

Continuous Data (3 of 4)

Aug 1, 2023. V2.2 --- This chapter is being heavily edited; It is very much Work in Progress

Continuous Data Across Categories: Group-wise Statistics and Data Manipulation

Our second chapter takes us a step further in our exploration of continuous data. Here, we delve into the intersection of continuous data and categorical variables, examining how the former can be split, summarized, and compared across different levels of one or more categorical variables.

We bring to light methods for generating statistics per group and sophisticated data manipulation techniques. This includes processes like grouping, filtering, and summarizing continuous data, contingent on categorical variables. When it comes to visualizations, our focus shifts towards creating juxtaposed box plots, segmented histograms, and density plots that reveal the distribution of continuous data across varied categories.

To achieve this, we exemplify the use of libraries such as `dplyr` and `ggplot2`.

Incorporating categorical variables into our analysis elevates the depth of our comprehension of the data. It allows us to shift from a broad, holistic view to a more concentrated inspection of specific data segments. This approach empowers us to produce a more granular and nuanced interpretation of our data.

6. **Data:** Let us work with the same `mtcars` data from the previous chapter. Suppose we have run the following code:

```
# Load the required libraries, suppressing annoying startup messages
library(tibble)
suppressPackageStartupMessages(library(dplyr))
# Read the mtcars dataset into a tibble called tb
data(mtcars)
tb <- as_tibble(mtcars)
attach(tb)
# Convert several numeric columns into factor variables
```

```

tb$cyl <- as.factor(tb$cyl)
tb$vs <- as.factor(tb$vs)
tb$am <- as.factor(tb$am)
tb$gear <- as.factor(tb$gear)

attach(tb)

```

The following objects are masked from tb (pos = 3):

```
am, carb, cyl, disp, drat, gear, hp, mpg, qsec, vs, wt
```

Bivariate Continuous and Categorical data

1. Bivariate Relationship between Weight (wt) and Transmission (am)
2. Display a summary table showing the descriptive statistics of weight of the cars broken down by transmission (am=1 or am=0)

aggregate()

```

aggregate(tb$wt,
          by = list("am" = tb$am),
          mean)

```

	am	x
1	0	3.768895
2	1	2.411000

```

aggregate(tb$wt,
          by = list("am" = tb$am),
          sd)

```

	am	x
1	0	0.7774001
2	1	0.6169816

tapply()

```
tapply(tb$wt, tb$am, mean)
```

```
      0      1  
3.768895 2.411000
```

```
tapply(tb$wt, tb$am, sd)
```

```
      0      1  
0.7774001 0.6169816
```

Visualizing Means – mean plot showing the average weight of the cars, broken down by transmission (am=1 & am=0)

```
library(gplots)
```

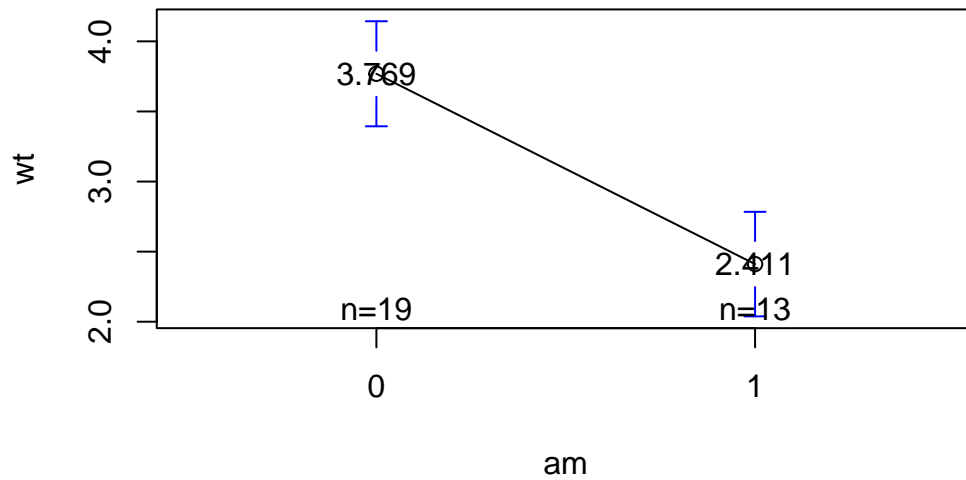
Attaching package: 'gplots'

