Fric et al. Re-analysis Code

Vaughn Shirey & Elise Larsen

Current version 2-Dec-2020; initiated 9-Mar-2020

*

Begin Analysis

This code chunk sets up the workspace and loads necessary packages. If phest is not already installed, remove comment from install line.

```
# load libraries
library(tidyverse)
library(ggplot2)
library(ggExtra)
library(gridExtra)
#library(devtools); install_github("willpearse/phest")
library(phest)
library(readxl)
library(lubridate)
```

*

Data Import and Formatting

data.csv file was downloaded from https://doi.org/10.6084/m9.figshare.9946934 (https://doi.org/10.6084/m9.figshare.9946934) (https://figshare.com/articles/Phenology_responses_of_temperate_butterflies_-_Supplementary_data/9946934 (https://figshare.com/articles/Phenology_responses_of_temperate_butterflies_-_Supplementary_data/9946934))

This cvs file contains the occurrence data used in Fric et al. (2020), which they downloaded from gbif. The file includes separate data tables for each dataset, which have been concatenated into one file. These data tables have the same fields but are not formatted as a single data table; individual datasets were all written into one data file, including headers and row indices in each dataset. This first set of code reformats the data & writes formatted data files.

```
load("data/occurrences.RData")

#Fric et al removed all 1st of month observations and removed one species due to late season nes
ts
fricdata<-filter(alldata, day!=1, name!="Euphydryas aurinia")

summary(fricdata)
```

```
##    row.index         name           decimalLongitude   decimalLatitude
##  Min.   :    1   Length:275457      Min.   :-162.559   Min.   : 5.787
##  1st Qu.: 2340   Class :character   1st Qu.:  -2.676   1st Qu.:52.823
##  Median : 7074   Mode  :character   Median :   9.564   Median :55.775
##  Mean   :15039                      Mean   :   6.716   Mean   :56.354
##  3rd Qu.:20814                      3rd Qu.:  23.763   3rd Qu.:60.677
##  Max.   :85273                      Max.   :  59.333   Max.   :71.216
##
##       year          month        country              day
##  Min.   :1616   Min.   : 1.0   Length:275457      Min.   : 2.00
##  1st Qu.:1992   1st Qu.: 6.0   Class :character   1st Qu.: 9.00
##  Median :2002   Median : 7.0   Mode  :character   Median :16.00
##  Mean   :1996   Mean   : 6.5                      Mean   :16.19
##  3rd Qu.:2009   3rd Qu.: 7.0                      3rd Qu.:24.00
##  Max.   :2015   Max.   :12.0                      Max.   :31.00
##  NA's   :57
##     SuccDay          rndLat           alt              region
##  Min.   :  2.0   Min.   : 6.00   Min.   :-2666.74   Length:275457
##  1st Qu.:164.0   1st Qu.:53.00   1st Qu.:   23.25   Class :character
##  Median :186.0   Median :56.00   Median :   64.24   Mode  :character
##  Mean   :181.2   Mean   :56.29   Mean   :  113.64
##  3rd Qu.:201.0   3rd Qu.:61.00   3rd Qu.:  110.77
##  Max.   :361.0   Max.   :71.00   Max.   : 4305.17
##
```

```
#Save formatted and filtered occurrence data used by Fric et al.
save(fricdata,file="data/occurrences_FricAnalysis.RData")
```

*

Data Exploration

Data have now been formatted and filtered to mirror the data used by Fric et al. (2020) and stored into the “occur” tibble.

The following code explores some aspects of the data use in the Fric et al. analysis, but a more complete exploration is in the DataCuration file.

```
#Tally the number of observations per dataset & calculate how each dataset spans latitude, year, altitude
spans.summary<-fricdata %>%
  group_by(name, region) %>%
  add_count(name="fric_n") %>% ## n. records
  group_by(name, region, fric_n) %>%
  summarize(lat_span=(max(rndLat, na.rm=T)-min(rndLat, na.rm=T)),
         year_span=(max(year, na.rm=T)-min(year, na.rm=T)),
         alt_span=round((max(alt, na.rm=T)-min(alt, na.rm=T)),0))
```

```
## `summarise()` regrouping output by 'name', 'region' (override with `.groups` argument)
```

```
#calculate # latitudes, onsets, terminations, flight curves = 0
endpt.summary<-fricdata %>%
  group_by(name, region, rndLat) %>%
  # count no. records by latitudinal band
  add_count(name="n_recs") %>%
  #filter to onset & offset dates and label onset dates and offset dates
  filter(SuccDay==min(SuccDay) | SuccDay==max(SuccDay)) %>%
  mutate(onset=ifelse(SuccDay==min(SuccDay),1,0),    term=ifelse(SuccDay==max(SuccDay),1,0)) %>%
  group_by(name, region) %>%
  #create summary statistics by species & region
  summarize(n_lat=length(unique(rndLat)), n_onset=sum(onset), n_term=sum(term), n_flightcurve0s=
sum(n_recs==1) )
```

```
## `summarise()` regrouping output by 'name' (override with `.groups` argument)
```

```
#combine summary tables
fric.data.summary<-merge(spans.summary, endpt.summary, by=intersect(names(spans.summary), names
(endpt.summary)))
rm(spans.summary)
summary(fric.data.summary)
```

```
##      name              region              fric_n          lat_span    
##  Length:110         Length:110         Min.   :   15.0   Min.   :10.00  
##  Class :character   Class :character   1st Qu.:   82.5   1st Qu.:24.00  
##  Mode  :character   Mode  :character   Median :  190.5   Median :27.00  
##                                        Mean   : 2504.2   Mean   :26.38  
##                                        3rd Qu.: 1108.2   3rd Qu.:30.00  
##                                        Max.   :51819.0   Max.   :64.00  
##    year_span        alt_span        n_lat          n_onset          n_term     
##  Min.   : 64.0   Min.   : 530   Min.   : 5.00   Min.   : 5.00   Min.   : 5.00  
##  1st Qu.:102.0   1st Qu.:2105   1st Qu.:13.25   1st Qu.:15.00   1st Qu.:14.25  
##  Median :117.0   Median :2699   Median :18.00   Median :19.00   Median :20.00  
##  Mean   :127.0   Mean   :2713   Mean   :19.03   Mean   :20.54   Mean   :20.25  
##  3rd Qu.:138.8   3rd Qu.:3463   3rd Qu.:25.00   3rd Qu.:27.00   3rd Qu.:26.00  
##  Max.   :399.0   Max.   :5163   Max.   :33.00   Max.   :39.00   Max.   :35.00  
##  n_flightcurve0s 
##  Min.   : 0.000  
##  1st Qu.: 2.000  
##  Median : 3.000  
##  Mean   : 3.409  
##  3rd Qu.: 5.000  
##  Max.   :10.000  
```

*

Explore data by altitude & latitude

This code chunk explores the spatiotemporal representation in the fric.data dataset.

