



Migratory protogyny and condition-dependent arrival in Icelandic Red-necked Phalaropes

Oscar G. Miranda, Pedro Rodrigues, Tamás Székely, Renáta Szarvas & José O. Valdebenito

27 June 2025

Contents

1	Set working directory	1
2	Libraries	2
3	Spring arrival time	2
3.1	Load census data	2
3.2	Binomial generalized linear model	2
4	Figure 1 a	5
5	Leukocyte profiles	7
5.1	Load leukocyte data	7
5.2	Data wrangling	8
5.3	Mean and standard deviation per sex	8
6	Morphometry	10
6.1	Loading and wrangling data	10
6.2	Mean and standard deviation per sex	10
7	Body condition across time	14
7.1	Linear model	14
8	Figure 1 b	16

1 Set working directory

```
setwd("C:/Documentos/University of Debrecen/Valdebenito/Phalarope")
```

2 Libraries

```
pacman::p_load(ggplot2, dplyr, cowplot, readxl, tidyr, broom, car, patchwork, lmtest, smatr, strucchange, lubridate)
```

3 Spring arrival time

3.1 Load census data

```
p<-read.csv(file = "Data/PhalaropeCensus-22-23.csv", header = T)

p[sapply(p,is.integer)]<-lapply(p[sapply(p,is.integer)],as.numeric)

p$year<-as.factor(p$year)

p$day<-as.Date(p$day,format="%Y-%m-%d")

p$day1<-as.character(p$day)

p$day1<-sub("2022-", "", p$day1)

p$day1<-sub("2023-", "", p$day1)

p$day1<-as.Date(p$day1,format="%m-%d")

p2022 <- subset(p, p$year == "2022")
```

3.2 Binomial generalized linear model

```
glm_fit <- glm(cbind(m, f) ~ as.numeric(day), family = binomial, data = p2022)
summary(glm_fit)
```

```
##
## Call:
## glm(formula = cbind(m, f) ~ as.numeric(day), family = binomial,
##      data = p2022)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -3.442e+03  7.150e+02  -4.814 1.48e-06 ***
## as.numeric(day)  1.797e-01  3.735e-02   4.813 1.49e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 35.2615  on 8  degrees of freedom
## Residual deviance:  6.5196  on 7  degrees of freedom
```

```
## AIC: 36.456
##
## Number of Fisher Scoring iterations: 4
```

To evaluate whether residuals showed temporal autocorrelation—which would violate the assumption of independence—we examined the autocorrelation function (ACF) of model residuals.

```
res <- residuals(glm_fit, type = "deviance")
acf_plot <- acf(res, lag.max = 10, plot = FALSE)

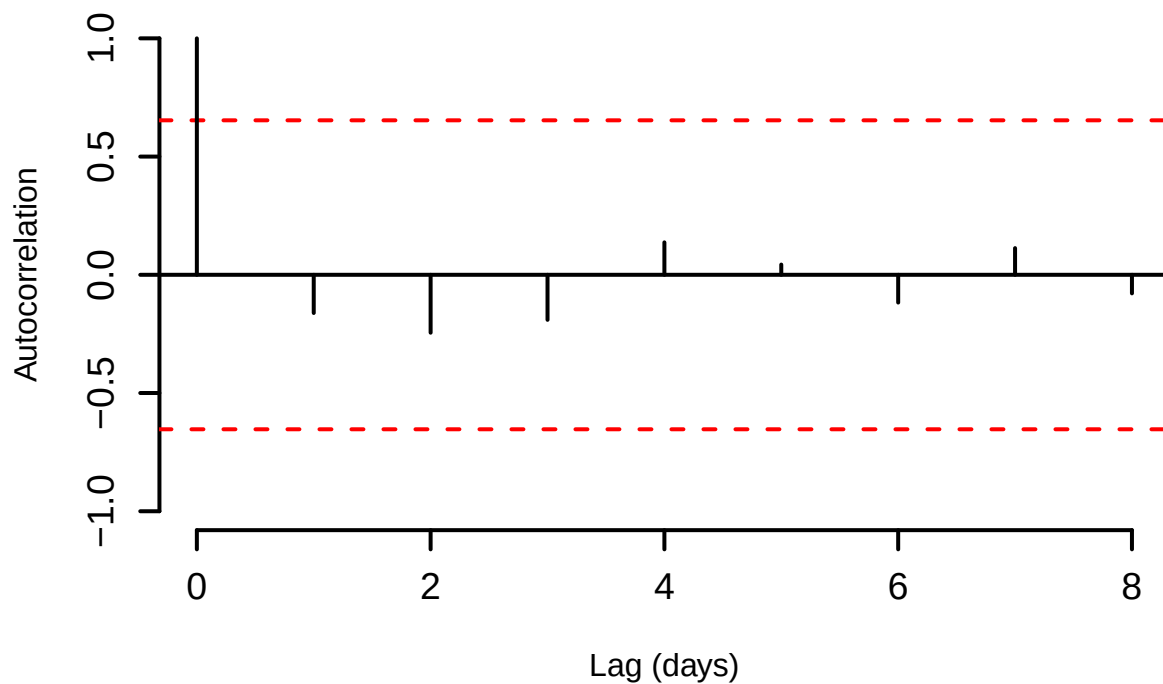
ci <- qnorm((1 + 0.95)/2) / sqrt(length(res))
ylim <- c(-1, 1)

plot(NA, xlim = range(acf_plot$lag), ylim = ylim,
     xlab = "Lag (days)", ylab = "Autocorrelation",
     main = "", axes = FALSE)

axis(1, lwd = 2, col = "black", cex.axis = 1.2)
axis(2, lwd = 2, col = "black", cex.axis = 1.2)

abline(h = 0, col = "black", lwd = 2)
abline(h = c(-ci, ci), col = "red", lty = 2, lwd = 2)

for (i in 1:length(acf_plot$acf)) {
  lines(x = rep(acf_plot$lag[i], 2),
        y = c(0, acf_plot$acf[i]),
        lwd = 2,
        col = "black")
}
```



