

## Section 3: Plots for categorical variable

---

Review of important functions covered in this section:

### Functions

pie	Draws pie chart
barplot	Draws a barplot
dotchart	Draws a Cleveland's Dot Plots

### Exercise

We'll work with `warpbreaks` dataset (The Number of Breaks in Yarn during Weaving).

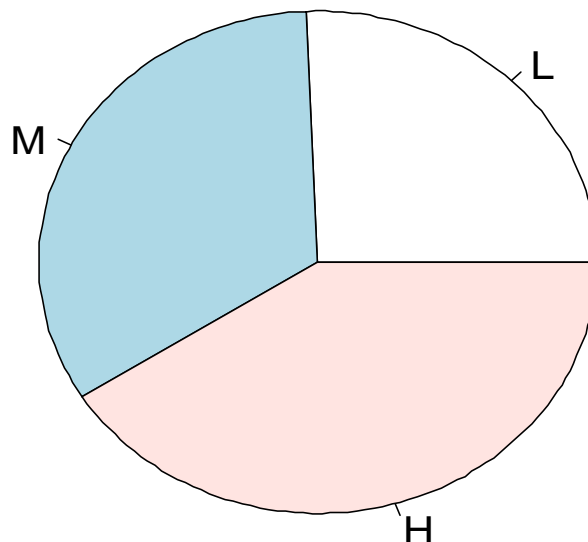
- Make a table of tension variable in the `warpbreaks` dataset, when the number of breaks are at most 30. Save it as R object `tab`.

```
tab = table(subset(warpbreaks, breaks <= 30)$tension)
```

- Use this table to draw a pie chart.

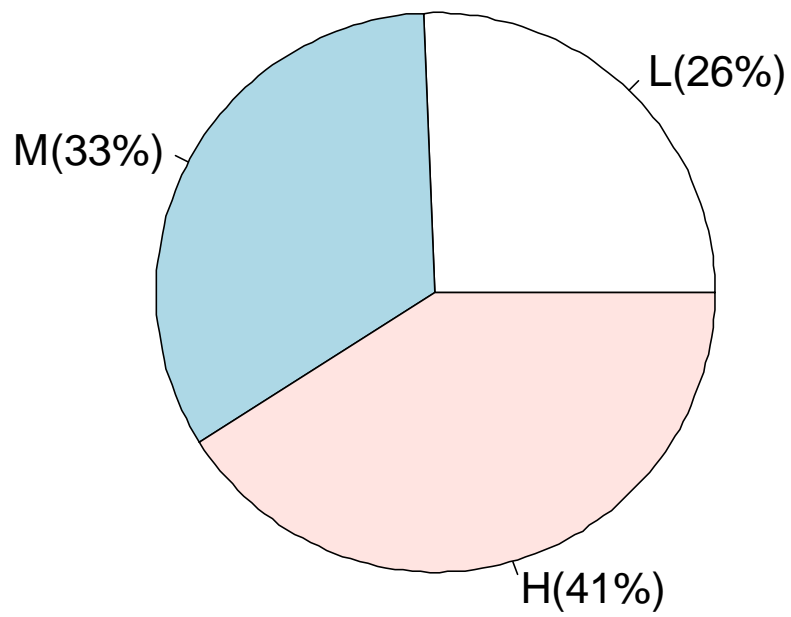
```
pie(tab, main="Pie chart")
```

### Pie chart



c. Draw a pie chart with the above data where percentage of every category will be mentioned.

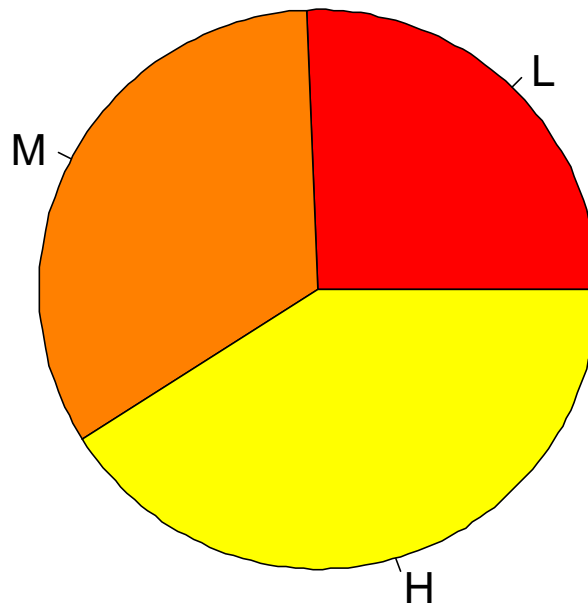
```
p = round(tab/sum(tab)*100)
pie(tab, labels = paste(names(tab), "(",p, "%)", sep = ""))
```



d. Use heat color palettes in the previous plot. (Hint: ? heat.colors).

```
pie(tab, main="Pie chart", col = heat.colors(3))
```