Create Figure 1: Occurrences by altitude & latitude

This code outputs Larsen & Shirey Figure 1, which uses the 4 species presented in Fric et al. Figure 1, to demonstrate the spatiotemporal biases as well as the prevalence of flight periods with a duration of 0 days.

```
summary(fricdata$alt)
```

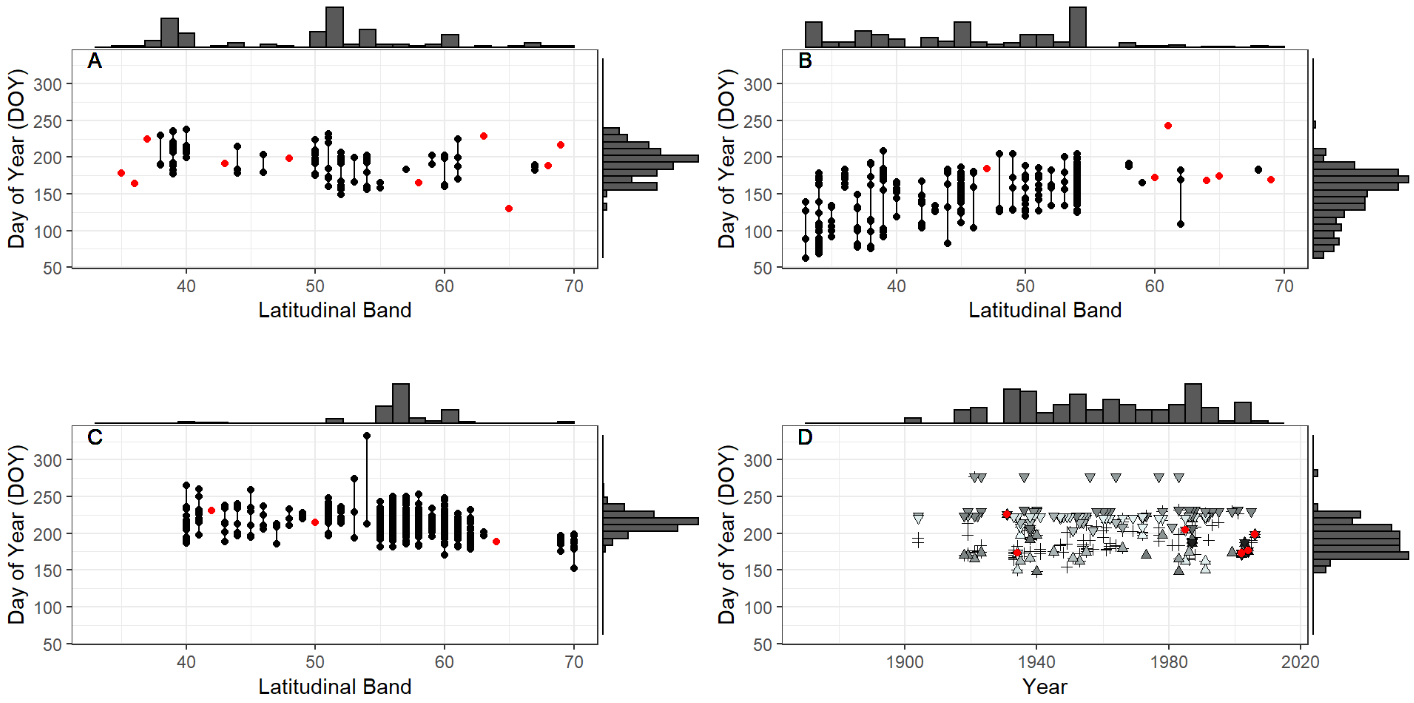
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
## -2666.74    23.25    64.24   113.64   110.77  4305.17
```

```
#hist(fricdata$alt)
summary(fricdata$decimalLatitude)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   5.787  52.823  55.775  56.354  60.677  71.216
```

```
#hist(fricdata$decimalLatitude)
##Create Figure 1
#species list
fric.datasets<-fricdata %>% group_by(name, region) %>% tally()

fig1sp<-c("Agriades glandon","Glaucopsyche lygdamus","Hesperia comma","Parnassius smintheus")

#Filter data to these species
fig1data<-fricdata %>%
  filter(name %in% fig1sp)

#Get onset & termination dates (SuccDay)
f1.pheno.data<-fig1data %>%
  group_by(name, region, rndLat) %>%
  mutate(onset=min(SuccDay), term=max(SuccDay), fp=term-onset, singles=ifelse(length(SuccDay)==1
,1,0))

f1.pheno.data2<-f1.pheno.data %>%
  filter(SuccDay==onset | SuccDay==term)

#A list to store plot panels
tempplot<-list()
fig1panels<-list()

tags<-c("A","B","C","D")

#Create Panels
for(i in 1:2) {
  #paneltitle<-paste(fig1sp[i],"N. America")
  tempplot[[i]] <- ggplot(filter(f1.pheno.data, name==fig1sp[i], region=="N. America"), aes(x=rn
dLat, y=SuccDay, color=as.factor(singles))) +
    theme_bw() +
    theme(legend.position="none", plot.margin = margin(1,1,1,1, "in")) +
    geom_segment(data=filter(f1.pheno.data2, name==fig1sp[i], region=="N. America"), aes(x=rndLa
t, y=onset, xend=rndLat, yend=term)) +
    geom_point(aes(color=as.factor(singles))) +
    scale_color_manual(values=c("black","red")) +
    xlim(min(f1.pheno.data$rndLat),max(f1.pheno.data$rndLat)) + ylim(min(f1.pheno.data$SuccDay),
max(f1.pheno.data$SuccDay)) +
    labs(x="Latitudinal Band", y="Day of Year (DOY)", title="") + geom_text(x=min(f1.pheno.data
$rndLat), y=max(f1.pheno.data$SuccDay), label=tags[i])
  # with marginal histograms
  fig1panels[[i]] <- ggMarginal(tempplot[[i]], type="histogram")
  }

i<-3 #H. comma panel in Fric et al. is from Europe
#paneltitle<-paste(fig1sp[i],"Europe")
  tempplot[[i]] <- ggplot(filter(f1.pheno.data, name==fig1sp[i], region=="Europe"), aes(x=rndLa
t, y=SuccDay, color=as.factor(singles))) +
    theme_bw() +
    theme(legend.position="none", plot.margin = margin(1,1,1,1, "in")) +
    geom_segment(data=filter(f1.pheno.data2, name==fig1sp[i], region=="Europe"), aes(x=rndLat, y
=onset, xend=rndLat, yend=term)) +
    geom_point(aes(color=as.factor(singles))) +
    scale_color_manual(values=c("black","red")) +
    xlim(min(f1.pheno.data$rndLat),max(f1.pheno.data$rndLat)) + ylim(min(f1.pheno.data$SuccDay),max(f1.pheno.data$SuccDay)) +
    labs(x="Latitudinal Band", y="Day of Year (DOY)", title="") + geom_text(x=min(f1.pheno.data$rndLat), y=max(f1.pheno.data$SuccDay), label=tags[i])

