Kevin Ma Calendar Schedule

STAT231: Google Calendar Analysis

Kevin Ma

Due 3/19/21

Contents

0.1 Wrangling Calendar

```
#importing calendar
path <- "/Users/kevinma/Github/Stat231/Homework"
filename <- "kma8222178@gmail.com.ics"

my_calendar0 <- ical_parse_df(file = paste0(path,"/",filename)) %>%
  mutate(start_datetime = with_tz(start, tzone = "America/New_York")
    , end_datetime = with_tz(end, tzone = "America/New_York")
    , length_seconds = end_datetime - start_datetime
    , date = floor_date(start_datetime, unit = "day"))
```

```
#Data wrangling: Create subgroups for each type of activity, filter for relevant dates, get day of the
my_calendar1 <- my_calendar0 %>%
  filter(date > "2021-02-24") %>%
  mutate(
    day = weekdays(date),
    sub_group = case_when(summary == "Computer Science" |
                            summary == "Thesis" |
                            summary == "Data Science" ~ "Classes",
                          summary == "Tennis" |
                            summary == "Workout" ~ "Exercise",
                          summary == "Breakfast" |
                            summary == "Lunch" |
                            summary == "Dinner" ~ "Meals",
                          summary == "Sleep" ~ "Sleep"),
    #if summary is "breakfast, lunch, or dinner", subroup = "meals"
    length_hour = as.numeric(round(length_seconds / 3600, digits = 2))
  )
```

The questions I wanted to answer in this assignment are the following:

1. What is the breakdown of time I spend between eating, schoolwork, and exercise?
2. What is the specific breakdown between my different classes, tennis vs. working out, and different meals?
3. With the total time that I am awake, how productive am I?

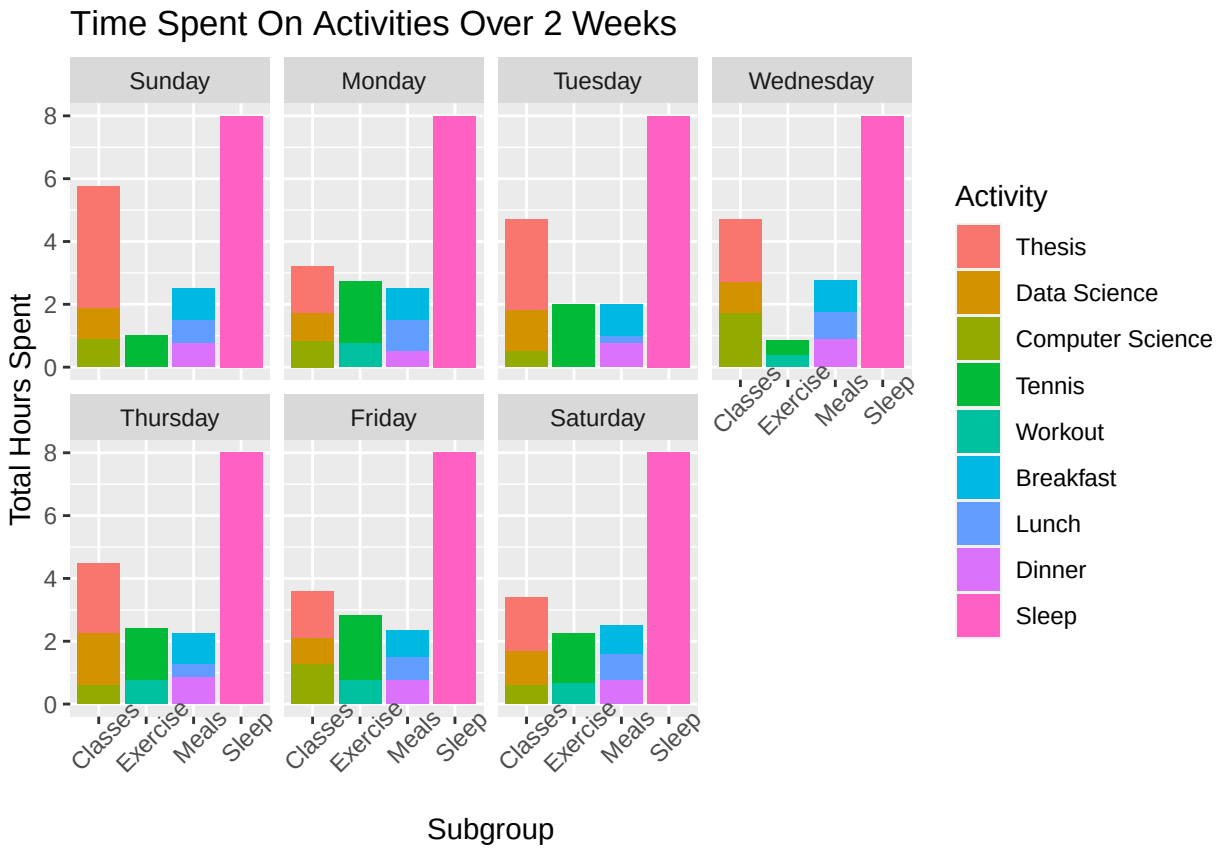
0.2 Graphic 1: Bar Graph Class Breakdown

