STAT 414 - todo for part 3 submittion

Kyle Nessen, Justin Mai, Aiden Kelly, Arneh Begi

2024-10-28

# Load necessary packages  
library(lme4)

Loading required package: Matrix

library(ggplot2)  
library(dplyr)

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':  
  
 filter, lag

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

library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ lubridate 1.9.3 ✔ tibble 3.2.1  
✔ purrr 1.0.2 ✔ tidyr 1.3.1  
✔ readr 2.1.5

── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
✖ tidyr::expand() masks Matrix::expand()  
✖ dplyr::filter() masks stats::filter()  
✖ dplyr::lag() masks stats::lag()  
✖ tidyr::pack() masks Matrix::pack()  
✖ tidyr::unpack() masks Matrix::unpack()  
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggeffects)  
  
#Seeing grove as a random effect  
df <- suppressMessages(read\_csv('allgr\_array\_KianaRawdat.csv'))  
  
# Update 'butterfly\_present' column in R  
df$butterfly\_present <- ifelse(df$array == "Cluster", 1, 0)  
  
head(df)

# A tibble: 6 × 22  
 ...1 array month.day temp.avg hum.avg dew.pt.avg light.avg light.min  
 <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
1 1 Cluster 01-01 11.7 46.6 -1.45 26005. 0  
2 2 Cluster 01-02 9.36 57.2 -0.605 25292. 0  
3 3 Cluster 01-03 10.4 62.5 1.62 25430. 0  
4 4 Cluster 01-04 10.6 71.1 3.56 22243. 0  
5 5 Cluster 01-05 9.77 88.7 7.71 766. 0  
6 6 Cluster 01-06 12.4 94.5 10.9 12197. 0  
# ℹ 14 more variables: light.max <dbl>, temp.min <dbl>, temp.max <dbl>,  
# temp.std <dbl>, light.std <dbl>, dew.pt.std <dbl>, hum.std <dbl>,  
# hum.max <dbl>, hum.min <dbl>, dp.max <dbl>, dp.min <dbl>, daynum <dbl>,  
# grove <chr>, butterfly\_present <dbl>

seasonal\_summary <- df %>%  
 group\_by(grove, array, butterfly\_present) %>%  
 summarise(  
 # Temperature metrics  
 temp\_mean = mean(temp.avg, na.rm = TRUE),  
 temp\_range = max(temp.max, na.rm = TRUE) - min(temp.min, na.rm = TRUE),  
 temp\_std = mean(temp.std, na.rm = TRUE),  
 diurnal\_range = mean(temp.max - temp.min, na.rm = TRUE),  
 isothermality = (mean(temp.max - temp.min, na.rm = TRUE) /   
 (max(temp.max, na.rm = TRUE) - min(temp.min, na.rm = TRUE))) \* 100,  
   
 # Temperature stability metrics  
 temp\_stability = 1 / mean(temp.std, na.rm = TRUE), # Higher values = more stable temperatures  
 temp\_extremity = abs(mean(temp.avg, na.rm = TRUE) - median(temp.avg, na.rm = TRUE)), # Distance from typical conditions  
   
 # Humidity metrics  
 hum\_mean = mean(hum.avg, na.rm = TRUE),  
 hum\_range = max(hum.max, na.rm = TRUE) - min(hum.min, na.rm = TRUE),  
 hum\_std = mean(hum.std, na.rm = TRUE),  
   
 # Dew point metrics  
 dewpt\_mean = mean(dew.pt.avg, na.rm = TRUE),  
 dewpt\_range = max(dp.max, na.rm = TRUE) - min(dp.min, na.rm = TRUE),  
   
 # Light metrics  
 light\_mean = mean(light.avg, na.rm = TRUE),  
 light\_range = mean(light.max - light.min, na.rm = TRUE),  
 light\_std = mean(light.std, na.rm = TRUE),  
   
 # Vapor Pressure Deficit (VPD) approximation  
 # Using Magnus formula for saturated vapor pressure  
 sat\_vp = 0.61078 \* exp((17.27 \* temp\_mean) / (temp\_mean + 237.3)),  
 actual\_vp = sat\_vp \* (hum\_mean / 100),  
 vpd = sat\_vp - actual\_vp,  
   
 # Count of extreme days  
 cold\_days = sum(temp.min < 0, na.rm = TRUE), # Days below freezing  
 humid\_days = sum(hum.max > 90, na.rm = TRUE), # Very humid days  
   
 # Sample size  
 n\_observations = n()  
 ) %>%  
 ungroup()

`summarise()` has grouped output by 'grove', 'array'. You can override using  
the `.groups` argument.

