HW2

Jordan Hilton

April 8, 2019

Let's load our data:

```
d <- rd("Cars93.csv")
```

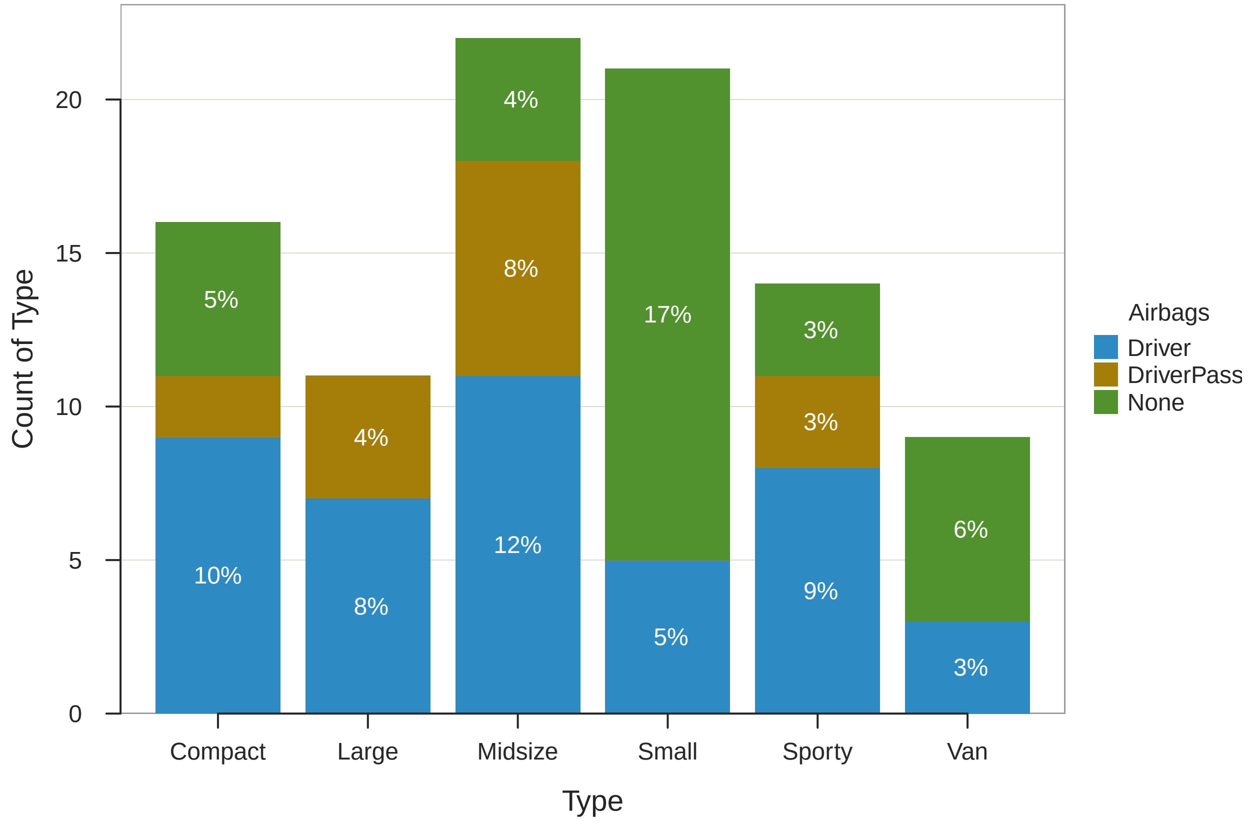
```
##
## >>> Suggestions
## To read a csv or Excel file of variable labels, var.labels=TRUE
##    Each row of the file:  Variable Name, Variable Label
## Details about your data, Enter:  details()  for d, or  details(name)
##
## Data Types
## ------------------------------------------------------------
## character: Non-numeric data values
## integer: Numeric data values, integers only
## double: Numeric data values with decimal digits
## ------------------------------------------------------------
##
##      Variable                  Missing  Unique
##          Name     Type  Values  Values  Values   First and last values
## ------------------------------------------------------------------------------------------
##  1        Make character     93       0      32   Acura  Acura ... Volvo  Volvo
##  2        Type character     93       0       6   Small  Midsize ... Compact  Midsize
##  3    MinPrice    double     93       0      79   12.9  29.2  25.9 ... 22.9  21.8  24.8
##  4    MidPrice    double     93       0      81   15.9  33.9  29.1 ... 23.3  22.7  26.7
##  5    MaxPrice    double     93       0      79   18.8  38.7  32.3 ... 23.7  23.5  28.5
##  6     MPGcity   integer     93       0      21   25  18  20 ... 18  21  20
##  7    MPGhiway   integer     93       0      22   31  25  26 ... 25  28  28
##  8     Airbags character     93       0       3   None  DriverPass ... Driver  DriverPass
##  9  DriveTrain character     93       0       3   Front  Front  Front ... Front  Rear  Front
## 10   Cylinders character     93       0       6   4  6  6 ... 6  4  5
## 11      Engine    double     93       0      26   1.8  3.2  2.8 ... 2.8  2.3  2.4
## 12          HP   integer     93       0      57   140  200  172 ... 178  114  168
## 13         RPM   integer     93       0      24   6300  5500  5500 ... 5800  5400  6200
## 14     RevMile   integer     93       0      78   2890  2335  2280 ... 2385  2215  2310
## 15      Manual   integer     93       0       2   1  1  1 ... 1  1  1
## 16     FuelCap    double     93       0      38   13.2  18  16.9 ... 18.5  15.8  19.3
## 17     PassCap   integer     93       0       6   5  5  5 ... 4  5  5
## 18      Length   integer     93       0      51   177  195  180 ... 159  190  184
## 19   Wheelbase   integer     93       0      27   102  115  102 ... 97  104  105
## 20       Width   integer     93       0      16   68  71  67 ... 66  67  69
## 21       Uturn   integer     93       0      14   37  38  37 ... 36  37  38
## 22    RearSeat character     93       0      25   26.5  30  28 ... 26  29.5  30
## 23      LugCap character     93       0      17   11  15  14 ... 15  14  15
## 24      Weight   integer     93       0      81   2705  3560  3375 ... 2810  2985  3245
## 25      Source character     93       0       2   nonUSA  nonUSA ... nonUSA  nonUSA
## ------------------------------------------------------------------------------------------
```

1 Bar Chart

a.

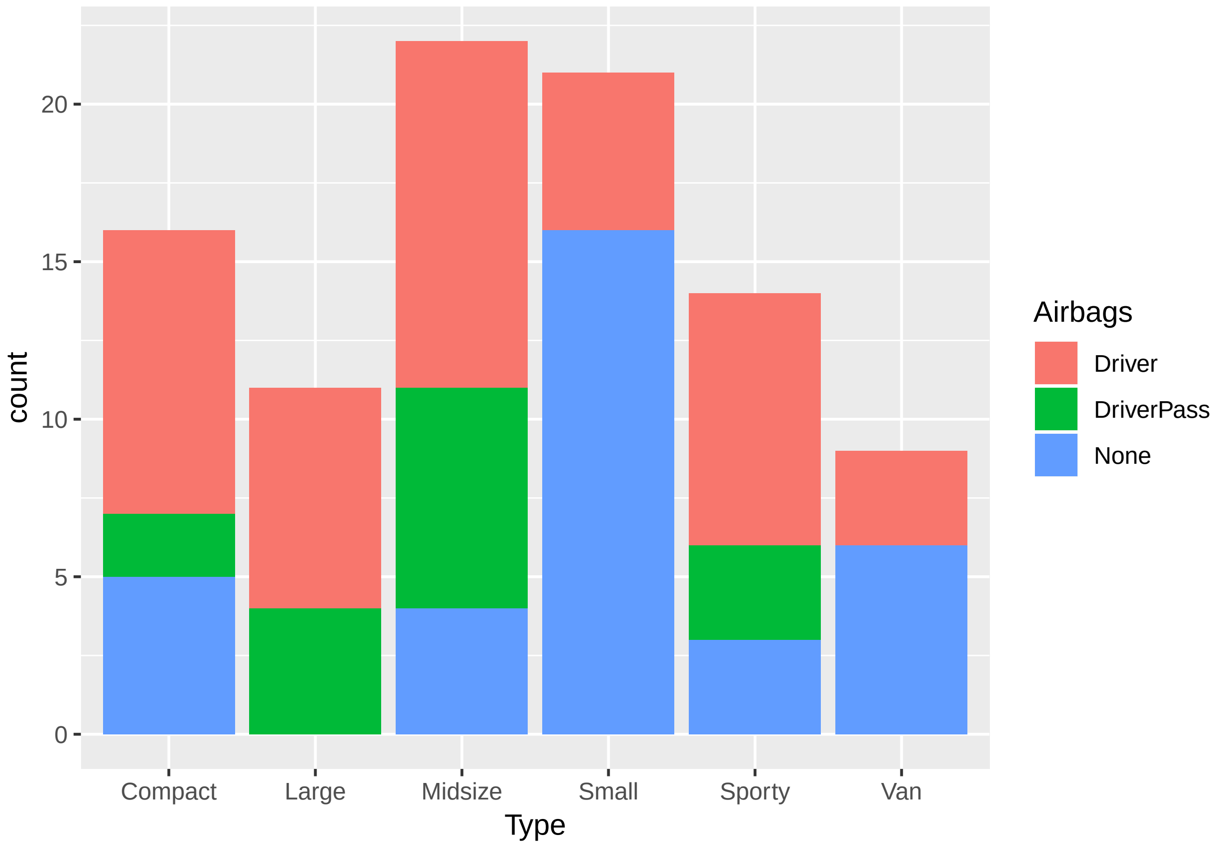
Here's the bar chart for type of car by airbag configuration in lessR:

```r
bc(Type, by=Airbags, quiet=TRUE)
```



and ggplot2:

