R workshop

Jackie Maud

21 January 2019

Day 2

Tidy Data in spreadsheets

The functions for tidying data are:

:: tells which package to use function from

tidyr::spread() tidyr::gather()

Spread

3 principal arguments

1. the data
2. the *key* column variable will become the new column names
3. the *value* column variable whch will fill the new column variables

Use surveys dataset

Make from LONG to WIDE (spread)

```
surveys<-read_csv(here::here("read_data", "surveys.csv"))
```

```
## Parsed with column specification:
## cols(
##   record_id = col_double(),
##   month = col_double(),
##   day = col_double(),
##   year = col_double(),
##   plot_id = col_double(),
##   species_id = col_character(),
##   sex = col_character(),
##   hindfoot_length = col_double(),
##   weight = col_double()
## )
```

```
library(tidyverse)

#create a wide data format of surveys using spread

#first create a summary

surveys_gw <- surveys %>%
    drop_na(weight) %>%
      group_by(species_id) %>%
  summarize(mean_weight = mean(weight))


str(surveys_gw)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    25 obs. of  2 variables:
##  $ species_id : chr  "BA" "DM" "DO" "DS" ...
##  $ mean_weight: num  8.6 43.2 48.9 120.1 159.2 ...
```

```
wide_surveys_gw <-surveys_gw %>%
  spread(key = species_id, value = mean_weight)
```

gather()

Now back to long data from wide

gather takes 4 arguments

1. data
2. *key*
3. *value*
4. names of columns we use to fill the key variable (or drop)

```
long_surveys_gw <-wide_surveys_gw %>%
  gather(key = species_id, value = mean_weight)
```

Sending Tidy Data

Changelog

- Update your change log with changes to raw data/project

Data dictionary

*Create to define our variables

```
tidy_gsi <- read_csv(here::here("read_data", "tidy_gsi.csv"))
```

```
## Parsed with column specification:
## cols(
##   hakai_id = col_character(),
##   stock_1 = col_character(),
##   region_1 = col_double(),
##   prob_1 = col_double(),
##   stock_2 = col_character(),
##   region_2 = col_double(),
##   prob_2 = col_double(),
##   stock_3 = col_character(),
##   region_3 = col_double(),
##   prob_3 = col_double(),
##   stock_4 = col_character(),
##   region_4 = col_double(),
##   prob_4 = col_double(),
##   stock_5 = col_character(),
##   region_5 = col_double(),
##   prob_5 = col_double()
## )
```

```
view(tidy_gsi)
```