Individuals of unknown sex were excluded from this analysis to avoid potential bias, as we could not assume their sex assignment was random or consistent across time. This approach allowed us to confidently assess trends among individuals of known sex only.

Nonetheless, the trend is the same, since the proportion of unknowns did not change significantly over time

```
glm_unk <- glm(cbind(unk, total_n.indv - unk) ~ as.numeric(day),
               family = binomial, data = p2022)
summary(glm_unk)
```

```
##
## Call:
## glm(formula = cbind(unk, total_n.indv - unk) ~ as.numeric(day),
##      family = binomial, data = p2022)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   10373.2759   6780.1042    1.53   0.126
## as.numeric(day)   -0.5422    0.3543   -1.53   0.126
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 8.7536  on 8  degrees of freedom
## Residual deviance: 1.6808  on 7  degrees of freedom
## AIC: 9.4971
##
## Number of Fisher Scoring iterations: 8
```

```
glm_fit <- glm(cbind(m, f + unk) ~ as.numeric(day), family = binomial, data = p2022)
summary(glm_fit)
```

```
##
## Call:
## glm(formula = cbind(m, f + unk) ~ as.numeric(day), family = binomial,
##      data = p2022)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -3501.7190    710.3871  -4.929 8.25e-07 ***
## as.numeric(day)    0.1829     0.0371   4.929 8.27e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 37.1842  on 8  degrees of freedom
## Residual deviance:  6.7652  on 7  degrees of freedom
## AIC: 36.717
##
## Number of Fisher Scoring iterations: 4
```

