

# Data Analysis for: Forward genetics in Wolbachia: Regulation of Wolbachia proliferation by the amplification and deletion of an addictive genomic island

Elves H Duarte & Luis Teixeira

22 February, 2021

## Contents

<b>Global settings</b>	<b>2</b>
<b>S1 Fig   Wolbachia recovery time</b>	<b>4</b>
<b>S2 Fig   Effect EMS females fecundity and Wolbachia titers</b>	<b>8</b>
S2A-B Fig   Effect on fecundity . . . . .	8
S2C Fig   Effect on Wolbachia titers (experiment 1) . . . . .	13
S2D Fig   Effect on Wolbachia titers (experiment 2) . . . . .	16
<b>Fig 1   Isolation over-proliferative Wolbachia variants</b>	<b>19</b>
Fig 1A   Isolation Line 1A :: wMelPop2 . . . . .	19
Fig 1B   Isolation Line 2A :: wMelOctoless . . . . .	21
Fig 1C   Wolbachia titres isogenic genetic background . . . . .	22
S3A Fig   Isolation Line 2B . . . . .	24
S3B Fig   Isolation Line 3A :: wMelOctoless2 . . . . .	25
S3C Fig   Isolation Line 3B . . . . .	26
S3D Fig   Isolation Line 3C . . . . .	28
<b>S5 Fig   Characterization of wMelOctoless2</b>	<b>29</b>
<b>S6 Fig   Confirmation Octomom copy number variation</b>	<b>33</b>
<b>S7 Fig   Selection of flies with desired Octomom copy number</b>	<b>46</b>
<b>Fig 3   Wolbachia Proliferation in adults</b>	<b>48</b>
Fig 3   Dataset . . . . .	48
Fig 3   Statistical analysis . . . . .	49
full model. doubling times . . . . .	49
direct comparison of octoless and csb . . . . .	53
comparison of new variants to cs_b . . . . .	55
copy number influence on wMelPop and wMelPop2 . . . . .	60
Fig 3A-C . . . . .	66
S10A-C Fig . . . . .	68
S11 Fig   Wolbachia titers day of eclosion . . . . .	71
<b>S12 Fig Phenotypes wMelPop2 and wMelPop controlled Octomom copy number</b>	<b>74</b>
S12A   Control Octomom copy number . . . . .	74
S12B   Proliferation wMelPop2 and wMelPop . . . . .	80

S12C   Lifespan wMelPop2 and wMelPop . . . . .	85
<b>Fig 4   Wolbachia proliferation in development</b>	<b>90</b>
Fig 4   Standard curve analysis . . . . .	90
Fig 4   Statistical analysis . . . . .	93
Fig 4   final figure . . . . .	99
<b>S13 Fig   Dynamics Octomom copy number</b>	<b>100</b>
S13A Fig   Octomom copy number development . . . . .	100
S13B   Octomom copy number dynamics in adults . . . . .	102
<b>Fig 5   Wolbachia life-shortening phenotype</b>	<b>108</b>
Fig 5   Dataset . . . . .	108
Fig 5   Statistical analysis . . . . .	109
Fig 5 and S14 Fig panels male lifespan . . . . .	120
S14D-E Fig   Wolbachia life-shortening phenotype :: Females . . . . .	127
S15 Fig   Bidirectional crosses . . . . .	130
Fig 5E-G   Correlation Wolbahcia life-shortening phenotype VS Wolbachia doubling time or titres at time 0 . . . . .	136
Fig 5E-G panels . . . . .	137
S16 Fig panels   Correlation Wolbachia life-shortening phenotype VS titers . . . . .	140
<b>Fig 6   Wolbachia anti-viral protection</b>	<b>143</b>
Fig 6   Dataset . . . . .	143
Fig 6   Statistical analysis . . . . .	144
Fig 6A-D . . . . .	149
S16A-B Fig . . . . .	152
Fig 6E   Correlation protection, titres, and proliferation . . . . .	154
Fig 6C correlation figure . . . . .	155
S16D-E Fig . . . . .	156

## Global settings

```
library(multcompView)
library(tidyverse)

## -- Attaching packages -----
## v ggplot2 3.3.0      v purrr  0.3.4
## v tibble  3.0.1      v dplyr  1.0.4
## v tidyr   1.0.2      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.5.0

## -- Conflicts -----
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(lmerTest)

## Loading required package: lme4
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following objects are masked from 'package:tidyr':
```

```

##
##   expand, pack, unpack
##
## Attaching package: 'lmerTest'
## The following object is masked from 'package:lme4':
##
##   lmer
## The following object is masked from 'package:stats':
##
##   step
library(reshape2)

##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##   smiths
library(multcomp)

## Loading required package: mvtnorm
## Loading required package: survival
## Loading required package: TH.data
## Loading required package: MASS

##
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##   select
##
## Attaching package: 'TH.data'
## The following object is masked from 'package:MASS':
##
##   geyser
library(emmeans)
library(cluster)
library(scales)

##
## Attaching package: 'scales'
## The following object is masked from 'package:purrr':
##
##   discard
## The following object is masked from 'package:readr':
##
##   col_factor

```

```
library(coxme)

## Loading required package: bdsmatrix
##
## Attaching package: 'bdsmatrix'
## The following object is masked from 'package:base':
##
##      backsolve

library(car)

## Loading required package: carData
## Registered S3 methods overwritten by 'car':
##   method                      from
##   influence.merMod             lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod      lme4
##   dfbetas.influence.merMod     lme4
##
## Attaching package: 'car'
## The following object is masked from 'package:dplyr':
##
##      recode
## The following object is masked from 'package:purrr':
##
##      some
```

## S1 Fig | Wolbachia recovery time

```
#### S1 data
s1data=read.csv("../original_data/S1 Data.csv", na.strings=c("", "NA", "Undetermined"), stringsAsFactors=FALSE)

#### wolbachia recovery over fly generations
results_qpcr_wolb_recovery=s1data%>%
  dplyr::group_by(fly_generation)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="T_0", reference_gene="RPL32", efficiency_ref=2, efficiency_cal=1))

## Warning: `funs()` is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

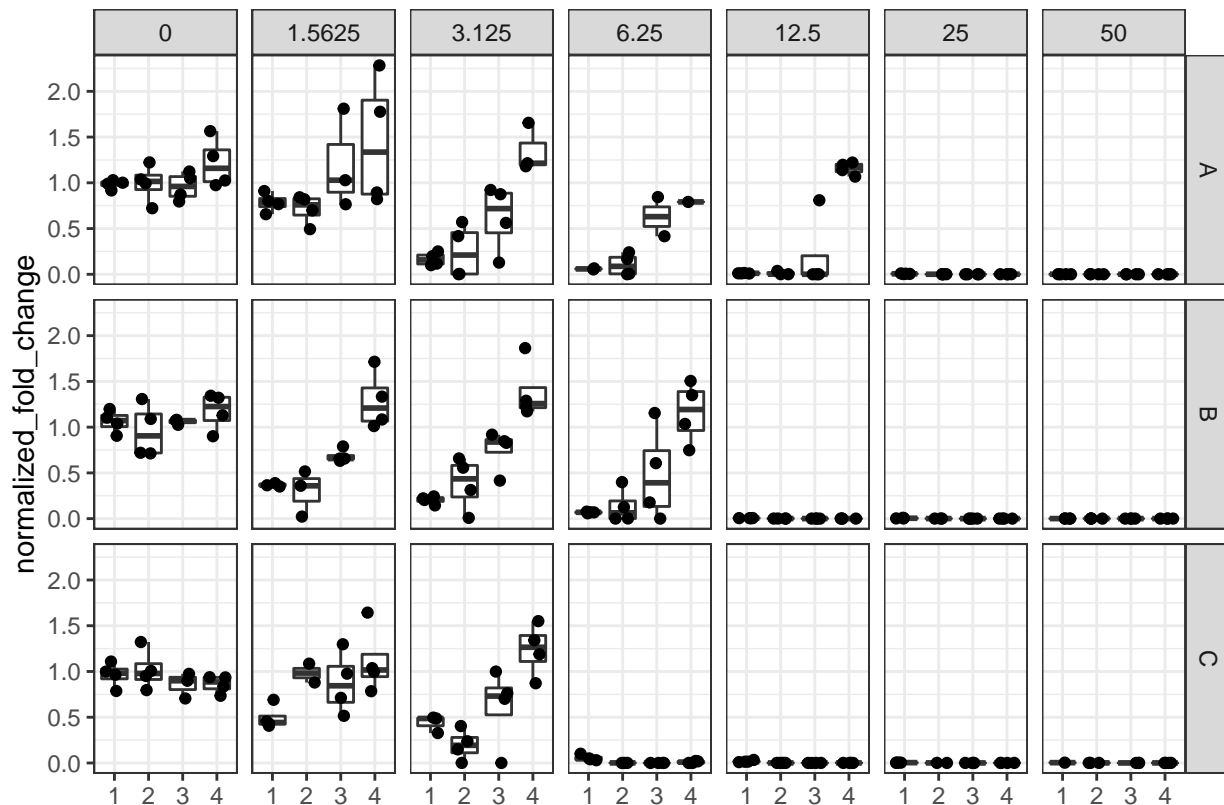
```

### adding new variables
results_qpcr_wolb_recovery=results_qpcr_wolb_recovery%>%
  tidyr::separate(sample, c(NA, "tetracycline", "replicate", NA) , sep = "_", remove = FALSE)

### converting tetracycline variable to factor
results_qpcr_wolb_recovery$tetracycline=factor(results_qpcr_wolb_recovery$tetracycline, levels=c("0", "1.5625", "3.125", "6.25", "12.5", "25", "50"))

### plot
ggplot(results_qpcr_wolb_recovery, aes(x=factor(fly_generation), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.25)+
  facet_grid(replicate~tetracycline)+
  #scale_y_continuous(name="", limits=c(0,NA), breaks=seq(0, 2,0.5))+
  scale_x_discrete(name="")+
  theme_bw()

```



```

### titre range in generation 1 in treated flies
results_qpcr_wolb_recovery %>%
  dplyr::filter(fly_generation == 1 & tetracycline != "0")%>%
  dplyr::select("normalized_fold_change") %>%
  summary()

```

```

## Adding missing grouping variables: `fly_generation`

## fly_generation normalized_fold_change
## Min.      :1      Min.      :0.0003107

```

```
## 1st Qu.:1      1st Qu.:0.0045261
## Median :1      Median :0.0299666
## Mean :1       Mean :0.1511052
## 3rd Qu.:1     3rd Qu.:0.2215953
## Max. :1       Max. :0.9100383

### removing panels where flies do not recover
results_qpcr_wolb_recovery_analysis <- results_qpcr_wolb_recovery %>%
  filter(!tetracycline %in% c("25", "50")) %>%
  filter(!(tetracycline == "12.5" & replicate %in% c("B","C"))) %>%
  filter(!(tetracycline == "6.25" & replicate == "C"))

### the model
model_wolb_recover_lmer=lmer(normalized_fold_change~fly_generation*tetracycline + (1|replicate), data=r

## boundary (singular) fit: see ?isSingular
#this leads to a singular fit indicate no variance associated with replicates
model_wolb_recover_lmer2=lm(fold_change~fly_generation*tetracycline, data=results_qpcr_wolb_recovery_an

# significance of the model
Anova(model_wolb_recover_lmer2)

## Anova Table (Type II tests)
##
## Response: fold_change
##
##          Sum Sq Df F value    Pr(>F)
## fly_generation    10.3734    1 124.439 < 2.2e-16 ***
## tetracycline       10.0076    4  30.013 < 2.2e-16 ***
## fly_generation:tetracycline  4.3417    4  13.021 3.081e-09 ***
## Residuals        13.7546 165
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_wolb_recover_lmer2, cor=FALSE)

##
## Call:
## lm(formula = fold_change ~ fly_generation * tetracycline, data = results_qpcr_wolb_recovery_analysis)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78939 -0.18189  0.01192  0.15972  1.00301
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.023607   0.102079  10.028 < 2e-16 ***
## fly_generation    0.001066   0.037437   0.028  0.977
## tetracycline1.5625 -0.723890   0.148311  -4.881 2.47e-06 ***
## tetracycline3.125 -1.260327   0.147037  -8.571 7.02e-15 ***
## tetracycline6.25  -1.427808   0.175793  -8.122 1.02e-13 ***
## tetracycline12.5  -1.539582   0.207578  -7.417 5.98e-12 ***
## fly_generation:tetracycline1.5625  0.220114   0.053612   4.106 6.33e-05 ***
## fly_generation:tetracycline3.125  0.341133   0.053885   6.331 2.22e-09 ***
## fly_generation:tetracycline6.25  0.332045   0.066172   5.018 1.34e-06 ***
```

```

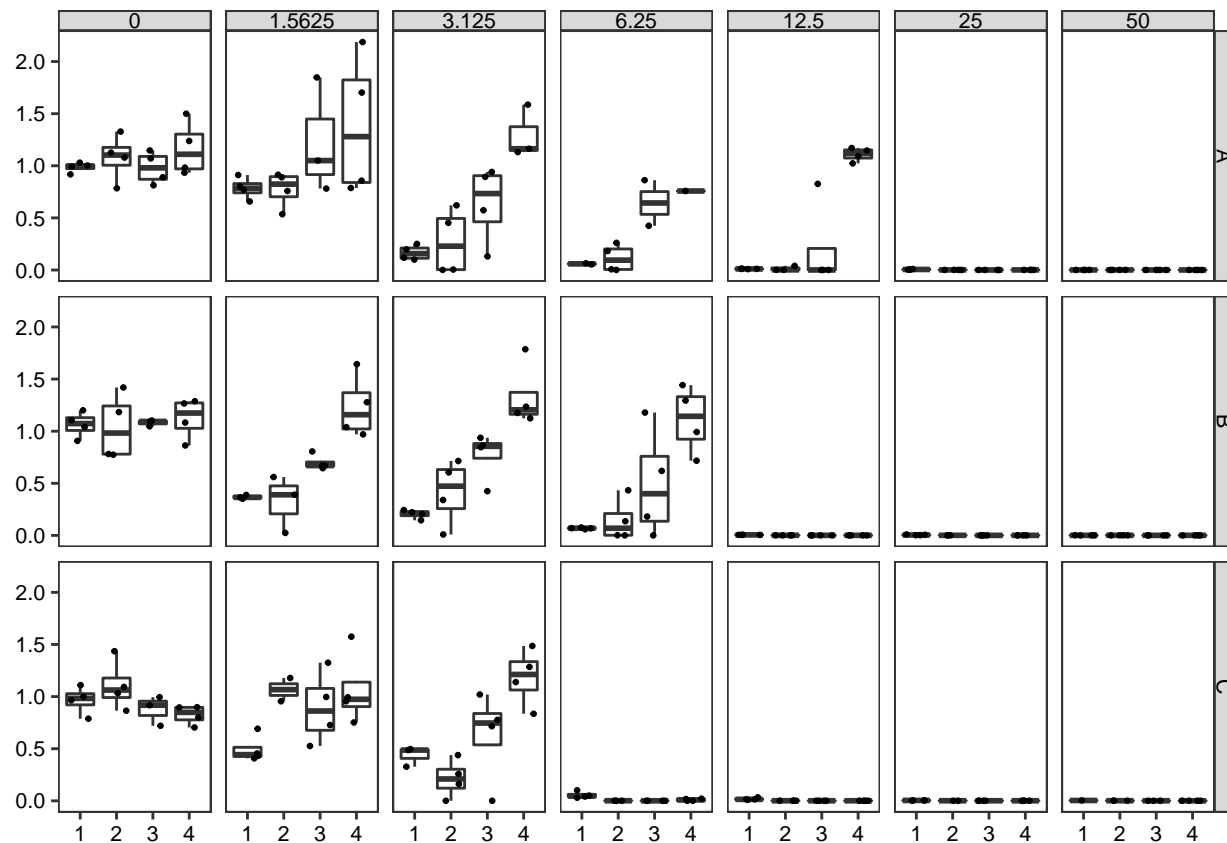
## fly_generation:tetracycline12.5    0.343275    0.075006    4.577 9.25e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2887 on 165 degrees of freedom
## Multiple R-squared:  0.6451, Adjusted R-squared:  0.6258
## F-statistic: 33.33 on 9 and 165 DF,  p-value: < 2.2e-16

# multiple comparison
multiple_comparison_wolb_recovery=emmeans::emmeans(model_wolb_recover_lmer2, ~ tetracycline|fly_generat.
#
contrast(multiple_comparison_wolb_recovery, "trt.vs.ctrl", adj="holm")

## fly_generation = 1:
## contrast estimate SE df t.ratio p.value
## 1.5625 - 0 -0.5038 0.1019 165 -4.943 <.0001
## 3.125 - 0 -0.9192 0.1003 165 -9.161 <.0001
## 6.25 - 0 -1.0958 0.1185 165 -9.246 <.0001
## 12.5 - 0 -1.1963 0.1427 165 -8.383 <.0001
##
## fly_generation = 2:
## contrast estimate SE df t.ratio p.value
## 1.5625 - 0 -0.2837 0.0673 165 -4.214 <.0001
## 3.125 - 0 -0.5781 0.0657 165 -8.794 <.0001
## 6.25 - 0 -0.7637 0.0771 165 -9.906 <.0001
## 12.5 - 0 -0.8530 0.0943 165 -9.046 <.0001
##
## fly_generation = 3:
## contrast estimate SE df t.ratio p.value
## 1.5625 - 0 -0.0635 0.0665 165 -0.956 0.3405
## 3.125 - 0 -0.2369 0.0662 165 -3.579 0.0009
## 6.25 - 0 -0.4317 0.0812 165 -5.314 <.0001
## 12.5 - 0 -0.5098 0.0931 165 -5.474 <.0001
##
## fly_generation = 4:
## contrast estimate SE df t.ratio p.value
## 1.5625 - 0 0.1566 0.1003 165 1.561 0.4816
## 3.125 - 0 0.1042 0.1012 165 1.029 0.7119
## 6.25 - 0 -0.0996 0.1265 165 -0.787 0.7119
## 12.5 - 0 -0.1665 0.1404 165 -1.186 0.7119
##
## P value adjustment: holm method for 4 tests

### final plot
(siffig=ggplot(results_qpcr_wolb_recovery, aes(x=factor(fly_generation), y=fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.3, size=0.5)+
  facet_grid(replicate~tetracycline)+
  scale_y_continuous(name="", limits=c(0,NA), breaks=seq(0, 2,0.5))+
  scale_x_discrete(name="")+
  theme_figs())

```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S1 Fig.pdf", plot=s1fig, device="pdf", dpi=300, units="in", width=650/96,
}

```

## S2 Fig | Effect EMS females fecundity and Wolbachia titers

### S2A-B Fig | Effect on fecundity

```
### S2 Data
s2data=read.csv("../original_data/S2 Data.csv")

### number of eggs and adults per EMS treatment per individual fly (replicate=fly in table)
eff_ems_fecundity=dplyr::group_by(s2data, ems_doses, replicate)%>%
  dplyr::summarise(sum_number_eggs=sum(number_eggs, na.rm=TRUE),
                  sum_number_adults=sum(number_adults, na.rm=TRUE)) %>%
  ungroup()

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

# plot :: number of eggs
ggplot(eff_ems_fecundity, aes(x=ems_doses, y=sum_number_eggs, group=ems_doses))+
  geom_boxplot(outlier.shape=NA) +
  geom_jitter(width=0.25, height=0)+
  scale_y_continuous(name="",

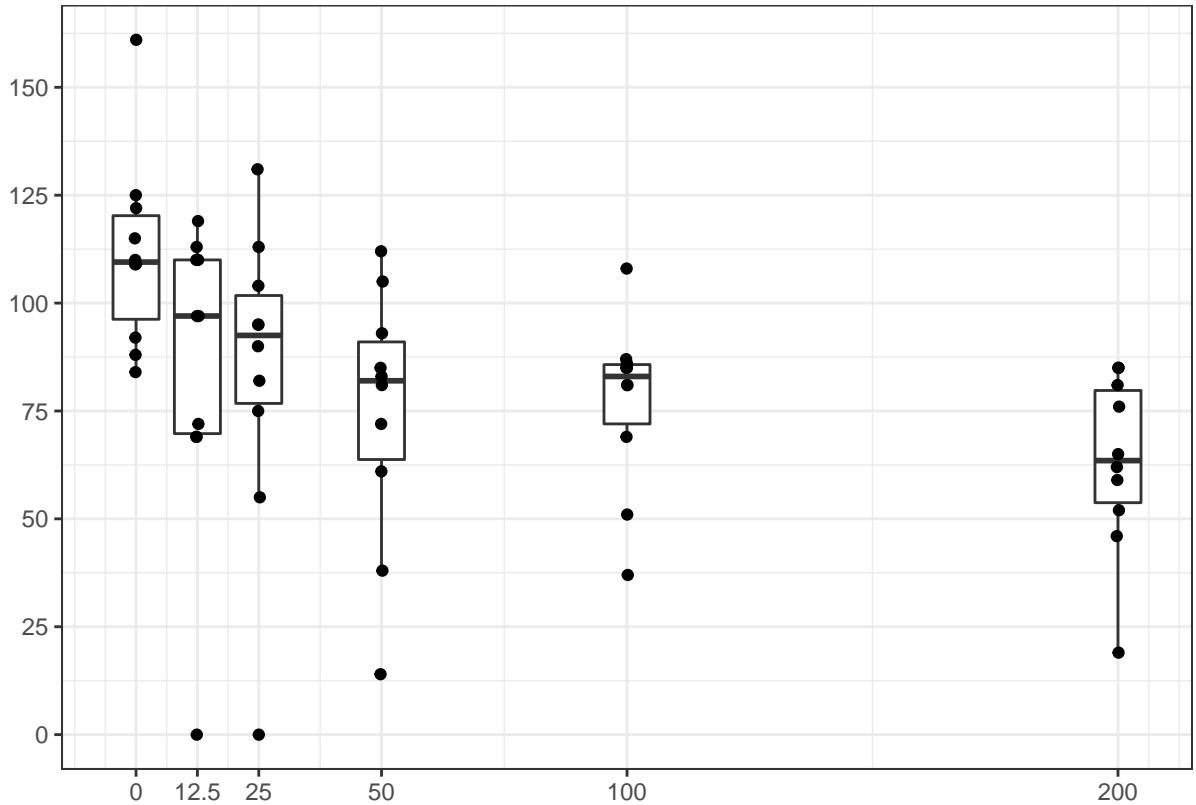
```



```

breaks=seq(0,200,25))+
scale_x_continuous(name="", breaks=c(0,12.5,25,50,100,200),
labels=c(0,12.5,25,50,100,200))+
theme_bw()

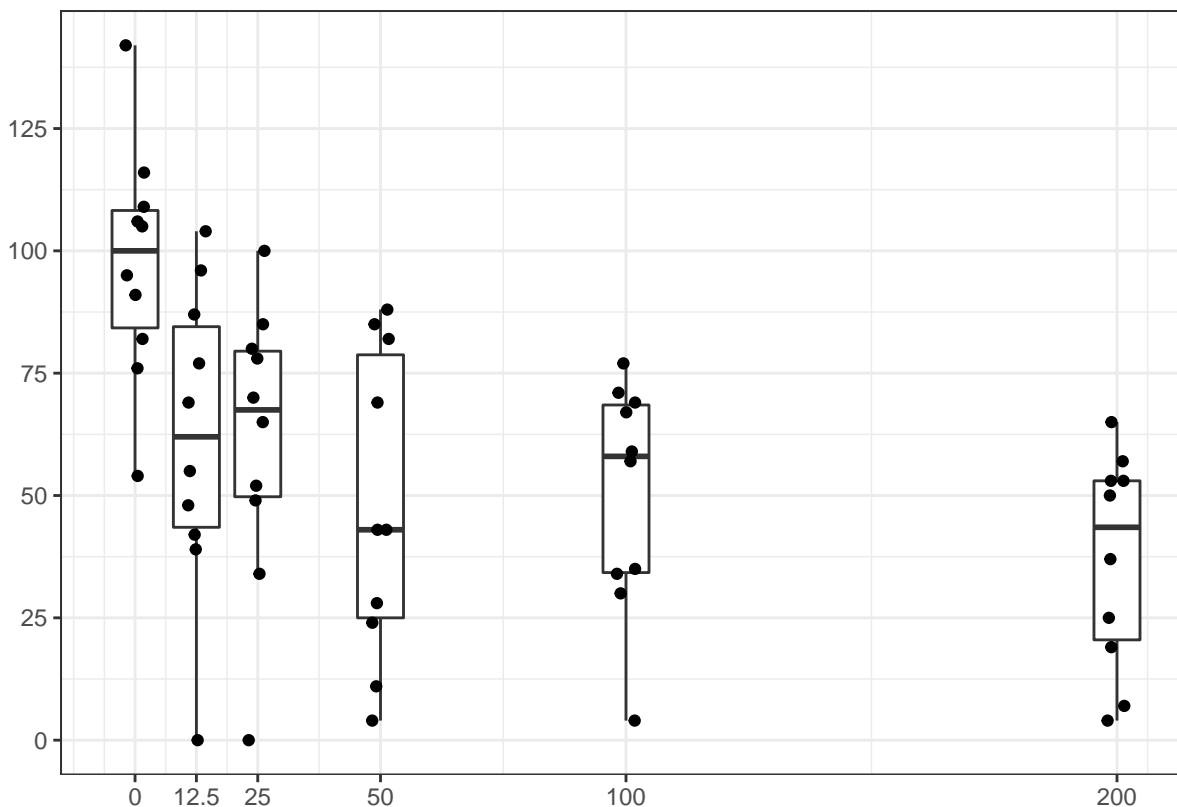
```



```

# plot :: number of adults
ggplot(eff_ems_fecundity, aes(x=ems_doses, y=sum_number_adults, group=ems_doses))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=2, height=0)+
  scale_y_continuous(name="",
breaks=seq(0,150,25))+
scale_x_continuous(name="", breaks=c(0,12.5,25,50,100,200),
labels=c(0,12.5,25,50,100,200))+
theme_bw()

```



```
## Anova Table (Type II tests)
##
## Response: sum_number_eggs
##               Sum Sq Df F value    Pr(>F)
## log(ems_doses + 1) 12729   1  16.909 0.0001251 ***
## Residuals          43663  58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

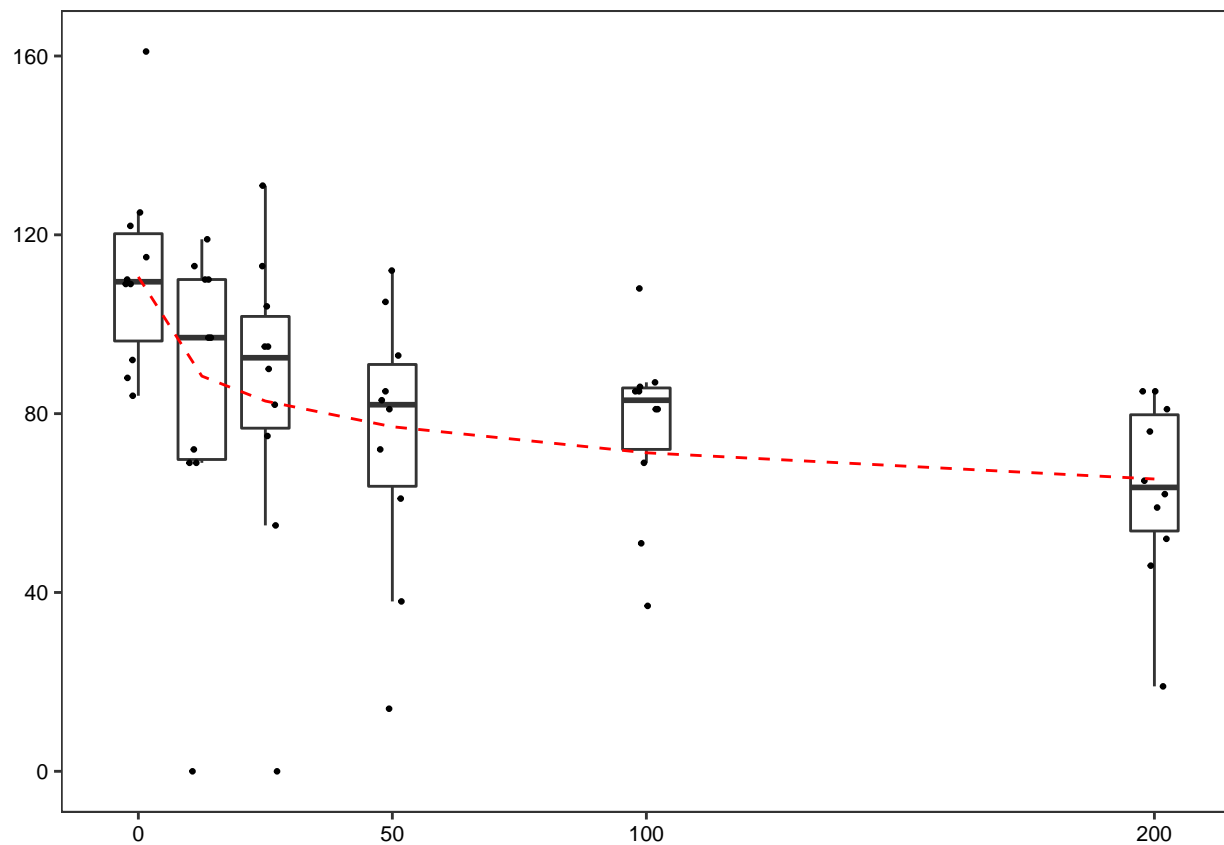
##
## Call:
## lm(formula = sum_number_eggs ~ log(ems_doses + 1), data = eff_ems_fecundity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -88.401 -16.154   6.557  15.653  50.413
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    110.587     7.676  14.406 < 2e-16 ***
## log(ems_doses + 1)  -8.524     2.073  -4.112 0.000125 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.44 on 58 degrees of freedom
## Multiple R-squared:  0.2257, Adjusted R-squared:  0.2124
## F-statistic: 16.91 on 1 and 58 DF, p-value: 0.0001251
```

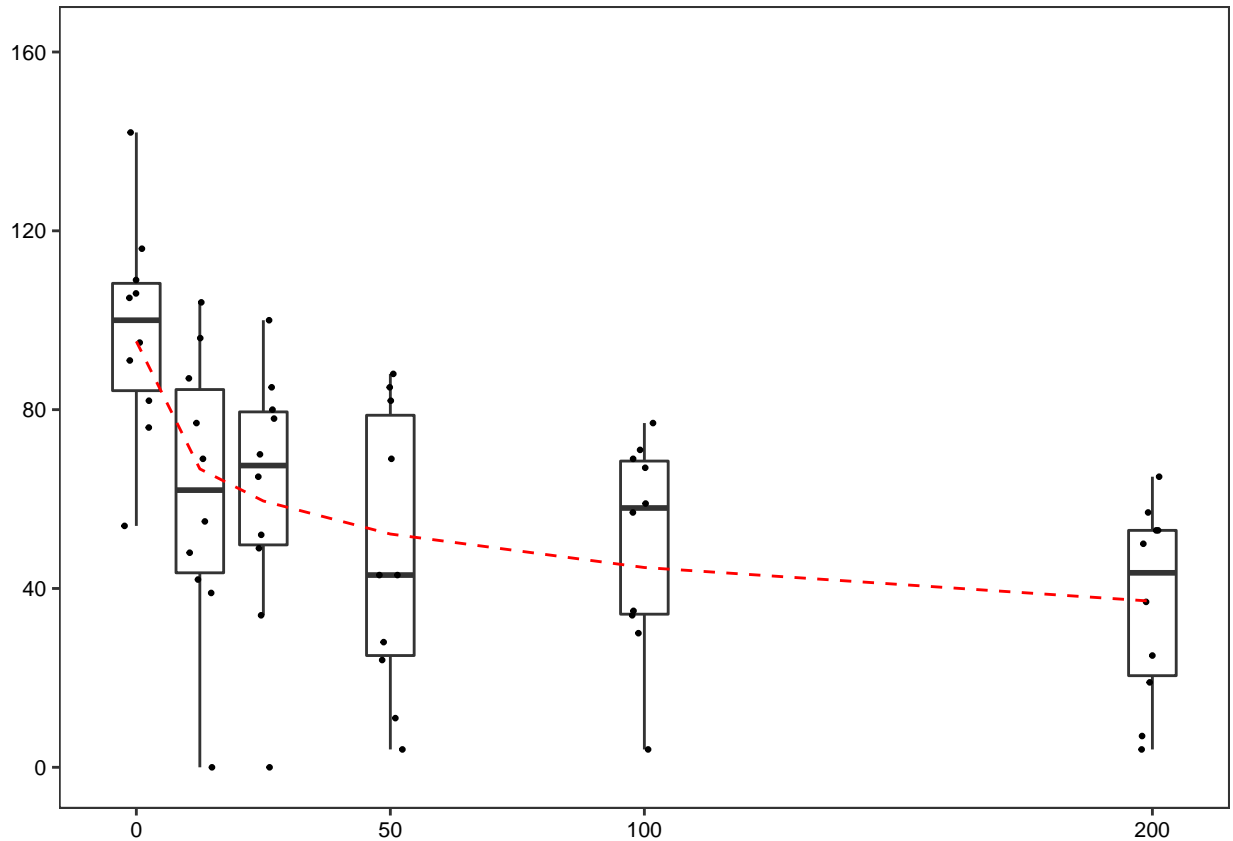
```

## Anova Table (Type II tests)
##
## Response: sum_number_adults
##           Sum Sq Df F value    Pr(>F)
## log(ems_doses + 1) 21040  1 30.058 9.546e-07 ***
## Residuals          40599 58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Call:
## lm(formula = sum_number_adults ~ log(ems_doses + 1), data = eff_ems_fecundity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -66.746 -18.299   3.845  20.300  46.731
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      95.269      7.402  12.871 < 2e-16 ***
## log(ems_doses + 1) -10.959      1.999  -5.483 9.55e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.46 on 58 degrees of freedom
## Multiple R-squared:  0.3413, Adjusted R-squared:  0.33
## F-statistic: 30.06 on 1 and 58 DF,  p-value: 9.546e-07

```





S2C Fig | Effect on Wolbachia titers (experiment 1)

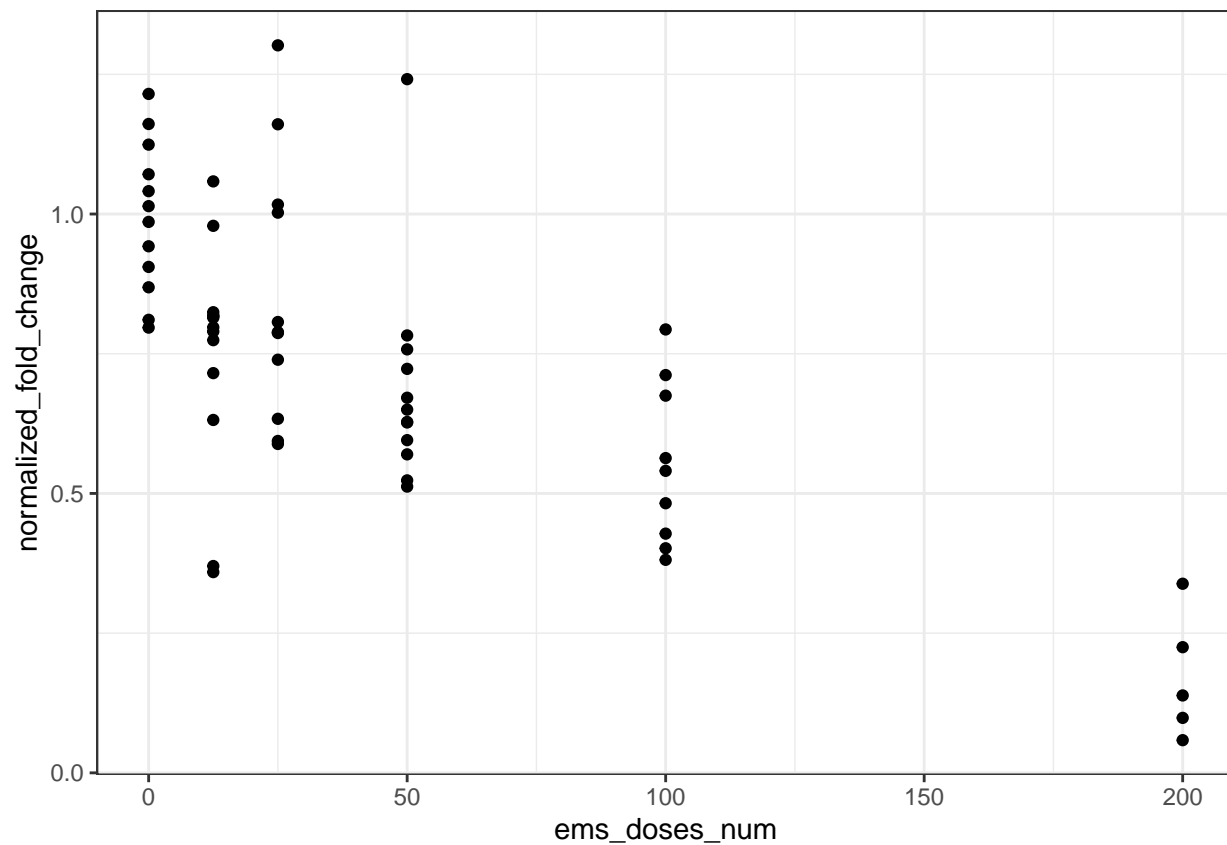
```
### S3 Data
s3data=read.csv("../original_data/S3 Data.csv", stringsAsFactors = TRUE)

### determining relative wolbachia titers
results_qpcr_ems_wolb=qpcr.pfaffl(data=s3data, calibrator_sample="E_0", reference_gene="RPL32", efficien

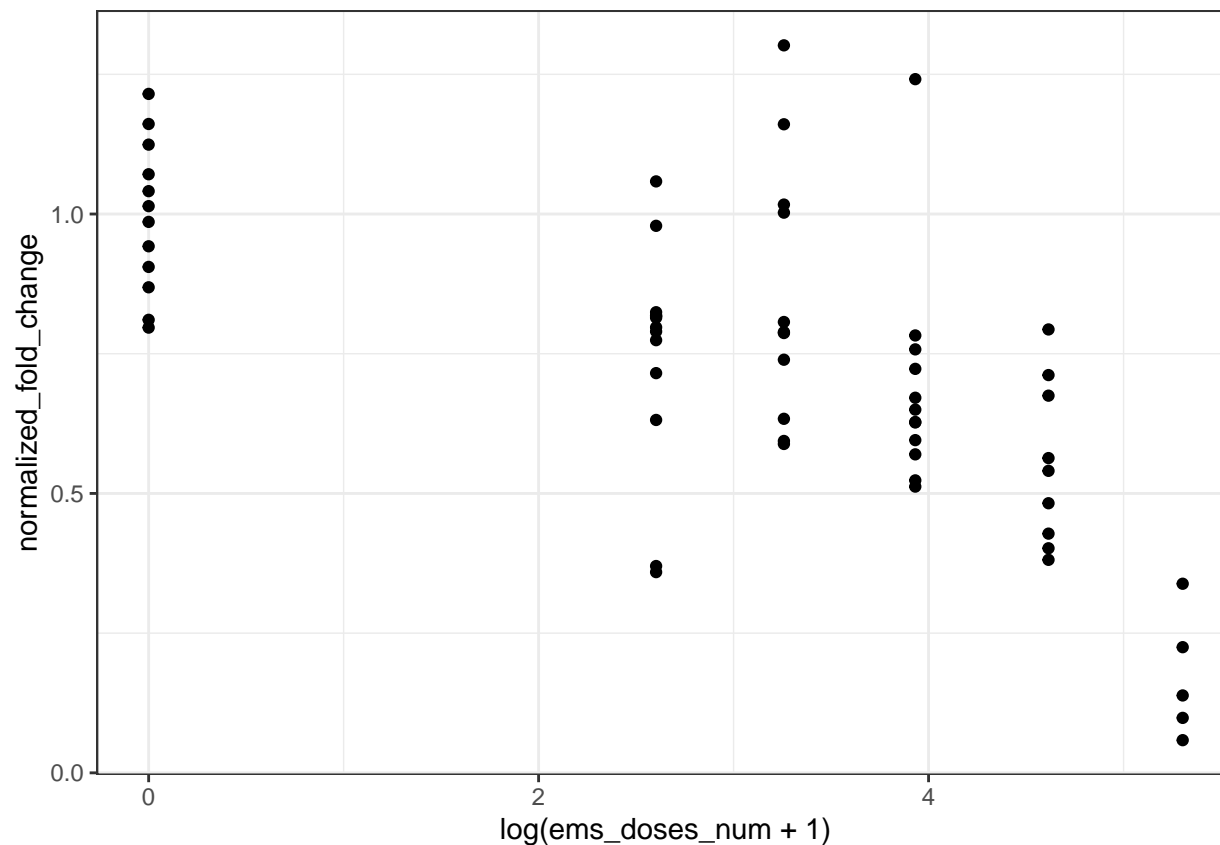
### adding new variables
results_qpcr_ems_wolb=results_qpcr_ems_wolb %>%
  tidyr::separate(sample, c(NA, "ems_doses", "replicate") , sep = "_", remove = FALSE)

# adding a numeric variable for EMS doses
results_qpcr_ems_wolb$ems_doses_num=as.numeric(as.character(results_qpcr_ems_wolb$ems_doses))

### plot
ggplot(results_qpcr_ems_wolb, aes(x=ems_doses_num, y=normalized_fold_change))+
  geom_point()+
  theme_bw()
```



```
ggplot(results_qpcr_ems_wolb, aes(x=log(ems_doses_num+1), y=normalized_fold_change))+  
  geom_point()+  
  theme_bw()
```



```
### statistical analysis
## Wolbachia loads
model_eff_ems_wolb=nls(normalized_fold_change ~ exp(-b*ems_doses_num) + c, data=results_qpcr_ems_wolb,

# summary of the model
summary(model_eff_ems_wolb)

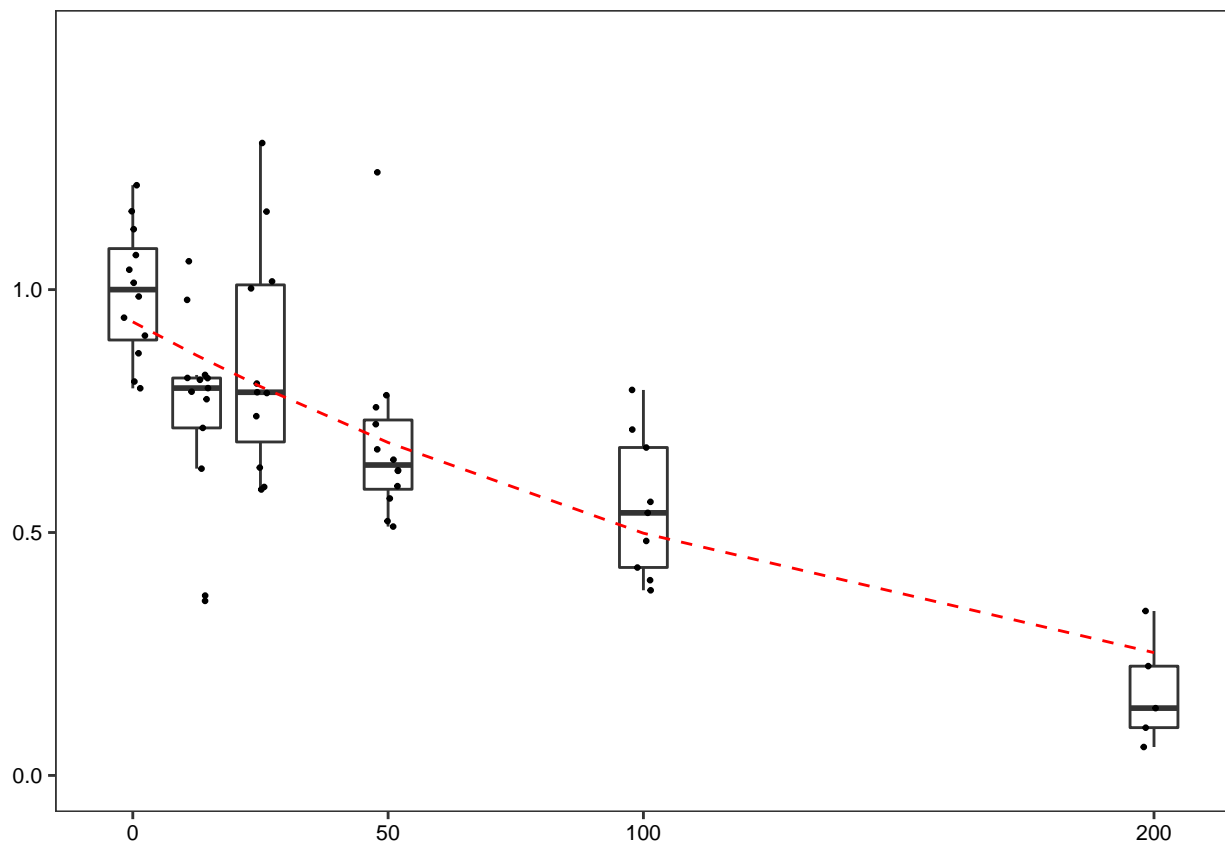
##
## Formula: normalized_fold_change ~ exp(-b * ems_doses_num) + c
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## b  0.005709   0.001137   5.020 4.91e-06 ***
## c -0.066462   0.039020  -1.703  0.0937 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1906 on 60 degrees of freedom
##
## Number of iterations to convergence: 5
## Achieved convergence tolerance: 1.031e-06

### Predicting data using the models
## Wolbachia loads
eff_ems_wolb_titers_pred <- data.frame(prediction=predict(model_eff_ems_wolb, newdata=results_qpcr_ems_wolb,
```

```

### plot
## Relative Wolbachia titer
(s2cfig=ggplot(results_qpcr_ems_wolb, aes(x=ems_doses_num, y=normalized_fold_change))+
  geom_boxplot(aes(group=ems_doses), outlier.shape=NA)+
  geom_jitter(width=2.5, size=0.5)+
  stat_summary(data=eff_ems_wolb_titers_pred, aes(x=ems_doses, y=prediction),
    fun="mean", colour="red", geom="line", linetype="dashed")+
  scale_x_continuous(name="", breaks=c(0,50,100,200),
    labels=c(0,50,100,200))+
  scale_y_continuous(name="", limits=c(0,1.5), breaks=seq(0,1.25,0.5))+
  theme_figs())

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S2C Fig.pdf", plot=s2cfig, device="pdf", dpi=300, units="in", width=220/96, height=150/96)
}

```

## S2D Fig | Effet on Wolbachia titers (experiment 2)

```

### S4 Data
s4data=read.csv("../original_data/S4 Data.csv", stringsAsFactors = TRUE)

### determining relative wolbachia titers
results_qpcr_high_ems_wolb=qpcr.pfaffl(data=s4data, calibrator_sample="E_0", reference_gene="RPL32", ef

```



```

### adding new variables
results_qpcr_high_ems_wolb=results_qpcr_high_ems_wolb%>%
  tidyr::separate(sample, c(NA, "ems_doses", "replicate") , sep = "_", remove = FALSE)

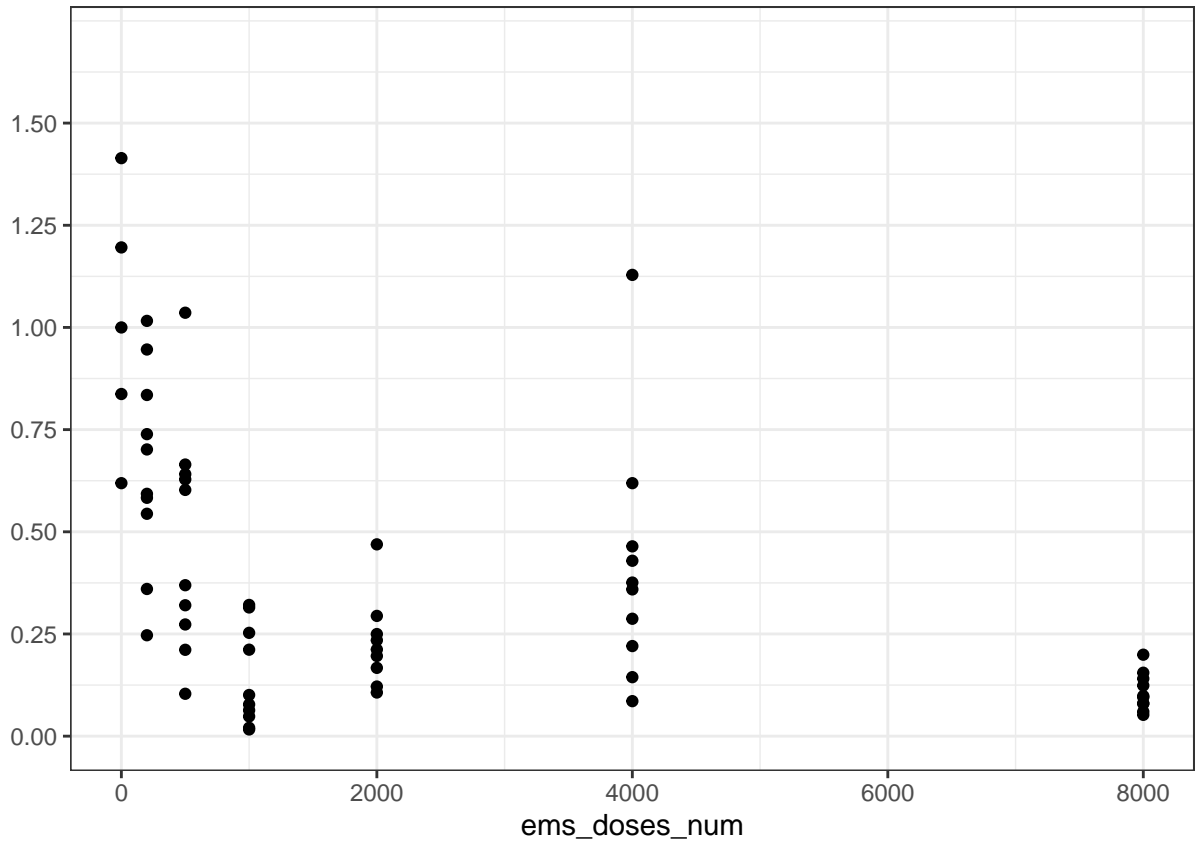
# adding a numeric variable for EMS doses
results_qpcr_high_ems_wolb$ems_doses_num=as.numeric(as.character(results_qpcr_high_ems_wolb$ems_doses))

#summarize data
results_qpcr_high_ems_wolb %>%
  group_by(ems_doses_num) %>%
  dplyr::summarise(mean = mean(normalized_fold_change))

## # A tibble: 7 x 2
##   ems_doses_num mean
## *           <dbl> <dbl>
## 1             0 1.01
## 2           200 0.656
## 3           500 0.485
## 4          1000 0.143
## 5          2000 0.228
## 6          4000 0.411
## 7          8000 0.108

### plot
ggplot(results_qpcr_high_ems_wolb, aes(x=ems_doses_num, y=normalized_fold_change))+
  geom_point()+
  scale_y_continuous(name="", limits=c(0,1.7), breaks=seq(0, 1.5,0.25))+
  theme_bw()

```



```

### statistical analysis
## Wolbachia loads
model_eff_high_ems_wolb=nls(normalized_fold_change ~ exp(-b*ems_doses_num) + c, data=results_qpcr_high_)

# summary of the model
summary(model_eff_high_ems_wolb)

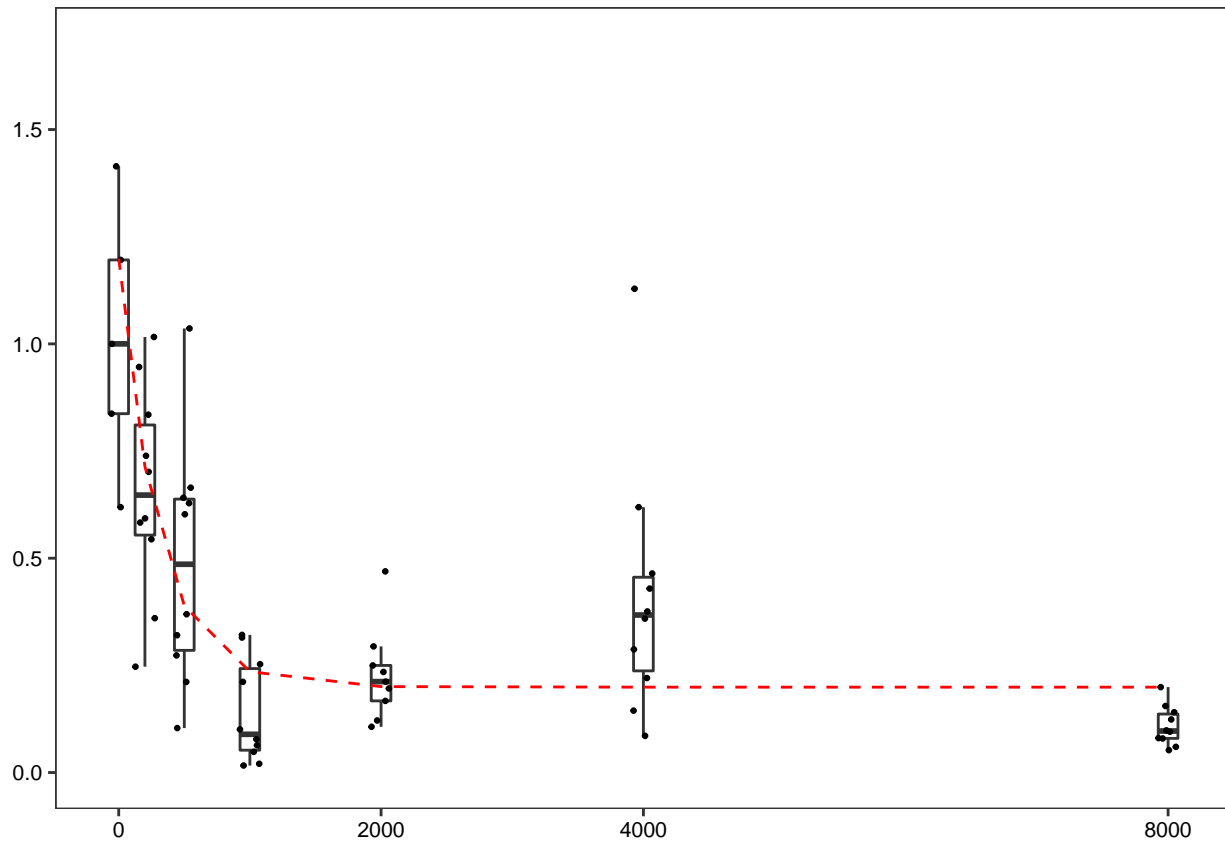
##
## Formula: normalized_fold_change ~ exp(-b * ems_doses_num) + c
##
## Parameters:
##   Estimate Std. Error t value Pr(>|t|)
## b 0.0033249  0.0006881   4.832 9.24e-06 ***
## c 0.1991796  0.0392948   5.069 3.87e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.24 on 62 degrees of freedom
##
## Number of iterations to convergence: 5
## Achieved convergence tolerance: 1.088e-06

### Predicting data using the models
# Wolbachia loads
eff_high_ems_wolb_titers_pred <- data.frame(prediction=predict(model_eff_high_ems_wolb, newdata=results_

### plot

```

```
## Relative Wolbachia titre
(s2dfig=ggplot(results_qpcr_high_ems_wolb, aes(x=ems_doses_num, y=normalized_fold_change))+
  geom_boxplot(aes(group=ems_doses_num), outlier.shape=NA)+
  geom_jitter(size=0.5) +
  stat_summary(data=eff_high_ems_wolb_titers_pred, aes(x=ems_doses, y=prediction),
    fun="mean", colour="red", geom="line", linetype="dashed")+
  scale_x_continuous(name="", breaks=c(0, 2000, 4000, 8000),
    labels=c(0, 2000, 4000, 8000))+
  scale_y_continuous(name="", limits=c(0,1.7), breaks=seq(0, 1.5,0.5))+
  theme_figs())
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S2D Fig.pdf", plot=s2dfig, device="pdf", dpi=300, units="in", width=220/96)
}
```

**Fig 1 | Isolation over-proliferative Wolbachia variants**

**Fig 1A | Isolation Line 1A :: wMelPop2**

```
### S5 Data
s5data=read.csv("../original_data/S5 Data.csv", sep=";", na.strings=c("", "Undetermined"), stringsAsFactors=FALSE)

### determining the relative Wolbachia titres
```

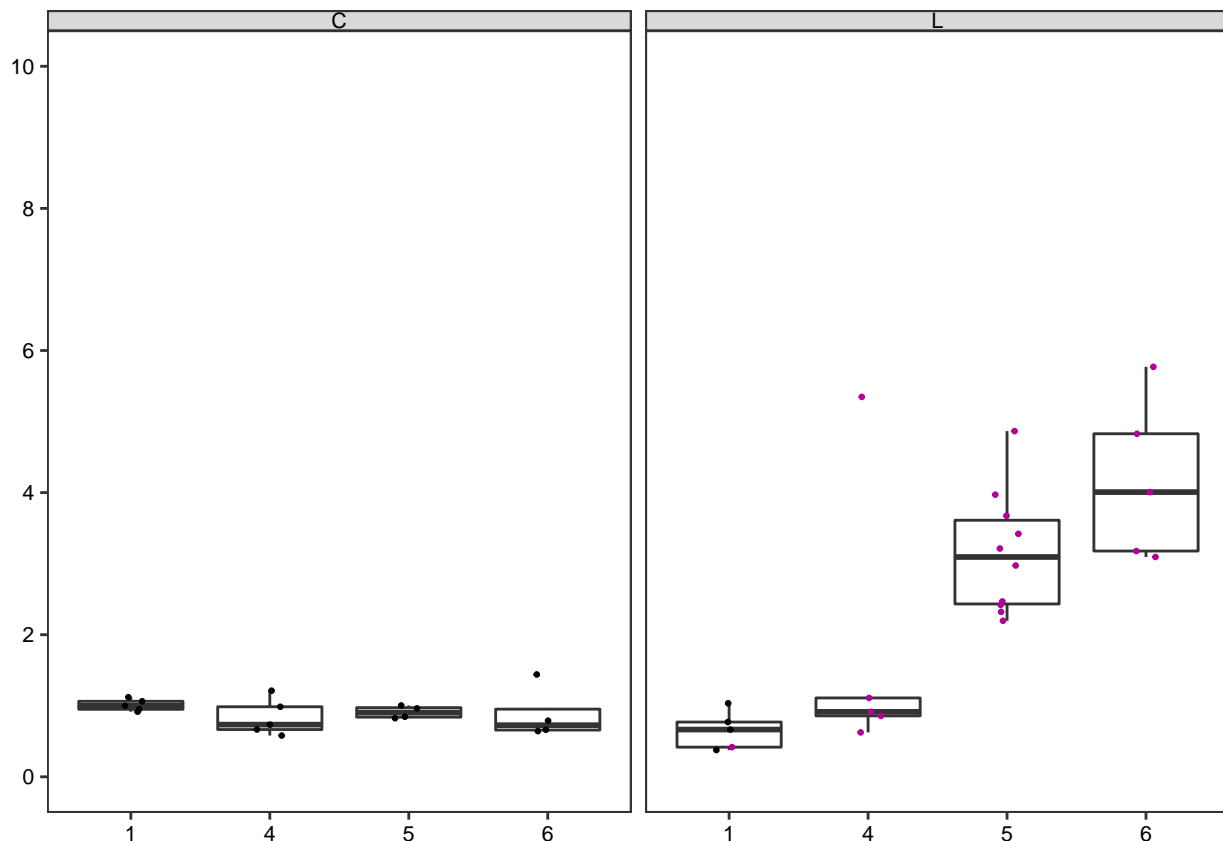
```

results_isolation_line1A=group_by(s5data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_

### adding new variables
results_isolation_line1A=results_isolation_line1A%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),
    sep="_", remove=FALSE)%>%
  rowwise()%>%
  dplyr::mutate(overproliferative="L_D" %in% paste0(line, "_", female))%>%
  dplyr::filter(keep_samples=="Y")

### plot
(fig1a=ggplot(results_isolation_line1A, aes(x=factor(fly_generations), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
  facet_grid(.~line)+
  scale_y_continuous(name="", limits=c(0, 10), breaks=seq(0, 10, 2))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#b30098"))+
  theme_figs()+ theme(legend.position="none"))

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 1A.pdf", plot=fig1a, device="pdf", dpi=300, units="in", width=260/96,

```

```
}
```

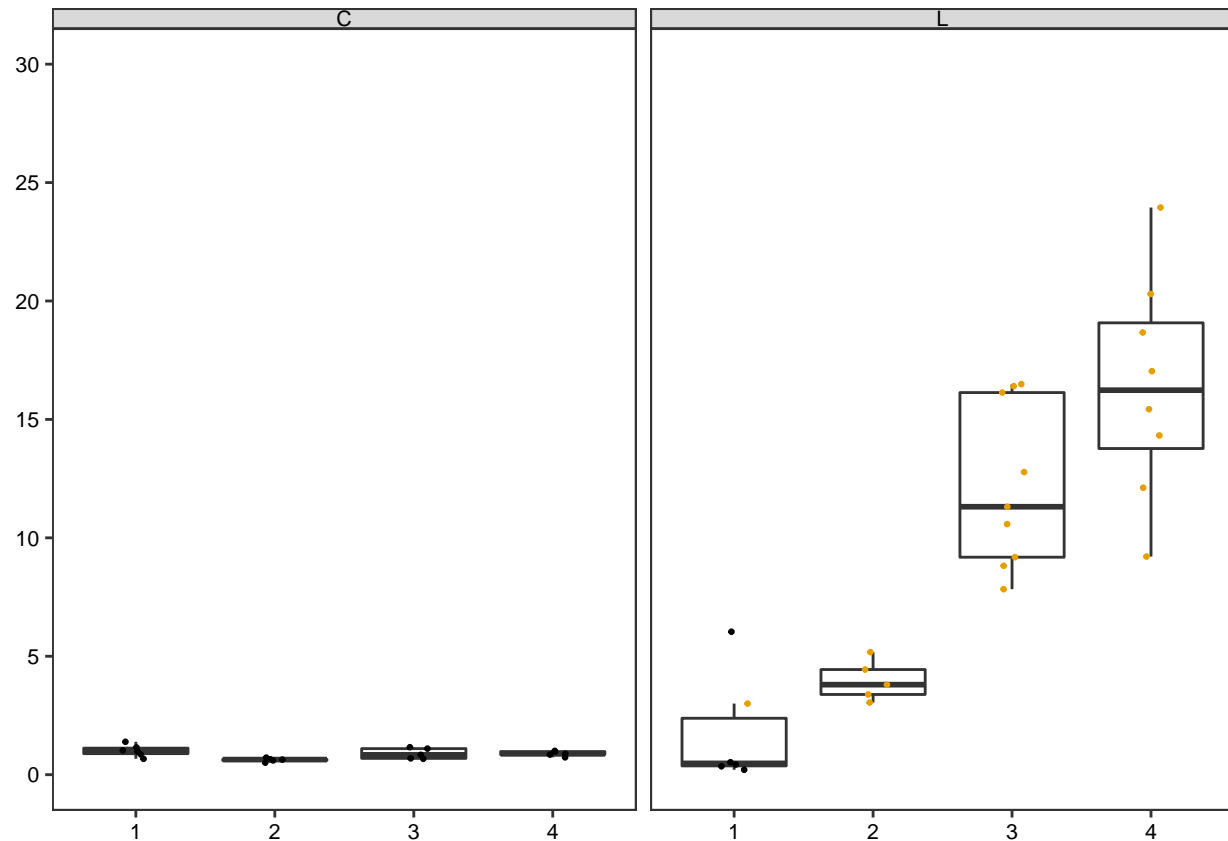
## Fig 1B | Isolation Line 2A :: wMelOctoless

```
### S6 Data
s6data=read.csv("../original_data/S6 Data.csv", sep=" ", na.strings=c(" ", "Undetermined"), stringsAsFactors=FALSE)

### determining the relative Wolbachia titres
results_isolation_line2a=group_by(s6data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_cal=1))

### adding new variables
results_isolation_line2a=results_isolation_line2a%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),
    sep="_", remove=FALSE)%>%
  rowwise()%>%
  dplyr::mutate(overproliferative="L_B" %in% paste0(line, "_", female))%>%
  dplyr::filter(keep_samples=="Y")

### plot
(fig2a=ggplot(results_isolation_line2a, aes(x=factor(fly_generations), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
  facet_grid(.~line)+
  scale_y_continuous(name="", limits=c(0, 30), breaks=seq(0, 30, 5))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#e69f00"))+
  theme_figs()+ theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 2A.pdf", plot=fig2a, device="pdf", dpi=300, units="in", width=260/96,
}

