IMPORTANT NOTE: This problem is optional, and will not count towards your grade. We have created this problem to give you extra practice with the topics covered in this unit.

VISUALIZING ATTRIBUTES OF PAROLE VIOLATORS (OPTIONAL)

In the crime lecture, we saw how we can use heatmaps to give a 2-dimensional representation of 3-dimensional data: we made heatmaps of crime counts by time of the day and day of the week. In this problem, we'll learn how to use histograms to show counts by one variable, and then how to visualize 3 dimensions by creating multiple histograms.

We'll use the parole data parole.csv from Unit 3. Before, we used this data to predict parole violators. Now, let's try to get a little more insight into this dataset using histograms. As a reminder, the variables in this dataset are:

- **male** = 1 if the parolee is male, 0 if female
- **race** = 1 if the parolee is white, 2 otherwise
- **age** = the parolee's age in years at the time of release from prison
- **state** = a code for the parolee's state. 2 is Kentucky, 3 is Louisiana, 4 is Virginia, and 1 is any other state. These three states were selected due to having a high representation in the dataset.
- **time.served** = the number of months the parolee served in prison (limited by the inclusion criteria to not exceed 6 months).
- **max.sentence** = the maximum sentence length for all charges, in months (limited by the inclusion criteria to not exceed 18 months).
- **multiple.offenses** = 1 if the parolee was incarcerated for multiple offenses, 0 otherwise.
- **crime** = a code for the parolee's main crime leading to incarceration. 2 is larceny, 3 is drug-related crime, 4 is driving-related crime, and 1 is any other crime.
- **violator** = 1 if the parolee violated the parole, and 0 if the parolee completed the parole

without violation.

PROBLEM 1.1 - LOADING THE DATA

Using the read.csv function, load the dataset parole.csv and call it parole. Since male, state, and crime are all unordered factors, convert them to factor variables using the following commands:

parole$male = as.factor(parole$male)

parole$state = as.factor(parole$state)

parole$crime = as.factor(parole$crime)

What fraction of parole violators are female?

? **Answer:** 0.1794872

EXPLANATION

This can be found by using table:

table(parole$male, parole$violator)

The total number of violators is 78, and 14 of them are female.

You have used 0 of 3 submissions

PROBLEM 1.2 - LOADING THE DATA

In this dataset, which crime is the most common in Kentucky?

- Larceny
- Drug-related crime ✔
- Driving-related crime
- Other

?

EXPLANATION

This can be found by using table:

table(parole$state, parole$crime)

The code 2 corresponds to Kentucky, and the most common crime is 3, which corresponds to Drug-related crime.

You have used 0 of 1 submissions

PROBLEM 2.1 - CREATING A BASIC HISTOGRAM

Recall from lecture that in ggplot, we need to specify the dataset, the aesthetic, and the geometry. To create a histogram, the geometry will be geom_histogram. The data we'll use is parole, and the aesthetic will be the map from a variable to the x-axis of the histogram.

Create a histogram to find out the distribution of the age of parolees, by typing the following command in your R console (you might need to load the ggplot2 package first by typing library(ggplot2) in your R console):

ggplot(data = parole, aes(x = age)) + geom_histogram()

By default, geom_histogram divides the data into 30 bins. Change the width of the bins to 5 years by adding the argument "binwidth = 5" to geom_histogram.

Note that by default, histograms create bins where the left endpoint is included in the bin, but the right endpoint isn't. So the first bin in this histogram represents parolees who are

between 15 and 19 years old. The last bin in this histogram represents parolees who are between 65 and 69 years old.

What is the age bracket with the most parolees?

- 20-24 ✔
- 25-29
- 30-34
- 35-39

?

EXPLANATION

You can generate the histogram with a bin width of 5 with the command:

ggplot(data = parole, aes(x = age)) + geom_histogram(binwidth=5)

The tallest bar corresponds to the age bracket with the most parolees, which is 20-24.

