Heart Rate Capabilities of Wrist Worn Monitors: A Time Series Analysis Approach

A Two Part Analysis on Model Comparison of Heart Rate across Single Aerobic Session and the Physiological Models between Eight Lagged Biometric Variables

Magdalene Mlynek
STAT 4825
December 11, 2019

INTRODUCTION

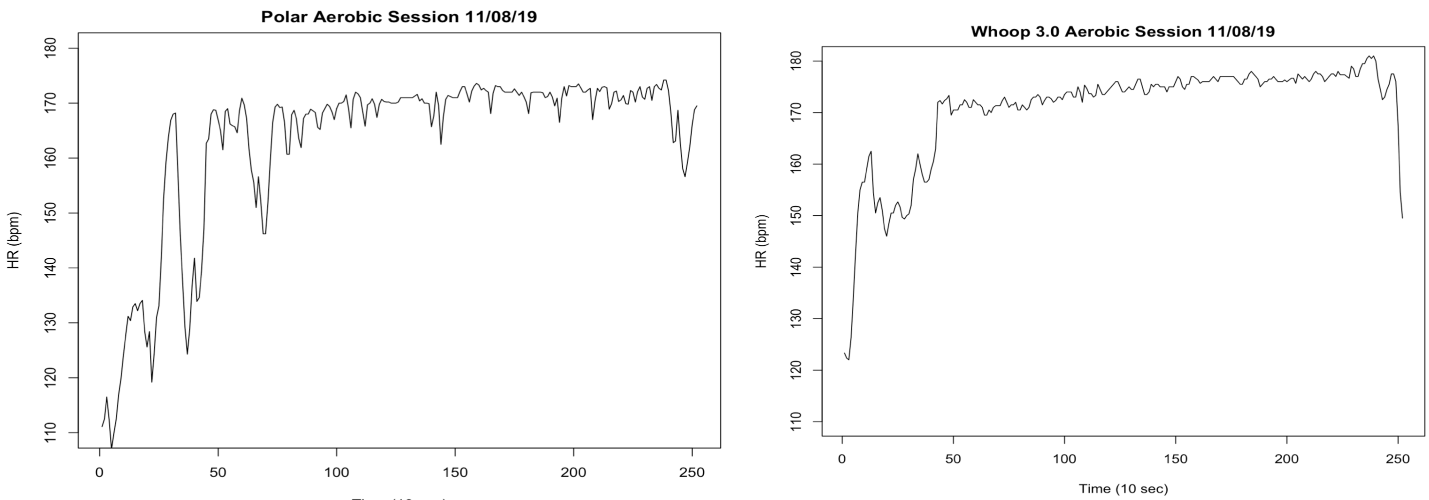
In recent years, personal heart rate monitors have grown in popularity and are used anyone from professional athletes to the average weekend warrior. Athletes in particular have begun to utilize this technology to monitor their training and recovery status. These personal fitness tracking devices can track many variables at any given moment, including biomarkers such as heart rate, heart rate variability, heart rate zone, resting heart rate, as well as training characteristics such as distance and pace, and sleep variables such as time spent in each sleep phase. In addition to activity and biomarker tracking, many of the more sophisticated monitors are able to input the data into a model or algorithm to provide a score that indicates how well the user recovered or the strain of their training. This data can also be utilized to adjust the athletes training in order to allow the athletes to recover and minimize risk of injury. Most of these heart rate monitors are worn on the wrist or around the chest. However, "Convenience and comfort of the wrist-based devices has enabled them to largely replace chest straps that employ electrodes that measure cardiac electrical activity" (Pasadyn, 2019). The heart rate monitor worn on the chest has been set as the gold standard for heart rate monitors and is used to determine the accuracy of wrist worn monitors. Because the chest-worn heart rate monitors lack much of the sophistication and ease that the wrist monitors have, recent studies have attempted to compare the accuracy of the monitors. In this two-part analysis, I will (1) compare the heart rate capabilities of the Polar Ignite and WHOOP 3.0 wrist worn fitness tracker during an aerobic training session using ARIMA models, and (2) determine the physiological relationship between eight daily, lagged variables collected by the WHOOP 3.0 over the course of one month, using the vector autoregression method (VAR).

Data Collection Method

For the first analysis, I wore the two monitors during a longer aerobic training session done on November 8, 2019. After making sure that the two monitors were snug and they were set to record a training session, I completed 10 kilometers on the ergometer at high effort, beginning at ~06:15:00.0 and ending at ~06:45:45.8 (total time = 40 minutes, 45.8 seconds).

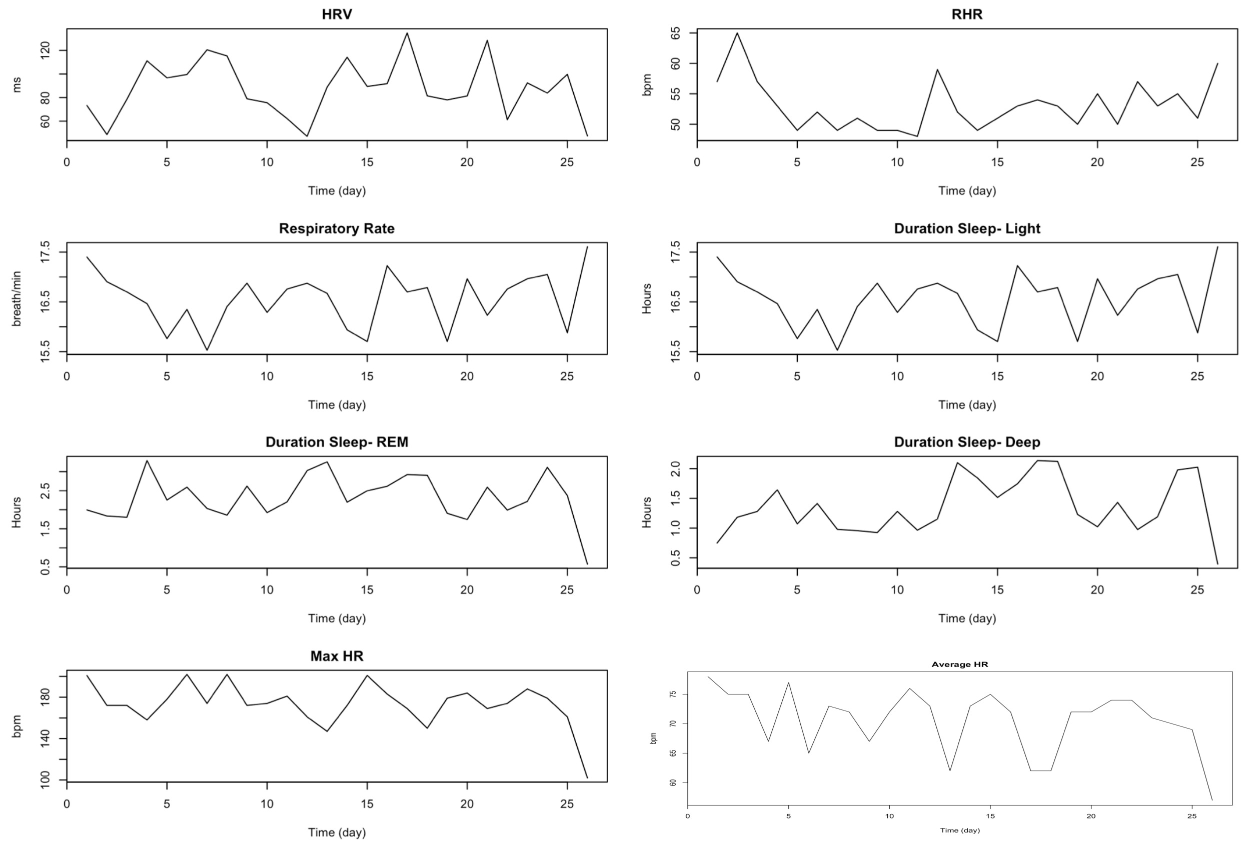
I was able to download the heart rate dataset from the Polar watch as a csv file from the Polar Flow website. The file gave a heart rate for every second. On the other hand, I had difficulties accessing the data from this session on the WHOOP website. I was unable to download a file, but was able to see a plot of my heart rate over the duration of the session, from which I was able to record the observations by hand. However, from the graph on the WHOOP website, I was only given 6-9 observations per minute. I used a simple smoothing technique for each minute during the session in order to have 6 observations per minute, where the first observation of the minute corresponded to xx:00, the second corresponded to xx:10, and so on. In order to mimic this database in the polar dataset, I averaged the observations for the first 10 seconds to give one observation corresponding to xx:00, the second 10 observations to correspond to xx:10 and so on. Using this management scheme, we have 6 observations for each minute, giving one observation every 10 seconds, for both the Polar and the WHOOP monitors across the duration of the session (n=252). The two files used in this part of the analysis are "polar_1108_endurance_session.csv" and "whoop_1108_endurance_session.csv".

Below are the time plots for both straps, showing HR (bpm) versus time (10 seconds). We can notice a small difference between the plots and we will determine if this difference is significant by comparing their models.



Plots 1 & 2- Time series of aerobic session on Polar and WHOOP monitors

For part two, I was able to download the daily values of (1) heart rate variability, (2) resting heart rate, (3) respiratory rate, (4) duration of light sleep, (5) duration of REM sleep, (6) duration of deep sleep, (7) average heart rate and (8) maximum heart rate. I was able to collect these values over the course of three months (September – November 2019), however I was missing too many data to be able to utilize all of the data using a time series analysis. This is most likely because the battery on the monitor died. For this reason, I chose the longest duration of continuous data throughout the 3 months, which was September 26- October 21 (n=26). I completed part two of the analysis using the “whoop_3.0.csv” file.



Plots 3-10: 8 daily variables over one month (26 days)

GOAL OF THE ANALYSIS

In a recent study, the accuracy of the heart rate monitor feature of four wrist worn monitors were compared to a telemetry-based chest strap monitor, at six different treadmill speed and exertion levels, using a sample size of 50 healthy, athletic adults. This study found the four wrist worn monitors demonstrated a "moderate to high level of accuracy" compared to the chest worn monitor (Pasadyn, 2019). Another study used a single subject design to determine the accuracy of two different wrist worn heart rate monitors as compared to the gold standard reference method, an ambulatory electrocardiogram, over a 24-hour period across 5 different daily conditions (sitting, walking, running, activities of daily living and sleeping). This study found that the two wrist worn monitors were "generally highly accurate" compared to the ECG across the 24 hours (Nelson, 2019).

As an athlete, I personally have used different heart rate monitors to track my training, sleep and recovery. I purchased the WHOOP 3.0 strap in August 2019 in order to gain a better understanding how my sleep was affecting my athletic performance and recovery. In September 2019, I received a Polar Ignite through UConn Athletics. Both of these watches collect similar data, were worn 24/7 and display a recovery and sleep score (WHOOP 3.0 also calculates a training score, which Polar Ignite does not). It should be noted that for the duration of the data collection, the WHOOP 3.0 was worn on the right (dominant) wrist and the Polar Ignite was worn on the left (non-dominant) wrist. During my first week of wearing the two different, wrist worn heart rate monitors, I noticed a difference in the duration of my sleep cycles as well as my HRV and RHR. Interested in determining the significance of this difference, I realized I did not have the resources to compare the monitors to the gold standard ECG, I decided to research only if there was a significant difference in the heart rate capabilities, recovery and sleep variables. In this project, I decided to use time series analysis to determine any differences between the models created from the two monitors during an aerobic training session. I also decided to complete a vector series analysis on only the WHOOP 3.0 data, collected daily over the course of a one-month period, to determine the relationship between 8 variables and over what lags they are related.

For part one of the analysis, I used the ARIMA(p, d, q) model and in part two I utilized the VAR(n) model.

Comparison of Recovery, Sleep and Training Status

Since August 2019, I have worn the WHOOP Strap 3.0, a wrist strap that tracks HRV, resting heart rate, continuous heart rate, sleep performance and other variables to display three different scores for the user each day. The recovery score uses HRV, resting heart rate and sleep cycles (calibrated to the user's baseline) to calculate the user's recovery on a scale 0-100%. The Strain score uses heart rate data collected 24/7 to measure the cardiovascular load over the course of the day and during the course of a training session, given on a scale of 0-21. The mobile app also gives a more detailed heart rate record on the mobile app, including time spent in each heart rate zone. Lastly, the sleep score utilizes the data collected from your sleep cycles, including the duration spent in light, REM and deep sleep and awake, as well as RHR, respiratory rate and number of interruptions, to calculate an overall sleep score on a scale of 0-100%.

Received through UConn Athletics, I have been using the Polar Ignite since September 2019, and has very similar features to the WHOOP 3.0. The monitor gives a detailed daily activity report which includes percentage activity of the physical activity goal set by the user, number of steps, number of calories burned, max heart rate and average heart rate. This monitor is also able to collect HRV, RHR, beat-to-beat interval, respiratory rate, total time spent
For each training session, the monitor gives the calories burned and the minimum, maximum, average heart rates and time spent in each heart rate zone. For each training session it uses these variables to calculate a cardio load value. In outdoor workouts, it can also determine distance, speed, and altitude during the session.

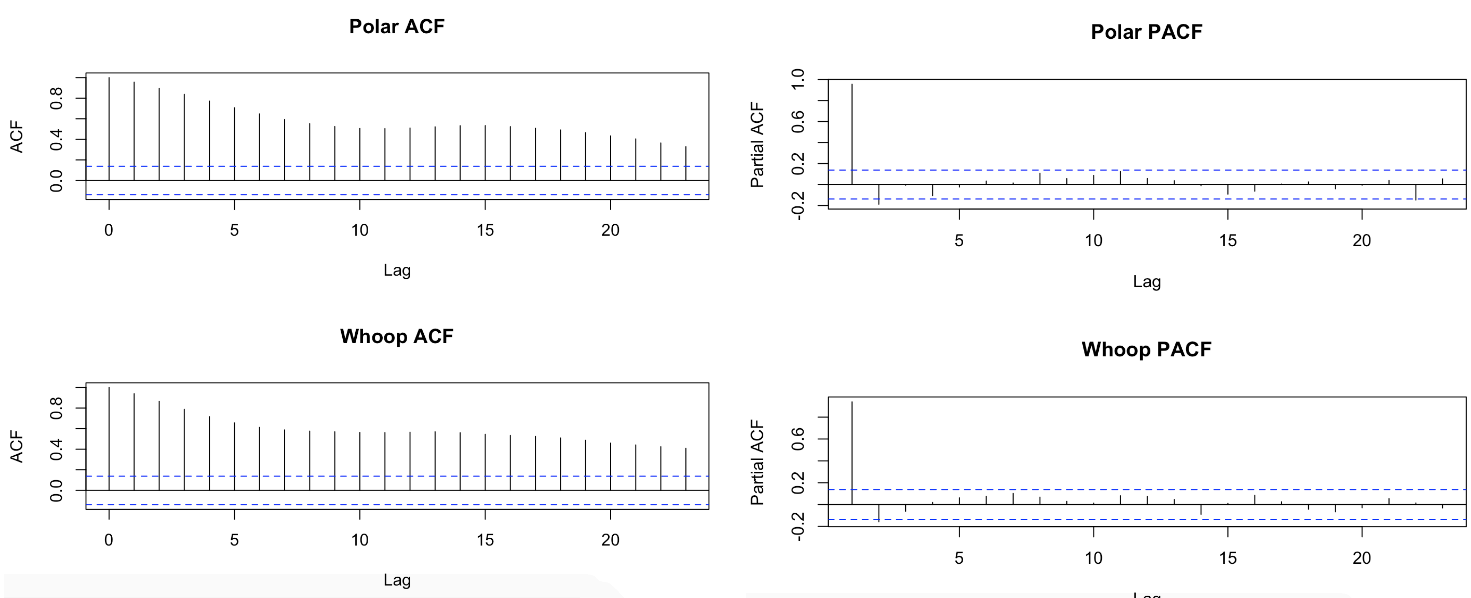
We should notice that while there are differences between the way these two monitors calculate and display training load, sleep quality and recovery status, the variables collected are very similar.

COMPREHENSIVE DATA ANALYSIS

Part 1: WHOOP 3.0 Versus Polar Ignite Heart Rate Capabilities

Data Preparation
In comparing the heart rate capabilities of the two monitors over the course of an aerobic training session, I first prepared the data sets. I decided to split the dataset into a calibration and holdout dataset in order to ensure that my model is adequate. The first 80% of observations (observations 1-202) were put into the calibration set from which I created the model. The holdout set, observations (203-252) was used to test the model by seeing how well the model can predict these observations. Calibration and holdout sets were created from both the WHOOP and Polar time series datasets.

Model Selection
In creating the models, I first created ACF and PACF plots to find any obvious lags. Looking at the PACF plot, lag 2 is outside of the 95% confidence interval bands so it seems that both datasets might be AR(2) models:



Plots 11-14: ACF and PACF plots for WHOOP and Polar HR datasets over aerobic session

I used the auto.arima() function in R to confirm the model, but found that the **ARIMA(0,1,1) model** was more appropriate for the Polar time series, and the **ARIMA(0,1,1) model with drift** was more appropriate for the WHOOP time series. I ran the AR(2) and ARIMA(0,1,1) models for the Polar series and found that the AIC, ME, and MAPE were all lower in the ARIMA(0,1,1) model. Three out of the 6 selection criteria indicate that the AR(2) model is better, while the other three indicated that the ARIMA(0,1,1) model is more appropriate. Ultimately, I chose to use the ARIMA(0,1,1) model because while there were very small differences in the ME, RMSE, MAE, MPE and MAPE, the AIC value is much lower in this model. Following the same model selection process and comparing the selection criteria for the WHOOP data, I found that the AIC, ME, RMSE, MAE, MPE and MAPE are all lower in the ARIMA(0,1,1) with drift model, indicating that this is the more appropriate model.

