Summarize and Plot

Jameson Watts, Ph.D.

10/3/2019

Agenda

1. Expectations for the mid-term
2. Review (and upgrade)
3. Visualization techniques

Mid-term

Expectation and format

1. 45 minutes of review
2. 120 minutes to complete
3. Must be submitted as both Rmd file and knitted HTML
4. Open everything (notes, book, internet)… except communication
5. Questions will get progressively more difficult

Review

Joining, Mutation, and Strings

First things first, let's load in my new data on monthly rainfall and take a look. What do you notice?

```
rain <- read_csv("../resources/rainfall.csv")
rain
```

```
## # A tibble: 48 x 13
##     Year   Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1  1970 13.5   4.46  1.92  2.63  1.36  0.85  0.01 NA     1.81  3.25  7.18
##  2  1971  6.49  4.34  6.93  4.05  1.89  2.47  0.01  1.49  3.98  3.09  6.27
##  3  1972  7.98  4.68  4.96  3.79  2.4   0.69  0.12  0.14  2.07  0.7   3.77
##  4  1973  5.64  1.62  3.5   1.69  1.11  1.48 NA     0.8   2.8   2.79 15.2
##  5  1974 10.9   5.56  7.95  1.48  0.9   0.41  1.8   0.11  0.28  2.15  7.42
##  6  1975  4.96  4.68  4.22  2.2   1.66  0.81  0.51  1.96  0     5.51  6.06
##  7  1976  5.47  6.92  3.66  2     1.33  1.04  0.67  1.89  1.13  1.51  1.13
##  8  1977  0.88  2.83  3.33  0.62  3.76  0.73  0.26  1.7   2.36  2.37  6.19
##  9  1978  5.67  3.54  1.23  3.5   2.97  0.48  1.07  2.56  2.64  0.37  4.5
## 10  1979  2.84  7.19  2.17  2.82  2.2   0.65  0.3   0.7   2.19  6.06  3.83
## # … with 38 more rows, and 1 more variable: Dec <dbl>
```

Tidyr 1.0.0

Tidyr has replaced spread() and gather() with pivot_wider() and pivot_longer(). I encourage you to read about the developments here.

```
rain %>%
  rename("year"="Year") %>%
  pivot_longer(-year,names_to = "month", values_to = "rainfall")
```

```
## # A tibble: 576 x 3
##     year month rainfall
##    <dbl> <chr>    <dbl>
##  1  1970 Jan      13.5
##  2  1970 Feb       4.46
##  3  1970 Mar       1.92
##  4  1970 Apr       2.63
##  5  1970 May       1.36
##  6  1970 Jun       0.85
##  7  1970 Jul       0.01
##  8  1970 Aug      NA
##  9  1970 Sep       1.81
## 10  1970 Oct       3.25
## # … with 566 more rows
```

Exercise

1. Load in the wine data
2. Get rid of prices that are NA
3. Only keep Oregon wines
4. Extract the year from the title (as numeric)
5. Join with rainfall data
6. Pivot longer

Solution

Note: See here for an intuitive tutorial on joins.

```
wine <- read_csv("../resources/winemag-data.csv") %>%
  filter(!is.na(price)) %>%
  filter(province=="Oregon") %>%
  mutate(year = as.numeric(str_extract(title,"(\\d{4})"))) %>%
  left_join(rain, by=c("year"="Year")) %>%
  pivot_longer(16:27,names_to = "month", values_to = "rainfall")

wine %>%
  select(title, month, year, rainfall)
```

```
## # A tibble: 64,308 x 4
##    title                                          month  year rainfall
##    <chr>                                          <chr> <dbl>    <dbl>
##  1 Rainstorm 2013 Pinot Gris (Willamette Valley) Jan    2013     1.63
##  2 Rainstorm 2013 Pinot Gris (Willamette Valley) Feb    2013     1.42
##  3 Rainstorm 2013 Pinot Gris (Willamette Valley) Mar    2013     2.21
##  4 Rainstorm 2013 Pinot Gris (Willamette Valley) Apr    2013     2.39
##  5 Rainstorm 2013 Pinot Gris (Willamette Valley) May    2013     2.94
##  6 Rainstorm 2013 Pinot Gris (Willamette Valley) Jun    2013     1.02
##  7 Rainstorm 2013 Pinot Gris (Willamette Valley) Jul    2013     0
##  8 Rainstorm 2013 Pinot Gris (Willamette Valley) Aug    2013     0.35
##  9 Rainstorm 2013 Pinot Gris (Willamette Valley) Sep    2013     7.05
## 10 Rainstorm 2013 Pinot Gris (Willamette Valley) Oct    2013     0.63
## # … with 64,298 more rows
```