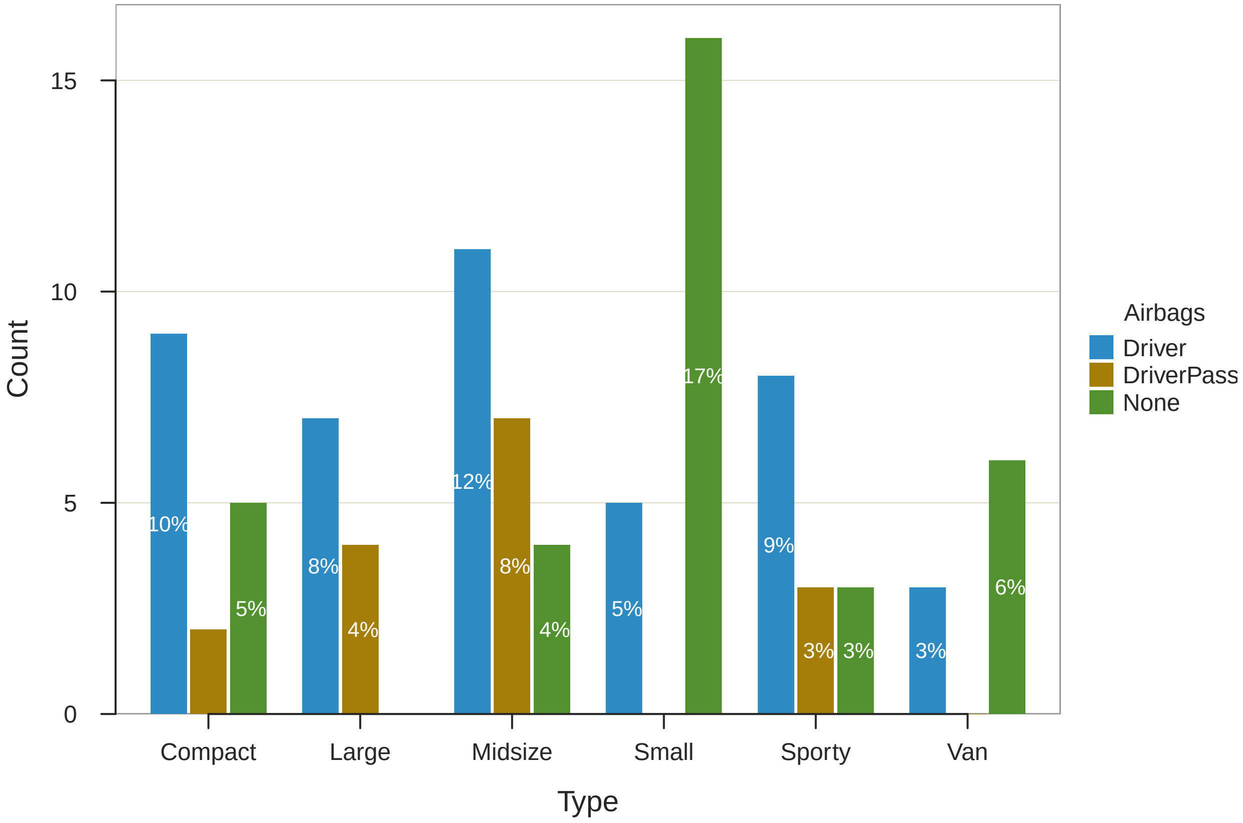
```r
ggplot(d, aes(Type, fill=Airbags)) + geom_bar()
```



b.

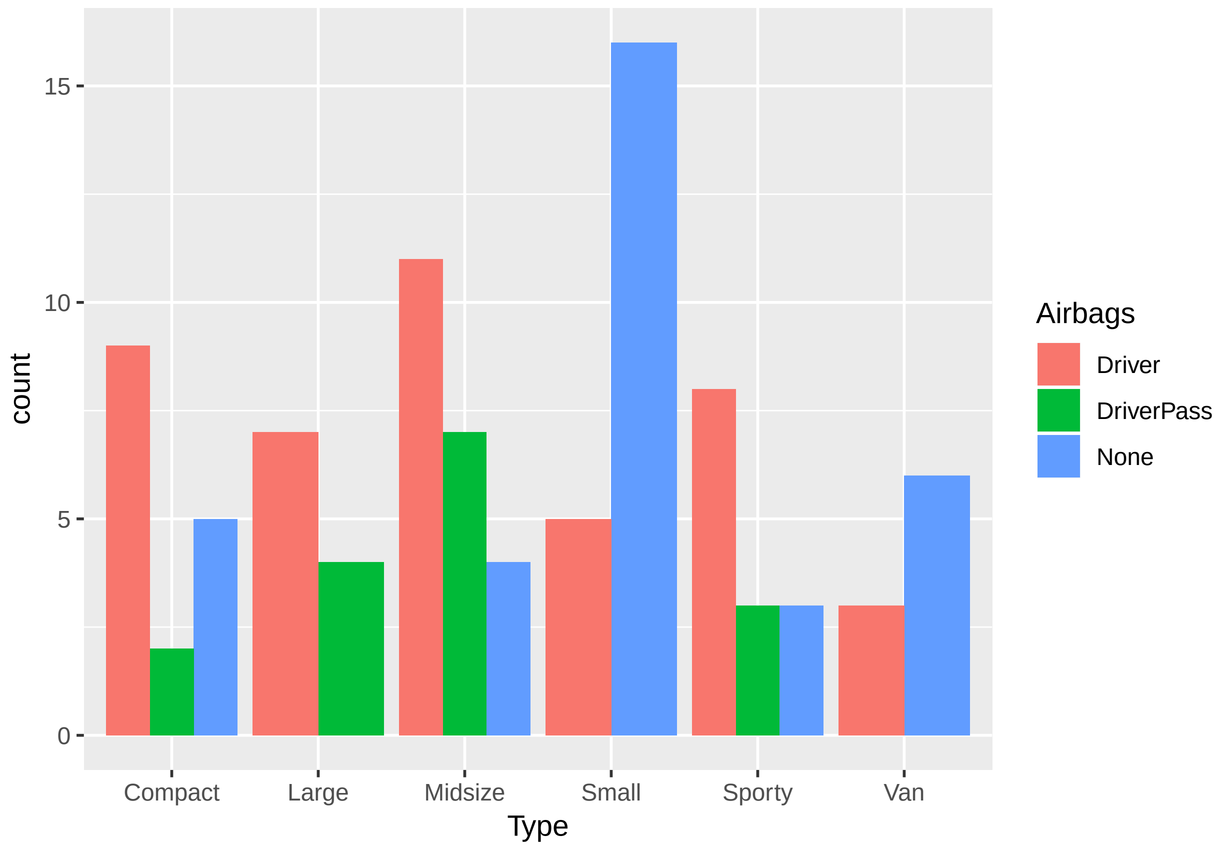
Here's the side-by-side bar chart in lessR for the same data:

```
bc(Type, by=Airbags, beside=TRUE, quiet=TRUE)
```



and ggplot2:

```r
ggplot(d, aes(Type, fill=Airbags)) + geom_bar(position="dodge")
```



