Tidy Tuesday (10/19/2021)

```
library(tidyverse)
library(tidytuesdayR)
library(ggmap)
```

Data Tidying

Loading in the data and specifying specific data types for each column. `pumpkins` will serve as the main `tibble` for the remainder of the analysis.

```
set.seed(20211019)

tuesdata = tidytuesdayR::tt_load('2021-10-19')
```

```
##
##  Downloading file 1 of 1: `pumpkins.csv`
```

```
pumpkins = tuesdata$pumpkins %>%
  select(-variety) %>%
  mutate(
    place = as.integer(place),
    weight_lbs = as.double(weight_lbs),
    grower_name = as.factor(grower_name),
    city = as.factor(city),
    state_prov = as.factor(state_prov),
    country = as.factor(country),
    gpc_site = as.factor(gpc_site),
    seed_mother = as.factor(seed_mother),
    pollinator_father = as.factor(pollinator_father),
    ott = as.double(ott),
    est_weight = as.double(est_weight),
    pct_chart = as.double(pct_chart)
  )
```

Top Seed Mothers

Which seed mothers are best at producing the highest-placed pumpkins? Using `seed_mother`s that appear more than 26 times (number chosen solely for maximal plotting information), we notice that most `seed_mother`s follow a similar trend: the higher the weight, the greater the place (observed here as a negative relationship). Growers should look for seeds that provide higher places more consistently than other seeds. Seeds such as `211 MacKinnon`, `302 Kent`, and `316 Edwards` appear to almost guarantee a high position if the pumpkin exceeds a certain weight, whereas other seeds like `2145 McMullen`, `1985 Miller` and `1756 Lancaster` show a linear relationship between weight and placement.

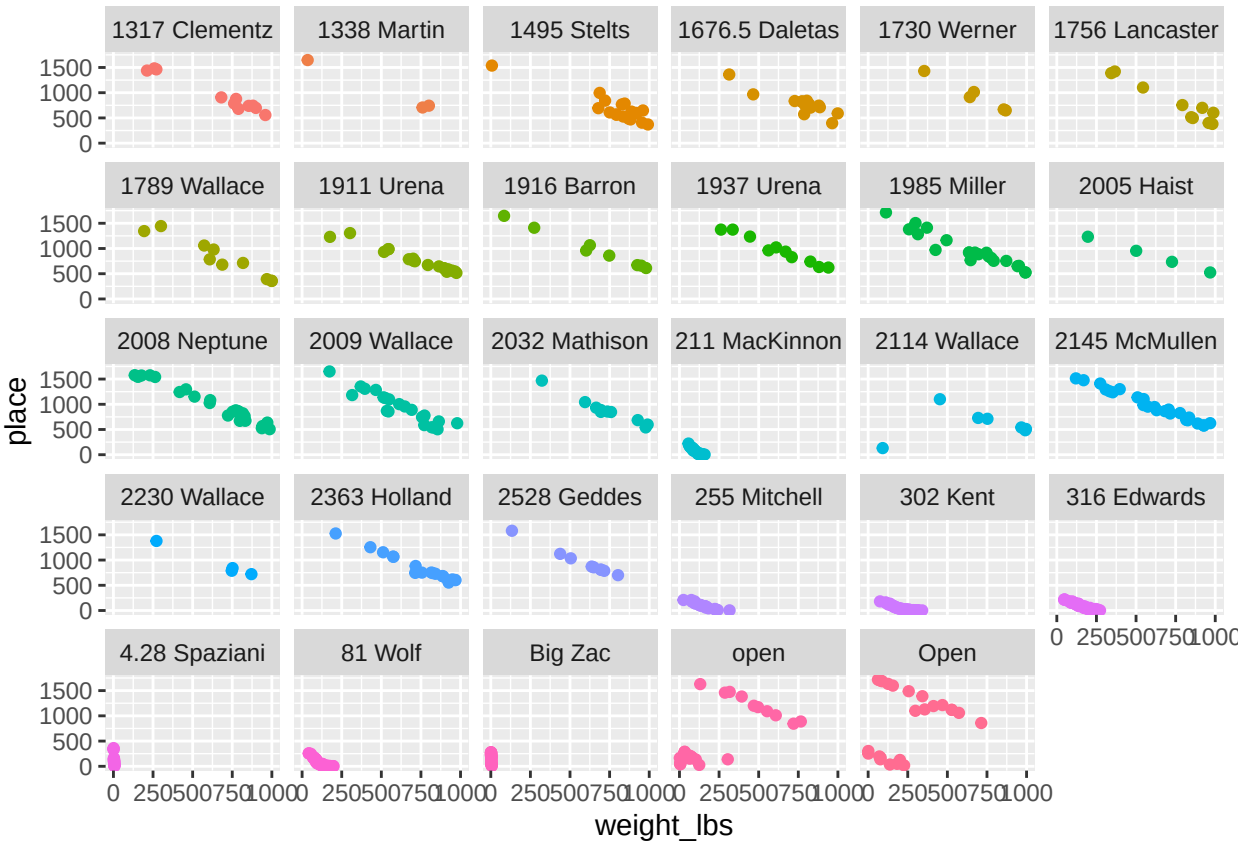
```
top_seed_mothers = pumpkins %>%
  select(place, weight_lbs, city, state_prov, gpc_site, seed_mother, pollinator_father) %>%
  filter((!is.na(seed_mother)) & (seed_mother != 'Unknown') & (seed_mother != 'unknown')) %>%
  group_by(seed_mother) %>%
  summarize(n = n()) %>%
  filter(n > 26) %>%
  arrange(desc(n))

top_seed_mothers
```

```
## # A tibble: 29 x 2
##    seed_mother       n
##    <fct>         <int>
##  1 2145 McMullen   122
##  2 2009 Wallace    104
##  3 1985 Miller      89
##  4 1911 Urena       72
##  5 2363 Holland     61
##  6 81 Wolf          61
##  7 2008 Neptune     60
##  8 1495 Stelts      52
##  9 316 Edwards      47
## 10 1317 Clementz    42
## # ... with 19 more rows
```

```
pumpkins %>%
  filter(seed_mother %in% top_seed_mothers$seed_mother) %>%
  ggplot(aes(x=weight_lbs, y=place, colour=seed_mother)) +
    geom_point(position='jitter', show.legend=FALSE) +
    facet_wrap(~ seed_mother)
```

```
## Warning: Removed 783 rows containing missing values (geom_point).
```


