Univariate data: additional topics

ENS-215

09-Feb-2022

Let's load in the packages we'll use today. You probably don't have the `ggridges` package yet so first go ahead and install it in your Package window.

```
library(tidyverse)
library(stats)
library(ggridges)
```

Before we move on, let's load in a univariate dataset that we can work with in today's lecture. We'll load in the NOAA monthly precipitation dataset that we've worked with prior

```
precip_data <- read_csv("https://stahlm.github.io/ENS_215/Data/NOAA_State_Precip_LabData.csv")
```

```
precip_data <- precip_data %>%
  mutate(time_period = if_else(Year >= 1950,"Post-1950","Pre-1950"))
```

Take a quick look at the data to refamiliarize yourself with it.

Now let's create a new dataset that just has the precipitation data for NY.

```
ny_precip <- precip_data %>%
  filter(state_cd == "NY")
```

Visualizing and interpreting univariate distributions

Let's continue learning about how to generate visualizations that display a univariate data distribution and how to interpret these data.

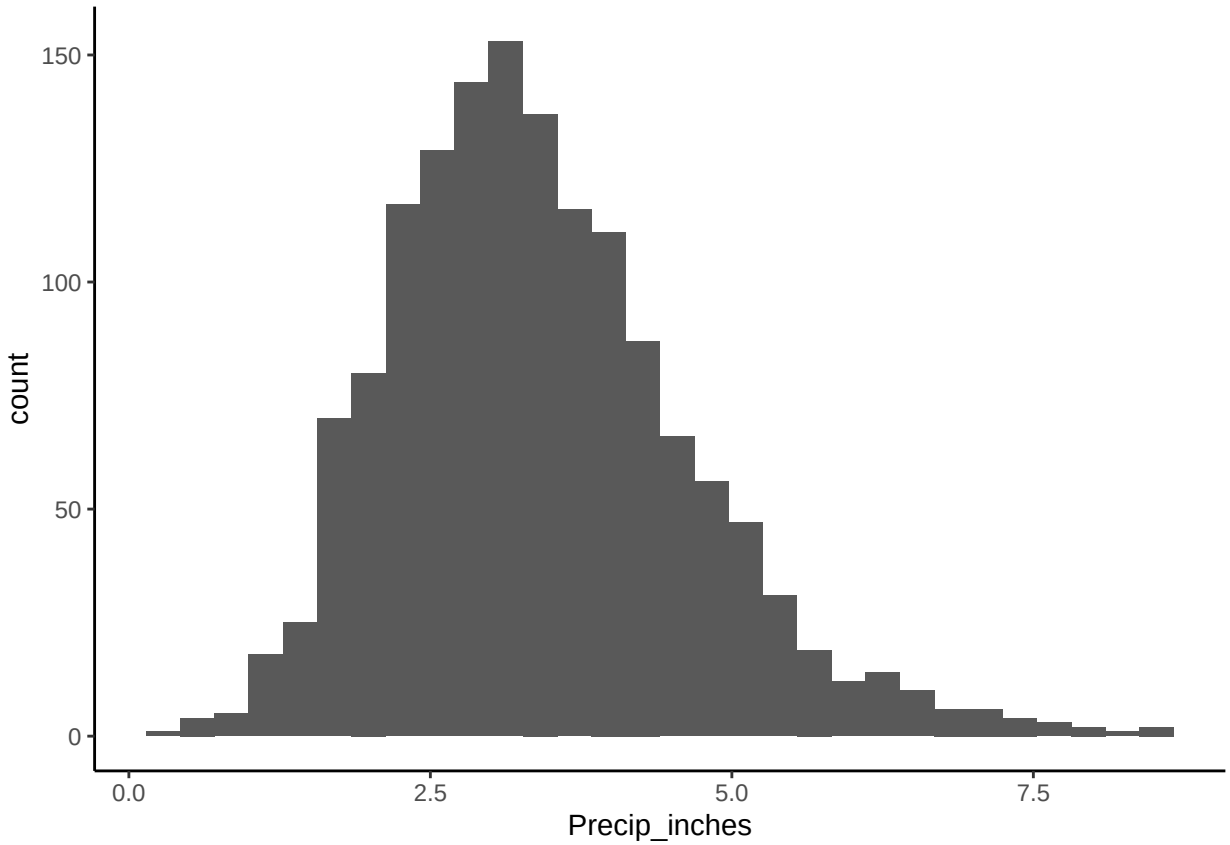
Histograms and density plots

Histograms are another method of displaying the distribution of univariate data. A histogram **bins** the values and plots the frequency of values falling into each bin. Typically the bins are of equal width