  # with marginal histogram
  fig1panels[[i]] <- ggMarginal(tempplot[[i]], type="histogram")

##### Figure 1d  2020-07-29 update uses YEAR and DAY to mirror Fric et al.
i<-4
#paneltitle<-paste(fig1sp[i],"N. America")
tempplot[[i]]<- ggplot(filter(f1.pheno.data, name==fig1sp[i], region=="N. America"), aes(x=year, y=SuccDay, fill=decimalLatitude)) +
  geom_point(shape=3) +
  theme_bw() +
  theme(legend.position="none", plot.margin = margin(1,1,1,1, "in")) +
  geom_point(data=filter(f1.pheno.data2, name==fig1sp[i], region=="N. America"), aes(x=year, y=onset, fill=decimalLatitude), shape=24) +
  geom_point(data=filter(f1.pheno.data, name==fig1sp[i], region=="N. America"), aes(x=year, y=term, fill=decimalLatitude), shape=25) +
  scale_fill_gradient(low="azure1", high="black") +
  geom_point(data=filter(f1.pheno.data2, name==fig1sp[i], region=="N. America", singles==1), aes(x=year, y=SuccDay), color="red", shape=16) +
  xlim(min(f1.pheno.data$year),max(f1.pheno.data$year)) + ylim(min(f1.pheno.data$SuccDay),max(f1.pheno.data$SuccDay)) +
  labs(x="Year", y="Day of Year (DOY)", title="") + geom_text(x=min(f1.pheno.data$year), y=max(f1.pheno.data$SuccDay), label=tags[i])

# with marginal histogram
fig1panels[[i]]  <- ggMarginal(tempplot[[i]], type="histogram")

grid.arrange(grobs=fig1panels[c(1:4)], nrow=2, ncol=2, top="Visualization of data used in Fric et al. for \n (A) Agriades glandon      (B)Glaucopsyche lygdamus \n (C)Hesperia comma      (D)Parnassius smintheus")
```

Visualization of data used in Fric et al. for
(A) Agriades glandon (B)Glaucopsyche lygdamus
(C)Hesperia comma (D)Parnassius smintheus



```
#Used to create figure 1 pdf
#pdf_filename<-("outputs/Larsen&Shirey2020_FinalFig1.pdf")
#ggsave(pdf_filename, arrangeGrob(grobs=fig1panels[ds1], nrow=2, ncol=2), width=6, height=6, uni
ts="in", scale=1,dpi=600)
```

*

Data curation

Data have now been formatted, identified by region, and summarized.

The following code chunk applies the filters used in the Larsen & Shirey reanalysis and calculates summary data density statistics for all species present in Fric's results to output to Supplemental Table 1.

Our reanalysis excludes datasets along two axes - data density, and voltinism. This code examines data along the data density axis. Unlike Fric et al., we include first day of the month records. We curate raw occurrence data with the following filters prior to estimating phenometrics:

- 1 - remove Euphydryas aurinia (as Fric et al. did)
- 2 - altitude in [0m,500m]
- 3 - DOY in (60,330) which corresponds to start of march to late november
- 4 - 10 or more records when data is grouped by species, region, year, and latitudinal band

```
#Summarize data availability for Larsen & Shirey re-analysis
#Now, filter data for altitude & for cases with 10 or more records by species-region-year-latitu
de
all.datasets<-alldata %>% group_by(name, region) %>% tally()
new.data.summary<-alldata %>%
  filter(between(alt,0,500), name!="Euphydryas aurinia", month %in% c(3:11)) %>%
  # calculate data availability by species, region, latitude & year
  group_by(name, region, rndLat, year) %>%
  add_count(name="group_n") %>% ## n. observations per group
  filter(group_n>=10) %>% ### filter by 10 or more observations in group
  # calculate reanalysis statistics by species & region
  group_by(name, region) %>%
  add_count(name="curated_n_obs") %>%
  group_by(name, region, curated_n_obs) %>%
  #calculate summary statistics applying data filters
  summarize(curated_n_lat=length(unique(rndLat)),  curated_n_fcurve=length(unique(paste(rndLat,y
ear))),
            curated_lat_span=(max(rndLat, na.rm=T)-min(rndLat, na.rm=T)),
            curated_year_span=(max(year, na.rm=T)-min(year, na.rm=T)),
            curated_alt_span=round((max(alt, na.rm=T)-min(alt, na.rm=T)),0))
```

```
## `summarise()` regrouping output by 'name', 'region' (override with `.groups` argument)
```

```
#combine summary tables
supptable1<-merge(fric.data.summary, new.data.summary, by=intersect(names(fric.data.summary), na
mes(new.data.summary)), all.x=T)
head(supptable1)
```

```
##                      name     region fric_n lat_span year_span alt_span n_lat
## 1       Agriades glandon N. America    110       34       103     4042    26
## 2    Amblyscirtes vialis N. America     88       29       133     2775    19
## 3 Anthocharis cardamines     Europe  31849       32       168     2595    33
## 4       Anthocharis sara N. America    218       28       111     4417    21
## 5  Aphantopus hyperantus     Europe  30598       25       399     2102    26
## 6        Aporia crataegi     Europe   5172       64       165     2520    33
##   n_onset n_term n_flightcurve0s curated_n_obs curated_n_lat curated_n_fcurve
## 1      27     27              10            NA            NA               NA
## 2      19     19               7            NA            NA               NA
## 3      39     35               2         29134            17              393
## 4      22     22               6            NA            NA               NA
## 5      27     28               1         27879            15              330
## 6      34     35               2          4055            10              108
##   curated_lat_span curated_year_span curated_alt_span
## 1               NA                NA               NA
## 2               NA                NA               NA
## 3               16                80              499
## 4               NA                NA               NA
## 5               14                79              488
## 6               13                74              268
```

