Analysis 2021

Stewart Brehaut

9/10/2021

Please install and load the following packages. Also, you may have to have Rtools installed on your system or you may be unable to load some of the libraries.

```
library(rnoaa)
```

```
## Warning: package 'rnoaa' was built under R version 4.0.5
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## Registered S3 method overwritten by 'httr':
##   method           from
##   print.cache_info hoardr
```

```
## -- Attaching packages --------------------------------------- tidyve
rse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.3     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.0     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
## Warning: package 'tibble' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## Warning: package 'forcats' was built under R version 4.0.5
```

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

```
library(tidyr)
library(magrittr)
```

```
## Warning: package 'magrittr' was built under R version 4.0.5

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract

library(dplyr)
library(lubridate)

## Warning: package 'lubridate' was built under R version 4.0.5

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(ISLR)

## Warning: package 'ISLR' was built under R version 4.0.5

library(skimr)

## Warning: package 'skimr' was built under R version 4.0.5

library(RcppRoll)

## Warning: package 'RcppRoll' was built under R version 4.0.5

library(mgcv)

## Loading required package: nlme

##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse

## This is mgcv 1.8-31. For overview type 'help("mgcv-package")'.

library(gamair)

## Warning: package 'gamair' was built under R version 4.0.5

library(broom)
```

```
## Warning: package 'broom' was built under R version 4.0.5
```

```
library(fitdistrplus)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Loading required package: survival
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
library(tibble)
library(ggplot2)
```

```
options(noaakey = "pSWkTBgIBNbEgIsfxzIjFOMsAOANVGxN")
perth <- ncdc( datasetid = 'GHCND', stationid = 'GHCND:ASN00009021',
               startdate ='2013-07-01', enddate = '2014-06-30', limit =
 1000)$data
```

I have used data from the National Centers for Environmental Information (NOAA). The NOAA has a large amount of weather data from Perth.

```
nrow(perth)
```

```
## [1] 1000
```

The number of rows in the data set is 1000. The data was limited to 1000 rows when I downloaded the data.

```
perth %>%
summarise(start_date = min(date),
end_date = max(date))
```

```
## # A tibble: 1 x 2
##   start_date          end_date
```

```
##   <chr>               <chr>
## 1 2013-07-01T00:00:00 2014-03-07T00:00:00
```

The time period covered by the data is July 1st 2013 to March 7th 2014. Because we were limited to 1000 rows of data.

Model Planning -

The final model will be used to help solve the overcrowding in Western Australian hospitals. Improving the number of patients able to see a doctor before leaving, improving patient satisfaction, and other metrics will improve the Western Australian hospital system.

The potential users of the model include;

The public, the Western Australian state government, the Western Australian health system. The public will benefit knowing how long their wait time could be and how efficient the system is. The state government can use the model to reduce health expenditure and better focus investment. The health system can use the model to improve health efficiency and provide a better service and reduce wait times.

Relationship and data -

I want to explore the relationship between the date and attendance. I imagine there will be more patients arriving at a hospital on certain dates. The date or time of year also indicates the weather. The response variable is the date and the predictor variable is Attendance. The variables are collected routinely enough (every day) and they are made available soon enough to be used for prediction. A linear model is the most simple to implement and the least computationally expensive. The data has been collected for a long time but the data we're using is only of less than a one year period, part of that period was an extreme heat wave which might not happen again for a long time so data in the future may not have the same characteristics in the future. This is a limitation.

Statistical methods used to generate the model include Linear Model (LM) and a General Additive Model (GAM) which includes a multivariate GAM. An LM is being used because it is a model that describes response variables in a linear combination of predictor variables. It is simple, easy to interpret, and we can use it to give us more information about the curves in the data. The second model used is a GAM. A GAM allows us to model non-linear data while it will still be explainable and provide us insight. The relationship between hospital attendance and the weather could be linear or non-linear so both model approaches are required.

Re-using old code

```
ed_data_link <- 'govhack3.csv'
top_row <- read_csv(ed_data_link, col_names = FALSE, n_max = 1)

## Rows: 1 Columns: 64
```

```
## -- Column specification --------------------------------------------
------------
## Delimiter: ","
## chr  (9): X2, X9, X16, X23, X30, X37, X44, X51, X58
## lgl (55): X1, X3, X4, X5, X6, X7, X8, X10, X11, X12, X13, X14, X15,
X17, X18...

##
## i Use `spec()` to retrieve the full column specification for this da
ta.
## i Specify the column types or set `show_col_types = FALSE` to quiet
this message.
```

```
second_row <- read_csv(ed_data_link, n_max = 1)
```

```
## New names:
## * `` -> ...1
## * `` -> ...3
## * `` -> ...4
## * `` -> ...5
## * `` -> ...6
## * ...
```

```
## Rows: 1 Columns: 64
```

```
## -- Column specification --------------------------------------------
------------
## Delimiter: ","
## chr (64): ...1, Royal Perth Hospital, ...3, ...4, ...5, ...6, ...7,
...8, Fr...

##
## i Use `spec()` to retrieve the full column specification for this da
ta.
## i Specify the column types or set `show_col_types = FALSE` to quiet
this message.
```

```
column_names <- second_row %>%
  unlist(., use.names=FALSE) %>%
  make.unique(., sep = "__") # double underscore

column_names[2:8] <- str_c(column_names[2:8], '0', sep='__')

daily_attendance <-
  read_csv(ed_data_link, skip = 2, col_names = column_names)
```

```
## Rows: 365 Columns: 64
```

```
## -- Column specification --------------------------------------------
------------
## Delimiter: ","
## chr (23): Date, Tri_1__0, Tri_5__0, Tri_1__1, Tri_1__2, Tri_2__2, Tr
```

```
i_5__2, ...
## dbl (41): Attendance__0, Admissions__0, Tri_2__0, Tri_3__0, Tri_4__0
, Attend...

##
## i Use `spec()` to retrieve the full column specification for this da
ta.
## i Specify the column types or set `show_col_types = FALSE` to quiet
this message.
```

```
(
hospitalnames <- top_row %>%
unlist(., use.names=FALSE) %>%
na.omit()
)
```

```
## [1] "Royal Perth Hospital"
## [2] "Fremantle Hospital"
## [3] "Princess Margaret Hospital For Children"
## [4] "King Edward Memorial Hospital For Women"
## [5] "Sir Charles Gairdner Hospital"
## [6] "Armadale/Kelmscott District Memorial Hospital"
## [7] "Swan District Hospital"
## [8] "Rockingham General Hospital"
## [9] "Joondalup Health Campus"
## attr(,"na.action")
##  [1]  1  3  4  5  6  7  8 10 11 12 13 14 15 17 18 19 20 21 22 24 25
26 27 28 29
## [26] 31 32 33 34 35 36 38 39 40 41 42 43 45 46 47 48 49 50 52 53 54
55 56 57 59
## [51] 60 61 62 63 64
## attr(,"class")
## [1] "omit"
```

```
daily_attendance <- daily_attendance %>%
gather(key = list_var,
value = values,
-Date)

daily_attendance <- daily_attendance %>%
separate(list_var,
into = c("list_var",
"hospital_number"),
sep="__",
remove=TRUE) %>%
mutate(hospital_number = as.numeric(hospital_number) + 1) %>%
mutate(hospitalnames = hospitalnames[hospital_number]) %>%
dplyr::select(Date, hospitalnames, list_var, values)

daily_attendance <- daily_attendance %>% spread(list_var,values)
```

```
daily_attendance <- daily_attendance %>% mutate(Date = dmy(Date))

daily_attendance <- daily_attendance %>%
     mutate_at(c(3:9), as.numeric)

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

daily_attendance <- daily_attendance %>%
  mutate_if(is.numeric, funs(ifelse(is.na(.), 0, .)))

## Warning: `funs()` was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))

select_hospital <- 'Rockingham General Hospital'
rockingham <- daily_attendance %>%
filter(hospitalnames == select_hospital) %>%
dplyr::select(-hospitalnames) %>%
arrange(Date)
```

For these models we will be using the Attendance from the Rockingham General Hospital.

```
rockingham <- transform(rockingham, day = as.numeric(format(Date, '%j')
))
```

This creates the day column for the rockingham hospital, Janurary 1st is Day 1. So this datasets begins on day 182, July 1st.