Let's generate a histogram of the monthly precipitation data for NY to highlight how they look and their utility. To generate a histogram we use `geom_histogram()`. Notice that we only need to pass a single variable (in this case `Precip_inches`). The histogram displays values of the variable on the x-axis and the number of occurrences (counts) within bins (ranges) of the variable.

```
ny_precip %>%
  ggplot(aes(Precip_inches)) +
  geom_histogram() +
  theme_classic()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



You can see that the bins (columns) are of equal width and their height corresponds to the number of observations falling within that bin. For instance, the bin with the most observations is from 2.98-3.26 inches and has 153 observations.

Histograms allow you to identify how frequently values of the variable of interest are observed (*in the above example monthly precipitation in inches*). By looking at the histogram and comparing bar heights you can determine the relative frequencies in each bin.

Looking at the above histogram we can see that while observed monthly precipitation values range from near zero to about 8 inches, the vast majority of observations fall within 2 to 5 inches. While observations outside of the range 2 to 5 inches do occur, we can see that they are relatively infrequent.

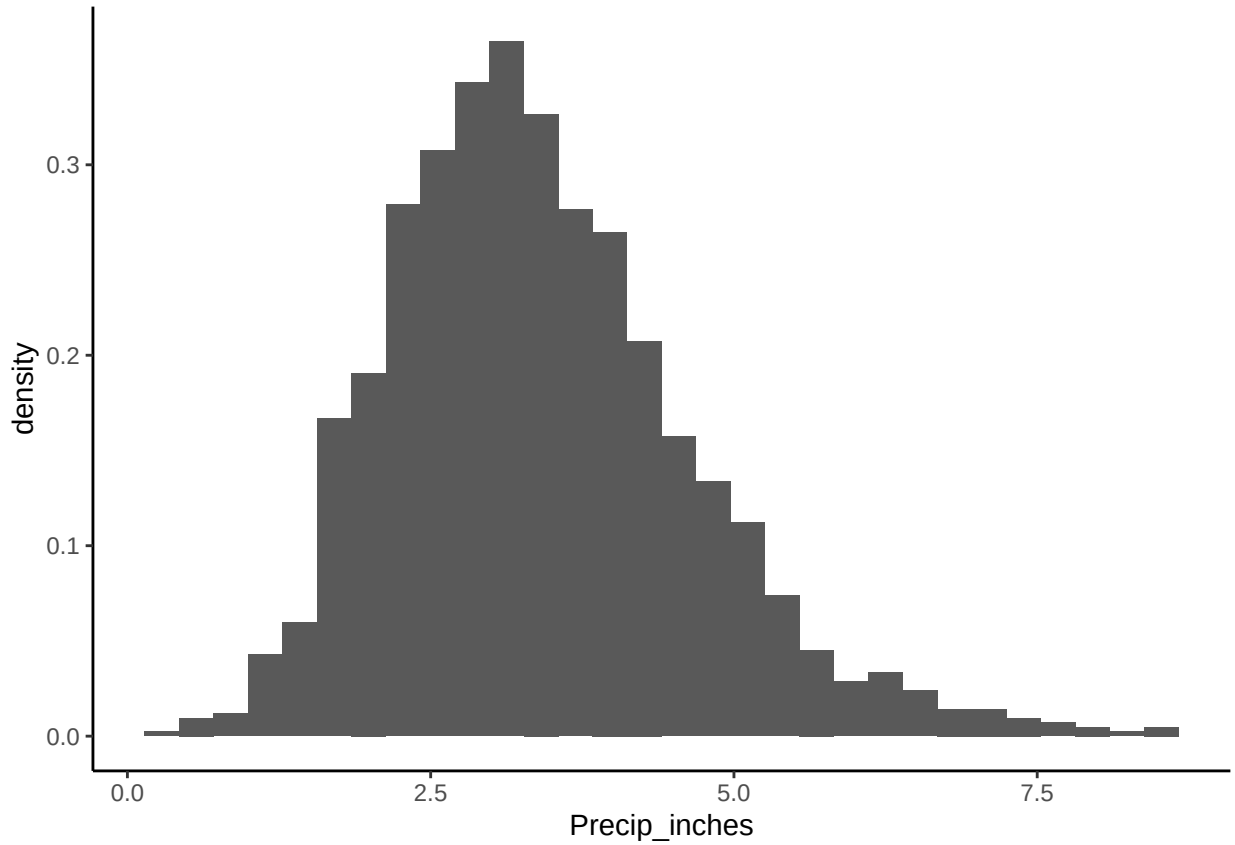
You can specify the number of bins using the `bins =` argument to allow for greater control over your histogram's appearance and the level its level of detail.

Oftentimes we are interested in displaying relative frequencies on the y-axis and not the absolute number of observations in each bin. To display the relative frequency you can specify `stat(density)` or `stat(ndensity)` in your `aes()` function. The `density` and `ndensity` display the bin heights that have been normalized so that the area integrates to one (`density`) or so that the height of the tallest bin is one (`ndensity`)

The example below demonstrates the use of `stat(density)`

```
ny_precip %>%
  ggplot(aes(Precip_inches, stat(density))) +
  geom_histogram() +
  theme_classic()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

