Marker bat acoustic models draft 2

2022-11-28

# **Main objectives**

Model total nightly bat activity for all ground level detectors Create a binary response (bat activity per site per night = 1 or 0) model and a count response model Plot predictions:

* One predicition dataset with time and wind speeds set to a single average value
  + Plot the difference between habitats over time (nights in the season )
* One prediction data set with a full range of time and wind speeds
  + Plot the predicted cut off temperatures and wind speeds for total bat activity

# set project working directory according to the user system info   
# Otherwise, all the data (inputs and outputs, including figures) can be stored on a shared OneDrive folder  
  
user <- Sys.info()['effective\_user']   
user  
# this should print your nmbu user name - "apmc" in my case.   
  
wd <- getwd()  
wd  
# "C:/Users/apmc/OneDrive - Norwegian University of Life Sciences/Documents/1. PhD\_Main/GitHub\_link/MarkerBatAcousticAnalyses/MarkerBatAcousticAnalyses2022"  
  
  
##########################################################  
#### Work environment set up ####  
##########################################################  
library(knitr)  
library(data.table)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.0 ✔ purrr 0.3.5   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.3 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::between() masks data.table::between()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::first() masks data.table::first()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::last() masks data.table::last()  
## ✖ purrr::transpose() masks data.table::transpose()

library(beepr)  
library(lubridate)

## Loading required package: timechange  
##   
## Attaching package: 'lubridate'  
##   
## The following objects are masked from 'package:data.table':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday, week,  
## yday, year  
##   
## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(purrr)  
#renv::install("rstudio/renv")  
library(renv)

##   
## Attaching package: 'renv'  
##   
## The following object is masked from 'package:purrr':  
##   
## modify  
##   
## The following objects are masked from 'package:stats':  
##   
## embed, update  
##   
## The following objects are masked from 'package:utils':  
##   
## history, upgrade  
##   
## The following objects are masked from 'package:base':  
##   
## autoload, load, remove

library(stringr)  
library(janitor)

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(anytime)  
library(kableExtra)

## Warning in !is.null(rmarkdown::metadata$output) && rmarkdown::metadata$output  
## %in% : 'length(x) = 2 > 1' in coercion to 'logical(1)'

##   
## Attaching package: 'kableExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## group\_rows

library(papeR)

## Loading required package: car  
## Loading required package: carData  
##   
## Attaching package: 'car'  
##   
## The following object is masked from 'package:dplyr':  
##   
## recode  
##   
## The following object is masked from 'package:purrr':  
##   
## some  
##   
## Loading required package: xtable  
## Registered S3 method overwritten by 'papeR':  
## method from  
## Anova.lme car   
##   
## Attaching package: 'papeR'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## summarise, summarize  
##   
## The following object is masked from 'package:utils':  
##   
## toLatex

library(skimr)  
library(vtable)  
library(gratia)  
library(DHARMa)

## This is DHARMa 0.4.6. For overview type '?DHARMa'. For recent changes, type news(package = 'DHARMa')

library(mgcv)

## Loading required package: nlme  
##   
## Attaching package: 'nlme'  
##   
## The following object is masked from 'package:dplyr':  
##   
## collapse  
##   
## This is mgcv 1.8-41. For overview type 'help("mgcv-package")'.

library(tidymv)

## tidymv will be deprecated. Users are recommended  
## to check out the in-progress replacement tidygam  
## (https://github.com/stefanocoretta/tidygam).

##########################################################  
#### Import data, set up directories ####  
##########################################################  
  
# All input data can be found on a shared OneDrive folder - we can both share the same input folder but we should have different Output folders.   
  
# for Katrine   
#input <- ""  
  
# for Reed   
input <- "C:/Users/apmc/OneDrive - Norwegian University of Life Sciences/2. Marker 2019-2020/Marker 2022/SecondDraftAnalyses/Input/forModels"  
  
dataset1 <- "guild\_behavior\_summarytable\_site.csv"   
# df2C for the dataset aggregated by guild and bahavior from Marker aggregated to night\_all bats  
dataset2 <- "totalbatpass\_summarytable\_withbinary\_batpass\_night\_aggregated\_site data.csv"  
# df2C for the dataset aggregated to all batsa from Marker aggregated to night\_all bats  
dataset3 <- "nightlyaggregated\_zeros\_binary\_behavior and guild.csv"  
# dataset aggregated to guild and behavior before being table transformed   
dataset4 <- "nightlyaggregatedMarker2020\_totalbats\_zeroinserted\_weather\_binary.csv"  
# dataset aggregated to night by total bats before table transformed   
dataset5 <- "guild\_behavior\_batpass\_summarytable\_trimmed MRE social and met tower.csv"  
  
  
path1 <- str\_c(input, "/", dataset1)  
path2 <- str\_c(input, "/", dataset2)  
path3 <- str\_c(input, "/", dataset3)  
path4 <- str\_c(input, "/", dataset4)  
path5 <- str\_c(input, "/", dataset5)  
  
bats\_gb <- read\_csv(path1) # 11412 obs of 20 variables

## New names:  
## Rows: 11412 Columns: 20  
## ── Column specification  
## ──────────────────────────────────────────────────────── Delimiter: "," chr  
## (6): Site, guild, behavior, Habitat, Facility, Locality dbl (13): ...1,  
## Batpass\_sum, Batpass\_mean, Batpass\_max, batpass01\_sum, batp... date (1): night  
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ  
## Specify the column types or set `show\_col\_types = FALSE` to quiet this message.  
## • `` -> `...1`

bats\_tot <- read.csv(path2) # # 951 obs of 18 variables variables  
bats\_gb\_simple <- read.csv(path3) # 11412 obs of 17 vars   
bats\_tot\_simple <- read.csv(path4) # 951 obs of 15 variables  
bats\_gb\_trim <- read.csv(path5) # 4866 obs of 20 variables   
  
# for Reed  
output <- "C:/Users/apmc/OneDrive - Norwegian University of Life Sciences/2. Marker 2019-2020/Marker 2022/SecondDraftAnalyses/Reed/Outputs"  
  
# for Katrine   
#output <- ""  
  
##   
 file.name <- "Step3.Marker bat acoustics models draft 2"  
#   
 todays\_date <- Sys.Date()  
#   
dir.name <- str\_c(output,"/", file.name, "\_", todays\_date)  
 dir.name  
#   
 dir.create(dir.name) # be careful not to recreate existing directories

## Warning in dir.create(dir.name): 'C:\Users\apmc\OneDrive -  
## Norwegian University of Life Sciences\2. Marker 2019-2020\Marker  
## 2022\SecondDraftAnalyses\Reed\Outputs\Step3.Marker bat acoustics models draft  
## 2\_2022-12-14' already exists

output\_today <- dir.name  
output\_today

### Recommended Youtube videos on GAM to understand the R code below

<https://www.youtube.com/watch?v=q4_t8jXcQgc> <https://www.youtube.com/watch?v=sgw4cu8hrZM&t=4038s>

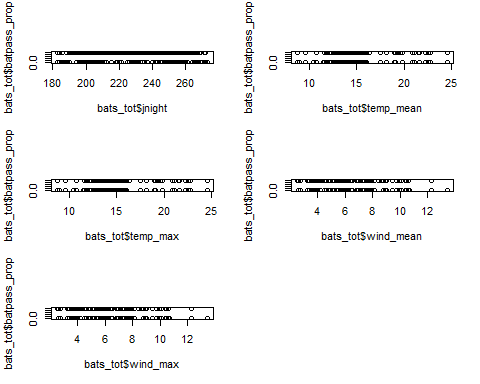
The “bats\_tot” dataset Nightly aggregated total bat activity

# Preparing the dataset for modeling

Starting with bats total

## 'data.frame': 951 obs. of 18 variables:  
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ night : chr "2020-07-01" "2020-07-01" "2020-07-01" "2020-07-01" ...  
## $ Site : chr "N02" "N08" "N10" "P02" ...  
## $ Batpass\_sum : int 2 6 0 0 0 0 1 0 7 0 ...  
## $ Batpass\_mean : int 2 6 0 0 0 0 1 0 7 0 ...  
## $ Batpass\_max : int 2 6 0 0 0 0 1 0 7 0 ...  
## $ batpass01\_sum : int 1 1 0 0 0 0 1 0 1 0 ...  
## $ batpass01\_length: int 1 1 1 1 1 1 1 1 1 1 ...  
## $ wind\_mean : num 4.83 4.83 4.83 4.83 4.83 ...  
## $ wind\_min : num 4.83 4.83 4.83 4.83 4.83 ...  
## $ wind\_max : num 4.83 4.83 4.83 4.83 4.83 ...  
## $ temp\_mean : num 14.8 14.8 14.8 14.8 14.8 ...  
## $ temp\_min : num 14.8 14.8 14.8 14.8 14.8 ...  
## $ temp\_max : num 14.8 14.8 14.8 14.8 14.8 ...  
## $ batpass\_prop : int 1 1 0 0 0 0 1 0 1 0 ...  
## $ Habitat : chr "Natural" "Natural" "Natural" "TurbinePad" ...  
## $ Facility : chr "North" "North" "South" "North" ...  
## $ Locality : chr "Turbine2" "Turbine8" "Turbine10" "Turbine2" ...

