Marker bat acoustic models draft 2

2022-11-28

### Notes from meeting with Katrine 29.11.2022

## 1. For the total bat data set, with y = binary nightly bat activity

1. Adjust the k value for the bats\_tot model to find one that better suits the data
2. Illustrate the wind/temperature cutoffs.

## At some point, it may be necessary to re-aggregate the weather data to Facility rather than across both facilities

# 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"  
  
  
  
### Here is some example code from Pierre that I would like to implement soon which allows us to adjust all the directories based on the user  
  
# if(Sys.info()['user'] == 'pidu') { ## Pierre  
# gitDir <- 'C:/myDocuments/AlpineWolf'  
# dataDir <- 'C:/Users/pidu/Dropbox (AQEG)/AQEG Team Folder/AlpineWolf/01\_Data'   
# analysisDir <- 'C:/Users/pidu/Dropbox (AQEG)/AQEG Team Folder/AlpineWolf/02\_Analysis'  
# simulationDir <- 'C:/Users/pidu/Dropbox (AQEG)/AQEG Team Folder/AlpineWolf/03\_Simulations'  
# meetingDir <- 'C:/Users/pidu/Dropbox (AQEG)/AQEG Team Folder/AlpineWolf/04\_Meetings'  
# reportDir <- 'C:/Users/pidu/Dropbox (AQEG)/AQEG Team Folder/AlpineWolf/06\_Report'  
# } else if(Sys.info()['user'] == 'virginia') { ## Virginia  
# gitDir <- '/Users/virginia/Dropbox/Mac/Documents/GitHub/AlpineWolf'  
# dataDir <- '/Users/virginia/Dropbox/AlpineWolf/01\_Data'  
# analysisDir <- '/Users/virginia/Dropbox/AlpineWolf/02\_Analysis'  
# simulationDir <- '/Users/virginia/Dropbox/AlpineWolf/03\_Simulations'  
# meetingDir <- '/Users/virginia/Dropbox/AlpineWolf/04\_Meetings'  
# reportDir <- '/Users/virginia/Dropbox/AlpineWolf/06\_Report'  
#   
  
##########################################################  
#### 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   
  
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>

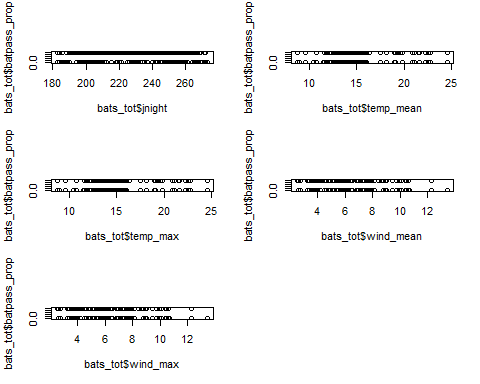
The “bats\_tot” dataset Nightly aggregated total bat activity

# 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  
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   
##

Fitting he full model (I recommend not making the model more complex than this) Will have to do trial-and-error to get the k right

m1 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") + #bs = "re" is equivalent to adding Locality as random intercept in a mixed model (GAMM)  
 s(jnight, by = Habitat, bs = "gp", k =90) + #allows for separate shapes of relationships with jnight for each habitat, bs = "gp" takes care of temporal autocorrelation  
 Habitat + #main effect of habitat (testing if means differ between habitats)  
 s(temp\_mean) + #does not seem to make much difference whether I use max or mean temp, so I chose mean (feels less like cherry-picking, I think...)  
 s(wind\_mean) + #same as for temp  
 ti(temp\_mean,wind\_mean) , #interaction between temp and wind  
 data = bats\_tot, method = "REML", #REML is not default, but is highly recommended by experts  
 family = binomial, select=TRUE) #may have to shift to quasibinomial if overdispersion, select = TRUE gives you automatic model selection  
  
summary(m1) # not \*quite\* all terms significant - not Locality or jnight\*Habitat

##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + s(temp\_mean) +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.03617 0.22151 -0.163 0.87   
## HabitatNatural 2.55800 0.29955 8.539 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.24277 0.46442 6.982 2.9e-12 \*\*\*  
## ---  
## 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) 0.3433791 6 0.358 0.386901   
## s(jnight):HabitatMeteorologicalTower 0.0000476 72 0.000 0.877886   
## s(jnight):HabitatNatural 2.5330933 87 8.624 0.012154 \*   
## s(jnight):HabitatTurbinePad 5.6185246 88 40.312 < 2e-16 \*\*\*  
## s(temp\_mean) 2.2539415 9 20.685 1.02e-06 \*\*\*  
## s(wind\_mean) 3.2134647 9 20.946 7.11e-06 \*\*\*  
## ti(temp\_mean,wind\_mean) 4.8139882 16 17.980 0.000105 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.369 Deviance explained = 37.5%  
## -REML = 319.49 Scale est. = 1 n = 951

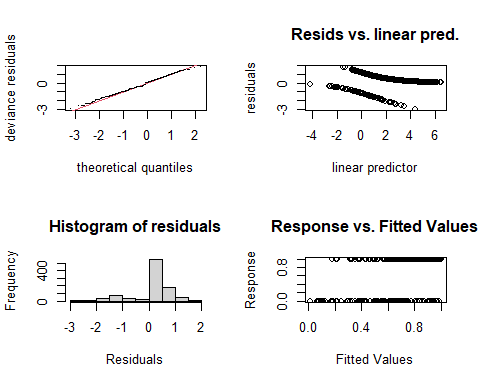
# output below

### Katrine:

The reason why I use GAM with a rather simple random effect instead of GAMM, is that there are fewer options (e.g., distributions) available in GAMM

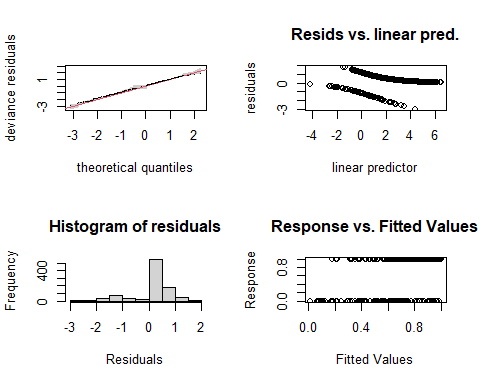
Model validation (may also use dHARMA to get more fancy validation plots) Note that validation if binomial models is usually a pain and does not give meaningful validation plots However, it works in our case, because of the aggregated nature of the data :-) I found a similar example in <https://www.youtube.com/watch?v=sgw4cu8hrZM&t=4038s>

par(mfrow = c(2,2))  
gam.check(m1) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 21 iterations.  
## Gradient range [-3.297958e-05,1.583296e-05]  
## (score 319.4884 & scale 1).  
## Hessian positive definite, eigenvalue range [2.344473e-06,0.7420669].  
## Model rank = 312 / 312   
##   
## 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) 8.00e+00 3.43e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 4.76e-05 0.95 0.225   
## s(jnight):HabitatNatural 8.90e+01 2.53e+00 0.95 0.165   
## s(jnight):HabitatTurbinePad 8.90e+01 5.62e+00 0.95 0.220   
## s(temp\_mean) 9.00e+00 2.25e+00 0.96 0.400   
## s(wind\_mean) 9.00e+00 3.21e+00 0.94 0.075 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 4.81e+00 0.85 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m1, 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 21 iterations.  
## Gradient range [-3.297958e-05,1.583296e-05]  
## (score 319.4884 & scale 1).  
## Hessian positive definite, eigenvalue range [2.344473e-06,0.7420669].  
## Model rank = 312 / 312   
##   
## 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) 8.00e+00 3.43e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 4.76e-05 0.95 0.20   
## s(jnight):HabitatNatural 8.90e+01 2.53e+00 0.95 0.24   
## s(jnight):HabitatTurbinePad 8.90e+01 5.62e+00 0.95 0.23   
## s(temp\_mean) 9.00e+00 2.25e+00 0.96 0.35   
## s(wind\_mean) 9.00e+00 3.21e+00 0.94 0.10 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 4.81e+00 0.85 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

overdispersion.m1 <- sum( residuals(m1, "pearson")^2 ) / m1$df.residual  
overdispersion.m1

## [1] 0.80653

# 0.80653 - looking good!   
  
## Not overdispersed but the k values are off.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

The k-index should be approx. 1 or larger Play play around with k….

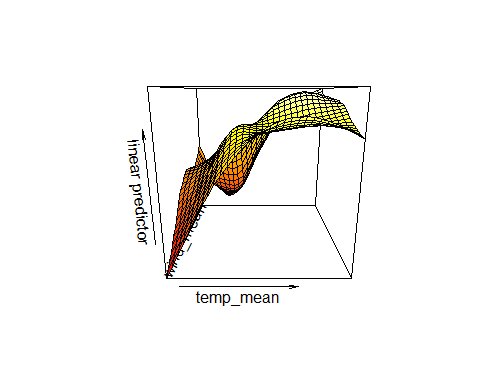
bmp(file.path(output\_today, "checking k dimensions m1 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m1, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m1, scales ="free") #function 'draw' is from package 'gratia'  
draw(m1, scales ="fixed")  
  
dev.off()

## png   
## 2

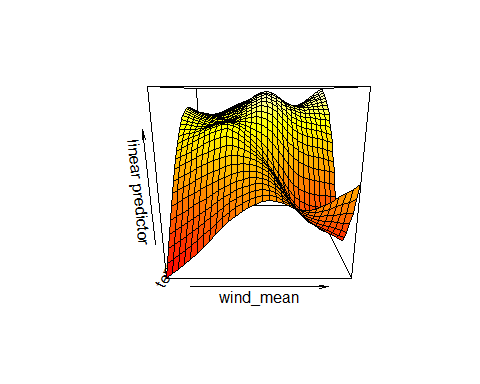
## Visualize the 3D GAM  
bmp(file.path(output\_today, "3D gam m1 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
  
vis.gam(m1, view = c("temp\_mean", "wind\_mean")) #Not a very elegant plot, you can find prettier solutions, but helps understand the interaction between temp and wind  
vis.gam(m1, view = c("wind\_mean", "temp\_mean"))   
  
dev.off()

## png   
## 2

vis.gam(m1, view = c("temp\_mean", "wind\_mean"))



vis.gam(m1, view = c("wind\_mean", "temp\_mean"))



# m2

### model temperature as linear effect

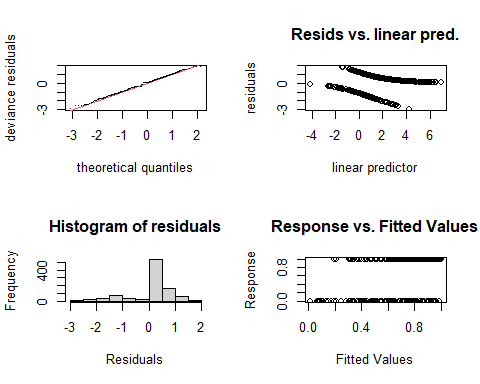
m2 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k =90) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
summary(m2)

##   
## 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) + ti(temp\_mean, wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.55691 1.17998 -3.862 0.000113 \*\*\*  
## HabitatNatural 2.56588 0.30225 8.489 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.28247 0.48768 6.731 1.69e-11 \*\*\*  
## temp\_mean 0.29795 0.07437 4.006 6.17e-05 \*\*\*  
## ---  
## 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) 3.408e-01 6 0.356 0.386191   
## s(jnight):HabitatMeteorologicalTower 6.452e-05 77 0.000 0.887360   
## s(jnight):HabitatNatural 2.843e+00 87 10.464 0.006340 \*\*   
## s(jnight):HabitatTurbinePad 6.074e+00 88 40.997 < 2e-16 \*\*\*  
## s(wind\_mean) 3.236e+00 9 16.919 4.24e-05 \*\*\*  
## ti(temp\_mean,wind\_mean) 5.346e+00 16 18.645 0.000382 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.365 Deviance explained = 37.4%  
## -REML = 320.47 Scale est. = 1 n = 951

Only ti(temp\_mean, wind\_mean) is significant…

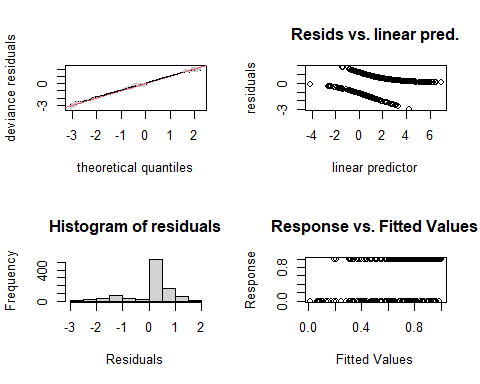
### Basis dimension (k) checking results:

par(mfrow = c(2,2))  
gam.check(m2)



##   
## Method: REML Optimizer: outer newton  
## full convergence after 18 iterations.  
## Gradient range [-9.303033e-05,4.008943e-05]  
## (score 320.4746 & scale 1).  
## Hessian positive definite, eigenvalue range [2.88257e-06,0.7431721].  
## Model rank = 304 / 304   
##   
## 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) 8.00e+00 3.41e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 6.45e-05 0.95 0.195   
## s(jnight):HabitatNatural 8.90e+01 2.84e+00 0.95 0.190   
## s(jnight):HabitatTurbinePad 8.90e+01 6.07e+00 0.95 0.210   
## s(wind\_mean) 9.00e+00 3.24e+00 0.94 0.075 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.35e+00 0.85 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m2, rep=500)



##   
## Method: REML Optimizer: outer newton  
## full convergence after 18 iterations.  
## Gradient range [-9.303033e-05,4.008943e-05]  
## (score 320.4746 & scale 1).  
## Hessian positive definite, eigenvalue range [2.88257e-06,0.7431721].  
## Model rank = 304 / 304   
##   
## 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) 8.00e+00 3.41e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 6.45e-05 0.95 0.20   
## s(jnight):HabitatNatural 8.90e+01 2.84e+00 0.95 0.18   
## s(jnight):HabitatTurbinePad 8.90e+01 6.07e+00 0.95 0.19   
## s(wind\_mean) 9.00e+00 3.24e+00 0.94 0.10 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.35e+00 0.85 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Interpreting gam.check() output  
## First check if the model converges - this does after 18 iterations!  
## If it didn't then the results would not be meaningful   
## Now each smooth term is evaluated to see if the residuals of these terms are randomly distributed.   
## If the p value is low for a smooth term, then the residuals are not random and most likely the K value should be increased.   
## In this case, the p value of tensor product interaction term (ti) for temp and mean is very low and the current k value is 16. Notice that the other k values on the smooth terms are either 9 or close to 90.   
## Tensor product smooths and interaction, playing with k for these types of terms in not the same as for normal smooth terms... need to specify both the k-values i.e. k = c(4,5)   
  
## quick rundown on some of the lexicon behind tensor terms:   
# te() is like x\*Z or x + z + x:z if you want to write it all out  
# ti() is like X:Z only  
##   
  
overdispersion.m2 <- sum( residuals(m2, "pearson")^2 ) / m2$df.residual  
overdispersion.m2

## [1] 0.7804294

# Now at [1] 0.7804294, a bit under dispersed.

### Basis dimension (k) checking results:

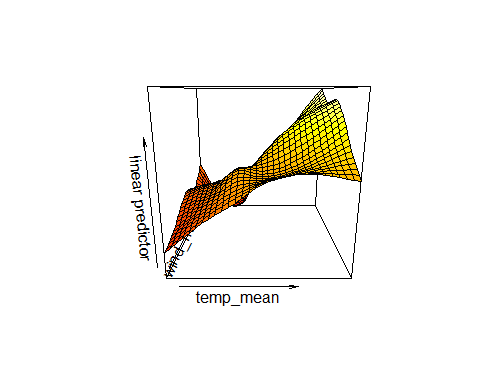
bmp(file.path(output\_today, "checking k dimensions m2 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m2, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m2, scales ="free") #function 'draw' is from package 'gratia'  
draw(m2, scales ="fixed")  
  
dev.off()

## png   
## 2

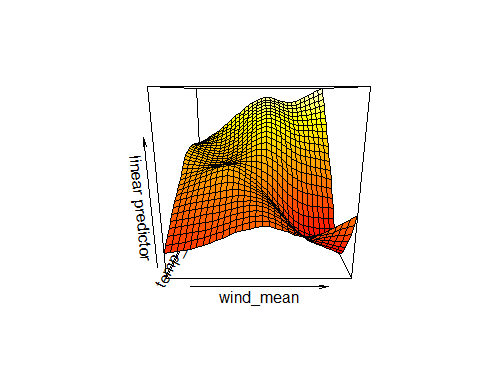
## Visualize the 3D GAM  
bmp(file.path(output\_today, "3D gam m2 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m2, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m2, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m2, view = c("temp\_mean", "wind\_mean"))



vis.gam(m2, view = c("wind\_mean", "temp\_mean"))



## Recap

This model is more under dispersed than the last one and the issue of an ill-fitting k value is still there.

summary(m1) # R-sq.(adj) = 0.365 Deviance explained = 37.4% summary(m2) # R-sq.(adj) = 0.369 Deviance explained = 37.5%

However, this doesn’t seems to make a huge difference in the overall fit of the model to the data

Changing the the mean temperature term to a linear variable doesn’t appear to make a huge difference on the model fit, if anything it is a slightly poorer fit of the data. So I will keep this as a smooth term and adjust k values on the ti(temp\_mean, wind\_mean term instead)

Some notes from Katrine:

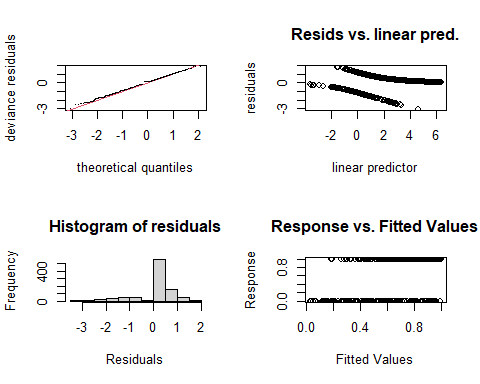
# m3

### Model temperature as linear effect, assign higher k values to the ti() term

m3 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k =90) +   
 Habitat +   
 s(temp\_mean) +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(10, 10)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
summary(m3)