```
summary(supptable1)
```

```
##      name              region              fric_n           lat_span    
##  Length:110         Length:110         Min.   :   15.0   Min.   :10.00  
##  Class :character   Class :character   1st Qu.:   82.5   1st Qu.:24.00  
##  Mode  :character   Mode  :character   Median :  190.5   Median :27.00  
##                                        Mean   : 2504.2   Mean   :26.38  
##                                        3rd Qu.: 1108.2   3rd Qu.:30.00  
##                                        Max.   :51819.0   Max.   :64.00  
##                                                                         
##    year_span        alt_span        n_lat          n_onset          n_term     
##  Min.   : 64.0   Min.   : 530   Min.   : 5.00   Min.   : 5.00   Min.   : 5.00  
##  1st Qu.:102.0   1st Qu.:2105   1st Qu.:13.25   1st Qu.:15.00   1st Qu.:14.25  
##  Median :117.0   Median :2699   Median :18.00   Median :19.00   Median :20.00  
##  Mean   :127.0   Mean   :2713   Mean   :19.03   Mean   :20.54   Mean   :20.25  
##  3rd Qu.:138.8   3rd Qu.:3463   3rd Qu.:25.00   3rd Qu.:27.00   3rd Qu.:26.00  
##  Max.   :399.0   Max.   :5163   Max.   :33.00   Max.   :39.00   Max.   :35.00  
##                                                                                
##  n_flightcurve0s  curated_n_obs     curated_n_lat    curated_n_fcurve
##  Min.   : 0.000   Min.   :   10.0   Min.   : 1.000   Min.   :  1.00  
##  1st Qu.: 2.000   1st Qu.:   36.5   1st Qu.: 1.500   1st Qu.:  2.50  
##  Median : 3.000   Median :  361.0   Median : 3.000   Median : 23.00  
##  Mean   : 3.409   Mean   : 3858.6   Mean   : 6.055   Mean   : 77.93  
##  3rd Qu.: 5.000   3rd Qu.: 3928.0   3rd Qu.:10.000   3rd Qu.:124.50  
##  Max.   :10.000   Max.   :47617.0   Max.   :17.000   Max.   :393.00  
##                   NA's   :55        NA's   :55       NA's   :55      
##  curated_lat_span curated_year_span curated_alt_span
##  Min.   : 0.000   Min.   :  0.00    Min.   :  0.0   
##  1st Qu.: 0.500   1st Qu.:  8.00    1st Qu.:214.0   
##  Median : 5.000   Median : 34.00    Median :379.0   
##  Mean   : 6.545   Mean   : 43.29    Mean   :324.7   
##  3rd Qu.:11.500   3rd Qu.: 74.00    3rd Qu.:467.5   
##  Max.   :18.000   Max.   :123.00    Max.   :499.0   
##  NA's   :55       NA's   :55        NA's   :55
```

```
#output summary table to csv file
#write_csv(supptable1, "Larsen&Shirey_stats_supp_table1.csv")
rm(fric.data.summary, new.data.summary, endpt.summary)
```

*

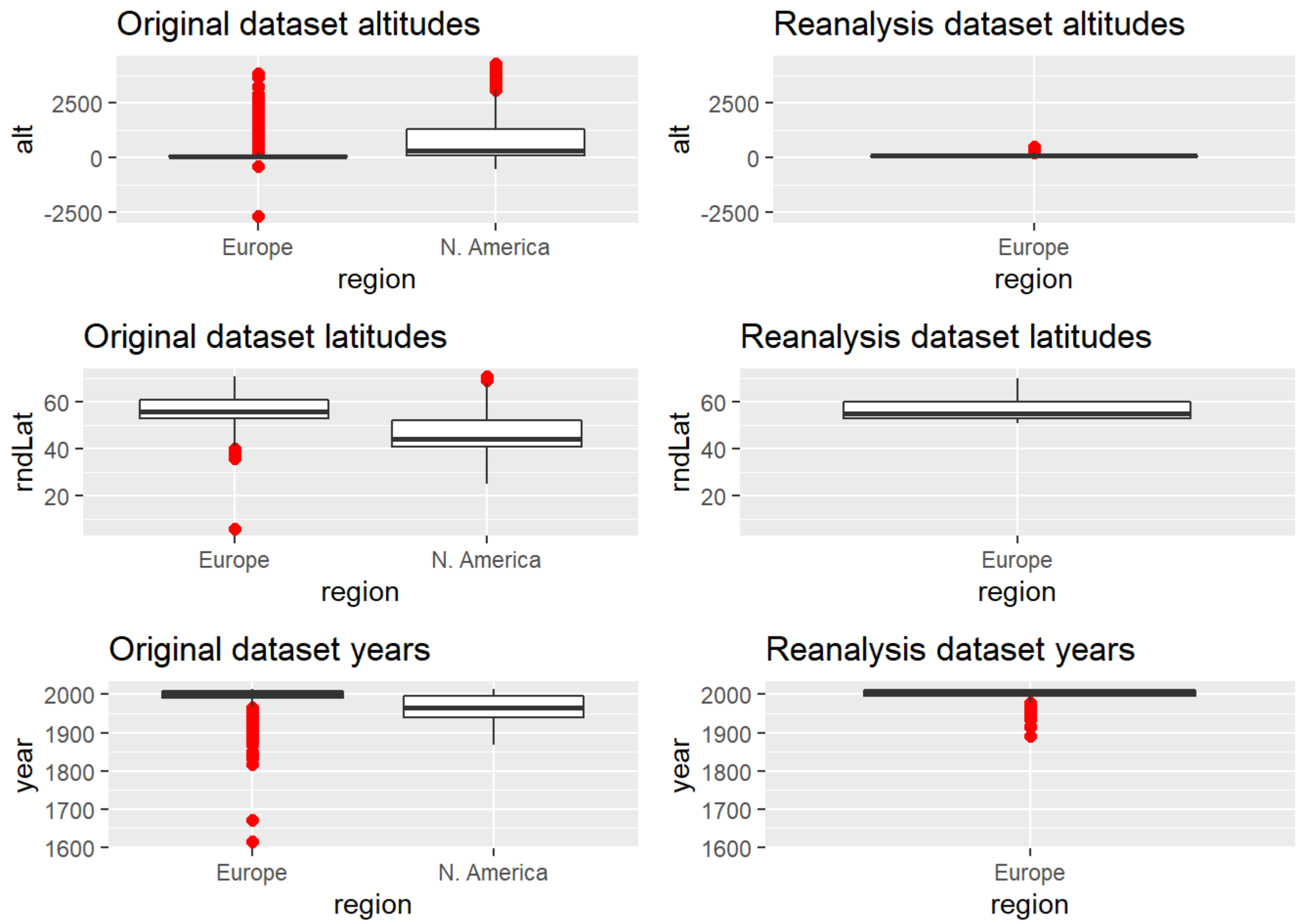
Data curation for reanalysis

This code filters occurrence data for reanalysis by voltinism and data density, and visualizes some differences between datasets curated for the original analysis and this reanalysis. We only include datasets with sufficient data for calculating phenometrics at 3 or more distinct latitudinal bands, so that a linear model can be applied.

