Fric et al. critiques: data curation

Elise Larsen & Vaughn Shirey

Updated 7-Dec-2020; Begun Feb-2020

Here we explore the occurrence data from Fric et al. (2020)

This gives a detailed account of some data curation issues we observed in the Fric et al. data and curation. This file inputs the data/occurrence.RData and fric_supplements/ele13419-suo-0003-tables2.xlsx files and outputs the data/occurrences_FricAnalysis.RData

```
rm(list=ls())
# load libraries
library(tidyverse)
library(readxl)
library(ggplot2)
library(ggExtra)
library(gridExtra)
# install.packages("viridis")
library(viridis)
```

```
## Warning: package 'viridis' was built under R version 4.0.3
```

Data Input

We import the formatted occurrence data and explore the independent variables used in the Fric et al. analysis.

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

#Revisit list of names from results file to limit data to that used by Fric et al.
#Which of these names shows up in the results?
result.names<-unique(na.omit(read_excel("fric_supplements/ele13419-sup-0003-tables2.xlsx", sheet="~latitude", range="A3:A11
3"))$Species)
resultnames<-(strsplit(result.names, " "))
result.names<-tibble(name=character(),genus=character(),spep=character())
for(i in 1:length(resultnames)) {
  genus<-paste(resultnames[[i]][1])
  spep<-paste(resultnames[[i]][2])
  name<-paste(genus,spep,sep=" ")
  temp.names<-tibble(name=as.character(name),genus=as.character(genus),spep=as.character(spep))
  result.names<-bind_rows(result.names,temp.names)
}
rm(resultnames, genus, spep, name, temp.names)

#Fric et al also removed all 1st of month observations according to their methods
fricdata<-filter(alldata, day!=1, name %in% result.names$name)
summary(fricdata)
```

```
##    row.index         name           decimalLongitude   decimalLatitude
##  Min.   :    1   Length:257972      Min.   :-162.559   Min.   : 5.787
##  1st Qu.: 2341   Class :character   1st Qu.:  -2.676   1st Qu.:52.711
##  Median : 7274   Mode  :character   Median :   9.551   Median :55.638
##  Mean   :15624                      Mean   :   6.529   Mean   :56.296
##  3rd Qu.:22563                      3rd Qu.:  23.672   3rd Qu.:60.649
##  Max.   :85273                      Max.   :  59.333   Max.   :71.216
##
##       year          month           country               day
##  Min.   :1616   Min.   : 1.000   Length:257972      Min.   : 2.00
##  1st Qu.:1992   1st Qu.: 6.000   Class :character   1st Qu.: 9.00
##  Median :2002   Median : 7.000   Mode  :character   Median :16.00
##  Mean   :1996   Mean   : 6.519                      Mean   :16.19
##  3rd Qu.:2009   3rd Qu.: 7.000                      3rd Qu.:24.00
##  Max.   :2015   Max.   :12.000                      Max.   :31.00
##  NA's   :53
##     SuccDay          rndLat           alt              region
##  Min.   :  2.0   Min.   : 6.00   Min.   :-2666.74   Length:257972
##  1st Qu.:165.0   1st Qu.:53.00   1st Qu.:   23.25   Class :character
##  Median :187.0   Median :56.00   Median :   64.24   Mode  :character
##  Mean   :181.8   Mean   :56.23   Mean   :  114.26
##  3rd Qu.:202.0   3rd Qu.:61.00   3rd Qu.:  109.48
##  Max.   :361.0   Max.   :71.00   Max.   : 4305.17
##
##       doy
##  Min.   :  2
##  1st Qu.:166
##  Median :188
##  Mean   :183
##  3rd Qu.:203
##  Max.   :365
##  NA's   :53
```

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

Data exploration: altitude (elevation)

(We defer to the Fric et al use of “altitude” for clarity)

Early on in data exploration we were concerned with the range of altitude values in the data. One aspect of our data exploration for altitude involved examining outliers and spot-checking specific occurrence records in GBIF, which were either below 0m or in the top quartile of altitudes. Looking at these records led us to understand that

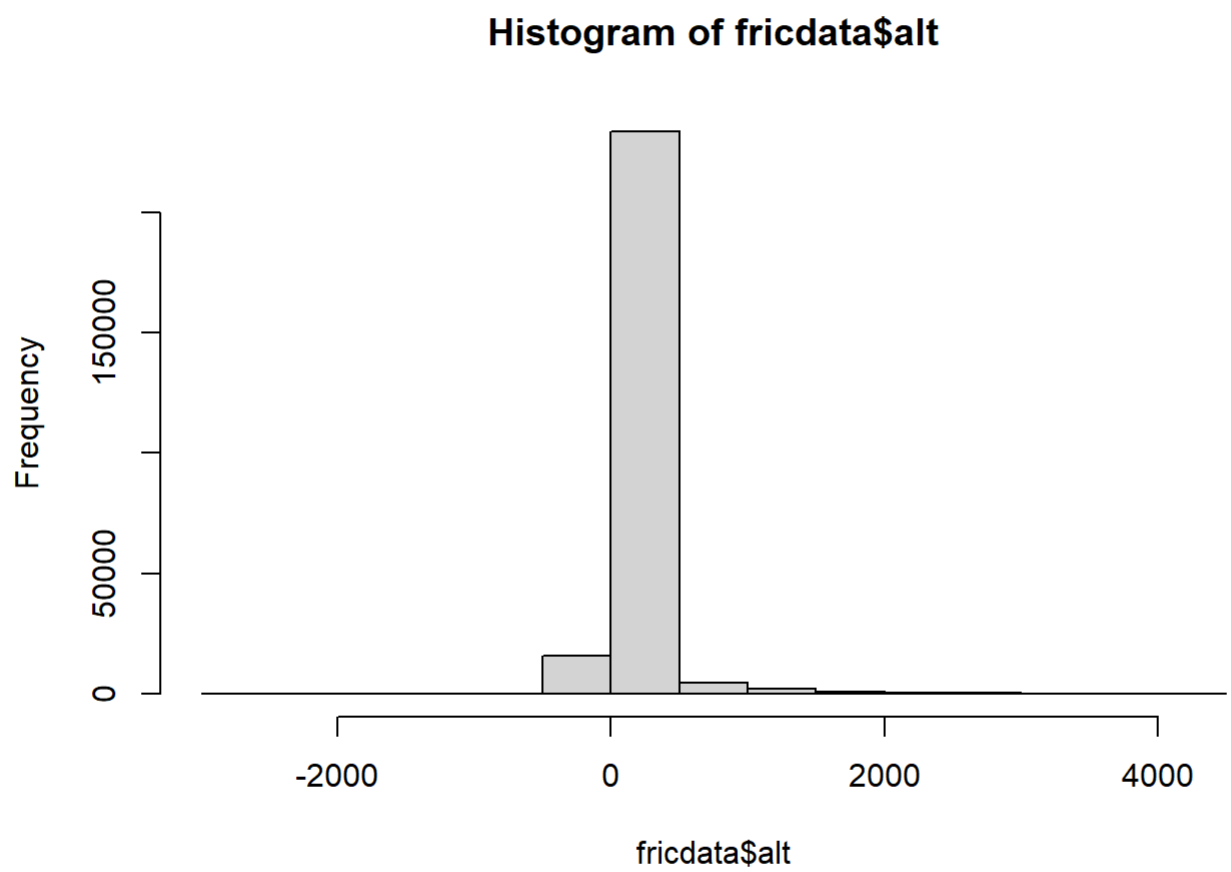
- 1. GIS coordinates had often been assigned by placename, or were otherwise inaccurate, and
  2. 2. altitudes obtained by using the Google API to extract altitude for coordinates did not provide reliable altitudes for the underlying occurrences.

Here we examine broad patterns and specific outlier cases.

```
#basic range & frequency in data
summary(fricdata$alt)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -2666.74    23.25    64.24   114.26   109.48  4305.17
```

```
hist(fricdata$alt)
```




```
#how many records below 0?
print(paste(nrow(filter(fricdata,alt<0)),"records below sea level represent", round(nrow(filter(fricdata,alt<0))/nrow(fricda
ta)*100,2),"percent of all ocurrence records. We examined lat/long for many of these records and all examined locations were
in bodies of water.",sep=" "))
```

```
## [1] "9974 records below sea level represent 3.87 percent of all ocurrence records. We examined lat/long for many of these
records and all examined locations were in bodies of water."
```

