README\_R\_functions

Hannah Allen

April 19, 2017

Table of Contents

# Function names listed below

# age.R

**Purpose**: Calculate age of patient in year, month, and day of life for any dataframe with patient's medical record number, date of birth, and date on therapy.

**What you need to know**: This function requires the lubridate package and uses the interval function in the lubridate package to get age in years and months. If you do not have lubridate installed then enter the following to install and library the package

install.packages("lubridate")  
library(lubridate)

If the data frame you are using does not have date of birth then you will need to add that in before using age.R. The subset\_dopkt.R function will add date of birth to your data frame if you have patient medical record number and date on therapy.

**How to use**: Paste the lines below into your R code and modify per criteria where table\_name\_calculated is the name of the data frame you want the day of life, month, and year calculated age values to show up in and table\_name is the name of the data frame you want to calculate day of life, month, and year age values from.

source("https://raw.githubusercontent.com/borumlab/functions/master/age.R")  
table\_name\_calculated <- age(table\_name)

# baseline\_bmi.R

**Purpose**: Subsets a population if they have a baseline BMI.

**What you need to know**: This function uses the table called anthropometrics\_id\_research and the column headers called "BMI\_DAY", 'DAY\_TYPE", and "MRNUMBER". This function will only create a list of medical record numbers for patients that have a BMI value in BMI\_DAY for DAY\_TYPE equal to 1 (meaning baseline).

**How to use**: Paste the lines below into your R code and modify per criteria where baseline\_list\_bmi is the name of the data frame with the list of medical record numbers that have bmi values at baseline.

source("https://raw.githubusercontent.com/borumlab/functions/master/baseline\_bmi.R")  
baseline\_list\_bmi <- baseline\_bmi()

# baseline\_foodomics.R

**Purpose**: Subsets a population if they have a baseline foodomics.

**What you need to know**: This function uses the table called foodomics\_daily\_diet\_research and the column headers called "MRNUMBER". This function will only create a list of medical record numbers for patients that have a medical record number in this table (meaning the patient has foodomics data).

**How to use**: Paste the lines below into your R code and modify per criteria where baseline\_list\_foodomics is the name of the data frame with the list of medical record numbers that have foodomics values at baseline.

source("https://raw.githubusercontent.com/borumlab/functions/master/baseline\_foodomcis.R")  
baseline\_list\_food <- baseline\_foodomics()

# baseline\_meds.R

**Purpose**: Subsets a population if they have a Med Load greater than a value x at baseline.

**What you need to know**: This function uses the table called med\_data\_id\_research and the column headers "MED\_LOAD\_DAY", "DAY\_TYPE", and "MRNUMBER". This function will only create a list of medical record numbers for patients that have a Med Load value in MED\_LOAD\_DAY for DAY\_TYPE equal to 1 (meaning baseline) and the Med Load value must be greater than x.

**How to use**: Paste the lines below into your R code and modify per criteria where baseline\_list\_meds is the name of the data frame with the list of medical record numbers that have Med Load values at baseline greater than x and x is the value you want Med Load to be greater than (e.g. 0).

source("https://raw.githubusercontent.com/borumlab/functions/master/baseline\_meds.R")  
baseline\_list\_meds <- baseline\_meds(x)

# baseline\_sz.R

**Purpose**: Subsets a population if they have a Seizure Load greater than a value x at baseline

**What you need to know**: This function uses the table called seizure\_data\_id\_research and the column headers "SEIZURE\_LOAD\_DAY", "DAY\_TYPE", and "MRNUMBER". This function will only create a list of medical record numbers for patients that have a Seizure Load value in SEIZURE\_LOAD\_DAY for DAY\_TYPE equal to 1 (meaning baseline) and the Seizure Load value must be greater than x.

**How to use**:Paste the lines below into your R code and modify per criteria where baseline\_list\_seizure is the name of the data frame with the list of medical record numbers that have Seizure Load values at baseline greater than x and x is the value you want Seizure Load to be greater than (e.g. 0).

source("https://raw.githubusercontent.com/borumlab/functions/master/baseline\_seizures.R")  
baseline\_list\_seizures <- baseline\_sz(x)

# clean\_anthros.R

**Purpose**: Retains anthropometric values calculated from organizations as recommended by CDC, ASPEN, AND

**What you need to know**: This function uses the table called anthropometrics\_id\_research and the column headers "DAY\_TYPE", "AGE\_MO", "AGE\_DOL", WHO\_HT\_Z\_DAY", "CDC\_HT\_Z\_DAY", "NHANES\_HT\_Z\_DAY", "WHO\_WT\_Z\_DAY", "CDC\_WT\_Z\_DAY", "NHANES\_WT\_Z\_DAY", "WHO\_BMI\_Z\_DAY", "CDC\_BMI\_Z\_DAY", "NHANES\_BMI\_Z\_DAY", "WHO\_UAC\_Z\_DAY", "NHANES\_UAC\_Z\_DAY", "WHO\_TSF\_Z\_DAY", "NHANES\_TSF\_Z\_DAY", "WHO\_SSF\_Z\_DAY", "NHANES\_SSF\_Z\_DAY". This function will only keep height, weight and bmi z-score fro WHO for children <2 years of age and CDC fo children >= 2 years of age up to 20 years of age, and NHANES after 20 years of age. This function will only keep UAC, TSF, and SSF from WHO if <=1856 days of life otherwise it will use NHANES data.