Analysing data

Importing from Hakai Data Portal

Switched to data_wrangling script to import data into our read_data file

Chl_a, fish and sockeye stock ID data

```
library(here)
```

```
## here() starts at C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2
```

```
fish <-read_csv(here("read_data", "fish.csv"))
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   X1 = col_double(),
##   action = col_logical(),
##   date = col_date(format = ""),
##   package_id = col_logical(),
##   fish_time_out = col_logical(),
##   fish_time_dewar = col_logical(),
##   fork_length_field = col_double(),
##   height_field = col_double(),
##   weight_field = col_logical(),
##   date_processed = col_date(format = ""),
##   weight = col_double(),
##   standard_length = col_double(),
##   fork_length = col_double(),
##   photo_number = col_logical(),
##   comments = col_logical(),
##   quality_log = col_logical()
## )
```

```
## See spec(...) for full column specifications.
```

```
## Warning: 2861 parsing failures.
##  row          col           expected actual
## 1393 photo_number 1/0/T/F/TRUE/FALSE   3142 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2/r
## 2093 photo_number 1/0/T/F/TRUE/FALSE   2835 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2/r
## 2247 photo_number 1/0/T/F/TRUE/FALSE   3204 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2/r
## 2527 photo_number 1/0/T/F/TRUE/FALSE   3137 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2/r
## 2745 photo_number 1/0/T/F/TRUE/FALSE   2009 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop2/r
## .... ............ .................. ...... ....................................................
## See problems(...) for more details.
```

```
chla <- read_csv(here("read_data", "chla.csv"))
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   action = col_logical(),
##   date = col_date(format = ""),
##   work_area = col_character(),
```

```
##   survey = col_character(),
##   site_id = col_character(),
##   gather_lat = col_logical(),
##   gather_long = col_logical(),
##   collection_method = col_logical(),
##   pressure_transducer_depth = col_logical(),
##   collected = col_datetime(format = ""),
##   preserved = col_datetime(format = ""),
##   analyzed = col_datetime(format = ""),
##   lab_technician = col_character(),
##   project_specific_id = col_character(),
##   hakai_id = col_character(),
##   is_blank = col_logical(),
##   is_solid_standard = col_logical(),
##   filter_size_mm = col_logical(),
##   filter_type = col_character(),
##   calibration = col_datetime(format = "")
##   # ... with 8 more columns
## )
## See spec(...) for full column specifications.
```

```
## Warning: 15289 parsing failures.
##  row         col           expected     actual
## 2627 gather_lat  1/0/T/F/TRUE/FALSE 50.11505   'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop1
## 2627 gather_long 1/0/T/F/TRUE/FALSE -125.22168 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop1
## 2628 gather_lat  1/0/T/F/TRUE/FALSE 50.11505   'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop1
## 2628 gather_long 1/0/T/F/TRUE/FALSE -125.22168 'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop1
## 2629 gather_lat  1/0/T/F/TRUE/FALSE 50.11505   'C:/Users/HP/Documents/R/R projects_Jackie/R-workshop1
## .... ........... .................. .......... ..................................................
## See problems(...) for more details.
```

```
tidy_gsi <- read_csv(here("read_data", "tidy_gsi.csv"))
```

```
## Parsed with column specification:
## cols(
##   hakai_id = col_character(),
##   stock_1 = col_character(),
##   region_1 = col_double(),
##   prob_1 = col_double(),
##   stock_2 = col_character(),
##   region_2 = col_double(),
##   prob_2 = col_double(),
##   stock_3 = col_character(),
##   region_3 = col_double(),
##   prob_3 = col_double(),
##   stock_4 = col_character(),
##   region_4 = col_double(),
##   prob_4 = col_double(),
##   stock_5 = col_character(),
##   region_5 = col_double(),
##   prob_5 = col_double()
## )
```

```
fish %>%
  count(species)
```

```
## # A tibble: 6 x 2
##   species     n
##   <chr>   <int>
## 1 CK         12
## 2 CO         98
## 3 CU       1689
## 4 HE        282
## 5 PI        860
## 6 SO       3497
```

```
fish_d09 <- fish %>%
  filter(site_id == "D09") %>%
  select(hakai_id, jsp_survey_id, seine_id, date, species, site_id, fork_length, weight) %>%
  mutate(k = (10^5 * weight) / fork_length^3) %>%
  drop_na(k)
```

Annoying things that will get you

Factors

```
str(fish_d09)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    832 obs. of  9 variables:
##  $ hakai_id     : chr  "U4802" "U4776" "U4728" "U4801" ...
##  $ jsp_survey_id: chr  "DE112" "DE112" "DE112" "DE112" ...
##  $ seine_id     : chr  "DE112N1" "DE112N1" "DE112N1" "DE112N1" ...
##  $ date         : Date, format: "2015-05-20" "2015-05-20" ...
##  $ species      : chr  "SO" "SO" "SO" "SO" ...
##  $ site_id      : chr  "D09" "D09" "D09" "D09" ...
##  $ fork_length  : num  106 106 97 102 102 97 96 95 128 101 ...
##  $ weight       : num  10.1 11.3 8.8 9.9 8.7 8.4 7.9 8.1 19 9.8 ...
##  $ k            : num  0.848 0.949 0.964 0.933 0.82 ...
```

```
#or

class(fish_d09$species)
```

```
## [1] "character"
```

```
#coerce a column to be a factor

fish_d09$species <- factor(fish_d09$species)

levels(fish_d09$species)
```

```
## [1] "CO" "CU" "HE" "PI" "SO"
```

If you have factors that are numbers, don't try to do maths with these Under the hood R will treat your factor levels as numbers.