Skills upgrade: all about case_when()

Sometimes you want to do a bunch of if/else in your mutate all at once.

```
wine %>%
  mutate(month_number =
    case_when(
      month=="Jan" ~ 1,
      month=="Feb" ~ 2,
      month=="Mar" ~ 3,
      month=="Apr" ~ 4,
      month=="May" ~ 5,
      month=="Jun" ~ 6,
      month=="Jul" ~ 7,
      month=="Aug" ~ 8,
      month=="Sep" ~ 9,
      month=="Oct" ~ 10,
      month=="Nov" ~ 11,
      month=="Dec" ~ 12,
    )
  )
```

Exercise

1. Partner up and choose a driver
2. Use case_when() and/or str_detect() with regular expressions
3. To create a new variable called “character”
4. With values of ‘tart,’ ‘spicy,’ ‘bold’ and ‘cherry’
5. For Oregon wines with those terms in their description
6. Then plot the density of log(price) by character

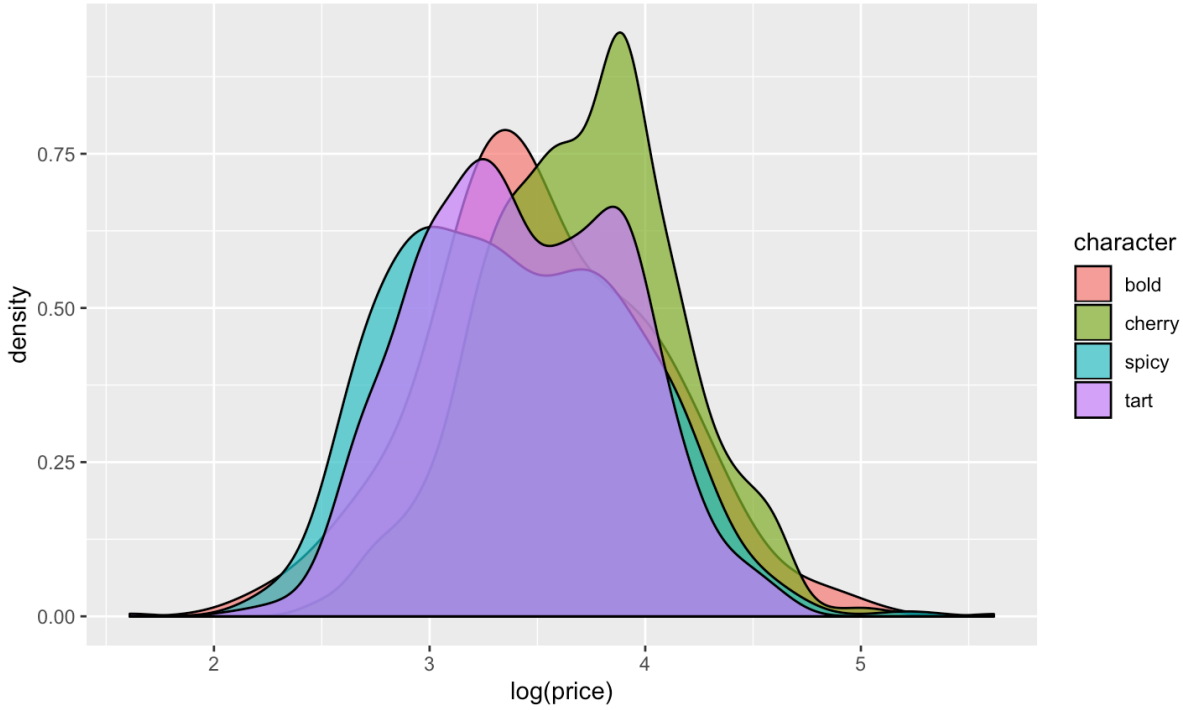
Hint: you may want to pivot_wider() first

