

# AppendixSIV

*Julie van der Hoop*

*12/8/2017*

This R Markdown document details the statistical approach for the paper van der Hoop et al. 201X “Turbulent flow reduces oxygen consumption in the labriform swimming shiner perch, *Cymatogaster aggregata*”

The data represent mean oxygen consumption rates (MO<sub>2</sub>; mg O<sub>2</sub>/kg/h) from three measurements at each speed (repetition number) of increments from 0.5 Body Lengths (BL) per second [BL/s] up to a maximum swimming speed, for different Fish that swam each in High (“T”) and Low (“L”) turbulence flow conditions in a respirometer. Bacterial MO<sub>2</sub> (mg O<sub>2</sub>/h) for each trial are subtracted from the measured MO<sub>2</sub>.

```
## 1. Data handling

# load data
library(readxl)
fish <- read_xlsx("FHL_FishVO2_allreps.xlsx")

# make the things that need to be factors factors
fish$Fish <- as.factor(fish$Fish)
fish$Flow <- as.factor(fish$Flow)
fish$Rep <- as.factor(fish$Rep)

## 2. Model Fitting
library(nlme)

# We want to fit something of the form
#  $O_2 \sim a + b \cdot \text{Speed}^c$ 
# estimate a,b,c and have a random effect for fish
# each a,b,c has two levels, one for each treatment (Low-Turbulence and High-Turbulence)

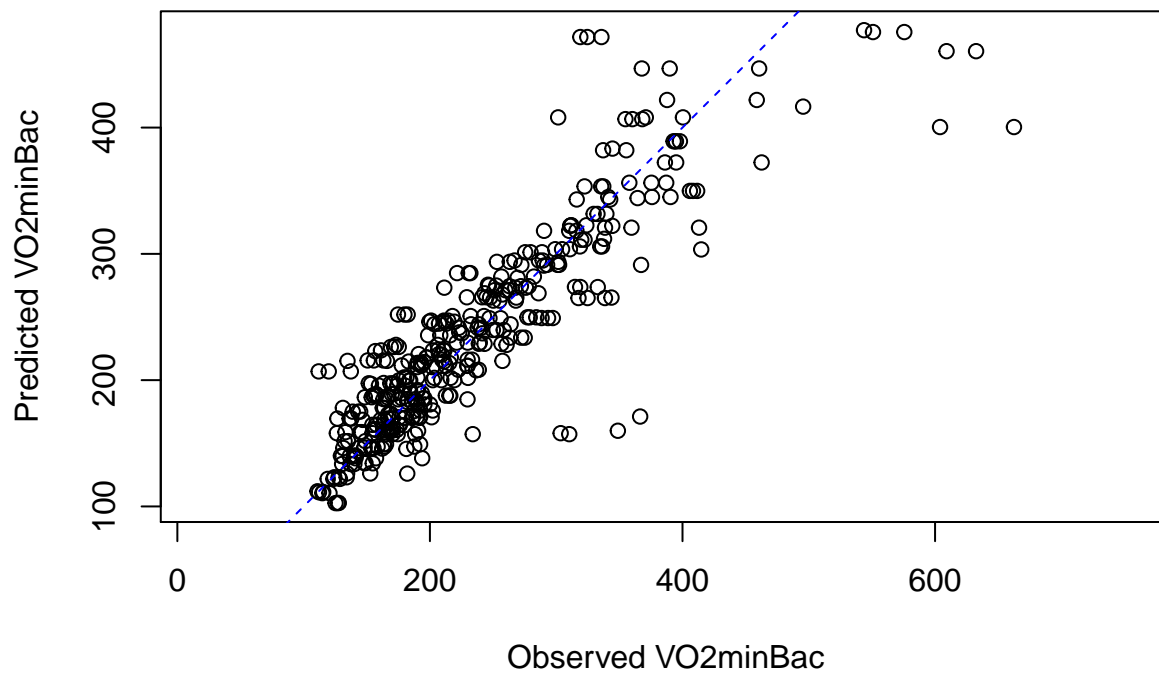
# NOTES:
# - formula is NOT a standard R formula
# - fixed specifies the form for the fixed effects (the a,b,c parameters)
# - random says what the random effects are (random intercept for a)
modr <- nlme(VO2minBac~a+b*Speed^c,
             fixed = a+b+c~Flow,
             random = a~1|Fish,
             start=c(168.7038472,0, 6,0, 2.2036793, 0),
             data=fish)

# Check model summary
summary(modr)

## Nonlinear mixed-effects model fit by maximum likelihood
## Model: VO2minBac ~ a + b * Speed^c
## Data: fish
##      AIC      BIC    logLik
## 4009.96 4041.523 -1996.98
##
## Random effects:
## Formula: a ~ 1 | Fish
```

```
##          a.(Intercept) Residual
## StdDev:      30.46156 43.61433
##
## Fixed effects: a + b + c ~ Flow
##              Value Std.Error DF   t-value p-value
## a.(Intercept) 157.03508 13.258172 369 11.844399 0.0000
## a.FlowT       -8.18116 11.560896 369 -0.707658 0.4796
## b.(Intercept) 13.79768 3.597576 369 3.835272 0.0001
## b.FlowT       -3.00009 6.308700 369 -0.475548 0.6347
## c.(Intercept) 2.08103 0.172225 369 12.083225 0.0000
## c.FlowT       -0.18525 0.371927 369 -0.498087 0.6187
## Correlation:
##          a.(In) a.FlWt b.(In) b.FlWt c.(In)
## a.FlowT      -0.378
## b.(Intercept) -0.481 0.551
## b.FlowT       0.274 -0.849 -0.569
## c.(Intercept) 0.450 -0.516 -0.992 0.564
## c.FlowT      -0.208 0.778 0.457 -0.983 -0.460
##
## Standardized Within-Group Residuals:
##          Min          Q1          Med          Q3          Max
## -3.49889427 -0.48623188 -0.03961182 0.33271482 6.00907570
##
## Number of Observations: 382
## Number of Groups: 8
```

```
# plot observed vs predicted
plot(fish$VO2minBac, predict(modr),
     asp=1,
     xlab="Observed VO2minBac", ylab="Predicted VO2minBac")
abline(a=0,b=1, col="blue", lty=2)
```

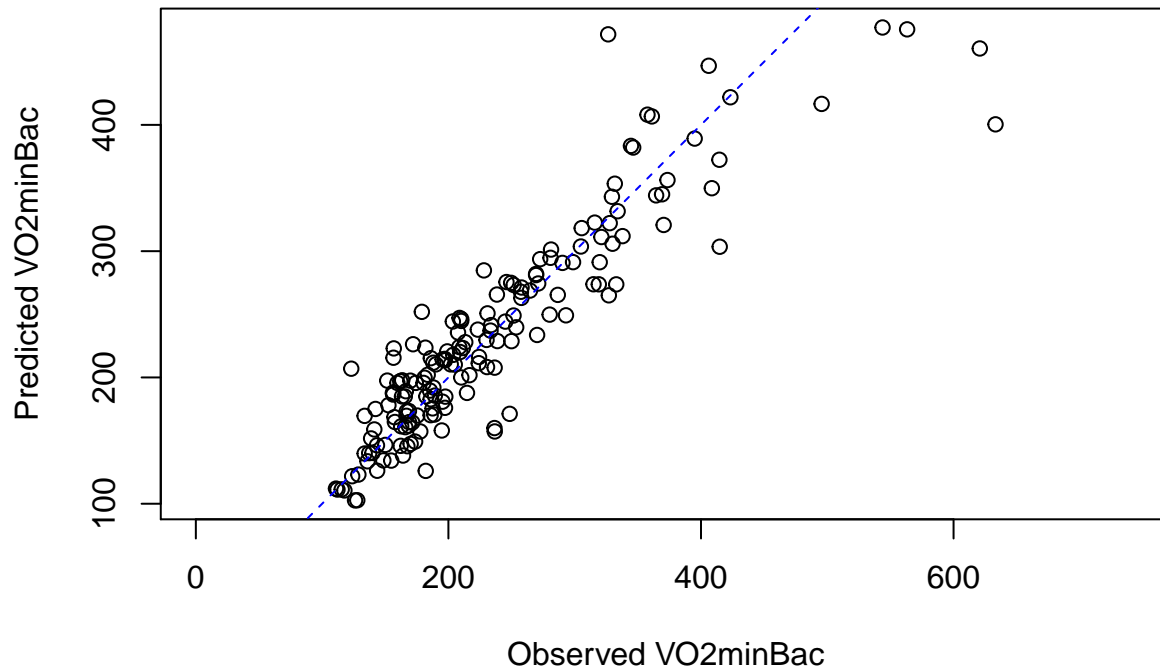


```

# same but average over repetitions
# (a bit cleaner)
plot(aggregate(fish$VO2minBac, list(fish$Speed, fish$Flow, fish$Fish), mean)$x,
     aggregate(predict(modr), list(fish$Speed, fish$Flow, fish$Fish), mean)$x,
     main="modr - aggregated observed vs predicted",
     asp=1,
     xlab="Observed VO2minBac", ylab="Predicted VO2minBac")
abline(a=0,b=1, col="blue", lty=2)

```

### modr – aggregated observed vs predicted

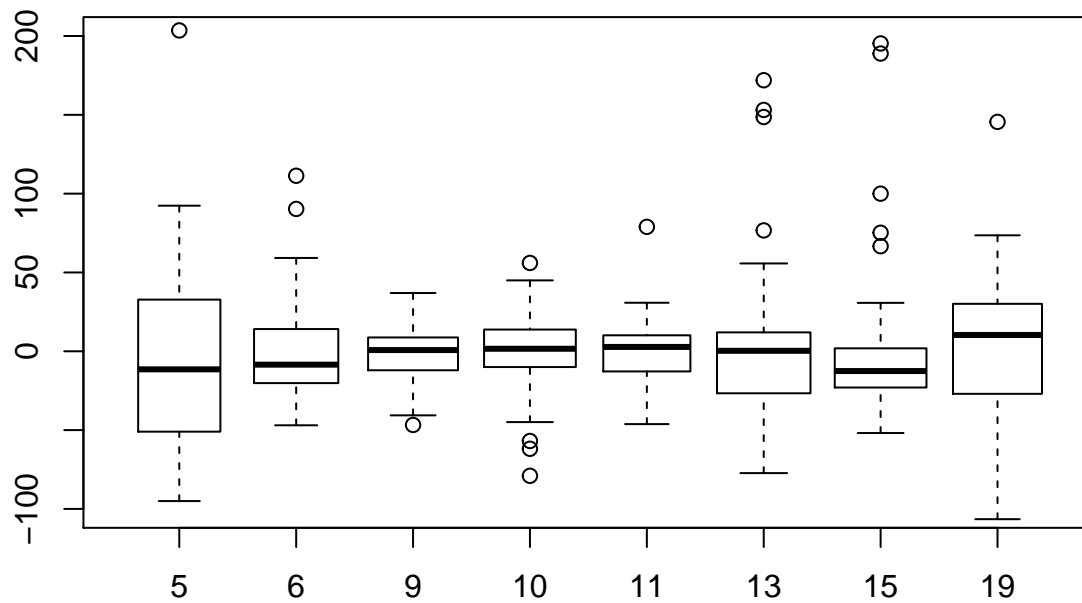


```

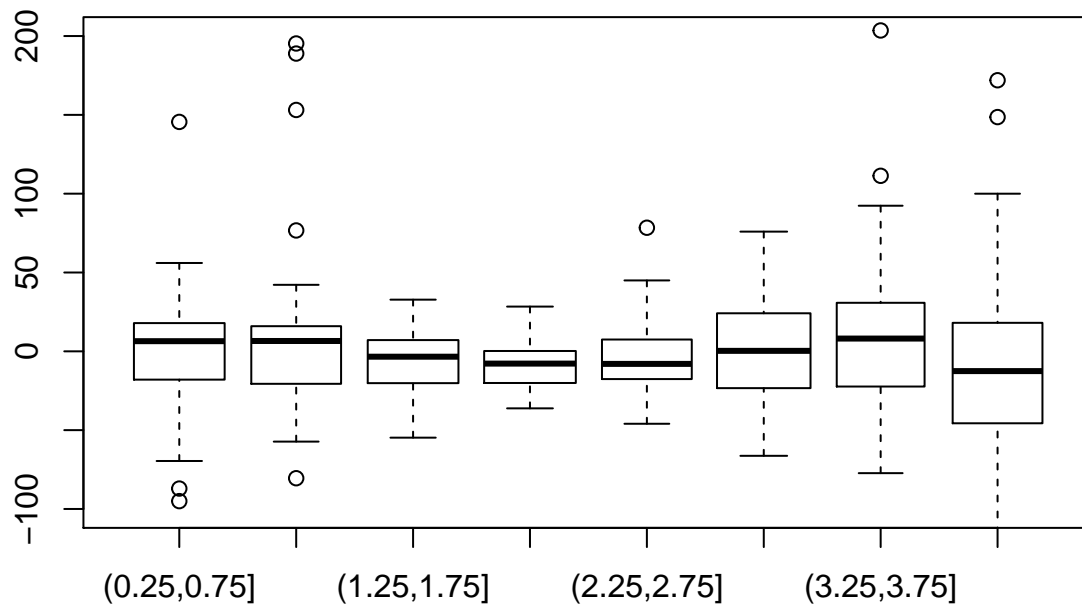
## 3. Model checking

# per fish residuals - this looks okay, no major variations
bp_dat <- data.frame(resids = residuals(modr),
                    Fish    = fish$Fish)
boxplot(resids~Fish, bp_dat, varwidth=TRUE, ylim=c(-100, 200))

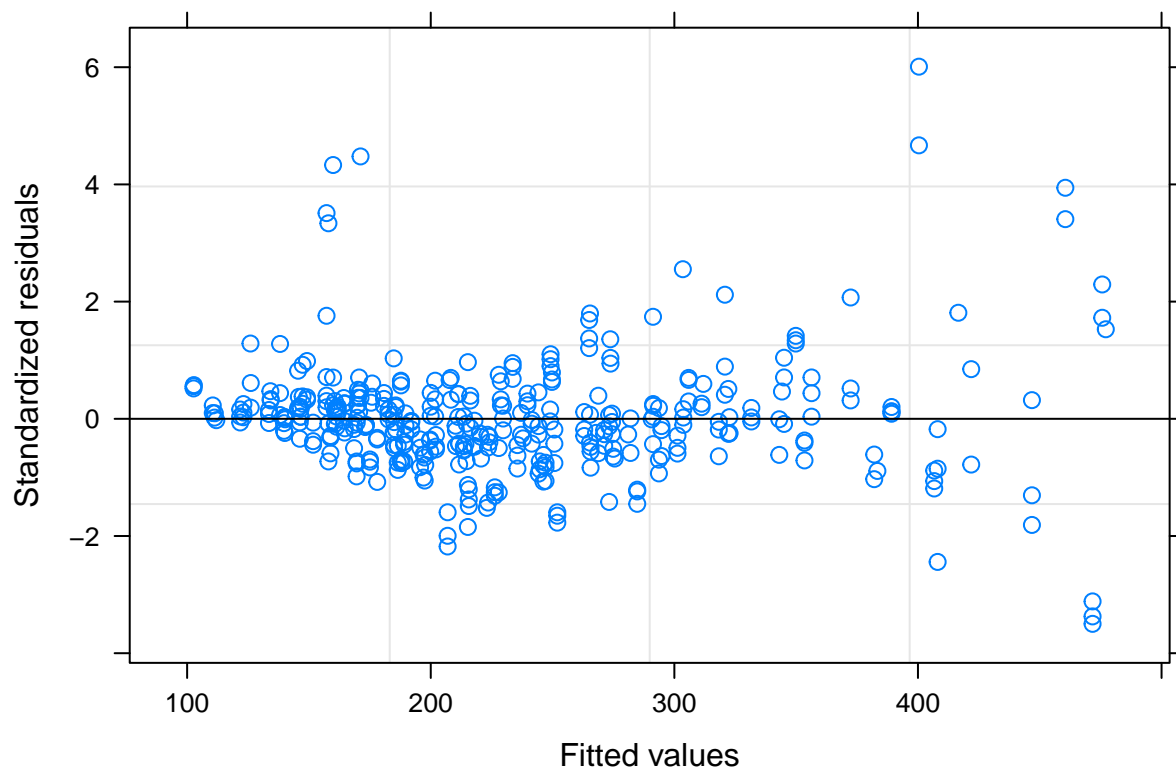
```