Dates

read_csv(): treats ISO date standards (yyyy-mm-dd) as a DATE object read.csv(): treats them as characters - not ideal

Lubridate package - to help with dates

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:here':
##
##     here
```

```
## The following object is masked from 'package:base':
##
##     date
```

```
# Extract data components (day, month, year, Julian day/yday)

fish_d09 <- fish_d09 %>%
  mutate(year = year(date),
         month = month(date),
         week = week(date),
         yday = yday(date))

# to change format of column to DATE

#fish_d09$date <- as.date(fish_d09$date)
```

Can do maths with lubridate

- periods *intervals* durations

Joining data

Data:

chla tidy_gsi *fish_d09

```
# see dplyr cheatsheet for help on this

left_join(fish_d09, tidy_gsi, by = "hakai_id")
```

```
## # A tibble: 832 x 28
##    hakai_id jsp_survey_id seine_id date       species site_id fork_length
##    <chr>    <chr>         <chr>    <date>     <fct>   <chr>         <dbl>
##  1 U4802    DE112         DE112N1  2015-05-20 SO      D09             106
##  2 U4776    DE112         DE112N1  2015-05-20 SO      D09             106
##  3 U4728    DE112         DE112N1  2015-05-20 SO      D09              97
##  4 U4801    DE112         DE112N1  2015-05-20 SO      D09             102
##  5 U4777    DE112         DE112N1  2015-05-20 SO      D09             102
##  6 U4779    DE112         DE112N1  2015-05-20 SO      D09              97
##  7 U4778    DE112         DE112N1  2015-05-20 SO      D09              96
##  8 U4800    DE112         DE112N1  2015-05-20 SO      D09              95
##  9 U4780    DE112         DE112N1  2015-05-20 SO      D09             128
## 10 U350     DE112         DE112N1  2015-05-20 SO      D09             101
## # ... with 822 more rows, and 21 more variables: weight <dbl>, k <dbl>,
## #   year <dbl>, month <dbl>, week <dbl>, yday <dbl>, stock_1 <chr>,
## #   region_1 <dbl>, prob_1 <dbl>, stock_2 <chr>, region_2 <dbl>,
## #   prob_2 <dbl>, stock_3 <chr>, region_3 <dbl>, prob_3 <dbl>,
```

```
## #   stock_4 <chr>, region_4 <dbl>, prob_4 <dbl>, stock_5 <chr>,
## #   region_5 <dbl>, prob_5 <dbl>
```

```
right_join(fish_d09, tidy_gsi, by = "hakai_id")
```

```
## # A tibble: 1,187 x 28
##    hakai_id jsp_survey_id seine_id date       species site_id fork_length
##    <chr>    <chr>         <chr>    <date>     <fct>   <chr>         <dbl>
##  1 U10      <NA>          <NA>     NA         <NA>    <NA>             NA
##  2 U16      <NA>          <NA>     NA         <NA>    <NA>             NA
##  3 U17      <NA>          <NA>     NA         <NA>    <NA>             NA
##  4 U21      <NA>          <NA>     NA         <NA>    <NA>             NA
##  5 U25      <NA>          <NA>     NA         <NA>    <NA>             NA
##  6 U31      <NA>          <NA>     NA         <NA>    <NA>             NA
##  7 U35      <NA>          <NA>     NA         <NA>    <NA>             NA
##  8 U42      <NA>          <NA>     NA         <NA>    <NA>             NA
##  9 U43      <NA>          <NA>     NA         <NA>    <NA>             NA
## 10 U7       <NA>          <NA>     NA         <NA>    <NA>             NA
## # ... with 1,177 more rows, and 21 more variables: weight <dbl>, k <dbl>,
## #   year <dbl>, month <dbl>, week <dbl>, yday <dbl>, stock_1 <chr>,
## #   region_1 <dbl>, prob_1 <dbl>, stock_2 <chr>, region_2 <dbl>,
## #   prob_2 <dbl>, stock_3 <chr>, region_3 <dbl>, prob_3 <dbl>,
## #   stock_4 <chr>, region_4 <dbl>, prob_4 <dbl>, stock_5 <chr>,
## #   region_5 <dbl>, prob_5 <dbl>
```