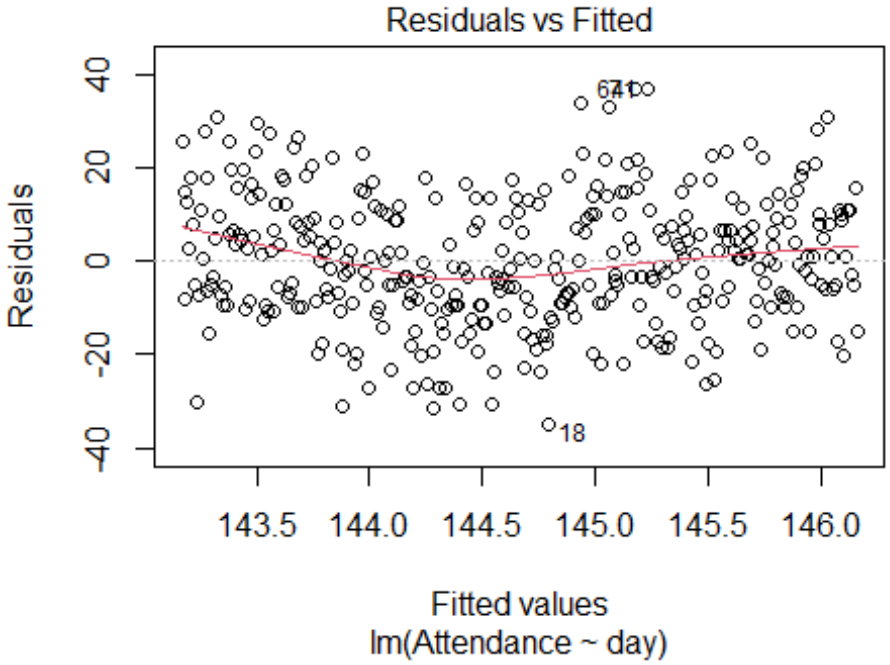
```
lm_rockingham <- lm(Attendance~day , data = rockingham)
lm_rockingham
```

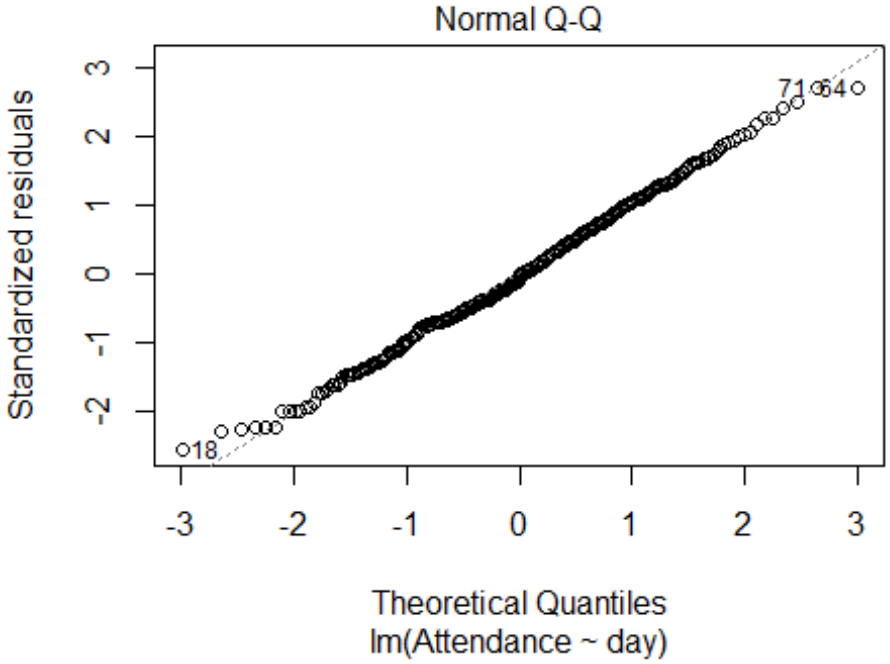
```
##
## Call:
## lm(formula = Attendance ~ day, data = rockingham)
##
## Coefficients:
## (Intercept)          day
##   1.432e+02    8.232e-03
```