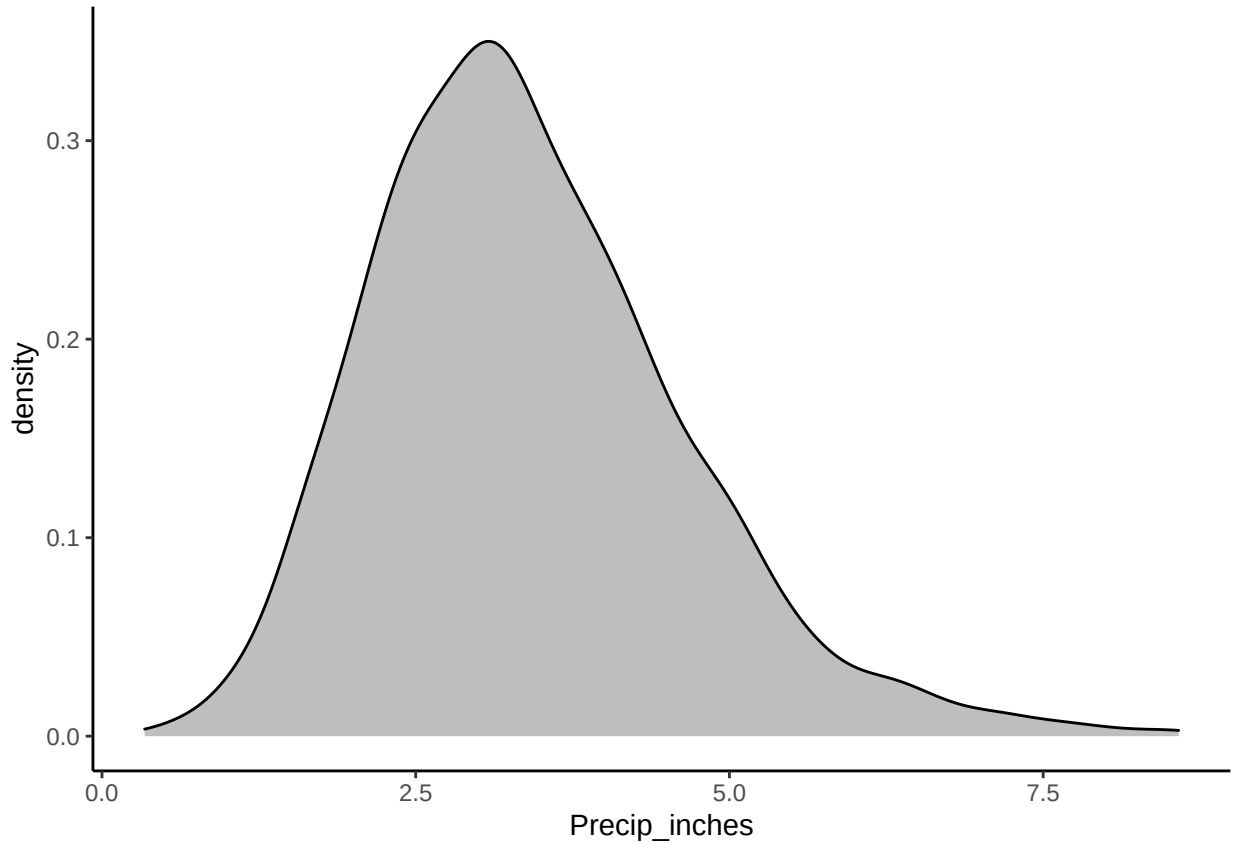


Now generate a similar histogram, this time using `bins = 50` and setting the stat to `ndensity`

```
# Your code here
```

You can generate a smoothed version of a histogram, which is refered to as a **density curve**, using the `geom_density()` function. The area under the density curve integrates to one.

```
ny_precip %>%
  ggplot(aes(Precip_inches)) +
  geom_density(fill = "grey") +
  theme_classic()
```



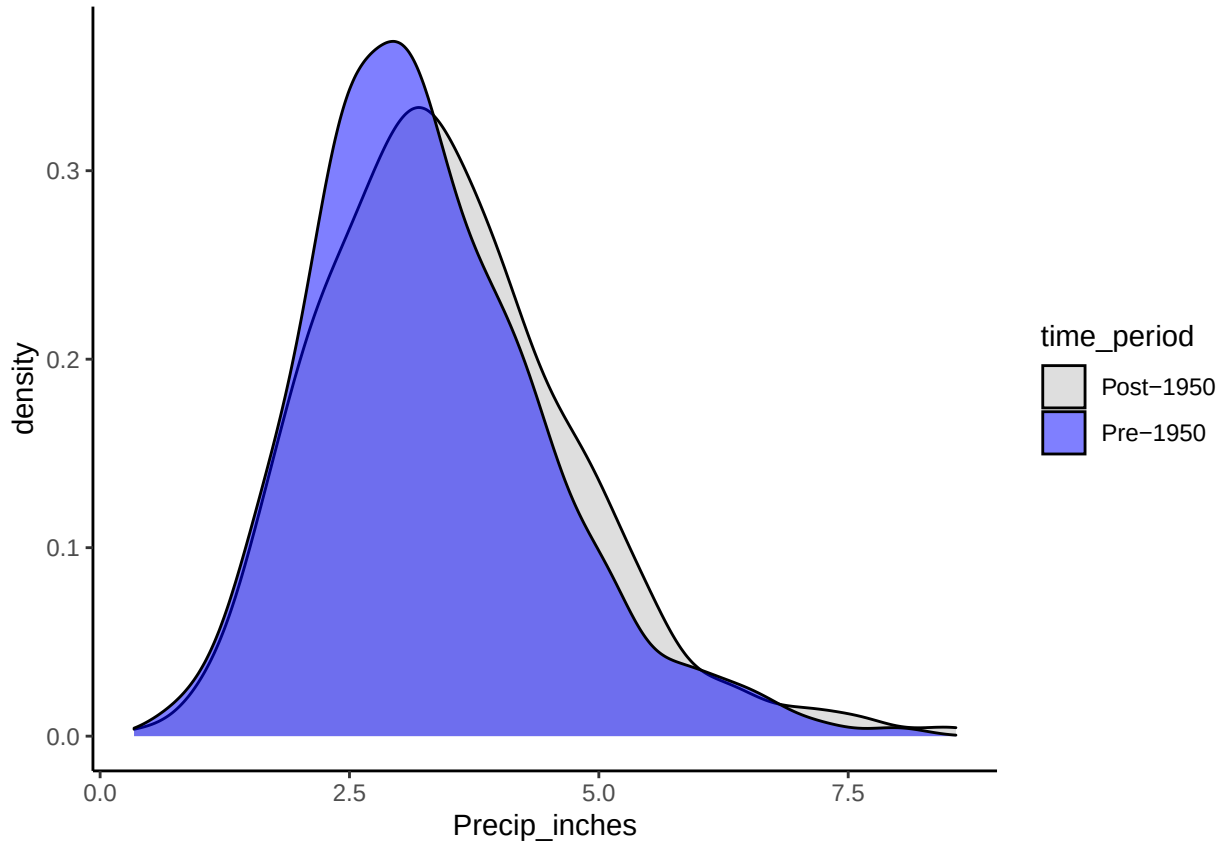
- Does the histogram/density plot provide more information than a box plot?

- Based on the density plot, is the above distribution symmetric or skewed?
- What else can you conclude based on the density plot? Look back at the stats overview section to recall how we describe distributions.

Density curves are a great way to compare distributions between groups. We can examine how the modes, ranges, central tendancies, and relative frequencies across values vary between the different groups.

Let's use density curves to examine how the distribution of monthly precipitation has changed over time in the state of New York.

```
ny_precip %>%
  ggplot(aes(Precip_inches, group = time_period, fill = time_period)) +
  geom_density(alpha = 0.5) +
  scale_fill_manual(values = c("grey", "blue")) +
  theme_classic()
```