You have used 0 of 1 submissions

PROBLEM 2.2 - CREATING A BASIC HISTOGRAM

Redo the histogram, adding the following argument to the geom_histogram function: color="blue". What does this do? Select all that apply.

- [ ] Changes the fill color of the bars
- [ ] Changes the background color of the plot
- [x] Changes the outline color of the bars
- [ ] Changes the color of the axis labels

?

EXPLANATION

You can generate the histogram by typing:

ggplot(data = parole, aes(x = age)) + geom_histogram(binwidth=5, color="blue")

Adding the color argument changes the outline color of the bars.

You have used 0 of 2 submissions

PROBLEM 3.1 - ADDING ANOTHER DIMENSION

Now suppose we are interested in seeing how the age distribution of male parolees compares to the age distribution of female parolees.

One option would be to create a heatmap with age on one axis and male (a binary variable in our data set) on the other axis. Another option would be to stick with histograms, but to create a separate histogram for each gender. ggplot has the ability to do this automatically using the facet_grid command.

To create separate histograms for male and female, type the following command into your R console:

ggplot(data = parole, aes(x = age)) + geom_histogram(binwidth = 5) + facet_grid(male ~ .)

The histogram for female parolees is shown at the top, and the histogram for male parolees is shown at the bottom.

What is the age bracket with the most female parolees?

- 20-24
- 25-29
- 30-34
- 35-39 ✔

?

EXPLANATION

Looking at the histogram at the top, we can see that the tallest bar corresponds to the age bracket 35-39.

You have used 0 of 1 submissions

PROBLEM 3.2 - ADDING ANOTHER DIMENSION

Now change the facet_grid argument to be ".~male" instead of "male~.". What does this do?

- Creates histograms of the male variable, sorted by the different values of age.
- Puts the histograms side-by-side instead of on top of each other. ✔
- This doesn't change anything - the plot looks exactly the same as it did before.

?

EXPLANATION

You can create the new plot with the command:

ggplot(data = parole, aes(x = age)) + geom_histogram(binwidth = 5) + facet_grid(.~male)

This puts the plots side-by-side instead of on top of each other.

You have used 0 of 1 submissions

PROBLEM 3.3 - ADDING ANOTHER DIMENSION

An alternative to faceting is to simply color the different groups differently. To color the data points by group, we need to tell ggplot that a property of the data (male or not male) should be translated to an aesthetic property of the histogram. We can do this by setting the fill parameter within the aesthetic to male.

Run the following command in your R console to produce a histogram where data points are colored by group:

ggplot(data = parole, aes(x = age, fill = male)) + geom_histogram(binwidth = 5)

Since we didn't specify colors to use, ggplot will use its default color selection. Let's change this by defining our own color palette. First, type in your R console:

colorPalette = c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")

This is actually a colorblind-friendly palette, desribed on this Cookbook for R page. Now, generate your histogram again, using colorPalette, with the following command:

ggplot(data = parole, aes(x = age, fill = male)) + geom_histogram(binwidth = 5) + scale_fill_manual(values=colorPalette)

What color is the histogram for the female parolees?

- Orange
- Black ✔

?

EXPLANATION

From the previous question, we saw that the female parolee histogram was much

smaller than the male parolee histogram. So it looks like the female histogram is the black-colored one. We can also read this from the legend.

You have used 0 of 1 submissions

PROBLEM 3.4 - ADDING ANOTHER DIMENSION

Coloring the groups differently is a good way to see the breakdown of age by sex within the single, aggregated histogram. However, the bars here are stacked, meaning that the height of the orange bars in each age bin represents the total number of parolees in that age bin, not just the number of parolees in that group.

An alternative to a single, stacked histogram is to create two histograms and overlay them on top of each other. This is a simple adjustment to our previous command.

We just need to:

1) Tell ggplot not to stack the histograms by adding the argument position="identity" to the geom_histogram function.

2) Make the bars semi-transparent so we can see both colors by adding the argument alpha=0.5 to the geom_histogram function.