Selection Criteria:

Selection Criteria	AR(2)	ARIMA(0,1,1)
AIC	1063.08	1051.37
ME	0.2436157	0.2231089
RMSE	3.264746	3.265977
MAE	2.177656	2.180326
MPE	0.1232946	0.1443099
MAPE	1.444731	1.44145

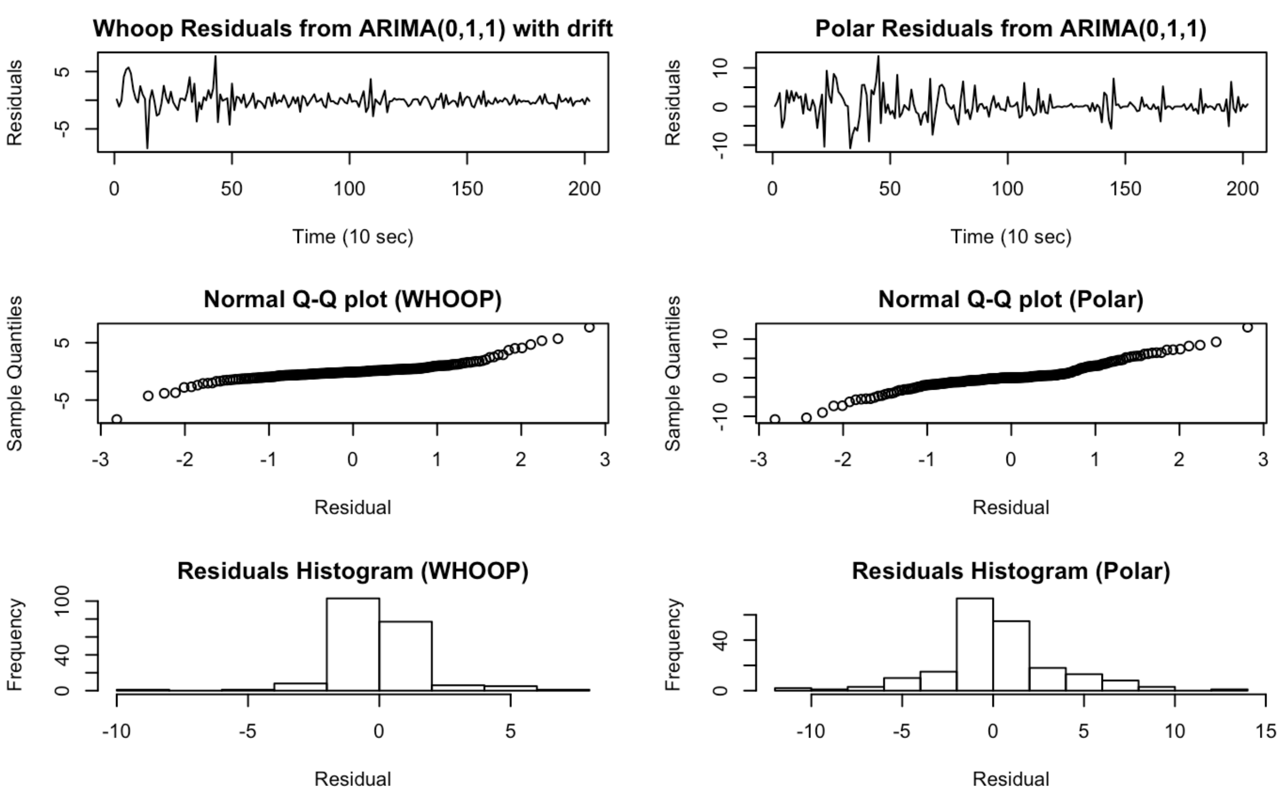
Table 1- Selection criteria for Polar monitor

Selection Criteria	AR(2)	ARIMA(0,1,1) with drift
AIC	763.7	749.01
ME	0.1721025	-0.001669503
RMSE	1.550568	1.531327
MAE	0.9748338	0.9341931
MPE	0.1067633	0.01295204
MAPE	0.5986983	0.571342

Table 2- Selection criteria for WHOOP monitor

Model Accuracy

Several analyses were completed for both Polar and WHOOP data, on the residuals of these models to determine the accuracy, including the Shapiro-Wilk test, Ljung-Box Portmanteau test, McLeod-Li Portmanteau test and the prediction of the holdout sets.



Plots 15-20: Residual plots to determine model accuracy

In the WHOOP model adequacy testing, I also used the Shapiro-Wilks test for normality of the residuals and the Ljung-Box test to test for significant auto-correlations and model adequacy:

WHOOP MODEL	Shapiro- Wilk Test	Ljung-Box Test
Hypotheses	H_0: normal H_A: not normal	H_0: $\rho_1 = \rho_2 = 0$ H_A: at least one auto-correlation not equal to zero
p-value	1.29e^-13	0.5462
Conclusion	**Residuals not normally distributed**	**Model is adequate**

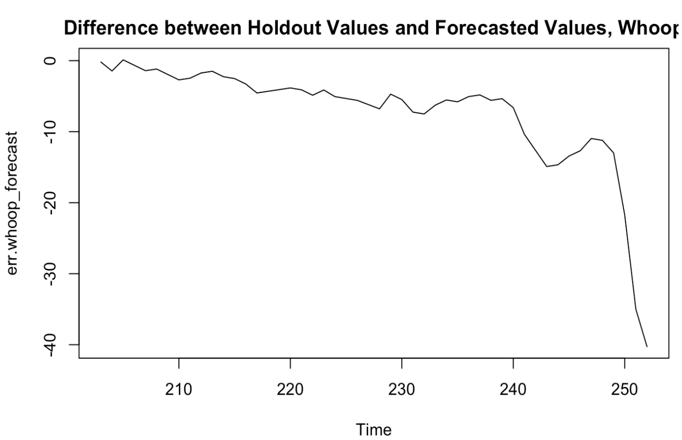
Table 3: Model adequacy test results for WHOOP model

Similarly, I found the following for the Polar model:

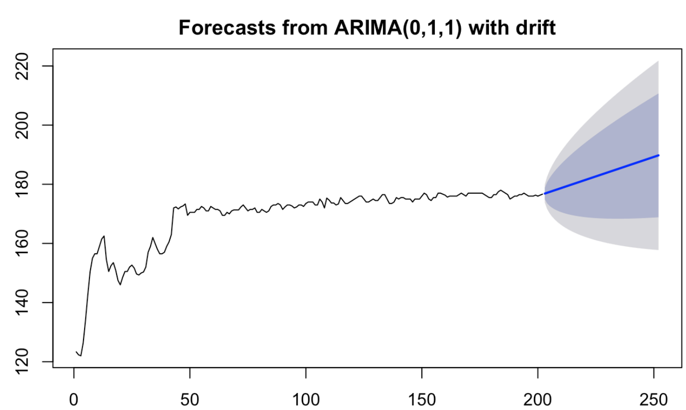
POLAR MODEL	Shapiro- Wilk Test	Ljung-Box Test
Hypotheses	H_0: normal H_A: not normal	H_0: $\rho_1 = \rho_2 = 0$ H_A: at least one auto-correlation not equal to zero
p-value	2.808*10^-7	0.9704
Conclusion	**Residuals not normally distributed**	**Model is adequate**

Table 4: Model adequacy test results for Polar model

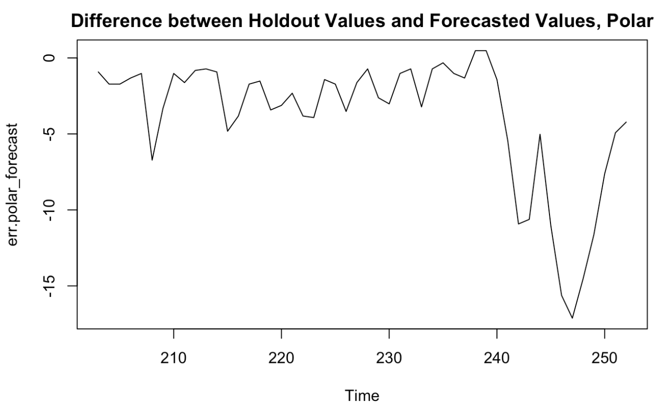
Looking at the difference between the holdout values and the forecasted values, we get the following plots:



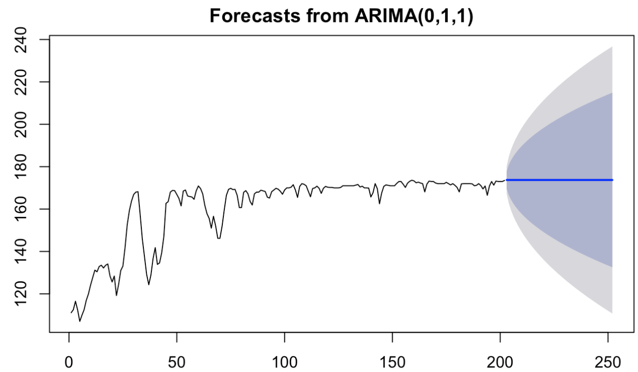
Plot 21: Difference between Holdout and Forecasted Values, WHOOP



Plot 22: Forecasted Values from ARIMA(0,1,1) with drift model, WHOOP



Plot 23: Difference between Holdout and Forecasted Values, Polar



Plot 24: Forecasted Values from ARIMA(0,1,1) model, Polar

From these plots and the ME, MPE, MSE, MAE and MAPE values calculated, we can see that the farther the models predicts into the future, the worse the models performs.

Model Parameter Comparison

In order to compare the models of the WHOOP and Polar monitors, I compared the model parameters using a 95% confidence interval and t test. It should be noted that the WHOOP model has a drift coefficient, while the Polar model does not. Below are the two mathematical models:

$$\text{Polar}: \ y_t = y_{t-1} + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

$$\text{WHOOP}: \ y_t = y_{t-1} + \beta_0 + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

With estimated model parameters:

$$\text{Polar}: \ y_t = y_{t-1} + 0.39\varepsilon_{t-1} + \varepsilon_t$$

$$\text{WHOOP}: \ y_t = y_{t-1} + 0.2637 + 0.5056\varepsilon_{t-1} + \varepsilon_t$$

The 95% CI of the Polar θ_1 parameter is (0.260, 0.522), while the 95% CI of the WHOOP θ_1 parameter is (0.389, 0.622) and the 95% CI of the WHOOP drift parameter is (0.147, 0.380). From the confidence interval of the drift parameter, we can conclude that the drift is not equal to 0 and is a statistically significant parameter in the model. I then conducted a t test to determine if the MA parameters are equal where H_0: $\theta_{1,\text{WHOOP}} = \theta_{1,\text{Polar}}$ and H_A: $\theta_{1,\text{WHOOP}} \neq \theta_{1,\text{Polar}}$. From this t test (p=0.2834), and we can conclude that there is not difference between the MA parameters, however the WHOOP model has a significant drift while the Polar model does not.

PART 2: Physiological Models between Eight Lagged WHOOP 3.0 Biometric Variables

Over one month of WHOOP data collection, I have analyzed the time series relationship across 8 variables using daily values over the course of one month (September 26, 2019- October 21, 2019). The variables I am interested in are:

- **Heart rate variability** (HRV, mms)- daily value corresponds to HRV during previous night of sleep
- **Resting heart rate** (RHR, bpm)- daily value corresponds to resting heart rate during previous night of sleep
- **Respiratory rate** (breaths/min)- daily value corresponds to respiratory rate during previous night of sleep
- **Duration in light sleep** (hours)- daily value corresponds to duration of light sleep during previous night of sleep
- **Duration in REM sleep** (hours)- daily value corresponds to duration of REM sleep during previous night of sleep
- **Duration in deep sleep** (hours)- daily value corresponds to duration of deep sleep during previous night of sleep
- **Average heart rate** (bpm)- daily value corresponds to average heart rate of that day, calculated at the end of the day (not including during sleep)
- **Maximum Heart Rate** (bpm)- daily value corresponds to maximum heart rate of that day, calculated at the end of the day

In setting up my dataset, I had data from September 26, 2019 to November 14, however I was missing data for October 22-October 30, so I decided to only use the data from September 26-October 21 to build the model. I considered using the na.interpolation() function in R, however I felt that 9 days of data was far too much to estimate, which could have a negative impact on the model.

I chose to use a VAR model because this would allow me to test the time series relationships across all the variables and produce one model used to predict each variable and lag according to the other 7.

After removing the October 31- November 14 observations from the dataset, I used plotted the time series and used the auto.arima() function on each variable, and was able to determine that all 8 models were white noise processes. In order avoid any collinearity problems, I checked using the correlation plot and matrix:

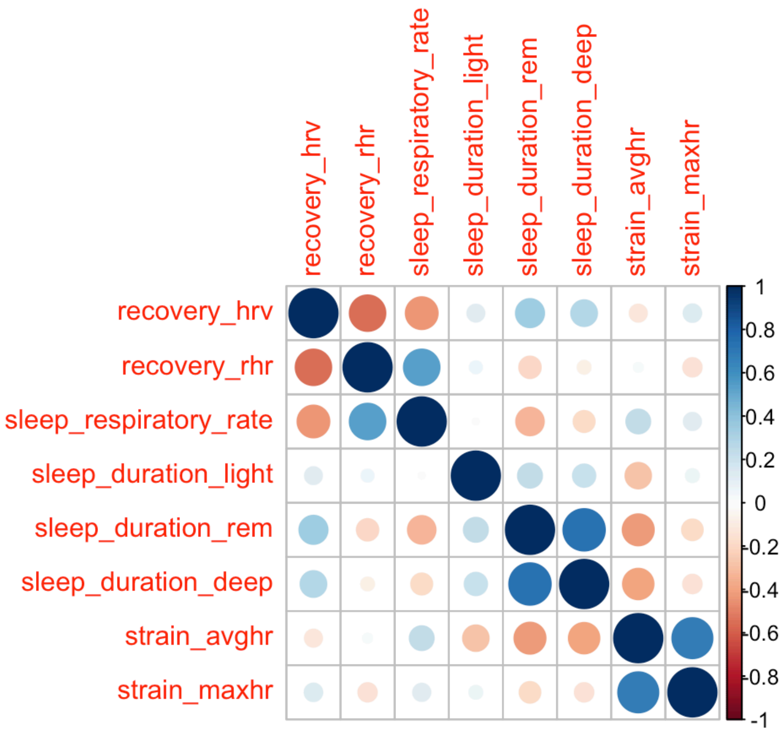


Figure 1: Correlation Plot for 8 daily variables

From this correlation plot, I noticed that there may be a relationship between the following variables:

- RHR and HRV (p=0.0000)
- Resp Rate and HRV (p=0.0006)
- Resp Rate and RHR (p=0.0000)
- Duration REM and HRV (p=0.0090)
- Duration REM and Resp Rate (p=0.0120)
- Duration Deep sleep and HRV (p=0.0298)
- Duration Deep Sleep and REM (p=0.0000)
- Avg HR and Duration Light (p=0.0332)
- Avg HR and Duration REM (p=0.0012)
- Avg HR and Duration Deep (p=0.0019)
- Max HR and Avg HR (p=0.0000)

Using the cor() function, I determined that I did not need to remove any of the variables from the time series because none of the values were over 0.95.

Then with the VARselect() function, I selected the best model based on the SC criterion and found that it was the **VAR(2) model**, where lag 2 is significant. With this information I can estimate the model parameters for each of the 8 models. The detailed output for these models are shown in the Appendix. The following figure shows the relationships between the variables and lags.

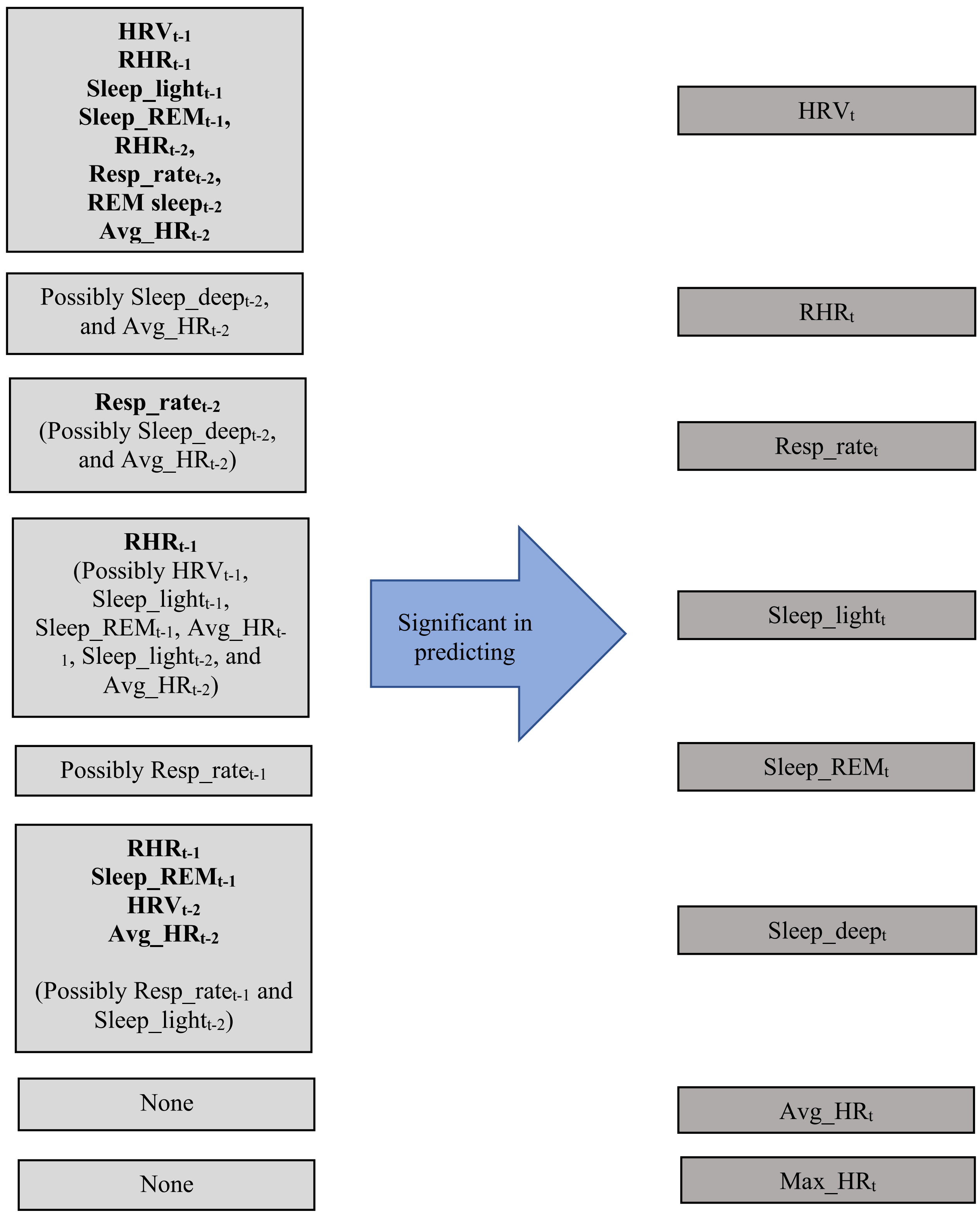


Figure 2: Summary of lagged variables significant in predicting another variable

Below is the VAR(2) model :

