# Example Use

Thomas Gaertner

4/13/2021

## Introduction

This R Markdown document shows the usage of the package `cino1`.

```
# Install local package
install.packages("~/Documents/Masterarbeit/Package/nofone/", repos = NULL, type="source")
```

```
## Installing package into '/home/thgaertner/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```
## Warning in install.packages("~/Documents/Masterarbeit/Package/nofone/", :
## installation of package '/home/thgaertner/Documents/Masterarbeit/Package/
## nofone/' had non-zero exit status
```

```
# load package
library(cinof1)
```

### Data

In this package, a sample data frame is included. It contains data for 300 patients within an n of 1 study. The data has the following structure:

- *patient_id*: Unique patient identifier
- *date*: Date of data points
- *day*: Day in study
- *Block*: identifies treatment block
- *Activity*: Dummy variable for steps per day
- *treatment*: Dummy variable for 2 treatments as factors
- *Uncertain_Low_Back_Pain*: Dummy variable for Uncertain loq back pain on scale 1-15

```
load("data/simpatdat.rda")
# Summarize Data
summary(simpatdat)
```

```
##    patient_id           date           day              treatment
##  Min.   : 0.00   2018-01-01:  20   Min.   :  1.00   Treatment_1:1120
##  1st Qu.: 4.75   2018-01-02:  20   1st Qu.: 28.75   Treatment_2:1120
##  Median : 9.50   2018-01-03:  20   Median : 56.50
##  Mean   : 9.50   2018-01-04:  20   Mean   : 56.50
##  3rd Qu.:14.25   2018-01-05:  20   3rd Qu.: 84.25
##  Max.   :19.00   2018-01-06:  20   Max.   :112.00
##                  (Other)   :2120
##  Uncertain_Low_Back_Pain     block         Activity
##  Min.   : 6.000          Min.   :1.00   Min.   :   45.19
##  1st Qu.: 9.000          1st Qu.:1.75   1st Qu.: 5563.32
```

```
##  Median : 9.000          Median :2.50   Median : 6910.56
##  Mean   : 9.231          Mean   :2.50   Mean   : 6943.34
##  3rd Qu.:10.000          3rd Qu.:3.25   3rd Qu.: 8290.16
##  Max.   :12.000          Max.   :4.00   Max.   :14084.38
##
```

