

Discovering Associations - Model 1

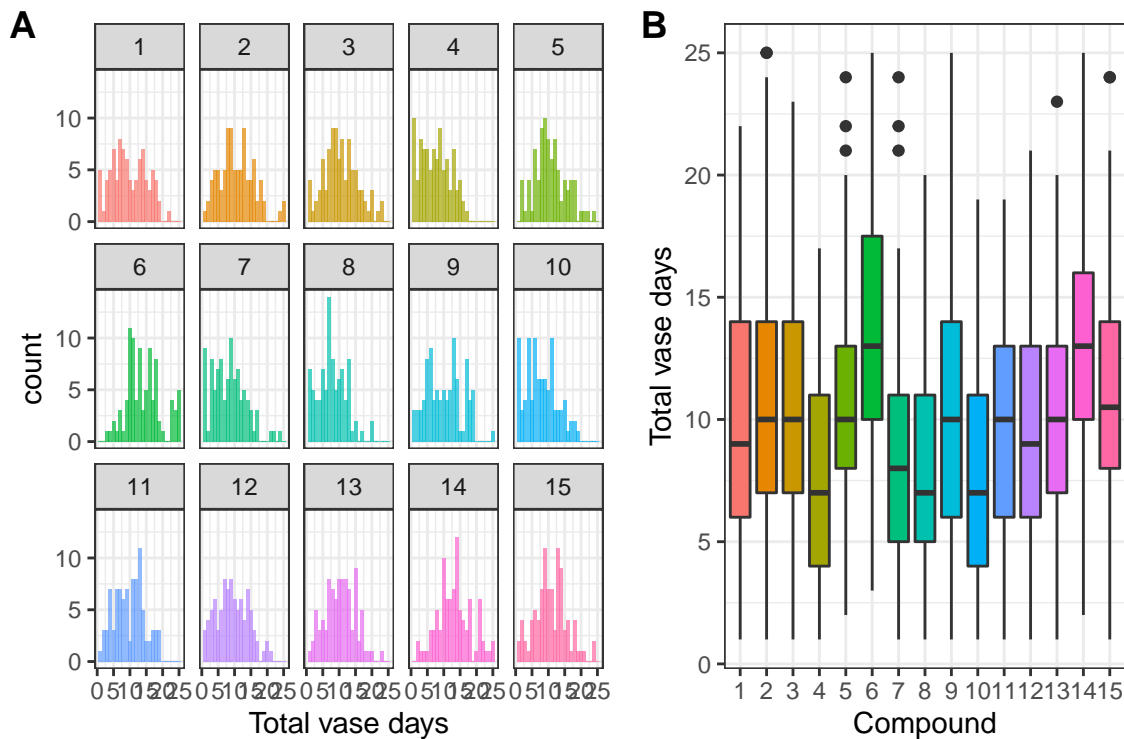
Count data of rose vase life days.

Poisson-mixed effects modelling.

Out of 1440 datapoints, we have 60 missing outcomes (total vase days), which is 4.1% of the data. More data description, can copy some from simulation description.

Checking the distribution of the Vase Days of the count data. It looks like there is a larger degree of overdispersion than the poisson model currently accounts for (nope, it's just conditional on the compound: the goodfit should be done per compound).

```
p1<-ggplot(d, aes(x=tot.vase.days, fill=compound))+geom_histogram(alpha=0.7, bins=25)+facet_wrap(~compound)
p2<-ggplot(d, aes(x=compound,y=tot.vase.days, fill=compound))+geom_boxplot()+theme_bw()+
  theme(legend.position = "none")+ylab("Total vase days")+xlab("Compound")
cowplot::plot_grid(p1,p2, rel_widths = c(0.4,0.33), labels = c("A","B"), scale = 0.97)
```



```
ggsave("figures/mod1_overview.pdf", height=4, width=10)
```

```
#gf <- goodfit(d$tot.vase.days[d$compound==1], "poisson")
#summary(gf)
#plot(gf, type="standing", scale="raw") #I guess the fit would be better if we did it per compound
```

