Graphics with ggplot2

STAT 220

Bastola
January 16 2022

Layered Grammar of Graphics

- Essentials
    - Data
    - **Aesthetic mappings**
    - **Geometric objects**

- Additional elements
    - Facets
    - **Coordinate system**
    - **Statistical transformations**
    - **Position adjustments**
    - Scales
    - Theme

Common **ggplot2** options

```
ggplot(data) +    # data
  <geometry_funs>(aes(<variables>)) + # aesthetic variable mapping
  <label_funs> +  # add context
  <facet_funs> +  # add facets (optional)
  <coordinate_funs> +  # play with coords (optional)
  <scale_funs> + # play with scales (optional)
  <theme_funs> # play with axes, colors, etc (optional)
```

- See the Rstudio cheatsheets for more details

Hate Crime and income inequality

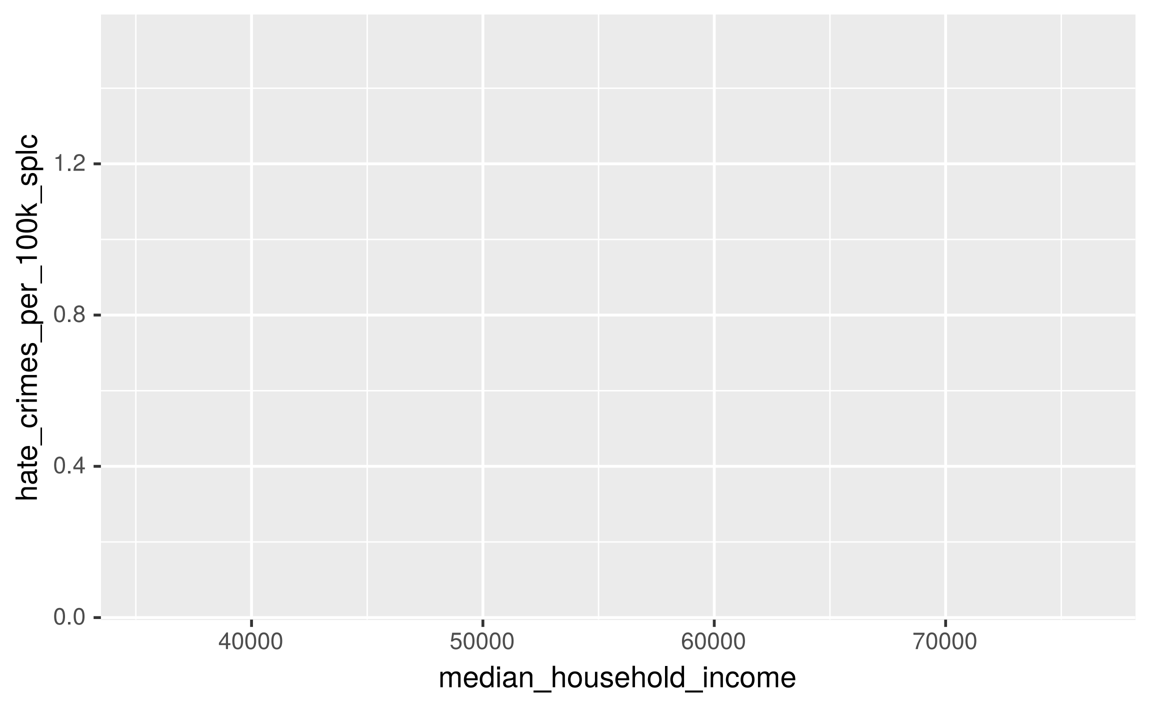
A FiveThirtyEight article published in 2017 claimed that higher rates of hate crimes were tied to greater income inequality.

```
glimpse(hate_crimes)
Rows: 51
Columns: 17
$ X                                        <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10…
$ state                                    <chr> "Alabama", "Alaska", "Arizona…
$ median_household_income                  <int> 42278, 67629, 49254, 44922, 6…
$ share_unemployed_seasonal                <dbl> 0.060, 0.064, 0.063, 0.052, 0…
$ share_population_in_metro_areas          <dbl> 0.64, 0.63, 0.90, 0.69, 0.97,…
$ share_population_with_high_school_degree <dbl> 0.821, 0.914, 0.842, 0.824, 0…
$ share_non_citizen                        <dbl> 0.02, 0.04, 0.10, 0.04, 0.13,…
$ share_white_poverty                      <dbl> 0.12, 0.06, 0.09, 0.12, 0.09,…
$ gini_index                               <dbl> 0.472, 0.422, 0.455, 0.458, 0…
$ share_non_white                          <dbl> 0.35, 0.42, 0.49, 0.26, 0.61,…
$ share_voters_voted_trump                 <dbl> 0.63, 0.53, 0.50, 0.60, 0.33,…
$ hate_crimes_per_100k_splc                <dbl> 0.12583893, 0.14374012, 0.225…
$ avg_hatecrimes_per_100k_fbi              <dbl> 1.8064105, 1.6567001, 3.41392…
$ state_code                               <chr> "AL", "AK", "AZ", "AR", "CA",…
$ region                                   <chr> "South", "West", "West", "Sou…
$ division                                 <chr> "East South Central", "Pacifi…
$ support                                  <chr> "Trump", "Trump", "Split", "T…
```

Layering geoms

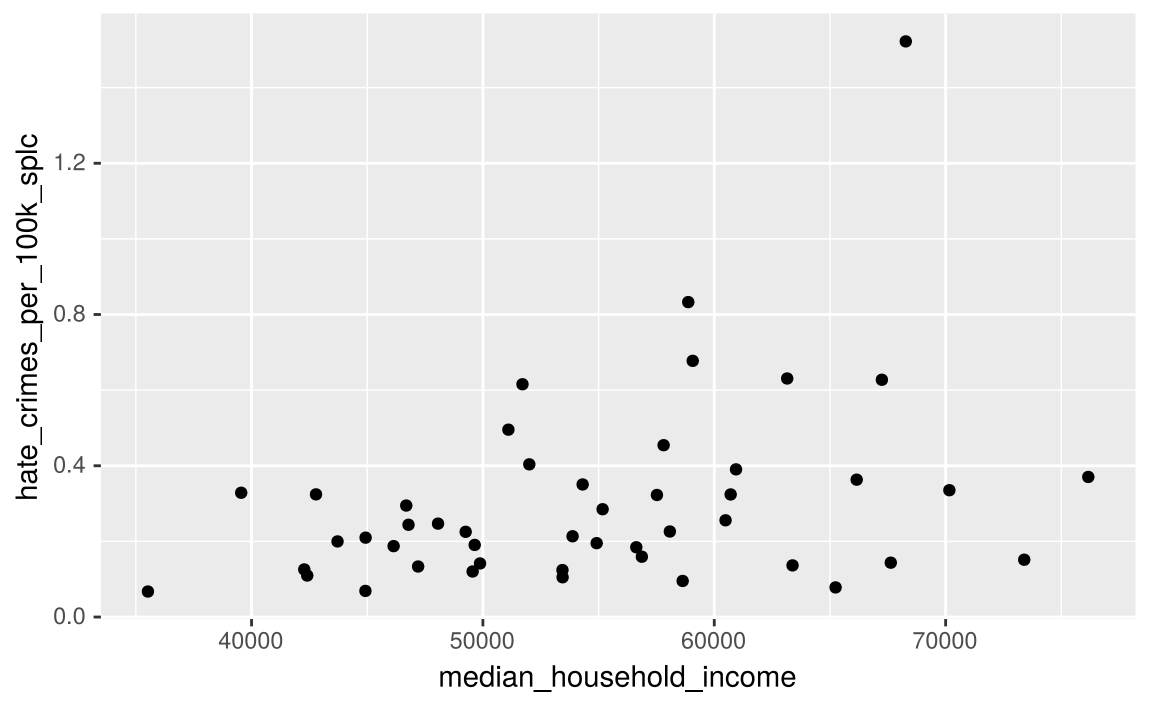
When you are iteratively building plots, it's useful to store the base plot as an object

```
base <- ggplot(hate_crimes, aes(x=median_household_income, y=hate_crimes_per_100k_splc))
base
```



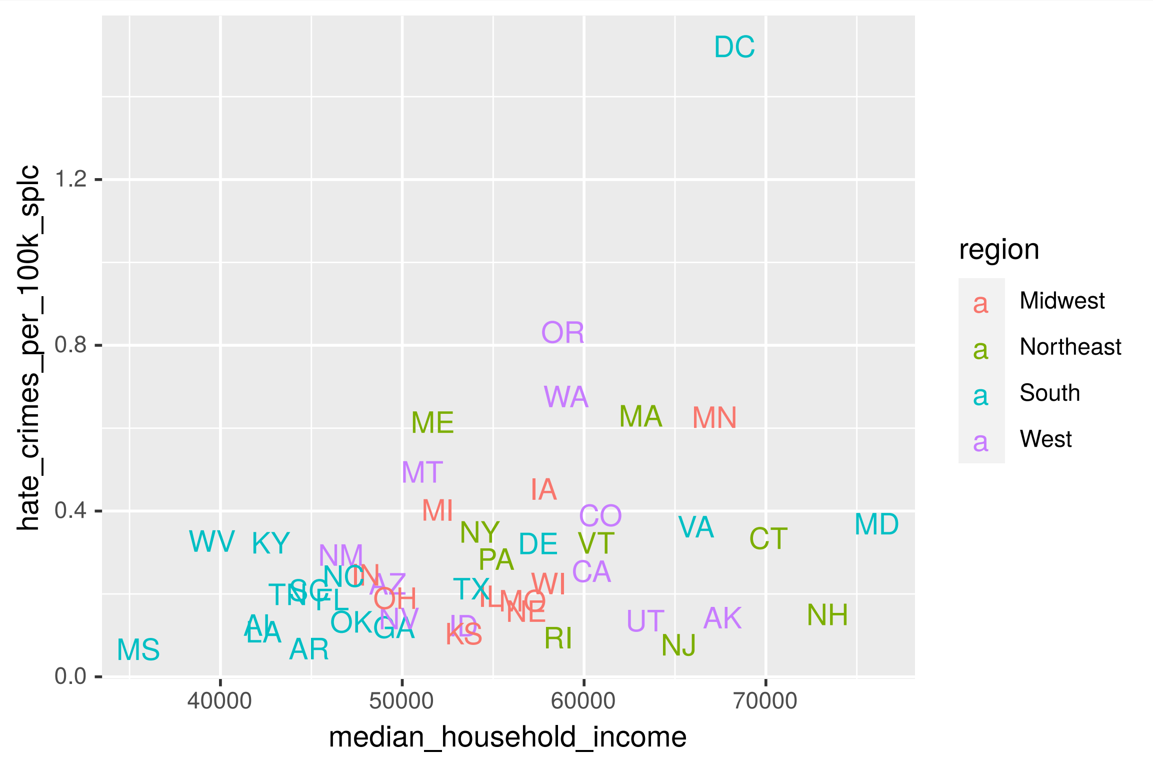
Layering geoms

```
base +
  geom_point()
```