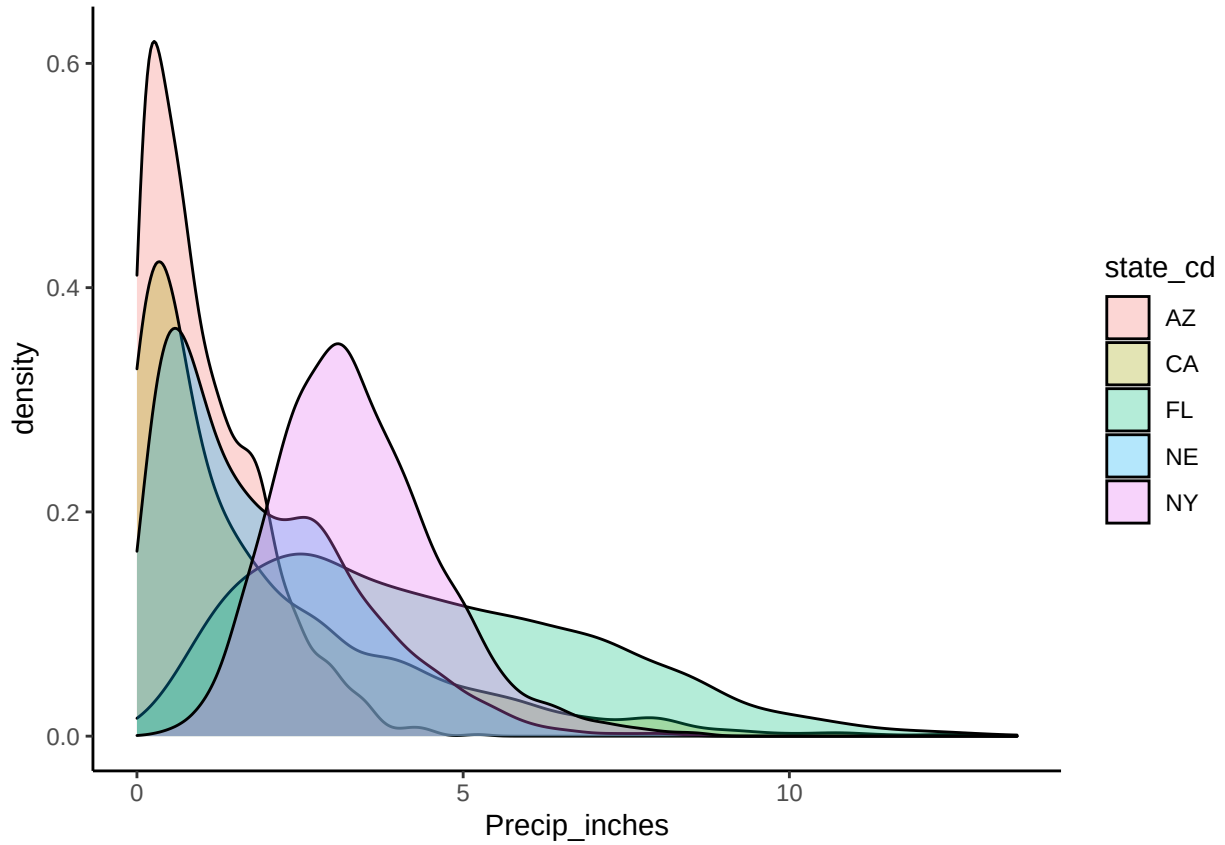


- Describe the distributions (symmetry/skewness, modes, . . .)
  - Add vertical lines showing the medians for each of the periods
- Has the distribution of precipitation changed over time? With you neighbor discuss the potential implications of any changes.

You saw above how overlaying density curves can be a great way to compare the distribution of a univariate data between multiple groups. With the example above we were comparing just two groups so the graphic wasn't too cluttered and we were able to easily interpret the results. However, when you want to compare more than 2 or 3 groups, the graphic can become difficult to read.

In the example below we are comparing the distribution of monthly precipitation between five US states. You can see that with five groups the graphic is becoming unwieldy.

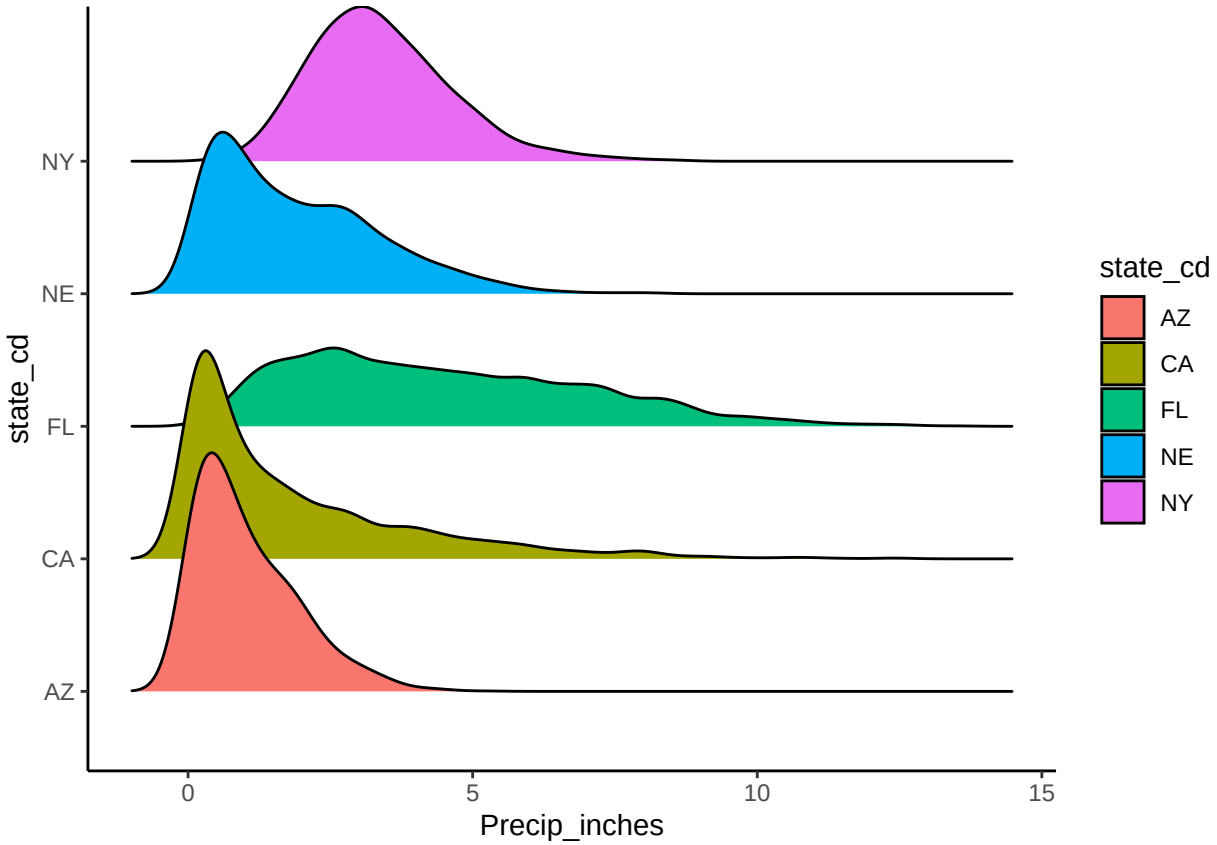
```
precip_data %>%
  filter(state_cd %in% c("NY","FL","CA","AZ","NE")) %>%
  ggplot() + geom_density(aes(x = Precip_inches, fill = state_cd), alpha = 0.3) +
  theme_classic()
```