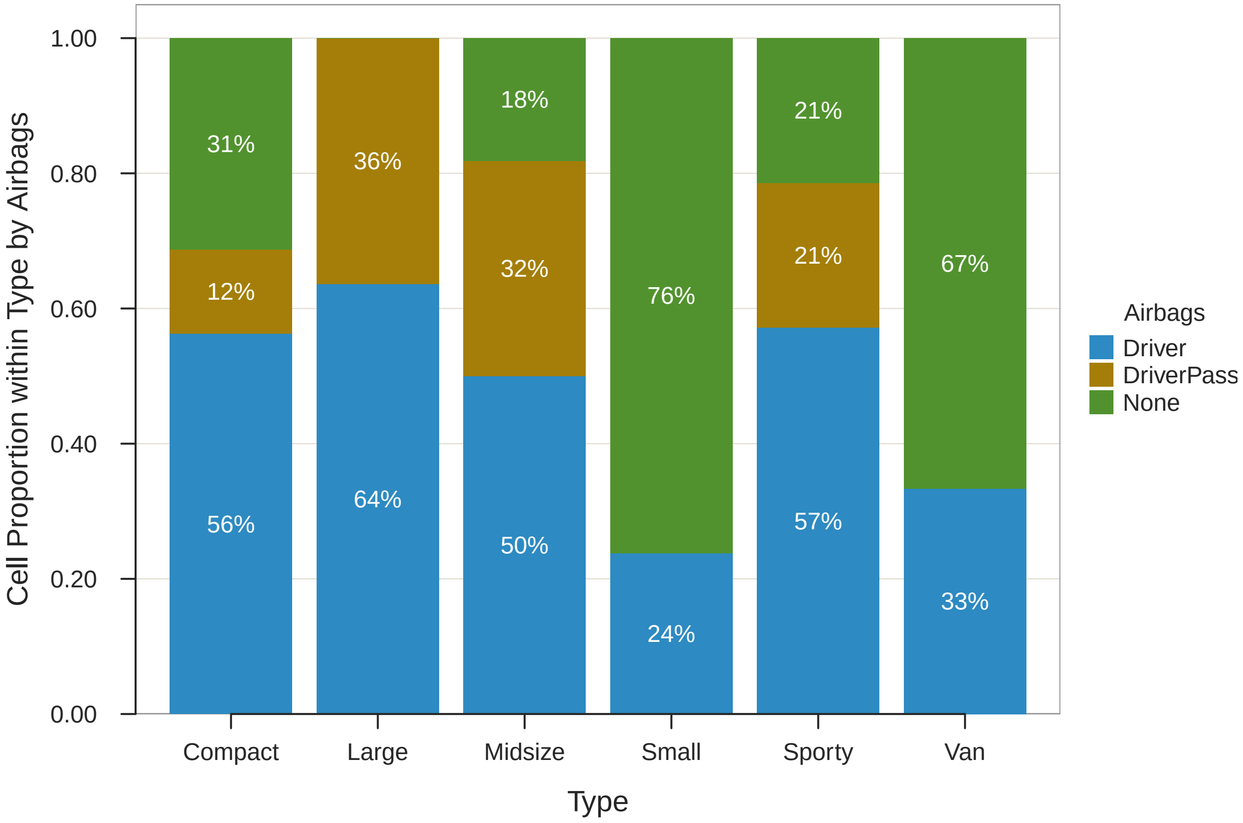
c.

Small cars frequently have no airbags- seems unsafe! It seems like midsize and large cars most frequently have both driver and passenger side airbags.

d.

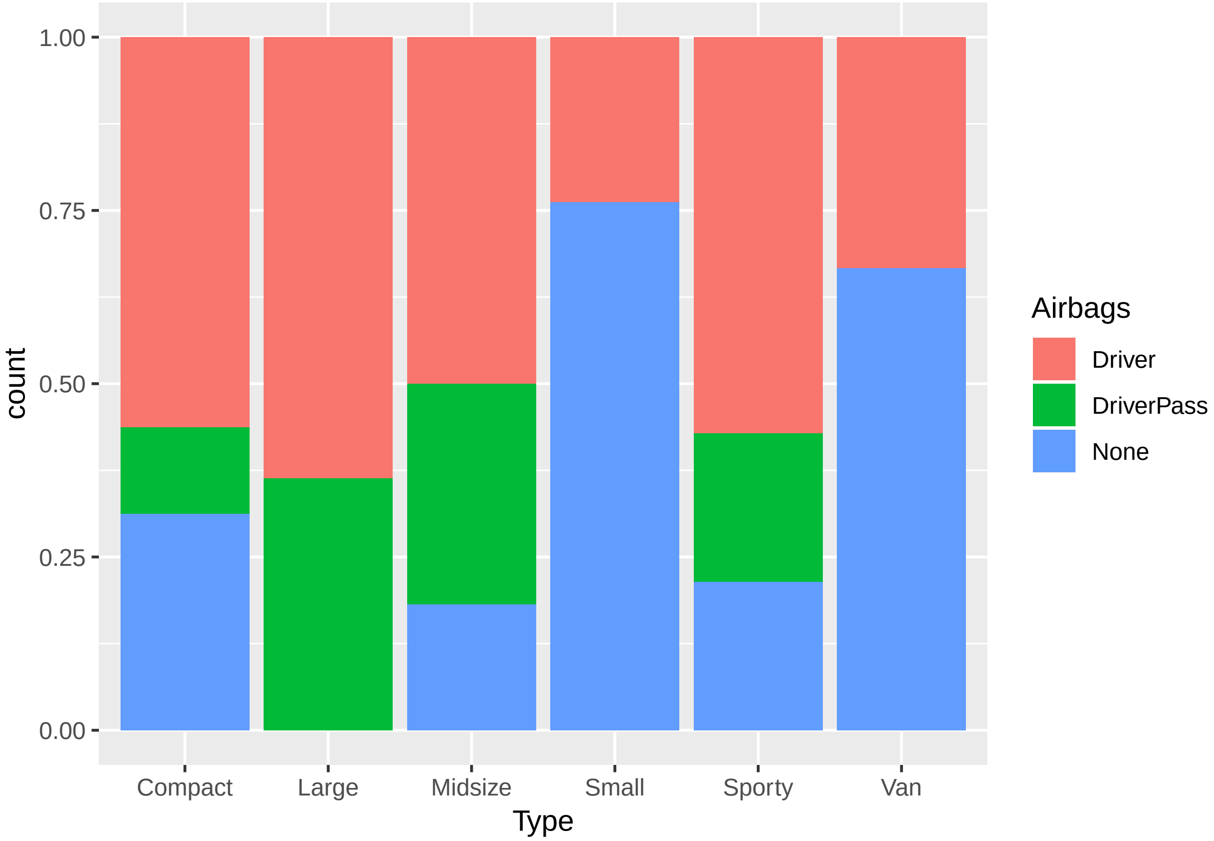
Here's the lessR bar chart by proportion:

```
bc(Type, by=Airbags, quiet=TRUE, stat.x="proportion")
```



and ggplot2:

```
ggplot(d, aes(Type, fill=Airbags)) + geom_bar(position="fill")
```



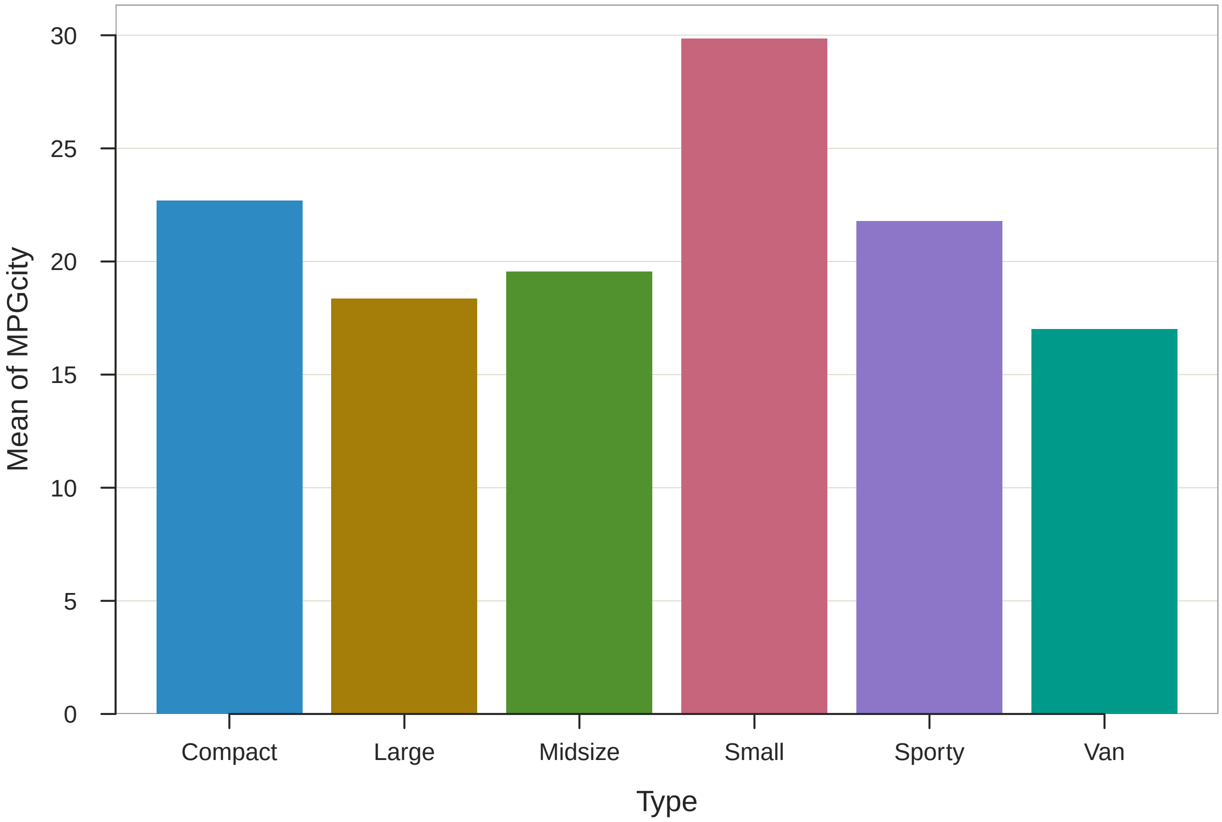
e.