## [1] "X" "night" "Site" "Batpass\_sum"   
## [5] "Batpass\_mean" "Batpass\_max" "batpass01\_sum" "batpass01\_length"  
## [9] "wind\_mean" "wind\_min" "wind\_max" "temp\_mean"   
## [13] "temp\_min" "temp\_max" "batpass\_prop" "Habitat"   
## [17] "Facility" "Locality"



### Steps to prepare binary response (1=batpass, 0 = no batpass)

# This is now only nightly aggregated so I am not sure this has the same intended effect...   
  
bats\_tot$yes\_batpass <- bats\_tot$batpass01\_sum #batpass01\_sum is sum of '1' values in batpass01, i.e. number of observ. hours per night when batpass recorded  
  
## this dataset was not aggregaed up from hourly aggregated data, just to nightly so it will be slightly different but still essentially the same variable.   
  
bats\_tot$no\_batpass <- bats\_tot$batpass01\_length - bats\_tot$batpass01\_sum #batpass01\_length is number of observ. hours per night  
  
summary(bats\_tot)

## X night Site Batpass\_sum   
## Min. : 1.0 Length:951 Met45 : 70 Min. : 0.00   
## 1st Qu.:238.5 Class :character Met95 : 70 1st Qu.: 1.00   
## Median :476.0 Mode :character N08 : 60 Median : 6.00   
## Mean :476.0 N10 : 60 Mean : 20.44   
## 3rd Qu.:713.5 N11 : 60 3rd Qu.: 20.00   
## Max. :951.0 N14 : 60 Max. :659.00   
## (Other):571   
## Batpass\_mean Batpass\_max batpass01\_sum batpass01\_length  
## Min. : 0.00 Min. : 0.00 Min. :0.0000 Min. :1   
## 1st Qu.: 1.00 1st Qu.: 1.00 1st Qu.:1.0000 1st Qu.:1   
## Median : 6.00 Median : 6.00 Median :1.0000 Median :1   
## Mean : 20.44 Mean : 20.44 Mean :0.8181 Mean :1   
## 3rd Qu.: 20.00 3rd Qu.: 20.00 3rd Qu.:1.0000 3rd Qu.:1   
## Max. :659.00 Max. :659.00 Max. :1.0000 Max. :1   
##   
## wind\_mean wind\_min wind\_max temp\_mean   
## Min. : 2.571 Min. : 2.571 Min. : 2.571 Min. : 8.789   
## 1st Qu.: 4.486 1st Qu.: 4.486 1st Qu.: 4.486 1st Qu.:13.419   
## Median : 6.012 Median : 6.012 Median : 6.012 Median :14.930   
## Mean : 6.249 Mean : 6.249 Mean : 6.249 Mean :15.647   
## 3rd Qu.: 7.857 3rd Qu.: 7.857 3rd Qu.: 7.857 3rd Qu.:17.494   
## Max. :13.413 Max. :13.413 Max. :13.413 Max. :24.579   
##   
## temp\_min temp\_max batpass\_prop Habitat   
## Min. : 8.789 Min. : 8.789 Min. :0.0000 MeteorologicalTower:140   
## 1st Qu.:13.419 1st Qu.:13.419 1st Qu.:1.0000 Natural :391   
## Median :14.930 Median :14.930 Median :1.0000 TurbinePad :420   
## Mean :15.647 Mean :15.647 Mean :0.8181   
## 3rd Qu.:17.494 3rd Qu.:17.494 3rd Qu.:1.0000   
## Max. :24.579 Max. :24.579 Max. :1.0000   
##   
## Facility Locality jnight yes\_batpass   
## North:463 MeteorologicalTower:140 Min. :183.0 Min. :0.0000   
## South:488 Turbine10 :120 1st Qu.:207.0 1st Qu.:1.0000   
## Turbine11 :120 Median :228.0 Median :1.0000   
## Turbine14 :120 Mean :228.2 Mean :0.8181   
## Turbine8 :120 3rd Qu.:250.5 3rd Qu.:1.0000   
## Turbine2 :119 Max. :273.0 Max. :1.0000   
## (Other) :212   
## no\_batpass   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0000   
## Mean :0.1819   
## 3rd Qu.:0.0000   
## Max. :1.0000   
##

## Convert all bat passes over 50 per night to 50  
  
#Make a new column in the dataset  
bats\_tot$batpass50 <- bats\_tot$Batpass\_sum  
  
# reassign batpass values over 50 to 50  
bats\_tot$batpass50[bats\_tot$Batpass\_sum>50] <- 50  
## Do NOT use this as a response variable in models.   
  
  
  
### Drop the met tower   
  
df <- bats\_tot %>% filter(Habitat != "MeteorologicalTower") %>% droplevels()  
summary(df)

## X night Site Batpass\_sum   
## Min. : 1.0 Length:811 N08 : 60 Min. : 0.00   
## 1st Qu.:218.5 Class :character N10 : 60 1st Qu.: 2.00   
## Median :466.0 Mode :character N11 : 60 Median : 9.00   
## Mean :468.2 N14 : 60 Mean : 23.68   
## 3rd Qu.:716.5 P02 : 60 3rd Qu.: 24.00   
## Max. :951.0 P04 : 60 Max. :659.00   
## (Other):451   
## Batpass\_mean Batpass\_max batpass01\_sum batpass01\_length  
## Min. : 0.00 Min. : 0.00 Min. :0.0000 Min. :1   
## 1st Qu.: 2.00 1st Qu.: 2.00 1st Qu.:1.0000 1st Qu.:1   
## Median : 9.00 Median : 9.00 Median :1.0000 Median :1   
## Mean : 23.68 Mean : 23.68 Mean :0.8705 Mean :1   
## 3rd Qu.: 24.00 3rd Qu.: 24.00 3rd Qu.:1.0000 3rd Qu.:1   
## Max. :659.00 Max. :659.00 Max. :1.0000 Max. :1   
##   
## wind\_mean wind\_min wind\_max temp\_mean   
## Min. : 2.571 Min. : 2.571 Min. : 2.571 Min. : 8.789   
## 1st Qu.: 4.486 1st Qu.: 4.486 1st Qu.: 4.486 1st Qu.:13.419   
## Median : 6.012 Median : 6.012 Median : 6.012 Median :14.930   
## Mean : 6.230 Mean : 6.230 Mean : 6.230 Mean :15.610   
## 3rd Qu.: 7.822 3rd Qu.: 7.822 3rd Qu.: 7.822 3rd Qu.:17.494   
## Max. :13.413 Max. :13.413 Max. :13.413 Max. :24.579   
##   
## temp\_min temp\_max batpass\_prop Habitat   
## Min. : 8.789 Min. : 8.789 Min. :0.0000 Natural :391   
## 1st Qu.:13.419 1st Qu.:13.419 1st Qu.:1.0000 TurbinePad:420   
## Median :14.930 Median :14.930 Median :1.0000   
## Mean :15.610 Mean :15.610 Mean :0.8705   
## 3rd Qu.:17.494 3rd Qu.:17.494 3rd Qu.:1.0000   
## Max. :24.579 Max. :24.579 Max. :1.0000   
##   
## Facility Locality jnight yes\_batpass no\_batpass   
## North:463 Turbine10:120 Min. :183.0 Min. :0.0000 Min. :0.0000   
## South:348 Turbine11:120 1st Qu.:204.5 1st Qu.:1.0000 1st Qu.:0.0000   
## Turbine14:120 Median :227.0 Median :1.0000 Median :0.0000   
## Turbine2 :119 Mean :227.4 Mean :0.8705 Mean :0.1295   
## Turbine4 :104 3rd Qu.:251.0 3rd Qu.:1.0000 3rd Qu.:0.0000   
## Turbine8 :120 Max. :273.0 Max. :1.0000 Max. :1.0000   
## Turbine9 :108   
## batpass50   
## Min. : 0.00   
## 1st Qu.: 2.00   
## Median : 9.00   
## Mean :15.59   
## 3rd Qu.:24.00   
## Max. :50.00   
##

## ——————————————————————————–

# Binary response

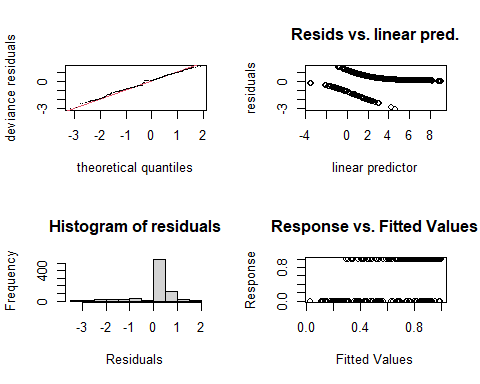
### s(Locality, bs = “re”)

### Removed the temp \* wind interaction

m1 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 90) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 15),   
 data = df, method = "REML",   
 family = binomial, select = TRUE)   
  
summary(m1)