$$
\begin{bmatrix} HRV_t \\ RHR_t \\ Resp\_rate_t \\ Sleep\_light_t \\ Sleep\_REM_t \\ Sleep\_deep_t \\ Avg\_HR_t \\ Max\_HR_t \end{bmatrix}
= \begin{bmatrix}
0.6382 & -0.082 & -0.004 & 0.023 & 0.0147 & 0.014 & -0.051 & 0.384 \\
5.98 & -0.62 & -0.089 & 0.165 & 0.085 & 0.131 & -0.36 & 2.48 \\
13.034 & -1.529 & 0.0726 & 0.317 & 0.689 & 0.482 & 1.03 & 7.24 \\
22.513 & -1.553 & -0.326 & 0.895 & 0.225 & 0.175 & 2.65 & 15.88 \\
42.098 & -4.264 & -0.785 & 1.006 & 0.703 & 0.884 & 1.084 & 31.15 \\
-28.41 & 3.233 & -0.211 & -0.884 & -0.575 & -0.452 & 2.31 & -26.67 \\
2.6899 & 0.0617 & -0.017 & 0.118 & 0.068 & 0.035 & 0.34 & 2.98 \\
0.453 & -0.0352 & 0.002 & -0.004 & 0.003 & 0.009 & 0.07 & 0.371
\end{bmatrix}
\begin{bmatrix} HRV_{t-1} \\ RHR_{t-1} \\ Resp\_rate_{t-1} \\ Sleep\_light_{t-1} \\ Sleep\_REM_{t-1} \\ Sleep\_deep_{t-1} \\ Avg\_HR_{t-1} \\ Max\_HR_{t-1} \end{bmatrix}
+ \begin{bmatrix}
-0.4195 & -0.014 & -0.001 & 0.003 & -0.013 & -0.015 & 0.102 & 0.288 \\
2.952 & -0.098 & -0.044 & -0.022 & -0.014 & -0.006 & 0.2 & 1.91 \\
-43.42 & 2.769 & 0.77 & -0.259 & -0.51 & -0.41 & 3.48 & -9.11 \\
-13.717 & 2.854 & 0.095 & -1.077 & -0.366 & -0.706 & 3.32 & -4.49 \\
25.799 & -1.236 & 0.128 & -0.263 & -0.268 & 0.014 & -3.96 & 9.85 \\
-7.817 & 6.547 & 1.108 & -0.161 & -0.013 & 0.06 & -6.76 & -29.78 \\
-5.122 & 0.931 & 0.145 & -0.194 & -0.1 & -0.153 & -0.171 & -4.85 \\
0.338 & 0.0547 & 0.009 & -0.008 & -0.015 & 0.003 & -0.164 & -0.404
\end{bmatrix}
\begin{bmatrix} HRV_{t-2} \\ RHR_{t-2} \\ Resp\_rate_{t-2} \\ Sleep\_light_{t-2} \\ Sleep\_REM_{t-2} \\ Sleep\_deep_{t-2} \\ Avg\_HR_{t-2} \\ Max\_HR_{t-2} \end{bmatrix}
+ \begin{bmatrix} w_{HRV,t} \\ w_{RHR,t} \\ w_{Resp\_rate,t} \\ w_{Sleep\_light,t} \\ w_{Sleep\_REM,t} \\ w_{Sleep\_deep,t} \\ w_{Avg\_HR,t} \\ w_{Max\_HR,t} \end{bmatrix}
$$

Where the general model for the VAR(2) model is:

$$\tilde{x}_t = \Phi_1 * \tilde{x}_{t-1} + \Phi_2 * \tilde{x}_{t-2} + \tilde{w}_t$$

DISCUSSION OF RESULTS AND SUMMARY

In order to improve this analysis, I would recommend that we collect more data over time and test the two monitors over a variety of aerobic and resistance training sessions at different intensities and lengths of time. A limitation of the second part of the analysis is that there were many missing data when the monitor battery died and was not able to collect data. In part one of the analysis, I also was not able to easily download all the data and I had to manipulate both datasets to make them the same format in order to be able to compare them. Therefore, another improvement for this analysis could be getting permission from the company to access the raw data instead of taking them from the session graph on the website. This could also mean that I would be able to have more than 6 heart rate observations per minute, which would improve the models. I would also like to compare more than just two wrist worn monitors, with more resources, I would have liked to test multiple monitors on the market.

The goals of this data analysis were achieved. First, I was able to determine that over the course of an aerobic session, the Polar and WHOOP gave a similar heart rate time series model. Both monitors found an ARIMA(0,1,1) model, however the WHOOP was able to detect a drift in addition of the ARIMA model, while the Polar was not. We found that the MA parameters of the two models were not statistically significant. In the second part of the analysis, I found many interesting relationships in the VAR(2) model across eight different daily variables, the most interesting being that today's HRV value is predicted well by yesterday's HRV, RHR, duration of REM sleep and duration of light sleep, as well as the RHR, respiratory rate, duration of REM sleep and Average HR values from two days ago.

This analysis is gives one method in comparing the accuracy of different monitors and one method in comparing the physiological relationships between the daily variables these monitors are capable of collecting. Further research must be done to determine the accuracy of these wrist worn monitors to the gold standard of heart rate monitoring, and determine how these monitors could be utilized to improve athletic performance and possibly the quality of care for cardiac patients.

References

Nelson, B. W., & Allen, N. B. (2019). Accuracy of Consumer Wearable Heart Rate Measurement During an Ecologically Valid 24-Hour Period: Intraindividual Validation Study. *JMIR mHealth and uHealth*, *7*(3), e10828. doi:10.2196/10828

Pasadyn, S. R., Soudan, M., Gillinov, M., Houghtaling, P., Phelan, D., Gillinov, N., … Desai, M. Y. (2019). Accuracy of commercially available heart rate monitors in athletes: a prospective study. *Cardiovascular diagnosis and therapy*, *9*(4), 379–385. doi:10.21037/cdt.2019.06.05

Polar Ignite: High-quality fitness watch with GPS. (n.d.). Retrieved from https://www.polar.com/us-en/ignite.

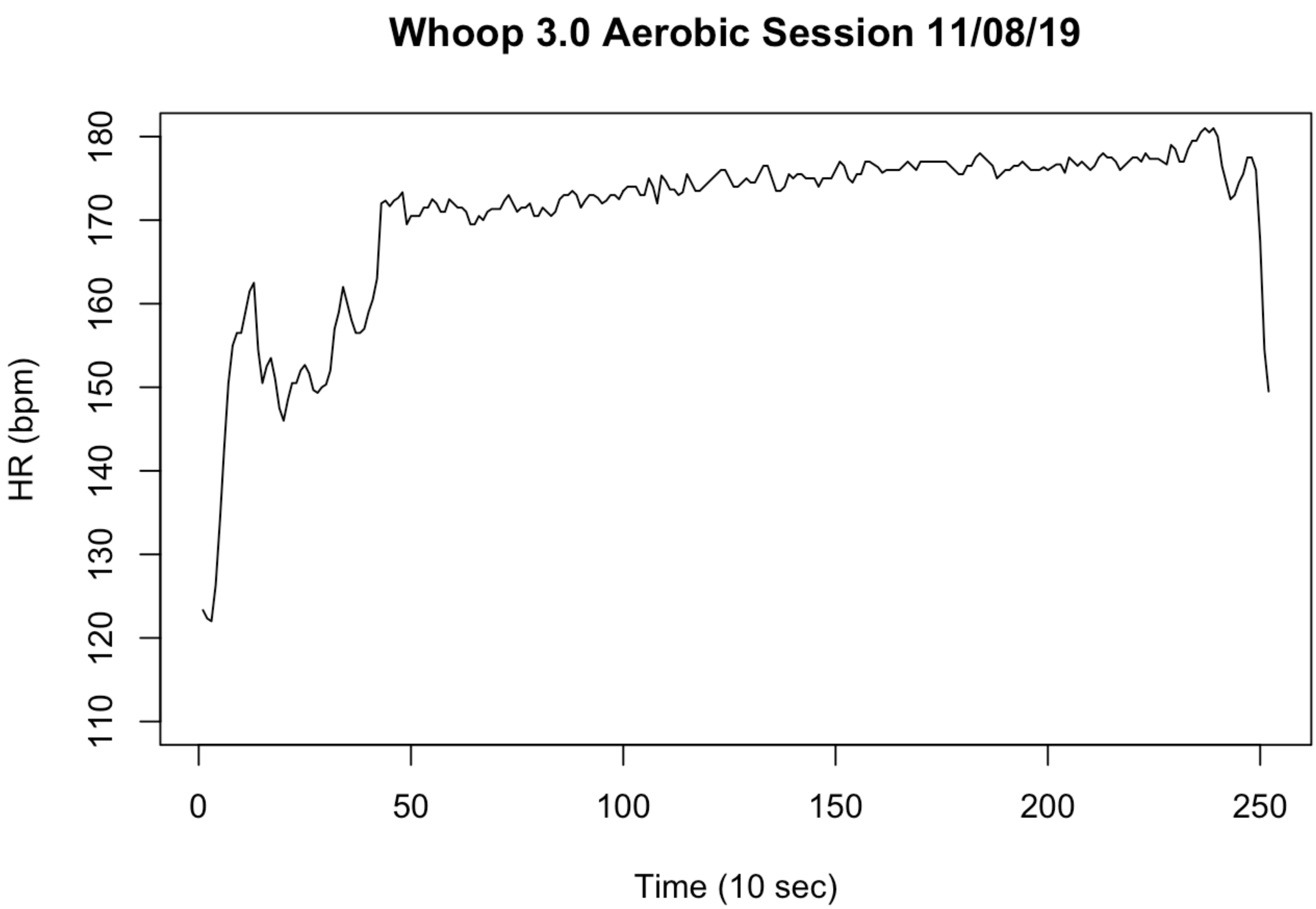
WHOOP Experience - Recovery, strain and sleep metrics optimize training. (n.d.). Retrieved from https://www.whoop.com/experience/#recovery.

R Code

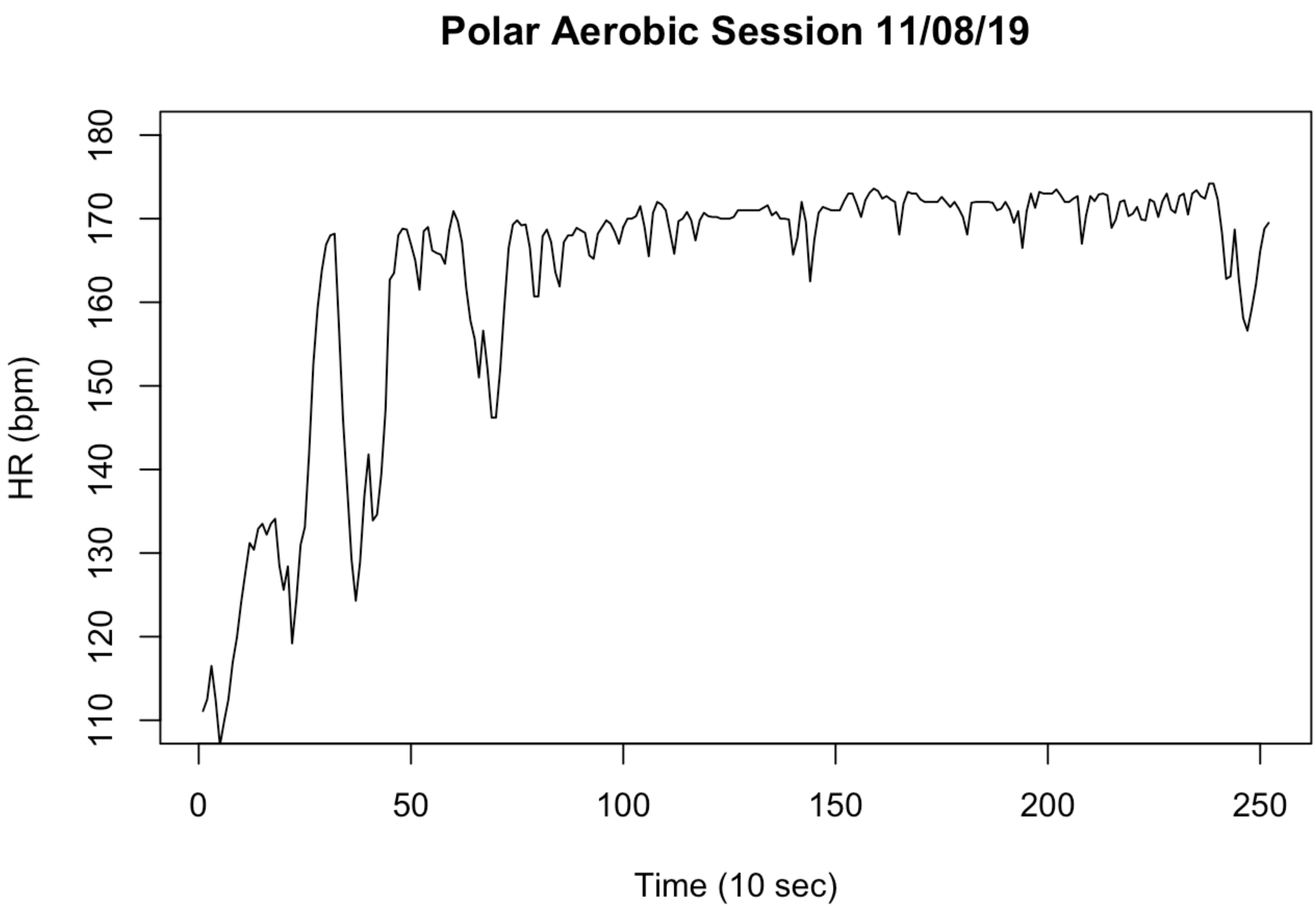
Appendix File PART A- Comparing HR capabilities of the Polar Ignite and the Whoop 3.0 over one aerobic session (10k erg, 40:45.8, 2:02.2/500m)

1. Create time series plot for WHOOP and Polar datasets

```
whoop_1108_endurancesession_data <- read.csv("~/Downloads/whoop_1108_endurancesession_data.csv")
polar_1108_endurancesession_data <- read.csv("~/Downloads/polar_1108_endurancesession_data.csv")
ts.plot(whoop_1108_endurancesession_data[2], ylab="HR (bpm)", xlab="Time (10 sec)", main="Whoop 3.0 Aerobic Session 11/08/19", ylim=c(110,180))
```




```
ts.plot(polar_1108_endurancesession_data[2], ylab="HR (bpm)", main="Polar Aerobic Session 11/08/19", xlab="Time (10 sec)", ylim=c(110,180))
```



2. Prepare data- split into calibration and holdout sets for WHOOP and Polar, turn HR values into time series

```
## WHOOP
whoop_1108_endurancesession_data_CALIBRATION <- whoop_1108_endurancesession_data[-(20
3:252), ]
whoop_1108_endurancesession_data_HOLDOUT <- whoop_1108_endurancesession_data[-(1:202)
, ]
whoop_1108_endurancesession_data_CALIBRATION <- ts(whoop_1108_endurancesession_data_C
ALIBRATION[2])
whoop_1108_endurancesession_data_HOLDOUT <- ts(whoop_1108_endurancesession_data_HOLDO
UT[2])

## Polar
polar_1108_endurancesession_data_CALIBRATION <- polar_1108_endurancesession_data[-(20
3:252), ]
polar_1108_endurancesession_data_HOLDOUT <- polar_1108_endurancesession_data[-c(1:202
), ]
polar_1108_endurancesession_data_CALIBRATION <- ts(polar_1108_endurancesession_data_C
ALIBRATION[2])
polar_1108_endurancesession_data_HOLDOUT <- ts(polar_1108_endurancesession_data_HOLDO
UT[2])
```

3. Model Identification

```
library(tidyverse)
```

```
## ── Attaching packages ───────────────────────────────────────────────
───────── tidyverse 1.2.1 ──
```

```
## ✔ ggplot2 3.1.1     ✔ purrr   0.3.0
## ✔ tibble  2.0.1     ✔ dplyr   0.8.3
## ✔ tidyr   0.8.2     ✔ stringr 1.3.1
## ✔ readr   1.3.1     ✔ forcats 0.4.0
```

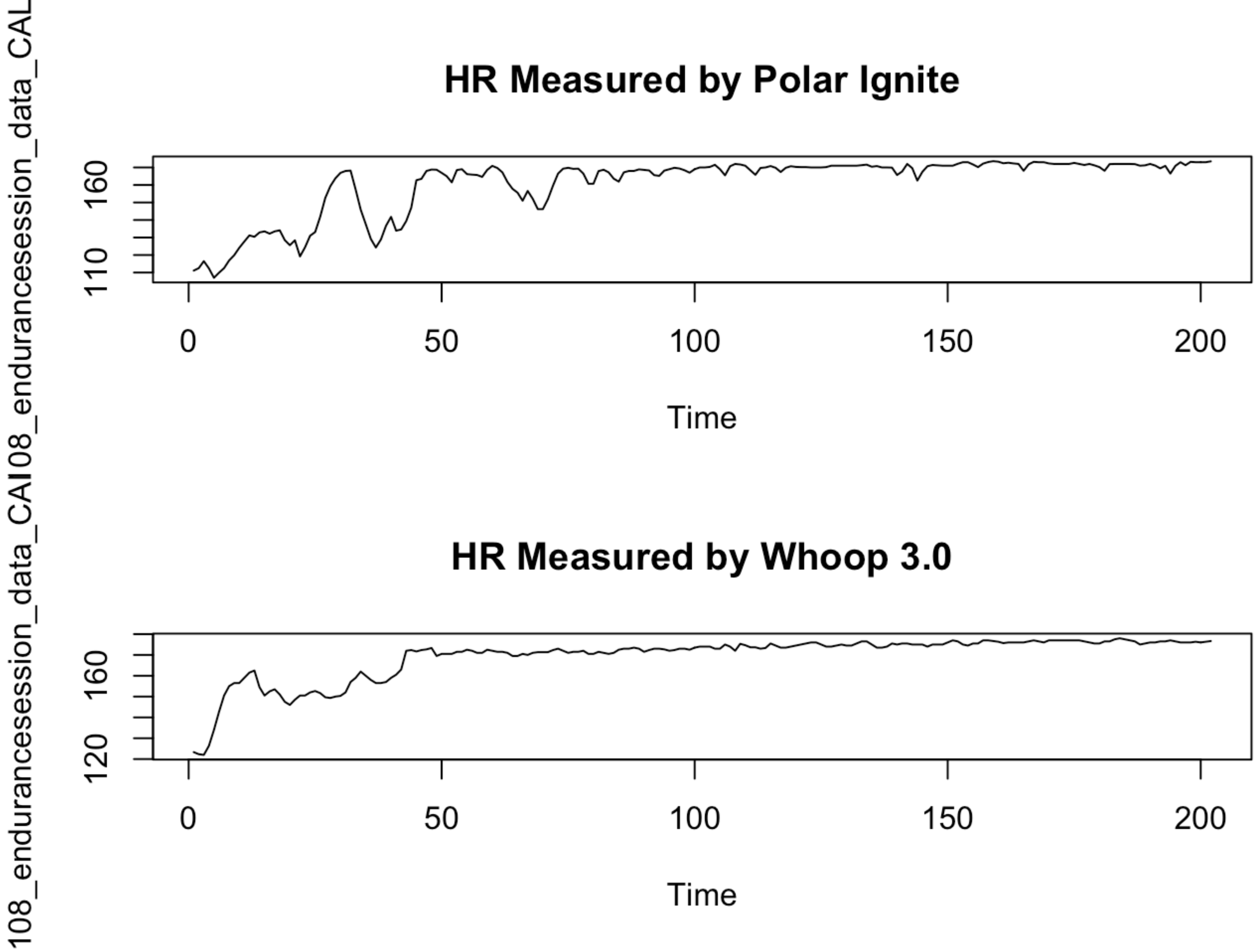
```
## ── Conflicts ────────────────────────────────────────────────────────
── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(forecast)
library(fpp2)
```