```

Fig 1C | Wolbachia titres isogenic genetic background

```
### S7 Data
s7data=read.csv("../original_data/S7 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors = TRUE)

### determining relative wolbachia titres
results_qpcr_titers_iso=qpcr.pfaffl(data=s7data, calibrator_sample="Control", reference_gene="RPL32", e=2)

### adding new variables
results_qpcr_titers_iso=results_qpcr_titers_iso%>%
  tidyr::separate(sample, c("line", "female", "replicate"), sep="_", remove=FALSE)

### the statistics
## the model
model_titers_iso=lm(normalized_fold_change~line, data=results_qpcr_titers_iso)

# significance of the model
car::Anova(model_titers_iso)

```

```

## Anova Table (Type II tests)
##
## Response: normalized_fold_change
##           Sum Sq Df F value    Pr(>F)
## line       745.56  2  32.267 2.868e-07 ***
## Residuals  254.16 22
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_titers_iso)

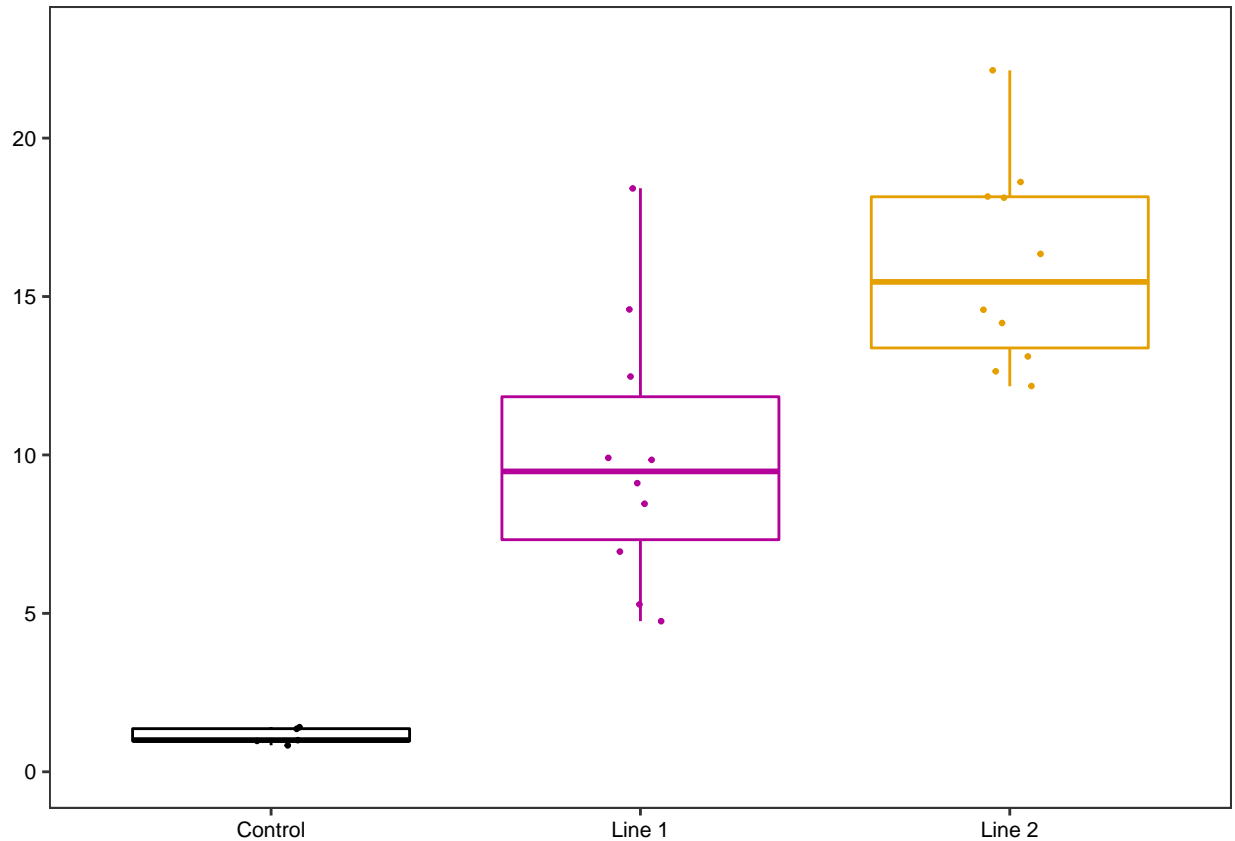
##
## Call:
## lm(formula = normalized_fold_change ~ line, data = results_qpcr_titers_iso)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.2253 -1.8367 -0.1279  2.1158  8.4370
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.116      1.520   0.735    0.47
## lineLine 1     8.864      1.862   4.761 9.41e-05 ***
## lineLine 2    14.888      1.862   7.997 5.94e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.399 on 22 degrees of freedom
## Multiple R-squared:  0.7458, Adjusted R-squared:  0.7227
## F-statistic: 32.27 on 2 and 22 DF,  p-value: 2.868e-07

## multiple comparison
model_titers_iso_multcomp=emmeans::emmeans(model_titers_iso, ~ line)
#
contrast(model_titers_iso_multcomp, "trt.vs.ctrl", adj="holm")

## contrast      estimate    SE df t.ratio p.value
## Line 1 - Control    8.86 1.86 22  4.761  0.0001
## Line 2 - Control   14.89 1.86 22  7.997  <.0001
##
## P value adjustment: holm method for 2 tests

### plot
(fig1c=ggplot(results_qpcr_titers_iso, aes(x=line, y=normalized_fold_change, color=line))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.1, size=0.5)+
  scale_y_continuous(name="", limits=c(0, 23), breaks=seq(0, 20, 5))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#b30098", "#e69f00"))+
  theme_figs()+theme(legend.position="none"))

```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 1C.pdf", plot=fig1c, device="pdf", dpi=300, units="in", width=160/96,
}
}
```

### S3A Fig | Isolation Line 2B

```
### S8 Data
s8data=read.csv("../original_data/S8 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors=TRUE)

### determining the relative Wolbachia titers
results_isolation_line2b=group_by(s8data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_

### adding new variables
results_isolation_line2b=results_isolation_line2b%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),
    sep="_", remove=FALSE)%>%
  rowwise()%>%
  dplyr::mutate(overproliferative="L_A" %in% paste0(line, "_", female))%>%
  dplyr::filter(keep_samples=="Y")

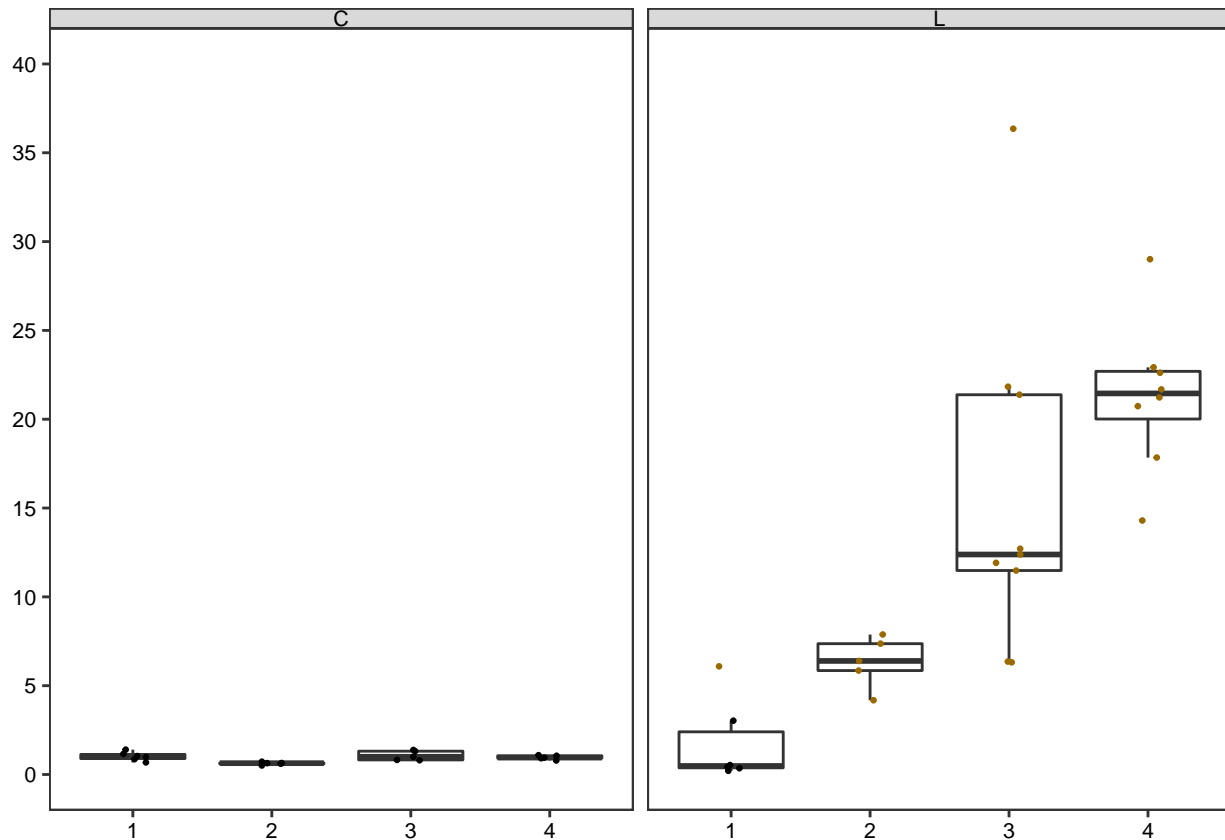
### plot
(s3afig=ggplot(results_isolation_line2b, aes(x=factor(fly_generations), y=fold_change))+
```



```

geom_boxplot(outlier.shape=NA)+
geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
facet_grid(.~line)+
scale_y_continuous(name="", limits=c(0, 40), breaks=seq(0, 40, 5))+
scale_x_discrete(name="")+
scale_color_manual(values=c("#000000", "#9a6a00"))+
theme_figs()+ theme(legend.position="none")

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S3A Fig.pdf", plot=s3afig, device="pdf", dpi=300, units="in", width=260/96, height=180/96)
}

```

## S3B Fig | Isolation Line 3A :: wMelOctoless2

```

### S9 Data
s9data=read.csv("../original_data/S9 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors=TRUE)

### determining the relative Wolbachia titers
results_isolation_line3a=group_by(s9data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_cal=1))

### adding new variables
results_isolation_line3a=results_isolation_line3a%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),

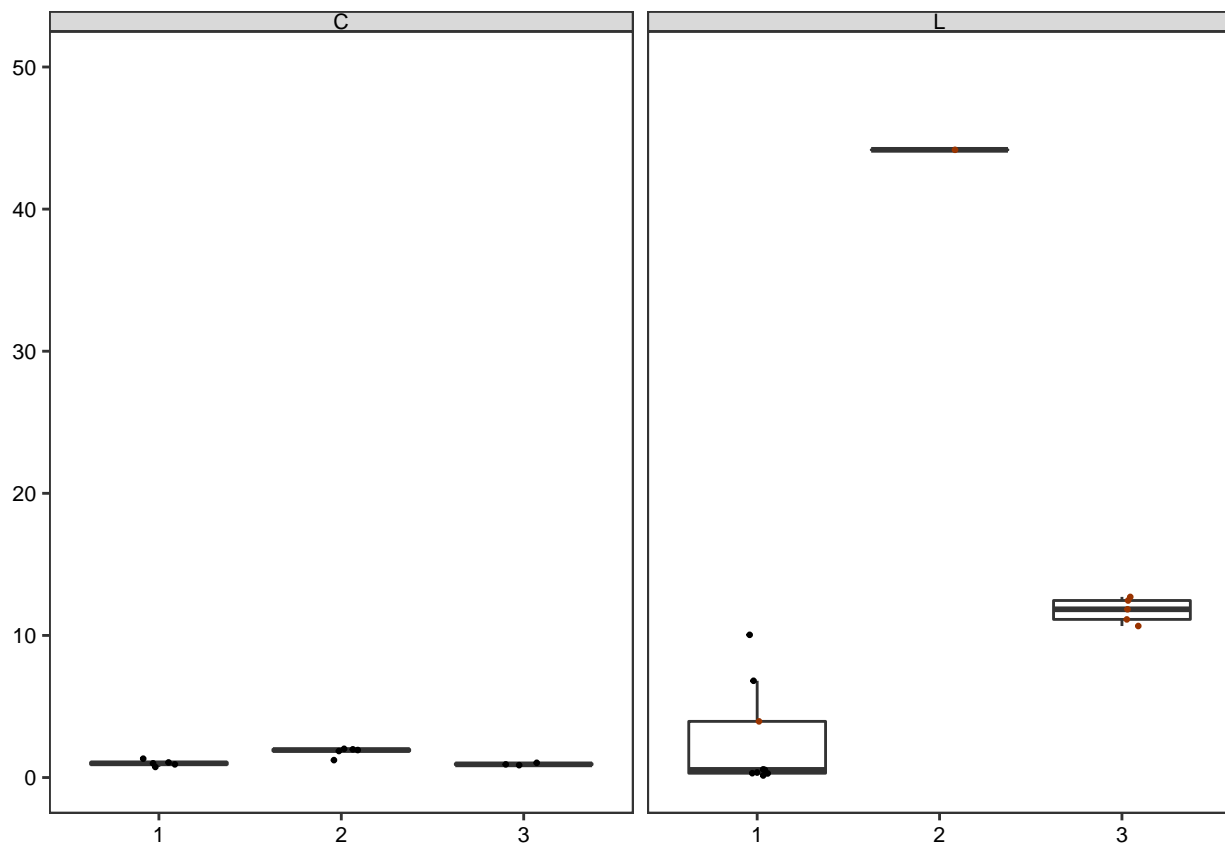
```

```

        sep="_", remove=FALSE)%>%
rowwise()%>%
dplyr::mutate(overproliferative="L_G" %in% paste0(line, "_", female))%>%
dplyr::filter(keep_samples=="Y")

### plot
(s3bfig=ggplot(results_isolation_line3a, aes(x=factor(fly_generations), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
  facet_grid(.~line)+
  scale_y_continuous(name="", limits=c(0, 50), breaks=seq(0, 50, 10))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#993300"))+
  theme_figs()+ theme(legend.position="none"))

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S3B Fig.pdf", plot=s3bfig, device="pdf", dpi=300, units="in", width=240/96, height=160/96)
}

```

## S3C Fig | Isolation Line 3B

```

### S10 Data
s10data=read.csv("../original_data/S10 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors=TRUE)

```

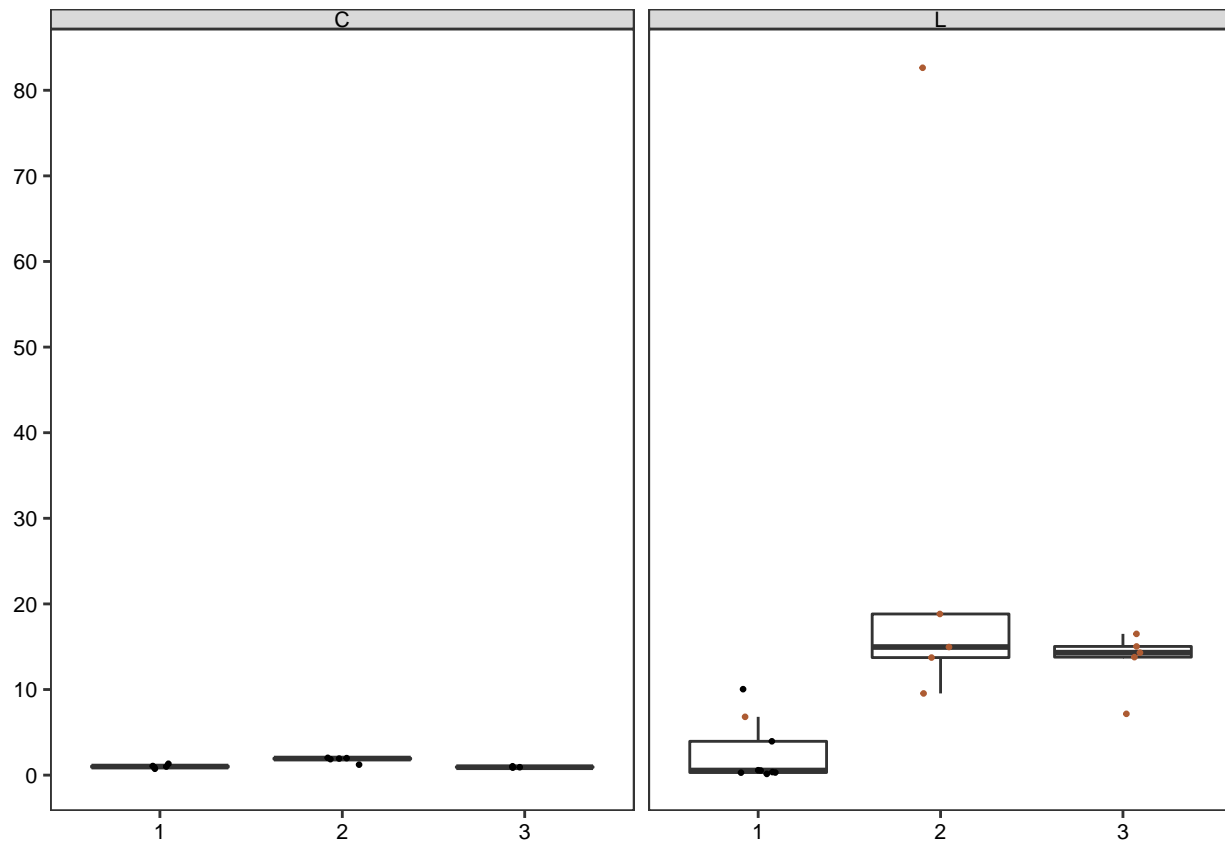
```

### determining the relative Wolbachia titres
results_isolation_line3b=group_by(s10data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_

### adding new variables
results_isolation_line3b=results_isolation_line3b%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),
    sep="_", remove=FALSE)%>%
  rowwise()%>%
  dplyr::mutate(overproliferative="L_A" %in% paste0(line, "_", female))%>%
  dplyr::filter(keep_samples=="Y")

### plot
(s3cfig=ggplot(results_isolation_line3b, aes(x=factor(fly_generations), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
  facet_grid(.~line)+
  scale_y_continuous(name="", limits=c(0, 83), breaks=seq(0, 80, 10))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#ad5b32"))+
  theme_figs()+ theme(legend.position="none"))

```



```

# saving the pdf
if(save_files==TRUE)
{

```

```

  ggsave(filename="../results/S3C Fig.pdf", plot=s3cfig, device="pdf", dpi=300, units="in", width=240/96, height=160/96)
}

```

## S3D Fig | Isolation Line 3C

```

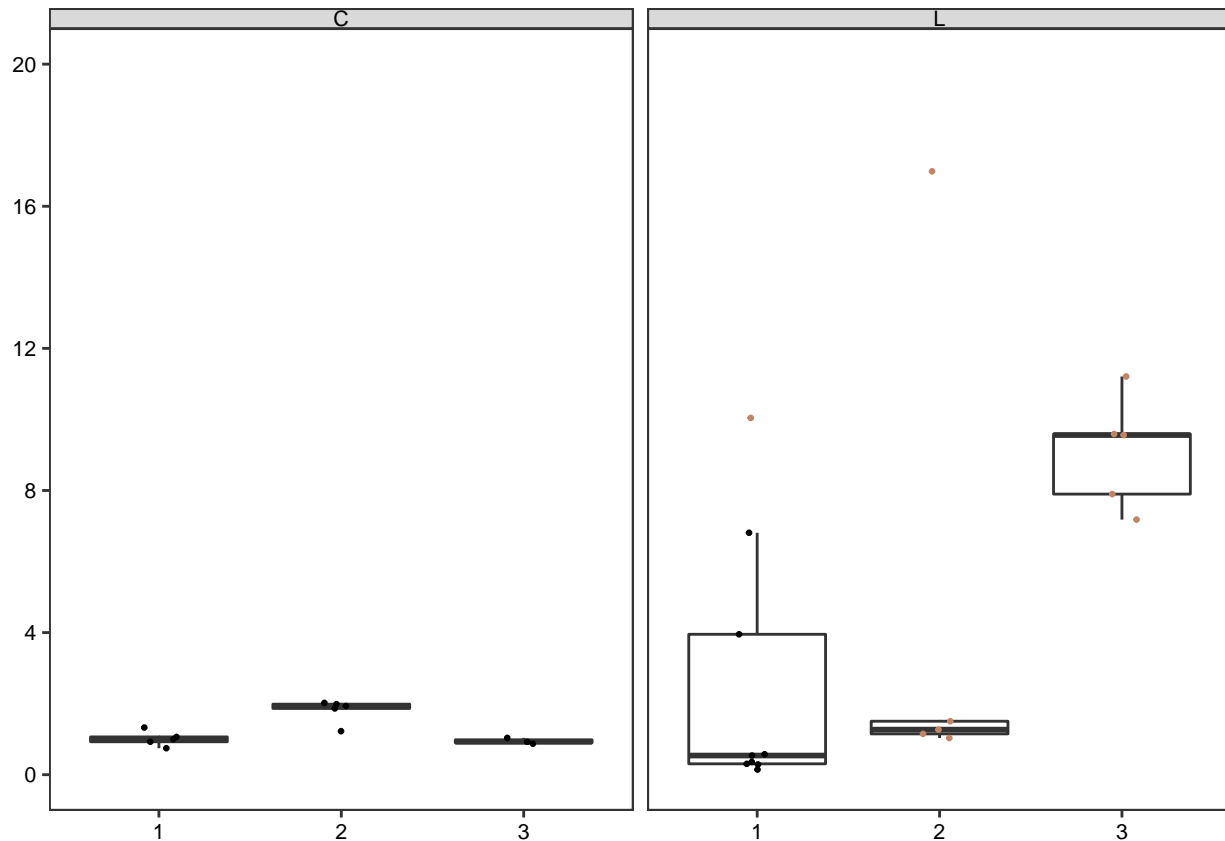
### S11 Data
s11data=read.csv("../original_data/S11 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors=TRUE)

### determining the relative Wolbachia titers
results_isolation_line3c=group_by(s11data, fly_generations)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="C_Y", reference_gene="RPL32", efficiency_ref=2, efficiency_cal=1))

### adding new variables
results_isolation_line3c=results_isolation_line3c%>%
  tidyr::separate(sample, c("line", "reference", "keep_samples", "selected", "female", "replicate"),
    sep="_", remove=FALSE)%>%
  rowwise()%>%
  dplyr::mutate(overproliferative="L_I" %in% paste0(line, "_", female))%>%
  dplyr::filter(keep_samples=="Y")

### plot
(s3dfig=ggplot(results_isolation_line3c, aes(x=factor(fly_generations), y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(aes(color=overproliferative), width=0.1, size=0.5)+
  facet_grid(.~line)+
  scale_y_continuous(name="", limits=c(0, 20), breaks=seq(0, 20, 4))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#c18466"))+
  theme_figs()+ theme(legend.position="none"))

```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S3D Fig.pdf", plot=s3dfig, device="pdf", dpi=300, units="in", width=240/96, height=160/96)
}
```

## S5 Fig | Characterization of wMelOctoless2

```
### S12 Data
s12data=read.csv("../original_data/S12 Data.csv", na.strings=c("", "Undetermined"), stringsAsFactors = FALSE)

### determining relative wolbachia titers
results_qpcr_titers_octoless2=qpcr.pfafill(data=s12data, calibrator_sample="Cal-wMelCS-b", reference_genome="wMelCS-b")

### adding new variables
results_qpcr_titers_octoless2=results_qpcr_titers_octoless2%>%
  tidyr::separate(sample, c("wolbachia", "time_point", "replicate"), sep="_", remove=FALSE)%>%
  dplyr::filter(wolbachia!="Cal-wMelCS-b")%>%
  dplyr::mutate(days=as.numeric(as.character(time_point)))

### the statistics
## titre on day 0
model_titer_octoless2_day0=lm(normalized_fold_change~wolbachia, data=dplyr::filter(results_qpcr_titers_octoless2, time_point=="0"))
```

```

# significance of the model
car::Anova(model_titer_octoless2_day0)

## Anova Table (Type II tests)
##
## Response: normalized_fold_change
##           Sum Sq Df F value    Pr(>F)
## wolbachia 174.482  2  87.571 1.574e-12 ***
## Residuals  26.898 27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_titer_octoless2_day0)

##
## Call:
## lm(formula = normalized_fold_change ~ wolbachia, data = dplyr::filter(results_qpcr_titers_octoless2,
##   days == 0))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.00224 -0.11862 -0.03324  0.40791  2.80953
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.8362     0.3156   2.649   0.0133 *
## wolbachiaMel0ctoless  5.4767     0.4464  12.269 1.49e-12 ***
## wolbachiaMel0ctoless2  4.6558     0.4464  10.430 5.71e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9981 on 27 degrees of freedom
## Multiple R-squared:  0.8664, Adjusted R-squared:  0.8565
## F-statistic: 87.57 on 2 and 27 DF,  p-value: 1.574e-12

# multiple comparison
(multiple_titer_octoless2_day0=emmeans::emmeans(model_titer_octoless2_day0, pairwise~wolbachia, adj="holm"))

## $emmeans
##   wolbachia      emmean      SE df lower.CL upper.CL
## wMelCS-b         0.836 0.316 27    0.0306    1.64
## wMel0ctoless      6.313 0.316 27    5.5073    7.12
## wMel0ctoless2     5.492 0.316 27    4.6864    6.30
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
##   contrast              estimate      SE df t.ratio p.value
## wMelCS-b - wMel0ctoless    -5.477 0.446 27 -12.269 <.0001
## wMelCS-b - wMel0ctoless2   -4.656 0.446 27 -10.430 <.0001
## wMel0ctoless - wMel0ctoless2  0.821 0.446 27   1.839 0.0769
##
## P value adjustment: holm method for 3 tests

```

```

#
multcomp::cld(multiple_titer_octoless2_day0$emmeans, adj="holm")

##   wolbachia      emmean    SE df lower.CL upper.CL .group
##   wMelCS-b        0.836 0.316 27   0.0306    1.64    1
##   wMelOctoless2    5.492 0.316 27   4.6864    6.30    2
##   wMelOctoless     6.313 0.316 27   5.5073    7.12    2
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05

## proliferation
model_titers_octoless2=lm(normalized_fold_change~wolbachia*days, data=results_qpcr_titers_octoless2)

# significance of the model
car::Anova(model_titers_octoless2)

## Anova Table (Type II tests)
##
## Response: normalized_fold_change
##              Sum Sq Df F value    Pr(>F)
## wolbachia      1400.64  2 221.859 < 2.2e-16 ***
## days           840.06  1 266.127 < 2.2e-16 ***
## wolbachia:days  353.27  2  55.957 6.883e-14 ***
## Residuals      170.46 54
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_titers_octoless2)

##
## Call:
## lm(formula = normalized_fold_change ~ wolbachia * days, data = results_qpcr_titers_octoless2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.1324 -0.9921 -0.0528  0.3593  6.0376
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.83624    0.56184   1.488   0.142
## wolbachiaMelOctoless    5.47671    0.79456   6.893 6.15e-09 ***
## wolbachiaMelOctoless2    4.65579    0.79456   5.860 2.86e-07 ***
## days            0.08924    0.11351   0.786   0.435
## wolbachiaMelOctoless:days  1.49967    0.16052   9.342 7.18e-13 ***
## wolbachiaMelOctoless2:days  1.43984    0.16052   8.970 2.78e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.777 on 54 degrees of freedom
## Multiple R-squared:  0.9383, Adjusted R-squared:  0.9326
## F-statistic: 164.4 on 5 and 54 DF, p-value: < 2.2e-16

```

```

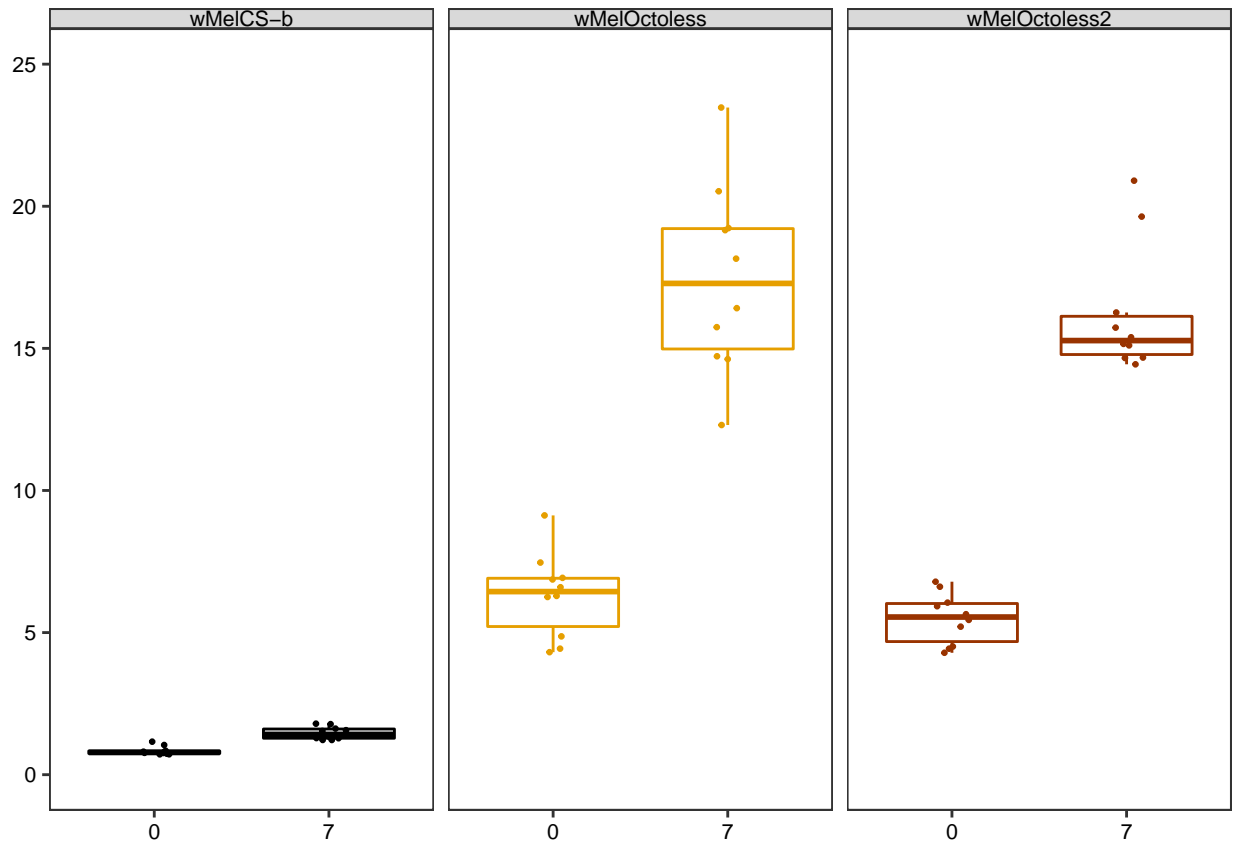
## multiple comparison
## multiple comparison
(multiple_model_titers_octoless2=emmeans::lstrends(model_titers_octoless2, pairwise~wolbachia, var="days",

## $lstrends
## wolbachia      days.trend      SE df lower.CL upper.CL
## wMelCS-b        0.0892 0.114 54   -0.191    0.37
## wMel0ctoless     1.5889 0.114 54    1.308    1.87
## wMel0ctoless2    1.5291 0.114 54    1.249    1.81
##
## Results are averaged over the levels of: days
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
## contrast              estimate      SE df t.ratio p.value
## wMelCS-b - wMel0ctoless    -1.4997 0.161 54 -9.342  <.0001
## wMelCS-b - wMel0ctoless2   -1.4398 0.161 54 -8.970  <.0001
## wMel0ctoless - wMel0ctoless2  0.0598 0.161 54  0.373  0.7108
##
## Results are averaged over the levels of: days
## P value adjustment: holm method for 3 tests
#
multcomp::cld(multiple_model_titers_octoless2$lstrends, adj="holm")

## wolbachia      days.trend      SE df lower.CL upper.CL .group
## wMelCS-b        0.0892 0.114 54   -0.191    0.37    1
## wMel0ctoless2    1.5291 0.114 54    1.249    1.81    2
## wMel0ctoless     1.5889 0.114 54    1.308    1.87    2
##
## Results are averaged over the levels of: days
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05
### plot
(s5fig=ggplot(results_qpcr_titers_octoless2, aes(x=time_point, y=normalized_fold_change, color=wolbachia))
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.1, size=0.5)+
  facet_grid(~wolbachia)+
  scale_y_continuous(name="", limits=c(0, 25), breaks=seq(0, 25, 5))+
  scale_x_discrete(name="")+
  scale_color_manual(values=c("#000000", "#e69f00", "#993300ff"))+
  theme_figs()+theme(legend.position="none"))

```





```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S5 Fig.pdf", plot=s5fig, device="pdf", dpi=300, units="in", width=450/96,
}

```

## S6 Fig | Confirmation Octomom copy number variation

```
### S13 Data
s13data=read.csv("../original_data/S13 Data.csv", na.strings=c("", "Undetermined", "NA"), stringsAsFacto

### determining copy number of individual Octomom genes
# WD0505
wd505_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0505"))%>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0507
wd507_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0507"))%>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0508
wd508_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0508"))%>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0509

```

```

wd509_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0509"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0510
wd510_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0510"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0511
wd511_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0511"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0512
wd512_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0512"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0513
wd513_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0513"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0514
wd514_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0514"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# WD0519
wd519_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="WD0519"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

# rpoD
rpoD_qpcr_results=droplevels(subset(s13data, gene=="WSP" | gene=="rpoD"))>%
  qpcr.pfaffl(data=., calibrator_sample="wMelCS-b", reference_gene="WSP", efficiency_ref=2, efficiency_

### merging the results (only sample names and fold change)
results_qpcr_copies_octomom_genes=data.frame(rbind(wd505_qpcr_results[,c(1,8,9)],
                                                    wd507_qpcr_results[,c(1,8,9)],
                                                    wd508_qpcr_results[,c(1,8,9)],
                                                    wd509_qpcr_results[,c(1,8,9)],
                                                    wd510_qpcr_results[,c(1,8,9)],
                                                    wd511_qpcr_results[,c(1,8,9)],
                                                    wd512_qpcr_results[,c(1,8,9)],
                                                    wd513_qpcr_results[,c(1,8,9)],
                                                    wd514_qpcr_results[,c(1,8,9)],
                                                    wd519_qpcr_results[,c(1,8,9)],
                                                    rpoD_qpcr_results[,c(1,8,9)]),
                                              gene=rep(c("WD0505", "WD0507", "WD0508", "WD0509", "WD0510", "WD0511", "WD0512", "WD0513", "WD0514", "WD0519", "rpoD"),
                                                    times=10))

# converting variables to factor
results_qpcr_copies_octomom_genes$gene=factor(results_qpcr_copies_octomom_genes$gene, levels=c("WD0505", "WD0507", "WD0508", "WD0509", "WD0510", "WD0511", "WD0512", "WD0513", "WD0514", "WD0519", "rpoD"))

### processing
results_qpcr_copies_octomom_genes=results_qpcr_copies_octomom_genes%>%
  tidyr::separate(sample, c("wolbachia", "replicate"), sep="_", remove=FALSE)

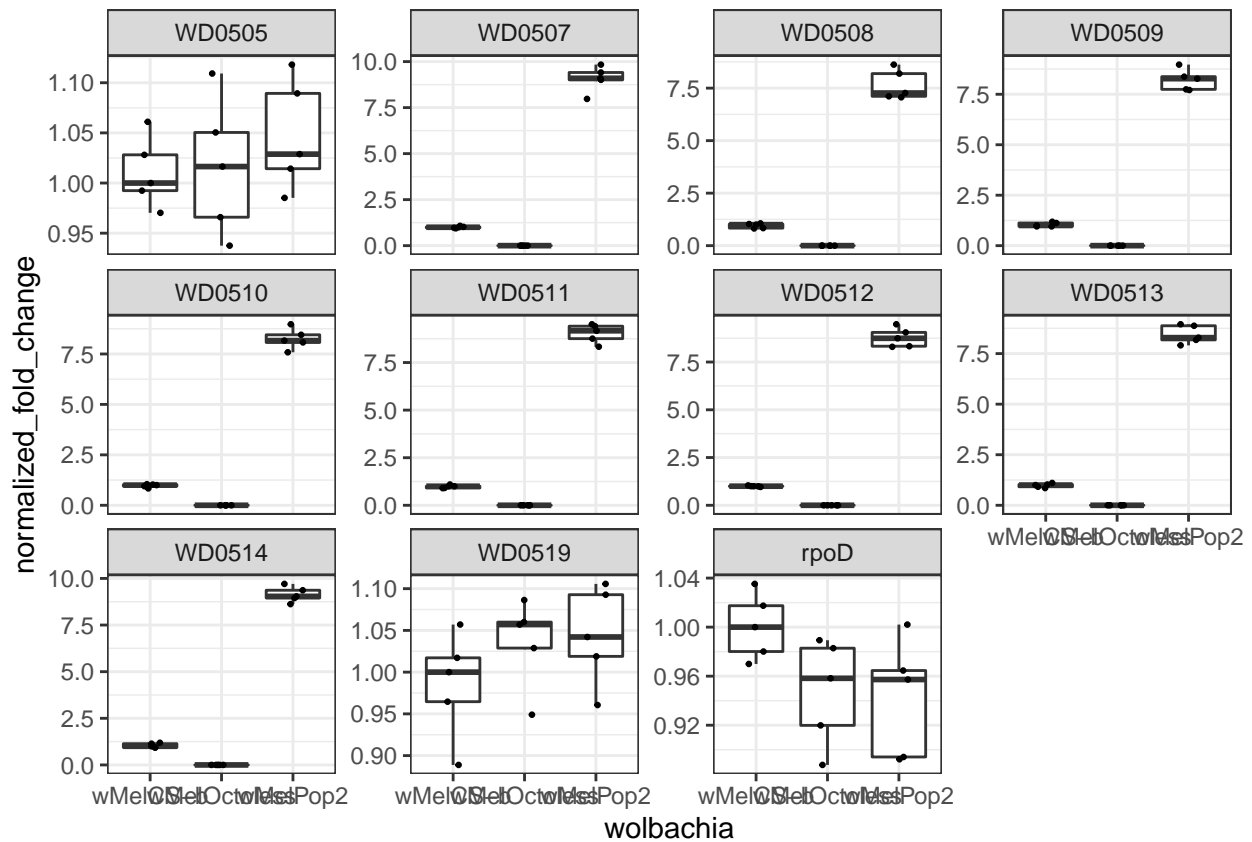
```

```

# replacing 'NaN' on wMelOctoless to '0'
results_qpcr_copies_octomom_genes$fold_change[results_qpcr_copies_octomom_genes$wolbachia=="wMelOctoless"] = 0
#
results_qpcr_copies_octomom_genes$normalized_fold_change[results_qpcr_copies_octomom_genes$wolbachia=="wMelOctoless"] = 0

### plot
ggplot(results_qpcr_copies_octomom_genes, aes(x=wolbachia, y=normalized_fold_change))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_wrap(~gene, scales = "free_y")+
  theme_bw()

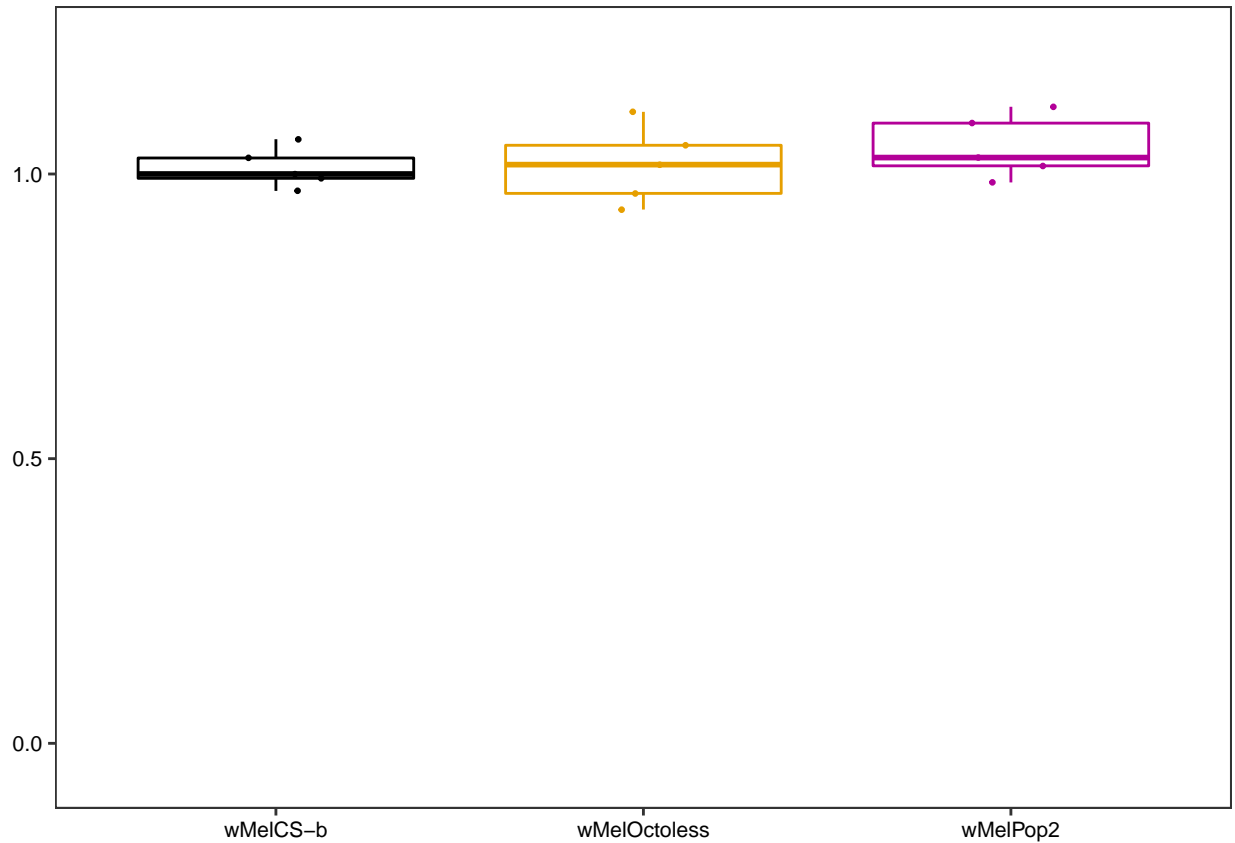
```



```

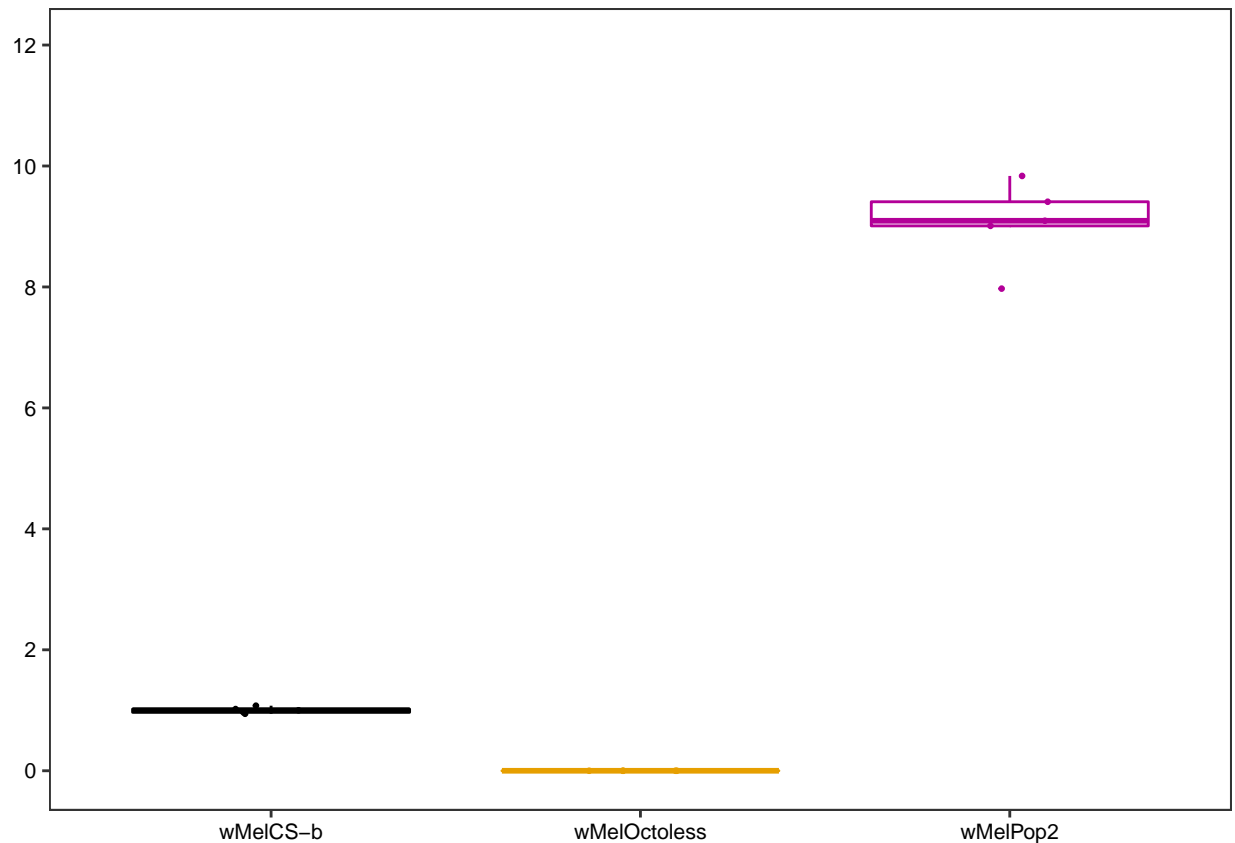
### plots
## WD0505
(s6fig_505=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0505"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(outlier.shape=NA, size=0.5)+
  geom_jitter(size=0.5, width=0.15)+
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098"))+
  scale_y_continuous(limits=c(-0.05, 1.23), breaks=seq(0, 1.5, 0.5))+
  theme_figs()+ theme(legend.position="none"))

```



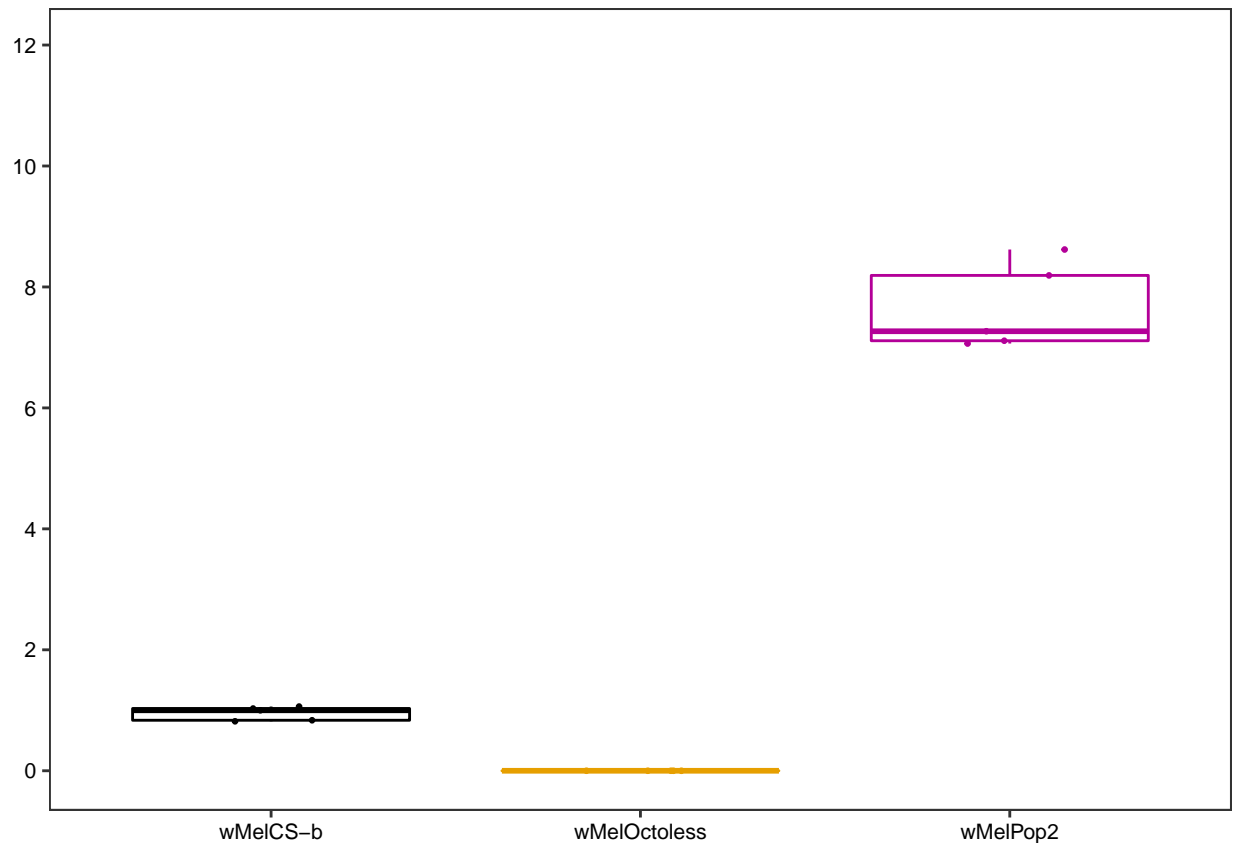
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0505.pdf", plot=s6fig_505, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0507
(s6fig_507=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0507"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



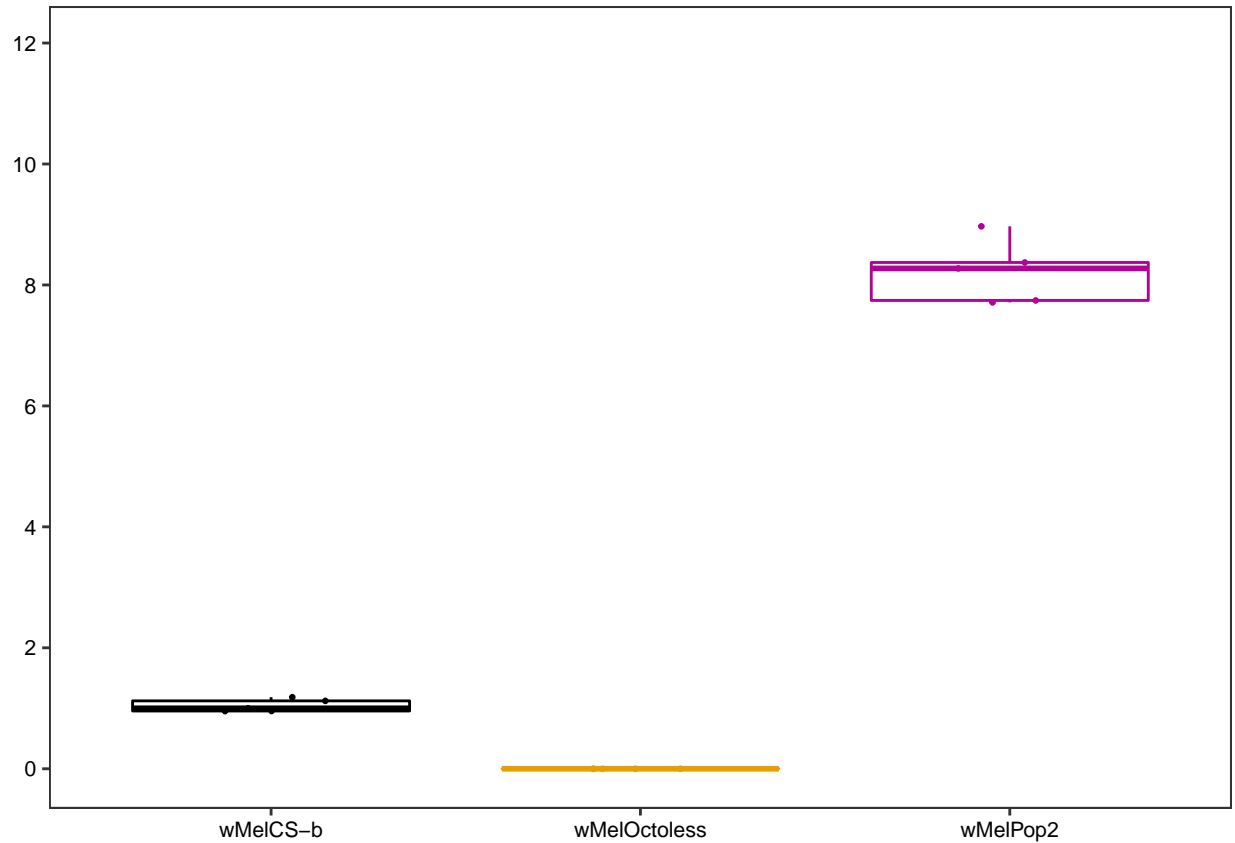
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0507.pdf", plot=s6fig_507, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0508
(s6fig_508=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0508"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



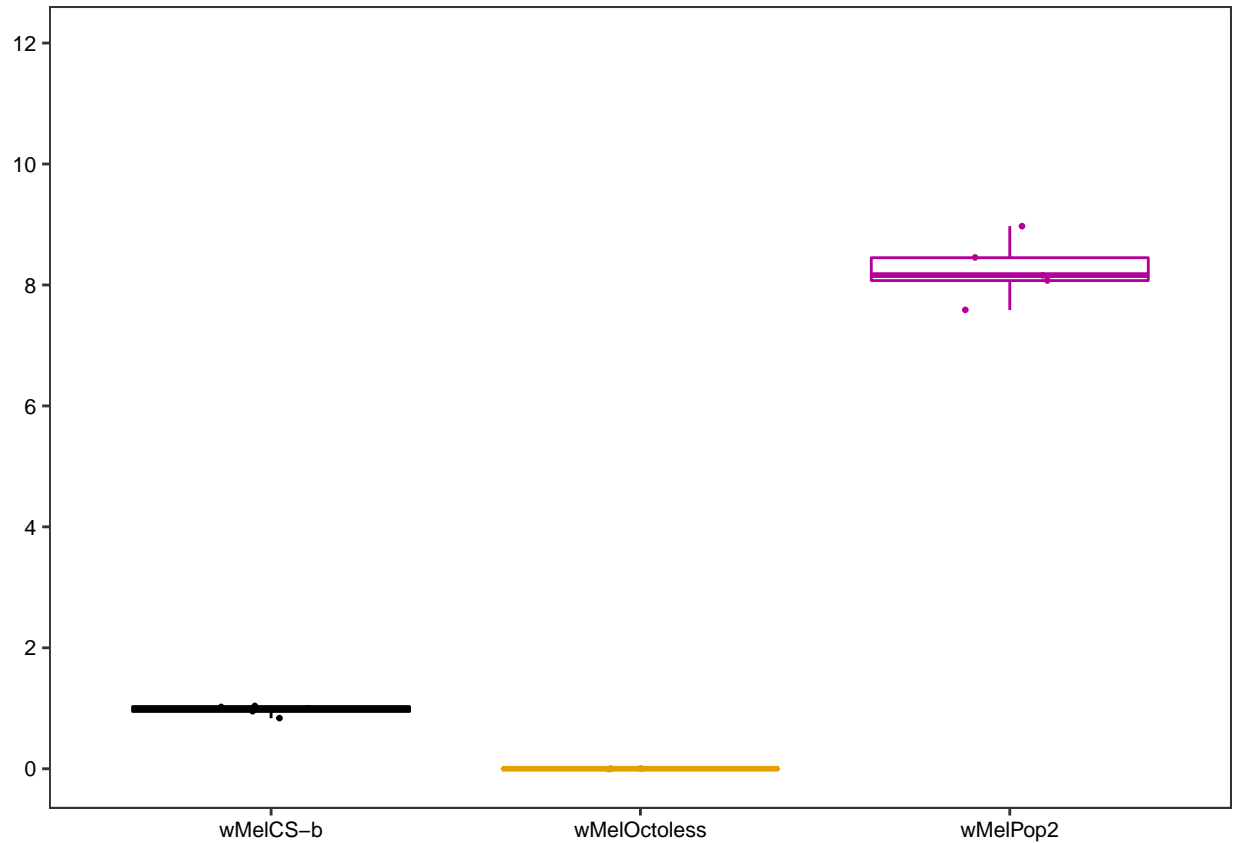
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0508.pdf", plot=s6fig_508, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0509
(s6fig_509=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0509"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0509.pdf", plot=s6fig_509, device="pdf", dpi=300, units="in", width=10, height=10)
}

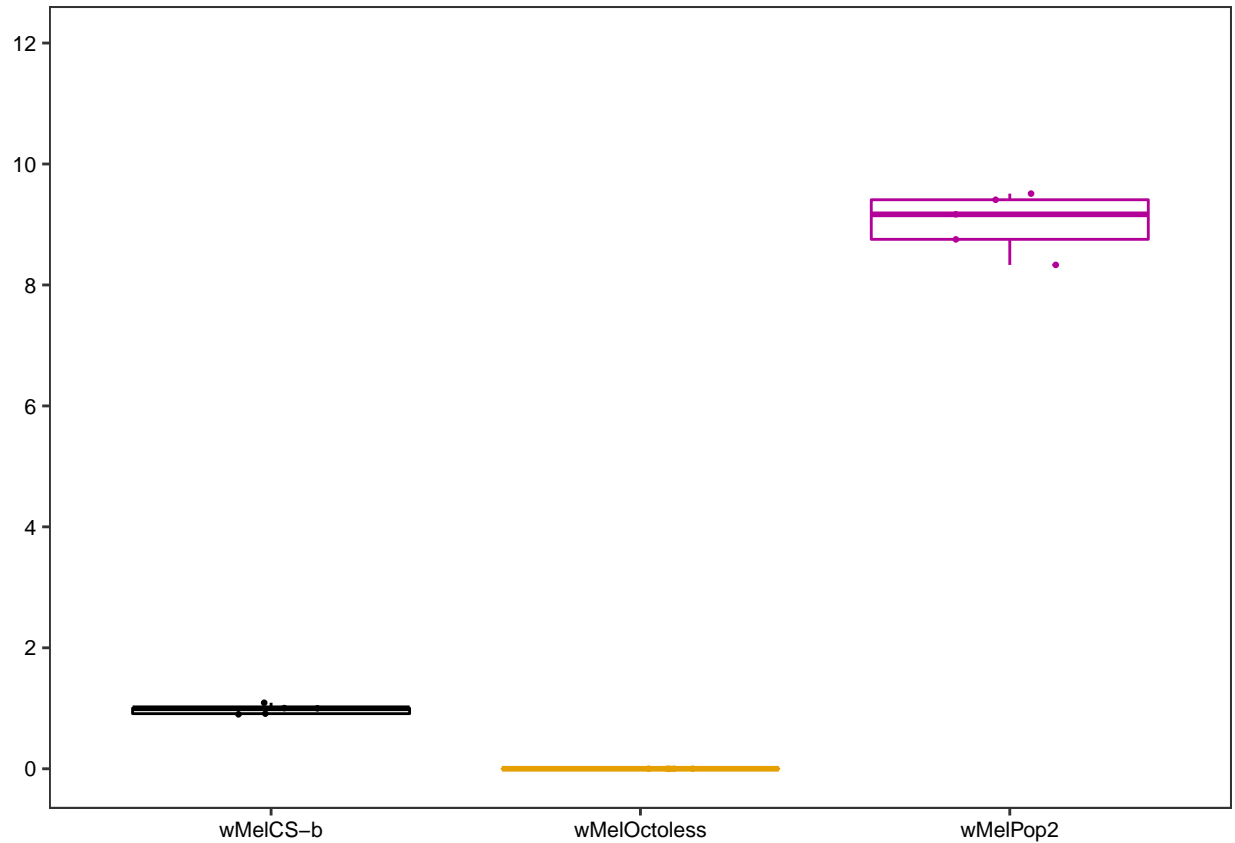
## WD0510
(s6fig_510=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0510"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0510.pdf", plot=s6fig_510, device="pdf", dpi=300, units="in", width=10, height=10)
}

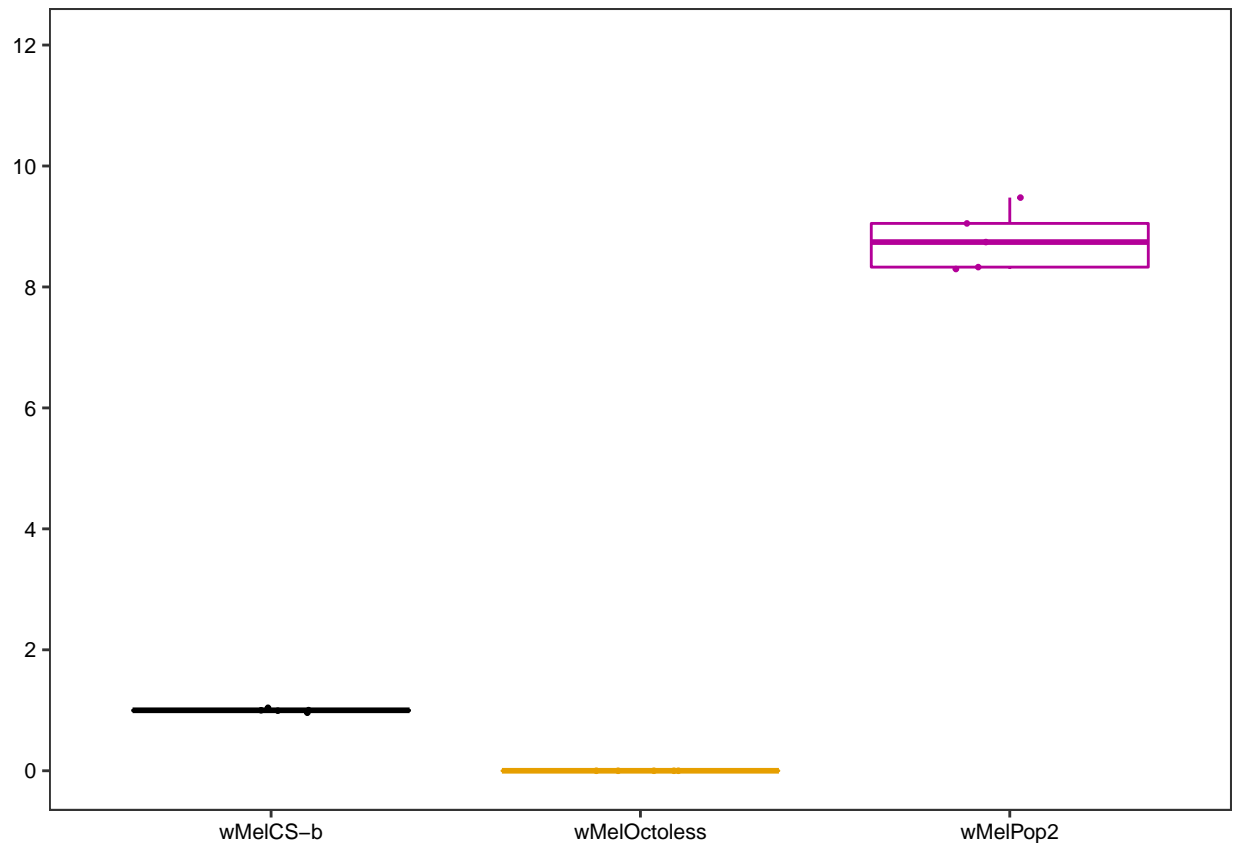
## WD0511
(s6fig_511=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0511"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```