# Check the results  
glimpse(seasonal\_summary)

Rows: 45  
Columns: 24  
$ grove <chr> "AB", "AB", "AB", "AB", "AB", "BL", "BL", "BL", "BL"…  
$ array <chr> "Cluster", "NE", "NW", "SE", "SW", "Cluster", "NE", …  
$ butterfly\_present <dbl> 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0…  
$ temp\_mean <dbl> 14.30882, 12.96351, 12.84848, 14.29949, 12.82595, 11…  
$ temp\_range <dbl> 39.353, 24.236, 28.398, 41.527, 34.857, 27.341, 36.8…  
$ temp\_std <dbl> 5.602221, 2.883554, 3.086711, 6.223597, 4.079612, 3.…  
$ diurnal\_range <dbl> 21.502544, 10.468926, 12.182324, 22.173375, 17.78767…  
$ isothermality <dbl> 54.64016, 43.19577, 42.89853, 53.39508, 51.03043, 49…  
$ temp\_stability <dbl> 0.1785006, 0.3467942, 0.3239694, 0.1606788, 0.245121…  
$ temp\_extremity <dbl> 0.006028761, 0.209001774, 0.052235174, 0.108946128, …  
$ hum\_mean <dbl> 66.77266, 68.43785, 71.14692, 66.72285, 67.98397, 88…  
$ hum\_range <dbl> 97.0, 89.5, 92.5, 99.0, 90.5, 79.5, 78.0, 79.5, 84.0…  
$ hum\_std <dbl> 13.245999, 10.939452, 10.315429, 15.224467, 13.20570…  
$ dewpt\_mean <dbl> 6.112396, 5.836885, 6.189558, 6.096470, 6.484920, 8.…  
$ dewpt\_range <dbl> 39.1, 31.5, 35.7, 39.8, 27.8, 21.8, 23.8, 24.0, 25.6…  
$ light\_mean <dbl> 11650.050, 2112.700, 1545.939, 19236.307, 7781.737, …  
$ light\_range <dbl> 128347.53, 43631.79, 18227.57, 158266.86, 134098.05,…  
$ light\_std <dbl> 28100.567, 4868.427, 2793.732, 41786.625, 21269.673,…  
$ sat\_vp <dbl> 1.630872, 1.494150, 1.482942, 1.629887, 1.480755, 1.…  
$ actual\_vp <dbl> 1.0889764, 1.0225644, 1.0550675, 1.0875071, 1.006676…  
$ vpd <dbl> 0.5418952, 0.4715860, 0.4278744, 0.5423800, 0.474079…  
$ cold\_days <int> 0, 0, 0, 0, 0, 5, 5, 6, 6, 6, 0, 0, 0, 0, 0, 0, 0, 0…  
$ humid\_days <int> 33, 34, 39, 40, 24, 27, 8, 33, 22, 22, 28, 12, 14, 3…  
$ n\_observations <int> 68, 68, 68, 72, 68, 42, 41, 42, 42, 41, 65, 72, 70, …

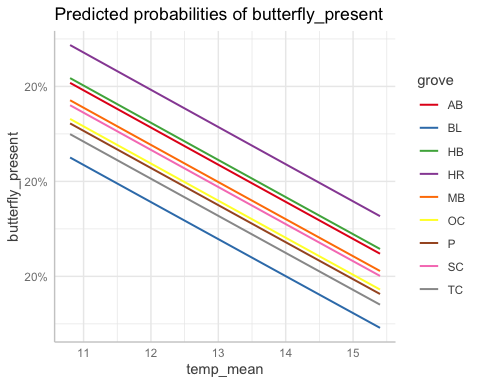
logistic\_model <- glm(butterfly\_present ~ grove + temp\_mean, data = seasonal\_summary, family = binomial)  
summary(logistic\_model)