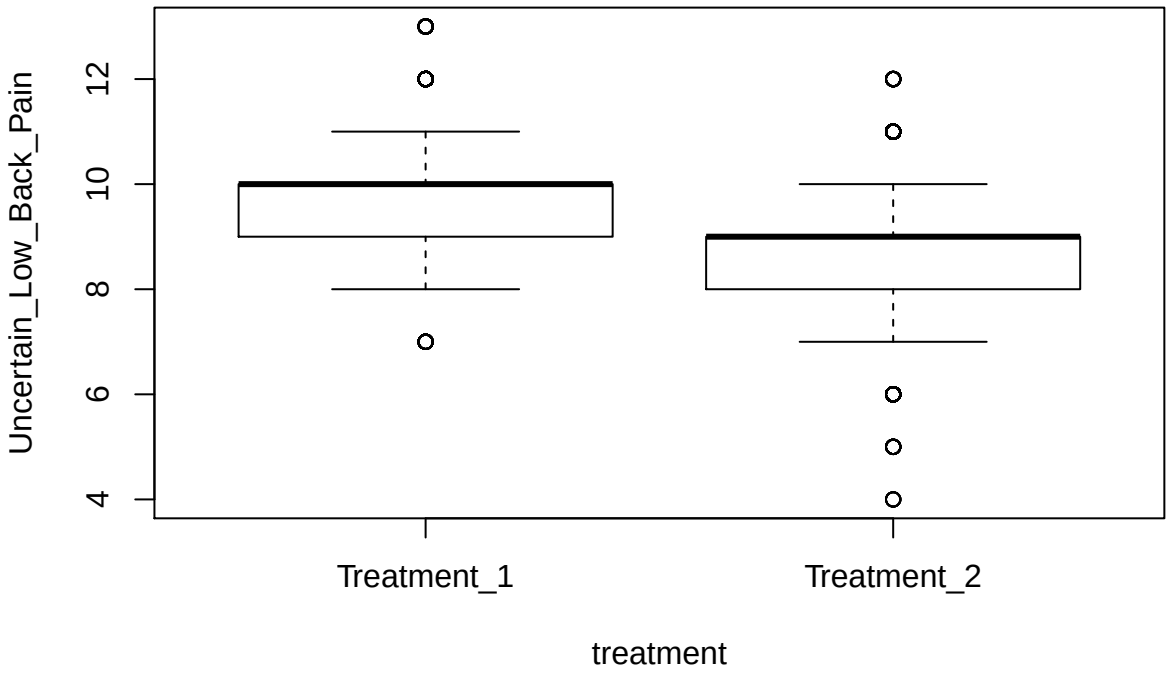
## Basic Functions

Basic functions for analyse N-of-1 studys are for example wilcox test or comparative plots. These two functions are provided in this package.

### Comparative Plot

To get a first idea about the data and the difference between treatment 1 and treatment 2, a comparative plot could be used. It shows the outcome on the y-Axis against the different treatments on the x-Axis,

```
# Define outcome and exposure column
outcome <- "Uncertain_Low_Back_Pain"
exposure <- "treatment"
# Plot outcome among different exposures
comparative.plot(simpatdat, exposure = exposure, outcome = outcome)
```




