

Figure 3b

```
library(readxl)
library(lubridate)

##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
library(mgcv)

## Loading required package: nlme
## This is mgcv 1.8-34. For overview type 'help("mgcv-package")'.
library(ggplot2)
library(ggsignif)
library(readr)
library(emmeans)
library(tidyr)
library(dplyr)

##
## Attaching package: 'dplyr'
## The following object is masked from 'package:nlme':
##
##     collapse
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
library(plyr)

## -----
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## -----
##
## Attaching package: 'plyr'
## The following objects are masked from 'package:dplyr':
##
```

```

##      arrange, count, desc, failwith, id, mutate, rename, summarise,
##      summarize
#load in raw data
dat <- data.frame(read_csv("3b_data.csv"))

##
## -- Column specification -----
## cols(
##   `Repeat(Date)` = col_datetime(format = ""),
##   Algae = col_character(),
##   Treatment = col_character(),
##   Well_ID = col_character(),
##   Larvae = col_double(),
##   `Number of Symbionts` = col_double()
## )

head(dat)

##   Repeat.Date. Algae Treatment      Well_ID Larvae Number.of.Symbionts
## 1  2019-07-12 SSB01      DMSO S_D_2019-07-12      1              0
## 2  2019-07-12 SSB01      DMSO S_D_2019-07-12      2              0
## 3  2019-07-12 SSB01      DMSO S_D_2019-07-12      3              0
## 4  2019-07-12 SSB01      DMSO S_D_2019-07-12      4              3
## 5  2019-07-12 SSB01      DMSO S_D_2019-07-12      5              4
## 6  2019-07-12 SSB01      DMSO S_D_2019-07-12      6              0

#clean up raw data
dat$Algae <- factor(dat$Algae, levels= c("SSB01", "M. gaditana", "N. oculata", "C. velia"))
dat$Treatment <- factor(dat$Treatment)
dat$Repeat.Date. <- factor(as.character(dat$Repeat.Date.))
dat$Well_ID <- factor(dat$Well_ID)
dat$Number.of.Symbionts <- ifelse(dat$Number.of.Symbionts>1,1,dat$Number.of.Symbionts)
head(dat)

##   Repeat.Date. Algae Treatment      Well_ID Larvae Number.of.Symbionts
## 1  2019-07-12 SSB01      DMSO S_D_2019-07-12      1              0
## 2  2019-07-12 SSB01      DMSO S_D_2019-07-12      2              0
## 3  2019-07-12 SSB01      DMSO S_D_2019-07-12      3              0
## 4  2019-07-12 SSB01      DMSO S_D_2019-07-12      4              1
## 5  2019-07-12 SSB01      DMSO S_D_2019-07-12      5              1
## 6  2019-07-12 SSB01      DMSO S_D_2019-07-12      6              0

#model data with generalised linear mixed modes
summary <- data.frame()

for(i in unique(dat$Algae))
{
  #i <- unique(dat$Algae)[1]
  Dat <- droplevels(dat[dat$Algae==i,])
  ##Regression model:
  mod <- gam(Number.of.Symbionts ~ Treatment + s(Repeat.Date.,bs='re') + s(Well_ID,bs='re'),data=Dat,fam="poisson")
  S <- summary(mod)$p.table
  if(S[2,4]>0.1)
  {
    p <- ""
  }
}