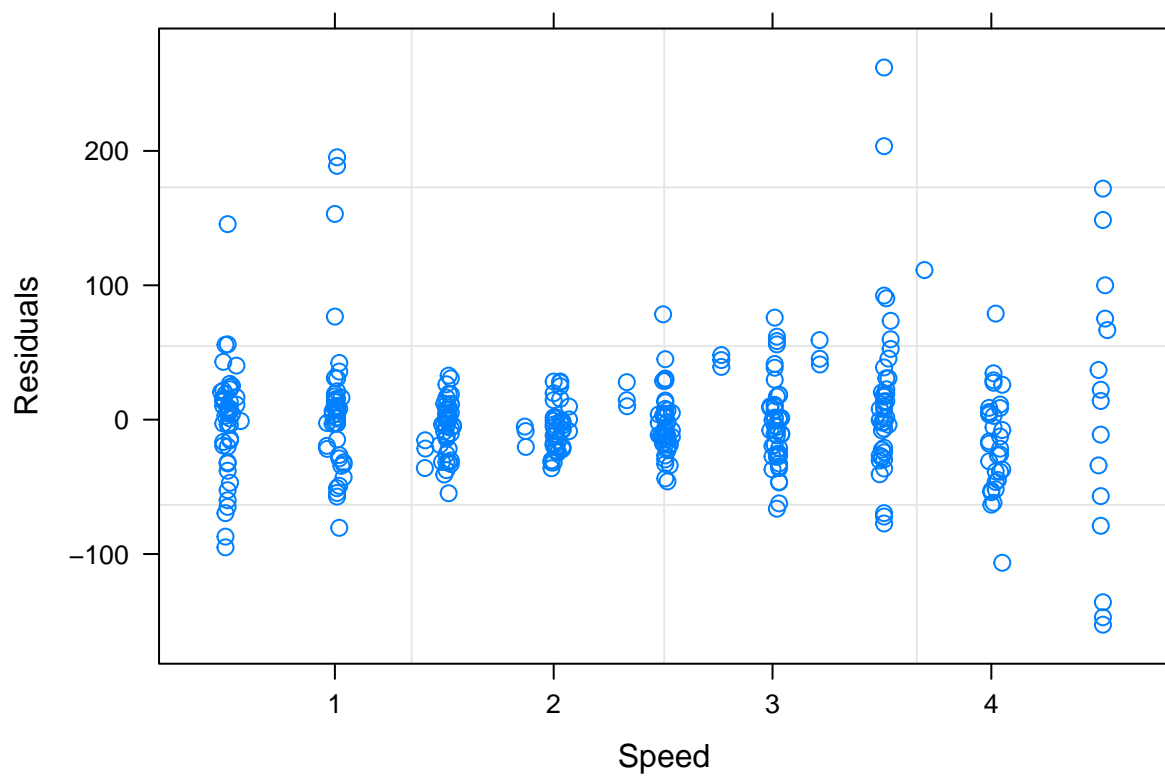
```
# residuals by Speed
# - these look okay too, maybe some increase in variance at higher speeds
dat <- data.frame(residuals = residuals(modr),
#               Speed      = cut(fish$Speed, seq(0.25,4.75,0.5)))
               Speed      = cut(fish$Speed, c(seq(0.25,3.75,0.5), 4.75)))
boxplot(residuals~Speed, dat, varwidth=TRUE, ylim=c(-100, 200))
```



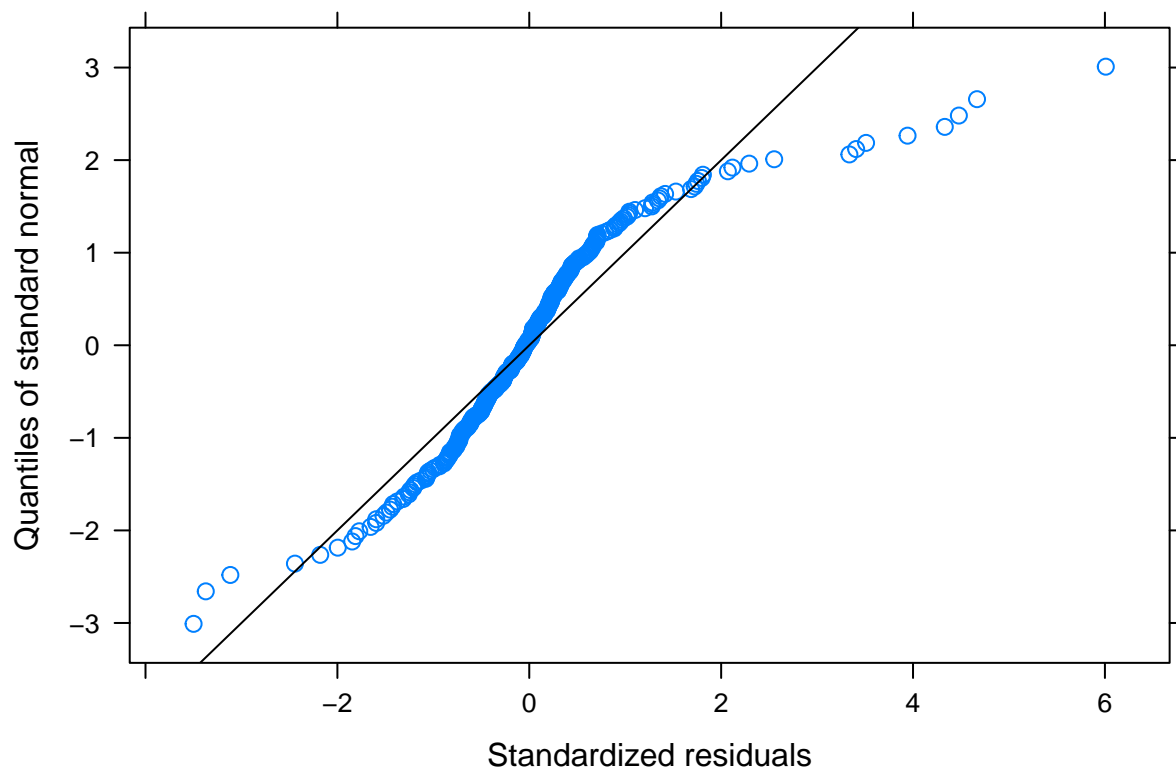
```
# predictions vs. residuals
plot(modr)
```



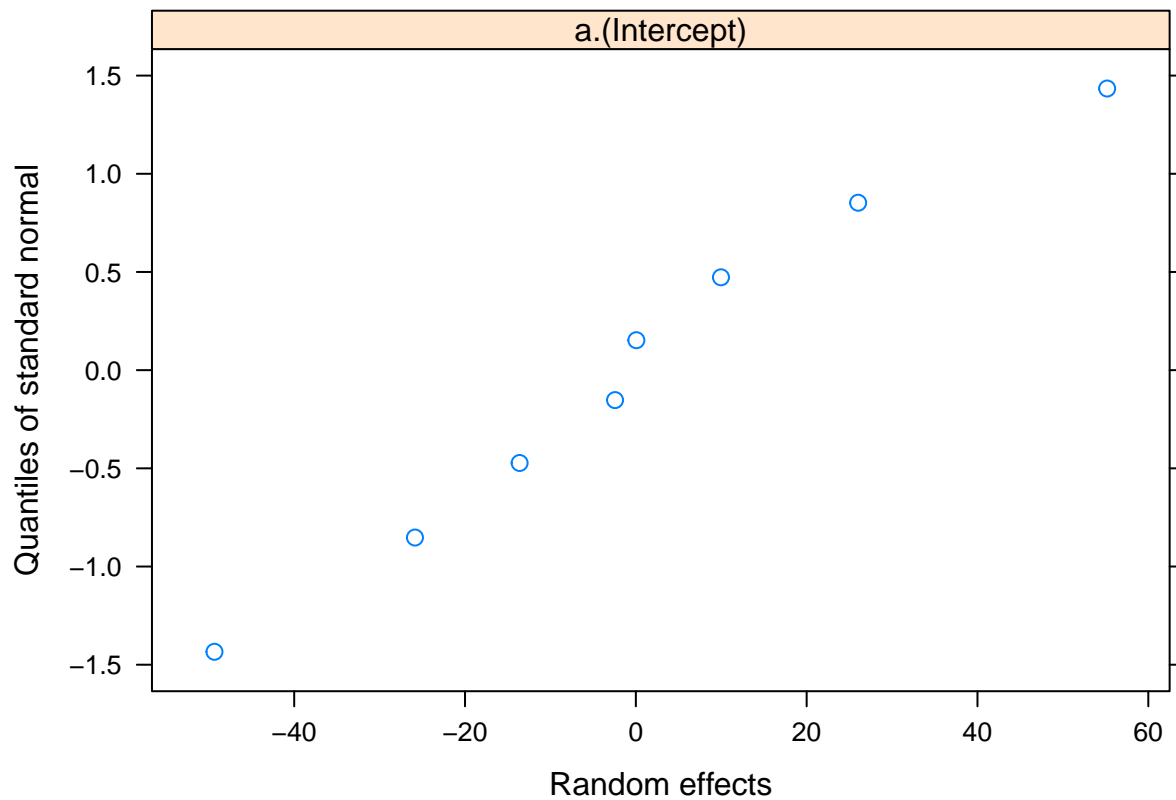
```
plot(modr, resid(.) ~ Speed)
```



```
# what about a q-q plot of the residuals  
# - not perfect: looks like we aren't doing such a good job in the tails?  
qqnorm(modr, abline=c(0,1))
```



