RESEARCH DATA CENTRE (FDZ)
of the German Federal Employment Agency (BA)
at the Institute for Employment Research (IAB)

BERD Academy

BERD @NFDI

HOW TO TIDY AND ANONYMIZE RAW SMARTPHONE GEOLOCATION DATA

CODE AND PRACTITIONER'S EXAMPLES FROM THE IAB-SMART PROJECT

Andreas Filser
Research Data Centre
Institute for Employment Research (FDZ-IAB)

IAB-SMART STUDY

- Field time: January 4th – August 31st 2018
  - Proprietary *Android* app active from January 8th to August 31st 2018
  - Participants recruited from PASS (Panel Arbeitsmarkt und Soziale Sicherung)
  - 618 participants of 4,293 recruitees installed the app at least once
- IAB-SMART Geolocation records
  - Aimed to collect geolocation every 30 minutes

IAB-SMART-MOBILITY: MOBILITY INDICATORS FROM GEOLOCATION DATA

Part of IAB-SMART by

Module publication funded by

Module to be released this month (probably)

Subscribe to IAB-FDZ newsletter for updates

IAB MOBILITY

Weekday	Weekend	Total
Median visited locations	Median visited locations	Total visited locations
Median distance	Median distance	Total distance
Median variance geolocations	Median variance geolocations	Total variance geolocations
Share geolocations home cluster during daytime (6am to midnight)	Share geolocations home cluster during daytime (6am to midnight)	
Median visited locations	Median visited locations	

PREPARATION STEPS

Verify that smartphone users are PASS participants

1. Structure the raw geodata (into 30-minute intervals)
2. Drop geolocation data with errors and inaccurate data
   - Pitfalls, peculiarities
   - Find gaps and missing data
3. Sample definition: Minimum number of records per day …
4. Construct indicators using the final sample

LITERATURE & BASIC POINTS

- Müller et al. (2022). Analyzing GPS data for psychological research: A tutorial. *Advances in Methods and Practices in Psychological Science, 5*(2), 25152459221082680.
- Jongs et al. (2020). A framework for assessing neuropsychiatric phenotypes by using smartphone-based location data. *Translational psychiatry, 10*(1), 211.
- Saeb et al. (2015). Mobile phone sensor correlates of depressive symptom severity in daily-life behavior: an exploratory study. *Journal of medical Internet research, 17*(7), e4273.


Tutorial

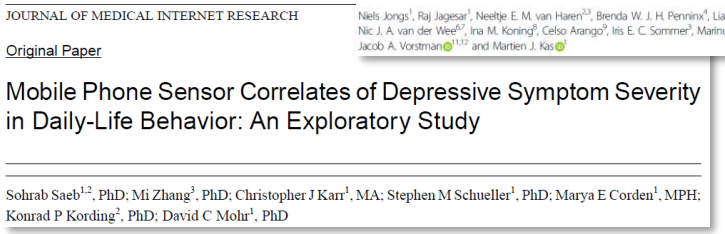
Analyzing GPS Data for Psychological Research: A Tutorial

Sandrine R. Müller, Joseph B. Bayer, Morgan Quinn Ross, Jerry Mount, Clemens Stachl, Gabriella M. Harari, Yung-Ju Chang, and Huyen T. K. Le


Translational Psychiatry

ARTICLE Open Access

A framework for assessing neuropsychiatric phenotypes by using smartphone-based location data


JOURNAL OF MEDICAL INTERNET RESEARCH

Original Paper

Mobile Phone Sensor Correlates of Depressive Symptom Severity in Daily-Life Behavior: An Exploratory Study

Sohrab Saeb, PhD; Mi Zhang, PhD; Christopher J Karr, MA; Stephen M Schueller, PhD; Marya E Corden, MPH; Konrad P Kording, PhD; David C Mohr, PhD

- **{tidyverse}** approach

- **{dtplyr}**: **{data.table}** speed and **{tidyverse}** syntax consistency

{DTPLYR}

```
> library(dtplyr)
> geo_dt <- lazy_dt(geo_full)

> class(geo_dt)
[1] "dtplyr_step_first" "dtplyr_step"

> geo_dt %>% count(na_lat=is.na(lat),na_lon= is.na(lon))
Source: local data table [2 x 3]
Call:   copy(`_DT2`)[, `:=`(na_lat = is.na(lat), na_lon = is.na(lon))][,
    .(n = .N), keyby = .(na_lat, na_lon)]

  na_lat na_lon       n
  <lgl>  <lgl>    <int>
1 FALSE  FALSE  2152086
2 TRUE   TRUE   4854869

# Use as.data.table()/as.data.frame()/as_tibble() to access results
```