```
#FILTER DATA BY VOLTINISM

#get species list without evidence of multiple generations
#Euphydryas aurinia is not included in the voltinism file
voltindata<-read_csv("data/voltinism.csv")
```

```
## Parsed with column specification:
## cols(
##   id = col_double(),
##   name_datafile = col_character(),
##   name_resultsfile = col_character(),
##   region = col_character(),
##   Voltinism = col_character(),
##   Voltinism_source = col_character(),
##   `In reanalysis?` = col_double(),
##   Why_excluded = col_character()
## )
```

```
voltindata<-na.omit(voltindata[,c(1:8)])
voltindata<-voltindata %>% select(name=name_resultsfile,region,Voltinism)
multi<-c("Bivoltine", "Multivoltine","Sometimes bivoltine","Possible bivoltinism in some subsp."
,"Unconfirmed reports of second brood")
univoltine<-filter(voltindata, !Voltinism %in% multi)
rm(voltindata, multi)

#filter occurrence dataset to these species
reanalysis.data<-merge(alldata, univoltine, by=intersect(names(alldata),names(univoltine)))

#filter data by altitude and data density
reanalysis.data<-reanalysis.data %>%
  filter(between(alt,0,500), month %in% c(3:11)) %>%
  # calculate data availability by species, region, latitude & year
  group_by(name, region, rndLat, year) %>%
  add_count(name="group_n") %>% ## n. observations per group
  filter(group_n>=10) %>% #only groups with at least 10 observations
  group_by(name, region) %>% #group by "dataset"
  mutate(nlat=length(unique(rndLat))) %>% #count how many distinct latitudinal bands included
  filter(nlat>=3) # need at least 3 latitudinal bands

#visualize some differences
plotcompar<-list()
plotcompar[[1]]<-ggplot(data=fricdata, aes(x=region, y=alt) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Original dataset altitudes")

plotcompar[[2]]<-ggplot(data=reanalysis.data, aes(x=region, y=alt) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Reanalysis dataset altitudes") + ylim(min(fricdata$alt),max(fricdata$alt))

plotcompar[[3]]<-ggplot(data=fricdata, aes(x=region, y=rndLat) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Original dataset latitudes")

plotcompar[[4]]<-ggplot(data=reanalysis.data, aes(x=region, y=rndLat) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Reanalysis dataset latitudes") + ylim(min(fricdata$rndLat), max(fricdata$rndLat))

plotcompar[[5]]<-ggplot(data=filter(fricdata, !is.na(year)), aes(x=region, y=year) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Original dataset years")

plotcompar[[6]]<-ggplot(data=reanalysis.data, aes(x=region, y=year) ) +
  geom_boxplot(outlier.colour="red", outlier.shape=16, outlier.size=2, notch=FALSE) + ggtitle(la
bel="Reanalysis dataset years") + ylim(min(fricdata$year, na.rm=T), max(fricdata$year, na.rm=T))

grid.arrange(grobs=plotcompar[c(1:6)], nrow=3)
```



*

Estimate phenometrics using phest

This chunk of code estimates onset and offset phenometrics by species-region-year-latitudinal_band using curated data.

We use the phest package to estimate onset and offset of flight periods based on occurrence data, when at least 10 observations exist for a species-region-year-latitudinal_band unit. The phest package applies a weibull distribution. Please note that this chunk does take a few minutes to run. Also, warnings are automatically generated by “phest” when a correction is applied to the phenometric estimate. Additionally, “phest” throws a warning for CI estimation. We have explored these warnings and don’t believe that there is any problem continuing with the estimates produced; therefore we have suppressed the warning messages here.

We have added an easy way to select whether to estimate the phenometrics directly or to load them from a saved .RData file. To bypass the weibull estimation, set calc.new.metrics to FALSE. To run the weibull estimation, set calc.new.metrics to TRUE.

```
rm(plotcompar)

#If you want to just load the previously estimated phenometrics, set this to FALSE.
calc.new.metrics<-TRUE
#we'd prefer to use calendar day
reanalysis.data<-reanalysis.data %>%mutate(doy=yday(as.Date(paste(year,month,day, sep="-"),"%Y-%m-%d")))


datasets.ls<-reanalysis.data %>% group_by(name, region) %>% tally()

#For each species & region, calculate phenometrics
if(calc.new.metrics) {
  pheno.est<-data.frame(name=character(0),region=character(0),year=integer(0),rndLat=integer(0),onset.est=numeric(0),onset.low=numeric(0),onset.high=numeric(0),offset.est=numeric(0),offset.low=numeric(0),offset.high=numeric(0))

  for(rowi in 1:nrow(datasets.ls)){ # for each unique dataset
    namei<-datasets.ls$name[rowi]
    regi<-datasets.ls$region[rowi]
    index <- 1 # create/reset an indexer
    pheno.estimates <- list() # create/refresh a blank list per group
    rowi.data<-filter(reanalysis.data, name==namei, region==regi)
    for(yr in unique(rowi.data$year)){ # and each unique year
      for(lat in unique(rowi.data$rndLat)){ # and each unique latitude
        temp <- filter(rowi.data, rndLat==lat, year==yr) # filter the occurrence data for each group

        if(nrow(temp) > 9){ # if there are at least 10 occurrences, then...
          estimates <- c(namei, regi, yr, lat, nrow(temp),
                         suppressWarnings(weib.limit(temp$doy, upper=FALSE, alpha=0.05)),  suppressWarnings(weib.limit(temp$doy, upper=TRUE, alpha=0.05))) # calculate estimates for the group: onset, offset
          pheno.estimates[[index]] <- estimates # shuttle those into a list
          index <- index+1
        } #end if enough occurrences
      } #end lat
    } #end yr
    df <- data.frame(matrix(unlist(pheno.estimates), nrow=length(pheno.estimates), byrow=TRUE),stringsAsFactors=FALSE)
    names(df)<-c("name","region","year","rndLat","n","onset.est","onset.low","onset.high","offset.est","offset.low","offset.high")
    pheno.est<-rbind(pheno.est, df)
  }
  for(coli in 3:11) {
    pheno.est[,coli]<-as.numeric(pheno.est[,coli])
  }

  #Format & store data
  pheno.data<-pheno.est %>%
    mutate(unit=paste(name, rndLat, year,sep="-")) %>%
    select(unit,onset.est,offset.est,name,region,rndLat,year,n) %>%
    mutate(onset=round(onset.est,0),term=round(offset.est,0))
  pheno.data<-na.omit(pheno.data)
  #Weibull estimator doesn't bound so
  #We bounded all onset & termination metrics y [60,330], limiting flight periods to March - Nov
ember
  pheno.data$onset[pheno.data$onset<60]<-60
  pheno.data$term[pheno.data$term>330]<-330

  save(pheno.data, file="data/phenometrics.RData")
} else {
  #If we want to skip phest and phenometric estimation:
  load("data/phenometrics.RData")
}
```

*

Statistical models for phenometrics

This code uses estimated onset and offset phenometrics in linear models to examine phenological patterns with latitude and year. Other statistical models may be more appropriate for a de novo analysis, but here we want our statistical model to parallel the Fric et al. model in intention, but using multiple regression instead of residual regression.