Initial Model (Model 0).

```
#https://bbolker.github.io/mixedmodels-misc/glmmFAQ.html#overdispersion
overdisp_fun <- function(model) {
  rdf <- df.residual(model)
  rp <- residuals(model,type="pearson")
  Pearson.chisq <- sum(rp^2)
  prat <- Pearson.chisq/rdf #pearson residuals over the residual degrees of freedom
  pval <- pchisq(Pearson.chisq, df=rdf, lower.tail=FALSE)
  c(chisq=Pearson.chisq, ratio=prat, rdf=rdf, p=pval)
}
#inspired by the first answer here for nested data formulation https://stats.stackexchange.com/question
glmer_out <- glmer(tot.vase.days ~ compound + species + garden + (1|rater) + (1|subplotID/bushID), fami
summary(glmer_out)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: poisson ( log )
## Formula: tot.vase.days ~ compound + species + garden + (1 | rater) + (1 |
## subplotID/bushID)
## Data: d
##
##      AIC      BIC   logLik deviance df.resid
##  7222.6   7327.2  -3591.3   7182.6     1360
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7555 -0.7878 -0.0460  0.6249  3.7431
##
## Random effects:
##  Groups              Name              Variance Std.Dev.
##  bushID:subplotID (Intercept) 0.01129  0.1062
##  subplotID         (Intercept) 0.05483  0.2342
##  rater              (Intercept) 0.12488  0.3534
## Number of obs: 1380, groups:  bushID:subplotID, 96; subplotID, 16; rater, 6
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.135109   0.171292  12.465 < 2e-16 ***
## compound2    0.070573   0.046365   1.522 0.127983
## compound3    0.063235   0.046126   1.371 0.170402
## compound4   -0.272908   0.050428  -5.412 6.24e-08 ***
## compound5    0.062466   0.046941   1.331 0.183275
## compound6    0.346817   0.043616   7.952 1.84e-15 ***
## compound7   -0.167142   0.048947  -3.415 0.000638 ***
## compound8   -0.248027   0.050442  -4.917 8.79e-07 ***
## compound9    0.042656   0.046612   0.915 0.360130
## compound10  -0.223922   0.049918  -4.486 7.26e-06 ***
## compound11   0.003953   0.047305   0.084 0.933398
```

```
## compound12 -0.055445 0.047967 -1.156 0.247726
## compound13 0.046949 0.046541 1.009 0.313081
## compound14 0.292410 0.044307 6.600 4.12e-11 ***
## compound15 0.082048 0.045998 1.784 0.074467 .
## species2 -0.033621 0.028069 -1.198 0.231002
## garden2 0.186685 0.120431 1.550 0.121109
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Correlation matrix not shown by default, as p = 17 > 12.
## Use print(x, correlation=TRUE) or
## vcov(x) if you need it
```

```
-2*logLik(glmer_out)
```

```
## 'log Lik.' 7182.625 (df=20)
```

```
AIC(glmer_out)
```

```
## [1] 7222.625
```

```
BIC(glmer_out)
```

```
## [1] 7327.222
```

```
overdisp_fun(glmer_out) #quite ok! no significance, yey. i think this might be more relevant than good
```

```
##          chisq      ratio      rdf      p
## 1416.6181038 1.0416310 1360.0000000 0.1392638
```