```
#how many records are above 500m?
print(paste(nrow(filter(fricdata,alt>500)),"records above 500m represent", round(nrow(filter(fricdata,alt>500))/nrow(fricdat
a)*100,2),"percent of all ocurrence records. We examined lat/long and location for a small subset of high altitude records a
nd found vague place names had been used for geolocation.",sep=" "))
```

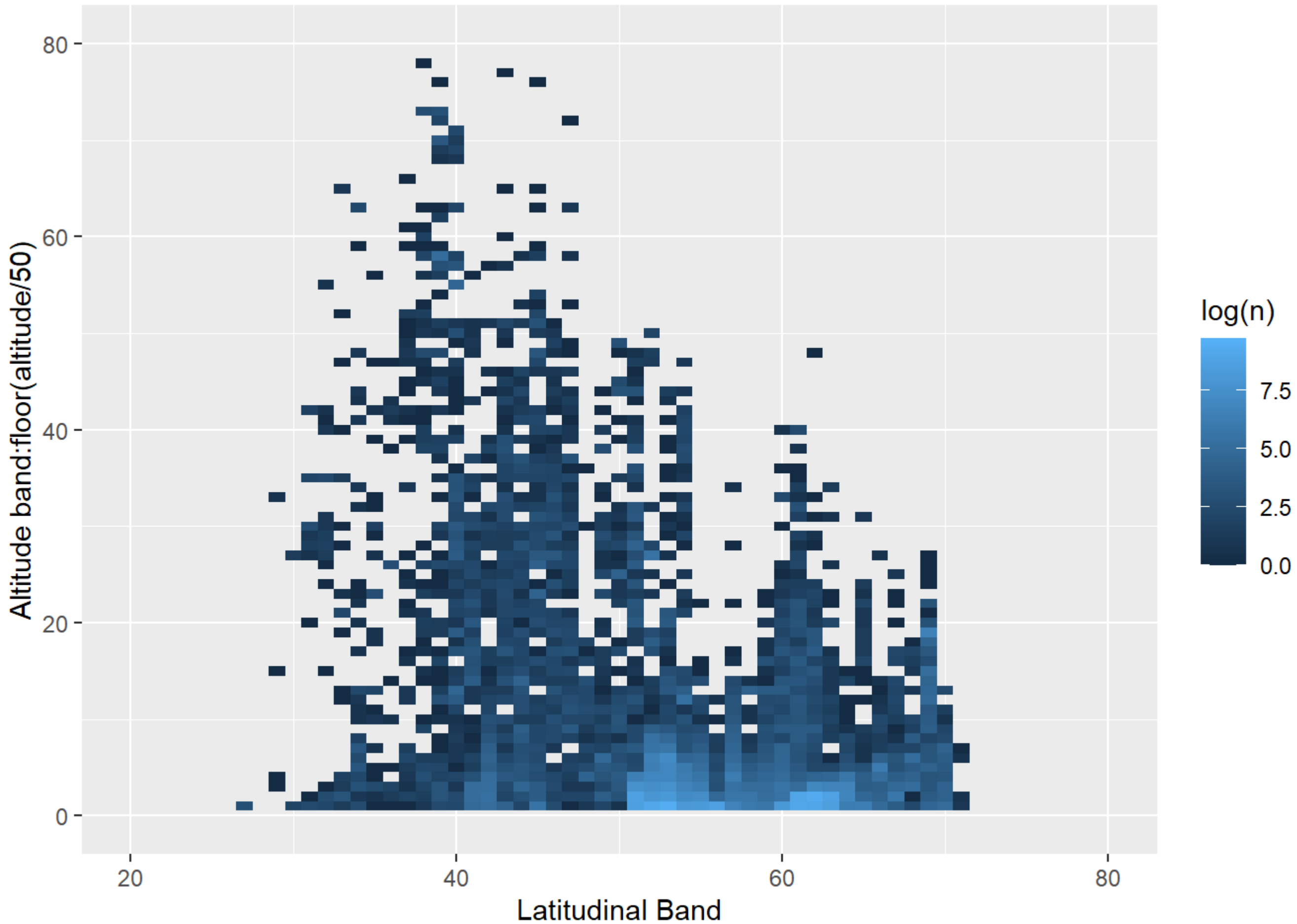
```
## [1] "8629 records above 500m represent 3.34 percent of all ocurrence records. We examined lat/long and location for a sma
ll subset of high altitude records and found vague place names had been used for geolocation."
```

```
#How many in the 0-500m range
print(paste(nrow(filter(fricdata,between(alt,0,500))),"records  within 0-500m represent", round(nrow(filter(fricdata,between
(alt,0,500)))/nrow(fricdata)*100,2),"percent of all ocurrence records. For reanalysis, we can constrain data to these record
s with minimal impact on data density. ",sep=" "))
```

```
## [1] "239369 records  within 0-500m represent 92.79 percent of all ocurrence records. For reanalysis, we can constrain dat
a to these records with minimal impact on data density. "
```

```
altdata<-fricdata %>% mutate(alt.grp=floor(alt/50))  %>%
  group_by(alt.grp, rndLat) %>% tally()
# Heatmap
ggplot(altdata, aes(rndLat, alt.grp, fill= log(n))) +
  geom_tile() + labs(x="Latitudinal Band", y="Altitude band:floor(altitude/50)") +
  xlim(20,80) + ylim(0,80)
```

```
## Warning: Removed 37 rows containing missing values (geom_tile).
```



Outliers appear to be a problem with altitude. Reviewing GBIF records, this appears to be primarily due to the assumption by Fric et al. that the GIS coordinates are precise and that the google API would provide accurate and reliable altitude metrics. Based on the records we spot-checked, when GBIF includes elevation, the values do not match those used in the analysis.

A few examples including the lowest and highest alt records, as well as some additional records selected arbitrarily from the extreme quantiles of altitude:

- 1953 Anthocharis sara record (row.index 166; altitude -525.96m) is from https://www.gbif.org/occurrence/1039154960 (https://www.gbif.org/occurrence/1039154960); geocoordinates were assigned via vertnet in 2015. These coordinates are located in the ocean. The GBIF record traces to https://collections.peabody.yale.edu/search/Record/YPM-ENT-729028 (https://collections.peabody.yale.edu/search/Record/YPM-ENT-729028) which simply gives a locality of “North America; USA; California; Los Angeles County; Rolling Hills”. Rolling Hills, CA is ~10km east of the given lat/long according to our estimation using googlemaps.
- 1991 Parnassius smintheus record (row.index 38; altitude 4048m) is from https://www.gbif.org/occurrence/1039027733 (https://www.gbif.org/occurrence/1039027733) (which gives elevation of 3810m). The GBIF record traces to

https://collections.peabody.yale.edu/search/Record/YPM-ENT-430824 (https://collections.peabody.yale.edu/search/Record/YPM-ENT-430824) which gives a locality of “North America; USA; Colorado; Summit County; Loveland Pass, 3810 m”. The actual collection altitude is provided by the source, and is different than that used in the analysis.

- 1918 Euphydryas chalcedona record (row.index 139; altitude 4305m) is the highest record in the data. It’s from https://www.gbif.org/occurrence/1039181223 (https://www.gbif.org/occurrence/1039181223). The GBIF record traces to https://collections.peabody.yale.edu/search/Record/YPM-ENT-819202 (https://collections.peabody.yale.edu/search/Record/YPM-ENT-819202) which gives a locality of “North America; USA; California; Siskiyou County; Mount Shasta” There is a city named Mount Shasta, CA that incorporated in 1905 that is at elevation 1100m and the peak of Mount Shasta is 4320. It is unclear whether the locality refers to the mountain or to the city; either way it is unlikely that an altitude so close to the peak of the mountain is the best choice for this specimen.

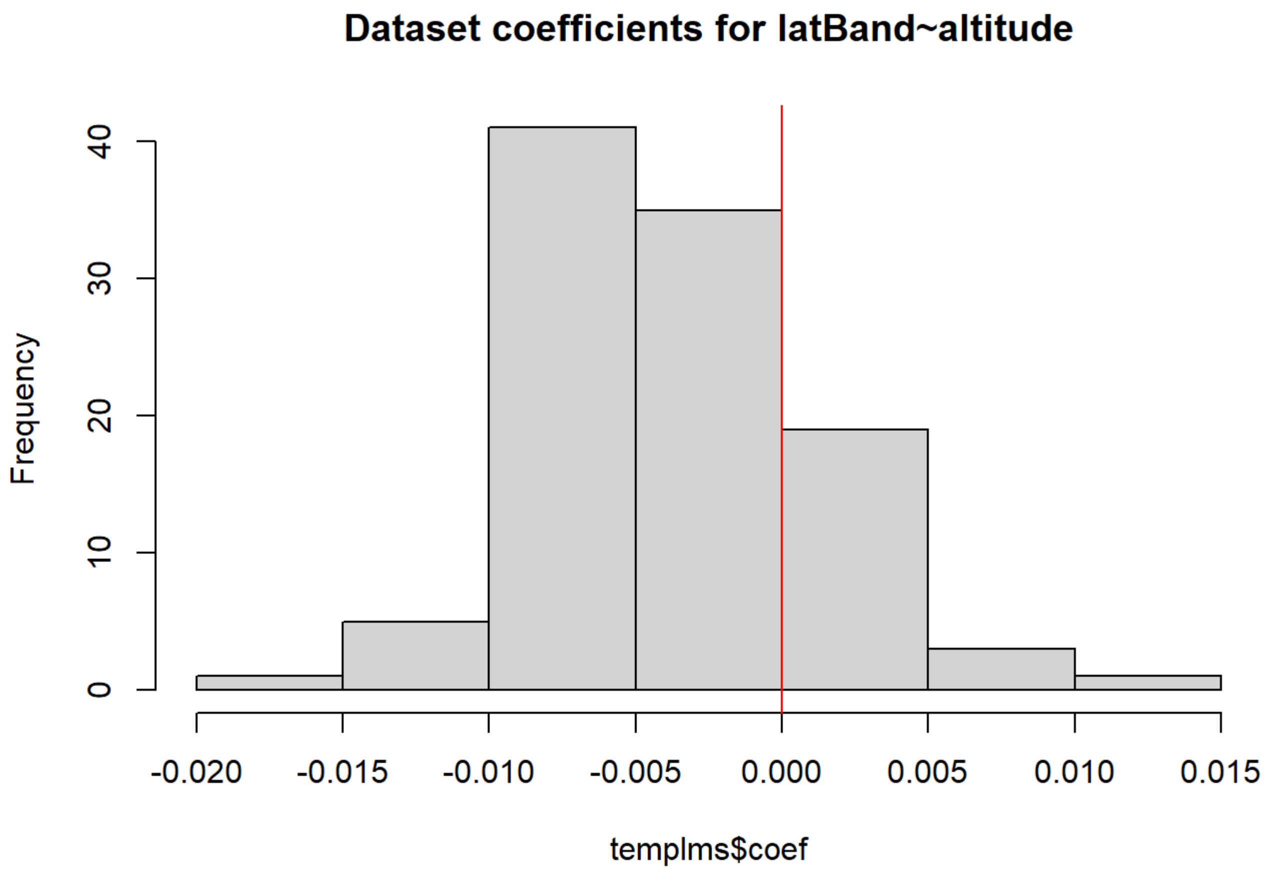
So far those examples are all North America - does this problem exist in Europe too?

- A Lycaena hippothoe record from 1995 (row.index 2160; altitude 3274m) is from https://www.gbif.org/occurrence/2570253925 (https://www.gbif.org/occurrence/2570253925) which lists an inferred elevation of 2000m.
- A Lycaena virgaureae record from 2002 (row.index 4501; altitude -85.8m) appears to match https://www.gbif.org/occurrence/173651704 (https://www.gbif.org/occurrence/173651704) which is located in the Gulf of Bothnia, though GBIF assigns an elevation of 0m. Considering the lat/long are (65,23) most likely those coordinates are imprecise.

Altitude ~ Latitude collinearity

Fric et al. used regression of residuals for corrected analyses. Regression of residuals is not recommended, particularly if there could be collinearity among explanatory variables. We examined the collinearity by modeling rndLat ~ altitude and rndLat~ year, where rndLat represents the latitudinal bands used in analysis. The observed collinearity indicates that regression of residuals analyses would produce biased parameter estimates.

