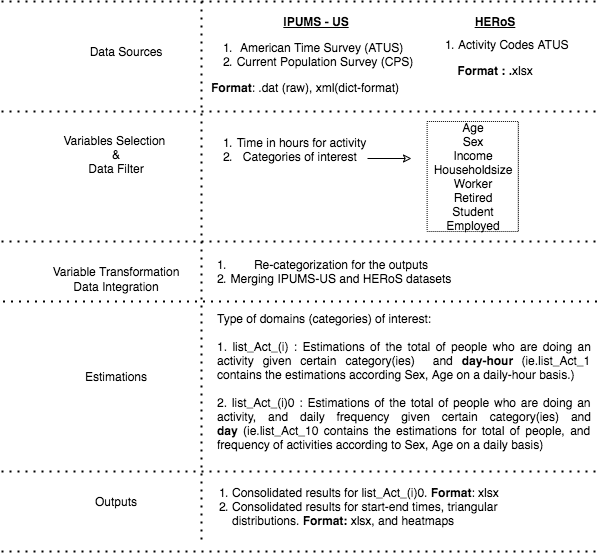
Documentation

Analysis of Daily Activities for Texas- US

## Introduction

This is a document that describes the structure of the code 01\_ATUS\_Analysis.R. The code intends to estimate several characteristics from the American Time Survey (ATUS) for 2019. The following Figure explains the general structure of this code.



The primary data sources are extracted from the IPUMS webpage (<https://www.atusdata.org/atus-action/variables/group>). The extract builder is developed to combine two survey data sources: the American Times Use Survey (ATUS) survey with the Current Population Survey (CPS). The CPS is a monthly survey in the US to monitor the main statistics about unemployment, labour force and population. The sample obtained for the ATUS survey is attached to the sample design for the CPS. In that sense, main characteristics of the population such as demographics, socioeconomic conditions, and labour can be selected and merged to the time use variables from the ATUS. Therefore, the final sample includes information about the household, person and activity.

There is a remark about the weights in the ATUS survey (WT06). The weights report the number of person-days in the calendar quarter that a single ATUS respondent represents. For instance, a respondent with a value of 5,000,000 represents approximately 55,000 people; it is 5000000 divided by 91 days in a quarter.

The variables selected from the ATUS survey are described in Table 1. The second file is used to processing the data is the list of activities of interests (Activity Codes ATUS.xslx). The file contains several sheets which describe the mapping of activities in the ATUS survey and the master\_codes for the HERoS model. Specifically, the Activities-ATUS, and Occupation\_ATUS sheets have additional columns with the HERoS categories that best match the description.

The following are some remarks about the ATUS survey:

* The weights in the ATUS survey (WT06). The weights report the number of person-days in the calendar quarter that a single ATUS respondent represents. For instance, a respondent with a value of 5,000,000 represents approximately 55,000 people; it is 5000000 divided by 91 days in a quarter.
* The survey is representative at the state level but not at the county or city level. Hence, a sub-sample for Texas (code 48) was filtered to find the estimations of the activities.

The following sections in this document describe how the code can be used and reproduced for further analysis.

## Data Sources: ATUS + ACTIVITY\_CODES (ACTIVITIES and OCCUPATIONS)

For reading the IPUMS data in R, it is necessary to download and load the package ipumsr. To read the data, you have to download the raw data in .DAT format and the variable dictionary in .xml format. Both of them are required for this step, and they need to have the same file names.

# # ATUS   
ddi <- read\_ipums\_ddi("input/atus\_00009.xml")  
df\_ATUS <- read\_ipums\_micro(ddi)  
  
# # ACTIVITY CODES   
df\_ActHeros <- read.xlsx2("input/Activity Codes ATUS.xlsx", sheetName = "Activities-ATUS")  
df\_OccupHeros <- read.xlsx2("input/Activity Codes ATUS.xlsx", sheetName = "Occupation\_ATUS", )  
head(df\_OccupHeros)

## Filtering the data (Texas), Variable Selection, and Transformation

By using the COUNTY column, the data for Texas con be obtained (the code is 48). The dataset is loaded in a tibble format with the code and label attached to the same object. Thus, the transformation corresponds to the transformation in a numeric format for merging purposes. The following chunk includes an example of it.