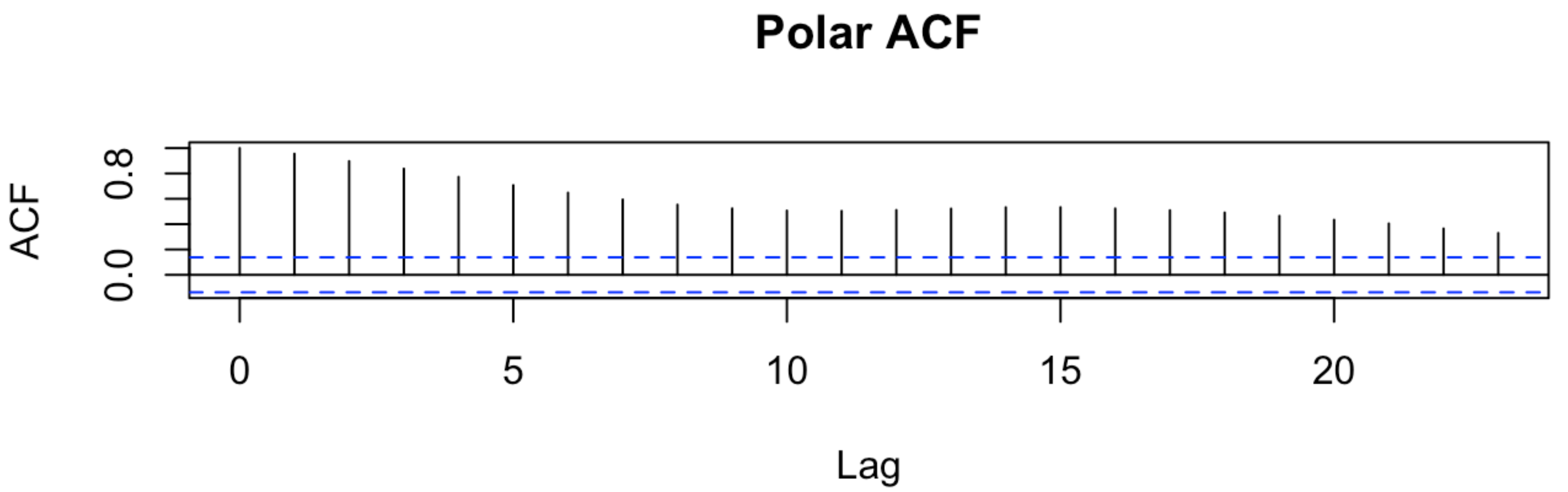
```
## Loading required package: fma
```

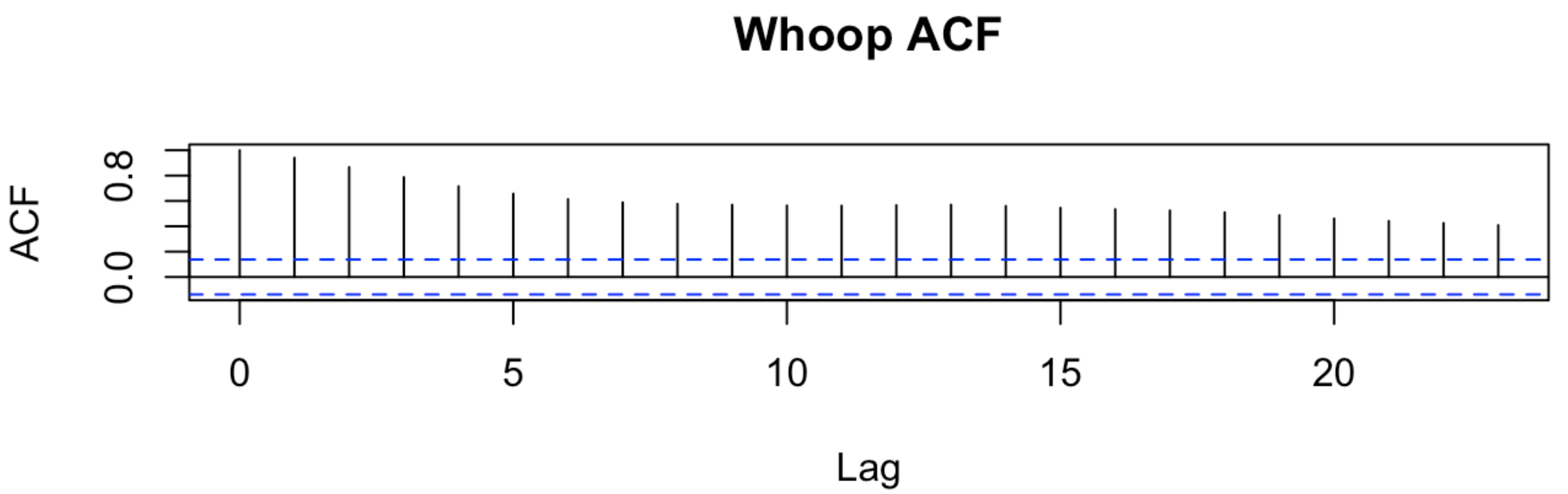
```
## Loading required package: expsmooth
```

```
par(mfrow=c(2, 1))
ts.plot(polar_1108_endurancesession_data_CALIBRATION, main="HR Measured by Polar Igni
te")
ts.plot(whoop_1108_endurancesession_data_CALIBRATION, main="HR Measured by Whoop 3.0"
)
```

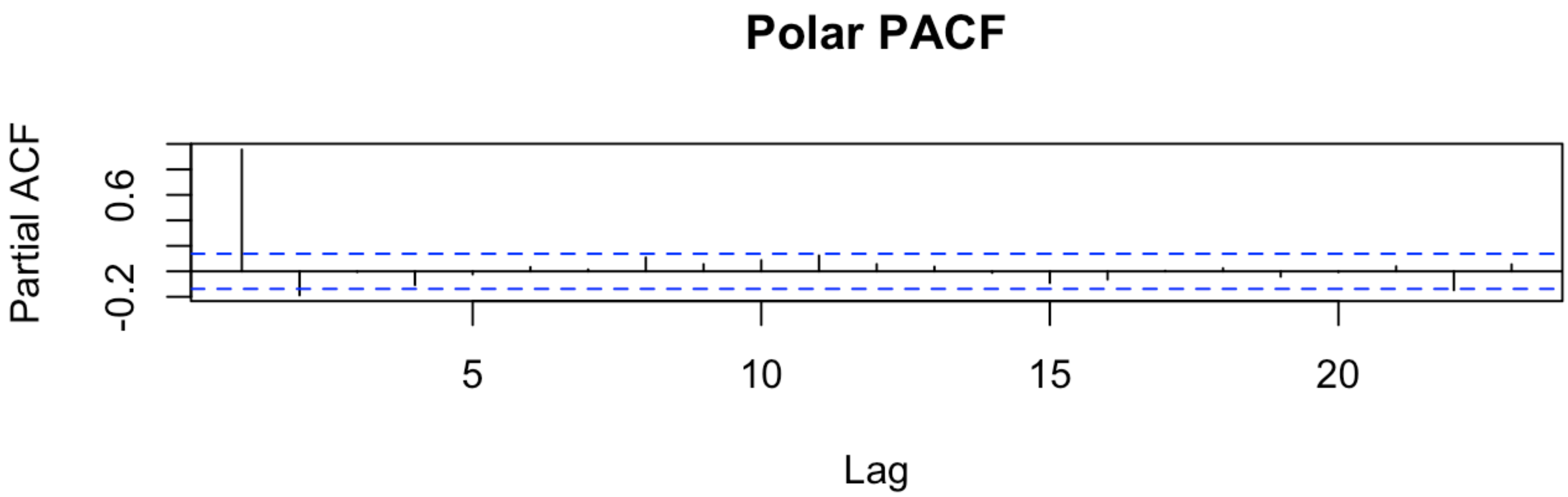


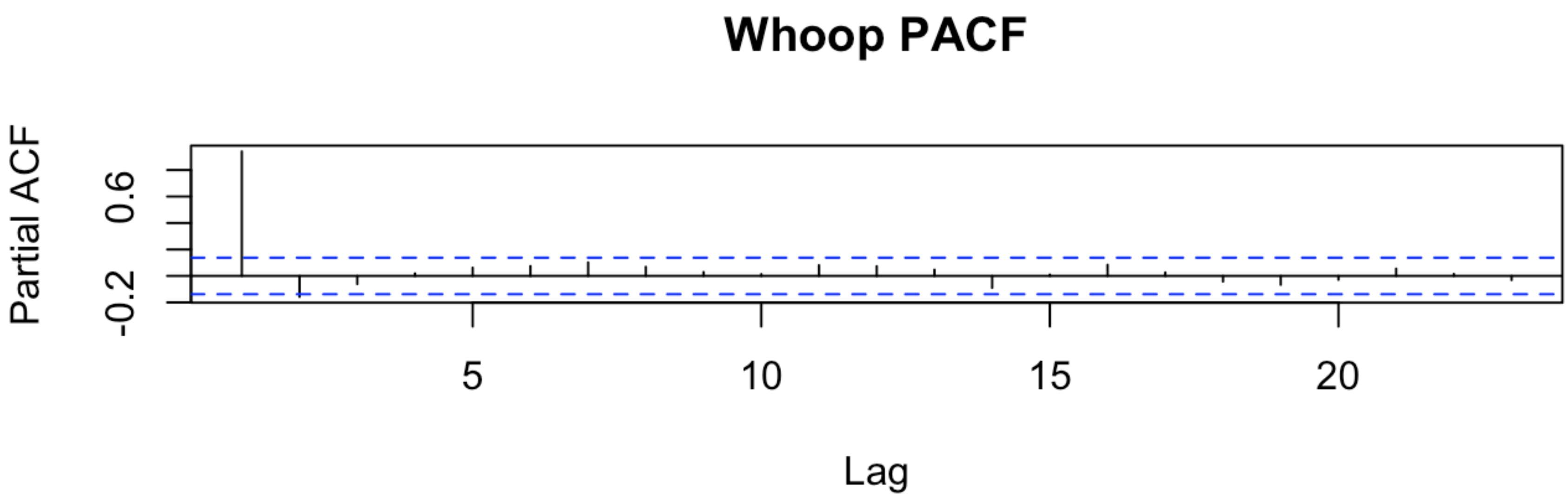

```
par(mfrow=c(2,1))
acf(polar_1108_endurancesession_data_CALIBRATION, main="Polar ACF")
acf(whoop_1108_endurancesession_data_CALIBRATION, main="Whoop ACF")
```






```
par(mfrow=c(2,1))
pacf(polar_1108_endurancesession_data_CALIBRATION, main="Polar PACF")
pacf(whoop_1108_endurancesession_data_CALIBRATION, main="Whoop PACF")
```






```
##POLAR-- from pacf looks like lag 2 is significant, AR(2)
polar.fit1<-auto.arima(polar_1108_endurancesession_data_CALIBRATION)
summary(polar.fit1)
```

```
## Series: polar_1108_endurancesession_data_CALIBRATION
## ARIMA(0,1,1)
##
## Coefficients:
##          ma1
##       0.3912
## s.e.  0.0669
##
## sigma^2 estimated as 10.77:  log likelihood=-523.68
## AIC=1051.37   AICc=1051.43   BIC=1057.97
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE    MAPE     MASE
## Training set 0.2231089 3.265977 2.180326 0.1443099 1.44145 0.936422
##                     ACF1
## Training set 0.002590945
```

```r
polar.fit2<- arima(polar_1108_endurancesession_data_CALIBRATION, order=c(2,0,0))
summary(polar.fit2)
```

```
##
## Call:
## arima(x = polar_1108_endurancesession_data_CALIBRATION, order = c(2, 0, 0))
##
## Coefficients:
##          ar1      ar2  intercept
##       1.3572  -0.3778   156.7373
## s.e.  0.0650   0.0662    10.0796
##
## sigma^2 estimated as 10.66:  log likelihood = -527.54,  aic = 1063.08
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.2436157 3.264746 2.177656 0.1232946 1.444731 0.9352752
##                    ACF1
## Training set -0.0110953
```

```r
##but from the auto.arima function, we can see that ARIMA(0,1,1) model is more approp
riate

## WHOOP--from pacf looks like lag 2 is significant, AR(2)
whoop.fit1<-auto.arima(whoop_1108_endurancesession_data_CALIBRATION)
whoop.fit2 <- arima(whoop_1108_endurancesession_data_CALIBRATION, order=c(2,0,0))
```

```
## Warning in arima(whoop_1108_endurancesession_data_CALIBRATION, order =
## c(2, : possible convergence problem: optim gave code = 1
```

```r
summary(whoop.fit1)
```

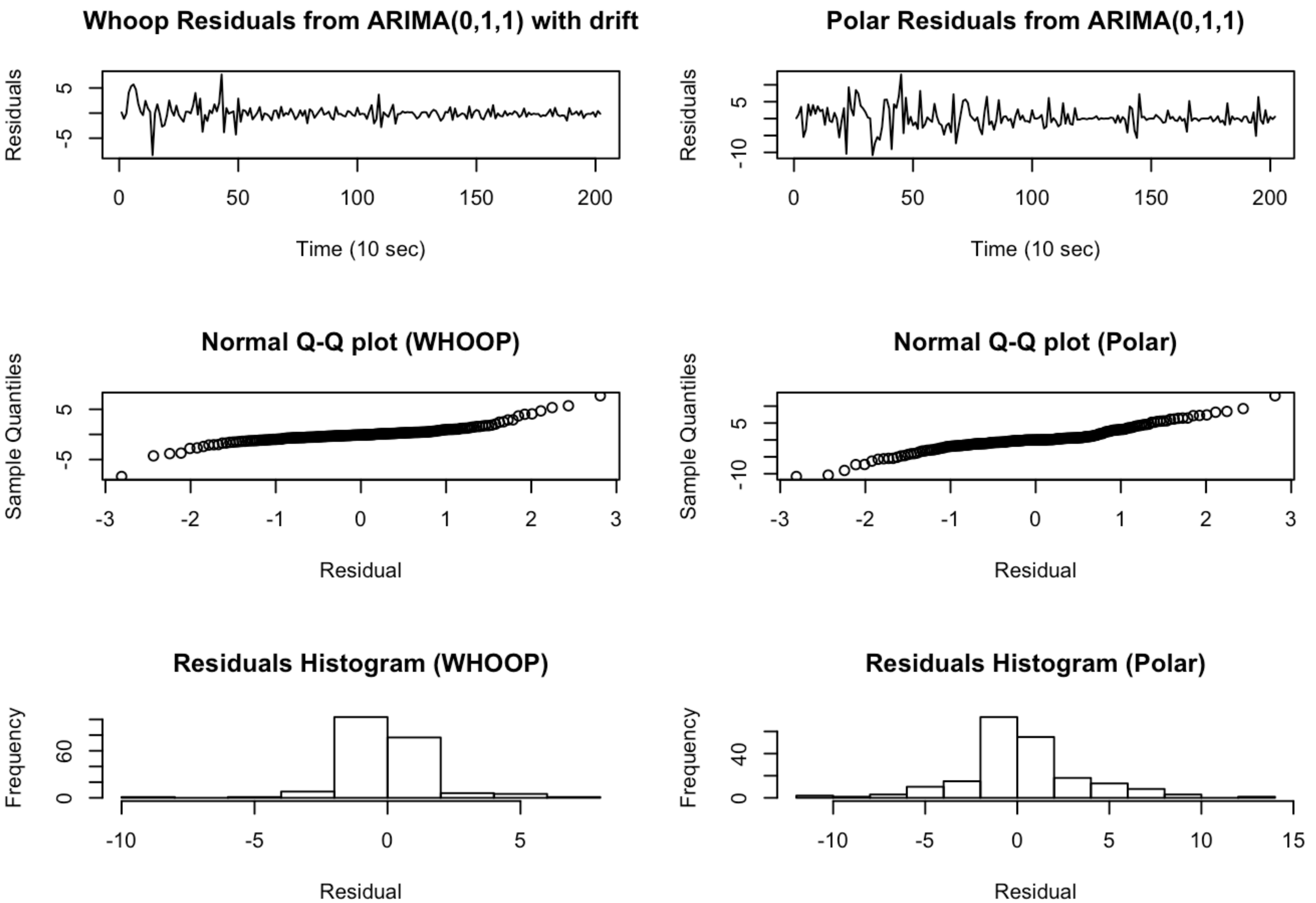
```
## Series: whoop_1108_endurancesession_data_CALIBRATION
## ARIMA(0,1,1) with drift
##
## Coefficients:
##          ma1   drift
##       0.5056  0.2637
## s.e.  0.0595  0.1628
##
## sigma^2 estimated as 2.38:  log likelihood=-371.5
## AIC=749.01   AICc=749.13   BIC=758.92
##
## Training set error measures:
##                       ME     RMSE       MAE        MPE     MAPE      MASE
## Training set 0.001669503 1.531327 0.9341931 0.01295204 0.571342 0.9357449
##                     ACF1
## Training set 0.04215014
```

```
summary(whoop.fit2)
```

```
##
## Call:
## arima(x = whoop_1108_endurancesession_data_CALIBRATION, order = c(2, 0, 0))
##
## Coefficients:
##          ar1      ar2  intercept
##       1.4799  -0.4866   161.6894
## s.e.  0.0619   0.0631    13.0332
##
## sigma^2 estimated as 2.404:  log likelihood = -377.85,  aic = 763.7
##
## Training set error measures:
##                     ME     RMSE       MAE       MPE      MAPE      MASE
## Training set 0.1721025 1.550568 0.9748338 0.1067633 0.5986983 0.9764531
##                     ACF1
## Training set 0.03665674
```

```
##but from auto.arima function, we can see that the ARIMA(0,1,1) model WITH DRIFT is
more appropriate
```

```
## PART 3-- MODEL ACCURACY
##whoop
par(mfrow=c(3, 2))
plot(whoop.fit1$residuals, xlab="Time (10 sec)", ylab="Residuals", main="Whoop Residu
als from ARIMA(0,1,1) with drift")
plot(polar.fit1$residuals, ylab="Residuals", xlab="Time (10 sec)", main="Polar Residu
als from ARIMA(0,1,1)")
qqnorm(whoop.fit1$residuals,main="Normal Q-Q plot (WHOOP)",xlab="Residual")
qqnorm(polar.fit1$residuals,main="Normal Q-Q plot (Polar)",xlab="Residual")
hist(whoop.fit1$residuals,main="Residuals Histogram (WHOOP)",xlab="Residual")
hist(polar.fit1$residuals,main="Residuals Histogram (Polar)",xlab="Residual")
```




```
####################
whoop.fit1.resid <- whoop.fit1$residuals

##Shapiro-Wilk test for normality
shapiro.test(whoop.fit1.resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  whoop.fit1.resid
## W = 0.84015, p-value = 1.229e-13
```

```
##  H0: normal
## HA : not normal
##conclusion: residuals not normally distributed

# Ljung-Box Portmanteau test for Model Adequacy
whoop.box <- Box.test(whoop.fit1.resid,type="Ljung")
whoop.box
```

```
##
##  Box-Ljung test
##
## data:  whoop.fit1.resid
## X-squared = 0.36424, df = 1, p-value = 0.5462
```

```
##  H0: p1=p2=0 (auto correlations are equal to 0, model does not show lack of fit)
##  HA: at least one is not equal to 0
##  p=.5462 so model is adequate

polar.fit1.resid <- polar.fit1$residuals

##Shapiro-Wilk test for normality
shapiro.test(polar.fit1.resid)
```