## Wilcox Test

To validate, that there is a difference in both treatments, the Wilcox test could be used. It calculates the p-value for the null hypothesis, that there location shift is equal to zero.

```r
# Define outcome and exposure column
outcome <- "Uncertain_Low_Back_Pain"
exposure <- "treatment"
# Perform Wilcox test among different exposures
wilcox.nofone(simpatdat, exposure = exposure, outcome = outcome)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Uncertain_Low_Back_Pain by treatment
## W = 1048691, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

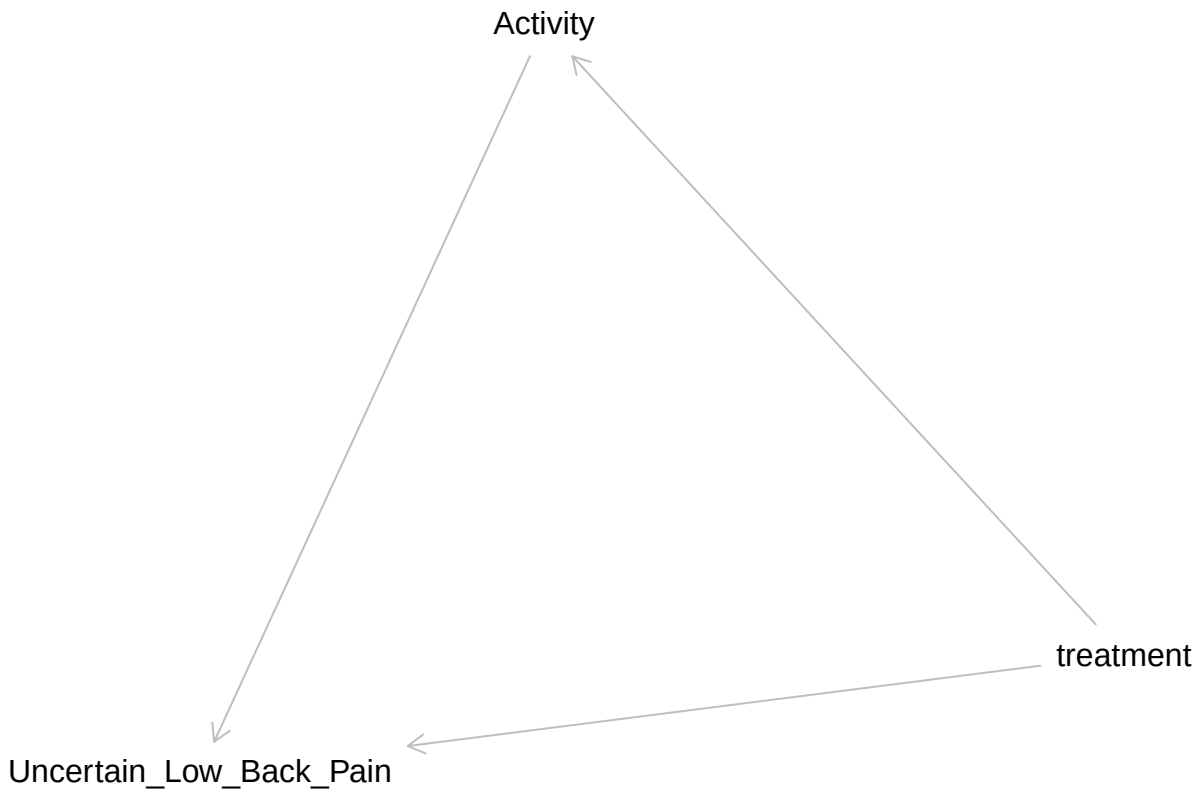
## Bayesian

Bayesian Networks are used to calculated the probability of outcome variables adjusted for confounders. For that, a dag is required, which identifies the relations between the variables. In this implementation, also lags are included and could be specified in the dag by adding `.lag=` to the variable name.

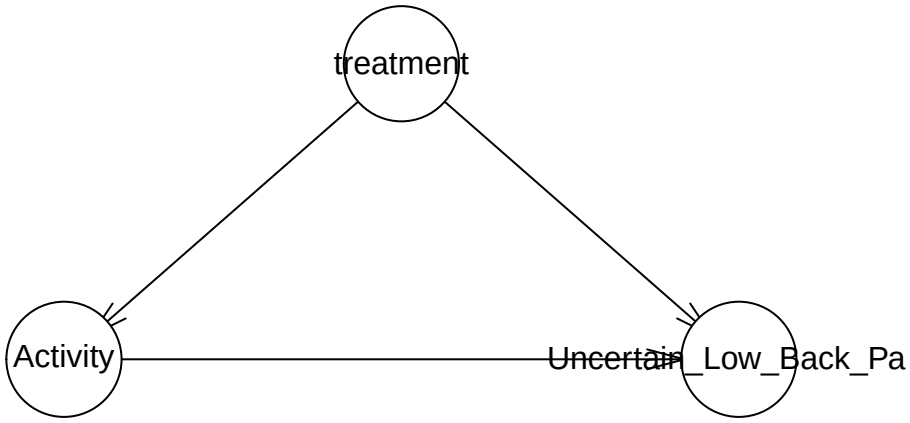
### Preprocess Data

```r
# specify column names
id <- "patient_id"
time_col <- "day"

