Interday Stability and Intraday Variability

Jade Benson

2/8/2021

#Interday Stability

Measures how stable rhythm is over the study window; how similar are the individual's day-night patterns. Calculated by taking the variance of each hour around its mean for the three days.

$$IS = \frac{n \sum_{h=1}^{p} (\bar{x}_h - \bar{x})^2}{p \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Where: n = total number of data p = number of data per day (24) x_h = hourly means $\bar{X}$ = mean of all data x_i = individual data points

#Intraday Variability Measures how fragmented the rhythm is throughout the study window; what is the rate of shifting between rest and activity (within days). Calculated by taking the difference of lagged hour points and dividing by the overall variance.

$$IV = \frac{n \sum_{i=2}^{n} (x_i - x_{i-1})^2}{(n-1) \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Where: n = total number of data p = number of data per day (24) $\bar{X}$ = mean of all data x_i = individual data points x_{i-1} = data points from the hour prior

#Data Preparation Now I will walk through the steps to calculate these two measures to help with replicability and transparency. The code will be useful for debugging/double-checking - but visuals and other data checks will be performed throughout to be interpretable.

Start with the data cleaning. Create actigraphy record with the relevant su_ids that have the full three nights of actigraphy and subset all records down to only 72hr. In this version, I'm keeping the missing values (NOT recoding to 0) and I'm not running the change point analysis to pick up any patterns of off wrist time (since we're subsetting to first 72hr). However, there were a few cases where the initial period didn't have much data so we might have to revisit this if we want to use the change point analysis in our subsetting to pick a representative sample of actigraphy.

```
#read in the data
actigraphy <- read_dta("C:/Users/Jade/Documents/W2_Actigraphy/activity.dta")
```

```
#check the # of unique su_ids - 793
#length(unique(actigraphy$su_id))

#need to remove the extra su_ids
del.list<-c("10003091", "10004581", "10005091", "10007281", "10007850", "10009211",
            "10009851", "10010181", "10010651", "10011090", "10011131", "10011781",
            "10012511", "10013171", "10013641", "10014481", "10014750", "10016641",
            "10018210", "10019001", "10019471", "10020521", "10020931", "10022151",
            "10022211", "10022481", "10023131", "10023521", "10024091", "10025091",
            "10026231", "10026491", "10027611", "10028481", "10030090", "10030931",
            "10032130", "10032951", "10033020", "10034201", "10035010", "10035081",
            "10035181", "10036731", "10036781", "10037100", "10037791", "10039311",
            "10039651", "10039951", "10040491", "10042131", "10042491", "10043641")


#Remove these extra su_ids, create new dataset so don't edit original
sleepsample<- actigraphy[!actigraphy$su_id%in%del.list,]

#check subsetting worked - 739
#length(unique(sleepsample$su_id))

#over 12,000,000 missing values in the original dataset
#I originally recoded these to 0 but upon reflection, I think we should keep them so it doesn't skew an
#will just subset down to 72hr each and see how that affects the missingness

#Remove su_ids without 3 nights of actigraphy as defined in W2 NSHAP file

#filter out the actigraphs without three nights of data
sleepData <- read_dta("C:/Users/Jade/Documents/Sleep Mortality/sleepData.dta")
#missing at least one night of actigraph data
missingAct <- sleepData %>% filter(is.na(sleepData$actigraph_sleep1) | is.na(sleepData$actigraph_sleep2)
m_act <- as.vector(unlist(missingAct["su_id"]))
#remove these from the dataset
sleepsample <- sleepsample[!sleepsample$su_id%in%m_act,]

#check removing short actigraphy records worked - 689
length(unique(sleepsample$su_id))
```

```
## [1] 689
```

```
#currently activity is measured in epochs (every 15 sec)
#want to create MAXACT (maximum actigraphy count per minute)
head(sleepsample)
```

```
## # A tibble: 6 x 17
##   su_id  line datetime            interval_status off_wrist_status activity
##   <chr> <dbl> <dttm>                    <dbl+lbl>        <dbl+lbl>    <dbl>
## 1 1000~ 12078 2011-01-13 10:19:15      1 [active]     0 [on wrist]       59
## 2 1000~ 12079 2011-01-13 10:19:30      1 [active]     0 [on wrist]      143
## 3 1000~ 12080 2011-01-13 10:19:45      1 [active]     0 [on wrist]       78
## 4 1000~ 12081 2011-01-13 10:20:00      1 [active]     0 [on wrist]       35
## 5 1000~ 12082 2011-01-13 10:20:15      1 [active]     0 [on wrist]       54
```

```
## 6 1000~ 12083 2011-01-13 10:20:30      1 [active]     0 [on wrist]       39
## # ... with 11 more variables: marker <dbl>, white_light <dbl>, red_light <dbl>,
## #   green_light <dbl>, blue_light <dbl>, sleep_wake <dbl+lbl>,
## #   mobility <dbl+lbl>, s_w_status <dbl+lbl>, corrected <dbl>, major <dbl>,
## #   major_flag <dbl>
```

```
##Create MAXACT
  #create key to filter max values on by group
  sleepsample$min <- format(sleepsample$datetime, "%Y-%m-%d %H:%M")

  #select only rows with max values per minute per subject
  maxact_df <-sleepsample %>%
    group_by(su_id, min) %>%
    slice(which.max(activity))


  #probably only need su_id, datetime, activity (maybe interval_status and off_wrist_status)
  #add logmax for later
  maxact_df <- maxact_df %>%
    dplyr::select(su_id, datetime, min, activity, interval_status, off_wrist_status) %>%
    rename(maxact = activity) %>%
    mutate(logmax = log10(maxact + 1))

#view the data
head(maxact_df)
```

```
## # A tibble: 6 x 7
## # Groups:   su_id, min [6]
##   su_id datetime            min   maxact interval_status off_wrist_status logmax
##   <chr> <dttm>              <chr>  <dbl>       <dbl+lbl>        <dbl+lbl>  <dbl>
## 1 1000~ 2011-01-13 10:19:30 2011~    143      1 [active]     0 [on wrist]  2.16
## 2 1000~ 2011-01-13 10:20:15 2011~     54      1 [active]     0 [on wrist]  1.74
## 3 1000~ 2011-01-13 10:21:15 2011~    111      1 [active]     0 [on wrist]  2.05
## 4 1000~ 2011-01-13 10:22:00 2011~     51      1 [active]     0 [on wrist]  1.72
## 5 1000~ 2011-01-13 10:23:45 2011~      1      3 [rest]       0 [on wrist]  0.301
## 6 1000~ 2011-01-13 10:24:15 2011~    118      3 [rest]       0 [on wrist]  2.08
```

```
#no more missing values and no time off wrist
#looks like some high maxact values (we've already seen this outlier case that's causing that)
summary(maxact_df)
```

```
##     su_id              datetime                       min
##  Length:2880322     Min.   :2010-09-09 08:00:15   Length:2880322
##  Class :character   1st Qu.:2010-10-20 15:46:30   Class :character
##  Mode  :character   Median :2010-11-16 11:32:15   Mode  :character
##                     Mean   :2010-12-13 08:19:12
##                     3rd Qu.:2011-01-29 20:34:11
##                     Max.   :2011-06-08 12:56:00
##      maxact        interval_status off_wrist_status     logmax
##  Min.   :   0.00   Min.   :1.000   Min.   :0        Min.   :0.000
##  1st Qu.:   0.00   1st Qu.:1.000   1st Qu.:0        1st Qu.:0.000
##  Median :  21.00   Median :1.000   Median :0        Median :1.342
##  Mean   :  58.59   Mean   :2.213   Mean   :0        Mean   :1.099
```

```
##  3rd Qu.:  93.00   3rd Qu.:4.000   3rd Qu.:0        3rd Qu.:1.973
##  Max.   :2172.00   Max.   :4.000   Max.   :0        Max.   :3.337
```

```
#subset all data to only include 72hr
act_72 <- maxact_df %>%
  group_by(su_id) %>%
  mutate(id = row_number())

act_72 <- act_72[which(act_72$id <=4320), ]
head(act_72)
```

```
## # A tibble: 6 x 8
## # Groups:   su_id [1]
##   su_id datetime            min   maxact interval_status off_wrist_status logmax
##   <chr> <dttm>              <chr>  <dbl>       <dbl+lbl>        <dbl+lbl>  <dbl>
## 1 1000~ 2011-01-13 10:19:30 2011~    143      1 [active]     0 [on wrist]  2.16
## 2 1000~ 2011-01-13 10:20:15 2011~     54      1 [active]     0 [on wrist]  1.74
## 3 1000~ 2011-01-13 10:21:15 2011~    111      1 [active]     0 [on wrist]  2.05
## 4 1000~ 2011-01-13 10:22:00 2011~     51      1 [active]     0 [on wrist]  1.72
## 5 1000~ 2011-01-13 10:23:45 2011~      1      3 [rest]       0 [on wrist]  0.301
## 6 1000~ 2011-01-13 10:24:15 2011~    118      3 [rest]       0 [on wrist]  2.08
## # ... with 1 more variable: id <int>
```

```
summary(act_72)
```

```
##     su_id              datetime                      min
##  Length:2803110     Min.   :2010-09-09 08:00:15   Length:2803110
##  Class :character   1st Qu.:2010-10-20 15:54:45   Class :character
##  Mode  :character   Median :2010-11-16 13:27:30   Mode  :character
##                     Mean   :2010-12-13 14:39:32
##                     3rd Qu.:2011-01-30 03:47:00
##                     Max.   :2011-06-08 12:56:00
##      maxact        interval_status off_wrist_status     logmax
##  Min.   :   0.00   Min.   :1.000   Min.   :0        Min.   :0.000
##  1st Qu.:   0.00   1st Qu.:1.000   1st Qu.:0        1st Qu.:0.000
##  Median :  21.00   Median :1.000   Median :0        Median :1.342
##  Mean   :  58.78   Mean   :2.213   Mean   :0        Mean   :1.101
##  3rd Qu.:  94.00   3rd Qu.:4.000   3rd Qu.:0        3rd Qu.:1.978
##  Max.   :2172.00   Max.   :4.000   Max.   :0        Max.   :3.337
##        id
##  Min.   :   1
##  1st Qu.:1018
##  Median :2035
##  Mean   :2045
##  3rd Qu.:3052
##  Max.   :4320
```

```
#save for easy future use
#write.csv(act_72, "C:/Users/Jade/Documents/Sleep Mortality/act_72")
```