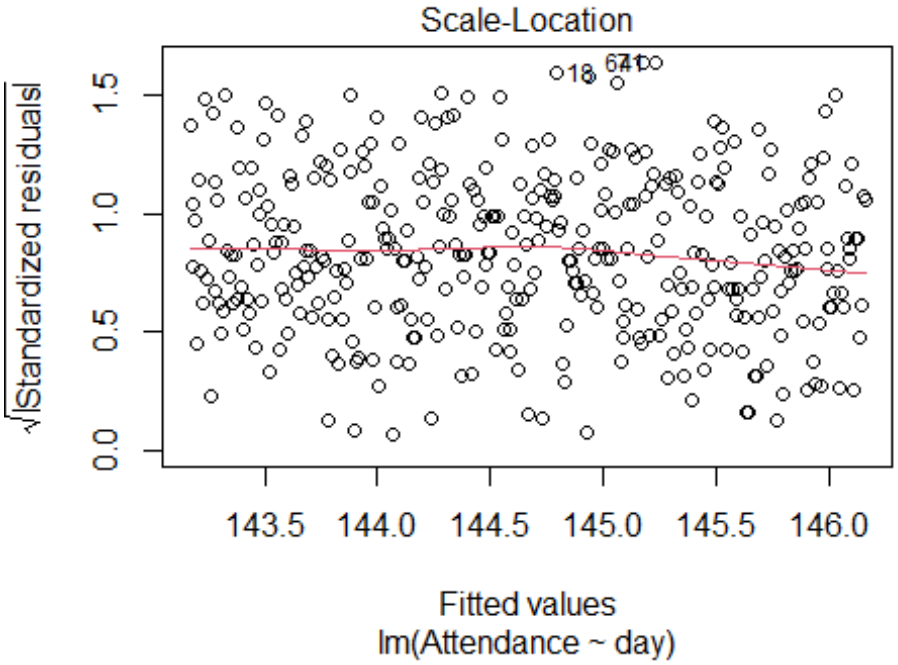
```
lm_rockingham %>%
  summary
```

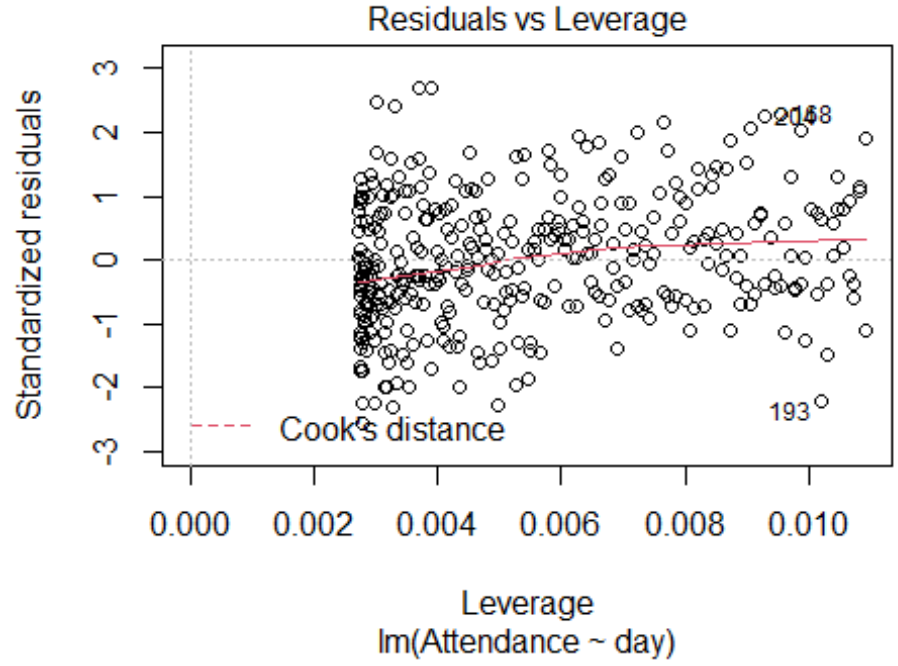
```
##
## Call:
## lm(formula = Attendance ~ day, data = rockingham)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.797  -9.201  -0.802   9.593  36.824
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.432e+02  1.436e+00   99.67   <2e-16 ***
## day         8.232e-03  6.802e-03    1.21    0.227
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.69 on 363 degrees of freedom
## Multiple R-squared:  0.004018,   Adjusted R-squared:  0.001274
## F-statistic: 1.465 on 1 and 363 DF,  p-value: 0.227
```

```
plot(lm_rockingham)
```


Residuals vs Fitted
Residuals
Fitted values
lm(Attendance ~ day)
18
64
71


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Attendance ~ day)
18
64
71





In this LM there are some outliers or influential observations that have the potential to skew our data. In the Residuals vs Leverage plot you can see 2 or 3 outliers in the top right, and one outlier on the bottom right "193". There are at least three observable outliers. There are also many other observations which are close to the outliers

With this many outliers they should not be removed. This plot suggest we should take a non-linear approach to modeling Attendance and Date for Rockingham. The other plots produced by the LM also suggest non-linearity because of the high number of outliers and the uneven spread, the fit is not great. This is why we must also fit a GAM.

```
gam_rockingham <- gam(Attendance ~ s(day),data=rockingham,method="REML"
)
gam_rockingham

##
## Family: gaussian
## Link function: identity
##
## Formula:
## Attendance ~ s(day)
##
## Estimated degrees of freedom:
## 4.6  total = 5.6
##
## REML score: 1459.964

summary(gam_rockingham)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## Attendance ~ s(day)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  144.666      0.687   210.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##          edf Ref.df     F  p-value
## s(day) 4.603  5.645 5.876 1.32e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0824   Deviance explained =  9.4%
## -REML =   1460  Scale est. = 172.27    n = 365

coef(gam_rockingham)

## (Intercept)    s(day).1    s(day).2    s(day).3    s(day).4    s(day
).5
##  144.665753   -6.668068    8.293167    4.241071    2.391315    1.195
```

```
775
##    s(day).6    s(day).7    s(day).8    s(day).9
##    2.817916   -0.971886    2.262498   -1.294835
```

```
augment(gam_rockingham)
```

```
## # A tibble: 365 x 8
##    Attendance   day .fitted .se.fit .resid   .hat .sigma  .cooksd
##         <dbl> <dbl>   <dbl>   <dbl>  <dbl>  <dbl> <lgl>     <dbl>
##  1        155   182    140.    1.47  15.0  0.0125 NA     0.00300
##  2        145   183    140.    1.47   4.90 0.0125 NA     0.000320
##  3        151   184    140.    1.47  10.8  0.0125 NA     0.00155
##  4        129   185    140.    1.47 -11.3  0.0125 NA     0.00171
##  5        122   186    140.    1.47 -18.4  0.0125 NA     0.00453
##  6        137   187    141.    1.47  -3.55 0.0125 NA     0.000168
##  7        158   188    141.    1.47  17.3  0.0126 NA     0.00401
##  8        134   189    141.    1.47  -6.79 0.0126 NA     0.000614
##  9        128   190    141.    1.47 -12.9  0.0126 NA     0.00222
## 10        145   191    141.    1.47   3.97 0.0126 NA     0.000211
## # ... with 355 more rows
```

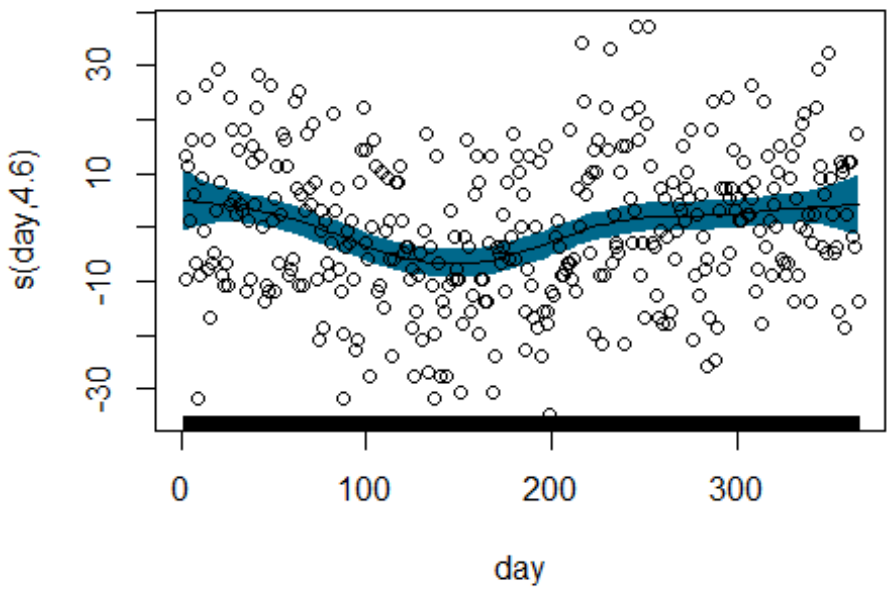
```
glance(gam_rockingham)
```

```
## # A tibble: 1 x 7
##      df logLik   AIC   BIC deviance df.residual  nobs
##   <dbl>  <dbl> <dbl> <dbl>    <dbl>       <dbl> <int>
## 1  5.60 -1455. 2925. 2955.   61913.        359.   365
```