A better plot

```
base +
  geom_text(aes(label=state_code, color=region))
```



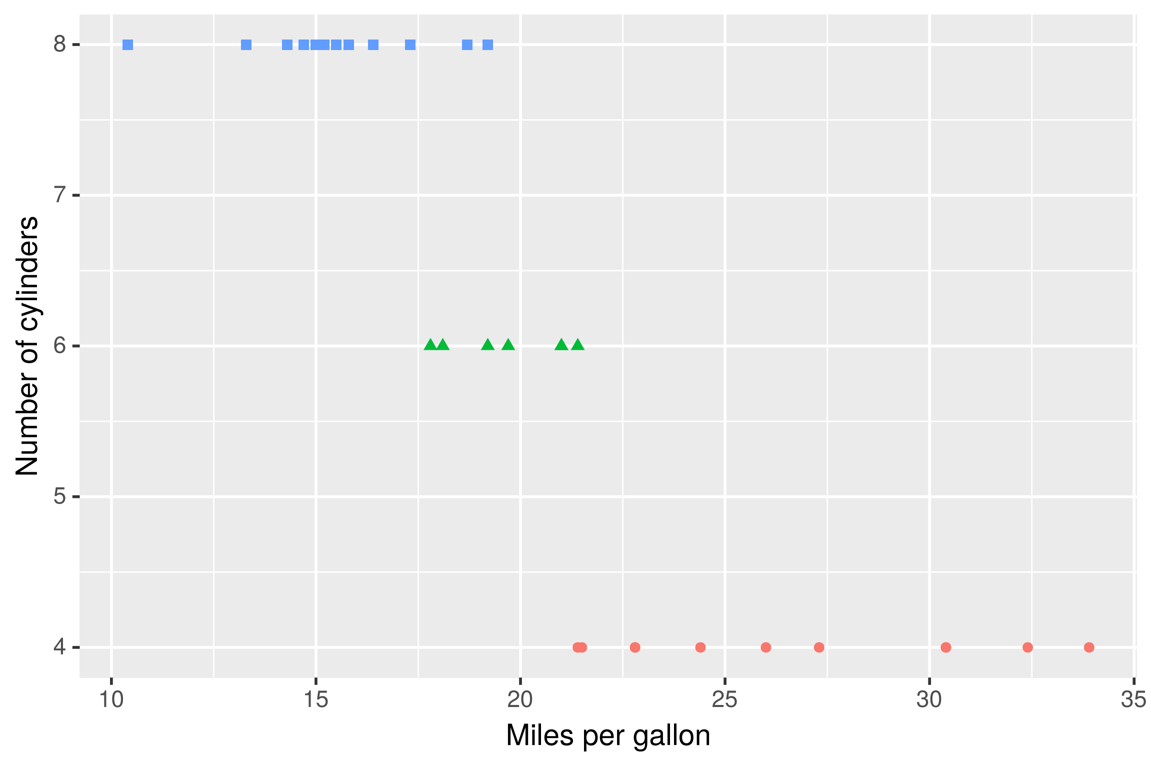
Labeling your graphics

In `ggplot2` you can add/change the title, subtitle, caption, and x- and y-axis labels by adding a `labs()` layer. Below is an example illustrating it’s use:

```
ggplot(data = mpg) +
geom_point(mapping = aes(x = displ, y = hwy)) + labs(
    title = "Put your informative title here",
    subtitle = "and your subtitle here",
    x = "New x label",
    y = "New y label",
    caption = "Put a caption here"
  )
```

Plotting variables

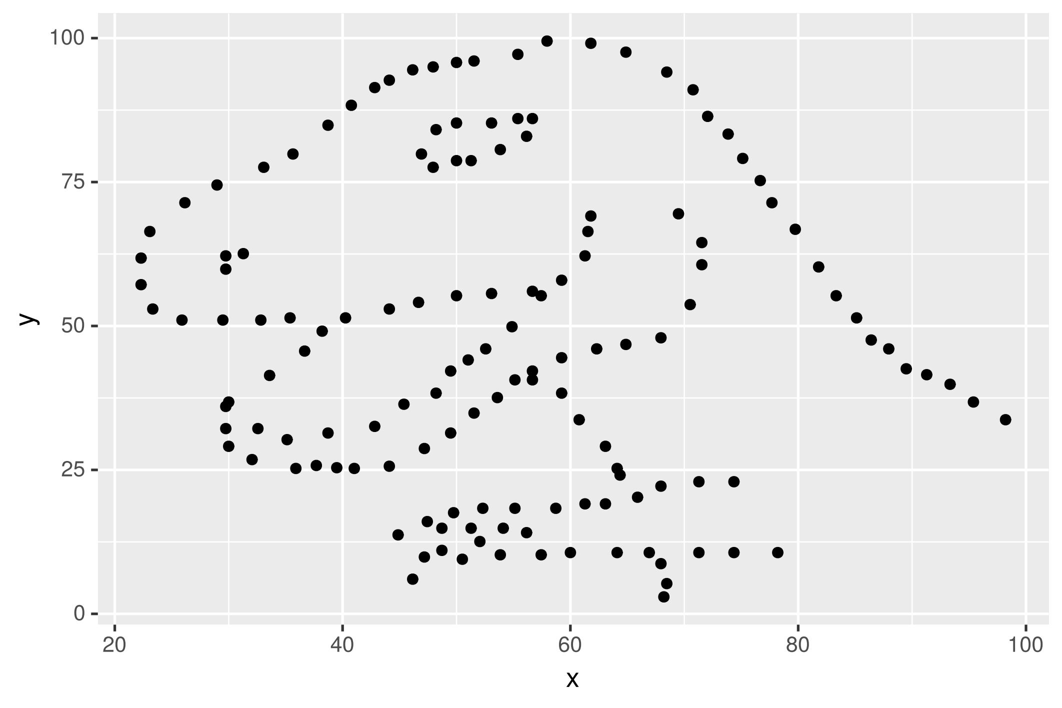
```
ggplot(data = mtcars, aes(x=mpg, y=cyl)) +
  geom_point(aes(col=as.factor(cyl), pch=as.factor(cyl))) +
  labs(x='Miles per gallon', y='Number of cylinders') +
  guides(col=FALSE, pch=FALSE)
```



Your Turn 1

Please git clone the GitHub repository `03-visualizations`.

The data frame in this exercise is called `datasaurus_dozen` and it's in the `datasauRus` package. This single data frame contains 13 datasets.



05:00

Statistical transformations: Default stats

Common geom	stat
`geom_histogram()`	`stat_bin()`
`geom_bar()`	`stat_count()`
`geom_smooth()`	`stat_smooth()`
`geom_boxplot()`	`stat_boxplot()`
`geom_density()`	`stat_density()`

- Every geom has a default stat
- Often no need to explicitly specify the stat
- Check help files: `?geom_bar`

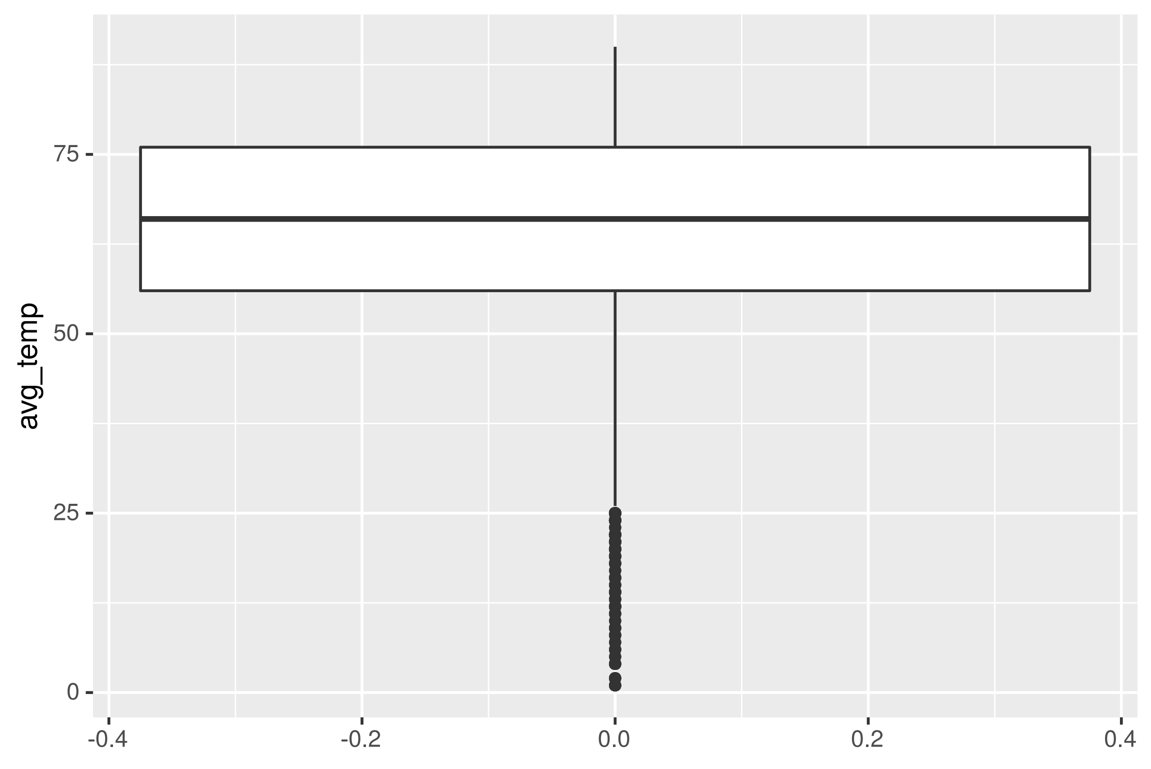
Weather Dataset

The data set `Weather` contains data on weather-related variables for several world cities.

```
#install.packages(mosaicData)
library(mosaicData)
data(Weather)
glimpse(Weather)
Rows: 3,655
Columns: 25
$ city          <chr> "Auckland", "Auckland", "Auckland", "Auckland", "Aucklan…
$ date          <date> 2016-01-01, 2016-01-02, 2016-01-03, 2016-01-04, 2016-01…
$ year          <dbl> 2016, 2016, 2016, 2016, 2016, 2016, 2016, 2016, 2016, 20…
$ month         <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,…
$ day           <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 1…
$ high_temp     <dbl> 68, 68, 77, 73, 69, 69, 71, 77, 69, 71, 75, 69, 71, 75, …
$ avg_temp      <dbl> 65, 66, 72, 66, 62, 63, 66, 70, 66, 66, 67, 66, 66, 68, …
$ low_temp      <dbl> 62, 64, 66, 60, 55, 57, 60, 64, 64, 62, 59, 62, 62, 62, …
$ high_dewpt    <dbl> 64, 64, 70, 66, 55, 54, 59, 72, 68, 63, 61, 66, 61, 63, …
$ avg_dewpt     <dbl> 60, 63, 67, 60, 52, 51, 54, 67, 61, 58, 58, 62, 57, 61, …
$ low_dewpt     <dbl> 55, 61, 64, 54, 48, 46, 50, 59, 55, 55, 54, 59, 54, 59, …
$ high_humidity <dbl> 100, 100, 100, 100, 82, 88, 83, 100, 100, 88, 94, 100, 8…
$ avg_humidity  <dbl> 82, 94, 91, 76, 69, 65, 65, 92, 81, 76, 72, 87, 73, 80, …
$ low_humidity  <dbl> 68, 88, 74, 53, 56, 46, 53, 83, 64, 64, 53, 78, 64, 65, …
$ high_hg       <dbl> 30.15, 30.04, 29.80, 30.12, 30.21, 30.24, 30.24, 30.01, …
```