Call:  
glm(formula = butterfly\_present ~ grove + temp\_mean, family = binomial,   
 data = seasonal\_summary)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)  
(Intercept) -1.3534392 8.4052542 -0.161 0.872  
groveBL -0.0049007 2.0109330 -0.002 0.998  
groveHB 0.0003127 1.5831345 0.000 1.000  
groveHR 0.0024766 1.7012948 0.001 0.999  
groveMB -0.0011403 1.6073259 -0.001 0.999  
groveOC -0.0023656 1.6910780 -0.001 0.999  
groveP -0.0026604 1.7190054 -0.002 0.999  
groveSC -0.0014668 1.6242951 -0.001 0.999  
groveTC -0.0033655 1.7967859 -0.002 0.999  
temp\_mean -0.0024430 0.6194403 -0.004 0.997  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 45.036 on 44 degrees of freedom  
Residual deviance: 45.036 on 35 degrees of freedom  
AIC: 65.036  
  
Number of Fisher Scoring iterations: 4

plot(ggpredict(logistic\_model, c("temp\_mean","grove"), type="re"),show\_ci=FALSE)

Warning: Some of the focal terms are of type `character`. This may lead to  
 unexpected results. It is recommended to convert these variables to  
 factors before fitting the model.  
 The following variables are of type character: `grove`

Data were 'prettified'. Consider using `terms="temp\_mean [all]"` to get  
 smooth plots.



#ranef(logistic\_model)

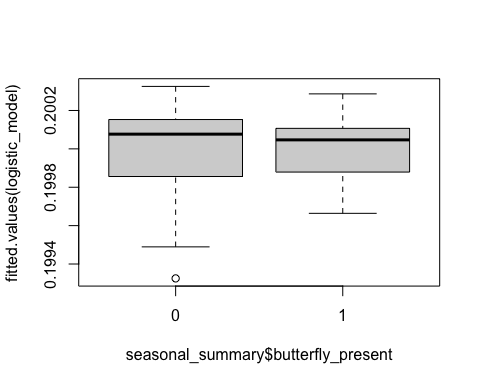
head(fitted.values(logistic\_model))

1 2 3 4 5 6   
0.1996641 0.2001898 0.2002348 0.1996677 0.2002436 0.2000469

pred <- fitted.values(logistic\_model) > .2  
table(pred,seasonal\_summary$butterfly\_present)

pred 0 1  
 FALSE 12 4  
 TRUE 24 5

boxplot(fitted.values(logistic\_model)~seasonal\_summary$butterfly\_present)

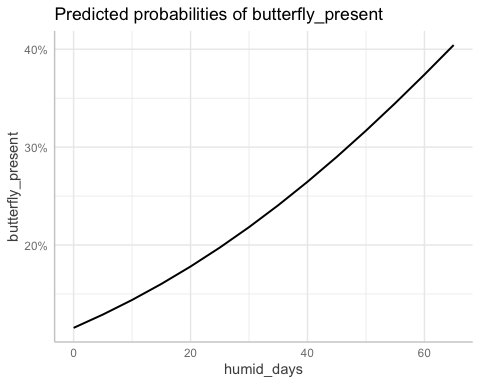


model1 <- glm(butterfly\_present ~ temp\_mean + humid\_days + hum\_mean, data = seasonal\_summary, family = binomial)  
summary(model1)

Call:  
glm(formula = butterfly\_present ~ temp\_mean + humid\_days + hum\_mean,   
 family = binomial, data = seasonal\_summary)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)  
(Intercept) -8.32704 12.45113 -0.669 0.504  
temp\_mean 0.26870 0.54897 0.489 0.625  
humid\_days 0.02538 0.02934 0.865 0.387  
hum\_mean 0.03904 0.08909 0.438 0.661  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 45.036 on 44 degrees of freedom  
Residual deviance: 43.650 on 41 degrees of freedom  
AIC: 51.65  
  
Number of Fisher Scoring iterations: 4

plot(ggpredict(model1, c("humid\_days"), type="re"),show\_ci=FALSE)

Data were 'prettified'. Consider using `terms="humid\_days [all]"` to get  
 smooth plots.