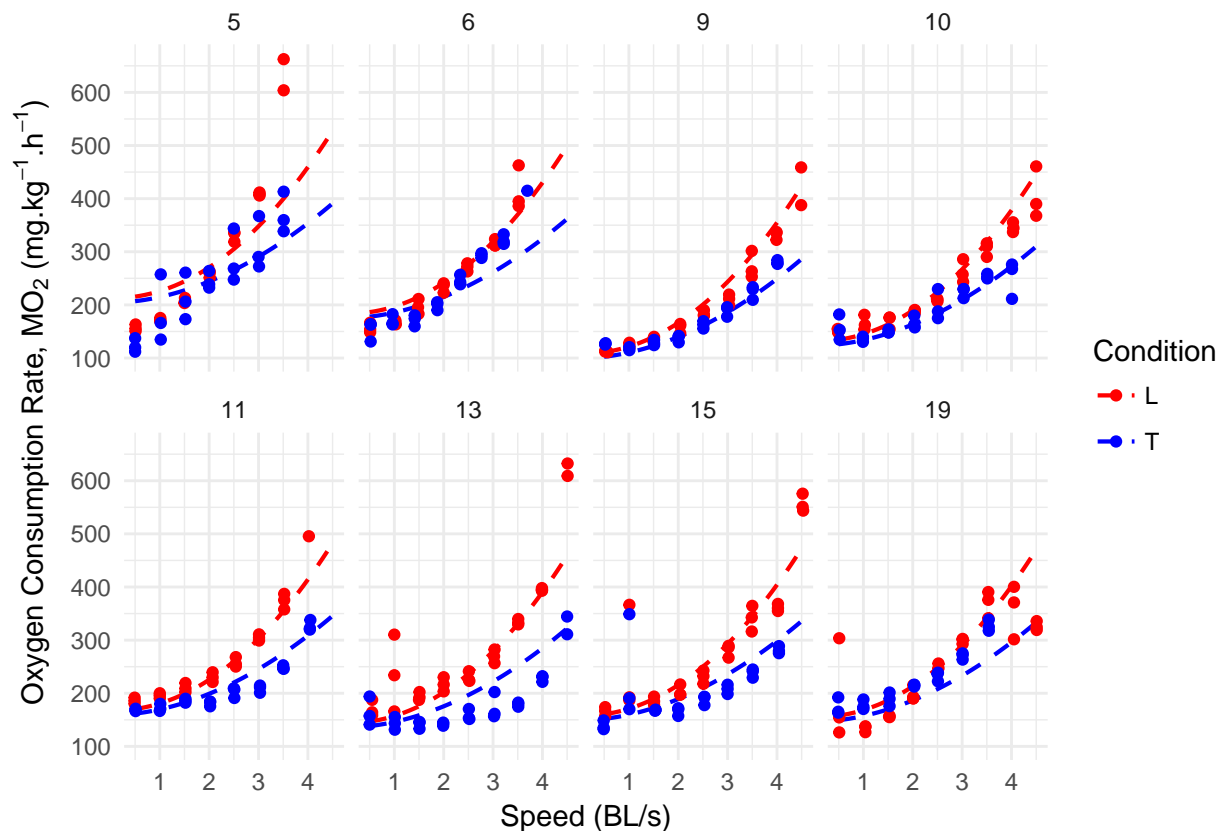
```
# q-q of the random effects - they should be normal
qqnorm(modr, ~ranef(.))
```



```
## 4. prediction plots
library(ggplot2)

# plot predictions per Fish
# make prediction grid, then predict
preddat <- expand.grid(Speed = seq(0.5, 4.5, by=0.1),
                      Flow = c("L", "T"), # L = LTF, T = HTF
                      Fish = unique(fish$Fish))
plotty <- predict(modr, preddat, level=1)
plotty <- cbind(preddat, V02minBac=plotty)

p <- ggplot(plotty, aes(x=Speed, y=V02minBac, group=Flow, colour=Flow)) +
  geom_line(size=0.75, linetype=2) +
  geom_point(data=fish) +
  facet_wrap(~Fish, nrow=2) +
  theme_minimal() +
  # scale_colour_brewer(type="qual")
  scale_color_manual(values=c('red', 'blue')) +
  scale_shape_manual(values=c(17, 16))+
  labs(x="Speed (BL/s)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^{-1}*h^{-1}*")"),
  print(p)
```



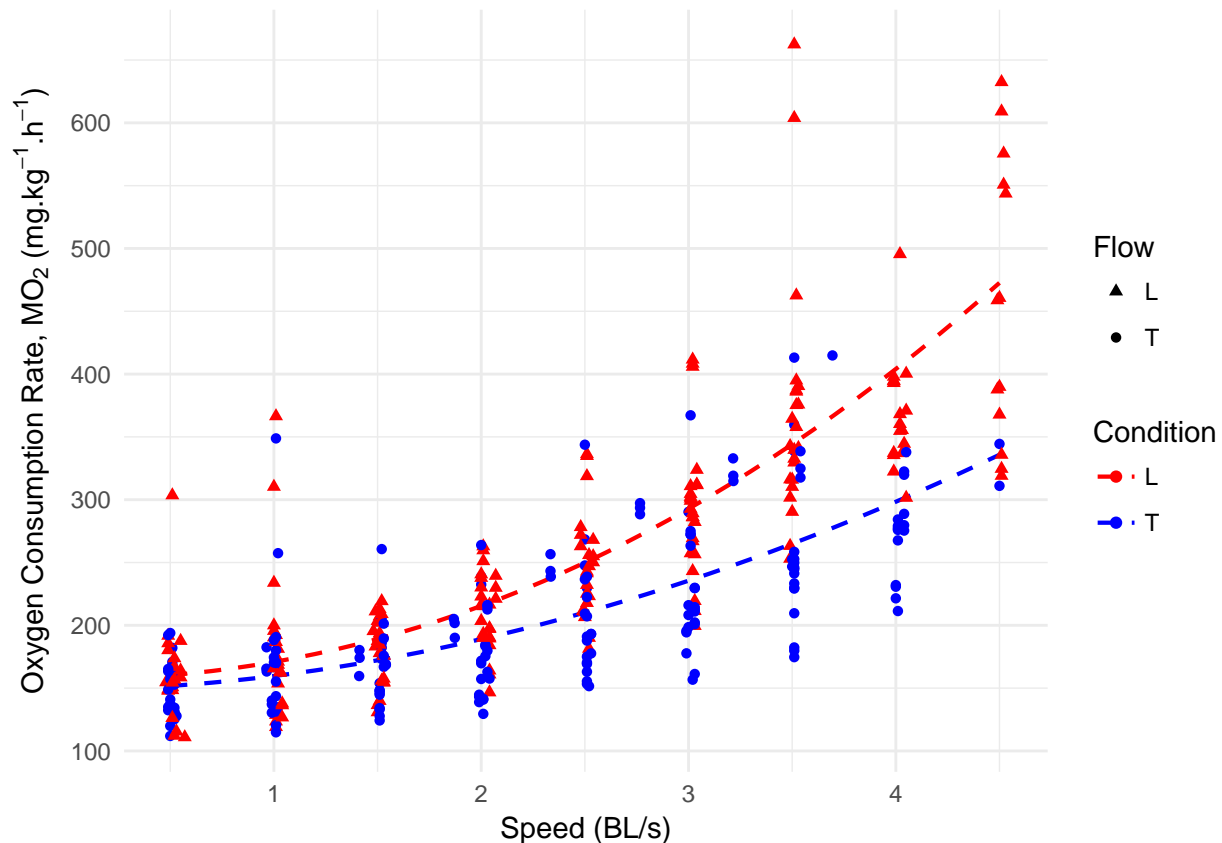
```
# plot predictions averaged over Fish
# make prediction grid, then predict
preddat <- expand.grid(Speed = seq(0.5, 4.5, by=0.1),
                      Flow = c("L", "T"))
plotty <- predict(modr, preddat, level=0)
```

```

plotty <- cbind(predat, V02minBac=plotty)

p <- ggplot(plotty, aes(x=Speed, y=V02minBac, group=Flow, colour=Flow)) +
  geom_line(size=0.75, linetype=2) +
  geom_point(data=fish, size=1.5, aes(shape=Flow)) +
  theme_minimal() +
  scale_colour_manual(values=c("red", "blue")) +
  scale_shape_manual(values=c(17, 16))+
  labs(x="Speed (BL/s)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^{-1}*h^{-1}*")"),
  print(p)

```



```

# Supplemental figure of true speed instead of Body lengths per second to compare with
# PIV measurements that are in real instead of relative units
file <- system.file("V02", "V02.xlsx", package = "xlsx")
data <- read_excel("V02.xlsx", 7) # includes body lengths of fish

data$Cond <- as.factor(data$Cond)

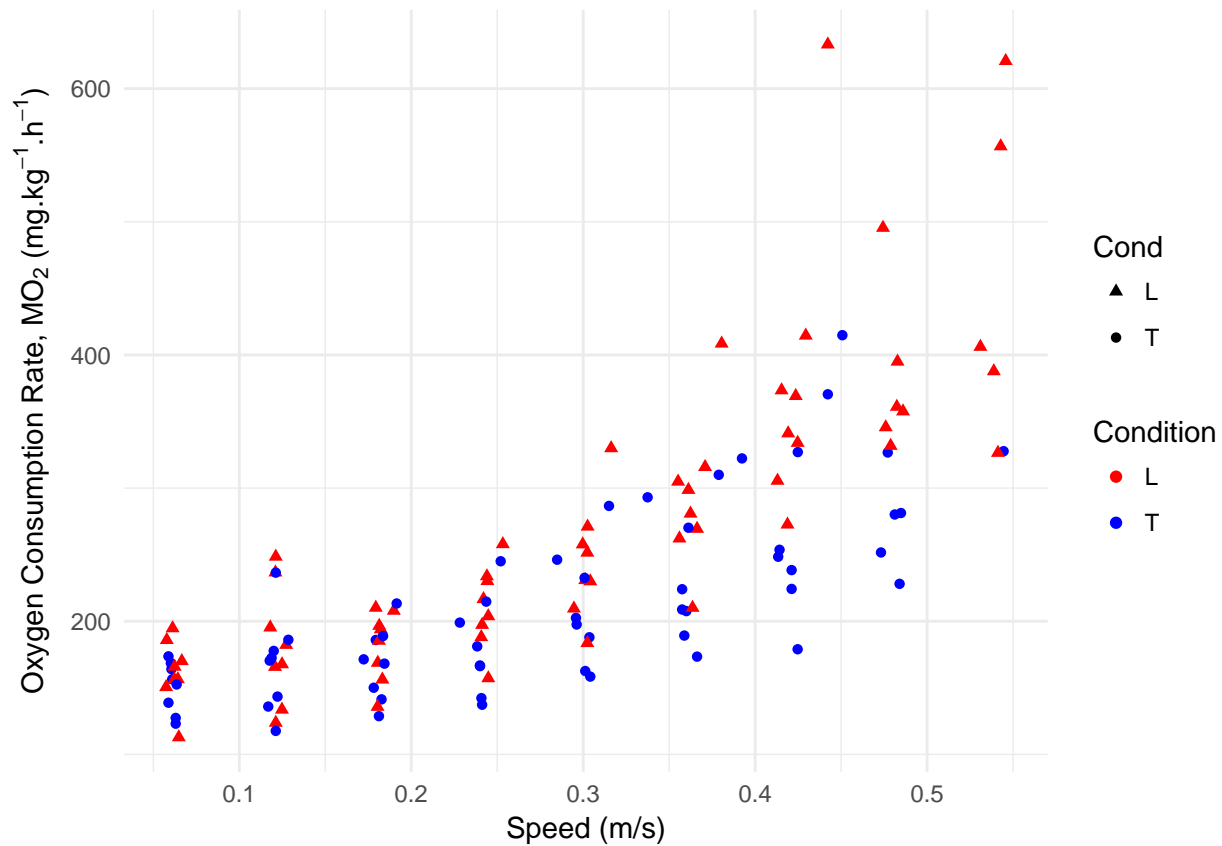
p <- ggplot(data, aes(x=Truespeed, y=MeanV02minBac, group=Cond, colour=Cond)) +
  # geom_line(size=0.75, linetype=2) +
  geom_point(data=data, size=1.5, aes(shape=Cond)) +
  theme_minimal() +
  scale_colour_manual(values=c("red", "blue")) +
  scale_shape_manual(values=c(17, 16))+
  labs(x="Speed (m/s)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^{-1}*h^{-1}*")"),

# png("FigureS5.png", width = 6, height = 4, units = 'in', res = 300)

```



```
print(p) # Make plot
```



```
# dev.off()
```

To assess how  $\text{MO}_2$  changed with fin beat frequency, we fitted a generalized additive model (Wood, 2017) to the  $\text{MO}_2$ , using pectoral or caudal fin beat (Hz) as separate explanatory variables. We used factor-smooth interactions (Baayen et al., 2016) to fit two levels of a smoothed function of measured beat frequency, one for each flow condition. Such terms fit the base level of the factor as a smooth, then model deviations from that smooth for the other level, thus information is shared between the models while allowing for a flexible relationship that makes no assumptions about functional form. Fish ID was included as a random effect. Models were fitted by restricted maximum likelihood.

```
## Load in and take first look at data
```

```
hz <- read.csv("FishV02Hz.csv")
```

```
# make codes for the speeds
```

```
hz$speedcode <- "H"
```

```
hz$speedcode[hz$speed == 0.5] <- "L"
```

```
hz$speedcode[hz$speed == 1.5] <- "M"
```

```
# speeds set at 0.5, 1.5 and then the max speed achieved for both LTF and HTF
```

```
library(ggplot2)
```

```
# quick visualisation
```

```
p <- ggplot(hz) +
```

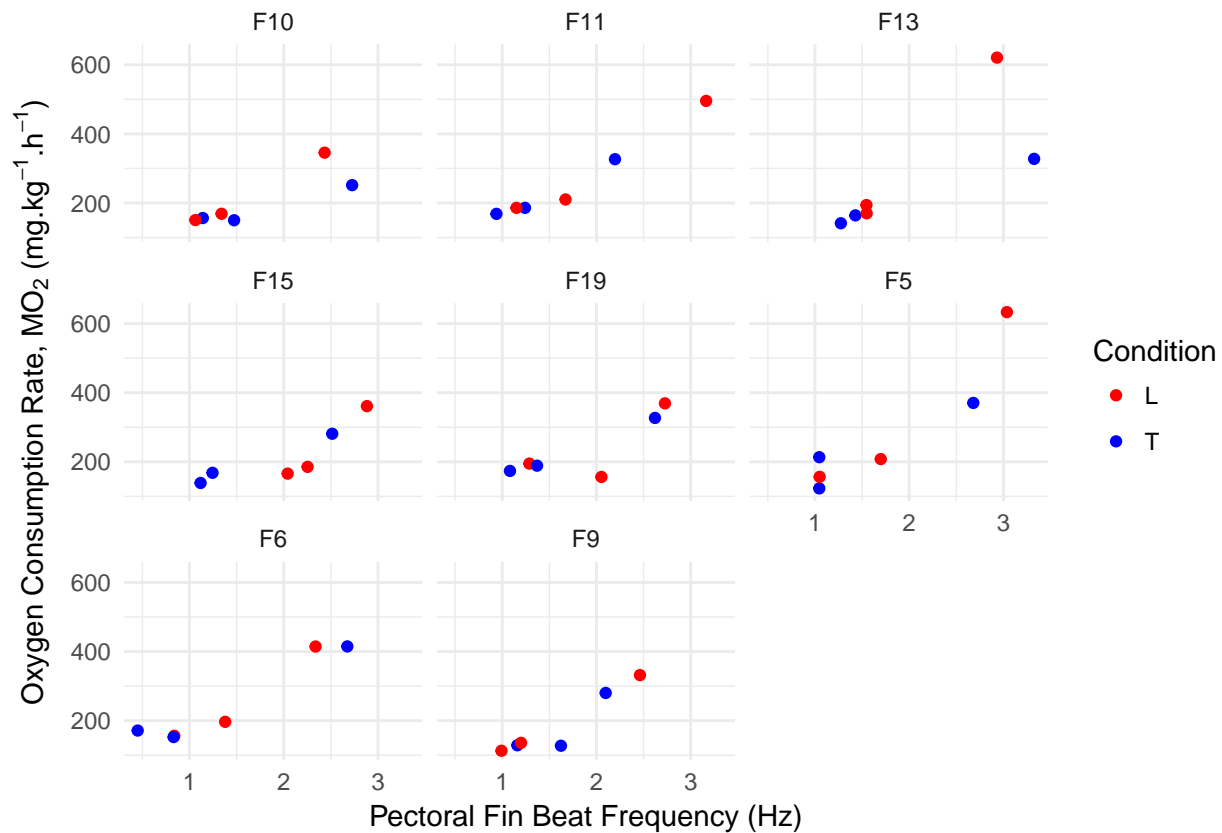
```
  #geom_point(aes(x=speed, y=V02minBac, colour=cond)) +
```

```
  geom_point(aes(x=PecHz, y=V02minBac, colour=cond)) +
```

```

theme_minimal() +
scale_shape_manual(values=c(16, 17)) +
scale_color_manual(values=c('red', 'blue')) +
labs(x="Pectoral Fin Beat Frequency (Hz)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg
facet_wrap(~Fish)
print(p)