Thankfully, we can rely on the `geom_density_ridges()` function from the `ggridges` package. This function puts each density curve on its own baseline – allowing us to easily compare across groups while keeping the graphic uncluttered. The graphic below displays the exact same data as the previous example, though this time the graphic is much easier to read.

```
precip_data %>%
  filter(state_cd %in% c("NY","FL","CA","AZ","NE")) %>%
  ggplot(aes(x = Precip_inches, y = state_cd)) +
  geom_density_ridges(aes(fill = state_cd)) +
  theme_classic()
```

```
## Picking joint bandwidth of 0.329
```



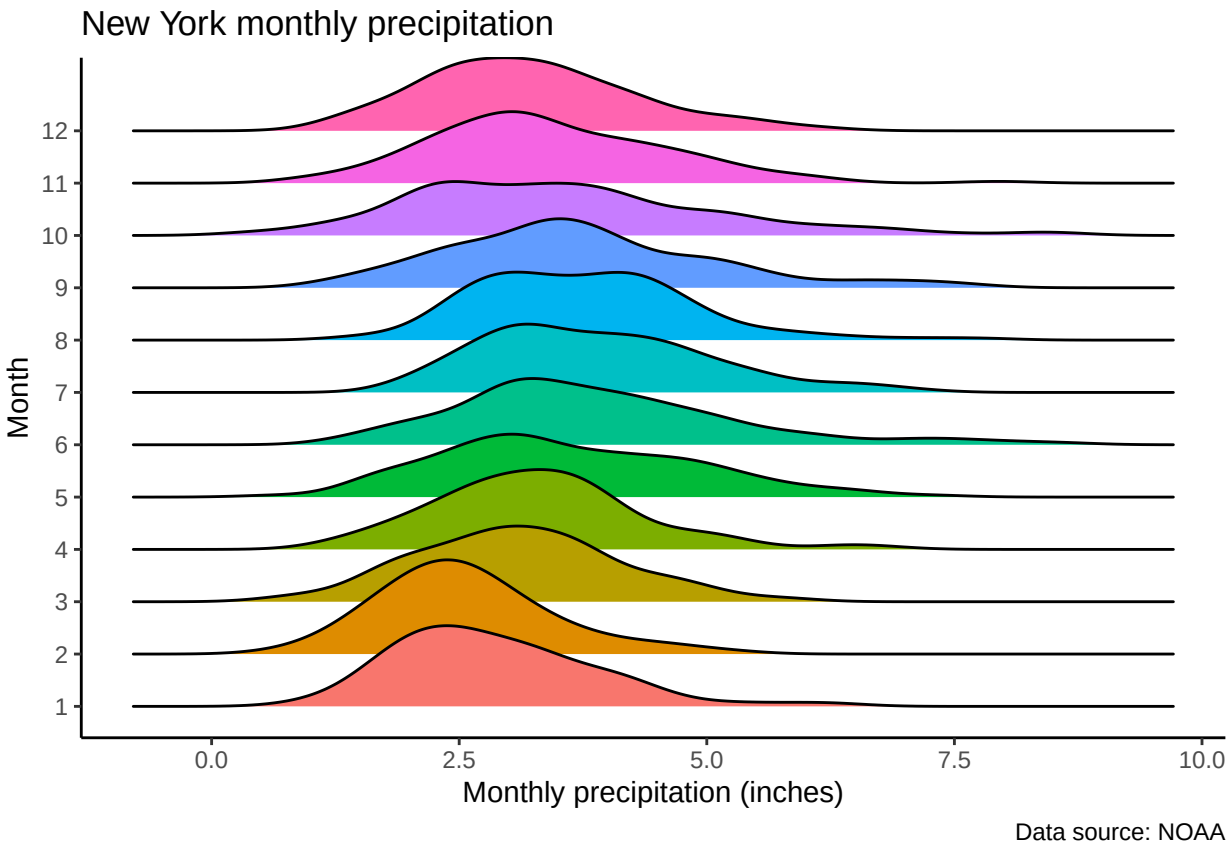
Exercise

1. Examine the precipitation distribution between each month for the state of Florida (FL). You should use a `geom_density_ridges` plot as part of your analysis. You can also generate a summary table with precipitation statistics for each month.

- Based on your analysis describe the distribution of precipitation within each month and across months
  - Is there a strong seasonal pattern in precipitation?
  - Are the monthly distributions (some or all) skewed? Left-skewed? Right-skewed?
  - Do some months display a wide range of precipitation values (i.e. high variance) and do others have a tighter distribution of values?
  - With your neighbors discuss any potential environmental and/or societal issues that might result from the precipitation patterns observed.
  - When you finish the above, you should try the same exercise on another state. California is pretty interesting to check (as I'm sure many other states are as well).

Below is an example of what your graphic should like like – note that I am plotting NY here and you should plot FL. Try to make your graphic presentation quality (look back at our lectures on presentation quality graphics for additional guidance).

```
## Picking joint bandwidth of 0.377
```



2. Create density curves that examine how the distribution of precipitation has changed from pre-1950 to post-1950 for the following US states: "NY","NH","VT","ME".

Do this with a single call to `ggplot()`. On the graphic for each state you should have two density curves (one for each time period). Hint: you will need to rely on faceting here.

- Have conditions become wetter or drier in the Northeast?
- Have extreme events become more or less frequent?
- Has the variablity in monthly precipitation changed? With your neighbor discuss some of the implications of any changes.

Quantile plots and cumulative distributions

Histograms and density curves display the density of observations across the range of values that the variable of interest takes. At a given value on the x-axis, the higher/taller the curve/bar the greater the frequency of observations at (or around) that value.