#ranef(logistic\_model)

Focus on differentials (like group mean centering) How does cluster site compare to other stations focus on variables that offer the most variability

model0 <- glm(butterfly\_present ~ 1, data = seasonal\_summary, family = binomial)  
summary(model0)

Call:  
glm(formula = butterfly\_present ~ 1, family = binomial, data = seasonal\_summary)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -1.3863 0.3727 -3.72 0.000199 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 45.036 on 44 degrees of freedom  
Residual deviance: 45.036 on 44 degrees of freedom  
AIC: 47.036  
  
Number of Fisher Scoring iterations: 4

model1 <- glm(butterfly\_present ~ hum\_std, data = seasonal\_summary, family = binomial)  
summary(model1)

Call:  
glm(formula = butterfly\_present ~ hum\_std, family = binomial,   
 data = seasonal\_summary)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)  
(Intercept) 0.1975 2.2180 0.089 0.929  
hum\_std -0.1469 0.2059 -0.714 0.475  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 45.036 on 44 degrees of freedom  
Residual deviance: 44.499 on 43 degrees of freedom  
AIC: 48.499  
  
Number of Fisher Scoring iterations: 4

anova(model0,model1)

Analysis of Deviance Table  
  
Model 1: butterfly\_present ~ 1  
Model 2: butterfly\_present ~ hum\_std  
 Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
1 44 45.036   
2 43 44.499 1 0.5372 0.4636

# Null model  
model0 <- glm(butterfly\_present ~ 1, data = seasonal\_summary, family = binomial)  
  
# Temperature-based models  
model\_temp1 <- glm(butterfly\_present ~ temp\_mean, data = seasonal\_summary, family = binomial)  
model\_temp2 <- glm(butterfly\_present ~ temp\_range, data = seasonal\_summary, family = binomial)  
model\_temp3 <- glm(butterfly\_present ~ temp\_stability, data = seasonal\_summary, family = binomial)  
model\_temp4 <- glm(butterfly\_present ~ temp\_mean + temp\_range, data = seasonal\_summary, family = binomial)  
model\_temp5 <- glm(butterfly\_present ~ isothermality, data = seasonal\_summary, family = binomial)  
model\_temp6 <- glm(butterfly\_present ~ temp\_mean + temp\_stability, data = seasonal\_summary, family = binomial)  
model\_temp7 <- glm(butterfly\_present ~ diurnal\_range + cold\_days, data = seasonal\_summary, family = binomial)  
  
# Humidity-based models  
model\_hum1 <- glm(butterfly\_present ~ hum\_mean, data = seasonal\_summary, family = binomial)  
model\_hum2 <- glm(butterfly\_present ~ hum\_std, data = seasonal\_summary, family = binomial)  
model\_hum3 <- glm(butterfly\_present ~ hum\_range, data = seasonal\_summary, family = binomial)  
model\_hum4 <- glm(butterfly\_present ~ hum\_mean + hum\_std, data = seasonal\_summary, family = binomial)  
model\_hum5 <- glm(butterfly\_present ~ humid\_days, data = seasonal\_summary, family = binomial)  
  
# Combined temperature and humidity models  
model\_th1 <- glm(butterfly\_present ~ temp\_mean + hum\_mean, data = seasonal\_summary, family = binomial)  
model\_th2 <- glm(butterfly\_present ~ temp\_stability + hum\_std, data = seasonal\_summary, family = binomial)  
model\_th3 <- glm(butterfly\_present ~ temp\_mean + hum\_mean + temp\_range, data = seasonal\_summary, family = binomial)  
model\_th4 <- glm(butterfly\_present ~ vpd, data = seasonal\_summary, family = binomial)  
model\_th5 <- glm(butterfly\_present ~ temp\_stability + humid\_days, data = seasonal\_summary, family = binomial)  
  
# Light-based models  
model\_light1 <- glm(butterfly\_present ~ light\_mean, data = seasonal\_summary, family = binomial)  
model\_light2 <- glm(butterfly\_present ~ light\_std, data = seasonal\_summary, family = binomial)  
model\_light3 <- glm(butterfly\_present ~ light\_range, data = seasonal\_summary, family = binomial)  
  