AMPPS R Markdown FINAL.Rmd

Knit on Save | Knit | Run

Source | Visual | Outline

```
# Average / sd number of GPS records
rows_george <- nrow(devicedata[devicedata$Name == "George",])
rows_jerry <- nrow(devicedata[devicedata$Name == "Jerry",])
rows_joe <- nrow(devicedata[devicedata$Name == "Joe",])
rows_josephine <- nrow(devicedata[devicedata$Name == "Josephine",])

mean(c(rows_george,rows_jerry,rows_joe,rows_josephine)) # 122169.8
sd(c(rows_george,rows_jerry,rows_joe,rows_josephine)) # .5
```

Data Pre-processing

Exclusions / Data Cleaning

First, we want to ensure data quality. We will remove the points that have an accuracy of 100 or above (meaning that there is a 68% chance that the true location is within 100 meters of the recorded location).

```{r inaccurate}
# Remove data with accuracy >100 meters
devicedata <- devicedata[devicedata$acc<100,]
```

We then drop the points with missing latitude/longitude or time.

```{r missing}
# Drop locations without coordinates or timestamps
devicedata <- devicedata[!(is.na(devicedata$lat)), ]
devicedata <- devicedata[!(is.na(devicedata$timestamp)), ]
```

There might be duplicates points We will delete them as well.

```{r duplicates}
# Remove duplicate recordings
devicedata <- distinct(devicedata,userID,timestamp,.keep_all=TRUE)
```

Converting Time

By default, smartphone apps and GPS loggers generate data with epoch time. We will convert epoch time to date-time format. Here we use US Eastern time as an example. You can type "?with_tz" for more information about changing time zones.

```{r time format/zone}
# Reformat time from epoch time (seconds since 1/1/1970) to datetime (day / month / year, hour / minute / second)
```

Table of contents:

- Tutorial
- Setup
  - Packages
  - Load Data
- Data Pre-processing
  - Exclusions / Data Clea…
  - Converting Time
  - Data Quality Dataframe
  - Study Area Boundary
  - Hour and Day Indices
- Key Locations
  - Clustering
  - Home
- Mobility Features
  - Total Distance
  - Unique Locations
  - Places Visited
  - Time Spent at Key Loc…
  - Time Spent at Home
  - Travel Time
  - Entropy
  - Routine Index
  - Summary
- API Pull
  - ggmap
  - googleway
- References

1:1 | AMPPS TUTORIAL | R Markdown

Müller et al. (2022). Analyzing GPS data for psychological research: A tutorial.

SQL QUERY: {ODBC}

```
library(odbc)

con <- dbConnect(odbc(),
                 Driver = "SQL Server",
                 Server = "N2915017",
                 Database = "SMART",
                 UID = "Your Username",
                 Trusted_Connection = "True",
                 Port = 1433 )

# get location data
geo_raw <- dbGetQuery(con,'
  SELECT
        LocationInfoOnEnd_LocationLatitude  as lat,
        LocationInfoOnEnd_LocationLongitude as long,
        LocationInfoOnEnd_LocationAccuracyVertical as accu,
        TimeInfoOnEnd_TimestampTableau as time,
        LocationInfoOnEnd_LocationProvider as prov,
        Code as regcode
        FROM ltr
                  ')
```

RAW GEOLOCATION DATA

ID	Date	Time	LAT	LON	Accuracy
23	25.02.2018	13:46:31	0,0000	0,0000	0
23	25.02.2018	14:23:34	49,4421	11,1053	50
23	25.02.2018	14:51:41	49,4206	11,1142	150
23	25.02.2018	15:14:04	49,4574	11,0744	20
23	...	...	...	...	...
23	...	...	...	...	...
23	...	...	...	...	...
23	...	...	...	...	...
23	30.08.2018	12:58:13	49,4206	11,1142	1000
23	30.08.2018	13:22:25	49,4574	11,0744	50