4 Figure 1 a

```
p2022 <- p[p$year == 2022, ]
p2022$day_year <- yday(p2022$day)

glm_fit <- glm(cbind(m, f) ~ day_year, family = binomial, data = p2022)

newdata <- data.frame(day_year = seq(min(p2022$day_year),
                                     max(p2022$day_year), length.out = 100))

pred <- predict(glm_fit, newdata = newdata, type = "link", se.fit = TRUE)

crit <- qnorm(0.975) # 95% CI
newdata$fit <- plogis(pred$fit)
newdata$lwr <- plogis(pred$fit - crit * pred$se.fit)
newdata$upr <- plogis(pred$fit + crit * pred$se.fit)

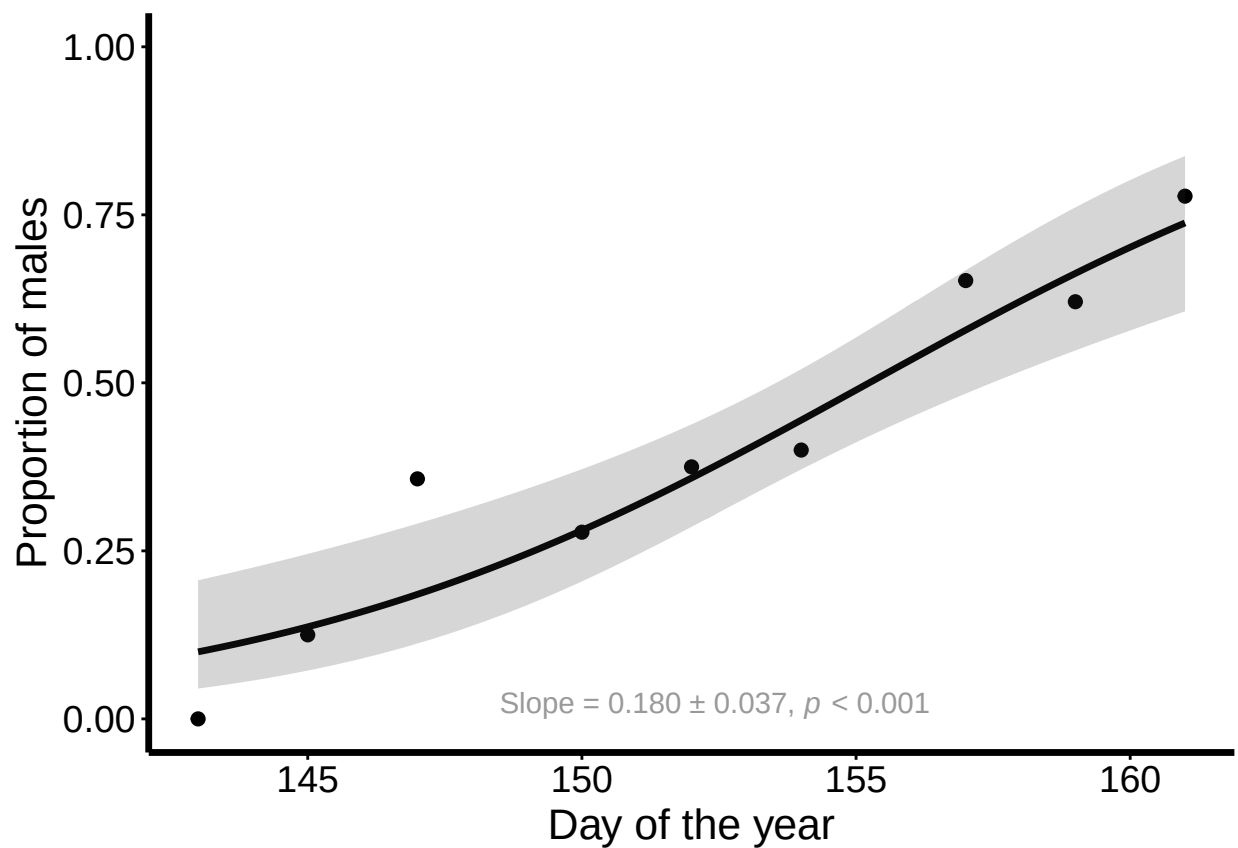
plot1 <- ggplot(p2022, aes(x = day_year, y = m / (m + f))) +
  geom_point(size = 2, color = "black") +
  geom_line(data = newdata, aes(x = day_year, y = fit), color = "black",
           size = 1.2, inherit.aes = FALSE) +
  geom_ribbon(data = newdata, aes(x = day_year, ymin = lwr, ymax = upr),
            alpha = 0.2, inherit.aes = FALSE) +
  annotate("text",
         x = 148.5, y = 0.02,
         label = expression("Slope = 0.180 ± 0.037, " * italic(p) * " < 0.001"),
         hjust = 0, size = 4, color = "grey60") +
```

```

ylab("Proportion of males") +
xlab("Day of the year") +
scale_y_continuous(
  limits = c(0, 1),
  breaks = seq(0, 1, 0.25),
  labels = c("0.00", "0.25", "0.50", "0.75", "1.00")
) +
theme(
  legend.position = "none",
  panel.background = element_rect(fill = NA),
  panel.grid = element_blank(),
  axis.line = element_line(size = 1.2, colour = "black"),
  axis.text = element_text(size = 14, color = "black"),
  axis.title = element_text(size = 16, color = "black"),
  axis.ticks = element_line(color = "black"),
  plot.title = element_text(face = "plain", size = 14)
)