# Load data
load("data/ex.dag.rda")
load("data/simpatdat.rda")

plot(ex.dag)
```

```
## Plot coordinates for graph not supplied! Generating coordinates, see ?coordinates for how to set you
```



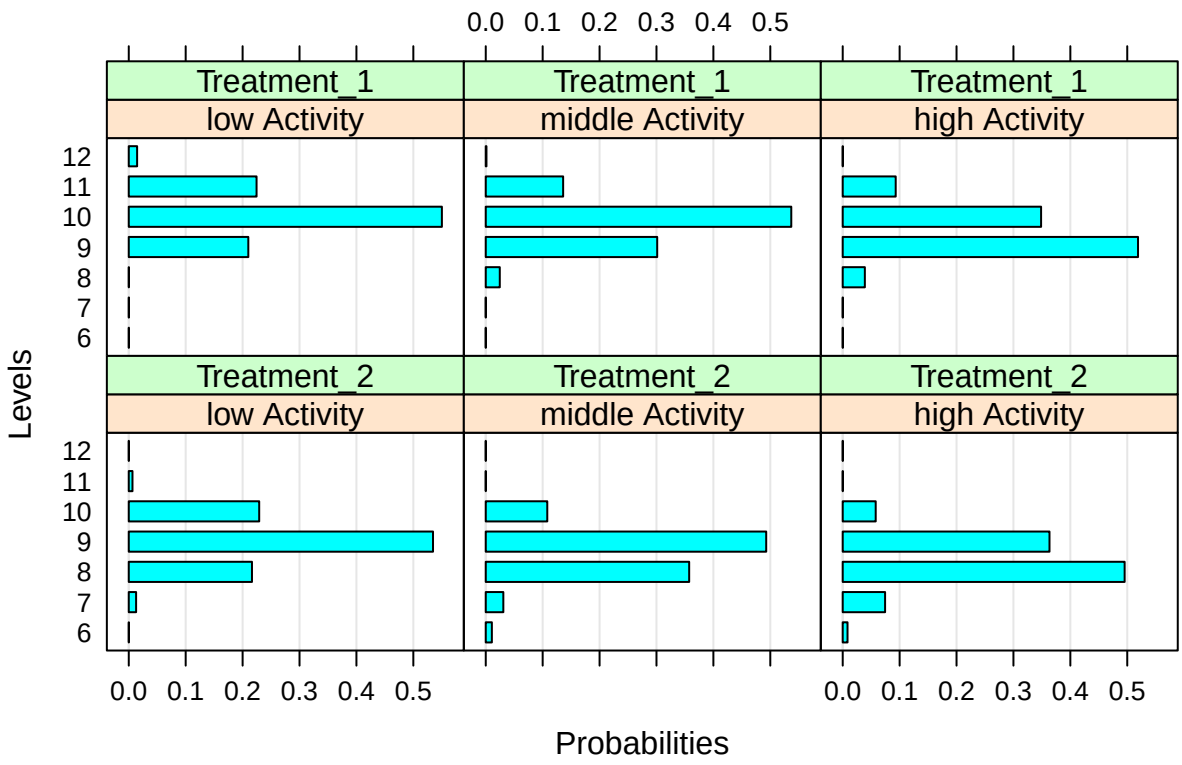

```
# Dag preprocessing
bn.dag <- bn.prep.dag(ex.dag)
```




```
# Data Preprocessing (Factorization)
simpatdat$Uncertain_Low_Back_Pain <- as.factor(simpatdat$Uncertain_Low_Back_Pain)
simpatdat$Activity <- cut(simpatdat$Activity, 3, labels=c("low Activity", "middle Activity", "high Acti
bn.data <- bn.prep.data(bn.dag, simpatdat, id, time_col)
bn.data <- na.omit(bn.data)
```

**Fit and Plot Results**

```
fitted.bn <- bn.fit.dag(bn.data, bn.dag, method="bayes")

library(bnlearn)
bn.fit.barchart(fitted.bn$Uncertain_Low_Back_Pain)
```




## G-Estimation

G-Estimation is used to adjust the analysis for causal inferences. For that, three different methods are implemented

### Load Data