```
inner_join(fish_d09, tidy_gsi, by = "hakai_id")  #rows that have all required data
```

```
## # A tibble: 147 x 28
##    hakai_id jsp_survey_id seine_id date       species site_id fork_length
##    <chr>    <chr>         <chr>    <date>     <fct>   <chr>         <dbl>
##  1 U350     DE112         DE112N1  2015-05-20 SO      D09             101
##  2 U349     DE112         DE112N1  2015-05-20 SO      D09             104
##  3 U357     DE112         DE112N1  2015-05-20 SO      D09             101
##  4 U355     DE112         DE112N1  2015-05-20 SO      D09              98
##  5 U362     DE112         DE112N1  2015-05-20 SO      D09              89
##  6 U356     DE112         DE112N1  2015-05-20 SO      D09             103
##  7 U363     DE112         DE112N1  2015-05-20 SO      D09             101
##  8 U347     DE112         DE112N1  2015-05-20 SO      D09             102
##  9 U361     DE112         DE112N1  2015-05-20 SO      D09              98
## 10 U319     DE121         DE121N1  2015-05-24 SO      D09             102
## # ... with 137 more rows, and 21 more variables: weight <dbl>, k <dbl>,
## #   year <dbl>, month <dbl>, week <dbl>, yday <dbl>, stock_1 <chr>,
## #   region_1 <dbl>, prob_1 <dbl>, stock_2 <chr>, region_2 <dbl>,
## #   prob_2 <dbl>, stock_3 <chr>, region_3 <dbl>, prob_3 <dbl>,
## #   stock_4 <chr>, region_4 <dbl>, prob_4 <dbl>, stock_5 <chr>,
## #   region_5 <dbl>, prob_5 <dbl>
```

```
anti_join(fish_d09, tidy_gsi, by = "hakai_id")   #rows that DON'T have a match, i.e. NAs
```

```
## # A tibble: 685 x 13
##    hakai_id jsp_survey_id seine_id date       species site_id fork_length
##    <chr>    <chr>         <chr>    <date>     <fct>   <chr>         <dbl>
##  1 U4802    DE112         DE112N1  2015-05-20 SO      D09             106
##  2 U4776    DE112         DE112N1  2015-05-20 SO      D09             106
##  3 U4728    DE112         DE112N1  2015-05-20 SO      D09              97
##  4 U4801    DE112         DE112N1  2015-05-20 SO      D09             102
```

```
##  5 U4777    DE112         DE112N1  2015-05-20 SO      D09             102
##  6 U4779    DE112         DE112N1  2015-05-20 SO      D09              97
##  7 U4778    DE112         DE112N1  2015-05-20 SO      D09              96
##  8 U4800    DE112         DE112N1  2015-05-20 SO      D09              95
##  9 U4780    DE112         DE112N1  2015-05-20 SO      D09             128
## 10 U348     DE112         DE112N1  2015-05-20 SO      D09              94
## # ... with 675 more rows, and 6 more variables: weight <dbl>, k <dbl>,
## #   year <dbl>, month <dbl>, week <dbl>, yday <dbl>
```