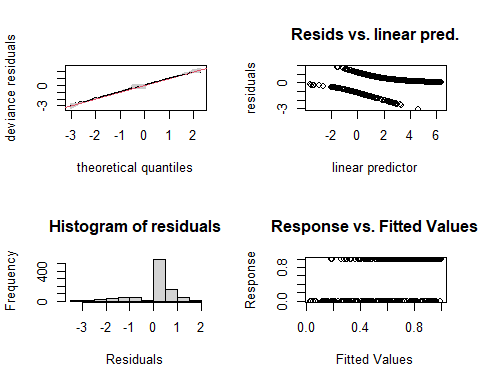
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + s(temp\_mean) +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean, k = c(10, 10))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.07275 0.27487 -0.265 0.791   
## HabitatNatural 2.59932 0.35530 7.316 2.56e-13 \*\*\*  
## HabitatTurbinePad 3.26354 0.48562 6.720 1.81e-11 \*\*\*  
## ---  
## 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) 1.474e+00 6 1.859 0.2680   
## s(jnight):HabitatMeteorologicalTower 5.361e-06 70 0.000 0.4510   
## s(jnight):HabitatNatural 1.684e+00 87 6.149 0.0191 \*   
## s(jnight):HabitatTurbinePad 5.154e+00 88 37.284 2.11e-06 \*\*\*  
## s(temp\_mean) 3.037e+00 9 32.109 < 2e-16 \*\*\*  
## s(wind\_mean) 4.350e+00 9 36.591 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.830e+00 70 27.342 2.24e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.388 Deviance explained = 38.9%  
## -REML = 314.34 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m3) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 30 iterations.  
## Gradient range [-3.892589e-05,1.566755e-05]  
## (score 314.339 & scale 1).  
## Hessian positive definite, eigenvalue range [4.443067e-07,1.086379].  
## Model rank = 368 / 368   
##   
## 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) 8.00e+00 1.47e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 5.36e-06 0.97 0.38   
## s(jnight):HabitatNatural 8.90e+01 1.68e+00 0.97 0.41   
## s(jnight):HabitatTurbinePad 8.90e+01 5.15e+00 0.97 0.44   
## s(temp\_mean) 9.00e+00 3.04e+00 0.99 0.56   
## s(wind\_mean) 9.00e+00 4.35e+00 0.96 0.31   
## ti(temp\_mean,wind\_mean) 7.20e+01 2.83e+00 0.87 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m3, 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 30 iterations.  
## Gradient range [-3.892589e-05,1.566755e-05]  
## (score 314.339 & scale 1).  
## Hessian positive definite, eigenvalue range [4.443067e-07,1.086379].  
## Model rank = 368 / 368   
##   
## 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) 8.00e+00 1.47e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 5.36e-06 0.97 0.37   
## s(jnight):HabitatNatural 8.90e+01 1.68e+00 0.97 0.34   
## s(jnight):HabitatTurbinePad 8.90e+01 5.15e+00 0.97 0.38   
## s(temp\_mean) 9.00e+00 3.04e+00 0.99 0.56   
## s(wind\_mean) 9.00e+00 4.35e+00 0.96 0.26   
## ti(temp\_mean,wind\_mean) 7.20e+01 2.83e+00 0.87 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m3 <- sum( residuals(m3, "pearson")^2 ) / m3$df.residual  
overdispersion.m3

## [1] 0.8053636

# 0.8053636 - looking good! However, the k index on the ti() term is still off.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

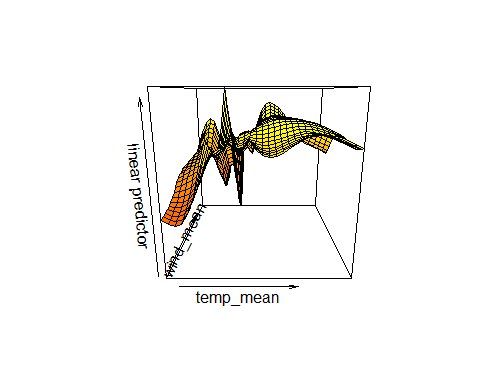
#plot(m1, pages=1,scheme=2,shade=TRUE)  
bmp(file.path(output\_today, "checking k dimensions m3 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m3, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m3, scales ="free") #function 'draw' is from package 'gratia'  
draw(m3, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

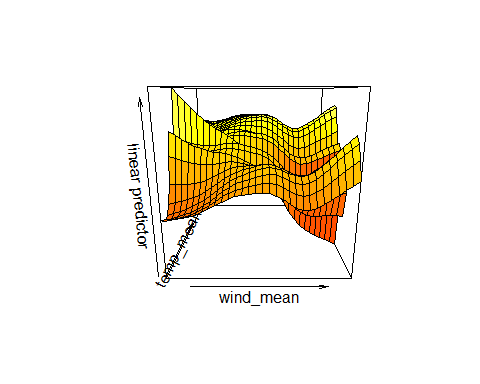
bmp(file.path(output\_today, "3D gam m3 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m3, view = c("temp\_mean", "wind\_mean")) #Not a very elegant plot, you can find prettier solutions, but helps understand the interaction between tempp and wind  
vis.gam(m3, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m3, view = c("temp\_mean", "wind\_mean")) #Not a very elegant plot, you can find prettier solutions, but helps understand the interaction between tempp and wind



vis.gam(m3, view = c("wind\_mean", "temp\_mean"))



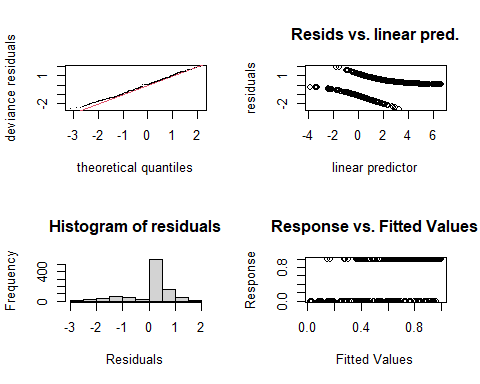
# m4

### Try once more with slightly lower k values for the ti() term

m4 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k =90) +   
 Habitat +   
 s(temp\_mean) +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(15, 15)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
summary(m4) # R-sq.(adj) = 0.407 Deviance explained = 43%

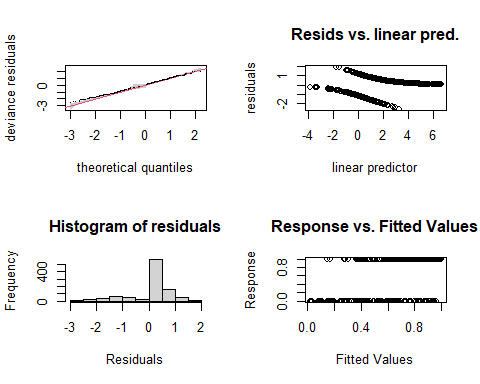
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + s(temp\_mean) +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean, k = c(15, 15))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.1568 0.2604 -0.602 0.547   
## HabitatNatural 2.6544 0.3340 7.947 1.91e-15 \*\*\*  
## HabitatTurbinePad 3.2914 0.4491 7.329 2.32e-13 \*\*\*  
## ---  
## 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) 9.875e-01 6 1.119 0.33311   
## s(jnight):HabitatMeteorologicalTower 1.169e-04 69 0.000 0.84483   
## s(jnight):HabitatNatural 1.129e+00 87 2.327 0.09823 .   
## s(jnight):HabitatTurbinePad 4.375e+00 88 28.914 < 2e-16 \*\*\*  
## s(temp\_mean) 2.858e+00 9 33.482 < 2e-16 \*\*\*  
## s(wind\_mean) 1.887e+00 9 8.537 0.00181 \*\*   
## ti(temp\_mean,wind\_mean) 2.408e+01 72 63.466 7e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.407 Deviance explained = 43%  
## -REML = 312.94 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m4) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 13 iterations.  
## Gradient range [-5.112826e-05,8.448452e-06]  
## (score 312.9422 & scale 1).  
## Hessian positive definite, eigenvalue range [6.540159e-06,2.140766].  
## Model rank = 368 / 368   
##   
## 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) 8.00e+00 9.88e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 1.17e-04 1.01 0.85   
## s(jnight):HabitatNatural 8.90e+01 1.13e+00 1.01 0.90   
## s(jnight):HabitatTurbinePad 8.90e+01 4.37e+00 1.01 0.88   
## s(temp\_mean) 9.00e+00 2.86e+00 1.02 0.92   
## s(wind\_mean) 9.00e+00 1.89e+00 1.00 0.80   
## ti(temp\_mean,wind\_mean) 7.20e+01 2.41e+01 0.91 0.02 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m4, 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 13 iterations.  
## Gradient range [-5.112826e-05,8.448452e-06]  
## (score 312.9422 & scale 1).  
## Hessian positive definite, eigenvalue range [6.540159e-06,2.140766].  
## Model rank = 368 / 368   
##   
## 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) 8.00e+00 9.88e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 8.90e+01 1.17e-04 1.01 0.86   
## s(jnight):HabitatNatural 8.90e+01 1.13e+00 1.01 0.90   
## s(jnight):HabitatTurbinePad 8.90e+01 4.37e+00 1.01 0.88   
## s(temp\_mean) 9.00e+00 2.86e+00 1.02 0.94   
## s(wind\_mean) 9.00e+00 1.89e+00 1.00 0.80   
## ti(temp\_mean,wind\_mean) 7.20e+01 2.41e+01 0.91 0.02 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m4 <- sum( residuals(m4, "pearson")^2 ) / m4$df.residual  
overdispersion.m4

## [1] 0.5971965

# 0.5971965 - super duper under dispersed , but the fit of the k value is slightly better.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

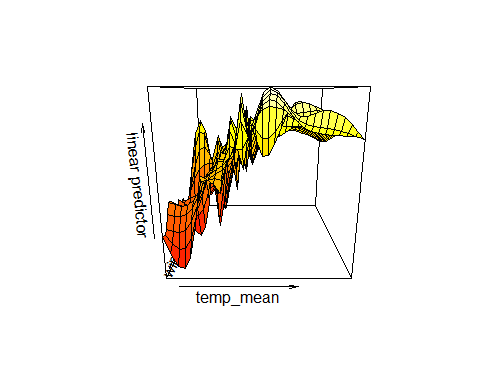
bmp(file.path(output\_today, "checking k dimensions m4 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m4, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m4, scales ="free") #function 'draw' is from package 'gratia'  
draw(m4, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

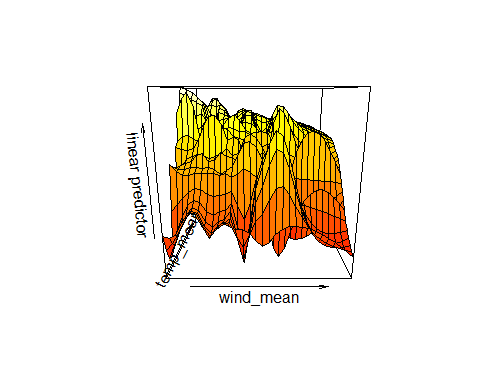
bmp(file.path(output\_today, "3D gam m4 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m4, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m4, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m4, view = c("temp\_mean", "wind\_mean"))



vis.gam(m4, view = c("wind\_mean", "temp\_mean"))



# m5

### Remove the met tower

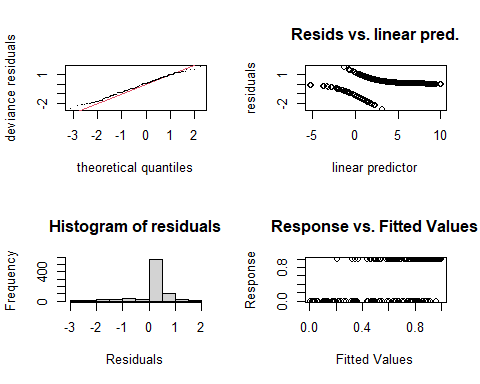
### Prune the dataset

bats\_ground <- bats\_tot %>% filter(Habitat != "MeteorologicalTower") %>% droplevels  
# 811 observations instead of 951

m5 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k =90) +   
 Habitat +   
 s(temp\_mean) +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(15, 15)) ,   
 data = bats\_ground, method = "REML",   
 family = binomial, select=TRUE)   
summary(m5) # R-sq.(adj) = 0.503 Deviance explained = 54.6%

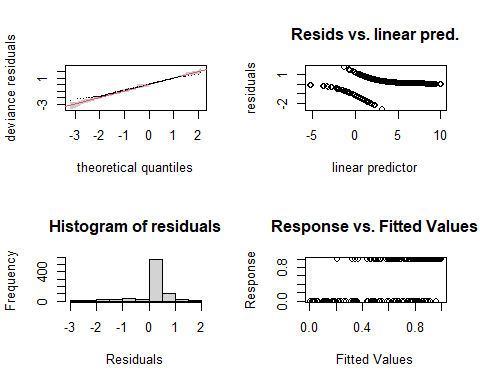
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + s(temp\_mean) +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean, k = c(15, 15))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 3.3114 0.5073 6.527 6.69e-11 \*\*\*  
## HabitatTurbinePad 0.5369 0.4275 1.256 0.209   
## ---  
## 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) 1.340 6 1.598 0.2994   
## s(jnight):HabitatNatural 1.917 87 5.047 0.0132 \*   
## s(jnight):HabitatTurbinePad 4.216 88 26.370 < 2e-16 \*\*\*  
## s(temp\_mean) 2.918 9 19.002 1.75e-06 \*\*\*  
## s(wind\_mean) 3.425 9 15.482 4.41e-05 \*\*\*  
## ti(temp\_mean,wind\_mean) 27.232 72 76.375 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.503 Deviance explained = 54.6%  
## -REML = 211.39 Scale est. = 1 n = 811

par(mfrow = c(2,2))  
gam.check(m5) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 12 iterations.  
## Gradient range [-5.428416e-05,3.214907e-05]  
## (score 211.3931 & scale 1).  
## Hessian positive definite, eigenvalue range [1.702042e-06,2.493665].  
## Model rank = 277 / 277   
##   
## 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 1.34 NA NA  
## s(jnight):HabitatNatural 89.00 1.92 1.05 0.99  
## s(jnight):HabitatTurbinePad 89.00 4.22 1.05 1.00  
## s(temp\_mean) 9.00 2.92 1.05 0.98  
## s(wind\_mean) 9.00 3.42 1.05 0.98  
## ti(temp\_mean,wind\_mean) 72.00 27.23 0.97 0.53

gam.check(m5, 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 12 iterations.  
## Gradient range [-5.428416e-05,3.214907e-05]  
## (score 211.3931 & scale 1).  
## Hessian positive definite, eigenvalue range [1.702042e-06,2.493665].  
## Model rank = 277 / 277   
##   
## 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 1.34 NA NA  
## s(jnight):HabitatNatural 89.00 1.92 1.05 0.98  
## s(jnight):HabitatTurbinePad 89.00 4.22 1.05 0.99  
## s(temp\_mean) 9.00 2.92 1.05 0.99  
## s(wind\_mean) 9.00 3.42 1.05 0.99  
## ti(temp\_mean,wind\_mean) 72.00 27.23 0.97 0.47

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m5 <- sum( residuals(m5, "pearson")^2 ) / m5$df.residual  
overdispersion.m5

## [1] 0.3749742

# 0.3749742 - super duper under dispersed , but the fit of the k value is much better.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

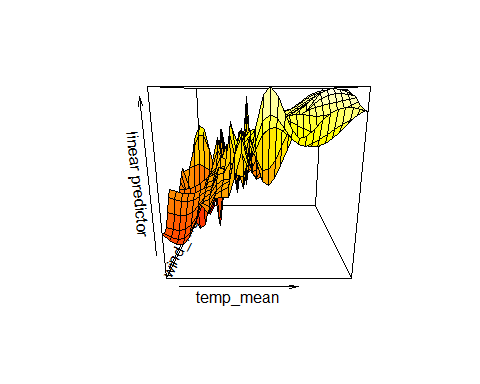
bmp(file.path(output\_today, "checking k dimensions m5 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m5, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m5, scales ="free") #function 'draw' is from package 'gratia'  
draw(m5, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

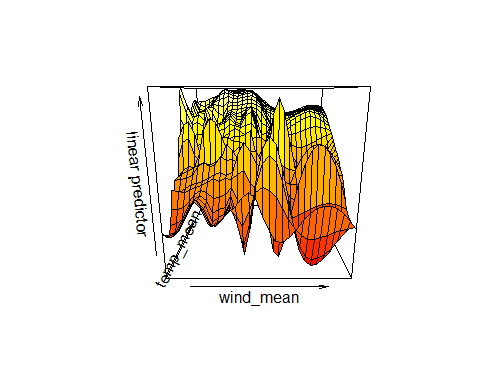
bmp(file.path(output\_today, "3D gam m5 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m5, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m5, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m5, view = c("temp\_mean", "wind\_mean"))



vis.gam(m5, view = c("wind\_mean", "temp\_mean"))



# m6

### Dataset does not include the met tower

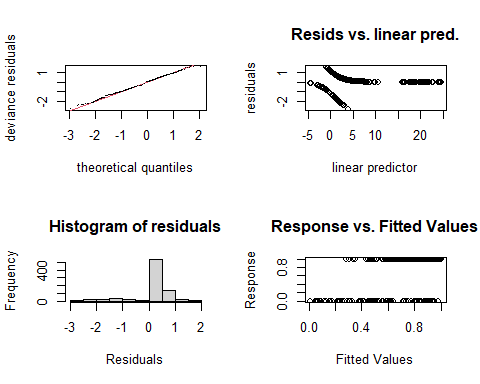
### Otherwise, the same model as m1

bats\_ground <- bats\_tot %>% filter(Habitat != "MeteorologicalTower") %>% droplevels  
# 811 observations instead of 951

m6 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k =90) +   
 Habitat +   
 s(temp\_mean) +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean) ,   
 data = bats\_ground, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m6) # R-sq.(adj) = 0.415 Deviance explained = 44.4%