**How to use**: Paste the lines below into your R code and modify per criteria where anthropometrics\_id\_research is the name of the data frame where you want the "cleaned" anthropometrics to go.

source("https://raw.githubusercontent.com/borumlab/functions/master/clean\_anthros.R")  
anthropometrics\_id\_research <- clean\_anthros()

# deidentify.R

**Purpose**: Deidentify patient medical record numbers with a KGID and deletes other identify columns with date

**What you need to know**: This function uses the table kgid\_mrnumber\_link\_research and columns "MRNUMBER"" and "KGID" to deidentify any table that has MRNUMBER. This function also deletes the following identifier colums: PKT\_INITIATED\_DATE, DOB,DATE,MRNUMBER. This function cannot be used for the demographics sheet since there are additional identifiers in that sheet that are not met with this function.

**How to use**: Paste the lines below into your R code and modify per criteria where table\_name\_deidentify is the name of the data frame where you want the deidentified data to go and table\_name is the name of the data frame you want to deidentify.

source("https://raw.githubusercontent.com/borumlab/functions/master/deidentify.R")  
table\_name\_deidentify <- deidentify(table\_name)

# deidentify\_demo.R

**Purpose**: Deidentify patient information in the demogrpahics table

**What you need to know**: This function uses the table demographics\_id\_calculated and columns that include date, name, city, state, zip code, county.

**How to use**: Paste the lines below into your R code and modify per criteria where demographics\_id\_calculated is the name of the data frame where you want the deidentified data to go.

source("https://raw.githubusercontent.com/borumlab/functions/master/deidentify\_demo.R")  
demographics\_id\_calculated <- deidentify\_demo()

# desc\_stat.R

**Purpose**: To calculate descriptive statistics on a parameter within a population

**What you need to know**: This function requires the name of the data frame and column and time frame you wish to do descriptive statistics. This function calculates the following descriptive statistics: total number of values, mean, median, standard deviation, standard error, maximum, minimum, range, 5th percetnile, 25th percentile, 50th percentile, 75th percentile, 95th percentile, variance, and coefficient of variation. It will only calculate descriptive stats on one time frame at a time.

**How to use**:Paste the lines below into your R code and modify per criteria where table name is the name of the data frame you want the descriptive statistics to go, df is the name of the data frame you want to do the descriptive statistics on, column name is the name of the column within the data frame you want to do the descriptive statistics on, and DOPKT\_value is a numerical value you want to do descriptive statistics on (e.g. 30) you may also enter "all" and this function will do descriptive statistics on all data with the specified data fram and column name.

source("https://raw.githubusercontent.com/borumlab/functions/master/desc\_stat.R")  
table\_name <- desc\_stat(df, "column name", DOPKT\_value)

# ifc.R

**Purpose**: Subsets a population that signed the informed consent before x date

**What you need to know**: This function uses the table demographics\_id\_calculated and column header called "PKT\_PROSPECTIVE\_DATE". This will eliminate all patients that did not sign the informed consent before x date from the table.

**How to use**: Paste the lines below into your R code and modify per criteria where demographics\_id\_calculated is the name of the data frame where you want the patients that meet the criteria to go and mm/dd/yyyy is the date you want all patients to have signed the informed consent by.

source("https://raw.githubusercontent.com/borumlab/functions/master/ifc.R")  
demographics\_id\_calculated <- ifc('mm/dd/yyyy')

# irb.R

**Purpose**: Subsets a population that is on a x IRB protocol number

**What you need to know**: This function uses the table demographics\_id\_calculated and column header called "IRB\_NUMBER". This will eliminate all patients that are not on x IRB protocol number.

**How to use**: Paste the lines below into your R code and modify per criteria where demographics\_id\_calculated is the name of the data frame where you want the patients that meet the criteria to go and TYPE IRB# HERE is the IRB protocol number you want patients to be on.

source("https://raw.githubusercontent.com/borumlab/functions/master/irb.R")  
demographics\_id\_calculated <- irb("TYPE IRB# HERE")

# medscore.R

**Purpose**: Calculate med score for time intervals equal to x number of days

**What you need to know**: This function uses the tables med\_data\_id\_research and a data frame with the list of medical record numbers you want to use after going through inclusion and exclusion criteria. If you enter 30 as the time period value then med score will be calculated for time intervals equal to 30 days where 1 is the beginning of the time period and 30 is the end of the time period. Refer to manuscript x for med score calculations.

**How to use**:Paste the lines below into your R code and modify per criteria where table\_name is the name of the data frame where you want the calclation to go and time period value is the numerical value you want med score calulcated at specified intervals, and population list is the name of the data frame with the listo f medical record numbers that meet the inclusion and exclusion criter.

source("https://raw.githubusercontent.com/borumlab/functions/master/medscore.R")  
table\_name<-medscore(time period value, population list)

# mrlist.R

**Purpose**:

**What you need to know**:

**How to use**:

# outcomescore.R

**Purpose**:

**What you need to know**:

**How to use**:

# query\_patient\_pkt.R

**Purpose**:

**What you need to know**:

**How to use**:

# subset\_dopkt.R

**Purpose**:

**What you need to know**:

**How to use**:

# subset\_pop.R

**Purpose**:

**What you need to know**:

**How to use**:

# szscore.R

**Purpose**:

**What you need to know**:

**How to use**:

# total\_time\_on\_pkt.R

**Purpose**:

**What you need to know**:

**How to use**: