max(c(dfs[,3],dfh[,3]))+mean(dfs[,1])/50

  ycoordred<-c(0.95,1.95,2.95,3.95,4.95,5.95)
  ycoordblue<-c(1.05,2.05,3.05,4.05,5.05,6.05)

  ploth<-ggplot(data=dfh, aes(y=ycoordred, x=effect, xmin=lower, xmax=upper))
+
    geom_point(color='red',size=2.4) +
    geom_errorbarh(height=.1,color='red',size=1.4) +
    scale_y_continuous(name = "", breaks=1:nrow(dfh))+
    xlim(rangemin, rangemax)+
    annotate("rect", xmin = metalb, xmax = metaub, ymin=0, ymax = Inf,
             alpha = .1,fill = "blue")+
    geom_vline(xintercept = metamean, colour="blue")+
    theme(aspect.ratio=1)+
    geom_point(data=dfs,aes(y=ycoordblue, x=effect),color='blue',size=2.4) +
    geom_errorbarh(data=dfs,aes(y=ycoordblue,
x=effect),height=.1,size=1.4,color='blue') +
    theme(text = element_text(size = 24)) +
    xlab("") +
    ylab("")

  grid.arrange(ploth, ncol=1)

  distHom<-(abs(ds[,1]-metamean)/meta$se)
  distHet<-(abs(dh[,1]-metamean)/meta$se)

  insideHom<-c()
  for(i in 1:6){
    insideHom<-c(insideHom,ds[i,1]>metalb && ds[i,1]<metaub)
  }
  suminsideHom<-sum(insideHom)

  insideHet<-c()
  for(i in 1:6){
    insideHet<-c(insideHet,dh[i,1]>metalb && dh[i,1]<metaub)
  }
  suminsideHet<-sum(insideHet)

  print(mean(distHom))
  print(mean(distHet))
```

```
  outp<-list(distHom,distHet,suminsideHom,suminsideHet)
  names(outp)<-c("Distances Hom group","Distances Het group","Inside CI
Hom","Inside CI Het")
  outp
}

makedotplot<-function(var,ylabel){
    ggplot(aes(y = var, x = LAB_CATEGORY, colour = BREEDER_ID), data = ddp) +
    facet_wrap(~`LAB_BATCH`, ncol = 6) +
    labs(x="Laboratory", y=ylabel,colour="Breeder")+
    geom_jitter(aes(colour = BREEDER_ID),width=0.25,height=0) +

scale_color_manual(values=c("#192e5b","#1d65a6","#72a2c0","#00743f","#f2a104"
,"#f25117"))
}
```

Data Import

```
data <- read.delim("curateddata.dat", header = TRUE,
na.strings=c("","NA","Na",NA),sep="\t")
data$LAB_BATCH<-as.factor(data$LAB_BATCH)
```

For Kreatinin we identified two extreme outlier. These values are outside the range of physiological plausible values and are therefore excluded (set to NA).

```
 data$KREATININ[data$KREATININ>100]<-NA

observedvalues<-c(
length(na.omit(data$BODYWEIGHT_SAC)),
length(na.omit(data$RELATIVE_ADRENALWEIGHT)),
length(na.omit(data$EPM_TTD)),
length(na.omit(data$EPM_OAE)),
length(na.omit(data$EPM_OAT)),
length(na.omit(data$TOTALPROTEIN)),
length(na.omit(data$ALBUMIN)),
length(na.omit(data$GLOBULINE)),
length(na.omit(data$KREATININ)),
length(na.omit(data$ALAT_GPT)),
length(na.omit(data$ASAT_GOT)),
length(na.omit(data$BILIRUBIN)),
length(na.omit(data$GLUCOSE)),
length(na.omit(data$TRIGLYZERIDE))
)

missingvaluetable<-data.frame(c(
                  "BODYWEIGHT_SAC","RELATIVE_ADRENALWEIGHT","EPM_TTD",
                  "EPM_OAE","EPM_OAT","TOTALPROTEIN",
                  "ALBUMIN","GLOBULINE","KREATININ","ALAT_GPT",

"ASAT_GOT","BILIRUBIN","GLUCOSE","TRIGLYZERIDE"),observedvalues)
```

```r
names(missingvaluetable)<-c("Phenotype measure","Number of recorded values")
missingvaluetable
```

```
##         Phenotype measure Number of recorded values
## 1          BODYWEIGHT_SAC                       308
## 2  RELATIVE_ADRENALWEIGHT                       308
## 3                 EPM_TTD                       299
## 4                 EPM_OAE                       299
## 5                 EPM_OAT                       299
## 6            TOTALPROTEIN                       308
## 7                 ALBUMIN                       308
## 8               GLOBULINE                       308
## 9               KREATININ                       306
## 10               ALAT_GPT                       308
## 11               ASAT_GOT                       308
## 12              BILIRUBIN                       308
## 13                GLUCOSE                       308
## 14           TRIGLYZERIDE                       308
```

```r
length(na.omit(data$BODYWEIGHT_SAC)) # number of animals for which we have data
```

```
## [1] 308
```

```r
sum(observedvalues) # total number of recorded data points
```

```
## [1] 4283
```

```r
ddp<-data
ddp$LAB_CATEGORY<-as.character(ddp$LAB_CATEGORY)
ddp$LAB_CATEGORY[ddp$LAB_CATEGORY=="Het"]<-"HET"
ddp$LAB_CATEGORY[ddp$LAB_CATEGORY=="Hom"]<-"STA"
ddp$LAB_CATEGORY<-as.factor(ddp$LAB_CATEGORY)

ddp$BREEDER_ID<-as.character(ddp$BREEDER_ID)
ddp$BREEDER_ID[ddp$BREEDER_ID=="CrlFRA"]<-"BS 1"
ddp$BREEDER_ID[ddp$BREEDER_ID=="CrlGER"]<-"BS 2"
ddp$BREEDER_ID[ddp$BREEDER_ID=="CrlUK"]<-"BS 3"
ddp$BREEDER_ID[ddp$BREEDER_ID=="Envigo1"]<-"BS 4"
ddp$BREEDER_ID[ddp$BREEDER_ID=="Envigo2"]<-"BS 5"
ddp$BREEDER_ID[ddp$BREEDER_ID=="Janvier"]<-"BS 6"
ddp$BREEDER_ID<-as.factor(ddp$BREEDER_ID)

ddp$LAB_BATCH<-as.character(ddp$LAB_BATCH)
ddp$LAB_BATCH[ddp$LAB_BATCH=="1"]<-"LAB 1"
ddp$LAB_BATCH[ddp$LAB_BATCH=="2"]<-"LAB 2"
ddp$LAB_BATCH[ddp$LAB_BATCH=="3"]<-"LAB 3"
ddp$LAB_BATCH[ddp$LAB_BATCH=="4"]<-"LAB 4"
ddp$LAB_BATCH[ddp$LAB_BATCH=="5"]<-"LAB 5"
ddp$LAB_BATCH[ddp$LAB_BATCH=="6"]<-"LAB 6"
ddp$LAB_BATCH<-as.factor(ddp$LAB_BATCH)
```

```r
pdf("DotPlot_1.pdf", width = 5.5, height = 4)
makedotplot(ddp$BODYWEIGHT_SAC,"Bodyweight (g)")
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_2.pdf", width = 5.5, height = 4)
makedotplot(data$RELATIVE_ADRENALWEIGHT,"Adrenal weight")
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_3.pdf", width = 5.5, height = 4)
makedotplot(data$EPM_TTD,"Distance travelled")
```

```
## Warning: Removed 24 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_4.pdf", width = 5.5, height = 4)
makedotplot(data$EPM_OAE,"Open arm entries")
```

```
## Warning: Removed 24 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_5.pdf", width = 5.5, height = 4)
makedotplot(data$EPM_OAT,"Time in open arms")
```

```
## Warning: Removed 24 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_6.pdf", width = 5.5, height = 4)
makedotplot(data$TOTALPROTEIN,"Total protein")
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_7.pdf", width = 5.5, height = 4)
makedotplot(data$ALBUMIN,"Albumin")
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_8.pdf", width = 5.5, height = 4)
makedotplot(data$GLOBULINE,"Glubuline")
```

```
## Warning: Removed 15 rows containing missing values (geom_point).
```

```r
Sys.sleep(1)
pdf("DotPlot_9.pdf", width = 5.5, height = 4)
makedotplot(data$KREATININ,"Kreatinine")
```

```
## Warning: Removed 17 rows containing missing values (geom_point).
```

```
Sys.sleep(1)
pdf("DotPlot_10.pdf", width = 5.5, height = 4)
makedotplot(data$ALAT_GPT,"ALAT GPT")

## Warning: Removed 15 rows containing missing values (geom_point).

Sys.sleep(1)
pdf("DotPlot_11.pdf", width = 5.5, height = 4)
makedotplot(data$ASAT_GOT,"ASAT GOT")

## Warning: Removed 15 rows containing missing values (geom_point).

Sys.sleep(1)
pdf("DotPlot_12.pdf", width = 5.5, height = 4)
makedotplot(data$BILIRUBIN,"Bilirubin")

## Warning: Removed 15 rows containing missing values (geom_point).

Sys.sleep(1)
pdf("DotPlot_13.pdf", width = 5.5, height = 4)
makedotplot(data$GLUCOSE,"Glucose")

## Warning: Removed 15 rows containing missing values (geom_point).

Sys.sleep(1)
pdf("DotPlot_14.pdf", width = 5.5, height = 4)
makedotplot(data$TRIGLYZERIDE,"Triglyceride")

## Warning: Removed 15 rows containing missing values (geom_point).

Sys.sleep(1)
```

Question 1: Taken all labs and all breeders together we have a 6-by-6 crossed design that allows us disentangle variance components due to breeder (genetic and early environmental influence) and lab (environmental influence during development). We will do this using GLMMs for singler variables and MANOVA for the combined set of response variables.

GLMMs

```
mod<-lmer(BODYWEIGHT_SAC ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq1<-r.squaredGLMM(mod)

## Warning: 'r.squaredGLMM' now calculates a revised statistic. See the help
page.

rsq1

##            R2m       R2c
## [1,] 0.3731481 0.4624065

anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                       Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID            72.992  14.598     5 124.78  5.0138 0.0003211 ***
## LAB_BATCH            116.935  23.387     5 124.76  8.0322 1.357e-06 ***
## BREEDER_ID:LAB_BATCH 307.439  12.297    25 123.88  4.2236 3.873e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(RELATIVE_ADRENALWEIGHT ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data =
data)
rsq2<-r.squaredGLMM(mod)
rsq2
```

```
##            R2m       R2c
## [1,] 0.3556029 0.5135121
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID           25.103  5.0205     5 121.85  4.0162  0.002082 **
## LAB_BATCH            74.319 14.8638     5 121.85 11.8905 2.226e-09 ***
## BREEDER_ID:LAB_BATCH 35.499  1.4200    25 121.14  1.1359  0.314764
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(EPM_TTD ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
```

```
## boundary (singular) fit: see ?isSingular
```

```
rsq3<-r.squaredGLMM(mod)
rsq3
```

```
##            R2m       R2c
## [1,] 0.2083672 0.2083672
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                       Sum Sq Mean Sq NumDF DenDF F value   Pr(>F)
## BREEDER_ID           1641840  328368     5   263  3.1276 0.009244 **
## LAB_BATCH            2038028  407606     5   263  3.8823 0.002064 **
## BREEDER_ID:LAB_BATCH 3870292  154812    25   263  1.4745 0.071940 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(EPM_OAE ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
```

```
## boundary (singular) fit: see ?isSingular
```

```
rsq4<-r.squaredGLMM(mod)
rsq4
```

```
##            R2m       R2c
## [1,] 0.2316827 0.2316827
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## BREEDER_ID           1179.0  235.81     5   263  4.1859   0.00112 **
## LAB_BATCH            1738.2  347.64     5   263  6.1710 1.989e-05 ***
## BREEDER_ID:LAB_BATCH 1509.1   60.36    25   263  1.0716   0.37575
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(EPM_OAT ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq5<-r.squaredGLMM(mod)
rsq5
```

```
##            R2m       R2c
## [1,] 0.2767675 0.3398237
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID            16618  3323.6     5 128.90  2.1044   0.06895 .
## LAB_BATCH             79284 15856.8     5 127.90 10.0399 4.005e-08 ***
## BREEDER_ID:LAB_BATCH  56128  2245.1    25 124.95  1.4215   0.10691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(TOTALPROTEIN ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq6<-r.squaredGLMM(mod)
rsq6
```

```
##            R2m       R2c
## [1,] 0.3612606 0.4185395
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID            161.90   32.38     5 119.83  1.4962    0.1961
## LAB_BATCH            2054.73  410.95     5 119.79 18.9896 7.159e-14 ***
## BREEDER_ID:LAB_BATCH  504.49   20.18    25 118.82  0.9325    0.5612
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(ALBUMIN ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq7<-r.squaredGLMM(mod)
rsq7
```

```
##            R2m      R2c
## [1,] 0.1980164 0.263322
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID            71.72  14.343     5 122.38  1.3043    0.2665
## LAB_BATCH            327.00  65.400     5 122.34  5.9472 5.868e-05 ***
## BREEDER_ID:LAB_BATCH 242.67   9.707    25 121.34  0.8827    0.6278
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(GLOBULINE ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq8<-r.squaredGLMM(mod)
rsq8
```

```
##            R2m       R2c
## [1,] 0.4329526 0.4911407
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
## BREEDER_ID            45.59   9.118     5 121.71  1.5263 0.1865
## LAB_BATCH            812.54 162.509     5 121.68 27.2031 <2e-16 ***
## BREEDER_ID:LAB_BATCH 165.81   6.632    25 120.72  1.1102 0.3420
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(KREATININ ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq9<-r.squaredGLMM(mod)
rsq9
```

```
##            R2m       R2c
## [1,] 0.1017332 0.3471128
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
## BREEDER_ID           137.71  27.541     5 121.18  1.1123 0.3574
## LAB_BATCH            232.27  46.453     5 121.20  1.8762 0.1034
## BREEDER_ID:LAB_BATCH 253.51  10.141    25 120.61  0.4096 0.9941
```

```
mod<-lmer(ALAT_GPT ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
```

```
## boundary (singular) fit: see ?isSingular
```

```
rsq10<-r.squaredGLMM(mod)
rsq10
```

```
##            R2m       R2c
## [1,] 0.3000174 0.3000174
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## BREEDER_ID            33755    6751     5   272  0.6291    0.6777
## LAB_BATCH            602198  120440     5   272 11.2236 7.523e-10 ***
## BREEDER_ID:LAB_BATCH 214742    8590    25   272  0.8005    0.7408
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(ASAT_GOT ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq11<-r.squaredGLMM(mod)
rsq11
```

```
##            R2m       R2c
## [1,] 0.1688388 0.1951272
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
## BREEDER_ID            24878    4976     5 125.09  0.7356 0.598102
## LAB_BATCH            161879   32376     5 125.02  4.7867 0.000489 ***
## BREEDER_ID:LAB_BATCH 125977    5039    25 123.93  0.7450 0.801110
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(BILIRUBIN ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq12<-r.squaredGLMM(mod)
rsq12
```

```
##            R2m       R2c
## [1,] 0.2074393 0.2720302
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID           0.3666 0.07333     5 128.22  0.4257 0.8300542
## LAB_BATCH            4.7313 0.94625     5 128.18  5.4939 0.0001289 ***
## BREEDER_ID:LAB_BATCH 6.5761 0.26304    25 127.18  1.5272 0.0672845 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(GLUCOSE ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq13<-r.squaredGLMM(mod)
rsq13
```

```
##            R2m       R2c
## [1,] 0.2876334 0.3612279
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## BREEDER_ID             6.73   1.346     5 125.41  0.3262    0.8964
## LAB_BATCH            297.60  59.521     5 125.37 14.4256 3.753e-11 ***
## BREEDER_ID:LAB_BATCH 130.97   5.239    25 124.41  1.2697    0.1960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod<-lmer(TRIGLYZERIDE ~ BREEDER_ID * LAB_BATCH+(1|CAGE_ID),data = data)
rsq14<-r.squaredGLMM(mod)
rsq14
```

```
##           R2m       R2c
## [1,] 0.187464 0.2110081
```

```
anova(mod)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                      Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
## BREEDER_ID           1.6825 0.33649     5 127.17  3.4569 0.005803 **
## LAB_BATCH            1.2427 0.24855     5 127.10  2.5534 0.030808 *
## BREEDER_ID:LAB_BATCH 3.6481 0.14592    25 126.00  1.4991 0.076430 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
propv_explained<-
c(rsq1[1],rsq2[1],rsq3[1],rsq4[1],rsq5[1],rsq6[1],rsq7[1],rsq8[1],rsq9[1],rsq
10[1],rsq11[1],rsq12[1],rsq13[1],rsq14[1])
mean(propv_explained)
```

```
## [1] 0.2636374
```

```
min(propv_explained)
```

```
## [1] 0.1017332
```

```
max(propv_explained)
```

```
## [1] 0.4329526
```

MANOVA

```
mdata<-na.omit(data)

outcomes<-cbind(

mdata$BODYWEIGHT_SAC,mdata$RELATIVE_ADRENALWEIGHT,mdata$EPM_TTD,
                    mdata$EPM_OAE,mdata$EPM_OAT,mdata$TOTALPROTEIN,
```

```
mdata$ALBUMIN,mdata$GLOBULINE,mdata$KREATININ,mdata$ALAT_GPT,

mdata$ASAT_GOT,mdata$BILIRUBIN,mdata$GLUCOSE,mdata$TRIGLYZERIDE)

dim(outcomes)

## [1] 295  14

nvar<-dim(outcomes)[2]
corcoef<-cor(outcomes)
mean(abs(corcoef))

## [1] 0.2139327

100*(sum(abs(corcoef)<0.5)/(nvar*(nvar-1))) # % of cases where corrcoef was
<0.5

## [1] 95.6044

mmodel<-manova(outcomes ~ LAB_BATCH * BREEDER_ID,data=mdata)
modsummary<-summary(mmodel)
modsummary

##                       Df  Pillai approx F num Df den Df    Pr(>F)
## LAB_BATCH              5 1.57222   8.1905     70   1250 < 2.2e-16 ***
## BREEDER_ID             5 0.56213   2.2619     70   1250 3.885e-08 ***
## LAB_BATCH:BREEDER_ID  25 1.59059   1.3279    350   3626 8.966e-05 ***
## Residuals            259
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

100*(modsummary$stats[1:3,2]/nvar) # % var explained by factors (eta-squared)

##            LAB_BATCH           BREEDER_ID LAB_BATCH:BREEDER_ID
##            11.230152             4.015217            11.361362

eta_squared(mmodel)

## # Effect Size for ANOVA (Type I)
##
## Parameter            | Eta2 (partial) |       95% CI
## -------------------------------------------------
## LAB_BATCH            |           0.31 | [0.08, 1.00]
## BREEDER_ID           |           0.11 | [0.04, 1.00]
## LAB_BATCH:BREEDER_ID |           0.11 | [0.12, 1.00]
##
## - One-sided CIs: upper bound fixed at [1.00].
```

Question 2: Do animals from different breeders have overall different phenotypes? This is the question, whether based on the combined values of the outcome variables we can predict from which breeder an animal was coming.

```r
# LDA by Breeder

evarB<-cbind(mdata$BREEDER_ID,outcomes)
evarB<-as.data.frame(evarB)
names(evarB)<-c("BREEDER_ID",
                "BODYWEIGHT_SAC","RELATIVE_ADRENALWEIGHT","EPM_TTD",
                "EPM_OAE","EPM_OAT","TOTALPROTEIN",
                "ALBUMIN","GLOBULINE","KREATININ","ALAT_GPT",
                "ASAT_GOT","BILIRUBIN","GLUCOSE","TRIGLYZERIDE")
discrimB<-lda(BREEDER_ID ~
             BODYWEIGHT_SAC + RELATIVE_ADRENALWEIGHT + EPM_TTD +
            EPM_OAE + EPM_OAT + TOTALPROTEIN +
             ALBUMIN + GLOBULINE + KREATININ + ALAT_GPT +
             ASAT_GOT + BILIRUBIN + GLUCOSE + TRIGLYZERIDE, data = evarB)
discrimB

## Call:
## lda(BREEDER_ID ~ BODYWEIGHT_SAC + RELATIVE_ADRENALWEIGHT + EPM_TTD +
##     EPM_OAE + EPM_OAT + TOTALPROTEIN + ALBUMIN + GLOBULINE +
##     KREATININ + ALAT_GPT + ASAT_GOT + BILIRUBIN + GLUCOSE + TRIGLYZERIDE,
##     data = evarB)
##
## Prior probabilities of groups:
##         1         2         3         4         5         6
## 0.1728814 0.1661017 0.1762712 0.1694915 0.1389831 0.1762712
##
## Group means:
##   BODYWEIGHT_SAC RELATIVE_ADRENALWEIGHT  EPM_TTD  EPM_OAE   EPM_OAT
## 1       28.21706               8.641686 1478.003 20.03922  90.68212
## 2       28.22694               8.337700 1269.922 14.55102  73.26988
## 3       27.72365               8.424134 1384.181 19.71154 109.23919
## 4       29.00220               7.695923 1192.285 15.44000  76.14989
## 5       28.47610               7.465841 1279.190 18.02439  86.86440
## 6       27.71769               8.922970 1330.756 17.63462  88.53983
##   TOTALPROTEIN  ALBUMIN GLOBULINE KREATININ  ALAT_GPT ASAT_GOT BILIRUBIN
## 1     38.59216 26.25882  12.33333  12.25490  82.37255 159.9412 0.7333333
## 2     38.50408 26.75918  11.92857  14.22449  85.63265 147.3673 0.8061224
## 3     40.01346 27.59038  12.42308  11.42308  69.82692 141.8846 0.7307692
## 4     40.82600 28.46600  12.36000  12.34000 159.50000 175.0600 0.7500000
## 5     39.05366 26.86829  12.18537  11.04878 128.80488 190.1951 0.7365854
## 6     35.84615 25.60000  10.13974  11.92308  87.78846 173.5000 0.7115385
##    GLUCOSE TRIGLYZERIDE
## 1 12.94216     1.177255
## 2 12.92327     1.072245
## 3 12.84615     1.021538
## 4 12.88440     1.192800
## 5 12.50366     1.265366
## 6 11.87442     1.070769
##
## Coefficients of linear discriminants:
```

```
##                                  LD1           LD2           LD3
LD4
## BODYWEIGHT_SAC          0.1259255163 -0.0402689439 -2.940345e-02
0.0668722511
## RELATIVE_ADRENALWEIGHT -0.3387331595 -0.0209564838 -1.103237e-01 -
0.0811370629
## EPM_TTD                -0.0007743039  0.0003722017  9.350514e-06
0.0011207529
## EPM_OAE                -0.0230845897  0.0496134049  2.093212e-02
0.0217044585
## EPM_OAT                -0.0030186349  0.0031000265  1.064027e-02 -
0.0051488153
## TOTALPROTEIN            0.0647874689  0.4242778830  2.296915e-01 -
0.3351770627
## ALBUMIN                 0.0411996231 -0.4666269592 -2.554052e-02
0.2402392128
## GLOBULINE              -0.0110340440 -0.4599376091 -9.622066e-02
0.5736016530
## KREATININ              -0.0059783397 -0.0756928105 -3.773371e-02
0.0353667312
## ALAT_GPT                0.0053228720 -0.0005932311  3.695197e-04 -
0.0012132726
## ASAT_GOT                0.0021688167  0.0051435900 -3.299147e-05 -
0.0002919332
## BILIRUBIN              -0.1214475541  0.0966601168 -7.069985e-01
0.0268067055
## GLUCOSE                -0.2684247401 -0.1603051421 -1.238643e-01
0.0034501046
## TRIGLYZERIDE            0.7125287585  1.8560783160 -1.677014e+00
0.8686206983
##                                  LD5
## BODYWEIGHT_SAC         -0.2206093739
## RELATIVE_ADRENALWEIGHT -0.4811089315
## EPM_TTD                -0.0006990114
## EPM_OAE                -0.0153466531
## EPM_OAT                 0.0033304650
## TOTALPROTEIN           -0.5261784790
## ALBUMIN                 0.4261500128
## GLOBULINE               0.5104075218
## KREATININ              -0.0201775744
## ALAT_GPT               -0.0043987154
## ASAT_GOT                0.0011640959
## BILIRUBIN               0.5181536309
## GLUCOSE                -0.0127882245
## TRIGLYZERIDE            0.0258275275
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4    LD5
## 0.4946 0.1889 0.1711 0.1031 0.0423
```

```
predB <- predict(discrimB)$class
predtabB<-table(evarB$BREEDER_ID, predB, dnn = c('Actual Group','Predicted
Group'))
predtabB
```

```
##              Predicted Group
## Actual Group  1  2  3  4  5  6
##            1 22  7  6  4  3  9
##            2  7 16 10  5  3  8
##            3  9  6 22  4  3  8
##            4  4  7  6 24  6  3
##            5  3  1  5  8 17  7
##            6  9  7  7  7  1 21
```

```
diag(prop.table(predtabB, 1))
```

```
##         1         2         3         4         5         6
## 0.4313725 0.3265306 0.4230769 0.4800000 0.4146341 0.4038462
```

```
sum(diag(prop.table(predtabB)))
```

```
## [1] 0.4135593
```

```
match<-as.factor(evarB$BREEDER_ID)==predB


lda.data <- cbind(evarB, predict(discrimB)$x)
pdf("LDABs3.pdf", width = 4, height = 4)
ggplot(lda.data, aes(LD1, LD2)) +
  #xlim(-5,6) +
  #ylim(-5.5,4.5) +
  coord_cartesian(xlim = c(-6,6.5), ylim = c(-5.5,4.5), expand = FALSE) +
  scale_y_continuous( breaks = c(-4,-2,0,2,4) ) +
  #scale_x_continuous( limits = c(-5,6), expand = c(0,0) ) +
  #scale_y_continuous( limits = c(-5.5,4.5), expand = c(0,0) ) +
  theme(panel.grid.major = element_blank(), panel.grid.minor =
element_blank(),panel.background = element_blank(),axis.line =
element_line(colour = "black"),legend.position = "none") +
  stat_ellipse(aes(color = as.factor(evarB$BREEDER_ID))) +
  geom_point(aes(shape = match,color = as.factor(evarB$BREEDER_ID)),size=3)+
  scale_shape_manual(values = c(21,16)) +

scale_color_manual(values=c("#192e5b","#1d65a6","#72a2c0","#00743f","#f2a104"
,"#f25117"))

# Creating HeatMap for correct classification
colnames(predtabB) <- paste0("B",seq(1,6))
rownames(predtabB) <- paste0("B",seq(1,6))

pdf("PredB.pdf", width = 4, height = 4)
pheatmap(predtabB,
```

```
         cluster_rows = FALSE,
         cluster_cols = FALSE,
         cellheight=20,
         cellwidth = 20,
         color = colorRampPalette(c("white", "firebrick3"))(50),
         angle_col = 0,
         fontsize = 12)
```

Question 3: Do animals tested at different labs have overall different dramatypes? This is the question, whether based on the combined values of the outcome variables we can predict at which lab an animal was tested.

```
# LDA by Lab

evarL<-cbind(mdata$LAB_BATCH ,outcomes)
evarL<-as.data.frame(evarL)
names(evarL)<-c("LAB_BATCH",
                  "BODYWEIGHT_SAC","RELATIVE_ADRENALWEIGHT","EPM_TTD",
                  "EPM_OAE","EPM_OAT","TOTALPROTEIN",
                  "ALBUMIN","GLOBULINE","KREATININ","ALAT_GPT",
                  "ASAT_GOT","BILIRUBIN","GLUCOSE","TRIGLYZERIDE")
discrimL<-lda(LAB_BATCH ~
              BODYWEIGHT_SAC + RELATIVE_ADRENALWEIGHT + EPM_TTD +
             EPM_OAE + EPM_OAT + TOTALPROTEIN +
              ALBUMIN + GLOBULINE + KREATININ + ALAT_GPT +
              ASAT_GOT + BILIRUBIN + GLUCOSE + TRIGLYZERIDE, data = evarL)
discrimL

## Call:
## lda(LAB_BATCH ~ BODYWEIGHT_SAC + RELATIVE_ADRENALWEIGHT + EPM_TTD +
##     EPM_OAE + EPM_OAT + TOTALPROTEIN + ALBUMIN + GLOBULINE +
##     KREATININ + ALAT_GPT + ASAT_GOT + BILIRUBIN + GLUCOSE + TRIGLYZERIDE,
##     data = evarL)
##
## Prior probabilities of groups:
##         1         2         3         4         5         6
## 0.1796610 0.1762712 0.1728814 0.1322034 0.1661017 0.1728814
##
## Group means:
##   BODYWEIGHT_SAC RELATIVE_ADRENALWEIGHT  EPM_TTD  EPM_OAE   EPM_OAT
## 1       28.94906               9.811185 1337.996 14.60377  78.79698
## 2       27.86500               7.725146 1350.865 20.98077  86.81154
## 3       28.58627               8.020150 1232.016 13.52941  75.15843
## 4       26.76795               7.486886 1227.570 16.43590  74.69718
## 5       28.40816               8.463282 1317.001 19.69388  78.31727
## 6       28.34588               7.937272 1459.653 20.13725 129.29804
##   TOTALPROTEIN  ALBUMIN GLOBULINE KREATININ  ALAT_GPT ASAT_GOT BILIRUBIN
## 1     32.50566 24.75283   7.75283  13.50943  79.32075 201.0943 0.5584906
## 2     40.15000 27.54808  12.60192  13.65385 214.92308 188.9038 0.9826923
## 3     42.28431 28.36863  14.09216  11.96078  58.39216 121.4510 0.8333333
```

```
## 4     38.17436 25.95897  12.07350  10.97436 143.30769 180.8718 0.7512821
## 5     38.17551 26.40000  11.77551  12.24490  57.95918 166.5306 0.6632653
## 6     41.48235 28.31373  13.16863  10.62745  60.03922 126.2745 0.6784314
##    GLUCOSE TRIGLYZERIDE
## 1 10.61906    0.9786792
## 2 14.07154    1.2055769
## 3 13.82431    1.1370588
## 4 12.21872    1.1410256
## 5 12.14612    1.1616327
## 6 13.02686    1.1554902
##
## Coefficients of linear discriminants:
##                                   LD1           LD2           LD3
LD4
## BODYWEIGHT_SAC         -2.442164e-01 -0.1142513443  0.1246456179
2.145939e-01
## RELATIVE_ADRENALWEIGHT -5.174259e-01 -0.1304382899  0.0452071973
2.671795e-01
## EPM_TTD                -6.757982e-06 -0.0002424899  0.0009870295 -
6.896173e-05
## EPM_OAE                -3.258713e-03  0.0233004441  0.0327940187 -
2.110231e-02
## EPM_OAT                -8.859340e-04 -0.0093977401  0.0154833644 -
4.595346e-03
## TOTALPROTEIN            8.012487e-02  0.1291390729 -0.0259576889 -
4.842944e-01
## ALBUMIN                -8.947931e-02 -0.2271766776  0.1353096003
4.737358e-01
## GLOBULINE               2.609025e-01 -0.2392257435 -0.0912888327
4.315812e-01
## KREATININ              -3.070394e-02  0.0105288456  0.0084262916
4.142104e-02
## ALAT_GPT                5.419403e-04  0.0062295598  0.0031938880
6.193721e-04
## ASAT_GOT               -2.806901e-03  0.0034787893  0.0005718021 -
2.452257e-03
## BILIRUBIN               3.574640e-01  0.8163093052  0.1307315638
7.818137e-01
## GLUCOSE                -1.718985e-02  0.0707369183  0.0798874333
3.644395e-01
## TRIGLYZERIDE           -5.320215e-01  0.5483320195 -0.1277224901 -
6.623722e-01
##                                   LD5
## BODYWEIGHT_SAC         -0.0768439368
## RELATIVE_ADRENALWEIGHT -0.0905375704
## EPM_TTD                 0.0005810986
## EPM_OAE                -0.1240178930
## EPM_OAT                 0.0103943002
## TOTALPROTEIN            0.1302191512
## ALBUMIN                -0.1115977914
```

```
## GLOBULINE              -0.1771657375
## KREATININ              -0.0367953563
## ALAT_GPT                0.0039570970
## ASAT_GOT               -0.0019726026
## BILIRUBIN               0.1356125003
## GLUCOSE                -0.0021548361
## TRIGLYZERIDE           -1.1836588358
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4    LD5
## 0.5621 0.2278 0.0979 0.0776 0.0345
```

```
predL <- predict(discrimL)$class
predtabL<-table(evarL$LAB_BATCH, predL, dnn = c('Actual Group','Predicted
Group'))
predtabL
```

```
##             Predicted Group
## Actual Group  1  2  3  4  5  6
##            1 43  1  1  0  7  1
##            2  3 31  3  6  6  3
##            3  0  2 35  3  3  8
##            4  0  6  4 21  7  1
##            5  7  1  6  3 27  5
##            6  1  2  2  3  5 38
```

```
diag(prop.table(predtabL, 1))
```

```
##         1         2         3         4         5         6
## 0.8113208 0.5961538 0.6862745 0.5384615 0.5510204 0.7450980
```

```
sum(diag(prop.table(predtabL)))
```

```
## [1] 0.6610169
```

```
match<-as.factor(evarL$LAB_BATCH)==predL
```

```
lda.data <- cbind(evarL, predict(discrimL)$x)
pdf("LDALs3.pdf", width = 4, height = 4)
ggplot(lda.data, aes(LD1, LD2)) +
  #xlim(-7.5,4.5) +
  #ylim(-4,6) +
  coord_cartesian(xlim = c(-7.5,5.5), ylim = c(-4,6), expand = FALSE) +
  scale_x_continuous( breaks = c(-6,-4,-2,0,2,4) ) +
  theme(panel.grid.major = element_blank(), panel.grid.minor =
element_blank(),panel.background = element_blank(),axis.line =
element_line(colour = "black"),legend.position = "none") +
  stat_ellipse(aes(color = as.factor(evarL$LAB_BATCH))) +
  geom_point(aes(shape = match,color = as.factor(evarL$LAB_BATCH)),size=3)+
  scale_shape_manual(values = c(21,16)) +
```

```
  scale_color_manual(values=c("#1F78B4", "#5AAE61",
"#FDAE61","#C51B7D","#6A3D9A","#5B3857"))

# Creating HeatMap for correct classification
colnames(predtabL) <- paste0("L",seq(1,6))
rownames(predtabL) <- paste0("L",seq(1,6))

pdf("PredL.pdf", width = 4, height = 4)
pheatmap(predtabL,
         cluster_rows = FALSE,
         cluster_cols = FALSE,
         cellheight=20,
         cellwidth = 20,
         color = colorRampPalette(c("white", "firebrick3"))(50),
         angle_col = 0,
         fontsize = 12)
```

Question 4: Does heterogenization increase within-group variance? This is a question of comparing variances of the heterogenized and standardized groups. The six labs are our six replicates. Comparing variances is done with an F-ratio test, however the classical F-test is rather sensible to outliers. The more robust and preferred version is Levene's test for equal variances. We have in total 15 outcome variables; for each outcome we get 6 f-ratios (one for each lab). Figure 1 shows the ratios of SD(heterogenized)/SD(standardized) –i.e. a ratio of >1 means the SD of the heterogenized group was larger than of the standardized group.

```
# LeveneTest for equal variances

varnames<-
c("BODYWEIGHT_SAC","RELATIVE_ADRENALWEIGHT","EPM_TTD","EPM_OAE","EPM_OAT","TO
TALPROTEIN","ALBUMIN","GLOBULINE","KREATININ","ALAT_GPT","ASAT_GOT","BILIRUBI
N","GLUCOSE","TRIGLYZERIDE")

levtests<-list()
for(i in 1:length(varnames)){
  levtests[[length(levtests)+1]] <-HetVariances(varnames[i])
} # This gives for each of the 15 variables the outcome of the leveneTests

singlep<-c()
for(i in 1:length(varnames)){
singlep<-c(singlep,levtests[[i]]$`p-val`)
}
singlep<0.05
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
FALSE
## [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
FALSE
## [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
```

```
spv<-rep(seq(1,14),each=6)[singlep<0.05]
spl<-rep(seq(1,6),14)[singlep<0.05]
sigdev<-c(varnames[spv],spl,singlep[(spv-1)*6+spl],(levtests[[spv]]$`f-
val`)[3])
names(sigdev)<-c("Measure","Lab","p-vale","f-value")
sigdev # measure with unequal variances
```

```
##              Measure                  Lab               p-vale
##       "TOTALPROTEIN"                  "3" "0.0443087246897567"
##              f-value
##   "4.26113941306357"
```

```
c(min(levtests[[spv]]$`f-val`),max(levtests[[spv]]$`f-val`)) # Range of f-
values
```

```
## [1] 0.006094408 4.261139413
```

```
combinedp<-c()
for(i in 1:length(varnames)){
combinedp<-c(combinedp,levtests[[i]]$`combined p-val`)
}
min(combinedp) # smallest combined p-value
```

```
## [1] 0.2307594
```

```
fvalues<-data.frame(rbind(levtests[[1]]$`f-val`,levtests[[2]]$`f-
val`,levtests[[3]]$`f-val`,levtests[[4]]$`f-val`,levtests[[5]]$`f-val`,
           levtests[[6]]$`f-val`,levtests[[7]]$`f-val`,levtests[[8]]$`f-
val`,levtests[[9]]$`f-val`,levtests[[10]]$`f-val`,
           levtests[[11]]$`f-val`,levtests[[12]]$`f-val`,levtests[[13]]$`f-
val`,levtests[[14]]$`f-val`),combinedp)
names(fvalues)<-c("Lab1","Lab2","Lab3","Lab4","Lab5","Lab6","P(combined)")
row.names(fvalues)<-varnames
fvalues
```

```
##                                Lab1         Lab2       Lab3       Lab4
## BODYWEIGHT_SAC          2.0917515063 0.0348992514 0.13142370 4.00805607
## RELATIVE_ADRENALWEIGHT 1.3325133083 0.0158524768 0.05110775 0.03181027
## EPM_TTD                 0.3839557728 0.1242128752 3.50802449 0.11275885
## EPM_OAE                 0.0014450625 0.0287709360 2.55732625 0.08886813
## EPM_OAT                 0.1518909978 2.2701735922 1.13589439 0.48840877
## TOTALPROTEIN            0.0060944077 0.1098354468 4.26113941 0.68525185
## ALBUMIN                 0.0119599359 1.1342549392 3.73376072 0.69367348
## GLOBULINE               0.0131912004 0.1305618847 0.23632168 0.07578559
```