IS and IV require a more detailed data preparation, the above will be used in the cosine modeling. We need to calculate hourly mean actigraphy counts.

```r
##Create HRAVGS

#create key to filter average hourly values on su_id
hravgs <- act_72 %>%
  mutate(hr = format(datetime, "%Y-%m-%d %H"))

#length(hravgs$su_id)

#there may be hours with too little data to be meaningful
#arbitrarily, if less than 10 min of data exclude it
hravgs <- hravgs %>%
  group_by(su_id, hr) %>%
  filter(n() >= 10)
#length(hravgs$su_id) - does exclude ~1,500 obs

#mean maxact count for each hour
hravgs <- hravgs %>%
  group_by(su_id, hr) %>%
  summarize(hract = mean(maxact))
```

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

```r
#the warning just letting me know that I've grouped twice (which is what I want!)
head(hravgs)
```

```
## # A tibble: 6 x 3
## # Groups:   su_id [1]
##   su_id    hr            hract
##   <chr>    <chr>         <dbl>
## 1 10000100 2011-01-13 10  25.5
## 2 10000100 2011-01-13 11  16.3
## 3 10000100 2011-01-13 12  29.4
## 4 10000100 2011-01-13 13  50.2
## 5 10000100 2011-01-13 14  25.7
## 6 10000100 2011-01-13 15  39.2
```

```r
#save for easy future use
#write.csv(hravgs, "C:/Users/Jade/Documents/Sleep Mortality/hr_avgs")
```