In this graph I want to compare how I spend my day across different days. I want to see the total amount of time I spend on each activity, sorted by subgroup, for each day. Each panel represents a day of the week, while each stacked bar represents a subgroup. Within each stacked bar, the colors denote the activity. This format allows me to see how long on average I spend on an activity for a given day. For example, I can see that each Sunday I spend on average 4 hours working on my thesis.

```
my_calendar2 <- my_calendar1
#Order the days of the calendar
my_calendar2$day <- factor(my_calendar2$day,
                           levels = c("Sunday", "Monday", "Tuesday", "Wednesday",
                                      "Thursday", "Friday", "Saturday"))
#Order the activities in the legend
my_calendar2$summary <- factor(my_calendar2$summary,
                               levels = c("Thesis", "Data Science", "Computer Science",
                                          "Tennis", "Workout", "Breakfast", "Lunch",
                                          "Dinner","Sleep"))
#Wrangle data set: Find the average time spent on each activity. I record Sunday to Wednesday twice and
my_calendar2 <- my_calendar2 %>%
  mutate(
    average_1hour = case_when(day == "Sunday" |
                                day == "Monday" |
                                day == "Tuesday" |
                                day == "Wednesday" ~ length_hour/2,
                              day == "Thursday" |
                                day == "Friday" |
                                day == "Saturday" ~ length_hour/3)
    ) %>%
  group_by(sub_group, summary, day) %>%
  summarize(tot_time = sum(average_1hour))
```

```
## `summarise()` has grouped output by 'sub_group', 'summary'. You can override using the `.groups` argu
```

```
#plot the visualization
ggplot(my_calendar2, aes(x = sub_group, y = tot_time))+
  geom_col(aes(fill = summary)) +
  facet_wrap(~day, nrow = 2, ncol = 4)+
  labs(x = "Subgroup", y = "Total Hours Spent", fill = "Activity") +
  ggtitle("Time Spent On Activities Over 2 Weeks") +
  theme(axis.text.x = element_text(angle = 45))
```


Time Spent On Activities Over 2 Weeks
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Total Hours Spent
0
2
4
6
8
Classes
Exercise
Meals
Sleep
Subgroup
Activity
Thesis
Data Science
Computer Science
Tennis
Workout
Breakfast
Lunch
Dinner
Sleep