```
datasets<-pheno.data %>%
  group_by(name, region) %>%
  tally()
pheno.data<-na.omit(pheno.data)
fric_FP<-fricdata %>%
  group_by(name,region,rndLat) %>%
  summarize(onset=min(SuccDay),term=max(SuccDay),FP=term-onset)
```

```
## `summarise()` regrouping output by 'name', 'region' (override with `.groups` argument)
```

```
verify.order<-pheno.data %>%
  mutate(FP=term-onset)
summary(verify.order$FP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    6.00   44.00   56.00   62.14   72.00  252.00
```

```
print(paste("Across datasets our estimated flight periods average ", round(mean(verify.order$FP,
na.rm=T))," days, and range from ", min(verify.order$FP, na.rm=T), " days to ",max(verify.order
$FP, na.rm=T), " days. In the original analysis, the average flight period duration was ", round
(mean(fric_FP$FP, na.rm=T)), " days, with a range of ",min(fric_FP$FP, na.rm=T),"-", max(fric_FP
$FP, na.rm=T), " days.",sep=""))
```

```
## [1] "Across datasets our estimated flight periods average 62 days, and range from 6 days to 2
52 days. In the original analysis, the average flight period duration was 50 days, with a range
of 0-359 days."
```

```
rm(verify.order)

#Loop through datasets, run model for phenology by species & region, and store LM parameters
onsetpheno<-list()
termpheno<-list()
onset1<-NULL
term1<-NULL
axes<-NULL

for(rowi in 1:nrow(datasets)) {
  pheno.rowi<-pheno.data %>%
    filter(name==datasets$name[rowi], region==datasets$region[rowi])
#estimate model params for onset
  onset.lm<-summary(lm(onset~rndLat+year, data=pheno.rowi))$coefficients #estimate model params
 for termination
  term.lm<-summary(lm(term~rndLat+year, data=pheno.rowi))$coefficients
#store
  onsetpheno[[rowi]]<-onset.lm
  termpheno[[rowi]]<-term.lm

    #onset
  temponset<-matrix(unlist(onset.lm[c(2:3),]), ncol=4, byrow=F)
  onset1<-rbind(onset1, temponset)
  axes<-c(axes,row.names(onset.lm)[c(2:3)])
#termination
  tempterm<-matrix(unlist(term.lm[c(2:3),]), ncol=4, byrow=F)
  term1<-rbind(term1, tempterm)
  rm(pheno.rowi,onset.lm,term.lm,temponset,tempterm)
  }

#Create results dataframes: onset
onset1<-as.data.frame(onset1)
colnames(onset1)<-c("param.est","param.se","param.t","param.p")
onset1$param<-axes
onset1$metric<-"onset"
onset1$name<-rep(datasets$name, each=2)
onset1$region<-rep(datasets$region, each=2)
onset1$n<-rep(datasets$n, each=2)

#Create results dataframes: termination
term1<-as.data.frame(term1)
colnames(term1)<-c("param.est","param.se","param.t","param.p")
term1$param<-axes
term1$metric<-"termination"
term1$name<-rep(datasets$name, each=2)
term1$region<-rep(datasets$region, each=2)
term1$n<-rep(datasets$n, each=2)


result<-bind_rows(onset1, term1)
result<-result %>%
  mutate(response=ifelse(param.p<0.05,ifelse(param.est>0,1,-1),0))
```

```
#NOT in manuscript but exploratory: Using only coefficients and significance without confidence
 intervals, what phenological patterns are present?
slopediff<-NULL
for(spi in unique(result$name)) {
  d.start<-ifelse(filter(result,param=="rndLat",name==spi,metric=="onset")$response>0,"later","s
ame")
  d.duration<- ifelse(filter(term1,param=="rndLat",name==spi)$param.est-filter(onset1,param=="rn
dLat",name==spi)$param.est<0,"shorter",ifelse(filter(term1,param=="rndLat",name==spi)$param.est-
filter(onset1,param=="rndLat",name==spi)$param.est>0,"longer","same"))
  slopediff<-c(slopediff, paste(d.start,d.duration,sep="."))
}
table(slopediff)
```

```
## slopediff
##  later.longer later.shorter   same.longer  same.shorter 
##             1            13             3             5
```

*

Compare statistical results to Fric et al.

This code uses model outputs and compares them to the results of the Fric et al. analysis. It outputs Figure 2.

```
##Results and visualizations

##Import Fric results:
load("data/Fric_results.RData")
datasets$set<-paste(datasets$name,datasets$region,sep="-")
fric.results$reanalyzed<-0
fric.results$reanalyzed[c(match(datasets$set,fric.results$set),match(datasets$set,fric.results$s
et)+105)]<-1

#Model 1 = Fric Direct regression, all species
fric1<-fric.results %>%
  filter(model=="lat") %>%
  mutate(modelnum=1, modelname='SR-105') %>%
  select(name,region,onset.coef,onset.response,term.coef,term.response,modelnum,modelname)

#Model 3 = Fric Direct regression, reanalyzed species
fric3<-fric.results %>%
  filter(model=="lat", reanalyzed==1) %>%
  mutate(modelnum=3, modelname='SR-22') %>%
  select(name,region,onset.coef,onset.response,term.coef,term.response,modelnum,modelname)

#Model 2 = Fric residual regression, all species
fric2<-fric.results %>%
  filter(model=="corr") %>%
  mutate(modelnum=2, modelname='RR-105') %>%
  select(name,region,onset.coef,onset.response,term.coef,term.response,modelnum,modelname)

#Model 4 = Fric residual regression, reanalyzed species
fric4<-fric.results %>%
  filter(model=="corr", reanalyzed==1) %>%
  mutate(modelnum=4, modelname='RR-22') %>%
  select(name,region,onset.coef,onset.response,term.coef,term.response,modelnum,modelname)

