Formative Assessment 3

PATAYON, SPIKE LEE-ROY V

2025-02-25

Instruction

1. Create a histogram on the diamonds dataset, for example with

ggplot() + geom_histogram(aes(x = carat), data = diamonds)

Re-write this using the layer function like we did in class. Hint: if you don’t know what the default values for some of the aspects of the plot, examine p$layers.

2. Remember that a histogram is a plot with stat_bin and geom_bar. Modify your histogram code so that it uses a different geom, for example geom_line or geom_point. This should be simple once you have the layer specification of a histogram.

3. In your histogram (the one plotted with bars that you created in question 1), add an aesthetic mapping from one of the factor variables (maybe color or clarity) to the fill or color aesthetic.

4. What is the default position adjustment for a histogram? Try changing the position adjustment in the histogram you created in question 3 to something different (hint: try dodge).

load the ggplot2:

```
library(ggplot2)
```

use data() to get the diamond data set:

```
data("diamonds")
```

display diamonds:

```
head(diamonds)
```

```
## # A tibble: 6 × 10
##   carat cut       color clarity depth table price     x     y     z
##   <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
## 2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
## 3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
## 4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
## 5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
## 6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
```

for extra information, we will summarize each variable in the data set:

```
summary(diamonds)

##      carat               cut        color        clarity          depth
##  Min.   :0.2000   Fair     : 1610   D: 6775   SI1    :13065   Min.
:43.00
##  1st Qu.:0.4000   Good     : 4906   E: 9797   VS2    :12258   1st
Qu.:61.00
##  Median :0.7000   Very Good:12082   F: 9542   SI2    : 9194   Median
:61.80
##  Mean   :0.7979   Premium  :13791   G:11292   VS1    : 8171   Mean
:61.75
##  3rd Qu.:1.0400   Ideal    :21551   H: 8304   VVS2   : 5066   3rd
Qu.:62.50
##  Max.   :5.0100                     I: 5422   VVS1   : 3655   Max.
:79.00
##                                     J: 2808   (Other): 2531
##      table           price               x                y
##  Min.   :43.00   Min.   :  326   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:56.00   1st Qu.:  950   1st Qu.: 4.710   1st Qu.: 4.720
##  Median :57.00   Median : 2401   Median : 5.700   Median : 5.710
##  Mean   :57.46   Mean   : 3933   Mean   : 5.731   Mean   : 5.735
##  3rd Qu.:59.00   3rd Qu.: 5324   3rd Qu.: 6.540   3rd Qu.: 6.540
##  Max.   :95.00   Max.   :18823   Max.   :10.740   Max.   :58.900
##
##        z
##  Min.   : 0.000
##  1st Qu.: 2.910
##  Median : 3.530
##  Mean   : 3.539
##  3rd Qu.: 4.040
##  Max.   :31.800
##
```

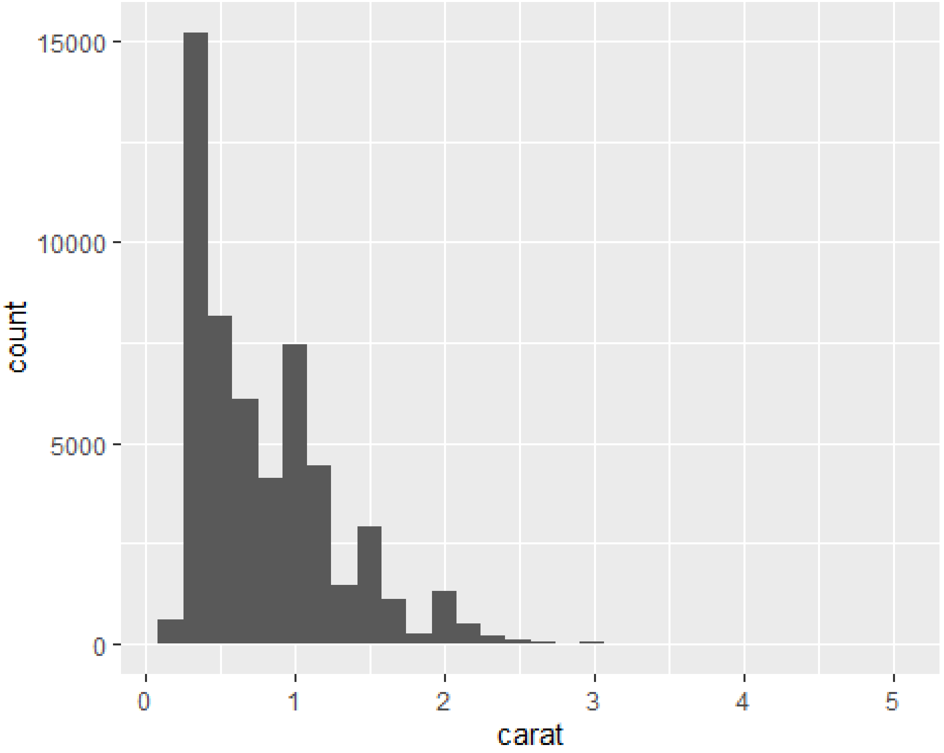
1. create a histogram on the diamonds dataset, for example with

ggplot() + geom_histogram(aes(x = carat), data = diamonds)

