Doc\_ATUS\_Analysis

## Introduction

This document explains the analysis performed on the American Time Survey (ATUS) for 2019 to find insights about the main citizens’ daily activities in Harris county. Firstly, The approach includes to estimate the probability of doing an activity on a given day and hour, and the estimated start and end times of doing said activity for a given population profile. Secondly, the approach takes into account the validation of results by creating a simulated population based on the Census data to have a reliable comparison between original estimation and simulation. The Figure 1 shows the two main steps to the general approach.

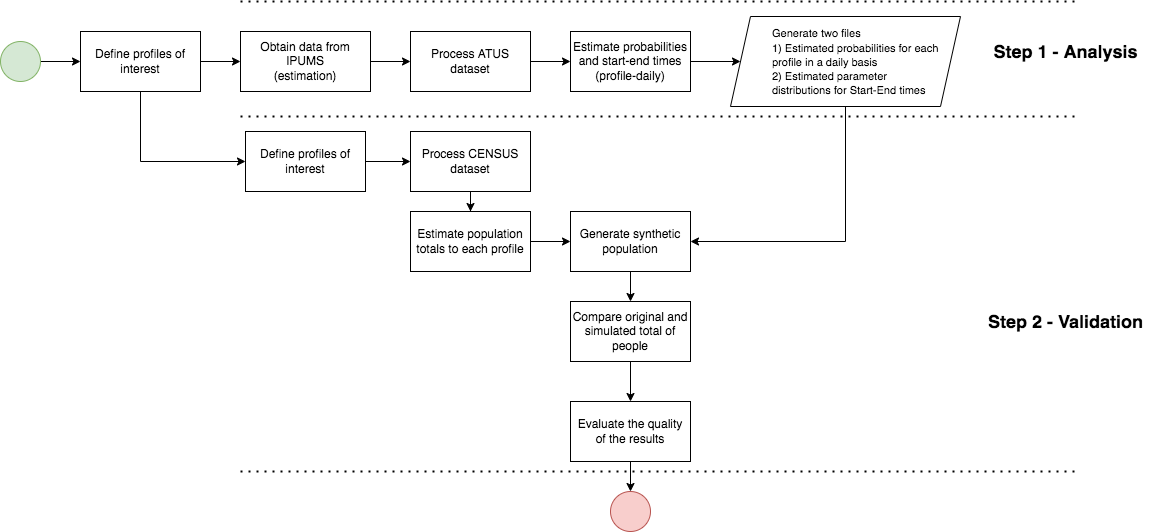


Figure 1: General approach

## Step 1 - Analysis

### Flow chart explaining the analysis of results

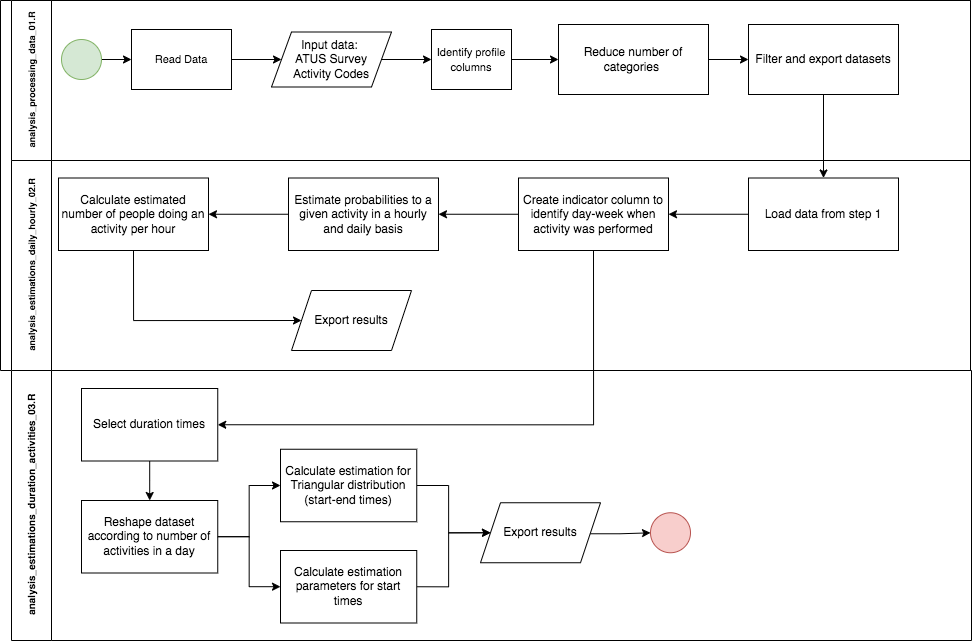


Figure 1: General approach

### Data sources

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 (See Table 1). Therefore, the final sample includes information about the household, person and activity.

—-Include TABLE with names profiles—–

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 obtained in order to get representative estimations at this level. The results for Texas can be further used at a lower scale to understand the activities patterns for Harris.