```
#Additional issues with altitude
#Given the use of regression of residuals, we were concerned that collinearity among independent variables could have led to
biased results.

#How many datasets have significant collinearity between altitude and latitude?
templms<-NULL
datasets<-fricdata %>% group_by(name, region)  %>% tally()
for (spi in 1:nrow(datasets)) {
  tempdata<-fricdata %>% filter(name==datasets$name[spi],region==datasets$region[spi])
  spilm<-summary(lm(rndLat~alt, data=tempdata))
  templms<-rbind(templms,c(nrow(tempdata), spilm$coefficients[2,1],  spilm$coefficients[2,4], spilm$r.squared))
}
templms<-as.data.frame(templms)
names(templms)<-c("n","coef","pval","r2")
hist(templms$coef, main="Dataset coefficients for latBand~altitude")
abline(v=0,col="red")
```




```
summary(templms)
```

```
##        n              coef                pval               r2           
##  Min.   :   15   Min.   :-0.019376   Min.   :0.00000   Min.   :0.0000222  
##  1st Qu.:   78   1st Qu.:-0.006861   1st Qu.:0.00000   1st Qu.:0.0280076  
##  Median :  189   Median :-0.004516   Median :0.00000   Median :0.1909175  
##  Mean   : 2457   Mean   :-0.003832   Mean   :0.06654   Mean   :0.2824787  
##  3rd Qu.: 1067   3rd Qu.:-0.001088   3rd Qu.:0.00851   3rd Qu.:0.5261002  
##  Max.   :51819   Max.   : 0.014635   Max.   :0.86050   Max.   :0.8487862
```

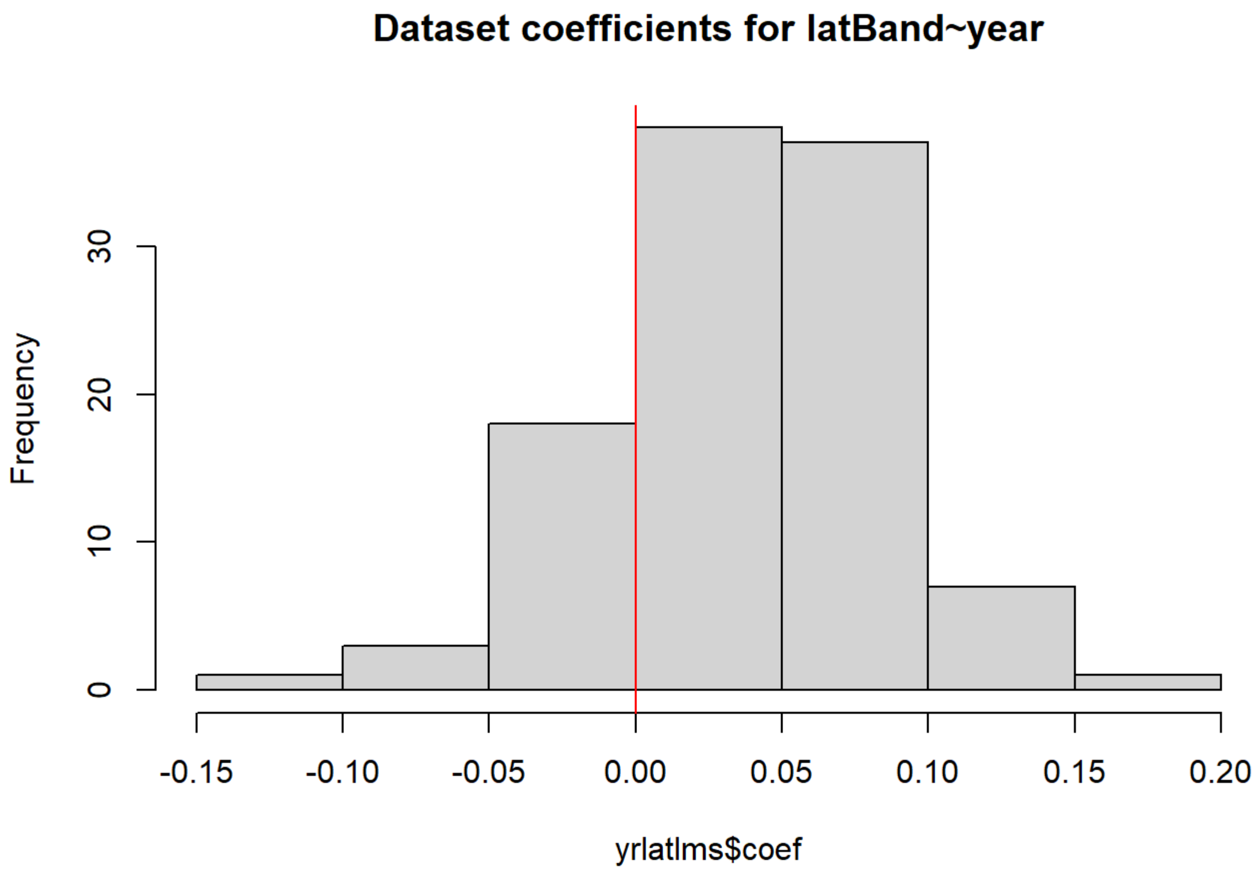
```
round(nrow(filter(templms,pval<0.05))/nrow(templms),2)
```

```
## [1] 0.85
```

```
#How many datasets have significant collinearity
print(paste(nrow(filter(templms,pval<0.05)),"datasets have significant collinearity, representing", round(nrow(filter(templm
s,pval<0.05))/nrow(templms)*100,1),"percent of all datasets. For datasets with significant collinearity, the mean coefficien
t is",round(mean(templms$coef[templms$pval<0.05]),3),"(which translates to a slope of", round(1/mean(templms$coef[templms$pv
al<0.05]),0),"meters per degree latitude) and mean r-squared is",round(mean(templms$r2[templms$pval<0.05]),3)," - therefore
 regression of residuals is likely producing bias parameters.",sep=" "))
```

```
## [1] "89 datasets have significant collinearity, representing 84.8 percent of all datasets. For datasets with significant
collinearity, the mean coefficient is -0.004 (which translates to a slope of -224 meters per degree latitude) and mean r-squ
ared is 0.33  - therefore regression of residuals is likely producing bias parameters."
```

```
#How many datasets have significant collinearity between year and latitude?
yrlatlms<-NULL
for (spi in 1:nrow(datasets)) {
  tempdata<-fricdata %>% filter(name==datasets$name[spi],region==datasets$region[spi])
  spilm<-summary(lm(rndLat~year, data=tempdata))
  yrlatlms<-rbind(yrlatlms,c(nrow(tempdata), spilm$coefficients[2,1],  spilm$coefficients[2,4], spilm$r.squared))
}
yrlatlms<-as.data.frame(yrlatlms)
names(yrlatlms)<-c("n","coef","pval","r2")
hist(yrlatlms$coef, main="Dataset coefficients for latBand~year")
abline(v=0,col="red")
```