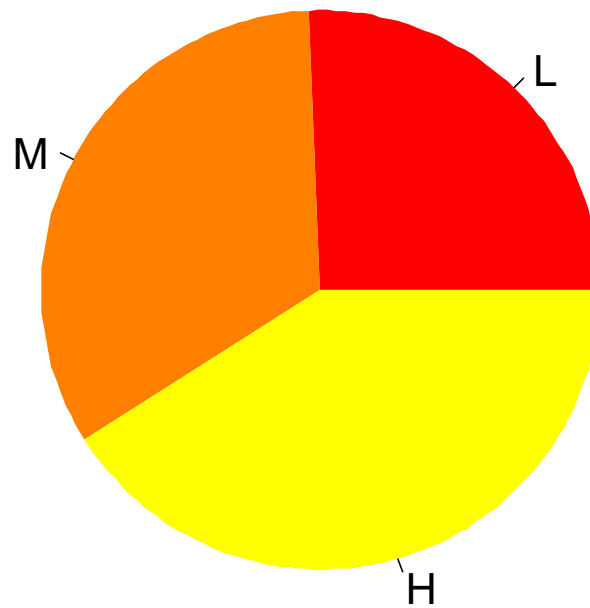
## Pie chart



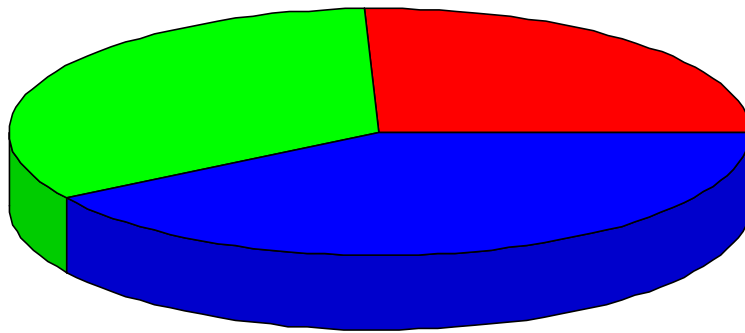
e. Get rid of the border in the above plot.

```
pie(tab, main="Pie chart", col = heat.colors(3), border = NA)
```

## Pie chart



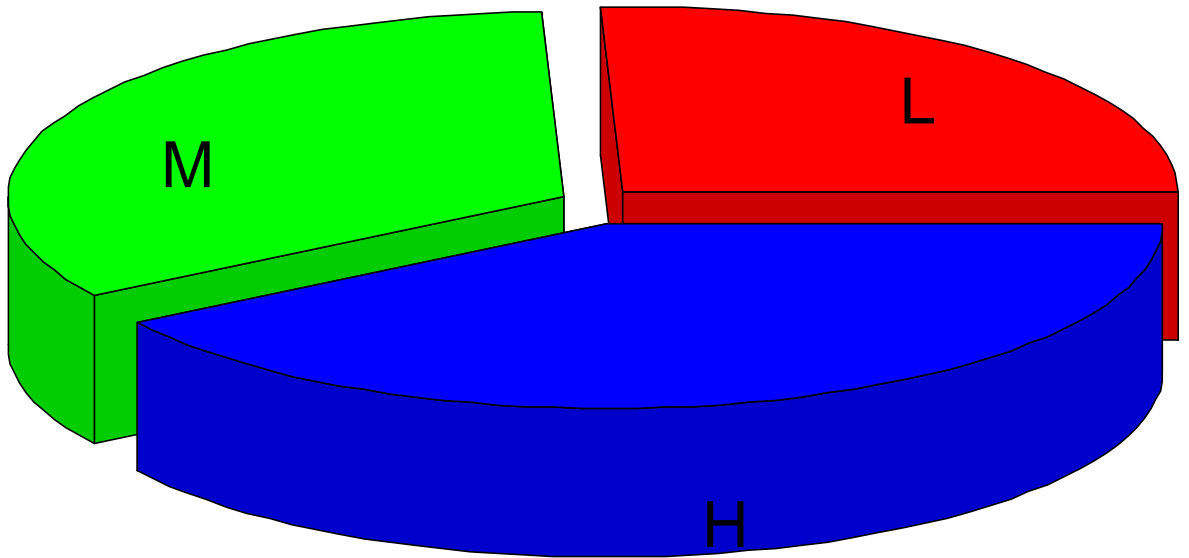
- f. Install and load the plotrix package. Hint: use `?install.packages` and `?load`
- ```
install.packages("plotrix")  
require(plotrix)
```
- g. Use `pie3D` function from the plotrix to draw the data used previously.
- ```
pie3D(tab)
```



