Bellabeat Notebook

Below is an explanation and my code for my Google Data Analytics Certification Course Capstone project.

I chose to do my project for a fictional organization called Bellabeat, which produces fitness smart devices. My objective with this project is to analyze Bellabeat smart usage data in order to gain insight into how people are using their smart devices, and make marketing recommendations based on the data.

I ended up using the data to compare the exercise practices of Bellabeat users to nonusers, and found that based on the data provided by Bellabeat and compared to non-users, Bellabeat users tended to exercise more than nonusers and that Bellabeat users are 29% more likely to be meeting exercise goals to combat heart Disease (per CDC guidelines) than nonusers. Below is the process I used to come to that conclusion, and to ultimately visualize the data.

The associated presentation can be found here.

Step 1: Install and load the necessary packages for this analysis. ggplot2 will be used for visualizations later and dplyr for the analysis process.

```
install.packages("tidyverse", repos="https://cloud.r-project.org/")
```

```
## Installing package into 'C:/Users/cmatt/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\cmatt\AppData\Local\Temp\RtmpQrUbCv\downloaded_packages
```

```
install.packages("ggplot2", repos="https://cloud.r-project.org/")
```

```
## Installing package into 'C:/Users/cmatt/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'ggplot2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\cmatt\AppData\Local\Temp\RtmpQrUbCv\downloaded_packages
```

```
install.packages("dplyr", repos="https://cloud.r-project.org/")
```

```
## Installing package into 'C:/Users/cmatt/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)
```

```
## package 'dplyr' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'dplyr'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\cmatt\AppData\Local\R\win-library\4.2\00LOCK\dplyr\libs\x64\dplyr.dll
## to C:\Users\cmatt\AppData\Local\R\win-library\4.2\dplyr\libs\x64\dplyr.dll:
## Permission denied
```

```
## Warning: restored 'dplyr'
```

```
##
## The downloaded binary packages are in
##  C:\Users\cmatt\AppData\Local\Temp\RtmpQrUbCv\downloaded_packages
```

```
library("tidyverse")
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library("ggplot2")

library("dplyr")
```

Step 2: Pull in the necessary CSV for analysis.

```
dailyIntensities <- read.csv("csvs/dailyintensities_merged.csv")
```

Step 3: Clean & Filter the data for analysis.

Gather data for each ID from dailyintensities: count the number of FairlyActiveMinutes and VeryActiveMinutes per day as countIntensitiesMinutes.

```
countIntensitiesMinutes <- dailyIntensities %>%
  select(Id, ActivityDay, FairlyActiveMinutes, VeryActiveMinutes)
```

And then add together all the count intensity minutes per individual person, or Id.

```
totalIntensitiesMinutes <- countIntensitiesMinutes %>%
  group_by(Id) %>%
  summarise(totalActive = sum(FairlyActiveMinutes + VeryActiveMinutes, na.rm = TRUE))
```

Step 4: Analyze the data.

Context for further analysis: There are roughly 31 days worth of data in this dataset. The CDC recommends 2 hrs 30 minutes of moderate intensity exercise per week to combat heart disease. Since the dataset doesn't go week-to week I needed to do some calculations to get an idea of what this works out to per month. To get a rough idea of this, I divided 150 minutes by 7 to get 21.4 minutes of moderate to intense exercise needed per day. This was then multiplied by 31 days, giving me 664.3 minutes needed per month to meet the CDC's monthly minimum requirements to combat heart disease.

This calc is to declare 664.3 as a data frame to use in the next steps of analysis. cdcRecommendedMonthlyActiveMinutes.

```
cdcRecommendedMonthlyActiveMinutes <- 664.3
```

The below calculation will filter the number of individual people who get greater than or equal to 664.3 minutes of fair to active minutes of activity. Assign name to the filtered list as metCdcGuidance.

```
metCdcGuidance <- totalIntensitiesMinutes %>%
  filter(totalActive >= cdcRecommendedMonthlyActiveMinutes)
```

Context for further analysis: 17 out of 33 individuals met or exceeded the CDC's recommendations, which ends up being roughly 52% of Bellabeat's users. According to a Time magazine article only 23% of Americans

get the CDC's recommended amount of exercise. Data Source.

