MIVI

Will Bennett

12/14/2023

Effects of Latitude and Elevation on the Phenology of an Invasive Grass

Japanese Stiltgrass (Microstegium vimineum)

Data source: iNaturalist (link)
Total observations (North America): 13,782
Observations needing phenology annotation: 5,270 (link for reviewing)

Setup

```r
# set lwd
setwd("~/Documents/MIVI/")

library(dplyr)
library(ggplot2)
library(elevatr)
library(corrtable)
library(table1)
library(knitr)
```

Loading data

Export iNaturalist Data

1. Export all *Microstegium vimineum* observations from iNat with columns (`id`, `observed_on`, `latitude`, `longitude`, `place_state_name`, `place_country_name`): Link
2. Export all *M. vimineum* observations with phenology 'No Evidence of Flowering' with column `id`: Link
3. Export all *M. vimineum* observations with phenology 'Flowering' with column `id`: Link
4. Export all *M. vimineum* observations with phenology 'Fruiting' with column `id`: Link

```r
# 1
mivi_all <- read.csv("./MIVI-ALL.csv") %>%
    mutate(date=as.Date(observed_on, format="%Y-%m-%d")) %>% select(-observed_on)

# 2
mivi_young <- read.csv("MIVI-YOUNG.csv") %>%
    mutate(stage="Vegetation")

# 3
mivi_flowering <- read.csv("./MIVI-FLOWERING.csv") %>%
    mutate(stage="Flowering")
mivi_flowering <- mivi_flowering %>% left_join(mivi_all, by="id")

# 4
mivi_fruiting <- read.csv("./MIVI-FRUITING.csv") %>%
    mutate(stage="Fruiting")


# join each based on id
mivi_all <- mivi_all %>% left_join(mivi_young, by="id")
mivi_all <- mivi_all %>% left_join(mivi_fruiting, by="id") %>%
    mutate(stage = coalesce(stage.x, stage.y)) %>% select(-stage.x, -stage.y)
mivi_all <- rbind(mivi_all, mivi_flowering)

# memory cleanup
rm(mivi_young, mivi_flowering, mivi_fruiting)
```

Retrieve Elevation Information

```r
mivi_annotated <- mivi_all %>% filter(!is.na(stage))

coords <- data.frame(x=mivi_annotated$longitude,
                     y=mivi_annotated$latitude, ele_id=mivi_annotated$id)
# pipe to 'slice(1:100)' to get a subset for reducing retrieval time

# retrieve elevation from USGS (takes a while)
elevations <- get_elev_point(coords, prj=4326, src="epqs")

mivi_all <- mivi_all %>% left_join(elevations, by=join_by("id" == "ele_id")) %>%
    select(-elev_units, -geometry)

rm(coords, elevations)

write.csv(mivi_all, file="./MIVI-PROCESSED.csv", na='')
```

Or load from processed file

```r
mivi_all <- read.csv("./MIVI-PROCESSED.csv") %>% select(-X) %>%
    mutate(date=as.Date(date, format="%Y-%m-%d"))
```

Data processing

```r
# remove known incorrect records
mivi_all <- subset(mivi_all, id != "130398055")

# Get Julian day
mivi_all <- mivi_all %>% mutate(julian = as.integer(strftime(date, format="%j")))

# make phenology a factor type (not necessary)
mivi_all$stage <- factor(mivi_all$stage, ordered=TRUE,
                         levels=c("Vegetation", "Flowering", "Fruiting"))

# Select only observations with phenology data
mivi_annotated <- mivi_all %>% filter(!is.na(stage))

# Group into quartiles by latitude
mivi_annotated$group <- ntile(mivi_annotated$latitude, 4)
```

Descriptive Tables

```r
# All observations by country
table1(~ place_country_name, data=mivi_all)
```

	Overall
	(N=14162)
place_country_name	
Canada	26 (0.2%)
United States	14136 (99.8%)

```r
# All observations by state (includes Ontario)
table1(~ place_state_name, data=mivi_all)
```

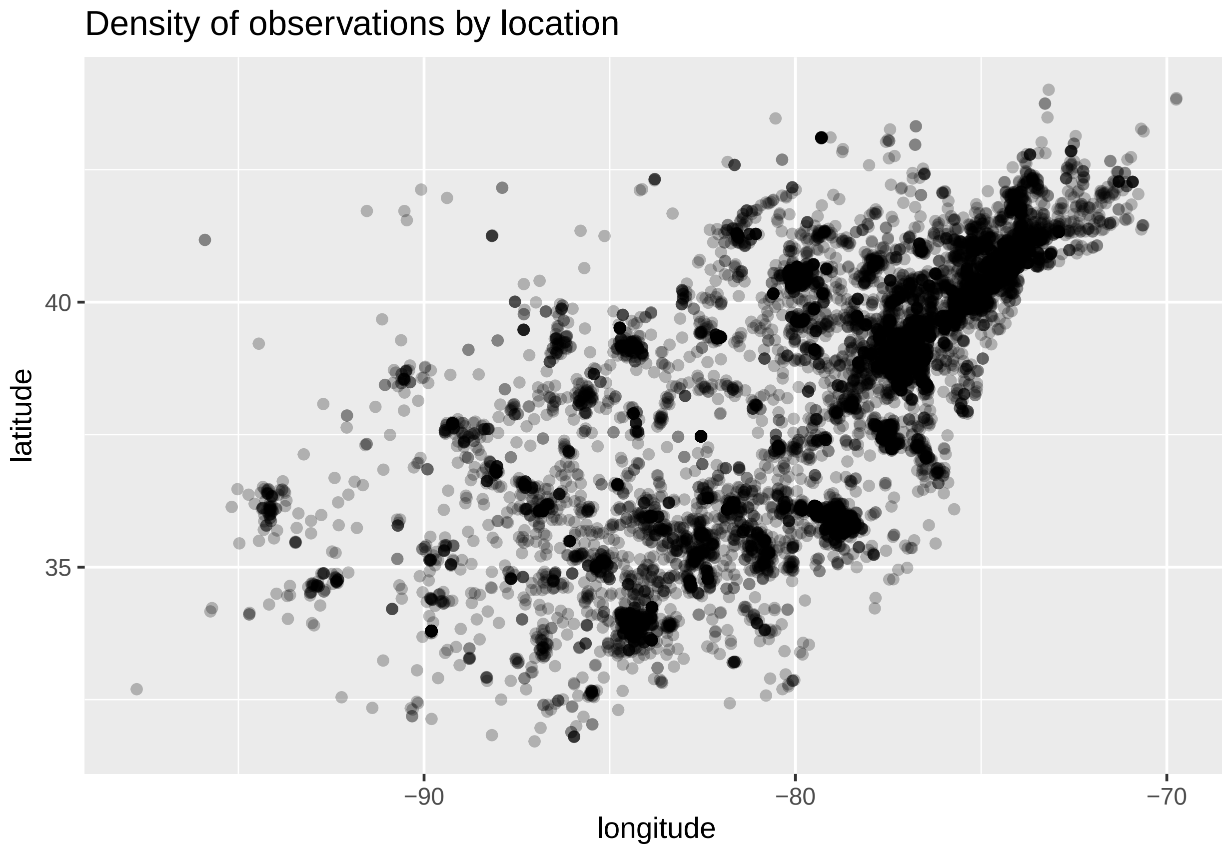
	Overall
	(N=14162)
place_state_name	
Alabama	212 (1.5%)
Arkansas	163 (1.2%)
Connecticut	229 (1.6%)
Delaware	228 (1.6%)
District of Columbia	127 (0.9%)
Georgia	597 (4.2%)
Illinois	123 (0.9%)
Indiana	137 (1.0%)
Iowa	3 (0.0%)
Kentucky	350 (2.5%)
Louisiana	2 (0.0%)
Maine	4 (0.0%)
Maryland	1746 (12.3%)
Massachusetts	91 (0.6%)
Michigan	7 (0.0%)
Mississippi	133 (0.9%)
Missouri	56 (0.4%)
Nebraska	2 (0.0%)
New Jersey	1307 (9.2%)
New York	784 (5.5%)
North Carolina	2032 (14.3%)
Ohio	548 (3.9%)
Oklahoma	9 (0.1%)
Ontario	26 (0.2%)
Pennsylvania	1982 (14.0%)
Rhode Island	17 (0.1%)
South Carolina	246 (1.7%)
Tennessee	652 (4.6%)
Texas	1 (0.0%)
Vermont	15 (0.1%)
Virginia	2000 (14.1%)
West Virginia	333 (2.4%)

```r
table1(~ latitude + longitude + elevation + julian + place_country_name | stage, data=mivi_all %>% muta
```

