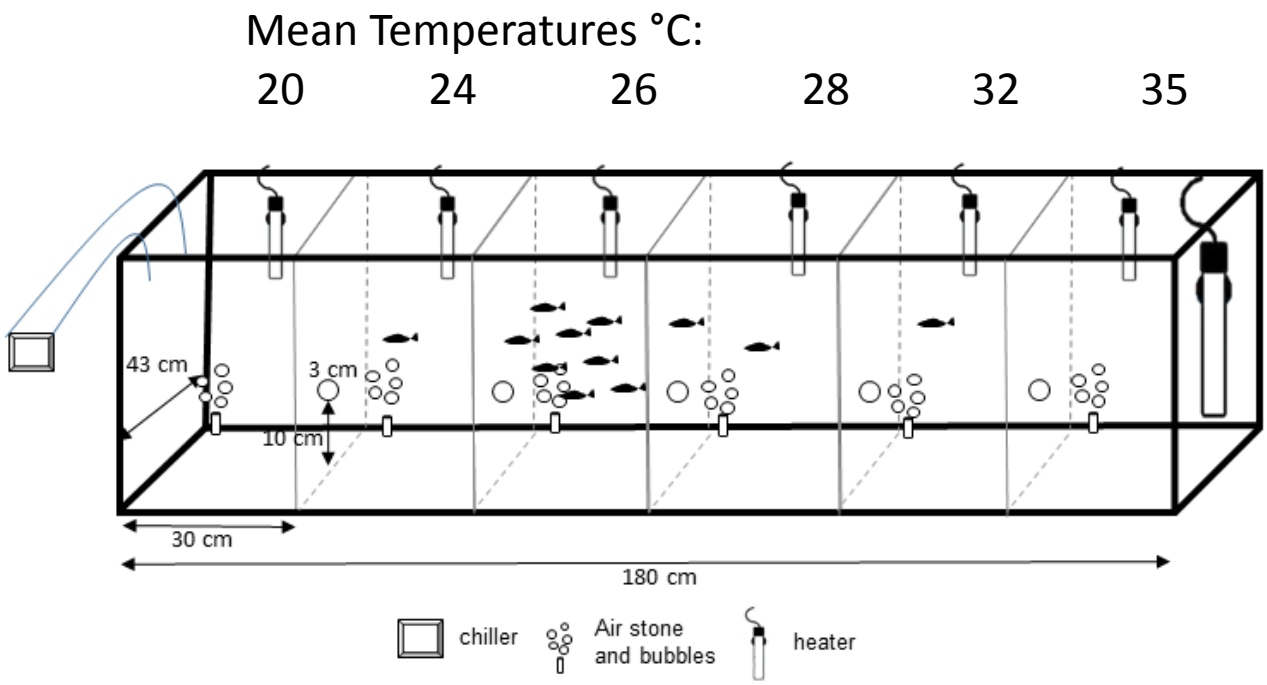

Figure S1. Schematic drawing of the experimental setup used in experiment 1, based off the methods detailed in Rey et al (2015). We included a 1cm deep layer of gravel in all chambers but this was not included in the figures for clarity.