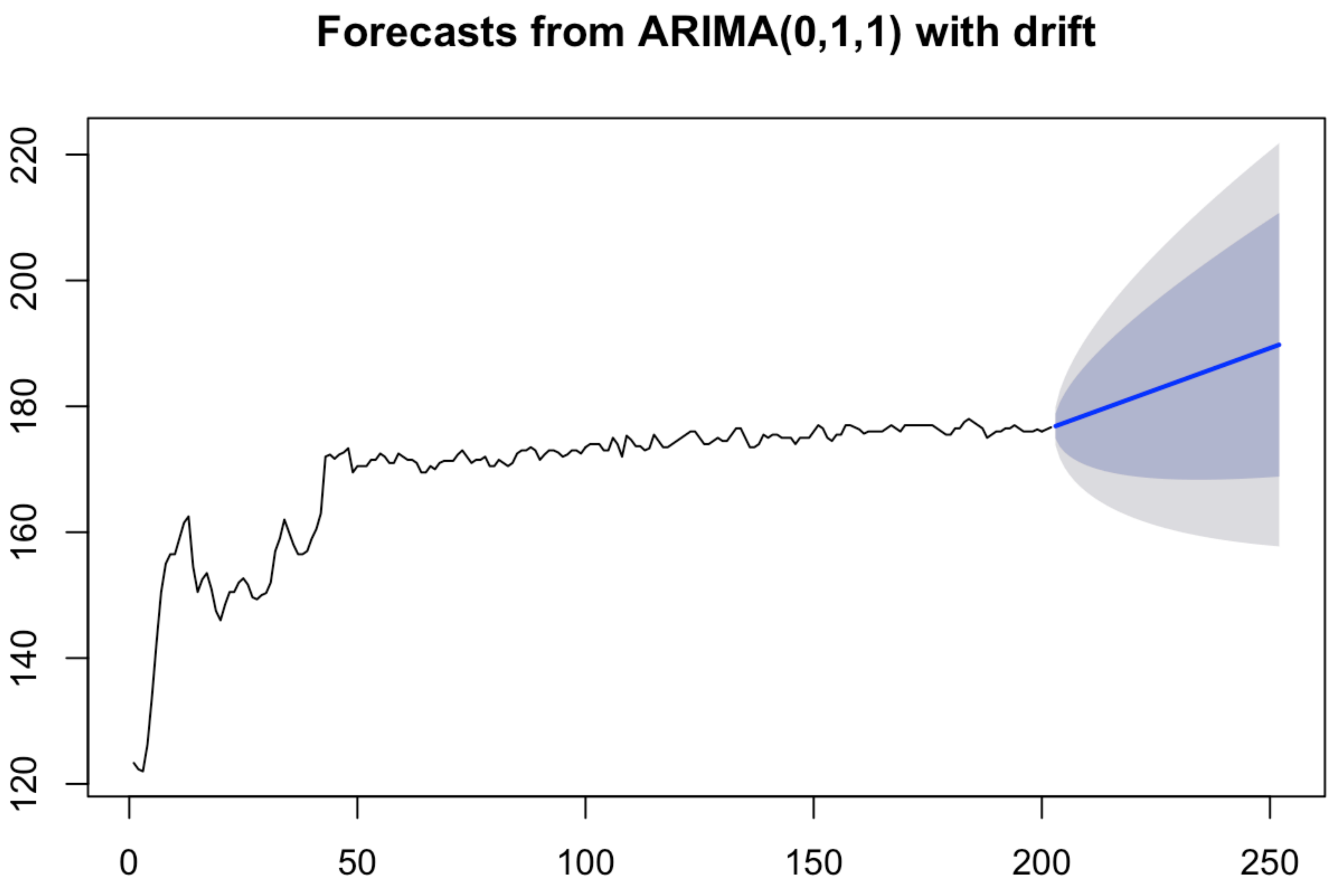
```
##
##  Shapiro-Wilk normality test
##
## data:  polar.fit1.resid
## W = 0.94158, p-value = 2.808e-07
```

```
##  H0: normal
## HA : not normal
##conclusion: residuals not normally distributed

# Ljung-Box Portmanteau test for Model Adequacy
whoop.box <- Box.test(polar.fit1.resid,type="Ljung")
whoop.box
```

```
##
##  Box-Ljung test
##
## data:  polar.fit1.resid
## X-squared = 0.0013763, df = 1, p-value = 0.9704
```

```
##  H0: p1=p2=0 (auto correlations are equal to 0, model does not show lack of fit)
##  HA: at least one is not equal to 0
##  p=.9704 so model is adequate
```

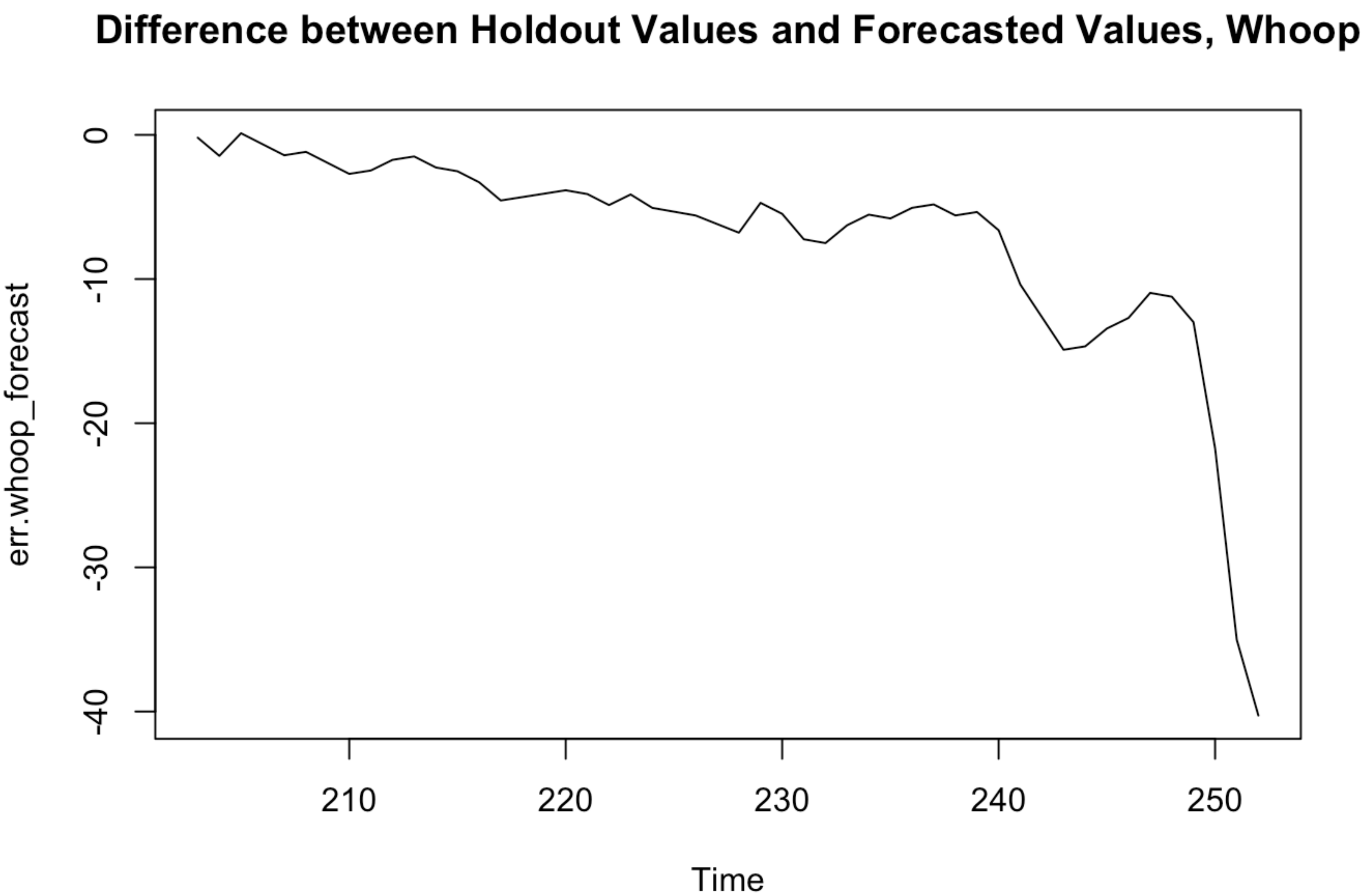
```
# Forecast Holdout Observations
whoop_forecast <- forecast(whoop.fit1, h=50)
plot(whoop_forecast)
```




```
err.whoop_forecast <- as.numeric(whoop_1108_endurancesession_data_HOLDOUT)-whoop_fore
cast$mean
err.whoop_forecast
```

```
## Time Series:
## Start = 203
## End = 252
## Frequency = 1
##  [1]  -0.1928107  -1.4564785   0.1131870  -0.6504808  -1.4141486
##  [6]  -1.1778164  -1.9414842  -2.7051521  -2.4688199  -1.7324877
## [11]  -1.4961555  -2.2598233  -2.5234911  -3.2871589  -4.5508267
## [16]  -4.3144945  -4.0781623  -3.8418301  -4.1054979  -4.8691657
## [21]  -4.1328336  -5.0631681  -5.3268359  -5.5905037  -6.1875048
## [26]  -6.7845059  -4.7148404  -5.4785082  -7.2421760  -7.5058438
## [31]  -6.2695116  -5.5331794  -5.7968473  -5.0605151  -4.8241829
## [36]  -5.5878507  -5.3515185  -6.6151863 -10.3788541 -12.6425219
## [41] -14.9061897 -14.6698575 -13.4335253 -12.6971931 -10.9608609
## [46] -11.2245288 -12.9881966 -21.7518644 -35.0155322 -40.2792000
```

```
plot(err.whoop_forecast, main="Difference between Holdout Values and Forecasted Value
s, Whoop")
```



```
# Forecast Evaluation Criteria based on Holdout Prediction

me.err.whoop=mean(err.whoop_forecast)
mpe.err.whoop=100*(mean(err.whoop_forecast/as.numeric(whoop_1108_endurancesession_dat
a_HOLDOUT)))
mse.err.whoop=sum(err.whoop_forecast**2)/length(err.whoop_forecast)
mae.err.whoop=mean(abs(err.whoop_forecast))
mape.err.whoop=100*(mean(abs((err.whoop_forecast)/as.numeric(whoop_1108_endurancesess
ion_data_HOLDOUT))))
me.err.whoop
```

```
## [1] -7.259339
```

```
mpe.err.whoop
```

```
## [1] -4.261032
```

```
mse.err.whoop
```

```
## [1] 110.9757
```

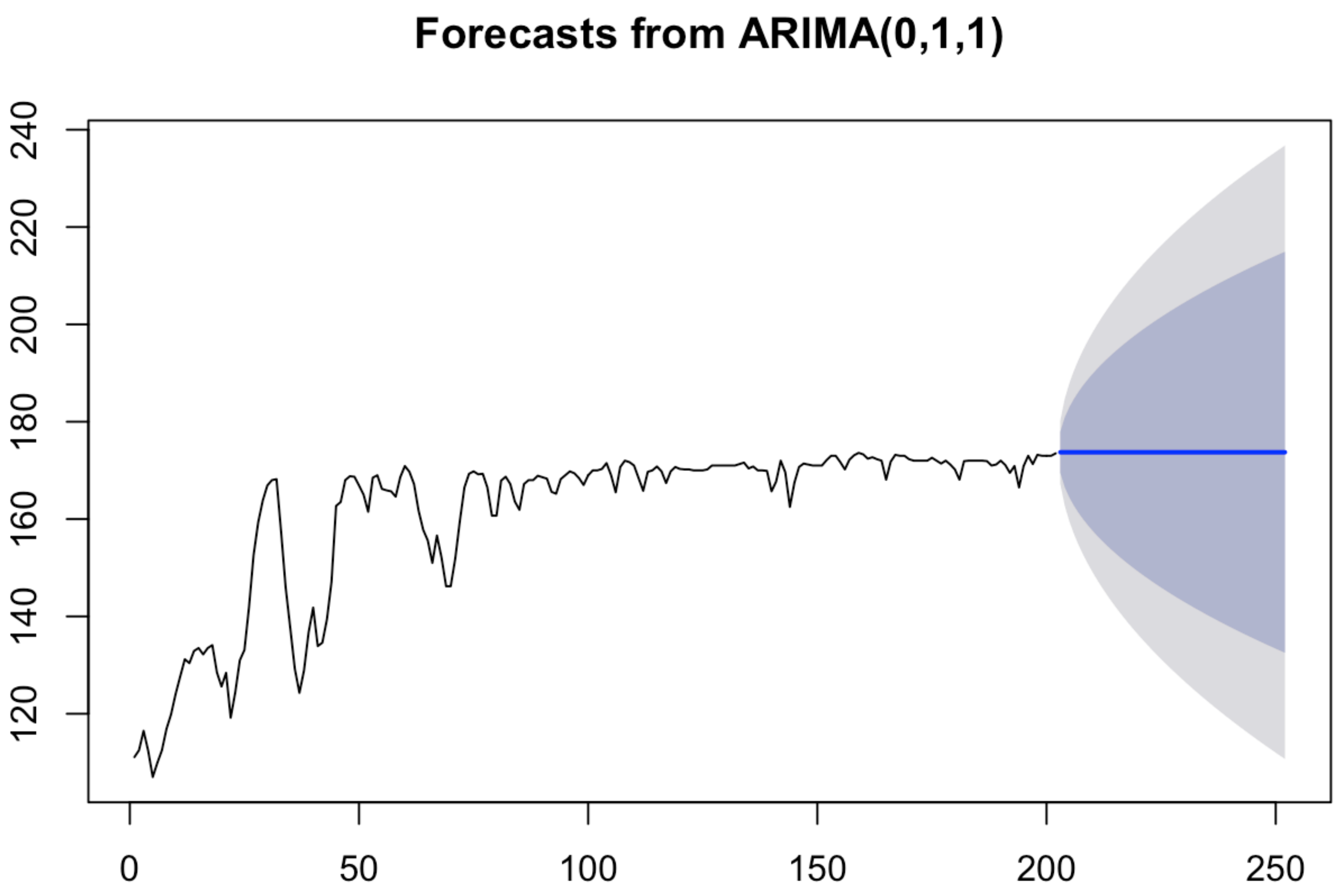
```
mae.err.whoop
```

```
## [1] 7.263866
```

```
mape.err.whoop
```

```
## [1] 4.263583
```