#Model 5 = Reanalysis multiple regression
temp<-pivot_wider(filter(result, param=="rndLat"), id_cols =c(name, region),names_from=metric,va
lues_from=c(param.est,param.p, response) )
print("The reanalysis result table has fields:")
```

```
## [1] "The reanalysis result table has fields:"
```

```
names(result)
```

```
##  [1] "param.est" "param.se"  "param.t"   "param.p"   "param"     "metric"
##  [7] "name"      "region"    "n"         "response"
```

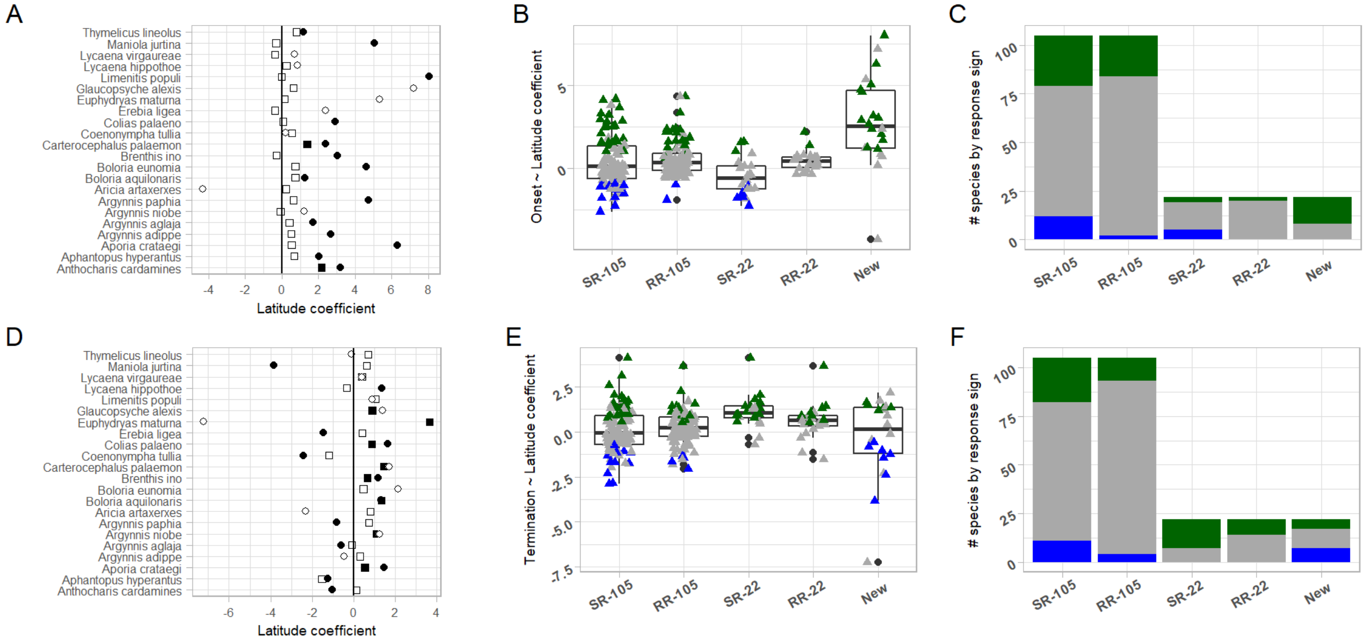
```
print("From which the following fields are created using pivot_wider:")
```

```
## [1] "From which the following fields are created using pivot_wider:"
```

```
names(temp)
```

```
## [1] "name"                  "region"                "param.est_onset"      
## [4] "param.est_termination" "param.p_onset"         "param.p_termination"  
## [7] "response_onset"        "response_termination"
```

```
#Here we select the fields we need and name them to correspond to the Fric result tables
result5<-temp %>%
  select(name, region, onset.coef=param.est_onset, onset.response=response_onset, term.coef=param.est_termination, term.response=response_termination) %>%
  mutate(modelnum=5, modelname="New")
rm(temp)

#Combine all results into 1 data frame
result.compar<-as.data.frame(rbind(fric1,fric2,fric3,fric4,result5))
#result.compar$modelnum<-as.factor(result.compar$modelnum)
result.compar$s1<-1
##Create Figure 2
colorscheme<-c("blue", "darkgray", "darkgreen")
ts<-8
ar=2/3
ar1=1
#modelnames<-c("SR105","RR105","SR22","RR22","New")
#Panels A, D: compare coefficients
#Panel A: Onset coefficients
onset.sp<-ggplot(data=filter(result.compar, as.numeric(modelnum)>3), aes(x=name, y=onset.coef, shape=as.factor(modelnum), fill=as.factor(onset.response))) +
  geom_point(color="black") +
  scale_shape_manual(values=c(22,21)) +
  scale_fill_manual(values=c("white","black")) +
  geom_hline(yintercept=0) +
  scale_y_continuous(breaks=seq(-8,8,2)) +
  labs(x="", y="Latitude coefficient") + coord_flip() +
  theme_light()  +
  theme(legend.position = "none", axis.title=element_text(size=ts-1),  axis.text=element_text(size=ts-2), aspect.ratio=ar1, plot.margin = margin(2, 2, 2, 2, unit = "pt"))
#onset.sp
#Panel D: Termination coefficients
term.sp<-ggplot(data=filter(result.compar, as.numeric(modelnum)>3), aes(x=name, y=term.coef, shape=as.factor(modelnum), fill=as.factor(term.response))) +
  geom_point(color="black") +
  scale_shape_manual(values=c(22,21)) +
  scale_fill_manual(values=c("black","white","black")) +
  geom_hline(yintercept=0) +
  scale_y_continuous(breaks=seq(-8,8,2)) +
  labs(x="", y="Latitude coefficient") + coord_flip() +
  theme_light()  +
  theme(legend.position = "none", axis.title=element_text(size=ts-1),  axis.text=element_text(size=ts-2), aspect.ratio=ar1, plot.margin = margin(2, 2, 2, 2, unit = "pt"))
#term.sp