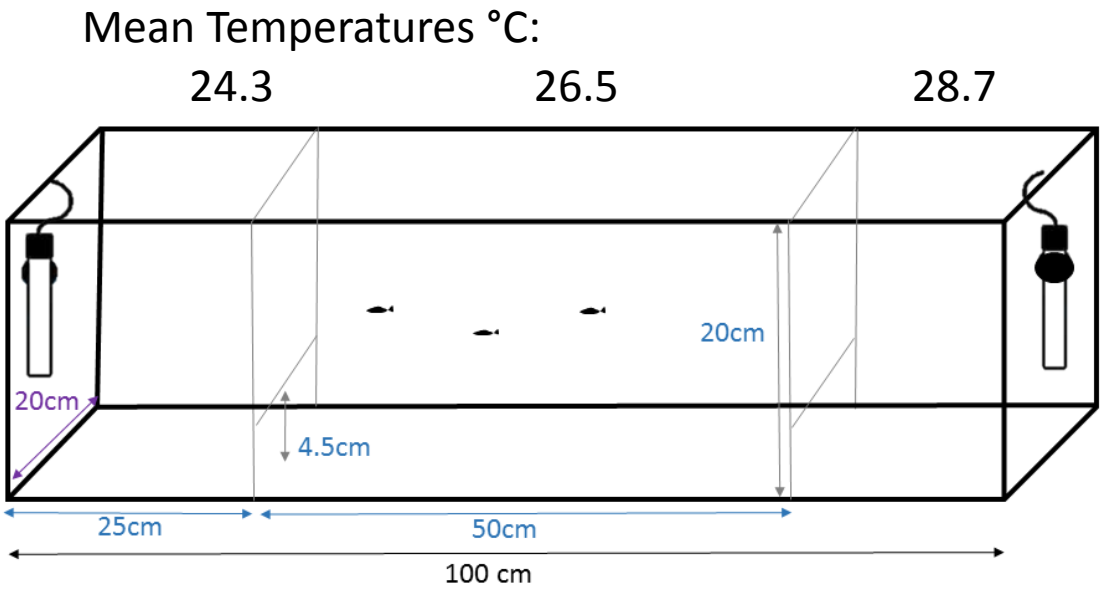

Figure S2. Experimental setup used in experiment 2. Mean temperatures were consistent for the top 15 cm of the warm chamber, but dropped to just over 26.6 in the last 5 cm. Here 0.5 cm deep layer of gravel covered the entire bottom of the tank.