```



```

## Fit a GAM
library(mgcv)

```

```
## Warning: package 'mgcv' was built under R version 3.4.2
```

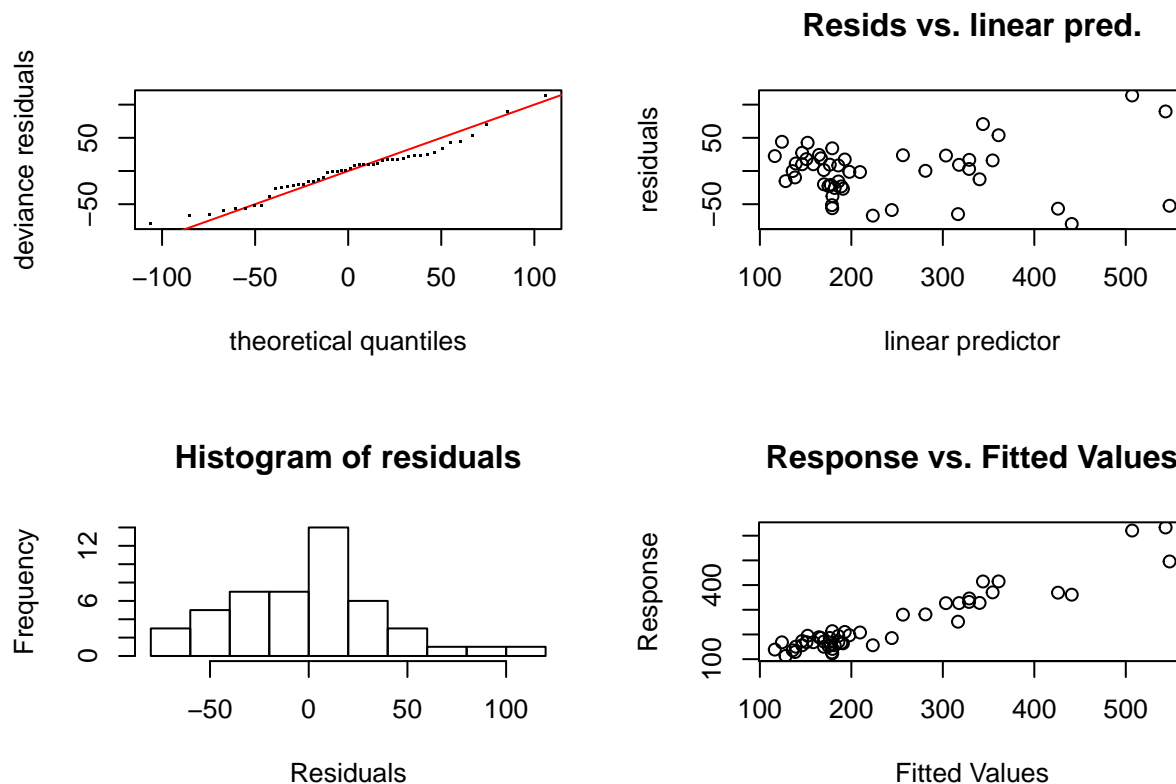
```
## This is mgcv 1.8-22. For overview type 'help("mgcv-package")'.
```

```

# this is fitting a model that says:
# O2 varies as a function of pectoral hz, but we should estimate a different
# curve for each condition. We also think that fish acts like a random effect
b_pec <- gam(VO2minBac~s(PecHz, cond, bs="fs") + s(Fish, bs="re"),
            data=haz, method="REML")

# check this model
gam.check(b_pec)

```



```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.0009921423,0.0009264311]
## (score 263.7614 & scale 2105.723).
## Hessian positive definite, eigenvalue range [1.314663e-05,24.26885].
## Model rank = 29 / 29
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##           k'   edf k-index p-value
## s(PecHz,cond) 20.00 6.65  1.03  0.52
## s(Fish)       8.00 4.89   NA    NA
```

*# Little heteroskedasticity (top right). Looks like the model predicts well (bottom  
# right). Residuals look normal enough (left side). Text output shows that we have  
# allowed for sufficient flexibility in our model.*

```
summary(b_pec)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## V02minBac ~ s(PecHz, cond, bs = "fs") + s(Fish, bs = "re")
##
## Parametric coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  236.07      70.81   3.334  0.00201 **
```

```

## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df      F p-value
## s(PecHz,cond) 6.655     19 16.349 < 2e-16 ***
## s(Fish)       4.885      7  2.183 0.00858 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.862   Deviance explained = 89.6%
## -REML = 263.76   Scale est. = 2105.7      n = 48

# v. high % deviance explained, will continue to include the fish random effect
# to explicitly include between-fish variability

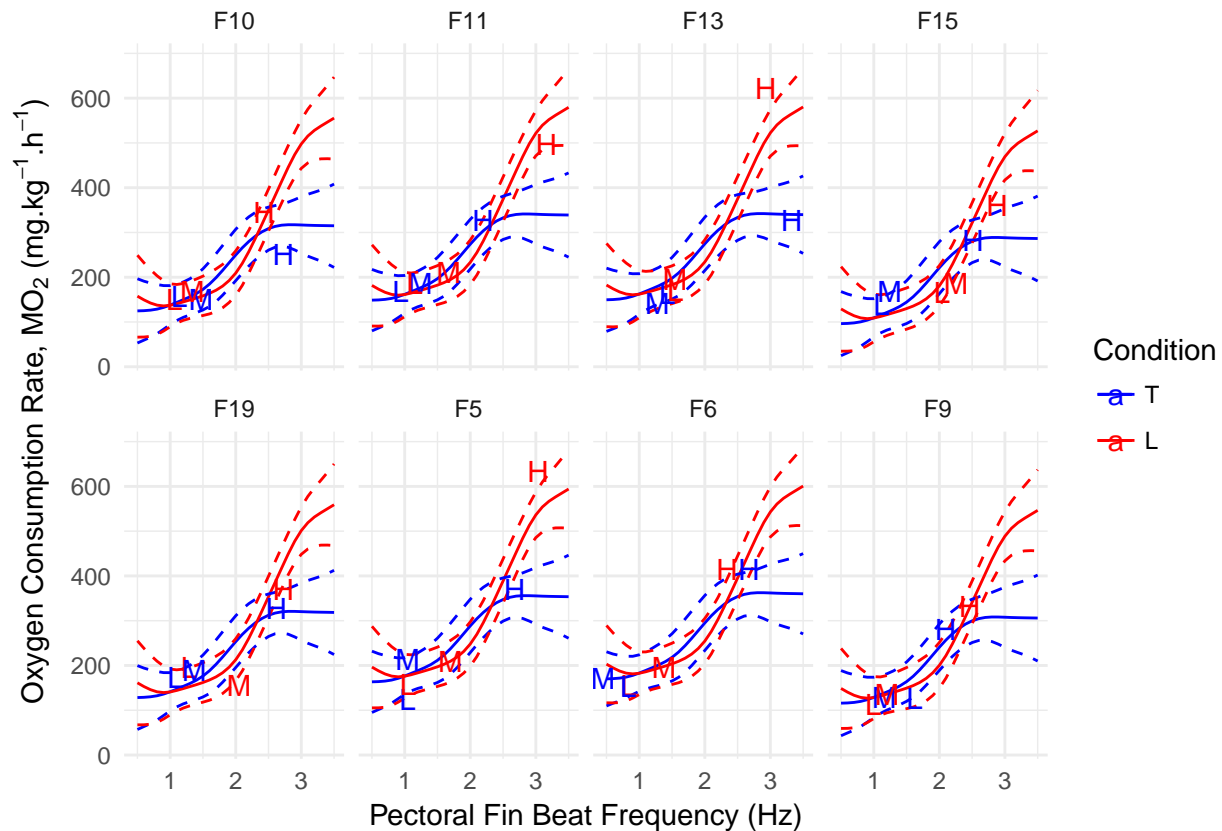
# now to make predictions, make a grid of values with all the pectoral hz we need
# plus values for the condition and fish
preddat <- expand.grid(PecHz = seq(0.5, 3.5, by=0.1),
                      cond  = c("T", "L"),
                      Fish  = unique(hz$Fish))

# make predictions, also predict the standard error
pr <- predict(b_pec, preddat, type="response", se=TRUE)
preddat$V02minBac <- pr$fit
# generate the CIs
preddat$upper <- pr$fit + 2*pr$se.fit
preddat$lower <- pr$fit - 2*pr$se.fit

# what does that look like?
p <- ggplot(hz) +
  geom_line(aes(x=PecHz, y=V02minBac, colour=cond, group=cond), data=preddat) +
  geom_line(aes(x=PecHz, y=upper, colour=cond, group=cond), linetype=2, data=preddat) +
  geom_line(aes(x=PecHz, y=lower, colour=cond, group=cond), linetype=2, data=preddat) +
  geom_text(aes(x=PecHz, y=V02minBac, colour=cond, label=speedcode)) +
  # scale_colour_brewer(type="qual") +
  theme_minimal() +
  scale_color_manual(values=c('blue', 'red')) +
  labs(x="Pectoral Fin Beat Frequency (Hz)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.k
  facet_wrap(~Fish, nrow=2)

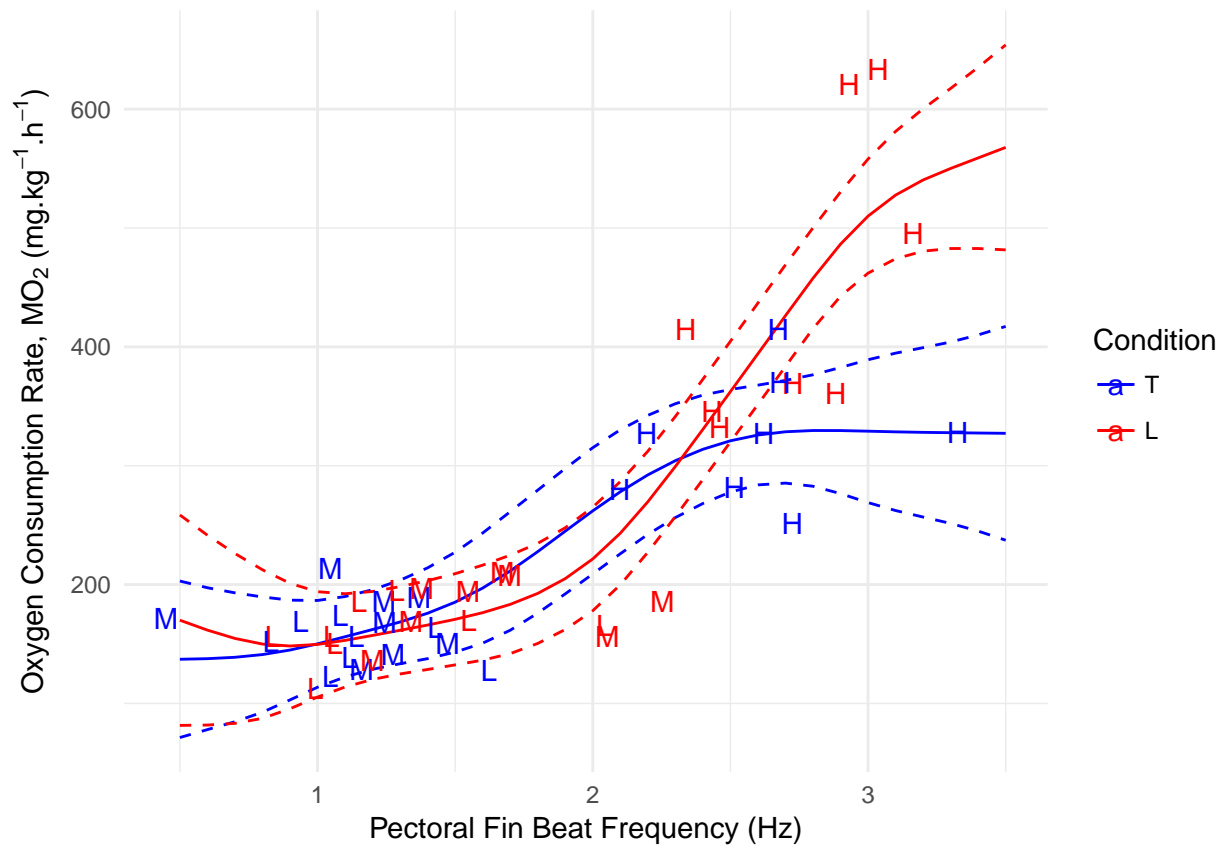
print(p)