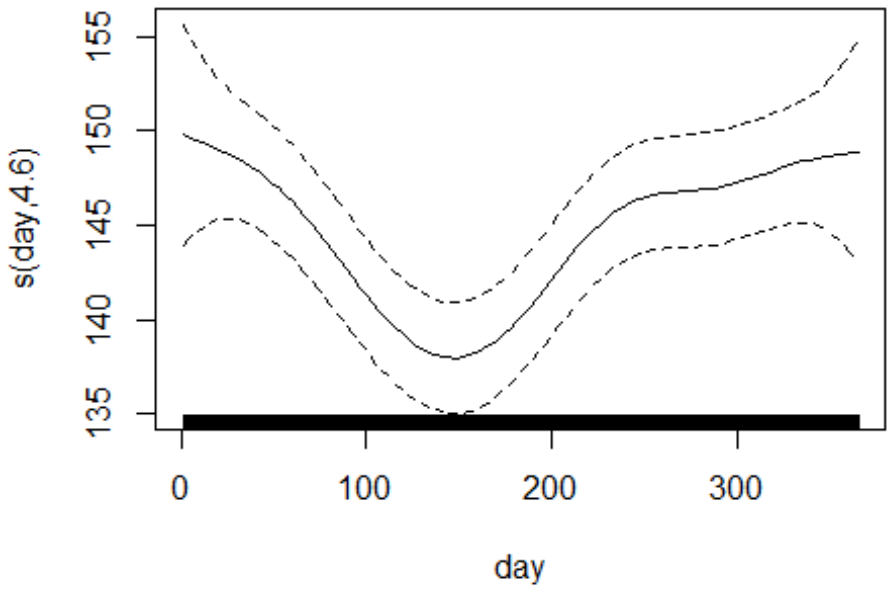
```
tidy(gam_rockingham)
```

```
## # A tibble: 1 x 5
##   term     edf ref.df statistic   p.value
##   <chr>  <dbl>  <dbl>     <dbl>     <dbl>
## 1 s(day)  4.60   5.64      5.88 0.0000132
```

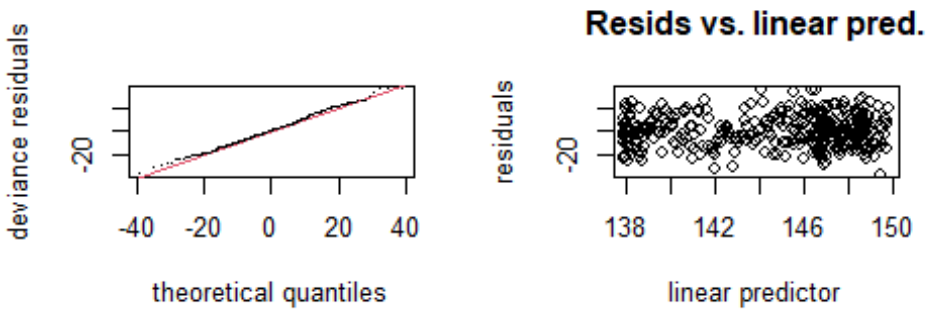
```
plot(gam_rockingham, rug=TRUE,residuals = TRUE, pch=1, cex=1, shade=TRU
E, shade.col="deepskyblue4")
```

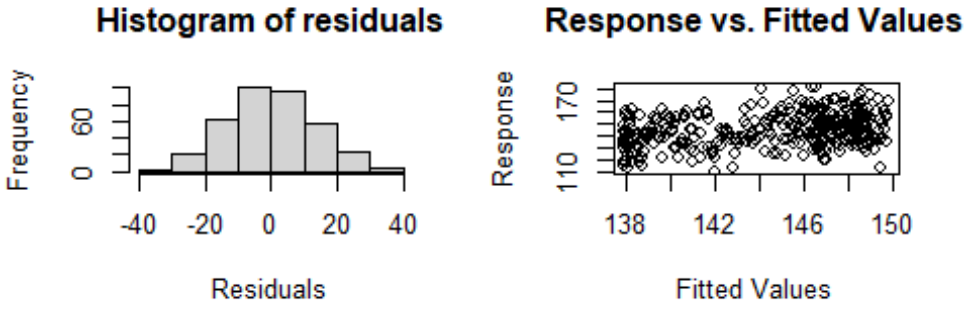



```r
plot(gam_rockingham, seWithMean = TRUE, shift=coef(gam_rockingham)[1])
```




```r
gam.check(gam_rockingham)
```






```
##
## Method: REML   Optimizer: outer newton
## full convergence after 6 iterations.
## Gradient range [-3.641487e-07,3.831252e-09]
## (score 1459.964 & scale 172.2693).
## Hessian positive definite, eigenvalue range [0.7818649,181.518].
## Model rank =  10 / 10
##
## 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(day) 9.0 4.6    0.82  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
concurvity(gam_rockingham, full=TRUE)
```

```
##                  para       s(day)
## worst    3.040639e-25 3.061451e-25
## observed 3.040639e-25 2.206430e-28
## estimate 3.040639e-25 6.425409e-28
```

We only have one variable, so we don't need to give attention to concurvity.

The generated data above shows the p-value being so low is significant, this suggests the residuals are not just randomly distributed. The GAM has 9 basis

functions, thus it has 9 coefficients. It has 6 iterations The model would be better if there were more basis functions. For example, a basis function of 20 encompasses a larger function space than 9. More basis functions would make the model smoother and better. The coefficients for the GAM are better than the LM.

The plots above show the average number of people arriving at Rockingham hospital is around 150 a day, when other variables are average. The GAM explains 9.4% of the variance, which isn't a great result but better than the LM.

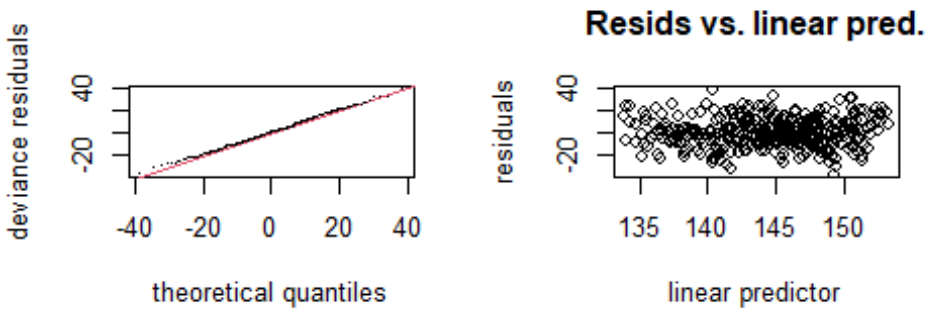
```
rockingham <- transform(rockingham, week = as.numeric(format(Date, '%W'
)))
```

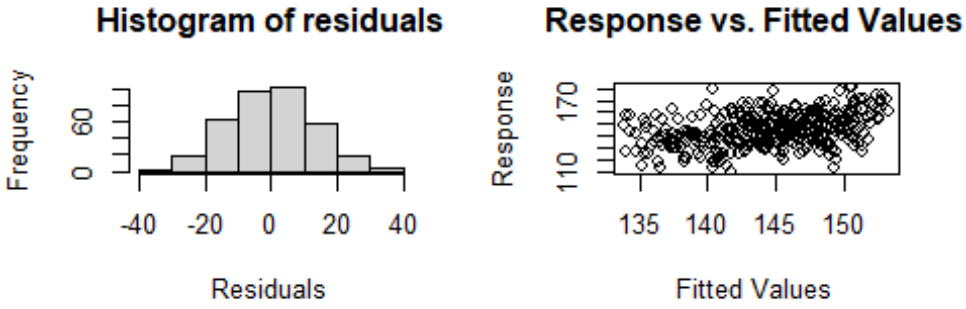
This adds a week column to the rockingham dataset. The 1st week of Janurary is week 0 the last week of the year is 52.

```
gam_rockingham2 <- gam(Attendance ~ s(day) + s(week),data=rockingham,me
thod="REML")
gam_rockingham2
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Attendance ~ s(day) + s(week)
##
## Estimated degrees of freedom:
## 4.97 1.01  total = 6.98
##
## REML score: 1448.347
```

```
gam.check(gam_rockingham2)
```






```
##
## Method: REML   Optimizer: outer newton
## full convergence after 9 iterations.
## Gradient range [-0.001367172,0.004004984]
## (score 1448.347 & scale 165.4781).
## Hessian positive definite, eigenvalue range [0.0004381903,181.0178].
## Model rank =  19 / 19
##
## 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(day)  9.00 4.97    0.83  <2e-16 ***
## s(week) 9.00 1.01    0.82  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(gam_rockingham2)
```