```

plot1



5 Leukocyte profiles

5.1 Load leukocyte data

```
w <- read.csv("Data/wbc_phalarope_merged.csv") %>%
  mutate(across(where(is.character), as.factor),
         across(where(is.integer), as.numeric),
         across(c(1:5), as.factor))
w
```

##	ring	year	date	time	sex	weight	wing	tarsus	het	lim	eos	bas	mon	RBC1	RBC2
## 1	01	2021	2306	1405	M	30.2	106	20.1	12	5	0	0	0	140	200
## 2	02	2021	2306	2038	M	34.7	109	20.8	2	11	0	0	0	200	200
## 3	03	2021	2406	1904	M	37.8	105	20.6	NA	NA	NA	NA	NA	NA	NA
## 4	04	2021	2406	1945	F	44.5	115	20.5	NA	NA	NA	NA	NA	NA	NA
## 5	05	2021	2406	2046	F	39.3	118	21.9	0	10	0	0	0	200	200
## 6	06	2021	2506	1801	F	48.0	114	20.4	NA	NA	NA	NA	NA	NA	NA
## 7	07	2021	2506	1906	F	38.6	115	21.4	1	5	0	0	0	200	220
## 8	08	2022	606	2013	M	34.1	106	28.8	2	7	0	0	0	280	200
## 9	09	2022	706	2048	F	51.6	120	21.6	9	14	0	0	0	260	160
## 10	10	2022	806	1934	M	40.0	110	21.3	5	25	2	0	0	280	200
## 11	11	2022	906	1052	M	38.5	113	20.8	24	24	2	0	0	240	280
## 12	12	2022	906	1134	F	38.2	117	20.7	6	7	0	0	0	240	160
## 13	13	2022	906	1216	F	49.2	116	20.5	11	16	4	0	0	200	200
## 14	14	2023	106	1903	F	46.1	121	21.1	NA	NA	NA	NA	NA	NA	NA
## 15	15	2023	306	1110	F	42.0	121	21.9	NA	NA	NA	NA	NA	NA	NA
## 16	17	2023	1306	2035	M	39.5	114	21.5	NA	NA	NA	NA	NA	NA	NA
## 17	19	2023	2606	1255	M	32.8	115	21.7	NA	NA	NA	NA	NA	NA	NA
## 18	XX	2021	2306	1920	M	31.2	110	21.4	3	6	0	0	0	220	200
##	RBC3	Mean_RBC	Field	RBC_total	HET.10000	LIMF.10000	EOZ.10000	BAS.10000							
## 1	200	180.0000	43	7740.000	15.503876	6.459948	0.0000000							0	
## 2	200	200.0000	34	6800.000	2.941176	16.176471	0.0000000							0	
## 3	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 4	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 5	180	193.3333	33	6380.000	0.0000000	15.673981	0.0000000							0	
## 6	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 7	200	206.6667	34	7026.667	1.423150	7.115750	0.0000000							0	
## 8	220	233.3333	34	7933.333	2.521008	8.823529	0.0000000							0	
## 9	200	206.6667	56	11573.333	7.776498	12.096774	0.0000000							0	
## 10	160	213.3333	71	15146.667	3.301056	16.505282	1.3204225							0	
## 11	240	253.3333	159	40280.000	5.958292	5.958292	0.4965243							0	
## 12	160	186.6667	63	11760.000	5.102041	5.952381	0.0000000							0	
## 13	240	213.3333	70	14933.333	7.366071	10.714286	2.6785714							0	
## 14	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 15	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 16	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 17	NA	NA	NA	NA	NA	NA	NA	NA						NA	
## 18	200	206.6667	47	9713.333	3.088538	6.177076	0.0000000							0	
##	MON.10000	WBC_total	WBC.10000												
## 1	0	17	21.963824												
## 2	0	13	19.117647												
## 3	NA	NA	NA												

```
## 4      NA      NA      NA
## 5      0      10 15.673981
## 6      NA      NA      NA
## 7      0      6  8.538899
## 8      0      9 11.344538
## 9      0     23 19.873272
## 10     0     32 21.126761
## 11     0     50 12.413108
## 12     0     13 11.054422
## 13     0     31 20.758929
## 14     NA     NA     NA
## 15     NA     NA     NA
## 16     NA     NA     NA
## 17     NA     NA     NA
## 18     0      9  9.265614
```