Redo the plot, making both of these changes.

Which of the following buckets contain no female paroles? Select all that apply.

- [x] 15-19
- [ ] 20-24
- [ ] 25-29
- [ ] 30-34
- [ ] 35-39
- [ ] 40-44
- [ ] 45-49
- [ ] 50-54
- [x] 55-59
- [ ] 60-64
- [x] 65-69

?

EXPLANATION

This plot can be generated with the following command:

ggplot(parole, aes(x = age, fill = male)) + geom_histogram(binwidth = 5, position = "identity", alpha = 0.5) + scale_fill_manual(values=colorPalette)

If you look at the plot, you can see that there are no female parolees in the age groups 15-19, 55-59, and 65-69 (the bars have height zero).

You have used 0 of 2 submissions

PROBLEM 4.1 - TIME SERVED

Now let's explore another aspect of the data: the amount of time served by parolees. Create a basic histogram like the one we created in Problem 2, but this time with time.served on the x-axis. Set the bin width to one month.

What is the most common length of time served, according to this histogram?

- Between 2 and 3 months
- Between 3 and 4 months
- Between 4 and 5 months ✔
- Between 5 and 6 months

?

EXPLANATION

You can create this histogram with the following command:

ggplot(data = parole, aes(x = time.served)) + geom_histogram(binwidth = 1)

The highest bar corresponds to between 4 and 5 months.

You have used 0 of 1 submissions

PROBLEM 4.2 - TIME SERVED

Change the binwidth to 0.1 months. Now what is the most common length of time served, according to the histogram?

- Between 2.1 and 2.2 months
- Between 3.0 and 3.1 months ✔
- Between 4.2 and 4.3 months
- Between 4.8 and 4.9 months

?

EXPLANATION

You can change the binwidth by using the following command:

ggplot(data = parole, aes(x = time.served)) + geom_histogram(binwidth = .1)

Now, the highest bar corresponds to between 3.0 and 3.1 months.

Be careful when choosing the binwidth - it can significantly affect the interpretation of a histogram! When visualizing histograms, it is always a good idea to vary the bin size in order to understand the data at various granularities.

You have used 0 of 1 submissions

PROBLEM 4.3 - TIME SERVED

Now, suppose we suspect that it is unlikely that each type of crime has the same distribution of time served. To visualize this, change the binwidth back to 1 month, and use facet_grid to create a separate histogram of time.served for each value of the variable crime.

Which crime type has no observations where time served is less than one month? Recall that crime type #2 is larceny, #3 is drug-related crime, #4 is driving-related crime, and #1 is any other crime.

- Larceny
- Drug-related
- Driving-related ✔
- Other

?

For which crime does the frequency of 5-6 month prison terms exceed the frequencies of each other term length?

- Larceny
- Drug-related ✔
- Driving-related
- Other

?

EXPLANATION

This histogram can be generated using the command:

ggplot(data = parole, aes(x = time.served)) + geom_histogram(binwidth = 1) + facet_grid(crime ~ .)

You have used 0 of 1 submissions

PROBLEM 4.4 - TIME SERVED

Now, instead of faceting the histograms, overlay them. Remember to set the position and alpha parameters so that the histograms are not stacked. Also, make sure to indicate that the

fill aesthetic should be "crime".

In this case, faceting seems like a better alternative. Why?

- With four different groups, it can be hard to tell them apart when they are overlayed. ✔
- ggplot doesn't let us overlay plots with more than two groups.
- Overlaying the plots doesn't allow us to observe which crime type is the most common.

?

EXPLANATION

You can generate this plot with the following command:

ggplot(data=parole, aes(x=time.served, fill=crime)) + geom_histograph(binwidth=1, position="identity", alpha=0.5)

While overlaying the plots is allowed and lets us observe some attributes of the plots like the most common crime type, it can be hard to tell them apart and if they have similar values it can be hard to read.

You have used 0 of 1 submissions

Please remember not to ask for or post complete answers to homework questions in this discussion forum.