```
summary(yrlatlms)
```

```
##       n              coef                pval               r2
##  Min.   :   15   Min.   :-0.105938   Min.   :0.000000   Min.   :0.0000001
##  1st Qu.:   78   1st Qu.: 0.008515   1st Qu.:0.000000   1st Qu.:0.0132368
##  Median :  189   Median : 0.040297   Median :0.004441   Median :0.0507987
##  Mean   : 2457   Mean   : 0.039126   Mean   :0.142946   Mean   :0.0907969
##  3rd Qu.: 1067   3rd Qu.: 0.074053   3rd Qu.:0.123499   3rd Qu.:0.1258460
##  Max.   :51819   Max.   : 0.179087   Max.   :0.992704   Max.   :0.6066502
```

```
round(nrow(filter(yrlatlms,pval<0.05))/nrow(yrlatlms),2)
```

```
## [1] 0.62
```

```
#How many datasets have significant collinearity
print(paste(nrow(filter(yrlatlms,pval<0.05)),"datasets have significant collinearity, representing", round(nrow(filter(yrlat
lms,pval<0.05))/nrow(yrlatlms)*100,1),"percent of all datasets. For datasets with significant collinearity, the mean coeffic
ient is",round(mean(yrlatlms$coef[yrlatlms$pval<0.05]),3),"and mean r-squared is",round(mean(yrlatlms$r2[yrlatlms$pval<0.05
]),3),".",sep=" "))
```

```
## [1] "65 datasets have significant collinearity, representing 61.9 percent of all datasets. For datasets with significant
collinearity, the mean coefficient is 0.058 and mean r-squared is 0.135 ."
```

Data exploration: data density

- In Fric et al. (2020), datasets were analysed with as few as 15 ocurrence records.
- We examine the prevalence of singleton ocurrences, when just one ocurrence was available in a latitudinal band.

```
lat.summary1<-fricdata %>%
  group_by(name, region, rndLat) %>%
  summarize(lat.samplesize=n(),singleton=ifelse(lat.samplesize==1,1,0),dur=max(SuccDay)-min(SuccDay))
```

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

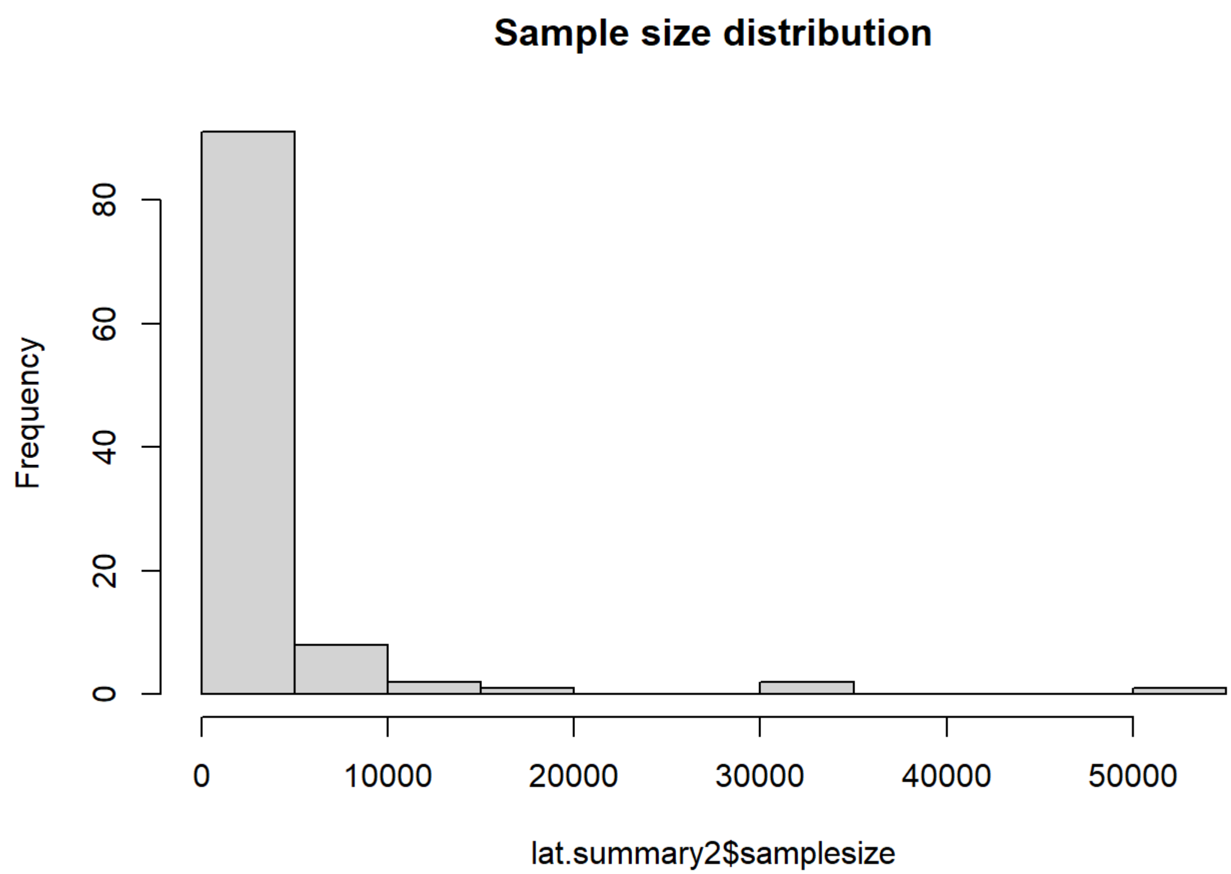
```
lat.summary2<-lat.summary1 %>%
  group_by(name,region) %>%
  summarize(samplesize=sum(lat.samplesize),latspan=max(rndLat)-min(rndLat),nlats=length(unique(rndLat)),n.singletons=sum(sin
gleton),prop.singletons=n.singletons/nlats)
```

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

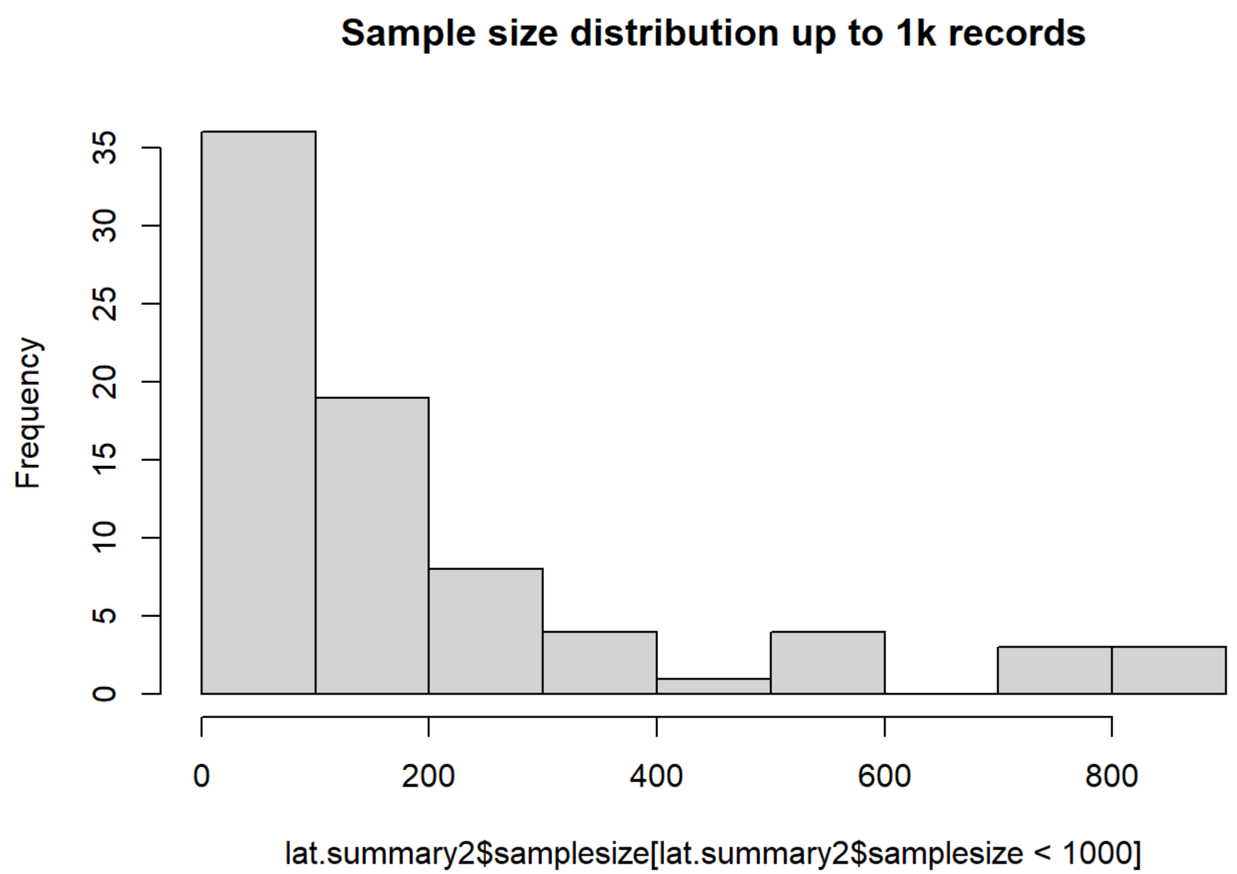
```
summary(lat.summary2)
```