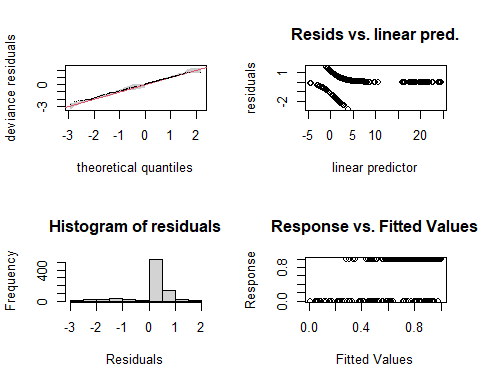
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 90) + Habitat + s(temp\_mean) +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 4.9092 2.2112 2.220 0.0264 \*  
## HabitatTurbinePad 0.6166 0.4621 1.334 0.1821   
## ---  
## 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) 1.685 6 2.192 0.2480   
## s(jnight):HabitatNatural 2.586 86 7.709 0.0219 \*   
## s(jnight):HabitatTurbinePad 5.084 87 36.285 1.51e-06 \*\*\*  
## s(temp\_mean) 6.249 9 32.676 < 2e-16 \*\*\*  
## s(wind\_mean) 5.413 9 27.644 1.65e-06 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.427 16 22.714 1.26e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.415 Deviance explained = 44.4%  
## -REML = 228.76 Scale est. = 1 n = 811

par(mfrow = c(2,2))  
gam.check(m6) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 16 iterations.  
## Gradient range [-8.270666e-05,7.882422e-05]  
## (score 228.7632 & scale 1).  
## Hessian positive definite, eigenvalue range [2.683698e-06,1.305434].  
## Model rank = 221 / 221   
##   
## 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 1.69 NA NA   
## s(jnight):HabitatNatural 89.00 2.59 0.91 0.020 \*   
## s(jnight):HabitatTurbinePad 89.00 5.08 0.91 0.045 \*   
## s(temp\_mean) 9.00 6.25 0.91 0.040 \*   
## s(wind\_mean) 9.00 5.41 0.90 0.035 \*   
## ti(temp\_mean,wind\_mean) 16.00 2.43 0.83 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m6, 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 16 iterations.  
## Gradient range [-8.270666e-05,7.882422e-05]  
## (score 228.7632 & scale 1).  
## Hessian positive definite, eigenvalue range [2.683698e-06,1.305434].  
## Model rank = 221 / 221   
##   
## 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 1.69 NA NA   
## s(jnight):HabitatNatural 89.00 2.59 0.91 0.040 \*   
## s(jnight):HabitatTurbinePad 89.00 5.08 0.91 0.040 \*   
## s(temp\_mean) 9.00 6.25 0.91 0.040 \*   
## s(wind\_mean) 9.00 5.41 0.90 0.025 \*   
## ti(temp\_mean,wind\_mean) 16.00 2.43 0.83 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m6 <- sum( residuals(m6, "pearson")^2 ) / m6$df.residual  
overdispersion.m6

## [1] 0.5615471

# 0.5615471 - super duper under dispersed , and the k indices are a mess. This is the worst so far.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

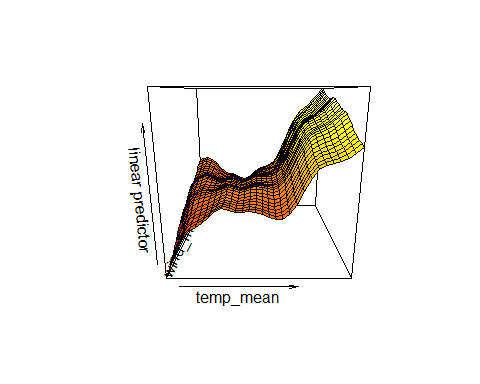
bmp(file.path(output\_today, "checking k dimensions m6 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m6, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m6, scales ="free") #function 'draw' is from package 'gratia'  
draw(m6, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

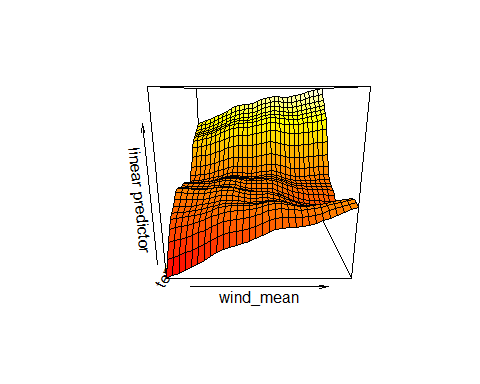
bmp(file.path(output\_today, "3D gam m6 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m6, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m6, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m6, view = c("temp\_mean", "wind\_mean"))



vis.gam(m6, view = c("wind\_mean", "temp\_mean"))



# m7

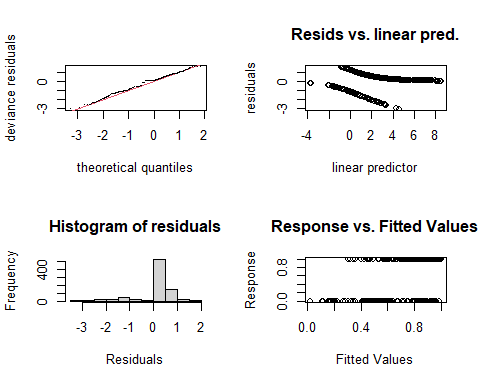
### Dataset does not include the met tower

### otherwise, remove adjust k term in the 3 night by habitat term

m7 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 s(temp\_mean) +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean) ,   
 data = bats\_ground, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m7) # R-sq.(adj) = 0.359 Deviance explained = 37.3%

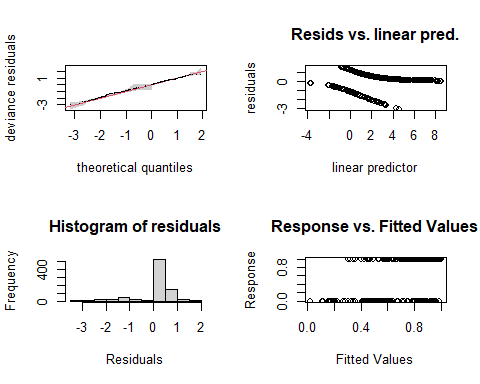
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp") + Habitat + s(temp\_mean) + s(wind\_mean) +   
## ti(temp\_mean, wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 2.7783 0.2671 10.400 <2e-16 \*\*\*  
## HabitatTurbinePad 0.6229 0.4391 1.418 0.156   
## ---  
## 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) 1.3965 6 1.719 0.280553   
## s(jnight):HabitatNatural 2.0686 11 6.138 0.031141 \*   
## s(jnight):HabitatTurbinePad 4.6336 11 36.984 < 2e-16 \*\*\*  
## s(temp\_mean) 0.9543 9 18.662 1.89e-06 \*\*\*  
## s(wind\_mean) 3.7346 9 35.613 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.2593 16 14.002 0.000112 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.359 Deviance explained = 37.3%  
## -REML = 226.05 Scale est. = 1 n = 811

par(mfrow = c(2,2))  
gam.check(m7) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 15 iterations.  
## Gradient range [-5.039703e-05,1.13311e-05]  
## (score 226.0539 & scale 1).  
## Hessian positive definite, eigenvalue range [9.407434e-07,0.7386722].  
## Model rank = 65 / 65   
##   
## 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) 7.000 1.397 NA NA   
## s(jnight):HabitatNatural 11.000 2.069 0.86 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 11.000 4.634 0.86 <2e-16 \*\*\*  
## s(temp\_mean) 9.000 0.954 0.85 <2e-16 \*\*\*  
## s(wind\_mean) 9.000 3.735 0.84 <2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 16.000 2.259 0.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m7, 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 15 iterations.  
## Gradient range [-5.039703e-05,1.13311e-05]  
## (score 226.0539 & scale 1).  
## Hessian positive definite, eigenvalue range [9.407434e-07,0.7386722].  
## Model rank = 65 / 65   
##   
## 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) 7.000 1.397 NA NA   
## s(jnight):HabitatNatural 11.000 2.069 0.86 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 11.000 4.634 0.86 <2e-16 \*\*\*  
## s(temp\_mean) 9.000 0.954 0.85 <2e-16 \*\*\*  
## s(wind\_mean) 9.000 3.735 0.84 <2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 16.000 2.259 0.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m7 <- sum( residuals(m7, "pearson")^2 ) / m7$df.residual  
overdispersion.m7

## [1] 0.826536

# 0.826536 - The k indices are a mess but it isn't under or overdispersed....   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

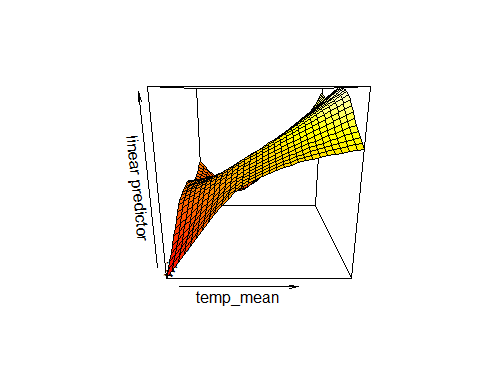
bmp(file.path(output\_today, "checking k dimensions m7 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m7, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m7, scales ="free") #function 'draw' is from package 'gratia'  
draw(m7, scales ="fixed")  
dev.off()

## png   
## 2

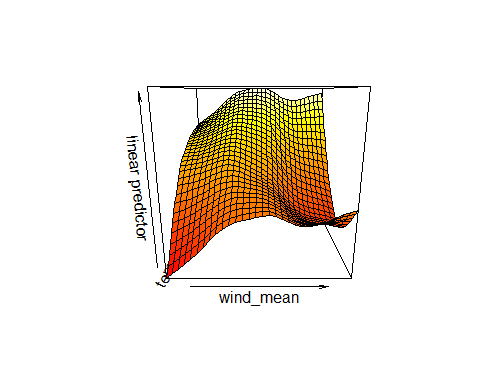
bmp(file.path(output\_today, "3D gam m7 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m7, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m7, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m7, view = c("temp\_mean", "wind\_mean"))



vis.gam(m7, view = c("wind\_mean", "temp\_mean"))



|  |
| --- |
| ## Recap |
| Ultimately the goal for this model group is to determine how weather patterns influence bat activity across the park. Therefore, it is better to include the met tower data. So pruning the dataset in this way is not helpful. |
| I could adjust high winds or temperatures to a certain cutoff to reduce some variability there but I am not ready to do that just yet. Instead, I am better there is a way to adjust the k values systematically across the smooth terms. I will go back to using temperature as a linear term, as it basically behaves this way naturally. |

# m8

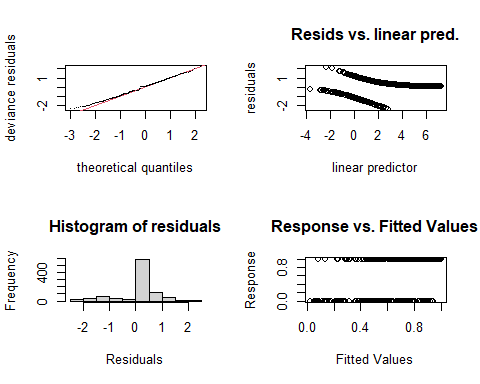
### Bats total dataset

### Adjust the k value of s(jnight, by = Habitat, bs = “gp”) to k = 15

m8 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 15) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(20,20)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m8)

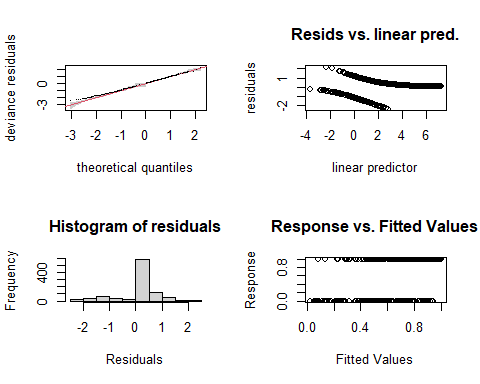
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 15) + Habitat + temp\_mean +   
## s(wind\_mean) + ti(temp\_mean, wind\_mean, k = c(20, 20))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.72566 1.06776 -3.489 0.000484 \*\*\*  
## HabitatNatural 2.60645 0.44985 5.794 6.87e-09 \*\*\*  
## HabitatTurbinePad 3.31813 0.54469 6.092 1.12e-09 \*\*\*  
## temp\_mean 0.23908 0.06401 3.735 0.000188 \*\*\*  
## ---  
## 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) 2.789e+00 7 4.513 0.133   
## s(jnight):HabitatMeteorologicalTower 1.911e-04 14 0.000 0.638   
## s(jnight):HabitatNatural 3.692e-04 14 0.000 0.388   
## s(jnight):HabitatTurbinePad 4.350e+00 14 26.924 5.36e-06 \*\*\*  
## s(wind\_mean) 2.765e+00 9 21.573 5.26e-07 \*\*\*  
## ti(temp\_mean,wind\_mean) 3.028e+01 81 67.932 1.88e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.41 Deviance explained = 43.7%  
## -REML = 316.32 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m8) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-5.845178e-05,4.033779e-05]  
## (score 316.3188 & scale 1).  
## Hessian positive definite, eigenvalue range [7.064051e-07,2.194453].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.79e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.91e-04 1.01 0.830   
## s(jnight):HabitatNatural 1.40e+01 3.69e-04 1.01 0.875   
## s(jnight):HabitatTurbinePad 1.40e+01 4.35e+00 1.01 0.855   
## s(wind\_mean) 9.00e+00 2.77e+00 1.01 0.830   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.03e+01 0.91 0.005 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m8, 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 9 iterations.  
## Gradient range [-5.845178e-05,4.033779e-05]  
## (score 316.3188 & scale 1).  
## Hessian positive definite, eigenvalue range [7.064051e-07,2.194453].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.79e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.91e-04 1.01 0.890   
## s(jnight):HabitatNatural 1.40e+01 3.69e-04 1.01 0.865   
## s(jnight):HabitatTurbinePad 1.40e+01 4.35e+00 1.01 0.890   
## s(wind\_mean) 9.00e+00 2.77e+00 1.01 0.850   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.03e+01 0.91 0.005 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m8 <- sum( residuals(m8, "pearson")^2 ) / m8$df.residual  
overdispersion.m8

## [1] 0.5588888

# 0.5588888 - The k indices look okay, but still under dispersed  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

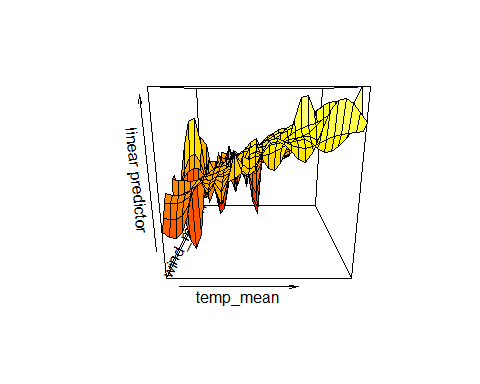
bmp(file.path(output\_today, "checking k dimensions m8 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m8, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m8, scales ="free") #function 'draw' is from package 'gratia'  
draw(m8, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

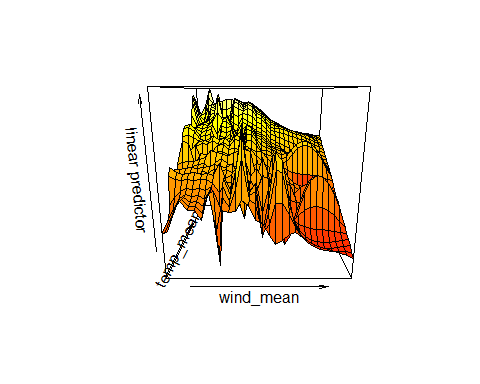
bmp(file.path(output\_today, "3D gam m8 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m8, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m8, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m8, view = c("temp\_mean", "wind\_mean"))



vis.gam(m8, view = c("wind\_mean", "temp\_mean"))



## Recap:

## This slightly improved the overall model fit, so I will keep this model but add another manual k term to the s(wind\_mean) term.

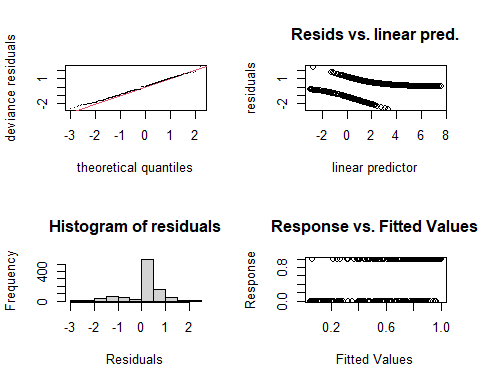
# m9

### Adjust the k value of s(jnight, by = Habitat, bs = “gp”) to k = 15

m9 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 15) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 20) +   
 ti(temp\_mean,wind\_mean, k = c(20,20)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m9)

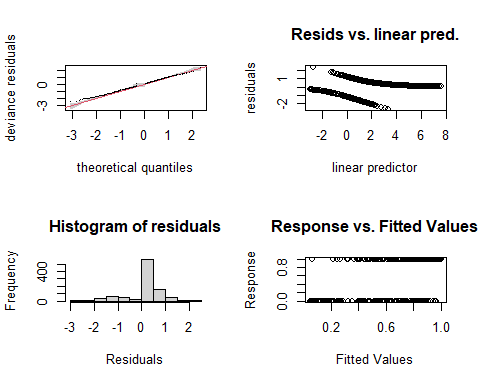
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 15) + Habitat + temp\_mean +   
## s(wind\_mean, k = 20) + ti(temp\_mean, wind\_mean, k = c(20,   
## 20))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.9096 1.1663 -2.495 0.01260 \*   
## HabitatNatural 2.7105 0.4208 6.442 1.18e-10 \*\*\*  
## HabitatTurbinePad 3.3548 0.5406 6.205 5.46e-10 \*\*\*  
## temp\_mean 0.1886 0.0713 2.645 0.00817 \*\*   
## ---  
## 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) 2.189e+00 7 3.072 0.20255   
## s(jnight):HabitatMeteorologicalTower 1.756e-04 14 0.000 0.52948   
## s(jnight):HabitatNatural 2.724e+00 14 9.238 0.00602 \*\*   
## s(jnight):HabitatTurbinePad 4.877e+00 14 35.210 < 2e-16 \*\*\*  
## s(wind\_mean) 8.380e+00 19 49.190 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.286e+01 71 46.878 4.71e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.432 Deviance explained = 45.3%  
## -REML = 318.05 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m9) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 10 iterations.  
## Gradient range [-4.44288e-05,1.328535e-05]  
## (score 318.0472 & scale 1).  
## Hessian positive definite, eigenvalue range [5.91704e-06,1.491984].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.19e+00 NA NA  
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.76e-04 1.03 0.98  
## s(jnight):HabitatNatural 1.40e+01 2.72e+00 1.03 0.96  
## s(jnight):HabitatTurbinePad 1.40e+01 4.88e+00 1.03 0.95  
## s(wind\_mean) 1.90e+01 8.38e+00 1.02 0.94  
## ti(temp\_mean,wind\_mean) 7.10e+01 2.29e+01 0.93 0.11

gam.check(m9, 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 10 iterations.  
## Gradient range [-4.44288e-05,1.328535e-05]  
## (score 318.0472 & scale 1).  
## Hessian positive definite, eigenvalue range [5.91704e-06,1.491984].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.19e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.76e-04 1.03 0.96   
## s(jnight):HabitatNatural 1.40e+01 2.72e+00 1.03 0.95   
## s(jnight):HabitatTurbinePad 1.40e+01 4.88e+00 1.03 0.94   
## s(wind\_mean) 1.90e+01 8.38e+00 1.02 0.93   
## ti(temp\_mean,wind\_mean) 7.10e+01 2.29e+01 0.93 0.05 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m9 <- sum( residuals(m9, "pearson")^2 ) / m9$df.residual  
overdispersion.m9