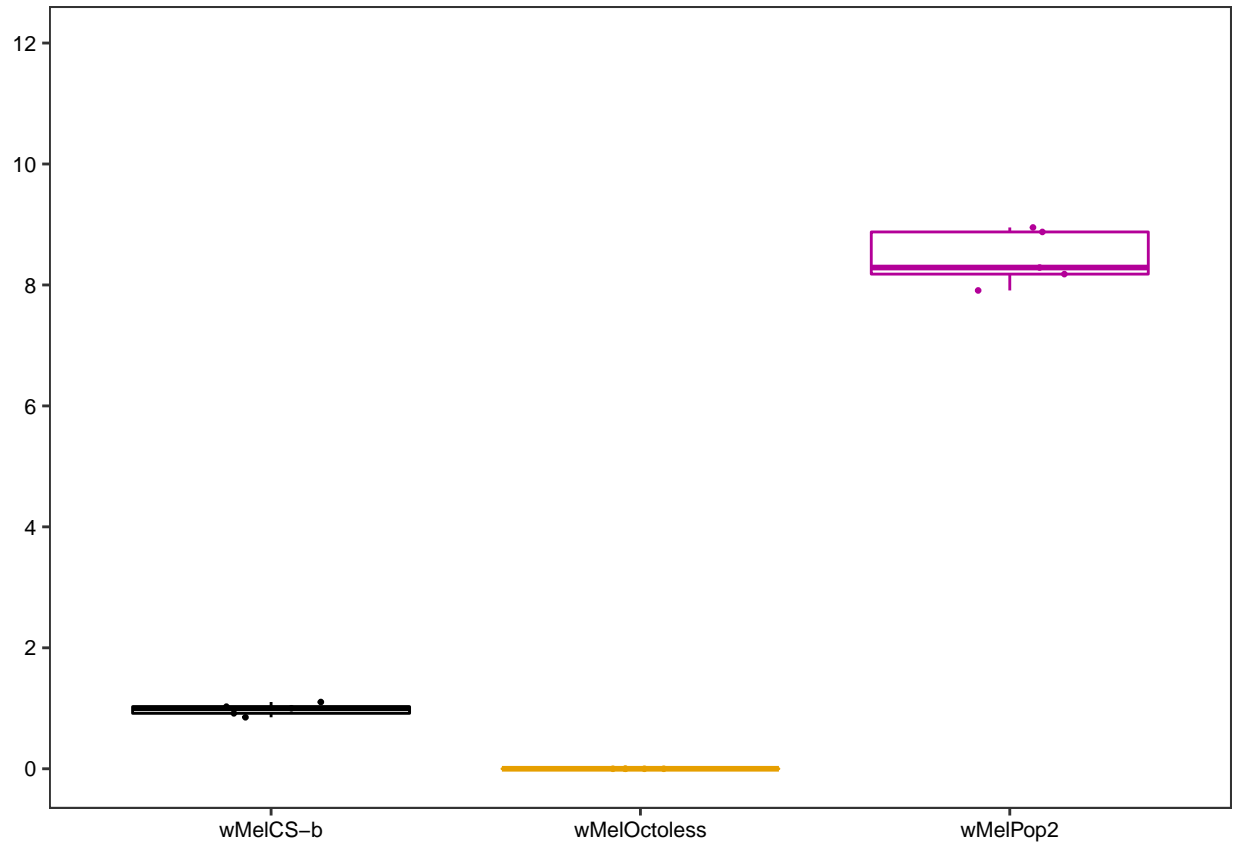
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0511.pdf", plot=s6fig_511, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0512
(s6fig_512=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0512"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



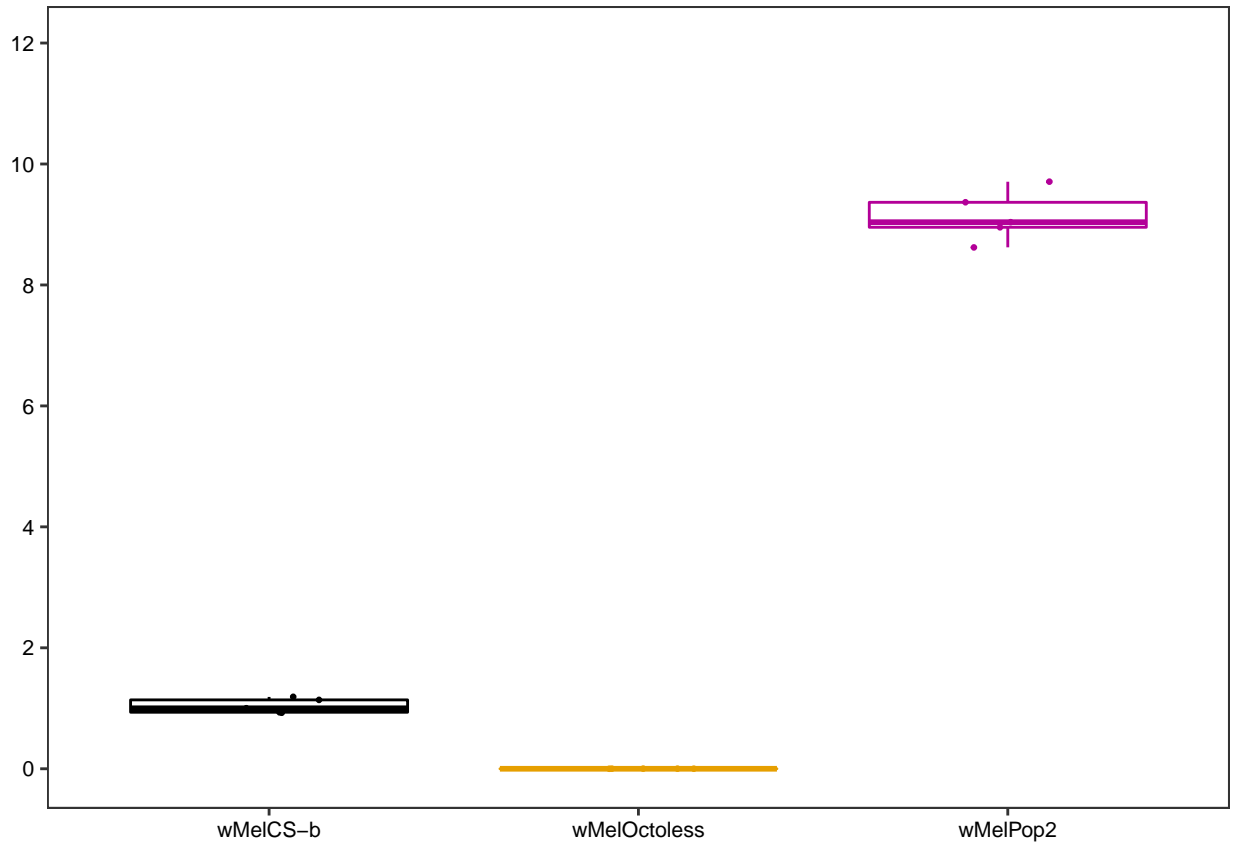
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0512.pdf", plot=s6fig_512, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0513
(s6fig_513=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0513"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



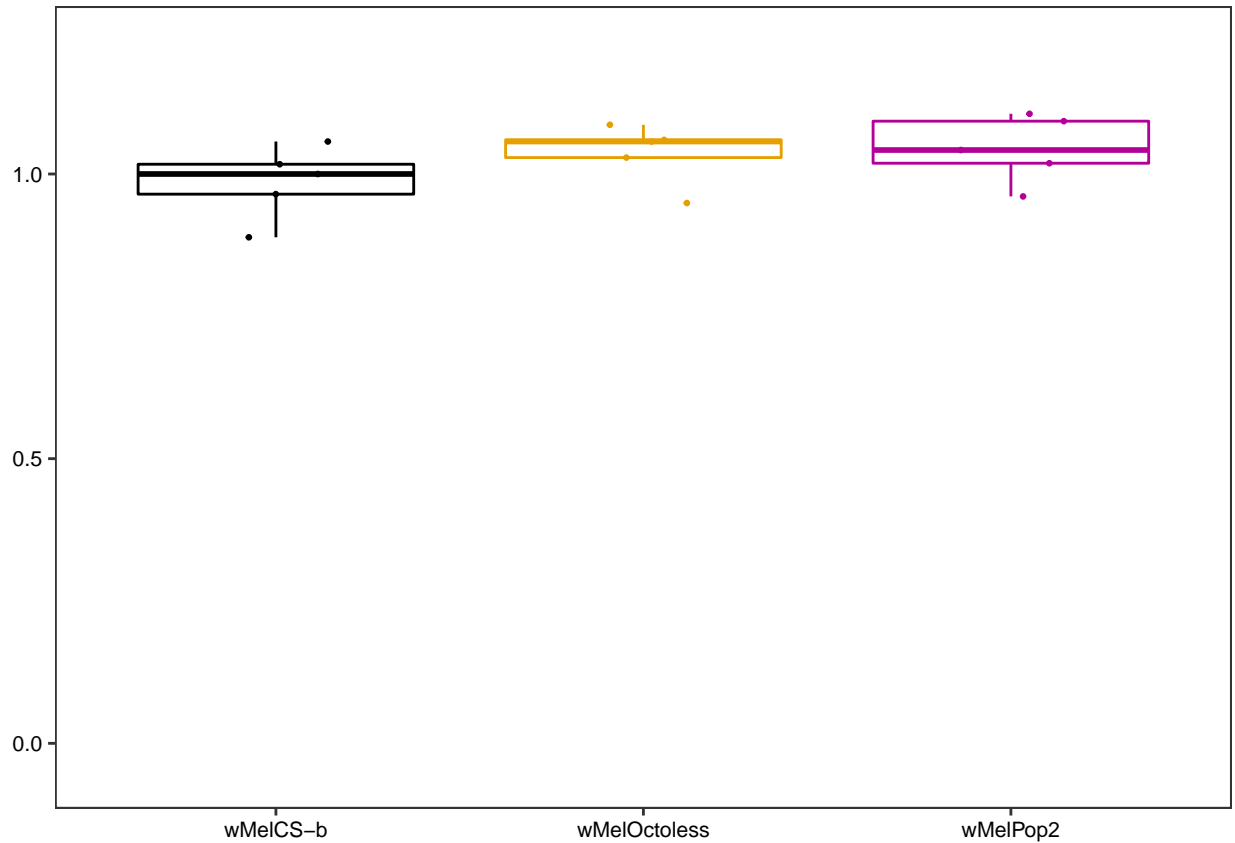
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0513.pdf", plot=s6fig_513, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0514
(s6fig_514=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0514"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 12), breaks=seq(0,12,2)) +
  theme_figs() + theme(legend.position="none"))
```



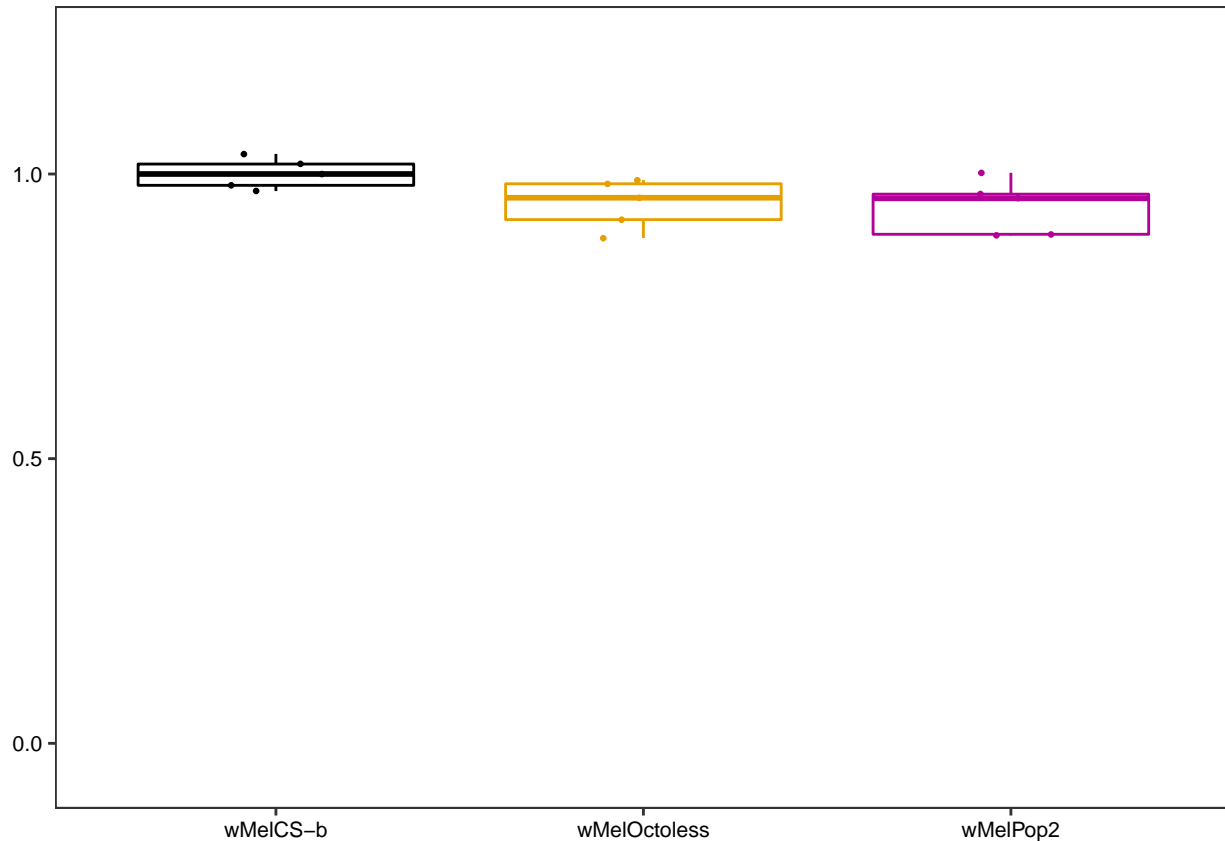
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0514.pdf", plot=s6fig_514, device="pdf", dpi=300, units="in", width=10, height=10)
}

## WD0519
(s6fig_519=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="WD0519"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 1.23), breaks=seq(0, 1.5, 0.5)) +
  theme_figs() + theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig WD0519.pdf", plot=s6fig_519, device="pdf", dpi=300, units="in", width=10, height=10)
}

## rpoD
(s6fig_rpod=ggplot(dplyr::filter(results_qpcr_copies_octomom_genes,
                                gene=="rpoD"), aes(x=wolbachia, y=normalized_fold_change, color=wolbachia)) +
  geom_boxplot(outlier.shape=NA, size=0.5) +
  geom_jitter(size=0.5, width=0.15) +
  scale_color_manual(values=c("#000000", "#e69f00", "#b30098")) +
  scale_y_continuous(limits=c(-0.05, 1.23), breaks=seq(0,1.5, 0.5)) +
  theme_figs() + theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S6 Fig rpoD.pdf", plot=s6fig_rpod, device="pdf", dpi=300, units="in", width=10, height=10)
}
```

## S7 Fig | Selection of flies with desired Octomom copy number

```
### S14 Data
s14data=read.csv("../original_data/S14 Data.csv", na.strings=c("", "NA", "Undetermined"), stringsAsFactors=FALSE)

### wolbachia titres over fly generations
results_qpcr_octomom_copy_dynamics=s14data%>%
  dplyr::group_by(generation)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="CSB", reference_gene="WSP", efficiency_ref=2, efficiency_target=1))

### adding new variables
results_qpcr_octomom_copy_dynamics=results_qpcr_octomom_copy_dynamics%>%
  tidyr::separate(sample, c("wolbachia", NA), sep="_", remove=FALSE)

## converting variable to factor
results_qpcr_octomom_copy_dynamics$wolbachia=factor(results_qpcr_octomom_copy_dynamics$wolbachia, levels=c("wolbachia", NA))

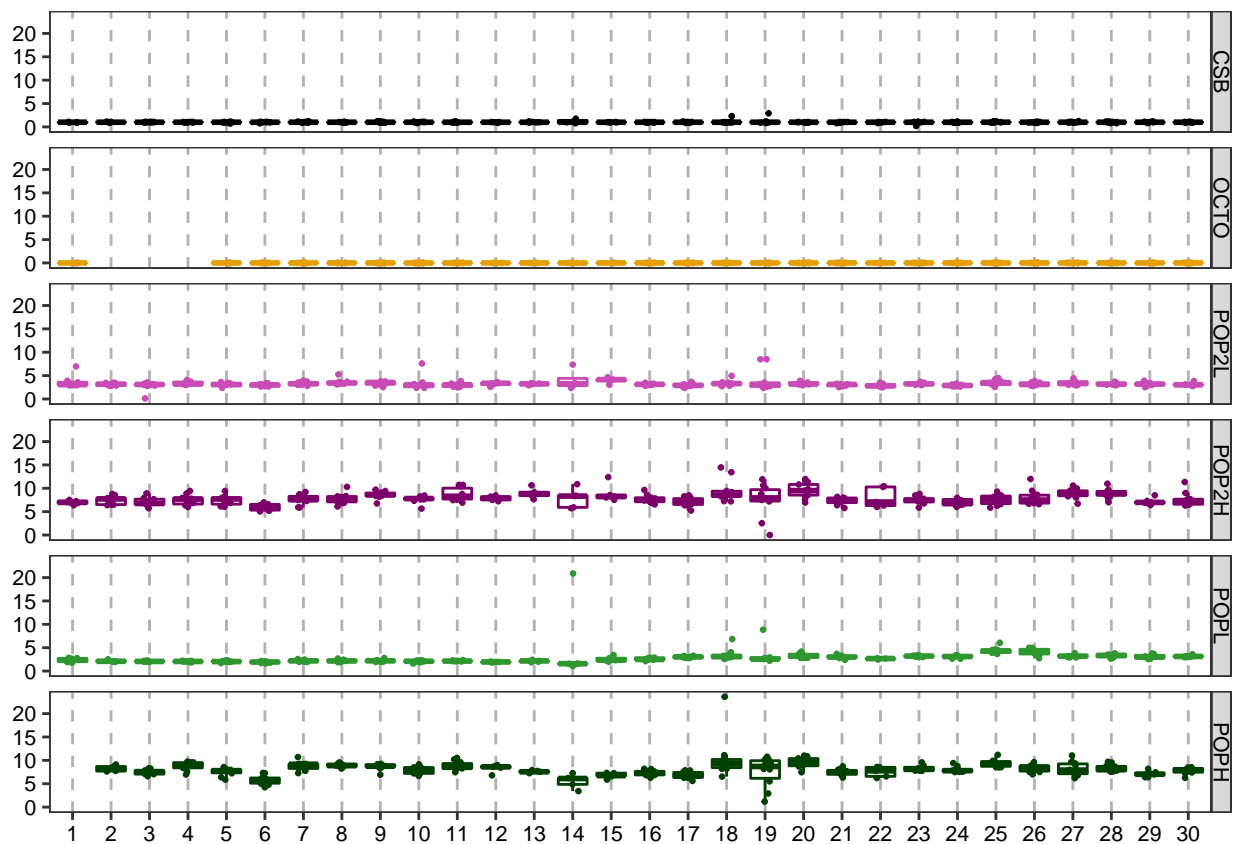
## filtering out Wolbachia-free samples
copy_dynamics_figure=droplevels(subset(results_qpcr_octomom_copy_dynamics, results_qpcr_octomom_copy_dynamics$wolbachia=="wolbachia"))
```

```

## replacing 'NaN' in wMelOctoless to '0'
# fold change
copy_dynamics_figure$fold_change[copy_dynamics_figure$wolbachia=="OCTO" & is.nan(copy_dynamics_figure$fold_change)] = 0
# normalized fold change
copy_dynamics_figure$normalized_fold_change[copy_dynamics_figure$wolbachia=="OCTO" & is.nan(copy_dynamics_figure$normalized_fold_change)] = 0

### the plot
(s7fig=ggplot(copy_dynamics_figure, aes(y=normalized_fold_change, x=factor(generation), color=wolbachia)) +
  geom_boxplot(outlier.shape=NA) +
  geom_jitter(width=0.15, size=0.5) +
  facet_grid(wolbachia~.) + theme_figs() +
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")) +
  theme(legend.position="none",
    panel.grid.major.x=element_line(size=0.5, linetype="dashed", color=grey(0.7)))

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S7 Fig.pdf", plot=s7fig, device="pdf", dpi=300, units="in", width=700/96,
}

```

**Fig 3 | Dataset**

48



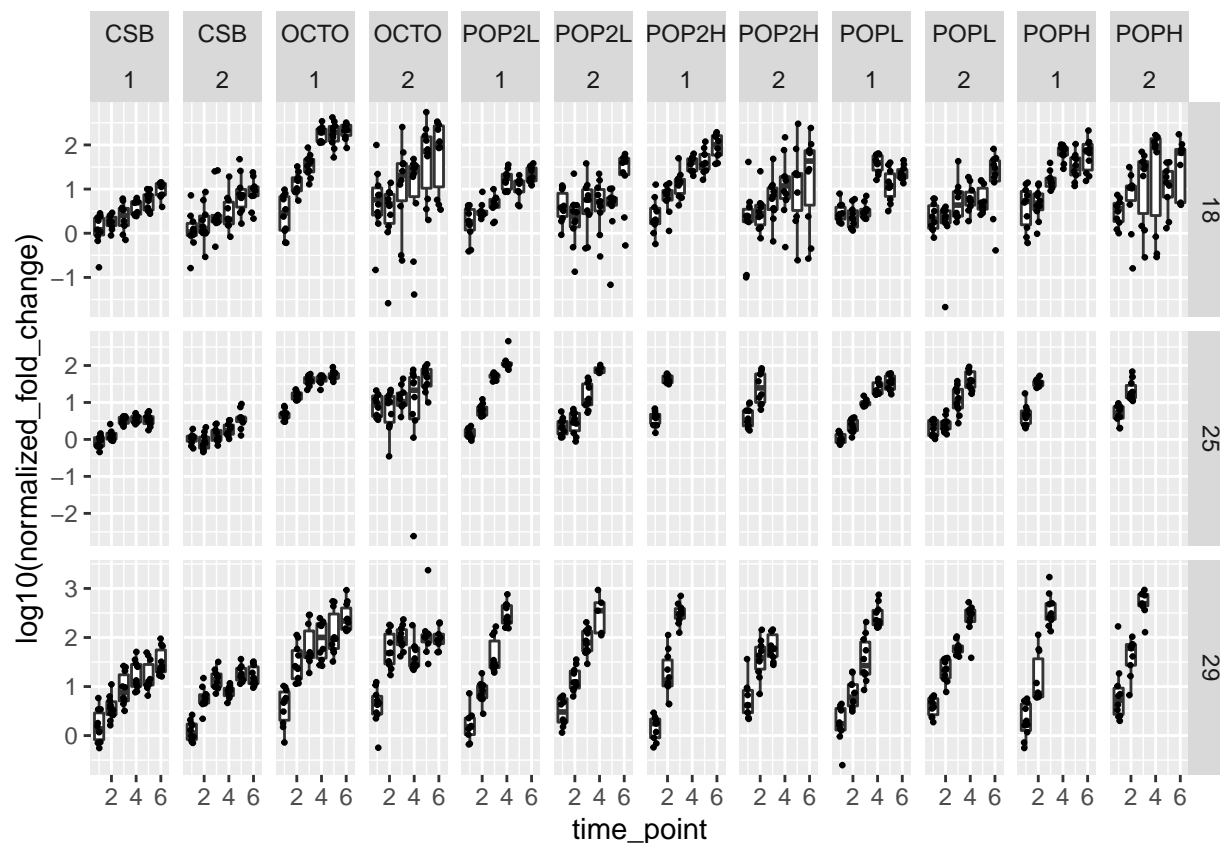


Fig 3 | Statistical analysis

full model. doubling times

```
### analysis
# converting temperature to factor
results_qpcr_wolb_proliferation$temperature <- as.factor(results_qpcr_wolb_proliferation$temperature)

## full model
model_proliferation_all_temps=lmer(log(normalized_fold_change)~day*wolbachia*temperature +(1|experiment))

# significance of the model
car::Anova(model_proliferation_all_temps)

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##
##           Chisq Df Pr(>Chisq)
## day           1013.030  1 < 2.2e-16 ***
## wolbachia       868.082  5 < 2.2e-16 ***
## temperature    1030.380  2 < 2.2e-16 ***
## day:wolbachia    40.005  5  1.49e-07 ***
## day:temperature  518.135  2 < 2.2e-16 ***
## wolbachia:temperature  172.298 10 < 2.2e-16 ***
## day:wolbachia:temperature  362.842 10 < 2.2e-16 ***
## ---
```

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

# summary
summary(model_proliferation_all_temps, corr=FALSE)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(normalized_fold_change) ~ day * wolbachia * temperature +
## (1 | experimental_replicate)
## Data: results_qpcr_wolb_proliferation
##
## REML criterion at convergence: 4989.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -8.8150 -0.4529  0.0354  0.5710  3.9638
##
## Random effects:
## Groups              Name                Variance Std.Dev.
## experimental_replicate (Intercept) 0.0335     0.183
## Residual                      1.0739     1.036
## Number of obs: 1670, groups:  experimental_replicate, 2
##
## Fixed effects:
##
##              Estimate Std. Error    df t value
## (Intercept)    1.161e-01  2.119e-01  6.651e+00  0.548
## day            4.129e-02  5.541e-03  1.633e+03  7.451
## wolbachiaOCT0  1.189e+00  2.372e-01  1.633e+03  5.010
## wolbachiaPOP2L  6.217e-01  2.372e-01  1.633e+03  2.621
## wolbachiaPOP2H  7.808e-01  2.372e-01  1.633e+03  3.291
## wolbachiaPOPL   5.558e-01  2.372e-01  1.633e+03  2.343
## wolbachiaPOPH   1.270e+00  2.375e-01  1.633e+03  5.350
## temperature25  -2.284e-01  2.457e-01  1.633e+03 -0.930
## temperature29   7.093e-01  2.372e-01  1.633e+03  2.990
## day:wolbachiaOCT0  3.236e-02  7.835e-03  1.633e+03  4.130
## day:wolbachiaPOP2L -2.074e-03  7.835e-03  1.633e+03 -0.265
## day:wolbachiaPOP2H  1.718e-02  7.835e-03  1.633e+03  2.192
## day:wolbachiaPOPL  3.616e-03  7.835e-03  1.633e+03  0.462
## day:wolbachiaPOPH  8.879e-03  7.884e-03  1.633e+03  1.126
## day:temperature25  8.589e-03  1.184e-02  1.633e+03  0.725
## day:temperature29  1.292e-01  1.934e-02  1.633e+03  6.683
## wolbachiaOCT0:temperature25  7.727e-01  3.474e-01  1.633e+03  2.224
## wolbachiaPOP2L:temperature25 -1.152e-01  3.562e-01  1.633e+03 -0.323
## wolbachiaPOP2H:temperature25  6.581e-01  3.771e-01  1.633e+03  1.745
## wolbachiaPOPL:temperature25 -1.178e-01  3.497e-01  1.633e+03 -0.337
## wolbachiaPOPH:temperature25  3.836e-01  3.772e-01  1.633e+03  1.017
## wolbachiaOCT0:temperature29  3.446e-01  3.356e-01  1.633e+03  1.027
## wolbachiaPOP2L:temperature29 -6.640e-01  3.522e-01  1.633e+03 -1.885
## wolbachiaPOP2H:temperature29 -4.852e-01  3.621e-01  1.633e+03 -1.340
## wolbachiaPOPL:temperature29 -4.259e-01  3.494e-01  1.633e+03 -1.219
## wolbachiaPOPH:temperature29 -9.507e-01  3.599e-01  1.633e+03 -2.641
## day:wolbachiaOCT0:temperature25 -1.598e-02  1.675e-02  1.633e+03 -0.954
## day:wolbachiaPOP2L:temperature25  1.505e-01  2.068e-02  1.633e+03  7.278
## day:wolbachiaPOP2H:temperature25  2.400e-01  4.861e-02  1.633e+03  4.937
## day:wolbachiaPOPL:temperature25  7.738e-02  1.767e-02  1.633e+03  4.380
```

```

## day:wolbachiaPOPH:temperature25 1.894e-01 4.861e-02 1.633e+03 3.895
## day:wolbachiaOCT0:temperature29 2.777e-03 2.742e-02 1.633e+03 0.101
## day:wolbachiaPOP2L:temperature29 3.719e-01 4.234e-02 1.633e+03 8.785
## day:wolbachiaPOP2H:temperature29 4.608e-01 5.887e-02 1.633e+03 7.827
## day:wolbachiaPOPL:temperature29 3.273e-01 4.032e-02 1.633e+03 8.118
## day:wolbachiaPOPH:temperature29 6.073e-01 5.821e-02 1.633e+03 10.434
##
## Pr(>|t|)
## (Intercept) 0.601750
## day 1.50e-13 ***
## wolbachiaOCT0 6.02e-07 ***
## wolbachiaPOP2L 0.008860 **
## wolbachiaPOP2H 0.001018 **
## wolbachiaPOPL 0.019249 *
## wolbachiaPOPH 1.01e-07 ***
## temperature25 0.352696
## temperature29 0.002834 **
## day:wolbachiaOCT0 3.81e-05 ***
## day:wolbachiaPOP2L 0.791221
## day:wolbachiaPOP2H 0.028493 *
## day:wolbachiaPOPL 0.644479
## day:wolbachiaPOPH 0.260259
## day:temperature25 0.468480
## day:temperature29 3.20e-11 ***
## wolbachiaOCT0:temperature25 0.026276 *
## wolbachiaPOP2L:temperature25 0.746381
## wolbachiaPOP2H:temperature25 0.081107 .
## wolbachiaPOPL:temperature25 0.736144
## wolbachiaPOPH:temperature25 0.309373
## wolbachiaOCT0:temperature29 0.304715
## wolbachiaPOP2L:temperature29 0.059568 .
## wolbachiaPOP2H:temperature29 0.180410
## wolbachiaPOPL:temperature29 0.223121
## wolbachiaPOPH:temperature29 0.008343 **
## day:wolbachiaOCT0:temperature25 0.340161
## day:wolbachiaPOP2L:temperature25 5.23e-13 ***
## day:wolbachiaPOP2H:temperature25 8.76e-07 ***
## day:wolbachiaPOPL:temperature25 1.26e-05 ***
## day:wolbachiaPOPH:temperature25 0.000102 ***
## day:wolbachiaOCT0:temperature29 0.919338
## day:wolbachiaPOP2L:temperature29 < 2e-16 ***
## day:wolbachiaPOP2H:temperature29 8.90e-15 ***
## day:wolbachiaPOPL:temperature29 9.22e-16 ***
## day:wolbachiaPOPH:temperature29 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 36 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it

## comparing temperature per Wolbachia
multcomp_all_temps=emmeans::lstrends(model_proliferation_all_temps, consec~temperature | wolbachia, var=

```

```
## estimating doubling times +/- se and 95% confidence interval
```

```
(doubling_time_all <- as.data.frame(multcomp_all_temps$lstrends) %>%
  mutate(doubling_time = log(2)/day.trend) %>%
  mutate(doubling_lower.CL = log(2)/(upper.CL)) %>%
  mutate(doubling_upper.CL = log(2)/(lower.CL)) %>%
  mutate(lower = log(2)/(day.trend+SE)) %>%
  mutate(upper = log(2)/(day.trend-SE)))
```

##	temperature	wolbachia	day.trend	SE	df	lower.CL	upper.CL
## 1	18	CSB	0.04128594	0.005541166	1633.000	0.03041740	0.05215448
## 2	25	CSB	0.04987444	0.010468045	1633.000	0.02934223	0.07040665
## 3	29	CSB	0.17050137	0.018524076	1633.000	0.13416792	0.20683482
## 4	18	OCTO	0.07364565	0.005539169	1633.000	0.06278103	0.08451027
## 5	25	OCTO	0.06625299	0.010468045	1633.000	0.04572078	0.08678520
## 6	29	OCTO	0.20563796	0.018632532	1633.001	0.16909178	0.24218414
## 7	18	POP2L	0.03921148	0.005539169	1633.000	0.02834686	0.05007611
## 8	25	POP2L	0.19831421	0.016021728	1633.036	0.16688891	0.22973951
## 9	29	POP2L	0.54035233	0.037258093	1633.072	0.46727365	0.61343102
## 10	18	POP2H	0.05846320	0.005539169	1633.000	0.04759858	0.06932783
## 11	25	POP2H	0.30700853	0.046814521	1633.000	0.21518570	0.39883137
## 12	29	POP2H	0.64844625	0.055325240	1633.002	0.53993034	0.75696216
## 13	18	POPL	0.04490202	0.005539169	1633.000	0.03403740	0.05576665
## 14	25	POPL	0.13087304	0.011882228	1633.218	0.10756703	0.15417905
## 15	29	POPL	0.50142757	0.034943443	1633.002	0.43288888	0.56996626
## 16	18	POPH	0.05016514	0.005608885	1633.004	0.03916377	0.06116651
## 17	25	POPH	0.24812320	0.046814521	1633.000	0.15630036	0.33994603
## 18	29	POPH	0.78672586	0.054616942	1633.000	0.67959922	0.89385250
##	doubling_time	doubling_lower.CL	doubling_upper.CL	lower	upper		
## 1	16.788939	13.2902702	22.787851	14.8022632	19.3915660		
## 2	13.897843	9.8449103	23.622850	11.4868843	17.5897119		
## 3	4.065347	3.3512112	5.166266	3.6669517	4.5608601		
## 4	9.411923	8.2019279	11.040711	8.7535362	10.1774041		
## 5	10.462126	7.9869283	15.160440	9.0346430	12.4253445		
## 6	3.370716	2.8620668	4.099237	3.0906750	3.7065619		
## 7	17.677147	13.8418739	24.452344	15.4890963	20.5850761		
## 8	3.495197	3.0171005	4.153345	3.2339289	3.8023904		
## 9	1.282769	1.1299513	1.483386	1.2000254	1.3777681		
## 10	11.856128	9.9981093	14.562351	10.8300235	13.0970212		
## 11	2.257746	1.7379455	3.221158	1.9590221	2.6639629		
## 12	1.068935	0.9156959	1.283771	0.9849038	1.1686438		
## 13	15.436880	12.4294214	20.364281	13.7416887	17.6091689		
## 14	5.296333	4.4957286	6.443863	4.8554928	5.8252159		
## 15	1.382348	1.2161197	1.601213	1.2922905	1.4858966		
## 16	13.817307	11.3321359	17.698682	12.4277773	15.5566746		
## 17	2.793561	2.0389918	4.434713	2.3501476	3.4432057		
## 18	0.881053	0.7754604	1.019935	0.8238582	0.9467815		

```
doubling_time_all$wolbachia=factor(doubling_time_all$wolbachia, levels=c("CSB", "OCTO", "POP2L", "POP2H"))
```

```
if(save_files==TRUE)
{
  write.csv(doubling_time_all,"../results/Wolbachia_Doubling_Time_Adults.csv")
}
```

## direct comparison of octoless and csb

```
### Analysis of wMelCS_b and wMelOctoless only
# the model with interaction between Wolbachia and temperature
model_prol_octo_csb=lmer(log(normalized_fold_change)~day*wolbachia*temperature +(1|experimental_replica

# significance
car::Anova(model_prol_octo_csb)

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##               Chisq Df Pr(>Chisq)
## day              373.6556   1 < 2.2e-16 ***
## wolbachia        594.7092   1 < 2.2e-16 ***
## temperature      391.9037   2 < 2.2e-16 ***
## day:wolbachia     18.3213   1 1.866e-05 ***
## day:temperature   88.2933   2 < 2.2e-16 ***
## wolbachia:temperature  5.6883   2  0.05818 .
## day:wolbachia:temperature 0.9226   2  0.63047
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# A simpler model removing triple interaction and wolbachia*temperature interaction shown not to be sig
model_prol_octo_csb_simpler=lmer(log(normalized_fold_change)~day*temperature + day*wolbachia +(1|experim

# significance
car::Anova(model_prol_octo_csb_simpler)

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##               Chisq Df Pr(>Chisq)
## day              372.239   1 < 2.2e-16 ***
## temperature      390.377   2 < 2.2e-16 ***
## wolbachia        592.393   1 < 2.2e-16 ***
## day:temperature   87.934   2 < 2.2e-16 ***
## day:wolbachia     16.386   1 5.167e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_prol_octo_csb_simpler, corr=FALSE)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(normalized_fold_change) ~ day * temperature + day * wolbachia +
## (1 | experimental_replicate)
## Data: filter(results_qpcr_wolb_proliferation, wolbachia == "CSB" |
## wolbachia == "OCTO")
##
## REML criterion at convergence: 2041.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -8.4551 -0.4603  0.0573  0.5191  3.0987
```

```

##
## Random effects:
##   Groups              Name      Variance Std.Dev.
## experimental_replicate (Intercept) 0.08569  0.2927
## Residual                1.12332  1.0599
## Number of obs: 677, groups: experimental_replicate, 2
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  -9.799e-02  2.475e-01  1.895e+00  -0.396    0.732
## day          4.571e-02  4.951e-03  6.680e+02   9.233 < 2e-16 ***
## temperature25 1.580e-01  1.777e-01  6.680e+02   0.889    0.374
## temperature29 8.822e-01  1.716e-01  6.680e+02   5.140 3.60e-07 ***
## wolbachiaOCT0 1.617e+00  1.217e-01  6.680e+02  13.279 < 2e-16 ***
## day:temperature25 5.983e-04  8.565e-03  6.680e+02   0.070    0.944
## day:temperature29 1.304e-01  1.402e-02  6.680e+02   9.302 < 2e-16 ***
## day:wolbachiaOCT0 2.352e-02  5.809e-03  6.680e+02   4.048 5.77e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# see estimates of growth for both Wolbachia and different temperatures
(multcomp_octo_cs_all_temps=emmeans::lstrends(model_prol_octo_csb_simpler, ~temperature | wolbachia, var=

## wolbachia = CSB:
##   temperature day.trend      SE  df lower.CL upper.CL
## 18             0.0457 0.00495 668  0.0338  0.0576
## 25             0.0463 0.00811 668  0.0268  0.0658
## 29             0.1761 0.01375 668  0.1431  0.2091
##
## wolbachia = OCT0:
##   temperature day.trend      SE  df lower.CL upper.CL
## 18             0.0692 0.00495 668  0.0574  0.0811
## 25             0.0698 0.00811 668  0.0504  0.0893
## 29             0.1996 0.01375 668  0.1667  0.2326
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates

# contrasts of Wolbachia growth at different temperatures - equal for both since there is no triple int
(multcomp_octo_cs_between_temps=emmeans::lstrends(model_prol_octo_csb_simpler, consec~temperature, var=

## $lstrends
##   temperature day.trend      SE  df lower.CL upper.CL
## 18             0.0575 0.00401 668  0.0478  0.0671
## 25             0.0581 0.00757 668  0.0399  0.0762
## 29             0.1879 0.01344 668  0.1556  0.2201
##
## Results are averaged over the levels of: wolbachia
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
##   contrast estimate      SE  df t.ratio p.value
## 25 - 18  0.000598 0.00857 668  0.070    0.9443

```

```
## 29 - 25 0.129823 0.01542 668 8.418 <.0001
##
## Results are averaged over the levels of: wolbachia
## Degrees-of-freedom method: kenward-roger
## P value adjustment: holm method for 2 tests
```

## comparison of new variants to cs\_b

```
### analysis
# converting temperature to factor
results_qpcr_wolb_prol_csb <- filter(results_qpcr_wolb_proliferation, wolbachia != "POPH" & wolbachia != "POPH")

# full model of csb and new variants
model_proliferation_csb_temps=lmer(log(normalized_fold_change)~day*wolbachia*temperature +(1|experimental_replicate), data=results_qpcr_wolb_prol_csb)

# significance of the model
car::Anova(model_proliferation_csb_temps)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##               Chisq Df Pr(>Chisq)
## day              692.297  1 < 2.2e-16 ***
## wolbachia        702.950  3 < 2.2e-16 ***
## temperature      673.534  2 < 2.2e-16 ***
## day:wolbachia     29.221  3 2.013e-06 ***
## day:temperature   262.000  2 < 2.2e-16 ***
## wolbachia:temperature 103.549 6 < 2.2e-16 ***
## day:wolbachia:temperature 214.989 6 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# summary
summary(model_proliferation_csb_temps, corr=FALSE)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(normalized_fold_change) ~ day * wolbachia * temperature +
## (1 | experimental_replicate)
## Data: results_qpcr_wolb_prol_csb
##
## REML criterion at convergence: 3492.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -8.7002 -0.4478  0.0550  0.5443  3.3400
##
## Random effects:
##   Groups              Name              Variance Std.Dev.
## experimental_replicate (Intercept) 0.07266  0.2696
## Residual                  1.08769  1.0429
## Number of obs: 1164, groups: experimental_replicate, 2
##
## Fixed effects:
##               Estimate Std. Error      df t value
```

```

## (Intercept)          1.164e-01  2.546e-01  3.026e+00  0.457
## day                  4.129e-02  5.577e-03  1.139e+03  7.405
## wolbachiaOCT0       1.188e+00  2.387e-01  1.139e+03  4.977
## wolbachiaPOP2L      6.214e-01  2.387e-01  1.139e+03  2.603
## wolbachiaPOP2H      7.805e-01  2.387e-01  1.139e+03  3.269
## temperature25       -2.287e-01  2.473e-01  1.139e+03 -0.925
## temperature29       7.091e-01  2.388e-01  1.139e+03  2.970
## day:wolbachiaOCT0   3.235e-02  7.885e-03  1.139e+03  4.103
## day:wolbachiaPOP2L -2.083e-03  7.885e-03  1.139e+03 -0.264
## day:wolbachiaPOP2H  1.717e-02  7.885e-03  1.139e+03  2.177
## day:temperature25   8.580e-03  1.192e-02  1.139e+03  0.720
## day:temperature29  1.291e-01  1.946e-02  1.139e+03  6.635
## wolbachiaOCT0:temperature25 7.730e-01  3.497e-01  1.139e+03  2.211
## wolbachiaPOP2L:temperature25 -1.118e-01  3.585e-01  1.139e+03 -0.312
## wolbachiaPOP2H:temperature25 6.584e-01  3.795e-01  1.139e+03  1.735
## wolbachiaOCT0:temperature29 3.453e-01  3.378e-01  1.139e+03  1.022
## wolbachiaPOP2L:temperature29 -6.578e-01  3.545e-01  1.139e+03 -1.856
## wolbachiaPOP2H:temperature29 -4.824e-01  3.644e-01  1.139e+03 -1.324
## day:wolbachiaOCT0:temperature25 -1.597e-02  1.686e-02  1.139e+03 -0.948
## day:wolbachiaPOP2L:temperature25 1.499e-01  2.081e-02  1.139e+03  7.200
## day:wolbachiaPOP2H:temperature25 2.400e-01  4.892e-02  1.139e+03  4.906
## day:wolbachiaOCT0:temperature29 2.725e-03  2.759e-02  1.139e+03  0.099
## day:wolbachiaPOP2L:temperature29 3.698e-01  4.261e-02  1.139e+03  8.678
## day:wolbachiaPOP2H:temperature29 4.603e-01  5.924e-02  1.139e+03  7.770
## Pr(>|t|)
## (Intercept)          0.67851
## day                  2.55e-13 ***
## wolbachiaOCT0       7.45e-07 ***
## wolbachiaPOP2L      0.00937 **
## wolbachiaPOP2H      0.00111 **
## temperature25       0.35521
## temperature29       0.00304 **
## day:wolbachiaOCT0   4.37e-05 ***
## day:wolbachiaPOP2L  0.79167
## day:wolbachiaPOP2H  0.02966 *
## day:temperature25   0.47181
## day:temperature29  4.99e-11 ***
## wolbachiaOCT0:temperature25 0.02725 *
## wolbachiaPOP2L:temperature25 0.75514
## wolbachiaPOP2H:temperature25 0.08300 .
## wolbachiaOCT0:temperature29 0.30682
## wolbachiaPOP2L:temperature29 0.06376 .
## wolbachiaPOP2H:temperature29 0.18582
## day:wolbachiaOCT0:temperature25 0.34357
## day:wolbachiaPOP2L:temperature25 1.09e-12 ***
## day:wolbachiaPOP2H:temperature25 1.07e-06 ***
## day:wolbachiaOCT0:temperature29 0.92135
## day:wolbachiaPOP2L:temperature29 < 2e-16 ***
## day:wolbachiaPOP2H:temperature29 1.74e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 24 > 12.

```



```

## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it

### multiple comparison
## comparing temperature per Wolbachia
(multcomp_csb_temps=emmeans::lstrends(model_proliferation_csb_temps, consec~temperature | wolbachia, va

## $lstrends
## wolbachia = CSB:
##   temperature day.trend      SE    df lower.CL upper.CL
##  18             0.0413 0.00558 1139   0.0279   0.0547
##  25             0.0499 0.01054 1139   0.0246   0.0751
##  29             0.1704 0.01864 1139   0.1257   0.2151
##
## wolbachia = OCT0:
##   temperature day.trend      SE    df lower.CL upper.CL
##  18             0.0736 0.00557 1139   0.0603   0.0870
##  25             0.0663 0.01054 1139   0.0410   0.0915
##  29             0.2055 0.01875 1139   0.1605   0.2504
##
## wolbachia = POP2L:
##   temperature day.trend      SE    df lower.CL upper.CL
##  18             0.0392 0.00557 1139   0.0258   0.0526
##  25             0.1977 0.01613 1139   0.1590   0.2363
##  29             0.5381 0.03750 1139   0.4482   0.6281
##
## wolbachia = POP2H:
##   temperature day.trend      SE    df lower.CL upper.CL
##  18             0.0585 0.00557 1139   0.0451   0.0718
##  25             0.3070 0.04711 1139   0.1941   0.4200
##  29             0.6479 0.05568 1139   0.5144   0.7814
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
## wolbachia = CSB:
##   contrast estimate      SE    df t.ratio p.value
##  25 - 18    0.00858 0.0119 1139   0.720  0.4718
##  29 - 25    0.12054 0.0214 1139   5.629 <.0001
##
## wolbachia = OCT0:
##   contrast estimate      SE    df t.ratio p.value
##  25 - 18   -0.00739 0.0119 1139  -0.620  0.5352
##  29 - 25    0.13924 0.0215 1139   6.473 <.0001
##
## wolbachia = POP2L:
##   contrast estimate      SE    df t.ratio p.value
##  25 - 18    0.15844 0.0171 1139   9.286 <.0001
##  29 - 25    0.34049 0.0408 1139   8.343 <.0001
##
## wolbachia = POP2H:
##   contrast estimate      SE    df t.ratio p.value
##  25 - 18    0.24855 0.0474 1139   5.239 <.0001

```

```

## 29 - 25    0.34091 0.0729 1139  4.674  <.0001
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: holm method for 2 tests
#
multcomp::cld(multcomp_csb_temps$lstrends, adj="holm")

## wolbachia = CSB:
## temperature day.trend      SE    df lower.CL upper.CL .group
## 18          0.0413 0.00558 1139   0.0279   0.0547   1
## 25          0.0499 0.01054 1139   0.0246   0.0751   1
## 29          0.1704 0.01864 1139   0.1257   0.2151   2
##
## wolbachia = OCTO:
## temperature day.trend      SE    df lower.CL upper.CL .group
## 25          0.0663 0.01054 1139   0.0410   0.0915   1
## 18          0.0736 0.00557 1139   0.0603   0.0870   1
## 29          0.2055 0.01875 1139   0.1605   0.2504   2
##
## wolbachia = POP2L:
## temperature day.trend      SE    df lower.CL upper.CL .group
## 18          0.0392 0.00557 1139   0.0258   0.0526   1
## 25          0.1977 0.01613 1139   0.1590   0.2363   2
## 29          0.5381 0.03750 1139   0.4482   0.6281   3
##
## wolbachia = POP2H:
## temperature day.trend      SE    df lower.CL upper.CL .group
## 18          0.0585 0.00557 1139   0.0451   0.0718   1
## 25          0.3070 0.04711 1139   0.1941   0.4200   2
## 29          0.6479 0.05568 1139   0.5144   0.7814   3
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05
## comparing Wolbachia per temperature
(multcomp_csb_wolb=emmeans::lstrends(model_proliferation_csb_temps, pairwise~wolbachia | temperature, v

## $lstrends
## temperature = 18:
## wolbachia day.trend      SE    df lower.CL upper.CL
## CSB          0.0413 0.00558 1139   0.0273   0.0552
## OCTO          0.0736 0.00557 1139   0.0597   0.0876
## POP2L          0.0392 0.00557 1139   0.0253   0.0532
## POP2H          0.0585 0.00557 1139   0.0445   0.0724
##
## temperature = 25:
## wolbachia day.trend      SE    df lower.CL upper.CL
## CSB          0.0499 0.01054 1139   0.0235   0.0762
## OCTO          0.0663 0.01054 1139   0.0399   0.0926
## POP2L          0.1977 0.01613 1139   0.1573   0.2380
## POP2H          0.3070 0.04711 1139   0.1891   0.4249
##

```

```

## temperature = 29:
## wolbachia day.trend      SE    df lower.CL upper.CL
## CSB          0.1704 0.01864 1139   0.1238   0.2171
## OCTO         0.2055 0.01875 1139   0.1586   0.2524
## POP2L        0.5381 0.03750 1139   0.4443   0.6320
## POP2H        0.6479 0.05568 1139   0.5086   0.7872
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
##
## $contrasts
## temperature = 18:
## contrast      estimate      SE    df t.ratio p.value
## CSB - OCTO    -0.03235 0.00789 1139 -4.103 0.0002
## CSB - POP2L    0.00208 0.00789 1139  0.264 0.7917
## CSB - POP2H   -0.01717 0.00789 1139 -2.177 0.0890
## OCTO - POP2L   0.03443 0.00788 1139  4.368 0.0001
## OCTO - POP2H   0.01518 0.00788 1139  1.926 0.1088
## POP2L - POP2H -0.01925 0.00788 1139 -2.442 0.0590
##
## temperature = 25:
## contrast      estimate      SE    df t.ratio p.value
## CSB - OCTO    -0.01638 0.01490 1139 -1.099 0.2719
## CSB - POP2L   -0.14778 0.01926 1139 -7.672 <.0001
## CSB - POP2H   -0.25713 0.04828 1139 -5.326 <.0001
## OCTO - POP2L  -0.13140 0.01926 1139 -6.822 <.0001
## OCTO - POP2H  -0.24076 0.04828 1139 -4.987 <.0001
## POP2L - POP2H -0.10936 0.04980 1139 -2.196 0.0566
##
## temperature = 29:
## contrast      estimate      SE    df t.ratio p.value
## CSB - OCTO    -0.03508 0.02644 1139 -1.327 0.2045
## CSB - POP2L   -0.36773 0.04188 1139 -8.781 <.0001
## CSB - POP2H   -0.47750 0.05872 1139 -8.132 <.0001
## OCTO - POP2L  -0.33265 0.04193 1139 -7.934 <.0001
## OCTO - POP2H  -0.44243 0.05875 1139 -7.530 <.0001
## POP2L - POP2H -0.10977 0.06713 1139 -1.635 0.2045
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: holm method for 6 tests
#
multcomp::cld(multcomp_csb_wolb$lstrends, adj="holm")

## temperature = 18:
## wolbachia day.trend      SE    df lower.CL upper.CL .group
## POP2L          0.0392 0.00557 1139   0.0253   0.0532   1
## CSB            0.0413 0.00558 1139   0.0273   0.0552   1
## POP2H          0.0585 0.00557 1139   0.0445   0.0724  12
## OCTO           0.0736 0.00557 1139   0.0597   0.0876   2
##
## temperature = 25:
## wolbachia day.trend      SE    df lower.CL upper.CL .group
## CSB            0.0499 0.01054 1139   0.0235   0.0762   1

```

```
## OCTO          0.0663 0.01054 1139    0.0399    0.0926    1
## POP2L          0.1977 0.01613 1139    0.1573    0.2380    2
## POP2H          0.3070 0.04711 1139    0.1891    0.4249    2
##
## temperature = 29:
## wolbachia day.trend      SE    df lower.CL upper.CL .group
## CSB          0.1704 0.01864 1139    0.1238    0.2171    1
## OCTO          0.2055 0.01875 1139    0.1586    0.2524    1
## POP2L          0.5381 0.03750 1139    0.4443    0.6320    2
## POP2H          0.6479 0.05568 1139    0.5086    0.7872    2
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 4 estimates
## P value adjustment: holm method for 6 tests
## significance level used: alpha = 0.05
```

## copy number influence on wMelPop and wMelPop2

```
### Octomom copy number influence in wMelPop and wMelPOP2
# filter data and make new columns with genotype and octomom copy number
results_qpcr_pops_proliferation <- results_qpcr_wolb_proliferation %>%
  filter(wolbachia!="CSB" & wolbachia!="OCTO") %>%
  mutate(genotype = wolbachia) %>%
  mutate(genotype = dplyr::recode(genotype,"POP2L" = "CSB", "POP2H" = "CSB", "POPL" = "POP", "POPH" = "POP2H"))
  mutate(copy_number = wolbachia) %>%
  mutate(copy_number = dplyr::recode(copy_number,"POP2L" = "low", "POP2H" = "high", "POPL" = "low", "POPH" = "high"))
  droplevels()

### linear model to test effect of genotype and copy number
## full model
model_prol_pop=lmer(log(normalized_fold_change)~day*genotype*copy_number*temperature +(1|experimental_replicate))

# significance
car::Anova(model_prol_pop)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##
```

	Chisq	Df	Pr(>Chisq)
## day	647.8960	1	< 2.2e-16 ***
## genotype	0.0551	1	0.814335
## copy_number	196.7544	1	< 2.2e-16 ***
## temperature	697.5901	2	< 2.2e-16 ***
## day:genotype	1.3298	1	0.248837
## day:copy_number	10.5288	1	0.001175 **
## genotype:copy_number	4.9308	1	0.026382 *
## day:temperature	777.3056	2	< 2.2e-16 ***
## genotype:temperature	20.1211	2	4.273e-05 ***
## copy_number:temperature	29.6508	2	3.643e-07 ***
## day:genotype:copy_number	1.0041	1	0.316314
## day:genotype:temperature	12.1984	2	0.002245 **
## day:copy_number:temperature	24.3527	2	5.151e-06 ***

```

## genotype:copy_number:temperature      0.1168  2   0.943288
## day:genotype:copy_number:temperature  4.4524  2   0.107938
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## simpler model
# simplified model without interactions that are not significant
model_prol_pop_simpler=lmer(log(normalized_fold_change)~day*genotype*temperature+day*copy_number*temper

# significance of the model
car::Anova(model_prol_pop_simpler)

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##               Chisq Df Pr(>Chisq)
## day           642.0025  1 < 2.2e-16 ***
## genotype        0.0549  1  0.814811
## temperature     692.0672  2 < 2.2e-16 ***
## copy_number     195.7470  1 < 2.2e-16 ***
## day:genotype      2.4554  1  0.117119
## day:temperature   769.1887  2 < 2.2e-16 ***
## genotype:temperature  23.9060  2  6.440e-06 ***
## day:copy_number   10.4260  1  0.001243 **
## temperature:copy_number  29.5641  2  3.804e-07 ***
## day:genotype:temperature  16.9564  2  0.000208 ***
## day:temperature:copy_number  24.0999  2  5.845e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_prol_pop_simpler)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(normalized_fold_change) ~ day * genotype * temperature +
##         day * copy_number * temperature + (1 | experimental_replicate)
## Data: results_qpcr_pops_proliferation
##
## REML criterion at convergence: 2929.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.9388 -0.4840  0.0348  0.6091  3.8776
##
## Random effects:
##  Groups                Name             Variance Std.Dev.
##  experimental_replicate (Intercept) 0.01023  0.1012
##  Residual                      1.04203  1.0208
## Number of obs: 993, groups:  experimental_replicate, 2
##
## Fixed effects:
##               Estimate Std. Error      df t value
## (Intercept)    0.599182   0.159975 17.013799   3.745
## day            0.042684   0.004730 974.001366   9.023
## genotypePOP    0.211249   0.165278 974.000487   1.278

```

```
## temperature25          -0.295543    0.220681 974.038919 -1.339
## temperature29           0.167087    0.222782 974.071302  0.750
## copy_numberhigh         0.436230    0.165278 974.000487  2.639
## day:genotypePOP        -0.001254    0.005473 974.004078 -0.229
## day:temperature25       0.160486    0.015437 974.214803 10.396
## day:temperature29       0.479150    0.033682 974.320767 14.226
## genotypePOP:temperature25 -0.115044    0.253674 974.012734 -0.454
## genotypePOP:temperature29 -0.004556    0.258199 974.045420 -0.018
## day:copy_numberhigh     0.012307    0.005473 974.004078  2.249
## temperature25:copy_numberhigh 0.646072    0.266170 974.053073  2.427
## temperature29:copy_numberhigh -0.176694    0.261369 974.011369 -0.676
## day:genotypePOP:temperature25 -0.073352    0.017820 974.050356 -4.116
## day:genotypePOP:temperature29 -0.002112    0.041422 974.138936 -0.051
## day:temperature25:copy_numberhigh 0.099392    0.034483 974.078022  2.882
## day:temperature29:copy_numberhigh 0.185783    0.046124 974.045051  4.028
##
## Pr(>|t|)
## (Intercept)            0.00161 **
## day                    < 2e-16 ***
## genotypePOP            0.20150
## temperature25          0.18081
## temperature29          0.45344
## copy_numberhigh        0.00844 **
## day:genotypePOP        0.81886
## day:temperature25      < 2e-16 ***
## day:temperature29      < 2e-16 ***
## genotypePOP:temperature25 0.65028
## genotypePOP:temperature29 0.98593
## day:copy_numberhigh     0.02476 *
## temperature25:copy_numberhigh 0.01539 *
## temperature29:copy_numberhigh 0.49918
## day:genotypePOP:temperature25 4.18e-05 ***
## day:genotypePOP:temperature29 0.95934
## day:temperature25:copy_numberhigh 0.00403 **
## day:temperature29:copy_numberhigh 6.06e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 18 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it
```

```
## multiple comparison
```

```
# compare Wolbachia with different Octomom copy number at the different temperatures
```

```
(multcomp_pop_all_temps=emmeans::lstrends(model_prol_pop_simpler, pairwise~copy_number | temperature, v
```

```
## $lstrends
## temperature = 18:
##   copy_number day.trend      SE   df lower.CL upper.CL
##   low          0.0421 0.00386 974    0.0334    0.0507
##   high         0.0544 0.00388 974    0.0456    0.0631
##
## temperature = 25:
##   copy_number day.trend      SE   df lower.CL upper.CL
##   low          0.1659 0.00980 975    0.1439    0.1879
```

```
## high          0.2776 0.03261 974    0.2044    0.3508
##
## temperature = 29:
## copy_number day.trend      SE  df lower.CL upper.CL
## low          0.5202 0.02515 974    0.4637    0.5766
## high         0.7182 0.03829 974    0.6323    0.8042
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
##
## $contrasts
## temperature = 18:
## contrast estimate      SE  df t.ratio p.value
## low - high  -0.0123 0.00547 974  -2.249  0.0248
##
## temperature = 25:
## contrast estimate      SE  df t.ratio p.value
## low - high  -0.1117 0.03405 974  -3.281  0.0011
##
## temperature = 29:
## contrast estimate      SE  df t.ratio p.value
## low - high  -0.1981 0.04580 974  -4.325  <.0001
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
```

```
#
multcomp::cld(multcomp_pop_all_temps$lstrends, adj="holm")
```

```
## temperature = 18:
## copy_number day.trend      SE  df lower.CL upper.CL .group
## low          0.0421 0.00386 974    0.0334    0.0507    1
## high         0.0544 0.00388 974    0.0456    0.0631    2
##
## temperature = 25:
## copy_number day.trend      SE  df lower.CL upper.CL .group
## low          0.1659 0.00980 975    0.1439    0.1879    1
## high         0.2776 0.03261 974    0.2044    0.3508    2
##
## temperature = 29:
## copy_number day.trend      SE  df lower.CL upper.CL .group
## low          0.5202 0.02515 974    0.4637    0.5766    1
## high         0.7182 0.03829 974    0.6323    0.8042    2
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
```

```
# compare differences between temperatures for each Octomom copy number
```

```
(multcomp_pop_temps_each=emmeans::lstrends(model_prol_pop_simpler, consec~ temperature | copy_number, v
```

```
## $lstrends
```

```

## copy_number = low:
## temperature day.trend      SE  df lower.CL upper.CL
## 18          0.0421 0.00386 974   0.0328   0.0513
## 25          0.1659 0.00980 975   0.1424   0.1894
## 29          0.5202 0.02515 974   0.4598   0.5805
##
## copy_number = high:
## temperature day.trend      SE  df lower.CL upper.CL
## 18          0.0544 0.00388 974   0.0451   0.0637
## 25          0.2776 0.03261 974   0.1994   0.3558
## 29          0.7182 0.03829 974   0.6264   0.8101
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
## copy_number = low:
## contrast estimate      SE  df t.ratio p.value
## 25 - 18      0.124 0.0105 975 11.758 <.0001
## 29 - 25      0.354 0.0270 974 13.135 <.0001
##
## copy_number = high:
## contrast estimate      SE  df t.ratio p.value
## 25 - 18      0.223 0.0328 974  6.797 <.0001
## 29 - 25      0.441 0.0503 974  8.762 <.0001
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
## P value adjustment: holm method for 2 tests
#
multcomp::cld(multcomp_pop_temps_each$lstrends, adj="holm")

## copy_number = low:
## temperature day.trend      SE  df lower.CL upper.CL .group
## 18          0.0421 0.00386 974   0.0328   0.0513   1
## 25          0.1659 0.00980 975   0.1424   0.1894   2
## 29          0.5202 0.02515 974   0.4598   0.5805   3
##
## copy_number = high:
## temperature day.trend      SE  df lower.CL upper.CL .group
## 18          0.0544 0.00388 974   0.0451   0.0637   1
## 25          0.2776 0.03261 974   0.1994   0.3558   2
## 29          0.7182 0.03829 974   0.6264   0.8101   3
##
## Results are averaged over the levels of: genotype
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05

```



```

# influence of genotype
(multcomp_pop_all_genotype=emmeans::lstrends(model_prol_pop_simpler, pairwise~genotype | temperature, v

## $lstrends
## temperature = 18:
## genotype day.trend      SE  df lower.CL upper.CL
## CSB          0.0488 0.00386 974   0.0402   0.0575
## POP          0.0476 0.00388 974   0.0389   0.0563
##
## temperature = 25:
## genotype day.trend      SE  df lower.CL upper.CL
## CSB          0.2590 0.01964 974   0.2149   0.3031
## POP          0.1844 0.01838 974   0.1432   0.2257
##
## temperature = 29:
## genotype day.trend      SE  df lower.CL upper.CL
## CSB          0.6209 0.03135 974   0.5505   0.6912
## POP          0.6175 0.03017 974   0.5498   0.6852
##
## Results are averaged over the levels of: copy_number
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
##
## $contrasts
## temperature = 18:
## contrast estimate      SE  df t.ratio p.value
## CSB - POP  0.00125 0.00547 974 0.229   0.8189
##
## temperature = 25:
## contrast estimate      SE  df t.ratio p.value
## CSB - POP  0.07461 0.01696 974 4.399   <.0001
##
## temperature = 29:
## contrast estimate      SE  df t.ratio p.value
## CSB - POP  0.00337 0.04106 974 0.082   0.9347
##
## Results are averaged over the levels of: copy_number
## Degrees-of-freedom method: kenward-roger
#
multcomp::cld(multcomp_pop_all_genotype$lstrends, adj="holm")