5.2 Data wrangling

```
hemato_data<-w[, c("ring","sex","HET.10000","LIMF.10000","EOZ.10000","WBC.10000")]
hemato_data$Heterophils<-hemato_data$HET.10000/hemato_data$WBC.10000
hemato_data$Limphocytes<-hemato_data$LIMF.10000/hemato_data$WBC.10000
hemato_data$Eosinophils<-hemato_data$EOZ.10000/hemato_data$WBC.10000
hemato_data$H.L<-hemato_data$HET.10000/hemato_data$LIMF.10000
hemato_data<-hemato_data[, c("ring","sex","Heterophils","Limphocytes","Eosinophils","H.L","WBC.10000")]
hemato_data
```

```
##      ring sex Heterophils Limphocytes Eosinophils      H.L WBC.10000
## 1     01  M  0.7058824   0.2941176   0.0000000 2.4000000 21.963824
## 2     02  M  0.1538462   0.8461538   0.0000000 0.1818182 19.117647
## 3     03  M           NA           NA           NA           NA           NA
## 4     04  F           NA           NA           NA           NA           NA
## 5     05  F  0.0000000   1.0000000   0.0000000 0.0000000 15.673981
## 6     06  F           NA           NA           NA           NA           NA
## 7     07  F  0.1666667   0.8333333   0.0000000 0.2000000  8.538899
## 8     08  M  0.2222222   0.7777778   0.0000000 0.2857143 11.344538
## 9     09  F  0.3913043   0.6086957   0.0000000 0.6428571 19.873272
## 10    10  M  0.1562500   0.7812500   0.0625000 0.2000000 21.126761
## 11    11  M  0.4800000   0.4800000   0.0400000 1.0000000 12.413108
## 12    12  F  0.4615385   0.5384615   0.0000000 0.8571429 11.054422
## 13    13  F  0.3548387   0.5161290   0.1290323 0.6875000 20.758929
## 14    14  F           NA           NA           NA           NA           NA
## 15    15  F           NA           NA           NA           NA           NA
## 16    17  M           NA           NA           NA           NA           NA
## 17    19  M           NA           NA           NA           NA           NA
## 18    XX  M  0.3333333   0.6666667   0.0000000 0.5000000  9.265614
```