Journal of Experimental Biology • Supplementary information

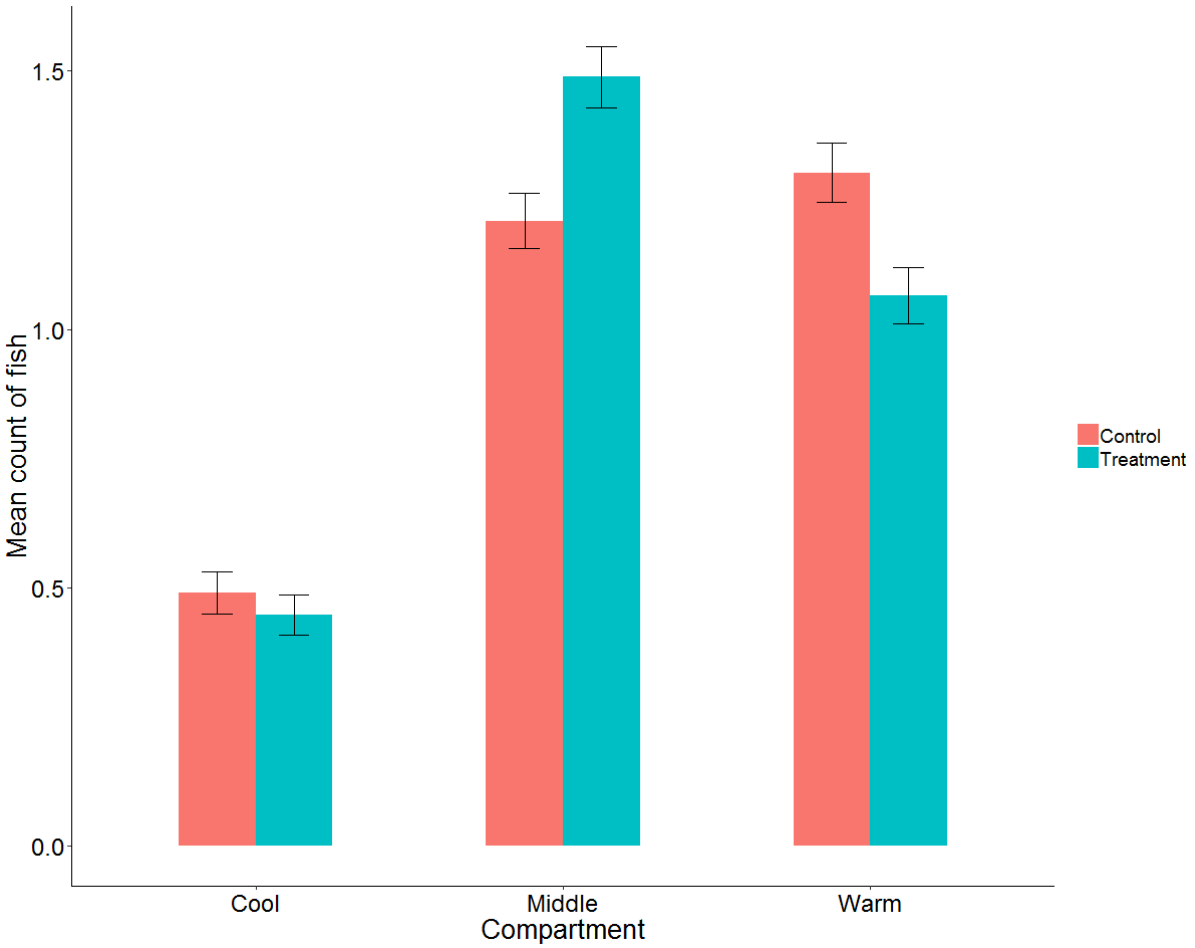

Figure S3. Mean (± s.e.) numbers of fish in each chamber across the two treatments, stressed (Treatment) and unstressed (Control), taken from all count intervals, every five minutes for 2.5 hours. This gives proxy data for compartment utilisation by the fish to supplement Fig 2.

Journal of Experimental Biology • Supplementary information

Script: R code used

Experiment 1

Below is the full R code we used to analyse the data collected for experiment 1.

```
library(ggplot2)
library(plyr)
library(nlme)
library(lme4)
library(mgcv)
library(lattice)
library(lmerTest)
library(MASS)
library(gamm4)
library(lattice)
library(lme4)
library(ggplot2)
library(sp)
library(gstat)
#import data
zebrafish2<- read.table("zebrafish_JEB.txt",header=TRUE)
#Define factors as factors
zebrafish2$Group  <- factor(zebrafish2$Group)
zebrafish2$Treatment  <- factor(zebrafish2$Treatment)
zebrafish2$total<-12
# Defined correctly?
table(zebrafish2$Group)
table(zebrafish2$Treatment)

############################################################
# Data visualization

p <- ggplot()
```

Journal of Experimental Biology • Supplementary information

```
p <- p + geom_point(data = zebrafish2,
                    aes(y = Count, x = Chamber),
                    shape = 1,
                    size = 1)
p <- p + xlab("chamber number") +
       ylab("Number of fish")
p <- p + theme(text = element_text(size = 15))
p
# Is this a linear or non-linear pattern? - looks non-linear!


p <- p + facet_grid(.~ Treatment)
p


#Plot  count vs time
p <- ggplot()
p <- p + geom_point(data = zebrafish2,
                    aes(y = Count, x = Chamber),
                    shape = 16,
                    size = 2)
p <- p + facet_wrap( ~ Time)
p


#standardise continuous variables
zebrafish2$Time2<-(zebrafish2$Time-mean(zebrafish2$Time))/(sd(zebrafish2$Time))#scaling time
zebrafish2$chamber2<-(zebrafish2$Chamber-
mean(zebrafish2$Chamber))/(sd(zebrafish2$Chamber))#scaling chamber
############################################################
# E. Interactions
# Is the quality of the data good enough for an interaction term?


p <- ggplot()
```

```
p <- p + geom_point(data = zebrafish2,
                    aes(y = Count, x = chamber2),
                    shape = 1,
                    size = 1)
p <- p + xlab("Chamber") + ylab("count")
p <- p + theme(text = element_text(size = 15))
p <- p + geom_smooth(data = zebrafish2,
                     aes(x = chamber2,
                         y = Count))

p <- p + facet_grid(. ~ Time2, scales = "fixed")

p
#yes, could potentially use interaction term

#Models:

gam6 =
gamm(cbind(Count,total)~s(chamber2,Time2)+Treatment,random=list(Group=~1),family=binomial,
method="REML",data=zebrafish2,correlation=corSpher(form =~ 1|chamber2,nugget = TRUE, fixed =
FALSE), niterPQL=50)

gam.check(gam6$gam)#ok
plot(gam6$gam)
E6 <- resid(gam6$lme, type = "normalized")
acf(E6)# looks better!
plot(gam6$lme)##looks better
dev.off()
vis.gam(gam6$gam,view=c("chamber2","Time2"),theta=30,phi=30,type="response",color="gray")

#model without the fixed effect treatment:
```