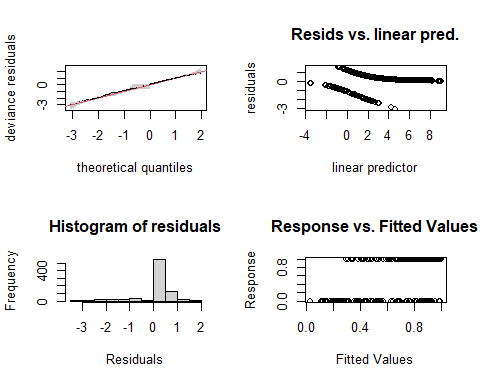
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + temp\_mean +   
## s(wind\_mean, k = 15)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.86886 1.40198 -2.046 0.040728 \*   
## HabitatTurbinePad 0.57156 0.43151 1.325 0.185324   
## temp\_mean 0.36848 0.09642 3.822 0.000133 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df Chi.sq p-value   
## s(Locality) 1.207 6 1.424 0.30669   
## s(jnight):HabitatNatural 2.498 87 9.728 0.00585 \*\*   
## s(jnight):HabitatTurbinePad 4.415 88 34.946 < 2e-16 \*\*\*  
## s(wind\_mean) 9.786 14 59.184 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.39 Deviance explained = 40.6%  
## -REML = 226.86 Scale est. = 1 n = 811

## Diagnostics ##   
## windows()  
par(mfrow = c(2,2))  
gam.check(m1)



##   
## Method: REML Optimizer: outer newton  
## full convergence after 11 iterations.  
## Gradient range [-5.425428e-05,4.118416e-05]  
## (score 226.8558 & scale 1).  
## Hessian positive definite, eigenvalue range [4.359367e-05,1.843758].  
## Model rank = 202 / 202   
##   
## 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(Locality) 7.00 1.21 NA NA   
## s(jnight):HabitatNatural 89.00 2.50 0.89 0.010 \*\*  
## s(jnight):HabitatTurbinePad 89.00 4.42 0.89 0.035 \*   
## s(wind\_mean) 14.00 9.79 0.88 0.005 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m1, rep=500)



##   
## Method: REML Optimizer: outer newton  
## full convergence after 11 iterations.  
## Gradient range [-5.425428e-05,4.118416e-05]  
## (score 226.8558 & scale 1).  
## Hessian positive definite, eigenvalue range [4.359367e-05,1.843758].  
## Model rank = 202 / 202   
##   
## 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(Locality) 7.00 1.21 NA NA   
## s(jnight):HabitatNatural 89.00 2.50 0.89 0.01 \*\*  
## s(jnight):HabitatTurbinePad 89.00 4.42 0.89 0.01 \*\*  
## s(wind\_mean) 14.00 9.79 0.88 0.01 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#rep=500 gives you a polygon on the QQ plot, which the observed values should lie within  
  
# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m1 <- sum( residuals(m1, "pearson")^2 ) / m1$df.residual  
overdispersion.m1

## [1] 0.7636035

## Notes on model fit

A bit under dispersed and the K values do not fit perfectly but I guess it will work for now.

### Basis dimension (k) checking results:

bmp(file.path(output\_today, "checking k dimensions m1 bats total.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m1, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
draw(m1, scales ="free") #function 'draw' is from package 'gratia'  
draw(m1, scales ="fixed")  
  
dev.off()

## png   
## 2

# Make prediction figures

### Better visualize the wind and temperature cutoffs

Make make different prediction datasets: - One with a single averaged value for wind and temperature - One with a range of wind and temperature values Also test using different turbine localities (turbines 2 and 9)

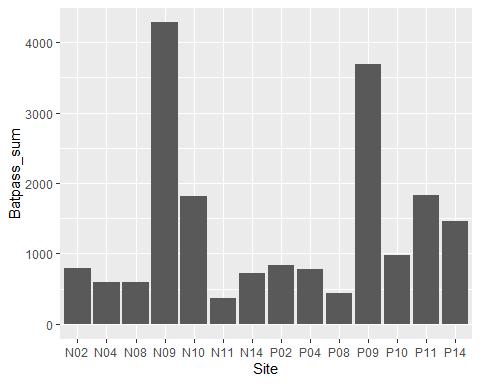
names(df)

## [1] "X" "night" "Site" "Batpass\_sum"   
## [5] "Batpass\_mean" "Batpass\_max" "batpass01\_sum" "batpass01\_length"  
## [9] "wind\_mean" "wind\_min" "wind\_max" "temp\_mean"   
## [13] "temp\_min" "temp\_max" "batpass\_prop" "Habitat"   
## [17] "Facility" "Locality" "jnight" "yes\_batpass"   
## [21] "no\_batpass" "batpass50"

levels(df$Locality)

## [1] "Turbine10" "Turbine11" "Turbine14" "Turbine2" "Turbine4" "Turbine8"   
## [7] "Turbine9"

# Quick summary in the differences in bat activity recorded at the different sites  
ggplot(df) + geom\_bar(aes(x=Site, y = Batpass\_sum), stat = "identity")



# -----------------------------------------------------------------------------  
# Wind and temperature are fixed  
  
# Turbine 9  
newdata1 <- expand.grid(temp\_mean = 14,  
 wind\_mean = 7,  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine9",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
#Turbine 2  
newdata1a <- expand.grid(temp\_mean = 14,  
 wind\_mean = 7,  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine2",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
# -----------------------------------------------------------------------------  
# Wind and temperature are a range   
  
# Turbine 9  
newdata2 <- expand.grid(temp\_mean= c(4, 6, 8,10, 12, 14, 16, 18, 20),  
 wind\_mean = c(2, 4, 6, 8, 10, 12, 14, 16),  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine9",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
  
  
  
# Turbine 2  
newdata2a <- expand.grid(temp\_mean= c(4, 6, 8,10, 12, 14, 16, 18, 20),  
 wind\_mean = c(2, 4, 6, 8, 10, 12, 14, 16),  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine2",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
  
  
  
  
############################################  
# # Fitting predictions  
############################################  
  
# Wind and temperature are fixed  
  
# Turbine 9  
fit1 <- predict.gam(m1, newdata = newdata1, se.fit=TRUE, type = "response")  
fit1df <- data.frame(newdata1, fit1)  
fit1df <- transform(fit1df, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit1df)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. :14 Min. :7 Natural :91 Turbine9:182 Min. :183.0   
## 1st Qu.:14 1st Qu.:7 TurbinePad:91 1st Qu.:205.2   
## Median :14 Median :7 Median :228.0   
## Mean :14 Mean :7 Mean :228.0   
## 3rd Qu.:14 3rd Qu.:7 3rd Qu.:250.8   
## Max. :14 Max. :7 Max. :273.0   
## fit se.fit upper lower   
## Min. :0.2217 Min. :0.007736 Min. :0.4840 Min. :-0.04068   
## 1st Qu.:0.9106 1st Qu.:0.024747 1st Qu.:1.0056 1st Qu.: 0.81250   
## Median :0.9447 Median :0.034317 Median :1.0093 Median : 0.87644   
## Mean :0.9132 Mean :0.041111 Mean :0.9955 Mean : 0.83102   
## 3rd Qu.:0.9601 3rd Qu.:0.049076 3rd Qu.:1.0130 3rd Qu.: 0.90682   
## Max. :0.9928 Max. :0.148111 Max. :1.0558 Max. : 0.97723

# Turbine 2  
fit1a <- predict.gam(m1, newdata = newdata1a, se.fit=TRUE, type = "response")  
fit1adf <- data.frame(newdata1a, fit1a)  
fit1adf <- transform(fit1adf, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit1adf)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. :14 Min. :7 Natural :91 Turbine2:182 Min. :183.0   
## 1st Qu.:14 1st Qu.:7 TurbinePad:91 1st Qu.:205.2   
## Median :14 Median :7 Median :228.0   
## Mean :14 Mean :7 Mean :228.0   
## 3rd Qu.:14 3rd Qu.:7 3rd Qu.:250.8   
## Max. :14 Max. :7 Max. :273.0   
## fit se.fit upper lower   
## Min. :0.1997 Min. :0.008783 Min. :0.4380 Min. :-0.03851   
## 1st Qu.:0.8993 1st Qu.:0.027551 1st Qu.:1.0053 1st Qu.: 0.79021   
## Median :0.9374 Median :0.038378 Median :1.0098 Median : 0.86093   
## Mean :0.9042 Mean :0.044443 Mean :0.9931 Mean : 0.81534   
## 3rd Qu.:0.9548 3rd Qu.:0.054702 3rd Qu.:1.0140 3rd Qu.: 0.89530   
## Max. :0.9918 Max. :0.142103 Max. :1.0616 Max. : 0.97411