#Interday Stability Calculation

We now have a dataset with hourly actigraphy averages (hract) for each of the su_ids of eligible sleep study respondents. Let's create the interday stability first.

```r
#if using the already prepared data, start here
#hravgs <- read_csv("C:/Users/Jade/Documents/Sleep Mortality/hr_avgs")

#Create data frame with IS values
IS_table <- data.table(su_id = unique(hravgs$su_id),
                       n = 0,
                       mean_m = 0,
                       var_m = 0,
```

```
                        is_denominator = 0,
                        is_numerator = 0,
                        Is= 0
                        )
p <- 24

for (i in unique(hravgs$su_id)){

  #create subset for each su_id
  suid_sub <- subset(hravgs, hravgs$su_id == i)

  #Even with the subsetting down to 4,320 minutes of data, there were still respondents with >72hr of d
  #I'm fixing it this way but we can think about if this is best approach and why problem is happening
  suid_sub$num_hrs <- seq.int(nrow(suid_sub))
  suid_sub <- subset(suid_sub, num_hrs <=72)

  #n - how many hours? (should all be less than 72)
  n <- length(suid_sub$hract)
  IS_table[su_id== i,2] <- n

  #mean of all hourly actigraph measures (Xbar)
  mean_m <- mean(suid_sub$hract)
  IS_table[su_id == i,3] <- mean_m

  #variance of hourly actigraph measures
  var_m <- var(suid_sub$hract)
  IS_table[su_id == i,4] <- var_m

  ##IS

    #denominator
    is_denom <- var_m * (n -1)*p
    IS_table[su_id == i,5] <- is_denom

    #numerator

      #extract just the hours (without the day)
      suid_sub$hrs <- stri_sub(suid_sub$hr, -2, -1)

      #match hours and calculate their means across the days (x_h)
      suid_sub <- suid_sub %>%
        group_by(hrs) %>%
        summarize(hrm = mean(hract))

      #mean actigraphy values by hour (x_h) - mean of all the hourly values (xbar)
      suid_sub$ns <- (suid_sub$hrm - mean_m)^2
      is_num <- n*sum(suid_sub$ns)
      IS_table[su_id == i,6] <- is_num


    #interday stability
    i_s <- is_num/is_denom
```

```
    IS_table[su_id == i,7] <- i_s


}

IS_table %>%
  skimr::skim()
```

Table 1: Data summary

Name	Piped data
Number of rows	689
Number of columns	7
Column type frequency:	
character	1
numeric	6
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
su_id	0	1	8	8	0	689	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
n	0	1	68.27	4.61	48.00	65.00	70.00	72.00	72.00	
mean_m	0	1	58.93	21.02	8.51	43.94	56.55	70.35	204.25	
var_m	0	1	2654.34	2087.88	76.90	1330.76	2222.55	3387.92	29935.19	
is_denominator	0	1	4280093.41	3116180.24	131038.25	2164346.00	3577217.00	5449055.00	13009557.24	
is_numerator	0	1	3180483.26	2616358.76	76291.08	1505543.00	2620207.00	4632848.00	13803090.28	
Is	0	1	0.73	0.13	0.24	0.65	0.74	0.82	1.06	

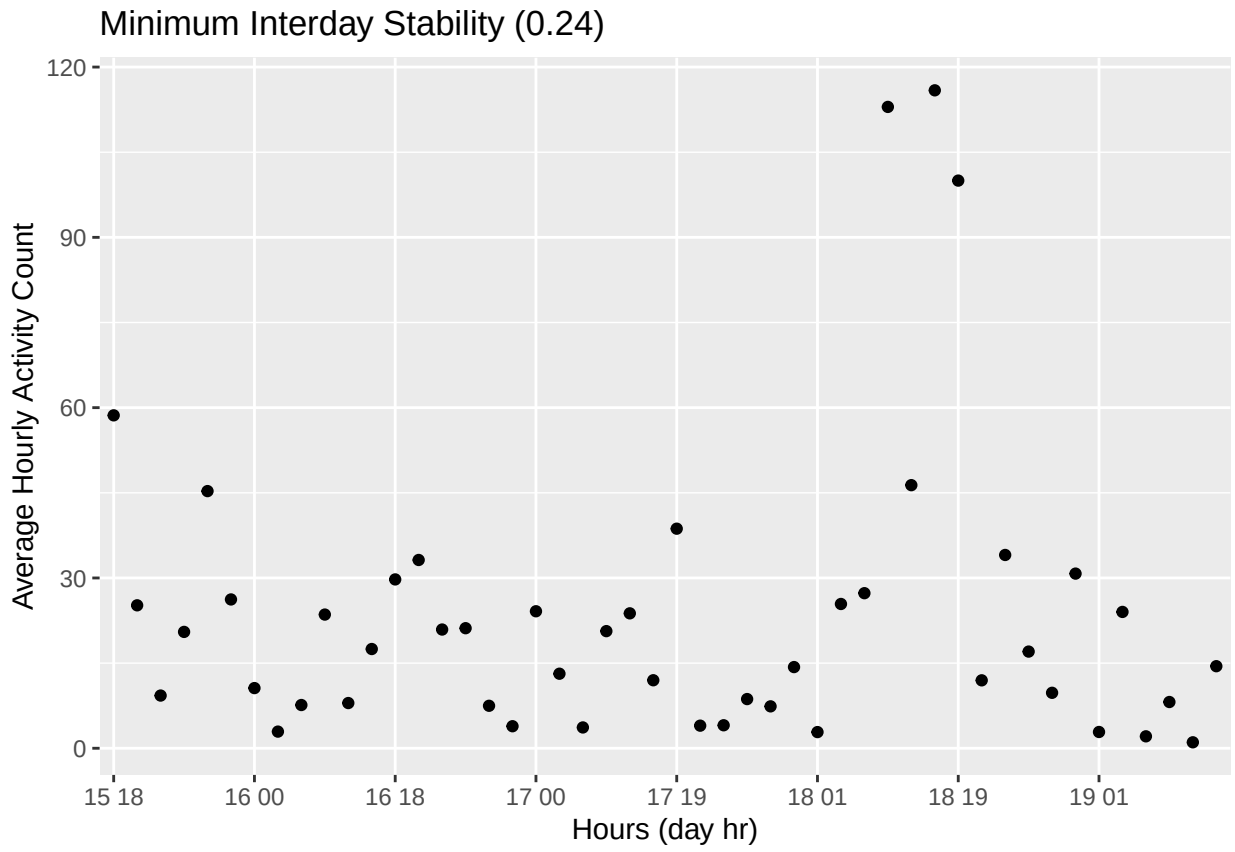
```
summary(IS_table)
```

```
##     su_id                n             mean_m           var_m
##  Length:689         Min.   :48.00   Min.   :  8.506   Min.   :   76.9
##  Class :character   1st Qu.:65.00   1st Qu.: 43.939   1st Qu.: 1330.8
##  Mode  :character   Median :70.00   Median : 56.551   Median : 2222.6
##                     Mean   :68.27   Mean   : 58.930   Mean   : 2654.3
##                     3rd Qu.:72.00   3rd Qu.: 70.349   3rd Qu.: 3387.9
##                     Max.   :72.00   Max.   :204.248   Max.   :29935.2
##  is_denominator      is_numerator            Is
##  Min.   :  131038   Min.   :   76291   Min.   :0.2402
##  1st Qu.: 2164347   1st Qu.: 1505544   1st Qu.:0.6505
##  Median : 3577218   Median : 2620208   Median :0.7424
```