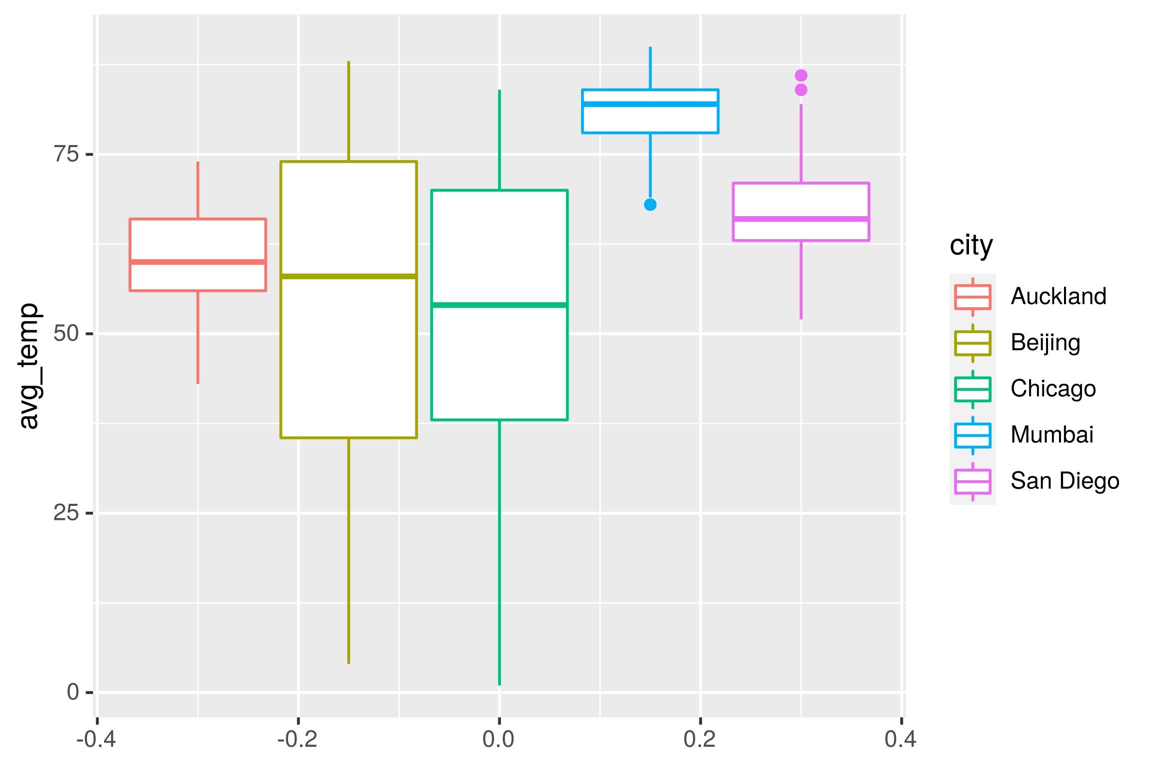
Weather patterns

```
ggplot(Weather, aes(y=avg_temp)) + geom_boxplot()
```



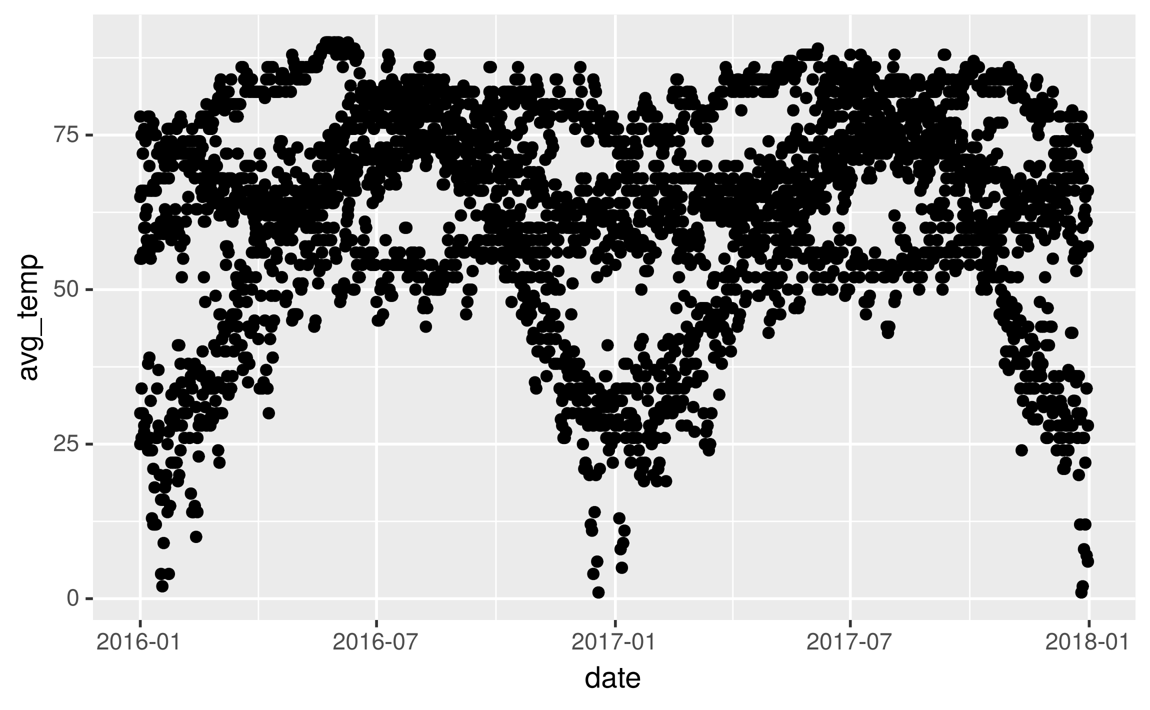
Weather patterns

```
ggplot(Weather, aes(y=avg_temp, group=city)) + geom_boxplot(aes(color=city))
```



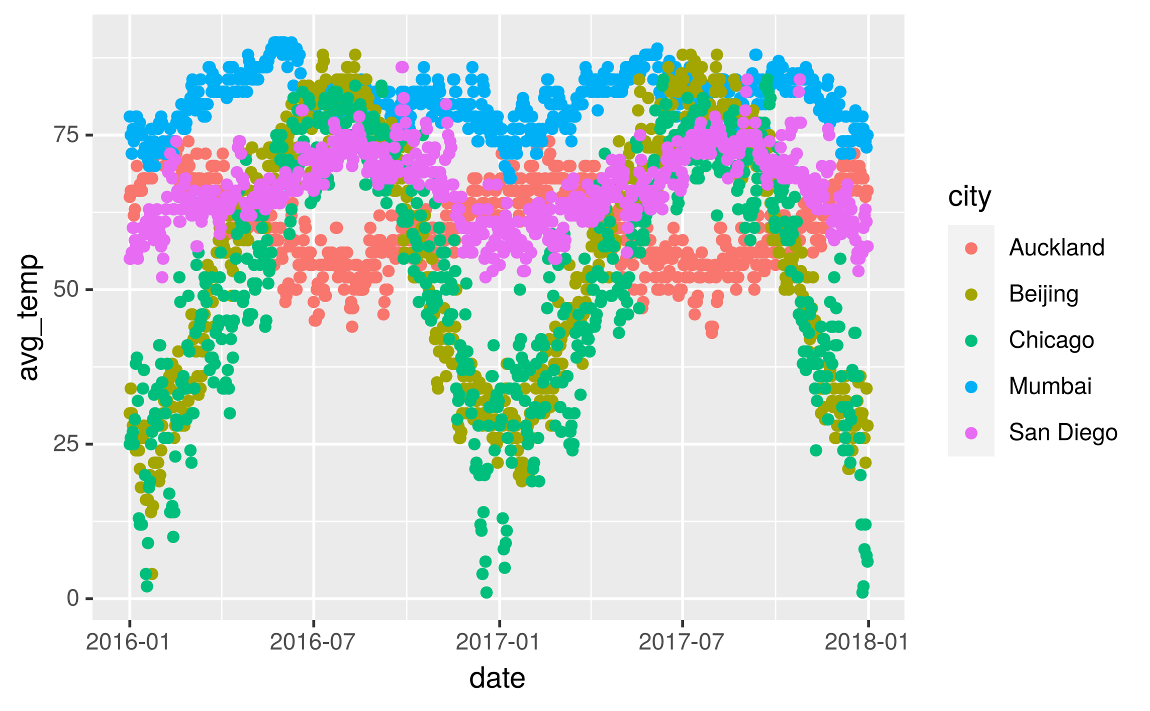
Weather patterns

```
ggplot(Weather, aes(x=date, y=avg_temp))+geom_point()
```



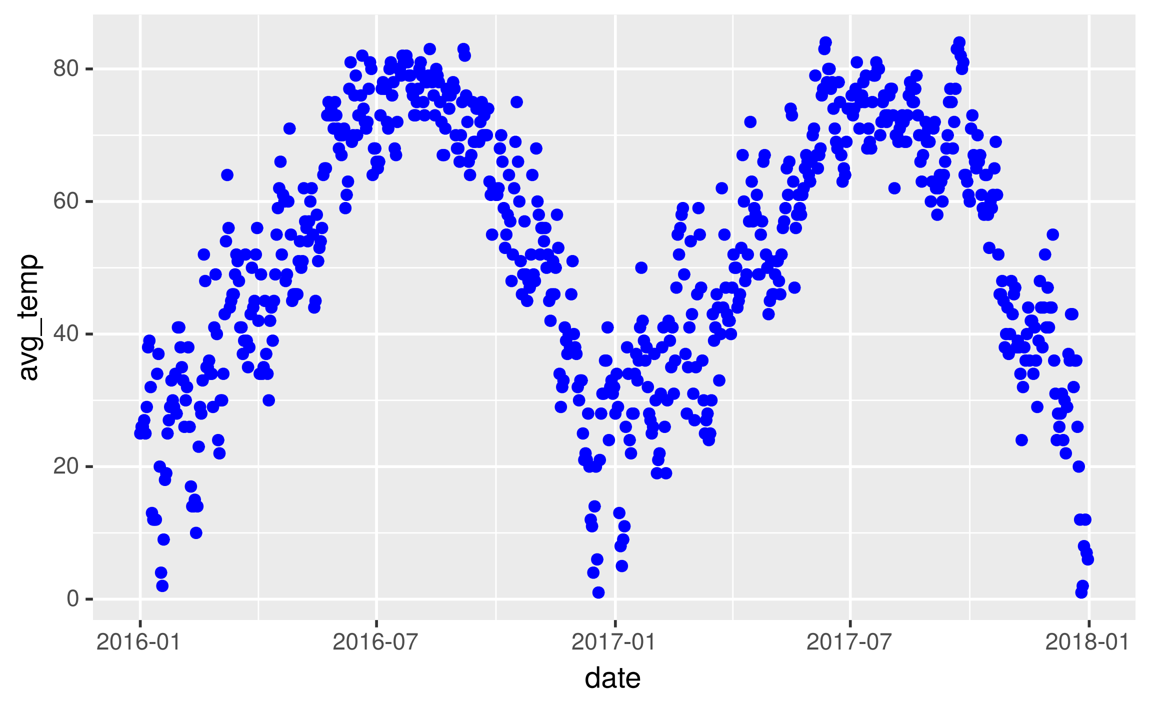
Weather patterns

```
ggplot(Weather, aes(x=date, y=avg_temp)) +
  geom_point(aes(color=city))
```