# -----------------------------------------------------------------------------  
# Wind and temperature are a range   
  
# Turbine 9  
fit2 <- predict.gam(m1, newdata = newdata2, se.fit=TRUE, type = "response")  
fit2df <- data.frame(newdata2, fit2)  
fit2df <- transform(fit2df, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit2df)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. : 4 Min. : 2.0 Natural :6552 Turbine9:13104 Min. :183   
## 1st Qu.: 8 1st Qu.: 5.5 TurbinePad:6552 1st Qu.:205   
## Median :12 Median : 9.0 Median :228   
## Mean :12 Mean : 9.0 Mean :228   
## 3rd Qu.:16 3rd Qu.:12.5 3rd Qu.:251   
## Max. :20 Max. :16.0 Max. :273   
## fit se.fit upper lower   
## Min. :0.0000 Min. :0.0000001 Min. :0.0000003 Min. :-0.8513   
## 1st Qu.:0.1089 1st Qu.:0.0108969 1st Qu.:0.3673764 1st Qu.:-0.0374   
## Median :0.6319 Median :0.0668432 Median :1.0035045 Median : 0.1487   
## Mean :0.5457 Mean :0.1231173 Mean :0.7919842 Mean : 0.2995   
## 3rd Qu.:0.9440 3rd Qu.:0.1806918 3rd Qu.:1.0465183 3rd Qu.: 0.8449   
## Max. :0.9997 Max. :0.6555147 Max. :1.8675469 Max. : 0.9987

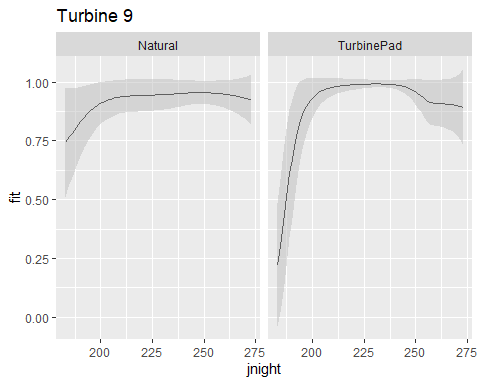
# Turbine 2  
fit2a <- predict.gam(m1, newdata = newdata2a, se.fit=TRUE, type = "response")  
fit2adf <- data.frame(newdata2a, fit2a)  
fit2adf <- transform(fit2adf, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit2adf)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. : 4 Min. : 2.0 Natural :6552 Turbine2:13104 Min. :183   
## 1st Qu.: 8 1st Qu.: 5.5 TurbinePad:6552 1st Qu.:205   
## Median :12 Median : 9.0 Median :228   
## Mean :12 Mean : 9.0 Mean :228   
## 3rd Qu.:16 3rd Qu.:12.5 3rd Qu.:251   
## Max. :20 Max. :16.0 Max. :273   
## fit se.fit upper lower   
## Min. :0.0000 Min. :0.0000001 Min. :0.0000003 Min. :-0.8405   
## 1st Qu.:0.0967 1st Qu.:0.0118544 1st Qu.:0.3291051 1st Qu.:-0.0399   
## Median :0.6008 Median :0.0695916 Median :1.0028264 Median : 0.1094   
## Mean :0.5331 Mean :0.1232111 Mean :0.7795610 Mean : 0.2867   
## 3rd Qu.:0.9366 3rd Qu.:0.1790094 3rd Qu.:1.0451045 3rd Qu.: 0.8253   
## Max. :0.9996 Max. :0.6585591 Max. :1.8587697 Max. : 0.9985

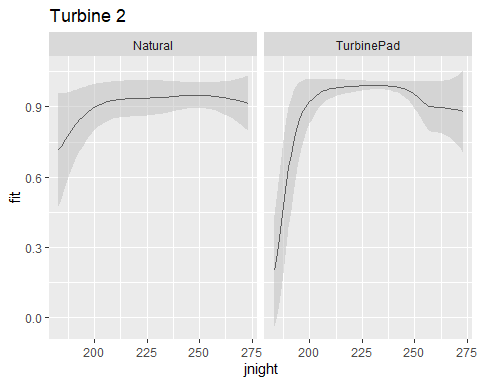
# Plotting predictions

## Comparing habitats across seasons

# Turbine 9  
fit1df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line() + facet\_wrap(~Habitat) + ggtitle("Turbine 9") +   
 geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5)



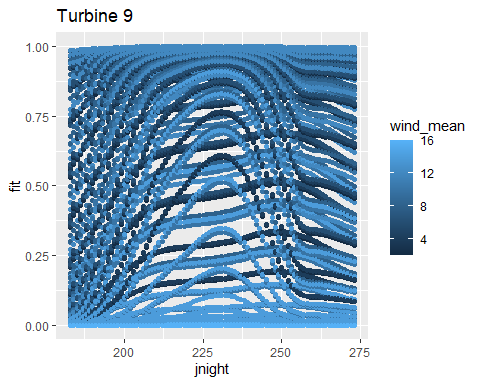
# -----------------------------------------------------------------------------  
  
# Turbine 2  
fit1adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line() + facet\_wrap(~Habitat) + ggtitle ("Turbine 2") +  
 geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5)



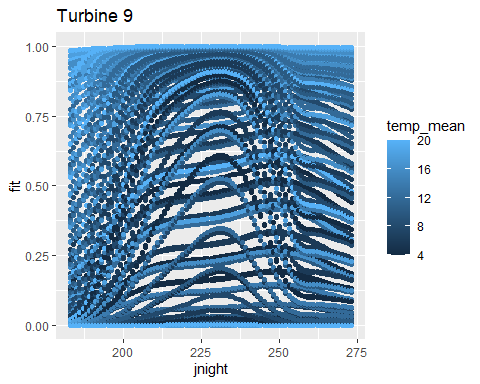
## Add confidence intervals, clean up the background

## Looking for temperature and wind speed cut offs

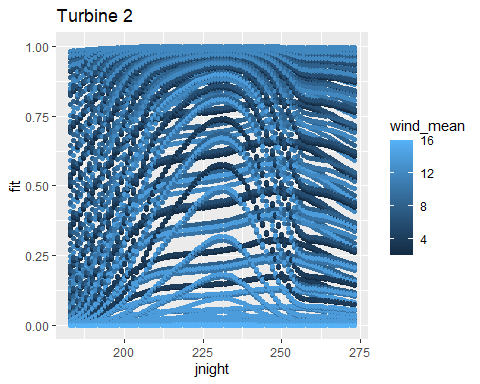
# Using colored points to represent average nightly wind speeds and temperatures... perhaps there is a more meaningful way to represent this.   
  
# Turbine 9 - wind   
fit2df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = wind\_mean)) + ggtitle("Turbine 9")



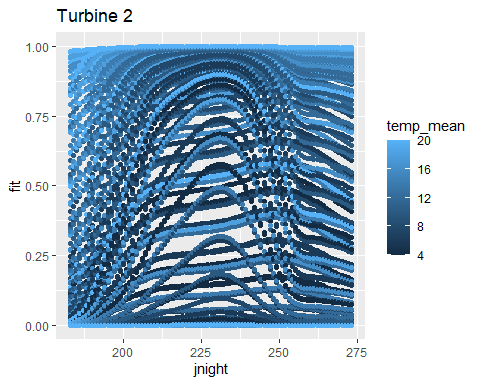
# Turbine 9 - temp   
fit2df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ggtitle("Turbine 9")



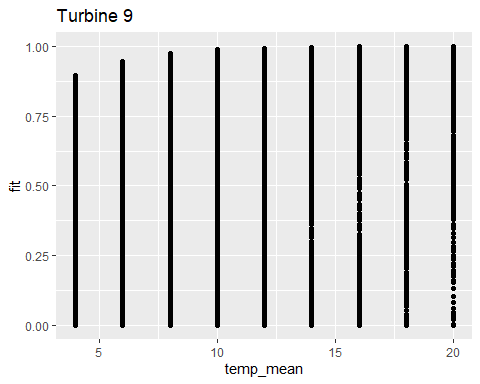
# -----------------------------------------------------------------------------  
  
# Turbine 2 - wind  
fit2adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = wind\_mean)) + ggtitle ("Turbine 2")



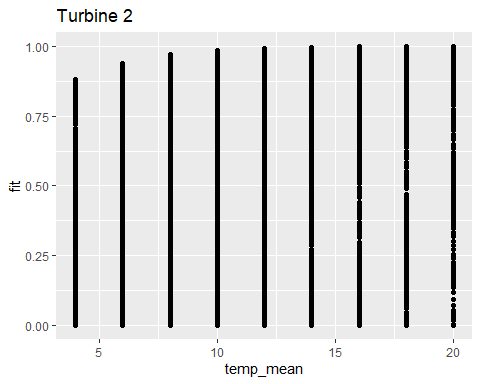
# Turbine 2 - temp   
fit2adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ggtitle ("Turbine 2")



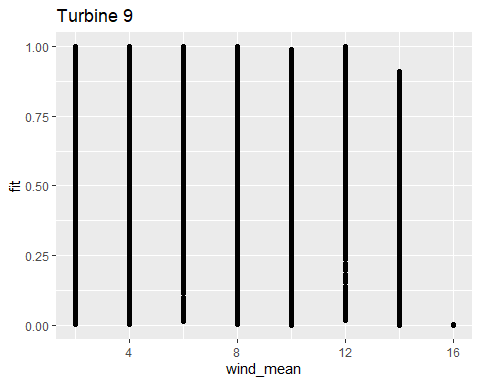
# -----------------------------------------------------------------------------  
  
# Temp is the response variable   
# Turbine 9  
fit2df %>% ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point() + ggtitle("Turbine 9")