1317 Clementz
1338 Martin
1495 Stelts
1676.5 Daletas
1730 Werner
1756 Lancaster
1789 Wallace
1911 Urena
1916 Barron
1937 Urena
1985 Miller
2005 Haist
2008 Neptune
2009 Wallace
2032 Mathison
211 MacKinnon
2114 Wallace
2145 McMullen
2230 Wallace
2363 Holland
2528 Geddes
255 Mitchell
302 Kent
316 Edwards
4.28 Spaziani
81 Wolf
Big Zac
open
Open
place
weight_lbs
0
500
1000
1500
2000
2500

Top Region Producers

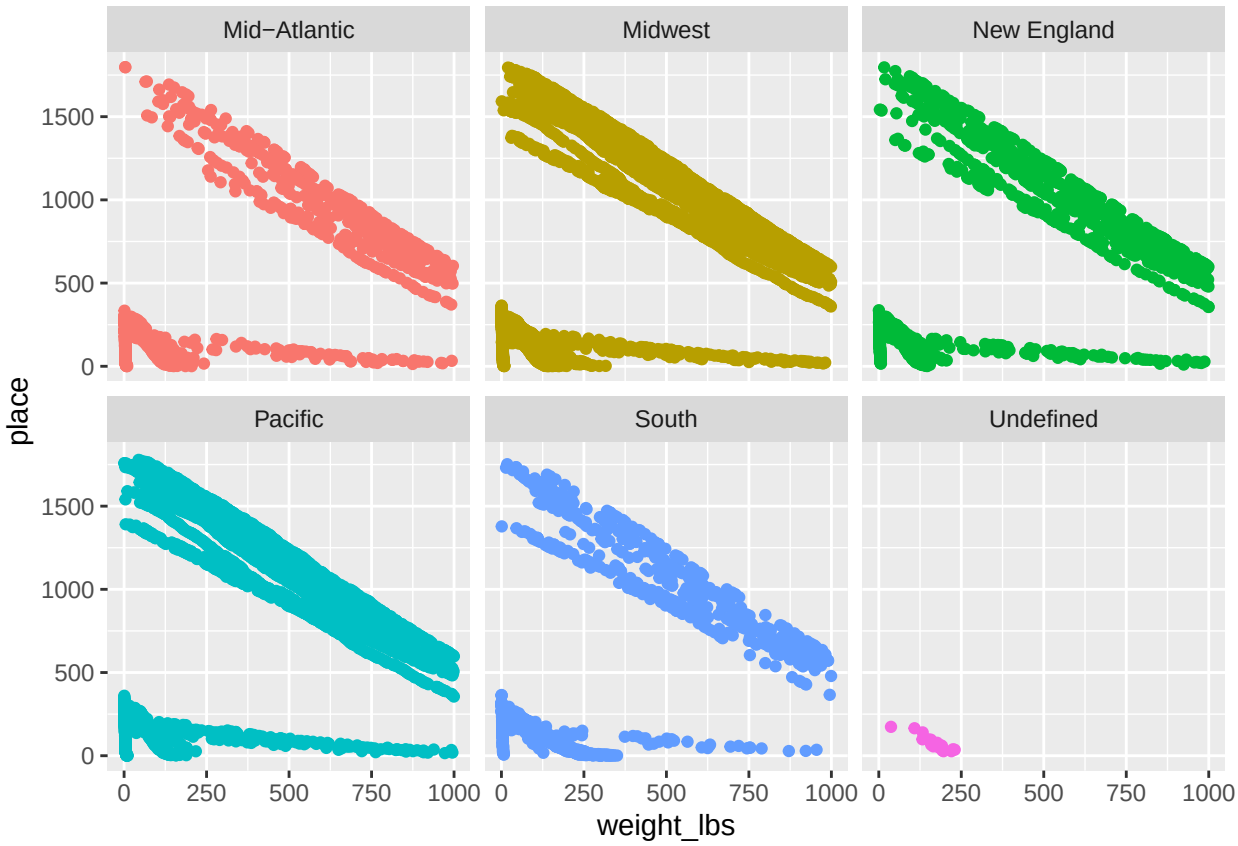
Do specific regions produce better than others? If up to me, I'd rather have `latitude-longitude` data - which is totally possible, I just couldn't find a convenient solution that provided access to `lat-long` given `city-state` pairings (thanks, Google). However, we notice very similar (almost identical) trends between all regions, implying that competitions are relatively fair across all regions (i.e. one does not gain an advantage competing in one region than they do any other region).

```
new_england = c('Connecticut', 'Maine', 'Massachusetts', 'New Hampshire',
                'Rhode Island', 'Vermont')
mid_atlantic = c('New Jersey', 'New York', 'Pennsylvania')
east_north_central = c('Illinois', 'Indiana', 'Michigan', 'Ohio',
                       'Wisconsin')
west_north_central = c('Iowa', 'Kansas', 'Minnesota', 'Missouri', 'Nebraska',
                       'North Dakota', 'South Dakota')
south_atlantic = c('Delaware', 'Florida', 'Georgia', 'Maryland',
                   'North Carolina', 'South Carolina', 'Virginia',