Re-write this using the layer function like we did in class. Hint: if you don’t know what the default values for some of the aspects of the plot, examine p$layers.

```
p <- ggplot(data = diamonds) +
  layer(
    mapping = aes(x = carat),
    stat = "bin",
    geom = "bar",
    position = "stack",
    data = diamonds
  )
print(p)
```

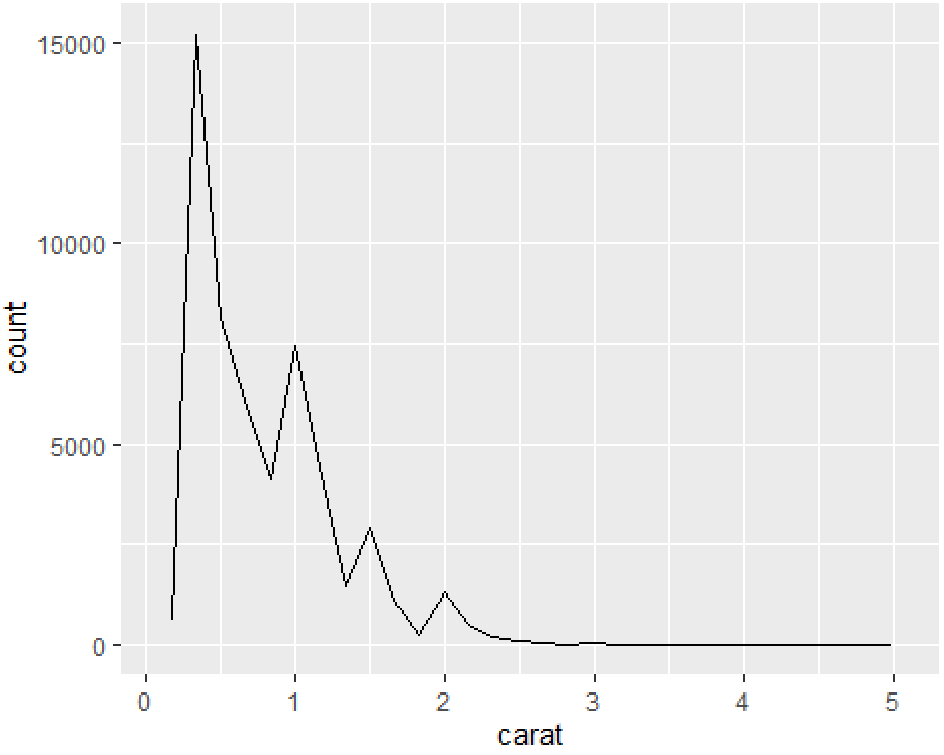
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



2. Remember that a histogram is a plot with stat_bin and geom_bar. Modify your histogram code so that it uses a different geom, for example geom_line or geom_point. This should be simple once you have the layer specification of a histogram.

```r
p_line <- ggplot(data = diamonds) +
  layer(
    mapping = aes(x = carat, y = after_stat(count)),
    stat = "bin",
    geom = "line",
    position = "identity",
    data = diamonds
  )
print(p_line)
```