There is an important remark about using 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 second file the list of activities of interests (Activity Codes ATUS.xslx). The purpose of this file is to create a correspondence between the activities in both sources. In that sense, the file contains the mapping of activities in the ATUS survey and the master\_codes for the HERoS model.

### General explanation

The analysis is divided into three steps: data processing, estimating hourly-daily probabilities of a given activity and estimating parameters of start times.

The data processing step includes the file analysis\_processing\_data\_01.R. The process starts by reading the ATUS, activities and occupation files. Then, according to the Texas code (48), this information is merged and filtered . The next step includes the re-categorization of values to reduce the number of categories and improve the precision of the estimations for each profile.

Estimating hourly-daily probabilities start by identifying the hour and day of the week on which the reported activity is performed. The following part of the function performs this task by differentiating activities with start and end times on the same day and two consecutive days (reported end time on a day after activity starts). For instance, if the reported start time is 8:00 and the end time is 14:00, the function identifies a first case scenario where the activity is performed during the day. The second case corresponds to the start time at 18:00 and the end time at 5:00, which means that the day after the activity ended. Both cases are considered during the programming stage to understand the patterns of activities better.

df\_indicator\_consolidated <- fun\_indicator\_activity(df\_activity = df\_activity,   
 indicator\_hour\_activity = indicator\_hour\_activity)

where df\_activity is the data.frame for the corresponding iteration and indicator\_hour\_activity is a list of matrices to be filled accordingly.

The next stage includes calculating the total number of people who performed the activity for each profile. It corresponds to the numerator of the estimated probability. It is obtained by using the sampling weights to calculate totals on a given hour and day. This part of the function calculates for both cases:

for(ll in 1:nrow(df\_combined\_target\_profiles)){  
 profile\_iteration <- as.character(unlist(df\_combined\_target\_profiles[ll, 1:2], use.names = FALSE))  
 profile\_type <- as.character(unlist(df\_combined\_target\_profiles[ll, 3], use.names = FALSE))  
 estimates\_hourly <- fun\_HourEstim(data = df\_indicator\_consolidated\_long, profile = profile\_iteration, activity=activity)  
 estimates\_daily <- fun\_DayEstim(data = df\_indicator\_consolidated, profile = profile\_iteration, activity=activity)  
   
 list\_estimates\_hourly[[activity]][[profile\_type]][[ll]] <- estimates\_hourly  
 list\_estimates\_daily[[activity]][[profile\_type]][[ll]] <- estimates\_daily  
 }

Due to the sampling design is not representative per hour or day, the daily number of people is not constant and variates according to the combination profile, day, hour, and set of activities reported. Given this constraint, the denominator is obtained through the number of people who said sleeping as an activity, avoiding having profiles of people with lower probabilities of sleeping. It corresponds to the section 3 in the code. The following part of the code calculates the probability

estim\_activity\_probability <- estim\_activity\_probability %>% mutate(Probability = EstimPP/EstimPP\_Total)  
 estim\_activity\_probability$Activity <- activity  
 estim\_activity\_probability$Probability <- with(estim\_activity\_probability,   
 ifelse(is.na(Probability), 0 , Probability))  
 estim\_activity\_probability$nTimesDay <- with(estim\_activity\_probability,   
 ifelse(is.na(nTimesDay), 0 , nTimesDay))

The fourth section calculates the probability of performing an activity per hour and day, which is used at the validation step to compare simulated and the original number of people.

The third step in the analysis aims to calculate the empirical distribution of the start times. The code analysis\_estimations\_duration\_activities\_03.R calculates the empirical distribution as the probability for specific start times in a typical day. The following part of the code calculates the probabilities, as well as, the estimated average and standard deviation of the duration.