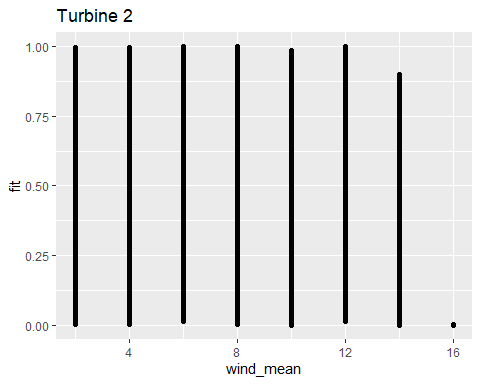
# Turbine 2  
fit2adf %>% ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point() + ggtitle ("Turbine 2")



# Wind is the response variable   
# Turbine 9  
fit2df %>% ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point() + ggtitle("Turbine 9")



# Turbine 2  
fit2adf %>% ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point() + ggtitle ("Turbine 2")



## Not terribly insightful - Could try plotting predictions for each temperature degree change or wind speed change?   
## Not seeing much of a difference between the turbines so far.   
# low wind, high temperature   
# high wind, low temperature  
# low wind, low temperature  
# high wind, high temperature   
  
## Still does not help with plotting cut offs though...

|  |
| --- |
| ## Katrine’s suggestion/code for visualizing different wind speeds and temperatures ### Binary model predictions |
| ```r # For plotting temp |
| newdata\_temp<- expand.grid(temp\_mean = c(4, 10, 16, 20), wind\_mean = seq(min(dfwind\_mean), length = 300), Habitat = levels(df$Habitat), Locality = “Turbine9”, jnight = 212) |
| # Turbine 9, fewer wind and temp values fit\_temp <- predict.gam(m1, newdata = newdata\_temp, se.fit=TRUE, type = “response”) fit\_temp\_df <- data.frame(newdata\_temp, fit\_temp) fit\_temp\_df <- transform(fit\_temp\_df, upper = fit + (2*se.fit), lower = fit-(2*se.fit)) summary(fit\_temp\_df) ``` |
| ## temp\_mean wind\_mean Habitat Locality ## Min. : 4.0 Min. : 2.571 Natural :1200 Turbine9:2400 ## 1st Qu.: 8.5 1st Qu.: 5.281 TurbinePad:1200 ## Median :13.0 Median : 7.992 ## Mean :12.5 Mean : 7.992 ## 3rd Qu.:17.0 3rd Qu.:10.702 ## Max. :20.0 Max. :13.413 ## jnight fit se.fit upper ## Min. :212 Min. :0.02588 Min. :0.0009114 Min. :0.08646 ## 1st Qu.:212 1st Qu.:0.64740 1st Qu.:0.0084324 1st Qu.:1.00202 ## Median :212 Median :0.93889 Median :0.0530723 Median :1.00770 ## Mean :212 Mean :0.78283 Mean :0.1046597 Mean :0.99215 ## 3rd Qu.:212 3rd Qu.:0.99062 3rd Qu.:0.1790121 3rd Qu.:1.05365 ## Max. :212 Max. :0.99924 Max. :0.4551274 Max. :1.48528 ## lower ## Min. :-0.4117 ## 1st Qu.: 0.0821 ## Median : 0.8277 ## Mean : 0.5735 ## 3rd Qu.: 0.9735 ## Max. : 0.9973 |
| r # Plotting temp at various wind speeds new\_labels <- c("4" = "temp 4", "10" = "temp 10", "16" = "temp 16", "20" = "temp 20") # Using binary response model, turbine 9 predictions plt1 <- fit\_temp\_df %>% ggplot( aes(x=wind\_mean, y = fit, group = factor(temp\_mean))) + geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5) + geom\_line() + facet\_wrap(~ temp\_mean, scales ='fixed', labeller = labeller(temp\_mean = new\_labels)) + labs(x = "Average wind speed m/s", y ="Proportion of hours per night with batpass>0") plt1 |
|  |
| ```r ## —————————————————————————- |
| ## For plotting wind summary(df$temp\_mean) ``` |
| ## Min. 1st Qu. Median Mean 3rd Qu. Max. ## 8.789 13.419 14.930 15.610 17.494 24.579 |
| ```r # Turbine 9, fewer temp and wind variables newdata\_wind<- expand.grid(temp\_mean = seq(min (dftemp\_mean), length = 300), wind\_mean = c(2, 8, 12, 16), Habitat = levels(df$Habitat), Locality = “Turbine9”, jnight = 212) |
| # Turbine 9, fewer wind and temp values fit\_wind <- predict.gam(m1, newdata = newdata\_wind, se.fit=TRUE, type = “response”) fit\_wind\_df <- data.frame(newdata\_wind, fit\_wind) fit\_wind\_df <- transform(fit\_wind\_df, upper = fit + (2*se.fit), lower = fit-(2*se.fit)) summary(fit\_wind\_df) ``` |
| ## temp\_mean wind\_mean Habitat Locality ## Min. : 8.789 Min. : 2.0 Natural :1200 Turbine9:2400 ## 1st Qu.:12.737 1st Qu.: 6.5 TurbinePad:1200 ## Median :16.684 Median :10.0 ## Mean :16.684 Mean : 9.5 ## 3rd Qu.:20.631 3rd Qu.:13.0 ## Max. :24.579 Max. :16.0 ## jnight fit se.fit upper ## Min. :212 Min. :0.0000074 Min. :0.0000428 Min. :0.000093 ## 1st Qu.:212 1st Qu.:0.3782308 1st Qu.:0.0020542 1st Qu.:0.773533 ## Median :212 Median :0.9655359 Median :0.0088438 Median :1.003923 ## Mean :212 Mean :0.7122659 Mean :0.0464635 Mean :0.805193 ## 3rd Qu.:212 3rd Qu.:0.9949260 3rd Qu.:0.0393697 3rd Qu.:1.029160 ## Max. :212 Max. :0.9998569 Max. :0.5492066 Max. :1.635464 ## lower ## Min. :-0.5970960 ## 1st Qu.:-0.0002458 ## Median : 0.8867931 ## Mean : 0.6193389 ## 3rd Qu.: 0.9819160 ## Max. : 0.9993799 |
| ```r ## Plotting wind speeds at various temperatures |
| new\_labels <- c(“2” = “wind 2 m/s”, “8” = “wind 8 m/s”, “12” = “wind 12 /s”, “16” = “wind 16 m/s”) |
| # Using binary response model, turbine 9 predictions plt2 <- fit\_wind\_df %>% ggplot(aes(x=temp\_mean, y = fit, group = factor(wind\_mean)), fill = ‘grey’, alpha = 0.5) + geom\_line() + facet\_wrap(~ wind\_mean, scales =‘fixed’, labeller = labeller(wind\_mean = new\_labels)) + labs(x = “Average temperature”, y =“Proportion of hours per night with batpass>0”) plt2 ``` |
|  |

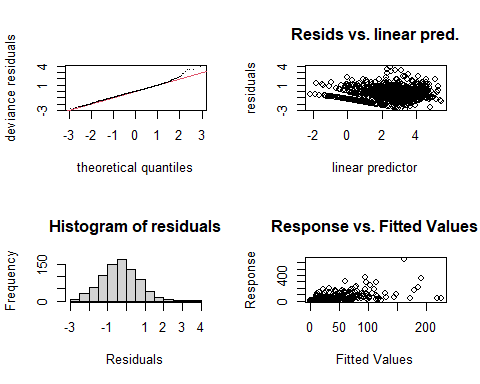
# m2

### Count response

m2 <- gam(Batpass\_sum ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 90) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 20) ,   
 data = df, method = "REML",   
 family = nb(), select=TRUE)   
  
summary(m2)

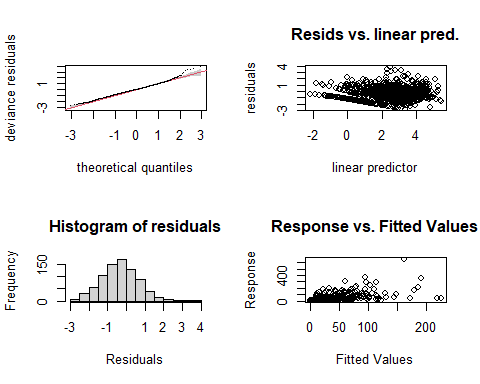
##   
## Family: Negative Binomial(1.179)   
## Link function: log   
##   
## Formula:  
## Batpass\_sum ~ s(Locality, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp", k = 90) + Habitat + temp\_mean + s(wind\_mean, k = 20)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.38745 0.43693 -0.887 0.375   
## HabitatTurbinePad -0.01596 0.07224 -0.221 0.825   
## temp\_mean 0.18111 0.02301 7.872 3.48e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Approximate significance of smooth terms:  
## edf Ref.df Chi.sq p-value   
## s(Locality) 5.865 6 261.96 <2e-16 \*\*\*  
## s(jnight):HabitatNatural 7.057 88 103.79 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 12.632 88 187.94 <2e-16 \*\*\*  
## s(wind\_mean) 3.907 19 81.85 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.37 Deviance explained = 59.1%  
## -REML = 2915.4 Scale est. = 1 n = 811