Journal of Experimental Biology • Supplementary information

```
gam6b =
gamm(cbind(Count,total)~s(chamber2,Time2),random=list(Group=~1),family=binomial,method="RE
ML",data=zebrafish2,correlation=corSpher(form =~ 1|chamber2,nugget = TRUE, fixed = FALSE),
niterPQL=50)

#compare both models

#library(itsadug)

AIC(gam6$lme,gam6b$lme)

#          df      AIC

# gam6$lme   8 825.6905

# gam6b$lme  7 819.8443


#model with no treatment effect is better!

#model without the interaction

gam6c =
gamm(cbind(Count,total)~s(chamber2,k=3)+Time2+Treatment,random=list(Group=~1),family=bino
mial,method="REML",data=zebrafish2,correlation=corSpher(form =~ 1|Time2,nugget = TRUE, fixed
= FALSE), niterPQL=50)

#how to compare?

AIC(gam6b$lme,gam6c$lme)


#     df         AIC

# gam6b$lme  7 8.198443e+02

# gam6c$lme  8 4.328029e+07

#model with interaction is better!

#CHOOSE MODEL GAM6B


gam6b =
gamm(cbind(Count,total)~s(chamber2,Time2),random=list(Group=~1),family=binomial,method="RE
ML",data=zebrafish2,correlation=corSpher(form =~ 1|chamber2,nugget = TRUE, fixed = FALSE),
niterPQL=50)


#other approach not taken into account possible autocorrelation


zebrafish2_15<-zebrafish2[zebrafish2$Time==15,]
```

Journal of Experimental Biology • Supplementary information

```
zebrafish2_15<-zebrafish2_15[!zebrafish2_15$Chamber %in% c('1', '5','6'),]
zebrafish2_15.t = xtabs(Count ~ Treatment +Chamber, data = zebrafish2_15)
chisq.test(zebrafish2_15.t)

zebrafish2_30<-zebrafish2[zebrafish2$Time==30,]
zebrafish2_30<-zebrafish2_30[!zebrafish2_30$Chamber %in% c('1', '5','6'),]
zebrafish2_30.t = xtabs(Count ~ Treatment +Chamber, data = zebrafish2_30)
chisq.test(zebrafish2_30.t)

zebrafish2_60<-zebrafish2[zebrafish2$Time==60,]
zebrafish2_60<-zebrafish2_60[!zebrafish2_60$Chamber %in% c('1', '5','6'),]
zebrafish2_60.t = xtabs(Count ~ Treatment +Chamber, data = zebrafish2_60)
chisq.test(zebrafish2_60.t)

zebrafish2_120<-zebrafish2[zebrafish2$Time==120,]
zebrafish2_120<-zebrafish2_120[!zebrafish2_120$Chamber %in% c('1', '5','6'),]
zebrafish2_120.t = xtabs(Count ~ Treatment +Chamber, data = zebrafish2_120)
chisq.test(zebrafish2_120.t)
```