Solution (code)

```
wine <- wine %>%
  pivot_wider(names_from = month, values_from = rainfall) %>%
  mutate(character=
    case_when(
      str_detect(description,"[Tt]art") ~ 'tart',
      str_detect(description,"[Sp]icy") ~ 'spicy',
      str_detect(description,"[Bb]old") ~ 'bold',
      str_detect(description,"[Cc]herry") ~ 'cherry'
    )
  )
```

Solution (graph)

```
wine %>%
  filter(!is.na(character)) %>%
  ggplot(aes(log(price), fill=character))+
    geom_density(alpha=.7)
```



Visualization basics

Overview

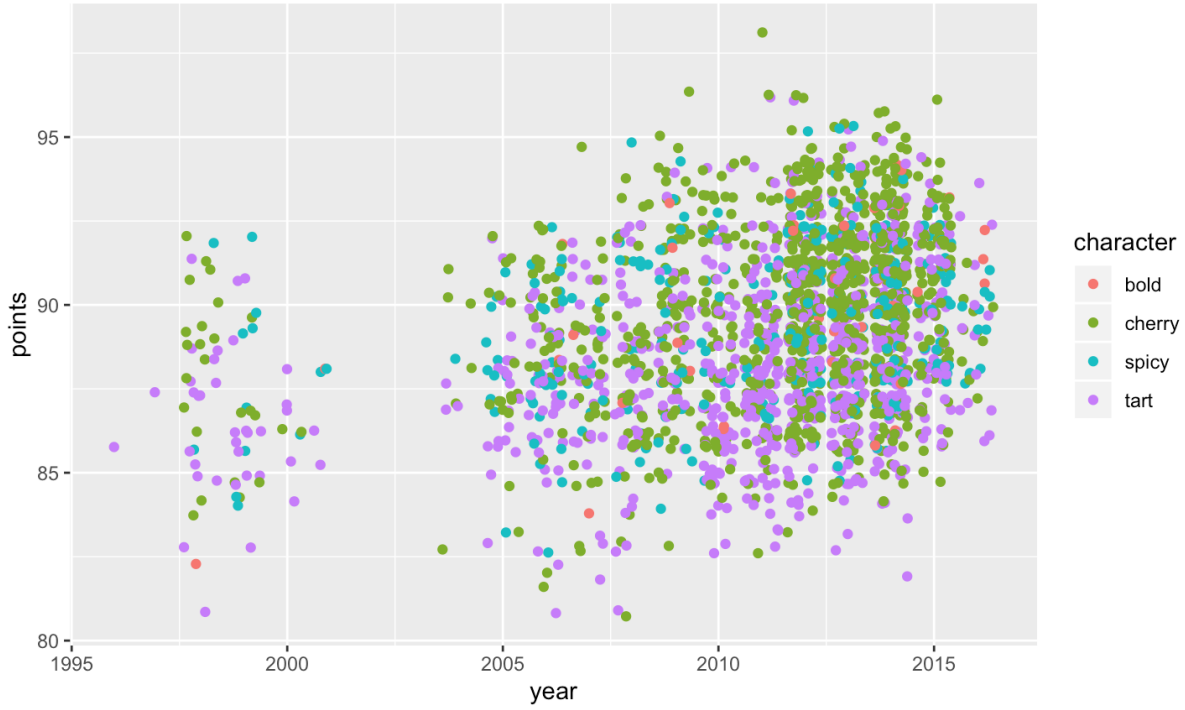
- Aesthetics
  - x =
  - y =
  - fill =
  - color =
- Geometry
  - Line plots
  - Bar plots
  - Histograms
  - Violin plots

Types of plots

- Line and scatter
  - geom_point()
  - geom_jitter()
  - geom_line()
- Bar
  - geom_bar()
  - geom_col()
- Histograms
  - geom_histogram()
  - geom_density()
- Box
  - geom_box()
  - geom_voilin()