```
##      name              region           samplesize      latspan    
##  Length:105         Length:105         Min.   :   15   Min.   :10.0  
##  Class :character   Class :character   1st Qu.:   78   1st Qu.:24.0  
##  Mode  :character   Mode  :character   Median :  189   Median :27.0  
##                                        Mean   : 2457   Mean   :26.3  
##                                        3rd Qu.: 1067   3rd Qu.:30.0  
##                                        Max.   :51819   Max.   :64.0  
##      nlats       n.singletons    prop.singletons  
##  Min.   : 5.0   Min.   : 0.000   Min.   :0.00000  
##  1st Qu.:13.0   1st Qu.: 2.000   1st Qu.:0.09375  
##  Median :18.0   Median : 3.000   Median :0.19048  
##  Mean   :18.9   Mean   : 3.429   Mean   :0.20831  
##  3rd Qu.:25.0   3rd Qu.: 5.000   3rd Qu.:0.33333  
##  Max.   :33.0   Max.   :10.000   Max.   :0.60000
```

```
#Visualize range of sample sizes
hist(lat.summary2$samplesize, main="Sample size distribution")
```




```
#look at the lower end of sample sizes, where most datasets are
hist(lat.summary2$samplesize[lat.summary2$samplesize<1000], main="Sample size distribution up to 1k records")
```



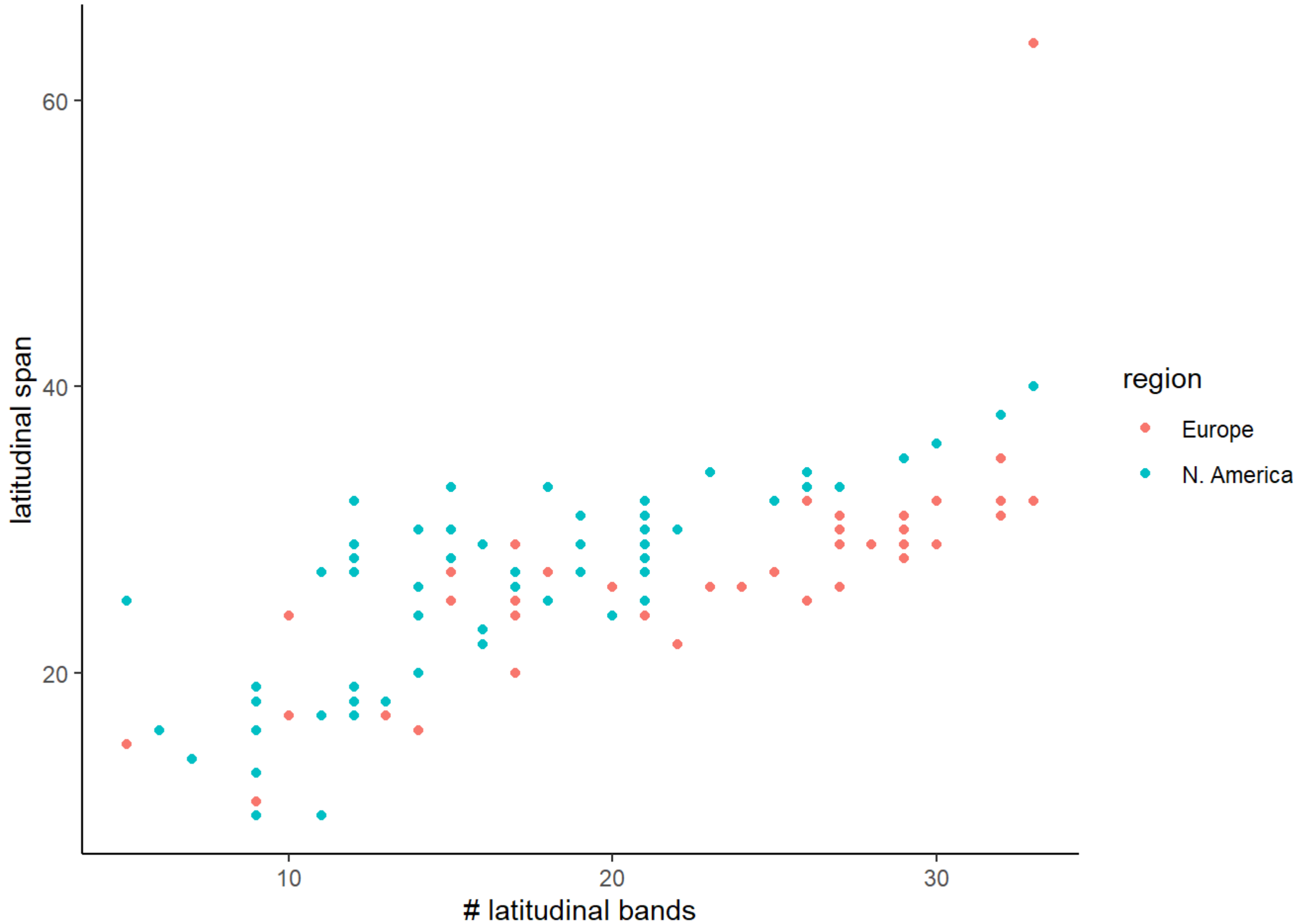

```
nrow(lat.summary2 %>% filter(samplesize<100))
```

```
## [1] 36
```

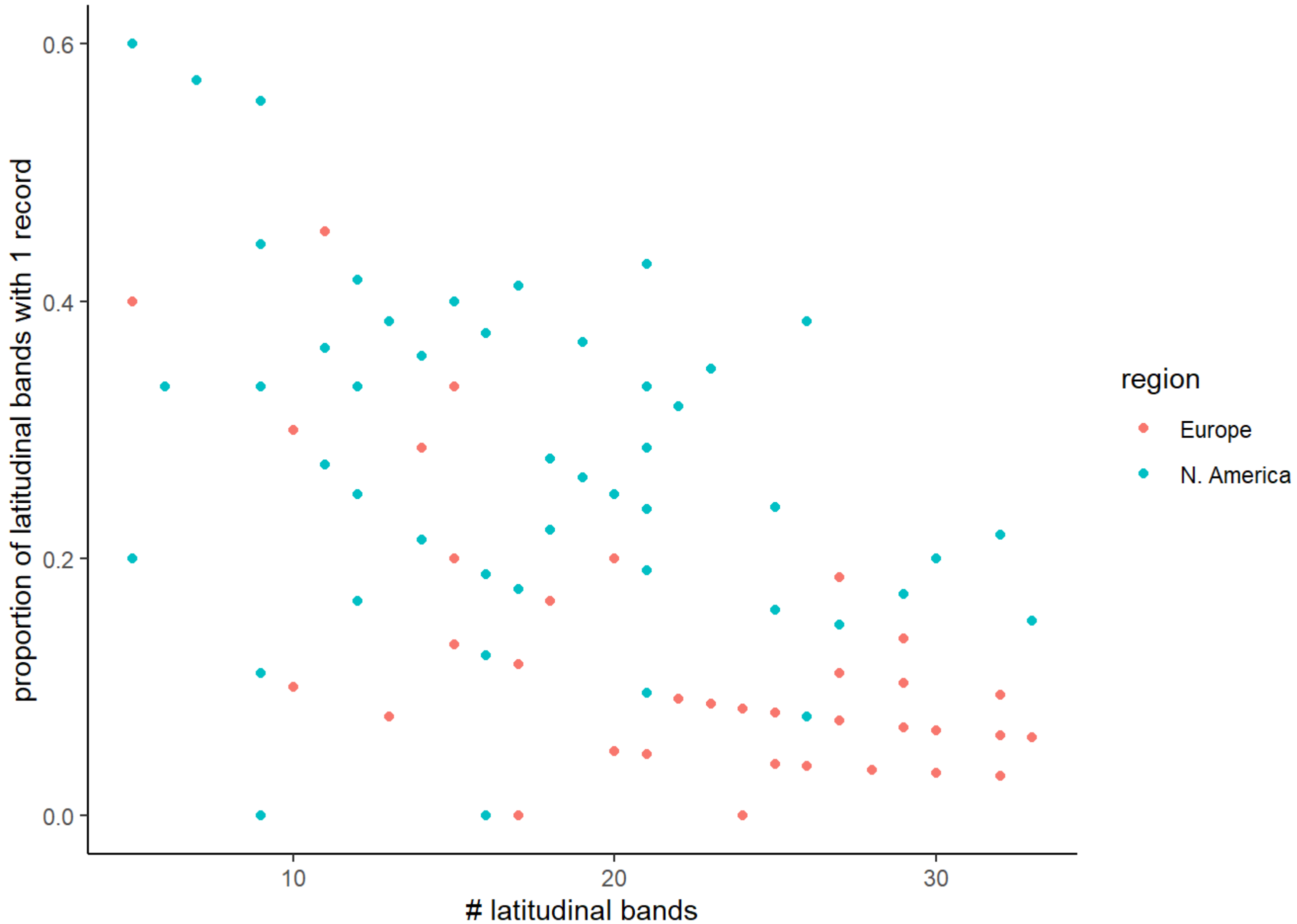
```
print(paste(nrow(lat.summary2 %>% filter(samplesize<100)),"datasets have less than 100 ocurrence records."))
```