5.3 Mean and standard deviation per sex

```
t_hemato_data<-as.data.frame(t(hemato_data))
t_hemato_data
```



```
##          V1          V2  V3  V4          V5  V6          V7          V8
## ring          01          02  03  04          05  06          07          08
## sex          M          M  M  F          F  F          F          M
## Heterophils 0.7058824 0.1538462 <NA> <NA> 0.0000000 <NA> 0.1666667 0.2222222
## Lymphocytes 0.2941176 0.8461538 <NA> <NA> 1.0000000 <NA> 0.8333333 0.7777778
## Eosinophils 0.0000000 0.0000000 <NA> <NA> 0.0000000 <NA> 0.0000000 0.0000000
## H.L          2.4000000 0.1818182 <NA> <NA> 0.0000000 <NA> 0.2000000 0.2857143
## WBC.10000    21.963824 19.117647 <NA> <NA> 15.673981 <NA> 8.538899 11.344538
##          V9          V10        V11        V12        V13  V14  V15  V16
## ring          09          10          11          12          13  14  15  17
## sex          F          M          M          F          F  F  F  M
## Heterophils 0.3913043 0.1562500 0.4800000 0.4615385 0.3548387 <NA> <NA> <NA>
## Lymphocytes 0.6086957 0.7812500 0.4800000 0.5384615 0.5161290 <NA> <NA> <NA>
## Eosinophils 0.0000000 0.0625000 0.0400000 0.0000000 0.1290323 <NA> <NA> <NA>
## H.L          0.6428571 0.2000000 1.0000000 0.8571429 0.6875000 <NA> <NA> <NA>
## WBC.10000    19.873272 21.126761 12.413108 11.054422 20.758929 <NA> <NA> <NA>
##          V17          V18
## ring          19          XX
## sex          M          M
## Heterophils <NA> 0.3333333
## Lymphocytes <NA> 0.6666667
## Eosinophils <NA> 0.0000000
## H.L          <NA> 0.5000000
## WBC.10000    <NA> 9.265614
```

```
colnames(t_hemato_data)<-t_hemato_data[c(1),]
colnames(t_hemato_data)[18]<-18
t_hemato_data<-t_hemato_data[-c(1),]
t_hemato_data$Parameter<-rownames(t_hemato_data)
t_hemato_data<-t_hemato_data[,c(19,1:18)]
rownames(t_hemato_data)<-NULL

hemato_data_m<-t_hemato_data[,c(2,3,9,11,12,19)]
rownames(hemato_data_m)<-t_hemato_data$Parameter
hemato_data_m<-hemato_data_m[-c(1),]
hemato_data_m$Sex<-"Male"
colnames(hemato_data_m)<-c("indv1","indv2","indv3","indv4","indv5","indv6","Sex")
hemato_data_m$Parameter<-rownames(hemato_data_m)
hemato_data_m<-hemato_data_m[,c(8,1:7)]
rownames(hemato_data_m)<-NULL

hemato_data_f<-t_hemato_data[,c(6,8,10,13,14)]
rownames(hemato_data_f)<-t_hemato_data$Parameter
hemato_data_f<-hemato_data_f[-c(1),]
hemato_data_f$Sex<-"Female"
hemato_data_f$indv6<-NA
hemato_data_f<-hemato_data_f[,c(1:5,7,6)]
colnames(hemato_data_f)<-c("indv1","indv2","indv3","indv4","indv5","indv6","Sex")
hemato_data_f$Parameter<-rownames(hemato_data_f)
hemato_data_f<-hemato_data_f[,c(8,1:7)]
rownames(hemato_data_f)<-NULL

hemato_data_mf<-as.data.frame(rbind(hemato_data_m,hemato_data_f))
hemato_data_mf[,c(2:7)]<-lapply(hemato_data_mf[,c(2:7)],as.numeric)
```

```
hemato_data_mf$Mean <- rowMeans(hemato_data_mf[, -c(1,8)], na.rm = TRUE)
hemato_data_mf$SD <- apply(hemato_data_mf[, c(2:7)], 1, sd, na.rm = TRUE)
hemato_data_mf[,c(1,8)]<-lapply(hemato_data_mf[,c(1,8)],as.factor)
```

```
View(hemato_data_mf)
```

```
hemato_data_mf
```

```
##      Parameter      indv1      indv2      indv3      indv4      indv5      indv6
## 1  Heterophils  0.7058824  0.1538462  0.2222222  0.1562500  0.4800000  0.3333333
## 2  Lymphocytes  0.2941176  0.8461538  0.7777778  0.7812500  0.4800000  0.6666667
## 3  Eosinophils  0.0000000  0.0000000  0.0000000  0.0625000  0.0400000  0.0000000
## 4      H.L.    2.4000000  0.1818182  0.2857143  0.2000000  1.0000000  0.5000000
## 5   WBC.10000 21.9638240 19.1176470 11.3445380 21.1267610 12.4131080 9.2656140
## 6  Heterophils  0.0000000  0.1666667  0.3913043  0.4615385  0.3548387      NA
## 7  Lymphocytes  1.0000000  0.8333333  0.6086957  0.5384615  0.5161290      NA
## 8  Eosinophils  0.0000000  0.0000000  0.0000000  0.0000000  0.1290323      NA
## 9      H.L.    0.0000000  0.2000000  0.6428571  0.8571429  0.6875000      NA
## 10 WBC.10000 15.6739810  8.5388990 19.8732720 11.0544220 20.7589290      NA
##      Sex      Mean      SD
## 1   Male  0.34192235 0.21715484
## 2   Male  0.64099432 0.21324797
## 3   Male  0.01708333 0.02740514
## 4   Male  0.76125542 0.85879353
## 5   Male 15.87191533 5.50205851
## 6  Female  0.27486964 0.18845327
## 7  Female  0.69932390 0.20977082
## 8  Female  0.02580646 0.05770500
## 9  Female  0.47750000 0.36075761
## 10 Female 15.17990060 5.35069819
```

6 Morphometry

6.1 Loading and wrangling data

```
data <- read.csv("Data/phalaropes_al_captures.csv")
df <- data[!data$Age=="Chick", c(3,6,7,8,14:19)]
View(df)
df$Sex. <- as.factor(df$Sex.)
colnames(df)[c(5,7:9)] <- c("sex", "weight", "wing", "tarsus")
```