```
## KREATININ               0.0001682574 0.0008137871 2.44225272 0.21026941
## ALAT_GPT                0.0458843854 0.6083436128 0.25009093 0.02765608
## ASAT_GOT                2.1704791270 0.2643037431 0.08544564 1.06005964
## BILIRUBIN               0.3994543142 0.0117275598 1.20836624 0.57897795
## GLUCOSE                 0.1024660840 0.4012638870 0.54313053 1.17111251
## TRIGLYZERIDE            1.3145703594 2.7291127495 0.33900446 0.03645172
##                                Lab5        Lab6 P(combined)
## BODYWEIGHT_SAC          0.371718304 0.970194441   0.2939120
## RELATIVE_ADRENALWEIGHT 1.409677459 1.558840485   0.6560496
## EPM_TTD                 0.396648623 0.485338299   0.5647414
## EPM_OAE                 1.156419737 0.385602982   0.7097580
## EPM_OAT                 0.001724017 1.543153722   0.4725422
## TOTALPROTEIN            3.807625493 0.158726655   0.2307594
## ALBUMIN                 0.972750774 0.934108846   0.2712033
## GLOBULINE               0.134296759 2.588197966   0.8390240
## KREATININ               0.189651132 0.007755494   0.9130989
## ALAT_GPT                1.945509850 0.025802750   0.8525209
## ASAT_GOT                0.066240296 0.244253339   0.6946605
## BILIRUBIN               1.665205162 0.008993301   0.7068453
## GLUCOSE                 0.074101276 0.668440920   0.7756354
## TRIGLYZERIDE            0.031425464 0.985910845   0.5072289
```

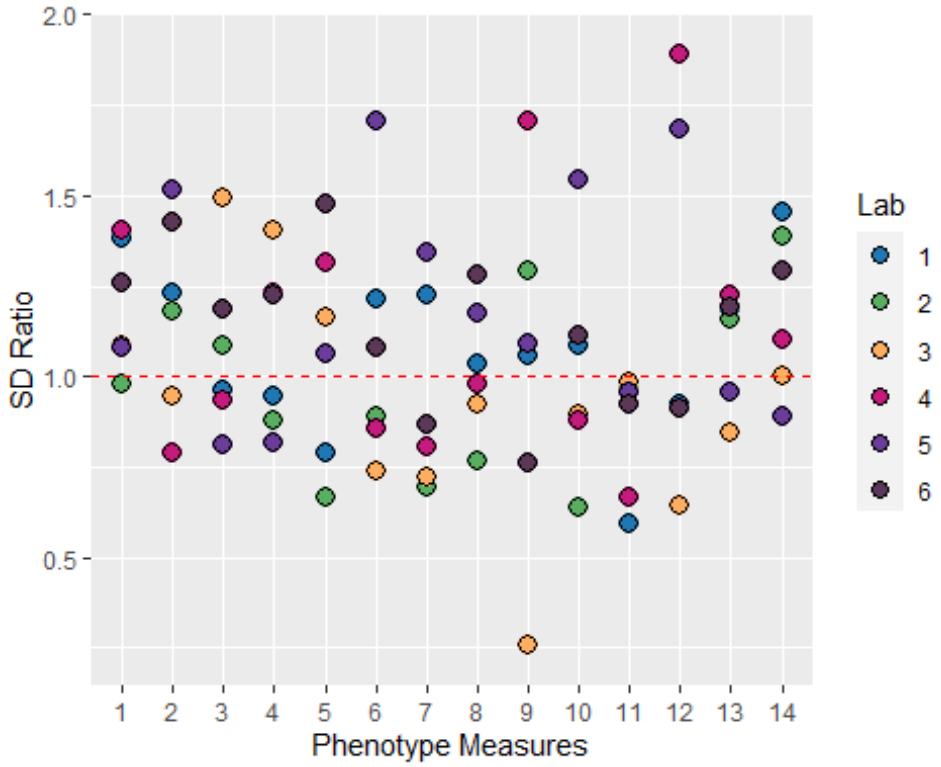
```
sdrats<-c()
for(i in 1:length(varnames)){
sdrats<-c(sdrats,sdorat(eval(parse(text=paste("data$",varnames[i])))))
}
```

```
summary(sdrats<1)
```

```
##    Mode   FALSE    TRUE
## logical      45      39
```

```
outcome<-as.factor(rep(seq(1,14),each=6))
Lab<-as.factor(rep(seq(1,6),14))
df<-data.frame(sdrats,outcome,Lab)
plot<-ggplot(df, aes(x=outcome, y=sdrats, fill = Lab))+
  xlab("Phenotype Measures")+
  ylab("SD Ratio")+
  geom_dotplot(binaxis='y', stackdir='center')+
  geom_hline(yintercept=1, linetype="dashed", color = "red")+
  scale_fill_manual(values=c("#1F78B4", "#5AAE61",
"#FDAE61","#C51B7D","#6A3D9A","#5B3857"))
plot
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value
with `binwidth`.
```




```
outcome<-as.factor(rep(seq(1,14),each=6))
Lab<-as.factor(rep(seq(1,6),14))
df<-data.frame(sdrats,outcome,Lab)
pdf("SDPlot.pdf", width = 6, height = 6)
ggplot(df, aes(x=sdrats, y=outcome, fill = Lab))+
  ylab("")+
  xlab("SD Ratio")+
  theme(aspect.ratio=2)+
  geom_dotplot(binaxis='y', stackdir='center')+
  geom_vline(xintercept=1, linetype="dashed", color = "red", size=0.8)+
  scale_fill_manual(values=c("#1F78B4", "#5AAE61",
"#FDAE61","#C51B7D","#6A3D9A","#5B3857"))+
  scale_y_discrete(labels=c(
    "BODYWEIGHT_SAC","RELATIVE_ADRENALWEIGHT","EPM_TTD",
    "EPM_OAE","EPM_OAT","TOTALPROTEIN",
    "ALBUMIN","GLOBULINE","KREATININ","ALAT_GPT",
    "ASAT_GOT","BILIRUBIN","GLUCOSE","TRIGLYZERIDE"))

## Bin width defaults to 1/30 of the range of the data. Pick better value
with `binwidth`.
```

Question 5: Does heterogenization lead to lower between-lab variation? To address this question, I ran for each outcome variable a mixed model of the form outcome~Lab+(1|cage); separate models for the Heterogenised and Homogeneous cohorts. The focus is then on the marginal R-squared. If it is large(er) that means that more of the

overall variation is due to between-lab variation. We therefore would expect, that the marginal r-squared is always larger in the homogeneous cohorts.

```
rshet<-c()
rshom<-c()
for(i in 1:length(varnames)){
dhet<-data[data$LAB_CATEGORY == 'Het', ]
modhet<-lmer(eval(parse(text=varnames[i])) ~ LAB_BATCH+(1|CAGE_ID),data =
dhet)
rshet<-rbind(rshet,r.squaredGLMM(modhet))
dhom<-data[data$LAB_CATEGORY == 'Hom', ]
modhom<-lmer(eval(parse(text=varnames[i])) ~ LAB_BATCH+(1|CAGE_ID),data =
dhom)
rshom<-rbind(rshom,r.squaredGLMM(modhom))
}
```

```
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
## boundary (singular) fit: see ?isSingular
```

```
c(min(rshet[,1]),max(rshet[,1])) # range of var explained by Lab in het group
```

```
## [1] 0.04963594 0.40409014
```

```
c(min(rshom[,1]),max(rshom[,1])) # range of var explained by Lab in hom group
```

```
## [1] 0.03759407 0.40325302
```

```
median(rshet[,1])
```

```
## [1] 0.153344
```

```
median(rshom[,1])
```

```
## [1] 0.1533434
```

```
hetplus<-sum((rshom[,1]-rshet[,1])>0) # Number of cases where explained var
in hom is larger
hetplus
```