#Panels B, E: response boxplots
#Pandel B: Onset
onset.c<-ggplot(data=result.compar, aes(x=reorder(modelname,modelnum), y=onset.coef)) +
  geom_boxplot(aes(group=reorder(modelname,modelnum))) +
  geom_jitter(data=filter(result.compar),  aes(x=reorder(modelname,modelnum), y=onset.coef, color=as.factor(onset.response)), width=0.2, height=0, shape=17) +
  labs(x="", y="Onset ~ Latitude coefficient") +
  scale_color_manual(values=colorscheme) +
  theme_light() +
  theme(legend.position = "none", axis.title=element_text(size=ts-1, face="plain"), axis.text=element_text(size=ts-1, angle=30, hjust=0.8, face="bold"), aspect.ratio=ar, plot.margin = margin(2, 2, 2, 2, unit = "pt"))
#onset.c
#Panel E: termination
term.c<-ggplot(data=result.compar, aes(x=reorder(modelname,modelnum), y=term.coef)) +
  geom_boxplot(aes(group=reorder(modelname,modelnum))) +
  geom_jitter(data=filter(result.compar),  aes(x=reorder(modelname,modelnum), y=term.coef, color=as.factor(term.response)), width=0.2, height=0, shape=17) +
  labs(x="", y="Termination ~ Latitude coefficient") +
  scale_color_manual(values=colorscheme) +
  theme_light() +
  theme(legend.position = "none", axis.title=element_text(size=ts-1, face="plain"), axis.text=element_text(size=ts-1, angle=30, hjust=0.8, face="bold"), aspect.ratio=ar, plot.margin = margin(2, 2, 2, 2, unit = "pt"))
#Panels C, F: stacked barplots
#Panel c: Onset responses
onset.st<-ggplot(data=result.compar, aes(x=(reorder(modelname,modelnum)), y=s1, fill=as.factor(onset.response))) +
  geom_bar(position=position_stack(reverse=T), stat="identity")  +
  scale_fill_manual(values=colorscheme) +
  labs(x="", y="# species by response sign") + theme_light() +
  theme(legend.position = "none", axis.title=element_text(size=ts-1, face="plain"), axis.text=element_text(size=ts-1, angle=30, hjust=0.8, face="bold"), aspect.ratio=ar, plot.margin = margin(2, 2, 2, 2, unit = "pt"))

#Panel F: Termination responses
term.st<-ggplot(data=result.compar, aes(x=reorder(modelname,modelnum), y=s1, fill=as.factor(term.response))) +
  geom_bar(position=position_stack(reverse=T), stat="identity") +
  scale_fill_manual(values=colorscheme) +
  theme_light() + labs(x="", y="# species by response sign") +
  theme(legend.position = "none", axis.title=element_text(size=ts-1, face="plain"), axis.text=element_text(size=ts-1, angle=30, hjust=0.8, face="bold"), aspect.ratio=ar, plot.margin = margin(2, 2, 2, 2, unit = "pt"))

#term.st


##Combine panels into Figure 2:
p1<-onset.sp+labs(tag="A")
p2<-onset.c+labs(tag="B")
p3<-onset.st+labs(tag="C")
p4<-term.sp+labs(tag="D")
p5<-term.c+labs(tag="E")
p6<-term.st+labs(tag="F")

#pdf_filename<-("output/LarsenShirey2020_Fig2.pdf")
grid.arrange(ncol=3, grobs=list(p1, p2, p3, p4, p5, p6),  widths=c(1.2,1,1), bottom="These figures show the difference between the results of our reanalysis ('New') and Fric et al.'s \n results (SR=Single Regression, RR=Regression of Residuals; 105 = all 105 datasets, 22 =  reanalyzed datasets).")
```



These figures show the difference between the results of our reanalysis ('New') and Fric et al.'s results (SR=Single Regression, RR=Regression of Residuals; 105 = all 105 datasets, 22 = reanalyzed datasets).

```
#fig2<-grid.arrange(ncol=3, grobs=list(p1, p2, p3, p4, p5, p6), top="\n\n", bottom="\n\n", left
="\n\n", right="\n\n", width=10, height=5)
#ggsave(pdf_filename, arrangeGrob(fig2, nrow=1), width=10, height=5, scale=1.5, dpi=600,units="i
n")
```

*

Create statistics for results table (Supplemental Table 2)

This code outputs a results table that is a partial Supplemental Table 2 - it is currently missing the ‘year’ analyses from Fric et al., as our focus is on the latitudinal patterns.

```
#fric.table<-pivot_wider(filter(result, reanalyzed==1), id_cols =c(name, region,),names_from=metric,values_from=c(param.est,param.p, response) )

#Here we are building supplemental table 2 with fields: name_resultsfile, region, phenometric, indep.variable, Fric_singleRegression_Sign, Fric_resid.regress_sign, Reanalysis_sign, Reanalysis_p, Reanalysis_coefficient, Fric_resid.regress_p, Fric_resid.regress_coefficient, Fric_singleRegression_p, Fric_singleRegression_coefficient

table2<-result %>%
  select(name_resultsfile=name, region, phenometric=metric, indep.variable=param, Reanalysis_sign=response,Reanalysis_p=param.p,Reanalysis_coefficient=param.est)
table2$indep.variable[table2$indep.variable=="rndLat"]<-"latitude"

#onset SR
fric.table2a<-fric.results %>%
  filter(reanalyzed==1, model=="lat") %>%
  select(name_resultsfile=name, region, Fric_SR_Sign=onset.response, Fric_SR_p=onset.p_mean, Fric_SR_coef=onset.coef) %>%
  mutate(phenometric="onset", indep.variable="latitude")

#term SR
fric.table2b<-fric.results %>%
  filter(reanalyzed==1, model=="lat") %>%
  select(name_resultsfile=name, region, Fric_SR_Sign=term.response, Fric_SR_p=term.p_mean, Fric_SR_coef=term.coef) %>%
  mutate(phenometric="termination", indep.variable="latitude")

fric.table2<-rbind(fric.table2a,fric.table2b)

table2<-merge(table2,fric.table2,by=intersect(names(table2),names(fric.table2)), all.x=T, all.y=T)

#onset RR
fric.table2a<-fric.results %>%
  filter(reanalyzed==1, model=="corr") %>%
  select(name_resultsfile=name, region, Fric_RR_Sign=onset.response, Fric_RR_p=onset.p_mean, Fric_RR_coef=onset.coef) %>%
  mutate(phenometric="onset", indep.variable="latitude")
#term RR
fric.table2b<-fric.results %>%
  filter(reanalyzed==1, model=="corr") %>%
  select(name_resultsfile=name, region, Fric_RR_Sign=term.response, Fric_RR_p=term.p_mean, Fric_RR_coef=term.coef) %>%
  mutate(phenometric="termination", indep.variable="latitude")

fric.table2<-rbind(fric.table2a,fric.table2b)

table2<-merge(table2,fric.table2,by=intersect(names(table2),names(fric.table2)), all.x=T)

##This partial supplementary table 2 does not include the year results from Fric.
#write.csv(table2,file="outputs/supp_table2_part.csv")
```

This is the end of this analysis. Code for Supplemental Figure 1 has been moved to a separate Rmarkdown file.