```



```
# ignore fish and predict while integrating out the fish random effect
preddat <- expand.grid(PecHz = seq(0.5, 3.5, by=0.1),
                      cond = c("T", "L"), Fish="F5")
pr <- predict(b_pec, preddat, type="response", se=TRUE, exclude="s(Fish)")
preddat$V02minBac <- pr$fit
preddat$upper <- pr$fit + 2*pr$se.fit
preddat$lower <- pr$fit - 2*pr$se.fit

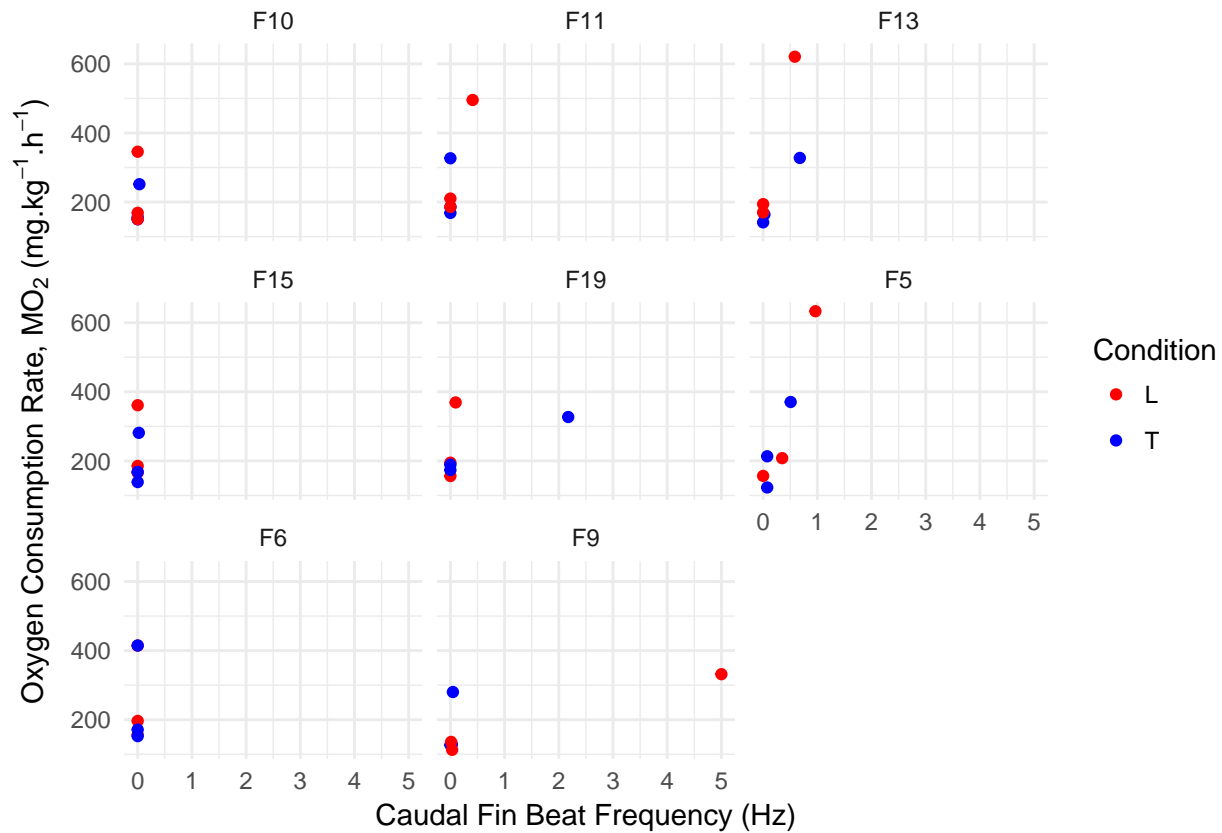
# what does that look like?
p <- ggplot(hz) +
  geom_line(aes(x=PecHz, y=V02minBac, colour=cond, group=cond), data=preddat) +
  geom_line(aes(x=PecHz, y=upper, colour=cond, group=cond), linetype=2, data=preddat) +
  geom_line(aes(x=PecHz, y=lower, colour=cond, group=cond), linetype=2, data=preddat) +
  geom_text(aes(x=PecHz, y=V02minBac, colour=cond, label=speedcode)) +
  # scale_colour_brewer(type="qual") +
  # scale_shape_manual(values=c(16, 17))+
  scale_color_manual(values=c('blue', 'red'))+
  theme_minimal() +
  labs(x="Pectoral Fin Beat Frequency (Hz)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg
print(p)
```



We were interested in testing the relationship between oxygen consumption and caudal fin use or frequency; however, the caudal fin was rarely used and was 0 in 65% of the experimental conditions. Model formulation and figures of the raw data are included below to illustrate the limitations of the data and the observed caudal fin use patterns.

```
# quick visualisation
p <- ggplot(hz) +
  geom_point(aes(x=CaudHz, y=VO2minBac, colour=cond)) +
  theme_minimal() +
  scale_shape_manual(values=c(16, 17)) +
  scale_color_manual(values=c('red', 'blue')) +
  labs(x="Caudal Fin Beat Frequency (Hz)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^-1.h^-1))
  facet_wrap(~Fish)

print(p)
```



```
# try the same kind of model as above for the caudal fin frequency
# need to reduce k (smooth complexity) as we don't have many unique values
b_caud <- gam(VO2minBac~s(CaudHz, cond, bs="fs", k=5) + s(Fish, bs="re"),
              data=haz, method="REML")
```

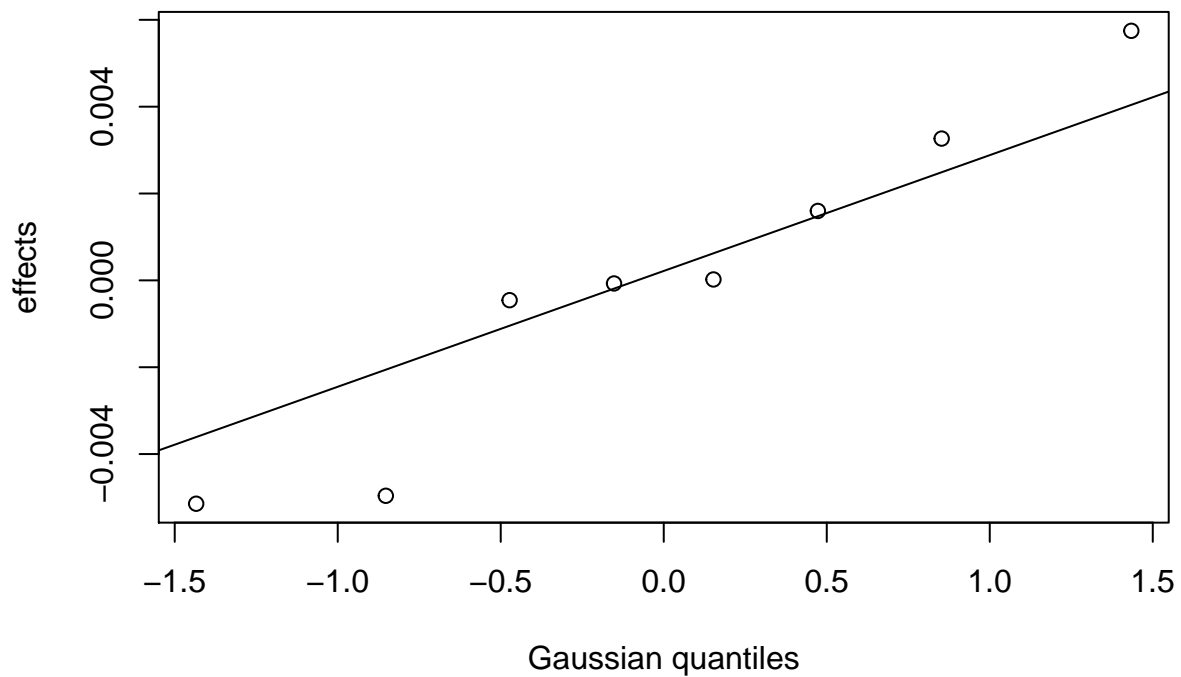
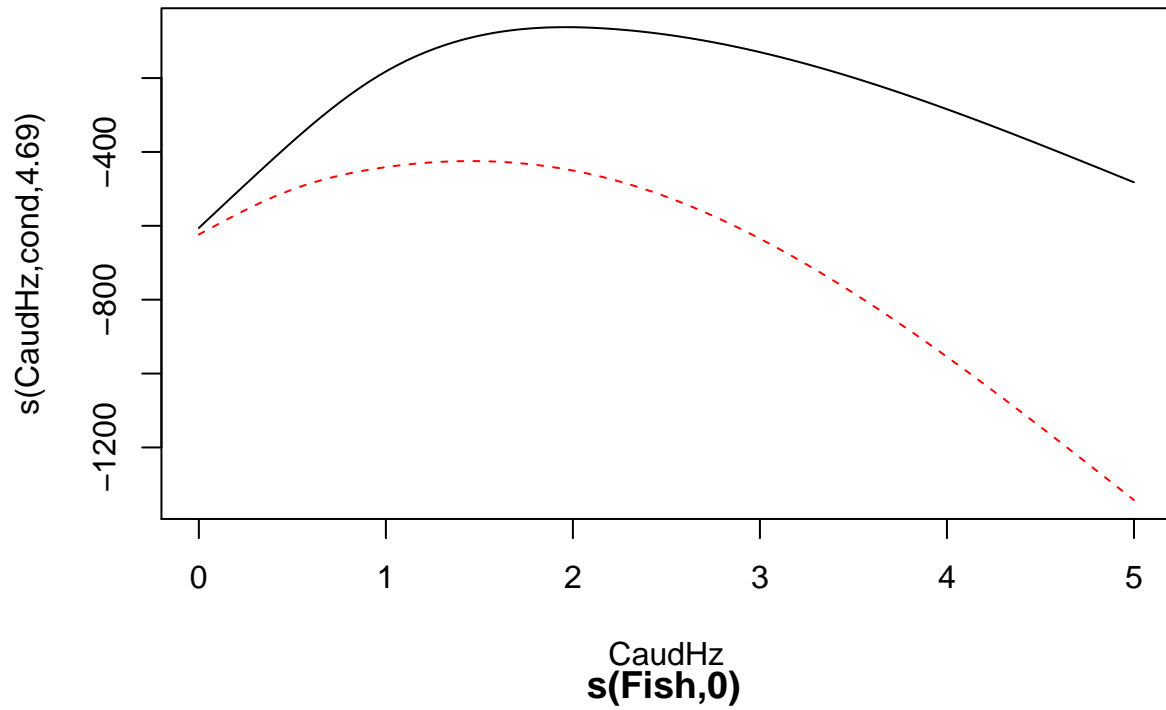
```
# model check
summary(b_caud)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## VO2minBac ~ s(CaudHz, cond, bs = "fs", k = 5) + s(Fish, bs = "re")
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   816.9      245.4    3.329  0.00181 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df      F p-value
## s(CaudHz,cond) 4.6886317     9 5.759 1.73e-07 ***
## s(Fish)         0.0007536     7 0.000   0.501
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## R-sq.(adj) = 0.524   Deviance explained = 57.2%
## -REML = 286.99   Scale est. = 7268       n = 48
```

*# Much less deviance explained than before*

```
plot(b_caud)
```



*# Do we believe that O2 is n-shaped in caudal hz? Unlikely. This is not a suitable model.*



```

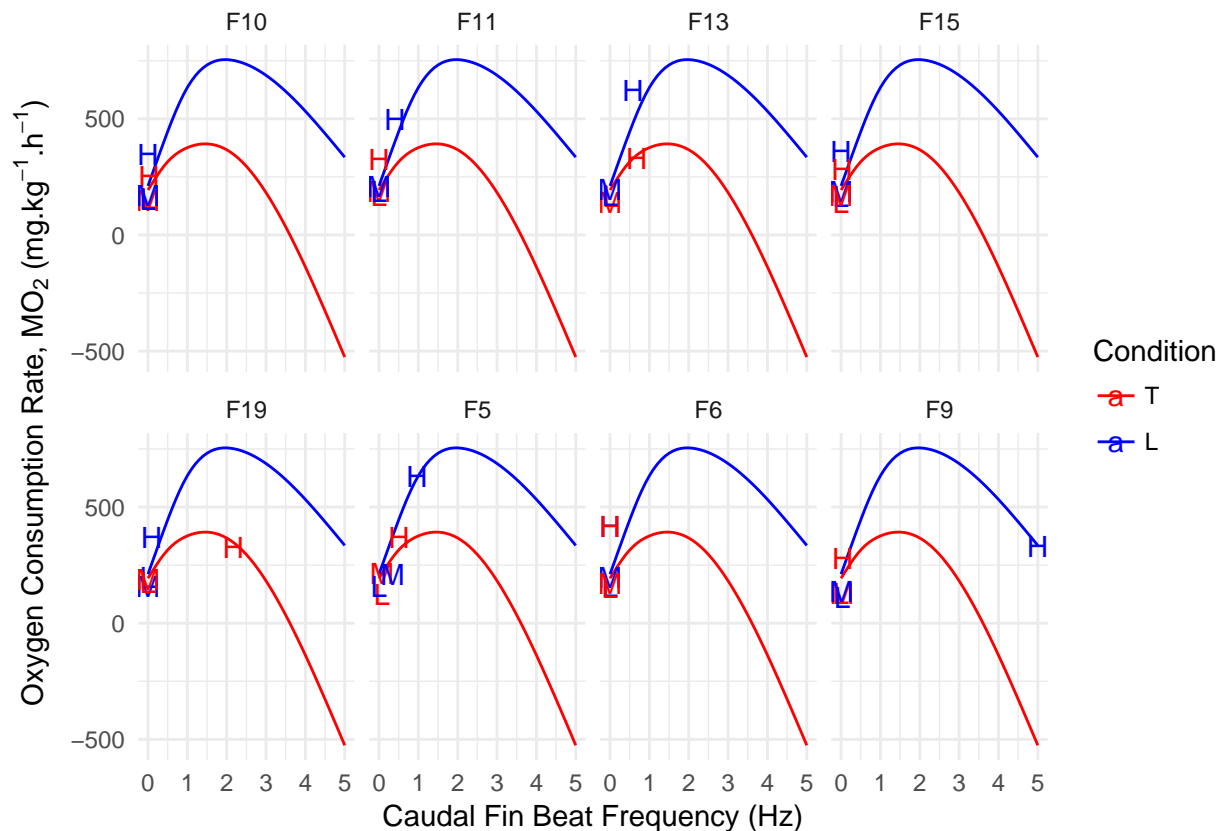
preddat <- expand.grid(CaudHz = seq(0, 5, by=0.1),
                      cond = c("T", "L"),
                      Fish = unique(hz$Fish))

preddat$V02minBac <- predict(b_caud, preddat, type="response")

p <- ggplot(hz) +
  geom_line(aes(x=CaudHz, y=V02minBac, colour=cond, group=cond), data=preddat) +
  geom_text(aes(x=CaudHz, y=V02minBac, colour=cond, label=speedcode)) +
  # scale_colour_brewer(type="qual") +
  theme_minimal() +
  labs(x="Caudal Hz", colour="Condition") +
  scale_shape_manual(values=c(16, 17)) +
  scale_color_manual(values=c('red', 'blue')) +
  labs(x="Caudal Fin Beat Frequency (Hz)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^-1)")) +
  facet_wrap(~Fish, nrow=2)

print(p)

