AccelData_Processing

Chris Moore

December 6, 2018

Required packages

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------

## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.1     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0

## -- Conflicts ---------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(RSQLite)
library(PhysicalActivity)
devtools::load_all("./Accelerometer")
```

```
## Loading Accelerometer
```

Step 1: Prepare data

1a) Take 1-sec epoch data exproted by accelerometer software and add a time stamp, reintegrate to 60-sec, add a day variable, and classify wear time to create 'PrepedData' files

```
?acceldata_prep
```

```
## Rendering development documentation for 'acceldata_prep'
```

```
acceldata_prep(input_dir = "./Data", file_pattern = "*1secAGdata.csv",
               output_dir = "./Outputs", subjectID_length = 2,
               subjectID_start = 1)
```

```
## [1] "PrepedData file written in ./Outputs for subject CM"
## [1] "PrepedData file written in ./Outputs for subject EA"
## [1] "PrepedData file written in ./Outputs for subject SD"
## [1] "PrepedData file written in ./Outputs for subject ZG"
```

Also created custom function for reintegrating 1-sec epoch files to 60-sec epoch ('reintegrate_accel' function in R folder, just to try, function not complete with documentation)

1b) Import 'PrepedData' files, use sleep logs to filter out data collected during sleep, and create 'FilteredWakeData' files

```
?sleep_filter
```

```
## Rendering development documentation for 'sleep_filter'
```

```
sleep_filter(acceldata_dir = "./Outputs", sleeplog_dir = "./Data",
             output_dir = "./Outputs", subjectID_length = 2,
             accelfile_pattern = "^PrepedData", sleeplog_pattern = "*SleepLog",
             accelfile_IDstart = 12, sleeplog_IDstart = 1)
```

```
## [1] "FilteredWakeData file written in ./Outputs for subject CM"
## [1] "FilteredWakeData file written in ./Outputs for subject EA"
## [1] "FilteredWakeData file written in ./Outputs for subject SD"
## [1] "FilteredWakeData file written in ./Outputs for subject ZG"
```

Step 2: Examine physical activity with counts

Import prepped and filtered data from Step 1

```
import_data <- read.csv("./Outputs/FilteredWakeData_CM.csv")
```

2) Classify physical activity intensity and summarize MVPA and sedentary time per day, per week and for weekday vs weekend

```
?pai_week_summary
```

```
## No documentation for 'pai_week_summary' in specified packages and libraries:
## you could try '??pai_week_summary'
```

```
pai_week_summary(import_data)

pai_byday
```

```
## # A tibble: 27 x 3
## # Groups:   weekday [?]
##    weekday  pai       minutes
##    <fct>    <ord>       <int>
##  1 Friday   sedentary     365
##  2 Friday   light         110
##  3 Friday   moderate       66
##  4 Friday   vigorous       51
##  5 Monday   sedentary     140
##  6 Monday   light           2
##  7 Monday   moderate        5
##  8 Saturday sedentary     646
##  9 Saturday light         186
## 10 Saturday moderate       60
## # ... with 17 more rows
```

```
pai_weekavg
```

```
## # A tibble: 4 x 2
##   pai       minutes
##   <ord>       <dbl>
## 1 sedentary   454.
## 2 light       141
## 3 moderate     51.3
## 4 vigorous     61.7
```

```
pai_byweekday_end
```

```
## # A tibble: 8 x 3
## # Groups:   week_day_end [?]
##   week_day_end pai       minutes
##   <chr>        <ord>       <dbl>
## 1 weekday      sedentary   426.
## 2 weekday      light       123.
## 3 weekday      moderate     46.6
## 4 weekday      vigorous     83.2
## 5 weekend      sedentary   524.
## 6 weekend      light       186.
## 7 weekend      moderate     63
## 8 weekend      vigorous     18.5
```

Step 3: Examine physical activity with steps

3) Summarize total steps for each day, average steps/day over the week, and average steps/day on weekdays vs weekends

```
?steps_week_summary
```

```
## No documentation for 'steps_week_summary' in specified packages and libraries:
## you could try '??steps_week_summary'
```

```
steps_week_summary(import_data)

steps_byday
```

```
## # A tibble: 7 x 2
##   weekday   steps
##   <fct>     <int>
## 1 Friday     8012
## 2 Monday      191
## 3 Saturday   5777
## 4 Sunday     5953
## 5 Thursday  23169
## 6 Tuesday    8457
## 7 Wednesday 13628
```