Variation in

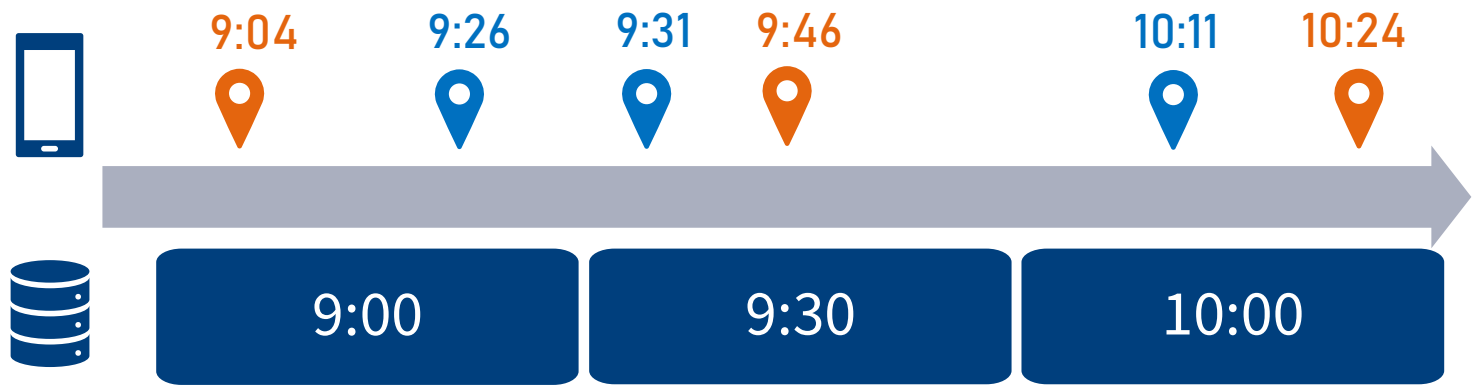
- App usage/installation periods
- Time stamps for location records
- Accuracy / location collection method

➢ Structure your raw location data

GOAL: CREATE PANEL DATA WITH 48 INTERVALS PER DAY

ID	Date	Time	LAT	LON	Accuracy	Std_Time	sample_lab
23	25.02.2018					00:00	pre installation
23	...					...	...
23	25.02.2018					12:30	first installation
23	25.02.2018					13:00	installed, no location
23	25.02.2018	13:46:31	0,0000	0,0000	0	13:30	invalid location
23	25.02.2018	14:23:34	49,4421	11,1053	50	14:00	in sample
23	25.02.2018	14:51:41	49,4206	11,1142	150	14:30	in sample
23	25.02.2018	15:14:04	49,4574	11,0744	1000	15:00	too imprecise
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	30.08.2018	12:58:13	49,4206	11,1142	150	12:30	less than 8 hours/day
23	30.08.2018	13:22:25	49,4574	11,0744	50	13:00	less than 8 hours/day
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	31.08.2018					13:30	after last signal

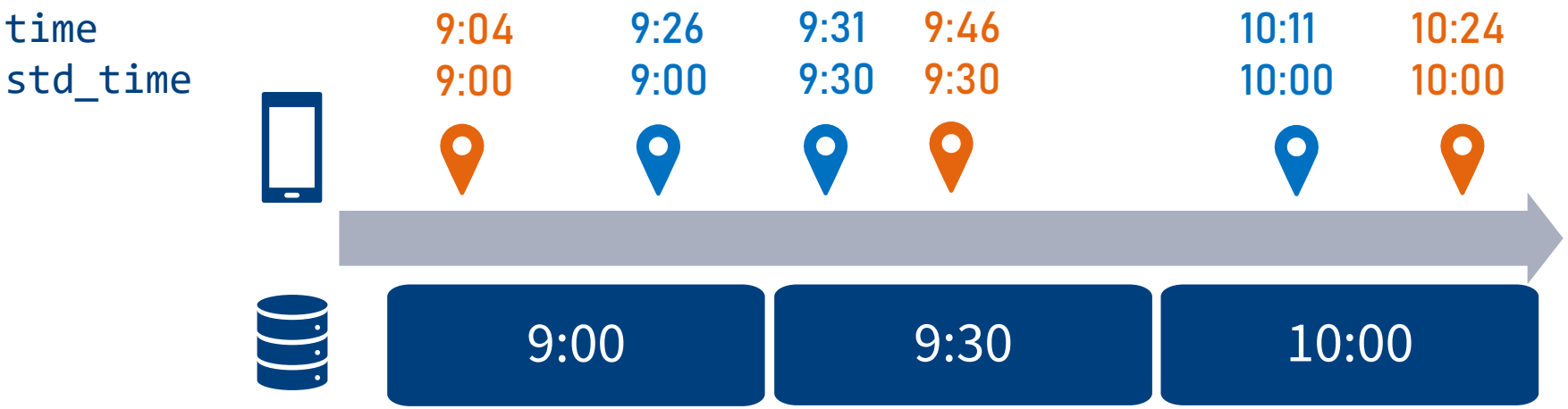
CREATE PANEL DATA WITH 48 INTERVALS PER DAY




```
date_panel <-
  correct_df %>% # data.frame with verified participants
  mutate(slots = interval(ymd("2018-1-8"), ymd("2018-9-01"))/ days(1) * 48 ) %>%
  uncount(slots,.id = "slot") %>%
  group_by(id) %>%
  mutate( std_time = ymd("2018-1-8") + dhours(.5*(slot-1)) ) %>%
  ungroup()
```