```



```
## what about caudal use? Will have the same data limitation (when Hz = 0, use = 0)
```

```

# quick visualisation
p <- ggplot(hz) +
  geom_point(aes(x=Cauduse, y=V02minBac, colour=cond)) +
  theme_minimal() +
  scale_shape_manual(values=c(16, 17)) +

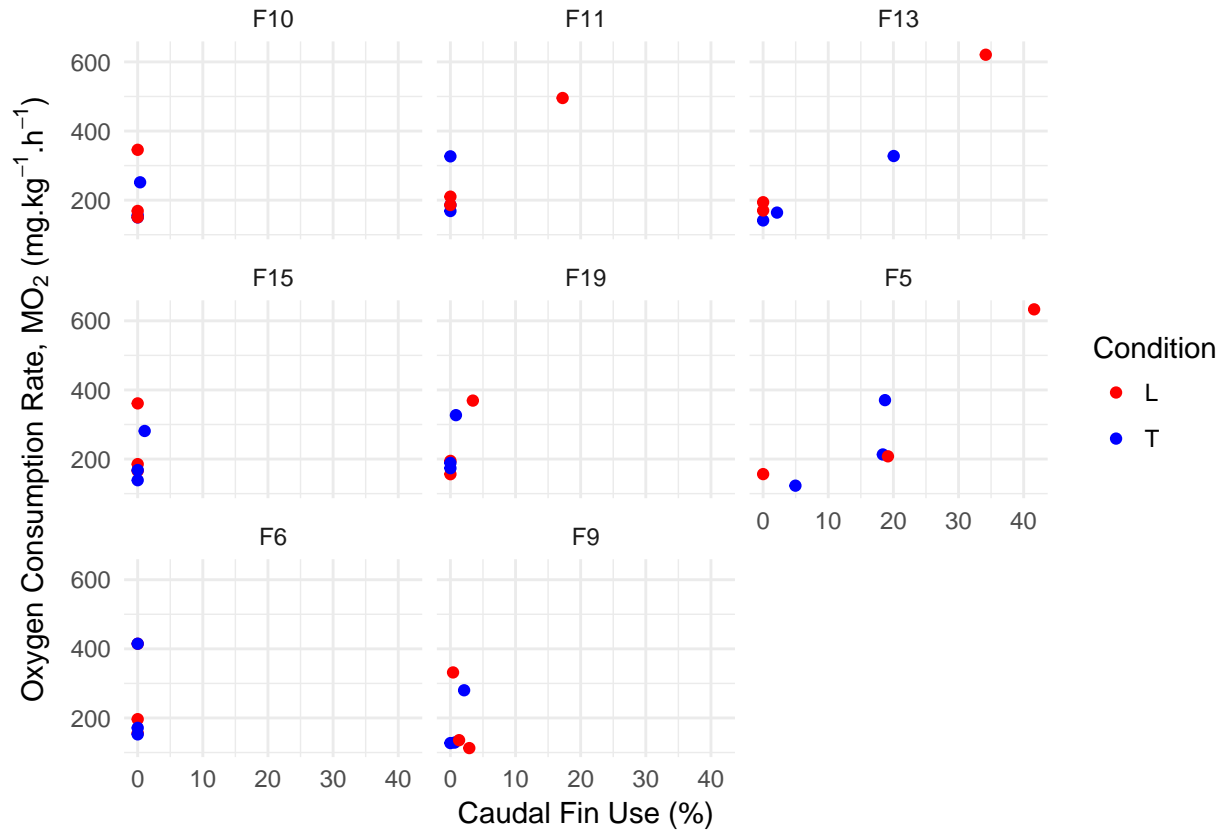
```

```

scale_color_manual(values=c('red', 'blue')) +
labs(x="Caudal Fin Use (%)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^{-1}*h^{-1})*")) +
facet_wrap(~Fish)

print(p)

```



```

# try the same kind of model as above for the caudal fin use
# need to reduce k (smooth complexity) as we don't have many unique values
b_caud <- gam(VO2minBac~s(Cauduse, cond, bs="fs", k=5) + s(Fish, bs="re"),
              data=haz, method="REML")

```

```

# model check
summary(b_caud)

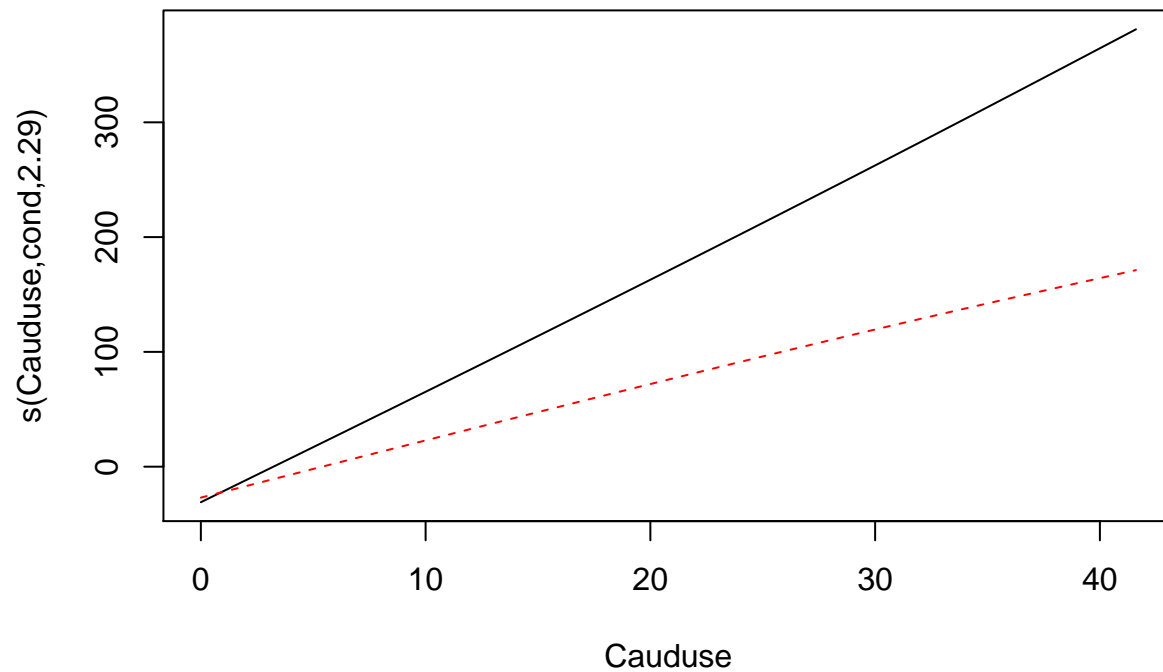
```

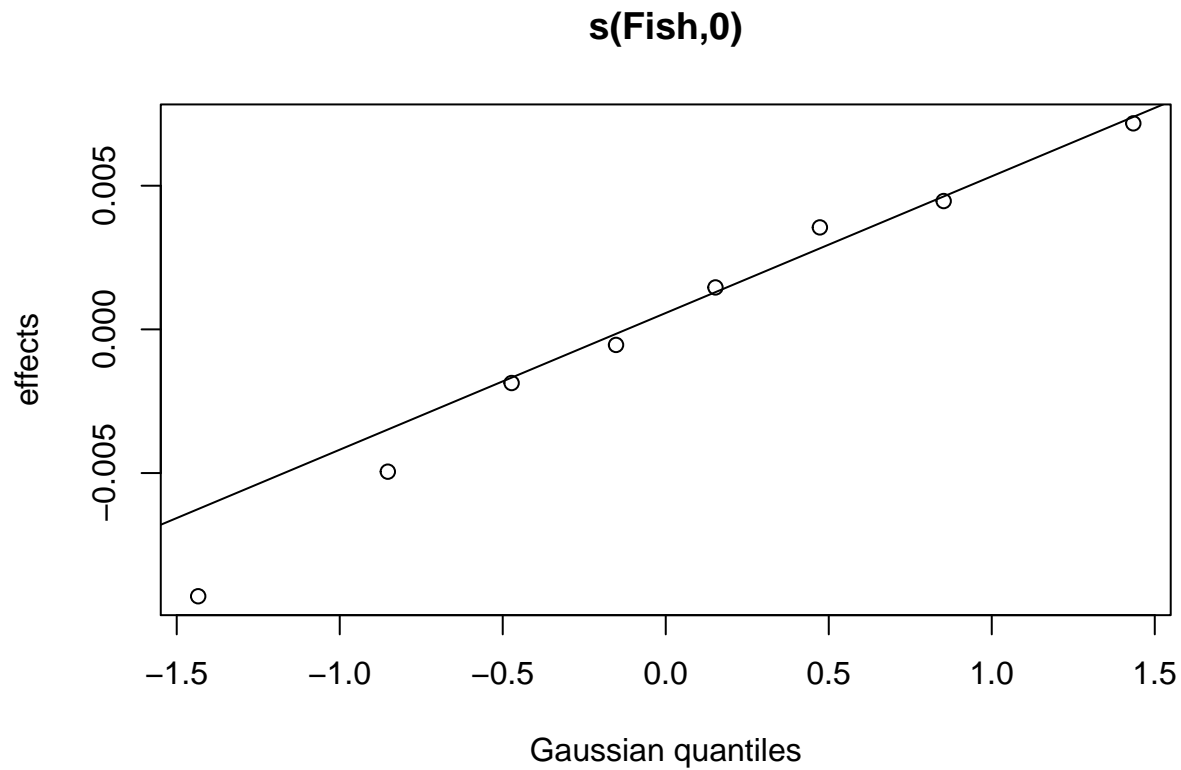
```

##
## Family: gaussian
## Link function: identity
##
## Formula:
## VO2minBac ~ s(Cauduse, cond, bs = "fs", k = 5) + s(Fish, bs = "re")
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   235.85      17.24   13.68  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##

```

```
## Approximate significance of smooth terms:
##               edf Ref.df    F  p-value
## s(Cauduse,cond) 2.288983     9 4.73 5.49e-08 ***
## s(Fish)         0.001065     7 0.00   0.532
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.475   Deviance explained = 50.1%
## -REML = 282.88   Scale est. = 8019.5    n = 48
# Again, less deviance explained than before
plot(b_caud)
```