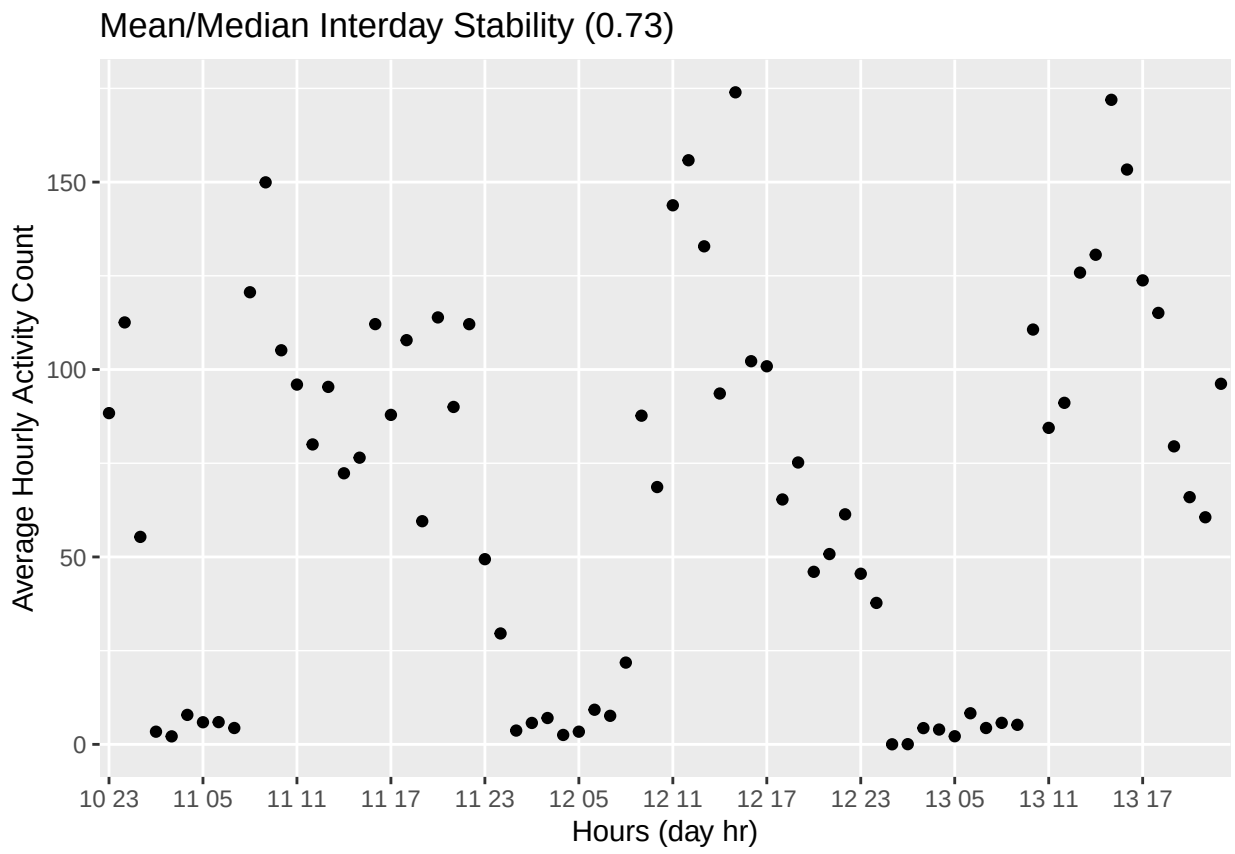
```
##  Mean   : 4280094   Mean   : 3180484   Mean   :0.7308
##  3rd Qu.: 5444906   3rd Qu.: 4132848   3rd Qu.:0.8179
##  Max.   :51009557   Max.   :38303090   Max.   :1.0564
```

I'm subsetting this dataset twice to ensure <72hr of data. I'm not so confident about the second time that's included in this loop. Without it, there are many participants with hr counts over 72 but all of the respondents only have 4,320 minutes of data so perhaps we should keep it?