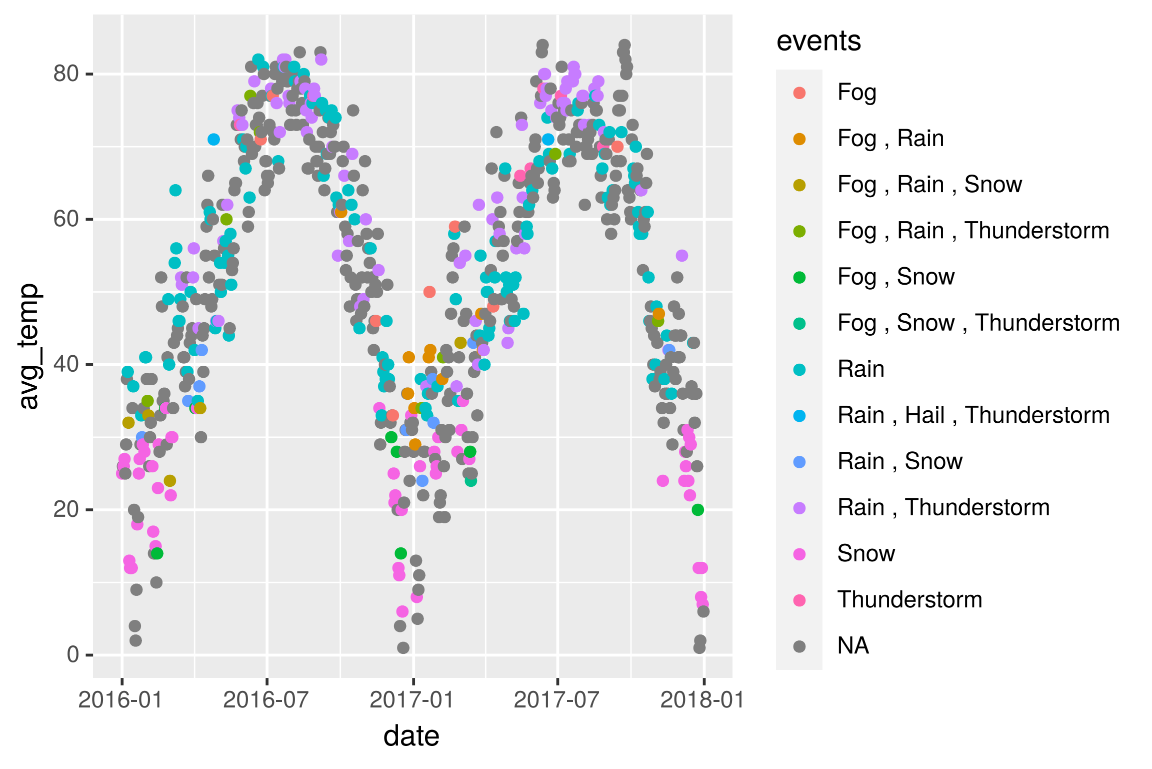
Weather patterns in Chicago

```
Chicago <- Weather %>% filter(city=='Chicago')
ggplot(Chicago, aes(x=date, y=avg_temp)) +
  geom_point(color='blue')
```



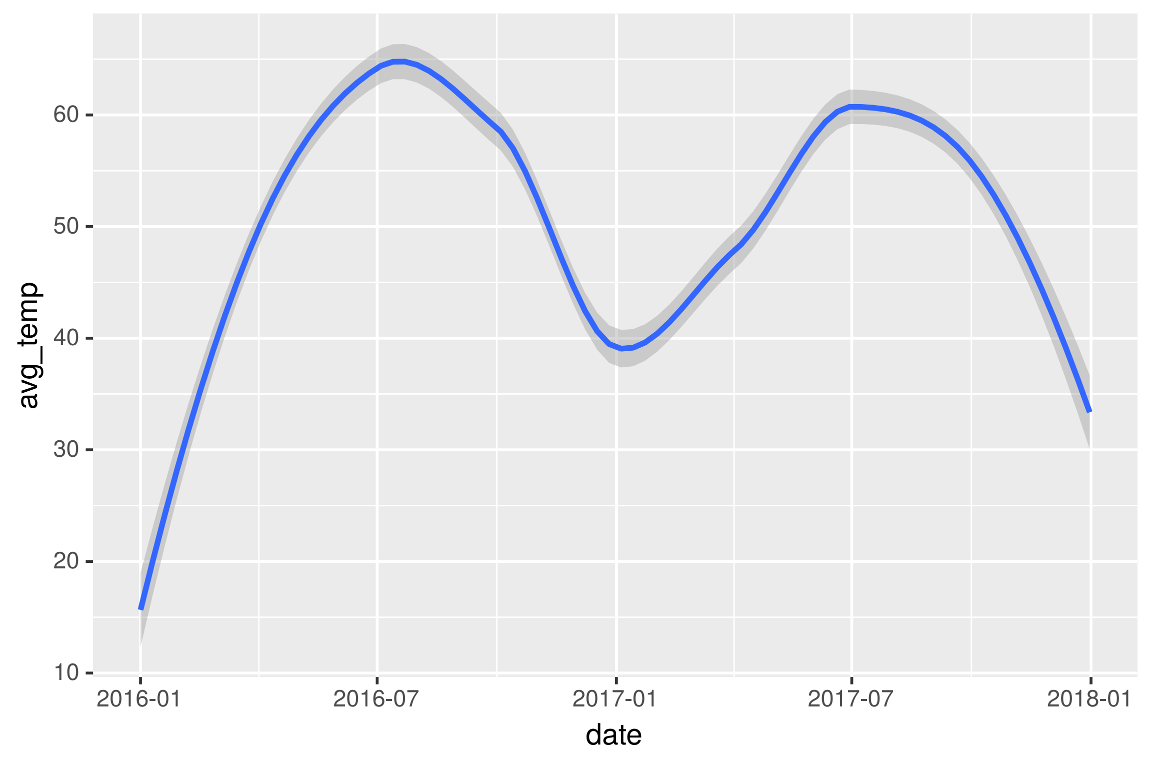
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=date, y=avg_temp)) +
  geom_point(aes(color=events))
```



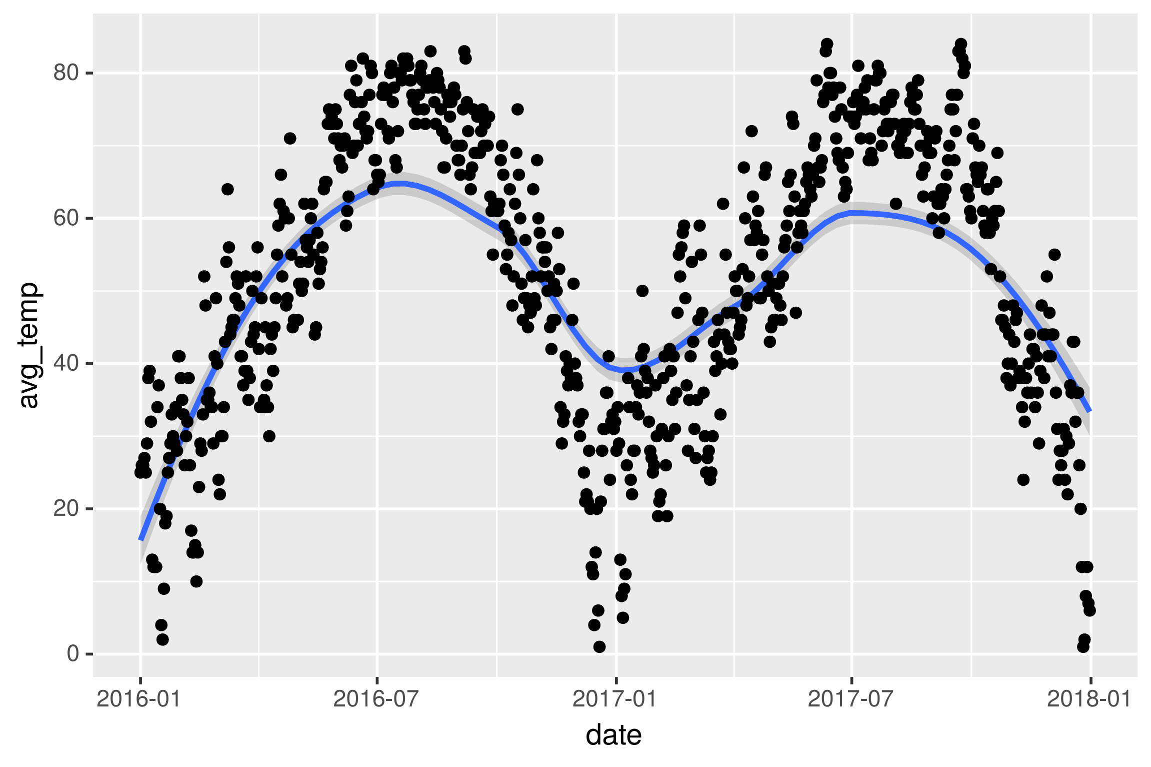
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=date, y=avg_temp)) +
  geom_smooth()
```



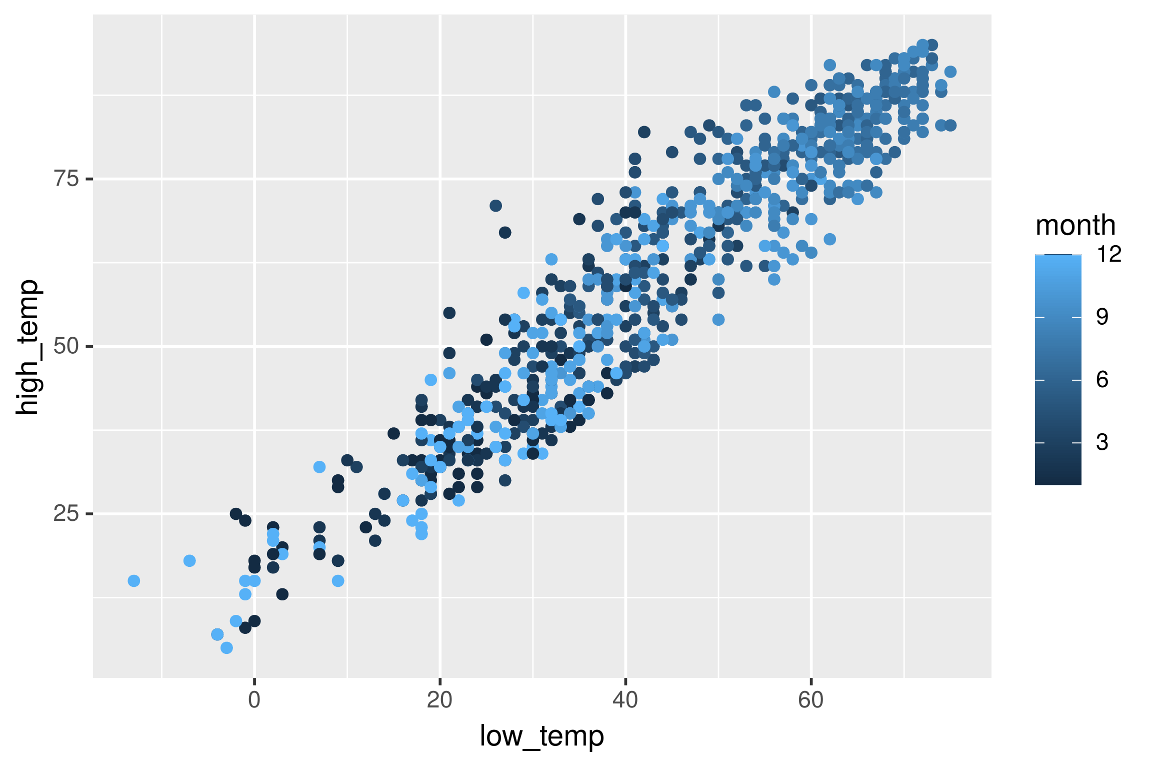
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=date, y=avg_temp)) +
  geom_smooth()+ geom_point()
```



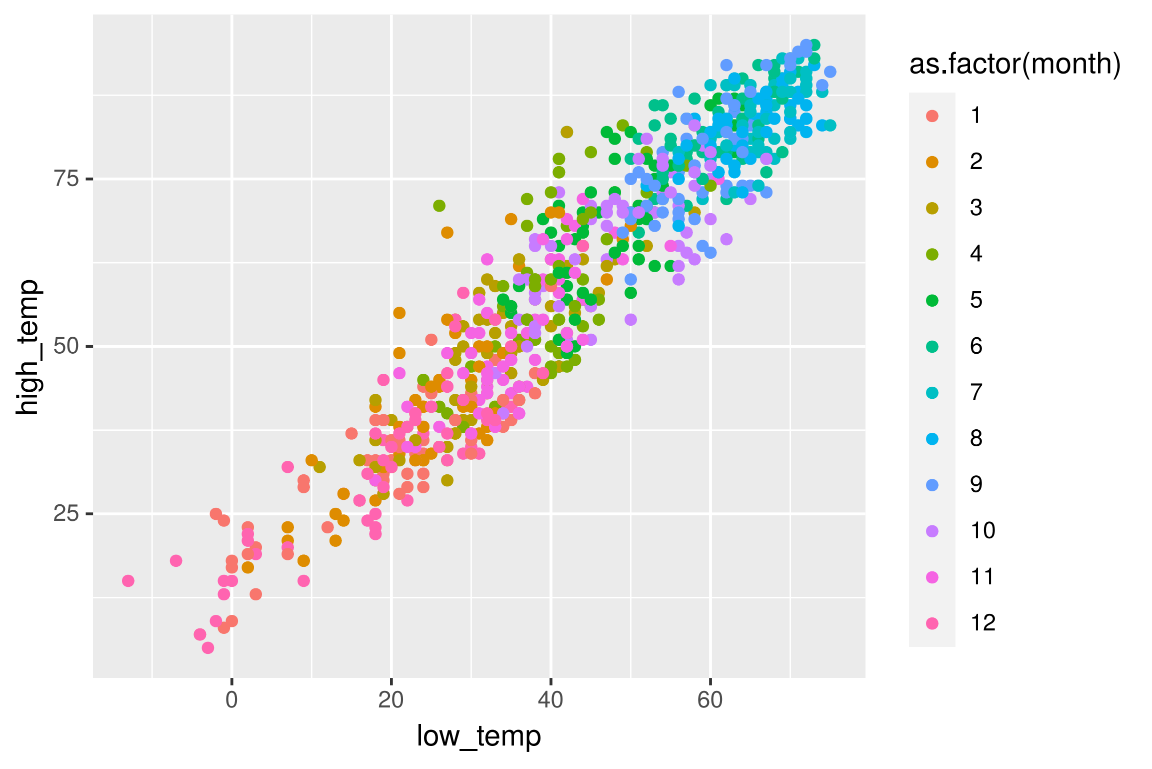
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=low_temp, y=high_temp)) +
  geom_point(aes(color=month))
```