## [1] 0.566872

# 0.566872 - The k indices look okay, but still under dispersed  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

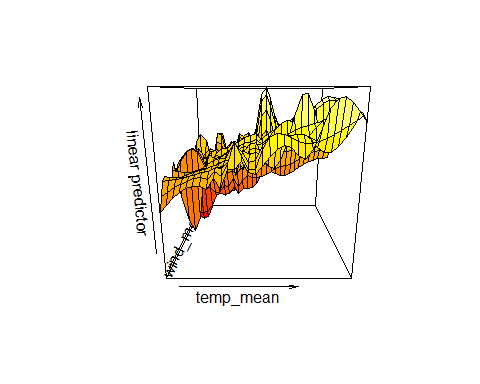
bmp(file.path(output\_today, "checking k dimensions m9 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m9, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m9, scales ="free") #function 'draw' is from package 'gratia'  
draw(m9, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

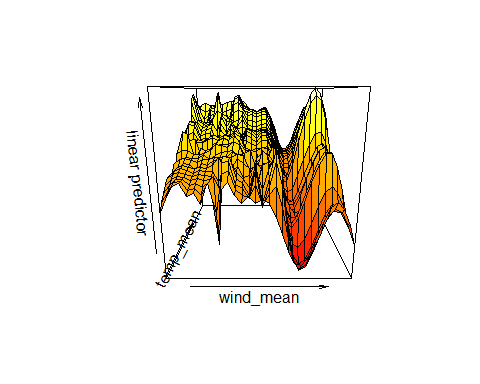
bmp(file.path(output\_today, "3D gam m9 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m9, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m9, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m9, view = c("temp\_mean", "wind\_mean"))



vis.gam(m9, view = c("wind\_mean", "temp\_mean"))



## This slightly improved the overall model fit, but still have the same issues overall.

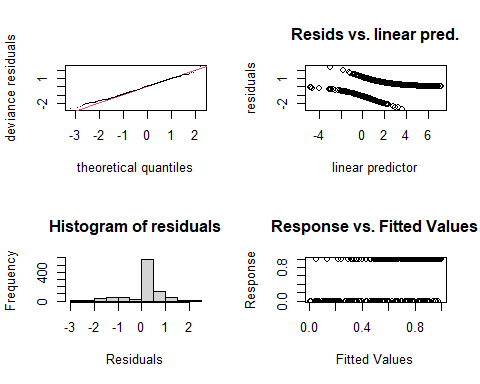
# m10

### Adjust ti(temp\_mean,wind\_mean, k = c(25,25))

m10 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 15) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 20) +   
 ti(temp\_mean,wind\_mean, k = c(25,25)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m10)

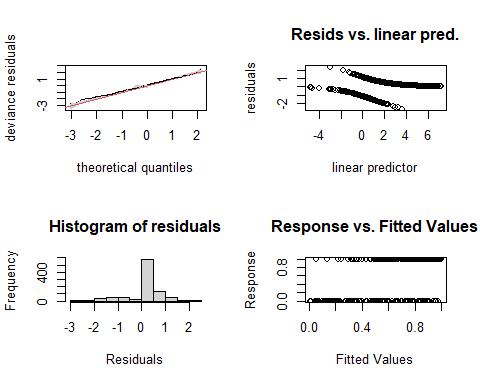
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 15) + Habitat + temp\_mean +   
## s(wind\_mean, k = 20) + ti(temp\_mean, wind\_mean, k = c(25,   
## 25))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.1135 1.0553 -3.898 9.70e-05 \*\*\*  
## HabitatNatural 2.6919 0.3999 6.731 1.68e-11 \*\*\*  
## HabitatTurbinePad 3.3225 0.4969 6.686 2.29e-11 \*\*\*  
## temp\_mean 0.2631 0.0650 4.047 5.18e-05 \*\*\*  
## ---  
## 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) 2.073e+00 6 2.86 0.214   
## s(jnight):HabitatMeteorologicalTower 1.414e-04 14 0.00 0.950   
## s(jnight):HabitatNatural 7.033e-05 14 0.00 0.573   
## s(jnight):HabitatTurbinePad 4.148e+00 14 25.13 2.45e-06 \*\*\*  
## s(wind\_mean) 3.042e+00 19 32.77 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.956e+01 71 73.33 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.426 Deviance explained = 44.6%  
## -REML = 309.94 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m10) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-5.598939e-05,6.30014e-06]  
## (score 309.9379 & scale 1).  
## Hessian positive definite, eigenvalue range [1.184873e-06,2.490004].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.07e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.41e-04 1.02 0.93   
## s(jnight):HabitatNatural 1.40e+01 7.03e-05 1.02 0.94   
## s(jnight):HabitatTurbinePad 1.40e+01 4.15e+00 1.02 0.92   
## s(wind\_mean) 1.90e+01 3.04e+00 1.02 0.95   
## ti(temp\_mean,wind\_mean) 7.10e+01 2.96e+01 0.92 0.02 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m10, 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 9 iterations.  
## Gradient range [-5.598939e-05,6.30014e-06]  
## (score 309.9379 & scale 1).  
## Hessian positive definite, eigenvalue range [1.184873e-06,2.490004].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 2.07e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.40e+01 1.41e-04 1.02 0.94   
## s(jnight):HabitatNatural 1.40e+01 7.03e-05 1.02 0.93   
## s(jnight):HabitatTurbinePad 1.40e+01 4.15e+00 1.02 0.92   
## s(wind\_mean) 1.90e+01 3.04e+00 1.02 0.94   
## ti(temp\_mean,wind\_mean) 7.10e+01 2.96e+01 0.92 0.02 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m10 <- sum( residuals(m10, "pearson")^2 ) / m10$df.residual  
overdispersion.m10

## [1] 0.6045513

# 0.6045513 - The k indices look okay, but still under dispersed - though not as bad as before!   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

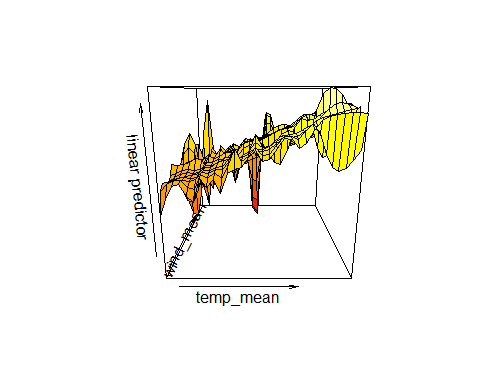
bmp(file.path(output\_today, "checking k dimensions m10 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m10, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m10, scales ="free") #function 'draw' is from package 'gratia'  
draw(m10, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

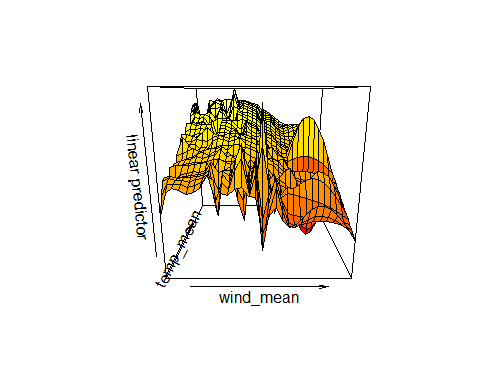
bmp(file.path(output\_today, "3D gam m10 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m10, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m10, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m10, view = c("temp\_mean", "wind\_mean"))



vis.gam(m10, view = c("wind\_mean", "temp\_mean"))



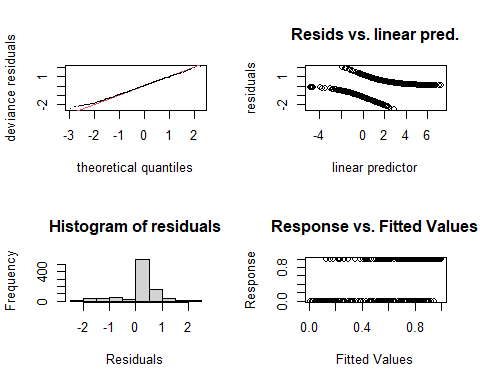
# m11

### Adjust ti(temp\_mean,wind\_mean, k = c(30,30))

m11 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 15) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 20) +   
 ti(temp\_mean,wind\_mean, k = c(30,30)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m11)

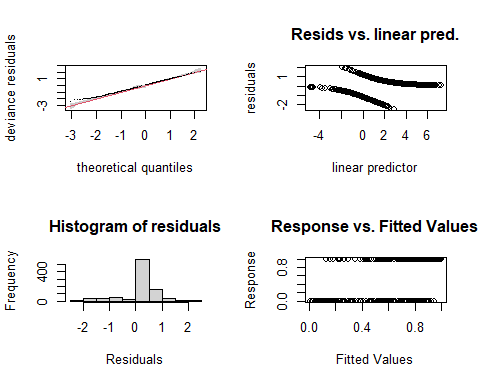
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 15) + Habitat + temp\_mean +   
## s(wind\_mean, k = 20) + ti(temp\_mean, wind\_mean, k = c(30,   
## 30))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.37217 1.14616 -2.942 0.00326 \*\*   
## HabitatNatural 2.74688 0.50943 5.392 6.97e-08 \*\*\*  
## HabitatTurbinePad 3.40354 0.61078 5.572 2.51e-08 \*\*\*  
## temp\_mean 0.20453 0.06697 3.054 0.00226 \*\*   
## ---  
## 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) 3.214e+00 6 5.855 0.0881 .   
## s(jnight):HabitatMeteorologicalTower 2.637e-05 14 0.000 0.9370   
## s(jnight):HabitatNatural 6.236e-05 14 0.000 0.5332   
## s(jnight):HabitatTurbinePad 4.769e+00 14 26.164 1.28e-05 \*\*\*  
## s(wind\_mean) 2.422e+00 19 22.812 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 3.352e+01 71 83.330 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.445 Deviance explained = 46.8%  
## -REML = 307.25 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m11) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 14 iterations.  
## Gradient range [-1.27053e-05,4.364198e-05]  
## (score 307.2456 & scale 1).  
## Hessian positive definite, eigenvalue range [5.450552e-06,3.216063].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 3.21e+00 NA NA  
## s(jnight):HabitatMeteorologicalTower 1.40e+01 2.64e-05 1.06 1.00  
## s(jnight):HabitatNatural 1.40e+01 6.24e-05 1.06 0.99  
## s(jnight):HabitatTurbinePad 1.40e+01 4.77e+00 1.06 1.00  
## s(wind\_mean) 1.90e+01 2.42e+00 1.05 0.98  
## ti(temp\_mean,wind\_mean) 7.10e+01 3.35e+01 0.95 0.21

gam.check(m11, 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 14 iterations.  
## Gradient range [-1.27053e-05,4.364198e-05]  
## (score 307.2456 & scale 1).  
## Hessian positive definite, eigenvalue range [5.450552e-06,3.216063].  
## Model rank = 144 / 144   
##   
## 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) 8.00e+00 3.21e+00 NA NA  
## s(jnight):HabitatMeteorologicalTower 1.40e+01 2.64e-05 1.06 1.00  
## s(jnight):HabitatNatural 1.40e+01 6.24e-05 1.06 0.99  
## s(jnight):HabitatTurbinePad 1.40e+01 4.77e+00 1.06 1.00  
## s(wind\_mean) 1.90e+01 2.42e+00 1.05 0.98  
## ti(temp\_mean,wind\_mean) 7.10e+01 3.35e+01 0.95 0.21

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m11 <- sum( residuals(m11, "pearson")^2 ) / m11$df.residual  
overdispersion.m11

## [1] 0.5309308

# 0.5309308 - The k indices look good but now the under dispersion is worse - I have gone too far in hiking up the k values for the ti() term.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

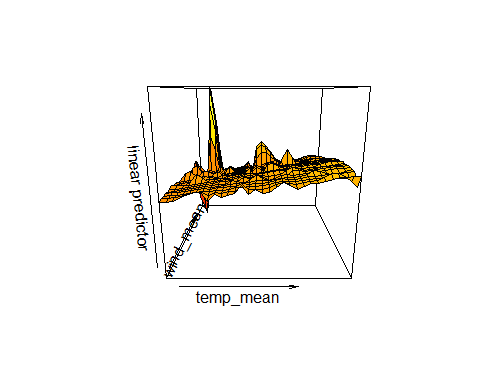
bmp(file.path(output\_today, "checking k dimensions m11 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m11, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m11, scales ="free") #function 'draw' is from package 'gratia'  
draw(m11, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

bmp(file.path(output\_today, "3D gam m11 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m11, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m11, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m11, view = c("temp\_mean", "wind\_mean"))



vis.gam(m11, view = c("wind\_mean", "temp\_mean"))

## 

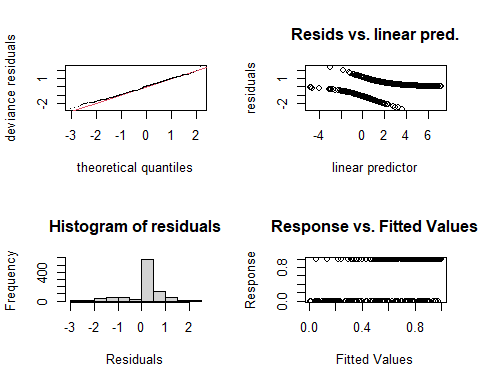
# m12

### s(jnight, by = Habitat, bs = “gp”, k = 20)

m12 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 20) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 20) +   
 ti(temp\_mean,wind\_mean, k = c(25,25)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m12)

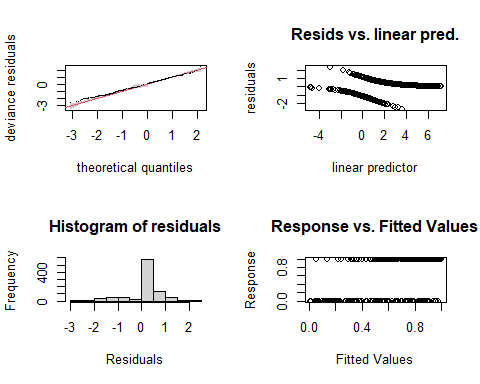
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 20) + Habitat + temp\_mean +   
## s(wind\_mean, k = 20) + ti(temp\_mean, wind\_mean, k = c(25,   
## 25))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.11369 1.05527 -3.898 9.69e-05 \*\*\*  
## HabitatNatural 2.69143 0.40031 6.723 1.77e-11 \*\*\*  
## HabitatTurbinePad 3.32200 0.49728 6.680 2.38e-11 \*\*\*  
## temp\_mean 0.26310 0.06499 4.048 5.16e-05 \*\*\*  
## ---  
## 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) 2.079e+00 6 2.871 0.213   
## s(jnight):HabitatMeteorologicalTower 1.420e-04 18 0.000 0.950   
## s(jnight):HabitatNatural 7.028e-05 19 0.000 0.574   
## s(jnight):HabitatTurbinePad 4.191e+00 19 25.176 4.06e-06 \*\*\*  
## s(wind\_mean) 3.041e+00 19 32.719 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.954e+01 71 73.230 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.426 Deviance explained = 44.6%  
## -REML = 309.94 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m12) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-5.622482e-05,6.291195e-06]  
## (score 309.9356 & scale 1).  
## Hessian positive definite, eigenvalue range [1.185148e-06,2.488254].  
## Model rank = 159 / 159   
##   
## 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) 8.00e+00 2.08e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.90e+01 1.42e-04 1.02 0.910   
## s(jnight):HabitatNatural 1.90e+01 7.03e-05 1.02 0.945   
## s(jnight):HabitatTurbinePad 1.90e+01 4.19e+00 1.02 0.905   
## s(wind\_mean) 1.90e+01 3.04e+00 1.02 0.955   
## ti(temp\_mean,wind\_mean) 7.10e+01 2.95e+01 0.92 0.025 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m12, 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 9 iterations.  
## Gradient range [-5.622482e-05,6.291195e-06]  
## (score 309.9356 & scale 1).  
## Hessian positive definite, eigenvalue range [1.185148e-06,2.488254].  
## Model rank = 159 / 159   
##   
## 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) 8.00e+00 2.08e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.90e+01 1.42e-04 1.02 0.92   
## s(jnight):HabitatNatural 1.90e+01 7.03e-05 1.02 0.92   
## s(jnight):HabitatTurbinePad 1.90e+01 4.19e+00 1.02 0.92   
## s(wind\_mean) 1.90e+01 3.04e+00 1.02 0.94   
## ti(temp\_mean,wind\_mean) 7.10e+01 2.95e+01 0.92 0.02 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m12 <- sum( residuals(m12, "pearson")^2 ) / m12$df.residual  
overdispersion.m12

## [1] 0.604303

# 0.604303 - Under dispersed and the k indices are still off.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

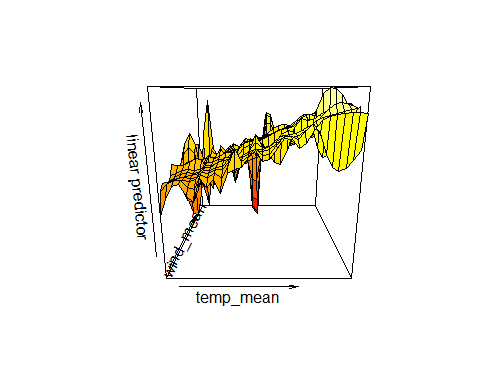
bmp(file.path(output\_today, "checking k dimensions m12 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m12, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m12, scales ="free") #function 'draw' is from package 'gratia'  
draw(m12, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