### Creating Texas data.frame  
df\_ATUS\_Texas <- df\_ATUS[grep("^48", df\_ATUS$COUNTY, ignore.case=F), ]  
### Changing the type of object inside R   
df\_ATUS\_Texas$ACTIVITY2 <- as.numeric(as.character(df\_ATUS\_Texas$ACTIVITY))

## 3) Integration and Recategorization:

Both data sources are integrated by selecting the corresponding columns in the Activity Codes file, and those individuals without any HERoS activity associated are removed from the data. Moreover, the version 2019 of this survey is also filtered at this step.

# # Filtering dataset to individuals who have any HERoS activity associated. Data for 2019   
df\_ATUS\_TexHrOS <- df\_ATUS\_Texas %>% filter(CategoryHEROS != "" & YEAR == 2019)

The variables Age and Income are recategorized as the Bureau of Statistics official reports. The Age variable is categorized by using the cohorts 14-24, 25-43, 44-63, 64-84, 85 or more. Income is categorized by using the limits: Less than $14,999, $15,000 to $24,999, $25,000 to $34,999, $35,000 to $49,999, $50,000 to $59,999, $60,000 to $74,999, $75,000 to $99,999, $100,000 to $149,999, $150,000 and over. For Household Size the categories are limited to 1, 2, 3, 4, and 5 or more.

## 4) Defining Estimation Objects

The code does not generate all the combinations of interest automatically. It means that you have to run it manually and adapt the estimation generator accordingly. There are two types of principal estimations: hourly-daily and daily-weekly. The definition of the combination of categories is given by the objects named form\_#. For instance:

# # character vector with the combinations of caracteristics of interest.   
form\_1 <- c("CohAge" , "SEX2")  
form\_2 <- c("SEX2")  
form\_6\_1 <- c("IncomeNew","Occup\_HEROS\_Worker")  
form\_6\_2 <- c("IncomeNew", "Occup\_HEROS\_Retired")

Objects with only one digit after the name, e.g. list\_Act\_1, correspond to estimates of the total number of people doing an activity given the hour of the day. The estimates are consolidated per person (EstimPP) through the function fun\_hourEstim. Objects with tens numbers after the name, e.g. list\_Act\_10, are estimates of the total of people doing an activity given the day of the week. In the case you want to add another category (e.g. ethnic origin), you will need to add another object to store the results and change the main loop with the intermediate objects accordingly.

The main loop goes through the vector of activities. For this case the total number of activities is 27, and it is defined by the subCategory at the Activity Codes Excel file. A brief description of the loop follows the following points:

1- Filter the activity from the main dataset, and obtained from the start time and stop time of activity the hour reported in the survey.

2- There are some activities which start on the day before and stop on the day after. For instance, the activity Sleeping needs to be analyzed by partitioning the sleeping dataset into two parts: the first corresponds to individuals who start and stop sleeping on the same day (START\_H <= STOP\_H), and the second is the subset of individuals who might start late at night and stop early morning next day (START\_H > STOP\_H).

In the case of frequency of activities, the approach was to sum up all the records for every activity separately given the day. For the start and stop times, the analysis follows the same principle of assigning the stop times on the day after.

3- The estimates are supported by the package survey, which is suitable when you have a sample design that defines the sample weights. The package functions allow estimating the total of people by adding the weights to the calculation. The same applies to the average frequency estimation of doing an activity. Moreover, the output includes the standard deviation of the estimate (se), which can be used afterwards for other purposes, for instance, interval estimations. The remark about the weights applies at this instance because the total of people estimated needs to be resized to the real population numbers.

## 5) Main Outputs:

Section 5.1 Probability and Average daily frequencies of doing an Activity. It is consolidated through the function fun\_EstimOutput. The main estimates are associated to the daily-weekly results described previously. The loop returns all the estimation objects that are inputs for this function. Said function rename and organized the estimates, and leave them in a data.frame format suitable for exporting the data in an Excel or CSV file. These files are separately exported according to one (single) and two (combined) characteristics.