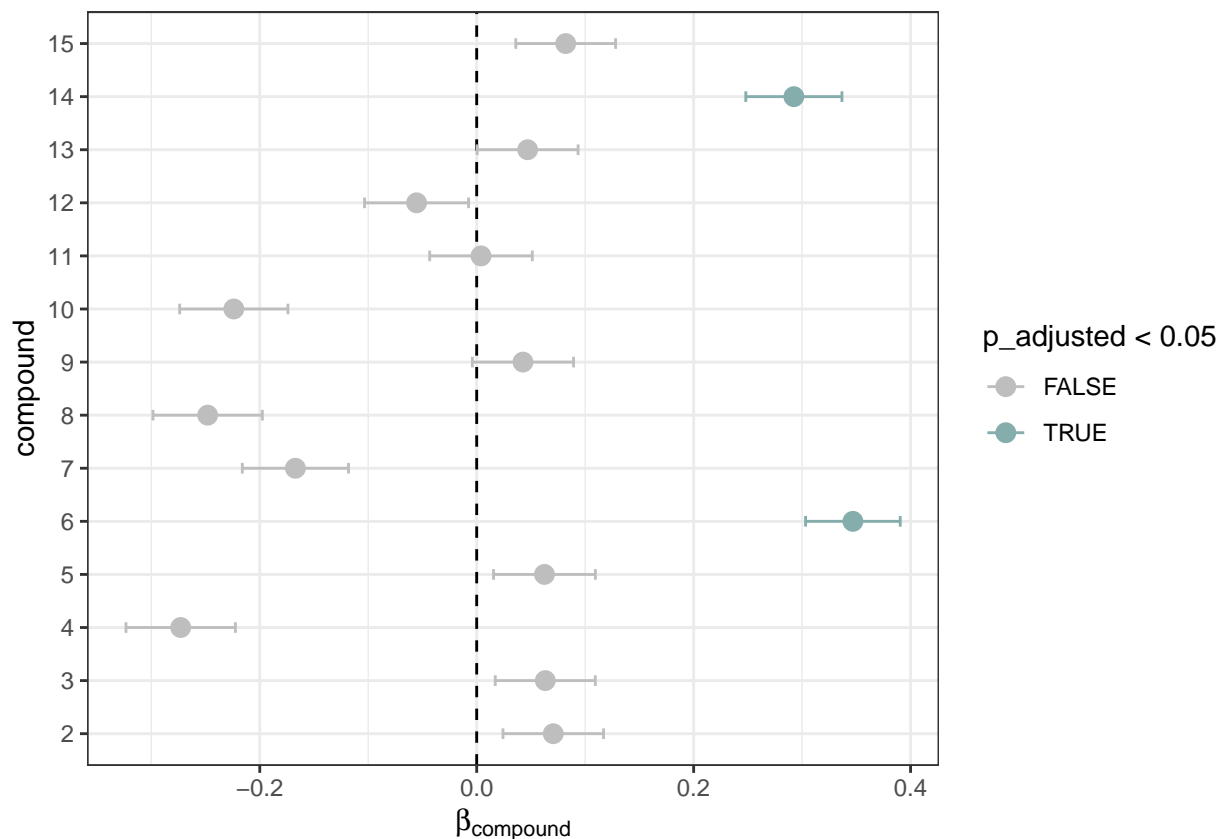
```
#https://fukamilab.github.io/BIO202/04-B-binary-data.html#glm_for_count_data
drop1(glmer_out, test = "Chi")
```

```
## Single term deletions
##
## Model:
## tot.vase.days ~ compound + species + garden + (1 | rater) + (1 |
## subplotID/bushID)
##          npar      AIC      LRT Pr(Chi)
## <none>          7222.6
## compound    14 7615.7 421.07 <2e-16 ***
## species      1 7222.1  1.43 0.2315
## garden       1 7222.9  2.23 0.1356
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

glmer_coefficients<-as.data.frame(summary(glmer_out)$coeff) %>%
  rownames_to_column("predictor") %>%
  filter(grepl("compound",predictor)) %>%
  dplyr::rename(pval=`Pr(>|z|)` ) %>%
  #we want to have p-adjusted (Holm) values for one-sided test H.alt: lambda(compound)>lambda(water)
  dplyr::mutate(one_sided_pval=ifelse(`z value`>0, pval/2, (1-pval/2)),
    p_adjusted=p.adjust(one_sided_pval, method="holm"),
    significant_higher=ifelse(p_adjusted<0.05, T, F))
ggplot(glmer_coefficients %>%
  mutate(compound=factor(gsub("compound","",predictor), levels=2:15)),
  aes(x=compound, y=Estimate, color=p_adjusted<0.05))+
  geom_hline(yintercept=0, linetype="dashed")+
  geom_errorbar(aes(ymin=Estimate - `Std. Error`, ymax=Estimate + `Std. Error`), width=0.2)+geom_point(s
  scale_color_manual(values=c("grey", "#85ADAC"))+coord_flip()+
  ylab(expression(beta["compound"]))

```



Conclusion: compounds 6 and 14 significantly increase rose vase days (mention estimates +- sd error, also backcalculated in days, alpha, one-sided Wald test, maybe the exact z and p-values, correction Holm).

Models comparison.

Model 1. Dropping Compounds.

```

glmer_out1 <- glmer(tot.vase.days ~ species + garden + (1|rater) + (1|subplotID/bushID), family=poisson)
# -2*logLik(glmer_out1)
# AIC(glmer_out1)
# BIC(glmer_out1)
# LRT.1=2*(logLik(glmer_out)-logLik(glmer_out1))
# c(LRT.1,1-pchisq(LRT.1,1))

mod1<-as.data.frame(anova(glmer_out1,glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 1","Full model"),
         dropped_predictor=c("Compound","-"))%>%
  relocate(c(Model, dropped_predictor))

```

Model 2. Dropping species.

```

glmer_out2 <- glmer(tot.vase.days ~ compound + garden + (1|rater) + (1|subplotID/bushID), family=poisson)

mod2<-as.data.frame(anova(glmer_out2,glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 2","Full model"),
         dropped_predictor=c("Species","-"))%>%
  relocate(c(Model, dropped_predictor))

```

Model 3. Dropping garden.

```

glmer_out3 <- glmer(tot.vase.days ~ compound + species + (1|rater) + (1|subplotID/bushID), family=poisson)

mod3<-as.data.frame(anova(glmer_out3,glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 3","Full model"),
         dropped_predictor=c("Garden","-"))%>%
  relocate(c(Model, dropped_predictor))