Here the proportions are clearer- we can see that small cars and vans both have large proportions of vehicles with no airbags, and that large and midsize vehicles have the highest proportions of vehicles with both driver and passenger airbags.

f.

I'm interpreting this question to be asking for the mean of city MPG by type of car (since summing the city MPG of different models of cars doesn't make much sense); here's the relevant bar chart.

```
bc(Type, y=MPGcity, stat.yx="mean", quiet=TRUE)
```



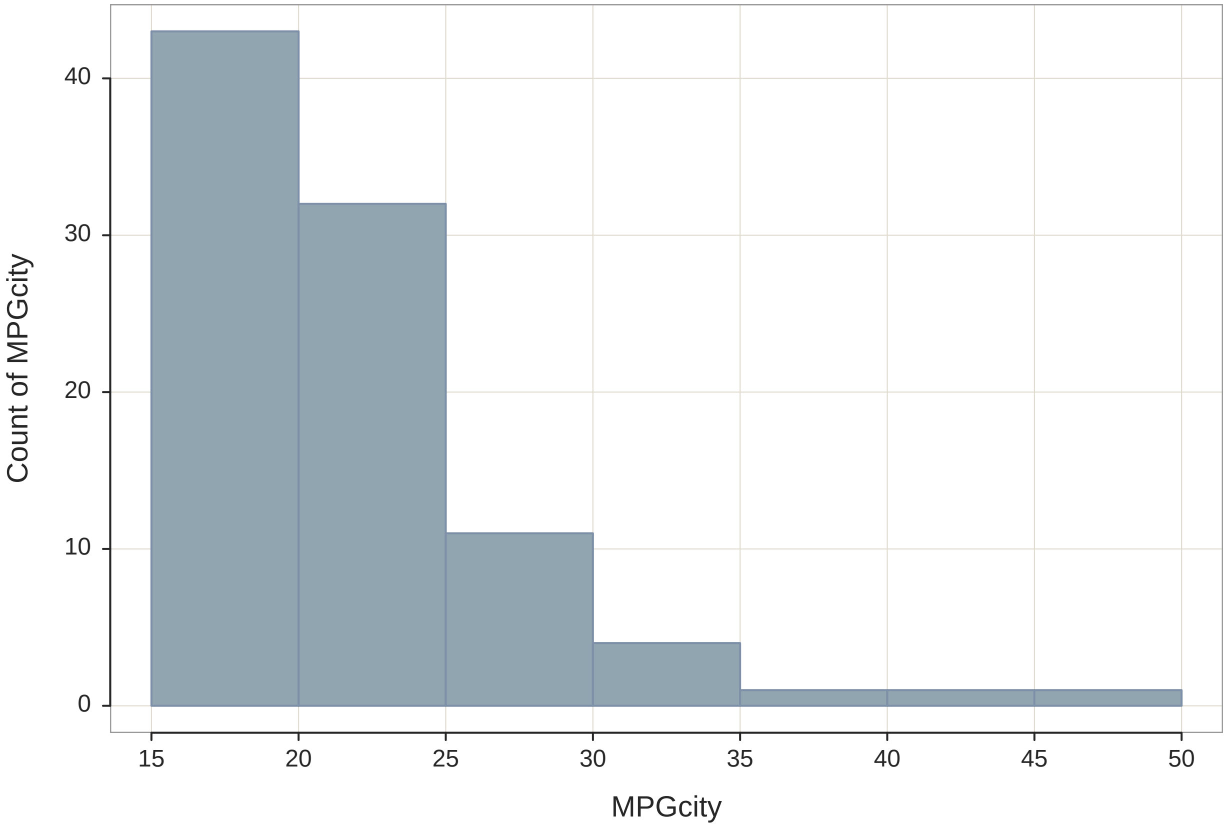
We can see that small cars have the best mileage, while vans have the worst.

2 Histogram

a.

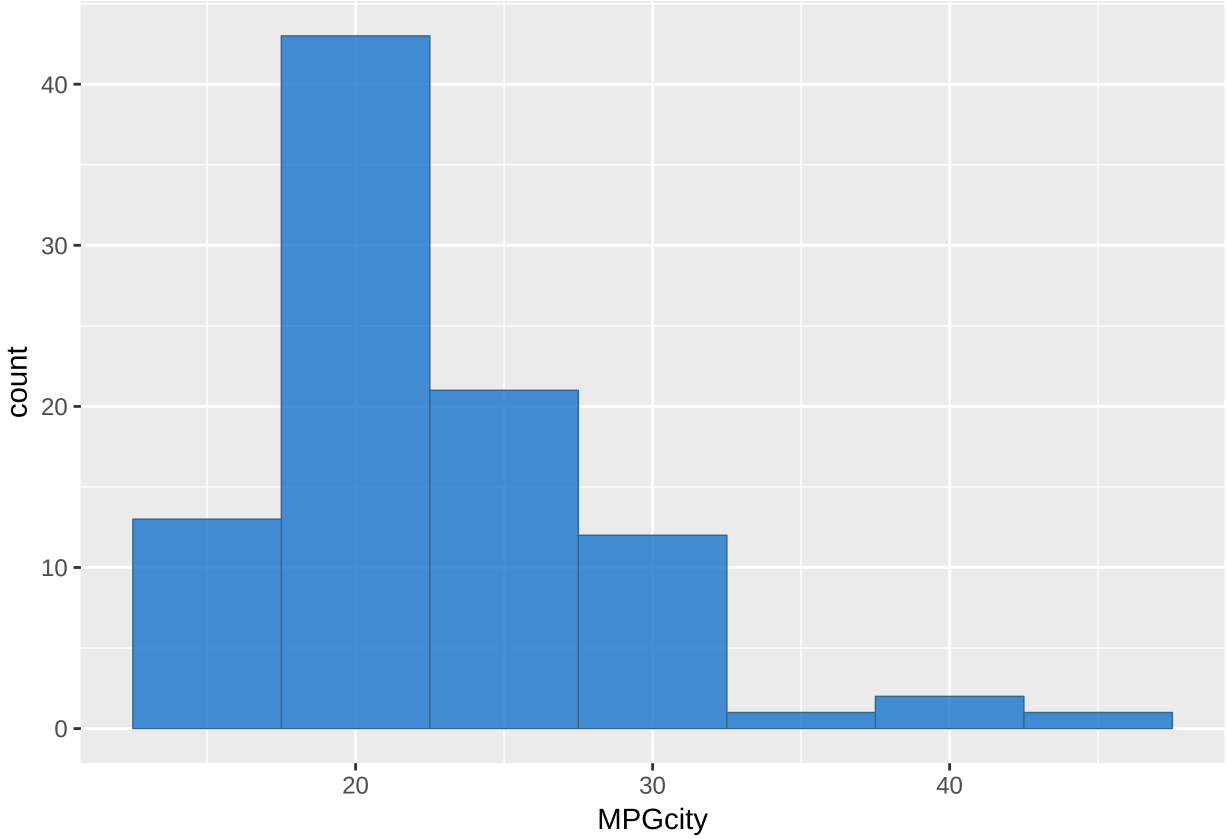
Here's the histogram for city MPG using lessR:

```
Histogram(MPGcity, quiet=TRUE)
```



and ggplot2, using the same bin width:

```
ggplot(d, aes(MPGcity)) +
  geom_histogram(binwidth=5, fill="dodgerblue3", color="steelblue4",
                 alpha=.8, size=.25)
```