# Complex multivariate models  
model\_complex1 <- glm(butterfly\_present ~ temp\_mean + hum\_mean + light\_mean, data = seasonal\_summary, family = binomial)  
model\_complex2 <- glm(butterfly\_present ~ temp\_stability + hum\_std + light\_std, data = seasonal\_summary, family = binomial)  
model\_complex3 <- glm(butterfly\_present ~ temp\_mean + vpd + light\_mean, data = seasonal\_summary, family = binomial)  
model\_complex4 <- glm(butterfly\_present ~ isothermality + hum\_range + light\_range, data = seasonal\_summary, family = binomial)  
  
# Compare models using AIC  
library(AICcmodavg)

Attaching package: 'AICcmodavg'

The following object is masked from 'package:lme4':  
  
 checkConv

# Create a list of models  
models <- list(model0, model\_temp1, model\_temp2, model\_temp3, model\_temp4, model\_temp5,  
 model\_temp6, model\_temp7, model\_hum1, model\_hum2, model\_hum3, model\_hum4,  
 model\_hum5, model\_th1, model\_th2, model\_th3, model\_th4, model\_th5,  
 model\_light1, model\_light2, model\_light3, model\_complex1, model\_complex2,  
 model\_complex3, model\_complex4)  
  
# Create model names  
model\_names <- c("null", "temp1", "temp2", "temp3", "temp4", "temp5", "temp6", "temp7",  
 "hum1", "hum2", "hum3", "hum4", "hum5", "th1", "th2", "th3", "th4", "th5",  
 "light1", "light2", "light3", "complex1", "complex2", "complex3", "complex4")  
  
# Compare models  
aictab(cand.set = models, modnames = model\_names)

Model selection based on AICc:  
  
 K AICc Delta\_AICc AICcWt Cum.Wt LL  
null 1 47.13 0.00 0.15 0.15 -22.52  
hum5 2 48.18 1.05 0.09 0.23 -21.95  
hum2 2 48.78 1.66 0.06 0.30 -22.25  
temp5 2 49.01 1.88 0.06 0.35 -22.36  
hum1 2 49.02 1.89 0.06 0.41 -22.37  
hum3 2 49.05 1.92 0.06 0.47 -22.38  
th4 2 49.14 2.01 0.05 0.52 -22.43  
light3 2 49.28 2.15 0.05 0.57 -22.50  
light1 2 49.30 2.17 0.05 0.62 -22.51  
temp3 2 49.31 2.18 0.05 0.67 -22.51  
light2 2 49.32 2.19 0.05 0.72 -22.52  
temp2 2 49.32 2.19 0.05 0.77 -22.52  
temp1 2 49.32 2.19 0.05 0.82 -22.52  
th5 3 50.45 3.33 0.03 0.85 -21.93  
th2 3 50.76 3.63 0.02 0.87 -22.09  
th1 3 50.97 3.84 0.02 0.89 -22.19  
hum4 3 51.03 3.90 0.02 0.91 -22.22  
temp7 3 51.48 4.35 0.02 0.93 -22.45  
temp6 3 51.61 4.48 0.02 0.94 -22.51  
temp4 3 51.62 4.49 0.02 0.96 -22.52  
complex2 4 52.24 5.11 0.01 0.97 -21.62  
complex4 4 52.74 5.61 0.01 0.98 -21.87  
complex1 4 53.21 6.08 0.01 0.99 -22.10  
complex3 4 53.33 6.20 0.01 0.99 -22.16  
th3 4 53.34 6.21 0.01 1.00 -22.17

calculate\_relative\_differences <- function(seasonal\_summary) {  
 # Get names of numeric columns (excluding 'grove' if it exists)  
 numeric\_cols <- names(df)[sapply(df, is.numeric)]  
 numeric\_cols <- numeric\_cols[numeric\_cols != "grove"]  
   
 # Calculate relative differences for each numeric column  
 df\_with\_diffs <- df %>%  
 group\_by(grove) %>%  
 mutate(across(  
 all\_of(numeric\_cols),  
 ~ . - mean(., na.rm = TRUE),  
 .names = "{col}\_rel\_diff"  
 )) %>%  
 ungroup()  
   
 return(df\_with\_diffs)  
}