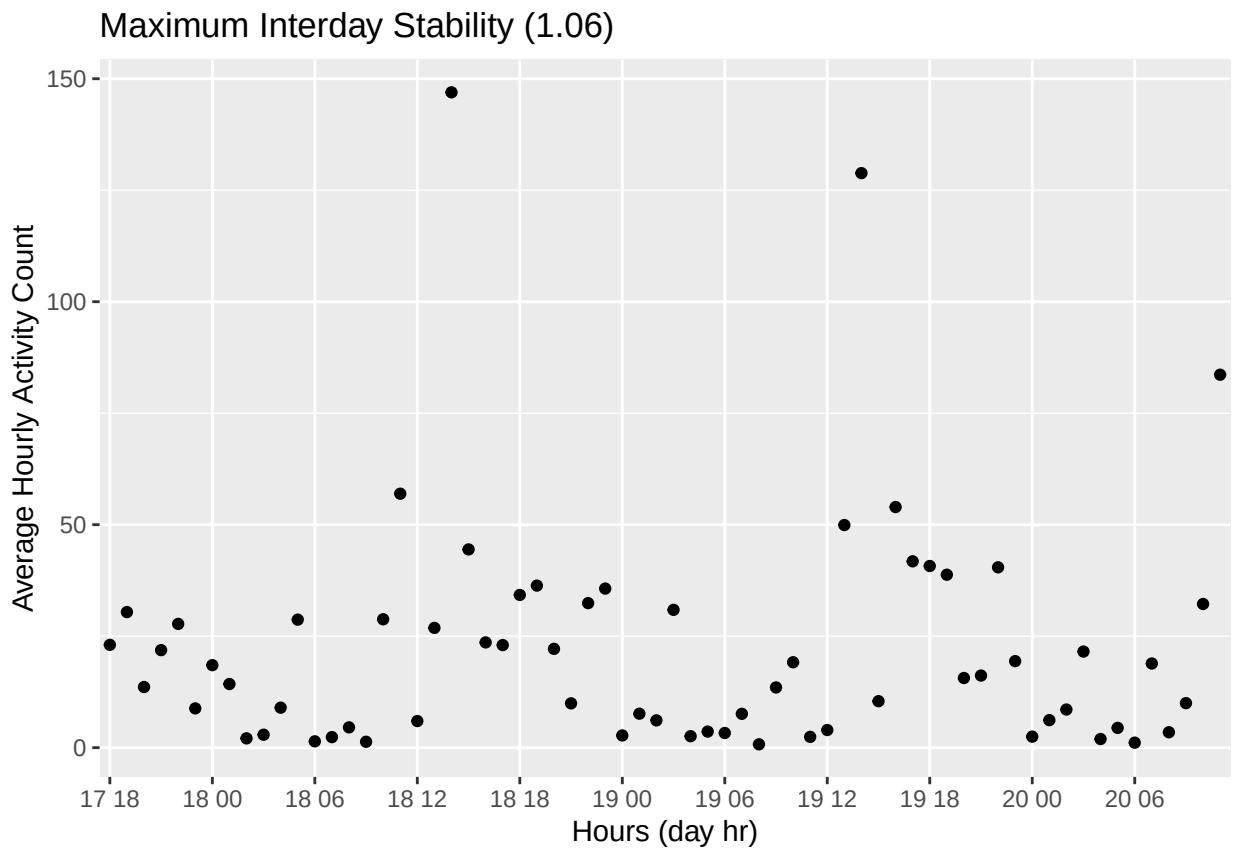
```
#makes graphing easier
plot_ISIV <- function(suid, mytitle){

  plot_subset <-hravgs %>%
  subset(su_id== suid) %>%
  mutate(hrs = stri_sub(hr, -5, -1)) %>%
  mutate(labelhr = stri_sub(hr, -2, -1) )

plot_subset %>%
  ggplot(aes(x = hrs,
             y = hract)) +
  geom_point() +
  labs(title = mytitle,
     y = "Average Hourly Activity Count",
     x = "Hours (day hr)"
     ) +
  scale_x_discrete(limits = plot_subset$hrs, breaks = plot_subset$hrs[seq(1, length(plot_subset$hrs), b

}
#minimum IS (0.24)
#SU_ID = 10037310
plot_ISIV("10037310", "Minimum Interday Stability (0.24)")
```




```
#Mean/Median IS (0.73)
#SU_ID  10005080
plot_ISIV("10005080", "Mean/Median Interday Stability (0.73)")
```




```
#Max IS (1.06)
#SU_ID 10004150
plot_ISIV("10004150", "Maximum Interday Stability (1.06)")
```



#Intraday Variability

```
##Create dataframe with IV values
IV_table <- data.table(su_id = unique(hravgs$su_id),
                       n = 0,
                       mean_m = 0,
                       var_m = 0,
                       iv_denominator = 0,
                       iv_numerator = 0,
                       IV = 0
)
p <- 24


for (i in unique(hravgs$su_id)){

  #subset all su_ids
  suid_sub <- subset(hravgs, hravgs$su_id== i)

  #total number of hours - segment to 72
  #again, still not very confident about this step
  suid_sub$num_hrs <- seq.int(nrow(suid_sub))
  suid_sub <- subset(suid_sub, num_hrs <=72)

  #n - how many hours in the subset? AFTER subset
  n <- length(suid_sub$hract)
```

```
  IV_table[su_id == i,2] <- n

  #mean for all the hourly actigraph values
  mean_m <- mean(suid_sub$hract)
  IV_table[su_id == i,3] <- mean_m

  #variance for all the hourly actigraph values
  var_m <- var(suid_sub$hract)
  IV_table[su_id == i,4] <- var_m


  ##IV
    #denominator
    iv_denom <- var_m * (n-1)^2
    IV_table[su_id == i,5] <- iv_denom

    #numerator
      #create lagged values (xi-1)
      suid_sub$lag <- dplyr::lag(suid_sub$hract)

      #subtract lagged values from xi and square them
      suid_sub$lagss <- (suid_sub$hract - suid_sub$lag)^2

      #sum the xi - xi-1 and multiply by n
      iv_num <- n*sum(suid_sub$lagss[2:n])
      IV_table[su_id == i,6] <- iv_num

      #intraday variability
      i_v <- iv_num/iv_denom
      IV_table[su_id == i,7] <- i_v



}

IV_table %>%
  skimr::skim()
```