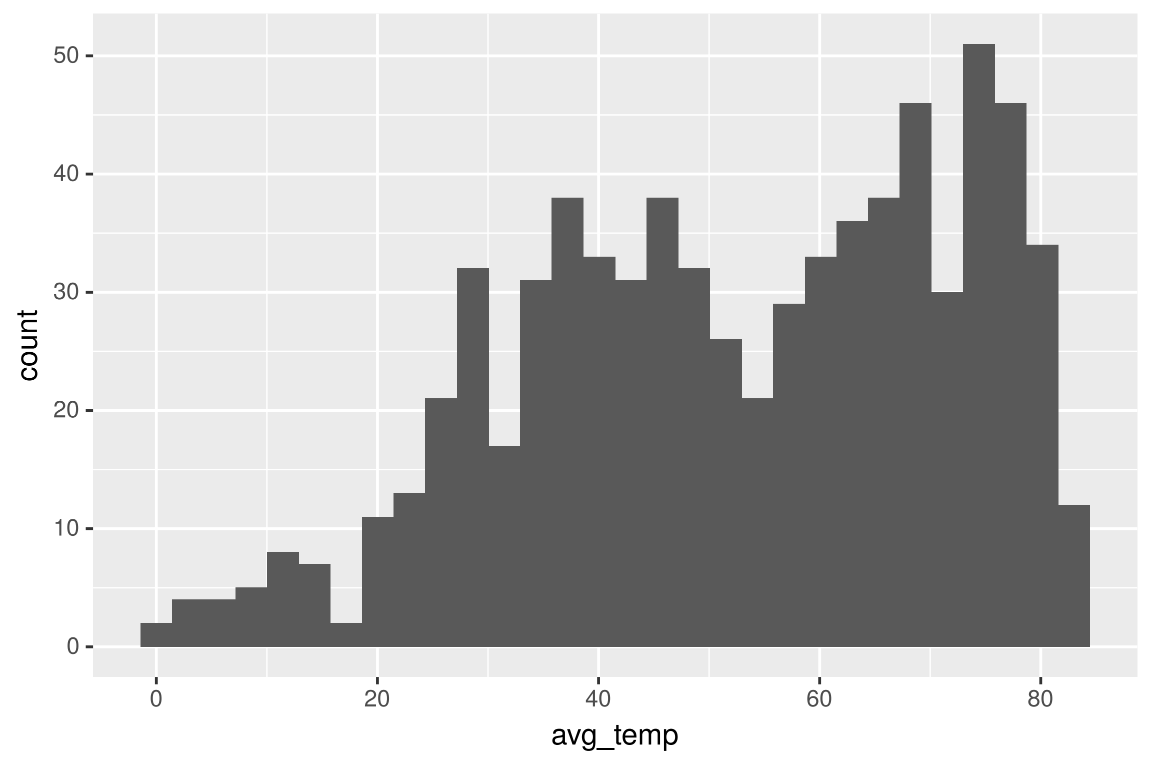
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=low_temp, y=high_temp)) +
  geom_point(aes(color=as.factor(month)))
```



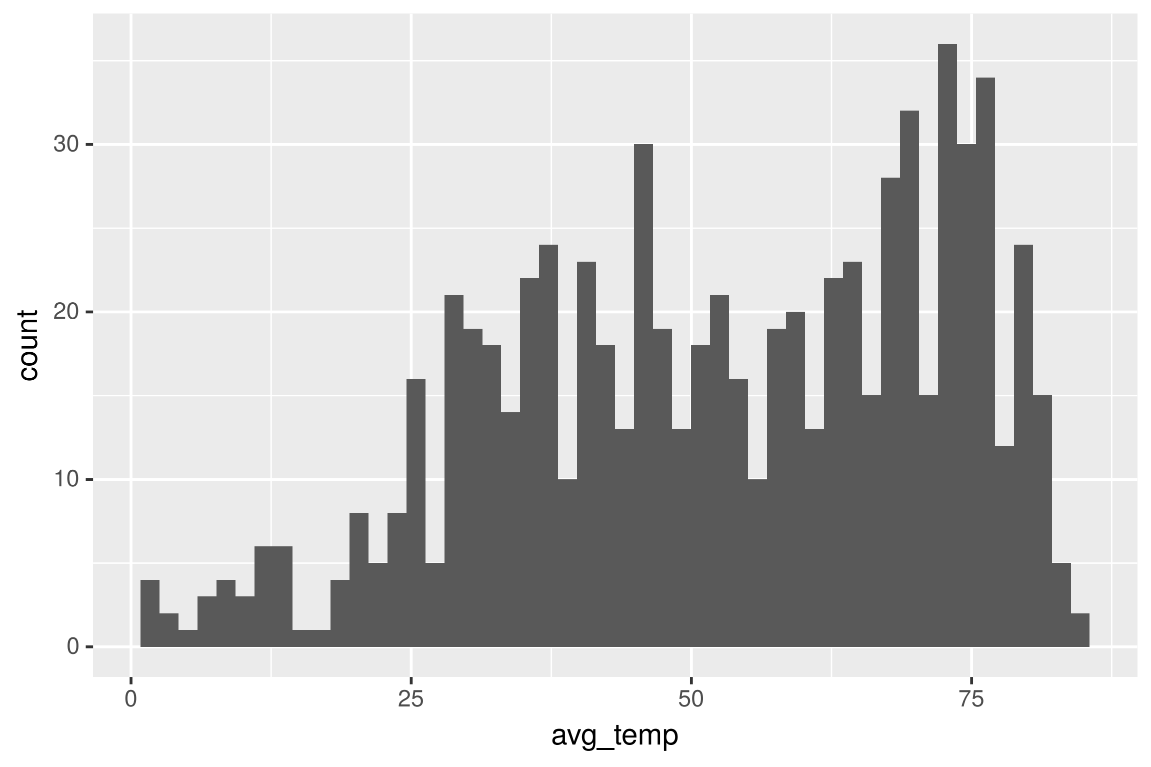
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp)) +
  geom_histogram()
```



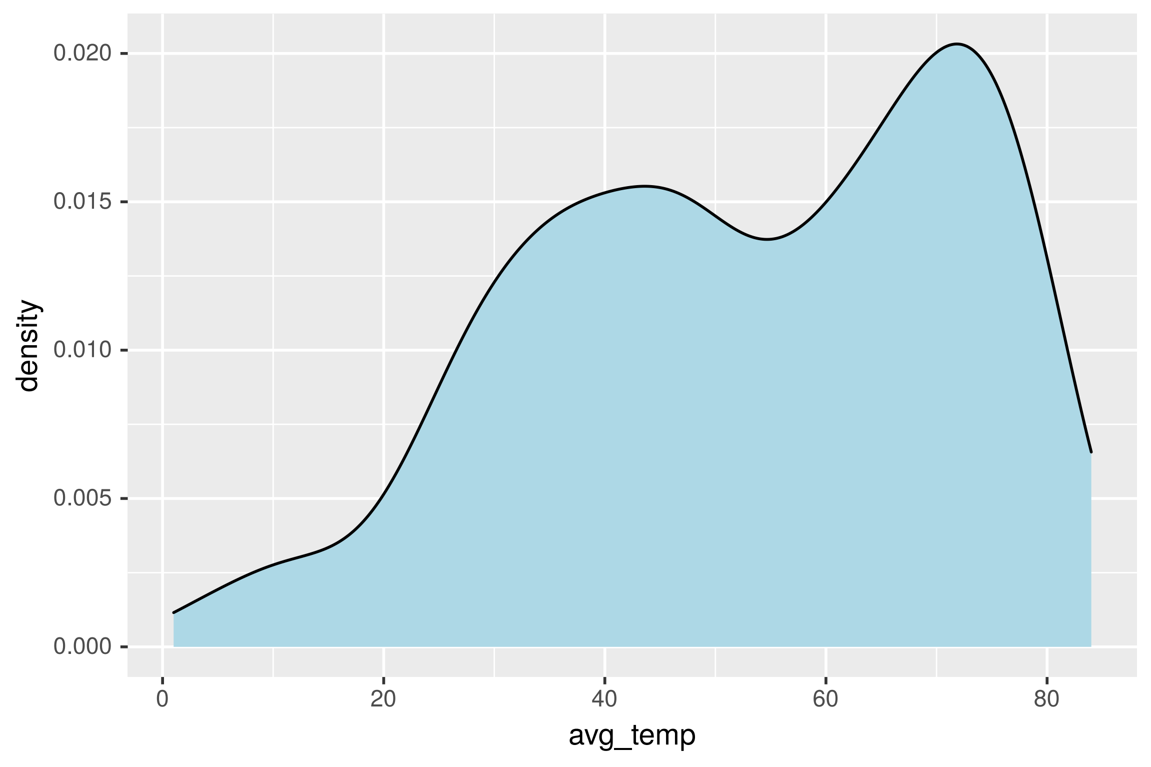
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp)) +
  geom_histogram(bins = 50)
```



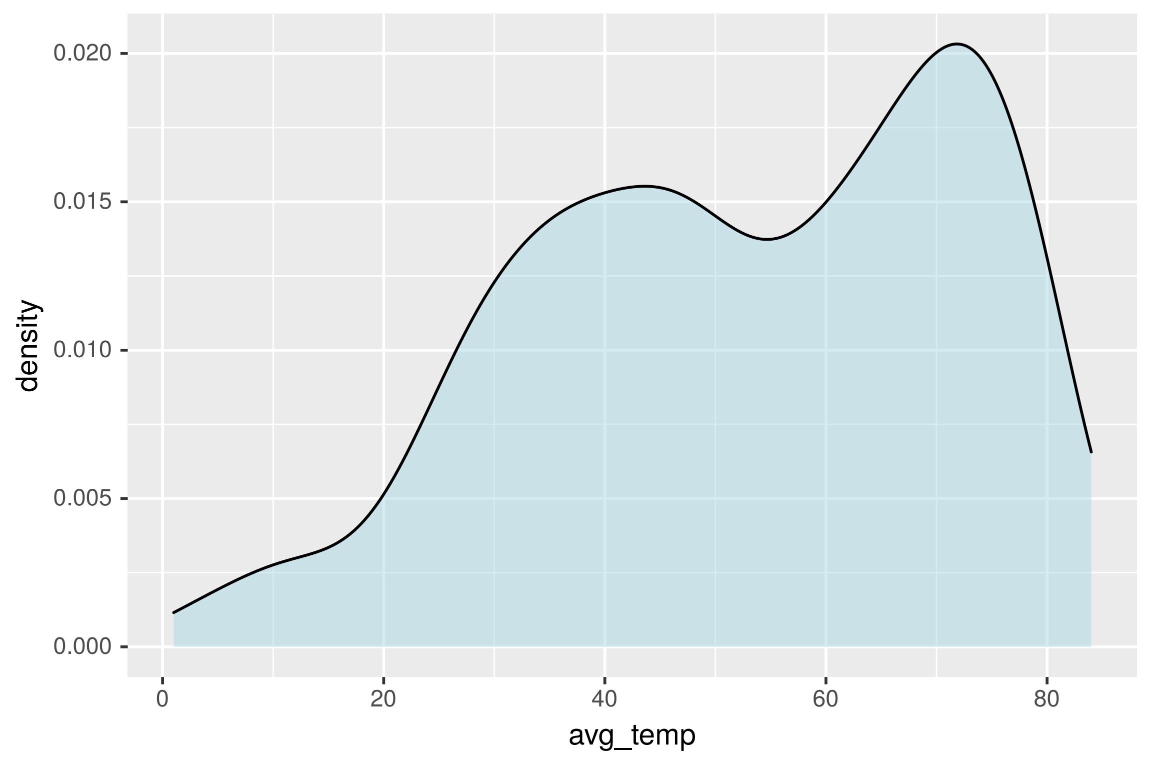
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp)) +
  geom_density(fill='lightblue')
```