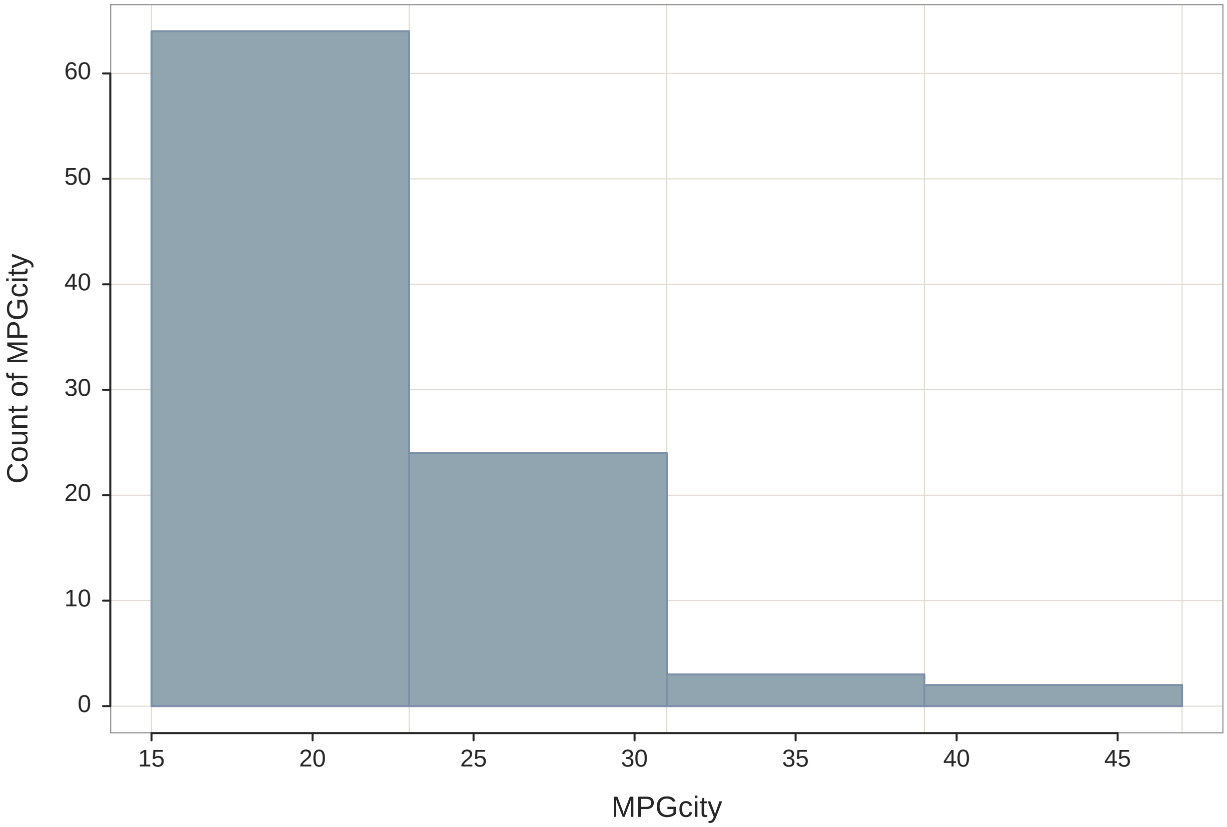


Note the steep dropoff after 25 MPG; this data may be from before hybrids were common.

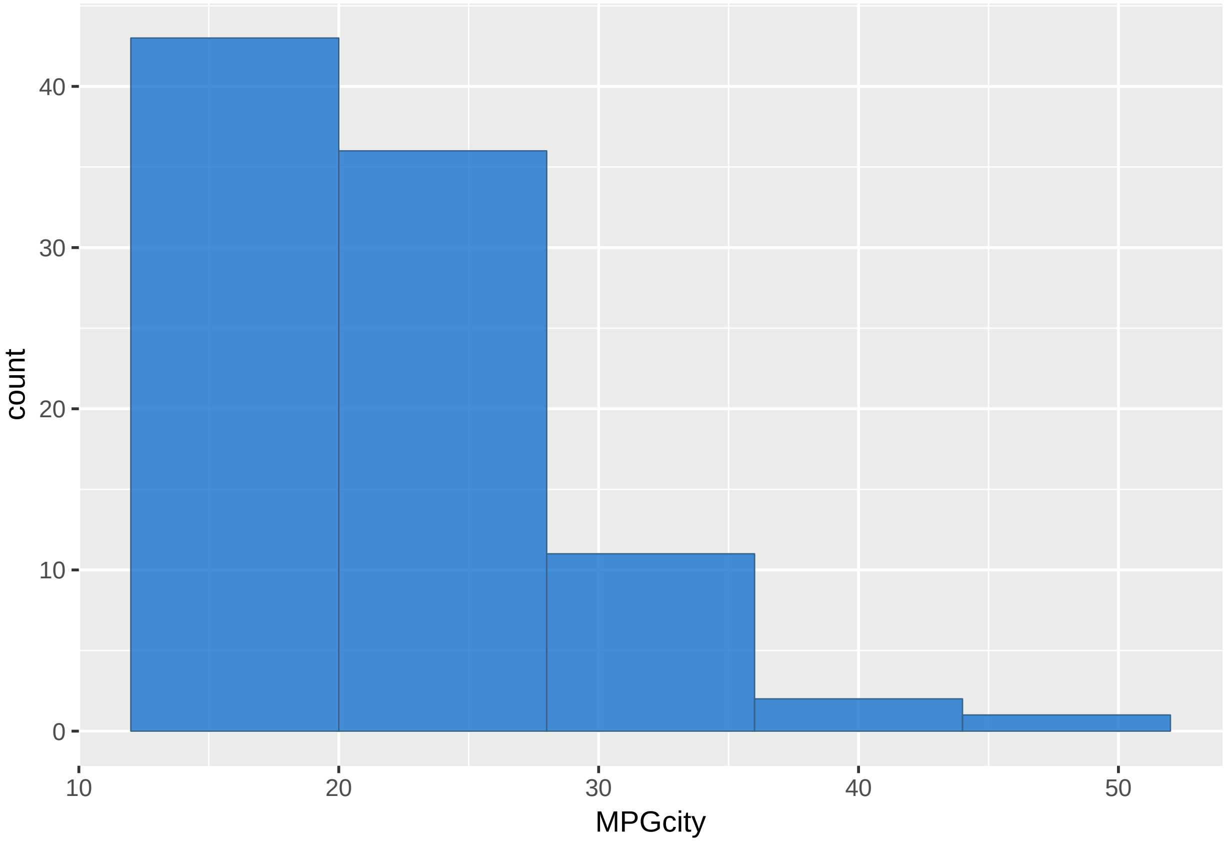
b.

Here are the same two plots with more appropriate bin widths. I've increased the width to 8 to more clearly show the divide between normal and high-mileage vehicles.

```
Histogram(MPGcity, bin.width=8, quiet=TRUE)
```



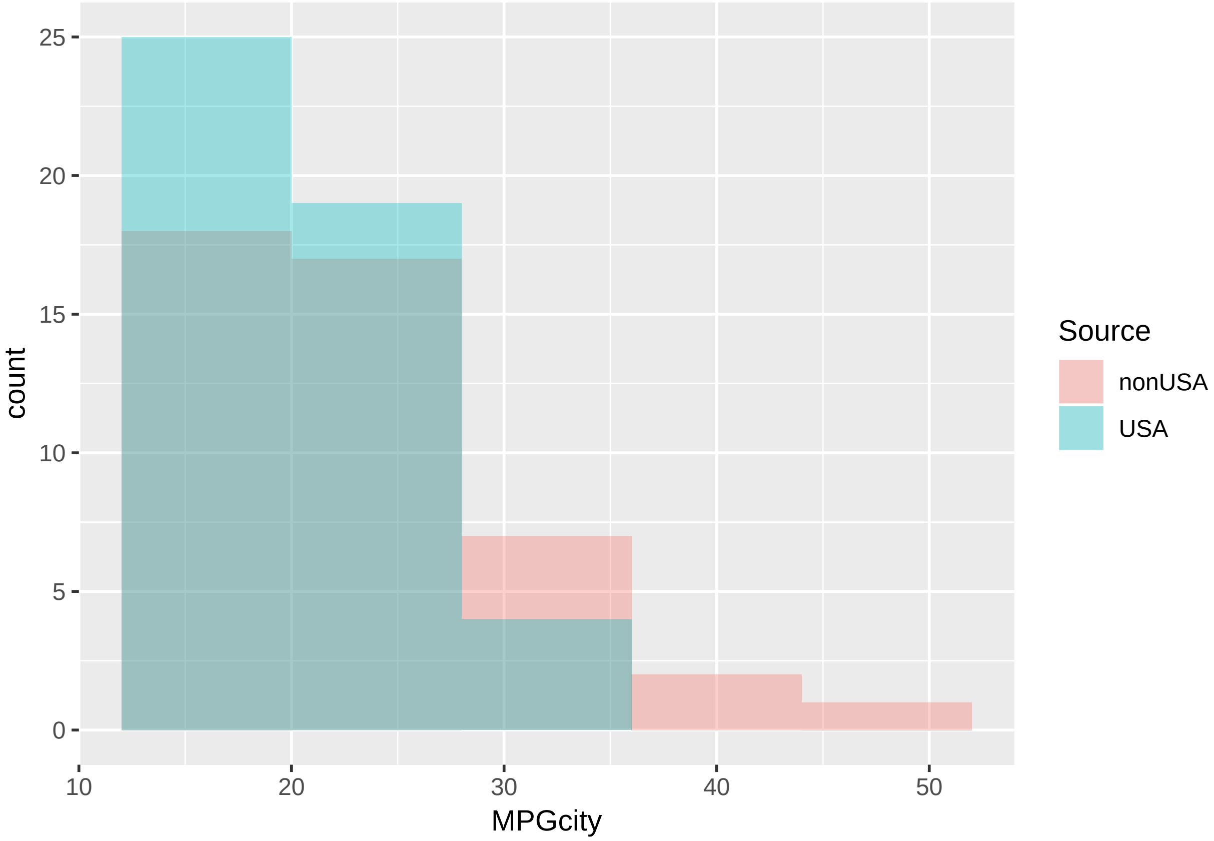
```
ggplot(d, aes(MPGcity)) +
  geom_histogram(binwidth=8, fill="dodgerblue3", color="steelblue4",
                 alpha=.8, size=.25)
```



c.