Experiment 2 Preference index

```
library(lattice)
library(readxl)
library(data.table)
library(sjPlot)
library(ggplot2)
library(ggsignif)
library(coefplot2)
library(lme4)
library(car)
library(effects)
library(lsmeans)
library(lmerTest)
```

```
library(rptR)

library(broom)

detach("package:lmerTest", unload=TRUE)

library(plyr)

library(dplyr)

library(pBrackets)


##import data

Zeb5MinOnly=read_excel("…/ESM1_Data.xlsx", sheet = "Expt2_5Min")

head(ZebData)

head(Zeb5MinOnly)

str(ZebData)


Zeb5MinOnly$Condition = as.factor(Zeb5MinOnly$Condition)

Zeb5MinOnly$Group = as.factor(Zeb5MinOnly$Group)

Zeb5MinOnly$Treatment = as.factor(Zeb5MinOnly$Treatment)



zebby5Min <-
ddply(Zeb5MinOnly,c("CountTime","Condition"),summarise,nCounts=length(CountTime), Median =
mean(J_Hot,na.rm=TRUE), JacobsIndex = mean(J_Hot,na.rm=TRUE), Tse=
sd(J_Hot,na.rm=TRUE)/(nCounts)^0.5)


zebby5Min


ggplot(zebby5Min,aes(x=CountTime,y=JacobsIndex))+ theme_classic()+

  geom_point(size=6,aes(shape=Condition))+ geom_errorbar(aes(ymin=JacobsIndex-Tse,
ymax=JacobsIndex+Tse,width=.1)) +ylim(-1.2,1.2) + labs(y = "Jacob's Index", x = "Time (min)")+

  theme(axis.text=element_text(size=18,color = "black"), axis.title=element_text(size=20,color =
"black")) + theme(legend.text=element_text(size=18)) +theme(legend.position = "top") +
theme(legend.title = element_blank()) + scale_shape_manual(values=c(19, 2)) +

   theme(axis.title.y = element_text(margin = margin(r=3)), axis.title.x = element_text(margin =
margin(b=3)))
```

```
grid.brackets(300, 445, 95, 445, lwd=2, col="grey")

grid.brackets(825, 445, 302, 445, lwd=2, col="black")


##Plot for Fig S.3: Time spent in warmer compartment according to groups overall

#1#Caluclate Mean count    of fish in each chamber


#For Warm chamber

CumCountWarm <- ddply(Zeb5MinOnly,c("Condition"),summarise,nCounts=length(Group), Mean = mean(Warm,na.rm=TRUE), Tse= sd(Warm,na.rm=TRUE)/(nCounts)^0.5, Sd = sd(Warm))

#Rename chamber

CumCountWarm$Chamber <- "Warm"


#Quick plots

ggplot(CumCountWarm,aes(x=Condition,y=Mean ))+ theme_classic()+

  geom_point(size=6, shape=18) + geom_errorbar(aes(ymin=Mean-Tse, ymax=Mean+Tse,width=.1))


ggplot(CumCountWarm,aes(x=Condition,y=Mean, fill = Condition ))+ theme_classic()+

  geom_bar(size=6, shape=18, stat = "identity") + geom_errorbar(aes(ymin=Mean-Tse, ymax=Mean+Tse,width=.1))


#for CoolChamber

CumCountCool <- ddply(Zeb5MinOnly,c("Condition"),summarise,nCounts=length(Group), Mean = mean(norm,na.rm=TRUE), Tse= sd(norm,na.rm=TRUE)/(nCounts)^0.5, Sd = sd(norm))

#Plots

ggplot(CumCountCool,aes(x=Condition,y=Mean, fill = Condition ))+ theme_classic()+

geom_bar(size=6, shape=18, stat = "identity") + geom_errorbar(aes(ymin=Mean-Tse, ymax=Mean+Tse,width=.1))

#Rename chamber

CumCountCool$Chamber <- "Cool"


#for Mid

CumCountMid <- ddply(Zeb5MinOnly,c("Condition"),summarise,nCounts=length(Group), Mean = mean(Mid,na.rm=TRUE), Tse= sd(Mid,na.rm=TRUE)/(nCounts)^0.5, Sd = sd(Mid))
```