                   'Washington D.C.', 'West Virginia')
east_south_central = c('Alabama', 'Kentucky', 'Mississippi', 'Tennessee')
west_south_central = c('Arkansas', 'Lousiana', 'Oklahoma', 'Texas')
mountain = c('Arizona', 'Colorado', 'Idaho', 'Montana', 'Nevada', 'New Mexico', 'Utah', 'Wyoming')
pacific = c('Alaska', 'California', 'Hawaii', 'Oregon', 'Washington')

geo_pumps = pumpkins %>%
  filter(country == 'United States') %>%
  select(-c('grower_name', 'country', 'gpc_site', 'seed_mother', 'pollinator_father')) %>%
  mutate(region = ifelse(state_prov %in% new_england, 'New England',
                  ifelse(state_prov %in% mid_atlantic, 'Mid-Atlantic',
                  ifelse(state_prov %in% east_north_central, 'Midwest',
                  ifelse(state_prov %in% west_north_central, 'Midwest',
                  ifelse(state_prov %in% south_atlantic, 'South',
                  ifelse(state_prov %in% east_south_central, 'South',
                  ifelse(state_prov %in% west_south_central, 'South',
                  ifelse(state_prov %in% mountain, 'Pacific',
                  ifelse(state_prov %in% pacific, 'Pacific', 'Undefined'))))))))
         )

geo_pumps %>%
  ggplot() +
  geom_point(aes(x=weight_lbs, y=place, colour=region),
             position='jitter', show.legend=FALSE) +
  facet_wrap(~ region)
```

```
## Warning: Removed 4960 rows containing missing values (geom_point).
```


Mid−Atlantic
Midwest
New England
Pacific
South
Undefined
place
1500
1000
500
0
0
250
500
750
1000
weight_lbs