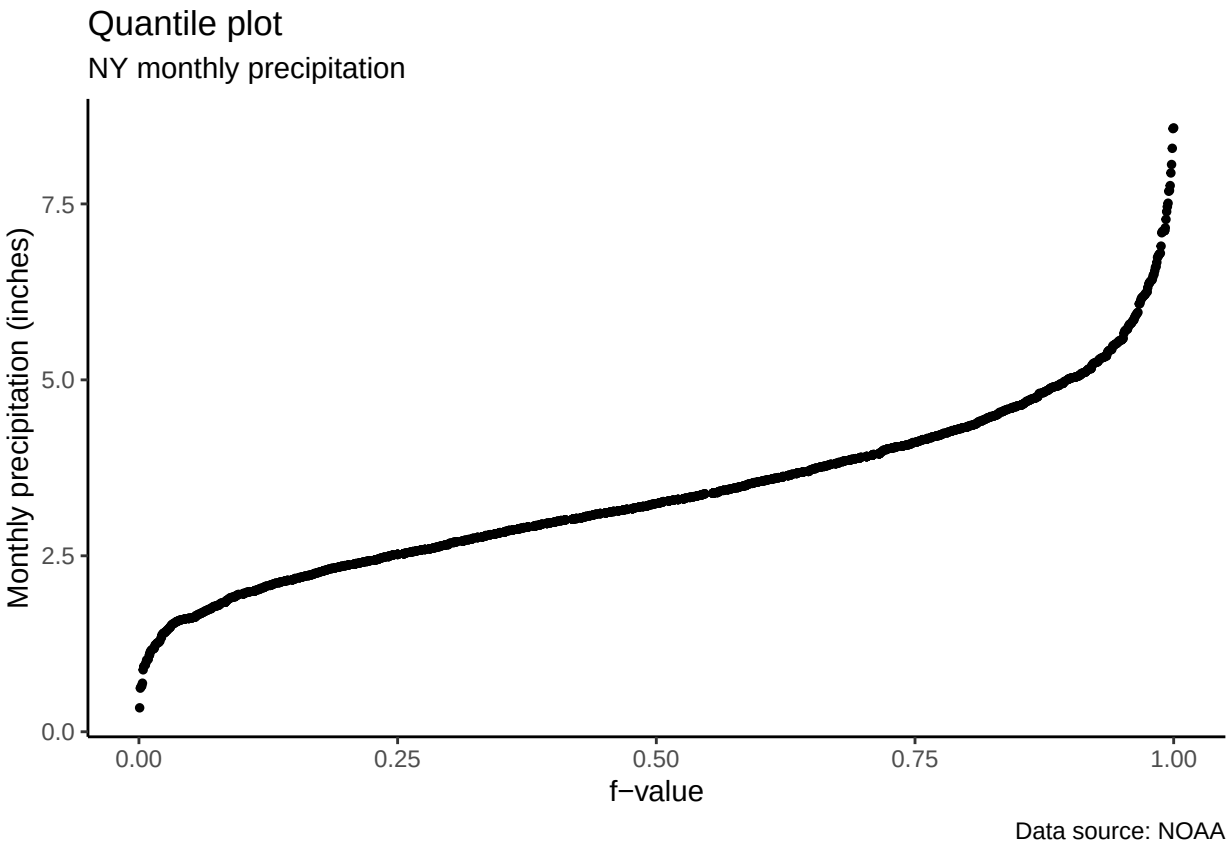
Another way of displaying a variables distribution is with **quantile plots** or with **cumulative distribution plots**, which are quantile plots where the axes have been flipped.

Note that we can use the `cume_dist()` function to compute the *f-value* for each observation (i.e. the percentage of observations $\leq$ to that observation).

Let's create a quantile plot showing the distribution of monthly precipitation in New York.

```
ny_precip %>%
  ggplot(aes(x = cume_dist(Precip_inches), y = Precip_inches)) +
  geom_point(size = 1) +
  theme_classic() +
```

```
labs(title = "Quantile plot",
     subtitle = "NY monthly precipitation",
     x = "f-value",
     y = "Monthly precipitation (inches)",
     caption = "Data source: NOAA")
```



The quantile plot is very useful when characterizing and describing the distribution of a variable's values. The **f-value** corresponding to a given value of the variable (in this case monthly precipitation) indicates that a proportion f of the observations are less than or equal to the corresponding variable value.

For instance, in the above graphic, an *f-value* of 0.25 corresponds to 2.5275 inches. This means that 25% of the observations have a precipitation value less than or equal to 2.5275 inches.