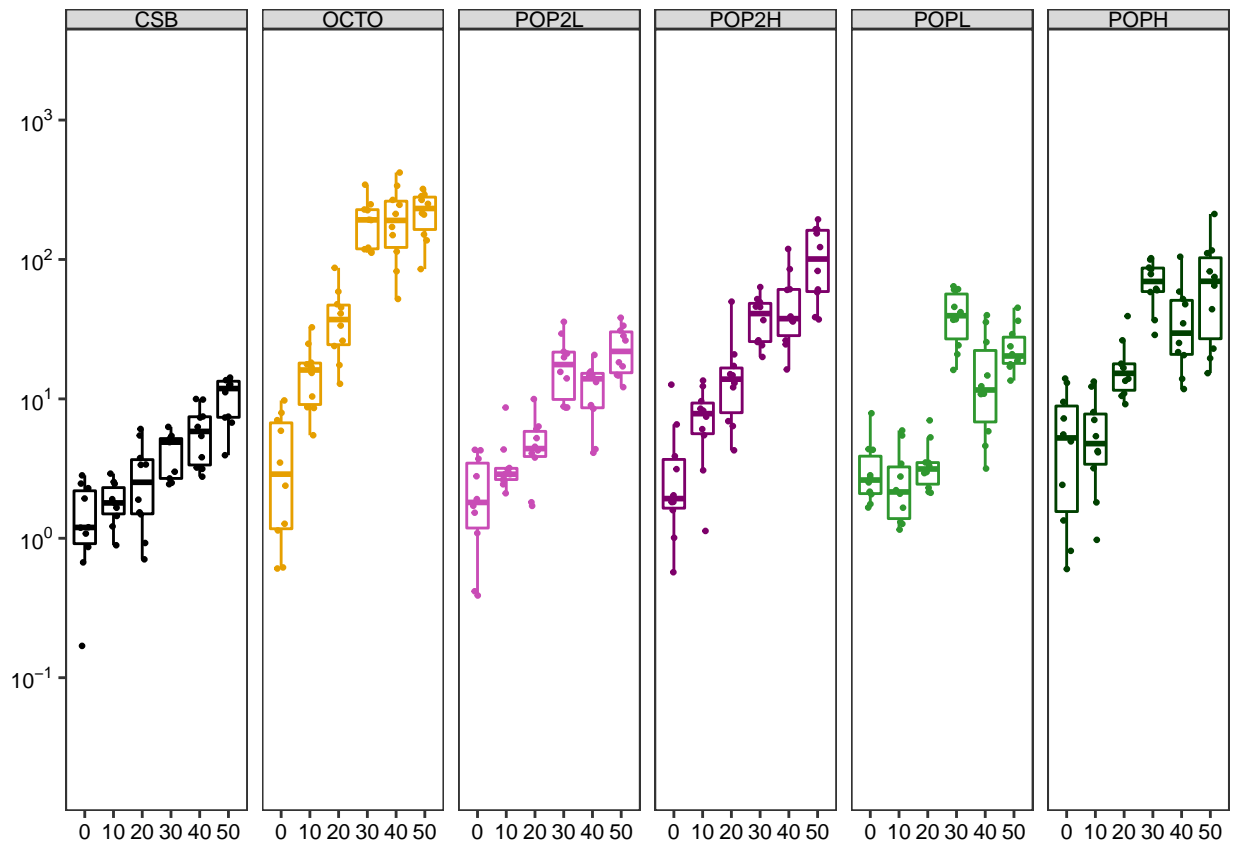
## temperature = 18:
## genotype day.trend      SE  df lower.CL upper.CL .group
## POP          0.0476 0.00388 974   0.0389   0.0563   1
## CSB          0.0488 0.00386 974   0.0402   0.0575   1
##
## temperature = 25:
## genotype day.trend      SE  df lower.CL upper.CL .group
## POP          0.1844 0.01838 974   0.1432   0.2257   1
## CSB          0.2590 0.01964 974   0.2149   0.3031   2
##
## temperature = 29:
## genotype day.trend      SE  df lower.CL upper.CL .group

```

```
## POP          0.6175 0.03017 974    0.5498    0.6852  1
## CSB          0.6209 0.03135 974    0.5505    0.6912  1
##
## Results are averaged over the levels of: copy_number
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## significance level used: alpha = 0.05
```

Fig 3A-C

```
### plots
## 18°C
(fig3a=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="18", experimental_replicate=
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.02, 2.5e3),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","10","20","30","40","50"))+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))))
```

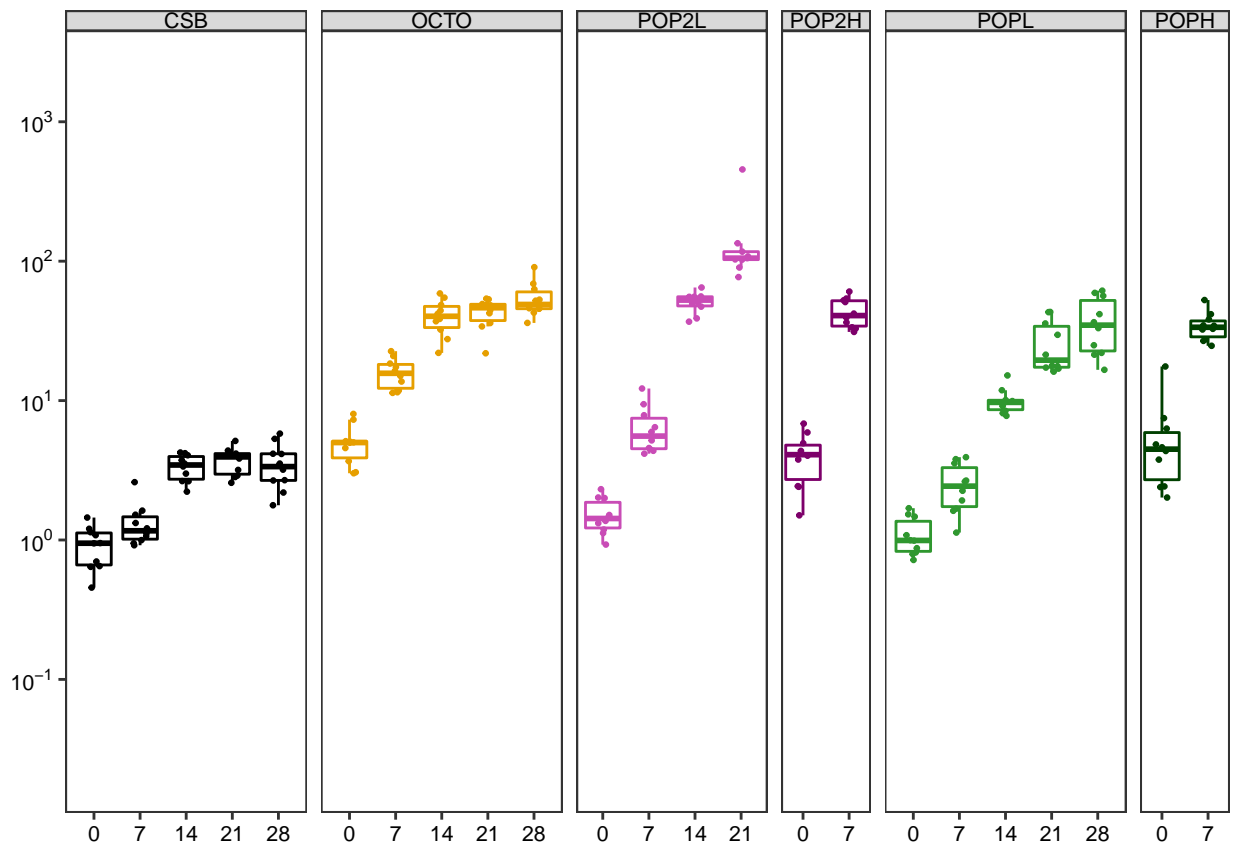


```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 3A.pdf", plot=fig3a, device="pdf", dpi=300, units="in", width=700/96,
}

## 25°C
(fig3b=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="25", experimental_replicate=
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(.~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.02, 2.5e3),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,5,1), labels=c("0","7","14","21","28"))+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))))

```

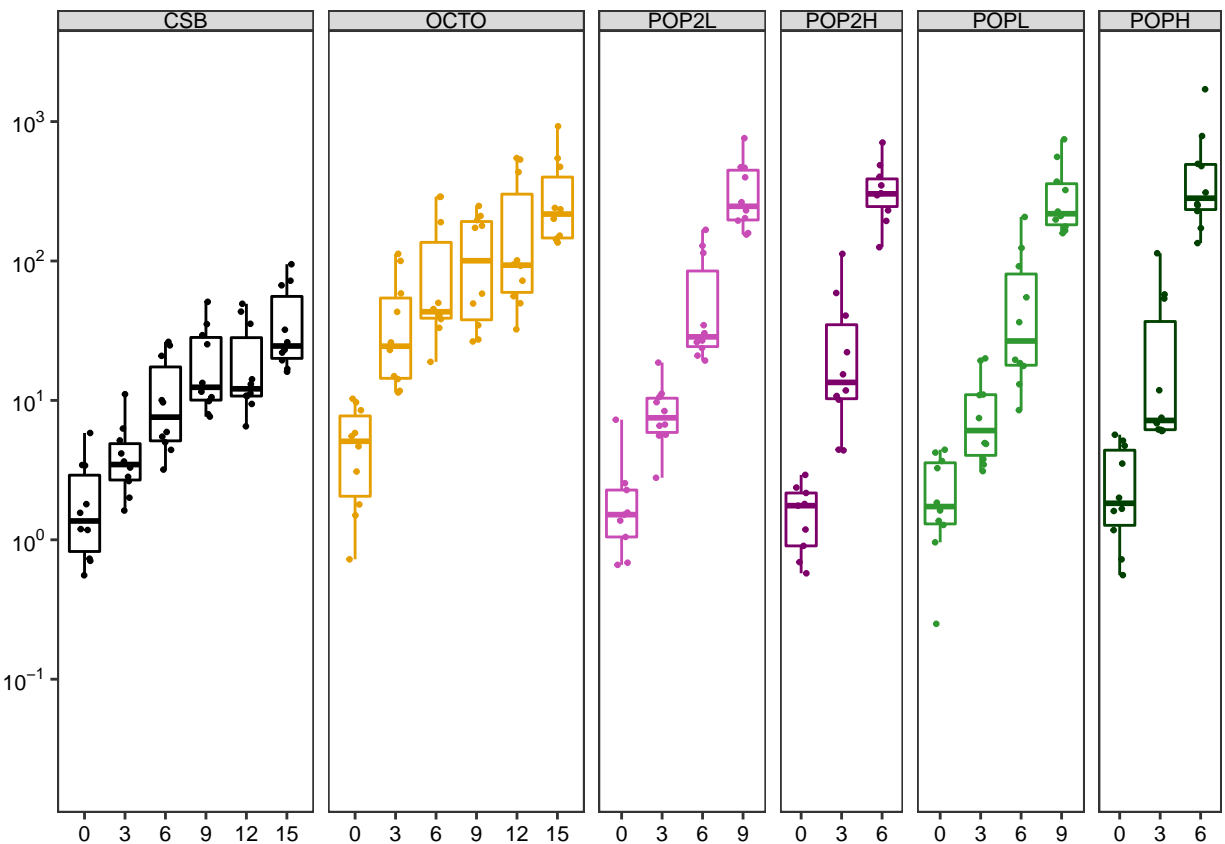


```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 3B.pdf", plot=fig3b, device="pdf", dpi=300, units="in", width=700/96,
}

```

```
## 29°C
(fig3c=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="29", experimental_replicate=
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.02, 2.5e3),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","3","6","9","12", "15"))+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 3C.pdf", plot=fig3c, device="pdf", dpi=300, units="in", width=700/96,
}
```

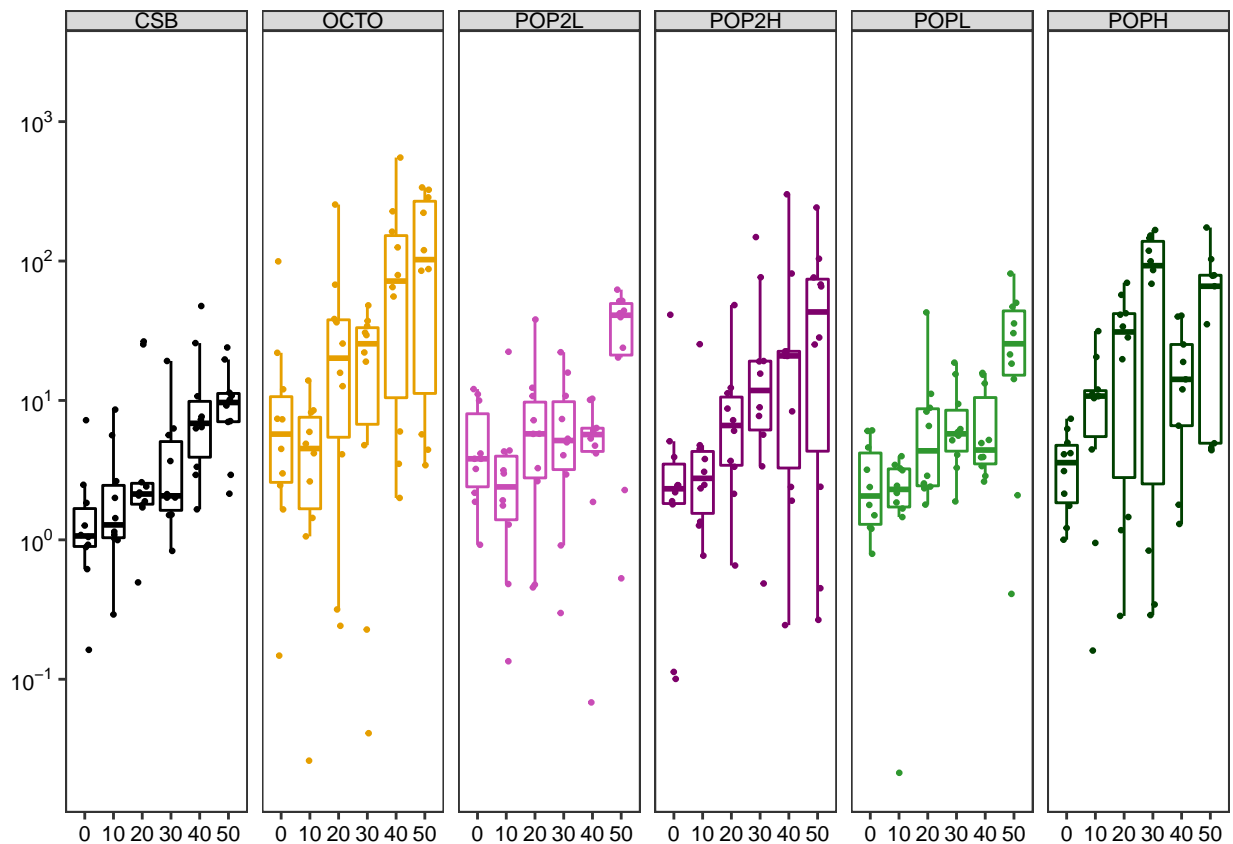
## S10A-C Fig

```
## S10A Fig
(s9afig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="18", experimental_replicate=
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
```

```

geom_jitter(width=0.15, size=0.5)+
facet_grid(~wolbachia, scales="free_x", space="free_x")+
theme_figs()+ theme(legend.position="none")+
scale_y_log10(name="", limits=c(0.02, 2.5e3),
              labels=trans_format('log10',math_format(10^.x)),
              breaks=c(1e-1,1e0,1e1,1e2,1e3))+
scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","10","20","30","40","50"))+
scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S10A Fig.pdf", plot=s10afig, device="pdf", dpi=300, units="in", width=700, height=700)
}

```

```

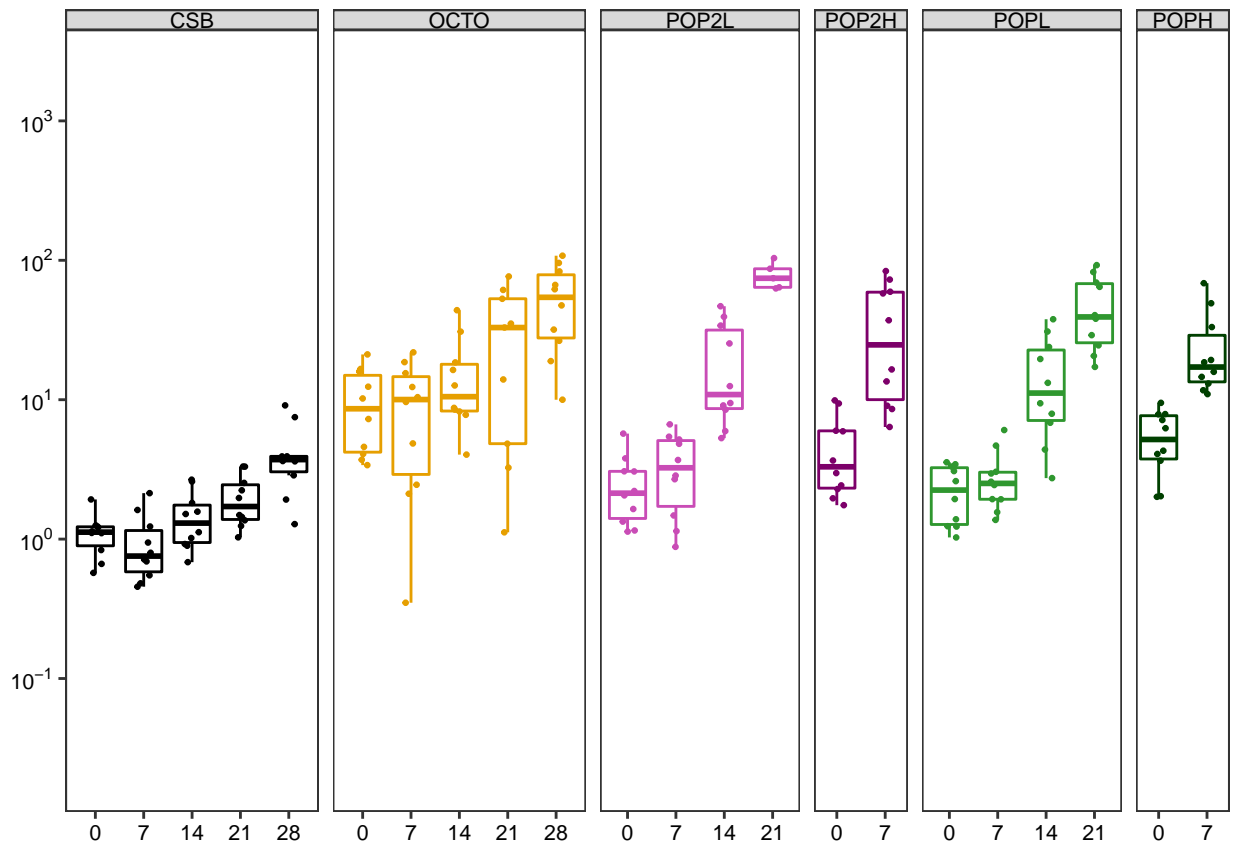
## S10B Fig
(s10bfig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="25", experimental_replicate==1),
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.02, 2.5e3),
                labels=trans_format('log10',math_format(10^.x)),
                breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))

```

```
scale_x_continuous(name="", breaks=seq(1,5,1), labels=c("0","7","14","21","28"))+
scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")))
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```

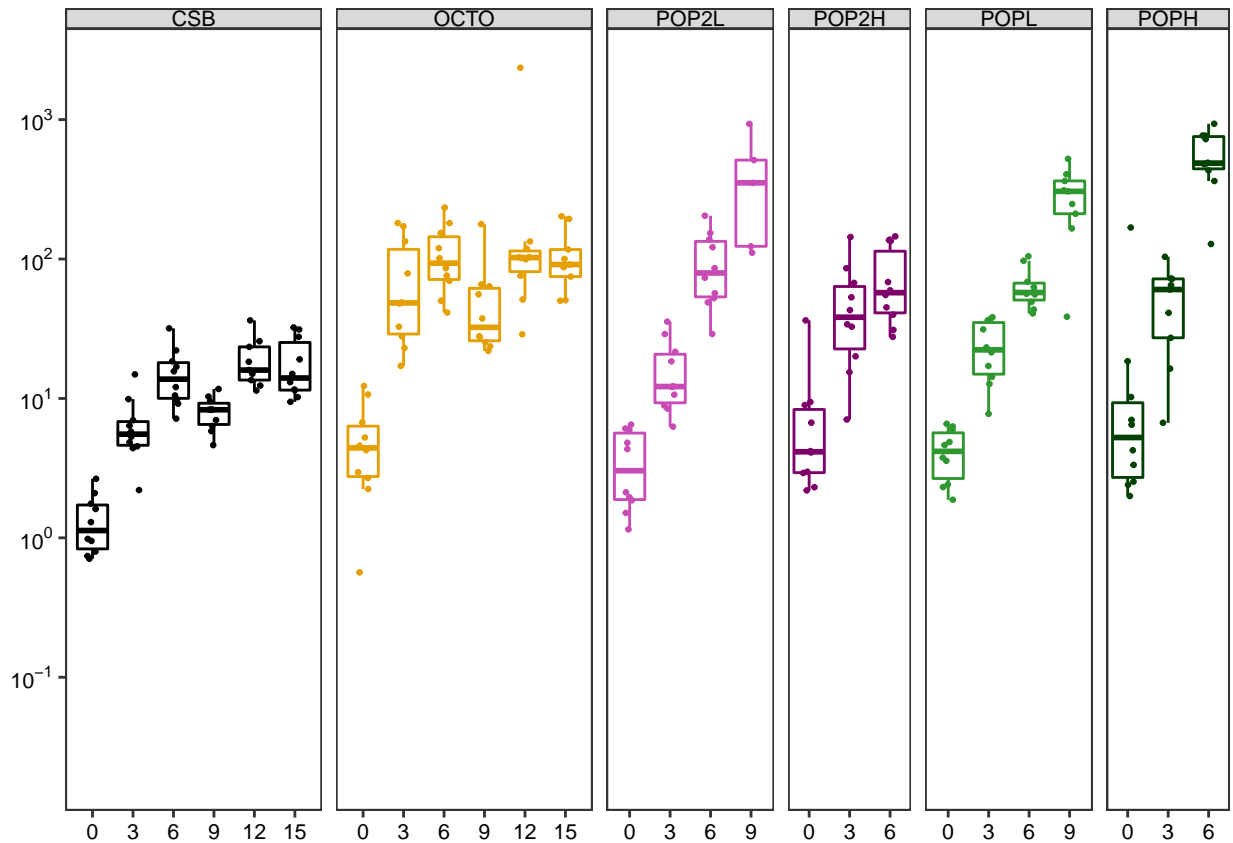
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S10B Fig.pdf", plot=s10bfig, device="pdf", dpi=300, units="in", width=700, height=700)
}
```

```
## S10C Fig
```

```
(s10cfig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, temperature=="29", experimental_replicate==1),
  aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.02, 2.5e3),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","3","6", "9", "12","15"))+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S10C Fig.pdf", plot=s10cfig, device="pdf", dpi=300, units="in", width=700,
}

```

## S11 Fig | Wolbachia titers day of eclosion

```
### the model
## model by Wolbachia titers
model_wolb_titer_day0=lmer(log(normalized_fold_change)~wolbachia+(1|experimental_replicate), data=dplyr

# significance of the model
car::Anova(model_wolb_titer_day0)

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(normalized_fold_change)
##           Chisq Df Pr(>Chisq)
## wolbachia 98.225  5 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_wolb_titer_day0, corr=FALSE)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [

```

```
## lmerModLmerTest]
## Formula: log(normalized_fold_change) ~ wolbachia + (1 | experimental_replicate)
## Data: dplyr::filter(results_qpcr_wolb_proliferation, time_point == "1")
##
## REML criterion at convergence: 912.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.1997 -0.5322 -0.0612  0.5774  4.2350
##
## Random effects:
## Groups              Name                Variance Std.Dev.
## experimental_replicate (Intercept) 0.05903  0.2430
## Residual                    0.72343  0.8505
## Number of obs: 358, groups: experimental_replicate, 2
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    0.1406    0.2039    1.7373  0.690    0.571
## wolbachiaOCTO    1.3529    0.1553   351.0000  8.712 < 2e-16 ***
## wolbachiaPOP2L    0.6348    0.1559   351.0011  4.070 5.80e-05 ***
## wolbachiaPOP2H    0.8886    0.1559   351.0011  5.698 2.57e-08 ***
## wolbachiaPOPL    0.6222    0.1553   351.0000  4.007 7.52e-05 ***
## wolbachiaPOPH    1.2167    0.1553   351.0000  7.835 5.64e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# multiple comparison
(multiple_wolb_titer_day0=emmeans::emmeans(model_wolb_titer_day0, list(pairwise~wolbachia), adj="holm"))

## $`emmeans of wolbachia`
##   wolbachia emmean    SE    df lower.CL upper.CL
## CSB         0.141 0.204 1.74    -2.78    3.06
## OCTO        1.494 0.204 1.74    -1.42    4.41
## POP2L        0.775 0.204 1.75    -2.09    3.64
## POP2H        1.029 0.204 1.75    -1.83    3.89
## POPL         0.763 0.204 1.74    -2.15    3.68
## POPH         1.357 0.204 1.74    -1.56    4.27
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 6 estimates
##
## $`pairwise differences of wolbachia`
## contrast      estimate    SE    df t.ratio p.value
## CSB - OCTO     -1.3529 0.155 351  -8.712 <.0001
## CSB - POP2L     -0.6348 0.156 351  -4.070 0.0006
## CSB - POP2H     -0.8886 0.156 351  -5.698 <.0001
## CSB - POPL      -0.6222 0.155 351  -4.007 0.0007
## CSB - POPH      -1.2167 0.155 351  -7.835 <.0001
## OCTO - POP2L     0.7181 0.156 351   4.605 0.0001
## OCTO - POP2H     0.4643 0.156 351   2.977 0.0187
## OCTO - POPL      0.7307 0.155 351   4.706 <.0001
## OCTO - POPH      0.1362 0.155 351   0.877 0.7620
```



```

## POP2L - POP2H -0.2538 0.157 351 -1.621 0.3539
## POP2L - POPL 0.0126 0.156 351 0.081 0.9357
## POP2L - POPH -0.5819 0.156 351 -3.731 0.0016
## POP2H - POPL 0.2664 0.156 351 1.708 0.3539
## POP2H - POPH -0.3281 0.156 351 -2.104 0.1805
## POPL - POPH -0.5945 0.155 351 -3.828 0.0012
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 15 tests

#
multcomp::cld(multiple_wolb_titer_day0$emmeans of wolbachia`, adj="holm")

## wolbachia emmean SE df lower.CL upper.CL .group
## CSB 0.141 0.204 1.74 -2.78 3.06 1
## POPL 0.763 0.204 1.74 -2.15 3.68 2
## POP2L 0.775 0.204 1.75 -2.09 3.64 2
## POP2H 1.029 0.204 1.75 -1.83 3.89 23
## POPH 1.357 0.204 1.74 -1.56 4.27 34
## OCTO 1.494 0.204 1.74 -1.42 4.41 4
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 6 estimates
## P value adjustment: holm method for 15 tests
## significance level used: alpha = 0.05

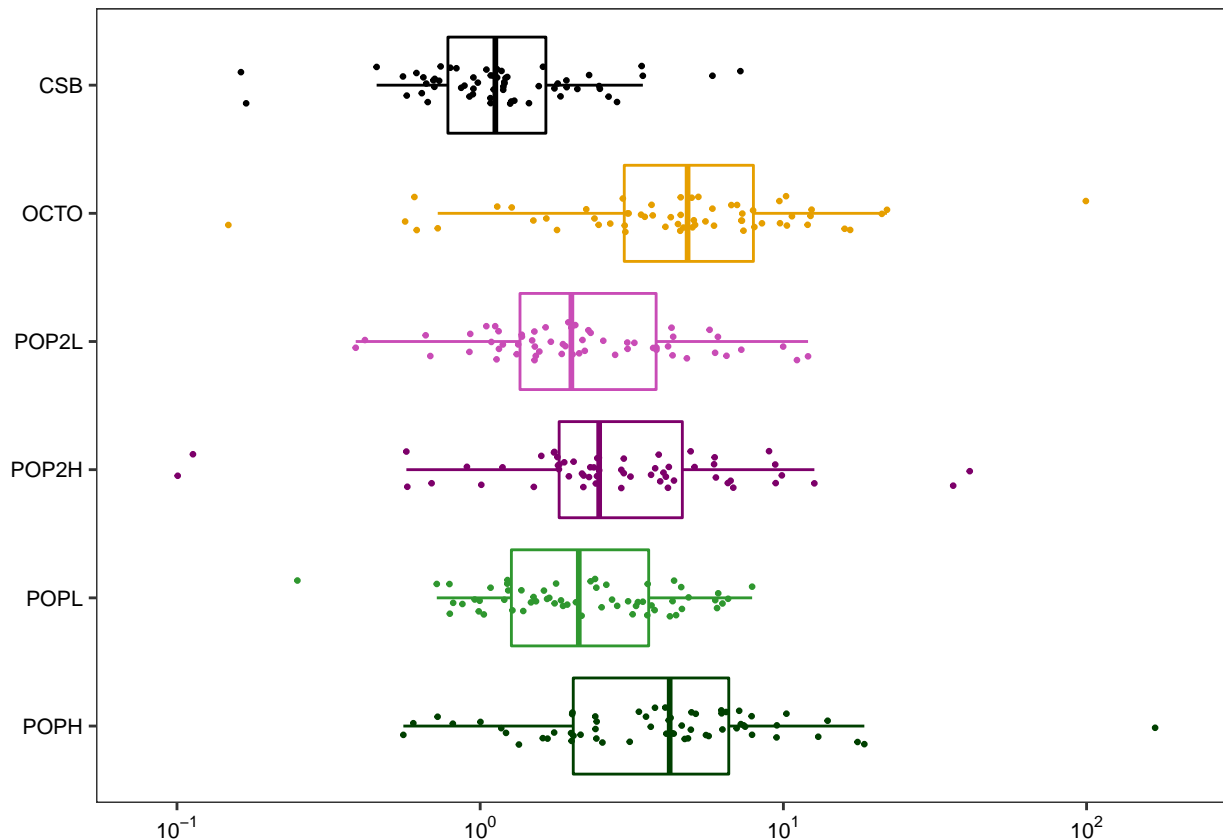
## extract coefficients from full model
(time_0_levels <- as.data.frame(multiple_wolb_titer_day0$emmeans) %>%
  mutate(mean = exp(emmean)) %>%
  mutate(lower_db = exp(lower.CL)) %>%
  mutate(upper_db = exp(upper.CL)) %>%
  mutate(wolbachia2 = wolbachia) %>%
  mutate(wolbachia = dplyr::recode(wolbachia, "CSB" = "wMelCS_b", "POP2L" = "wMelPop2-Low", "POP2H" = "wMelPop2-High", "POPL" = "wMelPop2-Low", "POPH" = "wMelPop2-High")))

## wolbachia emmean SE df lower.CL upper.CL mean
## 1 wMelCS_b 0.1406498 0.2038917 1.737288 -2.775868 3.057168 1.151021
## 2 wMelOctoless 1.4935425 0.2038917 1.737288 -1.422976 4.410061 4.452842
## 3 wMelPop2-Low 0.7754182 0.2043940 1.754431 -2.088465 3.639301 2.171500
## 4 wMelPop2-High 1.0292248 0.2043940 1.754431 -1.834658 3.893108 2.798895
## 5 wMelPop-Low 0.7628228 0.2038917 1.737288 -2.153695 3.679341 2.144321
## 6 wMelPop-High 1.3573303 0.2038917 1.737288 -1.559188 4.273848 3.885806
## lower_db upper_db wolbachia2
## 1 0.06229536 21.26724 CSB
## 2 0.24099585 82.27445 OCTO
## 3 0.12387715 38.06523 POP2L
## 4 0.15966806 49.06314 POP2H
## 5 0.11605451 39.62027 POPL
## 6 0.21030682 71.79741 POPH

### plot
(s1fig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation, time_point=="1"),
  aes(x=reorder(wolbachia, desc(wolbachia)), y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(outlier.shape=NA)+

```

```
geom_jitter(width=0.15, size=0.5)+
theme_figs()+ theme(legend.position="none")+
scale_y_log10(name="", limits=c(0.08,200),
              labels=trans_format('log10',math_format(10^.x)),
              breaks=c(1e-1,1e0,1e1,1e2))+
scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
coord_flip())
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S11 Fig.pdf", plot=s11fig, device="pdf", dpi=300, units="in", width=300/96)
}
```

## S12 Fig Phenotypes wMelPop2 and wMelPop controlled Octomom copy number

### S12A | Control Octomom copy number

```
### S16 Data
s16data=read.csv("../original_data/S16 Data.csv", stringsAsFactors = TRUE)

### determining octomom copy number per experimental replicate
results_qpcr_octomom_csb_and_pop = s16data %>%
  dplyr::group_by(experimental_replicate)%>%
```

```

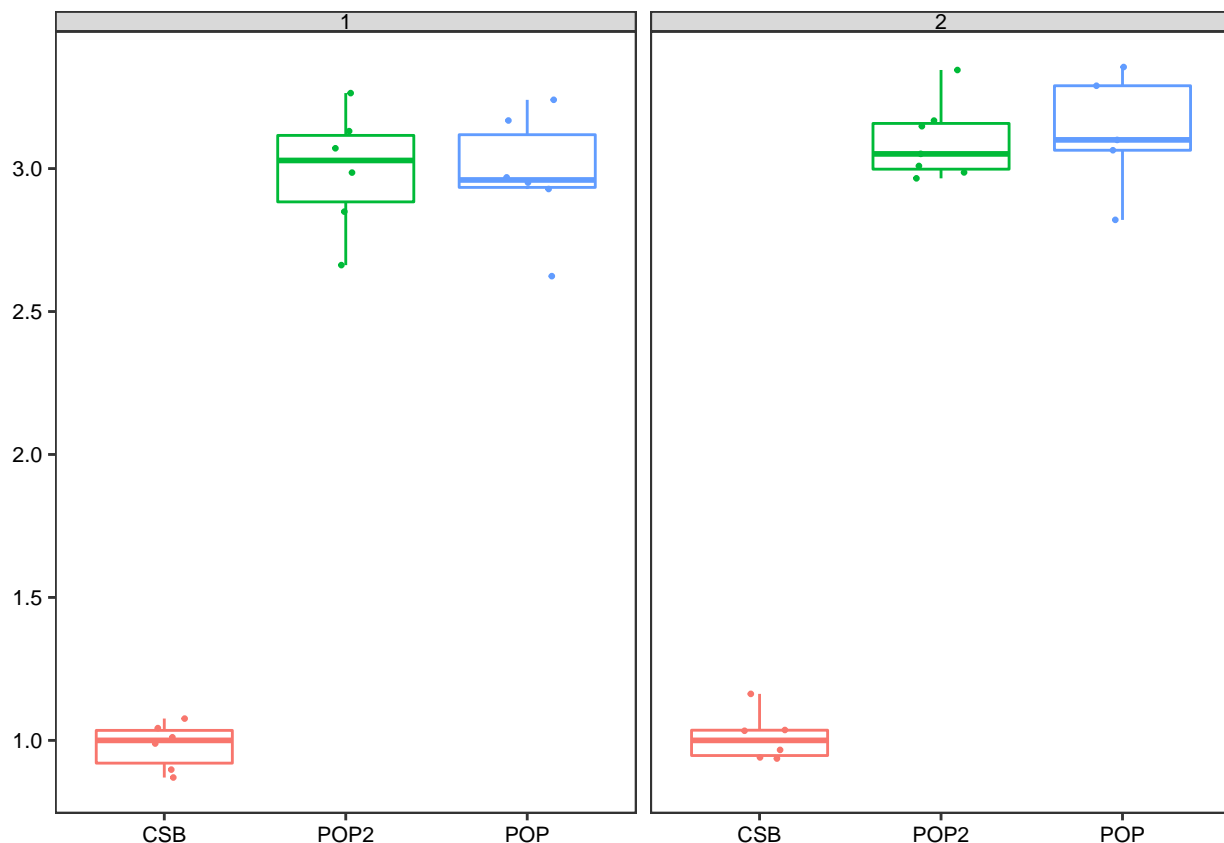
do(qpcr.pfafl(data=., calibrator_sample="CSB", reference_gene="WSP", efficiency_ref=2, efficiency_ta

## adding new variables
results_qpcr_octomom_csb_and_pop=results_qpcr_octomom_csb_and_pop%>%
  tidyr::separate(sample, c("wolbachia", "replicate"), "_", remove=FALSE)

# converting variable to factor
results_qpcr_octomom_csb_and_pop$wolbachia=factor(results_qpcr_octomom_csb_and_pop$wolbachia, levels=c(

### plot
ggplot(results_qpcr_octomom_csb_and_pop, aes(x=wolbachia, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~experimental_replicate)+
  theme_figs() + theme(legend.position="none")

```



```

### analysis
## first replicate
model_copy_number_csb_vs_pop_rep1=lm(normalized_fold_change~wolbachia, data=results_qpcr_octomom_csb_and

# significance of the model
car::Anova(model_copy_number_csb_vs_pop_rep1)

## Anova Table (Type II tests)
##
## Response: normalized_fold_change

```

```
##           Sum Sq Df F value    Pr(>F)
## wolbachia 16.0952  2  243.62 3.663e-12 ***
## Residuals  0.4955 15
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_copy_number_csb_vs_pop_rep1)

##
## Call:
## lm(formula = normalized_fold_change ~ wolbachia, data = results_qpcr_octomom_csb_and_pop,
##     subset = experimental_replicate == "1")
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35639 -0.07559 -0.00006  0.09064  0.27011
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.9812     0.0742   13.22 1.13e-09 ***
## wolbachiaPOP2    2.0128     0.1049   19.18 5.78e-12 ***
## wolbachiaPOP     1.9990     0.1049   19.05 6.38e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1818 on 15 degrees of freedom
## Multiple R-squared:  0.9701, Adjusted R-squared:  0.9662
## F-statistic: 243.6 on 2 and 15 DF,  p-value: 3.663e-12

# multiple comparison
(multiple_copy_number_csb_vs_pop_rep1=emmeans::emmeans(model_copy_number_csb_vs_pop_rep1, list(pairwise

## $`emmeans of wolbachia`
##   wolbachia emmean      SE df lower.CL upper.CL
##   CSB        0.981 0.0742 15    0.781    1.18
##   POP2       2.994 0.0742 15    2.794    3.19
##   POP        2.980 0.0742 15    2.780    3.18
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $`pairwise differences of wolbachia`
## contrast estimate      SE df t.ratio p.value
## CSB - POP2  -2.0128 0.105 15 -19.181 <.0001
## CSB - POP   -1.9990 0.105 15 -19.051 <.0001
## POP2 - POP    0.0137 0.105 15  0.131 0.8976
##
## P value adjustment: holm method for 3 tests

#
multcomp::cld(multiple_copy_number_csb_vs_pop_rep1$`emmeans of wolbachia`, adj="holm")

##   wolbachia emmean      SE df lower.CL upper.CL .group
##   CSB        0.981 0.0742 15    0.781    1.18    1
##   POP        2.980 0.0742 15    2.780    3.18    2
```

```

## POP2          2.994 0.0742 15      2.794      3.19      2
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05

## second replicate
model_copy_number_csb_vs_pop_rep2=lm(fold_change~wolbachia, data=results_qpcr_octomom_csb_and_pop, subset=experimental_replicate=="2")

# significance of the model
car::Anova(model_copy_number_csb_vs_pop_rep2)

## Anova Table (Type II tests)
##
## Response: fold_change
##           Sum Sq Df F value    Pr(>F)
## wolbachia 18.3098  2   409.3 8.195e-14 ***
## Residuals  0.3355 15
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_copy_number_csb_vs_pop_rep2)

##
## Call:
## lm(formula = fold_change ~ wolbachia, data = results_qpcr_octomom_csb_and_pop,
##     subset = experimental_replicate == "2")
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31182 -0.07669 -0.03597  0.06794  0.25377
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.03363    0.06106   16.93 3.48e-11 ***
## wolbachiaPOP2  2.12669    0.08321   25.56 8.80e-14 ***
## wolbachiaPOP   2.15706    0.09056   23.82 2.48e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1496 on 15 degrees of freedom
## Multiple R-squared:  0.982, Adjusted R-squared:  0.9796
## F-statistic: 409.3 on 2 and 15 DF, p-value: 8.195e-14

## multiple comparison
(multiple_copy_number_csb_vs_pop_rep2=emmeans::emmeans(model_copy_number_csb_vs_pop_rep2, list(pairwise~wolbachia)))

## $`emmeans of wolbachia`
##      wolbachia emmean      SE df lower.CL upper.CL
## CSB           1.03 0.0611 15      0.869      1.20
## POP2          3.16 0.0565 15      3.008      3.31
## POP           3.19 0.0669 15      3.011      3.37
##
## Confidence level used: 0.95

```

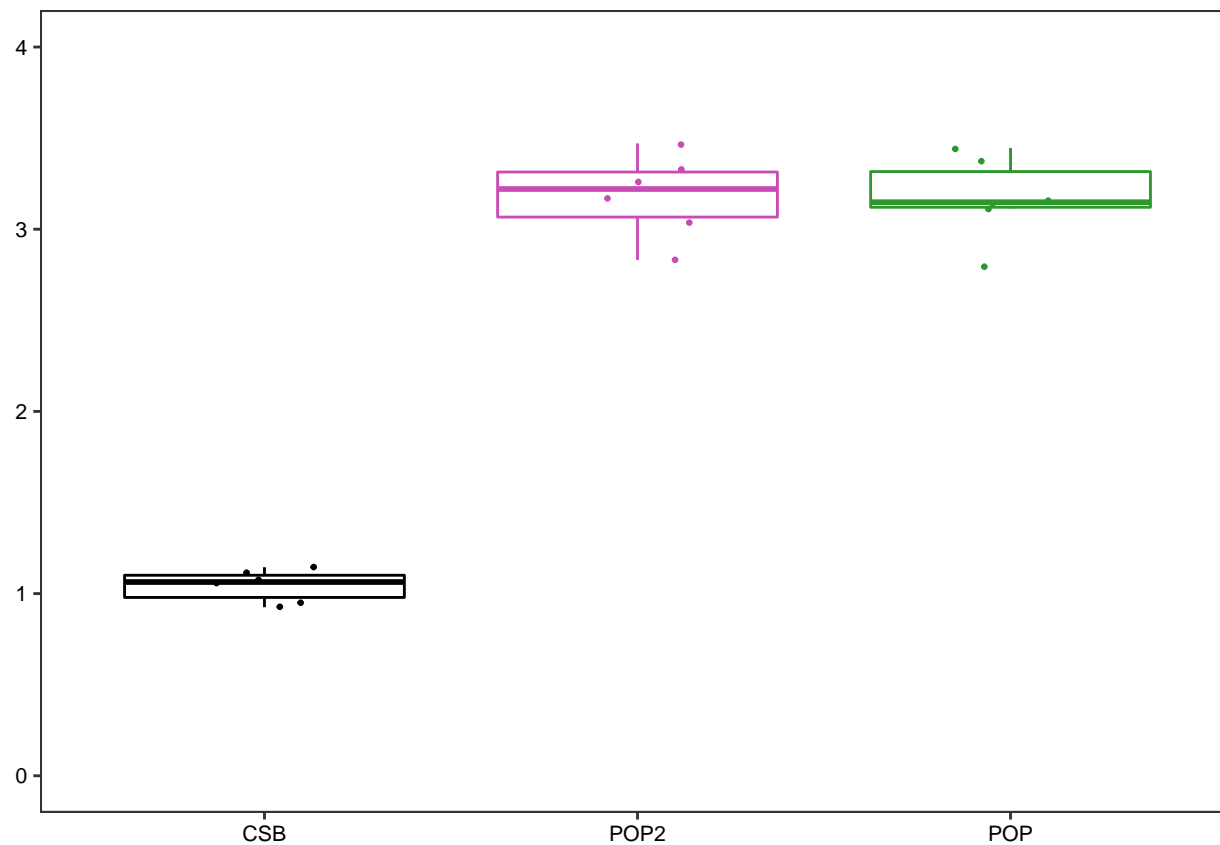
```

## Conf-level adjustment: bonferroni method for 3 estimates
##
## $`pairwise differences of wolbachia`
## contrast estimate SE df t.ratio p.value
## CSB - POP2 -2.1267 0.0832 15 -25.559 <.0001
## CSB - POP -2.1571 0.0906 15 -23.819 <.0001
## POP2 - POP -0.0304 0.0876 15 -0.347 0.7336
##
## P value adjustment: holm method for 3 tests
#
multcomp::cld(multiple_copy_number_csb_vs_pop_rep2$`emmeans of wolbachia`, adj="holm")

## wolbachia emmean SE df lower.CL upper.CL .group
## CSB 1.03 0.0611 15 0.869 1.20 1
## POP2 3.16 0.0565 15 3.008 3.31 2
## POP 3.19 0.0669 15 3.011 3.37 2
##
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05

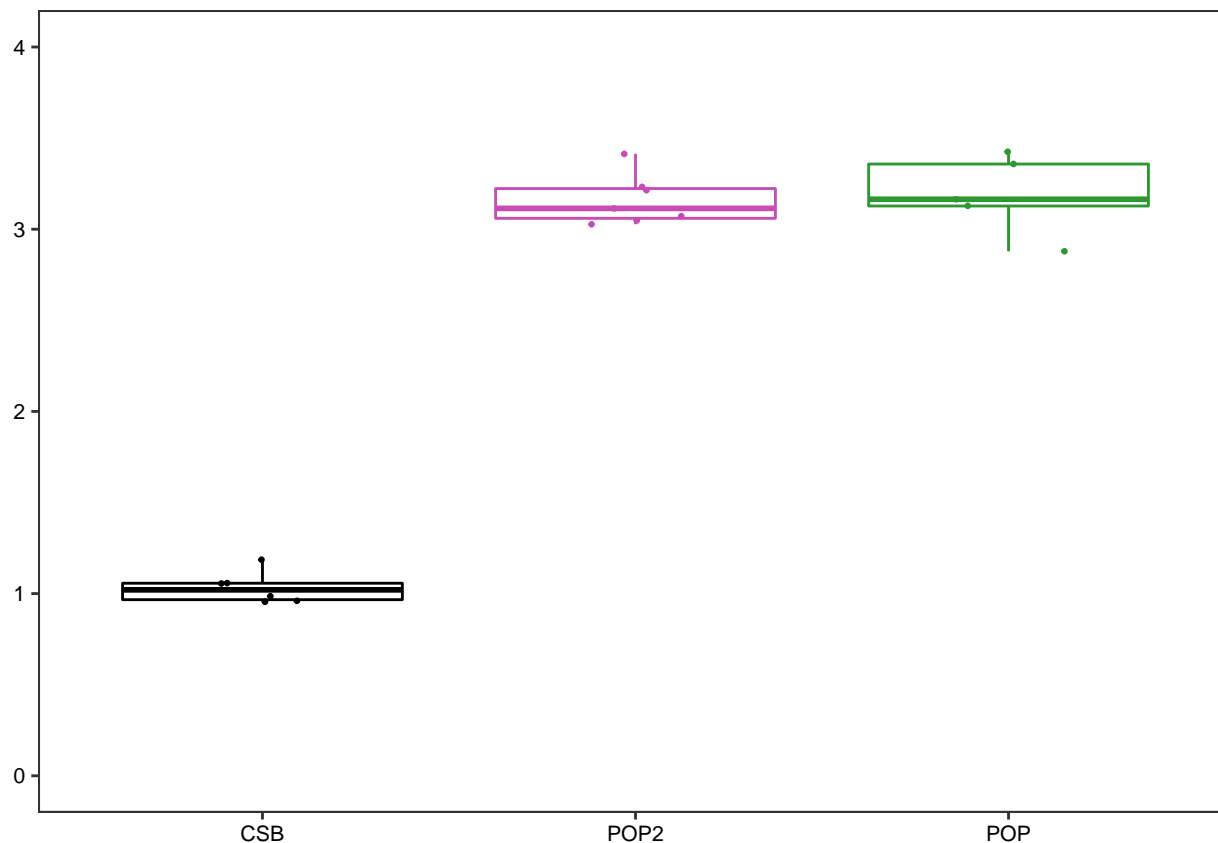
### plots
## replicate 1
(s12a1fig=ggplot(dplyr::filter(results_qpcr_octomom_csb_and_pop,
                               experimental_replicate=="1"),
                 aes(x=wolbachia, y=fold_change, color=wolbachia))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  theme_figs() + theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  scale_y_continuous(limits=c(0,4), breaks=seq(0,4,1)))

```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12A1 Fig.pdf", plot=s12a1fig, device="pdf", dpi=300, units="in", width=10, height=10)
}

## replicate 2
(s12a2fig=ggplot(dplyr::filter(results_qpcr_octomom_csb_and_pop,
                              experimental_replicate=="2"),
                 aes(x=wolbachia, y=fold_change, color=wolbachia))+
  geom_boxplot(outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  theme_figs() + theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  scale_y_continuous(limits=c(0,4), breaks=seq(0,4,1)))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12a2 Fig.pdf", plot=s12a2fig, device="pdf", dpi=300, units="in", width=10, height=10)
}
```

## S12B | Proliferation wMelPop2 and wMelPop

```
### S17 Data
s17data=read.csv("../original_data/S17 Data.csv", stringsAsFactors = TRUE)

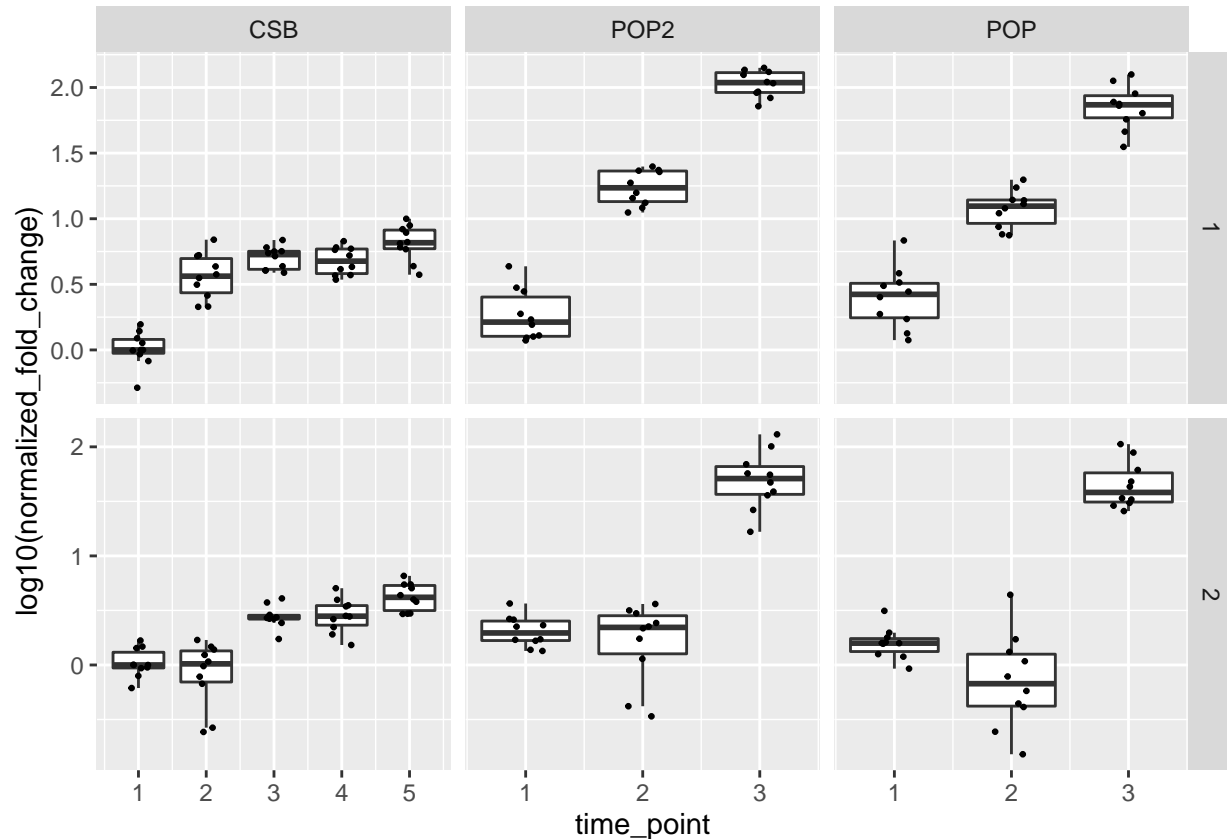
### wolbachia proliferation dynamics
results_qpcr_wolb_proliferation_csb_and_pop=s17data%>%
  dplyr::group_by(experimental_replicate, plate)%>%
  do(qpcr.pfafl(data=., calibrator_sample="CSB_A", reference_gene="RPL32", efficiency_ref=2, efficiency=1))

## adding new variables
results_qpcr_wolb_proliferation_csb_and_pop=results_qpcr_wolb_proliferation_csb_and_pop%>%
  tidyr::separate(sample, c("wolbachia", "time_point", "replicate"), "_", remove=FALSE)%>%
  filter(wolbachia!="CSB-REF")

# converting variables to factors or continuous variables
results_qpcr_wolb_proliferation_csb_and_pop$wolbachia=factor(results_qpcr_wolb_proliferation_csb_and_pop$wolbachia)
#
results_qpcr_wolb_proliferation_csb_and_pop$time_point=as.numeric(as.factor(results_qpcr_wolb_proliferation_csb_and_pop$time_point))
```



```
### plot
ggplot(results_qpcr_wolb_proliferation_csb_and_pop, aes(x=time_point, y=log10(normalized_fold_change))) +
  geom_boxplot(outlier.shape=NA, aes(group=time_point)) +
  geom_jitter(width=0.15, size=0.5) +
  facet_grid(experimental_replicate~wolbachia, scales="free")
```



```
### analysis
model_exponential_wolb_proliferation_csb_and_pop=lmer(log(fold_change)~time_point*wolbachia+(1|experimenter))

# significance of the model
car::Anova(model_exponential_wolb_proliferation_csb_and_pop)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: log(fold_change)
##              Chisq Df Pr(>Chisq)
## time_point      309.97  1 < 2.2e-16 ***
## wolbachia        293.87  2 < 2.2e-16 ***
## time_point:wolbachia 186.18  2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# summary
summary(model_exponential_wolb_proliferation_csb_and_pop, corr=FALSE)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
```

```

## lmerModLmerTest]
## Formula:
## log(fold_change) ~ time_point * wolbachia + (1 | experimental_replicate)
## Data: results_qpcr_wolb_proliferation_csb_and_pop
##
## REML criterion at convergence: 510.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.4313 -0.3876  0.1676  0.6107  1.9222
##
## Random effects:
## Groups              Name             Variance Std.Dev.
## experimental_replicate (Intercept) 0.3481    0.5900
## Residual                0.5544    0.7446
## Number of obs: 220, groups: experimental_replicate, 2
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   -0.20948   0.45228   1.34175  -0.463 0.707171
## time_point      0.40017   0.05265  213.00000   7.601 9.26e-13 ***
## wolbachiaPOP2   -1.19598   0.30850  213.00000  -3.877 0.000141 ***
## wolbachiaPOP    -1.18375   0.30850  213.00000  -3.837 0.000164 ***
## time_point:wolbachiaPOP2  1.41269   0.12897  213.00000  10.954 < 2e-16 ***
## time_point:wolbachiaPOP   1.27004   0.12897  213.00000   9.848 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## multiple comparison
(multiple_exponential_wolb_proliferation_csb_and_pop=emmeans::lstrends(model_exponential_wolb_prolifera

## $lstrends
## wolbachia time_point.trend      SE df lower.CL upper.CL
## CSB              0.40 0.0527 213    0.273    0.527
## POP2             1.81 0.1177 213    1.529    2.097
## POP              1.67 0.1177 213    1.386    1.954
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
## contrast estimate      SE df t.ratio p.value
## CSB - POP2   -1.413 0.129 213 -10.954 <.0001
## CSB - POP    -1.270 0.129 213  -9.848 <.0001
## POP2 - POP    0.143 0.166 213   0.857 0.3925
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: holm method for 3 tests

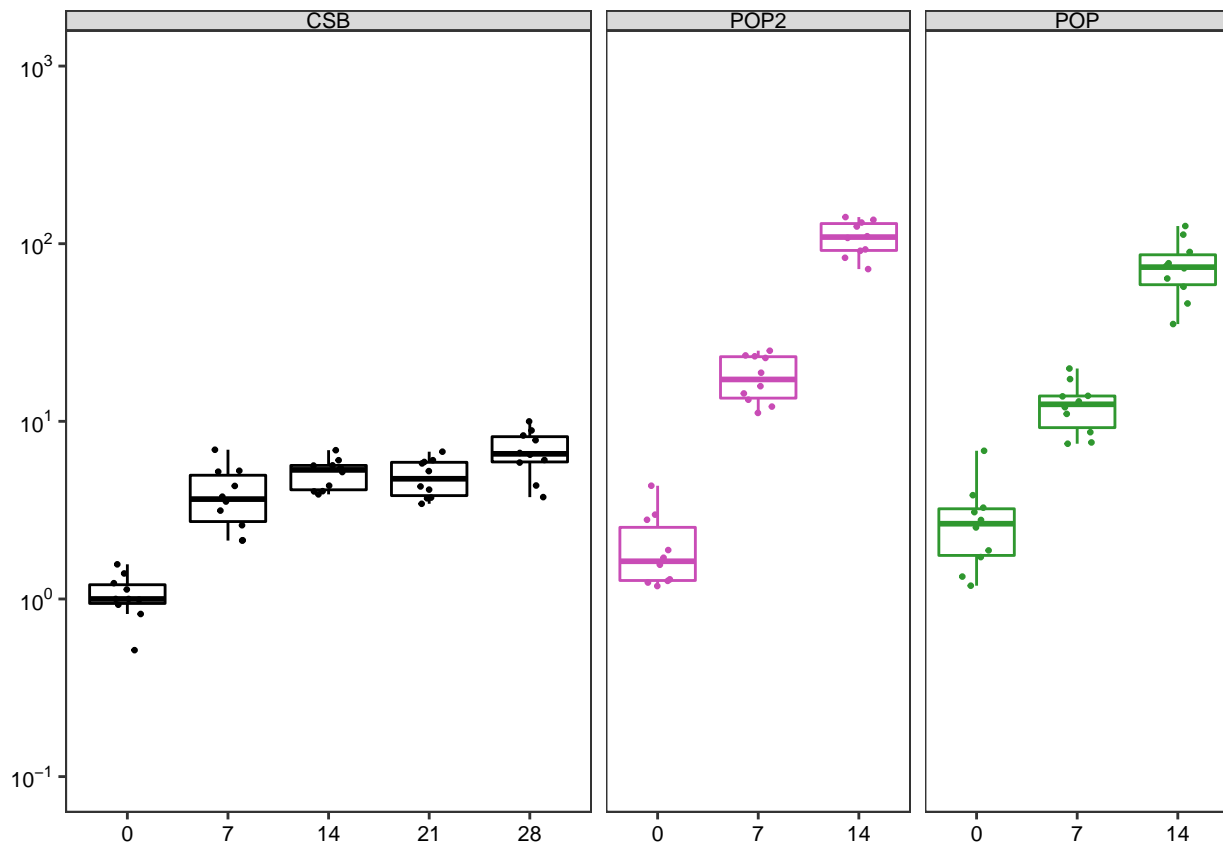
#
multcomp::cld(multiple_exponential_wolb_proliferation_csb_and_pop$lstrends, adj="holm")

## wolbachia time_point.trend      SE df lower.CL upper.CL .group
## CSB              0.40 0.0527 213    0.273    0.527    1
## POP              1.67 0.1177 213    1.386    1.954    2

```

```
## POP2                1.81 0.1177 213    1.529    2.097    2
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05
```

```
### plots
## replicate 1
(s12b1fig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation_csb_and_pop,
                               experimental_replicate=="1"),
                 aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.1, 1e3),
               labels=trans_format('log10',math_format(10^.x)),
               breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,5,1), labels=c("0","7","14","21","28"))+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f")))
```



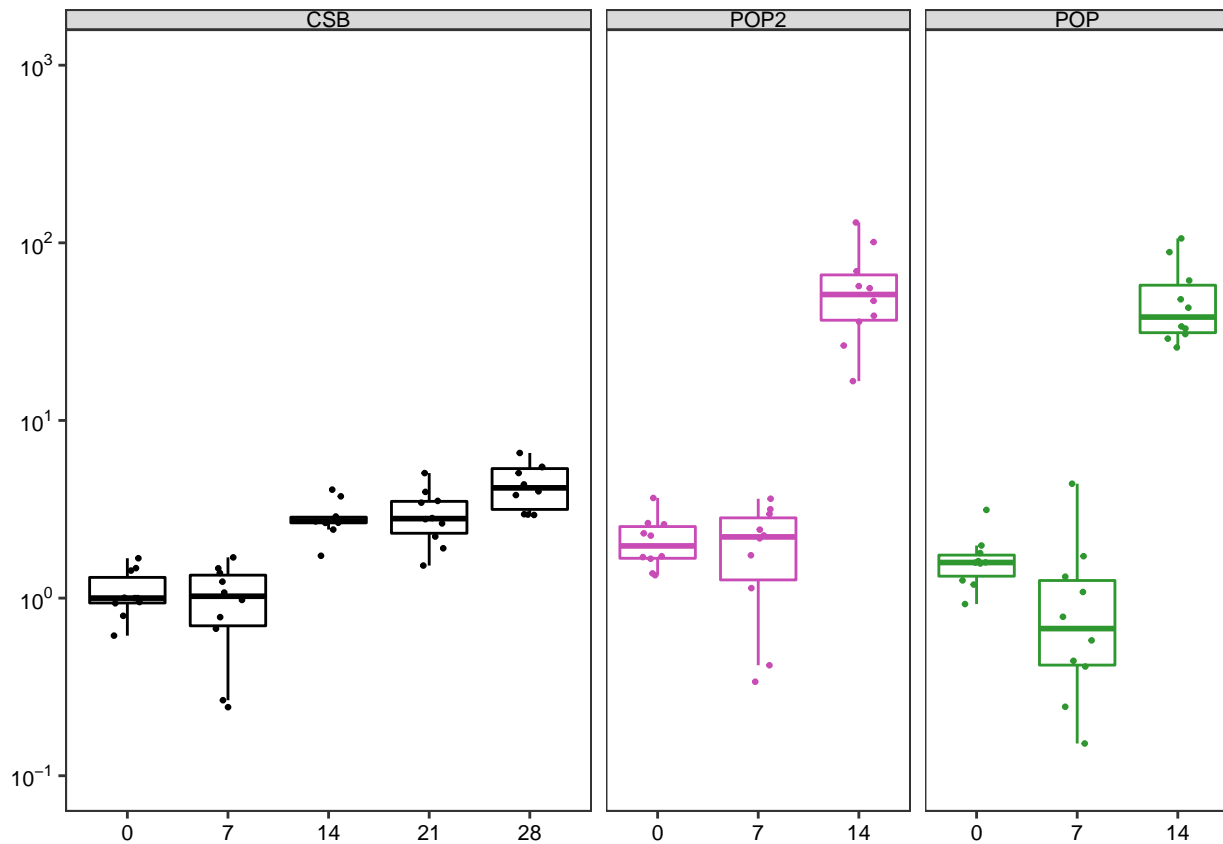
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12B1 Fig.pdf", plot=s12b1fig, device="pdf", dpi=300, units="in", width=30, height=15)
```