The following object is masked from 'package:stats':

```
lowess
```

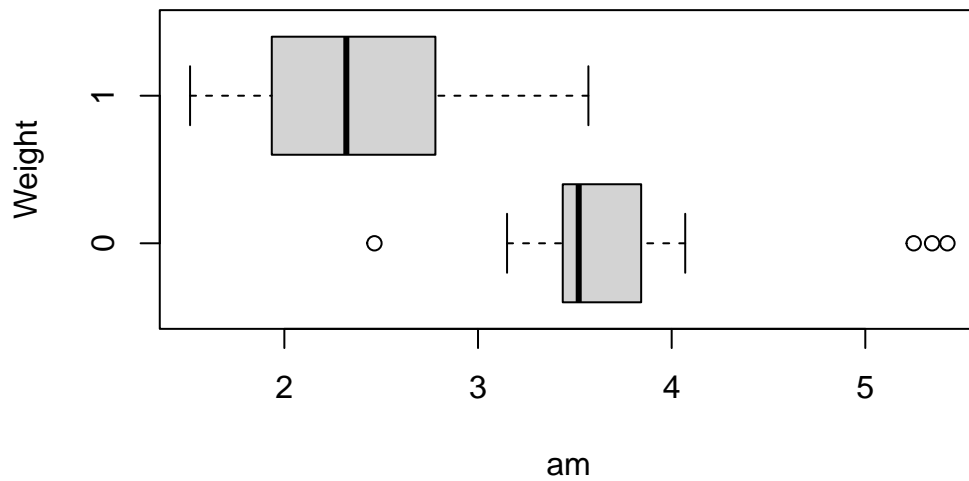
```
plotmeans(wt ~ am  
          ,data = mtcars  
          ,mean.labels = TRUE  
          ,digits=3  
          ,main = "Mean (wt) by am = {0,1} "  
          )
```

Mean (wt) by am = {0,1}



Visualizing Median using Box Plot – median weight of the cars broken down by transmission (am=1 & am=0)

```
boxplot(wt~am
, xlab = "am"
, ylab = "Weight"
, horizontal = TRUE
)
```



Bivariate Relationship between Weight (wt) and Cylinders (cyl)

Display a summary table showing the mean weight of the cars broken down by cylinders (cyl=4,6,8)

```
psych::describeBy(wt, cyl)
```

Descriptive statistics by group

group: 4

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	11	2.29	0.57	2.2	2.27	0.54	1.51	3.19	1.68	0.3	-1.36	0.17

group: 6

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	7	3.12	0.36	3.21	3.12	0.36	2.62	3.46	0.84	-0.22	-1.98	0.13

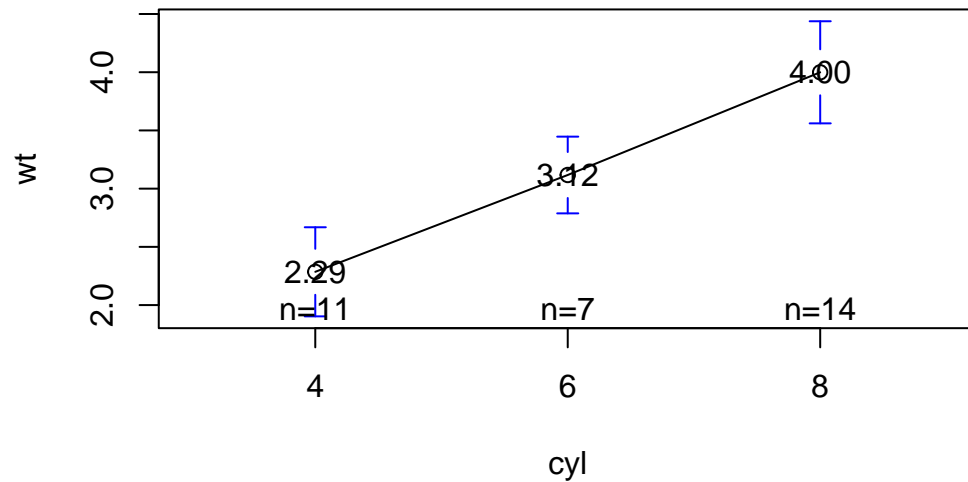
group: 8

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	14	4	0.76	3.76	3.95	0.41	3.17	5.42	2.25	0.99	-0.71	0.2

Show a mean plot showing the mean weight of the cars broken down by cylinders (cyl=4,6,8)