```
#view()  displays results for last function (if haven't created it as new df)
```

ggplot2

To build a ggplot:

ggplot(data = DATA, mapping = aes(MAPPINGS)) + GEOM_FUNCTION()

Example:

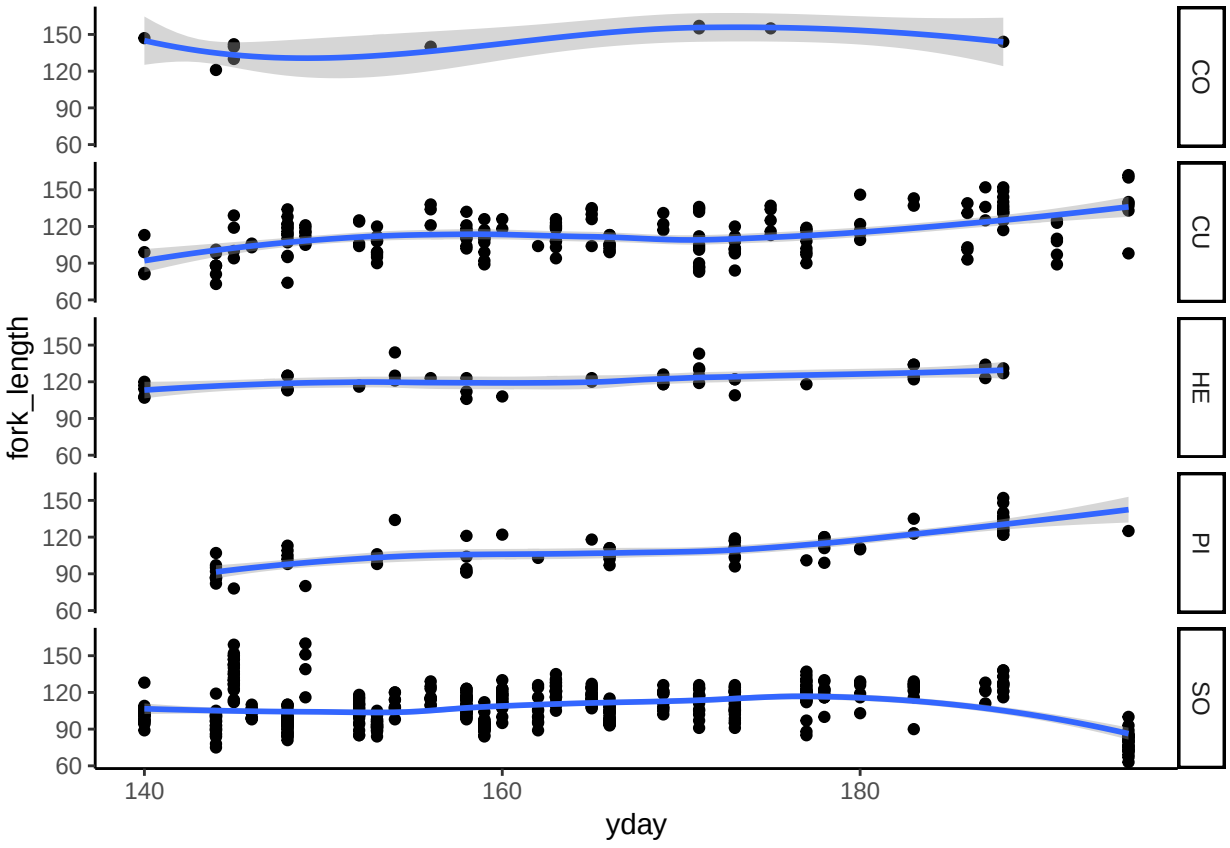
ggplot(data = surveys, mapping = aes(species, weight)) +

#+geom_point()

```
ggplot(fish_d09, aes()) +
  geom_point(aes(x = yday, y = fork_length))+
  geom_smooth(aes(x = yday, y = fork_length), model = lm)+
  theme_classic()+
  facet_grid(species~.)    #separates data by specified variable
```