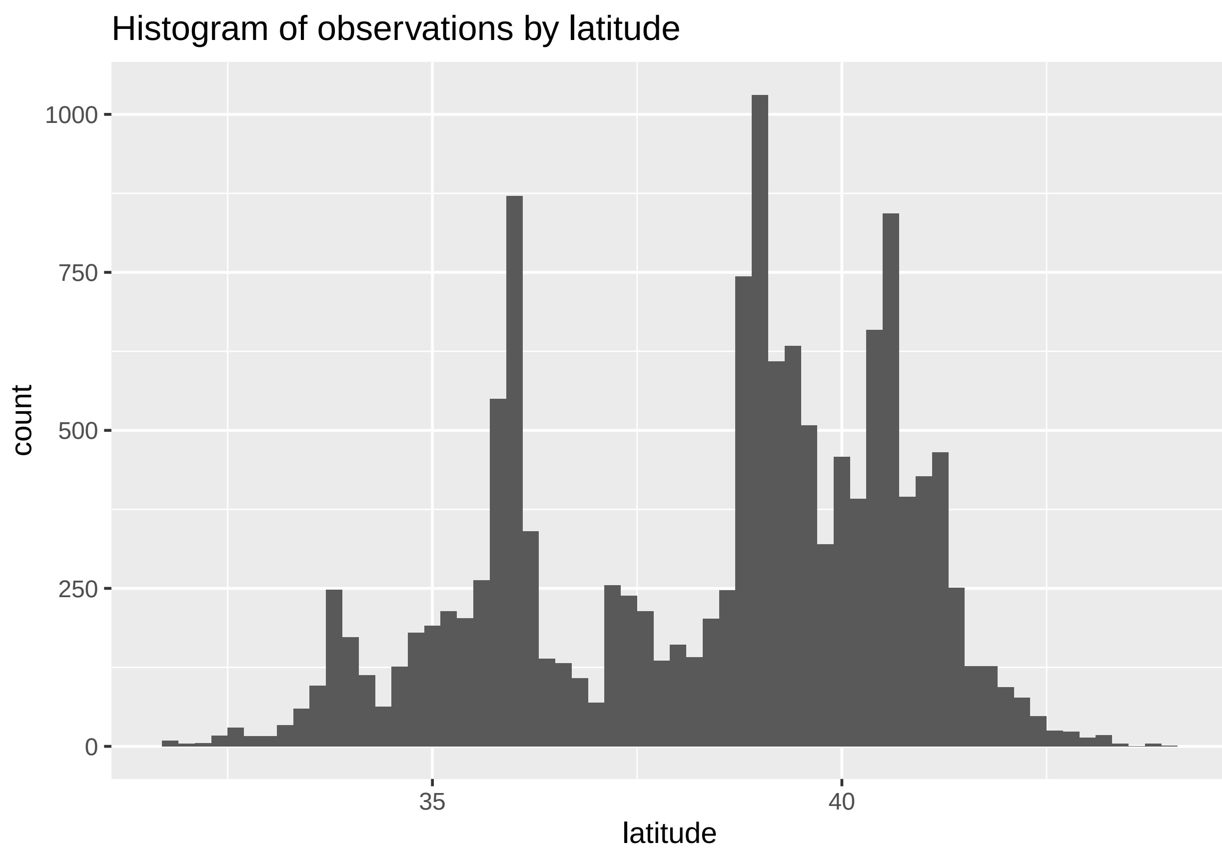
	Vegetation	Flowering	Fruiting	NA	Overall
	(N=6964)	(N=254)	(N=1930)	(N=5014)	(N=14162)
latitude					
Mean (SD)	38.1 (2.38)	39.3 (1.78)	39.0 (2.19)	38.5 (2.21)	38.4 (2.31)
Median [Min, Max]	38.8 [31.8, 44.0]	39.6 [32.6, 41.5]	39.3 [31.7, 43.5]	39.1 [32.2, 43.0]	39.0 [31.7, 44.0]
longitude					
Mean (SD)	-79.2 (4.28)	-77.7 (3.79)	-78.6 (4.29)	-79.1 (4.34)	-79.0 (4.30)
Median [Min, Max]	-78.2 [-97.7, -69.7]	-77.0 [-92.9, -72.9]	-77.4 [-95.8, -70.6]	-77.8 [-95.2, -70.8]	-77.8 [-97.7, -69
elevation					
Mean (SD)	193 (193)	162 (174)	201 (205)	NA (NA)	194 (195)
Median [Min, Max]	134 [-0.300, 1720]	110 [0.400, 1010]	138 [-0.270, 1600]	NA [NA, NA]	134 [-0.300, 172
Missing	39 (0.6%)	2 (0.8%)	11 (0.6%)	5014 (100%)	5066 (35.8%)
julian					
Mean (SD)	182 (44.6)	257 (10.9)	269 (16.9)	236 (43.9)	214 (53.1)
Median [Min, Max]	176 [60.0, 327]	256 [224, 285]	267 [207, 330]	238 [1.00, 365]	221 [1.00, 365]
place_country_name					
Canada	20 (0.3%)	0 (0%)	6 (0.3%)	0 (0%)	26 (0.2%)
United States	6944 (99.7%)	254 (100%)	1924 (99.7%)	5014 (100%)	14136 (99.8%)