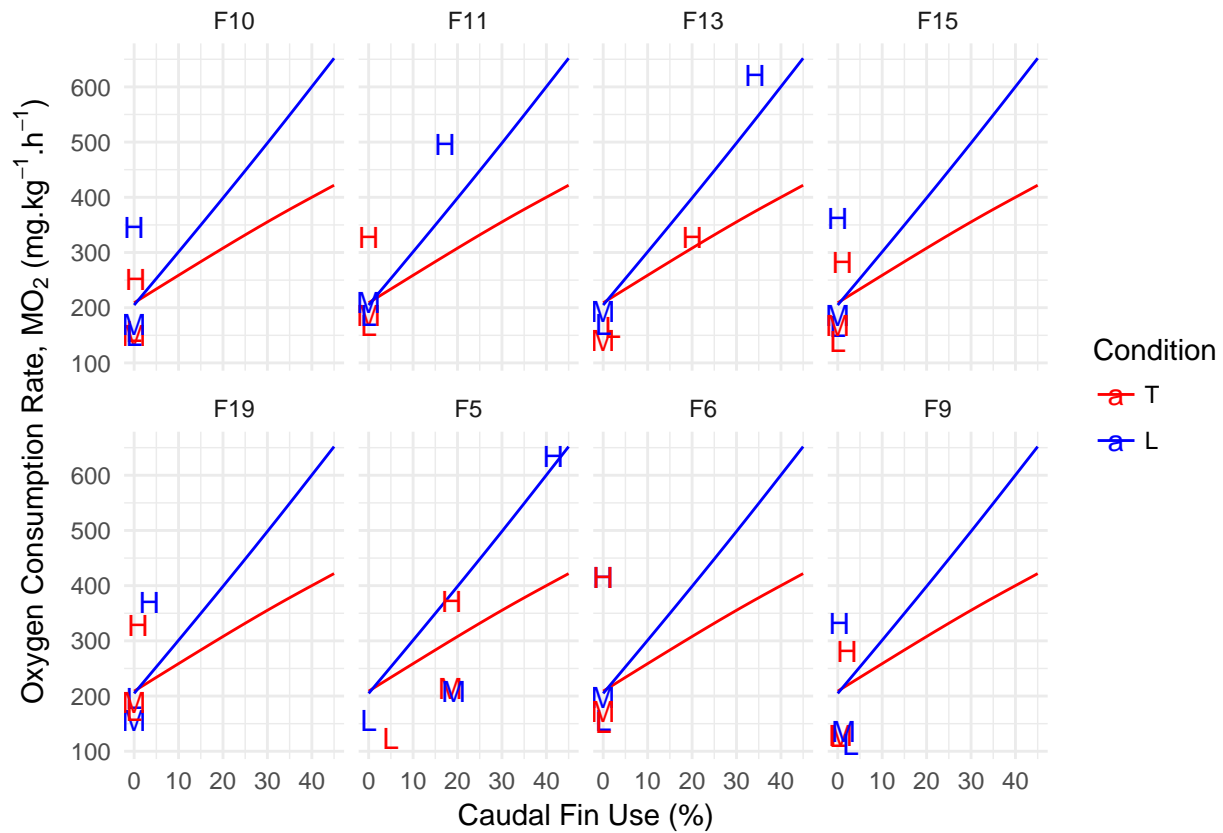


```
preddat <- expand.grid(Cauduse = seq(0, 45, by=1),
                      cond = c("T", "L"),
                      Fish = unique(hz$Fish))

preddat$VO2minBac <- predict(b_caud, preddat, type="response")

p <- ggplot(hz) +
  geom_line(aes(x=Cauduse, y=VO2minBac, colour=cond, group=cond), data=preddat) +
  geom_text(aes(x=Cauduse, y=VO2minBac, colour=cond, label=speedcode)) +
  # scale_colour_brewer(type="qual") +
  theme_minimal() +
  labs(x="Caudal Hz", colour="Condition") +
  scale_shape_manual(values=c(16, 17)) +
  scale_color_manual(values=c('red', 'blue')) +
  labs(x="Caudal Fin Use (%)", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg^{-1}*h^{-1})*
  facet_wrap(~Fish, nrow=2)

print(p)
```



```
## and, again for CV of pectoral fin beat frequency (following Roche et al. 2014)
```

```
# quick visualisation
```

```
p <- ggplot(hz) +
```

```
  geom_point(aes(x=CV.Pec, y=VO2minBac, colour=cond)) +
```

```
  theme_minimal() +
```

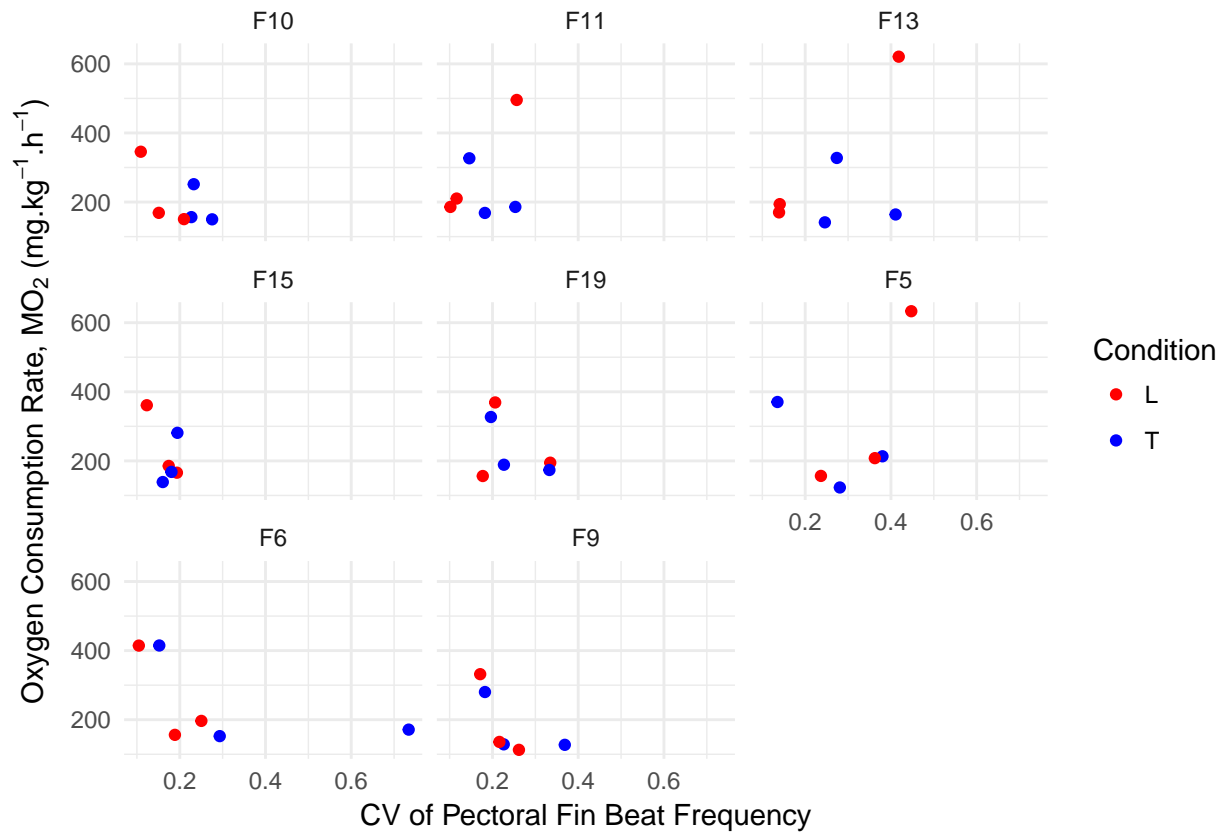
```
  scale_shape_manual(values=c(16, 17)) +
```

```
  scale_color_manual(values=c('red', 'blue')) +
```

```
  labs(x="CV of Pectoral Fin Beat Frequency", y = expression("Oxygen Consumption Rate, MO"[2]* " ("*mg.kg⁻¹.h⁻¹))
```

```
  facet_wrap(~Fish)
```

```
print(p)
```



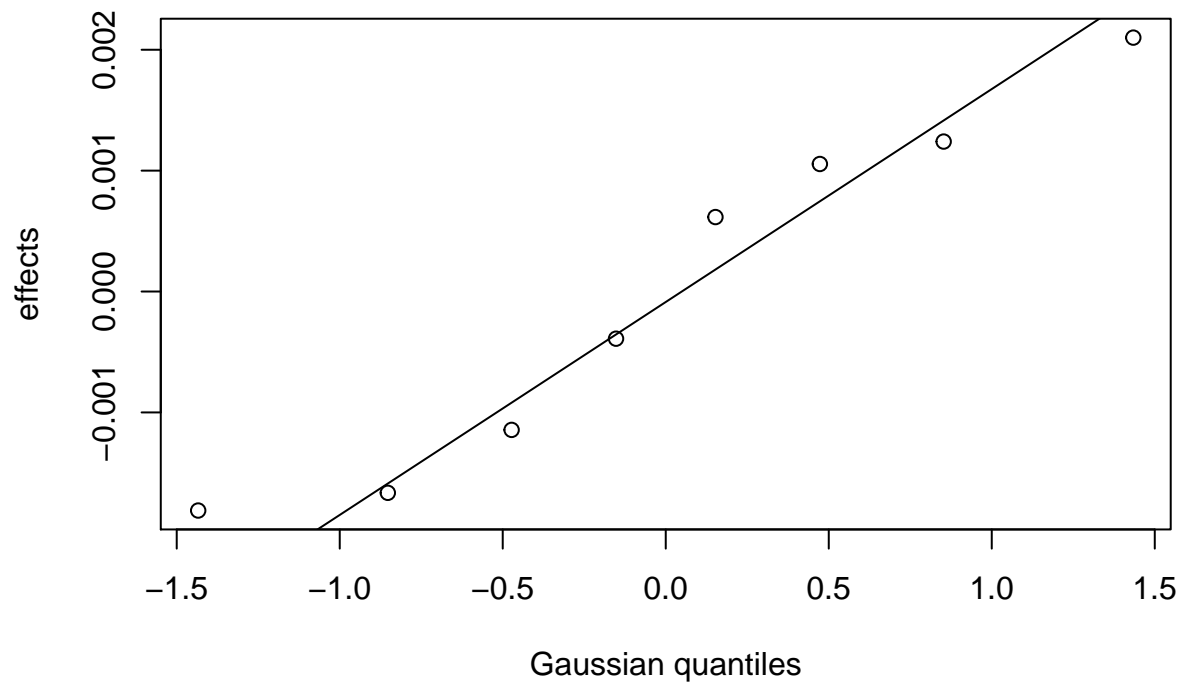
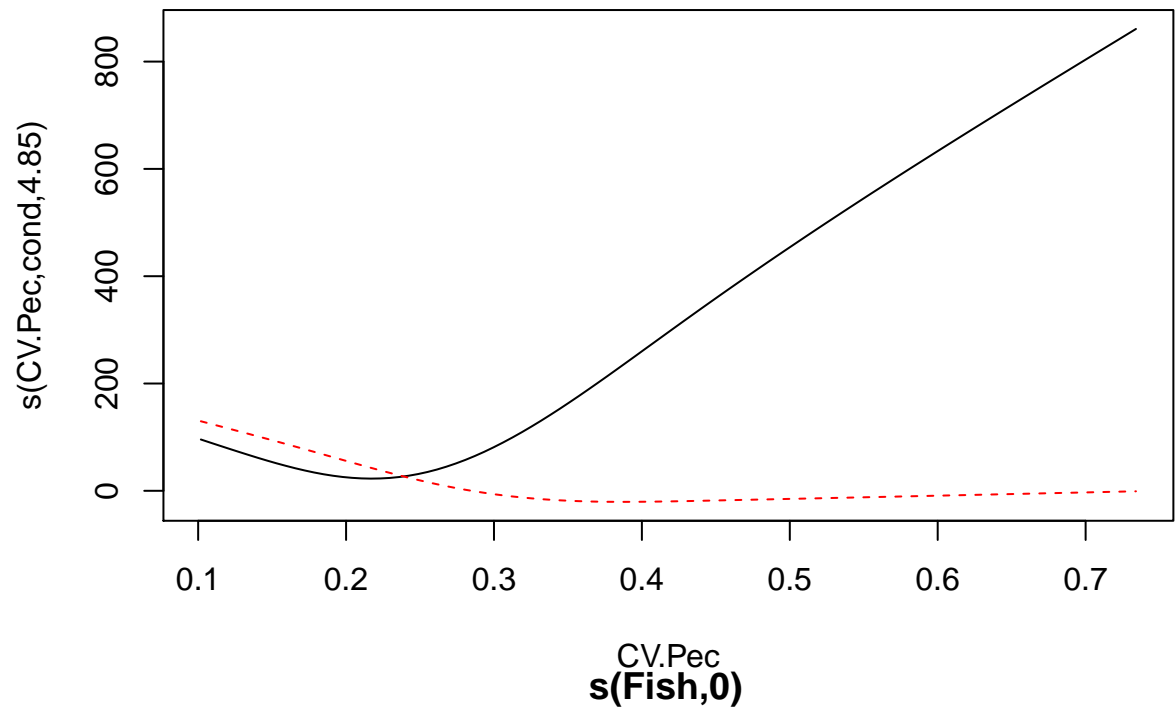
```
# Again, fitting the model
# need to reduce k (smooth complexity) as we don't have many unique values
b_cvpec <- gam(VO2minBac~s(CV.Pec, cond, bs="fs", k=5) + s(Fish, bs="re"),
              data=h2, method="REML")
```

```
# model check
summary(b_cvpec)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## VO2minBac ~ s(CV.Pec, cond, bs = "fs", k = 5) + s(Fish, bs = "re")
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  180.79      95.78    1.888   0.066 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df    F  p-value
## s(CV.Pec,cond) 4.8544608    9 3.29 0.000121 ***
## s(Fish)         0.0004511    7 0.00 0.947203
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## R-sq.(adj) = 0.387   Deviance explained = 45%
## -REML = 290.95   Scale est. = 9375.6   n = 48
```

```
plot(b_cvpec)
```