6.2 Mean and standard deviation per sex

6.2.1 Wing

```
round(mean(df$wing[df$sex == "Female"]),1)
```

```
## [1] 117.4
```

```
round(sd(df$wing[df$sex == "Female"]),1)
```

```
## [1] 2.7
```

```
round(mean(df$wing[df$sex == "Male"]),1)
```

```
## [1] 109.8
```

```
round(sd(df$wing[df$sex == "Male"]),1)
```

```
## [1] 3.7
```

6.2.2 Body mass

```
round(mean(df$weight[df$sex == "Female"]),1)
```

```
## [1] 44.2
```

```
round(sd(df$weight[df$sex == "Female"]),1)
```

```
## [1] 4.9
```

```
round(mean(df$weight[df$sex == "Male"]),1)
```

```
## [1] 35.4
```

```
round(sd(df$weight[df$sex == "Male"]),1)
```

```
## [1] 3.7
```

6.2.3 Scaled mass index (SMI)

Should we use wing or tarsus?

```
cor(log(df$weight), log(df$wing)) #wing is better
```

```
## [1] 0.6513992
```

```
cor(log(df$weight), log(df$tarsus))
```

```
## [1] -0.2169538
```

Should we calculate SMI separately by sex or altogether? We need to see if the allometric relationship varies significantly between the sexes.

```
df$smi.sex <- NA
```

```
SMA.males <- sma(log(weight) ~ log(wing), data = df[df$sex=="Male",])  
summary(SMA.males)
```

```
## Call: sma(formula = log(weight) ~ log(wing), data = df[df$sex == "Male",  
##      ])  
##  
## Fit using Standardized Major Axis  
##  
## -----  
## Coefficients:  
##           elevation      slope  
## estimate    -11.111905  3.123576  
## lower limit -23.794986  1.428906  
## upper limit   1.571176  6.828110  
##  
## H0 : variables uncorrelated  
## R-squared : 0.06487684  
## P-value : 0.50835
```

```
df[df$sex=="Male",]$smi.sex <- df[df$sex=="Male",]$weight*((mean(df[df$sex=="Male",]$wing, na.rm = T))/  
SMA.males$estimate)  
summary(SMA.females)
```

```
## Call: sma(formula = log(weight) ~ log(wing), data = df[df$sex == "Female",  
##      ])  
##  
## Fit using Standardized Major Axis  
##  
## -----  
## Coefficients:  
##           elevation      slope  
## estimate    -19.436754  4.872114  
## lower limit -40.092290  2.186784  
## upper limit   1.218782 10.854980  
##  
## H0 : variables uncorrelated  
## R-squared : 0.009313183  
## P-value : 0.80492
```

```
df[df$sex=="Female",]$smi.sex <- df[df$sex=="Female",]$weight*((mean(df[df$sex=="Female",]$wing, na.rm = T)/
```

Since there is no significant sexual difference in the relationship between body mass and wing, we can take the smi combined.

```
df$smi <- NA
```

```
SMA <- sma(log(weight) ~ log(wing), data = df)  
summary(SMA)
```

```
## Call: sma(formula = log(weight) ~ log(wing), data = df)
```

```
##
```

```
## Fit using Standardized Major Axis
```

```
##
```

```
## -----
```

```
## Coefficients:
```

```
##           elevation      slope
```

```
## estimate    -12.697374  3.459508
```

```
## lower limit  -19.280220  2.337615
```

```
## upper limit   -6.114527  5.119831
```

```
##
```

```
## H0 : variables uncorrelated
```

```
## R-squared : 0.424321
```

```
## P-value : 0.0034058
```

```
df$smi <- df$weight*((mean(df$wing, na.rm = T))/(df$wing))^SMA$groupsummary$Slope
```

```
round(mean(df$smi[df$sex == "Female"]),1)
```

```
## [1] 39.5
```

```
round(sd(df$smi[df$sex == "Female"]),1)
```

```
## [1] 5.2
```

```
round(mean(df$smi[df$sex == "Male"]),1)
```

```
## [1] 40.1
```

```
round(sd(df$smi[df$sex == "Male"]),1)
```

```
## [1] 5.4
```