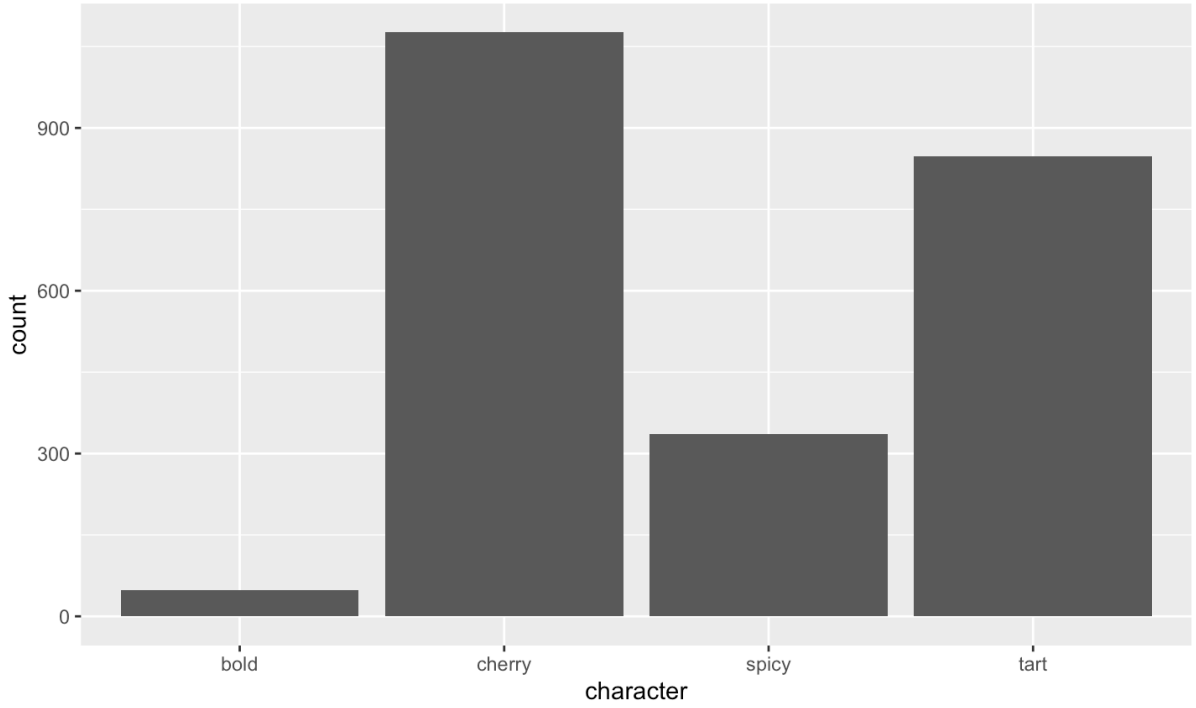
Scatter plot example

```
wine %>%
  filter(year>1995) %>%
  filter(!is.na(character)) %>%
  ggplot(aes(x=year, y=points, color=character)) +
    geom_jitter()
```



Bar plot example

```
wine %>%
  filter(!is.na(character)) %>%
  ggplot(aes(character))+
    geom_bar()
```