```

```

if(S[2,4]<0.1)
{
  p <- ""
}
if(S[2,4]<0.05)
{
  p <- "*"
}
if(S[2,4]<0.01)
{
  p <- "**"
}
if(S[2,4]<0.001)
{
  p <- "***"
}
newline <- data.frame(Algae=i,outcome=S[1,1],SE=S[1,2],type="DMSO",p=NA,pvalue=NA)
summary <- rbind(summary,newline)
newline <- data.frame(Algae=i,outcome=S[1,1] + S[2,1],SE=S[2,2],type="XMD17-109",p=p, pvalue=S[2,4])
summary <- rbind(summary,newline)
}

summary$up <- plogis(summary$outcome + 1.96*summary$SE)
summary$down <- plogis(summary$outcome - 1.96*summary$SE)
summary$outcome <- plogis(summary$outcome)

summary

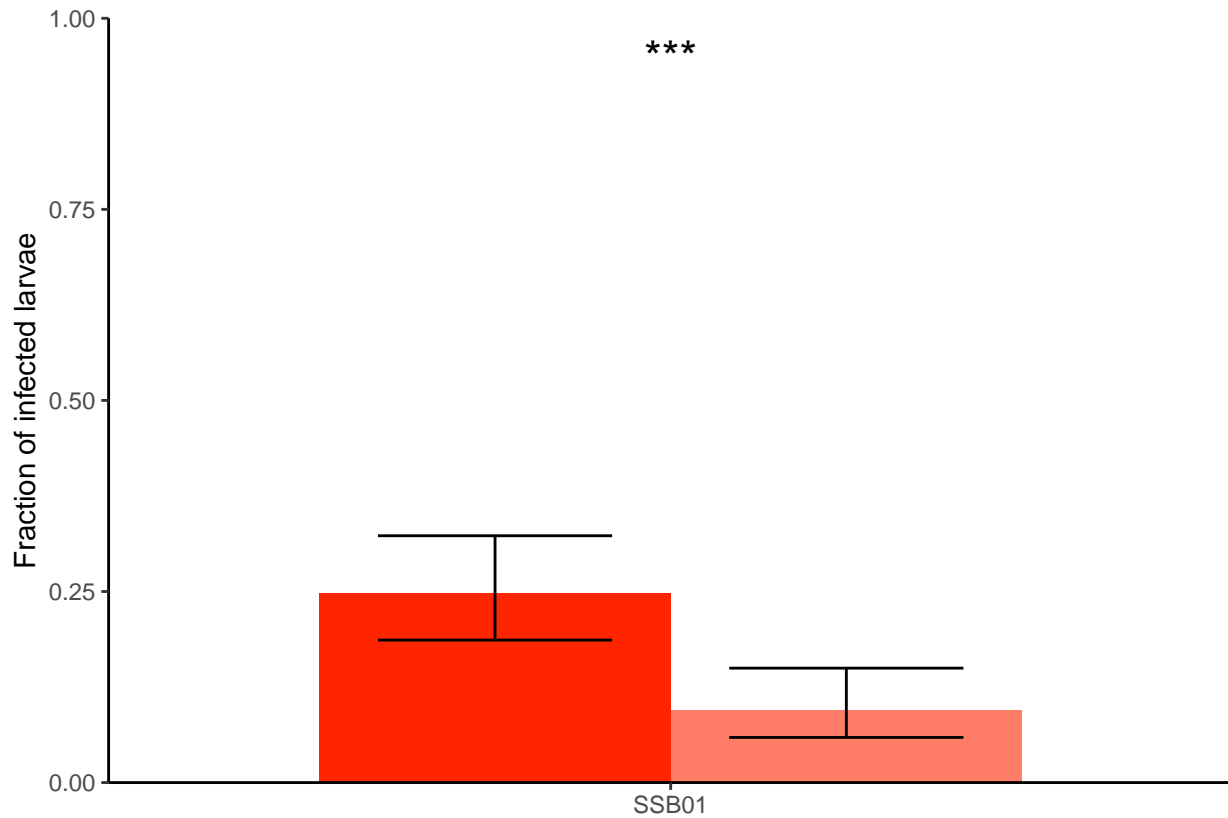
##   Algae   outcome      SE    type    p      pvalue      up      down
## 1 SSB01 0.24848297 0.1869547   DMSO <NA>      NA 0.3229411 0.18646420
## 2 SSB01 0.09512168 0.2631678 XMD17-109 *** 1.334644e-05 0.1497155 0.05905294

write_csv(summary, "3b_summary.csv")

#Plot graph
P1<-ggplot(data = summary,aes(x = Algae, y = outcome, group = type)) +
  geom_bar(stat = "identity", width = 0.75, size=1.5, position = "dodge", fill = c("#FF2500",alpha("#FF2500",0.5)),
  geom_errorbar(aes(ymin=down, ymax=up), width=.5, position =position_dodge(.75)) +
  geom_text(data = summary, aes(x=Algae,y=0.95),label=summary$p,size=6) +
  theme_classic() +
  labs(x = "", y = "Fraction of infected larvae") +
  scale_y_continuous(expand = c(0, 0), limits = c(0,1), labels = scales::number_format(accuracy = 0.01))
plot(P1)

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

```



```
ggsave("Figure_3b.pdf",width=5.75,height=11,units="cm")
```

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

```
# get n count per treatment
```

```
ndat <- ddpby(dat, c("Algae", "Treatment", "Repeat.Date."), summarise,  
              N = length(Larvae))
```

```
n_Figure <- ddpby(ndat, c("Algae", "Treatment"), summarise,  
                  N = length(Repeat.Date.))
```

```
n_Figure
```

```
##   Algae Treatment N
```

```
## 1 SSB01      DMSO 6
```

```
## 2 SSB01 XMD17-109 6
```

```
write_csv(n_Figure, "3b_n.csv")
```