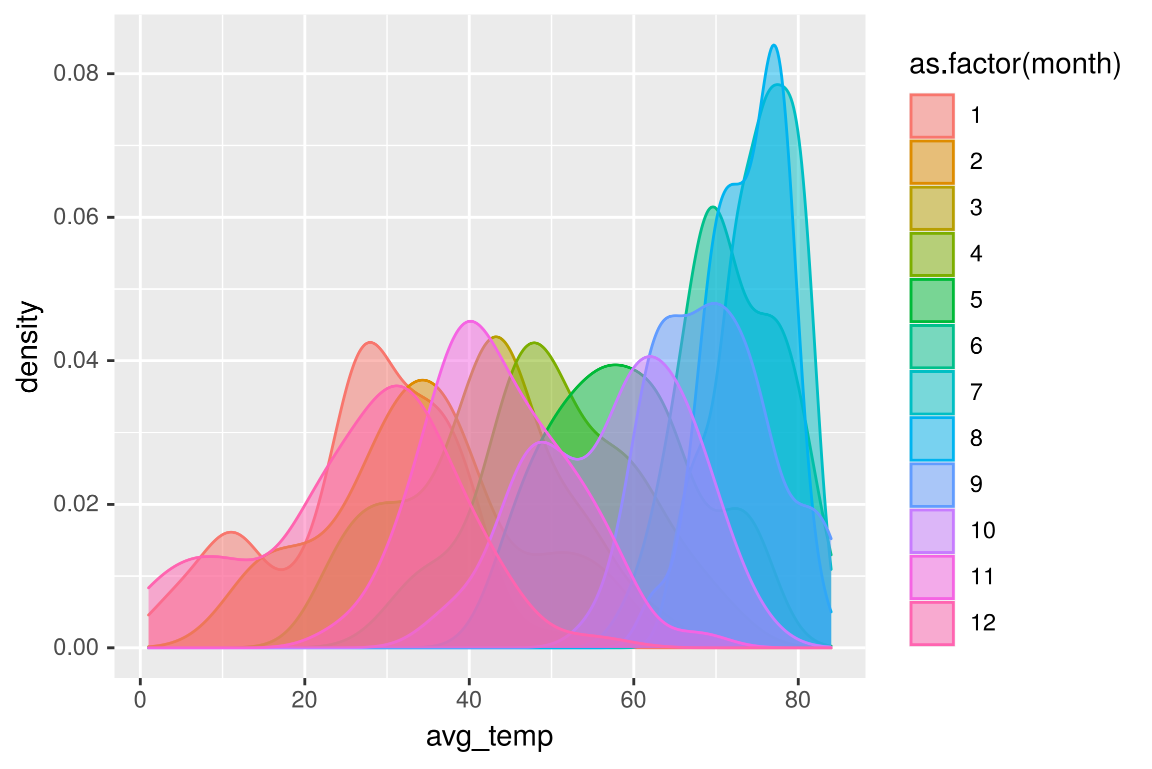
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp)) +
  geom_density(fill='lightblue', alpha=0.5)
```



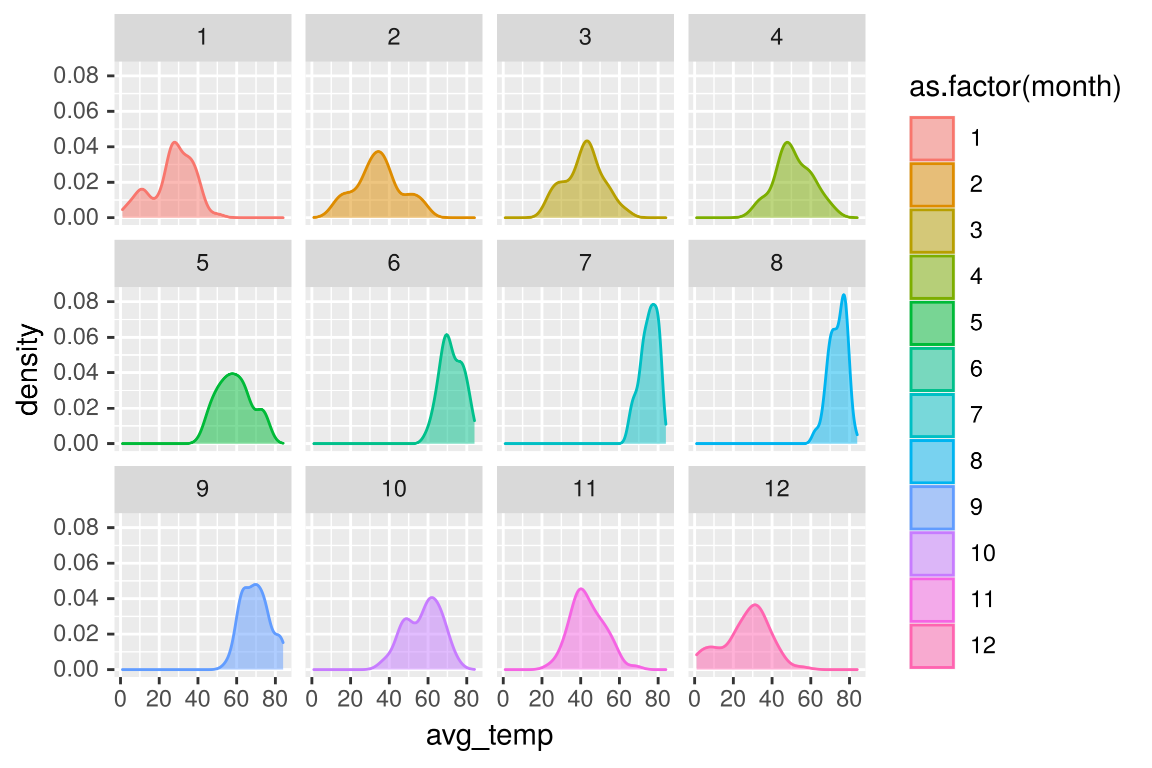
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp, group=month)) +
  geom_density(aes(color=as.factor(month), fill=as.factor(month)), alpha=0.5)
```



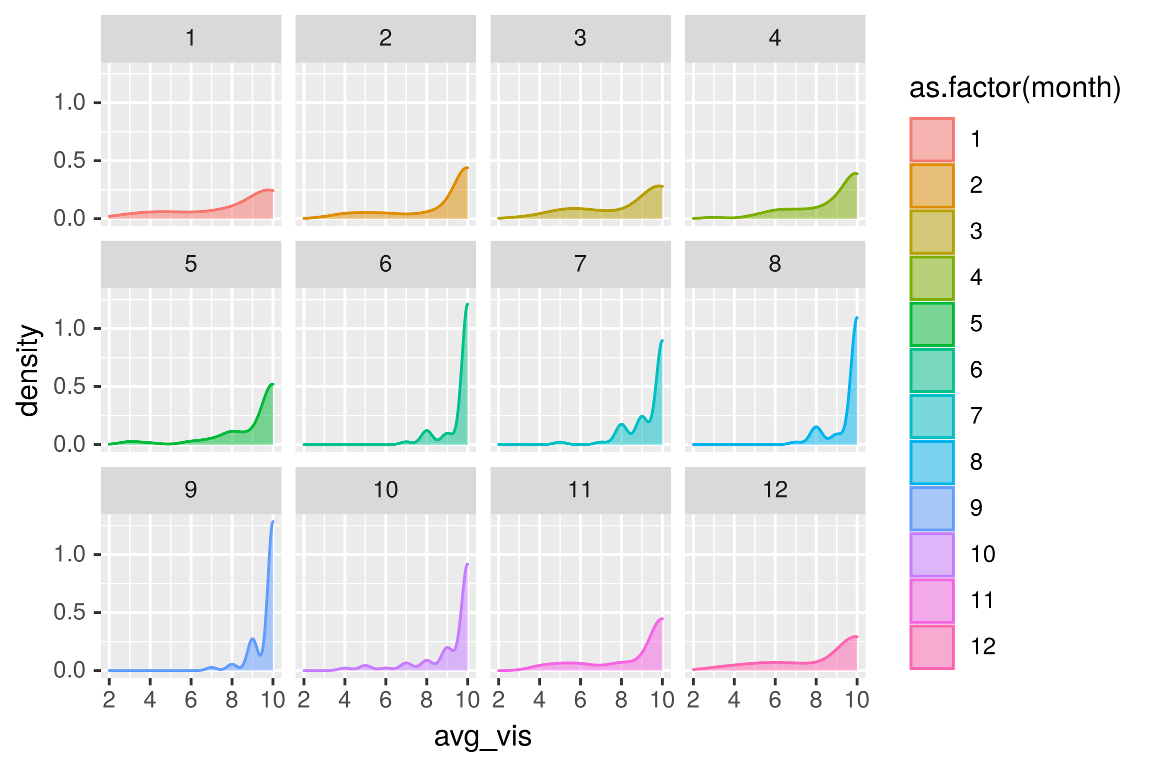
Weather patterns in Chicago

```
ggplot(Chicago, aes(x=avg_temp, group=month)) +
  geom_density(aes(color=as.factor(month), fill=as.factor(month)), alpha=0.5) +
  facet_wrap(~month, nrow=3)
```