```

Model 4.1 - Dropping bushID random effect.

```

glmer_out4 <- glmer(tot.vase.days ~ compound + garden + species + (1|rater) + (1|subplotID), family=poisson)

mod4<-as.data.frame(anova(glmer_out4,glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 4","Full model"),
         dropped_predictor=c("BushID","-"))%>%
  relocate(c(Model, dropped_predictor))

```

Model 4.2 - Dropping subplotID random effect

```

glmer_out5 <- glmer(tot.vase.days ~ compound + garden + species + (1|rater) + (1|bushID), family=poisson)

mod5<-as.data.frame(anova(glmer_out5,glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 5","Full model"),
         dropped_predictor=c("subplotID","-"))%>%
  relocate(c(Model, dropped_predictor))

```

Dropping rater as random effect.

```
glmer_out6 <- glmer(tot.vase.days ~ compound + garden + species + (1|subplotID/bushID), family=poisson(
mod6<-as.data.frame(anova(glmer_out6, glmer_out, test="LRT")) %>%
  mutate(Model=c("Model 6", "Full model"),
         dropped_predictor=c("Rater", "-")) %>%
  relocate(c(Model, dropped_predictor))
```

Model 7 (dropping Species and Garden) is generated and compared to models 2 (dropping Species) and 3 (dropping Garden).

```
glmer_out7 <- glmer(tot.vase.days ~ compound + (1|subplotID/bushID) + (1|rater), family=poisson(link =
# -2*logLik(glmer_out6)
# AIC(glmer_out6)
# BIC(glmer_out6)
# LRT.6.2=2*(logLik(glmer_out2)-logLik(glmer_out6))
# c(LRT.6.2, 1-pchisq(LRT.6.2, 1))
# LRT.6.3=2*(logLik(glmer_out3)-logLik(glmer_out6))
# c(LRT.6.3, 1-pchisq(LRT.6.3, 1))

mod7.1<-as.data.frame(anova(glmer_out7, glmer_out2, test="LRT")) %>%
  mutate(Model=c("Model 7", "Model 2"),
         dropped_predictor=c("Species", "Garden", "Species")) %>%
  relocate(c(Model, dropped_predictor))

mod7.2<-as.data.frame(anova(glmer_out7, glmer_out3, test="LRT")) %>%
  mutate(Model=c("Model 7", "Model 3"),
         dropped_predictor=c("Species", "Garden", "Garden")) %>%
  relocate(c(Model, dropped_predictor))
```

Model 7 is compared with models lacking any of the random effects. Model 8 without subplotID

```
glmer_out8 <- glmer(tot.vase.days ~ compound + (1|bushID) + (1|rater), family=poisson(link = "log"), da
# -2*logLik(glmer_out7)
# AIC(glmer_out7)
# BIC(glmer_out7)
# LRT.7=2*(logLik(glmer_out6)-logLik(glmer_out7))
# c(LRT.7, 1-pchisq(LRT.7, 1))

mod8<-as.data.frame(anova(glmer_out8, glmer_out7, test="LRT")) %>%
  mutate(Model=c("Model 8", "Model 7"),
         dropped_predictor=c("Species", "Garden", subplotID, "Species", "Garden")) %>%
  relocate(c(Model, dropped_predictor))
```

Model 9 without bushID.

```
glmer_out9 <- glmer(tot.vase.days ~ compound + (1|subplotID) + (1|rater), family=poisson(link = "log"),
# -2*logLik(glmer_out8)
# AIC(glmer_out8)
# BIC(glmer_out8)
# LRT.8=2*(logLik(glmer_out6)-logLik(glmer_out8))
```

```
# c(LRT.8,1-pchisq(LRT.8,1))

mod9<-as.data.frame(anova(glmer_out9,glmer_out7, test="LRT")) %>%
  mutate(Model=c("Model 9","Model 7"),
         dropped_predictor=c("Species, Garden, bushID","Species, Garden")) %>%
  relocate(c(Model, dropped_predictor))
```