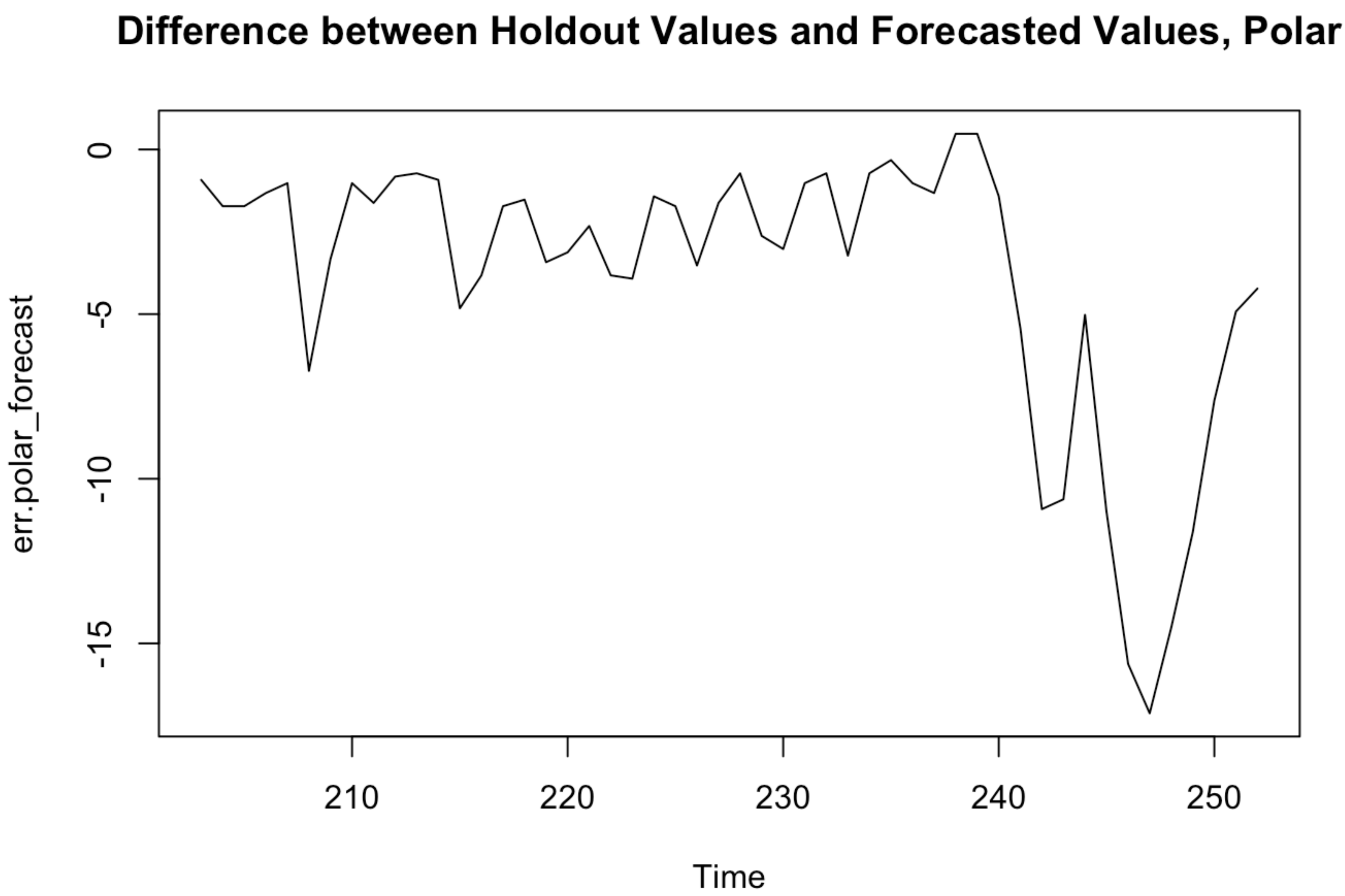
```
##polar
# Forecast Holdout Observations
polar_forecast <- forecast(polar.fit1, h=50)
plot(polar_forecast)
```




```
err.polar_forecast <- as.numeric(polar_1108_endurancesession_data_HOLDOUT)-polar_fore
cast$mean
err.polar_forecast
```

```
## Time Series:
## Start = 203
## End = 252
## Frequency = 1
##  [1]  -0.9232505  -1.7232505  -1.7232505  -1.3232505  -1.0232505
##  [6]  -6.7232505  -3.3232505  -1.0232505  -1.6232505  -0.8232505
## [11]  -0.7232505  -0.9232505  -4.8232505  -3.8232505  -1.7232505
## [16]  -1.5232505  -3.4232505  -3.1232505  -2.3232505  -3.8232505
## [21]  -3.9232505  -1.4232505  -1.7232505  -3.5232505  -1.6232505
## [26]  -0.7232505  -2.6232505  -3.0232505  -1.0232505  -0.7232505
## [31]  -3.2232505  -0.7232505  -0.3232505  -1.0232505  -1.3232505
## [36]   0.4767495   0.4767495  -1.4232505  -5.4232505 -10.9232505
## [41] -10.6232505  -5.0232505 -11.0232505 -15.6232505 -17.1232505
## [46] -14.5232505 -11.6232505  -7.6232505  -4.9232505  -4.2232505
```

```
plot(err.polar_forecast, main="Difference between Holdout Values and Forecasted Value
s, Polar")
```




```
# Forecast Evaluation Criteria based on Holdout Prediction

me.err.polar=mean(err.polar_forecast)
mpe.err.polar=100*(mean(err.polar_forecast/as.numeric(polar_1108_endurancesession_dat
a_HOLDOUT)))
mse.err.polar=sum(err.polar_forecast**2)/length(err.polar_forecast)
mae.err.polar=mean(abs(err.polar_forecast))
mape.err.polar=100*(mean(abs((err.polar_forecast)/as.numeric(polar_1108_endurancesess
ion_data_HOLDOUT))))
me.err.polar
```

```
## [1] -3.851251
```

```
mpe.err.polar
```

```
## [1] -2.332021
```

```
mse.err.polar
```

```
## [1] 32.35815
```

```
mae.err.polar
```

```
## [1] 3.88939
```

```
mape.err.polar
```

```
## [1] 2.353915
```

```
## 95% CI for theta1 (ma1 coeff) of polar model
polar.ma.CI_U <- polar.fit1$coef + 1.96*0.0669
polar.ma.CI_L <- polar.fit1$coef - 1.96*0.0669
polar.ma.CI_L
```

```
##       ma1
## 0.2601164
```

```
polar.ma.CI_U
```

```
##       ma1
## 0.5223644
```

```
## 95% CI for theta1 (ma1 coeff) of whoop model
whoop.ma.CI_U <- whoop.fit1$coef + 1.96*0.0595
whoop.ma.CI_L <- whoop.fit1$coef - 1.96*0.0595
whoop.ma.CI_L
```

```
##       ma1     drift
## 0.3890294 0.1470478
```

```
whoop.ma.CI_U
```

```
##       ma1     drift
## 0.6222694 0.3802878
```

```
##95% CI of theta1 of polar model-- (0.260, 0.522)
##95% CI of theta1 of whoop model-- (0.389, 0.622)

##t test to compare theta1 values for whoop and polar
##polar sd--
sd.polar <- .0669*sqrt(202)
sd.whoop <- 0.0595*sqrt(202)
se.pooled <- sqrt(((sd.polar^2)/202)+((sd.whoop^2)/202))
t <- (.3912404-.5056)/se.pooled
t
```

```
## [1] -1.277314
```

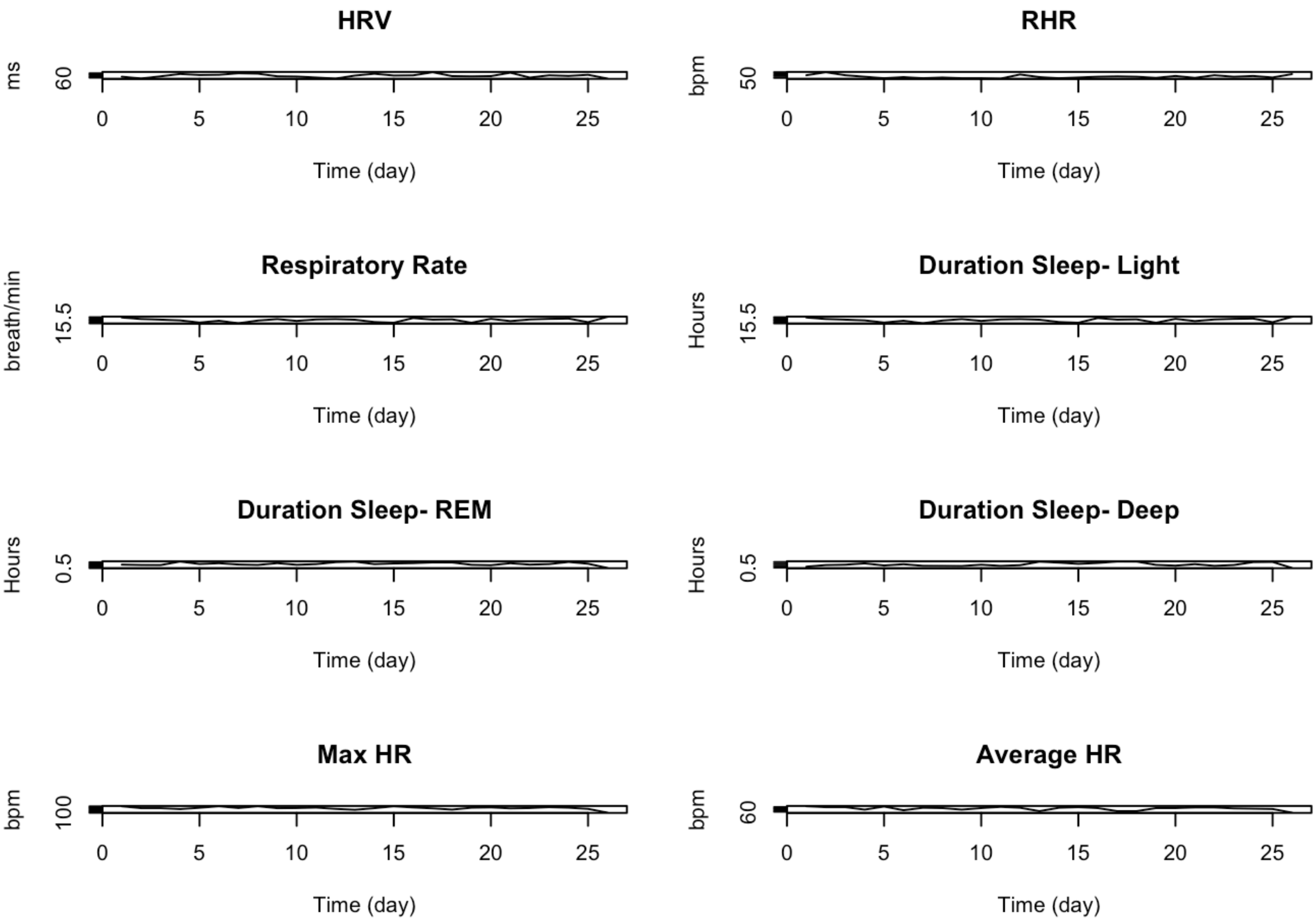
```
pt(t, (202+202-2))
```

```
## [1] 0.1011144
```

```
##H0: theta1(polar)=theta1(whoop)
##HA: theta1(polar) not equal to theta1(whoop)
##    p-value=0.2834-- fail to reject H0

##########CONCLUSION-- no difference between the model parameters for whoop and polar
```