# windows()  
par(mfrow = c(2,2))  
gam.check(m2) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 11 iterations.  
## Gradient range [-0.0006201695,0.00029558]  
## (score 2915.398 & scale 1).  
## Hessian positive definite, eigenvalue range [0.0002537826,300.2946].  
## Model rank = 207 / 207   
##   
## 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(Locality) 7.00 5.87 NA NA   
## s(jnight):HabitatNatural 89.00 7.06 0.83 0.01 \*\*  
## s(jnight):HabitatTurbinePad 89.00 12.63 0.83 0.01 \*\*  
## s(wind\_mean) 19.00 3.91 0.83 0.01 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m2, rep=500) #rep=500 gives you a polygon on the QQ plot, which the observed values should lie within



##   
## Method: REML Optimizer: outer newton  
## full convergence after 11 iterations.  
## Gradient range [-0.0006201695,0.00029558]  
## (score 2915.398 & scale 1).  
## Hessian positive definite, eigenvalue range [0.0002537826,300.2946].  
## Model rank = 207 / 207   
##   
## 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(Locality) 7.00 5.87 NA NA   
## s(jnight):HabitatNatural 89.00 7.06 0.83 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 89.00 12.63 0.83 0.010 \*\*   
## s(wind\_mean) 19.00 3.91 0.83 0.005 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m2 <- sum( residuals(m2, "pearson")^2 ) / m2$df.residual  
overdispersion.m2

## [1] 1.284253

### Basis dimension (k) checking results:

bmp(file.path(output\_today, "checking k dimensions m2 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m2, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m2, scales ="free") #function 'draw' is from package 'gratia'  
draw(m2, scales ="fixed")  
   
dev.off()

## png   
## 2

The K values are slightly off but the dispersion is not bad and the diagnostic plots looks okay

# Make prediction figures

### Better visualize the wind and temperature cutoffs

Make make different prediction datasets: - One with a single averaged value for wind and temperature - One with a range of wind and temperature values Also test using different turbine localities (turbines 2 and 9)

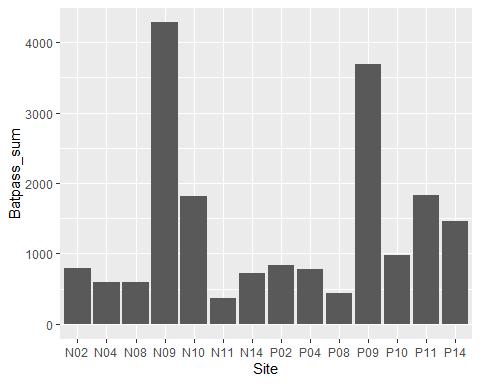
names(df)

## [1] "X" "night" "Site" "Batpass\_sum"   
## [5] "Batpass\_mean" "Batpass\_max" "batpass01\_sum" "batpass01\_length"  
## [9] "wind\_mean" "wind\_min" "wind\_max" "temp\_mean"   
## [13] "temp\_min" "temp\_max" "batpass\_prop" "Habitat"   
## [17] "Facility" "Locality" "jnight" "yes\_batpass"   
## [21] "no\_batpass" "batpass50"

levels(df$Locality)

## [1] "Turbine10" "Turbine11" "Turbine14" "Turbine2" "Turbine4" "Turbine8"   
## [7] "Turbine9"

# Quick summary in the differences in bat activity recorded at the different sites  
ggplot(df) + geom\_bar(aes(x=Site, y = Batpass\_sum), stat = "identity")



# -----------------------------------------------------------------------------  
# Wind and temperature are fixed  
  
# Turbine 9  
newdata1 <- expand.grid(temp\_mean = 14,  
 wind\_mean = 7,  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine9",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
#Turbine 2  
newdata1a <- expand.grid(temp\_mean = 14,  
 wind\_mean = 7,  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine2",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
# -----------------------------------------------------------------------------  
# Wind and temperature are a range   
  
# Turbine 9  
newdata2 <- expand.grid(temp\_mean= c(4, 6, 8,10, 12, 14, 16, 18, 20),  
 wind\_mean = c(2, 4, 6, 8, 10, 12, 14, 16),  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine9",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
  
  
# Turbine 2  
newdata2a <- expand.grid(temp\_mean= c(4, 6, 8,10, 12, 14, 16, 18, 20),  
 wind\_mean = c(2, 4, 6, 8, 10, 12, 14, 16),  
 Habitat = levels(df$Habitat),  
 Locality = "Turbine2",  
 jnight = seq(min(df$jnight), max(df$jnight), 1))  
  
  
############################################  
# # Fitting predictions  
############################################  
  
# Wind and temperature are fixed  
  
# Turbine 9  
fit1 <- predict.gam(m2, newdata = newdata1, se.fit = TRUE, type = "response")  
fit1df <- data.frame(newdata1, fit1)  
fit1df <- transform(fit1df, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit1df)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. :14 Min. :7 Natural :91 Turbine9:182 Min. :183.0   
## 1st Qu.:14 1st Qu.:7 TurbinePad:91 1st Qu.:205.2   
## Median :14 Median :7 Median :228.0   
## Mean :14 Mean :7 Mean :228.0   
## 3rd Qu.:14 3rd Qu.:7 3rd Qu.:250.8   
## Max. :14 Max. :7 Max. :273.0   
## fit se.fit upper lower   
## Min. : 1.286 Min. : 0.7548 Min. : 2.795 Min. :-0.2238   
## 1st Qu.:23.586 1st Qu.: 4.6154 1st Qu.: 32.886 1st Qu.:14.0543   
## Median :36.797 Median : 7.4641 Median : 51.672 Median :21.7587   
## Mean :43.179 Mean : 8.9110 Mean : 61.001 Mean :25.3568   
## 3rd Qu.:67.630 3rd Qu.:12.7868 3rd Qu.: 93.823 3rd Qu.:39.5782   
## Max. :91.706 Max. :20.8057 Max. :132.942 Max. :52.2434

# Turbine 2  
fit1a <- predict.gam(m2, newdata = newdata1a, se.fit = TRUE, type = "response")  
fit1adf <- data.frame(newdata1a, fit1a)  
fit1adf <- transform(fit1adf, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit1adf)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. :14 Min. :7 Natural :91 Turbine2:182 Min. :183.0   
## 1st Qu.:14 1st Qu.:7 TurbinePad:91 1st Qu.:205.2   
## Median :14 Median :7 Median :228.0   
## Mean :14 Mean :7 Mean :228.0   
## 3rd Qu.:14 3rd Qu.:7 3rd Qu.:250.8   
## Max. :14 Max. :7 Max. :273.0   
## fit se.fit upper lower   
## Min. : 0.2548 Min. :0.1474 Min. : 0.5496 Min. :-0.04007   
## 1st Qu.: 4.6733 1st Qu.:0.9457 1st Qu.: 6.5691 1st Qu.: 2.77314   
## Median : 7.2911 Median :1.4297 Median :10.1098 Median : 4.45272   
## Mean : 8.5556 Mean :1.7863 Mean :12.1283 Mean : 4.98288   
## 3rd Qu.:13.4004 3rd Qu.:2.6360 3rd Qu.:18.7533 3rd Qu.: 7.99944   
## Max. :18.1710 Max. :3.9893 Max. :26.1496 Max. :10.19229

# -----------------------------------------------------------------------------  
# Wind and temperature are a range   
  
# Turbine 9  
fit2 <- predict.gam(m2, newdata = newdata2, se.fit = TRUE, type = "response")  
fit2df <- data.frame(newdata2, fit2)  
fit2df <- transform(fit2df, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit2df)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. : 4 Min. : 2.0 Natural :6552 Turbine9:13104 Min. :183   
## 1st Qu.: 8 1st Qu.: 5.5 TurbinePad:6552 1st Qu.:205   
## Median :12 Median : 9.0 Median :228   
## Mean :12 Mean : 9.0 Mean :228   
## 3rd Qu.:16 3rd Qu.:12.5 3rd Qu.:251   
## Max. :20 Max. :16.0 Max. :273   
## fit se.fit upper lower   
## Min. : 0.01539 Min. : 0.01413 Min. : 0.0437 Min. : -7.4896   
## 1st Qu.: 3.07628 1st Qu.: 1.24588 1st Qu.: 5.7208 1st Qu.: 0.2585   
## Median : 8.91639 Median : 3.01897 Median : 15.2166 Median : 2.7106   
## Mean : 21.85944 Mean : 5.60078 Mean : 33.0610 Mean : 10.6579   
## 3rd Qu.: 25.03974 3rd Qu.: 6.93208 3rd Qu.: 38.4502 3rd Qu.: 11.4369   
## Max. :290.88915 Max. :65.16826 Max. :421.2257 Max. :161.2041

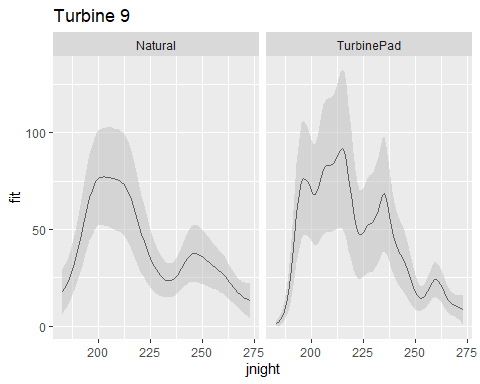
# Turbine 2  
fit2a <- predict.gam(m2, newdata = newdata2a, se.fit = TRUE, type = "response")  
fit2adf <- data.frame(newdata2a, fit2a)  
fit2adf <- transform(fit2adf, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
summary(fit2adf)