Model 10 without rater.

```
glmer_out10 <- glmer(tot.vase.days ~ compound + (1|subplotID/bushID), family=poisson(link = "log"), data=
# -2*logLik(glmer_out9)
# AIC(glmer_out9)
# BIC(glmer_out9)
# LRT.9=2*(logLik(glmer_out6)-logLik(glmer_out9))
# c(LRT.9,1-pchisq(LRT.9,1))

mod10<-as.data.frame(anova(glmer_out10,glmer_out7, test="LRT")) %>%
  mutate(Model=c("Model 10","Model 7"),
         dropped_predictor=c("Species, Garden, Rater","Species, Garden")) %>%
  relocate(c(Model, dropped_predictor))

modselection<-bind_rows(mod1,mod2,mod3,mod4,mod5,mod6,mod7.1,mod7.2,mod8,mod9,mod10) %>%
  dplyr::rename("Dropped predictor"=dropped_predictor) %>%
  mutate(npar=as.factor(npar),
         Df=as.factor(Df))

print(xtable(modselection,digits=c(1,1,1,1,1,1,1,1,2,1,3)),
      include.rownames=FALSE, file="output_xtables/modelselection1.tex") #only added additional hlines
```

Results using model 7

```
p3<-plot(glmer_out7)
summary(glmer_out7)

## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: poisson ( log )
## Formula: tot.vase.days ~ compound + (1 | subplotID/bushID) + (1 | rater)
## Data: d
##
##      AIC      BIC   logLik deviance df.resid
##  7222.3   7316.4  -3593.1   7186.3     1362
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7683 -0.7958 -0.0394  0.6268  3.6864
##
## Random effects:
## Groups           Name          Variance Std.Dev.
## bushID:subplotID (Intercept) 0.01157  0.1076
```

```
## subplotID      (Intercept) 0.06436  0.2537
## rater          (Intercept) 0.12580  0.3547
## Number of obs: 1380, groups:  bushID:subplotID, 96; subplotID, 16; rater, 6
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.211664   0.162032  13.650 < 2e-16 ***
