Are Sleep Trackers Effective?

Shashi Roshan (sroshan2@illinois.edu)

Abstract

Personal sleep tracking devices are becoming more and more popular. Statistical learning techniques are used to determine if it is possible to effectively predict time asleep from data that would be available without the aid of a sleep tracker.

Introduction

It is without question that sleep is a very important process for both learning and memory. [^1] For optimal learning, sleep, both in quality and quantity, is required before and after learning. Depending on certain demographic factors, there are different sleep prescriptions, but for adults, a minimum of seven hours is needed to avoid impairment. [^2] Recently, the link between shift work and cancer has been well established. While more study is needed, there seems to be growing evidence that lack of sleep may play a strong causal role in many cancers. [^3] As the public has become more aware of the importance of sleep, the use of "smart" devices to track sleep has risen. Many sleep trackers provide a wealth of information including not only time asleep, but also details such as time spent in the various stages of sleep. (Light, deep, REM.)

The effectiveness of these sleep devices is still in question. [^4] While the breadth of data that they make available is interesting, the most important by far is the total time asleep. (Asleep being defined clinically, not by simply being in bed.) The additional data, such as time in REM sleep, is interesting, however it is unclear what the target values should be, and more importantly, how we could affect change in these numbers. In contrast, there is a wealth of advice on how to increase quality and time spent asleep. [^5] If total time asleep is the only data worth tracking, is a smart device actually necessary? Is it possible to estimate time asleep based on simple metrics such as time spent in bed?

Statistical learning techniques were applied to a four month sample of data from a Fitbit [^6] user. Time spent in bed was used to predict total time asleep. The results indicate that this prediction can be made with a reasonably small amount of error. However, practical and statistical limitations suggest the need for further investigation.

Methods

Data

The data was accessed via the data export tool provided by Fitbit. [^7] It was collected using a Fitbit Versa 2 by a single subject, a 32 year old adult male living in Ohio and working as a professor. The Fitbit Versa

2 uses both motion and heart rate variability [^8] to predict when the user is sleeping. The collection dates were a series of consecutive days in autumn of 2018. The two quantities of interest in the data are the time spent asleep and the time spent in bed each time the user sleeps. (A user could sleep more than once a day. For example, a two hour nap in the afternoon.) If the former can be predicted from the latter, the device seems unnecessary. (Time spent in bed could simply be tracked manually by a user. Although, it should be noted that one of the benefits of the devices is the automatic tracking of this quantity, which is probably more accurate than manual human tracking.)

```
sleep = read_csv(file = "data/fitbit-export-20190904.csv")
```

```
sleep_trn = head(sleep, n = 100)
sleep_tst = tail(sleep, n = 22)

sleep_est = head(sleep_trn, n = 80)
sleep_val = head(sleep_trn, n = 20)
```

Modeling

In order to predict time asleep given time in bed, three modeling techniques were considered: linear models, k-nearest neighbors models, and tree models. No transformations were considered with the linear model. Default tuning parameters were used to train the two non-parametric models. Only time in bed was used as a predictor variable.

```
lm_mod   = lm(min_asleep ~ time_bed,     data = sleep_est)
knn_mod  = knnreg(min_asleep ~ time_bed, data = sleep_est)
tree_mod = rpart(min_asleep ~ time_bed,  data = sleep_est)
```

Evaluation

To evaluate the ability to predict time asleep with these models, the data was split into estimation, validation, and testing sets. Because of the dependence structure of the data, that is the consecutive nature of the days, the data was split chronologically. That is, the test set is the last 20% of the data chronologically. (And similarly for the validation data.) This is done to evaluate the ability to predict future nights of sleep from past data. Error metrics and graphics are reported using the validation data in the Results section.

Results

```
calc_rmse = function(actual, predicted) {
  sqrt(mean( (actual - predicted) ^ 2) )
}
```