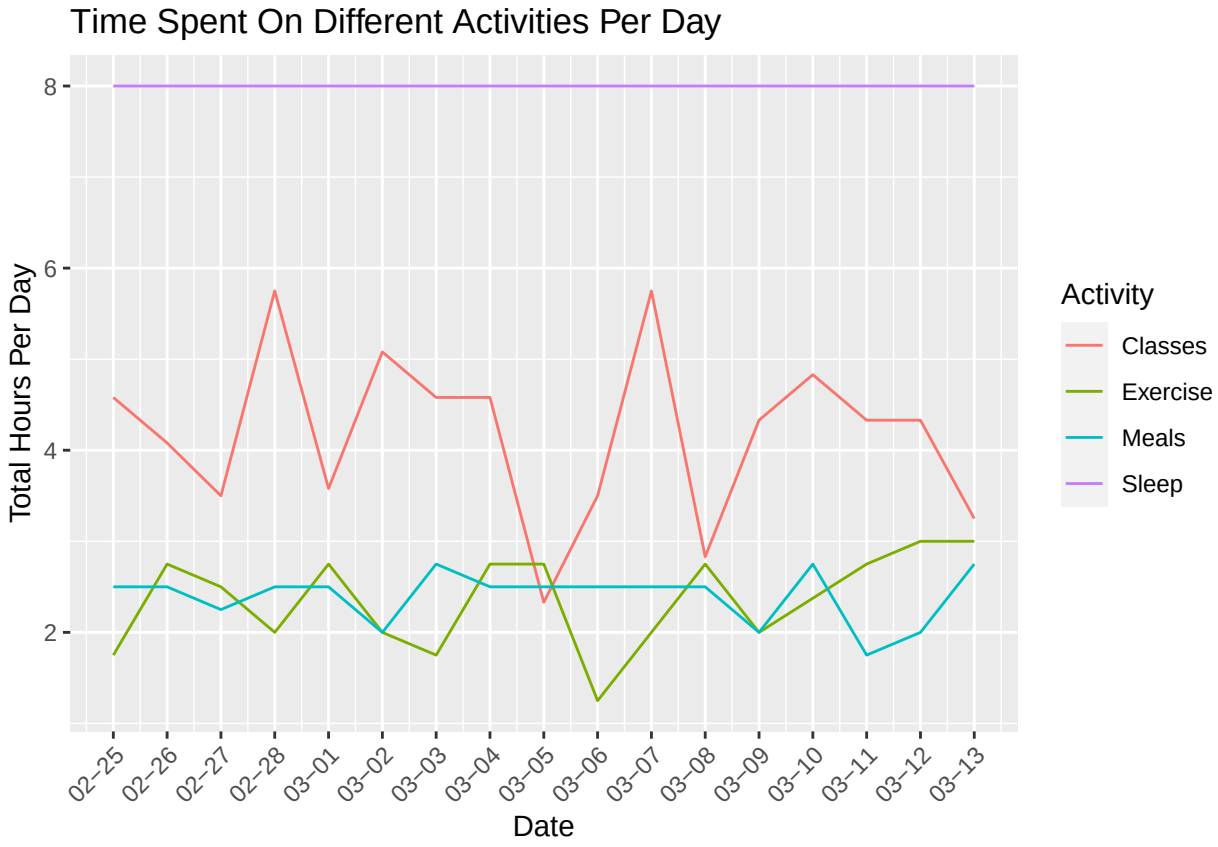
0.3 Graphic 2: Line Graph Visualization

For this visualization, I wanted to record how much time I allocated to class, exercise, meals, and sleep for each day. I had to condense related activities into subgroups. I also had to make sure that there was data representing the total number of hours for each subgroup of activities, as some days I would play tennis and workout or had two classes. I decided to use a line graph that shows the total number of hours spent on each of the four subgroup activities over the recording period.

```
#Data wrangling: Want the sub_group, the date, and the total time spent per sub_group
total_time <- my_calendar1 %>%
  group_by(sub_group, date) %>%
  summarise(total_time = sum(length_hour))
```

```
## `summarise()` has grouped output by 'sub_group'. You can override using the `.groups` argument.
```

```
#Plotting the line chart
 ggplot(total_time, aes(x = as.Date(date), y = total_time, color = sub_group)) +
  geom_line() +
  labs(x = "Date", y = "Total Hours Per Day", color = "Activity") +
  ggtitle("Time Spent On Different Activities Per Day") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_x_date(breaks= "days", date_labels = "%m-%d")
```