## compound2    0.070518   0.046365   1.521 0.128280
## compound3    0.063256   0.046126   1.371 0.170255
## compound4   -0.273139   0.050428  -5.416 6.08e-08 ***
## compound5    0.062239   0.046941   1.326 0.184866
## compound6    0.346864   0.043616   7.953 1.82e-15 ***
## compound7   -0.167281   0.048947  -3.418 0.000632 ***
## compound8   -0.248209   0.050442  -4.921 8.62e-07 ***
## compound9    0.042485   0.046612   0.911 0.362052
## compound10  -0.223806   0.049917  -4.484 7.34e-06 ***
## compound11   0.003961   0.047305   0.084 0.933272
## compound12  -0.055097   0.047966  -1.149 0.250693
## compound13   0.046986   0.046541   1.010 0.312699
## compound14   0.292286   0.044307   6.597 4.20e-11 ***
## compound15   0.081856   0.045998   1.780 0.075147 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 15 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it
```

```
#this is where emmeans package comes in
glmer1.rg <- ref_grid(glmer_out7)
glmer1.emm.s <- emmeans(glmer1.rg, "compound")

backcalc<-confint(glmer1.emm.s, level = .9, type = "response") %>%
  mutate(predictor=paste0("compound",compound))

# ctrl<-exp(summary(glmer_out)$coeff)[1,1]
# comp6<-exp(summary(glmer_out)$coeff)[6,1]
# comp14<-exp(summary(glmer_out)$coeff)[14,1]
# ctrl*comp6-ctrl
# ctrl*comp14-ctrl
ctrl1_noExp<-summary(glmer_out7)$coeff[1,1]
ctrl1<-exp(summary(glmer_out7)$coeff)[1,1]
ctrl1_stderr<-exp(summary(glmer_out7)$coeff)[1,2]

glmer_coefficients<-as.data.frame(summary(glmer_out7)$coeff) %>%
  rownames_to_column("predictor") %>%
  filter(grepl("compound",predictor)) %>%
  dplyr::rename(pval=`Pr(>|z|)` ) %>%
#we want to have p-adjusted (Holm) values for one-sided test H.alt: lambda(compound)>lambda(water)
dplyr::mutate(one_sided_pval=ifelse(`z value`>0, pval/2, (1-pval/2)),
  p_adjusted=p.adjust(one_sided_pval, method="holm"),
```



```

    significant_higher=ifelse(p_adjusted<0.05, T, F),
    Estimate_days=exp(Estimate)*ctrl1,
