

Annotations Example

Moriah

9/26/2019

The dataframe **sleep** from the package **datasets** contains 20 observations on 3 variables and represent the results of an experiment in which two groups of students were given different treatments and the change in the hours of sleep that they got was recorded.

* **extra**: change in hours of sleep (type = numeric)

* **group**: drug given (type = factor)

* **ID**: patient ID (type = factor)

```
head(sleep)  #preview data
```

```
##   extra group ID
## 1   0.7     1  1
## 2  -1.6     1  2
## 3  -0.2     1  3
## 4  -1.2     1  4
## 5  -0.1     1  5
## 6   3.4     1  6
```

```
#change column name
```

```
#in order to prevent confusion when using
```

```
#group_by() in piping
```

```
colnames(sleep)[colnames(sleep)=="group"] <- "treatment"  #rename column
```

```
head(sleep)  #see that column name is changed
```

```
##   extra treatment ID
## 1   0.7           1  1
## 2  -1.6           1  2
## 3  -0.2           1  3
## 4  -1.2           1  4
## 5  -0.1           1  5
## 6   3.4           1  6
```

Let's take a look at the average effects of each treatment.

```
##find mean change in sleep for each treatment##
```

```
averages <- sleep %>%  #store results to object named 'averages'
```

```
  group_by(treatment) %>%  #group observations by treatment
```

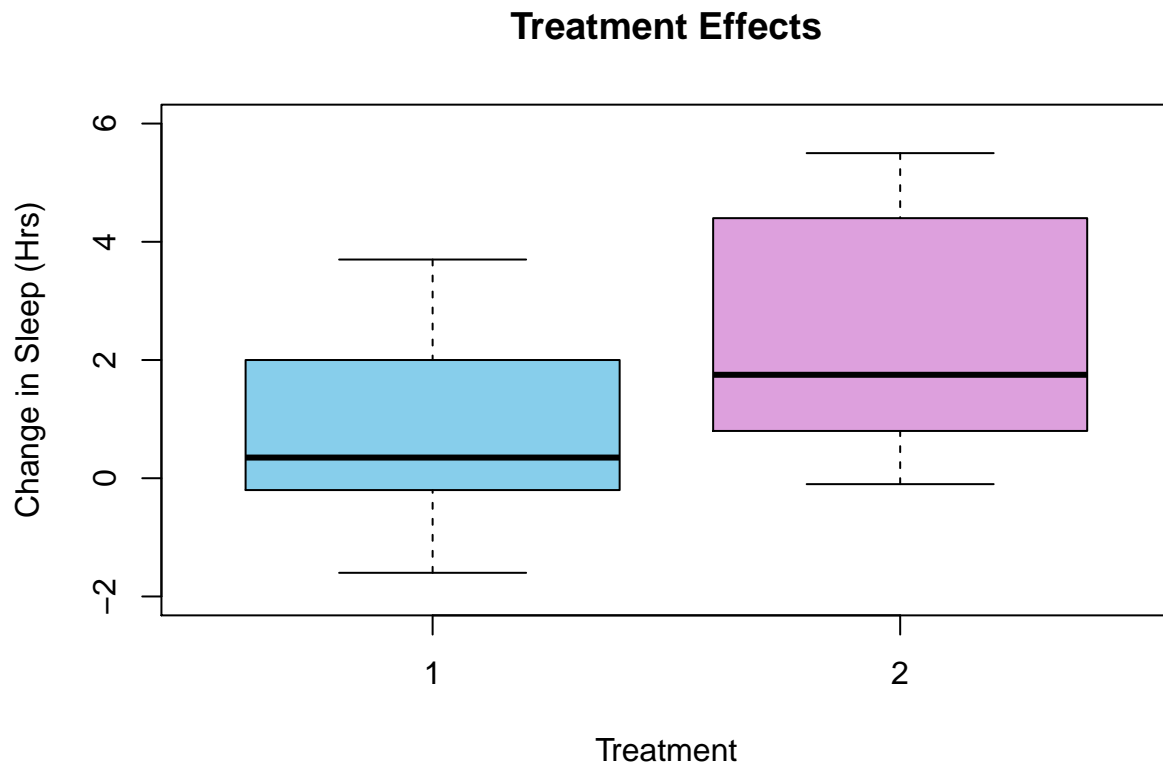
```
  summarise(avg = mean(extra))  #calculate mean change in hours of sleep
```

```
averages  #see results
```

```
## # A tibble: 2 x 2
##   treatment    avg
##   <fct>      <dbl>
## 1 1          0.75
## 2 2          2.33
```

Now let's compare the boxplots of the effects of each treatment.

```
##create a boxplot##  
boxplot(extra ~ treatment, data = sleep,  
         xlab = "Treatment",      #label x-axis  
         ylab = "Change in Sleep (Hrs)", #label y-axis  
         main = "Treatment Effects", #title boxplots  
         names = c("1", "2"),    #label treatment types  
         col = c("skyblue", "plum"), #choose color for each boxplots  
         ylim = c(-2,6)         #increase range of y-axis  
         )
```



Finally, let's create a simple function called `eval_effect` that returns whether a value represents an increase, decrease, or no change. Then, let's apply that function to the column 'extra' in the sleep dataset and create a new column called 'effect'.

```
##create function eval_effect##
#input: value representing observed change
#output: statement about type of change
eval_effect <- function(observation){
  if (observation > 0){      #label a positive value as an increase
    return("increase")}
  else if (observation < 0){  #label a negative value as a decrease
    return("decrease")}
  else {                     #label a zero value as no change
    return("no change")}
}

##effect column##
#create column
sleep$effect <- apply(sleep[, 1], 1, eval_effect)    #apply eval_effect function to observations
#view data with new column
sleep
```

##	extra	treatment	ID	effect
## 1	0.7	1	1	increase
## 2	-1.6	1	2	decrease
## 3	-0.2	1	3	decrease
## 4	-1.2	1	4	decrease
## 5	-0.1	1	5	decrease
## 6	3.4	1	6	increase
## 7	3.7	1	7	increase
## 8	0.8	1	8	increase
## 9	0.0	1	9	no change
## 10	2.0	1	10	increase
## 11	1.9	2	1	increase
## 12	0.8	2	2	increase
## 13	1.1	2	3	increase
## 14	0.1	2	4	increase
## 15	-0.1	2	5	decrease
## 16	4.4	2	6	increase
## 17	5.5	2	7	increase
## 18	1.6	2	8	increase
## 19	4.6	2	9	increase
## 20	3.4	2	10	increase