```
## [1] 8
```

```
binom.test(hetplus, length(varnames))
```

```
##
##  Exact binomial test
##
## data:  hetplus and length(varnames)
## number of successes = 8, number of trials = 14, p-value = 0.7905
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.2886094 0.8233889
## sample estimates:
## probability of success
##              0.5714286
```

```
data.frame(varnames,rshet[,1],rshom[,1])
```

```
##                   varnames rshet...1. rshom...1.
## 1           BODYWEIGHT_SAC 0.16379408 0.10053230
## 2  RELATIVE_ADRENALWEIGHT 0.19561680 0.37885523
## 3                  EPM_TTD 0.11694561 0.08687899
## 4                  EPM_OAE 0.12693336 0.13748972
## 5                  EPM_OAT 0.19838145 0.15918170
## 6             TOTALPROTEIN 0.31335577 0.36991640
## 7                  ALBUMIN 0.14289396 0.17219691
## 8               GLOBULINE 0.40409014 0.40325302
## 9               KREATININ 0.08662138 0.03759407
## 10               ALAT_GPT 0.19514079 0.34191430
## 11               ASAT_GOT 0.10787042 0.14750520
## 12              BILIRUBIN 0.10896208 0.13573967
## 13                GLUCOSE 0.26284350 0.19852956
## 14           TRIGLYZERIDE 0.04963594 0.07865587
```

Question 6: If we treat the 6 replicates of heterogenized groups as one set of studies for meta analysis and the 6 replicates of homogenized groups as another set of studies for meta analysis, do the forest plots between these two sets differ? Do we find higher replicability (in terms of coverage) in the heterogenized group? Do the two groups differ in terms of their dance around the (meta-analytic) mean?

```
# Multi Lab

s1<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==1,]
s2<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==2,]
s3<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==3,]
s4<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==4,]
s5<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==5,]
s6<-data[data$LAB_CATEGORY=="Hom" & data$LAB_BATCH==6,]

h1<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==1,]
h2<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==2,]
h3<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==3,]
h4<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==4,]
h5<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==5,]
```

```r
h6<-data[data$LAB_CATEGORY=="Het" & data$LAB_BATCH==6,]


pdf("forestBODYWEIGHT_SAC.pdf", width = 3, height = 4)
makeforestplots(11) # BODYWEIGHT_SAC
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 0.9811448
## [1] 1.472004
```

```
## $`Distances Hom group`
## [1] 1.2561152 2.0322706 0.8102727 0.9423405 0.3761551 0.4697147
##
## $`Distances Het group`
## [1] 1.1347090 2.4086633 1.7833121 2.6615059 0.1478755 0.6959585
##
## $`Inside CI Hom`
## [1] 5
##
## $`Inside CI Het`
## [1] 4
```

```r
Sys.sleep(2)
pdf("forestRELATIVE_ADRENALWEIGHT.pdf", width = 3, height = 4)
makeforestplots(17) # RELATIVE_ADRENALWEIGHT
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 2.318343
## [1] 1.426996
```

```
## $`Distances Hom group`
## [1] 6.05363011 2.29574344 0.95561620 3.27104029 1.30537595 0.02864997
##
## $`Distances Het group`
## [1] 5.61501270 0.02917246 0.72094779 0.19605992 1.65254064 0.34824235
##
## $`Inside CI Hom`
## [1] 3
##
## $`Inside CI Het`
## [1] 5
```

```r
Sys.sleep(2)
pdf("forestEPM_TTD.pdf", width = 3, height = 4)
makeforestplots(19) # EPM_TTD
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 0.9860665
## [1] 1.394534
```

```
## $`Distances Hom group`
## [1] 0.03232682 1.31087794 1.26519333 0.16807619 1.90996756 1.22995740
##
## $`Distances Het group`
## [1] 1.0510865 2.0378719 0.1801117 1.3274634 0.9572256 2.8134471
##
## $`Inside CI Hom`
## [1] 6
##
## $`Inside CI Het`
## [1] 4
```

```
Sys.sleep(2)
pdf("forestEPM_OAE.pdf", width = 3, height = 4)
makeforestplots(21) # EPM_OAE
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 1.536678
## [1] 1.396913
```

```
## $`Distances Hom group`
## [1] 0.75303834 1.60734952 2.18288867 0.05494058 2.56058308 2.06127026
##
## $`Distances Het group`
## [1] 1.06452188 3.12344480 0.90218372 0.08671504 1.08786245 2.11674946
##
## $`Inside CI Hom`
## [1] 3
##
## $`Inside CI Het`
## [1] 4
```

```
Sys.sleep(2)
pdf("forestEPM_OAT.pdf", width = 3, height = 4)
makeforestplots(22) # EPM_OAT
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 1.196488
## [1] 1.499186
```

```
## $`Distances Hom group`
## [1] 0.04507765 0.65563397 1.40333729 1.46674633 0.24820756 3.35992544
##
## $`Distances Het group`
## [1] 1.3952865 0.3358908 0.8544352 0.8851646 1.4032441 4.1210951
##
## $`Inside CI Hom`
## [1] 5
##
```

```
## $`Inside CI Het`
## [1] 5
```

```
Sys.sleep(2)
pdf("forestTOTALPROTEIN.pdf", width = 3, height = 4)
makeforestplots(23) # TOTALPROTEIN
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 2.187621
## [1] 1.729917
```

```
## $`Distances Hom group`
## [1] 5.6299291 2.2015469 0.8878184 1.1311701 0.7047265 2.5705360
##
## $`Distances Het group`
## [1] 4.343074072 0.163800616 4.007614331 0.007342209 0.502252707
1.355416662
##
## $`Inside CI Hom`
## [1] 3
##
## $`Inside CI Het`
## [1] 4
```

```
Sys.sleep(2)
pdf("forestALBUMIN.pdf", width = 3, height = 4)
makeforestplots(24) # ALBUMIN
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 1.459265
## [1] 1.019367
```

```
## $`Distances Hom group`
## [1] 2.9311512 1.3708613 0.3335284 0.5831993 0.8592189 2.6776313
##
## $`Distances Het group`
## [1] 1.28417819 0.01045295 3.22997739 0.27704084 0.27998644 1.03456388
##
## $`Inside CI Hom`
## [1] 4
##
## $`Inside CI Het`
## [1] 5
```

```
Sys.sleep(2)
pdf("forestGLOBULINE.pdf", width = 3, height = 4)
makeforestplots(25) # GLOBULINE
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 2.323696
## [1] 2.151561

## $`Distances Hom group`
## [1] 6.8358610 2.4153954 1.8494941 1.3525935 0.1498813 1.3389526
##
## $`Distances Het group`
## [1] 6.6033918 0.2664848 3.3612981 0.1195464 1.3187963 1.2398516
##
## $`Inside CI Hom`
## [1] 4
##
## $`Inside CI Het`
## [1] 4
```

```
Sys.sleep(2)
pdf("forestKREATININ.pdf", width = 3, height = 4)
makeforestplots(26) # KREATININ
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 1.302805
## [1] 1.450015

## $`Distances Hom group`
## [1] 1.1854404 1.7739168 1.7174876 1.6901480 0.1523884 1.2974465
##
## $`Distances Het group`
## [1] 2.0140886 1.8061621 1.5437647 1.1048088 0.6874988 1.5437647
##
## $`Inside CI Hom`
## [1] 6
##
## $`Inside CI Het`
## [1] 5
```

```
Sys.sleep(2)
pdf("forestALAT_GPT.pdf", width = 3, height = 4)
makeforestplots(27) # ALAT_GPT
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 3.49246
## [1] 2.189732

## $`Distances Hom group`
## [1]  1.09962991 13.39704327  0.03302659  5.08761149  0.67045581
0.66699591
##
## $`Distances Het group`
## [1] 0.72973062 7.08387593 0.79255190 4.22237017 0.27890140 0.03096346
##
```