## temp\_mean wind\_mean Habitat Locality jnight   
## Min. : 4 Min. : 2.0 Natural :6552 Turbine2:13104 Min. :183   
## 1st Qu.: 8 1st Qu.: 5.5 TurbinePad:6552 1st Qu.:205   
## Median :12 Median : 9.0 Median :228   
## Mean :12 Mean : 9.0 Mean :228   
## 3rd Qu.:16 3rd Qu.:12.5 3rd Qu.:251   
## Max. :20 Max. :16.0 Max. :273   
## fit se.fit upper lower   
## Min. : 0.00305 Min. : 0.002794 Min. : 0.00864 Min. :-1.4605   
## 1st Qu.: 0.60954 1st Qu.: 0.249807 1st Qu.: 1.14239 1st Qu.: 0.0469   
## Median : 1.76672 Median : 0.605291 Median : 3.02804 Median : 0.5248   
## Mean : 4.33129 Mean : 1.113161 Mean : 6.55761 Mean : 2.1050   
## 3rd Qu.: 4.96144 3rd Qu.: 1.383337 3rd Qu.: 7.63555 3rd Qu.: 2.2511   
## Max. :57.63757 Max. :12.535556 Max. :82.27104 Max. :33.0041

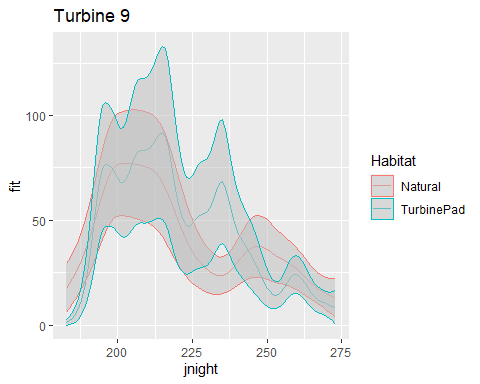
# Plotting predictions

## Comparing habitats across seasons

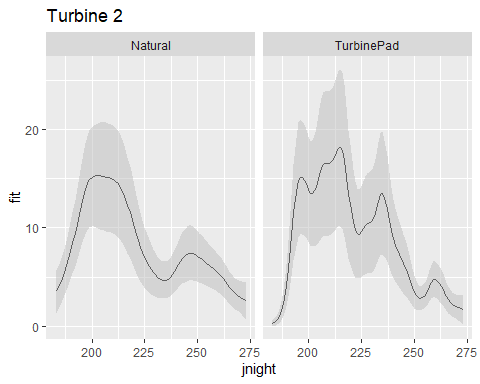
# Turbine 9  
fit1df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line() + facet\_wrap(~Habitat) + ggtitle("Turbine 9") +  
 geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5)



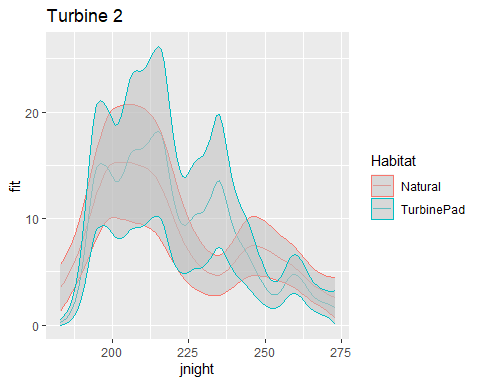
fit1df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line(aes(colour = Habitat)) + ggtitle("Turbine 9") +  
 geom\_ribbon(aes(ymin = lower, ymax = upper, color = Habitat), fill = 'grey', alpha = 0.5)



# -----------------------------------------------------------------------------  
  
# Turbine 2  
fit1adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line() + facet\_wrap(~Habitat) + ggtitle ("Turbine 2") +  
 geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5)



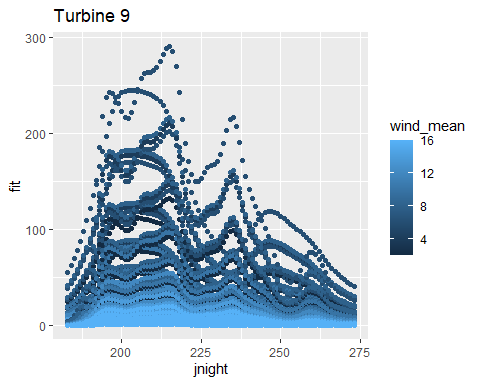
fit1adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_line(aes(colour = Habitat)) + ggtitle ("Turbine 2") +  
 geom\_ribbon(aes(ymin = lower, ymax = upper, color = Habitat), fill = 'grey', alpha = 0.5)



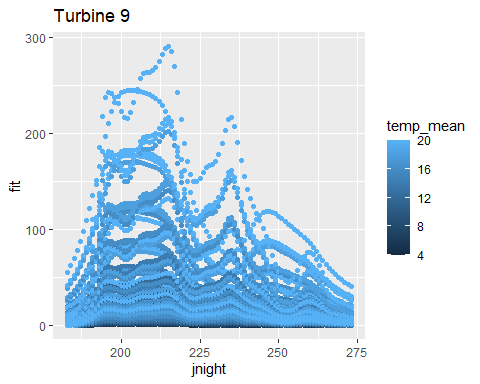
## Add confidence intervals, clean up the background

## Looking for temperature and wind speed cut offs

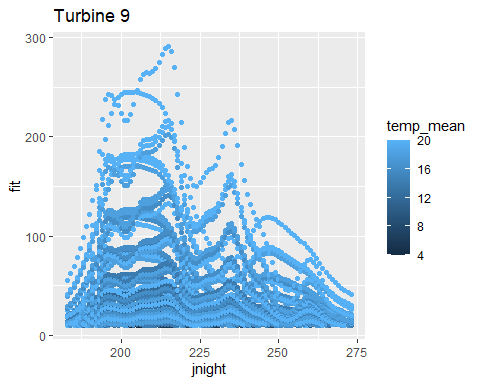
# Using colored points to represent average nightly wind speeds and temperatures... perhaps there is a more meaningful way to represent this.   
  
# Turbine 9 - wind   
fit2df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = wind\_mean)) + ggtitle("Turbine 9")



# Turbine 9 - temp   
fit2df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ggtitle("Turbine 9")

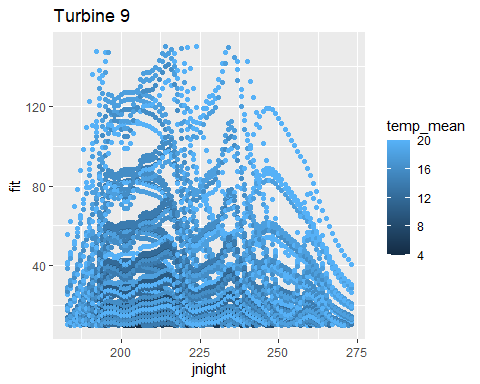


# See what this looks like for bat pass > 0   
# Turbine 9 - temp   
fit2df %>% dplyr::filter(fit > 10) %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ggtitle("Turbine 9")

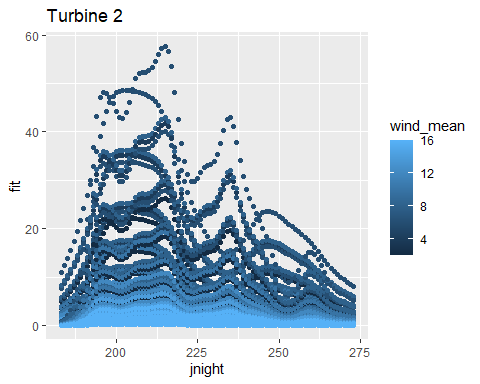


fit2df %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ylim(c(10,150)) + ggtitle("Turbine 9")

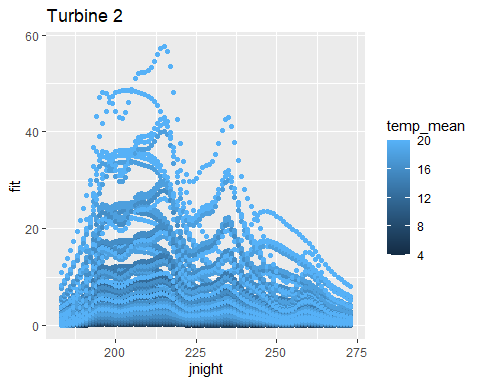
## Warning: Removed 7197 rows containing missing values (`geom\_point()`).