Part B- analyze whoop data over month, 8 varaibles, determine how they are related and over what lags they are related

```
##Insert data- all variables, all observations 09/26/2019-11/30/2019, notice NAs
whoopdata_3.0 <- read.csv("~/Downloads/whoopdata-3/whoopdata_3.0.csv")

### Split dataset into 2 sets (keeping all other varaibles the same):
###     training/model set is 09/26/19-10/21/19 (because this was the larger run of d
ata)
###     test set is 10/31/19-11/14/19 (this will be used to test the model found from
the training set)
whoopdata_0926_1021<- whoopdata_3.0[c(1:26), ]
whoopdata_1031_1114<- whoopdata_3.0[c(36:50), ]

#plots
par(mfrow=c(4, 2))
whoop_hrv_ts <- ts(whoopdata_0926_1021$recovery_hrv)
ts.plot(whoop_hrv_ts, main="HRV", ylab="ms", xlab="Time (day)")
whoop_rhr_ts <- ts(whoopdata_0926_1021$recovery_rhr)
ts.plot(whoop_rhr_ts, main="RHR", ylab="bpm", xlab="Time (day)")
whoop_resp_rate_ts <- ts(whoopdata_0926_1021$sleep_respiratory_rate)
ts.plot(whoop_resp_rate_ts, main="Respiratory Rate", ylab="breath/min", xlab="Time (d
ay)")
whoop_sleep_light_ts <- ts(whoopdata_0926_1021$sleep_respiratory_rate)
ts.plot(whoop_sleep_light_ts,main="Duration Sleep- Light", ylab="Hours", xlab="Time (
day)")
whoop_sleep_rem_ts <- ts(whoopdata_0926_1021$sleep_duration_rem)
ts.plot(whoop_sleep_rem_ts, main="Duration Sleep- REM", ylab="Hours", xlab="Time (day
)")
whoop_sleep_deep_ts <- ts(whoopdata_0926_1021$sleep_duration_deep)
ts.plot(whoop_sleep_deep_ts, main="Duration Sleep- Deep", ylab="Hours", xlab="Time (d
ay)")
whoop_maxhr_ts <- ts(whoopdata_0926_1021$strain_maxhr)
ts.plot(whoop_maxhr_ts, main="Max HR", ylab="bpm", xlab="Time (day)")
whoop_avghr_ts <- ts(whoopdata_0926_1021$strain_avghr)
ts.plot(whoop_avghr_ts, main="Average HR", ylab="bpm", xlab="Time (day)")
```




```r
library(tidyverse)
library(forecast)
library(fpp2)

whoop_hrv_fit <- auto.arima(whoop_hrv_ts)
whoop_rhr_fit <- auto.arima(whoop_rhr_ts)
whoop_resp_rate_fit <- auto.arima(whoop_resp_rate_ts)
whoop_sleep_light_fit <- auto.arima(whoop_sleep_light_ts)
whoop_sleep_rem_fit <- auto.arima(whoop_sleep_rem_ts)
whoop_sleep_deep_fit <- auto.arima(whoop_sleep_rem_ts)
whoop_avghr_fit <- auto.arima(whoop_avghr_ts)
whoop_maxhr_fit <- auto.arima(whoop_maxhr_ts)

summary(whoop_hrv_fit)
```

```
## Series: whoop_hrv_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##          mean
##       87.7530
## s.e.   4.5909
##
## sigma^2 estimated as 569.9:  log likelihood=-118.87
## AIC=241.75   AICc=242.27   BIC=244.26
##
## Training set error measures:
##                         ME     RMSE      MAE       MPE     MAPE      MASE
## Training set -4.373238e-15 23.40918 18.67416 -8.421021 24.42607 0.7540655
##                   ACF1
## Training set 0.1100225
```

```
summary(whoop_rhr_fit)
```

```
## Series: whoop_rhr_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##          mean
##       53.1154
## s.e.   0.7907
##
## sigma^2 estimated as 16.91:  log likelihood=-73.14
## AIC=150.28   AICc=150.81   BIC=152.8
##
## Training set error measures:
##                        ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 2.695143e-12 4.031863 3.150888 -0.5402901 5.822519 0.7799227
##                   ACF1
## Training set 0.2150927
```

```
summary(whoop_sleep_light_fit)
```

```
## Series: whoop_sleep_light_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##          mean
##       16.5572
## s.e.   0.1060
##
## sigma^2 estimated as 0.3039:  log likelihood=-20.9
## AIC=45.8   AICc=46.32   BIC=48.31
##
## Training set error measures:
##                         ME      RMSE       MAE        MPE     MAPE
## Training set -7.378717e-15 0.5405595 0.4522831 -0.1077004 2.749396
##                   MASE        ACF1
## Training set 0.7213861 -0.08390482
```

```
summary(whoop_sleep_rem_fit)
```

```
## Series: whoop_sleep_rem_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##         mean
##       2.3212
## s.e.  0.1141
##
## sigma^2 estimated as 0.3518:  log likelihood=-22.8
## AIC=49.6   AICc=50.12   BIC=52.12
##
## Training set error measures:
##                         ME      RMSE       MAE      MPE     MAPE     MASE
## Training set -1.936656e-13 0.5815815 0.4583659 -12.5498 27.98264 0.781347
##                    ACF1
## Training set 0.03974611
```

```
summary(whoop_sleep_deep_fit)
```

```
## Series: whoop_sleep_rem_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##         mean
##       2.3212
## s.e.  0.1141
##
## sigma^2 estimated as 0.3518:  log likelihood=-22.8
## AIC=49.6   AICc=50.12   BIC=52.12
##
## Training set error measures:
##                         ME      RMSE       MAE      MPE     MAPE     MASE
## Training set -1.936656e-13 0.5815815 0.4583659 -12.5498 27.98264 0.781347
##                    ACF1
## Training set 0.03974611
```

```
summary(whoop_avghr_fit)
```

```
## Series: whoop_avghr_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##          mean
##       70.5769
## s.e.   1.0179
##
## sigma^2 estimated as 28.01:  log likelihood=-79.71
## AIC=163.42   AICc=163.94   BIC=165.93
##
## Training set error measures:
##                        ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 3.464169e-12 5.190028 4.168639 -0.5889657 6.156207 0.8337278
##                   ACF1
## Training set 0.0644377
```

```
summary(whoop_maxhr_fit)
```

```
## Series: whoop_maxhr_ts
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##           mean
##       173.2692
## s.e.    3.9698
##
## sigma^2 estimated as 426.1:  log likelihood=-115.09
## AIC=234.19   AICc=234.71   BIC=236.7
##
## Training set error measures:
##                         ME     RMSE      MAE       MPE     MAPE      MASE
## Training set -4.373371e-15 20.24192 13.40237 -1.790323 8.683849 0.7053877
##                   ACF1
## Training set 0.1306159
```

```
###All White Noise processes, so lets look at the vector time series model instead

##check correlations, I used all data because it gave me more observations, its ok if you have missing data because correlations doesnt need to be time series, this is jus
t background info to see if there is anything interesting
cor(whoopdata_3.0, use="pairwise")
```

```
##                          recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv                1.0000000  -0.54246168             -0.4357430
## recovery_rhr               -0.5424617   1.00000000              0.5431748
## sleep_respiratory_rate     -0.4357430   0.54317484              1.0000000
## sleep_duration_light        0.1240174   0.06320028             -0.0136364
## sleep_duration_rem          0.3400483  -0.19879692             -0.3278822
## sleep_duration_deep         0.2854707  -0.07250913             -0.1897825
## strain_avghr               -0.1255966   0.03347984              0.2440619
## strain_maxhr                0.1336224  -0.14610984              0.1276505
##                          sleep_duration_light sleep_duration_rem
## recovery_hrv                       0.12401739          0.3400483
## recovery_rhr                       0.06320028         -0.1987969
## sleep_respiratory_rate            -0.01363640         -0.3278822
## sleep_duration_light               1.00000000          0.2476376
## sleep_duration_rem                 0.24763764          1.0000000
## sleep_duration_deep                0.21500802          0.7362322
## strain_avghr                      -0.28015469         -0.4142152
## strain_maxhr                       0.07275458         -0.1842661
##                          sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                      0.28547071  -0.12559656   0.13362244
## recovery_rhr                     -0.07250913   0.03347984  -0.14610984
## sleep_respiratory_rate           -0.18978249   0.24406189   0.12765052
## sleep_duration_light              0.21500802  -0.28015469   0.07275458
## sleep_duration_rem                0.73623220  -0.41421516  -0.18426615
## sleep_duration_deep               1.00000000  -0.39955108  -0.14792095
## strain_avghr                     -0.39955108   1.00000000   0.68835299
## strain_maxhr                     -0.14792095   0.68835299   1.00000000
```

```
whoop.cor <- cor(whoopdata_3.0, use = 'pairwise')
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
corrplot(whoop.cor)
```

```
## Warning in corrplot(whoop.cor): Not been able to calculate text margin,
## please try again with a clean new empty window using {plot.new();
## dev.off()} or reduce tl.cex
```

```
library("Hmisc")
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
cor <- rcorr(as.matrix(whoopdata_3.0))
cor
```

```
##                         recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv                    1.00        -0.54                  -0.44
## recovery_rhr                   -0.54         1.00                   0.54
## sleep_respiratory_rate         -0.44         0.54                   1.00
## sleep_duration_light            0.12         0.06                  -0.01
## sleep_duration_rem              0.34        -0.20                  -0.33
## sleep_duration_deep             0.29        -0.07                  -0.19
## strain_avghr                   -0.13         0.03                   0.24
## strain_maxhr                    0.13        -0.15                   0.13
##                         sleep_duration_light sleep_duration_rem
## recovery_hrv                            0.12               0.34
## recovery_rhr                            0.06              -0.20
## sleep_respiratory_rate                 -0.01              -0.33
## sleep_duration_light                    1.00               0.25
## sleep_duration_rem                      0.25               1.00
## sleep_duration_deep                     0.22               0.74
## strain_avghr                           -0.28              -0.41
## strain_maxhr                            0.07              -0.18
##                         sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                           0.29        -0.13         0.13
## recovery_rhr                          -0.07         0.03        -0.15
## sleep_respiratory_rate                -0.19         0.24         0.13
## sleep_duration_light                   0.22        -0.28         0.07
## sleep_duration_rem                     0.74        -0.41        -0.18
## sleep_duration_deep                    1.00        -0.40        -0.15
## strain_avghr                          -0.40         1.00         0.69
## strain_maxhr                          -0.15         0.69         1.00
##
## n
##                         recovery_hrv recovery_rhr sleep_respiratory_rate
```

```
## recovery_hrv                    58           58                      58
## recovery_rhr                    58           58                      58
## sleep_respiratory_rate          58           58                      58
## sleep_duration_light            58           58                      58
## sleep_duration_rem              58           58                      58
## sleep_duration_deep             58           58                      58
## strain_avghr                    58           58                      58
## strain_maxhr                    58           58                      58
##                        sleep_duration_light sleep_duration_rem
## recovery_hrv                             58                 58
## recovery_rhr                             58                 58
## sleep_respiratory_rate                   58                 58
## sleep_duration_light                     58                 58
## sleep_duration_rem                       58                 58
## sleep_duration_deep                      58                 58
## strain_avghr                             58                 58
## strain_maxhr                             58                 58
##                        sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                            58           58           58
## recovery_rhr                            58           58           58
## sleep_respiratory_rate                  58           58           58
## sleep_duration_light                    58           58           58
## sleep_duration_rem                      58           58           58
## sleep_duration_deep                     58           58           58
## strain_avghr                            58           61           61
## strain_maxhr                            58           61           61
##
## P
##                        recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv                        0.0000       0.0006
## recovery_rhr           0.0000                    0.0000
## sleep_respiratory_rate 0.0006       0.0000
## sleep_duration_light   0.3537       0.6374       0.9191
## sleep_duration_rem     0.0090       0.1347       0.0120
## sleep_duration_deep    0.0298       0.5886       0.1536
## strain_avghr           0.3475       0.8030       0.0649
## strain_maxhr           0.3173       0.2738       0.3396
##                        sleep_duration_light sleep_duration_rem
## recovery_hrv           0.3537               0.0090
## recovery_rhr           0.6374               0.1347
## sleep_respiratory_rate 0.9191               0.0120
## sleep_duration_light                        0.0609
## sleep_duration_rem     0.0609
## sleep_duration_deep    0.1051               0.0000
## strain_avghr           0.0332               0.0012
## strain_maxhr           0.5873               0.1662
##                        sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv           0.0298              0.3475       0.3173
## recovery_rhr           0.5886              0.8030       0.2738
## sleep_respiratory_rate 0.1536              0.0649       0.3396
```

```
## sleep_duration_light   0.1051                0.0332       0.5873
## sleep_duration_rem     0.0000                0.0012       0.1662
## sleep_duration_deep                          0.0019       0.2678
## strain_avghr           0.0019                             0.0000
## strain_maxhr           0.2678                0.0000
```

```
### Relationship might exist between:
##    RHR and HRV (p=0.0000)
##    Resp Rate and HRV (p=0.0006)
##    Resp Rate and RHR (p=0.0000)
##    Duration REM and HRV (p=0.0090)
##    Duration REM and Resp Rate (p=0.0120)
##    Deep sleep and HRV (p=0.0298)
##    Deep Sleep and REM (p=0.0000)
##    Avg HR and Duration Light (p=0.0332)
##    Avg HR and Duration REM (p=0.0012)
##    Avg HR and Duration Deep (p=0.0019)
##    Max HR and Avg HR (p=0.0000)


## then check if we should remove any variables from the time series
cor(whoopdata_0926_1021) ###none over 0.95 so leave all variables in
```

```
##                         recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv               1.0000000  -0.57861356            -0.50900237
## recovery_rhr              -0.5786136   1.00000000             0.62999495
## sleep_respiratory_rate    -0.5090024   0.62999495             1.00000000
## sleep_duration_light       0.2080116  -0.14077362            -0.07418001
## sleep_duration_rem         0.3588951  -0.24492083            -0.11605829
## sleep_duration_deep        0.4589825  -0.20289252            -0.21136359
## strain_avghr              -0.0763689  -0.08405426            -0.27966469
## strain_maxhr               0.2621704  -0.28219976            -0.26579452
##                         sleep_duration_light sleep_duration_rem
## recovery_hrv                      0.20801157          0.3588951
## recovery_rhr                     -0.14077362         -0.2449208
## sleep_respiratory_rate           -0.07418001         -0.1160583
## sleep_duration_light              1.00000000          0.4310873
## sleep_duration_rem                0.43108729          1.0000000
## sleep_duration_deep               0.37479600          0.7267518
## strain_avghr                     -0.09337336         -0.1268298
## strain_maxhr                      0.38786358          0.1482276
##                         sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                     0.45898252  -0.07636890   0.26217042
## recovery_rhr                    -0.20289252  -0.08405426  -0.28219976
## sleep_respiratory_rate          -0.21136359  -0.27966469  -0.26579452
## sleep_duration_light             0.37479600  -0.09337336   0.38786358
## sleep_duration_rem               0.72675176  -0.12682982   0.14822760
## sleep_duration_deep              1.00000000  -0.30308735  -0.03674036
## strain_avghr                    -0.30308735   1.00000000   0.65018788
## strain_maxhr                    -0.03674036   0.65018788   1.00000000
```

```
### Lets move on to VAR model (make a model for all the variables)
library(vars)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following objects are masked from 'package:fma':
##
##     cement, housing, petrol
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Loading required package: strucchange
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
##
## Attaching package: 'strucchange'
```

```
## The following object is masked from 'package:stringr':
##
##     boundary
```

```
## Loading required package: urca
```

```
## Loading required package: lmtest
```

```
whoop_3best <- VARselect(whoopdata_0926_1021,lag.max=10,type="none")
whoop_3best
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      2      2      2      3
##
## $criteria
##                  1    2    3    4    5    6    7    8    9   10
## AIC(n)    8.324898 -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf
## HQ(n)     8.483150 -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf
## SC(n)    11.415253 -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf -Inf
## FPE(n) 9079.702632  NaN    0    0    0    0    0    0    0    0
```

```
 ###based on SC criterion, VAR(2) model is best

## make 8 models (one VAR model), one for each of the variables, notice with other va
riables at which lags are statistically significant
whoop3.0.fit <- VAR(whoopdata_0926_1021, p=2, type="none")
summary(whoop3.0.fit)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: recovery_hrv, recovery_rhr, sleep_respiratory_rate, sleep_du
ration_light, sleep_duration_rem, sleep_duration_deep, strain_avghr, strain_maxhr
## Deterministic variables: none
## Sample size: 24
## Log Likelihood: -180.909
## Roots of the characteristic polynomial:
## 0.9971 0.9971 0.9969 0.9887 0.9887 0.9753 0.9753 0.858 0.858 0.8142 0.6814 0.6814
0.6671 0.6671 0.4641 0.4641
## Call:
## VAR(y = whoopdata_0926_1021, p = 2, type = "none")
##
##
## Estimation results for equation recovery_hrv:
## =============================================
## recovery_hrv = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l1 + sle
ep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain_avghr.
l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_rate.l2
+ sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + strain_a
vghr.l2 + strain_maxhr.l2
##
##                           Estimate Std. Error t value Pr(>|t|)    
## recovery_hrv.l1             0.6382     0.2634   2.423 0.041651 *  
## recovery_rhr.l1             5.9800     1.5241   3.924 0.004396 ** 
## sleep_respiratory_rate.l1  13.0343     7.2744   1.792 0.110934    
## sleep_duration_light.l1    22.5129     8.5704   2.627 0.030327 *  
## sleep_duration_rem.l1      42.0981    10.9983   3.828 0.005034 ** 
## sleep_duration_deep.l1    -28.4056    16.3861  -1.734 0.121232    
## strain_avghr.l1             2.6899     1.3306   2.022 0.077861 .  
## strain_maxhr.l1             0.4530     0.3581   1.265 0.241483    
## recovery_hrv.l2            -0.4195     0.1917  -2.189 0.060046 .  
## recovery_rhr.l2             2.9524     1.1232   2.629 0.030242 *  
## sleep_respiratory_rate.l2 -43.4198     6.9810  -6.220 0.000254 ***
## sleep_duration_light.l2   -13.7175    10.5391  -1.302 0.229286    
## sleep_duration_rem.l2      25.7993     9.3463   2.760 0.024661 *  
## sleep_duration_deep.l2     -7.8173    13.3764  -0.584 0.575051    
## strain_avghr.l2            -5.1224     1.8253  -2.806 0.022971 *  
## strain_maxhr.l2             0.3379     0.2592   1.304 0.228563    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 13.23 on 8 degrees of freedom
## Multiple R-Squared: 0.9932,  Adjusted R-squared: 0.9797
## F-statistic: 73.33 on 16 and 8 DF,  p-value: 6.684e-07
##
##
## Estimation results for equation recovery_rhr:
```

```
## ===========================================
## recovery_rhr = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l1 + sle
ep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain_avghr.
l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_rate.l2
+ sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + strain_a
vghr.l2 + strain_maxhr.l2
##
##                            Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1            -0.08240    0.05992  -1.375   0.2063
## recovery_rhr.l1            -0.61974    0.34673  -1.787   0.1117
## sleep_respiratory_rate.l1 -1.52897    1.65488  -0.924   0.3826
## sleep_duration_light.l1   -1.55341    1.94970  -0.797   0.4486
## sleep_duration_rem.l1     -4.26425    2.50202  -1.704   0.1267
## sleep_duration_deep.l1     3.23354    3.72771   0.867   0.4110
## strain_avghr.l1            0.06166    0.30269   0.204   0.8437
## strain_maxhr.l1           -0.03521    0.08146  -0.432   0.6770
## recovery_hrv.l2           -0.01372    0.04360  -0.315   0.7610
## recovery_rhr.l2           -0.09806    0.25551  -0.384   0.7111
## sleep_respiratory_rate.l2  2.76881    1.58812   1.743   0.1194
## sleep_duration_light.l2    2.85370    2.39756   1.190   0.2681
## sleep_duration_rem.l2     -1.23624    2.12622  -0.581   0.5770
## sleep_duration_deep.l2     6.54650    3.04303   2.151   0.0636 .
## strain_avghr.l2            0.93128    0.41524   2.243   0.0552 .
## strain_maxhr.l2            0.05472    0.05897   0.928   0.3806
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 3.009 on 8 degrees of freedom
## Multiple R-Squared: 0.9989,  Adjusted R-squared: 0.9967
## F-statistic:   457 on 16 and 8 DF,  p-value: 4.677e-10
##
##
## Estimation results for equation sleep_respiratory_rate:
## =====================================================
## sleep_respiratory_rate = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rat
e.l1 + sleep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + str
ain_avghr.l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respirator
y_rate.l2 + sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2
+ strain_avghr.l2 + strain_maxhr.l2
##
##                             Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1            -0.003733   0.010784  -0.346   0.7381
## recovery_rhr.l1            -0.089176   0.062403  -1.429   0.1909
## sleep_respiratory_rate.l1  0.072569   0.297842   0.244   0.8136
## sleep_duration_light.l1   -0.326375   0.350903  -0.930   0.3795
## sleep_duration_rem.l1     -0.784706   0.450310  -1.743   0.1196
## sleep_duration_deep.l1    -0.211345   0.670908  -0.315   0.7608
## strain_avghr.l1           -0.017031   0.054478  -0.313   0.7626
## strain_maxhr.l1            0.002025   0.014662   0.138   0.8936
```

```
## recovery_hrv.l2          -0.001290   0.007847  -0.164   0.8735  
## recovery_rhr.l2          -0.043828   0.045986  -0.953   0.3685  
## sleep_respiratory_rate.l2  0.769880   0.285827   2.694   0.0273 *
## sleep_duration_light.l2    0.094978   0.431509   0.220   0.8313  
## sleep_duration_rem.l2      0.128475   0.382674   0.336   0.7457  
## sleep_duration_deep.l2     1.107728   0.547680   2.023   0.0777 .
## strain_avghr.l2            0.144955   0.074734   1.940   0.0884 .
## strain_maxhr.l2            0.009090   0.010613   0.857   0.4166  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## 
## Residual standard error: 0.5416 on 8 degrees of freedom
## Multiple R-Squared: 0.9996,  Adjusted R-squared: 0.9989 
## F-statistic:  1394 on 16 and 8 DF,  p-value: 5.439e-12 
## 
## 
## Estimation results for equation sleep_duration_light: 
## ===================================================== 
## sleep_duration_light = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.
l1 + sleep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strai
n_avghr.l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_
rate.l2 + sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + 
strain_avghr.l2 + strain_maxhr.l2 
## 
##                            Estimate Std. Error t value Pr(>|t|)  
## recovery_hrv.l1            0.022720   0.012183   1.865   0.0992 .
## recovery_rhr.l1            0.164852   0.070498   2.338   0.0475 *
## sleep_respiratory_rate.l1  0.317000   0.336476   0.942   0.3737  
## sleep_duration_light.l1    0.893696   0.396420   2.254   0.0542 .
## sleep_duration_rem.l1      1.005857   0.508721   1.977   0.0834 .
## sleep_duration_deep.l1    -0.883914   0.757934  -1.166   0.2771  
## strain_avghr.l1            0.118229   0.061545   1.921   0.0910 .
## strain_maxhr.l1           -0.004447   0.016564  -0.268   0.7951  
## recovery_hrv.l2            0.002942   0.008865   0.332   0.7485  
## recovery_rhr.l2           -0.021592   0.051951  -0.416   0.6886  
## sleep_respiratory_rate.l2 -0.258513   0.322902  -0.801   0.4465  
## sleep_duration_light.l2   -1.077445   0.487481  -2.210   0.0581 .
## sleep_duration_rem.l2     -0.263184   0.432312  -0.609   0.5596  
## sleep_duration_deep.l2    -0.161455   0.618721  -0.261   0.8007  
## strain_avghr.l2           -0.194335   0.084427  -2.302   0.0503 .
## strain_maxhr.l2           -0.008417   0.011989  -0.702   0.5025  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## 
## Residual standard error: 0.6119 on 8 degrees of freedom
## Multiple R-Squared: 0.9851,  Adjusted R-squared: 0.9554 
## F-statistic: 33.14 on 16 and 8 DF,  p-value: 1.482e-05 
## 
```

```
##
## Estimation results for equation sleep_duration_rem:
## ===================================================
## sleep_duration_rem = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l1
+ sleep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain_a
vghr.l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_rat
e.l2 + sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + str
ain_avghr.l2 + strain_maxhr.l2
##
##                            Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1            0.014703   0.012539   1.173   0.2747
## recovery_rhr.l1            0.084642   0.072556   1.167   0.2770
## sleep_respiratory_rate.l1  0.688925   0.346301   1.989   0.0819 .
## sleep_duration_light.l1    0.224743   0.407995   0.551   0.5968
## sleep_duration_rem.l1      0.703344   0.523575   1.343   0.2160
## sleep_duration_deep.l1    -0.575061   0.780064  -0.737   0.4821
## strain_avghr.l1            0.068151   0.063342   1.076   0.3133
## strain_maxhr.l1            0.003343   0.017047   0.196   0.8494
## recovery_hrv.l2           -0.012773   0.009124  -1.400   0.1991
## recovery_rhr.l2           -0.014786   0.053468  -0.277   0.7891
## sleep_respiratory_rate.l2 -0.505205   0.332330  -1.520   0.1670
## sleep_duration_light.l2   -0.365731   0.501715  -0.729   0.4868
## sleep_duration_rem.l2     -0.267611   0.444935  -0.601   0.5642
## sleep_duration_deep.l2    -0.012812   0.636787  -0.020   0.9844
## strain_avghr.l2           -0.100619   0.086893  -1.158   0.2803
## strain_maxhr.l2           -0.014461   0.012339  -1.172   0.2749
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.6298 on 8 degrees of freedom
## Multiple R-Squared: 0.9776,  Adjusted R-squared: 0.9328
## F-statistic: 21.81 on 16 and 8 DF,  p-value: 7.343e-05
##
##
## Estimation results for equation sleep_duration_deep:
## ====================================================
## sleep_duration_deep = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l
1 + sleep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain
_avghr.l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_r
ate.l2 + sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + s
train_avghr.l2 + strain_maxhr.l2
##
##                            Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1            0.013714   0.008465   1.620   0.1439
## recovery_rhr.l1            0.130569   0.048986   2.665   0.0286 *
## sleep_respiratory_rate.l1  0.481576   0.233803   2.060   0.0734 .
## sleep_duration_light.l1    0.174539   0.275455   0.634   0.5440
## sleep_duration_rem.l1      0.883984   0.353488   2.501   0.0369 *
## sleep_duration_deep.l1    -0.451817   0.526655  -0.858   0.4159
```

```
## strain_avghr.l1             0.035026   0.042765   0.819   0.4365
## strain_maxhr.l1             0.009008   0.011509   0.783   0.4564
## recovery_hrv.l2            -0.015038   0.006160  -2.441   0.0405 *
## recovery_rhr.l2            -0.006461   0.036099  -0.179   0.8624
## sleep_respiratory_rate.l2 -0.410970   0.224371  -1.832   0.1044
## sleep_duration_light.l2   -0.705801   0.338729  -2.084   0.0707 .
## sleep_duration_rem.l2       0.013545   0.300395   0.045   0.9651
## sleep_duration_deep.l2      0.059508   0.429922   0.138   0.8933
## strain_avghr.l2            -0.152833   0.058665  -2.605   0.0314 *
## strain_maxhr.l2             0.003348   0.008331   0.402   0.6983
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.4252 on 8 degrees of freedom
## Multiple R-Squared: 0.9719,  Adjusted R-squared: 0.9157
## F-statistic:  17.3 on 16 and 8 DF,  p-value: 0.0001752
##
##
## Estimation results for equation strain_avghr:
## =============================================
## strain_avghr = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l1 + sle
ep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain_avghr.
l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_rate.l2
+ sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + strain_a
vghr.l2 + strain_maxhr.l2
##
##                           Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1            -0.0509     0.1446  -0.352    0.734
## recovery_rhr.l1            -0.3621     0.8368  -0.433    0.677
## sleep_respiratory_rate.l1   1.0271     3.9940   0.257    0.804
## sleep_duration_light.l1     2.6493     4.7055   0.563    0.589
## sleep_duration_rem.l1       1.0835     6.0386   0.179    0.862
## sleep_duration_deep.l1      2.3100     8.9968   0.257    0.804
## strain_avghr.l1             0.3388     0.7306   0.464    0.655
## strain_maxhr.l1             0.0705     0.1966   0.359    0.729
## recovery_hrv.l2             0.1020     0.1052   0.969    0.361
## recovery_rhr.l2             0.1996     0.6167   0.324    0.754
## sleep_respiratory_rate.l2   3.4801     3.8329   0.908    0.390
## sleep_duration_light.l2     3.3191     5.7865   0.574    0.582
## sleep_duration_rem.l2      -3.9613     5.1316  -0.772    0.462
## sleep_duration_deep.l2     -6.7645     7.3443  -0.921    0.384
## strain_avghr.l2            -0.1711     1.0022  -0.171    0.869
## strain_maxhr.l2            -0.1642     0.1423  -1.154    0.282
##
##
## Residual standard error: 7.263 on 8 degrees of freedom
## Multiple R-Squared: 0.9964,  Adjusted R-squared: 0.9893
## F-statistic: 139.9 on 16 and 8 DF,  p-value: 5.204e-08
##
```

```
##
## Estimation results for equation strain_maxhr:
## ===========================================
## strain_maxhr = recovery_hrv.l1 + recovery_rhr.l1 + sleep_respiratory_rate.l1 + sle
ep_duration_light.l1 + sleep_duration_rem.l1 + sleep_duration_deep.l1 + strain_avghr.
l1 + strain_maxhr.l1 + recovery_hrv.l2 + recovery_rhr.l2 + sleep_respiratory_rate.l2
+ sleep_duration_light.l2 + sleep_duration_rem.l2 + sleep_duration_deep.l2 + strain_a
vghr.l2 + strain_maxhr.l2
##
##                           Estimate Std. Error t value Pr(>|t|)
## recovery_hrv.l1             0.3844     0.4141   0.928    0.380
## recovery_rhr.l1             2.4805     2.3964   1.035    0.331
## sleep_respiratory_rate.l1   7.2404    11.4377   0.633    0.544
## sleep_duration_light.l1    15.8808    13.4753   1.179    0.272
## sleep_duration_rem.l1      31.1528    17.2927   1.802    0.109
## sleep_duration_deep.l1    -26.6737    25.7641  -1.035    0.331
## strain_avghr.l1             2.9815     2.0921   1.425    0.192
## strain_maxhr.l1             0.3713     0.5630   0.659    0.528
## recovery_hrv.l2             0.2877     0.3013   0.955    0.368
## recovery_rhr.l2             1.9085     1.7659   1.081    0.311
## sleep_respiratory_rate.l2  -9.1118    10.9762  -0.830    0.431
## sleep_duration_light.l2    -4.4904    16.5707  -0.271    0.793
## sleep_duration_rem.l2       9.8543    14.6954   0.671    0.521
## sleep_duration_deep.l2    -29.7763    21.0319  -1.416    0.195
## strain_avghr.l2            -4.8547     2.8699  -1.692    0.129
## strain_maxhr.l2            -0.4044     0.4075  -0.992    0.350
##
##
## Residual standard error: 20.8 on 8 degrees of freedom
## Multiple R-Squared: 0.9952,  Adjusted R-squared: 0.9856
## F-statistic: 103.7 on 16 and 8 DF,  p-value: 1.703e-07
##
##
##
## Covariance matrix of residuals:
##                        recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv                174.995     -31.6738               -2.42975
## recovery_rhr                -31.674       9.0564                0.36319
## sleep_respiratory_rate       -2.430       0.3632                0.29337
## sleep_duration_light         -5.887       0.7050                0.04061
## sleep_duration_rem           -2.012      -0.2004                0.02979
## sleep_duration_deep          -1.426       0.2226                0.05441
## strain_avghr                 69.434      -9.6125               -1.50691
## strain_maxhr                131.905     -18.9264               -2.99808
##                        sleep_duration_light sleep_duration_rem
## recovery_hrv                       -5.88732           -2.01202
## recovery_rhr                        0.70498           -0.20040
## sleep_respiratory_rate              0.04061            0.02979
## sleep_duration_light                0.37441            0.16634
## sleep_duration_rem                  0.16634            0.39659
```

```
## sleep_duration_deep                      0.07556           0.23210
## strain_avghr                            -1.43669          -1.71879
## strain_maxhr                             0.08942           0.75210
##                        sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                      -1.42622      69.4341    131.90516
## recovery_rhr                       0.22263      -9.6125    -18.92638
## sleep_respiratory_rate             0.05441      -1.5069     -2.99808
## sleep_duration_light               0.07556      -1.4367      0.08942
## sleep_duration_rem                 0.23210      -1.7188      0.75210
## sleep_duration_deep                0.18077      -0.9048      1.47713
## strain_avghr                      -0.90475      52.7531    127.88518
## strain_maxhr                       1.47713     127.8852    432.62496
##
## Correlation matrix of residuals:
##                        recovery_hrv recovery_rhr sleep_respiratory_rate
## recovery_hrv                 1.0000      -0.7956               -0.33911
## recovery_rhr                -0.7956       1.0000                0.22282
## sleep_respiratory_rate      -0.3391       0.2228                1.00000
## sleep_duration_light        -0.7273       0.3828                0.12254
## sleep_duration_rem          -0.2415      -0.1057                0.08734
## sleep_duration_deep         -0.2536       0.1740                0.23628
## strain_avghr                 0.7227      -0.4398               -0.38305
## strain_maxhr                 0.4794      -0.3024               -0.26612
##                        sleep_duration_light sleep_duration_rem
## recovery_hrv                      -0.727335           -0.24152
## recovery_rhr                       0.382847           -0.10574
## sleep_respiratory_rate             0.122545            0.08734
## sleep_duration_light               1.000000            0.43166
## sleep_duration_rem                 0.431664            1.00000
## sleep_duration_deep                0.290423            0.86686
## strain_avghr                      -0.323271           -0.37577
## strain_maxhr                       0.007026            0.05742
##                        sleep_duration_deep strain_avghr strain_maxhr
## recovery_hrv                       -0.2536       0.7227     0.479395
## recovery_rhr                        0.1740      -0.4398    -0.302366
## sleep_respiratory_rate              0.2363      -0.3831    -0.266122
## sleep_duration_light                0.2904      -0.3233     0.007026
## sleep_duration_rem                  0.8669      -0.3758     0.057418
## sleep_duration_deep                 1.0000      -0.2930     0.167032
## strain_avghr                       -0.2930       1.0000     0.846527
## strain_maxhr                        0.1670       0.8465     1.000000
```