Section 5.2) Heatmaps. It summarises the hourly-daily outputs in weekdays and weekends plots. The function fun\_plotHeatmapcan be used to generate the png files. It’s important to remark that the heatmaps sum up 100% per activity (per row).

Section 5.3 Obtaining the triangular distribution for each activity. It reads from the re-processed start and stop times columns the collapsed records and re-organized them into new columns. These files are separately exported according to one (single) and two (combined) characteristics.

**Note:** There is an additional estimation for the triangular distribution for Sleeping start times with more than five hours of duration to identify night sleeping patterns. In addition, you can find another category called “Sleeping>5Hrs” in the output data.

## Main Functions

fun\_HourEstim <- function(Xdat, X\_days = df\_days, X\_ACT = m\_ACT){  
 #' Rescaling estimations  
 #'  
 #' @description This function calculates the total number of people doing an activity  
 #' in the whole year. It divides the estimation using the weights WT06 from ATUS from 365 days.  
 #'  
 #' @param Xdat data.frame. It corresponds to the estimations returned from svyby.   
 #' @param X\_days data.frame. It correponds to table that describes the days of the week.  
 #' @param X\_ACT character. character with the name of the Activity   
 #' @return Two additional columns, total people (EstimPP), standard deviation of the estimation (sePP), and Activity   
}  
  
fun\_DayEstim <- function(Xdesign = surv\_des, Xformula , X\_days = df\_days, X\_ACT = m\_ACT){  
 #' Rescaling estimations  
 #'  
 #' @description This function calculates the total number of people doing an activity  
 #' in the whole year. It divides the estimation using the weights WT06 from ATUS from 365 days.  
 #'  
 #' @param Xdat data.frame. It corresponds to the estimations returned from svyby.   
 #' @param X\_days data.frame. It correponds to table that describes the days of the week.  
 #' @param X\_ACT character. character with the name of the Activity   
 #' @return Two additional columns, total people (EstimPP), standard deviation of the estimation (sePP), and Activity   
}  
  
fun\_EstimOutput <- function(Xdf\_AllActivities = df\_AllActivities,   
 Xlist\_Act,   
 Xformula = form\_1, Xdf\_days = df\_days,   
 Xv\_activs = v\_activs,   
 pathFile = "./outputs/files",   
 colNames = Xformula){  
 #' Consolidated output for all the activities (Probability and Frequency)  
 #'  
 #' @description This function calculates the estimation of the probability and frequency of diong a certain activity.   
 #'  
 #' @param Xdf\_AllActivities data.frame. It contains individuals in the rows, and in the columns characteristics such as SEX, Age, Income, etc.   
 #' and columns associated to each activity that indicates if the activity was done by each individual.   
 #' @param Xlist\_Act list. It contains all the estimations for all the activities.   
 #' @param Xformula character vector. It repressents each combination of characteristics i.e. c("Age", "Sex")  
 #' @param Xdf\_days data.frame. It correponds to table that describes the days of the week.  
 #' @param Xv\_activs character vector. It has all the activities of interest.  
 #' @param pathFile character. A path to the output folder.   
 #' @param colNames character vector. It has the names to rename the columns for the characteristics of interest.   
 #' @return list, csv files. List with the estimation of the probability and frequency of doing a certain activity given the profile and day of the week  
 #'   
}  
  
fun\_plotHeatmap <- function(Xlist\_Act, Xformula, t\_week = week\_day, week\_title = "WeekDay"){  
 #' Heatmap plot   
 #'  
 #' @description This function generates the heatmaps for daily activities.  
 #'  
 #' @param Xlist\_Act list. It contains all the estimations for all the activities.   
 #' @param Xformula character vector. It has each combination of characteristics i.e. c("Age", "Sex")  
 #' @param t\_week character vector. It has all the weekdays or weekend names.  
 #' @param week\_title character. It is either WeekDay if t\_week are weekday, or WeekEnd if t\_week is weekend.  
 #' @return plots in png format. By default to the output folder.   
}