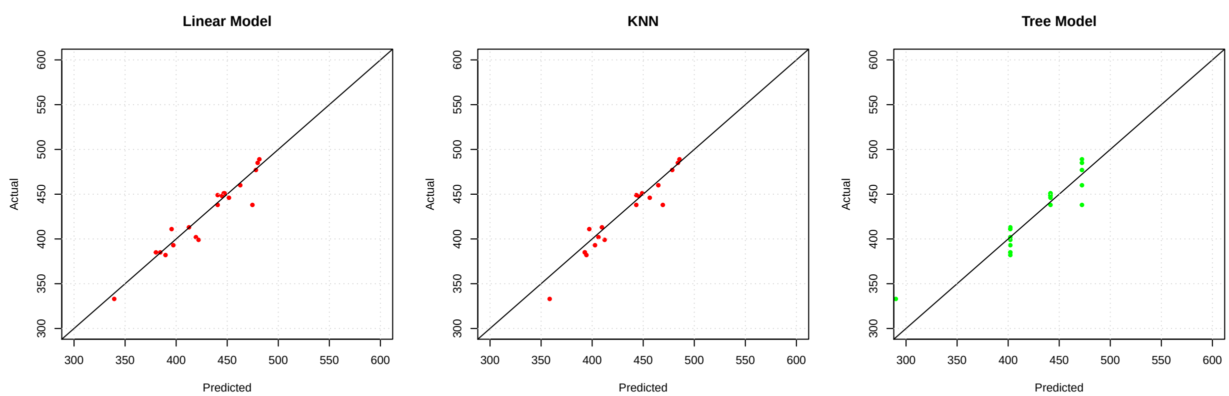
```
sleep_val_rmse = c(calc_rmse(actual = sleep_val$min_asleep,
                             predicted = predict(lm_mod, sleep_val)),
                   calc_rmse(actual = sleep_val$min_asleep,
                             predicted = predict(knn_mod, sleep_val)),
                   calc_rmse(actual = sleep_val$min_asleep,
                             predicted = predict(tree_mod, sleep_val)))
```

```r
model_results = tibble(
ml_model = c("lm_mod","knn_mod","tree_mod"), RMSE = sleep_val_rmse
)

kable(model_results) %>%
  kable_styling(bootstrap_options = c("striped"), full_width=F)
```

ml_model	RMSE
lm_mod	11.82016
knn_mod	11.41247
tree_mod	16.17842

```r
par(mfrow = c(1, 3))
plot(predict(lm_mod, sleep_val), sleep_val$min_asleep,
     xlim = c(300, 600), ylim = c(300, 600), pch = 20, col = "red",
     xlab = "Predicted", ylab = "Actual",
     main = "Linear Model")
abline(a = 0, b = 1, col = "black")
grid()
plot(predict(knn_mod, sleep_val), sleep_val$min_asleep,
     xlim = c(300, 600), ylim = c(300, 600), pch = 20, col = "red",
     xlab = "Predicted", ylab = "Actual",
     main = "KNN")
abline(a = 0, b = 1, col = "black")
grid()
plot(predict(tree_mod, sleep_val), sleep_val$min_asleep,
     xlim = c(300, 600), ylim = c(300, 600), pch = 20, col = "green",
     xlab = "Predicted", ylab = "Actual",
     main = "Tree Model")
abline(a = 0, b = 1, col = "black")
grid()
```



Discussion

```r
sleep_tst_rmse = calc_rmse(actual = sleep_tst$min_asleep,
                           predicted = predict(lm_mod, sleep_tst))

sleep_tst_rmse
```

```
## [1] 9.224105
```

Selected model : Linear model. Training RMSE 9.224105.

From the above results, linear model and KNN model seems to be good models for predicting time asleep. While the performance of linear model and KNN is almost similar, a linear model can be favored since it is a simple model. Also, KNN model seems to be overpredicting the values. Tree model seems to be predicting only few discrete values.

- Statistical limitations :
  - Dataset may be small.
  - Dataset is taken from one user, and can be biased towards their sleep habits.
- Practical limitation :
  - If a user often forgets to capture sleep data, then a sleep tracking device is useful.
- In Future analysis :
  - We can take data from multiple users
  - Take a larger data set
  - Use more features

Appendix

Data Dictionary

- `start_time` - The date and time which the device detected the user has gone to bed due to lack of motion. (But not necessarily started sleep.)
- `end_time` - The date and time which the device detected that the user is no longer in bed, due to motion.
- `min_asleep` - The total sleep time, in minutes. This is meant to estimate a clinical measure of sleep. (Not simply time in bed.)
- `min_awake` - The time spent in bed, but awake, in minutes.
- `num_awake` - The number of times the user "awoke" during their time in bed.
- `time_bed` - Duration between `start_time` and `end_time`. The sum of `min_asleep` and `min_awake`. In other words, total time in bed, in minutes.
- `min_rem` - Total time spent in REM sleep, in minutes.
- `min_light` - Total time spent in light sleep, in minutes.
- `min_deep` - Total time spent in deep sleep, in minutes.