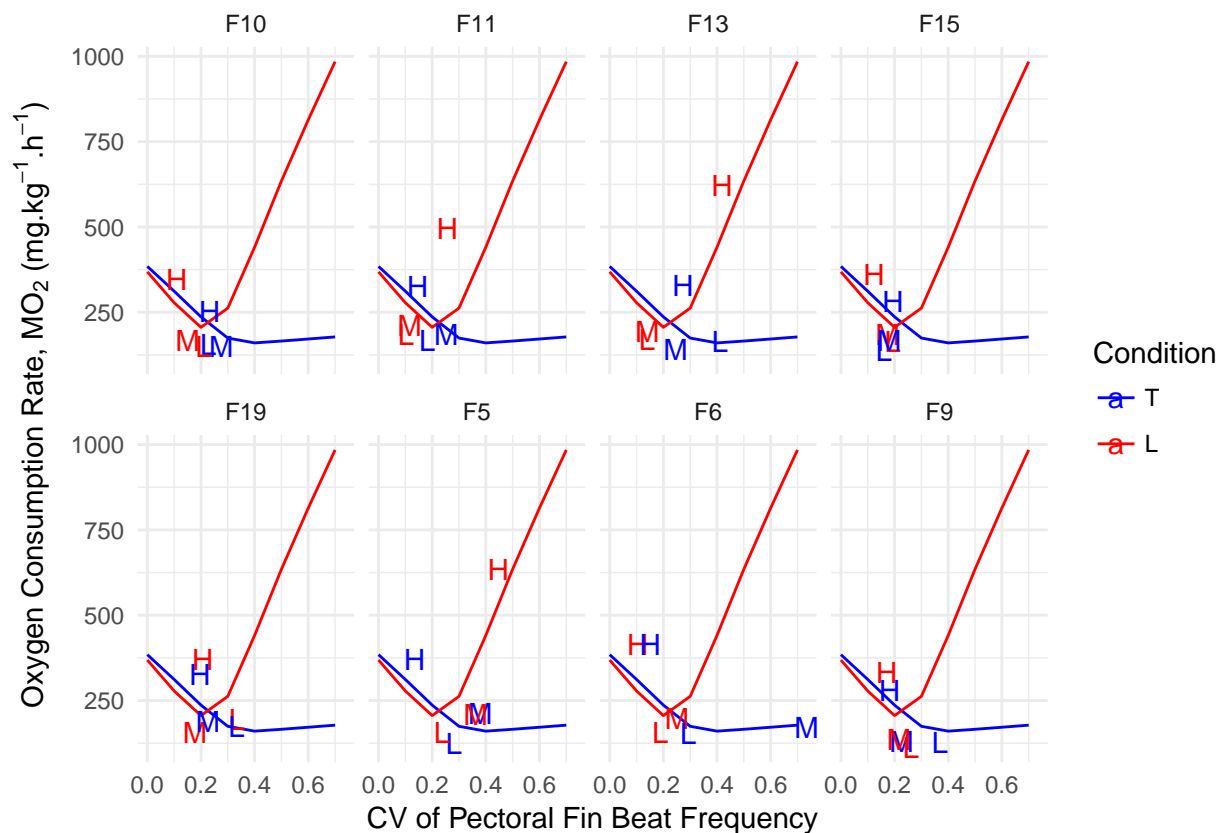


```
# generate predictive grid
preddat <- expand.grid(CV.Pec = seq(0, 0.75, by=0.1),
  cond = c("T", "L"),
  Fish = unique(hz$Fish))
```

```
preddat$V02minBac <- predict(b_cvpec, preddat, type="response")
```

```
p <- ggplot(hz) +
  geom_line(aes(x=CV.Pec, y=V02minBac, colour=cond, group=cond), data=preddat) +
  geom_text(aes(x=CV.Pec, y=V02minBac, colour=cond, label=speedcode)) +
  # scale_colour_brewer(type="qual") +
  theme_minimal() +
  # scale_shape_manual(values=c(16, 17)) +
  scale_color_manual(values=c('blue', 'red')) +
  labs(x="CV of Pectoral Fin Beat Frequency", y = expression("Oxygen Consumption Rate, MO"["2"]* " ("*mg.l"["-1"]*h"["-1"]))) +
  facet_wrap(~Fish, nrow=2)
```

```
print(p)
```



```
# plot all together by integrating out fish random effect
preddat <- expand.grid(CV.Pec = seq(0, 0.75, by=0.1),
  cond = c("T", "L"), Fish="F5")
pr <- predict(b_cvpec, preddat, type="response", se=TRUE, exclude="s(Fish)")
preddat$V02minBac <- pr$fit
preddat$upper <- pr$fit + 2*pr$se.fit
preddat$lower <- pr$fit - 2*pr$se.fit

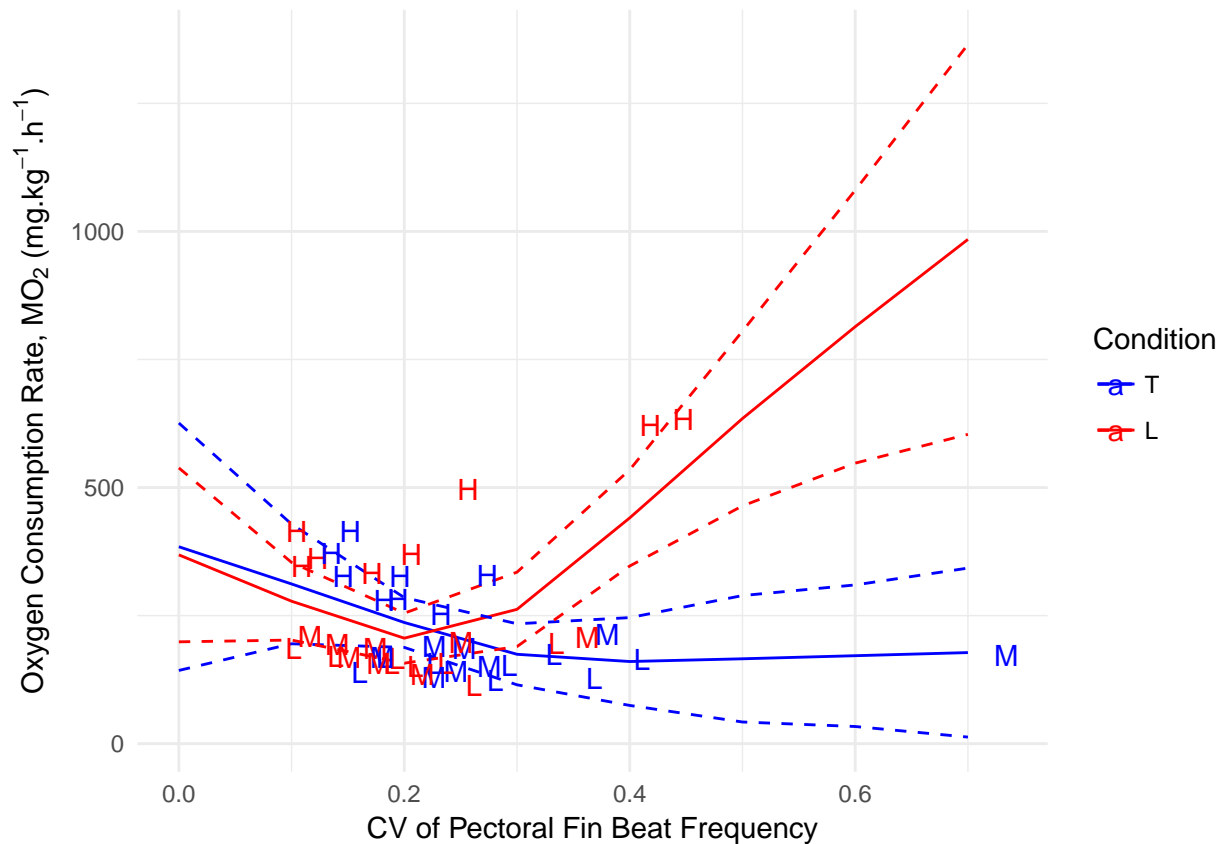
# what does that look like?
p <- ggplot(hz) +
  geom_line(aes(x=CV.Pec, y=V02minBac, colour=cond, group=cond), data=preddat) +
  geom_line(aes(x=CV.Pec, y=upper, colour=cond, group=cond), linetype=2, data=preddat) +
```



```

geom_line(aes(x=CV.Pec, y=lower, colour=cond, group=cond), linetype=2, data=preddat) +
geom_text(aes(x=CV.Pec, y=VO2minBac, colour=cond, label=speedcode)) +
# scale_colour_brewer(type="qual") +
# scale_shape_manual(values=c(16, 17))+
scale_color_manual(values=c('blue', 'red'))+
theme_minimal() +
labs(x="CV of Pectoral Fin Beat Frequency", y = expression("Oxygen Consumption Rate, MO" [2] * " ("*mg.l
print(p)

```



To investigate the position of fish in the tank, we compared the centroid locations of the snout of each fish through the 180s sample period, and standard deviation from that centroid position, in different speed\*flow combinations. This investigation along with the heatmaps (Figure 5) illustrate the overall positioning of individuals within the tank; however our interpretation is limited due to the number of response parameters compared to the number of animals tested and the factors to be considered.

```
library(R.matlab)
```

```
## R.matlab v3.6.1 (2016-10-19) successfully loaded. See ?R.matlab for help.
```

```
##
```

```
## Attaching package: 'R.matlab'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## getOption, isOpen
```

```
# data saved by Julie van der Hoop from MATLAB
```

```
move <- readMat("FHL_centroiddata_Dec2017.mat")
```

```

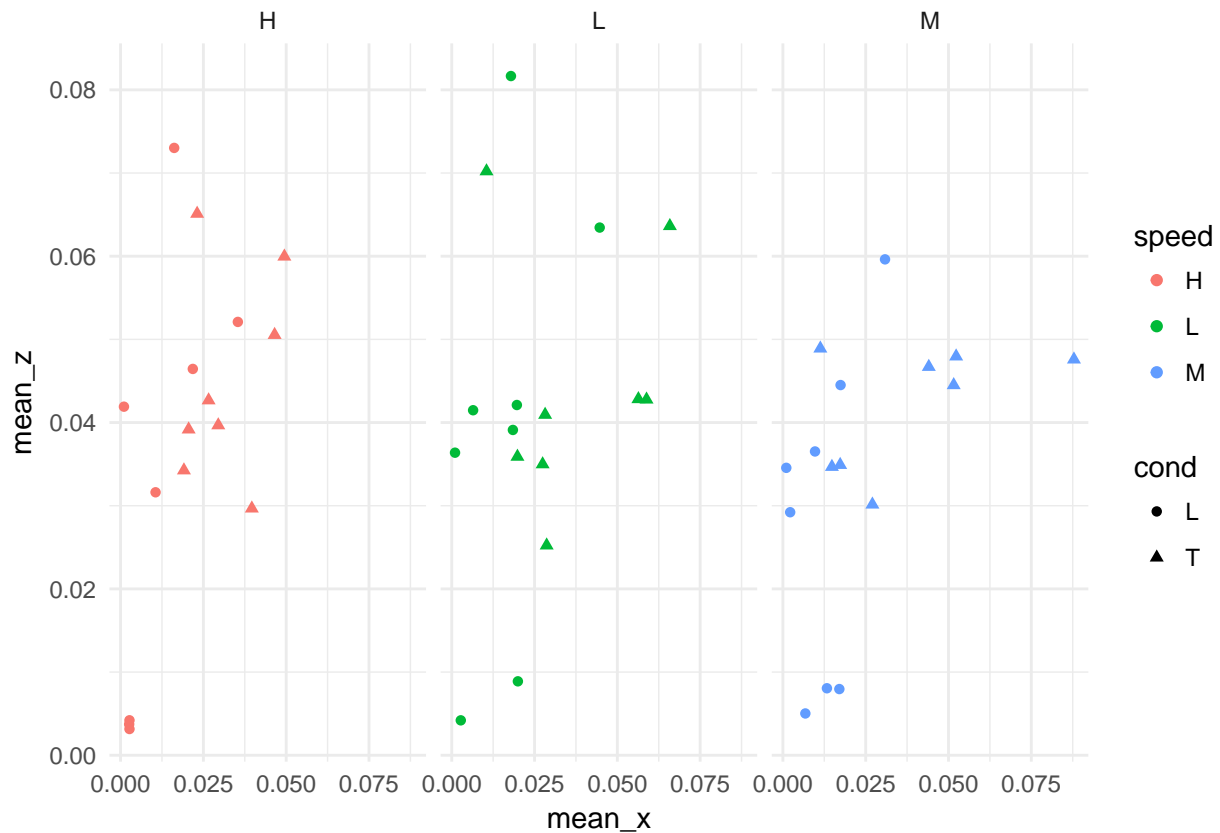
# data description from JvdH
#
# fish, condition, speed are the values for the 48 trials: low, medium high speed at L and T.
# H_ = high speed, M_ = medium speed, L_ = low speed
# x, y, z dimensions
# L, T conditions
#
# 4320 = all in one vector (M_zT = 4320 x 1, all of the values vertically concatenated)
# 540 x 8 = separated by 8 individuals (one column per fish). 540 positions of the snout of each animal
#
# Focused Approach: mn_cx, mn_cy, mn_cz = mean centroids calculated from those snout positions

# quick viz
#library(rgl)
#with(move, plot3d(H.xL.r[,1], H.yL.r[,1], H.zL.r[,1], type="l"))

summary_pos <- with(move, data.frame(Fish = unlist(fish),
                                     cond = c(rep("L", 24), rep("T", 24)),
                                     speed = rep(rep(c("H", "M", "L"), 8), 2),
                                     mean_x = c(apply(H.xL.r, 2, sd),
                                                  apply(M.xL.r, 2, mean),
                                                  apply(L.xL.r, 2, mean),
                                                  apply(H.xT.r, 2, mean),
                                                  apply(M.xT.r, 2, mean),
                                                  apply(L.xT.r, 2, mean)),
                                     mean_y = c(apply(H.yL.r, 2, sd),
                                                  apply(M.yL.r, 2, mean),
                                                  apply(L.yL.r, 2, mean),
                                                  apply(H.yT.r, 2, mean),
                                                  apply(M.yT.r, 2, mean),
                                                  apply(L.yT.r, 2, mean)),
                                     mean_z = c(apply(H.zL.r, 2, sd),
                                                  apply(M.zL.r, 2, mean),
                                                  apply(L.zL.r, 2, mean),
                                                  apply(H.zT.r, 2, mean),
                                                  apply(M.zT.r, 2, mean),
                                                  apply(L.zT.r, 2, mean))))

library(ggplot2)
p <- ggplot(summary_pos) +
  geom_point(aes(x=mean_x, y=mean_z, colour=speed, shape=cond)) +
  facet_wrap(~speed) +
  theme_minimal()
print(p)

```



```
library(reshape2)
summary_pos_melt <- melt(summary_pos, id=c("Fish", "cond", "speed"))

summary_pos_melt$speed <- factor(summary_pos_melt$speed,
                                levels(summary_pos_melt$speed)[c(2,3,1)])

# Position with condition/speed
p <- ggplot(summary_pos_melt, aes(x=speed, y=value, colour=cond, shape=cond)) +
  geom_point() +
  geom_smooth(aes(group=cond), method="lm") +
  facet_wrap(~variable) +
  scale_color_manual(values=c('red', 'blue'))+
  scale_shape_manual(values=c(17, 16))+
  theme_minimal()
print(p)
```