```
steps_weekavg
```

```
## # A tibble: 1 x 1
##   steps
##   <dbl>
## 1 9312.
```

```
steps_byweekday_end
```

```
## # A tibble: 2 x 2
##   week_day_end  steps
##   <chr>         <dbl>
## 1 weekday      10691.
## 2 weekend       5865
```

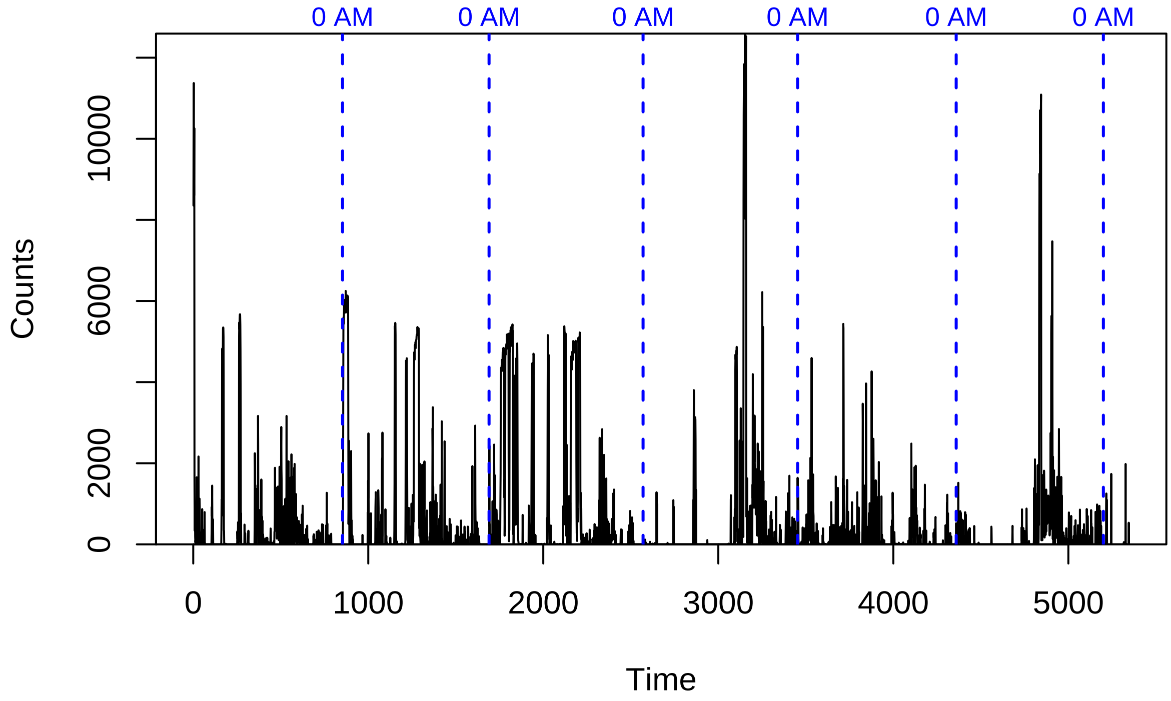
Step 4:Data visualization

4a) Plot counts vs time using 'PhysicalActivity' package