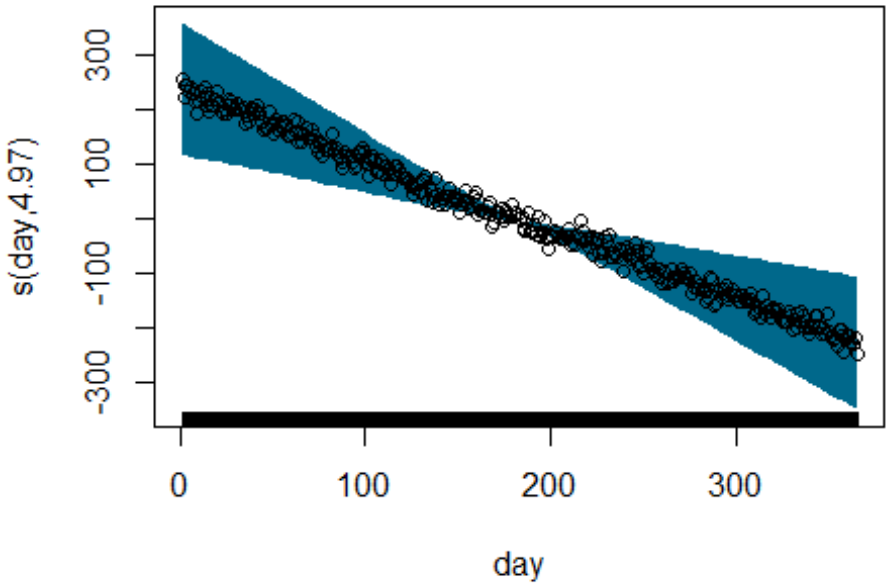
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Attendance ~ s(day) + s(week)
##
## Parametric coefficients:
```

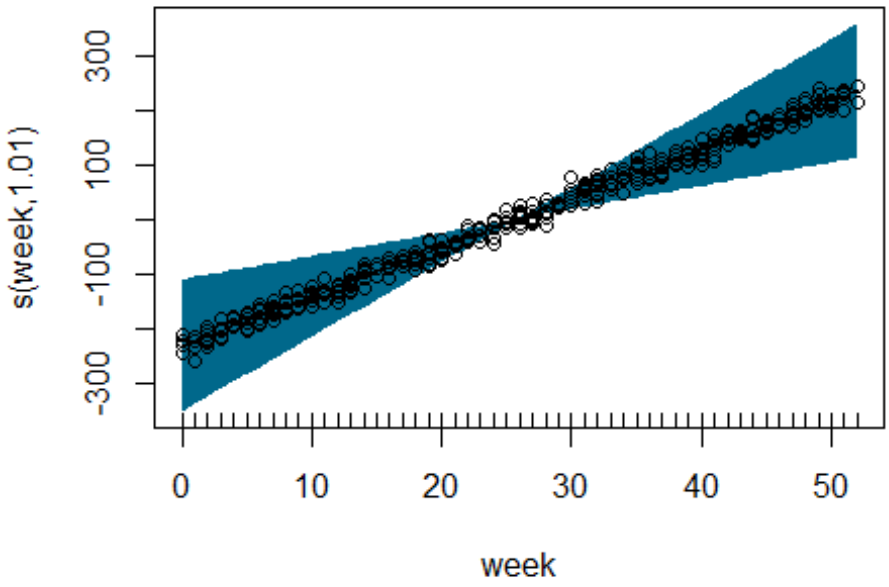
```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 144.6658     0.6733   214.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##           edf Ref.df      F  p-value
## s(day)  4.968  6.059  7.411 1.52e-07 ***
## s(week) 1.009  1.012 14.448 0.000162 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.119   Deviance explained = 13.3%
## -REML = 1448.3  Scale est. = 165.48    n = 365
```

```
coef(gam_rockingham2)
```

```
##   (Intercept)      s(day).1      s(day).2      s(day).3      s(day).
4
##  1.446658e+02 -7.771295e+00  1.006512e+01  5.225508e+00  3.344374e+0
0
##      s(day).5      s(day).6      s(day).7      s(day).8      s(day).
9
##  1.803780e+00  3.962863e+00 -1.323986e+00  4.778642e+00 -1.360786e+0
2
##     s(week).1     s(week).2     s(week).3     s(week).4     s(week).
5
##  1.685940e-02  3.893194e-02  7.834790e-03 -1.863760e-02  7.728485e-0
3
##     s(week).6     s(week).7     s(week).8     s(week).9
##  1.914111e-02 -6.011626e-03  5.516104e-02  1.347239e+02
```

```
plot(gam_rockingham2,rug=TRUE,residuals = TRUE, pch=1, cex=1, shade=TRU
E, shade.col="deepskyblue4")
```





The plots above are of the multivariate GAM. The summary of the second game model also shows it explains 13.3% of the variation this is almost 4% more than the first model. Comparing the residuals in the Q-Q plot of the 1st and 2nd model, I think the 2nd

model looks slightly better. More of the residuals are closer to the line especially at the bottom and top of the line. The pattern is a little hard to see but noticeable.

```
AIC(gam_rockingham2)

## [1] 2911.421

AIC(gam_rockingham)

## [1] 2924.875

AIC(lm_rockingham)

## [1] 2950.139
```

The AIC scores show the best model is the multivariate GAM using weeks and days. 2nd best is the single variable GAM using just days. I think the AIC combined with the analysis of the residuals and the fact the second GAM explains more of the variation. That the 2nd GAM which includes weeks and the day is the better model compared to the LM and single variable GAM.

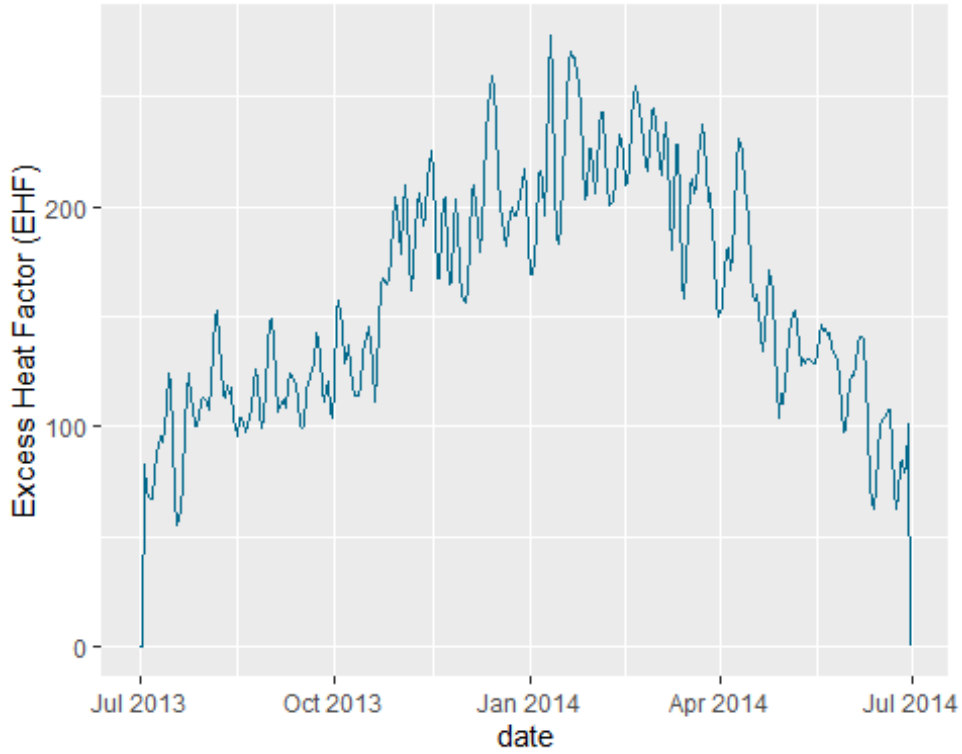
```
sapply(rockingham, class)