Create data frame for the number of americans that get the CDC's recommended amount of exercise. Presented as a percent (23%).

```
nonuserCountMetPercentCdc <- 23
```

Get the number of individual IDs who met the CDC guidance - This will be used later to calculate a percent.

```
bellabeatCountMetCdc <- nrow(metCdcGuidance)
```

Count total Bellabeat users - this will be used later to calculate a percent.

```
bellabeatAllCount <- nrow(totalIntensitiesMinutes)
```

Calculate a percent of Bellabeatusers who met CDC guidance by comparing it to all unique Bellabeat users in dataset.

```
bellabeatCountMetPercentCdc <- round((bellabeatCountMetCdc / bellabeatAllCount) * 100)
```

Create dataframe for Bellabeat comparative bar chart. This will pull in percentages for nonusers as well as bellabeat users. Subgroup and Percent create columns that will be used in the next dataframe.

```
subgroup <- c("nonuser", "bellabeat user")
percent <- c(nonuserCountMetPercentCdc, bellabeatCountMetPercentCdc)
```

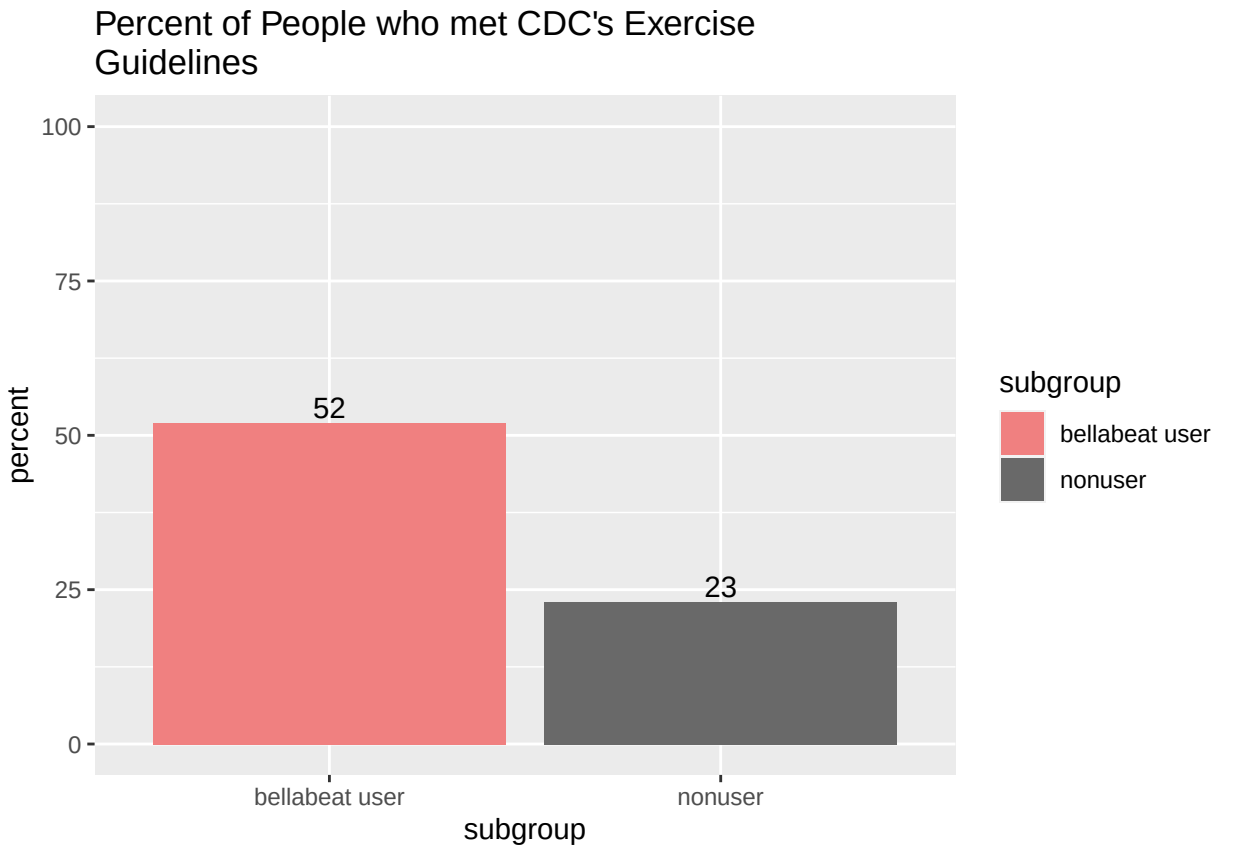
compareBellabeatNonusersMetCdc adds subgroup and percent into a dataframe which will be used to construct the bar chart.

```
compareBellabeatVsNonusersMetCdc <- data.frame(subgroup, percent)
```

Step 5: Data Visualization: Construct the Bar Chart.

Demonstrate the above statistics into a comparative barchart.

```
ggplot(compareBellabeatVsNonusersMetCdc, aes(x = subgroup, y = percent, fill = subgroup)) +
  geom_bar(position = "dodge", stat = "identity") +
  ylim(0, 100) +
  labs(title = stringr::str_wrap("Percent of People who met CDC's Exercise Guidelines", width = 45)) +
  scale_fill_manual(values = c(
    "lightcoral",
    "dimgray"
  )) +
  geom_text(aes(label = percent), position = position_dodge(width = .9), vjust = -.25)
```




```
theme(aspect.ratio = 8 / 8)
```

```
## List of 1
##  $ aspect.ratio: num 1
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi FALSE
##  - attr(*, "validate")= logi TRUE
```