Table 4: Data summary

Name	Piped data
Number of rows	689
Number of columns	7
Column type frequency:	
character	1
numeric	6
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
su_id	0	1	8	8	0	689	0

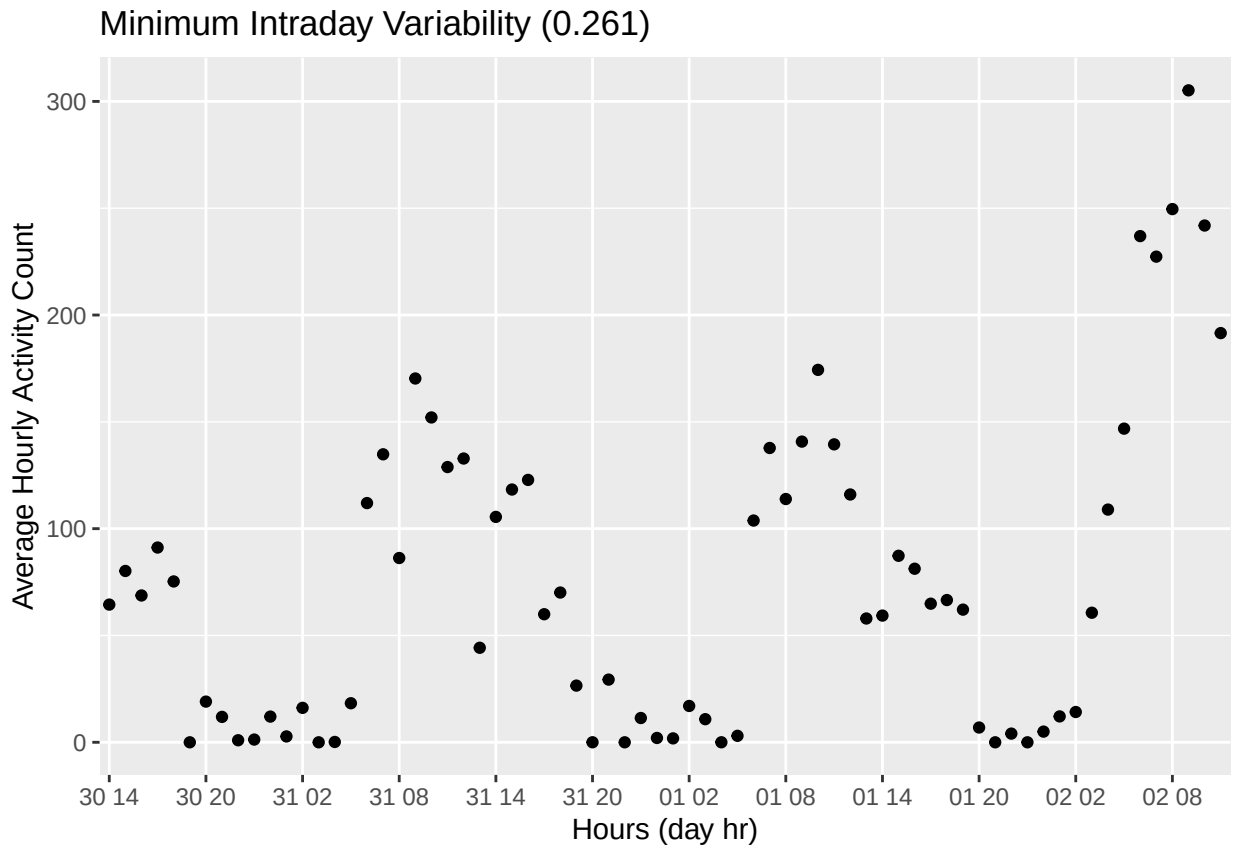
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
n	0	1	68.27	4.61	48.00	65.00	70.00	72.00	72.00	
mean_m	0	1	58.93	21.02	8.51	43.94	56.55	70.35	204.25	
var_m	0	1	2654.34	2087.88	76.90	1330.76	2222.55	3387.92	29935.19	
iv_denominator	0	1	12042651	9857071	387655	6121128	9849417	15556617	150903273.49	
iv_numerator	0	1	8638038	7365018	572344.48	4803245	7170754	10441947	81382125.81	
IV	0	1	0.79	0.27	0.26	0.60	0.75	0.95	2.09	