Weather patterns in Chicago

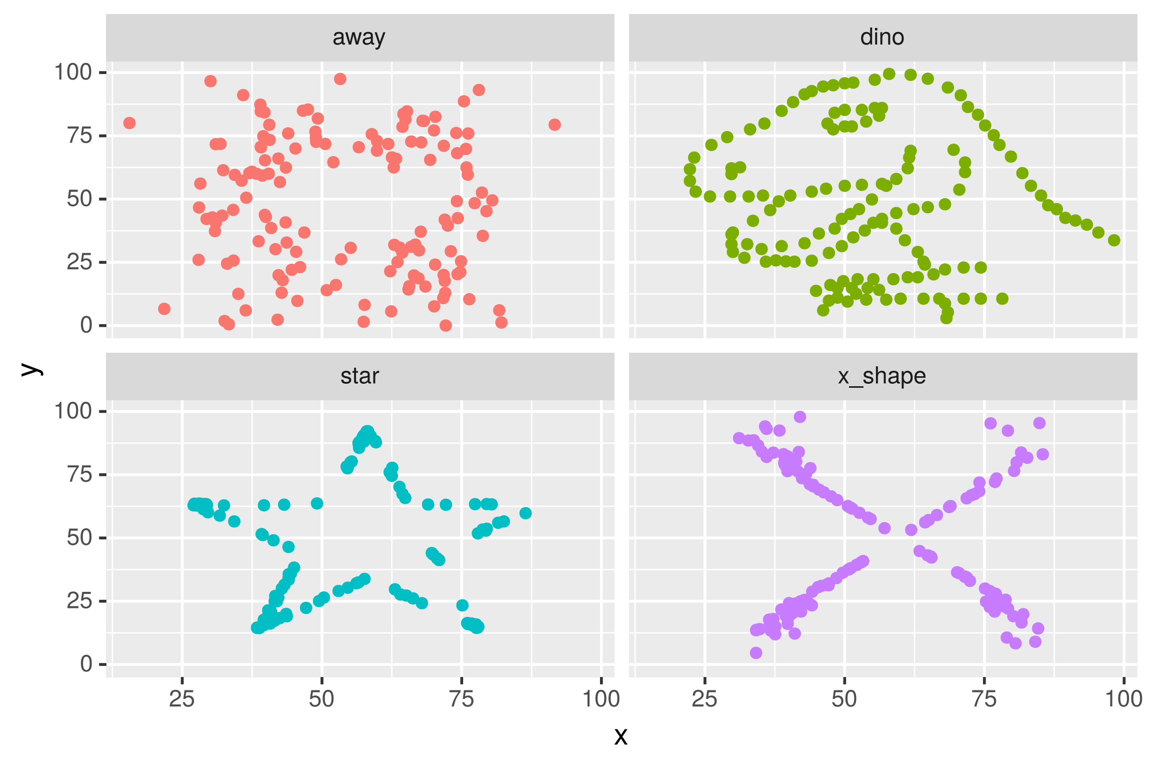
```r
ggplot(Chicago, aes(x=avg_vis, group=month)) +
  geom_density(aes(color=as.factor(month), fill=as.factor(month)), alpha=0.5) +
  facet_wrap(~month, nrow=3)
```



Your Turn 2

- `datasaurus_dozen` in the `datasauRus` package contains 13 datasets. We will use 4 of these datasets for this exercise.

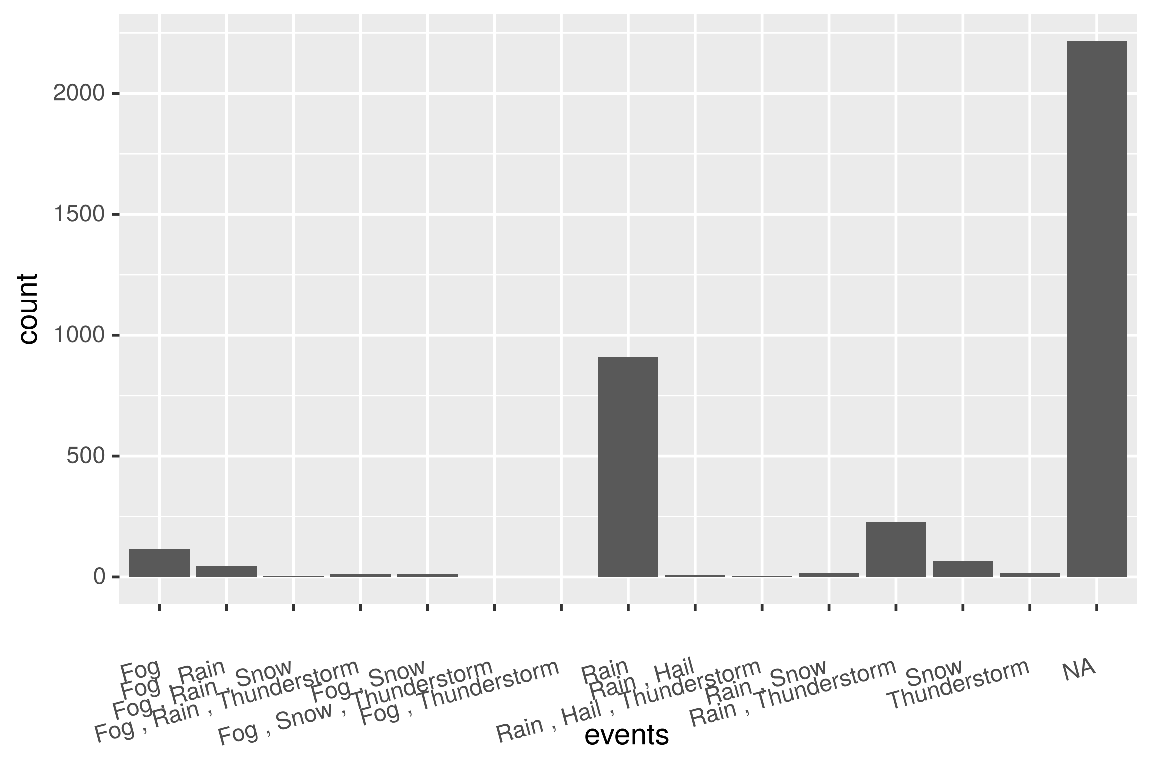
- Create this ribbon of plots using `facet_wrap`.



05:00

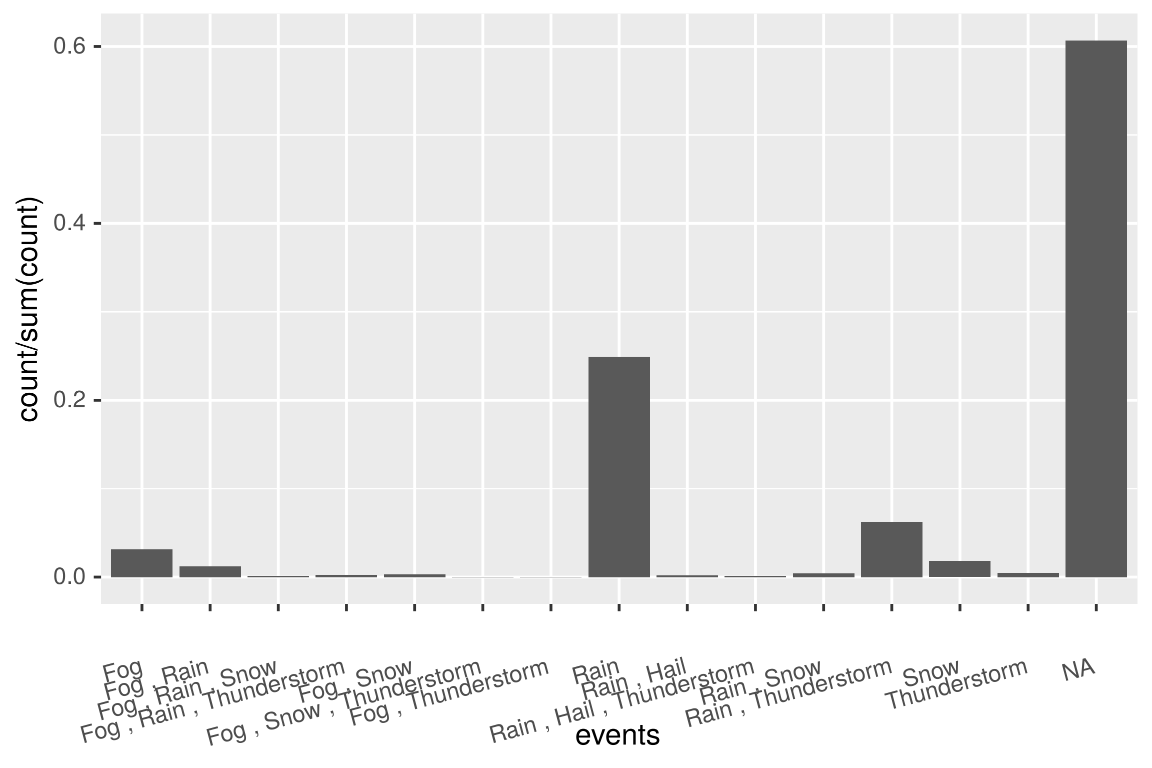
Weather events

```
ggplot(Weather, aes(x=events)) +
  geom_bar()+
  theme(axis.text.x = element_text(angle = 15, vjust = 0.5, hjust=1))
```



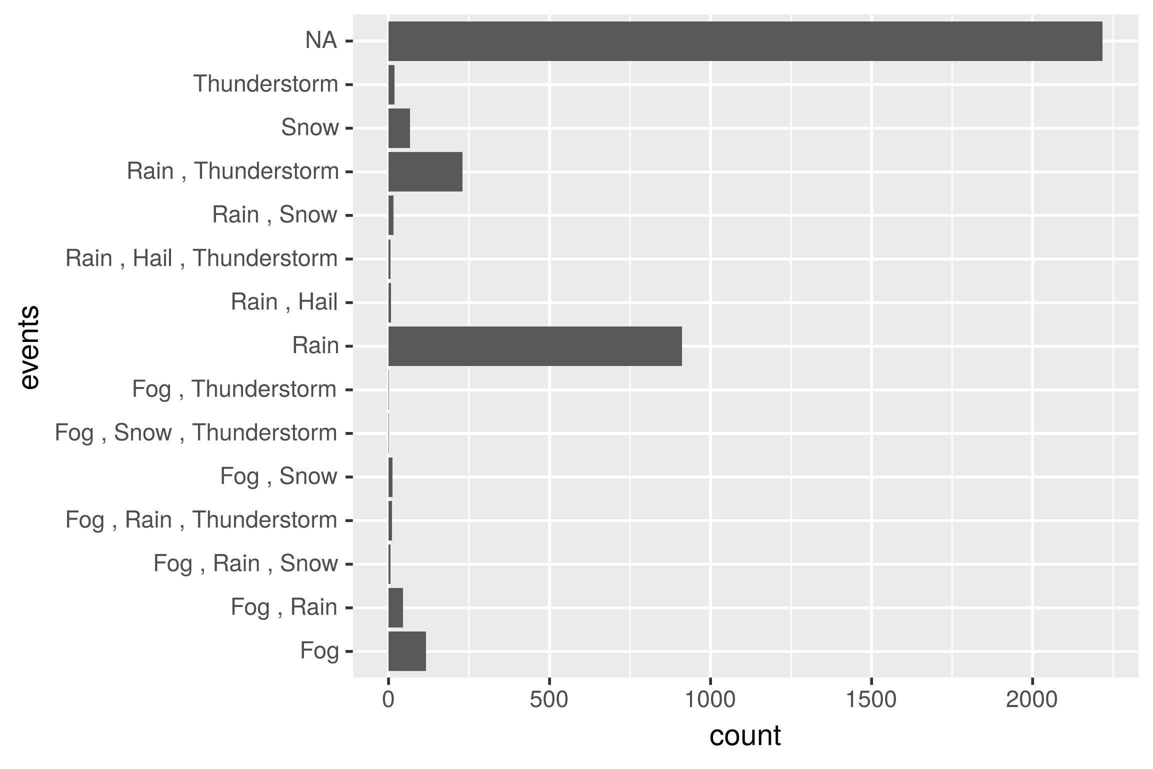
Weather events — proportions

```
ggplot(Weather, aes(x=events, y = ..count../sum(..count..))) +   # change y-axis to proportion
  geom_bar()+
  theme(axis.text.x = element_text(angle = 15, vjust = 0.5, hjust=1))
```