```
## [1] "36 datasets have less than 100 ocurrence records."
```

```
ggplot(data=lat.summary2, aes(x=nlats, y=latspan, color=region)) + geom_point() + theme_classic() +
  labs(x="# latitudinal bands", y="latitudinal span")
```




```
ggplot(data=lat.summary2, aes(x=nlats, y=prop.singletons, color=region)) + geom_point() + theme_classic() +
  labs(x="# latitudinal bands", y="proportion of latitudinal bands with 1 record")
```

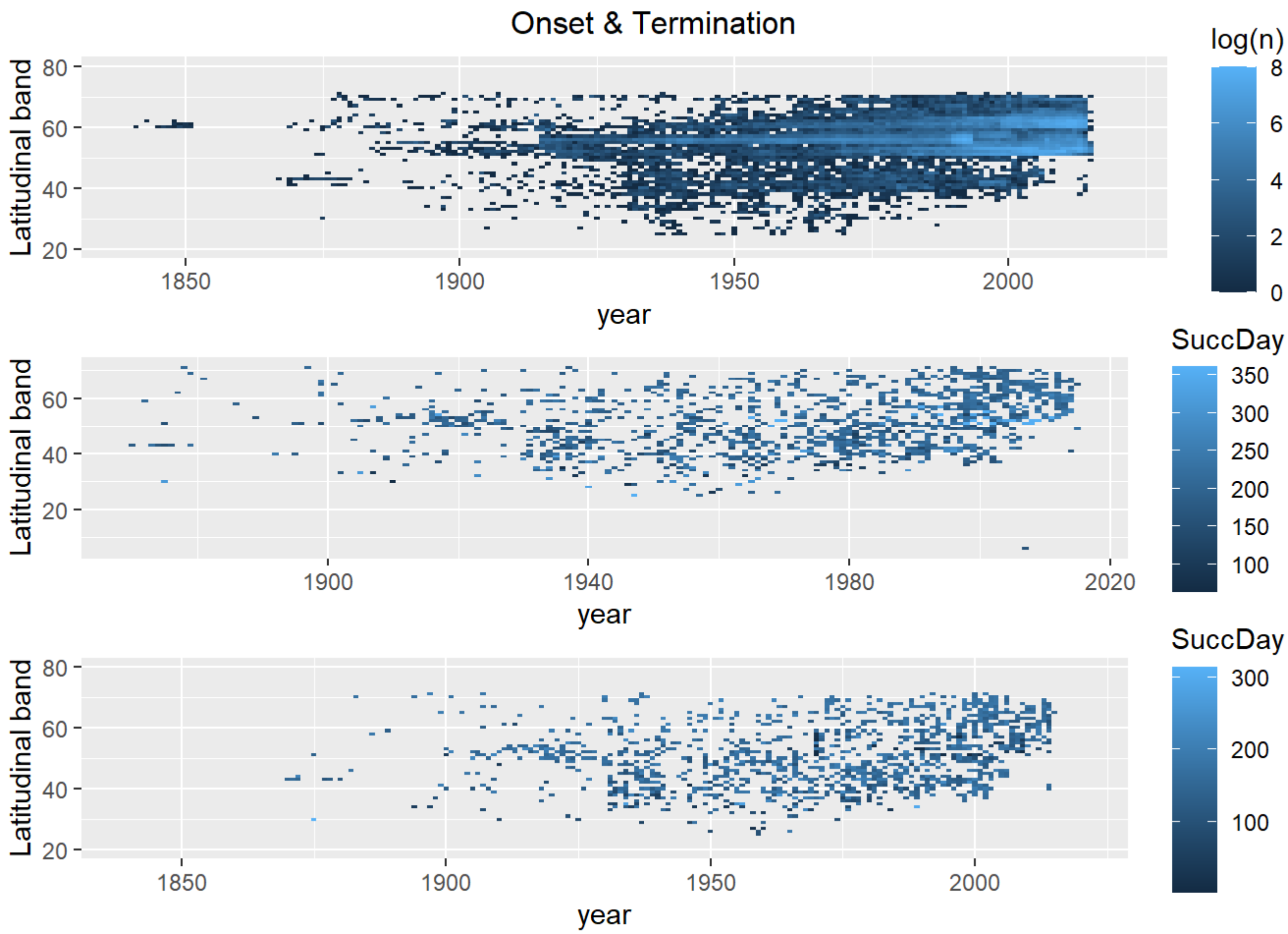


Data exploration: year

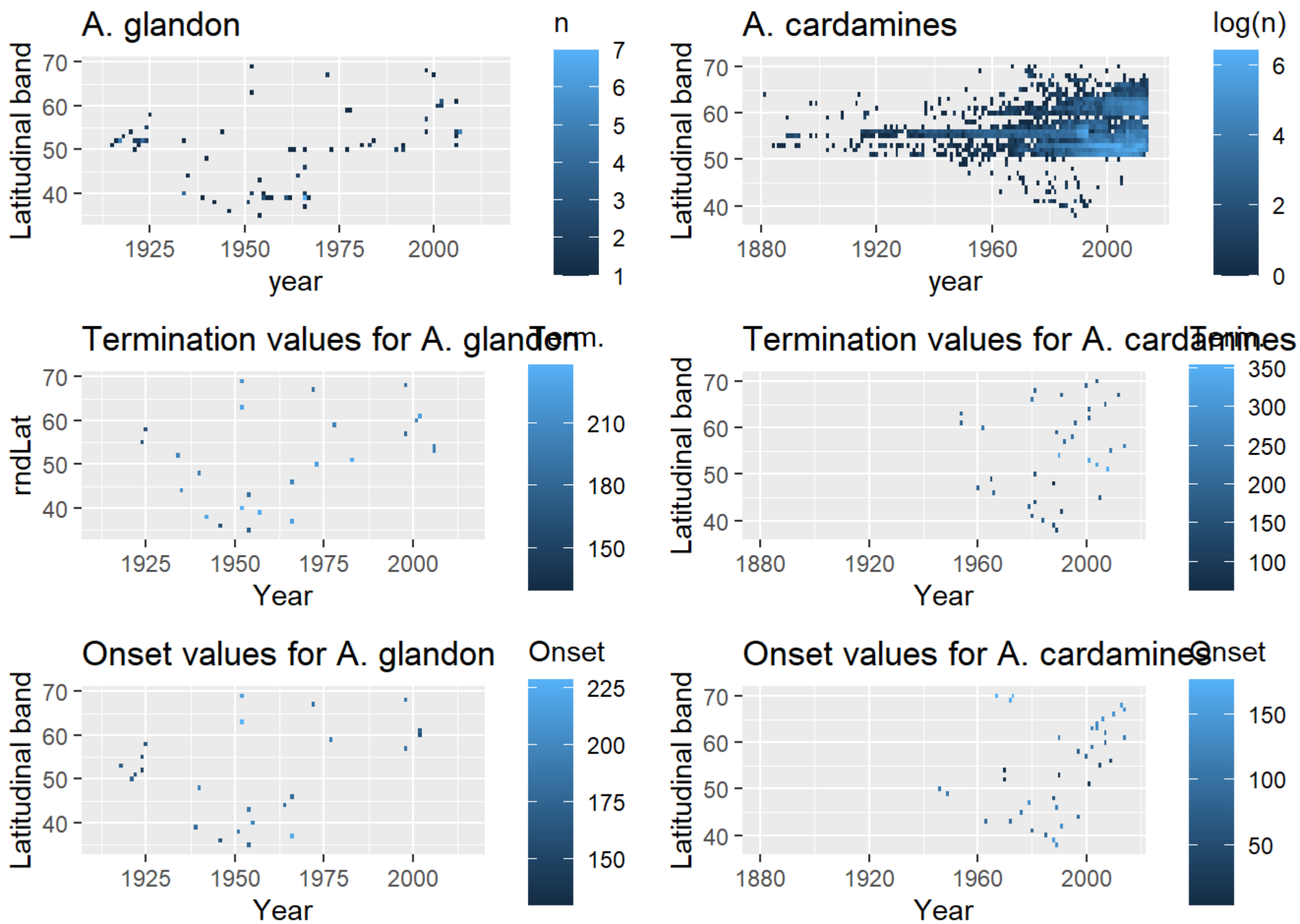
As expected, most data are quite recent. By selecting the min and max day of year per latitudinal band as onset & termination, the authors vastly decrease their sample size and remove most of the variation along the year and altitude axes

We arbitrarily selected one species with a low sample size and one species with a large sample size, to visualize.

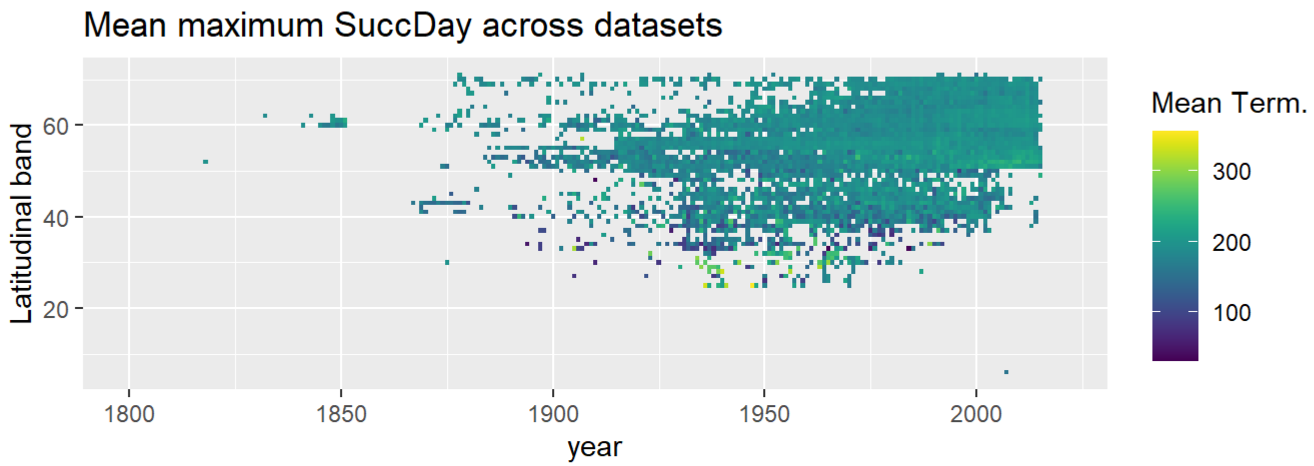
```
yrdata<-fricdata%>% group_by(year, rndLat) %>% tally()
# Heatmap
peakp1<-ggplot(yrdata, aes(year, rndLat, fill= log(n))) +
  geom_tile() + xlim(1840,2020) + ylim(20,80) + ylab("Latitudinal band")