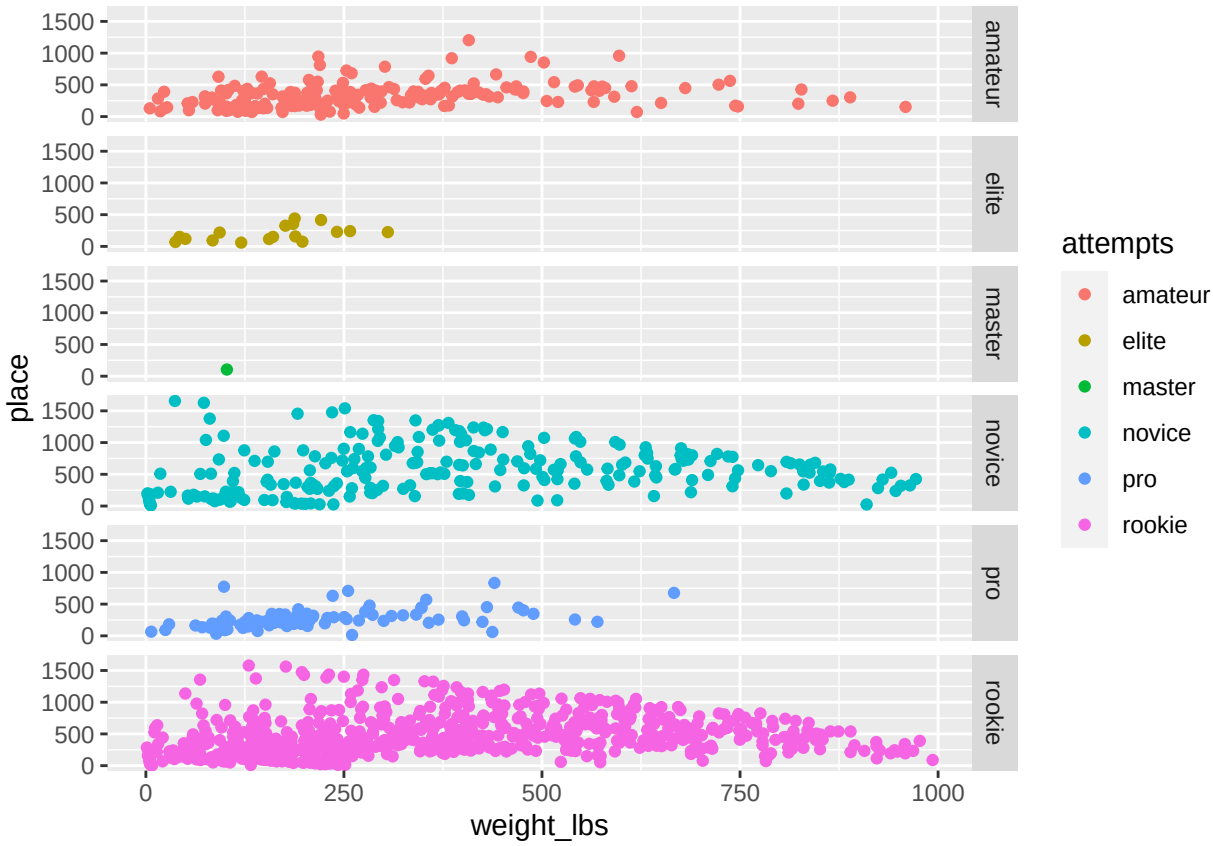
Repeat Offenders

How well do specific groups (based on number of attempts) perform? It seems like the more attempts one takes, the better off they fair, even if their pumpkins are of less weight. Granted there is increasingly less data supporting this per group, it is worth noting that the relationship between `weight_lbs` and `place` changes per group (where relationships can be clearly observed).

```
repeat_offenders = pumpkins %>%
  group_by(grower_name) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  filter(n >= 5)

pumpkins %>%
  filter(grower_name %in% repeat_offenders$grower_name) %>%
  group_by(grower_name) %>%
  summarize(
    place = mean(place, na.rm=TRUE),
    weight_lbs = mean(weight_lbs, na.rm=TRUE),
    count = n()
  ) %>%
  mutate(attempts = ifelse(count == max(count), 'master',
                    ifelse(max(count) >= count & count > 60, 'elite',
                    ifelse(60 >= count & count > 30, 'pro',
                    ifelse(30 >= count & count > 15, 'amateur',
                    ifelse(15 >= count & count > 5, 'rookie', 'novice')))))) %>%
  ggplot(aes(x=weight_lbs, y=place, colour=attempts)) +
    geom_point() +
    facet_grid(attempts ~ .)
```

```
## Warning: Removed 35 rows containing missing values (geom_point).
```


place
1500
1000
500
0
amateur
elite
master
novice
pro
rookie
0
250
500
750
1000
weight_lbs
attempts
amateur
elite
master
novice
pro
rookie