Basic Descriptive Plots

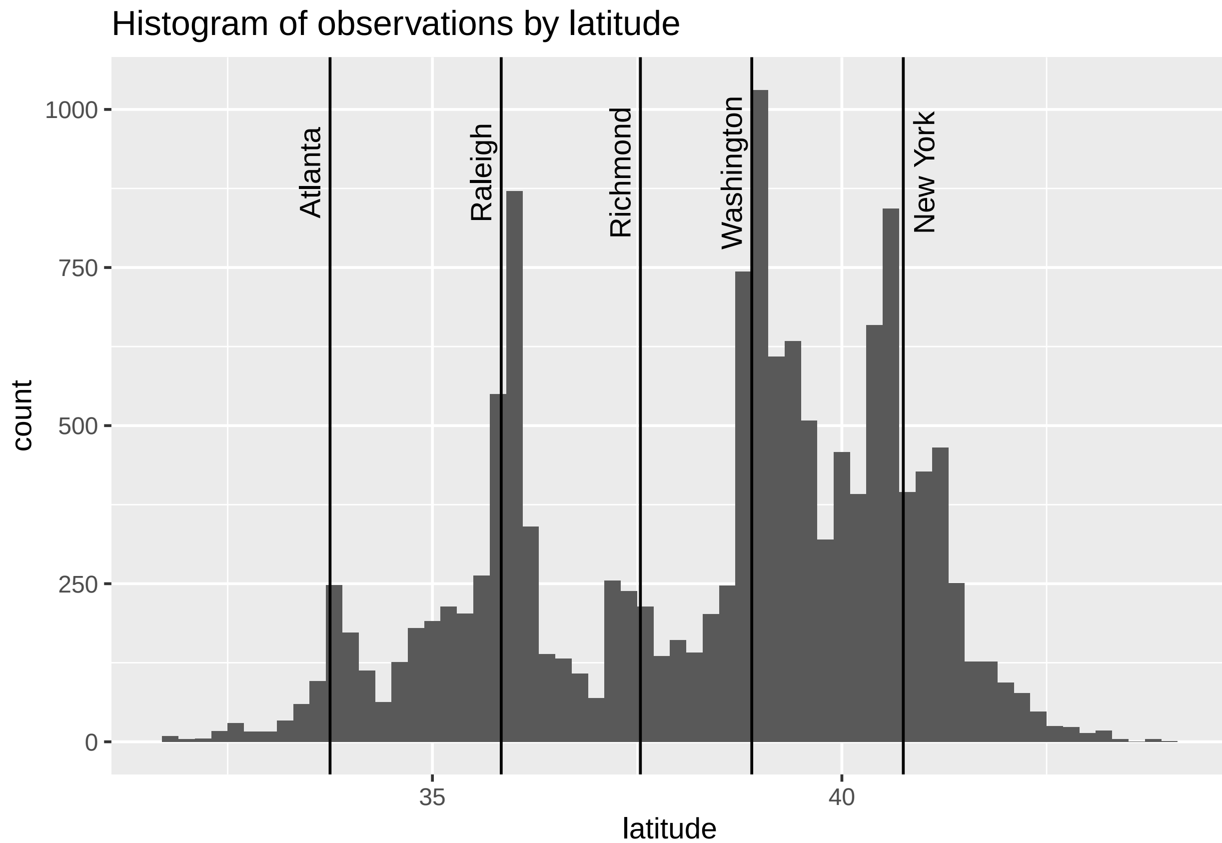
```r
# Density of all observations by lat/lon
ggplot(mivi_all, aes(x=longitude,y=latitude)) + geom_point(alpha=0.25) +
    labs(title="Density of observations by location")
```



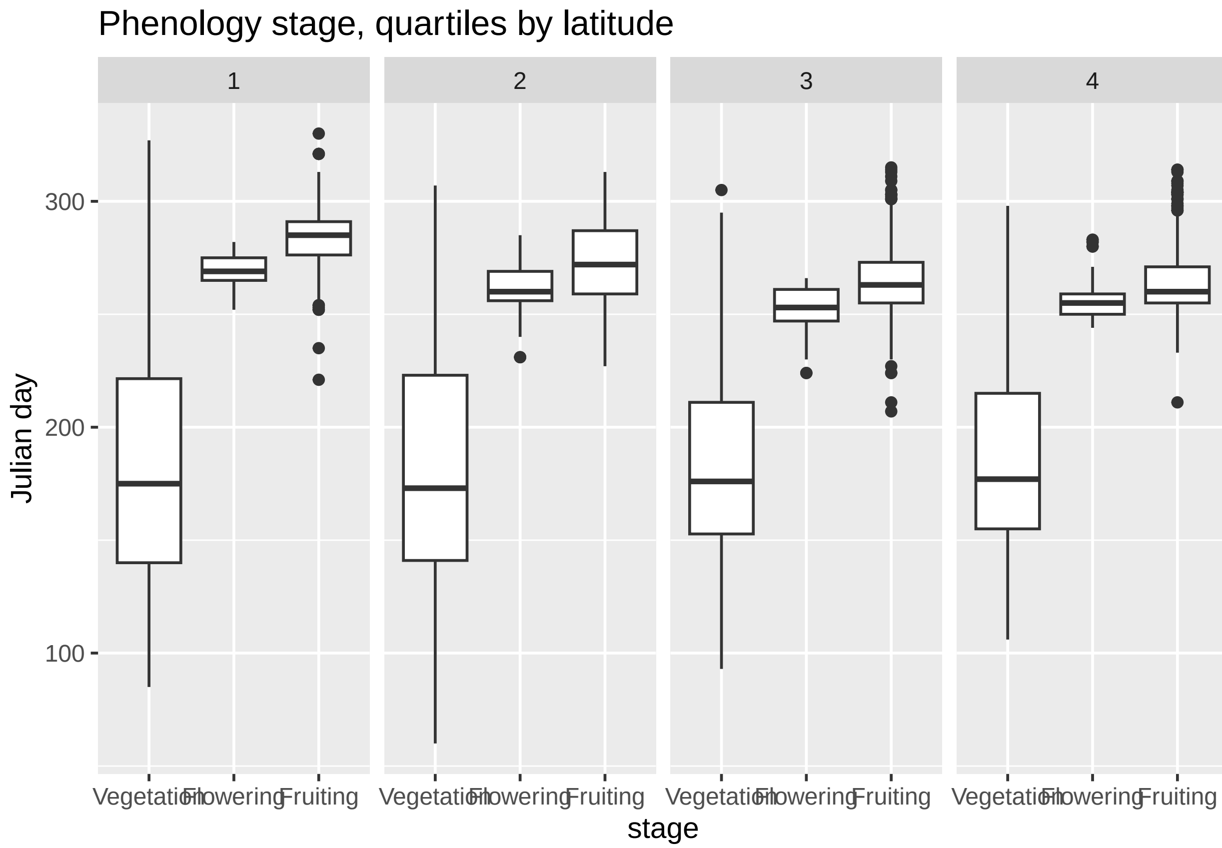

```r
# Histogram of all observations by lat
ggplot(mivi_all, aes(x=latitude)) + geom_histogram(binwidth=0.2) +
    labs(title="Histogram of observations by latitude")
```




```r
# Histogram with city labels
ggplot(mivi_all, aes(x=latitude)) + geom_histogram(binwidth=0.2) +
    geom_vline(xintercept=33.75) + annotate("text", x=33.5, y=900, label="Atlanta", angle=90) +
    geom_vline(xintercept=35.84) + annotate("text", x=35.59, y=900, label="Raleigh", angle=90) +
    geom_vline(xintercept=37.54) + annotate("text", x=37.29, y=900, label="Richmond", angle=90) +
    geom_vline(xintercept=38.9) + annotate("text", x=38.65, y=900, label="Washington", angle=90) +
    geom_vline(xintercept=40.75) + annotate("text", x=41, y=900, label="New York", angle=90) +
    ylab("count") + labs(title="Histogram of observations by latitude")
```