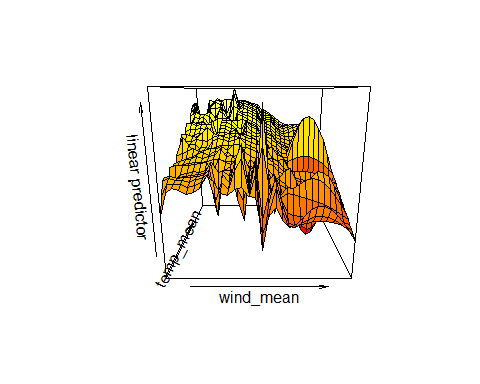
bmp(file.path(output\_today, "3D gam m12 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m12, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m12, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m12, view = c("temp\_mean", "wind\_mean"))



vis.gam(m12, view = c("wind\_mean", "temp\_mean"))



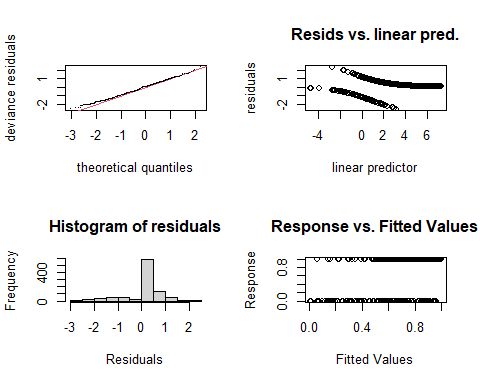
# m13

### s(wind\_mean, k = 25)

m13 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp", k = 20) +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, k = 25) +   
 ti(temp\_mean,wind\_mean, k = c(25,25)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m13)

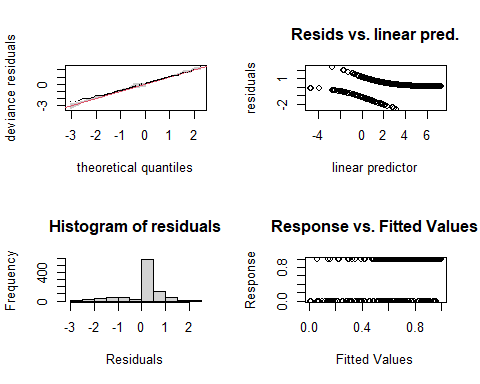
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Locality, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp", k = 20) + Habitat + temp\_mean +   
## s(wind\_mean, k = 25) + ti(temp\_mean, wind\_mean, k = c(25,   
## 25))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.52421 1.00191 -3.517 0.000436 \*\*\*  
## HabitatNatural 2.66658 0.35313 7.551 4.31e-14 \*\*\*  
## HabitatTurbinePad 3.31133 0.45674 7.250 4.17e-13 \*\*\*  
## temp\_mean 0.22864 0.06251 3.658 0.000255 \*\*\*  
## ---  
## 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) 1.353e+00 6 1.616 0.300   
## s(jnight):HabitatMeteorologicalTower 2.267e-04 18 0.000 0.930   
## s(jnight):HabitatNatural 1.761e-04 19 0.000 0.461   
## s(jnight):HabitatTurbinePad 4.170e+00 19 26.308 1.35e-07 \*\*\*  
## s(wind\_mean) 3.401e+00 24 32.057 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 2.740e+01 66 70.485 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.415 Deviance explained = 43.9%  
## -REML = 310.3 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m13) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 8 iterations.  
## Gradient range [-8.799493e-05,1.582536e-05]  
## (score 310.3014 & scale 1).  
## Hessian positive definite, eigenvalue range [2.685605e-06,2.595629].  
## Model rank = 159 / 159   
##   
## 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) 8.00e+00 1.35e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.90e+01 2.27e-04 1.02 0.90   
## s(jnight):HabitatNatural 1.90e+01 1.76e-04 1.02 0.92   
## s(jnight):HabitatTurbinePad 1.90e+01 4.17e+00 1.02 0.94   
## s(wind\_mean) 2.40e+01 3.40e+00 1.01 0.85   
## ti(temp\_mean,wind\_mean) 6.60e+01 2.74e+01 0.91 0.03 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m13, 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 8 iterations.  
## Gradient range [-8.799493e-05,1.582536e-05]  
## (score 310.3014 & scale 1).  
## Hessian positive definite, eigenvalue range [2.685605e-06,2.595629].  
## Model rank = 159 / 159   
##   
## 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) 8.00e+00 1.35e+00 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.90e+01 2.27e-04 1.02 0.925   
## s(jnight):HabitatNatural 1.90e+01 1.76e-04 1.02 0.890   
## s(jnight):HabitatTurbinePad 1.90e+01 4.17e+00 1.02 0.930   
## s(wind\_mean) 2.40e+01 3.40e+00 1.01 0.865   
## ti(temp\_mean,wind\_mean) 6.60e+01 2.74e+01 0.91 0.015 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m13 <- sum( residuals(m13, "pearson")^2 ) / m13$df.residual  
overdispersion.m13

## [1] 0.5874084

# 0.5874084 - Under dispersed and the k indices are still off.   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

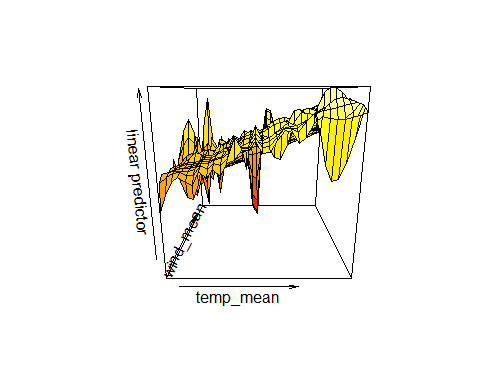
bmp(file.path(output\_today, "checking k dimensions m13 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m13, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m13, scales ="free") #function 'draw' is from package 'gratia'  
draw(m13, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

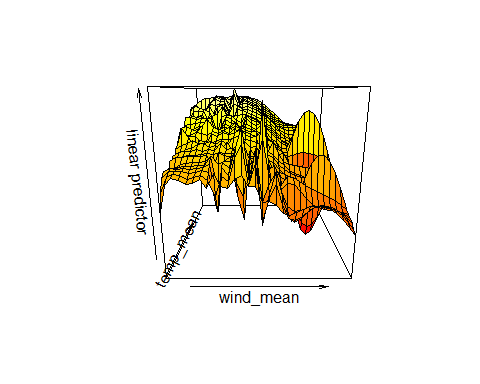
bmp(file.path(output\_today, "3D gam m13 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m13, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m13, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m13, view = c("temp\_mean", "wind\_mean"))



vis.gam(m13, view = c("wind\_mean", "temp\_mean"))



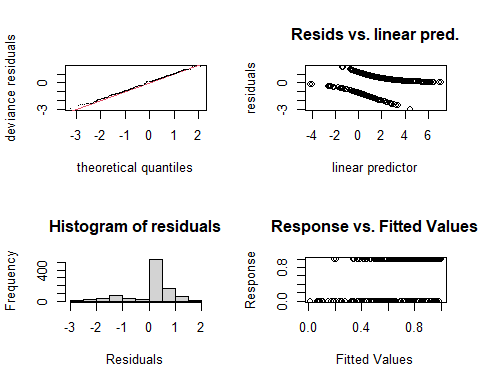
# m14

### s(Facility, bs = “re”) rather than Locality

m14 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m14)

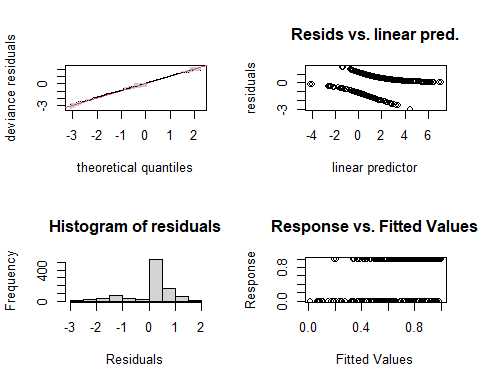
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Facility, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp") + Habitat + temp\_mean + s(wind\_mean) +   
## ti(temp\_mean, wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.66230 1.18192 -3.945 7.99e-05 \*\*\*  
## HabitatNatural 2.64078 0.30382 8.692 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.28684 0.45035 7.298 2.91e-13 \*\*\*  
## temp\_mean 0.30115 0.07428 4.054 5.03e-05 \*\*\*  
## ---  
## 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(Facility) 4.004e-01 1 0.647 0.203279   
## s(jnight):HabitatMeteorologicalTower 3.954e-05 11 0.000 0.902691   
## s(jnight):HabitatNatural 2.756e+00 11 10.046 0.007271 \*\*   
## s(jnight):HabitatTurbinePad 4.780e+00 11 39.929 < 2e-16 \*\*\*  
## s(wind\_mean) 3.285e+00 9 16.600 5.81e-05 \*\*\*  
## ti(temp\_mean,wind\_mean) 5.340e+00 16 18.806 0.000367 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.359 Deviance explained = 36.7%  
## -REML = 321 Scale est. = 1 n = 951

#windows()  
par(mfrow = c(2,2))  
gam.check(m14) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 14 iterations.  
## Gradient range [-1.054294e-05,6.634679e-06]  
## (score 320.9973 & scale 1).  
## Hessian positive definite, eigenvalue range [5.584768e-06,0.7392065].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.00e+00 4.00e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 3.95e-05 0.95 0.21   
## s(jnight):HabitatNatural 1.10e+01 2.76e+00 0.95 0.14   
## s(jnight):HabitatTurbinePad 1.10e+01 4.78e+00 0.95 0.17   
## s(wind\_mean) 9.00e+00 3.29e+00 0.93 0.07 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.34e+00 0.84 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m14, 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 14 iterations.  
## Gradient range [-1.054294e-05,6.634679e-06]  
## (score 320.9973 & scale 1).  
## Hessian positive definite, eigenvalue range [5.584768e-06,0.7392065].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.00e+00 4.00e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 3.95e-05 0.95 0.16   
## s(jnight):HabitatNatural 1.10e+01 2.76e+00 0.95 0.16   
## s(jnight):HabitatTurbinePad 1.10e+01 4.78e+00 0.95 0.15   
## s(wind\_mean) 9.00e+00 3.29e+00 0.93 0.07 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.34e+00 0.84 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m14 <- sum( residuals(m14, "pearson")^2 ) / m14$df.residual  
overdispersion.m14

## [1] 0.8134479

# 0.8134479 - under dispersion is fixed by the k values are still not good  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

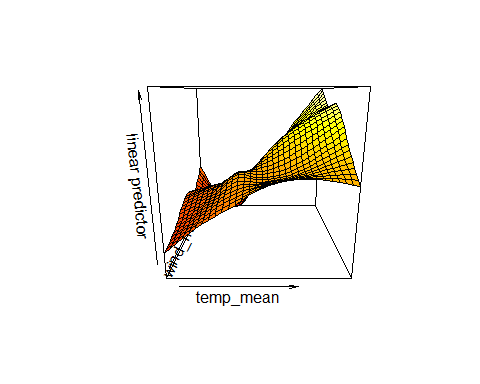
bmp(file.path(output\_today, "checking k dimensions m14 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m14, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m14, scales ="free") #function 'draw' is from package 'gratia'  
draw(m14, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

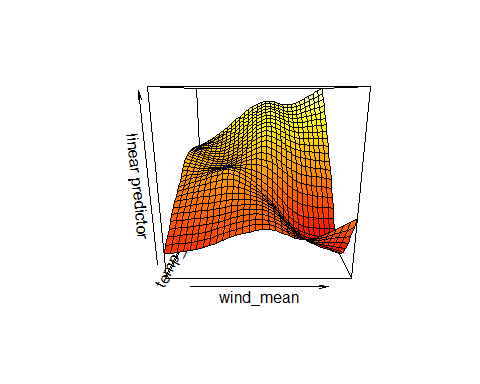
bmp(file.path(output\_today, "3D gam m14 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m14, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m14, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m14, view = c("temp\_mean", "wind\_mean"))



vis.gam(m14, view = c("wind\_mean", "temp\_mean"))



## Using facility rather than locality as a random effect did not seem to make much a difference, can try it again once more with adjusting the k values for the ti() term.

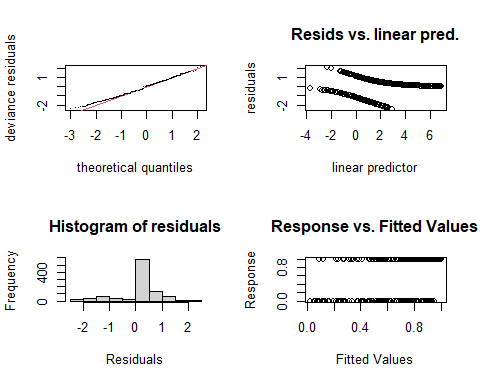
# m15

### s(Facility, bs = “re”) rather than Locality

m15 <- gam(cbind(yes\_batpass,no\_batpass) ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(20,20)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m15)

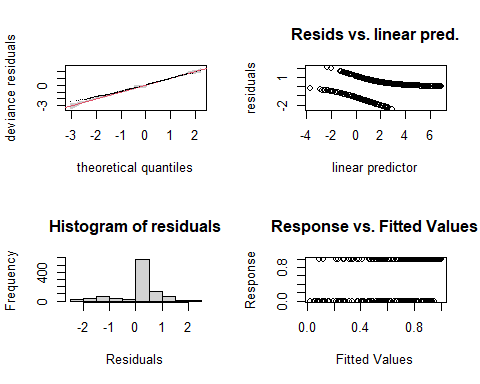
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## cbind(yes\_batpass, no\_batpass) ~ s(Facility, bs = "re") + s(jnight,   
## by = Habitat, bs = "gp") + Habitat + temp\_mean + s(wind\_mean) +   
## ti(temp\_mean, wind\_mean, k = c(20, 20))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.8277 1.0296 -3.718 0.000201 \*\*\*  
## HabitatNatural 2.6919 0.3046 8.838 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.4038 0.4295 7.925 2.28e-15 \*\*\*  
## temp\_mean 0.2396 0.0640 3.743 0.000182 \*\*\*  
## ---  
## 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(Facility) 5.072e-01 1 0.96 0.168   
## s(jnight):HabitatMeteorologicalTower 1.708e-04 11 0.00 0.657   
## s(jnight):HabitatNatural 2.188e-05 11 0.00 0.367   
## s(jnight):HabitatTurbinePad 4.217e+00 11 27.62 3.63e-06 \*\*\*  
## s(wind\_mean) 2.609e+00 9 20.02 2.08e-06 \*\*\*  
## ti(temp\_mean,wind\_mean) 3.071e+01 81 69.45 1.48e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.405 Deviance explained = 43.1%  
## -REML = 316.77 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m15) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-5.052429e-05,3.787661e-06]  
## (score 316.77 & scale 1).  
## Hessian positive definite, eigenvalue range [1.898618e-07,2.257754].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.00e+00 5.07e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 1.71e-04 1.01 0.86   
## s(jnight):HabitatNatural 1.10e+01 2.19e-05 1.01 0.92   
## s(jnight):HabitatTurbinePad 1.10e+01 4.22e+00 1.01 0.83   
## s(wind\_mean) 9.00e+00 2.61e+00 1.01 0.92   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.07e+01 0.90 0.01 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m15, 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 9 iterations.  
## Gradient range [-5.052429e-05,3.787661e-06]  
## (score 316.77 & scale 1).  
## Hessian positive definite, eigenvalue range [1.898618e-07,2.257754].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.00e+00 5.07e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 1.71e-04 1.01 0.920   
## s(jnight):HabitatNatural 1.10e+01 2.19e-05 1.01 0.860   
## s(jnight):HabitatTurbinePad 1.10e+01 4.22e+00 1.01 0.900   
## s(wind\_mean) 9.00e+00 2.61e+00 1.01 0.900   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.07e+01 0.90 0.015 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m15 <- sum( residuals(m15, "pearson")^2 ) / m15$df.residual  
overdispersion.m15

## [1] 0.574884

# 0.574884 - under dispersion is there and the k values are still not good  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

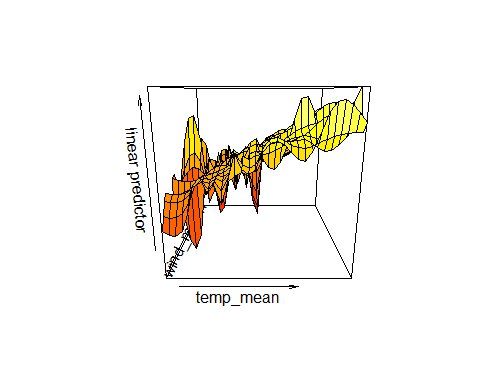
bmp(file.path(output\_today, "checking k dimensions m15 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m15, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m15, scales ="free") #function 'draw' is from package 'gratia'  
draw(m15, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

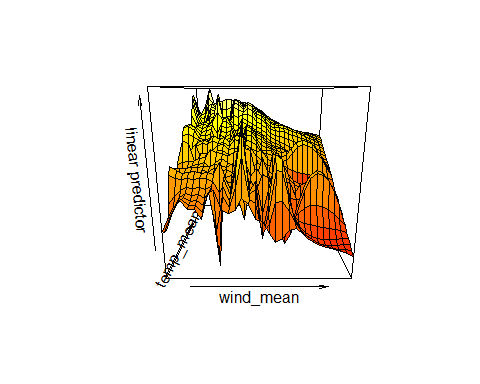
bmp(file.path(output\_today, "3D gam m15 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m15, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m15, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m15, view = c("temp\_mean", "wind\_mean"))



vis.gam(m15, view = c("wind\_mean", "temp\_mean"))



I am not sure what else to do at this point other than use a different response variable. # try batpass\_prop

# m16

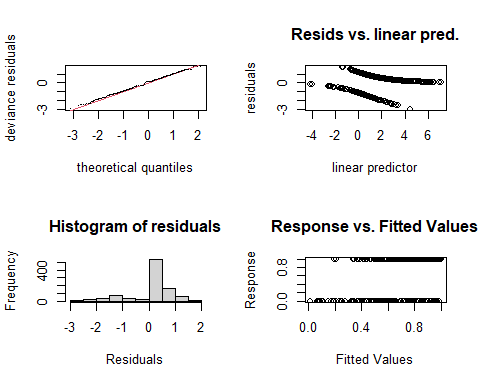
### y = batpass\_prop

### s(Facility, bs = “re”) rather than Locality

m16 <- gam(batpass\_prop ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean),   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m16)