Journal of Experimental Biology • Supplementary information

```
#Plots

ggplot(CumCountMid,aes(x=Condition,y=Mean ))+ theme_classic()+

  geom_point(size=6, shape=18) + geom_errorbar(aes(ymin=Mean-Tse, ymax=Mean+Tse,width=.1))

ggplot(CumCountMid,aes(x=Condition,y=Mean, fill = Condition ))+ theme_classic()+

  geom_bar(width=0.5, stat = "identity") + geom_errorbar(aes(ymin=Mean-Tse,
ymax=Mean+Tse,width=.1))

#Rename chamber

CumCountMid$Chamber <- "Middle"

##For all chambers at same time merge:

ForSupp <- rbind(CumCountCool,CumCountMid)

ForSuppAll <- rbind(ForSupp,CumCountWarm)


##Plot for figure S.3

ggplot(ForSuppAll,aes(x=Chamber,y=Mean, fill =Condition ))+ theme_classic()+

  geom_bar(width=0.5, stat = "identity", position = "dodge") + geom_errorbar(aes(ymin=Mean-Tse,
ymax=Mean+Tse),width=.2, position=position_dodge(.5)) +

labs(y = "Mean count of fish", x = "Compartment")+

theme(axis.text=element_text(size=18,color = "black"), axis.title=element_text(size=20,color =
"black")) +

 theme(legend.text=element_text(size=14)) + theme(legend.title = element_blank())



#staistical analysis


###Do fish in initial xx minutes post confinement have significantly different Jacob's index of
preference for warm area to those fish that were not confined?

ZebFirst15<- subset(Zeb5MinOnly, subset = CountTime <= 20)

ZebFirst15

ZebFirst40<- subset(Zeb5MinOnly, subset = CountTime <= 46)

ZebFirst40

ZebNotFirst15<- subset(Zeb5MinOnly, subset = CountTime > 46)

ZebNotFirst15
```

Journal of Experimental Biology • Supplementary information

```
ZebLast15<- subset(Zeb5MinOnly, subset = CountTime >= 130)

ZebLast15


plotme = ddply(ZebFirst15,c("Condition"),summarise,MedianJacobsIndex =
median(J_Hot,na.rm=TRUE), MeanJacobsIndex = mean(J_Hot,na.rm=TRUE), Tse=
sd(J_Hot,na.rm=TRUE))  #/(nCounts)^0.5

plotme

ggplot(plotme,aes(x=Condition,y=MeanJacobsIndex, color=Condition))+ theme_classic()+

  geom_point(size=6, shape=18)+ geom_errorbar(aes(ymin=MeanJacobsIndex-Tse,
ymax=MeanJacobsIndex+Tse,width=.1)) +ylim(-1.5,1.5) + labs(y = "JacobsIndex", x = "Time")+

  theme(axis.text=element_text(size=18,color = "black"), axis.title=element_text(size=20,color =
"black")) + theme(legend.text=element_text(size=10)) +theme(legend.position = "top") +
theme(legend.title=element_blank() +

 theme(axis.title.y = element_text(margin = margin(r=3)), axis.title.x = element_text(margin =
margin(b=3))))


ggplot(data = ZebFirst15, aes(y = J_Hot, x = factor(CountTime))) +ylim(-1.1,1.1) + labs(y = "Preference
score (Jacon's Index 'R')", x = "Time (minutes)")+

  geom_boxplot(aes(fill = Condition))


ggplot(data = Zeb5MinOnly, aes(y = J_Hot, x = factor(CountTime))) +ylim(-1.1,1.1)+ labs(y =
"Preference score", x = "Time (minutes)")+

  geom_boxplot(aes(fill = Condition))




plot(J_Hot ~ Condition, data = ZebFirst15, ylim=c(-1.5,1.5), main ="Initial 15 min")

boxplot(J_Hot ~ Condition + CountTime, data =Zeb5MinOnly , ylim=c(-1,1), main ="After initial min")

plot(J_Hot ~ Condition, data = ZebData, ylim=c(-1,1), main ="All time min")
```

Journal of Experimental Biology • Supplementary information

##t test for quick comaprison - In first 15 minutes do fish in different treatments have different preferences for the hot chamber?

t.test(ZebFirst15$J_Hot~ZebFirst15$Condition) ## sig

t.test(ZebFirst15$J_Hot~ZebFirst15$Condition, var.equal = TRUE) ## sig

t.test(ZebNotFirst15$J_Hot~ZebNotFirst15$Condition) ## sig

t.test(ZebNotFirst15$J_Hot~ZebNotFirst15$Condition, var.equal = TRUE) ## sig

##check for normality

qqPlot(ZebFirst15$J_Hot) ## decent...