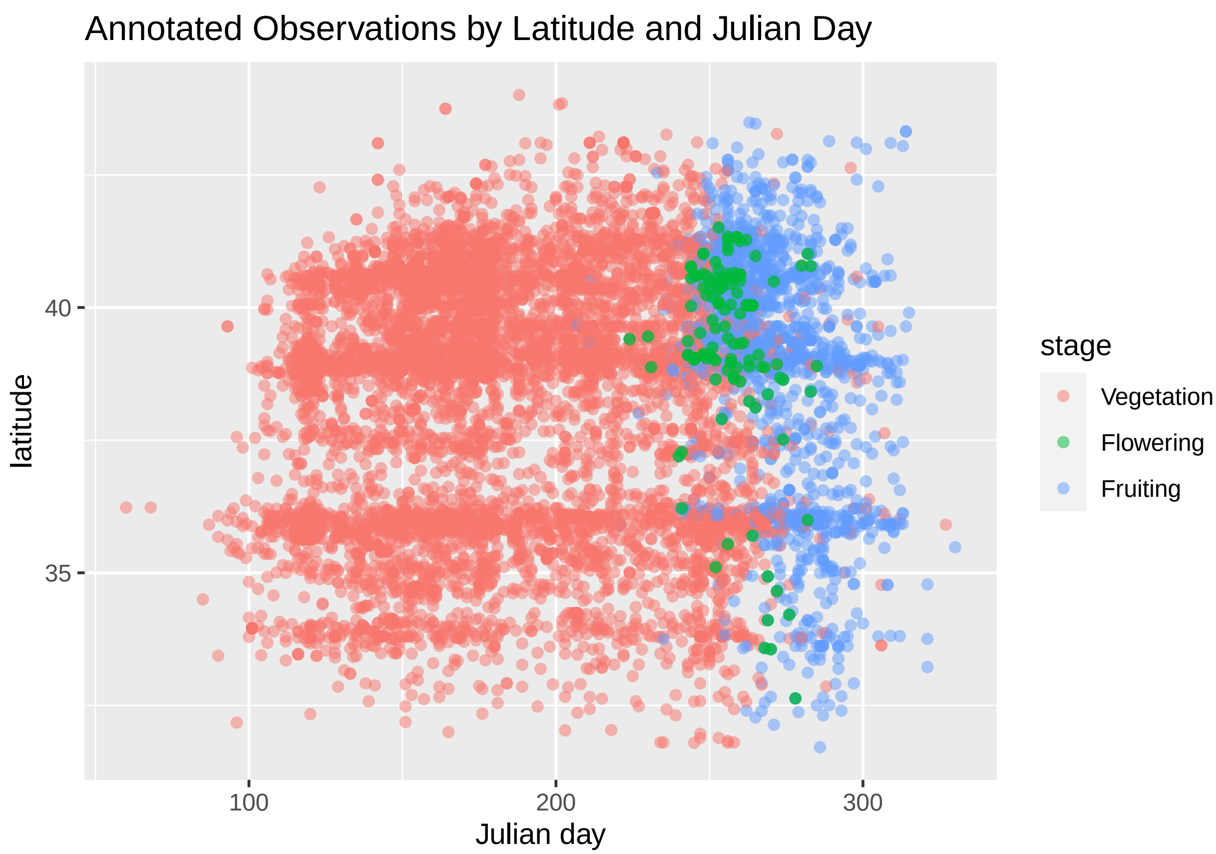



```
# Boxplots of phenology by quartile (notches break hinges for Flowering)
# Slight earlier trend in Fruiting visible
ggplot(mivi_annotated, aes(julian, stage)) + geom_boxplot() + facet_grid(~group) +
    coord_flip() + xlab("Julian day") + labs(title="Phenology stage, quartiles by latitude")
```

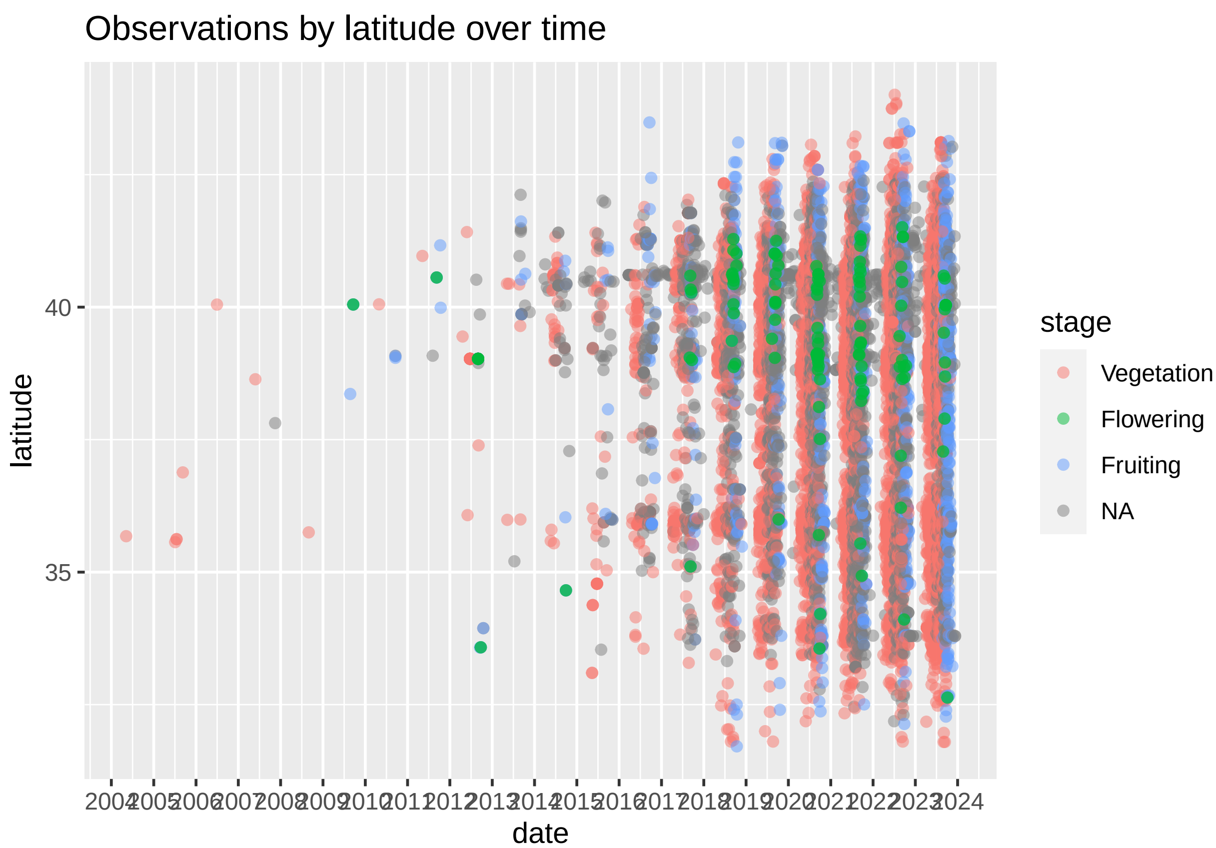


Time Series Plots

```
# Latitude against Julian day
ggplot(mivi_annotated, aes(julian, latitude)) + geom_point(aes(color=stage), alpha=0.5) +
    scale_color_hue() + xlab("Julian day") +
    labs(title="Annotated Observations by Latitude and Julian Day")
```