##       Date Admissions Attendance      Tri_1      Tri_2      Tri_3
   Tri_4
##     "Date"  "numeric"  "numeric"  "numeric"  "numeric"  "numeric"  "
numeric"
##      Tri_5        day       week
##  "numeric"  "numeric"  "numeric"
```

The day of the week variables are all numeric. If the variables were changed from numeric to another type of data this would make fitting a model to the data very difficult. The numeric values can be computed and scaled. If it was changed to categorical or ordinal it would be hard to do a linear regression time series model and it would change the model.

```
options(noaakey = "pSWkTBgIBNbEgIsfxzIjFOMsAOANVGxN")
perth_avg <- ncdc(datasetid = 'GHCND', datatypeid = 'TAVG', stationid =
'GHCND:ASN00009021', startdate = '2013-07-01', enddate = '2014-06-30',
limit = 1000
)$data

library(RcppRoll)
ehf_data <- perth_avg %>%
  dplyr::mutate(after=roll_meanr(lead(value),3)) %>%
  dplyr::mutate(ehf = after-30)

ehf_data <- mutate_at(ehf_data, c("ehf"), ~replace(., is.na(.), 0))
```

I replaced the NA ehf values with 0, I'm not sure if this was the correct thing to do or not. I had lots of trouble creating the ehf_data and the ehf column inside the ehf_data. I hope it is running correctly.

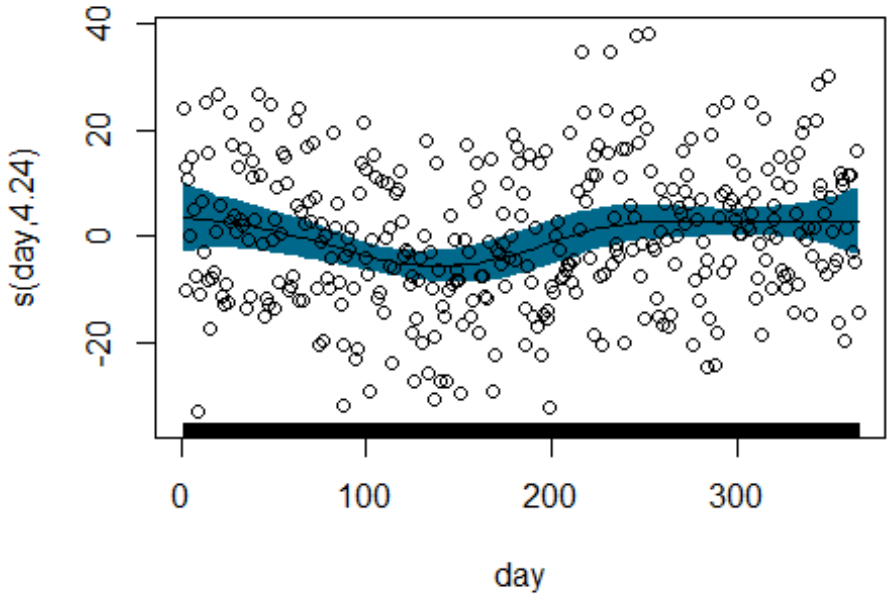
```
perth_avg %>%
  mutate(date = ymd_hms(ehf_data$date)) %>%
  ggplot(aes(x=date, y=ehf_data$ehf)) +
  geom_line(color='deepskyblue4') +
  ylab("Excess Heat Factor (EHF)")
```

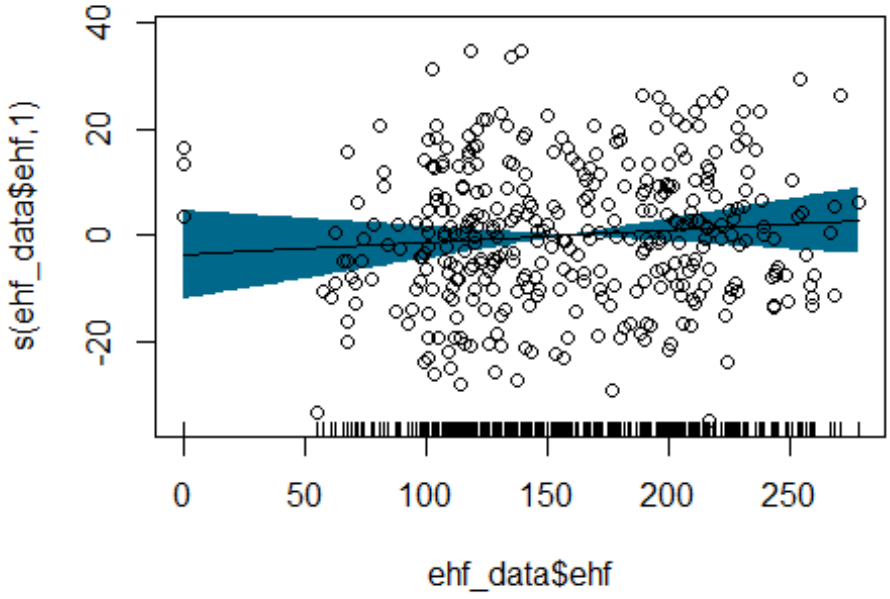


The above plot shows the extreme heat factor was very high from December to March.

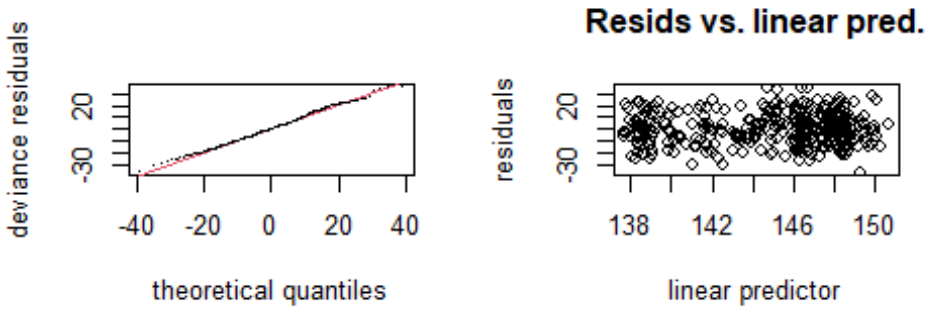
```
gam_ehf <- gam(Attendance ~ s(day) + s(ehf_data$ehf), data=rockingham,
method="REML")

glance(gam_ehf)

## # A tibble: 1 x 7
##      df logLik   AIC   BIC deviance df.residual  nobs
##   <dbl>  <dbl> <dbl> <dbl>    <dbl>       <dbl> <int>
## 1  6.24 -1455. 2927. 2959.   61985.        359.   365

summary(gam_ehf)