Here's the ggplot2 overlapping histogram for city MPG by source:

```r
ggplot(d, aes(MPGcity, fill=Source)) +
  geom_histogram(position="identity",binwidth=8,
                 alpha=.35, size=.25)
```



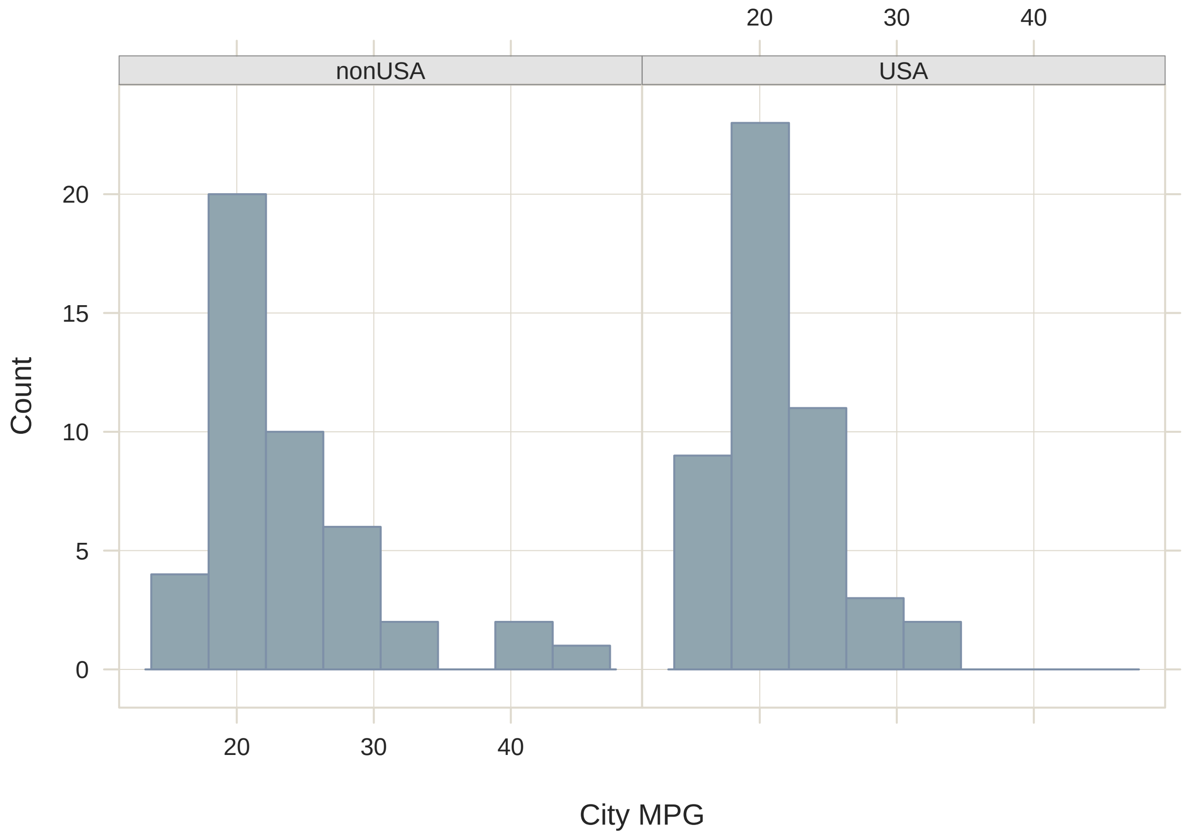
It looks like the non-USA cars in this sample skew toward being more fuel efficient.

d.

Here's the side-by-side histogram for city MPG by source from lessR:

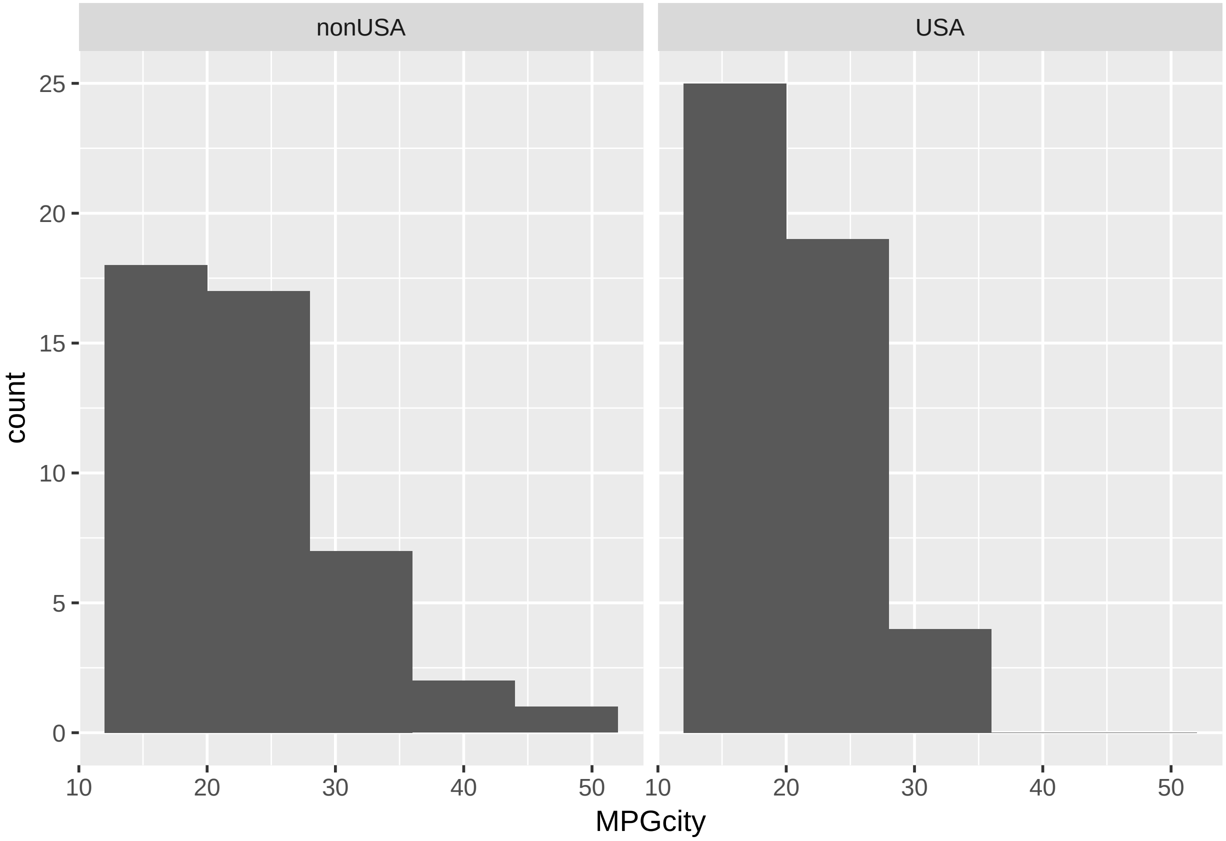
```
hs(MPGcity, by1=Source, quiet=TRUE, ylab="Count", xlab="City MPG")
```

```
## [Trellis graphics from Deepayan Sarkar's lattice package]
```



and ggplot2:

```
ggplot(d, aes(MPGcity)) +
  geom_histogram(binwidth=8) + facet_grid(cols=vars(Source))
```