```
## Warning: Ignoring unknown parameters: model
```

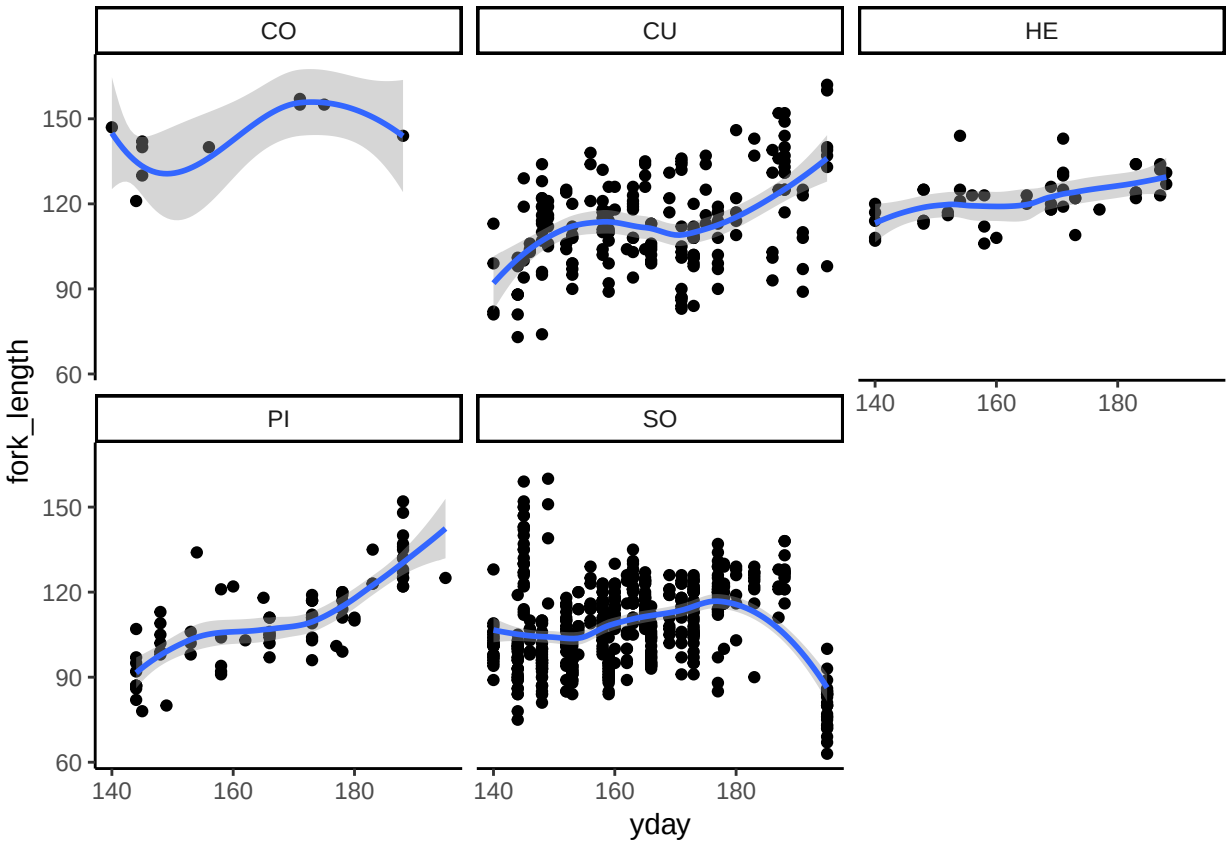
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#or facet_wrap(species~.)

ggplot(fish_d09, aes()) +
  geom_point(aes(x = yday, y = fork_length))+
  geom_smooth(aes(x = yday, y = fork_length), model = lm)+
  theme_classic()+
  facet_wrap(species~.)
```

```
## Warning: Ignoring unknown parameters: model
```

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




```
# check out different bg themes, _bw (gridlines) and _classic (no gridlines)
```

Cookbook for R http://www.cookbook-r.com/

How to adjust legends, axes, etc.