##
## Family: gaussian
```

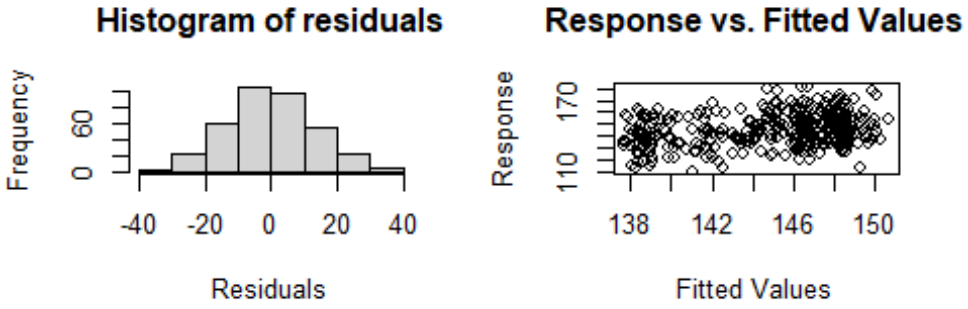
```
## Link function: identity
##
## Formula:
## Attendance ~ s(day) + s(ehf_data$ehf)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  144.666      0.688   210.3   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                   edf Ref.df     F p-value
## s(day)          4.239  5.290 4.010 0.00126 **
## s(ehf_data$ehf) 1.002  1.003 0.769 0.38103
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0797   Deviance explained = 9.29%
## -REML = 1458.3  Scale est. = 172.78    n = 365

plot(gam_ehf, rug=TRUE, residuals = TRUE, pch=1, cex=1, shade=TRUE, sha
de.col="deepskyblue4")
```






```
gam.check(gam_ehf)
```






```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.0004872719,0.0002667599]
## (score 1458.336 & scale 172.7763).
## Hessian positive definite, eigenvalue range [0.0004866886,181.0143].
## Model rank =  19 / 19
##
## 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(day)          9.00 4.24    0.82  <2e-16 ***
## s(ehf_data$ehf) 9.00 1.00    0.97    0.23
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I think it makes the model somewhat worse that the 2nd GAM but better than the 1st GAM and the LM. The explained deviance is lower and the residual plots look more scattered and worse than the 2nd GAM. So, it makes it somewhat better than the 1st two models. However the p-value for the EHF variable as shown above is 0.26 which would indicate it doesn't have an affect or if it does it's very small.

```
gam_with_extra <- gam(Attendance ~s(day)+s(ehf_data$ehf)+s(ehf_data$val
ue), data=rockingham, method="REML")
```

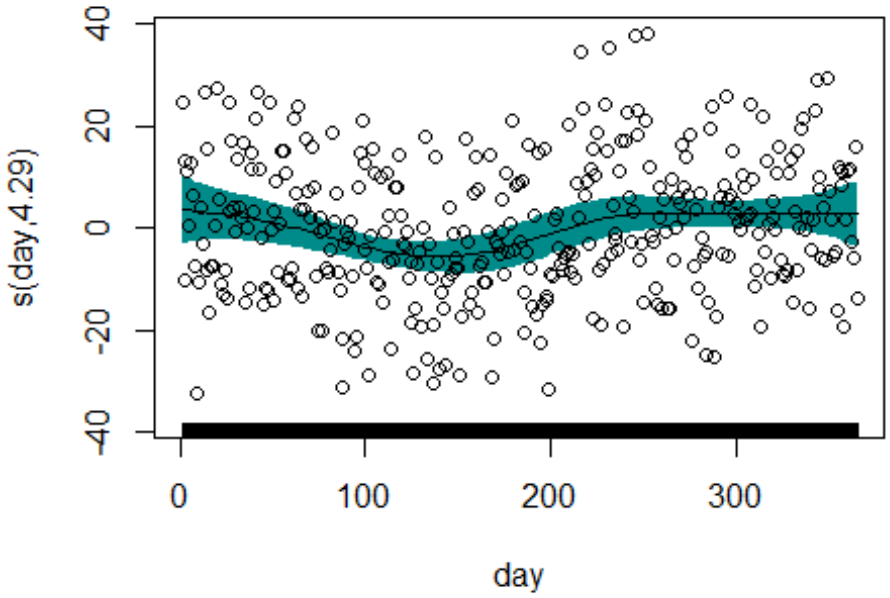
```r
glance(gam_with_extra)
```

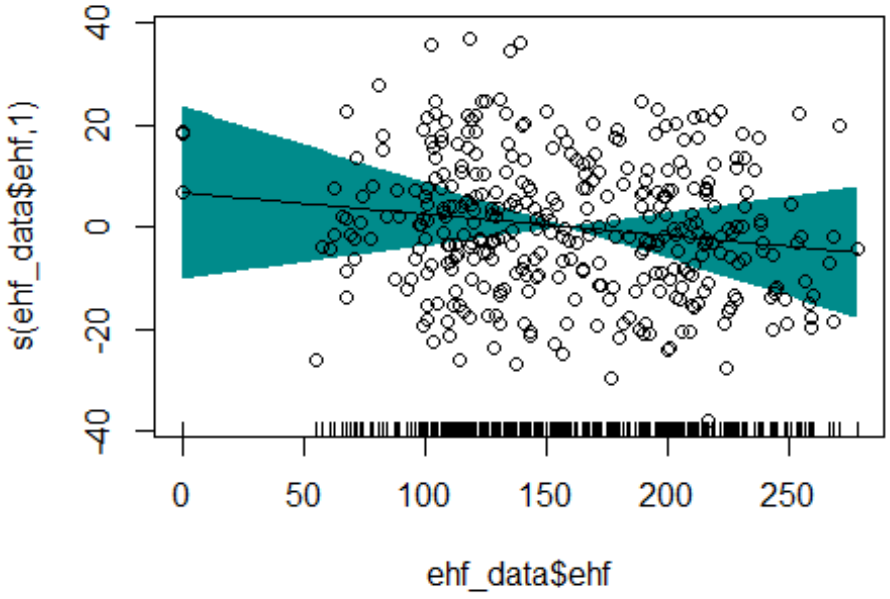
```
## # A tibble: 1 x 7
##      df logLik   AIC   BIC deviance df.residual  nobs
##   <dbl>  <dbl> <dbl> <dbl>    <dbl>       <dbl> <int>
## 1  7.46 -1454. 2927. 2964.   61548.        358.   365
```