##F test for equal variance

res.ftest <- var.test(J_Hot ~ Condition, data = ZebFirst15)

res.ftest ## equal variance

##t test for comaprison of 'Post' period

t.test(DPost15min$J_Hot~DPost15min$Treatment) ## Not sig - So No!

t.test(DPost15min$J_Hot~DPost15min$Treatment, var.equal = TRUE) ## No

##check for normality

qqPlot(ZebNotFirst15$J_Hot) ## no

##F test for equal variance

Journal of Experimental Biology • Supplementary information

```
res.ftest <- var.test(J_Hot ~ Treatment, data = ZebNotFirst15)
res.ftest ## very equal variance

##t test for  comaprison of 'End' period
t.test(ZebLast15$J_Hot~ZebLast15$Condition) ## Not sig  -

t.test(ZebLast15$J_Hot~ZebLast15$Condition, var.equal = TRUE)  ## No

##check for normality
qqPlot(ZebNotFirst15$J_Hot)  ## no
##F test for equal variance
res.ftest <- var.test(J_Hot ~ Condition, data = ZebNotFirst15)
res.ftest ## very equal variance

##t test for  comaprison of 'Total' period

t.test(Zeb5MinOnly$J_Hot~Zeb5MinOnly$Condition) ## Yes, sig

t.test(Zeb5MinOnly$J_Hot~Zeb5MinOnly$Condition, var.equal = TRUE)  ## Yes


##check for normality

qqPlot(ZebLast15$J_Hot)  ## no
##F test for equal variance
res.ftest <- var.test(J_Hot ~ Condition, data = ZebLast15)
res.ftest ## very equal variance


## t-test for each CountInterval
```

Journal of Experimental Biology • Supplementary information

```
##Assumptions

##Check for Normality with S-Wilkes test
with(Zeb5MinOnly, shapiro.test(J_Hot[Condition == "Treatment"]))# p = 0.0001 - Failed, use Wilcoxon
# Shapiro-Wilk normality test
with(Zeb5MinOnly, shapiro.test(J_Hot[Condition == "Control"])) # p = 0.0006

#Wilcoxon test - reported in results

resAll <- wilcox.test(J_Hot ~ Condition, data = Zeb5MinOnly,
                      exact = FALSE)
resAll

#First 20 minutes
resFirst <- wilcox.test(J_Hot ~ Condition, data = ZebFirst15,
                        exact = FALSE)
resFirst

#First 46 minutes
resFirst40 <- wilcox.test(J_Hot ~ Condition, data = ZebFirst40,
                          exact = FALSE)
resFirst40

#last 15 minutes
resEnd <- wilcox.test(J_Hot ~ Condition, data = ZebNotFirst15,
                      exact = FALSE)
resEnd
```

Experiment 2 R code – Breakpoint analysis

Below is the full R code we used to extract the inflection point of change in the time series of the Jacobs preference index for the net and control treatment time series.

Journal of Experimental Biology • Supplementary information

```
library(caTools)
data_exp3 = data.frame(read.csv('Exp3_forR.csv', header = TRUE))

#----separate in control and treatment---#
control = data_exp3[data_exp3$Condition == 'Control',]
net = data_exp3[data_exp3$Condition == 'Net',]

#==============----------NET TREATMENT--------=================#

data_in = net
ucount = unique(data_in$CountTime)
mean_control = rep(NA, length(ucount))

for (i in 1:length(ucount)){
  dat = data_in[data_in$CountTime == ucount[i],]
  mean_control[i] = mean(dat$J_Hot)
}
net_series = mean_control

#----plot chamber preferences vs. counts for net treatment----#
frame()
par(mfrow = c(1,1))
plot(ucount, mean_control, type = 'l', ylim = c(-2,2), col = 'red', xlab = 'count', ylab = 'preference
index', main = 'Net Treatment Response')


#=========-----Calculate inflection points using a loess smoothing function-----====#

#---step 1: Smooth data----#
frame()
```