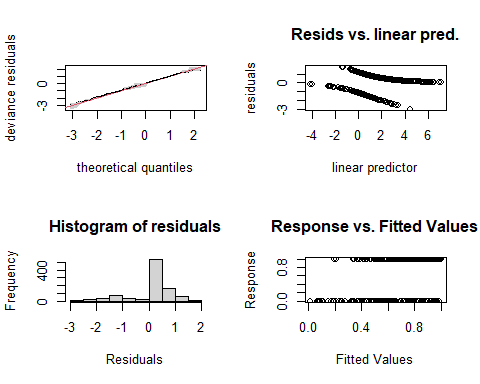
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## batpass\_prop ~ s(Facility, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean,   
## wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.66230 1.18192 -3.945 7.99e-05 \*\*\*  
## HabitatNatural 2.64078 0.30382 8.692 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.28684 0.45035 7.298 2.91e-13 \*\*\*  
## temp\_mean 0.30115 0.07428 4.054 5.03e-05 \*\*\*  
## ---  
## 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(Facility) 4.004e-01 1 0.647 0.203279   
## s(jnight):HabitatMeteorologicalTower 3.954e-05 11 0.000 0.902691   
## s(jnight):HabitatNatural 2.756e+00 11 10.046 0.007271 \*\*   
## s(jnight):HabitatTurbinePad 4.780e+00 11 39.929 < 2e-16 \*\*\*  
## s(wind\_mean) 3.285e+00 9 16.600 5.81e-05 \*\*\*  
## ti(temp\_mean,wind\_mean) 5.340e+00 16 18.806 0.000367 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.359 Deviance explained = 36.7%  
## -REML = 321 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m16) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 14 iterations.  
## Gradient range [-1.054294e-05,6.634679e-06]  
## (score 320.9973 & scale 1).  
## Hessian positive definite, eigenvalue range [5.584768e-06,0.7392065].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.00e+00 4.00e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 3.95e-05 0.95 0.14   
## s(jnight):HabitatNatural 1.10e+01 2.76e+00 0.95 0.18   
## s(jnight):HabitatTurbinePad 1.10e+01 4.78e+00 0.95 0.12   
## s(wind\_mean) 9.00e+00 3.29e+00 0.93 0.07 .   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.34e+00 0.84 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m16, 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 14 iterations.  
## Gradient range [-1.054294e-05,6.634679e-06]  
## (score 320.9973 & scale 1).  
## Hessian positive definite, eigenvalue range [5.584768e-06,0.7392065].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.00e+00 4.00e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 3.95e-05 0.95 0.180   
## s(jnight):HabitatNatural 1.10e+01 2.76e+00 0.95 0.180   
## s(jnight):HabitatTurbinePad 1.10e+01 4.78e+00 0.95 0.115   
## s(wind\_mean) 9.00e+00 3.29e+00 0.93 0.045 \*   
## ti(temp\_mean,wind\_mean) 1.60e+01 5.34e+00 0.84 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m16 <- sum( residuals(m16, "pearson")^2 ) / m16$df.residual  
overdispersion.m16

## [1] 0.8134479

# 0.8134479 - under dispersion is there and the k values are still not good  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

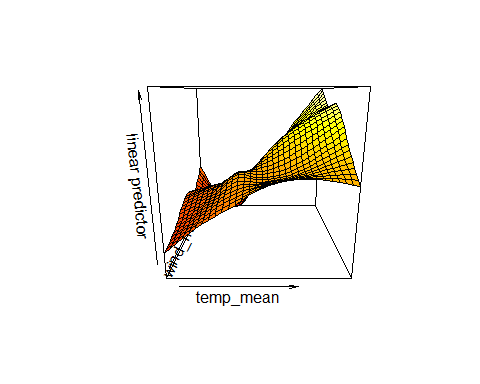
bmp(file.path(output\_today, "checking k dimensions m16 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m16, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m16, scales ="free") #function 'draw' is from package 'gratia'  
draw(m16, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

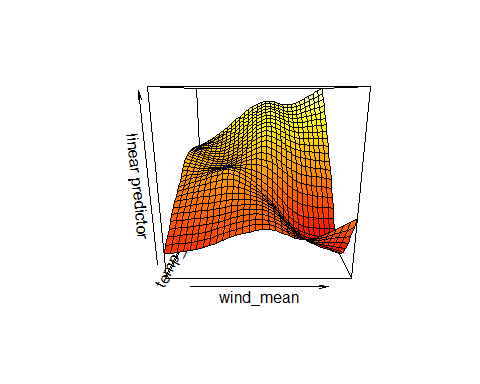
bmp(file.path(output\_today, "3D gam m16 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m16, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m16, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m16, view = c("temp\_mean", "wind\_mean"))



vis.gam(m16, view = c("wind\_mean", "temp\_mean"))



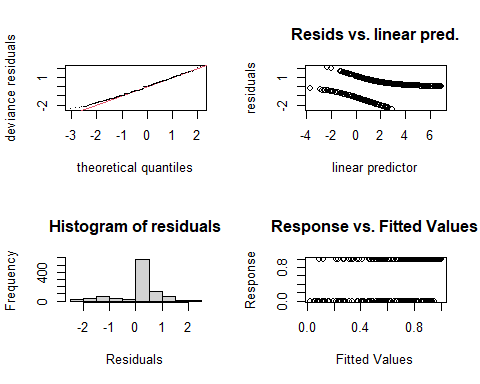
# m17

### s(Facility, bs = “re”) rather than Locality

m17 <- gam(batpass\_prop ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(20,20)) ,   
 data = bats\_tot, method = "REML",   
 family = binomial, select=TRUE)   
  
summary(m17)

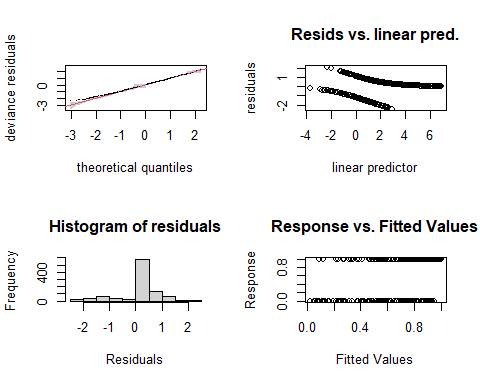
##   
## Family: binomial   
## Link function: logit   
##   
## Formula:  
## batpass\_prop ~ s(Facility, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean,   
## wind\_mean, k = c(20, 20))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.8277 1.0296 -3.718 0.000201 \*\*\*  
## HabitatNatural 2.6919 0.3046 8.838 < 2e-16 \*\*\*  
## HabitatTurbinePad 3.4038 0.4295 7.925 2.28e-15 \*\*\*  
## temp\_mean 0.2396 0.0640 3.743 0.000182 \*\*\*  
## ---  
## 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(Facility) 5.072e-01 1 0.96 0.168   
## s(jnight):HabitatMeteorologicalTower 1.708e-04 11 0.00 0.657   
## s(jnight):HabitatNatural 2.188e-05 11 0.00 0.367   
## s(jnight):HabitatTurbinePad 4.217e+00 11 27.62 3.63e-06 \*\*\*  
## s(wind\_mean) 2.609e+00 9 20.02 2.08e-06 \*\*\*  
## ti(temp\_mean,wind\_mean) 3.071e+01 81 69.45 1.48e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.405 Deviance explained = 43.1%  
## -REML = 316.77 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m17) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-5.052429e-05,3.787661e-06]  
## (score 316.77 & scale 1).  
## Hessian positive definite, eigenvalue range [1.898618e-07,2.257754].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.00e+00 5.07e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 1.71e-04 1.01 0.865   
## s(jnight):HabitatNatural 1.10e+01 2.19e-05 1.01 0.860   
## s(jnight):HabitatTurbinePad 1.10e+01 4.22e+00 1.01 0.895   
## s(wind\_mean) 9.00e+00 2.61e+00 1.01 0.860   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.07e+01 0.90 0.005 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m17, 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 9 iterations.  
## Gradient range [-5.052429e-05,3.787661e-06]  
## (score 316.77 & scale 1).  
## Hessian positive definite, eigenvalue range [1.898618e-07,2.257754].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.00e+00 5.07e-01 NA NA   
## s(jnight):HabitatMeteorologicalTower 1.10e+01 1.71e-04 1.01 0.90   
## s(jnight):HabitatNatural 1.10e+01 2.19e-05 1.01 0.88   
## s(jnight):HabitatTurbinePad 1.10e+01 4.22e+00 1.01 0.86   
## s(wind\_mean) 9.00e+00 2.61e+00 1.01 0.85   
## ti(temp\_mean,wind\_mean) 8.10e+01 3.07e+01 0.90 0.01 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m17 <- sum( residuals(m17, "pearson")^2 ) / m17$df.residual  
overdispersion.m17

## [1] 0.574884

# 0.574884 - under dispersion is there and the k values are still not good  
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

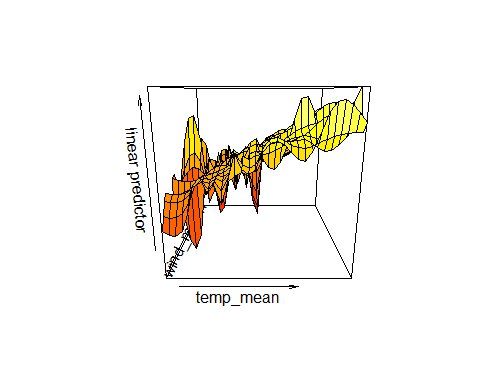
bmp(file.path(output\_today, "checking k dimensions m17 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m17, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m17, scales ="free") #function 'draw' is from package 'gratia'  
draw(m17, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

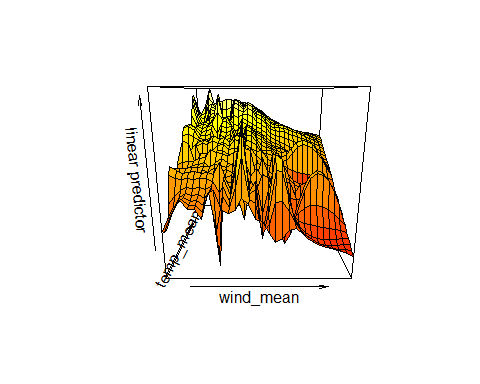
bmp(file.path(output\_today, "3D gam m17 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m17, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m17, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m17, view = c("temp\_mean", "wind\_mean"))



vis.gam(m17, view = c("wind\_mean", "temp\_mean"))



This did not help, it seems like ti(temp\_mean,wind\_mean) is the main issue regardless of what I do.

I am going to switch to using a negative binomial model.

# Troubelshooting under dispersed binomial gams with random effects

Making a higher k value can help with improving the model fit but it tends to lead to underdispersion in this case. Removing the met tower data makes the models easier to work with.

To Address under dispersion in binary GAMs:

<https://dergipark.org.tr/en/download/article-file/922496> This article suggests using an extended Altham distribution (EAD) family for under dispersed binomial count data- however I cannot find any commands in the mgcv package that will allow me to call that distribution.

The MM package does have the option to use this family but before changing packages, it is perhaps wiser to try and use counts as a response variable

<https://cran.r-project.org/web/packages/MM/vignettes/Gianfranco.pdf>

# m18

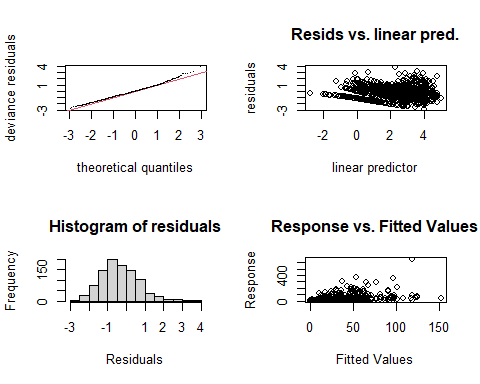
### try model 1 with a negative binomial gamily insead of typical binomail

### Response variable = Batpass\_sum

m18 <- gam(Batpass\_sum ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean) ,   
 data = bats\_tot, method = "REML",   
 family = nb(), select=TRUE)   
  
summary(m18)

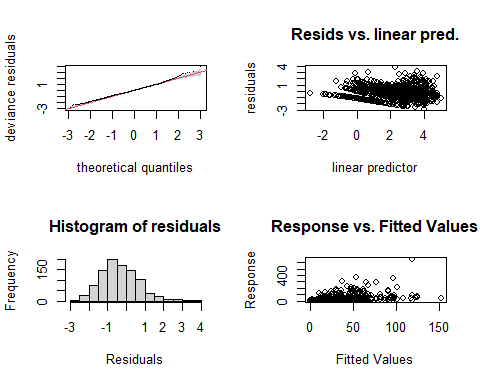
##   
## Family: Negative Binomial(1)   
## Link function: log   
##   
## Formula:  
## Batpass\_sum ~ s(Facility, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean,   
## wind\_mean)  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.92721 0.67293 -4.350 1.36e-05 \*\*\*  
## HabitatNatural 2.80061 0.13657 20.507 < 2e-16 \*\*\*  
## HabitatTurbinePad 2.79773 0.13607 20.561 < 2e-16 \*\*\*  
## temp\_mean 0.17445 0.02686 6.495 8.33e-11 \*\*\*  
## ---  
## 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(Facility) 0.9941 1 166.028 < 2e-16 \*\*\*  
## s(jnight):HabitatMeteorologicalTower 0.9245 11 2.587 0.06893 .   
## s(jnight):HabitatNatural 5.7186 11 91.216 < 2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 7.9758 11 152.842 < 2e-16 \*\*\*  
## s(wind\_mean) 3.5535 9 74.919 < 2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 5.1796 16 15.819 0.00213 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.236 Deviance explained = 58.4%  
## -REML = 3187.3 Scale est. = 1 n = 951

par(mfrow = c(2,2))  
gam.check(m18) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 9 iterations.  
## Gradient range [-0.001600668,0.001726144]  
## (score 3187.286 & scale 1).  
## Hessian positive definite, eigenvalue range [9.261805e-05,345.4133].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.000 0.994 NA NA   
## s(jnight):HabitatMeteorologicalTower 11.000 0.925 0.89 0.45   
## s(jnight):HabitatNatural 11.000 5.719 0.89 0.41   
## s(jnight):HabitatTurbinePad 11.000 7.976 0.89 0.49   
## s(wind\_mean) 9.000 3.553 0.87 0.28   
## ti(temp\_mean,wind\_mean) 16.000 5.180 0.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m18, 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 9 iterations.  
## Gradient range [-0.001600668,0.001726144]  
## (score 3187.286 & scale 1).  
## Hessian positive definite, eigenvalue range [9.261805e-05,345.4133].  
## Model rank = 64 / 64   
##   
## 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(Facility) 2.000 0.994 NA NA   
## s(jnight):HabitatMeteorologicalTower 11.000 0.925 0.89 0.49   
## s(jnight):HabitatNatural 11.000 5.719 0.89 0.42   
## s(jnight):HabitatTurbinePad 11.000 7.976 0.89 0.46   
## s(wind\_mean) 9.000 3.553 0.87 0.24   
## ti(temp\_mean,wind\_mean) 16.000 5.180 0.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m18 <- sum( residuals(m18, "pearson")^2 ) / m18$df.residual  
overdispersion.m18

## [1] 1.24094

# 1.24094 - k indicies have simimlar issues but now we are overdispersed!   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

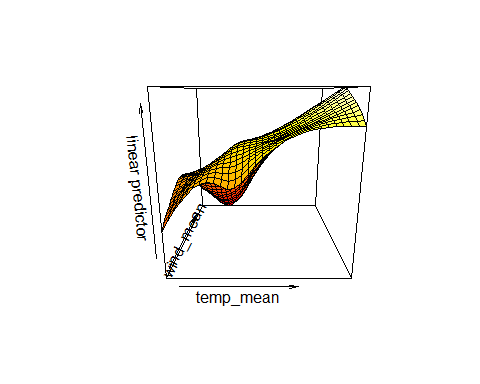
bmp(file.path(output\_today, "checking k dimensions m18 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m18, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m18, scales ="free") #function 'draw' is from package 'gratia'  
draw(m18, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

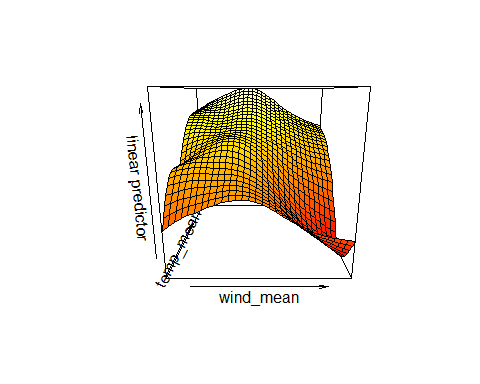
bmp(file.path(output\_today, "3D gam m18 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m18, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m18, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m18, view = c("temp\_mean", "wind\_mean"))



vis.gam(m18, view = c("wind\_mean", "temp\_mean"))



# m19

### try model 1 with a negative binomial gamily insead of typical binomail

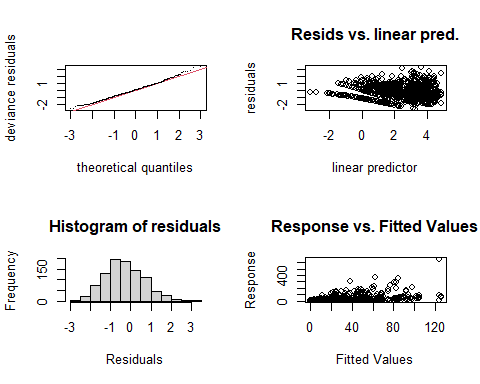
### Response variable = Batpass\_sum

### Adjust the k value on ti(temp\_mean,wind\_mean)

m19 <- gam(Batpass\_sum ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(25,25)) ,   
 data = bats\_tot, method = "REML",   
 family = nb(), select=TRUE)   
  
summary(m19)