#### Step 2 - Validation

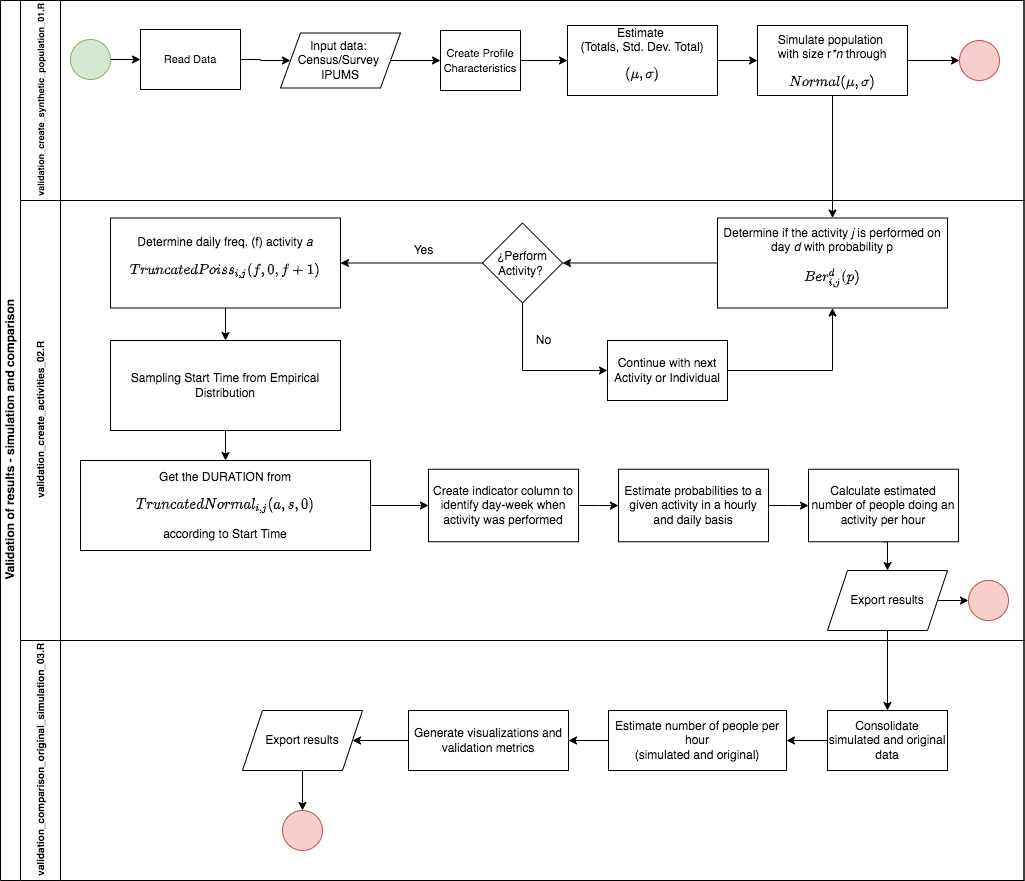


Figure 3: General approach

### Data sources

The main data source for this step is the US census data. The data is obtained from the IPUMS webpage (<https://usa.ipums.org/usa-action/variables/group>). The process to select the data follows the list of variables of interests related to the profiles previously defined at the Step 1. Due to the Census data does not captures the same information as ATUS, a set of the most similar variables were chosen and categorized accordingly (See Table 2):

—- Include table —-

The first step (validation\_create\_synthetic\_population\_01.R) reads the census data and calculate the estimated total number of people to each profile and the standard deviation of the total. This part of the code uses the survey package to generate the estimations per profile

svy\_Texas <- svydesign( id =~ 1 , data = df\_ATUS\_Texas\_Heros , weights =~ PERWT)  
est\_Pop <- svyby(~People, profile\_formula, svy\_Texas, na.rm = T, svytotal)

Moreover, the est\_Pop output is used as an input to a normal distribution, generating the total number of people in the simulation. For computational limitations, a random sample of 0.01% of the simulated population is generated. This parameter can change according to the local computational capacity. Therefore, this code recreates the same structure as the ATUS survey for the simulated population in order to have the required inputs for the analysis functions and the probabilities of doing an activity. Each simulated population is stored in the list df\_GeneratePopulation\_List.

The second step (validation\_create\_activities\_02.R) loads the simulated population, probabilities per hour and day, and estimated start times and duration. The code goes through the set of profiles to obtain a complete dataset with all the activities assigned in a given hour-day. The following is the explanation of each step inside the main loop: The iteration starts with another loop that goes through the set of activities. Let’s assume the activity k. 1- The column Indicator\_Activity identifies if the activity k is performed with probability p. Obtained from the dataset Weekly\_Probability\_Activity\_Combined. The column Frequency\_Activity captures the random frequency according to a truncated Poisson distribution with average nTimesDay and truncated interval (0, nTimesDay +1) To transform the dataset into a long dataset according to the frequency. Therefore, if a person performs an activity two times a day, the new dataset contains two rows for each time. The get\_start\_time function takes the dataset and returns a combination of start times and duration. The function first obtains the start time from the empirical distribution and then the duration according to the selected start time. 6- The stop time is obtained by summing the start time and duration.

The 2.2 section of the code reuses step 2 from the analysis section to calculate hour and daily probabilities. In particular, the function fun\_indicator\_activity is applied again to identify the day hours in which the activity is performed. This last step is the main output of this code and contains all the information required to compare the simulated and original population.

Secondly, due to the absolute number of people in the simulation is not comparable with the original, the comparison uses the total from the simulation and the proportion from the original. This step assumes that the proportion (or the probability) from the original data is a reliable estimation. Therefore, for the comparison the total number of people in a given hour follows the equation:

where T is the number of simulated people in a given hour and P is the proportion of people doing an activity from the original data. At the code, the following chunk defines the calculation:

df\_estimate\_hourly\_original <- df\_estimate\_hourly\_original %>%  
 mutate(Total\_People\_Hour\_Activity\_Original = Total\_People\_Hour\_Simulation \* probability\_hourly)

Finally, the visualization includes scatterplots to validate the simulation. And the function fun\_MSE returns metrics, e.g. Mean-Squared-Error or the Standard Residual, which are both adapted to this case.