```
?plotData
```

```
## starting httpd help server ... done
```

```
plotData(import_data, cts = "axis1", TS = "TimeStamp")
```

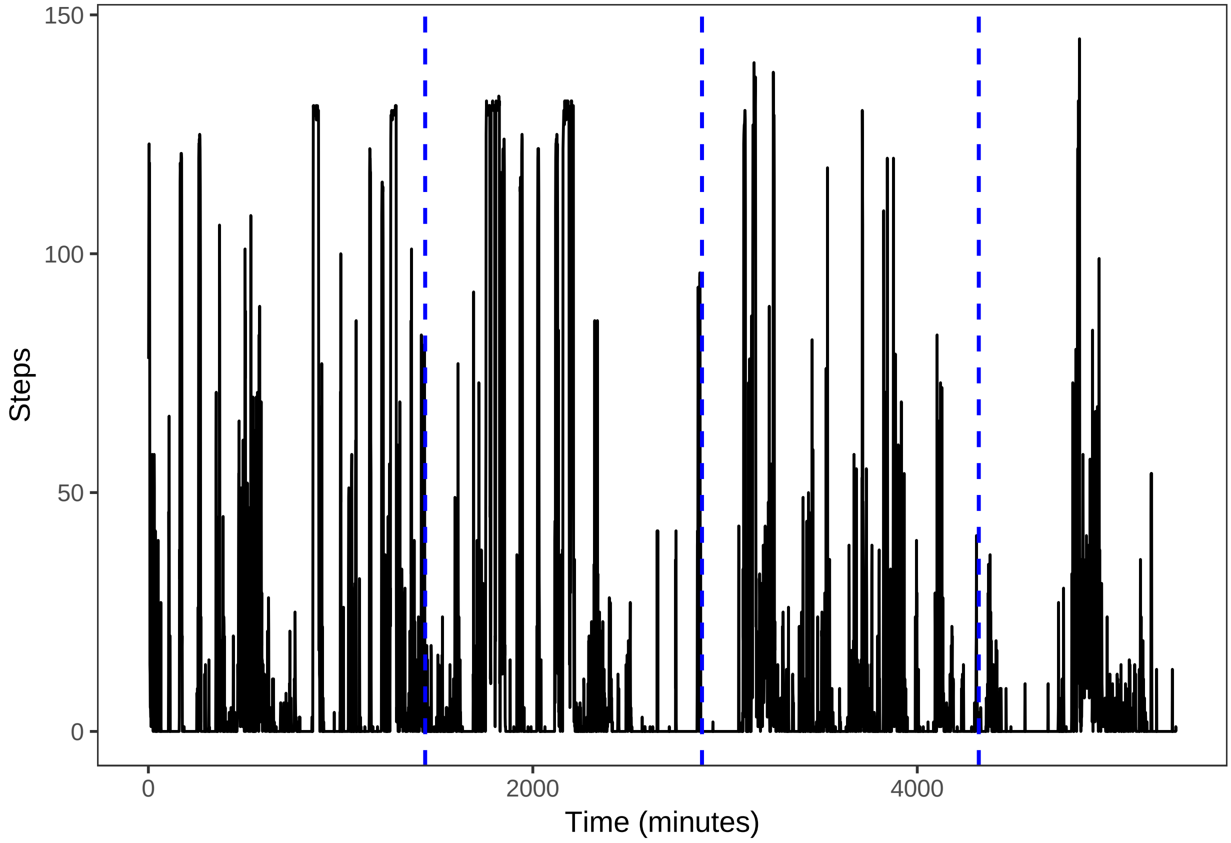


4b) Plot steps vs time

```
?plot_steps
```

```
## Rendering development documentation for 'plot_steps'
```

```
plot_steps(import_data)
```



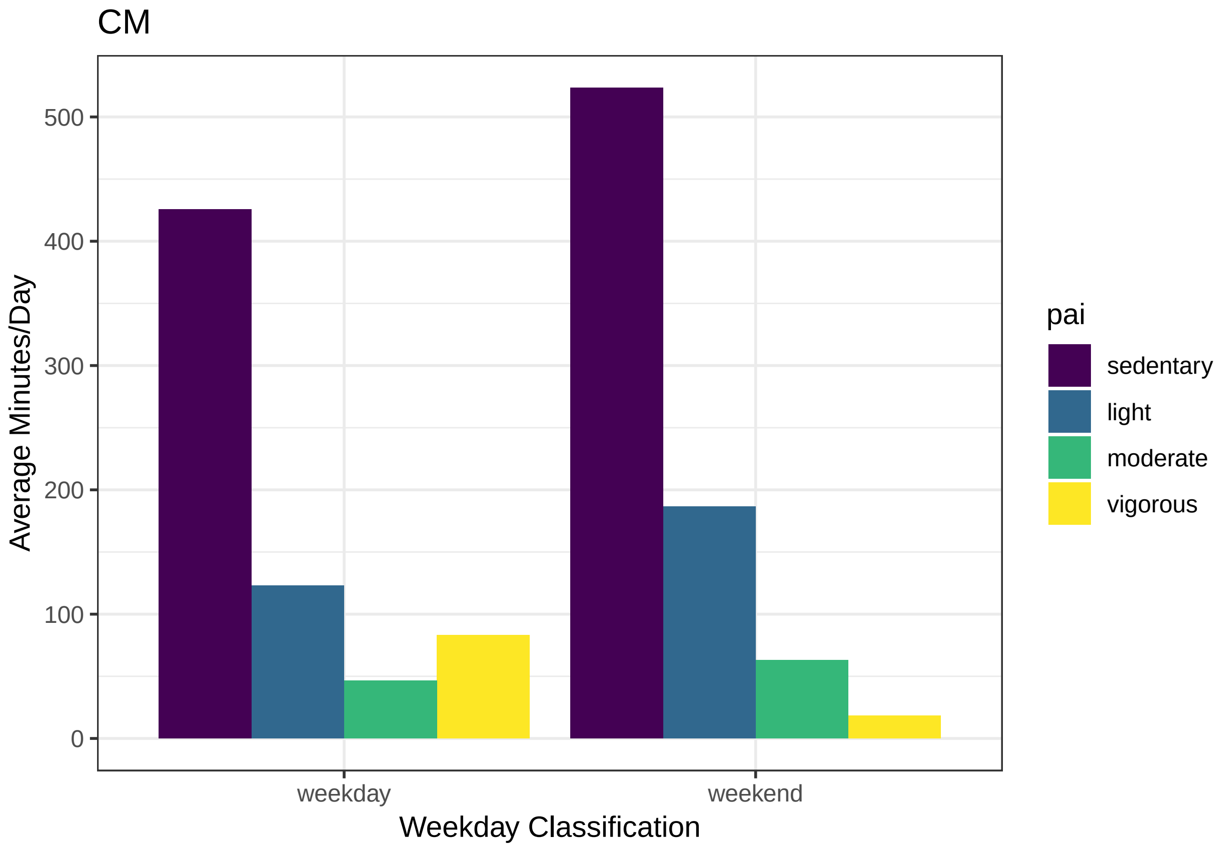
4c) Plot comparison of weekday versus weekend physical activity