```

}

## replicate 2
(s12b2fig=ggplot(dplyr::filter(results_qpcr_wolb_proliferation_csb_and_pop,
                               experimental_replicate=="2"),
                 aes(x=time_point, y=normalized_fold_change, color=wolbachia))+
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(0.1, 1e3),
               labels=trans_format('log10',math_format(10^.x)),
               breaks=c(1e-1,1e0,1e1,1e2,1e3))+
  scale_x_continuous(name="", breaks=seq(1,5,1), labels=c("0","7","14","21","28"))+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f")))

```



```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12B2 Fig.pdf", plot=s12b2fig, device="pdf", dpi=300, units="in", width=30)
}

```

## S12C | Lifespan wMelPop2 and wMelPop

```
### S18 Data
s18data=read.csv("../original_data/S18 Data.csv", stringsAsFactors = TRUE)

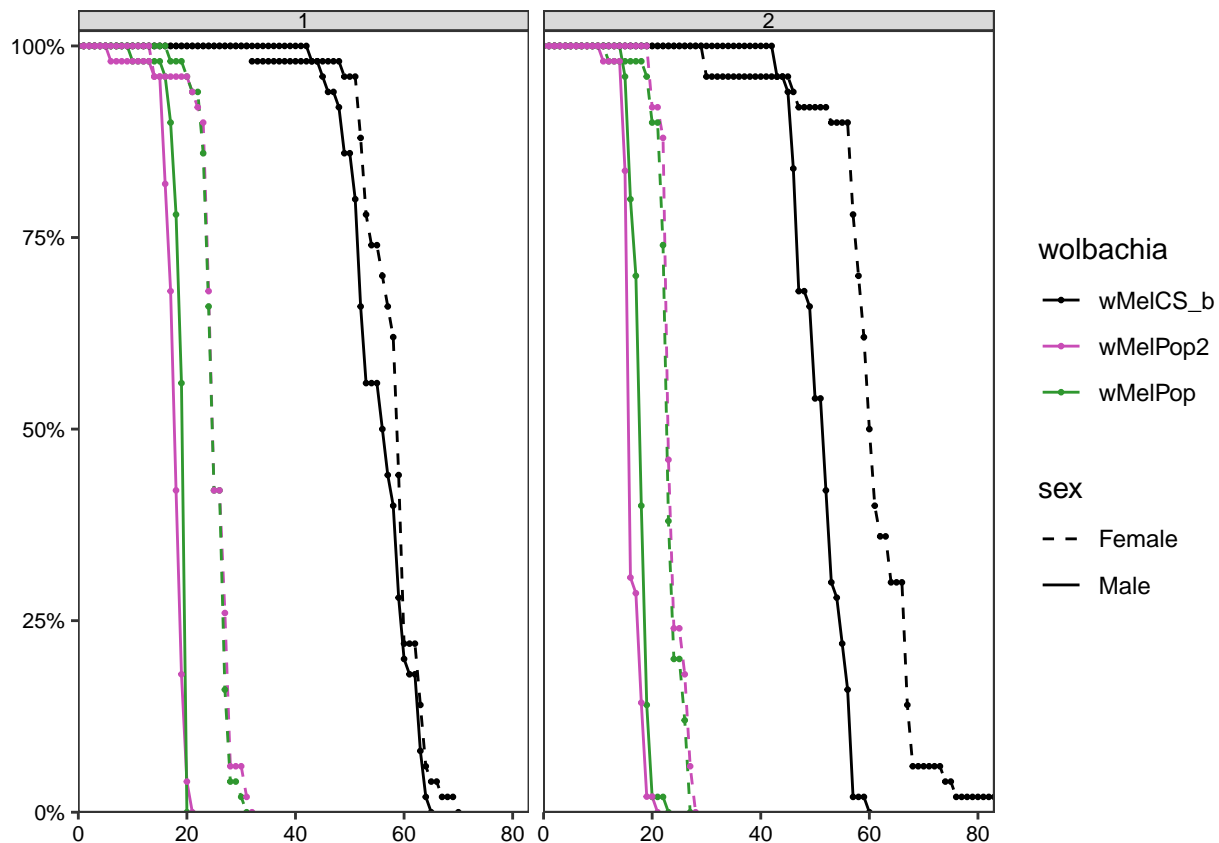
### extracting Kaplan estimates (+/- confidence intervals) from the data
results_lifespan_csb_and_pop=survfit(Surv(day,status)~wolbachia+sex+experimental_replicate, data=s18data)

## converting survival model to a dataframe
results_lifespan_csb_and_pop_dataset=as.data.frame(summary(results_lifespan_csb_and_pop, times=c(0:160)))

## adding new variables to the dataframe
results_lifespan_csb_and_pop_dataset=results_lifespan_csb_and_pop_dataset%>%
  tidyr::separate(strata, c("wolbachia", "sex", "experimental_replicate"), ", ", remove=FALSE)%>%
  dplyr::mutate(wolbachia=str_replace(wolbachia,"wolbachia=", ""),
               sex=str_replace(sex,"sex=", ""),
               experimental_replicate=str_replace(experimental_replicate,"experimental_replicate=", ""))

## converting variables to factor
results_lifespan_csb_and_pop_dataset$wolbachia=factor(results_lifespan_csb_and_pop_dataset$wolbachia, levels=c("wMelPop", "wMelPop2", "wMelPop3"))

### plot
ggplot(data=results_lifespan_csb_and_pop_dataset, aes(x=time, y=surv, linetype=sex, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+
  facet_grid(.~experimental_replicate)+ theme_figs()+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)
```



```
### analysis
model_lifespan_csb_and_pop = coxme::coxme(Surv(day, status)~wolbachia + (1|sex) + (1|unique_id), data=s

# significance of the model
car::Anova(model_lifespan_csb_and_pop)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Surv(day, status)
##           Df  Chisq Pr(>Chisq)
## wolbachia  2 167.55 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# summary
summary(model_lifespan_csb_and_pop, corr=FALSE)
```

```
## Cox mixed-effects model fit by maximum likelihood
## Data: s18data
## events, n = 598, 600
## Iterations= 11 93
##           NULL Integrated Fitted
## Log-likelihood -3235.885 -2607.407 -2520.845
##
##           Chisq df p AIC BIC
## Integrated loglik 1256.96 4.00 0 1248.96 1231.38
## Penalized loglik 1430.08 51.56 0 1326.96 1100.44
```

```

##
## Model: Surv(day, status) ~ wolbachia + (1 | sex) + (1 | unique_id)
## Fixed coefficients
##               coef exp(coef)  se(coef)      z p
## wolbachiaMelPop  8.936457  7604.206  0.7202290 12.41 0
## wolbachiaMelPop2  9.234534 10244.885  0.7177037 12.87 0
##
## Random effects
##   Group      Variable Std Dev  Variance
##   sex        Intercept 2.1002170 4.4109115
##   unique_id Intercept 0.8372961 0.7010648

# multiple comparison
(multiple_lifespan_csb_and_pop=emmeans::emmeans(model_lifespan_csb_and_pop, pairwise ~ wolbachia, adj="none"))

## $emmeans
##   wolbachia emmean    SE df asymp.LCL asymp.UCL
##   wMelCS_b   -6.06 0.470 Inf    -7.18    -4.93
##   wMelPop     2.88 0.276 Inf     2.22     3.54
##   wMelPop2    3.18 0.273 Inf     2.52     3.83
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
##   contrast          estimate    SE df z.ratio p.value
##   wMelCS_b - wMelPop    -8.936 0.720 Inf -12.408 <.0001
##   wMelCS_b - wMelPop2   -9.235 0.718 Inf -12.867 <.0001
##   wMelPop - wMelPop2    -0.298 0.284 Inf  -1.048 0.2945
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 3 tests

multcomp::cld(multiple_lifespan_csb_and_pop$emmeans, adj="holm")

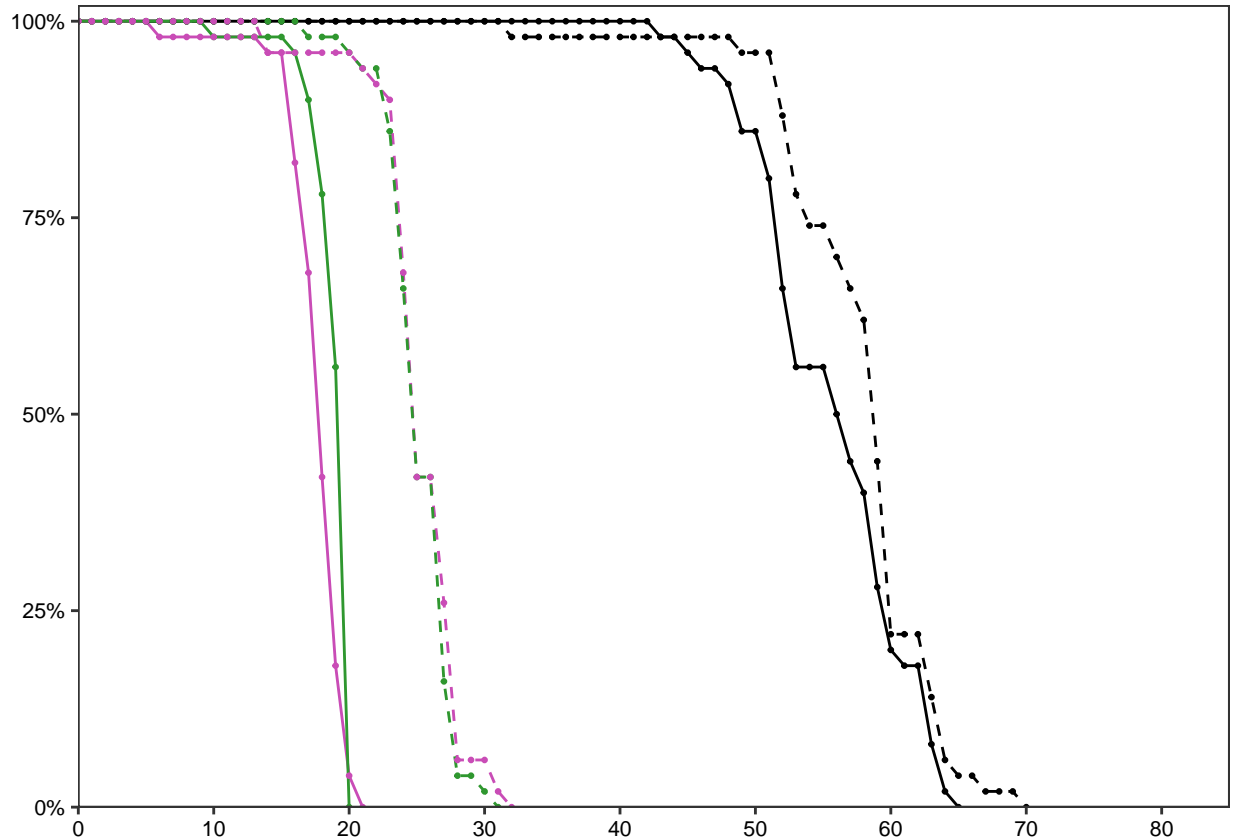
##   wolbachia emmean    SE df asymp.LCL asymp.UCL .group
##   wMelCS_b   -6.06 0.470 Inf    -7.18    -4.93    1
##   wMelPop     2.88 0.276 Inf     2.22     3.54    2
##   wMelPop2    3.18 0.273 Inf     2.52     3.83    2
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
## P value adjustment: holm method for 3 tests
## significance level used: alpha = 0.05

## extracting coefficients
coeff_cox_model_lifespan_csb_and_pop=as.data.frame(multiple_lifespan_csb_and_pop$emmeans)%>%
  group_by(wolbachia)%>%
  summarize(Coeff=emmean, SE=SE)%>%
  mutate(Coeff=Coeff-Coeff[wolbachia=="wMelCS_b"],
         wolbachia=factor(wolbachia, levels=c("wMelCS_b", "wMelPop", "wMelPop2")))

### plots

```

```
## replicate 1
(s12c1fig=ggplot(data=dplyr::filter(results_lifespan_csb_and_pop_dataset,
                                   experimental_replicate=="1"),
                 aes(x=time, y=surv, linetype=sex, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```

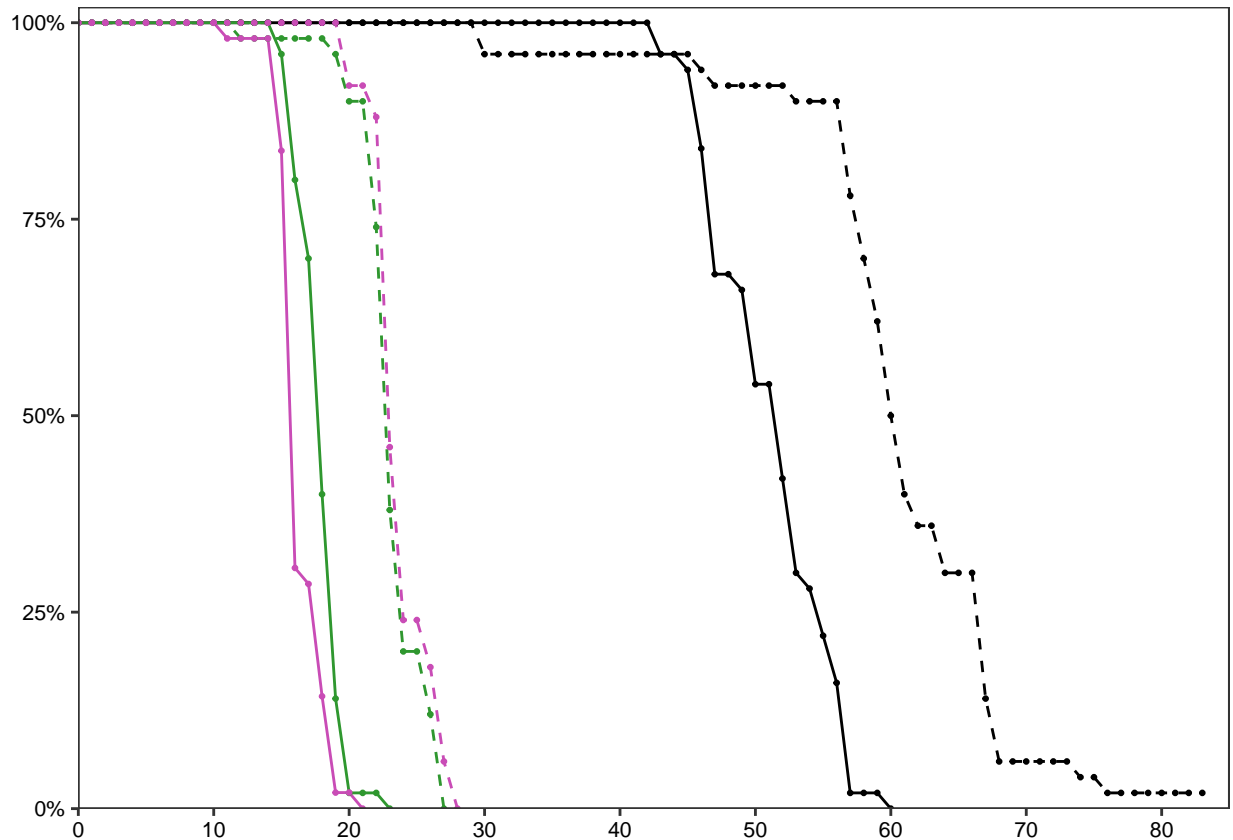


```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12C1 Fig.pdf", plot=s12c1fig, device="pdf", dpi=300, units="in", width=20)
}
```

```
## replicate 2
(s12c2fig=ggplot(data=dplyr::filter(results_lifespan_csb_and_pop_dataset,
                                   experimental_replicate=="2"),
                 aes(x=time, y=surv, linetype=sex, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
```

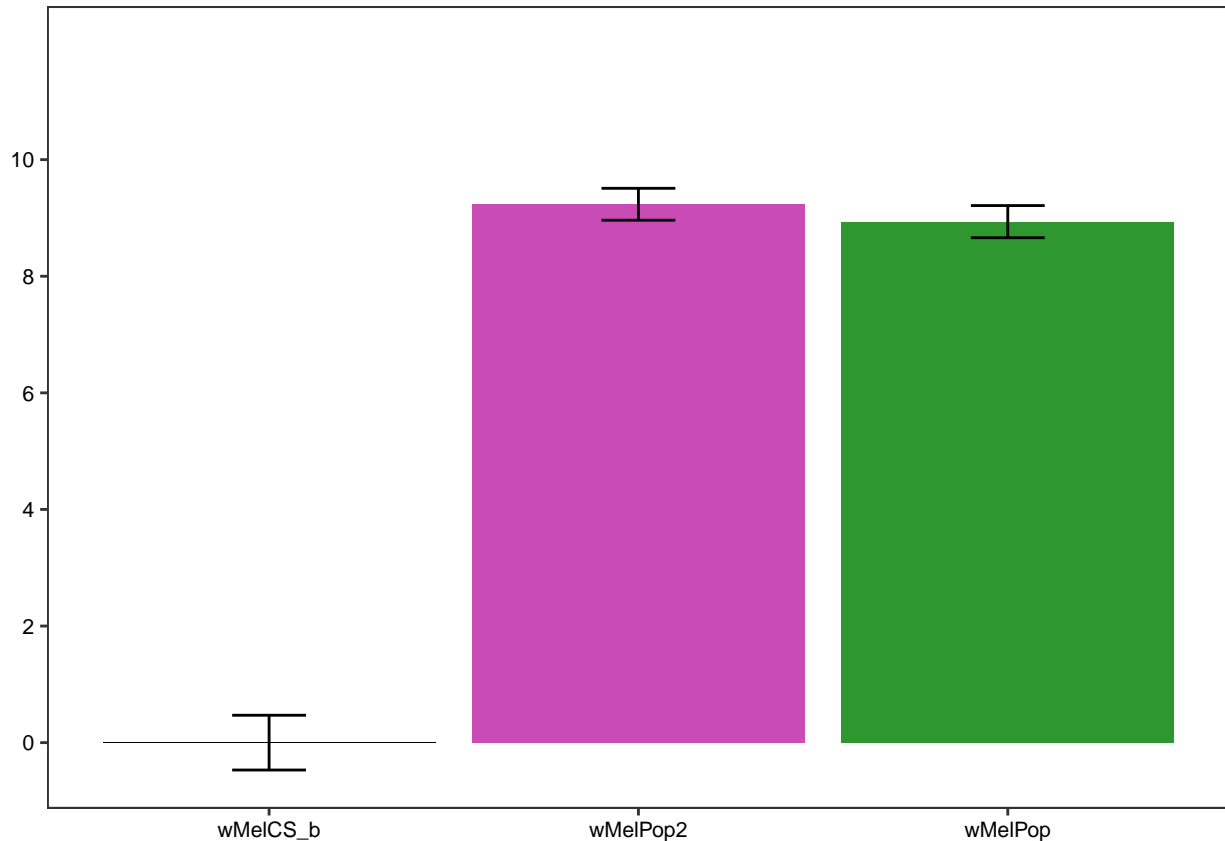


```
scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12C2 Fig.pdf", plot=s12c2fig, device="pdf", dpi=300, units="in", width=20, height=10)
}
```

```
# coefficients
(s12dfig=ggplot(coeff_cox_model_lifespan_csb_and_pop, aes(x=wolbachia, y=Coeff, fill=wolbachia))+
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=Coeff-SE, ymax=Coeff+SE), width=0.2, size=0.5)+
  scale_fill_manual(values=c("#000000", "#c94cb6", "#2f972f"))+
  theme(legend.position="none")+
  scale_y_continuous(limits=c(-0.5, 12), breaks=seq(0,10, 2)))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S12D Fig.pdf", plot=s12df, device="pdf", dpi=300, units="in", width=150, height=100)
}
```

Fig 4 | Wolbachia proliferation in development

Fig 4 | Standard curve analysis

```
### import dataset
s19data=read.csv("../original_data/S19 Data.csv", stringsAsFactors=TRUE)

### estimate mean and SD per sample concentration and per plate
standard_analysis=s19data%>%
  dplyr::filter(grepl("STD", sample))%>%
  dplyr::group_by(plate, sample)%>%
  dplyr::summarise(CTS=mean(cts, na.rm=TRUE), SD=sd(cts, na.rm=TRUE))%>%
  tidyr::separate(sample, c(NA, "dilution"), sep = "_", remove = FALSE)%>%
  dplyr::mutate(dilution=dplyr::recode(dilution, "1"=7.56e7, "2"=7.56e6, "3"=7.56e5,
                                       "4"=7.56e4, "5"=7.56e3, "6"=7.56e2, "7"=7.56e1))

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

#remove highest dilution, it is out of the linear response
standard_analysis <- filter(standard_analysis, dilution != 7.56e1)
```

```

### run liner model on standards per plate
table_models_standard_curve=standard_analysis%>%
  dplyr::group_by(plate)%>%
  do(model=lm(Cts~log10(dilution), data=))

# extracting parameters of the model
(parameters_models_standard_curve=table_models_standard_curve%>%
  do(data.frame(
    plate = .$plate,
    intercept = coef(summary(.$model))[,1][1],
    intercept.se = coef(summary(.$model))[,2][1],
    slope = coef(summary(.$model))[,1][2],
    slope.se = coef(summary(.$model))[,2][2])))

```

```

## # A tibble: 6 x 5
## # Rowwise:
##   plate intercept intercept.se slope slope.se
##   <fct>      <dbl>      <dbl> <dbl>    <dbl>
## 1 plate1      43.7         2.16 -3.34    0.382
## 2 plate2      49.8         0.676 -4.74    0.133
## 3 plate3      46.5         2.29 -4.13    0.405
## 4 plate4      48.8         2.99 -4.16    0.529
## 5 plate5      46.8         2.00 -4.14    0.355
## 6 plate6      48.1         2.18 -4.08    0.386

```

```

### summarizing the Cts values per sample
(sumamry_wolb_absolute_titers=s19data%>%
  dplyr::filter(gene=="WSP")%>%
  dplyr::group_by(plate, sample)%>%
  dplyr::summarise(Cts=mean(cts, na.rm=TRUE),
    SD=sd(cts, na.rm=TRUE)))

```

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

```

## # A tibble: 336 x 4
## # Groups:   plate [6]
##   plate sample    Cts    SD
##   <fct> <fct>    <dbl> <dbl>
## 1 plate1 CSB_A_10  31.8 0.313
## 2 plate1 CSB_B_2   31.6 0.0693
## 3 plate1 CSB_B_8   34.6 0.940
## 4 plate1 CSB_C_8   28.3 0.0219
## 5 plate1 CSB_C_9   29.1 0.0184
## 6 plate1 CSB_D_3   25.8 0.0127
## 7 plate1 CSB_D_9   27.6 0.0926
## 8 plate1 CSB_E_3   24.8 0.0368
## 9 plate1 CSB_E_7   24.8 0.120
## 10 plate1 CSB_G_2   24.6 0.0735
## # ... with 326 more rows

```

```

### merge the cts values with models estimates

```

```

table_results_absolute_wolbachia_titers=merge(sumamry_wolb_absolute_titers, parameters_models_standard_

```

```

### estimating absolute wolbachia titer

```

```

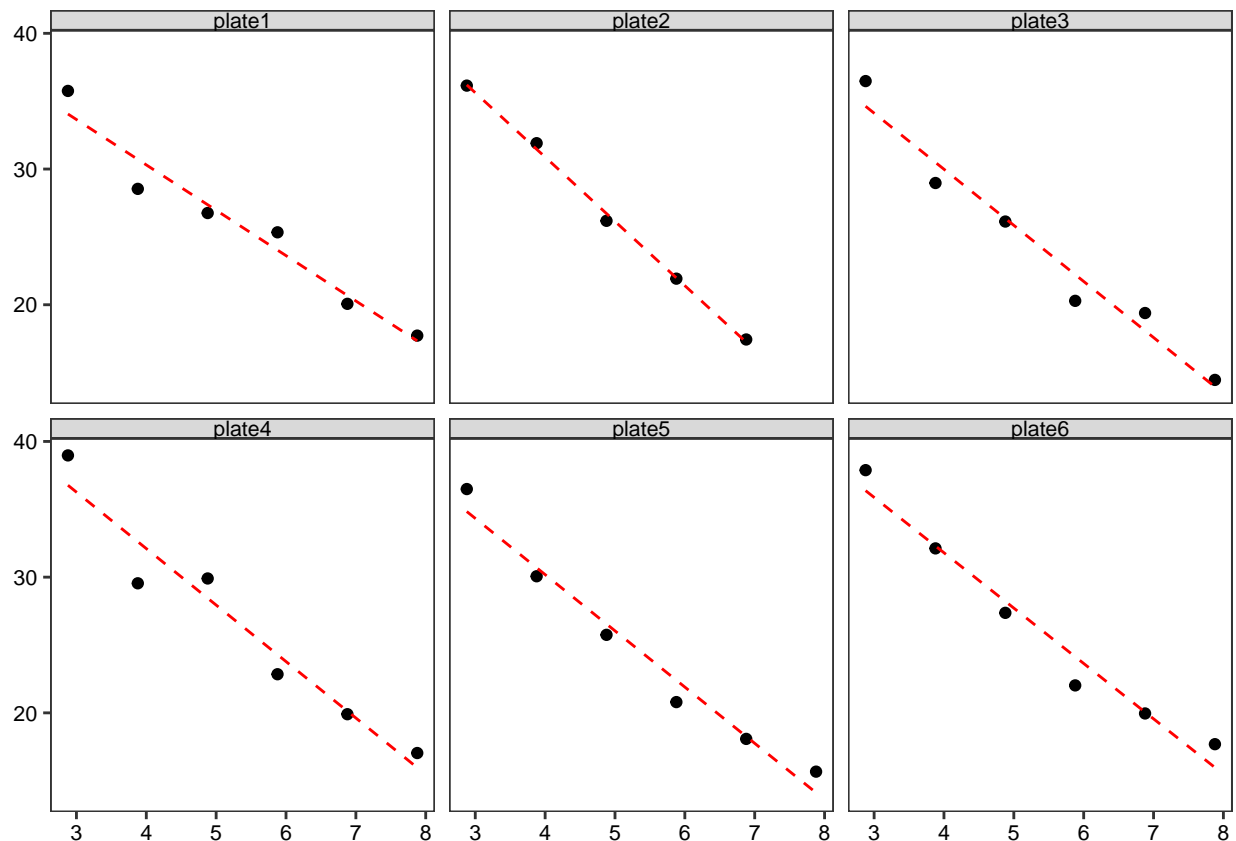
table_results_absolute_wolbachia_titers=table_results_absolute_wolbachia_titers%>%

```

```
dplyr::filter(!(grepl("STD|Ref", sample)))>%
tidyr::separate(sample, c("wolbachia", "time_point", "replicate"), sep="_", remove = FALSE)>%
dplyr::mutate(titers=10**((Cts - intercept)/slope))>%
#filter(wolbachia!="STD", time_point!="Ref")>%
dplyr::mutate(pool=dplyr::recode(time_point, "A"=10, "B"=10, "C"=10, "D"=10,
                                "E"=10, "F"=1, "G"=1, "HF"=1, "HM"=1),
              hours=dplyr::recode(time_point, "A"=2, "B"=26, "C"=50, "D"=74,
                                "E"=86, "F"=120, "G"=168, "HF"=240, "HM"=240))

### plot the standard curve per plate
ggplot(standard_analysis, aes(x=log10(dilution), y=Cts))+
  geom_point()+ theme_figs()+ facet_wrap(~plate, ncol=3)+
  geom_smooth(method="lm", se=FALSE, color="red", linetype="dashed", size=0.5)
```

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



```
## add a new variable which correct titers per individual
table_results_absolute_wolbachia_titers$titers2 <- (table_results_absolute_wolbachia_titers$titers / ta

### wolbachia genome copy dynamics
(wolbachia_genome_copies=table_results_absolute_wolbachia_titers)>%
  dplyr::group_by(wolbachia, time_point)>%
  dplyr::summarise(genome_copies=mean(titers2, na.rm=TRUE),
                  Se=sd(titers2, na.rm=TRUE)/sqrt(n()))
```

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

```
## # A tibble: 27 x 4
## # Groups:   wolbachia [3]
##   wolbachia time_point genome_copies      Se
##   <chr>      <chr>      <dbl>    <dbl>
## 1 CSB        A           4897.   1409.
## 2 CSB        B           5629.   2188.
## 3 CSB        C          11333.  3433.
## 4 CSB        D          53406. 16555.
## 5 CSB        E          87077. 26845.
## 6 CSB        F         907061. 369416.
## 7 CSB        G         952925. 304896.
## 8 CSB        HF         850797. 241488.
## 9 CSB        HM         626335. 178202.
## 10 OCTO      A           4586.   1641.
## # ... with 17 more rows
```

Fig 4 | Statistical analysis

```
## model wolbachia titers in embryos
model_wolbachia_titer_embryos=lm(log(titers2)~wolbachia, data=filter(table_results_absolute_wolbachia_t,
# significance of the model
car::Anova(model_wolbachia_titer_embryos)

## Anova Table (Type II tests)
##
## Response: log(titers2)
##           Sum Sq Df F value Pr(>F)
## wolbachia  0.503  2   0.139 0.8709
## Residuals 45.203 25

# summary
summary(model_wolbachia_titer_embryos)

##
## Call:
## lm(formula = log(titers2) ~ wolbachia, data = filter(table_results_absolute_wolbachia_titers,
##   hours == 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.3639 -0.5006  0.1709  1.1155  1.6964
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.0293980  0.4482209  17.914 8.93e-16 ***
## wolbachiaOCTO -0.2863815  0.6338801  -0.452   0.655
## wolbachiaPOP2  0.0009879  0.6178299   0.002   0.999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.345 on 25 degrees of freedom
## Multiple R-squared:  0.011, Adjusted R-squared:  -0.06812
## F-statistic: 0.139 on 2 and 25 DF,  p-value: 0.8709
```

```
(number_estimates_embryo <- emmeans::emmeans(model_wolbachia_titer_embryos, ~wolbachia, adj="none"))
```

```
## wolbachia emmean SE df lower.CL upper.CL
## CSB 8.03 0.448 25 7.11 8.95
## OCTO 7.74 0.448 25 6.82 8.67
## POP2 8.03 0.425 25 7.15 8.91
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
```

```
(estimates_embryo <- summary(number_estimates_embryo) %>%
  as.data.frame() %>%
  mutate(emmean = exp(emmean)) %>%
  mutate(lower.CL = exp(lower.CL)) %>%
  mutate(upper.CL = exp(upper.CL)))
```

```
## wolbachia emmean SE df lower.CL upper.CL
## 1 CSB 3069.893 0.4482209 25 1219.5896 7727.389
## 2 OCTO 2305.416 0.4482209 25 915.8826 5803.084
## 3 POP2 3072.927 0.4252197 25 1280.0181 7377.148
```

```
## wolbachia titers in adults (time_point distinguishes male from female)
# full model
model_wolbachia_titer_adults=lm(log(titers2)~wolbachia*time_point,
                                data=filter(table_results_absolute_wolbachia_titers,
                                              hours==240))
```

```
# significance of the model
car::Anova(model_wolbachia_titer_adults)
```

```
## Anova Table (Type II tests)
##
## Response: log(titers2)
## Sum Sq Df F value Pr(>F)
## wolbachia 27.078 2 17.0161 1.958e-06 ***
## time_point 3.818 1 4.7986 0.0329 *
## wolbachia:time_point 1.740 2 1.0937 0.3424
## Residuals 42.170 53
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# estimate genome copy numbers from full model
```

```
(multiple_wolbachia_adults_number=emmeans::emmeans(model_wolbachia_titer_adults, pairwise~wolbachia | t
```

```
## $emmeans
## time_point = HF:
## wolbachia emmean SE df lower.CL upper.CL
## CSB 13.2 0.282 53 12.6 13.8
## OCTO 15.0 0.282 53 14.4 15.5
## POP2 14.4 0.282 53 13.8 15.0
##
## time_point = HM:
## wolbachia emmean SE df lower.CL upper.CL
## CSB 13.0 0.297 53 12.4 13.6
## OCTO 14.6 0.282 53 14.0 15.2
```

```
## POP2          13.4 0.282 53      12.9      14.0
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $contrasts
## time_point = HF:
## contrast      estimate      SE df t.ratio p.value
## CSB - OCT0     -1.763 0.399 53 -4.418 <.0001
## CSB - POP2     -1.201 0.399 53 -3.012 0.0040
## OCT0 - POP2      0.561 0.399 53 1.407 0.1654
##
## time_point = HM:
## contrast      estimate      SE df t.ratio p.value
## CSB - OCT0     -1.552 0.410 53 -3.787 0.0004
## CSB - POP2     -0.390 0.410 53 -0.952 0.3452
## OCT0 - POP2      1.162 0.399 53 2.912 0.0052
##
## Results are given on the log (not the response) scale.
(estimates_adults <- as.data.frame(multiple_wolbachia_adults_number$emmeans) %>%
  mutate(emmean = exp(emmean)) %>%
  mutate(lower.CL = exp(lower.CL)) %>%
  mutate(upper.CL = exp(upper.CL)))

##   wolbachia time_point   emmean      SE df lower.CL upper.CL
## 1      CSB          HF 543790.1 0.2820736 53 308831.3 957505.2
## 2     OCT0          HF 3168833.6 0.2820736 53 1799656.2 5579680.3
## 3     POP2          HF 1808054.8 0.2820736 53 1026837.4 3183621.7
## 4      CSB          HM 462285.5 0.2973316 53 254629.8 839288.4
## 5     OCT0          HM 2182462.7 0.2820736 53 1239472.6 3842879.0
## 6     POP2          HM 683030.4 0.2820736 53 387909.3 1202679.5

# simpler model
model_wolbachia_titer_adults_simpler=lm(log(titers2)~wolbachia+time_point,
                                         data=filter(table_results_absolute_wolbachia_titers,
                                                       hours==240))

# significance of the model
car::Anova(model_wolbachia_titer_adults_simpler)

## Anova Table (Type II tests)
##
## Response: log(titers2)
##           Sum Sq Df F value    Pr(>F)
## wolbachia  27.078  2 16.9583 1.832e-06 ***
## time_point   3.818  1  4.7823 0.03302 *
## Residuals  43.910 55
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_wolbachia_titer_adults_simpler)

##
## Call:
```

```

## lm(formula = log(titers2) ~ wolbachia + time_point, data = filter(table_results_absolute_wolbachia_t,
##   hours == 240))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0280 -0.7092 -0.1344  0.7796  1.4376
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   13.3705     0.2328  57.445 < 2e-16 ***
## wolbachiaOCT0    1.6664     0.2863   5.820 3.13e-07 ***
## wolbachiaPOP2    0.8050     0.2863   2.812 0.00682 **
## time_pointHM   -0.5090     0.2328  -2.187 0.03302 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8935 on 55 degrees of freedom
## Multiple R-squared:  0.4098, Adjusted R-squared:  0.3776
## F-statistic: 12.73 on 3 and 55 DF,  p-value: 1.987e-06
# multiple comparison between Wolbachia variants
(multiple_wolbachia_dev_adults=emmeans::emmeans(model_wolbachia_titer_adults_simpler, pairwise~wolbachia

## $emmeans
##   wolbachia emmean    SE df lower.CL upper.CL
##   CSB        13.1 0.205 55     12.6    13.6
##   OCT0        14.8 0.200 55     14.3    15.3
##   POP2        13.9 0.200 55     13.4    14.4
##
## Results are averaged over the levels of: time_point
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
##   contrast      estimate    SE df t.ratio p.value
##   CSB - OCT0    -1.666 0.286 55  -5.820 <.0001
##   CSB - POP2    -0.805 0.286 55  -2.812  0.0071
##   OCT0 - POP2    0.861 0.283 55   3.049  0.0071
##
## Results are averaged over the levels of: time_point
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 3 tests
multcomp::cld(multiple_wolbachia_dev_adults$emmeans)

##   wolbachia emmean    SE df lower.CL upper.CL .group
##   CSB        13.1 0.205 55     12.6    13.6      1
##   POP2        13.9 0.200 55     13.4    14.4      2
##   OCT0        14.8 0.200 55     14.3    15.3      3
##
## Results are averaged over the levels of: time_point
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates

```



```

## P value adjustment: tukey method for comparing a family of 3 estimates
## significance level used: alpha = 0.05
## model growth from egg to white pre-pupae
model_wolbachia_proliferation_development <- lm(log(titers2) ~ hours*wolbachia, data=filter(table_result,
# significance of the model
car::Anova(model_wolbachia_proliferation_development)

## Anova Table (Type II tests)
##
## Response: log(titers2)
##           Sum Sq Df F value Pr(>F)
## hours      651.69  1 259.7043 < 2e-16 ***
## wolbachia    17.08  2   3.4032 0.03563 *
## hours:wolbachia 10.94  2   2.1799 0.11632
## Residuals    411.53 164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(multcomp_development_wolb_levels=emmeans::emmeans(model_wolbachia_proliferation_development, pairwise~
## NOTE: Results may be misleading due to involvement in interactions

## $emmeans
##   wolbachia emmean      SE df lower.CL upper.CL
## CSB         9.72 0.208 164     9.21    10.2
## OCT0        10.40 0.208 164     9.90    10.9
## POP2        10.34 0.216 164     9.82    10.9
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
##   contrast      estimate      SE df t.ratio p.value
## CSB - OCT0   -0.6869 0.294 164  -2.335  0.0623
## CSB - POP2   -0.6265 0.300 164  -2.091  0.0762
## OCT0 - POP2   0.0605 0.300 164   0.202  0.8404
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 3 tests

## estimating doubling time
(multcomp_development=emmeans::lstrends(model_wolbachia_proliferation_development, pairwise~wolbachia, v

## $lstrends
##   wolbachia hours.trend      SE df lower.CL upper.CL
## CSB         0.0422 0.00537 164   0.0316  0.0528
## OCT0        0.0580 0.00537 164   0.0474  0.0686
## POP2        0.0491 0.00529 164   0.0387  0.0596
##
## Confidence level used: 0.95
##
## $contrasts
##   contrast      estimate      SE df t.ratio p.value
## CSB - OCT0  -0.01583 0.00760 164  -2.083  0.0388

```

```
## CSB - POP2 -0.00695 0.00754 164 -0.921 0.3582
## OCT0 - POP2 0.00888 0.00754 164 1.177 0.2409
```

```
#
(doubling_time_dev <- as.data.frame(multcomp_development$lstrends) %>%
  mutate(doubling_time_hours = log(2)/hours.trend) %>%
  mutate(doubling_time = (log(2)/hours.trend)/24) %>%
  mutate(doubling_lower.CL = (log(2)/(upper.CL))/24) %>%
  mutate(doubling_upper.CL = (log(2)/(lower.CL))/24) %>%
  mutate(lower_db = (log(2)/(hours.trend-SE))/24) %>%
  mutate(upper_db = (log(2)/(hours.trend+SE))/24))
```

```
## wolbachia hours.trend SE df lower.CL upper.CL
## 1 CSB 0.04216302 0.005374355 164 0.03155117 0.05277487
## 2 OCT0 0.05799297 0.005374355 164 0.04738112 0.06860482
## 3 POP2 0.04911425 0.005293739 164 0.03866158 0.05956692
## doubling_time_hours doubling_time doubling_lower.CL doubling_upper.CL
## 1 16.43969 0.6849872 0.5472516 0.9153743
## 2 11.95226 0.4980109 0.4209782 0.6095493
## 3 14.11295 0.5880397 0.4848518 0.7470241
## lower_db upper_db
## 1 0.7850551 0.6075458
## 2 0.5488767 0.4557732
## 3 0.6590779 0.5308252
```

```
## simplify model to remove interaction
model_wolbachia_proliferation_development_simpler <- lm(log(titers2) ~ hours + wolbachia, data=filter(t
```

```
# significance of the model
car::Anova(model_wolbachia_proliferation_development_simpler)
```

```
## Anova Table (Type II tests)
##
## Response: log(titers2)
## Sum Sq Df F value Pr(>F)
## hours 651.69 1 256.0643 < 2e-16 ***
## wolbachia 17.08 2 3.3555 0.03728 *
## Residuals 422.47 166
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# summary
summary(model_wolbachia_proliferation_development_simpler)
```

```
##
## Call:
## lm(formula = log(titers2) ~ hours + wolbachia, data = filter(table_results_absolute_wolbachia_titers,
## hours < 160))
##
## Residuals:
## Min 1Q Median 3Q Max
## -7.2869 -0.6890 0.0559 1.1842 2.6440
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.750105 0.280922 24.028 <2e-16 ***
```

```
## hours          0.049750    0.003109   16.002   <2e-16 ***
## wolbachiaOCT0 0.697666    0.296242    2.355   0.0197 *
## wolbachiaPOP2 0.632546    0.301751    2.096   0.0376 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.595 on 166 degrees of freedom
## Multiple R-squared:  0.6122, Adjusted R-squared:  0.6052
## F-statistic: 87.36 on 3 and 166 DF,  p-value: < 2.2e-16

## growth in pupae
model_wolbachia_proliferation_pupae <- lm(log(titers2) ~ hours*wolbachia, data=filter(table_results_abs

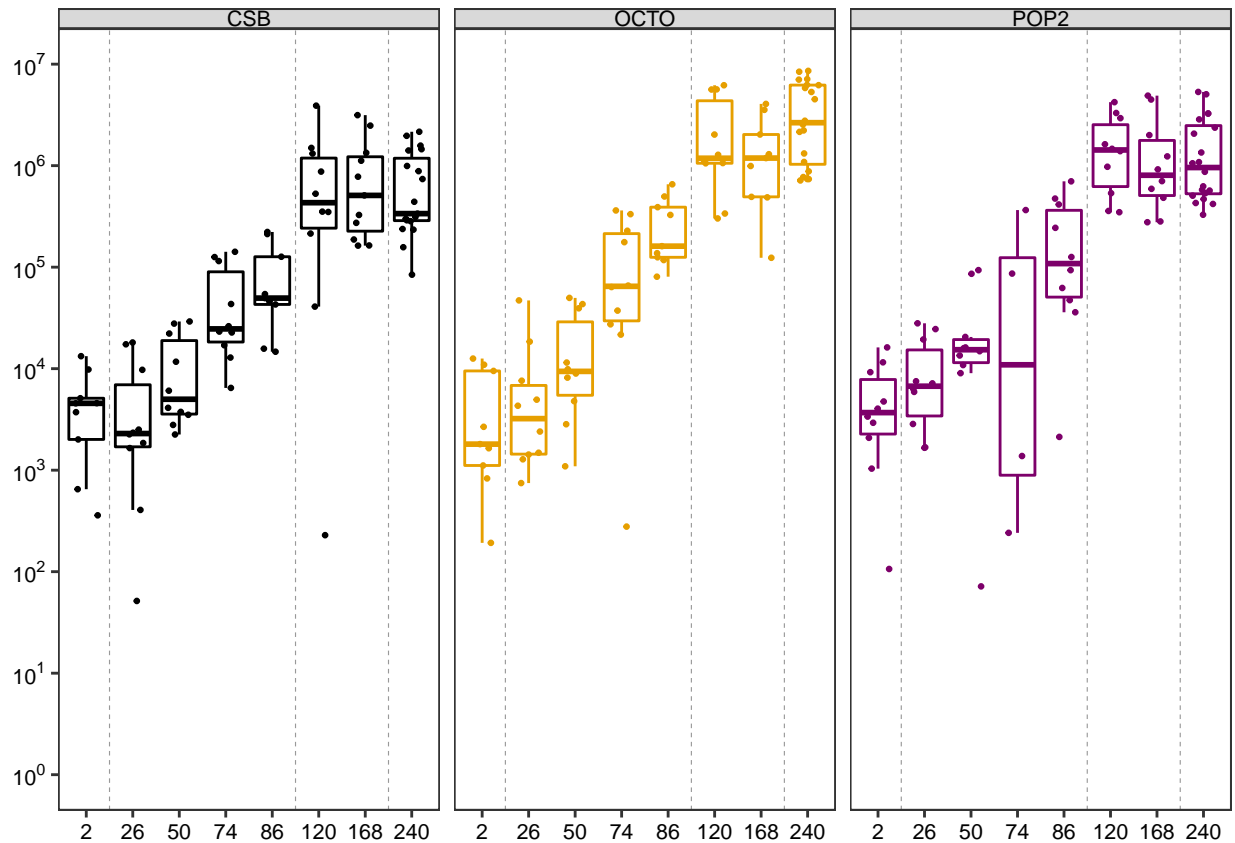
# significance of the model
car::Anova(model_wolbachia_proliferation_pupae)

## Anova Table (Type II tests)
##
## Response: log(titers2)
##              Sum Sq Df F value    Pr(>F)
## hours          3.410  1  2.2761    0.1342
## wolbachia      41.661  2 13.9033 3.997e-06 ***
## hours:wolbachia  2.343  2   0.7820    0.4600
## Residuals     169.300 113
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fig 4 | final figure

```
### final figure
# panels
panels_development_discrete=data.frame(xstart=c(0,1.5,5.5,7.5), xend=c(1.5,5.5,7.5,8.5), stage=c("eggs"

# figure
(fig4=ggplot()+
  geom_boxplot(data = table_results_absolute_wolbachia_titers,
               aes(x=factor(hours), titers/pool, color=wolbachia), outlier.shape=NA)+
  geom_jitter(data = table_results_absolute_wolbachia_titers,
              aes(x=factor(hours), titers/pool, color=wolbachia), width=0.25, size=0.5)+
  facet_grid(.~wolbachia)+
  geom_vline(xintercept=c(1.5,5.5,7.5), linetype="dashed", color="gray60", size=0.2)+
  scale_y_log10(name="", limits=c(1e0, 1e7),
               labels=trans_format('log10',math_format(10^.x)),
               breaks=c(1e0,1e1,1e2,1e3,1e4,1e5,1e6,1e7))+
  scale_color_manual(values=c("#000000", "#e69f00", "#7d006a"))+
  scale_fill_manual(values=c("#000000", "gray80", "gray60", "gray40"))+
  theme_figs()+ theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 4.pdf", plot=fig4, device="pdf", dpi=300, units="in", width=550/96, height=400/96)
}
```

## S13 Fig | Dynamics Octomom copy number

### S13A Fig | Octomom copy number development

```
### determining relative wolbachia titers
results_octomom_copy_development=droplevels(subset(s19data, s19data$gene!="RPL32"))%>%
  group_by(plate)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="CSB_Ref", reference_gene="WSP", efficiency_ref=2, efficiency_cal=1))

## adding new variables
results_octomom_copy_development=results_octomom_copy_development%>%
  tidyr::separate(sample, c("wolbachia", "time_point", "replicate"), "_", remove=FALSE)%>%
  filter(time_point!="Ref")%>%
  mutate(hours=dplyr::recode(time_point, "A"=2, "B"=26, "C"=50, "D"=74,
                             "E"=86, "F"=120, "G"=168, "HF"=240, "HM"=240))

#remove Octoless
results_octomom_copy_development <- filter(results_octomom_copy_development, wolbachia != "OCTO")
```

```

### analysis
model_octomom_copy_development=lm(normalized_fold_change~hours*wolbachia, data=results_octomom_copy_development)

## significance
car::Anova(model_octomom_copy_development)

## Anova Table (Type II tests)
##
## Response: normalized_fold_change
##           Sum Sq Df F value Pr(>F)
## hours           0.76  1  0.4800 0.4894
## wolbachia      1208.23  1 759.4715 <2e-16 ***
## hours:wolbachia  0.00  1  0.0005 0.9829
## Residuals       264.09 166
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## summary
summary(model_octomom_copy_development)

##
## Call:
## lm(formula = normalized_fold_change ~ hours * wolbachia, data = results_octomom_copy_development)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.6365 -0.3854 -0.1076  0.1467  4.8644
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.084e+00  2.291e-01   4.732 4.74e-06 ***
## hours          7.763e-04  1.645e-03   0.472  0.638
## wolbachiaPOP2    5.328e+00  3.268e-01  16.305 < 2e-16 ***
## hours:wolbachiaPOP2 4.955e-05  2.314e-03   0.021  0.983
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.261 on 166 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.8209, Adjusted R-squared:  0.8177
## F-statistic: 253.6 on 3 and 166 DF, p-value: < 2.2e-16

### panels
panels_development_discrete=data.frame(xstart=c(0,1.5,5.5,7.5), xend=c(1.5,5.5,7.5,8.5), stage=c("eggs", "larvae", "pupa", "adult"))

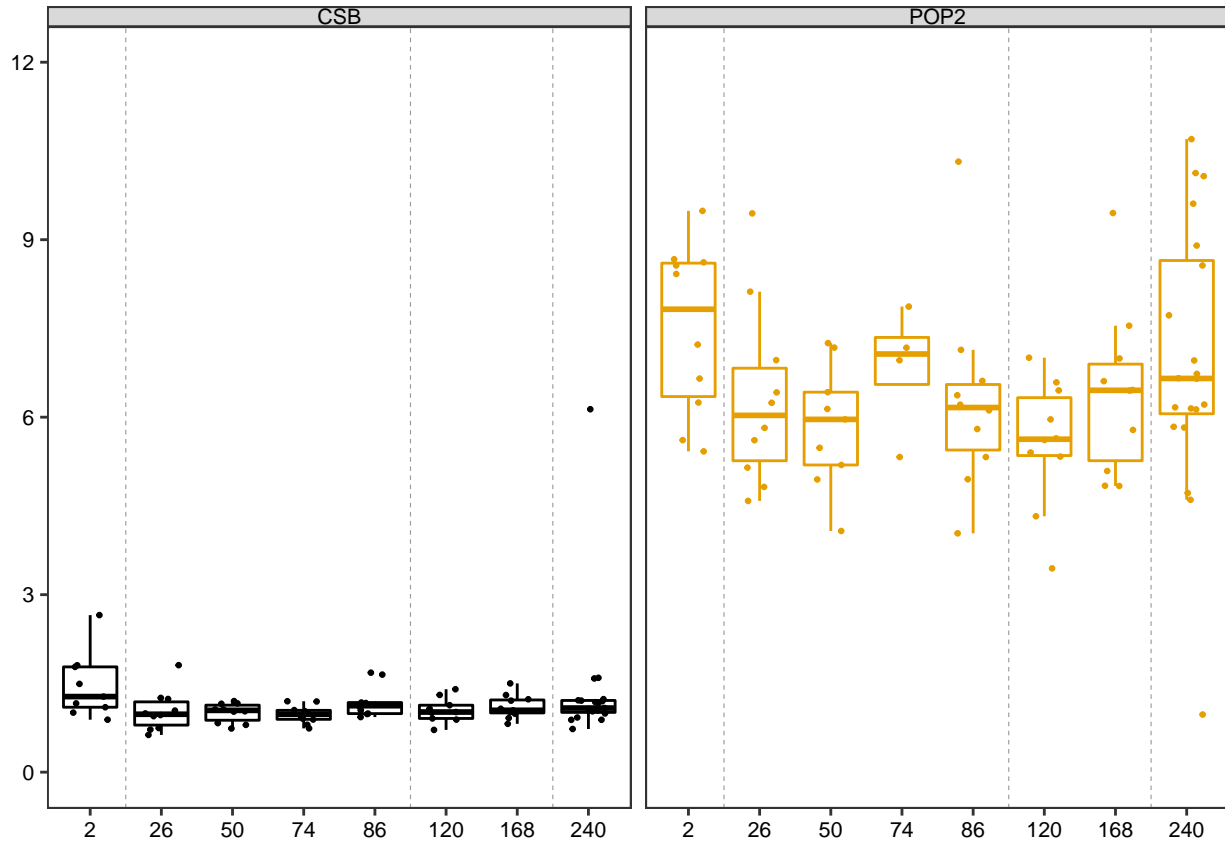
(s13afig=ggplot()+
  geom_boxplot(data = results_octomom_copy_development,
    aes(x=factor(hours), y=normalized_fold_change, color=wolbachia), outlier.shape=NA)+
  geom_jitter(data = results_octomom_copy_development,
    aes(x=factor(hours), y=normalized_fold_change, color=wolbachia), width=0.25, size=0.5)+
  facet_grid(.~wolbachia)+
  geom_vline(xintercept=c(1.5,5.5,7.5), linetype="dashed", color="gray60", size=0.2)+
  scale_y_continuous(limits=c(-0.01,12), breaks=seq(0,12,3))+
  scale_color_manual(values=c("#000000", "#e69f00", "#7d006a"))+
  scale_fill_manual(values=c("#000000", "gray80", "gray60", "gray40"))+

```

```
theme_figs()+ theme(legend.position="none"))
```

```
## Warning: Removed 2 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S13A Fig.pdf", plot=s13afig, device="pdf", dpi=300, units="in", width=550, height=400)
}
```

## S13B | Octomom copy number dynamics in adults

```
### S20 data
s20data=read.csv("../original_data/S20 Data.csv", stringsAsFactors = TRUE)

### dynamics octomom copy number
results_qpcr_octomom_dynamics=s20data%>%
  dplyr::group_by(experimental_replicate, temperature, plate)%>%
  do(qpcr.pfaffl(data=., calibrator_sample="CSB-REF", reference_gene="WSP", efficiency_ref=2, efficiency_cal=1))

## adding new variables
results_qpcr_octomom_dynamics=results_qpcr_octomom_dynamics%>%
  tidyr::separate(sample, c("wolbachia", "time_point", "replicate"), "_", remove=FALSE)%>%
```

```

filter(wolbachia!="CSB-REF")

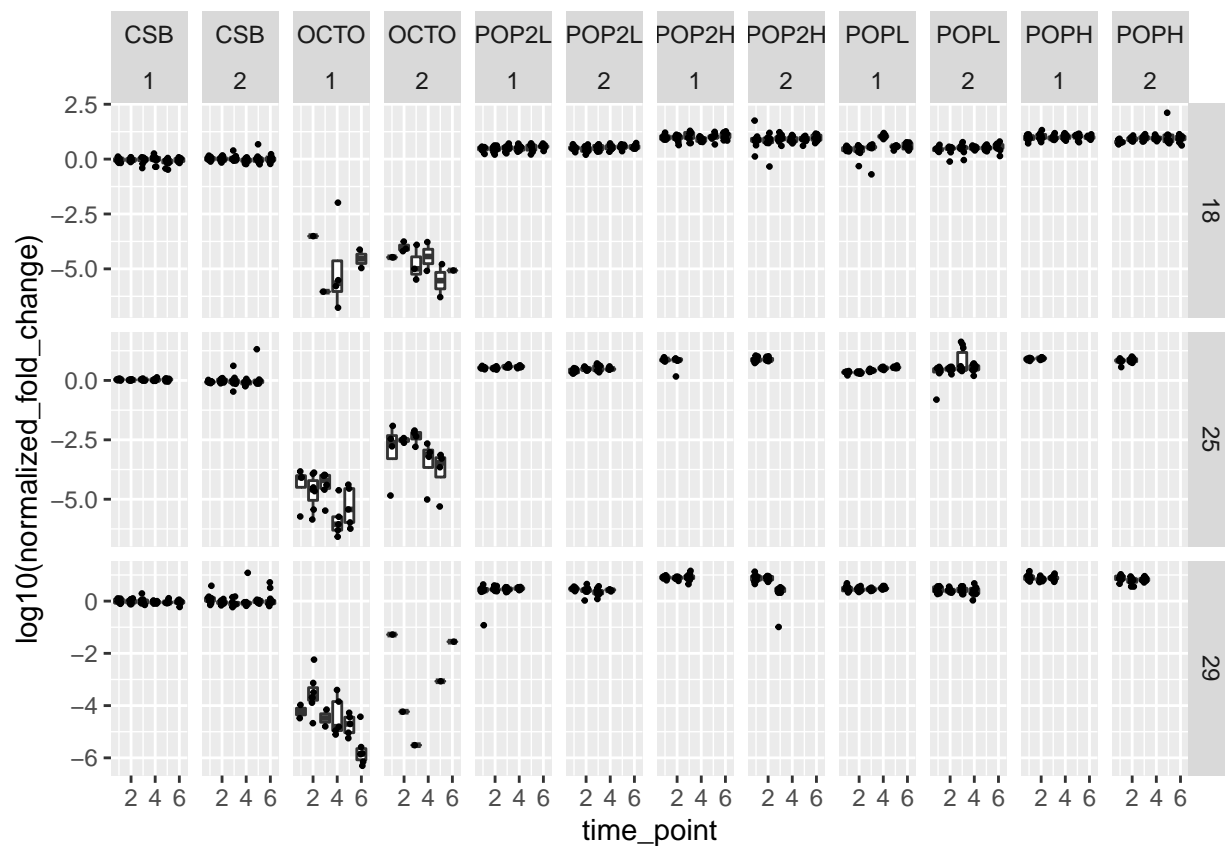
# converting variables to factors or continuous variables
results_qpcr_octomom_dynamics$wolbachia=factor(results_qpcr_octomom_dynamics$wolbachia, levels=c("CSB",
#
results_qpcr_octomom_dynamics$time_point=as.numeric(as.factor(results_qpcr_octomom_dynamics$time_point))

### plot
ggplot(results_qpcr_octomom_dynamics, aes(x=time_point, y=log10(normalized_fold_change)))+
  geom_boxplot(outlier.shape=NA, aes(group=time_point))+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(temperature~wolbachia+experimental_replicate, scales="free")

```

## Warning: Removed 240 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 240 rows containing missing values (geom\_point).