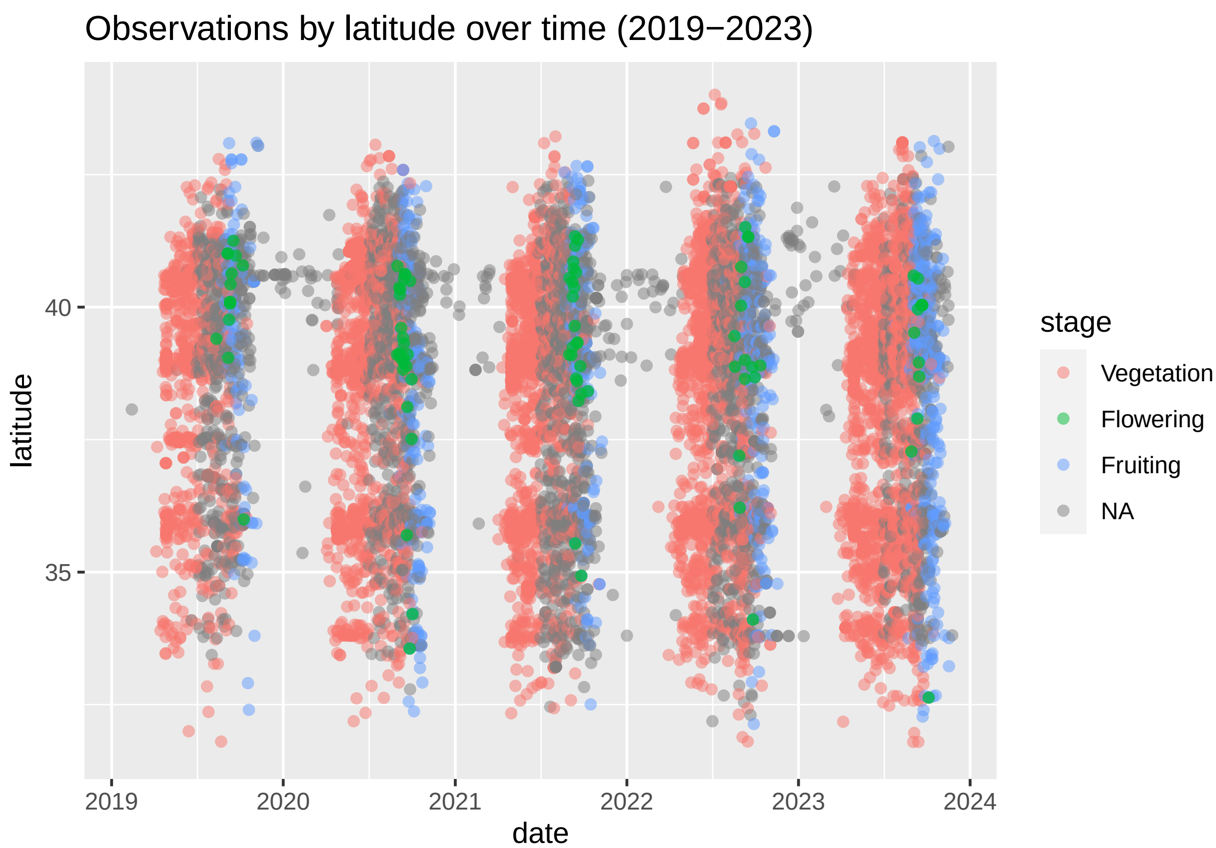



```r
# Latitude against Julian day
# ggplot(mivi_all, aes(julian, latitude)) + geom_point(aes(color=stage), alpha=0.5) +
#     scale_color_hue() + xlab("Julian day") +
#     labs(title="Observations by Latitude and Julian Day")

# Time series by latitude, color by stage
ggplot(mivi_all, aes(date, latitude)) + geom_point(aes(color=stage), alpha=0.5) +
    scale_x_date(date_breaks = "1 year", date_labels = "%Y") + scale_color_hue() +
    labs(title="Observations by latitude over time")
```




```
# Zoom in on recent data
timeclip <- c(as.Date("2019-02-01"), as.Date("2023-11-30"))
ggplot(mivi_all, aes(date, latitude)) + geom_point(aes(color=stage), alpha=0.5) +
    scale_x_date(limits=timeclip, date_breaks = "1 year", date_labels = "%Y") + scale_color_hue() +
    labs(title="Observations by latitude over time (2019-2023)")
```

```
## Warning: Removed 1799 rows containing missing values (‘geom_point()‘).
```



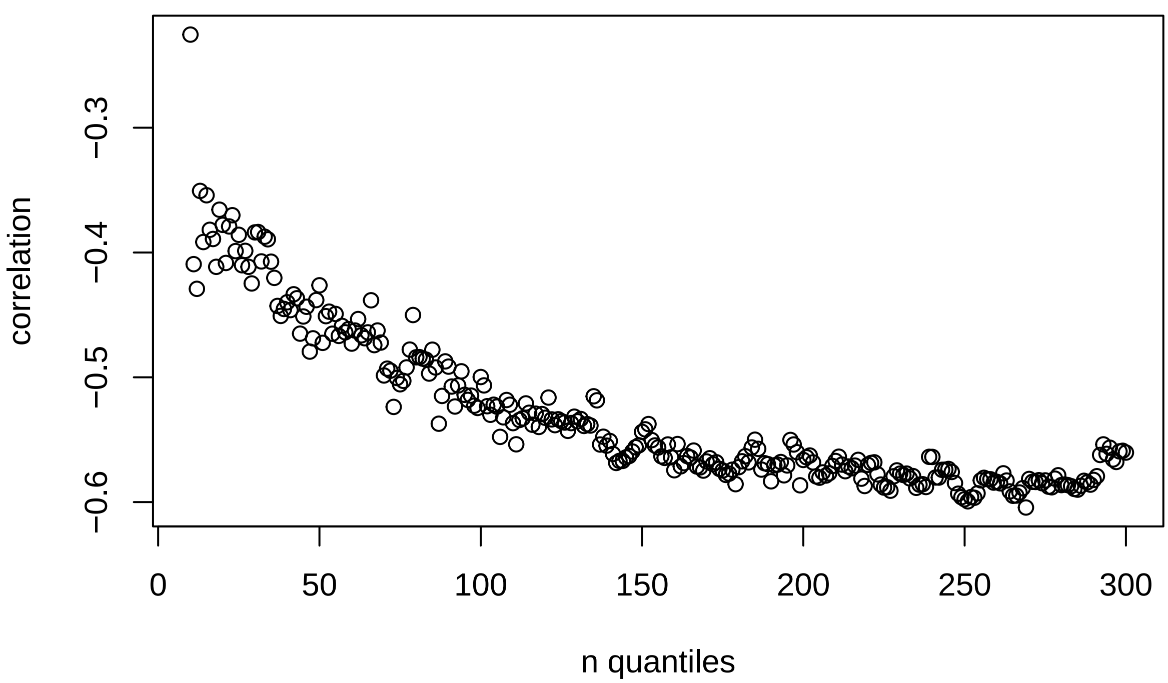
Analysis!

```
# Note: returns Inf if there are none in the selection
first_fruit <- function(df) {
    df <- df %>% filter(stage == "Fruiting")
    if(nrow(df)==0) {return (Inf)}
    return (min(df$julian))
}
```