- h. Use radius, explode, height, levels arguments of the plot3D function to customize the plot.  
(Hint: ?pie3D)

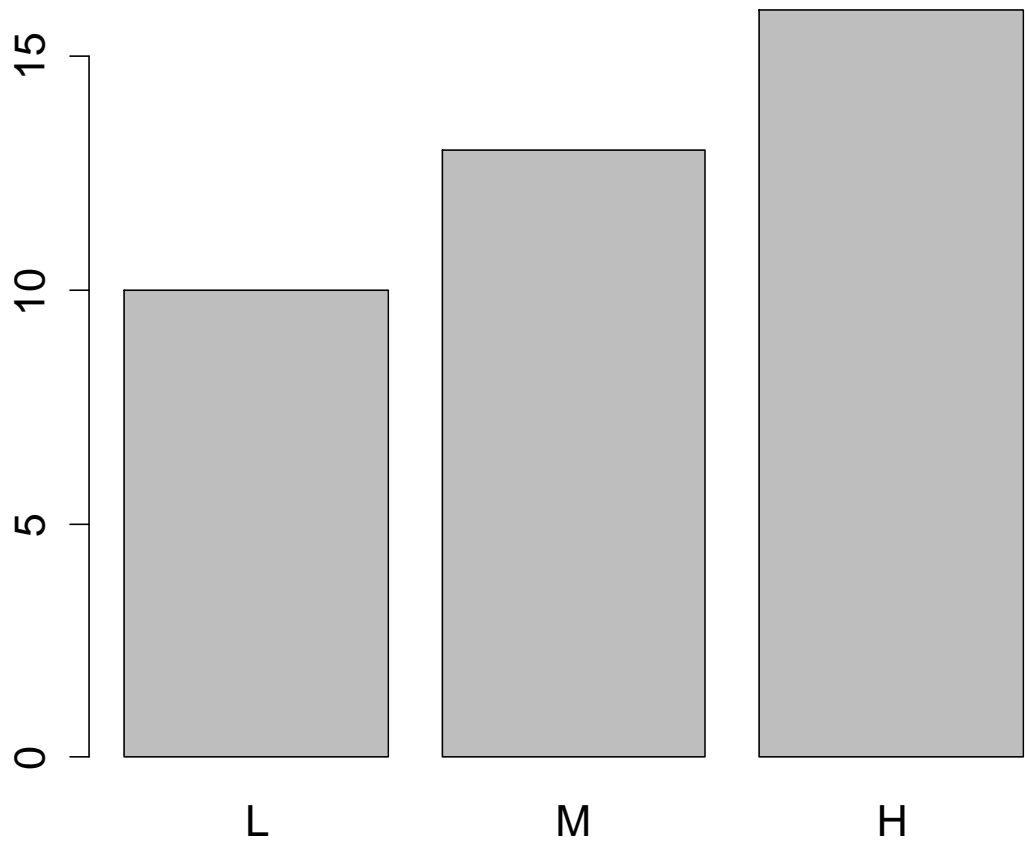
```
pie3D(tab, radius=1.5,explode=0.1,height = .2, main="3D Pie chart",  
labels = levels(warpbreaks$tension))
```

## 3D Pie chart



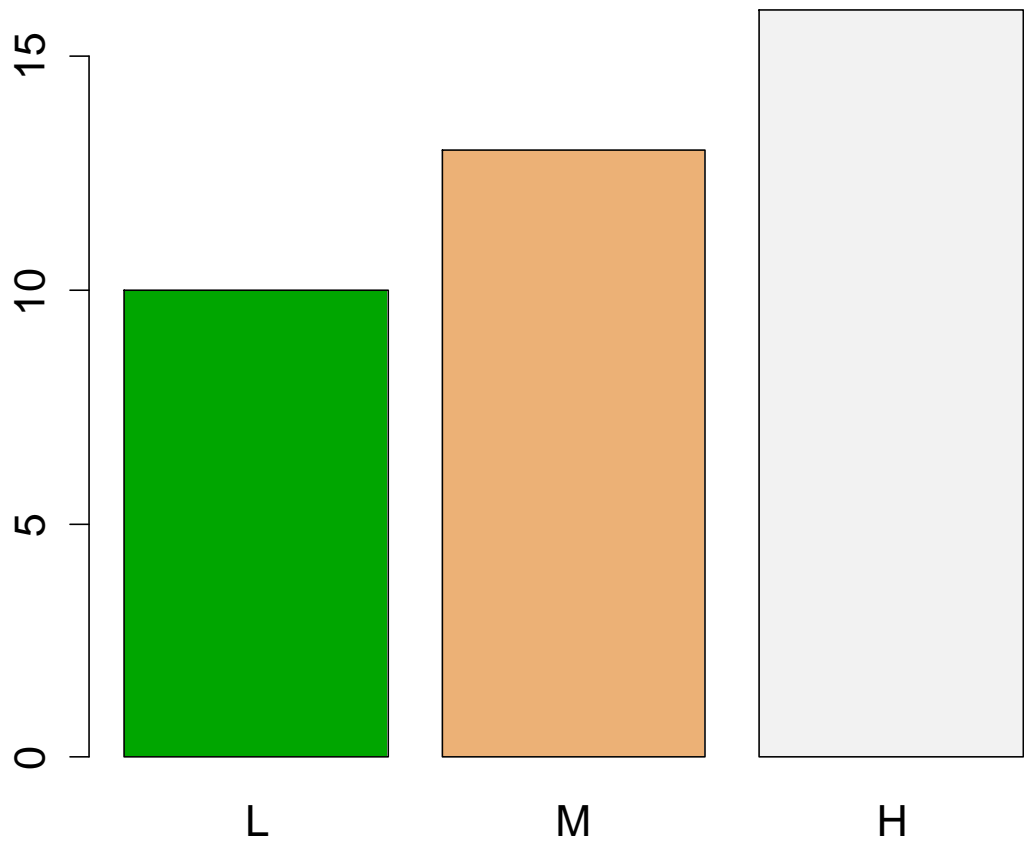
- i. Draw a barplot of the above data.

```
barplot(tab)
```