```
?plot_weekday_vs_end # REP mvpa_plot
```

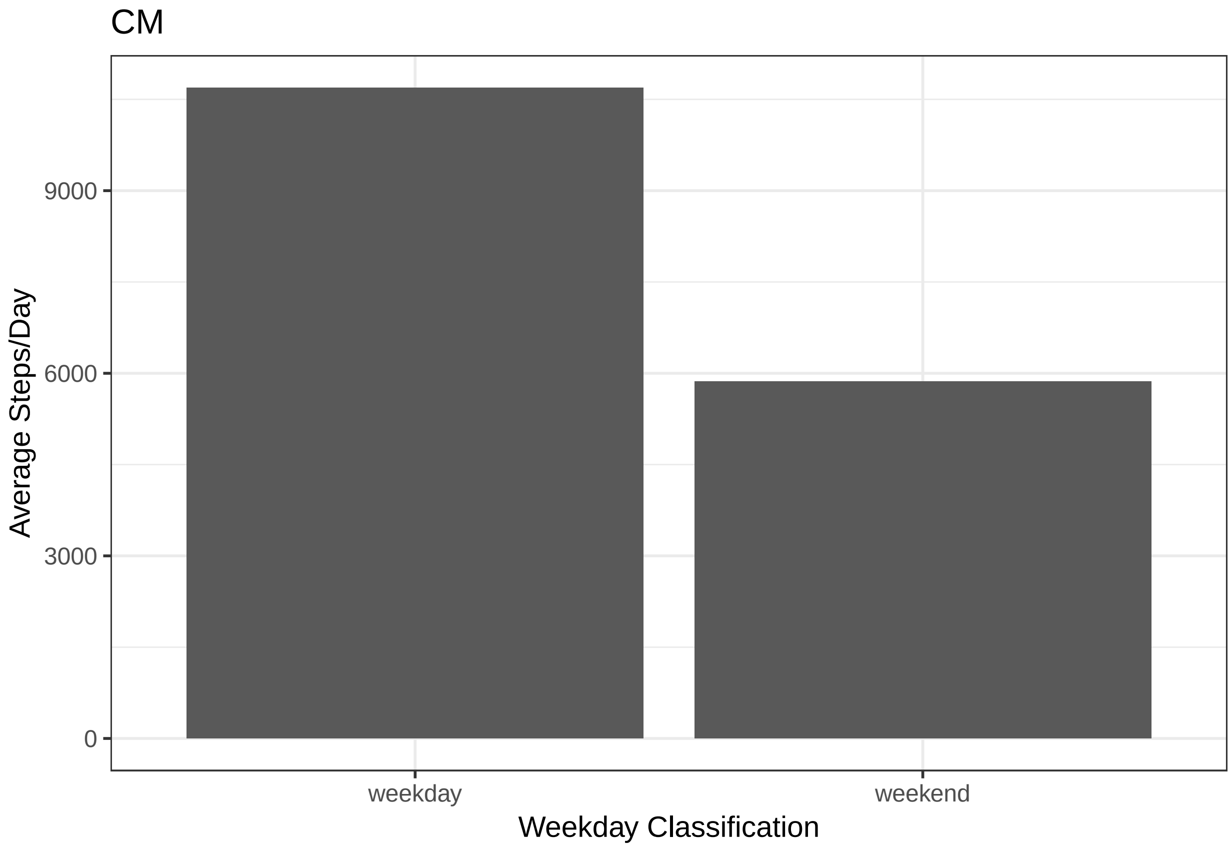
```
## Rendering development documentation for 'plot_weekday_vs_end'
```

```
plot_weekday_vs_end(pai_byweekday_end, steps_byweekday_end, "CM")

plot_pai_dayend
```



```
plot_steps_dayend
```


CM
Average Steps/Day
9000
6000
3000
0
weekday
weekend
Weekday Classification