0.4 Graph 3: Table

In this table, I want to summarize the amount of time spent on each subgroup for each weekday. In order to do this, I needed the total hours from each subgroup for each day, so I group by date and subgroup and use the `summarize` function to create the total_hours variable. Then, I pivot wider to turn the dataset into a "wide" format. I add the "Other" variable to denote time I spend on miscellaneous activities. Furthermore I create the "Productive to Other Ratio" where I calculate the ratio between classes, exercise, and meals with "Other". The second ratio I create is the ratio between the time spent on classes per day versus the time spent on total activities during the day.

```
#Want wide data because want hours per subgroup per day
caltable <- my_calendar1 %>%
  group_by(date, sub_group) %>%
  summarise(total_hours = sum(length_hour)) %>%
  pivot_wider(id_cols = date, names_from = sub_group, values_from = total_hours)
```

```
## `summarise()` has grouped output by 'date'. You can override using the `.groups` argument.
```

```
#Remove NA values by changing to zero
caltable[is.na(caltable)] = 0
#Rename date column, add "other" column, add a weekday column, add hours productive ratio and classes t
new_caltable <- caltable %>%
  rename(Date = date)%>%
  mutate(
    Other = 24 - Classes - Meals - Exercise - Sleep,
    Day = weekdays(Date),
    `Productive to Other Ratio` = round(sum(Classes, Exercise, Meals)/Other, digits = 2 ),
    `Classes to Total Ratio` = round(Classes/sum(Classes, Exercise, Meals, Other, Sleep),
                                     digits = 2)
  )%>%
  select(Day, Classes, Meals, Exercise, Sleep, Other, `Productive to Other Ratio`, `Classes to Total Ra
```

```
## Adding missing grouping variables: `Date`
```

```
#Use kable package to create the table
kable(new_caltable, booktabs = TRUE, linesep = "", align = "c") %>%
  kable_styling(latex_options = "HOLD_position") %>%
  row_spec(0, bold = TRUE) %>%
  pack_rows("Week 1", 1, 7) %>%
  pack_rows("Week 2", 8, 13) %>%
  pack_rows("Week 3", 14, 17)
```

Date	Day	Classes	Meals	Exercise	Sleep	Other	Productive to Other Ratio	Classes
Week 1								
2021-02-25	Thursday	4.58	2.50	1.75	8	7.17	1.23	
2021-02-26	Friday	4.08	2.50	2.75	8	6.67	1.40	
2021-02-27	Saturday	3.50	2.25	2.50	8	7.75	1.06	
2021-02-28	Sunday	5.75	2.50	2.00	8	5.75	1.78	
2021-03-01	Monday	3.58	2.50	2.75	8	7.17	1.23	
2021-03-02	Tuesday	5.08	2.00	2.00	8	6.92	1.31	
2021-03-03	Wednesday	4.58	2.75	1.75	8	6.92	1.31	
Week 2								
2021-03-04	Thursday	4.58	2.50	2.75	8	6.17	1.59	
2021-03-05	Friday	2.33	2.50	2.75	8	8.42	0.90	
2021-03-06	Saturday	3.50	2.50	1.25	8	8.75	0.83	
2021-03-07	Sunday	5.75	2.50	0.00	8	7.75	1.06	
2021-03-08	Monday	2.83	2.50	2.75	8	7.92	1.02	
2021-03-09	Tuesday	4.33	2.00	2.00	8	7.67	1.09	
Week 3								
2021-03-10	Wednesday	4.83	2.75	0.00	8	8.42	0.90	
2021-03-11	Thursday	4.33	1.75	2.75	8	7.17	1.23	
2021-03-12	Friday	4.33	2.00	3.00	8	6.67	1.40	
2021-03-13	Saturday	3.25	2.75	3.00	8	7.00	1.29	

0.5 Summary

0.6 Reflection