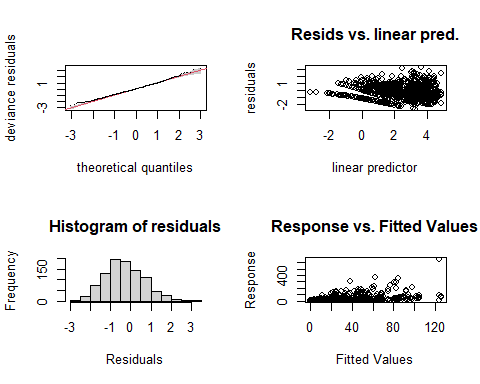
##   
## Family: Negative Binomial(1.138)   
## Link function: log   
##   
## Formula:  
## Batpass\_sum ~ s(Facility, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean,   
## wind\_mean, k = c(25, 25))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.29230 0.70729 -4.655 3.24e-06 \*\*\*  
## HabitatNatural 2.85183 0.13371 21.328 < 2e-16 \*\*\*  
## HabitatTurbinePad 2.83522 0.13312 21.298 < 2e-16 \*\*\*  
## temp\_mean 0.18846 0.02912 6.471 9.76e-11 \*\*\*  
## ---  
## 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(Facility) 0.9949 1 191.857 <2e-16 \*\*\*  
## s(jnight):HabitatMeteorologicalTower 0.3404 11 0.438 0.218   
## s(jnight):HabitatNatural 5.1292 11 47.069 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 7.3779 11 88.616 <2e-16 \*\*\*  
## s(wind\_mean) 3.5287 9 59.924 <2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 47.3180 81 155.242 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.243 Deviance explained = 64.8%  
## -REML = 3166.8 Scale est. = 1 n = 951

#windows()  
par(mfrow = c(2,2))  
gam.check(m19) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 5 iterations.  
## Gradient range [-0.0004690067,0.001813994]  
## (score 3166.767 & scale 1).  
## Hessian positive definite, eigenvalue range [0.0001318026,314.4461].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.000 0.995 NA NA  
## s(jnight):HabitatMeteorologicalTower 11.000 0.340 1.02 1.00  
## s(jnight):HabitatNatural 11.000 5.129 1.02 1.00  
## s(jnight):HabitatTurbinePad 11.000 7.378 1.02 1.00  
## s(wind\_mean) 9.000 3.529 1.01 1.00  
## ti(temp\_mean,wind\_mean) 81.000 47.318 0.88 0.16

gam.check(m19, 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 5 iterations.  
## Gradient range [-0.0004690067,0.001813994]  
## (score 3166.767 & scale 1).  
## Hessian positive definite, eigenvalue range [0.0001318026,314.4461].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.000 0.995 NA NA  
## s(jnight):HabitatMeteorologicalTower 11.000 0.340 1.02 1.00  
## s(jnight):HabitatNatural 11.000 5.129 1.02 1.00  
## s(jnight):HabitatTurbinePad 11.000 7.378 1.02 1.00  
## s(wind\_mean) 9.000 3.529 1.01 1.00  
## ti(temp\_mean,wind\_mean) 81.000 47.318 0.88 0.18

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m19 <- sum( residuals(m19, "pearson")^2 ) / m19$df.residual  
overdispersion.m19

## [1] 1.119004

# 1.119004 - dispersion is fine and the k values fit too! TADA!   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

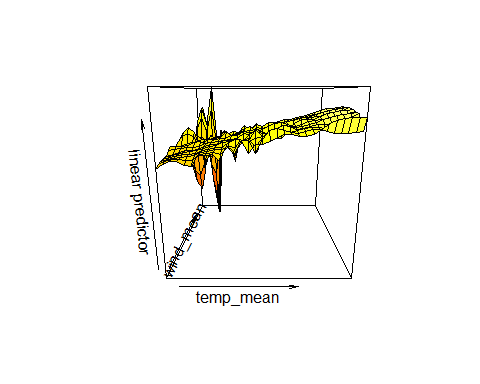
bmp(file.path(output\_today, "checking k dimensions m19 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m19, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m19, scales ="free") #function 'draw' is from package 'gratia'  
draw(m19, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

bmp(file.path(output\_today, "3D gam m19 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m19, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m19, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

vis.gam(m19, view = c("temp\_mean", "wind\_mean"))



vis.gam(m19, view = c("wind\_mean", "temp\_mean"))

## 

# m20

### try model 1 with a negative binomial gamily insead of typical binomail

### Response variable = Batpass\_sum

### Adjust the k value on ti(temp\_mean,wind\_mean)

### s(Locality, bs = “re”)

m20 <- gam(Batpass\_sum ~  
 s(Locality, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean) +   
 ti(temp\_mean,wind\_mean, k = c(25,25)) ,   
 data = bats\_tot, method = "REML",   
 family = nb(), select=TRUE)   
  
summary(m20)

##   
## Family: Negative Binomial(1.362)   
## Link function: log   
##   
## Formula:  
## Batpass\_sum ~ s(Locality, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean,   
## wind\_mean, k = c(25, 25))  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.97930 0.82462 -3.613 0.000303 \*\*\*  
## HabitatNatural 2.25330 0.72635 3.102 0.001921 \*\*   
## HabitatTurbinePad 2.24404 0.72624 3.090 0.002002 \*\*   
## temp\_mean 0.19850 0.02854 6.956 3.5e-12 \*\*\*  
## ---  
## 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) 5.8881 6 312.566 <2e-16 \*\*\*  
## s(jnight):HabitatMeteorologicalTower 0.2838 11 0.342 0.23   
## s(jnight):HabitatNatural 4.6293 11 33.252 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 7.2138 11 80.748 <2e-16 \*\*\*  
## s(wind\_mean) 3.7857 9 66.655 <2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 50.9340 81 206.836 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.419 Deviance explained = 69.7%  
## -REML = 3119.6 Scale est. = 1 n = 951

Family: Negative Binomial(1.362) Link function: log

Formula: cbind(Batpass\_sum) ~ s(Locality, bs = “re”) + s(jnight, by = Habitat, bs = “gp”) + Habitat + temp\_mean + s(wind\_mean) + ti(temp\_mean, wind\_mean, k = c(25, 25))

Parametric coefficients: Estimate Std. Error z value Pr(>|z|)  
(Intercept) -2.97930 0.82462 -3.613 0.000303  ***HabitatNatural 2.25330 0.72635 3.102 0.001921***  *HabitatTurbinePad 2.24404 0.72624 3.090 0.002002*  ***temp\_mean 0.19850 0.02854 6.956 3.5e-12***  — 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) 5.8881 6 312.566 <2e-16  ***s(jnight):HabitatMeteorologicalTower 0.2838 11 0.342 0.23***  
***s(jnight):HabitatNatural 4.6293 11 33.252 <2e-16***  s(jnight):HabitatTurbinePad 7.2138 11 80.748 <2e-16  ***s(wind\_mean) 3.7857 9 66.655 <2e-16***  ti(temp\_mean,wind\_mean) 50.9340 81 206.836 <2e-16 \*\*\* — Signif. codes: 0 ‘***’ 0.001 ’****’ 0.01 ’*’ 0.05 ‘.’ 0.1 ’ ’ 1

R-sq.(adj) = 0.419 Deviance explained = 69.7% -REML = 3119.6 Scale est. = 1 n = 951

bmp(file.path(output\_today, "diagnostic plots bats\_tot m20.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(2,2))  
gam.check(m20) #look at the plots, but also the output to check if k needs to be adjusted

##   
## Method: REML Optimizer: outer newton  
## full convergence after 7 iterations.  
## Gradient range [-0.0007920654,0.0009818129]  
## (score 3119.643 & scale 1).  
## Hessian positive definite, eigenvalue range [2.878821e-06,280.0782].  
## Model rank = 135 / 135   
##   
## 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) 8.000 5.888 NA NA   
## s(jnight):HabitatMeteorologicalTower 11.000 0.284 1.01 1.000   
## s(jnight):HabitatNatural 11.000 4.629 1.01 1.000   
## s(jnight):HabitatTurbinePad 11.000 7.214 1.01 1.000   
## s(wind\_mean) 9.000 3.786 1.00 1.000   
## ti(temp\_mean,wind\_mean) 81.000 50.934 0.88 0.085 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

gam.check(m20, rep=500) #

##   
## Method: REML Optimizer: outer newton  
## full convergence after 7 iterations.  
## Gradient range [-0.0007920654,0.0009818129]  
## (score 3119.643 & scale 1).  
## Hessian positive definite, eigenvalue range [2.878821e-06,280.0782].  
## Model rank = 135 / 135   
##   
## 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) 8.000 5.888 NA NA   
## s(jnight):HabitatMeteorologicalTower 11.000 0.284 1.01 1.000   
## s(jnight):HabitatNatural 11.000 4.629 1.01 0.995   
## s(jnight):HabitatTurbinePad 11.000 7.214 1.01 1.000   
## s(wind\_mean) 9.000 3.786 1.00 1.000   
## ti(temp\_mean,wind\_mean) 81.000 50.934 0.88 0.075 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

dev.off()

## png   
## 2

# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m20 <- sum( residuals(m20, "pearson")^2 ) / m20$df.residual  
overdispersion.m20

## [1] 1.150768

# 1.150768 - dispersion is fine and the k values fit too! TADA!   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.150768 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

bmp(file.path(output\_today, "checking k dimensions m20 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
  
plot(m20, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)  
  
draw(m20, scales ="free") #function 'draw' is from package 'gratia'  
draw(m20, scales ="fixed")  
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

bmp(file.path(output\_today, "3D gam m20 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m20, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m20, view = c("wind\_mean", "temp\_mean"))   
dev.off()

## png   
## 2

# Comparing m19 and m20

Both models perform similarily well at a glance, but how do they compare using BIC?

BIC(m19, m20)

## df BIC  
## m19 75.88930 6612.675  
## m20 84.26171 6535.970

# df BIC  
# m19 75.88930 6612.675  
# m20 84.26171 6535.970   
# the model with locality performs better, despite the higher degrees of freedom.   
#

# Make prediction figures

### Better visualize the wind and temperature cutoffs

### May be helpful to reference later:

<https://cran.r-project.org/web/packages/tidymv/vignettes/plot-smooths.html> <https://drmowinckels.io/blog/2019-11-16-plotting-gamm-interactions-with-ggplot2/> <https://www.imsbio.co.jp/RGM/R_rdfile?f=itsadug/man/pvisgam.Rd&d=R_CC>

# from this blog   
# https://drmowinckels.io/blog/2019-11-16-plotting-gamm-interactions-with-ggplot2/  
  
# library(itsadug)   
# pvisgam(m20, view = c("wind\_mean", "temp\_mean"))   
#   
# ct\_int\_pred <- expand.grid(temp\_mean= c(8,10, 12, 14, 16, 18, 20),  
# wind\_mean = c(2, 4, 6, 8, 10, 12),   
# Habitat = levels(bats\_tot$Habitat),   
# Locality = levels(bats\_tot$Locality),   
# jnight = seq(min(bats\_tot$jnight), max(bats\_tot$jnight), 1))  
#   
# ct\_int\_pred <- predict(m20, newdata = ct\_int\_pred,   
# se.fit = TRUE) %>%   
# as\_tibble() %>%   
# cbind(ct\_int\_pred)  
#   
# dim(ct\_int\_pred)  
# # 91728 7  
#   
# ggplot(ct\_int\_pred, aes(x=temp\_mean, y=wind\_mean)) +   
# geom\_tile(aes(fill = fit)) +  
# geom\_contour(aes(z = fit), colour = "white")  
#   
# ct\_int\_pred %>%  
# ggplot(aes(x=wind\_mean, y=fit)) +  
# geom\_line(aes(colour=wind\_mean))  
#   
# ## This does not look all that great unfortunately   
#   
# # from this:  
# # https://stackoverflow.com/questions/73738521/how-to-visualize-gam-results-with-contour-tile-plot-using-ggplot2  
#   
# # Try using the function exclude.too.far() to prune the data a bit   
#   
# drop <- exclude.too.far(ct\_int\_pred$temp\_mean, ct\_int\_pred$temp\_mean, bats\_tot$temp\_mean, bats\_tot$wind\_mean, dist = 0.1)  
#   
# df\_pred <- ct\_int\_pred |>  
# mutate(fitted = if\_else(drop, NA\_real, fitted))  
#   
# # Error in `mutate()`:  
# # ! Problem while computing `fitted = if\_else(drop, NA\_real\_, fitted)`.  
# # Caused by error in `if\_else()`:  
# # ! `false` must be a double vector, not a function.  
# # Run `rlang::last\_error()` to see where the error occurred.  
#   
# ------------------------------------------------------  
# summary(bats\_tot$temp\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 8.789 13.419 14.930 15.647 17.494 24.579   
#   
# summary(bats\_tot$wind\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 2.571 4.486 6.012 6.249 7.857 13.413   
#   
# #Scetching the results - difference between Habitats  
# pdata <- with(bats\_tot,  
# expand.grid(temp\_mean= c(8,10, 12, 14, 16, 18, 20),  
# wind\_mean = c(2, 4, 6, 8, 10, 12),   
# Habitat = levels(bats\_tot$Habitat),   
# Locality = levels(bats\_tot$Locality),   
# jnight = seq(min(jnight), max(jnight), 1))) #1 to keep the integer format  
#   
# summary(pdata)  
# head(pdata)  
# tail(pdata)  
#   
# fit <- data.frame(predict(m20, newdata=pdata, se.fit=TRUE, type = 'response'))  
# fit <- transform(fit, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
# pred <- cbind(pdata,fit)  
#   
# head(pred)  
#   
# plt2 <- ggplot(pred, aes(x=wind\_mean, y = fit)) +  
# geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5) +  
# geom\_line() + facet\_wrap( ~ Habitat, scales ='free\_y') +  
# labs(x = "Mean wind speeds per night (m/s)", y ="Bat passes per night")  
# plt2 # doesn't make much sense to have negative bat passes...   
  
  
# this approach for plotting predictions make a ginormous data set which is pretty much impossible to plot unfortunately.   
  
# m20\_p <- predict\_gam(m20)  
# m20\_p   
# dim(m20\_p)  
# #3000000 7  
#   
# summary(m20\_p)  
# summary(m20\_p$temp\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 8.789 12.656 16.684 16.684 20.712 24.579   
# summary(m20\_p$wind\_mean)  
 # Min. 1st Qu. Median Mean 3rd Qu. Max.   
 # 2.571 5.226 7.992 7.992 10.758 13.413   
  
#write.csv(m20\_p, file.path(output\_today, "prediction model 20.csv"))  
  
# This crashes my computer unfortunately: need to create a smaller preditciton dataset probably.   
  
# p <- m20\_p %>% ggplot(aes(x = temp\_mean, y = fit)) + geom\_smooth\_ci(Habitat)  
# p   
#   
#   
# p1 <- m20\_p %>% ggplot(aes(x = wind\_mean, y = fit)) + geom\_smooth\_ci(Habitat)  
# p1

# m21

### Facility rather than Locality as a random effect

### try something similar to model 20, with some slightly different smoother terms

### Response variable = Batpass\_sum

summary(bats\_tot$wind\_mean)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.571 4.486 6.012 6.249 7.857 13.413

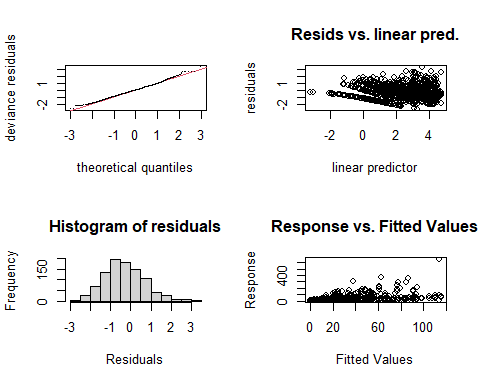
summary(bats\_tot$temp\_mean)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 8.789 13.419 14.930 15.647 17.494 24.579

# Min. 1st Qu. Median Mean 3rd Qu. Max.   
# 2.571 4.486 6.012 6.249 7.857 13.413   
# > summary(bats\_tot$temp\_mean)  
# Min. 1st Qu. Median Mean 3rd Qu. Max.   
# 8.789 13.419 14.930 15.647 17.494 24.579  
  
# Trying with some slightly different smoother terms   
  
m21 <- gam(Batpass\_sum ~  
 s(Facility, bs = "re") +   
 s(jnight, by = Habitat, bs = "gp") +   
 Habitat +   
 temp\_mean +   
 s(wind\_mean, bs = "cc", k = 10) +   
 ti(temp\_mean,wind\_mean, k = c(25,25), bs = "cc"),   
 data = bats\_tot, method = "REML",   
 family = nb(), select=TRUE)   
  
summary(m21)

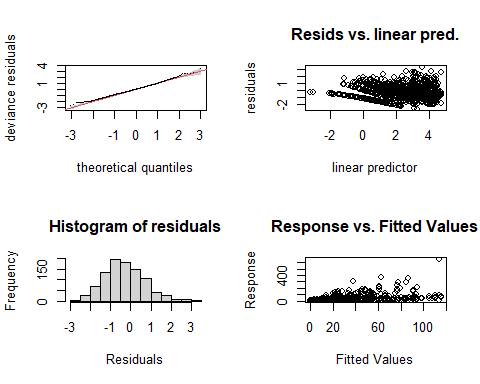
##   
## Family: Negative Binomial(1.136)   
## Link function: log   
##   
## Formula:  
## Batpass\_sum ~ s(Facility, bs = "re") + s(jnight, by = Habitat,   
## bs = "gp") + Habitat + temp\_mean + s(wind\_mean, bs = "cc",   
## k = 10) + ti(temp\_mean, wind\_mean, k = c(25, 25), bs = "cc")  
##   
## Parametric coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.5605 0.6874 -5.18 2.22e-07 \*\*\*  
## HabitatNatural 2.8489 0.1342 21.23 < 2e-16 \*\*\*  
## HabitatTurbinePad 2.8281 0.1336 21.17 < 2e-16 \*\*\*  
## temp\_mean 0.2063 0.0268 7.70 1.36e-14 \*\*\*  
## ---  
## 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(Facility) 0.9949 1 190.782 <2e-16 \*\*\*  
## s(jnight):HabitatMeteorologicalTower 0.3259 11 0.423 0.223   
## s(jnight):HabitatNatural 5.5401 11 61.143 <2e-16 \*\*\*  
## s(jnight):HabitatTurbinePad 7.4695 11 96.958 <2e-16 \*\*\*  
## s(wind\_mean) 3.8224 8 48.500 <2e-16 \*\*\*  
## ti(temp\_mean,wind\_mean) 47.1721 82 151.769 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## R-sq.(adj) = 0.238 Deviance explained = 64.8%  
## -REML = 3168.6 Scale est. = 1 n = 951

windows()  
par(mfrow = c(2,2))  
gam.check(m21) #look at the plots, but also the output to check if k needs to be adjusted