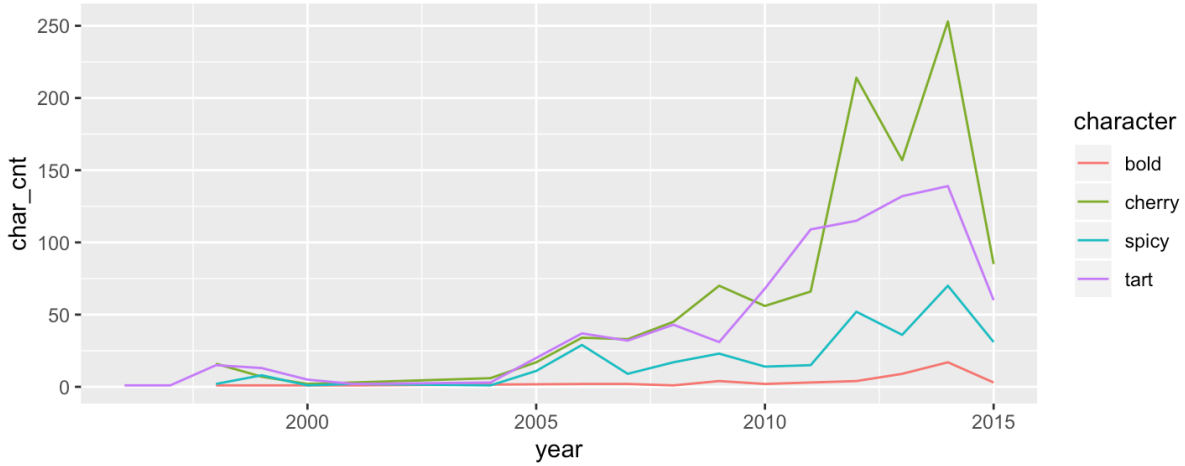


Exercise

Plot the counts of each character of wine between 1995 and 2015

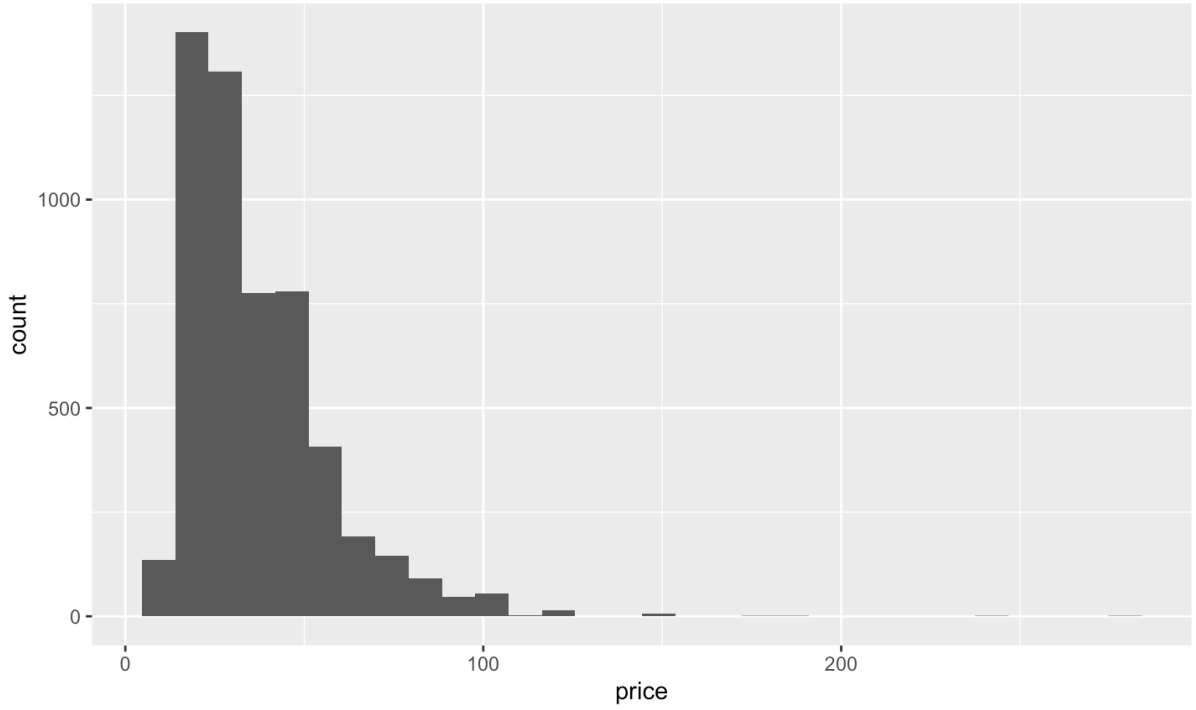
Solution

```
wine %>%
  filter(year>1995 & year <= 2015) %>%
  filter(!is.na(character)) %>%
  group_by(year,character) %>%
  summarise(char_cnt=n()) %>%
  ggplot(aes(year,char_cnt, color=character))+
    geom_line()
```