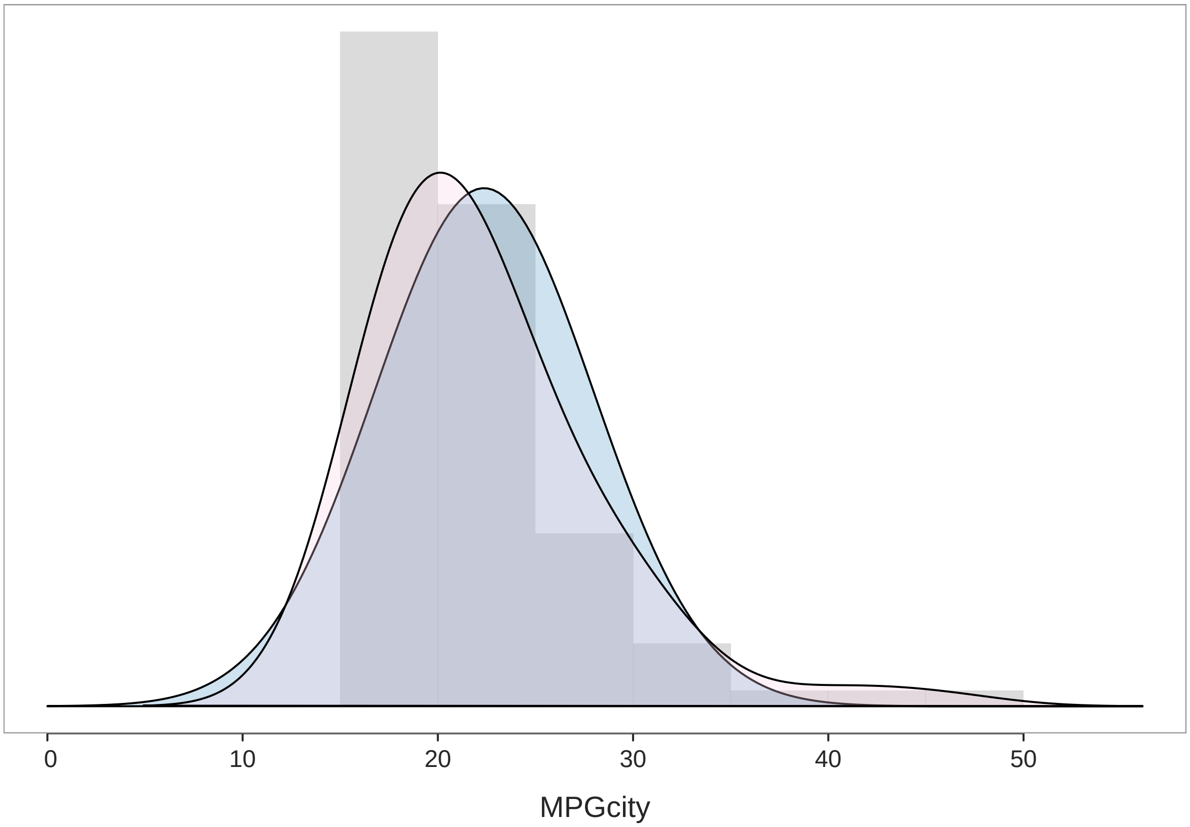


This shows the same comparison as in the overlapping histogram- the non-USA distribution skews more efficient.

e.

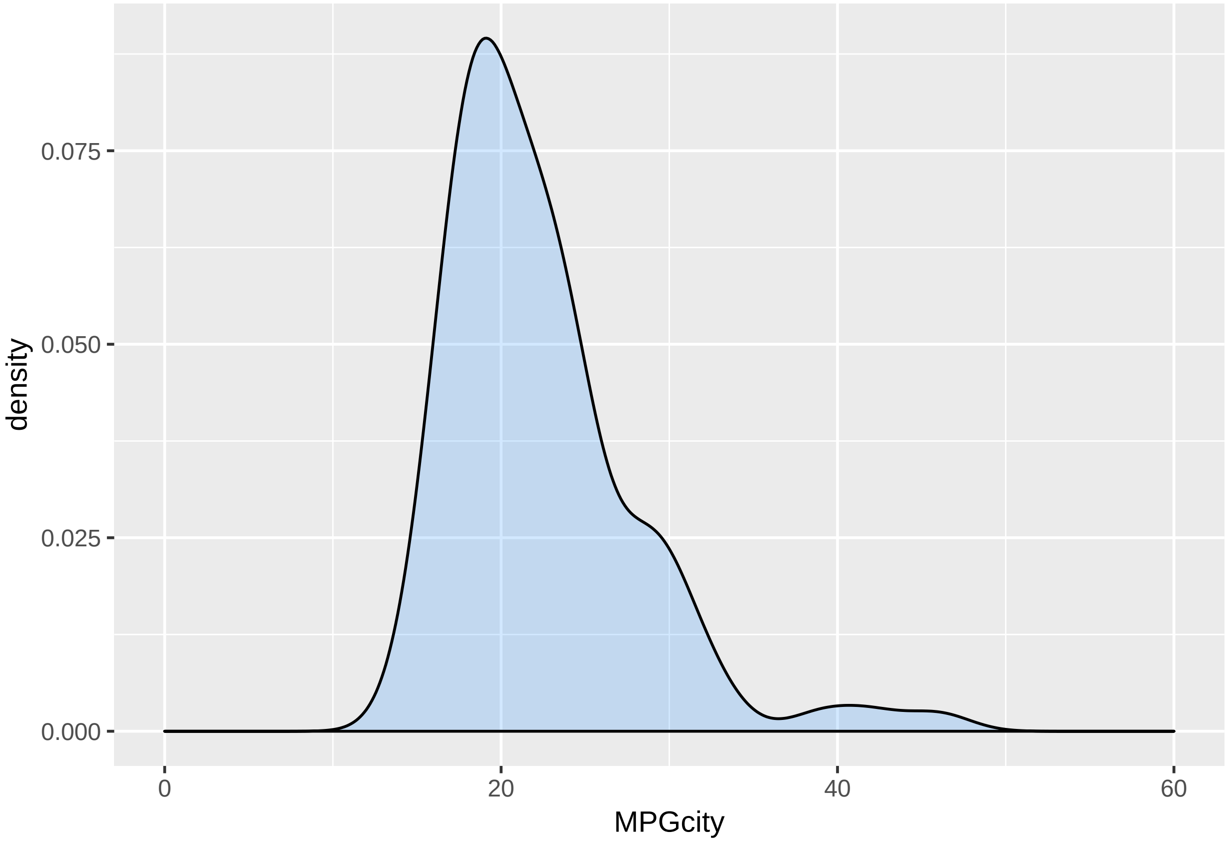
Here's the density curve for city MPG using lessR:

```
Density(MPGcity, x.min=0,quiet=TRUE)
```



and ggplot2:

```
ggplot(d, aes(MPGcity)) + geom_density(alpha=.2, fill="dodgerblue") + xlim(0,60)
```