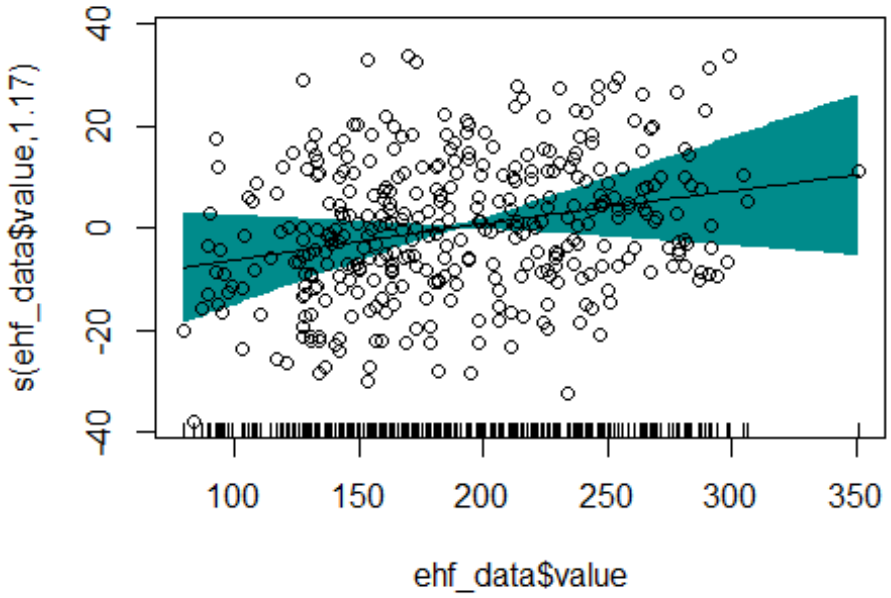
```r
summary(gam_with_extra)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Attendance ~ s(day) + s(ehf_data$ehf) + s(ehf_data$value)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 144.6658     0.6868   210.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                     edf Ref.df     F p-value
## s(day)            4.288  5.339 4.063 0.00107 **
## s(ehf_data$ehf)   1.001  1.003 0.615 0.43352
## s(ehf_data$value) 1.171  1.316 1.243 0.21691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.083   Deviance explained = 9.93%
## -REML = 1455.5  Scale est. = 172.14    n = 365
```

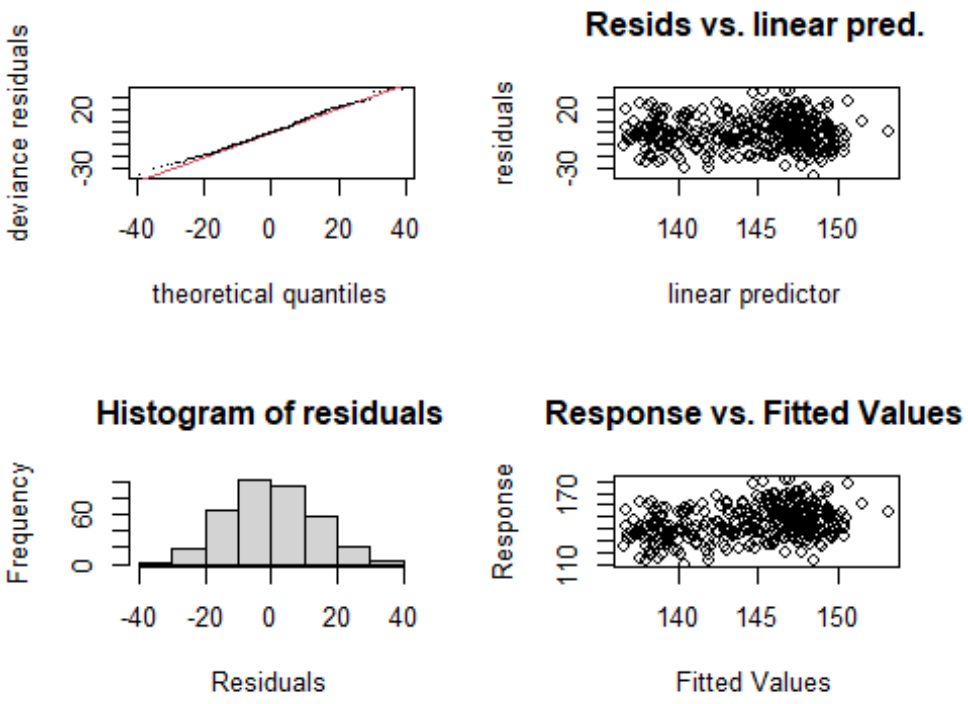
```r
plot(gam_with_extra, rug=TRUE, residuals = TRUE, pch=1, cex=1, shade=TRUE, shade.col="cyan4")
```


s(day,4.29)
40
20
0
-20
-40
0
100
200
300
day


s(ehf_data$ehf,1)
40
20
0
-20
-40
0
50
100
150
200
250
ehf_data$ehf




```
gam.check(gam_with_extra)
```



```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.0005121112,0.0004762333]
## (score 1455.48 & scale 172.1445).
## Hessian positive definite, eigenvalue range [0.0005117501,180.5146].
## Model rank =  28 / 28
##
## 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(day)             9.00 4.29    0.82  <2e-16 ***
## s(ehf_data$ehf)    9.00 1.00    0.97    0.26
## s(ehf_data$value) 9.00 1.17    1.08    0.91
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I wasn't able to find data sufficient additional data to include and I wasn't sure what data to include. So to include an additional extreme weather feature, I have used the average daily temperature value as a variable in the GAM. As you can see above the P-value of this additional variable is almost 1. So in this model it has almost no impact on hospital attendance at all. I don't think the introduction of this variable has really improved the model at all.

There are a number of extreme weather events that could impact hospital demand. Floods, hurricanes, storms, extreme cold, and heat. Exploring these in a model would be worthwhile.

We used historical hospital attendance data to create our regression models. In using this historical data, we must also believe that the historical attendance data has some predictive power to predict future hospital attendance numbers. Using historical data has limitations. The older historical data is, the more inaccurate and outdated it usually is. Making long predictions into the future requires more and more historical data to be accurate. Historical data doesn't always show an accurate underlying trend. Extreme events like hurricanes, floods, extreme heat waves can distort a normal trend. In this analysis, using the hospital attendance data there was also also a heat wave during the same period this could have distorted the trend and made it difficult to use a regression model to make future predictions (Connector, 2021).

In this analysis we are exploring Western Australian hospital demand and its connection to weather events, most notably heat waves. In this analysis I don't think we need to focus the regression model on understanding a process or making future predictions. I think it's possible to do both. but I think our regression model is better used for understanding a process. Namely, understanding what happens to a hospital during an extreme heatwave or other weather event. For this analysis we only use one year of data. This isn't enough to make very accurate long-term

predictions. As explained above our prediction ability is limited by having only one year of data, also during this year there was an extreme weather event. The year is perhaps not typical of other years and it would be improper to use this model for long term prediction. The resulting LM would be a poor at predicting. It would be better to use the model to try and explain and understand what the health system, WA government, and the public can do during an extreme weather event and attending a hospital. If we wanted to use the model for long-term prediction, we would need to collect more data and remove some of the extreme outliers such as the weather events to predict future hospital attendance accurately (IBM, 2021).

I think my analyses have answered or somewhat answered the questions that I set out to answer. It would be easier to answer these questions if we had many more years' worth of data especially if we had data on previous extreme weather events and their impact on hospital attendance. The models made in this analysis have shown that extreme weather events do have an impact on hospital attendance, but it doesn't explain all the variation. With higher temperatures you are likely to see more people attending a hospital. There are numerous articles online detailing this effect. (https://www.abc.net.au/news/2019-12-02/perth-has-gone-three-summers-without-heatwave-but-its-warming-up/11753340) This article shows that extreme heat deaths in Australia account for 55% of all-natural hazard deaths. The models created for this analysis can be used by the WA government, and health service to develop preventative measures to mitigate the increase in hospital attendance due to extreme heat. This could include education programs, government alerts and text messages during hot periods. The public can use this information to be better informed about the impact of heat waves, that heat waves cause more deaths and hospital attendance than all other natural hazard events.

Connector. 2021. Limitations of historical data. [online] Available at: https://support.onesaas.com/hc/en-us/articles/204756914-Limitations-of-historical-data [Accessed 17 September 2021].

Ibm.com. 2021. About Linear Regression | IBM. [online] Available at: https://www.ibm.com/topics/linear-regression [Accessed 20 September 2021].