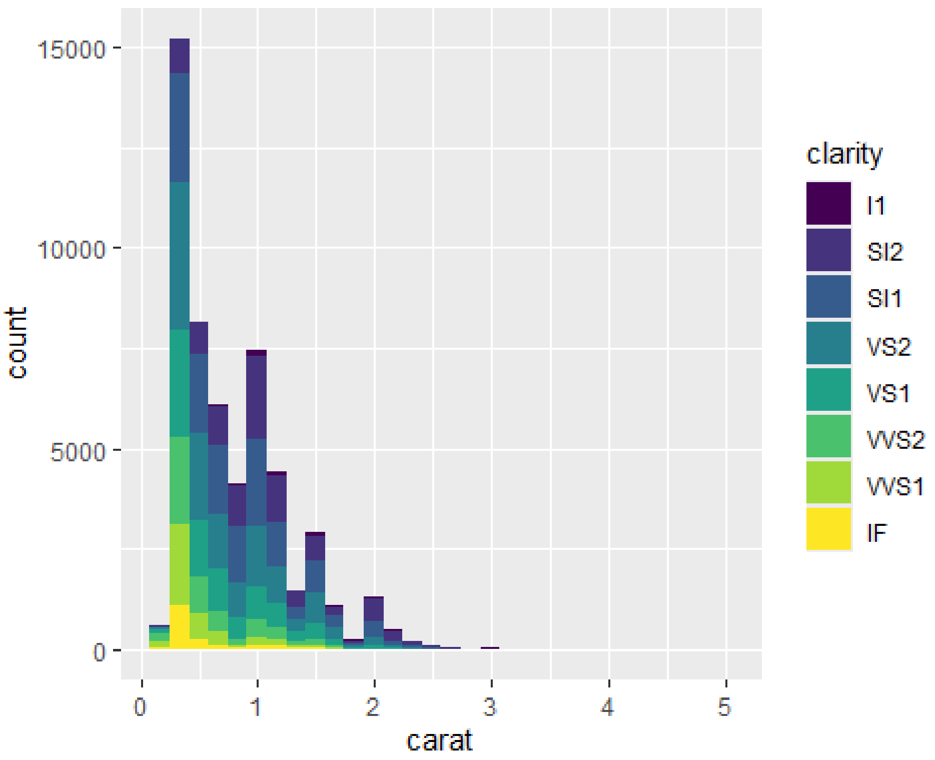
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



3. In your histogram (the one plotted with bars that you created in question 1), add an aesthetic mapping from one of the factor variables (maybe color or clarity) to the fill or color aesthetic.

```
p_fill <- ggplot(data = diamonds) +
  layer(
    mapping = aes(x = carat, fill = clarity),
    stat = "bin",
    geom = "bar",
    position = "stack",
    data = diamonds
  )
print(p_fill)
```

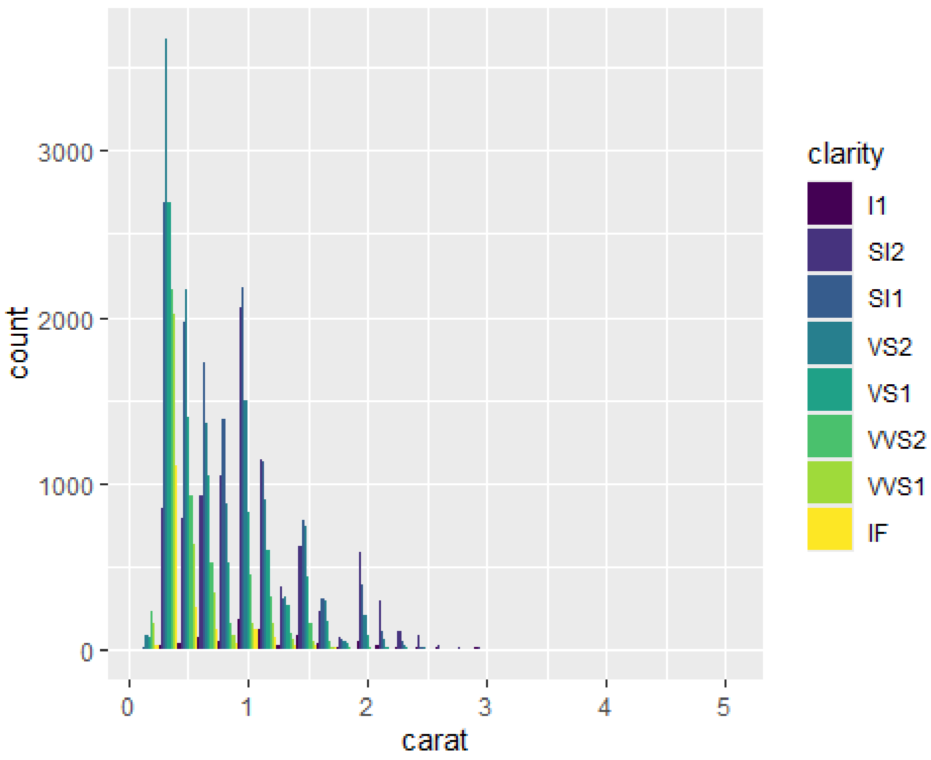
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



4. What is the default position adjustment for a histogram? Try changing the position adjustment in the histogram you created in question 3 to something different (hint: try dodge).

```
p_dodge <- ggplot(data = diamonds) +
  layer(
    mapping = aes(x = carat, fill = clarity),
    stat = "bin",
    geom = "bar",
    position = "dodge",
    data = diamonds
  )
print(p_dodge)

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


count
3000
2000
1000
0
0
1
2
3
4
5
carat
clarity
I1
SI2
SI1
VS2
VS1
VVS2
VVS1
IF