    Std.Error_days=exp(`Std. Error`)*ctrl1_stderr) %>%
left_join(backcalc)

## Joining, by = "predictor"

#recheck backcalculation
cor(glmer_coefficients$Estimate_days,glmer_coefficients$rate)

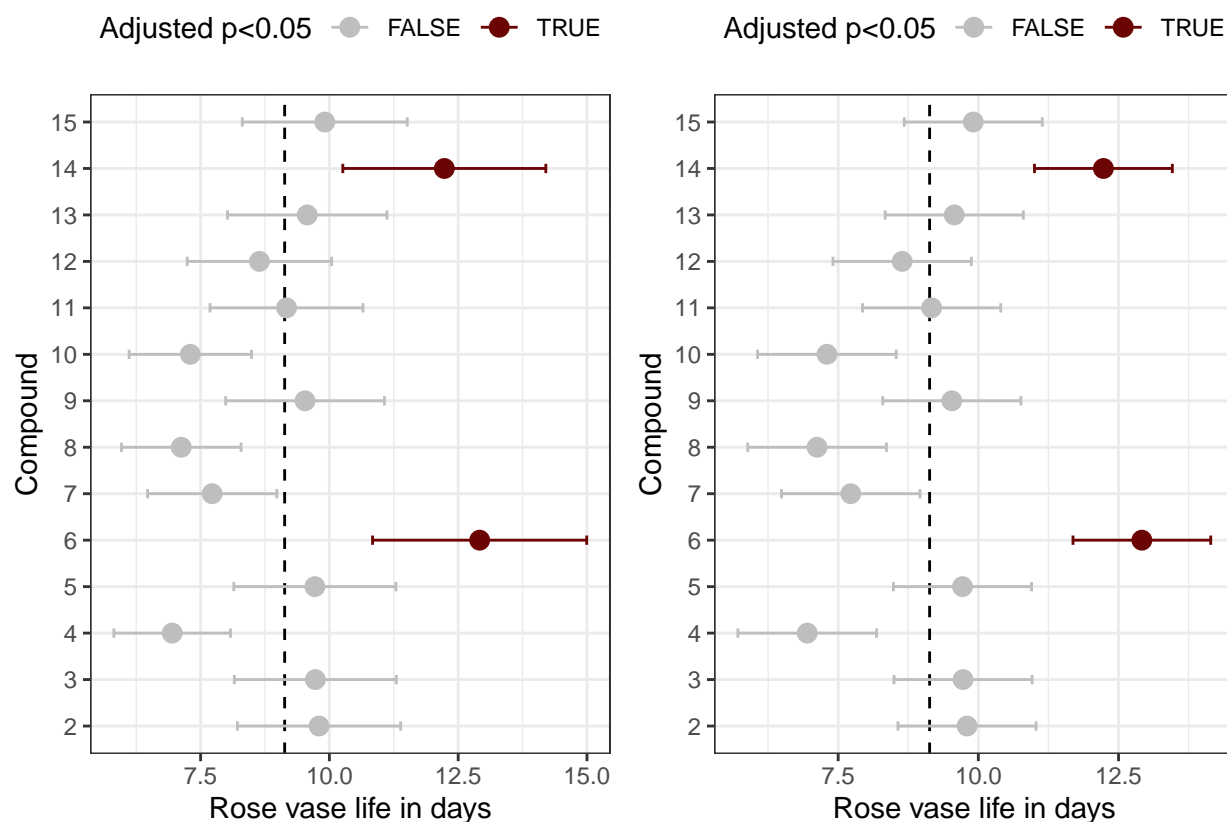
## [1] 1

pCIs<-ggplot(glmer_coefficients %>%
  mutate(compound=factor(gsub("compound","",predictor), levels=2:15)),
  aes(x=compound, y=Estimate_days, color=p_adjusted<0.05))+
  geom_hline(yintercept=ctrl1, linetype="dashed")+
  geom_errorbar(aes(ymin=asyp.LCL, ymax=asyp.UCL), width=0.2)+geom_point(size=3)+theme_bw()+
  scale_color_manual(values=c("grey","#6B0504"), name="Adjusted p<0.05")+coord_flip()+
  ylab("Rose vase life in days")+
  xlab("Compound")+
  theme(legend.position = "top")

cowplot::plot_grid(
ggplot(glmer_coefficients %>%
  mutate(compound=factor(gsub("compound","",predictor), levels=2:15)),
  aes(x=compound, y=Estimate_days, color=p_adjusted<0.05))+
  geom_hline(yintercept=ctrl1, linetype="dashed")+
  geom_errorbar(aes(ymin=Estimate_days - SE, ymax=Estimate_days + SE), width=0.2)+geom_point(size=3)+th
  scale_color_manual(values=c("grey","#6B0504"), name="Adjusted p<0.05")+coord_flip()+
  ylab("Rose vase life in days")+
  xlab("Compound")+
  theme(legend.position = "top"),

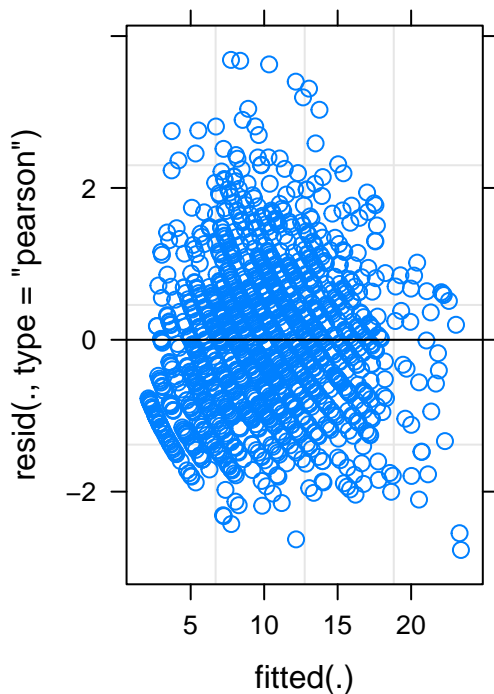
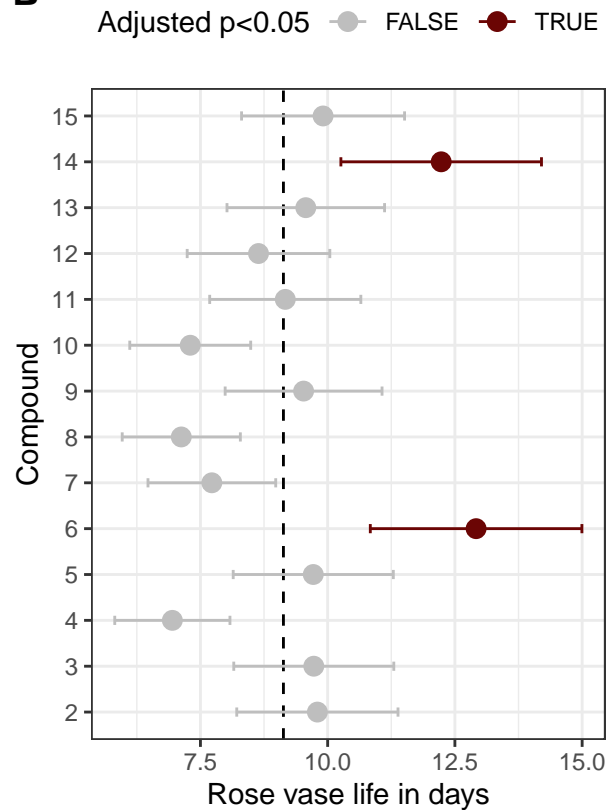
ggplot(glmer_coefficients %>%
  mutate(compound=factor(gsub("compound","",predictor), levels=2:15)),
  aes(x=compound, y=Estimate_days, color=p_adjusted<0.05))+
  geom_hline(yintercept=ctrl1, linetype="dashed")+
  geom_errorbar(aes(ymin=Estimate_days - Std.Error_days, ymax=Estimate_days + Std.Error_days), width=0.
  scale_color_manual(values=c("grey","#6B0504"), name="Adjusted p<0.05")+coord_flip()+
  ylab("Rose vase life in days")+
  xlab("Compound")+
  theme(legend.position = "top")
)