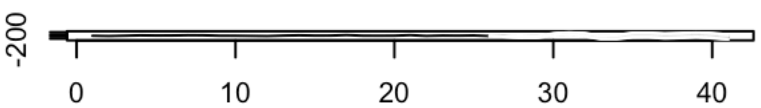
```
##Lets now use the model to predict the next 15 and see if they are similar to the da
ta we got after the missing week from 10/26-10/30 (starting day 6 of the 15 predicted
days)
# Prediction using a VAR(2) model with Intercept
whoop.pred <- predict(whoop3.0.fit,n.ahead=15,ci=0.95)
whoop.pred
```

```
## $recovery_hrv
##            fcst      lower      upper        CI
##  [1,] -10.05891  -35.98673  15.868905  25.92782
##  [2,] -64.80787 -127.15148  -2.464263  62.34361
##  [3,] -18.58036 -120.42707  83.266347 101.84671
##  [4,] 179.58836   59.68219 299.494536 119.90617
##  [5,] 326.28011  181.60659 470.953625 144.67352
##  [6,] 163.27400  -18.79702 345.345024 182.07102
##  [7,] -92.45490 -286.14567 101.235869 193.69077
##  [8,] -62.74666 -294.24410 168.750778 231.49744
##  [9,] 140.66435 -116.61248 397.941179 257.27683
## [10,] 152.82140 -108.49470 414.137501 261.31610
## [11,]  48.02695 -229.48103 325.534918 277.50797
## [12,]  97.00851 -183.13591 377.152935 280.14442
## [13,] 187.28667  -97.15393 471.727266 284.44060
## [14,]  83.07026 -205.94954 372.090061 289.01980
## [15,] -70.25380 -363.16222 222.654614 292.90842
##
## $recovery_rhr
##           fcst      lower     upper        CI
##  [1,] 56.98790  51.089518  62.88628  5.898382
##  [2,] 37.71991  30.278168  45.16166  7.441747
##  [3,] 42.70737  32.400314  53.01443 10.307061
##  [4,] 72.47494  59.901370  85.04852 12.573574
##  [5,] 65.52751  50.076651  80.97838 15.450863
##  [6,] 20.55092   1.370406  39.73143 19.180513
##  [7,] 16.25875  -6.302320  38.81982 22.561072
##  [8,] 63.99912  32.958617  95.03963 31.040505
##  [9,] 79.82264  47.071524 112.57375 32.751115
## [10,] 41.91856   3.145056  80.69206 38.773503
## [11,] 23.44758 -15.700183  62.59535 39.147768
## [12,] 48.85743   4.872168  92.84268 43.985258
## [13,] 61.64172  16.223588 107.05986 45.418134
## [14,] 43.98071  -2.506940  90.46837 46.487655
## [15,] 35.94342 -11.036919  82.92376 46.980342
##
## $sleep_respiratory_rate
##           fcst     lower    upper       CI
##  [1,] 18.16371 17.102127 19.22529 1.061582
##  [2,] 14.58155 12.919556 16.24355 1.661998
##  [3,] 12.40935 10.364171 14.45453 2.045180
##  [4,] 15.91699 13.162825 18.67115 2.754160
##  [5,] 17.59750 14.473709 20.72129 3.123791
##  [6,] 13.52843  9.957041 17.09981 3.571384
##  [7,] 11.90249  8.195883 15.60910 3.706610
##  [8,] 16.54598 12.027091 21.06487 4.518888
##  [9,] 18.24491 13.362659 23.12716 4.882252
## [10,] 13.68540  8.506320 18.86447 5.179077
## [11,] 11.71408  6.488100 16.94006 5.225980
## [12,] 15.37086  9.448040 21.29369 5.922824
```

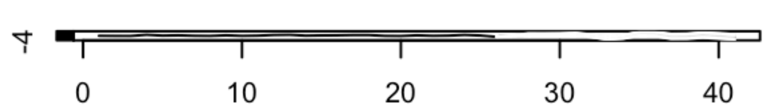
```
## [13,] 17.08900 10.881731 23.29627 6.207270
## [14,] 14.91642  8.574309 21.25854 6.342113
## [15,] 14.14912  7.759157 20.53908 6.389963
##
## $sleep_duration_light
##            fcst      lower    upper       CI
##  [1,] 0.8281830 -0.3710997 2.027466 1.199283
##  [2,] 3.1656004  1.2898570 5.041344 1.875743
##  [3,] 4.9701394  3.0165191 6.923760 1.953620
##  [4,] 2.6823960  0.4345659 4.930226 2.247830
##  [5,] 0.1521367 -2.1626046 2.466878 2.314741
##  [6,] 2.3101246 -0.5152032 5.135452 2.825328
##  [7,] 5.7415438  2.7561318 8.726956 2.985412
##  [8,] 4.1591302  0.8741205 7.444140 3.285010
##  [9,] 0.2130655 -3.3190543 3.745185 3.532120
## [10,] 0.7348914 -3.0024694 4.472252 3.737361
## [11,] 4.2593598  0.1516020 8.367118 4.107758
## [12,] 4.3847699  0.2017019 8.567838 4.183068
## [13,] 1.5708393 -2.8856120 6.027291 4.456451
## [14,] 0.9455018 -3.5394782 5.430482 4.484980
## [15,] 2.8384622 -1.8859206 7.562845 4.724383
##
## $sleep_duration_rem
##              fcst      lower     upper       CI
##  [1,]  1.68364733  0.4493469 2.9179478 1.234300
##  [2,]  3.56981318  2.0022274 5.1373990 1.567586
##  [3,]  3.41985790  1.6825955 5.1571203 1.737262
##  [4,]  3.41027631  1.5009757 5.3195770 1.909301
##  [5,]  4.01793507  1.9978212 6.0380489 2.020114
##  [6,]  1.81169008 -0.3115257 3.9349059 2.123216
##  [7,] -1.38178640 -3.7390700 0.9754972 2.357284
##  [8,]  0.42746701 -2.4142209 3.2691549 2.841688
##  [9,]  4.91114230  1.7595469 8.0627377 3.151595
## [10,]  4.52412611  0.9493704 8.0988819 3.574756
## [11,]  0.99153621 -3.0139983 4.9970707 4.005534
## [12,]  1.14755052 -2.9269489 5.2220499 4.074499
## [13,]  3.12106277 -1.2348673 7.4769928 4.355930
## [14,]  1.69800549 -2.7103879 6.1063989 4.408393
## [15,]  0.04597254 -4.4143069 4.5062520 4.460279
##
## $sleep_duration_deep
##             fcst      lower     upper        CI
##  [1,]  0.4381396 -0.3951899 1.2714691 0.8333295
##  [2,]  2.1756309  0.8863630 3.4648988 1.2892679
##  [3,]  2.2266040  0.9236974 3.5295106 1.3029066
##  [4,]  1.5731824  0.1410589 3.0053059 1.4321235
##  [5,]  2.8097925  1.2890479 4.3305370 1.5207446
##  [6,]  2.9841229  1.3692172 4.5990287 1.6149057
##  [7,] -0.0384644 -1.9183945 1.8414657 1.8799301
##  [8,] -1.4242023 -3.4982802 0.6498756 2.0740779
```

```
##  [9,]  1.4416967 -1.0444068 3.9278002 2.4861035
## [10,]  3.2822051  0.6766017 5.8878085 2.6056034
## [11,]  1.4258908 -1.5549639 4.4067456 2.9808548
## [12,]  0.4401227 -2.6232269 3.5034722 3.0633496
## [13,]  1.8247775 -1.3684905 5.0180454 3.1932680
## [14,]  1.5576558 -1.7022058 4.8175174 3.2598616
## [15,] -0.2124617 -3.5229593 3.0980359 3.3104976
##
## $strain_avghr
##            fcst      lower     upper       CI
##  [1,]  47.87757  33.641945  62.11319 14.23562
##  [2,]  74.94804  58.557930  91.33816 16.39011
##  [3,] 104.48587  84.965468 124.00627 19.52040
##  [4,]  71.55877  44.852458  98.26508 26.70631
##  [5,]  13.80014 -14.546203  42.14649 28.34635
##  [6,]  29.84694  -8.557865  68.25175 38.40481
##  [7,]  94.14225  47.272664 141.01184 46.86959
##  [8,]  98.15688  48.187141 148.12661 49.96973
##  [9,]  50.05447  -6.111145 106.22008 56.16561
## [10,]  45.07698 -11.869003 102.02296 56.94598
## [11,]  78.76363  16.890791 140.63646 61.87284
## [12,]  75.25388  12.582237 137.92552 62.67164
## [13,]  43.35755 -20.466880 107.18197 63.82443
## [14,]  45.04158 -19.479525 109.56268 64.52110
## [15,]  73.28726   6.223918 140.35059 67.06334
##
## $strain_maxhr
##            fcst      lower    upper        CI
##  [1,]  62.37167   21.60506 103.1383  40.76661
##  [2,] 126.97028   61.21512 192.7254  65.75516
##  [3,] 230.81289  156.68994 304.9358  74.12295
##  [4,] 173.39626   80.74099 266.0515  92.65527
##  [5,]  56.23528  -42.70326 155.1738  98.93854
##  [6,] 132.18908   16.59223 247.7859 115.59685
##  [7,] 287.38783  161.25724 413.5184 126.13058
##  [8,] 210.71827   76.21039 345.2261 134.50788
##  [9,]  27.14600 -116.12777 170.4198 143.27377
## [10,]  80.93971  -76.06966 237.9491 157.00937
## [11,] 268.42007   93.52165 443.3185 174.89842
## [12,] 247.68151   65.97257 429.3904 181.70894
## [13,]  88.83673 -108.14072 285.8142 196.97745
## [14,]  78.49636 -120.99722 277.9899 199.49358
## [15,] 177.60031  -36.01526 391.2159 213.61557
```

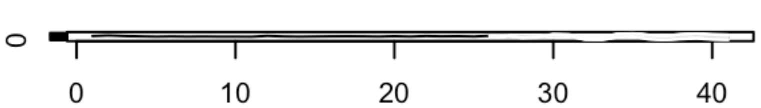
```
fanchart(whoop.pred)
```



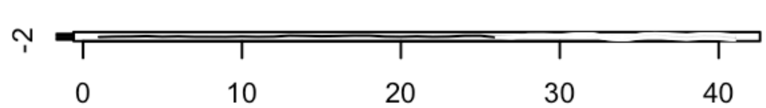
Fanchart for variable recovery_hrv



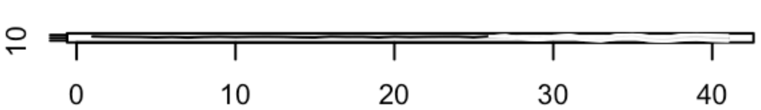
Fanchart for variable sleep_duration_rem



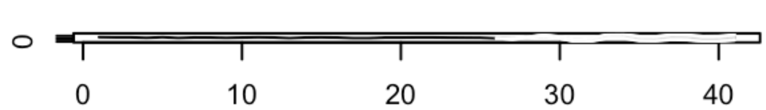
Fanchart for variable recovery_rhr



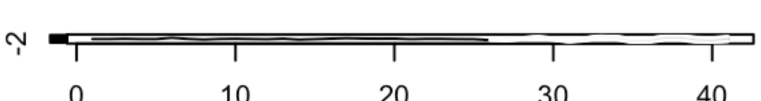
Fanchart for variable sleep_duration_deep



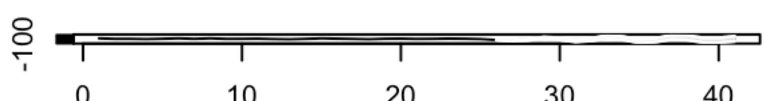
Fanchart for variable sleep_respiratory_rate



Fanchart for variable strain_avghr



Fanchart for variable sleep_duration_light



Fanchart for variable strain_maxhr

```
##these are not helpful predcitions because the CI is so wide, this is because it is
difficult to predict any more than 3 days into the future.
```