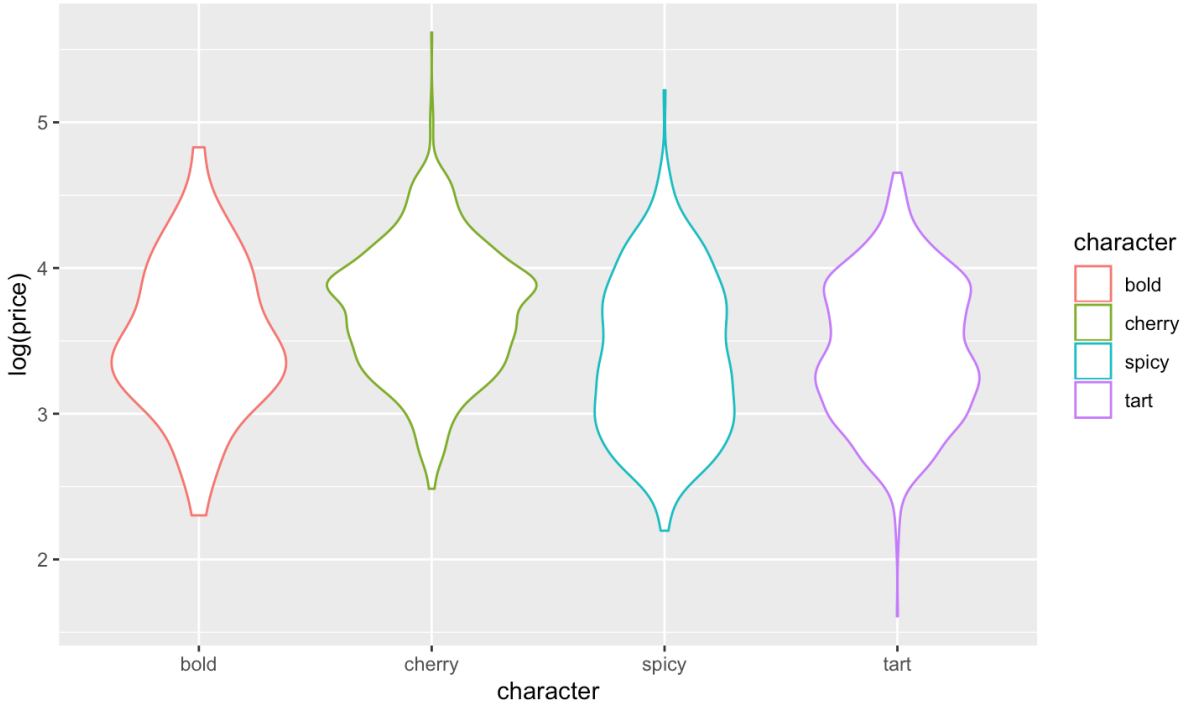
Histogram example

```
wine %>%
  ggplot(aes(price))+
    geom_histogram()
```



Violin Plots

```
wine %>%
  filter(!is.na(character)) %>%
  ggplot(aes(character,log(price), color=character))+
    geom_violin()
```



Long Exercise

Use any of the techniques that you've learned thus far to answer the following:

Is there a relationship between rainfall and wine quality in Oregon?

One simple solution (and some bonus code)

```
rains <- rain %>%
  rename("year"="Year") %>%
  pivot_longer(-year,names_to = 'month',values_to = 'rainfall') %>%
  mutate(rainfall=ifelse(is.na(rainfall),0,rainfall)) %>%
  filter(month %in% c('May','Jun','Jul','Aug','Sep')) %>% #note the %in% operator
  group_by(year) %>%
  summarise(summer_rain=sum(rainfall))

wines <- wine %>%
  filter(points > 88) %>%
  group_by(year) %>%
  summarize(avg_price=mean(price), avg_points=mean(points)) %>%
  left_join(rains)
```

And a graph

```
wines %>%
  ggplot(aes(log(summer_rain),avg_points))+
    geom_point()+
    geom_smooth(method = lm)
```