STANDARDIZE RAW TIME IN GEOLOCATION DATA

```
std_time = lubridate::floor_date(time, "30 mins")
```




```
date_panel %>% left_join(geo_raw, by = c("id","std_time"))
```

7,006,955 rows
≈60MB

MEASUREMENT ACCURACY & MISSINGS

Provider	Min	Median	75%	85%	90%	Max	N
GPS	0.0	13.0	28.0	45.0	51.0	732.0	223,254
Network	1.0	20.7	34.4	98.4	800.0	10,718,928.0	1,200,289
Fused	2.4	4,898.4	10,014.4	13,804.8	75,130.9	10,718,928.0	46,174
Total	0.0	20.6	36.0	96.1	900.0	10,718,928.0	1,469,717

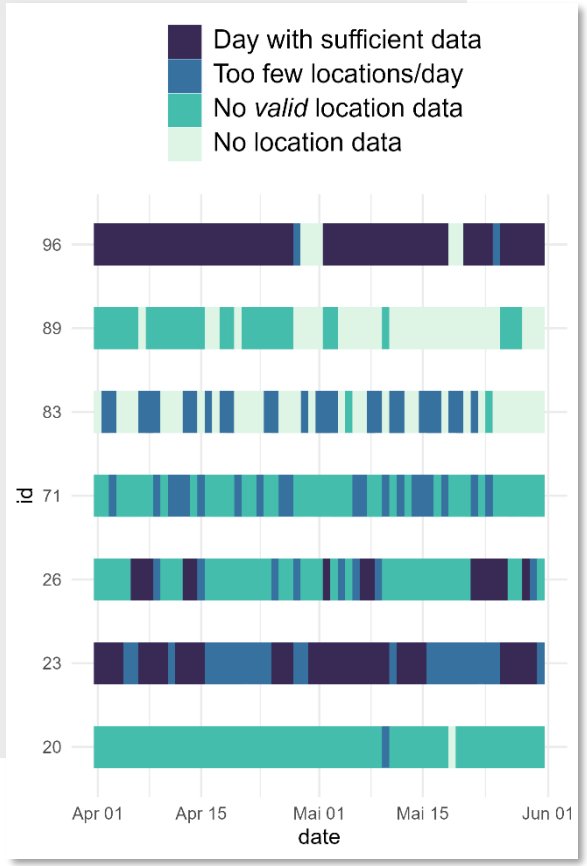
Step	Persons	% Persons	Geodata	% Geodata
Installed and verified	618	100 %		
Geodata consent	567	92 %		
Valid geolocation measurements	543	88 %	1,465,541	100%
Accurate geolocation measurements (< 350m)	543	88 %	1,288,050	88%
At least seven valid days	398	64 %	1,218,715	83%

PANEL DATA WITH 48 INTERVALS PER DAY

ID	Date	Time	LAT	LON	Accuracy	Std_Time	sample_lab
23	25.02.2018					00:00	pre installation
23	...					...	...
23	25.02.2018					12:30	first installation
23	25.02.2018					13:00	installed, no location
23	25.02.2018	13:46:31	0,0000	0,0000	0	13:30	invalid location
23	25.02.2018	14:23:34	49,4421	11,1053	50	14:00	in sample
23	25.02.2018	14:51:41	49,4206	11,1142	150	14:30	in sample
23	25.02.2018	15:14:04	49,4574	11,0744	1000	15:00	too imprecise
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	30.08.2018	12:58:13	49,4206	11,1142	150	12:30	less than 8 hours/day
23	30.08.2018	13:22:25	49,4574	11,0744	50	13:00	less than 8 hours/day
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
23	31.08.2018					13:30	after last signal