```
load("data/simpatdat.rda")
```

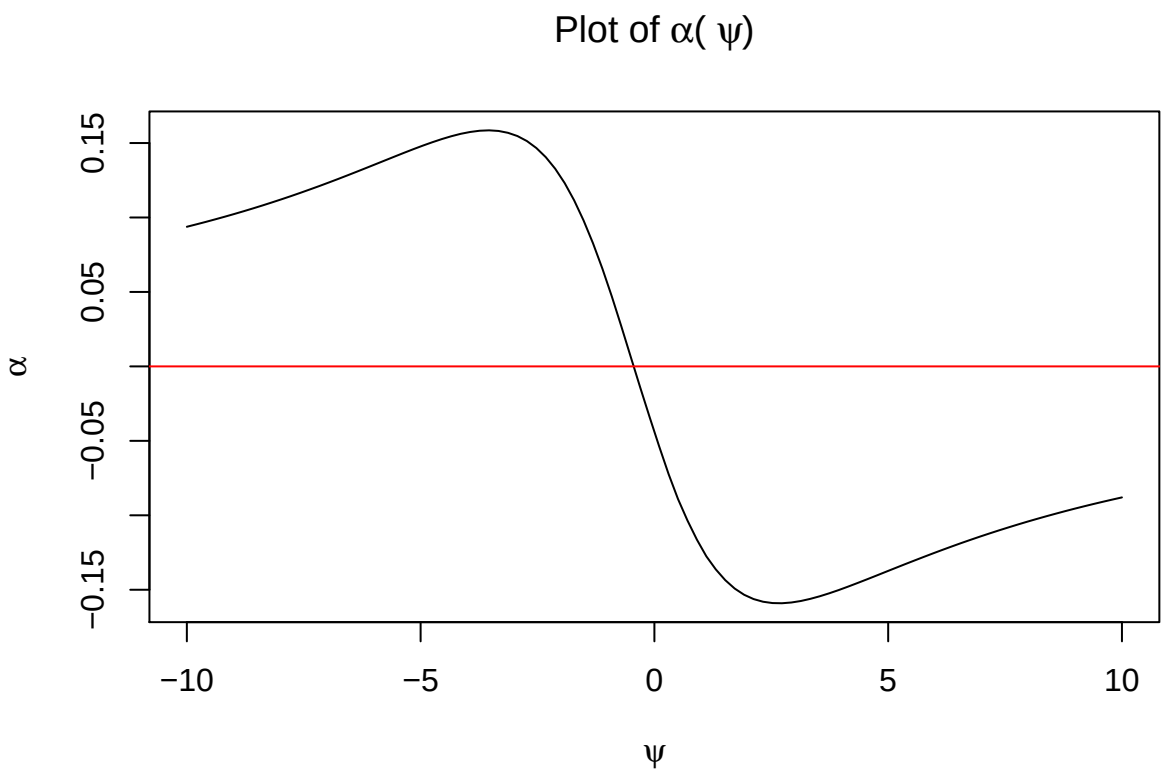
### Fit G-Estimation by Iteration

It iterates over several values for $\psi$ and returns a dataframe with $\psi$ and corresponding $\alpha$

```
outcome <- "Uncertain_Low_Back_Pain"
exposure <- "treatment"
confounder <- c("Activity")
id <- "patient_id"
df <- nofgest(simpatdat, outcome, exposure, confounder, id, method="iterate", steps=100, upper_bound_psi
```

This function is useful to plot a curve for $\alpha$ and $\psi$.

```
plot(df , type="l", main=expression(paste("Plot of ", alpha, "( ", psi,")")),
        xlab=expression(psi),
        ylab=expression(alpha))
# Add a second line
lines( c(-100,100),c(0,0), type = "l", col = "red")
```




**Fit G-Estimation by Recursive Mean**

This function approximate $\psi$ by an interval search.

```
outcome <- "Uncertain_Low_Back_Pain"
exposure <- "treatment"
confounder <- c("Activity")
id <- "patient_id"
nofgest(simpatdat, outcome, exposure, confounder, id, method="rec_mean")
```

```
## [1] "Converged! Optimal Psi:  -0.443288747439986"
## [1] "Number of iterations: 50"
```

```
## $upper_psi
## [1] -0.4432887
##
## $upper_beta
## [1] -6.491118e-16
##
## $lower_psi
## [1] -0.4432887
##
## $lower_beta
## [1] 1.881633e-16
##
## $n_it
## [1] 50
```

### Fit G-Estimation by Recursive Improved

This function approximate $\psi$ by an optimized interval search.

```
outcome <- "Uncertain_Low_Back_Pain"
exposure <- "treatment"
confounder <- c("Activity")
id <- "patient_id"
nofgest(simpatdat, outcome, exposure, confounder, id, method="rec")
```

```
## [1] "Converged! Optimal Psi:  -0.443288747439985"
## [1] "Number of iterations: 8"
```

```
## $upper_psi
## [1] -0.4432887
##
## $upper_beta
## [1] -1.980295e-16
##
## $lower_psi
## [1] -0.490945
##
## $lower_beta
## [1] 0.004872248
##
## $n_it
## [1] 8
```