##   
## Method: REML Optimizer: outer newton  
## full convergence after 7 iterations.  
## Gradient range [-0.0004979467,0.0002692148]  
## (score 3168.622 & scale 1).  
## Hessian positive definite, eigenvalue range [6.439617e-07,314.5129].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.000 0.995 NA NA  
## s(jnight):HabitatMeteorologicalTower 11.000 0.326 1.02 1.0  
## s(jnight):HabitatNatural 11.000 5.540 1.02 1.0  
## s(jnight):HabitatTurbinePad 11.000 7.469 1.02 1.0  
## s(wind\_mean) 8.000 3.822 1.01 1.0  
## ti(temp\_mean,wind\_mean) 82.000 47.172 0.88 0.2

gam.check(m21, 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 7 iterations.  
## Gradient range [-0.0004979467,0.0002692148]  
## (score 3168.622 & scale 1).  
## Hessian positive definite, eigenvalue range [6.439617e-07,314.5129].  
## Model rank = 129 / 129   
##   
## 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(Facility) 2.000 0.995 NA NA  
## s(jnight):HabitatMeteorologicalTower 11.000 0.326 1.02 1.00  
## s(jnight):HabitatNatural 11.000 5.540 1.02 1.00  
## s(jnight):HabitatTurbinePad 11.000 7.469 1.02 1.00  
## s(wind\_mean) 8.000 3.822 1.01 1.00  
## ti(temp\_mean,wind\_mean) 82.000 47.172 0.88 0.21

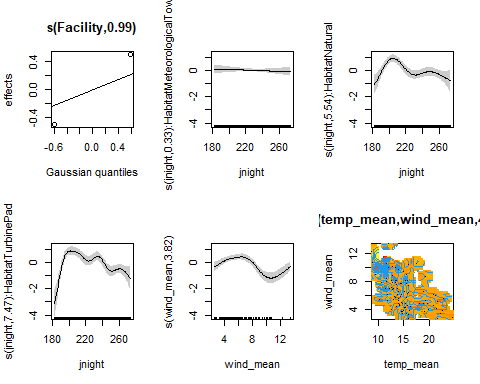
# Now the smooth terms pass the basis dimension test!   
  
overdispersion.m21 <- sum( residuals(m21, "pearson")^2 ) / m21$df.residual  
overdispersion.m21

## [1] 1.110624

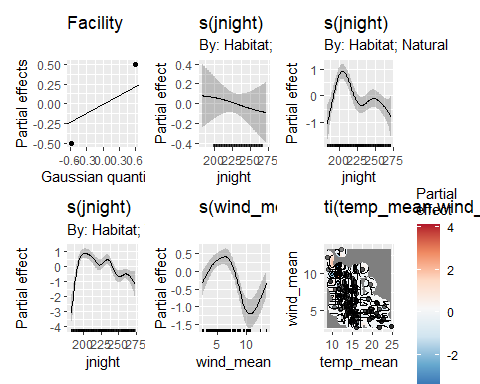
# 1.110624 - k indicies are fixed and no over / under dispersion   
  
# From Katrine's Step3B GAM modelling script:   
#[1] 1.022701 #this value should ideally be 0.8-1.2, but it's not very bad!

### Basis dimension (k) checking results:

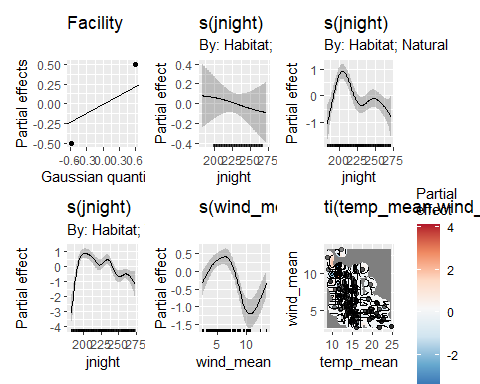
bmp(file.path(output\_today, "checking k dimensions m21 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
windows()  
plot(m21, pages=1,scheme=2,shade=TRUE,seWithMean = TRUE)



draw(m21, scales ="free") #function 'draw' is from package 'gratia'



draw(m21, scales ="fixed")



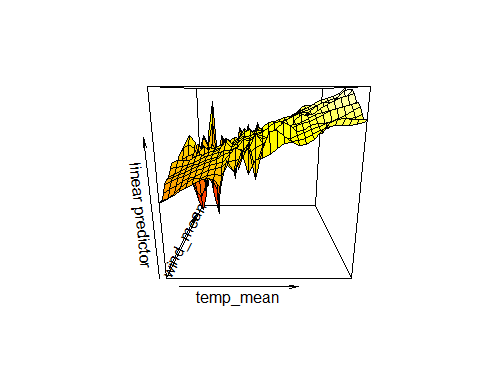
# wind is too wiggly, but there is no relationship for the met tower(probably too little data...)   
dev.off()

## png   
## 2

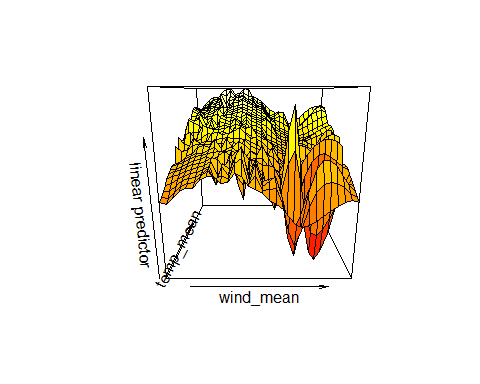
windows()  
bmp(file.path(output\_today, "3D gam m21 bats\_tot.jpg"), width = 9, height = 6, units = "in", res = 350)  
par(mfrow = c(1,2))  
vis.gam(m21, view = c("temp\_mean", "wind\_mean"))   
vis.gam(m21, view = c("wind\_mean", "temp\_mean")) # still too wiggly though....   
dev.off()

## png   
## 2

vis.gam(m21, view = c("temp\_mean", "wind\_mean"))



vis.gam(m21, view = c("wind\_mean", "temp\_mean"))



# Make prediction figures

### Better visualize the wind and temperature cutoffs

### May be helpful to reference later:

<https://cran.r-project.org/web/packages/tidymv/vignettes/plot-smooths.html> <https://drmowinckels.io/blog/2019-11-16-plotting-gamm-interactions-with-ggplot2/> <https://www.imsbio.co.jp/RGM/R_rdfile?f=itsadug/man/pvisgam.Rd&d=R_CC>

names(bats\_tot)

## [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"

# from this blog   
# https://drmowinckels.io/blog/2019-11-16-plotting-gamm-interactions-with-ggplot2/  
  
# A few different ways to make predictions   
  
## option 1   
  
library(itsadug)

## Loading required package: plotfunctions

##   
## Attaching package: 'plotfunctions'

## The following object is masked from 'package:ggplot2':  
##   
## alpha

## Loaded package itsadug 2.4 (see 'help("itsadug")' ).

##   
## Attaching package: 'itsadug'

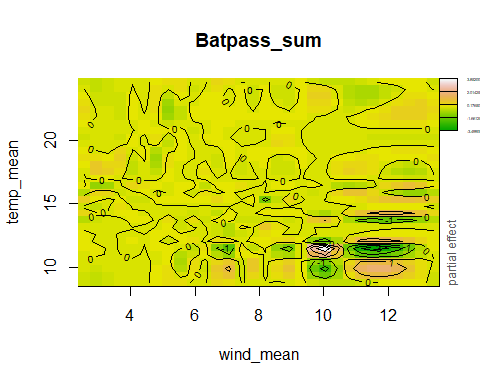
## The following objects are masked from 'package:tidymv':  
##   
## find\_difference, get\_difference, summary\_data

## The following object is masked from 'package:renv':  
##   
## diagnostics

pvisgam(m21, view = c("wind\_mean", "temp\_mean"))

## Tensor(s) to be plotted: ti(temp\_mean,wind\_mean)

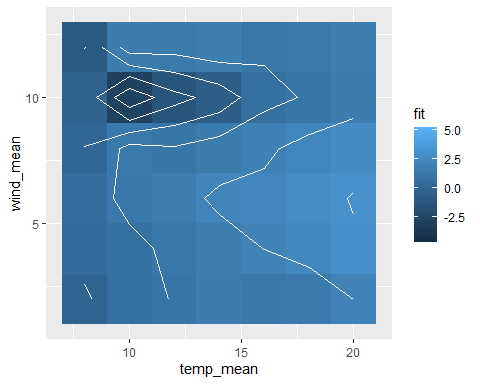
## Warning in gradientLegend(round(c(min.z, max.z), 3), n.seg = 3, pos = 0.875, :  
## Increase right margin to fit labels or decrease the number of decimals, see  
## help(gradientLegend).



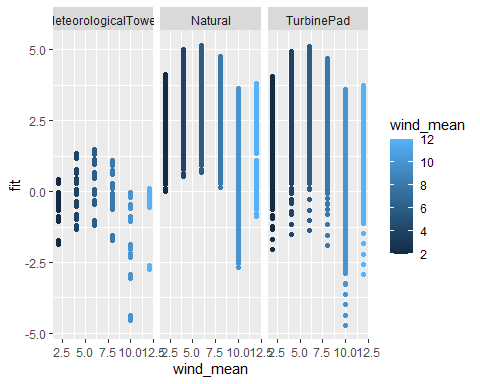
ct\_int\_pred <- expand.grid(temp\_mean= c(8,10, 12, 14, 16, 18, 20),  
 wind\_mean = c(2, 4, 6, 8, 10, 12),  
 Habitat = levels(bats\_tot$Habitat),  
 Facility = "South",  
 jnight = seq(min(bats\_tot$jnight), max(bats\_tot$jnight), 1))  
  
ct\_int\_pred1 <- predict(m21, newdata = ct\_int\_pred,  
 se.fit = TRUE) %>%  
 as\_tibble() %>%  
 cbind(ct\_int\_pred)  
  
############################################  
  
# # option 2   
  
fit <- predict.gam(m21, newdata = ct\_int\_pred, type = "response")  
ct\_int\_pred2 <- data.frame(ct\_int\_pred, fit)  
summary(ct\_int\_pred2)

## temp\_mean wind\_mean Habitat Facility   
## Min. : 8 Min. : 2 MeteorologicalTower:3822 South:11466   
## 1st Qu.:10 1st Qu.: 4 Natural :3822   
## Median :14 Median : 7 TurbinePad :3822   
## Mean :14 Mean : 7   
## 3rd Qu.:18 3rd Qu.:10   
## Max. :20 Max. :12   
## jnight fit   
## Min. :183 Min. : 0.0089   
## 1st Qu.:205 1st Qu.: 0.9302   
## Median :228 Median : 5.2169   
## Mean :228 Mean : 13.0302   
## 3rd Qu.:251 3rd Qu.: 16.4448   
## Max. :273 Max. :175.7510

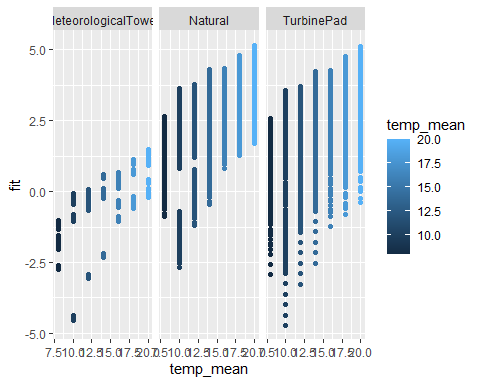
############################################  
# Plotting predicitons   
  
ggplot(ct\_int\_pred1, aes(x=temp\_mean, y=wind\_mean)) +  
 geom\_tile(aes(fill = fit)) +  
 geom\_contour(aes(z = fit), colour = "white")



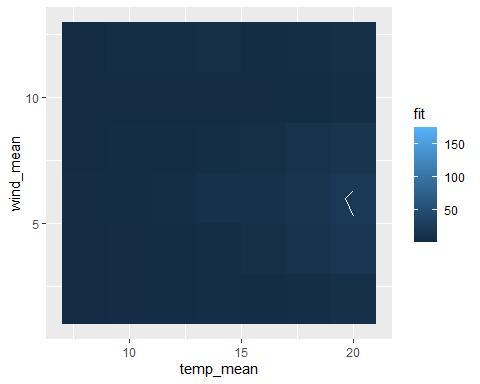
ct\_int\_pred1 %>%  
 ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point(aes(colour=wind\_mean)) + facet\_wrap(~Habitat)



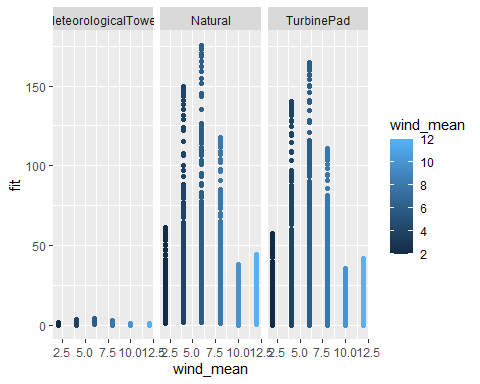
ct\_int\_pred1 %>%  
 ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point(aes(colour=temp\_mean)) + facet\_wrap(~Habitat)



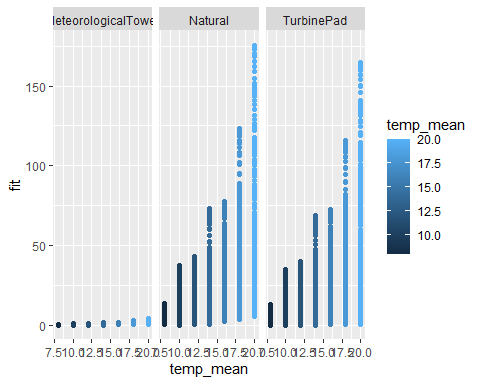
#   
# ## This still does not look all that great unfortunately   
windows()  
ggplot(ct\_int\_pred2, aes(x=temp\_mean, y=wind\_mean)) +  
 geom\_tile(aes(fill = fit)) +  
 geom\_contour(aes(z = fit), colour = "white")



ct\_int\_pred2 %>%  
 ggplot(aes(x=wind\_mean, y=fit)) +  
 geom\_point(aes(colour=wind\_mean)) + facet\_wrap(~Habitat)



ct\_int\_pred2 %>%  
 ggplot(aes(x=temp\_mean, y=fit)) +  
 geom\_point(aes(colour=temp\_mean)) + facet\_wrap(~Habitat)



# # from this:  
# # https://stackoverflow.com/questions/73738521/how-to-visualize-gam-results-with-contour-tile-plot-using-ggplot2  
#   
# # Try using the function exclude.too.far() to prune the data a bit   
#   
# drop <- exclude.too.far(ct\_int\_pred$temp\_mean, ct\_int\_pred$temp\_mean, bats\_tot$temp\_mean, bats\_tot$wind\_mean, dist = 0.1)  
#   
# df\_pred <- ct\_int\_pred |>  
# mutate(fitted = if\_else(drop, NA\_real, fitted))  
#   
# # Error in `mutate()`:  
# # ! Problem while computing `fitted = if\_else(drop, NA\_real\_, fitted)`.  
# # Caused by error in `if\_else()`:  
# # ! `false` must be a double vector, not a function.  
# # Run `rlang::last\_error()` to see where the error occurred.  
#   
# ------------------------------------------------------  
# summary(bats\_tot$temp\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 8.789 13.419 14.930 15.647 17.494 24.579   
#   
# summary(bats\_tot$wind\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 2.571 4.486 6.012 6.249 7.857 13.413   
#   
# #Scetching the results - difference between Habitats  
# pdata <- with(bats\_tot,  
# expand.grid(temp\_mean= c(8,10, 12, 14, 16, 18, 20),  
# wind\_mean = c(2, 4, 6, 8, 10, 12),   
# Habitat = levels(bats\_tot$Habitat),   
# Locality = levels(bats\_tot$Locality),   
# jnight = seq(min(jnight), max(jnight), 1))) #1 to keep the integer format  
#   
# summary(pdata)  
# head(pdata)  
# tail(pdata)  
#   
# fit <- data.frame(predict(m20, newdata=pdata, se.fit=TRUE, type = 'response'))  
# fit <- transform(fit, upper = fit + (2\*se.fit), lower = fit-(2\*se.fit))  
# pred <- cbind(pdata,fit)  
#   
# head(pred)  
#   
# plt2 <- ggplot(pred, aes(x=wind\_mean, y = fit)) +  
# geom\_ribbon(aes(ymin = lower, ymax = upper), fill = 'grey', alpha = 0.5) +  
# geom\_line() + facet\_wrap( ~ Habitat, scales ='free\_y') +  
# labs(x = "Mean wind speeds per night (m/s)", y ="Bat passes per night")  
# plt2 # doesn't make much sense to have negative bat passes...   
  
  
# this approach for plotting predictions make a ginormous data set which is pretty much impossible to plot unfortunately.   
  
# m20\_p <- predict\_gam(m20)  
# m20\_p   
# dim(m20\_p)  
# #3000000 7  
#   
# summary(m20\_p)  
# summary(m20\_p$temp\_mean)  
# # Min. 1st Qu. Median Mean 3rd Qu. Max.   
# # 8.789 12.656 16.684 16.684 20.712 24.579   
# summary(m20\_p$wind\_mean)  
 # Min. 1st Qu. Median Mean 3rd Qu. Max.   
 # 2.571 5.226 7.992 7.992 10.758 13.413   
  
#write.csv(m20\_p, file.path(output\_today, "prediction model 20.csv"))  
  
# This crashes my computer unfortunately: need to create a smaller preditciton dataset probably.   
  
# p <- m20\_p %>% ggplot(aes(x = temp\_mean, y = fit)) + geom\_smooth\_ci(Habitat)  
# p   
#   
#   
# p1 <- m20\_p %>% ggplot(aes(x = wind\_mean, y = fit)) + geom\_smooth\_ci(Habitat)  
# p1