#Onset heatmap
onsetdata<-fricdata%>% group_by(name, region, rndLat) %>% filter(SuccDay==min(SuccDay)) %>% select(name, region, rndLat, yea
r, SuccDay)
onsetp1<-ggplot(onsetdata, aes(year, rndLat, fill= SuccDay)) +
  geom_tile() + xlim(1840,2020) + ylim(20,80) + ylab("Latitudinal band")
termdata<-fricdata%>% group_by(name, region, rndLat) %>% filter(SuccDay==max(SuccDay)) %>% select(name, region, rndLat, yea
r, SuccDay)
termp1<-ggplot(termdata, aes(year, rndLat, fill= SuccDay)) +
  geom_tile() + ylab("Latitudinal band")
grid.arrange(peakp1,termp1,onsetp1, top="Onset & Termination")
```

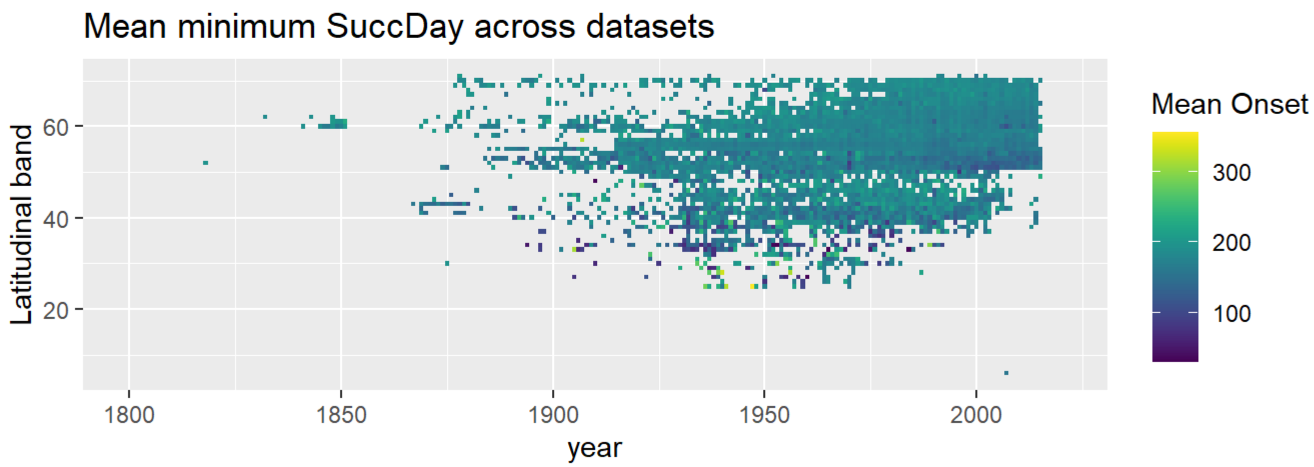

Onset & Termination
log(n)
8
6
4
2
0
Latitudinal band
80
60
40
20
1850
1900
1950
2000
year
SuccDay
350
300
250
200
150
100
Latitudinal band
60
40
20
1900
1940
1980
2020
year
SuccDay
300
200
100
Latitudinal band
80
60
40
20
1850
1900
1950
2000
year

```
## Let's look at 2 species as examples
#Agriades glandon (only in N. America)
agdata<-fricdata %>% filter(name=="Agriades glandon") %>% group_by(year, rndLat) %>% tally()
# Heatmap
peakp1<-ggplot(agdata, aes(year, rndLat, fill= n)) + xlim(min(agdata$year),max(agdata$year)) +
  geom_tile()  + ylab("Latitudinal band") + ggtitle('A. glandon')

#Onset heatmap
ag1<-fricdata%>% filter(name=="Agriades glandon")%>% group_by(name, region, rndLat) %>% filter(SuccDay==min(SuccDay)) %>% se
lect(name, region, rndLat, year, SuccDay)
onsetp1<-ggplot(ag1, aes(year, rndLat, fill= SuccDay)) +
  geom_tile()  + labs(y="Latitudinal band", x="Year", fill="Onset", title="Onset values for A. glandon") + xlim(min(agdata$y
ear),max(agdata$year))
ag2<-fricdata%>% filter(name=="Agriades glandon")%>% group_by(name, region, rndLat) %>% filter(SuccDay==max(SuccDay)) %>% se
lect(name, region, rndLat, year, SuccDay)
termp1<-ggplot(ag2, aes(year, rndLat, fill= SuccDay)) +
  geom_tile() + labs("Latitudinal band", x="Year", fill="Term.", title="Termination values for A. glandon") + xlim(min(agdat
a$year),max(agdata$year))
#grid.arrange(peakp1,termp1,onsetp1)

#Anthocharis cardamines = only in Europe
acdata<-fricdata %>% filter(name=="Anthocharis cardamines") %>% group_by(year, rndLat) %>% tally()
# Heatmap
peakp2<-ggplot(acdata, aes(year, rndLat, fill= log(n))) + xlim(1880,max(acdata$year)) +
  geom_tile()  + ylab("Latitudinal band") + ggtitle('A. cardamines')