```

### replacing wMelOctoless 'NaN'
## fold change
results_qpcr_octomom_dynamics$fold_change[results_qpcr_octomom_dynamics$wolbachia=="OCTO" & is.nan(results_qpcr_octomom_dynamics$fold_change)] = 0

## normalized fold change
results_qpcr_octomom_dynamics$normalized_fold_change[results_qpcr_octomom_dynamics$wolbachia=="OCTO" & is.nan(results_qpcr_octomom_dynamics$normalized_fold_change)] = 1

#get new day column applying a transforming factor to time_point

```

```

results_qpcr_octomom_dynamics <- results_qpcr_octomom_dynamics %>%
  mutate(day = as.character(temperature)) %>%
  mutate(day = dplyr::recode(day, "18" = "10", "25" = "7", "29" = "3")) %>%
  mutate(day = as.numeric(day)) %>%
  mutate(day = ((time_point-1)*day))

```

*#make temperature a factor*

```

results_qpcr_octomom_dynamics <- results_qpcr_octomom_dynamics %>%
  ungroup() %>%
  mutate(temperature = as.factor(as.character(temperature)))

```

*### summary Octomom copy number dynamics*

```

simplified_results_qpcr_octomom_dynamics=results_qpcr_octomom_dynamics%>%
  dplyr::group_by(temperature, wolbachia, time_point)%>%
  dplyr::summarise(Median=median(normalized_fold_change, na.rm=TRUE),
                  IntQRRange=IQR(normalized_fold_change, na.rm=TRUE),
                  Mean=mean(normalized_fold_change, na.rm=TRUE),
                  SD=sd(normalized_fold_change, na.rm=TRUE))

```

## `summarise()` has grouped output by 'temperature', 'wolbachia'. You can override using the `.groups`

*### models for statistical analysis*

*## all together*

*# removed Octoless and wMelCS\_b*

```

model_octomom_dynamics_all=lmer(normalized_fold_change~day*wolbachia*temperature+(1|experimental_replicat

```

*# significance*

```

car::Anova(model_octomom_dynamics_all)

```

## Analysis of Deviance Table (Type II Wald chisquare tests)

##

## Response: normalized\_fold\_change

	Chisq	Df	Pr(>Chisq)
## day	5.4076	1	0.02005 *
## wolbachia	281.3864	3	< 2e-16 ***
## temperature	6.4957	2	0.03886 *
## day:wolbachia	5.0374	3	0.16908
## day:temperature	1.6162	2	0.44570
## wolbachia:temperature	6.8451	6	0.33541
## day:wolbachia:temperature	2.4887	6	0.86973

## ---

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

*## simpler model*

```

model_octomom_dynamics_all_simpler=lmer(normalized_fold_change~day+wolbachia+temperature+(1|experimental

```

*# significance of the model*

```

car::Anova(model_octomom_dynamics_all_simpler)

```

## Analysis of Deviance Table (Type II Wald chisquare tests)

##

## Response: normalized\_fold\_change

	Chisq	Df	Pr(>Chisq)
## day	6.8553	1	0.008838 **
## wolbachia	289.1324	3	< 2.2e-16 ***



```

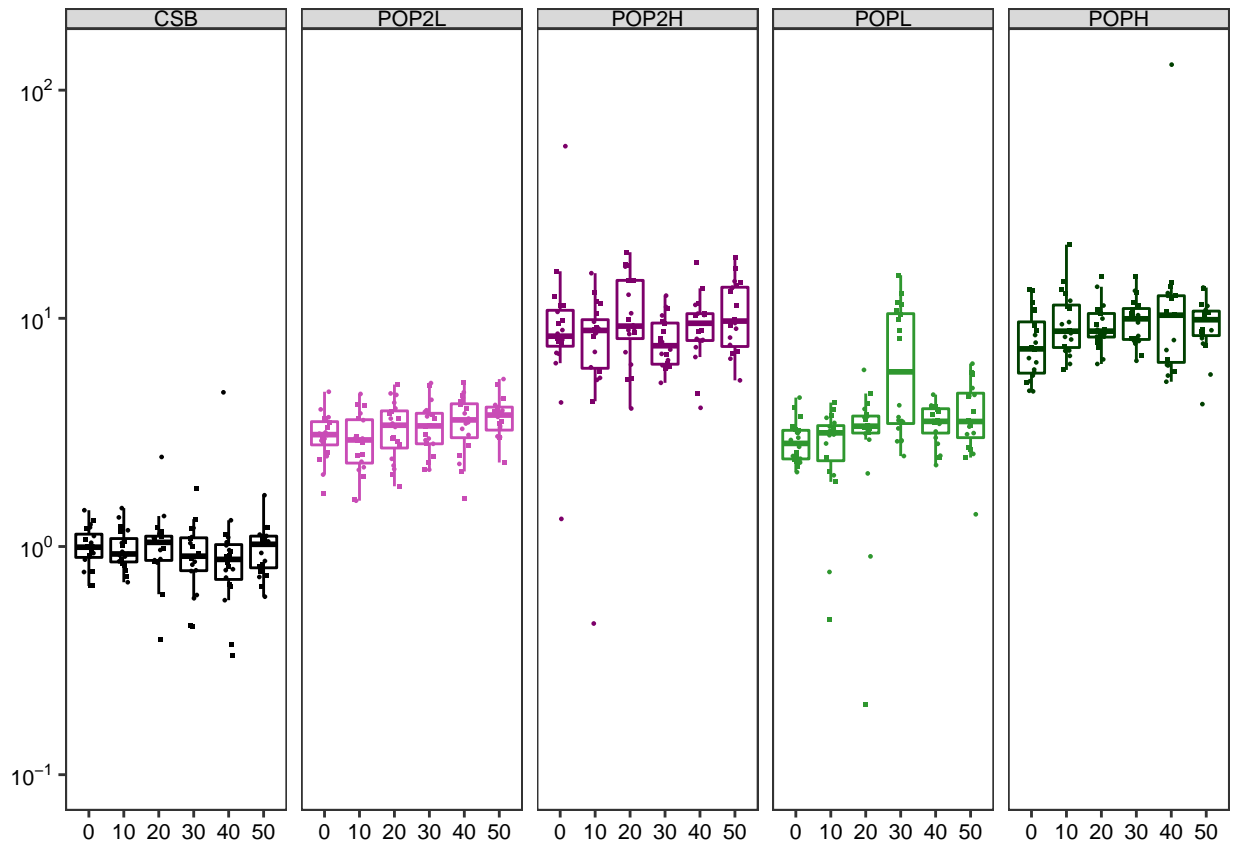
## temperature    6.3600  2    0.041585 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_octomom_dynamics_all_simpler, corr=FALSE)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: normalized_fold_change ~ day + wolbachia + temperature + (1 |
##     experimental_replicate)
## Data:
## filter(results_qpcr_octomom_dynamics, wolbachia != "OCTO" & wolbachia !=
##     "CSB")
##
## REML criterion at convergence: 5963.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.7027 -0.2500 -0.0678  0.1136 24.4115
##
## Random effects:
## Groups              Name              Variance Std.Dev.
## experimental_replicate (Intercept)  0.02242 0.1497
## Residual                        23.82527 4.8811
## Number of obs: 992, groups:  experimental_replicate, 2
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    3.15419    0.47864  40.89769   6.590 6.34e-08 ***
## day            0.03230    0.01233  984.62055   2.618  0.00897 **
## wolbachiaPOP2H  5.35326    0.44642  984.19031  11.992 < 2e-16 ***
## wolbachiaPOPL   0.42126    0.41471  984.00015   1.016  0.30998
## wolbachiaPOPH   5.80815    0.44871  984.00839  12.944 < 2e-16 ***
## temperature25  -0.50165    0.43901  984.78941  -1.143  0.25345
## temperature29  -1.14404    0.45516  984.44106  -2.513  0.01211 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

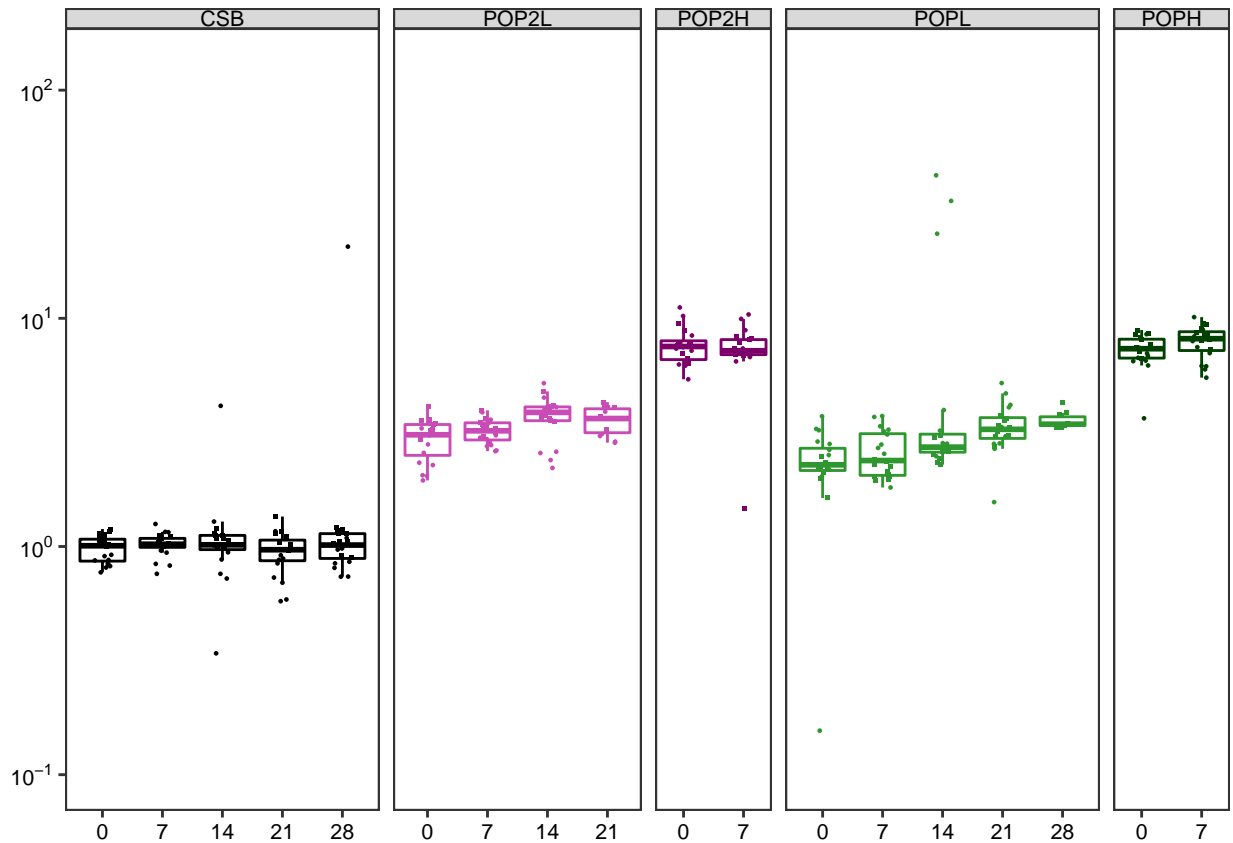
### plots
## 18°C
(s13b1fig=ggplot(dplyr::filter(results_qpcr_octomom_dynamics, temperature=="18", wolbachia!="OCTO"),
  aes(x=time_point, y=normalized_fold_change, color=wolbachia, shape=factor(experimental_replicate)),
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(.~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(1e-1, 1.3e2),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2))+
  scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","10","20","30","40","50"))+
  scale_shape_manual(values=c(15,16))+
  scale_color_manual(values=c("#000000", "#c94cb6", "#7d006a", "#2f972f", "#004000"))))

```



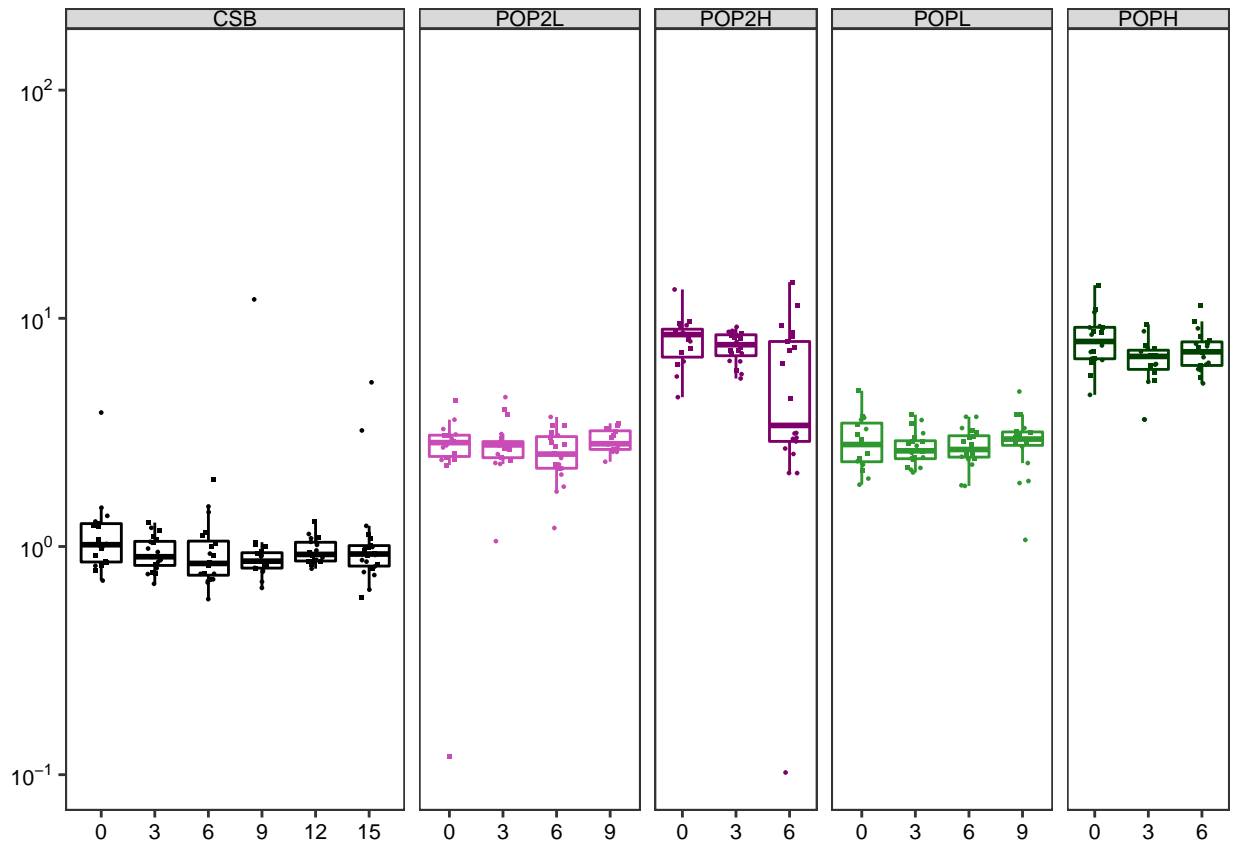
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S13B1 Fig.pdf", plot=s13b1fig, device="pdf", dpi=300, units="in", width=500, height=500)
}

## 25°C
(s13b2fig=ggplot(dplyr::filter(results_qpcr_octomom_dynamics, temperature=="25", wolbachia!="OCTO"),
  aes(x=time_point, y=normalized_fold_change, color=wolbachia, shape=factor(experimental_replicate)),
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(1e-1, 1.3e2),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2))+
  scale_x_continuous(name="", breaks=seq(1,5,1), labels=c("0","7","14","21","28"))+
  scale_shape_manual(values=c(15,16))+
  scale_color_manual(values=c("#000000", "#c94cb6", "#7d006a", "#2f972f", "#004000")))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S13B2 Fig.pdf", plot=s13b2fig, device="pdf", dpi=300, units="in", width=500, height=500)
}

## 29°C
(s13b3fig=ggplot(dplyr::filter(results_qpcr_octomom_dynamics, temperature=="29", wolbachia!="OCTO"),
  aes(x=time_point, y=normalized_fold_change, color=wolbachia, shape=factor(experimental_replicate)),
  geom_boxplot(aes(group=time_point), outlier.shape=NA)+
  geom_jitter(width=0.15, size=0.5)+
  facet_grid(~wolbachia, scales="free_x", space="free_x")+
  theme_figs()+ theme(legend.position="none")+
  scale_y_log10(name="", limits=c(1e-1, 1.3e2),
    labels=trans_format('log10',math_format(10^.x)),
    breaks=c(1e-1,1e0,1e1,1e2))+
  scale_x_continuous(name="", breaks=seq(1,6,1), labels=c("0","3","6","9","12","15"))+
  scale_shape_manual(values=c(15,16))+
  scale_color_manual(values=c("#000000", "#c94cb6", "#7d006a", "#2f972f", "#004000")))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S13B3 Fig.pdf", plot=s13b3fig, device="pdf", dpi=300, units="in", width=5
}

```

**Fig 5 | Wolbachia life-shortening phenotype**

**Fig 5 | Dataset**

```
### S21 Data
s21data=read.csv("../original_data/S21 Data.csv", stringsAsFactors = TRUE)

### extracting Kaplan estimates (+/- confidence intervals) from the data
results_lifespan_all_flies=survfit(Surv(day,status)~wolbachia+temperature+sex+experimental_replicate, d

## converting survival model to a dataframe
results_lifespan_all_flies_dataset=as.data.frame(summary(results_lifespan_all_flies, times=c(0:160))$c

## adding new variables to the dataframe
results_lifespan_all_flies_dataset=results_lifespan_all_flies_dataset%>%
  tidyr::separate(strata, c("wolbachia", "temperature", "sex", "experimental_replicate"), ", ", remove=
  mutate(wolbachia=str_replace(wolbachia, "wolbachia=", ""),
         temperature=str_replace(temperature, "temperature=", ""),
         sex=str_replace(sex, "sex=", ""))

```

```

experimental_replicate=str_replace(experimental_replicate, "experimental_replicate=", "")

## converting variables to factor
results_lifespan_all_flies_dataset$wolbachia=factor(results_lifespan_all_flies_dataset$wolbachia, level=)

### plot
ggplot(data=results_lifespan_all_flies_dataset, aes(x=time, y=surv, color=wolbachia, linetype=sex))+
  geom_line(size=0.5)+ geom_point(size=0.5)+
  facet_grid(experimental_replicate~temperature, scales="free")+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  ylab("Survival") + xlab("Days post eclosion")+
  scale_x_continuous(expand=c(0,0))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)

```

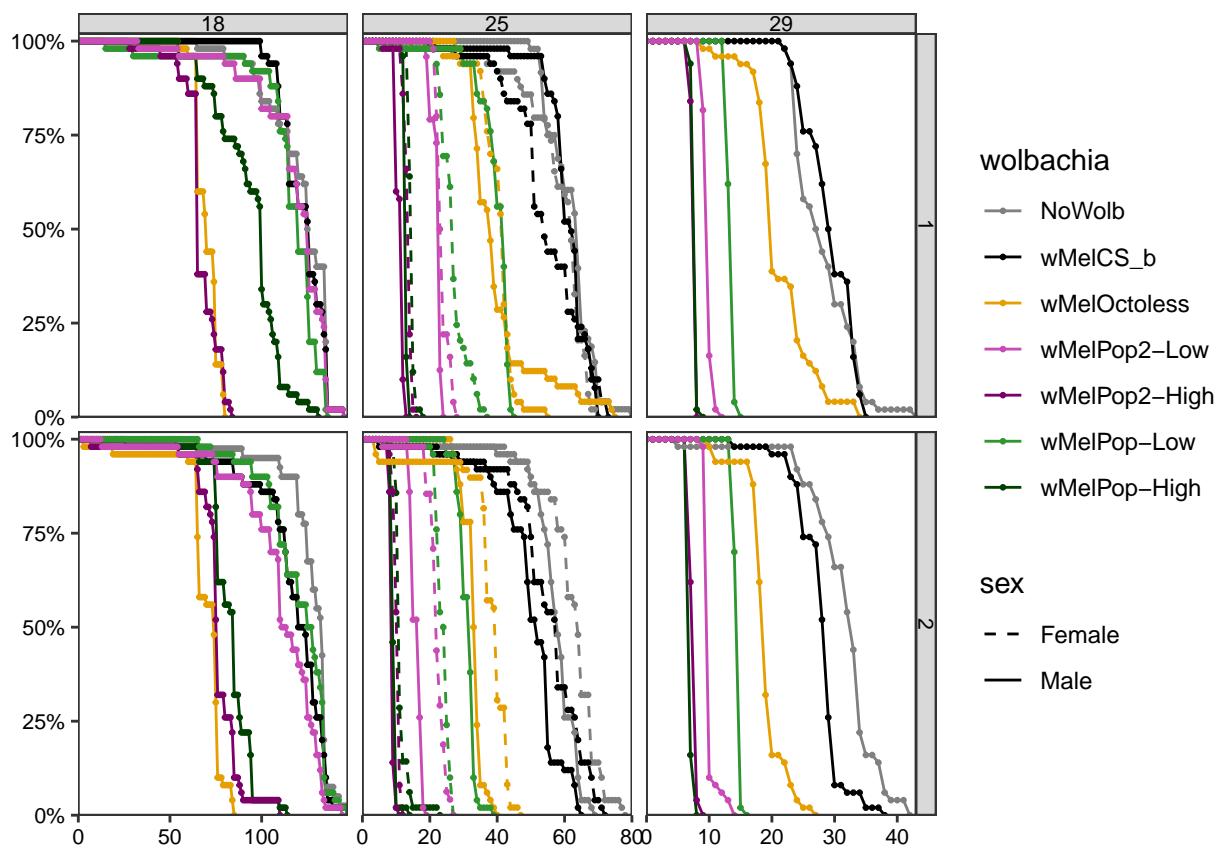


Fig 5 | Statistical analysis

```

### make temperature a discreet factor
wolbachia_lifespan <- s21data %>%
  mutate(temperature = as.factor(as.character(temperature)))

### the model
model_eff_wolb_lifespan_male = coxme::coxme(Surv(day, status)~wolbachia*temperature + (1|unique_id)+(1|

# significance of the model

```

```
car::Anova(model_eff_wolb_lifespan_male)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Surv(day, status)
##           Df    Chisq Pr(>Chisq)
## wolbachia      6 1014.21 < 2.2e-16 ***
## temperature    2 1532.29 < 2.2e-16 ***
## wolbachia:temperature 12  893.89 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# summary
summary(model_eff_wolb_lifespan_male)
```

```
## Cox mixed-effects model fit by maximum likelihood
## Data: dplyr::filter(wolbachia_lifespan, sex == "Male")
## events, n = 2077, 2101
## Iterations= 18 399
##           NULL Integrated   Fitted
## Log-likelihood -13803.38 -10295.14 -10014.3
##
##           Chisq      df p      AIC      BIC
## Integrated loglik 7016.47  22.00 0 6972.47 6848.42
## Penalized loglik 7578.15 182.04 0 7214.06 6187.57
##
## Model: Surv(day, status) ~ wolbachia * temperature + (1 | unique_id) + (1 | experimental_replicat
## Fixed coefficients
##           coef      exp(coef) se(coef)      z
## wolbachiaMelCS_b      0.2231523 1.250011e+00 0.4050964  0.55
## wolbachiaMelOctoless      5.0069281 1.494449e+02 0.3909519 12.81
## wolbachiaMelPop-High      2.9927646 1.994073e+01 0.3517815  8.51
## wolbachiaMelPop-Low      0.4438057 1.558628e+00 0.3955147  1.12
## wolbachiaMelPop2-High      4.6257638 1.020807e+02 0.3887612 11.90
## wolbachiaMelPop2-Low      0.6533417 1.921953e+00 0.3914004  1.67
## temperature25      6.5757659 7.174950e+02 0.4065583 16.17
## temperature29     11.2507037 7.693404e+04 0.4620515 24.35
## wolbachiaMelCS_b:temperature25      0.3030093 1.353927e+00 0.5612257  0.54
## wolbachiaMelOctoless:temperature25    -1.7361654 1.761947e-01 0.5499159 -3.16
## wolbachiaMelPop-High:temperature25    11.5779060 1.067138e+05 0.6212047 18.64
## wolbachiaMelPop-Low:temperature25      3.4179971 3.050825e+01 0.5433355  6.29
## wolbachiaMelPop2-High:temperature25    11.0456191 6.266881e+04 0.6464148 17.09
## wolbachiaMelPop2-Low:temperature25      8.2764778 3.930327e+03 0.5701665 14.52
## wolbachiaMelCS_b:temperature29      0.4192543 1.520827e+00 0.5478996  0.77
## wolbachiaMelOctoless:temperature29    -1.9043492 1.489195e-01 0.5343195 -3.56
## wolbachiaMelPop-High:temperature29     12.1502329 1.891381e+05 0.6331070 19.19
## wolbachiaMelPop-Low:temperature29      6.7405331 8.460116e+02 0.5796424 11.63
## wolbachiaMelPop2-High:temperature29    10.1010451 2.436846e+04 0.6562574 15.39
## wolbachiaMelPop2-Low:temperature29      9.8145768 1.829854e+04 0.6127083 16.02
##
##           p
## wolbachiaMelCS_b      5.8e-01
## wolbachiaMelOctoless      0.0e+00
## wolbachiaMelPop-High      0.0e+00
## wolbachiaMelPop-Low      2.6e-01
```

```

## wolbachiaMelPop2-High          0.0e+00
## wolbachiaMelPop2-Low          9.5e-02
## temperature25                  0.0e+00
## temperature29                  0.0e+00
## wolbachiaMelCS_b:temperature25 5.9e-01
## wolbachiaMel0ctoless:temperature25 1.6e-03
## wolbachiaMelPop-High:temperature25 0.0e+00
## wolbachiaMelPop-Low:temperature25 3.2e-10
## wolbachiaMelPop2-High:temperature25 0.0e+00
## wolbachiaMelPop2-Low:temperature25 0.0e+00
## wolbachiaMelCS_b:temperature29 4.4e-01
## wolbachiaMel0ctoless:temperature29 3.7e-04
## wolbachiaMelPop-High:temperature29 0.0e+00
## wolbachiaMelPop-Low:temperature29 0.0e+00
## wolbachiaMelPop2-High:temperature29 0.0e+00
## wolbachiaMelPop2-Low:temperature29 0.0e+00
##
## Random effects
## Group          Variable Std Dev  Variance
## unique_id      Intercept 0.7387308 0.5457232
## experimental_replicate Intercept 0.4957041 0.2457226
##
## multiple comparison
(multcomp_survival_males <- emmeans::emmeans(model_eff_wolb_lifespan_male, pairwise~wolbachia | tempera

## $emmeans
## temperature = 18:
## wolbachia      emmean    SE df asymp.LCL asymp.UCL
## NoWolb         -11.2828 0.343 Inf   -12.205   -10.361
## wMelCS_b        -11.0596 0.333 Inf   -11.954   -10.165
## wMel0ctoless    -6.2759 0.285 Inf    -7.042    -5.510
## wMelPop-High    -8.2900 0.282 Inf    -9.047    -7.533
## wMelPop-Low    -10.8390 0.331 Inf   -11.728    -9.950
## wMelPop2-High   -6.6570 0.294 Inf    -7.447    -5.867
## wMelPop2-Low   -10.6294 0.329 Inf   -11.514    -9.745
##
## temperature = 25:
## wolbachia      emmean    SE df asymp.LCL asymp.UCL
## NoWolb         -4.7070 0.283 Inf    -5.469    -3.945
## wMelCS_b        -4.1809 0.274 Inf    -4.918    -3.443
## wMel0ctoless    -1.4362 0.281 Inf    -2.193    -0.680
## wMelPop-High     9.8637 0.338 Inf     8.955    10.772
## wMelPop-Low     -0.8452 0.265 Inf    -1.558    -0.132
## wMelPop2-High   10.9644 0.351 Inf    10.021    11.908
## wMelPop2-Low     4.2228 0.258 Inf     3.528     4.917
##
## temperature = 29:
## wolbachia      emmean    SE df asymp.LCL asymp.UCL
## NoWolb         -0.0321 0.281 Inf    -0.788     0.723
## wMelCS_b         0.6103 0.273 Inf    -0.125     1.345
## wMel0ctoless     3.0705 0.268 Inf     2.349     3.792
## wMelPop-High    15.1109 0.399 Inf    14.038    16.184
## wMelPop-Low      7.1523 0.295 Inf     6.359     7.946
## wMelPop2-High   14.6947 0.401 Inf    13.615    15.775
## wMelPop2-Low    10.4358 0.340 Inf     9.521    11.350

```

```

##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
##
## $contrasts
## temperature = 18:
## contrast      estimate      SE  df z.ratio p.value
## NoWolb - wMelCS_b      -0.223 0.405 Inf  -0.551 1.0000
## NoWolb - wMelOctoless   -5.007 0.391 Inf -12.807 <.0001
## NoWolb - wMelPop-High   -2.993 0.352 Inf  -8.507 <.0001
## NoWolb - wMelPop-Low    -0.444 0.396 Inf  -1.122 1.0000
## NoWolb - wMelPop2-High  -4.626 0.389 Inf -11.899 <.0001
## NoWolb - wMelPop2-Low   -0.653 0.391 Inf  -1.669 0.6655
## wMelCS_b - wMelOctoless -4.784 0.379 Inf -12.624 <.0001
## wMelCS_b - wMelPop-High -2.770 0.347 Inf  -7.971 <.0001
## wMelCS_b - wMelPop-Low  -0.221 0.382 Inf  -0.577 1.0000
## wMelCS_b - wMelPop2-High -4.403 0.379 Inf -11.601 <.0001
## wMelCS_b - wMelPop2-Low -0.430 0.381 Inf  -1.130 1.0000
## wMelOctoless - wMelPop-High 2.014 0.332 Inf   6.073 <.0001
## wMelOctoless - wMelPop-Low  4.563 0.376 Inf  12.134 <.0001
## wMelOctoless - wMelPop2-High 0.381 0.379 Inf   1.004 1.0000
## wMelOctoless - wMelPop2-Low  4.354 0.373 Inf  11.662 <.0001
## wMelPop-High - wMelPop-Low  2.549 0.348 Inf   7.315 <.0001
## wMelPop-High - wMelPop2-High -1.633 0.364 Inf  -4.483 0.0001
## wMelPop-High - wMelPop2-Low  2.339 0.354 Inf   6.614 <.0001
## wMelPop-Low - wMelPop2-High -4.182 0.376 Inf -11.108 <.0001
## wMelPop-Low - wMelPop2-Low -0.210 0.388 Inf  -0.541 1.0000
## wMelPop2-High - wMelPop2-Low  3.972 0.376 Inf  10.569 <.0001
##
## temperature = 25:
## contrast      estimate      SE  df z.ratio p.value
## NoWolb - wMelCS_b      -0.526 0.388 Inf  -1.355 0.2302
## NoWolb - wMelOctoless   -3.271 0.388 Inf  -8.424 <.0001
## NoWolb - wMelPop-High   -14.571 0.508 Inf -28.670 <.0001
## NoWolb - wMelPop-Low    -3.862 0.372 Inf -10.386 <.0001
## NoWolb - wMelPop2-High  -15.671 0.521 Inf -30.061 <.0001
## NoWolb - wMelPop2-Low   -8.930 0.414 Inf -21.588 <.0001
## wMelCS_b - wMelOctoless -2.745 0.377 Inf  -7.289 <.0001
## wMelCS_b - wMelPop-High -14.045 0.500 Inf -28.091 <.0001
## wMelCS_b - wMelPop-Low  -3.336 0.370 Inf  -9.013 <.0001
## wMelCS_b - wMelPop2-High -15.145 0.513 Inf -29.514 <.0001
## wMelCS_b - wMelPop2-Low  -8.404 0.403 Inf -20.846 <.0001
## wMelOctoless - wMelPop-High -11.300 0.477 Inf -23.708 <.0001
## wMelOctoless - wMelPop-Low -0.591 0.375 Inf  -1.576 0.2302
## wMelOctoless - wMelPop2-High -12.401 0.491 Inf -25.259 <.0001
## wMelOctoless - wMelPop2-Low -5.659 0.374 Inf -15.149 <.0001
## wMelPop-High - wMelPop-Low  10.709 0.461 Inf  23.214 <.0001
## wMelPop-High - wMelPop2-High -1.101 0.370 Inf  -2.974 0.0088
## wMelPop-High - wMelPop2-Low  5.641 0.387 Inf  14.569 <.0001
## wMelPop-Low - wMelPop2-High -11.810 0.476 Inf -24.800 <.0001
## wMelPop-Low - wMelPop2-Low -5.068 0.355 Inf -14.256 <.0001
## wMelPop2-High - wMelPop2-Low  6.742 0.408 Inf  16.543 <.0001
##

```



```
## temperature = 29:
## contrast estimate SE df z.ratio p.value
## NoWolb - wMelCS_b -0.642 0.369 Inf -1.741 0.1635
## NoWolb - wMelOctoless -3.103 0.362 Inf -8.582 <.0001
## NoWolb - wMelPop-High -15.143 0.525 Inf -28.847 <.0001
## NoWolb - wMelPop-Low -7.184 0.423 Inf -16.971 <.0001
## NoWolb - wMelPop2-High -14.727 0.527 Inf -27.949 <.0001
## NoWolb - wMelPop2-Low -10.468 0.471 Inf -22.242 <.0001
## wMelCS_b - wMelOctoless -2.460 0.361 Inf -6.818 <.0001
## wMelCS_b - wMelPop-High -14.501 0.516 Inf -28.077 <.0001
## wMelCS_b - wMelPop-Low -6.542 0.412 Inf -15.884 <.0001
## wMelCS_b - wMelPop2-High -14.084 0.519 Inf -27.153 <.0001
## wMelCS_b - wMelPop2-Low -9.826 0.461 Inf -21.311 <.0001
## wMelOctoless - wMelPop-High -12.040 0.489 Inf -24.641 <.0001
## wMelOctoless - wMelPop-Low -4.082 0.372 Inf -10.973 <.0001
## wMelOctoless - wMelPop2-High -11.624 0.491 Inf -23.693 <.0001
## wMelOctoless - wMelPop2-Low -7.365 0.430 Inf -17.127 <.0001
## wMelPop-High - wMelPop-Low 7.959 0.408 Inf 19.523 <.0001
## wMelPop-High - wMelPop2-High 0.416 0.392 Inf 1.062 0.2883
## wMelPop-High - wMelPop2-Low 4.675 0.375 Inf 12.452 <.0001
## wMelPop-Low - wMelPop2-High -7.542 0.410 Inf -18.410 <.0001
## wMelPop-Low - wMelPop2-Low -3.284 0.339 Inf -9.681 <.0001
## wMelPop2-High - wMelPop2-Low 4.259 0.376 Inf 11.335 <.0001
##
```

```
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 21 tests
```

```
#
multcomp::cld(multcomp_survival_males$emmeans, adj="holm")
```

```
## temperature = 18:
## wolbachia emmean SE df asymp.LCL asymp.UCL .group
## NoWolb -11.2828 0.343 Inf -12.205 -10.361 1
## wMelCS_b -11.0596 0.333 Inf -11.954 -10.165 1
## wMelPop-Low -10.8390 0.331 Inf -11.728 -9.950 1
## wMelPop2-Low -10.6294 0.329 Inf -11.514 -9.745 1
## wMelPop-High -8.2900 0.282 Inf -9.047 -7.533 2
## wMelPop2-High -6.6570 0.294 Inf -7.447 -5.867 3
## wMelOctoless -6.2759 0.285 Inf -7.042 -5.510 3
##
```

```
## temperature = 25:
## wolbachia emmean SE df asymp.LCL asymp.UCL .group
## NoWolb -4.7070 0.283 Inf -5.469 -3.945 1
## wMelCS_b -4.1809 0.274 Inf -4.918 -3.443 1
## wMelOctoless -1.4362 0.281 Inf -2.193 -0.680 2
## wMelPop-Low -0.8452 0.265 Inf -1.558 -0.132 2
## wMelPop2-Low 4.2228 0.258 Inf 3.528 4.917 3
## wMelPop-High 9.8637 0.338 Inf 8.955 10.772 4
## wMelPop2-High 10.9644 0.351 Inf 10.021 11.908 5
##
```

```
## temperature = 29:
## wolbachia emmean SE df asymp.LCL asymp.UCL .group
## NoWolb -0.0321 0.281 Inf -0.788 0.723 1
## wMelCS_b 0.6103 0.273 Inf -0.125 1.345 1
## wMelOctoless 3.0705 0.268 Inf 2.349 3.792 2
```

```
## wMelPop-Low      7.1523 0.295 Inf      6.359      7.946      3
## wMelPop2-Low     10.4358 0.340 Inf      9.521     11.350      4
## wMelPop2-High    14.6947 0.401 Inf     13.615     15.775      5
## wMelPop-High     15.1109 0.399 Inf     14.038     16.184      5
```

```
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
## P value adjustment: holm method for 21 tests
## significance level used: alpha = 0.05
```

```
#compare temperatures by Wolbachia
```

```
(multcomp_survival_males_by_wolb <- emmeans::emmeans(model_eff_wolb_lifespan_male, consec~temperature |
```

```
## $emmeans
```

```
## wolbachia = NoWolb:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                -11.2828 0.343 Inf  -12.1034  -10.462
## 25                 -4.7070 0.283 Inf   -5.3849   -4.029
## 29                 -0.0321 0.281 Inf   -0.7044    0.640
```

```
##
```

```
## wolbachia = wMelCS_b:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                -11.0596 0.333 Inf  -11.8558  -10.263
## 25                 -4.1809 0.274 Inf   -4.8372   -3.524
## 29                 0.6103 0.273 Inf   -0.0436    1.264
```

```
##
```

```
## wolbachia = wMelOctoless:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                 -6.2759 0.285 Inf   -6.9573   -5.594
## 25                 -1.4362 0.281 Inf   -2.1096   -0.763
## 29                 3.0705 0.268 Inf    2.4283    3.713
```

```
##
```

```
## wolbachia = wMelPop-High:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                 -8.2900 0.282 Inf   -8.9640   -7.616
## 25                 9.8637 0.338 Inf    9.0550   10.672
## 29                 15.1109 0.399 Inf   14.1561   16.066
```

```
##
```

```
## wolbachia = wMelPop-Low:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                -10.8390 0.331 Inf  -11.6303  -10.048
## 25                 -0.8452 0.265 Inf   -1.4799   -0.211
## 29                 7.1523 0.295 Inf    6.4462    7.858
```

```
##
```

```
## wolbachia = wMelPop2-High:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                 -6.6570 0.294 Inf   -7.3604   -5.954
## 25                 10.9644 0.351 Inf   10.1250   11.804
## 29                 14.6947 0.401 Inf   13.7338   15.656
```

```
##
```

```
## wolbachia = wMelPop2-Low:
```

```
## temperature      emmean      SE df asymp.LCL asymp.UCL
## 18                -10.6294 0.329 Inf  -11.4164   -9.843
## 25                 4.2228 0.258 Inf    3.6049    4.841
```

```

## 29          10.4358 0.340 Inf    9.6219    11.250
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 3 estimates
##
## $contrasts
## wolbachia = NoWolb:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18          6.58 0.407 Inf 16.174 <.0001
## 29 - 25          4.67 0.391 Inf 11.967 <.0001
##
## wolbachia = wMelCS_b:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18          6.88 0.398 Inf 17.273 <.0001
## 29 - 25          4.79 0.380 Inf 12.622 <.0001
##
## wolbachia = wMelOctoless:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18          4.84 0.399 Inf 12.133 <.0001
## 29 - 25          4.51 0.370 Inf 12.185 <.0001
##
## wolbachia = wMelPop-High:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18         18.15 0.523 Inf 34.737 <.0001
## 29 - 25          5.25 0.390 Inf 13.455 <.0001
##
## wolbachia = wMelPop-Low:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18          9.99 0.435 Inf 22.953 <.0001
## 29 - 25          8.00 0.419 Inf 19.082 <.0001
##
## wolbachia = wMelPop2-High:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18         17.62 0.536 Inf 32.891 <.0001
## 29 - 25          3.73 0.379 Inf  9.836 <.0001
##
## wolbachia = wMelPop2-Low:
## contrast estimate    SE  df z.ratio p.value
## 25 - 18         14.85 0.468 Inf 31.744 <.0001
## 29 - 25          6.21 0.396 Inf 15.682 <.0001
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 2 tests
#compare contrasts between Wolbachia variants and wMelCS line
contrast(multcomp_survival_males_by_wolb$contrasts,method="trt.vs.ctrl1", ref = 2, by="contrast")

## contrast = 25 - 18:
## contrast1          estimate    SE  df z.ratio p.value
## NoWolb - wMelCS_b      -0.303 0.561 Inf -0.540 0.9569
## wMelOctoless - wMelCS_b -2.039 0.543 Inf -3.758 0.0010
## wMelPop-High - wMelCS_b 11.275 0.611 Inf 18.450 <.0001
## wMelPop-Low - wMelCS_b   3.115 0.532 Inf  5.853 <.0001
## wMelPop2-High - wMelCS_b 10.743 0.639 Inf 16.807 <.0001

```

```
## wMelPop2-Low - wMelCS_b      7.973 0.555 Inf 14.377 <.0001
##
## contrast = 29 - 25:
## contrast1      estimate      SE  df z.ratio p.value
## NoWolb - wMelCS_b      -0.116 0.534 Inf -0.218 0.9970
## wMelOctoless - wMelCS_b  -0.284 0.513 Inf -0.555 0.9535
## wMelPop-High - wMelCS_b   0.456 0.544 Inf  0.838 0.8561
## wMelPop-Low - wMelCS_b   3.206 0.541 Inf  5.924 <.0001
## wMelPop2-High - wMelCS_b -1.061 0.536 Inf -1.978 0.2062
## wMelPop2-Low - wMelCS_b   1.422 0.550 Inf  2.584 0.0495
##
## Results are given on the log (not the response) scale.
## P value adjustment: dunnettx method for 6 tests
```

### ## Females

```
model_eff_wolb_lifespan_female = coxme::coxme(Surv(day, status)~wolbachia + (1|unique_id), data=dplyr::
```

### # significance of the model

```
car::Anova(model_eff_wolb_lifespan_female)
```

### ## Analysis of Deviance Table (Type II tests)

```
##
```

```
## Response: Surv(day, status)
```

```
##           Df  Chisq Pr(>Chisq)
```

```
## wolbachia  6 639.37 < 2.2e-16 ***
```

```
## ---
```

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

### # summary

```
summary(model_eff_wolb_lifespan_female)
```

```
## Cox mixed-effects model fit by maximum likelihood
```

```
## Data: dplyr::filter(wolbachia_lifespan, sex == "Female")
```

```
## events, n = 694, 700
```

```
## Iterations= 5 56
```

```
##           NULL Integrated Fitted
```

```
## Log-likelihood -3852.011 -3003.804 -2906.4
```

```
##
```

```
##           Chisq      df p      AIC      BIC
```

```
## Integrated loglik 1696.41  7.00 0 1682.41 1650.62
```

```
## Penalized loglik 1891.22 62.13 0 1766.97 1484.77
```

```
##
```

```
## Model: Surv(day, status) ~ wolbachia + (1 | unique_id)
```

```
## Fixed coefficients
```

```
##           coef      exp(coef) se(coef)      z  p
## wolbachiawMelCS_b      0.5634496 1.756722e+00 0.4427408  1.27 0.2
## wolbachiawMelOctoless  3.7501296 4.252659e+01 0.3583729 10.46 0.0
## wolbachiawMelPop-High 12.7970025 3.611333e+05 0.5966993 21.45 0.0
## wolbachiawMelPop-Low   7.6258617 2.050547e+03 0.4910288 15.53 0.0
## wolbachiawMelPop2-High 13.8827606 1.069563e+06 0.6113314 22.71 0.0
## wolbachiawMelPop2-Low   8.7000803 6.003394e+03 0.5249679 16.57 0.0
##
```

```
## Random effects
```

```
## Group      Variable Std Dev  Variance
```

```
## unique_id Intercept 0.8003335 0.6405337
```

```
## multiple comparison
(multiple_comparison_lifespan_female=emmeans::emmeans(model_eff_wolb_lifespan_female, list(pairwise~wolb

## $`emmeans of wolbachia`
##   wolbachia      emmean      SE df asymp.LCL asymp.UCL
##   NoWolb        -6.760 0.350 Inf    -7.702    -5.82
##   wMelCS_b       -6.196 0.351 Inf    -7.140    -5.25
##   wMelOctoless   -3.010 0.246 Inf    -3.673    -2.35
##   wMelPop-High    6.037 0.347 Inf     5.105     6.97
##   wMelPop-Low     0.866 0.267 Inf     0.148     1.58
##   wMelPop2-High   7.123 0.361 Inf     6.152     8.09
##   wMelPop2-Low    1.940 0.292 Inf     1.155     2.73
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
##
## $`pairwise differences of wolbachia`
##   contrast                estimate      SE df z.ratio p.value
##   NoWolb - wMelCS_b         -0.563 0.443 Inf  -1.273 0.2031
##   NoWolb - wMelOctoless      -3.750 0.358 Inf -10.464 <.0001
##   NoWolb - wMelPop-High      -12.797 0.597 Inf -21.446 <.0001
##   NoWolb - wMelPop-Low       -7.626 0.491 Inf -15.530 <.0001
##   NoWolb - wMelPop2-High     -13.883 0.611 Inf -22.709 <.0001
##   NoWolb - wMelPop2-Low      -8.700 0.525 Inf -16.573 <.0001
##   wMelCS_b - wMelOctoless     -3.187 0.373 Inf  -8.549 <.0001
##   wMelCS_b - wMelPop-High     -12.234 0.595 Inf -20.556 <.0001
##   wMelCS_b - wMelPop-Low      -7.062 0.488 Inf -14.484 <.0001
##   wMelCS_b - wMelPop2-High    -13.319 0.610 Inf -21.840 <.0001
##   wMelCS_b - wMelPop2-Low     -8.137 0.524 Inf -15.526 <.0001
##   wMelOctoless - wMelPop-High  -9.047 0.512 Inf -17.683 <.0001
##   wMelOctoless - wMelPop-Low  -3.876 0.390 Inf  -9.936 <.0001
##   wMelOctoless - wMelPop2-High -10.133 0.528 Inf -19.179 <.0001
##   wMelOctoless - wMelPop2-Low  -4.950 0.422 Inf -11.737 <.0001
##   wMelPop-High - wMelPop-Low   5.171 0.437 Inf  11.832 <.0001
##   wMelPop-High - wMelPop2-High -1.086 0.428 Inf  -2.535 0.0293
##   wMelPop-High - wMelPop2-Low  4.097 0.455 Inf   9.005 <.0001
##   wMelPop-Low - wMelPop2-High  -6.257 0.465 Inf -13.458 <.0001
##   wMelPop-Low - wMelPop2-Low  -1.074 0.416 Inf  -2.584 0.0293
##   wMelPop2-High - wMelPop2-Low  5.183 0.449 Inf  11.553 <.0001
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 21 tests
#
multcomp::cld(multiple_comparison_lifespan_female$emmeans, adj="holm")

##   wolbachia      emmean      SE df asymp.LCL asymp.UCL .group
##   NoWolb        -6.760 0.350 Inf    -7.702    -5.82    1
##   wMelCS_b       -6.196 0.351 Inf    -7.140    -5.25    1
##   wMelOctoless   -3.010 0.246 Inf    -3.673    -2.35    2
##   wMelPop-Low     0.866 0.267 Inf     0.148     1.58    3
##   wMelPop2-Low    1.940 0.292 Inf     1.155     2.73    4
##   wMelPop-High    6.037 0.347 Inf     5.105     6.97    5
##   wMelPop2-High   7.123 0.361 Inf     6.152     8.09    6
```

```

##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
## P value adjustment: holm method for 21 tests
## significance level used: alpha = 0.05

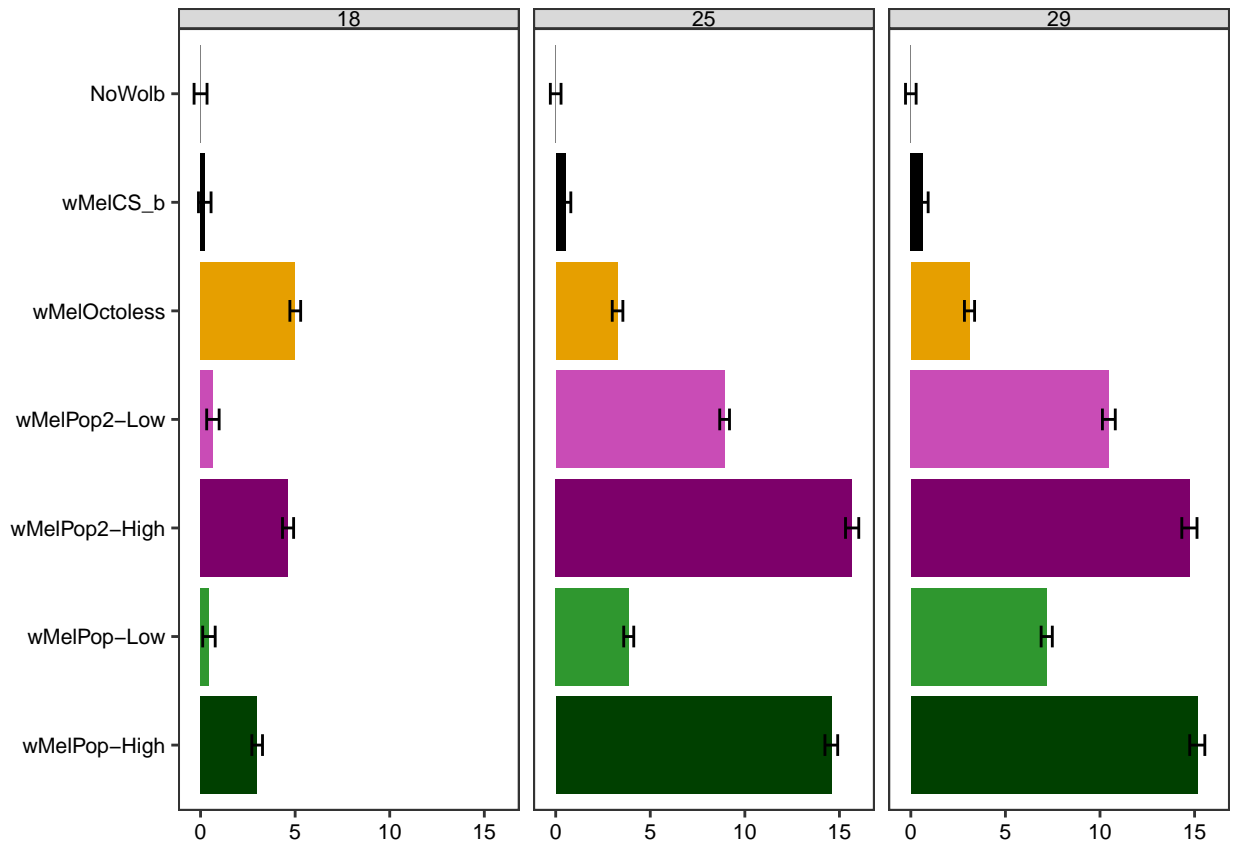
### extracting coefficients from the models
## for males
coeff_cox_model_lifespan_males = as.data.frame(multcomp_survival_males$emmeans)%>%
  group_by(temperature, wolbachia)%>%
  summarise(emmean=emmean, SE=SE)%>%
  mutate(normalized_emmean=emmean-emmean[wolbachia=="NoWolb"],
         wolbachia=factor(wolbachia, levels=c("NoWolb", "wMelCS_b", "wMelOctoless", "wMelPop2-Low", "wMelPop2-High")))

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

## for females
coeff_cox_model_lifespan_females = as.data.frame(multiple_comparison_lifespan_female$emmeans)%>%
  group_by(wolbachia)%>%
  summarise(emmean=emmean, SE=SE)%>%
  mutate(normalized_emmean=emmean-emmean[wolbachia=="NoWolb"],
         wolbachia=factor(wolbachia, levels=c("NoWolb", "wMelCS_b", "wMelOctoless", "wMelPop2-Low", "wMelPop2-High")))

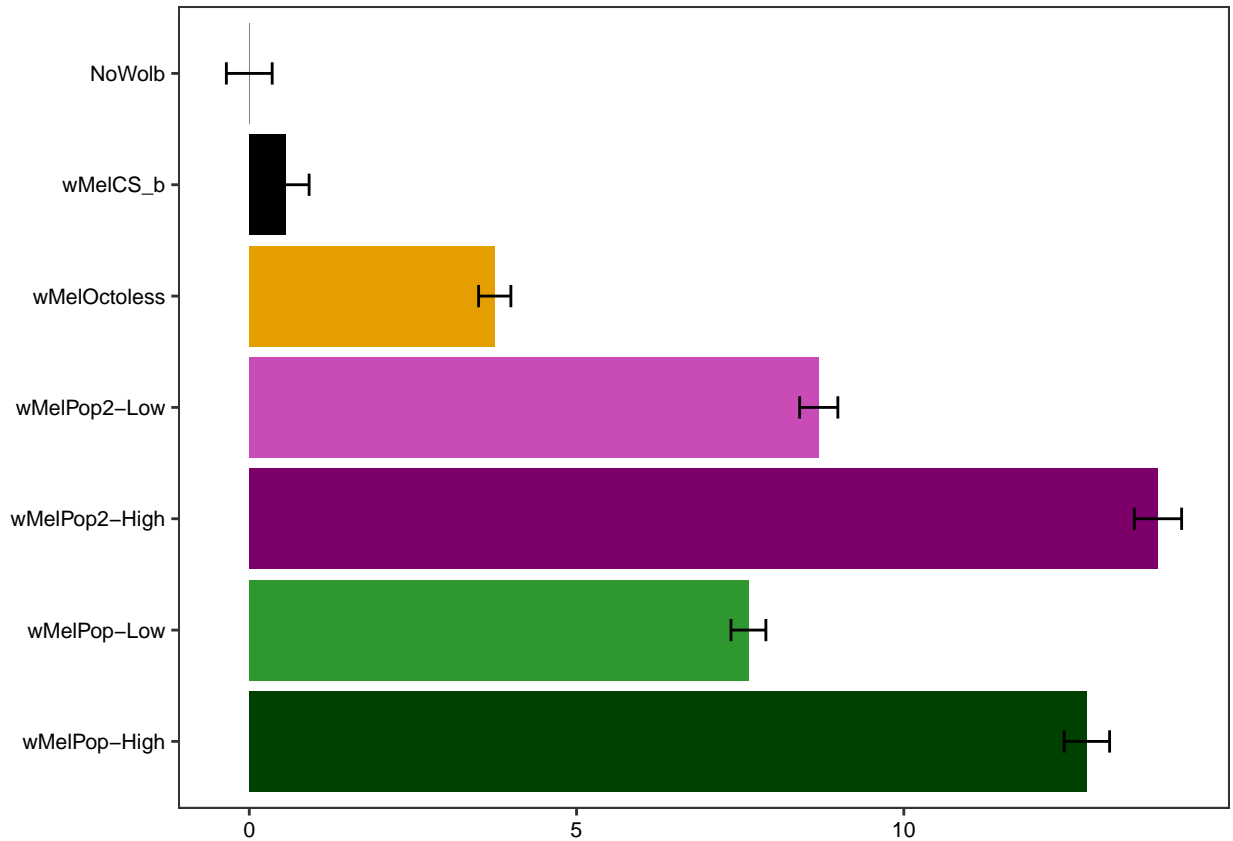
### plots
# coefficients of males
ggplot(coeff_cox_model_lifespan_males, aes(x=reorder(wolbachia, desc(wolbachia)), y=normalized_emmean,
      y_min=normalized_emmean-SE, y_max=normalized_emmean+SE)) +
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=normalized_emmean-SE, ymax=normalized_emmean+SE), width=0.2, size=0.5)+
  facet_grid(.~temperature)+
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
                    labels=c("NoWolb", "wMelCS_b", "wMelOctoless", "wMelPop2-Low", "wMelPop2-High")) +
  coord_flip() +theme(legend.position="none")+
  scale_y_continuous(limits=c(NA, NA))

```



*# coefficients of females*

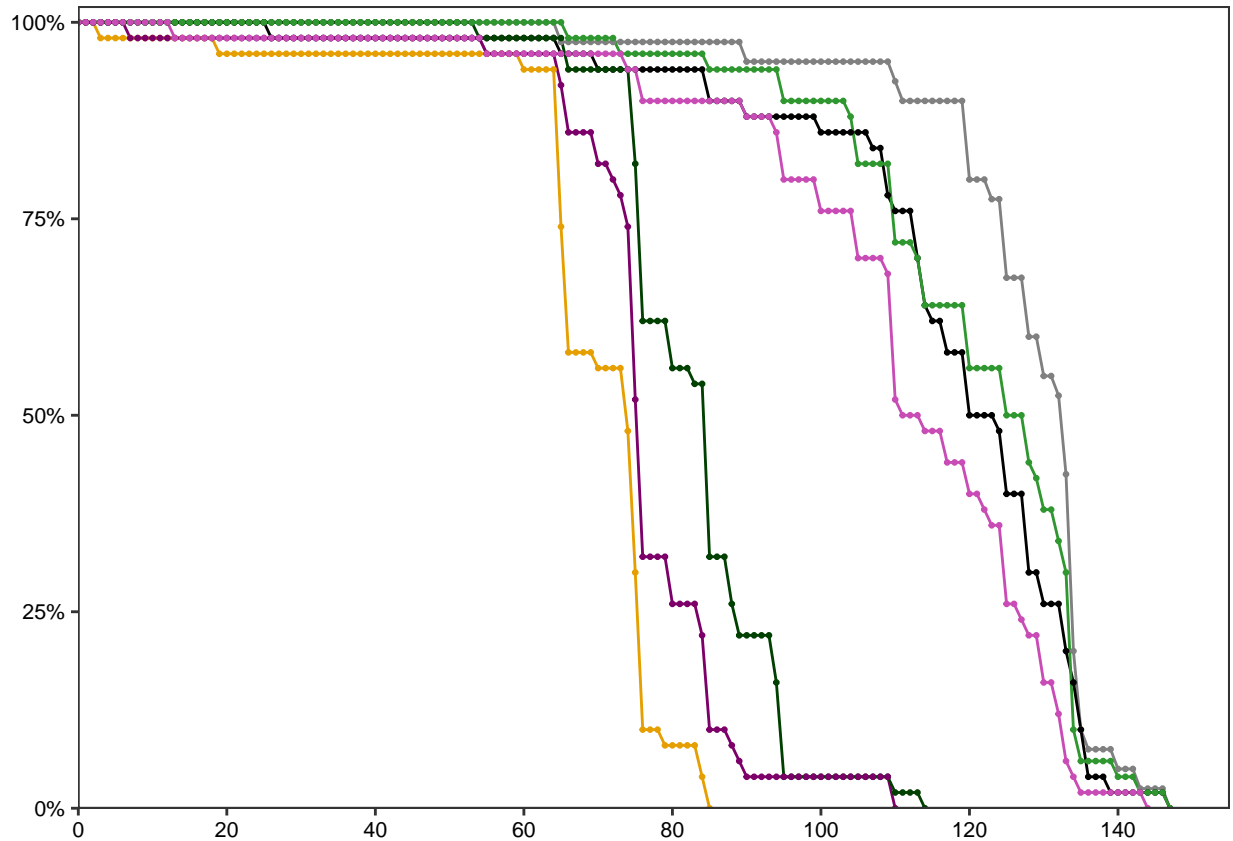
```
ggplot(coeff_cox_model_lifespan_females, aes(x=reorder(wolbachia, desc(wolbachia)), y=normalized_emmean)) +
  geom_col() + theme_figs() +
  geom_errorbar(aes(ymin=normalized_emmean-SE, ymax=normalized_emmean+SE), width=0.2, size=0.5) +
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")) +
  coord_flip() + theme(legend.position="none") +
  scale_y_continuous(limits=c(NA, NA))
```