Latitude

```
get_lat_quants <- function(df, n) {
    # create groups
    df$group <- ntile(df$latitude, n)

    # calculate latitude variables for each group
    a <- df %>% group_by(group) %>% summarize(avglat=mean(latitude),
                                              minlat=min(latitude),
                                              maxlat=max(latitude))

    # first fruiting date in each group
```

```
    b <- df %>% group_by(group) %>% group_map(~first_fruit(.x))

    a <- bind_cols(a, do.call(rbind.data.frame, b)[,1], .name_repair = "unique_quiet") %>%
        mutate(firstfruit = ...5) %>% select(-...5)
    a <- remove_missing(a, finite=TRUE, na.rm=TRUE)  # remove any Inf's

    return (a)
}

# Loop to determine best quantile amount for correlation
quants = data.frame()

for(i in 10:300) {
    # print(i)
    a <- get_lat_quants(mivi_annotated, i)
    quants <- rbind(quants, data.frame(i, cor(a$firstfruit, a$avglat)))
}
rm(i, a)

colnames(quants) <- c("n quantiles", "correlation")
plot(quants)
```

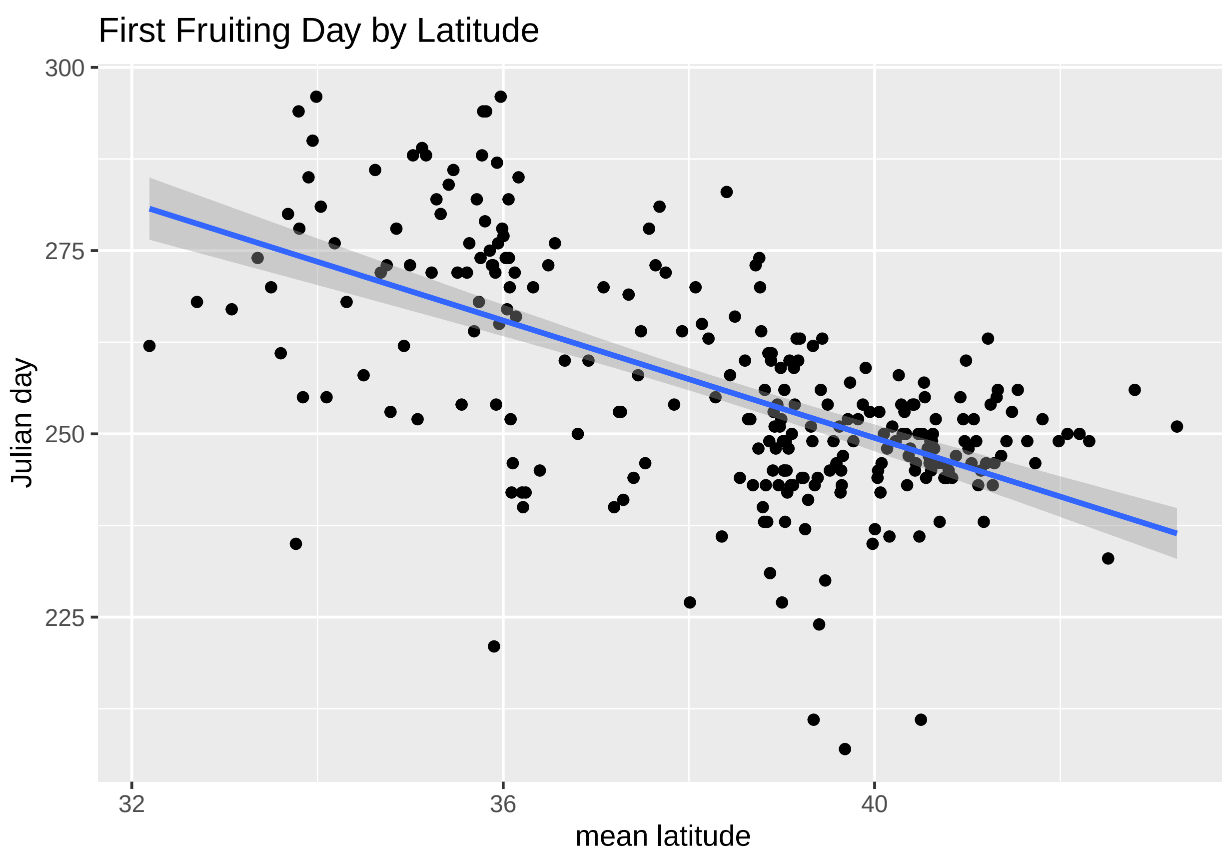



```
y <- which.min(quants$corr)
n <- quants[y,1]  # select n with strongest correlation
paste0("n quantiles for best correlation is: ", n)
```