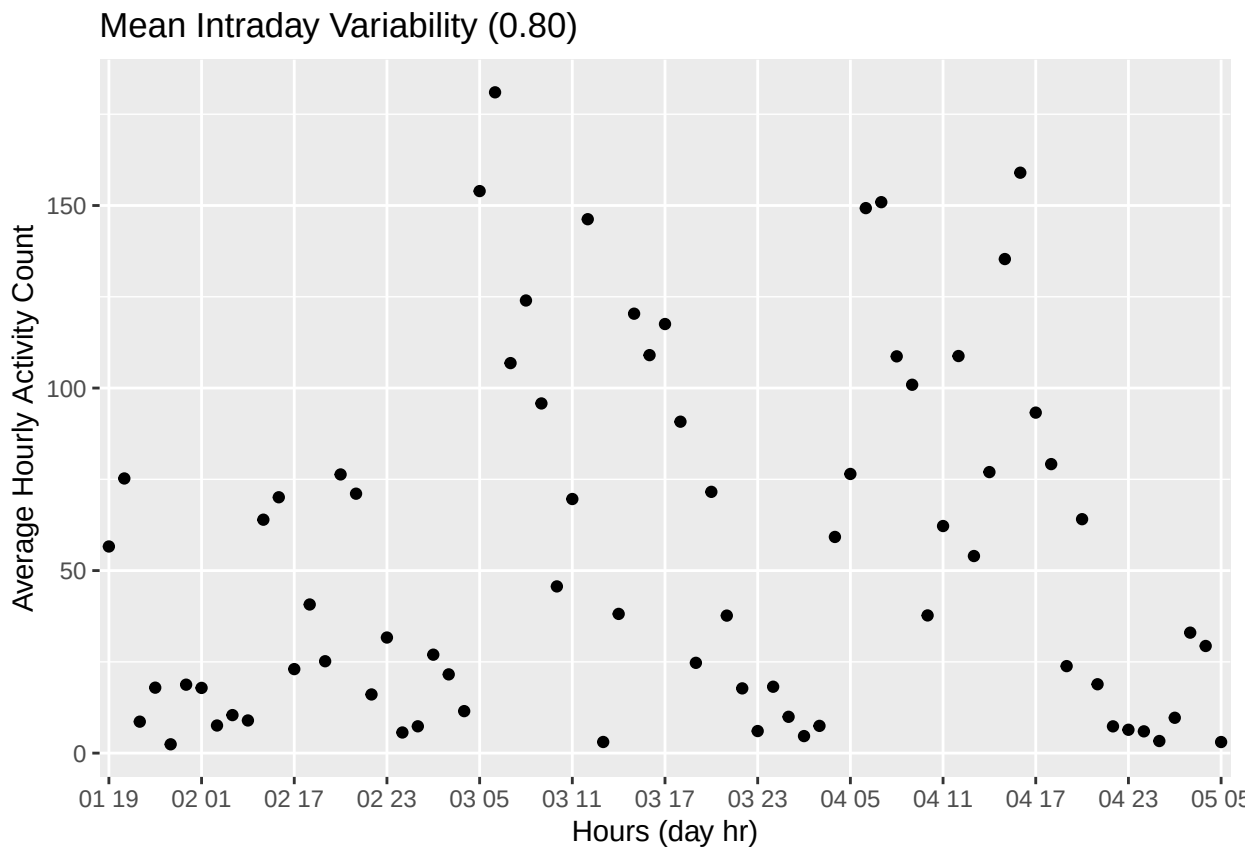
```
summary(IV_table)
```

```
##     su_id                n             mean_m           var_m
##  Length:689         Min.   :48.00   Min.   :  8.506   Min.   :   76.9
##  Class :character   1st Qu.:65.00   1st Qu.: 43.939   1st Qu.: 1330.8
##  Mode  :character   Median :70.00   Median : 56.551   Median : 2222.6
##                     Mean   :68.27   Mean   : 58.930   Mean   : 2654.3
##                     3rd Qu.:72.00   3rd Qu.: 70.349   3rd Qu.: 3387.9
##                     Max.   :72.00   Max.   :204.248   Max.   :29935.2
##  iv_denominator       iv_numerator            IV
##  Min.   :   387655   Min.   :  572345   Min.   :0.2607
##  1st Qu.:  6121128   1st Qu.: 4803245   1st Qu.:0.6002
##  Median :  9849417   Median : 7170754   Median :0.7481
##  Mean   : 12042651   Mean   : 8638039   Mean   :0.7948
##  3rd Qu.: 15556617   3rd Qu.:10441947   3rd Qu.:0.9463
##  Max.   :150903273   Max.   :81382126   Max.   :2.0906
```

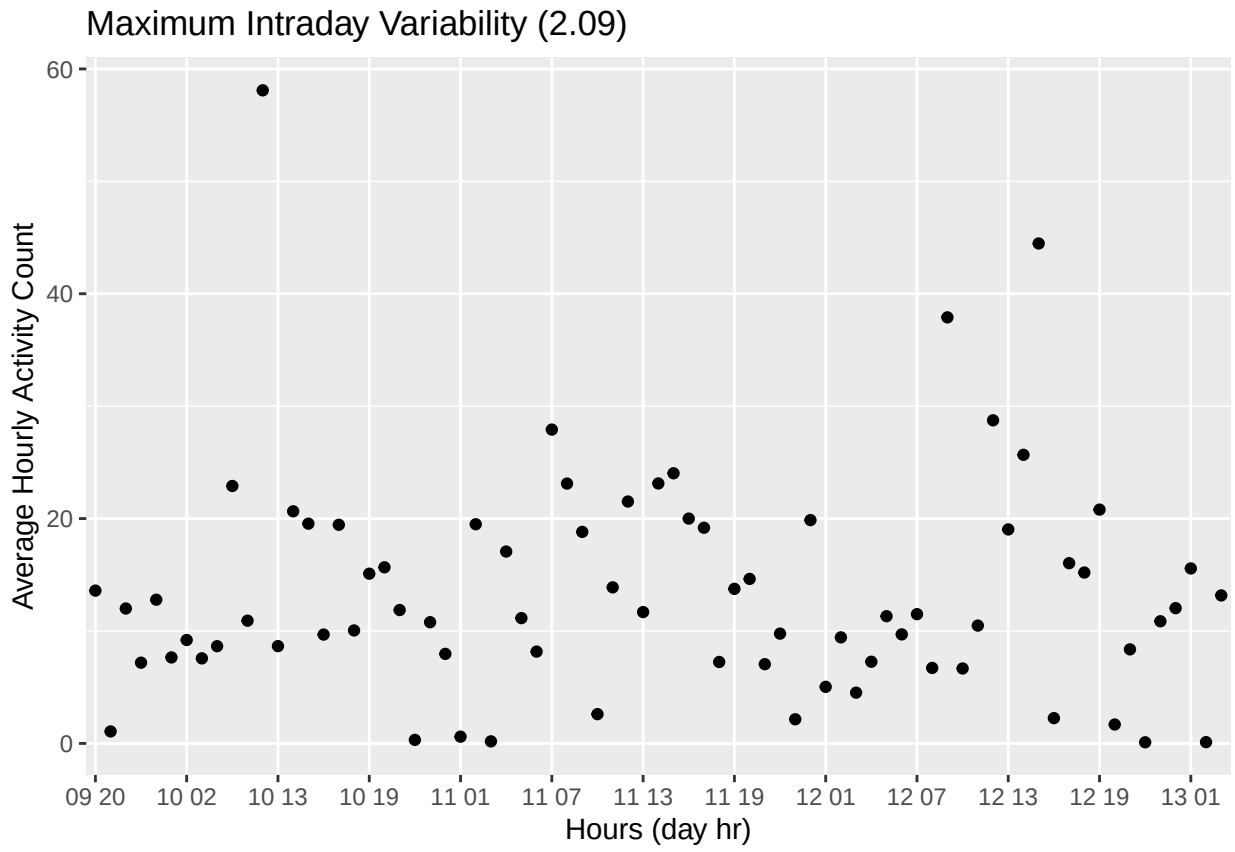
```
#minimum IV (0.261)
#suid 10007110
plot_ISIV("10007110", "Minimum Intraday Variability (0.261)")
```




```
#Mean IV (0.80)
#suid   10001170
plot_ISIV("10001170", "Mean Intraday Variability (0.80)" )
```




```
#Maximum IV (2.086)
#suid 10009310
plot_ISIV("10009310", "Maximum Intraday Variability (2.09)")
```




```
IS_IV <- data.table(su_id = unique(hravgs$su_id),
                    IS = IS_table$Is,
                    IV = IV_table$IV)

IS_IV %>%
  skimr::skim()
```

