Feedback loops: Using data insight to stimulate data sharing across NHS Accident and Emergency departments.

B209460

03/07/2022

# Introduction

This script will load the the NHS England accident and emergency attendances and admissions (ae\_attendances) dataset from the NHSRdatasets package. It will briefly view, explore and tabulate the NHS England accident and emergency attendances and admissions (ae\_attendances) data set and save it to the project ‘RawData’ folder. It will then examine the four-hour waiting time target performance for England as a whole, selecting a subset of the variables needed. It will then partitioned the data subset into training and testing data and save them to the project working ‘Data’ folder, ready for downstream exploratory analysis. A data capture tool will be developed within Python, utilising a selection of widgets for user input. Details will be provided below as well as links to python scripting.

A full data dictionary will also be enclosed within this script.

Copyright statement: This script is the product of University of Edinburgh.

# Loading ae\_attendances from NHSRdatasets

## Background to data

The NHS England accident and emergency attendances and admissions (ae\_attendances). Is part of the NHSRdatasets package which has been created to support skills development in the NHS-R community. The dataset contains reported attendances, four-hour breaches and admissions for all A&E departments in England for the years 2016/17 through 2018/19 (Apr-Mar).

## Loading required packages

## Uploading the data

## Viewing the data and assessing for missing values

We will quickly view the data to understand its structure and assess the data quality.

The dataset consists of 12,765 rows of data and six columns with different classes. A date variable, period, two character variables (or factors), org\_code and type, and three numeric (double precision) variables, attendances, breaches and admissions.The data is also complete and so no further action required to handle missing data.

## Subsetting the data

Separating data into training and testing sets is vital for evaluating data collection and analysis tools. most of the data will be used for training, and a smaller portion of the data will be used for testing. To develop and evaluate our data capture tool, I will split the raw data into training and testing sets. We, therefore, need to add an index column to the raw data so we can link the partitioned data sets to the raw data if required in the future. We will use the rowid\_to\_column() function to convert row identities to a column named index.

## Formatting and saving raw data

To improve data legibility, The data has been formatted with a 1000 seperator for each of the numeric fields (attendances, breaches and admissions) and the date field has been formatted to “MMM/YY”.

Data is then saved into the RawData folder within the project working directory

## Selecting variables of interest, subsetting and saving data

We will now use the dplyr package select() function to select the required variables. The dplyr package is loaded by the tidyverse package, as one of its core components. dplyr provides a grammar of data manipulation, providing a consistent set of verbs that solve the most common data manipulation challenges

After, data will be saved into the RawData folder

## Exploring data subset

First lets summarise the data to get a better understanding of the variables we will use. I will use the dplyr glimpse function to review the variable types before finding the mean attendances and breaches for each org code, which I will sort in descending order to understand organisations are busiest. I will then derive the total time period of the observed data to better understand the summary statistics

The glimpse function has assisted in my understanding of the data structure. The index column is type integer, period is type date, bother attendance and breaches are of type double and organisation code is of type factor.

Through grouping and summarising I can also see that over a three year period, the busiest organisations were FRIMLEY HEALTH NHS FOUNDATION TRUST followed by CENTRAL LONDON COMMUNITY HEALTHCARE NHS TRUST and MID YORKSHIRE HOSPITALS NHS TRUST.

## Seperating into test and train datasets

I will seperate the raw dataset into test and train datasets for the development of my data capture tool. I will select 10 rows of data from my raw dataset to test the data capture tool. I will first calultae the proportion that these ten rows represent of my overall dataset and then use the createDataPartition() function from the caret package to splint our raw data into test and training data sets.The ‘set.seed()’ function is a random number generator, which is useful for creating random objects that can be reproduced. This will make sure that every time we run this script, we will partition the raw data into the same test and training data. After brief formatting, the two seperate datasets will then be saved as test and training.

# Data Capture tool

The data capture tool will be constructuted using a python script and will capture each of the variables described in the above subset from th ae\_attendances dataset from NHSRDatasets

The tool can be accessed via the Github repository, link below

## Consent

The tool will collect informed consent from participants using a widgit. The default option of which, is set to decline consent.

## Access

All scripts, Rawdata, Outputs and plans associated with this project can be found within the following Github repository.

[link to Github repository](https://github.com/B209460/B209460-Assessment)

# Data dictionary for test data

The data dictionary associated with the test data collected using the Data Capture tool is saved within the RawData folder and contrains descriptions of each variable.

The dictionary was produced using the R dataMeta package, the steps for this can be followed using R script “Creating a data dictionary.r”, which is saved within the project RScripts folder. It can also be re-created using the below code