j. Change the color of the above barplot to terrain color palettes.

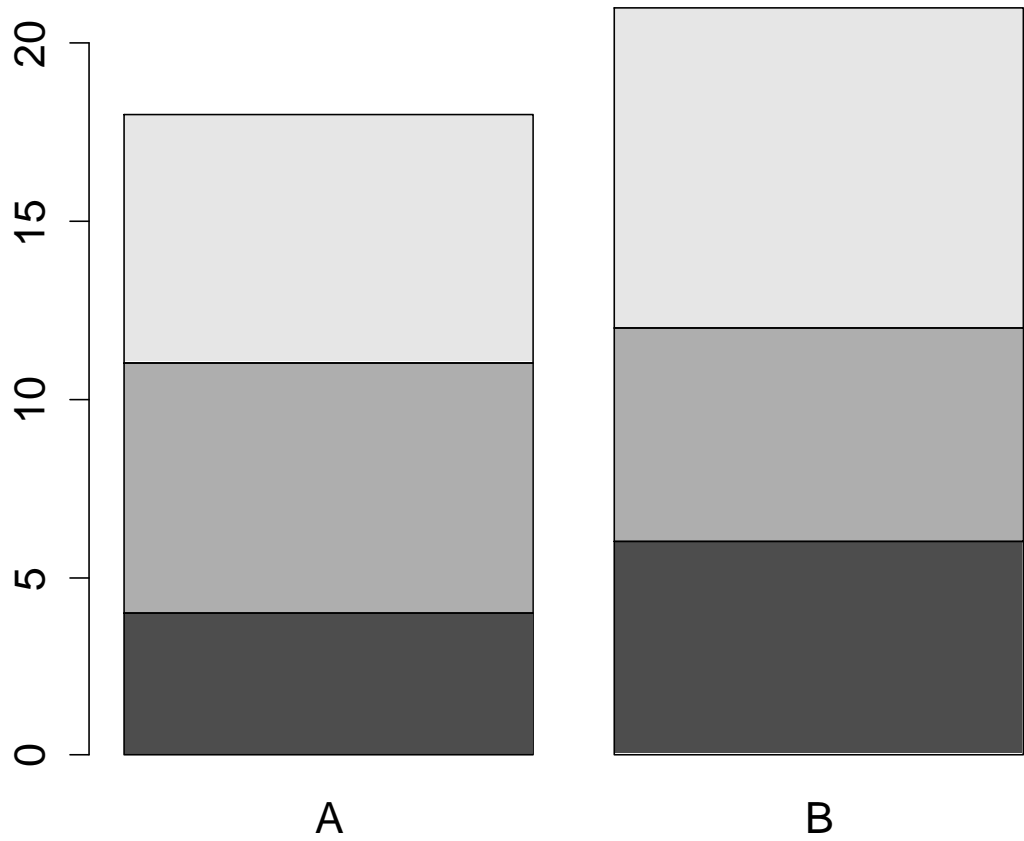
```
barplot(tab, col = terrain.colors(3))
```





- k. Cross tabulate tension and wool variable of the warpbreaks dataset when the number of breaks are at most 30 and plot the output using barplot.

```
bles30 = subset(warpbreaks, breaks <= 30)  
barplot(table(bles30$tension, bles30$wool))
```



I. Draw a dot chart of the `tab` data and specify the labels.

```
dotchart(as.numeric(tab), labels=names(table(warpbreaks$tension)))
```