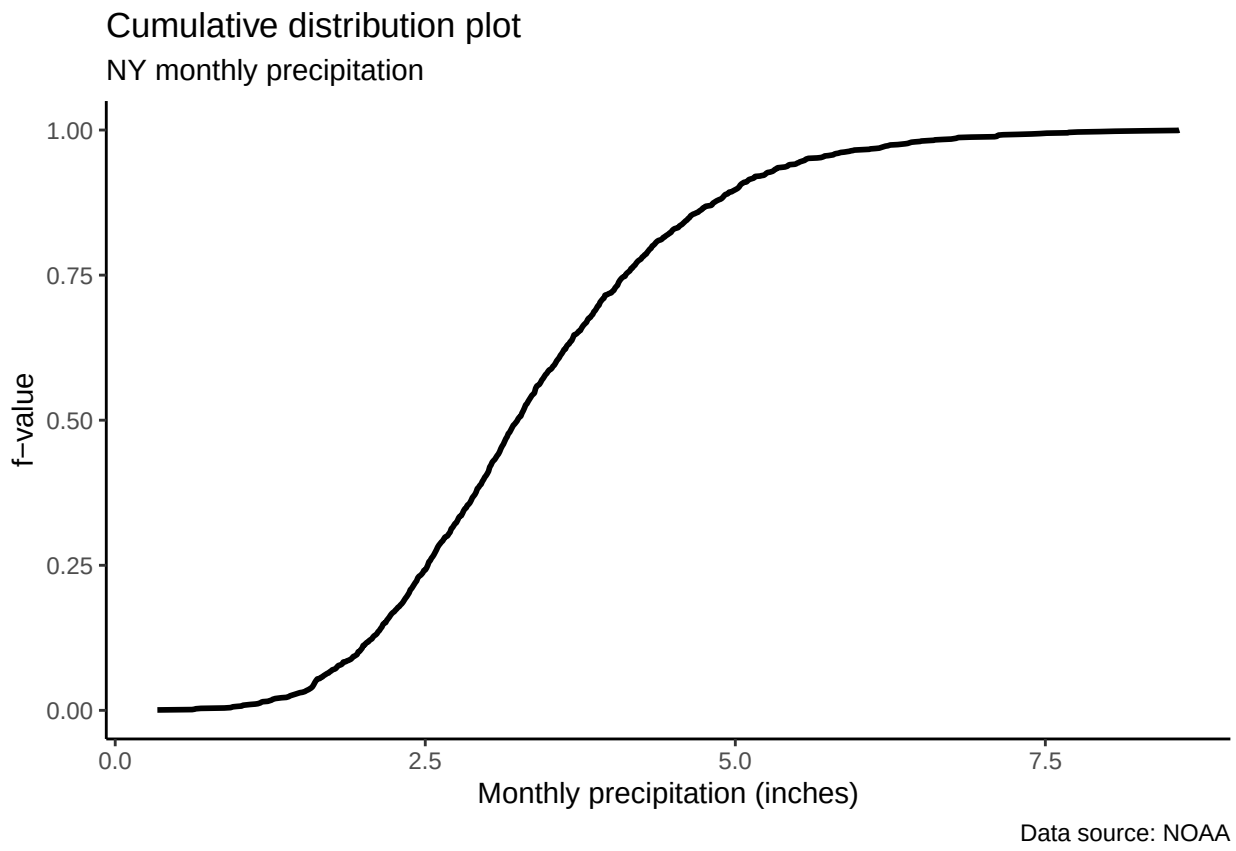
In a quantile plot, sections of the plot where the curve is flat indicate a high density of observations over the corresponding range of y-values. For example, note how the curve is very flat between *f-values* of 0.25 and 0.75. This reveals that as you go from the 25^{th} to 75^{th} percentile there is very little change in the monthly precipitation value – thus 50% of the observations fall in a very tight range of precipitation values (in this case 2.5275 to 3.25 inches).

Furthermore, steep areas of the curve indicate a low density of observations over the corresponding range of y-values. For instance, the top 10% of samples (from f-values 0.9 to 1.0) has a precipitation range from 5.025 to 8.58 inches. This is a huge range in values as we move 10 percentage points (from 90^{th} to 100^{th} percentile). Compare this with the 50% of observations between f-values of 0.25 to 0.75, that fall within a much narrower range of precipitation values (2.5275 to 3.25 inches).

Cumulative distribution plots Cumulative distribution plots are identical to a quantile plot, except the x and y axes are switched. Cumulative distributions provide another way of examining the distribution of univariate data.

Let's create a cumulative distribution plot showing the distribution of monthly precipitation in New York.

```
ny_precip %>%
  ggplot(aes(x = Precip_inches, y = cume_dist(Precip_inches))) +
  geom_line(size = 1) +
  theme_classic() +
  labs(title = "Cumulative distribution plot",
       subtitle = "NY monthly precipitation",
       x = "Monthly precipitation (inches)",
       y = "f-value",
       caption = "Data source: NOAA")
```



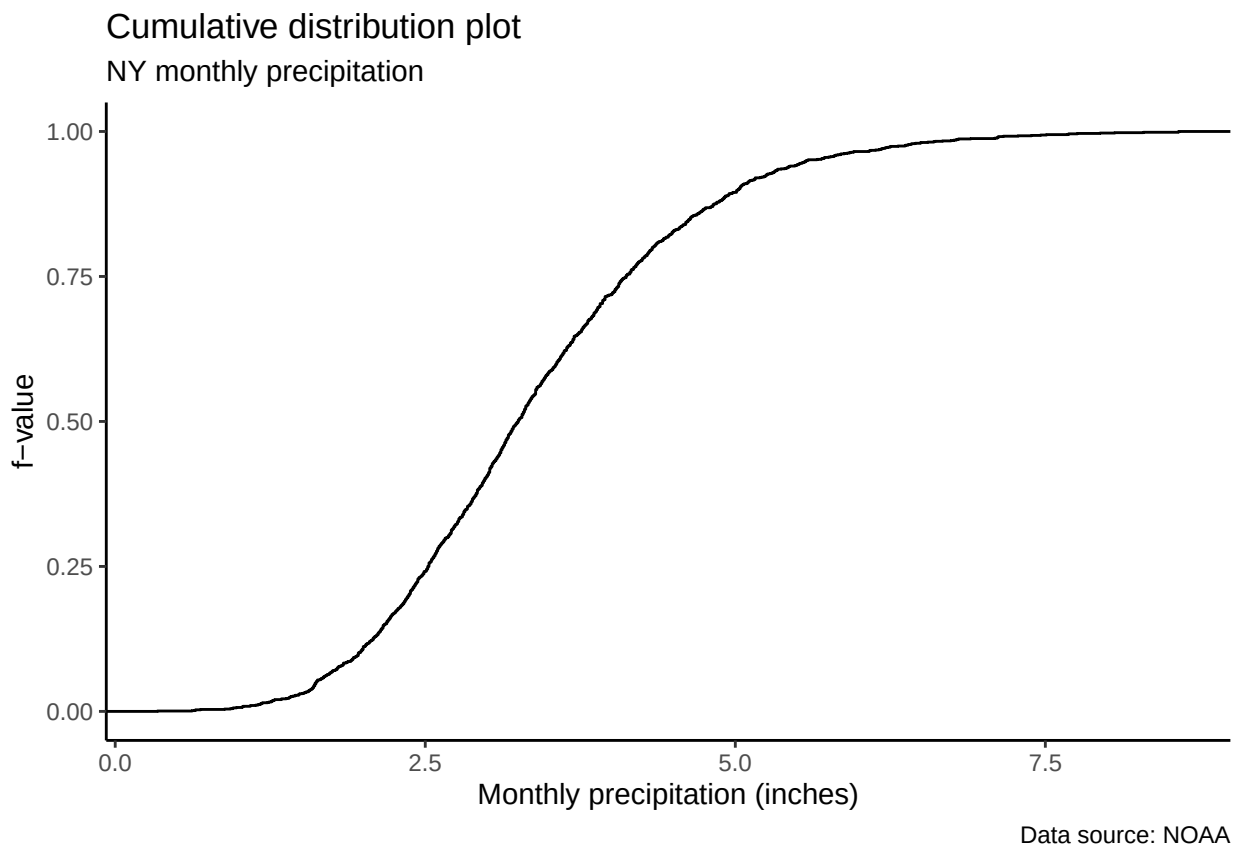
You interpret cumulative distributions similar to how you interpret quantile plots – however you need to keep in mind that their axes are switched.