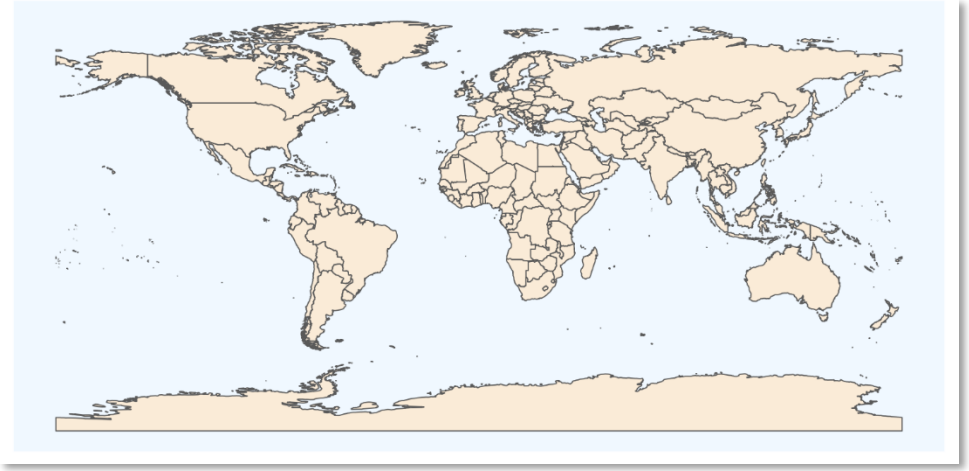
PLOT YOUR DATA PART: PATTERNS OF MISSINGS/LOCATION DATA

```
geo_dt %>%
  distinct(id,date,sample_lab) %>%
  ggplot(aes(y = id,x=date, fill = sample_lab)) +
  geom_tile(height = .5) +
  scale_fill_viridis_d(option = "mako",begin = .2) +
  theme_minimal()
```



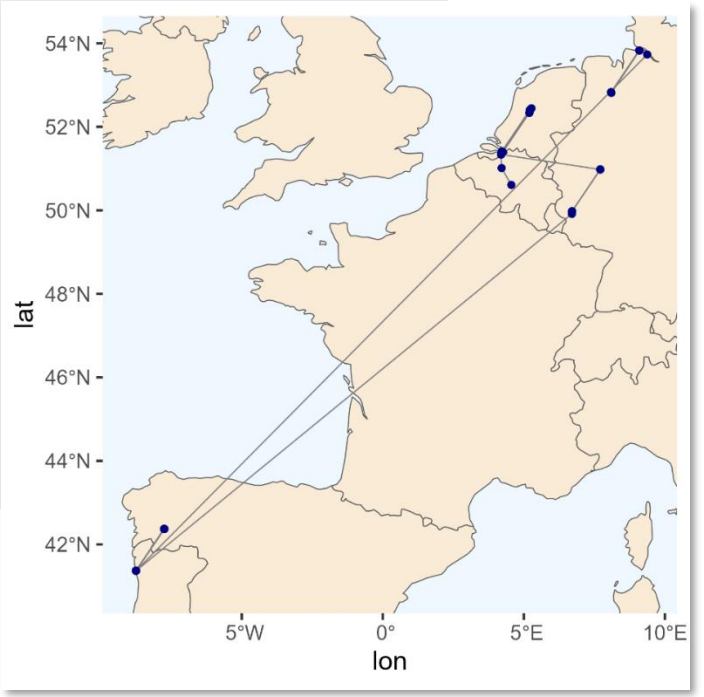
PLOT YOUR DATA 2: QUICK & EASY MAPPING USING {NATURALEARTH}

```
library("rnaturalearth")

world <- ne_countries(scale = "medium", returnclass = "sf")

ggplot(data = world) +
  geom_sf(fill = "antiquewhite") +
  theme(panel.grid = element_blank(),
        panel.background = element_rect(fill = "aliceblue"))
```



PLOT YOUR DATA 2: QUICK & EASY MAPPING USING {NATURALEARTH}

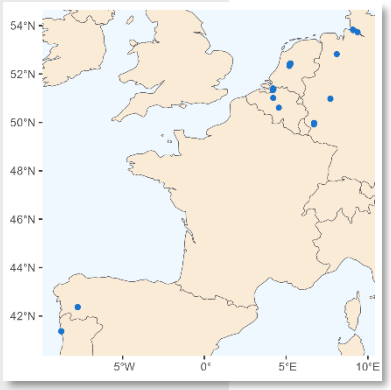
```
geo_dt_id99 <- geo_dt %>% filter(id==99)

ggplot(data = world) +
  geom_sf(fill = "antiquewhite") +
  theme(panel.grid = element_blank(),
        panel.background = element_rect(fill = "aliceblue")) +
  geom_segment(data = geo_dt_id99,
               aes(x=lon, y=lat,
                   yend = lag(lat,1),
                   xend = lag(lon,1)),
               color = "grey50", linewidth = .25 ) +
  geom_point(data = geo_dt_id99,
             aes(x=lon,y=lat),size = .95,
             color = "navy") +
  coord_sf(xlim = c( -9,9.5),
           ylim = c( 41,54))
```