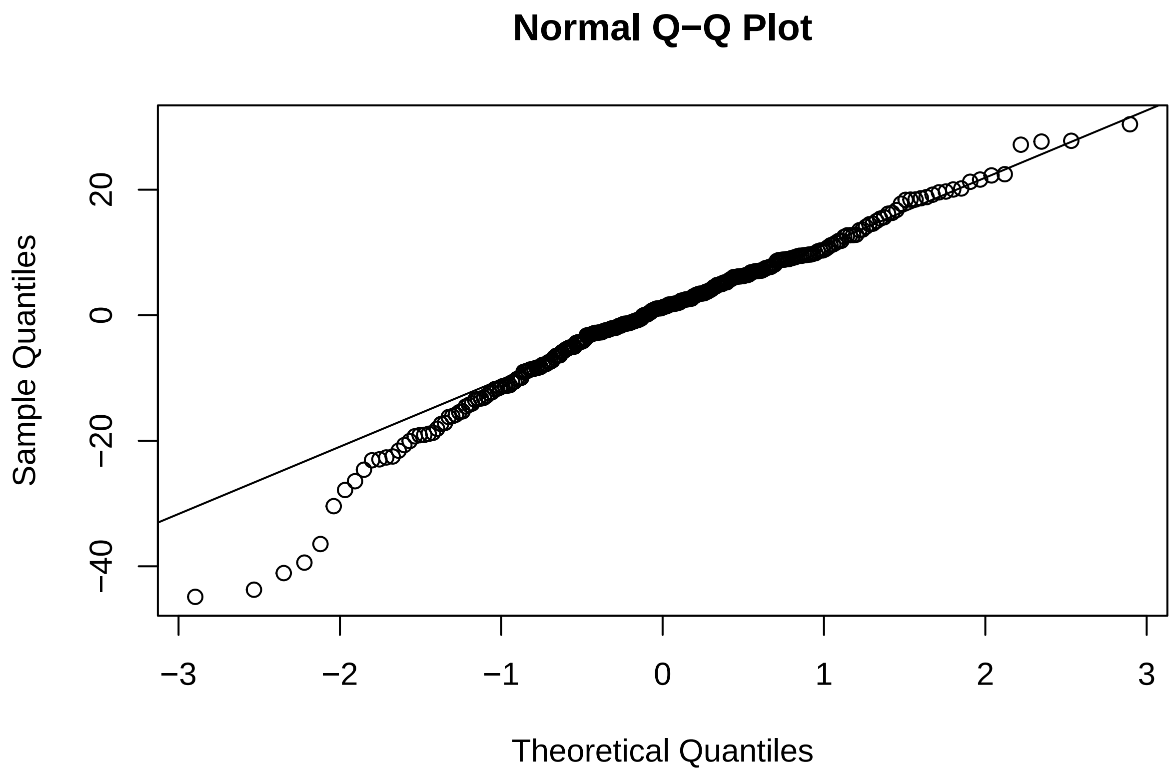
```
## [1] "n quantiles for best correlation is: 269"
```

```r
rm(quants, y)


# Latitude linear model
data <- get_lat_quants(mivi_annotated, n)
model_lat <- lm(firstfruit~avglat, data=data)

# Plot linear model
print(
    ggplot(data, aes(avglat, firstfruit)) + geom_point() + geom_smooth(method='lm') +
        ylab("Julian day") + xlab("mean latitude") + labs(title="First Fruiting Day by Latitude")
)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```




```r
# Q-Q residual plot
res <- resid(model_lat)
qqnorm(res)
qqline(res)
```




```r
paste0("Average absolute residual: ", format(mad(res), digits=6))
```

```
## [1] "Average absolute residual: 11.1664"
```

```r
# Pearson's correlation test
cor.test(data$firstfruit, data$avglat, alternative="less")
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$firstfruit and data$avglat
## t = -12.301, df = 263, p-value < 2.2e-16
## alternative hypothesis: true correlation is less than 0
## 95 percent confidence interval:
##  -1.0000000 -0.5358625
## sample estimates:
##        cor
## -0.6043377
```

```r
correlation_matrix(data, use="lower")
```

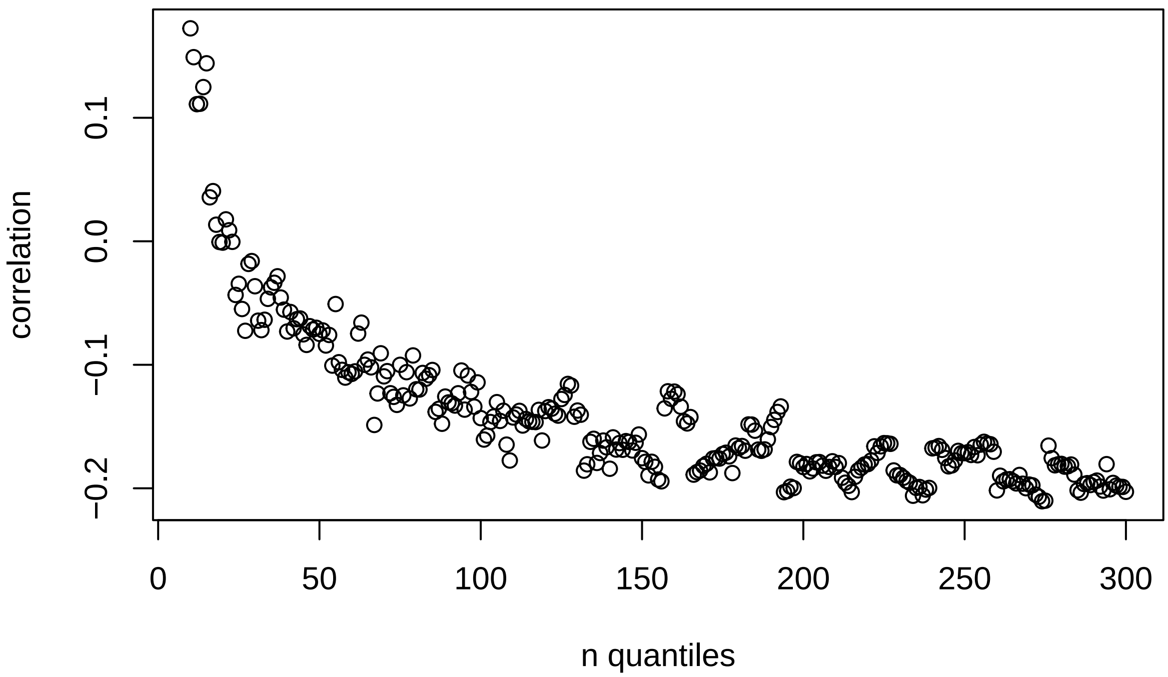
```
##            group       avglat      minlat      maxlat      firstfruit
## group      " 1.000   " ""          ""          ""          ""
## avglat     " 0.978***" " 1.000   " ""          ""          ""
```

```
## minlat     " 0.977***" " 1.000***" " 1.000   " ""          ""
## maxlat     " 0.978***" " 1.000***" " 0.999***" " 1.000   " ""
## firstfruit "-0.591***" "-0.604***" "-0.604***" "-0.604***" " 1.000   "
```

```
# kable(correlation_matrix(data, use="lower"), booktabs=TRUE, format="latex")
# kable(correlation_matrix(data, use="lower"), booktabs=TRUE)

rm(data, model_lat, res)
```

Elevation

```
get_ele_quants <- function(df, n) {
    # Create groups
    df$group <- ntile(mivi_annotated$elevation, n)

    # mean lat for each group
    a <- df %>% group_by(group) %>% summarize(avgele=mean(elevation))

    # first fruiting date in each group
    b <- df %>% group_by(group) %>% group_map(~first_fruit(.x))

    a <- bind_cols(a, do.call(rbind.data.frame, b)[,1], .name_repair = "unique_quiet") %>%
        mutate(firstfruit = ...3) %>% select(-...3)
    a <- remove_missing(a, finite=TRUE, na.rm=TRUE)  # remove any Inf's

    return (a)
}

# Loop to determine best quantile amount for correlation
quants = data.frame()

for(i in 10:300) {
    # print(i)
    a <- get_ele_quants(mivi_annotated, i)
    quants <- rbind(quants, data.frame(i, cor(a$firstfruit, a$avgele)))
}
rm(i, a)