Geom list:

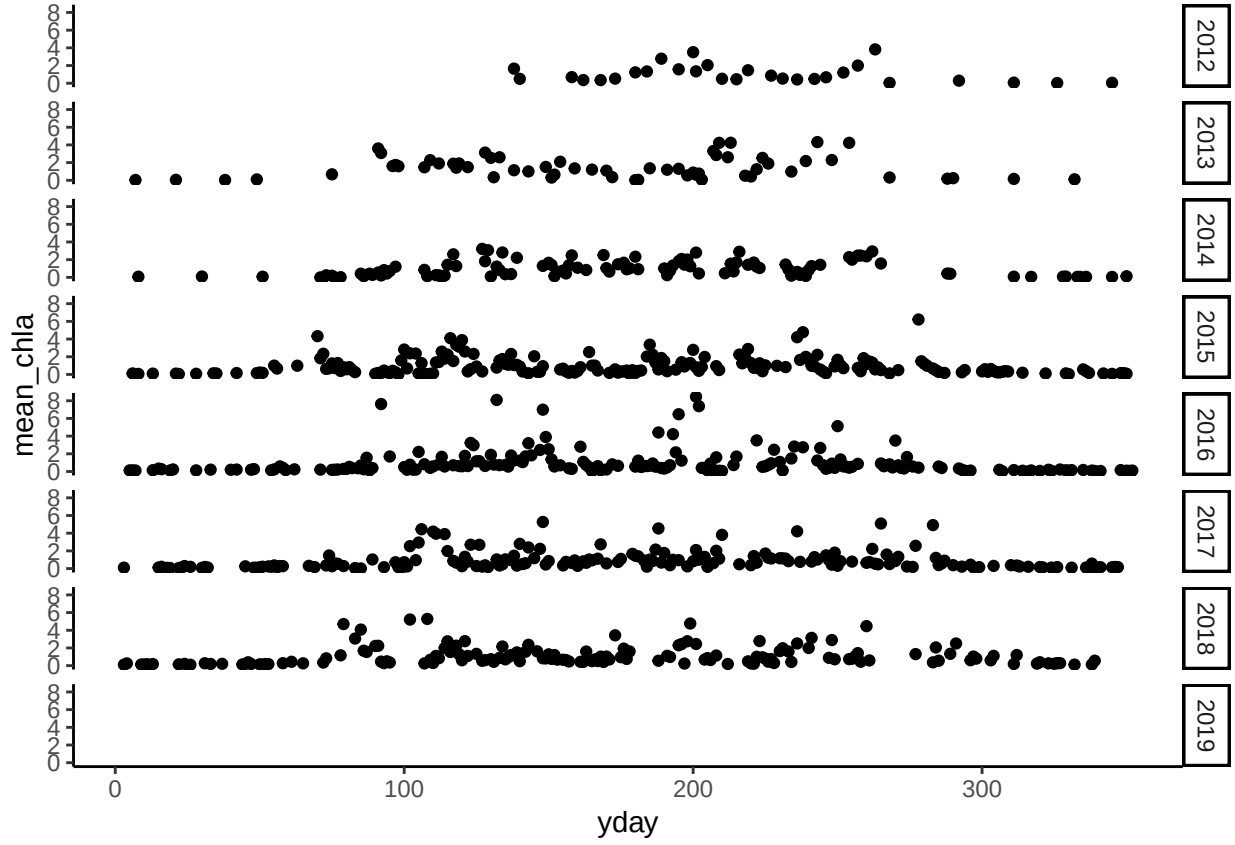
tidyverse.org/reference

https://dplyr.tidyverse.org/reference/index.html

Playtime

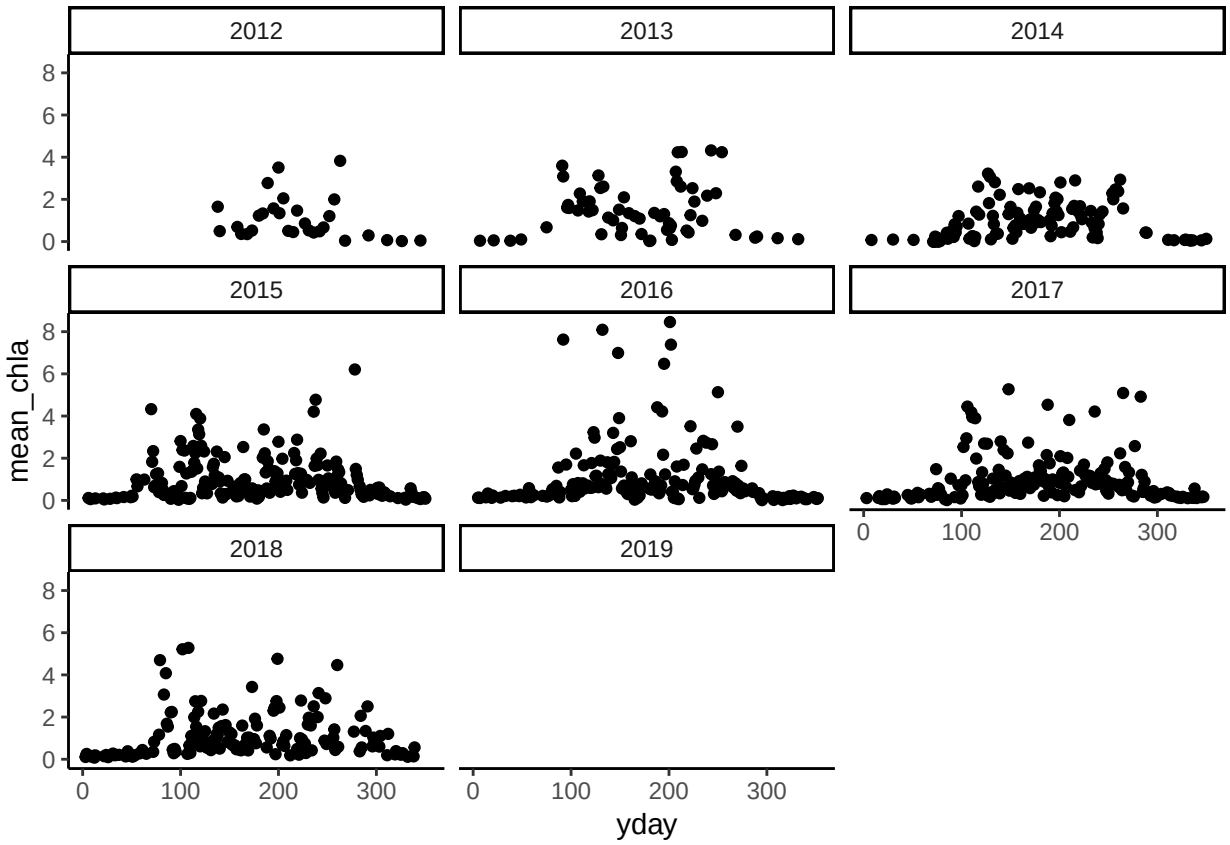
```
chla_date <- chla %>%
    group_by(date) %>%
    summarize(mean_chla = mean(chla, na.rm = TRUE))

chla_date <- chla_date %>%
  mutate(year = year(date),
         month = month(date),
         week = week(date),
         yday = yday(date))


ggplot(chla_date, aes()) +
  geom_point(aes(x = yday, y = mean_chla))+
    theme_classic()+
    facet_grid(year~.)
```

```
## Warning: Removed 51 rows containing missing values (geom_point).
```




```
ggplot(chla_date, aes()) +
  geom_point(aes(x = yday, y = mean_chla))+
    theme_classic()+
    facet_wrap(year~.)
```

```
## Warning: Removed 51 rows containing missing values (geom_point).
```


2012
2013
2014
2015
2016
2017
2018
2019
mean_chla
yday
0
2
4
6
8
0
100
200
300