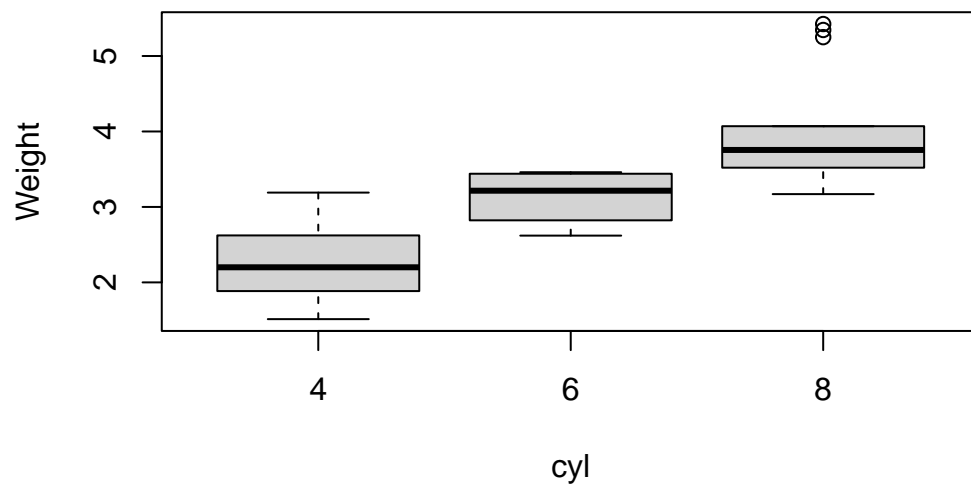
```
library(gplots)
plotmeans(wt ~ cyl,
           data = mtcars,
           , mean.labels = TRUE
           , digits=2
           , main = "Mean (wt) by cyl = {4,6,8} ")
```

Mean (wt) by cyl = {4,6,8}



Show a box plot showing the median weight of the cars broken down by cylinders (cyl=4,6,8)

```
boxplot(wt ~ cyl,  
        xlab = "cyl", ylab = "Weight"  
        )
```



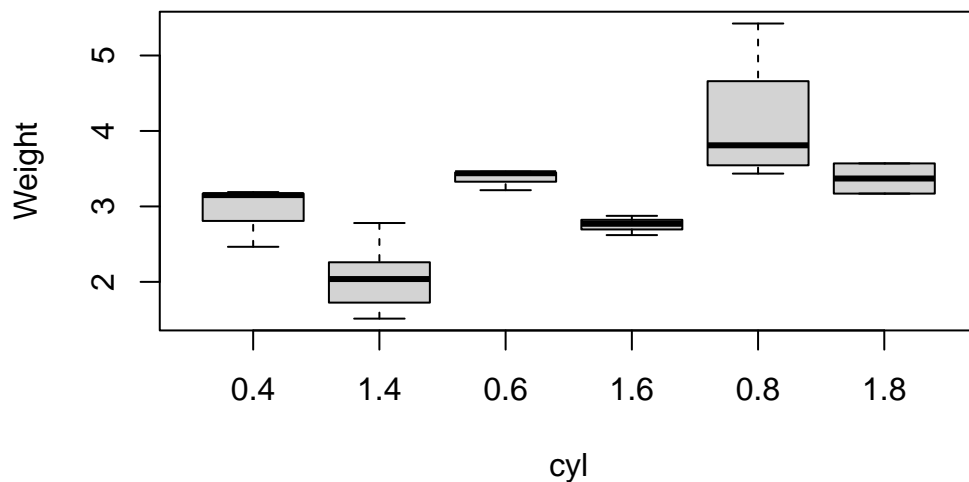
Distribution of Weight (wt) by Cylinders (cyl = {4,6,8}) and Transmission Type (am = {0,1})

```
aggregate(wt,  
          by = list("am" =am, "cyl" = cyl),  
          mean)
```

	am	cyl	x
1	0	4	2.935000
2	1	4	2.042250
3	0	6	3.388750
4	1	6	2.755000
5	0	8	4.104083
6	1	8	3.370000

Visualization - Show a box plot showing the mean weight of the cars broken down by Transmission Type (am=1 & am=0) & cylinders (cyl=4,6,8)

```
boxplot(wt ~ am:cyl  
        , xlab = "cyl"  
        , ylab = "Weight"  
        )
```



Visualization - Show a mean plot showing the mean weight of the cars broken down by Transmission Type (am=1 & am=0) & cylinders (cyl=4,6,8)

```
library(gplots)
plotmeans(wt ~ interaction(am, cyl, sep = ", "))
  , data = mtcars
  , mean.labels = TRUE
  , digits=2
  , connect = FALSE
  , main = "Mean (wt) by cyl = {4,6,8} & am = {0,1}"
  , xlab= "cyl = {4,6,8} & am = {0,1}"
  , ylab="Average Weight"
  )
```

Mean (wt) by cyl = {4,6,8} & am = {0,1}