```



```
p4<-ggplot(glmer_coefficients %>%
  mutate(compound=factor(gsub("compound","",predictor), levels=2:15)),
  aes(x=compound, y=Estimate_days, color=p_adjusted<0.05))+
  geom_hline(yintercept=ctrl1, linetype="dashed")+
  geom_errorbar(aes(ymin=Estimate_days - SE, ymax=Estimate_days +SE), width=0.2)+geom_point(size=3)+
  scale_color_manual(values=c("grey","#B0504"), name="Adjusted p<0.05")+coord_flip()+
  ylab("Rose vase life in days")+
  xlab("Compound")+
  theme(legend.position = "top")
#scale_()
#theme(axis.title.x = element_text(size=15))

cowplot::plot_grid(p3,p4, scale = c(0.93,0.99), labels = c("A","B"))
```

A**B**

```
ggsave("figures/mod1_diagn_outcome.pdf", height=4.8, width=8.3)
```

```
ggsave("figures/mod1_diagnostics.pdf",cowplot::plot_grid(p3), height=3, width=4.5)
```

```
ggsave("figures/mod1_outcome.pdf",p4, height=5, width=4.5)
```

Contrasts

Now do contrasts of the promising compounds.

<https://stats.stackexchange.com/questions/60352/comparing-levels-of-factors-after-a-glm-in-r>
<https://stat.ethz.ch/~meier/teaching/anova/contrasts-and-multiple-testing.html>

```
wrap_glht_output<-function(glht_output){
```

```
  pq<-summary(glht_output)$test
```

```
  mtests <- cbind(pq$coefficients, pq$sigma, pq$tstat, pq$pvalues)
```

```
  error <- attr(pq$pvalues, "error")
```

```
  pname <- switch(glht_output$alternativ,
```

```
    less = paste("Pr(<", ifelse(glht_output$df ==0, "z", "t"), ")"), sep = ""),
```

```
    greater = paste("Pr(>", ifelse(glht_output$df == 0, "z", "t"), ")"), sep = ""),
```

```
    two.sided = paste("Pr(>|", ifelse(glht_output$df == 0, "z", "t"), "|")", sep = ""))
```

```
  colnames(mtests) <- c("Estimate", "Std. Error", ifelse(glht_output$df ==0, "z value", "t value"), pna
```

```

    return(mtests)
}

summary(glmer_out7)

## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
## Family: poisson ( log )
## Formula: tot.vase.days ~ compound + (1 | subplotID/bushID) + (1 | rater)
## Data: d
##
##      AIC      BIC    logLik deviance df.resid
##  7222.3   7316.4  -3593.1   7186.3     1362
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.7683 -0.7958 -0.0394  0.6268  3.6864
##
## Random effects:
## Groups              Name              Variance Std.Dev.
## bushID:subplotID (Intercept) 0.01157  0.1076
## subplotID         (Intercept) 0.06436  0.2537
## rater              (Intercept) 0.12580  0.3547
## Number of obs: 1380, groups:  bushID:subplotID, 96; subplotID, 16; rater, 6
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.211664   0.162032  13.650 < 2e-16 ***
## compound2    0.070518   0.046365   1.521 0.128280
## compound3    0.063256   0.046126   1.371 0.170255
## compound4   -0.273139   0.050428  -5.416 6.08e-08 ***
## compound5    0.062239   0.046941   1.326 0.184866
## compound6    0.346864   0.043616   7.953 1.82e-15 ***
## compound7   -0.167281   0.048947  -3.418 0.000632 ***
## compound8   -0.248209   0.050442  -4.921 8.62e-07 ***
## compound9    0.042485   0.046612   0.911 0.362052
## compound10  -0.223806   0.049917  -4.484 7.34e-06 ***
## compound11   0.003961   0.047305   0.084 0.933272
## compound12  -0.055097   0.047966  -1.149 0.250693
## compound13   0.046986   0.046541   1.010 0.312699
## compound14   0.292286   0.044307   6.597 4.20e-11 ***
## compound15   0.081856   0.045998   1.780 0.075147 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 15 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it

x<-glht(glmer_out7, linfct=mcp(compound=c("`6` - `14` == 0")))
mtests1<-wrap_glht_output(x)

```

```
xtable(mtests1, digits=4)
```

```
## % latex table generated in R 4.1.1 by xtable 1.8-4 package
## % Thu Jun 2 18:08:33 2022
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrrr}
## \hline
## & Estimate & Std. Error & z value & Pr(>|z|) & \\
## \hline
## `6` - `14` & 0.0546 & 0.0401 & 1.3618 & 0.1733 & \\
## \hline
## \end{tabular}
## \end{table}
```

```
print(xtable(mtests1,digits=c(1,2,3,3,4)), file="output_xtables/mod1_contrast_compound6vs14.tex")
```

```
rawCI<-confinf(glmer1.emm.s, calc = c(n = ~.wgt.)) %>%
  mutate(Predictor=paste0("compound",compound),
         Predictor=ifelse(compound==1,"(Intercept)",Predictor))

glmer_mainTable<-as.data.frame(summary(glmer_out7)$coeff) %>%
  rownames_to_column("Predictor") %>%
  left_join(rawCI %>% dplyr::select(Predictor,n)) %>%
  mutate(lowerCI=Estimate-qt(0.9,df=n-1)*`Std. Error`,
         upperCI=Estimate+qt(0.9,df=n-1)*`Std. Error`, #use 0.9 because we are doing a one sided test h
         n=as.factor(n))
```

```
## Joining, by = "Predictor"
```

```
print(xtable(glmer_mainTable,digits=c(1,1,2,3,3,4,1,3,3)),
      include.rownames=FALSE, file="output_xtables/mod1_mainTable.tex")
```

```
#ranef(glmer_out7)
#https://stackoverflow.com/questions/8526681/extract-random-effect-variances-from-lme4-mer-model-object
glmer_randomTable<-as.data.frame(VarCorr(glmer_out7)) %>% dplyr::select(-var2)

print(xtable(glmer_randomTable,digits=c(1,1,1,3,3)),
      include.rownames=FALSE, file="output_xtables/mod1_randomTable.tex")
```