Table 7: Data summary

Name	Piped data
Number of rows	689
Number of columns	3
Column type frequency:	
character	1
numeric	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
su_id	0	1	8	8	0	689	0

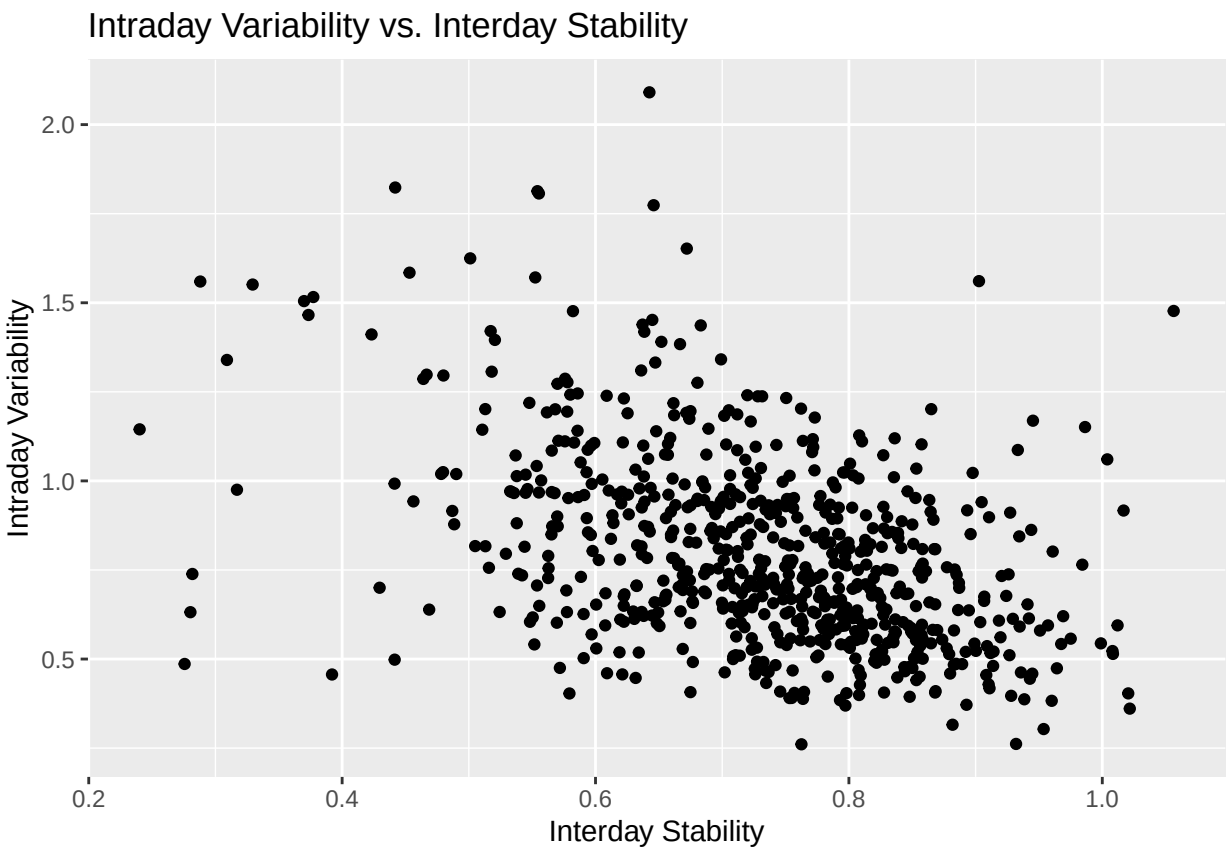
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
IS	0	1	0.73	0.13	0.24	0.65	0.74	0.82	1.06	
IV	0	1	0.79	0.27	0.26	0.60	0.75	0.95	2.09	

```
#correlation between the two
cor(IS_IV$IS, IS_IV$IV)
```

```
## [1] -0.436782
```

```
IS_IV %>%
  ggplot(aes(x = IS,
             y = IV)) +
  geom_point() +
  labs(title = "Intraday Variability vs. Interday Stability",
     y = "Intraday Variability",
     x = "Interday Stability"
     )
```



```
#write this to dta to use later and to send to Elena
write.dta(IS_IV, "C:/Users/Jade/Documents/Sleep Mortality/IS_IV_72.dta" )
```