Looking at cumulative distribution above, you can quickly see that 90% of the precipitation observations are less than 5.025 inches. You can also see that while monthly precipitation in excess of 7.5 inches has been observed, it is exceedingly rare as it appears that $< 1\%$ of observations exceed 7.5 inches ($> 99\%$ of observations are less than 7.5 inches).

FYI, you can also create a cumulative distribution plot using the `stat_ecdf()` function which is from the `ggplot2` package

```
ny_precip %>%
  ggplot(aes(Precip_inches)) +
  stat_ecdf() +
  theme_classic() +
  labs(title = "Cumulative distribution plot",
```

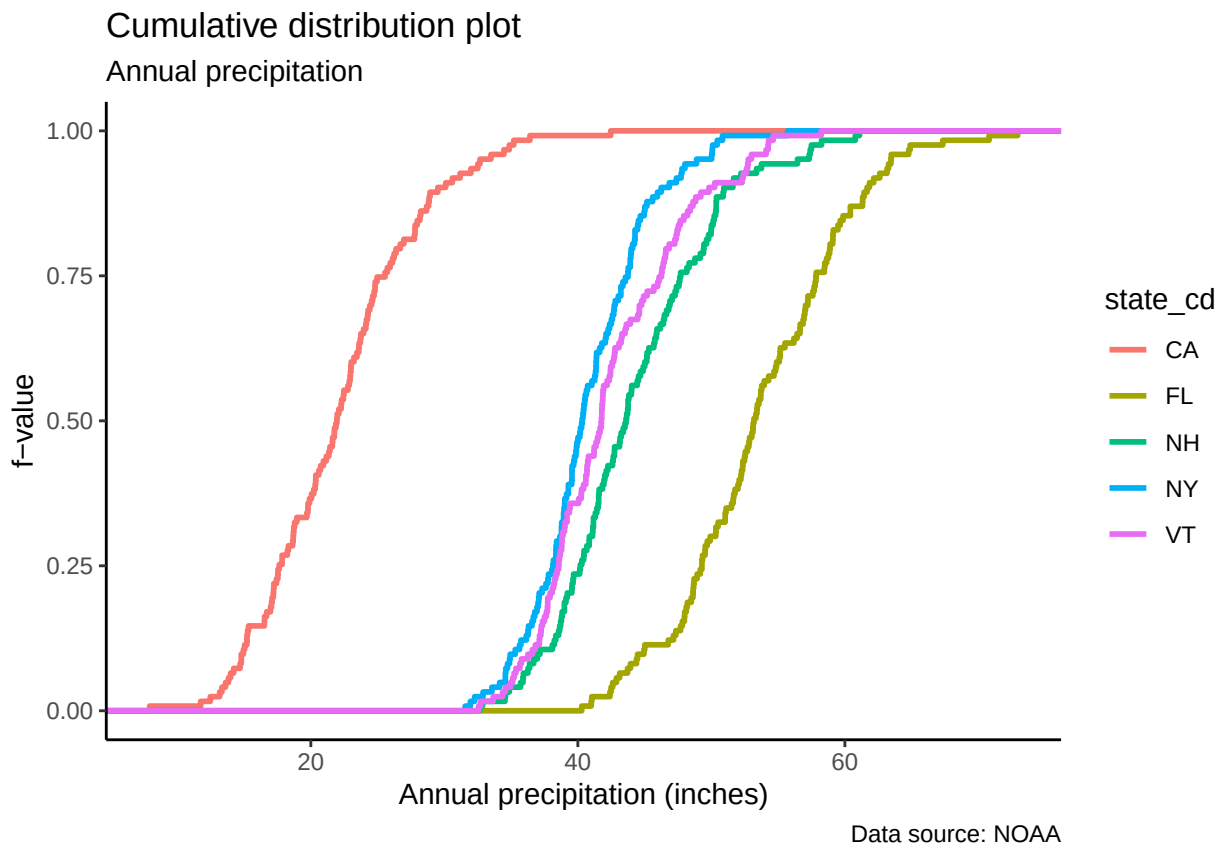
```
       subtitle = "NY monthly precipitation",
       x = "Monthly precipitation (inches)",
       y = "f-value",
       caption = "Data source: NOAA")
```



Exercise Create a cumulative distribution plot of **annual** (not monthly) precipitation

- You should show the cumulative distribution for "CA","FL","NY","VT","NH" on a single graphic (a curve for each state)
- Once you've created the graphic examine the results and think about the distribution of annual precipitation for each state (e.g. are the values tightly centered around the median, are their significant outliers,...)

```
## `summarise()` has grouped output by 'state_cd'. You can override using the
## `.groups` argument.
```



Challenge

We've been examining the distribution of monthly precipitaton data and have made some interesting observations.
We've seen that in many states the amount of precipitation varies significantly between the different months. Based on this observation I'd like to pose a challenge.

Can you determine which states have the most precipitation, as a fraction of that state's total annual precipitation, fall in just three months? Note that the months do not need to be consecutive – for instance, it could be the case that in a given state the three wettest month are Feb, May, Sept and combined they make up 60% of the total annual precipitation.

This question is more than just a programming challenge – it yields insight into how unequal (or equal) the distribution of rainfall is across the months. In an extreme case three months could provide 100% of a state's precip (very unequal distribution in time). Conversely, in a very uniform distribution of precipitation with time, three months would provide 25% of a state's precip. How evenly a state's precip is spread out with time has important ecological and societal implications.

Try giving this challenge a go (you'll need to rely on many of the `dplyr` tools you've learned).

```
# Your code here
```