**Fig 5 and S14 Fig panels male lifespan**

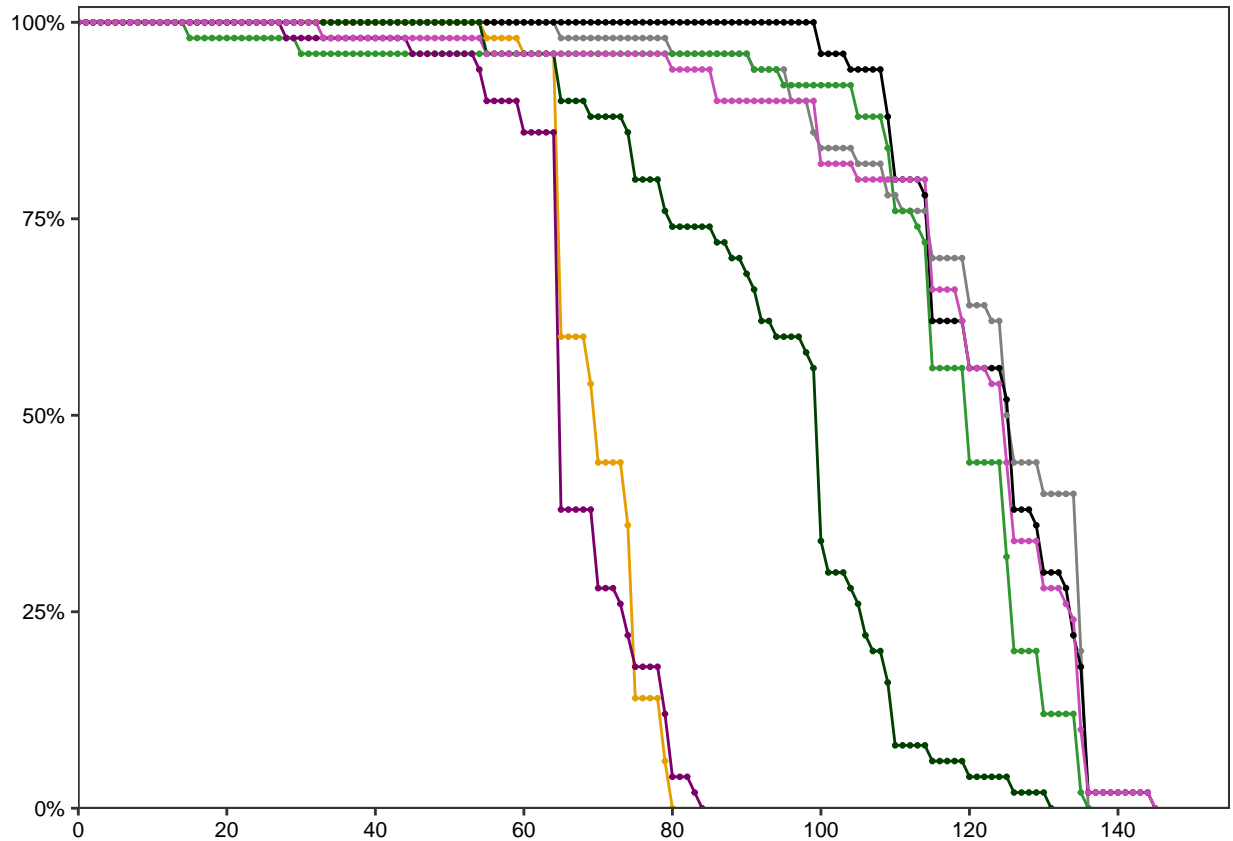
```
### plots
## Males
# 18°C
(fig5a=ggplot(data=filter(results_lifespan_all_flies_dataset, temperature=="18" & experimental_replicate=="1")+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")+
  scale_x_continuous(expand=c(0,0), limits=c(0, 155), breaks=c(0,20,40,60,80,100,120,140,160))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```





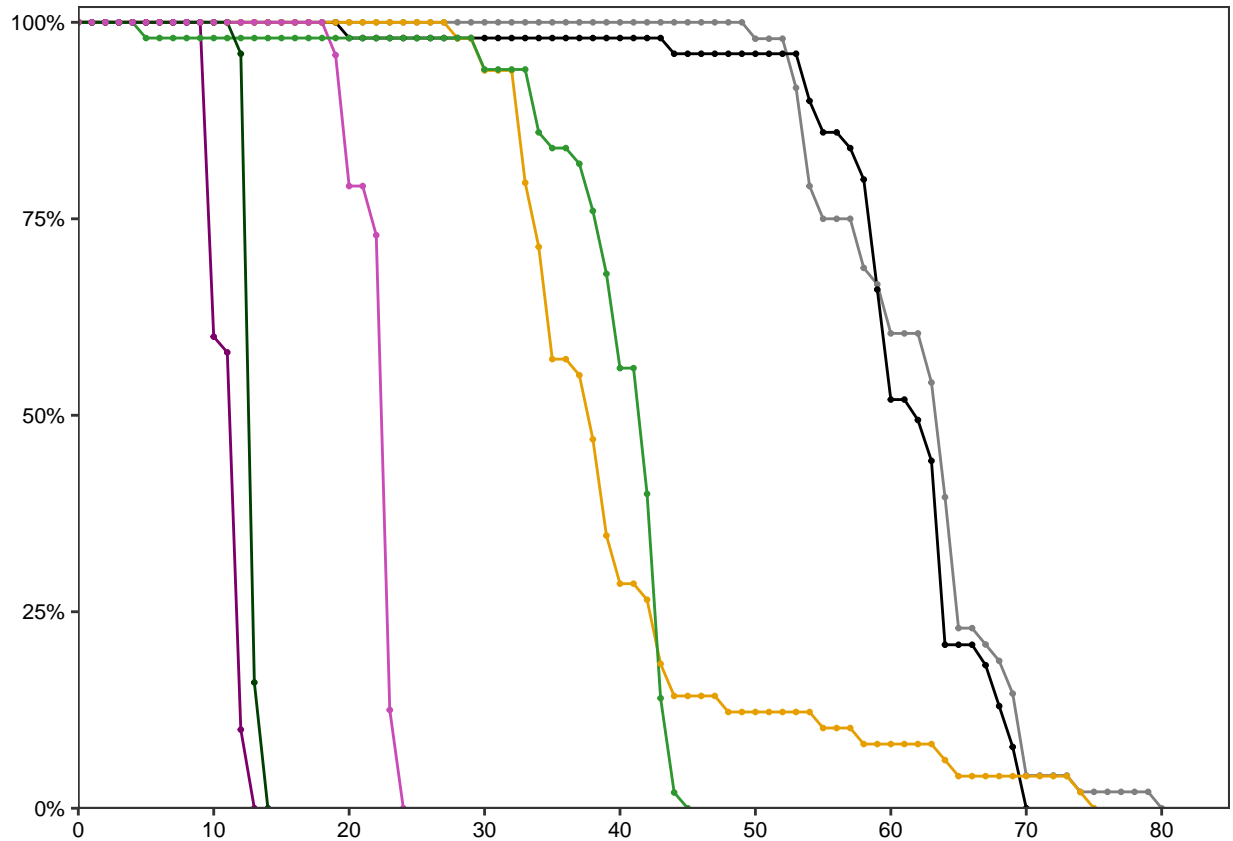
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5A.pdf", plot=fig5a, device="pdf", dpi=300, units="in", width=230/96,
}

(s14afig=ggplot(data=filter(results_lifespan_all_flies_dataset, temperature=="18" & experimental_replicates=="18"),
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
  scale_x_continuous(expand=c(0,0), limits=c(0, 155), breaks=c(0,20,40,60,80,100,120,140,160))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



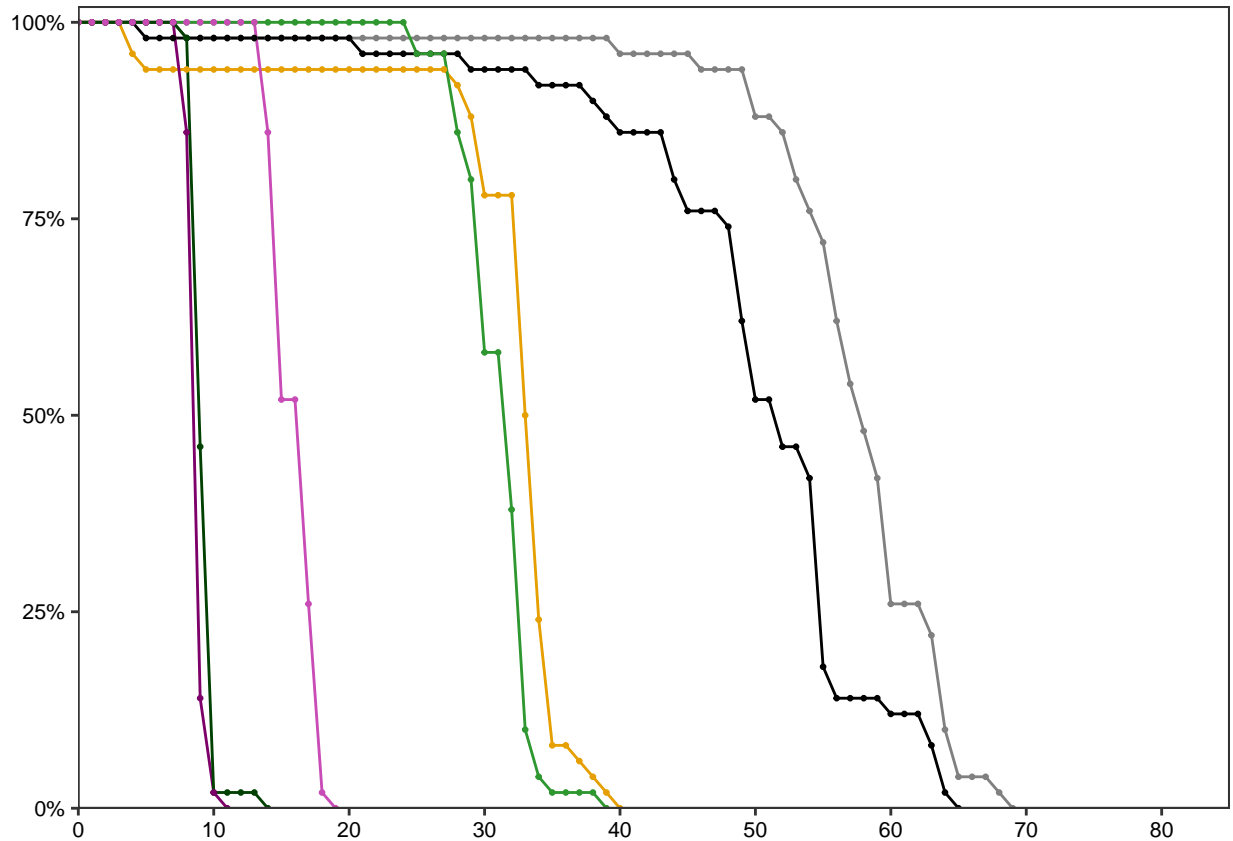
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S14A Fig.pdf", plot=s14afig, device="pdf", dpi=300, units="in", width=230, height=150)
}

# 25°C
(fig5b=ggplot(data=filter(results_lifespan_all_flies_dataset, sex!="Female" & temperature=="25" & experiment=="S14A"),
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000", "#000000"),
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



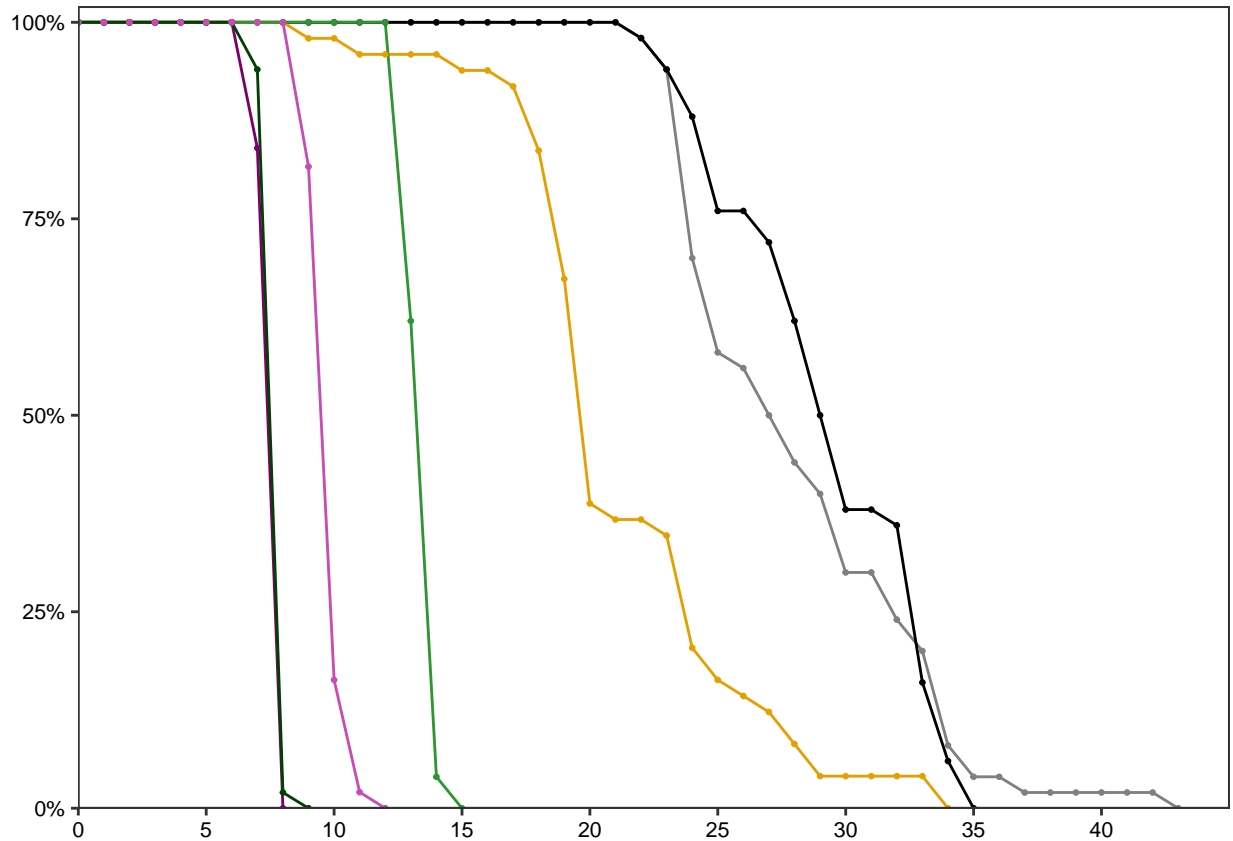
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5B.pdf", plot=fig4b, device="pdf", dpi=300, units="in", width=230/96,
}

(s14bfig=ggplot(data=filter(results_lifespan_all_flies_dataset, sex!="Female" & temperature=="25" & exp
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



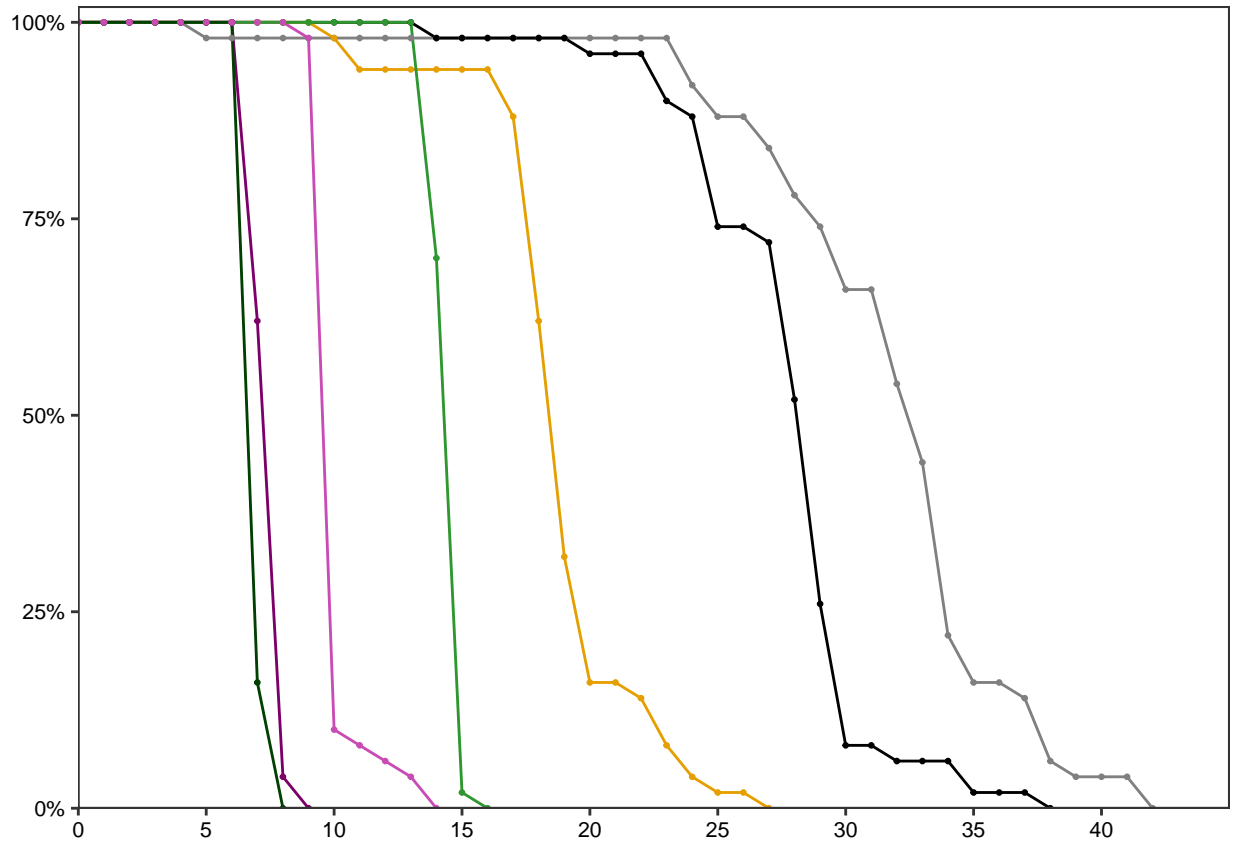
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S14B Fig.pdf", plot=s14bfig, device="pdf", dpi=300, units="in", width=230, height=150)
}

# 29°C
(fig5c=ggplot(data=filter(results_lifespan_all_flies_dataset, sex!="Female" & temperature=="29" & experiment=="S14B"),
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
  scale_x_continuous(expand=c(0,0), limits=c(0,45), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



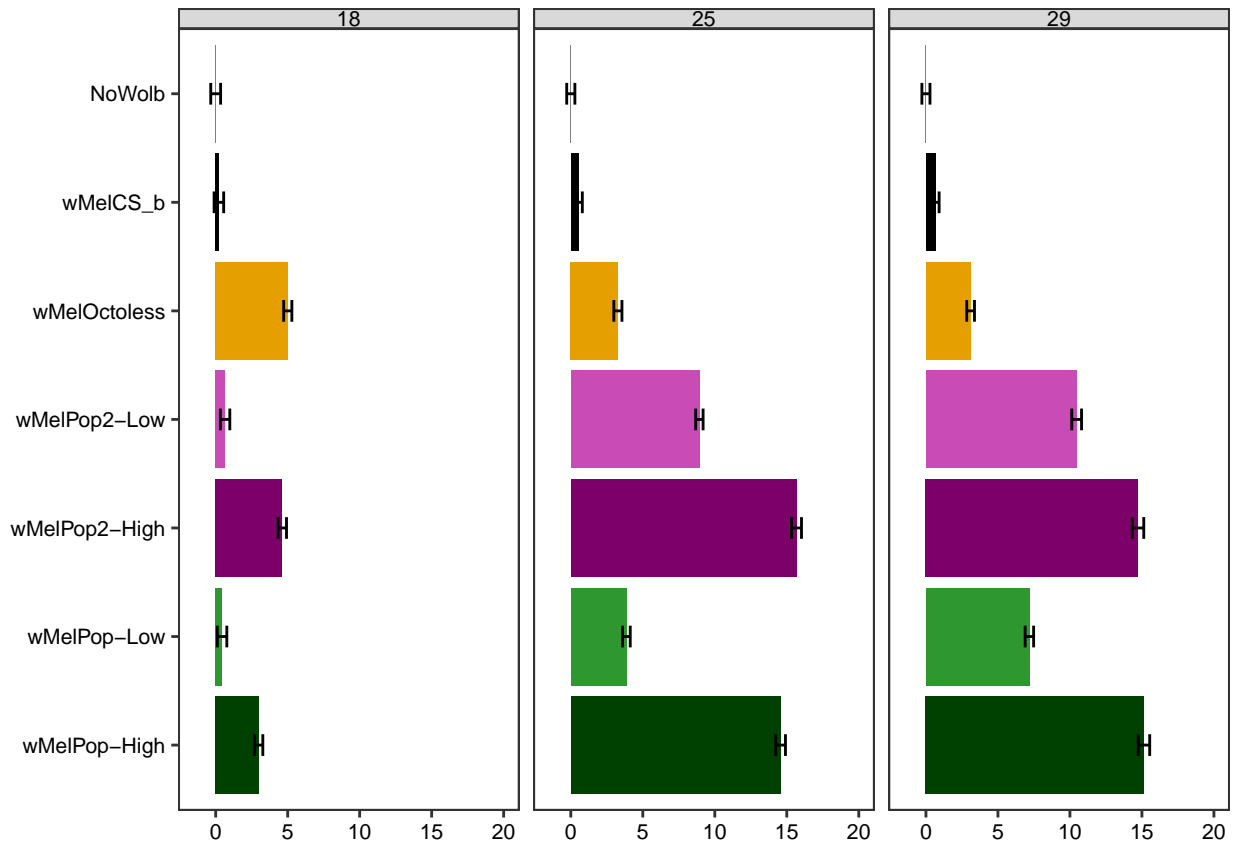
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5C.pdf", plot=fig5c, device="pdf", dpi=300, units="in", width=230/96,
}

(s14cfig=ggplot(data=filter(results_lifespan_all_flies_dataset, sex!="Female" & temperature=="29" & exp
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
  scale_x_continuous(expand=c(0,0), limits=c(0,45), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S14C Fig.pdf", plot=s14cfig, device="pdf", dpi=300, units="in", width=230, height=150)
}

## the coefficients
(fig5d=ggplot(coeff_cox_model_lifespan_males, aes(x=reorder(wolbachia, desc(wolbachia)), y=normalized_emmean, yerr=normalized_emse)) +
  geom_col() + theme_figs() +
  geom_errorbar(aes(ymin=normalized_emmean-SE, ymax=normalized_emmean+SE), width=0.2, size=0.5) +
  facet_grid(.~temperature) +
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")) +
  theme(legend.position="none") + coord_flip() +
  scale_y_continuous(limits=c(-1.5, 20), breaks=seq(0,20,5))
```



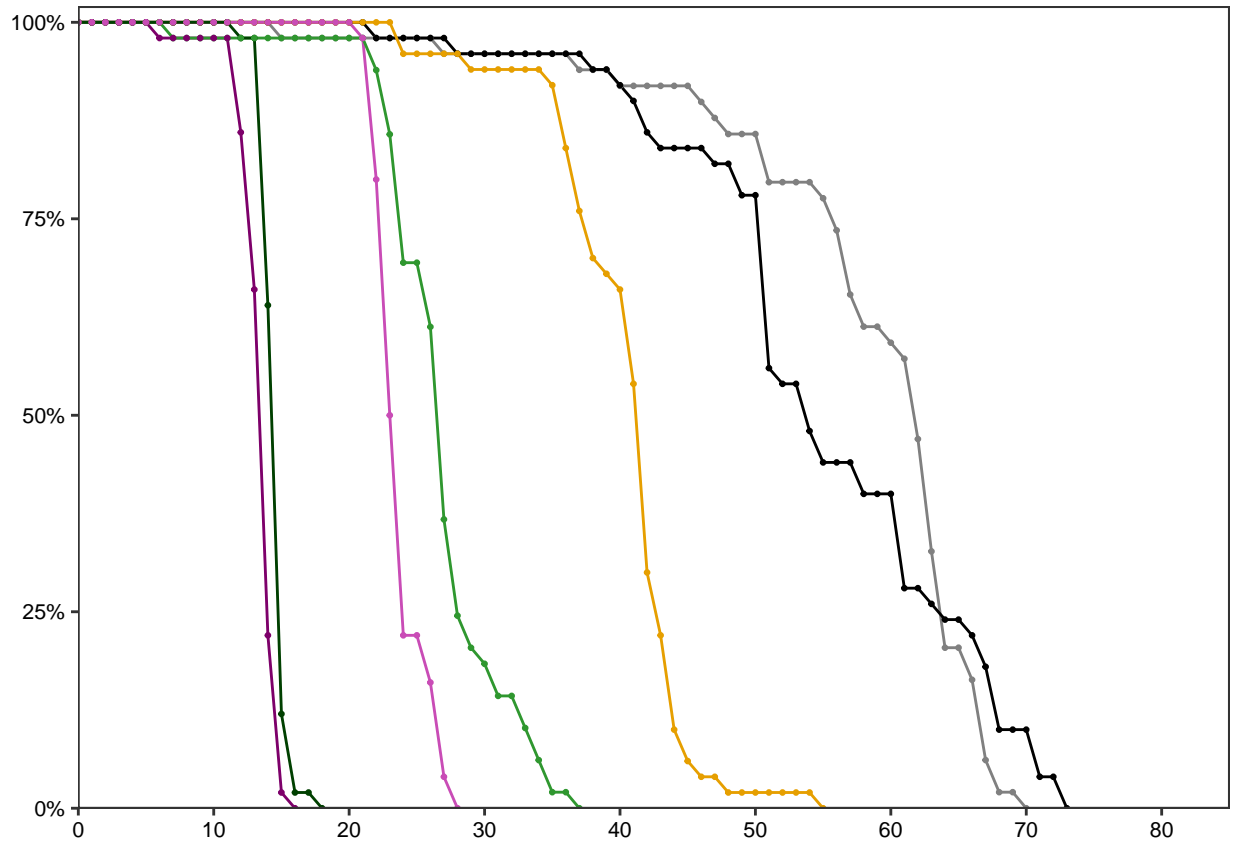
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5D.pdf", plot=fig5d, device="pdf", dpi=300, units="in", width=600/96,
}

```

## S14D-E Fig | Wolbachia life-shortening phenotype :: Females

```
### plots
## replicate 1
(s14drep1fig=ggplot(data=filter(results_lifespan_all_flies_dataset, sex=="Female" & experimental_replicate==1),
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))

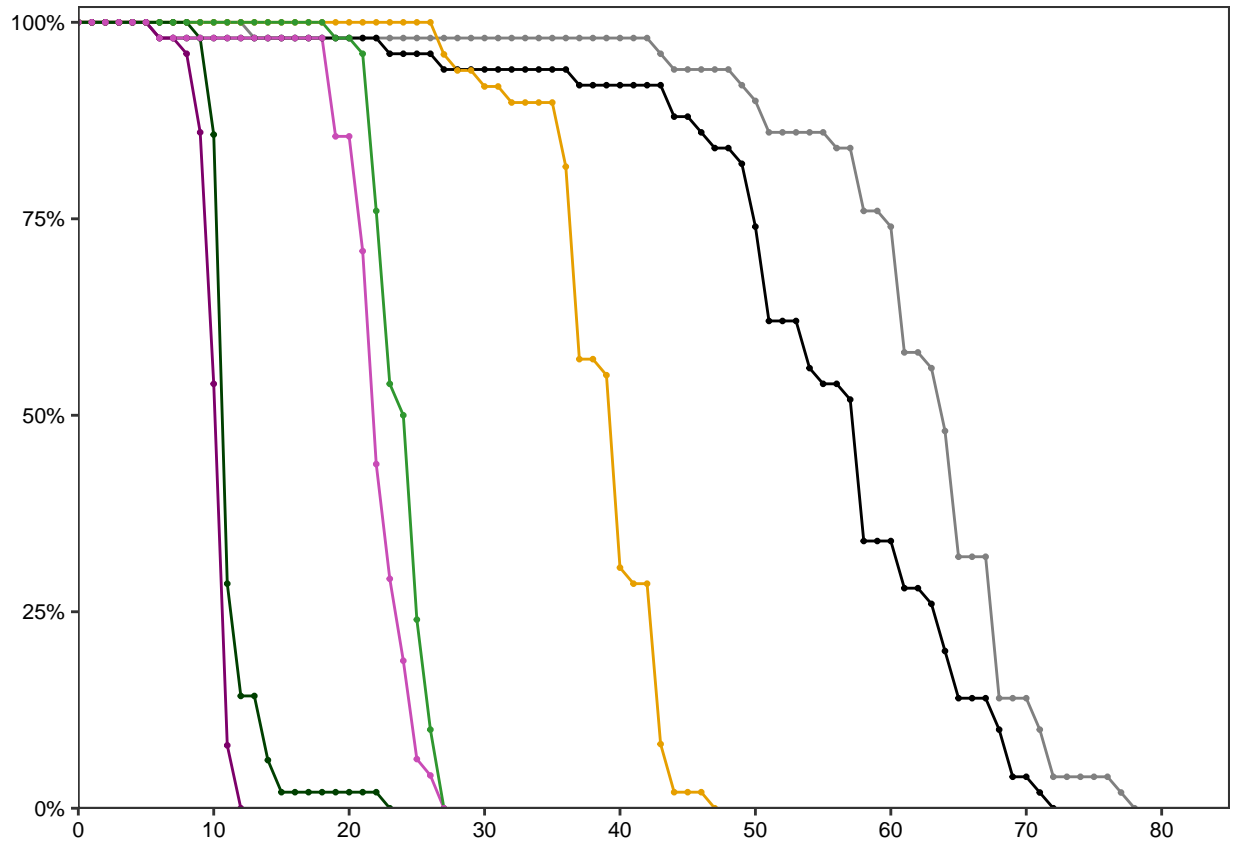
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S14Drep1 Fig.pdf", plot=s14drep1fig, device="pdf", dpi=300, units="in", w
}

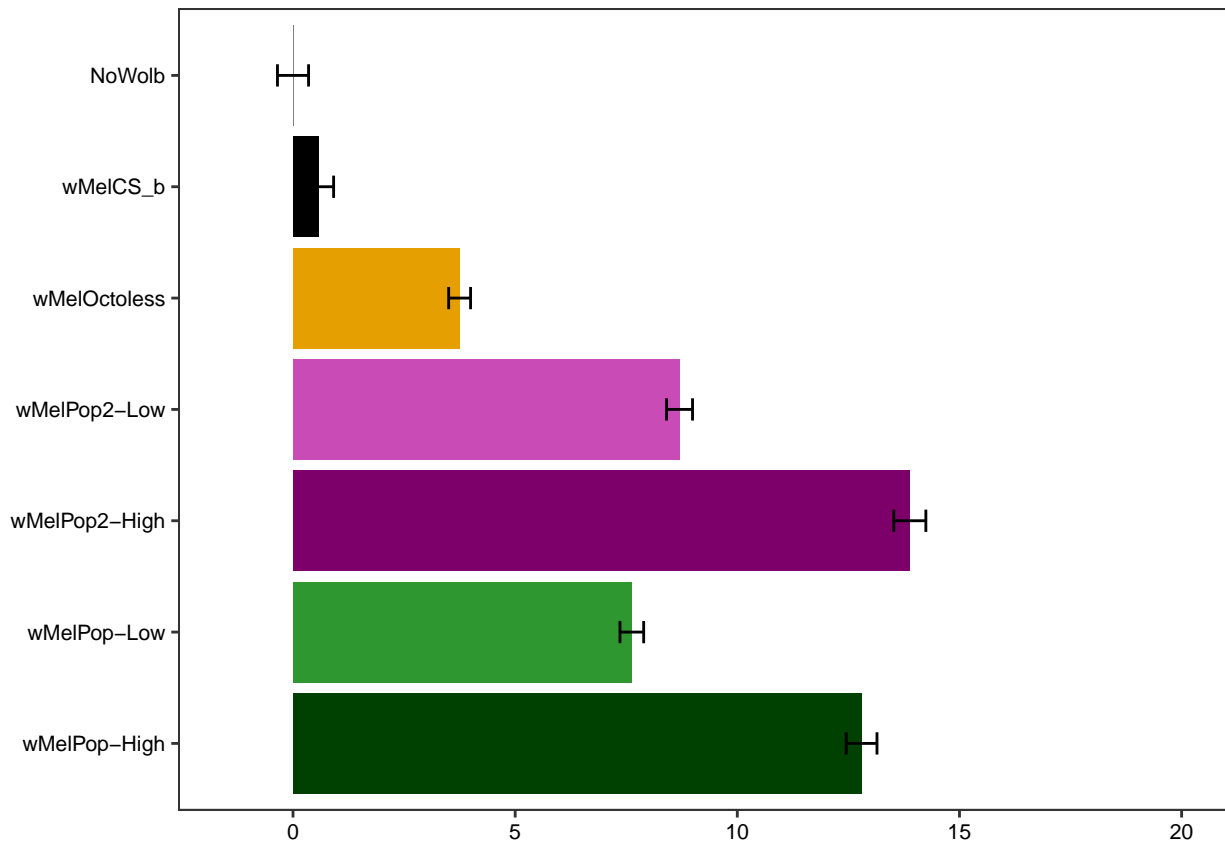
## replicate 2
(s14drep2fig=ggplot(data=filter(results_lifespan_all_flies_dataset, sex=="Female" & experimental_replic
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000
  scale_x_continuous(expand=c(0,0), limits=c(0,85), breaks=c(0,10,20,30,40,50,60,70,80))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```





```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S14Drep2 Fig.pdf", plot=s14drep2fig, device="pdf", dpi=300, units="in", w
}

# coefficients
(s14efig=ggplot(coeff_cox_model_lifespan_females, aes(x=reorder(wolbachia, desc(wolbachia)), y=normalized
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=normalized_emmean-SE, ymax=normalized_emmean+SE), width=0.2, size=0.5)+
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000
  theme(legend.position="none")+ coord_flip()+
  scale_y_continuous(limits=c(-1.5, 20), breaks=seq(0,20,5)))
```



```
# saving the pdf
if(save_files==TRUE)
{
  gggsave(filename="../results/S14E Fig.pdf", plot=s14efig, device="pdf", dpi=300, units="in", width=260,
}

```

## S15 Fig | Bidirectional crosses

```
### S22 Data
s22data=read.csv("../original_data/S22 Data.csv", stringsAsFactors = TRUE)

## extracting Kaplan estimates (+/- confidence intervals) from the data
results_lifespan_bidirectional_crosses=survfit(Surv(day,status)~cross+experimental_replicate, data=s22data)

# conferting the dataste to a data frame
results_lifespan_bidirectional_dataset=as.data.frame(summary(results_lifespan_bidirectional_crosses, times=1000))

# adding new variables to the dataset
results_lifespan_bidirectional_dataset=results_lifespan_bidirectional_dataset%>%
  tidyr::separate(strata, c("mating_direction", "experimental_replicate"), ", ", remove=FALSE)%>%
  dplyr::mutate(mating_direction=str_replace(mating_direction, "cross=", ""),
               experimental_replicate=str_replace(experimental_replicate, "experimental_replicate=", ""))
  tidyr::separate(mating_direction, c("inherited_wolbachia", NA), " X ", remove=FALSE)

# converting variables to factor
results_lifespan_bidirectional_dataset$mating_direction=factor(results_lifespan_bidirectional_dataset$mating_direction)

```

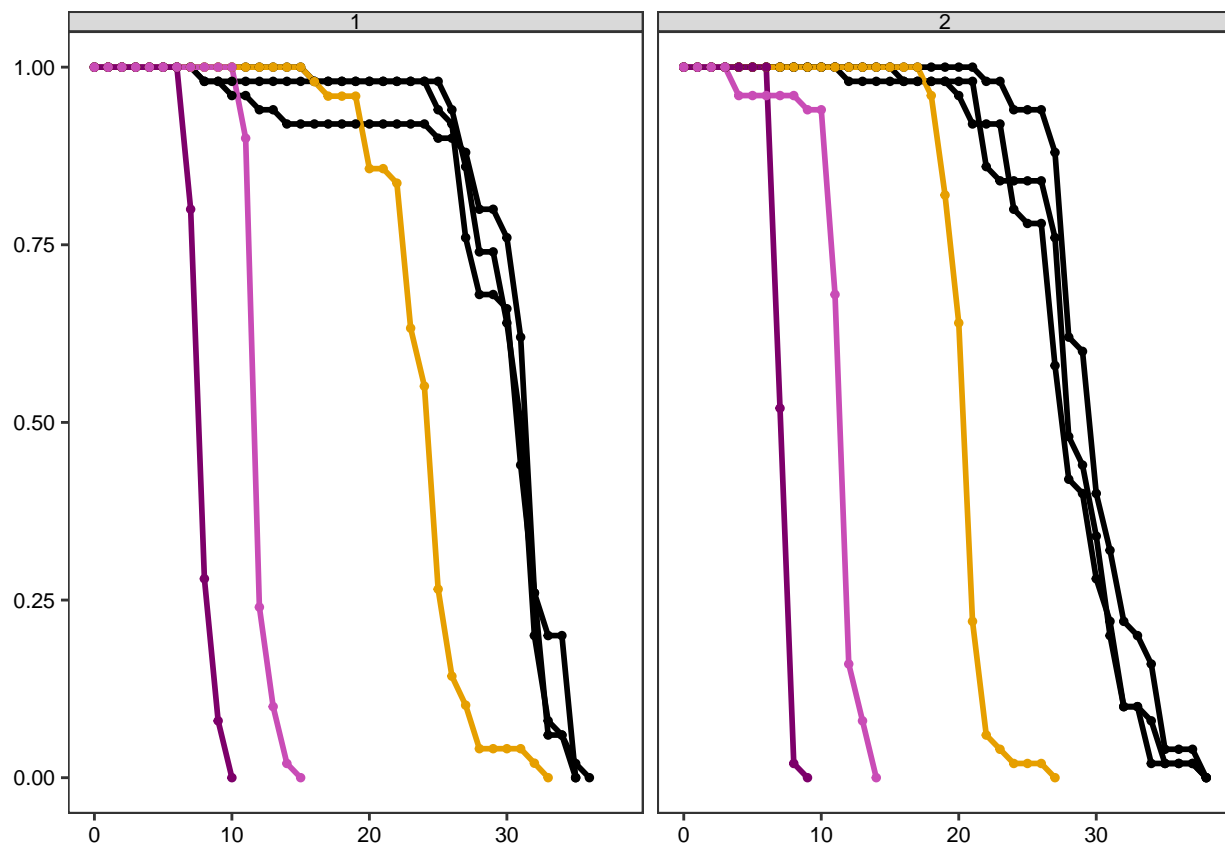
```

# make new variable for type of reciprocal cross
results_lifespan_bidirectional_dataset$matings_cross <- dplyr::recode(as.character(results_lifespan_bidirectional_dataset$matings_cross),
  "CSB X OCTO" = "OCTO",
  "OCTO X CSB" = "OCTO",
  "CSB X POP2L" = "POP2L",
  "POP2L X CSB" = "POP2L",
  "CSB X POP2H" = "POP2H",
  "POP2H X CSB" = "POP2H"
)

# releavel factor
results_lifespan_bidirectional_dataset$matings_cross <- factor(results_lifespan_bidirectional_dataset$matings_cross)

### plot
ggplot(data=results_lifespan_bidirectional_dataset, aes(x=time, y=surv, color=matings_cross)) +
  geom_line(size=1) + geom_point(size=1) +
  facet_grid(.~experimental_replicate) + theme_figs() +
  theme(legend.position="none") +
  scale_color_manual(values=c("#000000", "#000000", "#000000", "#e69f00", "#c94cb6", "#7d006a"))

```



```

### analysis
model_lifespan_bidirectional_cross=coxme::coxme(Surv(day, status) ~ cross + (1|experimental_replicate))

# signifcnce of the model
car::Anova(model_lifespan_bidirectional_cross)

```

```

## Analysis of Deviance Table (Type II tests)
##
## Response: Surv(day, status)
##      Df  Chisq Pr(>Chisq)
## cross  5 404.32 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_lifespan_bidirectional_cross)

## Cox mixed-effects model fit by maximum likelihood
##   Data: s22data
##   events, n = 599, 600
##   Iterations= 9 94
##               NULL Integrated      Fitted
## Log-likelihood -3236.039 -2650.774 -2613.554
##
##               Chisq    df p      AIC      BIC
## Integrated loglik 1170.53  7.00 0 1156.53 1125.76
## Penalized loglik 1244.97 33.08 0 1178.81 1033.42
##
## Model: Surv(day, status) ~ cross + (1 | experimental_replicate) + (1 | unique_id)
## Fixed coefficients
##               coef      exp(coef) se(coef)      z      p
## crossCSB X POP2H -0.08816543 9.156094e-01 0.2056718 -0.43 0.67
## crossCSB X POP2L -0.23146258 7.933724e-01 0.2060225 -1.12 0.26
## crossOCTO X CSB   2.58288162 1.323522e+01 0.2362758 10.93 0.00
## crossPOP2H X CSB 11.88099057 1.444936e+05 0.6923179 17.16 0.00
## crossPOP2L X CSB  7.27973324 1.450601e+03 0.5139492 14.16 0.00
##
## Random effects
## Group              Variable Std Dev  Variance
## experimental_replicate Intercept 0.4265329 0.1819303
## unique_id              Intercept 0.3309596 0.1095343

## multiple comparison
(multiple_lifespan_bidirectional_cross=emmeans::emmeans(model_lifespan_bidirectional_cross, list(pairwi

## $`emmeans of cross`
## cross      emmean    SE df asymp.LCL asymp.UCL
## CSB X OCTO  -3.571 0.227 Inf    -4.17    -2.972
## CSB X POP2H -3.659 0.227 Inf    -4.26    -3.060
## CSB X POP2L -3.802 0.228 Inf    -4.40    -3.200
## OCTO X CSB  -0.988 0.215 Inf    -1.55    -0.421
## POP2H X CSB   8.310 0.509 Inf     6.97     9.652
## POP2L X CSB   3.709 0.340 Inf     2.81     4.606
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 6 estimates
##
## $`pairwise differences of cross`
## contrast              estimate      SE df z.ratio p.value

```

```
## CSB X OCTO - CSB X POP2H    0.0882 0.206 Inf    0.429 0.9711
## CSB X OCTO - CSB X POP2L    0.2315 0.206 Inf    1.123 0.7837
## CSB X OCTO - OCTO X CSB     -2.5829 0.236 Inf   -10.932 <.0001
## CSB X OCTO - POP2H X CSB    -11.8810 0.692 Inf   -17.161 <.0001
## CSB X OCTO - POP2L X CSB     -7.2797 0.514 Inf   -14.164 <.0001
## CSB X POP2H - CSB X POP2L    0.1433 0.205 Inf    0.697 0.9711
## CSB X POP2H - OCTO X CSB     -2.6710 0.237 Inf   -11.274 <.0001
## CSB X POP2H - POP2H X CSB   -11.9692 0.693 Inf   -17.283 <.0001
## CSB X POP2H - POP2L X CSB    -7.3679 0.514 Inf   -14.325 <.0001
## CSB X POP2L - OCTO X CSB     -2.8143 0.239 Inf   -11.799 <.0001
## CSB X POP2L - POP2H X CSB   -12.1125 0.693 Inf   -17.468 <.0001
## CSB X POP2L - POP2L X CSB    -7.5112 0.515 Inf   -14.577 <.0001
## OCTO X CSB - POP2H X CSB     -9.2981 0.670 Inf   -13.879 <.0001
## OCTO X CSB - POP2L X CSB    -4.6969 0.484 Inf    -9.697 <.0001
## POP2H X CSB - POP2L X CSB    4.6013 0.489 Inf    9.401 <.0001
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 15 tests
```

```
#
multcomp::cld(multiple_lifespan_bidirectional_cross$`emmeans of cross`, adj="holm")
```

```
## cross      emmean    SE  df asymp.LCL asymp.UCL .group
## CSB X POP2L -3.802 0.228 Inf    -4.40    -3.200    1
## CSB X POP2H -3.659 0.227 Inf    -4.26    -3.060    1
## CSB X OCTO  -3.571 0.227 Inf    -4.17    -2.972    1
## OCTO X CSB  -0.988 0.215 Inf    -1.55    -0.421    2
## POP2L X CSB  3.709 0.340 Inf     2.81     4.606    3
## POP2H X CSB  8.310 0.509 Inf     6.97     9.652    4
##
```

```
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 6 estimates
## P value adjustment: holm method for 15 tests
## significance level used: alpha = 0.05
```

```
## extracting the coefficients
```

```
coeff_cox_lifespan_bidirectional_cross=group_by(as.data.frame(
  multiple_lifespan_bidirectional_cross$`emmeans of cross`), cross)%>%
  summarize(Coeff=emmean, SE=SE)%>%
  mutate(Coeff=Coeff-Coeff[cross=="CSB X OCTO"])
```

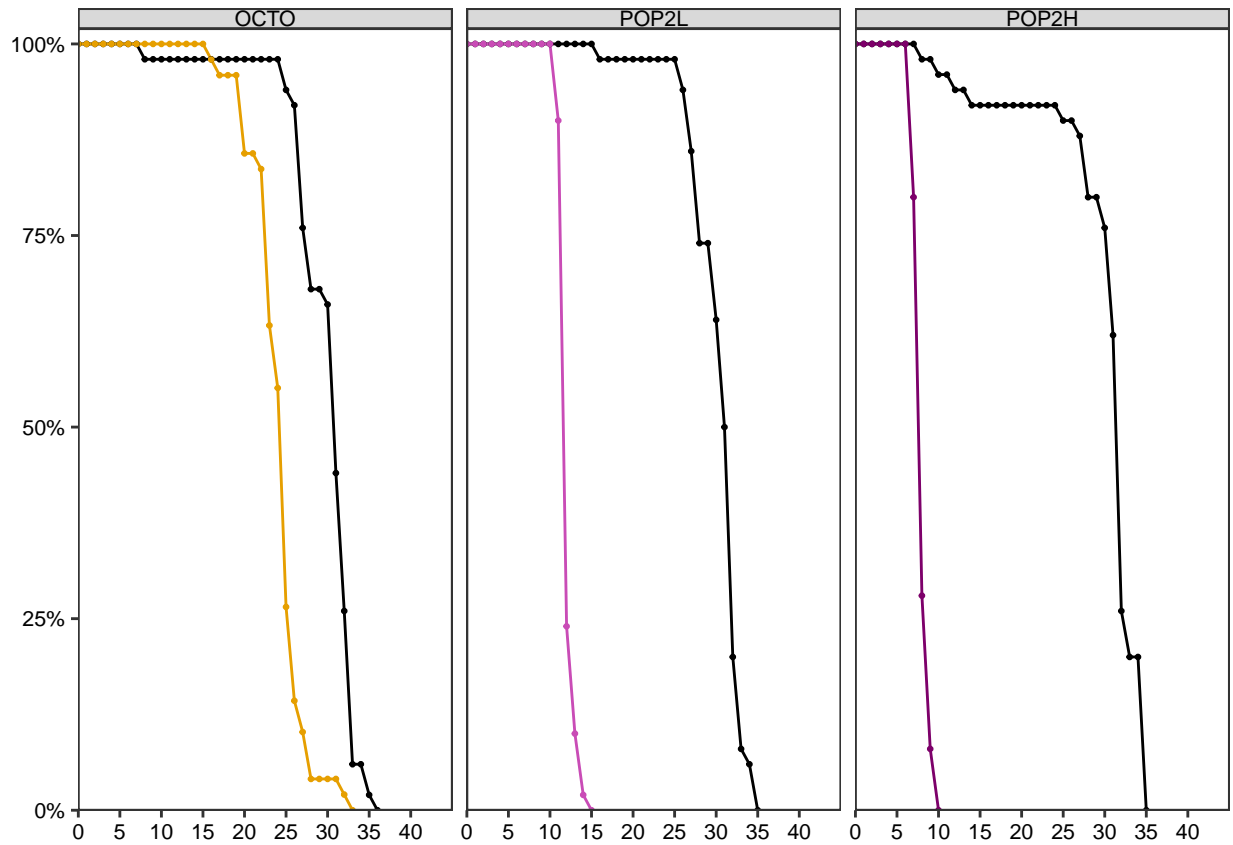
```
# ordering levels
```

```
coeff_cox_lifespan_bidirectional_cross$cross=factor(coeff_cox_lifespan_bidirectional_cross$cross, level=
```

```
### plots
```

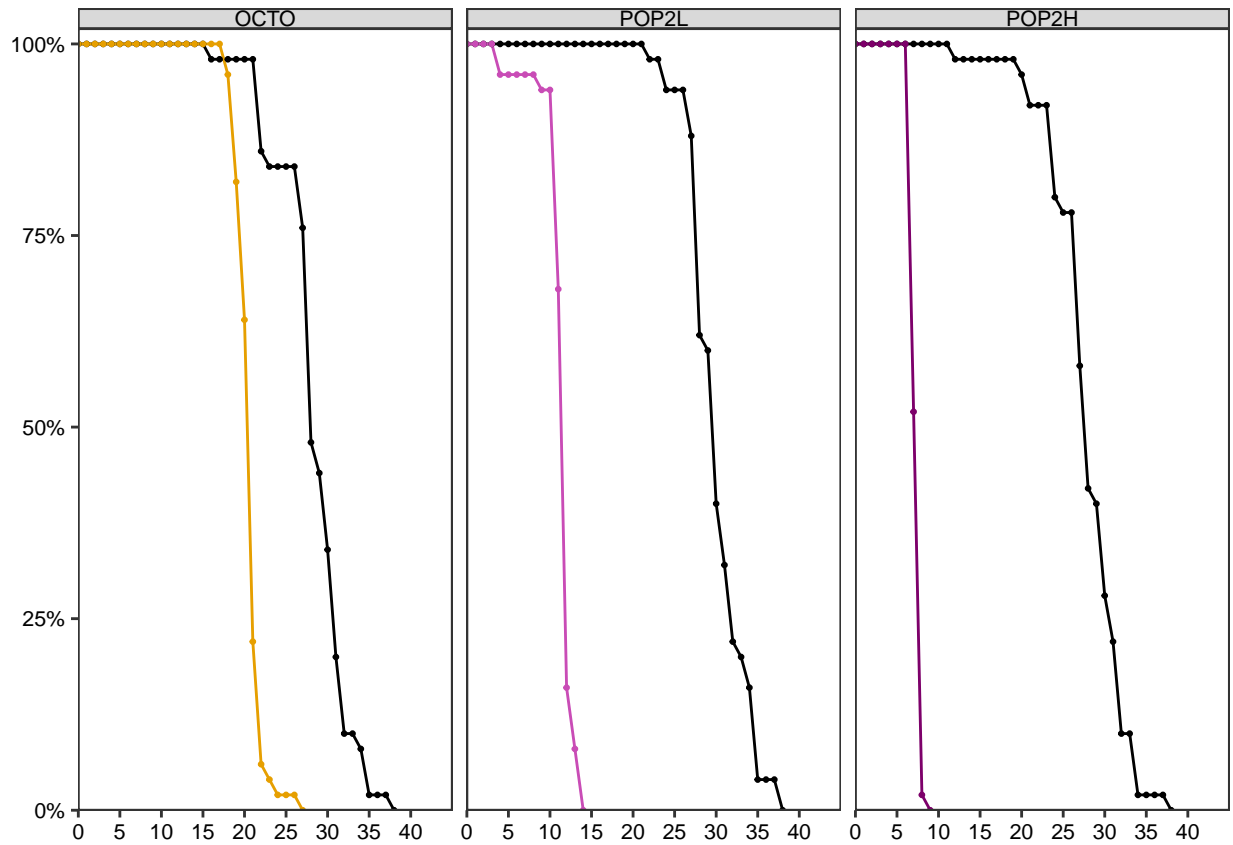
```
## S15A Fig
```

```
(s15afig=ggplot(data=dplyr::filter(results_lifespan_bidirectional_dataset, experimental_replicate=="1")
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#000000", "#000000", "#000000", "#e69f00", "#c94cb6", "#7d006a"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,45), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02), expand=c(0,0), labels=percent)+
  facet_grid(.~mating_cross) +
  theme_figs() +
  theme(legend.position="none"))
```



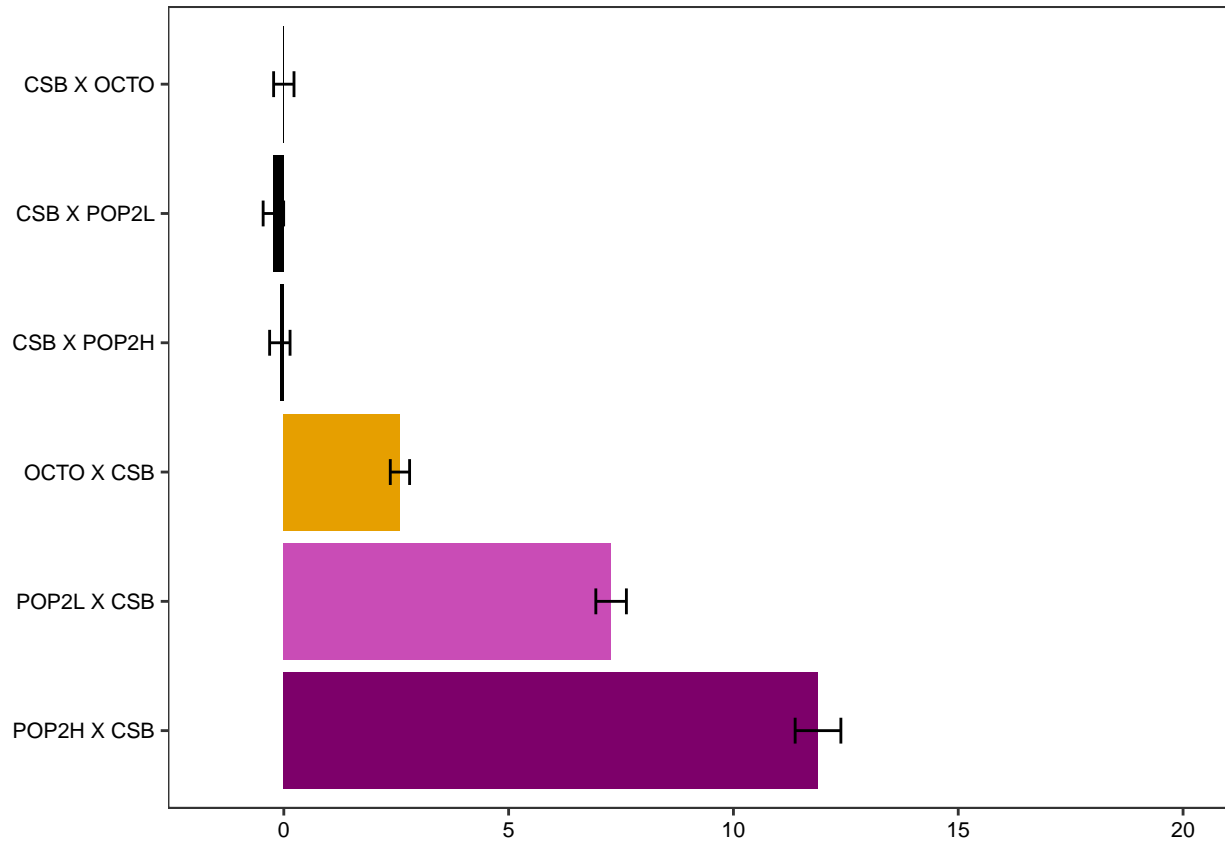
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S15A Fig.pdf", plot=s15afig, device="pdf", dpi=300, units="in", width=420, height=280)
}

## S15 B Fig
(s5bfig=ggplot(data=dplyr::filter(results_lifespan_bidirectional_dataset, experimental_replicate=="2"),
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#000000", "#000000", "#000000", "#e69f00", "#c94cb6", "#7d006a"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,45), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02), expand=c(0,0), labels=percent)+
  facet_grid(~mating_cross) +
  theme_figs() +
  theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S15B Fig.pdf", plot=s15bfig, device="pdf", dpi=300, units="in", width=420, height=280)
}

## the coefficients
(s15cfig=ggplot(coeff_cox_lifespan_bidirectional_cross, aes(x=reorder(cross, desc(cross)), y=Coeff, fill=fill_color)),
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=Coeff-SE, ymax=Coeff+SE), width=0.2, size=0.5) +
  scale_fill_manual(values=c("#000000", "#000000", "#000000", "#e69f00", "#c94cb6", "#7d006a")) +
  theme(legend.position="none")+ coord_flip()+
  scale_y_continuous(limits=c(-1.5, 20), breaks=seq(0,20,5)))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S15C Fig.pdf", plot=s15cfig, device="pdf", dpi=300, units="in", width=240, height=120)
}
```

**Fig 5E-G | Correlation Wolbahcia life-shortening phenotype VS Wolbachia doubling time or titres at time 0**

```
# correlate CHR coefficients with doubling time
# use doubling times calculate in {r fig_3 :: wolbachia_proliferation_dynamics :: analysis and doubling time}
# use coefficients calculate in {r fig_5 :: effect_wolb_host_lifespan :: analysis} - coeff_cox_model_lifespan_males

### adding new variables
# wolbachia doubling time in adults
doubling_time_all <- doubling_time_all %>%
  unite("id", "temperature", "wolbachia", sep = "_", remove = FALSE)

# Cox coefficients for Wolbachia life-shortening phenotype
cox_lifespan_all <- coeff_cox_model_lifespan_males %>%
  unite("id", "temperature", "wolbachia", sep = "_", remove = FALSE)

# merging Wolbachia doubling time & Cox coefficients
corr_chr_vs_proliferation=merge(doubling_time_all, cox_lifespan_all, by="id")
```



```

### analysis
(results_correlation_chr_vs_proliferation=corr_chr_vs_proliferation%>%
  dplyr::group_by(temperature.x)%>%
  dplyr::summarise(Estimate=cor.test(doubling_time, normalized_emmean)$estimate,
    R2=paste0(round((cor.test(doubling_time, normalized_emmean)$estimate)**2,2)*100,"%"),
    P.value=cor.test(doubling_time, normalized_emmean)$p.value))

## # A tibble: 3 x 4
##   temperature.x Estimate R2      P.value
## * <fct>          <dbl> <chr>   <dbl>
## 1 18              -0.943 89%    0.00485
## 2 25              -0.863 75%    0.0267
## 3 29              -0.924 85%    0.00840

# correlate CHR coefficients with time 0 levels
# use coefficients above
# use time 0 titres from {r s11_fig :: wolbachia_titer_day_of_eclosion :: analysis} - time_0_levels

time_0_levels <- time_0_levels %>%
  mutate(Titers = emmean, SE.Titer=SE)%>%
  mutate(Titers=Titers-Titers[wolbachia=="wMelCS_b"])

# merging titers day of eclosion & Cox coefficients
corr_chr_vs_day0=merge(cox_lifespan_all, time_0_levels, by="wolbachia")

### analysis
(results_correlation_chr_vs_day0 = corr_chr_vs_day0 %>%
  dplyr::group_by(temperature)%>%
  dplyr::summarise(Estimate=cor.test(Titers, normalized_emmean)$estimate,
    R2=paste0(round((cor.test(Titers, normalized_emmean)$estimate)**2,2)*100,"%"),
    P.value=cor.test(Titers, normalized_emmean)$p.value))

## # A tibble: 3 x 4
##   temperature Estimate R2      P.value
## * <fct>          <dbl> <chr>   <dbl>
## 1 18              0.804 65%    0.0541
## 2 25              0.480 23%    0.336
## 3 29              0.448 20%    0.373

```

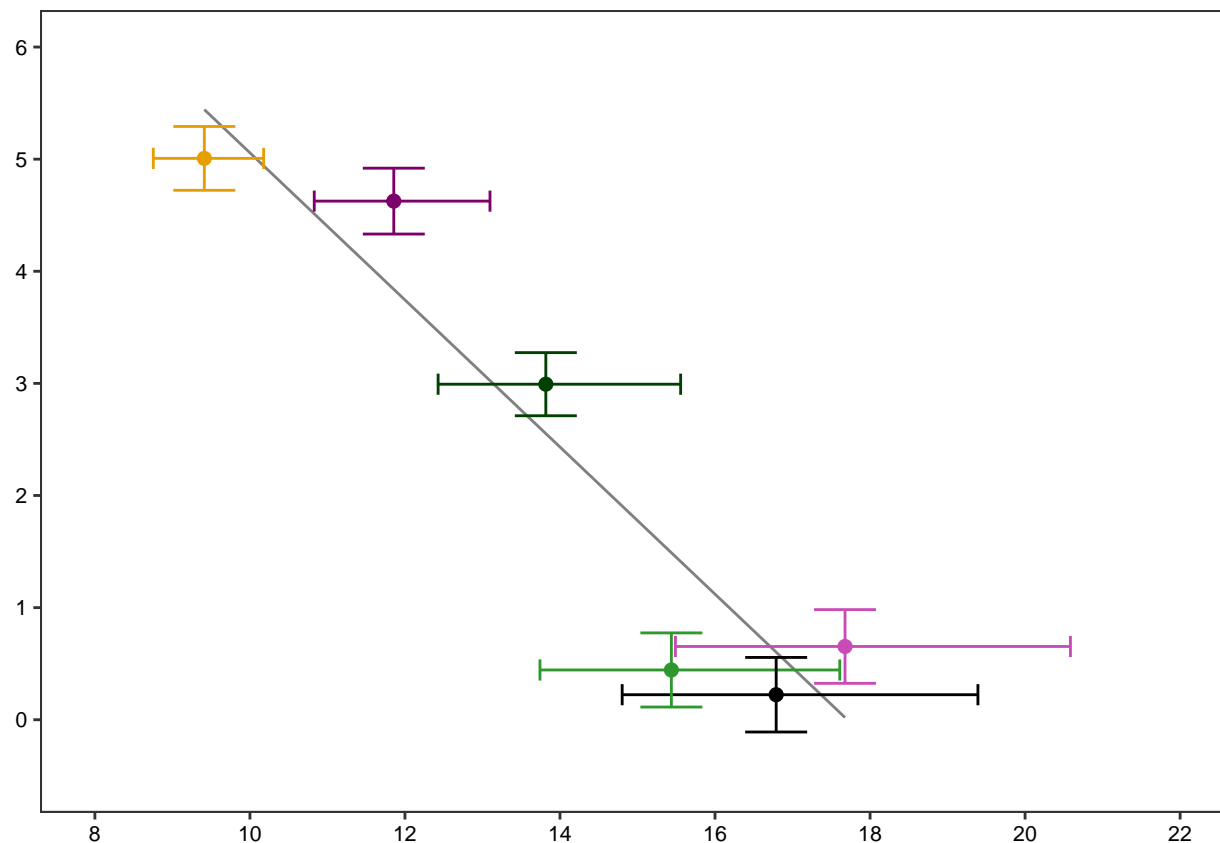
Fig 5E-G panels

```

### plot
# Fig 5E
(fig5e=ggplot(dplyr::filter(corr_chr_vs_proliferation, temperature.x=="18"),
  aes(x=doubling_time, y=normalized_emmean, color=wolbachia.x))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=upper, xmin=lower))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.y, ymin=normalized_emmean-SE.y))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-0.5, 6), breaks=seq(0, 6, 1))+
  scale_x_continuous(limits=c(8,22), breaks=seq(8,22,2)))

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

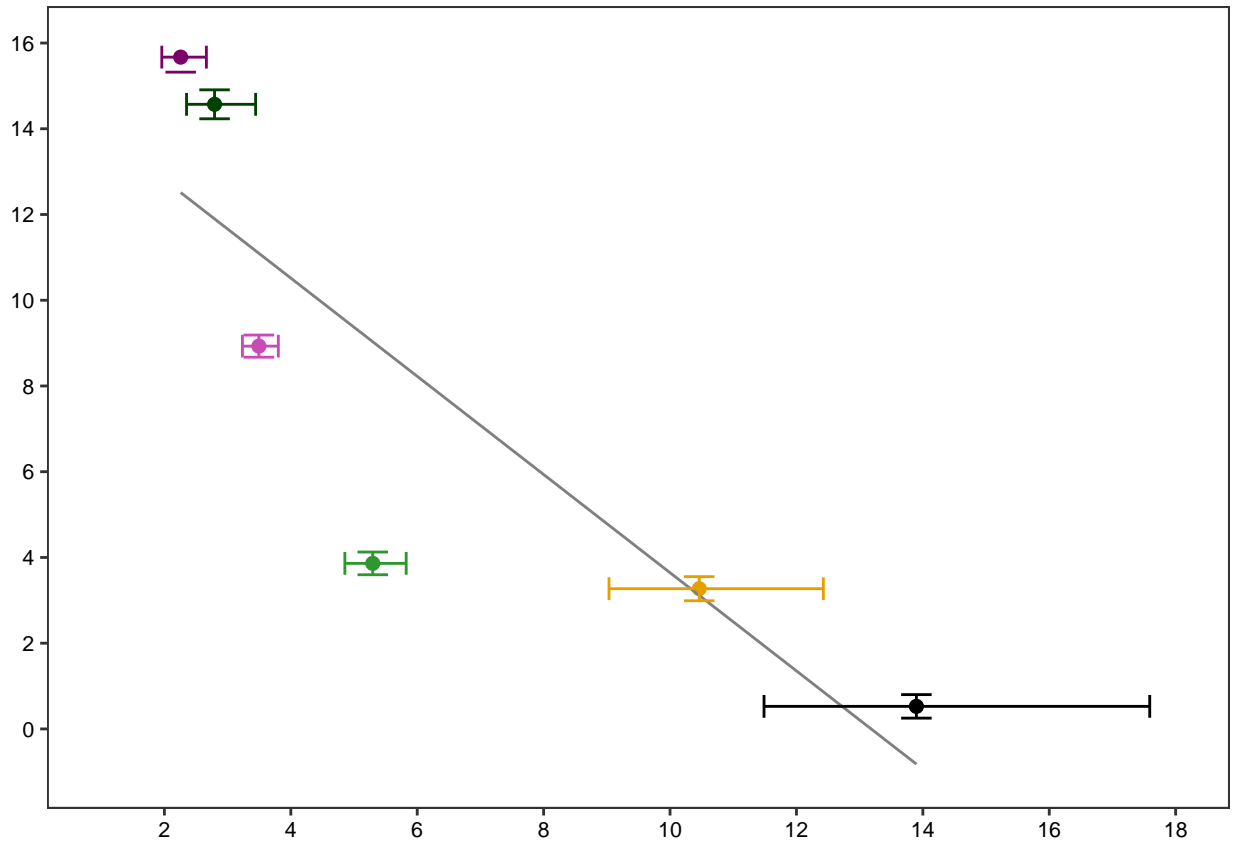
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5E.pdf", plot=fig5e, device="pdf", dpi=300, units="in", width=200/96,
}

# Fig 5F
(fig5f=ggplot(dplyr::filter(corr_chr_vs_proliferation, temperature.x=="25"),
  aes(x=doubling_time, y=normalized_emmean, color=wolbachia.x))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=upper, xmin=lower))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.y, ymin=normalized_emmean-SE.y))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-1, 16), breaks=seq(0, 16, 2))+
  scale_x_continuous(limits=c(1, 18), breaks=seq(2,18,2)))

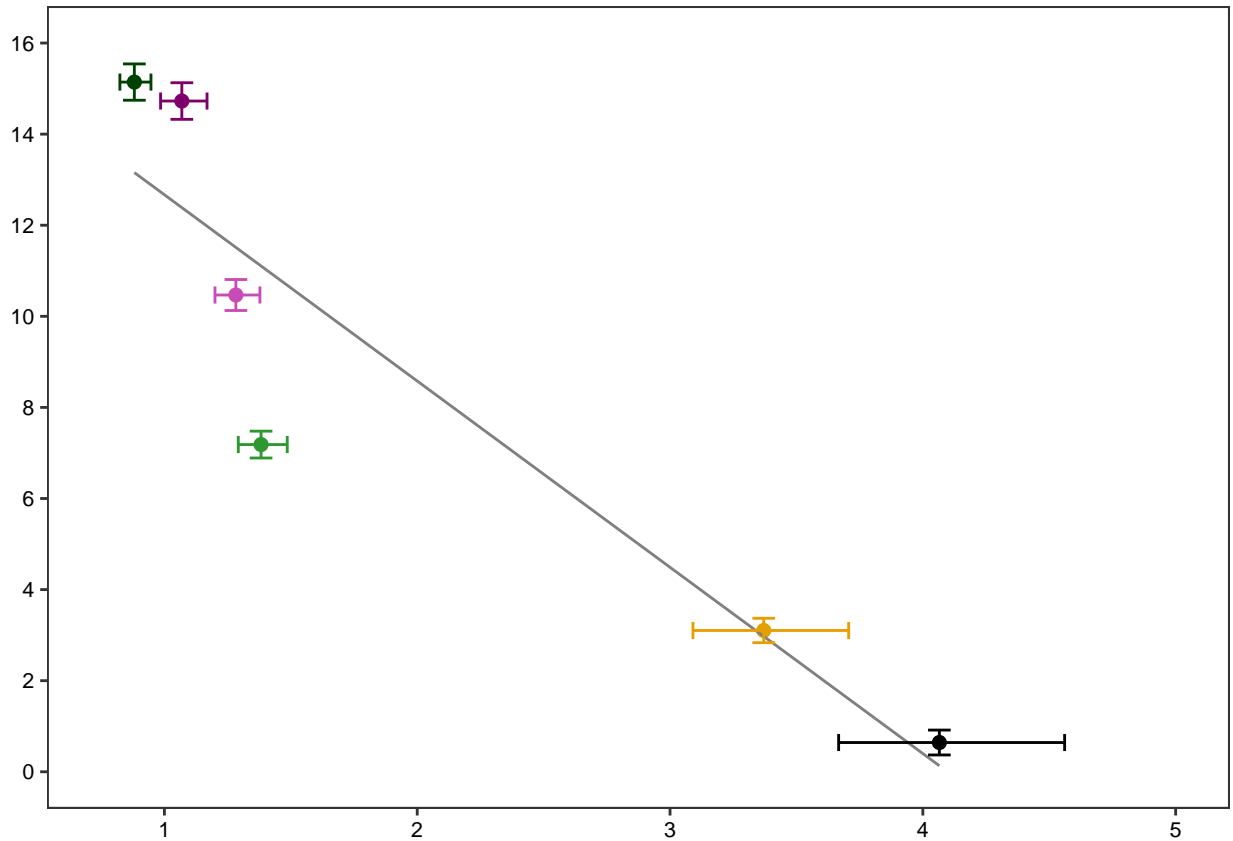
## `geom_smooth()` using formula 'y ~ x'
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5F.pdf", plot=fig5f, device="pdf", dpi=300, units="in", width=200/96,
}

# Fig 5G
(fig5g=ggplot(dplyr::filter(corr_chr_vs_proliferation, temperature.x=="29"),
  aes(x=doubling_time, y=normalized_emmean, color=wolbachia.x))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=upper, xmin=lower))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.y, ymin=normalized_emmean-SE.y))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(0,16), breaks=seq(0, 16, 2))+
  scale_x_continuous(limits=c(0.75,5), breaks=seq(1,5,1)))

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

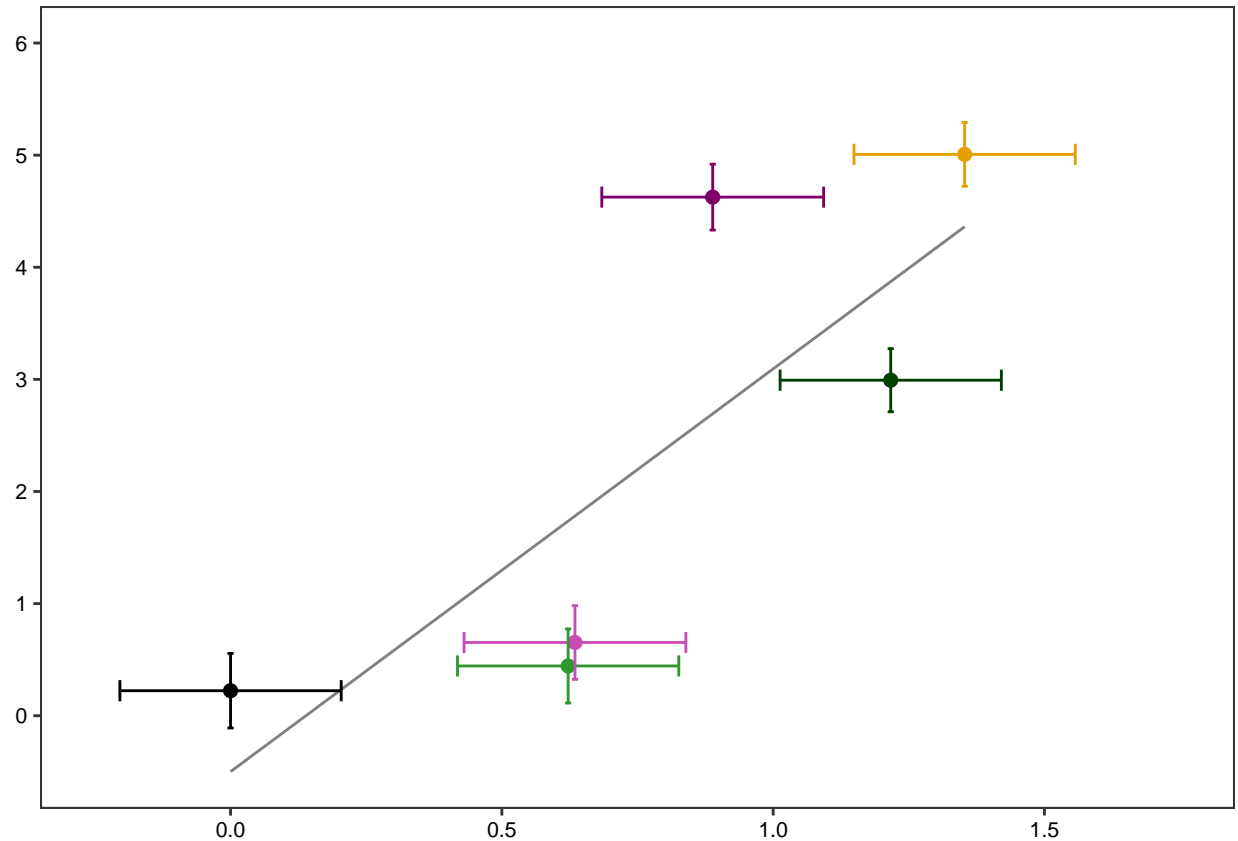


```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 5G.pdf", plot=fig5g, device="pdf", dpi=300, units="in", width=200/96,
}
}
```