#Onset heatmap
ac1<-fricdata%>% filter(name=="Anthocharis cardamines")%>% group_by(name, region, rndLat) %>% filter(SuccDay==min(SuccDay))
 %>% select(name, region, rndLat, year, SuccDay)
onsetp2<-ggplot(ac1, aes(year, rndLat, fill= SuccDay)) +
  geom_tile()  + labs(y="Latitudinal band",x="Year",fill="Onset", title="Onset values for A. cardamines") + xlim(1880,max(ac
data$year))
ac2<-fricdata%>% filter(name=="Anthocharis cardamines")%>% group_by(name, region, rndLat) %>% filter(SuccDay==max(SuccDay))
 %>% select(name, region, rndLat, year, SuccDay)
termp2<-ggplot(ac2, aes(year, rndLat, fill= SuccDay)) +
  geom_tile() + labs(y="Latitudinal band",x="Year",fill="Term.", title="Termination values for A. cardamines") + xlim(1880,m
ax(acdata$year))
#grid.arrange(peakp1,termp1,onsetp1)
grid.arrange(peakp1,peakp2,termp1,termp2,onsetp1,onsetp2, nrow=3)
```

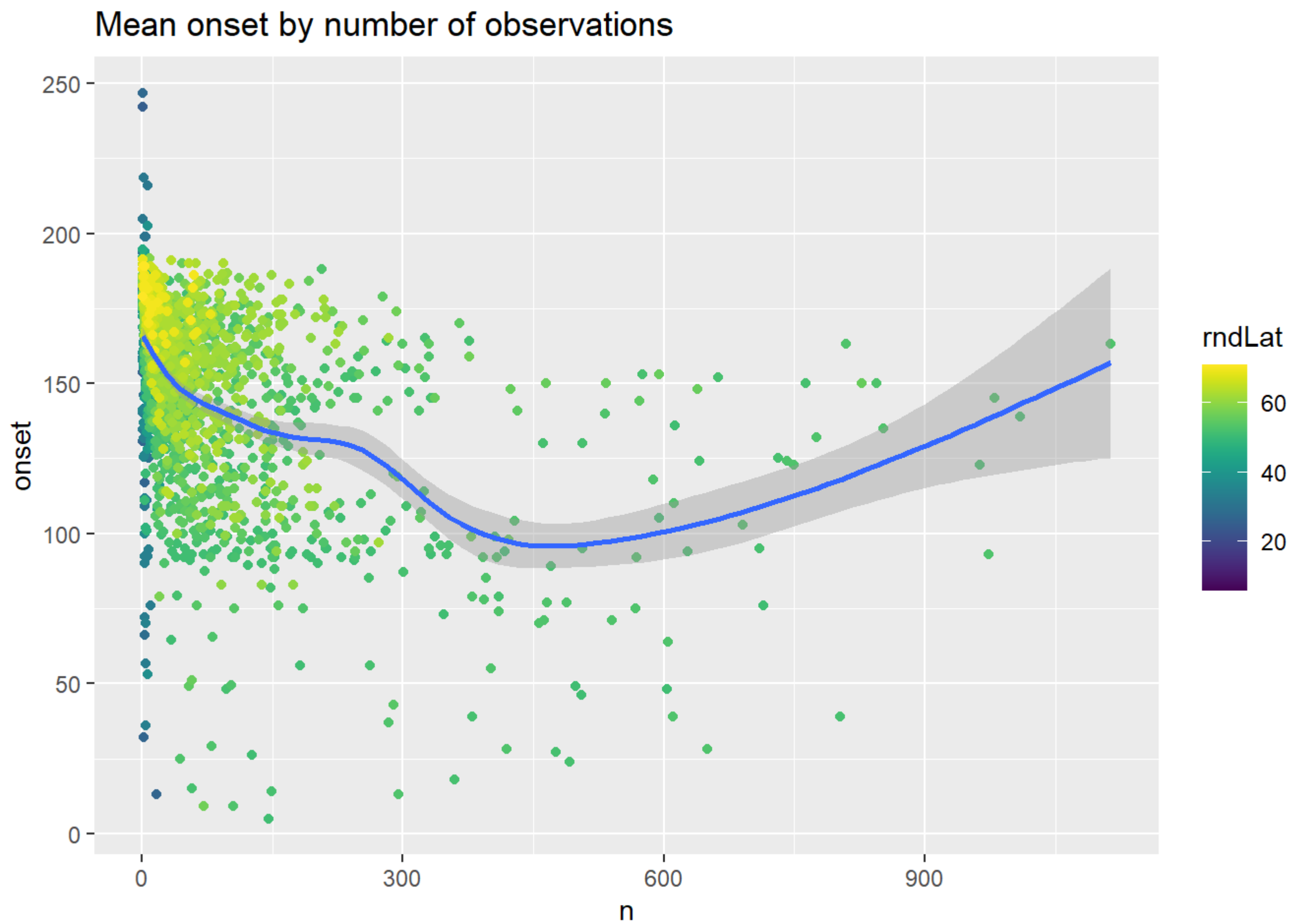

A. glandon
Latitudinal band
year
n
A. cardamines
Latitudinal band
year
log(n)
Termination values for A. glandon
rndLat
Year
Term.
Termination values for A. cardamines
Latitudinal band
Year
Term.
Onset values for A. glandon
Latitudinal band
Year
Onset
Onset values for A. cardamines
Latitudinal band
Year
Onset

```
rm(ag1,ag2,ac1,ac2)
yrdata<-fricdata%>% group_by(year, rndLat, name, region) %>% add_count() %>% summarize(MinSD=min(SuccDay), MaxSD=max(SuccDa
y), n=length(n))

yrdata1<-yrdata %>%group_by(year, rndLat) %>% summarize(meanmin=mean(MinSD, na.rm=T),meanmax=mean(MaxSD,na.rm=T), nrec=mean
(n, na.rm=T))

# Heatmap: onset
onsetp1<-ggplot(yrdata1, aes(year, rndLat, fill= meanmin)) +
  geom_tile() + scale_fill_viridis() +
  labs(y="Latitudinal band", fill="Mean Onset", title="Mean minimum SuccDay across datasets") + xlim(1800,2020)
# Heatmap: term
termp1<-ggplot(yrdata1, aes(year, rndLat, fill= meanmax)) +
  scale_fill_viridis() +
  geom_tile() + labs(y="Latitudinal band", fill="Mean Term.", title="Mean maximum SuccDay across datasets")+ xlim(1800,2020)
grid.arrange(termp1,onsetp1)
```

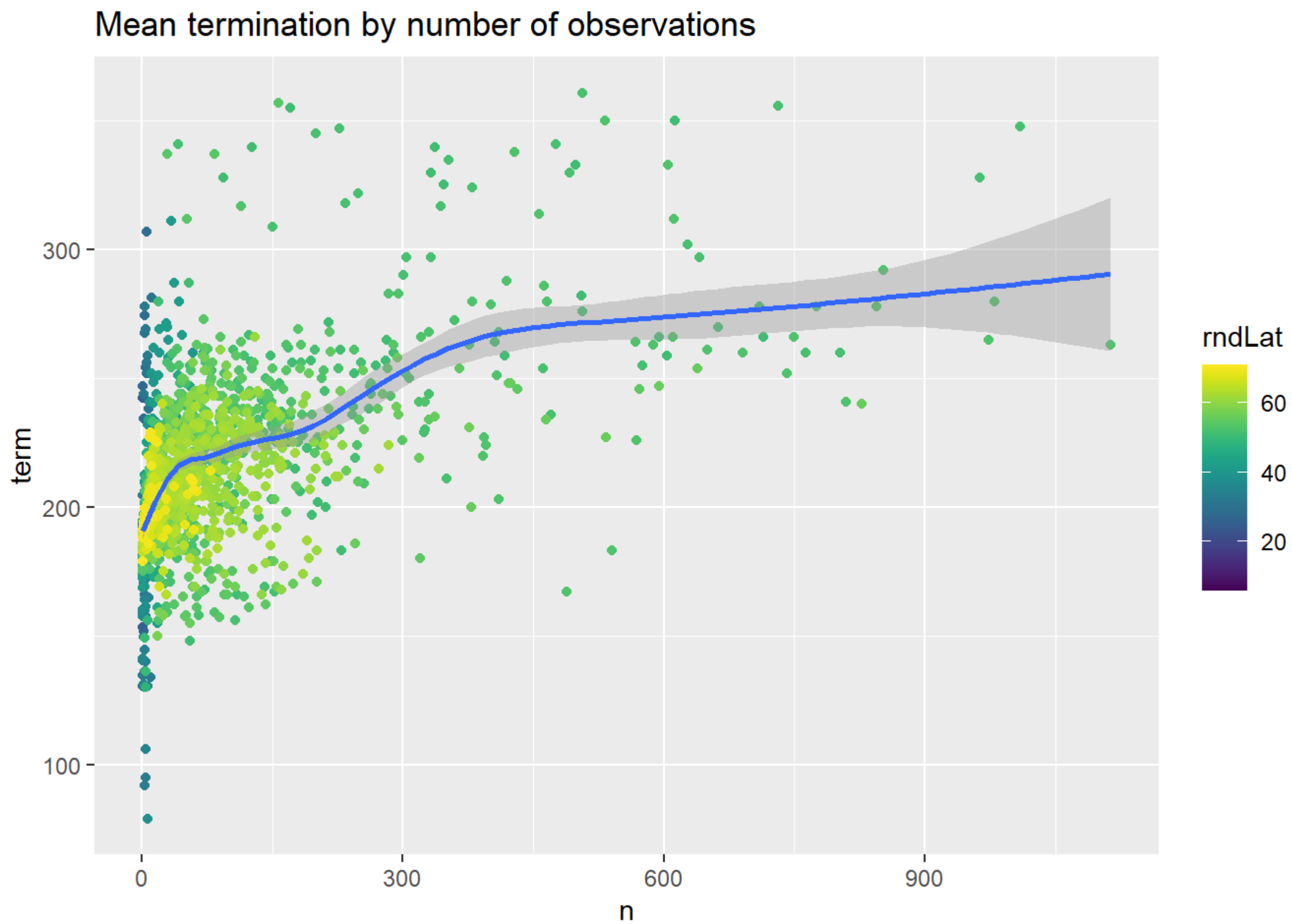





```
bylat<-yrdata %>% group_by(rndLat,n) %>% summarize(onset=mean(MinSD),term=mean(MaxSD))
ggplot(data=bylat, aes(x=n, y=onset, color=rndLat)) + geom_point() + geom_smooth() + scale_color_viridis() + labs(title="Mea
n onset by number of observations")
```




```
ggplot(data=bylat, aes(x=n, y=term, color=rndLat)) + geom_point() + geom_smooth() + scale_color_viridis() + labs(title="Mean termination by number of observations")
```


Mean termination by number of observations
term
n
300
200
100
0
300
600
900
rndLat
60
40
20

End of File.