colnames(quants) <- c("n quantiles", "correlation")
plot(quants)
```

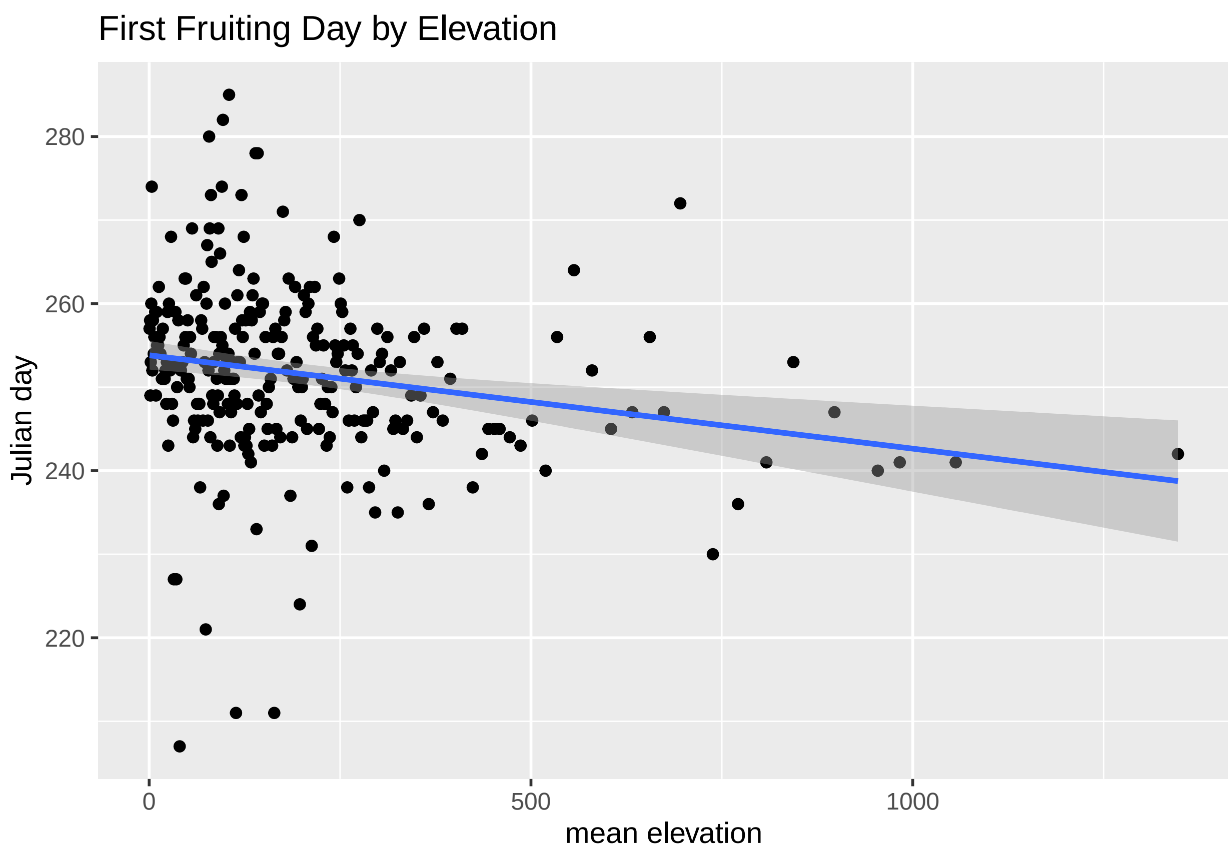



```
y <- which.min(quants$corr)
n <- quants[y,1]  # select n with strongest correlation
paste0("n quantiles for best correlation is: ", n)
```

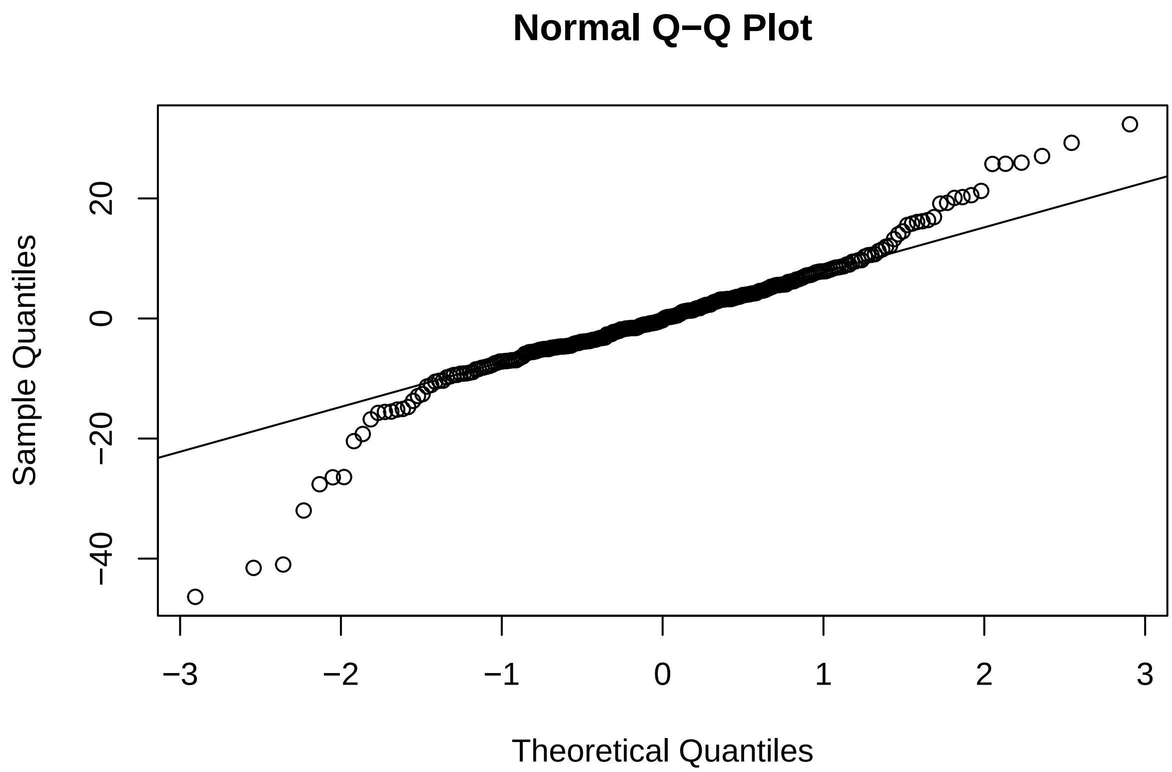
```
## [1] "n quantiles for best correlation is: 274"
```

```
rm(quants, y)


# Elevation linear model
data = get_ele_quants(mivi_annotated, n)
model_ele <- lm(firstfruit~avgele, data=data)

# Plot linear model
ggplot(data, aes(avgele, firstfruit)) + geom_point() + geom_smooth(method='lm') +
    ylab("Julian day") + xlab("mean elevation") + labs(title="First Fruiting Day by Elevation")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
# Q-Q residual plot
res <- resid(model_ele)
qqnorm(res)
qqline(res)
```




```r
paste0("Average absolute residual: ", format(mad(res), digits=6))
```

```
## [1] "Average absolute residual: 7.21881"
```

```r
# Pearson's correlation test
cor.test(data$firstfruit, data$avgele, alternative="less")
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$firstfruit and data$avgele
## t = -3.5446, df = 271, p-value = 0.0002315
## alternative hypothesis: true correlation is less than 0
## 95 percent confidence interval:
##  -1.0000000 -0.1130995
## sample estimates:
##        cor
## -0.2104939
```

```r
rm(data, model_ele, res)
```