```
## $`Inside CI Hom`
## [1] 4
##
## $`Inside CI Het`
## [1] 4
```

```
Sys.sleep(2)
pdf("forestASAT_GOT.pdf", width = 3, height = 4)
makeforestplots(28) # ASAT_GOT
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 1.714727
## [1] 1.094719
```

```
## $`Distances Hom group`
## [1] 2.4901992 2.7422265 1.3633187 1.3570379 0.6028796 1.7327024
##
## $`Distances Het group`
## [1] 1.8336277 0.6244270 1.7091884 0.9658946 0.4750161 0.9601604
##
## $`Inside CI Hom`
## [1] 4
##
## $`Inside CI Het`
## [1] 6
```

```
Sys.sleep(2)
pdf("forestBILIRUBIN.pdf", width = 3, height = 4)
makeforestplots(29) # BILIRUBIN
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## [1] 1.234921
## [1] 1.146347
```

```
## $`Distances Hom group`
## [1] 1.01052999 2.56362949 1.51019301 0.09146041 0.80933017 1.42438004
##
## $`Distances Het group`
## [1] 1.9017995 2.9097586 0.5152808 0.7447650 0.3246387 0.4818384
##
## $`Inside CI Hom`
## [1] 5
##
## $`Inside CI Het`
## [1] 5
```

```
Sys.sleep(2)
pdf("forestGLUCOSE.pdf", width = 3, height = 4)
makeforestplots(30) # GLUCOSE
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 1.719878
## [1] 2.03423

## $`Distances Hom group`
## [1] 2.9483033 2.2460745 1.8035210 0.5921614 1.0697040 1.6595036
##
## $`Distances Het group`
## [1] 3.4284648 2.9132051 3.2307768 0.1688665 1.8169358 0.6471312
##
## $`Inside CI Hom`
## [1] 4
##
## $`Inside CI Het`
## [1] 3
```

```
Sys.sleep(2)
pdf("forestTRIGLYZERIDE.pdf", width = 3, height = 4)
makeforestplots(31) # TRIGLYZERIDE
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 0.8875245
## [1] 0.8721626

## $`Distances Hom group`
## [1] 1.6592692 0.9423728 0.4754285 0.6007903 1.0174436 0.6298425
##
## $`Distances Het group`
## [1] 1.3186548 0.7580118 1.0000830 0.4316932 0.1962184 1.5283141
##
## $`Inside CI Hom`
## [1] 6
##
## $`Inside CI Het`
## [1] 6
```

```
m1<-makeforestplots(11) # BODYWEIGHT_SAC
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 0.9811448
## [1] 1.472004
```

```
m2<-makeforestplots(17) # RELATIVE_ADRENALWEIGHT
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 2.318343
## [1] 1.426996
```

```
m3<-makeforestplots(19) # EPM_TTD
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 0.9860665
## [1] 1.394534

m4<-makeforestplots(21) # EPM_OAE

## Warning: Ignoring unknown aesthetics: x

## [1] 1.536678
## [1] 1.396913

m5<-makeforestplots(22) # EPM_OAT

## Warning: Ignoring unknown aesthetics: x

## [1] 1.196488
## [1] 1.499186

m6<-makeforestplots(23) # TOTALPROTEIN

## Warning: Ignoring unknown aesthetics: x

## [1] 2.187621
## [1] 1.729917

m7<-makeforestplots(24) # ALBUMIN

## Warning: Ignoring unknown aesthetics: x

## [1] 1.459265
## [1] 1.019367

m8<-makeforestplots(25) # GLOBULINE

## Warning: Ignoring unknown aesthetics: x

## [1] 2.323696
## [1] 2.151561

m9<-makeforestplots(26) # KREATININ

## Warning: Ignoring unknown aesthetics: x

## [1] 1.302805
## [1] 1.450015

m10<-makeforestplots(27) # ALAT_GPT

## Warning: Ignoring unknown aesthetics: x

## [1] 3.49246
## [1] 2.189732

m11<-makeforestplots(28) # ASAT_GOT
```

```
## Warning: Ignoring unknown aesthetics: x

## [1] 1.714727
## [1] 1.094719

m12<-makeforestplots(29) # BILIRUBIN

## Warning: Ignoring unknown aesthetics: x

## [1] 1.234921
## [1] 1.146347

m13<-makeforestplots(30) # GLUCOSE

## Warning: Ignoring unknown aesthetics: x

## [1] 1.719878
## [1] 2.03423

m14<-makeforestplots(31) # TRIGLYZERIDE

## Warning: Ignoring unknown aesthetics: x

## [1] 0.8875245
## [1] 0.8721626

alldist<-c(m1$`Distances Hom group`,m2$`Distances Hom group`,m3$`Distances
Hom group`,m4$`Distances Hom group`,m5$`Distances Hom group`,m6$`Distances
Hom group`,m7$`Distances Hom group`,m8$`Distances Hom group`,m9$`Distances
Hom group`,m10$`Distances Hom group`,m11$`Distances Hom group`,m12$`Distances
Hom group`,m13$`Distances Hom group`,m14$`Distances Hom group`,m1$`Distances
Het group`,m2$`Distances Het group`,m3$`Distances Het group`,m4$`Distances
Het group`,m5$`Distances Het group`,m6$`Distances Het group`,m7$`Distances
Het group`,m8$`Distances Het group`,m9$`Distances Het group`,m10$`Distances
Het group`,m11$`Distances Het group`,m12$`Distances Het group`,m13$`Distances
Het group`,m14$`Distances Het group`)

df<-
data.frame(alldist,as.factor(rep(seq(1,2),each=84)),as.factor(rep(rep(seq(1,1
4),each=6),2)),as.factor(rep(seq(1,6),28)))
names(df)<-c("Distance","HomHet","Measure","Lab")
distmod<-lmer(Distance~HomHet+(1|Measure)+(1|Lab),data=df)
distmod

## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: Distance ~ HomHet + (1 | Measure) + (1 | Lab)
##    Data: df
## REML criterion at convergence: 638.7547
## Random effects:
##  Groups   Name        Std.Dev.
##  Measure  (Intercept) 0.2632
##  Lab      (Intercept) 0.5407
```

```
##  Residual             1.5531
## Number of obs: 168, groups:  Measure, 14; Lab, 6
## Fixed Effects:
## (Intercept)      HomHet1
##       1.579        0.088

r.squaredGLMM(distmod)

##               R2m       R2c
## [1,] 0.002800508 0.1328127

anova(distmod)

## Type III Analysis of Variance Table with Satterthwaite's method
##        Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## HomHet 1.3009  1.3009     1   148  0.5393 0.4639

allinsideHom<-c(m1$`Inside CI Hom`,m2$`Inside CI Hom`,m3$`Inside CI
Hom`,m4$`Inside CI Hom`,m5$`Inside CI Hom`,m6$`Inside CI Hom`,m7$`Inside CI
Hom`,m8$`Inside CI Hom`,m9$`Inside CI Hom`,m10$`Inside CI Hom`,m11$`Inside CI
Hom`,m12$`Inside CI Hom`,m13$`Inside CI Hom`,m14$`Inside CI Hom`)/6
100*c(mean(allinsideHom),1.96*sd(allinsideHom)/sqrt(length(varnames)))

## [1] 73.809524  9.511127

# % of estimates within CI (coverage rate) for Hom (plusminus SD)

allinsideHet<-c(m1$`Inside CI Het`,m2$`Inside CI Het`,m3$`Inside CI
Het`,m4$`Inside CI Het`,m5$`Inside CI Het`,m6$`Inside CI Het`,m7$`Inside CI
Het`,m8$`Inside CI Het`,m9$`Inside CI Het`,m10$`Inside CI Het`,m11$`Inside CI
Het`,m12$`Inside CI Het`,m13$`Inside CI Het`,m14$`Inside CI Het`)/6
100*c(mean(allinsideHet),1.96*sd(allinsideHet)/sqrt(length(varnames)))

## [1] 76.19048  7.43519

# % of estimates within CI (coverage rate) for Het
```