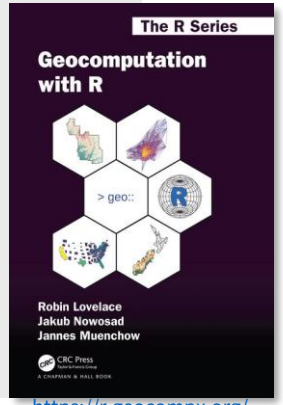


⚠ Approach ignores projection

PLOT YOUR DATA 2: NOT SO EASY MAPPING USING PROJECTION

```
library(sf)

# Set WGS84 (crs = 4326) projection
geodata <- st_as_sf(geo_dt, coords = c("lon", "lat"), crs = 4326)
world_proj <- st_transform(world, crs = 4326)

ggplot(data = world_proj) +
  geom_sf(fill = "antiquewhite") +
  geom_sf(data = geodata, color = "dodgerblue3") +
  theme(panel.grid = element_blank(),
        panel.background = element_rect(fill = "aliceblue")) +
  coord_sf(xlim = c( -9,9.5),
           ylim = c( 41, 54))
```





https://r.geocompx.org/

CLUSTERING GEOLOCATION MEASUREMENTS

```
# Function for clustering
db2 <- function(x) {
  geodata <- x %>% st_coordinates()
  cluster = dbscan::dbscan(geodata, eps = 600, minPts = 2)$cluster
  return (data.frame(cluster))
}

# Apply function to cluster points
geodata_cluster_df <-
  geodata %>%
  group_by(userID) %>%
  group_modify(~db2(.x)) %>%
  ungroup()
```

Tutorial

aps ASSOCIATION FOR PSYCHOLOGICAL SCIENCE

Analyzing GPS Data for Psychological Research: A Tutorial

Sandrine R. Müller[1], Joseph B. Bayer[2,3], Morgan Quinn Ross[2], Jerry Mount[4], Clemens Stachl[5], Gabriella M. Harari[6], Yung-Ju Chang[7], and Huyen T. K. Le[8]

Advances in Methods and Practices in Psychological Science
April-June 2022, Vol. 5, No. 2, pp. 1–11

sagepub.com/journals-permissions
DOI: 10.1177/25152459221082680
www.psychologicalscience.org/AMPPS
SAGE

CALCUATE DISTANCES

```
# group by participants & nest data
geo_dist_prep <- geo_dt %>%
                  select(regcode,lat,lon) %>%
                  rename(latitude = lat, longitude = lon) %>%
                  group_by(regcode) %>%
                  nest()

distance_matrix <-
      geo_dist_prep$data %>%
      future_map(~geodist::geodist(.x, measure = "haversine"))
```

SUMMARY

- Structure your data
- Plot your data and check for inconsistencies
- Use the great tutorials by Müller et al, Geocomputation with R

- Have fun!

CLUSTERING GEOLOCATION MEASUREMENTS

- Home
  - Cluster where a person is most frequently between 12 a.m. and 6 a.m.
  - Additional restriction on number of nights with measurements

- Unique Locations
  - Number of clusters per day…

Weekday	Weekend	Total
Median visited locations	Median visited locations	Total visited locations
Median distance	Median distance	Total distance
Median variance geolocations	Median variance geolocations	Total variance geolocations
Share geolocations home cluster during daytime (6am to midnight)	Share geolocations home cluster during daytime (6am to midnight)	
Median visited locations	Median visited locations	

Funded by

BERD @NFDI

Special thank you to Sebastian Bähr and Georg Haas for sharing their knowledge on IAB-SMART

THANK YOU!

andreas.filser2@iab.de

IAB FDZ RESEARCH DATA CENTRE (FDZ)
of the German Federal Employment Agency (BA)
at the Institute for Employment Research (IAB)

IAB INSTITUT FÜR ARBEITSMARKT- UND BERUFSFORSCHUNG
Die Forschungseinrichtung der Bundesagentur für Arbeit

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

IN OMNIBUS VERITAS UNIVERSITÄT MANNHEIM