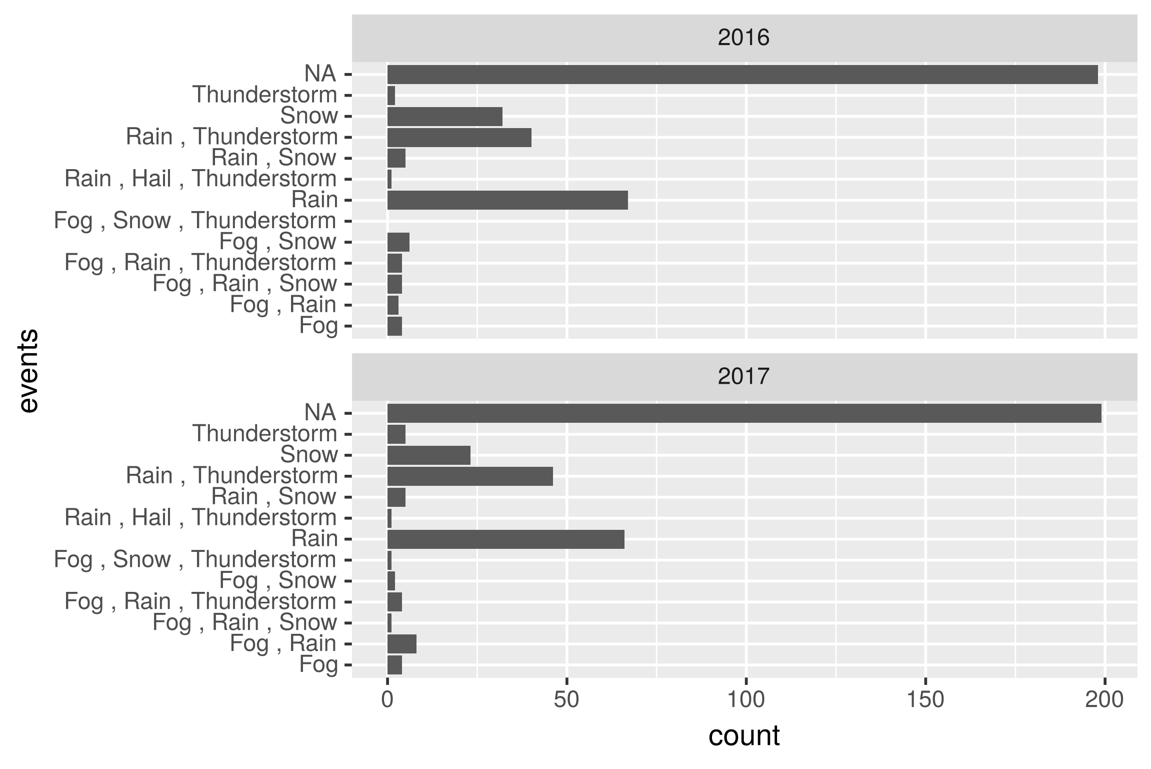
Weather events

```
ggplot(Weather, aes(x=events)) +
  geom_bar() +
  coord_flip()
```



Weather events in Chicago

```
ggplot(Chicago, aes(x=events)) +
  geom_bar() +
  coord_flip() +
  facet_wrap(~year, nrow=2)
```

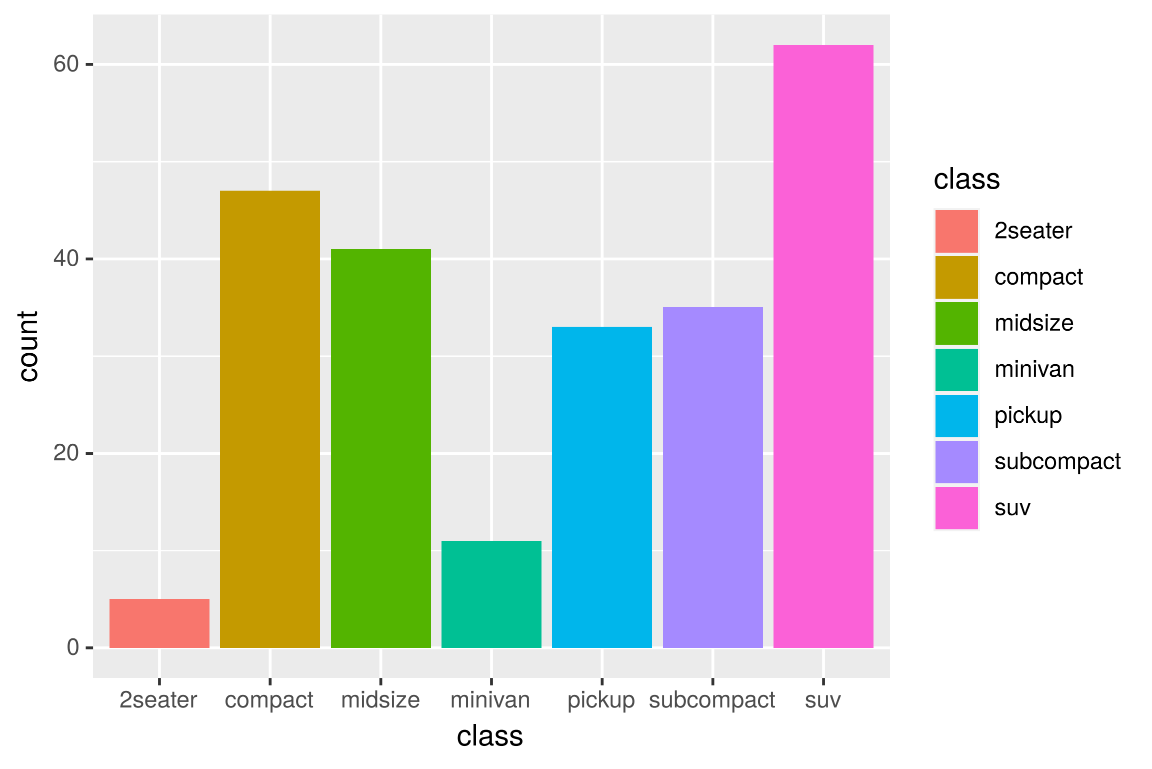


Your Turn 3

03:00

- The `mpg` data set is loaded with the `tidyverse`. Run `?mpg` for info.

- Create this bar chart of vehicle `class`

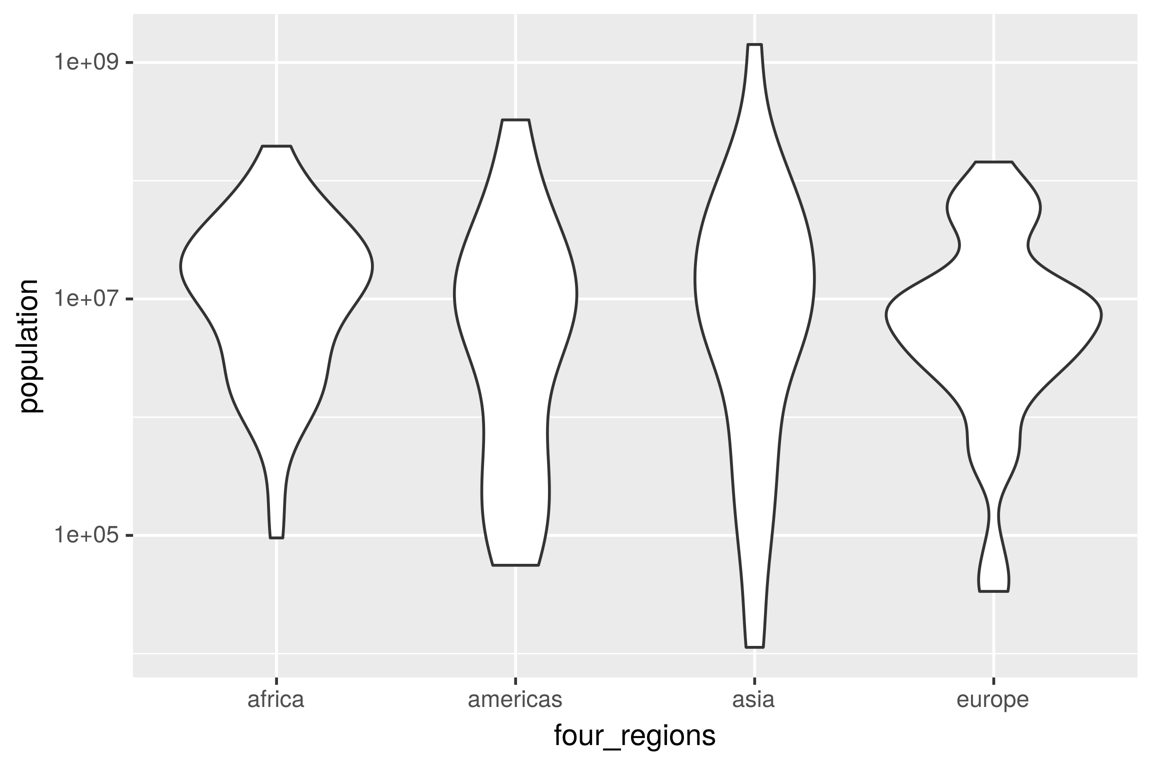


Gapminder dataset

```
gap_dat <-read.csv("https://raw.githubusercontent.com/deepbas/statdatasets/main/gapminder2018.csv")
gapminder <- gap_dat %>% filter(year == 2018)
glimpse(gapminder)
Rows: 193
Columns: 8
$ country         <chr> "Afghanistan", "Albania", "Algeria", "Andorra", "Angol…
$ year            <int> 2018, 2018, 2018, 2018, 2018, 2018, 2018, 2018, 2018, …
$ income          <int> 1870, 12400, 13700, 51500, 5850, 21000, 18900, 8660, 4…
$ life_expectancy <dbl> 58.7, 78.0, 77.9, NA, 65.2, 77.6, 77.0, 76.0, 82.9, 81…
$ population      <int> 36400000, 2930000, 42000000, 77000, 30800000, 103000, …
$ four_regions    <chr> "asia", "europe", "africa", "europe", "africa", "ameri…
$ eight_regions   <chr> "asia_west", "europe_east", "africa_north", "europe_we…
$ six_regions     <chr> "south_asia", "europe_central_asia", "middle_east_nort…
```

Your Turn 4

The `gapminder` dataset provides values for life expectancy, GDP per capita, and population, every five years, from 1960 to 2018. Use `gapminder` dataset to answer the given set of problems.



10:00