## S16 Fig panels | Correlation Wolbachia life-shortening phenotype VS titers

```
### S16A Fig
(figs16a=ggplot(dplyr::filter(corr_chr_vs_day0, temperature=="18"),
  aes(x=Titers, y=normalized_emmean, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=Titers+SE.Titer, xmin=Titers-SE.Titer))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.x, ymin=normalized_emmean-SE.x))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-0.5, 6), breaks=seq(0, 6, 1))+
  scale_x_continuous(limits=c(-0.25,1.75), breaks=seq(0,1.5,0.5)))

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



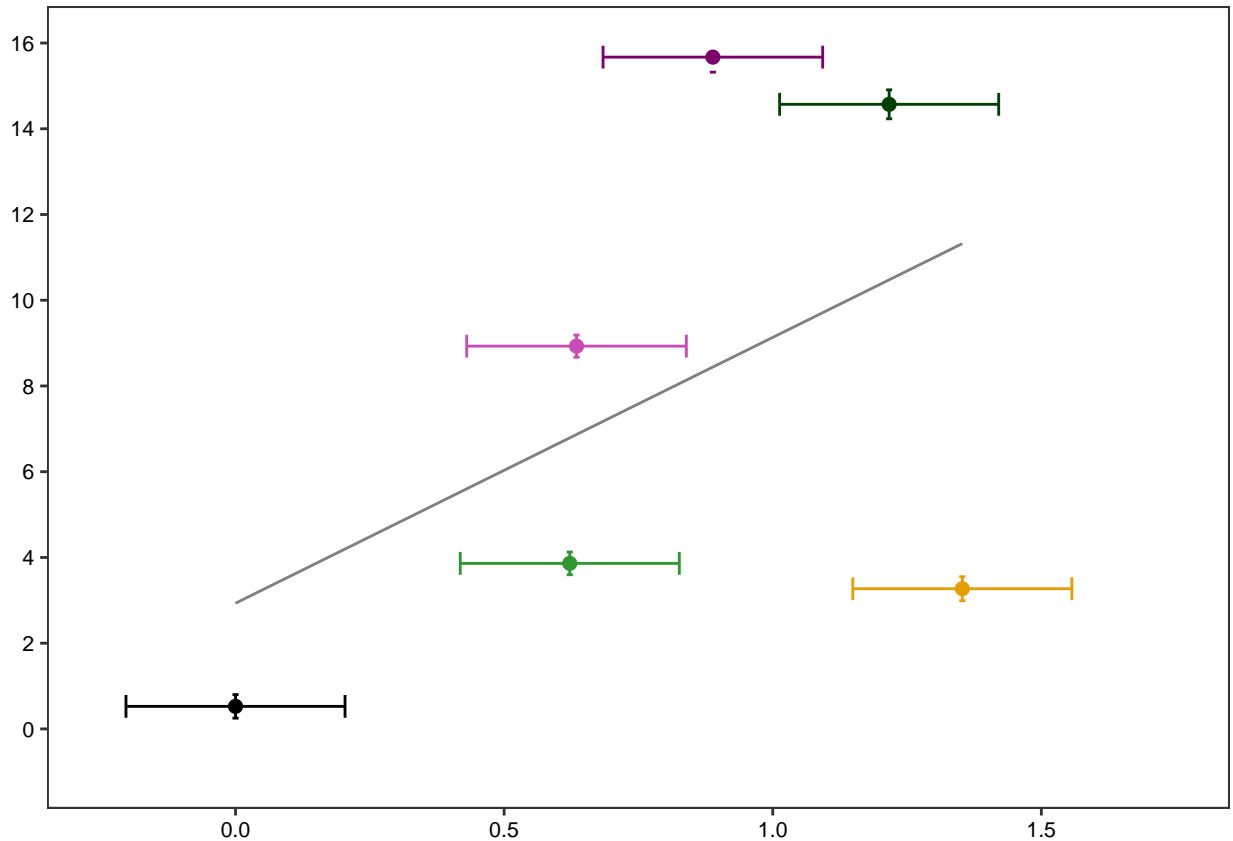
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16A Fig.pdf", plot=figs16a, device="pdf", dpi=300, units="in", width=200,
}

```

```
### S16B Fig
(figs16b=ggplot(dplyr::filter(corr_chr_vs_day0, temperature=="25"),
  aes(x=Titters, y=normalized_emmean, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=Titters+SE.Titer, xmin=Titters-SE.Titer))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.x, ymin=normalized_emmean-SE.x))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-1, 16), breaks=seq(0, 16, 2))+
  scale_x_continuous(limits=c(-0.25,1.75), breaks=seq(0,1.5,0.5)))

```

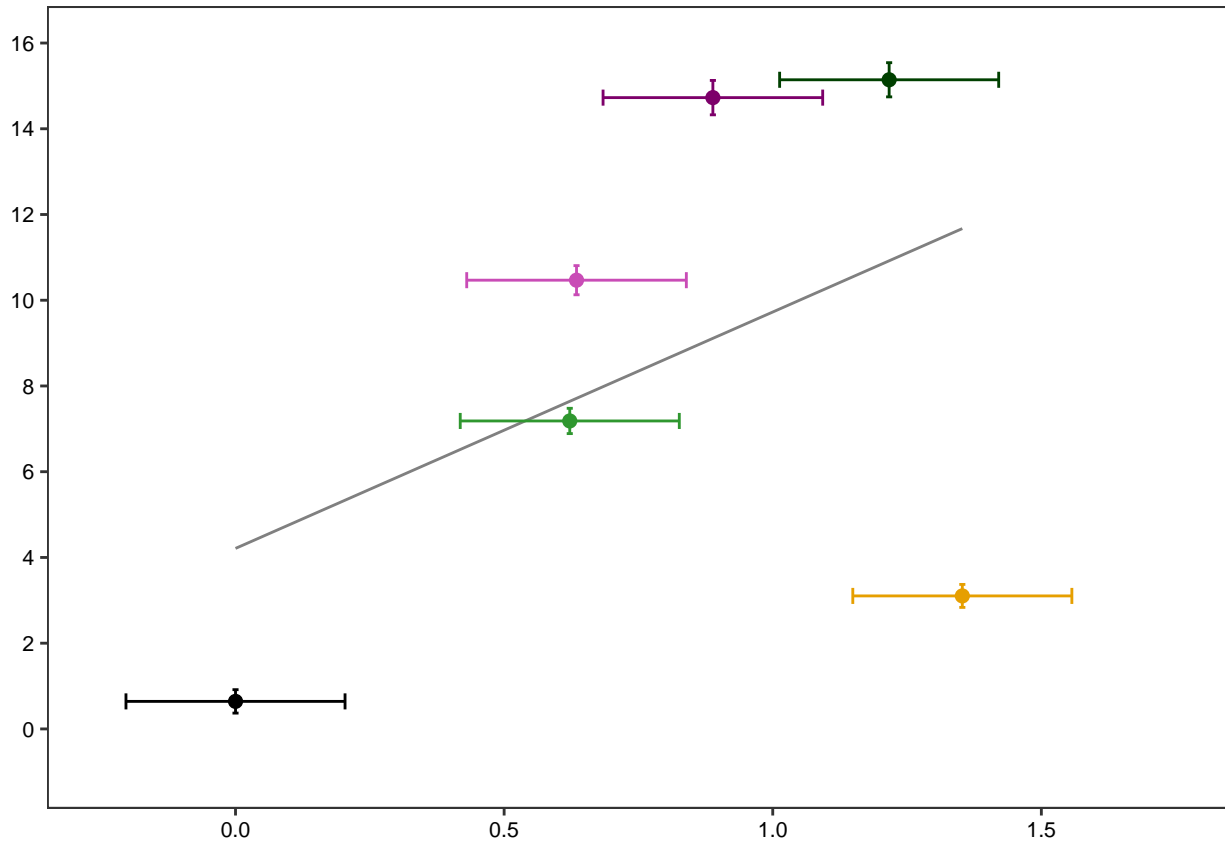
```
## `geom_smooth()` using formula 'y ~ x'
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16B Fig.pdf", plot=figs16b, device="pdf", dpi=300, units="in", width=200, height=200)
}

### S16C Fig
(figs16c=ggplot(dplyr::filter(corr_chr_vs_day0, temperature=="29"),
  aes(x=Titers, y=normalized_emmean, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=Titers+SE.Titer, xmin=Titers-SE.Titer))+
  geom_errorbar(aes(ymax=normalized_emmean+SE.x, ymin=normalized_emmean-SE.x))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-1, 16), breaks=seq(0, 16, 2))+
  scale_x_continuous(limits=c(-0.25,1.75), breaks=seq(0,1.5,0.5)))

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



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16C Fig.pdf", plot=figs16c, device="pdf", dpi=300, units="in", width=200, height=200)
}
```

**Fig 6 | Wolbachia anti-viral protection**

**Fig 6 | Dataset**

```
### S23 Data
s23data=read.csv("../original_data/S23 Data.csv", stringsAsFactors = TRUE)

## choosing Wolbachia-free as reference
s23data$wolbachia=factor(s23data$wolbachia, levels=c("NoWolb", "CSB", "OCT0", "POP2L", "POP2H", "POPL", "POPL2"))

### extracting Kaplan estimates (+/- confidence intervals) from the data
results_survival_dcv=survfit(Surv(day,status)~wolbachia+infection+experimental_replicate, data=s23data, method="kaplan")

## converting survival model to a dataframe
results_survival_dcv_dataset=as.data.frame(summary(results_survival_dcv, times=c(0:50)))[c("surv", "upper", "lower")]

## adding new variables to the dataframe
results_survival_dcv_dataset=results_survival_dcv_dataset%>%
  tidyr::separate(strata, c("wolbachia", "infection", "experimental_replicate"), ", ", remove=FALSE)%>%
```

```

dplyr::mutate(wolbachia=str_replace(wolbachia, "wolbachia=", ""),
             infection=str_replace(infection, "infection=", ""),
             infection=str_replace(infection, " ", ""),
             experimental_replicate=str_replace(experimental_replicate, "experimental_replicate=", "")

## converting variables to factor
results_survival_dcv_dataset$wolbachia=factor(results_survival_dcv_dataset$wolbachia, levels=c("NoWolb"

### plot
ggplot(data=results_survival_dcv_dataset, aes(x=time, y=surv, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+
  facet_grid(experimental_replicate~infection, scales="free")+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  ylab("Survival") + xlab("Days post eclosion")+
  scale_x_continuous(expand=c(0,0))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)

```

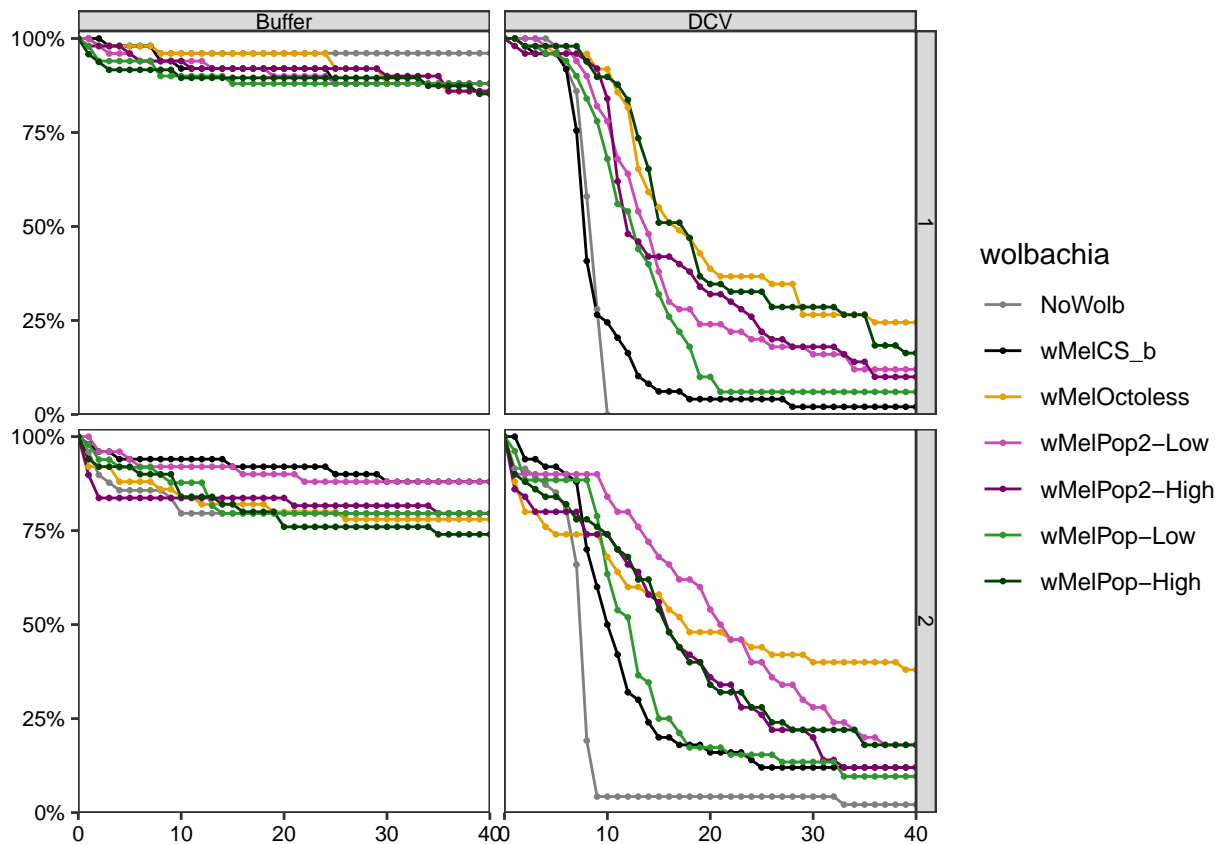


Fig 6 | Statistical analysis

```

### analysis
## dcv
model_cox_survival_dcv=coxme::coxme(Surv(day, status) ~ wolbachia + (1|unique_id) + (1|replicate), data=

```



```

# significance of the model
car::Anova(model_cox_survival_dcv)

## Analysis of Deviance Table (Type II tests)
##
## Response: Surv(day, status)
##           Df  Chisq Pr(>Chisq)
## wolbachia  6 148.22 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# summary
summary(model_cox_survival_dcv)

## Cox mixed-effects model fit by maximum likelihood
## Data: dplyr::filter(s23data, infection == "DCV")
## events, n = 606, 696
## Iterations= 32 197
##           NULL Integrated    Fitted
## Log-likelihood -3545.602 -3442.264 -3405.944
##
##           Chisq    df p    AIC    BIC
## Integrated loglik 206.68  8.00 0 190.68 155.42
## Penalized loglik 279.32 34.42 0 210.47  58.78
##
## Model: Surv(day, status) ~ wolbachia + (1 | unique_id) + (1 | replicate)
## Fixed coefficients
##           coef exp(coef) se(coef)      z      p
## wolbachiaCSB -0.8364751 0.4332350 0.2047625 -4.09 4.4e-05
## wolbachiaOCTO -2.1802577 0.1130124 0.2179750 -10.00 0.0e+00
## wolbachiaPOP2L -1.8710393 0.1539636 0.2100666 -8.91 0.0e+00
## wolbachiaPOP2H -1.6980347 0.1830429 0.2083992 -8.15 3.3e-16
## wolbachiaPOPL -1.3363392 0.2628060 0.2055577 -6.50 8.0e-11
## wolbachiaPOPH -1.9013025 0.1493739 0.2112228 -9.00 0.0e+00
##
## Random effects
## Group      Variable Std Dev    Variance
## unique_id Intercept 0.305459384 0.093305435
## replicate Intercept 0.064413063 0.004149043

# multiple comparison
(multiple_cox_survival_dcv=emmeans::emmeans(model_cox_survival_dcv, list(pairwise~wolbachia), adj="holm

## $`emmeans of wolbachia`
## wolbachia emmean SE df asymp.LCL asymp.UCL
## NoWolb 1.4082 0.139 Inf 1.035 1.7812
## CSB 0.5717 0.134 Inf 0.211 0.9320
## OCTO -0.7721 0.144 Inf -1.159 -0.3855
## POP2L -0.4629 0.136 Inf -0.827 -0.0983
## POP2H -0.2898 0.134 Inf -0.650 0.0707
## POPL 0.0718 0.132 Inf -0.284 0.4275
## POPH -0.4931 0.137 Inf -0.861 -0.1249
##
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95

```

```
## Conf-level adjustment: bonferroni method for 7 estimates
```

```
##
```

```
## $`pairwise differences of wolbachia`
```

contrast	estimate	SE	df	z.ratio	p.value
NoWolb - CSB	0.8365	0.205	Inf	4.085	0.0005
NoWolb - OCTO	2.1803	0.218	Inf	10.002	<.0001
NoWolb - POP2L	1.8710	0.210	Inf	8.907	<.0001
NoWolb - POP2H	1.6980	0.208	Inf	8.148	<.0001
NoWolb - POPL	1.3363	0.206	Inf	6.501	<.0001
NoWolb - POPH	1.9013	0.211	Inf	9.001	<.0001
CSB - OCTO	1.3438	0.214	Inf	6.293	<.0001
CSB - POP2L	1.0346	0.206	Inf	5.023	<.0001
CSB - POP2H	0.8616	0.204	Inf	4.213	0.0003
CSB - POPL	0.4999	0.203	Inf	2.466	0.1094
CSB - POPH	1.0648	0.207	Inf	5.144	<.0001
OCTO - POP2L	-0.3092	0.213	Inf	-1.449	0.7371
OCTO - POP2H	-0.4822	0.212	Inf	-2.271	0.1622
OCTO - POPL	-0.8439	0.212	Inf	-3.980	0.0008
OCTO - POPH	-0.2790	0.214	Inf	-1.301	0.7731
POP2L - POP2H	-0.1730	0.205	Inf	-0.843	0.9735
POP2L - POPL	-0.5347	0.205	Inf	-2.613	0.0808
POP2L - POPH	0.0303	0.207	Inf	0.146	0.9735
POP2H - POPL	-0.3617	0.203	Inf	-1.780	0.4509
POP2H - POPH	0.2033	0.206	Inf	0.985	0.9735
POPL - POPH	0.5650	0.206	Inf	2.748	0.0600

```
##
```

```
## Results are given on the log (not the response) scale.
```

```
## P value adjustment: holm method for 21 tests
```

```
#
```

```
multcomp::cld(multiple_cox_survival_dcv$`emmeans of wolbachia`)
```

wolbachia	emmean	SE	df	asympt.LCL	asympt.UCL	.group
OCTO	-0.7721	0.144	Inf	-1.159	-0.3855	1
POPH	-0.4931	0.137	Inf	-0.861	-0.1249	12
POP2L	-0.4629	0.136	Inf	-0.827	-0.0983	12
POP2H	-0.2898	0.134	Inf	-0.650	0.0707	12
POPL	0.0718	0.132	Inf	-0.284	0.4275	23
CSB	0.5717	0.134	Inf	0.211	0.9320	3
NoWolb	1.4082	0.139	Inf	1.035	1.7812	4

```
##
```

```
## Results are given on the log (not the response) scale.
```

```
## Confidence level used: 0.95
```

```
## Conf-level adjustment: bonferroni method for 7 estimates
```

```
## P value adjustment: tukey method for comparing a family of 7 estimates
```

```
## significance level used: alpha = 0.05
```

```
# extracting the coefficients
```

```
coeff_cox_survival_dcv=as.data.frame(multiple_cox_survival_dcv$`emmeans of wolbachia`)%>%
  group_by(wolbachia)%>%
```

```
  summarise(DCV.Coeff=emmean, DCV.Coeff.SE=SE)%>%
```

```
  mutate(DCV.Coeff=DCV.Coeff-DCV.Coeff[wolbachia=="NoWolb"],
```

```
         wolbachia=factor(wolbachia, levels=c("NoWolb", "CSB", "OCTO", "POP2L", "POP2H", "POPL", "POPH"))
```

```

## buffer
model_cox_survival_buffer=coxme::coxme(Surv(day, status) ~ wolbachia + (1|experimental_replicate) + (1|

# significance of the model
car::Anova(model_cox_survival_buffer)

## Analysis of Deviance Table (Type II tests)
##
## Response: Surv(day, status)
##           Df  Chisq Pr(>Chisq)
## wolbachia  6 5.1197    0.5286

# summary
summary(model_cox_survival_buffer)

## Cox mixed-effects model fit by maximum likelihood
##   Data: dplyr::filter(s23data, infection == "Buffer")
##   events, n = 107, 696
##   Iterations= 5 28
##
##              NULL Integrated      Fitted
## Log-likelihood -691.4245  -686.4512 -684.9633
##
##              Chisq  df          p    AIC    BIC
## Integrated loglik  9.95 8.00 0.268790 -6.05 -27.44
## Penalized loglik 12.92 6.91 0.070819 -0.89 -19.35
##
## Model: Surv(day, status) ~ wolbachia + (1 | experimental_replicate) +      (1 | unique_id)
## Fixed coefficients
##
##              coef exp(coef) se(coef)      z      p
## wolbachiaCSB   -0.06563583 0.9364718 0.4084543 -0.16 0.87
## wolbachiaOCTO    0.38339532 1.4672580 0.3728215  1.03 0.30
## wolbachiaPOP2L  -0.05955486 0.9421838 0.4084490 -0.15 0.88
## wolbachiaPOP2H   0.35515478 1.4264014 0.3771853  0.94 0.35
## wolbachiaPOPL    0.27855682 1.3212217 0.3820456  0.73 0.47
## wolbachiaPOPH    0.52455230 1.6897022 0.3653660  1.44 0.15
##
## Random effects
##   Group              Variable Std Dev      Variance
## experimental_replicate Intercept 0.3618572731 0.1309406861
## unique_id              Intercept 0.0197753662 0.0003910651

# multiple comparison
(multiple_cox_survival_buffer=emmeans::emmeans(model_cox_survival_buffer, list(pairwise~wolbachia), adj

## $`emmeans of wolbachia`
##   wolbachia emmean   SE df asymp.LCL asymp.UCL
## NoWolb    -0.2011 0.263 Inf   -0.909    0.507
## CSB        -0.2667 0.263 Inf   -0.974    0.441
## OCTO        0.1823 0.222 Inf   -0.416    0.780
## POP2L      -0.2607 0.263 Inf   -0.968    0.447
## POP2H       0.1541 0.228 Inf   -0.459    0.767
## POPL        0.0775 0.233 Inf   -0.551    0.706
## POPH        0.3235 0.214 Inf   -0.252    0.899
##
## Results are given on the log (not the response) scale.

```

```
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
##
## $`pairwise differences of wolbachia`
## contrast      estimate      SE df z.ratio p.value
## NoWolb - CSB    0.06564 0.408 Inf  0.161  1.0000
## NoWolb - OCTO  -0.38340 0.373 Inf -1.028  1.0000
## NoWolb - POP2L  0.05955 0.408 Inf  0.146  1.0000
## NoWolb - POP2H -0.35515 0.377 Inf -0.942  1.0000
## NoWolb - POPL  -0.27856 0.382 Inf -0.729  1.0000
## NoWolb - POPH  -0.52455 0.365 Inf -1.436  1.0000
## CSB - OCTO     -0.44903 0.373 Inf -1.204  1.0000
## CSB - POP2L    -0.00608 0.408 Inf -0.015  1.0000
## CSB - POP2H    -0.42079 0.377 Inf -1.116  1.0000
## CSB - POPL     -0.34419 0.382 Inf -0.901  1.0000
## CSB - POPH     -0.59019 0.365 Inf -1.616  1.0000
## OCTO - POP2L   0.44295 0.373 Inf  1.188  1.0000
## OCTO - POP2H   0.02824 0.338 Inf  0.083  1.0000
## OCTO - POPL    0.10484 0.344 Inf  0.305  1.0000
## OCTO - POPH   -0.14116 0.325 Inf -0.434  1.0000
## POP2L - POP2H -0.41471 0.377 Inf -1.099  1.0000
## POP2L - POPL  -0.33811 0.382 Inf -0.885  1.0000
## POP2L - POPH  -0.58411 0.365 Inf -1.599  1.0000
## POP2H - POPL   0.07660 0.348 Inf  0.220  1.0000
## POP2H - POPH  -0.16940 0.330 Inf -0.513  1.0000
## POPL - POPH   -0.24600 0.336 Inf -0.733  1.0000
##
## Results are given on the log (not the response) scale.
## P value adjustment: holm method for 21 tests
```

```
#
multcomp::cld(multiple_cox_survival_buffer$`emmeans of wolbachia`)
```

```
## wolbachia emmean      SE df asymp.LCL asymp.UCL .group
## CSB      -0.2667 0.263 Inf  -0.974    0.441  1
## POP2L    -0.2607 0.263 Inf  -0.968    0.447  1
## NoWolb   -0.2011 0.263 Inf  -0.909    0.507  1
## POPL      0.0775 0.233 Inf  -0.551    0.706  1
## POP2H     0.1541 0.228 Inf  -0.459    0.767  1
## OCTO      0.1823 0.222 Inf  -0.416    0.780  1
## POPH      0.3235 0.214 Inf  -0.252    0.899  1
##
```

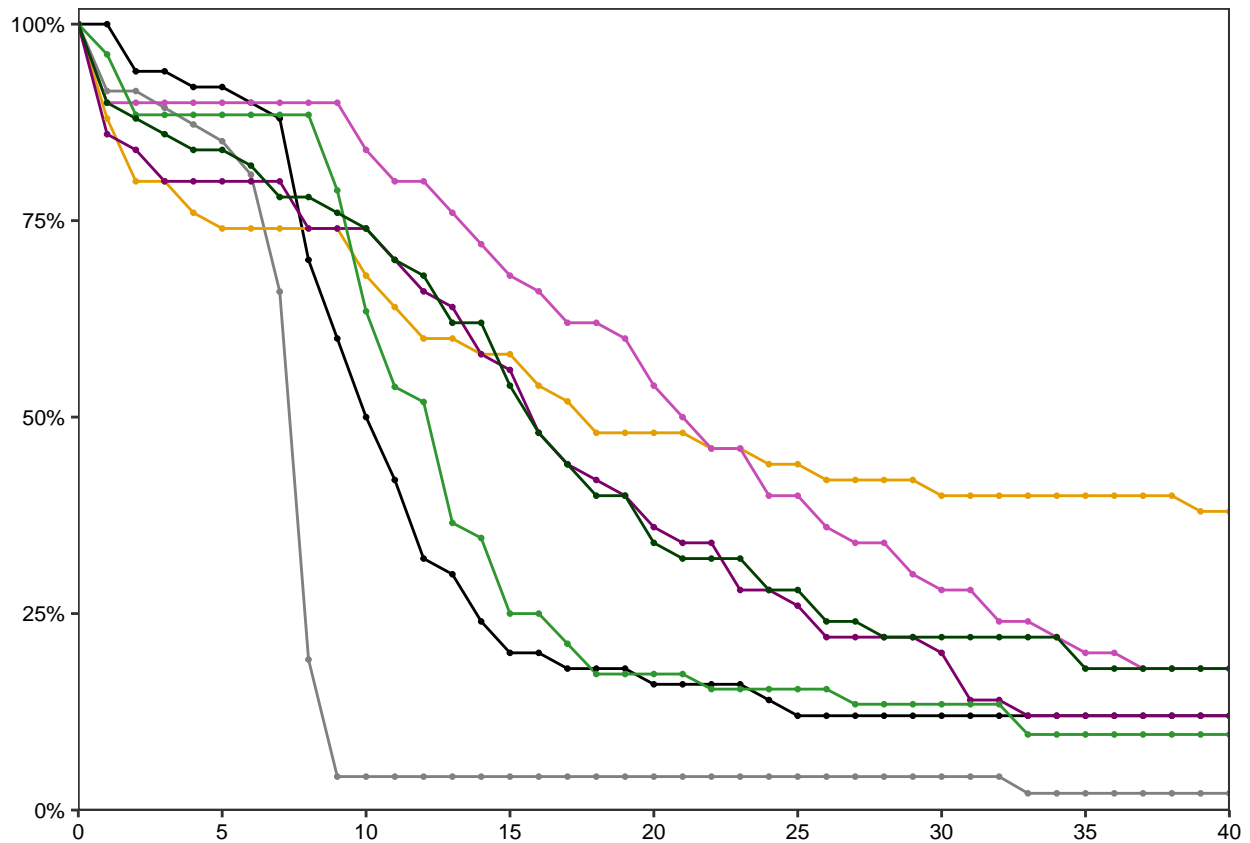
```
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 7 estimates
## P value adjustment: tukey method for comparing a family of 7 estimates
## significance level used: alpha = 0.05
```

```
# extracting the coefficients
```

```
coeff_cox_survival_buffer=as.data.frame(multiple_cox_survival_buffer$`emmeans of wolbachia`)%>%
  group_by(wolbachia)%>%
  summarise(DCV.Coeff=emmean, DCV.Coeff.SE=SE)%>%
  mutate(DCV.Coeff=DCV.Coeff-DCV.Coeff[wolbachia=="NoWolb"],
         wolbachia=factor(wolbachia, levels=c("NoWolb", "CSB", "OCTO", "POP2L", "POP2H", "POPL", "POPH")))
```

Fig 6A-D

```
### plots
## fig6a
(fig6a=ggplot(data=dplyr::filter(results_survival_dcv_dataset,
                                experimental_replicate=="2" & infection=="DCV"),
              aes(x=time, y=surv, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"),
                    linetype=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,40), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02), expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



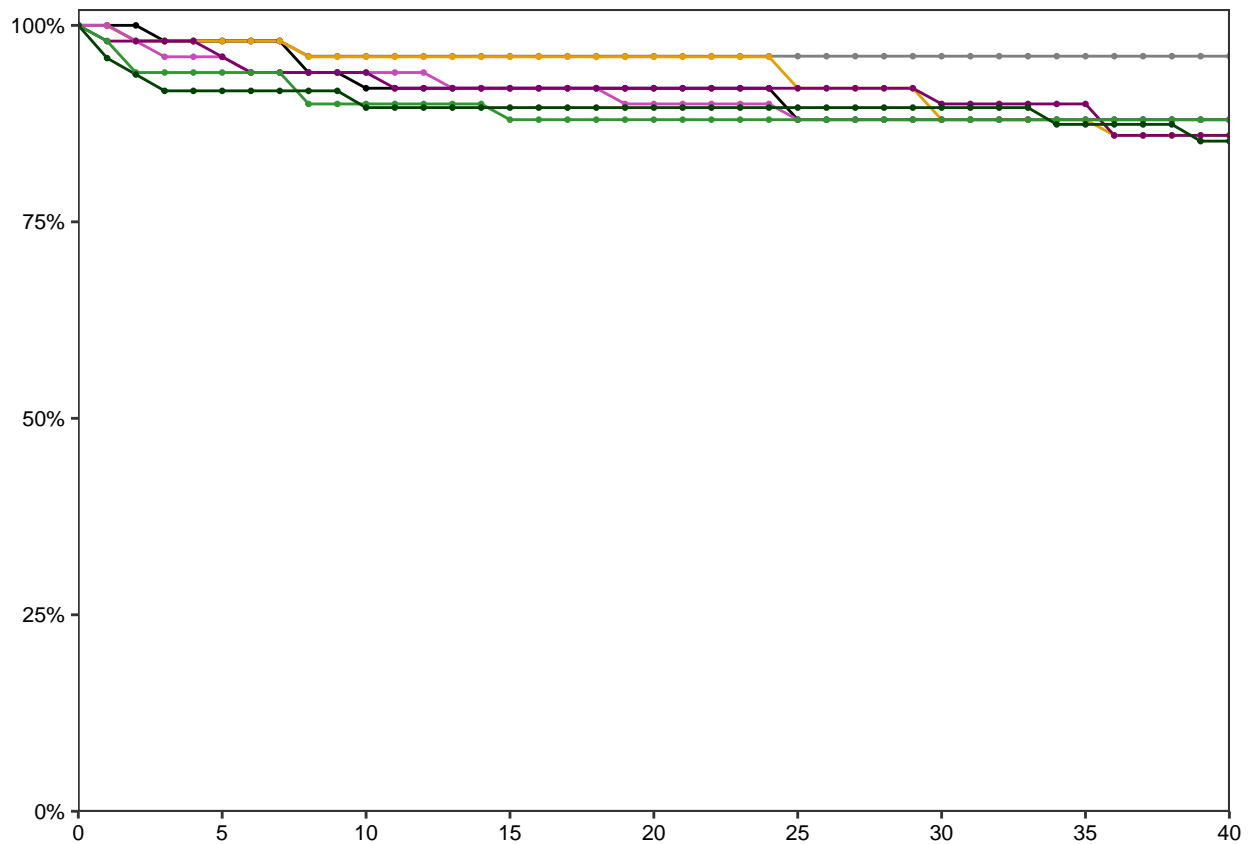
```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 6A.pdf", plot=fig6a, device="pdf", dpi=300, units="in", width=230/96,
)
}
```

```
## fig6b
(fig6b=ggplot(data=dplyr::filter(results_survival_dcv_dataset,
                                experimental_replicate=="1" & infection=="Buffer"),
              aes(x=time, y=surv, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
```

```

scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))
scale_linetype_manual(values=c("dashed", "solid"))+
scale_x_continuous(expand=c(0,0), limits=c(0,40), breaks=c(0,5,10,15,20,25,30,35,40))+
scale_y_continuous(limits=c(0,1.02), expand=c(0,0), labels=percent)+
theme(legend.position="none")

```

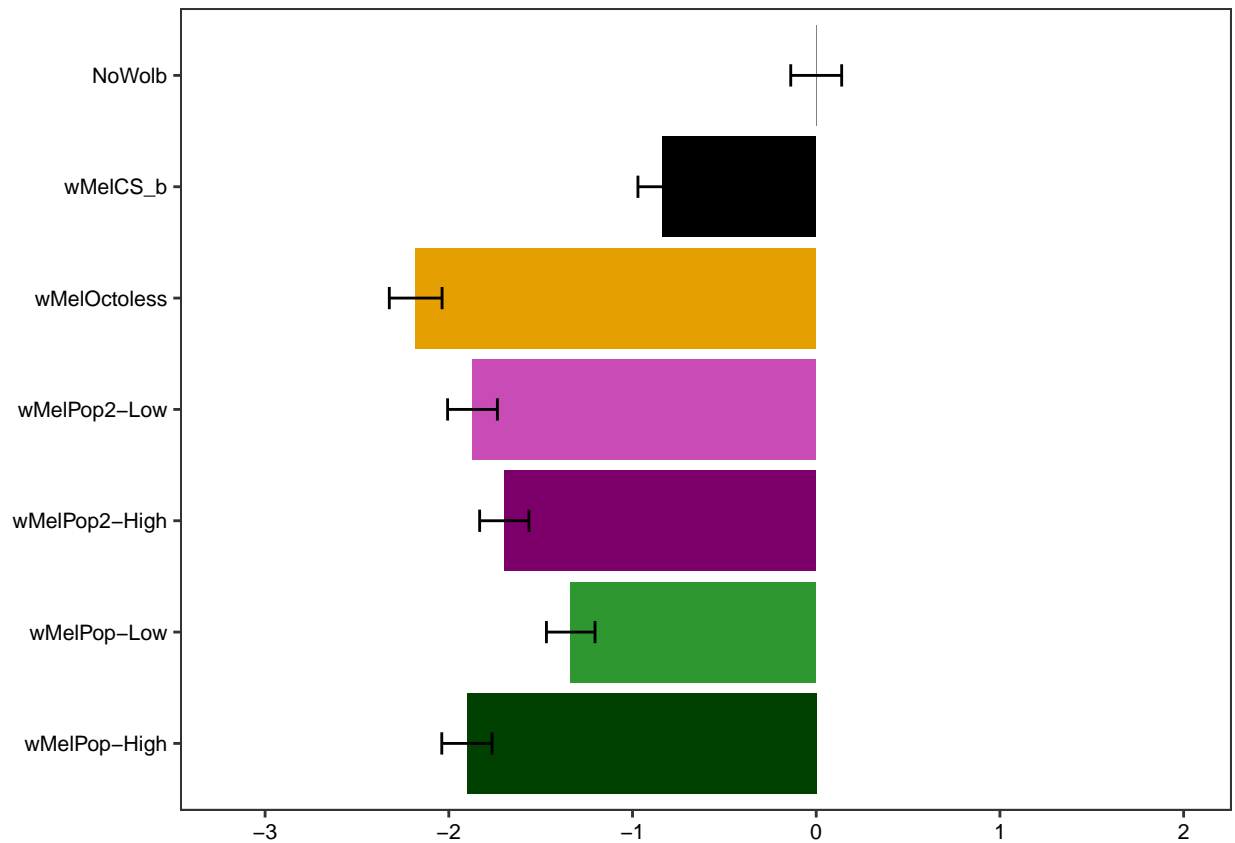


```

# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 6B.pdf", plot=fig6b, device="pdf", dpi=300, units="in", width=230/96,
}

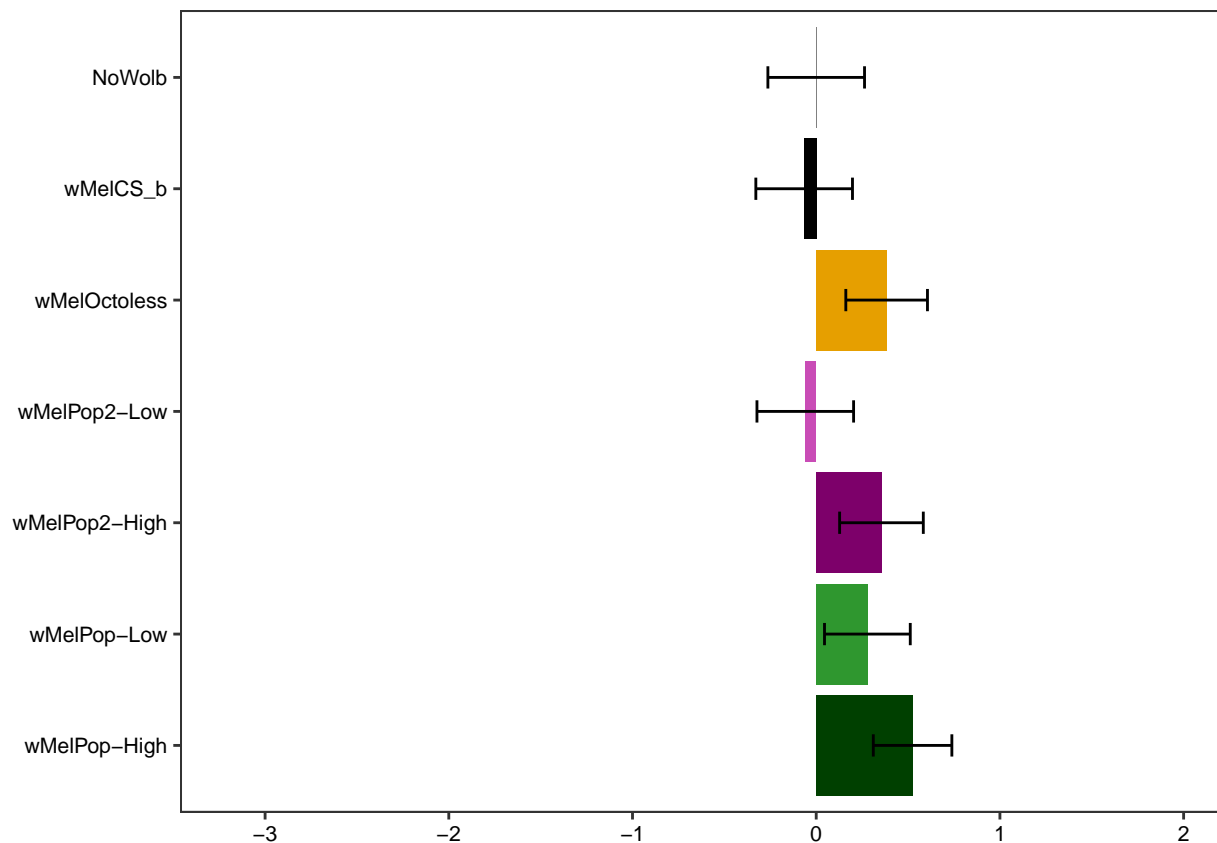
# figure 6c
(fig6c=ggplot(coeff_cox_survival_dcv, aes(x=reorder(wolbachia, desc(wolbachia)), y=DCV.Coeff, fill=wolbachia))
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=DCV.Coeff-DCV.Coeff.SE, ymax=DCV.Coeff+DCV.Coeff.SE), width=0.2, size=0.5)+
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  theme(legend.position="none")+ coord_flip()+
  scale_y_continuous(limits=c(-3.2,2), breaks=seq(-3,2,1)))

```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 6C.pdf", plot=fig6c, device="pdf", dpi=300, units="in", width=260/96,
}

# figure 6d
(fig6d=ggplot(coeff_cox_survival_buffer, aes(x=reorder(wolbachia, desc(wolbachia)), y=DCV.Coeff, fill=wo)
  geom_col()+theme_figs()+
  geom_errorbar(aes(ymin=DCV.Coeff-DCV.Coeff.SE, ymax=DCV.Coeff+DCV.Coeff.SE), width=0.2, size=0.5)+
  scale_fill_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))
  theme(legend.position="none")+ coord_flip()+
  scale_y_continuous(limits=c(-3.2, 2), breaks=seq(-3,2,1)))
```

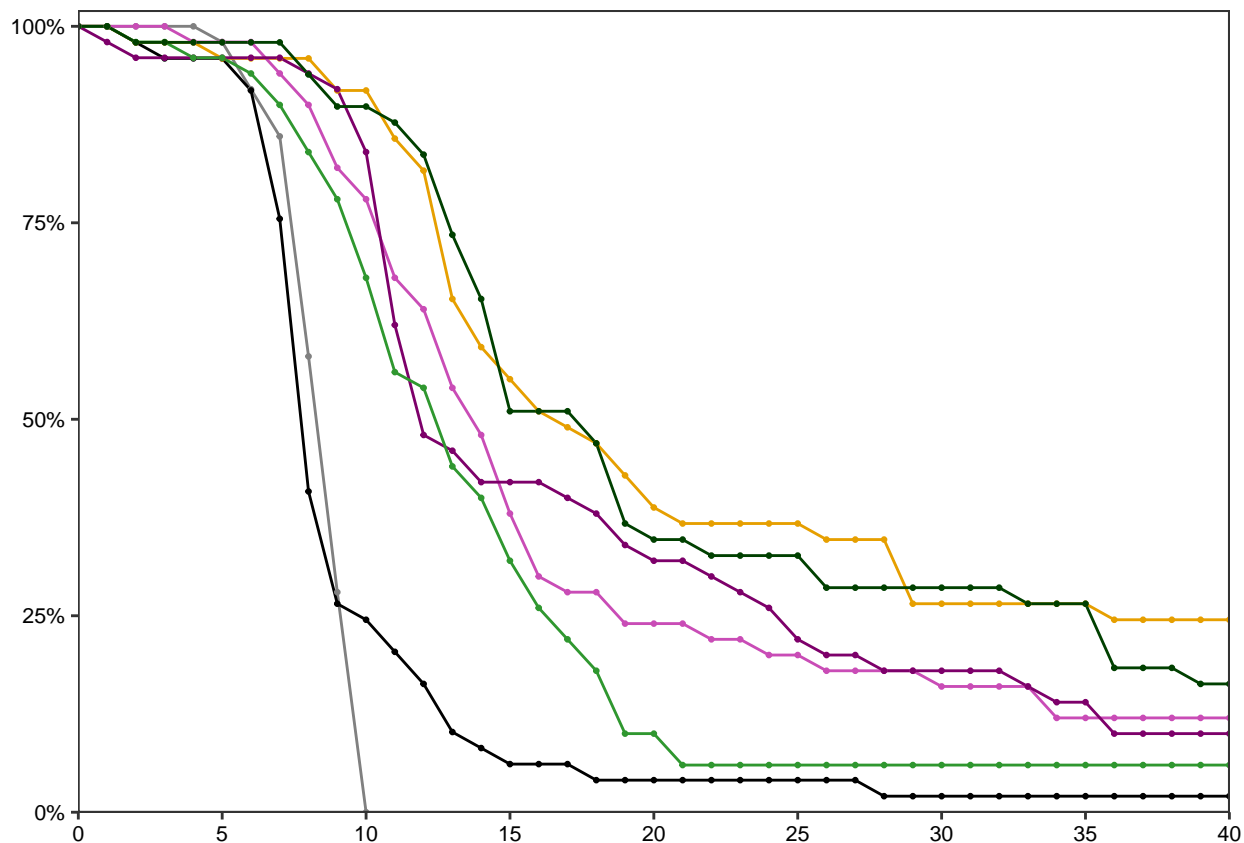


```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 6D.pdf", plot=fig6d, device="pdf", dpi=300, units="in", width=260/96,
}
}
```

## S16A-B Fig

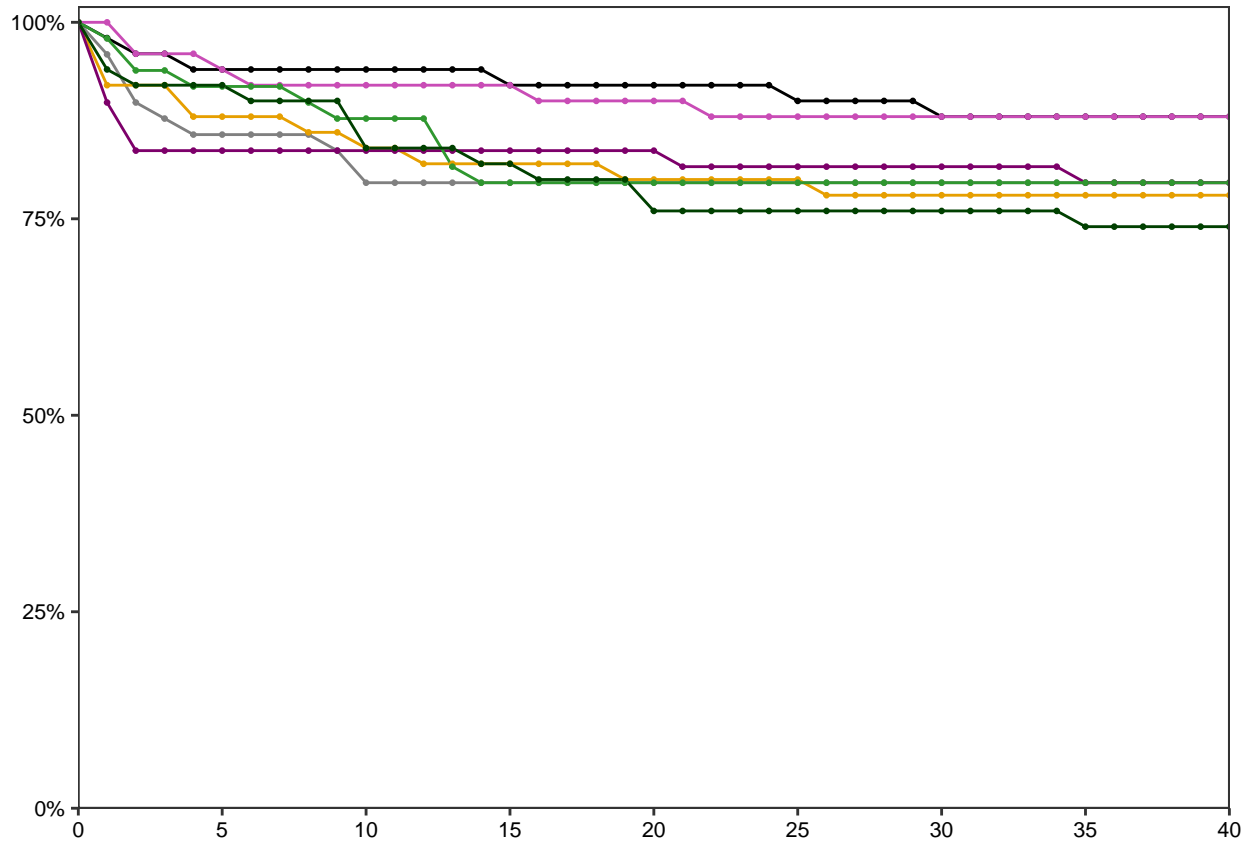
```
## figs16a
(s16afig=ggplot(data=dplyr::filter(results_survival_dcv_dataset,
                                experimental_replicate=="1" & infection=="DCV"),
  aes(x=time, y=surv, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,40), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02),expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```





```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16A Fig.pdf", plot=s16afig, device="pdf", dpi=300, units="in", width=230, height=150)
}

## S16B Fig
(s16bfig=ggplot(data=dplyr::filter(results_survival_dcv_dataset,
                                   experimental_replicate=="2" & infection=="Buffer"),
               aes(x=time, y=surv, color=wolbachia))+
  geom_line(size=0.5)+ geom_point(size=0.5)+ theme_figs()+
  scale_color_manual(values=c("#808080", "#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_linetype_manual(values=c("dashed", "solid"))+
  scale_x_continuous(expand=c(0,0), limits=c(0,40), breaks=c(0,5,10,15,20,25,30,35,40))+
  scale_y_continuous(limits=c(0,1.02), expand=c(0,0), labels=percent)+
  theme(legend.position="none"))
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16B Fig.pdf", plot=s16fig, device="pdf", dpi=300, units="in", width=230/
}

```

Fig 6E | Correlation protection, titres, and proliferation

```
# correlate CHR coefficients of antiviral protection with doubling time
# use doubling times calculate in {r fig_3 :: wolbachia_proliferation_dynamics :: analysis and doubling
# use coefficients calculate in {r fig_6 :: protection_against_dcv :: analysis} - coeff_cox_survival_dc

#merge the two datasets
corr_protection_vs_proliferation <- merge(coeff_cox_survival_dcv, doubling_time_all, by="wolbachia") %>%
  filter(temperature != "29")

### analysis
(results_correlation_protection_vs_proliferation=corr_protection_vs_proliferation%>%
  dplyr::group_by(temperature)%>%
  dplyr::summarise(Estimate=cor.test(doubling_time, DCV.Coeff)$estimate,
    R2=paste0(round((cor.test(doubling_time, DCV.Coeff)$estimate)**2,2)*100,"%"),
    P.value=cor.test(doubling_time, DCV.Coeff)$p.value))

## # A tibble: 2 x 4
##   temperature Estimate R2      P.value
## * <fct>          <dbl> <chr>    <dbl>

```

```
## 1 18          0.596 35%      0.212
## 2 25          0.469 22%      0.348

# correlate CHR coefficients of antiviral protection with time 0 levels
# use time 0 titres from {r s11_fig :: wolbachia_titer_day_of_eclosion :: analysis} and altered in {r f

### merging lifespan and proliferation datasets
corr_protection_vs_titer_day0=merge(coeff_cox_survival_dcv, time_0_levels, by="wolbachia")

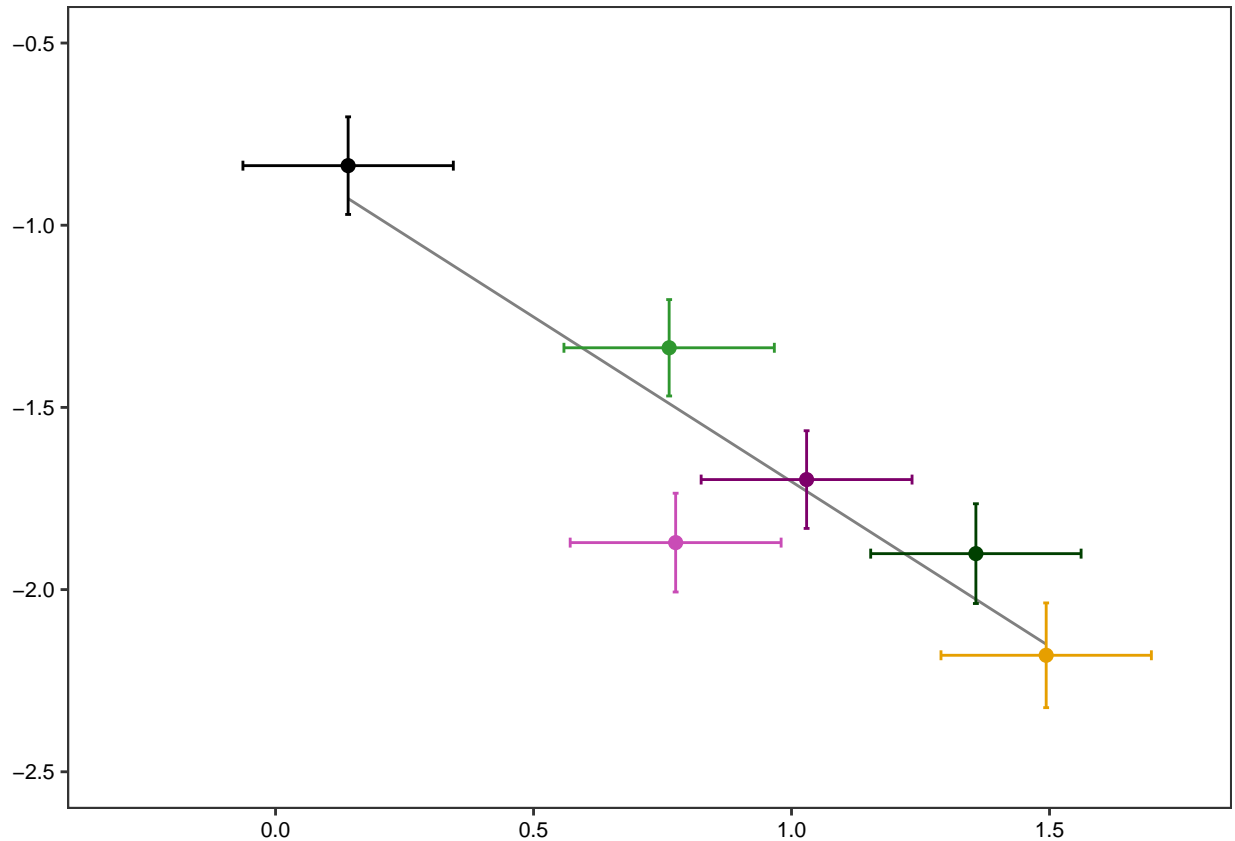
### analysis
(results_correlation_protection_vs_d0 = corr_protection_vs_titer_day0 %>%
  dplyr::summarise(Estimate=cor.test(DCV.Coeff, emmean)$estimate,
    R2=paste0(round((cor.test(DCV.Coeff, emmean)$estimate)**2,2)*100,"%"),
    P.value=cor.test(DCV.Coeff, emmean)$p.value))

##      Estimate R2    P.value
## 1 -0.9157314 84% 0.0103526
```

Fig 6C correlation figure

```
### plot
(fig6e=ggplot(corr_protection_vs_titer_day0, aes(x=emmean, y=DCV.Coeff, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=emmean-SE, xmin=emmean+SE))+
  geom_errorbar(aes(ymax=DCV.Coeff-DCV.Coeff.SE, ymin=DCV.Coeff+DCV.Coeff.SE))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-2.5, -0.5), breaks=seq(-2.5,-0.5,0.5))+
  scale_x_continuous(limits=c(-0.3,1.75), breaks=seq(0,1.5,0.5)))

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

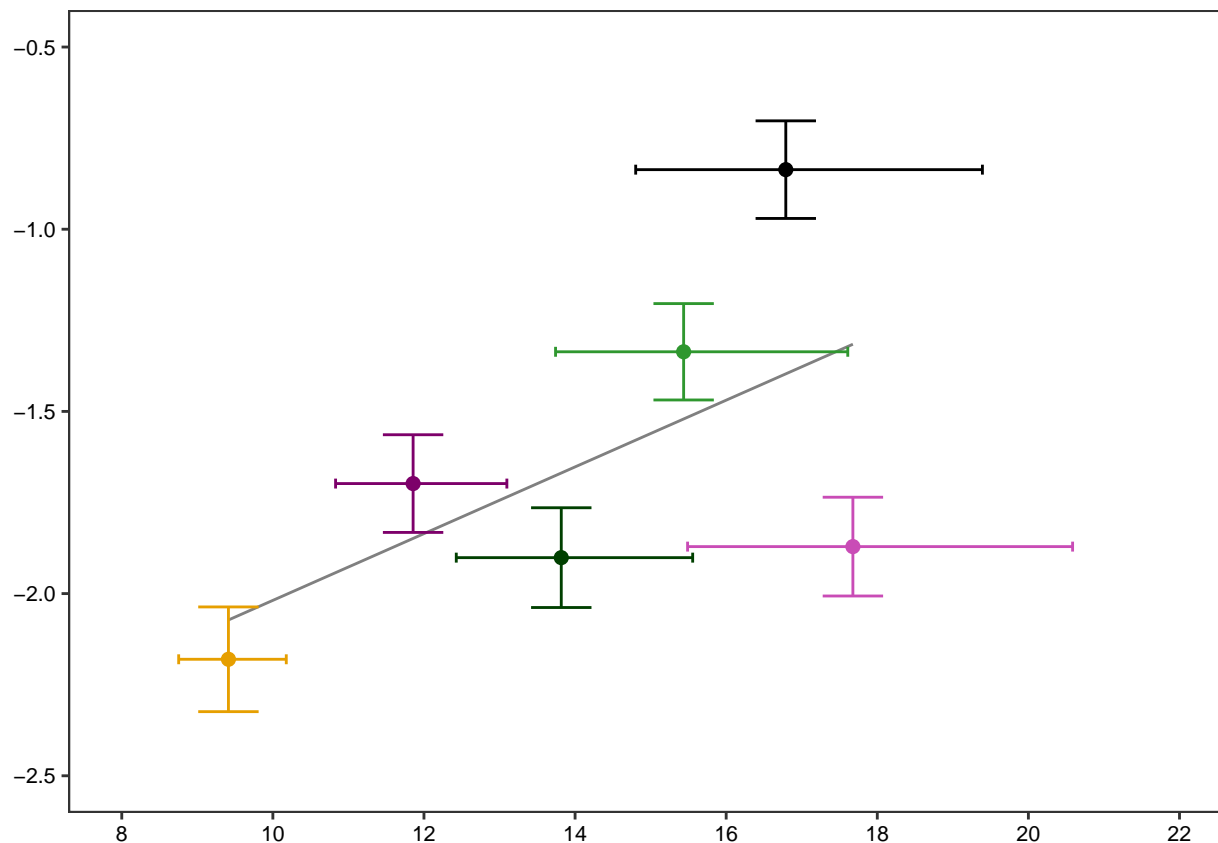


```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/Fig 6E.pdf", plot=fig6e, device="pdf", dpi=300, units="in", width=200/96,
}
}
```

## S16D-E Fig

```
### plot
# S16D Fig
(figs16d=ggplot(dplyr::filter(corr_protection_vs_proliferation, temperature=="18"),
  aes(x=doubling_time, y=DCV.Coeff, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=upper, xmin=lower))+
  geom_errorbar(aes(ymax=DCV.Coeff-DCV.Coeff.SE, ymin=DCV.Coeff+DCV.Coeff.SE))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000"))+
  scale_y_continuous(limits=c(-2.5, -0.5), breaks=seq(-2.5,-0.5,0.5))+
  scale_x_continuous(limits=c(8,22), breaks=seq(8,22,2)))

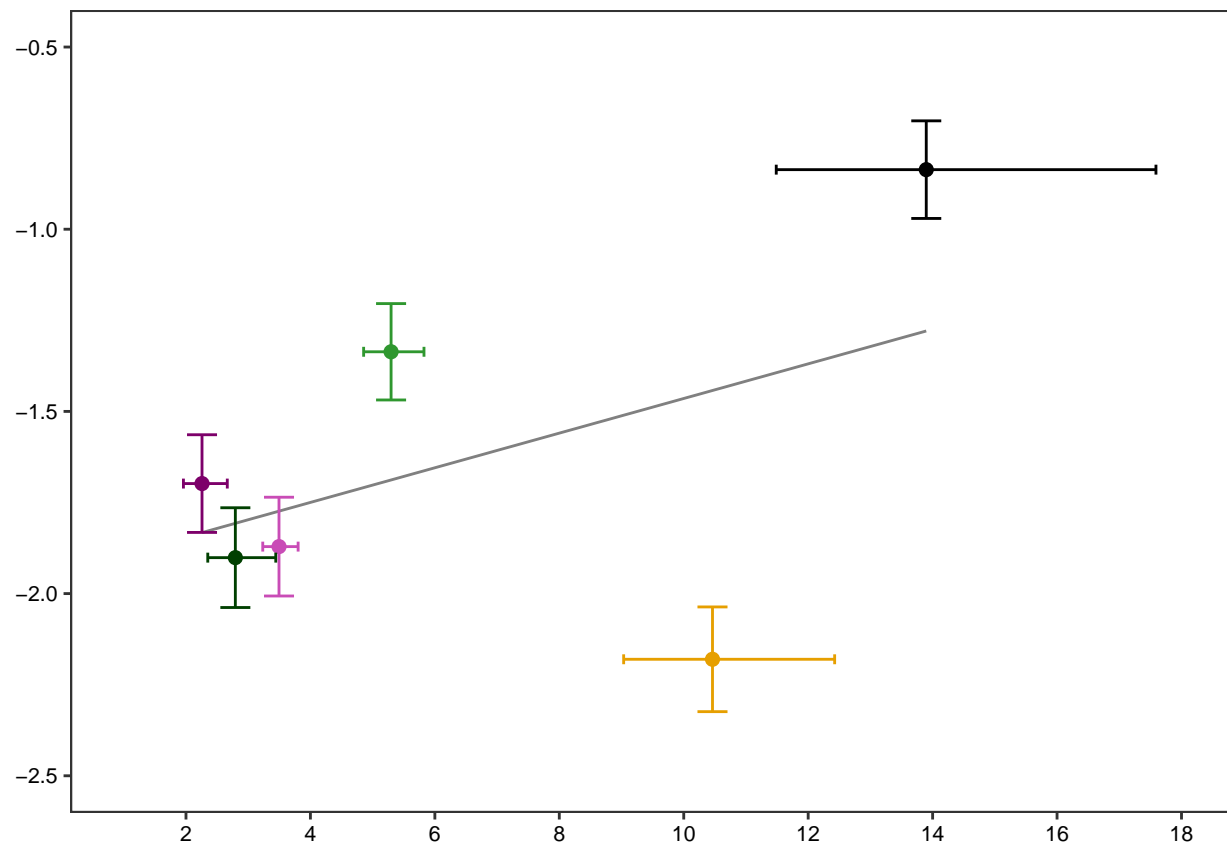
## `geom_smooth()` using formula 'y ~ x'
```



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16D Fig.pdf", plot=figs16d, device="pdf", dpi=300, units="in", width=200, height=200)
}

# S16E Fig
(figs16e=ggplot(dplyr::filter(corr_protection_vs_proliferation, temperature=="25"),
  aes(x=doubling_time, y=DCV.Coeff, color=wolbachia))+
  geom_point(size=2) +
  geom_smooth(method="lm", color="#808080", se=FALSE, size=0.5)+
  geom_errorbarh(aes(xmax=upper, xmin=lower))+
  geom_errorbar(aes(ymax=DCV.Coeff-DCV.Coeff.SE, ymin=DCV.Coeff+DCV.Coeff.SE))+
  theme_figs()+ theme(legend.position="none")+
  scale_color_manual(values=c("#000000", "#e69f00", "#c94cb6", "#7d006a", "#2f972f", "#004000")) +
  scale_y_continuous(limits=c(-2.5, -0.5), breaks=seq(-2.5,-0.5,0.5))+
  scale_x_continuous(limits=c(1,18), breaks=seq(2,18,2)))

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



```
# saving the pdf
if(save_files==TRUE)
{
  ggsave(filename="../results/S16E Fig.pdf", plot=figs16e, device="pdf", dpi=300, units="in", width=200,
}

```