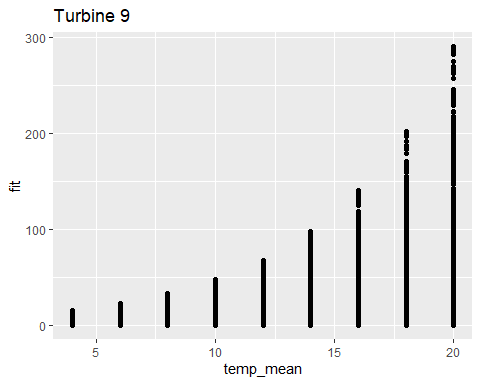
## I want to try and make this figure but make it a line plot with predictions for a series of temperatures... .  
  
# -----------------------------------------------------------------------------  
  
# Turbine 2 - wind  
fit2adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = wind\_mean)) + ggtitle ("Turbine 2")



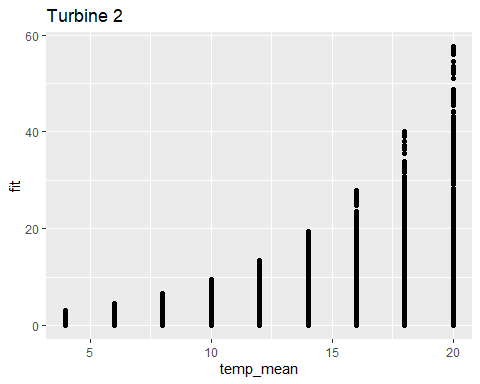
# Turbine 2 - temp   
fit2adf %>% ggplot(aes(x=jnight, y=fit)) +  
 geom\_point(aes(color = temp\_mean)) + ggtitle ("Turbine 2")



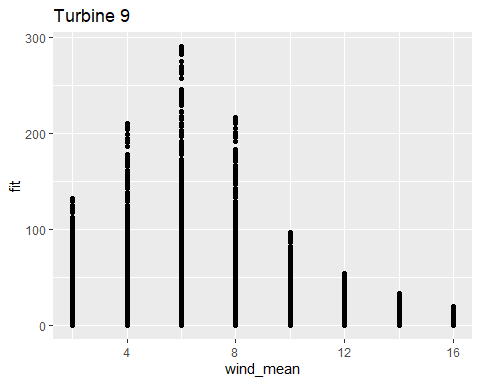
# -----------------------------------------------------------------------------  
  
# Temp is the response variable   
# Turbine 9  
fit2df %>% ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point() + ggtitle("Turbine 9")



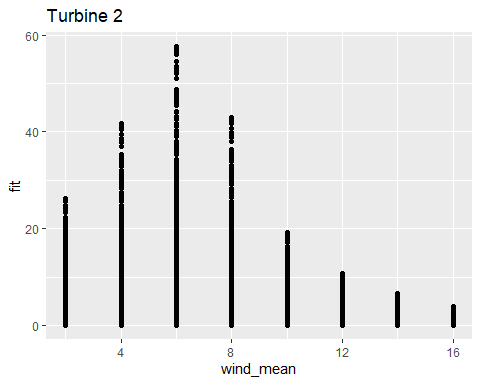
# Turbine 2  
fit2adf %>% ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point() + ggtitle ("Turbine 2")



# Wind is the response variable   
# Turbine 9  
fit2df %>% ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point() + ggtitle("Turbine 9")



# Turbine 2  
fit2adf %>% ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point() + ggtitle ("Turbine 2")



## These last few plots are a bit more helpful.

|  |
| --- |
| ## Katrine’s suggestion/code for visualizing different wind speeds and temperatures ### count data predictions |
| ```r # For plotting temp |
| newdata\_temp<- expand.grid(temp\_mean = c(4, 10, 16, 20), wind\_mean = seq(min(dfwind\_mean), length = 300), Habitat = levels(df$Habitat), Locality = “Turbine9”, jnight = 212) |
| # Turbine 9, fewer wind and temp values fit\_temp <- predict.gam(m2, newdata = newdata\_temp, se.fit=TRUE, type = “response”) fit\_temp\_df <- data.frame(newdata\_temp, fit\_temp) fit\_temp\_df <- transform(fit\_temp\_df, upper = fit + (2*se.fit), lower = fit-(2*se.fit)) summary(fit\_temp\_df) ``` |
| ## temp\_mean wind\_mean Habitat Locality ## Min. : 4.0 Min. : 2.571 Natural :1200 Turbine9:2400 ## 1st Qu.: 8.5 1st Qu.: 5.281 TurbinePad:1200 ## Median :13.0 Median : 7.992 ## Mean :12.5 Mean : 7.992 ## 3rd Qu.:17.0 3rd Qu.:10.702 ## Max. :20.0 Max. :13.413 ## jnight fit se.fit upper ## Min. :212 Min. : 1.703 Min. : 0.8678 Min. : 3.439 ## 1st Qu.:212 1st Qu.: 11.509 1st Qu.: 3.7326 1st Qu.: 18.943 ## Median :212 Median : 33.590 Median : 8.3395 Median : 50.376 ## Mean :212 Mean : 62.133 Mean :13.8779 Mean : 89.889 ## 3rd Qu.:212 3rd Qu.: 93.192 3rd Qu.:18.7007 3rd Qu.:129.890 ## Max. :212 Max. :276.947 Max. :60.6374 Max. :398.212 ## lower ## Min. : -0.07348 ## 1st Qu.: 3.75640 ## Median : 15.53426 ## Mean : 34.37710 ## 3rd Qu.: 55.98197 ## Max. :155.69591 |
| r # Plotting temp at various wind speeds new\_labels <- c("4" = "temp 4", "10" = "temp 10", "16" = "temp 16", "20" = "temp 20") # Using binary response model, turbine 9 predictions plt1 <- fit\_temp\_df %>% ggplot( aes(x=wind\_mean, y = fit, group = factor(temp\_mean))) + geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5) + geom\_line() + facet\_wrap(~ temp\_mean, scales ='fixed', labeller = labeller(temp\_mean = new\_labels)) + labs(x = "Average wind speed m/s", y ="Proportion of hours per night with batpass>0") plt1 |
|  |
| ```r ## —————————————————————————- |
| ## For plotting wind summary(df$temp\_mean) ``` |
| ## Min. 1st Qu. Median Mean 3rd Qu. Max. ## 8.789 13.419 14.930 15.610 17.494 24.579 |
| ```r # Turbine 9, fewer temp and wind variables newdata\_wind<- expand.grid(temp\_mean = seq(min (dftemp\_mean), length = 300), wind\_mean = c(2, 8, 12, 16), Habitat = levels(df$Habitat), Locality = “Turbine9”, jnight = 212) |
| # Turbine 9, fewer wind and temp values fit\_wind <- predict.gam(m2, newdata = newdata\_wind, se.fit=TRUE, type = “response”) fit\_wind\_df <- data.frame(newdata\_wind, fit\_wind) fit\_wind\_df <- transform(fit\_wind\_df, upper = fit + (2*se.fit), lower = fit-(2*se.fit)) summary(fit\_wind\_df) ``` |
| ## temp\_mean wind\_mean Habitat Locality ## Min. : 8.789 Min. : 2.0 Natural :1200 Turbine9:2400 ## 1st Qu.:12.737 1st Qu.: 6.5 TurbinePad:1200 ## Median :16.684 Median :10.0 ## Mean :16.684 Mean : 9.5 ## 3rd Qu.:20.631 3rd Qu.:13.0 ## Max. :24.579 Max. :16.0 ## jnight fit se.fit upper ## Min. :212 Min. : 2.096 Min. : 1.525 Min. : 5.146 ## 1st Qu.:212 1st Qu.: 15.376 1st Qu.: 6.354 1st Qu.: 29.245 ## Median :212 Median : 35.785 Median : 11.972 Median : 61.682 ## Mean :212 Mean : 70.176 Mean : 19.650 Mean :109.475 ## 3rd Qu.:212 3rd Qu.: 89.681 3rd Qu.: 24.403 3rd Qu.:137.701 ## Max. :212 Max. :470.127 Max. :136.472 Max. :743.071 ## lower ## Min. :-16.179 ## 1st Qu.: 1.052 ## Median : 14.102 ## Mean : 30.876 ## 3rd Qu.: 44.108 ## Max. :197.183 |
| ```r ## Plotting wind speeds at various temperatures |
| new\_labels <- c(“2” = “wind 2 m/s”, “8” = “wind 8 m/s”, “12” = “wind 12 /s”, “16” = “wind 16 m/s”) |
| # Using binary response model, turbine 9 predictions plt2 <- fit\_wind\_df %>% ggplot(aes(x=temp\_mean, y = fit, group = factor(wind\_mean)), fill = ‘grey’, alpha = 0.5) + geom\_line() + facet\_wrap(~ wind\_mean, scales =‘fixed’, labeller = labeller(wind\_mean = new\_labels)) + labs(x = “Average temperature”, y =“Proportion of hours per night with batpass>0”) plt2 ``` |
|  |
| ## For knitting |
| r setwd(output\_today) # knit the document getwd() |
| ## [1] "C:/Users/apmc/OneDrive - Norwegian University of Life Sciences/2. Marker 2019-2020/Marker 2022/SecondDraftAnalyses/Reed/Outputs/Step3.Marker bat acoustics models draft 2\_2022-12-14" |

Only for the sake of testing - use a similar model that does not have an interaction between jnight and habitat to see if the results are similar.