Journal of Experimental Biology • Supplementary information

```
par(mfrow = c(3,1))

par(mar = c(3,5,2,3))

y = net_series

x = ucount

plot(x,y, type = 'l', ylim = c(-2,2), ylab = 'Jacobs index', lwd = 2, cex.lab = 1.8, cex.axis = 1.5, xlab = '')

lo <- loess(y~x) #use a loess smoothing function to smooth data

xl <- seq(min(x),max(x), (max(x) - min(x))/1000)

out = predict(lo,xl)

#plot smoothed index

plot(xl, out, type = 'l', lwd = 2, cex.lab = 1.8, cex.axis = 1.5, ylab = 'smoothed Jacobs index', xlab = '')



#---step 2: find places in the smoothed y values where the change in y switches sign.

infl <- c(FALSE, diff(diff(out)>0)!=0)


# add points to the graph where these inflections occur.

diff_out = diff(out)

plot(xl[1:length(xl)-1], diff_out,type = 'l', lwd = 2, cex.lab = 1.8, cex.axis = 1.5,  xlab = 'time (min)', ylab
= 'differenced smoothed Jacobs index')

infl_t = which(infl==TRUE)

points(xl[infl_t[1]], diff_out[infl_t[1]], pch = 21, cex = 2)

legend('topright', 'inflection point', pch = 21, cex = 1.4)


#---extract inflection point---#

inf_net = xl[infl_t[1]] #46.6 min


#==============----------CONTROL TREATMENT--------=================#

data_in = control

ucount = unique(data_in$CountTime)

mean_control = rep(NA, length(ucount))
```

Journal of Experimental Biology • Supplementary information

```
for (i in 1:length(ucount)){
  dat = data_in[data_in$CountTime == ucount[i],]
  mean_control[i] = mean(dat$J_Hot)
}

control_series = mean_control

#----plot preference index vs. counts for control groups----#
frame()
par(mfrow = c(1,1))
plot(ucount,control_series, type = 'l', ylim = c(-2,2), xlab = 'count', ylab = 'preference index', main =
'Control Treatment Response')

#=========-----Calculate inflection points using a loess smoothing function-----====#

#---step 1: Smooth data----#
frame()
par(mfrow = c(3,1))
par(mar = c(3,5,2,3))
y = control_series
x = ucount
plot(x,y, type = 'l', ylim = c(-2,2), ylab = 'Jacobs index', lwd = 2, cex.lab = 1.8, cex.axis = 1.5, xlab = '')
lo <- loess(y~x) #uses a loess smoothing function
xl <- seq(min(x),max(x), (max(x) - min(x))/1000)
out = predict(lo,xl)
#---plot smoothed Jacobs Index---#
plot(xl, out, type = 'l', lwd = 2, cex.lab = 1.8, cex.axis = 1.5, ylab = 'smoothed Jacobs index', xlab = '')

#---step 2: find places in the smoothed y values where the change in y switches sign.
infl <- c(FALSE, diff(diff(out)>0)!=0)
```

```
# add points to the graph where these inflections occur.

diff_out = diff(out)

plot(xl[1:length(xl)-1], diff_out,type = 'l', lwd = 2, cex.lab = 1.8, cex.axis = 1.5,  xlab = 'time (min)', ylab = 'differenced smoothed Jacobs index')

infl_t = which(infl==TRUE)

points(xl[infl_t[1]], diff_out[infl_t[1]], pch = 21, cex = 2)

legend('topright', 'inflection point', pch = 21, cex = 1.4)


#---extract inflection point---#

inf_control = xl[infl_t[1]] #31.4 min
```

Table S1. Raw data from all zebrafish hyperthermia experiments.

Click here to Download Table S1

Journal of Experimental Biology • Supplementary information