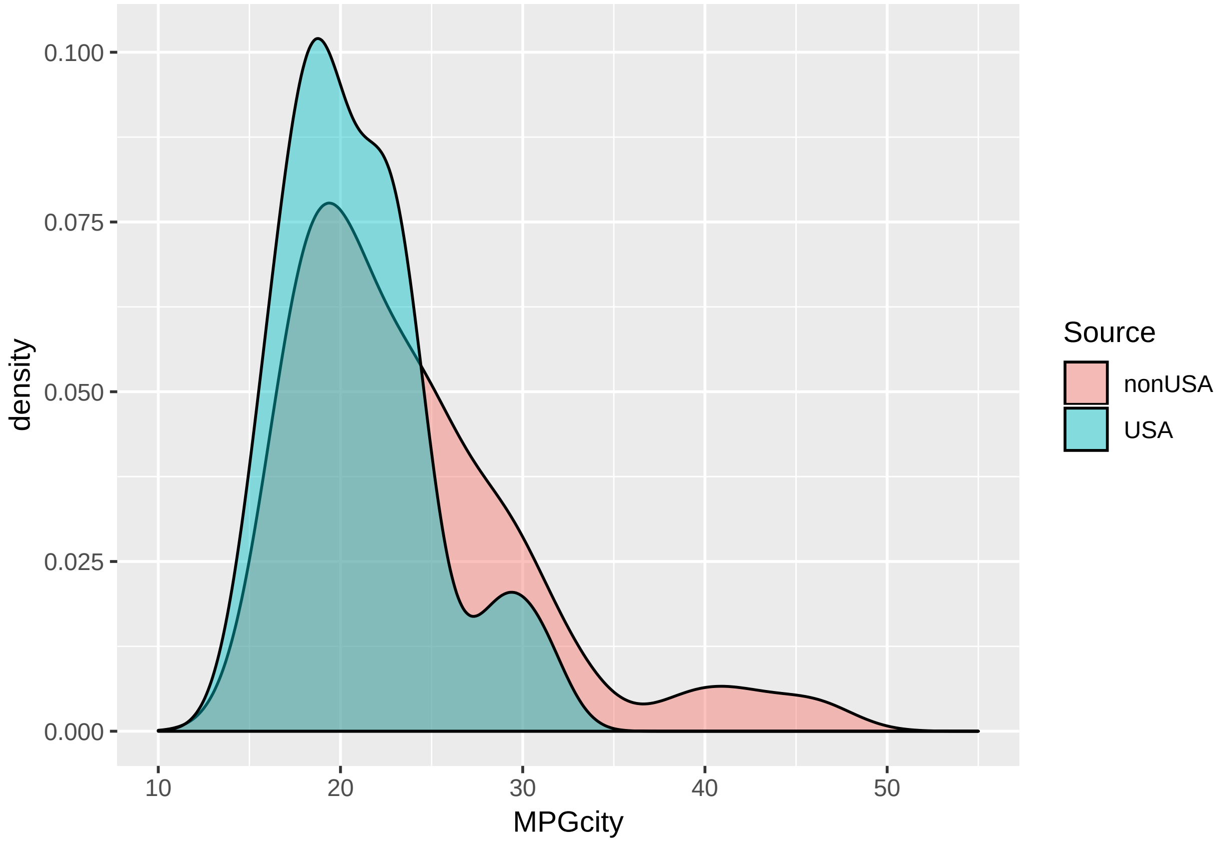


Again we can see the sharp peak in cars that get around 20 MPG in the city.

f.

Here's the overlapping density plot in ggplot2:

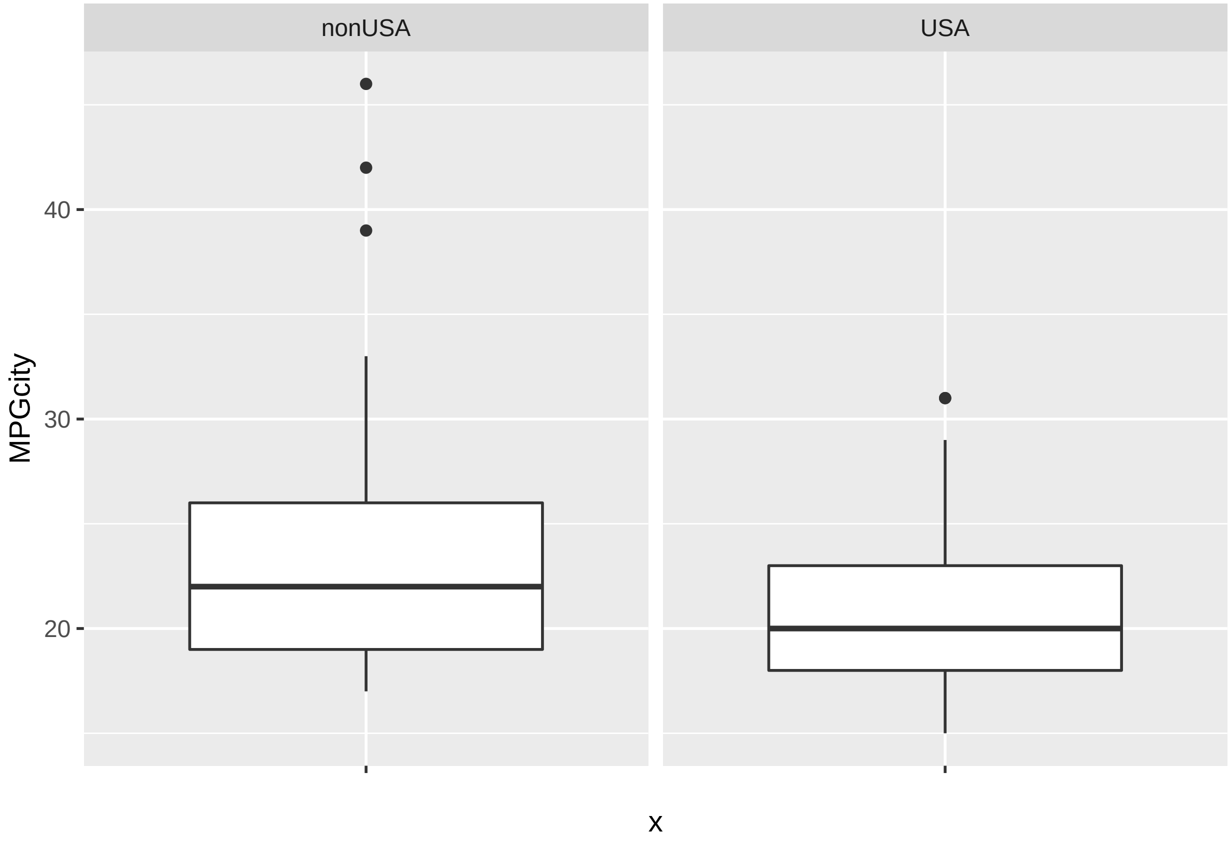
```
ggplot(d, aes(MPGcity, fill=Source)) +
  geom_density(position="identity", alpha=.45)+xlim(10,55)
```



g.

Here are the ggplot2 box plots for city MPG by source:

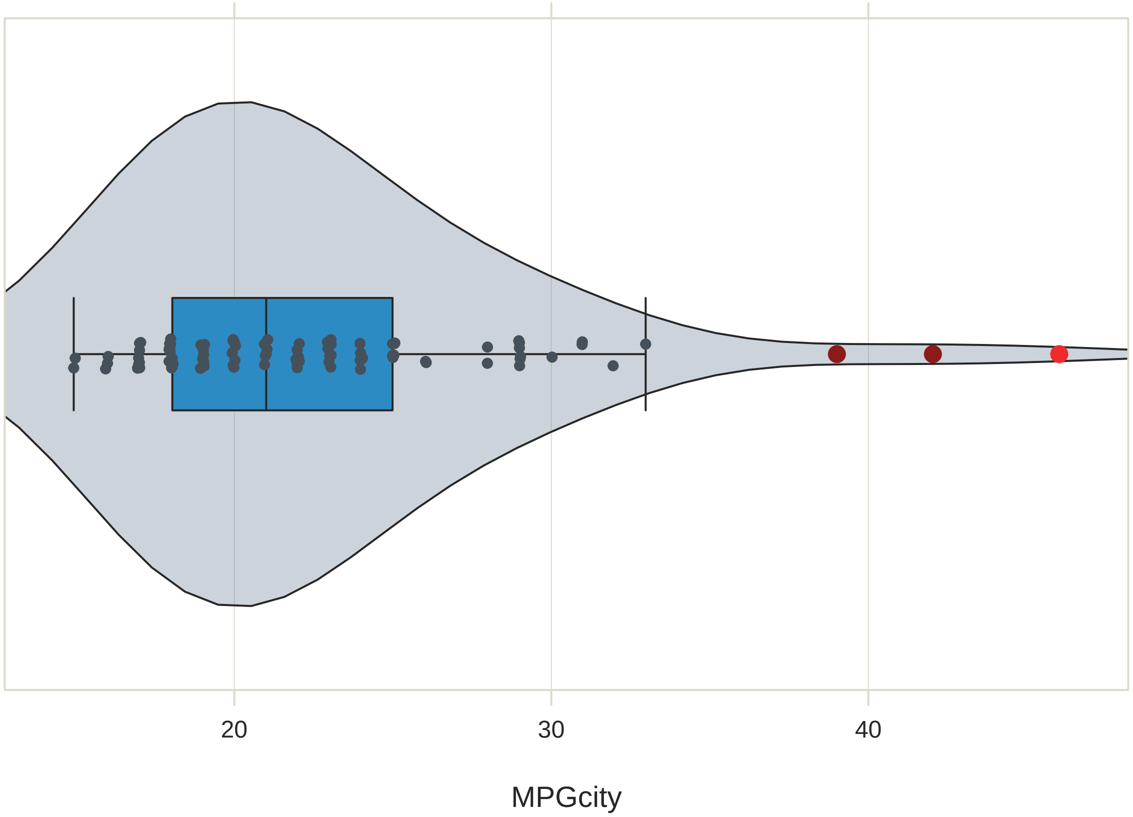
```r
ggplot(d, aes(x="", y=MPGcity)) +
  geom_boxplot() + facet_grid(cols=vars(Source))
```



h.

Here's the integrated VBS for city MPG using lessR:

```
Plot(MPGcity, quiet=TRUE)
```



i.

The full VBS plot presents the same distribution in so many different ways that you can get a lot more information from it- for instance, the identification of the extremity of the 3 outliers on the right, the density of the ponits around the mean, and how neatly most of the distribution fits within the range of the box plot. This level of detail also makes it much busier and more difficult to read- the important thing to get out of looking at this distribution is just that you have 3 outliers on the right and a bunch of points clumped around the mean, which the histogram communicates just as well and much more simply.