6.2.4 T-test

```
t.test(wing~sex, df)
```

```
##
```

```
## Welch Two Sample t-test
```

```
##
```

```
## data: wing by sex
```

```
## t = 5.0525, df = 14.698, p-value = 0.0001522
```

```
## alternative hypothesis: true difference in means between group Female and group Male is not equal to
```

```
## 95 percent confidence interval:
```

```
##    4.426638 10.906695
```

```
## sample estimates:
```

```
## mean in group Female    mean in group Male
```

```
##           117.4444           109.7778
```

```
t.test(weight~sex, df)
```

```
##
## Welch Two Sample t-test
##
## data: weight by sex
## t = 4.2764, df = 14.763, p-value = 0.0006854
## alternative hypothesis: true difference in means between group Female and group Male is not equal to
## 95 percent confidence interval:
## 4.379892 13.108996
## sample estimates:
## mean in group Female mean in group Male
## 44.16667 35.42222
```

```
t.test(smi~sex, df)
```

```
##
## Welch Two Sample t-test
##
## data: smi by sex
## t = -0.23777, df = 15.988, p-value = 0.8151
## alternative hypothesis: true difference in means between group Female and group Male is not equal to
## 95 percent confidence interval:
## -5.880225 4.694269
## sample estimates:
## mean in group Female mean in group Male
## 39.49130 40.08428
```

7 Body condition across time

We need to calculate the day from first arrival of each bird captured across the years.

16 May - first bird spotted in site in 2021 14 May - first bird spotted in site in 2022 15 May - first bird spotted in site in 2023

```
df$day <- c(39,39,39,40,40,40,41,41,24,25,26,27,27,27,30,43,18,20)
```

7.1 Linear model

We compared between different models to see whether both day-of-the-year or sex had an effect on the body condition

```
M0 <- lm(log(smi) ~ 1, data = df)
M1 <- lm(log(smi) ~ day, data = df)
M2 <- lm(log(smi) ~ day + sex, data = df)
M3 <- lm(log(smi) ~ day * sex, data = df)
M4 <- lm(log(smi) ~ day + I(day^2) + sex, data = df)

AIC(M0, M1, M2, M3, M4)
```

```
##      df      AIC
## M0  2 -19.73874
## M1  3 -17.74444
## M2  4 -15.81543
## M3  5 -16.15198
## M4  5 -15.98713
```

```
anova(M1, M2, M3, M4, test = "F")
```

```
## Analysis of Variance Table
##
## Model 1: log(smi) ~ day
## Model 2: log(smi) ~ day + sex
## Model 3: log(smi) ~ day * sex
## Model 4: log(smi) ~ day + I(day^2) + sex
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1      16 0.28178
## 2      15 0.28067  1  0.001109 0.0630 0.8055
## 3      14 0.24650  1  0.034167 1.9405 0.1853
## 4      14 0.24877  0 -0.002268
```

#The simplest model is the best one, and is not significant

```
summary(M1)
```

```
##
## Call:
## lm(formula = log(smi) ~ day, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.224460 -0.108060 -0.009934  0.088285  0.231315
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.6846544  0.1293719  28.481 3.88e-15 ***
## day         -0.0002745  0.0038560  -0.071   0.944
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1327 on 16 degrees of freedom
## Multiple R-squared:  0.0003166, Adjusted R-squared:  -0.06216
## F-statistic: 0.005067 on 1 and 16 DF, p-value: 0.9441
```

```
shapiro.test(residuals(M1))
```

```
##
## Shapiro-Wilk normality test
##
## data: residuals(M1)
## W = 0.97583, p-value = 0.8972
```

```
bptest(M1)
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: M1  
## BP = 1.666, df = 1, p-value = 0.1968
```

```
sctest(M1, type = "rec-matched")
```

```
##  
## M-fluctuation test  
##  
## data: M1  
## f(efp) = 0.89107, p-value = 0.6462
```

```
cooksds <- cooks.distance(M1)  
which(cooksds > 4/length(cooksds))
```

```
## 16  
## 16
```

8 Figure 1 b

```
p1 <- ggplot(data=df, aes(x=day, y=log(smi), shape=sex))+  
  geom_point(size = 4, alpha=0.9)+  
  xlab("Days from first arrival")+  
  ylab("Scaled mass index (log)") +  
  scale_shape_manual(values=c(1, 16))+  
  theme(  
    legend.position = "none",  
    panel.background = element_rect(fill = NA),  
    panel.grid = element_blank(),  
    axis.line = element_line(size = 1.2, colour = "black"),  
    axis.text = element_text(size = 14, color = "black"),  
    axis.title = element_text(size = 16, color = "black"),  
    axis.ticks = element_line(color = "black"),  
    plot.title = element_text(face = "plain", size = 14)  
  )  
p1
```